Solar Energy 158 (2017) 462–473

Contents lists available at ScienceDirect

Solar Energy

journal homepage: www.elsevier.com/locate/solener

Experimental validation and comparison of direct solar shading calculations within building energy simulation tools: Polygon clipping and pixel counting techniques

Ana Paula de Almeida Rocha[a,*], Ricardo C.L.F. Oliveira[b], Nathan Mendes[a]

[a] *Pontifical Catholic University of Parana, 80215-901 Curitiba, PR, Brazil*
[b] *School of Electrical and Computer Engineering, University of Campinas – UNICAMP, 13083-872 Campinas, SP, Brazil*

ARTICLE INFO

Keywords:
Solar energy
Sunlit surface fraction
Pixel counting technique
Polygon clipping technique
Experimental validation

ABSTRACT

This paper presents an experimental validation procedure of two solar shading calculation techniques - pixel counting (PxC) and polygon clipping (PgC) - and an inter-software comparison to highlight the capabilities and efficiency of each solar shading calculation method. For the first purpose, digital images were taken from the surfaces of small-scale mock-ups specially constructed to generate experimental data for validating simulation results obtained by cases using three different tools: EnergyPlus (PgC based), Shading II SketchUp plug-in (PxC based) and Domus (PxC based). This first task has shown, for prototypes with simple geometries, that all techniques present results in good agreement with the experimental data. However, for a prototype with a hollowed shading device, the PgC-based technique produced results far from the experimental ones since it is not appropriated to simulate multi-hollowed polygons. In order to further explore the capabilities of the two shading calculation techniques, an inter-software comparison has also been carried out for a complex case, considering different building shading solutions, including non-planar trees. The results, in general, have shown that the PxC technique is not limited to geometrical complexities and leads to an accurate and a very fast assessment of sunlit surface fraction. It has also been shown a difference as high as 10 times on the prediction of a daily-integrated solar heat gain by using the two different techniques.

1. Introduction

Solar direct gain control is an important aspect to reduce building heat gain and cooling requirements and to enhance the daylight quality of indoor rooms. A possibility to accomplish this task is by employing, for instance, simple forms, such as overhangs, awnings and louvers, or by complex geometries, such as screens and trees (Freewan, 2014; Kirimtat et al., 2016). With the evolution of parametric CAD systems and digital fabrication, the use of sophisticated shading geometries has become more common by architects and designers. Considering that nowadays computer simulations are used on the design of many energy efficient strategies, including appropriate shading devices, it is important that building energy simulation (BES) software find a way to deal with this evolution of architectural forms. In particular, shading calculation methods should be able to analyze a wide range of geometric configurations.

The most common methods for calculating the sunlit area on exterior surfaces in BES tools are the trigonometric and, projection and clipping operations. The first one uses trigonometric relationships to predict the areas of the cast shadows, however, it is limited to a few simple shading devices, such as overhangs and fins (Szokolay, 2008; Cascone et al., 2011b). Polygon clipping (PgC) methods (Weiler and Atherton, 1977; Blinn and Newell, 1978; Vatti, 1992) use projection and successive clipping of polygons, allowing the simulation of more complex geometries when compared to the trigonometric technique. Although the PgC based methods are used by many simulation programs - such as ESP-r, BLAST, DOE-2, TRNSYS and EnergyPlus - they have some limitations regarding the type and number of polygons. For example, in EnergyPlus, two methods of PgC are currently in use: Convex Weiler-Atherton (CWA) and Sutherland-Hodgman (SH). While the original model of the Weiler-Atherton enables the clipping of a concave polygon with holes, the version implemented in EnergyPlus does not support concave shadowing surfaces or holes. This implementation is only accurate if both casting and receiving surfaces are convex, *i.e.*, the thermal zone cannot have interior angles higher than 180°. On the other hand, the Sutherland-Hodgman algorithm works better with non-convex receiving surfaces, which means that the exterior walls surfaces may be concave, however, the hollowed surfaces

---

[*] Corresponding author.
E-mail addresses: anarocha4arq@gmail.com (A.P.d.A. Rocha), ricfow@dt.fee.unicamp.br (R.C.L.F. Oliveira), nathan.mendes@pucpr.br (N. Mendes).

http://dx.doi.org/10.1016/j.solener.2017.10.011
Received 8 November 2016; Received in revised form 19 June 2017; Accepted 3 October 2017
0038-092X/

cannot be still evaluated (EnergyPlus, 2013). Other limitation is related to the number of projected shading polygons on the receiving surface, which is dependent on the complexity of the shading elements. Sometimes, the shadow shapes are too complex, with multiple overlaps and intersections of projected polygons so that the use of the clipping method becomes impractical (Maestre et al., 2013). Therefore, the building geometry is modeled in a more simplified form in order to generate compatible geometric models with the simulation tool, which can lead to large differences between simulation and reality (Jones and Greenberg, 2011). Additionally, the computational time is strongly dependent on the desired accuracy. Calculating sunlit area for each day is more accurate but it is also more time consuming (Maestre et al., 2015), thus, BES tools commonly recommend to use a greater time period for the calculations to yield faster results (EnergyPlus, 2013).

Although some efforts have been made to increase the accuracy of the methods based on projection and clipping operations (Liu et al., 2007; Cascone et al., 2011a; Maestre et al., 2013), another approach for direct solar shading calculations, called pixel counting (PxC) technique, has become competitive with the development of software and hardware associated with computer graphics. It renders the building's scene using orthogonal projection from the vantage point of the Sun and, for each time step, calculates the number of visible pixels belonging to each surface (Yezioro and Shaviv, 1994; Niewienda and Heidt, 1996; Shaviv and Yezioro, 1997). Although the technique is not exact due to the effect of pixellation, Jones et al. (2012) have already shown that it can be successfully used to calculate the sunlit fraction on the facade of a building. They have compared results of projected sunlit surface fraction (PSSF) obtained by PxC with analytical solutions. For all cases, the incident beam radiation calculated using PxC was within 1% of the analytical value. In addition, the technique had no difficulty to deal with concave or rounded surfaces, and even worked for surfaces with complex double curvature or hollowed surfaces. Despite all those advantages, the PxC has not yet been evaluated on a BES software. A current tool that uses this approach is the Shading II SketchUp plug-in[1] (Yezioro and Shaviv, 1994), which calculates the sunlit fraction on exterior surfaces but has no engine for building energy simulation. In order to enhance the design of more complex shading devices, the PxC approach has been implemented using OpenGL in the building energy simulation program Domus (Mendes et al., 2003), for the calculation of the sunlit fraction and direct solar energy on both external and internal surfaces.

In this context, this paper aims first to experimentally validate results from the two solar shading prediction algorithms (PgC and PxC), showing that the solar shading calculation techniques produce accurate results, close to reality. Then, to highlight the main differences among the techniques, some features are compared, such as their simulation capability for complex geometries and their computational cost.

2. Solar shading simulation tools

This section presents some features related to the solar shading calculations of the three software investigated in this work: EnergyPlus, Shading II SketchUp plug-in and Domus. The focus is on the methods for solar shading calculations.

2.1. EnergyPlus BES software

EnergyPlus[2] is a modular and structured code developed on the basis of the most popular features of BLAST and DOE-2 (Crawley et al., 2001). It is capable of modelling both energy consumption - for heating, cooling, ventilation, lighting and plug and process loads - and water use in buildings (EnergyPlus, 2016).

For the solar shading calculations, this worldwide used program has a shading module based on the BLAST and TARP shadowing algorithms, which includes coordinate transformation, shadow overlap and PgC methods. To start the calculations, all building coordinates undergo a geometric transformation (rotation of all elements and the Sun) such that the surfaces for which the shadow is cast - receiving polygon (RP) - and surfaces casting the shadows on the receiving surface - shading polygon (SP) - are coplanar. After, portions of SPs that are submerged in the RP are eliminated and the other polygons are projected following the direction of the solar beam. Then, clipping operations are performed between the shadow and the receiving surface polygons and the sunlit area is calculated from the final polygon.

As already mentioned, two PgC methods are implemented in EnergyPlus: Convex Weiler-Atherton and Sutherland-Hodgman. Both methods determine the intersecting points of the boundary of both shadow and receiving polygons, by finding all vertices of the overlap. The vertices are transformed back into Cartesian coordinates and the area is computed. Considering a closed, planar polygon of n sequential vertices $(x_1,y_1),(x_2,y_2),\dots,(x_n,y_n)$, its area is given by:

$$Area = \frac{1}{2}\sum_{i=1}^{n}(x_i y_{i+1} - x_{i+1} y_i) \tag{1}$$

where $(x_{n+1},y_{n+1}) = (x_1,y_1)$

For the Sun position calculation, the values of the solar declination angle and the equation of time are based on Astronomical Algorithms (Meeus (1999), *apud* EnergyPlus (2013)). EnergyPlus performs solar shading calculation during the simulation process and allows the frequency of shading assessments to vary. As default, a simulation frequency of one day for each 20 days with a 15-min time step is used.

2.2. Shading II SketchUp plug-in

Yezioro and Shaviv (1994) developed a SketchUp plug-in, called Shading,[3] for shadow analysis for any given design, using the orthogonal projection and the pixel counting technique. However, their results are not connected to any BES software.

For the solar shading calculations, the first step is to draw on the computer screen an orthogonal projection of the building from the Sun's point of view at a particular geographic location and at a particular time (month and hour). Then, the sunlit area of the image is calculated by counting pixels according to their color, using a bitmap technique (Yezioro and Shaviv, 1994).

Double buffer is applied to define the sunlit fraction of a surface. In computer graphics, double buffer can be used to perform smooth animations and to avoid the flickering caused by showing incomplete images. For that, the computer stores two pictures: the first one, stored in the front buffer, is displayed on the screen and the other one, stored in the back buffer, is hidden. The last picture is displayed only after the complete generation of the new image. Thus, for calculating sunlit fraction of each building surface, two orthogonal projections from the Sun's point of view are created: one including shading devices and obstructions and the other without. The first one is shown in the front buffer and seen by the user, and, the second, is drawn in the back buffer. As the pixels of the shaded area get the color of the shading elements, these pixels are not counted. The number of pixels belonging to each surface is computed for both images and the ratio A_{sunlit}/A_{total} are founded for each visible surface.

2.3. Domus BES software

Domus software[4] is a whole-building simulation tool for analysis of both thermal comfort and energy use. It has been developed to model

---

[1] http://ayezioro.technion.ac.il/Downloads/ShadingII/index.php

[2] https://www.energyplus.net/

[3] http://ayezioro.technion.ac.il/Downloads/ShadingII/index.php

[4] http://www.domus.pucpr.br/