PLEA 2017 EDINBURGH

Design to Thrive

Towards Dynamic Real-Time Daylight Simulation

Dr Andrew Marsh[1] and Dr Spyros Stravoravdis[2]

[1] PerformativeDesign.com, andrew@performativedesign.com;

[2] Department of Built Environment, University of Greenwich, UK, s.stravoravdis@greenwich.ac.uk

Abstract: *Real-time visual feedback can greatly assist the process of identifying and understanding complex cause and effect relationships. Whilst not all physical processes in the performance analysis of buildings can be simulated in real-time, some can. This paper introduces a web-based daylight simulation tool that uses variants of the daylight coefficients and split-flux methods implemented on the GPU to calculate the spatial distribution of daylight factors across a simple rectangular room. By combining real-time simulation with modern games technology, it is possible to create a highly visual and interactive design environment that allows all the governing parameters of internal daylighting - including room geometry - to be manipulated by the user in real-time, with detailed contextual results updating dynamically with each interactive change. The purpose of this tool is primarily educational, allowing users to gain a comprehensive understanding of the relationships between room dimensions, window placement and daylight distribution through a process of deliberate investigative play. A detailed parametric comparison with Radiance simulations resulted in a small modifier to the implemented method that produces a robust correlation which the authors argue makes this tool a valuable educational resource with potential applications in early design decision-making.*

Keywords: Daylight, GPU, Real-time, simulation

Introduction

This paper describes a prototype web-based tool that utilises WebGL and custom shaders to dynamically calculate the spatial distribution of daylight within a rectangular room and update it in real-time as the user interactively manipulates room dimensions, surface properties, window sizes and their positions within the envelope.

The choice of daylight calculation method and implementation details are described as well as the process of comparing results against those of the same room configurations simulated in Radiance. Whilst validation is important in order to show that the approach taken does not yield misleading results, the main aim of the comparative analysis was to gain a better understanding of where the implemented method diverges and why.

This work is part of on-going research into the development of design tools that utilise analytical calculations performed on the graphics-processing unit (GPU) to create highly dynamic and interactive building simulation environments. Thus, the primary goal is to undertake analysis that is fast enough to provide dynamic visual feedback during model manipulations and accurate enough for that feedback to be meaningful and useful.

The Spatial Model

The geometric model used in this prototype tool is limited to a simple rectangular room with no internal obstructions, but with adjustable wall thickness and any number of rectangular windows in any wall. The width and depth of window frames are also customisable and can include any number of mullions and transoms. Some example screenshots of the room model and its controls are shown in Figure 1.

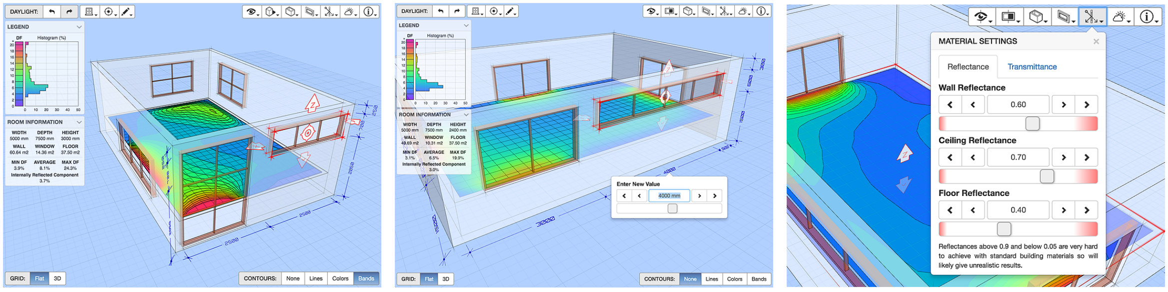

Figure 1: Screenshot of the simple rectangular room model used in the tool and some of the controls for interactively manipulating it.

The decision to use such a simple model in the initial implementation was made because an axially-aligned rectangular plan shape is well suited to simplified daylight analysis methods (Winkelmann and Selkowitz, 1985), is broadly representative of a wide range of actual design conditions likely to be faced by designers and greatly simplifies the development of interactive user manipulation techniques for the room and its apertures.

It is intended that future tools will better accommodate the detailed geometry and specific materiality of actual design spaces and allow CAD/BIM data to be directly imported. However, as a proof-of-concept, this tool is a necessary first step towards that goal.

Daylight Calculation Method

There are basically three methods for simulating the spatial distribution of diffuse illuminance within a room - these being the split-flux, radiosity and ray-tracing methods. As the split-flux method is widely used for daylight estimation at early design stage, and is the default daylight calculation method used within EnergyPlus (DOE, 2017), it was the method selected for implementation within this work. The very first implementation of the split-flux method was actually based on the BRE Daylight Factor Protractor approach (BRE, 1986). This approach was particularly interesting because the values in each protractor already account for the effects of angular-dependent transmission through glass, the luminous distribution of an overcast sky and the conversion of aperture solid angle to daylight factor contribution. Thus, by digitising the daylight factor protractor and converting that data to high-resolution look-up tables, a significant number of quite complex calculations can be sidestepped.

Using the GPU

The look-up tables for daylight factor are easily encoded as two-dimensional textures. The GPU was then used to calculate the plan and sectional angles of each aperture from each point on the spatial daylight grid. These angles were used to linearly interpolate between looked-up texture values to derive the sky component of the daylight factor at each point.

Straightforward trigonometric calculations such as these are trivial for most modern GPUs and they are able to process grid points in parallel. Quantifying the exact capacity for parallelisation on any GPU is a fraught process as there are many interdependent criteria as well as different chip architectures and therefore different nomenclatures. However as a rough comparative measure, nVidia GeForce GPU chips range from around 700 parallel units/cores to as many as 4000 at the high end. AMD Radeon GPUs range from around

1,000 to nearly 2,500 whilst the Intel HD 500 series range from 110 to 1100 (TechPowerUp, 2017). Even the GPUs found in most phones and tablets are able to process more than a hundred points at once. At 250mm centres, a 6x10m room would contain around 960 grid points over the work-plane.

Profiling the execution of the tool during interactive user manipulation of a 6x10m room shows that a full recalculation and update of the model and user interface takes around 12-15 milliseconds on a standard 2014 MacBook Pro with an i7 CPU, and around 28-36 milliseconds on an iPad Air and Galaxy Note 4 phone. The actual daylight calculations across the grid take up less than 3% of that time on the MacBook Pro and only 4% on the iPad and Galaxy Note. More importantly, the majority of that 3 to 4% is actually spent encoding the updated room and aperture data into shader uniforms on the CPU and then decoding the GPU results buffer back to grid array values. The remaining 96 to 97% of that CPU and GPU time is spent regenerating and re-tessellating the polygons forming room geometry and daylight grid contours, as well as updating the various graphical user interface elements.

Thus, this research has found that this approach to daylighting in such a simple model can be so highly optimised that it becomes an insignificant component of the frame-by-frame workload required to visualise changes in real-time.

Whilst this was something of a surprise, even more surprising was the fact that the same approach can be coded entirely in JavaScript, without using the GPU at all, and the execution times are very similar. The dynamic code optimisation capabilities of JavaScript compilers in most modern browsers means that highly repetitive numeric calculations are quickly optimised to near native speeds. Thus the time spent by the CPU decoding the GPU results is roughly equivalent to it actually performing all the calculations itself.

All of this essentially meant that there was significant capacity for additional computation, on both the GPU or CPU, before the frame rates of dynamic updates were noticeably impacted.

Switching to Daylight Coefficients

This additional computational capacity allowed some of the limitations of the protractor-based method to be reconsidered. As protractors have only been published for a very specific set of sky luminance distributions and glazing types, handling the full 16 CIE Standard General Sky types for climate based daylight modelling (CBDM) and accommodating more complex glazing and shading systems requires a different approach.

Recent work on CBDM has used hemispheric subdivision techniques to simulate skylight and sunlight as a series of individual sky patches, each with varying luminance (Bourgeois et al, 2008). Replacing the protractor-based look-up tables with an array of sky patches and then using the GPU to determine which patches are visible from each grid point has several advantages:

- The horizontal and vertical angles of each sky patch can be easily encoded as a two dimensional texture and sent to the GPU in the same way as look-up tables,
- Patch visibility can be cached for each grid point, along with additional information such as the aperture through which it is visible or a reference to any internal or external obstructing surface(s),
- The sky dome is considered to be sufficiently distant that the angles of each patch are the same for each grid point, meaning that angles of transmission through

glazing or surface incidence do not need to be cached as they can be easily determined by look-up as and when required,

- Knowing surface intersection and/or aperture transmission angles for each patch allows more complex glazing, shading and light redirection systems to be analysed and even BRDF/BSDF functions incorporated,
- Cached aperture and obstruction data can be used to further optimise interactive manipulations - such as changing window frame configurations or resizing an aperture which requires only those patches previously passing through that aperture or obstructed by the containing wall to be recalculated, and
- Having each grid point check the same number of patches and therefore reference the same texture coordinates at the same time and in the same order is a process particularly suitable for the single instruction, multiple data (SIMD) architectures of almost all modern GPUs, making it fast and efficient even when several thousand sky patches are used.

Comparison of Results

Both the BRE and EnergyPlus versions of the split-flux method are based on early work by Lynes (1979) on the derivation of a daylight factor formula for side-lit rectangular rooms. Some researchers have found that Lynes' formula and its close derivatives have a useful correlation with both measured daylight data (Crisp, Littlefair, 1984) and Radiance simulations of the same spaces (Reinhart, 2010). Detailed studies by Versage et al. (2010) and Yoon et al (2014) indicate that, for south oriented windows, the split-flux method predicts higher illuminances than the radiosity and ray-tracing algorithms as the distance from windows increases. However, follow-up studies by Yoon et al (2014) comparing other orientations indicate that this is not always the case and that much depends on the settings used in each radiosity or ray-tracing run.

All of these studies comparing the split-flux with other methods have focused on absolute accuracy and differences in individual values. However, the aim of this tool is to identity and illustrate relationships, so relative accuracy when calculation parameters change is of more importance. For example, does predicted daylight fall by the same percentage in all methods when the window area is halved? If the relative trends match and overall correlation is high, then the results can still provide meaningful and useful insight and design guidance even if the absolute values do not exactly match.

Comparison with Radiance

Radiance (Ward and Rubinstein, 1988) is a widely used and highly validated daylight simulation program (Mardeljevic, 1997) developed by Greg Ward and Lawrence Berkeley National Laboratories. It is based on a variant of the ray-tracing method and is used at all levels of lighting and daylighting design as the reference simulation tool.

To investigate both the absolute and relative accuracy, as well as the overall correlation, results from the GPU-based method were compared with those from spaces with exactly the same configuration simulated in Radiance. As comprehensive datasets of measured light levels in real rooms are limited (Osborne and Donn, 2011), and those datasets mainly exist as a result of having been used to validate Radiance (Mardaljevic, 2000), validating this tool against Radiance allows for comparison over a much wider range of room sizes and aperture layouts than would be possible using measured data alone.

Also, to ensure that any correlation or otherwise was not simply a matter of configuration coincidence, a parametric comparison was undertaken over a range of room and aperture sizes, aperture positions, frame sizes, surface properties and work-plane heights.

A detailed description of the process used to convert room parameters to Radiance models with directly comparable configurations, as well as full details of each parametric run with associated Radiance input and output files, are available as a supplement to this paper available via: *http://performativedesign.com/data/RT-Daylight-Report-2017.pdf*

Results

A potentially high correlation with Radiance was immediately obvious when first developing the means to import and view Radiance results within the prototype tool, and dynamically switching between the two results sets. The overall daylight distribution patterns matched very well, however there was an apparent linear offset between the two. Radiance results were consistently lower than those from the simplified GPU method, being around 75% of the absolute grid point values calculated by the simplified GPU method. An example of this is shown in Figure 2.

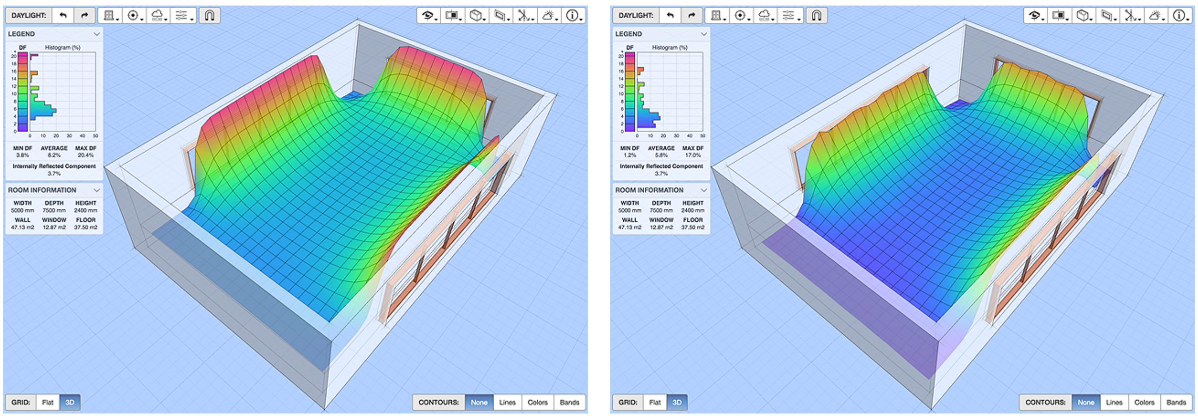

Figure 2: The prototype tool showing both GPU (left) and Radiance (right) results for the same model, the same distribution pattern but slightly different absolute values.

Overall Correlation

The detailed analysis of comparative results bore this out, showing that the correlation between raw daylight factor distributions generated by the GPU method and Radiance using the same room configurations was typically above 0.995 across a wide band of room parameter values. The correlation was found to reduce only under the following conditions.

Work-Plane Above Window Heads

Correlation falls to zero when the work-plane is located above the head height of apertures. In the GPU method there is no direct contribution from apertures located entirely below the work-plane, so the only value at each grid point is the average internally reflected component which makes the daylight distribution entirely uniform. In contrast, Figure 3 shows what is happening in Radiance. In this example, reflections from the ground and window sill illuminate an area of the ceiling directly above the window. As the work-plane is very high and close to the ceiling, it receives some illuminance from this bright patch which creates a slight variation across the Radiance grid. Even the slightest variation when compared to an entirely uniform grid will results in a zero correlation.

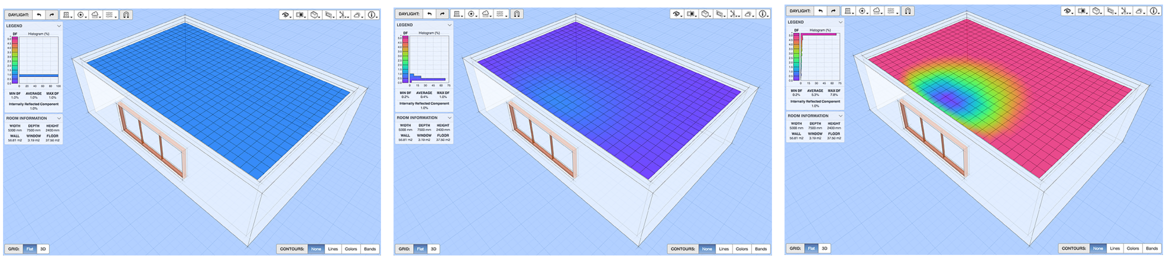

Figure 3: A comparison of daylight distribution when the work-plane is above the window, showing the GPU method (left), Radiance simulation (center) and a map of the difference between the two values at each grid point multiplied by 10 (right).

Very Small Windows

Without using the Radiance *mkillum* utility, correlation begins to fall below 0.995 when the total area of apertures becomes very small compared to the total internal surface area of the room. This is because of unevenness in Radiance results when ambient values alone are not sufficient to correctly model the internal daylight distribution. Under these conditions, even very high ambient settings that require calculation times of several hours cannot produce sufficiently accurate results for a high correlation. Figure 4 shows just how relative window size affects render quality in Radiance, as well as how significantly increasing the ambient parameters still does not solve splotchiness issues.

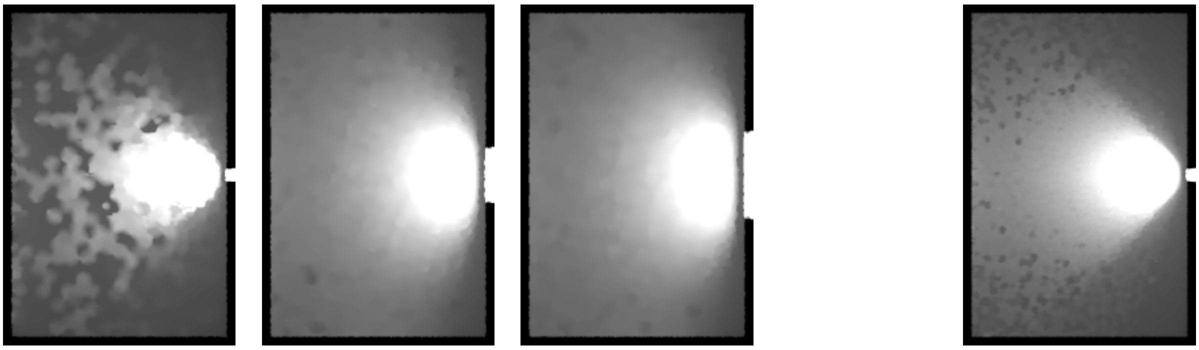

Figure 4: Example variations in ambient unevenness in Radiance using default settings with different aperture areas (left) and using very high ambient settings to compensate for a small window (right).

However, replacing apertures with *illum* sources using calculated luminance/radiance based on external sky conditions is one solution to this problem. To accommodate this, a switch from ambient calculations to the use of *illum* sources could be implemented whenever the total aperture area is below 7.5% of the total internal surface area. Though expeditious, such an approach would require additional consideration as it changes the Radiance calculation mode from indirect to direct so, whilst the results may be superficially comparable, the two methodologies are very different.

Non-Cuboid Room Geometry

Another condition that causes unevenness in Radiance results, and therefore a reduction in correlation, is when one dimension of the room is very small compared to the other two and the only windows are positioned in the smallest wall. For example, when the room is very long but also very thin, or when it is very large in plan but has a low ceiling height. This begins to occur when the smallest dimension is approximately 25% of the average of the other two.

Compensating for Relative Differences

Whilst the overall correlation is very good, there were still differences in the absolute values of daylight factors calculated by the two methods. Closer investigation by isolating the direct and diffuse components in each method show that this difference occurs almost entirely in the diffuse component. When the internal reflectances for all materials are set to zero, the resulting direct-only raw sky components of the daylight factor in both methods match very closely.

Some relationships between absolute differences and both surface reflection and work-plane height were identified, and lines of best fit generated from the comparative data. A brute force approach was then used to test a range of best-fit modifiers, separately and in combination. This involved iterating through all the available run data across all aperture configurations and parameter values many times to find the modifier with the maximum overall correlation and minimum overall difference between grid points. Whilst a number of arrangements of best-fit functions did show a reduction in overall average absolute differences, they all resulted in a reduction in overall correlation.

After trying a range of different best-fit functions and simple scaling, it was found that applying just a simple linear modifier of 0.52 to the internally reflected component of the GPU method achieved the highest overall correlation together with the minimum overall absolute difference between all grid point values across all aperture configurations and parameter values.

Obviously the true nature of differences in diffuse values between the two methods is very complex and, as the two methods calculate diffuse effects in very different ways, their absolute daylight factor values will never exactly match. However, by applying a simple linear modifier, the absolute values for most common room configurations with relatively small window frames, average surface reflectances and no single dimension less that 25% of the other two, can be made to match quite closely. A full set of comparative charts for different room and aperture configurations are provided in the detailed appendix.

Conclusion

The comparative analysis showed that daylight factors across a horizontal work-plane, calculated using the GPU method implemented in this work and those from Radiance using the same simple room model, are very highly correlated. It has also shown how a small modifier applied to the internally reflected component of the GPU method can provide a significant reduction in overall differences between absolute daylight factor values across a wide parameter range when compared to Radiance simulations.

At the same time, this work has also verified some of the limitations of the split-flux method already noted by others, in that it is less suited to rooms with very high average surface reflectances and geometry that has one internal dimension less than 25% of the other two.

However, this method is significantly more stable than Radiance across a wider range of room, window and surface parameters, and when dealing with small or widely separated windows. Also, even though there are conditions under which absolute values can diverge, the high correlation between daylight distribution patterns generated by this method and Radiance suggests that the primary cause and effect relationships involved are being accurately captured and that changes to room configuration are being accurately reflected as relative changes in the distribution pattern.

As the method is also fast enough to allow for real-time visual feedback of daylight distribution as calculation parameters are dynamically manipulated by the user, the authors argue that this makes the prototype tool developed here a valuable educational resource - allowing users to interactively investigate the relationships between daylight distribution, room dimensions, surface properties, window sizes and their positions within the envelope.

This work has also shown that, when used with a relatively simple rectangular room model, the process of calculating spatial daylight distributions can be so highly optimised that it becomes a trivial component of animation frame updates. This provides significant potential for future extension of the GPU method to handle dynamic and cumulative sky luminance conditions, climate-based daylight modelling and more complex shading and glazing systems.

References

Bourgeois, D., Reinhart, C.F., Ward G. (2008). A Standard Daylight Coefficient Model for Dynamic Daylighting Simulations. Building Research and Information 2008;36 (1):68e82.

BRE. (1986). Estimating Daylight in Buildings: Parts 1 and 2. BREPress.

Crisp, V.H.C., Littlefair, P.J. (1984). Average Daylight Factor Prediction, Proceedings of the CIBSE National Lighting Conference, Robinson College, Cambridge, UK.

Lynes, J.A. (1968). Principles of Natural Lighting. Applied Science Publishers, Ltd., London, pp. 129.

Lynes, J.A. (1979). A Sequence for Daylighting Design. Lighting Research & Technology, 11, pp. 102-106.

Mardeljevic, J. (1997). Validation of a Lighting Simulation Program: A Study Using Measured Sky Brightness Distributions, Proceedings of the 8th European Lighting Conference, Amsterdam, pp. 555-569.

Mardaljevic, J. (2000). Daylight simulation: validation, sky models and daylight coefficients. PhD Thesis. Leicester, UK: De Montfort University.

Osborne, J., Donn M. (2011). Defining parameters for a quality daylight simulation validation dataset. International Building Performance Simulation Association 2011. Sydney. Pp. 1481-1488.

Reinhart, C.F., LoVerso, V.R.M. (2010). A Rules of Thumb-based Design Sequence for Diffuse Daylight, Lighting Research and Technology 42(1), pp. 7-31.

TechPowerUp, (2017). GPU Database, Online: https://www.techpowerup.com/gpudb/ (accessed on 24 March 2017).

The U.S. DOE, (2017). EnergyPlus Engineering Reference: The Reference to EnergyPlus Calculations. https://energyplus.net/sites/default/files/pdfs_v8.3.0/EngineeringReference.pdf (accessed on 24 March 2017).

Versage, R., Melo, A.P., Lamberts, R. (2010). Impact of different daylighting simulation results on the prediction of total energy consumption. In Proceedings of the SimBuild 2010 Conference, New York, NY, USA, 11–13 August 2010; pp. 1–7.

Ward, G., Rubinstein, F. (1988). A New Technique for Computer Simulation of Illuminated Spaces, Journal of Illuminating Engineering Society, I, pp. 80-91.

Winkelmann, F., Selkowitz, S. (1985). Daylighting Simulation in the DOE-2 Building Energy Analysis Program. Energy and Buildings, v. l8, pp.271-286.

Yoon, Y.B., Lee, K.H. (2013). Comparing Radiosity and Split-flux illuminance algorithm in Energyplus. In Proceedings of the Biannual Autumn Conference of Korean Institute of Architectural Sustainable Environment and Building System (KIAEBS), Seoul, Korea.

Yoon, Y.B., Jeong, W.R., Lee, K.H. (2014) Window material daylighting performance assessment algorithm: Comparing radiosity and split-flux methods. Energies 2014, 7, pp. 2362–2376.

Yoon, Y.B., Manandhar R., Lee K.H. (2014). Comparative Study of Two Daylighting Analysis Methods with Regard to Window Orientation and Interior Wall Reflectance, Energies 2014, 7, pp. 5825-5846.