Rapid Algorithm for Modeling Daylight Distributions in Office Buildings

M.A. Lehar\textsuperscript{a} and L.R. Glicksman\textsuperscript{b}
Massachusetts Institute of Technology

\textsuperscript{a} Building Technology Laboratory, Room 3-412, 77 Massachusetts Ave., Cambridge, MA 02139, USA
\textsuperscript{b} Room 5-418, 77 Massachusetts Ave., Cambridge, MA 02139, USA

Abstract

We present a daylighting simulation tool, designed to predict the distribution of daylight in an office room using a rapid calculation procedure. Results from this simulation are compared to the output from a professional rendering program, and are found to agree within 10\% normalized error. A method for finding average light levels in an office room is described, and these calculations are used to infer a minimum required energy cost to supplement daylighting with artificial lighting. Finally, we present programming strategies that have been used to reduce computation times from 15 minutes to 3-5 seconds.

Keywords: daylighting, rapid visualization, energy load calculation

Introduction

Daylighting simulations have achieved a high standard of accuracy and realism in the last decade. Algorithms have been developed for interpreting brightness and color variations in the same manner as the human eye, and ever-increasing processor speeds mean that more ambitious and finely detailed scenes can be presented. In this sense, daylighting tools have become more realistic, but they remain difficult to use and still take many hours or days to render. In general, these visualization studies require so much investment that they can be applied to one or two finalized design concepts only. In other words, they are only used when there is no longer any possibility of changing the design in response to the results of the simulation.

In developing a daylight simulation technique that emphasizes calculation speed and ease of use, we hope to provide a tool that can be used in the early stages of a building’s design to improve daylight coverage and reduce energy consumption. Besides giving simulation a larger role in the development of design concepts, such a tool would permit many different designs to be compared easily, so that unusual or radical lighting designs would get more of a hearing than they do at present. The speed of the simulation is such that the same design can even be repeatedly tested in different sun positions and for different times of the year, providing a more complete analysis of the design and allowing a year-round calculation of supplemental lighting costs.

The purpose of the daylight prediction tool is to give an accurate estimate of the amount of daylight entering an office room during work hours. Based on this information, the required investment of energy in artificial lighting can be predicted for any office room. The tool uses data from location-specific weather files\textsuperscript{(1)} for its hourly lighting calculations. The first part of this study describes the technique used to simulate and display daylight distributions in a simple rectilinear space with a window set in one of the sides. The method is similar to that employed by Choi and Mistrick\textsuperscript{(2)}, with several new innovations designed to increase the speed of calculation. The accuracy of the procedure is checked against the output from a comparable Radiance\textsuperscript{(3)} room model, and found to diverge by no more than 10\% from the brightness levels calculated by the more detailed Radiance algorithm. The second part goes on to explain how the technique can be used to develop year-round histories of daylight coverage and predict the energy required for supplemental lighting. Using the model presented here, a desktop PC can deliver within a few seconds a tailored estimate of the implied year-round electricity
consumption of light fixtures in the simulated space. A working version of the model can be accessed at the Design Advisor[4] website.

1. VISUALIZATION TOOL

Simulation Setup

The virtual “room” used in the simulation is a rectilinear space with a window opening in one wall. The wall containing the window, the two walls flanking the window, the floor, and the ceiling together make up the basic geometry of the room. The rear wall is left out because it is the viewing plane in the user’s visual display. The omission is acceptable from the standpoint of the simulation because reflections from the rear wall would contribute minimally to the daylighting on other surfaces. Other elements include an optional set of blinds positioned in the plane of the window and a transparent workplane with the same dimensions as the floor, but raised to the elevation of a desk or work surface (Fig. 1). The workplane is transparent in the sense that it does not cast shadows or reflect light onto the other surfaces. Its only purpose is to measure the amount of light that reaches the desk level.

![Figure 1. Layout of the basic room model](image)

Each of the room’s panels, including the window and the workplane, is referred to as a “surface” in the simulation code. The current state of a surface is defined by a two-dimensional array of brightness values representing the area of the surface. The value at each grid position represents the brightness, or radiosity of the surface at that location. The simulation proceeds by assigning initial radiosity values to each of these array elements, then calculating for each position how much reflected light it receives from all other positions in the room (Fig. 2).

Simulating daylighting inside an enclosed space consists of two basic operations. The first task, which we will refer to as initialization, is to model the penetration of the sun’s direct rays into different parts of the room. The second, which we call iteration, is to carry out the reflection of this direct sunlight, as well as diffuse daylight, from one interior surface to another. The room is modeled as though unlit by electric light sources, revealing the amount of further illuminance that must be supplied by electric lights to meet a prespecified lighting requirement at the workplane level. By contrast, Choi and Mistick explicitly model electric light sources.

![Figure 2. The amount of light incident on element i is the sum of contributions from every other element in the room](image)

A sunlight pattern can be digitized to a coarse or fine mesh, in which each element records a radiosity value at its location.

![Figure 3. A sunlight pattern can be digitized to a coarse or fine mesh, in which each element records a radiosity value at its location](image)
Initialization

Before we can project sunlight through a window opening to illuminate the room interior, we must first devise a way to assign illuminance values selectively to different parts of the room. Following Choi and Mistick, we divide each of the room’s interior surfaces into rectilinear grids of many small elements, as shown in Fig. 3. Each element is given a numerical value corresponding to the level of surface brightness at its position, measured as a radiosity. Certain elements can then be assigned higher radiosities than others, depending on where the sunlight falls.

To determine which areas are directly illuminated by sunlight, and which are left in shadow, we make geometric calculations based on sun position and measurements of the window opening. The sun-brightened area of the floor, for example, is limited in the dimension parallel to the window by the right and left margins of the window area. In the other dimension (towards/away from the window plane), the limit is set by the window sill and head heights (Fig. 4).

Figure 4. Sunlight illuminating part of the floor of the room

The discrete elements that make up the virtual “surfaces” used in our simulation can each be set to a higher or lower value, depending on whether each element lies more in sunlight or shadow. The magnitude of the radiosity of a surface element in direct sunlight can be determined from the solar illuminance, the window transmissivity, and the absorptivity of the surface.

The initialization algorithm applies to direct sunlight only. To include diffuse sunlight in the room model is a much simpler matter. Before we move to the next stage of calculating reflections we set the surface elements corresponding to the “transparent” part of the forward wall to have a certain radiosity value. This window area acts like a source of diffuse sunlight, with the radiosity adjusted to match the given intensity of the diffuse sunlight component attenuated by a glass transmissivity factor. The contribution of the diffuse sunlight is then added to the direct sunlight distribution over each of the surfaces in the room.

Reflection Iterations

All surface elements in the room must be assigned brightnesses due to their being in or out of the path of direct sunlight from the window, before a simulation of reflected light, which we call an “iteration,” can be carried out. Once the initial radiosity values have been set, reflections and re-reflections from the interior surfaces can be calculated. Any element lying completely outside the path of direct sunlight has its initial radiosity value set to 0. For simplicity, all surfaces in the room are assumed to reflect light diffusely.

Figure 5. Summation of incoming reflections for one element

No information need be retained as to the original direction from which a ray of sunlight has struck a surface element, because there is no preferred direction in which it will be redirected after striking the surface. If one surface element is reflecting sunlight, any other element that does not lie in the same plane will receive a certain fraction of the light reflected. That fraction is given by the view factor between the two elements. All reflected light can be accounted for if view factors have been calculated for every possible permutation of two elements. The radiosity of element i is rendered as a sum of the radiosities of all n other surface
elements $j$ that do not share a surface with $i$, each multiplied by a view factor:

$$R_i = (1 - \alpha_i) \sum_j VF_{ji} \cdot R_j$$  \hspace{1cm} (1)$$

where $\alpha_i$ represents the absorptivity of element $i$. The equation holds when all elements have the same area.

A certain fraction of the light reflected from one surface element to another will again be reflected back into the room. To calculate the effect of this “second bounce,” we simply repeat the operation used to obtain the first set of reflections. By repeating these “iterations,” updating, at each step, the radiosities of each element, we eventually arrive at an equilibrium of surface radiosities, after which point there is no detectable change in the relative brightness of different parts of the room.

**Blind Reflections**

We have extended the model of Choi and Mistick to include blinds. Accordingly, the daylighting program accounts for light reflecting off blind surfaces, but the distribution on the walls and ceiling of light emanating from the blinds is too detailed to be represented meaningfully at low resolutions. Therefore, these reflections are calculated prior to the iteration phase, so that light from the blinds is applied to the interior surfaces along with the initial sunlight distribution. To this end, we consider the blind slat material to be a diffuse reflector in respect to the sun’s direct rays. Direct sunlight striking the blinds is reflected diffusely onto the surfaces of the room before the radiosity equilibrium is carried out during the iteration phase. Consequently, the iterations involve only the room’s 5 principal interior surfaces.

By adding blinds to the basic room model, we change only the initial distribution of sunlight in the room. Blinds span the window opening and can be tilted at any angle. Light intercepted by the blinds is either reflected into the room, absorbed, or reflected to the exterior. To account for the light reflected into the room, we assume that each blind slat is a diffuse reflector of uniform emissivity.

Although, in a real room, blinds would participate in a radiation exchange with the other surfaces, we account only for sunlight initially reflected from the blind slats, making this reflected light part of the initial sunlight distribution that prevails in the room before reflections between surfaces are calculated. A blind surface is considered to be illuminated by the sun if the line of sight between the sun and the center of the blind slat is not interrupted by the edge of the adjacent blind. A section of the ceiling (or wall) is illuminated in turn by light reflected from the blind slat if there is a line of sight between the center of the section and the center of the slat (Fig. 7). In calculating the view factor between the blind slat and sections of the inside surfaces, we use the discrete view factor sum developed above for the radiation exchange between surfaces (4). In this calculation, each blind slat is treated as a single element whose position is the geometric center of the slat. However, because this simplification causes areas of the ceiling directly over the blind slat centers to receive more light than those areas over the ends of the slat, the calculated view factors are averaged in the direction along the slat, and that average value used for the view factor between blinds and ceiling.

To account for the obstruction of direct sunlight by blinds in the window, we determine the position of the blind shadows. The extent of the shadow cast by each blind slat on the walls and floor is calculated using the same basic trigonometry from which the boundaries of sunlit regions were found.
Composing the Room Model

The time required to calculate each complete set of diffuse reflections (a complete set consists of one reflection for each surface element) rises geometrically with the resolution of the surface arrays. The total number of calculations goes as the 4th power of the rank of the surface array, so that a room composed of surfaces with radiosity arrays measuring 10 elements on a side will take 16 times as long to compute as a room with arrays that are 5 on a side.

Figure 7. A coarse grid is used to determine reflections, the initial sunlight pattern is removed, the resulting pattern is smoothed, and a finer pattern of initial sunlight is added in the final frame.

Because of the time required to compute reflections, we use a low-resolution grid of elements to represent the room when running the reflection iterations. At the same time, a separate model of the room is stored at a much higher resolution, but is used only for representing the initial distribution of direct sunlight in the room. This strategy of using two distinct resolutions to increase calculation speed is the principal difference between our model and that of Choi and Mistrick, in which a single resolution was used for both direct and diffuse sunlight. Our approach arrives at a single room model by combining together the high- and low-resolution surface maps. The composite model shows a crisp delineation between directly-lit regions and those that are initially in shadow, superimposed on a cruder plot showing the distribution of secondary light from reflections (Fig. 8). Since reflected radiation is spread diffusely about the room, one would expect it to have a smooth distribution, so that using a coarse grid to represent reflected light might introduce major errors if the grid were not then re-smoothed. To relieve the artificial "blockiness" of the reflected-light pattern, a bilinear interpolation function is used to smooth the edges of the low-resolution cells.

Figure 8. Radiosity values from each surface element in the room are organized into bins according to their magnitudes. a.) shows the population of each bin, and the running tally of values is shown in b.), rising to 100% of the total population at the right-hand side of the chart.

Displaying Regions of Light and Shade

To display the room model as a realistic rendered image, we map radiosity levels from the surface grids to a perspective drawing of the walls, floor, and window. To complete this mapping, it will not be sufficient to do a linear transformation of the range of illuminances in the room into shades of gray on a computer monitor. Even if we use walls with a high reflectivity (~0.7), the room image will look saturated with white in areas directly lit by the sun, and very dark elsewhere (Fig. 10a). In a real room, the eye does not perceive luminance...
values in proportion to their absolute magnitudes. Instead, the visual system increases contrast in certain bands of the illuminance range, and decreases it in others, depending on the relative abundance of light at different levels of intensity. Contrast is increased for levels of luminance that are well-represented in the world-image, and reduced for under-represented levels. The method presented by Larson, Rushmeier, and Piatko\cite{5} imitates human perception by using a Cumulative Population Density function to map world-illuminances to display-illuminances. We have followed this algorithm in its basic approach. Once the simulation has run to completion, we record the radiosity value at each of the grid positions in the fine mesh representing the room. We divide the spectrum of radiosities, from least to greatest, into 100 equal partitions, and assign each radiosity value from the room to its corresponding bin. By keeping track of the number of gridpoint values assigned to each bin, we build up a population histogram for the room radiosity values (Fig. 9a). In fact, we do not use the actual radiosity values, but the logarithms of the values, to compress the spectrum and better approximate the human “subjective response” to brightness levels (Larson, \textit{8}). Using the histogram of radiosity values, we can easily construct a “cumulative distribution function” (Fig. 9b), which spans the same range of radiosities as the histogram, but shows the number of readings with values \textit{at or below} a given radiosity range, rather than within that range. The CD function is defined in Larson in the following way:

\[
P(b) = \frac{\sum_{b_i < b} f(b_i)}{\sum_{b_i} f(b_i)}
\]  

where \( f(b_i) \) is the number of radiosity samples to be found within the log-radiosity range \( b_i \), and \( P(b) \) is the value of the CD at a given brightness value. The function \( P(b) \) can be used as a transfer function, showing the mapping from the log of radiosities recorded in the simulation to the log of levels of gray shown in the visual:

\[
\log(R_{\text{display}}) = P(b) \cdot \log(R_{\text{absolute}})
\]  

\textbf{Daylighting Results}

The daylighting tool was originally designed to produce a visual output—a 3-dimensional view of a room that would give a qualitative sense of the light distribution in a room lit only by natural light. This picture output from the

\textbf{Figure 9.} Display methods for the room model. In a., radiosity values are linearly mapped to levels of grayscale. In b., the transfer function is corrected to represent the image that the human eye would perceive when looking at a view of the real room display. The transfer function can be further modified using methods presented in Larson, but for our relatively simple display purposes, the original CD function appears to be sufficient. The image resulting from the CD transfer function in Fig. 9b is shown in Fig. 10b.
program is based on numerical results for the radiosities at each data point inside the room, which are each associated with a region of the picture and prescribe the particular shade of gray to be displayed in that region. The data point values are also used in a second feature of the daylighting tool, in which the simulation cycles through many hours of daylight in a given room and records the amount by which the local radiosity falls short of some required minimum during each hour. By summing the shortfalls from all points on the grid, an hour-by-hour history of the total lighting “deficit” can be assembled, and a prediction made as to the amount of supplementary lighting required.

The accuracy of the daylighting model can be judged by comparing the numerical output values from the simulation with the output from third-party rendering software. We used Lawrence Berkeley Labs’ Radiance[3] on a Sony VAIO Pentium2 with 64 Mb RAM. Radiance is a general-purpose rendering program that can be programmed to illuminate any surface geometry and accommodate any degree of transparency or reflectivity in the surface materials, as well as handling light sources and color.

![Floor Radiosity, Wall-to-Wall](image)

**Figure 10.** Illuminance distribution across the floor of the room, at a distance of 3m from the plane of the window.

We can perform a numerical comparison between models created with our software and those created using Radiance. This involves running both at the same resolution and matching surface radiosity values point by point. Our comparison for this study was carried out using a room with a floor area of 4m wide by 6m deep (in the direction away from the window plane), and 4m in height. The room is located at 30 degrees north latitude with a south-facing window that does not contain blinds. The window sill is set at .5m above the floor, and the head of the window at .5m below the ceiling. All interior surfaces have 30% reflectance, and all surfaces reflect diffusely.

The interior surfaces of the room models are divided into grids measuring 15 by 15 elements each. We have allowed 2 reflection-iterations to occur before measuring the resulting brightness distributions recorded by the grid elements. The sun angle has been set at 21 degrees elevation, 60 degrees azimuth west of south, similar to the conditions shown in the perspective views of Fig. 10. A stark division can be seen in the graph in Fig. 11 for the MIT model at the threshold between direct sunlight and shadow, halfway along the x-axis. While the boundary between light and shadow is much less distinct in the Radiance model, roughly the same absolute brightness levels are reported, particularly in the region at the right of the graph, representing the westward area of the floor where only reflected light has penetrated.

![Floor Radiosity, Window-to-Back (Fine Scale)](image)

**Figure 11.** Running the models at a higher resolution produces no significant difference in the distributions.

Our model cannot practically run reflection calculations at a resolution higher than 15x15x15. To rule out the possibility that the sharp kinks in the MIT plots are merely due to the low resolution and small number of permitted reflection cycles, we ran separate simulation having a disproportionately dense concentration of 30 gridpoints in the wall-to-wall (width) dimension, with the resolutions in the height and depth dimensions reduced to 7. The results (Fig. 12) were not significantly changed, even after increasing the number of reflection cycles from 2 to 10.
The Radiance room model uses the assumption of clear-sky conditions. The window has a 70% transmissivity, of which 80% is specular and 20% diffuse, and 0% surface roughness. The total strength of the sunlight incident on the vertical plane of the window is 205 W/m². The distribution of light intensities in the Radiance room has not been post-processed using Larson's histogram-based equalization scheme, so all brightness values are taken directly from the ray-tracing calculations. To calibrate the MIT model with the Radiance parameters, we apply attenuation factors to the initial sunlight brightness $G_i$ used in the Radiance model. In the MIT model, the radiosity of the directly lit portions of the floor and walls is initially set to the value

$$R_i = \frac{f_{\text{direct}} \tau_{\text{window}}}{\rho_{\text{surface}}} G_i$$  \hspace{1cm} (4)$$

where $f_{\text{direct}}$ is the direct component of the sunlight, $\tau_{\text{window}}$ is the transmissivity of the window, and $\rho_{\text{surface}}$ is the reflectance of the interior surfaces. A longitudinal brightness profile, spanning a line drawn along the floor from the window plane to the back of the room, provides another assessment of the quality of the match between Radiance and MIT models. Fig. 13a. shows a profile running along the centerline (meridian) of the room, and 13b. shows a parallel profile, crossing a part of the floor where the pattern of direct sunlight is somewhat broader. As in Fig. 11, the MIT plot shows a sharper delineation between areas of direct sunlight and partial shadow, but the peak brightnesses occur at the same location in both models.

The visual output from the MIT model is similar to a Radiance image when the same scenarios are run in parallel. Regions of light and dark in the Radiance image are well preserved in the more rapid MIT simulation (Fig. 14).

2. PREDICTIONS OF LONG-TERM ENERGY USE

So far, a method has been described with the object of creating a display image of the room model using a fixed sun position. A visual output showing the room at a certain time of day might be useful as an aesthetic design aid, but provides no quantitative feedback on the room design, and is of no help in planning for energy efficiency. The amount of energy consumed in electrical lighting fixtures is zero in a room that already receives enough natural sunlight through the windows to meet the lighting requirements of the occupant. But if, in the course of lighting a room, the sunlight also heats the interior space excessively, energy may have to be expended on air conditioners to control the temperature. Therefore, the total energy use will be minimized by an optimization of window size and position that accounts for the heating and cooling requirements of the building. To estimate the amount of electrical lighting energy used in a room during a given hour, we define a “desired light level” consistent with the kind of activity...
that will be taking place in the room, and pertaining to an invisible

![Figure 13. Visual output from the MIT program, without blinds (top) and with (bottom). Comparable simulations in Radiance are pictured at right, along with the duration of the test](image)

surface, some distance above the floor, called the “workplane” (Fig. 15). The workplane is a surface in its own right, like the walls, floor, and window. The initialization and iteration methods can be used to apply direct sunlight and reflections from other surfaces, and the resulting light pattern on the workplane could be displayed in a finished rendering. We can also use the illuminance values for each element of the workplane to compare the intensity of light due to daylight with the desired light level. In those regions of the workplane where natural daylight does not meet the lighting requirement, electrical lighting must make up the difference. The total electrical energy requirement for the hour can be expressed as follows:

$$ I_{art} = G_{\text{min}} - G_{\text{daylight}} $$

(5)

where $I_{art}$ is the illuminance required from artificial lights, $G_{\text{min}}$ is the prescribed minimum illuminance for the workplane, and $G_{\text{daylight}}$ is the local natural illuminance due to direct sunlight and reflections. Rewriting $G_{\text{daylight}}$ in terms of the radiosity $R$ of the workplane element in question,

$$ I_{art} = G_{\text{min}} - \frac{R}{\rho} = G_{\text{min}} - \frac{R}{(1 - \alpha)} $$

(6)

where $\rho$ is the reflectivity of the workplane and $\alpha$ is the absorptivity.

![Figure 14. The “Workplane” surface within the simulated room model](image)

This summation can now be compared with the cooling energy for the room during the same 1-hour period. By running the simulation once for each of the working hours in the day, and repeating this cycle at least once for each month of the year, we build a reliable history of electrical energy use. In our model, we assume that the electrical lighting can be continuously varied, and each fixture individually controlled, to achieve the desired lighting level. This represents the extreme of efficiency, at which the sum total of illumination from electrical sources is no greater than the amount by which the illumination of each surface element falls short of the requirement. The illuminance total is given by

$$ \bar{I}_{art} A_{workplane} = \sum_{i=1}^{m} \sum_{j=1}^{n} G_{min} - \frac{R_{ij}}{(1 - \alpha)} $$

(7)

where the workplane has dimensions $m \times n$ and $R_{ij}$ is the radiosity at coordinate $(i, j)$, always excluding those values of $i, j$ for which $G_{\text{min}} < \frac{R_{ij}}{(1 - \alpha)}$, since these elements do not
contribute to the lighting deficit. A lighting system that was not minutely variable, and provided the same light intensity everywhere in the room, would be more usual. The illuminance total of such a system would depend only on the brightness level of the dimmest element of the workplane, since by providing enough illumination to satisfy the lighting requirement at that dimmest point, we meet or exceed the requirement at every other coordinate. This total is expressed as

\[ I_{art} A_{workplane} = (m \times n)(G_{\text{min}} - G_{inf}) \]  

where \( G_{inf} = \frac{R_{inf}}{(1 - \alpha)} \) is the least illumination received by any element of the workplane.

### Time-Saving Strategies

The program creates new workplane brightness maps for every hour between 8am and 6pm of each day of the year. In general, the computational expense of running an entirely new simulation for each of those hours would be prohibitive. To reduce the time required to build up a full-year history of hourly lighting deficits, we have used several streamlining strategies:

1. **View Factor Storage**
   The most time-consuming step of the rendering procedure is the calculation of the view factors between every allowable pairing of surface elements within the room. This computation only needs to be performed once for any one room geometry, so in the case of the hourly workplane calculations, we refer to stored view factor values for each new calculation.

2. **Low-Resolution Rendering**
   When calculating annual histories of lighting energy requirements, we can use a low-resolution grid to represent the sunlight distribution on the interior surfaces of the room. The total lighting deficit recorded for each hour is effectively an average “brightness difference,” taken over the entire workplane, between the required lighting value and the amount received as daylight. If our aim is to capture this average value accurately, rather than display a crisp boundary between areas of sunlight and shade in a visualization exercise, we can afford a much lower grid resolution to record surface radiosity values. In our original simulation, which was used for a display showing the daylighting effects in a three-dimensional view of an office room, we maintained a coarse grid that we used to calculate element-to-element view factors and carry out reflection calculations. The reflected light distribution recorded on this coarse grid was separated from the initial distribution of sunlight, and overlaid on a fine grid of 100x100 elements on each surface, which contained the highly detailed pattern of the original sunlight distribution. In the workplane calculation, it is only this fine grid whose resolution is changed. We reduce it to 20x20 grid elements on each surface, to significantly reduce the time required to establish a new sunlight pattern in the room at the start of each simulation hour.

3. **Superposition of Illuminances**
   The sunlight pattern in our simulated room changes dramatically as the sun position changes throughout the day. But from one day to the next at the same hour of the day, the sunlight pattern and associated reflections change only slightly when normalized for average intensity. We have demonstrated that the change in sun altitude angle is slight enough that for our purposes it can be treated as constant throughout the month, giving an error of less than 1/10th of 1 percent in the lighting deficit recorded for each hour. Using this simplification, we need to run only 12 daylighting configurations for each month of the year - one for each business hour of one day - and apply the “daylight template” so created to each other day of the month to compile the complete history of hourly deficit values. Certain parameters will vary from one day to the next, but in our simulation these are limited to 1.) the intensity of direct solar illuminance, 2.) the intensity of diffuse illuminance, and 3.) the setting of the window blinds. We allow that the levels of direct solar and diffuse solar, read from the appropriate location-specific weather data file can vary independently of each other. No single scaling value can be used to match a daylight template to any day of the month at the given hour. We therefore maintain two separate daylight distributions for each of a given month’s set of 12 templates. One distribution contains the initial distribution of direct solar and the reflections resulting only from the direct
solar. The other contains the reflections resulting only from the diffuse solar (there is no “initial distribution” of diffuse solar, as it is modeled by an initial diffuse brightness over the surface of the window). These two distributions can then be scaled independently and added together to produce the complete daylighting distribution for any day of the month at a particular hour. If the direct illuminance recorded for hour \(i\) on the first day of month \(j\) is given by \(I_{\text{direct},ij}^{\text{direct}}\), and the diffuse illuminance by \(I_{\text{diffuse},ij}^{\text{diffuse}}\), the scaling factor for the radiosity value of any element of the daylight distribution on day \(k\) of the month is defined:

\[
R_{\text{element}}^k \equiv \frac{I_{\text{direct},ijk}^{\text{direct}}}{I_{\text{direct},ij}^{\text{direct}}} R_{\text{element},\text{direct}}^1 + \frac{I_{\text{diffuse},ijk}^{\text{diffuse}}}{I_{\text{diffuse},ij}^{\text{diffuse}}} R_{\text{element},\text{diffuse}}^1
\tag{9}
\]

where \(R_{\text{element},\text{direct}}^1\) is the radiosity of the corresponding element of the diffuse illuminance distribution from the template, \(R_{\text{element},\text{diffuse}}^1\) is the radiosity from the direct illuminance distribution, and \(R_{\text{element}}^k\) is the resulting total radiosity value for the element on day \(k\) of the month. \(I_{\text{direct},ijk}\) and \(I_{\text{diffuse},ijk}\) are given by the weather file for the particular hour and day of the month in question. If the room has been configured with blinds, two parallel simulations are produced; one in which the blinds are always open, and another in which they are “closed” by being set an arbitrary fixed angle oblique to the sun’s rays. We have used the method described in Arons\[6\] to calculate the effective transmissivity of venetian blinds for direct and diffuse transmission. These are represented by the factors \(\tau^b\) in the following restatement of (10):

\[
R_{\text{element}}^k \equiv \tau^b_{\text{direct}} \frac{I_{\text{direct},ijk}^{\text{direct}}}{I_{\text{direct},ij}^{\text{direct}}} R_{\text{element},\text{direct}}^1 + \tau^b_{\text{diffuse}} \frac{I_{\text{diffuse},ijk}^{\text{diffuse}}}{I_{\text{diffuse},ij}^{\text{diffuse}}} R_{\text{element},\text{diffuse}}^1
\tag{10}
\]

Using the open-blinds and closed-blinds versions of the room model, a complete history of hourly lighting deficits can be built up for a room whose blinds open and close at arbitrary intervals. The overall lighting deficit calculation (9) is conducted each hour, and the radiosity value for each element of the workplane surface is chosen either from the blinds-open value of \(R_{\text{element}}^k\) or the blinds-closed one, depending on the setting of the blinds for the current hour.

**Results**

The calculation procedure outlined above can be used to show the effect of siting on the annual lighting requirement. In Fig. 16a, the lighting deficits, averaged by the month, are shown for an office room situated in London with a west-facing view from a single, wall-spanning window without blinds, a workplane 0.5m off the floor, a room 4m high and 6m wide, with 0.6-m margins above and below the window (height = 2.8m). The interior walls have an emissivity of 0.4. For a minimum lighting requirement at the workplane, we are using 400 lux, the European Union standard. This is also the maximum possible deficit – the amount of lighting that must be provided artificially when no light enters from outside (night conditions).

The depth of the room is varied along the abscissa in Fig. 16b, showing an exponential increase in average electricity demand as the room becomes deeper in the dimension perpendicular to the window. Fig. 17 shows the variation of required lighting energy by room orientation. A north-facing room (in the northern hemisphere) receives no direct sunlight, while a room with a southern prospect receives some direct light during every hour that the sun is visible.
Figure 15. Change in the lighting deficit with varying room depth for a west-facing room in London with a lighting requirement of 400 lux

A west-facing building is exposed to direct rays for only half the day, but because of the low angle of incidence towards sunset, may receive more daylight overall than a south-facing window, as in the case of Cairo (Fig. 17a). As we would expect, at higher latitudes the incidence angle on the south face decreases, and the share of direct sunlight goes up relative to the west-facing window, as in Fig. 17b (Anchorage). Data has only been recorded for business hours (8am to 6pm), so the sun will be visible to the west for a longer time than it can be seen through an east-facing window. Fig. 18 shows a range of average deficits for various cities ordered from north to south. In Cairo, the most southerly location, electrical lighting accounts for only 3.9% of the total lighting requirement for the working surfaces.

Conclusion

The rapid diffuse simulation has two main advantages over conventional daylighting techniques that use ray-tracing and specular reflections. First, the rapid response time allows the tool to be employed in the early stages of a building’s design, when many ideas need to be compared quickly to identify promising concepts. A first-pass analysis of this kind is only useful if it can provide results that meet a certain criterion of accuracy. It is unlikely that any parameters affecting daylighting would have been determined to within less than a 10% margin of error in a building design that is still in a conceptual phase. Given that such designs are not expected to be completely specific, the error measurable in our simplified diffuse
model of daylighting would not impose any new uncertainty on an architectural proposal for daylighting. The second main advantage of a rapid simulation is that it can be used for a long-term analysis of daylighting in which evaluations are made hundreds of times, rather than just once. A more detailed rendering tool, while providing a slightly more accurate view of a workspace at a single instant in time, would not be practical for understanding the overall performance of the space under many different daylighting conditions throughout the year.


