A rapid and scalable radiation transfer model for complex urban domains

Matthew Overby\textsuperscript{a}, Peter Willemsen\textsuperscript{a,\textsuperscript{*}}, Brian N. Bailey\textsuperscript{b}, Scot Halverson\textsuperscript{a}, Eric R. Pardyjak\textsuperscript{b,\textsuperscript{**}}

\textsuperscript{a}University of Minnesota Duluth, Department of Computer Science, Duluth, MN 55812, USA
\textsuperscript{b}University of Utah, Department of Mechanical Engineering, Salt Lake City, UT 84112, USA

Abstract

An important component of urban microclimate is the radiative heat transfer between the myriad elements that make up the urban fabric. While great progress has been made in developing radiation models for idealized urban spaces, operational simulations of fully resolved city-scale domains remain elusive. As a result, simplifications and assumptions must be made. Such compromises may limit utility, and reveal a need for new, scalable microclimate models. This paper presents a novel, physically-based, building-resolving radiation transfer model (QESRadiant) that utilizes ray tracing techniques accelerated with graphics processing units (GPUs). QESRadiant builds on computer graphics methods, incorporating approaches for global illumination and light transport. Tests show that our methods can rapidly simulate the radiation balance for millions of surfaces with unique shapes and properties using a single consumer-class workstation on the order of several minutes. High-resolution (0.5 m) street canyon radiation budgets are validated using field data covering nine months. The results show the model is able to predict radiative fluxes in the canyon with overall average $R^2$ of 0.77 and mean error of 14.5 Wm$^{-2}$. Solar radiation was extremely sensitive to geometric obstructions, demonstrating the need for high-quality field experiments that resolve urban details affecting the radiation balance.

Key words: Graphics processing units, Heat transfer modeling, Ray tracing, Sustainability, Urban street canyon

Preprint submitted to Elsevier 4 February 2016
1 Introduction

In the past decades, substantial effort has gone into understanding the various mechanisms that contribute to urban climate (Arnfield, 2003; Oke, 1987). While great progress has been made in understanding individual aspects of urban climate, there is still a lack of understanding regarding the interaction of large numbers of different urban elements amongst each other and across scales. Further, there is a dearth of models that include sufficient microscale details that can be used to make place-based decisions and to understand the impacts of these decisions on the larger-scale urban environment (Pataki et al., 2011). There is a critical need for the development of these types of simulation methodologies for designing healthy and efficient cities. One approach involves the parametrization of key physical processes at relatively coarse scales for use within mesoscale weather models (e.g., Grimmond et al., 2010; Martilli et al., 2002; Masson, 2005). An alternative approach toward achieving these goals is to implement a detailed bottom-up representation by resolving the key processes at small scales. With this approach, simulations are designed to resolve urban form details such as buildings and vegetation as well as their complex mass, momentum, and energy transport processes in urban environments.

Our group has been working to develop a building and vegetation resolving system through the GEnUSiS project (Green Environmental Urban Simulations for Sustainability). As part of the project, we are developing a fast modeling system for calculating and visualizing the transfer of heat and moisture in cities for improved design of urban green environments. Such environments include large vegetative spaces, residential gardens, pervious parking lots, green and/or reflective rooftops, and artificial water bodies. This complete modeling system that includes mass, momentum, and heat transfer in urban areas is called QUIC EnvSim (QES). QES builds on the QUIC Dispersion Modeling System (Brown et al., 2013; Pardyjak and Brown, 2001; Singh et al., 2008, 2011) that was designed as a building-resolving tool for rapidly computing wind, turbulence, and concentration fields in cities. QES can be run as a
stand alone system, or interoperate with QUIC.

In this paper, we focus on the implementation of an efficient graphics processing unit (GPU) based ray tracing radiative heat transfer module of QES called QESRadiant. Our computational approach builds upon past work from the field of computer graphics that approximates and simulates the transport of visual radiation (light). The methods and models of QESRadiant can simulate key radiative transfer mechanisms for user specified wavebands (e.g., shortwave or longwave radiation) in realistic building-resolving urban domains at a user-specified resolution. Such urban spaces may be made up of hundreds to millions of buildings with unique shapes, sizes, and reflectance/absorption properties. Due to the high efficiency of our methods and throughput power of GPUs, these domains can be simulated on a single consumer-class desktop computer or laptop, with simulation times on the order of seconds to minutes. While the presence of vegetation is extremely important, this paper will focus on modeling the urban radiation balance without vegetation. Modeling radiation in vegetative urban domains is discussed in Bailey et al. (2014). We validate QESRadiant by recreating a full-scale urban street canyon field study (Eliasson et al., 2006; Offerle et al., 2007). Measured flux densities for various radiation bands are compared to simulated flux densities at several positions within the canyon. Six statistical measurements are used to assess QESRadiant’s ability to predict radiative flux densities at several positions in the urban canyon within acceptable error.

1.1 Building-resolving radiation modeling efforts

The methodologies used to simulate radiative transport processes in urban areas have evolved over the years as computational resources have improved. Most of the earlier urban radiation exchange models focused on idealized street canyon geometries to compute shortwave and longwave radiation interactions between two buildings using view factors that could be derived analytically (e.g., Terjung and Louie, 1997; Terjung and O’Rourke, 1980). However, the difficulty of determining analytical view factors increases with complexity of modeled geometry. Due to the rise in computing power, it has become more common to calculate view factors in a statistical or Monte Carlo manner. For small groups of three-dimensional buildings, Kobayashi and Takamura (1994) used a Monte Carlo ray tracing method to simulate outgoing longwave radiation from an array of cubical buildings. Krayenhoff and Voogt (2007) used ray tracing to determine visibility, but computed view factors between surfaces with exact formulae. The presence of urban vegetation presents substantial complexities that should also be accounted for, such as the simplified ray tracing methods of Asawa et al. (2008). In many applications, radiative fluxes are not the end product of the urban modeling system, but rather it is used...
as an input to other models. For instance, Matzarakis et al. (2006) developed the *RayMan* software which is able to model groups of buildings for thermal comfort applications.

The above models all must make compromises in their level of physical description and domain size. In order to simulate city-scales, assumptions of homogeneity must be made at some level. Recently, Yang and Li (2013) have developed a method similar to Krayenhoff and Voogt (2007) that uses ray tracing to determine view factors, but then uses the Gebhart method (Gebhart, 1971) to handle multiple reflections. Their methodology can simulate hundreds of buildings with multiple reflections using a novel data storage compression scheme. As previously noted, GPUs can be used to accelerate computations, which was illustrated by Takizawa et al. (2006). They did not use a ray tracing sampling approach, but instead used the GPU to assist with radiosity methods for radiation exchange. Other multi-building radiation interaction tools such as the IES-Virtual Environment 6.0.2, ApacheSim/RADIANCE used by Strömann-Andersen and Sattrup (2011) can handle multiple idealized buildings and include energy use modules. However, they generally offset computational cost by using simplified physical models that do not fully represent vegetation, anisotropic scattering, or local wind transport.

1.2 *Computer graphics and the simulation of light*

Computer graphics research over the last several decades has mostly used rasterization and ray tracing, two techniques for generating visual representations of generalized 3D physical worlds. Rasterization algorithms project geometry onto a camera’s image plane converting the geometry into image pixels. These techniques generally require sophisticated image-based projection methods to compute the effects of reflections or shadows (Fernando et al., 2001; Segal et al., 1992; Williams, 1978). Nonetheless, rasterization techniques have become the dominant method for visual image generation in contemporary video games due to the parallel computational support of matrix and vector operations on GPUs. Alternatively, ray tracing algorithms may simulate the physical nature of light by performing intersections between light or viewing rays and 3D scene objects to determine visibility, shadows, and the propagation of light energy (Cook et al., 1984; Whitted, 1980). While computationally more expensive, ray tracing-based methods more accurately approximate light transport through stochastic sampling mechanisms supporting necessary scattering and absorption transport paths between different surface material types.

Ray tracing methods have evolved into sophisticated, realistic image synthesis algorithms approximating the light transport equation proposed by Kajiya (1986). Kajiya’s equation analogs the radiative transport equation but
focuses on the scattering of visible light between all surfaces allowing for different surface scattering attributes. Kajiya’s efforts formed the core Monte Carlo ray sampling underpinnings for path tracing (Kajiya, 1986), photon mapping (Jensen, 1996), and other stochastically-based sampling mechanisms that maintain the conservation of energy. Our method utilizes the stochastic sampling approaches related to ray tracing algorithms, and applies these techniques to discretized urban domains.

1.3 GPU acceleration

The use of GPUs for acceleration was chosen over classic cluster and supercomputing methods for several reasons. Due to consumer demand for sophisticated video game graphics, the performance capability and affordability of GPUs have seen significant growth in the last ten years. Adopting GPU methods allow QES to run on a wide range of consumer-grade desktop machines and benefit from future graphics hardware improvements. Additionally, radiation exchange with ray tracing requires global sum operations for the accumulation of energy. In many multi-node distributed frameworks such as Message Passing Interface (MPI Forum, 2015), this requires expensive collective communication. This problem is dramatically mitigated on a GPU, in which global memory is resident on the same device as the compute cores.

Typically, programmers create shaders (computer graphics functions) to compute rendering effects and pixel colors using graphics application programming interfaces (APIs) such as OpenGL (Khronos Group, 2015). Many shaders are specifically defined to operate on the GPU using single-precision floating point format. However, there has recently been a trend to adapt GPU-based programming to general purpose computing (GPGPU) (Owens et al., 2007). GPGPU is now represented in numerous scientific fields, such as fluids simulation, computational mathematics, and visualization. To facilitate this trend, NVIDIA Corporation (Santa Clara, CA) released a programmable platform called Compute Unified Device Architecture (CUDA) (Nickolls et al., 2008). In CUDA, kernel functions are defined to operate in parallel on the GPU and can be applied to a broader range of applications than shaders. Such features that extend capabilities include double precision, dynamic parallelism, and thread communication and synchronization. For more information on scientific computing with NVIDIA CUDA, see Luebke (2008).

QESRadiant uses GPU accelerated ray tracing to simulate radiative heat transfer. Ray tracing has the feature of being inherently parallelizable in which ray-object intersection tests for different rays are concurrent. However, complexities in recursive ray tracing provide a great challenge and limit effectiveness of GPU-based ray tracing. In 2010, NVIDIA released OptiX, the general
purpose GPU ray tracing engine (Parker et al., 2010). Developers are able to create their own ray generation and intersection kernels, as well as define the primary interactions of a ray tracer. OptiX will compile this information and efficiently perform ray-object intersection tests on the GPU using CUDA. QESRadiant relies heavily on the optimizations provided by OptiX. Many of the strategies, program flow, and data containers are specifically designed to interoperate with OptiX.

1.4 Heat transfer modeling

The QESRadiant module is a part of a grid-resolving computational system where each discrete surface has its own energy balance, defined as

\[ Q^* = Q_H + Q_E + Q_G. \]  

(1)

Eq. 1 includes net all-wave radiation flux \( Q^* \), sensible heat flux \( Q_H \), latent heat flux \( Q_E \), and ground or conductive heat flux \( Q_G \). QUIC EnvSim is modular, so different methods can be used to compute the terms of the surface energy balance equation. However, the focus of this paper is to present QESRadiant’s ability to compute the net all-wave radiative components of \( Q^* \), the driving force of the surface energy balance.

The net all-wave radiation \( Q^* \) modeled in this paper is decomposed into short-wave (\( K \)) and longwave (\( L \)) radiation. Each band is also separated into incident (\( \downarrow \)) and reflected or emitted (\( \uparrow \)) counterparts. This equation is defined as

\[ Q^* = K^\downarrow - K^\uparrow + L^\downarrow - L^\uparrow. \]  

(2)

2 Modeling methodology

2.1 Domain representation

Domains in QES attempt to maintain compatibility with the QUIC Dispersion Modeling System, hence buildings and ground geometry are defined according to the needs of QUIC. However, additional urban structures such as vegetation and sensors can be present in a QES domain as well.
2.1.1 Buildings

Buildings are represented as grid-aligned three-dimensional boxes with positive, arbitrary dimensions. A building can be made up of several boxes that can be stacked or placed next to others. The ground is represented by a two-dimensional horizontal plane. Faces of each box and the ground plane are further segmented into smaller two-dimensional planes called patches. Patches of the ground plane will exist below buildings, but such patches will not emit longwave radiation. The length, width, and height of a patch may vary, but must be the same for all patches within the domain (required by QUIC). The domain in QES can be specified in three ways: an XML document, programmatically, or a QUIC Project file. QUIC Projects can also be supplemented with an XML document to specify additional urban structures. A domain can consist of one to several million patches. Increasing the number of patches for a domain can potentially increase model accuracy at the expense of also increasing computational cost.

Each patch has multiple physical properties that affect its interaction with the environment. One such property is albedo ($\alpha$), which is defined as the ratio of reflected to incoming radiation and typically depends on the wavelength of incoming radiation. For the urban environment, we define a total albedo for the entire shortwave ($\alpha_s$) and longwave ($\alpha_L$) spectra. Similarly, each patch is given a total, hemispherical emissivity value ($\epsilon$), which is defined as the ratio of emitted radiation flux to the emission flux of a black body. The user has the option of defining these properties manually, or by assigning it a material type with predefined values for each material from Santamouris (2013). For energy to be conserved, the surface emissivity and longwave albedo must sum to one. QES will automatically verify this in the case of user-specified values.

2.1.2 Sensors

Virtual sensors are two dimensional planes that gather information about the simulated environment. Their purpose is to simulate real-world sensors and diagnostic tools, such as a pyranometer measuring solar irradiance. Unlike
buildings, they can be arbitrarily positioned in the world and be of any unique size. In QESRadiant, a sensor will record (but not augment) radiation that passes through its face in the direction of its normal. Sensors are used to compare simulated radiation fluxes to observed or measured fluxes in the real world. This allows model validation and is used to determine model accuracy in section 3.

2.2 Radiation transport

To simulate the transfer of radiation from one medium to another, rays are emitted from patches, the sun, and the atmosphere with an initial energy. When the ray intersects a patch, fractions of that energy are absorbed, reflected, and scattered. The amount of energy that is absorbed, reflected, or scattered depends on the wavelength of the radiation being simulated, as well as the physical properties of the patch. This process is illustrated as a flowchart in Fig. 2. Total hemispherical radiation flux density is denoted $K$ and $L$ ($Wm^{-2}$) for shortwave and longwave, respectively. The power or energy rate of an individual ray is $E$ ($Wray^{-1}$), and $N$ number rays are launched.

Single precision floating point format is most often used in QESRadiant when calculating radiation exchange on the GPU. This increases the computational efficiency of kernel execution. However, double-precision floating-point format is used only when necessary for values outside the capability of single precision. This requires a compute capability of 1.3 or higher, which includes nearly all NVIDIA GPUs made after 2009.

2.2.1 Shortwave transfer

Shortwave radiation is received by patches in three ways. Radiation that reaches the patch directly from the sun is considered direct shortwave energy. Energy that has been scattered by the atmosphere and eventually reaches a patch is diffuse shortwave energy. Diffuse shortwave is sometimes called scattered energy in literature, but we reserve that term for energy that is scattered or reflected by surfaces, the third way a patch can receive shortwave energy. The initialization of direct and diffuse shortwave ray energy rate is defined by its unobstructed flux. The direct solar flux ($S_{dr}$) and diffuse solar flux ($S_{df}$) can be assigned values directly (e.g., from measurements) or simulated with a solar flux model. If $S_{dr}$ does come from measurements, it must be weighted by the cosine of the zenith angle. They are considered unobstructed in the sense that the value corresponds to the incoming flux above the urban canopy, not augmented by urban structures.

If measurement values are not available for $S_{dr}$ and $S_{df}$, they must be simulated.
Program begins

<table>
<thead>
<tr>
<th>i=1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Set initial energy:</td>
</tr>
<tr>
<td>if i=1, E_{init} = E_{s,dr}</td>
</tr>
<tr>
<td>if i=2, E_{init} = E_{s,ℓ,df}</td>
</tr>
<tr>
<td>if i=3, E_{init} = E_{ℓ}</td>
</tr>
<tr>
<td>if i=4, program ends</td>
</tr>
</tbody>
</table>

Launch ray

Scatter energy

(a)

Scatter energy

Set ray energy:
1) E_{init} = E_{scatter}
2) E_{scatter} = 0

Launch ray

Check scattered energy

E_{scatter} ≤ ⊗

Patch absorbs energy:
1) K, L += E_{scatter}
2) E_{scatter} = 0
3) Stop scattering

(b)

Launch ray

Set ray energy:
E_{ray} = E_{init}

Sky absorbs energy:
1) E_{ray} = 0

Test for intersection

Hit sky

Stop reflecting

Patch absorbs energy:
1) K, L += E_{ray}/A_p
2) E_{ray} = 0

(c)

Patch absorbs energy, stores scattered energy for later, then launches reflected ray:
1) K, L += (1-α)E_{ray}/A_p
2) E_{scatter} += (αδ_d)E_{ray}
3) E_{ray} = (αδ_s)E_{ray}

E_{ray} > ⊗

Hit patch

Check ray energy

E_{ray} ≤ ⊗

Fig. 2. The primary operations of QESRadiant for a single time step. Radiation is initially emitted from the sun, the atmosphere, and patches in (a). A ray continues to specularly reflect until all energy (E_{ray}) has been absorbed in (c). Scattered energy is emitted separately in (b) where energy may continue to be absorbed, reflected, and scattered. If a rays energy is below a threshold (⊗), all energy is absorbed by the patch.

doi:10.1016/j.uclim.2015.11.004
with a model. One such model implemented in QESRadiant is described by Monteith and Unsworth (2008) and is a cloudless sky radiation model used to calculate total solar radiation. The model is modified so that the relative air mass \((m)\) is computed according to Atwater and Ball (1978). To distribute the energy into diffuse and direct components, the model is also altered based on measurements by Liu and Jordan (1960), which distinguishes the relationship between diffuse, direct, and global solar radiative flux density. Currently, this flux model does not take the turbidity of the atmosphere into account.

In Eq. 3 below, \(\tau_m\) is the molecular extinction coefficient, \(\tau_a\) is the aerosol extinction coefficient, \(\theta_s\) is the solar zenith angle, and \(S_0\) is the solar constant \((\approx 1367 \text{ Wm}^{-2})\). Both \(\tau_m\) and \(\tau_a\) are based off measurements from Liu and Jordan (1960), but the user may adjust these values if they are known. \(S_{dr}\) and \(S_{df}\) may be computed as followed:

\[
m = \frac{35}{\sqrt{1 + 1224 \cos^2(\theta_s)}},
\]

\[
S_{dr} = S_0 \{ \exp[-m(\tau_m + \tau_a)] \} \cos(\theta_s) \quad \text{[Wm}^{-2}],
\]

\[
S_{df} = 0.3S_0 \{ 1 - \exp[-m(\tau_m + \tau_a)] \} \cos(\theta_s) \quad \text{[Wm}^{-2}].
\]

Direct solar radiation is simulated by launching collimated rays from the sun toward the ground surface. To accomplish this, the diameter of the entire domain is determined by computing a bounding sphere of all terrestrial geometry. This diameter is used to determine the area of the disk that acts as the emitting plane with a normal parallel to the solar vector. This approach is illustrated in Fig. 3b. The Blanco-Muriel et al. (2001) model is used for computing the solar position, which reports a higher accuracy compared to many other models and provides source code. \(N_{dr}\) ray origins are distributed across the surface of the disk in a stratified manner and are randomly displaced a small amount (jittered) to reduce aliasing effects. Source code for the function used to sample points on a disk is available in Shirley and Chiu (1997). For a domain with a diameter \(d\), the amount of energy contained by a direct solar ray is:

\[
E_{s,dr} = \pi d^2 S_{dr}/(4N_{dr}) \quad \text{[W]}.
\]

Diffuse solar radiation is simulated by uniformly distributing \(N_{df}\) rays in a hemisphere about every ground patch, translating the origin outside the domain, then launching them in the direction of the ground patches. This method is illustrated in Fig. 3a. Additional detail on how a hemisphere is sampled is presented in section 2.4. This per-ray energy is weighted by the cosine of the ray zenith angle \(\theta_r\). Because the integral of cosine over the hemisphere is equal to 1/2, this weighting is multiplied by 2 so that energy is conserved. The diffuse
solar energy rate per ray is

\[ E_{s,df} = S_{df} 2 \cos(\theta_r) A_p / N_{df} \quad [W]. \]  

(5)

2.2.2 Longwave transfer

To simulate longwave emission, rays are launched from all patches in a hemisphere about its surface. The energy rate for each longwave ray emitted from a patch is given by

\[ E_\ell = \epsilon \sigma_s T_p^4 2 \cos(\theta_r) A_p / N_\ell \quad [W]. \]  

(6)

where \( \epsilon \) is the patch emissivity, \( \sigma_s \) is the Stefan-Boltzmann constant, \( T_p \) is the absolute temperature of the patch, \( A_p \) is the area of the patch, and \( N_\ell \) is the number of rays launched per patch. When computing \( Q^* \), the total longwave flux emitted from a patch is

\[ L^\uparrow = \epsilon \sigma_s T_p^4 \quad [\text{Wm}^{-2}]. \]  

(7)

Longwave radiation can also be scattered by the atmosphere, but is modeled in a way such that the atmosphere is emitting it. This energy is simulated like diffuse shortwave in which rays are distributed in a hemisphere about each ground patch and launched inward. The energy rate per ray emitted by the atmosphere depends on the total diffuse longwave radiative flux density \( U_{df} \) and ray zenith angle \( \theta_r \). Like \( S_{df} \) and \( S_{dr} \), \( U_{df} \) can be specified directly from measurements or simulated with a model. One diffuse longwave flux model implemented in QESRadiant is the Yang and Li (2013) model, in which \( U_{df} \) is computed from the dew point \( T_{dp} \) (in degrees Celsius), dry bulb temperature \( T_{air} \) (in Kelvin), sky emissivity \( \epsilon_{sky} \), and local hour \( t \) (hours past midnight in local time). \( U_{df} \) is computed as followed:

\[ T_{sky} = T_{air} \sqrt[4]{0.711 + 0.0056 T_{dp} + 0.000073 T_{dp}^2 + 0.013 \cos(15t)} \quad [\text{K}], \]  

\[ U_{df} = \sigma_s \epsilon_{sky} T_{sky}^4 \quad [\text{Wm}^{-2}]. \]  

(8)

Surface weather data sets freely available from MesoWest (Horel et al., 2002) can be automatically loaded by QES to supply \( T_{dp} \) and \( T_{air} \) for a given time, date, and geographical location. If \( \epsilon_{sky} \) is not known, QES has an option to compute it from models described in Chen et al. (1995). The amount of longwave energy emitted by the atmosphere per ray is

\[ E_{\ell,df} = U_{df} 2 \cos(\theta_r) A_p / N_\ell \quad [W]. \]  

(9)
2.2.3 Absorption, reflection, and scattering

To reduce the memory overhead of QESRadiant, only the total net radiation of solar energy \( K_{\text{net}} \) on a patch is stored during the simulation. Incoming and emitted longwave energies are still stored as separate terms since direct access to these terms is usually needed to determine surface temperatures in land surface models (LSMs). Therefore, we can rewrite the net radiation equation modeled by QESRadiant as:

\[
Q^* = K_{\text{net}} + L_{\downarrow} - L_{\uparrow} \quad [\text{Wm}^{-2}].
\] (10)

Incoming shortwave and longwave radiation is absorbed, reflected, and scattered by patches. For a ray with an initial energy rate \( E \), waveband \( \lambda \), and a patch with albedo \( \alpha_\lambda \), the amount of energy absorbed from one ray is \( E(1.0 - \alpha_\lambda) \). The sum of absorbed energy for all rays that intersected the patch \( n \) is used to compute \( K_{\text{net}} \) and \( L_{\downarrow} \). That is,

\[
K_{\text{net}} \approx \frac{1}{A_p} \sum_{i=1}^{n} E_i(1.0 - \alpha_s) \quad [\text{Wm}^{-2}].
\] (11)

\[
L_{\downarrow} \approx \frac{1}{A_p} \sum_{i=1}^{n} E_i(1.0 - \alpha_\ell) \quad [\text{Wm}^{-2}].
\] (12)

The amount of energy that is scattered and reflected depends on the material of the patch. This property is determined by a material’s specular fraction \( \delta_s \) and diffuse fraction \( \delta_d \) where \( \delta_s + \delta_d = 1 \). For example, highly reflective surfaces like glass or glossy paint would have a high \( \delta_s \) and low \( \delta_d \). Conversely, soil or gravel would have a much higher \( \delta_d \) and lower \( \delta_s \). The amount of energy specularly reflected is \( E(\alpha_\lambda \delta_s) \) and the amount diffusely scattered is \( E(\alpha_\lambda \delta_d) \). Thus, each patch is able to diffusely scatter and specularly reflect radiation. This results in a model that is an efficient simplification of a more complicated bidirectional reflectance distribution function (Roujean et al., 1992). However, \( \delta_s \) and \( \delta_d \) are not defined in literature and are only approximations.

If a ray does not intersect a structure and reaches the sky, all of its energy is absorbed by the atmosphere. Otherwise, a ray will continue to be reflected and scattered until it no longer carries a significant amount of energy. Once a ray’s energy drops below a predefined threshold \( \otimes \) the rest of the energy is absorbed by the intersected patch and the ray path is terminated. This approach significantly reduces machine precision errors when dealing with very small ray energies.

For specular reflections, a new ray is launched at the point of intersection.
an incidence angle $\theta_{inc}$, the angle of reflection $\theta_{refl} = \theta_{inc}$ about the surface normal is used, and the azimuthal angle of direction is unchanged (Fig. 3d). Scattered energy is isotropic, so new rays are launched in a hemisphere in the same manner as emitted longwave radiation (Fig. 3c). Unlike specular reflections, rays are not scattered immediately upon incidence. To reduce the number of rays required for scattering radiation, the energy that would be scattered is stored temporarily in memory and is accumulated on a patch over time. When atmospheric, solar, and thermally-emitted rays have ceased reflecting specularly, this accumulated to-be-scattered energy is emitted at once and may continue to be absorbed, reflected, and scattered. Eventually all ray energies will drop below the energy threshold. As a result, the scattering of radiation requires iterative relaunches of rays until all energy has been absorbed. In practice, this process nearly always completes in less than ten iterations.

2.3 Sky view factor

The sky view factor (SVF) may be defined as the fraction of radiation leaving a patch that is intercepted by the sky (Johnson and Watson, 1984). SVF is not explicitly used to model the transfer of radiation within QES. However, due to its utility in urban modeling it can be simulated with QES as a separate module called QESViewfactor (see Appendix A). A number of approximations for the SVF have been proposed (e.g., Kastendeuch, 2013 and references therein). With our approach, an intersection test is performed whereby rays are launched outward in a hemisphere about the center of every patch. The hemisphere is sampled according to section 2.4. If the $i^{th}$ ray reaches the edge of the domain without intersecting a surface, $\varphi_i = 1$, otherwise $\varphi_i = 0$. The SVF is then computed by summing the number of rays that did not intersect a surface, adjusting the value by the cosine of the ray zenith angle $\theta_i$, and dividing by the total number of rays launched per patch ($N_\ell$), that is:

$$F_{sky} \approx \sum_{i=1}^{N_i} \frac{2 \cos(\theta_i) \varphi_i}{N_\ell}. \quad (13)$$

A similar approach for computing view factors was introduced by Wallace et al. (1989) for radiosity calculation. Their methods are similar to Eq. (13) and show how the differential area to finite area form factor is an adequate approximation, with enough samples.
Fig. 3. Illustrations of ray launch patterns. In (a), diffuse solar and atmospheric longwave rays are launched toward the center of every ground patch. Their ray origins are positioned outside the domain in a hemisphere about the ground patch. This shows the launch for one such ground patch, while the three other ground patch launches are not shown. In (b), a sphere that encloses the entire domain is used to compute the domain diameter. A disk with a normal parallel to the solar vector is used for simulating direct solar energy, and is determined by this diameter. Ray origins are stratified across the disk and launched toward the buildings and ground. In (c), radiation is scattered diffusely, and new rays are launched in a hemisphere from a patch center. The rays are evenly distributed, but their energies are weighted by the cosine of the zenith angle. In (d), a new ray is launched as an ideal specular reflection.
2.4 Hemispherical ray emission

Ray directions are distributed on the hemisphere above a patch by computing evenly distributed points on a hemisphere about the patch center. The points on the hemisphere are sampled using a modification of the spiral points algorithm from Saff and Kuijlaars (1997). To obtain a more evenly distributed set of points, the Golden Ratio $(1 + \sqrt{5})/2$ is introduced in the traversal of the spiral nodes. This function is defined in Appendix B.

After the hemisphere has been sampled, the ray directions are rotated by a random azimuthal angle between 0 and $2\pi$. This random number is the same for each ray of the same hemisphere, but will differ from other sampled hemispheres. This reduces the effects of aliasing, in which unrealistic patterns may be introduced due to uniform sampling.

3 Model validation

This model has been both verified and validated according to definitions set by the American Institute of Aeronautics and Astronautics (AIAA, 1998). Verification is not included in this paper for brevity, but includes several tests to ensure the models perform as expected in ideal domains. Specifically, it has been verified that energy is conserved, that simulated view factors between neighboring surfaces compare well to an analytical solution, and that a completely exposed patch or sensor receives the correct amount of energy emitted by the sun/atmosphere.

Six statistical measures were used to validate the models of QESRadiant. These include the refined index of agreement (IA) from Willmott et al. (2012), coefficient of determination ($R^2$), fractional bias (FB), model bias (MB), normalized mean square error (NMSE), and mean error (ME). For an observed or measured data set $X$, a predicted or simulated data set $Y$, the average of a data set overbar, and the number of elements in a data set $n$, these measures are computed as followed:

\[
A = \sum_{i=1}^{n}(|Y - X|), \quad B = 2 \sum_{i=1}^{n}(|X - \overline{X}|),
\]

\[
IA = \begin{cases} 
1 - A/B & \text{if } A \leq B, \\
B/A - 1 & \text{if } A > B.
\end{cases}
\]  

(14)

\[
R^2 = \frac{\left[\sum_{i=1}^{n}(X_i - \overline{X})^2 - \sum_{i=1}^{n}(X_i - Y_i)^2\right]}{\sum_{i=1}^{n}(X_i - \overline{X})^2}.
\]  

(15)
\[ FB = \frac{2(Y - X)}{(X + Y)}. \]  
(16)

\[ MB = \frac{Y - X}{X} \quad [\text{Wm}^{-2}]. \]  
(17)

\[ NMSE = \frac{[(X - Y)^2]}{(X \times Y)}. \]  
(18)

\[ ME = \sqrt{(X - Y)^2} \quad [\text{Wm}^{-2}]. \]  
(19)

IA is an index value designed for measuring model performance and is a more robust measure of model agreement than \( R^2 \). It is dimensionless and bounded by -1.0 and 1.0. \( R^2 \) is a measure of how well the variance is explained by the model, and values less than 0 are set to 0. \( R^2 \) will then take on values over the range: \( 0 \leq R^2 \leq 1 \). For both IA and \( R^2 \), a higher value typically indicates a better model predictor. The fractional bias is the model bias normalized by the average of the model and observation. FB will have a value of \(-2 \leq FB \leq 2\), with zero being a perfect score (no model bias). For positive values of \( X \) and \( Y \), \( FB > 0 \) and \( MB > 0 \) indicates model over-prediction, while \( FB < 0 \) and \( MB < 0 \) indicates model under-prediction. NMSE is the average squared error of the model normalized by the average of the model and observation.

### 3.1 Urban street canyon model validation

Street canyons represent important and well-studied prototypical elements of cities and their microclimates. An urban street canyon can be described as a street or pathway that is bordered by opposing walls. The surfaces within the canyon have less exposure to the sky (smaller SVF) and retain more heat than their open terrain counterparts. Studies have found that urban canyon geometries affect surface energy balance fluxes (Nunez and Oke, 1977). Thus, to validate QESRadiant we recreate an idealization of the Gothenburg street canyon study (Eliasson et al., 2006; Offerle et al., 2007) during which extensive measurements of wind, temperature, and radiation quantities were measured at different locations within a street canyon for over a year. Specifically, we compare QESRadiant’s simulated shortwave and longwave radiation flux densities at various points within the 3D street canyon.
3.1.1 Domain

The original field experiment was conducted in Gothenburg, Sweden (57°42’ N, 11°58’ E). Virtual sensors described in section 2.1.2 and shown in Fig. 5 were positioned in the canyon according to radiometer positions described in Elaisson et al. (2006). Because virtual sensors only record values in one direction, two directionally-opposing sensors were used for each radiometer to record incoming and outgoing shortwave and longwave fluxes. A QES-generated top down visualization of the canyon is shown in Fig. 4. In addition to illustrating its virtual representation, Fig. 4 also shows sky view factor and shortwave radiation visualization capabilities.

Due to the limitations imposed by QES for urban domains, certain concessions were made when modeling the geometry of the canyon. These stem from the requirement that buildings in QES cannot be rotated and must be discretized into grid aligned patches. First, the real urban canyon is not precisely oriented North to South or East to West. To accommodate this, after the solar azimuth is computed by QES for the given time and date, it is increased by an additional 20 degrees. Since this validation experiment is only simulating radiation transfer, the solar direction is the only component affected by the canyon’s non-cardinal aligned orientation. Second, the rooftop of the buildings that make up the canyon walls are slanted in the real world. To approximate this, patches were stacked on the simulated rooftop in a stair-step like pattern (see Fig. 5). Further, minor geometry details of such as specific brickwork, ledges, street lamps, overhead cables, and other small obstructions were not modeled.

The patch materials of the simulated canyon were varied in attempt to accurately recreate the real canyon. All patches were squares with sides of length

---

Fig. 4. Contour plots illustrating (left) sky view factor and (right) shortwave radiation for the simulated urban canyon in Gothenburg, Sweden using QUIC EnvSim.
0.5 m, thus the height and width of the canyon was rounded to the nearest 0.5 m. Because the specific physical properties of the building surfaces were not known, they were assigned the following: $\alpha_s=0.3$, $\alpha_\ell=0$, and $\epsilon=1$. Eliasson et al. (2006) describe the canyon walls as 25% windows, but information on exact location and size of the windows was not available. Thus, we considered the entire wall to be a diffuse material ($\delta_d=1$, $\delta_s=0$).

### 3.1.2 Simulation settings

QESRadiant is able to compute shortwave and longwave atmospheric fluxes as described in sections 2.2.1 and 2.2.2. However, these values were measured by the radiometer that was positioned above the urban canopy (sensor R3). Instead of using models to compute the atmospheric fluxes incident on the canyon, the measured global solar ($K_\downarrow$) was distributed into diffuse and direct components. This was done by first computing $S_{df}$ according to Eq. 3. If the measured global radiation is less than the computed diffuse solar flux (i.e., $K_\downarrow < S_{df}$), then the diffuse radiative flux is set to the measured global radiation ($S_{df} = K_\downarrow$) and the direct solar flux is set to zero ($S_{dr} = 0$). Otherwise $S_{df}$ is used as computed and $S_{dr} = K_\downarrow - S_{df}$. This approach allowed the model to respond to cloud cover in which a reduction of the measured global solar increases the ratio of $S_{df}$ to $S_{dr}$.

As shown in Fig. 5, five simulated radiometers were compared to the experimental data and are labeled based on their position within the urban canyon. These include: upper center – C4, lower center – C2, eastern lower – E2, and western lower – W2. R3 was used to drive the model inputs. Each sensor measured the incoming and outgoing shortwave and longwave fluxes with respect to the lower center of the canyon.

The temperature of the patches within the canyon wall and ground were also driven with measurements. These temperatures were measured in the study using thermocouples on each wall, and one thermocouple on the ground. In our simulation, the entire ground was set to the temperature recorded by the ground thermocouple. Patch wall temperatures were obtained by linearly interpolating between each of three thermocouples (lower, middle, and upper). All other patches not specified from thermocouple measurements were set to 290 K. The simulation was run from 4 July 2003 through 27 March 2004 with 5 minute time steps. The total number of time steps was nearly 74k.

### 3.1.3 Validation results

Figures 6 and 7 are scatter plots paired in space and time, comparing flux density of all radiation sensors. No attempt was made to remove cloudy or otherwise imperfect-condition days from the data set.
Fig. 5: A schematic of radiometer and thermocouple locations for the urban canyon test case used in section 3.1. The original study was performed by Eliasson et al. (2006) and Offerle et al. (2007). Radiometer R3 is not shown.
An overview of statistical measurements is presented in Table 1. To aid in interpretation, Fig. 5 shows the orientation of the radiation sensors. Understanding the orientation is particularly important for interpreting sensors E2 and W2, which where oriented so that the sensors were parallel to the street canyon walls.

**Sensor sampling strategy validation:**

Unobstructed flux values ($S_{drr}$, $S_{df}$, and $U_{df}$) were specified using measurements from R3. Then, the virtual R3 sensor was used to calculate the simulated unobstructed flux, which was compared to the true measurement. This test provides a metric for the effectiveness of our stochastic sampling methodology since we would expect a perfect correlation if the rays are sufficiently sampled. Indeed, radiation fluxes at this location showed high IA ($K^↓$: 1.0, $L^↓$: 0.95), high $R^2$ ($K^↓$: 1.0, $L^↓$: 0.99), low FB ($K^↓$: -0.03, $L^↓$: 0.0), low MB ($K^↓$: -2.7 Wm$^{-2}$, $L^↓$: -0.8 Wm$^{-2}$), low NMSE ($K^↓$: 0.0, $L^↓$: 0.0), and low ME ($K^↓$: 4.7 Wm$^{-2}$, $L^↓$: 5.1 Wm$^{-2}$). Results for R3’s $K^↑$ and $L^↑$ are not reported, because without modeling the entire urban area these values are not fully represented by the model.

**Longwave radiation fluxes:**

As indicated in Table 1 as well as Fig. 6, simulated longwave radiation displayed a strong, linear correlation. C4 $L^↓$ showed the lowest correlation of the eight longwave sensors with an IA of 0.75, $R^2$ of 0.76, FB of -0.07, MB of -19.7 Wm$^{-2}$, NMSE of 0.01, and ME of 24.8 Wm$^{-2}$. This is likely due to the model’s inability to adequately represent the complex rooftop geometry. That is, exposure to atmospheric longwave was less realistic, and rooftop temperatures were not driven with measurements. The other sensors had IA values of $\approx 0.9$, $R^2$ between 0.93 and 0.97, FB between -0.02 and 0.02, MB between -6.2 and 6.0 Wm$^{-2}$, NMSE $\approx 0.0$, and ME between 6.9 and 11.5 Wm$^{-2}$. This high correlation indicates that our methodology for the interpolation of the measured surface temperatures was reasonable and our ray emission strategies are sufficient.

**Shortwave radiation fluxes:**

Most $K^↓$ and $K^↑$ sensors showed satisfactory agreement, with an average IA of 0.79 and average $R^2$ of 0.56. However, all shortwave scatter plots display regular over- and under-prediction, resulting in a branch-like separation (see Figure 7). This behavior may be explained by examining time series plots of shortwave radiation for three consecutive clear-sky days (Fig. 8) for sensors: C4 $K^↓$, C2 $K^↓$, and E2 $K^↑$. Figure 8a illustrates how imprecise canyon geometry may cause over-predictions in the morning and evenings at the top of the canyon (C4 $K^↓$). Figure 8b shows how the predicted shortwave flux at the inner-center of the canyon (C2 $K^↓$) follows closely to the measured flux. How-
ever, the modeled sensor may move between shade and direct solar exposure at a different time step than the measured sensor, causing over/under-predictions of several hundred Wm$^{-2}$. Minor unmodeled obstructions and local geometry (such as street lamps, wires, brickwork, and small reflective surfaces) are likely the cause of many over-predictions, especially noticeable in E2 $K^{\uparrow}$ (Fig. 8c) which had the lowest $R^2$ of 0.0. Another noticeable trend is the hysteresis in E2 $K^{\downarrow}$ (Fig. 7e), also likely caused by imperfect canyon wall geometry offsetting the time of solar exposure.

These results demonstrate how K is highly sensitive to local geometry and time step. Missing the exact moment a sensor moves between shade and solar exposure, even by a few minutes or less, may cause significant under-prediction or over-prediction. Thus, $R^2$ for radiation fluxes paired in space and time does not provide the best measure of model correlation. IA, however, is less sensitive to a missed interval and provides a better measure of model performance.

**Execution Time:**

Hardware specifications are described in Appendix C. The average run time for a single time step over a diurnal cycle was 4.7 seconds on machine C.2. The urban canyon test case consisted of 115,736 patches, 1,024 rays were used per patch hemisphere, and 61.44M rays were used for direct solar radiation. The number of rays per hemisphere was determined to be sufficient based on preliminary analysis by Halverson (2012) and unit test verification experiments.
<table>
<thead>
<tr>
<th>Sensor</th>
<th>IA</th>
<th>$R^2$</th>
<th>FB</th>
<th>MB (Wm$^{-2}$)</th>
<th>NMSE</th>
<th>ME (Wm$^{-2}$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>R3 K↓</td>
<td>1.0</td>
<td>1.0</td>
<td>-0.03</td>
<td>-2.7</td>
<td>0.0</td>
<td>4.7</td>
</tr>
<tr>
<td>R3 L↓</td>
<td>0.95</td>
<td>0.99</td>
<td>0.0</td>
<td>-0.8</td>
<td>0.0</td>
<td>5.1</td>
</tr>
<tr>
<td>C4 L↓</td>
<td>0.75</td>
<td>0.76</td>
<td>-0.07</td>
<td>-19.7</td>
<td>0.01</td>
<td>24.8</td>
</tr>
<tr>
<td>C4 L↑</td>
<td>0.89</td>
<td>0.94</td>
<td>-0.01</td>
<td>-3.3</td>
<td>0.0</td>
<td>9.2</td>
</tr>
<tr>
<td>C2 L↓</td>
<td>0.88</td>
<td>0.93</td>
<td>0.02</td>
<td>6.0</td>
<td>0.0</td>
<td>11.5</td>
</tr>
<tr>
<td>C2 L↑</td>
<td>0.93</td>
<td>0.97</td>
<td>0.01</td>
<td>1.7</td>
<td>0.0</td>
<td>6.9</td>
</tr>
<tr>
<td>E2 L↓</td>
<td>0.92</td>
<td>0.96</td>
<td>-0.01</td>
<td>-1.8</td>
<td>0.0</td>
<td>7.9</td>
</tr>
<tr>
<td>E2 L↑</td>
<td>0.90</td>
<td>0.96</td>
<td>-0.02</td>
<td>-6.2</td>
<td>0.0</td>
<td>8.4</td>
</tr>
<tr>
<td>W2 L↓</td>
<td>0.91</td>
<td>0.96</td>
<td>-0.01</td>
<td>-3.8</td>
<td>0.0</td>
<td>7.8</td>
</tr>
<tr>
<td>W2 L↑</td>
<td>0.90</td>
<td>0.96</td>
<td>-0.01</td>
<td>-3.2</td>
<td>0.0</td>
<td>8.1</td>
</tr>
<tr>
<td>C4 K↓</td>
<td>0.92</td>
<td>0.90</td>
<td>0.19</td>
<td>14.0</td>
<td>0.40</td>
<td>47.9</td>
</tr>
<tr>
<td>C4 K↑</td>
<td>0.67</td>
<td>0.51</td>
<td>-0.08</td>
<td>-0.5</td>
<td>0.99</td>
<td>5.7</td>
</tr>
<tr>
<td>C2 K↓</td>
<td>0.86</td>
<td>0.79</td>
<td>0.14</td>
<td>4.9</td>
<td>1.5</td>
<td>43.1</td>
</tr>
<tr>
<td>C2 K↑</td>
<td>0.71</td>
<td>0.64</td>
<td>-0.37</td>
<td>-1.0</td>
<td>1.6</td>
<td>3.4</td>
</tr>
<tr>
<td>E2 K↓</td>
<td>0.77</td>
<td>0.37</td>
<td>0.26</td>
<td>1.5</td>
<td>2.8</td>
<td>9.6</td>
</tr>
<tr>
<td>E2 K↑</td>
<td>0.73</td>
<td>0.0</td>
<td>0.38</td>
<td>4.0</td>
<td>6.8</td>
<td>27.0</td>
</tr>
<tr>
<td>W2 K↓</td>
<td>0.88</td>
<td>0.86</td>
<td>-0.02</td>
<td>-0.1</td>
<td>0.98</td>
<td>5.3</td>
</tr>
<tr>
<td>W2 K↑</td>
<td>0.81</td>
<td>0.44</td>
<td>0.32</td>
<td>4.2</td>
<td>3.3</td>
<td>24.0</td>
</tr>
</tbody>
</table>

Table 1
Refined index of agreement (IA), coefficient of determination ($R^2$), fractional bias (FB), model bias (MB), normalized mean square error (NMSE), and mean error (ME) for the urban canyon test case with QESRadiant. Perfect scores for each measure include: IA and $R^2 = 1$; FB, MB, NMSE, and ME = 0. Sensor locations and direction of fluxes are shown in the urban canyon schematic (Fig. 5).
Fig. 6. Longwave radiation fluxes for all sensors of the urban canyon test case. The simulated period was 4 July 2003 through 27 March 2004 at 5 minute intervals.
Fig. 7. Shortwave radiation fluxes for all sensors of the urban canyon test case. The simulated period was 4 July 2003 through 27 March 2004 at 5 minute intervals.
Fig. 8. Time series of radiation sensors for the period covering 30 July 2003 – 1 August 2003. Shortwave radiation is highly sensitive to local geometry. Unmodeled obstructions like lamp posts and imprecise rooftop geometry may result in significant over and under-predictions.
4 Computational performance

It is difficult to compare the computational efficiency of QES to other urban modeling frameworks and systems. Others (e.g., Bruse, 2015; Krayenhoff and Voogt, 2007; Matzarakis et al., 2006) do not report details on simulation run time or hardware portability. Therefore, in this paper, we report simulation run times for all test cases presented, without comparing to other modeling frameworks. These run times are shown in their respective sections (urban canyon test case: section 3.1.3, Salt Lake City SVF case: Appendix A). We also present an experiment to demonstrate the scalability of QESRadiant in section 4.1. While all run time results in this paper are for a single GPU, multi-GPU performance is presented in Overby (2014). All reported run times do not include start up or compilation, but do include memory transfers between the host and device.

4.1 Domain scaling and capacity

This section attempts to quantify what is meant by the proposition that QES is designed to handle large and complex domains. The following test consists of measuring QESRadiant’s run time for an expanding scene (i.e., domain). For each iteration, new 1 m$^3$ buildings are added one meter above a ground surface in a checkerboard-like pattern as illustrated in Fig. 9b. While not representative of a real urban layout, it provides a complex domain to stress test the system in which ray launches result in a high number of intersections at different structures. The domain was expanded until an OptiX exception error indicated that the available memory had been exceeded. Run times were recorded along with domain size. The experiment was run on a single device of a GeForce GTX 690 (normally a multi-device GPU, see Appendix C.2). The sample size for diffuse and patch-emitted longwave rays ($N_{df}$, $N_{ℓ}$) was 256. Direct ray sample size ($N_{dr}$) was a function of the number of patches on the ground plane times 256. Instead of setting the latitude, longitude, date, and time, the solar zenith angle was forced to 45 degrees to ensure complex shortwave interaction.

QESRadiant was able to run a scene containing over 8.02 million patches in 6 minutes and 10 seconds. After this point, the simulation required more memory than was available on the GPU (2GB). Fig. 9a shows that as the size of the domain increased, run time increased linearly.
Fig. 9. Scalability of QESRadiant until hardware limitations are met. In (a), machine C.2 was able to run QES radiant for a domain of over 8 million patches in 6 minutes and 10 seconds. In (b), the test domain continued to expand, adding additional 1 m$^3$ buildings until available GPU memory was exceeded.

5 Summary

QUIC EnvSim is a high performance programmable modeling framework for simulating mass, momentum, and heat transfer in complex urban environments. Modules can be developed within the framework to simulate specific components of the urban microclimate and be combined with other new or existing modules.

This paper introduces the QES module QESRadiant, which computes the transfer of radiation between the ground, buildings, vegetation, and the atmosphere. Buildings are represented as grid-aligned three-dimensional boxes with positive, arbitrary dimensions. A building can be made up of several boxes that can be stacked or placed next to others. The faces of each box are further segmented into smaller two-dimensional planes called patches. QESRadiant simulates longwave and shortwave transfer using the GPU accelerated ray tracing engine NVIDIA OptiX. This is done by emitting rays from patches, vegetation, the sun, and the atmosphere, with an initial energy. When the ray intersects a terrestrial object, fractions of that energy are absorbed, reflected, and scattered. Energy is recorded at every patch within the domain and is used to compute the net radiation component of the surface energy balance.

By leveraging OptiX and CUDA, QESRadiant can model the radiation balance of approximately 8.02 million patches in 6 minutes and 10 seconds (for one time step) on a single desktop workstation with 2GB of GPU memory. Simulation run time linearly increases with domain size. Therefore, as GPUs are allocated...
more dedicated memory and improve in performance, so will the capabilities of QESRadiant.

The models of QESRadiant were validated by simulating a study of an urban street canyon in Gothenburg, Sweden as described by Eliasson et al. (2006) and Offerle et al. (2007). A 3D urban canyon provides an excellent testbed for validation because the surfaces within the canyon have less exposure to the sky and retain more heat throughout the day, affecting the surface energy balance. Radiometers were located at various positions within the canyon, and wall temperatures were driven with measured values from thermocouples. The simulations ran from 4 July 2003 until 27 March 2004 with 5 minute intervals for a total of nearly 74k simulations. Six statistical measures were used to gauge model performance: refined index of agreement (IA), coefficient of determination ($R^2$), fractional bias (FB), model bias (MB), normalized mean square error (NMSE), and mean error (ME). Simulated longwave correlated with observed longwave exceptionally well, with averages IA: 0.88, $R^2$: 0.93, $|FB|$: 0.02, $|MB|$: 5.7 Wm$^{-2}$, $|NMSE|$: 0.0, and ME: 10.6 Wm$^{-2}$. Measures for shortwave energy appeared to be statistically sensitive to the time step and local geometry. Virtual and measured sensors would go between shade and direct solar energy at different times. This is likely due to imprecise building roof geometry and minor obstructions (e.g., street lamps, wires). This occurrence resulted in regular over- and under-predictions, and made $R^2$ less adequate for illustrating model accuracy. However, IA is not as sensitive to the missed time step and indicated acceptable agreement. Shortwave had averages IA: 0.79, $R^2$: 0.56, $|FB|$: 0.22, $|MB|$: 3.8 Wm$^{-2}$, $|NMSE|$: 2.30, and ME: 20.8 Wm$^{-2}$.

This validation test case elucidated several important points. First, there is a need for high-quality urban experimental data sets like the Gothenburg, Sweden case to test models like QES that can simulate at the half-meter scale. Further, high resolution geometric details and properties of the urban fabric are needed for model validation. Simulating the balance of radiation in urban spaces with ray tracing provides a new level of physical realism, and can aptly be applied to a wide range of domains.

6 Acknowledgments

This material is based upon work supported by the National Science Foundation under the following grant numbers: 0828214, 1134580 and 1133590. We are grateful to all of the GEnUSiS project team members for their substantive contributions, particularly, Prof. Rob Stoll. We appreciate the development efforts by Daniel Alexander at the early stages of the project, for his contributions toward verifying view factors and the development of a simplified surface energy balance. Dr. David Schroeder was also of great help, in which
the use of his visual analysis tools helped in the verification of QES models. We are indebted to Drs. Brian Offerle and Fredrik Lindberg and collaborators for providing us with the street canyon validation data (as well as key information about the data) from Gothenburg, Sweden. Finally we thank Dr. Michael Brown and Prof. Sue Grimmond for allowing us to use their sky view factor images and data from Salt Lake City.

Appendix A: Salt Lake City sky view factor test case

Sky view factors are an important measurement when analyzing the effects of urban form on the microclimate. In a study by Brown et al. (2001), SVF measurements were recorded throughout downtown Salt Lake City, Utah at various locations. These measurements were calculated from fish-eye photographs taken by an upward pointed camera with a hemispheric lens. The latitude, longitude, and height of the camera placement for a given measurement were recorded in addition to the SVF. The study took place in October, 2000 during the DOE CBNP URBAN field experiment.

To validate the ability of QESViewfactor to simulate SVF, we compare simulated values to measurements from Brown et al. (2001). In Fig. A.1b, each black bordered circle is a camera placement. The inner color of the circle represents measured SVF reported by Brown et al. (2001). In addition to computing SVF, QES can also generate fish-eye views. Figure A.2 shows side-by-side comparisons of QES-generated fish-eye views and photographs by Brown et al. (2001). In the QES-generated image, a blue dot represents a ray that reached the sky, while a black dot indicates it was blocked by a building. The photograph was taken with a digital camera using a Nikon FC-E8 hemispheric lens. Note that fish-eye views are generated by QES for visual comparisons. The method by which Brown et al. (2001) uses to compute SVF is not used. SVFs are calculated in QESViewfactor using the methods defined in section 2.3.

The scatter plot showing correlation can be seen in Fig. A.1a. However, two conditions related to limitations of the building data set cause inaccuracies between the simulated and reported SVFs. First, discrepancies in building geometries are evident, as the data set is not an exact representation of downtown Salt Lake City, Utah. In some cases, entire buildings are missing as illustrated in Fig. A.2b. Second, the absence of vegetation may cause large errors, especially in park or greenspace areas as shown in Fig. A.2c.

Simulating SVFs for Salt Lake City took 0.51 seconds on machine C.2. The domain consisted of 201,647 patches, and 1,024 rays were used per patch hemisphere.
Fig. A.1. Comparison of predicted to measured (Brown et al., 2001) SVF for downtown Salt Lake City, Utah. (a) shows a scatter plot of sky view factor comparisons. Shown measures are index of agreement (IA), fractional bias (FB), and normalized mean square error (NMSE). (b) shows contours of SVF for downtown Salt Lake City, Utah, computed with QES. The inner color of the black bordered circles represents measured SVFs.
Fig. A.2. Comparison of simulated (left) and photographic (right) fish-eye images for Downtown Salt Lake City, Utah. Photographs are used with permission from Brown et al. (2001). Limitations in data sets like approximated building dimensions or missing buildings/vegetation may cause errors in model agreement.
Appendix B: Functions and formulae

B.1: Hemispherical ray direction

Points on the hemisphere are sampled using a modification of the spiral points algorithm from Saff and Kuijlaars (1997). The index of the ray is used to compute the direction in 3D space.

<table>
<thead>
<tr>
<th>Function hemisphere direction</th>
</tr>
</thead>
<tbody>
<tr>
<td>ray idx is the nth ray of a hemisphere</td>
</tr>
<tr>
<td>offset = 1 / samples per hemisphere</td>
</tr>
<tr>
<td>y = ray idx * offset − 1 + ( offset / 2 )</td>
</tr>
<tr>
<td>r = sqrt( 1 − y^2 )</td>
</tr>
<tr>
<td>phi = ray_id * Pi * ( 3 − sqrt(5) )</td>
</tr>
<tr>
<td>direction = &lt; cos( phi ) * r, y, sin( phi ) * r &gt;</td>
</tr>
<tr>
<td>rotate to project in direction of patch normal</td>
</tr>
<tr>
<td>return direction</td>
</tr>
</tbody>
</table>

Appendix C: Device specifications

C.1: GeForce GTX Titan

<table>
<thead>
<tr>
<th>OS:</th>
<th>Ubuntu 12.04 LTS</th>
</tr>
</thead>
<tbody>
<tr>
<td>RAM:</td>
<td>64 GB RAM</td>
</tr>
<tr>
<td>CPU:</td>
<td>8-core Intel Xeon E5-2690, 3.00GHz</td>
</tr>
<tr>
<td>GPU:</td>
<td>NVIDIA GeForce GTX Titan 6GB GDDR5</td>
</tr>
<tr>
<td>Software:</td>
<td>NVIDIA OptiX 3.0.0, CUDA 5.0</td>
</tr>
</tbody>
</table>
C.2: GeForce GTX 690

<table>
<thead>
<tr>
<th>OS:</th>
<th>Ubuntu 12.04 LTS</th>
</tr>
</thead>
<tbody>
<tr>
<td>RAM:</td>
<td>16 GB RAM</td>
</tr>
<tr>
<td>CPU:</td>
<td>4-core Intel i7-3820, 3.60GHz</td>
</tr>
<tr>
<td>GPU:</td>
<td>NVIDIA GeForce GTX 690, 2GB GDDR5 per GPU</td>
</tr>
<tr>
<td>Software:</td>
<td>NVIDIA OptiX 3.0.1, CUDA 5.0</td>
</tr>
</tbody>
</table>

References


doi:10.1016/j.uclim.2015.11.004


