Development of a composite model for predicting urban surface temperature distribution in the context of GIS

Development of a composite model for predicting urban surface temperature distribution in the context of GIS

Ziang Cui[1,2], Thomas Leduc[1], Auline Rodler[2,3] and Marjorie Musy[2,3],

[1] Nantes Université, ENSA Nantes, École Centrale Nantes, CNRS, AAU-CRENAU, UMR 1563, F-44000 Nantes, France
[2] IRSTV, FR 2488, F-44000, Nantes, France
[3] CEREMA, BPE, F-44000 Nantes, France

ziang.cui@crenau.archi.fr

Abstract. In this paper, we present a new model for simulating radiation heat transfer and surface temperatures in urban environments, which is a prerequisite for simulating outdoor mean radiant temperature for pedestrians. We propose a new model whose novelty lies in a new algorithm for simulating diffuse radiation by combining the Nusselt unit sphere method and Monte Carlo ray tracing algorithm. This combined use of different methods is automatically activated depending on the complexity of the scene and, in particular, the arrangement of the facets facing each other. The model is implemented in a Python-based tool, t4gpd, which combines the capabilities of geographic information science and related technologies (GIS &T) with efficient ray-casting solutions. The model is tested on a theoretical mock-up of nine virtual buildings in Nantes under summer and winter weather conditions and compared with SOLENE-Microclimat results. The view factor and solar radiation flux simulation results agree well with the reference solution, with a significant reduction in simulation computation time. However, the model has limitations due to the exclusion of vegetation evapotranspiration characteristics, wind profile model, and soil with complete stratigraphy. Overall, this new model provides an efficient tool for simulating outdoor pedestrian thermal comfort in urban environments and has potential for further development and validation.

1. Introduction

The urban environment is a dynamic and complex system that is constantly evolving due to urbanization and climate change. The temperature difference between the urban fringe and the urban core, known as the urban heat island (UHI), has significant impacts on the urban environment, including increased energy consumption for cooling, deterioration of air quality, and negative impacts on the health of residents [1]. Therefore, the development of an urban surface temperature simulation model has become an important research topic that can shed light on the complex interactions among various urban factors and provide information for urban planning and management strategies to mitigate the urban heat island effect.

Over the years, several different urban surface temperature simulation tools have been developed. SOLWEIG [2] is a Python plugin under GIS software, which applies a parameterization of the surface temperature for the different land covers without considering the inter-reflection of solar radiation within the city [3]. To account for radiative transfer, one of the most important tasks is to determine

the view factor. It indicates the fraction of radiant energy exchanged between different surfaces and is closely related to the propagation of radiation. Solene-Microclimat addresses this challenge with an analytical contour integration method [4], which can be considered as a deterministic model. It offers the advantage of low error, but its scope is limited due to lack of flexibility. CityComfort+ is based on Radiance, which uses the Monte Carlo ray tracing method [5] for radiative transfer simulation. It can be considered as a stochastic model that provides flexibility and adaptability since it generates results based on random samples. However, a disadvantage of all these methods is their high computational cost, especially for complex 3D models.

The aim of this paper is to develop a simplified and GIS tool that employs a novel view factor calculator that combines the strengths of both deterministic and stochastic model methods. To account for the differences in accuracy, computational effort, and scope, we evaluate these methods separately in different scenarios with varying complexity in urban environments. In the meantime, thanks to the Python tool t4gpd [6], this model was able to convert a 2D footprint into 3D model for simulating radiative heat transfer in GIS software. To validate this model, we compared it with SOLENE-Microclimat. The input values include the 3D model divided into a triangular irregular network with the same orientation, geographic location, and uniform time and climate conditions. We generate three sets of results to verify the accuracy of our view factor calculator and to assess our capability in simulating radiative heat transfer, including incoming view factor, solar radiation, and surface temperature.

2. Methods

2.1. Input data preprocessing

We illustrate our model in a theoretical 3D mockup of an orthogonal grid consisting of nine randomly distributed virtual buildings with random heights corresponding to a standard city configuration. To achieve a more accurate representation of the surface and the distribution of surface temperature, this 3D mockup is subdivided into a triangular irregular network, using the three-dimensional finite element mesh generator GMSH [7] (see Figure 1a). Moreover, all surfaces are set to the same thickness.

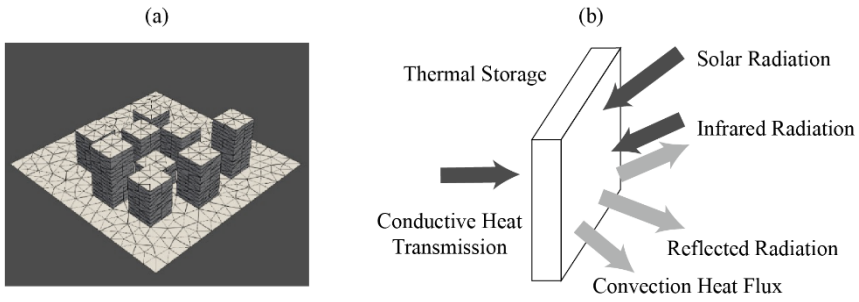

Figure 1. (a) Geometry of 3D mockup. (b) Energy balance for urban surface

2.2. Thermal balance of building surfaces

In our proposed model, urban surfaces are treated as independent objects that do not consider the conductive contribution. For each individual surface, we consider the following heat transfer mechanisms: solar radiation, infrared radiation from the environment (including the atmosphere and surrounding surfaces), reflected radiation, thermal storage and convective heat exchange (as shown in Figure 1b). Conductive heat transfer is not considered because it would require knowledge of the internal temperature and properties of the materials involved, which is difficult to obtain. The local energy equilibrium equation is as follows:

$$E_{storage} + E_{solar} + E_{IR} + E_{convection} = 0 \quad (1)$$

Where $E_{storage}$, E_{solar}, E_{IR}, $E_{convection}$ are the thermal storage, solar radiation received by surface, infrared radiation heat exchange and convective heat transmission. Thermal storage can be represented by the following equation:

$$E_{storage} = \rho C_p d \frac{\mathrm{d}T}{dt} \quad (2)$$

Where ρ, C_p, d, dT, dt are density, heat capacity, thickness of the building material, temperature variation of surface, and timestep respectively. This expression illustrates the relationship between thermal storage and the physical properties of the material, as well as the rate of temperature change, which is useful for simulating surface temperature variations.

2.3. Surface temperature simulation flowchart

The surface temperature is derived using the radiosity method [8]. The first step is to determine the view factor after mesh generation by GMSH. Then, utilizing the weather data and the corresponding time period, the solar radiation (including direct, diffuse, and reflected solar radiation) is calculated since it is independent of the building surface temperature. For timestep *i*, the infrared radiation (E_{IR}) is calculated based on the building surface temperature obtained from the previous timestep *i-1*. Then, the equilibrium equation (1) is applied to obtain the updated surface temperature. If the temperature variation falls below a predefined threshold, the simulation proceeds to the next timestep. Otherwise, the updated surface temperature is recalculated until the temperature variation falls below the threshold. The flowchart for the surface temperature simulation looks as follow:

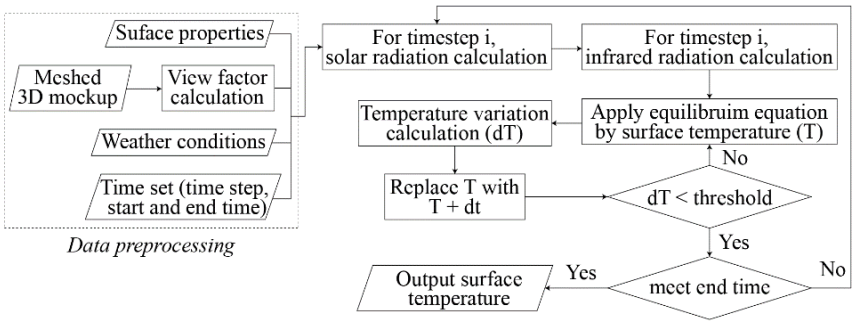

Figure 2. Flowchart of surface temperature simulation

3. View factor calculation optimization

The view factor plays an important role in energy exchange, which is highly dependent on size, shape, relative orientation, and distance from surfaces. The view factor is the fraction of radiant energy transmitted from source surface S_j to target surface S_i (symbols used throughout this paper). The flow accepted by S_i is proportional to the view factor $F_{j\to i}$, which is defined by a double surface integral. However, direct integration introduces difficulties to evaluate view factor analytically except in some special cases (aligned parallel rectangles, perpendicular rectangles with a common edge, etc.). The state-of-the-art methods of calculating view factor can be divided into two main categories:

- Deterministic model: determination of the view factor between an infinitesimal and a finite area. e.g. Infinitesimal surface approximation method (ISA), Nusselt unit sphere method, etc.
- Stochastic model: determination of the view factor by statistical sampling. e.g. Monte Carlo ray tracing method.

3.1. Optimized infinitesimal surface approximation method

Infinitesimal surface approximation method is a simple vector calculation first introduced by J. J. MacFarlane in 2003 [9]. This method has a great advantage in terms of calculation timing. However, it suffers from accuracy problems when the surfaces are too close to each other and we cannot treat the surface as infinitesimal. Below is the flowchart of our optimized ISA method using the '5 times rule' which guarantees an accuracy of less than 2.5% [10] (Figure 3a). To obtain a reliable evaluation of the corresponding view factor, the distance between two surfaces should be greater than five times the maximum projected width of the source surface.

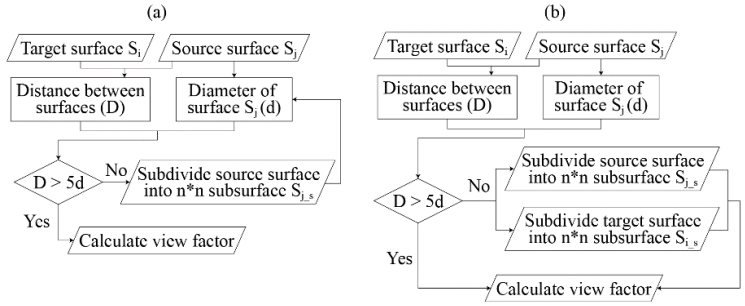

Figure 3. (a) Flowchart of optimized ISA method. (b) Flowchart of optimized Nusselt method.

3.2. Optimized Nusselt unit sphere method

The Nusselt unit sphere method was once developed by Wilhelm Nusselt as an experimental method [11], whose principle evolved from the definition of view factor. A double projection mechanism of the target surface gives the view factor. However, it suffers from the same problem as ISA method which could be optimized by adapting the '5 times rule'. This optimization is reflected in the flowchart by the inclusion of additional steps for dividing the source surface in Figure 3b.

3.3. Optimized view factor calculator

After careful examination of three cases in Section 4.1, we present the final model for calculating the view factor in Figure 4. The critical aspect of this model is to determine the distance between the two planes and whether they share the same edge before calculating the view factor.

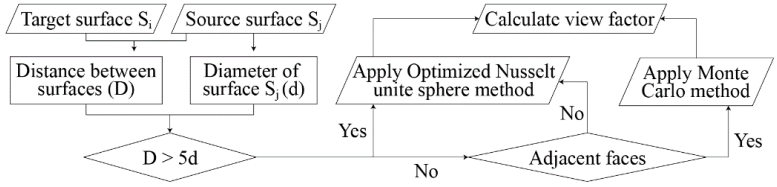

Figure 4. Flowchart of optimized view factor calculator

4. Results

4.1. View factor calculator performance

In order to test the performance of the three simulation methods mentioned previously (optimized ISA method, optimized Nusselt unit sphere method, and Monte Carlo ray tracing method), it is necessary to consider the different configurations of surfaces that may occur. These configurations can be divided into three categories (Figure 5c):

- Parallel faces, where two surfaces are oriented in the same direction without intersecting.

- Perpendicular faces with a common edge.
- Non-parallel faces, but perpendicular to the ground.

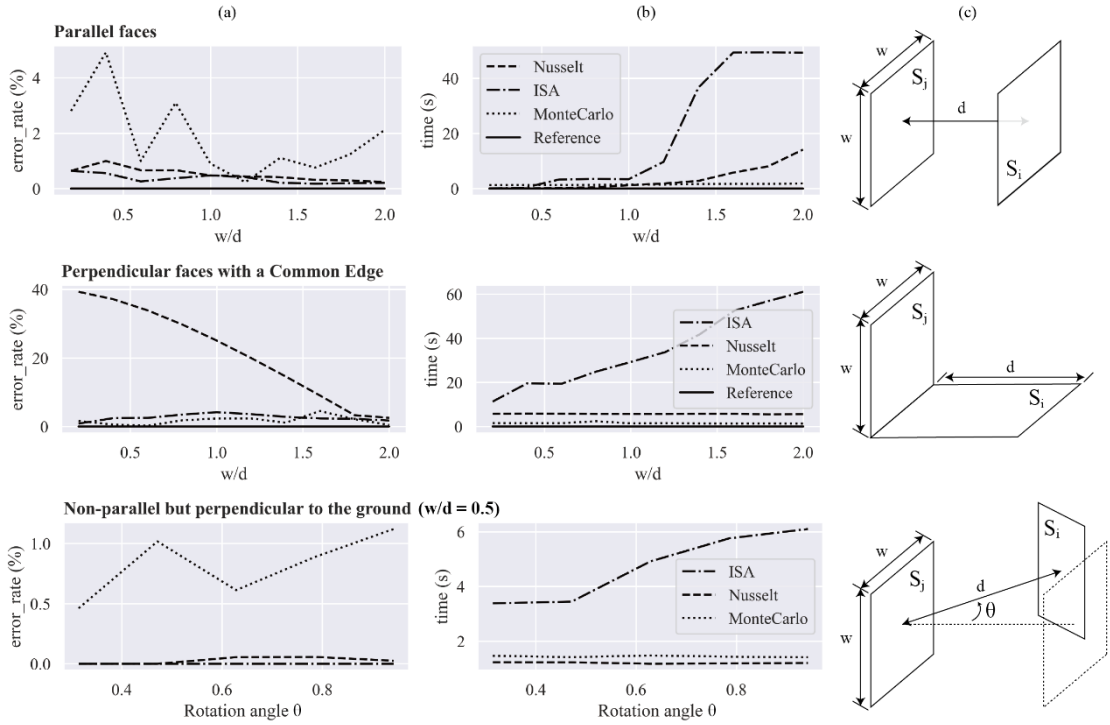

Figure 5. Performance of three methods (reference value can be determined numerically in Parallel faces and Perpendicular faces [12]), including (a) Error Rate Performance, (b) Computational Timing Results, (c) 3D Plane Illustration.

It should be noted that the reference value of view factor can be determined numerically only for parallel faces and perpendicular faces, while we use the output of ISA method as the reference value in third case. From Figure 5, it can be seen that Nusselt and ISA methods perform well in scenarios where there are no common edges and provide stable and accurate results. Although ISA method provides the most stable and accurate results, its computation time is much longer than the other two methods, especially when two faces are close to each other. Although Nusselt method provides high computational efficiency, it is not suitable for scenarios with common edges. The Monte Carlo method, on the other hand, provides high accuracy in all three cases and is particularly suitable for applications where faces share a common edge or the distance between them is smaller than their own diameter.

Table 1. Dataset for simulation model.

Latitude	47.16	Albedo	0.5
Longitude	-1.60	Emissivity	0.94
Cloudiness (octa)	2	Density (kgm^{-3})	2400
Relative humidity (%)	40	Initial temperature (°C)	16.6
Air temperature (°C)	20	Heat Capacity ($Jkg^{-1}K^{-1}$)	910
Wind speed (ms^{-1})	10	Thickness (m)	0.1
Date	16/08/20	Time-step (hour)	1

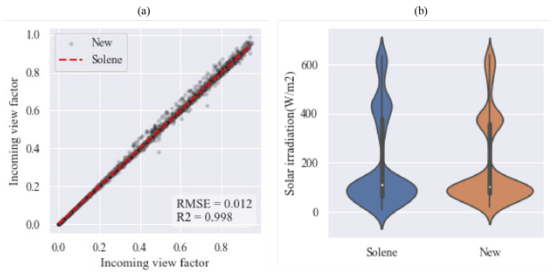

Figure 6. (a) Incoming view factor. (b) Solar irradiation at 12 p.m.

4.2. Surface temperature simulator performance

Using Table 1 as input values, we calculate three components on both model: the incoming view factor, solar irradiation, and surface temperature. The incoming view factor represents the sum of view factors of each surface to the visible surfaces, while solar irradiation encompasses both direct and indirect solar radiation received by each plane. By observing Figure 6a, we can find that our view factor calculator performs very well, and its results fit well with the results of Solene. The difference in the value of solar irradiance in Figure 6b is mainly due to the difference in the algorithms used by the two models to determine whether the surface is under sunlit. In the end, the average error rate of the surface temperature simulated by two models is lower than 6%.

5. Conclusions

In our work, we developed a simplified simulator for urban surface temperatures, aiming for a balance between simulation accuracy and cost. The model captures the radiative transfer process in urban environments, but still requires further development. Comparison of our results with Solene-Microclimat showed good agreement. Future work includes incorporating more experimental data and exploring multiprocessing to improve simulation speed. Our model is a valuable tool for urban design and planning decisions, but further research in this area is needed.

References

[1] Li, D., & Bou-Zeid, E. (2013). Synergistic interactions between urban heat islands and heat waves: The impact in cities is larger than the sum of its parts. *Journal of applied Meteorology and Climatology*, *52*(9), 2051-2064.

[2] Lindberg, Fredrik, Björn Holmer, and Sofia Thorsson. "SOLWEIG 1.0–Modelling spatial variations of 3D radiant fluxes and mean radiant temperature in complex urban settings." *International journal of biometeorology* 52 (2008): 697-713.

[3] Gaitani, Niki, et al. "EVALUATION OF THE SOLWEIG MODEL WITH HIGH_RESOLUTION SPATIAL MEASUREMENTS OF SURFACE TEMPERATURES."

[4] Miguet, F., & Groleau, D. (2002). A daylight simulation tool for urban and architectural spaces—application to transmitted direct and diffuse light through glazing. *Building and environment*, *37*(8-9), 833-843.

[5] Huang, J., Cedeno-Laurent, J. G., & Spengler, J. D. (2014). CityComfort+: A simulation-based method for predicting mean radiant temperature in dense urban areas. *Building and Environment*, *80*, 84-95.

[6] Leduc T. (2022). t4gpd (0.5.0). Zenodo. https://doi.org/10.5281/zenodo.5771916

[7] Geuzaine, C., & Remacle, J. F. (2009). Gmsh: A 3-D finite element mesh generator with built-in pre-and post-processing facilities. *International journal for numerical methods in engineering*, *79*(11), 1309-1331.

[8] I. Ashdown, *Radiosity - a programmer's perspective.* 01 1994.

[9] MacFarlane, J. J. (2003). VISRAD—A 3-D view factor code and design tool for high-energy density physics experiments. *Journal of Quantitative Spectroscopy and Radiative Transfer*, *81*(1-4), 287-300.

[10] Baum, D. R., Rushmeier, H. E., & Winget, J. M. (1989). Improving radiosity solutions through the use of analytically determined form-factors. *ACM Siggraph Computer Graphics*, *23*(3), 325-334.

[11] Wilhelm Nusselt, Graphische bestimmung des winkelverhaltnisses bei der wärmestrahlung, *Zeitschrift des Vereines Deutscher Ingenieure*, 72(20):673 1928.

[12] Bergman, T. L., Bergman, T. L., Incropera, F. P., Dewitt, D. P., & Lavine, A. S. (2011). *Fundamentals of heat and mass transfer*. John Wiley & Sons.