PAPER • OPEN ACCESS

Procedural modeling buildings for finite element method simulation

View the article online for updates and enhancements.

Procedural modeling buildings for finite element method simulation

G Besuievsky[1], E García-Nevado[1], G Patow[1] and B Beckers[2]

[1]ViRVIG, Universitat de Girona, Girona, Spain
[2]Urban Phyics Joint Laboratory, E2S UPPA Université de Pau et des Pays de l'Adour, France

E-mail: `gonzalo@imae.udg.edu`

Abstract. Finite element methods for heat simulation at urban scale require mesh-volume models, where the meshing process requires a special attention in order to satisfy FEM requirements. In this paper we propose a procedural volume modeling approach for automatic creation of mesh-volume buildings, which are suitable for FEM simulations at urban scale. We develop a basic rule-set library and a building generation procedure that guarantee conforming meshes. In this way, urban models can be easily built for energy analysis. Our test-case shows a street created with building prototypes that fulfill all the requirements for being loaded in a FEM numerical platform such as Cast3M (`www-cast3m.cea.fr`).

1. Introduction

Describing heat exchanges at building or district scales is an important issue in order to propose guidelines for sustainable urban design (see Figure 1). Simulating thermography by Finite Element Method (FEM) is an attractive approach when accurate solutions are pursued for representing and visualizing long-wave radiation exchanges for digital models [1]. FEM provides a solution of partial differential equations through a system of algebraic equations with a finite number of unknowns nodes, and require the discretization of the geometry into elements. In the case of 3D buildings, this mesh should consist of volumetric elements. Also, the resulting mesh must be conforming to ensure suitable simulation conditions. To date, there is no available solution for automatically preparing geometrical models for FEM simulation. Furthermore, all volumetric and surface properties, such as conductance or reflectance, should be appropriately set for all elements. Making this work by hand is a daunting, repetitive and error-prone task for any new building and that takes several working hours.

In this paper we propose a procedural modeling approach to improve model mesh generation to obtain a general automatic solution for FEM. Procedural modeling techniques are currently used for algorithmic efficient building and city generation in the entertainment industry, mainly for creating contents for computer games and movies. In general, these approaches focus on surface-based modeling. Here we propose a volume-based approach to introduce a set of procedural rules for volumetric mesh element generation of building models. We provide a parametric methodology that allows creating building models, including the basic architectural elements with their corresponding semantics required for indoor/outdoor energy simulations. Using this approach, we obtain urban models that comprise architectural elements with significant energy impact but that are rarely present in urban thermal simulations, such as

windows, doors, floors, and balconies. The main contribution of our work is to provide a tool for easily designing building configurations that may benefit thermal simulation analysis at both building and district scales.
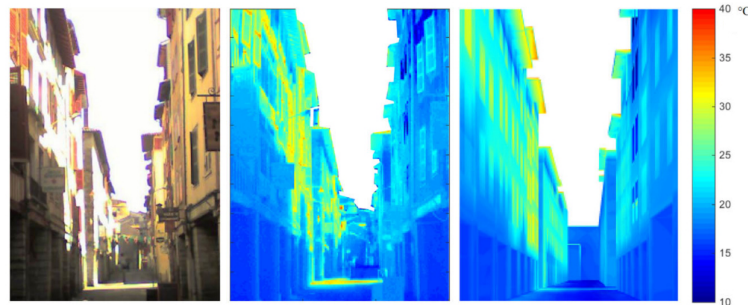

Figure 1. Urban thermography. Photograph of the street (left), measured (middle) and simulated thermographies (right) [1].

2. Building model generation

The initial requisite of FEM is re-meshing the input geometry. In mechanics and most 3D physics studies, the most common choice is discretizing it using tetrahedral elements. For architectural applications, using hexahedra is more convenient since buildings are usually dominated by vertical and horizontal flat surfaces comprising several layers [2]. In such geometries, the use of hex meshes significantly reduces the number of mesh elements and, thus, the model size. This choice also makes it easier to control the number of layers and their subdivisions, a critical parameter regarding final accuracy. Unlike tetrahedra, hexahedra might form structured meshes, which allow for more efficient information storage systems [3]. These computational advantages can be particularly relevant in urban studies, which can easily exceed a hundred thousand mesh elements even for a small street.

Furthermore, meshes for FEM simulations should be conforming, meaning that adjacent elements must share the contact face completely (see Figure 2). The common practice to fulfill this requirement is to use generic mesh generators (e.g., Gmsh) to discretize a 3D model created with CAD tools. Acuña et al. [4] proposed an alternative approach specially conceived for architectural applications. The idea was to generate the building model and its mesh, all at once, by respecting the following modeling rules that allow for meeting both architectural logic and FEM requirements (see Figure 3):

- A building is an assembly of floors, roofs, and walls, all modeled as parallelepipeds. The intersections between wall-wall or floor-wall create new parallelepipeds in between. These new volumes stand for pillars or beams in actual buildings, commonly associated with thermal bridges (Figure 3a).

- The openings for doors and windows are holes in the walls with a frame. These elements represent the structural reinforcements formed by lintels, sills, and jambs. For conforming reasons, the vertical and horizontal frame surfaces will re-subdivide the entire building model (Figure 3b).

- Building models should include facade protrusions with significant shading effects, such as eaves and balconies. These elements are modeled as new volumes, extruded upon the opening frames or floor slabs (Figure 3c).

- By default, building components can comprise multiple layers. Each one acts as a cutting plane for the rest of the model to maintain the mesh conforming (Figure 3d).

- Due to their crucial role in the energy behavior of buildings, windows deserve separate consideration. They are modeled as thin volumes, meshed independently from the rest of the building. In this way, it is possible to achieve greater accuracy in solar load computations without excessively increasing the number of mesh elements (Figure 3e).

Different thermal and daylight studies have relied on a street model developed in standard CAD tools following these rules [1], evidencing the potential of this modelling approach and, at the same time, the limitations for its manual implementation: too time-consuming, prone to geometrical errors and hard to modify. Procedural techniques are a promising way to explore to overcome these limitations.
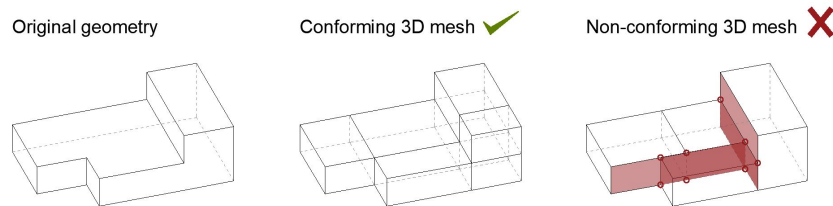

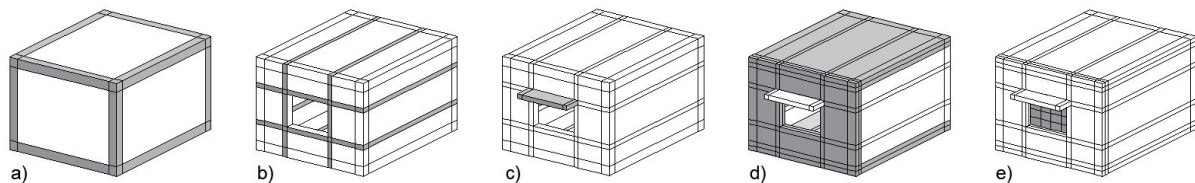**Figure 2.** Conform and non-conform examples

Figure 3. Generation rules

In this work we develop a procedural method to automate the process. One of the attractive features of procedural techniques is that they operate through command rules that can be parameterized. By customizing parameters, it is possible to model real cases and alternative scenarios with minimal effort and avoiding geometrical errors, and to generate models with different levels of refinement while maintaining its architectonic coherence. Procedural techniques can also help to enrich model semantics more straightforwardly by creating labeling systems. This capability facilitates not only the assignment of material properties to the model, but also the readability of the simulation results.

3. A procedural volume approach
Our main goal is to create a user-friendly tool to generate volumetric mesh buildings with the requirements explained in Section 2. For this purpose, we provide a system for prototype creation that guarantees conforming and can be exported for FEM simulation. We develop a procedural volume rule-set to create generic prototypes. Then, the prototypes are instantiated and configured with setting parameters to generate a 3D volume model. The following sections describe the details of our system.

3.1. Procedural volume rules
Our method is based on a rule-based shape grammar [5], that is, starting from an initial axiom shape, rules are iteratively applied, replacing shapes with other shapes. Similar to

Barroso and Patow's work [6], we used a volume-based approach that implements volume rules to generate conforming volume-mesh building, also adding material properties. Our main procedural commands are (see Figure 4):

- **CreateVol** creates a volume at a 3D *center* point with dimension $sizeX/Y/Z$ for each direction.
  $A \rightarrow CreateVol(sizeX, sizeY, sizeZ, center)\{B\}$

- **SubdivVol** splits a volume in a given *axis* direction with dimension $size_i$ into sub-volumes
  $A \rightarrow SubdivVol(axis)\{size_1 : B1| \ldots |size_n : Bn\}$

- **ReapeatVol** repeats a subdivision in a given axis direction as many times as possible with dimension *size*
  $A \rightarrow RepeatVol(axis, size)\{B\}$

- **ReplaceFace** replaces a face *id* of a volume by extruding a volume in the normal direction with dimension *size*
  $A \rightarrow ReplaceFace(id, size)\{B\}$

- **Filter** filters a set of *labels* for a given product, where *labels* can be any tag associated to the product. Boolean operations (intersection, union or subtraction) of tags can be also applied for filtering. This is a useful rule for assigning material properties and classifying architectural elements.
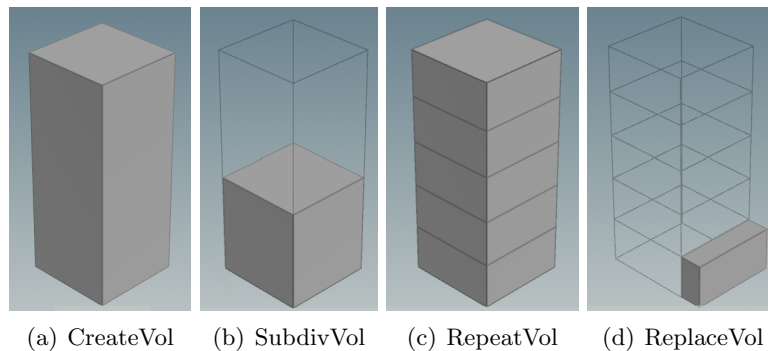 $A \rightarrow Filter(labels)\{B\}$

(a) CreateVol   (b) SubdivVol   (c) RepeatVol   (d) ReplaceVol

Figure 4. Volume Rules

3.2. Modeling procedure

Our building modeling procedure follows the concepts steps described in Section 2, but replacing manual tasks with procedural rules. First, the mass volume is created from a parallelepiped body using $CreateVol$. Then, the successive steps create the roof, the walls for all facades and floors, using a sequence of the $SubdivVol$ and $RepeatVol$ commands (Figure 5). To guarantee model conforming, these split commands should be applied to all elements in a given direction. This is an important difference comparing to the classical procedural modelling of facades with surfaces. The following steps model facade details using also the splitting commands combined with the $Filter$ command. For the facade protrusions, such as solar protections or balconies, we use the $ReplaceFace$ command. Finally, openings with glazing surfaces are modeled as special mesh-volume elements, where the size of the mesh can also be set. An important issue is that all generated products are labeled with semantic meaning (i.e., wall, door, or window) and material properties (e.g., concrete, wood, or insulation). These tags are maintained throughout the generation process and used to associate the actual material properties in the FEM simulation.
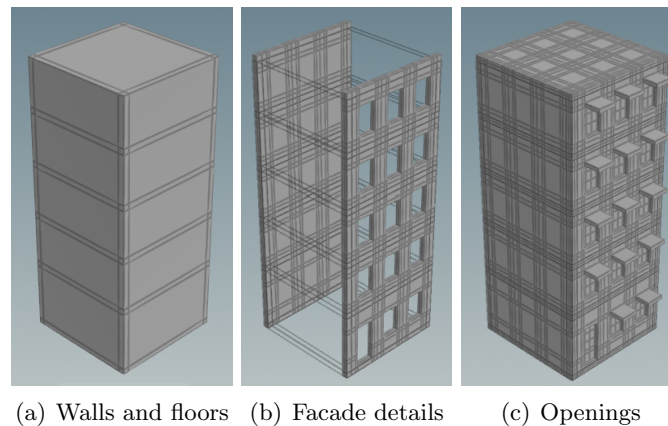
(a) Walls and floors   (b) Facade details   (c) Openings

Figure 5. Modeling Procedure

3.3. Implementing building prototypes

Our method is implemented on top of SideFX's Houdini platform (`www.sidefx.com`). All commands described in Section 3 are implemented as a combination of network nodes and embedded Python code. For the prototypes, we use the Houdini Digital Assets (HDA), where a new node type can be defined and used to build libraries. For the parameter settings, we use Houdini's channel linking mechanism. Based on this approach, we can create a user interface that allows for interactively designing the main building components (Figure 6).
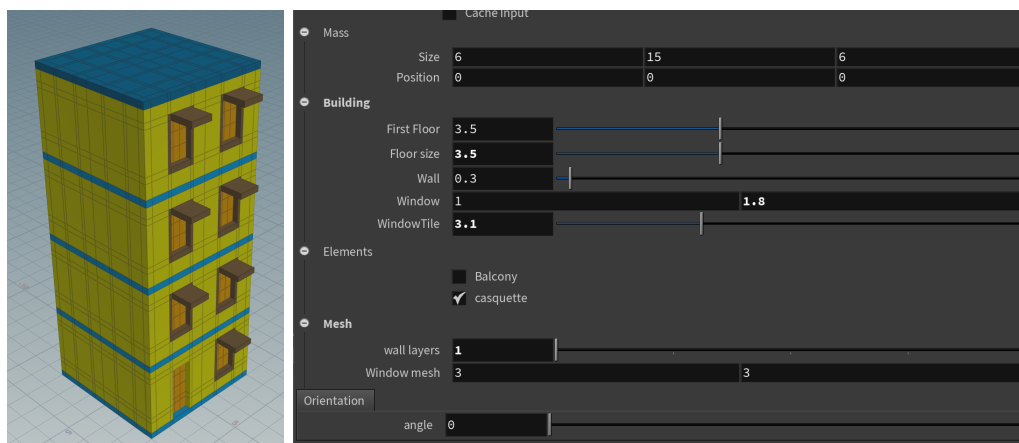

Figure 6. Building prototype (left) and user interface (right). Any construction parameter can be set to modify the instantiated building.

4. Result: A district model creation

We show results of a district model that can be load into the Cast3M platform. The whole building set and the street pavement are automatically created from our code. First, we create a layout of the street, blocks, and building lots. Then, for each footprint, a building prototype is instantiated, facing the street. Any construction parameter can be modified through the user interface to rebuild the model. Finally, the geometric model and all contained labels are exported to the OBJ format for Cast3M. In this example, the whole street is composed of 10 buildings with random heights that generate 188k volume elements.
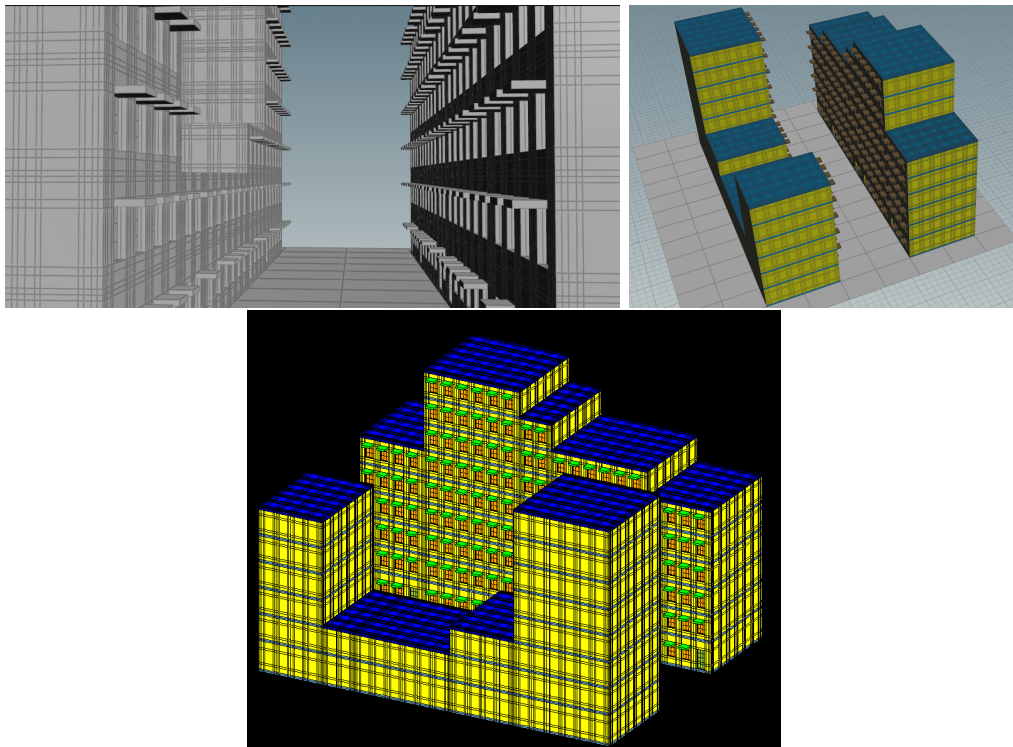
Figure 7. Two views of the district model generated (top) and loaded in Cast3m (bottom)

5. Conclusion and future work

Our procedural volume method provides a tool to automatically generate conforming building models suitable for FEM simulation at the district scale. The main advantages of this approach are that it avoids the manual daunting modeling work and that the resulting model is parametric. That is, the properties of any building element, such as the thickness of a wall layer, can be easily modified to regenerate the model. We believe that our method may encourage detailed simulations at a district scale, essential for a better understanding of the urban surface temperatures and, thus, city climate.

Our future work includes the development of new commands to generate roof variations, the treatment of non-rectangular building footprints and the creation of a library of typical architectural prototypes for further use in real study cases.

References

[1] Aguerre J P, García-Nevado E, Acuña J, Fernández E and Beckers B 2020 *Computer Graphics Forum* **39** 377–391
[2] Garcia-Nevado E and Beckers B 2021 *Radiative Exchanges in the Dense City* (Springer International Publishing) pp 321–349
[3] Lo D S 2018 *Finite element mesh generation* (Taylor & Francis Group)
[4] Acuña J, Lefort V, Lawrence C and Beckers B 2018 *EduBIM2018 : Journées de l'Enseignement de la Maquette Numérique et du BIM en France*
[5] Müller P, Wonka P, Haegler S, Ulmer A and Van Gool L 2006 *ACM Trans. Graph.* **25** 614–623
[6] Barroso S and Patow G 2012 *Spanish Computer Graphics Conference*