Engineering with Computers (2007) 23:109–122
DOI 10.1007/s00366-006-0053-7

ORIGINAL ARTICLE

Dimensional reduction of 3D building models using graph theory and its application in building energy simulation

Christoph van Treeck · Ernst Rank

Received: 19 April 2006 / Accepted: 14 August 2006 / Published online: 21 November 2006
© Springer-Verlag London Limited 2006

Abstract Since the various people involved in the design process for a building project tend to hold conflicting views, this inevitably leads to a range of disparate models for planning and calculation purposes. In order to interpret the relevant geometrical, topological and semantical data for any given building model, we identify a structural component graph, a graph of room faces, a room graph and a relational object graph as aids and explain algorithms to derive these relations. We start with a building model by transferring its geometrical, topological and semantical data into a volume model, decomposing the latter into a so-called connection model and then extracting all air volume bodies and hulls of the model by means of further decomposition into elementary cyclic connection components. The technique is demonstrated within the scope of building energy simulation by deriving both a dimensionally reduced object model required for setting up a thermal multizone model and a geometrical model for defining single or multiple computational fluid dynamic domains in a building together with incidence matrices correlating these models. The algorithm is basically applicable to any building energy simulation tool.

C. van Treeck (✉) · E. Rank
Computational Civil and Environmental Engineering,
Technische Universität München, Arcisstr. 21,
80290 Munich, Germany
e-mail: treeck@bv.tum.de

E. Rank
e-mail: rank@bv.tum.de

Keywords Air volume body extraction · Building information model (BIM) · Building energy simulation · Building product modeling · Dimension reduction · Geometrical and topological analysis · Industry Foundation Classes

1 Introduction

1.1 Linking CAGD with numerical simulation

When it comes to linking computer-aided geometric design (CAGD) tools with numerical simulation techniques applied in civil engineering, one of the major problems to be overcome is the range of conflicting views of a building project, as proposed by the different people involved in the design process. The planning and calculation models derived from these individual concepts vary considerably. For example, the architect's room-based planning differs significantly from a HVAC (heating, ventilating and air-conditioning) or a structural engineer's view of a building and its components. The collaboration of these disciplines is accordingly a complicated process due to the prevailing absence of a common model.

There is also a tendency to avoid applying detailed simulation techniques during the design process because of the costs involved in defining numerical models. What is needed to solve the time-consuming, error-prone process of obtaining and sharing these data is a means of exchanging data between the construction and simulation applications. In our opinion, the Industry Foundation Classes (IFC) represent a promising approach toward ensuring software interoperability in the building industry [10]. The IFC are an

object-oriented, semantical model of all components, attributes, properties and relationships of and within a building project. The model is designed to support information models of different building and construction domains—data related to the design process, the completion, the whole life cycle up to the disposal of the respective building.

Assuming a 3D building model is available—not yet common practise in the design process—partial models have to be derived by *interpreting* data with respect to a specific task—such as creating a new, separate model for performing either a structural or a thermal simulation. Semantics and object relations accordingly play an important role, sometimes the transfer also leads to a change in the data's level of detail. The latter may be the case if, for example, a mechanical model or a zonal thermal model is derived from a geometric model. Moreover, multiphysical simulations require a knowledge of the linkage within the model hierarchy wherever coupling strategies are implemented.

The object of this paper is to discuss this topic in the context of building energy simulation based on results emerging from a research project [20] aimed at coupling thermal building energy simulation with computational fluid dynamics (CFD) methods.

1.2 Physical context

While focusing on building energy simulation for demonstrating the application of the developed decomposition technique, we also briefly sketch the physical background. Whole building energy performance simulation on an annual basis with high temporal resolution (seconds to hours) restricts the spatial resolution to a rough zonal discretization. Common methods make use of an anisotropic finite volume like method [4] by establishing energy balances involving short and long wave radiation processes, transient heat conduction through the structure, surface convection and enthalpy changes. Heat flux through the structure is thereby approximated in a one-dimensional form for each component. The discretization generally leads to a set of nonlinear equations with unknown temperatures to be solved at each time-step. Mass flow rates between different zones, each represented by either a single node or a few nodes, are computed using an artificial nodal network with pressure values as unknowns. The resulting nonlinear equation set is simultaneously coupled with the aforementioned thermal network. Both sets have to be solved iteratively. For an office building, for example, it is usual to obtain equation sets with a few hundred to a thousand unknowns that have to be solved at each time-step for a whole year's simulation.

As, on the other hand, high-resolution techniques, such as detailed CFD calculations for resolving more complex flow patterns, are limited to single zones and short time intervals, we developed in [17] a partitioned solution approach in order to establish a coupling between a CFD code [19] and a thermal multizone building model. We accordingly started with algorithms previously proposed by Beausoleil-Morrison [2] and Hensen [9]. The numerical coupling of both approaches makes it possible to create a CFD model with realistic boundary conditions for local thermal comfort studies. In turn, the applicability of zonal models may be extended to typical scenarios like natural ventilation in building design.

1.3 Algorithmic requirements

Bearing the physical context described above and its mathematical model in mind, we can summarize the algorithmic requirements with respect to the geometrical discretization of a building model, as depicted in Fig. 1.

A *geometrical model*, such as a boundary represented (B-rep) or a constructive solid geometry (CSG) model [3], serves as the basis for establishing a CFD model. It is equipped with *attributes* such as boundary conditions. Depending on the numerical method, an initial surface mesh is used for creating a tetrahedral volume mesh, for example. Other methods, such as lattice Boltzmann techniques, apply tree-based spatial

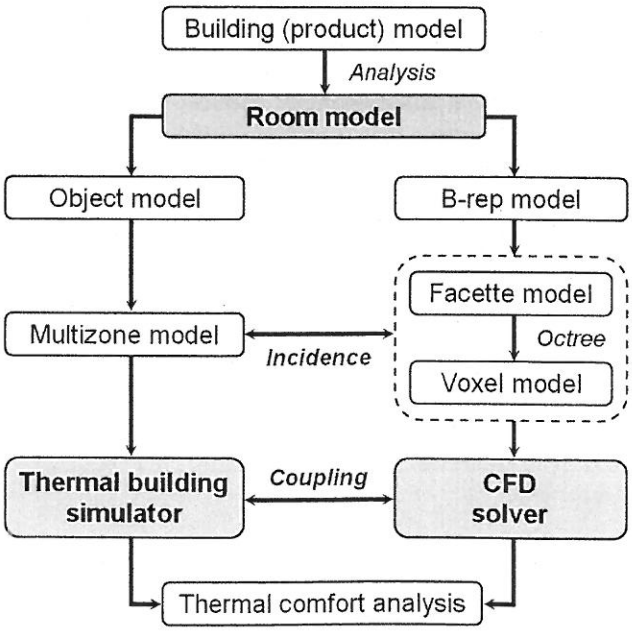


Fig. 1 Sequence of operations within discretization process

algorithms to create a 3D voxel model, starting from a triangulated surface mesh [22].

A *zonal model* in turn is a dimensionally reduced model. It can be described by an object model that represents the building structure in a hierarchical manner, i.e. the model is organized in storeys, rooms, building components, layers, materials, etc.

In order to automatically set up both models—starting from a unique building model—we therefore have to derive a dimensionally reduced object model and a geometrical (B-rep) model. The prerequisites for establishing a numerical coupling between both approaches are *incidence matrices* relating models and components. In other words, a CFD simulation requires volume bodies of air volumes together with boundary conditions while a thermal multizone simulation basically needs a collection of building or plant components with information on their interconnections.

This paper accordingly presents a technique based on graph theory that allows for interpreting the geometrical, topological and semantical data of a building (product) model as well as for automatically extracting indoor air volume bodies and hulls contained within the model, as indicated in Fig. 2. The figure shows a single floor of an office building with corresponding air volume bodies (four in this case) obtained by our algorithm which is implemented in a CAD system.

Furthermore, the relations obtained by the developed spatial and topological analysis may serve as basis for queries by means of a spatial query language [7] for building information models (BIMs).

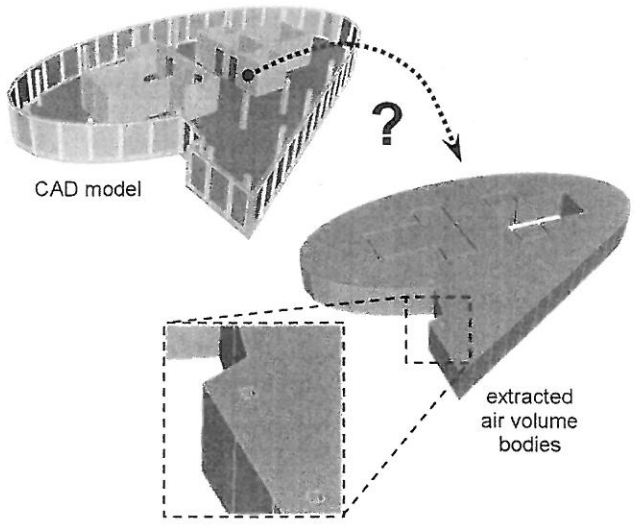


Fig. 2 Single floor of multistorey office building and corresponding set of air volumes (four in this case)

1.4 Review: common practise

There are several common techniques for extracting air volume bodies from a CAD models;

- One well-known method makes use of a plan view or a horizontal projection of a model. A 2D surface defined by a polygonal shape constructed with a specific number of wall components is extruded to form single volume bodies—which is also known as a sweeping model. There are obviously certain limitations, as a unique floor height is assumed and connections between rooms cannot be taken into account, see the application in [14], for example.
- Convex hull algorithms [12] can be applied for creating volume bodies. A point cloud for input purposes may be obtained from the surfaces of surrounding objects. Simplifications such as the assumed convexivity restrict the applicability of the approach, as shapes with re-entrant angles are not permitted. However, buildings are especially 'affected' by enclosed columns and typically non-convex shapes, for example.
- Using z-buffering techniques, the plan aspect of a model is decomposed into a number of horizontal slice planes, each slice plane is rastered into a uniform grid with predefined cell size. Rooms are recognized as connected sets of pixels showing a consistent floor height. The major drawback of this approach is that the inner nodes are user-assigned in order to specify their room context. Pixel-based room objects are unsuitable candidates for a CFD simulation; relations between room nodes and surrounding geometric objects are not unique, as geometric objects require further decomposition (see Sect. 3 below). However, this approach is suitable for obtaining a graph describing paths between rooms within a building model, as required for shortest path searches in emergency scenarios, as demonstrated in [6].

The approaches known to the authors are accordingly unsuitable for obtaining room volumes that can serve as a basis for deriving a room model of the kind required for a zonal discretization of a building model, for meshing objects with respect to fluid simulation and for correlating the model hierarchy.

2 Definitions

2.1 Conventions

In order to distinguish the conventional operators intersection, union and difference operating on sets

from boolean operations operating on geometrical objects, the latter are preceded by a $\cap_b, \cup_b, \setminus_b$. Relations of topological nature are indicated by t by the notation $\in_t$ of topological sub elements.

2.2 Geometrical model

We initially consider for the subsequent analysis a set of rigid bodies given by a CAD system in the context of typical AEC objects[1] used in building design such as walls, windows, doors, columns, beams, plates, shells, etc.

For the definition of rigid bodies we follow Bungartz et al. [3]. Rigid bodies are translationally and rotationally invariant, represent three-dimensional structures, exhibit neither isolated points nor isolated or dangling edges or surfaces and are bounded, regular and semianalytic subsets of $\mathbb{R}^3$. For the precise mathematical description in terms of point set theory it is referred to [3] and the references therein.

The surface representation of a body in terms of a B-rep model further requires that surfaces are closed shells, that faces possess an orientation (law of Möbius) and that faces are not allowed to intersect one another [3].

2.3 Topological model

The topological structure of the considered bodies is described by a *vef*-graph[2] [3]. As the local regularity of a radial-edge model [21] will be of considerable advantage in terms of the geometrical and topological analysis we select this scheme as data structure. The basic structure was chosen in accordance with the ACIS geometric kernel [5]. The C++ based implementation provides means to import and export ACIS entities.

The hierarchy of our radial-edge data structure implements vertices, edges, coedges, loops, faces and bodies. Edges are aware of their start/end vertices and geometry and thus have an orientation. A face object aggregates a normal vector and a list of loops and 'possesses' geometry. Loops describe the polygonal shape of sub-faces by each aggregating a list of coedges. Coedges themselves serve as a topological element, point to their underlying edge and have an orientational sense with respect to this edge. Edges aggregate a set of all coedges pointing to themselves, the so-called partner coedges. Coedges are aligned in a mathematically positive sense around a corresponding face normal and vice versa if representing a hole. Coedges of faces coinciding at common edges have a reversed sense of orientation if these faces belong to the shell of a valid closed B-rep body as defined by the law of Möbius, cf. [3], see Fig. 3.

[1] Architecture, engineering and construction

[2] *vertex, edge, face*

3 Connection model definition

Prior to the analysis, we transfer geometrical data contained in a building (product) model into a solid B-rep volume model. Following the definitions given in Sect. 2, the set of these volume objects is defined as collectivity $\Omega \subset \mathbb{R}^3$. During this process, the building fabric is analyzed with respect to its layer structure. Multilayered components with intersections, such as walls, are blended accordingly. We use the interface of a toolbox system [8] for parsing physical IFC data. The procedure is described in [17] in detail.

The collectivity Ω is decomposed into the so-called *connection model* M_B. All elements c_i of the set M_B are again rigid bodies according to the above given definitions. In order to accomplish the decomposition according to Fig. 4, we need—in addition to the graphs which will be defined in the next section—a relational graph

$$G_C = (\Omega; R_C), \quad R_C \subseteq \Omega \times \Omega \tag{1}$$

with relation R_C denoting plane connections between all n components ω_i of Ω. At those locations with coinciding elements of Ω (see left-hand side of Fig. 4), components are decomposed into *coupling objects* $a_i, b_i \in M_K$ and *difference objects* $d_i \in M_D$, each again representing a rigid body $c_i \in M_B$.

Mathematically, the classification of these objects arises from their interface types within the model hierarchy. It is important to require that, in the sense of a *vef*-graph-based data structure, local intersections between difference objects result in common edges and/or nodes only. This behavior will be a determining

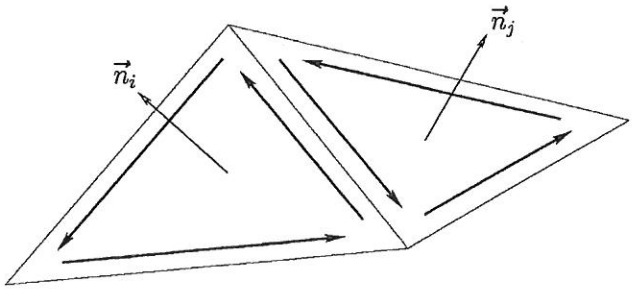


Fig. 3 Coedges with reversed sense of orientation

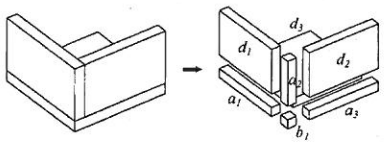


Fig. 4 Decomposition into connection model

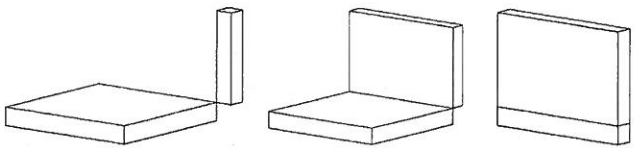


Fig. 5 Interface types: *type v* (*left*), *type ve* (*center*) and *type vef* (*right*)

factor in the subsequent analysis. Using the *vef*-graph representation, interface types are classified according to Fig. 5:

- *Type vef.* The intersection of components in plane contact yields surfaces, edges and vertices.
- *Type ve.* If objects are connected by an edge, the intersection yields edges and vertices.
- *Type v.* Connected objects have common vertices only.

Geometrical intersections between difference objects—if not empty—are always of type *v* or *ve* (and not of type *vef*), while intersections between coupling objects or between coupling and difference objects are of type *v*, *ve* or *vef*.

Consider both components ω_i and ω_j of Fig. 6 which are in plane contact. Where the front surface of object ω_i abuts on the surface of object ω_j we identify the *imprint* face $f_{i,j}$. Component ω_j is subsequently decomposed into a coupling object a_1 and a difference object d_2 using Boolean operations according to Algorithm 1 [17].

To obtain the difference model, all elements are processed that are located in the upper triangular

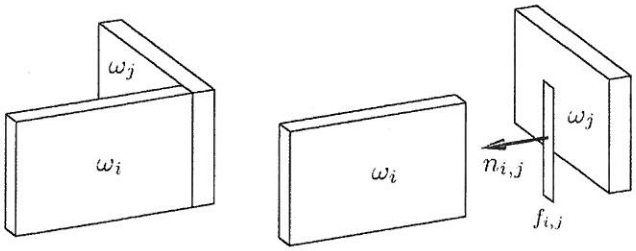


Fig. 6 Identification of imprint face $f_{i,j}$ and normal vector $n_{i,j}$ of components ω_i and ω_j

Algorithm 1 Decomposition into connection model

```
1: for i = 0; i < n; i++ do
2:   for j = i + 1; j < n; j++ do
3:     if R_C(i,j) = true then
4:       Do decomposition:
5:         Create imprint face f_{i,j}
6:         Create normal vector n_{i,j}
7:         Copy ω_i to ω_i'
8:         Translate ω_i' along -n_{i,j} by Δd
9:         Copy ω_j to ω_j'
10:        Get connection objects:
11:          a_1 := ω_i' ∩_b ω_j' ⊂ M_K1
12:          d_1 := ω_i ⊂ M_D
13:          d_2 := ω_j \_b a_1 ⊂ M_D
14:      end if
15:    end for
16:  end for
```

matrix of the symmetric relation R_C (without reflexive relations). The set of coupling objects M_K can be further subdivided into a set M_{K1} of connection bodies and the set M_{K2} of connection bodies linking objects of the partial model M_{K1} by again applying the above described algorithm. We therefore define

$$M_{K1} \subseteq M_K\,, \quad M_{K2} \subset M_K\,, \quad M_{K1} \cap M_{K2} = \varnothing\,, \tag{2}$$

$$M_{K1} \cup M_{K2} = M_K \subset M_B\,. \tag{3}$$

Having r coupling elements a_i with $i = 1,...,r$ and s coupling objects b_j of the connection model with $j = 1,...,s$ and further t difference objects d_k with $k = 1,...,t$, we can write

$$\bigcup_{i=1,...,r} a_i = M_{K1} \subseteq M_K\,, \tag{4}$$

$$\bigcup_{j=1,...,s} b_j = M_{K2} \subset M_K \quad \text{and} \tag{5}$$

$$\bigcup_{k=1,...,t} d_k = M_D \subseteq M_B\,. \tag{6}$$

The combination of both sets M_K and M_D results in the set of all n connection objects M_B with $n = r + s + t$:

$$M_K \cup M_D = M_B\,. \tag{7}$$

The recursive decomposition process is detailed by Romberg et al. [16]. For implementational aspects and the treatment of special cases it is referred to [17].

4 Graph definition

We identify four graphs as necessary for analyzing the topological structure of a building and the relations

between its individual components [18]. These graphs are defined as follows.

It should be stressed that these graphs are in general not provided by a CAD model, even if it is a question of a building product model or a parametric system. If any information is available, it usually comes in the form of relations describing mutually connected wall components together with blending shapes or associated links between AEC objects, both of course useful for the design process.

Given the geometrical model of a building, we create a *structural component graph* (cf. Fig. 7)

$$G_B = (M_B; R_{PC}), \quad R_{PC} \subseteq M_B \times M_B \tag{8}$$

which defines the relation R_{PC} of plane connections between the set of all B-rep volume bodies $c_i \in M_B$ of the connection model. To be more precise, it denotes the occurrence of plane connections between all coupling and difference objects. The resulting structural component graph is an undirected, symmetric graph.

Using the local regularity of the radial-edge data structure [21], the topological and geometrical relations R_{NF} between all the faces of the solid model $f_i \in M_F$ can be derived by the *graph of room faces* (cf. Fig. 7)

$$G_F = (M_F; R_{NF}), \quad R_{NF} \subseteq M_F \times M_F \tag{9}$$

which is necessary to extract a set of closed B-rep bodies of the model, each representing an indoor air volume body. M_F is thereby defined as the set of all surfaces of all components of the connection model M_B. Accordingly, the sense of orientation is an important property of the faces. Based on these relations, we determine the *room graph* (cf. Fig. 8) by partitioning G_F into equivalence classes and subsequent condensation, which is described in Sect. 5. M_{AV} is the set of all indoor air volume bodies and hull faces contained in the model. Details are given in Sect. 5.5.

$$G_{AV} = (M_{AV}; R_R), \quad R_R \subseteq M_{AV} \times M_{AV} \tag{10}$$

If we know the set of objects $r_i \in M_{AV}$, we can classify components of M_B. For example, walls can be identified as being outside, interzonal or inside walls. The latter analysis requires the definition of the relation R_I, which defines adjacencies between components M_B and air volume/hull objects of M_{AV}, and can be expressed by the *relational object graph* (cf. Fig. 8), which is defined as

$$G_I = (M_B, M_{AV}; R_I), \quad R_I \subseteq M_B \times M_{AV}. \tag{11}$$

The following sections describe the fundamental procedure required to derive these relations and gives their precise definitions.

5 Partitioning into equivalence classes

Based on the connection model definition and the structural component graph, we now extract the set of closed B-rep shells M_{AV} contained in the model, i.e. all indoor air volumes and hulls. The idea is to recursively analyze adjacency relations between all faces of objects which are part of the set M_B.

As specified by the algorithmic requirements in Sect. 1.3, we provide means for *detecting* indoor air volumes rather than drawing new objects manually, for *semantically identifying* components together with their interconnections and for *relating* hierarchic models.

The proposed graph-theory approach makes it possible to detect arbitrary sets of closed and non-closed bodies within a model—independently if they are either convex or concave or if they exist in a manifold or non-manifold environment. For the sake of simplicity, we will focus on objects with plane surfaces. Note that the approach is of an intrinsic topological nature and

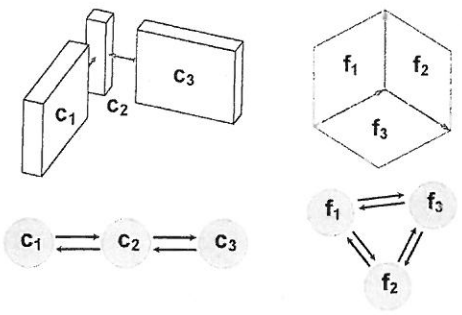


Fig. 7 Structural component graph (*left*) and graph of room faces (*right*)

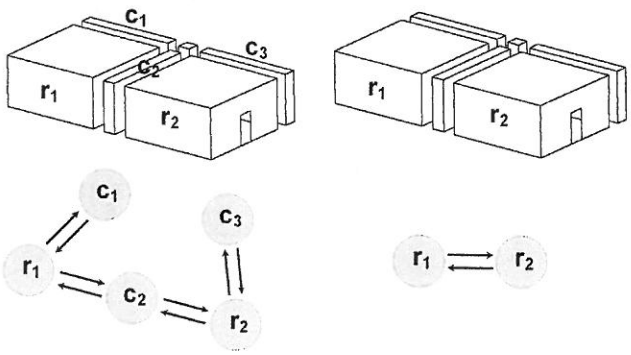


Fig. 8 Relational object graph (*left*) and room graph (*right*)

can thus be easily extended to arbitrarily curved objects if described by a B-rep model.

5.1 Collapsing to radial-edge data structure

The faces of all the solid entities contained in the set of connection objects M_B are copied onto the set of faces M_F, i.e. the volume model is degenerated into a surface model. Subsequently, these faces' sense of orientation, i.e. the direction of their normal vectors, is reversed, as we intend to find the boundaries of indoor air volume bodies rather than defining rigid bodies at this point. The topology of a B-rep model assumes by definition that face normals point to the exterior side of a solid body. Figure 9 illustrates this procedure with four wall objects.

The B-rep model is collapsed into a radial-edge data structure [17, 21]. This is done for two reasons. First, for the sake of model consistency, there should only be one single instance of each vertex, edge and face at each location. Vertices are conflated in an ε-environment to smooth inaccuracies due to round-off errors. Second, the radial-edge structure makes it possible to arrange topological coedge elements with respect to the geometrical alignment of their respective faces.

5.2 Further decomposition

Plane connections between connection objects are removed, because these faces obviously do not contribute to the value set of air volume bodies or hulls (see Fig. 10). These connections are defined by the relation R_{PC}.

Coinciding edges are detected and further decomposed. For example, this case may occur in situations where objects that are interrupted by an opening are connected by means of a continuous floor surface, as shown in Fig. 11.

5.3 Topological analysis

The key issue of the following analysis is to define the relation R_{NF} that strongly depends on the *topological* and *geometrical* configuration of the face elements in space. We follow the principles of the above given definition of rigid bodies. Another restriction is that we aim to find connected surfaces enveloping bodies with smallest volumes in each case.

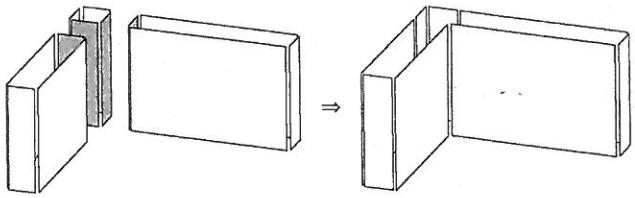


Fig. 10 Removal of plane connections denoted by R_{PC}

Essentially, we formulate two criteria, a topological and a geometrical one. Concatenating these criteria yields a symmetric, anti-reflexive and unweighted graph of room faces G_F. In the sense of graph theory we extract connected components [13] of this graph.

For example, Fig. 12 shows on the right-hand side the graph of room faces of an air volume body with an enclosed pillar. A surface in the model on the left-hand side in Fig. 12 corresponds to a node in the graph. Links between nodes in the graph describe adjacent surfaces that are connected by an edge (interface of type *ve*, cf. Sect. 3). The top face marked is connected to eight other surfaces: the four edges of the outermost polygon are in contact with the 'outside walls', while the four edges defining the hole are connected to the faces of the penetrating pillar.

Consider a relation $R_{F,a} \subseteq M_F \times M_F$ which denotes connections between all faces of M_F with common edges in the sense of Fig. 12. It is evident that an indoor air volume can be represented by a connection component, while a connection component does not necessarily represent an air volume object.

The *topological criterion* for a connection component describing a closed air volume body thus requires that

- each edge of each face has exactly one connection to another face where
- leaf vertices are not permitted and that

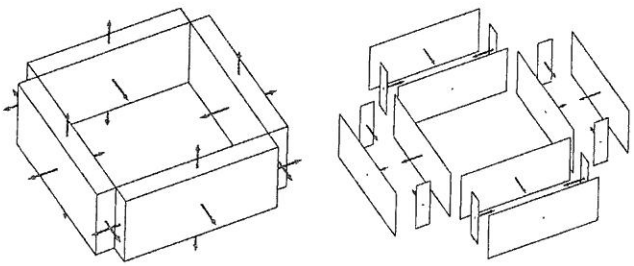


Fig. 9 Inversion of the faces' sense of orientation

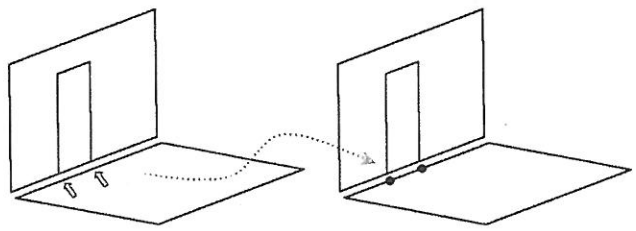


Fig. 11 Decomposition of coinciding curves

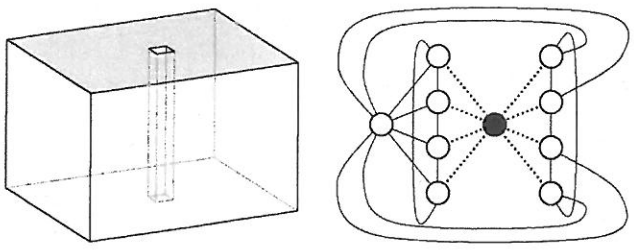


Fig. 12 Graph of room faces of indoor air volume with enclosed pillar

- components are cyclically connected,
- sub-graphs are blocks and free of articulation vertices, i.e. vertices that each exist as the only connection between two different nodes.

As mentioned above, the rule of Möbius defines the sense of orientation of coedges that are reversed in this case. With respect to relation $R_{F,a}$ this implies the extension $R_{F,o} \subseteq M_F \times M_F$ defining an appropriate correlation of the orientational sense. We write

$$R_{F,o} := \{(f_i, f_j) \in M_F \times M_F \mid \exists \mathbf{c}_a \in_t f_i, \exists \mathbf{c}_b \in_t f_j : (\mathbf{e}(\mathbf{c}_a) = \mathbf{e}(\mathbf{c}_b) \wedge \mathbf{c}_a = -\mathbf{c}_b)\}, \tag{12}$$

i.e. there is a regular pair (f_i, f_j) with each element that has a coedge $\mathbf{c}$ with the same underlying edge $\mathbf{e}$ and a reversed sense of orientation.

5.4 Geometrical analysis

In order to obtain sets of connected surfaces contained in the model enveloping bodies with smallest volumes, a *geometrical criterion* shall ensure to get immediately next neighboring faces spanning the smallest possible volume where more than two faces are connected by a common edge (see also Sect. 2.2 for the definition of rigid bodies). In addition, the face orientations must match each other in order to describe valid B-rep bodies, with all face normals pointing to the outside, for example.

The radial-edge data structure accordingly makes it possible to organize sets of coedges. A set of partner coedges (cf. Sect. 5.1) contains a list of coedges that each belong to the faces connected to the same underlying edge. If more than two faces are connected to an edge, it is imperative to choose the face yielding the enveloping surface of the smallest respective volume.

Coedges are grouped in a mathematically negative sense around their edge $\mathbf{e}$. We define a so-called radial-edge vector $\mathbf{r}_i$ as a cartesian product of a coedge vector $\mathbf{c}_i$ and the normal vector of the underlying loop $\mathbf{n}_i$. The vector $\mathbf{r}_i$ points into its corresponding face; in the case of holes, the orientation is reversed.

$$\mathbf{r}_i := \mathbf{n}_i \times \mathbf{c}_i \tag{13}$$

The problem is thus reduced by one dimension. Using this radial-edge vector and according to Figs. 13 and 14, angles between neighboring faces with respect to the edge $\mathbf{e}$ can be readily computed and coedges arranged. For example, Fig. 14 shows an edge $\mathbf{e}$ with three connected faces. The sorted list of coedges results in the set $\{ \mathbf{c}_{1,3}, \mathbf{c}_{3,4}, \mathbf{c}_{2,4} \}$.

The latter criterion can be expressed using the weighted relation $R_{F,w} \subseteq M_F \times M_F$, which evaluates the spatial arrangement of coedges with respect to an underlying edge. This relation is anti-reflexive, i.e. free of loops. The undirected graph becomes a directed graph. If we define the angle $\varphi = \angle_{\mathbf{a}}(,c)$ with $0 \le \varphi \le 2\pi$ as the angle between vectors $\mathbf{b}$ and $\mathbf{c}$ in a plane normal to vector $\mathbf{a}$, if vector $\mathbf{b}$ is rotated in a mathematically positive sense into vector $\mathbf{c}$, we obtain

$$R_{F,w} := \{(f_i, f_j) \in M_F \times M_F \mid (w_{i,j} = \min)\} \tag{14}$$

where the weighting set of elements $w_{i,j} \in W_{F,w}$ is defined by

$$W_{F,w} := \{w_{i,j} = \angle_{\mathbf{c}_a}(a, \mathbf{r}_b) \mid \exists \mathbf{c}_a \in_t f_i, \exists \mathbf{c}_b \in_t f_j : (\mathbf{e}(\mathbf{c}_a) = \mathbf{e}(\mathbf{c}_b))\}. \tag{15}$$

Let us take another look at the example sketched in Figs. 13 and 14 where the three faces f_1, f_2 and f_3—denoted by their normal vectors n_1, n_2 and n_3—are aligned around their common edge. Relation $R_{F,o}$ describes the property of same orientations with regard to adjacent

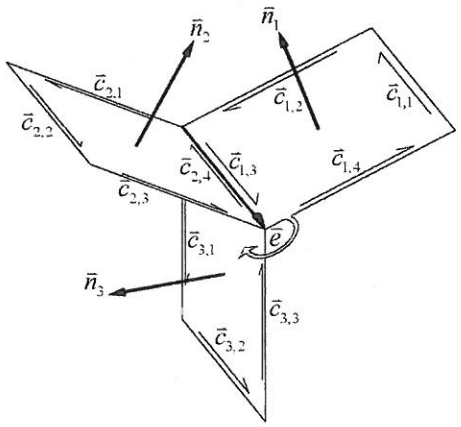


Fig. 13 Analysis of spatial arrangement of coedges $\mathbf{c}_{\alpha,\beta}$ with respect to same underlying edge $\mathbf{e}$. In the example, the first coedge index α refers to the surface number while the second one indexes the individual coedges of each surface. Coedge $\mathbf{c}_{3,4}$ is hidden by surface 2

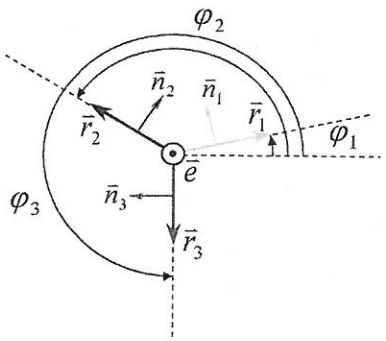


Fig. 14 Angular dependencies according to Fig. 13

faces. Joining faces f_2 and f_3 does not allow us to create a part of the surface of a valid volume body. Relation $R_{F,w}$ evaluates the geometrical alignment with respect to the angular dependencies. Assuming $\varphi_1 = 30°$, $\varphi_2 = 150°$ and $\varphi_3 = 260°$, we get $\delta\varphi_{1,2} = 120°$ and $\delta\varphi_{2,3} = 110°$. Disregarding the topological criterion, the next neighboring face for f_3 results in f_2 and not f_1, because $110° < 230°$ (cf. Fig. 15, center).

Concatenating relations $R_{F,o}$ and $R_{F,w}$ finally yields the relation $R_{NF} \subseteq M_F \times M_F$. Using R_{NF}, we obtain a symmetric, anti-reflexive and unweighted graph of room faces G_F.

$$R_{NF} := R_{F,o} \circ R_{F,w} \subseteq M_F \times M_F \,. \tag{16}$$

5.5 Condensation to room graph

Finally, the graph G_F can be partitioned into n basic equivalence classes $[\lambda_i]$ with $i = 1,...,n$ using the equivalence relation Z_{NF}. The latter is obtained by evaluating the reflexive-transitive hull of relation $R_{NF} \subseteq M_F \times M_F$. The stability index s is accordingly the smallest exponent unless the result of the union is affected by additional terms R^m with $m > s$.

$$\begin{aligned} Z_{NF} := R_{NF}^{\star} &= I \sqcup R_{NF} \sqcup R_{NF}^2 \sqcup \cdots \sqcup R_{NF}^s \\ &\subseteq M_F \times M_F \end{aligned} \tag{17}$$

Hence, the set M_F can be condensed to the quotient set M_{AV} using the mapping Φ_{AV}. We obtain a reduced graph $G_{\mathrm{AV}} = (M_{\mathrm{AV}}; R_R)$ which we denote as room graph:

$$\Phi_{\mathrm{AV}} : M_F \longrightarrow M_{\mathrm{AV}} \quad \text{with} \quad M_{\mathrm{AV}} = \frac{M_F}{Z_{NF}} \tag{18}$$

$$R_R = \Phi_{\mathrm{AV}}^{\mathrm{T}} R_{NF} \Phi_{\mathrm{AV}} \tag{19}$$

A closed B-rep shell is accordingly obtained if, and only if, a sub graph determined by a representative λ_i denotes a basic cyclic connected component, i.e. the sub graph contains cycle and separation edges only.

For example, Fig. 16 shows on the left-hand side the part of a connection model of two rooms linked by an open door. Accordingly, the graph analysis yields

- the model hull (top of right-hand side of Fig. 16) and
- the corresponding air volume body—a single one in this case (bottom of right-hand side of Fig. 16).

The body specified by the hull face therefore corresponds to the outer domain, as all face normals point to the interior.

If we use the Gauss integral theorem to compute volumes of bodies $V([\lambda_i])$, we can add another criterion for identifying these situations. By replacing the volume integral in the numerical quadrature with a face integral, we obtain negative but finite values as indicators.

Note that this technique also makes it possible to detect modeling inaccuracies, such as non-closed shells. These patches can be highlighted by a wireframe model indicating (and in particular locating) the cause of an incorrect configuration.

We therefore separate the set M_{AV} into a set of 'valid' indoor air volume bodies $M_{\mathrm{AV,valid}}$, a set of 'hull objects' $M_{\mathrm{AV,hull}}$ and a set of non-closed B-rep bodies $M_{\mathrm{AV,incomplete}}$. In cases where the number of elements of $M_{\mathrm{AV,incomplete}}$ is greater than zero, we are able to identify modeling inaccuracies.

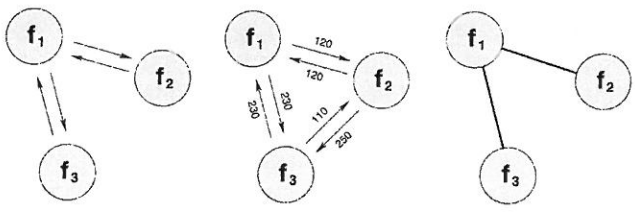


Fig. 15 Relations $R_{F,o}$ (*left*), $R_{F,w}$ (*center*) and R_{NF} (*right*)

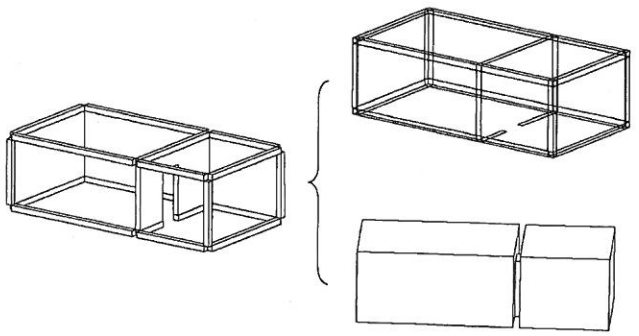


Fig. 16 Air volume extraction of model with two rooms connected by open door (front walls of connection model plotted transparently on the *left-hand side*)

$$M_{\mathrm{AV}} = M_{\mathrm{AV,valid}} \cup M_{\mathrm{AV,hull}} \mathrm{AV, incomplete} \quad (20)$$

Assigning elements to their corresponding equivalence classes is achieved by forming the reflexive-transitive hull of the adjacency relation for cyclic edges $R^*_{NF,\mathbf{Z}}$:

$$\begin{aligned}
&[\lambda_i] \in M_{\mathrm{AV,valid}} \\
&\qquad \Longleftrightarrow \quad \lambda_i \lambda_i^{\mathrm{T}} \sqsubseteq R^{\star}_{NF,} \quad \wedge \quad V([\lambda_i]) > 0 \\
&[\lambda_i] \in M_{\mathrm{AV,hull}} \\
&\qquad \Longleftrightarrow \quad \lambda_i \lambda_i^{\mathrm{T}} \sqsubseteq R^{\star}_{NF,\mathbf{Z}} \quad \wedge \quad V([\lambda_i]) < 0 \\
&[\lambda_i] \in M_{\mathrm{AV,incomplete}} \\
&\qquad \Longleftrightarrow \quad \text{else}
\end{aligned} \quad (21)$$

In this way, a closed B-rep shell is determined by a basic cyclic connected graph, provided the aforementioned topological and geometrical criteria are fulfilled with respect to the definition of rigid bodies: surfaces can be identified as being closed, faces have an orientation and faces are not allowed to intersect one another.

5.6 Summary: algorithm

We can accordingly summarize the sequence of operations in Algorithm 2.

5.7 Implementation strategies

As matrix operations involve a considerable computational investment, our algorithmic implementation makes use of an object-oriented data structure of the geometrical model and is implemented using a depth-first search tree in C++. We refer to [17] for all the mathematical derivation and implementational aspects.

Algorithm 2 Derivation of the graph of room faces G_F and the room graph G_{AV}

1: Copy faces f_i of all bodies $\in M_B$ to M_F
2: **for all** faces $f_i \in M_F$ **do**
3: Delete f_i, if part of plane connections denoted by G_B
4: Invert orientational sense of f_i: $\mathbf{n}_i = -\mathbf{n}_i$
5: Collapse f_i into *radial-edge* data structure
6: Compute $\mathbf{n}_j$ of all loops $l_j \in_t f_i$, set *flag* = *valid* if $\mathbf{n}_i = \mathbf{n}_j$ or *flag* = *hole* else
7: **end for**
8: Decompose coinciding edges of all faces $f_i \in M_F$ and connected coedges
9: Create relation $R_{F,o}$ according to (12)
10: Compute *radial-edge* vectors $\mathbf{r}_i$ of all edges
11: Create relation $R_{F,w}$ according to (14) with weighting set $W_{F,w}$ according to (15)
12: Concatenate relations $R_{NF} = R_{F,o} \circ R_{F,w}$
13: ⇒ **Compute** $G_F = (M_F; R_{NF})$
14: Create equivalence relation $Z_{NF} = R_{NF}^*$
15: Do decomposition: $[\lambda] = Z_{NF}\, \lambda$
16: Compute mapping $\Phi_{\mathrm{AV}} : M_F \longrightarrow M_{\mathrm{AV}}$
17: Compute $R_R = \Phi_{\mathrm{AV}}{}^{\mathrm{T}} R_{NF} \Phi_{\mathrm{AV}}$
18: ⇒ **Compute** $G_{\mathrm{AV}} = (M_{\mathrm{AV}}; R_R)$
19: Assign bodies $[\lambda_i]$ according to (21) to sets M_{AV}

6 Semantical identification

We obtain the relational object graph G_I (cf. Fig. 8) with its incidence matrix $R_I \subseteq M_B \times M_{\mathrm{AV}}$ by correlating structural components with the set of air volumes and the hull of the model. R_I is a heterogeneous binary relation formed by regular pairs $(c_i, [\lambda_j])$ of components $c_i \subset M_B$ and air volume bodies $[\lambda_j] \subseteq M_{\mathrm{AV}}$. Assuming plane faces f_a and f_b there is a regular pair $(c_i, [\lambda_j])$ if, and only if, both objects are in plane contact, i.e. both normal vectors $\mathbf{n}$ are antiparallel and the sets of edges $\mathbf{e}$ are equivalent. In our software, objects are identified by assigning unique unified identifiers to them.

$$\begin{aligned}
R_I := \{ &(c_i, [\lambda_j]) \in M_B \times M_{\mathrm{AV}} \mid \\
&\exists f_a \in_t c_i, \exists f_b \in_t [\lambda_j] : \\
&(\mathbf{n}_a = -\mathbf{n}_b \wedge \\
&\quad \forall \mathbf{e} \in_t f_a : \mathbf{e} \in_t f_b \wedge \forall \mathbf{e} \in_t f_b : \mathbf{e} \in_t f_a) \}
\end{aligned} \quad (22)$$

With the help of the relational object graph, analyzing components of the connection model with respect to their semantics and storing this information in the model, using body and face attributes, is fairly straightforward. Structural components, i.e. volume objects, such as walls, slabs or plates can be identified as *external, outside, internal, interzonal* elements (or

invalid if adjacent to a non-closed air volume body). An example will be given in the next section.

In the following section, we consider components of the difference model $M_D \subseteq M_B$ only. In order to set up boundary conditions in simulations, interfaces between the air and components are characterized by attributing the objects' faces accordingly. Face attributes are {*ambient*, *airvolume*, *none*, *invalid*} depending on neighboring entities respectively. Given $R_I \subseteq M_B \times M_{\mathrm{AV}}$ we obtain Algorithm 3 for the component identification [17].

7 Sample application

As opposed to the manual grouping of objects and assignment of functions to the respective components, a knowledge of the derived graphs establishes the basis for a more in-depth analysis of the model. This will be demonstrated using the sample model given in Fig. 17, which takes three storeys with integrated inner courtyard into account.

Algorithm 3 Identification of components $\subset M_D$

1: **for all** $[\lambda_i] \in M_{\mathrm{AV}}$ **do**
2: **for all** $d_j \subset M_D$ with $d_j \lambda_i^{\mathrm{T}} \sqsubset R_I$ **do**
3: Determine $\{f_b \mid f_a \in_t [\lambda_i] \wedge f_b \in_t d_j \wedge \mathbf{n}_a = -\mathbf{n}_b\}$
4: **if** $\lambda_i \subset M_{\mathrm{AV,valid}}$ **then**
5: Set face attribute $a(f_b)$ of d_j to *airvolume*
6: **else if** $\lambda_i \subset M_{\mathrm{AV,incomplete}}$ **then**
7: Set face attribute $a(f_b)$ of d_j to *invalid*
8: **end if**
9: **end for**
10: **end for**
11: **for all** $d_i \subset M_D$ **do**
12: Obtain surfaces $f_a, f_b \in_t d_i$
13: **if** $a(f_a) = \mathit{invalid} \vee a(f_b) = \mathit{invalid}$ **then**
14: Set volume attribute $v(d_i) = \mathit{invalid}$
15: **else if** $a(f_a) = \mathit{airvolume} \wedge a(f_b) = \emptyset$ **then**
16: Set face attribute $a(f_b) = \mathit{ambient}$
17: Set volume attribute $v(d_i) = \mathit{outside}$
18: **else if** $a(f_a) = \emptyset \wedge a(f_b) = \mathit{airvolume}$ **then**
19: Set face attribute $a(f_a) = \mathit{ambient}$
20: Set volume attribute $v(d_i) = \mathit{outside}$
21: **else if** $a(f_a) = \mathit{airvolume} \wedge a(f_b) = \mathit{airvolume}$ **then**
22: **if** $| d_i^{\mathrm{T}} R_I | = 1$ **then**
23: Set volume attribute $v(d_i) = \mathit{interior}$
24: **else**
25: Set volume attribute $v(d_i) = \mathit{interzonal}$
26: **end if**
27: **else if** $a(f_a) = \emptyset \wedge a(f_b) = \emptyset$ **then**
28: Set face attribute $a(f_a) = a(f_b) = \mathit{none}$
29: **if** $M_{\mathrm{AV,valid}} \cap d_i = d_i$ **then**
30: Set volume attribute $v(d_i) = \mathit{interior}$
31: **else**
32: Set volume attribute $v(d_i) = \mathit{exterior}$
33: **end if**
34: **end if**
35: **end for**

7.1 Model decomposition

Figure 18 shows parts of the decomposed connection model and Fig. 19 displays the set of air volumes obtained.

It is possible to aggregate air volume bodies themselves in order to form contiguous areas. Intuitively, handling air volume bodies given by the building design and current configuration is the most common way to define different zones. A CAD user *visually understands* the building structure rather than just 'storing' self-defined lists of walls and objects in a dialog box.

7.2 Object model

By applying Algorithm 3, it is possible to identify components M_D semantically, as shown in Fig. 20. Using the relational object graph G_I, reducing the complexity of the model becomes fairly straightforward by deriving an object-oriented model, as depicted

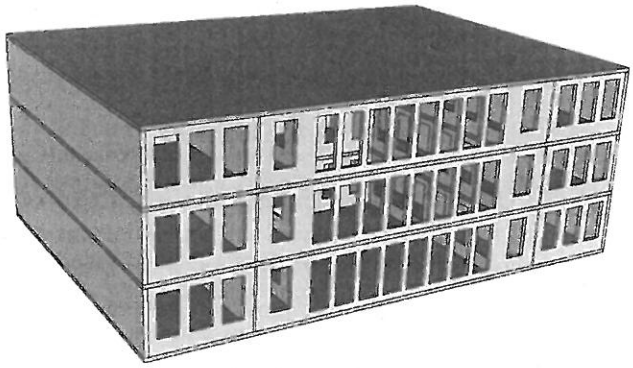


Fig. 17 Sample model with inner courtyard

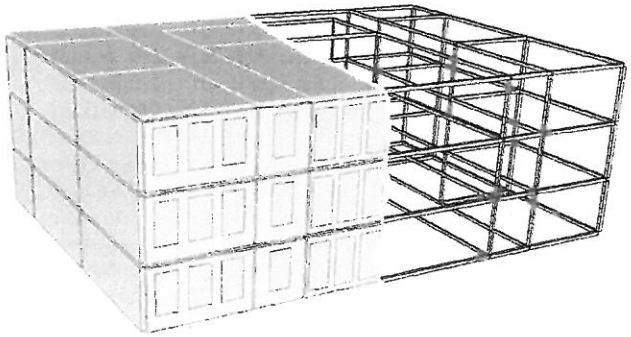


Fig. 18 Decomposed model with difference objects (*left*) and coupling objects (*right*)

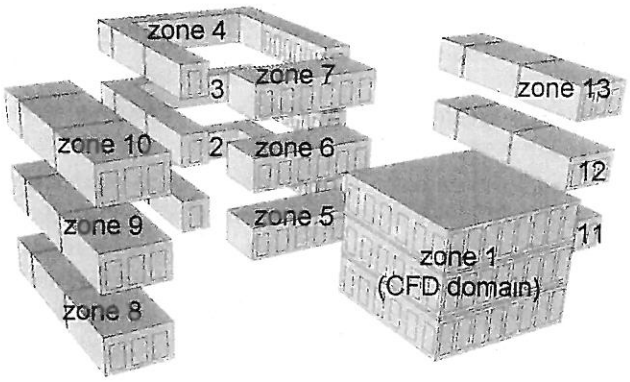


Fig. 19 Extracted air volumes and aggregation to zones, e.g. zone 8 is composed of three single air volume objects

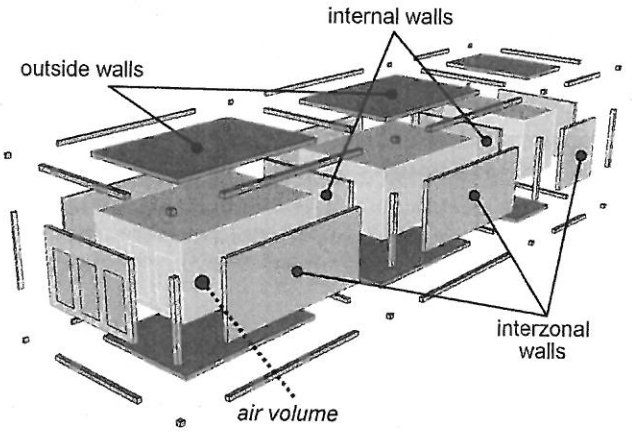


Fig. 20 Semantical identification of components

in Fig. 21. Buildings are hierarchically structured systems and are usually organized in storeys, rooms and building components. We dispense with this view in favor of recognizing zones as aggregations of air volume bodies with associated components.

In this case, the layout of the object model was chosen according to the thermal multizone building model proposed by Nytsch-Geusen [11]. Note that this technique is generally applicable to many building energy simulation tools, as the software technique makes use of an XML based, object-oriented data structure based on the document object model provided by the QT library [15].

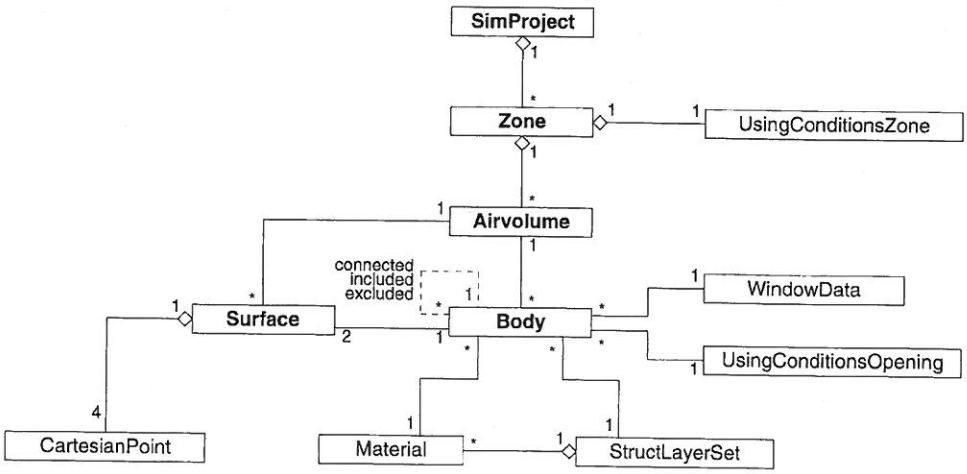
A simulation project aggregates a number of zones, where the latter aggregate one or more air volumes. Air volume objects are aware of the corresponding set of adjacent bodies and their semantics. Structural elements themselves are composed of a multilayered structure with respective individual materials. Although they form part of the geometric model, we also store the surface geometry and vertex coordinates. This makes it possible to directly transfer the object model only, without the need to exchange the B-rep model for establishing a multizone model and linking it to one or more CFD domains.

7.3 Linking to a CFD domain

As indicated in Sect. 1.2, and limited by the computing facilities available, a detailed CFD computation is restricted to individual zones, such as an inner courtyard, and within defined time intervals only. Figure 22 shows the formation of air volumes and the facetted surface mesh of the CFD domain, i.e. the atrium with staircase and occupants inside. As the thermal building model provides CFD with appropriate boundary conditions (and vice versa, as the case may be), both models are linked using the relational object graph.

To be more precise, the aim is to link the surface of the object's layer alongside the flow model to the computational domain, as shown in Fig. 23.

Fig. 21 Multizone object model



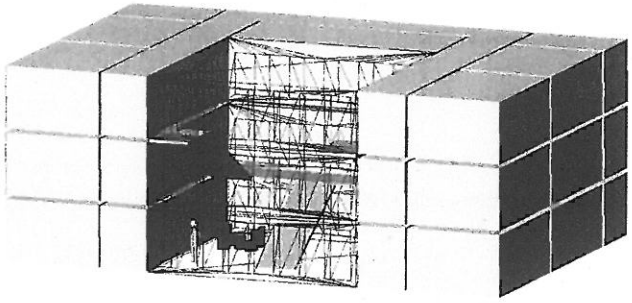


Fig. 22 Facetted surface mesh of CFD domain

8 Conclusion and outlook

In this paper, we have discussed algorithms for analyzing and interpreting building (product) model-based geometry in order to support the linking of computer-aided design tools with numerical simulation techniques. The focus was on the key topological and geometrical aspects. We have presented a technique based on graph theory that makes it possible to derive both a dimensionally reduced object model, required for setting up a thermal multizone model, and a geometrical model for defining single or multiple CFD domains in a building model together with incidence matrices correlating these models. The incidence matrices are an essential precondition for numerically coupling both approaches, for example by automatically providing a CFD model with boundary conditions obtained during a thermal multizone simulation and vice versa. We start from a CAD or a building product model. The algorithms presented here are basically applicable to any building energy simulation tool.

As mentioned in the first section, the application of detailed simulation tools is often avoided in civil engineering practise because of the costs incurred by model definition and the cumbersome process of sharing and exchanging data between applications. We hope that this integrated technique will facilitate this situation and we also anticipate that three-dimensional modeling techniques will substitute the still widespread draft-oriented, two-dimensional modeling approach in the near future.

Our research group is currently concentrating on the formal definition of a spatial query language for BIMs. It provides formal definitions using point-set theory and point-set topology for 3D spatial data types and for directional, topological, metric and boolean operators operating on these types. An object-relational database management system will be adopted for the implementation of 3D spatial query processing. The algorithms presented in this paper serve as a basis for model recognition and interpretation, as required for spatial queries.

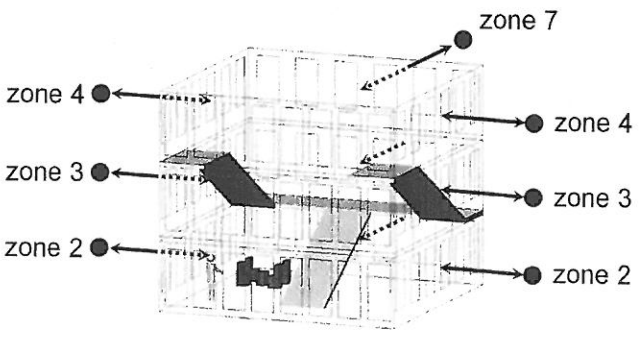


Fig. 23 Links between dimensionally reduced object model and geometrical model

Acknowledgments The authors' acknowledgements are due to Richard Romberg and André Borrmann for many valuable discussions. This work is being sponsored by grant No. AZ 468/01 awarded by the *Bayerische Forschungsstiftung* [1]. The results presented in this paper form part of the work being undertaken within the research project *SIMFAS*, which aims at coupling thermal building energy simulation with CFD methods [17].

References

1. http://www.forschungsstiftung.de
2. Beausoleil-Morrison I (2000) The adaptive coupling of heat and air flow modelling within dynamic whole building simulation. PhD Thesis, University of Strathclyde, Glasgow
3. Bungartz H-J, Griebel M, Zenger Ch (2004) Introduction to computer graphics, 2nd edn. Charles-River Media, Boston
4. Clarke JA (2001) Energy simulation in building design, 2nd edn. Butterworth-Heinemann, Oxford
5. Corney J, Lim T (2001) 3D modeling with ACIS. Saxe-Coburg Publications, Stirling
6. Drexl T (2003) Entwicklung intelligenter Pfadsuchsysteme für Architekturmodelle am Beispiel eines Kiosksystems. Diploma Thesis, Institut für Informatik, Technische Universität München
7. Egenhofer M, Franzosa R (1991) Pointset topological spatial relations. Int J Geogr Inf Syst 5(2):161–174
8. Eurostep Group AB (2000) The IFC STEP toolbox, version 2.X, Beschreibung der IFC Toolbox Classic. http://www.eurostep.com
9. Hensen JLM (1999) A comparison of coupled and de-coupled solutions for temperature and air flow in a building. ASHRAE Trans 105(2):962–969
10. International Allicance for Interoperability (IAI). http://www.iai-international.org
11. Nytsch-Geusen C, Bartsch G (2001) An object-oriented multizone thermal building model based on the simulation environment SMILE. In: Proceedings of the 7th IBPSA conference on building simulation, Rio de Janeiro, Brazil
12. O'Rourke J (1998) Computational geometry in C, 2nd edn. Cambridge University Press, London
13. Pahl PJ, Damrath R (2001) Mathematical foundations of computational engineering. Springer, Berlin Heidelberg New York

14. Petersen M, Meißner U (2000) Energieoptimierte Gebäudeplanung mit verteilter Informationsmodellierung. In: Proceedings of the IKM, Bauhaus-Universität Weimar, Weimar
15. QT library online reference (2005) http://www.trolltech.com
16. Romberg R, Niggl A, van Treeck C, Rank E (2004) Structural analysis based on the product model standard IFC. In: ICCCBE, Xth international conference on computing in civil and building engineering, Weimar
17. van Treeck C (2004) Gebaeudemodell-basierte simulation von Raumluftstroemungen. PhD Thesis, Munich Technical University
18. van Treeck C, Rank E (2004) Analysis of building structure and topology based on graph theory. In: ICCCBE, Xth international conference on computing in civil and building engineering, Weimar
19. van Treeck C, Rank E, Krafczyk M, Tölke J, Nachtwey B (2006) Extension of a hybrid thermal LBE scheme for Large-Eddy simulations of turbulent convective flows. Comput Fluids 35:863–871
20. van Treeck C, Rank E, Schrag T, Katz C (2005) Abschlussbericht zum Forschungsprojekt SIMFAS. Funding: Bayerische Forschungsstiftung, AZ:468/01, Lehrstuhl fuer Bauinformatik, Technische Universitaet Muenchen
21. Weiler K (1988) The radial-edge structure: a topological representation for non-manifold geometric boundary modeling. In: Wozny M, McLaughlin H, Encarnacao J (eds) Geometric Modeling for CAD Applications. Elsevier, Amsterdam, North-Holland, Amsterdam, pp 3–36
22. Wenisch P, van Treeck C, Borrmann A, Rank E, Wenisch O (2006) Computational steering on distributed systems: indoor comfort simulations as a case study of interactive CFD on supercomputers. Int J Parallel Emergent Distrib Syst (in press)