SIMULATION BASED ON THE PRODUCT MODEL STANDARD IFC

Christoph van Treeck, Richard Romberg, and Ernst Rank
Lehrstuhl für Bauinformatik, Technische Universität München
Arcisstr. 21, D-80290 München, Germany, http://www.inf.bv.tum.de
{treeck, romberg, rank}@bv.tum.de

ABSTRACT
We consider a strictly three-dimensional modeling technique as a basis for numerical simulations. Applications range from
• a high-resolution indoor air flow modeling using the so-called thermal lattice Boltzmann method,
• combined with a multizone network model for the building energy simulation,
• to the scope of structural engineering, i.e. a three-dimensional high order solid finite element analysis of a building.

Starting from a building product model described by the Industry Foundation Classes (IFC) standard, we derive an intermediate geometric B-rep model based on the ACIS geometric kernel (Spatial Corp.). This ‘consistent’ (e.g. free from gaps) and ‘corrected’ (e.g. due to intersections) geometric model is decomposed into a so-called connection model. We describe the analysis and discretization of this strictly volume-oriented model, which serves as the basis for all numerical simulations and thus enables automatic mesh generation with respect to different numerical schemes. A database manages additional attributes and the relation between IFC and ACIS objects. The last section sketches how structural simulation and thermal building simulation can be embedded into a common simulation environment.

INTRODUCTION
The present work evolved from two current research projects with different objectives but the task to generally base applications on a common geometric model. With this modus operandi we focus on one of the various problems associated with co-operation and communication between engineers. Given that each discipline owns a different view and interpretation of its specific product part, it is difficult to define a common basic model for interoperability.

The use of a thermal building simulation program, for example, requires the configuration of geometrical and physical properties in a numerical model. Gathering this information is expensive and error-prone, because appropriate data usually cannot be imported directly from the previous work of participating partners. The same situation arises for a structural engineer, furthermore geometric information usually has to be dimensionally reduced or broken down into a number of mechanical sub-models (beam, slab, shell, ...) for structural analysis.

The lack of a common model for a building can be identified as a main difficulty in the collaboration of different disciplines. A promising basis for a solution is offered by the International Alliance for Interoperability (IAI), providing its Industry Foundation Classes (IFC) as an object oriented description of building product model data to ensure software interoperability in the building industry (IAI, 2003). This standard data exchange format serves as a basis for our common geometrical model and, hence, our numerical simulations. The following two subsections briefly sketch the goals of both underlying projects.

Thermal building simulation
Subject of one research project is the coupling of dynamic building energy simulation with CFD methods for the analysis of energy balance and thermal comfort in buildings based on the product modeling standard IFC (van Treeck, 2001 and Rank, 2003). We currently develop a complementary method to numerically simulate fluid flows and convective heat transfer using the so-called hybrid thermal lattice Boltzmann method (HTLBE) according to the work of (Lallemand and Luo, 2002) and (d’Humières, 2002). In a later stage of the project we will extend an existing large-eddy turbulence model (Krafczyk, 2001) by appropriate wall functions in order to accurately compute buoyant boundary-layer flows. For the thermal building simulation, we use the simulation environment SMILE (Nytsc-Geusen, 2001). A coupling and iterative solution process will be achieved by a loose-coupling algorithm between both numerical schemes, the multizone model and our 3D HTLBE CFD code. Results will be presented in forthcoming publications.

Structural simulation
The counterpart of the above mentioned project aims at solid modeling as a basis of co-operative planning
in structural engineering (Rank, 2002). In this part of the simulation process we also start from a three dimensional IFC building model, which is firstly decomposed into a so-called ‘connection model’ and further transferred into a numerical analysis model, i.e. a finite element mesh.

The structural simulation is performed in a strictly three-dimensional manner. The \textit{p-version of the finite element method} is used, which, in contrast to the \textit{h-version}, fixes the size of elements and increases the polynomial degree of the displacement field instead. The polynomial degree of each component of the field can be chosen individually and may also be varied in all three local directions. This anisotropic Ansatz allows an efficient computation of three-dimensional plate- and shell-like structures (Düster, 2002).

\textbf{A sample building}

This section shows preliminary studies using a sample building, consisting of walls, doors, windows, openings and slabs. Considering structural engineering, the example on the left hand side of figure 2 shows a plot of von Mises stresses together with displacements caused by vertical loads applied to the walls. The CFD model on the right hand side was used in order to analyze temperature stratification and potential draft risk within the atrium during a hot summer day. We implemented an unidirectional coupling interface between the SMILE multizone model and the CFD package CFX-5 (van Treeck, 2003). The picture on the right shows streamlines of turbulent flow injected at an opening.

\textbf{IFC BASED SIMULATION}

The IAI was founded in order to provide “a basis for process improvement and information sharing in the construction and facilities management (AEC/FM) industries” (IAI, 2003). All effort is reflected in a multivendor capable standard, the Industry Foundation Classes (IFC). The goal of this product model standard is to define an integral, object-oriented and semantical model of all components, attributes, properties and relationships of and within a ‘building product’ and to gather information about its originating process, life cycle and disposal. To account for global relationships, in many countries so-called ‘chapters’ have been established which work on several projects by extending the IFC object data model, e.g. BS-8 (HVAC extension), BS-7 (building performance monitoring) and ST-4 (load definition and rating). Meanwhile, the IAI consists of nine chapters with over 650 members worldwide. The IFC product model is specified using the modeling language EXPRESS, which has been used to define STEP based product models within ISO 10303 before. Since 2002, the current release IFC 2.x is certified as ISO/PAS 16739 standard. For further information refer to (IAI, 2003) and the references therein.

While the main focus was placed within release 1.51 on data concerning a building’s shell, IFC 2.x has been extended considerably. In addition to the architecture domain it now contains different domains like facility management, HVAC, construction management, etc. This complexity affects the number of classes which are arranged in a hierarchy consisting of five layers, i.e. a layer is described by another layer in more detail due to specialization and multiple inheritance between classes.

For example, all geometric entities (e.g. wall, column, slab) are collected by the class \texttt{IfcProduct}. These objects share properties as location \texttt{IfcLocalPlacement} or representation \texttt{IfcProductRepresentation}. The class \texttt{IfcProduct} can be divided into a number of subclasses up to an ultimate depth level, arranged in the \texttt{Resource Layer}, which contains abstract supertypes and thus is the basis for all specializations. Examples are a direction vector, \texttt{IfcDirection}, or the coordinates of a cartesian point, \texttt{IfcCartesianPoint}. All sub-classes define corresponding attributes and relationships. Clearly, this structure allows for the management of AEC/FM related data. Compared to the exchange of elementary drawing data between CAD systems using the DXF standard, for example an \texttt{IfcWall} object is recognized as a wall by different platforms rather than being identified as an accumulation of lines and faces only. In order to hide some complexity, the IAI defines so-called exchange views, which enable handling of application specific data (Lämmle, 2003).

Currently a number of commercial applications provide IFC interfaces, usually for IFC 1.51, but within the
‘second generation’ some of them already have been certified for version 2.x (ISG, 2003). In most cases, tools are limited to geometry only and construction material characteristics are not yet supported.

Because even the exchange of geometry representing objects and the definition of zones and corresponding spaces is a non-trivial task, we focus on this specific topic here. Since the IFC standard is not yet well established, more experience in exchanging data and more IFC based simulation applications are required to enable interoperability among tools and to allow this relatively new technology to mature.

IFC toolbox

In order to comfortably operate on physical IFC data and to reduce the amount of parsing these data, we use the Eurostep IFC toolbox (Eurostep, 2003). The toolbox maps EXPRESS types defined in the IFC scheme to an inheritable C++ class structure and provides an integrative interface to access and manage IFC instances. An entity corresponds to a C++ class and is composed of types, functions and aggregations (bag, list, set and array) defined within the toolbox and other entities. The library also adds some basic functionality to read and write IFC files.

Simulation framework

All numerical simulations as described above are based on the same, strictly three-dimensional volume model (Rank, 2002). This allows for developing algorithms for automatic mesh generation with respect to different numerical schemes.

Model transfer

Before decomposition, we transfer geometric information contained within a physical IFC building product model definition to a solid B-rep model. With respect to further transformations due to application dependent discretizations, we are working on the basis of the ACIS geometric kernel by Spatial Corp. (Corney, 2001). As described above, we use the C++ toolbox interface in order to access IFC instances. The following steps have to be processed to extract all geometry related entities contained in an arbitrary IFC data file:

- Analysis of each entities’ product shape, which contains position and geometric representation,
- analysis of voids within objects,
- (multiple) transformations due to relative placements,
- incorporation of material layers and
- appropriate blending of connected walls.

Check for consistency

Within the IFC context, each geometric entity is managed with its individual representation, which can be attribute driven (sweeping), boundary represented (B-rep) or constructive solid (CSG) geometry. Consequently, shared nodes of adjacent entities appear more than once and can differ due to roundoff errors. These inconsistencies are caused by modeling inaccu-

racies or by the computer-internal representation of floating point numbers, if e.g. objects have been transformed. In order to use IFC based geometry for a numerical simulation, these inconsistencies have to be recognized and eliminated before mesh generation. As topology or interconnection relationships are not available for all entities, the effort of checking each element is of second order, i.e. each entity has to be checked against each other with respect to inconsistencies. Due to the large number of geometric elements within a building model, we use an octree based space partitioning scheme for the analysis in order to reduce computing and memory resources considerably (Rank, Zenger et al., 2003).

Internal data structure

The software engineering strategy behind this work is the development of a C++ based, object-oriented, platform independent and modular framework which makes use of a number of libraries, such as QT, ACIS, HOOPS and the Eurostep Toolbox.

In order to retain information contents in the context of IFC when transferring data to the volume oriented model, the ACIS geometric model (SAT) was exten-
ded by an object oriented data structure (DAT). Each ACIS entity gets a unified unique identifier (UUID) as attribute, corresponding to IfcGloballyUniqueId of the IFC object. (The methodology does not obligatory depend on IFC, i.e., if IFC is not the origin, a different UUID is generated.) Each geometric node within the data base now corresponds to an ACIS entity and each entity knows about its former IFC affiliation. So-called structural nodes allow to organize the data base in a broader context using an object oriented scheme. For example, an entity related to an IfcWall object contains more than geometric information only. Thus, our data base holds an abstract wall object being decomposed into a number of layers which in turn consist of material and surface properties together with geometric information, relations to adjacent walls and openings.

![Image](Image)

**Figure 4:** IFC geometry is transferred to the ACIS geometric model (SAT) and, using UUID attributes, linked to a data base (DAT).

Furthermore, the data base is extended with respect to simulation. Depending on the specific simulation task (structural analysis, thermal building simulation, CFD), additional material properties and relevant boundary conditions are stored.

**Product shape and component structure**

All design related properties of building product entities as e.g. IfcWall are represented by IfcBuildingElement and its inherited attributes. Each object owns a unique identifier (IfcGloballyUniqueId), a local positioning (IfcLocalPlacement), a layer structure (IfcMaterialSelect), references to openings which are part of the object (IfcRelVoidsElement) and a product shape describing geometry and position.

In order to extract geometric information, this structure has to be analyzed accordingly. Three-dimensional geometric representations can be defined by a B-rep, an attribute driven or a CSG model. Our code is capable of importing both B-rep and sweeping based IFC geometry including an analysis of material layer sets.

**Coordinate transformation**

During the transfer to the ACIS kernel, up to four transformations per element have to be performed in order to place objects in a global Cartesian coordinate system.

Considering the number of translations and rotations, clearly, geometric representations within IFC are not suitable to serve as geometric base itself. Basic geometric entities (not to mistake geometry for topology) as points, lines and faces occur more than once and inconsistencies result due to roundoff errors caused by multiple transformations. Furthermore, discrepancy persists between geometric representation of objects within IFC and objects being visualized by a CAD system. A system has to interpret data, such as e.g. intersecting walls, and has to visualize them in a modified manner.

Consequently, to overcome this situation, geometry has to be transferred into a B-rep model but without loosing information contents of the object model.

**Wall intersections**

Objects which consist of multiple layers, e.g. IfcWall or IfcRoofSlab, have to be decomposed according to their structure. IfcMaterialSelect assembles layers by providing either a list of materials or a whole structure using a layer set. For each layer as well as for the union of this set we generate ACIS bodies and append the data base by layer entities together with material characteristics. It is mentioned, that we distinguish between load supporting elements, which are extracted prior to decomposition, and elements for the thermal analysis, where informations about the layer structure are obligatory.

![Image](Image)

**Figure 5:** Placement of an opening, cf. (IAI, 2003). Up to four transformation may be necessary in order to obtain an object’s global positioning.

In order to correctly describe interconnections with respect to simulation tasks, we use both a layer by layer blending and an inclined intersection of objects.

**Figure 6:** Wall intersections in CAD systems: left: layer by layer (Nemetschek Allplan), right: inclined (Autodesk ADT), (Lämmle, 2003).
corresponding layers. Layers are extruded and intersected accordingly. For further details it is referred to (Lämmle, 2003).

**MODEL DECOMPOSITION**

The solid B-rep model is not yet adequate for numerical simulation. Concerning structural analysis, a decomposition into objects being better suitable for a finite element meshing is desired. Thermal building simulation needs, on the other hand, a derived model allowing the definition of the interior flow regime. As an intermediate step between the first geometric model and the application dependent partial models, a so-called connection model will now be defined.

The solid model is decomposed into connection bodies at all locations where components coincide. Considering the geometric intersection, besides these so-called coupling objects, which we describe in detail below, objects within the connection model hold collective nodes and edges only. This approach has important advantages:

- Considering the mechanical sub-model, connections of mechanical relevance can be described by connection bodies only. Apart from mechanical connections and geometric objects, this sub-model contains informations about material data and load definitions added by an engineer. The model serves as direct input for an automatic hexahedral mesh generation as described below.
- Within the scope of building energy simulation, continuous elements as continuous slabs are broken down into a number of sub-elements. This allows for a zonal description of individual rooms without further modifications.
- For the zonal and CFD simulation, indoor air volumes can be described using a connected graph of faces of all adjacent elements within a zone. Thus, the indoor air volume is accurately modeled by its surface in a B-rep manner. Furthermore, this face set could be used for computing view factors for radiative heat transfer calculations.

**Connection model definition**

Based on their different semantics, we distinguish between coupling and difference objects during model decomposition.

We consider the consistent geometric model \( \Omega \subseteq \mathbb{R}^3 \) as set of one or more B-rep bodies. Particularly, this is a closed set which we denominate collectivity. This collectivity is decomposed into \( m + n \) objects, namely \( m \) coupling and \( n \) difference objects. Thus, for all \( m + n \) elements \( \omega_i \) with \( i = 1, \ldots, m \) and \( \omega_j \) with \( j = 1, \ldots, n \) we obtain

\[
\bigcup_{i=1}^{m} \omega_i = \Omega^C \subseteq \mathbb{R}^3, \quad (1)
\]

\[
\bigcup_{j=1}^{n} \omega_j = \Omega^D \subseteq \mathbb{R}^3, \quad (2)
\]

where \( \Omega^C \) denotes the set of coupling objects and \( \Omega^D \) the set of difference objects with

\[
\Omega^C \subseteq \Omega, \quad (3)
\]

\[
\Omega^D \subseteq \Omega. \quad (4)
\]

The union of both sub-sets again results in the collectivity \( \Omega \):

\[
\Omega^C \cup \Omega^D = \Omega \subseteq \mathbb{R}^3 \quad (5)
\]

**Intersection types**

Each coupling and difference object \( \omega_i, \omega_j \subseteq \mathbb{R}^3 \) is a closed B-rep body being described by nodes, edges and faces. Assuming a consistent model and depending on adjacencies and relationships, we distinguish between the following types of intersection:

- Type NEF: The intersection of objects with adjacent faces consists of nodes, edges and faces.
- Type NE: The intersection of objects with adjacent edges consists of nodes and edges.
- Type N: The intersection of objects with adjacent nodes consists of nodes only.

Objects can be characterized according to their intersections:

- Intersections between difference objects are of type N or NE.
- Intersections between coupling objects are of type N, NE or NEF.
- Intersections between difference and coupling objects are of type N, NE or NEF.

**Decomposition algorithm**

Using Boolean operations intersection and difference, we obtain a connection model with the above described objects. After decomposition, coupling objects possess hexahedral structure while difference objects can have different topologies. It should be mentioned, that our approach does not aim at meshing general spatial structures into hexahedral elements but is capable of decomposing objects being related to building products, like plates, beams, columns and slabs.

*Figure 7: Connection model with coupling and difference objects.*
We consider the above defined geometric model, the collectivity \( \Omega \subseteq \mathbb{R}^3 \), which consists of a set of objects \( G_k \) with
\[
G_k \subseteq \Omega, \quad k = 1,...,N \quad \text{and} \quad \bigcup_{k=1}^{N} G_k = \Omega.
\]

According to figure 8, for each pair of objects \( G_i \) and \( G_s \) of this set which own a relation, we run the algorithm which is depicted by figure 9.

**Figure 9: Algorithm to obtain connection model.**

**Relational building graph of plane connections**

During this application, we create and analyze various relations between objects using graph theory. Relations are considered in the sense that geometric objects, like bodies and faces, are connected to (an)other object(s) by connection types NEF (plane contact, cf. fig. 8) or NE (contact along edges) only. In the former case, a component graph specifies all objects which have plane contact by faces. This graph is used during the decomposition process and appended/updated during the algorithm accordingly.

**Example**

Figures 10 to 14 show a set of objects with their relations stored in an adjacency matrix and the decomposition into a connection model. Knowing about all object relations, for each pair of objects imprint faces are generated and the model is decomposed. Accordingly, the algorithm is continued recursively.

It is important to note that the algorithm is independent of the angle between objects. For a detailed review of handling particular cases and ‘experimental results’ concerning processing a complex reference building it is referred to (Seitz, 2002).

**Indoor air volume extraction**

In contrast to the graph of plane connections, a so-called edge graph is used for the analysis of air volumes by handling the set of all faces which are connected by edges only, i.e. which are of connection type NE. The algorithm of extracting all indoor air volumes being contained by the model is as follows:

All faces of the previously decomposed model are extracted to a list. Their sense of orientation, i.e. the
direction of normal vectors, is reversed. As the B-rep topology requires normal vectors pointing to the exterior side, this procedure is necessary, because we aim at describing air volumes rather than solids now. Faces that belong to the component graph are removed accordingly, because these entities are not part of an air volume’s surface. Subsequently, the edge graph is created. This requires the analysis of all face-face connections with respect to their direction and topological alignment. For all B-rep analysis, we use ACIS functionality.

All subsets of this graph representing connected graphs are analyzed with respect to obtaining minimum closed B-rep shells. In the connected graph, a path of edges and vertices exists between every pair of vertices in the graph. Leaf vertices, i.e. vertices being connected to one edge only, are not accepted here and every edge of the graph has to be occupied exactly once.

Figure 15 shows an example of a connected edge graph of an indoor air volume with inclosed column. Nodes within the graph correspond to faces which are linked by edges. The air volume consists of ten faces containing holes in two cases. Thus, for example, the top face is bounded by eight edges which connect the top face to its eight neighboring faces of the four walls and the column. Analogously, in the graph, the top face is depicted as node with eight links.

![Figure 15: Connected edge graph of indoor air volume with inclosed column.](image)

### DISCRETIZATION

The above described data structure and model decomposition now serves as precondition for mesh generation, interactive zone definition and space-tree based grid generation.

#### Structural analysis

Considering the structural analysis, starting from the connection model we create a hexahedral mesh for the FEM analysis. With respect to the numerical method, the p-version of the FEM (Düster, 2002), the model is further decomposed. For example, at all edges of slabs and plates, boundary hexahedrons are separated. Because all difference objects coincide at edges only and edges are decomposed accordingly, each object can be meshed independently of others. Assuming that all difference objects, such as slabs, plates and shells, own a dedicated face structure, we create a 2D mesh which is extruded to volume elements subsequently (Rank, 2002).

![Figure 16: Separation of boundary hexahedrons, extrusion of 2D mesh into hexahedral elements.](image)

#### Interactive zone definition

Based on the above described method, indoor air volumes are extracted and provided for an interactive zone definition. Hence, walls can be identified as interior, exterior or internal objects. A user interface allows for arranging and modifying objects and for setting up simulation parameters. As for the structural analysis, subject of our effort currently is the connection of preprocessing algorithms with both simulation kernels SMILE and our CFD code in order to obtain an integrative simulation tool.

![Figure 17: Decomposition of single room and extraction of indoor air volume.](image)

![Figure 18: Decomposition and discretization of a building product model (cf. fig. 1 and 2) with respect to structural analysis and simulation of indoor air flow. The triangulated surface mesh serves as input for an octree grid generation.](image)
Fluid domain discretization

Given a B-rep body of an indoor air volume and knowing about all relations between objects of the multizone model, we create a faceted surface mesh with attributes consisting of triangular elements. For the tessellation process, we use ACIS functionality. Subsequently, the computational domain is discretized using a space-tree based data structure, as we previously published (Crouse et al., 2002). The integration of this 'octree-generator' into our program is in process.

CONCLUSION

Interoperability among the AEC/FM related industry depends on a common language for different semantics. A promising solution represents the IFC standard but this relatively new technology is not yet well established. Particularly required are applications that are based on this exchange format. In this paper we show an approach to decompose IFC based geometry with respect to different simulation tasks. We identify processing of product model based geometry as a significant problem considering data transfer between applications.

ACKNOWLEDGMENTS

The authors acknowledge A. Lämmle and A.G. Seitz for supporting this work by their diploma theses.

This work is financially supported by the Bayerische Forschungsstiftung (Bavarian Research Foundation, http://www.forschungsstiftung.de) under grant No. AZ 468/01 (SIMFAS) and the Deutsche Forschungsgemeinschaft (DFG), program of emphasis 1103, "Vernetzt-kooperative Planungsprozesse im Konstruktiven Ingenieurbau".

REFERENCES


Rank E., Crouse B. and van Treeck C. 2002. Numerical simulation of air flow for civil engineering constructions on the basis of a product data model, 9th Int. Conf. on Computing in Civil and Building Engineering, Taipei, Taiwan.


