CHAPTER 3
DEFINITIONS AND NOTATION

In this chapter definitions and notations of quantities used in the thesis are introduced. The notation given below expresses mesh and model entity relationships in a concise form [5]. Other definitions are introduced in the relevant chapters as necessary.

3.1 Notations

3.1.1 Set notation

\[
\begin{align*}
\emptyset & \quad \text{Unordered set} \\
\sqcup & \quad \text{Ordered set} \\
\uparrow & \quad \text{Cyclically ordered set} \\
\downarrow & \quad \text{Set in which ordering of elements is unspecified and may be unordered, ordered or cyclic}
\end{align*}
\]

3.1.2 Geometric model notations

\[
\begin{align*}
G & \quad \text{Geometric model} \\
G_i^d & \quad i^{th} \text{ geometric model entity of order } d \ (d = 0, 1, 2, 3 \text{ for vertices, edges, faces and regions respectively}) \\
(g_i^d)_j & \quad j^{th} \text{ use of } i^{th} \text{ geometric model entity of order } d \\
\partial G_i^d & \quad \text{Boundary of model entity } G_i^d \\
\overline{G_i^d} & \quad \text{Closure of } G_i^d, G_i^d \cup \partial G_i^d
\end{align*}
\]

3.1.3 Mesh notations

\[
\begin{align*}
M & \quad \text{Mesh or discretization of the geometric model} \\
M_i^d & \quad i^{th} \text{ mesh entity of order } d \ (d = 0, 1, 2, 3 \text{ for vertices, edges, faces and regions respectively}) \\
(m_i^d)_j & \quad j^{th} \text{ use of } i^{th} \text{ mesh entity of order } d
\end{align*}
\]
\[ \partial M_i^d \quad \text{Boundary of entity } M_i^d \]
\[
\overline{M_i^d} \quad \text{Closure of } M_i^d, M_i^d \cup \partial M_i^d \\
\square \quad \text{Classification of } M_i^d \text{ on } G_j^D \quad \text{which is the unique association of } M_i^d \\
\quad \text{with } G_j^D \text{ if } M_i^d \text{ forms all or part of the discretization of } G_j^D. \\
\quad \text{Classification of } M_i^d \text{ on } G_j^D \text{ is written notationally as } M_i^d \sqsubseteq G_j^D. \\
\quad \text{It follows that } d \leq D. \\

3.1.4 Adjacencies

\[ \varphi^{d_1}(T_i^{d_1}) \quad \text{The set of entities in model } T \text{ of dimension } d \]
\[ \quad \text{that are adjacent to an entity or set of entities } \phi \text{ of dimension } d_1 \]
\[
(T_1^{d_1}, T_2^{d_2}, \ldots, T_n^{d_n})(T_i^{d_1}) \quad \text{The set of entities of dimension } d \quad \text{that are adjacent to the entities } T_1^{d_1}, T_2^{d_2}, \ldots, T_n^{d_n}. \\

3.2 Definitions

The basic input for any automatic mesh generator is a properly defined geometric model and a set of meshing attributes prescribed on the model. All models are expected to contain a definition of their topology, and a definition of the geometry underlying the topological entities (points, curves and surfaces).

3.2.1 Geometric model definitions and concepts (Also see [43, 51])

Geometric models may, in general, be 2-manifold or non-manifold. Informally, non-manifold models are models which are general combinations of solids, surfaces and wires. A more formal definition is given below. The difference between 2-manifold and non-manifold models is illustrated in Figure 3.1.

**Definition 3.1** 2-manifold models are geometric models in which the local neighborhood of every point on the model boundary is topologically equivalent or homeomorphic\(^2\) to a disk [43, 72-74].

\(^2\)Intuitively, this means the local neighborhood of every point on the boundary may be transformed into a disk without any cutting, tearing or otherwise making points that were separate, coincident in the new form. See [49] for a discussion of topology and homeomorphism.
Definition 3.2 All models that are not 2-manifold are non-manifold.

Geometric models may have two types of non-manifold faces - embedded faces and interfaces (See Figure 3.2).

Definition 3.3 Interface faces in geometric models are faces that partly bound two different model regions, one on each side.

Definition 3.4 Embedded faces in geometric models are faces that are connected to the same region on both sides.

The data structure used to represent the model in this work is based on the Radial Edge Data Structure developed by Kevin Weiler [72–74]. The Radial edge data structure presents the idea of uses to represent how topological entities are used by others in a non-manifold model. Every face in the model has two face uses, one on each side of the face. An edge carries as many pairs of uses as there are pairs of face uses coming into it. Each edge use has an edge use mate. A vertex carries as many uses as there are edge uses coming into it and each vertex use has one vertex use mate. This is illustrated in Figure 3.3 (adapted from [72]).

The radial edge data structure is more detailed than the minimum amount of information required to represent many common types of non-manifold model. The representation is therefore reduced by fusing edge use mates together to form a single “edge use” connected to two face uses. Similarly, vertex uses are condensed so
that the minimum number of uses are present at any vertex. Such a data structure is referred to as the \textit{Minimal Use Data Structure} [4]. Conceptually, the minimal use data structure builds a representation of the non-manifold topology such that the connected face uses locally form a 2-manifold at any point. For example, every edge use in a minimal use is connected to two or no faces uses. This is even true for embedded faces in the model. For example, a rectangular face completely embedded in a region, has two faces with common edge and vertex uses (like a pillow case). Similarly, the use topology at any vertex use can always be represented locally as a 2-dimensional disk. Lastly, by its very nature the use topology at any face use is always a 2-dimensional disk.

\textbf{Definition 3.5} Given an entity use, \((g^d_i), j\), \(d = 0, 1, 2\), the \textit{collection of its connected face uses}, \(\{(g^2_k), m \mid (g^d_i), j \subset \partial((g^2_k), m)\}\) in a minimal use representation of a non-manifold model is called a \textbf{manifold}.

At any point on a model face there are always two manifolds of face uses. One, both or none of these manifolds may partly or completely bound a model region. Points on edges and vertices may have multiple face use manifolds connected to them. As with faces, each of these manifolds may or may not form part of the
boundary of model region. At any point it is not possible to travel from one of its mesh manifold to another without penetrating a boundary model entity. The concept of a face use manifold is similar to the idea of a separation surface defined by Weiler [72]. Weiler describes a separation surface as “a complete surface formed by the juncture of faces around a vertex that effectively separates the space immediately around the vertex into two half-spaces, distinguishable from each other because the surfaces are orientable.” Separation surfaces may be made up of one or more model faces as long as they form a continuous surface at the vertex. One or more separation surfaces may exist at any vertex. 2-manifold models are characterized by the presence of only one model region and only one manifold at each point on the model boundary connected to the region.

With the help of the minimal use representation and its collection of face uses into manifolds, dealing with a non-manifold boundary becomes equivalent to dealing with a set of 2-manifold boundaries. Figure 3.4 shows the minimal use representation equivalent of the non-manifold situations shown in Figure 3.3. For geometric modelers using a purely 2-manifold representations, non-manifold models may still be built up in a non-manifold data structure from multiple 2-manifold components if the appropriate additional information is specified [63, 67].
3.2.2 Mesh definitions and concepts

The representation for the mesh [5, 57, 58, 66] used in this research is based on concepts from geometric modeling. The mesh consists of mesh vertices, edges, faces and regions. If necessary, the mesh may also represent vertex, edge and face uses. Each entity in the mesh has a unique classification with respect to the model.

**Definition 3.6** Classification is the unique association of a mesh entity, $M^d_i$, to a geometric model entity, $G^d_j$, ($d_i \leq d_j$) to indicate that $M^d_i$ forms part or all of the discretization of $G^d_j$ but not $\partial G^d_j$. The classification operator is denoted by $\sqsubset$ and $M^d_i \sqsubset G^d_j$ is used to denote the classification of $M^d_i$ on $G^d_j$.

The geometry of mesh vertices is described by points associated with them. The geometry of mesh edges and faces is not stored for linear elements. For higher order elements, the geometry of edges and faces is a polynomial or other interpolation implicitly defined through additional information stored with the mesh entities or with the geometric model in terms of the shape of the model entities they are classified on.

**Definition 3.7** The connectivity of a mesh or model entity to other mesh or model entities respectively is called Topological Adjacency or simply Adjacency.
Definition 3.8 A valid mesh is one that correctly approximates the geometry of an object.

The implication of this definition is that the mesh should topologically and geometrically equivalent or congruent to the geometric model. Schroeder and Shepard [57,58] lay down the following conditions for validity of a mesh:

1. The mesh should be topologically compatible with the geometric model.

2. The mesh should be geometrically similar to the geometric model.

Definition 3.9 Topological Compatibility: Given a mesh consisting of mesh entities $M_i^d$ with boundary entities $M_i^{d-1}$ classified on the closure of a geometric model entity $G_j^d$ with boundary entities $G_j^{d-1}$, the mesh is topologically compatible with the model if

1. Each $M_i^{d-1} \subseteq G_j^d$ is connected to two and only two $M_i^d \subseteq G_j^d$.

2. Each $M_i^{d-1} \subseteq G_j^{d-1}$ is connected to as many number of $M_i^d \subseteq G_j^d$ as the number of times $G_j^{d-1}$ is used by $G_j^d$.

A mesh is topologically compatible with a geometric model if it is compatible with each of the model entities.

Geometric similarity [57,58] is the relationship of the mesh geometry to the model geometry and is a way of expressing the condition that in the limit of refinement the geometry of a mesh should exactly match that of the geometric model. This idea may be expressed more practically in a number of ways, one of which is expressed by Schroeder and Shepard [57,58] as follows:

Definition 3.10 Geometric Similarity: A mesh of order $d$, comprised of $N$ entities $M_i^d$ is geometrically similar to a model entity, also of order $d$ ($G_k^d$), if $M_i^d \subseteq G_k^d$, $\forall i = 1, \ldots, N$ and the parametric intersection of any two entities of the mesh is $\emptyset$, i.e. $M_i^d \cap^* M_j^d = \emptyset$, with the parametrization being with respect to some appropriately defined space.
Since the mesh is required to topologically compatible with the geometric model, one can define a concept for meshes that is analogous to a face manifold or separation surface in geometric models.

**Definition 3.11** A mesh manifold is a set of mesh face uses around a vertex, connected by edge uses, that locally separate the three dimensional space into two halves.

Some examples of mesh face use manifolds are shown in Figure 3.5. In Figure 3.5a, mesh manifolds for a mesh vertex classified on a model face, $M_v^0 \subseteq G_0^2$, is shown. Since the topology of a model face by itself is 2-manifold, the mesh vertex has two mesh manifolds ($S_v^0$ and $S_v^1$), one for each side of the model face. In Figure 3.5b, mesh manifolds are shown for two mesh vertices classified on model vertices in a non-manifold model. In the figure, $G_1^2$ is an embedded face making edge contact with two model faces $G_0^2$ and $G_2^2$. The mesh manifolds in the picture are depicted only with respect to the model region common to all three faces. The local topology at $M_a^0$ is non-manifold and two mesh manifolds, $S_a^0$ and $S_a^1$, exist at the vertex with respect to just one side of the model faces $G_0^2$ and $G_1^2$. At $M_b^0$, only one mesh manifold, $S_b^0$, exists in the model region under consideration. Note how this mesh manifold wraps around the free edge of the embedded face and uses both sides of mesh faces classified on the embedded model face. In Figure 3.5c, a non-manifold topological situation at a model vertex is depicted. Assuming that each mesh face shown is classified on a model face embedded inside a model region the mesh vertex has 3 mesh manifolds connected to it as shown in the figure.
Figure 3.5: Examples of mesh face use manifolds.