Triangular Mesh and Boundary Representation Combined Approach for 3D CAD Lightweight Representation for Collaborative Product Development

Cong Hong Phong Nguyen
3D Digital Engineering Laboratory,
Department of Mechanical Engineering,
Graduate School,
Chung-Ang University,
Seoul 06974, South Korea
e-mail: hphong1990@cau.ac.kr

Young Choi[1]
3D Digital Engineering Laboratory,
School of Mechanical Engineering,
College of Engineering,
Chung-Ang University,
Seoul 06974, South Korea
e-mail: yychoi@cau.ac.kr

The lightweight representation of three-dimensional computer-aided design (3D CAD) models has drawn much attention from researchers as its usefulness in collaborative product development is vast. Existing approaches are mostly based on feature depression or mesh-based simplification. In this article, a new approach for 3D CAD lightweight representation based on combining triangular mesh representation and boundary representation (B-rep) is proposed. The corresponding data structure as well as the conversion method from original data given in B-rep was developed. Considered as an essential application in collaborative product development, a case study on the visualization process of large-scale assembly models represented in the proposed lightweight representation was also conducted. The validation of the approach was performed via experiments with 3D CAD models in SAT format and by benchmarking with the conventional all-faceted approach with the same level of mesh resolution. [DOI: 10.1115/1.4041777]

Keywords: 3D CAD lightweight representation, mesh and B-rep combined representation, 3D geometric modeling, collaborative product development

1 Introduction

In modern product development processes, the demand for three-dimensional computer-aided design (3D CAD) models is enormous since they support many stages in the product development process. In the modern product development pipeline, the involvement of many engineering fields other than geometric design is required at early stages of the product design pipeline. Hence, 3D CAD models need to be shared between engineering teams frequently, especially via the Internet [1,2]. Furthermore, the collaborative product design process can take advantage of utilizing 3D CAD models for fast visualization during product design reviews [1–3]. In addition, 3D CAD models could be valuable in publishing 3D technical reports and product catalogs for marketing or serve as 3D graphical guidance at manufacturing shops. Thus, in collaborative product development, 3D models are required to be efficient in terms of both data storage and data processing.

Unfortunately, original 3D CAD models without modification are usually not directly applicable throughout the product lifecycle management (PLM) system [3]. First, original CAD models are not efficient for frequent data sharing, especially via the Internet. Mostly represented in boundary representation (B-rep) schemes, they are heavy with geometric information and other attributes that are normally useless in applications other than 3D design. Additionally, the existence of assembly constraints between parts in assembly models is another issue that also increases the complexity of 3D CAD models. The interoperability between 3D CAD and other engineering applications is also another concern. Although many CAD systems provide their own collaboration environments, which allow the users to manipulate their 3D data directly (e.g., product data management and finite element analysis applications), the usefulness of these solutions is restricted within each CAD system. Finally, utilizing the original 3D CAD data directly is usually not allowed owing to the security risk in design information since most downstream applications do not need the utilization of the full 3D CAD data [3,4].

Therefore, there is a need for a lightweight representation of 3D CAD models, which satisfies the requirements of PLM systems and overcomes most of the limitations on model complexity, interoperability, and design information security risk with the original 3D CAD data [5]. In this article, we propose a new approach for the lightweight representation of 3D CAD data, which reduces the data storage requirements of 3D CAD files while keeping the data processing efficiency at a reasonable level. The lightweight representation is achieved by altering the geometric representation scheme of 3D objects. Herein, geometric objects are represented using a scheme, which is a combination of triangular mesh representation and B-rep. This type of hybrid representation can be considered as an optimal solution for the development of the 3D CAD lightweight representation in terms of both data storage and processing efficiency.

The rest of the paper is organized as follows: Section 2 contains a description of the background on geometric representation schemes for 3D modeling in CAD and a review of some related works in the field of 3D CAD lightweight representation. In Sec. 3, the development of the lightweight representation and its conversion from 3D CAD data given in B-rep is presented. A case study on the visualization process of large-scale assembly models represented by the proposed lightweight CAD format (the most typical application in most collaborative product development systems) is presented in Sec. 4. Finally, conclusions are given in Sec. 5.

2 Literature Review

2.1 Previous Geometric Representation Schemes in Three-Dimensional Computer-Aided Design Modeling. Geometric representation schemes have had a considerable impact on the

[1]Corresponding author.

Contributed by the Computers and Information Division of ASME for publication in the Journal of Computing and Information Science in Engineering. Manuscript received July 23, 2018; final manuscript received October 9, 2018; published online November 19, 2018. Assoc. Editor: Charlie C.L. Wang.

efficiency of computer-aided geometric design applications. The choice of 3D-shape representations is vast and depends on the algorithm performed on them. The two most well-known geometric representation schemes for 3D CAD modeling are B-rep [6] and (polygonal or) mesh representation [7].

Boundary representation with canonical geometry and nonuniform rational B-spline (NURBS) for curve and surface modeling is considered as the most effective method for 3D modeling and is currently used in most CAD systems. The advantages of B-rep are its popularity and flexibility in geometric modeling, especially in combination with NURBS [6,8]. However, the main drawback of B-rep is that its data structure is heavy and full of topological connections and abstract topology objects (such as loops, shells, and lumps) [6], which increases the demand for data storage, particularly when the 3D models become more complex.

In contrast to the exact geometric representations in B-rep for which NURBS and canonical geometry are used, mesh representation represents geometric objects approximately by linear (for one-dimensional) and planar (for two-dimensional) geometry. Because any geometric object can be represented as a set of faces, edges, and vertices, the mesh representation data structure is more straightforward than that of B-rep. Additionally, in comparison with B-rep, mesh representation is more general and flexible in surface modeling. While B-rep tends to divide complex surfaces into more straightforward surface patches, mesh representation represents these complex objects by a set of planar facets. However, there is a drawback in representing 3D objects with the mesh representation scheme in that the size of mesh models increases abruptly when high accuracy is required [9].

There are also other well-known representation schemes for 3D CAD modeling, such as the constructive solid geometry representation and the decomposition representations (e.g., voxel representation and cell decomposition). However, their usage is far from the application of 3D CAD lightweight representation and so they are not discussed in detail here.

2.2 Related Work on the Lightweight Representation of Three-Dimensional Computer-Aided Design Models. The key to achieving a lightweight representation of 3D CAD data is to reduce the data storage consumption of the geometric objects since the geometric information consumes the majority of the data storage in a 3D CAD file. Several methods for 3D CAD file size reduction have been developed, including geometric data compression (domain-specific, 3D graphic compaction), CAD model simplification, and the reference-instance scheme [10]. As has been reported in the literature, approaches for achieving the lightweight representation of 3D CAD data fall into three main categories: (1) approaches utilizing the standard lightweight 3D file format, (2) mesh-based representation approaches where the compression and data size reduction technique is performed based on mesh representation, and (3) 3D CAD model simplification approaches where the size reduction is achieved by the simplification of the 3D CAD models. Figure 1 shows some of the notable studies in the development of 3D CAD lightweight representation.

There are some existing international standards for the lightweight representation of 3D objects which are customized for Web3D applications. X3D [11] and U3D [12] are the most common formats for fast visualization and simulation via the Internet. For fast visualization, most geometric objects are represented in mesh form except for elementary objects such as boxes and cylinders. Moreover, for faster real-time rendering, most of the engineering data associated with the CAD model are removed. Additionally, several conventional techniques to reduce the size of lightweight files such as the reference-instance scheme and the domain-specific compression technique have also been applied, but because most of the engineering data in the design models are eliminated in these two formats, their usage in engineering collaboration is limited. The solution to this problem is that the engineering information should be attached with lightweight and is only loaded when required. Some notable efforts in extending the usage of standard lightweight formats are the studies by Kim et al. [3] and Geng et al. [13]. JT [14] and 3D extensible markup language (XML) [15] are two other open standard formats provided by Siemens PLM Corp and Dassault Systemes, respectively; they are commonly adopted for engineering data sharing in collaborative product development. Besides graphics data represented by facets and B-rep schemes (in JT) and NURBS-like freeform surfaces (in 3D XML), both formats allow the engineering data to be stored. Furthermore, a relatively high compression rate (up to 90%) compared with the original 3D CAD data can be achieved with these standard formats.

Following the mesh-based approach, Song and Chung in a series of studies proposed a lightweight format customized for a web-based dimensional verification system, which utilizes triangular mesh for 3D modeling with the attachment of edge

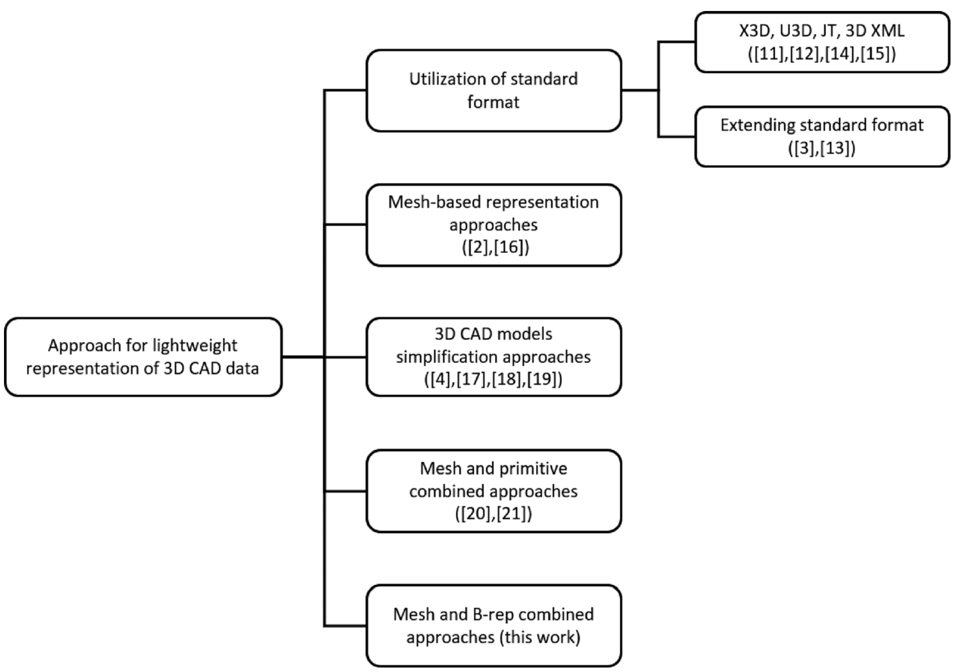


Fig. 1 Related approaches to lightweight representation of 3D CAD

information to support the dimensional verification process instead of having a full geometric design dataset [2,16]. Their results show that the reduction of the lightweight file can reach up to 90% and the lightweight data generation time is not considerably increased even when the sizes of the input CAD files vary widely.

Attention has also been paid to the approach of 3D CAD model simplification to reduce the size of the CAD files. Liu et al. proposed a method for size reduction in the development of a lightweight format for CAD collaboration based on feature-based simplification customized for large-scale assembly models [4]. First, the assembly model is traversed and unnecessary features such as filets, chamfers, and non-important holes are removed. Consequently, the solid model is converted into a shell model to remove all inner entities since the use of a full feature model is sometimes unnecessary. Finally, the assembly models are converted into a single part by merging all the components in the assembly into one, thereby reducing the size of the lightweight output file. In similar work by Kwon et al. [17], the authors also approached the problem by removing non-important features to reduce the CAD file size but keeping the essential features for the assemblies such as holes and connection ports. The approach results were auspicious with a size reduction of up to 27% with the tested models, and the 3D CAD model is suitable for use in large-scale assembly models. Kim et al. similarly attacked the problem with feature-based models translated from B-rep data [18]. The method resulted in a multiresolution model with a different level of detail and the level of lightweightness being determined by the user. Gao et al. detected and suppressed unnecessary features such as ribs and blended features (e.g., filet and chamfer features) from a mesh-based CAD model [19]. The results from using their approach seemed to be better than the other mesh-based simplification methods, albeit a current limitation of the method is its mesh segmentation process and a controllability issue for the level of simplification.

Related to the purpose of reducing the size of mesh-based models, Lafrange et al. proposed an approach that involves the insertion of 3D primitives to the mesh-based model for data compression [20,21]. The main idea was to replace the mesh patches with primitive 3D objects (e.g., planes, cylinders, cones, tori, and spheres) to reduce the model file size. The approach showed some positive results with a reasonable compression rate, but a limitation of the method was the inability to exact piecewise 3D-primitives attached in a cluster, which reduced its flexibility. Furthermore, the method is applicable to large-scale mesh-based models obtained from laser scanners; thus, the practicality of the approach when applied to a lightweight representation in collaborative product development is still in doubt.

3 The Development of the Mesh and Boundary Representation Combined Lightweight Representation

As can be found in the literature for applications in collaborative product development other than 3D design where controllability and accuracy are required, most of them, such as reviewing or graphical support-based ones, do not need the full design information of the 3D CAD models. Therefore, mesh representation is preferred over B-rep for lightweight representations in engineering collaborations owing to its simplicity and flexibility in 3D modeling even though it consumes more data storage than B-rep at a certain level of required accuracy (Fig. 2). Furthermore, with certain types of geometric object (such as canonical geometric objects), representing them in B-rep with canonical geometry for the curves and surfaces modeling can be advantageous over the mesh representation scheme, particularly when the required accuracy is high.

To overcome the limitations of mesh representation in a 3D CAD lightweight representation, the idea is to combine mesh representation with B-rep into one representation scheme. This can maximize the advantages of mesh representation in representing complex geometric objects while minimizing its limitation with the appearance of B-rep objects when required. The potential of the approach of combining analytic and discretized geometry models

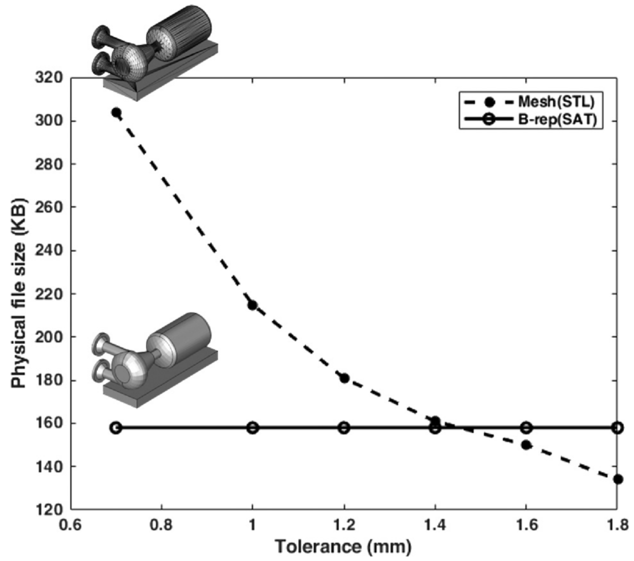


Fig. 2 The size difference of a 3D model represented in B-rep and mesh representation

has been proved in a series of studies by Lafrange et al. [20,21]; their approach showed a high compression rate while the processing speed and memory consumption were kept at reasonable levels.

There are also several approaches and data structures available for combining mesh representation and B-rep for 3D modeling. The first option is to adopt the existing B-rep data structure directly, the reason being that mesh representation can also be considered as a particular case of B-rep where the face entities only hold planar geometry and are bounded by a set of line segments called polyline. Thus, the B-rep data structure can be used for mesh representation where each facet is represented as a planar face bounded by a set of straight-line loops. However, there is a critical issue in this approach in that the data structure is extremely heavy due to the existence of irrelevant purely abstract topological objects such as loop and shell entities. Another approach is to consider the surface mesh as one type of surface in B-rep geometry; the surface mesh objects are represented as B-rep objects with geometric information represented by a set of facets, and the bounding edges of the mesh face are represented by a set of line segments. Nevertheless, the limitation here is that with the mesh-based face, there is the additional appearance of loop topological entities, which are useless for representing surface mesh objects.

Ferrari proposed an enhanced approach in which the author considered a mesh-based object as one type of face (called the mesh-face) bounded by a set of polylines [22]. The continuity between the mesh and NURBS faces is determined by the concept of approximate continuity, which was introduced by Besl [23]. The approach has overcome the limitation of creating useless loop topological entities by separating mesh faces with NURBS-based B-rep faces. However, there is a potential problem because unlike B-rep, which commonly has shared edges between adjacent faces, there are two types of edge at the interface of a mesh-face and a NURBS-based face in two different forms, which basically represent just one physical geometric edge. This duplication issue reduces the efficiency of the proposed data structure regarding data storage. Figure 3 illustrates how 3D CAD data using the mesh and B-rep combined model can be represented via the aforementioned methods.

The proposed lightweight representation was developed to overcome the problem with the existing approaches in developing the combined mesh and B-rep representation scheme. In the proposed representation scheme, a body can contain either a set of B-rep faces called "H-Brep" faces or a set of meshed faces called "H-Mesh" faces. A simple face-vertex data structure is adopted for the H-Mesh face data structure while the conventional B-rep data structure is applied for the representation of H-Brep faces.

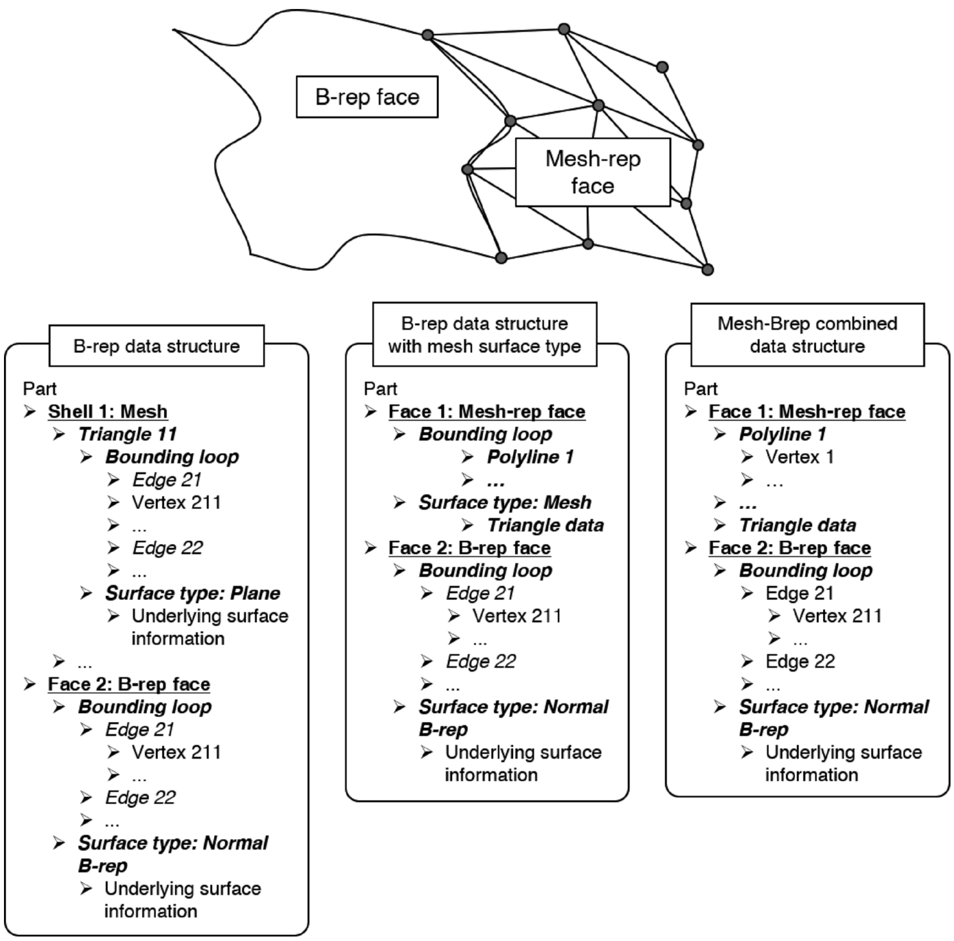


Fig. 3 Approaches for 3D data representation in the combined mesh and B-rep combined model

The duplication of edge representation mentioned previously is overcome by the existence of a hybrid edge with dual geometric representation. Details of the proposed lightweight representation scheme and its conversion from B-rep are discussed in detail in the following three sections.

3.1 Mesh and Boundary Representation Combined Representation Scheme. Figure 4 illustrates the data structure for the proposed lightweight mesh and B-rep combined lightweight representation scheme. The concept of topology and geometry for 3D modeling similar to the B-rep data structure is adopted. While the geometry plays the role of representing physical shapes of geometric objects such as their underlying curves and the surfaces or their facet data, the topology comprises the glue to connect the geometric entities to form the 3D model.

In contrast to existing approaches, we add both a new type of face (the H-Mesh face) and new types of curve and surface geometry entities (the triangle set and the polyline). The benefit of doing so is to keep the advantages of both mesh representation and B-rep data structures in the geometric representation and separate discretized and analytical geometric objects. In the proposed lightweight representation scheme, a body contains a set of faces, which can be either H-Brep or H-Mesh. Note that for better size reduction, several purely abstract topological entities such as lumps and shells are not adopted since their usage in applications other than 3D design is very limited. Thus, multibody models are represented as separate bodies. As illustrated in Fig. 4, the H-Brep faces are defined by their underlying surfaces and bounding loops, which are constructed with a series of edges. Each edge has its curve information and is limited by start and end vertices. In contrast, an H-Mesh face is defined by its triangle set and bounding edges with polyline geometry information.

However, the most remarkable development in the proposed approach is the concept of a hybrid edge entity, which has a dual geometric representation and contains both curve and polyline data. There are three possible cases of an edge existing: (1) the edge is shared by two H-Brep faces, (2) the edge is shared by two H-Mesh faces, and (3) the edge belongs to both an H-Brep and an H-Mesh face. In the first two cases, modeling the interface is relatively straightforward, and the existing data structure can execute the task efficiently. However, in the third case, when the edge is required to have two different representation forms, the problem is more complicated. As mentioned previously, modeling the interface between the H-Brep face and the H-Mesh face is the current limitation of most existing approaches. In these, the B-rep and mesh representation form of the same bounding edge both exist, and thus increase the data storage requirement. In our proposed data structure, with dual geometric representation, an edge entity can be the bounding edge of either an H-Brep face or an H-Mesh face, thereby providing the ability to model the interface between adjacent H-Brep and H-Mesh faces. A shared edge between H-Brep and H-Mesh faces can be represented as a hybrid edge, and as a result, the edge entities in the proposed data structure can be classified into three types: (1) pure H-Brep edges, (2) pure H-Mesh edges, and (3) hybrid edges. The concept of the hybrid edge is also the advantage of our proposed method over existing ones for the mesh and B-rep combined representation.

Figure 5 illustrates a 3D model represented in mesh representation, B-rep, and the mesh and B-rep combined representation schemes.

3.2 Lightweight Three-Dimensional Computer-Aided Design File Structure. Figure 6 depicts the proposed lightweight file structure. As shown, the lightweight physical document

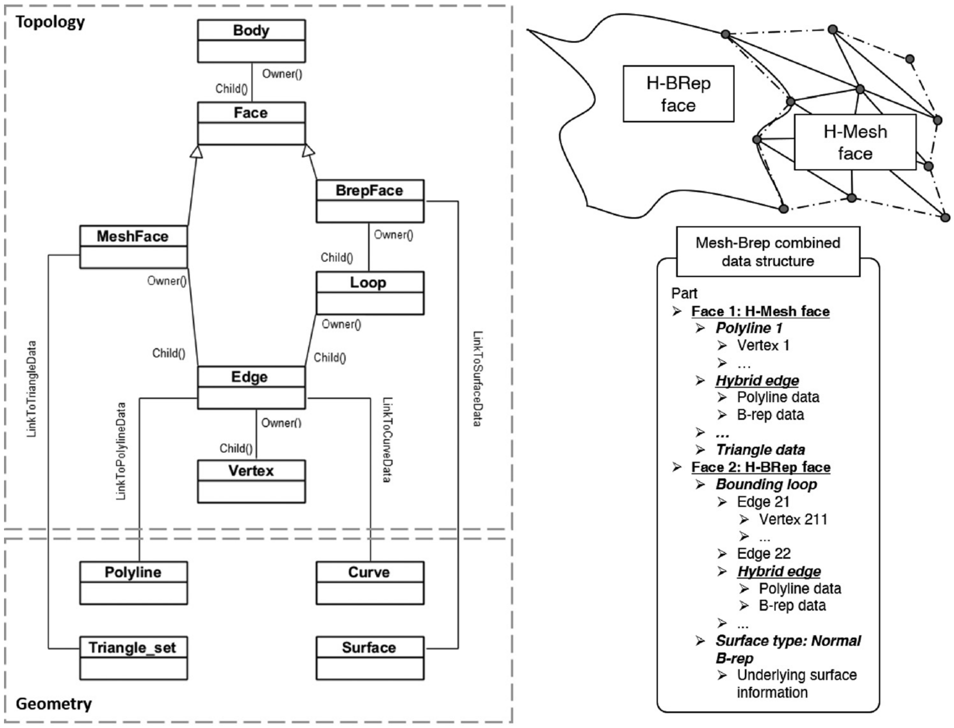


Fig. 4 Proposed mesh and B-rep combined representation data structure

contains three main blocks: (1) the model data block, (2) the geometry data block, and (3) the extended data block.

The model block includes the product assembly with subassemblies and parts. The relative position of each sub-assembly and part with respect to the global coordinate system is determined by the transformation information. Each part entity has a connection to its geometry and extra information. Here, the reference-instance scheme, which is prevalent in existing standard lightweight CAD formats, is also supported. Different parts can share the same geometry, rendering, and metadata information; thus duplication of information is avoided with the proposed scheme.

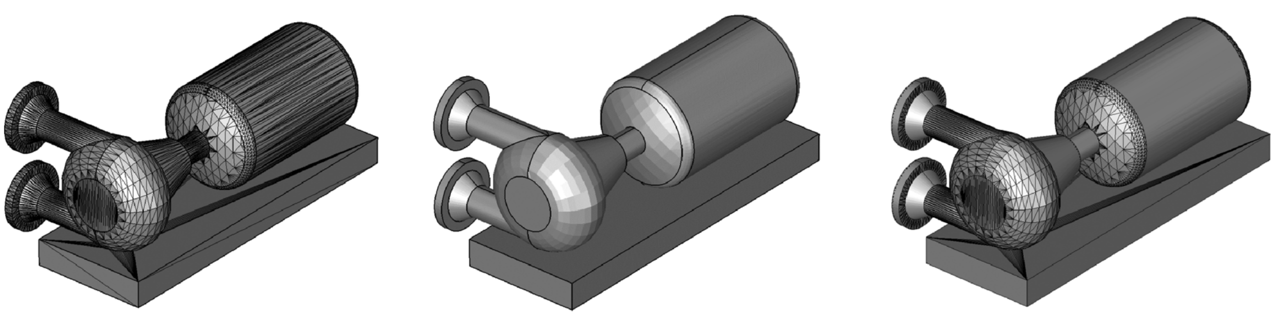
The geometry data block contains the geometric representation data of each part entity. The geometric data contain a list of geometric objects represented in the combined mesh representation and B-rep scheme, as described in Sec. 3.1.

The extended data block is used to extend the usage of the proposed lightweight file format. Some examples of extended data are rendering data for supporting fast visualization (such as light, view, texture, and material resource information) and metadata containing markup data for reviewing (such as for engineering analysis and product manufacturing). The type of extended data can be varied depending on the application of the lightweight file. However, this is not discussed here as it is out of the scope of this study.

For standardization and fast development, the lightweight format was developed based on the XML [24]. Since XML is a standard self-defined language, the proposed concept can be easily expanded and integrated into existing collaboration systems or adopted as the core for a newly developed collaborative product development system. The XML document is compressed for size reduction with the commonly used compression technology provided by Zlib [25].

3.3 Lightweight File Conversion From Boundary Representation Data. In the collaboration between engineering applications, the original data are mostly from design systems. Most standard CAD formats for exchanging CAD data such as STEP, IGES, SAT, and Parasolid utilize the concept of B-rep. Thus, the conversion from B-rep format into the proposed lightweight format needs to be considered.

Figure 7 shows a description of the B-rep data conversion process. First, the process will parse through the B-rep model and classify face entities into two groups: a set of faces, which should be converted into mesh representation (H-Mesh), and a set of faces, which should be kept as it is in the B-rep form (H-Brep). After being classified, two groups of face entities will be



(*a*) (*b*) (*c*)

Fig. 5 The three different 3D model representation schemes: (*a*) mesh representation, (*b*) B-rep, and (*c*) mesh and B-rep combined

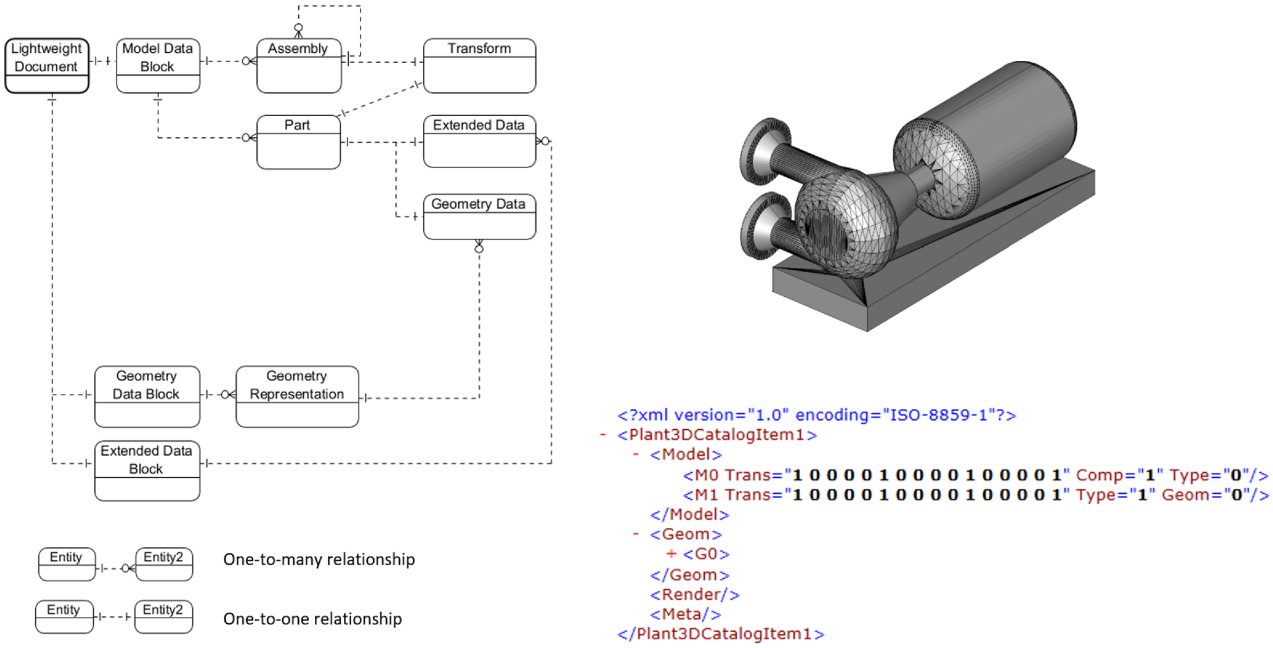


Fig. 6 The proposed lightweight file structure and its representation in XML format

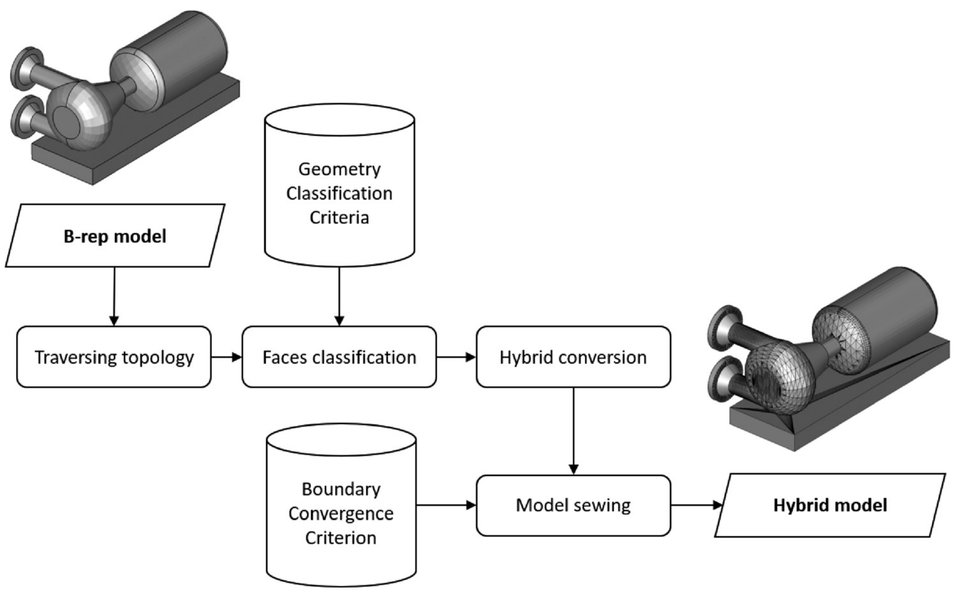


Fig. 7 Lightweight format conversion process from B-rep data

converted into the proposed face entities. The process is executed face-by-face in sequence to produce the mesh and B-rep combined model. However, at this stage, there is no connection between faces; thus, the acquired model is considered as a "geometry soup." Therefore, a model sewing process is required to connect the geometric entities topologically and eliminate duplicates. As mentioned, there are two crucial steps in the conversion process: (1) the classification of the input B-rep faces and (2) the convergence criteria to connect the adjacent H-Brep and H-Mesh faces topologically.

3.3.1 Face Classification. From the input B-rep data, determining which types of geometry should be represented as H-Brep objects and which should be represented as H-Mesh objects is essential because the classification affects the downstream computation and applications. Given the fact that most lightweight 3D CAD models are mainly used in visualization-based applications (e.g., reviewing, publishing 3D reports and catalogs, and 3D simulation), we propose classification criteria based on the requirement of the visualization process.

In conventional B-rep, three types of geometries could be represented, namely analytic geometry, procedural geometry, and NURBS geometry [26]. Analytic geometry is defined by a classical explicit mathematic expression (e.g., canonical geometry), whereas most of the more complex geometries are defined with the concept of NURBS [8]. In addition, procedural geometry, which usually arises from modeling operations such as intersection, offsetting, blending, or sweeping [26] could be represented effectively with either the analytic or the NURBS geometry concept [8]. Although the NURBS geometry concept could also be used for representing elementary geometric objects such as canonical geometry, common CAD kernels [26,27] tend to

Table 1 Curve and surface classification in the proposed lightweight representation scheme

Type	B-rep geometry type	Proposed representation geometry type	Defining parameter	Memory cost (bytes)
Surface	Elementary analytic surface	Plane	Root point Normal vector	24
		Cone	Center point Direction vector Cone angle Radius	36
		Circular torus	Center point Normal vector Minor radius Major radius	32
		Sphere	Center point Radius	16
	NURBS and trimmed analytic surface	Triangle set	Facet data Vertex data	Dependent on the meshing option
Curve	Elementary analytic curve	Straight line	Root point Normal vector	24
		Ellipse	Center point Major axis Normal vector Radius Radius ratio	44
	NURBS curve	Line segment set	Vertex data	Dependent on the meshing option

represent that geometry in analytical form rather than in NURBS form. Thus, the classification is reduced into two types, i.e., analytic geometry and NURBS geometry.

From the aspect of data storage efficiency, at the average level of accuracy, elementary geometric entities such as canonical geometry (e.g., plane, cone, torus, and sphere) would contain less data when they are saved in analytical form. Moreover, those geometric objects are relatively straightforward; therefore, the time consumed for faceting those entities is reasonable when the mesh model is required whereas that of NURBS geometry is usually long. In practice, elementary geometric entities are mostly faceted by template-based methods, while NURBS geometry must be sent through a computational process for faceting, which takes longer. Furthermore, when the controllability of the 3D model is not required in applications other than 3D design (such as in reviewing), keeping those complex geometries in their exact NURBS form is not necessary. In addition, in the case of a trimmed surface, which is a particular type of procedural geometry that is widely used in 3D computer-aided geometric modeling, the meshing process would be very complicated as mentioned in the literature [28–32], rendering it unsuitable for the fast processing requirements of the collaboration system. Additionally, representing complex geometric objects in mesh representation is more convenient and general, especially when a new type of geometry is added.

Thus, the idea of B-rep face classification herein is to represent non-trimmed elementary analytic surfaces such as planes, cylinders, cones, tori, and spheres in the B-rep form while converting the rest into mesh representation for convenient downstream operations. A summary of the geometries supported in the proposed lightweight representation scheme and their memory requirements are listed in Table 1.

3.3.2 Model Sewing. After the conversion of the classified B-rep faces, the acquired hybrid model is a set of faces without any topological connections between them. Utilizing the topological connection from the original B-rep model, the sewing process is conducted to connect these faces topologically for efficiency in data storage and downstream computation.

Algorithm 1 illustrates the sewing process with the prior knowledge on the shared edge information from the original 3D CAD data. The topological information containing the shared edges information of the adjacent faces is attached to the converted combined mesh representation and B-rep model. In the sewing process, all of the edge (both B-rep edges and polylines) in the model are added to a candidate list. For each candidate edge or polyline and with the prior knowledge of the topological connection, the neighborhood edge coincident with the candidate edge can be elucidated. If the candidate edge and its neighborhood are both polylines or B-rep edges, they are converged, which means that one of the edges is eliminated.

Consequently, the information about the owner face of the eliminated edge is attached to the other one. If the candidate edge and its neighborhood are different edge types, the convergence criteria described in Definition 1 are checked. If these are satisfied, a hybrid edge containing both the polyline and B-rep edge information is constructed and used to replace both candidate edges; else, the candidate edge is set as free. After the sewing process, some free edges may appear, and to converge these, a model healing process is needed. In this case, a joint operator as described in Ref. [22] can be utilized to solve the problem.

Algorithm 1. Sewing edges of mesh and B-rep combined body

Given:
A list of all the polylines and B-rep edges in the mesh and B-rep combined model $\boldsymbol{C}$
For each C_{candi} in $\boldsymbol{C}$
Check the neighborhood edge C_{check} of C_{candi} in $\boldsymbol{C}$
If C_{check} and C_{candi} are both B-rep edges or polylines
Converge C_{check} and C_{candi}
Update the owner face information of C_{check} and C_{candi}
Remove C_{check} and C_{candi} from $\boldsymbol{C}$
Else
Check the convergence criteria
If the convergence criteria are satisfied
Create a hybrid edge HE containing information on both C_{check} and C_{candi}
Update the topological information of C_{check} and C_{candi} to HE
Delete C_{check} and C_{candi}
Else
Set C_{candi} as free
Remove C_{candi} from $\boldsymbol{C}$
End for
Return:
A mesh and B-rep combined sewed body.

In the model sewing process, the most vital issue is to determine the convergence criteria by which the H-Mesh faces can be jointed to H-B-rep faces topologically. In this approach, the approximation of the C^0 continuity concept (AG^0) for the

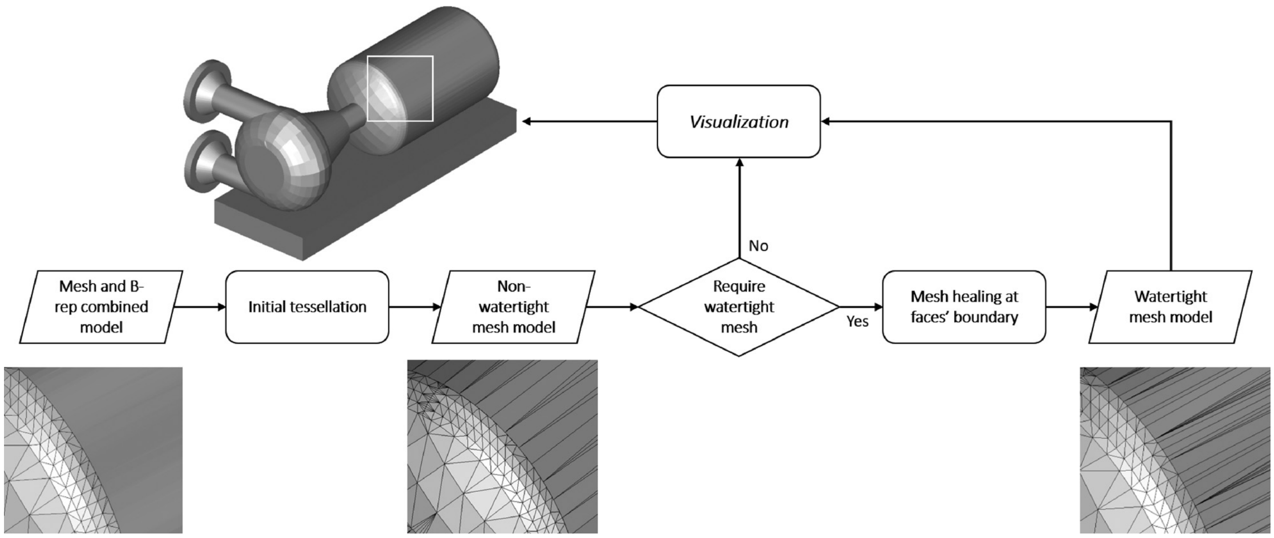


Fig. 8 Visualization pipeline of the mesh and B-rep combined model

convergence criteria is adopted. The concept was first introduced by Besl [23] and finalized mathematically by Ferrari [22] in a set of approximate geometric continuity definitions (AG^0 and AG^1). Definition 1 describes the convergence criteria of our proposed approach.

The convergence value τ is set based on the scale of objects rather than on a global value. In our proposed method, the convergence value is set based on each part's oriented bounding box dimensions with a relative preset tolerance. By doing so, one avoids the inappropriate error caused by relatively small components in large assembly models. The convergence value is defined as

$$\tau = \frac{\sqrt{\mathrm{BoundBox}_x^2 + \mathrm{BoundBox}_y^2 + \mathrm{BoundBox}_z^2}}{\mathrm{RelativeTol}} \tag{1}$$

DEFINITION 1. The criteria for H-Mesh and H-Brep edge convergence

Given:

- *A polyline:* $c_M\{\boldsymbol{V}, \boldsymbol{E}\}$
- *An analytic curve in the B-rep form:* $c_B(t)$

If c_M *and* c_B *satisfy*
$\delta_H(c_M, c_B) = \max\left(\overline{\delta_H}(c_M, c_B), \overline{\delta_H}(c_B, c_M)\right) < \tau$
where

- $\overline{\delta_H}(c_M, c_B) = \max_{a\in V_i} \min_{b\in c_B} d(a, b)$
  - $d(a, b)$ *is the Euclidean point distance.*
- $\overline{\delta_H}(c_B, c_M) = \max_{b\in c_B} \min_{a\in E_i} d(b, a)$
  - $d(a, b)$ *is the Euclidean point-to-line-segment distance.*

Then $\boldsymbol{c_M}$ *and* $\boldsymbol{c_B}$ *are coincident.*

4 Case Study: Visualization of Large-Scale Assembly Models Represented by the Mesh and Boundary Representation Combined Representation Scheme

For application rather than 3D design, the ability to efficiently visualize 3D large-scale models is essential for any lightweight representation format since this function is required in most collaborative applications such as reviewing, 3D technical report and catalog publishing, or model streaming via the Internet. Therefore, the visualization process of 3D CAD lightweight models is taken as a necessary case study. Since the proposed lightweight model is not a pre-tessellated model similar to facet-based formats such as JT, X3D, or U3D or a full B-rep scheme such as STEP or IGES, the visualization process of the mesh and B-rep combined representation model is slightly different. The processing pipeline is described in Sec. 4.1, and some experimental studies are presented in Sec. 4.2.

4.1 Visualization Pipeline. Figure 8 illustrates the visualization process of a model represented by the mesh and B-rep combined representation. In general, the triangulation process for visualization purpose with the hybrid model and the conventional B-rep model is similar. At the first step, H-Brep faces in the combined representation model will be triangularized independently. However, the acquired mesh model at this stage is "non-watertight" with gaps between adjacent faces, especially between an H-Mesh face and a faceted H-Brep face, thereby causing poor visualization quality. The mesh healing process is applied based on the application case and the user requirements.

For the mesh healing problem, two common approaches are usually applied: (1) incrementally meshing with constraints acquired from the adjacent faces' faceted data or (2) conducting a mesh healing process after the initial tessellation process. In this work, the non-watertight mesh issue is overcome by applying a mesh healing process owing to its suitability with the meshing pipeline.

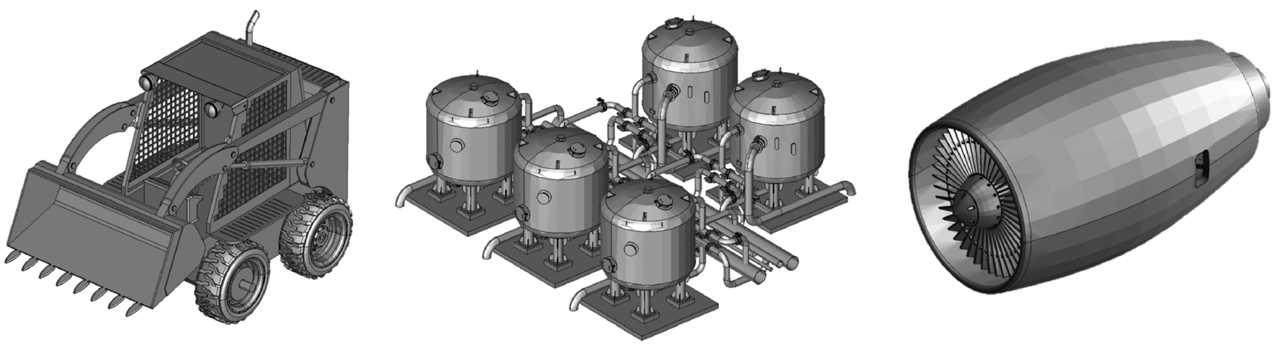
There are many approaches available for solving mesh healing problems classified into two categories: (1) mesh-based and (2) region-based [33]. While the mesh-based approaches tend to heal the mesh by moving the existing edges and vertices, the region-based approaches tend to fill in the gap between two mesh patches with new polygons. Ju concluded that the mesh-based approach is suitable for fixing small errors, whereas the region-based approach is mostly used for reconstructing poor quality models with holes or gaps [34].

Algorithm 2. Mesh healing algorithm for a pretessellated hybrid body

Given*: A hybrid body HB*$\{\boldsymbol{F}, \boldsymbol{E}, \boldsymbol{V}\}$ *has been pretessellated. Each shared edge* E_j *of two adjacent faces* F_m *and* F_n *is attached with faceted data and corresponding faces faceted data.*
For *each* E_j
Merge all faceted data that have been attached to the edge with the algorithm proposed in Ref. [33].
Update to the mesh of F_m *and* F_n *if the new vertex is added.*
End For
Return*: a Hybrid body with watertight faceted data.*

Table 2 The Input data characteristic and experiment result

	Test model 1 bulldozer	Test model 2 sand filter system	Test model 3 jet engine
Number of bodies	100	235	1130
Input SAT file size (kB)	56,050	63,776	108,331
Analytic surface count	3054	6045	7612
Analytic trimmed surface count	4558	4565	967
NURBS surface count	1720	0	3139



(*a*) (*b*) (*c*)

Fig. 9 Test model visualizations: (*a*) test model 1—the Bulldozer, (*b*) test model 2—the sand filter system, and (*c*) test model 3—the jet engine

Table 3 Experiment results

		Test model 1	Test model 2	Test model 3
Lightweight file size (kB)	All faceted model	2661	3078	3122
	Mesh and B-rep combined with duplicate shared edge	2092	2418	2044
	The proposed approach	1791	2240	1785
Loading time (s)	Without healing	2.370	6.543	4.661
	With healing	17.388	24.655	25.750

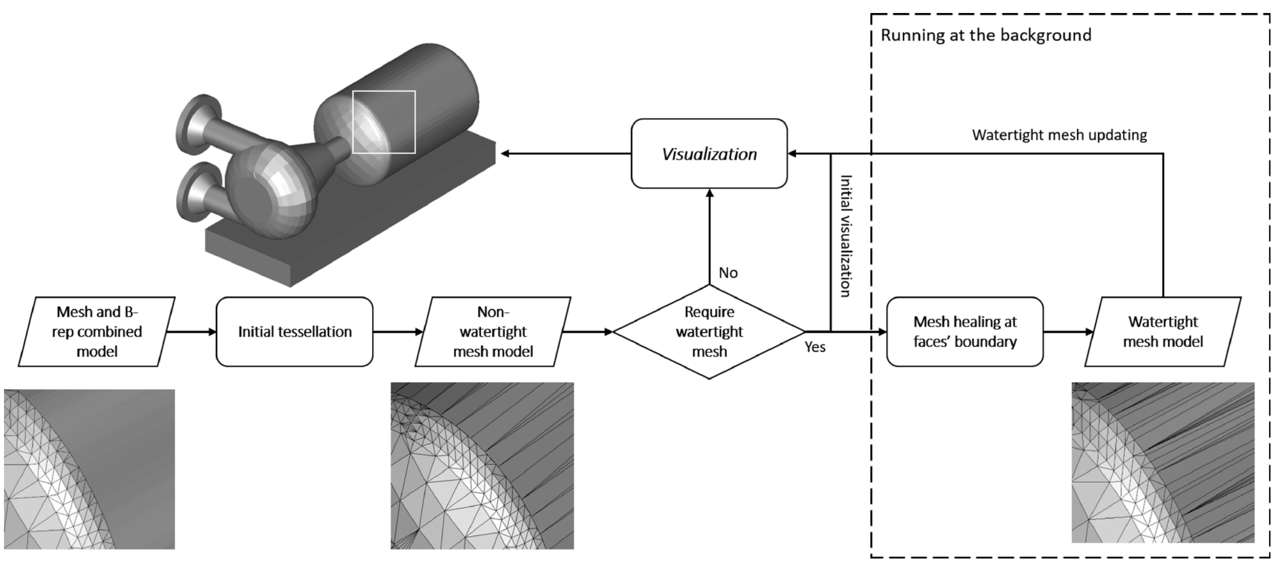


Fig. 10 Adjusted visualization pipeline

Because the input hybrid model satisfies the AG^0 criteria, the gap between adjacent faceted faces is relatively small enough for applying the mesh-based approach. Given the proposed data structure, the proposed healing process would be performed on every edge of the faceted model. In the case of two-manifold modeling, after the initial tessellation process, each edge would have two sets of faceted data corresponding to two adjacent faces, which could be considered as two polylines. If the two polylines are not exactly matching with each other, gaps between two adjacent faceted faces appear. Thus, the final goal of the healing process is to match all the vertices along those two polylines to remove all possible gaps. The proposed healing process includes two consecutive processes: (1) merging two polyline edges and (2) updating the mesh patches if new vertices are added. The healing algorithm was adopted from the algorithm proposed by Smith et al. [33], which was modified to suit the application case. Algorithm 1 describes the proposed mesh healing process.

Some assumptions are needed to ensure the stability of the process. First, there should be no edges with lengths smaller than the preset threshold. For ensuring the assumption, a preprocessing to analyze the faceted model and remove small entities is necessary. Second, it should be ensured that two faceting data sets of the given edge satisfy the AG^0 criteria; otherwise, the algorithm might fail because of an infinite loop.

4.2 Experimental Study. A validation process was performed on three publicly available 3D CAD models in SAT format acquired from Ref. [35]. The proposed approach was implemented using the C++ programming language with multiple third-party libraries, namely the ACIS Toolkit with the HOOPS Visualize Library [26,36], the Open CASCADE Toolkit [27], and the Geometric Tool Library [37]. Table 2 reports the test models' characteristics and Fig. 9 shows their visualization. The experiment was conducted on a laboratory computer with an AMD®Ryzen Threadripper central processing unit at 3.50 GHz, 12 cores, 20 threads, and 32.0 GB RAM. The mesh resolution was set up with a normal tolerance of 40 deg and a surface tolerance of 0.004 times the length of the diagonal of the body's bounding box.

Table 3 contains the experimental results of the three tested models. The output file size and loading time for the visualizations were adopted as the assessment criteria. For data reduction ability benchmarking, the proposed lightweight representation was compared with two other lightweight models: the all-faceted model and the mesh representation and B-rep combined with duplicate H-Mesh and H-Brep faces' shared edges model at the same level of mesh resolution. In Table 3, it can be seen that the proposed approach attained better data storage reduction than the other two, averaging 34% and 11%, respectively. This is because unlike the all-faceted model, most of the elementary geometric entities are kept in analytic form under B-rep (except the trimmed surfaces), which requires less data storage than mesh representation. Furthermore, unlike the mesh and B-rep combined representation scheme with duplicate H-Mesh and H-Brep faces' shared edges approach, the proposed approach eliminates the duplicate issue at the interface between the H-Mesh and H-B-rep faces by the concept of the hybrid edge, thus reducing the data storage requirement of the lightweight 3D CAD model.

However, the loading times for visualizing the 3D test models showed some notable issues. From the experimental result, the required loading time with the healing process was far longer than without it. This issue can be explained by the fact that the mesh healing process is a computationally time-consuming process. Although the loading time without healing is acceptable, that with healing seems to be unsuitable for the real-time processing required for visualization or 3D streaming over the Internet. To overcome the problem of healing time, we propose that the mesh healing process is performed in the background of the 3D viewer and the non-watertight mesh model is used for the initial visualization. Consequently, the watertight mesh is updated in the scene after the healing process has completed. Thus, in practice, the workflow initially described in Fig. 8 should be adjusted as illustrated in Fig. 10. By doing so, the proposed visualization pipeline becomes adequately practical for industrial use.

5 Conclusion

In this article, a mesh and B-rep combined representation approach was proposed for a 3D CAD lightweight representation in collaborative product development. The conversion from B-rep data was developed with proposed geometry classification and boundary convergence criteria. A case study of the visualization process of large-scale assembly models represented by the proposed lightweight representation was also conducted. The validation process with three test datasets proved the usefulness and practicality of the proposed approach.

From the experimental results, the proposed lightweight representation scheme showed some advantages over the conventional all-faceted representation regarding lightweight file size while the loading time for visualization was kept at a reasonable level with an adjusted visualization pipeline. Moreover, the potential of the proposed method is not limited to a lightweight file size and good loading speed. For data exchange in collaborative product development with applications other than 3D CAD, representing complex geometries in faceted representation is more general as any curves and surfaces can be approximated by a set of linear and planar geometries. Furthermore, complex geometric surfaces characterized by faceted representation are more robust and efficient for fast visualization because they do not require a reconstruction process for rendering data generation. This characteristic demonstrates the flexibility in the 3D modeling ability of the proposed mesh and B-rep combined representation, making it suitable for CAD data exchange in collaborative product development. Overcoming the limitation of duplicate geometric entities from the existing approaches in combining mesh and B-rep into one representation scheme, the results of this work could provide the industry with another technique for lightweight representation of 3D CAD data for sharing and transferring in engineering collaboration, especially when combined with other successful techniques (e.g., domain-specific compression and reference–instance mechanisms).

In addition, the conversion back to B-rep from the proposed lightweight representation if needed is straightforward since the faces have already been divided, and so only the surface fitting process is required, thereby making this process straightforward.

In the future, expanding the work to encompass other applications such as process planning, dimension verification, and interference checking based on the proposed format will be conducted as well as the development of quantifiable classification criteria (e.g., via curvature analysis) for more optimal classification results. Moreover, further effort will be made to improve the visualization process to meet industrial requirements.

Acknowledgment

This work was in part supported by the Technology Innovation Program (Development of a lightweight 3D model-based digital collaboration system to support the establishment of efficient collaboration environment in engineering project) funded by the Ministry of Trade, Industry & Energy (MOTIE, Korea) and the Basic Science Research Program through the National Research Foundation of Korea (NRF), funded by the Ministry of Education.

Funding Data

- Korea Evaluation Institute of Industrial Technology (10080662).
- National Research Foundation of Korea (NRF-2015R1D1A1A0106).

References

[1] Kim, I., Cho, G., Hwang, J., Li, J., and Han, S., 2010, *Visualization of Neutral Model of Ship Pipe System Using X3D*, Springer-Verlag, Berlin.

[2] Song, I.-H., and Chung, S.-C., 2009, "Data Format and Browser of Lightweight CAD Files for Dimensional Verification Over the Internet," J. Mech. Sci. Technol., **23**(5), pp. 1278–1288.

[3] Kim, E. H., Hwang, J., Hahm, G.-J., and Lee, J. H., 2015, "3D CAD Model Visualization on a Website Using the X3D Standard," Comput. Ind., **70**, pp. 116–126.

[4] Liu, W., Zhou, X., Zhang, X., and Niu, Q., 2015, "Three-Dimensional (3D) CAD Model Lightweight Scheme for Large-Scale Assembly and Simulation," Int. J. Comput. Integr. Manuf., **28**(5), pp. 520–533.

[5] Fuh, J. Y. H., and Li, W. D., 2005, "Advances in Collaborative CAD: The-State-of-the Art," Comput. Aided Des., **37**(5), pp. 571–581.
[6] Chiyokura, H., 1988, *Solid Modeling With Designbase: Theory and Implementation*, Addison-Wesley, Boston, MA.
[7] Botsch, M., Kobbelt, L., Pauly, M., Alliez, P., and Levy, B., 2010, *Polygon Mesh Processing*, CRC Press, Boca Raton, FL.
[8] Piegl, L., and Tiller, W., 1997, *The NURBS Book*, 2nd ed., Springer-Verlag, Berlin.
[9] Anand, V. B., 1993, *Computer Graphics and Geometric Modeling for Engineers*, Wiley, Hoboken, NJ.
[10] Ding, L., Davies, D., and McMahon, C. A., 2009, "The Integration of Lightweight Representation and Annotation for Collaborative Design Representation," Res. Eng. Des., **20**(3), pp. 185–200.
[11] Web-3D Consortium, 2018, "X3D," Web-3D Consortium, Mountain View, CA, accessed Oct. 29, 2018, http://www.web3d.org/x3d/what-x3d
[12] ECMA International Standards, 2018, "Universal 3D File Format," ECMA International Standards, Geneva, Switzerland, accessed Oct. 29, 2018, http://www.ecma-international.org/publications/standards/Ecma-363.htm
[13] Geng, J., Tian, X., Bai, M., Jia, X., and Liu, X., 2014, "A Design Method for Three-Dimensional Maintenance, Repair and Overhaul Job Card of Complex Products," Comput. Ind., **65**(1), pp. 200–209.
[14] Siemens, 2018, "JT Open," Siemens PLM Software, Plano, TX, accessed Oct. 29, 2018, https://www.plm.automation. siemens.com/en/products/open/jtopen/
[15] Dassault Systemes, 2018, "3D XML," Dassault Systemes, Paris, France, accessed Oct. 29, 2018, http://www.3ds.com/3dxml
[16] Song, I.-H., and Chung, S.-C., 2008, "Integrated CAD/CAE/CAI Verification System for Web-Based PDM," Comput. Aided Des. Appl., **5**(5), pp. 676–685.
[17] Kwon, S., Kim, B. C., Mun, D., and Han, S., 2015, "Simplification of Feature-Based 3D CAD Assembly Data of Ship and Offshore Equipment Using Quantitative Evaluation Metrics," Comput. Aided Des., **59**, pp. 140–154.
[18] Kim, B. C., and Mun, D., 2014, "Feature-Based Simplification of Boundary Representation Models Using Sequential Iterative Volume Decomposition," Comput. Graph., **38**, pp. 97–107.
[19] Gao, S., Zhao, W., Lin, H., Yang, F., and Chen, X., 2010, "Feature Suppression Based CAD Mesh Model Simplification," Comput. Aided Des., **42**(12), pp. 1178–1188.
[20] Lafarge, F., Keriven, R., and Brédif, M., 2010, "Insertion of 3-D-Primitives in Mesh-Based Representations: Towards Compact Models Preserving the Details," IEEE Trans. Image Process., **19**(7), pp. 1683–1694.
[21] Lafarge, F., Keriven, R., and Brédif, M., 2009, "Combining Meshes and Geometric Primitives for Accurate and Semantic Modeling," British Machine Vision Conference (BMVC), London, Sept. 7–10, pp. 1–11.
[22] Ferrari, G., 2017, "A Numerical Proposal of an Extended Solid Modeling System," Ph.D. thesis, University of Bologna, Bologna, Italy.
[23] Besl, P., 1998, "Hybrid Modeling for Manufacturing Using NURBS, Polygons, and 3D Scanner Data," IEEE International Symposium on Circuits and Systems (ISCAS'98), Monterey, CA, May 31–June 3, pp. 484–487.
[24] W3C, 2018, "Extensible Markup Language (XML)," W3C, MA, accessed Oct. 29, 2018, https://www.w3.org/XML/
[25] Zlib, 2018, "Zlib," Zlib, accessed Oct. 29, 2018, https://zlib.net/
[26] Dassault Systemes, 2018, "The ACIS Toolkit," Opencascade, France, accessed Oct. 29, 2018, https://doc.spatial.com/
[27] Opencascade, 2018, "Open Cascade Technology," Spatial Corp., Dassault Systemes, CO, accessed Oct. 29, 2018, https://www.opencascade.com
[28] Piegl, L. A., and Tiller, W., 1998, "Geometry-Based Triangulation of Trimmed NURBS Surfaces," Comput. Aided Des, **30**(1), pp. 11–18.
[29] Shimada, K., and Gossard, D. C., 1998, "Automatic Triangular Mesh Generation of Trimmed Parametric Surfaces for Finite Element Analysis," Comput. Aided Geom. Des., **15**(3), pp. 199–222.
[30] Ristic, M., Brujic, G., and Handayani, S., 2000, "CAD-Based Triangulation of Unordered Data Using Trimmed NURBS Models," J. Mater. Process. Technol., **107**(1–3), pp. 60–70.
[31] Chae, S.-W., and Kwon, K.-Y., 2001, "Quadrilateral Mesh Generation on Trimmed NURBS Surfaces," J. Mech. Sci. Technol., **15**(5), pp. 592–601.
[32] Marur, S. R., 2005, "On the Meshing of Trimmed 3D Surfaces," Adv. Eng. Software, **36**(5), pp. 338–345.
[33] Smith, B. M., Tautges, T. J., and Wilson, P. P. H., 2010, "Sealing Faceted Surfaces to Achieve Watertight CAD Models," 19th International Meshing Roundtable, Chattanooga, TN, Oct. 3–6, pp. 177–194.
[34] Ju, T., 2009, "Fixing Geometric Errors on Polygonal Models: A Survey," J. Comput. Sci. Technol., **24**(1), pp. 19–29.
[35] Stratasys Solutions, 2018, "GrabCAD," Stratasys Solutions, Cambridge, MA, accessed Oct. 29, 2018, https://grabcad.com/
[36] Techsoft 3D, 2018, "The HOOPS Visualize Library," Techsoft 3D, Bend, OR, accessed Oct. 29, 2018, https://www.techsoft3d.com/products/hoops/visualize/
[37] Geometric Tools, 2018, "Geometric Tools," David Eberly, accessed Oct. 29, 2018, https://www.geometrictools.com/