Advances in Engineering Software 81 (2015) 7–16

Contents lists available at ScienceDirect

Advances in Engineering Software

journal homepage: www.elsevier.com/locate/advengsoft

Development of a gap searching program for automotive body assemblies based on a decomposition model representation

Jaedeuk Yun [a], Sunghoon Lee [b], Yoongho Jung [a,*]

[a] School of Mechanical Engineering, Pusan National University, Geumjeong-gu, Busan 609-735, South Korea
[b] Applied Technology Department, R&D Center, MIDAS Information Technology Co., Bundang-gu, Seongnam, Gyeonggi-do 463-400, South Korea

ARTICLE INFO

Article history:
Received 4 July 2014
Received in revised form 13 September 2014
Accepted 12 October 2014
Available online 25 November 2014

Keywords:
Mismatch
Octree
Voxel
Collaborative work
Virtual assembly
Automotive

ABSTRACT

Automobiles, aircraft, and ships require tremendously many parts to be assembled. For developing such large assemblies, most companies accelerate the design process by having many design engineers in different functional or sectional design groups working concurrently. However, interferences and gaps can be found when the parts and sub-assemblies of different design groups are to be assembled. These error cause design changes and additional repair processes, resulting in an unexpected increase in costs and time delays. While the interference problem has been resolved by digital mockup and concurrent engineering methodology, many cases of the gap problem in the automotive industry have been covered by temporary treatments when the gaps are small enough to be filled with sealants. This kind of fast fix can cause leakage into the engine chamber and passenger cabin when the gap size is too big for filling or when the sealant gets old, which can turn fatal. With this research, we have developed a program to automatically find gaps between the parts of an assembly so that design engineers can correct their designs before the manufacturing stage begins. By using the method of decomposition model representation, the program can visualize gaps between complex car body parts as well as estimate their volumetric information. It can also automatically define the boundary between a gap and the exterior space. Although we have reviewed the benefits of the program by applying it to car development, it can also be applied to aircraft and ship designs comprising several parts.

1. Introduction

Manufacturing and assembling complex machinery such as automobiles, aircraft, or ships require long development periods and huge number of parts to be assembled. For such large assemblies, most design and manufacturing companies shorten the development period by employing many design engineers working concurrently. Concurrent design [1] by functional or sectional design groups may reduce the development time, but it may still cause problems of interference, collision, and gaps between parts to be assembled, which are designed by different designers and different design groups. Most designers tend to concentrate on the parts for which they are responsible, instead of checking the assemblability of their own work with the parts designed by other engineers; this methodology often causes errors. These design errors delay development time and increase manufacturing costs caused by modifications to the original design.

Regarding problems of interference or collision between parts, engineers can verify their design with real mock-up or digital mock-up (DMU) systems that can assemble parts virtually before the manufacturing stage [2,3]. On the other hand, gaps between the parts may cause other severe problems in automobiles. Most parts of automobiles are made of thin plates to reduce the vehicle weight, and there are many empty spaces between thin parts. If there are gaps between the parts of a car facing the exterior, rainwater and water from puddles on roadways can enter the engine chamber and the cabin directly, or flow into unwanted spaces through gaps between the body parts. This situation may induce corrosion of interior parts, or fatal problems of electric shock and malfunction of the electrical system. While some of these gaps between parts are intended to increase the stiffness of the car body structure, most unwanted gaps are caused by design errors when different parts of the assembly are designed by many different engineers. In order to resolve this problem, designers check cross-sectional drawings of the assembly. However, this process is impractical as it takes too much time for design engineers to check the shape of a gap in a complicated three-dimensional assembly with two-dimensional sectional drawings.

* Corresponding author. Tel.: +82 51 510 2469; fax: +82 51 512 1722.
E-mail addresses: yjdeuk@pusan.ac.kr (J. Yun), lsh1006@midasit.com (S. Lee), yhj@pusan.ac.kr (Y. Jung).

http://dx.doi.org/10.1016/j.advengsoft.2014.10.003
0965-9978/

Various studies have been done on resolving the problems of interference and collision between the parts of an assembly. Song and Chung [4] suggested a DMU system using XML for real-time interference checking and design verification of several parts in a virtual environment based on a product data management (PDM) system. Song et al. [5] suggested a web-based interference verification system for mold design processes. Shi et al. [6] suggested a collisions evaluation method in virtual environment for collaborative assembly. Chen [7] also developed a new system for virtual assembly of parts and real time collision detection between parts. However, all of the above literature focuses on interference and collision between parts and not on gaps between the parts of a complicated assembly.

In this paper, we propose a method of automatically searching for gaps between parts of complex shape with free surfaces based on the mixed octree–voxel decomposition method. With this method, designers can check for gaps between parts during the design stage in order to prevent errors due to gaps before the manufacturing and assembly stages. The application of the proposed method is not limited to the automobile industry but can be extended to the aerospace and ship building industries as well.

2. Overview of the gap visualization algorithm

This research suggests a method to locate and visualize gaps within assemblies with parts of thin plates. Most assemblies, such as automobiles, aircraft, and ships are made of thin plates with free surfaces; therefore, the target shape of the parts, with which this research is concerned, is the free-form shape. Since the gaps between these parts include free-form boundary surfaces, we represent the given assembly using decomposition models; this method is efficient in calculation time and is not limited by the complexity of the given parts. Thereafter, we use Boolean operations to subtract the decomposition models from a fixed space, and thereby identify any gaps in the design. The most widely known methods used to express decomposition models are voxels with uniform elements [8,9] and octrees with elements of various sizes [10–13]. In this paper, we use a mixed octree–voxel method [14].

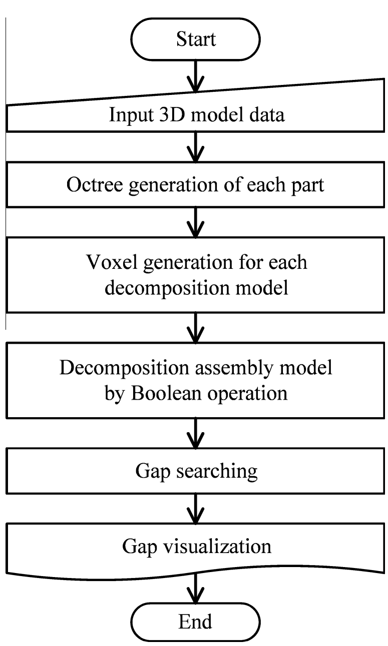

Fig. 1. Gap visualization algorithm.

The overall process of the proposed algorithm is shown in Fig. 1, which starts by inputting three-dimensional models in the stereolithographic (STL) data format. The STL data format is widely adopted as the standard format for computer-aided design (CAD) model data. Therefore, the program can use models from various CAD systems. Another advantage of the STL format is its small data size for representing complex parts. Since the objective of this research is to develop a program that can be applied to large assemblies having several parts, large data sizes for each model could cause severe execution problems.

Next, after representing each part as octrees of given resolutions at the "octree generation of each part" step, the algorithm uses the classification information of the octants to divide all "full" and "empty" octants into voxels of the same size as the minimum octant; this is done at the "voxel generation for each decomposition model" step. Furthermore, after mapping the decomposition model of each part to its corresponding position in the assembly, the program generates a decomposition assembly model using Boolean operations on the decomposition models of all of the parts. Finally, it searches for voxels in gaps to be visualized at the "gap searching" and "gap visualization" steps. A more detailed explanation is given in the following sections.

3. Decomposition assembly model

Gaps are defined in this research as "interior empty spaces between parts." When the geometry of the parts is complicated, the geometry of the gap is complicated as well. For the automatic gap searching and visualization processes, which are not limited by the complexity of the object, we express the containing volume of the given assembly as a decomposition model. When the decomposition model of the assembly is generated, the empty elements of the decomposition model can be classified as either "exterior" or "gap" elements. In order to generate a decomposition model of the assembly, the proposed method first generates a decomposition model of each part of the assembly.

3.1. Octree generation

The most widely known methods for expressing decomposition models are voxel and octree representations. Voxels represent a three dimensional object with uniform sized cells, whereas octrees represent elements of various sizes. The voxel representation method has advantages, such as the fact that it is simple to represent a given object and easy to find neighbors of a given element. However, small voxels, corresponding to higher resolutions, require more memory for model representation and higher calculation times to classify each voxel as interior or exterior to the given object.

On the other hand, octree representations starts by generating an initial cube, called a "root," which is the minimal containing box of a given object. Thereafter, it divides the root into eight children, called "octants," of which each octant is classified as "full," "empty," or "partial" according to the spatial relation of the octant with respect to the given object, as shown in Fig. 2.

If an octant is completely inside the object, it is classified as a "full" octant, while if it is completely outside the object, it is classified as "empty". When an octant is neither "full" nor "empty," it is classified as "partial," which means that it contains the boundary of a given object and it is to be divided into eight further octants. This process continues recursively until the input level of the tree structure is reached. The octree representation method recursively divides only "partial" elements that contain the boundary of the given object into elements of desired resolution, which results in