Assurance of Product Data

Chun-Fong You[1] and Tung-Hua Chan[2]

[1]National Taiwan University, you@ntu.edu.tw
[2]National Taiwan University, r93522603@ntu.edu.tw

ABSTRACT

Each CAD system has its own data structure for representing topological and geometrical entities. Transferring data between two systems may require conversion between models. This investigation describes the conversion rules for different CAD systems, as well as checking criteria and repairing sequences for recovering the fault model. The checking result is also stored at the database for further reference. Finally, a prototype system is implemented to verify the proposed algorithm, which is demonstrated to attain high product data quality.

Keywords: product data quality, conversion, repair, CAD model.

1. INTRODUCTION

The flow of product information among the different enterprises is significant problem during the product life cycle. Many CAD/CAM systems are applied in the design and manufacturing departments. Each CAD system has a different architecture, not only in the representation of topology but also in the representation of mathematics. Therefore, the development of the product success and the integration of some products are limited.

The file formats for exchanging the standard CAD systems include IGES and STEP (**ST**andard for the **E**xchange **P**roduct Model Data). IGES is mainly used to transfer geometric and drawing data, while STEP is applied to transfer of shape, non-shape, design and manufacturing data. Shape data comprises both geometrical and topological information. Figure 1 illustrates the topological and geometric entities of STEP part 42 [1].

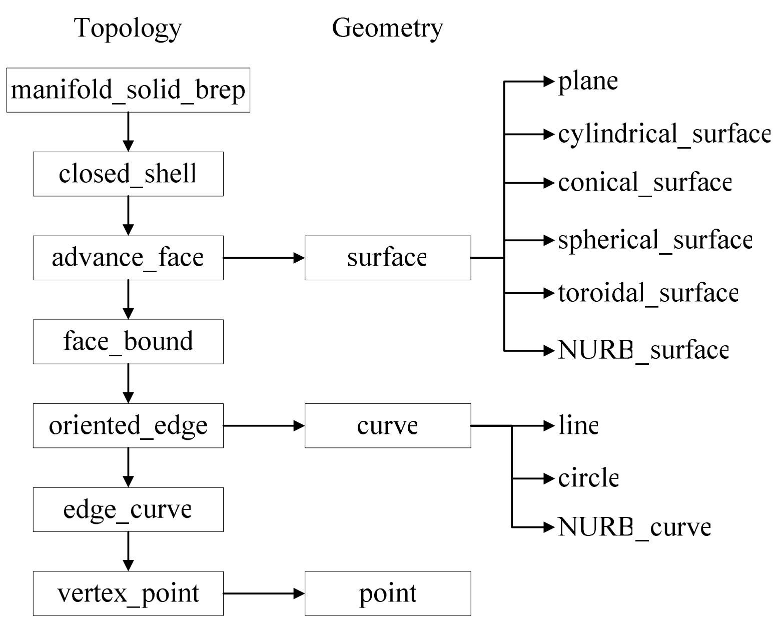
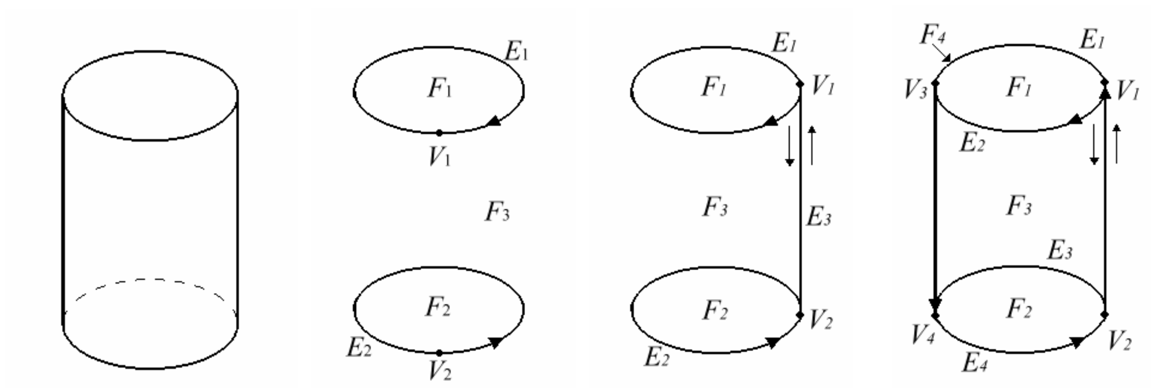


Fig. 1. Topological and geometric entities proposed at STEP part 42.

Different CAD systems represent topology and geometry differently. For example, different CAD vendor systems represent solid cylinders in different ways, as shown in Figure 2 and Table 1. A solid cylinder may have three faces or four faces, depends on the representation of closed and open faces. The representation of the geometric circle may be described as a circle or B-spline curve.



Solid cylinder   Model A   Model B   Model C

Fig. 2. Representation of solid cylinder.

Tab. 1. Representation comparisons of the solid cylinder.

	Model A	*Model B*	*Model C*
CAD System	UniGraph	IDEAS	CATIA, Pro/Engineer
Number of faces	3	3	4
Number of edges	2	3	6
Number of vertices	2	2	4
Type of curve	circle	B_spline_curve	circle
Type of surface	cylindrical_surface	B_spline_surface	cylindrical_surface

The data inconsistency is not only encountered on the representation of the topological entities, but also on the representation of geometric entities. Figure 3 illustrates an example of two solid cylinders intersecting perpendicularly. The curve may exist at the intersection of two cylindrical surfaces. The radius of a larger cylindrical surface is 100mm, and the axis of the cylindrical surface is on the z-axis, while the radius of smaller cylindrical surface is 50mm, and the axis of cylindrical surface is on the y-axis. Table 2 presents the representation of the intersection curve from different CAD system.

The performance of a B-spline curve may be affected by the degree, the number of control points and the number of knot vectors on the curve. The blending function of the B-spline curve can be calculated using the Cox and de Boor [2] iterative equations.

$$N_{1,i} = \begin{cases} 1 & \text{for} \quad x_i \leq x \leq x_{i+1} \\ 0 & \text{otherwise} \end{cases}$$

and

$$N_{k,i}(x) = \frac{(x - x_i)N_{k-1,i}(x)}{(x_{i+k-1} - x_i)} + \frac{(x_{i+k} - x)N_{k-1,i+1}(x)}{(x_{i+k} - x_{i+1})}$$

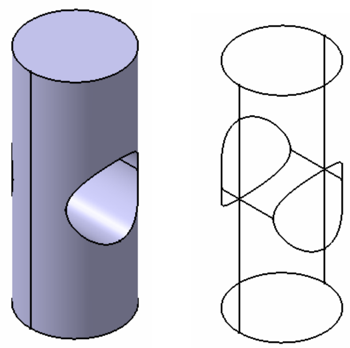


Fig. 3. Cylinder is subtracted by other vertical cylinder.

Tab. 2. Comparison of the intersection curve.

	ACIS	*UniGraph*	*CATIA*
Order of B-spline curve	4	4	6
Numbers of Control Point	130	80	21
Numbers of knot vectors	134	84	27
Knot vector	0.0, 8.20, 16.40,…,508.62, 516.82	-0.03125, 0, 0.03125, 0.0625,0.09375,…, 0.96875, 1.0, 1.03125	0.0, 17.15, 46.83,…, 154.96, 188.34
Knot multiplicity	4,2,2,…,2, 4	2,2,2,2,2,…,2,2,2	6,3,3,3,…,3,3,6

Conversions between CAD system models may lead to tedious repair operations. If a model does not satisfy functional requirements, then engineers may spend significant effort improving the receiving model, especially in the automobile industry.

Recently the Japan Automotive Manufacturers Association (JAMA) [3] and Automotive Industry Action Group (AIAG) [4] recently presented some checking criteria of the CAD model for the assurance of the product data. The Automotive Industry Action Group (AIAG) has defined product data quality in the following items, "Quality Product Model Data is constructed accurately and completely represent the geometric model (mathematical data), and accurately and completely represent all additional information in a way that can be shared and used by multiple users and managed with a minimum of effort". The publication strategy of definition of product data quality is close resembles to that of the STEP. Some repairing processes must still be implemented.

Some commercial packages [5]~[8] are available on the market for handling the checking and repairing capabilities of the CAD models. The PDQ-NP [7] package is binding on the IDEAS CAD system, and the Q-checker [8] is binding on the CATIA CAD system. Some investigations [9]-[11] have focused on the repairing articles. Owing to the diversity of the data structure adopted by the CAD system, some automatic healing processes of CAD models must be further addressed.

The STEP file format was utilized herein in this study to investigate the recovering techniques among the different CAD systems. Therefore, the mapping tables between each entity of CAD system with proposed system, and the entities that are not suitable for this mechanism are further explored and automatically transformed to suitable entities. Some criteria of conversion and repair algorithm are presented as follows.

2. CONVERSION ISSUE

If the representation of solid model has different topological and geometric data structure, then a neutral data structure that contains the complete solutions of all representations for all CAD vendor systems can be constructed. The following guidelines should be adopted to maintain date neutrally:

(a) Every curve should be represented in terms of regular curve, such as straight line, circle, rather than in terms of a B-spline curve.
(b) A surface should be represented in terms of regular surface, such as plane, cylindrical surface, conical surface, rather than in terms of a B-spline surface.
(c) A B-spline curve should have a low degree.
(d) Edges should be open.
(e) Faces should be open.

2.1 Geometric inconsistency

The B-spline curve generated from the UniGraph CAD system, as shown in Table 2, appears to be a unique case. The conversion of this curve can be achieved using the following two methods:

(a) Coordinates at knot vectors 0.0, 0.03125, 0.0625, 0.09375, …, 0.96875, 1.0 are calculated, and then interpolated as a B-spline curve with degree 3.
(b) Multiple knots are inserted at knots 0.0, and 1.0, and the B-spline curve is then trimmed from 0.0 to 1.0. Finally, the knot is removed at 0.0, 0.03125, 0.0625, 0.09375,…, 0.96875, 1.0. The knot multiplicity of final curve is [4,1,1,…,1,4].

Method (b) is more accurate than method (a).

2.2 Topological inconsistency

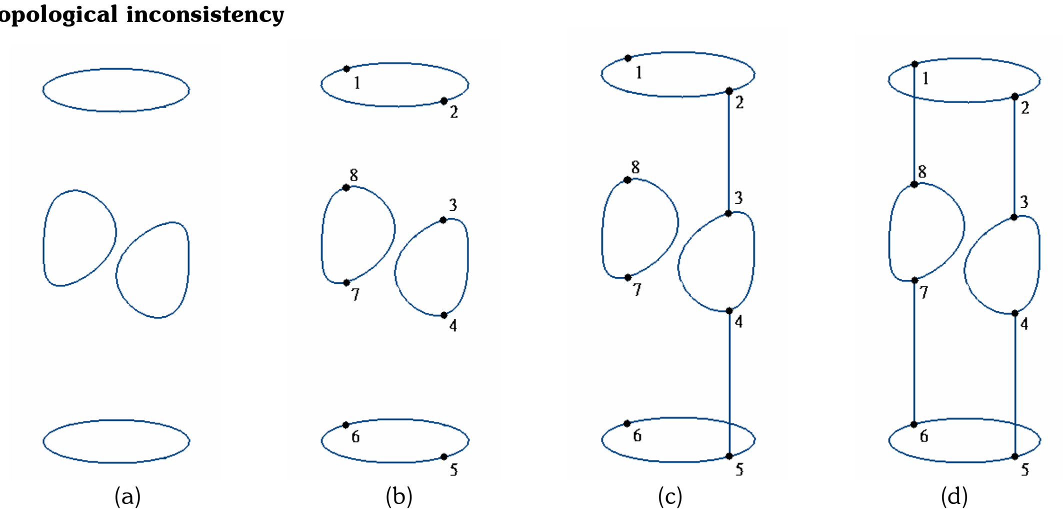


Fig. 4. Split operation for closed face.

Figure 2 can be adopted for this consideration and taken a larger cylindrical surface for illustration as shown in Figure 4. The cutting plane is determined from the top and bottom circles and passed through points 1, 2, 5. The split operation is as follows:

(a) Calculate the intersecting points between the cutting plane and all edges. These intersection points are denoted as vertices 1-8 (Figure 4(b).
(b) Split the edges from these intersection points into open edges using the Euler operator MEV.
(c) Find intersection curves of the cutting plane and larger cylindrical surface, and denote these intersection curves as $\overline{25}$ or $\overline{16}$.
(d) Split the line $\overline{25}$ from their intersection points, causing the line $\overline{25}$ to split into three lines $\overline{23}, \overline{34}, \overline{45}$.

(e) Classify each line in terms of where it lies on the cylindrical surface. Only lines $\overline{23}$ and $\overline{45}$ lie on the cylindrical surface.

(f) Use Euler operators MEKH to fulfill the inserting operation for edges $\overline{23}$ and $\overline{45}$ (Figure 4(c)).

(g) Apply the same procedures to line $\overline{16}$, performing only the last step using Euler operator MEF. A new face corresponding to a larger cylindrical surface is built (Figure 4(d)).

3. REPAIR ISSUES

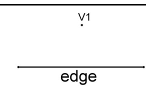
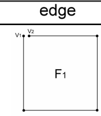
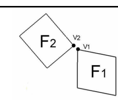
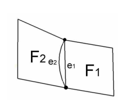
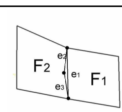
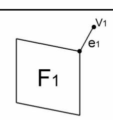
Most solid modeling systems are based on the B-rep representation, with the body, shell, face, loop, edge, and vertex as the data entities. Strict rules are also followed to maintain its validity, integrity. The following criteria must be met in two-manifold objects:

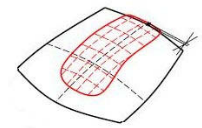
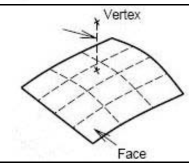
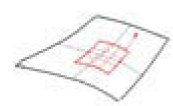
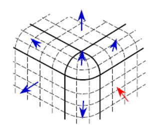
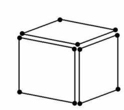
(a) Every edge belongs to exactly two faces.
(b) Every vertex is surrounded by a single cycle of edges and faces.
(c) Faces may not intersect each other except at common edges or vertices.

Problems may occur as soon as one CAD system attempts to import entities from another CAD system. Some fault problems such as the tiny edges and silver faces cannot be visibly detected and should be recovery using proposed algorithms.

A valid model can be reconstructed using a bottom-up approach. First, a vertex-vertex gap and duplicated vertices are checked and repaired. The duplicated edges and T-junction edges are then checked after the vertex connections are assured. Unnecessary data entities are eliminated. Finally, a face orientation is checked to assure the correctness of face orientation. Table 3 presents the sequences of the repair items.

Tab. 3. Sequence of repair items.

Sequence	*Check items*		*Drawing*	*Illustration*
1	Large vertex-edge gap			Gap between a vertex and the corresponding edge is greater than 0.01 mm.
2	Merge vertices	Vertex-vertex gap on face		Gap between the ending points of adjacent edges is greater than specified value.
3		Duplicate vertices		
4	Merge edges	Duplicate edges		Two or more edges are coincident, to within a specified tolerance. Edges must have the same end-points before they can be co-incident.
5		T-junction edge		The end-point of one edge intersects another edge.
6	Deleted entities	Unused edges Unused vertices Tiny edge Tiny face		The edge is not used by any face. Area of the face is too small.

7	Vertex and edge on face	Large vertex face gap		Distance between a vertex and its corresponding face is greater than a given value.
8		Large edge face gap		Distance between an edge and the surface is above the given accuracy.
9	Face orientation	Inconsistent face on surface		Direction of face normal is inconsistent with surface.
10		Inconsistent face in shell		Adjacent faces with opposite normal along their common boundary.
11	Build body			

4. IMPLEMENTATION

A prototype based on the JAMA and AIAG checking criteria was implemented to verify the algorithm proposed in the previous sections. This prototype was implemented in the SpringSolid system using topological and geometric entities, and is compatible with the STEP Part42 standards. The SpringSolid system was developed by the Solid Model Lab. at National Taiwan University. The input file format of the prototype system is IGES, and STEP, while the output file format is STEP.

Figure 5 illustrates the proposed mechanism. Since the model file is loaded by a CAD system, it can establish the checking items and tolerances for each criterion. The information of checking item is first transmitted to the server for calculation, and the result from server is then send back to the client for visualization. Operations for are either manual or automatic. Automatic operation is controlled by the kernel of the proposed algorithms, while manual operation is managed by users and can be assisted by the knowledge-base within the system. After finishing the checking and repairing procedure, users can export the necessary information into database to build the knowledge base and correct the rules of product data quality. The key issue for the knowledge base point is to arrange reasonable sequences and determine the likely problems in advance. The following approaches can be applied to establish the rule:

(a) Confirm the CAD vendor of the model
Identify the CAD vendor of the model and search the knowledge base to identify the most frequently encountered errors of the CAD vendor system. This method can provide the early checking and repair processes, good rendering appearances in the graphic user interface before the uses take any action.

(b) Batch mode operation
The checking and repairing procedures can be prepared in batch mode after creating the rule described as above. One model file or multiple model files can be sequentially computed. The checking result is stored in the database, and the user can then search the data base for reference.

Some test cases were employed to demonstrate the characteristics of the proposed algorithms. The test cases are described as follows:

(a) Tiny face

Figure 6 illustrates a tiny face [4]. A tiny face is first deleted (Figure 6(b)). The edge-edge gap on the three faces is then detected. To remedy the gap, the intersection point of two edges is found on the surface parametric (U, V) domain (Figure 6(c)), and projected to the global coordinate system. Finally, some duplicated edges and vertices are merged (Figure 6(d)).

(b) Tiny edge
Figure 7 shows some tiny edges. All tiny edges are removed based on the tiny edge criterion (Figure 7(b)). The vertices are preserved to reconstruct a B-spline curve (Figure 7(c)). Reconstruct a B-spline curve is tedious efforts and error-prone. Above approach can be fulfilled using merge edges and then remove small tiny curve segments.

(c) Edge orientation
The starting and ending point of the edges are checked and the edge sequences are also identified as shown in Figure 8.

(d) A model is used for the demonstration of the healing assistant in the CATIA CAD system and shown in Figure 9. The file size of model is about 2.5 MB, and total number of faces is 188. Figure 10 illustrates the checking report after the condition of two-manifold model is satisfied using the proposed algorithm. Figure 11 shows the rendering of final body after the checking and repairing procedures has been done. The most serious problems in this case are vertex-vertex gap and duplicated edges.

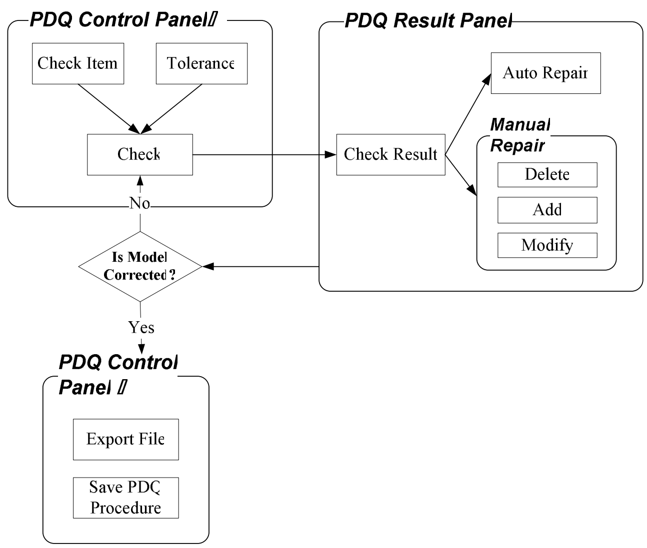
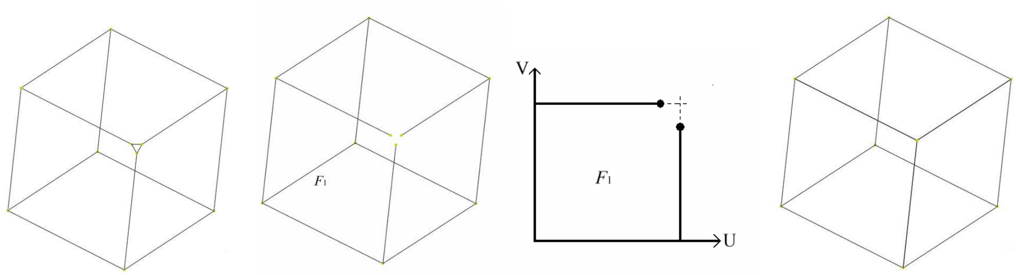
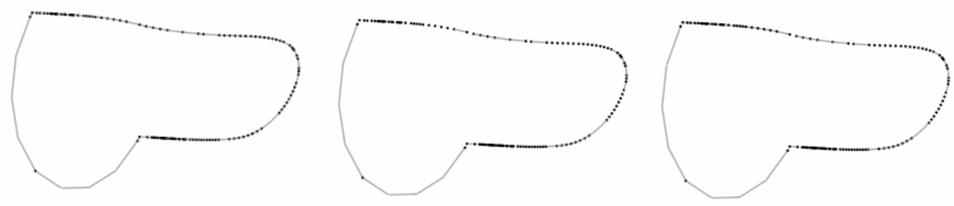


Fig. 5. Proposed mechanism.



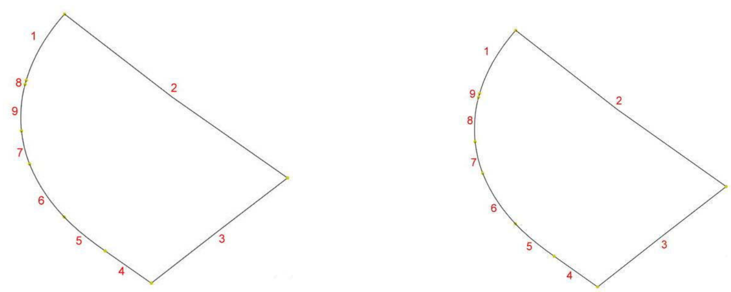
(a) (b) (c) (d)

Fig. 6. Tiny Face.



(a) (b) (c)

Fig. 7. Tiny Edge.



(a) (b)

(b)

Fig. 8. Edge orientation.

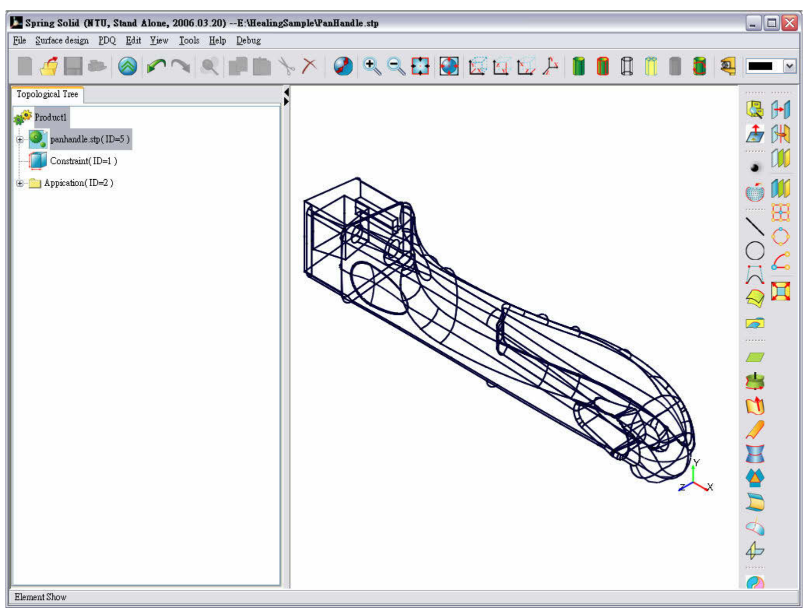
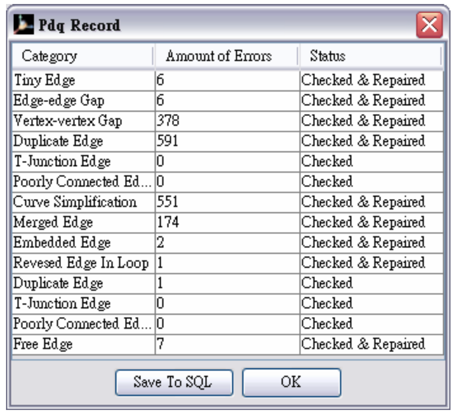


Fig. 9. Test example.



Pdq Record

Category	Amount of Errors	Status
Tiny Edge	6	Checked & Repaired
Edge-edge Gap	6	Checked & Repaired
Vertex-vertex Gap	378	Checked & Repaired
Duplicate Edge	591	Checked & Repaired
T-Junction Edge	0	Checked
Poorly Connected Ed...	0	Checked
Curve Simplification	551	Checked & Repaired
Merged Edge	174	Checked & Repaired
Embedded Edge	2	Checked & Repaired
Revesed Edge In Loop	1	Checked & Repaired
Duplicate Edge	1	Checked
T-Junction Edge	0	Checked
Poorly Connected Ed...	0	Checked
Free Edge	7	Checked & Repaired

Save To SQL   OK

Fig. 10. Checking report.

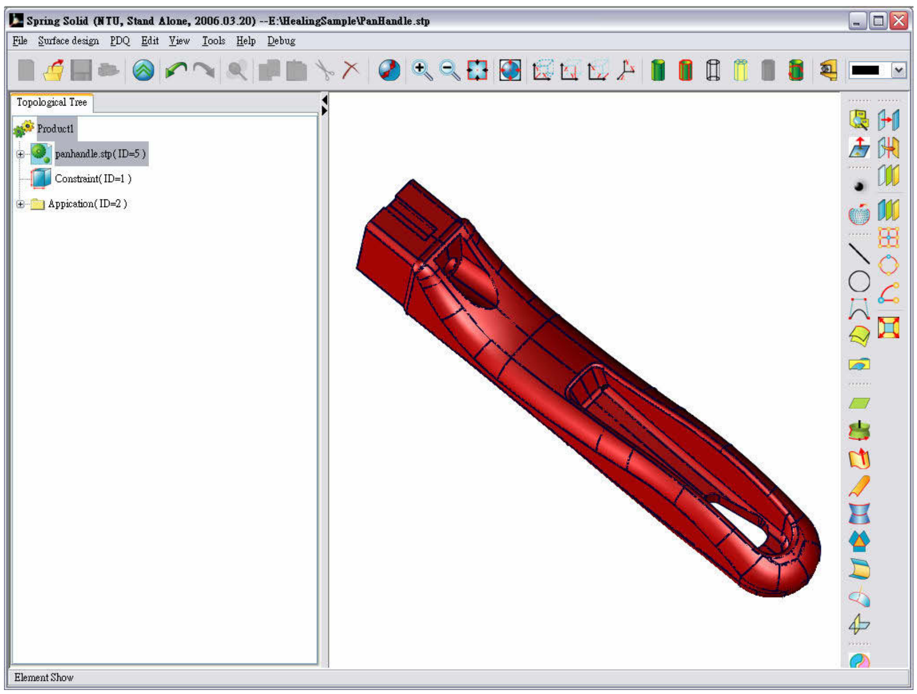


Fig. 11. Rendering of the model after the repairing process has been done.

5. CONCLUSION

This investigation has demonstrated the feasibility of a prototype system of the proposed algorithms in handling the recovering fault model. The developed algorithms were tested on a model from the industry. The data collected from the test samples, some check items such as the edge-edge gap on the vertex, and vertex-vertex coincidency, edge orientations, are crucial check item. Due to the diversity representation of the topological and geometric entities, a robust algorithm for the automatic repairing procedures should be further investigated. Meanwhile, a correct inference of the knowledge base should be extended to include models from various industries.

6. ACKNOWLEDGEMENT
The authors would like to thank the National Science Council of the Republic of China, Taiwan for financially supporting the research under Contract No. NSC 91-2212E002-085.

7. REFERENCES

[1] ISO 10303 – Part 42, Integrated generic resources: geometric and topological representation.
[2] De Boor. C., On Calculating with B-splines, *Journal of Approximation Theory*, Vol. 6, No. 1, 1972, pp 50-62.
[3] JAMA/JAPIA, PDQ Guidelines fundamental, Japan Automobile Manufacturing Association, Inc., CAD group, 2004.
[4] SASIG, Product Data Quality Guidelines for the Global Automotive Industry, Version 2.1, 2005.
[5] FEGS Ltd, CADFix: A product for CAD data exchange, repair, and upgrade, International TechneGroup, Inc.
[6] CAD Doctor, http://www.elysiuminc.com/fr_index.html?.proinfo/caddoctorinfo.html, Elysium Inc.
[7] PDQNP, http://www.elysiuminc.com/prodinfo/pqdnpinfo.html, Elysium Inc.
[8] Q-checker, User manual, Version 1.11 for CATIA, Transcat PLM GmbHh & Co.
[9] Dinesh Shikhare, Healing of 3D geometry database, http://angelfire.lycos.com/spaces2/dineshshikhare/writings/healing.pdf
[10] SASIG, Product Data Quality Guidelines for the Global Automotive Industry, Version 2.1, 2005.
[11] Contero, M., Company, P., Vila, C. and Aleixos, N., Product data quality and collaborative engineering, *IEEE Computer Graphics and Applications*, Vol. 22, No. 3, 2002, pp 32-42.
[12] Gu, H., Chase, T. R., Cheney, D. C., Bailey, T. T. and Johnson, D., Identifying, Correcting, and avoiding errors in computer-aided design models which affect interoperability, *Transaction of the ASME*, Vol. 1, 2001, pp 156-166.