Research and Application of Collision Detection on Steel Structure in Virtual Pre-Assembly Environment

Zhu Yubin¹, Yao Jian², Xu Youwu¹, Zhang Yuling²
¹Institute of Structural Engineering, Zhejiang University, Hangzhou, 310058, China
²College of Urban Construction, Zhejiang Shuren University, Hangzhou, 310015, China
E-mail: yaojian5803@qq.com

Abstract. In order to detect the possible collision of steel structural in virtual pre-assembly environment, a collision detection program based on the bounding box algorithm and the spatial triangle intersection detection algorithm was developed. And the feasibility and effectiveness of the collision detection program in virtual pre-assembly were verified by an example. The results showed that: according to this method, the collision detection in virtual pre-assembly has good feasibility and is suitable to practical projects.

1. Introduction
Steel structures are generally made in the factory (including enlarged components) and transported to the construction site for installation. Entity pre-assembly is generally required in the factory to ensure components can be accurately installed and not collided after installation. Pre-assembly includes the dimensional error of the steel component, the position error of the bolt hole and whether the component will collide with each other. At present, many steel structures are modelling complex (such as various reticulated shell structures, spatial curved structures, etc.), which brings many difficulties to pre-assembly. The entity pre-assembly not only takes up the site and equipment of the factory, but also consumes a lot of manpower and material resources. It needs a lot of costs and difficulties to meet the requirements of the construction of more complex steel structures in the future.

In recent years, with the rapid development of computer technology, the virtual pre-assembly technology has gradually emerged and has been applied well in more and more projects. Computer Aided Design (CAD) can be used to simulate the process of entity assembly, it can assist or even replace the original entity pre-assembly process, so as to reduce costs and improve efficiency.

In order to solve the problem of collision detection in virtual pre-assembly of steel structures, this paper developed a collision detection program based on the bounding box algorithm and the spatial triangle intersection detection algorithm. An example shows that the collision detection method proposed in this paper can be referenced in the practical project.

2. Virtual pre-assembly in engineering application
Some engineering projects in China have optimized the original assembly process by computer, which has improved the accuracy and efficiency of assembly. Such as the terminal building of Kunming New Airport [1], the third ring belt truss layer of Shanghai Center Building [2], and the steel structure engineering of Hangzhou Raffles City [3]. However, the application of this technology in China is not yet mature. At present, this technology is mainly used in complex steel structure projects. The basic
steps of pre-assembly are roughly similar, that is, comparing the coordinates of some control points obtained from the design model with that measured from the actual components, the manufacturing error can be checked. The components should be trimmed and reassembled if these manufacturing precision does not meet the requirements.

In the practical projects, there may be thousands of components and much more control points. If the coordinate conversion and matching are manually operated, the workload is huge and even impossible to establish the measured model. However, the above-mentioned engineering application examples are all manually operated. In reference [4] and reference [5], more advanced measurement techniques were applied, and the virtual pre-assembly systems were used instead of manual coordinate transformation and coordinate fitting. Good results were obtained. However, these systems are virtually assembled through the principle of bolt hole matching, the shape of the components is not considered. Therefore, collision between the components cannot be detected.

3. The research status of virtual collision detection

Domestic and foreign scholars have done a lot of researches on collision detection in virtual pre-assembly environments, and proposed mainstream collision detection algorithms such as bounding box hierarchy method, distance tracking method, and space decomposition method [6-8]. Reference [9] proposed a hybrid strategy-based hierarchical bounding box real-time collision detection method to improve the speed and accuracy of collision detection in virtual surgery. Reference [10] studies the collision detection method of the large-scale lifting machine's virtual tower crane which commonly used in building construction. Through the collision detection strategy based on the bounding box and the level bounding box, the collision detection in virtual lifting construction was realized.

According to the problem of the collision between component connections and installation space reserved for bolt installation in the designing software of detail structural steelwork (STXT), reference [11] studies how to perform collision detection of 3D models in STXT quickly and effectively, as well as components and nodes. Based on the idea of bounding box, the program was achieved using the vector method to detect the outline and distinguish different types of boundary intersection.

At present, the collision detection in virtual environment has been studied and applied in some fields. However, there are few studies on construction and there is no engineering practice. Collision detection technology is an essential part of a virtual pre-assembly system. Due to the complex modeling of steel structures, there may be multiple intersections of components at some nodes, which may cause unreasonable collisions after pre-assembly. Therefore, virtual pre-assembly system needs to provide collision detection. The reference [11] proposes to quickly detect the collision of the three-dimensional model in STXT, but did not solve the possible collision in the actual construction of steel structure. In this paper, the collision detection of steel structural member pre-assembly is studied, which makes up for the deficiency of the current virtual pre-assembly technology and makes a useful attempt for the final replacement of the entity pre-assembly, which can be used for engineering application.

4. The basic principle of the collision algorithm

The reference [11] uses the axial bounding box and traverses the vertices of each 3D model to calculate their maximum coordinates in the x, y and z directions. The size of the bounding box can be determined by these coordinates and a cuboid bounding box can be created. For example, for a cylinder, the cuboid bounding box created after using the axial bounding box is shown in dashed lines in Fig.1.
Figure 1. Comparisons between calculation and test results

In engineering, the actual components need to be detected before virtual pre-assembly. But their dimensions are often different from the theoretical dimensions and the coordinates of components are usually unknown. To perform virtual collision detection on actual components, it is necessary to know the three-dimensional coordinates of the actual component’s position for building component’s model. This paper uses TRITOP optical measurement system to obtain the three-dimensional coordinates of the key points. After computer programming, the bounding box algorithm is used to build the approximate shape of the components according to these three-dimensional coordinates, and then the collision detection in virtual environment is realized by the three-dimensional space collision algorithm.

4.1 The bounding box algorithms

Collision detection algorithm can be divided into three major categories [12]: bounding box layer method, distance tracking method, spatial decomposition method. These algorithms use different principles to simplify the collision detection between objects. At the same time, the accuracy of these three types of algorithms is also different. This paper uses the simpler bounding box hierarchy method as the main algorithm for virtual collision detection.

The bounding box hierarchy mainly include: axis-aligned bounding boxes(AABB) [13], bounding balls [14], and oriented bounding boxes(OBB) [15]. The characteristics of these three algorithms are listed in Table 1. It can be seen that the AABB bounding box method is poorly compact, but the AABB is simple. The bounding balls is generally applied to spherical objects, and the tightness to cuboids is poor and is rarely used. Although the AABB bounding box is more common, and is more simple for its consistency with the axis direction, it also has disadvantages. It doesn't rotate with the object, that is, when the object turns an angle, the AABB bounding box will increase the large gap, which is not very good for more accurate collision detection. At this time, the OBB bounding box is needed. It can always generate the smallest cuboid bounding box along the main direction of the object, and it can be used for more accurate collision detection as the object rotates arbitrarily. Most of the components are bar-shaped or cuboid, and taking into account that the shooting generated points are non-coordinated, the OBB bounding box is chosen as the main algorithm for fitting the object.

| Bounding box method | Tightness | speed  | Difficulty | Accuracy | Application area |
|---------------------|-----------|--------|------------|----------|------------------|
| AABB                | poor      | fast   | easy       | poor     | wide             |
| Bounding Balls      | poor      | fast   | general    | general  | wide             |
| OBB                 | general   | general| general    | general  | general          |
4.2 The principle of OBB
According to the vertices of the object, the feature vector is obtained through principal component analysis, which is the main axis of the OBB. Principal component analysis is a set of linearly independent variables which is transformed from a set of potentially relevant variables through orthogonal transformation.

Firstly, covariance between the three variables x, y, and z was calculated:
\[ \text{cov}(X_i, X_j) = E[(X_i - \mu_i)(X_j - \mu_j)] \] (1)

The covariance matrix can be obtained by the covariance calculation formula (1).
\[
A = \begin{bmatrix}
\text{cov}(x, x) & \text{cov}(x, y) & \text{cov}(x, z) \\
\text{cov}(y, x) & \text{cov}(y, y) & \text{cov}(y, z) \\
\text{cov}(z, x) & \text{cov}(z, y) & \text{cov}(z, z)
\end{bmatrix}
\] (2)

The main diagonal elements represent the variance of the variable. The larger elements of main diagonal represent a stronger signal. The non-primary diagonal elements represent the covariance between the variables. The larger elements of non-primary diagonal represent data distortion. In order to reduce the distortion, the linear combination between variables can be redefined by diagonalizing the covariance matrix. The elements of the covariance matrix are real and symmetric.

The three feature vectors of the covariance matrix A are orthogonal and can be used as a basis for normalization, which represent the direction of the OBB. The size of the OBB can be determined by calculating the maximum and minimum values of each element in the object on this basis respectively. Fifteen scalars are required to store an OBB (there are 9 scalars for the 3 base vectors representing the direction and 6 scalars for the range). Finally, the three-dimensional coordinates of the eight corners of the OBB box can be calculated from these 15 scalars. Through the eight corner points, the geometric size and spatial position of a bounding box can be determined.

4.3 The principle of three-dimensional space collision algorithm
Collisions between objects in virtual environment can eventually be decomposed into intersections between several triangles of the object [16]. The bounding box obtained above can be first split into 6 rectangles. For each rectangle, it can be further divided into 2 triangles, for a total of 12 triangles. Therefore, the collision of two cuboids can be converted into the intersection detection of several triangles. The space intersection detection method for triangles is depicted as follows.

There are ΔABC and ΔPQR whose vertex coordinates are known, Ψ is the support hyperplane of ΔABC. Follow these steps to judge if they intersect or not.

1. Judge if ΔPQR intersects with Ψ
   If ΔABC intersects with ΔPQR, then the three vertices of ΔPQR must be positioned on both sides of Ψ.
   Calculate the normal of Ψ:
   \[ \overrightarrow{n} = \overrightarrow{AB} \times \overrightarrow{AC} = (b - a) \times (c - a) \]
   let \( sp = \overrightarrow{n} \cdot \overrightarrow{AP} \), \( sq = \overrightarrow{n} \cdot \overrightarrow{AQ} \), \( sr = \overrightarrow{n} \cdot \overrightarrow{AR} \), judge:
   1) If \( sp \), \( sq \), \( sr \) are all positive or negative, then ΔABC and ΔPQR do not intersect;
   2) If they are all 0, then ΔABC and ΔPQR are coplanar and can be converted into intersection of triangles in 2D plane space.
   3) If one is 0 and the other two sign are the same, either the two triangles do not intersect or intersect at one vertex. This converts to judge whether the vertex is inside ΔABC in 2D plane space.
   4) If there are two 0, such as \( sp \) and \( sq \), then one side \( PQ \) of ΔPQR lies on plane Ψ, and these two triangles do not intersect or intersect on \( PQ \). This converts into intersection between the line segment and the triangle in 2D plane space.
   5) Otherwise, ΔPQR and Ψ intersect and go to the next step.
2. Calculate the intersection point of ΔPQR and Ψ
The vertices P, Q, R are located on both sides of Ψ. Suppose Q and R be on the same side, then line segments \( \overline{PQ} \) and \( \overline{PR} \) intersect with Ψ respectively, and the intersection point is the intersection of ΔPQR and Ψ.

The line segments \( \overline{PQ} \) can be expressed as a vector: \( \overrightarrow{P+1 \times PQ} \), in which \( 0 \leq t \leq 1 \), then the intersection H between \( \overline{PQ} \) and Ψ can be calculated as follows:

\[
h = \frac{n \cdot \overrightarrow{PQ}}{(n \cdot \overrightarrow{PA} - n \cdot \overrightarrow{AQ})} = \frac{sp}{(sp - sq)}
\]

The intersection F between line \( \overline{PR} \) and Ψ is:

\[
f = \frac{pl \times PR}{(pl \times PA - pl \times AQ)} = \frac{sp}{(sp - sr)}
\]

3. Judge if line \( \overline{HF} \) intersects with ΔABC

The intersection of ΔPQR and Ψ can be obtained as HF from step 2. Then ΔABC intersects with ΔPQR if and only if HF intersects with ΔABC. Since HF is coplanar with ΔABC, it can be converted into the intersection test between the line segment and the triangle in 2D plane space.

In 2D plane space, the line where the line segment lies divides the plane into two disjoint half-open planes. The points in the two half-open planes are respectively located on the left or right side of the line segment. The following conditions can be used to judge whether line segment \( \overline{HF} \) intersects ΔABC:

1. If the vertices A, B, C are on the same side of the line HF, then line segments \( \overline{HF} \) and ΔABC do not intersect;

2. Otherwise, suppose vertices A and B are on one side and C is on the other side, then, intersect with the line HF respectively. If H and F are on the right side of \( \overline{HF} \) or on the right side of \( \overline{HF} \) at the same time, then \( \overline{HF} \) and ΔABC do not intersect; otherwise, they intersect.

5. Application of collision algorithm

The three-dimensional size of each component must be measured to ensure that the design and construction requirements are met before site installation. A number of measurement targets are then arranged on the individual component surface to build bounding box. For example, when the member is a cuboid, measurement targets are arranged on eight corner points, contour lines, and six surfaces of the cuboid. The initial three-dimensional coordinates of the target are then measured using an optical measurement system. After testing by the GPA pre-assembly [4] successfully, the coordinate value of the target point converted from the initial three-dimensional coordinates is imported into the virtual collision detection program (see Appendix A). The collision detection is divided into two steps. Firstly, the minimum bounding boxes of each component are fitted through the OBB method. Then the collision detection in virtual environment is achieved through the spatial triangle intersection detection algorithm. If the test result is colliding, it indicates that there are defects in the structural design. The theoretical design drawing needs to be checked, the components need to be reassembled and tested for collision. If the test result is not colliding, virtual pre-assembly of the structure is successful and the pre-assembly on the spot can be directly performed.

The following example gives a brief description of the application of the virtual collision detection method.

Fig.2 shows that the structural unit consists of two I-shaped members (2-1 and 2-2) and two channel members (2-3 and 2-4). Two I-shaped members are respectively inserted into two channel members and connected with the channel members by bolts. After ensuring that the components met the design and construction requirements and the GPA pre-assembly was successful, virtual collision detection was performed on the structural unit. Since both ends of the I-shaped member are inserted
into the inside of the channel members, both ends of the I-shaped member may collide with the web of the channel members, so collision detection is required.

Since the cross-section of each structural element is not a single rectangle but an I-shape or a channel shape consisting of three rectangular, when the OBB method was used, each component was divided into three parts: upper and lower flanges and web. The corresponding sub-boxes were simulated for each part using the OBB method, so that an accurate bounding box model was obtained. Finally, the spatial triangle intersection detection algorithm was used to detect the bounding box of different components. After program was run, the result was "no collision between member 2-1 and members 2-3, 2-4; no collision between member 2-2 and members 2-3, 2-4". The coordinates of the bounding box corners of all components were calculated in the program, and the model of the bounding box was drawn based on these coordinates, as shown in Fig. 3. The actual model was assembled and the results were shown in Fig. 4. Comparing the two figures, it shows that the calculation result is consistent with the actual pre-assembly result.

In practical projects, the cross-section of components is more diverse. For the angle steel, channel steel, I-shaped steel and other members of different cross-section types, the method of this paper can be used for virtual collision detection. At the same time, the program can also be optimized to expand the types of optional cross-section, and the engineering-oriented software can be compiled further to meet the actual engineering needs.
6. Conclusion
In this paper, the feasibility of collision detection intersects with the line HF respectively n in virtual pre-assembly environment was discussed. Based on the bounding box algorithm and the spatial triangle intersection detection algorithm, a moderately accurate collision detection program was compiled. Finally, the feasibility of collision detection algorithm in practical engineering was proved by an example.

Collision detection of steel structures in virtual pre-assembly environment perfects the detection process of the current virtual pre-assembly technology, makes virtual pre-assembly results more reasonable and reliable, and lays a good foundation for virtual pre-assembly technology to eventually replace the entity pre-assembly.

Acknowledgement
The authors are grateful for the financial support received from the Science and Technology Department of Zhejiang Province (Project No.: LY18E080014).

References
[1] Tang Jiyu, Wu Liuning, Liao Gonghua. Application of Simulation Assembly Technology for Steel Structure in Kunming New Airport Terminal [J]. Construction Technology, 2009, 38(12):p21-24. (in Chinese)
[2] Ding Yifeng, Lu Hua, Li Wenjie. The Application of Information Technology Pre-Assembly in Finished Product Inspection & Test of Steel Structures[J]. Journal of Information Technology in Civil Engineering and Architecture, 2012, (1):p52-56. (in Chinese)
[3] Zhang Dafei, Zhang Jianping, Du Bingbing, Wang Tao, Chen Hui. Simulation of Pre-Assembled Technology Used for Project of Steel Structure In Hangzhou Raffles City [J]. Steel Construction,2014,(06):p56-60. (in Chinese)
[4] Li Yang. Method Study of Virtual Trial Assembly of Steel Structure[D].Zhejiang University,2015. (in Chinese)
[5] Tamai S, Yagata Y, Hosoya T. New technologies in fabrication of steel bridges in Japan[J]. Journal of Construcional Steel Research, 2002, 58(1):p151–192.
[6] Noborio H, Fukuda S, Arimoto S. Fast Interference Check Method Using Octree Representation [J]. Advanced Robotics. 1989, 3(3): p193-212.
[7] Suri S, Hubbard P M, Hughes J J. Collision Detection in Aspect and Scale Bounded Polyhedra [A].Proc. 9th ACM-SIAM Sympos. Discrete Algorithms, 1998, p127-136.
[8] Wei Ying-mei. Research on Collision Detection in Virtual Environment [D]. National University of Defense Technology,October, 2000.
[9] Xie Qianru. Research on Collision Detection and Soft Tissue simulation Technolony for Virtual Surgery[D]. Northwest University, 2013. (in Chinese)
[10] Hou Xiaoting, Li Changhua. Research on Collision Detection for Virtual Hoisting Construction Scene[J]. Journal Of System Simulation, 2014(01):p76-84. (in Chinese)
[11] Zhang Xiaolong, Ma Encheng, Xia Xuyong, Jin Jiaru, Chen Yulin. Research and Application of Collision Detection Technology on the 3D Model of Steel Structure[J]. Journal of Information Technology in Civil Engineering and Architecture,2009,(02):p51-54. (in Chinese)
[12] Zhou Yunbo, Yan Qingdong, Li Hongcai. Collision Detection Algorithms Analysis in Virtual Environment [J]. Journal Of System Simulation, 2006(S1):p103-107. (in Chinese)
[13] GINO VAN DEN BER GEN. Efficient Collision Detection of Complex Deformable Models using AABB Trees [J]. Journal of Graphics Tools, 1999, 4(2): p1-13.
[14] Welzl E. Smallest enclosing disks (balls and ellipsoids)[C]/ of Lectures and Notes in Computer Science. Berlin:Springer, 1991. 1991:p359--370.
[15] Gottschalk S, Lin M C, Manocha D. OBB Tree: A Hierarchical Structure for Rapid Interference Detection [C]//Computer Graphics (SIGGRAPH' 96), New Orleans, LA, USA, 1996, p30: 1712180.

[16] Wang Xiaorong. Research of Collision Detection Algorithms Based on AABB D]. Central China Normal University, 2007. (in Chinese)