An encryption approach for product assembly models

Cai, XT, Wang, S, Lu, X & Li, W

Original citation & hyperlink:
Cai, XT, Wang, S, Lu, X & Li, W 2017, 'An encryption approach for product assembly models' *Advanced Engineering Informatics*, vol 33, pp. 374-387
https://dx.doi.org/10.1016/j.aei.2016.12.001

DOI 10.1016/j.aei.2016.12.001
ISSN 1474-0346

Publisher: Elsevier

An Encryption Approach for Product Assembly Models

X.T. Cai[1,2], S. Wang[2], X. Lu[2], W.D. Li[2*]

[1]School of Computer Science and Technology, Wuhan University, Wuhan, China

[2]Faculty of Engineering, Environment and Computing, Coventry University, Coventry, UK

[*]Corresponding author: email: aa3719@coventry.ac.uk

Abstract

In a collaboration environment, it is a challenge how to effectively share the information needed for collaboration while protecting other confidential information in a product assembly model. In this paper, an innovative encryption approach for assembly models to support collaboration is presented. This approach is content based encryption and effective for the secure sharing of feature-based assembly models. In the approach, a classification algorithm for features in an assembly model to be shared or protected during collaboration has been first developed. An encryption algorithm for a feature has been then designed to ensure the parameterization, topological and geometrical validity, and self-adaptability of the encrypted feature. An algorithm for parts with multiple encryption features has been developed. Based on the above algorithms, parts are finally assembled and the geometry and topology of the assembling structure are kept un-changed to enhance collaborators' interoperability. The characteristics and innovations of the approach include: 1) the approach is feature based, integrative into the main-stream commercial Computer Aided Design (CAD) systems, and flexible to meet various users' needs for encrypting features selected by users during collaboration, 2) in the approach, the topological and geometrical validity of an assembly model after encryption is maintained to ensure effective collaboration on the assembly, and 3) the approach is parametrically controlled through adjusting position and size parameters so as to ensure the user friendliness of using the approach. A case study with complex geometries and assembly structures has been used to validate the effectiveness and robustness of the approach in industrial applications.

Keywords: Assembly encryption, Assembly features, Encrypted features, Feature classification

1. Introduction

Product development enterprises consider their product models as core intellectual properties [1, 2]. In order to support collaborative product development effectively, flexible encryption approaches on product models (e.g., Computer Aided Design (CAD) models) are imperative to ensure effective information sharing for collaboration as well as protection of other private information in the models [3, 4].

An assembly contains critical assembly structure information and the parts in it contain abundant design features, design procedure and feature parameters. However, the related security research on assemblies have focused on the part level. It is still far away from meeting industrial requirements [5]. To protect the assemblies as well as supporting the flexibly sharing and collaboration in a network based manufacturing environment, an innovative encryption approach for product assembly models is presented in this paper. The innovations of the approach include:

(1) The approach is feature based and can be integrated to main-stream feature based CAD systems that have been widely used in industries. Through the feature based encryption mechanism for an assembly model, users' needs of encrypting selected features will be met, and the assembly features and structure will be maintained to facilitate collaboration;

(2) The approach is based on the geometrical deformation of features. By maintaining assembly features while deforming other selected features in an assembly model, the validity of the structure in the assembly model is ensured while other features to be protected by geometrical deformation;

(3) The approach is designed based on a parametrically controlled mechanism to enhance user friendliness. Geometrical deformation for encrypting features in an assembly model is controllable by users through adjusting position and size parameters defined in the parametric mechanism of the encryption approach.

The remainder of this paper is organized as follows. In Section 2, related research work is surveyed. Section 3 introduces the details of the encryption approach for assembly models. A case study to validate the approach is given in Section 4. Finally, conclusions are given in Section 5.

2. Related Work

In the past years, a number of related research projects have been conducted. The research can be classified as three categories [6] - (1) collaboration access control, (2) simplification of product models for collaboration, and (3) feature-based product model encryption. The related work are summarized below:

Collaboration access control has been widely used for data security in a network environment. Mechanisms were designed to authorize users to access the shared data. In early times, generic access control methods were used for protecting product models during collaboration [7-11]. Later on, in consideration of the complexity of design data, some dedicated access control methods were developed, such as access control based CAD architecture [12], ADOSX system that can handle CPD (collaborative product development) between two enterprises by Stevens [13], a secure access control mechanism for 3D models [14], a security approach for a distributed product data management system [15], etc. Moreover, taking account of the frequent sharing of design data, sharing space based access control methods were developed, in which a secure sharing space was designed [16-18]. Moreover, for further improving model security during collaboration, access control mechanisms were reinforced by introducing digital signature or watermark mechanisms for product model protection [19-22]. Generally speaking, although different access levels for the models can be defined, the protecting granularity is not small enough and the confidential information of every access level cannot be arranged flexibly by the model owner. As such, the developed methods on product models are still not flexible enough to support collaboration.

In order to protect the private information of product models during collaboration, methods for simplification of product models (e.g., as different Level Of Details (LODs)) were developed. Cera proposed a LOD based access control method for CAD models [23, 24]. Chu developed a mechanism for sharing of LOD CAD models for product collaboration [25, 26]. Li proposed a matrix-based modularization approach for CAD models [27]. The above approaches are mainly aimed at part models. To support assembly based collaboration, some research work focus on the mechanism of simplifying assemblies. In the simplification process of assembly models, internal parts and features that are not visible from outside are the first to be removed. Kanai and Yu detected invisible features by pre-rendering the models from multiple view directions [28, 29]. Han proposed an approach to automatically create a simplified assembly models with the desired LOD value [30, 31]. The simplification methods were mainly designed for the purpose of efficient model sharing via the Internet with limited bandwidth to support collaboration. The levels of details cannot be designated by the model owner. As such, users' collaboration requirements on secure sharing of assemblies could not be addressed effectively.

Encryption research was conducted to protect CAD models. Special encryption of for 3D models was proposed [32]. However, the developed encryption methods are not feature based, which are not easily integrated with the main-stream CAD systems used in industries and inflexible to operate during collaboration.

The authors of this paper have actively developed related research in recent years. The feature based encryption methods developed by the authors have provided flexible mechanisms for users to select features for encryption, and the encryption process on features can be controlled by users in a parametric means [33, 34]. On the other hand, collaboration is usually carried out in an assembly level, while the related research conducted on the feature level has limited the effectiveness of the methods in applications.

3. Encryption Approach for Assembly Models

A collaboration application scenario for encrypting an assembly model is illustrated in Figure 1. For instance, there are two collaborators to work on an assembly model. Part 1 and Part 2 are designed by Collaborator 1 as a sub-assem_1, to be assembled with Part 3 designed by Collaborator 2 as an assembly model. During collaboration, assembly features and associate features are kept the same to indicate the assembly relationships while other features are encrypted for design information protection.

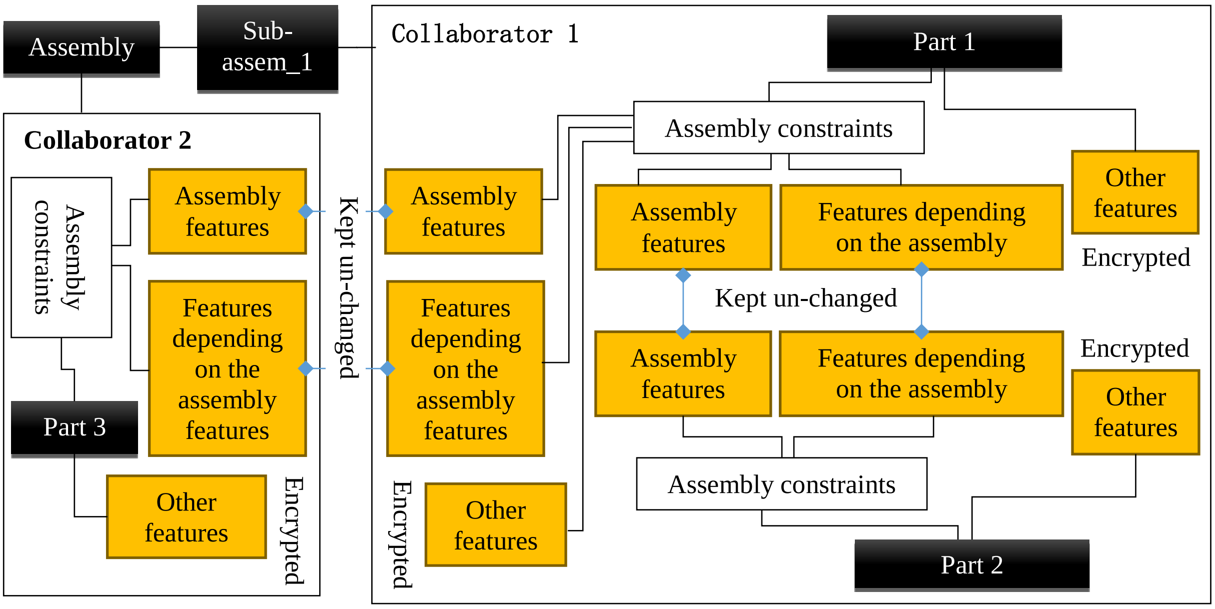


Figure 1: The collaboration scenario for encrypting an assembly model.

The encryption process of the approach is depicted in Figure 2. The approach is implemented through the following three algorithms:

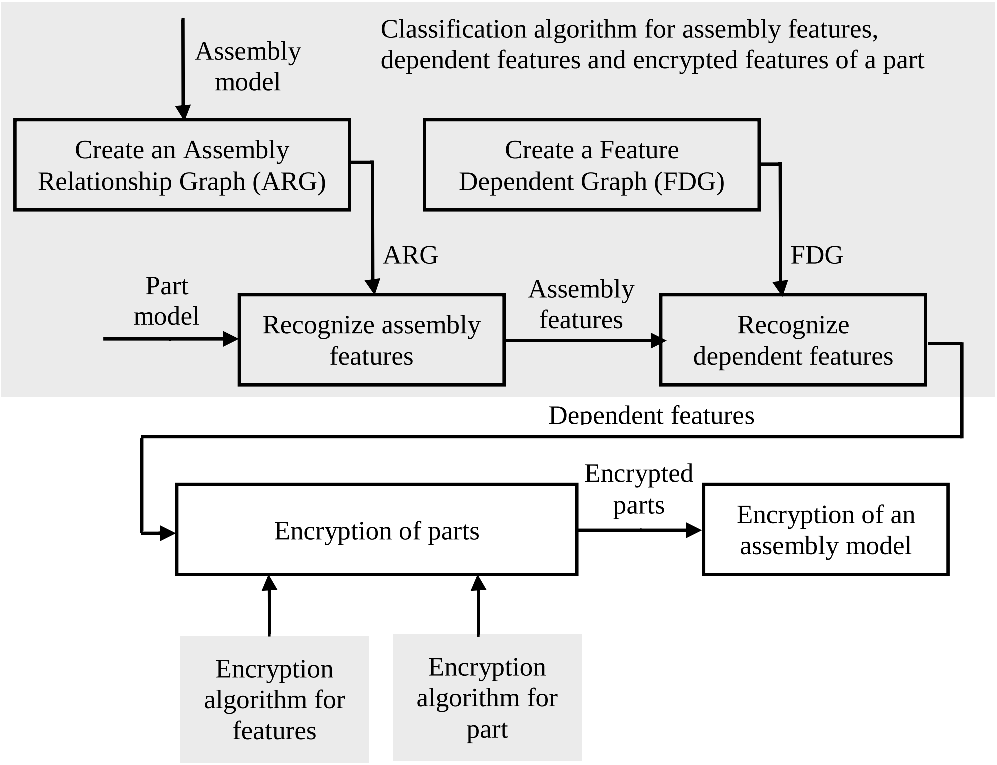

Assembly model
Classification algorithm for assembly features, dependent features and encrypted features of a part
Create an Assembly Relationship Graph (ARG)
Create a Feature Dependent Graph (FDG)
ARG
FDG
Part model
Recognize assembly features
Assembly features
Recognize dependent features
Dependent features
Encrypted parts
Encryption of parts
Encryption of an assembly model
Encryption algorithm for features
Encryption algorithm for part

Figure 2: The encryption process and algorithms of the approach

(1) Classification algorithm for assembly features, dependent features and encrypted features - In order to satisfy assembly constraints, assembly relations and related features should be maintained after the encryption of other features and the parts. A classification algorithm is developed for classifying assembly features, dependent features for the assembly and features for encryption. An Assembly Relationship Graph (ARG) and a Feature Dependent Graph (FDG) are developed to support the above process;

(2) Encryption algorithm for features – The algorithm is based on an encryption transformation matrix, which is parametrically controlled by users, and adaptively adjustable during encryption to keep geometric and topological validation. Based on the encrypting matrix, a feature can be encrypted adaptively to guarantee the validity of the part model containing the encrypted features.

(3) Encryption algorithm for parts – The encrypted features are encoded one by one. The process will follow the creation order of features to ensure the validity of the part for assembling after the encryption of the dependent features.

The details of the algorithms are explained in the following sub-sections.

3.1 Classification algorithm for assembly features, dependent features and encrypted features

The classification of assembly features, dependent features and features for encryption is one of the key issues in the encryption process of an assembly model. Several definitions are given below.

Definition 1: Assembly Feature (AF). The features are used for assembling between two parts in an assembly model.

Definition 2: Dependent Feature (DF). The features that should be maintained to support the maintenance of the topological entities used for assembling during the encrypting of parts for an assembly model (including the AFs).

Definition 3: Encrypted Feature (EF). The features that should be encrypted during the encryption process of an assembly model.

To classify DFs and EFs of a part, AFs should be recognized first. In order to facilitate the process, an Assembly Relationship Graph (ARG) is defined to represent the assembly relationships between the parts of an assembly model.

Definition 4: Assembly Relationship Graph (ARG). It is a graph to show the structure and assembly constraints of an assembly. *N(t, id)* denotes a node of the ARG, where *t* is the type of the node (assembly model or part model) and id is the name of the node; $S(N(assembly, id_1), N(t, id_2))$ is a directed edge from $N(t, id_2)$ to $N(assembly, id_1)$ to denote an assembly relationship between $N(assembly, id_1)$ and $N(t, id_2)$; $C(N(t, id_1), N(t, id_2), Tp_1, Tp_2)$ is a undirected edge between $N(t, id_1)$, $N(t, id_2)$ to denotes an assembly constraint where the constraint is conducted by the topological entities Tp_1 and Tp_2.

As illustrated in Figure 3, the left part - a main sub-assembly of a machine is named saddle, and the right part is the ARG of the saddle. The saddle is made up of a part (*N(part, saddle_top)*) and a sub-assembly(*N(assembly, saddle_base)*), and the saddle_base contains four parts (*N(part, saddle_handle)*,*N(part, saddle_body), N(part, indexing), N(part, indexing_cover)*). The purple un-directed edge shows the assembly constraints.

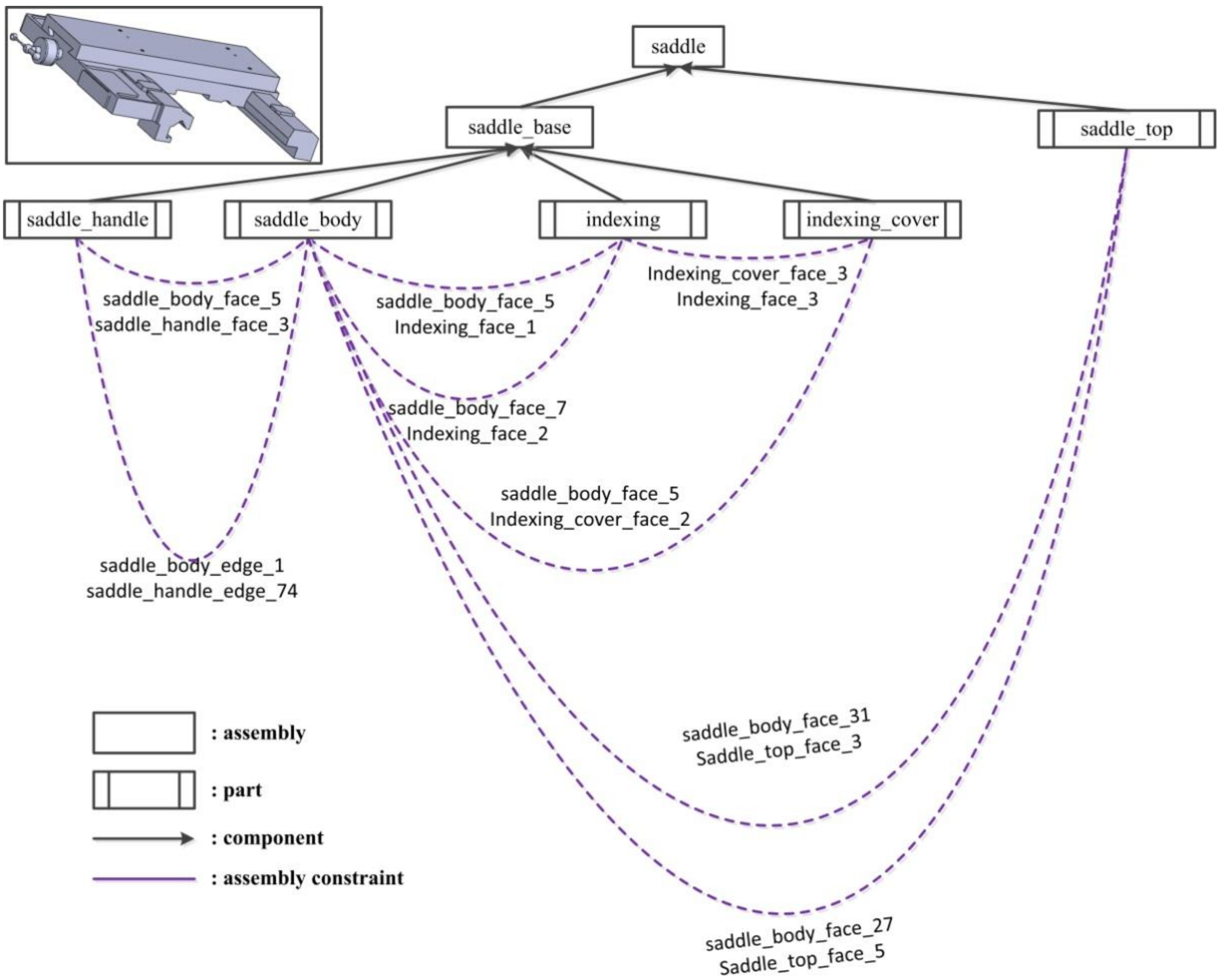


Figure 3: The ARG of the saddle.

Based on ARG, AFs can be recognized using the following Algorithm ***AF_recognition( )****(pseudo code)*:

```
AF_recognition(Assembly, Part)
1. Assembly is an assembly
2. Part is a part in the Assembly
3. Initialize a set for assembly features: AF{}
4. Create the ARG of the Assembly;
5. Search the ARG and store all the topological entities of the assembly constraints related to the Part
   into a set:  A_constraints{ }
6. Add the features of the Part which are related to the topological entities in the A_constraints{ }
   into AF{ }
7. Return (AF{});
```

Based on the algorithm, AFs of all the parts in the saddle are recognized as shown in Figure 4.

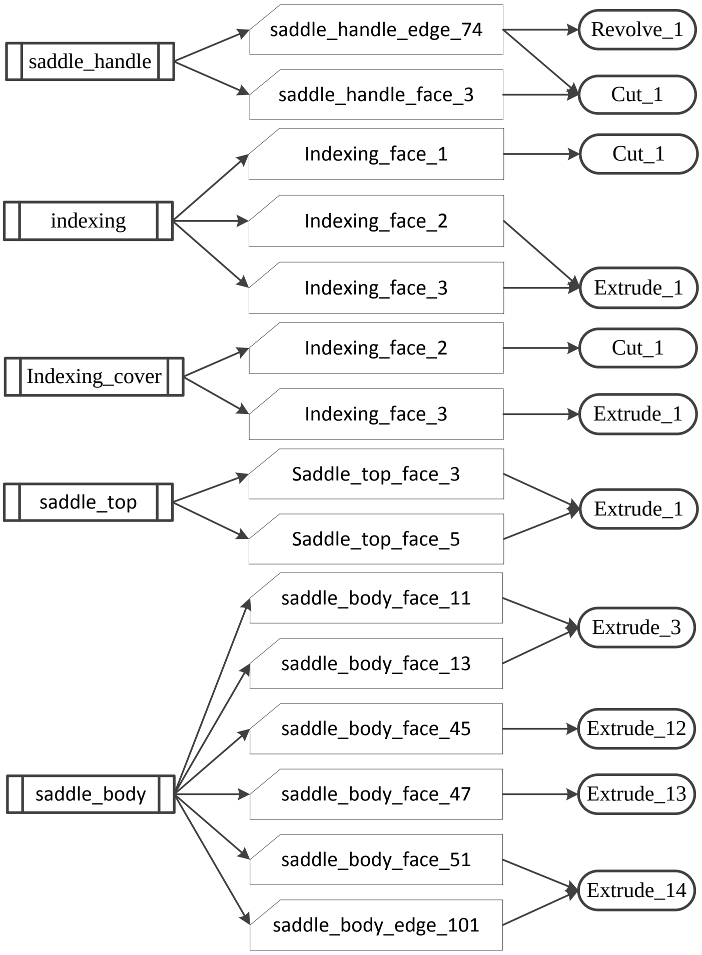


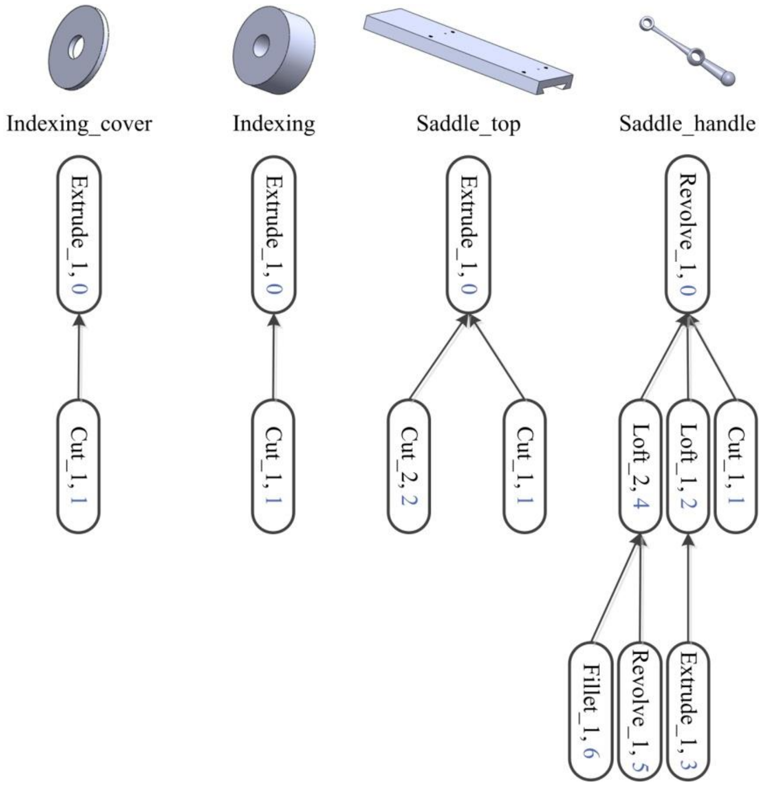
Figure 4: AFs recognized from all the parts in the saddle.

To keep AFs un-encrypted is not sufficient to maintain the topological entities used for assembling. The deformation of an AF's parent features will also lead to the changes of the topological entities used for assembling. As such, except AFs, AFs' parent features, i.e., DFs, should be identified for encryption.

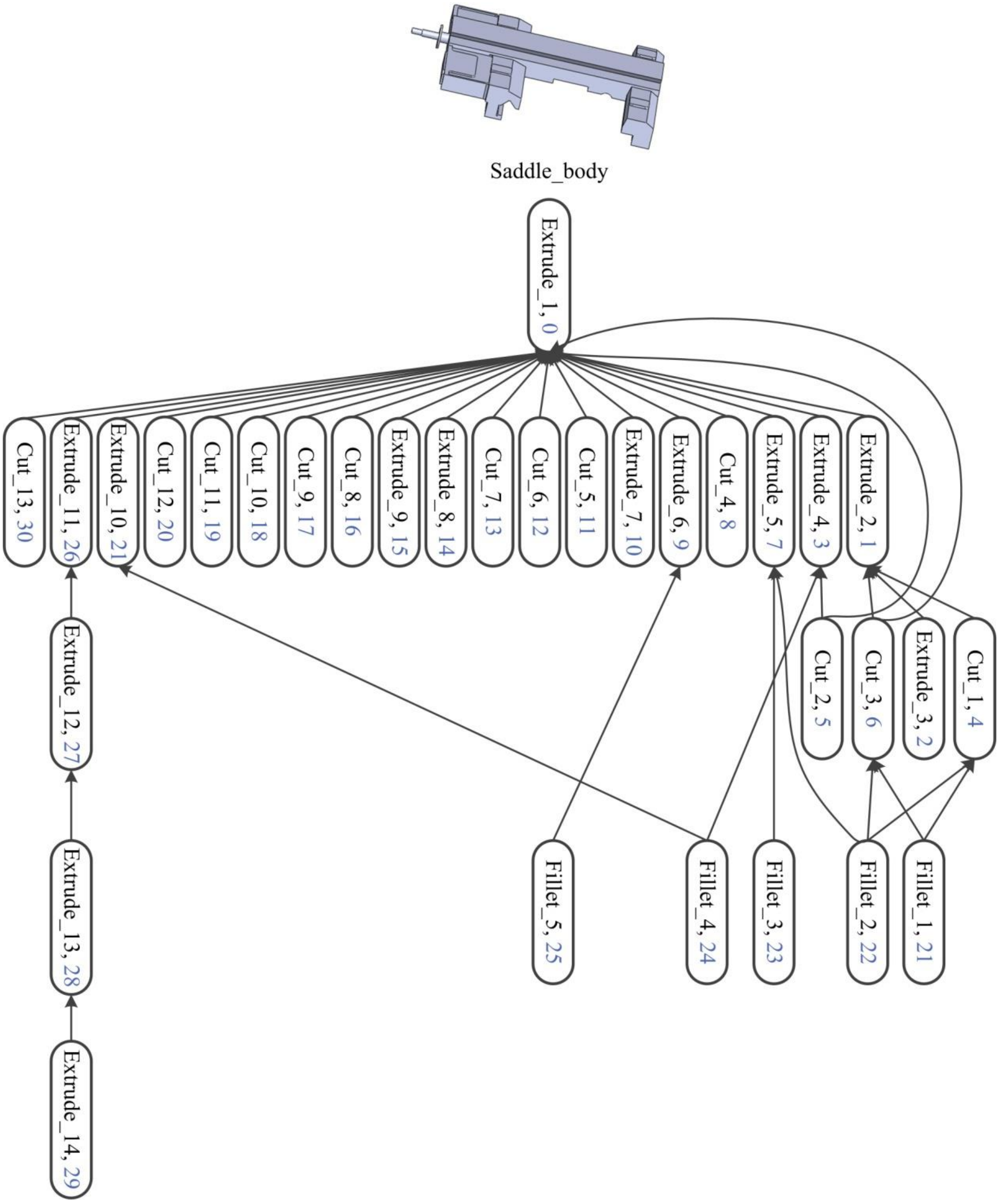
In order to recognize DFs, a definition of Feature Dependent Graph (FDG) is built up as following to represent the structure of a part.

Definition 5: Feature Dependent Graph (FDG). It is a graph to represent the structure of a part. *N(id, order)*is a node to denote a feature of the part, *id* denotes the id of *the* feature, *order* denotes the creation order of the feature where the *order* of the first feature is 0. *E(N(id*$_1$*, order*$_1$*), N(id*$_2$*, order*$_2$*))* is a directed edge from *N(id*$_2$*, order*$_2$*)* (the child feature) to *N(id*$_1$*, order*$_1$*)* (the parent feature) to describe the dependent relationship.

Figure 5 illustrates the FDG of each part in the saddle.



a) The FDG of Indexing_cover, Indexing, Saddle_top and Saddle_handle



b) The FDG of Saddle_body

Figure 5: The FDG of each part in the saddle.

Based on the FDG and AFs recognized via the classification algorithm, DFs can be recognized according to the following algorithm ***DF_recognition( )*** *(pseudo code).*

DF_recognition(Part, Assembly)

```
Part is a part
Assembly is an assembly and be re-numbered
```

```
AF{ }=AF_recognition(Assembly, Part) // AF{ } is the recognized AF set
Create the FDG of the Part;
Initialize a DF set: DF{ };
Add all the AFs into the DF{};
Add all the parent features of AFs into DF{};
Return (DF{});
```

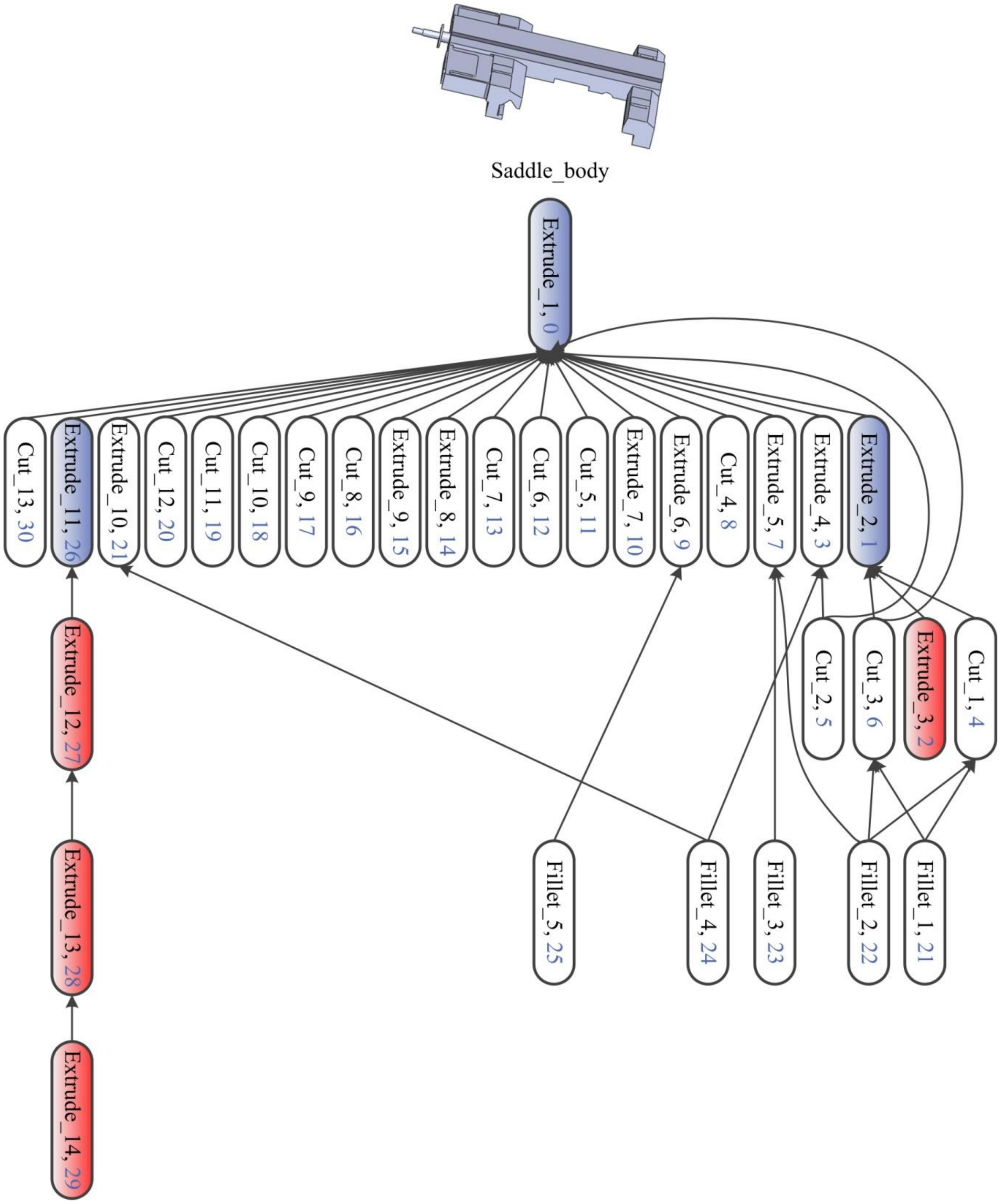


Figure 6: Recognized DFs of the Saddle_body.

As shown in Figure 6, it is the FDG of the saddle_body part in the saddle, The red features (Extrude_3, Extrude_12, Exturde_13, Extrude_14) are AFs. Based on the classification algorithm of DFs, except AFs, the blue features are also identified as DFs.

The identified DFs need to be maintained in the encrypting of parts, and all the remaining features are EFs. As shown in Figure 6, all the white features are EFs.

3.2Encryption algorithm for features

Assemblies are composed of a grope of part models, and features are the basic elements of part models and determine the shape and parameters of part models, as thus, encryption algorithms for features will be essential for the encryption approach for assembly models. The detail encryption algorithm for features are described below.

The features in CAD systems can be classified as the following five types:

- A primary feature is sketch based, and its shape is decided by its constructive sketches;
- A dependent feature depends on primary features, such as chamfers, fillets, corners, etc.;
- A complex feature is a group of features arranged by a certain pattern or mirror operation;
- A combined feature is the combination of two or more features, such as countersunk holes, etc.;
- An auxiliary feature is used as the reference for building upon other features, such as axis, plane, etc.

Based on the above analysis, the shapes of the dependent features, complex features and combined features in a part are finally determined by the related primary features, and the auxiliary feature will not influence the shape and volume of the part. As thus, the shapes of all the features in a part are determined by the sketches of primary features, and the encryption of the sketches can achieve the encryption of the features then the part. The shape of the sketch is controlled by its vertices p_{ij} (controlling points) which can be expressed as a *nm*-dimensional matrix as the following Representation (1):

$$S = \begin{pmatrix} p_{11} & p_{12} & \cdots & p_{1m} \\ p_{21} & p_{22} & \cdots & p_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ p_{n1} & p_{n2} & \cdots & p_{nm} \end{pmatrix}, (m = 2 \,||\, m = 3); \tag{1}$$

Where the value of *m* is decided by the dimension of the sketch (m=2 if 2-dimension otherwise 3 for 3-dimension).

The encryption of the sketch can be conducted by an encryption transformation matrix (denoted as *A*), represented as the following Representation (2):

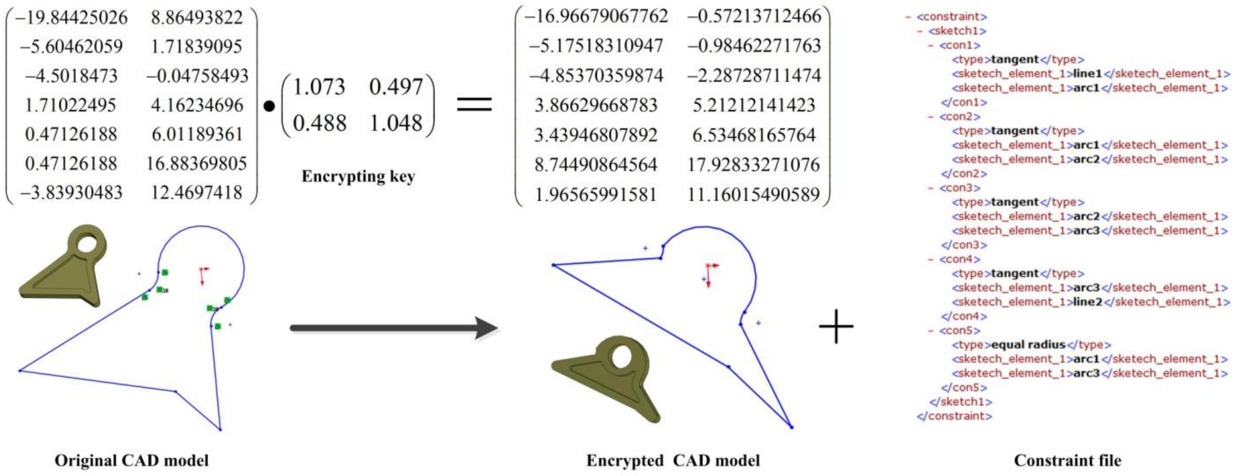
$$A = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1m} \\ a_{21} & a_{22} & \cdots & a_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mm} \end{pmatrix}, (m = 2 \,||\, m = 3), |\,\mathrm{A}\,| \neq 0; \qquad (2)$$

Where |A|≠0 to ensure that the encryption is reversible (decryption).

The encryption of a sketch can be formularized as Representation (3):

$$S' = S \cdot A \qquad (3)$$

In addition to the sketch elements, a sketch contains a group of constraints which are auxiliary for sketch creation and contain rich design knowledge and experiences, the constraints would not influence the shape of the sketch but restrict the change of the sketch. Therefore, in order to support the encryption of the sketch, its constraints should be retrieved and stored separately first. Hence, three parts would be generated after the encryption of a sketch, encrypted sketch, key and constraint file which are stored as an XML file. Applying the encryption of sketches to a feature, an encrypted feature, an XML file containing an encrypting matrix and the constraints information of all the related original sketches are generated. As illustrated in Figure 7, the left part shows an original part, its sketch and sketch matrix, and the right shows the encrypting result.



$$\begin{pmatrix} -19.84425026 & 8.86493822 \\ -5.60462059 & 1.71839095 \\ -4.5018473 & -0.04758493 \\ 1.71022495 & 4.16234696 \\ 0.47126188 & 6.01189361 \\ 0.47126188 & 16.88369805 \\ -3.83930483 & 12.4697418 \end{pmatrix} \bullet \begin{pmatrix} 1.073 & 0.497 \\ 0.488 & 1.048 \end{pmatrix} = \begin{pmatrix} -16.96679067762 & -0.57213712466 \\ -5.17518310947 & -0.98462271763 \\ -4.85370359874 & -2.28728711474 \\ 3.86629668783 & 5.21212141423 \\ 3.43946807892 & 6.53468165764 \\ 8.74490864564 & 17.92833271076 \\ 1.96565991581 & 11.16015490589 \end{pmatrix}$$

```
- <constraint>
  - <sketch1>
    - <con1>
        <type>tangent</type>
        <sketech_element_1>line1</sketech_element_1>
        <sketech_element_1>arc1</sketech_element_1>
      </con1>
    - <con2>
        <type>tangent</type>
        <sketech_element_1>arc1</sketech_element_1>
        <sketech_element_1>arc2</sketech_element_1>
      </con2>
    - <con3>
        <type>tangent</type>
        <sketech_element_1>arc2</sketech_element_1>
        <sketech_element_1>arc3</sketech_element_1>
      </con3>
    - <con4>
        <type>tangent</type>
        <sketech_element_1>arc3</sketech_element_1>
        <sketech_element_1>line2</sketech_element_1>
      </con4>
    - <con5>
        <type>equal radius</type>
        <sketech_element_1>arc1</sketech_element_1>
        <sketech_element_1>arc3</sketech_element_1>
      </con5>
    </sketch1>
  </constraint>
```

Figure 7: An example of sketch encryption

In order to improve the user friendliness, security and validity of encrypted features and models, the encryption transformation matrix (denoted as *A*) will be further enhanced from three aspects: parameterization, randomicity and self-adaptability.

The encryption degree of a model would be parametrically controllable to guarantee geometrical validity, as thus, to make the A in the above Representation (2) to be parameterized. Representation (2) can be re-written as Representation (4):

$$A = \begin{pmatrix} a_{11} & 0 & \mathrm{K} & 0 \\ 0 & a_{22} & \mathrm{K} & 0 \\ \mathrm{M} & \mathrm{M} & \mathrm{O} & \mathrm{M} \\ 0 & 0 & \mathrm{L} & a_{mm} \end{pmatrix} + \begin{pmatrix} 0 & a_{12} & \mathrm{K} & a_{1m} \\ a_{21} & 0 & \mathrm{K} & a_{2m} \\ \mathrm{M} & \mathrm{M} & \mathrm{O} & \mathrm{M} \\ a_{m1} & a_{m2} & \mathrm{L} & 0 \end{pmatrix} = Z + D \tag{4}$$

Z and *D* can be further specialized to *Z'* and *D'* as the Representations (5):

$$Z' = \begin{pmatrix} 1+\gamma & 0 & \mathrm{K} & 0 \\ 0 & 1+\gamma & \mathrm{K} & 0 \\ \mathrm{M} & \mathrm{M} & \mathrm{O} & \mathrm{M} \\ 0 & 0 & \mathrm{L} & 1+\gamma \end{pmatrix}; \quad D' = \begin{pmatrix} 0 & \beta & \mathrm{K} & \beta \\ \beta & 0 & \mathrm{K} & \beta \\ \mathrm{M} & \mathrm{M} & \mathrm{O} & \mathrm{M} \\ \beta & \beta & \mathrm{L} & 0 \end{pmatrix} \tag{5}$$

The sketch *S* can be transformed based on the matrix *Z* as the Representation (6). Obviously, *Z* based transformation is a size scaling.

$$S' = S \times Z' = (1+\gamma) \begin{pmatrix} p_{11} & p_{12} & \mathrm{K} & p_{1m} \\ p_{21} & p_{22} & \mathrm{K} & p_{2m} \\ \mathrm{M} & \mathrm{M} & \mathrm{O} & \mathrm{M} \\ p_{n1} & p_{n2} & \mathrm{L} & p_{nm} \end{pmatrix} \tag{6}$$

Similarly, the sketch *S* can be transformed based on the matrix *D*as Representation (7). Comparably, *D* based transformation changes the coordinates inconsistently that is a deformation transformation.

$$S' = S \times D' = \begin{pmatrix} \sum_{k=1}^{m} \beta p_{1k} - p_{11} & \sum_{k=1}^{m} \beta p_{1k} - p_{12} & \mathrm{K} & \sum_{k=1}^{m} \beta p_{1k} - p_{1m} \\ \sum_{k=1}^{m} \beta p_{2k} - p_{21} & \sum_{k=1}^{m} \beta p_{2k} - p_{22} & \mathrm{K} & \sum_{k=1}^{m} \beta p_{2k} - p_{2m} \\ \mathrm{M} & \mathrm{M} & \mathrm{O} & \mathrm{M} \\ \sum_{k=1}^{m} \beta p_{nk} - p_{n1} & \sum_{k=1}^{m} \beta p_{nk} - p_{n2} & \mathrm{L} & \sum_{k=1}^{m} \beta p_{nk} - p_{nm} \end{pmatrix} \tag{7}$$

Based on the above analysis, the encrypting matrix *A* can be changed to *A'* as Representation (8).

$$A' = Z' + D' \tag{8}$$

It can conclude that, the encrypting matrix *A'* is controlled by the parameters of *γ and β* and *A'* based encryption is size scaling and deformation mixed transformation whose transforming scale is controlled by the parameters *γ* and *β*. Figure 8 shows a sample of *A'* based encryption. On the top of Figure 8, a part and its feature tree, sketch, feature types and sketch matrix are given. On the bottom of Figure 8, it can be found that, it is size scaling based on the parameter *γ* and deformation transformation based on the parameter *β*, and the *A'* based encryption is the mix transformation of size scaling and deformation.

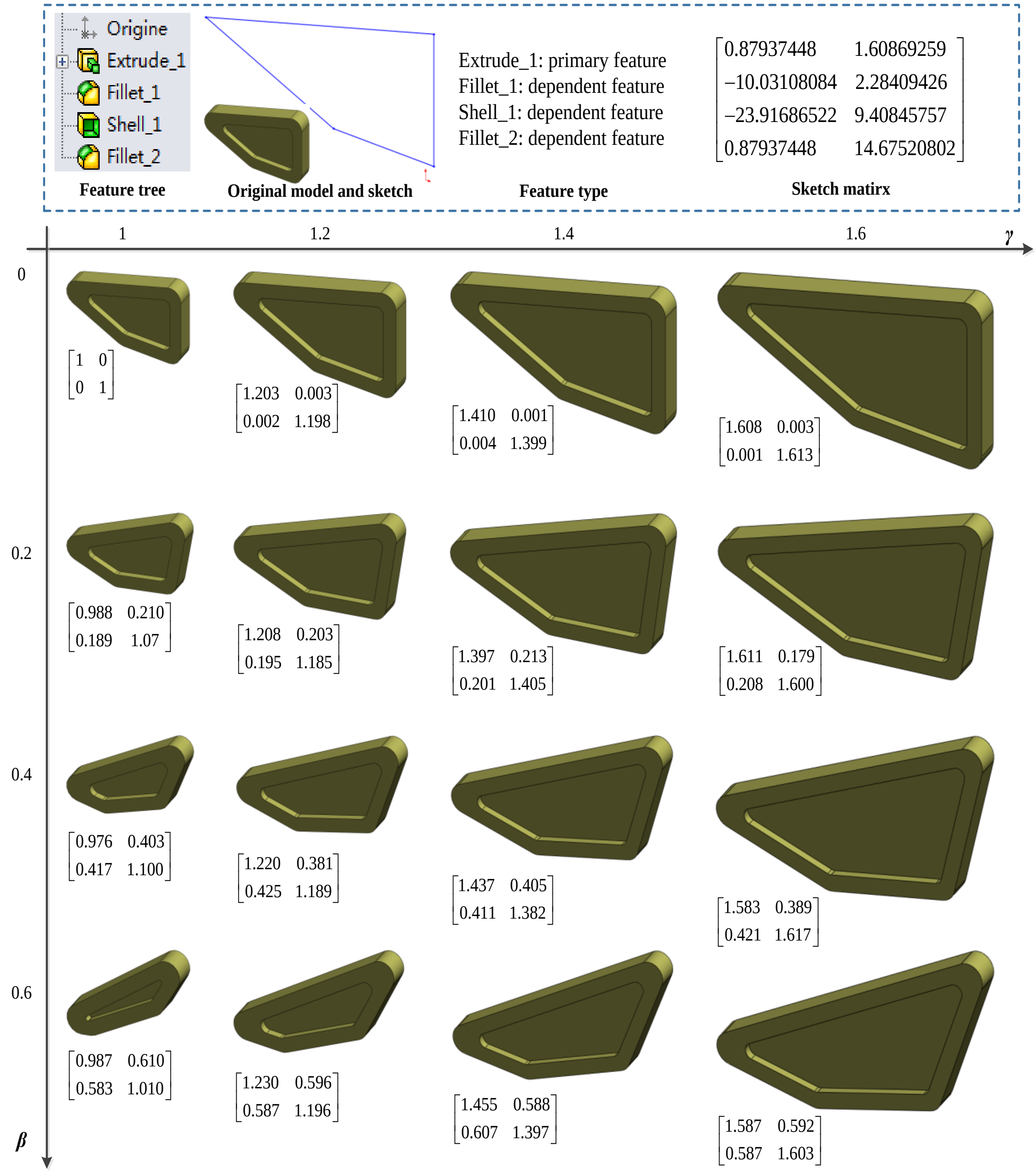


Figure 8: A sample of*A'* based encryption.

In order to improve the randomness of *A'* to improve the security of the encryption and maintain the parameterization of *A'* to support the self-adaptivity of the encryption, the elements of *A'* should be random within a small range, and *A'* can be re-written to be *A''* as Representation (9).

$$A''=\begin{pmatrix}1+\mu^n b_{11} & \mu^n c_{12} & \cdots & \mu^n c_{1m}\\ \mu^n c_{21} & 1+\mu^n b_{22} & \cdots & \mu^n c_{2m}\\ \vdots & \vdots & \ddots & \vdots\\ \mu^n c_{m1} & \mu^n c_{m2} & \cdots & 1+\mu^n b_{mm}\end{pmatrix},\begin{cases}b_{ii}=Random(0.95\alpha,1.05\alpha)\\ c_{ij}=Random(0.95\beta,1.05\beta)\end{cases},\begin{cases}m=2\,||\,m=3\\ 0<\mu<1\\ n\geq 0\\ \alpha>-1\end{cases},|A''|\neq 0 \quad (9)$$

Based on *A''*, the encryption is self-adaptive, owing to the *A''* is approaching to identity matrix based on the increasing of *n*. As thus the encrypted feature is approaching to the original feature. The above conclusion can be proved below.

Definitions:

Let F denote a feature, and F' be the encrypted feature from F by applying A''

Let S denote a sketch and S∈F, and P denote a vertex of S and P is ($p_1, p_2, ...p_m$)

Let S' denote the encrypted result of S based on A''

Let P' denote the encrypted result of P based on A''

Let Encryption Distance ED=Distance(P, P') be the distance between vertex P and vertex P'

Let"→" stand for approaching

Step1:*Prove [∀P∈F, if(P'→P) then F'→F]*

∵∀S∈F, if(S'→S) then F'→F

∵∀P∈S, if(P'→P) then S'→S

∴∀P∈F, if(P'→P) then F'→F

Step2:*Prove [if(n→+∞) then P'→P]*

Assumption：$n_2>n_1$, A_{n1}''denotes n in A'' is given the value n_1, A_{n2}''denotes n in A'' is given the value n_2;

∵$P'_1=P\cdot A_{n1}''$

$\therefore ED^2(P, P'_1)=Distance(P, P'_1)= \mu^{2n_1}\sum_{i=1}^{m}\left(b_{ii}p_i+\sum_{j=1}^{m,j\neq i}c_{ji}p_j\right)^2$

$\because P'_2=P\cdot A_{n2}''$

$\therefore ED^2(P, P'_2)=Distance(P, P'_2)= \mu^{2n_2}\sum_{i=1}^{m}\left(b_{ii}p_i+\sum_{j=1}^{m,j\neq i}c_{ji}p_j\right)^2$

$\because 0<\mu<1$ *and* $n_2>n_1$

$\therefore ED^2(P, P'_2)- ED^2(P, P'_1)= (\mu^{2n_2}-\mu^{2n_1})\sum_{i=1}^{m}\left(b_{ii}p_i+\sum_{j=1}^{m,j\neq i}c_{ji}p_j\right)^2<0$

$\therefore$ *P'₂ is closer to P than P'₁*

$\therefore$ *if(n→+∞) then P'→P*

Step3:*Prove [if(n=+∞) then P'=P]*

$\because \lim_{x\to\infty}\mu^n b_{ii}=0 \; and \; \lim_{x\to\infty}\mu^n c_{ij}=0$

$$\therefore A''=\begin{bmatrix}1 & 0 & K & 0\\ 0 & 1 & K & 0\\ M & M & O & M\\ 0 & 0 & L & 1\end{bmatrix}$$

$\therefore P'=P\cdot A''=P$

Conclusion: With an increased *n*, *A''* will approach to an identity matrix, and the encrypted feature will also approach to its original feature.

A'' based encryption algorithm for features is given as follows *(pseudo code)*.

_Feature_encryption (p, f, γ, β,μ)_

```
1.  P is a model;
2.  f is the feature needed to be encrypted;
3.  Generate the A'' based on the γ, β and μ
4.  Generate all the sketch_matrices of f: S_set{ } ;
```

```
5.  While (S_set{}≠NULL)
6.  {
7.     Get a sketch_matrix from $S_{nm}$_set{}: s;
8.     Encrypt the s based on the A”
9.     If P is invalid
10.    { Decrypt the s;
11.       Adjust the A” by n++;
12.       Goto 8;
13.    }
14.  }
15. Return (A”);
```

Due to the *A’’* based encryption is self-adaptive, after the encryption, the encrypted part is valid. Moreover, the parameter of γ controls the size scale, the parameter of β controls the deformation scale and the parameter of μ controls the efficiency of encryption, their values can be decided according to the user’s requirement.

An example of feature encryption is given in Figure 9, and it can be found that, the encrypted part is invalid until the value of *n* increases to 5.

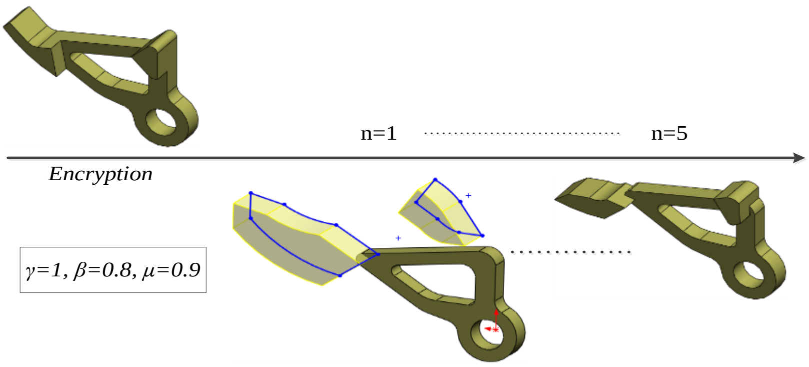


Figure 9: An example of feature encryption.

3.3Encryption algorithm for parts

After the classification of features, EFs of a part should be encrypted. The feature encryption will conduct according to the creation order of features based on the FDG. The encryption of part is detailed as the following algorithm of ***Part_encryption( )****(pseudo code)*.

```
Part_encryption ( Part, Assembly, $\gamma$, $\beta$, $\mu$ )

Part is the original part;
Assembly is the original assembly;
Create the FDG of the Part
Get the feature number from the FDG: F_n;
DF{ }=DF_recognition(Part, Assembly)
Initiate a XML file: Key_file;
i=1;
while (i<F_n)
{
    Encrypt the ith features in Part : f_i ;
    Retrieve the constraints information of the f_i and record it into the Key_file;
    A=Feature_encryption(Part, f_i, $\gamma$, $\beta$, $\mu$);
    Record A into the Key_file;
    i++
}
```

As illustrated in Figure 10, all the parts of the saddle are encrypted based on the encryption algorithm for parts.

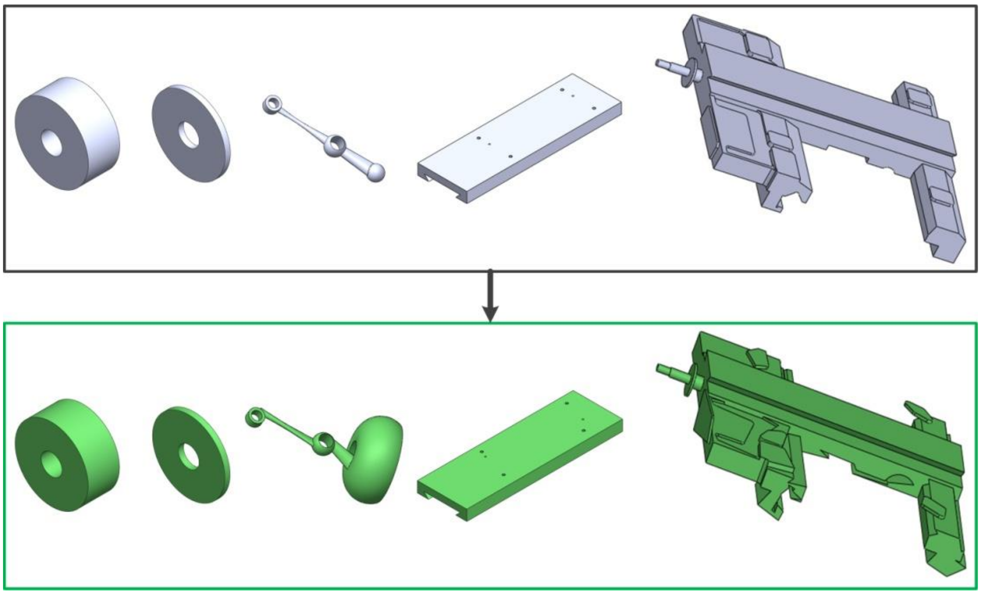


Figure 10: Encryption of each part in the saddle.

3.4 Assembly of encrypted parts

Based on the algorithm of ***Part_encryption( ),*** the encryption of an assembly can be conducted by the encryption of its parts. To illustrate encryption for an assembly model, as illustrated in Figure 11, it is the final encrypting result of the saddle.

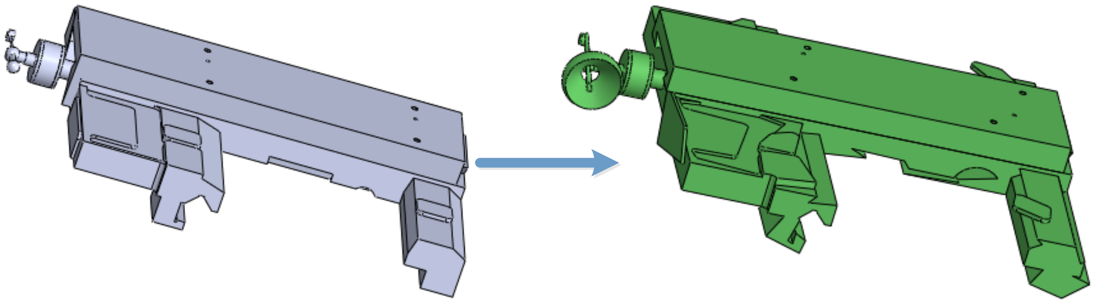


Figure 11: Encrypting of the saddle.

The encrypted assembly and encrypting key (XML files) will be stored separately. If a collaborator requests an assembly for sharing, the encrypted assembly can be authorized by the assembly owner. According to the requirement of the collaborator, the shared parts of the assembly can be designated by the assembly owner. Then, the shared parts will be decrypted based on the related keys, and finally, the decrypted and encrypted parts of the assembly will be assembled to a secure assembly sent to the collaborator.

3.5 Discussion

Algorithm security analysis: Firstly, According to the value ranges of α and β, the space of encryption keys have 10^{3m^2} possibility at least, so that the crack based on the testing of possible encryption keys is therefore computationally difficult. Secondly, if somebody would try to decipher an encrypted CAD model by testing all the possible encryption keys, he or she may get a group of valid cracking results. It is impossible to tell which one is correct. Thirdly, as the encryption transformation matrix is not a periodic matrix, the existing main attack methods for matrix based encryption are invalid (such as the chosen cipher-text attack, plain-text attack and so on).

Time complexity analysis: The time complexity of the algorithms are analyzed below:

- Encryption algorithm of a feature, Feature_encryption( ): T(n)=O(n^2)
- Encryption algorithm of part models, Part_encryption( ): T(n)=O(n^3)
- Encryption algorithm of assemblies, Assembly of encrypted parts: T(n)= O(n^4)

It is can be conclude that, the encryption algorithm of assemblies has low time consumption.

Validity analysis: Firstly, the content based encryption is adopted in this paper and the assembly features are maintained, which can be used for the assembling after the encryption. As thus, after the encryption, the assemblies are still valid. Because all the parts in the assemblies are still there, the external dimensions is feasible. Secondly, the encrypted features and parts can be designated by the model owner flexibly, so that, the encryption of assemblies is flexibly. Finally, based on the encryption of related features, all the confidential information is hidden. Above all, the encryption algorithm of assemblies presented in this paper can satisfy the collaboration requirements on secure sharing of assemblies.

4. Case Study

A real example (provided by the SolidWorks 2012) is given below to validate the approach. In this case, the machine is collaboratively designed by three sites. The task of Site 1 is motor head, the task of Site 2 is machine bed, and the task of the Site 3 is machine base. The Site 1 requests the two assemblies from the other two sites for assembly analysis, as shown in Figure 12. The machine bed is a key assembly which needs to be protected during sharing and interoperation, and the machine base is public.

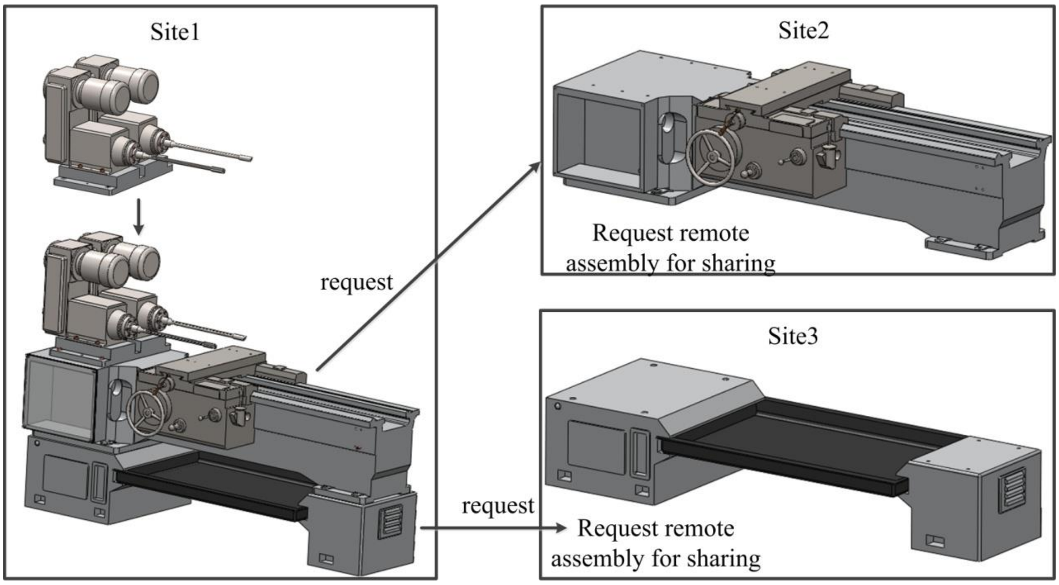

Site1
request
Site2
Request remote assembly for sharing
Site3
request
Request remote assembly for sharing

Figure 12: Collaboratively design of the machine.

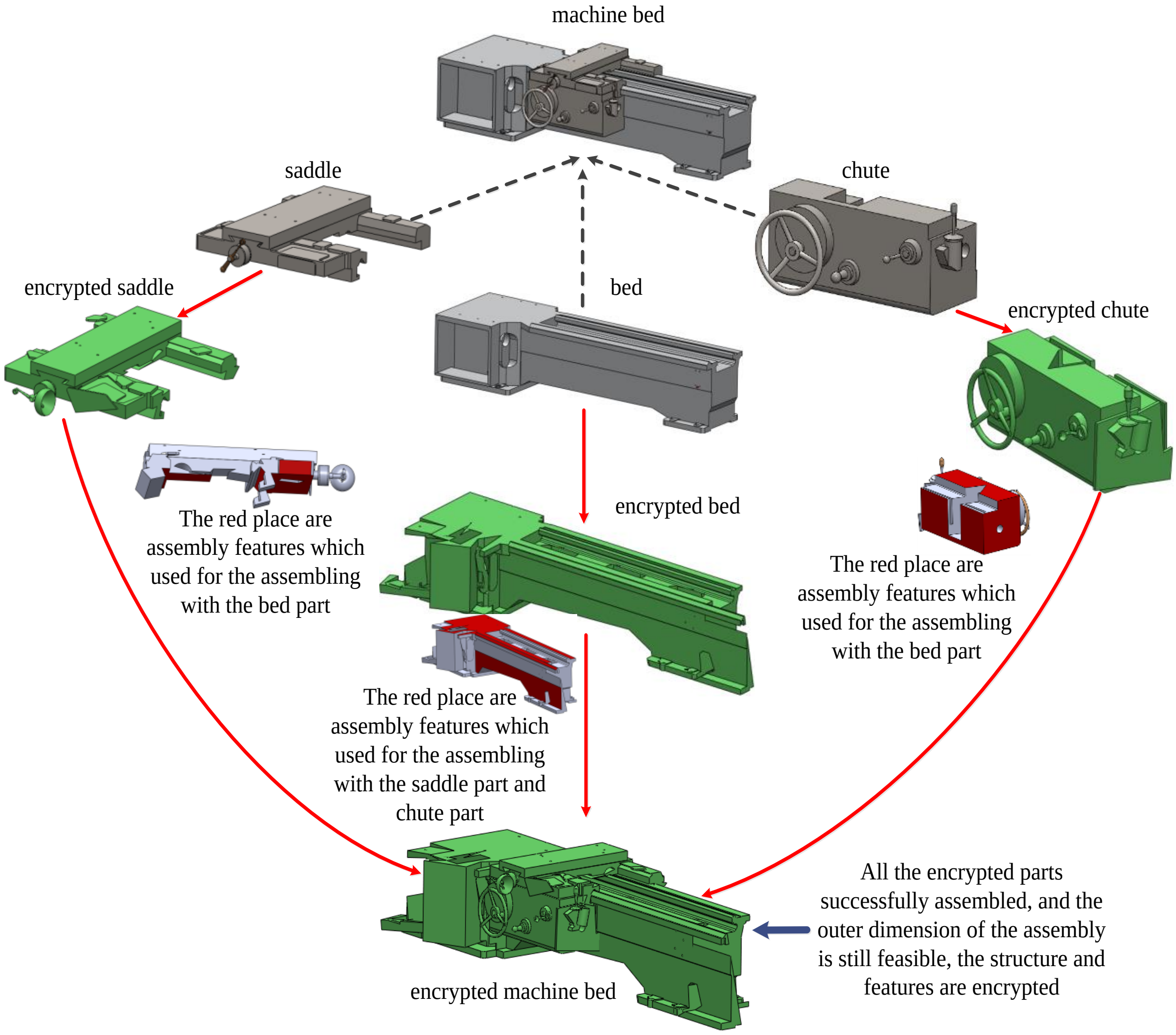

machine bed
saddle
chute
bed
encrypted saddle
encrypted chute
encrypted bed
The red place are assembly features which used for the assembling with the bed part
The red place are assembly features which used for the assembling with the bed part
The red place are assembly features which used for the assembling with the saddle part and chute part
All the encrypted parts successfully assembled, and the outer dimension of the assembly is still feasible, the structure and features are encrypted
encrypted machine bed

Figure 13: Encryption of the machine bed.

Step1: Encryption of the machine bed

The machine bed needs to be encrypted for security. The encryption processes of saddle part and bed part have been described in detail in Sections 3 and 4 (as shown in Figure 3, Figure 4, Figure 5, Figure 6, Figure 10 and Figure 11). Figure 13 shows that the machine bed is made up of two sub-assemblies (saddle and chute) and a part (bed), the grey models are the original models and the green models are encrypted models. In the encrypted models, the models shows the maintained topological entities (red faces) used for assembling. The final encrypted machine bed is totally different from the original machine bed, the information of parts and structure has been protected effectively.

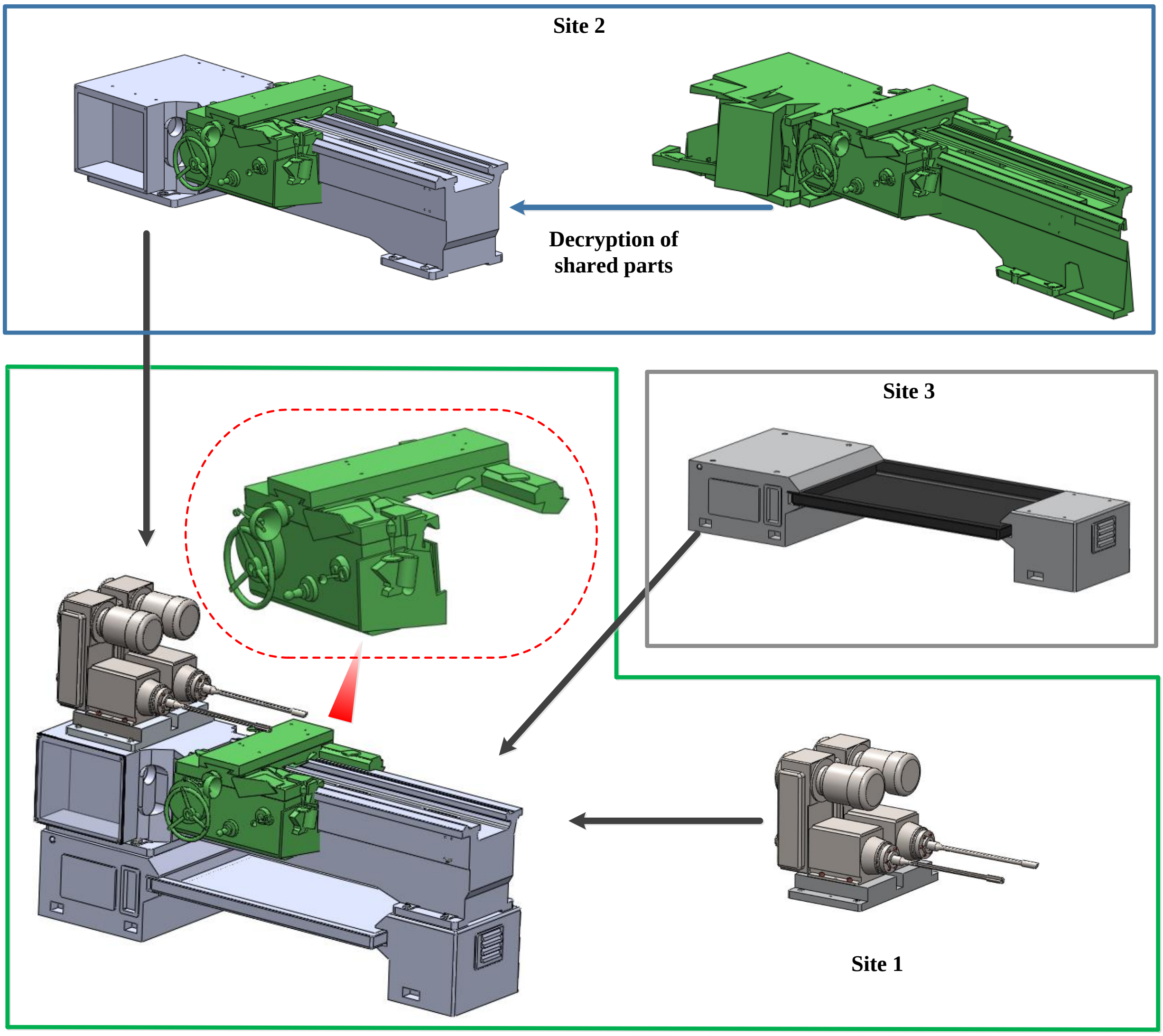


Figure 14: Secure sharing of the machine bed.

Step2: Secure sharing of the machine bed

When the Site 1requests the machine bed and the machine base for assembly analysis, the bed is authorized by the assembly owner for the sharing of machine bed. As shown in Figure 14, the encrypted machine bed is authorized to a secure assembly. In the secure assembly, the saddle and chute are still encrypted, but the bed is decrypted and it can be interoperated flexibly. Moreover, the structure of the machine and the information of the saddle and chute are protected effectively.

5. Conclusions

In order to minimize the secure risk for an assembly model for collaboration, an innovative encryption approach is presented in this paper. In the approach, to satisfy assembly constraints, assembly relations and related features should be maintained after the encryption of the parts in the assembly model. The approach consists of three algorithms. First, a classification algorithm has been developed to identify assembly features, dependent features for the assembly and features for encryption. Second, an encryption algorithm for features in apart based on an encryption transformation matrix has been designed. Based on the above algorithms, an encryption algorithm for parts has been devised to encrypt features in parts and ensure that the encrypted parts are still assembliable. The research innovations include:

(1) The approach is innovative to encrypt features in an assembly model. The feature based encryption mechanism will be easily integrated into the main-stream CAD systems and to meet various users' needs for selected feature encryption during collaboration. The validity of an assembly model, the geometry and structure of the assembly model after encryption are maintained. As such, the encryption mechanism will improve users' interoperability during collaboration;

(2) The approach provides an innovative parametric mechanism for encrypting features and parts in an assembly model. This parametrically controllable mechanism will facilitate users for encryption by adjusting position and size parameters defined in the approach, so as to greatly enhance users' friendliness and flexibility during collaboration.

As the outer dimension is always very important in the product design, it is hope that, after the encryption, the outer dimension is still maintained. How to maintain the outer dimension after the encryption is a key problem of future research.

Acknowledgment

This paper is supported by the National Science Foundation of China (Grant No.61303215) and (Grant No.61373107) and the EU FP7 SMARTER project (FP7-PEOPLE-2013-IAPP-610675).

References

[1] Q. Huang, H. Nouri, K. Xu, et al. Statistical predictive modeling and compensation of geometric deviations of three-dimensional printed products.Journal of Manufacturing Science and Engineering.,2014, 136, pp.061008.1-061008.10.

[2] S. Hauck,S. Knol. Data security for web-based CAD. Design Automation Conference., San Francisco, 1998, pp.788-793.

[3] WD. Li, J. Mehnen.Cloud Manufacturing (Springer Series in Advanced Manufacturing. Springer., 2013.

[4] XT. Cai, XX. Li, FZ. He, et al. Flexible Concurrency Control for Legacy CAD to Construct Collaborative CAD Environment. Journal of Advanced Mechanical Design, Systems, and Manufacturing.,2012, 3(6), pp.324-339.

[5] Y. Zeng, L. Wang,X. Deng, et al.Secure collaboration in global design and supply chain environment: Problem analysis and literature review. Computers in Industry., 2012, 63, pp.545-556.

[6] J. Stjepandić, H. Liese, A.J.C. Intellectual Property Protection. Chapter in Concurrent Engineering in the 21st Century. 2015, pp 521-551.

[7] R. Conway, W. Maxwell, H. Morgan. On the implementation of security measures in information system. Communications of the ACM., 1972, 15(4), pp.211-220.

[8] R. Sandhu, E. Coyne, H, Feinstein.Role-based access control models. IEEE Computer., 1996, 29(2), pp.38-47.

[9] S. Oh, S. Park.Task-Role-Based access control model. Information System., 2003, 28(6), pp.533-562.

[10] J. Park, R. Sandhu. Towards usage control models : Beyond traditional access control.Proceedingsof the 7th ACM SympOn Access Control Models and Technologies., 2002,California, pp.57-64.

[11] J. Park, R.Sandhu. The UCONABC usage control model.ACM Transaction on Information andSystem Security., 2004,7(1), pp.128-174.

[12] A. Hoeven, OT. Bosch, R. Leuken, et al.A flexible access control mechanism for CAD frameworks.Proceedings of the conference on European Design Automation., 1994, Los Alamito, pp.188-193.

[13] G. Stevens, V. Wulf. A new dimension in access control: studying maintenance engineering across organizational boundaries. Proceedings of 2002 ACM conference on CSCW., 2002, pp.196-205.

[14] CD. Cera, T. Kim, I. Braude, et al. Role-Based Viewing for secure collaborative modeling, Pennsylvania: Technical Report DU-CS-03-04. Drexel University, Department of Computer Science, 2003.

[15] KK. Leong, KM. Yu, WB. Lee. A security model for distributed product data management system. Computers in Industry., 2003,50, pp.179-193.

[16] B. Adrianand B. Steve. An access control framework for multi-user collaborative environment.Proceedings of the international ACM SIGGROUP conference on supporting group work., 1999, pp.140-149.

[17] K. Rouibah. and S. Ould-Ali. Dynamic data sharing and security in a collaborative product definition management system. Robotics and Computer-Integrated Manufacturing., 2007, 23, pp.217-233.

[18] HB. Chang, KK. Kim, YD.Kim. The research of security system for sharing engineering drawings. IEEE Computer Society. Washington, USA, 2007, pp.319-322.

[19] LH. Yao, J. Shao, GQ. Sheng, et al. Research on a security model of data in computer supported collaborative design integrated with PDM system. IITA 2007: Workshop on Intelligent Information Technology Application., 2007, pp.91-94.

[20] HB. Chang, KK. Kim, YD Kim. The development of security system for sharing CAD drawings in U-environment. Computing and Informatics., 2008, 27(5), pp.731-741.

[21] C. Speiera, JM.Whippleb,DJ. Clossc, et al. Global supply chain design considerations: Mitigating product safety and security risks. Journal of Operations Management., 2011, 29, pp.721-736.

[22] H. Xiangand M. Li.The research of network security mechanism based collaborative design. Advanced Design Technology., 2012, 421, pp.406-409.

[23] CD. Cera, I.Braude, T.Kim, et al.Hierarchical role-based viewing for multilevel information security in collaborative CAD. Journal of Computer and Information Science in Engineering., 2006,1(6), pp.2-10.

[24] T. Kim, CD. Cera, WC. Regli, et al.Multi-Level modeling and access control for data sharing in collaborative design. Advanced Engineering Informatics., 2006, 20, pp.47-57.

[25] CH. Chu, YH. Chan, PH. Wu. 3D streaming based on multi-LOD models for networked collaborative design. Computers in Industry., 2008, 59, pp.863-872.

[26] CH. Chu, PH. Wu, YC. Hsu. Multi-agent collaborative3Ddesignwithgeometricmodel at differentlevelsofdetail. Robotics and Computer-Integrated Manufacturing., 2009, 25, pp.334-347.

[27] S. Li. and M. Mirhosseini. A matrix-based modularization approach for supporting secure collaboration inparametric design.Computers in Industry., 2012, 63,pp.619-631.

[28] S. Kanai, D. Iyoda, Y. Endo, et. al. Appearance preservingsimplification of 3D CAD model with large-scale assembly structures. Int JInteract Des Manuf., 2012, 6(3), pp139–54.

[29] JF Yu, H Xiao , J Zhang, et.al. CAD Model Simplification for Assembly Field.The International Journal of Advanced Manufacturing Technology., 2013, 9(68), pp 2335-2347

[30] Y Kang, BC Kim, D Mun, S Han. Method to simplify ship outfitting and offshore plant equipment three-dimensional (3-D) computer-aided design (CAD) data for construction of an equipment catalog.Journal of Marine Science and Technology., 2014, 2(19), pp 185-196

[31] S Kwon, BC Kim, D Mun, S Han. Simplification of feature-based 3D CAD assembly data of ship andoffshore equipment using quantitative evaluation metrics. Computer-Aided Design., 2015, 59, pp140-154

[32] KN. Naveenand JN Thomas.Flexible optical encryption with multiple users and multiple security levels.Optics Communications., 2011,284, pp735-739.

[33] XT. Cai, FZ. He, WD. Li,et..al. Encryption Based Partial Sharing of CAD Models. Integrated Computer-Aided Engineering., 2015, 22, 243-260.

[34] XT. Cai, WD. Li, FZ. He, et.al. Customized Encryption of Computer Aided Design Models for Collaboration in Cloud Manufacturing Environment. Journal of Manufacturing Science and Engineering., 2015, 137, pp040905-1- 040905-10.