



Regular patterns of repeated elements in CAD assembly model retrieval

Katia Lupinetti, Lisa Chiang, Franca Giannini, Marina Monti & Jean-Philippe Pernot

To cite this article: Katia Lupinetti, Lisa Chiang, Franca Giannini, Marina Monti & Jean-Philippe Pernot (2017) Regular patterns of repeated elements in CAD assembly model retrieval, Computer-Aided Design and Applications, 14:4, 516-525, DOI: [10.1080/16864360.2016.1257193](https://doi.org/10.1080/16864360.2016.1257193)

To link to this article: <http://dx.doi.org/10.1080/16864360.2016.1257193>



Published online: 06 Mar 2017.



Submit your article to this journal [↗](#)



View related articles [↗](#)



View Crossmark data [↗](#)

Regular patterns of repeated elements in CAD assembly model retrieval

Katia Lupinetti ^{a,b}, Lisa Chiang ^a, Franca Giannini ^a, Marina Monti ^a and Jean-Philippe Pernot ^b

^aCNR-IMATI, Genova, Italy; ^bArts et Métiers ParisTech, LSIS UMR CNRS 7296, Aix En Provence, France

ABSTRACT

A multitude of online repositories and a large amount of specific CAD model databases are currently available. Therefore, it is crucial to have access to these data in an easy and in a multi-perspective manner according to multiple access keys, not only in terms of annotation data or shape similarity, but also in terms of specific characteristics. In this perspective, this paper proposes a multi-level approach for CAD assembly model retrieval, which exploits assembly specific information related not only to the shape of the constituting components but also to peculiar assembly information, such as kinematic joints and component arrangements. In many cases, most of this information is not explicitly stored, thus tools for its extraction must be provided. In this paper, we focus on the detection of regular patterns of repeated elements in CAD assemblies and on their exploitation for the browsing and retrieval of assembly models.

KEYWORDS

CAD assembly retrieval;
regular pattern detection;
assembly descriptor

1. Introduction

The extensive use of CAX tools in industry leads to impressive collections of CAD models stored both in legacy and in online databases. When designing new products, designers usually reuse existing models to speed up the design process [14]. However, nowadays, databases are so large that designers struggle to retrieve product data such as drawings, simulation models and results, technical reports and CAD models. So far, the most common solutions for the retrieval of assembly models rely on the use of Product Data Management (PDM) systems. PDM systems ease the organization and management of product data by enabling designers to structure the product information and to add textual metadata to describe the content. Although PDM systems efficiently manage text-based queries, they do not always fulfil designers' needs. Indeed, sometimes designers prefer to query the system for CAD models geometrically or structurally similar to an existing one (single part or assembly). Content-based algorithms enhanced by additional geometric characteristics are an alternative to extend text-based search capabilities to retrieve 3D models of parts. A wide literature is available and some commercial systems are appearing providing shape-based model retrieval. Cardone *et al.* [3] identify several scenarios in which the search for shape similarity plays a key role, while Jayanti *et al.* [9] and Tangelder *et al.* [13] provide a complete overview of 3D shape descriptors.

However, these descriptors focus solely on the shape of a single component.

An effective assembly search cannot be limited to simple shape comparison among components, but requires also information that is implicitly encoded in the CAD models, e.g. the relationships and the joint constraints. This implicit information must be made explicit to become usable. Deshmunk *et al.* [6] propose various scenarios and some capabilities for assembly searching. Hu *et al.* [8] propose a tool to retrieve assemblies by representing them in a watertight polygon mesh. A vector space descriptor is used to decompose an assembly into different meshes corresponding to the parts of the assembly. Identical parts are merged and a weight based on the number of occurrences is attached to each part in the vector. Nevertheless, this descriptor ignores the relative positions of parts and their constraints. Moreover, the retrieval method is weak in local matching. Miura and Kanai [11] extend their assembly model by including structural information and other useful data, e.g. contact and interference connections as well as geometric constraints. However, some information must be specified by the user. This descriptor has two main limitations. First, it does not consider high-level information, such as kinematic pairs or general assembly shape. Secondly, it lacks various search criteria that might be of interest to the designers. A more complete assembly descriptor is proposed by Chen *et al.* [4] and relies on the

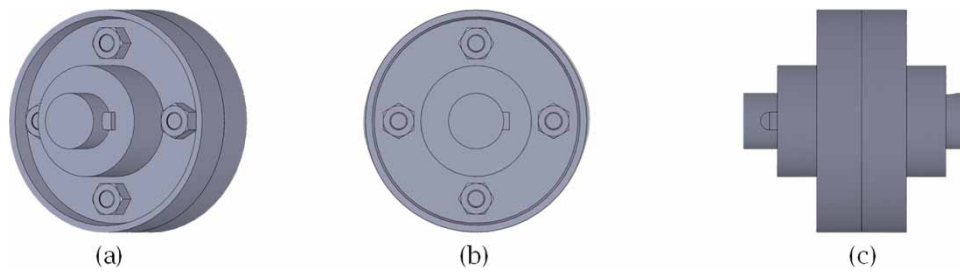


Figure 1. Assembly of two main parts with circular patterns of screws and bolts: (a) 3D view, (b) Front view, (c) Right view.

product structure and the relationships between the different parts of the assembly rather than dealing with the shape of the whole assembly. The assembly descriptor takes into account different information levels including the topological structure, the relationships between the components of the assembly, as well as the geometric information. Additionally, the descriptor enables designers to ask rough and incomplete queries that make the search approach more flexible.

We propose a framework for the retrieval of globally and/or partially similar assembly models according to different user-specified search criteria [10]. It is based on an assembly descriptor, called Enriched Assembly Model (EAM), organized in several layers that enable multi-level queries. Dedicated modules are foreseen to supply all the information to be encoded in the model through the analysis of the geometry and the layout of the CAD assembly model. Similarly to the assembly descriptors presented in [4] and [6], the proposed EAM is able to support user requests at different specification levels. Differently than [6], it does not require the user to add manually some information. Moreover, differently than [4], the mapping algorithm is not limited to the identification of assembly models with the same structure in terms of sub-assemblies; indeed, same products may be structured differently depending on the designer's purposes.

Among the various information supported by the proposed EAM, this paper focuses on the identification of regular patterns formed by repeated elements in an assembly and on the exploitation of this information for the assembly retrieval. Fig. 1 shows an example of an assembly model with circular patterns of screws, bolts and nuts. To the best of our knowledge, current solutions for assembly model retrieval are not able to exploit fully this kind of information for the identification of similarities.

The rest of the paper is organized as follows. Section 2 introduces the Enriched Assembly Model, illustrating its main characteristics and content. Section 3 describes the developed pattern recognition module. In section 4, the retrieval method is presented, focusing on its capability to satisfy a wide range of combination of search criteria to

better adhere to the user needs. Section 5 presents some of the obtained results for assembly retrieval. The conclusion section ends the paper summarizing the main achievements and future work.

2. The enriched assembly model

Retrieval purposes require combination of different kind of information to be efficient and address user's aims properly. To increase retrieval performances, the EAM encodes information at four main layers – statistic, structure, interface and shape, as illustrated in Fig. 2. The various layers aim at characterizing the assembly at different levels of detail to allow scalable search queries.

The statistic layer contains numerical attributes to allow a quick search and filtering. It includes three categories: assembly statistics, part statistics and interface statistics. The first include figures related to the overall assembly:

- number of sub-assemblies,
- number of principal parts,
- number of fasteners,
- number of thin parts,
- number of patterns of a specific type.

The part statistics indicate the percentage of a specific type of surface and the number of maximal faces of a specific type (planar, cylindrical, conical, toroidal and free form); while the interface statistics comprise the number of a specific joint type and the number of elements in contact for a specific type of contact.

The structural layer encodes the hierarchical assembly structure as specified by the designer plus high-level information on part arrangement, i.e. regular patterns of repeated parts. We use a tree structure to represent this layer. The root corresponds to the entire assembly model, the intermediate nodes are associated with the sub-assemblies and the leaves characterize the parts. In this representation, we attach attributes to the entire assembly as well as to its sub-assemblies to encode the high-level information specifying arrangements of repeated parts.

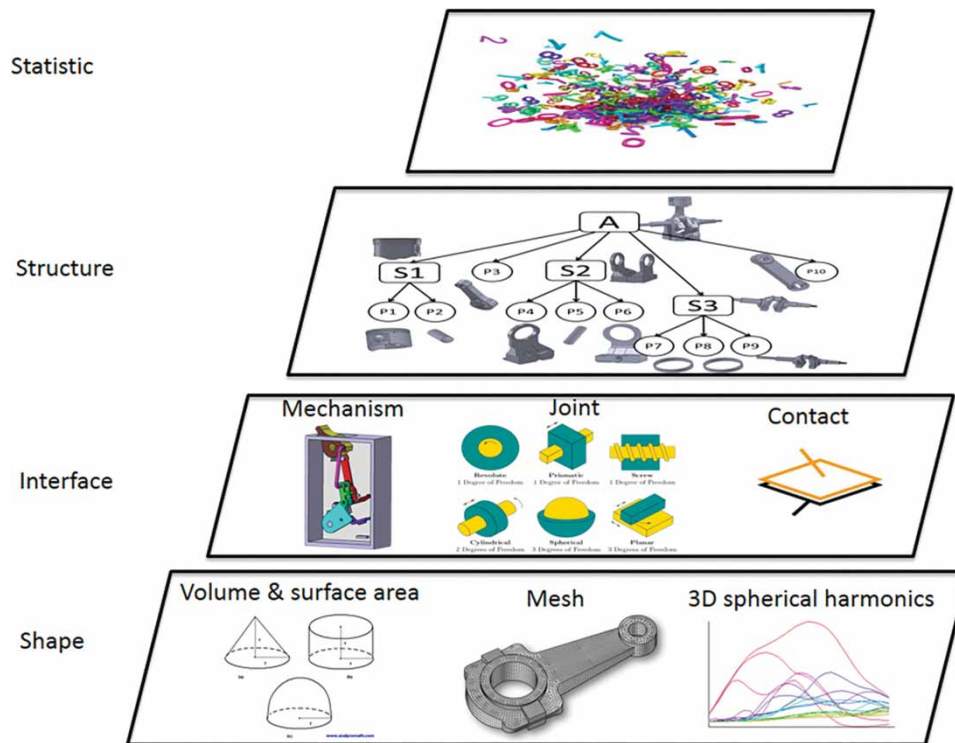


Figure 2. The multi-layered Enriched Assembly Model.

The attributes specify all the patterns contained in the assembly (or sub-assemblies). For each pattern is indicated the type (linear translational, circular translational, circular rotational and reflectional) and the involved parts (as links).

The interface layer provides a hierarchical specification of the interfaces among the parts in the assembly. It has four levels (mechanism, synthesis, joint and contact) to allow both rough and precise retrievals. The highest level indicates the final motions between two main parts defined in the entire assembly. The synthesis level groups together parts with the same functionality, as the parts that compose bearing, coupling or shaft. The joint level contains information on the motion between two components considering all their contacts, while the contact level is the lowest one and it represents the degree of freedom between all the faces of the parts involved in joints.

The shape layer includes various shape descriptors of both the sub-assemblies and their elements to provide shape information at different levels of detail and it is exploited for the retrieval with exact and rough queries with imprecise shapes.

The proposed EAM is a very rich model. The apparent redundancy of the stored information offers the possibility to have scalable queries and filters for result refinement. The complete EAM is computed only for the stored models, whereas for the query model only the data at

the detail level required by the query are computed and exploited for the matching, thus reducing the complexity of the system.

3. The identification of regular patterns of repeated parts

Among the tools that analyze a CAD assembly model to extract the information to be stored in the EAM, here we focus on the method for the identification of regular configurations in a set of Repeated Parts (RPs) in an assembly.

RPs are identified in the assembly model as multiple instances of the same object or by considering parts having identical values in the part statistics data layer and same volume and surface area in the shape descriptors layer. The parts identified through the statistic data and shape descriptors may be similar but not identical: successive geometric checking will discard non identical parts.

The method applies a series of grouping and filtering processes to reduce the complexity and the number of elements on which to perform the symmetry rule detection [5]. The computation is simplified by the consideration that if a set of congruent sub-parts is characterized by a regular arrangement, then also the respective centroids do [5]. Vertices of the model and other characteristic points are used to compute the centroid for every RP.

We focused on symmetric regularities characterized by a constant distance between two centroids of two consecutive RPs. Thus, we group together all the RPs whose centroids are at a constant distance d .

We define a d -adjacency matrix as follows. Let $\{C_i\}$ be a set of points in \mathbb{R}^3 and $d > 0$ a real number, we call d -adjacency matrix the $n \times n$ symmetric matrix M_d such that:

$$M_d(i, j) = \begin{cases} 1 & \text{if } |\text{dist}(C_i, C_j) - d| < \varepsilon \\ 0 & \text{if } |\text{dist}(C_i, C_j) - d| \geq \varepsilon \end{cases} \quad i, j = 0, \dots, n-1 \quad (1)$$

where ε is a tolerance value that is classically set up to 10^{-4} as in many engineering software and dist returns the Euclidean distance between points.

A d -adjacency matrix can be viewed as a network of points in \mathbb{R}^3 each of them connected to one or more points of the network by a straight arc of length d . A list of adjacency matrices at constant distance d is then created, one for each distance d found between the centroids of the RPs. The adjacency matrices are used to quickly identify the sequences of equidistant centroids and then the possible patterns involving as many as possible RPs.

Given a d -adjacency matrix, we call branch of the centroid C_j , a centroid C_k , $k \neq j$, such that $M_d(k, j) = 1$. We will also say that C_j and C_k are connected at distance d . We classify a centroid C_i considering the number s_i of centroids C_k at distance d from C_i in the matrix:

- if $s_i = 1$, C_i is called extreme point;
- if $s_i = 2$, C_i is called simple point;
- if $s_i > 2$, C_i is called multi-branch point.

In the example provided in Fig. 3, points 1, 2, 3 and 11 are multi-branch points, points 0, 9, 13 and 18 are extreme, all the others are simple points.

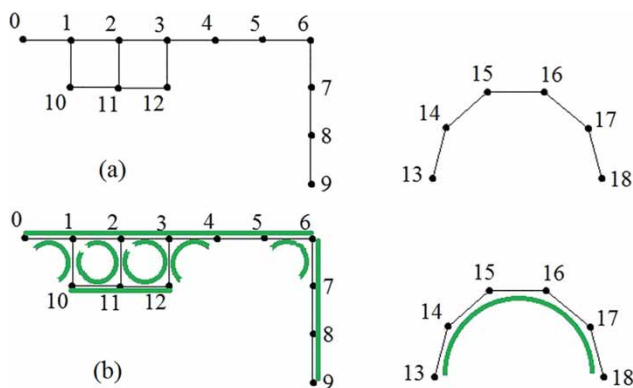


Figure 3. Example of detected paths of centroids in a d -adjacency matrix.

At first, the path detection algorithm aims at identifying all the possible paths in the centroid network represented by a given d -adjacency matrix. These paths correspond to sequences of at least three centroids satisfying specific geometric conditions. The developed method focuses on regular arrangements of RPs whose centroids all lie either on a line or on a circle.

A path is built step by step, by first choosing an initial seed path of three centroids (seed1, seed2, seed3) and, once the type of the path that is going to be built has been established, by adding every time a new centroid to the current path if possible. If the three initial points are aligned, it will be a seed path of type linear; otherwise, it will be a seed path of type circular. In both cases, at first, the attempt of expansion is done in the “seed1 to seed2” direction; when the expansion in this direction is no longer possible then a second expansion attempt is done in the “seed2 to seed1” direction.

The expansion from a seed set ends when the maximum expansion is reached in both the directions. Let us see when the expansion is possible. Let (C_0, \dots, C_{k-1}) , be current path, with $k > 2$, C be the associated curve, “ C_0 to C_1 ” be the considered direction, the expansion in that direction is possible if and only if a branch of C_{k-1} lying on C and different from C_{k-2} exists. If such a point does not exist the expansion in that direction is no more possible. Furthermore, some starting points could lead to many different paths: this is the case of the multi-branch points and of the simple points.

To find all the existing paths it is necessary to explore every branch of the starting point (seed1) and to consider every branch as second point (seed2). In the same way, a second point could have more than one branch and we must consider every possible third point (seed3) to be sure to find all possible paths. In order to have adequate starting seeds, only multi-branch and simple points (if no multi-branch points are present) are considered as seed1.

A path of centroids gives an outline of the RPs placement but it is necessary to verify the correct orientation of the corresponding RPs to assess that the identified path really indicates a regular pattern of repeated components. Currently, this phase of the algorithm is developed for RPs containing exclusively planar and cylindrical faces and for the following types of pattern: linear translational, circular translational, circular rotational and reflection. An example of each type of pattern is reported in Fig. 4.

The algorithm is based on the verification that the entities of the RPs satisfy the same transformation rule of the related centroids. Specifically, for centroids lying on a linear path the candidate pattern is the linear translational, whilst for centroids lying on a circular path, the candidate patterns are the circular translational or

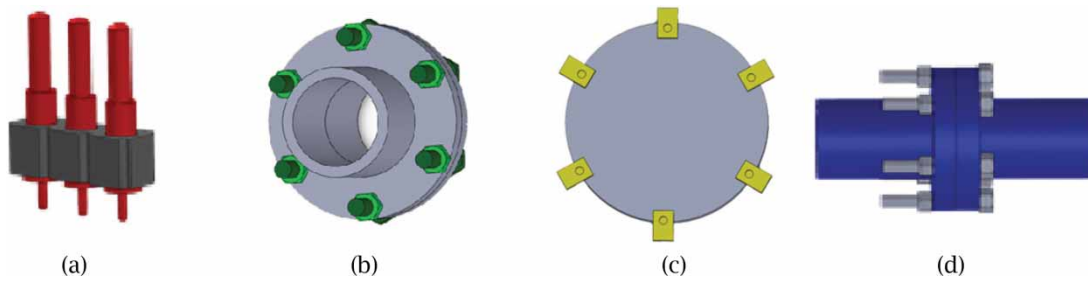


Figure 4. Types of detected patterns: (a) Linear translational, (b) Circular translational, (c) Circular rotational, (d) Reflection.

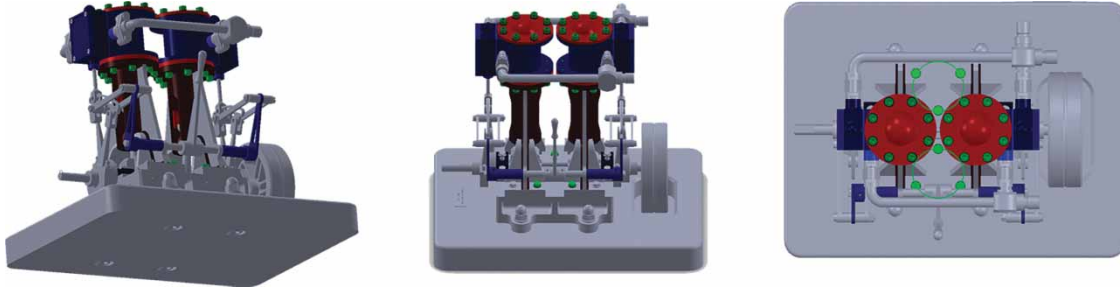


Figure 5. Examples of detected patterns of repeated components in an assembly.

rotational. Reflection patterns are verified only in case of a set of RPs with only two elements.

Therefore, when a path of centroids is identified, for any pair of RPs corresponding to two consecutive centroids in the path, two levels of check are performed. The first check considers the real vertices in the RPs, while the second exploits the surface information of the faces. First, the method verifies if for any vertex in one RP, there is a vertex in the successive RP that is obtained applying to the first vertex the transformation under verification. If the test on the vertices is positive, then a second level of verification is performed on the face orientation. In case of linear translational pattern, for each planar face in the first RP it is verified if there exist a planar face in the successive RP with the same normal. For each cylindrical face, the method checks if there exists a corresponding translated cylindrical face in the second RP by exploiting axis and edge information [5].

Finally, the process analyses the couples of RP not included in any recognized pattern, if any, to verify if they satisfy any regular arrangement (i.e. translation, 180° rotation or reflection).

Fig. 5 shows an example of pattern detection in a complex assembly model where the different nuances of red, green and blue correspond to different linear translation, circular translation and reflection patterns, respectively. The assembly is made up of 355 components including 321 parts (71 distinct) and 34 sub-assemblies (19 distinct).

4. The retrieval system

The tool developed for the retrieval of assembly models according to different similarity criteria grounds on the EAM described in section 2. Thanks to the richness of information included in the EAM, the exploration of the assembly models stored in a repository can be performed using and combining various search keys.

If the user asks for retrieving all the models similar to an existing CAD model, the system automatically generates its complete assembly descriptor. Conversely, to speed up the computation, if the user is looking for models with some specific characteristics, he/she may specify the attributes of interest among those described in the different EAM layers; the system automatically creates an assembly descriptor limited to the specified layers and attributes, which are then used as search criteria during the matching process. Optionally a percentage of allowed variation on the different elements can also be specified to relax the retrieval process. It allows a pre-filtering of the candidate most similar models through the verification of the concerned statistics values. For example, the user may express ranges in which two assemblies may be considered similar, e.g. allowed percentage of different patterns of components or relations, thus the system applies a filtering based on these statistics to reduce the number of models to be compared.

The EAM is represented by an attributed graph, i.e. a graph where nodes and arcs have associated attributes.

This structure allows encoding geometric and topological information. Moreover, it owns invariant property for geometric data and it can be enriched thanks to the use of attributes. In the EAM, the attributes attached to nodes and arcs describe the extracted data to express explicitly assembly information. In this structure a node is associated with the components of the assembly model (entire assembly, sub-assemblies and parts), while the arcs encode different types of relationships (e.g. parent relations, joints, constraints, part arrangements). The attributes for labeling the regular patterns are encoded in the arcs that link repeated parts in the same arrangement. In particular, they characterize the *type of pattern* (linear translational, circular translational, circular rotational and reflection), the *step* (i.e. the distance between each repeated element), the *length* for linear translation pattern or the *pattern center*, its *radius and the angle* for circular translational and rotational pattern. Moreover, the number of each pattern type is added in the statistic layer.

Adopting this representation, if two models have a common feature, then their attributed graphs must have a common sub-graph. The similarity assessment between two EAMs can then be performed by matching their attributed graphs and finding their maximum common subgraph (MCS). The identification of the MCS is a well-known NP-hard problem and among the various

techniques proposed for its solution [2], we chose the detection of the maximal clique of the association graph. The association graph is a support graph that reflects user's query criteria; a node in the association graph corresponds to a pair of compatible nodes in the two attributed graphs according to the specified criteria. Associated arcs connect nodes if they have equivalent relations expressed as arcs connecting the corresponding nodes in the attributed graphs.

A clique is a sub-graph in which for each couple of nodes a connecting arc exists; then, the maximum clique in the association graph corresponds to maximum common sub-graph between the two compared attributed graphs.

A clique can be characterized through the number of its nodes and arcs by the function (2):

$$f(C) = \frac{k(k-1)}{2} - h \quad (2)$$

where k is the number of nodes and h the number of arcs in C .

Since a clique is a graph completely connected, i.e. each node has $k-1$ arcs, thus if $f(C)=0$ then C is a clique. According to this generalization, the MC problem is an optimization problem, which aims to find the minimum of the function (2). Similarly to the work of You and Tsai [17], to find the minimum of $f(C)$ we apply

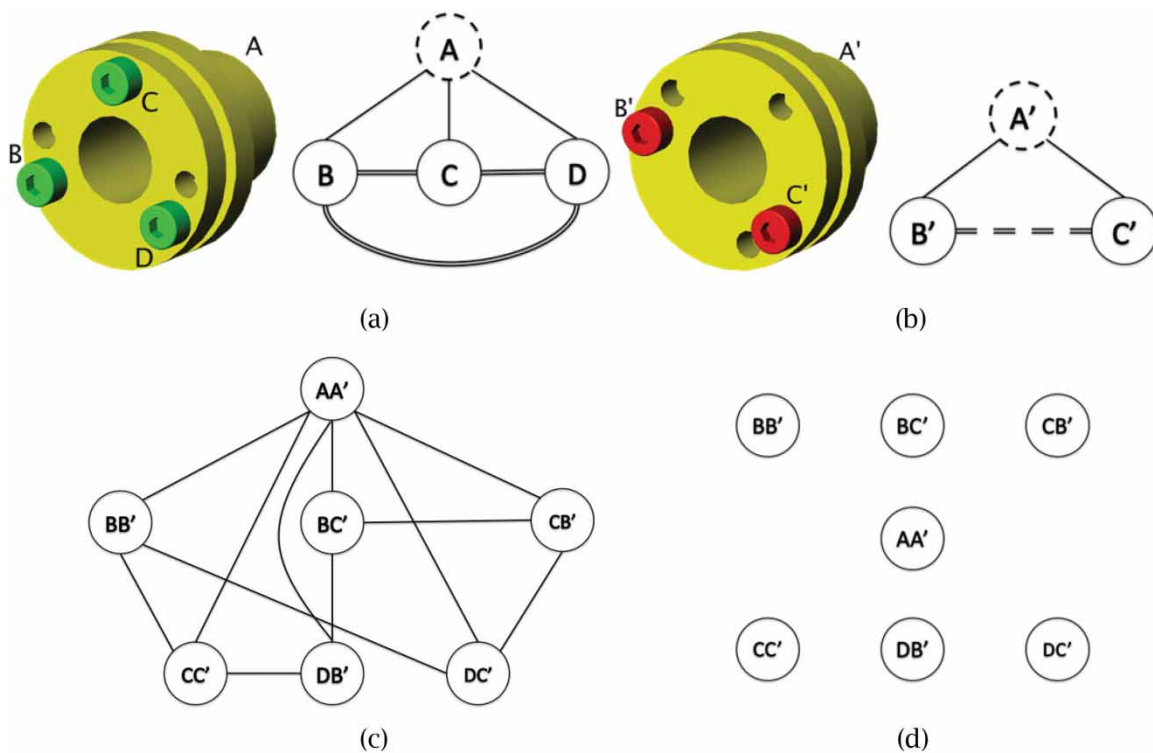


Figure 6. Equivalence between MCS and MC problem: (a) First model and its attributed graph G , (b) Second model and its attributed graph G' , (c) Association graph in case of query for similar joints, (d) Association in case of query for similar joints and same type of pattern.

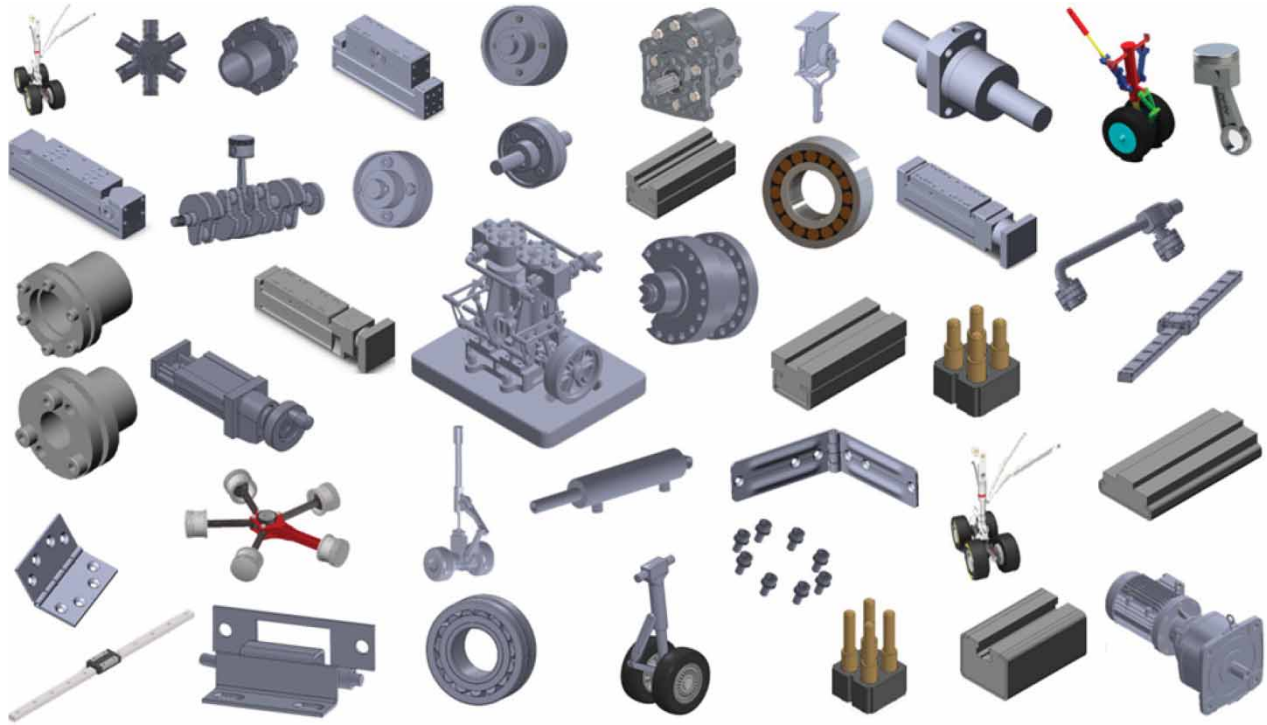


Figure 7. Portion of models in the dataset.

the simulated annealing technique, which is an heuristic approach, originally used in thermodynamic systems, to find good solutions for optimization problems, even in the presence of noisy data. If a clique is present in the association graph, then the two graphs have a common sub-graph, so they match locally at 100%. The measure of the local similarity between two graphs is given by equation (3).

$$\mu = 1 - \frac{f(C)}{k(k-1)/2} \quad (3)$$

On the other hand, to express the global similarity, i.e. the percentage that the maximum common sub-graph covers on the entire graph, we consider the ratio between the number of nodes in the clique and the number of nodes in the comparison model.

To show the association graph creation mechanism, Fig. 6 depicts an example of two assembly models, their EAM attributed graphs (G and G') and two possible association graphs corresponding to two different queries. The nodes in the attributed graphs in Fig. 6(a) and Fig. 6(b) represent the parts in the CAD model and the same line type indicates that the nodes have the same spherical harmonic value (i.e. they correspond to parts with similar shape). The arcs represent the relationships (joint and pattern) between the parts.

More precisely, the model in Fig. 6(a) has a pattern of screws, which is represented with double arcs in its

attributed graph G , while the other arcs characterize a joint of rotation type.

The model in Fig. 6(b) has two screws arranged in a translational pattern. This peculiarity is expressed in the attributed graph by a double dashed arc. Also in this case, the other arcs characterize a joint of rotation type.

In the first query example (see Fig. 6(c)), we suppose that the user looks for assemblies in which parts with similar shape are connected by the same joint relationships, i.e. the two pairs of parts should have the same motion ability, i.e. the same degree of freedom. This means that the two arcs should have the same number of rotation and same number of translation. In this case, two nodes are put together in an association node according to their shape attribute (i.e. if the corresponding parts have similar shape), while the association arcs are added only if the joint arcs (between the related pairs of nodes in the attributed graphs) have the same attributes. Fig. 6(c) shows the resulting association graph which contains six possible cliques: $C1 = \{AA', BB', CC'\}$, $C2 = \{AA', BC', DB'\}$, $C3 = \{AA', CB', DC'\}$, $C4 = \{AA', CC', DB'\}$, $C5 = \{AA', BB', DC'\}$ and $C6 = \{AA', BC', CB'\}$. Each clique represents a possible sub-graph matching between the two attributed graphs G and G' . For our purpose, it is sufficient to extract one of them in the association graph. In this case the two models are locally similar at 100% since the association graph contains a clique, while the similarity global measure is $3/4 = 0.75$.



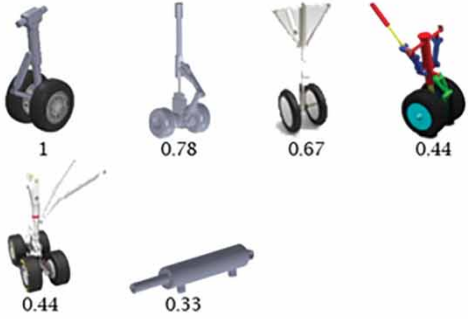

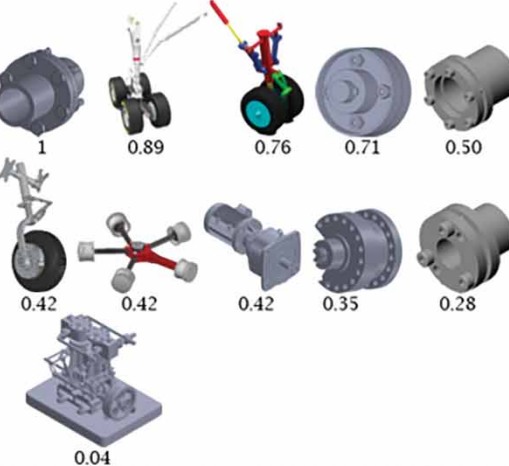
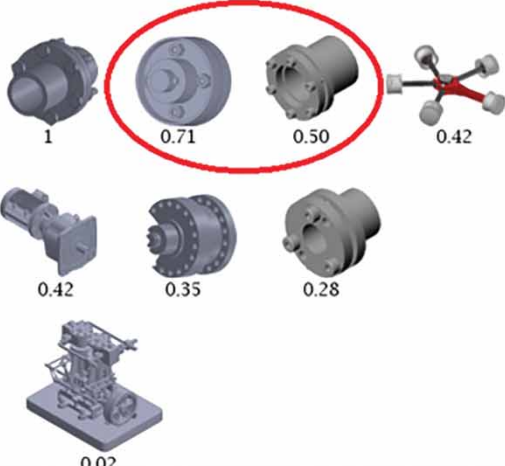

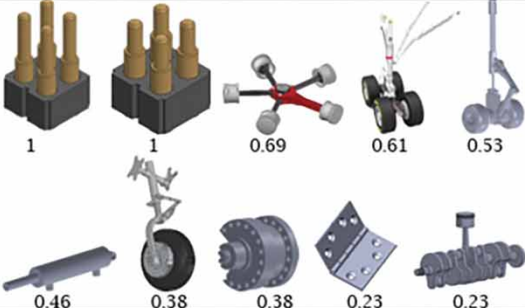
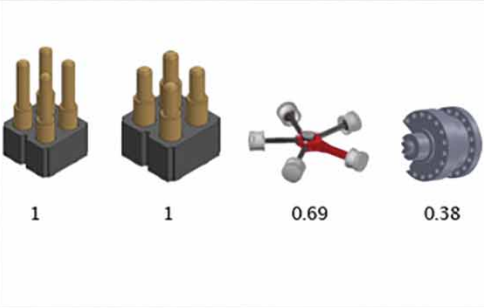
Query model	Retrieved models	
	Joint query	Joint and pattern query
		
		
		

Figure 8. Results for different query models.

In the second example (see Fig. 6(d)), the query is for similar parts with the same type of pattern and linked by same joints, thus pattern arcs are also involved in the computation of the association graph. In this case, no arcs are added in the association graph since there are not patterns of the same type in the two models. Thus no clique is present in the graph in Fig. 6(d) and considering both these criteria the models result different.

5. Results

The retrieval tool based on the EAM is developed as plugin of the commercial CAD system SolidWorks [12] and exploits its Application Programming Interface (API).

Assembly models encoded in STEP format (ISO 10303-203 and ISO 10303-214) are the inputs for the assembly retrieval framework. To validate the developed method, we collected models from public datasets as GrabCAD [7], Tracepart [15], 3D Content Center [18] and Visionair [16]. The dataset contains assemblies of different natures, as linear guides, couplings, landing gears and hinges. Fig. 7 shows a selection of the models included in the dataset that is used to validate the proposed approach.

Fig. 8 illustrates the results obtained with three different query models; the reported measures are evaluated considering the ratio between the number of nodes in the clique and the number of nodes in the query model. For each query model, two different requirements are considered for the search: the first one considers only the type

of joints (i.e. allowed motion between two parts) and the second one considers both the type of joints and the type of pattern of repeated parts of the query model. The first query does not take into account the shape of the entire assembly or the shape of the parts involved in the motion. The use of only this criterion is not sufficient to retrieve significant models in the dataset. The results improve and become more similar to the query model, by adding the request to look for models containing the same type of pattern.

The query model in the first example has two repeated wheels and tires; both of them form a translation pattern. Considering only the rotation joint as criterion for the similarity (first set of retrieved objects) the retrieval tool identifies also several false positive models. Requiring also pattern similarity in the query, we can discard many models that are not similar to the query and a landing gear with a single wheel, without patterns of repeated parts. The last retrieved model is a false positive, since it presents both a rotation joint and a translation pattern made by the two elements at the basis of the main cylindrical body. This result can be avoided by adding to the query other criteria such as the same shape of the elements involved in the patterns.

The second example is a coupling formed by 14 parts with two circular patterns, one of bolts and another of nuts. The search using only the condition on joints is again rather rough retrieving also landing gears and piston models. For the set of retrieved objects in the second column, the query was seeking for same joints and two circular patterns. Also this case shows how the use of the patterns reduces false positives. To better clarify the values of the reported measures, the matching of two models is illustrated deeper. In the first of the circled objects, four screws, four nuts and the two main parts match; the maximum clique has 10 nodes, thus 10 parts in the assembly are matched; since 10 parts are matched on 14 the global measure for it is $10/14 = 0.71$. The second object has 6 screws but not nuts. It verifies the matching with 7 parts and thus its global measure is $7/14 = 0.50$.

Considering the last retrieved object, its global similarity measure is low since the model is composed by a higher number of parts than the query model. The first search considering only the type of joints matches six pairs of screws and nuts similar to the query model (12 parts in all). While, the search for patterns and joints is able to recognize the circular pattern of screws present in the complex model and illustrated in section 3 (Fig. 5). In this case, 6 parts (6 screws of the circular pattern) match with the screws of query model. Having fewer matching parts, the global measure is lower than the previous one.

The third example shows a model with only rotation joints and four pins arranged in a circular pattern. Three different parts compose the single pin, thus the model has three circular patterns. The first set of models presents objects that partially have this characteristic. While the second set of objects is retrieved when searching for models with at least a circular pattern.

6. Conclusions

Tools to retrieve CAD assemblies in massive databases are required. Generally, only a criterion is not sufficient for a proper matching estimation. In this paper, we propose a descriptor, called Enriched Assembly Model (EAM) suitable for the hierarchical matching of CAD assembly models. To be generally applicable, the EAM is built from CAD assembly models in STEP file format and directly encodes some implicit information (e.g. contacts, joints, part arrangements). Various algorithms are being developed to extract and compute the data required for building the EAM information from the CAD assembly models. In this paper, we focus on the method to detect and exploit regular patterns of repeated parts within the assembly. The pattern detection method is limited to consider patterns of equidistant repeated parts with a linear or circular arrangement, which are those prevailing mechanical products. The method may be extended to consider pattern of repeated equidistant part arranged according to other regular curves, e.g. ellipses, by applying the same process to a sufficient number of centroids for the detection of the curve equation.

At the present state we focused on verifying the effectiveness of a search method, which relies on multiple criteria; in particular in this paper we focused on the exploitation of regular patterns to characterize assembly models and to improve the models' searching. In the next future, we will concentrate on the finalization of modules required for the extraction of the other information represented in the EAM.

Future work will also focus on the efficiency of the method with respect to large dataset of complex assemblies and on the refinement of the similarity measure to take into account all the characteristics included in the EAM.

ORCID

Katia Lupinetti  <http://orcid.org/0000-0002-0202-4909>

Lisa Chiang  <http://orcid.org/0000-0002-5992-3569>

Franca Giannini  <http://orcid.org/0000-0002-3608-6737>

Marina Monti  <http://orcid.org/0000-0002-1627-3551>

Jean-Philippe Pernot  <http://orcid.org/0000-0002-9061-2937>

References

- [1] Biasotti, S.; Marini, S.; Mortara, M.; Patanè, G.; Spagnuolo, M.; Falcidieno, B.: 3D Shape Matching through Topological Structures, *Discrete Geometry for Computer Imagery*, I. Nystrom *et al.* (Eds.): DGCI 2003, LNCS 2886, pp. 194–203, 2003. http://dx.doi.org/10.1007/978-3-540-39966-7_18
- [2] Bunke, H.; Foggia, P.; Guidobaldi, C.; Sansone, C.; Vento, M.: A comparison of algorithms for maximum common subgraph on randomly connected graphs, in *Structural, Syntactic, and Statistical Pattern Recognition*, T. Caelli *et al.* (Eds.): SSPR&SPR 2002, LNCS 2396, 2002, Springer, p. 123–132 http://dx.doi.org/10.1007/3-540-70659-3_12
- [3] Cardone, A.; Gupta, S. K.; Karnik, M.: A survey of shape similarity assessment algorithms for product design and manufacturing applications, *Journal of Computing and Information Science in Engineering*, 3(2), 2003, 109-118. <http://dx.doi.org/10.1115/1.1577356>
- [4] Chen, X.; Gao, S.; Guo, S.; Bai, J.: A flexible assembly retrieval approach for model reuse, *Computer-Aided Design*, 44(6), 2012, 554-574. <http://dx.doi.org/10.1016/j.cad.2012.02.001>
- [5] Chiang, L.; Giannini, F.; Monti, M.: Identification of regularities in CAD part and assembly models, IFIP WG5.1 12th Int. Conf. on Product Lifecycle Management, PLM'15, Doha, Oct. 19-21, 2015.
- [6] Deshmukh, A. S.; Banerjee, A.G.; Gupta, S. K.; Sriram, R. D.: Content-based assembly search: A step towards assembly reuse, *Computer-Aided Design*, 40(2), 2008, 244-261. <http://dx.doi.org/10.1016/j.cad.2007.10.012>
- [7] Grabcad workbench, <https://grabcad.com/library>.
- [8] Hu, K.-M.; Wang, B.; Yong, J.-H.; Paul, J.-C.: Relaxed lightweight assembly retrieval using vector space model, *Computer-Aided Design*, 45(3), 2013, 739-750. <http://dx.doi.org/10.1016/j.cad.2012.10.005>
- [9] Jayanti, S.; Kalyanaraman, Y.; Iyer, Y.; Ramani, K.: Developing an engineering shape benchmark for CAD models, *Computer-Aided Design*, 38(9), 2006, 939–953. <http://dx.doi.org/10.1016/j.cad.2006.06.007>
- [10] Lupinetti, K.; Giannini, F.; Monti, M.; Pernot, J.P.; CAD assembly descriptors for knowledge capitalization and model retrieval *Proceedings of TMCE 2016*, May 9-13, 2016, Aix-en-Provence, France, edited by I. Horváth, J.-P. Pernot, Z. Rusák.
- [11] Miura, T.; Kanai, S.; 3D Shape Retrieval considering Assembly structure, in *Proceedings of Asian Symposium for Precision Engineering and Nanotechnology*, 2009 (ASPEN 2009).
- [12] SolidWorks industrial designer <http://help.solidworks.com/>
- [13] Tangelder, J. W.; Veltkamp, R. C.: A survey of content based 3D shape retrieval methods, *Multi-media tools and applications*, 39(3), 2008, 441-471. <http://dx.doi.org/10.1007/s11042-007-0181-0>
- [14] Tao, S.; Huang, Z.: Assembly model retrieval based on optimal matching, *Software Engineering and Knowledge Engineering: Theory and Practice* ed. Springer, 327-336. http://dx.doi.org/10.1007/978-3-642-29455-6_46
- [15] Tracepart <http://www.tracepartonline.net/>
- [16] Visualisation Virtual Service <http://visionair.ge.imati.cnr.it/>
- [17] You, C.-F.; Tsai Y.-L.; 3D solid model retrieval for engineering reuse based on local feature correspondence, *The International Journal of Advanced Manufacturing Technology* 46.5-8 (2010): 649-661.
- [18] Content Central <http://www.3dcontentcentral.com/>