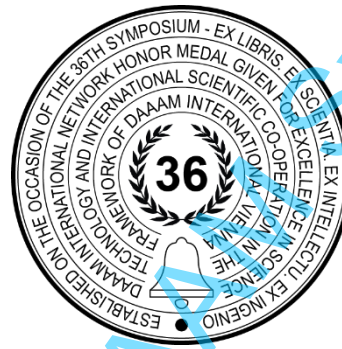


AN APPROACH FOR AUTOMATED CREATION OF THREE-DIMENSIONAL HISTORY-BASED PARAMETRIC MODELS

Velizar Zaharinov & Ivo Malakov



This Publication has to be referred as: Zaharinov, V[elizar] & Malakov, I[vo] (2025). An Approach for Automated Creation of Three-Dimensional History Based Parametric Models, Proceedings of the 36th DAAAM International Symposium, pp.xxxx-xxxx, B. Katalinic (Ed.), Published by DAAAM International, ISBN 978-3-902734-xx-x, ISSN 1726-9679, Vienna, Austria
DOI: 10.2507/36th.daaam.proceedings.xxx

Abstract

The paper proposes an approach for automated creation of three-dimensional history-based parametric models from non-parametric three-dimensional models. A review and analysis of existing approaches in the literature is made. The proposed approach is presented in the form of a functional structure. For each of the ten stages constituting the approach, the problems that need to be solved, the necessary input-output information and the results that need to be achieved are formulated. An example application for creating a parametric model of a part from a non-parametric model obtained by three-dimensional scanning is given. The particularities of the practical application of the approach are indicated.

Keywords: CAD; parametric model; history-based CAD; automated model construction.

1. Introduction

The paper proposes an approach for automated creation of three-dimensional history-based parametric models from non-parametric three-dimensional models. The proposed approach is developed to be used in the context of parametric history-based CAD environments. Nowadays, these environments are extremely common as tools for creating parts and assemblies of products from mechanical engineering [1], [2], automotive engineering [3], aircraft engineering, etc. The presence of many well-known CAD environments [4], [5], [6], [7], [8], supporting this type of modeling and used in the industry by leading manufacturers such as TOYOTA, INFINITI, OMRON, Mercedes-Benz, VOLVO, MAN, General Motors, AIRBUS, FUJITSU, SIEMENS and many others, is a confirmation of the benefits of its application in the product design process. In order to take advantage of all the benefits of parametric modeling, however, it is necessary to build the model by applying the appropriate parametric modeling tools. In the design process, a number of factors require the creation of non-parametric models. Situations that necessitate such models are, for example: geometry data of an object obtained by three-dimensional scanning; transferring models between different CAD environments, which requires going through model descriptions that are external (do not fully correspond to the description used naturally by the CAD environment); the model goes through a generative design process; the model is created through direct modeling, etc.

Manually creating parametric models from non-parametric models is a laborious and time-consuming process, in which geometric features can be lost if the process is not carried out methodically and accurately. Considerable flexibility

and experience are required, since each model has its own geometric features, and parameterization techniques require excellent knowledge of the CAD environment used. In addition, the work on parameterization takes up a significant amount of time for the designer, which could be used for other more creative activities related to the design and manufacturing process of the product. Taking into account the above, automating this process would lead to more accurate and faster parameterization of non-parametric models, which would relieve designers and save them time for more important and creative activities related to the added value of the product.

One of the essential issues in automating the process of creating parametric models from non-parametric ones is the automated recognition of geometric features. There are a number of developments devoted to this topic, which find application not only in CAD. The first such method, described in [9], was developed for application in group technology, for coding parts, and involves the classification of the faces that make up the parts into separate groups (primary and secondary). Based on this classification, the shape of the model is analyzed and geometric features such as holes, base surfaces, channels, pockets and other connections between the shapes that make up the part are extracted.

In [10] is a review of the publications made in the period from 1980 to 2016 in relation to methods for recognizing geometric features. The authors point out that the most publications, and therefore the widest scientific interest, in this period, are the methods focused on four main principles: rule/guideline-based approach, graph-based approach, volumetric decomposition approach, and artificial neural network approach.

The rule-based approach [11], [12], [13] uses the idea of an expert system. This is one of the earliest researched approaches. The features are summarized in the form of templates consisting of characteristic patterns of rules. The recognition process is carried out by applying these rules in an if-then analysis. If the predefined conditions are satisfied, then the corresponding structure in the part is recognized as a geometric feature.

To address some of the shortcomings of the rule-based approach, guideline-based methods have been developed. They are based on the idea that it is possible to search for an incomplete representation that can be used as a guideline for the existence of certain geometric features. The goal is to find a trace of a geometric feature that would be present in the model, even when combining several geometric features. This type of methods uses a two-step recognition procedure [10]. In the first step, guidelines are created using extraction rules based on various properties such as inferences from the geometry and topology of the model [14], [15], [16], [17], taxonomies of features [18], [19], [20], and combined probabilities for ranking potential guidelines for features [21], [22]. In the second step, guidelines are processed and can be directly compared by applying rules, with some developments also using a verification phase, after building feature volumes from guidelines and model boundary data.

The graph-based approach is among the most studied methods [10] because graphs are similar to the representation of solid models in three-dimensional CAD environments. In this type of model description, nodes and arcs usually represent faces and edges, which are assigned along with some properties such as convexity and concavity of edges, face type, perpendicularity, parallelism or tangentiality of edges and faces, etc. The features are then extracted in the form of a subgraph from the full graph. In [10] 20 methods based on the graph approach are reviewed.

The volume decomposition approach determines the volume of material removed from the workpiece during machining and decomposes the volume initially into intermediate volumes, after which the features are generated by combining the intermediate volumes based on certain rules. The methods implementing this approach can be broadly divided into two groups: decomposition by a described convex body and cell-based volume decomposition.

The artificial neural network approach mimics human perception and the learning process in humans [23]. The main advantage is that it does not use a rigid logical procedure (algorithm) for recognizing characteristic features, but rather applies arithmetic operations to process data, training the algorithm. This avoids the need to develop a strategy (logical sequence of actions, algorithms) for each particular case and provides flexibility, allowing for exceptions or incomplete recognition patterns.

The aim of this article is to propose a solution, in the form of an approach, that would reduce the labor intensity of using modern technologies for creating CAD models and increase their usefulness by making the use and editing of the created models easier. An opportunity to achieve this goal is seen by the authors in the automated creation of parametric models from non-parametric ones.

2. Automation approach

2.1. General function and input/output information

The general function of the approach is: "Automated creation of a three-dimensional history-based parametric CAD model". The general function is the relationship between the input that the approach expects and the output from it. The input and output are informational and the main flow is information processing (Fig. 1).

The approach expects two information inputs and produces one information output. One information input represents the description of the non-parametric CAD model. The second one gives the user the opportunity to adjust the process. The result of the approach is a parameterized history-based CAD model.

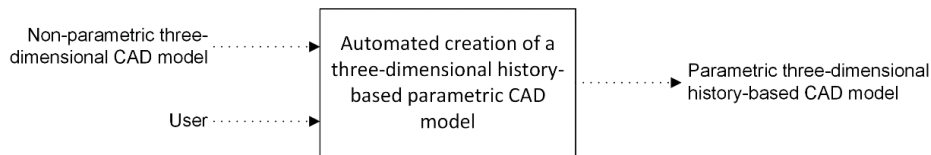


Fig. 1. General function of the approach for automated creation of three-dimensional history-based parametric models from non-parametric models

2.2. Structure of the approach

The individual stages of the proposed approach are shown in Fig. 2 in the form of a functional structure. The results obtained when performing the individual stages (functions) are also indicated. It is also shown where the inputs and outputs defined by the general function are connected (Fig. 1).

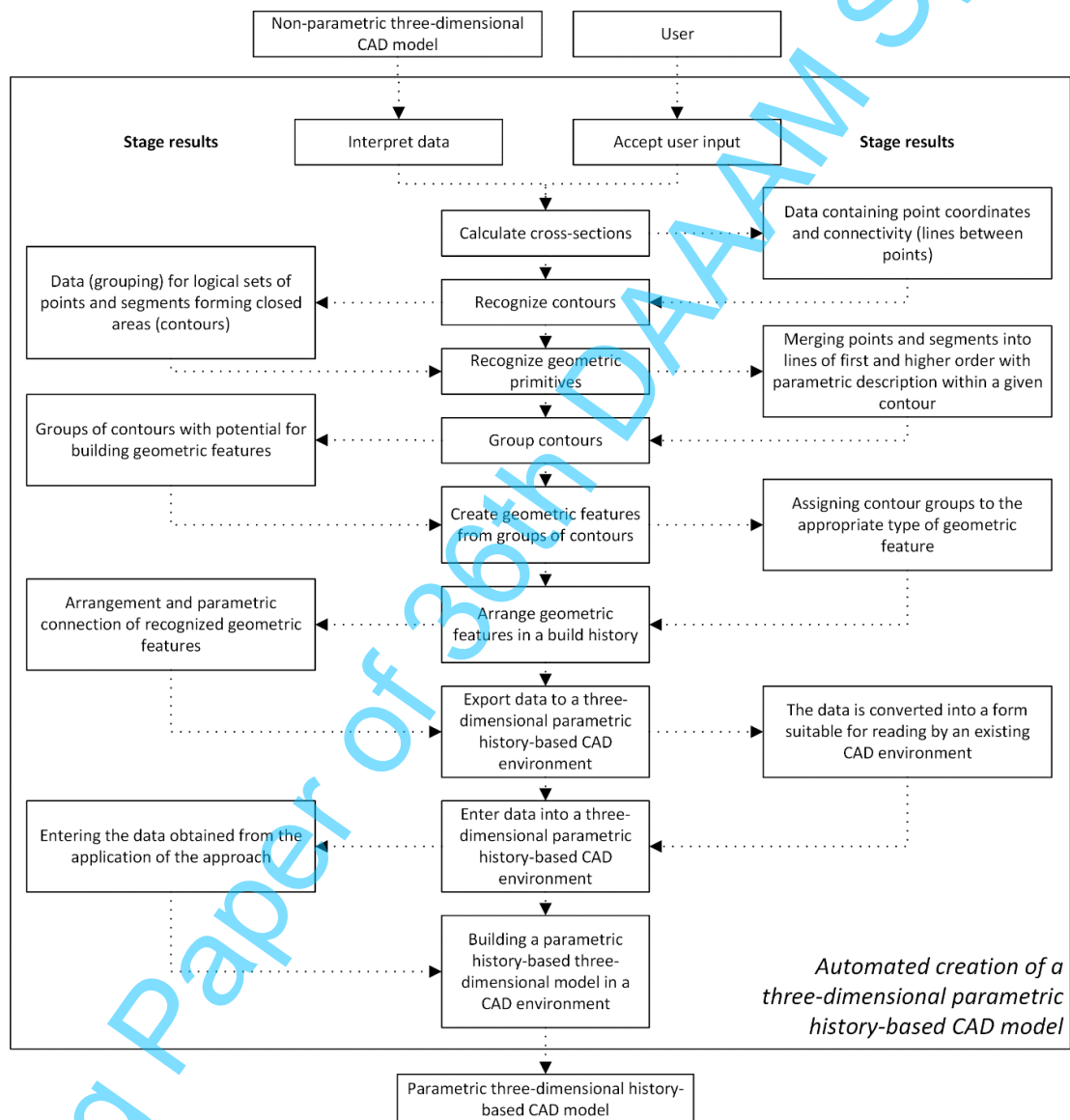


Fig. 2. Functional structure of the approach and results of the individual stages

3. Description of the approach

3.1. Stage 1. Input (includes interpreting data and accepting user input)

The transfer of data about the geometry of a three-dimensional CAD model is associated with the use of a certain format for recording the geometric information. There are a variety of CAD environments that offer a variety of ways to

record data about the model. The recording is done in a file with a corresponding extension indicating the type of data recording, i.e. the file format. In general, these file formats can be classified as “native” and “neutral”.

Native file formats are specific to a particular CAD environment and are usually protected by copyright. This makes it difficult to use native file formats for data transfer between competing CAD environments and requires the use of translators and validation. Another feature of native file formats is that they are not standardized, as they are developed by private companies. An advantage of native file formats over neutral ones is that they preserve all data (including parameterization and history) that can be described by the respective CAD system and integrated into the product model.

Neutral file formats offer the possibility of data transfer between different CAD environments. In addition, some of them are standardized. The disadvantage of these formats is that the parameterization and history present as information in the native format for the CAD environment are lost. However, these file formats preserve information about primitives such as cylinders, planes, points, edges, etc., related to the geometry of the model. This significantly facilitates the use of approaches and methods for parameterization, recognition of geometric features and creation of a history-based model. Table 2 shows some of the most common CAD systems for three-dimensional modeling and their native file formats (only for parametric three-dimensional models), and Table 3 shows some of the most commonly used neutral CAD file formats for data transfer for three-dimensional models.

All formats listed in Tables 1 and Table 2, except STL, include information about primitives, the so-called B-rep representation, which is standard in modern three-dimensional CAD systems. In STL files, only information about basic geometric primitives such as points and faces is stored. Due to the flexibility of the STL and OBJ file formats, they were chosen to describe the input data for the approach. This choice was made taking into account the following considerations: the ability to cover all four situations mentioned at the beginning of the introduction, in which parameterization is required; the formats are neutral and do not require the purchase of rights for use; simple data structure; the STL format is widely used and supported by a large part of modern CAD/CAM/CAE systems.

Although the OBJ format is not a traditional CAD format, it was chosen to support it in addition to the widely used STL because it allows the transfer of three-dimensional models that have been created through three-dimensional scanning, generative design methods, or direct modeling. In these situations, saving the data as an STL file is not always possible (most often due to lack of support in the model creation environment).

The main disadvantage of the chosen neutral file formats is that they offer an approximate description. This is because triangles are used to describe the geometry of the 3D model (Fig. 3), and not B-rep, as in STEP files for example. However, this disadvantage can also be seen as an advantage, since, using a single geometric primitive (triangle), the description can be generalized regardless of the type of modeling used to produce the 3D geometry (parametric, non-parametric).

The result of decoding the input data file is a set V of points v_i , each of which is defined by a triple of coordinate values $\{x_i, y_i, z_i\}$, $x_i, y_i, z_i \in \mathbb{R}$, and a set F of faces f_j , each of which is defined by the indices $\{\varphi_{j,1}, \varphi_{j,2}, \varphi_{j,3}\}$, $\varphi_{j,1}, \varphi_{j,2}, \varphi_{j,3} \in \mathbb{N}$, of three points from the set V . Further processing of the model geometry, in the course of applying the stages of the approach, will expect precisely these data as input.

CAD system	Native file format extensions
CATIA	*.CATPart; *.CATProduct
Creo	*.prt; *.asm
SolidEdge	*.par; *.asm
Inventor	*.ipt; *.iam
SolidWorks	*.sldprt; *.sldasm

Table 1. Native file formats

Name	Neutral file format extensions	Information
STEP	*.STEP; *.STP	Year of creation 1994; Standardized (ISO10303-242, 2020)
QIF	*.QIF	Year of creation 2013; Standardized (ISO23952:2020, 2020)
JT	*.JT	Year of creation 2007; Standardized (ISO14306:2017, 2017)
STL	*.STL	Year of creation 1987
IGES	*.IGS; *.IGES	Year of creation 1980; The first neutral CAD format
ACIS	*.SAT	Year of creation 1989; Created by Dassault Systemes
PARASOLID	*.X T; *.X B	Year of creation 1989; Created by Siemens

Table 2. Neutral file formats

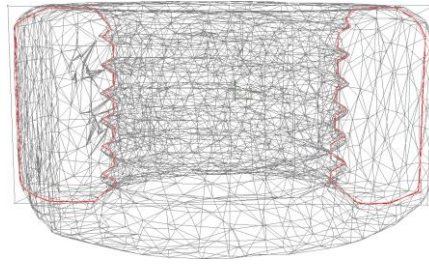


Fig. 3. A model of a part built from triangles with a depicted intersecting plane (rectangular frame) and curves obtained from the intersection of the plane with the model (depicted with overlaid red lines)

3.2. Stage 2. Calculate cross-sections

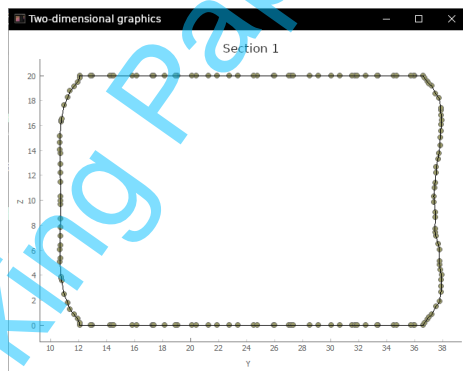
The goal of this stage is to calculate the curves obtained from the intersection of the three-dimensional model with a cutting plane. The input information required for the stage is a description of the three-dimensional model in the form of the sets V and F obtained from the decoding of an STL or OBJ file (Stage 1). The model at this stage is built from individual faces, in the form of triangles, constituting its outer surface. Each triangle consists of three points located at its vertices. They determine the intersection points of the sides of the triangles.

The result of this stage is a set of points S , defined by their coordinates, a set of lines E_S , defined by pairs of indices of elements of the set S (two points each), and a set of faces (triangles) F_S , defined by triples of indices of elements of the set S (three points each). This result completely defines the resulting section. Fig. 4 shows sample results. Fig. 4a shows a section without triangles, and Fig. 4b shows a section containing triangles.

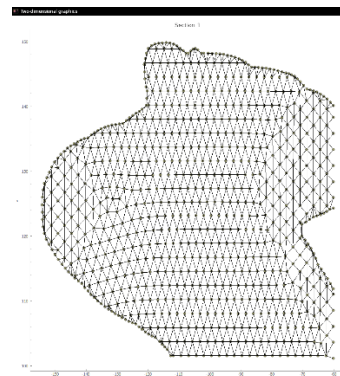
3.3. Stage 3. Recognize contours

The input for this stage is the result of the previous one, i.e. the sets S , E_S and F_S . The results of the intersection of the plane with the three-dimensional geometry of the model can be a variety of shapes – single points, points lying on a straight line, closed contours consisting of multiple points, contours containing faces (triangles), etc. In order to be able to create a parametric sketch of the section, it is necessary for these individual objects in the plane to be systematized (recognized) and separated from each other. The first step for this is the recognition of the individual contours in the set E_S , which carries the information about the connectivity of the points in the section. For the purposes of the approach, a “contour” is understood as a closed curve in the intersecting plane, composed of points, elements of S , arranged sequentially. The sequential arrangement of the points in the contour depends on the connectivity information contained in the set E_S . In practice, to create a geometric feature from a cross-section, it is necessary to produce a sketch containing only the contour, and the faces are not necessary. Moreover, they interfere with the recognition of the contour.

The obtaining (calculation) of the points from an arbitrary section does not follow a specific order. The search for the faces intersecting with the plane from the three-dimensional model is performed according to their order recorded in the STL or OBJ file. This order does not necessarily follow a specific regularity (if such a thing is assumed, it falls into a special case). Therefore, the points from the section are obtained in a random order and are arranged in the same way in the set S (in the order of their calculation). The set E_S , containing the lines from the section, and the set F_S , containing the faces, are also not arranged in any regular way. On the other hand, in order for the recognition of the contour to be possible, the points from it must be arranged one after the other (sequentially). It does not matter which is the starting point, or in which direction the contour will be traversed when recording the points, but the sequential arrangement is a condition for recognizing the contour as such and distinguishing it from the other objects in the intersecting plane.



(a)



(b)

Fig. 4. Two types of common sections: (a) containing contour(s) filled with faces (triangles) and (b) empty contour(s)

The result of this step is a set C containing all the contours of the section S , both those that contain faces in their interior and those without faces. If the section contains contours filled with faces (Fig. 4b) these must be removed. In addition, the set C also contains objects that are not contours - lines and single points, if such exist in the section. These objects are easily distinguished from contours, since the last element of the set does not point to the first (contours are closed, but lines and points are not).

3.4. Stage 4. Recognize geometric primitives

The goal of this stage is to parameterize the recognized contours. The idea is for the parameterized contours to be the basis for parametric sketches, from which geometric features can be created. The parameterization of three types of primitives is proposed: segments, arcs, and circles. The reason for choosing these three primitives is their widespread use in creating parametric sketches. The goal is also to minimize the number of parametric primitives used for simplicity and efficiency. Of course, based on the recognized contours, it is possible to parameterize other types of lines in the plane, which is only a matter of developing additional recognition algorithms. To implement the recognition, it is proposed to perform "segmentation" of the contours in the section. By "contour segmentation" is meant the division of the contour into separate sections, which are subsets of the contour, i.e. sequences of points belonging to the contour. All points of the segment are sequentially connected, i.e. the first point is connected to the second, the second to the third, etc. (Fig. 5).

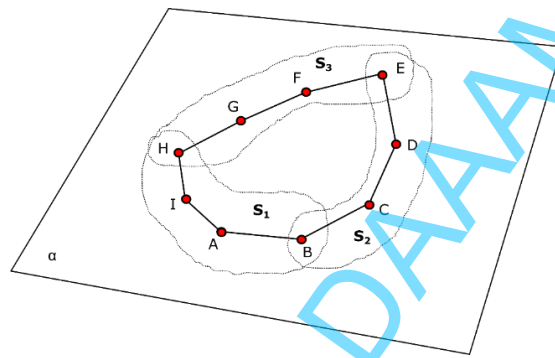


Fig. 5. Example of segmenting a contour. The segments are labelled S_1 , S_2 , and S_3 and are enclosed by dashed lines. The segmentation points are B, E, and H

It is possible to use multiple segmentation criteria, for example, an interval of permissible values for the angle between two adjacent contour segments. Multi-criteria decision-making for segmentation is also possible. The result of this stage is three sets for each recognized contour in the cross-section, which contain the recognized lines, arcs, and circles for the specific contour.

3.5. Step 5. Group contours

The grouping of contours is a stage that is important for the correct determination of the types of geometric features that can be constructed from the obtained cross-sections. To solve the problems at this stage, the following requirements can be formulated: it is necessary to obtain a sufficient number (sufficient "density") of cross-sections; it is necessary to formulate criteria and/or constraints for grouping; it is necessary to formulate a decision criterion.

The problem of the sufficiency or density of the sections produced in stage 2 of the approach is significantly related to the scale of the model and the accuracy with which it is desirable to parameterize it. A larger number of sections increases, in addition to the accuracy of parameterization, the computational time. Intuitively, it can be assumed that the optimization of the computational time is largely related to adaptive decision-making about the number of sections in certain regions of the three-dimensional model - in some parts, which need to be parameterized more precisely, the density of sections is high, and in others - low. This conclusion can also be reached after visual analysis by a human. The full automation, i.e., without human guidance, of the adaptive determination of the number of sections, has to be approached iteratively. It is necessary to initially calculate a small number of sections in order to be able to analyze the existing geometry of the model. The number can be determined from the overall dimensions of the three-dimensional model in the direction of the intersecting planes (normal to them). Based on the change in geometry between two adjacent sections, a decision can be made whether more or fewer sections are needed between them. The result of this stage is groups of contours, grouped according to their similarity and/or suitability for use as feature sketches.

3.6. Stage 6. Create geometric features from groups of contours

Different parametric history-based CAD environments offer the creation of different geometric features. Depending on the target CAD system, there may be some differences in the types of geometric features and their parameterization.

However, there are some common geometric features, such as those created by extruding a sketch and by rotating a sketch about an axis of symmetry. The mentioned common ways of creating geometric features also come in two varieties, those that add material to the model and those that remove material from it (producing cavities or holes). Even with just these four parametric geometry creation tools, it is possible to build complex 3D models. Therefore, assigning geometric features to groups of contours is a complex task. In many cases, combining several geometric features produces geometry that is difficult to determine its elementary components. The problem of classifying groups of contours can be approached in the following way: selection of a set of characteristic geometric features to which the groups of contours will be classified; analysis of the minimum number of parameters necessary for parameterization of the selected characteristic geometric features; formulation of classification criteria for each characteristic geometric feature; development of tools to automate the process. Many common geometric features require a two-dimensional drawing – a sketch – on which they are built. The sketch can be created from the contours in the group, with one of them being chosen as representative of the entire group. It is also possible to use averaging.

3.7. Stage 7. Arrange characteristic geometric features in a build history

The stage of arranging the recognized characteristic geometric features in a build history can be considered as a combinatorial problem, in which one must find such a combination of characteristic geometric features that it is possible to parameterize the overall model and that it has a certain meaning. A certain combination of the recognized characteristic geometric features represents a possible construction sequence. Some of the possible combinations will lead to invalid geometry. Depending on the number of recognized geometric features, for some models it will be possible to analyze all combinations, while for others the analysis time will be too long (combinatorial explosion). In these cases, a directed search for a solution will be necessary, as a possible approach is the use of characteristic ways of building models used in practice: for example, a certain direction of construction - from one initial plane to another final one, establishing a basic geometric feature, symmetrical construction, etc.

3.8. Stage 8. Export data to a three-dimensional parametric history-based CAD environment

At this stage, the sketches and geometric features are saved in a file format that allows the transfer of the data obtained from the analysis. The goals of this format are: to be open, so that it can be used by CAD environment manufacturers; to transfer the information in a compact and lossless way.

3.9. Stage 9. Enter data into a three-dimensional parametric history-based CAD environment

This is the stage in which an existing CAD environment reads and interprets the data from the file format for transferring the analysis made according to the proposed approach. The necessary preliminary preparation is carried out before executing the automated construction process. It is necessary to determine a sequence for interpreting the transferred data.

3.10. Stage 10. Building a parametric three-dimensional history-based model in a CAD environment

At this stage, using the capabilities of the CAD environment, a sequence of commands is executed to create a parametric model with build history, following the analysis results obtained within the approach. It is necessary to perform interpretation of the results of the approach in an existing CAD environment and automated execution of a sequence of commands in an existing CAD environment.

4. Application example

The input data (Stage 1) is a model obtained by 3D scanning (Artec 3D LTD, 2024) (Fig. 6a). This model has a relatively simple geometry, but due to its obtaining by 3D scanning, even the relatively simple geometry is significantly “weighted” due to the fact that an extremely large amount of geometric information is generated during scanning. In addition, modern 3D scanners achieve resolutions of 0.01 mm, which means that the model may also contain geometry that is not relevant to the technical documentation of the object (for example, traces of wear and imperfections, as can be seen in Fig. 6a). Usually, this type of geometry is not desirable to be restored during parameterization.

The model consists of 446 350 points defining 1 339 450 segments that make up 892 947 triangles. Sections (Stage 2) are calculated in the horizontal (Fig. 6b) and vertical (Fig. 9) axes of the model. The points that define the contours of the section in the horizontal axis (Fig. 6b) are very densely located, at small distances from each other. The geometry is not ideal, i.e. as in an engineering drawing, but reflects the real object with its imperfections. The chamfers on the edges of the model are not straight lines, not even parts of arcs, but combinations of straight sections and curves. These anomalies may be due to wear and/or the post-processing that is necessary after the three-dimensional scan to remove noise and unwanted geometry. In Stage 3, the contours in the sections are recognized – in this case, there are two in the horizontal and vertical axes (Fig. 7b and Fig. 9b). Then, segmentation and recognition of primitives (Stage 4) is applied – segments and arcs. Fig. 7b shows the left contour from Fig. 6b and the parameterized result after segmentation (Fig.

7c). Fig. 7c shows that the parameterized contour is far from following the shape of the original contour well. To improve the obtained results, a reduction of the triangles making up the three-dimensional model is applied. The reduction is performed in the non-parametric modeling application Blender 3.5 (Blender Foundation, 2024). The tool implemented in the cited application aims for minimal changes to the shape when performing the reduction. Fig. 8 shows the result of the same procedure for obtaining the parameterized contour from Fig. 7, but applied to the reduced model. From Fig. 8 it can be seen that the similarity between the parametric contour and the non-parametric one is significantly improved compared to Fig. 7. Stage 5 is performed after visual analysis of the obtained cross-sections in both directions. Three groups of contours are identified that are suitable for parametric construction of the part. This is the profile shown in Fig. 8c and the circle and pentagon from the cross-section shown in Fig. 9a (the parametrized pentagon is shown in Fig. 9c). Three characteristic features can be built from these three groups of contours (Stage 6). These are: extrusion of the pentagon (feature A), removal of material by extruding it in the shape of the circle from Fig. 9a (feature B) and removal of material by rotation around the central axis of the contour from Fig. 8c (feature C). The logical sequence (Stage 7) for building the model is to create first feature A, then B and finally C (Fig. 10a and b). Data output to a CAD environment (Stage 8) is performed via a DXF file. The DXF file contains the parametric curves - segments, arcs and circles, segmented and recorded with their basic parameters. The DXF file is introduced into the CAD environment (Stage 9) as separate sketches, the basis for building features. The parametric model is built using the tools of the CAD environment (Fig. 10c).



Fig. 6. Model of a three-dimensional scanned part

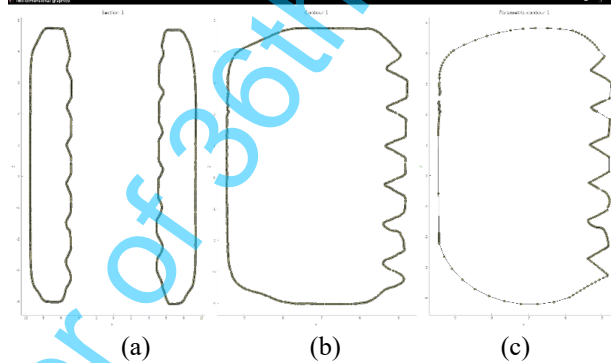


Fig. 7. Parameterization of a contour. (a) section along the horizontal axis, (b) contour; (c) parameterized contour

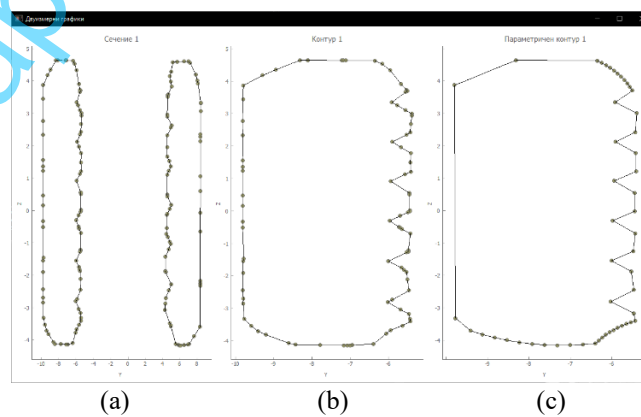


Fig. 8. Parameterization of a contour after reduction

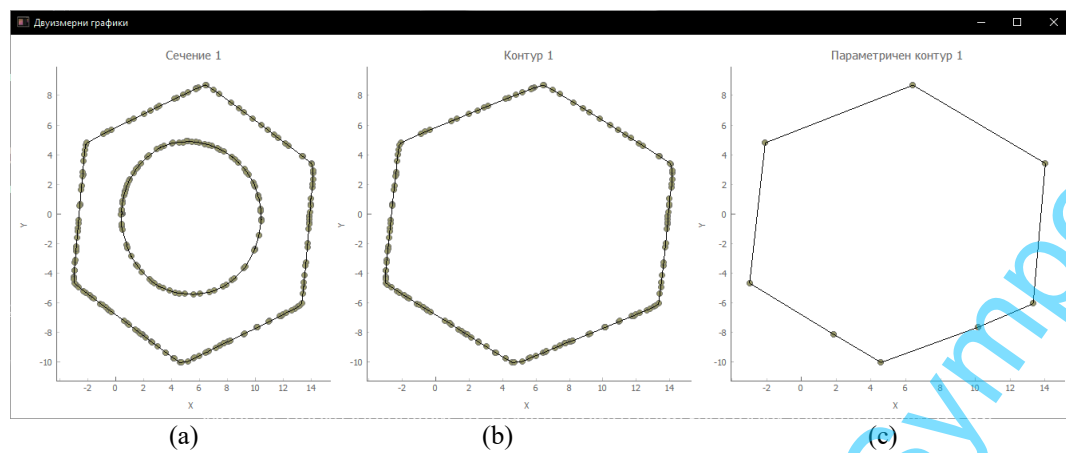


Fig. 9. Section and contour along the vertical axis of the model

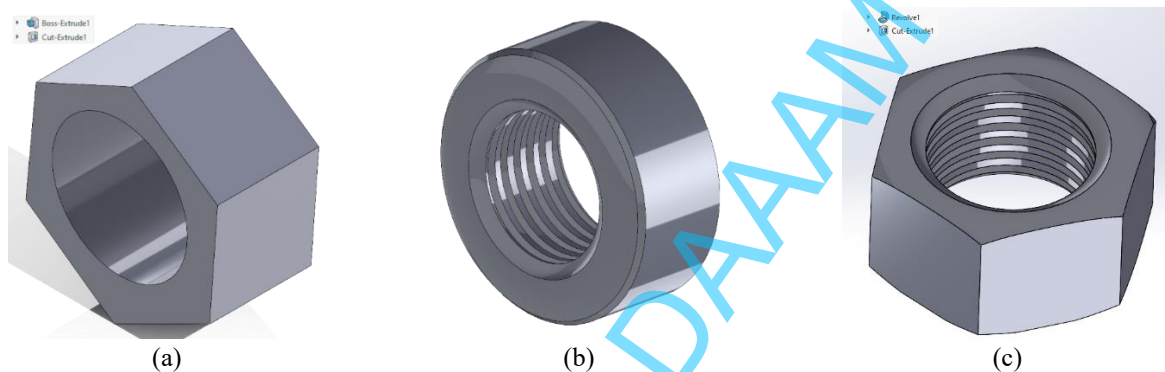


Fig. 10. Building the parametric model: (a) characteristics A and B, (c) characteristic B, (d) resulting model

5. Conclusion

This article proposes an approach to automate the process of obtaining a parametric history-based model with a from a non-parametric one. The focus of the approach is on three-dimensional models of technical products, which are used in a CAD environment to document the design and production process of a given product. The approach is described in the form of a functional structure, with each function being a separate stage. The expected results of each stage are indicated, as well as the problems that each stage must solve. An example application of the approach is shown, indicating the problems when working with real models.

The report describes the principles of the approach implementation, but does not go into depth of the mathematical and algorithmic foundations for implementing the proposed ideas, since each stage of the approach includes multiple mathematical models and algorithmic support, and the volume is limited. The authors believe that it is important to first present the overall approach as a working framework, and then to go into depth (the principle from general to particular).

A large part of the problems identified in the first four stages have been solved by the authors' developments, and another part is yet to be solved (from the fifth stage to the last stage of the approach). Work is underway to publish the algorithmic and software support for the first four stages and develop mathematical models, methods and algorithms for solving the formulated problems in the remaining stages. In addition to this work, testing the developed approach and tools with a more diverse set of models and ways to obtain them is envisaged as a future development.

6. Acknowledgments

Both authors are affiliated to Technical University Sofia and National Centre of Excellence „Mechatronics and Clean Technologies“, Sofia, Bulgaria.

This work was supported by European Regional Development Fund under "Research Innovation and Digitization for Smart Transformation" program 2021-2027 under the Project BG16RFPR002-1.014-0006 "National Centre of Excellence Mechatronics and Clean Technologies".

This work was accomplished by the Center of Competence for Mechatronics and Clean Technologies "Mechatronics, Innovation, Robotics, Automation and Clean Technologies" – MIRACle, with the financial support of contract No. BG16RFPR002-1.014-0019-C01, funded by the European Regional Development Fund (ERDF) through the Programme "Research, Innovation and Digitalisation for Smart Transformation" (PRIDST) 2021–2027.

7. References

- [1] Dichev, D., Zhelezarov, I., Georgiev, B., Karadzhov, T., Dicheva, R., & Hasanov, H. (2025). A Method for Measuring Angular Orientation with Adaptive Compensation of Dynamic Errors. *Sensors*, Vol. 25, No. 16, 4922, DOI: <https://doi.org/10.3390/s25164922>
- [2] Todorov, T.T.; Todorov, G.; Sofronov, Y. (2025) Process Parameter Optimization through Virtual Simulations for Enhanced Durability of Artificial Ski Slope Tiles, DOI: <https://doi.org/10.1088/1742-6596/3061/1/012008>.
- [3] Kamberov K. H., Todorov G. D., Ivanov Ts. T. (2024) Automotive Product Validation through Virtual Prototyping at Early Design Stage, AIP Conference Proceedings, Vol. 3064, No. 1, art. no. 030007, DOI: <https://doi.org/10.1063/5.0199194>.
- [4] <https://www.ptc.com/en/products/creo>, (2025). PTC Inc. Creo. Accessed on: 2025-09-23.
- [5] <https://www.3ds.com/products/catia> (2025) Dassault Systemes CATIA. Accessed on: 2025-09-23.
- [6] <https://www.solidworks.com/> (2025) Dassault Systemes SOLIDWORKS. Accessed on: 2025-09-23.
- [7] <https://www.autodesk.com/products/inventor/overview> (2025) Autodesk Inventor. Accessed on: 2025-09-23.
- [8] <https://solidedge.siemens.com/en/> (2025) SIEMENS Solid Edge. Accessed on: 2025-09-23.
- [9] Kyprianou, L. (1980). Shape classification in Computer-Aided Design, Ph.D. Dissertation, Cambridge University.
- [10] Shi, Y. Z. (2020). A critical review of feature recognition techniques, *Computer-Aided Design and Applications*, Vol. 17, No. 5, pp. 861-899, DOI: <https://doi.org/10.14733/cadaps.2020.861-899>.
- [11] Donaldson, I. C. (1993). Rule-based feature recognition for 2.5D machined components, *International Journal of Computer Integrated Manufacturing*, Vol. 6, No. 1-2, pp. 51-64, DOI: <https://doi.org/10.1080/09511929308944555>.
- [12] Vosniakos, G. D. (1993). A shape feature recognition framework and its application to holes in prismatic parts. *The International Journal of Advanced Manufacturing Technology*, Vol. 8, No. 6, pp. 345-351, DOI: <https://doi.org/10.1007/bf01751095>.
- [13] Chan, A. C. (1994). Process planning by recognizing and learning machining features. *International Journal of Computer Integrated Manufacturing*, Vol. 7, No. 2, pp. 77-99, DOI: <https://doi.org/10.1080/09511929408944597>
- [14] Bhandarkar, M. N. (2000). STEP-based feature extraction from STEP geometry for agile manufacturing. *Computers in Industry*, Vol. 41, No. 1, pp. 3-24.
- [15] Gao, J. Z. (2004). Extraction of machining features for CAD/CAM integration. *The International Journal of Advanced Manufacturing Technology*, Vol. 24, No. 7-8, pp. 573-581, DOI: <https://doi.org/10.1007/s00170-003-1882-9>.
- [16] Dimov, S. B. (2007). A hybrid method for feature recognition in computer aided design models. *Journal of Engineering Manufacture*, Vol. 221, No. 1, pp. 79-96, DOI: <https://doi.org/10.1243/09544054JEM437>.
- [17] Verma, A. R. (2008). A hint-based machining feature recognition system for 2.5 D parts. *International Journal of Production Research*, Vol. 46, No. 6, pp. 1515-1537, DOI: <https://doi.org/10.1080/00207540600919373>.
- [18] Xu, X. H. (1998). Recognition of rough machining features in 212D components. *Computer-Aided Design*, Vol. 30, No. 7, pp. 503-516, DOI: [https://doi.org/10.1016/S0010-4485\(97\)00090-0](https://doi.org/10.1016/S0010-4485(97)00090-0).
- [19] Fu, M. O. (2003). An approach to identify design and manufacturing features from a data exchanged part model. *Computer-Aided Design*, Vol. 35, No. 11, pp. 979-993, DOI: [https://doi.org/10.1016/s0010-4485\(02\)00160-4](https://doi.org/10.1016/s0010-4485(02)00160-4).
- [20] Nasr, E. K. (2006). A new methodology for extracting manufacturing features from CAD system. *Computers & Industrial Engineering*, Vol. 51, No. 3, pp. 389-415, DOI: <https://doi.org/10.1016/j.cie.2006.08.004>.
- [21] Vandenbrande, J. R. (1993). Spatial reasoning for the automatic recognition of machinable features in solid models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 15, No. 12, pp. 1269-1285, DOI: <https://doi.org/10.1109/34.250845>.
- [22] Han, J. R. (1997). Integration of feature based design and feature recognition. *Computer-Aided Design*, Vol. 29, No. 5, pp. 393-403, DOI: [https://doi.org/10.1016/S0010-4485\(96\)00079-6](https://doi.org/10.1016/S0010-4485(96)00079-6).
- [23] Topalova, I. & Staeva, Y. (2022). Classification of Two-dimensional Mechanical Parts Using a Convolutional Neural Network, *Proceedings of the 33rd DAAAM International Symposium*, pp.0042-0049, B. Katalinic (Ed.), Published by DAAAM International, ISBN 978-3-902734-36-5, ISSN 1726-9679, Vienna, Austria DOI: 10.2507/33rd.daaam.proceedings.007