



A TOOL FOR IDEALISATION OF CAD MODELS

M. Burić and D. Marjanović

Abstract

The idealisation of CAD models for FEA purpose is an iterative and time consuming process. In order to automate and speed it up, the present paper introduces a geometry simplification tool named CAD Idealiser, which has been developed using CATIA V5 automation. The macro is able to detect geometric details such as fillets, chamfers, holes & notches, and visualize them by means of different colors. In addition, it supports the user in suppression of detected geometric details. In order to verify the efficiency of the algorithm, several different CAD models have been subjected to idealisation.

Keywords: computer aided design (CAD), design automation, design tools, CATIA V5, macro

1. Introduction

Computer–Aided Design (CAD) and Computer–Aided Engineering (CAE) are two mutually correlated engineering fields of the design process. However, they are not well integrated because they inherently use different types of geometric models (Lee, 2005). A widely adopted method of the design process is to create an initially CAD model, and an iterative design process requiring periodic analyses and design changes is used to improve or refine the design (Lee, 2004), see Figure 1. CAD models are most often created based on the feature–based solid modelling techniques. Commonly those models are very complex containing an abounded number of design features. Geometric details such as very small fillets, chamfers, holes, notches, etc., might be identified as unimportant or irrelevant for analysis.

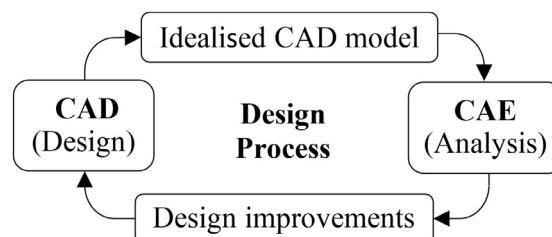


Figure 1. Conventional CAD–CAE loop

From the finite element analysis (FEA) viewpoint, geometric details are considered as hotspots for mesh generation resulting in poor quality (distortion) and unreasonable high number of mesh elements. Based on careful expert judgment, geometric details can be removed without significantly affecting the analysis results, while at the same time improving computational performance. This kind of transformation process is either referred to the term idealisation (Dabke et al., 1994; Lee, 2004; Hamdi et al., 2007) or simplification (Quadros and Owen, 2009; Thakur et al., 2009). However, the fact is that

even a simplified model is an idealised representation of the actual CAD model. Thakur et al. (2009) consider under the term idealisation the reduction of the CAD model dimensions.

The industrial practice has established two conventional approaches to conduct the idealisation process: by the CAE engineer (analyst) or by the CAD engineer (designer). Both approaches show certain advantages as well as disadvantages. The analyst's explicit knowledge and experience of the idealisation process are typically limited by the insufficient feature modelling capabilities of the CAE tools (Benaouali and Kachel, 2017). The designer's knowledge and experience are usually restricted when it comes to the idealisation process. Although the mentioned lacks in both cases can be overcome by training, the fact is that a significant amount of time is being spent on manual detection and suppression of geometric details. The idealisation process becomes, even more, time-consuming considering that most often multiple design iterations are required before achieving the optimal design solution.

2. Related work

The objective of this paper is to apply design automation for the use of idealisation of CAD models for FEA analysis. Over the past decades, different methods and techniques have been developed and proposed for conversion of the CAD geometry into analysis specific geometry. As the literature is quite exhausting in this field, within this paper only a survey of the most relevant techniques will be given. The study by Thakur et al. (2009) gives a more detail survey of simplification techniques and classifies them based on a type of operators used: surface, volumetric, dimension reduction and explicit-feature based.

Many techniques for recognition and suppression (defeaturing) of the geometric details are performed directly on boundary representation (B-Rep) models. Algorithms for the identification and suppression of different features (holes, blends with a circular cross-section, boss features, etc.) in B-Rep models were discussed by Joshi and Dutta (2003). An automatic method to suppress and repair constant radius round/fillet features, as well within B-rep models, has been presented by Zhu and Menq (2002). Sun et al. (2010) conducted simplification of B-Rep models using region suppression, while new surfaces to cover the suppressed regions were constructed using boundary loop decomposition, rather than conventional face extension and intersection methods. An automated face clustering method used as a pre-process of surface mesh generation were introduced in by Inoue et al. (2001) and Sheffer (2001). An extension of the face clustering method relying on the principles of virtual topology was presented by Sheffer et al. (1997). The method does not remove the details geometrically, instead only modifies the topology to simplify the meshing. Hamdi et al. (2012) proposed a hybrid method based on a combination of the elimination details and merging faces. Danglade et al. (2013) proposed an approach that uses machine learning techniques for identification of detailed features to be suppressed.

Lee (2005) proposed the multi-resolution models of a solid model simplified at various levels of details (LOD). The paper introduces an algorithm based on the effective volumes of features, instead of the topological entities. Further, Lee (2005) proposed the concept of multi-abstraction modelling in which the dimensionality of the features is varied at different levels of abstraction (LOA). Moreover, combinations of multiresolution and multi-abstraction models for the creation of various LODs and LOAs have been presented by Lee (2005) and Lee (2009).

In contrast to B-Rep techniques, other defeaturing techniques are focused on the discrete domain. Gao et al. (2010) proposed recognition and suppression of the detailed features on CAD mesh models, before the FEA mesh generation. Automatic identification of irrelevant features performed on a facet-based discrete B-Rep model has been proposed by Quadros and Owen (2009). Other discrete method proposed to conducted defeaturing as part of the FEA meshing process by ignoring or combining features (Mobely et al., 1998), or as a post-meshing procedure performing iterative edge-collapse operations on poor quality elements (Dey et al., 1997)

The dimension reduction techniques are based on the conversion of the CAD model dimensions (e.g. when a thin solid geometry is replaced by mid plane surfaces). Two common techniques are medial axis transform (MAT) (Foskey et al., 2003; Robinson et al., 2011) and mid-surface abstraction (Rezayat, 1996). The MAT technique is based on building a skeleton of the B-Rep geometry in order to obtain the median axis. A quite complete survey of the MAT method is given by Attali et al. (2009). The mid surface abstraction technique relies on the extraction of average surfaces of the B-Rep geometry.

In order to achieve additional time and cost reduction during the design process, CAD/CAE integration techniques have become more important in recent years. An automated CAD/CAE integration procedure implemented for the parametric design and structural analysis of aircraft wing structures was presented by Benaouali and Kachel (2017). Gujarathi and Ma (2011) proposed a CAD/CAE integration method using common data model (CDM) containing all required parametric information for both CAD modelling and CAE analysis. CDM enables to solve the problem of association of feature based semantic knowledge and iterations of CAD and CAE interaction cycles (Gujarathi and Ma, 2011). Wang et al. (2014) and Park and Dang (2014) were focused on developing an automated frame work for CAD/CAE integration.

When it comes to design automation, Russ (2012) developed a CAD model suppression tool for holes, rounds and chamfer features in Pro/Engineer using Toolkit automation possibilities. Some general applications of CATIA V5 automation were presented by Del-Rio Cidoncha et al. (2007).

Although the most state of the art CAE software packages shows limited feature modelling capabilities, some of them, as for example ANSYS SpaceClaim, are more convenient for model preparation. SpaceClaim supports the user in defeaturing of geometric details such as rounds, small holes and other irrelevant geometry. In addition, it provides merge, replace, and simplification operation of faces in order to reduce mesh complexity. The NX's Simcenter 3D contains geometry defeaturing and idealization tools to clean features like small holes, steps and slivers that can impact meshing speed and quality.

3. CAD Idealiser

State of the art commercial CAD software systems such as CATIA, Solidworks, NX, Creo, etc. have the ability to expand their capabilities by automating design tasks with the purpose to save time and costs, improve efficiency and standardise processes. CATIA Application Architecture (CAA) is the Application Programming Interface (API) which enables to develop and integrate user-specific applications for CATIA. CATIA V5 automation relies on several programming languages: CATScript, Visual Basic (VB6, VBScript, VBA, etc.) and C++. CATScript is portable version and non-GUI oriented programming language of Dassault System, similar to VBScript (a subset of VB). An advantage of CATScript is its easy use and recording ability, but it has limited flexibility and it's difficult to debug. Visual Basic for Application (VBA) is another subset of VB and is hosted in CATIA after V5R8. It provides a complete programming environment with an editor, debugger, and help object viewer. The scripting languages enable to develop macros in a reasonable time frame. A disadvantage of macros is that they don't run in their memory space, which sometimes can lead to freezing of CATIA while the macro is running. In addition, some automation objects are not accessible. C++ is the primary language of CAA and is more powerful since it's able to overcome all limitations of the scripting languages. The main disadvantage is its learning complexity, as well as time-consuming development of applications.

In order to speed-up and automate the idealisation process of CAD models, a CATIA V5 R18 macro named CAD Idealiser has been developed using VBA. CAD Idealiser contains a simple and intuitive Graphical User Interface (GUI) that guides the user through the idealisation process, which has been divided into three basic steps (see Figure 2): Search, Remove and Export. In the Search step the macro can to detect geometric details such as fillets, chamfers, holes and notches based on user defined maximum criteria, and visualize the search results using of different colours (fillets in red, chamfers in yellow, holes in green and notches in blue colour). In case the search process needs to be repeated, the CAD model colour can be reset to default (grey). However, the previous search results can as well be kept if necessary. CAD Idealiser is capable of analysing native models created in CATIA Part Design module, or foreign models imported from other CAD software systems in the form of STEP or IGS file formats. In the second step, Remove, the user can review the recorded search results in a list box and move them to another list box where they can be relatively quickly suppressed from the initial CAD model. For simpler identification, CAD Idealiser automatically highlights the selected feature from the list box in the 3D space and provides information of the maximum fillet radius, chamfer size, hole diameter or notch thickness. Finally, the Export step allows rapid transfer of the idealised CAD model to the desired location, as well as to choose one of the common CAE file formats (STEP, IGS or CATPart).

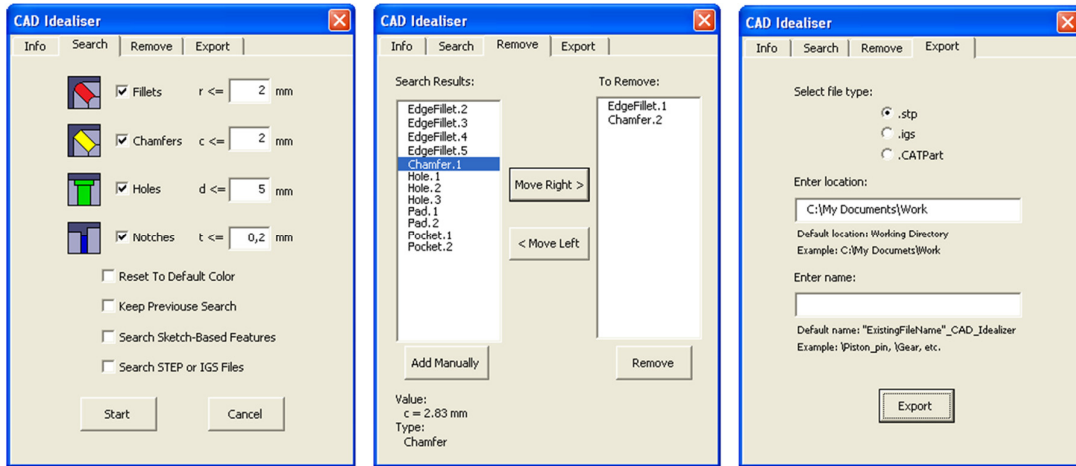


Figure 2. CAD Idealiser GUI – Search, Remove and Export steps

4. The algorithm

CATIA V5 automation structure can be divided into three types of elements (see Figure 3): collections, abstract objects and objects. A collection is a group or list of similar objects. An abstract object (e.g. prism, revolution, sweep, fillet, etc.) is not intended to be created as such but needs to be defined since an object or another abstract object derives from it. An object (e.g. pad, pocket, sketch, line, etc.) is an entity that can be created or selected and worked with it. Almost every object has a property and method. Property is a characteristic of the object (e.g. get radius size, get hole diameter, get chamfer angle, etc.), while a method is an action to be performed on the object (e.g. set radius size, set hole diameter, set chamfer angle, etc.).

Modern CAD software systems use predefined form features (extrusion, revolution, sweeping, etc.) for generation of 3D solid models. In the feature-based solid modelling technique, the modelling history is stored in a construction or feature tree, which is in CATIA V5 referred to as the specification tree. However, the final shape of solids is most often described by a widely used technique called B-Rep (Stroud and Nagy, 2011). The B-Rep technique creates a thin skin, which is in fact an interface between solid and non-solid. B-Rep models of solids are composed of two basic parts: the topology and the geometry. The topology defines the structure of the model, the geometry its shape. Topology can be divided into three main elements: faces, edges and vertices. Faces are surrounded by a set of edges, while vertices are points where several faces meet.

The CAD Idealiser search algorithm for detection of detail features is based on two main approaches: specification tree and B-Rep. The specification tree search approach is being conducted only for native CATIA models created in Part Design module, while the B-Rep approach is being applied for foreign models imported into CATIA from other CAD systems in the form of STEP or IGS file formats. To start the search process, only very simple user input in form of the maximum fillet radius r_{user} , chamfer size c_{user} , hole diameter d_{user} or notch thickness t_{user} is required. The search process is based on a decision tree algorithm where the main criterion to be fulfilled is that the detected feature properties are smaller or equal to user input values:

$$\begin{aligned}
 r_{\text{fillet}} &\leq r_{\text{user}} \\
 c_{\text{chamfer}} &\leq c_{\text{user}} \\
 d_{\text{hole}} &\leq d_{\text{user}} \\
 t_{\text{notch}} &\leq t_{\text{user}}
 \end{aligned}
 \tag{1}$$

The specification tree search approach fundamentally relies on detection of common CATIA part design features such as edge fillet, variable edge fillet, face fillet, tritangent fillet, auto fillet, chamfer, hole, pad and pocket. Each of these features is a CATIA automation object that possesses a corresponding set of properties and methods.

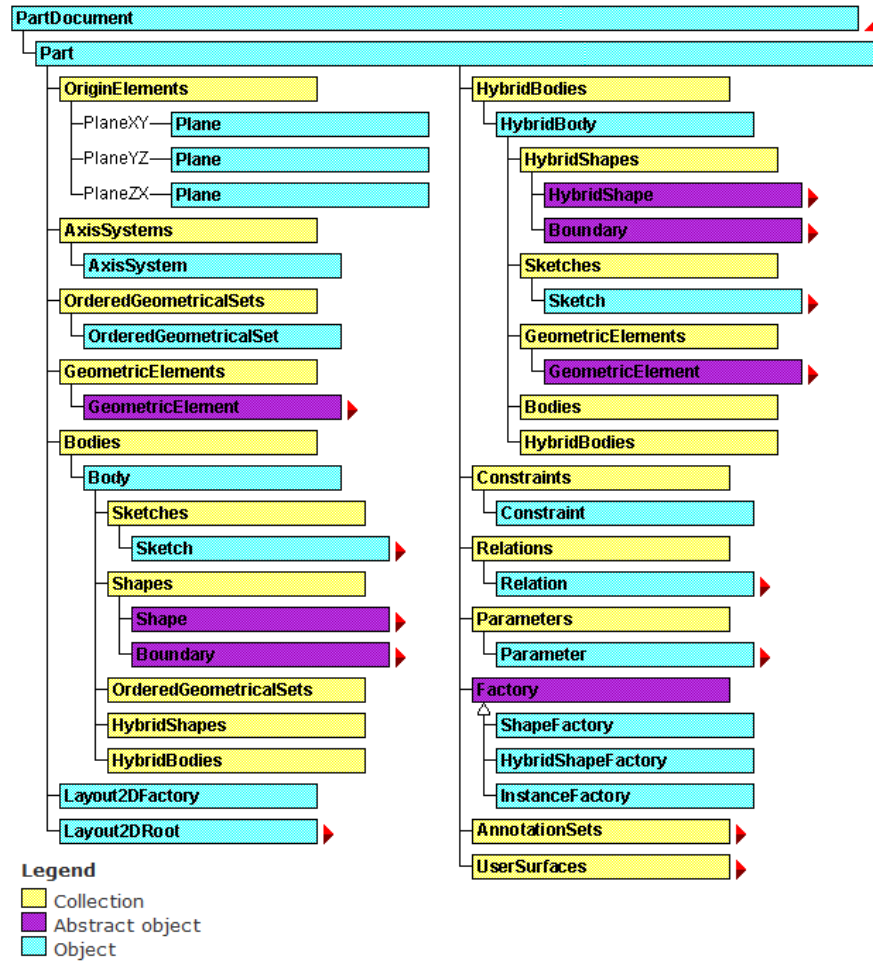


Figure 3. CATIA V5 Part Document automation objects (CATIA V5 R18 Help)

A similar principle in the algorithm was applied to all other mentioned automation features. However, during modelling process, some designers prefer to create fillets/rounds and holes indirectly through sketch-based features (see Figure 4), as this requires less computational effort. The algorithm is able to identify even such indirect features, as long as they are a result of the two most common sketch-based features: pad or pocket. In the first step, the algorithm identifies 2D circle automation objects which satisfy criteria: $d_{circle} \leq d_{user}$. In the second step, the identified circles are connected through their parent features to the corresponding topological elements (faces). However, it must be noticed that this will generally increase the overall search time. Mirror, liner and circular patterns are very often used during modelling process to replicate certain features (holes, ribs, symmetric parts, etc.). Those features are also taken into account by the search algorithm.

The B-Rep search approach, which is based on topology, is more complex and time consuming compared to the specification tree approach, but as well more powerful. The topology in CATIA V5 is divided into faces, edges and vertices. By measurement, faces can be classified by type into surfaces, cylinders, spheres, cones and planes, while edges into curves, circles and lines. A surface is a face that is curved in space. A cylinder is a face with straight parallel sides and a circular cross section. A cone is a face that tapers from a circular base to a point. A sphere is a round face with every point on its surface equidistant from its centre. A plane is considered as a flat and straight face surrounded by line edges. A line is a straight edge. A circle is a round planar edge whose boundary consists of points equidistant from the centre.

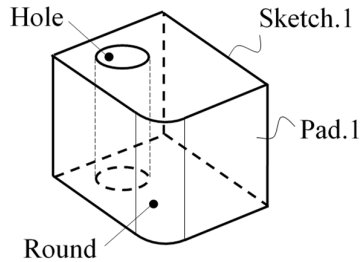


Figure 4. A round and hole created as a result of extrusion feature (Pad.1)

A curve is an edge that gradually deviates from being straight for some or all of its length. Edges, faces and vertices are boundary automation objects. However, the loops, which represent sets of faces connected by vertices bounding a face (CATIA V5 R18 Help), are for some reason not accessible via the VBA automation structure. In other words, this represents a serious limitation, as it is not known which edges and vertices belong to which faces, or vice versa. Alternatively, to overcome this limitation, the algorithm needs to conduct topological matching between faces and edges. Since the geometry continuously changes during the modelling process, topological objects are unstable. Therefore, to describe the topology, CATIA uses a unique generic name which is calculated by reading a kind of history graph of the topological objects (CATIA V5 R18 Help). The algorithm conducts topological matching between edges and faces by mutually comparing generic names.

The corresponding detailed features are represented only by a particular set of face types. Thus fillets/rounds are characterized by face types cylinder, sphere and surface. Chamfers are represented by plane and cone face types. Holes generally consist of cylinder pairs. Notches can be one of the following face types: plane, surface and cylinder.

Boundary automation objects (topological elements) possess a specific set of measurable properties such as length, centre of gravity (COG), radius, area, perimeter, volume, direction, axis, etc. The properties can be accessed by the so-called measurement automation object. To measure its property, the corresponding topological element needs to be selected first. Unfortunately, another disadvantage of CATIA V5 automation is that topological elements cannot be selected directly from the generic name. Thus for example, even the faces names are known from the generic name of the edge; they cannot be directly selected, i.e. measured.

The algorithm initially selects all the faces and edges, and groups them according to type (surface, sphere, cone, cylinder, plane, curve, circle and line). The grouped topological elements are stored together with their selection value, shortened generic name and specific properties (radius, area, perimeter, length, COG, axis, direction, etc.). The function of the selection value property is to enable reselection of the corresponding face or edge at some point of the algorithm. In the further steps the stored data is processed by the decision tree algorithm.

Once the identification of features has been successfully accomplished, the user can review the recorded results together with the corresponding property of each feature. Based on expert judgment, operations to suppress the features can be performed with the support of CAD Idealiser. When it comes to native parts, the defeaturing concept relies on deactivation of features in the specification tree. Although all features can be deactivated at once, certain caution is needed when proceeding so. For example, by deactivation of one feature, another feature might be deactivated as well, if they are in a parent-child relationship. Another example, some edge fillet feature might be set to tangent propagation mode and its deactivation could lead to the removal of the fillet at some edge where he's expected to be kept. Further, deactivation of some feature might produce an error in another feature caused by the loss of associated reference elements (e.g. projected vertices, edges, etc.) The best way to proceed is to carefully review all detected features, and ideally deactivate them one by one (or only several of them at once), and from the bottom to the top of the specification tree. For foreign parts, the defeaturing algorithm relies on generation of *Remove Face* features. This particular feature is able to remove a selected face or set of faces by extrapolation of adjacent faces, as shown in Figure 5. However, the user needs to manually move the desired faces to the suppression list box, which is then used as an input by the

algorithm to create *Remove Face* features. Additionally, the user needs to ensure that the set of faces is continuous and smooth in propagation.

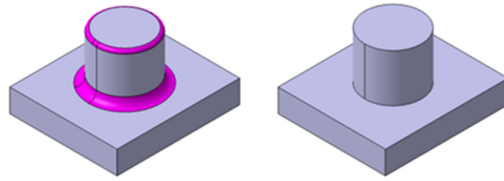


Figure 5. Example of applying *Remove Face* feature in CATIA V5

In some cases, the set needs to be further expanded or reduced in order to successfully create *Remove Face*. If the set of faces to be removed is not surrounded by a sufficient number of adjacent faces, or the extrapolation of them is far too complex, the creation of *Remove Face* can lead to failure. The best results can be achieved if the identified critical faces are removed step by step, instead of removing them all at once.

5. Verification

To verify the *CAD Idealiser* algorithm, five different CAD models have been subjected to idealisation process (Figure 6). The scope of investigation was focused on defeaturing of the CAD models, and not on the mesh generation and FEA results. Two groups were investigated: the native parts (modelled in CATIA V5 R18 environment) and the foreign parts (created by exporting native parts as STEP files and importing them again into CATIA).

Since generally one of the major goals of idealisation process is to reduce its duration, the research was focused on measuring the time required to detect and suppress features. The feature detection time can be measured precisely, while the suppression time is more subjective, as it is user dependent (the algorithm is not fully automated yet). Apart from the time, another important aspect of investigation was to determine feature detection efficiency of the algorithm, especially for the B-Rep based search approach. The feature detection efficiency can be expressed as the ratio between number of detected features and total number of features matching the search criteria.

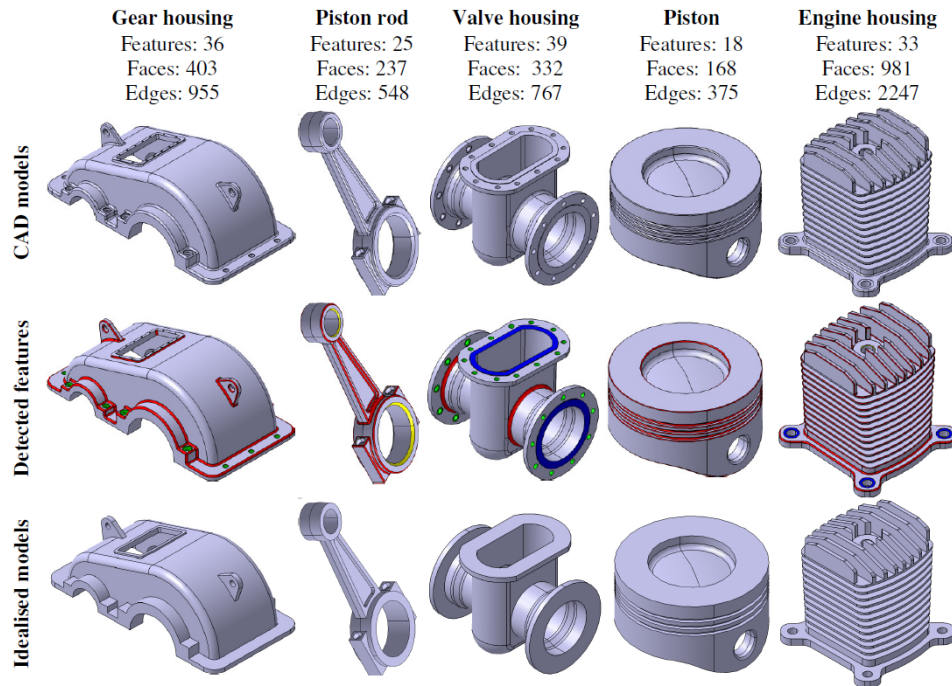


Figure 6. CAD models which were subjected to the idealisation process

CAD Idealiser decreases total idealisation time up to 70% compared to manual technique. When it comes to detection efficiency, specification tree-based search approach achieves 100%, while the B-Rep based approach achieves only slightly smaller detection efficiency (98–100%). The slightly smaller search efficiency is caused by the limitation of the algorithm to classify particular face types to the corresponding feature type. For example, if a face of type surface is surrounded by curved edges and is part of a fillet/round feature, the search algorithm is not able to determine the curvature of the surface or curve, which is a consequence of the CATIA design automation limitation. This means that in some rare cases features cannot be automatically identified, although their dimensions are part of the requested search range. Finally, the idealised models with all the critical features which were suppressed based on expert judgment are shown in Figure 6.

6. Conclusion

In the present work the design automation possibilities of commercial CAD software CATIA V5 were used to develop a CAD idealisation tool named CAD Idealiser. The macro supports the user in detection and suppression of geometric details (fillets, chamfers, holes and notches) in order to convert the CAD geometry into an analysis specific geometry suitable for mesh generation. CAD Idealiser is able to analyse native CATIA parts, as well as foreign parts from other software systems imported as neutral file formats (STEP or IGS). The search algorithm is based on decision tree taking into account user specified input. When it comes to native parts, features are detected by means of the specification tree, while features in foreign parts are detected by classification of topological entities (faces and edges). Since topological elements in CATIA V5 R18 are not correlated to each other, exhausting topological matching must be conducted. The feature suppression algorithm is not fully automated yet. However, it supports the user in suppression of features by deactivation (native parts) or by creation of remove face features (foreign parts). For verification of the algorithm, several different 3D parts have been subjected to the idealisation process. It was shown that the algorithm detects features efficiently and that the total idealisation time can be decreased up to 70% compared to manual technique.

As the work described here is limited to single part CAD models, the extension to assembly models is designated as a necessary next step. Additionally, further development of the algorithm will be focused on:

1. Fully automation of the feature detection algorithm in order to eliminate user input. Improvement of the algorithm to distinguish fillets and rounds (rounds are usually less important for analysis).
2. The B-Rep approach identifies each face as a particular feature. Improve the algorithm to recognize chained faces as one feature.
3. Automation of the defeaturing algorithm.
4. The Generative Shape Design (GSD) module in CATIA might be used to overcome the current exhausting topological matching problem. Basic idea is to generate Extract from all faces in the CAD model since this enables direct access to the edges and vertices. It also would have the advantage of leaving the original CAD model untouched, as the defeaturing would be performed on the extracted faces. Moreover, the extension to face clustering or even dimension reduction methods might become possible.

Finally, the present work shows that developing specific in-house applications using design automation can be of great benefit. Moreover, the extension of CAD software capabilities to detection and defeaturing of geometric details allows rapid preparation of analysis specific models directly at the origin of creation, instead of the CAE software.

References

- Attali, D., Boissonnat, J.D. and Edelsbrunner, H. (2009), "Stability and Computation of Medial Axes - a State-of-the-Art Report, Mathematics and Visualization", In: Möller, T., Hamann, B. and Russell, R.D. (Eds.), *Mathematical Foundations of Scientific Visualisation, Computer Graphics, and Massive Data Exploration*, Springer Berlin Heidelberg, pp. 109-125. https://doi.org/10.1007/b106657_6
- Benaouali, A. and Kachel, S. (2017), "An Automated CAD/CAE integration systems for the parametric design of aircraft wing structures", *Journal of Theoretical and Applied Mechanics*, Vol. 55 No. 2, pp. 447–459. <https://doi.org/10.15632/jtam-pl.55.2.447>

CATIA V5 R18 Help

- Dabke, P., Prabhakar, V. and Sheppard, S. (1994), "Using features to support finite element idealisations", *Proceedings of International Conference on Computers in Engineering, Vol. 1, Minneapolis, Minnesota, September 11-14, 1994*, ASME, pp. 183–193.
- Danglade, F., Pernot, J.P. and Veron, P. (2013), "On the use of Machine Learning to Defeature CAD Models for Simulation", *Computer-Aided Design and Applications*, Vol. 11 No. 3, pp. 358-368. <https://doi.org/10.1080/16864360.2013.863510>
- Del-Río Cidoncha, M.G., Martínez-Palacios, J. and Ortuno-Ortiz, F. (2007), "Task automation for modeling solids with CATIA V5", *Aircraft Engineering and Aerospace Technology*, Vol. 79 No. 1, pp. 53–59. <https://doi.org/10.1108/00022660710720494>
- Dey, S., Sheppard, M.S. and Georges, M.K. (1997), "Elimination of the adverse effects of small model features by the local modification of automatically generated meshes", *Engineering with Computers*, Vol. 13 No. 3, pp. 134–152. <https://doi.org/10.1007/BF01221211>
- Foskey, M., Lin, M.C. and Manocha, D. (2003), "Efficient computation of a simplified medial axis", *Journal of Computing and Information Science in Engineering*, Vol. 3 No. 4, pp. 274-284. <https://doi.org/10.1115/1.1631582>
- Gao, S., Zhao, W., Lin, H., Yang, F. and Chen, X. (2010), "Feature suppression based CAD mesh model simplification", *Computer-Aided Design*, Vol. 42 No. 12, pp. 1178–1188. <https://doi.org/10.1016/j.cad.2010.05.010>
- Gujarathi, G.P. and Ma, Y.S. (2011), "Parametric CAD/CAE integration using a common data model", *Journal of Manufacturing Systems*, Vol. 30 No. 3, pp. 118–132. <https://doi.org/10.1016/j.jmsy.2011.01.002>
- Hamdi, M., Aifaoui, B. and Benamara, A. (2007), "Idealisation of CAD geometry using design and analysis integration feature models", *Proceedings of ICED'07 / the 16th International Conference on Engineering Design, Paris, France, August 28-31, 2007*.
- Hamdi, M., Aifaoui, N. and Benamara, A. (2012), "CAD model simplification using a removing details and merging faces technique for a FEM simulation", *Journal of Mechanical Science and Technology*, Vol. 26 No. 11, pp. 3539-3548. <https://doi.org/10.1007/s12206-012-0869-6>
- Inoue, K., Itoh, T., Yamada, A., Furuhashi, T. and Shimada, K. (2001), "Face clustering of a large-scale CAD model for surface mesh generation", *Computer-Aided Design*, Vol. 33 No. 3, pp. 251–261. [https://doi.org/10.1016/S0010-4485\(00\)00124-X](https://doi.org/10.1016/S0010-4485(00)00124-X)
- Joshi, N. and Dutta, D. (2003), "Feature simplification techniques for free form surface models", *Journal of Computing and Information Science in Engineering*, Vol. 3 No. 3, pp. 177-186. <https://doi.org/10.1115/1.1603307>
- Lee, S.H. (2004), "CAD and CAE Integration Using Multi-Abstraction Non-Manifold Modelling Method", *Computer-Aided Design and Applications*, Vol. 1 No. 1-4, pp. 585-594. <https://doi.org/10.1080/16864360.2004.10738302>
- Lee, S.H. (2004), "Multi-resolution modeling for feature-based solid models using effective volumes of features", *Computer-Aided Design and Applications*, Vol. 1 No. 1-4, pp. 603-612. <https://doi.org/10.1080/16864360.2004.10738304>
- Lee, S.H. (2005), "A CAD-CAE integration approach using feature-based multi-resolution and multi-abstraction modeling techniques", *Computer-Aided Design*, Vol. 37 No. 9, pp. 941–955. <https://doi.org/10.1016/j.cad.2004.09.021>
- Lee, S.H. (2009), "Feature-based Multi-resolution and Multi-abstraction Non-manifold Modeling System to Provide Integrated Environment for Design and Analysis of Injection Molding Products", *Journal of Mechanical Science and Technology*, Vol. 23 No. 5, pp. 1331–1341. <https://doi.org/10.1007/s12206-009-0407-3>
- Mobely, A.V., Carroll, M.P. and Canann, S.A. (1998), "An object oriented approach to geometry defeaturing for finite element meshing", *Proceedings 7th International Meshing Roundtable*, pp. 547 – 563.
- Park, H.S. and Dang, X.P. (2014), "Structural optimization based on CAD/CAE integration and metamodeling techniques", *Computer-Aided Design*, Vol. 42 No. 10, pp. 889 – 902. <https://doi.org/10.1016/j.cad.2010.06.003>
- Quadros, W.R. and Owen, S.J. (2009), "Defeating CAD models using a Geometry-based size field and facet-based reduction operators", *Proceedings of the 18th International Meshing Roundtable*, Springer, Berlin, Heidelberg, pp. 301-318. https://doi.org/10.1007/978-3-642-04319-2_18
- Rezayat, M. (1996), "Mid-surface abstraction from 3d solid models: general theory and applications", *Computer-Aided Design*, Vol. 28 No. 11, pp. 905–915. [https://doi.org/10.1016/0010-4485\(96\)00018-8](https://doi.org/10.1016/0010-4485(96)00018-8)
- Robinson, T.T., Armstrong, C.G. and Fairey, R. (2011), "Automated mixed dimensional modeling from 2D and 3D CAD models", *Finite Elements in Analysis and Design*, Vol. 47 No. 2, pp. 151–165. <https://doi.org/10.1016/j.finel.2010.08.010>

- Russ, B.H. (2012), Development of a CAD model simplification framework for finite element analysis, PhD thesis, University of Maryland.
- Sheffer, A. (2001), "Model simplification for meshing using face clustering", *Computer-Aided Design*, Vol. 33 No. 13, pp. 925–934. [https://doi.org/10.1016/S0010-4485\(00\)00116-0](https://doi.org/10.1016/S0010-4485(00)00116-0)
- Sheffer, A., Blacker, T.D. and Bercovier, M. (1997), "Clustering: Automated detail suppression using virtual topology", *Joint ASME/ASCE/SES Summer Meeting*, ASME, pp. 57–64.
- Stroud, I. and Nagy, H. (2011), *Solid modelling and CAD systems: How to survive a CAD system*, Springer, London. <https://doi.org/10.1007/978-0-85729-259-9>
- Sun, R., Gao, S. and Zhao, W. (2010), "An approach to B–Rep model simplification based on region suppression", *Computers & Graphics*, Vol. 34 No. 5, pp. 556–564. <https://doi.org/10.1016/j.cag.2010.06.007>
- Thakur, A., Banerjee, A.G. and Gupta, S.K. (2009), "A survey of CAD model simplification techniques for physics-based simulation application", *Computer Aided Design*, Vol. 41 No. 2, pp. 65–80. <https://doi.org/10.1016/j.cad.2008.11.009>
- Wang, D., Hu, F., Ma, Z., Wu, Z. and Zhang, W. (2014), "A CAD/CAE integrated framework for structural design optimization using sequential approximation optimization", *Advanced in Engineering Software*, Vol. 76, pp. 56 – 68. <https://doi.org/10.1016/j.advengsoft.2014.05.007>
- Zhu, H. and Menq, C.H. (2002), "B–rep model simplification by automatic fillet/round suppressing for efficient automatic feature recognition", *Computer-Aided Design*, Vol. 34 No. 2, pp. 109–123. [https://doi.org/10.1016/S0010-4485\(01\)00056-2](https://doi.org/10.1016/S0010-4485(01)00056-2)

Mladen Burić, mag. ing. mech., Design Engineer
 University of Zagreb, Department of Design
 Zdenac 17, 52210 Rovinj, Croatia
 Email: mburic@stud.fsb.hr