Dormant deficiency: a novel concept to direct cause-effect CAD model analysis

Dormant Deficiency: A Novel Concept to Direct Cause-Effect CAD Model Analysis

Harald E. Otto OrcID: 0000-0002-4580-0429
Ferruccio Mandorli OrcID: 0000-0003-4864-5265

Department of Industrial Engineering and Mathematical Sciences, Faculty of Engineering, Polytechnic University of Marche, I-60131 Ancona, Italy

Corresponding Author: Ferruccio Mandorli, f.mandorli@staff.univpm.it

Abstract
Fully constrained features and sound associativity are prerequisites for robustness and alterability of parametric feature-based CAD models. However, errors in associativity are very difficult to detect with traditional static analysis approaches, due to effects that remain hidden until parameter changes and model re-creation take place. Currently, studies on associativity-related CAD model deficiency have not advanced to the point of being a part of model analysis. In this paper, the novel concept of dormant deficiency, together with a three-level classification, a graph-based knowledge network, a human readable visualization of cause and effect relationships, and a software tool are presented in a newly developed approach to dynamic CAD model analysis. Within this approach, dormant deficiencies are triggered to facilitate a methodic knowledge-driven method of detecting errors in associativity. This is achieved through systematic analysis of deficiency generating effects and their related symptoms, followed by systematic backtracking to their root causes. A selection of representative examples used for testing and evaluation of the approach is included within the empirical results from practice.

Keywords:
Parametric feature-based CAD model alteration, Design intent, Feature dependency, Robust CAD models, Dynamic CAD model properties, Formative feedback

Statements and Declarations
The authors have no competing interests to declare that are relevant to the content of this article.

1. Introduction

Developments in digital transformation (cf. Stark (2020), Moon and Park (2021)) are progressing rapidly worldwide. Integrated, information-driven approaches, such as product lifecycle management (PLM), aim to take into account all aspects of a product's life and its environment, from design through manufacture to final disposal (cf. Grieves (2006), Stark (2015), Elangovan (2020)). Within comprehensive approaches such as PLM, the feedback loop is of central interest. Within such product and process information feedback loops, output is examined in respect to a desired goal. Then inputs, models, and procedures are modified to regenerate output until the gap between what is desired and the respective output generated becomes acceptably small. As PLM is made up of various components including computer-aided design and computer-integrated manufacturing, three-dimensional spatial models are at the center of design, simulation, and evaluation.

The spatial product data and models allow for all this to be carried out in an entirely virtual digitized computer-based space where the so-called *information mirroring model*, also known as the *digital twin* (cf. Grieves (2006), Cerrone et al. (2014)) models all the characteristics and behaviors of a physical product. Creation of a digital twin starts before the physical product exists, and the digital twin can remain long after the final disposal of its physical counterpart. Within such 21st century concepts and developments, CAD models are becoming an indispensable pre-requisite, which requires them to be sound, flexible, and robust, so that they can be altered and regenerated without failure, while being further processed by various applications pertaining to individual product life cycle stages. This requires, among other factors, some means of reducing errors in associativity. These errors are usually introduced during modeling in the form of mistakes in the dependencies among geometric entities, features, and their respective parameters. They cause deficiencies and failures during CAD model regeneration. In general, it is quite challenging to specify sound modeling strategies which implement the correct feature-creation sequences and parameter settings that are required for robust parametric feature-based CAD models. Being able to design those sound modeling strategies is inherently difficult and represents a vast and complex problem space for creating robust CAD models. In order to make significant improvements, integrated solutions are required relating to the development of competency, skills, and software tools. However, the current state of development consists of rather isolated heuristics and approaches that can either reduce the problem space through methodology-based improvements to modeling strategies or help in detecting very limited sets of CAD model deficiencies after model creation has been finalized.

Recent work in this direction, such as the resilient modeling strategy (Gebhard (2017)), the explicit reference modeling methodology (Bodein et al. (2014)), and the robust sketch principles (Nerenst et al. (2023)), are promising approaches toward that goal. However, due to their limitations (see also discussions in Camba et al. (2016)), translation of those approaches into practice is still impeded and not widely pursued. In general, research and development for achieving sound, flexible, and robust CAD models are still in their infancy, and thus unable to provide systematic approaches and methods that can actually prevent errors in associativity. Thus, currently, in industrial and commercial settings, the creation of robust CAD models is mostly dependent on the skills and expertise of CAD users per se. In this context, it seems promising to aim first at supporting the efforts of CAD users – especially novices in educational settings – in the development of the skills and competencies required to create alterable CAD models. This can best be achieved by providing a framework and a software tool that can describe, help detect and trace the root causes of errors in associativity. In this manner, the aforementioned skill and competency development is supported through learning from errors by reflecting on mistakes in order to further insight and understanding.

The objective of the work presented in this paper is to provide a framework to support the dynamic analysis of altered parametric feature-based CAD models that fail during model regeneration due to errors in associativity. In particular, the aim is to support the user in the recognition and identification of CAD model deficiencies and in the provision of formative feedback aimed at supporting efforts to determine the causes of those deficiencies. The framework is based on the newly developed concept of dormant deficiency in its efforts to represent the phenomenon of errors in associativity. Those errors become evident in the form of deficiency generating effects and their related symptoms only through alteration and subsequent regeneration of inflexible and unrobust CAD models. The framework also utilizes systematic knowledge-driven means of using a network of linked graphs in combination with cause and effect diagrams to trace these effects back to their respective root causes. The concept of dormant deficiency was developed for and tested in the educational context to improve the learning experience. It provides, among other learning experience improving factors, a metric for students that facilitates a determination of the quality of a CAD model in respect to model robustness. Dormant deficiency represents a concept that comes to life for any user testing a parametric feature-based CAD model for errors in associativity. The framework also provides a classification of the type of dormant deficiency present, and determines its means of operationalization through a software-based feedback agent. However, the framework and approach are not limited to the educational context per se.

This paper is structured as follows. First, in section 2, an overview is provided of current developments in CAD model analysis and assessment, along with some background and discussion of current problems regarding robustness and alterability of parametric feature-based CAD models. Next, in section 3, an overview of the approach is given, along with details of the framework. The novel concept of dormant deficiency is outlined, and an explanation is given of how knowledge organization, encoding, and representation pertaining to root causes and effects associated with geometric entity and feature dependency are executed within the system architecture using the prototype software tool that has been developed and implemented. This is followed in section 4 by examples of analysis, backtracking, and subsequent knowledge formation and abstraction, showing the framework in action. In particular, this section focuses on how to use the software tool, and discusses cause and effect diagrams, as well as path graphs, to activate dormant deficiencies and correctly recognize and relate symptoms to the effects of particular dormant deficiency types. It then becomes possible to backtrack those symptoms to discover their actual root causes, which are, in fact, errors in feature associativity. Lastly, a brief summary is provided of outcomes achieved so far and an overview is given of work currently in progress. Some conclusions are drawn in the final section.

2. Background, scope, and objectives

2.1. Developments in CAD model analysis

Influence from global labor markets, together with educational reforms, has pushed CAD education further into the mainstream of higher education recently. This has resulted in a rapidly increasing number of students, with a wider than ever variety in educational backgrounds, enrolling in introductory level CAD courses and producing a flood of digital assignment submissions that need to be assessed and graded in a timely manner. In response to this overwhelming situation, work on software tools for autonomous grading, and also for quality and integrity checks, has been accelerated considerably in academia, and, to some extent, also in commercial sectors (cf. Bryan (2020), Garland and Grigg (2019), Johnson (2018), Joo (2018)). However, the frameworks and tools available for CAD model analysis and evaluation, and deployed within commercial and industrial settings, cannot be used directly in educational settings, due to differences in assessment criteria and evaluation goal settings. These differences focus mostly on issues related to application context, quality, and interoperability of CAD systems (see discussions and tool reviews in Gonzáles-Lluch et al. (2017), Gu et al. (2001)).

Within educational settings, recent approaches dedicated to the automation of CAD model analysis, assessment, and grading are obviously capable of considerably reducing the time required for analyzing and assessing CAD models created by students. However, the type and complexity of CAD model that can be analyzed, as well as the quality of the feedback that is generated (cf. Irons (2008), Hattie and Timperley (2007)), are still quite limited. This is apparent, for example, in approaches for technical drawings and 2D CAD files that can be found in Bryan (2020), Hekman and Gordon (2014), Ingale et al. (2017), Khaleel et al. (2020), and Younes and Bairaktarova (2022). Examples of recent approaches for 3D CAD models and related empirical studies are reported in Ault and Fraser (2013), Gonzáles-Lluch et al. (2019) and Kirstukas (2016). An interesting approach to providing visual feedback for automated CAD model grading using heat maps is reported in Jaakma and Kiviluoma (2019). Further discussions on the subject of automated CAD model grading, including a summary of the literature and pointers to gaps in research, can be found in Garland and Grigg (2019). Another major limitation of those approaches to the automation of CAD model analysis and grading is the inherent functional structure of having scripts and other pre-defined computerized means that are only capable of comparing submitted student work with a reference solution and determining what is correct or not in an inventory-checking-like manner. Hereby, the data space of errors, mistakes, and deficiencies – that were introduced into the CAD models by students during model creation – remains largely ignored. However, that is precisely the area where potentially valuable knowledge can be extracted to gain insight into the shortcomings and difficulties with which students may have struggled.

In regard to the computer-aided analysis of the alterability of CAD models – which represents a central part of CAD model quality – it seems that dedicated frameworks and software tools in both industrial/commercial settings and educational settings are not yet available. Perhaps this strongly reflects on the central shortcoming of most approaches, previous and current, which is their inherently static approach, namely focusing during CAD model analysis on only one version and one status of a model. Approaching a problem related to a dynamic issue such as analysis of the alterability of CAD models requires a newly envisioned approach characterized by adequate identification and formulation of the problem, so that an appropriate solution can be found.

2.2. Motivation, scope and objectives

With increasing digitalization and the proliferation of PLM, the flexibility and robustness of parametric feature-based CAD models are becoming issues of increasing relevance for research and practice in both academic/educational and industrial/commercial environments. These issues can be approached from various viewpoints. The two most prominent being pursued are aimed at determining ways to reduce or even prevent alterability-related issues and developing means to circumvent them. For example, in practice, alterability issues have been approached by commercial parametric CAD system developers through enrichment of the functionality of CAD modeling systems. This was achieved by allowing local modifications of the CAD model by means of a push/pull approach. This extended functionality of CAD modeling systems is now widely known as explicit and direct modeling. Although those modeling techniques do have features and feature-related parameters, they are defined on the fly during modeling. Therefore, those CAD modeling approaches are not able to fully take into account the same set of functionalities and operationalization techniques in regard to parametric feature-based modeling that asynchronous approaches do. Hence, those modeling techniques are missing out on some benefits of parametric feature-based modeling. This is especially relevant in regard to the creation of robust alterable feature-based CAD models, which are usually centered around a history-based sound modeling strategy appropriate to the design intent (see also discussions in Aranburu et al. (2022), Zou et al. (2023)). One needs to view these asynchronous and synchronous approaches from two different perspectives: CAD model creation from scratch and modification of CAD models. Further, one needs to take into account the complexity of the model and whether it is parts, part families, or assemblies that are being modeled. Approaches aimed at considerably reducing alterability-related issues include efforts to develop modeling guidelines and best practices, and new modeling methodologies that are based on robust modeling strategies and robust sketching principles. However, at the current stage of development, all these approaches still have considerable limitations impeding their full application in CAD practice. Moreover, despite their benefits for improving the robustness of parametric CAD models, it is by no means guaranteed that they will prevent the occurrence of errors in associativity. As these approaches are all developed from the perspective of prevention, they are not able to cope adequately with failures during model regeneration due to errors in associativity. Hence, they lack any means of providing effective feedback to the user, who otherwise has limited support from the CAD system for determining the cause of such model failures when they happen.

In regard to the matters of course mentioned above, the authors propose an approach based on a framework newly developed to tackle issues of CAD model alterability in a human-user and knowledge-centered manner. In particular, a systematic application of the new approach aims at the following. Firstly, formative feedback and learning from errors should be supported through affording the identification and better understanding of the causes that are behind the creation of inflexible and unrobust CAD models. Secondly, the expansion of knowledge pertaining to errors in associativity and CAD model robustness should be facilitated, and this will translate into support for a more detailed, insight- and knowledge-driven definition of modeling guidelines and best practices. Thirdly, hopefully the way will be paved for a more sophisticated and adequate definition of next-generation methods and software tools for CAD model analysis. The framework proposed is based on – among other components – the novel concept of dormant deficiency newly introduced to the CAD field, and on systematic cause and effect backtracking analysis that permits the identification and subsequent removal of the root causes that are the very essence of inflexible and unrobust parametric CAD models. Thus, this approach allows for a systematic and dynamic

CAD model analysis supported by a software tool that facilitates backtracking along a structured reasoning chain proceeding from symptom to effect, and ultimately to cause.

Traditional CAD model analysis is focused on the static properties of the models, that is, topological and geometric characteristics. Those characteristics are always analyzed within a particular model configuration, which, in most cases, is the configuration that is reached at the end of the modeling process. Finally, to assess the quality of the CAD model, those static properties are usually compared to their counterparts in a reference CAD model or reference solution. The inherent limitation of this approach is that there is no chance to capture and analyze the dynamic nature of a parametric CAD model. This is because a CAD model that appears to be sound and free of errors within a particular configuration can still become inconsistent and deficient within another configuration, due to model alteration. To tackle this issue, the authors propose a dynamic analysis approach and the novel concept of dormant deficiency. These provide a means to describe, identify, and analyze CAD model deficiencies caused by errors in associativity that become evident – through model deficiency generating effects and their related symptoms – only after a model alteration takes place.

Although a long road still lies ahead before reaching the goal of considerably reducing or even preventing alterability-related issues in the creation of parametric CAD models, the contribution made by the development of modeling guidelines and best practices cannot be denied. However, the rationale behind a certain modeling strategy suggestion within an application context is not always clear to the end user. This can result in modeling situations where the user is uncertain whether a particular guideline and best practice can really yield a benefit over another method. Moreover, if the method adopted fails to produce the result expected, there is usually no way for the user to know the cause of the failure. In such situations, the backtrack analysis that is part of the novel approach developed can be integrated with best practices and guidelines to help in amending some of these issues, as it supports the user and enables him or her to get to the bottom of the CAD model regeneration failure, that is to discover the root cause of the deficiency generating effects and their related symptoms.

3. Approach, framework, and development

3.1. Outline and approach

Modern parametric feature-based CAD systems provide various mechanisms to help a user to define and create the parameterization of a CAD model. Most of these mechanisms are applied at the two-dimensional feature profile level and they can be sub-divided into two groups, namely geometric constraints and dimensional constraints. Geometric constraints are used to define relationships among geometric entities. They can be applied by selecting the type of constraint and the entities to which the constraint must be applied. The order of selection may vary depending on the CAD system being used. When the constraint needs to be applied to special vertices linked to spatial locations, such as the center of an arc, or the midpoint of a segment, snap points can be used. The entities that can be selected for the constraint definition are the elements of the profile and elements of the model geometry that are projected in the profile plane. Geometric constraints constrain the morphology/topology of a CAD model as they preserve geometric properties. Dimensions are used to set the size and/or position of the geometric entities. A dimension can be active or passive, and these are referred to as *driving dimension* and *driven dimension*, respectively. Usually, a change in the value of a driving dimension activates the regeneration of a CAD model. The user can add driving dimensions until a profile becomes fully constrained. The value of a driving dimension can be either a fixed numerical value or a formula. Here the latter is used in its broader sense that includes algebraic equations, sets of predefined values, links to external data, etc. Driving dimensions constrain the size of the model as they provide dimensional variability. Further mechanisms used to define and create the parameterization of a CAD model are available at the three-dimensional level. Means for selection of the plane in which to locate feature profiles and definition of the extent of the extrusions of a feature are such examples. Critical modeling situations within which mistakes may result in errors in associativity can occur during the creation of both the feature tree and the parameterization.

Now, in order to analyze CAD models in regard to flexibility and robustness, the six general steps of the proposed approach can be outlined in a simplified manner as follows: (1) take a parametric feature-based CAD model, (2) make changes to the CAD model, (3) detect dormant deficiencies, (4) analyze dormant deficiencies, (5) find root causes, (6) document knowledge obtained through analysis, such as root causes.

Here, within the framework presented in this paper, dormant deficiencies are activated in a CAD model to determine whether the model actually contains any dormant deficiencies, and, if so, to find the type of deficiency and the root cause. This is performed through a CAD model alteration simulator that consists of a module integrated with an implemented software tool that is referred to as a *dormant deficiency detection tool* (DDD tool) as described in Mandorli and Otto (2022) (cf. step 2). The role of the DDD tool is to support the analysis process by systematically changing various CAD model parameters one at a time and providing means for the identification of various types of activated dormant deficiency (cf. step 3). The presence of any dormant deficiency becomes evident through model deficiency generating effects and their related symptoms. The nature of the symptoms, which come in the form of feature failures, error messages, and shape defects, helps to determine the type of dormant deficiency and the effect with which a symptom is associated (cf. step 4). Once the effect of a particular dormant deficiency has been determined through its symptom, the framework provides further support through its knowledge network, where knowledge on cause and effect relationships is stored for each type of dormant deficiency. This knowledge network can be traversed to backtrack from an effect to its probable root cause (cf. step 5). As the knowledge is encoded through sets of attributed graphs which can be mapped to a veridical graphical representation in the form of cause and effect diagrams, and vice versa, information required during cause and effect analysis is available in both a graphical representation for the human user and an encoded internal representation for computerized processing (cf. step 6). Backtracking to the root cause allows for shedding some light on and finally determining the inter-/intra-feature associativity issue, which was the cause of the dormant deficiency subject to analysis.

3.2. The concept of dormant deficiency

3.2.1. Overview and description of types

Dormant deficiencies can be conceptualized as errors in associativity, which were introduced during the modeling process due to mistakes in the specification of dependencies between geometric entities and features. However, the effect and impact of those mistakes on the CAD model remain dormant until an actual CAD model regeneration is triggered and executed through an alteration (cf. Figure 1).

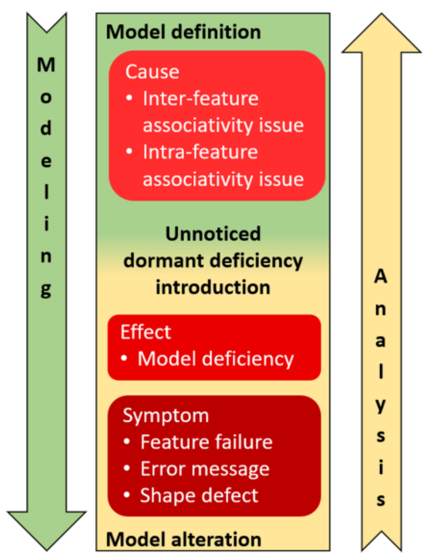


Fig. 1. Structured overview of the approach and the role of dormant deficiencies.

In this context, the outcome in regard to deficiencies is related to which of three different error situations occurs. Those are, in turn, related to features and their status, shape (topology/geometry), and design rationale. Accordingly, dormant deficiencies are classified as type I, type II, or type III, as follows.
Note that in the descriptions below the term 'feature' refers to the result of the application and execution of a modeling command from within a parametric feature-based CAD system, resulting in an entry in the feature history tree, and to all the topological/geometric entities generated and associated with such a feature history tree entry.

Type I dormant deficiencies: Faults in status

The effect of this type of deficiency is that a model change leads to features being either regenerated with unpredictable results or not regenerated at all.
The main symptom of a type I dormant deficiency which has been activated is a change in the feature status. The regenerated CAD model contains deficient features, which are labeled with a warning or failed status in the feature history tree.
The DDD tool provides fully autonomous detection of type I dormant deficiencies.

Type II dormant deficiencies: Faults in shape

The effect of this type of deficiency is that a regenerated CAD model does not contain any features labeled with a warning or failed status, but the shape of features is incoherent or even destroyed, such as in cases where a blind hole has become a through hole or an open slot has become closed.
The main symptom of a type II dormant deficiency that has been activated is a change in the number of geometric entities of the features, such as the number of side faces of a slot.
The DDD tool provides semi-automatic detection of type II dormant deficiencies.

Type III dormant deficiencies: Faults in design rationale

The effect of this type of deficiency is that the regenerated CAD model does not contain any features labeled with a warning or failed status, nor any features with a shape that is incoherent or destroyed. However, the shape does not meet the design requirements due to a loss of elementary function. The main symptom of a type III dormant deficiency which has been activated is a change in geometric properties such as lost symmetry conditions.
The DDD tool provides visual support for the detection of type III dormant deficiencies.

Note that in comparison to earlier descriptions in work by the authors (cf. Mandorli and Otto (2022)) referring to dormant deficiencies, those given above reflect a stronger and more explicit viewpoint in regard to both the effects and the symptoms. Integration and explicit use of the concept of symptoms along with the concept of effects right from the beginning, at the description level of dormant deficiencies, not only allows for more precise distinction and improved handling of them, but also supports efforts dedicated to their detection. Accurate detection is a crucial and necessary precondition for correctly understanding effects, and this understanding is, in turn, required to facilitate an adequate backtracking to discover the cause factors through which dormant deficiencies were introduced in the first place.
At this point, it should be made explicit that the type III dormant deficiency is the least specific type, while the type I is the most specific type, with the type II in an intermediate position. Therefore, the type III dormant deficiency, being the most general type of dormant deficiency, subsumes type II, which, in turn, subsumes type I. For example, if, after a CAD model regeneration, a previously sound feature emerges with a failure or warning status, those symptoms imply that a feature has not been regenerated properly, and thus a type I dormant deficiency has been activated. Additionally, if a feature has not been properly regenerated, or has not been regenerated at all, resulting in a fault in the shape, this is also consistent with a type II dormant deficiency. These effects in turn provide strong evidence that the functionality related to this feature has been lost, thus causing faults in the design rationale, which is also consistent with a type III dormant deficiency. However, if, for example, after a CAD model regeneration, a previously correctly positioned feature emerges in a different location or spatial position or orientation that violates its previous geometric property – in regard to actual design requirements of the CAD model – without any further faults in either status or shape, this indicates only the presence of a triggered type III dormant deficiency, because the symptom and effect do not relate to those of the more specific type II and type I dormant deficiencies.

3.2.2. Structuring of symptoms, effects, and causes

Symptoms and effects

In the case of type I dormant deficiencies, symptoms appear in the form of a change represented by labels indicating a warning or failure in the status of features, due to errors and deficiencies encountered during feature regeneration after a CAD model has been altered. As these symptoms are reported by the CAD system in relation to individual effects that were encountered during CAD model regeneration – usually as a result through the geometric modeling kernel – information on both the symptom and the related effect can be obtained from the CAD modeling environment. Within the framework and current implementation, as presented in this paper, symptoms and effects related to seven feature warnings and eleven feature failures (see also Verma and Weber (2021)) are encoded and integrated. Some representative examples are as follows.

- The input profile does not generate a valid part, resulting in no material being removed.
- Zero thickness (non-manifold body), resulting in an unsuccessful execution of operation.
- Sick parent in reference element (plane), resulting in a recompute warning.
- The parent or keypoint end of the cylinder selected is missing, resulting in a recompute warning.

This change in the manner of intersecting feature shapes is due, in turn, to a change in the dimension or position of those features. Dimensions and positions of features depend on the reference plane where the feature profile is located, the extrusion options, and the way the profile has been dimensioned and constrained. Therefore, in general, individual symptoms regarding a change in the number of geometric entities of a feature can be associated with specific cause and effect relationships, which facilitate the analysis and backtracking to causal factors and the root cause of a type I dormant deficiency. This can be approached by systematically examining the topology of features and their shapes – based on various types of entities such as faces, loops, and edges – that are subject to analysis. In cases where a change has been detected, the nature of the change is evaluated and related to an effect that is considered to be consistent with a type II dormant deficiency.

Note that, where symptoms and effects of a triggered type I dormant deficiency are detected and reported in one form or another by commercially available parametric feature-based CAD systems, most, if not all, of those detections can be considered sound and consistent, as they are principally based on the fundamentals of Brep-based solid modeling theory. However, suggestions for a possible root cause are seldom communicated by those systems, and, if they are, the suggestions are usually misguided. Moreover, corrective actions that are sometimes proposed by those systems are not necessarily always correct and consistent in regard to the actual effects as reported, and thus remain less helpful to the user.

In the case of type II dormant deficiencies, symptoms appear in the form of a change in the number of geometric entities of a feature, due to a change in the manner a feature shape intersects with other feature shapes. This change in the manner of intersecting feature shapes, in turn, is due to a change in the dimensions or positions of those features. Dimensions and positions of features depend on the reference plane where the feature profile is located, the extrusion options, and the way the profile has been dimensioned and constrained. Therefore, in general, individual symptoms regarding a change in the number of geometric entities of a feature can be associated with specific cause and effect relationships, which facilitate the analysis and backtracking to causal factors and to the root cause of the type II dormant deficiency. This can be approached by systematically examining the topology of features and their shapes – based on various types of entities such as faces, loops, and edges – that are subject to analysis. In cases where a change is detected, the nature of the change is evaluated and related to an effect that is considered to be consistent with a type II dormant deficiency. Some representative examples from the framework, as presented in this paper, are as follows.

- The total number of faces of a feature changes to zero, resulting in a feature being deleted.
- The number of bottom faces of a volume-removing feature changes to zero, resulting in a blind cutout becoming a through cutout.
- The number of loops in a top face is decreased, resulting in an internal loop (cutout or a protrusion) intersecting the side of the feature.

- The number of side faces increases, resulting in a feature that contains a split side.

Note that most, if not all, of these symptoms and effects that are caused by a triggered type II dormant deficiency, and are frequently encountered during CAD model analysis, remain undetected by all commercially available parametric feature-based CAD systems.
In the case of type III dormant deficiencies, the overall effect is a loss of design rationale. This loss of design rationale involves a portion of the overall shape of the component, which can no longer provide the function for which it has been designed. Functions that need to be preserved can be related, for example, to assembly conditions, precise positioning, motion guidance, friction conditions, and strength distribution. In such cases, symptoms appear in the form of a change in the geometric and topological relationships that are defined through geometric entities of a single feature (intra-feature context) and/or a set of related features (inter-feature context). Changes can also occur in the feature geometry, resulting in a shape that is neither incoherent with the feature nor destroyed (see also work on functional dimensioning in Otto and Mandorli (2015)). In general, such changes become visible after a CAD model regeneration in the form of a change in the dimension, position, and/or shape of those features. Dimensions and the positions of features depend on the reference plane where the feature profile is located, the extrusion options, and the way the profile has been dimensioned and constrained. In principle, those individual symptoms can be associated with specific cause and effect relationships, which facilitate the analysis and backtracking to causal factors and to the root cause of the type III dormant deficiency. However, to formulate and correctly apply those cause and effect relationships, the design requirements and constraints, that is the design rationale and engineering context of the CAD model subject to analysis, need to be known. In cases where such changes can be detected, the nature of the change can be evaluated and related to an effect that is consistent with a type III dormant deficiency. Some representative examples from the framework, as presented in this paper, are as follows.

- The coaxial center axes of a boss feature and its related threaded hole feature change to non-coaxial center axes, due to a change in the spatial location of those two features, caused by an incorrect selection of the keypoint to constrain the position of the hole center. Within a particular design context where protruding mounting features and fixture screws are used, this causes the loss of an important axis symmetry, resulting in unbalanced mechanical strength distribution.
- The distance between two mechanical fixing holes modeled with hole features changes, due to a change in the spatial location of those two features, which, in turn, is caused by incorrect constraining of the reciprocal positioning of the holes. A change in the spatial location of the fixing holes will have an adverse effect on the assembly, resulting in an assembly problem.
- The location of a keyway housing changes, due to a change in its spatial location, which is caused by an incorrect definition of the dimensioning of the keyway position. Such a change in the keyway position will result in an assembly problem and probably in erroneous torque transmission.

Note that all of those symptoms and effects that are caused by a triggered type III dormant deficiency are beyond what current commercially available parametric feature-based CAD systems are capable of handling. Hence, they remain undetected in CAD models in both educational settings and commercial/industrial settings.

Root causes and associativity

The nature and category of dependencies between geometric entities and features are determined by their range in regard to features, among other characteristics, and by the class of features with which they are associated, that is to say profile-based features or non-profile-based features, as listed in Table 1. This results in intra-feature dependencies and inter-feature dependencies. The former represent dependencies between geometric entities within one and the same feature, while the latter refer to dependencies between geometric entities of more than one feature. From this viewpoint, particular modeling situations can be grouped as listed in Table 2, where dormant deficiencies have been introduced, most likely by novices, and thus turn into critical situations (cf. Otto and Mandorli (2018)).

Profile-Based	Extruded cutout, Extruded protrusion Revolved cutout, Revolved protrusion Sweep, Loft, Helix Hole, Slot, Rib, Pattern
Non-Profile-Based	Round, Chamfer, Draft Mirror copy, Shelling

Table 1. Structured overview of feature types used within the framework, which are classified as either profile-based or non-profile-based.

	Intra-Feature	**Inter-Feature**
Profile-Based	profile creation	reference plane selection profile creation extrusion definition
Non-Profile-Based	N/A	entity selection

Table 2. Modeling situations in respect to feature class and range, that become critical if dormant deficiencies are introduced.

Note that, in the case of profile-based features, deficiencies within intra-feature dependencies are most likely to be introduced during profile creation, when associativity is created between the 2D geometric entities of the profile. Here, in cases where the profile is comprised of a basic non-complex outline, the CAD system usually creates rudimentary geometric constraints automatically. However, in the case of complex profiles, the user is required to explicitly define all the constraints required. Currently, most commercially available CAD systems provide functionality for testing the condition of those profiles, that is, whether they are fully constrained or not.

Dormant deficiency type	**Symptom**	**Effect**
Type I	change in feature status indicated through warning and failure labels	features are not regenerated or regenerated with unpredictable results
Type II	change in topology indicated through the number of geometric entities of a feature	feature shape is incoherent or even destroyed
Type III	change in geometric properties	loss in (mechanical engineering) functionality

Table 3. Overview of symptoms and effects in regard to dormant deficiency type.

3.2.3. Knowledge organization and representation

To support CAD model analysis and the determination of cause factors and root causes of dormant deficiencies, knowledge – on cause and effect relationships in regard to dormant deficiencies as presented elsewhere in this paper – has been structured and encoded into a network of linked attributed *r*-partite graphs as follows.

Preliminaries and terminology

First, some preliminaries on the basic structural concepts and terminology of graphs used in this paper, which are mostly oriented on the basics as presented in Beineke (2014), Diestel (2017) and Gross et al. (2019). Given a vertex set denoted by V and an edge set denoted by E, a heterogeneous relation defined as R: $V \rightarrow E$, can be characterized by a graph that is a pair $G = (V, E)$ of sets such that $E \subseteq V \times V$, with elements of E being 2-element subsets of V and $V \cap E = \emptyset$. The vertex set of a graph G is referred to as

$V(G)$ and its edge set as $E(G)$. The set of all the edges in E at a vertex v is denoted by $E(v)$. An edge denoted by $\{v_m, v_n\}$, is also written as (v_m,v_n), or simply v_m,v_n. However, if $v_m \in A$ and $v_n \in B$, then v_m,v_n represents an *A-B edge*. A graph G with vertex set V is also said to be a graph on V. To reduce the complexity of notation, we may replace the notation $v \in V(G)$ and $e \in E(G)$ with the shorter versions $v \in G$ and $e \in G$, respectively, thus being less strict about a graph G and its vertex or edge set in cases where this less strict viewpoint is applicable. The number of vertices of a graph G is its *order* and is denoted by $|V(G)|$. The number of edges of a graph G is denoted by $|E(G)|$. A vertex v is *incident* with an edge e, if $v \in e$ that is e is an edge at v. Moreover, two vertices are *endvertices* or simply *ends*, if they are incident with an edge. The two vertices v_m and v_n with $v_m \in G$ and $v_n \in G$ are *adjacent*, if $\{v_m, v_n\} \in G$. Two edges with $e_m \in G$ and $e_n \in G$ and $e_m \neq e_n$ are *adjacent* if they have an end (vertex) in common. A set $\mathcal{A} = \{A_1,..., A_n\}$ of disjoint subsets of a set A is called a *partition* of A if the union $\bigcup\mathcal{A}$ of all the sets $A_i \in \mathcal{A}$ is A and $\forall i$ with $i = 1...n$ $A_i \neq \emptyset$. Another partition $\{B_1,..., B_m\}$ of A is called a *refinement* of partition $\mathcal{A}$, if each B_i is contained in some A_j. If the context permits, those refinements of a partition $\mathcal{A}$ are simply referred to as *refining sets*. The sets and subsets with n elements are called *n-sets* and *n-subsets*, respectively. The set of all n-element subsets of A is denoted by $[A]^n$. We may speak of an r-partitioned graph denoted by $\dot{G}$, also referred to as an r-partite graph, if $V(\dot{G})$ admits a partitioning into r subset classes with $E(\dot{G})$ as a set of edges such that every edge has its ends in a different set (cf. Beineke (2014)). A *walk* in a graph G denoted by $W(G)$ represents an alternating sequence of vertices and edges defined as $W(G) = v_0,e_1,v_1,e_1, \ldots e_n,v_n$ with the vertices v_{j-1} and v_j being the endpoints of the edge e_j, v_0 being the *initial vertex*, v_n being the *terminal vertex*, and $v_1, \ldots, v_{n-1}$ being the *inner vertices*. This representation can be simplified by considering only sequences of adjacent vertices defined as $W(G) = v_0,v_1, \ldots ,v_n$. A *trail* in a graph G denoted by $T(G)$ is a walk in which no edge occurs more than once. A *path* in a graph G denoted by $P(G)$ – or simply denoted by P if the context permits – is a trail in which no inner vertex is repeated (cf. Gross et al. (2019)). The number of edges of a path is its *length*, written as $|E(P(G))|$. A path with $|V(P(G))| = |E(P(G))| + 1$ is also called a *path graph*. A path $P(G) = v_0,v_1, \ldots ,v_n$ is called a path from v_0 to v_n. If $V(P) \cap A = \{v_0\}$ and $V(P) \cap B = \{v_n\}$ it is also an *A-B* path. If the union $P(G)v \cup vQ(G)$ of two paths is again a path, this is denoted simply through path concatenation by PvQ. The inner path of $P(G)$ within a graph G, that is the sub-path denoted by $^{\circ}P(G)$ is defined such that $^{\circ}P(G) = v_1, \ldots, v_{n-1}$ where v_0 , $v_n \in V(P(G))$ and v_0 , $v_n \notin V(^{\circ}P(G))$.

Partitions and refinements

Now, in regard to a knowledge network denoted by *KN* concerning effects, causes, and root causes – that are defined through their respective main categories denoted by *EC*, *CC*, and *RC* – of dormant deficiencies denoted by *ECR*, a dormant deficiency analysis partition of *ECR* denoted by *DDA* can be defined as $DDA = \{EC^I,EC^{II},EC^{III},EC^B,CC^I,CC^{II},CC^{III},CC^B,RC\}$. Here the partition $\{EC^{I,1},..., EC^{I,m}\}$ refines the set of EC^I, which encodes all effect categories that are specific to type I dormant deficiencies. Similarly, the partitions $\{EC^{II,1},..., EC^{II,n}\}$ and $\{EC^{III,1},..., EC^{III,k}\}$ refine the sets EC^{II} and EC^{III}, which encode all effect categories that are specific in respect to type II and type III dormant deficiency. In a similar manner the sets CC^I,CC^{II},CC^{III}, which encode all cause categories that are specific to each respective dormant deficiency type, are refined by the cause (factor) partitions defined as $\{CC^{I,1},..., CC^{I,m}\}$, $\{CC^{II,1},..., CC^{II,n}\}$, and $\{CC^{III,1},..., CC^{III,k}\}$. Finally, the sets EC^B and CC^B, which encode the cause and effect categories that are considered basic to dormant deficiencies, are refined by the partitions $\{EC^{B,1},...,EC^{B,5}\}$ and $\{CC^{B,1},..., CC^{B,n}\}$. Note that in the case of EC^B the refining partition consists only of five subsets, because those basic effects are linked to the basic modeling situations – which can become critical if a dormant deficiency is introduced – regarding feature class and range, as described elsewhere in this paper (see also again Table 2 and Table 3).

Cause and effect relations

Based on the refined partitioning of *ECR* as described above, knowledge on individual cause and effect relationships within *KN* can be encoded by attributed r-partite graphs as follows. First, to obtain an unambiguous set of cause and effect relations, the subsets within each refining partition of EC^I, EC^{II}, EC^{III}, and EC^B need to be structured as one-element subsets. Next, relationships between an individual effect and associated cause factors within a cause category in regard to a type I dormant deficiency are encoded as 2-element subsets $E(\dot{G}^{I,i})$ represented by sets of edges such that every edge has its ends in a different set within an attributed *r*-partite graph $\dot{G}^{I,i}$. In particular, all edges $E(\dot{G}^{I,i})$ for $i = 1\ldots n$ must have one of their ends in a refining set of CC^{I}. The partitions of $\dot{G}^{I,i}$ typically consist of the refining subsets associated with EC^{I}, EC^{B}, and CC^{I}. Similarly, relationships between an individual effect and associated cause factors of a cause category in regard to type II and type III dormant deficiencies are encoded through sets of attributed *r*-partite graphs that are structured in a manner like their counterparts for the type I – of course considering then the refining subsets associated with EC^{II}, EC^{III}, etc. – and are denoted by $\dot{G}^{II,1}$, ..., $\dot{G}^{II,m}$, and $\dot{G}^{III,1}$, ..., $\dot{G}^{III,k}$. In the case of encoding relationships between an individual basic effect and associated cause factors of a basic cause category, edges of any associated attributed *r*-partite graph $\dot{G}^{B,m}$ that have one end in a refining set of CC^{B} and the other end not in a refining set of EC^{B}, always have ends in *RC*. This is consistent with the fact that a basic effect can usually be related to basic cause factors, which, in turn, can be traced back to the root cause of a dormant deficiency that is encoded as part of the subset *RC* of the dormant deficiency analysis partition *DDA* as defined earlier.

Vertex attributes

Attributes of the knowledge network *KN* are realized through vertex labeling of graphs $G = (V, E)$ that is based on a map va: $G \times DDA \times V(G) \times Name \rightarrow Label \times Color$, where the elements of the set *DDA* represent the categories that are available within the framework in regard to effect, cause, and root cause of the dormant deficiency. $V(G)$ is a vertex subject to ascription, that is to attribute application. *Name* is the name of the attribute. *Label* represents the attribute label with a value that is defined as a string of characters. The elements of *Color* represent the colors that are available within the framework and that were used along with set partitioning. Note that adjacent vertices, that is, pairs of vertices joined by an edge, must not have the same color. This condition is consistent with the approach taken to organize knowledge on cause and effect relationships as described elsewhere in this paper.

Analysis path structuring

Based on the above-described knowledge network *KN* concerning effects, cause factors, and root causes and the *DDA* – a dormant deficiency analysis partition of *ECR* – knowledge about the CAD model analysis process and its outcome can be structured and encoded as follows. Relationships between individual effects related to particular symptoms of an individual dormant deficiency type and either a final cause factor or a root cause that has been determined as a result of during analysis, are defined as sets of edges, that is, two-element subsets of $V(KN)$ that have their ends in *RC*, and one of the refining partitions of *EC* and *CC* respectively. Those pairs of effects and cause factors/root causes, in turn, can be further linked among each other to obtain sets of sequences that organize and encode entire effect/cause chains. For example, in the case of a type I dormant deficiency, results of a CAD model analysis where a root cause was determined to be linked to an effect that is based on a detected symptom, are encoded in general by an EC^{I}-RC path denoted by P^{I}. In particular, a path that links an effect represented by x_0 to a root cause represented by x_n is encoded by an $\{x_0\}$-$\{x_n\}$ path such that $V(P^{I}) \cap EC^{I} = \{x_0\}$ and $V(P^{I}) \cap RC = \{x_n\}$. Similarly, analysis results for cases where a cause factor is linked to an effect are encoded generally by an EC^{I}-CC^{B} path. However, in those cases the particular path encoding of a $\{x_0\}$-$\{x_n\}$ path needs to be structured such that $V(P^{I}) \cap EC^{I} = \{x_0\}$ and $V(P^{I}) \cap CC^{B} = \{x_n\}$. Note that taking into account paths from effects to cause factors is required, because during cause and effect analysis of CAD models in practice one may encounter cases – especially in regard to type II and type III dormant deficiencies – where determination of a specific root cause is not possible with the information available from both the CAD model and the design requirements to which the CAD model is subjected. The *j*-th path of a set of paths that encode and document analysis results of type I dormant deficiencies that were detected in the *k*-th CAD model is denoted by $P^{I,j,k}$. Similarly, paths that encode analysis results in regard to type II and type III dormant deficiencies are structured in a manner like their counterparts for the type I, but taking into account the sets associated with EC^{II}, EC^{III}, etc. Those paths are denoted in a similar manner to their type I dormant deficiency related counterparts, that is, for example, P^{II}, $P^{II,j,k}$, P^{III}, and so forth.

3.2.4. Knowledge encoding and visualization

A human-centric graphical manner of presentation is a necessary precondition to make use of and benefit from, and thus enable as well as support, the powerful human visual and cognitive capabilities during the computer-aided processes of analyzing, making sense of, and conveying information and knowledge that is externalized and usually encoded in a digital, non-visual form. However, visually representing knowledge and information encoded by mathematical structures and abstract data objects such as graphs and networks, which is commonly referred to as *graph drawing* and *node-link diagram visualization* (cf. Huang et al. (2009), Purchase (2002)), represents a challenging task. This is, among other issues, due to the fact that just providing any kind of information visualization does not in general translate automatically into effective and efficient communication and visual analysis. One major reason for that is related to the ambiguous and non-unique mapping between a graph as an abstract mathematical structure and its graphical representation in the form of a node-link diagram implementing a designed layout. Here, earlier and recent theoretical and empirical work on the quality of node-link diagram visualization, which, as a multi-dimensional construct, is related to various factors, provides some valuable pointers. First and foremost, issues of perceptual and cognitive factors and costs need to be considered. Here factors that relate to the number of connecting nodes, path crossings, and bendiness of paths, as reported, for example, in Ware et al. (2002), Zheng et al. (2005), and Papadopoulos and Voglis (2013), are important measures to reduce what is known as *visual clutter*, and this in turn helps to ease visual perception and cognitive efforts. Results of empirical work related to the aesthetics of such node-link diagram visualizations, as detailed in Purchase et al. (2002), were recently reported, for example, in Baum (2020) and De Luca et al. (2018). They are also partly interrelated with those factors just outlined, and further take into account criteria referring to visual mapping and the composition of visualization elements such as color mapping, alignment, symmetry, and visual density. Work is reported in Misue (2006), and Misue and Zhou (2011) on efforts to generate node-link diagram layouts that are easy to perceive and comprehend visually by using anchored maps instead of simple connecting edges between node sets, which are usually represented as overlapping lines. Another important aspect of the quality of node-link diagram visualization is the specific purpose of the visualization and the potential for supporting the provision of insight in regard to this context (see also discussions in Huang (2014), and Saraiya et al. (2006)).

For the purpose of the visualization – that is, the organization and graphical representation of knowledge related to the analysis of the cause and effect relationships of dormant deficiencies during CAD model analysis – a particular diagram with a stable visual pattern and a layout that is readable, useful, and easily understandable has been chosen. This diagram is referred to in the literature as an *Ishikawa diagram* or a *fishbone diagram*, but sometimes it is called a *herringbone diagram* or a *Fishikawa* (cf. Ishikawa (1991), Tague (2005), Bradley (2017)). The various names for this diagram are all based on the fact that the visual appearance of the central layout shape is akin to the side view of a fish skeleton. The Ishikawa diagram and its layout are specifically structured to display visually the possible causes of a specific problem, which can then be traced back to the individual factors and root causes that prompted that effect. Within the diagram layout – the spatially organized two-dimensional fishbone structure – the causes are usually grouped into main categories, which in turn are refined by adding cause factors. Then the latter can be further refined to identify potential root causes.

In order to achieve a high level of familiarity for the end user of the diagram visualization as implemented, the effects, cause categories, cause factors, and root causes contain only technical terms and descriptions that are specific to and meaningful within the domain knowledge and context of parametric feature-based CAD modeling. Note that within the work reported in this paper, all cause and effect relationships and root causes that were identified, structured, and graphically represented using the Ishikawa diagrams as shown and discussed, have been double-checked and confirmed through empirical analysis of all the CAD models used during development, implementation, and verification of the framework, the software tool, and the overall integrated visualization and simulation environment. The mapping of knowledge and information between the encoded digitized form and the visualization is realized through correspondence relationships. Those relationships are based on characteristics shared between the elements of graphs of the knowledge network and their counterparts in the Ishikawa diagram and can be described as follows.

The correspondence relationships between the various classes of partitions and refinements used in the r-partite graphs of the knowledge network and the type of elements employed in the Ishikawa diagrams are structured in a straightforward manner. The EC^I, CC^I, …, EC^{III}, CC^{III}, RC, and their refining partitions – as already described in detail elsewhere in this subsection – correspond to the effects, cause categories, and root causes represented in the diagrams.

For example, in the case of type I dormant deficiencies, the number of refining sets that form the part of r partitions in regard to CC^I within $\dot{G}^I$ corresponds to the number of cause categories in an Ishikawa diagram. Here the number of vertices of each refining partition of CC^I within $\dot{G}^I$ corresponds to the number of cause factors within each respective cause category in an Ishikawa diagram. The number of vertices that form the part of r partitions in regard to RC within $\dot{G}^I$ corresponds to the number of root causes within each respective cause factor in an Ishikawa diagram. Note that in the case of EC^I – due to its n-subsets $[\{EC^{I,1}, ..., EC^{I,m}\}]^n$ with $n = 1$ for those refining partitions – we have an explicit correspondence between the number of the set elements of the refining partitions of EC^I within each $\dot{G}^I$ and the effect in the Ishikawa diagrams. In the case of type II and type III dormant deficiencies, the correspondence relationships in regard to the number of elements between graphs and Ishikawa diagrams are formed in a similar manner by taking into account respective partitions and their refinements.

Correspondence between cause and effect relationships that are encoded within attributed r-partite graphs and their counterparts in the fishbone structure of the Ishikawa diagrams is based on edge sets, which – in the case of type I dormant deficiencies – are defined as $E(\dot{G}^I) = \{(v_i,v_j) \mid \forall i = 1 \ldots m, \forall j = 1 \ldots n\ (v_i \in EC^I) \wedge (v_j \in CC^I)\}$. Note that each v_i is the endpoint that makes those edges adjacent in regard to the effect linked to EC^I. As outlined above for other characteristics of elements, in the case of type II and type III dormant deficiencies, the correspondence between cause and effect relationships in attributed r-partitioned graphs and their counterparts in the fishbone structure of the Ishikawa diagrams is formed in a similar manner by taking into account respective partitions and their refinements.

Color correspondence between elements of the knowledge network graphs and the Ishikawa diagrams is realized through vertex attributes and the coloring of diagram element details. The latter is related to the description of an effect and its related cause categories, their individual factors, and root causes, and is visually represented as boxed text with a frame and a white background. Those diagram element details and the color in which they are visually presented correspond to the image of the vertex attributes as computed by the map *va*, which was described earlier. Currently, a dark blue color is used for cause categories and a light blue for cause factors, while dark red and black are used for effects and root causes, respectively. Taking into account color vision impairment, the color scheme has been designed to ensure that the color combinations used in the graphical representations are universally legible. With such a barrier-free design, the graphical representation is clear and accessible to both the color impaired and the viewer with full color vision. The current design is based on a customized color scheme, which employs differences in hue to represent differences between effect, cause category, and root cause. This color scheme design encapsulates a color sub-category, which presents a lightness pair for the blue hue. The legibility of the current customized color scheme design has been verified with a complementary software tool, namely *ColorOracle* (cf. Jenny and Kelso (2007)), an open-source simulator of color-impaired vision.

3.3. Framework and system structures

The approach and concept as introduced and presented earlier have been translated into a framework design and system architecture by considering a simulation-based CAD model analysis and assessment in regard to dormant deficiencies with a real-time oriented graphical user interface and a cross-linked view between the CAD modeling environment and the simulation environment that supports dedicated user interactions such as brushing (cf. Heer and Shneiderman (2012), Munzner (2014), Ware (2020)).

Simulation of the CAD model alteration process and assessment of the CAD model in regard to dormant deficiencies are approached in a straightforward manner by linking the simulation module and assessment algorithms to the CAD system. Various scenarios are then executed by the CAD system regarding test and breakpoint settings that are specified through the user interface and controlled by the software tool. These include CAD model regenerations in regard to parameters that were altered. As both the CAD modeling environment and the simulation-based analysis and assessment environment are cross-linked, the user can see and experience in real-time the symptoms and effects that dormant deficiencies, if present, can have on CAD model regeneration after model alterations have taken place.

Within this framework, dormant deficiencies are not only activated and detected, but also recorded and analyzed. This is done in regard to the critical alteration situation and the dependencies context in which the deficiencies appeared and the number and kind of deficiencies. This structured feedback, together with the user interactions and cross-linked view provided, offers basic information and functionality to assist in locating and analyzing mistakes committed earlier during the modeling process, which introduced the dormant deficiencies. A schematic overview on how this approach is translated into the framework design is given in Figure 2. First, symptoms are analyzed to determine the related effect in regard to a dormant deficiency type. Here information is provided by various system components as shown in Figure 2. Next, the root cause of the effect subject to analysis is determined.

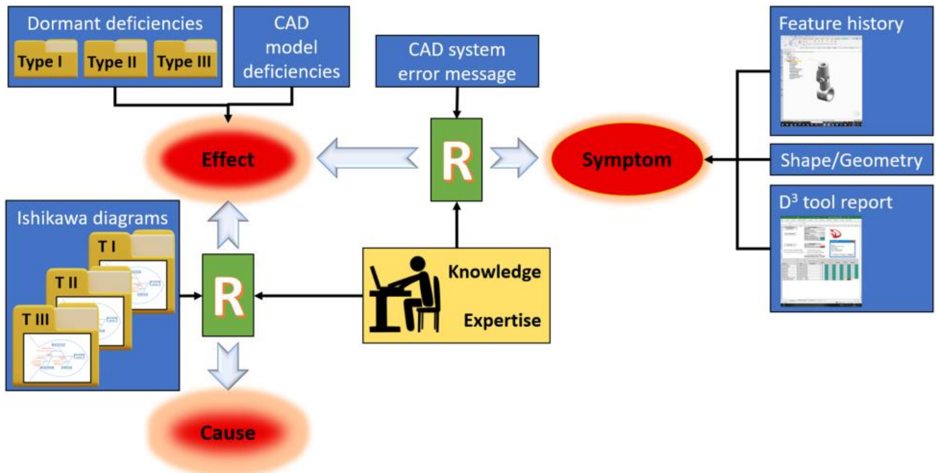


Fig. 2. Schematic overview of framework components and the flow of information across cause and effect relationships as well as effect and symptom relationships.

This step is supported through the knowledge network, which contains cause and effect relationships. Here information from the knowledge network that is encoded as linked attributed *r*-partite graphs is visualized as Ishikawa diagrams to support visual navigation and integration within the graphical user interface of the software tool and the CAD system. Eventually, triangulation of information from the data sources in regard to cause and effect relationships, critical situations, deficiencies, and dependencies across the simulation process and the modeling process is effected, and this is integrated with retrospective analysis. Then possible locations of errors in dependencies can be determined in the form of features and their associated geometry that was rendered invalid during CAD model regeneration. Note that with this simulation-based and process-oriented approach, CAD model assessment requires neither a finished modeling outcome nor a reference solution in the form of an ideal CAD model for comparison. Assessment can be performed with the software tool at any stage during the modeling process, and thus it fully supports formative assessment and feedback.

The software tool design is based on the approach of an alteration simulator operating on the parametric structure of parametric feature-based solid models. The software tool design is conceptually integrated with the current CAD course structure and resources in regard to CAD model analysis and assessment (cf. Mandorli and Otto (2022), Otto and Mandorli (2021)). The user can control the simulation process through an interactive graphical user interface. Once the user starts the simulation and analysis process, the software tool automatically collects and changes, one-by-one, all the dimensions that are related to sketches, profiles of features, and extensions of finite extrusions. Note that, due to the cross-linked structure between the software tool and the CAD system, each dimensional change is immediately reflected in the shape of the CAD model being regenerated. The user can define the proportions of the CAD model alterations that are used during the simulation process. The current system default is 10%. After each change in a dimension value, the number and type of features that have entered a failed or warning status are recorded within the report-based feedback section of the user interface. The user can also set breakpoints in order to pause the simulation and analysis process. Breakpoints can be set either after each change of a single dimension value or after a dimension has been both increased and decreased. Following the activation of breakpoints, the simulation and analysis process can be terminated, or it can be resumed with either the current breakpoint settings or an uninterrupted execution until the end of the process is reached. The graphical user interface has been developed and implemented within the Microsoft *Excel* Visual Basic for Applications (VBA) framework and is provided through the dashboard of the software tool. This interactive graphical user interface is a part of the cross-link structure and brushing supporting the visual simulation display. The visual simulation display, and in particular the user interface within the dashboard, have been designed not only to mediate communication and user interaction through computer screens, but also to inform and to provide feedback in an effective and efficient manner.

The design is guided by functional and data requirements and is grounded in principles of visual usability and the application of various insights and groundwork of cognitive and perceptual psychology. In particular, the principles of familiarity and consistency strongly affected basic decisions regarding the interface design, and subsequently the system development, as outlined next. Reasons for that were related to various factors. External consistency, and in particular familiarity, allows users to transfer expectations and experiences they have acquired from systems and interfaces that were similar, thus facilitating an almost immediate interaction without first having to overcome a tedious learning curve. The underlying rationale for this is the tendency for users to draw on their cumulative experience from other digital systems and experiences that support them in their understanding of how things work and what to expect when encountering something new (see also Jacob's Law and consistency in visual design and usability, for example in Schlatter and Levinson (2013), and Yablonski (2020)). The visual hierarchy of the visual simulation display has been structured based on a combination of position and contrast in size to relate the relative importance of elements to their visual prominence based on contrast (cf. Ware (2008)). For the cross-linked view pertaining to the CAD modeling environment, placement and proximity to frame (cf. discussions on Gestalt psychology principles related to visual interface design in Johnson (2020), and Ware (2020)) have been used to affect contrast and perception of hierarchy regarding the rendered CAD model shape and its subsequent model re-creations during the simulation process. For the interface of the dashboard, placement and proximity of elements and eye behavior have been used to improve perception of hierarchy (see also discussions on F-shaped reading patterns and the inverted pyramid principle in Lidwell et al. (2010), and Moyers (2017)). The design also includes a visual resting space, that is white space for the eye to rest on in a screen (cf. Schlatter and Levinson (2013)). Note that this visual resting space can also be used temporarily during assessment to literally park floating dialog windows, thus reducing visual clutter.

Within the visual interface design presented, color has been used strategically to highlight and draw attention. This policy is consistent throughout the CAD model assessment feedback in the data tables such as the initial conditions and the test results, where color is used to indicate and draw attention to dormant deficiencies detected and to features that were identified as not fully constrained, or otherwise found to be defective. Color is also used during brushing within the cross-linked view to highlight a dimension within the CAD modeling environment after its counterpart within the list-based data set in the dashboard has been selected. The color scheme used for the visual interface design has been customized so that it can be accessible to both the color impaired and the viewer with full color vision (cf. Harrower and Brewer (2003), Munzner (2014)). Finally, some consideration has been given to the overall appeal of the visual interface design as it affects perception of use due to the aesthetic-usability effect (cf. Kurosu and Kashimura (1995), Tractinsky et al. (2000)). This is because users often perceive interfaces with aesthetically pleasing designs to be more practical. Hence, an aesthetically and visually pleasing design can actually influence usability, extend credibility, and get users to be more tolerant of minor usability issues (see again Yablonski (2020)). In particular, the interface has been kept as a minimal design to allow students to focus on the task at hand and the feedback generated in real-time during simulation-based CAD model assessment. For example, this includes switching off the grid in the background and reducing the ribbon (menus and toolbars) to a minimum as system default. In addition, the entire wording within the dashboard has been designed to appeal to novice users by avoiding expert language and difficult domain-related terms and symbols. Instead of referring to 'brushing' or 'breakpoint scenario setting', phrases in plain English are employed in the interface, such as 'show dimension' and 'pause when value changes'. These additional design efforts are aimed at contributing to the overall appeal of the visual simulation display design by providing a personality for the interface that fits best with the characteristics of the persona of novices and student users and their expectations about what the software tool does and for whom it is meant.

The newly developed software tool features a technical architecture that leverages API-based functionality provided by commercially available CAD systems to support a modular and highly cohesive system architecture. At present, the simulator module kernel and knowledge related to the CAD model deficiency analysis are implemented within the CAD modeling environment as procedures and functions based on the VBA framework. Within the current implementation, the modeling environment deploys a commercially available parametric feature-based solid modeling system, namely *SolidEdge* from Siemens AG. Further details on the visual simulation display design and the software tool implementation can be found in Otto and Mandorli (2021).

4. Empirical results from practice

4.1. Overview

In order to test and evaluate both the approach and the prototype implementation, functionality, performance, and reliability have been assessed using 230 parametric feature-based CAD models created by students for CAD laboratory and course assignments in the previous and current academic year. To keep the presentation and discussions concise and transparent, the sample CAD models used in this paper are limited to a selection from a single exercise. This exercise is usually administered after the first quarter of the course and requires the creation of a CAD model with a non-complex shape (see Figure 3) that can be created by novices using a basic set of fewer than fifteen feature commands.

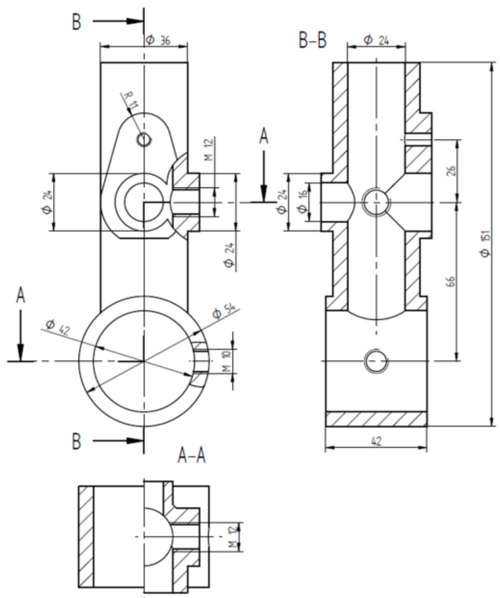


Fig. 3. Outline and overall dimensions of the sample CAD model from an actual CAD course assignment.

The goal of this exercise is to train students toward some of the expected learning outcomes of the course. These include an understanding of the importance of a well-designed modeling strategy with a focus on CAD model quality rather than on model shape, and an awareness of, as well as adherence to, various basic elements of CAD modeling guidelines and best practice. For this exercise those elements include the use of non-complex profiles whenever possible, defining only fully constrained profiles, and being aware of feature dependencies and critical modeling situations that can lead to dormant deficiencies in a CAD model. In the following sub-sections, examples are presented and discussed of actual intra-feature and inter-feature dormant deficiencies as encountered during analysis of student-created CAD models. Additionally, one complete case study is provided in order to exemplify the process of analysis, progressing from the identification of a symptom to discovery of the root cause for each type of dormant deficiency. This demonstrates a practical application of the approach grounded in the novel concept and its associated software tool. Note that during profile creation the definition of dimensions is strongly related to the definition of constraints, and that both contribute to the definition of fully constrained profiles. Dormant deficiencies are often introduced during profile creation due to a combination of wrong decisions related to the selection of constraint and dimension types, and the selection of entities that are inadequate for correctly setting those constraints and dimensions. Therefore, in the examples presented below, issues related to the definition of either dimensions or constraints treated within one sub-section may cross over into another sub-section.

4.2. Dormant deficiencies caused by intra-feature dependency errors

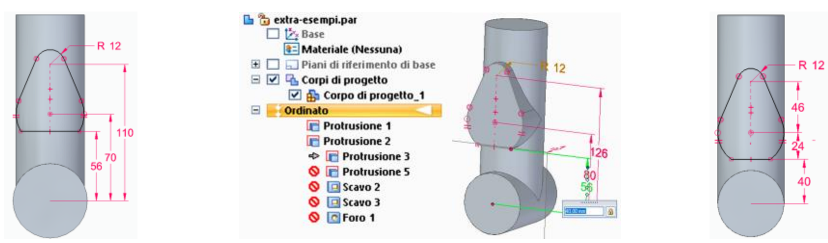
4.2.1. Profile creation and definition of dimensions

In the following, constrained profiles shown in the figures contain various graphic symbols (icons) that were used in the profile definitions to represent the various types of constraints. These are summarized in Table 4. Note that in some cases more than one constraint may be required, and this may result in several graphic symbols being displayed in the same location on the profile.

Description	Symbol
Connect (1 degree of freedom)	✖
Connect (2 degrees of freedom)	⊡
Concentric	◎
Equal	=
Horizontal/Vertical	┼
Tangent	○
Symmetric	] [
Parallel	//
Perpendicular	⏋

Table 4. Graphic symbols representing geometric constraints used in the profile definition.

Errors in the dimensioning used to define the profile of a feature are a common source of dormant deficiencies. For the sample CAD model in the intra-feature case the following situation has been encountered. Efforts to adjust the position of the frontal boss required the profile shown in Figure 4(a) to be vertically relocated by changing dimensions, which resulted in an inconsistent profile configuration as shown in Figure 4(b). This in turn is the result of a type I dormant deficiency that was introduced with the erroneous initial dimensioning (see again Figure 4(a)). After this feature parameter alteration, the re-creation of the CAD model showed deficiencies in the feature history and the shape/geometry (see again Figure 4(b)), which are symptoms typically indicating a type I dormant deficiency. To avoid the introduction of such a dormant deficiency, the dimensioning used to define the profile of the frontal boss should be specified, as shown in Figure 4(c).



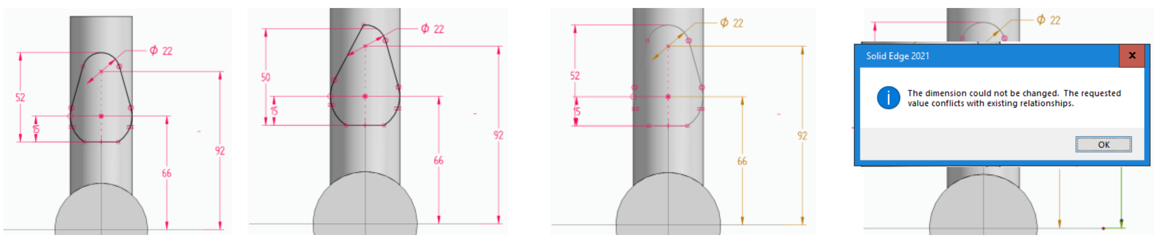
(a) (b) (c)

Fig. 4. Example of a profile definition that contains dimension-related intra-feature dependency errors. From left to right: (a) initial dimensioning used for the profile definition of the frontal boss, (b) altered dimensioning with feature history and re-created CAD model shape exhibiting symptoms of a type I dormant deficiency, (c) correct and alterable dimensioning that can be used for the profile definition of the frontal boss.

Note that the dimensioning as shown in Figure 4(c) is not only alterable without causing any CAD model deficiencies, and thus directly contributing to the creation of a robust CAD model, but also more efficient, as vertical relocation of the profile requires alteration of the value of only one parameter, whereas the erroneous dimensioning shown in Figure 4(a) required a change in the values of three of the parameters used for dimensioning the frontal boss profile.

4.2.2. Profile creation and definition of constraints

Another common source of dormant deficiencies is the definition of constraints. Taking up again the previous example of the frontal boss of the sample CAD model, a fully constrained profile (see Figure 5(a)), consisting of 6 geometric entities (3 arc segments and 3 straight segments), 14 constraints, and 5 dimensions has been encountered during the previous analysis. At this stage of the modeling process, the fully constrained profile does not seem to contain any deficiencies. However, altering the values of some of the parameters that define the profile reveals that there are, in fact, various deficiencies, including an erroneous shape (cf. Figure 5(b)), an unresolved profile (cf. Figure 5(c)), and even a parameter/constraint conflict error (cf. Figure 5(d)). All of these appeared after the re-creation of the altered CAD model. Again, these deficiencies are symptoms which typically indicate dormant deficiencies introduced through mistakes in the constraint definition used to create the profile.



(a) (b) (c) (d)

Fig. 5. Example of a profile definition that contains constraint-related intra-feature dependency errors. From left to right: (a) initial dimensioning used for the profile definition of the frontal boss, (b) alteration resulting in an incorrect profile shape for the frontal boss, (c) alteration resulting in an unresolved profile for the frontal boss, (d) alteration resulting in a parameter/constraint conflict error in the profile of the frontal boss.

In this example, dormant deficiencies were introduced due to a missing constraint between the central upper arc segment and the straight segment on left-hand side of the profile. Instead of such a constraint, an ill-placed dimension has been used. Note that, to add such a constraint to the overall profile definition in a consistent manner, one dimension first needs to be removed, as shown in Figure 6, in order to avoid yet another parameter/constraint conflict error.

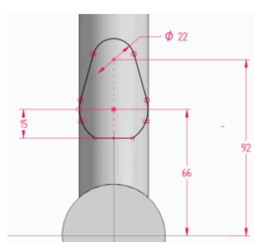
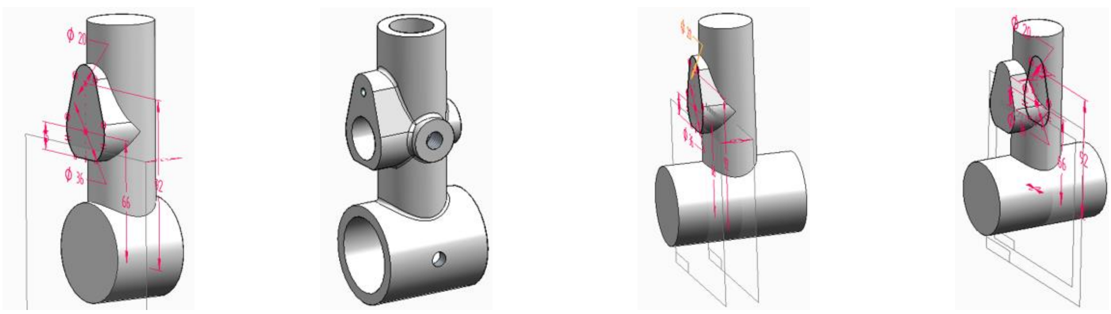


Fig. 6. Example of a robust and consistently alterable profile definition for the frontal boss consisting of a corrected set of basic dimensions and constraints.

4.3 Dormant deficiencies caused by inter-feature dependency errors

4.3.1. Reference plane selection

Within the context of inter-feature dependency errors, various cases were found where dormant deficiencies were introduced during reference plane selection, although this is a process that is usually considered to be neither complex nor difficult in nature. In the case of the sample CAD model discussed in this paper, for example, the frontal boss plane was aligned to the planar front (entrance) face of the base cylinder in a co-planar manner, as shown in Figure 7(a). Altering the length of the base cylinder then results in a change in the frontal boss height. This is readily discernable in the shape and geometry of the re-created CAD model shown in Figure 7(b). This represents a symptom typical of a type III dormant deficiency, as the volume of the frontal boss becomes either excessive or minute, in proportion to the alteration of the base cylinder length. Note that other symptoms typical of type I and type II dormant deficiencies are notably absent, for example features being labeled with warning or failed status, or incomplete shapes or incomplete geometry. Of course, if type I and/or type II dormant deficiencies were also present, those symptoms would be triggered and thus would also be readily discernable after the CAD model regeneration had been executed. In this context, to avoid dependency errors resulting in the introduction of dormant deficiencies, the reference plane selected for the frontal boss plane should lie parallel to the XZ plane of the CAD system's modeling space, taking into account that there should be a certain distance between those planes, as shown in Figure 7(c). Then, by employing a "to-the-next" extrusion option in the direction toward the vertical cylinder, the height and thus the volume of the frontal boss can be correctly defined in a manner independent of the base cylinder length.



(a) (b) (c) (d)

Fig. 7. Example of reference plane selection containing related inter-feature dependency errors. From left to right: (a) initial reference plane selection used for the frontal boss plane, (b) re-created CAD model shape exhibiting symptoms of a type III dormant deficiency after the base cylinder length has been changed, (c) correct reference plane selection for the frontal boss plane, (d) alternative correct reference plane selection for the frontal boss plane.

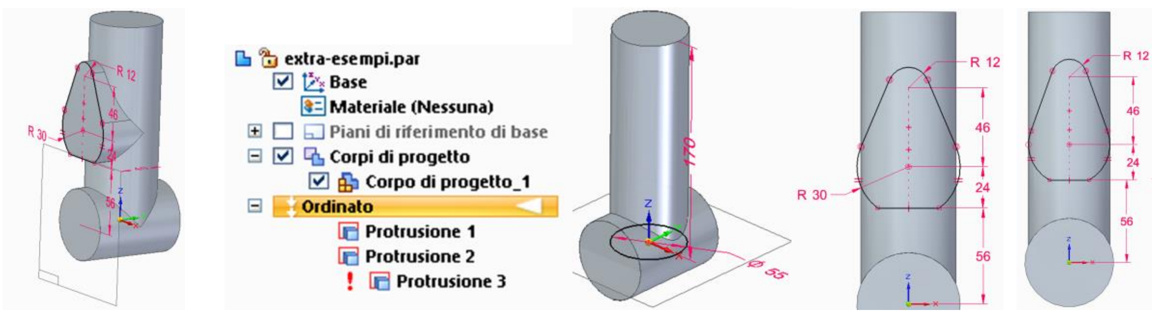
Note that directly selecting the XZ plane of the vertical cylinder and extruding the profile of the frontal boss in the outer direction, as shown in Figure 7(d), also produces a correct solution, avoiding the introduction of the type III dormant deficiency outlined above. However, taking into account this particular modeling situation and the operational limits of this kind of extrusion option, other types of dormant deficiencies may be introduced. This is because the profile of the frontal boss might become too large to be correctly projected onto the outer surface of the vertical cylinder. This, in turn, after CAD model re-creation, may result in a frontal boss with an incomplete or incoherent shape. For additional details related to this issue, see also Figure 8(c) and its discussion in the next sub-section.

4.3.2. Profile creation and definition of dimensions

As outlined elsewhere in the paper, cases where a dimension is wrongly used instead of a constraint can be treated as dependency errors in a profile definition pertaining to either a missing constraint or an error related to the definition of a dimension that should have been defined as a constraint. In the inter-feature example presented in this sub-section, the latter is the case.

Note that, within this context, adding a missing constraint without first removing the dimension parameter that was wrongly used will result in an error due to an over-constrained profile. However, removing the dimension parameter that was wrongly used while omitting to add the missing constraint will result in yet another deficiency due to the profile becoming under-constrained.

Taking up again the creation of the frontal boss of the sample CAD model, modeling is based on an extruded protrusion feature with a profile that is located in a vertical plane parallel to the XZ plane of the CAD system's modeling reference space. To create the volume of the frontal boss, the profile of the extrusion feature is extruded toward the vertical cylinder using a “to-the-next” extrusion option. The profile of the frontal boss has a radius dimension (cf. R30 in Figure 8(a)) which has assigned to it a numerical value that allows for the frontal boss to be aligned to the vertical cylinder in a tangential manner. Again, as in the previous cases discussed, at this stage of the modeling process the fully constrained profile does not seem to contain any deficiency. However, a decrease in the radius, and therefore diameter, of the vertical cylinder results in deficiencies during CAD model re-creation. Most prominently, there is an absence of the entire shape and geometry of the frontal boss (cf. Figure 8(b)), which could not be re-created at all by the CAD system. This error in shape and geometry, together with issues present in the feature history, are symptoms typical of dormant deficiencies, in this case a type I dormant deficiency. This dormant deficiency has been introduced during the profile definition where a fixed radius dimension value was used instead of creating a dependency between the profile of the frontal boss and a geometric entity of the vertical cylinder through a constraint (see Figure 8(c)). One possible solution is to replace the fixed radius dimension value with a constraint, which enforces a tangency between the silhouette edge of the vertical cylinder and the central top arc segment of the frontal boss profile. For the example discussed, this automatically computes the correct numerical radius dimension value for the profile’s two lateral arc segments (Figure 8(d)), which was previously assigned to a fixed numerical value.



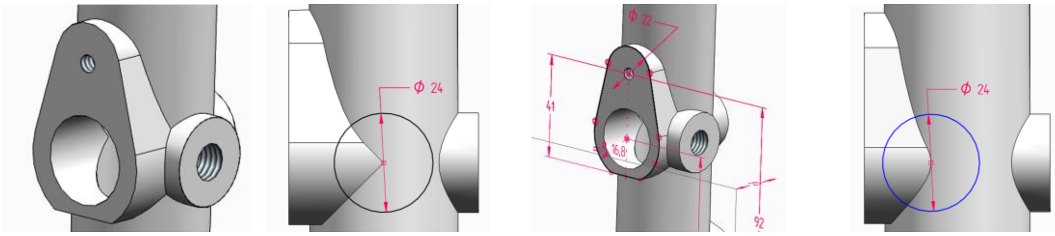
(a) (b) (c) (d)

Fig. 8. Example of a profile definition that contains dimension-related inter-feature dependency errors. From left to right: (a) initial dimensioning used for the profile definition of the frontal boss, (b) altered dimensioning with feature history and re-created CAD model shape exhibiting symptoms of a type I dormant deficiency, (c) dimensioning details of the profile definition of the frontal boss after alteration, (d) correct alterable dimensioning that can be used for the profile definition of the frontal boss. One fixed dimension has been replaced with an inter-feature constraint.

4.3.3. Profile creation and definition of constraints

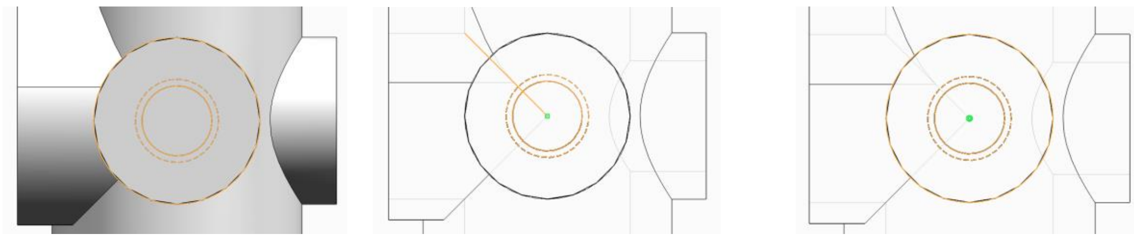
In what follows an example is presented that relates to errors committed during profile creation due to a mistake in the constraint definition. That mistake introduced a dormant deficiency, so that when the CAD model was altered and re-created, there was an error that had an adverse effect on the initial geometric entity selection, resulting in the wrong spatial positioning of features in regard to initial design requirements.

Taking again the sample CAD model, one design requirement is that the lateral boss and its threaded hole remain positioned in respect to the frontal boss as shown in Figure 9(a) and Figure 10(a). This was achieved by using a constraint that relates the center of the circular profile of the lateral boss to the endpoint of the edge of the contour of the frontal boss in a coincident manner as shown in Figure 9(b) and Figure 10(b). However, if the profile of the frontal boss is altered (unintentionally) in a way that it is not exactly tangential in respect to the vertical cylinder (see Figure 9(c)), the intersection edges between the frontal boss and the vertical cylinder will change (see Figure 9(d)) in a way that, in most cases, results in a loss of the edge endpoint used in the initial constraint definition. This, in turn, has an adverse impact on the constraint that regulates the proper positioning of the lateral boss in respect to the frontal boss, with the former being wrongly positioned because the center of its circular profile is now coincident with points along the contour of the frontal boss instead of a particular edge endpoint (see again Figure 9(b) and Figure (10)).



(a) (b) (c) (d)

Fig. 9. Example of a profile definition that contains constraint-related intra-feature dependency errors. From left to right: (a) correctly positioned lateral boss and lateral boss hole as specified by the design requirement, (b) enlarged view showing details of the initial constraint condition, (c) re-created CAD model shape exhibiting symptoms of a type III dormant deficiency after parameter alteration of the frontal boss, (d) enlarged view showing details of the error in the constraint condition after parameter alteration of the frontal boss.



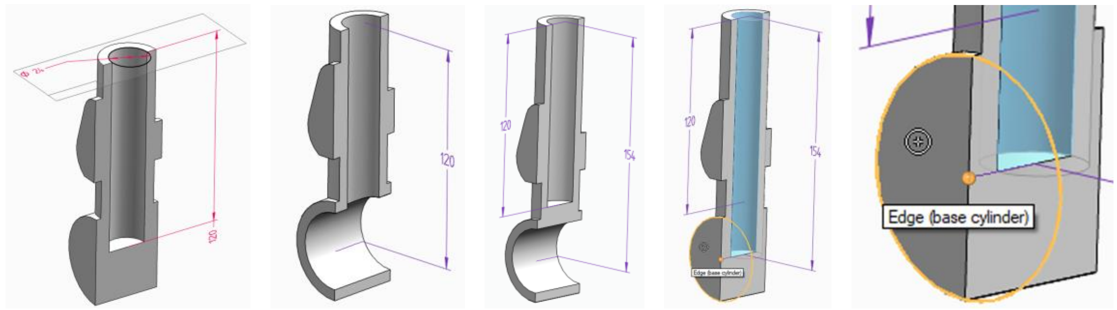
(a) (b) (c)

Fig. 10. Enlarged section of the example of a profile definition that contains constraint- and entity-related inter-feature dependency errors. From left to right: (a) correctly positioned lateral boss and lateral boss hole as specified by the design requirement, (b) mistake committed during geometric entity selection, (c) correctly selected circular profiles of the lateral boss and the lateral boss hole used in a constraint that spatially aligns them in a concentric manner.

A readily discernable symptom typical of a type III dormant deficiency appears when the lateral boss and its threaded hole are positioned in regard to the frontal boss in a manner that is not consistent with the intended alignment as previously described. To avoid the introduction of dormant deficiencies leading to this kind of critical situation during modeling, a different type of constraint is recommended. More details on this issue of constraint type selection are discussed elsewhere in this paper, in particular in the context of coincidence constraints between an edge endpoint and a profile center.

4.3.4. Definition of extrusion options

Besides dimensions and constraints, extrusion options are an important element for the creation of feature geometry and shape. Although neither their command structure nor their application is complex compared to other modeling operators, their proper selection and application should not be underestimated, as they are also a potential source for introducing dormant deficiencies. Within the sample CAD model, the case that follows can be considered a typical example of underestimating the impact of an extrusion option. In this case the option chosen was inappropriate for the modeling context in which it was applied, which subsequently resulted in the introduction of dormant deficiencies. To create the central vertical cutout, an extruded circular cutout feature of a finite depth (see Figure 11(a)) was used with a value equal to that of the vertical cylinder feature that was created through a vertically extruded circular protrusion, as shown in Figure 11(a). This modeling scenario results in a central vertical circular cutout that passes through once the circular horizontal cutout is created (cf. Figure 11(b)). However, if the height parameter value of the vertical cylinder feature is increased, this vertical circular cutout will no longer pass through, due to its finite length with a value that has not been adjusted in regard to the CAD model alteration. Re-creation of the CAD model during this modeling stage, as shown in Figure 11(c), depicts this unintended result together with symptoms typical of dormant deficiencies. In this case a type II dormant deficiency is indicated, as the shape and geometry of the central vertical cutout (feature) are inconsistent with those of the passing-through cutout. Note that the symptoms of this type of dormant deficiency are usually not recognized by the CAD system, as was the case here, thus no feature warnings, or errors, or any other CAD system-generated feedback, were available to indicate this deficiency.



(a) (b) (c) (d) (e)

Fig. 11. Example of an extrusion option definition that contains inter-feature dependency errors. From left to right: (a) initial dimension and definition of extrusion option used for the central vertical cutout, (b) cross-section of the correctly created CAD model up to the modeling stage where the vertical and horizontal cutouts were created, (c) cross-section of the re-created CAD model shape exhibiting symptoms of a type II dormant deficiency, (d) cross-section of the correctly re-created CAD model using an alternative extrusion option that is based on a snap point, (e) enlarged cross-section of the topological/geometric condition related to the alternative extrusion option that is based on a snap point.

However, during tool-assisted analysis, the DDD software tool detected and reported on this dormant deficiency, because the geometry of the central vertical cutout in regard to the number of faces and edges had changed during the CAD model re-creation after parameter alteration. To avoid this mistake and the consequent introduction of a dormant deficiency, a different type of extrusion option needs to be applied, which is based on a snap point instead of a fixed numerical dimension value. In the case presented, for example, the center of the base cylinder could be used as a snap point, as shown in Figure 11(d) and Figure 11(e). Correctly determining and defining the conditions of extrusion options is not always a trivial and straightforward process. Completely understanding the possible impact and the operational and functional consequences of extrusion options in the context of particular modeling situations can sometimes be quite a challenging task. Therefore, having formative feedback at various stages of the modeling process can increase awareness of and knowledge about the possible presence of dormant deficiencies and their causes. This will contribute to increasing the alterability, and thus the robustness, of feature-based CAD models. The next section offers detailed discussions and examples on how to employ the novel framework and approach in a practical manner. This can be achieved by engaging in a software tool assisted analysis in a systematic way to determine the causes of dormant deficiencies based on the symptoms and effects presented.

4.4. From symptom and effect to root cause of dormant deficiency

4.4.1. Analysis of type I dormant deficiencies

Recall that, in the case of type I dormant deficiencies, symptoms appear in the form of a change represented by labels indicating a warning or failure in the status of features. This is due to errors and deficiencies encountered during feature regeneration after a CAD model has been altered. These symptoms are reported by the CAD system in relation to individual effects encountered during CAD model regeneration. For the example discussed in this sub-section, those symptoms are shown in Figure 12, which depicts the entire DDD software tool interface, and in Figure 13, which shows an enlarged section of the CAD system view. Further analysis of these symptoms (see Figure 14(a)) leads to the effect, which is a failure in the regeneration of the rear boss hole feature (see again Figure 13(b)).

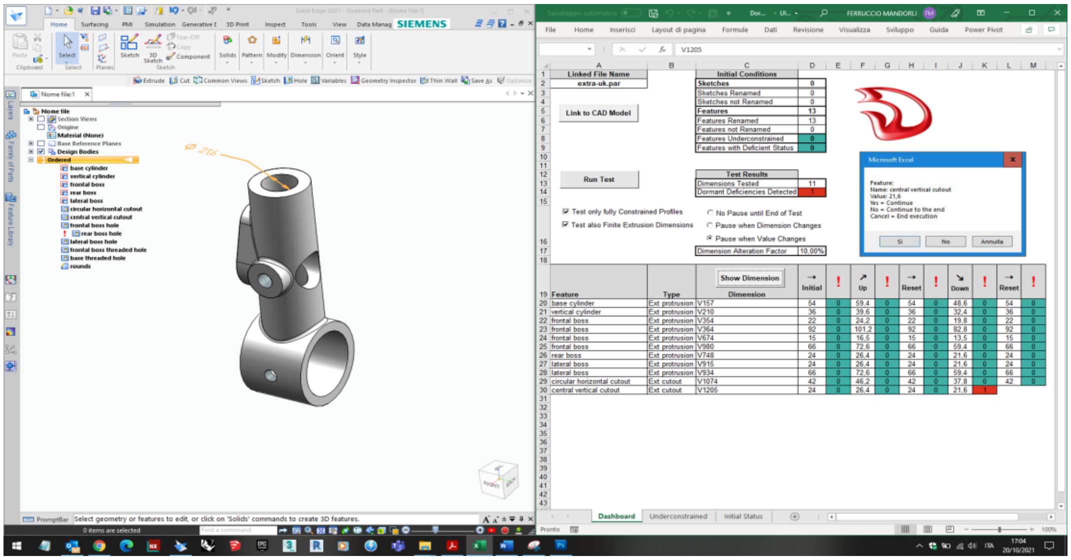
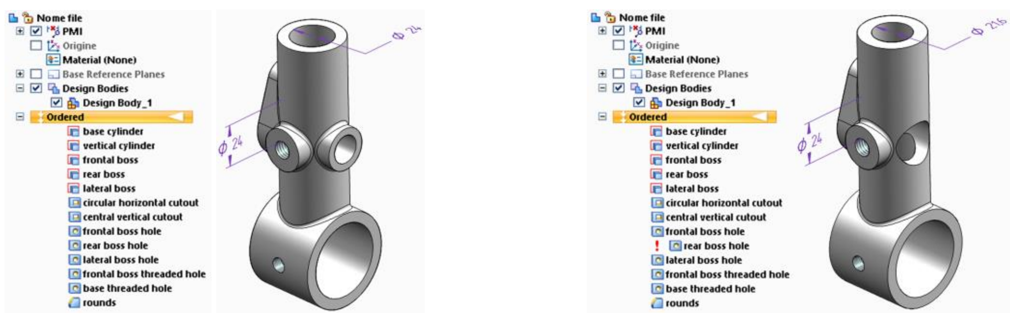
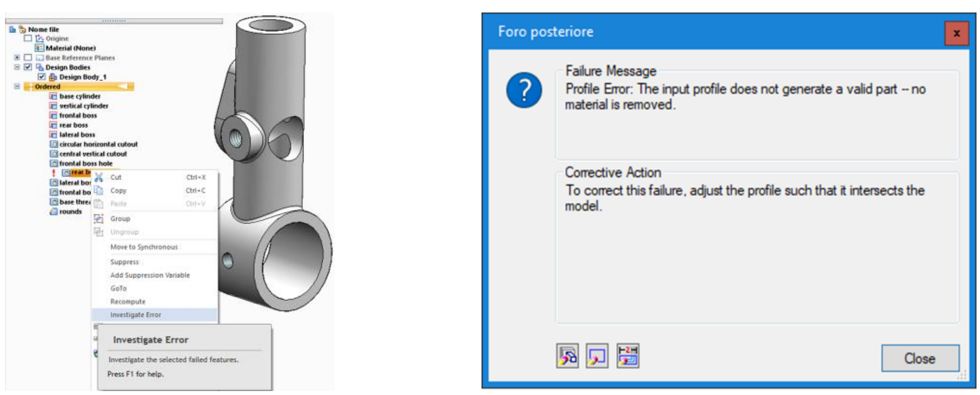


Fig. 12. Example of symptoms relating to the type I dormant deficiency that are readily discernable in the feature history and in the shape and geometry, and in the DDD software tool report as shown through the visual simulation display enabling a cross-linked view in real time during simulation and analysis.



(a) (b)

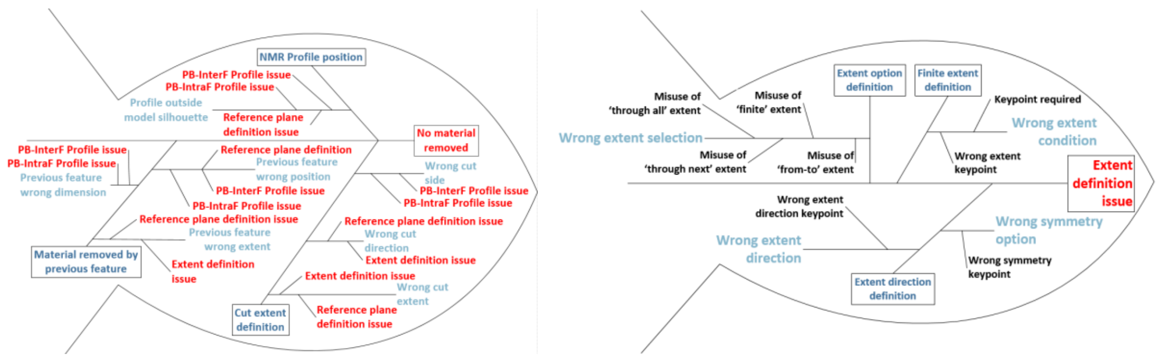
Fig. 13. Actual part where the type I dormant deficiency was discovered. From left to right: (a) CAD model and feature history with latent type I dormant deficiency, (b) CAD model and feature history with triggered type I dormant deficiency.



(a) (b)

Fig. 14. Analysis of type I dormant deficiency symptom and effect relationship. From left to right: (a) CAD model with triggered type I dormant deficiency, feature history, and symptoms of interest, (b) CAD system feature error message relating to the effect termed “No material removed”.

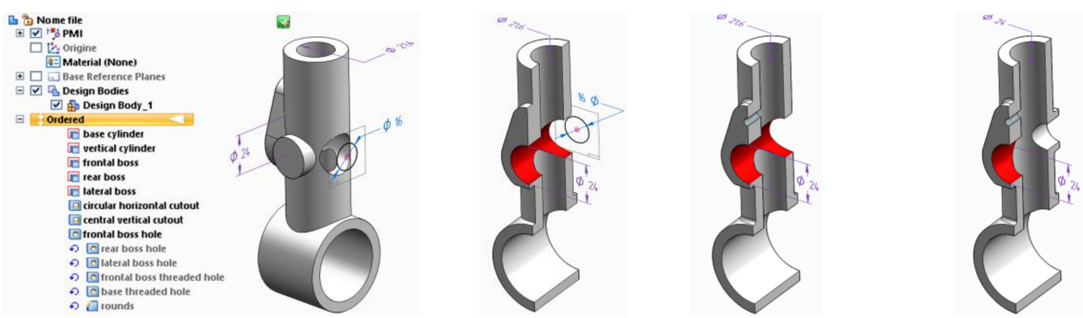
The problem encountered by the system is related – as an effect – to the error message "Profile Error: … - no material removed" (see Figure 14(b)), which is also an encoded type I dormant deficiency effect within the framework and current implementation. The graphical representation of the cause and effect relationship for this effect, as shown in Figure 15(a), suggests that there are three possible cause categories. To determine which one is the most likely, the failed feature needs to be analyzed again. It then becomes evident that the feature profile (see Figure 16(a) and Figure 16(b)) is located in a region where there is no material that can be removed, because it has already been removed by a previous feature.



(a) (b)

Fig. 15. Graphical representation of cause and effect relationships. From left to right: (a) Ishikawa diagram for type I dormant deficiency related effect termed "No material removed", (b) Ishikawa diagram for type I dormant deficiency related (basic) effect termed "Extent definition issue".

This situation is a consequence of the re-created CAD model shape shown in Figure 16(c), which is due to an alteration to the central vertical cutout dimension value as shown in Figure 16(c) and Figure 16(d). This provides sufficient evidence to pursue the effect-cause analysis under the cause category "Material removed by previous feature" (see again Figure 15(a)) within this diagram. The feature history tree and the CAD model shape reveal that the feature which removed more material than anticipated was the frontal boss hole feature (see again Figure 16(c) and Figure 16(d)).



(a) (b) (c) (d)

Fig. 16. Analysis of type I dormant deficiency related effect termed "No material removed". From left to right: (a) CAD model with triggered type I dormant deficiency, feature history, feature profile, and dimensions of interest, (b) cross-section of CAD model with triggered type I dormant deficiency, feature profile, and altered dimensions of interest, (c) cross-section of CAD model with triggered type I dormant deficiency and dimensions of interest, (d) cross-section of CAD model with latent type I dormant deficiency and dimensions of interest.

The reason behind this unintended feature interaction is the reduction in the diameter of the central vertical cutout feature. This in turn caused the frontal boss hole feature to execute a full through-all cutout instead of an extrude-through-next cutout, because the profile of this hole feature is now larger than its counterpart in the central vertical cutout, which was wrongly defined as a limit of the cutout extrusion. This insight allows us to fix the cause factor as "Previous feature wrong extent" and to determine the base effect,

namely "Extent definition issue" (see again Figure 15(a)), which leads to the diagram shown in Figure 15(b), eventually pointing to the root cause. The base effect determined in the diagram shown in Figure 15(b) relates to the three base cause categories, of which the "Extent option definition" is the most appropriate according to the analysis so far. This in turn leads via the basic cause factor "Wrong extent selection" to the eventual root cause, namely the "Misuse of through-next-extent".
The corresponding path graph (see also Figure 17) over effects, cause factors, and root causes, that is the EC^I-RC path $P^{I,i,j}$ – related to a refinement of EC^I – that encodes and documents analysis results of type I dormant deficiencies that were detected in the *j*-th CAD model during the *i*-th analysis process for the example presented, can be structured as follows. The path $P^{I,i,j} = x_0, \ldots, x_k$ needs to start with an edge $y_0 = \{x_0,x_1\}$ where $y_0 \in E(P^{I,i,j})$ with $x_0 \in EC^{I,r}$ and $x_1 \in CC^{I,m}$. Here the vertex x_0 is incident with y_0, that is y_0 is an edge at x_o, which in turn is determined through $V(P^{I,i,j}) \cap EC^{I,r} = \{x_0\}$ as a first vertex of the path.

➔ **$EC^{I,r}$:** *No material removed*

➔ **$CC^{I,m}$:** *Material removed by previous feature*

➢ *Previous feature wrong extent*

➔ **$EC^{B,n}$:** *Extent definition issue*

➔ **$CC^{B,l}$:** *Extent option definition*

➢ *Wrong extent selection*

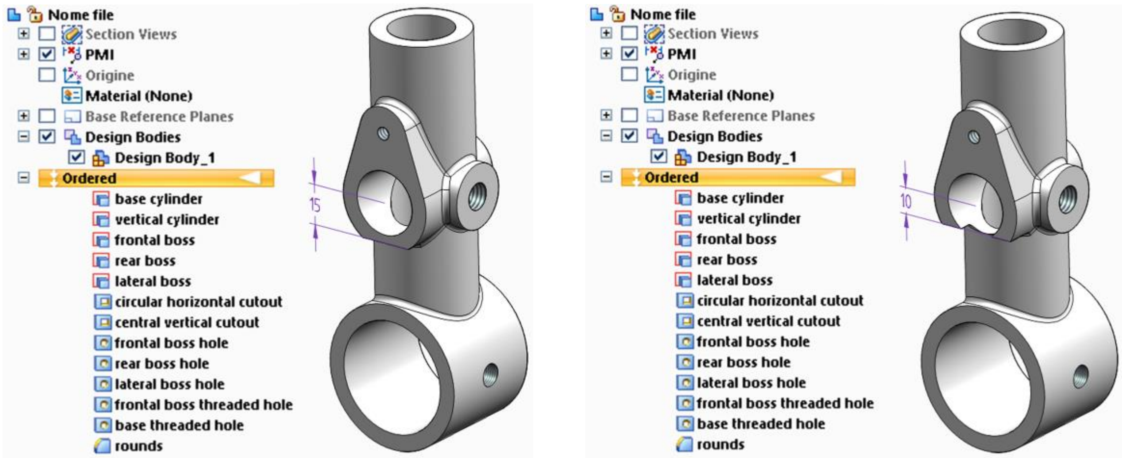
➔ **RC:** *Misuse of through-next-extent*

Fig. 17. Overview of the analysis of the type I dormant deficiency effect-cause path and related refining partitions of categories as used in the knowledge network.

The sub-path $^{\circ}P^{I,i,j}$ of the path graph is recursively defined through edge adjacency over refining effect-cause partitions within sub-paths ending when a basic effect-cause partition set is reached. For the current example, and the one recursion required, this can be described with $1 \leq b < c < k$ and edges $y_b = \{x_b,x_{b+1}\}$ and $y_c = \{x_c,x_{c+1}\}$ where $y_b \neq y_c$ with $y_b \in E(^{\circ}P^{I,i,j})$ and $y_c \in E(^{\circ}P^{I,i,j})$ for $x_b \in CC^{I,m}$ and $x_{b+1} \in EC^{B,n}$ such that $x_c \in EC^{B,n}$ and $x_{c+1} \in CC^{B,l}$. Recall that all effect partitions and their refining partitions – as defined elsewhere in the paper – are one-element partitions. Hence, under the conditions as outlined, the edges y_b and y_c have an end in common. The path $P^{I,i,j}$ needs to end with an edge $y_k = \{x_{k-1},x_k\}$ where $y_k \in E(P^{I,i,j})$ with $x_{k-1} \in CC^{B,l}$ and $x_k \in RC$. The vertex x_k incident with y_k and defined through $V(P^{I,i,j}) \cap RC = \{x_k\}$ then represents a terminal vertex of this path correctly ending in a root cause for the type I dormant deficiency effect subject to analysis.

4.4.2. Analysis of type II dormant deficiencies

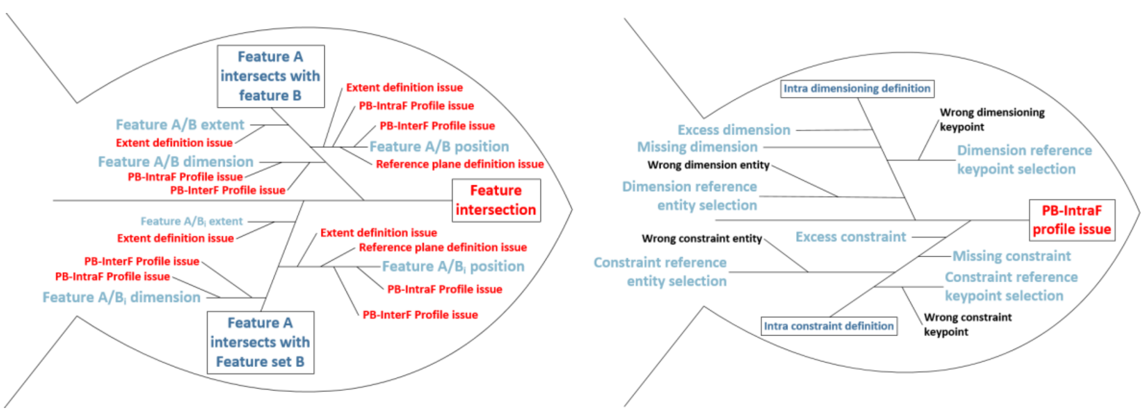
Analysis of type II dormant deficiencies requires more domain knowledge than is usually encoded and thus handled by the CAD system. This is due to the engineering design related modeling goals of the parts and family of components and their final proper geometry and shape, which are created through and represented by parametric feature-based CAD models. Application of the DDD software tool – similar to that described in the previous example but with a different CAD model – revealed a type II dormant deficiency, which could not be detected by the CAD system, because the symptom and effect relationship is beyond the reasoning capability of current commercial CAD systems. As can be seen in Figure 18, the CAD model before (see Figure 18(a)) and after the parameter alteration (see re-created version in Figure 18(b)) does not contain any symptom that relates to an effect indicating a fault in a feature status. However, as can be discerned from the CAD model shown Figure 18(b), there is visual evidence of symptoms in the form not only of incoherent feature shapes, but also of feature shapes that were destroyed during model re-creation without affecting the status of any of the features involved. This indicates a typical type II dormant deficiency effect, and thus problem, which will be looked into next.



(a) (b)

Fig. 18. Actual part where the type II dormant deficiency was discovered. From left to right: (a) CAD model, feature history, and dimensions of interest of latent type II dormant deficiency, (b) CAD model, feature history, and altered dimensions of interest for triggered type II dormant deficiency.

The symptoms outlined above are related to the frontal boss feature and the related hole feature (see again Figure 18(b)). Within the current analysis context, this indicates a type II dormant deficiency effect relating to "Feature intersection". The graphical representation of this effect-cause relationship, as shown in Figure 19(a), suggests that there are two possible cause categories. As there is only an intersection between two individual features, the effect-cause analysis can be pursued under the cause category "Feature A intersects with feature B" (see again Figure 19(a)). As the type II dormant deficiency in this example was triggered by a value change in one of the dimensions of the frontal boss (see again Figure 18(b)), this insight allows us to fix the cause factor as "Feature A/B dimension". Since we know that the problem is related to the profile definition within one feature (see again feature class and range in Table 2), the base effect can be determined as "PB-IntraF Profile issue" (see again Figure 19(a)). This in turn leads to the diagram shown in Figure 19(b).



(a) (b)

Fig. 19. Graphical representation of cause and effect relationships. From left to right: (a) Ishikawa diagram for type II dormant deficiency related effect termed "Feature intersection", (b) Ishikawa diagram for type II dormant deficiency related (basic) effect termed "PB-IntraF profile issue".

The base effect determined in the diagram shown in Figure 19(b) relates to two base cause categories, of which the "Intra constraint definition" is the more appropriate according to the analysis so far. This, in turn, leads to the basic cause factor "Missing constraint", from which, however, a more specific root cause cannot be further ascertained.

For the example discussed above, Figure 20 gives an overview of the analysis of the type II dormant deficiency effect-cause path and the related refining partitions of categories as used in the knowledge network. The corresponding path graph, that is the EC^{II}-CC^{B} path $P^{II,i,j}$ that encodes and documents analysis results of type II dormant deficiencies that were detected in the j-th CAD model during the i-th analysis process for the example presented, can be structured in a manner similar to that presented in the previous example of the type I dormant deficiency.

➔ $EC^{II,r}$: *Feature intersection*

➔ $CC^{II,m}$: *Feature A intersects with feature B*

➢ *Feature A/B dimension*

➔ $EC^{B,n}$: *PB-IntraF profile issue*

➔ $CC^{B,l}$: *Intra constraint definition*

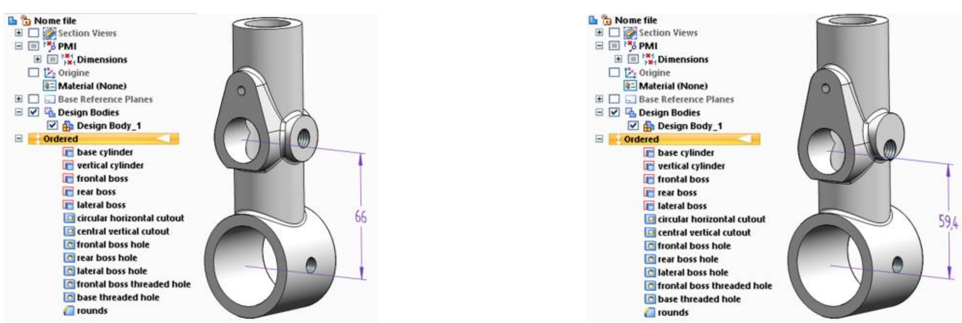
➢ *Missing constraint*

Fig. 20. Overview of the analysis of the type II dormant deficiency effect-cause path and related refining partitions of categories as used in the knowledge network.

Note that the above analysis is based on the assumption that the hole feature is correctly positioned in respect to the frontal boss feature. In other words, it is assumed that the hole feature is centered in respect to the arcs that define the lower half of the boss feature profile. This condition is also consistent with the exercise requirements.

4.4.3. Analysis of type III dormant deficiencies

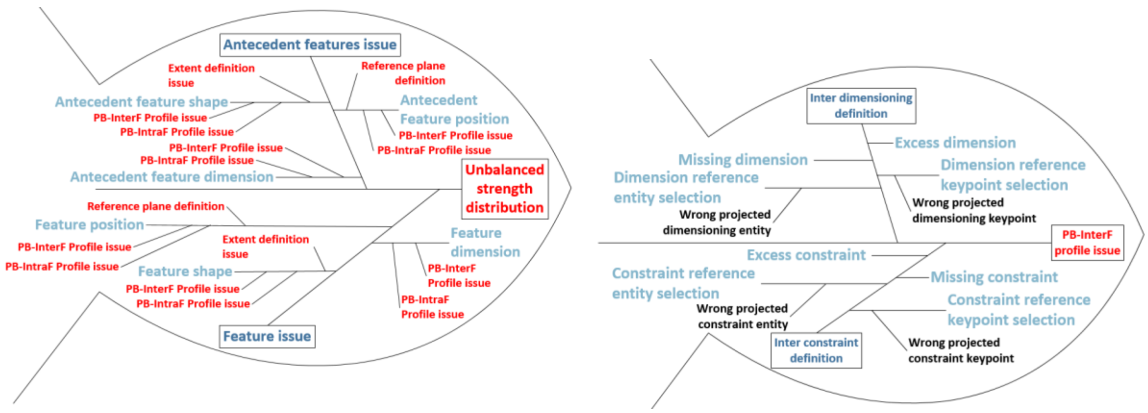
At this point it should be recalled that type III dormant deficiencies are the least specific type of dormant deficiencies. A regenerated CAD model with this type of deficiency does not contain any features labeled with warning or failed status, nor any features with a shape that is incoherent or even destroyed. However, the shape does not meet the design requirements, due to a loss of elementary function. The main symptoms of a type III dormant deficiency which has been activated are a change in geometric properties, such as a loss of symmetry conditions. This was the case for the symptom shown in Figure 21(b), which depicts a regenerated CAD model after the application of the DDD software tool, which revealed the type III dormant deficiency.



(a) (b)

Fig. 21. Actual part where the type III dormant deficiency was discovered. From left to right: (a) CAD model, feature history, and dimensions of interest of latent type III dormant deficiency, (b) CAD model, feature history, and altered dimensions of interest of triggered type III dormant deficiency.

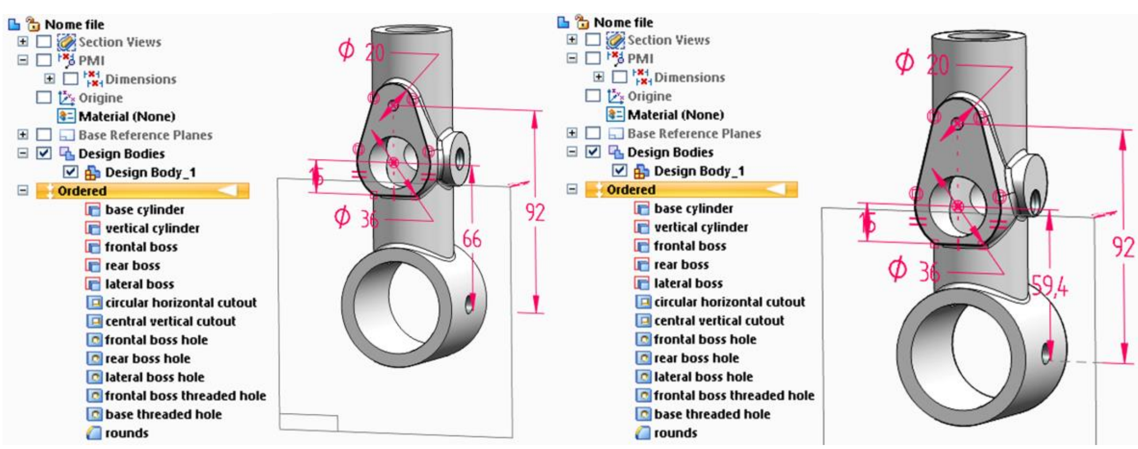
Note that, in some cases, a design specification might require a certain deviation from symmetry conditions. However, those deviations still have to take into account aspects of mechanical engineering and functionality. In the example shown in Figure 21, the threaded hole feature could be intentionally designed to be somewhat off-center in relation to the boss feature, but it cannot be located too close to the boundary of the boss due to issues regarding a possible concentration of tensions under mechanical stress related to the behavior of the material from which the part is to be manufactured. As the CAD model subject to analysis is from a CAD exercise assignment for novices, the loss of symmetry, that is the threaded hole feature being off-center in relation to the boss feature, can be taken to be a type III dormant deficiency symptom that relates to "Unbalanced strength distribution".



(a) (b)

Fig. 22. Graphical representation of cause and effect relationships. From left to right: (a) Ishikawa diagram for type III dormant deficiency related effect termed "Unbalanced strength distribution", (b) Ishikawa diagram for type III dormant deficiency related (basic) effect termed "PB-InterF profile issue".

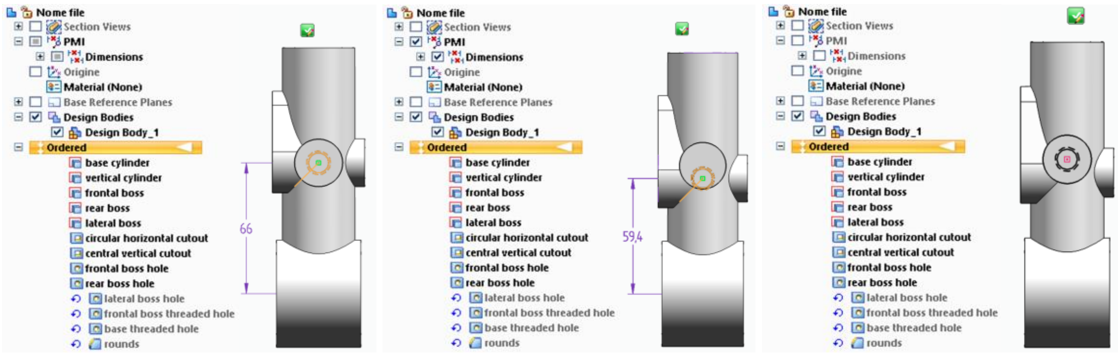
The graphical representation of the cause and effect relationship for this effect, as shown in Figure 22(a), suggests that there are two possible cause categories. As the problem is related to the boss feature and the threaded hole feature and the positioning of the latter, the effect-cause analysis can be pursued under the cause category "Feature issue" and the cause factor "Feature position" (see again Figure 22(a)).



(a) (b)

Fig. 23. Analysis of type III dormant deficiency related cause factor termed "Feature position". From left to right: (a) CAD model with latent type III dormant deficiency, feature history, and dimensions of interest, (b) CAD model with triggered type III dormant deficiency, feature history, and altered dimensions of interest.

At this point, with the knowledge that the problem is related to a profile definition that involves two features, the base effect can be determined, namely "PB-InterF Profile issue" (see also again feature class and range in Table 2). This in turn leads to the diagram shown in Figure 22(b). The base effect determined in the diagram shown in Figure 22(b) relates to two base cause categories, of which the "Inter constraint definition" is the more appropriate according to the analysis so far. This leads to five possible cause factors.



(a) (b) (c)

Fig. 24. Analysis of type III dormant deficiency related (basic) cause factor “Constraint reference keypoint selection”. From left to right: (a) CAD model with latent type III dormant deficiency, feature history, incorrect constraint reference keypoint, and dimensions of interest, (b) CAD model with triggered type III dormant deficiency, feature history, incorrect constraint reference keypoint, and altered dimensions of interest (d) amended CAD model without dormant deficiency, feature history, and correctly selected constraint reference keypoint.

Further analysis of this situation in relation to those possible factors reveals that the incorrect positioning of the threaded hole feature stems from constraints used within its profile definition, as shown in Figure 23. It seems that an error was made during the constraint reference selection, and that this problem continued through the cause factor “Constraint reference keypoint selection”, eventually leading to the root cause “Wrong projected constraint keypoint”. From the analysis carried out, it can be assumed that the type III dormant deficiency was introduced into the CAD model due to there being various overlapping snap points (see Figure 24), with the wrong one being selected accidentally as a constraint reference as shown in Figure 24(b). For the example discussed above, Figure 25 gives an overview of the analysis of the type III dormant deficiency effect-cause path and the related refining partitions of categories as used in the knowledge network.

➔ $EC^{III,r}$: *Unbalanced strength distribution*

➔ $CC^{III,m}$: *Feature issue*

➢ *Feature position*

➔ $EC^{B,n}$: *PB-InterF Profile issue*

➔ $CC^{B,l}$: *Inter constraint definition*

➢ *Constraint reference keypoint selection*

➔ **RC**: *Wrong projected constraint keypoint*

Fig. 25. Overview of the analysis of the type III dormant deficiency effect-cause path and related refining partitions of categories as used in the knowledge network.

The corresponding path graph, that is the EC^{III}-RC path $P^{III,i,j}$ that encodes and documents analysis results of type III dormant deficiencies that were detected in the *j*-th CAD model during the *i*-th analysis process for the example presented can be structured in a manner similar to that presented in the previous examples.

5. Conclusions and future work

This paper has presented and discussed a framework newly developed to support the dynamic analysis of altered parametric feature-based CAD models that fail during model regeneration due to errors in associativity. The framework is based on the novel concept of dormant deficiency and represents a first approach for introducing to the CAD field a concept that aims at systematically tackling and describing a dormant phenomenon, which is encountered – in the form of deficiency generating effects and their related symptoms – only through alteration and subsequent regeneration of parametric feature-based CAD models that are not flexible, sound, and robust. Within this new framework, dormant deficiency can be used as a key metric to describe a CAD model qualitatively and quantitatively in regard to flexibility and robustness. This can be achieved through identifying the type and number of dormant deficiencies being activated, if present, and the degree of parameter variation used during dynamic CAD model analysis.

Within the framework presented, knowledge on cause and effect relationships has been encoded and documented through sets of linked attributed *r*-partite graphs, which are the foundation of the knowledge network used in the approach. Currently, within this framework – in regard to the dormant deficiency types as introduced – basic knowledge has been compiled and encoded for the domain of parametric feature-based CAD in relation to various model deficiency generating effects and their related symptoms, cause categories and cause factors, root causes, and cause and effect relationships. Transitions of knowledge on cause and effect relationships between the encoded form and a veridical graphical representation through Ishikawa diagrams are based on sound and unambiguous correspondence relationships. Hence, through this visual aid, support is provided for the human user both for the traversing of cause and effect relationships during analysis and for the formation of new knowledge on those relationships and their respective categories and factors.

Formal encoding and documentation of knowledge within the graph-based knowledge network on cause and effect relationships, as well as individual analysis results in the form of path graphs, paves the way for new and unprecedented possibilities of furthering knowledge and insight driven analysis, eventually bringing CAD model analysis and assessment to a whole new level. At that level, perhaps, elements of artificial intelligence could also be used to aid correct and speedy recognition of effect and symptom relationships for type II and type III dormant deficiencies, thus allowing for a faster and more efficient analysis process.

The approach is scalable, from a knowledge-based perspective, as information on newly discovered deficiency generating effects and their related symptoms can be encoded and added to the knowledge network at any time. Such information is likely to be encountered during model analysis through dormant deficiencies being activated and subsequent determination of their cause factors and cause and effect relationships. The approach is also scalable from a systems-based perspective, as the current framework is not limited to the commercially available parametric CAD system that is currently being used, but can be applied to most parametric feature-based CAD systems that provide an API which can be linked to the current open system architecture of the framework. Using this approach and the prototype implementation, analysis can be performed at any stage during or after the modeling process. Thus it can be used to support students during exercise-based learning experiences and skill development and to assist teachers during CAD model analysis, the provision of formative feedback, and grading. As this approach is dynamic and simulation-based, no reference CAD model is required, and no reference solution is needed. Therefore, it overcomes a severe limitation inherent in most static inventory checking systems, like the autonomous analysis and grading systems that are currently being developed and used in academic and commercial/vocational settings.

From an educational point of view, learning how to reduce or prevent the introduction of type I and type II dormant deficiencies, which are, to a great extent engineering context independent – as they are regarded as CAD model defects in almost any engineering context – can be considered a foundation on which to develop the capability to reduce or prevent the introduction of type III dormant deficiencies. As type III dormant deficiencies depend on the engineering context, changes in a CAD model after regeneration may be acceptable in one engineering design context but not in another, thus leading to a possible type III dormant deficiency. Hence, reducing or avoiding the introduction of type I and type II dormant deficiencies relates more to learning the correct use of the modeling commands that create appropriately parameterized and constrained CAD models with suitable topology and geometry. However, avoiding the introduction of type III dormant deficiencies requires additional skills and competency for planning a correct modeling strategy while also taking into account a concrete engineering design context.

A wider and more detailed empirical evaluation of the approach presented in this paper is currently in progress within the educational context. In particular, the authors are currently examining how the dormant deficiency concept – as a key metric for CAD model quality in regard to flexibility and robustness – and a special student version of the software tool were received by students. This is based on their feedback in the form of an online survey that consists of various questionnaires. The questionnaires are structured to capture user experience and satisfaction, while also providing information on what student users expected and found useful. Administration of and data collection for the survey were executed through a learning management system and analyzed using a multi-method approach. The impact on improving performance outcomes and skills, as well as competency development, is also examined by analyzing student-created CAD models that were submitted by various student cohorts during CAD laboratory exercises both before and after introduction of the dormant deficiency concept and the software tool. Results of this empirical evaluation will be made available soon through a follow-up publication.

References

Aranburu A, Cotillas J, Justel D, Contero M, Camba J D. (2022) How does the modeling strategy influence design optimization and the automatic generation of parametric geometry variation?. Computer-Aided Design; 151:103364. https://doi.org/10.1016/j.cad.2022.103364

Ault H K, Fraser A. (2013) A comparison of manual vs. online grading for solid models. In: Proceedings of the 120th ASEE Annual Conference and Exposition. American Society for Engineering Education; Paper-No.: 7233.

Baum D. (2020) Exploring the design space of aesthetics with the repertory grid technique. In: Auber D, Valtr P (eds.) Graph Drawing and Network Visualization. Lecture Notes in Computer Science, Springer; vol. 12590: 308-323. https://doi.org/10.1007/978-3-030-68766-3_24

Beineke, L W. (2014) Families of graphs and digraphs. In: Gross J L, Yellen J, Zhang P (eds.) Handbook of Graph Theory. CRC Press, 21-30.

Bodein Y, Rose B, Caillaud E. (2014) Explicit reference modeling methodology in parametric CAD system. Computers in Industry, 65(1): 136 - 147. https://doi.org/10.1016/j.compind.2013.08.004

Bradley E. (2017) Reliability engineering: A life cycle approach. CRC Press / Taylor & Francis Group.

Bryan J A. (2020) Automatic grading software for 2D CAD files. Computer Applications in Engineering Education; 28(1): 51–61. https://doi.org/10.1002/cae.22174

Camba J, Contero M, Company P. (2016) Parametric CAD modeling: An analysis of strategies, for design reusability. Computer-Aided Design; 74: 18 – 31. https://doi.org/10.1016/j.cad.2016.01.003

Cerrone A, Hochhalter J, Heber G, and Ingraffea A. (2014) On the effects of modeling as manufactured geometry: Toward digital twin. International Journal of Aerospace Engineering; 439278: 1 - 10. https://doi.org/10.1155/2014/439278

De Luca F, Kobourov S, Purchase H. (2018) Perception of symmetries in drawings of graphs. In: Biedl T, Kerren A (eds.) Graph Drawing and Network Visualization. Lecture Notes in Computer Science, Springer; vol. 11282: 433-446. https://doi.org/10.1007/978-3-030-04414-5_31

Diestel R. (2017) Graph theory. Springer.

Elangovan U. (2020) Product Lifecycle Management (PLM), CRC Press.

Garland A P, Grigg S J. (2019) Evaluation of humans and software for grading in an engineering 3D CAD course. In: Proceedings of the 126th ASEE Annual Conference and Exposition. American Society for Engineering Education; Paper-No.: 26525.

Gebhard R. (2017) RMS Basics: Resilient Modeling Book. Volume 1, 6th edition, Richard Gebhard.

Gonzáles-Lluch C, Company P, Contero M, Camba J D, Colom J J. (2017) A case study on the use of model quality testing tools for the assessment of MCAD models and drawings. International Journal of Engineering Education; 33(5): 1643 - 1653.

Gonzáles-Lluch C, Company P, Contero M, Pérez-López D, Camba J D. (2019) On the effects of the fix geometric constraint in 2D profiles on the reusability of parametric 3D CAD models. Journal of Technology and Design Education; 29: 821 - 841. https://doi.org/10.1007/s10798-018-9458-z

Grieves M. (2006) Product Lifecycle Management: Driving the Next Generation of Lean Thinking, McGraw-Hill.

Gross J L, Yellen J, Anderson M. (2019) Graph theory and its applications. CRC Press / Taylor and Francis Group.

Gu H, Chase T R, Cheney D C, Bailey T, Johnson D. (2001) Identifying, correcting, and avoiding errors in computer-aided design models which affect interoperability. Journal of Computing in Information Science in Engineering; 1(2): 156-166. https://doi.org/10.1115/ 1.1384887

Hattie J, Timperley H. (2007) The power of feedback. Review of Educational Research; 77(1): 81 - 112. https://doi.org/10.3102/003465430298487

Harrower M, Brewer C A. (2003) ColorBrewer.org: An online tool for selecting colour schemes for maps. The Cartographic Journal; 40(1): 27 - 37. https://doi.org/10.1179/000870403235002042

Heer J, Shneiderman B. (2012) Interactive dynamics for visual analysis. Communications of the ACM 2012; 55(4): 45 - 54. https://doi.org/10.1145/2133806.2133821

Hekman K A, Gordon M T. (2014) Automated grading of first year student CAD work. ASEE Computers in Engineering Journal; 24(2): 16 - 24.

Huang W, Eades P, Hong, S-H. (2009) Measuring effectiveness of graph visualizations: A cognitive load perspective. Information Visualization; 8(3): 139-152.

Huang W. (2014) Evaluating overall quality of graph visualizations indirectly and directly. In: Huang W (ed.) Handbook of Human Centric Visualization. Springer, 373-390. https://doi.org/10.1007/978-1-4614-7485-2_14

Ingale S, Anirudh S, Bairaktarova D. (2017) CAD platform independent software for automatic grading of technical drawings. In: ASME International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, August 6 – 9, Cleveland, OH, USA, Paper-No.: DETC2017-67612. https://doi.org/10.1115/DETC2017-67612

Irons A. (2008) Enhancing Learning Through Formative Assessment and Feedback. Routledge.

Ishikawa K. (1991) Guide to quality control. Asian Productivity Organization.

Jaakma K, Kiviluoma P. (2019) Auto-assessment tools for mechanical computer aided design education. Heliyon; 35(1): 60-78. https://doi.org/10.1016/j.heliyon.2019.e02622

Jenny B, Kelso, N V. (2007) Color design for the color vision impaired. Cartographic Perspectives; 58: 61-67. https://doi.org/10.14714/CP58270

Johnson W R. (2018) Detecting plagiarism in SolidWorks CAD courses. In: Proceedings of the ASEE Annual Conference and Exposition. American Society for Engineering Education, Paper-No.: 23898. https://doi.org/10.18260/1-2--30286

Johnson, J. (2020) Designing with the Mind in Mind: Simple guide to understanding user interface design guidelines. Morgan Kaufmann.

Joo S.-H. (2018) Assessment of three dimensional CAD models using CAD application programming interface. In: Proceedings of the ASME International Mechanical Engineering Congress and Exposition, November 9 -15, Pittsburgh, PA, USA, Paper-No.: IMECE2018-87776. https://doi.org/10.1115/IMECE2018-87776

Khaleel M F, Sharkh M A, Kalil M. (2020) A cloud-based architecture for automated grading of computer-aided design student work using deep learning. In: Proceedings of the IEEE Canadian Conference on Electrical and Computer Engineering, August 30 - September 2, London, ON, Canada, pp. 1-5. https://doi.org/10.1109/CCECE47787.2020.9255825

Kirstukas S J. (2016) Development and evaluation of a computer program to assess student CAD models. In: Proceedings of the 123rd ASEE Annual Conference and Exposition. American Society for Engineering Education; Paper-No.: 15834.

Kurosu M, Kashimura K. (1995) Apparent usability vs. inherent usability: Experimental analysis on the determinants of the apparent usability. In: Proceedings of the ACM CHI Conference Companion on Human Factors in Computing Systems; 292-293. https://doi.org/10.1145/223355.223680

Lidwell W, Holden K, Butler K. (2010) Universal design principles. Rockport Publishers.

Mandorli F, Otto H E. (2021) A Systematic Approach to Innovative MCAD Education Based on Negative Knowledge Development and Formative Feedback, in: Rizzi C, Campana F, Bici M, Gherardini F, Ingrassia T, Cicconi P (eds.), Lecture Notes in Mechanical Engineering (Design Tools and Methods in Industrial Engineering II), Springer, 839-850. https://doi.org/10.1007/978-3-030-91234-5_85

Mandorli F, Otto H E. (2022) Improving the learning experience within MCAD education: A tool for students to assist in self-assessment during modeling exercises. Computer-Aided Design and Applications; 19 (3): 534 - 560. https://doi.org/10.14733/cadaps.2022.534-560

Misue K, Zhou Q. (2011) Drawing semi-bipartite graphs in anchor+matrix style. In: Proceedings of the 15th International Conference on Information Visualization. IEEE Computer Society; 26-31. https://doi.org/10.1109/IV.2011.24

Misue K. (2006) Drawing bipartite graphs as anchored maps. In: Proceedings of the Asia Pacific Symposium on Information Visualization. Australian Computer Society; 169-177.

Moon J, Park D. (2021) 3D Printing Signboard Production Using 3D Modeling Design, in: Kim J, Lee R (eds.), Data Science and Digital Transformation in the Fourth Industrial Revolution, Springer, 136 – 149. https://doi.org/10.1007/978-3-030-64769-8_9

Moyers S. (2017) The F pattern: Understanding how users scan content. UX Magazine; 7, Paper-No.: 1687.

Munzner, T. (2014) Visualization, Analysis and Design, CRC Press.

Nerenst T B, Ebro M, Nielsen M H, Eifler T, Nielsen K L. (2023) Parametric CAD modeling: New principles for robust sketch constraints. Computer-Aided Design and Applications; 20(1): 56 - 81. https://doi.org/10.14733/cadaps.2023.56-81

Otto H E, Mandorli F. (2015) A framework to support 3D explicit modeling education and practice. Computer-Aided Design and Applications; 12 (1): 104 - 117. https://doi.org/10.1080/16864360.2014.949581

Otto H E, Mandorli F. (2018) A framework for negative knowledge to support hybrid geometric modeling education for product engineering. Journal of Computational Design and Engineering; 5(1): 80-93. https://doi.org/10.1016/j.jcde.2017.11.006

Otto H E, Mandorli F. (2021) Parametric feature-based solid model deficiency identification to support learning outcomes assessment in CAD education, Computer-Aided Design and Applications; 18(2): 411-442. https://doi.org/10.14733/cadaps.2021.411-442

Papadopoulos C, Voglis C. (2013) Untangling graphs representing spatial relationships driven by drawing aesthetics. In: ACM International Conference Proceeding Series; 158-165. https://doi.org/10.1145/2491845.2491853

Purchase H C. (2002) Metrics for graph drawing aesthetics. Journal of Visual Languages and Computing; 13(5): 501-516. https://doi.org/10.1016/S1045-926X(02)90232-6

Purchase H C, Carrington D, Allder, J A. (2002) Empirical evaluation of aesthetics-based graph layout. Empirical Software Engineering; 7(3): 233-255. https://doi.org/10.1023/A:1016344215610

Saraiya P, North C, Lam V, Duca K. (2006) An insight-based longitudinal study of visual analytics. IEEE Transactions on Visualization and Computer Graphics; 12(6): 1511-1522.

Schlatter T, Levinson D. (2013) Visual usability: Principles and practices for designing digital applications. Morgan Kaufmann.

Stark J. (2015) Product Lifecycle Management, Volume 1: 21st Century Paradigm for Product Realisation, Springer.

Stark J. (2020) Digital Transformation of Industry: Continuing Change, Springer.

Tague, N. (2005) The quality toolbox. American Society for Quality Press.

Tractinsky N, Katz A S, Ikar D. (2000) What is beautiful is usable. Interaction with Computers; 13(2): 127–145.

Verma G, Weber M. (2021) SolidEdge 2022 Black Book, Cadcamcae Works.

Ware C, Purchase H, Colpoys L, McGill M. (2002) Cognitive measurements of graph aesthetics. Information Visualization; 1(2): 103-110. https://doi.org/10.1057/palgrave.ivs.9500013

Ware C. (2008) Visual thinking for design. Morgan Kaufmann.

Ware C. (2020) Information visualization: Perception for design, Elsevier.

Yablonski J. (2020) Laws of UX: Using psychology to design better products & services. O'Reilly.

Younes R, Bairaktarova D. (2022) ViTA: A flexible CAD-tool-independent automatic grading platform for two-dimensional CAD drawings. International Journal of Mechanical Engineering Education; 50(1): 135-157. https://doi.org/10.1177/0306419020947688

Zheng L, Song L, Eades P. (2005) Crossing minimization problems of drawing bipartite graphs in two clusters. In: Proceedings of the Asia Pacific Symposium on Information Visualization. Australian Computer Society; 33-37.

Zou Q, Feng H-Y, Gao S. (2023) Variational direct modeling: A framework towards integration of parametric modeling and direct modeling in CAD. Computer-Aided Design; 157:103465. https://doi.org/10.1016/j.cad.2022.103465