An Ontology-based Tool for Supporting the Constraining Strategy of MCAD Objects

Sofia Kyratzi[1] and Philip Azariadis[2]

[1]Universigy of the Aegean, skiratzi@aegean.gr
[2]Univeristy of West Attica, fazariadis@uniwa.gr

Corresponding author: Sofia Kyratzi, skiratzi@aegean.gr

Abstract. Current 3D modeling strategies focus on the creation of parametric MCAD models that effectively convey design intent. An important step to this effort is a proper constraining schema that establishes the geometric design intent of the model. We employ a direct and structured correspondence between different constraining schemas (called meta-constraints) and their inferring design intent (called intention regularities), and we propose an ontological-based framework that describes the geometric design intent of a 3D model and indicates a constraining strategy for the establishment of a predefined design intent.

Keywords: 3D parametric modeling, geometric design intent, ontologies, feature-based design.
DOI: https://doi.org/10.14733/cadaps.2024.659-676

1 INTRODUCTION

Current 3D modeling strategies focus on the creation of mechanical 3D parametric models (MCAD models) that effectively convey design intent. Design intent admits multiple definitions that all refer to the underlying rationale behind an object and the design decisions for model's geometry, and engineering/manufacturing information that is associated with it [14][16]. Modern parametric design environments support three design levels for the geometric definition of a 3D model, i.e., the sketch level, the feature level, and the model level [16][28]. Features are the backbone of a 3D model, while constraints and relations indicate the association of the multiple features in the model. Features, parameters, constraints, and relations are considered as a carrier of design intent and form the language of designers to communicate their design ideas [3][22].

Since a 3D model is usually modified throughout the whole product lifecycle, or it becomes the base for the creation of a new 3D model [6], modeling tools and constraining schemas that express design intent become an efficient tool for the designer to control the behavior of the geometric model during design and modification tasks [24]. Additionally, a modeling strategy that considers design intent, results in a flexible and robust model that allows changes to its structure without rendering to inconsistencies [5][24][28].

Among researchers in the field of parametric modeling there is a consensus for the need of efficient design and constraining strategies that establish design intent during modeling. The formation, though, of such strategies is a challenging task since there are multiple designing and constraining solutions for the geometric definition of a 3D model. Thus, still the expression of design intent in MCAD parametric models depends mainly on the experience of designers to establish an appropriate constraining schema that best capture the geometric and topologic attributes of the object and effectively reflects its modification behavior [32]. As Salehi & McMahon [29] point out, it is not an easy task for designers to identify possible methods to incorporate the design intent with geometric entities. From the research they conducted in automotive design industry, they concluded that designers show significant difficulties to identify appropriate parameters and even more to determine associative geometric entities and represent the relationships and constraints that will define the geometry of the 3D model.

We further emphasize that a key challenge towards the definition of a proper constraining schema that establishes the geometric design intent of the model, is the understanding of the design intent that the features or the sketches should convey. In previous research works [19][20], we proposed the Integrated Design Intent Architecture (IDI Architecture). The IDI Architecture indicates a direct and structured correspondence between different constraining schemas (called meta-constraints) and their inferring design intent (called intention regularities). IDI Architecture provides a structure to gradually capture the design intent of a model while it is being created (bottom – up approach) or signify a set of modeling steps and constraining schemas that comply with a predefined design intent (top-down approach).

In this research work, we further explore the employment of IDI Architecture in the context of the top-down approach. The top-down approach is related to the design of the modeling strategy, and it involves a succession of intention regularities from the upper (model) to the lower (sketch) level [20]. The description of these successions requires the development of a knowledge-based system in the domain of design intent and the corresponding constraint schemas. Under this scope, in this paper we revisit the IDI Architecture from an ontological perspective. We propose an ontological framework, named as IDI Ontology, that describes the geometric design intent of a 3D model in the context of the top-down approach. Being oriented around different design intents that are met in MCAD models, the ontological framework represents these design intents with respect to different constraining schemas that establish them. This is achieved by the association of meta-constraints with intention regularities in the three design levels. The ontological framework operates independently from the geometric attributes of features or specific methodological approaches. As a result, it contributes crosswise to the development of modeling strategies by incorporating design expertise concepts.

Having as a key challenge the design of a constraining strategy, the primary objective of IDI Ontology is to indicate, for predefined design intents, an appropriate sequence of intention regularities, i.e., design intents, at the three design levels. Starting from the model level and a given design intent, the proposed ontological framework signifies the design intent that the features/sketches should convey to assert the intent of the resulting model. The understanding of the design intent of the features in a model can effectively contribute to an efficient design strategy and consequently to the utilization of proper constraining schemas. Additionally, and by exploiting the IDI Architecture, the IDI Ontology indicates an appropriate constraining schema for a given design intent, in terms of meta-constraints or their traditional constraints counter parts [20].

The paper is structured as follows. Section 2 reviews two research fields that are related to the proposed work: the research work in the field of parametric modeling methodologies and design intent, and the employment of ontologies in the context of CAD models. In this section we further discuss the contribution of the proposed IDI Ontology with respect to existing methods. Section 3 includes a brief description of IDI Architecture and Section 4 presents the IDI Ontology that is based on this Architecture. Section 5 presents two examples for the design of an MCAD model that use the proposed ontology. Section 6 concludes the paper by discussing the advantages of the proposed approach to the design of a modeling a strategy and by presenting possible applications and future extensions.

2 RELATED WORK

2.1 Constraints and Design Intent in Parametric Modeling

CAD software utilizes a geometric definition for 3D models, which is translated into a parametric problem within the constraint solver. This problem is characterized by a set of constraints and parameters, which are essential tools for designers to express their design intent and define the 3D geometry of their models. Under this scope, research on parametric modeling and design intent expands in different paths related to efficient parametric modeling practices and knowledge-based approaches to CAD education [8][17-18][24-25], studies on the effect of different constraint schemas on the 3D model and frameworks for the communication and representation of design intent in 3D models [2][7][11][13][16][20][22], and studies related to how designers realize the design problem, consider and analyze their different options, and how they finally process during modeling and modification tasks [6][17-18]. These research paths explore under different perspectives the same problem, that of the development of an effective constraining and modeling strategy for the creation of 3D parametric objects that effectively convey their design intent. Ottey et al. [24] conducted an extensive study on the use of design intent in CAD education and modeling practices, which revealed a strong correlation between the effective representation and communication of design intent in MCAD models and the successful acquisition of MCAD tools by learners.

In the field of CAD education, there is a focus on a more strategic knowledge schema for CAD education that will help students understand the alternative methods by which a task may be done, and to use CAD systems as knowledge-intensive design and communication tools [22]. With a focus on issues related to MCAD education and practice, Otto & Mandorli [25] proposed functional dimensioning features to support a methodological approach to explicit modeling. Barbero et al., [5] focus on how and at what degree the summaries of design rules and design exercises can improve the student learning to modeling under different design concepts. Garikano et al., [15] set a number of structured activities that provide student with guidelines to think about the modelling rationale and consider alternative modeling strategies. Their method starts with the definition of a cognitive visual model that is the first step to the geometric interpretation and inference of the design intent.

In the field of modeling and constraining practices, Yin & Ma [33] proposed feature parameter maps with the aim to explicitly manage feature dependencies in a model through the usage of their defining parameters. In [12] Contero et al., proposed a set of quality criteria through the use of rubrics and studied the effect of modeling practices to the creation of quality CAD models. Camba et al., [9] reviewed and analyzed three major modeling methodologies, horizontal modeling, explicit reference modeling, and resilient modeling, towards reusability of the produced 3D models. In [28] Rynne & Gaughran emphasize the importance for designers to have a semantic knowledge of the tools they use for creating and manipulating geometry in any MCAD software. A crucial skill in this regard is the ability to decompose geometric elements, and cognitively assemble them in a way that aligns with the design intent. Bodein et al., [8] propose a practical approach for modeling complex parts within parametric CAD systems which prioritizes an explicit management of relationships between features, using explicit references. Aranburu et al. [4], identified inefficiencies in common industrial practices used for constructing 3D models, with the aim to present the extent of the problem of creating a robust and flexible 3D model.

2.2 Ontological Representations in CAD

Ontologies find multiple applications in engineering design, including the representation of a CAD model's geometry [26], the semantic integration of product and manufacturing information between CAD/CAE/CAM systems [1], and the semantic representation of features [23][27][31]. The main objective of semantic representation of features is to enhance the cognitive understanding of the final model and its representation within the field of engineering knowledge. In [30] Sanfilippo & Borgo reviewed the status of features with respect to their semantic representation and proposed an ontology-based perspective for the definition of a feature. They distinguish two types of features in their ontological structure, p-feature, that is the physical object and its counterpart i-feature, which is an information entity that includes the product description and its functional reasoning.

Following the ontological description of features presented in [30], Sanfilippo [31] provides a further ontological analysis of how features are understood in engineering knowledge with a focus on the fundamental (ontological) properties that features satisfy independently from specific applications and contexts of usage. Mandorli et al. [23], investigate the present feature-based MCAD limitations from an ontological perspective and propose an ontological analysis and re-classification of features to support semantic, and in particular functional, representations of features. In their work, they apply their ontological perspective to the hole feature. Another feature – oriented ontology is presented by Qin et al. [27] with feature types and different classifications of them to set the classes of the ontology. The ontology presents the semantic information of each feature in CAD domain and is employed for a semantic retrieval approach for heterogeneous 3D CAD models.

In the same research path, Chenga et al., [10] developed a method to enhance collaborative design processes by integrating an information model for representing design intents with a CAD model Ontology. The method involves the creation of a common CAD model ontology and the generation of a semantic network that captures the relationships between these components in the design intent information. In the context of CAD model retrieval, Ma et al. [21] investigated the potential of ontology semantic tree models for representing the design intent of 3D objects as design knowledge annotations.

2.3 Paper Contribution

The proposed IDI Ontology explores the associations of meta-constraints (i.e., constraint schemas) and the corresponding intention regularities (i.e., common design intents in MCAD models) at the sketch, feature, and model level. IDI Ontology represents the design intent of MCAD models in terms of constraint schemas in the three design levels. For a predefined set of design intents, its main focus is to provide a sequence of intention regularities and associating constraining schemas that establish it.

The IDI Ontology follows a different perspective to the description of design intent in comparison to existing ontological approaches for MCAD models. Following the multiple roles of a feature in a 3D model, most of the existing ontological frameworks are feature-oriented in an effort to depict the variety of semantic meanings of features. In these works, different feature types define the classes of the ontology while design intent, geometric description, and functional characteristics are properties of the different feature entities in the ontology. In IDI Ontology the key classes of the ontological framework are intention regularities and constraining schemas. The choice of a mathematical representation of features and standard constraints for their definitions are considered as a design choice towards a certain design intention. The advantage of such an approach is that it is targeted to the design intent and the corresponding constraining schemas and can be effectively employed for the design of constraining strategy independently of the CAD software that is used.

In MCAD modeling, there are often various ways of generating a feature. This variety of options requires additional effort from the CAD designer to understand the different alternatives and how to choose between them [6]. Based on the literature review of parametric modeling, most of the research works focus on modeling practices or assisting tools during the modeling phase, with only few theoretical studies to be involved to the preparatory steps that are related to the design of the constraining strategy [6][17-18]. The proposed IDI Ontology contributes to the design of the constraining strategy with a structured set of multiple alternative schemas and offers to the designers the vehicle to acquire an integrated insight on the design intent that should be established for a 3D object during modeling. Moreover, the IDI Ontology contributes also to the knowledge-based approach in CAD education. With the use of IDI Ontology, designers can access, study, and select different design and constraining approaches that establish the same design intent.

3 THE INTEGRATED DESIGN INTENT (IDI) ARCHITECTURE

The IDI Architecture is analyzed and discussed in [20]. Here we briefly present the framework in the context of the proposed IDI Ontology. The objective of IDI Architecture is to capture the design intent of a sketch/feature/model as this is generated by the constraining choices of a designer. It sets the design intent, via the pair "meta-constraints" – "intention regularities". Intention Regularities (IR) are defined as geometric or topologic patterns that appear in engineering objects and can be recognized as design intentions. Meta-constraints (MC) are constraints defined by the combination of geometric entities, attributes, and standard constraints that geometrically and/or semantically express an intention regularity. Meta-constraints and intention regularities are part of the constraining scheme of a 3D model and convey a pre-defined intention. Meta-constraints are considered as enhanced constraint schemas that include parameters, constraints and relations that establish and preserve a design intent/intention regularity. Thus, they manage to define the geometry of a feature along with capturing feature semantics. Accordingly, Integrated Design Intent refers to the design intent of a model that is generated by the intention regularities of its sketches and features.

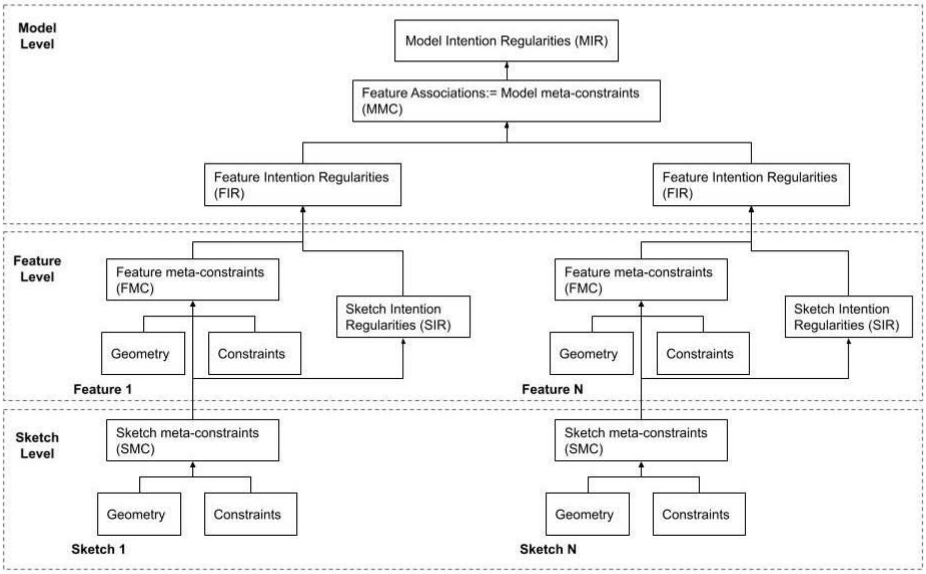


Figure 1: The components and associations of the IDI Architecture [20].

The data structure and the inheritance of IDI Architecture between modeling levels are shown in Figure 1. In [20] we identified and analyzed multiple meta-constraints and intention regularities at each design level. Tables 1 & 2 present indicative meta-constraints/intention regularities and the design intent they convey.

Sketch Level			
SMC/ SIR	*Sketch Design Intent*	*SMC / SIR*	*Sketch Design Intent*
SMC_Side/ SIR_Side	A sketch placed on one or two adjacent quadrants of the reference planes.	SMC_CenterSymmetric/ SIR_CenterSymmetric	Design of a cycle which center coincides with an axis.
SMC_FaceInnerLoop/ SIR_FaceInnerLoop	A sketch defines an internal loop to a closed boundary of a selected face.	SMC_Hole/ SIR_Hole	An inner loop that defines a hole to an outer loop.
		SMC_(XY)AxesSymmetric/ SIR_(XY)AxesSymmetric	Symmetry about one or both axes.
SMC_BoundaryCentered/ SIR_BoundaryCentered	Centrality with reference to an outer selected loop	SMC_XYAxesCentered/ SIR_XYAxesCentered	Centrality about one or both axes.

Feature Level					
Feature Constraints	*SMC*	*FMC*	*SIR*	*FIR*	*Feature Design Intent*
		FMC_Protrusion And	SIR_Center Symmetric	FIR_Axial Symmetric	An axial symmetric feature.
			SIR_Hole	FIR_PartialHole	A through hole up to a selected surface.
			SIR_Face InnerLoop	FIR_OnFace	A feature that lies on the face of a pre-existing feature.
			SIR_XYAxes Symmetric	FIR_Plane Symmetric(2)	A feature symmetric to two planes.
			SIR_Boundary Centered	FIR_OnFace Centered	A feature centered on an existing selected face.
			SIR_XYAxes Centered	FIR_Axes Centered (2)	An axis centered feature.
Symmetric		FMC_SPSymmetric		FIR_SP Symmetric	A feature symmetric to its sketch plane.
Angle=360°	SMC_Side	FMC_Axial Symmetric		FIR_Axial Symmetric	An axis symmetric feature.
Angle<360°		FMC_AxialShape		FIR_Axial	An axial feature.
Through-All	SMC_Face InnerLoop	FMC_ThroughHole		FIR_Through Hole	A through hole feature.
Blind		FMC_BlindHole		FIR_BlindHole	A blind hole feature.
Up-to-Selected-Surface		FMC_PartialHole		FIR_Partial Hole	A through hole up to a selected surface.

Table 1: Indicative meta-constraints and intention regularities at the sketch and feature level.

Each of the three design levels, i.e., sketch, feature and model, includes a set of meta-constraints and intention regularities, named respectively as SMC/SIR, FMC/FIR, and MMC/MIR. At sketch level there is a one-to-one correspondence between sketch meta-constraints and sketch intention regularities. At the feature level, each feature meta-constraint is linked to a mathematical representation, geometric attributes, feature constraints, and sketch meta-constraints. Feature meta-constraints reflect different constraining and geometric schemas that infer a specific design intention. Feature intention regularities are generated by the combination of feature meta-constraints and sketch intention regularities. Model meta-constraints reflect constraints attached between features which are not expressed by feature properties (i.e., within FMCs). The Model Intention Regularities are built upon the Feature Intention Regularities and feature associations and generate the Integrated Design Intent of the model.

Model Level			
Combination of FIRs		*MMC*	*Model Intention Regularities*
FIR_AxialSymmetric	FIR_AxialSymmetric	RefAxes	MIR_CoaxialSymmetric
FIR_AxialShape	FIR_AxialShape	RefAxes	MIR_Coaxial
	FIR_AxialSymmetric		
FIR_Symmetric	FIR_Symmetric	SPAttached	MIR_Symmetric
FIR_SPSymmetric	FIR_SPSymmetric	SPAttached	MIR_SinglePlaneSymmetric
	FIR_AxialSymmetric		

Table 2: Indicative meta-constraints and intention regularities at the model level.

4 THE IDI ONTOLOGICAL FRAMEWORK FOR THE TOP-DOWN APPROACH

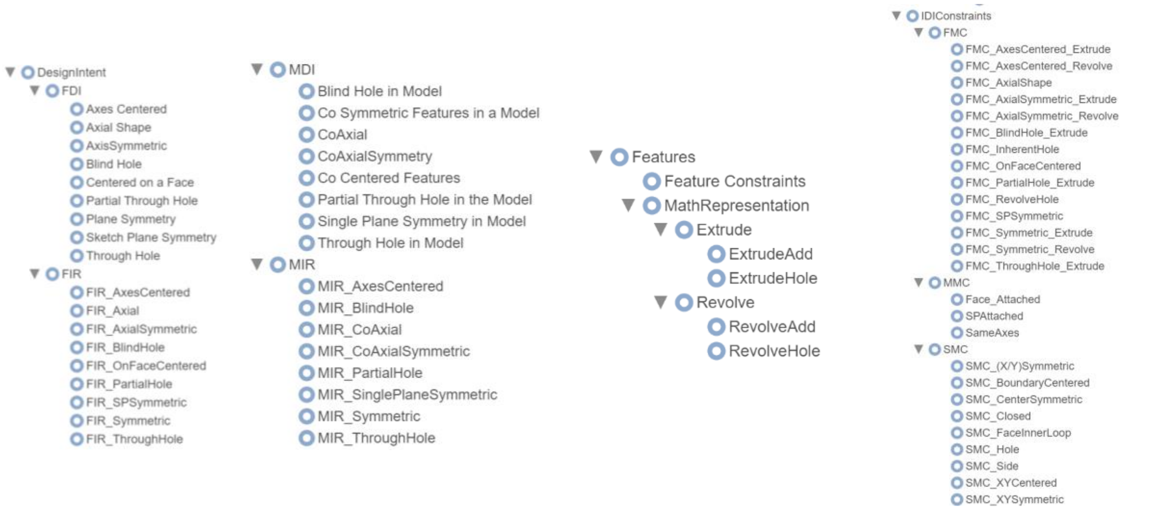
The IDI Ontology is a knowledge-based system that is built upon IDI Architecture and aims at capturing the domain knowledge of geometric design intent of 3D MCAD models. In the context of IDI Architecture, design intent is described by meta-constraints and intention regularities. Intention regularity is a direct expression of a design intent, and it is established by a meta-constraint. In our approach every design choice signifies a design intent, from standard constraining schemas to the selection of the mathematical representation (i.e., modeling tool) and feature attributes, such as remove/add material or feature geometric constraints. For the proper definition of the ontological framework, we specify as principal question: "What conditions/constraints should an MCAD object satisfy to have a certain design intent?".

In this research work we focus on "Hole", "Axial Symmetry" and "Full Symmetry" design intents and on "Extrude" and "Revolve" mathematical representations. In order to capture the rationale behind these different types of design intents and to support constraining and semantic paths at the three design levels, the ontology should store and use the information that express these design intents in terms of intention regularities and meta-constraints. A query is set on terms of model or feature design intents (sub-classes of Design Intent class in Figure 2), and the results expand to constraining schemas from the feature level (FMCs) towards to the sketch level (SMCs) combined with standard constraints or proper annotations when needed (IDI Constraints and Features classes in Figure 2). The ontology is organized in such a way so that each query concerns an individual design intent. If a model/feature includes more than one design intents, these should be queried independently. For example, a feature with an axial symmetric through hole, requires a query about the creation of an axial symmetric object and a query about the definition of a through hole.

Considering that the main components of a 3D model are the features, the ontology is implemented upon the design intent at the feature level. This implementation serves best the purpose of the ontological framework, since it manages to effectively capture, through Feature Intention Regularities (FIRs) and their associating Feature Meta-Constraints (FMCs), the design intents at the three design levels. This structure is further explained in the following subsections.

4.1 Class Hierarchy and Properties

Following the taxonomic structure of OWL Ontologies, the IDI Ontology includes a class for each entity that configures the design intent of a model. Thus, the components of IDI Architecture (Figure 1) are classified into three classes that form the three main branches of the ontology. The classes are the "Design Intent", "Features", and "IDI Constraints" class (Figure 2). "Design Intent" class includes expressions of geometric design intent both in verbalization manner and in the form of Model (MIR) and Feature (FIR) Intention Regularities. "Features" class involves main attributes of the geometric definition of features that are related to design intent rationale. The "IDI Constraints" class includes the meta-constraints in the sketch, feature and model level. SIRs are not included in this ontological framework, because they admit a one-to-one correspondence with SMCs. Classes are further divided into sub-classes that capture the different types of intention regularities and meta-constraints.



Design Intent subclasses Features subclasses IDI Constraints subclasses

Figure 2: The class hierarchy of IDI Ontology.

The "Design intent" class is divided in four sub-classes. The sub-classes FIR and MIR correspond to the intention regularities of the IDI Architecture. The sub-classes FDI (Feature Design Intent) and MDI (Model Design Intent) express a verbalization form of the design intentions in FIR and MIR classes, and they are introduced for a more intuitive interaction of MCAD designers with the ontological framework. They are employed to set the queries in the ontology. The four sub-classes are further divided into different types of design intentions (their description is presented in Table 1). For the "Hole" case, we distinguish three design intents, that of "Blind Hole", "Through Hole", and "Partial Hole". For the symmetry design intent, we consider "Axial Shape", "Axis Symmetry", and "Full Symmetry". These design intents, with an appropriate expression, are included as sub-classes in FIR and FID classes. At the model level, the ontology considers two features that can be "CoAxial", "CoAxialSymmetric", and "Symmetric", or to define a model with a through/blind/partial hole. Each case is also included as a sub-class to MIR and MID classes.

The "Features" class is divided to two sub-classes, "Feature Constraints" and "MathRepresentation". The "MathRepresentation" class includes as sub-classes the mathematical representation of extrude and revolve. The "remove material" or "add material" are considered as a design choice and imply additional meta-constraints for the definition of the feature. In this sense, each mathematical representation includes an add and hole sub-class that are associated with the add/remove material property. This strategy allows FMCs to be defined independently to whether they are applied to a protrusion or a depression. The "Feature Constraints" class includes, as individuals, constraints that concern the properties of a feature in terms of geometry. In current implementation we define individuals “Add Material”, “Angle360”, “AngleDif360”, “Remove Material”, “Specific Value”, “Through All”, “Symmetric” and “Up to Selected Surface”.

The “IDIConstraints” class is divided in three sub-classes, named as FMC, MMC and SMC, and correspond to the meta-constraints at the three design levels. MMC and SMC classes include as sub-classes a variety of meta-constraints at the model and sketch level respectively. The FMC class includes as sub-classes different meta-constraint schemas at the feature level. Each sub-class in the FMC class corresponds to an intention regularity from the FIR class and is related to some SMC sub-classes. The associations between FMC, SMC and FIR classes are captured through the object properties of the ontology (Table 3). For a design intent from the FIR class there may be different combinations of FMCs and SMCs that establish it. If, for a FIR, there are alternative constraining schemas, we associate this FIR with distinguished FMCs expressions so to capture these alternatives. These FMCs expressions are inserted as different sub-classes in FMC class. This distinction mainly occurs in terms of mathematic representations (e.g., the FMC_AxialSymmetric_Revolve and FMC_AxialSymmetric_Extrude sub-classes of Figure 2). This technique manages to capture the different meta-constraints that are indicated due to a mathematic representation and is in accordance with our design strategy of IDI Ontology that considers mathematical representation as a design intent choice.

The relations between the classes are set by the object and annotation properties of the ontology. Object properties are relationships that have as domain and range classes in the taxonomy (possibly the same) and are used to represent how elements of the classes are related to each other. We also include the annotation property “hasFIR” with the aim to associate FMC sub-class with FIR sub-class. The properties of the ontology are summarized in Table 3 and are further explained below in the context they are employed.

Property	***Domain***	***Range***	***Scope***
hasDesignIntent	MIR/FIR	MDI/FDI	Vocabular form of Design Intent.
haspFDI/hassFDI	MIR	FDI	The design intent of primal and secondary feature in the model.
hasMMC/hasFMC/hasSMC	MIR/FIR/FMC & Features	MMC/FMC/SMC	The model/feature/sketch meta-constraints that establish a FIR.
hasMathRepres	FMC	MathRepresentation	Relates the mathematical representation with a Feature meta-constraint expression.
hasDepth/Attribute/Angle	Features/FMC	Feature Constraints	Determine constraints and attributes at the feature level.

Table 3: The object properties of the IDI Ontology are used to convey design intent of 3D Model.

The “hasDesignIntent” property associates the vocabular form of design intent with the corresponding intention regularities in the domain of FIR and MIR classes. Each FIR/MIR subclass is connected through this property with an FDI/MDI subclass. The “hasDesignIntent” is employed to initiate the queries in the ontology in terms of design intent. Given two features and a design intent at the model level, the “haspFDI” and “hassFDI” properties indicated the design intent that should be established for each feature. These two properties along with the “hasDesignIntent” property convey the design intent from the model to the feature level.

The “hasFMC” property relates a feature intention regularity (i.e., the FIR sub-classes) with one or multiple feature meta-constraints (FMC sub-classes). This property asserts that the ontological framework will result to appropriate constraining schemas that will establish a predefined design intent. The “hasSMC” property has domain the sub-classes of Feature or FMC classes and range the SMC sub-classes. Each FMC entity or Feature entity has at least one defined “hasSMC” property. Thus, the “hasSMC” property conveys the design intent from the feature to the sketch level and asserts that the MCAD designer will acquire an integrated insight about the design intent at the three design levels. The "hasMathRepres" property relates each FMC sub-class with a mathematical representation and by this inserts the selection of modeling operation as a design intent decision within the constraining schema. Finally, the "hasDepth", "hasAttribute" and "hasAngle" properties serve the same scope, which is to indicate constraints and/or generic attributes that are related to the geometric definition of a feature.

Bellow, two types of design intents are discussed that are commonly met in MCAD models as they are represented in IDI Ontology in the context of top-down approach.

4.2 Types of Holes in IDI Ontology

In IDI Ontology the term "hole" does not refer to a hole feature, but to the "hole" as a design intent. In this research work we consider three types of holes, "Through Hole", "Blind Hole", and "Partial Hole" and we distinguish three intention regularities that correspond to the three types of holes, FIR_ThroughHole, FIR_BlindHole, and FIR_PartialHole. The representation of a hole in IDI Ontology focuses only on the appropriate constraints that establish and preserve the intention of creating a hole, at feature and sketch level. In the scope of this paper, a hole can be created by either extrude or revolve operations. Both mathematical representations require a closed sketch. This design intent is related with the Extrude and Revolve classes and is established with the SMC_Closed via the "hasSMC" object property. Additionally, the revolve mathematical representation necessitates the creation of a sided sketch and indicates it with the use of SMC_Sided constraint. For the integrated representation of a hole design intent, additional meta-constraints are required with respect to the hole type and the selected mathematical representation.

The through hole design intent, FIR_ThroughHole (Figure 3), is established by the FMC_ThroughHole_Extrude and FMC_RevolveHole meta-constraints that capture the two alternative options for the creation of a through hole. These two FMCs are linked respectively, via the "hasMathRepres" object property, with the "ExtrudeHole" and "RevolveHole" mathematical representations. Both representations require a "remove material" attribute. The "ExtrudeHole" requires additionally the SMC_FaceInnerLoop meta-constraint in the constraint schema, which implies that the sketch should always be placed inside the boundary of the face to be subtracted from. The FMC_ThroughHole_Extrude has its "hasDepth" property set to "Through All", which is a defining property for the establishment of the "through hole" design intent. The "RevolveHole" does not require, in terms of constraints, any additional to the SMC_Closed and SMC_Sided meta-constraints. In the case of revolve mathematical representation, the type of hole that can be created depends on the geometry of the sketched profile. The incorporation of meta-constraints for the profile of each hole type in IDI Ontology remains an area for future study.

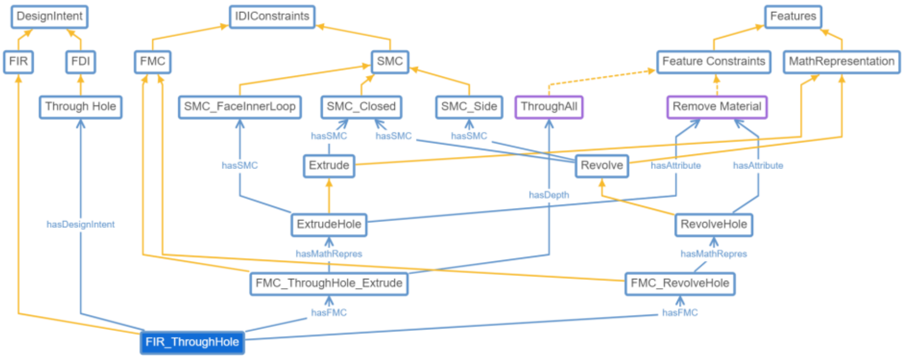


Figure 3: The constraint schemas that establish a through hole design intent.

The blind hole design intent, FIR_BlindHole (Figure 4), is linked, via the "hasFMC" property, with the FMC_BlindHole_Extrude and FMC_RevolveHole meta-constraints, which correspond to two alternative schemas that establish the design intent. The two FMCs are linked respectively, via the "hasMathRepres" object property, with the "ExtrudeHole" and "RevolveHole" mathematical representations. Consequently, the constraint schemas and attributes that are indicated for this case are similar to those defined for the "Through Hole" design intent. The only difference is the "hasDepth" property that is set to the "Specific Value" option. We introduce in IDI Ontology different classes for each type of hole, in order to represent different design intents.

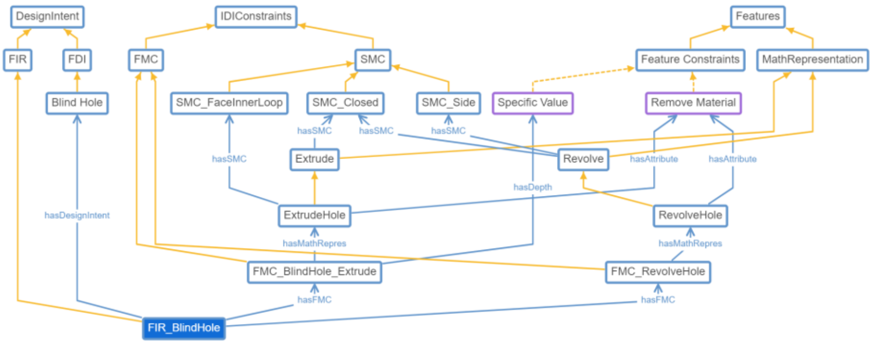


Figure 4: The constraint schemas that establish a blind hole design intent.

The partial through hole design intent, FIR_PartialHole (Figure 5), is implemented by three different feature meta-constraints, the FMC_PartialHole_Extrude, FMC_RevolveHole, and FMC_InherentHole that correspond to the three alternative constraint schemas that establish this design intent. The constraint schemas indicated by FMC_PartialHole_Extrude and FMC_PartialHole_Revolve are similar to those of the other two types of holes, with the difference that the "hasDepth" property is set to "Up to Selected Surface". The FMC_InherentHole refers to a hole that is automatically created when a sketch includes islands when designed. For that reason, it indicates, via the "hasSMC" property the SMC_Hole meta-constraint. Additionally, the FMC_InherentHole is assigned, via the "hasMathRepres", to the "ExtrudeAdd" mathematical representation and requires an "Add Material" attribute for the completion of the partial hole. Thus, the hole is created along with the extrusion of an outer profile/sketch and will constantly be attached to the surfaces of the extrusion.

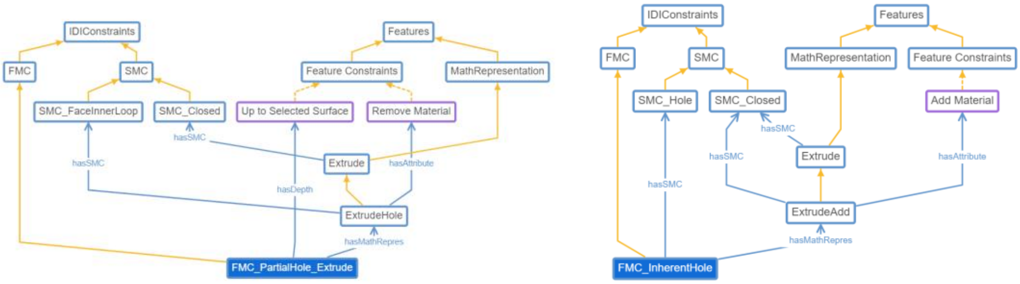


Figure 5: The constraint schemas that establish a partial though hole design intent.

4.3 Axial and Full Symmetry in IDI Ontology

In this paper we focus on two types of symmetry: Axial Symmetry and Full Symmetry. With the aim to describe the constraint schemas that capture these design intents, we define two intention regularities FIR_AxialSymmetric and FIR_Symmetric. Both axial symmetry and symmetry can be achieved with extrude and revolve operations. Thus, the IDI ontological framework is structured as follows.

The "Axial Symmetry" design intent, FIR_AxialSymmetry is linked, via "hasFMC" object property, to FMC_AxialSymmetric_Revolve and FMC_AxialSymmetric_Extrude (Figure 6), that correspond to two alternative constraints schemas with respect to the mathematical representations. The FMC_AxialSymmetric_Revolve is linked through the "hasMathRepres" with the revolve mathematical representation. Revolve representation requires the sketch to be SMC_Closed and SMC_Sided. FMC_AxialSymmetric_Revolve requires, additionally, via the "has Angle" object property the revolution angle to be 360 degrees. The FMC_AxialSymmetric_Extrude is associated with the extrude mathematical representation, which includes to the constraint schema the SMC_Closed. The FMC_AxialSymmetric_Extrude is further linked, through the "hasSMC" object property, with SMC_CenterSymmetric. SMC_CenterSymmetric indicates a cyclic profile to be centered about the origin and establishes the generation of an axial extruded model.

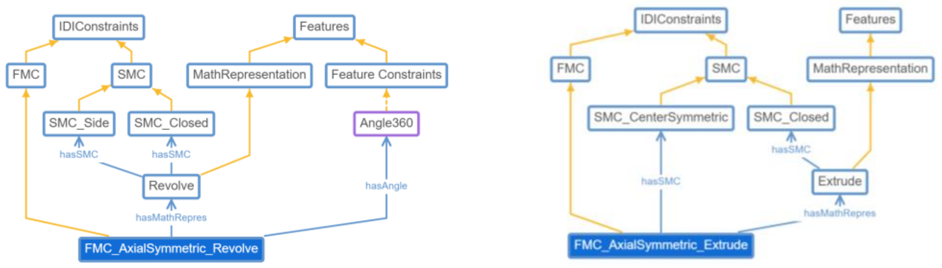


Figure 6: The constraint schemas that establish an axial symmetry design intent.

The "Full Symmetry" design intent, FIR_Symmetric (Figure 7), is linked with two feature meta-constraints, the FMC_Symmetric_Extrude and FMC_Symmetric_Revolve. These FMCs correspond to two alternative constraints schemas that establish design intent. FMC_Symmetric_Extrude is a combined meta-constraint and is further linked, via "hasFMC" property, with the FMC_SPSymmetric meta-constraint. FMC_SPSymmetric establishes the symmetry about the sketch plane. It is associated with the extrude mathematical representation and has the "hasDepth" property set to "Symmetric". In order for the symmetry about the two other planes to be ensured, the FMC_Symmetric_Extrude includes, via "hasSMC" property, the SMC_XYSymmetric meta-constraint, which indicates the sketch to be symmetric about the origin. FMC_Symmetric_Revolve is also a combined meta-constraint and is linked, via the "hasFMC" property, with FMC_AxialSymmetric_Revolve. FMC_AxialSymmetric_Revolve along with the SMC_(X/Y)Symmetric meta-constraint establish symmetry about the origin, where the SMC_(X/Y)Symmetric meta-constraint refers to a sketch that is symmetric either about X or Y axis.

5 EXAMPLE

During the MCAD parametric design phase, a primary challenge lies in developing a constraining strategy that captures the geometric design intent of the model. This process necessitates the designer's comprehension of the design intent conveyed by each feature or sketch and the identification of an appropriate constraining schema to establish them. The IDI Ontology aids in formulating a constraining strategy that embodies the desired design intent across the three design levels. We demonstrate its effectiveness through two examples.

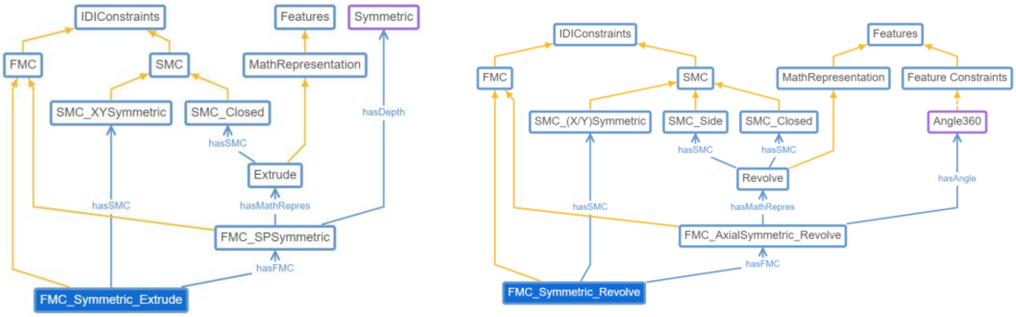


Figure 7: The constraint schemas that establish a plane symmetry design intent.

5.1 Example: Model with Axial Symmetry Design Intent

The example in Figure 8 (a) presents a mechanical part with three features: a revolve (base feature), and two extrude cuts (pocket feature and through hole). The base feature and pocket are coaxial features so the first query in the IDI ontology framework will indicate the design of two axial symmetric features (FDI: AxisSymmetric and MMC: SameAxes) (Figure 8(b)). This design intent is represented by FIR_AxialSymmetric intention regularity and can be established with a constraining schema indicated by FMC_AxialSymmetric_Revolve and/or FMC_AxialSymmetric_Extrude (Figure 8(c)). The designer selects for the base feature the FMC_AxialSymmetric_Revolve, which requires the introduction of 2 sketch meta-constraints and a feature constraint, so that the sketch profile is closed, sided in at least two adjacent quadrants and the angle of revolution equals 360°. For the pocket feature the designer selects the FMC_AxialSymmetric_Extrude, which requires the introduction of the sketch meta-constraint SMC_CenterSymmetric. Because the pocket feature is also a blind hole, the designer additionally queries for a constraining schema that establishes the blind hole intention regularity (FIR_BlindHole). This constraint schema is indicated by meta-constraints FMC_BlindHole_Extrude and/or FMC_BlindHole_Revolve. The designer selects the FMC_BlindHole_Extrude (Figure 8(d)), which requires the introduction of sketch meta-constraints SMC_Closed and SMC_FaceInnerLoop and the depth value of the extrusion. In a similar fashion, a constraint schema will be provided for the design of the third feature.

5.2 Example: Model with Sketch Plane Symmetry and Centrality Design Intent

The example in Figure 9 (a) presents a mechanical part with eight (8) extruded features: the base feature, the left and right stands which include respectively two holes, two ribs, and a cylindrical protrusion with a cylindrical coaxial cylindrical hole. The base feature, the two stands and the ribs are symmetric about a single plane. The first query in the IDI ontology framework will indicate the design of two sketch plane symmetric features or the design of a sketch plane symmetric feature and one axis symmetric (FDI: Sketch Plane Symmetry, MIR: SinglePlaneSymmetric and MMC: SPAttached) (Fig. 9(b)). The designer selects the design of sketch plane symmetric features for these three features. This design intent is represented by FIR_SPSymmetric intention regularity and can be established with a constraining schema indicated by FMC_SPSymmetric (Figure 9(c)). FMC_SPSymmetric requires the introduction of SMC_Closed meta-constraint and for the definition of the feature indicates for the depth value the “Symmetric” attribute.

The cylindrical protrusion is centered to the upper face of the base feature. The query for the creation of a featured with “FDI: Centered on a Face” employs the FIR_OnFaceCentered intention regularity, which can be established with a constraining schema indicated by FMC_OnFaceCentered (Figure 10 (a)). Alongside, the cylindrical protrusion is coaxial with a cylindrical hole.

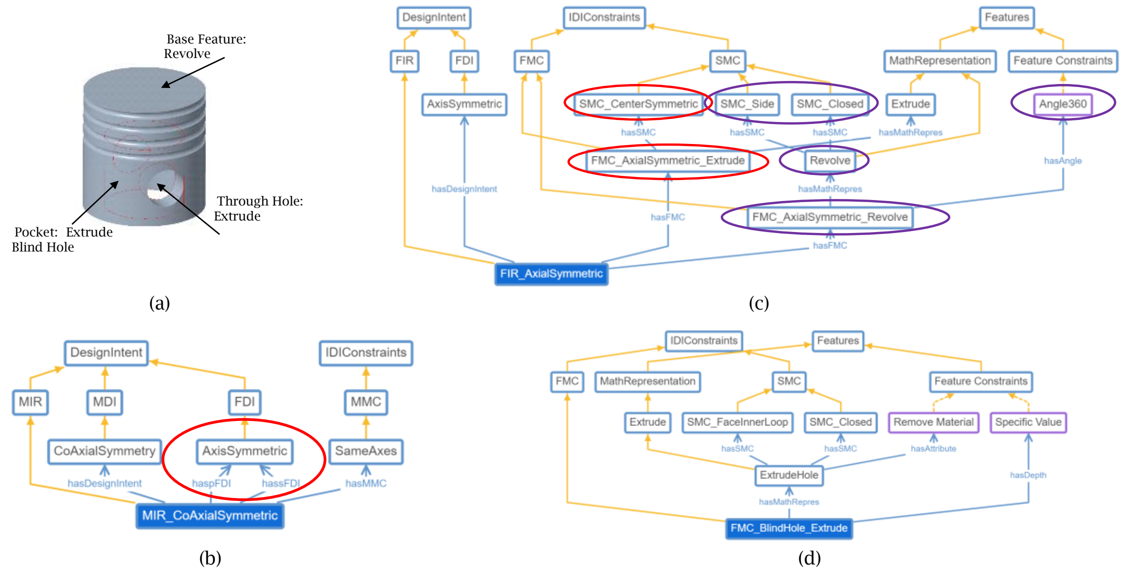


Figure 8: (a) For a given 3D model and a specific design intent, (b-d) the IDI Ontology proposes a set of meta-constraints / constraining schemas.

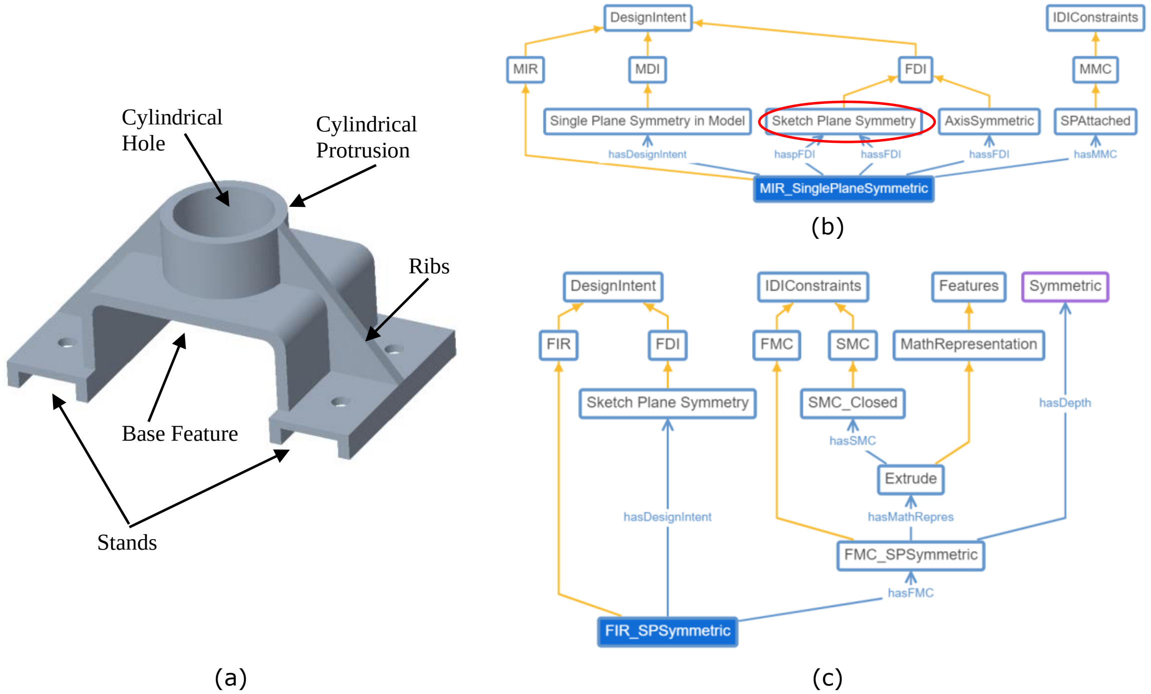
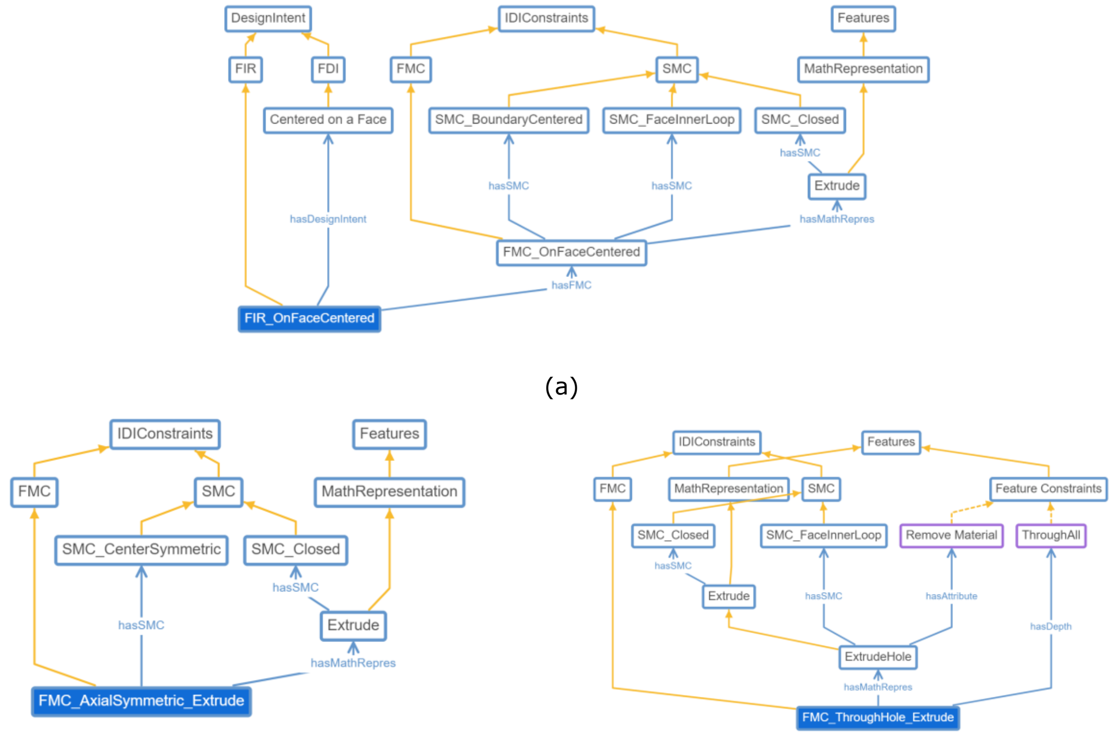


Figure 9: (a) For a given 3D model and "Single Plane Symmetry" design intent, (b-c) the constraint schemas that are indicated for the design of the three symmetric features.

For this design intent the designer queries for the creation of two coaxial symmetric features and the creation of a through hole. The corresponding constraint schemas that are compatible with the design intent for the specific model are given by FMC_AxialSymmetric_Extrude (Figure 10 (b)) and FMC_ThroughHole_Extrude meta-constraints (Figure 10 (c)). The constraint schema that is designated by IDI Ontology for the creation of the centered cylindrical protrusion is SMC_BoundaryCentered, SMC_FaceInnerLoop and SMC_Closed. This schema establishes the creation of a closed sketch that is centered to the boundary of a selected face and is constraint to always be placed inside of it. The constraint schema that is designated for the cylindrical hole includes the SMC_Closed, SMC_CenteredSymmetric and SMC_FaceInnerLoop meta-constraints. These meta-constraints establish the creation of a closed cyclic sketch where its center coincides with the center of cylindrical protrusion. The sketch will also define an internal loop to the face of cylindrical protrusion, a constraint that along with the through all attribute establishes the creation and preservation of the through hole. In a similar fashion, a constraint schema will be provided for the design of the small holes that lie on the two stands.



(b) (c)

Figure 10: The constraint schemas that are indicated for the design of (a) the cylindrical protrusion and (b-c) the coaxial symmetric through hole.

6 CONCLUSIONS

The IDI Ontology is designed to examine the underlying structure of IDI Architecture. It incorporates different design intents, expressed as intention regularities and various constraint schemas, expressed as meta-constraints, that play a crucial role in establishing intention regularities. By exploring these connections, the IDI Ontology provides valuable insights into how designers can effectively leverage the power of constraint-based design. In essence, the IDI Ontology serves as a comprehensive framework for modeling and analyzing the complex design processes that underpin modern industrial design.

Each design decision including the mathematical representation of a feature and the feature level standard constraints, is considered as an aspect of design rationale. Under this scope, the key classes of the ontology are multiple design intents that are commonly met at MCAD models and constraining schemas. The IDI ontology efficiently associates constraints with design intentions and by this manages to represent the design intent of a model at the three design levels. Features and sketches do not individually convey their semantic meaning, but they are considered as parts of an integrated structure where the semantic meaning of a feature is independently described and attached to it.

The IDI Ontology is an efficient tool towards the design of a constraining strategy for MCAD models. It can be effectively used by both novice and expert designers. As a framework it provides designers with proper constraining schemas that establish pre-defined design intents. Moreover, it indicates alternative constraining schemas, mainly in terms of different mathematical representations, that can establish the same design intent. The IDI Ontological framework's distinct features make it an effective tool for designing constraint strategies and enhancing CAD education. This framework serves the two main research areas of parametric modeling with design intent, demonstrating its versatility and utility.

The proposed paper focused on two groups of design intents: holes and axial/full symmetry. Future work on IDI Ontology will focus on the definition of additional meta-constraint and intention regularities pairs with the purpose to capture a large variety of design intentions at geometric and functionality levels, such as additional mathematical representations and engineering features including chamfers, rounds, shells, patterns. The proposed ontological framework can be further expanded with the aim to capture the design intent of a 3D model while it is created (bottom-up approach).

7 ANKNOWLEDGMENT

This research has been financially supported by the Initiative "Regional Excellence", under the Operational Program "Competitiveness, Entrepreneurship & Innovation" (EPAnEK), which is co-funded by Greece and European Union (MIS 5047046).

REFERENCES

[1] Ahmed, F.; Han, S.: Interoperability of product and manufacturing information using ontology, Concurrent Engineering Research and Applications, 23(3), 2015, 265-278. https://doi.org/10.1177/1063293X15590462

[2] Anderl, R.; Mendgen, R.: Modelling with constraints: theoretical foundation and application, Computer-Aided Design, 28(3), 1996, 155-168. https://doi.org/10.1016/0010-4485(95)00023-2

[3] Anderl, R.; Mendgen, R.: Analyzing and Optimizing Constraint-Structures in Complex Parametric CAD Models, In: Brüderlin B, Roller D, editors Geometric Constraint Solving and Applications, Berlin, Heidelberg, Springer, 1998, 58–81. https://doi.org/10.1007/978-3-642-58898-3_4.

[4] Aranburu, A.; Justel, D.; Contero, M.; Camba, J.D.: Geometric Variability in Parametric 3D Models: Implications for Engineering Design, Procedia CIRP, 109, 2022, 383-388. https://doi.org/10.1016/j.procir.2022.05.266

[5] Barbero, B.R.; Pedrosa, C.M.; Samperio, R.Z.: Learning CAD at university through summaries of the rules of design intent, International Journal of Technology and Design Education,27, 2017, 481–98. https://doi.org/10.1007/s10798-016-9358-z

[6] Barbero, B.R.; Pedrosa, K.M.; Peña,m G.K.: The importance of adaptive expertise in CAD learning: maintaining design intent, Journal of Engineering Design, 29(10), 2018, 569-595. http://doi.org/10.1080/09544828.2018.1519183

[7] Bettig, B.; Shah, J.: Derivation of a standard set of geometric constraints for parametric modeling and data exchange, Computer-Aided Design, 33(1), 2001, 17-33. https://doi.org/10.1016/S0010-4485(00)00058-0

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

[9] Camba, J.D.; Cosin, A.; Contero, M.: An Evaluation of Formal Strategies to Create Stable and Reusable Parametric Feature-Based 3D Models, Proceedings of the ASME 2014, International Mechanical Engineering Congress and Exposition, Volume 11, Montreal, Quebec, Canada, 2014. https://doi.org/10.1115/IMECE2014-37859

[10] Chenga, Y.; Heb, F.; Lv, C.; Cai, W.: On the role of generating textual description for design intent communication in feature-based 3D collaborative design, Advanced Engineering Informatics, 39, 2019, 331-346. https://doi.org/10.1016/j.aei.2019.02.003

[11] Company, P.; Naya, F.; Contero, M.; Camba, D.-J.: On the Role of Geometric Constraints to Support Design Intent Communication and Model Reusability, Computer-Aided Design and Applications, 17(1), 2020, 61-76. https://doi.org/10.14733/cadaps.2020.61-76

[12] Contero, M.; Company, P.; Salvador-Herranz, G.: Testing Rubrics for Assessment of Quality in CAD Modelling, Research in Engineering Education Symposium, Kuala Lumpur, Malaysia: 2013, 107–12.

[13] Contero, M.; Naya, F.; Pérez-López, D.; Company, P.; Camba, D.-J.: A Study on Sampling Strategies to Determine the Variability of Parametric History-Based 3D CAD Models, In Proc. of the International Mechanical Engineering Congress and Exposition, 2018. https://doi.org/10.1115/IMECE2018-87404

[14] Ganeshram, I.-R.; Mills, J.-J.: Design Intent in 2D CAD: Definition and Survey, Computer-Aided Design and Applications, 3(1-4), 2006, 259-267. https://doi.org/10.1080/ 16864360.2006.10738463

[15] Garikano, X.; Garmendia, M.; Manso, A.P.; Solaberrieta, E.: Strategic knowledge-based approach for CAD modelling learning, International Journal of Technology and Design Education, 29, 2019, 947–959. https://doi.org/10.1007/s10798-018-9472-1

[16] González-Lluch, C.; Company, P.; Contero, M.; Pérez-López, D.; Camba, D.-J.: On the effects of the fix geometric constraint in 2D profiles on the reusability of parametric 3D CAD models, International Journal of Technology and Design Education, 29(4), 2019, 821–841. https://doi.org/10.1007/s10798-018-9458-z

[17] Johnson, M.-D.; Diwakaran, R.P.: An educational exercise examining the role of model attributes on the creation and alteration of CAD models, Computers & Education, 57(2), 2011, 1749-1761. https://doi.org/10.1016/j.compedu.2011.03.018

[18] Johnson, M.; Diwakaran, R.: Assessing the Effect of Incentive on Computer-Aided Design Intent. In the Proc. of the ASME Design Engineering Technical Conference, 2019, 523-532. https://doi.org/ 10.1115/DETC2009-86644

[19] Kyratzi, S.; Azariadis, P.: A Constraint-based Framework to Recognize Design Intent during Sketching in Parametric Environments, Computer-Aided Design and Applications, 18, 2021, 545–60. https://doi.org/10.14733/cadaps.2021.545-560

[20] Kyratzi, S.; Azariadis, P.: Integrated Design Intent of 3D Parametric Models, Computer-Aided Design, 146, 2022, 103198. https://doi.org/10.1016/j.cad.2022.103198

[21] Ma, Q.; Song, L.; Xue, P.; Xie, D.; Zhou, M.; Shi, J.: A CAD Model Retrieval System Based on Design Intent, In: Tan, J., Gao, F., Xiang, C. (eds) Advances in Mechanical Design, Mechanisms and Machine Science, vol 55, Springer, Singapore, 2017. https://doi.org/10.1007/978-981-10-6553-8_46

[22] Mandorli, F.; Otto, H.E.; Raffaeli, R.: Explicit 3D functional dimensioning to support design intent representation and robust model alteration, Computer-Aided Design and Applications, 13, 2006, 108–23. https://doi.org/10.1080/16864360.2015.1059201.

[23] Mandorli, F.; Borgo, S.; Wiejak, P.: From form features to semantic features in existing MCAD: An ontological approach, Advanced Engineering Informatics, 44, 2020, 101088. https://doi.org/10.1016/j.aei.2020.101088
[24] Otey, J.; Company, P.; Contero, M.; Camba, J.D.: Revisiting the Design Intent Concept in the Context of Mechanical CAD Education, Computer-Aided Design and Applications, 15(1), 2018, 47–60. https://doi.org/10.1080/16864360.2017.1353733
[25] Otto H.-E.; Mandorli, F.: A Framework to Support 3D Explicit Modeling Education and Practice, Computer-Aided Design and Applications, 12(1), 2015, 104-117. https://doi.org/10.1080/ 16864360.2014.949581
[26] Pauwels, P.; Krijnen, T.; Terkaj, W.; Beetz J.: Enhancing the ifcOWL ontology with an alternative representation for geometric data, Automation in Construction, 80, 2017, 77-94. http://dx.doi.org/10.1016/j.autcon.2017.03.001
[27] Qin, F.; Gao, S.; Yang, X.; Li, M.; Bai, J.: An ontology-based semantic retrieval approach for heterogeneous 3D CAD models, Advanced Engineering Informatics, 30, 2016, 751-768. http://dx.doi.org/10.1016/j.aei.2016.10.001
[28] Rynne, A.; Gaughran, W.: Cognitive modeling strategies for optimum design intent in parametric modeling, Computers in Education Journal, 18, 2008. 55–68. https://doi.org/10.18260/1-2--2651.
[29] Salehi, V.; McMahon, C.: Action Research into the use of parametric associative CAD systems in an industrial context, Proceedings of ICED 09, the 17th International Conference on Engineering Design, Vol. 5, Design Methods and Tools (pt. 1), Palo Alto, CA, USA, 2009, 133-144.
[30] Sanfilippo, E.M.; Borgo, S.: What are features? An ontology-based review of the literature, Computer-Aided Design, 80, 2016, 9-18. https://doi.org/10.1016/j.cad.2016.07.001
[31] Sanfilippo, E.M.: Feature-based product modelling: an ontological approach, International Journal of Computer Integrated Manufacturing, 31(11), 2018, 1097-1110. https://doi.org/10.1080/ 0951192X.2018.1497814
[32] Yang, J.; Han, S.: Repairing CAD Model Errors Based on the Design History, Computer-Aided Design 38 (6), 2006, 627–640. https://doi.org/10.1016/j.cad.2006.02.007
[33] Yin, C.G.; Ma, Y.S.: Parametric feature constraint modeling and mapping in product development, Advanced Engineering Informatics, 26 (3), 2012, 539-552. https://doi.org/10.1016/j.aei.2012.02.010