Title:

A Constraint-based Framework to Recognize Design Intent during Sketching in Parametric Modeling Environments

Authors:
Sofia Kyratzi, skiratzi@aegean.gr, University of the Aegean
Philip Azariadis, azar@aegean.gr, University of the Aegean

Keywords:
Design Intent, Sketching, Parametric Modeling, Constraints

DOI: 10.14733/cadconfP.2020.253-257

Introduction:

Modern parametric CAD software provide the designers with tools to create 3D models that convey their objectives about the geometry, functionality, engineering and manufacturing of a product. The objectives are incorporated into the model's geometry with the use of parameters and constraints [8]. A model is considered efficient and robust when it transfers its design knowledge and it reacts in a predictable way to the changes of the parameters without being rendered into inconsistency [5-6]. The purpose or underlying rationale behind an object [14] and the design decisions that are taken during the development process are summarized under the concept of "Design Intent" [9]. The Design Intent concept admits multiple definitions (see [9],[16] and the references therein) that all urge to describe a notion which associates the modeling decisions about the geometry of an object with a higher level of engineering and manufacturing information that is related to shape [15].

Ganeshram & Mills [9] underly the importance of an intent-based design approach for the creation of a robust, reusable, modifiable 3D model. Since there is no unique modeling or dimensioning scheme for a part [2], it relies mostly on the designer's skills to select the most appropriate modeling strategy and employ these parameters and constraints that best capture the design intent. Several works [4],[12-13],[16-17] acknowledge this fact and focus on the study and development of strategies and approaches that improve the expression of design intent into CAD models. All of them emphasize that CAD education should follow a more strategic knowledge schema together with the declarative knowledge of the software tools. In parallel and in an effort to improve design intent representation in 3D models, numerous researchers [1-3],[7-8],[10-11],[15] study and analyze the constraints that are used in CAD software and their influence in design intent communication. Although current commercial CAD software usually employ "automatic" constraint detectors, in a recent research work, Company et al., [7] conclude that there is a need for intelligent parametric editors which can support the constraining tasks performed in parametric CAD environments.

Extending the aforementioned research, in this paper we propose a mechanism to enhance parametric constraint editors, which can be utilized to automatically recognize design intentions (such as symmetry, centrality etc.) during the sketching process in parametric CAD environments. We first study the most common types of constraints in CAD software and propose four new classifications to characterize the way they influence design intent. We identify design intentions in numerous of mechanical and industrial products and we introduce the term "intention regularities" to describe them. Then, we study three popular commercial CAD software (Creo Parametric, Inventor, and SolidWorks) in order to determine to what extend they assist their users to design on the base of intention regularities. Finally, we define a set of meta-constraints, which are based on the intention regularities and standard CAD constraints, and we propose a rule-based method for the implementation of meta-constraints in a sketching CAD environment.

2D Constraints and Design Intent in Sketching CAD Environment:

In parametric modelling environments, designers use sketch entities (such as lines, curves, arcs etc.) to define the geometry and the topology of a 2D profile. Parameters are employed to determine the size, position, and orientation of a sketch, and constraints set the relationships for the parameters and the sketch entities [1]. Every design decision conveys the intention of the designer. Although the designer recognizes his/her own design intent in the produced sketches, a third party, human or software, has to interpret the underlying design intent through the employed constraints.

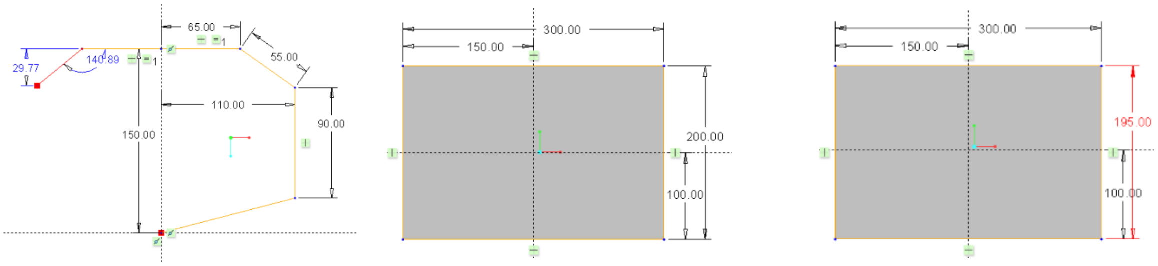
Most of the CAD software offer a set of constraints that are classified into four standard types, i.e., dimensional, geometric, ground and algebraic constraints [2]. Dimensional constraints are used to define the size of the profile or the distance between two entities. Geometric constraints impose non-algebraic relationships between sketch entities. This type usually includes the tangent, parallel, coincident, horizontal, vertical, equal, symmetric, and perpendicular constraint. Ground constraints orient and position the sketch with respect to the global coordinate system. Algebraic constraints impose restrictions on dimensional or ground constraints in the form of mathematical equations.

In this work, we consider that all constraints are explicitly defined. We propose four new classifications of constraints with respect to the way they convey design intent: (a) Automatic/ manual constraints: Automatic are the constraints that can be either applied by the designer or automatically applied by the CAD system. Manual are the constraints that are only applied by the designer; (b) Autonomous/associative constraints: Autonomous are the constraints that are applied to one sketch entity and associative are the constraints that relate two or more entities or other constraints; (c) Direct / Lateral constraints: A constraint is characterized as "direct" when the design intent is straightforwardly expressed by the semantic meaning of the constraint, such as parallelism or tangency. Lateral constraints if grouped together may indicate a design intent that is not directly expressed (e.g., multiple vertical constraints indicate parallel sketch entities), and (d) Strict / soft constraints: Strict constraints are explicitly defined by a designer or the CAD software to express a certain design intention, e.g., tangent constraint explicitly indicates that two entities should be tangent. Soft constraints do not express a design intent.

Considering the geometric, topologic and semantic meaning of each constraint, we analyze the four standard constraint types in terms of the introduced classifications. Dimensional constraints are in general manual, autonomous constraints with soft intention. They become associative, lateral and strict constraints when there are algebraic equations that correlate them. Algebraic constraints are manual, associative, strict constraints that express an implicit intention. With regard to the commercial software, all geometric constraints are automatic (except of symmetric constraint), direct and strict. But a substantive classification of the geometric constraints depends on the constraining schema of each sketch. For example, a horizontal constraint directly and strictly expresses an "horizontal line" intention, but when combined with horizontal and vertical constraints it is considered as lateral and strict constraint that implies "parallel", "equal", or "perpendicular" intentions. Finally, ground constraints are classified according to the constraint type that express them.

Intention Regularities:

We introduce the term "intention regularities" to describe the geometric or topologic patterns (e.g., symmetry, orthogonality etc.) that appear in engineering sketches and can be recognized as design intentions. In this research work, we focus on the intention regularities of "partial symmetry", "centered sketch", "side sketch", and "closed loop". "Partial symmetry" refers to the symmetry between two or more sketch entities around an axis, "centered sketch" is defined with respect to the X or/and Y axes, "side sketch" refers to a sketch that is placed on one quadrant of the projected planes, and "closed loop" is a sketch where its entities form a closed, non-self-intersecting loop that defines the profile of a solid model. We used the latest versions of the aforementioned CAD software to test whether they can recognize the above four intention regularities. Due to limited space, we use Creo Parametric to exhibit the results.

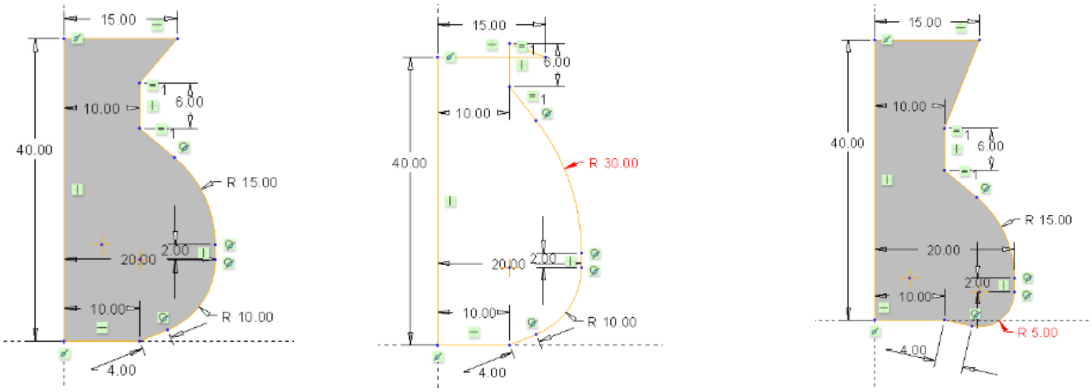


(a) (b) (c)

Fig. 1: (a) None of the tested CAD software understands the intention of the designer to create a symmetric oblique line, (b) The designer constrains the sketch so to be centered without using algebraic constraints, and (c) None of the software is able to interpret this intention.

Case Studies and Results

For the case of partial symmetry intention regularity, we study whether the CAD software recognize the designer's intention and assist him/her during sketching to create symmetric entities without using a mirror/symmetric operation (which requires multiple design steps). For the oblique line in Fig. 1(a), none of the above software suggest a symmetric constraint, although the designer's selection of equal lengths around the Y-axis implies such a case. The case of the "centered sketch" intention regularity (Fig. 1(b)) can be tested only when a sketch is fully constrained with strong constraints. In this case, the user constrains the sketch in such a way that implies a centered sketch intention. The fact that the implied intention is not maintained after a small modification of vertical dimensional constraint (Fig. 1(c)) indicates that none of the tested software associates the constraints and interprets the intention regularity. The "side sketch" and "closed loop" intention regularities can be recognized after the sketch is completed (Fig. 2(a)). Our case studies show that the CAD software do not consider the constraints imposed by the topology and geometry of the sketch. Neither they compute valid values [11] for the variable constraints (Fig. 2(b)) to maintain the closed loop intention regularity or propose appropriate constraints to capture the side sketch intention regularity (Fig. 2(c)).



(a) (b) (c)

Fig. 2: (a) The constrained sketch indicates a "close loop" and a "side sketch" intention. None of the CAD software recognize these intentions and allows for parametric values that destroy the (b) the close loop and (c) the side sketch.

Our case studies showed that the employed CAD software face constraints as independent entities which are not implicitly related to each other. Autonomous constraints are associated only if their relations are provided with algebraic or geometric constraints. To our conclusion, even when associative, direct, and strict constraints are used, CAD software do not include a mechanism to interpret intention regularities.

A Rule-based Framework to Capture Design Intent:

In this section, we propose a framework to capture the above four intention regularities. The proposed framework follows an “integrated design intent” approach. The term “integrated design intent” is used to describe the design intent of a sketch as this emerges from all constraints’ categories and from the geometric and topologic information that are conveyed in it. For its implementation we introduce the term “meta-constraint”. A meta-constraint is a constraint object that implies an intention regularity. It is comprised by a group of standard constraints (that are called linked constraints), sketch entities or other meta-constraints. Different arrangement and association of constraints and entities may represent the same meta-constraint. Meta-constraints can be applied over the standard constraints, in an upper layer forming an object-oriented structure. They are able to capture and re-apply a predefined design intent and gradually establish the integrated design intent of a sketch. They are automatic, associative constraints of direct and strict intention. According to the proposed framework, these attributes grant the meta-constraints and its linked standard constraints higher priority. When a designer deletes a meta-constraint, the linked constraints are also removed from the sketch (and the captured design intent) and the sketch remains with the constraint solution of the CAD software. Each meta-constraint admits a weight during sketching that indicates its plausibility. According to its weight, a meta-constraint may be provided as an automatic constraint, as a prompt choice, or as manual selection together with the standard constraint types. This research work defines four meta-constraints, in correspondence to the above four intention regularities. Bellow we briefly describe each meta-constraint and the proposed geometric or heuristic rules for the evaluation and implementation of them.

“Symmetric Entities” meta-constraint refers to the partial symmetry intention regularity. It is defined from at least two sketch entities and the X/Y axis and is linked with the equal, coincident, parallel, horizontal and vertical constraints. A background procedure should check for symmetry (with respect to X or Y axis) between the existing sketch entities and each new added entity. If symmetric intention is detected and/or one of the linked constraints is selected for the new entity, the meta-constraint admits a non-zero weight value and it appears as a prompt choice to the designer. The confirmation of this choice admits the meta-constraint with a height weight value and the meta-constraint evaluation continues.

The “Side Sketch” meta-constraint corresponds to the side sketch intention regularity. It is determined by the sketch geometry and is linked to the ground constraints of the sketch. It admits a height weight value if all the sketch entities are located on one quadrant of the sketching plane. If the meta-constraint is confirmed by the designer, the ground constraints (dimensional or geometric) become explicit, strong and are linked to the meta-constraint group.

The “Closed Loop” meta-constraint corresponds to the closed loop intention regularity. It is related to the geometry and topology of the sketch entities and it is linked to all the constraints of the sketch. When the sketch drawing is completed and a closed loop is created, the meta-constraint admits its highest value and is applied automatically. All standard constraints are linked to the meta-constraint group. For its implementation, the CAD software must be upgraded with an algorithm that calculates the valid values of the dimensional constraints that retain the closed loop.

The “Axes Centered” meta-constraint refers to the centered sketch intention regularity. It is based on geometric entities that span between two quadrants of the sketching plane and is linked with the corresponding dimensional and ground constraints. The meta-constraint is expressed through the automatic generation of the appropriate algebraic constraints. The values of the linked constraints are checked for an algebraic correlation that implies an axis centrality. If the constraints become strong, the meta-constraint admits a high weight and is applied automatically. Otherwise, it prompts as a choice for the designer. In the case that the designer explicitly defines the proper algebraic constraint, the meta-constraint is automatically applied.

Conclusions:

The major advantage of the proposed framework is that it considers all sketch entities and constraints as interrelated elements that convey an integrated design intention. The intention regularities and meta-constraints establish a robust fundamental base for the design intent capturing and communication problem. A study on different engineering products can result in numerous intention

© 2020 CAD Solutions, LLC, http://www.cad-conference.net

regularities, each of which can be distinctly defined. In this framework each intention regularity can be explicitly selected or removed by a designer during sketching. The proper management of meta-constraints can efficiently designate an integrated design intent of an object and preserve it during model modifications.

References:

[1] 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

[2] Ault, H.: Using geometric constraints to capture design intent, Journal for Geometry and Graphics, 3(1), 1999, 39-45.

[3] 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

[4] 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

[5] Camba, J.-D.; Cosin, A.; Contero, M.: An Evaluation of Formal Strategies to Create Stable and Reusable Parametric Feature-Based 3D Models, In Proc. of ASME International Mechanical Engineering Congress and Exposition, 2014. https://doi.org/10.1115 /IMECE2014-37859

[6] Cheng, Z.; Ma, Y.: A functional feature modeling method, Advanced Engineering Informatics, 33, 2017, 1–15. https://doi.org/10.1016/j.aei.2017.04.003

[7] Company, P.; Naya, F.; Contero, M.; Camba, J.-D.: 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

[8] Contero, M.; Naya, F.; Pérez-López, D.; Company, P.; Camba, J.-D.: 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

[9] 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

[10] González-Lluch, C.; Company, P.; Contero, M.; Pérez-López, D.; Camba, J.-D.: 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

[11] Hoffmann, C.M.; Kimb, K.-J.: Towards valid parametric CAD models, Computer-Aided Design, 33(1), 2001, 81-90. https://doi.org/10.1016/S0010-4485(00)00073-7

[12] 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

[13] 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

[14] Lee, J.; Lai, K.: What's in design rationale, Human-Computer Interaction, 6(3-4), 1991, 251– 280. https://doi.org/10.1207/s15327051hci0603&4_3

[15] 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(1), 2016, 108-123. https://doi.org/10.1080/16864360.2015.1059201

[16] 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

[17] 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

© 2020 CAD Solutions, LLC, http://www.cad-conference.net