

Automation of drafting execution by schemes definitions and feature recognition

Roberto Raffaelli¹ , Paolo Cicconi²  and Michele Germani² 

¹eCampus Online University, Italy; ²Polytechnic University of Marche, Italy

ABSTRACT

Despite of the widespread of annotated 3D virtual models, it is still important to produce engineering drawings for a correct communication among the people of the design and manufacturing chain. Automatic drafting through meta drawing definitions, called Drafting Schemes, is proposed in this paper to save time when drawing becomes a repetitive task. The approach produces drafts with a predetermined content in terms of views arrangement and dimensioning strategy, and guarantees robust results to the variations of the input geometry. In fact, the geometrical entities referenced by the generated dimensions are retrieved through dedicated search algorithms based on recognition of geometric features of the input model.

The approach has been implemented and tested on a family of gear motors showing feasibility and good results in terms of drafting time savings.

KEYWORDS

Drafting; Design Automation;
Feature Recognition;
Knowledge Based Systems

1. Introduction

In the product design field, Design Automation (DA) is commonly intended as the possibility to obtain geometric shapes of parts and assemblies starting from a set of initial specifications by means of autonomous software [8]. DA systems generally enhance the capabilities of parametric feature based 3D CAD systems to alter the geometry and the arrangement of parts following a set of predefined options. The scope of DA applications is centered on product families [4] characterized by a good level of configurability or dimensional variability. Typical examples are gearboxes, electric motors, metal shelves, hydraulic cylinders [11].

Indeed, the term configuration refers to approaches and tools to identify valid sets of input parameters given a set of independent variables chosen by a user and a set of commercial and/or technical constraints [7]. Proposed approaches include rule-based and constraint-based systems while research and industrial applications have shown good potentialities of such systems [5]. The link of DA systems to configuration engines has been also investigated [9] as effective design tools.

Although DA systems claim the ability to generate both 3D geometry and technical documentation, the phase of generating robust dimensioned drafts of the configured product is often neglected. Drafting is the traditional way of representing a product from a technical point of view while 3D representations has shown

undisputed capabilities of facilitating design activities and building virtual mockups. However, drafts are still used as the official way to document a product, communicate to production, propose and share data about variants in catalogues.

In the case of large families of configurable products, the possibility to have the technical drawing of every required family member may become critical. Moreover, products usually experience continuous updating in the shape of the components, number and arrangement of parts and addition of new variants. These aspects require constant drawing updating and resources involved in repetitive and low profile drafting tasks.

In this context, this paper presents a general approach to automatically generate technical drawings of members of product families. Some studies have already been conducted in the literature some years ago aiming at paradigms to generate views and dimensions of generic 3D mechanical components [2,3]. In particular, automated procedures to obtain dimensioning in relative short time has been developed in the form of macros or coding activities such as the well none Visual LISP in the AutoCAD system [12]. Nowadays, modern CAD tools propose functionalities to support the drafting activities in 3D-2D associativity environments. It usually refers to view arrangement templates in order to speed-up the execution of similar drawings. Dimensions can be added to such templates and the original reference to the 3D

model changed with a new one when necessary. However, this usually implies a loss of many geometrical references which need to be manually recovered.

It is worth to mention also the increasing use of 3D annotations through PMI tools which includes also dimensioning. When drafts are created from these models, dimensions and other markups are reported in the views according to the 3D definition plane of the markup and the projection direction of the view.

Some 3D CAD systems provide automated 2D dimensioning plug-ins which are able of browsing the geometry of a part and instantiate dimensions on the basis of embedded search algorithms [1, 13]. However, the goal of such systems is to identify the dimensions of the main geometric features of a mechanical component seeking to cover the whole set of dimensions and locations. These approaches shows some limitations. The first one concern the fact that the added dimensions depend on the algorithms and it is not possible to determine the desired sizing information. In fact, the dimensions are decided by the system on the basis of the geometric features of the product. The second one concerns the applicability which is restricted to single components.

In this paper the focus is set on the definition of abstract Drafting Schemes (DS) which are valid for a certain product family, or a subset of it, and it is applied to the generic member in order to obtain views arrangement and dimensions according to a predefined application purpose. For instance, a drawing of a group could be produced for manufacturing reasons or to be inserted in a technical catalogue or in a spare parts manual. Each of these applications requires different representation approaches, different sets of dimensions and variability in the additional data, for instance tolerances, which need to be reported in the drafts.

Since the product to be drafted assumes different variants and evolves in the shape thanks to the work of the designers, entities to be dimensioned change and require adequate search strategies. Therefore, such entities are retrieved through feature recognition algorithms which seek to recognize aggregates of entities with a clear design meaning such as pockets, holes, fillets, etc . . . , from the whole B-Rep geometry. To this aim feature recognition approaches have been employed [6, 10] and adapted in some parts. In particular, customized feature recognition functionalities are provided and exposed as functions which can be chosen in the DS definitions.

The novelty of the approach is given by such combination of drafting automation procedures with feature recognition algorithms. The flexibility in the definition of the searching strategies of the geometric entities to be dimensioned makes the systems easily reconfigurable on various application fields. Thus, the final user to define its own dimensioning pattern based on abstract definitions

of the entities to be sized. In particular, such definitions can be propagated through the variability of the product family members showing robustness and effectiveness.

The paper is organized as follows. A brief background of engineering dimensioning is provided to narrow the scope of the work. Then, the approach is presented focusing on the Drafting Scheme definition, the geometrical entities search strategy and the proposed system architecture. A gear motor test case is then reported along with some details on the implemented system and the reached results.

2. Background and definitions

Technical drawings are diffused in the majority of the engineering fields such as civil, architectural, structural, electrical and mechanical. Technical drawings formalize design, manufacturing and assembling data. The principles of representation are defined by international standard such as the ISO 128, which sets the general and specific graphical representation of objects on technical drawings, such as the graphical layout and styles to be adopted.

In particular, mechanical engineering drawings convey information for manufacturing of parts and assemblies of components. Drawings are based on orthogonal projections views of the part and annotations. Among the others, dimensions are essential annotations and are expressed as lines, numeric values and attributes. Even if dimensions are mainly used in part drawings to specify the sizes and locations of the single geometric features, they can be also found in assembly drawings in order to highlight overall dimensions and characterize the external interfaces.

Several graphical layouts are possible to dimension an engineering drawing. In order to comply to standards, drafts should be clear and easy to read. The standards include different arrangements, for instance baseline and chain dimensions. However, it is not only a matter of stylistic choices, since different dimension patterns influences the associate tolerances.

The aim of a drafter is to identify the best solution among the standards to correctly represent the design intent, that is the functionality of the product being represented. This goal can be achieved if the definition of the dimensions follows predefined rules in order to ensure the respect of the original objectives and cannot be solely derived from the geometry itself.

2.1. Dimension types

Dimensions measure the relative position between two or more geometric entities in a planar domain. Geometric entities include edges, faces, points, silhouette borders

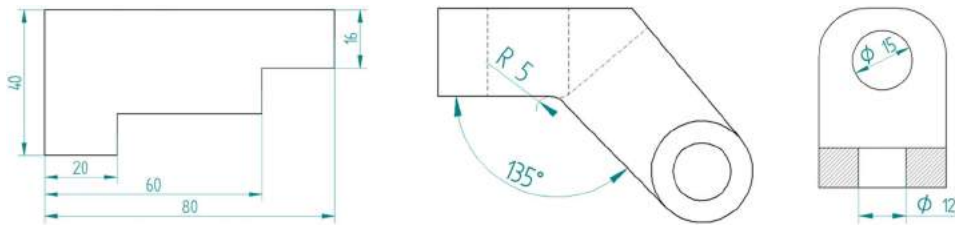


Figure 1. Types of dimensions considered in the paper: a) linear, b) angular and fillet, c) circular and diameter.

and also construction elements such as axes, center-marks and center-lines. Construction geometries are often prescribed by the standards or added on purpose for sizing aims.

In this work, four types of dimensions have been considered:

- **Linear dimension:** it is a distance, usually vertical or horizontal, between two geometric entities (Fig. 1(a)). In the mechanical field, linear sizing is expressed in millimeters.
- **Angular dimension:** it measures the angle between two geometric entities and it is specified in degrees (Fig. 1(b)).
- **Circular dimension:** it expresses the diameter of a cylinder or a circular element in general. The measuring line of the annotation is parallel to a diameter and lies in a plane normal to the cylinder axis (Fig. 1(c)).
- **Fillet dimension:** it is typically used for sizing the radius of a fillet or round entity (Fig. 1(b)).

3. Approach

The proposed approach splits in two main steps, namely an off-line phase and an on-line one. The first concerns the definition of a knowledge base in the form of Drafting Schemes which encapsulates the rules for the execution

of the drawings according to the characteristics of the product family and their final use (i.e. production, documentation, ...). A DS contains the rules to choose the views, the dimension types and positioning. A scheme applies to a set of similar 3D models usually corresponding to the variants of the same product family and is input in the off-line phase by a designer using a dedicated graphical interface.

The second phase refers to the actual, repetitive and autonomous on-line utilization of the system to realize dimensioned drafts.

3.1. Definition of the drafting scheme

The structure of a DS is reported in Fig. 2. It is an object-oriented data structure which acts as a prototype of the views structure and of the dimensions which are required to accomplish the desired drawings of a product.

Initially, a DS reports the list of sheets and views required by the draft. Attributes such as sheet sizes, units, view projection directions, scales are defined. Then, the list of dimensions belonging to each view appears. Each dimension is associated to presence rules which are to be satisfied for the dimensions to be created in the drawing. The most obvious and trivial rules regards the presence of the dimensioned part and the possibility to identify the target geometric entities. Other rules are connected

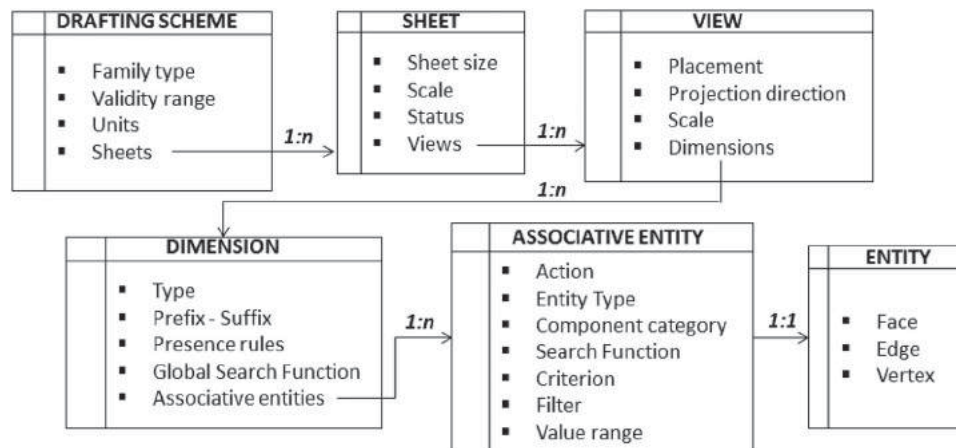


Figure 2. Structure and data contained by a Drafting Scheme.

to model attributes, for example the code. Rules helps to lower the number of required schemes since the same DS could be applicable to several variants of the same product family.

The most significant aspect of a scheme is the definition of the geometric entities (edges, faces, points, reference geometry) which are the target of the dimensions. In fact, the DS contains the criteria to search and identify the required entities through the whole product geometry by the help of feature recognition algorithms. In case of assembly models, the search domain is restricted to single parts which are identified by a *Part Category (PC)*. A category is an attribute of the components recoverable from the 3D model or a data management system. It could be the code itself or more likely a part class such as shaft, casing, screw.

The searching approach is based on multiple steps. Firstly, *Search Functions (SF)* are used to identify candidate entities, for instance the faces of a set of holes located in a certain part of the product and characterized by a certain diameter. *Adding Functions (AF)* are introduced to define construction entities derived from the 3D geometry. Then, a *Global Search Function (GSF)* is in charge of sorting the entities from the first step and selecting the ones actually needed to instantiate the dimension.

More specifically, in a DS a dimension is defined on three levels: *dimension data*, *associative entities*, *3D entities*, as follows:

- **Dimension data**, that is the set of information including the dimension type (angular, linear vertical, linear horizontal, etc.), additional annotations, such as

prefix, suffix, tolerances, and stylistic parameters like the text height. The *GSF* is also specified at this level, as the criteria to select the required geometric entities among the valid ones.

- **Associative entities** is a collection of proxy objects which link to an existing 2D entity (projected edge, projected face, silhouette curve, projected vertex) or to derivative geometries created in the model by the AFs. Normally two associative entities are enough to define a dimension, even if there are cases where more references are necessary for a complete definition. Examples are angular dimensions, symmetric twin dimensions or when a particular sizing direction is to be specified.
- **3D Entity** is the 3D geometric element which is projected in the drawing. Such elements are edges, faces, vertices of the B-Rep model transformed to 2D elements by the projecting algorithms of the CAD system. The associativity features of the modern CAD system ensure the link between the 3D and the projected 2D entity.

On the basis of these information, SFs are triggered as 3D entities search operators. In order to select the desired entities, SF can be called in cascade to progressively narrow the set of selected entities. For instance, the faces of the holes of a part could be selected as an initial step. Then, a second function picks the faces belonging to a certain portion of the model.

Table 1 report examples of SF. Some functions are general as *GetFaces* or *GetEdges*. They retrieve entities on the basis of filters such as directions, geometric types,

Table 1. Examples of *Search Functions* to recover 3D reference geometry.

Function Name	Description	Input parameters	Output entities
GetFaces	Generic function to search faces filtered by geometric type. Optionally, plane faces can be filtered by normal vector while revolved faces by axis. An additional filter on the location of the face relatively to the model bounding box is selectable.	Part category Geometric type Direction (opt.) Search limits (opt.)	Faces Edges of the faces
GetEdges	Generic function to search edges filtered by geometric type. Optionally, linear edges can be filtered by direction vector and arc edges by axis. An additional filter on the location of the edge relatively to the model bounding box is selectable.	Part category Geometric type Direction (opt.) Search limits (opt.)	Edges
GetHoles	Return hole features. Options can be specified on the size, the search limits, direction of the axis, diameter and depth of the hole	Part category Axis vector Search limits (opt.) Dimensional range (opt.)	Faces
GetPocket	Get the faces of a pocket	Part category Through pocket type (opt.)	Faces Edges of the faces
GetKeyway	Get the faces of a pocket shaped as a keyway	Part category	Faces Edges of the faces
GetChamfer	Get the faces of a chamfer feature	Part category Chamfer size range	Faces
GetFillet	Get the feature of a fillet feature	Part category Fillet size range	

dimensions. These functions are used to refine the entity search produced by another SF which operates on a feature recognition basis, such as *GetPocket* or *GetKeyway*. Among the possible filters, the search limits refer to the possibility to define a range of validity in relative terms toward the part bounding box. For instance, a certain dimension definition can require that the referred hole is located in the first half of the part along the x axis.

Other SFs are the ones based on Feature Recognition algorithms. An hybrid approach based on graph search and hint-based reasoning is here proposed [6]. The first refers to the graph formed by the nodes representing the faces connected with arcs, represented the edges, attributed with convexity information. Graph search techniques are used to find sub-graphs which are identifiable with a geometric feature. For instance, pockets are recognizable from star subgraphs, built around a bottom accessible face, and path subgraphs [10]. Hints refer to rules which seeks to identify a feature from the trace left by the manufacturing process. For instance, an hole is recognizable from the presence of a cluster of coaxial and closed connected faces.

The presence of the interacting features, that is features removing intersecting portions of material, makes the feature recognition process quite complicate. The combination of the two approaches has revealed to be the most promising solution. In case of the interaction of machining features, profile completing strategies are also necessary to recover the profile of a pocket whose boundary has been removed by successive features [10].

SFs implement such algorithms to recover the set of geometric elements which are the target of dimensioning: holes, pockets, fillets, chamfers, In particular, the output of a SF is the faces of the identified features filter by search limits or other geometrical constraints.

Table 2 reports examples of AFs. The construction entities such as centerlines, center marks and bolt circle lines are created in the dimensioning process and referred to construct the required dimensions. For example the cylindrical face of an hole is used to create an axis and position the hole itself.

Finally, the GSF are defined in the context of a specific dimension and makes the final selection of the required

geometric entities. In an assembly models GSF operates across the 3D entities extracted from the components of the whole model considering the relative orientation in the assembly model.

An example of GSF is the *GetExtremeEntities* which is used to add the overall dimension of an assembly. The function sorts the entities retrieved by the SFs for the single components of the assembly along the dimension direction and returns the two extreme ones.

Figure 3 reports an example of a procedure for retrieving hole centers and adding an horizontal dimension between the farthest ones. The examples involves the creation of the center marks and helps to highlight the difference between SF and GSF. The SF called "*GetHoles*" individuates the holes present in the plate. Optionally, a filter on the diameter could be added if necessary. Then, center marks are generated from the cylindrical faces of the holes. The GSF "*GetMaxDistance*" analyzes the position of each hole along the direction required by the dimension, i.e. horizontal, recovering the pair of the more distant holes. Finally the linear dimension reported in red is generated.

3.2. System workflow

Figure 4 outlines the workflow of a system based on the proposed approach. Two main domains are distinguished: the knowledge domain, i.e. the definition of the DS and the relative application rules, and the application domain related to the utilization phase of the system.

DSs are defined by a designer through a dedicated interface and stored by the system to be retrieved as new drafts are to be automatically generated. The input to this phase is given by the 3D geometric model of a product and the relative Bill of Material (BOM). The product could be either an assembly or a single part. Since the approach is oriented to families of alternatives, the model originates from interactive modelling activities or, more likely, from configuration modules of dedicated systems or PLM. The output is a dimensioned draft of the product represented by one or more orthogonal projection and the required annotations.

The drafts generation phase includes three application modules and the interaction with the CAD system

Table 2. Examples of *Adding Functions* to add derivative construction entities to the drawing.

Function Name	Description	Input parameters	Output entities
AddCenterMark	Add a center mark from an arc edge	Arc edge	Center mark
AddCenterLine	Add a center line from a symmetric entity or a pair of specular entities.	Symmetric entity or specular entities	Center line
AddBoltCircle	Add a bolt circle line through the centers of at least three holes	Hole centers	Bolt circle line

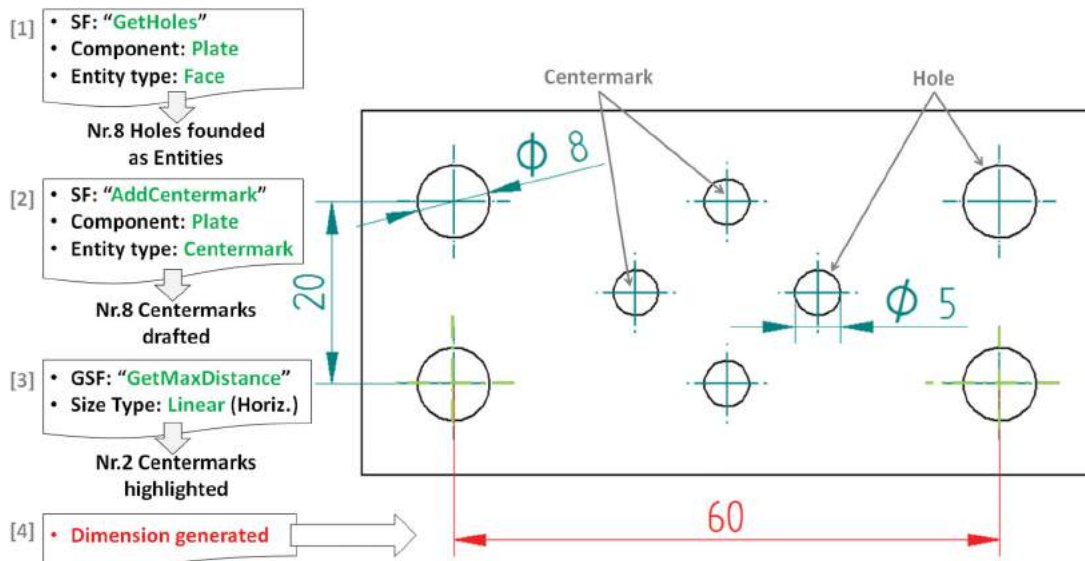


Figure 3. Steps in sorting geometrical entities while adding a horizontal distance between two holes.

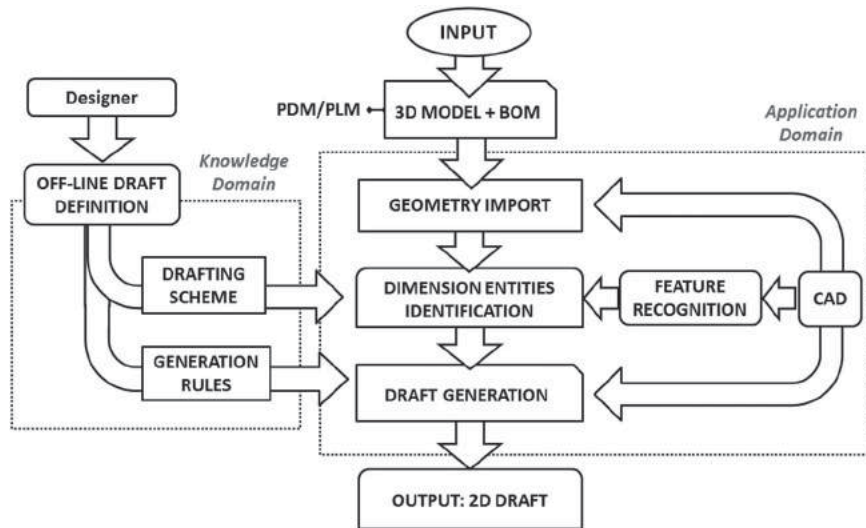


Figure 4. Workflow of the approach.

through the APIs to load the 3D geometry and to generate the 2D one. In particular, a first module, called *Geometry Import*, reads and analyzes the model geometry from the CAD and other additional attributes coming from the BOM, such as part codes, categories, material, description. The product geometry is read and converted in a B-Rep format which includes standardized geometric formulation for edges, faces and topology. This conversion step helps the successive entity search algorithms since the expected geometric formulations types are reduced. For instance, closed cylindrical face formulations as two halves are modified in a single cylindrical face. In this case face boundary loops are rearranged to this aim. Other functionalities includes the conversion of generic geometry representation in more specific ones.

For instance, first order NURBS faces are converted to planes.

The second module, namely *Dimension Entities Identification*, elaborates the geometry in order to find the entities to be sized following the Associative Entity list provided by the selected DS. In this phase, construction elements such as axes or a centre marks are added to the model and referenced if required.

The last module is the *Draft Generation*. It drives the autonomous execution of the technical drawing making it available in formats such as DXF or PDF. The sheets, views and dimensions to be drawn are provided by the selected Drafting Scheme depending on the presence of certain components or the spatial arrangement of some 3D parts according to the input 3D model.

4. Implementation of the approach

The proposed approach has been tested on the autonomous execution of technical drawings of a family of gear motor assemblies. The gear box can assume several configurations on the basis of the axes relative position, size, transmission ratio, number of reductions, type of flanges, mounting position, etc.. The drawings are generated for documenting the gear motor sizes and interfaces and are destined to machine designers.

4.1. Introduction to the test case

The experimental work has been conducted in collaboration with an important Italian manufacturer of gear motors, Bonfiglioli Riduttori Spa. A gear motor is a device used for reducing the rotational speed and increasing the torque in mechanical transmissions. Usually, the input power is supplied by an electric motor providing an high rotational speed which needs to be reduced. The gearing mechanism consists of spur or helical gears arranged to obtain the final gear ratio. Main components in a gear motor are: electric motor, gears, shafts, flanges and box (Fig. 5).

Dimensioned drafts of gear motors are required as documentation for customers. Machine and plant producers need drawings to understand the sizes and the interfaces of the product to evaluate the feasibility of its employment and installation.

In particular, the selected test case is focused on assemblies of a gear box family type “A” and the relative motor. The name “A” means that fast and slow shafts are

mounted with an angled position, in particular they are perpendicular each other. This class presents the possibility of different product variants with several geometric configurations. In fact, power ranges up to 150 kW with a range of gear ratios from 5.4 to 1715. In particular, the test case has been developed analyzing configurations named A60 and A30.

Figure 6 highlights the list of parameters which can produce geometric variants in the final model configuration. Many alternatives are possible from the combination of gear unit parameters such as input configuration, gear frame size, flange position, output flange size, number and type of output shafts. Actually, the motor unit is optional because the customer can decide to purchase only the gear box. In this case, the related technical drawing has to represent the input interface for mounting the motor unit in a second phase.

A drafting scheme has been initially formalized for the A60 model family. Three drawing views represent the product. A single DS has been sufficient to manage the several design options thanks to rules to activate and deactivate dimensions on the basis of code options and part presence.

A problem which has been faced concerns the absolute orientation of the input model. In fact, the scheme definition is conceived in a standard configuration while 3D models can vary their orientation. Therefore, an algorithm has been introduced to detected the model orientation on the basis of some fixed reference as the input and output shaft axes.

Figure 7. shows how the orientation is identified. Basically, the X-axis is given by the slow shaft. Since a shaft is

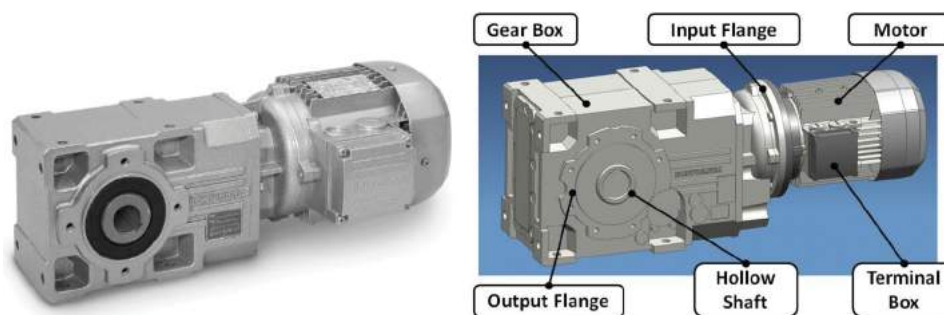


Figure 5. Example of a member of the family type “A” of gear motors analyzed as test case.

GEAR UNIT	▪ TYPE (A)	MOTOR	▪ MOTOR TYPE (M-compact, BN-IEC)
	▪ GEAR FRAME SIZE (05 ... 90)		▪ MOTOR SIZE (0B ... 5LA, 63A ... 280M)
	▪ REDUCTIONS (2, 3, 4)		▪ MOTOR MOUNTING (compact, B5)
	▪ VERSION (UH, UR, UD, US, UV, QF)		▪ TERMINAL BOX POSITION (W, N, E, S)
	▪ OUTPUT FLANGE SIZE (A, B, C) - optional		▪ BRAKE TYPE (FD, AFD; FA, BA)
	▪ FLANGE POSITION - optional		▪ BRAKE HAND RELEASE (R, RM)
	▪ INPUT CONFIGURATION (S, P, SK, SC, HS)		
	▪ MOUNTING POSITION (B3, B6, B7, B8, VA, VB)		

Figure 6. Parameters which determine the geometry variants for the gear motor Type A.

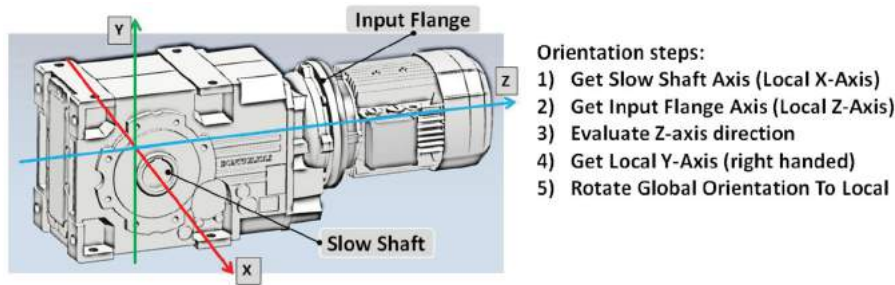


Figure 7. Steps in the determination of a standardized gear motor reference system.

an axisymmetric solid, its axis is defined by the majority of the revolution axes of its faces. The Z-axis is parallel to the axis of the input flange, i.e. the fast shaft. A plane is formed by the two axis centered in the middle of the slow axis. The orientation of the Y axis is finally determined considering that the motor axis lies in the positive half of the XZ plane. In this way, the drafting scheme describes the desired drawings in a standardized orientation and every model is transformed to this reference system.

4.2. System implementation

The proposed method has been implemented in the “bDrafter” tool, a Windows based application implemented on the .NET framework and the Siemens NX 9.0 CAD system. The software architecture is characterized by an object-oriented approach (Fig. 8) and is formed by a core, two Graphical User Interfaces (GUIs), a database of Drafting Schemes and connections to

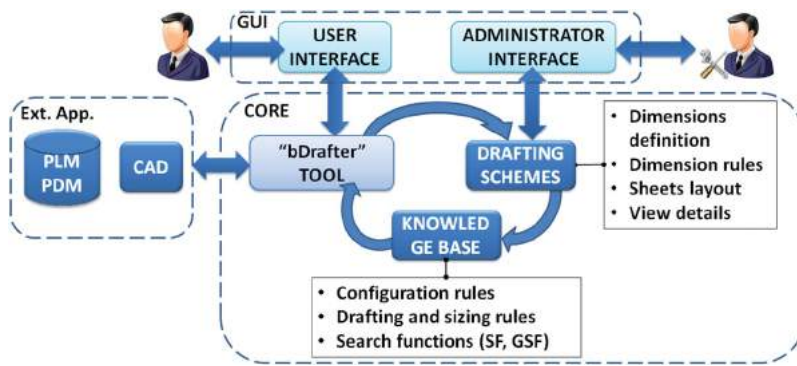


Figure 8. “bDrafter” software architecture.

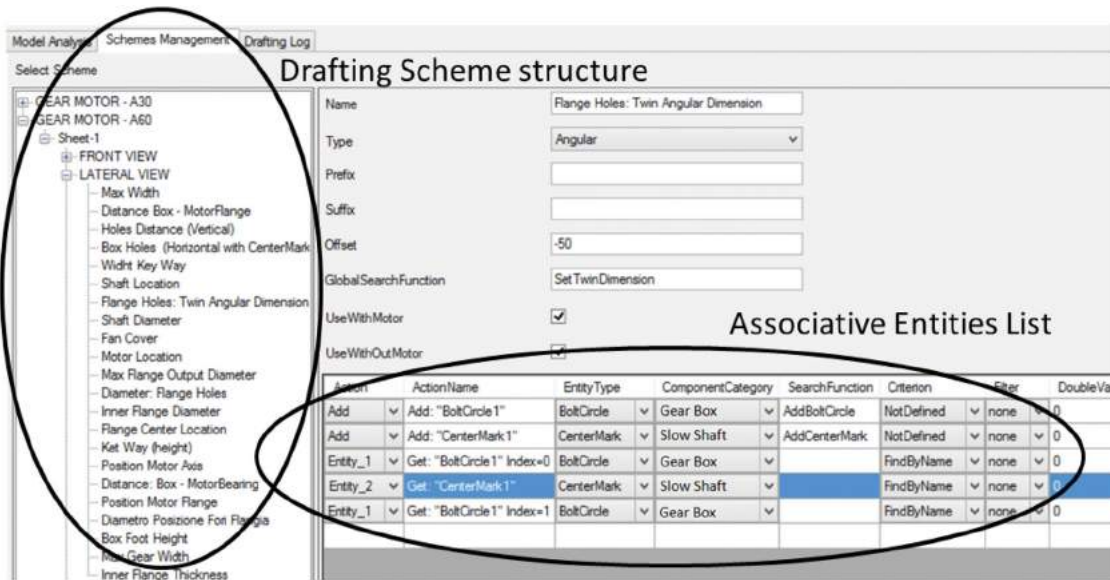


Figure 9. Administrator Interface of the “bDrafter” software.

external applications, namely CAD and PLM/PDM systems. The geometric kernel to represent the BRep model and the feature recognition algorithms has been provided as a library by a partner software company (Hyperlean Srl).

The administrator GUI is shown in Fig. 9. The knowledge base is formed by a collection of DSs, DS choosing rules, search functions (SF, GSF) and are input through the administration GUI. The administrator user can add

or edit drafting schemes according to the required content of the engineering drawings or new product families definitions. On the left hand side a tree represent the defined schemes, and the structure in terms of sheet, views and dimensions. On the right hand side it is possible to edit the data of each tree element.

In particular, a dimension is defined through some attributes and a table of association entities shown in Fig. 10. The administrator adds rows in order to search

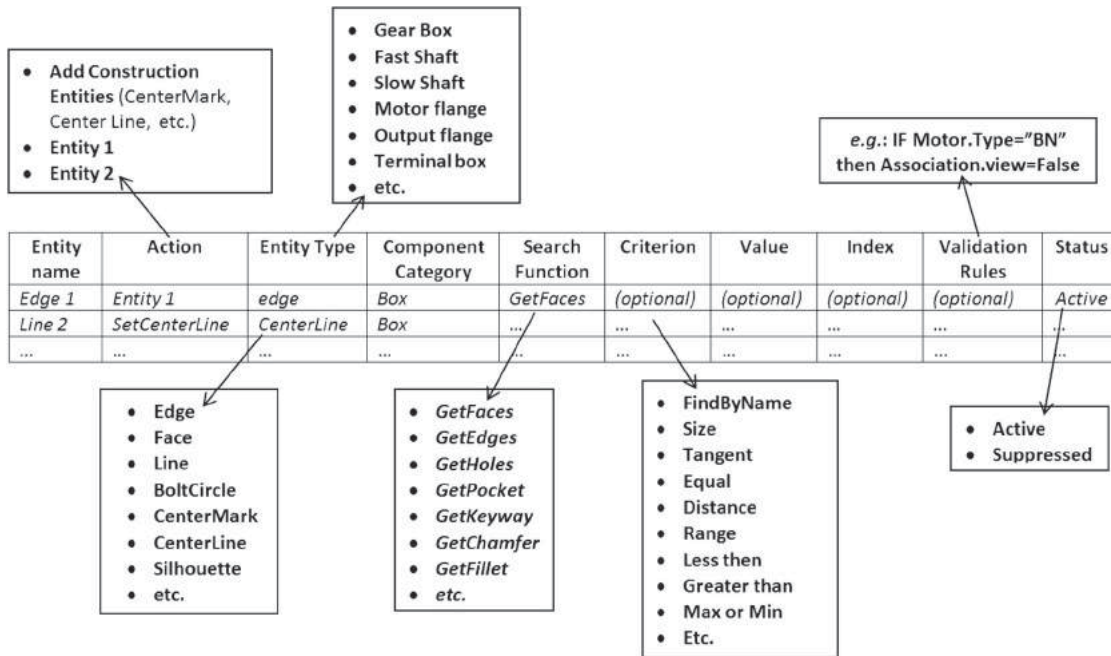


Figure 10. Associative entities table structure and content.

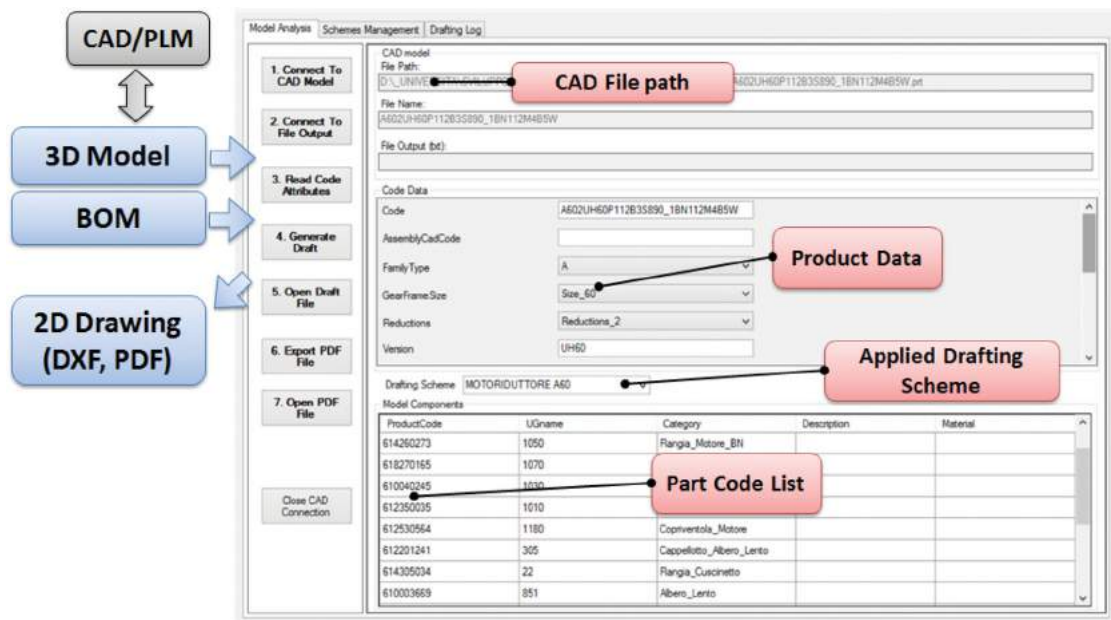


Figure 11. GUI of the prototype application tool (user interface).

and filter entities in the model, add construction elements, provide the final required entities for the dimension instantiation.

The user GUI allows to run a drafting process and is reported in Fig. 11. The user is guided by sequential steps. At first the 3D model to draft is loaded from a CAD system and the related BOM attributes loaded from the PLM/PDM systems. The code of the product and the BOM data are used to select the require DS and label each component with its category. The generation of the draft can be now started. This command launches the algorithms to connect to the CAD system to automate the generation of the views and for the identification of the dimension entities. Finally, the result is made available as DXF and PDF exports.

4.3. Results

The reported test case aimed to validate the approach and in particular the recognition of the correct geometric entities. Fig. 12 figure reports an example of a resulting

drawing concerning a gear motor A60. The generated draft of the assembly models should convey information required for mounting the product in a plant.

The drafting scheme used in the example includes one sheet, one base view and two projected views. About 30 dimensions have been defined and more than 70 geometric entities were involved in the process. The drafting scheme is robust to many variants of the input model, for instance the presence of the motor. In the Fig. 12, the dimensions created in absence of motor are shown.

The advantages emerged from the experimental tests concerns the rapidity of the execution, which is the main issue of any autonomous system.

Table 3 reports the results of different drafting processes performed using the “bDrafter” software. The comparison is among 4 different variants. Required dimensions vary between 24 and 33. The average generation time of the whole draft is about 32 sec. This time value includes the loading phase of the geometry, the feature recognition calculation and the creation of the final drawing as a 2D CAD models.

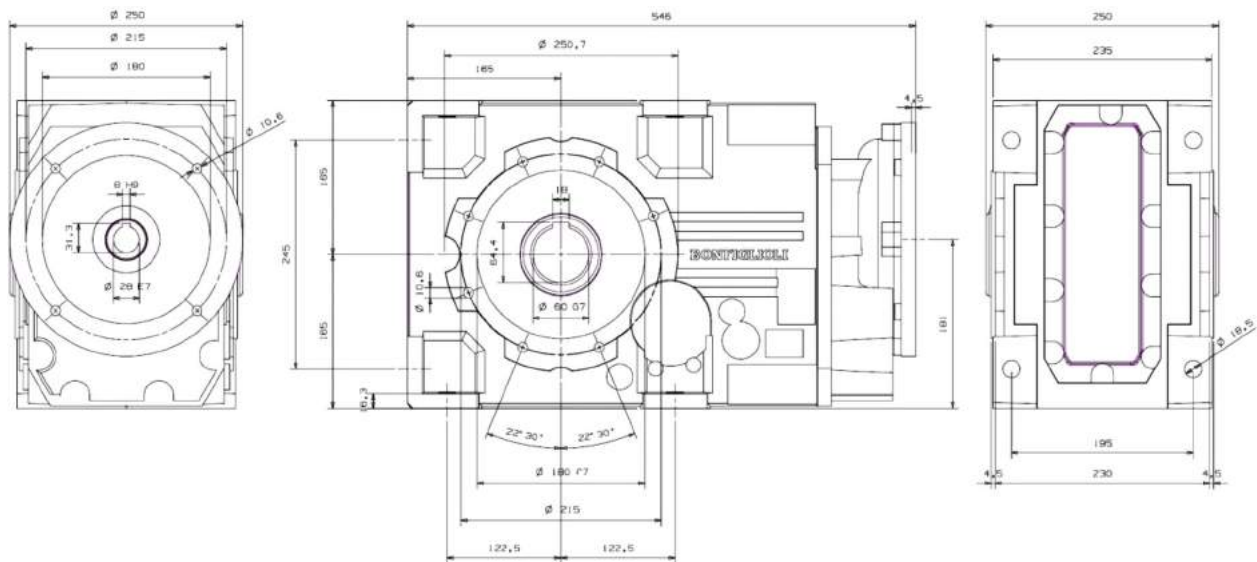


Figure 12. Example of dimensioned drawing obtained from the 3D model of a gear motor “A60.”

Table 3. Report of the results.

Model	A60+BN	A60	A30+BN	A30
Description	Gear motor with BN motor	Gear box without motor	Gear motor with BN motor	Gear box without motor
Number of dimensions	30	33	27	24
Automatically drafted dimensions	30	33	25	21
Generation time	35 sec	40 sec	30 sec	25 sec
Number manually reviewed dimensions	4	5	3	3
Reviewing time (file opening included)	35 sec	33 sec	30 sec	30 sec
Total drafting time by the “bDrafter” tool	70 sec	73 sec	70 sec	55 sec
Total manual drafting time	> 900 sec	> 900 sec	> 900 sec	> 900 sec
Total efficiency	87%	85%	81%	75%

In the reported results, the system is able to recognize all the dimensions defined for the A60 models while the A30 case can present failures in few cases. In particular, these failures regards angular dimensions. Anyway, missing dimensions as well as those not optimally located can be reviewed manually by a drafter-man in few minutes. Such reviewing activity includes the positioning of the texts, the distance of the measuring lines and so on.

The time for a manual realization of the engineering drawings is over 15 min for this kind of assembly models of gear motors. This time value has been provided by experienced drafters. The drafting time using “dDrafter” tool is reduced to values about 70 sec, thus an important gain in time reduction is demonstrated.

Thus, even considering the time for manual adjustments, the tool efficiency is very high. Efficiency is here measured as the ratio between the time gain over the time required by the manual approach. It results to be about 85% for A60 models and over 75% for A30 models.

Future testing activities and improvements in the software can increase the efficiency of the proposed tool up to 95%. Moreover, it becomes feasible to use the system in a completely automatic manner. As possible application, the drafts could be generated by a server from on line request of customers on the company web-site.

5. Conclusions

The traditional tools for the automation of modelling or drafting activities suffer the rigidity of the specific application contexts. In order to overcome such limitations, the proposed approach leverages flexible drafting schemes definitions which can be realized within a dedicated user interface without coding activities. Indeed, feature recognition algorithms allow the required dimensions to be associated to the correct geometric entities.

The approach has been experimented on the specific case of motor gearboxes showing encouraging results. The formulation of the method is general and context independent, thus it can be applied in several industrial situations.

The main limitations emerged from the experimentation phase regards the necessity to have an adequate number of entity search functions implemented in the system in order to cover the several dimensioning needs. Extending the approach to other design fields may require some coding activity to add other functions operating on the specific geometry. However, the approach is supported by the validity of the feature recognition algorithms which have shown good robustness both in the literature and in the experimentation reported in this paper.

As additional future work, a visual or graphic interface for retrieving, representing, and manipulating the Drafting Scheme could be added to the system. Some visual clues associating every parameter in the drafting scheme dialog window with relevant geometric features of 3D model, should improve the usability of the system by the final user.

Finally, the approach could be strengthened by some algorithms to support the best arrangement of dimensions around the model views.

ORCID

Roberto Raffaelli  <http://orcid.org/0000-0003-0301-454X>

Paolo Cicconi  <http://orcid.org/0000-0002-5154-8821>

Michele Germani  <http://orcid.org/0000-0003-1988-8620>

References

- [1] 3D to 2D and Dimensioning Automatically in AutoCAD, <https://www.youtube.com/watch?v=49CzbgCXCok>, Adaucogit Salt Software
- [2] Bond, A.H.; Ahmed, S.Z.: Knowledge-based automatic dimensioning, *Artificial Intelligence in Engineering*, 4(1), 1989, 32–40. [http://dx.doi.org/10.1016/0954-1810\(89\)90023-X](http://dx.doi.org/10.1016/0954-1810(89)90023-X)
- [3] Chen, K.Z.; Feng, X.A.; Lu, Q.S.: Intelligent dimensioning for mechanical parts based on feature extraction, *Computer Aided Design*, 33(13), 2001, 949–965. [http://dx.doi.org/10.1016/S0010-4485\(00\)00132-9](http://dx.doi.org/10.1016/S0010-4485(00)00132-9)
- [4] ElMaraghy, H.; Schuh, G.; ElMaraghy, W.; Piller, F.; Schönsleben, P.; Tseng, M.; Bernard, A.: Product variety management, *CIRP Annals - Manufacturing Technology*, 62(2), 2013, 629–652. <http://dx.doi.org/10.1016/j.cirp.2013.05.007>
- [5] Felfernig, A.: Standardized configuration knowledge representations as technological foundation for mass customization, *IEEE Transactions on Engineering Management*, 54(1), 2007, 41–56. <http://dx.doi.org/10.1109/TEM.2006.889066>
- [6] Han, J.; Pratt, M.; Regli, W.C.: Manufacturing Feature Recognition from Solid Models: A Status Report, *IEEE Transactions on Robotics and Automation*, 16(6), 2000, 782–796. <http://dx.doi.org/10.1109/70.897789>
- [7] Hulubei, T.; Fruder, E.C.; Wallace, R.J.: The Goldilocks problem, *Artificial Intelligence in Engineering, Design, Analysis and Manufacturing*, 17(1), 2003, 3–11. <http://dx.doi.org/10.1017/S0890060403171028L>
- [8] Lin, Y.S.; Shea, K.; Johnson, A.; Coultate, J.; Pears, J.: A method and software tool for automated gearbox synthesis, *Proceedings of ASME 2009 IDECT-CIE Conference.*, Paper No. DETC2009–86935, 5(A-B), 111–121, San Diego, CA, August 30–September 2, 2009. <http://dx.doi.org/10.1115/DETC2009-86935>
- [9] Raffaelli, R.; Mengoni, M.; Germani, M.: Improving the link between computer assisted design and configuration tools for the design of mechanical products, *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, 27(1), 2013, 51–64. <http://dx.doi.org/10.1017/S0890060412000388>

- [10] Rahmani, K.; Arezoo, B.: Boundary analysis and geometric completion for recognition of interacting machining features, *Computer Aided Design*, 38(8), 2006, 845–856. <http://dx.doi.org/10.1016/j.cad.2006.04.015>
- [11] Schotborgh, W.; Kokkeler, F.; Tragter, H.; Van Houten, E.: Why is design automation software not everywhere? Proceedings of the International Conference on Engineering Design, ICED'09, Stanford, CA, Design Society, 2009.
- [12] Visual LISP, <http://usa.autodesk.com/adsk/servlet/index?siteID=123112&id=770237>, Autodesk Inc.
- [13] XR 6 Solid Design - Automated 3D model to 2D Drawing, <http://www.onecnc.com/en/products/products-overview/design>, OneCNC Inc.