Science Arts & Métiers (SAM)

is an open access repository that collects the work of Arts et Métiers Institute of Technology researchers and makes it freely available over the web where possible.

This is an author-deposited version published in: https://sam.ensam.eu
Handle ID: .http://hdl.handle.net/10985/9314

To cite this version :

Alex BALLU, Luc MATHIEU, Jean-Yves DANTAN - Formal language for GeoSpelling - Journal of Computing and Information Science in Engineering p.35p. - 2014

Any correspondence concerning this service should be sent to the repository Administrator : archiveouverte@ensam.eu

Arts et Métiers Institute of Technology

Formal language for GeoSpelling

Alex BALLU
Univ. Bordeaux, I2M, UMR 5295
351 Cours de la Libération, 33400 TALENCE, France
alex.ballu@u-bordeaux.fr

Luc MATHIEU
LURPA, ENS Cachan – Paris Sud
61 Avenue du Président Wilson, 94235 CACHAN, France
luc.mathieu@u-psud.fr

Jean-Yves DANTAN
Arts et Métiers ParisTech, LCFC
4 rue A. Fresnel, 57078 METZ, France
jean-yves.dantan@ensam.eu

Manuscript

Abstract

In order to tackle the ambiguities of Geometrical Product Specification (GPS), GeoSpelling language has been developed to express the semantics of specifications. A detailed syntax of GeoSpelling is proposed in this paper. A specification is defined as a sequence of operations on the skin model. The syntax is based on instructions used in computer programming language: call to functions and flow control by condition and loop. In GeoSpelling, the call to functions corresponds to the declaration of operations; loops make it possible to manage a set of features with rigor and conditions to select features from a set.

Keywords

Geometrical Product Specification; Specifications semantic; GeoSpelling; Formal language

1. INTRODUCTION

Experience of metrology by coordinates led us to observe that the difficulties encountered by the industry with the language of standardized tolerances appear during the verification step. When faced with the physical part, the metrologist asks questions that the designer had not thought of. The metrologist, who does not find perfect shapes, encounters difficulties in defining a correct measuring process.

Standardized graphical language for GPS (Geometrical Product Specification) appears to be ambiguous. One needs to consult the standards to try to understand the specifications on a technical drawing.

We have been developing GeoSpelling for more than 15 years, in an attempt to overcome the difficulties experienced in understanding and defining the geometrical specifications. The objective of the GeoSpelling language is to enable the semantics of specifications to be expressed and to define their meaning clearly. This work revealed some ambiguities in the standards [1][2] and the concepts of GeoSpelling have been integrated into ISO 17450-1 standard [3].

Nevertheless, GeoSpelling is not relevant only to standardized specifications; it is also relevant for all geometrical specifications useful for expressing geometrical requirements during the development cycle of a mechanical product.

However, GeoSpelling still lacks a complete syntax for it to be completely unambiguous. As a specification is defined as a sequence of operations on the skin model, we propose a syntax similar to the syntax of procedural computer programming language. In computer programs, the instructions are essentially: call to functions and flow control by condition and loop. In GeoSpelling,

call to functions corresponds to the declaration of operations, but the concepts of condition or loop are absent and still need to be defined.

First, we review the main features of GeoSpelling, then the syntax, inspired from computer programming language, is described through examples.

2. MAIN CONCEPTS OF GEOSPELLING

GeoSpelling is a model used to describe ideal and non-ideal geometry. It enables us to express the specification from function to verification using a common language. This model is based on geometrical operations which are applied not only to ideal features, defined in a CAD system, but also to the non-ideal features representing a real part.

The language is generic. With GeoSpelling, standardized specifications can be described using only six operations and a limited number of basic characteristics.

2.1. Skin model

Our first idea was to imagine the parts with potential defects for tolerancing and to consider a part model with deviations. This was called the skin model [4][5][6]. The skin model (or non-ideal surface model) has been defined in ISO 17450-1 [3] as “a model of the physical interface of the workpiece with its environment”.

The skin model is a closed surface including geometric defects such as orientation and position deviations, and texture defects. It is distinct from the nominal part model, which only considers perfect shapes (Fig 1).

2.2. Specification operator

Our second idea was to establish a process based on successive operations to define geometric features and complex characteristics from the skin model.

From there, the following definition of geometric specification was created:

- a tolerance is a condition,
- on a dimension defined by a characteristic expressed,
- on a geometric feature or between geometric features,
- identified by operations from the skin model.

On this basis the ISO TC 213 committee has defined the concepts of specification operator and verification operator in ISO 17450-2 [7]. The specification operator is a sequence of operations from which the semantics of the specification can be defined. The verification operator defines the sequence of operations actually carried out to verify a particular part.

2.3. Geometric features

Two types of features are distinguished: ideal features and non-ideal features.

2.3.1. Ideal features

Ideal features are features such as:

- a plane of the nominal model of the part,
- a cylinder fitted to a nominally cylindrical real feature.

An ideal feature can be defined by an equation. Among the ideal features, we distinguish those that are completely independent of the real surface of the part, called nominal features, which

constitute the nominal model defined by the CAD system, and the associated features obtained by an operation of association from features of the skin model.

Ideal features are characterized by their invariance class [8] and their type. They are often unbounded, but they may be limited.

2.3.2. Non-ideal features

Non-ideal features are features such as:

- a surface portion of the skin model,
- a median line.

A non-ideal feature cannot be defined by an equation. It depends completely on the skin model. The first non-ideal feature defined in the process of operations to describe a specification is the skin model itself.

2.4. Operations

The operations used to define the features can be divided into six classes (Fig 2): partition, extraction, filtration, collection, association and construction.

Partition is an operation used to identify bounded feature(s) from non-ideal feature(s) or from ideal feature(s).

Filtration is an operation used to create a non-ideal feature by reducing the level of information of a non-ideal feature.

Extraction is an operation used to identify specific points from a non-ideal feature.

Collection is an operation used to consider more than one feature together.

Association is an operation used to fit ideal feature(s) to non-ideal feature(s) according to criteria.

Construction is an operation used to build ideal feature(s) from other ideal features under constraints.

2.5. Characteristics

Characteristics, useful in the definition of specifications, belong to five families:

- Intrinsic characteristic (Fig 3)
- Situation characteristic between ideal features (Fig 4)
- Situation characteristic between ideal and limited features
- Situation characteristic between ideal and non-ideal features (Fig 5)
- Situation characteristic between non-ideal features (Fig 6)

To ensure a good understanding of this paper, some details about characteristics must be given.

Distances between two features are based on functions of the distances from the points of one feature to the other. The distance of a point of one feature to the other is defined as the minimum distance from this point to the points of the other feature. The main functions are minimum, maximum, mean and mean of the squares (variance) of the distances. The distance based on the mean of the squares of the distances is denoted quadratic distance in this paper.

Mathematically, a distance is positive, however, signed distances are very useful to express geometrical specifications. The sign depends on the relative position of the point with respect to the feature. In 3D space, a signed distance could be defined with respect to a surface, where one side of

the feature is defined as positive, and the other as negative. If the feature is a material limit, the sign is defined positive towards the outside of the material and negative towards the inside.

For example, minimum material distance is very useful to indicate that a feature is outside material with respect to another feature. In this case, material distance is positive (Fig 7).

Taking into account the limited length of the paper and the large number of characteristics in the approach, we recommend reading appendix B of ISO 17450-1 [3] and [9].

3. FORMALIZATION OF THE SPECIFICATION OPERATOR

Geometrical specifications are used to illustrate the efficiency of the GeoSpelling model. First, the specifications are presented, then the semantics are carried out using GeoSpelling and finally this expression is formalized. The geometrical specifications are for parts of a workpiece holder for a milling machine, introduced by Anselmetti and fully described in [10].

3.1. Perpendicularity tolerance

Let us consider a simple specification: a perpendicularity tolerance of a plane with respect to a cylinder (Fig. 8).

3.2. Specification operator of the perpendicularity

As a first approach, one could interpret the specification as the angle between a plane and the perfect axis of a cylinder. However, the skin model representation of the part shows that the interpretation is not so straightforward (Fig. 9). There is no ideal plane and no ideal cylinder on the skin model!

According to ISO 1101 [11]:

- a non-ideal surface S1a is identified by an operation of partition from the skin model, S1a corresponds to the nominal cylinder NCY1a
- an ideal cylinder CY1a is identified by an operation of association, according to the following criteria
  - Constraint: Minimum material distance between CY1a and S1a ≥ 0 (CY1a is outside material)
  - Objective to minimize: Diameter of CY1a
- a non-ideal surface S1c is identified by an operation of partition from the skin model, S1c corresponds to the nominal plane NPL1c
- an ideal plane PL1c is identified by an operation of association, according to the following criteria
  - Constraint: Angle between PL1c and the axis of CY1a = 90°
  - Objective to minimize: Maximum distance between PL1c and S1c
- the maximum distance between the plane PL1c and the surface S1c is evaluated
- the value of the characteristic shall be less than or equal to half of the tolerance value 0.05.

3.3. Formal expression of the specification operator

To reduce the ambiguities even further, the specification operator could be expressed more formally, using a sequence of instructions like a computer program code (Table 1).

Note: it is not really a computer program, because the skin model is defined by an infinite number of parameters so as to be able to consider all types of geometrical variations (location, orientation, form, texture). Infinite describability implies that the model cannot be represented or used as a variable in a computer program [6].

The rules of syntax used in table 1 are described in the following paragraphs.

3.3.1. Comments

A comment begins with the character %.

3.3.2. Objects

By convention, the variables (objects) are in capital characters (except for surface identifier): SM1, NM1, NCY1a, S1a, D1aMIN, DEV…

The constructor to create an object of a class has the same name as the class: Cylinder() to create a cylinder, Plane()… Dmatmin(), Dmax(), Angle() are the constructors of the corresponding characteristics.

A constructor may have arguments. For example, Dmatmin(S1a,CY1a) creates a characteristic of type “Minimum material distance” between S1a and CY1a.

Cylinder() creates an object representing a cylinder, but the cylinder itself is not defined, the situation and diameter are not instantiated.

Similarly, Dmatmin(S1a,CY1a) creates the object representing the distance, but the distance itself is not evaluated (computed). The evaluation is done by the function Eval(D1c).

3.3.3. Operations

The operations of GeoSpelling are functions which have the same name: Partition(), Association()… They are not detailed in this paper.

For partition (Partition(SM1,NM1,NCY1a)), the inputs are:

- the skin model (SM1),
- the nominal model (NM1),

- the nominal surface corresponding to the feature to partition.

For association (Association(CY1a, D1aMIN>=0, Dia(CY1a))) the inputs are:

- the associated feature (CY1a),
- the constraints (D1aMIN>=0)
- the objective to minimize (Dia(CY1a)).

4. CALL TO FUNCTION AND LOOP INSTRUCTION

The next example is a position tolerance of a median line with a datum system. It shows that formalization is not so easy.

4.1. Position tolerance

Position tolerance is presented in figure 10 and the skin model in figure 11.

The formal expression of the specification is presented in table 2. Datum C and datum D are defined according to ISO 5459 [12]. The position deviation is the maximum distance from the median line (ML2e) to the datum axis D (Axis(CY2d)).

4.2. Call to function

In the position operator, the identification of the toleranced feature uses a call to the "MedianLine" function to define the median line: ML2e = MedianLine(S2e). The process for defining the median line is given in table 3. As in a computer program, a call to function makes a reusable code possible. It makes sense to define a function dedicated to the median line because this concept is often used in GPS.

The definition of a function begins with the keyword "function". In the example (Table 3):

- the name of the function is "MedianLine".

- the input is a non-ideal cylindrical surface (SCY),
- the output is the median line (ML).

In general, the number of inputs and outputs are not limited to one, as for the MedianLine function.

4.3. Loop instruction

According to ISO 14660-2 [13], the median line of a non-ideal cylindrical surface is: “the locus of centres of cross-sections, where

- the centres of cross-sections are centres of associated circles; and
- the cross-sections are perpendicular to the axis of the associated cylinder obtained from the extracted surface”

With GeoSpelling, we can translate this expression as (see figure 12):

- an ideal cylinder CY is identified by an operation of association, according to the least squares criteria
  - Objective to minimize: quadratic distance between CY and SCY (non-ideal cylindrical surface)
- a set of planes PL is identified by an operation of construction (with an infinite number of solutions), according to the following criteria
  - Constraint: Angle between PL and the axis of CY = pi/2
- for each plane PL
- a non-ideal circular line SCE is identified by an operation of partition from SCY, SCE is in the plane PL

- an ideal circle CE is identified by an operation of association, according to the least squares criteria
  - Objective to minimize: quadratic distance between CE and SCE
- the median line (ML) is identified by an operation of collection on all the centers of circles CE.

The median line operator is more complex than the previous operators because:

- there is an infinite number of planar sections PL solutions of the construction,
- the partition of the non-ideal circular line and the association of the circles must be done for each plane PL.

To meet these needs of expression, we propose a formalization in table 3. The construction of the planes is defined as the solution of a mathematical problem with a perpendicularity constraint with respect to the axis of CY. As there is no single solution, the variable SET_PL is the set of all the perpendicular planes. It is important to note that the number of solution planes is infinite. The consequence is that the planes cannot be computed.

The repetition of the partition and of the association for each plane is expressed with a loop instruction. We use the loop instruction “foreach”:

foreach.x.in X

Instruction 1;

Instruction 2;

…

end

or

foreach x.in X

Instruction 1;

Instruction 2;

…

y = …;

collect y in Y

For each value x of the set X, a series of instructions is produced. With the “collect” keyword closing the “foreach” instruction, the different values y are collected in a set Y.

5. CONDITIONAL INSTRUCTION

The two main types of control instruction in computer programs are loop instruction and conditional instruction. We have introduced the loop instruction, and the study of a size tolerance will illustrate a use of conditional instruction.

5.1. Envelope requirement

Consider the size tolerance for the cylinder NCY2e (fig. 10). The envelope requirement operator is simple to express (table 4).

Note: the objective of the association is to maximize Dia(CY2e) which is equivalent to minimizing –Dia(CY2e).

5.2. Size tolerance

The size tolerance is more complex. The size of a non-ideal cylindrical surface is based on the distances of the opposite points on the feature, where [ISO 14660-2]:

- “the connection line between the points includes the associated circle centre; and

- the cross-sections are perpendicular to the axis of the associated cylinder obtained from the extracted surface”

In this case:

- there is an infinite number of cross PL as for the cross sections in the case of the median line,
- there is an infinite number of connection lines in each cross section passing through the associated circle center,

It is not noted in the standard that intersection of the line connection with the non-ideal cylindrical surface must have two points. As the partition is done on non-ideal features, it is possible to have just one intersection point. We are going to introduce a conditional instruction to manage this case (table 5). Traditional syntax is used:

if condition

…

else

…

end

In the example, the condition addresses the number of points of the result of the partition

For the example, the condition is on the BIPOINT class. Normally BIPOINT is a set of two points, belonging to the class “BiPoint”. If the class of BIPOINT is the class “BiPoint”, then the distance between the two points is computed, otherwise, the variable is null.

During the collection of the distances, if D is null, no variable is collected.

After the loop, the minimum and the maximum of the set of distances is determined.

6. CONCLUSION

A formal expression of specifications based on GeoSpelling is illustrated using examples. In addition to operations, which are considered as the functions of a computer program, control instructions are used. Control instructions are a loop instruction of type “foreach” and a conditional instruction of type “if then else”.

The rigor of the proposed code makes it difficult to use because the function and instructions are low level instructions. Efforts must be made to simplify in order to obtain a user-friendly language. To this end, higher level functions must be developed.

The functional specification between two parts of an assembly is not studied in this paper. However, the formalization of GeoSpelling presented here enables such specifications to be expressed by introducing complementary concepts.

Applications of the formal language based on GeoSpelling go beyond the simple expression of the semantics of the specification. It could be used to simulate metrology, assembly or manufacturing sequences. These applications will be presented in future papers.

REFERENCES

[1] Ballu, A., Mathieu, L., 1995, "Univocal expression of functional and geometrical tolerances for design, manufacture and inspection," Proceedings 4th CIRP International Seminar on Computer Aided Tolerancing, Tokyo, Japan, pp.31-46.

[2] Srinivassan, V., 2001, "An integrated view of geometrical product specification and verification," Proceedings 7th CIRP International Seminar on Computer Aided Tolerancing, France.

[3] ISO 17450-1:2011, "Geometric Product Specification (GPS) – General concepts – Part 1: Model for geometrical specification and verification".

[4] Ballu, A., Mathieu, L., Dantan, J.-Y., 2001, "Global View of Geometrical Specifications," Proceedings 7th CIRP International Seminar on Computer-Aided Tolerancing, France, pp. 19-30.

[5] Dantan, J.Y., Ballu, A., Mathieu, L., 2001, "Expression des spécifications géométriques des produits – Classification des éléments géométriques," Journée thématique PRIMECA, ENSAM Aix en Provence, France, pp. 69-78.

[6] Anwer, N., Ballu, A., Mathieu, L., 2013, "The skin model, a comprehensive geometric model for engineering design," Annals of the CIRP, 62(1), pp. 143–146.

[7] ISO 17450-2:2012, Geometrical product specifications (GPS) - General concepts - Part 2: Basic tenets, specifications, operators, uncertainties and ambiguities.

[8] Clément, A., Desrochers, A., Rivière, A., 1991. "Theory and practice of 3D tolerancing for assembly," CIRP Seminar on Computer Aided Tolerancing, Penn State University, USA.

[9] Ballu, A., Dantan, J.-Y., Mathieu, L., 2010, "Language of specification: GeoSpelling, Geometric tolerancing of product," Ed. Villeneuve F and Mathieu L., pp. 23-53, ISBN 978-1-84821-118-6, ISTE Wiley.

[10] Villeneuve F and Mathieu L., 2010, "Geometric tolerancing of product," ISBN 978-1-84821-118-6, ISTE Wiley.

[11] ISO 1101:2012, "Geometrical product specifications (GPS) - Geometrical tolerancing - Tolerances of form, orientation, location and run-out".

[12] ISO 5459:2011, "Geometrical product specifications (GPS) -- Geometrical tolerancing -- Datums and datum systems".

[13] ISO 14660-2:1999, "Geometrical Product Specifications (GPS) -- Geometrical features -- Part 2: Extracted median line of a cylinder and a cone, extracted median surface, local size of an extracted feature".

Table Caption List

Table 1 Formal expression of perpendicularity

Table 2 Formal expression of position

Table 3 Formal description of the median line

Table 4 Formal expression of envelope

Table 5 Formal expression of size

Figure Captions List

Fig. 1 Nominal and “skin” models

Fig. 2 Operations

Fig. 3 Intrinsic characteristics: (a) diameter; (b) apex angle

Fig. 4 Situation characteristic between ideal features:

(a) distance point/straight line; (b) distance between two straight lines;

(c) angle straight line/plane

Fig. 5 Situation characteristic between ideal and non-ideal features:

(a) distance non-ideal line/straight line; (b) distance non-ideal line/circle

Fig. 6 Situation characteristic between non-ideal features

Fig. 7 Minimum and maximum material distances

Fig. 8 Perpendicularity tolerances

Fig. 9 Skin model for perpendicularity

Fig. 10 Position tolerance

Fig. 11 Skin model for the position

Fig. 12 Median line (ML)

```
%SM1: Skin model of the part
%NM1: Nominal model of the part
%NCY1a: Nominal cylinder of NM1
%NPL1c: Nominal plane of NM1

% Identification of the datum A

S1a = Partition(SM1,NM1,NCY1a);
CY1a = Cylinder();
D1aMIN = Dmatmin(S1a,CY1a);
CY1a = Association(CY1a,D1MIN>=0,Dia(CY1a));

% Identification of the toleranced feature

S1c = Partition(SM1,NM1,NPL1c);
PL1c = Plane();
D1c = Dmax(S1c,PL1c);
A1 = Angle(Axis(CY1a),PL1c);
PL1c = Association(PL1c,A1=pi/2,D1c);

% Evaluation of the perpendicularity deviation

DEV := Eval(D1c);
RESULT := Eval(DEFECT<=0.05/2);
```

Table 1. Formal expression of perpendicularity

```
% Identification of datum C

S2c = Partition(SM2,NM2,NPL2c);
PL2c = Plane();
D2cMAX = Dmatmax(S2,PL2c);
D2cMIN = Dmatmin(S2,PL2c);
PL2c = Association(PL2c, D2cMIN>=0,D2cMAX);

% Identification of datum D

S2d = Partition(SM2,NM2,NCY2d);
CY2d = Cylinder();
D2dMIN = Dmatmin(S2d,CY2d);
A2 = Angle(PL2c,Axis(CY2d));
CY2d = Association(CY2d,[D2dMIN>=0,A2=pi/2],Dia(CY2d));

% Identification of the toleranced feature

S2e = Partition(SM2,NM2,NCY2e);
ML2e = MedianLine(S2e);

% Evaluation of the position deviation

D2e =Dmax(ML2e,Axis(CY2d));
DEV := Eval(D2e);
RESULT := Eval(DEFECT<=0.08/2);
```

Table 2. Formal expression of position

```
function ML = MedianLine(SCY)

% ML: Median Line
% SCY: Non-ideal cylindrical surface

% Identification of the least-squares cylinder

CY = Cylinder();
DCY = Dquad(SCY,CY);
CY = Association(CY,,DCY);

% Identification of the perpendicular sections

PL = Plane();
A = Angle(Axis(CY),PL);
SET_PL = Solve(PL,A=pi/2);

% Identification of the median line

foreach PL in SET_PL
   SCE = Partition(SCY,PL);
   CE = Circle();
   DCE = Dquad(SCE,CE);
   CE = Association(CE,,SET_DCE);
collect Center(CE) in ML
```

Table 3. Formal description of the median line

```
% Identification of the toleranced feature
S2e = Partition(SM2,NM2,NCY2e);
% Identification of the maximum inscribed cylinder (Envelope)
CY2e = Cylinder();
D2eMIN = Dmatmin(S2e,CY2e);
CY2e = Association(CY2e,D2eMIN>=0,-Dia(CY2e));
% Evaluation of the envelope defect
DEV := Dia(CY2e)-(9.03-0.02);
RESULT := Eval(DEFECT<=0);
```

Table 4. Formal expression of envelope

```
% Identification of the toleranced feature
S2e = Partition(SM2,NM2,NCY2e);
% Identification of the least squares cylinder
CY2e = Cylinder();
DCY2e = Dquad(S2e,CY2e);
CY2e = Association(CY2e,,DCY2e);
% Identification of the perpendicular straight lines
SL2e = StraightLine();
D2e = Distance(SL2e,Axis(CY2e));
A2e = Angle(SL2e,Axis(CY2e));
SET_SL2e = Solve(SL2e,[D2e=0,A2e=90°]);
% Identification of the bipoints
foreach SL2e in SET_SL2e
   BIPOINT = Partition(S2e,SL2e;
   if GetClass(BIPOINT)='BiPoint'
      D = Eval(Distance(BIPOINT));
   else
      D = null;
   end
collect D in SET_D
% Evaluation of the size deviation
DEVMIN = Min(SET_D);
DEVMAX = Max(SET_D);
RESULT := Eval(DEVMIN>=9.01 & DEVMAX<=9.05);
```

Table 5. Formal expression of size

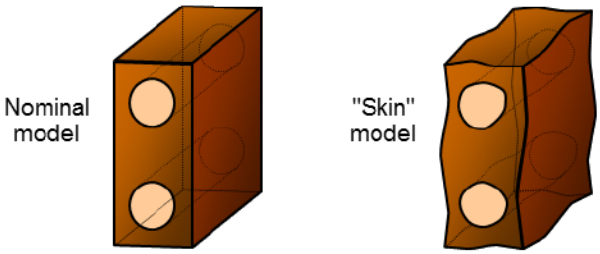

Fig. 1. Nominal and “skin” models

Fig. 2. Operations

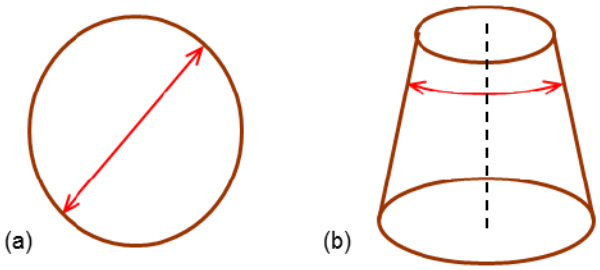

Fig. 3. Intrinsic characteristics: (a) diameter; (b) apex angle

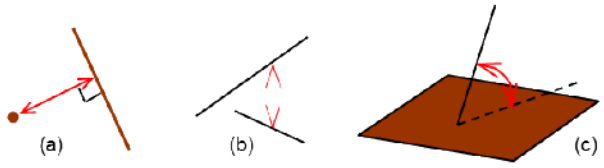

Fig. 4. Situation characteristic between ideal features:
(a) distance point/straight line; (b) distance between two straight lines; (c) angle straight line/plane

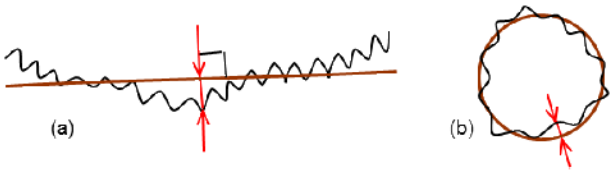

Fig. 5. Situation characteristic between ideal and non-ideal features:
(a) distance non-ideal line/straight line; (b) distance non-ideal line/circle

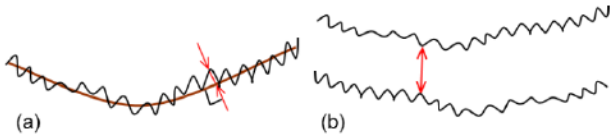

Fig. 6. Situation characteristic between non-ideal features

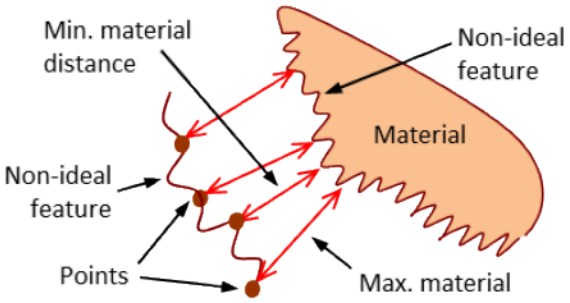

Fig. 7. Minimum and maximum material distances

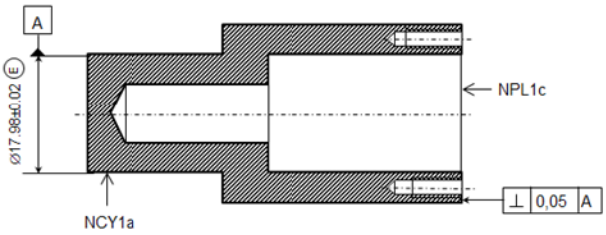

Fig. 8. Perpendicularity tolerances

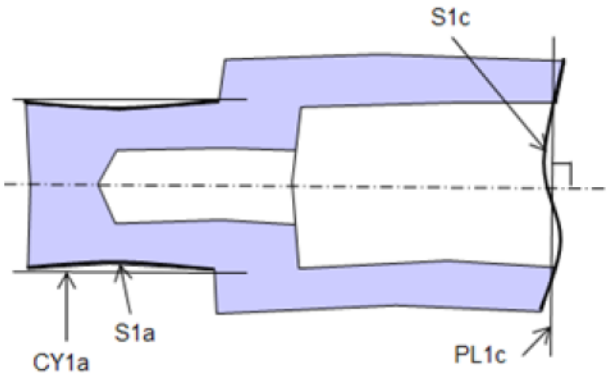

Fig. 9. Skin model for perpendicularity

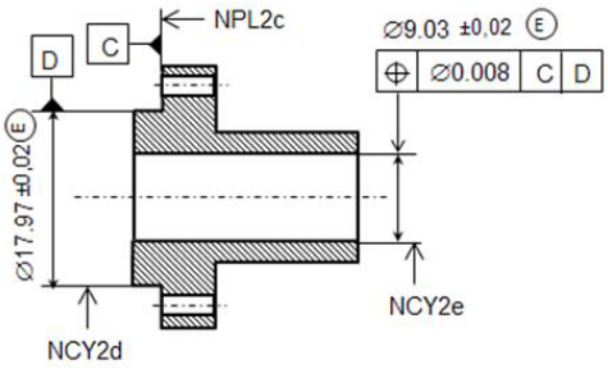

Fig. 10. Position tolerance

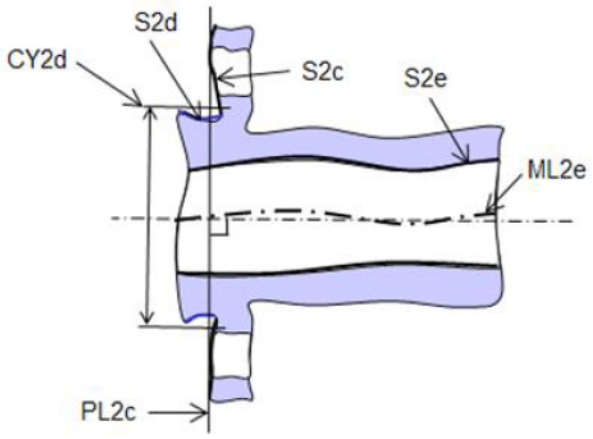

Fig. 11. Skin model for the position

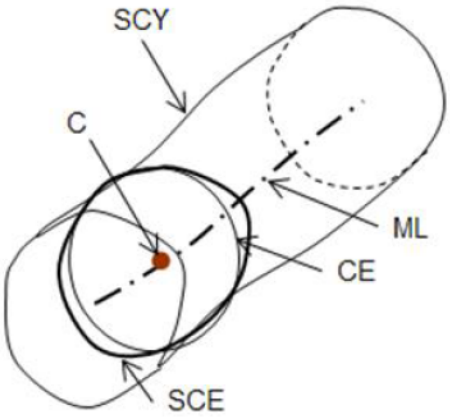

Fig. 12. Median line (ML)