

Chapter 2

Geometric Tolerance Analysis

Wilma Polini

Abstract This chapter focuses on five main literature models of geometric tolerance analysis – *vector loop*, *variational*, *matrix*, *Jacobian*, and *torsor* – and makes a comparison between them in order to highlight the advantages and the weaknesses of each, with the goal of providing a criterion for selecting the most suitable one, depending on the application. The comparison is done at two levels: the first is qualitative and is based on the analysis of the models according to a set of descriptors derived from what is available in the literature; the second is quantitative and is based on a case study which is solved by means of the five models. Finally, in addition to providing comparative insight into the five tolerance analysis models, some guidelines are provided as well, related to the development of a novel approach which is aimed at overcoming some of the limitations of those models.

2.1 Introduction

Increasing competition in industry leads to the adoption of cost-cutting programs in the manufacturing, design, and assembly of products. Current products are complex systems, often made of several assemblies and subassemblies, and including complex part geometries; a wide variety of different requirements must be satisfied at the design and manufacturing stages in order for such products to fulfill the desired functional requirements. The paradigm of concurrent engineering enforces an

W. Polini
Dipartimento di Ingegneria Industriale, Università degli Studi di Cassino,
Via Gaetano di Biasio 43, 03043 Cassino, Italy.
e-mail: polini@unicas.it

approach where product design and manufacturing process planning activities are carried out in parallel in highly communicative and collaborative environments, with the aim of reducing reworking times and discard rates. In recent years, the importance of assessing the effects of part tolerances on assembly products has increasingly been acknowledged as one of the most strategic activities to be pursued with the goal of ensuring higher production qualities at lower costs. In fact, while the need for assigning some type of dimensional and geometric tolerances to assembly components is widely recognized as a necessary step for ensuring a standardized production process and for guaranteeing the correct working of the assembly to the required levels of satisfaction, the relationships between the values assigned to such tolerances and final product functionality are more subtle and need to be investigated in greater detail. As defined by designer intent, assembled products are built to satisfy one or more functional prerequisites. The degree to which each of such functional prerequisites is satisfied by the product is usually strongly related to a few key dimensions of the final assembly itself. In related literature, such key dimensions are often called *key functional parameters* or *design dimensions*, or *functional requirements*, which is the term that will be adopted in this chapter. Functional requirements are typically the result of the stack-up of geometries and dimensions of several parts; and their final variability, also fundamental in determining overall functional performance, arises from the combined effects of the variabilities associated with the parts involved in the stack-up. Tolerance values assigned to parts and subassemblies become critical in determining the overall variability of functional requirements and, consequently, the functional performance of the final product. Moreover, when analyzing the chain of connected parts, it becomes relevant to identify those tolerances that – more than others – have an influence on the final outcome, since it has been shown that often the *70–30 rule* applies, meaning that 30% of the tolerances assigned to the components are responsible for 70% of the assembly geometric variation. The variabilities associated with dimensions and geometries of the assembly components combine, according to the assembly cycle, and generate the variability associated with the functional requirements. *Tolerance stack-up functions* are mathematical models aimed at capturing such combinations. They have that name because they are designed as functions whose output is the variation range of a functional requirement, and whose inputs are the tolerances assigned to assembly components. In other words, a tolerance analysis problem implies modeling and solving a tolerance stack-up function, in order to determine the nominal value and the tolerance range of a functional requirement starting from the nominal values and the tolerance ranges assigned to the relevant dimensions of assembly components.

In a typical tolerance analysis scenario, linear or nonlinear tolerance stack-up functions may need to be solved, depending on the problem being studied, and on the way it was modeled in the analysis. Alternative assembly cycles may be considered within the analysis in order to identify the one allowing assembly functionality with the maximum value of the tolerance range assigned to the components. The importance of an effectual tolerance analysis is widely recognized: significant problems may arise during the actual assembly process if the tolerance

analysis on a part or subassembly was not carried out or was done to an unsatisfactory level (Whitney 2004). It may even happen that the product is subjected to significant redesign because of unforeseen tolerance problems, which were not detected prior to actual assembly taking place. In this case business costs may be significantly high, especially when considering that 40–60% of the production costs are estimated as due to the assembly process (Delchambre 1996).

Many well-known approaches, or *models*, exist in the literature for tolerance analysis (Hong and Chang 2002; Shen *et al.* 2004). The *vector loop model* adopts a graph-like schematization where any relevant linear dimension in the assembly is represented by a vector, and an associated tolerance is represented as a small variation of such a vector (Chase *et al.* 1997; Chase 1999). Vectors are connected to form chains and/or loops, reflecting how assembly parts stack up together in determining the final functional requirements of the assembly. Stack-up functions are built by combining the variations associated with vectors involved in each chain into mathematical expressions, which can then be solved with different approaches.

The *variational model* has its roots in parametric geometric modeling, where geometry can be modeled by mathematical equations that allow shape and size attributes to be changed and controlled through a reduced set of parameters (Gupta and Turner 1993; Whitney *et al.* 1999; Li and Roy 2001). Parametric modeling can be used as the starting point for reproducing small variations in an assembly part, within the ranges defined by a given tolerance. In the *matrix model*, the approach aims at deriving the explicit mathematical representation of the geometry of each tolerance region (the portion of space where a feature is allowed to be, given a set of tolerances); this is done through displacement matrices, which describe the small displacements a feature is allowed to have without violating the tolerances (Desrochers and Rivière 1997; Clément *et al.* 1998).

In the *Jacobian model*, tolerance chains are modeled as sequences of connected pairs of relevant surfaces; displacement between such surfaces (whether nominal or due to variations allowed by tolerances) is modeled through homogeneous coordinate transformation matrices (Laperrière and Lafond 1999; Laperrière and Kabore 2001). The way the matrices are formulated draws inspiration from a common approach adopted in robotics which involves the use of Jacobian matrices.

Finally, in the *torsor model*, screw parameters are introduced to model 3D tolerance zones (Ballot and Bourdet 1995, 1997). The name derives from the data structure adopted to collect the screw parameters (*i.e.*, the torsor).

The five modeling approaches introduced above propose different solutions to specific aspects of the tolerance analysis problem; all have strong points and weaknesses that may make them inadequate for specific applications. Most aspects differentiating the approaches are related to how geometric variability of parts and assemblies is modeled, how joints and clearance between parts are represented, how stack-up functions are solved, and so forth. Moreover, it is difficult to find literature work where the different approaches are compared systematically with the help of one or more case studies aimed at highlighting the advantages and disadvantages of each.

viewpoint of a strict application of standardized tolerancing rules, is directly derived from the current practice adopted for the actual industrial product.

The case study is representative of all the main aspects and critical issues involved in a typical tolerance analysis problem, and it is simple enough to allow for the application of a simplified manual computation procedure to obtain the extreme values of the gap g for the special case where *only dimensional tolerances are considered*. The manual computation is based on searching for the worst-case conditions, *i.e.*, the combinations of part dimensions that give rise to the maximum and minimum gap values; since no geometric tolerances are considered, part geometries are assumed at nominal states.

The maximum value of the gap is calculated by considering the maximum height and width of the box, together with the minimum value of the radius of the disks:

$$\begin{aligned} g_{\max_{\text{dim}}} &= 80.5 - 19.95 - \sqrt{(19.95 \cdot 2)^2 - (50.40 - 19.95 - 19.95)^2} \\ &- 19.95 = 2.1064 \text{ mm.} \end{aligned} \quad (2.1)$$

In the same way, the minimum value of the gap is

$$\begin{aligned} g_{\min_{\text{dim}}} &= 79.5 - 20.05 - \sqrt{(20.05 \cdot 2)^2 - (49.8 - 20.05 - 20.05)^2} \\ &- 20.05 = 0.4909 \text{ mm.} \end{aligned} \quad (2.2)$$

The variability of the gap is the difference between the maximum or the minimum values and the nominal one:

$$\begin{aligned} \Delta g_{\text{dim}_1} &= +(g_{\max_{\text{dim}}} - g_N) = (2.1064 - 1.2702) \cong +0.84 \text{ mm,} \\ \Delta g_{\text{dim}_2} &= -(g_N - g_{\min_{\text{dim}}}) = -(1.2702 - 0.4909) \cong -0.78 \text{ mm.} \end{aligned} \quad (2.3)$$

Albeit operating on a simplified problem (geometric tolerances are neglected) the manual computation of the gap boundary values provides a useful support for the quantitative comparison of the five methods, at least when they are applied by considering dimensional tolerances only. The manually obtained, extreme gap values will be used as reference values later on, then the results of the five methods will be discussed.

Furthermore, the manual computation procedure highlights one of the fundamental issues analyzed in this chapter, *i.e.*, how hard it actually is to include geometric tolerances in any model attempting to represent geometric variability. The investigation of how this challenge is handled in the analysis of tolerance chains is of fundamental importance when analyzing the performance and limitations of the five approaches: under this assumption, it will be shown how each approach provides a different degree of support to the inclusion of geometric tolerances, and how each requires different modeling efforts, simplifications, and workarounds, in order to let geometric tolerances be included in the tolerance chain analysis problem.

2.3 The Vector Loop Model

The vector loop model uses vectors to represent relevant dimensions in an assembly (Chase *et al.* 1995, 1996). Each vector represents either a component dimension or an assembly dimension. Vectors are arranged in chains or loops to reproduce the effects of those dimensions that stack together to determine the resultant assembly dimensions. Three types of variations are modeled in the vector loop model: *dimensional variations*, *kinematic variations*, and *geometric variations*.

In a vector loop model, the magnitude of a geometric dimension is mapped to the length (L_i) of the corresponding vector. Dimensional variations defined by dimensional tolerances are incorporated as \pm variations in the length of the vector. Kinematic variations describe the relative motions among mating parts, *i.e.*, small adjustments that occur at assembly time in response to the dimensional and geometric variations of the components. In the vector loop model, kinematic variations are modeled by means of *kinematic joints*, *i.e.*, schematizations such as the *slider*. In vector loop models, there are six common joint types available for 2D assemblies and 12 common joints for 3D assemblies. At each kinematic joint, assembly adjustments are turned into ranges for the motions allowed by the joint (*i.e.*, degrees of freedom). A local datum reference frame (DRF) must be defined for each kinematic joint.

Geometric variations capture those variations that are imputable to geometric tolerances. These are modeled by adding additional degrees of freedom to the kinematic joints illustrated above. This introduces a simplification: although geometric tolerances may affect an entire surface, in vector loop models they are considered only in terms of the variations they induce at mating points, and only in the directions allowed by the type of kinematic joint. Depending on what type of geometric variation is represented by the tolerance and what motions are allowed at the kinematic joint, a geometric tolerance is typically modeled as an additional set of translational and rotational transformations (*e.g.*, displacement vectors, rotation matrices) to be added at the joint.

To better understand the vector loop model, the basic steps for applying it to a tolerance analysis problem are provided below (Gao *et al.* 1998; Faerber 1999; Nigam and Turner 1995):

1. *Create the assembly graph.* The first step is to create an assembly graph. The assembly graph is a simplified diagram of the assembly representing the parts, their dimensions, the mating conditions, and functional requirements, *i.e.*, the final assembly dimensions that must be measured in order to verify that the product is capable of providing the required functionality. An assembly graph assists in identifying the number of vector chains and loops involved in the assembly.
2. *Define the DRF for each part.* The next step is to define the DRF for each part. DRFs are used to locate relevant features on each part. If there is a circular contact surface, its center is considered as a DRF too.

3. *Define kinematic joints and create datum paths.* Each mating relation among parts is translated into a kinematic joint. Kinematic joints are typically located at contact points between parts. Datum paths are geometric layouts specifying the direction and orientation of vectors forming the vector loops; they are created by chaining together the dimensions that locate the point of contact of a part with another, with respect to the DRF of the part itself.
4. *Create vector loops.* With use of the assembly graph and the datum paths, vector loops are created. Each vector loop is created by connecting datums. Vector loops may be open or closed; an open loop terminates with a functional requirement, which can be measured in the final assembly (it could be either the size of a relevant gap in the final assembly, or any other functionally relevant assembly dimension); a closed loop indicates the presence of one or more adjustable elements in the assembly.
5. *Derive the stack-up equations.* The assembly constraints defined within vector-loop-based models may be mathematically represented as a concatenation of homogeneous rigid body transformation matrices:

$$\mathbf{R}_1 \cdot \mathbf{T}_1 \cdot \dots \cdot \mathbf{R}_i \cdot \mathbf{T}_i \cdot \dots \cdot \mathbf{R}_n \cdot \mathbf{T}_n \cdot \mathbf{R}_f = \mathbf{H}, \quad (2.4)$$

where \mathbf{R}_1 is the rotational transformation matrix between the x -axis and the first vector; \mathbf{T}_1 is the translational matrix for the first vector, $\mathbf{R}_{i,n}$ and $\mathbf{T}_{i,n}$ are the corresponding matrices for the vector at node i or node n , and \mathbf{R}_f is the final closure rotation, again with respect to the x -axis. \mathbf{H} is the resultant matrix. For example, in the 2D case the rotational and the translational matrices are as follows:

$$\mathbf{R}_i = \begin{bmatrix} \cos \phi_i & \sin \phi_i & 0 \\ \sin \phi_i & \cos \phi_i & 0 \\ 0 & 0 & 1 \end{bmatrix} \text{ and } \mathbf{T}_i = \begin{bmatrix} 1 & 0 & L_i \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix},$$

where ϕ_i is the angle between the vectors at node i , and L_i is the length of vector i . If the assembly is described by a closed loop of constraints, \mathbf{H} is equal to the identity matrix, otherwise \mathbf{H} is equal to the \mathbf{g} vector representing the resultant transformation that will lead to the identification of a functional requirement.

6. *Tolerance analysis* – assuming an assembly as made of p parts. Each part is represented by an \mathbf{x} vector of its relevant dimensions and by an α vector containing additional dimensions, added to take into account geometric tolerances. When parts are assembled together, the resulting product is characterized by a \mathbf{u} vector of the assembly variables and by a \mathbf{g} vector of measurable functional requirements. It is possible to write $L=J-P+1$ closed loops, where J is the number of the mates among the parts and P is the number of parts. For each closed loop

$$\mathbf{H}(x, u, \alpha) = 0, \quad (2.5)$$

while there is an open loop for each functional requirement that looks like

$$\mathbf{g} = \mathbf{K}(x, u, \alpha). \quad (2.6)$$

Equation 2.5 allows one to calculate \mathbf{g} after having solved the system of equations in Equation 2.4. The equations in Equation 2.4 are usually not linear; they can be solved in different ways, for example, by means of the direct linearization method:

$$\mathbf{dH} \cong \mathbf{A} \cdot \mathbf{dx} + \mathbf{B} \cdot \mathbf{du} + \mathbf{F} \cdot \mathbf{d\alpha} = 0, \quad (2.7)$$

$$\mathbf{dH} \cong -\mathbf{B}^{-1} \cdot \mathbf{A} \cdot \mathbf{dx} - \mathbf{B}^{-1} \cdot \mathbf{F} \cdot \mathbf{d\alpha}, \quad (2.8)$$

$$\mathbf{dg} \cong \mathbf{C} \cdot \mathbf{dx} + \mathbf{D} \cdot \mathbf{du} + \mathbf{G} \cdot \mathbf{d\alpha} = 0, \quad (2.9)$$

with $\mathbf{A}_{ij} = \partial \mathbf{H}_i / \partial x_j$, $\mathbf{B}_{ij} = \partial \mathbf{H}_i / \partial u_j$, $\mathbf{F}_{ij} = \partial \mathbf{H}_i / \partial \alpha_j$, $\mathbf{C}_{ij} = \partial \mathbf{K}_i / \partial x_j$, $\mathbf{D}_{ij} = \partial \mathbf{K}_i / \partial u_j$, and $\mathbf{G}_{ij} = \partial \mathbf{K}_i / \partial \alpha_j$.

From Equations 2.7–2.9,

$$\begin{aligned} \mathbf{dg} &\cong [\mathbf{C} - \mathbf{D} \cdot \mathbf{B}^{-1} \cdot \mathbf{A}] \cdot \mathbf{dx} + [\mathbf{G} - \mathbf{D} \cdot \mathbf{B}^{-1} \cdot \mathbf{F}] \cdot \mathbf{d\alpha} \\ &= \mathbf{S}_x \cdot \mathbf{dx} + \mathbf{S}_\alpha \cdot \mathbf{d\alpha}, \end{aligned} \quad (2.10)$$

where $\mathbf{S}_x = [\mathbf{C} - \mathbf{D} \cdot \mathbf{B}^{-1} \cdot \mathbf{A}]$ and $\mathbf{S}_\alpha = [\mathbf{G} - \mathbf{D} \cdot \mathbf{B}^{-1} \cdot \mathbf{F}]$ are named the “sensitivity” matrices. When the sensitivity matrices are known, it is possible to calculate the solution in the worst-case scenario as

$$\Delta g_i = \sum_k |\mathbf{S}_{x_{ik}} \cdot \mathbf{t}_{x_k}| + \sum_l |\mathbf{S}_{\alpha_{il}} \cdot \mathbf{t}_{\alpha_l}|, \quad (2.11)$$

while in the statistical scenario the solution can be obtained as a root sum of squares, as follows:

$$\Delta g_i = \left[\sum_k (\mathbf{S}_{x_{ik}} \cdot \mathbf{t}_{x_k})^2 + \sum_l (\mathbf{S}_{\alpha_{il}} \cdot \mathbf{t}_{\alpha_l})^2 \right]^{1/2}, \quad (2.12)$$

where k and l are the number of x dimensions and α geometric tolerances that influence the variable g_i , $\mathbf{S}_{x_{ik}}$ is the matrix of the coefficients of the k x variables inside the i -stack-up function of Equation 2.10, $\mathbf{S}_{\alpha_{il}}$ is the matrix of the coefficients of the l α variables inside the i stack-up function of Equation 2.10, and \mathbf{t}_{x_k} and \mathbf{t}_{α_l} are the vectors of the dimensional or the geometric tolerances of the x_k and α_l variables, respectively.

The direct linearization method is a very simple and rapid method, but it is approximated too. When an approximated solution is not acceptable, it is possible to use alternative approaches, such as numerical simulation by means of a Monte Carlo technique (Gao *et al.* 1998; Boyer and Stewart 1991).

The second closed loop joins the subassembly box disk 1 and disk 2 through contact points D and C. The open loop defines the gap width g . All the loops are defined counterclockwise. The \mathbf{R} and \mathbf{T} matrices are 2D; their elements are shown in Table 2.1.

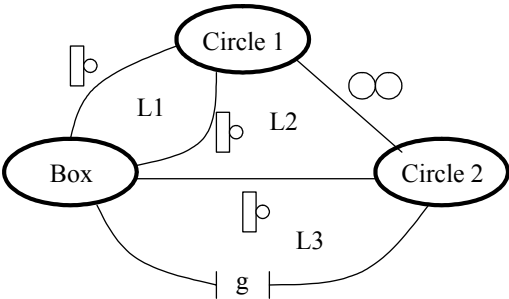


Figure 2.3 Assembly graph

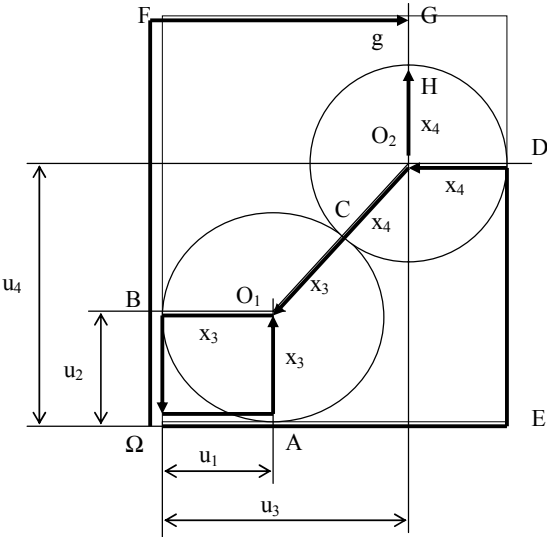


Figure 2.4 Datum paths of the vector loop model

Table 2.1 Elements of **R** and **T** matrices when the case study considers dimensional tolerances

Loop 1			Loop 2		Loop 3	
i	ϕ_i of R_i	L_i of T_i	ϕ_i of R_i	L_i of T_i	ϕ_i of R_i	L_i of T_i
1	0	u_1	0	x_1	0	x_1
2	90°	x_3	90°	u_4	90°	u_4
3	ϕ_{13}	x_3	90°	x_4	90°	x_4
4	90°	u_2	ϕ_{24}	x_4	ϕ_{34}	x_4
5	90°		0°	x_3	0°	g
6			ϕ_{26}	x_3	90°	u_3
7			90°	u_2	90°	x_2
8			90°		90°	

For the first loop, Equation 2.4 becomes

$$\mathbf{R}_1 \cdot \mathbf{T}_1 \cdot \mathbf{R}_2 \cdot \mathbf{T}_2 \cdot \mathbf{R}_3 \cdot \mathbf{T}_3 \cdot \mathbf{R}_4 \cdot \mathbf{T}_4 \cdot \mathbf{R}_f = \mathbf{I}, \quad (2.13)$$

which gives the system

$$\begin{aligned} u_1 + x_3 \cos(90 + \phi_{13}) + u_2 \cos(180 + \phi_{13}) &= 0, \\ x_3 + x_3 \sin(90 + \phi_{13}) + u_2 \sin(180 + \phi_{13}) &= 0, \\ \phi_{13} - 90 &= 0. \end{aligned} \quad (2.14)$$

For the second loop,

$$\mathbf{R}_1 \cdot \mathbf{T}_1 \cdot \mathbf{R}_2 \cdot \mathbf{T}_2 \cdot \mathbf{R}_3 \cdot \mathbf{T}_3 \cdot \mathbf{R}_4 \cdot \mathbf{T}_4 \cdot \mathbf{R}_5 \cdot \mathbf{T}_5 \cdot \mathbf{R}_6 \cdot \mathbf{T}_6 \cdot \mathbf{R}_7 \cdot \mathbf{T}_7 \cdot \mathbf{R}_f = \mathbf{I}, \quad (2.15)$$

which gives the system

$$\begin{aligned} x_1 - x_4 + x_4 \cos(180 + \phi_{24}) + x_3 \cos(180 + \phi_{24}) + \\ + x_3 \cos(180 + \phi_{24} + \phi_{26}) &= 0, \\ x_1 - x_4 + x_4 \cos(180 + \phi_{24}) + x_3 \cos(180 + \phi_{24}) + \\ + x_3 \cos(180 + \phi_{24} + \phi_{26}) &= 0, \\ \phi_{24} + \phi_{26} &= 0. \end{aligned} \quad (2.16)$$

For the third loop,

$$\mathbf{R}_1 \cdot \mathbf{T}_1 \cdot \mathbf{R}_2 \cdot \mathbf{T}_2 \cdot \mathbf{R}_3 \cdot \mathbf{T}_3 \cdot \mathbf{R}_4 \cdot \mathbf{T}_4 \cdot \mathbf{R}_5 \cdot \mathbf{T}_5 \cdot \mathbf{R}_6 \cdot \mathbf{T}_6 \cdot \mathbf{R}_7 \cdot \mathbf{T}_7 \cdot \mathbf{R}_f = \mathbf{G}, \quad (2.17)$$

which gives

$$g = x_2 - u_4 - x_4. \quad (2.18)$$

From the “*sensitivity*” analysis,

$$\mathbf{A} \cdot d\mathbf{x} + \mathbf{B} \cdot d\mathbf{u} = 0, \quad (2.19)$$

which gives

$$d\mathbf{u} = -\mathbf{B}^{-1} \cdot \mathbf{A} \cdot d\mathbf{x} = \mathbf{S}^u \cdot d\mathbf{x}, \quad (2.20)$$

where

$$\mathbf{du} = \{du_1, du_2, du_4, d\phi_{13}, d\phi_{24}, d\phi_{26}\}^T, \quad (2.21)$$

$$\mathbf{dx} = \{dx_1, dx_2, dx_3, dx_4\}^T = \{0.20, 0.50, 0.05, 0.05\}^T, \quad (2.22)$$

$$\mathbf{S}^u = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ -0.2582 & 0 & 2.2910 & 1.2910 \\ 0 & 0 & 0 & 0 \\ -0.0258 & 0 & 0.0323 & 0.0323 \\ 0.0258 & 0 & -0.0323 & -0.0323 \end{bmatrix}. \quad (2.23)$$

The gap g depends on the following x variables through the sensitivity coefficients:

$$\begin{aligned} dg &= dx_2 - dx_4 - du_4 = 0.2582 \cdot dx_1 + dx_2 - 2.2910 \cdot dx_3 \\ &\quad - 2.2910 \cdot dx_4. \end{aligned} \quad (2.24)$$

It is possible to calculate the solution in the worst case as

$$\Delta g_{WC} = \pm \sum |S_i| \cdot \Delta x_i = \pm 0.7807 \cong \pm 0.78 \text{ mm}. \quad (2.25)$$

The solution obtained is lower than the value obtained by means of the manual resolution method of about 4% [= (1.56–1.62)/1.62].

It is possible to calculate the solution in the statistical scenario (root sum of square) as

$$\Delta g_{Stat} = \pm \left[\sum (S_{x_{ik}} \cdot t_{x_k})^2 \right]^{1/2} = \pm 0.5158 \cong \pm 0.52 \text{ mm}. \quad (2.26)$$

2.3.2 Results of the Case Study with Geometric Tolerances

With reference to Figure 2.5, let x_1 and x_2 be the dimensions of the box, and x_3 and x_4 the diameters of the two disks; u_1 , u_2 , u_3 , and u_4 are the assembly (dependent) dimensions and g is the width of the gap between the top side of the box and the second disk. The DRFs and the datum paths are the same as in the previous case (see Section 2.3.1).

The vector loops are the same as in the previous case, but they have to take into consideration the geometric tolerances. To include geometric tolerances, the following variables must be added to the \mathbf{x} vector (note that mating points A and B are named after datums A and B the points lie upon):

- The flatness tolerance applied to the bottom surface of the box (datum A in the drawing) can be represented as a translation of the A point in the direction perpendicular to datum A, *i.e.*, perpendicular to the x -axis; this translation is described by the variable $\alpha_1 = T_{A1} = 0 \pm 0.10/2 = 0 \pm 0.05$ mm.
- The perpendicularity applied to the vertical left surface of the box (datum B) can be represented as a translation of point B in the direction perpendicular to datum B (the y -axis); this translation is described by the variable $\alpha_2 = T_{B2} = 0 \pm 0.10/2 = 0 \pm 0.05$ mm.
- The parallelism applied to the right side of the box (with respect to datum B) can be represented as a translation of point D, again in the direction perpendicular to datum B. It can be described by the variable $\alpha_3 = T_{D3} = 0 \pm 0.20/2 = 0 \pm 0.10$ mm.
- The circularity applied to disk 1 can be seen as points A, B, and C translating along the radius, and can be described by the variables $\alpha_4 = T_{A4} = 0 \pm 0.05/2 = 0 \pm 0.025$ mm, $\alpha_5 = T_{B4} = 0 \pm 0.05/2 = 0 \pm 0.025$ mm, and $\alpha_6 = T_{C4} = 0 \pm 0.05/2 = 0 \pm 0.025$ mm.
- The circularity applied to disk 2 can be represented as points C, D, and H translating along the radius, and can be described by the variables $\alpha_7 = T_{C5} = 0 \pm 0.05/2 = 0 \pm 0.025$ mm, $\alpha_8 = T_{D5} = 0 \pm 0.05/2 = 0 \pm 0.025$ mm, and $\alpha_9 = T_{H5} = 0 \pm 0.05/2 = 0 \pm 0.025$ mm.
- The parallelism applied to the top side of the box can be represented as a vertical translation of point G, and is described by the variable $\alpha_{10} = T_{G6} = 0 \pm 0.10/2 = 0 \pm 0.05$ mm.

The **R** and **T** matrices are 2D; their elements are shown in Table 2.2.

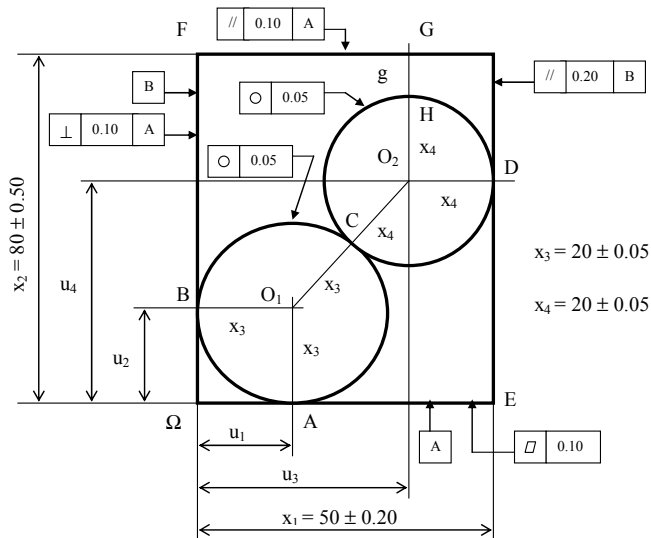


Figure 2.5 Assembly variables and tolerances of the vector loop model with geometric tolerances

Table 2.2 Elements of **R** and **T** matrices when the case study considers geometric tolerances

	Loop 1		Loop 2		Loop 3	
<i>i</i>	ϕ_i of R_i	L_i of T_i	ϕ_i of R_i	L_i of T_i	ϕ_i of R_i	L_i of T_i
1	0	u_1	0	x_1	0°	x_1
2	90°	$\alpha_1 = 0 \pm 0.05$	90°	u_4	90°	u_4
3	0°	$\alpha_4 = 0 \pm 0.025$	90°	$\alpha_3 = 0 \pm 0.1$	90°	$\alpha_3 = 0 \pm 0.1$
4	0°	x_3	0°	$\alpha_8 = 0 \pm 0.025$	0°	$\alpha_8 = 0 \pm 0.025$
5	ϕ_{13}	x_3	0°	x_4	0°	x_4
6	0°	$\alpha_5 = 0 \pm 0.025$	ϕ_{24}	x_4	ϕ_{34}	x_4
7	0°	$\alpha_2 = 0 \pm 0.05$	0°	$\alpha_7 = 0 \pm 0.025$	0°	$\alpha_9 = 0 \pm 0.025$
8	90°	u_2	0°	$\alpha_6 = 0 \pm 0.025$	0°	g
9	90°		0°	x_3	0°	$\alpha_{10} = 0 \pm 0.05$
10			ϕ_{26}	x_3		
11			0°	$\alpha_5 = 0 \pm 0.025$		
12			0°	$\alpha_2 = 0 \pm 0.05$		
13			90°	u_2		
14			90°			

Once the vector loops have been generated, the relative equations can be defined and solved. For the first loop, Equation 2.4 becomes

$$\mathbf{R}_1 \cdot \mathbf{T}_1 \cdot \mathbf{R}_2 \cdot \mathbf{T}_2 \cdot \mathbf{R}_3 \cdot \mathbf{T}_3 \cdot \mathbf{R}_4 \cdot \mathbf{T}_4 \cdot \mathbf{R}_5 \cdot \mathbf{T}_5 \cdot \mathbf{R}_6 \cdot \mathbf{T}_6 \cdot \mathbf{R}_7 \cdot \mathbf{T}_7 \cdot \mathbf{R}_8 \cdot \mathbf{T}_8 \cdot \mathbf{R}_f = \mathbf{I}, \quad (2.27)$$

which gives the system

$$\begin{aligned} u_1 + (x_3 + \alpha_2 + \alpha_5) \cdot \cos(90 + \varphi_{13}) + u_2 \cos(180 + \varphi_{13}) &= 0, \\ x_3 + \alpha_1 + \alpha_4 + (x_3 + \alpha_2 + \alpha_5) \cdot \sin(90 + \varphi_{13}) + u_2 \sin(180 + \varphi_{13}) &= 0, \\ \varphi_{13} - 90 &= 0. \end{aligned} \quad (2.28)$$

For the second loop,

$$\mathbf{R}_1 \cdot \mathbf{T}_1 \cdot \mathbf{R}_2 \cdot \mathbf{T}_2 \cdot \mathbf{R}_3 \cdot \mathbf{T}_3 \cdot \mathbf{R}_4 \cdot \mathbf{T}_4 \cdot \mathbf{R}_5 \cdot \mathbf{T}_5 \cdot \mathbf{R}_6 \cdot \mathbf{T}_6 \cdot \dots \cdot \mathbf{R}_{13} \cdot \mathbf{T}_{13} \cdot \mathbf{R}_f = \mathbf{I}, \quad (2.29)$$

which gives the system

$$\begin{aligned} x_1 - (x_4 + \alpha_3 + \alpha_8) + (x_3 + x_4 + \alpha_6 + \alpha_7) \cdot \cos(180 + \varphi_{24}) + \\ (x_3 + \alpha_2 + \alpha_5) \cdot \cos(180 + \varphi_{24} + \varphi_{26}) &= 0, \\ u_4 + (x_3 + x_4 + \alpha_6 + \alpha_7) \cdot \sin(180 + \varphi_{24}) + (x_3 + \alpha_2 + \alpha_5) \cdot \\ \sin(180 + \varphi_{24} + \varphi_{26}) - u_2 &= 0, \\ \varphi_{24} + \varphi_{26} &= 0. \end{aligned} \quad (2.30)$$

For the third loop,

$$\begin{aligned} & \mathbf{R}_1 \cdot \mathbf{T}_1 \cdot \mathbf{R}_2 \cdot \mathbf{T}_2 \cdot \mathbf{R}_3 \cdot \mathbf{T}_3 \cdot \mathbf{R}_4 \cdot \mathbf{T}_4 \cdot \mathbf{R}_5 \cdot \mathbf{T}_5 \cdot \mathbf{R}_6 \cdot \mathbf{T}_6 \cdot \dots \cdot \mathbf{R}_{11} \cdot \mathbf{T}_{11} \cdot \\ & \mathbf{R}_f = \mathbf{G}, \end{aligned} \quad (2.31)$$

which gives

$$g = x_2 + \alpha_{10} - u_4 - x_4 - \alpha_9. \quad (2.32)$$

Concerning the “*sensitivity*” analysis,

$$\mathbf{du} = -\mathbf{B}^{-1} \cdot \mathbf{A} \cdot \mathbf{dx} - \mathbf{B}^{-1} \cdot \mathbf{C} \cdot \mathbf{d\alpha} = \mathbf{S}^{ud} \cdot \mathbf{dx} + \mathbf{S}^{u\alpha} \cdot \mathbf{d\alpha}, \quad (2.33)$$

where

$$\mathbf{du} = \{du_1, du_2, du_4, d\varphi_{13}, d\varphi_{24}, d\varphi_{26}\}^T, \quad (2.34)$$

$$\mathbf{dx} = \{dx_1, dx_2, dx_3, dx_4\}^T = \{0.20, 0.50, 0.05, 0.05\}^T, \quad (2.35)$$

$$\mathbf{d\alpha} = \{d\alpha_1, \dots, d\alpha_{10}\}^T = \left\{ \begin{matrix} 0.05, 0.05, 0.10, 0.025, 0.025, 0.025, \\ 0.025, 0.025, 0.025, 0.05 \end{matrix} \right\}^T, \quad (2.36)$$

$$\mathbf{S}^{ud} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ -0.2582 & 0 & 2.2910 & 1.2910 \\ 0 & 0 & 0 & 0 \\ -0.0258 & 0 & 0.0323 & 0.0323 \\ 0.0258 & 0 & -0.0323 & -0.0323 \end{bmatrix}, \quad (2.37)$$

$$\mathbf{S}^{u\alpha} = \begin{bmatrix} 0 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0.2582 & 0 & 0.0258 & -0.0258 \\ 0 & 0 & 0.2582 & 0 & 0.0258 & -0.0258 \\ 0 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0.2582 & 0 & 0.0258 & -0.0258 \\ 0 & 0 & 1.0328 & 0 & 0.0064 & -0.0064 \\ 0 & 0 & 1.0328 & 0 & 0.0064 & -0.0064 \\ 0 & 0 & 0.2582 & 0 & 0.0258 & -0.0258 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T. \quad (2.38)$$

The solution shows that the variability of the gap width g can be represented by the following function of the \mathbf{x} vector:

$$\begin{aligned} dg = & dx_2 + d\alpha_{10} - dx_4 - du_4 - d\alpha_9 = 0.2582 \cdot dx_1 + dx_2 \\ & - 2.2910 \cdot dx_3 - 2.2910 \cdot dx_4 - d\alpha_1 - 0.2582 \cdot d\alpha_2 - 0.2582 \cdot d\alpha_3 \\ & - d\alpha_4 - 0.2582 \cdot d\alpha_5 - 1.0328 \cdot d\alpha_6 - 1.0328 \cdot d\alpha_7 - 0.2582 \cdot d\alpha_8 \\ & - d\alpha_9 + d\alpha_{10} \end{aligned} \quad (2.39)$$

It is possible to compute the solution with the worst-case approach:

$$\Delta g_{WC} = \pm \left(\sum |S_i| \cdot \Delta x_i + \sum |S_j| \cdot \Delta \alpha_j \right) = \pm 1.0340 \equiv \pm 1.03 \text{ mm}. \quad (2.40)$$

It is also possible to compute the solution with the statistical approach (root sum of squares):

$$\Delta g_{Stat} = \pm \left[\sum \left(S_{x_{ik}} \cdot t_{x_k} \right)^2 + \sum \left(S_{\alpha_{ij}} \cdot \Delta \alpha_j \right)^2 \right]^{1/2} = \pm 0.5361 \equiv \pm 0.54 \text{ mm}. \quad (2.41)$$

2.4 Further Geometric Tolerance Analysis Models

2.4.1 The Variational Model

A mathematical foundation of this model was proposed first by Boyer and Stewart (1991), and then by Gupta and Turner (1993). Later, several additional variants were proposed as well, and nowadays commercial computer aided tolerancing (CAT) software packages are based on this approach, such as eM-TolMate from UGS[®], 3-DCS from Dimensional Control Systems[®], and VisVSA from UGS[®].

The basic idea of the variational model is to represent the variability of an assembly, due to tolerances and assembly conditions, through a parametric mathematical model.

To create an assembly, the designer must define the nominal shape and the dimensions of each assembly component (this information is usually retrieved from CAD files). Then, the designer identifies the relevant features of each component and assigns dimensional and geometric tolerances to them. Each feature has its local DRF, while each component and the whole assembly have their own global DRF. In nominal conditions, a homogeneous transformation matrix (called **TN**) is defined that identifies the position of the feature DRF with respect to the part DRF. In real conditions (*i.e.*, manufactured part), the feature will be characterized by a roto-translational displacement with respect to its nominal position. This displacement is modeled to summarize the complete effects of the dimensional and geometric variations affecting the part by means of another matrix: the differential homogeneous transformation matrix (called **DT**). The variational model may take into account the precedence among the datums by setting the parameters of the **DT** matrix.

The variational model is not able to deal with the form tolerances such as the vector loop model does; this means that the actual feature shape is assumed unchanged, *i.e.*, feature shape variations are neglected. The position of the displaced feature in the part DRF can be simply obtained by matrix multiplication as a change of DRF.

The model is parametric because different types and amounts of variations can be modeled by simply altering the contents (parameters) of the **DT** matrix. In some cases, the localization of a feature affected by a variation may be defined by a transformation with respect to another feature in the same part which is affected by variations as well. Therefore, the material modifier condition is modeled by setting the parameters of the **DT** matrix.

Once the variabilities of the parts have been modeled, they must be assembled together. Another set of differential homogeneous transformation matrices is introduced to handle the roto-translational deviations introduced by each assembly mating relation. Such matrices are named **DA**, with the letter *A* (for “assembly”) to distinguish them from the matrices that have been used for parts. Those matrices are hard to evaluate, since they depend on both the tolerances imposed on the parts in contact and the assembly conditions. This model is not able to represent mating conditions with clearance. The problem of evaluating the differential matrix is analyzed in several literature works. A possible strategy consists in modeling the joint between the coupled parts by reconstructing the coupling sequence between the features (Berman 2005). Another possibility is to impose some analytical constraints on the assembly parameters (Whitney 2004).

When all the transformation matrices have been obtained, it is possible to express all the features in the same global DRF of the assembly. Finally, the functional requirements can be modeled in the form of functions, as follows:

$$FR = f(p_1, p_2, \dots, p_n), \quad (2.42)$$

where FR is the assembly functional requirement, p_1, \dots, p_n are the model parameters, and $f(p)$ is the stack-up function (usually not linear) obtained from the matrix multiplications described above. This model may be applied to assemblies involving joints which make a linear structure among the parts (linear stack-up function, see Figure 2.6a) and joints which make a complex structure among the parts (networks of stack-up functions, see Figure 2.6b), such as a vector loop does.

Once the stack-up functions have been modeled, there are two approaches to solve them: the worst-case approach and the statistical approach. The worst-case analysis consists in identifying the extreme configurations of the assembly under a given set of tolerances. In the variational approach, the problem is generally handled as an optimization (maximization and/or minimization) problem, under constraints defined by the tolerances themselves. The statistical approach is generally handled by assigning predefined probability density functions, *e.g.*, Gaussian, to the parameters identifying the main elements that contribute to the variation of each feature (often assumed independent, by simplification), and then solving the stack-up functions accordingly (Salomons *et al.* 1996).

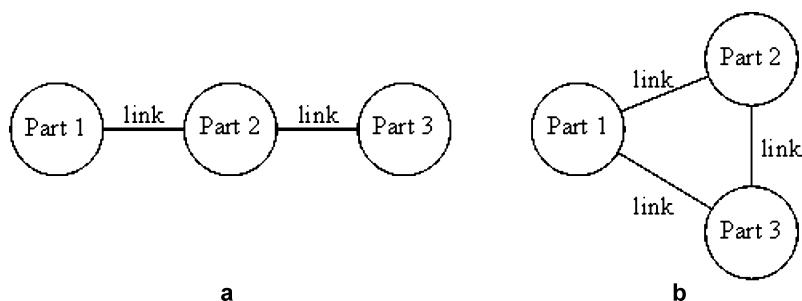


Figure 2.6 **a** Linear stack-up function, and **b** network stack-up function

To better illustrate the variational method, its basic steps are illustrated in the following:

1. *Create the assembly graph.* The first step is to create an assembly graph. The assembly graph is a simplified diagram of the assembly representing the parts, the features, the mating conditions, and the functional requirements.
2. *Define the DRF of each feature, of each part, and of the assembly.* The next step is to identify the local DRF of each feature and the global DRF of each part and of the assembly (usually the DRF of the assembly coincides with the DRF of the first part). DRFs are positioned depending on the surface type; from the DRFs, local parameters and the differential homogeneous transformation matrices \mathbf{DT} are defined.
3. *Transform the features.* Once the transformation matrices are known, each feature of a part is transformed into the global DRF of the part.
4. *Create the assembly.* With use of the assembly graph and the transformed features, the assembly conditions are extracted, *i.e.*, the assembly parameters included in the matrix \mathbf{DA} are calculated.
5. *Derive the equations of the functional requirements.* Once the assembly parameters are known, all the features can be expressed in the same global DRF of the assembly. At this point, the functional requirements are defined in terms of functions that can be solved by means of the previously described worst-case and/or statistical approaches.

2.4.2 The Matrix Model

Instead of deriving equations that model a specific displacement of a part or assembly as a function of given set of geometric dimensions (parameters) assuming specific values within the boundaries defined by tolerances (like in the variational approach), the matrix model aims at deriving an explicit mathematical representation of the boundary of the entire spatial region that encloses all possible displacements due to one or more variability sources. In order to do that, homogene-

ous transformation matrices are again considered as the foundation of the mathematical representation. A displacement matrix **DT** is used to describe any roto-translational variation a feature may be subjected to; the matrix is defined with respect to a local DRF. Since the goal is to represent the boundaries of the region of possible variations (*i.e.*, extreme values), the approach is intrinsically a worst-case approach. No statistical approach may be implemented, such as vector loop and variational models do. To represent boundaries, constraints must be added to the displacements modeled within the **DT** matrices. Displacement boundaries resulting from complex series of tolerances are solved by modeling the effects of each tolerance separately and by combining the resulting regions. Analogously, gaps/clearances are represented as if they were tolerance regions. Finally, by classifying the surfaces into several classes, each characterized by some type of invariance with respect to specific displacement types (*e.g.*, a cylinder is invariant to any rotation about its axis), one can simplify displacements and the resulting displacement matrix (Clément *et al.* 1994).

A similar approach is followed to model the dimensions acting as functional requirements of the assembly; since in this case the resulting region (of possible values) is essentially contained in a segment, segment boundaries must be computed by means of a worst-case approach (minimum–maximum distances between the two points). The two points defining the boundaries of the segment must be defined as the result of stack-up functions (Desrochers and Rivière 1997).

The matrix model is based on the positional tolerancing and the technologically and topologically related surfaces (TTRS) criteria (Clément *et al.* 1998). Geometric features are assumed as ideal, *i.e.*, the form tolerances are neglected, such as in the variational model. To better understand the matrix method for tolerance analysis, its basic steps are provided below:

1. *Transform the tolerances applied to the drawing.* The first step is to transform the tolerances applied to the drawing to make them compliant with the positional tolerancing and the TTRS criteria.
2. *Create the assembly graph.* The second step is to create an assembly graph. The assembly graph allows for identification of the global DRF and the linkages among the features to which the tolerances are assigned. The assembly parts should be in contact; the joints with clearance may not be considered.
3. *Define the local DRF of each part feature.* A DRF must be assigned to each part feature.
4. *Identify the measurable points for each functional requirement.* Points that locate the boundaries of each functional requirement must be identified and the path that connects them to the global DRF must be defined, taking into account all the tolerances stacking up along the way.
5. *Define the contributions of each single displacement and the related constraints.* It is necessary to define the contribution of each displacement to the total displacement region, and the constraints necessary to identify its boundaries. Each surface can be classified into one of the seven classes of invariant surfaces; this allows one to discard some displacements and to obtain a simpli-

fied displacement matrix. Additional information is necessary to specify the constraints ensuring that the feature remains inside the boundaries of the tolerance zone.

6. *Apply the superimposition principle and run the optimization.* If more than one tolerance is applied to the same part, the total effect is computed through the superimposition principle. For example, if n tolerances are applied to the same feature, in the local DRF, the displacement of a generic point belonging to the feature is simply defined as a sum of the single contributions. The aggregation of expressions obtained for each toleranced feature results in a constrained optimization problem, which can be solved with known, standard approaches. This model has been developed for assemblies involving joints which make a linear structure among the parts (linear stack-up function), while it is not able to deal with joints which make a complex structure among the parts (network stack-up function). The worst-case approach may be applied to the matrix model, since the statistical one has not been developed yet.

All the details of the model are described in depth in Marziale and Polini (2009b).

2.4.3 The Jacobian Model

In the terminology adopted by the Jacobian model approach, any relevant surface involved in the tolerance stack-up is referred to as a *functional element*. In the tolerance chain, functional elements are considered in pairs: the two paired surfaces may belong to the same part (internal pair), or to two different parts, and are paired since they interact as mating elements (kinematic pair, also referred to as an external pair). The parts should be in contact to be modeled by this model.

Transformation matrices can be used to locate a functional element of a pair with respect to the other: such matrices can be used to model the nominal displacement between the two functional elements, but also additional small displacements due to the variabilities modeled by the tolerances. The form tolerances are neglected. The main peculiar aspect of the *Jacobian* approach is how such matrices are formulated, *i.e.*, by means of an approach derived from the description of kinematic chains in robotics. The transformation that links two functional elements belonging to a pair, and that includes both nominal displacement and small deviations due to tolerances, can be modeled by a set of six virtual joints, each associated with a DRF. Each virtual joint is oriented so that a functional element may have either a translation or a rotation along its z -axis. The aggregation of the six virtual joints gives rise to the transformation matrix linking one functional element to the other functional element of the pair (Laperrière and Lafond 1999; Laperrière and Kabore 2001). The position of a point lying on the second functional element of a pair, which may be assumed as depicting the functional requirement under scrutiny, with respect to the DRF of the first functional

element (assumed as the global DRF) may be expressed by considering the three small translations and the three small rotations of the point in the global DRF through the product of a Jacobian matrix associated with the functional element with tolerances of all the functional element pairs involved (internal or kinematic) and a vector of small deviations associated with the functional element with tolerances of all the functional element pairs involved, expressed in the local DRF. The main element of the expression is the Jacobian matrix, which is relatively easy to compute, starting from the nominal position of the geometric elements involved. The tricky part, however, is to turn the assembly tolerances into displacements to assign them to the virtual joints defined for each functional element pair in the chain.

The main steps of the approach are described below:

1. *Identify the functional element pairs.* The first step is the identification of the functional element pairs (*i.e.*, pairs of relevant surfaces). The functional elements are arranged in consecutive pairs to form a stack-up function aimed at computing each functional requirement.
2. *Define the DRF for each functional element and the virtual joints.* The next step is to define a DRF for each functional element, and to create the chain of virtual joints representing the transformation that links the pair of functional elements. Once such information is available, the transformation matrix for each functional element can be obtained.
3. *Create the chain and obtain the overall Jacobian matrix.* The transformation matrices can be chained to obtain the stack-up function needed to evaluate each functional requirement. This model has been developed for assemblies involving joints which make a linear structure among the parts (linear stack-up function), while it is not able to deal with joints which make a complex structure among the parts (network of stack-up functions), such as the matrix model does.
4. Once the required stack-up function has been obtained, it may be solved by the usual methods in the literature (Salomons *et al.* 1996) for the worst-case or statistical approaches.
5. Finally, it is necessary to observe that this model is based on the TTRS criterion (Clément *et al.* 1998) and on the positional tolerancing criterion (Legoff *et al.* 1999). Therefore, the tolerances of a generic drawing need to be converted in accordance with the previously defined criteria, before carrying out the tolerance analysis.

2.4.4 The Torsor Model

The torsor model uses screw parameters to model 3D tolerance zones (Chase *et al.* 1996). Screw parameters are a common approach adopted in kinematics to describe motion, and since a tolerance zone can be seen as the region where a sur-

face is allowed to move, screw parameters can be used to describe it. Each real surface of a part is modeled by a substitution surface. A substitution surface is a nominal surface characterized by a set of screw parameters that model the deviations from the nominal geometry due to the applied tolerances. Seven types of tolerance zones are defined. Each one is identified by a subset of nonzero screw parameters, while the remaining ones are set to zero as they leave the surface invariant. The screw parameters are arranged in a particular mathematical operator called a *torsor*, hence the name of the approach. Considering a generic surface, if u_A , v_A , and w_A are the translation components of its point A, and α , β , and γ are the rotation angles (considered small) with respect to the nominal geometry, the corresponding torsor is

$$\mathbf{T}_A = \left\{ \begin{matrix} \alpha & u_A \\ \beta & v_A \\ \gamma & w_A \end{matrix} \right\}_{\mathbf{R}}, \quad (2.43)$$

where \mathbf{R} is the DRF that is used to evaluate the screw components.

To model the interactions between the parts of an assembly, three types of torsors (or small displacement torsor, SDT) are defined (Ballot and Bourdet 1997): a *part SDT* for each part of the assembly to model the displacement of the part; a *deviation SDT* for each surface of each part to model the geometric deviations from the nominal geometry; a *gap SDT* between two surfaces linking two parts to model the mating relation. The form tolerances are neglected and they are not included in the deviation SDT.

A union of SDTs is used to obtain the global behavior of the assembly. The aggregation can be done by considering that the worst-case approach computes the cumulative effect of a linear stack-up function of n elements by adding the single components of the torsors. This is not true for a network of stack-up functions, which has not been developed by the torsor model yet. The torsor method does not allow one to apply a statistical approach, since the torsor's components are intervals of the small displacements; they are not parameters to which it is possible to assign easily a probability density function.

The torsor model operates under the assumption that the TTRS and the positional tolerancing criteria are adopted, which means that the tolerances in the drawing may need to be updated before carrying out the tolerance analysis. The solution of stack-up functions arranged in a network has not been completely developed. Finally, it is worth pointing out that, in the relevant literature, the use of SDTs for modeling tolerance analysis problems tends to follow two main approaches: on one hand, SDTs are used to develop functions for computing the position of geometric elements (belonging to the assembly) as they are subjected to displacement allowed by tolerances (e.g., see Chase *et al.* 1996); on the other hand, SDTs are used to model entire spatial volumes that encapsulate all the possible points in space that may be occupied by geometric elements during their variations (e.g., see Laperrière *et al.* 2002). In the analysis of the case study, only the second approach has been considered, since it looks more promising.

The basic steps of the torsor model are described in the following (Villeneuve *et al.* 2001; Teissandier *et al.* 1999):

1. *Identify the relevant surfaces of each part and the relations among them.* The first step is to identify the relevant surfaces belonging to each part and the relationships among them; this information is usually collected in a *surfaces graph*. In this step the chains to relate the functional requirements to the relevant surfaces are identified.
2. *Derive the SDTs.* A deviation SDT needs to be associated with each relevant surface of each part. This leads to the evaluation of a global SDT for each part. Finally, the shape of the gap SDT is associated with each joint according to the functional conditions of the assembly.
3. *Obtain the functional requirement stack-up functions.* Compute the cumulative effects of the displacements and obtain the final linear stack-up function of each functional requirement.

2.5 Comparison of the Models

A first comparison of the previously described five models can be done by devising a set of indicators describing features, capabilities, and issues related to the application of such models to given tolerance stack-up problems. The indicators and their results for the five models are summarized in Table 2.3. The indicators were designed by drawing inspiration from what is available from the literature (Salomons *et al.* 1996), with the necessary adaptations. Each descriptor may assume one of the following three states: “X” if the model has a property, or it is capable of handling a specific aspect or issue of the problem, “–” if the same property is missing, or the model is not able to handle the aspect of the problem; “?” if the answer is uncertain, because it may depend on the specific tolerance analysis problem, or because there is not enough information to verify the capability. The first descriptor is the “analysis type”, which refers to the type of approach that can be adopted to solve the stack-up functions, *i.e.*, worst-case or statistical. The descriptor “tolerance type” indicates the kind of tolerance that the model may take into account: dimensional, form, or other geometric (no form) tolerances. The “envelope and independence” descriptors refers to the possibility of the model representing a dimensional tolerance when the envelope principle or the independence principle is specified. The “parameters from tolerances” descriptor indicates whether the model allows for translation of the applied tolerance ranges into the model parameter ranges. The “tolerance stack-up type” descriptor refers to the possibility of a model building and solving linear stack-up functions or networks of stack-up functions. The “joint type” descriptor refers to the joint types that the model may take into account, either with contact between the surfaces or with clearance. The “functional requirement schematization” descriptor refers to how a functional requirement can be represented by a feature or by a set of points belonging to a feature. The “tolerance zones interaction” descriptor indicates the capability of representing the interaction

The *Jacobian* model has the advantage that the Jacobian matrix can be easily calculated from nominal conditions, while displacements of the functional requirements can be directly related to displacements of the virtual joints; however, it is difficult to derive such virtual joint displacements from the tolerances applied to the assembly components. On the other hand, the *torsor* model may allow for an easy evaluation of the ranges of the small displacements directly from the tolerances applied to the assembly components, but then it is very difficult to relate these ranges to the ranges of the functional requirements of the assembly.

These two considerations have suggested the idea of a *unified Jacobian–torsor model* to evaluate the displacements of the virtual joints from the tolerances applied to the assembly components through the torsors and, then, to relate the displacements of the functional requirements to the virtual joint displacements through the Jacobian matrix (Laperrière *et al.* 2002; Desrochers *et al.* 2003). Although this is theoretically possible, since the deviations are usually small and, therefore, the equations can be linearized, the actual feasibility of this approach is still the subject of research.

Finally, the models considered have some common limitations. The first deals with the envelope rule: the models do not allow one to apply the envelope rule and the independence rule to different tolerances of the same part. The second is that there do not exist any criteria to assign a probability density function to the model parameters joined to the applied tolerances and that considers the interaction among the tolerance zones. The last point deals with the assembly cycle: the models are not able to represent all the types of coupling with clearance between two parts.

The solution of the case study by the five models considered is described in detail in Marziale and Polini (2009b). Table 2.4 summarizes the results for the functional requirement Δg as obtained by the application of the five models and compared with the solution obtained with manual computation when only dimensional

Table 2.4 Results of the comparison among the models applied to the case study with dimensional tolerances

Model	Type of analysis	Results (mm)
Exact solution	Worst case	+0.84
		−0.78
Vector loop	Worst case	±0.78
	Statistical case	±0.52
Variational	Worst case	±0.78
	Statistical case	±0.51
Matrix	Worst case	±0.70
	Statistical case	—
Jacobian	Worst case	±0.78
	Statistical case	±0.53
Torsor	Worst case	±0.78
	Statistical case	—

tolerances are considered. Table 2.5 shows the results when both dimensional and geometric tolerances are applied.

The results obtained by considering only the dimensional tolerances show that all the models give slightly underestimated results with the worst-case approach, when compared with the results obtained manually with the approach described earlier. The *matrix* model has the highest error (−14%), while all the other models provide the same result (−4%). This is probably due to the way the dimensional tolerances are schematized (*i.e.*, the first datum is nominal, the variability due to the dimensional tolerance is considered to be applied only on one of the two features delimiting the dimension). Moreover, the statistical approach gives similar results for all the models considered.

The results obtained by considering both dimensional and geometric tolerances show that all the models, except the *vector loop* model, give similar results with the worst-case approach. This is probably due to the fact that the *vector loop* model considers the effect of a set of tolerances applied to a surface as the sum of the effects due to each single tolerance applied to the same surface. The effects of the different tolerances are considered to be independent. Therefore, increasing the number of tolerances applied to the same surface increases the variability of the functional requirement. This means that the interaction among tolerances defined on the same surface are not properly handled.

All five models produce very similar results when the statistical approach is applied.

Moreover, the results in Tables 2.4 and 2.5 obtained from the *Jacobian* model and from the *torsor* model are basically identical. This is due to the fact that a simplification was adopted when modeling the problem, *i.e.*, the angles of the box were considered fixed at 90°. This assumption is due to the need to avoid the networks of stack-up functions that the two models are not able to deal with. It means that all the tolerances applied may involve only translations of the sides of the box.

Table 2.5 Results of the comparison among the models applied to the case study with dimensional and geometric tolerances

Model	Type of analysis	Results (mm)
Vector loop	Worst case	± 1.03
	Statistical case	± 0.54
Variational	Worst case	± 0.78
	Statistical case	± 0.50
Matrix	Worst case	± 0.69
	Statistical case	—
Jacobian	Worst case	± 0.78
	Statistical case	± 0.53
Torsor	Worst case	± 0.78
	Statistical case	—

2.6 Guidelines for the Development of a New Tolerance Analysis Model

None of the models proposed in the literature provide a complete and clear mechanism for handling all the requirements included in the tolerancing standards (Shen *et al.* 2004). This limitation is reflected also in the available commercial CAT software applications, which are based on the same models (Prisco and Giorleo 2002). As already discussed in detail in previous work (Marziale and Polini 2009b), the main limitations of the actual models are the following: they do not properly support the application of the envelope rule and of the independence rule to different dimensional tolerances on the same part as prescribed by the ISO and ASME standards; they do not handle form tolerances (except for the vector loop model); they do not provide mechanisms for assigning probability density functions to model parameters starting from tolerances and considering tolerance zone interactions; finally, they are not capable of representing all the possible types of part couplings that may include clearance.

Some guidelines are now presented, aimed at the development of a new model that addresses at least some of the limitations highlighted above.

A dimensional tolerance assigned to the distance between two features of a part or of an assembly (Figure 2.7) may be required with the application of the envelope rule or of the independence principle. In the second case, to correctly define the relationship between the features, it is necessary to add a geometric tolerance in order to limit the form and the orientation deviations. The envelope principle states that when a feature is produced at its maximum material condition (MMC), the feature must have a perfect form. The MMC of a feature is the size at which the most material is in the part. The MMC size establishes a perfect form boundary and no part of the feature must extend outside this boundary. As the size of a feature departs from the MMC, its form is permitted to vary. Any amount of variation is permitted, as long as the perfect form boundary is not violated. However, the size limits must not be violated either. Therefore, if the envelope principle is applied to a dimensional tolerance, an additional constraint has to be considered to build and to solve the stack-up functions of the assembly. Both the ASME Y14.5M (1994) and the ISO 8015 (1985) standards foresee the possibility that, also on the same part, dimensional tolerances may be assigned with or without the application of the envelope principle. Consequently the mathematical model that is used to schematize a dimensional tolerance in order to build and to solve the stack-up functions should necessarily take into consideration these two possibilities. To overcome this limitation of the models in the literature that are not able to consider these two cases (Desrochers *et al.* 2003), a possible solution is to consider a greater set of parameters to model the degrees of freedom of the planes delimiting the dimension considered due to the applied tolerances (dimensional, orientation, form). The envelope rule and the independence rule constrain those parameters in different ways.

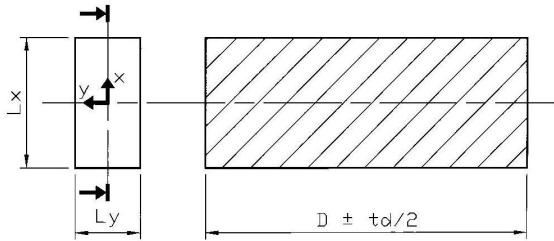


Figure 2.7 Two-dimensional dimensional tolerance

To model the form tolerance, it is possible to introduce a virtual transformation that is assigned to points of the surface to which a form tolerance is assigned. This approach was introduced by Chase *et al.* (1996) in their vector loop method for tolerance analysis.

In a statistical approach, a probability density function is assigned to each model parameter. Therefore, the tolerance analysis model has to determine the probability density function of each parameter according to the interaction of the tolerance zones. To do this, a possible solution is to decompose the possible deviation of a feature into different contributions – the dimensional, the form, the position, and the orientation ones – whose thickness is described by a model parameter. Each parameter may be considered independent of the others and it may be simulated by a probability density function which may be modeled by a Gaussian probability density function with a standard deviation equal to one sixth of the corresponding tolerance range. If more than one tolerance is applied to the same feature, the sum of the squares of the ranges of the applied tolerances is equal to the square of the range of the overall tolerance. For example, if the envelope principle is applied, the overall tolerance is the dimensional tolerance applied to the feature.

When two parts are assembled together, the mating surfaces form a joint. If the joint involves clearance, the clearance affects only some of the six small kinematic adjustments that define the position of a part with respect to the other. To evaluate the model parameters of the joint, it must be observed that they depend on the cumulative effects of the assembly constraints that must be satisfied by the coupled surfaces of the joint. The admissible values of the model parameters must be considered for each assembly constraint; therefore, all the constraints have to be considered and the resulting admissible values of the model parameters may be calculated as the intersection of the values previously defined. If the calculated domain of the admissible values of the model parameters is empty, the assembly is not possible. If the admissible domain contains a set of points, the convex hull, representing the boundary that encloses the points, may be determined. Once the boundary has been evaluated, it can be used in the tolerance analysis model as an additional constraint acting on the model parameters in the worst-case approach. It can be used to define the range of the probability density function that is assigned to the model parameters in the statistical approach (Marziale and Polini 2009a).

2.7 Conclusions

In this work, five different models available from the literature on tolerance analysis were compared through their application to a case study. None of the models proposed in the literature provide a complete and clear mechanism for handling all the requirements included in the tolerancing standards, and this limitation is reflected also in the available commercial CAT software. The main limitations include the following: no proper support for the application of the envelope rule and of the independence rule; cannot handle form tolerances (except for the vector loop model); no mechanisms for assigning probability density functions to model parameters starting from tolerances and considering tolerance zone interactions; no proper representation of all the possible types of part couplings that include clearance.

Guidelines were presented for the development of a new model aimed at addressing such highlighted limitations. Some suggestions were given to consider dimensional tolerance with the application of both the envelope principle and the independence principle, to take into account the real features and the interaction of the tolerance zones, to consider joints with clearance among the assembly components, and to adopt both the worst-case and the statistical approaches to solve the stack-up functions. The implementation of those suggestions in a new model and its application to case studies is the subject of ongoing research.

Acknowledgements This research was partially funded by the Italian Ministry of University and Research grant PRIN 2005 – “Characterization of the multiscale geometry of manufactured surfaces for process control and tolerance analysis”.

References

- ASME Y14.5M (1994) Dimensioning and tolerancing. American Society of Mechanical Engineering, New York
- Ballot E, Bourdet P (1995) Geometrical behaviour laws for computer aided tolerancing. In: Proceedings of the 4th CIRP seminar on computer aided tolerancing, University of Tokyo, April 1995
- Ballot E, Bourdet P (1997) A computational method for the consequences of geometric errors in mechanisms. In: Proceedings of the 5th CIRP seminar on computer aided tolerancing, Toronto, 27–29 April 1997
- Berman YO (2005) Shape and position uncertainty in mechanical assembly. PhD thesis, The Hebrew University, Jerusalem
- Boyer M, Stewart NF (1991) Modeling spaces for toleranced objects. *Int J Robot Res* 10:470–582
- Chase KW (1999) Multi-dimensional tolerance analysis (automated method). In: Drake PJR (ed) *Dimensioning and tolerancing handbook*. McGraw-Hill, New York
- Chase KW, Gao J, Magleby SP (1995) General 2-D tolerance analysis of mechanical assemblies with small kinematic adjustments. *J Des Manuf* 5:263–274
- Chase KW, Gao J, Magleby SP et al (1996) Including geometric feature variations in tolerance analysis of mechanical assemblies. *IIE Trans* 28:795–807
- Chase KW, Gao J, Magleby SP (1997) Tolerance analysis of 2- and 3D mechanical assemblies with small kinematic adjustments. In: Zhang HC (ed) *Advanced tolerancing techniques*. Wiley, New York

- Clément A, Rivière A, Temmerman M (1994) Cotation tridimensionnelle des systèmes mécaniques, théorie & pratique. Cachan, France
- Clément A, Rivière A, Serré P et al (1998) The TTRSs: 13 constraints for dimensioning and tolerancing. In: ElMaraghy HA (ed) Geometric design tolerancing: theories, standards and applications. Chapman & Hall, London
- Delchambre A (1996) CAD method for industrial assembly. Concurrent design of product, equipment and control systems. Wiley, New York
- Desrochers A, Rivière A (1997) A matrix approach to the representation of tolerance zones and clearances. *Int J Adv Manuf Technol* 13:630–636
- Desrochers A, Ghie W, Laperrière L (2003) Application of a unified Jacobian-torsor model for tolerance analysis. *J Comput Inf Sci Eng* 3:2–14
- Faerber PJ (1999) Tolerance analysis of assemblies using kinematically derived sensitivities. ADCATS report no. 99-3. <http://adcats.et.byu.edu/reportsandpublications.php>
- Gao J, Chase KW, Magleby SP (1998) Generalized 3-D tolerance analysis of mechanical assemblies with small kinematic adjustments. *IIE Trans* 30:367–377
- Gupta S, Turner JU (1993) Variational solid modelling for tolerance analysis. *IEEE Comput Graph Appl* 13:64–74
- Hong YS, Chang TC (2002) A comprehensive review of tolerancing research. *Int J Prod Res* 40:2425–2459
- ISO 8015 (1985) Fundamental tolerancing principle. International Organization for Standardization, Geneva
- Laperrière L, Lafond P (1999) Modelling tolerances and dispersions of mechanical assemblies using virtual joints. In: Proceedings of ASME design engineering technical conferences, September 12–15, Las Vegas, Nevada, USA
- Laperrière L, Kabore T (2001) Monte Carlo simulation of tolerance synthesis equations. *Int J Prod Res* 39:2395–2406
- Laperrière L, Ghie W, Desrochers A (2002) Statistical and deterministic tolerance analysis and synthesis using a unified Jacobian-torsor model. *CIRP Ann* 51:417–420
- Legoff O, Villeneuve F, Bourdet P (1999) Geometrical tolerancing in process planning: a tridimensional approach. *Proc Inst Mech Eng Part B* 213:635–640
- Li B, Roy U (2001) Relative positioning of toleranced polyhedral parts in an assembly. *IIE Trans* 33:323–336
- Marziale M, Polini W (2009a) Clearance joint modeling for tolerance analysis. In: Proceedings of the 11th CIRP international conference on CAT, Annecy, France, March 26–27
- Marziale M, Polini W (2009b) A review of two models for tolerance analysis: vector loop and matrix. *Int J Adv Manuf Technol* 43:1106–1123
- Nigam SD, Turner JU (1995) Review of statistical approaches to tolerance analysis. *Comput Aided Des* 27:6–15
- Prisco U, Giorleo G (2002) Overview of current CAT systems. *Integr Comput Aided Eng* 9:373–397
- Salomons OW, Haalboom FJ, Jonge Poerink HJ et al (1996) A computer aided tolerancing tool II: Tolerance analysis. *Comput Ind* 31:175–186
- Shen Z, Ameta G, Shah JJ et al (2004) A comparative study of tolerance analysis methods. *J Comput Inf Sci Eng* 5(3):247–256
- Teissandier D, Couétard Y, Gérard A (1999) A computer aided tolerancing model: proportioned assembly clearance volume. *Comput Aided Des* 31:805–817
- Villeneuve F, Legoff O, Landon Y (2001) Tolerancing for manufacturing: a three-dimensional model. *Int J Prod Res* 39:1625–1648
- Whitney DE (2004) Mechanical assemblies. Their design, manufacture and role in production development. Oxford University Press, New York
- Whitney DE, Mantripragada R, Adams JD et al (1999) Toward a theory for design of kinematically constrained mechanical assemblies. *Int J Robot Res* 18:1235–1248

Geometric Tolerances

Impact on Product Design, Quality Inspection and
Statistical Process Monitoring

Colosimo, B.M.; Senin, N. (Eds.)

2011, XVIII, 336 p., Hardcover

ISBN: 978-1-84996-310-7