DESIGN FOR TOLERANCE OF ELECTRO-MECHANICAL ASSEMBLIES

R. Sudarsan, Y. Narahari, K.W. Lyons, R.D. Sriram

National Institute of Standards and Technology

Gaithersburg, MD 20899, USA

{sudarsan,klyons,sriram}@cme.nist.gov, hari@csa.iisc.ernet.in

M.R. Duffey

The George Washington University

Washington, DC 20052, USA

duffey@seas.gwu.edu

Abstract

Tolerancing decisions during the design of electro-mechanical products profoundly affect cost and quality. Existing approaches to tolerance analysis and synthesis entail detailed knowledge of geometry of the assemblies and are mostly applicable during advanced stages of design, leading to a less than optimal design process. During the design process of assemblies, both the assembly structure and associated tolerance information evolve continuously and significant gains can be achieved by effectively using this information to influence the design of the assembly. Motivated by this, we identify and explore two goals of research that we believe can expand the scope of tolerancing to the entire design process. The first goal is to advance tolerancing decisions to the earliest possible stages of design. This issue raises the need for effective representation of tolerancing information during early stages of design and for effective assembly modeling. The second goal addresses the appropriate use of industry best practices and efficient computational approaches for tolerance analysis and synthesis. Pursuit of these goals leads to the definition of a multi-level approach that enables tolerancing to be addressed at successive stages of design in an incremental, continuous ongoing fashion. The resulting design process, which we call the Design for Tolerance process, integrates three important domains: (1) design activities at successive stages of design; (2) assembly models that evolve continuously through the design process; and (3) methods and best practices for tolerance analysis and synthesis. We demonstrate major steps of our proposed approach through a simple, yet illustrative, example.

1 Introduction

Tolerancing is a critical issue in the design of electro-mechanical assemblies. In a recent workshop at the National Institute of Standards and Technology (NIST) [1], several leading researchers from the industry, the academia, and the government emphasized the need for investigating assembly level tolerancing issues and for developing tolerancing standards related to assembly. Tolerancing is a major component of the OpenADE (Open Assembly Design Environment) architecture being developed and implemented at NIST [2]. Tolerances must be considered early in the design cycle to develop product specifications for quality assemblies that can be manufactured cost-effectively. However, during very early design (for example, conceptual design), not much is known about the layout or geometry of the product and it is difficult to make any meaningful decisions related to tolerancing. However, as the design process progresses and the product definition emerges in increasing clarity and detail, tolerance-related information becomes available gradually and incrementally. The success of *Design for X* (DFX) concepts have established the efficacy of providing feedback on downstream manufacturing concerns. Motivated by this, we look into effective ways to represent tolerancing information at successive stages of the design process. Using this information will facilitate the creation of quality designs that can be manufactured cost-effectively. An important task in this endeavor will be the integration of appropriate tools and approaches for tolerance analysis and synthesis.

1.1 Status of Design Tolerancing

Existing approaches to design tolerancing generally require detailed knowledge of geometry of the assemblies and are mostly applicable during advanced stages of design, leading to a less than optimal design process. The current industry practice is to assign tolerances only during advanced stages of design, after nominal dimensions have been fixed by designers. Many firms use Monte Carlo simulation to conduct tolerance analysis on a detailed geometric model of the product. Both *classical* and *statistical* tolerancing are currently practiced in industry [3]. There is now a vast

body of literature on tolerance analysis and synthesis, with several survey papers available on important topics [4–8]. There are several software packages available exclusively for tolerance analysis and synthesis [6]. Industry best practices in design tolerancing include the well-known Motorola six sigma program [9]. There are also proprietary methods and software such as HPD (Holistic Probabilistic Design) from Xerox [10], which promises to embed sophisticated analytical computations based on stochastic techniques.

There are some important recent efforts, albeit preliminary, that attempt tolerancing decisions during early stages of design. These include the work based on *key characteristics* [11]; and assembly-oriented design using assembly representations such as *datum flow chains* [12, 13]. Monte Carlo simulation is the tool used by these researchers for tolerance analysis. Though some important design related decisions can potentially be enabled by these approaches during early stages of design, the actual tolerance analysis would require a skeletal geometric visualization of the assembled product and detailed simulation.

1.2 Contributions and Outline

The objective of this work is to focus on successive stages of design and effectively use assembly and tolerance representations at successive levels of abstraction. Stage-appropriate tolerance analysis and synthesis computations can then be integrated into an effective design process from a tolerancing perspective. Figure 1 depicts a conceptual diagram that shows the three major threads in the proposed methodology:

- Assembly models for tolerancing (Section 2)

- Techniques and best practices for tolerancing or *Tolerance Engine* (Section 3)

- *Design for tolerance* process (Section 4)

In particular, in Section 4, we describe a multi-level design tolerancing architecture that integrates design process stages, assembly models for tolerancing, and tolerancing tools and techniques. In Section 5, we consider a simple, representative example and delineate the major steps of our approach.
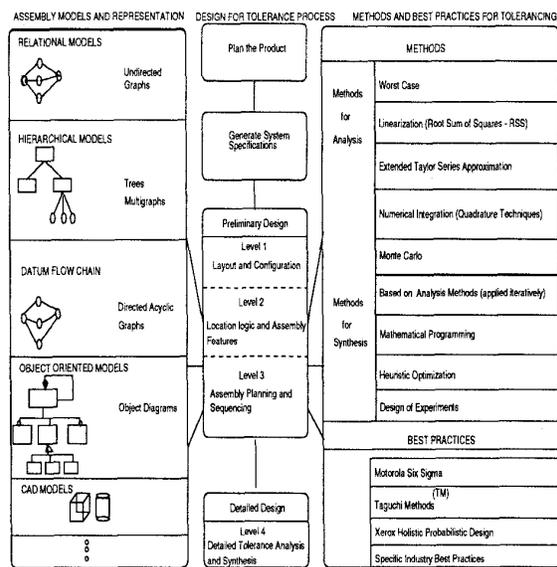

Figure 1: Assembly models, design process stages, and tolerancing tools.

2 Assembly Modeling and Tolerance Representation

2.1 Assembly Models

There are a variety of assembly models available that capture assembly information at different levels of abstraction in the design process and are useful in specific ways. We now provide a quick review of some relevant models.

Three types of assembly representations are popularly discussed in the literature: Solid models, relational models, and hierarchical models [14]. The solid models represent part positions in terms of their spatial coordinates. Relational models represent geometric relations between individual feature, part, and subassembly datums. The hierarchical models capture the assembly decomposition in a tree structure.

Solid models provide sufficient information for graphic display of the assembly but are not convenient for the purpose of tolerancing. For example, changes to the positions or dimensions of individual parts are not always propagated to neighboring parts in the assembly.

In relational models, also called *liaison diagrams*, parts are related by sets of *mating features*. The assemblies are usually modeled as undirected graphs where the nodes represent the parts and the arcs represent the contacts between them. These models cannot

1491

capture the order in which the geometrical relationships are established. Relational models have been used in analysis applications such as robot path planning, generation of feasible assembly sequences, and robot assembly planning. However, relational models alone are not adequate for tolerancing [14].

In a hierarchical model, an assembly is divided into sub-assemblies, which in turn are decomposed into sub-assemblies or individual parts. Though a hierarchical model captures assembly decomposition and high level precedence relationships in terms of its different levels, it does not assign any hierarchy on the order of establishment of liaisons between individual parts within a particular subassembly. Also, such a hierarchy is yet undeveloped during early design. A tree structure is most appropriate for representing a hierarchical model. Several variants of the hierarchical model have been employed [14, 15]. The hierarchical model has been used in assembly sequence analysis, kinematics analysis, and tolerance analysis (during detailed design).

A recent proposal for assembly modeling with emphasis on early design representation is that of *datum flow chains* (DFC) [12, 13]. A DFC is a directed acyclic graph that defines the hierarchy of dimensional relationships between parts in an assembly. Each node represents a part or a fixture or a defined feature on the part or fixture. A directed arc from Node A to Node B indicates that the datum at A determines the dimensional location of the part at B. A DFC abstractly captures the underlying location logic of an assembly and often enables a visualization of the way in which tolerances may propagate. They can be used early in the design process to represent evolving assembly configurations.

Since tolerance representation is the main focus of this work, we now look at assembly models for tolerancing.

2.2 Assembly Models for Tolerancing

Representation of assemblies for automatic generation of tolerance chains is the subject of a paper by Wang and Ozsoy [15]. Their model represents an assembly as a detailed data structure with information on assembly decomposition; (4×4) transformation matrix for each instance of a component and subassembly; mating features; mating conditions (against, parallel, fit); dimensions and tolerances of the mating features; etc. The above information is used to generate a tolerance chain algorithmically for any given assembly. The details required by this representation make it inappropriate for use in early stages of design.

In [16], Whitney *et al* have proposed a connective model of assembly that has the following information: mating features that build up the assembly; underlying co-ordinate structure of the assembly; and homogeneous (4 × 4) matrix transforms to represent standard Y14.5M-1982 dimensions and tolerances of each part. The transforms represent both the nominal relations between parts and the variations caused by geometric deviations allowed by the tolerances. The (4×4) matrix transforms can be used to propagate tolerances through an assembly. The above representation can be used in early design but the emphasis of the work is more on representation of geometric variations for statistical tolerance analysis rather than on assembly modeling during early design. Datum flow chains have been used more recently [12] to generate tolerance chains for assemblies during early design. The method uses the location logic embedded in DFCs with skeletal geometry of the assembly, combining it with the (4×4) matrix representation.

The models and tolerancing approaches discussed above are relevant at appropriate stages of the design process and thus constitute important elements of the methodology proposed in this paper.

3 Methods and Best Practices for Tolerancing

There are a variety of tools, techniques, and best practices for design tolerancing. These constitute an important element of the proposed methodology.

Tolerancing encompasses both tolerance analysis and tolerance synthesis. In tolerance analysis, the problem is to evaluate the effect of individual variances and deviations on the variability of certain key characteristics of the assembled product. Tolerance synthesis involves allocation or distribution of variances to individual components or sub-assemblies of an assembly based on the tolerance requirements on the assembly.

Tolerance analysis can be either classical or statistical. In classical tolerance analysis (also called worst-case tolerance analysis), the analysis considers the worst possible combinations of individual tolerances and examines the assembleability of the components. Statistical tolerancing is a more rigorous and realistic way of looking at tolerances. Various approaches to statistical tolerance analysis are surveyed in [7]. There are a few well-known industry best practices in statistical tolerance analysis such as the Motorola six sigma approach [9, 17]. The Xerox HPD approach [10] and Taguchi-based methods [18] are other methods being used in the industry. Monte Carlo simulation [6] is

1492

```
A1 : DESIGN PRODUCT

    A11 : Plan Products
    A12 : Generate Product Specifications
    A13 : Perform Preliminary design
        A131 : Develop Functional Decompositions
        A132 : Evaluate and Select Decompositions
        A133 : Develop Preliminary Configurations
        A134 : Consolidate Configurations
        A135 : Evaluate Alternative Designs
        A136 : Select Design
    A14 : Produce Detailed Designs
        A141 : Design System/Component
        A142 : Analyze System/Component
        A143 : Evaluate System/Component Design
        A144 : Optimize Designs
        A145 : Produce Assembly Drawings
        A146 : Finalize System/Component Design
```

Figure 2: Design activities according to SIMA reference architecture

the most expensive but also the most extensively used approach for tolerance analysis.

Tolerance synthesis can be deterministic or stochastic. Deterministic tolerance synthesis approaches are surveyed in [8]. Stochastic or statistical approaches to synthesis are based on statistical design optimization approaches such as Taguchi methods [18]. Usually, tolerance synthesis uses tolerance analysis in an iterative way. Thus all the tolerance analysis approaches are relevant for tolerance synthesis also.

4 Design for Tolerance

We now propose an approach called *Design for Tolerance (DFT)*, which will enable us to take tolerance related decisions throughout the entire design process. To clarify which stages of design we wish to focus on, we refer to the SIMA (Systems Integration for Manufacturing Applications) reference architecture specified at the National Institute of Standards and Technology [19]. Figure 2 shows the various phases and activities in the SIMA reference architecture.

4.1 Design Tolerancing: A Continuous Process

Potentially, tolerance considerations can influence the decisions taken at each of the above stages, in increasing level of detail. Also, the decisions taken at a particular stage influence and simplify the decisions taken in the downstream stages. Like other attributes of a product design, tolerance information changes over time, through successive stages from product planning to detailed design, through on-going production. Hence a robust tolerance representation is mutable and directly related to the design process representation. The incremental refinement of processes and tolerance representations proceeds in symbiotic fashion.

One way of viewing this continuous process of tolerance decision making is in terms of the pruning that this causes at successive stages in the space of feasible solutions to the design problem. Early on in the design process, the solution space has a staggering cardinality and the tolerancing decisions, if taken in a continuous ongoing fashion, can lead to substantial reduction in the space of possible solutions, thus making the design process efficient. Another way of viewing this is in terms of reducing the design iterations or design rework. In this sense, design for tolerance is similar in spirit to design for manufacturing/assembly which also have the effect of dramatically shrinking the space of solutions and reducing iterations. Furthermore, Design for Assembly (DFA) or Design for Manufacturing (DFM) or such other design related strategies may have close coupling with tolerance related decisions and may both influence and be influenced by tolerancing at various stages. Furthermore, this continuous process enables identification of the critical variables or dimensions to be considered for tolerancing.

4.2 Design Tolerancing: A Multilevel Approach

We focus on four different stages (or levels) in the design of an assembly. These stages are fairly representative and generic, and roughly correspond to some of the activities in the SIMA reference architecture. Figure 3 captures the essence of this architecture for DFT. The individual levels delineated here are:

- Level 1: Layout and Configuration

- Level 2: Location Logic and Assembly Features

- Level 3: Assembly Planning and Sequencing

- Level 4: Detailed Design

Roughly speaking, the first three levels can be grouped under *Preliminary Design* of the SIMA reference architecture while Level 4 can be categorized into *Detailed Design*. The first two stages of the SIMA architecture, namely, *Plan Products* and *Generate Product Specifications* occur before these levels and are not
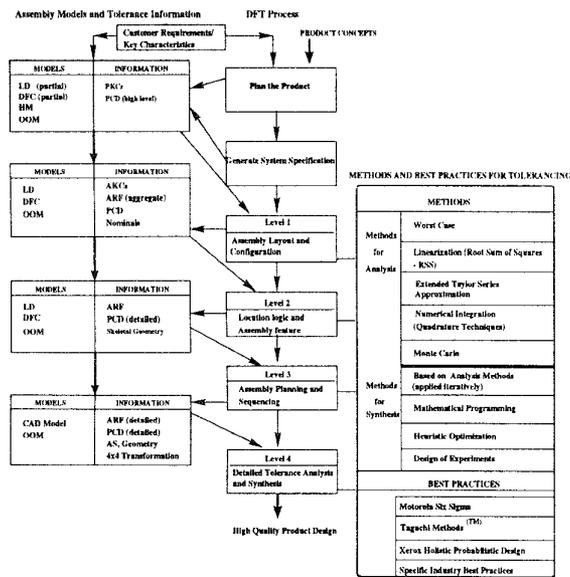
1493

Figure 3: An architecture for design for tolerance. Legend: LD - Liaison Diagram; DFC - Datum Flow Chain; HM - Hierarchical Model; OOM - Object Oriented Model; AS - Assembly Sequence; PKCs - Product Key Characteristics; AKCs - Assembly Key Characteristics; ARF - Assembly Response Function; PCD - Process Capability Data

considered here from the tolerancing viewpoint. These first two stages however provide critical inputs to the tolerancing process. See Figure 3.

The initial stages of the proposed process could be similar to the top down assembly design process, based on key characteristics [11, 12]. Once the product concept is known and engineering specifications are generated based on the key characteristics, Level 1 of the proposed process can commence.

Level 1 involves decisions regarding the product layout and configuration. Such decisions include: number of sub-assemblies, the configuration of critical sub-assemblies, distribution of components into sub-assemblies, and rough layout of the assembly. The information available at this level can be described in the form of a tree (assembly decomposition) or a partial DFC (to capture whatever location logic is known at this point). Candidate layouts or configurations can be identified and represented using these models. These layouts or configurations typically might differ in terms of ease of tolerancing. The tolerancing considerations here are at a high level and may be directly influenced by customer specifications. To effect such high level tolerancing decisions, aggregate level manufacturing process capability data will be required and

is usually available at this point. Simple statistical assumptions and probabilistic calculations can be used at this stage. The example in the next section clarifies this issue.

At the next level (Level 2), the following information is assumed to be available: assembly key characteristics; assembly response function (aggregate level); tolerance requirement at interfaces between major sub-assemblies and components; and relevant process capability data. The decisions here are concerned with the location logic (how to locate sub-assemblies and components with respect to one another) and with choosing the appropriate assembly features to go with the location logic. The choice of features itself might depend on the assembly sequence (not the detailed sequence but a precedence specification among major assembly steps). The DFC model is suitable to capture the available or evolving assembly information here. There is close coupling among selection of features, selection of assembly sequence, and creation of model here. Assembly models such as liaison diagrams and task graphs are also relevant at this level. Tolerance analysis can tell us which location logic is better from a tolerancing viewpoint and which set of assembly features would best accomplish tolerance achievement. This stage might also help us to find preliminary target values and tolerances for individual parts.

We proceed next to Level 3 where the detailed assembly response function, detailed process capability data, skeletal geometry of the assembly, assembly features, and specification of parametric or geometric tolerances of individual parts and features are assumed to be known. From the tolerance specification, one may derive (4×4) matrix transforms for the nominals and variabilities associated with the parts [16]. Selection of the detailed assembly sequence that achieves the required tolerance specifications in the best possible way could be decided here. The models that we employed in the previous stage, like DFC, liaison diagrams, etc., can again be used here, but with richer information now available. This kind of representation and analysis is presented in [15], where several data structures to capture tolerance related information are presented. With the information available here, one can also carry out tolerance synthesis.

Level 4 corresponds to the detailed design stage. At this level the following information is available; the complete assembly sequence, geometric data about the parts and features, detailed part level tolerance requirements, the assembly response function in complete form; and low level process capability data. Detailed tolerance analysis and synthesis can be carried
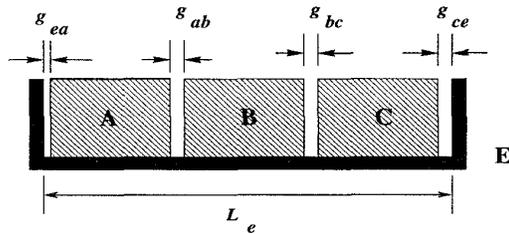
1494

Figure 4: An example assembly comprising an envelope and three components

out here. Most tolerancing studies and tolerancing tools available support this level of design.

The *Tolerance Engine* is the computational module for this DFT process. It consists of a variety of tools and approaches for tolerancing. Which tool or approach to employ at a given level of the DFT process needs careful thought and can depend on a variety of factors such as the product domain, nature of the assembly response function, number of variables involved, etc.

We note that each design stage above is iterative both internally (feedback within a level) and across the levels (feedback from a given level to a previous level).

The design process evolution is accompanied by a continuous refinement of the assembly model and the tolerancing information. This raises the issue of effective representation of this dynamic information. Object-oriented models (see for example, [20]) offer an attractive option here.

5 Example

Here, we present a simple and illustrative assembly example, give a rough sketch of its design process, and bring out the important role tolerancing considerations can play in successive stages of its design. We consider a simple mechanical assembly, say P, comprising three components A, B, C, and an envelope E in which the three components are to be placed. Figure 4 shows a view of this assembly. The diagram is conceptual and is not to be viewed as implying any geometry or shape. The conformance or functionality of the assembly is decided by the following criteria:

1. The gap g_{ab} between the components A and B should lie in a desired tolerance zone

2. The gap g_{bc} between the components B and C should lie in a desired tolerance zone

3. There should be no interference between A and E and between C and E.

A *tolerance zone* for a given element (size or feature or form) defines the range of allowable variations of the nominal element. For example, if the length of a part is of interest, then an interval around the nominal length becomes a tolerance zone. If R_{ab} and R_{bc} represent the tolerance zones for g_{ab} and g_{bc} respectively, the above criteria can be expressed as:

$$g_{ab} \in R_{ab}, \quad g_{bc} \in R_{bc}, \quad g_{ea} \geq 0, \quad g_{ce} \geq 0,$$

For the sake of simplicity, we shall consider here only parametric tolerances. Consequently, the tolerance zones become intervals. The discussion is similar for geometric tolerances also, with appropriate extensions and reinterpretation. Note that $g_{ab}, g_{bc}, g_{ea}, g_{ce}$ are all continuous random variables.

Let L_a, L_b, L_c, L_e denote the lengths of the components A, B, C, and E respectively. In the case of A, B, and C, these represent the length of the components; in respect of E, L_e represents the length of inner boundary of the envelope. Again, these lengths are all treated as continuous random variables. We now discuss how tolerance related decisions can be taken at various stages of design of the above assembly.

5.1 Selecting a Configuration

This is the earliest of the four stages in the design process discussed here. Figure 5 shows three possible ways of configuring the four components A, B, C, and E as product P; there could be other configurations as well. In Configuration 1, all four components are treated as individual components. In Configuration 2, P comprises E and a subassembly that consists of components A, B, and C. The motivation for considering this configuration might be that the said subassembly is available off-the-shelf from a known vendor. Likewise, Configuration 3 is another candidate. It is clear that the process capabilities and the associated parametric variations of the components and sub-assemblies will influence the choice of which configuration to choose.

At this stage, the selection of one of the above three configurations could be based on how well the configuration enables proper fitting of the components inside the envelope. This implies that the choice depends on a probability of the following form:
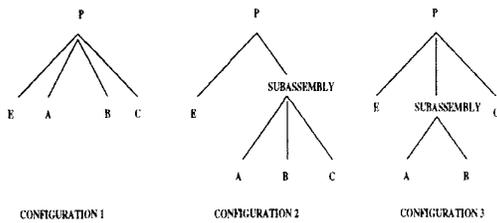
$$P\{L_a + L_b + g_{ab} + L_c + g_{bc} < L_e\}$$

1495

Figure 5: Three different configurations of the assembly

Figure 6: Datum flow chains representing different location logics

The following data is known about these configurations:

- In the case of Configuration 2, the term $L_a + L_b + L_c + g_{ab} + g_{bc}$ (which is a random variable) is known in aggregate form since the subassembly from a vendor consists of all three components A, B, and C. Data about g_{ab} and g_{bc} will also be available from this vendor. Also, L_e is known separately.

- With respect to Configuration 3, random variable $L_a + g_{ab} + L_b$ is known in aggregate form while L_c and L_e are known separately. Data about g_{ab} will also be available from the vendor, but g_{bc} is not known since it depends on the process of assembling.

- In the case of Configuration 1, only the distributions of L_a, L_b, L_c, and L_e are known. The distributions of the gaps g_{ab} and g_{bc} are decided by the assembly process.

Details of computation of the above probability are not included here for the sake of brevity. For such computations, the Xerox HPD method is one possible approach. Monte Carlo simulation also can be used here.

5.2 Selecting Location Logic and Assembly Features

In this stage of design, our interest is in fixing the location logic, which often allows the choice of assembly features, as explained below. Figure 6 shows three candidate DFCs. Candidate 1 captures assembly sequences where the component B is assembled last. In Candidate 2 assemblies, B is the first one to be assembled into the envelope, whereas Candidate 3 assemblies correspond to those sequences in which B is assembled in the middle between A and C (these two in any order). From the conceptual diagram of Figure 4, it is clear that Candidate 1 type of assembling
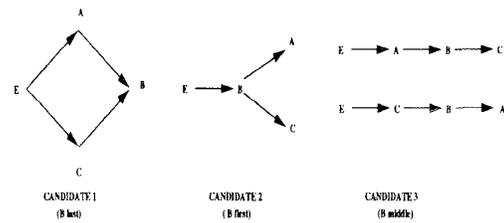
may necessitate A and C to have two mating features; Candidate 2 type may entail just one assembly feature each on A and C; and Candidate 3 type may require either A or C to have two features while the other may have just one feature.

One way of comparing the three candidates above is to compute the following probability of conformance in each case:

$$P\{(g_{ab} \in R_{ab}) \text{ and } (g_{bc} \in R_{bc}) \text{ and } (g_{ea} > 0) \text{ and } (g_{ce} > 0)\}$$

Note that the computation of the above probability is more detailed than what we computed in Section 5.1. Knowing the nominals and tolerances of E, A, B, and C and process capability data corresponding to the various assembly operations, one can compute the above probability in each of the three cases. We may remark that Candidate 2 is likely to be the best since it enables variation to be adjusted to where it is not important and also because only one assembly feature each may be required on A and C. On the other hand, if there is high variability in the dimension of B, then Candidate 3 may turn out to be a better choice.

5.3 Selection of Assembly Sequence

We observed in the previous subsection that Candidate 3 is likely to be better if there is high uncertainty in the dimension of B. If this candidate is chosen, then there are two possible sequences: E → A → B → C or E → C → B → A. Using the data available about the nominals, tolerances, and process capabilities for the individual parts, one can compute the probability of conformance and decide which sequence is better. For example, if A has more variability than C, then the second sequence is likely to be better than the first sequence, since the higher variation of A can be washed to where it is not important. In this case, this is intuitively clear. But in complex assemblies, one necessarily needs to carry out such analysis.

1496

5.4 Detailed Analysis and Synthesis

When the design process reaches advanced stages, tolerance analysis and synthesis can be done in increasing level of detail. For example, the following information may be known:

- Assembly sequence: Say, $E \rightarrow B \rightarrow A \rightarrow C$

- Nominals N_a, N_b, N_c, N_e and tolerances T_a, T_b, T_c, T_e of A, B, C, E, respectively.

- Process capabilities of different assembly steps in the assembly sequence

Given the above data, one can compute the probability of conformance. Also, detailed design issues such as selecting the best nominals, allocating tolerances optimally, and establishing standard deviations of critical processes can also be effectively addressed because of the detailed data available. There are several studies of this type available in the literature; for example, see the report by Harry and Stewart [17].

6 Summary

In this paper, we have outlined a continuous multi-level approach to design tolerancing of electro-mechanical assemblies. The architecture has three main threads: assembly models for tolerancing; techniques and best practices for tolerancing; and the evolving design process. We have delineated a four level approach and illustrated the methodology for a simple, representative mechanical assembly.

The work reported here raises several important issues to probe into: assembly representation, tolerancing standards, design process definition, and design process modeling. In particular, the research will lead to improved understanding of the assembly design process from a tolerancing angle and integration of various best practices at various stages of this design process. This in turn will provide a critical input to the formulation of assembly and tolerancing standards.

Acknowledgments

Disclaimer

References

[1] R. Sudarsan, K. Lyons, and M. Pratt (Editors). *Assembly Level Tolerancing: Standards and Implementation Issues.* Workshop Handout, National Institute of Standards and Technology, 1997.

[2] K.W. Lyons, S. Angster, P. Hart, Y. Narahari, V. Rajan, and R. Sudarsan. OpenADE: Open Assembly Design Environment. Technical report, National Institute of Standards and Technology, 1997 (Forthcoming).

[3] V. Srinivasan and M.A. O'Connor. Towards an ISO standard for statistical tolerancing. In F. Kimura, editor, *Proceedings of the Fourth CIRP Design Seminar on Computer-Aided Tolerancing*, pages 159–172. Chapman & Hall, London, 1996.

[4] D.H. Evans. Statistical tolerancing: The state of the art, Part II. Method for estimating moments. *Journal of Quality Technology*, 7:1–12, 1975.

[5] K.W. Chase and A. Parkinson. A survey of research in the application of tolerance analysis to the design of mechanical assemblies. *Research in Engineering Design*, 1(3):23–37, 1991.

[6] J.U. Turner. Current tolerancing packages. In V. Srinivasan and H. Voelcker, editors, *Proceedings of the 1993 International Forum on Dimensional Tolerancing and Metrology.* ASME, New York, 1993.

[7] S.D. Nigam and J.U. Turner. Review of statistical approaches to tolerance analysis. *Computer Aided Design*, 27(1):6–15, January 1995.

[8] A. Kusiak and C.X. Feng. Deterministic tolerance synthesis: A comparative study. *Computer Aided Design*, 27(10):759–768, October 1995.

[9] M.J. Harry. The nature of six sigma quality. Technical report, Government Electronics Group, Motorola Inc., Scottsdale, Arizona, 1987.

[10] J.M. Parks. Holistic approach and advanced techniques and tools for tolerance analysis and synthesis. In F. Kimura, editor, *Computer-Aided Tolerancing*, pages 251–269. Chapman and Hall, London, 1996.

[11] T. Cunningham, R. Mantripragada, D.Lee, A. Thornton, and D. Whitney. Definition, analysis, and planning of a flexible assembly process. In *Proceedings of the Japan-USA Symposium on Flexible Automation, Boston, USA*, pages 767–778, 1996.

[12] R. Mantripragada, T.W. Cunningham, and D.E. Whitney. Assembly oriented design: A new approach to designing assemblies. In M.J. Pratt, R.D. Sriram, and M.J. Wozny, editors, *Product Modeling for Computer Integrated Design and Manufacture*, pages 308–324. Chapman & Hall, 1997.

[13] D.E. Whitney. The potential for assembly modeling in product development and manufacturing. Technical report, The Center for Technology, Policy, and Industrial Development, MIT, Cambridge, MA, USA, 1996.

[14] S. Srikanth and J.U. Turner. Toward a unified representation of mechanical assembly. *Engineering with Computers*, 6(2):103–112, 1990.

[15] N. Wang and T.M. Ozsoy. Representation of assemblies for automatic tolerance chain generation. *Engineering with Computers*, 6(2):121–126, 1990.

[16] D.E. Whitney, O.L. Gilbert, and M. Jastrzebski. Representation of geometric variations using matrix transforms for statistical tolerance analysis in assemblies. *Research in Engineering Design*, 6:191–210, 1994.

[17] M.J. Harry and R. Stewart. Six sigma mechanical design tolerancing. Technical report, Government Electronics Group, Motorola Inc., Scottsdale, Arizona, 1988.

[18] R.N. Kacker. Taguchi methods. In *Handbook of Statistical Methods for Engineers*, pages 19.1–19.27. Mc-Graw Hill, 1988.

[19] E.J. Barkmeyer (Editor). SIMA (Systems Integration for Manufacturing Applications) reference architecture, Part 1: Activity models. Technical report, National Institute of Standards and Technology, Internal Report 5939, December 1996.

[20] S.R. Gorti and R.D. Sriram. From symbol to form: A framework for conceptual design. *Computer Aided Design*, 28(11):853–870, 1996.