Extended Variational Design Technology - Foundation For Integrated Design Automation

Jack C. H. Chung*, Teng-Shang Hwang, Chien-Tai Wu,
Yu Jiang, Jia-Yi Wang, Yong Bai, Hongliu Zou
SDRC
2000 Eastman Drive
Milford, Ohio 45150

ABSTRACT

Engineering design is an iterative process with a fundamental need for the consistent management and propagation of product dependencies. Constraint-based design provides a unified framework to meet this critical need, but there are known issues due to the complexity of the problem within three-dimensional space. This paper presents results from a decade of research into graph theory and numerical solution techniques to address issues such as user comprehension and multiplicity of solutions. The proposed solution, Extended Variational Design Technology (VGX), utilizes an innovative Drag and Drop user interaction paradigm to improve comprehension performance, and usability. It is also demonstrated that VGX technology can provide common mathematical foundation to support flexible and integrated product design, assembly and analysis.

1 INTRODUCTION

Engineering design is an iterative process encompassing requirement definition, concept design, detailed design, and design validation/analysis. Fundamental to this design process is the need for consistent management and propagation of constraints, relations, associations, dependencies and domain knowledge associated with the product. Constraint-based design provides a unified framework to meet these critical needs.

However, constraint management and solving in three-dimensional space is a very challenging task due to the characteristics of the problem. Recognized issues include non-linearity, large problem size, complexity of user comprehension, multiplicity of solutions, ill-conditioning, poor initial conditions, and large perturbations. Through a decade of research, we have developed Extended Variational Design Technology (VGX) using graph theory and numerical solution techniques in an attempt to address the above issues.

To improve user comprehension, an innovative approach using a Drag and Drop user interaction paradigm is proposed. In this virtual environment, the user can set up constraints, modify dimensions, change geometry size and location, and change dimensioning schemes by directly dragging the desired entity. The intuitiveness and simplicity of this user interaction paradigm greatly improves the usability of a 3D constraint-based design system.

VGX technology overcomes the temporal barrier in history processing. This breakthrough enables the support of 3D flexible constraining/dimensioning that is independent of model creation history. Furthermore, the complete mathematical representation of the VGX-based design achieves true integrated design automation. Mathematical analyses, such as mechanism/tolerance validation, can be performed without having to rebuild separate models. The technology has been implemented in a commercial Mechanical Design Automation (MDA) system and the initial results have proved its promising potentials.

This paper is organized as follows: Section 2 presents the modeling role of MDA systems in the current design process and a brief review of related research. Section 3 gives an overview of the technical foundation of VGX. Section 4 and 5 present the history-independent part/assembly design and drag and drop user-interaction paradigm, respectively. Section 6 illustrates the VGX-based downstream applications. The last section presents conclusions and future research areas.

2 BACKGROUND - MDA SYSTEMS AND THE ENGINEERING DESIGN PROCESS

Solid based MDA systems have received rapid acceptance over the past decade due to significant productivity gains from new capabilities in rapid prototyping, interference checking, mechanism animation, and better interfaces with analysis programs. The MDA industry also benefits from continued advances in computing technology and mathematical/geometric algorithms. All systems can now model complex parts and assemblies with complex operations such as filleting/blending, shelling, sweeping/lofting, and draft angles.

Is MDA technology really mature? Are current systems capable of supporting the complete mechanical design process, from preliminary concepts to detailed design documents? A review of the engineering design process can provide some insights into these questions.

2.1 Engineering Design Process

As illustrated in Figure 2.1, the engineering design process starts with product specification and goes through an iterative process of requirements analysis, conceptual design, detailed design, design

* E-mail: Jack.Chung@sdrc.com
Tel: (513)-576-2547

Fifth Symposium on Solid Modeling Ann Arbor MI

analysis, and manufacturing. It ends with a functional product which fulfills the product specification.

Within this process we have achieved islands of automation. The current MDA systems are able to model parts and assemblies very well but not to design them. These systems only play a role in the detailed design stage, a very late phase of the complete design process. Before design engineers can use their MDA systems, they have already turned design specifications into a definitive design layout; resolved the essential problems; evaluated the solution principles; optimized the design layout and determined the size and shape of parts. The design engineers must then enter parts and assemblies into the MDA systems to create an electronic version of the designed product.

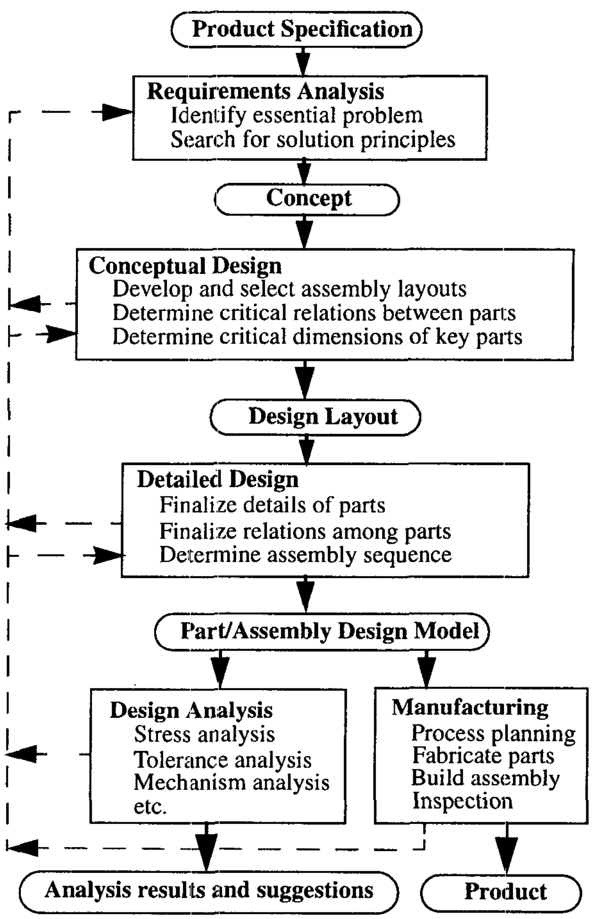


Figure 2.1 The engineering design process

2.2 Overview of Current Product Design

2.2.1 Part Design

All current mainstream MDA systems are based on sequential history processing. A part is built, as shown in Figure 2.2, by sequentially adding/subtracting features. The creation recipe of features captures design intent through sizing/positioning parameters.

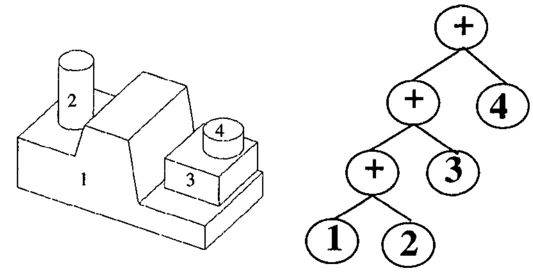


Figure 2.2 The part and the history tree

When parameters are modified, the part is updated by replaying the history recipe. The dependency between features is uni-directional, i.e., features at higher leaves of the history tree are always driven by the features below them. Reversing the dependency is a difficult task and may require a complete re-modeling of the part.

The inherent drawback of a history-based system is its dependence on the modeling sequence. It is well recognized that the modeling sequence may be more a result of convenience and expedience than implied user intent. Imposing sequential dependency between features can lead to significant design inflexibility and user confusion. For example, in the middle of constructing a part, the designer finds that the size or shape of the feature created several steps earlier must be driven by a new feature. This in general will present significant challenges to the user as well as the system. Furthermore, updating geometry with sequential history replay can result in unavoidable performance bottlenecks.

2.2.2 Assembly Design

Similar to part design, most MDA systems provide a hierarchical structure to capture and manage the assembly definition, including positioning and orientation methods. Typical assembly methods include:

- Position a part/sub-assembly in the specific global location/ orientation directly. There is no relative association between parts and sub-assemblies. For a complicated model, it would be very time consuming for engineers to correctly calculate each component's position and orientation. Furthermore, any design modification in shape or size of parts may affect the positional validity of adjacent parts.
- Sequentially place components one by one using geometric constraints, such as mating and alignment with respect to pre-existing parts/sub-assemblies. Each component's placement constraints must fully restrict the six degrees of freedom and follow a certain pre-set constraint sequence. The drawbacks with this approach are: a) can not easily assemble coupled systems as simple as a slider crank system; and b) can not easily accommodate a change in constraint schema after the assembly has been constructed.

2.2.3 Summary Current Product Design

The overviews of the engineering design process and current product design methodology illustrate that design is an iterative process. The key to improving design productivity is to provide an effective design change mechanism. Capturing and preserving the designer's intent through design changes is a critical task.

2.3 Constraint-Based Design Research

Design intent, as expressed through design specifications, comes in different forms, including: equality and inequality engineering relations; heuristic rules and optimization objectives; geometric constraints such as tangency and linear/angular dimensions; high-level relations such as associations and dependencies. Constraint management and propagation has been proposed as a viable solution to supporting diverse design intent. There has been significant research in this area over the past decade.

Parametric modeling [9, 16, 18] is based on the geometry construction process in drafting in which each geometric entity is constructed one at a time in a specific sequence. By remembering the construction sequence, the geometry can be reconstructed after a change in dimension. The limitations with this approach include lack of support for non-constructible geometry and engineering equations.

In the numerical, or variational geometry approach [6, 14, 15, 19], the geometric constraints are converted to mathematical equations and solved by numerical methods such as Newton-Raphson Method [10]. This approach is very general, yet it suffers from known numerical issues such as multiplicity of solutions, ill-conditioning, poor initial conditions, and large perturbations.

A node-edge graph can also be used to represent geometric entities and constraints [11, 17, 22]. The graph is then decomposed into smaller solvable sub-graphs by using bi-connectivity and tri-connectivity algorithms from graph theory. Each sub-graph is then solved and pieced together using rigid-body transforms. The solution is very robust and well-behaved. Unfortunately, some constraint problems result in graphs that are not decomposable.

Constraint propagation based on geometric reasoning in artificial intelligence has been extensively studied [1, 2, 4, 5, 12]. Each solvable constraint sub-problem can be represented by production rules or predicate logic with pre-conditions. Constraints on the model are fed to the system as state descriptors. Rules with all pre-conditions satisfied against the state descriptors will get fired and new state descriptors will get generated. This approach can handle both geometric constraints and heuristic rules. It lacks, however, the strong decomposition and solving capability inherent in the graph or numerically based approaches.

Kramer [13] applied concepts in mechanism analysis to analyze the degrees of freedom of each rigid-body. This method works well for mechanism and rigid-body assembly problems. It appears to be somewhat awkward when applied to variational part constraint problems. Recently Fudos and Hoffmann [8] developed an efficient constraint decomposition method based on rigid-body splitting. This method offers another new perspective to constraint solving and may be valuable to complement the shortcomings of other approaches.

3 TECHNICAL FOUNDATION

A geometric constraint network consists of geometry, constraints, dimensions (e.g. linear dimensions, angular dimensions, and radial dimensions), and engineering relations as illustrated in Figure 3.1 [7].

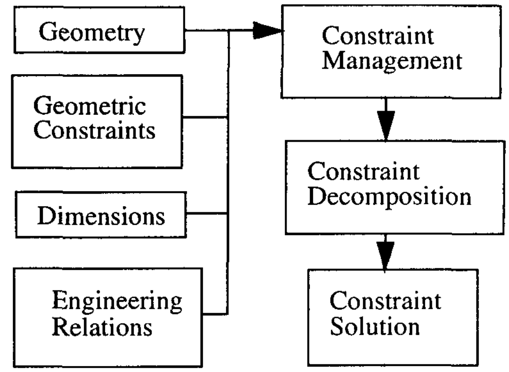


Figure 3.1 Overall functional diagram

Constraint management deals with constraint validation, constraint degree-of-freedom (DOF) analysis, and detection of over-constrained conditions to ensure the validity of the constraint network. Constraint decomposition breaks up the system into smaller sub-systems so that the solving can be done efficiently. Constraint solution has to deal with problems such as singularity, ill-conditioning, multiplicity of solutions, poor initial conditions and large perturbations to solve the system.

3.1 Numerical Approach

Figure 3.2 shows an overall functional diagram for the numerical approach.

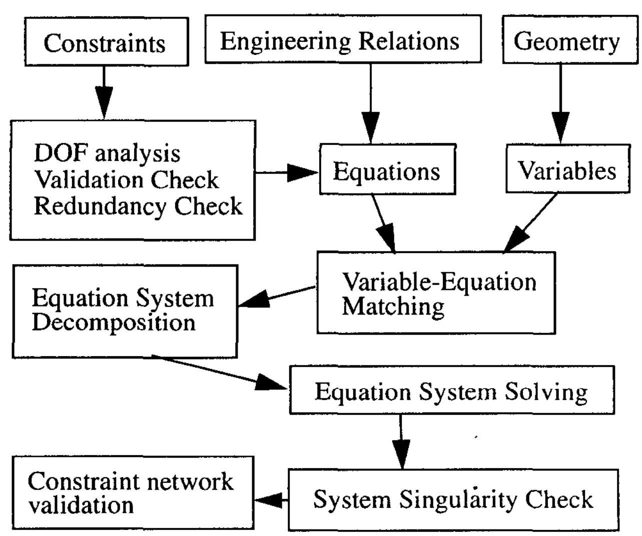


Figure 3.2 Overall numerical approach diagram

First the geometric entities are represented as variables, then constraints and dimensions are represented as mathematical equations. The graph-theoretic algorithm [19] is then applied to match each equation with a unique variable to prevent over-constraining. After equation-variable matching, the dependency and coupling between all equations are derived, leading to the breakup of the

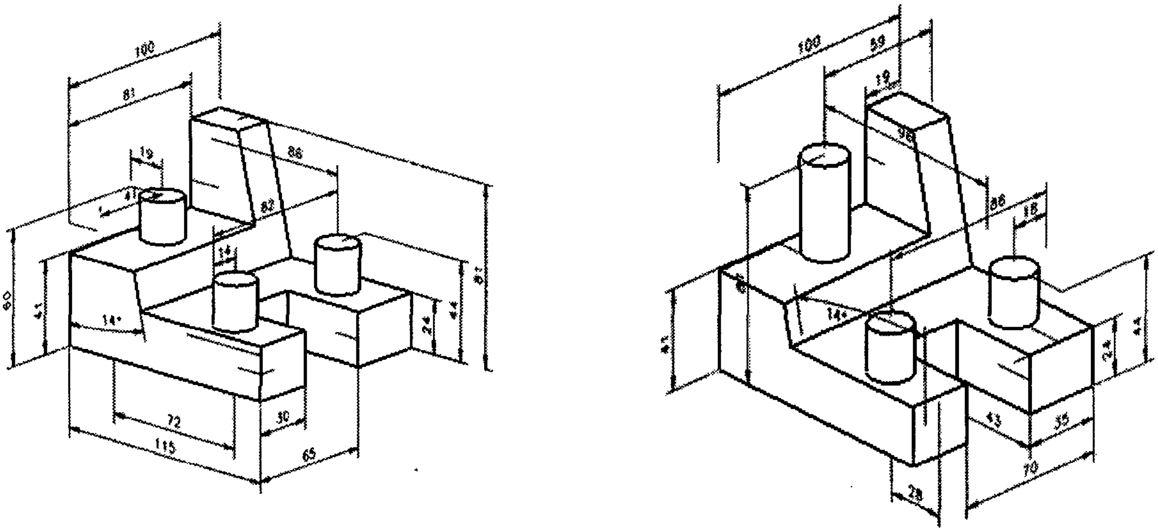


Figure 4.1 History-independent part design

Constraints and dimensions that apply among entities can be expressed as

$$C_i\,(E_j, E_k) \quad i = 1,\ldots,n$$

where E_j and E_k belong to either the same or different features and parts. Each constraint can be formulated numerically as a set of equations denoted by

$$\Phi^*(\mathbf{q}) = [\Phi_1(\mathbf{q}), \Phi_2(\mathbf{q}),\ldots, \Phi_m(\mathbf{q})]^T = \mathbf{0} \qquad (1)$$

where $\mathbf{q} = [\mathbf{u}^T, \mathbf{v}^T, \mathbf{w}^T]^T$ is the generalized coordinates in R^n space and $\mathbf{u}$ is a set of geometric variables representing geometric entities, $\mathbf{v}$ is a set of independent variables representing the linear or angular dimensions that are not derived from engineering equations, and $\mathbf{w}$ is a set of intermediate coordinates that are generalized coordinates other than geometric and independent variables.

Therefore the permissible configuration of a constrained system which is defined by nonlinear geometric, dimensional and engineering equations that do not depend explicitly on time can be expressed as

$$\Phi(\mathbf{q}) = [\Phi_1(\mathbf{q}), \Phi_2(\mathbf{q}),\ldots, \Phi_p(\mathbf{q})]^T = \mathbf{0} \qquad (2)$$

In order to have a well-posed formulation for the underlying system, it is important that constraints imposed to the system should be independent, except at certain critical configurations of the system. By solving Eq. 2, the parameters associated with geometric entities in the model can be determined and therefore the whole design can be updated accordingly. Examples are used in the following sub-sections to illustrate the power of VGX technology for variational part and assembly design.

4.1 Variational Part Design

By applying constraints and dimensions to the 3D model directly, the users are allowed the flexibility to change their constraining and dimensioning schemes at any point of time, independent of the feature construction sequence. This helps to eliminate the burden of having to understand the construction history in order to make design changes. Figure 4.1 shows a model created with different dimension schemes. Note that the creation order of the features is no longer important and either dimension scheme can be used to drive the design. The flexible dimensioning capability offered by VGX technology will for the first time allow MDA users to express their machining and inspection intent directly in the 3D model without having to resort to 2D drawings.

4.2 Variational Assembly Design

In variational assembly design, 3D constraints and dimensions can be applied to multiple instances in arbitrary orders. Figure 4.2 depicts four spheres which are tangent to each other with connectivity defined as shown below. Six tangent constraints can be added, modified, and deleted without any pre-set order. In addition, different constraint schemes can be used to create the same physical model as presented for part model creation in the previous subsection.

Each instance in an assembly contains several half space entities. Therefore, the new transformation of the instance is derived from the solved entities data. In the assembly application, VGX supports the part instance to be either rigid or variational, depending on the user's intent. The variational part network can be combined with the assembly network and solved simultaneously.

In addition, with the mathematical model embedded inside the constraint system, the remaining degrees of freedom of each rigid instance can be determined. Graphic display of animation along each DOF greatly enhances the understanding of the assembly model status.

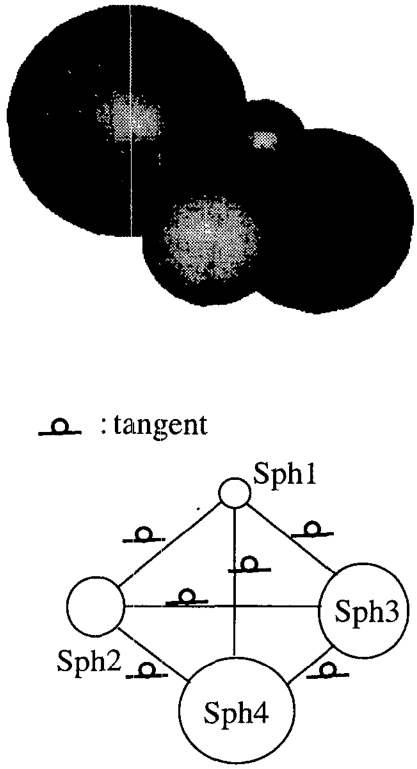


Figure 4.2 3D flexible constraining

Design changes can be made and automatically propagated to the entire system without any burden to redefine the constraints. Figure 4.3 shows the result of flipping all six tangent constraints without any constraint deletion. Not only can the users modify constraints and dimensions, they can also change the shape and size of part instances within the context of the assembly.

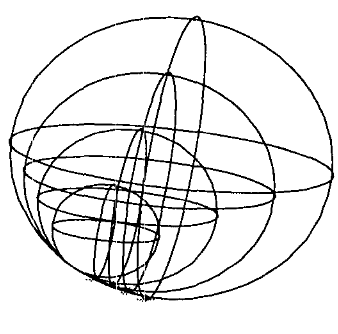


Figure 4.3 Design modification

5 DRAG AND DROP USER INTERACTION PARADIGM

Interactive modeling has existed for years, beginning with Ivan Sutherland's Ph.D work on the Sketchpad drawing system[20]. However, its use was limited to a simple system that involved no constraint and history. Eric A. Bier [3] presented an alternative called "dynamic snapping" for two-dimensional and three-dimensional design schemes which provided a means for the user to align objects interactively. These early systems, however, lacked sophisticated mechanisms for preserving the user's intent during dragging.

The power of VGX comes from its ability to decompose a complex constraint system into simple sequential algebraic or numerical solutions. VGX's excellent performance, stable solution-finding capability and ability to preserve user intent intuitively even for under-constrained models [17] enables the deployment of a drag and drop paradigm in the design process. Drag and drop provides a real time, dynamic, interactive, direct manipulation of design models. It allows the user to change size, shape and position of a selected geometric entity dynamically according to mouse movement as shown in a common extrude feature in Figure 5.1.

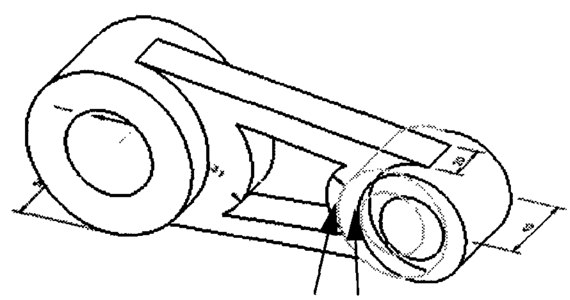


Figure 5.1 Change size, shape and position

Figure 5.2 shows how the system honors existing constraints and recognizes new constraints during dynamic dragging. Drag and Drop is a highly productive tool for the creation and modification of design models. It shortens the learning curve of a new user and enables any user to get to the final design faster and more precisely.

Figure 5.3 shows the schematic diagram for drag and drop support in a VGX system. A user first picks an entity with the cursor, then moves the cursor to the desired position. In the mean time the system interprets the mouse location and invokes the VGX solver, which in turns gives real time, dynamic, interactive feedback to the user. Users can explore and preview design variations before committing to the actual changes. Users can drag geometric entities or dimensions. During dragging, all user intent will be preserved. For example, a line knows how to stay on a surface and maintain an angle to another line. The dragged entity can recognize and snap to other entities with the recommended constraint. To tell the user which entity is being recognized and with what type of constraint, the system will display the recognized entity with a highlighted color, display the symbol of the type of constraint recommended, sound a bell, and flash the recognized entity. If the result is accepted, the design is updated and the constraint is added between the entities. It is anticipated that the coupling of drag and drop functionality with multi-media support will propel future virtual product design and product assembly.

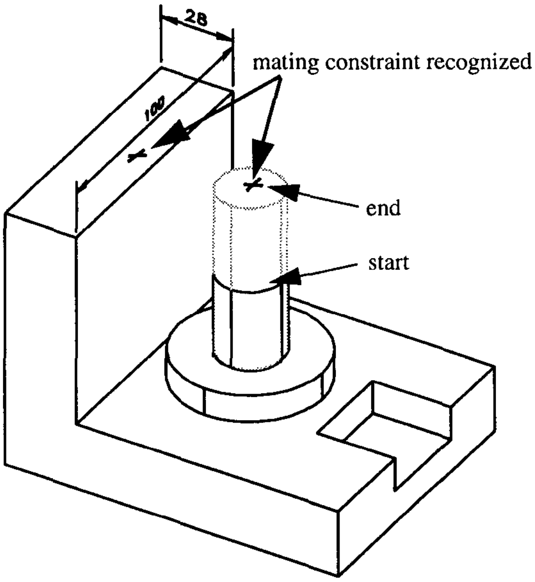


Figure 5.2 Dynamic constraint recognition

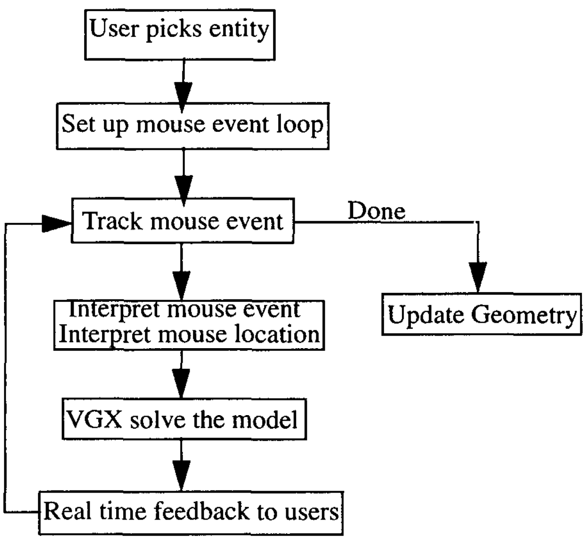


Figure 5.3 Drag and Drop paradigm in VGX system

6 INTEGRATED APPLICATIONS

Downstream applications such as tolerance and mechanism analyses are very useful to validate or refine a product design. The lack of a complete mathematical description in a conventional system severely restricts its ability to use a common model for design and analyses, which necessitates special expertise to rebuild analysis-specific models. The model-rebuilding process is very time-consuming and error-prone.

Since a complete mathematical description of a design is available in the VGX-based design model, critical mathematical calculations such as sensitivities or derivatives needed for downstream analyses can be directly obtained without further user input. For example, velocity and acceleration would be the first and second order derivatives of the generalized coordinates $\mathbf{q}$ with respect to time, and the sensitivity of the target dimension with respect to other independent dimensions is needed for worst case and statistical tolerance calculation. Based on Eq. 2 and the Implicit Function Theorem [21], the dependent variable $\mathbf{d}$ can be expressed implicitly as functions of the independent variables $\mathbf{c}$. The variational form of $\mathbf{d}$ can be expressed as $\delta\mathbf{d} = \mathbf{d_c}\delta\mathbf{c}$ where $\mathbf{d_c}$ is the partial derivative of $\mathbf{d}$ with respect to $\mathbf{c}$, and $\mathbf{d_c}$ can be calculated by taking the differential of Eq. 2 to get

$$\mathbf{\Phi_d}\,\delta\mathbf{d} + \mathbf{\Phi_c}\,\delta\mathbf{c} = \mathbf{0} \qquad (3)$$

Substituting $\delta\mathbf{d} = \mathbf{d_c}\delta\mathbf{c}$ into Eq. 3, the following relation can be obtained

$$\mathbf{d_c} = -(\mathbf{\Phi_d})^{-1}\,\mathbf{\Phi_c} \qquad (4)$$

The second order derivatives of $\mathbf{q}$ with respect to time can be derived from the same concept; i.e.,

$$\mathbf{q}_{tt} = -(\mathbf{\Phi_q})^{-1}\,[(\mathbf{\Phi_q}\,\mathbf{q}_t)_\mathbf{q}\,\mathbf{q}_t + 2\mathbf{\Phi}_{\mathbf{q}t}\,\mathbf{q}_t + \mathbf{\Phi}_{tt}] \qquad (5)$$

Eq. 4 is used for tolerance sensitivity and mechanism velocity calculation and Eq. 5 is used for mechanism acceleration calculation.

The ability to use a common model for design and analyses can significantly improve product development productivity. Two application areas in 3D tolerance analysis and mechanism analysis are further discussed below.

6.1 3D Tolerance Analysis

Tolerance selection and verification can have a significant impact on the cost and quality of the final product. Tolerance analysis provides the tools to analyze a critical dimension in a design model and to understand how it is impacted by the tolerances of other dimensions within a part or an assembly.

A variational design system represents the design model in terms of constraint/dimension variables and equations. Any target dimension Y to be analyzed can be considered to be a function of other constraints/dimensions (C_i) in the design model. Therefore, $Y = f(C_1, C_2, \ldots, C_n)$. The variations of $C_1, C_2, \ldots, C_n$ will contribute to the variation of Y. The worst case tolerance and statistical tolerance of Y can be calculated by Eqs. 6 and 7 respectively.

$$T_w = \sum_{i=1}^{n} \left| \frac{\partial f}{\partial C_i} \right| T_i \qquad (6)$$

$$T_s = \sqrt{\sum_{i=1}^{n} \left[\frac{\partial f}{\partial C_i} T_i \right]^2} \quad (7)$$

Where T_i is the tolerance of dimension C_i and the partial derivatives of f with respect to C_i can be calculated from Eq. 4.

The unique feature of tolerance analysis based on variational design is that the partial derivatives are the natural by-product of the variational solution process. Figure 6.1 shows a variational design model. Figure 6.2 shows the tolerance analysis result for dimension D38. The sensitivity of the target dimension with respect to each dimension and the contribution of each dimension shown in the table will help users to identify critical dimensions to achieve better design results.

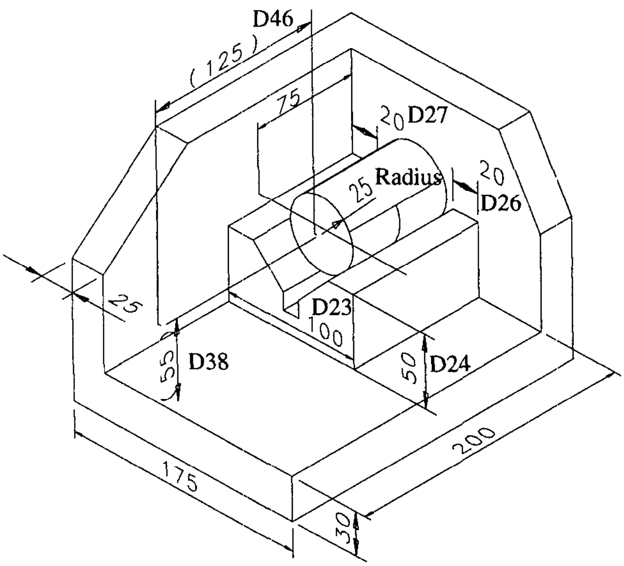
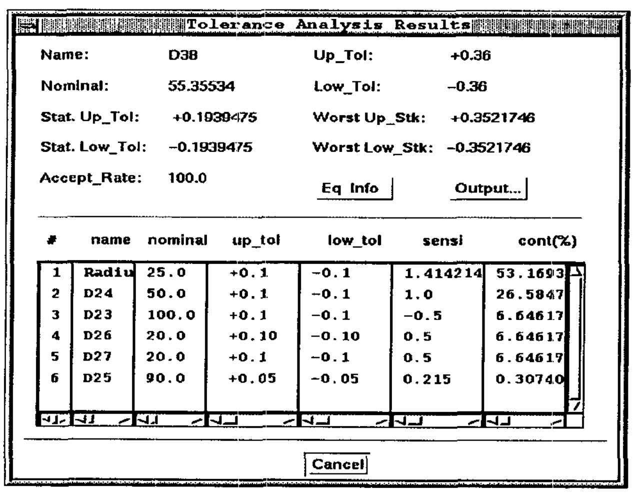


Figure 6.1 A 3D variational design model



Tolerance Analysis Results

Name: D38		Up_Tol:	+0.36
Nominal:	55.35534	Low_Tol:	-0.36
Stat. Up_Tol:	+0.1939475	Worst Up_Stk:	+0.3521746
Stat. Low_Tol:	-0.1939475	Worst Low_Stk:	-0.3521746
Accept_Rate:	100.0	Eq Info	Output...

#	name	nominal	up_tol	low_tol	sensi	cont(%)
1	Radiu	25.0	+0.1	-0.1	1.414214	53.1693
2	D24	50.0	+0.1	-0.1	1.0	26.5847
3	D23	100.0	+0.1	-0.1	-0.5	6.64617
4	D26	20.0	+0.10	-0.10	0.5	6.64617
5	D27	20.0	+0.1	-0.1	0.5	6.64617
6	D25	90.0	+0.05	-0.05	0.215	0.30740

Cancel

Figure 6.2 Tolerance analysis result for D38

6.2 3D Mechanism Analysis

VGX can perform mechanism analysis using assembly constraints and dimensions defined within variational assembly models. This capability allows an engineer to quickly evaluate the kinematics and dynamics of a mechanism without the burden of entering redundant information, such as joint definitions, in a format that mechanism analysis packages can understand. Furthermore, there is no need for rebuilding the mechanism models after making design changes to the MDA models.

The mathematical foundation for variational assembly based mechanism analysis is built on top of the first and second order derivative calculations. To determine the motion of the system, the engineer needs to define either additional driving functions that uniquely determine **q**(t) (kinematic analysis) or forces that act on the system, in which case **q**(t) is the solution of differential equations of motion (dynamic analysis). As shown in Figure 6.3, a robot is built using geometric constraints. Connectivity is depicted below. An inverse kinematic analysis is performed to control the end point of the robot arm to follow a circular trajectory with the orientation of the end instance fixed. The angle between body 3 and body 4 versus time and the acceleration in the X direction of the end effector are shown in Figure 6.4.

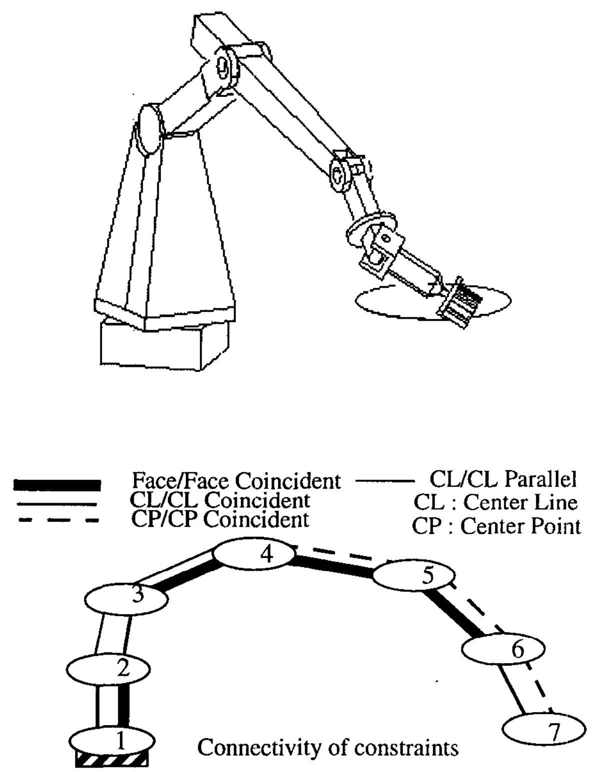


Figure 6.3 A robot assembly model

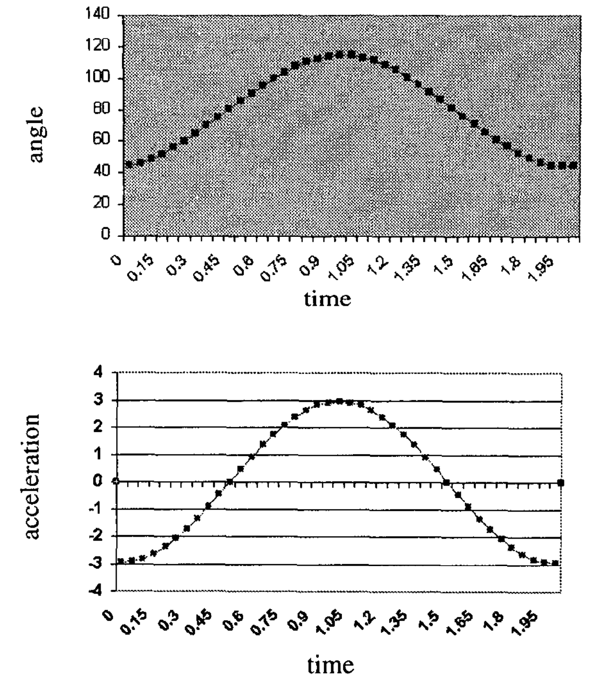


Figure 6.4 Inverse kinematic analysis result

7 CONCLUSION AND FUTURE WORK

In this paper, we have presented VGX as a unified framework towards supporting a more complete design process: from design specification, conceptual design, detailed design, to design validation. VGX technology provides a complete mathematical description of the design model which allows downstream tasks such as mechanism analysis, tolerance analysis and design optimization to be performed without having to rebuild analysis-specific models. In addition, the drag and drop user interaction paradigm when coupled with the VGX technology provides unprecedented ease of use when manipulating 3-dimensional design models. Overcoming the limitations and temporal barriers of the conventional history-based systems is one of the major contributions of this research.

To fully realize the productivity gains of VGX technology, significant future work in the following areas is needed:

- Performance and Robustness
  Though VGX technology provides excellent performance for moderate constraint problems (< 10,000 DOF's), as the complexity of designed products grow, the constraint network size will increase accordingly. Innovative algorithms are needed to meet the challenge of maintaining user interactivity. Furthermore, weaknesses in traditional numerical solvers such as non-convergence, solution jumping, ill-conditioning, poor initial conditions, and large perturbations are not acceptable. New methodologies to ensure the robustness of the solver are also needed.
- User Aids for Constraint Comprehension
  It is unreasonable to expect average MDA users to understand the mathematical complexity and intricacies of constraint formulation and solving. They must be provided with tools to easily visualize free degrees of freedom and constraint conflicts, or to show why no valid solution exists.
- Direct B-rep Manipulation
  In a history-based MDA system, any change to the model needs to trigger a history replay. This is not only time-consuming, but also makes it difficult to understand the impact of changes beforehand. VGX technology offers potentials for interactive and controlled localized changes which may enable direct B-rep update without the time-consuming replay process.
- Downstream Analyses Development
  In this paper we demonstrated the feasibility of applying VGX technology to mechanism and tolerance analyses. Much more work is needed to develop full-fledged mechanism and tolerance analyses. Furthermore, applications of VGX technology to design optimization, sculptured shape design, and 3D wireframe manipulation can also be explored.

ACKNOWLEDGMENTS

We wish to express our appreciation to all members of the VGX team at SDRC for their contribution in the development and implementation of VGX technology. We also would like to thank Naomi Lokay for reading and editing the manuscript and making many helpful suggestions.

REFERENCES

[1] Aldefeld, B., "Variation Of Geometry Based On Geometric Reasoning Method," *Computer-Aided Design*, Vol. 20, No. 3, April 1988.
[2] Anantha, Ram, Kramer, Glenn A., and Crawford, Richard H., "Assembly Modeling By Geometric Constraint Satisfaction", *Computer-Aided Design*, Vol. 28, No. 9 1996.
[3] Bier, Eric A., "Snap-Dragging: Interactive Geometric Design In Two And Three Dimensions," *Ph.D. Dissertation*, U.C. Berkeley, 1988.
[4] Borning, A., "Thinglab - An Object-Oriented System For Building Simulations Using Constraints," *Proc. Fifth International Joint Conference on Artificial Intelligence*, 1977.
[5] Bruderlin, B., "Using Prolog For Constructing Geometric Objects Defined By Constraints," *Proc. European Conference on Computer Algebra*, 1985.
[6] Chung, J. C. H., Klahs, J. W., Cook, R. L., and Sluiter, T., "Implementation Issues In Variational Geometry And Constraint Management," *Third International Conference on MDA/CAM, Robotics and Factory of the Future*, August 14-17, 1988.
[7] Chung, J. C. H., and Schussel, M. D., "Technical Evaluation Of Variational And Parametric Design," *Proc. ASME International Computers in Engineering Conference*, 1990.
[8] Fudos, I., and Hoffmann, C. M., "A Graph-Constructive Approach To Solving Systems Of Geometric Constraints," *ACM Trans. on Graphics*, Vol. 16, No. 2, April 1997.
[9] Hillyard, R. C., and Braid, I. C., "Analysis Of Dimension And Tolerances In Computer-Aided Mechanical Design," *Computer-Aided Design*, Vol. 10, No. 3, 1978.
[10] Hornbeck, R. W., *Numerical Methods*, Quantum Publishers, 1975.
[11] Hsu, C., "Graph-Based Approach For Solving Constraint Problems," *Ph.D. thesis*, Computer Science Department, University of Utah, May 1996
[12] Kimura, F., Suzuki, H., Ando, H., Sato, T., and Kinosada, A., "Variational Geometry Based On Logical Constraints And Its

Applications To Product Modeling," *Annals of the CIRP*, Vol. 36, No. 1, 1987.

[13] Kramer, G., "Using Degrees Of Freedom Analysis To Solve Geometric Constraint Systems," *Proc. ACM Solid Modeling Conference*, Austin, TX, 1991.

[14] Light, R. A., and Gossard, D. C., "Modification Of Geometric Models Through Variational Geometry," *Computer-Aided Design*, Vol. 14, No. 4, 1982.

[15] Lin, V. C., Gossard, D. C., and Light, R. A., "Variational Geometry In Computer-Aided Design," *Computer Graphics*, Vol. 15, No. 3, 1981.

[16] Newell, R. G., and Parden, G., "Parametric Design In Medusa System," in Warman, E. A. (ed.) *Computer Appl. in Production and Engineering*, North Holland Publishing Co., 1983.

[17] Owen, J. C., "Algebraic Solution For Geometry From Dimensioned Constraints," *Proc. ACM Solid Modeling Conference*, Austin, TX, 1991.

[18] Roller, D., "An Approach To Computer Aided Parametric Design," *Computer-Aided Design*, Vol. 23, No. 5, July 1991.

[19] Serrano, D., and Gossard, D. C., "Combining Mathematical Models With Geometric Models In CAE Systems," *Proc. ASME Computer in Eng. Conf.*, Chicago, IL, July 1986.

[20] Sutherland, I.E., "SKETCHPAD: A Man-Machine Graphical Communication System," *SJCC1963*, Spartan Books, Baltimore, Md. pp329

[21] Taylor, A. E. and Mann, W. R., (1976), *Advance Calculus*, 3rd ed., John Wiley & Sons, New York.

[22] Todd, P., "An Algorithm For Determining Consistency And Manufacturability Of Dimensioned Drawings," *Proc. CAD86*, Butterworths, 1986.