Artificial Intelligence for Engineering Design, Analysis and Manufacturing (1998), *12,* 419–429. Printed in the USA.

Precision machine design assistant: A constraint-based tool for the design and evaluation of precision machine tool concepts

BRADLEY S. HOMANN AND ANNA C. THORNTON

Mechanical Engineering Department, Massachusetts Institute of Technology, 77 Massachusetts Avenue, Cambridge, MA 02139, U.S.A.

(RECEIVED November 20, 1997; ACCEPTED June 1, 1998)

Abstract

Precision machines are necessary to manufacture parts and subassemblies that require tight tolerances. During the design of precision machines, like any design, it is critical that the best concept is chosen in the early stages of the design process because 80% of the final cost and quality of a product are designed in at this phase. In addition, changes and optimization late in the design process have limited impact on cost and quality. Typically, during the design of precision machines, engineers and skilled machinists develop several machine concepts and down select based on heuristics and past design experience rather than quantitative measures. This paper describes a computation tool, Precision Machine Design Assistant (PMDA), which automates basic machine error simulation and concept evaluation. The tool uses a combination of machine error motion modelling and constraint-based design methods. By combining these methods in a computational environment, multiple machine concepts may be rapidly modeled, analyzed, and compared. The goal of the program is to assist the designer in the selection of a superior concept for detail design. The PMDA methods and implementation are demonstrated in an example.

Keywords: Precision Machine Design; Error Budgets; Constraint-based Design; Design Automation

1. INTRODUCTION

Precision tools are needed to manufacture parts and subassemblies that require tight tolerances. Precision tool performance is defined by the accuracy, resolution, and repeatability of the machine. In addition, cost constraints are imposed by the customer. Typically, engineers and skilled machinists develop several machine concepts and, from that set, select a single concept that they think will achieve the machine requirements. Down selection is often based on heuristics and past design experience (Slocum, 1992). Once the detail design phase is completed, the machine is produced, tested, and modified.

Eighty percent of the final cost and quality of a product are designed in during the early stages of design. In addition, design changes and optimization late in the design process can have only limited impact on cost and quality. As a result, it is critical that the "best" concept is selected in the early stages of design. However, making informed decisions early in the design process is hard because the decision making process is done under time constraints and uncertainty. The authors have identified a need for better tools and methods to quantitatively and quickly evaluate precision machine concepts in the early design phase.

1.1. Conceptual design processes

Concept evaluation is the process of determining the feasibility and expected performance of a new precision machine design. While some aspects of machine tool behavior are not well understood, the majority of influences on overall machine errors are deterministic and can be predicted and included in the evaluation of individual machine concepts (Bryan, 1984). Ideally, quantitatively based estimates of performance characteristics such as machine accuracy, repeatability, resolution, and cost, are made for each concept. In addition, component parts, materials, and manufacturing processes are selected and evaluated during this phase.

Reprint requests to: Anna C. Thornton, Mechanical Engineering Department, Massachusetts Institute of Technology, Room: 3-449, 77 Mass. Ave., Cambridge, MA 02139, U.S.A. E-mail: acthornt@mit.edu

Despite the availability of quantitative methods, analysis is normally limited to simple calculations for individual component sizing or reaction force prediction. More advanced computational techniques are used later in the design process when the concept is finalized. In most cases, analysis of the relative performance of a set of concepts in the early stages is either cursory or not performed.

1.2. Computational design tools

Several researchers are developing tools to enable early modelling of design concepts. Many of the tools described in the literature are focused on the concept generation phase. These methods include mapping function to form, combinations of elements, and other geometry generation mechanisms. For example, Chakrabarti et al. (1992) and Li et al. (1996) have developed a system to generate machine tool concepts using a variety of analytic and heuristic-based approaches. Gorti and Sriram (1995) developed a system for the geometric design of a part based on function-symbol and symbol-form mapping. Colton and Dascanio (1991) developed the Intelligent Design System (IDS) for the geometric design of parts, allowing the designer to specify geometry and function based on a library of key mechanical features. Kannapan and Marshek (1990) propose a system for modelling machine elements based on first-order predicate and algebraic logic for design verification and selection.

In addition, a variety of commercial systems exist that can simulate the kinematic motion of a mechanism. Two examples are ProEngineer's Mechanism/Pro and Computer Aided Design Software's DADS. These systems are effective at modelling the motion of mechanisms and can be used to evaluate some basic constraints on geometry and forces. However, the constraint sets are limited in scope.

1.3. Computational methods for precision machine design

Two analyses should be included in a computational system to evaluate precision machines. The first is a set of tools to ensure that the accuracy, repeatability, and resolution requirements are achieved. The second is a set of tools to ensure that the design satisfies nonerror related design requirements such as cost, footprint, and travel, as well as, geometric fit between parts. The first can be achieved through the use of error analyses such as *error budgets* and the second achieved by using *constraint-based* design methods. Both impose limitations on the design space and must be evaluated simultaneously. For example, changes to minimize cost will have an impact on the error budgets, and selection of more precise parts, such as a bearing, may require changes to the geometry of shafts and housings.

1.3.1. Error budget

The performance of machine tools is highly dependent on the errors in the parts and their combined effect on the accuracy, repeatability, and accuracy of the tool/workpiece interaction. The errors associated with the components in a machine tool may be classified into four major types: geometric (deviations of the form of the component), thermal (deformations due to temperature gradients), load (induced deformations due to external or internal loads), and dynamic (deformations due to friction and vibration). While the sources of errors influencing the performance of a precision machine tool are diverse, the effects are not. Errors in components lead to errors in tool position which, in turn, result in variation in the parts produced. For example, wear in a grinding wheel will result in oversized parts.

Error budgets are used to quantify the maximum variation that can be withstood by a design (Slocum, 1992). One way of evaluating whether or not an error budget has been satisfied or violated is through the use of a kinematic model. This model is comprised of complex series of homogeneous transform matrices (HTMs) that describe how errors are propagated from a part to the error-sensitive areas of a design. These methods allow the designer to assign values to the component errors and determine the total error at the tool/workpiece interface point. In addition, the models can be used to determine the components most heavily influencing the tool/workpiece error. Limits on best and worst case performance may also be determined by varying the errors of the individual components in the machine.

The kinematics models used in error budgets use the same mathematics employed by commercial kinematics software but apply it differently. Commercial kinematic systems are used by designers to model the motion of a mechanism, to validate the design and to check for part-to-part impacts. The kinematic models discussed here can be used to model the motion of the machine but are primarily used to determine the results of errors on the final location of the tool and/or workpiece.

Kinematic models effectively quantify the effect of part and interface errors on machine performance; however, they must individually be built for each concept layout and component set. Unfortunately, error budgets are rarely rigorously applied during the concept design because manually developed error budgets are time consuming to build and difficult to modify. Most often, they are used to validate a design once the detail design is complete.

1.3.2. Constraint-based design

Error budgets are not enough to ensure the quality of a precision machine design. The selection of components, geometry, and dimensions are also constrained by geometric limitations and functional requirements (e.g., travel, cost, and footprint). These constraints can usually be represented mathematically as equality and inequality relationships between expressions.

For a large design, managing constraints has two difficulties: the specification of constraint sets and the search for a solution (Thornton & Johnson, 1996). First, each part and interface imposes constraints on a design. For example, a bearing imposes multiple load, environmental, and geometric constraints. It is difficult for a designer to ensure that all constraints have been specified. Second, the solution is hard to find because most of the design constraints are inequalities, nonlinear, and highly coupled. Searching for solutions in this type of space is difficult because it has many local minima and the feasible space is highly constrained.

1.4. Paper description

This paper describes a computation tool, Precision Machine Design Assistant (PMDA), to assist in basic machine error simulation and concept evaluation through a combination of machine error motion modelling and constraint-based design methods.

The rapid and robust specification of the kinematic models, errors, and constraints on the parts and interfaces is enabled through a standard library of common precision machine elements. These elements (i.e., bearings, rails, shafts, couplings, etc.) contain geometry, constraints, tables of commercially available solutions, and errors. In addition, a set of library interfaces is available that know how to connect between the library components. The interface objects contain the HTM connections and the interface constraints.

By combining error budgets, constraint engines and library systems in a computational environment, multiple machine concepts may be rapidly modeled, analyzed, and compared. This assists the designer in the selection of a superior concept for detail design. The key benefits come from the automatic generation and analysis of the HTM model and from the specification and management of a large set of coupled constraints. The rest of the paper describes the background theory, the PMDA tool, and an example of a PMDA model.

2. PMDA ELEMENTS

This section describes the theory behind the kinematic models and constraint-based methods used by PMDA. In addition, the use of standard elements to specify the error budgets and the constraints is described.

2.1. Error budgets

The first step in creating an error budget is to develop a kinematic model of the machine tool. The kinematic model is comprised of a series of HTMs that describe how errors are propagated from a component to the final tool/workpiece interaction. The kinematic model uses two types of HTMs: the HTMs that describe the part errors and the HTMs that describe how the parts are interconnected. Total machine performance is determined by including all of the component errors simultaneously, while sensitivity to a given error is determined by applying the errors individually. Upper and lower bounds on performance may be determined by modelling the best and worst cases.

2.1.1. Homogeneous transformational matrices

The core theory required for kinematic models is the specification and multiplication of the HTMs. A kinematic model translates and rotates errors by multiplying a set of 4-by-4 matrices. These transformations take place about the coordinate system of the component and allow the component error motions to be combined and reflected in the adjoining components. An overview of HTM modelling process is given below and is described in detail by Slocum (1992).

A standard set of HTMs is used to describe a set of translation and rotation errors. For example, a rotation through the angle θy about the Y axis is represented by Eq. (1).

$$ {}^{\mathrm{ref}}T_{\mathrm{local}} = \begin{bmatrix} \cos\theta_y & 0 & \sin\theta_y & 0 \\ 0 & 1 & 0 & 0 \\ -\sin\theta_y & 0 & \cos\theta_y & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{1} $$

Two similar matrices are used to rotate about the Y and Z axes. The HTM to translate a point by an amount ∂x, ∂y, and/or ∂z is shown in Eq. (2).

$$ {}^{\mathrm{ref}}T_{\mathrm{local}} = \begin{bmatrix} 1 & 0 & 0 & \delta_x \\ 0 & 1 & 0 & \delta_y \\ 0 & 0 & 1 & \delta_z \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{2} $$

For translational errors, the HTM for a component is built by decomposing each error into its X, Y, and Z components and placing them in the HTM described in Eq. (2). A similar process is used for rotational errors, however, the modelling process requires care as the order of multiplication is critical and subject to human error.

2.1.2. Kinematic modelling

The creation of a kinematic model involves determining the HTM for each component, as well as the HTMs that relate the parts to each other. First, the part HTMs are specified. Second, starting with the tool point, the HTM between adjoining components are specified, in series, until the reference coordinate frame for the machine is reached (i.e., ground). A similar process is completed for the workpiece and associated components. The end result is a series of HTMs describing the location of the tool point and the workpiece with respect to the machine reference frame. The errors are then specified and propagated through the HTMs to determine the final tool/workpiece error. This process must be repeated for each geometric operating condition of the machine.

2.1.3. Modelling difficulties

Four problems exist with the current manual specification of HTMs. First, as stated above, the kinematic model requires a significant amount of effort and time to construct. A typical HTM model for a precision machine tool can require up to several weeks to build and test. Second, it is difficult to ensure that all of the appropriate errors are included in the model. Third, the validity of the model requires the modeller to carefully specify the order of multiplication of rotational elements as well as ensure consistency in the reference frames. This is nontrivial for complex machines. Fourth, even small changes in the concept can require major modifications to the model. Because of these four problems, many designers develop overly simplistic error budgets or rely on intuition and experience to evaluate concepts.

2.2. Constraint-based design methods

A constraint is a relationship involving one or more parameters of a design. Development of any product can be viewed as a process of specifying and satisfying a set of such constraints (Thornton & Johnson, 1996). The responsibilities of the design team include the specification of the constraints and the selection of components and their dimensions to satisfy the set in an "optimum" fashion. However, the process of specifying and satisfying all of the constraints is complex. This complexity is, first, due to the inclusion of an individual design parameter in multiple constraints (i.e., coupling) (Watton, 1989). Second, the number of constraints associated with even a simple design can be large. Manually specifying and maintaining all of the necessary relationships can be time consuming and prone to error.

Several software systems, commercial and research, enable the management and solution of equality and inequality constraints. These systems use formalized constraint representation to manipulate and evaluate a large number of constraints. For example, *Design Sheet*, (Reddy & Fertig, 1996) developed by Rockwell Science Center allows for many equality constraints to be solved simultaneously. In addition, work has been done to find solutions to large sets of inequality and equality constraints using stochastic search algorithms (Thornton, 1996).

The use of computer-based tools for constraint management can be approached in one of two ways. The first requires the designer to describe the geometry and constraints for every component in a design. The second provides a library of generic components, interfaces, and features for inclusion within a design. The first approach is extremely time-consuming for the designer. The second approach works only if an adequate library of components, interfaces, and features is provided.

2.3. Design element library

The justification for using the second approach is based on what Johnson and Thornton (1991) called "the reasoned assumption that a wide range of engineering designs can be modeled using a reasonable and manageable number of function elements."

Most precision machines are comprised of a standard element such as a base, rails, connectors and bearings. A new precision machine has a limited number of "unique" parts. Although novel geometry may be used in the structural elements, the function and interaction of the structural elements with the other parts is consistent. As a result their geometry does not need to be specified at the concept stage of the design and a more generic "support" object can be used (Thornton & Johnson, 1996).

This assertion has been justified by analyzing several machine designs. Thornton (1993) described an analysis of the bill of materials of several machines—a grinder, lathe, and mill. Of the 2300 parts in the three machines, less than 150 (less than 8%) were classified as unique parts (i.e., those that could not be found in standard machine handbooks) (Spotts, 1978; Shigley & Mitchell, 1983).

3. PRECISION MACHINE DESIGN ASSISTANT

The PMDA software tool was developed to enable a design team to rapid model and evaluate a variety of machine tool concepts. The basic assumption behind the PMDA approach is that most of the components and interfaces in a precision machine are "standard."

The commonality of the elements in precision machines is used to enable rapid concept evaluation. PMDA modelling process is based on a library of standard elements that are used as the building blocks of the design. Each standard element contains the constraint and error information required to automatically build the error budgets and the constraint networks.

The process of modelling a machine tool in PMDA is shown in Figure 1. The process begins with the specification of the design elements and their interactions. Once the layout is completed, PMDA constructs the kinematic model of the machine and creates the constraint network. The total tool/workpiece error is then calculated, the constraint network is evaluated, and the results are presented to the user. The following sections describe the standard components, the automatic kinematic model builder, and the constraint engine.

3.1. Standard components and interface

The machine elements included in the design are selected by the user based on their ability to perform the required tasks within a prescribed set of conditions. PMDA enables three types of parts to be included in a design: standard parts that are selected from a catalogue (e.g., a bearing), parts that can be represented parametrically (e.g., a shaft), and unique parts that can be modeled by the user.

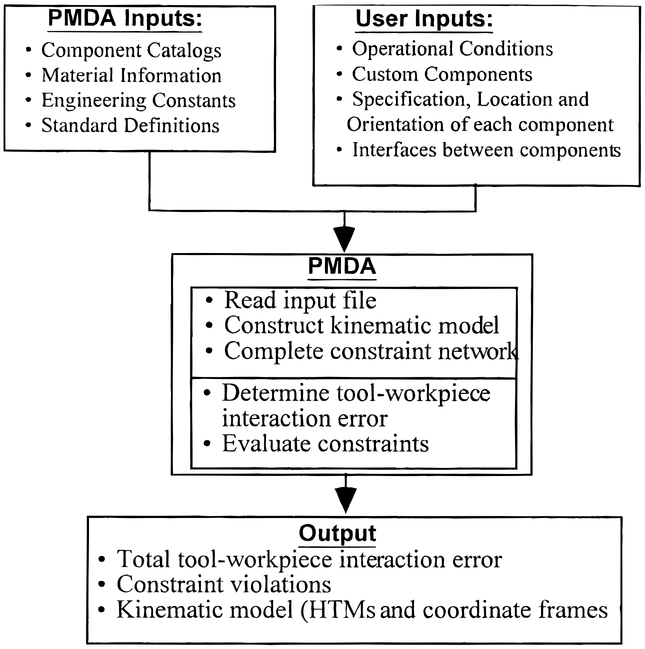


Fig. 1. PMDA software architecture.

PMDA use an object-oriented approach to modelling and analysis. Each component is modeled as an independent computational entity based on a standard representation of the mechanical component. Increasing levels of detail for each type of component are provided by additional levels of inheritance. The connections between components, referred to as interfaces, are also represented as objects.

3.1.1. Component representations

The object hierarchy for PMDA components is presented in Figure 2. The top level object, common to all components, is the *Block* class. It contains a variety of data including the component HTM, basic geometry, and the data required to connect a part to other components in the machine model. Specifically, the information stored in the *block* class includes the origin and normalized axes of the component coordinate system, the HTM relating this block to the previous block in the structural loop, and the maximum geometry (bounding box) of the component.

The next level of abstraction contains the generic component objects, or *templates*. Templates contain the definitions of component objects in PMDA. The templates also contain the geometry, constraints, the expected errors in the parts, and how these errors will be built into the HTM for the part. Values for the specific variables are provided by the third level of objects which contains information specific to the particular instance of the object. Both custom and manufacturer-specific information is found at this level. Custom objects are included as "blank" templates that allow the user to define the product dimensions and errors. Manufacturer-specific objects contain catalog-based, commercially available components. The valves are provided by the catalogs.

3.1.2. Interface objects

As stated in the previous section, two types of HTMs are used in the kinematic model: component-specific information and component-connectivity information. In PMDA, component-connectivity information is contained in the *interface* objects. An interface object is a computational representation of the physical interaction between two or more compatible mechanical components. For example, a circular workpiece is connected to a rotary-motion chuck through a clamping process.

The interface provides several types of information. First, the complete collection of interfaces in the machine tool model defines the structural loop from tool to workpiece. Second, interfaces impose constraints on the design. For example, a rotary bearing attached to a shaft must have the same rotational velocity as the shaft. Third, the interface class provides for error motions associated with the interfaces between components. For example, in a chuck/workpiece interface, mounting errors between the workpiece and the chuck will result in workpiece error motions.

Unlike the Block class for component objects, interfaces require a separate and complete representation for each pair of objects and can not use inheritance to reduce the object specification work. The library will theoretically require n^2 interfaces to be specified to capture all possible combinations of n elements. However, in reality, the interface set is much smaller because many interfaces in the set of n^2 interfaces are nonsense interfaces (e.g., the interface between a chuck and a bearing does not need to be specified). By filtering out the nonrational interfaces, library requirements have been significantly reduced.

3.2. Kinematic machine modelling

In PMDA, a kinematic machine model describes the position, connectivity, and relative locations of each component included in the machine tool assembly. This is accomplished through the use of HTMs to define the effect of component errors and the transformation between the connected components. The process for developing a kinematic model in PMDA is performed in four steps. The first two require user input and the second two are automatically executed by PMDA.

First, each component object is added to the model by the user. Second, the interfaces between the components are specified by the user. Third, as each interface is specified, the interface HTMs are built by PMDA. Fourth, PMDA defines the structural loops—the component "trails" between the workpiece and ground and the cutting tool and ground. This is done by tracing paths through the component and

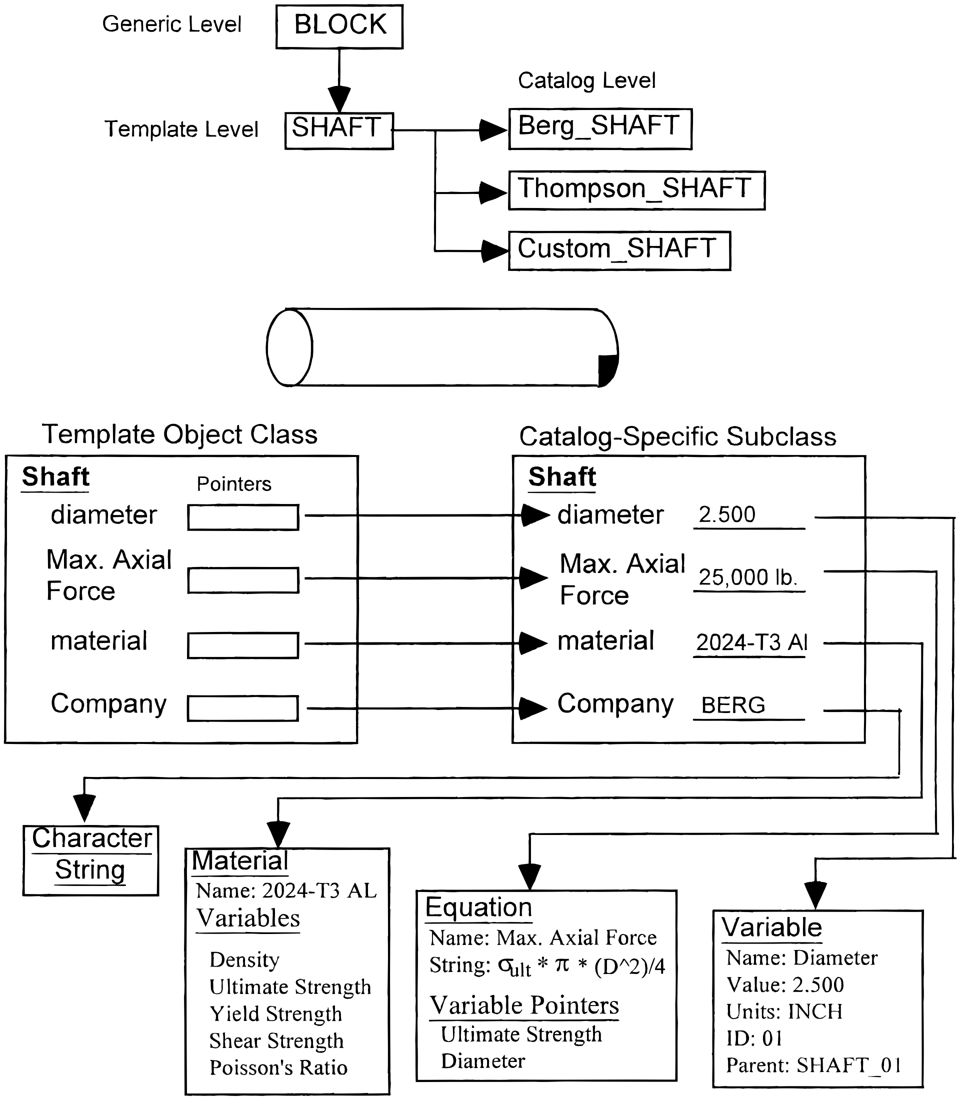


Fig. 2. Object structure.

interface descriptions and by building the list of HTMs that are multiplied to calculate the final errors.

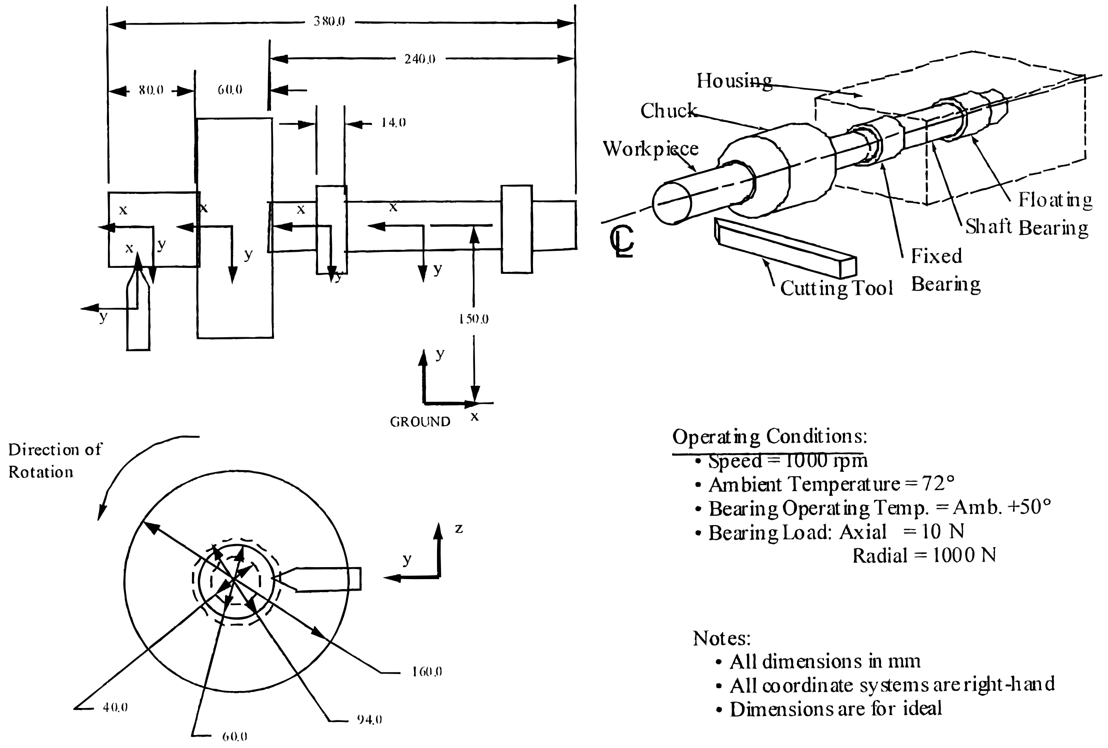
In addition to analyzing the machine in a given state, the user has the option of including time-varying effect, such as those generated by the motion of bearings and motors. These time-dependent factors are a result in changes in position and orientation of the parts and their internal defects. PMDA software includes the routines to simulate the relative position of moving elements with respect to stationary elements.

3.3. Constraint specification and evaluation

Constraint specification occurs when the component objects and interfaces are added to the model. Each component, object, and interface object has a set of constraints. As each object is added, its constraints are added to the constraint network. When it is time to evaluate the design, the value of each variable is set. Values may be set by several methods, including explicit specification by the user, calculation by equation, or automatically looked-up in a table. Evaluation of the constraint network takes place automatically after the tool/workpiece interaction errors are calculated for a machine concept. Violations are reported to the user. Currently, PMDA does not incorporate a method for automatically altering variables based on constraint violation; the values associated with a constraint violation must be adjusted manually, and the analysis is rerun to determine the effects of the change.

4. CASE STUDY—A MACHINE SPINDLE

One of the most common subassemblies used in machine tools is the rotary motion spindle. Spindles provide the rotational motion necessary for machining. To demonstrate



Operating Conditions:
- Speed = 1000 rpm
- Ambient Temperature = 72°
- Bearing Operating Temp. = Amb. +50°
- Bearing Load: Axial = 10 N
  Radial = 1000 N

Notes:
- All dimensions in mm
- All coordinate systems are right-hand
- Dimensions are for ideal

Fig. 3. Simple machine spindle model.

the functionality of PMDA, a spindle was modeled and the total tool/workpiece interaction error calculated. The constraints associated design were evaluated to determine if the spindle assembly met the performance and design requirements.

A diagram of a simplified spindle is shown in Figure 3. The shaft of the spindle is supported by two rotary-motion bearings. The forward bearing, referred to as the “fixed bearing,” is rigidly connected to the spindle housing and maintains the position of the shaft with respect to the housing. The other bearing, referred to as the “floating bearing,” is allowed to move slightly along the longitudinal axis of the spindle to allow for thermal expansion and contraction. The chuck is connected rigidly to the spindle shaft, and secures the workpiece to the spindle assembly through the use of clamps (“jaws”). The spindle is rotated through the use of a motor (not shown) attached to the end of the shaft.

Several error motions common in machine tool spindles are shown in Figure 4. Each of these errors arises from errors in either the individual mechanical components, the connection between components, or from the operation and setup of the spindle. *Axial runout* is the result of thermal growth in the shaft and the axial runout of the fixed bearing. *Radial runout* in the spindle is due to the radial runout of the fixed bearing and the off-center mounting errors between the shaft, chuck, and workpiece. *Droop* is the result of the weight of the workpiece and chuck causing the shaft to deflect. The errors are dependent on the rotational angle of the spindle (with respect to a fixed reference). In addition, the rotation of the spindle will result in changes in the deflection of the workpiece and chuck.

The following sections present the four-step method for analyzing this design with PMDA. The four steps are com-

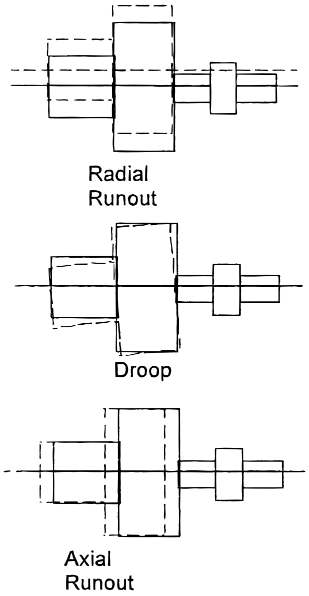


Fig. 4. Common machine spindle error motions.

ponent selection, kinematic modelling, constraint evaluation, and error calculation.

4.1. Component selection

The process of evaluating a machine concept with PMDA began with the selection of the components. For this example, the following components were selected for the design:

- shaft (BERG LS-1-12L)
- rotary bearing (NTN NU7204)
- chuck (custom, 60 mm × 160 mm diameter, three-tooth jaw)
- workpiece (custom, 80 mm × 60 mm diameter)
- cutting tool (custom, 80 mm × 20 mm × 20 mm)

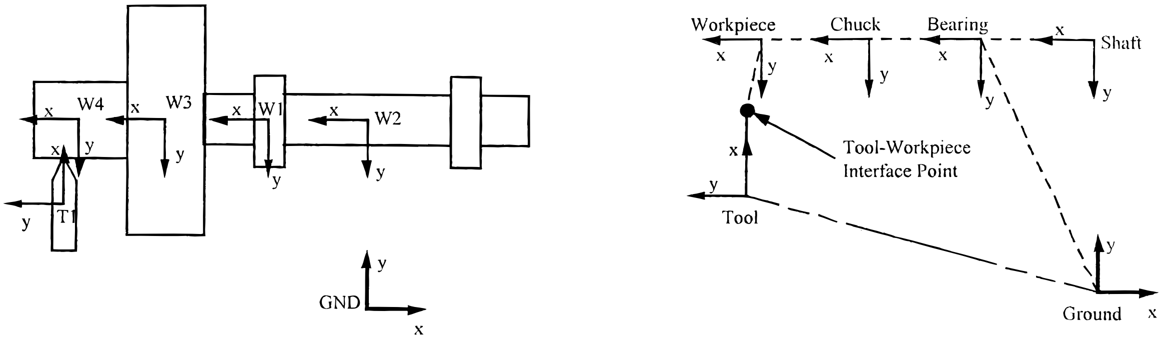
The shaft and rotary bearing were selected from catalogs read by PMDA. The remaining objects were custom objects whose parameters were specified by the user.

4.2. Kinematic modelling

After the mechanical components were selected and the component objects were added to the model, the kinematic model of the machine was built by linking the components through their interfaces. Five interfaces were used: Rotary Bearing to Ground, Shaft to Rotary Bearing, Chuck to Shaft, Workpiece to Chuck, and Tool to Ground.

The second step in the creation of the kinematic model involves the specification of the position and orientation of each component relative to each other. The standard interfaces "know" how they connect together and the user simply inputs the relative orientation of the parts. Once this step was completed, the HTMs were created automatically by PMDA. The positions and orientations of each component in the machine spindle are shown in Figure 5.

After the component and interface objects were created, the structural loops were defined by the PMDA system. The connections between components was assumed to be rigid, although mounting errors between components are included in the component objects. The floating bearing was assumed not to affect the error motions of the spindle assembly and was not included in the machine model. The tool loop consists only of the tool object (and the global reference frame), while the workpiece loop contains the workpiece, chuck, shaft, and rotary bearing objects. The structural loops for the machine spindle are also shown in Figure 5.



$$ {}^{W3}HTM_{W4} = \begin{bmatrix} 1.0 & 0.0 & 0.0 & 70.0 \\ 0.0 & 1.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 1.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 1.0 \end{bmatrix} $$

$$ {}^{W2}HTM_{W3} = \begin{bmatrix} 1.0 & 0.0 & 0.0 & 150.0 \\ 0.0 & 1.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 1.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 1.0 \end{bmatrix} $$

$$ {}^{W1}HTM_{W2} = \begin{bmatrix} 1.0 & 0.0 & 0.0 & -90.0 \\ 0.0 & 1.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 1.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 1.0 \end{bmatrix} $$

$$ {}^{GND}HTM_{W1} = \begin{bmatrix} -1.0 & 0.0 & 0.0 & -90.0 \\ 0.0 & -1.0 & 0.0 & 180.0 \\ 0.0 & 0.0 & 1.0 & 80.0 \\ 0.0 & 0.0 & 0.0 & 1.0 \end{bmatrix} $$

$$ {}^{T1}HTM_{TW} = \begin{bmatrix} 1.0 & 0.0 & 0.0 & TL/2 \\ 0.0 & 1.0 & 0.0 & TW/2 \\ 0.0 & 0.0 & 1.0 & TH/2 \\ 0.0 & 0.0 & 0.0 & 1.0 \end{bmatrix} $$

$$ {}^{W1}HTM_{TW} = \begin{bmatrix} 1.0 & 0.0 & 0.0 & X\_TL \\ 0.0 & 1.0 & 0.0 & WW/2 \\ 0.0 & 0.0 & 1.0 & Z\_TL \\ 0.0 & 0.0 & 0.0 & 1.0 \end{bmatrix} $$

TL = Tool Length TH = Tool Height
TW = Tool Width WW = Workpiece Width
X_TL = X Position of Toolpoint
Z_TL = Z Position of Toolpoint

$$ {}^{GND}HTM_{T1} = \begin{bmatrix} 0.0 & -1.0 & 0.0 & 0.0 \\ 1.0 & 0.0 & 0.0 & 1.5 \\ 0.0 & 0.0 & 1.0 & 12.0 \\ 0.0 & 0.0 & 0.0 & 1.0 \end{bmatrix} $$

Fig. 5. HTMs and structural loops.

Table 1. *Constraints and violations for tool element*

Constraint	Satisfaction
X_Straightness < *Max_X_Straightness*	Not satisfied
Y_Straightness < *Max_Y_Straightness*	Not satisfied
Z_Straightness < *Max_Z_Straightness*	Not satisfied
Tool_Mass < Max_Tool_Mass	Satisfied
Tool_Mass > Min_Tool_Mass	Satisfied
Tool_Length < Max_Tool_Length	Satisfied
Tool_Length > Min_Tool_Length	Satisfied
Applied_X_Loading < Height*Width*Sigma_Comp	Satisfied
Applied_X_Load < Height*Width*Sigma_Yield	Satisfied

4.3. Constraint evaluation

A representation of some of the constraints associated with the total machine spindle, as well as the violations as reported by PMDA are shown in Table 1. After the values for the constraints were specified and the user-defined parameters set, the network was evaluated. For the sample machine spindle, several constraints were reported as violated (indicated by italic type). The violations for the tool object indicate that the straightness values are in excess of the acceptable values and the tool requires modification.

4.4. Error calculation

The process of determining the tool/workpiece interaction error is a two-step process. First, the tool/workpiece interaction point under ideal conditions is determined. Second, the errors are propagated through the structural loop. The result is compared to the ideal point to determine the net error.

For the machine spindle example, the tool/workpiece interaction point is determined by the contact point of the tool with the workpiece. The complete set of error motions included in the machine spindle example, as well as the characteristic values of these error motions, are shown in Table 2.

Since the error motions of the workpiece loop vary with respect to the rotation angle of the spindle, the errors and associated workpiece point deflection is calculated for a full revolution of the spindle in one-degree increments. The objects in the tool loop and the associated errors are assumed to be static with respect to time, and required only one calculation to determine the deflection of the tool point.

The axial and radial displacement of the spindle error motions are shown in Figure 6. The axial deflection of the ideal workpoint results from the axial runout and surface finish effects of the bearing and the thermal expansion of the shaft. Note that the thermal expansion and axial bearing runout result in a static translation of the workpoint, while the surface finish effects change with respect to rotation angle. Figure 6 also shows the results of the center-offset, runout, and eccentricity errors from the objects in the workpiece structural loop on the global Y location of the workpiece.

4.5. Summary

The PMDA system was able to automatically and correctly generate the constraints and the HTM models as well as analyze the proposed configuration for a specific set of input errors. To validate the results, the HTM model and constraints were generated "by hand" and compared to the PMDA model. The PMDA system reduced the modelling time from several hours to a few minutes. In addition, the use of the PMDA modelling system reduced the chance of errors in the modelling process.

Table 2. *Error motion values for the example machine spindle*

Object	Error motion	Value	Angle
Workpiece	none	N/A	N/A
Chuck	Jaw center-offset	0.035 mm	$\Pi/1.0$
	Mounting offset	0.002 mm	$\Pi/5.37$
Shaft	Eccentricity (X/Y)	0.0035/−0.0008 mm	0°/90°
	Center-offset	0.00713 mm	Π
	Thermal gradient	+50° from ambient	N/A
Rotary bearing	Inner ring runout	0.008 mm	$6.2\Pi/4.13$
	Outer ring runout	0.0038 mm	$9.21\Pi/3.3$
	Axial runout	$.00352^{*}\mathrm{RND}^{\#}$	N/A
	Surface finish (min/max)	0.00082/0.00022	N/A
Tool	X-straightness	0.000243 mm/mm	N/A
	Y-straightness	0.000671 mm/mm	N/A
	Z-straightness	0.000189 mm/mm	N/A

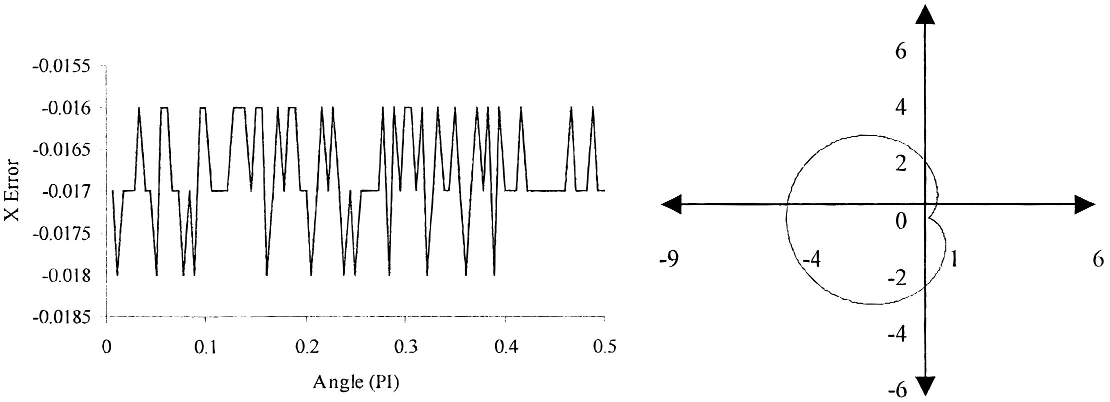


Fig. 6. Axial and radial displacement errors (mm and 1/100 mm).

5. CONCLUSION

This paper presents the PMDA tool, an object-oriented system used for the rapid evaluation of machine tool concepts. A kinematic model is created automatically when the user creates a design from a set of components and interfaces. The errors are evaluated using the set of HTMs generated by PMDA. The process of modelling both the constraint set and the error budget is accelerated and error-proofed through the use of standard representations of frequently used mechanical components and design interfaces.

The PMDA approach has several advantages. The first benefit comes from the ability to rapidly specify a design in a way that is natural to a designer. Standard components are selected and linked together. The modelling and constraints are generated and the quality of the design is immediately fed back to the user. The second benefit comes from the automatic and rapid generation of the kinematic models. Error motion magnitudes can be altered, sensitivity analysis, and what-if scenarios can be performed with minimal re-modelling time.

The first version of PMDA demonstrates that the rapid evaluation of precision machine tool concepts based on kinematic modelling, error motions, and design constraints, is feasible and useful in the precision machine tool design process. Future versions will include methods to optimize the design subject to the constraints, as well as, automatically select components from catalogs to optimize the design.

ACKNOWLEDGMENTS

The authors acknowledge the Naval Research Labs, Larry Shuette, Dan Gearing, and Dave Armoza specifically for the grant that enabled this work (grant No. N00014-95-G039) . In addition, thanks to Alex Slocum and Dan Frey.

REFERENCES

Bryan, J.B. (1984). The power of deterministic thinking in machine tool accuracy. *Proceeding of the First International Machine Tool Engineers Conference*.

Chakrabarti, A., Bligh, T.P., & Holden, T. (1992). “Towards a decision-support framework for the embodiment phase of mechanical design.” *Artificial Intell. Eng. 7*, 21–36.

Colton, J.S., & John, L. Dascanio II. (1991). An integrated, intelligent design environment. *Eng. Computers 7*, 11–22.

Foley, J.D., van Dam, A., Feiner, S.K., & Hughes, J.F. (1990). *Computer graphics—Principles and practice, 2nd ed.* Addison-Wesley Publishing Company, Reading, MA.

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

Johnson, A.L., & Thornton, A.C. (1991). Towards real CAD. *Design Studies 12(4)*, 232–236.

Kannapan, S.M., & Marshek, K.M. (1990). An algebraic and predicate logic approach to representation and reasoning in machine design. *J. Mechanical Machine Theory 25(3)*, 335–353.

Li, C.L., Tan, S.T., & Chan, K.W. (1996). A qualitative and heuristic approach to the conceptual design of mechanisms. *Engineering Applications in Artificial Intelligence 9(1)*, 17–31.

Reddy, S.Y., & Fertig K.W. (1996). Design sheet: A system for exploring design space. In *Artificial Intelligence in Design ’96*. (Gero, J.S. & Sudweeks, F., Eds.) pp. 347–366. Kluwer, Dordrecht, Netherlands.

Shigley, J.E., & Mitchell, L.D. (1983). *Mechanical engineering design*. McGraw-Hill Book Company, New York.

Slocum, A.H. (1992). *Precision machine design*. Prentice-Hall, Inc., Englewood Cliffs, NJ.

Spotts, M.F. (1978). *Design of machine elements*. Prentice-Hall Inc., Englewood Cliffs, NJ.

Thornton, A.C. (1993). *Constraint specification and satisfaction in embodiment design*. PhD Thesis. Dept. of Engineering, University of Cambridge.

Thornton, A.C. (1996). The use of constraint-based knowledge to improve the search for feasible designs. *Engig Appl. of Artif. Intell.*, *9(4)*, 393–402.

Thornton, A.C., & Johnson, A.L. (1996). CADET: A software support tool for constraint processes in embodiment design. *Res. Eng. Design 8*, 1–13.

Watton, J.D. (1989). *Automatic reformulation of mechanical design constraints to enhance qualitative and quantitative design decisions*. PhD Thesis, Carnegie Mellon University, Pittsburgh, PA.

Bradley Homann is a consultant working on virtual machining simulations for high-precision grinding machines. His professional interests include machine modelling and mechanical design. Mr. Homann received his M.S. from the Massachusetts Institute of Technology in June of 1997, and his B.S. in Aerospace Engineering from Embry-Riddle Aeronautical University, Prescott.

Anna Thornton received her Ph.D. from Cambridge University in 1993, and joined the Mechanical Engineering faculty as a half-time Lecturer for the period of January 1994 to June 1995. During this period she gained practical experience at Polaroid Corp. as a design engineer and with Concentra Corporation in design software development. In July 1995 she was appointed as an Assistant Professor. Her research includes work with the Fast and Flexible Manufacturing Project funded by ARPA, the Leaders for Manufacturing and Lean Aircraft Programs, the Center for Product Development and other industry research groups. Her research has focused on developing computational tools and formalized to assist in the design process. Specifically, she has focused on the area of variation risk management, constraint-based design and design automation methods.-