ANIMATED COMPUTER GRAPHICS MODELING OF ROLLING BEARING DYNAMICS

Dr Pradeep K. Gupta
Pradeep K Gupta Inc
117 southbury Road
Clifton Park, New York 12065

18 December 1991

Final Report for Period April 1991 to October 1991

AERO PROPULSION AND POWER DIRECTORATE
WRIGHT LABORATORY
AIR FORCE SYSTEMS COMMAND
WRIGHT-PATTERSON AIR FORCE BASE, OHIO 45433-6563

NOTICE

When Government drawings, specifications, or other data are used for any purpose other than in connection with a definitely Government-related procurement, the United States Government incurs no responsibility or any obligation whatsoever. The fact that the government may have formulated or in any way supplied the said drawings, specifications, or other data, is not to be regarded by implication, or otherwise in any manner construed, as licensing the holder, or any other person or corporation; or as conveying any rights or permission to manufacture, use, or sell any patented invention that may in any way be related thereto.

This report is releasable to the National Technical Information Service (NTIS). At NTIS, it will be available to the general public, including foreign nations.

This technical report has been reviewed and is approved for publication.

GARRY D. GIVAN, Proj Engr.
Lubrication Branch
Fuels and Lubrication Division
Aero Propulsion and Power Directorate

RONALD D. DAYTON, Chief
Lubrication Branch
Fuels and Lubrication Division
Aero Propulsion and Power Directorate

FOR THE COMMANDER

LEO S. HAROOTYAN, JR, Chief
Fuels and Lubrication Division
Aero Propulsion and Power Directorate

If your address has changed, if you wish to be removed from our mailing list, or if the addressee is no longer employed by your organization please notify WL/POSL, WPAFB, OH 45433-6563 to help us maintain a current mailing list.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

Form Approved OMB No. 0704-0188

1a. REPORT SECURITY CLASSIFICATION: UNCLASSIFIED	1b. RESTRICTIVE MARKINGS
2a. SECURITY CLASSIFICATION AUTHORITY	3. DISTRIBUTION/AVAILABILITY OF REPORT: Approved for public release; distribution unlimited
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE	
4. PERFORMING ORGANIZATION REPORT NUMBER(S): G-112-TR	5. MONITORING ORGANIZATION REPORT NUMBER(S): WL-TR-91-2129
6a. NAME OF PERFORMING ORGANIZATION: Pradeep K Gupta Inc; 6b. OFFICE SYMBOL (If applicable): PKG Inc	7a. NAME OF MONITORING ORGANIZATION: Aero Propulsion and Power Directorate WL/POSL
6c. ADDRESS (City, State, and ZIP Code): 117 Southbury Road, Clifton Park, New York 12065	7b. ADDRESS (City, State, and ZIP Code): Wright Laboratories, Wright-Patterson Air Force Base, Ohio 45433-6563
8a. NAME OF FUNDING/SPONSORING ORGANIZATION: Aero Propulsion and Power Directorate; 8b. OFFICE SYMBOL (If applicable): WL/POSL	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER: F33615-91-C-2132
8c. ADDRESS (City, State, and ZIP Code): Wright Laboratories, Wright-Patterson Air Force Base, Ohio 45433-6563	10. SOURCE OF FUNDING NUMBERS

PROGRAM ELEMENT NO.	PROJECT NO.	TASK NO.	WORK UNIT ACCESSION NO.
65502F	3005	02	00

11. TITLE (Include Security Classification) ANIMATED COMPUTER GRAPHICS MODELING OF ROLLING BEARING DYNAMICS

12. PERSONAL AUTHOR(S) Dr. Pradeep K. Gupta

13a. TYPE OF REPORT	13b. TIME COVERED	14. DATE OF REPORT (Year, Month, Day)	15. PAGE COUNT
Final	FROM APR91 TO OCT91	1991 December 18	28

16. SUPPLEMENTARY NOTATION

This is a Small Business Innovation Research (SBIR) Program, Phase 1 report.

17. COSATI CODES			18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
FIELD	GROUP	SUB-GROUP	Animation, Animated Graphics, Rolling Bearing, Bearing Dynamics

19. ABSTRACT (Continue on reverse if necessary and identify by block number)

A two-dimensional animated graphics model to simulate dynamic motions of the balls and cage, in a plane normal to the bearing axis, in an angular contact ball bearing is developed. The graphics modeling is based on the primitives available under the Programmers Hierarchical Interactive Graphics System (PHIGS), which is now an international standard and it is supported on a wide range of computer systems. Appropriate transformations to produce the animated motion of bearing elements are derived from the dynamic solutions provided by the bearing dynamics computer code ADORE. Necessary modifications to ADORE to store all the required solutions in an ASCII data file are performed. This newly developed data file provides the interface between ADORE and the graphics model. Since PHIGS graphics standard is very widely supported, the graphics model developed herein can be used on a wide range of computer systems. For the present investigation an IBM-RISC/6000 work station is used. Both the bearing code, ADORE, and the graphics code are executed on this computer system to produce the

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT: ☐ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT ☐ DTIC USERS	21. ABSTRACT SECURITY CLASSIFICATION
22a. NAME OF RESPONSIBLE INDIVIDUAL: Garry Given	22b. TELEPHONE (Include Area Code): 513-255-1286; 22c. OFFICE SYMBOL: WL/POSL

DD Form 1473, JUN 86 — Previous editions are obsolete.

SECURITY CLASSIFICATION OF THIS PAGE

UNCLASSIFIED

animated motion of bearing elements as a function of the applied operating conditions.

The two-dimensional displays generated in the present Phase I effort demonstrate the technical feasibility of the overall modeling approach and they provide a strong foundation for a more extensive development in Phase II of this project. In addition, the 2-D graphics model,developed in this project, is completely interfaced with the bearing dynamics computer code, ADORE. It can, therefore, be immediately used as design tool.

FOREWORD

This research was sponsored by the United States Air Force under the Defense Small Business Innovation Research (SBIR) Program, Air Force Contract F33615- 91-C-2132. The Air Force Project Manager was Mr. Garry Givan (WL/POSL). The present work constitutes Phase I effort of the overall program.

Table of Contents

1. INTRODUCTION

High performance gas turbine engine bearings are often subjected to adverse operating environments, where the mechanical, chemical and thermal interactions between the fundamental elements of the bearing become quite complex. The overall behavior of the bearings as a function of the operating environment very often determines the performance and operating life of the entire turbine engine system. Thus the main shaft bearings are among the critical components of gas turbine engines. Among the various types of bearings, rolling bearings, due to their high load support, stiffness and speed capabilities, are the most common type of bearings used in turbine engines. In addition, rolling bearings are employed in a wide range of other applications covering a rather large spectrum of operating loads and speeds. The applications include both DOD and commercial systems. Precision gyroscopes and momentum wheels used in communication satellites, helicopter transmissions, auxiliary power units, a wide range of automotive applications, cryogenic turbopumps and related space systems, and more recently the rolling bearings used in computer disk drives, are some examples. Due to such a wide application domain, modeling the performance of rolling bearings has been of significantly increasing interest over the past many years. Sophisticated mathematical and numerical procedures have been developed to model the subtle kinematic and dynamic phenomena in rolling bearings. The procedures have been implemented in advanced computer codes which integrate the classical differential equations of motion of the bearing elements to model the overall dynamic performance of the bearings under complex operating environment. However, as the complexity of the model increases, the results or performance predictions of a model also become quite complex and the need for computer tools to interpret the model predictions in very practical terms becomes vital to effective design and performance simulation. The commonly used computer print outputs or conventional two or three dimensional graphical representation of certain performance parameters become inadequate to fully comprehend the subtle interactions which are fundamental to the overall performance of the bearings. Such problems associated with the practical interpretation of the model predictions are more obvious when the advanced computer codes are used by bearing engineers for practical designs. With the advent of modern computer graphics technology, animated display of the dynamic motions, as predicted by the computer codes, provides the necessary bridge to effectively transfer the advanced technology to real practical system. Such a pictorial representation provides a very lively perspective of the overall bearing behavior and it requires minimum imagination from a designer to implement the most advanced technology to practical systems. The development of such animated graphics tools for rolling bearings is, therefore, the primary objective of this project. In the present Phase I effort the motions of balls and cage in an angular contact ball bearing are simulated in two dimensions to demonstrate the technical feasibility of the overall modeling approach, and provide a sound foundation for a more extensive development in Phase II.

Over the past decade a significant advancement has been made in modeling the complex dynamic behavior of rolling bearing elements. Computer codes, which integrate the differential equations of motion of the bearing elements and thereby provide a real-time simulation of bearing performance are now fairly widely used in the industry. However, as the motions become complex, it becomes increasingly difficult to relate the results to critical design parameters and the need for animated pictorial display becomes obvious. As an example, figure 1-1 shows the whirl orbits of the cage mass center, as produced by the bearing dynamics computer code ADORE [1], in a solid lubricated roller bearing designed for a small high-speed gas turbine engine. In addition to such orbit plots, there are other results which indicate out-of-plane coning of the cage, the interactions in the cage pockets and guide lands, and the extent of overall mechanical interaction when an instability is triggered. An effective compilation of all the results and subsequent interpretation of the results in terms of design significance is indeed a laborious task. Such a task can be more effectively performed by a animated graphics model which takes the fundamental components of motion as input to simply display the actual motion in an animated fashion on the computer screen. While the computer codes, such as ADORE, provide the basic components of motion of the bearing elements, the rapidly advancing computer hardware, and the emerging graphics software standards provide the tools required for the development of such animated graphics software. Personal computers and dedicated work stations now offer both the compute speed and graphics capabilities required to draw the images at a rate fast enough for a lively visualization of the simulated motion. The RISC (Reduced Instruction Set Computer) based processors, and the recent Intel 80486 type computer chips offer compute speed which is close to that available on a wide range of mainframe computers. In addition, graphics hardware options such as double buffering and Z-buffers provide high-speed image processing required for animated graphics. In the software area, the graphics primitive tools, such as GKS (Graphics Kernel System), and PHIGS (Programmers Hierarchical Interactive Graphics System) are fairly standard on a wide range of computer systems. Thus the software portability from one system to the other is greatly eased.

In the present effort, the available the bearing dynamics computer code ADORE is used to generate the time-varying motions of the balls and cage in an angular contact ball bearing typical of gas turbine engine application; using the graphics primitives available under the PHIGS standard, computer subprograms to generate images of the bearing elements and prepare graphics "structures" are written; appropriate transformation algorithms which take the motion generated by ADORE and move the various graphics structures are developed and coded into subprograms; finally, the transformation codes and the element image codes are combined to generate the animated graphics model for the bearing. An IBM-RISC/6000 computer work station is used in the present project. However, all the computer codes developed are actually machine independent. For simplicity, and for the purpose of proving the technical feasibility of the overall approach, all transformations and imaging are restricted to a two-dimensional plane normal to the bearing axis. Thus the view angle for the animated display is along the bearing axis. A number of parametric runs which show the animated display as a function of the applied operating conditions are generated to prove technical feasibility of the overall approach. These results along with the graphics codes developed in the present Phase I effort provide a sound baseline for a more rigorous development of three dimensional imaging and transformations with arbitrary view angles in the second phase of the project. The development of graphics modules for other types of bearing elements, such as cylindrical and tapered rollers, shall also be a subject of the Phase II effort.

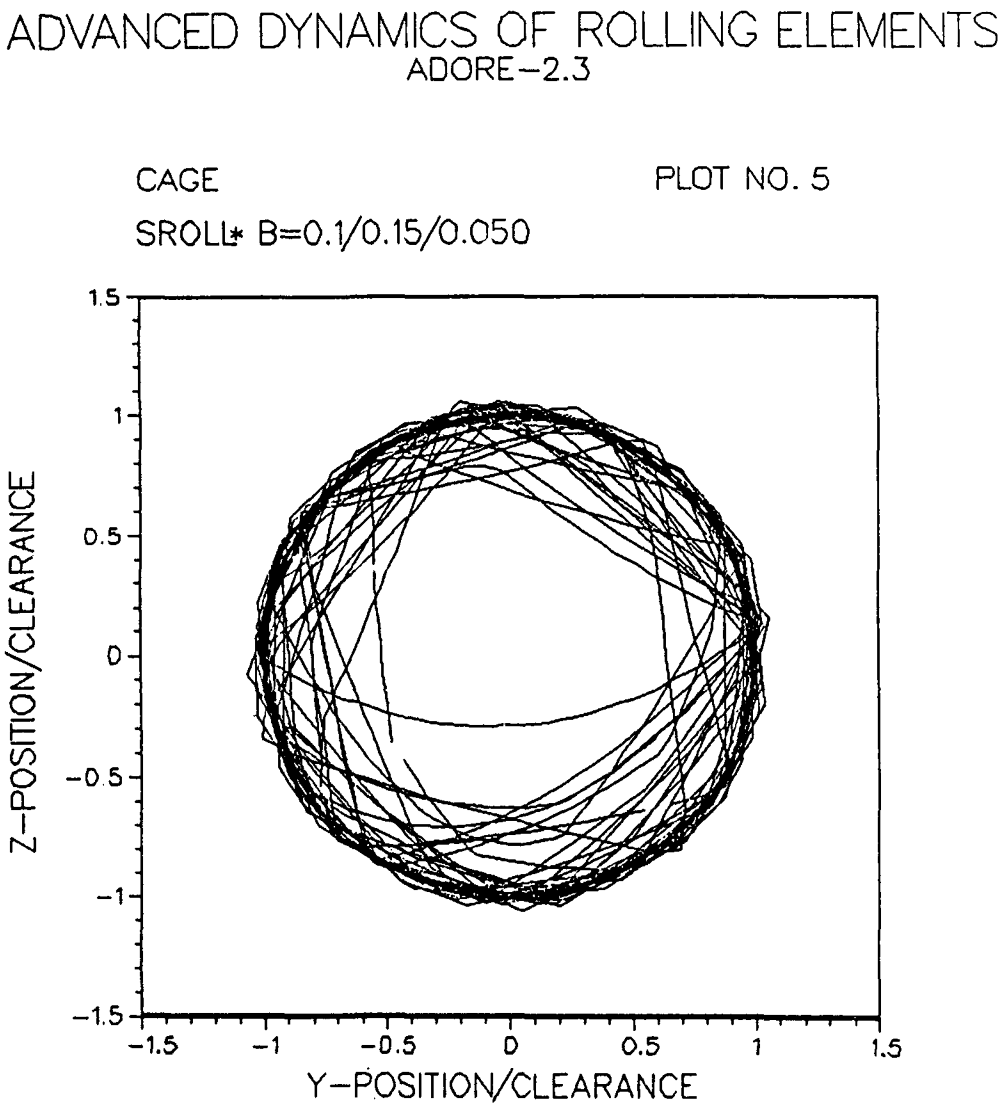


Figure 1-1 Typical onset of whirl instability in a high-speed cylindrical roller bearing.

2. TECHNICAL APPROACH

The animated graphics model is based on the primitives available under the PHIGS (Programmers Hierarchical Interactive Graphics System), which is an international standard for graphics development. The model is really a stand alone graphics facility, the input to which is supplied by bearing dynamics codes. Figure 2-1 shows a simplified outline of the proposed approach. An available bearing dynamics computer code ADORE is used to integrate the equations of motion of the bearing elements. The various components of motion are compiled in a data base. This data base provides an interface between graphics and bearing dynamics codes. In fact, this data is the primary input to the graphics model. The output from the graphics model is the animated display of pertinent elements. For example, in a ball bearing, the display includes motions of all the balls, cage, and the two races.

Any graphics software development is generally limited by the capabilities of available graphics hardware. Hardware variables such as computer display resolution, processor speed, available local storage and access times, are some of the variables which control the graphics development process. With the advent of the modern 80486 type personal computers and the RISC based work stations most of these capabilities are now easily available. The very recent IBM-RISC/6000 work stations offer both the compute speed and graphics capabilities. In particular the Model 550 offers compute speeds significantly higher than a number of main frame systems. Also, with appropriate options the graphics processing is also very attractive. In view of such capabilities, the IBM-RISC/6000 Model 550 with a power graphics accelerator option is selected as a baseline hardware for the present project. As the name implies, this work station employs the industry standard RISC architecture and it operates under the Unix operating system. The graphics accelerator offers an efficient graphics engine which employs the state-of-the-art double buffering and Z-buffers. Video interface to transfer the animated motion to conventional video tapes for easy viewing on a standard television set is also expected to be available from IBM in the very near future. Presently the work station is networked to the Tektronix XD8830 work station, which supports a direct video interface card.

Graphics animation essentially requires continued refreshing of an image on the computer display with varying position and orientation. If the moving object can be refreshed at a rate of about 30 frames a second, the display appears like that of a motion picture. Both the complexity of the image and the compute speed of the available hardware determine the overall refresh rate. The available bench marks on the IBM work station are very attractive. In addition, the expected future enhancements will further broaden the complexity limits of graphics imaging and animation. In order to maintain such an upward compatibility with computer hardware, and also for easy portability of the code to different computer systems, it is essential that fairly standard tools are used for the development of the graphics animation software. Use of the well established PHIGS package fulfills such a requirement. In order to enhance both the compute and display efficiency, PHIGS primitives permit the basic graphic images to be stored as "structures", which can be easily edited to apply time varying transformations. The approach is somewhat similar to Tektronix style graphics where images are stored in "retained graphics segments", however, the PHIGS structures are recognized as an accepted international standard. With such fundamental architectural details the overall graphics approach becomes quite simple; the primary objects are stored in editable structures; the

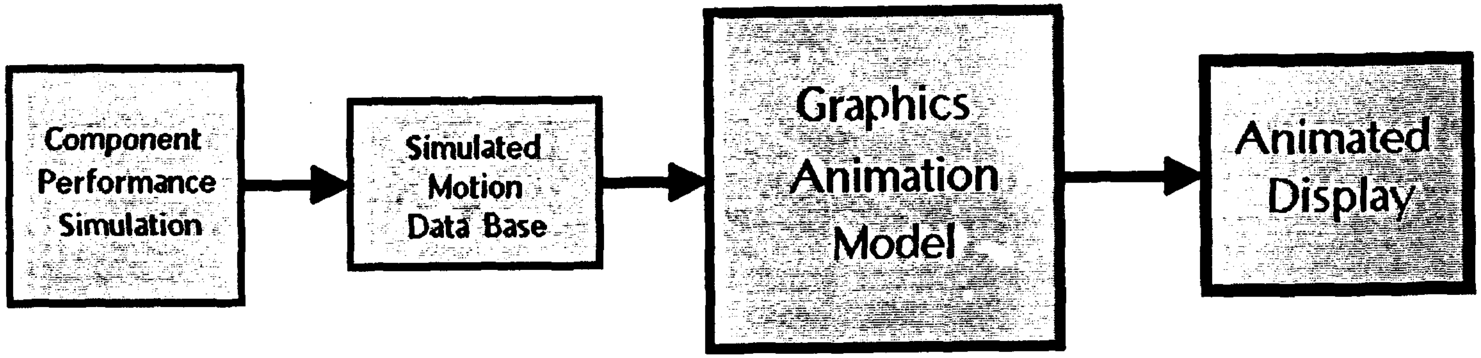


Figure 2-1 Overview of the graphics animation modeling approach.

transformations are determined from the real-time dynamic motion, such as that computed by ADORE; the graphic structures are then edited as a function of time to apply the changing transformations; and, finally the edited images are displayed on the computer screen.

Based on the above overview of the graphics modeling process, a more detailed outline of the overall approach is schematically shown in figure 2-2. The bearing dynamics computer code ADORE is executed to generate the simulated dynamic motion of bearing elements. The output is compiled in the form of a data base which contains the fundamental components of motion of all bearing elements. The PHIGS primitives are used to develop the graphics codes which generate the shape of bearing elements from the prescribed geometry. The output from these codes, e.g., shape of the bearing elements, can be stored in structures in the computer memory. The data base, obtained by executing ADORE, is now used to generate the transformation coordinates as a function of time. These transformations are then applied on the appropriate graphic structures by using the available editing functions. Finally, the modified images are displayed on the computer monitor. The process is repeated for each time step and the image is continuously refreshed. Thus an animated view is seen on the monitor.

With the primary objective of proving the technical feasibility of this overall approach, the present Phase I effort is restricted to modeling the ball and cage motion in two dimensions, for a turbine engine ball bearing. In other words, the view angle for the animated motion is be along the bearing axis. Whirl motion of the cage is considered in a plane normal to the bearing axis. Both the rotation about the bearing axis and whirl of the cage mass center is displayed to produced an animated picture. Before presenting the results, some details of the graphics model are presented in the next section of this report.

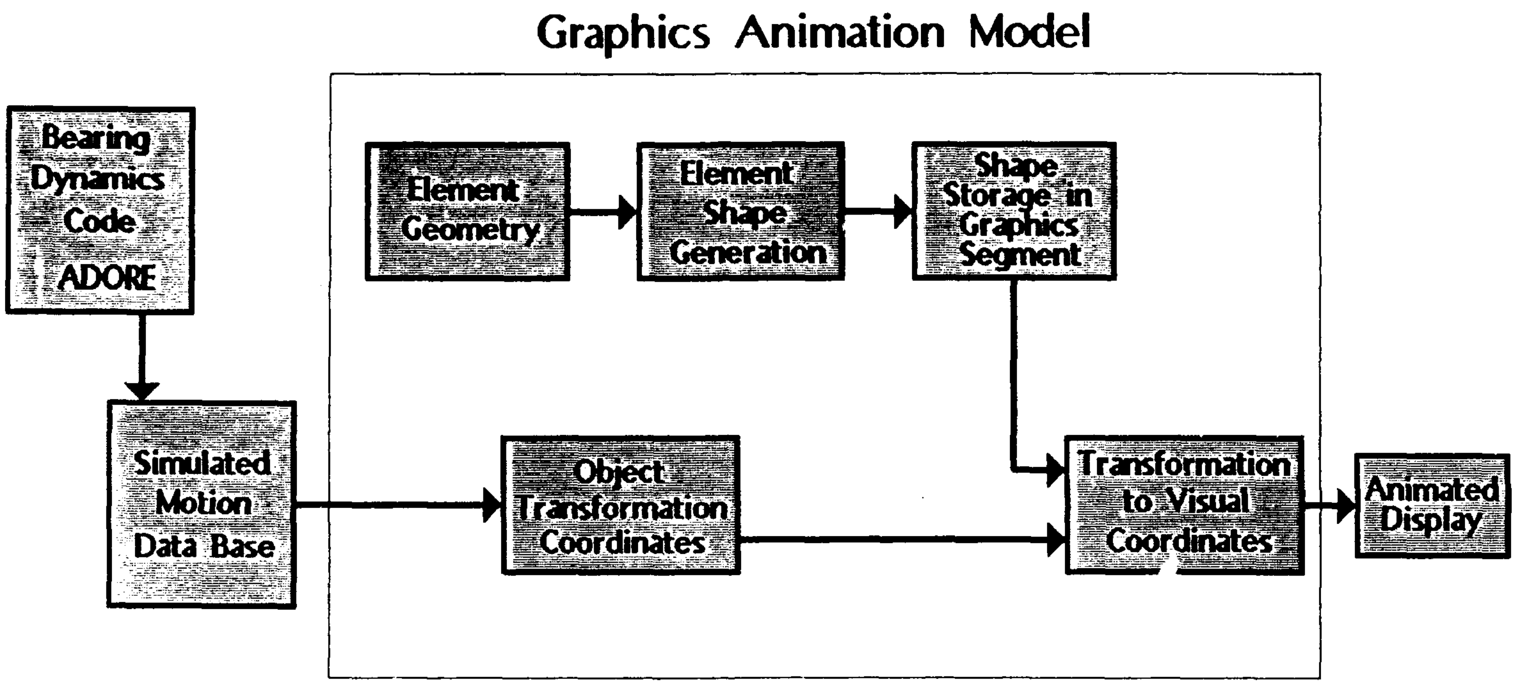


Figure 2-2 Schematic details of the graphics animation model.

3. GRAPHICS MODEL DEVELOPMENT

As discussed in the preceding section the graphics model is based on the PHIGS graphic primitives, which are recognized as a international standard for graphics development. Any graphic image is developed by a logical collection of graphic primitives or "elements". This collection of fundamental elements is named as a "structure". In other words a structure consists of a collection of a number of elements. On a more fundamental level each graphic element may have a number of attributes associated with it. Once a structure is constructed, each of its elements and attributes are completely editable. Such a capability constitutes the fundamental strength of the PHIGS standard. As a example a number of line and/or polygonal elements may be logical assembled to create a graphic object; this collection may be stored as a structure in terms of PHIGS terminology. The elements may have fundamental attributes such as line thickness, color, polygon fill pattern etc. In addition certain transformation elements may be included to define position and orientation of the object. Since the structure is completely editable, the color, position and orientation can be easily changed at any time. The process of animation, therefore, consists of repeated editing and display of the graphic structures. Based on such an understanding, the components of a graphics model for a bearing may be discussed in terms of the following.

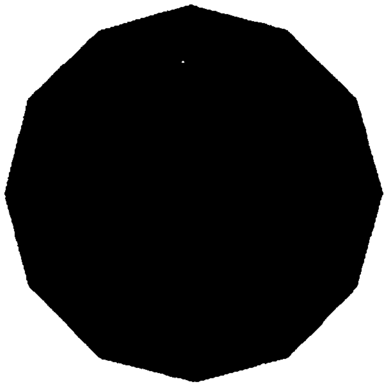
3.1 Ball Structure

A simple polygon-fill element is used to create the ball structure. Figure 3- 1a shows a typical polygon with 12 sides. As the number of sides increases the polygon will appear like a circle. In figure 3-1b, circular area is really a 36 sided polygon. Note that for ease of duplication, the areas shown in figure 3-1 are shaded black; they are really colored on the actual computer display.

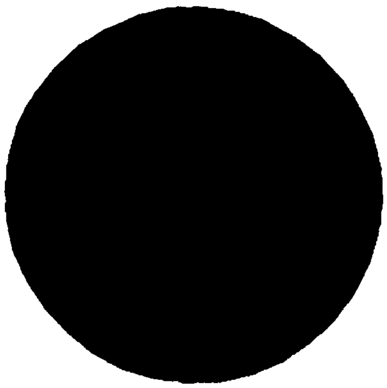
3.2 Cage Structure

The polygon-fill element is also used for the cage segment. The details of a single pocket are shown in figure 3-2a; again the actual color picture is shown in terms of several shades of gray. Within the pocket the two thin rectangles on the pocket walls are introduced with a changeable color. Normally the color of these rectangles is set to be the same as the cage color, and therefore the rectangles are not distinctly seen. Whenever the ball contacts the cage, the color of this rectangle on the appropriate wall is changed to a brighter and more pronounced color, such as bright red. Thus the indication that the ball has contacted the cage is clearly visible. The main area of the pocket seen with a lighter shade in figure 3-2a, is actually light blue. This color is normally set to the background color of the monitor for a better and more clear display of the bearing.

The actual size of the pocket section shown in figure 3-2a depends on the actual cage geometry. For a prescribed number of ball this pocket sector is repeated to form a complete cage. Figure 3-2b shows a cage with eighteen pockets. The thick coordinate frame is fixed at the cage center and this frame rotates with the cage. A thinner arrow, shown superposed on

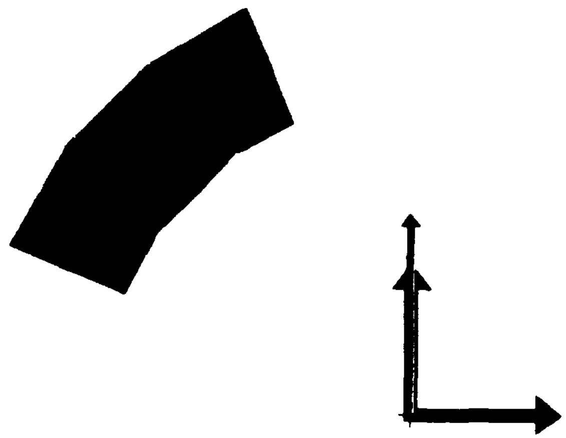


(a) Simulation by a 12-sided polygon.

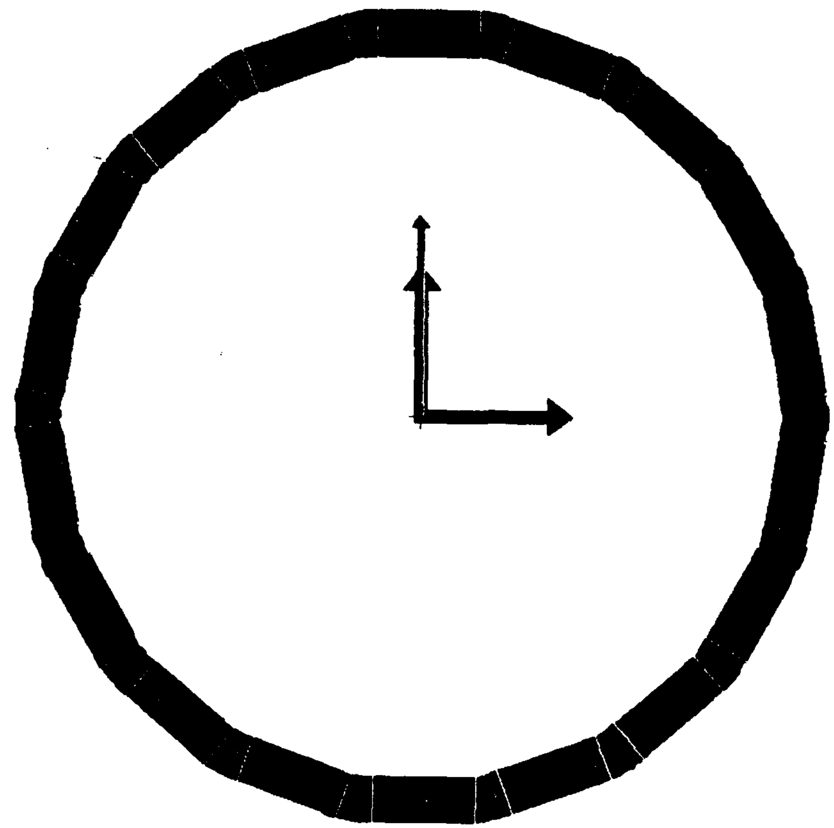


(b) Simulation by a 36-sided polygon.

Figure 3-1 Graphic structure elements of a ball.



(a) Details of a cage pocket sector.



(b) Complete cage structure.

Figure 3-2 Development of a graphic structure for the cage.

this coordinate frame, points to the position of contact between the cage and the race guiding land. This vector is also shown in the cage fixed reference frame. A constant angle between this vector and the cage fixed coordinate frame represents contact at a fixed point on the cage. Such a condition may result in excessive wear and possibly a cage failure. Thus potential cage problems can be easily identified while viewing the overall motion.

3.3 Race Structure

Typical structure of the bearing raceway is shown in figure 3-3. This is again, basically a solid fill polygon with a given color. A small white rectangle is drawn within the race area, so that race rotation can be more clearly seen in the animated display. Also, similar to the cage a race fixed coordinate frame is drawn at the race center. This coordinate frame will move with the raceway.

3.4 Composite Bearing Views

A view of the assembled bearing can be generated by executing the fundamental graphic structures defined above. The result is shown in figure 3-4. In addition to the ball, race and cage structures, a cage/race force (CRF) scale is overlaid in this view. Each time the cage contacts the guiding race the resulting contact force is displayed on this force scale. In order to show the mass center whirl orbit of the cage the cage displacement relative to the bearing center is plotted in this view with an enlarged scale. Typical orbits are shown by the somewhat circular lines in figure 3-4. When the bearing is subjected to a rotating radial load, race orbits are also plotted in this view. These orbits, as shown in figure 3-4, are purely circular. Note the two coordinate sets drawn in the same color as the races; the larger axes correspond to the stationary outer race while the smaller axes are fixed in the moving inner race.

In order to examine more detailed interactions of any ball in the cage pocket, another sectional view is created, as shown in figure 3-5. Here the motion of the ball is displayed relative to the cage pocket center. Thus the cage section is fixed while the ball moves in the pocket. Again a ball/cage force (BPF) scale is overlaid to show the magnitude of the pocket force whenever the ball contacts the cage. The position of contact is indicated by the arrow shown in the plan view of the pocket.

3.5 Transformations

All of the above graphic structures contain appropriate transformation elements which are edited to change the position and orientation of the computer display. Normally the transformation is applied by multiplying the appropriate vectors by a transformation matrix, the components of which are determined from the fundamental coordinates. Such transformations are also used in ADORE to locate the various bearing elements relative to each other. In fact, the models used in PHIGS are identical to the Euler angle type approach used in ADORE. This makes the interface between ADORE and the graphics model a very easy task.

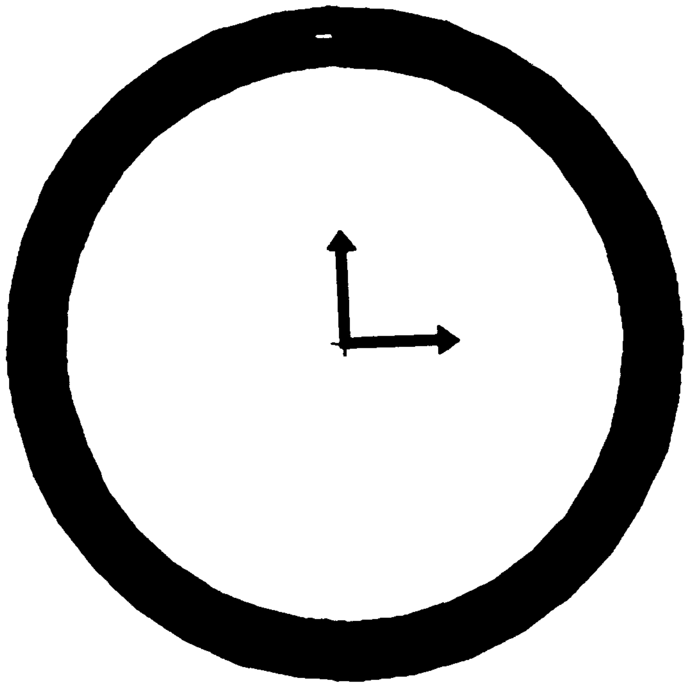


Figure 3-4 Race structure composed of polygon elements.

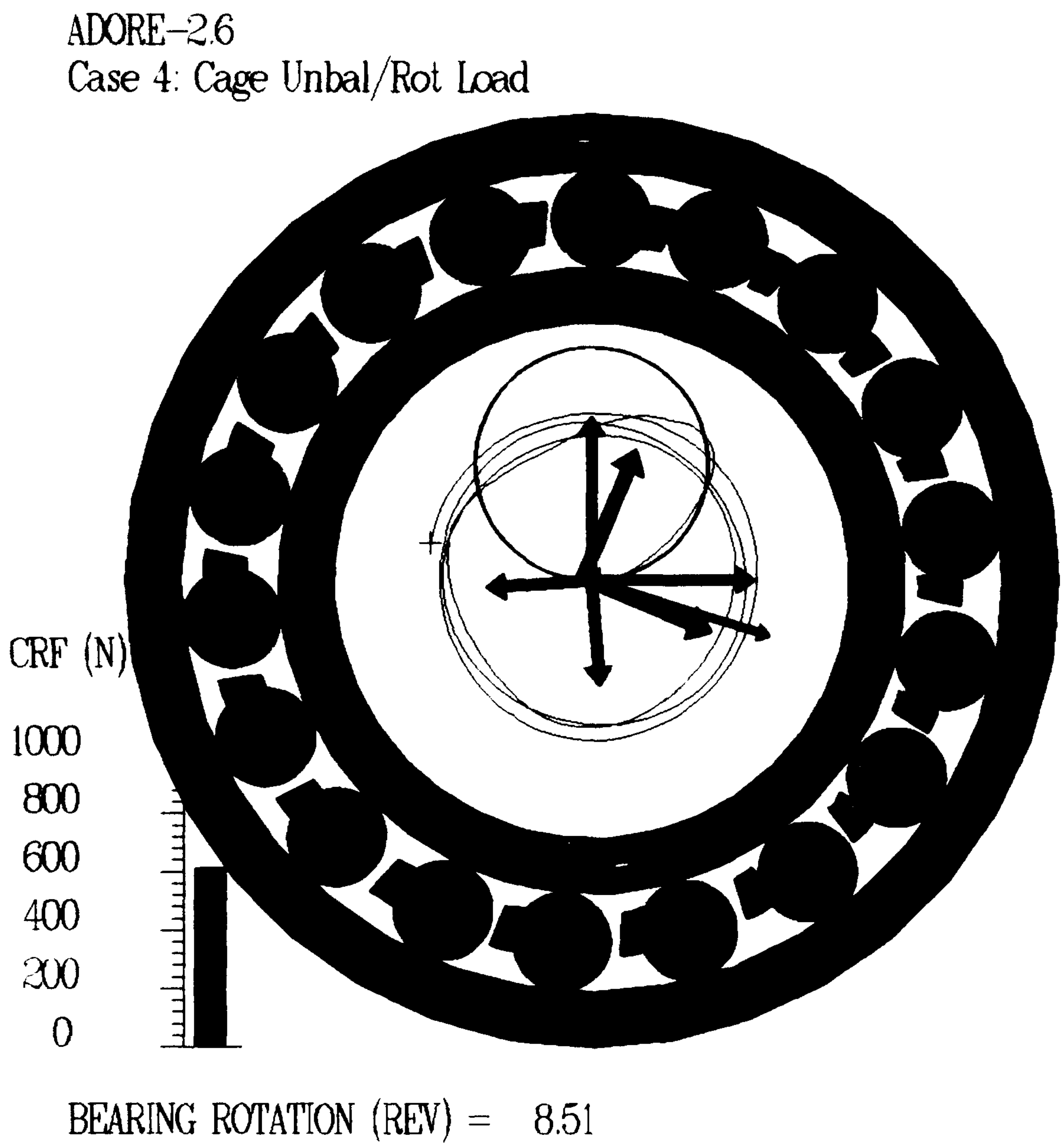


Figure 3-4 Typical view of the complete bearing.

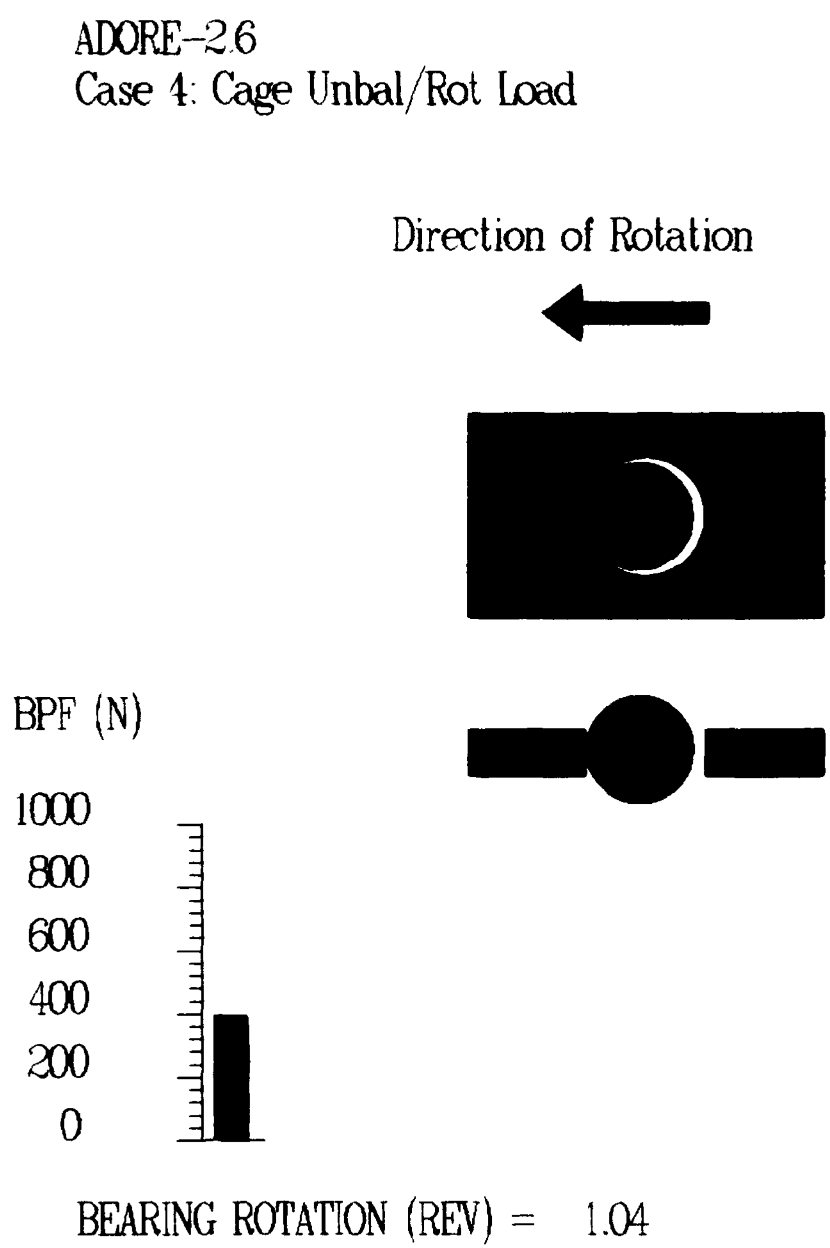


Figure 3-5 Typical view of ball pocket interaction.

There are really two general types of transformation required: translation and rotation. The PHIGS implementation of these transformations is discussed in extensive detail in several graphics texts; the work by Foley et al [2] is one example. Some mathematical formulations, as used in the present model, are briefly discussed below.

3.5.1 Translation

Translation of points from one coordinate set to the other basically consists of an additive vector which defines the separation between the two coordinate sets. In a two dimensional representation, as shown schematically in figure 3-6, a point $=P$ in the (x, y) plane $\mathbf{P}(x, y)$, can be transformed to the (x', y') system as

$$\mathbf{P}'(x', y') = \mathbf{P}(x, y) + \mathbf{D} \tag{3-1}$$

where the components of the different vectors are:

$$\mathbf{P} = \begin{Bmatrix} x \\ y \end{Bmatrix}, \quad \mathbf{P}' = \begin{Bmatrix} x' \\ y' \end{Bmatrix}, \quad \mathbf{D} = \begin{Bmatrix} d_x \\ d_y \end{Bmatrix}. \tag{3-2}$$

This additive operation is unfortunately different from the multiplicative treatment of rotational transformations, as discussed later. In order to easily combine the translation and rotation effects it is essential that all transformations are treated in a consistent way. Thus "homogeneous coordinates" are introduced. A third coordinate is added to the pair of numbers (x, y), and each point is represented by a triple (x, y, W). At the same time it is stated that the two sets of homogeneous coordinates (x, y, W) and (x', y', W'), represent the same point if and only if one is multiple of the other. Also, one of the homogeneous coordinates must be non-zero. When the W coordinate is nonzero, we can divide throughout by it and produce the commonly known Cartesian coordinates. The points with $W=0$ are called points at infinity; these are not of much interest here.

Typically the three coordinate sets represent a three-dimensional space, but here the triple set is used to represent points in a two-dimensional plane. In order to understand this connection consider all triples, (tx, ty, tW), $t \neq 0$, representing the same point. This results in a line in the 3-D space. If we homogenize the point by dividing by W, we get a point $(x, y, 1)$, or a 2-D point (x, y) in the $W=1$ plane.

In terms of the above definition each 2-D point is now a three-element row vector. The transformation must now be a (3x3) matrix. Such a representation of equation (3-1) can be shown to be

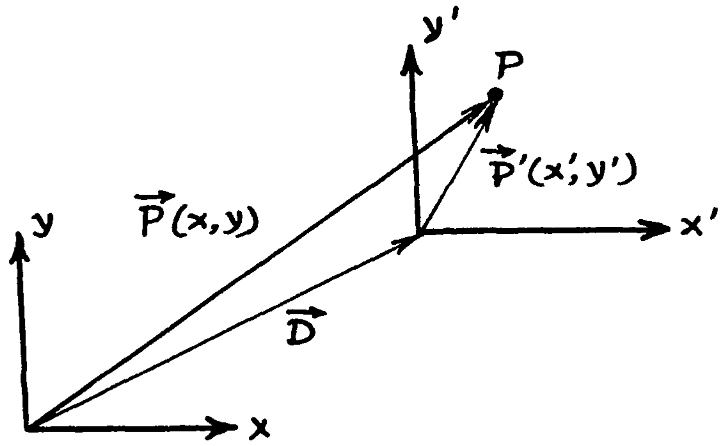


Figure 3-6 Schematic description of coordinates for translational transformation.

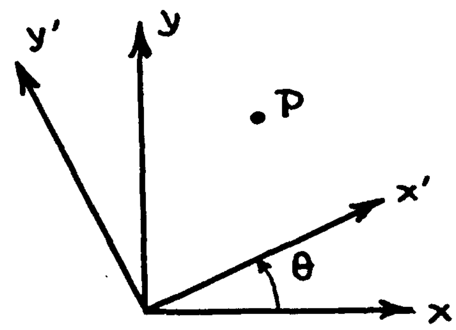


Figure 3-7 Schematic description of coordinates for rotational transformation.

$$\begin{Bmatrix} x' \\ y' \\ 1 \end{Bmatrix} = \begin{bmatrix} 1 & 0 & d_x \\ 0 & 1 & d_y \\ 0 & 0 & 1 \end{bmatrix} \begin{Bmatrix} x \\ y \\ 1 \end{Bmatrix} \qquad (3\text{-}3)$$

It can be easily shown [2] that in terms of the above definition the additive translations can be accomplished by cumulative multiplications of the transformation matrices. Thus the application is identical to the rotational transformation, discussed below. In fact, the translation and rotation matrices can be multiplied together to generate a composite matrix. However, the order in which the multiplication is performed must be carefully observed, since

$$[A]\,[B] \neq [B]\,[A] \qquad (3\text{-}4)$$

3.5.2 Rotation

When the coordinate frame is rotated by an angle θ about an axis normal to the (x, y) plane, as shown in figure 3-7, the resulting transformation can be shown to be [1,2]:

$$\begin{Bmatrix} x' \\ y' \end{Bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{Bmatrix} x \\ y \end{Bmatrix} \qquad (3\text{-}5)$$

In terms of the homogeneous coordinates, the above can be represented by an equivalent (3x3) matrix equation:

$$\begin{Bmatrix} x' \\ y' \\ 1 \end{Bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{Bmatrix} x \\ y \\ 1 \end{Bmatrix} \qquad (3\text{-}5)$$

3.6 Input Data Base

All inputs to the graphics model are supplied by the bearing dynamics computer code ADORE. Appropriate modifications to this code are made to generate a data base to contain the following information:

3.6.1 Bearing Geometry Data

The first part of the data set contains the required geometrical data to produce a scaled drawing of the bearing. In addition, other text information, such as program version, bearing specification, etc, are documented in this file, so that this information is contained in the titles contained in the bearing animation screen. For the 2-D animation considered in the present investigation, the following information is documented in this first part of the data set:

1. ADORE version
2. Bearing specification code
3. Type of units
4. Bearing type (presently, ball bearings only)
5. Number of rolling elements or balls
6. Number of variables in time-varying solutions
7. Ball diameter
8. Pitch diameter
9. Outer and inner diameters of the two races
10. Outer and inner diameters of the cage
11. Cage pocket and guide land clearances

3.6.2 Time Varying Solutions

Following the above nominal data, the dataset contains the solutions generated by ADORE at each time step. The fundamental coordinates of the bearing elements constitute the main part these solutions. Since ADORE permits all six degrees of freedom for each bearing element, there are six fundamental coordinates which describe the bearing element motion. These solutions are contained in a vector with six components, $\{x, y, z, \eta, \beta, \xi\}^T$, where (x, y, z) denote the mass center position, and (η, β, ξ) are the three transformation angles which define the angular orientation of the bearing element. Since the present program plots the motion only in two dimensions not all of these coordinate variables are presently used. Nevertheless they are available in the dataset for the 3-D development in Phase II. Presently the view angle for animation is restricted along the bearing axis (which is the x coordinate axis in ADORE), and therefore, only three coordinate variables (y, z, η) are used.

The data record produced at each time step contains the following specific information:

1. Time step number
2. Value of current time and race rotation
3. Fundamental coordinates for each rolling element, cage and races
4. Cage pockets forces and contact angles in each pocket
5. Ball position relative to pocket center in each pocket
6. Cage/Race force and contact angle

Once ADORE execution is completed, this database is attached as an input to the graphics code. Since ADORE may be executed with any of the available options, all capabilities of ADORE are seen in the graphics modeling; presently, of course, only in two dimensions and only for ball bearings.

4. RESULTS

Technical feasibility of the overall approach is demonstrated by a number of animations produced from ADORE runs. Since animation is the primary subject, discussion of results in terms of still pictures, which can be included in this report, becomes difficult. Therefore, a video tape, which is produced by direct video recording of the computer display screen via a video interface card, is included with this report. It may be essential to view this tape to appreciate the results of this investigation.

Although due to large differences in screen resolution between a graphics monitor and a normal television, the graphics quality of the presentations on the video tape is greatly restricted. In any case the overall value of the animations can be appreciated and, certainly the technical feasibility of the modeling approach can be demonstrated by the solutions recorded on this tape.

For all the computer runs a 100mm turbine engine ball bearing, which has been used in a number of previous investigations [3-5], is used in the present investigation. Details of the bearing geometry are identical to these earlier works, except that the cage clearances are greatly enlarged to demonstrate large relative motions between the ball and cage in the animated displays.

The bearing is assumed to operate at 20,000 rpm with the MIL-L-7808 type lubricant. A set of four solutions are contained on the video tape to demonstrate the animated outputs of the graphics code. The variations in operating conditions used, and some of the key observations in the animations are discussed below.

4.1 Case 1: Thrust Loaded Bearing

This is the nominal case a thrust load of 4,500 N is applied on the bearing. ADORE is executed over about 40 shaft revolutions. This took about 15 minutes of computer time on the IBM-RISC System 6000, Model 550. The data set generated for the graphics code is then interfaced with the animation model and the result is seen on the video tape under the label Case 1. Note that although the bearing is rotating at a constant speed of 20,000 rpm, the animation shows some variation in speed. This is really due to the variable time step size in ADORE simulations. Whenever the ball/cage collisions take place the time step is reduced to maintain numerical accuracy in the computed solutions; thus the step size continuously varies during the simulation. This results in the apparent variations seen in the animation.

Initial conditions for the dynamic simulation are such that the cage is displaced slightly in the radial direction and the whirl velocity is set equal to its angular velocity. As the bearing rotates the cage moves radially simply due to the gravity effects and contacts in the ball pockets are established. These are seen by the red flashing on the cage pocket walls. The magnitude of radial motion is, of course, of the order of cage/race guide clearance, which is quite a small number relative to the bearing dimensions. It is therefore difficult to see such a displacement on the animation screen where the full bearing is displayed. In order to solve this problem, the mass center motion of the cage is plotted with a somewhat enlarged scale; this results in the more clearly drawn whirl orbits shown in the green color (same as that of the cage) in the animation. As the radial motion of the cage continues, it eventually contacts the guiding inner race. The dotted circle in the animation represents the race guide land. As the inner surface of the cage touches this circle a contact is established. The actual position of the contact is indicated by the red arrow, which shows the possible cage/race load line in a cage fixed coordinate frame, which in turn is shown in green color (same color as the cage). Magnitude of the cage/race contact force is shown on the force scale, labeled as CRF (Cage Race Force). As the bearing continues to rotate, the initial transients tend to die out and the motion reaches a steady-state behavior, where the cage whirl orbit becomes somewhat circular and the cage/race force assumes a fairly steady value. Note that the cage/race contact angle, indicated by orientation of the red arrow with respect to the green cage fixed coordinate frame, continues to change. This indicates cage/race contacts at no preferred location on the cage as normally expected. Although the actual ADORE run was for over 40 revolutions, the animation on the tape is recorded for only about 10 revolutions, during which all salient features of this run are clearly seen and the bearing reaches a steady-state behavior.

Details of cage pocket interactions can be seen in the view of any selected pocket. The next animation on the tape shows the ball/cage interaction in pocket #1 under this nominal thrust loaded condition. Here the cage sector is fixed and the ball motion is shown relative to the cage pocket center. Position of possible ball contact is again indicated by the red arrow. If there is contact then the contact force is plotted on the force scale labeled as BPF (Ball Pocket Force). The ball/cage contact angle continues to flip between zero and 180 degrees. This simply means that sometime the cage drives the balls, while other times it is driven by the balls. Normally the cage contacts are for very short duration, as seen by the flashing red color on the force scale. If the contact is maintained over longer duration of time and a definite force is continuously displayed, excessive cage wear may be expected, which may eventually lead to cage failure.

4.2 Case 2: Combined Thrust and Radial Load

In the second case a radial load of 4,500 N is combined with the thrust load of 4,500 N used in Case 1. The resulting animation is labeled as Case 2 on the video tape. The cage motion is quite similar to that seen earlier in Case 1, except that the ball excursion in the cage pockets is greatly increased due to the radial load. This is seen in terms of more frequent ball/cage contacts. The pocket forces are also somewhat higher and so are the cage/race forces. The cage/race contact angle also varies continuously, indicating no preferred point of contact on the cage.

Except for the somewhat higher forces and more frequent cage/race contacts, the ball pocket simulations are similar to those seen in Case 1. These animations are, therefore, not included on the video tape.

4.3 Case 3: Cage Unbalance with Combined Thrust and Radial Load

In order to see some of the interesting dynamic effects reported earlier [4] with unbalanced cages, a 10 gm-cm radial cage unbalance is introduced in this case. The most notable effect seen in the animation is the fact that the cage/race contact angle, after the initial transients, does not vary over a large range. In fact the contact is restricted to a small arc on the cage. This is because the cage whirl velocity is exactly equal to its angular velocity, as reported earlier [4]. Actually, this animation is a graphical representation of experimental validation of ADORE predictions, since the experimental data also shows rather well defined points of cage/race contacts with unbalanced cages [4]. Such experimental validations greatly contribute to the design strengths of ADORE and the graphic animations developed in this investigation.

4.4 Case 4: Rotating Radial Load Combined with Cage Unbalance

In order to further illustrate the features of the graphics animation code, half of the applied radial load of 4,500 N rotates with the inner race in this case. The condition is really imposed by an orbiting inner race with the orbit radius corresponding to the magnitude of the rotating load. Similar to cage whirl motion, the race whirl is also plotted in the animation at an enlarged scale. Note the circular orbit drawn in blue (same color as that of the race) in the animation labeled as Case 4. The circular orbit is offset from the bearing center, since only half of the radial load is rotating with the race. Most of the other features of the animation are similar to those seen in the earlier cases.

To demonstrate another capability of the graphic animation code, the ball/cage simulations in this Case 4 are executed in a mode where the animation pauses as soon as the ball/cage contact is established. Now the animation, during the ball/cage collision, is controlled by a frame-to-frame basis by triggering an input device, such as the mouse button or any key on the keyboard. The rather slow "stepping" seen in the video recording of this animation is really a result of this pause and the manual frame by frame advancement. This permits a close examination of any ball/cage contact.

Since the graphics animation code simply displays the results produced by ADORE, all design capabilities of ADORE can be visually appreciated without any imagination of complex interactions or motions. Of course, the limitations are that presently the animation is restricted to ball bearings and the view angle is fixed along the bearing axis, which means that the motions are only seen in a plane normal to the bearing axis. Further development for 3-D visualization along arbitrary view angles for any type of bearings will subject of the Phase II effort.

5. CONCLUSIONS

Technical accomplishments of present Phase I research can be appreciated in terms of the following conclusions:

1. Graphics animation generated for the several cases of bearing operating conditions clearly demonstrate technical feasibility of the overall modeling approach. Graphics development based on the PHIGS international standard, and a database-type interface of the graphics code with the bearing dynamics code ADORE is proven for a more rigorous development of animations in three dimensions for all types of rolling bearings in Phase II.

2. The simulation of experimentally observed motions, such as those seen with cage unbalance, further demonstrate the practical significance of graphics animation modeling and sophisticated bearing dynamics computer codes, such as ADORE.

3. Parametric animations included on the video tape demonstrate immediate practical utility of the graphics model developed in the present project. The complete interface with ADORE, makes the graphics code immediately usable as a design tool for ball bearings.

6. RECOMMENDATIONS FOR DEVELOPMENTS IN PHASE II

With proven technical feasibility of the modeling approach the graphics animation model can now be further developed into a very powerful design tool. Some of the developments proposed for the second phase of this project are briefly discussed below:

6.1 2-D Display of Ball/Race Interaction

Similar to the ball/cage contact displays developed in Phase I, it is essential to display the details of ball/race contacts. Design parameters such as contact loads, stresses, slip rates, lubricant film thickness, heat generations, temperatures, etc., can be dynamically displayed.

6.2 2-D Models for Cylindrical and Tapered Roller Bearings

The 2-D animation modeling can be extended to all types of rolling bearings modeled by ADORE to generate a graphic interface for all ADORE capabilities. Primary emphasis in this development shall be in the area of graphics structure generation for different geometrical shapes.

6.3 Comprehensive Graphics Input Interface

Both for effective use of the 2-D models and as a prerequisite for the 3-D models, it is essential to develop a comprehensive graphics input interface. Under such input control any of the available displays can be effectively selected, animation speed can be easily controlled, the view angles for 3-D simulation can be dynamically changed, and similar other practical utilities, which significantly increase the effectiveness of the model, can be implemented.

6.4 3-D Model for Ball and Roller Bearings

The PHIGS graphics structures, developed for 2-D images, can now be extended to 3-D representation where each bearing element can be drawn as a solid. Graphics hardware capabilities, such as Z-buffers, hidden lines removal, shaded surfaces, etc., now become essential. Thus, in addition to the software development, it will be important to configure, or to some extent modify the available hardware.

6.5 Integrated Hardware/Software Prototype

Finally, before commercialization of the graphics animation model it will be essential to really develop a prototype system, where the practical strength of the model can be clearly demonstrated. The prototype is expected to consist of available computer hardware, with some possible modifications, and effective integration of the software for optimum performance.

7. REFERENCES

1. Gupta, P.K., ADVANCED DYNAMICS OF ROLLING ELEMENTS, Springer-Verlag, 1984.

2. Foley, J.D., van Dam, A., Feiner, S.K. and Hughes, J.F., COMPUTER GRAPHICS - PRINCIPLES AND PRACTICE, 2nd Edition, Addison Wesley, 1990.

3. Gupta, P.K., "On the Geometrical Imperfections in Ball Bearings", J. of Tribology, ASME Trans, Vol 110F, pp 13-18, 1988.

4. Gupta, P.K., Dill, J.F., Artuso, J.W. and Forster, N.H.,"Ball Bearing Response to Cage Unbalance", J. of Tribology, ASME Trans, Vol 108F, pp 462- 467, 1986.

5. Gupta, P.K., "Visco-Elastic Effects in MIL-L-7808 Type Lubricant - Part III: Model Implementation in Bearing Dynamics Computer Code", Presented at 1991 STLE Annual Meeting, Montreal, Canada, To be published in STLE Tribology Transactions.