PKG-TR-C-200-14

ADORE
Advanced Dynamics Of Rolling Elements

Version 6.00 and higher
User Manual

15 April 2014

by
Pradeep K. Gupta

THIS COMPUTER PROGRAM (ADORE) IS A PROPRIETARY SOFTWARE OF PRADEEP K. GUPTA, INC (PKG). REPRODUCTION IN WHOLE OR IN PART IS PROHIBITED. IT IS EXPRESSLY UNDERSTOOD THAT PKG ASSUMES ASBOLUTELY NO RESPONSIBILITY AND/OR LIABILITY FOR ANY DAMAGE WHICH COULD EITHER BE A DIRECT OR AN INDIRECT RESULT OF AN ERROR IN ADORE AND PKG DOES NOT WARRANT THAT ADORE SHALL BE FREE OF ANY ERRORS OR DEFECTS.

PKG
Pradeep K Gupta Inc
117 Southbury Road
Clifton Park, New York 12065-7714
U. S. A.

FOREWORD

The purpose of this manual is to provide adequate instructions for the use of the computer program ADORE. The manual contains general overview and description of input/output variables of ADORE for simulating the dynamic performance of rolling bearings. Details on the input/output facilities including all graphic processing of the results in also included in this manual.

Table of Contents

FOREWORD **2**

Table of Contents **3**

1. INTRODUCTION **5**

2. SYSTEM REQUIREMENTS AND ADORE INSTALLATION **12**

2.1 System Requirements 12

2.2 Media Contents 12
2.2.1 Disk1 12
2.2.2 Disk2 13
2.2.3 Disk3 13

2.3 Program Installation 13
2.3.1 ADORE Installation 13
2.3.2 AdrInput, AdrPlot and AGORE Installation 13
2.3.3 Setting up Environmental PATH variable 14

2.4 Program Execution 14
2.4.1 Executing AdrInput 15
2.4.2 Executing ADORE 16
2.4.3 Executing AdrPlot 17
2.4.4 Executing AGORE 18

3. ADORE INPUT DATA **20**

3.1 Program Mode and Output Control 21

3.2 Step Size Information and Thermal Environment 25

3.3 Program Options 34

3.4 Bearing Envelope 47

3.5 Rolling Element and Race Geometry 48

3.6 Inertial Parameters for Rolling Elements and Races 67

3.7 Cage Parameters 71

3.8 Material Properties 83

3.9 Operating Conditions 92

3.10 Traction and Friction Parameters 97

3.11 Gravity Effects 133

3.12 Inputs for User Programmable Routines 134

4. ADORE OUTPUT **135**

4.1 Print Output 135
4.1.1 Angular Velocities 135
4.1.2 Angular Positions 136
4.1.3 Rolling Element Contact Depth & Chordal Distance 136
4.1.4 Time Averaged Wear Rates 138
4.1.5 Rolling Element/Cage Contact Angle 138
4.1.6 Cage/Race Contact and Attitude Angles 139
4.1.7 Power Loss 135
4.1.8 Internal Clearance and Operating Fits 140
4.1.9 Fatigue Life 140
4.1.10 Rolling Element Orbital Velocity Ratio 140
4.1.11 Cage Angular Velocity Ratio 140
4.1.12 Cage Whirl Ratio 140

4.2 Plot Output 140

4.2.1 Power Dissipation and Life 141
4.2.2 Rolling Element Motion 142
4.2.3 Cage Motion 145
4.2.4 Race Motion 148
4.3 Graphics Animation Output 151
5. DATA MANAGEMENT IN ADORE 156
5.1 File DATA.txt 156
5.2 File PRINT.txt 156
5.3 File MASTER 156
5.4 File FINAL 157
5.5 Files SOL1 to SOL6 158
5.5.1 Header Information 158
5.5.2 Solution Records 159
5.6 File SOL7 165
5.6.1 Header Information 165
5.6.2 Solution Record 166
5.7 File SOL8 168
5.7.1 Header Information 168
5.7.2 Solution Record 169
5.8 File SOL9 169
6. USER PROGRAMMABLE FUNCTIONS AND SUBROUTINES 170
6.1 Subroutine ADRX0 170
6.2 Subroutine ADRX1 170
6.2.1 Adrx1 Example 1: Angular Acceleration on Inner Race 174
6.2.2 Adrx1 Example 2: Vibrational Loading 177
6.3 Subroutine ADRX2 179
6.4 Subroutine ADRX3 179
6.5 Subroutine ADRX4 180
6.6 Subroutine ADRX5 180
6.7 Subroutine ADRX6 180
6.8 Subroutine ADRX7 180
6.9 Subroutine ADRX8 180
6.10 Subroutine ADRX9 180
6.10.1 Adrx9 Example: Arbitrary Output in File SOL9 180

1. INTRODUCTION

ADORE is an advanced computer program for the real-time simulation of the dynamic performance of rolling bearings. The analytical foundation of ADORE essentially consists of the classical differential equations of motion and the analytical models for the interaction between the various bearing elements. The equations of motion are formulated in a generalized six- degrees-of-freedom system and the interaction models allow for arbitrary geometry of the bearing elements. Thus any arbitrary variation in bearing geometry, such as, geometrical imperfections or manufacturing tolerances, can be modeled and the influence of time varying operating conditions on the general stability of bearing elements can be investigated. ADORE may therefore prove to be a powerful tool for the design of rolling bearings where cage stability, rolling element skid and skew, complex lubrication mechanics and wear of bearing elements impose significant limitations on the performance of the rotor-bearing system.

The types of rolling bearings considered in ADORE include ball, cylindrical roller, tapered roller, spherical tapered roller and radially loaded single row spherical roller bearings. The bearings may be with or without cage and the cage may either be a one piece element or it may be segmented into several pieces. Throughout ADORE, depending on the type of bearing, the term “rolling element”, represents ball, cylindrical roller, spherical roller, tapered roller or spherical tapered roller, and the term “bearing elements” include rolling elements, cage and the outer and inner races. The analytical models in ADORE consist of the following:

1. Rolling element/race interactions.
2. Rolling element/cage interactions.
3. Cage/race interactions.
4. Race flange interactions for roller bearings.
5. External system interactions and constraints.

The rolling element/race interaction provides a model for the computation of normal and tractive forces at the rolling element to race interface. The classical theories of elasticity and elastohydrodynamic lubrication provide the foundations of this model. Rolling element to cage and the cage/race contacts are modeled in terms of the geometrical interaction and an arbitrary constitutive relation for the computation of normal and friction forces. For oil lubricated bearings the conventional hydrodynamic theory is used to model the hydrodynamic effects at the rolling element/cage and cage/race interface. In the case of roller bearings, the contact between the roller and the guide flange on the raceway is modeled in terms of the geometric interaction and the classical elastic contact mechanics. However, the load- deflection relation may be easily replaced by any arbitrary constitutive equation which may be derived from the experimental data obtained for a particular application. Similarly, the traction-slip relation at the roller/flange interface can be arbitrarily prescribed. Roller/flange interactions greatly influence the performance of tapered roller bearings. For cylindrical roller bearings, such interactions become significant when the roller skews due to bearing misalignment, geometrical imperfections or other operational considerations. External system interactions and constraints include models for the applied forces and moments exerted on the bearing elements as a result of their interaction with the operating environment. For example, churning and drag effects as a function of lubricant flow through the bearing, geometrical distortion of the bearing elements due to thermal gradients, shrink fits and centrifugal expansion of the races, and any prescribed loads and/or geometrical constraints on the bearing are considered in this category.

The general motion of any bearing element as a function of the applied forces and moments, computed from the above interactions, is considered in two parts:

1. Motion of the mass center.
2. Rotation of the element about its mass center.

The mass center motion is generally considered in an inertial (space fixed) coordinate frame, as shown below in figure 1. The mass center position may be defined either by the cartesian coordinate (x,y,z) or cylindrical coordinates (x,r,θ). A body-fixed coordinate frame $(\hat{x}, \hat{y}, \hat{z})$, at the element mass center and along the principal inertial axes may also be defined as shown below in figure 1. The angular orientation of the bearing element may then be defined by three angles which define the orientation of this body-fixed frame relative to the inertial frame.

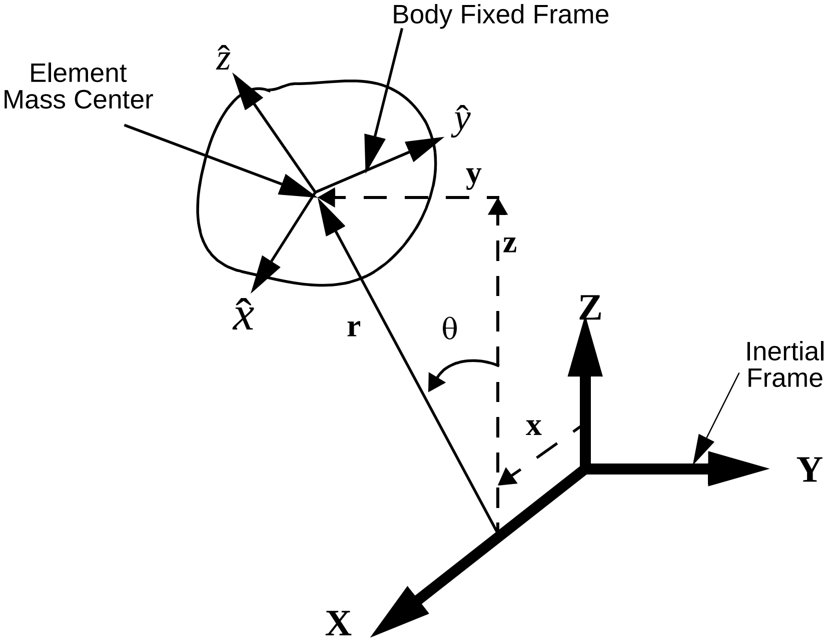


Figure 1. Base coordinate frame for mass center motion.

The three angles which define the angular orientation of the body-fixed frame relative to the fixed inertial frame are Euler-type angles and are defined as follows:

1. Rotation η about the X-axis to arrive at coordinates (x, y', z') ‘

2. Rotation ξ about the y' axis to get the coordinates $(x', y', \hat{z})$.

3. Rotation λ about the $\hat{z}$ axis to arrive at the final coordinate frame $(\hat{x}, \hat{y}, \hat{z})$.

The above transformations are schematically illustrated in figure 2. Similar to the Euler angles, the above transformations result in an orthogonal transformation matrix. Thus practical use of the transformation matrix is numerically very efficient.

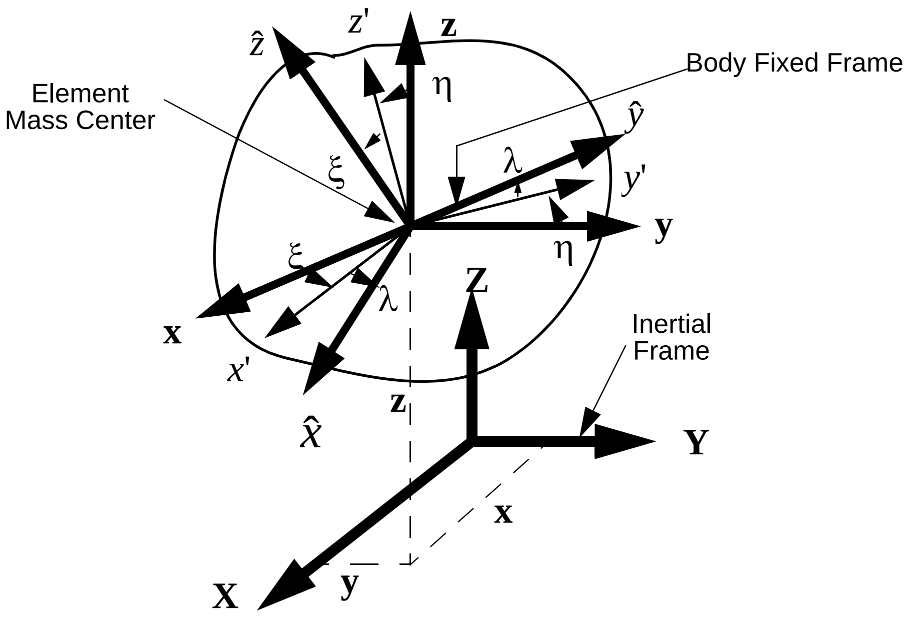


Figure 2. Coordinate transformation from inertial to body-fixed coordinates.

The three mass center coordinates along with the three angles defining the angular orientation constitute the six degrees of freedom available for the simulation of the general motion of the bearing element. These six fundamental coordinates when combined with the six corresponding velocities result in twelve differential equations of motion for each bearing elements. Thus for a bearing with N rolling elements, a one piece cage, and the outer and inner races, the model consists of a system of (N+3)*12 simultaneous first order differential equations. The set of differential equations is numerically integrated to obtain the real-time simulation of the bearing performance. A number of different integrating algorithms, including both explicit Runge-Kutta type formulas and the implicit Predictor-Corrector type algorithm, are available for efficient integration.

ADORE is highly modular in structure. The entire code is divided into a large number of subprograms. As shown schematically in figure 3, the nine basic modules of ADORE are:

1. ADRAn: Input/Output and quasi-static computation
2. ADRBn: Computation of derivatives or accelerations
3. ADRCn: Rolling element/race normal contact forces
4. ADRDn: Rolling element/race traction and lubricant effects
5. ADREn: Rolling element/cage and cage/race interactions
6. ADRFn: Computation of fatigue life
7. ADRGn: Numerical integration algorithms
8. ADRHn: Thermal interactions
9. ADRXn: User-programmable subroutines for special effects

The first three letters, ADR, in the module name represent an abbreviation of ADORE; the fourth letter denotes the module name; and the last letter, n, may assume any numeric value depending on the number of subprograms in the module.

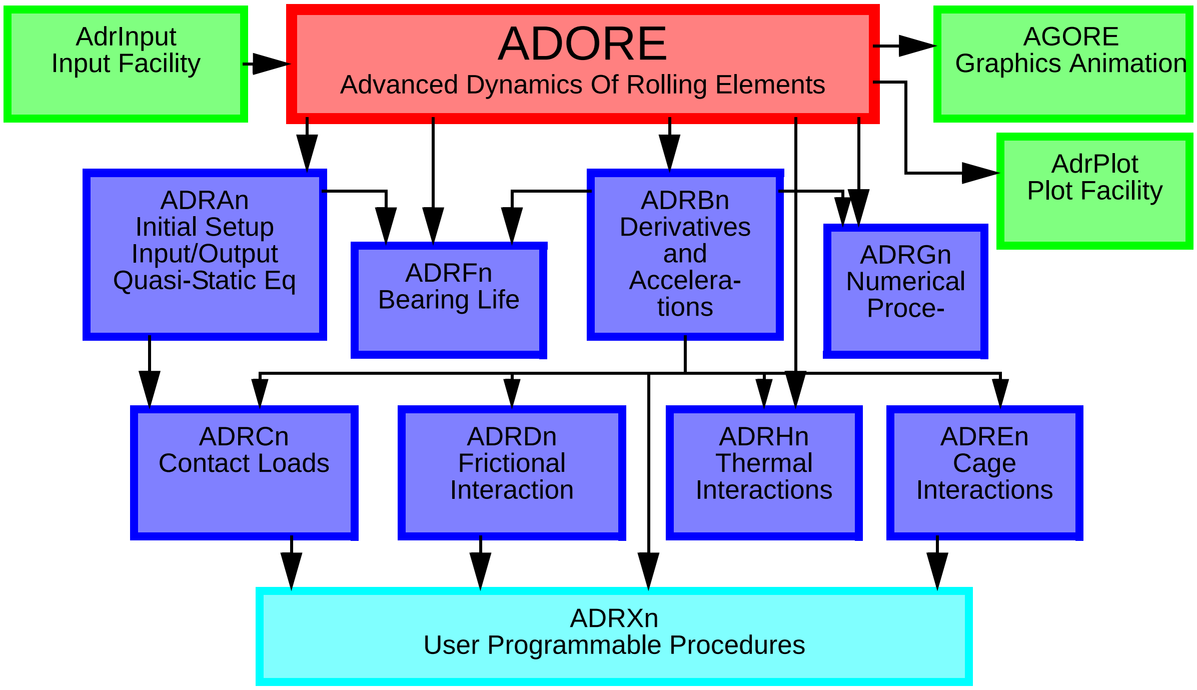


Figure 3. Modular structure of ADORE.

The input facility, AdrInput, is a stand-alone code which prepared the input data set for ADORE. The main program, ADORE calls the module ADRAn for input/output and the computation of the quasi-static solution. Bearing life is computed by calling ADRFn. In the present version of ADORE, the module ADRFn also contains a subroutine for the computation of churning and drag effects. For the dynamic analysis, the two primary modules called by ADORE are ADRBn and ADRGn for computing the accelerations and integrating the differential equations of motion, respectively. Since most of the integrating algorithms used are of order greater than one, ADRBn is also called by the integrator module ADRGn. The module AdrPlot is called by ADORE for plotting purposes and a few initial calls to ADRXn are simply for initialization and for any input/output which may be required by the user- programmable subroutines.

The heart of ADORE is the module ADRBn, which calls the three basic modules ADRCn, ADRDn and ADREn for the computation of rolling element/race normal forces, traction forces and the cage interactions, respectively. All the user-programmable subprograms may be called by any or all of these three modules and the derivative module ADRBn.

The quasi-static module in group ADRA, in addition to providing initial conditions for dynamic simulations, can also be used for computation of conventional design parameters. The overall program operation can actually be divided into three modes: quasi-static mode, dynamic mode and a post processing mode, where the computed results can be graphically displayed either in the form of plots or animation. These modes are schematically illustrated in figure 4.

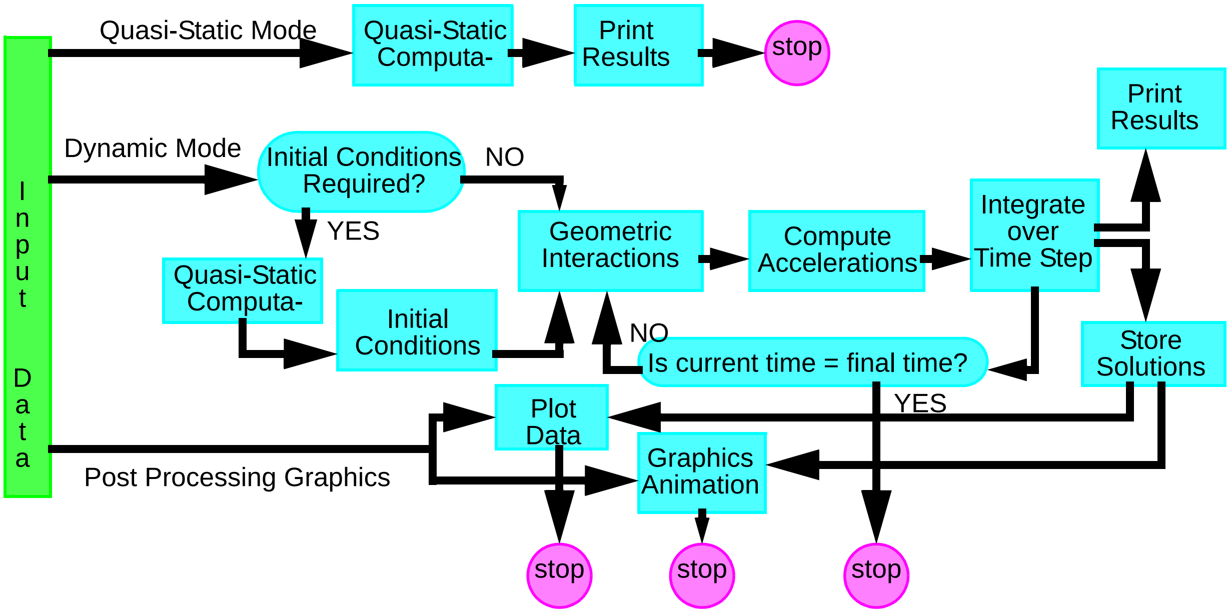


Figure 4. Basic operating modes of ADORE.

While ADORE code is in FORTRAN-90, the input facility, output plot facility and the graphics animation facility are all written in Java. The input facility provides a graphic interface to the user for preparation of input data required by ADORE. Based on the data entered selection of appropriate records is automatic. Thus the input preparation is quite efficient. The program also provides brief description of all data variables interactively.

Once, ADORE is executed for a given problem, the output data, in addition to print file, is stored in a number of data files which are input to the plot utility which provides a graphic display of all parameters in terms of 2-D graphs. Simple 2-D graphic primitives, available within the Java library are used to generate all the graphic output.

Very often the generalized motion of bearing elements as modeled by ADORE may be difficult to fully comprehend by simple two dimensional plots and the printed list of certain parameters. An alternate presentation of the results can be in the form of animated views in which the moving bearing elements may be seen as obtained by solving the equations of motion. The graphic animation facility, AGORE (Animated Graphics Of Rolling Elements) fulfills such an objective. Similar to plot data sets, the dynamic solutions generated by ADORE are stored in a data file, which is subsequently input to AGORE to obtain an animated view of the bearing.

The development approach is based on Java 2-D graphics primitives available as a part of the Java Development Kit. The model is a stand-alone graphics facility, input to which is supplied by bearing dynamics computer code, ADORE. The input basically consists of a data base which contains components of motion of the bearing elements. These fundamental components are used to develop appropriate transformations which are applied on the graphics structures corresponding to the bearing elements. Thus an animated display of bearing motion is produced.

Since graphic animation requires continued refreshing of an image, reasonably fast graphics processing is essential in order to run the animation effectively. In addition, relatively fast integer and floating point processing is required for a reasonable refresh rate.

Input data to AGORE is basically provided via an ASCII data set generated by ADORE. While the bearing element shapes are created by using the drawing primitives available in the Java libraries, the time-varying transformation matrices are computed from the input data base. These transformations are applied on the graphics structures and the modified images are displayed on the monitor to produced an animated motion.

A schematic overview of the technical approach for producing the animated displays in AGORE in shown below in figure 5.

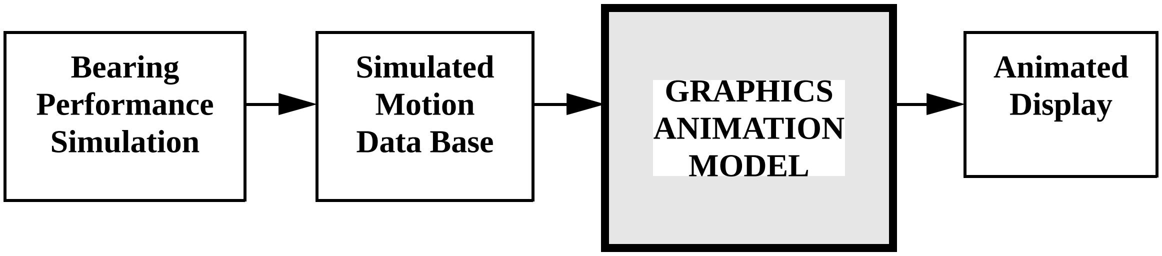


Figure 5. Overview of the approach to graphics animation modeling.

The 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. Output from the graphics model consists of animated displays of pertinent bearing elements. For example, in a ball bearing, the display includes motion of all the balls, cage and the two races.

Based on the above overview of the graphics modeling process, a more detailed outline of development approach used in AGORE is schematically shown in figure 6. The bearing dynamics computer code, ADORE, is executed to generate the simulated dynamics motion of bearing elements. The output is compiled in the form a data base which contains the fundamental components of motion of all bearing elements. The Java class libraries are used to develop the graphics codes which generate the shape of bearing elements from the prescribed geometry. The data base, obtained by using ADORE, is then used to generate the transformation coordinates as a function of time. These transformations are applied on the appropriate graphic elements. Finally, the modified images are posted on the computer monitor. The process is repeated for each time step to produce a continuously refreshed image. Thus an animated view of the bearing is seen on the monitor.

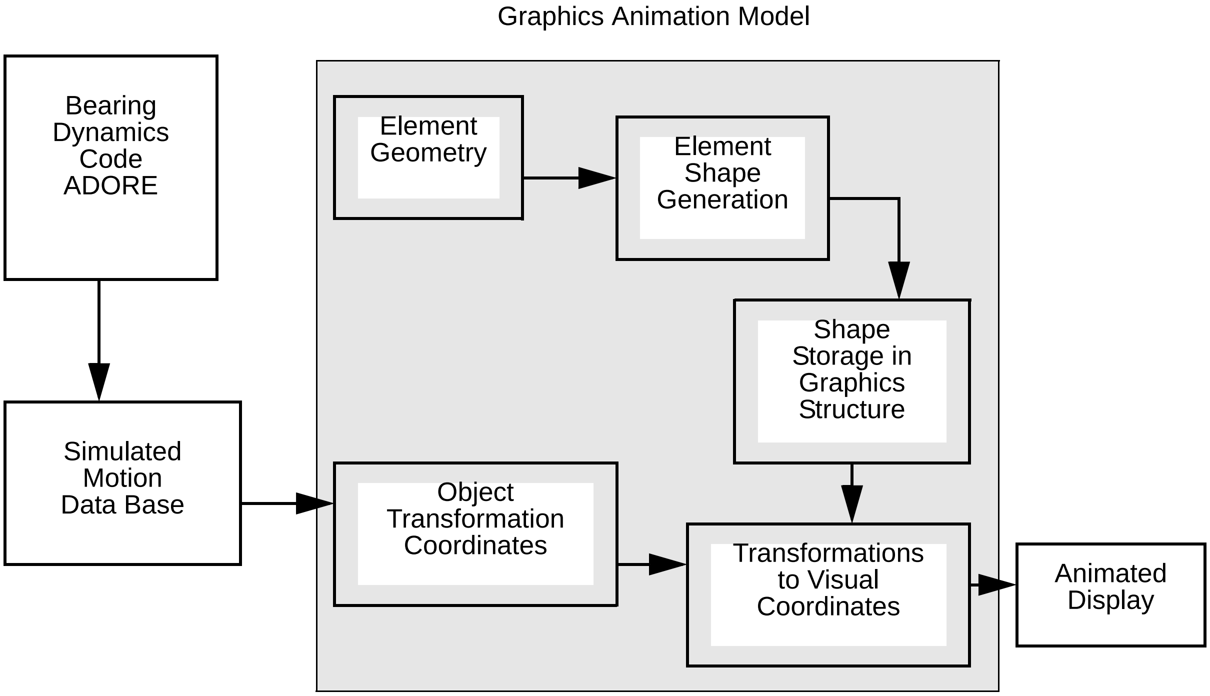

Graphics Animation Model
Bearing Dynamics Code ADORE
Simulated Motion Data Base
Element Geometry
Element Shape Generation
Shape Storage in Graphics Structure
Object Transformation Coordinates
Transformations to Visual Coordinates
Animated Display

Figure 6. Schematic outline of the graphics animation model.

Aside from the input data for the bearing geometry and operating conditions, the user-programmable subroutines provide efficient modeling of complex bearing applications. The required input data, the available output, the data management system and the user programmable subroutines are the primary subjects of this manual. The manual is divided into several chapters. The subjects covered in each of the chapters are briefly reviewed below:

Chapter 2: Computer system requirements, the media contents and some installation details.
Chapter 3: Description of all input data records.
Chapter 4: ADORE data file management system.
Chapter 5: The various user programmable subroutines.
Chapter 6: The graphics options available to process ADORE output.
Chapter 7: ADORE output parameters.

2. SYSTEM REQUIREMENTS AND ADORE INSTALLATION

ADORE is written in ANSI standard FORTRAN 90. The code may, therefore, be installed on virtually any computer system which supports FORTRAN 90. The basic system requirements, media contents and some installation details are subjects of this chapter.

2.1 System Requirements

ADORE is a platform independent software and it is distributed in source code form. The software can be installed on any computer system which supports the appropriate compilers. The following are minimum requirements for installation and effective use of ADORE on any computer system:

1. Central Random Access Memory (RAM) of 10 Mega Bytes.
2. Mass storage of 500 Mega Bytes.
   Larger storage may be required for very long simulations.
3. CD-ROM drive when the software has to be read from a compact disk.
4. A graphic display with appropriate graphics options.
5. A FORTRAN-90 compiler.
6. Java Development Kit for input, plot and graphic animation facilities.

Any FORTRAN-90 or FORTRAN-95 compiler may be used to compile the ADORE source code. Very often a development environment, such as the Microsoft Visual Studio, is available either with the compile or with the computer operating system. This environment may be readily used to compile the ADORE source code and produce appropriate executable.

The Java development kit is in public domain and it can be freely downloaded over the internet for Windows environment from Sun MIcro Systems Web site **http://java.sun.com**. On other platforms, the computer manufacturers may offer their own implementation of Java environment.

2.2 Media Contents

ADORE is normally distributed in source code form on a compact disk. The media content is divided into three subdirectories, labeled as Disk1, Disk2 and Disk3. In addition, a readMe.pdf file is included to provided latest essential information. The contents of each of the directories is outlined below.

2.2.1 Disk1

UpdateXX.pdf: A pdf file containing notes of the latest updates.

adoreManual.pdf: ADORE user’s manual.

adoreInput.txt: Text file containing description of all ADORE input records.

AdrxExamples: Subdirectory containing source codes of ADRX examples.

Ball: Subdirectory containing ball bearing test case.

Roller: Subdirectory containing roller bearing test case.

TaperedRoller: Subdirectory containing tapered roller bearing test case.

2.2.2 Disk2

***.f files:** ADORE FORTRAN-90 source files.

Makefile: File Makefile, a make file for Windows 7 operating system.

2.2.3 Disk3

setup.bat: Batch file to compile AdrInput, AdrPlot and AGORE on Windows system.

AdrInput.bat: Batch file to execute AdrInput.

AdrPlot.bat: Batch file to execute AdrPlot.

agore.bat: Batch file to execute AGORE.

Java: Subdirectory containing all Java source codes.

2.3 Program Installation

The installation procedure presented below is primarily for Windows 7 operating system with a fortran compiler and Java Development Kit already installed. For other systems the following may only provide general guidance. The pertinent development environment and/or compiler instructions should be used to develop specific installation steps.

On a Windows 7 system, assuming that the available installation disk is drive **d:**, carryout the following steps:

1.Create a directory **d:\Adore**
2.Create a subdirectory: **d:\Adore\bin**
3.Copy the program disk contents **d:\Adore directory**

Now all the disk contents will be in the directory **d:\Adore\Adore600**

2.3.1 ADORE Installation

ADORE installation is accomplished by a Makefile provided in the Disk2 subdirectory on the program disk. For other compilers, simply edit the Makefile to change the compiler command “lf95” to the applicable command for the available fortran compiler. After completing appropriate editing of the Makefile carryout the following steps:

1. Open the Command Prompt window to get a command window with a **c:** prompt.
2. Change directory to **d:\Adore\Adore600\Disk2**
3. Execute the command: **nmake**. This will compile all the source files and create an executable adore.exe
4. Copy the executable to the Adore bin directory by running the following command: **copy adore.exe d:\Adore\bin\Adore600.exe**

2.3.2 AdrInput, AdrPlot and Agore Installation:

Assuming that the Java development kit is installed, copy the supplied customized files so that they are located as follows on the computer system:

File setup.bat:	**d:\Adore\setup.bat**
File AdrInput.bat:	**d:\Adore\bin\AdrInput.bat**

File AdrPlot.bat:	**d:\Adore\bin\AdrPlot.bat**
File Agore.bat:	**d:\Adore\bin\Agore.bat**

Edit the path for **javac** compiler file in **setup.bat** if necessary. Now in the command prompt window move to the directory **d:\Adore** and execute the command: **setup**

This will compile all the java source files and create the appropriate class files for executing the AdrInput, AdrPlot and Agore facilities.

2.3.3 Setting up Environmental Path Variable

On a Windows 7 the environmental PATH statement may be modified as follows:

1. From the START menu, click Control Panels

2. Click on System and Security

3. Click on System

4. Click on Advanced system setting, seen on left panel on the screen

5. Click on Environmental Variables button

6. In the lower sub window "System variables.." scroll down to where you see Path variable

7. Click on Path to highlight Path variable

8. Click on the Edit button

9. Now click on the variable value to remove the highlighting and see a cursor bar

10. Use the right arrow key to move to the cursor to the end on this value

11. Type: **;d:\Adore\bin**

12. Now click OK on this window and the rest of them and close the control panel screen.

With the above setup ADORE may now be executed from any directory by simply typing Adore600 at command prompt. Like wise the input, plot and animation facilities, AdrInput, AdrPlot and Agore can be executed by typing the commands: AdrInput, AdrPlot and Agore respectively.

2.4 Program Execution

Since execution of ADORE creates several data files unique to the specific run, it is best to run each case in a specific subdirectory. It is first essential to execute the test cases supplied on the program disk to varying installation. For this purpose carryout the following steps:

1. Create a subdirectory d:\Adore\Test.
2. In the above test directory create a subdirectory d:\Adore\Test\Ball.
3. Copy the input data file DATA.txt located in the Ball subdirectory in the Disk1\Ball
4. From command prompt execute the command: Adore600.
6. Print output can be viewed by opening the file PRINT.txt with Notepad or WordPad. The results may be compared with those supplied on the program disk in \Disk1\Ball directory.
7. To execute the plot facility, type the command: AdrPlot

8. For now just click ok to first couple of screens and then use the File tab to open one of the output plot files, SOL1, SOL2 or SOL7
9. Click on the next button to see the various plots. In the end quit the application.
10. To execute the graphic animation facility, type the command: Agore
11. After clicking ok on first couple of screens click on the File tab to open file SOL8. This will show the bearing view.
12. Click the forward arrow button to make the bearing move.
13. You can now explore other views per directions supplied in Users manual.
14. In the end quit out of the application.

The above process may be repeated for the roller and tapered roller bearing cases if necessary.

In general the execution process consists of the following steps:

1. Execute input facility AdrInput to prepare the input data file.
2. Execute ADORE with the data file prepared in step 1.
3. Execute graphic and animation facilities to examine the results.

Since ADORE interfaces with a number of data files, it is generally best to execute ADORE and all the input and graphic facilities in a command line mode. This is particularly true for executing ADORE. The graphic input and output facilities, may be easily executed in command line mode.

2.4.1 Executing AdrInput

Execution of ADORE input facility AdrInput is accomplish either via command line or by double clicking on the appropriate application icon or its short cut. The graphic user interface provides all instructions for various data variables. Depending on the data values entered, AdrInput automatically prompts the user with applicable data records.

AdrInput starts with certain default values already entered in the various data files. However, after AdrInput has been executed once, and a data file is created, the user has the option of opening this existing data file via the FILE menu tab on the interactive input window. By doing this the data file is opened, all data values are read, and then displayed on the various input screens.

Upon completion of data entry, the data file must be saved before exiting the application. When using the save option, the user has the option to navigate to any arbitrary directory where the data file may be saved.

The various menu options to navigate through the program are as follows:

File Menu

The file menu in menu bar on top of the display window contains the following:

New: Selecting “New” under the file menu will create a new data file for the current data set. File name will be requested later when saving the data.

Open: An existing data file may be opened by selecting this option. Values from the data file shall be read and displayed as defaults. A file navigation window shall be displayed to assist in selection of the file to be opened.

Save: If a file is opened, the Save menu is available to replace the opened file with updated data at any time during execution of AdrInput.

Save As: When no file name defined, this option displays the file navigation window where a new name or an existing file to overwrite that data may be specified.

Quit: This option will terminate execution. However, a warning message indicating that all unsaved data will be destroyed. The Cancel button in this warning message may be used to cancel the Quit option and then the data may be saved.

Help

The Help menu contains some descriptive information about program use. Most of this information in displayed in message windows which may be closed by clicking the OK button in the windows. The various sub options are quite self-explanatory.

Go Back

At the bottom of the display window, clicking the “Go Back” option will bring back the previous data record for further updates. In case the first valid record is already displayed, then a message indicating such a fact shall be displayed.

Next Rec

Click “Next Rec” to move to the next data record.

Save & Exit

The “Save & Exit” option is equivalent to selecting Save and then Quit under the File menu. If the current file name is already known, the data will be saved in the file and AdrInput shall terminate; other wise the file navigation window shall be displayed to request a file name.Afte the data is saved, AdrInput shall terminate.

2.4.2 Executing ADORE

After creating the input data file with AdrInput, ADORE may be simple executed by in command line mode form the directory in which DATA.txt is stored, as illustrated above for the test case. Note that aside from the PRINT.txt output file, ADORE created several other data files with varying amounts of data, as described later in this manual. If any of these files exist in the working directory before executing ADORE, then the files are overwritten as the execution continues. In the event ADORE is being executed in a continuation mode, where the previously computed solutions are being advanced further in time, then the new data is appended to the old data in the existing data files. To facilitate such data handling it is always desirable to run each case from a different working directory as suggested above.

2.4.3 Executing AdrPlot

Similar to AdrInput, AdrPlot may be executed either in command line or in system graphic environment. After accepting the application disclaimer, the user is prompted to open a valid ADORE data set and set the initial default plot parameters. After the file is opened certain keywords in the file are validated to ascertain the file was generated by ADORE. If this validation procedure fails, the user is accordingly prompted. When the data set is valid, plotting may either be done under default parameters or new values may be set. If new values are desired, then the three inputs: start point, end point and data plot interval, are interactively requested. The entire file is now read and plot data is setup to display the various plots. Depending on the size of the file and speed of the available processor, this could take several minutes. Upon completion of the setup procedure the first plot is displayed.

Depending on the resolution of the monitor, the size of the graphic window may be have to be adjusted to display the graphs in acceptable form. However, the window size can only be changed once upon start of the application. Thus if the graphics are not acceptable, exit of the application, restart and change the window size after the first graph is displayed. These problems generally do not exist with high resolution monitors.

The various menu options to navigate through the program are as follows:

Open Plot File

Click this button to select another plot file. The plot options are requested again for the new data set.

Prev Plot

Clicking this button decrements and plot number by one and displays the new plot. If the window already contains the first plot, then an appropriate message is displayed.

Next Plot

Similar to the Prev Plot button, this button increments the plot number and displays the new plot. If the last plot is already in the graphics window, then an appropriate message is displayed.

Plot Number

In the event a specific plot is desired, then this button may be used to enter the desired plot number and display the appropriate plot.

Print

This option will prompt the user with the printer selection menu to select one of the connected printers on which the graphic output is desired. Note that this application does not have a “Page Setup” option, so if the graph does not fit the default page size, it is truncated. It is, therefore, best to save the graph as a jpeg image first, by using the next option, then printing the image with one of the other available applications.

Save JPG

By using the option the graphic image may be saved as a jpeg file. First time this option is selected a full path name for the file to be saved must be specified. Subsequent save will contain the previously selected path and file name appended by an incremental number. The default name can of course be changed if so desired.

Quit

This button will simply quit the application.

2.4.4 Executing AGORE

Similar to the other Java applications, AdrInput and AdrPlot, execution of AGORE is straight forward either via command line or by double clicking the application icon. After acceptance of the normal disclaimer the graphic window is displayed and the user is prompted to open a data set to be processed. The data set corresponds to the animation data file which contains the bearing motion as a function of time, as generated by ADORE. As this point the size of the graphics window may be interactively adjusted. After acceptable window adjusted, click the file menu tab to open the data set. Before the file navigation window is displayed the user is prompted to enter the number of time steps over which the animation is to performed. This number of steps corresponds to the number of time steps over which ADORE simulations were obtained. The number of steps for animation can be less than or equal to the number of solution steps in the data set. Depending on the amount of data it may take some time for AGORE to process the data, set up the various transformations, scales for pertinent data values, and other analytical details before the first image appears in the display area. After the image is displayed all user interactions are interactive.

The following options are available in the menu bar:

File

Open: Open data set.

Quit: Quit application.

View

Bearing Motion: Display composite bearing motion.

Cage Motion: Display cage motion.

Pocket Interaction: Display cage pocket interaction.

RE Motion: Display rolling element motion.

Race Motion: Display outer or inner race motion.

Flange Interaction: For roller bearings display outer or inner race flange interactions.

Help

About AGORE: Information about AGORE compatibility with ADORE version. Data set from all ADORE versions equal to or higher than that stated in this information will be acceptable.

For a give view the animated motion is controlled by the various options displayed to the right of the graphic display. The various options are:

>Play

Animate motion in forward direction.

<Play

Animate motion in reverse direction.

>Frame

Animate motion frame by frame in forward direction.

<Frame

Animate motion frame by frame in reverse direction.

Pause

Pause animated motion.

Print

Print the graphic image to available printer.

Save

Save the graphic image as a jpeg file.

Quit

Quit application.

3. ADORE INPUT DATA

ADORE input data file is a standard ASCII text file. It may be prepared by using any available text editor. Alternatively one of the ADORE input facilities may be used to prepare the input interactively. A detailed explanation of the various input variables is the subject of this chapter. section of the manual. Most the information presented below is also available on the interactive help screens, which are part of the input facilities.

Before discussing the data records in detail, the following brief comments about data format may be noted.

1. All the data is assembled in an ASCII text file.
2. The first variable on each data record, recID, is a text string, with a maximum of 12 characters, enclosed in single quotes. The string is simply read and printed out in the input data list. Although the string may contain any arbitrary information, it is recommended that the record title is coded here. This facilitates identification of invalid data records when executing ADORE.
3. All variable names beginning with letter a-h and o-z are real floating point numbers and it is essential to specify decimal point in appropriate location. These variable names are color coded to red in the following discussion. Variable names beginning with letter i-n are all integers and these must be coded with no decimal point. These variables are color coded to blue in the following discussion. All other variables are character variables and they must enclosed in single quotes, such as the variable recID. These variables are not color coded and they are left at the default text color.
4. The data is assembled in free format, as permitted by ANSI FORTRAN-90 standard. A comma or a space may be used as delimiter.
5. Not all data records are required all the time. The conditions under which the data record is required are indicated just below the record title.
7. Some variables refer to a base coordinate frame. All coordinate frames used in ADORE conform to the right hand screw rule, with X being the bearing axis and Z pointing radially upwards in the direction of applied radial load. The base coordinate frame is shown below in figure 7.
8. Either the SI or the English system of units may be used in ADORE. All dimensional quantities are expressed in fundamental units of mass, length, force, time and temperature. The various quantities used in the two system of units are tabulated below:

Table 1: System of Units Employed in ADORE

Quantity	English System	SI System
Mass	Pound Mass (lbm)	Kilogram Mass (kgm)
Length	Inch (in)	Meter (m)
Force	Pound Force (lbf)	Newton (N)
Time	Second (s)	Second (s)
Temperature	Degree Rankine (R)	Degree Kelvin (K)

ADORE input is divided into twelve sets of data record. A description of the various data records and variable in each of these sets is the subject of this chapter.

3.1 Program Mode and Output Control

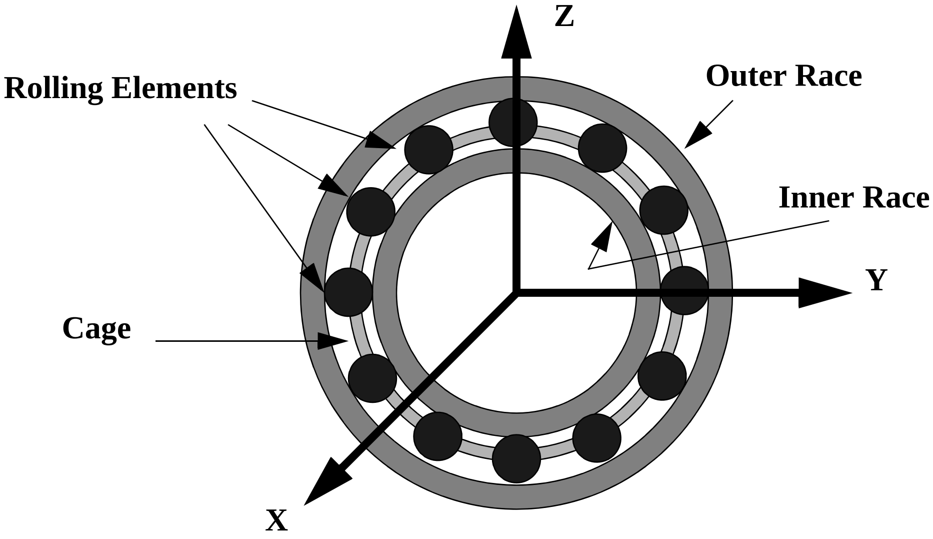


Figure 7. Base coordinate system.

Record 1

Program Mode and Output Control

This data record is always required.

recID

Record identifier - maximum 12 characters in single quotes.

mode

mode, is perhaps, the most important input variable. ADORE may be used to either carry out a simple quasi-static analysis or a dynamic analysis with varying degrees of constraints. An equilibrium analysis is performed in the quasi-static mode, and characteristics such as fatigue life, stiffness, general load distribution, etc., are computed. The dynamic mode is really the prime mode of operation, where the classical differential equations of motion are integrated as a function of time to obtain a real-time simulation of dynamic performance the bearing. The integration requires specification of initial conditions or solutions at starting value of time. Upon startup of a simulation these conditions may either be prescribed arbitrarily or a quasi-static analysis may be performed to set the initial conditions. In the event of a continuation run, where the simulations are advanced further in time, the solutions at previously computed time step may be used to set the initial conditions. The variable kIcOpt, discussed later on this record defines the pertinent option.

In the case of a quasi-static solution, the conventional race control hypothesis is used for ball bearings. In addition, the balls may be held in equilibrium against the gyroscopic moment by applying a fictitious friction force in the contact with the controlling race. Such a gyroscopic restraint is imposed when mode is set to -2. In the event, the equilibrium solution is desired without such a restraint, a value of -1 is used.

When the quasi-static mode is used to compute the initial conditions, it is not desirable to impose any fictitious constraints. In addition, by setting an appropriate value of variable kAngVel on record 3.3, the race control hypothesis may be replaced by arbitrary specification of the angular vector orientation and then the power dissipated in the ball/race contacts may be computed to determine a orientation which results in minimum energy dissipation. This solution may then be used to prescribe the initial conditions. For most conditions this option has been found to provide faster convergence to steady-state.

The fully generalized dynamic model with all six-degrees-of- freedom is invoked by mode =0. In terms of the required computer time, this is, perhaps, the most demanding mode of ADORE, since the time steps size is determined by the highest frequency in the system, which happens to correspond to the ball/race contact vibration. When such a high frequency vibration is not of interest, a time-varying equilibrium constraints may be imposed to eliminate the very high frequency motions. Thus permissible size of the time step may be significantly increased and performance simulation over extended times may be obtained in greatly reduced computing effort. Such a constraint is imposed by setting the value of mode to either 1 or 2. With mode =1, the mass center position of all rolling elements is determined by solving the axial and radial force equilibrium equations and the position of the races is held fixed; for a radially loaded bearing, this will result in a slight variation in the radial load on the bearing as the rolling elements travel in their orbit. With mode =2, however, both the position of the races and the rolling elements may be determined from the equilibrium equations; this will result in a fixed load but the relative position of the races may vary slightly. In terms of the required computational effort per unit rotation of the bearing, mode =1, is probably be most efficient for most bearing applications. For roller bearings with extensive roller skew, however, it may be necessary to let the roller mass center accelerate in accordance to the roller/race load variations resulting from the dynamic tilt and skew of the roller and an axial and radial equilibrium constraint may not be realistic; under such conditions, realistic simulation of the dynamic performance can only be obtained with mode =0.

Thus the program mode defined as follows:

-2 Quasi-static equilibrium solution with gyroscopic restraints as used in race control hypothesis for ball bearings.

-1 Quasi-static equilibrium solution without gyroscopic restraints.

0 Generalized dynamic simulation.

1 Dynamic simulation with equilibrium constraints on rolling elements where the radial and axial equilibrium is performed only for the rolling elements and the position of the race centers is either fixed or prescribed in accordance to any predetermined path.

2 Dynamic simulation with equilibrium constraints on both the rolling elements and the races. the equilibrium equations determine the position of all rolling elements and also the relative position of the two races. generally the outer race will be held fixed while relative position of the inner race is determined by the equilibrium equations.

kDCR

Dynamic constraints on the races:

0 Use defaults, where race mass centers are permitted to move in prescribed displacement field.

1 Specific constraints are included on record 2.2.

kDOF

Add selective suppression of degrees-of-freedom (DOF) on bearing elements to the constraints prescribed by “mode”.

0 No additional suppression of degrees of freedom.

1 Suppress axial translational DOF on rolling elements and cage

2 Suppress axial translational and transverse (y & z) rotational DOF on rolling elements and cage.

3 Suppress axial translational DOF on rolling elements only.

4 Suppress axial translational and transverse (y & z) rotational DOF on rolling elements only.

5 Suppress all degrees of freedom on rolling elements.

6 Arbitrary suppression in user subroutine Adrx1.

kIcOpt

Initial conditions option for dynamic mode:

0 Initial start-up run, compute initial conditions from quasi-static analysis.

1 Continuation of a previous run, read initial conditions from file FINAL and input data starting Record 3 from file MASTER.

During this startup run all bearing geometry and operating conditions data is stored in data set MASTER, along with the last solution vector computed in this run. This last solution vector may be used as initial condition for a subsequent run which simulates bearing performance over extended time. Such a run is called a continuation run, and it is selected by setting kIcOpt = 1. In such a mode, since all the bearing data is available in file MASTER, the input data records 3 to 11 are not required. However, any inputs required by the optional user subroutines must be prescribed in accordance to the requirements of the optional code. All these input records must follow immediately after the series 2 input records. After an initial startup run, any number of continuation runs may be executed in series. Each time a run is made, the solution vector in file MASTER is replaced by the current last solution vector. The bearing data, of course, remains preserved in its original form, as created in the startup run. Thus after the startup run only input records in series 1 and 2, along with any data required by the optional subroutines, are required. In order to maintain numerical continuity in the automated step sizing algorithm, it is essential to know the step size and maximum truncation error incurred at the last step. Using these values the starting step size, in a continuation run can be estimated in accordance to the same procedure as used in any continuous run. Thus numerical continuity in the step sizing scheme can be maintained. Such a continuity in step size equal to zero on record 2.1 in a continuation run. In such a case the values of last step size and truncation error, also available from file MASTER, are used to perform appropriate computation of the starting step size.

-1 Initial start-up run, read arbitrary initial conditions from file FINAL.

kFnOpt

File name option for dynamic mode:

0 use default file names.

1 file names prescribed on record 2.3.

kPrtOpt

Print output option to control the amount of print output at any time step.

The amount of print output from ADORE can be greatly controlled by the user. The first part of the output, which is always printed, consists of the input data containing the bearing geometry, material properties, inertial parameters, lubrication parameters, initial operating conditions, the various scale factors and any output produced by the user programmable subroutines.

Following this output ADORE prints the stiffness-speed table, if computed, or a one page output for the quasi-static solution if ADORE is run with mode < 0 on record 1. For a dynamic solution (mode >= 0), the print output at each time step is divide in four sections with consist of the following:

1. Rolling element parameters.
1a. Load distribution along roller no. 1
1b. Race flange interaction.
1c. Roller end and race flange wear distribution.
2. Race and cage parameters.
3. Applied parameters.
4. Time step summary.

The variable kPrtOpt is thus defined as follows:

-2 Print section 4 output only.

-1 Print sections 3 and 4 only.

0 Print output sections 2, 3 and 4.

n (n > 0) print all sections but print solutions for every nth rolling element. n=1 will print all rolling element solutions, n=2 will print solutions for every other rolling element and so on.

kPrtFreq

Frequency of time steps for print output. kPrtFreq =1 will print solutions at every step, kPrtFreq =2 will print at every other step etc. Time=0 corresponds to step #0.

kPltFreq

Frequency of time steps for plot output at which data is stored. kPltFreq =1 will store all solutions, kPltFreq =2, will store solutions at every other step and so on.

kAGraf

Graphics animation option:

0 Suppress graphic animation data file.

n (n > 0) prepare graphics animation data file and use the value n as frequency of time steps to store data in the graphics animation file.

kLifeFreq

Frequency of fatigue life computation. kLifeFreq =1 results in life computation at every time step, kLifeFreq =2 permits life computation at every other step, and so on. kLifeFreq =0 results in life computation at the first and last step only.

kTherm

Thermal analysis option:

0 no thermal analysis required

1 perform thermal analysis

maxStps

Maximum number of steps for this run.

The length of a run is defined either by the maximum number of steps, maxStps, specified here, or the final time, fTime, specified on Record 2.1, whichever is encountered first. Since the step size is generally variable it may not be possible to determine the actual number of steps for a prescribed final time and, therefore, it may be difficult to estimate the time required to complete the run. For this reason it may be desirable to terminate the run my the maximum number of steps, maxStps. This is simply accomplished be setting fTime to a very large value, which may be reached in the number of steps prescribed by maxStps.

nStps

Number of substeps within a step over which integration is performed but no data is saved.

For simulations over very large number of steps, it may not be necessary to process output data at every steps. In such a case nStps defines the number of steps over which output processing will be skipped after performing the integration. In fact, this skipped step will neither update the step counter leading to maxStps, nor enter the output selection algorithm defined by kPrtFreq, kPltFreq and kLifeFreq.

intMet

Integration algorithm defined as follows:

1-7 Explicit Runge-Kutta-Fehlberg method of order intMet.

11-18 Predictor-corrector method or order (intMet-10). An explicit method of order 4 is used to start the predictor-corrector process.
THIS OPTION IS PRESENTLY NOT AVAILABLE.

An initial trial value of intMet =5 is suggested.

3.2 Step Size Information and Thermal Environment

Record 2.1

Time Step Information

This record is only required for dynamic simulations, mode >= 0 on Record 1.

recID

Record identifier - maximum 12 characters in single quotes.

stpInit

Initial size of dimensionless time step (an initial trial) Suggested value = 0.050.

In order to facilitate modeling of all ranges of geometries and applied operating conditions, and permit computation of numerical truncation error to control convergence of the integration procedure, ADORE performs the entire analysis in dimensionless form. While the length and force scales for the dimensional organization are defined respectively by the rolling element radius and maximum applied load component, the time scale is defined by the natural frequency of rolling element to race contact vibration, which of course depends on bearing geometry and applied loads. Thus the time scale is not know apriori. It is therefore necessary to simply use the default, or any other time values on this record, for the initial run. Once this initial run is completed, the time scale will be printed in the output. This scale may then be used to divide the real time by to arrive at a dimensionless time.

For a continuation run, kIcOpt = 1 on Record 1, the starting step size, may be set equal to zero. In such a case the last step size, which is read from file MASTER, is used as the starting step size to maintain continuity in the step size optimization procedure.

stpMin

Minimum permissible size of dimensionless time step. Suggested default = 5.0e-04.

stpMax

Maximum permissible size of dimensionless time step. Suggested default = 0.50.

fTime

Final value of dimensionless time. Suggested default = 1000.

tol

Local truncation limit. Suggested default = 1.0e-06.

qFac

Ratio of contact load to maximum applied load, below which the rolling elements will be subject to equilibrium constraint under generalized dynamic mode (mode =0 on Record 1).

When performing generalized simulations with all six degrees-of-freedom, the rolling element to race vibration may be excessive under a large radial load when the rolling elements have to enter and exit the load zone. The problem becomes more complex for roller bearing when the entering and exiting rollers may be both misaligned and skewed. In order to take care of this problem ADORE assumes that the rolling element are subjected to an equilibrium constraint when the ratio of rolling element to race contact load and the applied radial load is less than or equal to qFac. For roller bearings, the equilibrium constraint also forces the roller to be perfected aligned, i.e., no misalignment or skew.

Record 2.2

Dynamic Force or Displacement Constraints

Data on this record is required when mode >= 0 and kDCR =1 on Record 1.

ADORE offers the option of either prescribing the forces or displacements on the bearing races. When forces are prescribed the race masses are used to compute accelerations, while no mass properties are necessary when race accelerations are prescribed. Like wise when moments are prescribed the angular accelerations are computed by dividing the applied moments by appropriate moments of inertia, while no inertial properties are necessary when angular accelerations are prescribed. These two options are generally referred to as “force field” and “displacement field” options corresponding to the conditions of proscribed forces and displacements or accelerations respectively. In a normal bearing operating under constant loads and speed, the rotational motions are constrained by the constant rotational velocity and thus all angular accelerations are set to zero. Like wise corresponding to the applied loads the relative race displacements are computed from equilibrium constraints and then the race mass center velocities and accelerations are set to zero. Thus the entire treatment is in displacement field. This is the default condition in ADORE.

When any mass center or angular acceleration on the races is desired under prescribed forces or moments, then kDCR must be set to 1 on Record 1 and the appropriate constraints must be prescribed on this record. It should be noted that if the accelerations are prescribed directly or when an equilibrium constraint under variable applied load is applied by setting mode =2 on Record 1, the default conditions are still valid and data on this record is not required. The data is only required when the races have to accelerate with given inertial properties under prescribed loads and moments. Further note that all exhalations and time-varying conditions are prescribed in user programmable subroutine Adrx1.

With reference to the base coordinate frame shown below, in figure 8, there are a total of six degrees of freedom for each of the races. Mass center motions in the (X, Y, Z) frame and rotation about the (X, Y, Z) axes.

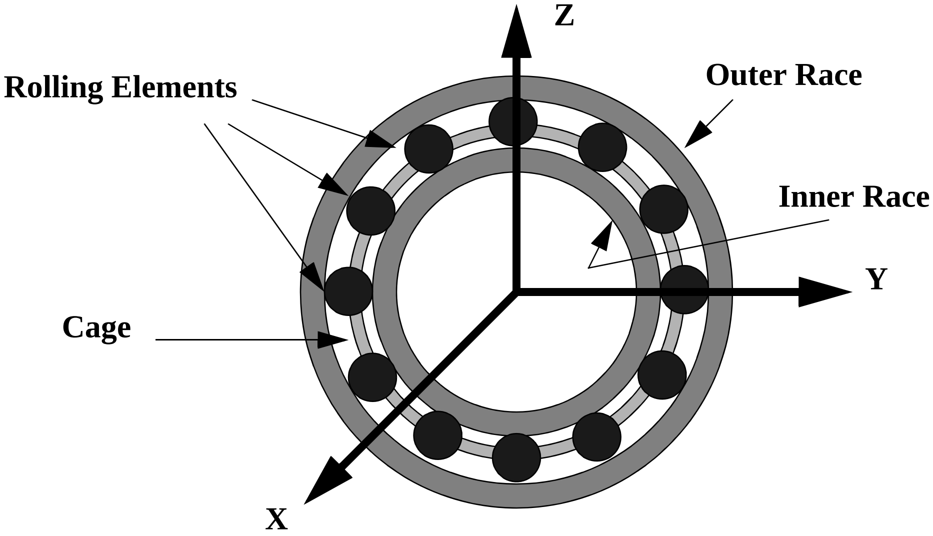


Figure 8. Base coordinate system.

Corresponding to these degrees of freedom there are six flags for each of the races. The values for these flags are set to either 0 or 1 corresponding to force field or displacement field options respectively. The default value is 1 for each component.

Although there is a provision on this record to prescribe each component independently, the following restrictions must be noted:

1. For any equilibrium constraint all kFD flags must be set to 1.
2. Moment constraints (kFD2x and kFD3x) must have equal values.

recID

Record identifier - maximum 12 characters in single quotes.

kFD11

Dynamic force or displacement constraint on outer race along x-axis (see general discussion above):

0 Race accelerates under prescribed load, which is input later on Record 9 and it may be subsequently updated in optional user subroutine Adrx1.

1 Race is held fixed at initial position, the subsequent position is computed by equilibrium constraint, or it accelerates under arbitrary accelerations prescribed in optional user subroutine Adrx1.

Default value is 1.

kFD21

Dynamic force or displacement constraint on outer race along y-axis (see general discussion above):

0 Race accelerates under prescribed load, which is input later on Record 9 and it may be subsequently updated in optional user subroutine Adrx1.

1 Race is held fixed at initial position, the subsequent position is computed by equilibrium constraint, or it accelerates under arbitrary accelerations prescribed in optional user subroutine Adrx1.

Default value is 1.

kFD31

Dynamic force or displacement constraint on outer race along z-axis (see general discussion above):

0 Race accelerates under prescribed load, which is input later on Record 9 and it may be subsequently updated in optional user subroutine Adrx1.

1 Race is held fixed at initial position, the subsequent position is computed by equilibrium constraint, or it accelerates under arbitrary accelerations prescribed in optional user subroutine Adrx1.

Default value is 1.

kFD12

Dynamic force or displacement constraint on inner race along x-axis (see general discussion above):

0 Race accelerates under prescribed load, which is input later on Record 9 and it may be subsequently updated in optional user subroutine Adrx1.

1 Race is held fixed at initial position, the subsequent position is computed by equilibrium constraint, or it accelerates under arbitrary accelerations prescribed in optional user subroutine Adrx1.

Default value is 1.

kFD22

Dynamic force or displacement constraint on inner race along y-axis (see general discussion above):

0 Race accelerates under prescribed load, which is input later on Record 9 and it may be subsequently updated in optional user subroutine Adrx1.

1 Race is held fixed at initial position, the subsequent position is computed by equilibrium constraint, or it accelerates under arbitrary accelerations prescribed in optional user subroutine Adrx1.

Default value is 1.

kFD32

Dynamic force or displacement constraint on inner race along z-axis (see general discussion above):

0 Race accelerates under prescribed load, which is input later on Record 9 and it may be subsequently updated in optional user subroutine Adrx1.

1 Race is held fixed at initial position, the subsequent position is computed by equilibrium constraint, or it accelerates under arbitrary accelerations prescribed in optional user subroutine Adrx1.

Default value is 1.

kMD11

Dynamic moment or rotational constraint on outer race along the x-axis (see general discussion under this record title):

0 Race accelerates under arbitrary moment prescribed in optional subroutine Adrx1.

1 Race rotates at fixed speed prescribed later in Record 9 or it may subsequently accelerate under arbitrary angular accelerations prescribed in optional subroutine Adrx1.

Default value is 1.

kMD21

Dynamic moment or rotational constraint on outer race along the y-axis (see general discussion under this record title):

0 Race accelerates under arbitrary moment prescribed in optional subroutine Adrx1.

1 Race rotates at fixed speed prescribed later in Record 9 or it may subsequently accelerate under arbitrary angular accelerations prescribed in optional subroutine Adrx1.

Default value is 1.

kMD31

Dynamic moment or rotational constraint on outer race along the z-axis (see general discussion under this record title):

0 Race accelerates under arbitrary moment prescribed in optional subroutine Adrx1.

1 Race rotates at fixed speed prescribed later in Record 9 or it may subsequently accelerate under arbitrary angular accelerations prescribed in optional subroutine Adrx1.

Default value is 1.

kMD12

Dynamic moment or rotational constraint on inner race along the x-axis (see general discussion under this record title):

0 Race accelerates under arbitrary moment prescribed in optional subroutine Adrx1.

1 Race rotates at fixed speed prescribed later in Record 9 or it may subsequently accelerate under arbitrary angular accelerations prescribed in optional subroutine Adrx1.

Default value is 1.

kMD22

Dynamic moment or rotational constraint on inner race along the y-axis (see general discussion under this record title):

0 Race accelerates under arbitrary moment prescribed in optional subroutine Adrx1.

1 Race rotates at fixed speed prescribed later in Record 9 or it may subsequently accelerate under arbitrary angular accelerations prescribed in optional subroutine Adrx1.

Default value is 1.

kMD32

Dynamic moment or rotational constraint on inner race along the z-axis (see general discussion under this record title):

0 Race accelerates under arbitrary moment prescribed in optional subroutine Adrx1.

1 Race rotates at fixed speed prescribed later in Record 9 or it may subsequently accelerate under arbitrary angular accelerations prescribed in optional subroutine Adrx1.

Default value is 1.

Record 2.3

Optional Data File Names

This data record is required only when mode >= 0 on kFnOpt =1 on Record 1

ADORE uses several data files, as discussed in the chapter named “Data Management in ADORE”. Although each of these files have a default name, data on this record permits the user to use any arbitrary names for the data files created and used by ADORE. All data file name are character variables with a maximum of ten characters enclosed in single quotes.

recID

Record identifier - maximum 12 characters in single quotes.

masName

Name of master data file, maximum 10 characters enclosed in single quotes.

Default name is MASTER.

finName

Name of the final solution file, maximum 10 characters enclosed in single quotes.

This file is also used to read arbitrary initial conditions when kIcOpt <0 on Record 1.

Default name is FINAL

pltNames1

Plot solution file for element #1, maximum 10 characters enclosed in single quotes.

Default name is SOL1.

pltNames2

Plot solution file for element #2, maximum 10 characters enclosed in single quotes.

Default name is SOL2.

pltNames3

Plot solution file for element #3, maximum 10 characters enclosed in single quotes.

Default name is SOL3.

pltNames4

Plot solution file for element #4, maximum 10 characters enclosed in single quotes.

Default name is SOL4.

pltNames5

Plot solution file for element #5, maximum 10 characters enclosed in single quotes.

Default name is SOL5.

pltNames6

Plot solution file for element #6, maximum 10 characters enclosed in single quotes.

Default name is SOL6.

pltNames7

Power dissipation and life solutions file, maximum 10 characters enclosed in single quotes.

Default name is SOL7.

pltNames8

Graphic animation data file, maximum 10 characters enclosed in single quotes.

Default name is SOL8.

Record 2.4

Thermal Analysis Options

This record is required when kTherm > 0 on Record 1.

Data on this record defines options for thermal analysis.

recID

Record identifier - maximum 12 characters in single quotes.

kCoolant

Type of coolant for the bearing:

0 No coolant

1 Lubricant cooled

2 Arbitrary coolant with prescribed properties

3 Liquid oxygen

4 Liquid hydrogen

5 Liquid nitrogen

6 Air

7 Water

kBaseTemp

Base temperature when kCoolant =0:

0 Housing temperature. The prescribed value is used as reference temperature on housing exterior surface

1 Shaft temperature. The prescribed value is used as reference temperature on interior shaft surface

kHTC

Rolling element heat transfer coefficient option:

0 Compute convective heat transfer coefficient for the rolling elements.

1 Use coefficient prescribed on Record 2.5.

kGeoMod

Constraint for thermal distortion of bearing elements:

0 Do not change bearing element geometry as a function of temperature.

1 Compute appropriate change in bearing geometry as a function of temperature.

Record 2.5

Additional Options for Thermal Analysis

The data record is required only when kTherm > 0 on Record 1.

All data on this record is dimensional. It is essential that the units conform to the unit code defined later on Record 3.2. The units given below in parenthesis correspond to the SI and English system of units, as discussed at the beginning of this chapter.

recID

Record identifier - maximum 12 characters in single quotes.

cTempIn

Inlet temperature of the coolant (K or R).

When kCoolant > 0 on Rec 2.4. cTempIn is the temperature of the coolant as it enters the bearing. For kCoolant =0, cTempIn is not used.

cFlowRate

Coolant flow rate (m^3/s or in^3/s),kCoolant > 0 on Rec 2.4.

For a prescribed coolant, kCoolant > 0, on Rec 2.4, cFlowRate is the volumetric flow rate of the prescribed coolant. For kCoolant =0, cFlowRate is not used.

reHTC

Convective heat transfer coefficient for rolling elements. Set reHTC =0, when heat transfer coefficient has to be computed, as defined by kHTC =0 on Rec 2.4.

aveTime

Actual time (s) over which heat generations are to be averaged for thermal interactions.

skipTime

Actual initial time (s) over which any update of bearing geometry due to thermal interactions will be skipped.

Record 2.6

Coolant Properties

This data required only when kCoolant = 2 on record 2.4.

Coolant properties must be specified at a temperature close to expected exit temperature of the coolant.

All data on this record is dimensional. It is essential that the units conform to the unit code defined later on Record 3.2. The units given below in parenthesis correspond to the SI and English system of units, as discussed at the beginning of this chapter.

recID

Record identifier - maximum 12 characters in single quotes.

xRo

Density (kgm/m^3 or lbm/in^3) of the coolant.

xMu

Viscosity (N.s/m^2 or lbf.s/in^2) of the coolant.

xCp

Heat capacity (N.m/kgm/K or lbf.in/lbm/R) of the coolant.

xK

Thermal conductivity (N/s/K or lbf/s/R) of the coolant.

Record 2.7

Initial Guess for Operating Temperature of the Bearing Elements

This data record is always required.

Temperature of the bearing elements will change as a function of thermal interactions. The data supplied on this record is used as initial estimates.

All data on this record is dimensional. It is essential that the units conform to the unit code defined later on Record 3.2. The units given below in parenthesis correspond to the SI and English system of units, as discussed at the beginning of this chapter.

recID

Record identifier - maximum 12 characters in single quotes.

hsngTemp

Housing temperature (K or R).

shftTemp

Shaft temperature (K or R).

raceTemp1

Outer race temperature (K or R).

raceTemp2

Inner race temperature (K or R).

reTemp

Rolling element temperature (K or R).

cageTemp

Cage temperature (K or R).

In the case of a continuation run, which advances previously computed solutions, the data contained in records 3 to 11 are not required. Any data required by the user programmable subroutines must follow immediately after record 2.7. All the bearing data are read from the data file MASTER, and initial conditions vector from file FINAL. These files along with all the plot solution files (SOL1 to SOL6), as generated by the previous run, must be present in the current working directory.

3.3 Program Options

Record 3.1

Bearing Specification Code

This record is always required.

recID

Record identifier - maximum 12 characters in single quotes.

runId

Bearing specification code or run identifier - maximum 36 characters in single quotes.

This string is used to identify the run. This code is printed on each page on print and plot output. In addition the code is stored in each of the data set created by the run. In case of a continuation run, this code is matched in each of the data files before starting the run. It is, therefore, important to use a unique code with each run.

Record 3.2

Program Options Set #1

This record is always required.

recID

Record identifier - maximum 12 characters in single quotes.

kUnit

Code for system of units defined as follows:

1 SI units.

2 English units.

See discussion of units at the beginning of this chapter.

kBrg

Bearing type:

1 Ball bearing (angular contact or radial).

2 Cylindrical roller bearing.

3 Spherical roller bearing.

4 Tapered roller bearing.

5 Spherical tapered roller bearing.

ADORE can model basically any type of bearing with a restriction that there may be only one row of rolling elements. Thus the treatment of spherical roller bearing (kBrg =3), which normally contains two rows of rollers, is somewhat restricted. Spherical roller bearings may only a radially loaded single row.

nRe

Number of rolling elements.

Limited to 40 by the parameter statement (maxRe=40) in module Parameters. In the event the number of rolling elements is greater than 40, then this parameter statement must be appropriately modified. Like wise if a value of 40 is too high for the intended applications then the value may be appropriately reduces. This will result in a reduction in the required run-time random access memory (RAM).

nCseg

Number of cage segments.

The bearing cage may be segmented into equal sectors, as shown below in figure 9. The segmentation is defined by taking out a small angular sector out of the normal cylindrical cage. Segmentation details are input later on Record 7.0.1.

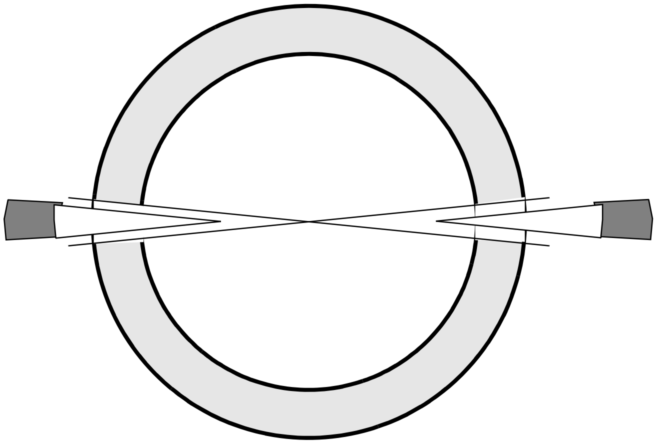


Figure 9. Exaggerated view of a two segment cage.

Maximum number of cage segments is limited to 3 by the parameter statement (maxCseg=3) in module Parameters. This statement may be appropriately modified if the number of cage segments is greater than 3.

For normal one piece cage, nCseg =1.

Also note that graphics animation is presently available only for a one piece cage.

kRaceFlex

Race flexibility switch for outer race:

0 rigid outer race.

1 flexible outer race.

This option is presently not available.

kReGeoImp

Code for geometrical imperfections in rolling elements:

0 ideal geometry.

1 imperfection on rolling element #1 only.

2 equal imperfections on all rolling elements.

3 imperfections are normally distributed.

4 imperfections are prescribed in subroutine Adrx8.

kRaceGeoImp1

Code for geometrical imperfections on outer race.

(presently not used).

kRaceGeoImp2

Code for geometrical imperfections on inner race.

(presently not used).

kFlngInd11

Existence of roller guide flange on the negative x-axis of the outer race:

0 No guide flange present.

1 Guide flange exists.

There could be a maximum four location for guide flanges on the races; two on the outer race and two on the inner race as shown below in Figure 10. The locations are references by positive and negative x-axis on the base coordinate frame.

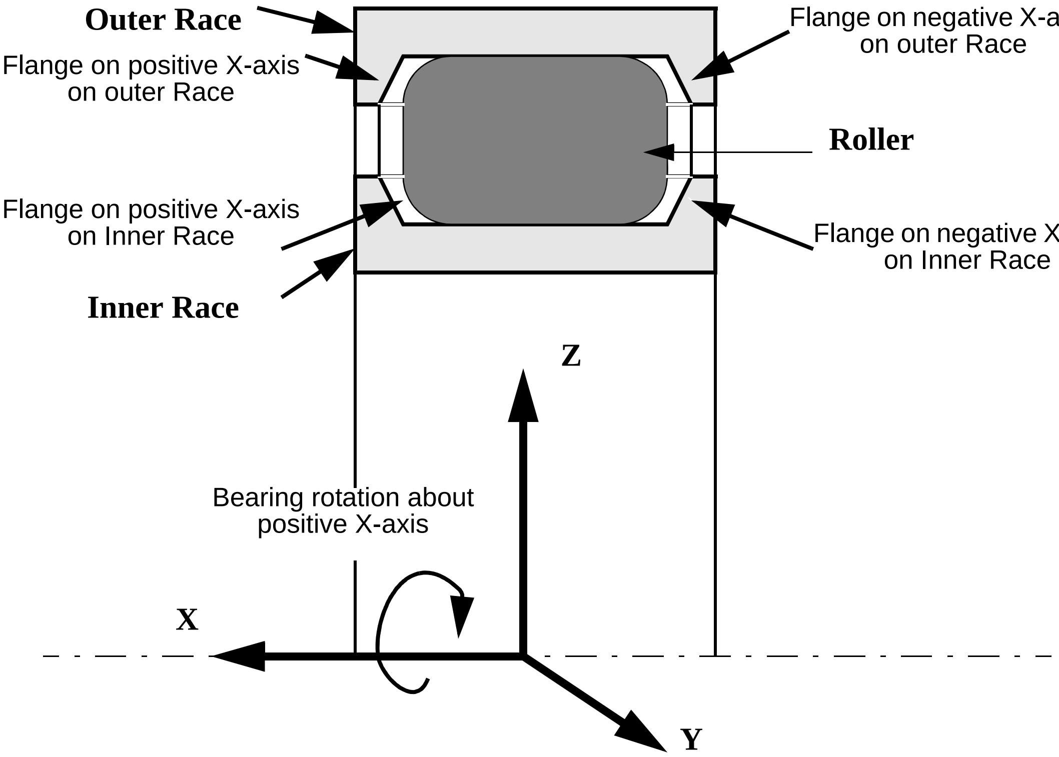

Outer Race
Flange on positive X-axis on outer Race
Flange on negative X-axis on outer Race
Roller
Flange on positive X-axis on Inner Race
Flange on negative X-axis on Inner Race
Inner Race
Z
Bearing rotation about positive X-axis
X
Y

Figure 10. Race guide flange definitions.

kFlngInd21

Existence of roller guide flange on the positive x-axis of the outer race:

0 No guide flange present.

1 Guide flange exists.

See discussion above under kFlngInd11.

kFlngInd12

Existence of roller guide flange on the negative x-axis of the inner race:

0 No guide flange present.

1 Guide flange exists.

See discussion above under kFlngInd11.

kFlngInd22

Existence of roller guide flange on the positive x-axis of the inner race:

0 No guide flange present.

1 Guide flange exists.

See discussion above under kFlngInd11.

Spherical bearing (kBrg =3) should be free of any race flanges.

kFlngInd11 = kFlngInd21 = kFlngInd12 = kFlngInd22 = 0.

Record 3.3

Program Options Set #2

This record is always required.

recID

Record identifier - maximum 12 characters in single quotes.

kFS1

Constraint along the x-axis for quasi-static solution:

0 prescribed force.

1 prescribed displacement.

ADORE accepts either force or displacement constraints along the X,Y,Z axes of the base coordinate system shown below in Figure 11. In other words either a load may be applied along a given axis or the races may be displaced by a given amount relative to each other. In the latter case the load generated by the imposed displacement is computed.Normally the thrust load is prescribed about the positive X-axis.

Such a flexibility is particularly useful in modeling a preloaded pair of angular contact ball bearing, where an initial run may be made with the prescribed preload at room temperature and the resulting axial displacement of the outer race relative to the inner may be noted. In subsequent runs, when the radial load, operating speed and temperature fields may be applied, the bearing may be constrained to an axial displacement noted in the initial run. Now the resulting value of thrust load may be noted and compared to the applied initial preload.This may give an insight into affect of applied operating conditions on actual preload and possibly diagnose skid problems.

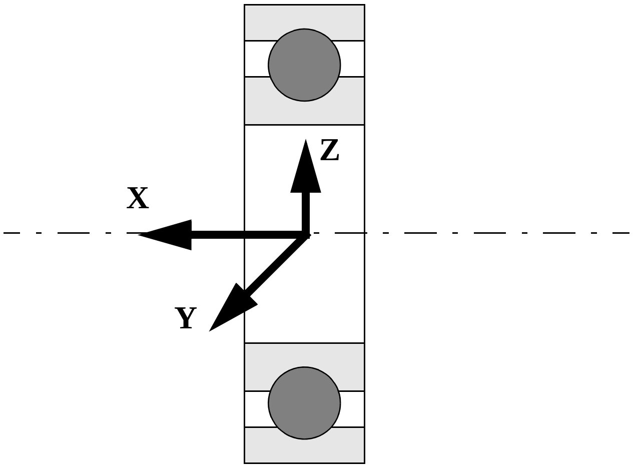


Figure 11. Bearing base coordinate system.

kFS2

Constraint along the y-axis for quasi-static solution:

0 prescribed force.

1 prescribed displacement.

See discussion above under kFS1.

kFS3

Constraint along the z-axis for quasi-static solution:

0 prescribed force.

1 prescribed displacement.

See discussion above under kFS1.

kFS4

Moment constraint along y axis for quasi-static solution:

0 prescribed moment.

1 prescribed misalignment.

Similar to the applied forces either moments may be prescribed about the transverse Y and Z axes or the bearing may be subjected to relative misalignment about these two axes as shown below. If the misalignment are prescribed then the computed moments are in the output. Likewise, when moments are prescribed the computed angular displacement of the race, or relative misalignment, is in the output.

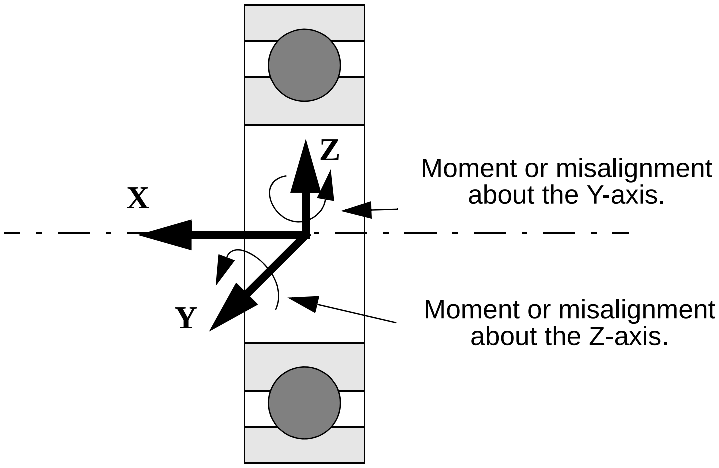

Z
X
Y
Moment or misalignment about the Y-axis.
Moment or misalignment about the Z-axis.

Figure 12. Schematic of applied moments of misalignments along the base coordinates.

Note that in angular contact ball bearings when a combined thrust (along X-axis) and radial (along Y-axis) loads are applied, the internal load distribution results in a moment about the transverse Y-axis. When the races are constrained to have zero misalignment this moment will be seen in the output. In the event a moment equilibrium is desired under such a condition, then the above constraint may be set to zero and also the value of applied moment, prescribed later on Record 9.1, may be set to zero. This will turn on moment equilibrium under zero external moment. Thus the misalignment generated by the internal moment, due to a combined thrust and radial load, will be computed.

kFS5

Moment constraint along z axis for quasi-static solution:

0 prescribed moment.

1 prescribed misalignment.

See discussion above under kFS4.

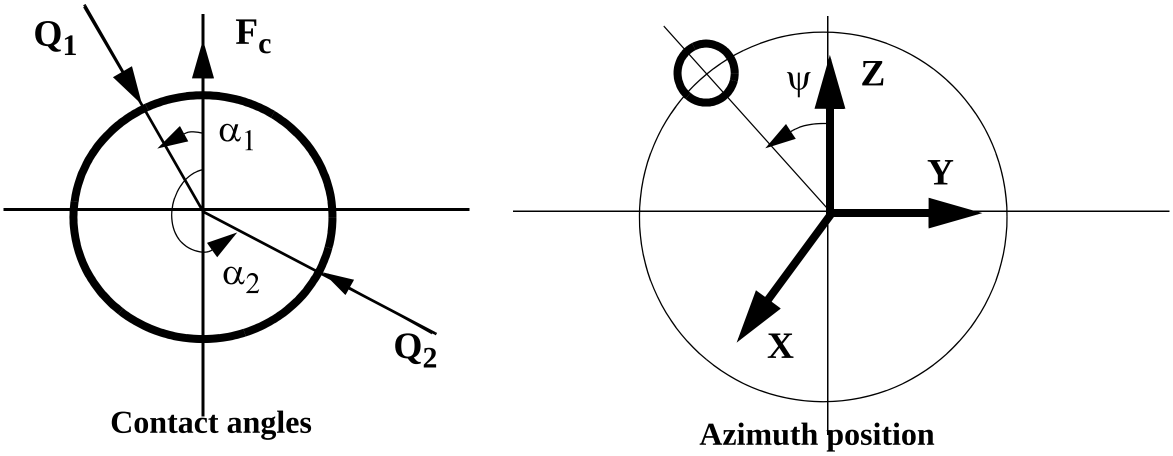
kAngVel

Quasi-static angular velocity constraint for ball bearings:

0 Compute angular velocities by minimizing heat generation in the contacts.

1 Use race control hypothesis.

When performing a static equilibrium the relative axial and radial position of the rolling elements may be computed by the axial and radial equations of equilibrium. Similarly the the relative position coordinates of one race relative to the other (X,Y,Z) may be computed by the three force equilibrium equations for the race. These relative positions will define the contact angles for ball bearings as shown below. For roller bearings the contact angles are already known and it is only necessary to compute roller position relative to the races.



Contact angles

Azimuth position

Figure 13. Contact angles and azimuth position for ball bearings.

Computation of angular and orbital velocity of the rolling elements can be computed by imposing a pure rolling constraint on the outer and inner races. These two constraints are adequate when the orientation of the angular velocity is known, such as for roller bearings, and the two unknowns are magnitude of angular and orbital velocities. For angular contact ball bearings however, the ball angular velocity vector is tilted and has two components about the X and Z axes as shown below. Therefore, there are three unknowns, two components of angular velocity vector and the magnitude of ball orbital velocity about the bearing axis. Thus in addition to the rolling constraints at the outer and inner races, an additional constraint is required to complete the analytical formulation for computation of angular velocities. To satisfy this additional requirement friction moments, under constant coefficient of friction, are computed on the outer and inner race contacts about the axes normal to the respective planes of contacts defined by the contact angles as shown below in figure 14. Now it is postulated that the ball angular velocity vector will be oriented such that the ball angular velocity component relative to the race, about an axis normal to the plane of contact is zero of the raceway which provides a larger friction moment. In other words, the relative “spin” on the raceway with higher friction moment is zero. Such a hypothesis is commonly known as “outer race control” or “inner race control”, corresponding to zero friction moment on outer or inner races respectively. The above constraint is applied to compute the ball angular velocities when kAngVel is set to 1.

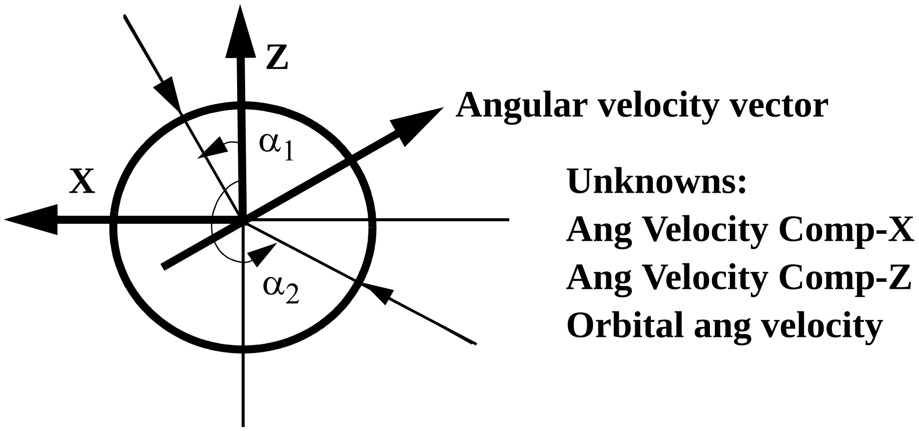


Figure 14. Angular velocity vector in its components in a ball bearing.

An alternate constraint on the orientation of the ball angular vector may be determined from energy considerations. For a given ball/race traction model the heat generated in the outer and inner race contacts may be computed as a function of the inclination of the ball angular velocity vector. A variation of the type shown in figure 15 is be observed.

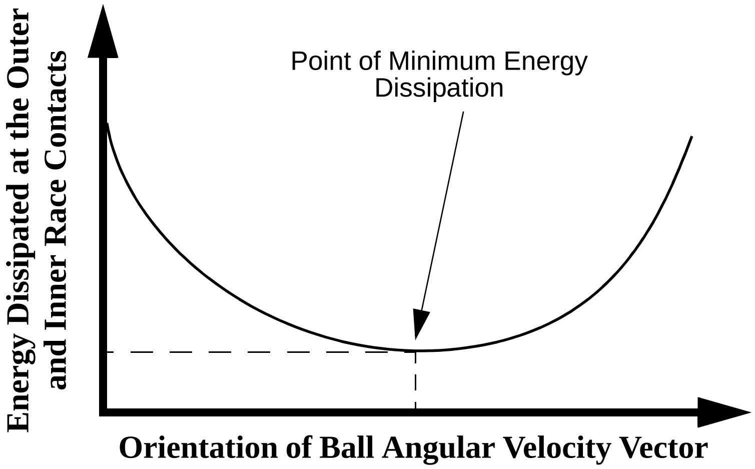


Figure 15. Schematic of energy dissipation as a function of ball angular velocity vector orientation.

It is now postulated that the ball angular velocity vector will orient itself such that the total energy dissipated in the outer and inner race contact is a minimum. Such a constraint is imposed by setting kAngVel = 0.

The above constraint is, of course, irrelevant for all roller bearings and also for ball bearings with a pure radial load.

kReEqCode

Normally ADORE uses the classical Newton-Raphson iterative procedure for solving the equilibrium equations. Under certain conditions, particularly with roller bearings subjected to misalignment, the equations may not convergence. Under such conditions this option obtain a solution by minimizing the root squared value of the residual load vector. The available options are:

0 Default value. Obtain solutions by classical Newton-Raphson procedures.

1 In addition to Newton-Raphson procedures, minimize the root mean squared deviation of the residual load vector when the Newton-Raphson iteractions do not converge.

kStif

Number of points in the stiffness speed table. Specify zero if no stiffness computation is desired.

Since the contact loads depend on operating speed, due to centrifugal effects, and stiffness is load dependent, the operating speed will have an effect of bearing stiffness. Such a variation is generally useful for rotor dynamics modeling where critical rotor speed and overall rotor response is computed. By setting kStif equal to a number greater than 0, ADORE will perform a quasi-static analysis to compute a bearing stiffness-speed table. There will be kStif points in the table and the initial and final speeds are defined later on Record 9.2.

kChrn

Churning code:

0 Neglect churning.

1 Include churning with lubricant properties derived from the lubricant model specified by parameter kTrac on Record 10.0.

2 Include churning with specified lubricant properties.

3 Include churning with liquid oxygen as churning media.

4 Include churning with liquid hydrogen as churning media.

5 Include churning with liquid nitrogen as churning media.

6 Include churning with air (at atmospheric pressure) as churning media.

7 Include churning with water as churning media.

kReMat

Material code for the rolling elements:

0 Default material (AISI 52100 bearing steel).

1 Material with properties specified on Record 8.1.

2 Material properties to be extracted from user data base via user subroutine ADRX0.

m Material code for property data base in ADORE. See available material codes below.

kRaceMat1

Material code for outer race:

0 Default material (AISI 52100 bearing steel).

1 Race material properties specified on record 8.2.1.

2 Material properties to be extracted from user data base via user subroutine ADRX0.

m Material code for property data base in ADORE. See available material codes below.

kRaceMat2

Material code for inner race:

0 Standard material (AISI 52100 bearing steel).

1 Race material properties specified on record 8.2.2.

2 Material properties to be extracted from user data base via user subroutine ADRX0.

m Material code for property data base in ADORE. See available material codes below.

kHsngMat

Material code for the housing:

0 Default material (Mild steel).

1 Material properties specified on records 8.3.

2 Material properties to be extracted from user data base via user subroutine ADRX0.

m Material code for property data base in ADORE. See available material codes below.

kShftMat

Material code for shaft:

0 Default material (Mild steel).

1 Material properties specified on records 8.4.

2 Material properties to be extracted from user data base via user subroutine ADRX0.

m Material code for property data base in ADORE. See available material codes below.

kLifeCons

Material constants for basic fatigue life computation:

0 Default constants.

1 Required constants specified on record 8.6.0

kLifeMod

STLE fatigue life modification code:

0 No life modifying factors

1 AISI 52100 steel

2 AISI M-1.

3 AISI M-2.

4 AISI M-10

5 AISI M-42

6 AISI M-50

7 AISI T-1 (18-4-1)

8	Halmo
9	WB-49
21	AMS 5749 (BG-42)
22	AMS 5900 (CRB-7)
23	AISI 440C
41	AMS 6278 (M-50 NIL)
42	AISI 4720
43	AISI 8620
44	AISI 9310
45	CBS 600
46	CBS 1000
47	Vasco X-2
99	Constants for computing life modifying factors prescribed on records 8.6.3 and 8.6.4.

For Tallian model only available codes are 1, 6 and 43. All others default to code 1.

kProc

Materials processing code, applicable when kPLifeMod has any valid value defined above:

1	Air Melt (AM)
2	Carbon Vacuum Deoxidation (CVD)
3	Vacuum Processing (VP)
4	Vacuum Arc Remelting (VAR)
5	ElectroFlux Remelting (EFR)
6	Double Vacuum Arc Remelting (VAR-VAR)
7	Vacuum Induction Remelting - Vacuum Arc Remelting (VIM-VAR)

For Tallian model available codes are 2 and 7, all others default to 2.

Presently available material codes (m) in ADORE database are:

m	Material
100	AISI 52100 Bearing Steel
101	M50 Bearing Steel
102	M50 VIM-VAR Bearing Steel
103	440C Stainless Steel
104	430 Ferratic Stainless Steel
105	410 Martenitic Stainless Steel
106	304 Austenitic Stainless Steel
107	AMS 5898 Cronidur 30 Stainless Steel
108	AMS 5643 (17-4PH) Stainless Steel
110	C1045 Steel
111	AISI 4340 Steel
112	Inconel 625 Alloy

113	Inconel 718 Alloy
114	AISI 304HN High Nitrogen Steel
120	M-50 Nil (Case hardened steel)
121	P-675 HTT (Case hardened steel)
122	P-675 LTT (Case hardened steel)
150	Si3N4 Silicon Nitride
151	Zirconium Oxide (ZrO20
160	Copper
161	Brass
162	Bronze
200	Bearing Grade Peek
201	Polyamide-Nylon
202	Armalon
203	Carbon Phenolic
204	Carbon Phenolic (10% MoS2)
205	Cotton Phenolic
206	Graphite
207	Teflon (PTFE)

Record 3.4

Program Options Set #3

This record is required only for dynamic simulations, mode > 0 on Record 1

recID

Record identifier - maximum 12 characters in single quotes.

kRotLoad

Dynamic simulation with a rotating radial load. Rotating load is simulated be letting the race ass center whirl in a circular orbit with a radius equal to fraction of the initial relative radial deflection of the races:

0	No race orbits.
1	Prescribed race orbits.

Pertinent details of race orbits are specified later on Record 9.3.

kRotFrame

Normally the bearing center is fixed in space at origin of a space fixed (inertial) coordinate frame and motion of all bearing elements are modeled relative to this inertial frame. When the entire bearing moves in space, additional transport and Corioliss terms must be applied in the equations of motion. This option permits such a simulation. The code for moving reference frame is specified as:

0 Base bearing frame is fixed in space.

1 Base frame travels in space.

Common applications with moving base coordinate frame where the bearing as a whole travels in space, include bearings used in planetary gear assemblies, or crankshafts of reciprocating engines. Such a motion with a constant angular velocity can be simply modeled with data supplied on Record 9.4. For more complex conditions it will be necessary to use the optional user subroutine Adrx1.

kReIP

Code for inertial parameters of the rolling elements:

0 Standard parameters (ideal geometry).

1 Inertial parameters for rolling element #1 prescribed on Record 6.1.

2 Use the values prescribed on Record 6.1 for all rolling elements.

3 Inertial parameters defined in subroutine Adrx8.

kRaceIP

Vector of length 2 containing the inertial parameters option for the races:

0 Standard parameters.

1 Inertial parameters specified on Records 6.2.k.

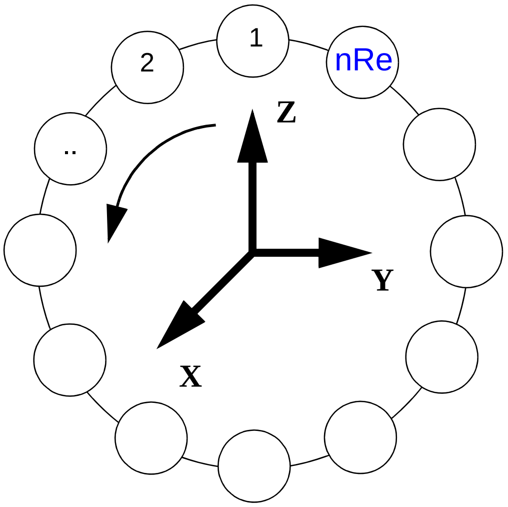
kNumPltElem

Number of elements (maximum 6) for which the plot output will be saved.

kPltElemInd

Vector of length kNumPltElem containing the indices of the elements in increasing order.

Bearing elements are numbered sequentially as shown below in figure 16. The indices 1 to nRe (see Record 3.2) correspond to the nRe rolling elements; (nRe +1) to (nRe + nCseg) correspond to the nCseg cage segments: and, (nRe + nCseg +1) and (nRe + nCseg +2) respectively corresponds to the outer and inner races.



Rolling Eements
1 to nRe

Cage Segments
(nRe + 1) to (nRe + nCseg)

Outer Race
(nRe + nCseg +1)

Inner Race
(nRe + nCseg +2)

Figure 16. Numbering sequence for the bearing elements.

3.4 Bearing Envelope

Record 4

Bearing Envelope

This record is always required.

Data on this record specifies the bearing envelope as shown in figure 17.

All data on this record is dimensional. It is essential that the units conform to the unit code defined later on Record 3.2. The units given below in parenthesis correspond to the SI and English system of units, as discussed at the beginning of this chapter.

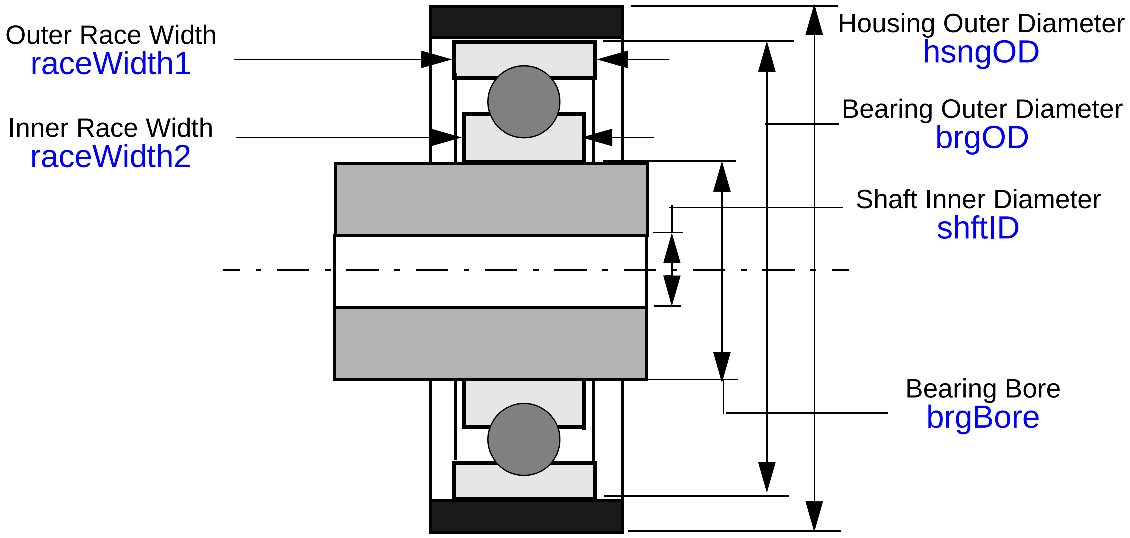


Figure 17. Definition of bearing envelope.

recID

Record identifier - maximum 12 characters in single quotes.

brgBore

Bearing bore (m or in), see figure 17 above.

brgOD

Outside diameter of bearing (m or in), see figure 17 above.

shftID

Shaft inside diameter (m or in) for a hollow shaft, see figure 17 above.

hsngOD

Housing outside diameter (m or in), see figure 17 above.

raceWidth1

Outer race width (m or in), see figure 17 above.

raceWidth2

Inner race width (m or in), see figure 17 above.

3.5 Rolling Element and Race Geometry

Record 5A

Ball Bearing Geometry

This data record is required only for ball bearings, kBrg = 1 on Record 3.2.

Some of the data on this record is dimensional. It is essential that the units conform to the unit code defined later on Record 3.2. The units given below in parenthesis correspond to the SI and English system of units, as discussed at the beginning of this chapter.

recID

Record identifier - maximum 12 characters in single quotes.

bReDia

Nominal ball diameter (m or in).

pitchDia

Pitch diameter (m or in).

freeConAng

Free contact angle (deg).

If this value is zero then the internal clearance given below is used to calculate the free contact angle

freeIntCls

Free internal clearance in the bearing (m or in).

raceCurFac1

Outer race curvature factor.

Race curvature factor is defined as the ratio of the radius of curvature of the race groove to the nominal ball diameter, bReDia.

raceCurFac2

Inner race curvature factor.

Defined same as the one for outer race above.

shoulderDia1

Diameter of outer race shoulder (m or in).

See figure 18 below. This variable is only used to check extent of contact on the inner race. Distance of the inner edge of the contact zone from the inner race shoulder is included in the print output.

shoulderDia2

Diameter of inner race shoulder (m or in).

See figure 18 below. This variable is only used to check extent of contact on the inner race. Distance of the inner edge of the contact zone from the inner race shoulder is included in the print output.

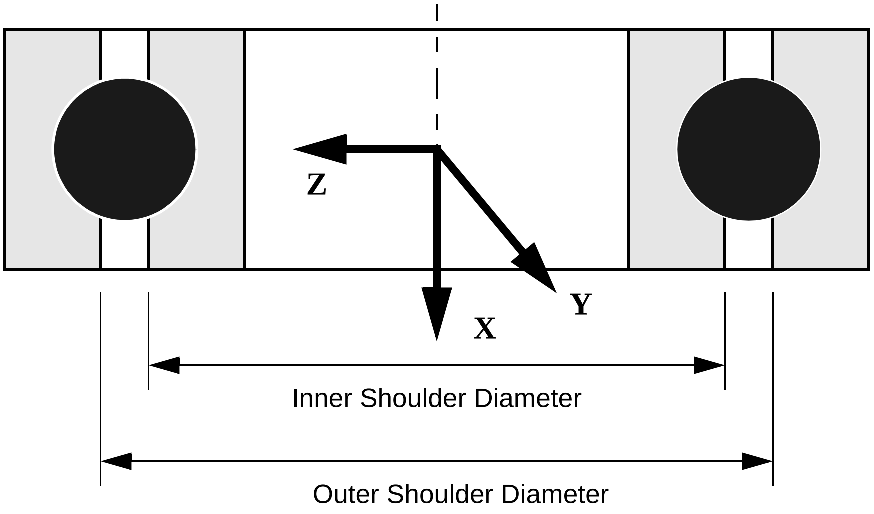


Figure 18. Definition of race shoulder diameters.

shimThickness1

Shim thickness (m or in) for split outer race.

In some bearing applications the races may be split in two parts, see figure 19 below. Then a shim of a given thickness is placed between the two parts of the races before the groove is ground. Upon assembly the shim is taken out, thus creating an arched configuration, where the ball can actually contact both parts, or arches of the race. The thickness of the shim used will affect the actual internal clearance and free contact angle.

Although ADORE does not model dual contacts on a race, this variable is used to make appropriate adjustment to bearing internal clearances is made and based on contact angle the possibility of dual contact is indicated in the print output. In addition the position of the inner edge of contact in relation to the central race split is also included in the print output.

shimThickness2

Shim thickness (m or in) for split inner race.

Although ADORE does not model dual contacts on a race, this variable is used to make appropriate adjustment to bearing internal clearances is made and based on contact angle the possibility of dual contact is indicated in the print output.

rmsAspHt1

Composite surface roughness (m or in) at outer race contact.

rmsAspHt2

Composite surface roughness (m or in) at inner race contact.

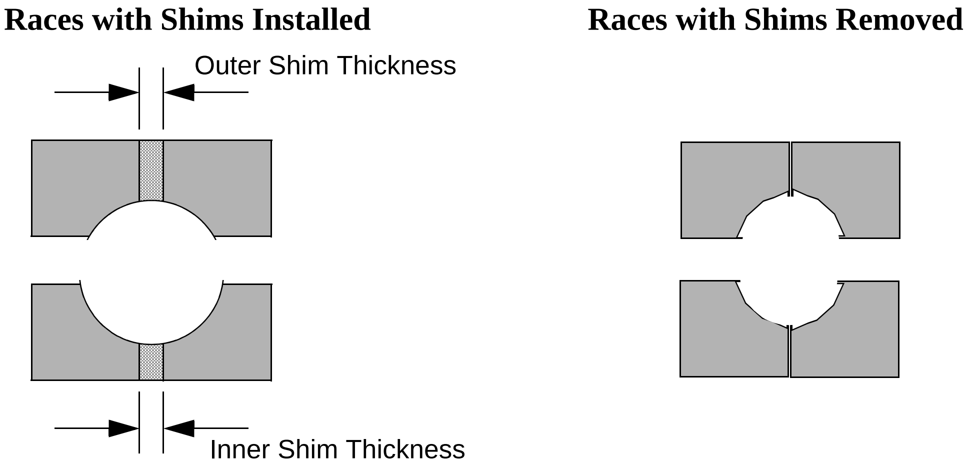


Figure 19. Geometry of split races.

Record 5B.1

Cylindrical Roller Bearing Geometry

This record is required for cylindrical roller bearings, kBrg = 2 on Record 3.2.

All the data on this record is dimensional. It is essential that the units conform to the unit code defined later on Record 3.2. The units given below in parenthesis correspond to the SI and English system of units, as discussed at the beginning of this chapter.

See figure 20 below for geometrical description of the various variables

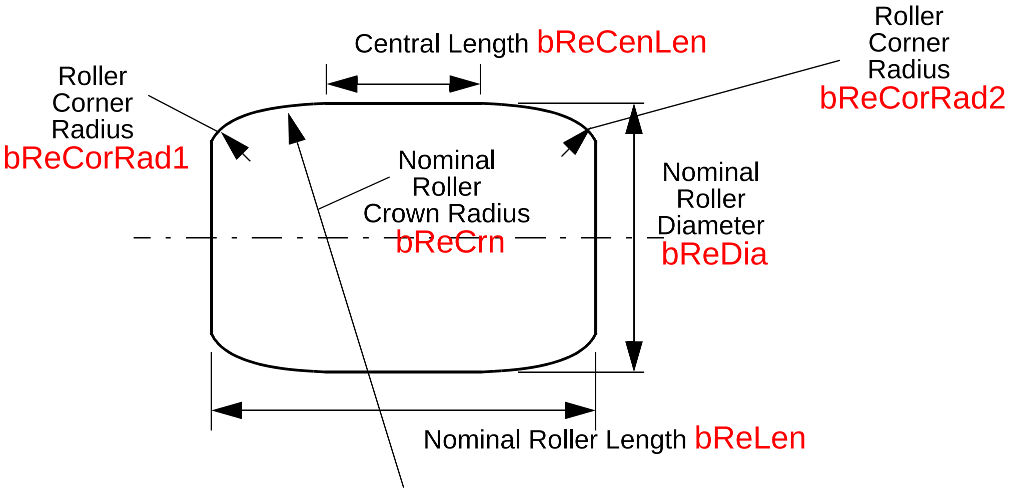


Figure 20. Geometrical parameters of a roller.

recID

Record identifier - maximum 12 characters in single quotes.

bReDia

Nominal roller diameter (m or in).

bReCrn

Nominal crown radius (m or in).

For infinite radius, specify 1.0e+10 and set bReLen = bReCenLen on this record.

bReLen

Nominal roller length (m or in).

bReCenLen

Nominal length of central land (m or in).

bReCorRad1

Nominal corner radius on the negative x-axis of roller (m or in).

bReCorRad2

Nominal corner radius on the positive x-axis of roller (m or in).

pitchDia

Pitch diameter (m or in).

freeIntCls

Free internal clearance or diametral play (m or in).

Record 5B.2

Cylindrical Roller Bearing Geometry - continued

This record is required for cylindrical roller bearings, kBrg = 2 on Record 3.2.

All the data on this record is dimensional. It is essential that the units conform to the unit code defined later on Record 3.2. The units given below in parenthesis correspond to the SI and English system of units, as discussed at the beginning of this chapter.

recID

Record identifier - maximum 12 characters in single quotes.

raceLandLmt1

Effective surface width (m or in) on the outer race, defined as a dimension of race surface along the roller length. Normally this dimension will be equal to the total race surface width minus any undercuts at the guide flange origins.

raceCenLen1

Central land width (m or in) on the outer race in case of partly crowned raceway.

This variable must be presently set to zero. It is reserved for future use.

raceCrn1

Outer race crown radius (m or in). This variable is for future use only. Presently it may be set to zero.

raceLandLmt2

Effective surface width (m or in) on the inner race, similar to the definition described above for raceLandLmt1.

raceCenLen2

Central land width (m or in) on the inner race in case of partly crowned raceway.

This variable must be presently set to zero. it is reserved for future use.

raceCrn2

Inner race crown radius (m or in). This variable is for future use only. Presently it may be set to zero.

rmsAspHt1

Composite surface roughness (m or in) at outer race contact.

rmsAspHt2

Composite surface roughness (m or in) at inner race contact.

Record 5C

Spherical Roller Bearing Geometry

This record is required for spherical roller bearings, kBrg = 3 on Record 3.2.

Some of the data on this record is dimensional. It is essential that the units conform to the unit code defined later on Record 3.2. The units given below in parenthesis correspond to the SI and English system of units, as discussed at the beginning of this chapter.

recID

Record identifier - maximum 12 characters in single quotes.

bReDia

Nominal roller diameter (m or in).

bReCrn

Nominal crown radius (m or in).

bReLen

Nominal roller length (m or in).

bReCorRad1

Nominal corner radius on the negative x-axis on roller (m or in).

bReCorRad2

Nominal corner radius on the positive x-axis of the roller (m or in).

pitchDia

Pitch diameter (m or in).

freeIntCls

Diametral clearance or play (m or in).

raceCurFac1

Outer race curvature factor.

Race curvature factor is defined as the ration of radius of curvature of the race groove to the nominal crown diameter of the roller (2*bReCrn). The definition is similar to the one used for ball bearings.

raceCurFac2

Inner race curvature factor.

Race curvature factor is defined as the ration of radius of curvature of the race groove to the nominal crown diameter of the roller (2*bReCrn). The definition is similar to the one used for ball bearings.

conAng

Tilt of the inner race surface with respect to the shaft axis (deg).

rmsAspHt1

Composite surface roughness (m or in) at outer race contact.

rmsAspHt2

Composite surface roughness (m or in) at inner race contact.

Record 5D.1

Tapered Roller Bearing Geometry

This record is required for tapered roller bearings, kBrg = 4 on Record 3.2.

All the data on this record is dimensional. It is essential that the units conform to the unit code defined later on Record 3.2. The units given below in parenthesis correspond to the SI and English system of units, as discussed at the beginning of this chapter.

recID

Record identifier - maximum 12 characters in single quotes.

bReDia

Nominal roller diameter (m or in) at the large end.

bReCrn

Nominal crown radius (m or in).

bReLen

Nominal roller length (m or in).

bReCenLen

Nominal length of central land (m or in).

bReEndRad1

Nominal end radius at the large end of the roller (m or in).

bReEndRad2

Nominal end radius at the small end of the roller (m or in).

bReCorRad1

Nominal corner radius on the negative x-axis of roller (m or in).

Negative x-axis points towards the large end of roller.

bReCorRad2

Nominal corner radius on the positive x-axis of roller (m or in).

Positive x-axis points toward the small end of roller.

raceTaper1

Outer race semi cone angle (deg).

raceTaper2

Inner race semi cone angle (deg).

Record 5D.2

Tapered Roller Bearing Geometry - continued

This record is required for cylindrical roller bearings, kBrg = 4 on Record 3.2.

All the data on this record is dimensional. It is essential that the units conform to the unit code defined later on Record 3.2. The units given below in parenthesis correspond to the SI and English system of units, as discussed at the beginning of this chapter.

recID

Record identifier - maximum 12 characters in single quotes.

raceLandLmt1

Effective surface width (m or in) on the outer race, defined as a dimension of race surface along the roller length. normally this dimension will be equal to the total race surface width minus any undercuts at the guide flange origins.

raceCenLen1

Central land width (m or in) on the outer race in case of partly crowned raceway.

This variable must be presently set to zero, it is reserved for future use.

raceCrn1

Outer race crown radius (m or in). This variable is for future use only. Presently it may be set to zero.

raceLandLmt2

Effective surface width (m or in) on the inner race, similar to the definition described above for inner race.

raceCenLen2

Central land width (m or in) on the inner race in case of partly crowned raceway.

This variable must be presently set to zero. it is reserved for future use.

raceCrn2

Inner race crown radius (m or in). This variable is for future use only. Presently it may be set to zero.

rmsAspHt1

Composite surface roughness (m or in) at outer race contact.

rmsAspHt2

Composite surface roughness (m or in) at inner race contact.

Record 5E

Spherical Tapered Roller Bearing Geometry

This record is required for cylindrical roller bearings, kBrg = 5 on Record 3.2.

Some the data on this record is dimensional. It is essential that the units conform to the unit code defined later on Record 3.2. The units given below in parenthesis correspond to the SI and English system of units, as discussed at the beginning of this chapter.

recID

Record identifier - maximum 12 characters in single quotes.

bReDia

Nominal roller diameter (m or in) at the large end.

bReCrn

Nominal crown radius (m or in).

bReLen

Nominal roller length (m or in).

bReEndRad1

Nominal end radius at large end of the roller (m or in)

bReEndRad2

Nominal end radius at small end of the roller (m or in)

raceTaper1

Outer race semi cone angle (deg).

raceTaper2

Inner race semi cone angle (deg).

raceCurFac1

Outer race curvature factor.

Race curvature factor is defined as the ration of radius of curvature of the race groove to the nominal crown diameter of the roller (2*bReCrn). The definition is similar to the one used for ball bearings.

raceCurFac2

Inner race curvature factor.

Race curvature factor is defined as the ration of radius of curvature of the race groove to the nominal crown diameter of the roller (2*bReCrn). The definition is similar to the one used for ball bearings.

rmsAspHt1

Composite surface roughness (m or in) at outer race contact.

rmsAspHt2

Composite surface roughness (m or in) at inner race contact.

Record 5F

Race Flange Geometry

This record is required when the races have guide flanges, kFlngIndxx > 0 on Record 3.2, which is normally the case for cylindrical and tapered roller bearings.

All the data on this record is dimensional. It is essential that the units conform to the unit code defined on Record 3.2. The units given below in parenthesis correspond to the SI and English system of units, as discussed at the beginning of this chapter.

Figure 21 below described the various geometrical variables on this record

.

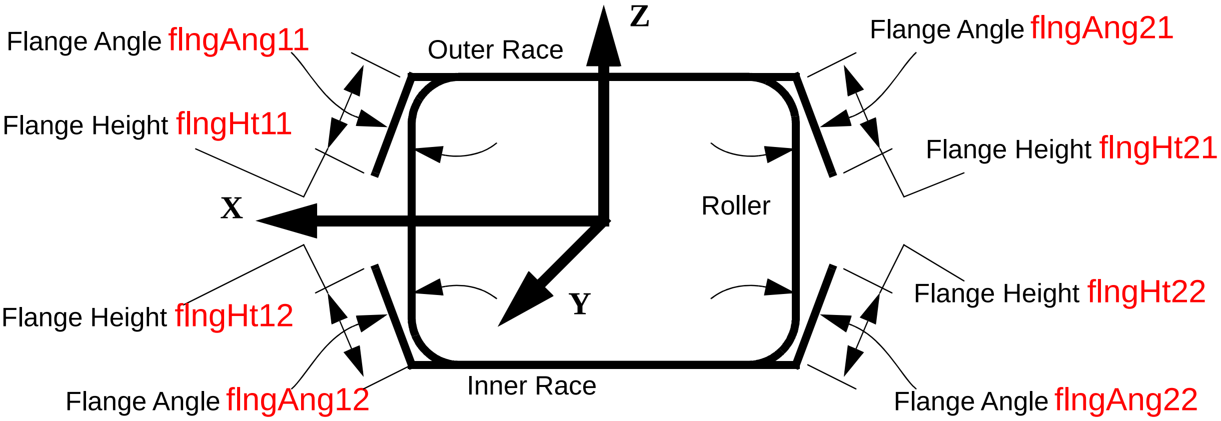


Figure 21. Race guide flange definitions.

recID

Record identifier - maximum 12 characters in single quotes.

flngAng11

Flange layback angle (deg), outer race, negative x-axis.

This value is only applicable when kFlngInd11 = 1 on Record 3.2.

flngAng21

Flange layback angle (deg), outer race, positive x-axis.

This value is only applicable when kFlngInd21 = 1 on Record 3.2.

flngAng12

Flange layback angle (deg), inner race, negative x-axis.

This value is only applicable when kFlngInd12 = 1 on Record 3.2.

flngAng22

Flange layback angle (deg), inner race, positive x-axis.

This value is only applicable when kFlngInd22 = 1 on Record 3.2.

flngHt11

Flange height (m or in), outer race, negative x-axis.

This value is only applicable when kFlngInd11 = 1 on Record 3.2.

flngHt21

Flange height (m or in), outer race, positive x-axis.

This value is only applicable when kFlngInd21 = 1 on Record 3.2.

flngHt12

Flange height (m or in), inner race, negative x-axis.

This value is only applicable when kFlngInd12 = 1 on Record 3.2.

flngHt22

Flange height (m or in), inner race, positive x-axis.

This value is only applicable when kFlngInd22 = 1 on Record 3.2.

flngCls1

Roller/flange axial clearance (m or in), outer race.

Roller/flange axial clearance is equal to the free axial travel of the roller between the guide flanges.

This value is only applicable when both kFlngInd11 and kFlngInd21 = 1 on Record 3.2.

flngCls2

Roller/flange axial clearance (m or in), inner race.

Roller/flange axial clearance is equal to the free axial travel of the roller between the guide flanges.

This value is only applicable when both kFlngInd12 and kFlngInd22 = 1 on Record 3.2.

Record 5G.1A

Geometric Imperfections in Ball Bearings

This record is required only when modeling geometric imperfections on balls in a ball bearing, 1 <= kReGeoImp <= 3 and kBrg = 1 Record 3.2

Geometrical imperfections on the balls are the restricted to variations in ball diameter. The overall shape of the balls is still assumed to be spherical.

All the data on this record is dimensional. It is essential that the units conform to the unit code defined later on Record 3.2. The units given below in parenthesis correspond to the SI and English system of units, as discussed at the beginning of this chapter.

recID

Record identifier - maximum 12 characters in single quotes.

reDiaVar

Variation in ball diameter (m or in).

Actual variation in ball diameter when kReGeoImp =1 or 2, or rms deviation when kReGeoImp =3 on Record 3.2. In the later case, the nominal diameter specified on Record 5A is assumed to be mean value.

Record 5G.1B

Geometrical imperfections on Rollers in a Roller Bearing

This record is required to model geometric imperfections on rollers in a roller bearing, 1 <= kReGeoImp <= 3 and kBrg > 1 Record 3.2

1<=kReGeoImp<=3 and kBrg > 1 on Rec 3.2

This record contains geometrical imperfections on rollers in a roller bearing.

Similar to rec 5G.1A, the data entered on this record depends on value of kReGeoImp. For kReGeoImp =1 or 2, the values specified below are actual variations, while for kReGeoImp =3, this record contains rms deviations and the actual variations on each rolling element are computed from a normal distribution. The nominal values prescribed on rec 5B to 5D are assumed to represent the mean values.

Orientation of the roller end face is prescribed by three transformation angles which define a coordinate frame, contained in the roller end face, relative to the roller coordinate frame.

All the data on this record is dimensional. It is essential that the units conform to the unit code defined later on Record 3.2. The units given below in parenthesis correspond to the SI and English system of units, as discussed at the beginning of this chapter.

recID

Record identifier - maximum 12 characters in single quotes.

reDiaVar

Variation in roller diameter (m or in).

See general discussion below the title of this record.

reCrnVar

Variation in crown radius (m or in).

See general discussion below the title of this record.

reLenVar

Variation in roller length (m or in).

See general discussion below the title of this record.

reCLVar

Variation in central land (m or in).

See general discussion below the title of this record.

reCLOffset

Axial offset of central land on roller (m or in).

See general discussion below the title of this record.

reEndFrame11

First transformation angle (deg) for roller end on negative x-axis.

Orientation of the end faces of the roller may be prescribed by three transformation angles relative to the base roller coordinate frame, as shown in figure 22. The three transformation angles will define the transformation from the base roller coordinate frame (X,Y,Z) to the end face coordinate frame (x,y,z). The end face coordinate axis x is normal to the end face, while the axes (y and z) lie in the plane of the end face.

.

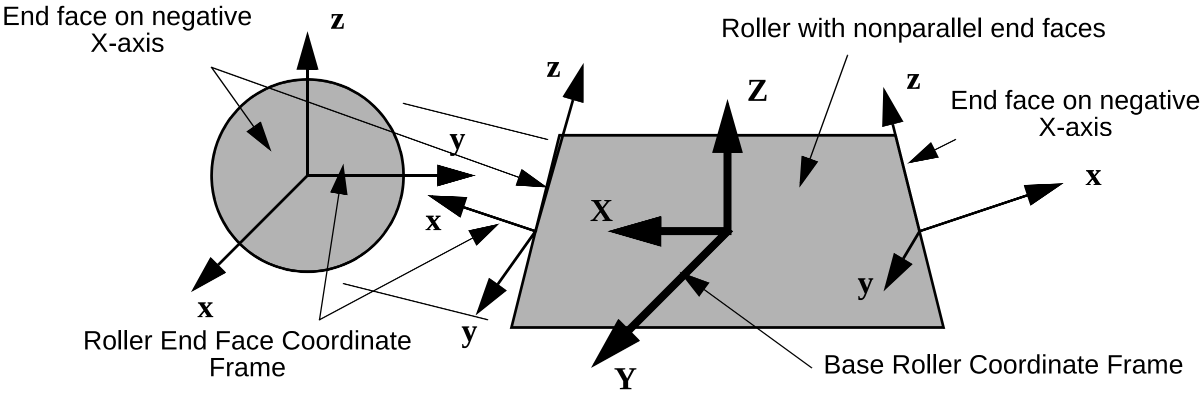

End face on negative X-axis
z
z
Roller with nonparallel end faces
Z
z
End face on negative X-axis
y
x
x
X
y
x
y
Roller End Face Coordinate Frame
Y
Base Roller Coordinate Frame

Figure 22. Geometrical definition of roller end faces.

reEndFrame21

Second transformation angle (deg) for roller end on negative x-axis.

See discussion above under variable reEndFrame11.

reEndFrame31

Third transformation angle (deg) for roller end on negative x-axis.

See discussion above under variable reEndFrame11.

reEndFrame12

First transformation angle (deg) for roller end on positive x-axis.

See discussion above under variable reEndFrame11.

reEndFrame22

Second transformation angle (deg) for roller end on positive x-axis.

See discussion above under variable reEndFrame11.

reEndFrame32

Third transformation angle (deg) for roller end on positive x-axis.

See discussion above under variable reEndFrame11.

Record 5G.2.1A

Geometrical Imperfections on Outer Race for Ball, Spherical and Spherical Tapered Roller Bearings

This record is required only when geometric imperfection are to be prescribed on the outer race for ball, spherical and spherical tapered roller bearings:

kRaceGeoImp1 > 0 and kBrg = 1, 3 or 5 on Record 3.2

For ball, spherical and tapered spherical roller bearings there may be two imperfections on the race: out-of-roundness and variation in race groove curvature. With the imperfection code, kRaceGeoImp1=1, race out-of-roundness is modeled by an elliptical profile, where the semi major and minor axes of the ellipse are defined as:

Semi-major axis = (r + a)

Semi-minor axis = (r + b)

where r is the nominal radius and the two parameters a and b define the radius variation.

With the imperfection code, kRaceGeoImp1=2, the race radius variation is prescribed by a sinusoidal variation around the race. The the magnitude of imperfection, *a*, is defined by amplitude, a_o, frequency ω and phase shift φ:

$$a = a_o \sin(\omega\theta + \varphi)$$

where θ is the angular position relative to the body-fixed z-axis, measured as a rotation about the body-fixed x-axis, which is also the bearing axis, as shown below in figure 23.

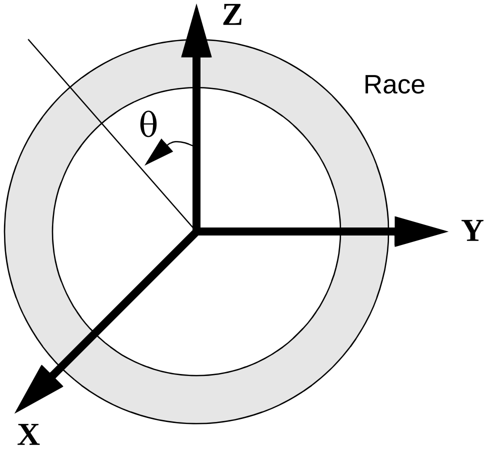


Figure 23. Angular coordinate in a race-fixed coordinate frame.

Thus three values, corresponding to amplitude a_o,frequency ω, and phase shift φ, define any geometric imperfection on the race.

The variation in race groove curvature is always prescribed in terms of a sinusoidal function discussed above.

Some of the data on this record is dimensional. It is essential that the units conform to the unit code defined later on Record 3.2. The units given below in parenthesis correspond to the SI and English system of units, as discussed at the beginning of this chapter.

recID

Record identifier - maximum 12 characters in single quotes.

rndVar11

For kRaceGeoImp1=1:

Deviation (m or in) of the semi-major axis of the elliptical race profile from the nominal race radius:

Semi-major axis = (nominal race radius + rndVar11)

For kRaceGeoImp1=2:

Amplitude (m or in) of Out-of-roundness, or variation in race radius corresponding to the sinusoidal function discussed above.

rndVar21

For kRaceGeoImp1=1:

Ratio of the semi major to minor axis deviation from the nominal race radius.

For kRaceGeoImp1=2:

Frequency (cycles) of out-of-roundness variation for the sinusoidal function.

rndVar31

For kRaceGeoImp1=1:

Orientation (deg) of the major axis relative to the body fixed z-axis of the race.

For kRaceGeoImp1=2:

Phase shift (deg) of out-of-roundness variation for the sinusoidal function.

cFacVar11

Amplitude of variation in curvature factor.

See discussion above under record title.

cFacVar21

Frequency (cycles) of curvature factor variation.

See discussion above under record title.

cFacVar31

Phase shift (deg) of curvature variation.

See discussion above under record title.

Record 5G.2.1B

Geometrical Imperfections on Outer Race for Cylindrical and Tapered Roller Bearings

This record is required only when geometric imperfections are to be prescribed on the outer race for cylindrical and tapered roller bearings:

kRaceGeoImp1 > 0 and kBrg = 2 or 4 on Record 3.2

For cylindrical and tapered roller bearings there are three imperfections: race out-of-roundness, central land offset and race taper. With the imperfection code, kRaceGeoImp1=1, the out-of-roundness is modeled by an elliptical profile, while a sinusoidal variation is considered with kRaceGeoImp1=2. Both the other imperfections, central land offset and race taper, are always modeled by a sinusoidal variation.

An elliptical variation if defined in terms of deviation of the semi major and minor axes from the nominal race radius. Thus if a and b are respectively the deviation of the semi major and minor axes from the nominal race radius then the elliptical profile is defined the following major and minor axes:

Semi-major axis = (nominal race radius + a)

Semi-minor axis = (nominal race radius + b)

The general form of a sinusoidal imperfection, a, is defined by a constant, A, amplitude, a_o, frequency ω and phase shift φ:

$$a = A + a_o \sin(\omega\theta + \varphi)$$

where θ is the angular position relative to the body-fixed z-axis, measured as a rotation about the body-fixed x-axis, which is also the bearing axis, as shown earlier in figure 23

With kRaceGeoImp1=2, the race radius is defined by the above sinusoidal function. However, the constant A is inherently set to zero, so this variable is not required for race radius variation. The variation in other parameters, such as race land offset and taper, are always defined by the general sinusoidal variation stated above.

Some of the data on this record is dimensional. It is essential that the units conform to the unit code defined later on Record 3.2. The units given below in parenthesis correspond to the SI and English system of units, as discussed at the beginning of this chapter.

recID

Record identifier - maximum 12 characters in single quotes.

rndVar11

For kRaceGeoImp1=1:

Deviation (m or in) of the semi-major axis of the elliptical race profile from the nominal race radius:

For kRaceGeoImp1=2:

Amplitude (m or in) of Out-of-roundness variation.

rndVar21

For kRaceGeoImp1=1:

Ratio of the semi major to minor axis deviation from the nominal race radius.

For kRaceGeoImp1=2:

Frequency (cycles) of out-of-roundness variation for the sinusoidal function.

rndVar31

For kRaceGeoImp1=1:

Orientation (deg) of the major axis relative to the body fixed z-axis of the race.

For kRaceGeoImp1=2:

Phase shift (deg) of out-of-roundness variation for the sinusoidal function.

rlOffset11

Constant (m or in) part of race land offset.

rlOffset21

Amplitude (m or in) of race land offset.

rlOffset31

Frequency (cycles) of race land offset.

rlOffset41

Phase shift (deg) of race land offset.

rlTaper11

Constant (rad) part of race land taper.

rlTaper21

Amplitude (rad) of race land taper.

rlTaper31

Frequency (cycles) of race land taper.

rlTaper41

Phase shift (deg) of race land taper.

Record 5G.2.2A

Geometrical Imperfections on Inner Race for Ball, Spherical and Spherical Tapered Roller Bearings

This record is required only when geometric imperfection are to be prescribed on the inner race for ball, spherical and spherical tapered roller bearings:

kRaceGeoImp2 > 0 and kBrg = 1, 3 or 5 on Record 3.2

For ball, spherical and tapered spherical roller bearings there may be two imperfections on the race: out-of-roundness and variation in race groove curvature. With the imperfection code, kRaceGeoImp2=1, race out-of-roundness is modeled by an elliptical profile, where the semi major and minor axes of the ellipse are defined as:

Semi-major axis = (r + a)

Semi-minor axis = (r + b)

where r is the nominal radius and the two parameters a and b define the radius variation.

With the imperfection code, kRaceGeoImp1=2, the race radius variation is prescribed by a sinusoidal variation around the race. The the magnitude of imperfection, a, is defined by amplitude, a_o, frequency ω and phase shift φ:

$$a = a_o \sin(\omega\theta + \varphi)$$

where θ is the angular position relative to the body-fixed z-axis, measured as a rotation about the body-fixed x-axis, which is also the bearing axis, as shown earlier in figure 23.

Thus three values, corresponding to amplitude a_o,frequency ω, and phase shift φ, define any geometric imperfection on the race.

The variation in race groove curvature is always prescribed in terms of a sinusoidal function discussed above.

Some of the data on this record is dimensional. It is essential that the units conform to the unit code defined later on Record 3.2. The units given below in parenthesis correspond to the SI and English system of units, as discussed at the beginning of this chapter.

recID

Record identifier - maximum 12 characters in single quotes.

rndVar12

For kRaceGeoImp2=1:

Deviation (m or in) of the semi-major axis of the elliptical race profile from the nominal race radius:

Semi-major axis = (nominal race radius + rndVar11)

For kRaceGeoImp2=2:

Amplitude (m or in) of Out-of-roundness, or variation in race radius corresponding to the sinusoidal function discussed above.

rndVar22

For kRaceGeoImp2=1:

Ratio of the semi major to minor axis deviation from the nominal race radius.

For kRaceGeoImp2=2:

Frequency (cycles) of out-of-roundness variation for the sinusoidal function.

rndVar32

For kRaceGeoImp2=1:

Orientation (deg) of the major axis relative to the body fixed z-axis of the race.

For kRaceGeoImp2=2:

Phase shift (deg) of out-of-roundness variation for the sinusoidal function.

cFacVar12

Amplitude of variation in curvature factor.

See discussion above under record title.

cFacVar22

Frequency (cycles) of curvature factor variation.

See discussion above under record title.

cFacVar32

Phase shift (deg) of curvature variation.

See discussion above under record title.

Record 5G.2.2B

Geometrical Imperfections on Inner Race for Cylindrical and Tapered Roller Bearings

This record is required only when geometric imperfections are to be prescribed on the inner race for cylindrical and tapered roller bearings:

kRaceGeoImp2 > 0 and kBrg = 2 or 4 on Record 3.2

For cylindrical and tapered roller bearings there are three imperfections: race out-of-roundness, central land offset and race taper. With the imperfection code, kRaceGeoImp2=1, the out-of-roundness is modeled by an elliptical profile, while a sinusoidal variation is considered with kRaceGeoImp2=2. Both the other imperfections, central land offset and race taper, are always modeled by a sinusoidal variation.

An elliptical variation if defined in terms of deviation of the semi major and minor axes from the nominal race radius. Thus if a and b are respectively the deviation of the semi major and minor axes from the nominal race radius then the elliptical profile is defined the following major and minor axes:

Semi-major axis = (nominal race radius + a)

Semi-minor axis = (nominal race radius + b)

The general form of a sinusoidal imperfection, a, is defined by a constant, A, amplitude, a_o, frequency ω and phase shift φ:

$$a = A + a_o \sin(\omega\theta + \varphi)$$

where θ is the angular position relative to the body-fixed z-axis, measured as a rotation about the body-fixed x-axis, which is also the bearing axis, as shown earlier in figure 23

With kRaceGeoImp2=2, the race radius is defined by the above sinusoidal function. However, the constant A is inherently set to zero, so this variable is not required for race radius variation. The variation in other parameters, such as race land offset and taper, are always defined by the general sinusoidal variation stated above.

Some of the data on this record is dimensional. It is essential that the units conform to the unit code defined later on Record 3.2. The units given below in parenthesis correspond to the SI and English system of units, as discussed at the beginning of this chapter.

recID

Record identifier - maximum 12 characters in single quotes.

rndVar12

For kRaceGeoImp2=1:

Deviation (m or in) of the semi-major axis of the elliptical race profile from the nominal race radius:

For kRaceGeoImp2=2:

Amplitude (m or in) of Out-of-roundness variation.

rndVar22

For kRaceGeoImp2=1:

Ratio of the semi major to minor axis deviation from the nominal race radius.

For kRaceGeoImp2=2:

Frequency (cycles) of out-of-roundness variation for the sinusoidal function.

rndVar32

For kRaceGeoImp2=1:

Orientation (deg) of the major axis relative to the body fixed z-axis of the race.

For kRaceGeoImp2=2:

Phase shift (deg) of out-of-roundness variation for the sinusoidal function.

rlOffset12

Constant (m or in) part of race land offset.

rlOffset22

Amplitude (m or in) of race land offset.

rlOffset32

Frequency (cycles) of race land offset.

rlOffset42

Phase shift (deg) of race land offset.

rlTaper12

Constant (rad) part of race land taper.

rlTaper22

Amplitude (rad) of race land taper.

rlTaper32

Frequency (cycles) of race land taper.

rlTaper42

Phase shift (deg) of race land taper.

3.6 Inertial Parameters for Rolling Elements and Races

Record 6.1

Inertial Parameters of Rolling Elements

Data on this record is required when optional data for the inertial parameters for rolling elements have to be prescribed, kReIP = 1 or 2 on Record 3.4.

All the data on this record is dimensional. It is essential that the units conform to the unit code defined later on Record 3.2. The units given below in parenthesis correspond to the SI and English system of units, as discussed at the beginning of this chapter.

recID

Record identifier - maximum 12 characters in single quotes.

bReMass

Rolling element mass (kgm or lbm).

kReIP = 1 on Record 3.4: Use this value for rolling element #1 only.
kReIP = 2 on Record 3.4: Use this value for all rolling elements.

bReMIx

Moment of inertia about (polar) x-axis (kgm*m^2 or lbm*in^2).

kReIP = 1 on Record 3.4: Use this value for rolling element #1 only.
kReIP = 2 on Record 3.4: Use this value for all rolling elements.

bReMIy

Moment of inertia about y-axis (kgm*m^2 or lbm*in^2).

kReIP = 1 on Record 3.4: Use this value for rolling element #1 only.
kReIP = 2 on Record 3.4: Use this value for all rolling elements.

bReMIz

Moment of inertia about z-axis (kgm*m^2 or lbm*in^2).

kReIP = 1 on Record 3.4: Use this value for rolling element #1 only.
kReIP = 2 on Record 3.4: Use this value for all rolling elements.

bReGeoCenX

X-component of rolling element geometric center relative to mass center (m or in) in rolling element geometric frame.

kReIP = 1 on Record 3.4: Use this value for rolling element #1 only.
kReIP = 2 on Record 3.4: Use this value for all rolling elements.

bReGeoCenY

Y-component of rolling element geometric center relative to mass center (m or in) in rolling element geometric frame.

kReIP = 1 on Record 3.4: Use this value for rolling element #1 only.
kReIP = 2 on Record 3.4: Use this value for all rolling elements.

bReGeoCenZ

Z-component of rolling element geometric center relative to mass center (m or in) in rolling element geometric frame.

kReIP = 1 on Record 3.4: Use this value for rolling element #1 only.
kReIP = 2 on Record 3.4: Use this value for all rolling elements.

bReFrameX

X-transformation angle (deg) to locate rolling element geometric reference frame relative to its principal axes frame.

kReIP = 1 on Record 3.4: Use this value for rolling element #1 only.
kReIP = 2 on Record 3.4: Use this value for all rolling elements.

bReFrameY

Y-transformation angle (deg) to locate rolling element geometric reference frame relative to its principal axes frame.

kReIP = 1 on Record 3.4: Use this value for rolling element #1 only.
kReIP = 2 on Record 3.4: Use this value for all rolling elements.

bReFrameZ

Z-transformation angle (deg) to locate rolling element geometric reference frame relative to its principal axes frame.

kReIP = 1 on Record 3.4: Use this value for rolling element #1 only.
kReIP = 2 on Record 3.4: Use this value for all rolling elements.

Record 6.2.1

Optional Inertial Parameters for the Outer Race

This record is required only when simulating acceleration of the outer race under arbitrary inertial parameters, mode = 0 on Record 1, and kRaceIP1 = 1 on Record 3.4.

All the data on this record is dimensional. It is essential that the units conform to the unit code defined later on Record 3.2. The units given below in parenthesis correspond to the SI and English system of units, as discussed at the beginning of this chapter.

recID

Record identifier - maximum 12 characters in single quotes.

raceMass1

Effective mass (kgm or lbf) of outer race.

raceMIx1

Outer race moment of inertia (kgm*m2 or lbm*in2) about its polar axis X.

raceMIy1

Outer race moment of inertia (kgm*m2 or lbm*in2) about its transverse axis Y.

raceMIz1

Outer race moment of inertia (kgm*m2 or lbm*in2) about its transverse axis Z.

raceGeoCenX1

X-component (m or in) of vector locating outer race geometric center relative to its mass center in race frame.

raceGeoCenY1

Y-component (m or in) of vector locating outer race geometric center relative to its mass center in race frame.

raceGeoCenZ1

Z-component (m or in) of vector locating outer race geometric center relative to its mass center in race frame.

raceFrameX1

X-transformation angle (deg) defining outer race geometric frame relative to its principal frame.

raceFrameY1

Y-transformation angle (deg) defining outer race geometric frame relative to its principal frame.

raceFrameZ1

Z-transformation angle (deg) defining outer race geometric frame relative to its principal frame.

Record 6.2.2

Optional Inertial Parameters for the Inner Race

This record is required only when simulating acceleration of the inner race under arbitrary inertial parameters, mode = 0 on Record 1, and kRaceIP2 = 1 on Record 3.4.

All the data on this record is dimensional. It is essential that the units conform to the unit code defined later on Record 3.2. The units given below in parenthesis correspond to the SI and English system of units, as discussed at the beginning of this chapter.

recID

Record identifier - maximum 12 characters in single quotes.

raceMass2

Effective mass (kgm or lbf) of inner race.

raceMIx2

Inner race moment of inertia (kgm*m^2 or lbm*in^2) about its polar axis X.

raceMIy2

Inner race moment of inertia (kgm*m^2 or lbm*in^2) about its transverse axis Y.

raceMIz2

Inner race moment of inertia (kgm*m^2 or lbm*in^2) about its transverse axis Z.

raceGeoCenX2

X-component (m or in) of vector locating inner race geometric center relative to its mass center in race frame.

raceGeoCenY2

Y-component (m or in) of vector locating inner race geometric center relative to its mass center in race frame.

raceGeoCenZ2

Z-component (m or in) of vector locating inner race geometric center relative to its mass center in race frame.

raceFrameX2

X-transformation angle (deg) defining inner race geometric frame relative to its principal frame.

raceFrameY2

Y-transformation angle (deg) defining inner race geometric frame relative to its principal frame.

raceFrameZ2

Z-transformation angle (deg) defining inner race geometric frame relative to its principal frame.

3.7 Cage Parameters

Record 7.0

Cage Options

This record is required only if a cage is present, nCseg > 0 on Record 3.2.

recID

Record identifier - maximum 12 characters in single quotes.

kPocType

Cage pocket shape code.

For ball bearings the available codes are:

0	Cylindrical pockets.
1	Spherical pockets
2	Elongated cylindrical pockets.
3	Rectangular pockets.
4	Conical pockets.

The various shapes are defined below in figure 24:

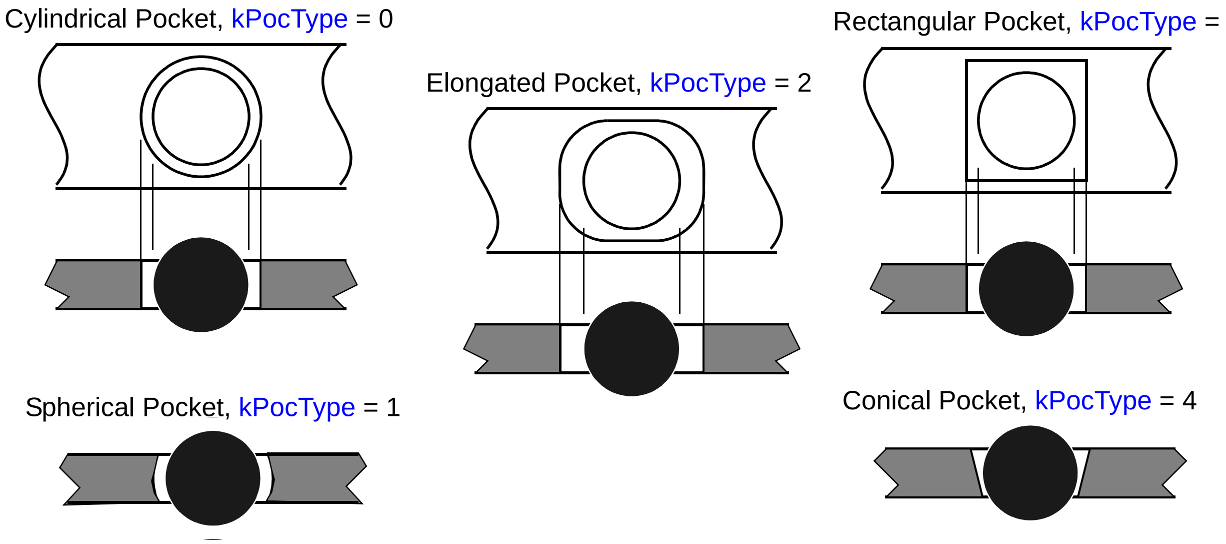

Cylindrical Pocket, kPocType = 0
Rectangular Pocket, kPocType =
Elongated Pocket, kPocType = 2
Spherical Pocket, kPocType = 1
Conical Pocket, kPocType = 4

Figure 24. Types of cage pockets for a ball bearing.

For all roller bearings, pocket shape options are:

-1	Cylindrical pockets for roller guided cage.
0	Rectangular pockets.

n (n > 0) Pair of cage pocket interaction surfaces in the cage pocket (maximum 3). a pair consists of two surfaces symmetrically located on the fore and aft side of the cage pocket.

The various pocket configurations are described below in figure 25:

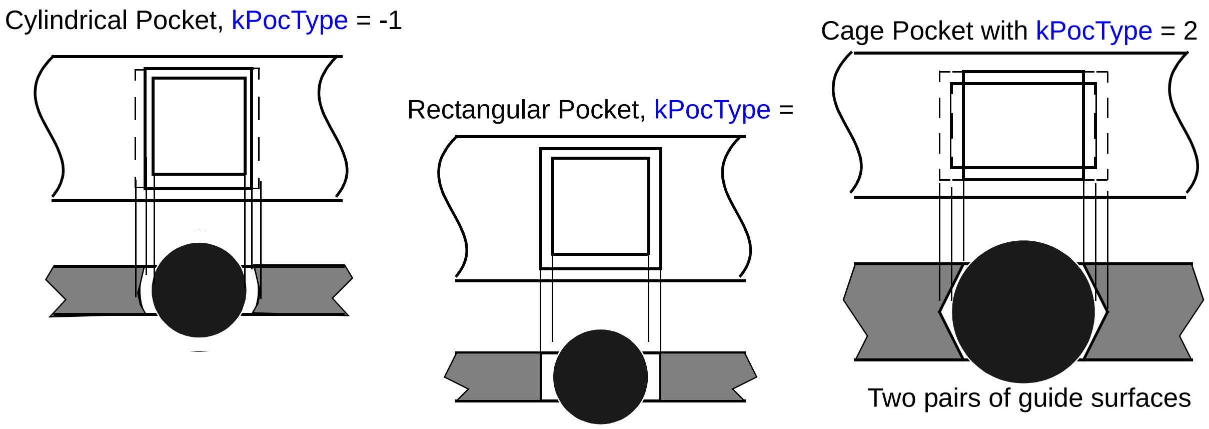

Cylindrical Pocket, kPocType = -1
Rectangular Pocket, kPocType =
Cage Pocket with kPocType = 2
Two pairs of guide surfaces

Figure 25. Types of cage pockets for a roller bearing.

kCagePocImp

Code for geometrical imperfections in cage pockets:

0 Ideal pocket geometry.

1 Only pocket #1 is imperfect.

2 Equal imperfections in all pockets.

3 Imperfections are normally distributed.

4 Imperfections are prescribed in subroutine Adrx8.

kCageGsImp

Geometrical imperfections on cage guide lands. This is only relevant when nGL > 0 on this record:

0 Ideal geometry, no imperfections.

1 Elliptical cage guide land imperfections data on record 7.5.

2 Sinusoidal variation in cage land radius imperfections data on record 7.5.

kRaceGsImp

Geometrical imperfections on race guide lands. This is only relevant when nGL > 0 on this record:

0 Ideal geometry, no imperfections.

1 Elliptical race guide land imperfections data on record 7.5.

2 Sinusoidal variation in race land radius imperfections data on record 7.5.

kPocHydro

Hydrodynamics code at rolling element/cage interaction:

0 Neglect hydrodynamics.

1 Include hydrodynamics.

kGsHydro

Hydrodynamics code at cage/race interface:

0 Neglect hydrodynamics.

1 Include hydrodynamics.

kCageMat

Material code for the cage:

0 Standard material (Mild steel).

1 Cage material properties specified on record 8.5.

2 Material properties to be extracted from user data base via user subroutine ADRX0.

m Material code for property data base in ADORE. See available material codes below.

kCageIP

Inertial parameters for the cage or cage segments:

0 Standard parameters (ideal geometry).

1 Inertial parameters specified on record 7.7.

nGL

Number of cage/race guide lands.

iCageGuide(i),i=1,nGL

Type of cage guidance on ith land, i=1,nGL

0 No race guidance.

1 Outer race guidance.

2 Inner race guidance.

Presently available material codes (m) in ADORE database are:

m	Material
100	AISI 52100 Bearing Steel
101	M50 Bearing Steel
102	M50 VIM-VAR Bearing Steel
103	440C Stainless Steel
104	430 Ferratic Stainless Steel
105	410 Martenitic Stainless Steel
106	304 Austenitic Stainless Steel
107	AMS 5898 Cronidur 30 Stainless Steel
108	AMS 5643 (17-4PH) Stainless Steel
110	C1045 Steel
111	AISI 4340 Steel

112	Inconel 625 Alloy
113	Inconel 718 Alloy
114	AISI 304HN High Nitrogen Steel
120	M-50 Nil (Case hardened steel)
121	P-675 HTT (Case hardened steel)
122	P-675 LTT (Case hardened steel)
150	Si3N4 Silicon Nitride
151	Zirconium Oxide (ZrO2)
160	Copper
161	Brass
162	Bronze
200	Bearing Grade Peek
201	Polyamide-Nylon
202	Armalon
203	Carbon Phenolic
204	Carbon Phenolic (10% MoS2)
205	Cotton Phenolic
206	Graphite
207	Teflon (PTFE)

Record 7.0.1

Cage Segmentation Details

The record is record is required only for segmented cage, nCseg > 1 on Record 3.2.

For a segmented cage, it is necessary that all segments be identical to each other and the segmentation takes place either in the center of the pockets or in the center of the wall between pockets. The geometry of a segmented cage is prescribed simply as if it were a full one piece cage. Segmentation is introduced by specifying the number of segments and the angular width of cut (degrees) used to segment the cage. No hydrodynamic effects (both in the cage pocket and at the cage/race interaction) may be considered with a segmented cage.

recID

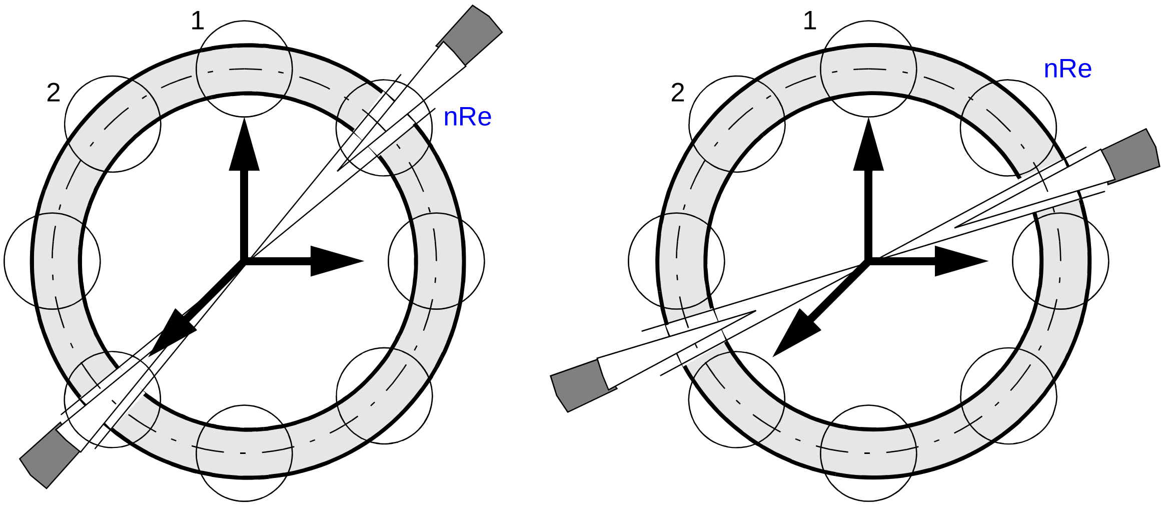
Record identifier - maximum 12 characters in single quotes.

lSeg

A vector of length nCseg (# of cage segments as defined on Record 3.2) containing the rolling element number located at start of the cage of segment.

lSeg(i) > 0 Segment #i starts just before rolling element #lSeg(i).
Segmentation is just before rolling element #lSeg(i), see figure 26 below.

lSeg(i) < 0 Segment #i starts at rolling element #lSeg(i).
Segmentation is through pocket #lSeg(i), see figure 26 below.



lSeg(i) > 0 Segmentation through the pocket

lSeg(i) > 0 Segmentation between the pockets

Figure 26. Cage segmentation details.

Note that segmentation should be such that all segments are identical to each other. Results may be unpredictable if the segments are not identical.

Record 7.1

Overall Cage Geometry

This record is required when a cage is present, nCseg > 0 on Record 3.2

All of the data on this record is dimensional. It is essential that the units conform to the unit code defined later on Record 3.2. The units given below in parenthesis correspond to the SI and English system of units, as discussed at the beginning of this chapter.

recID

Record identifier - maximum 12 characters in single quotes.

cageDia1

Cage outer diameter (m or in).

cageDia2

Cage inner diameter (m or in).

cageWidth

Cage width (m or in).

cageCls1

Cage/race outer diametral clearance (m or in).

cageCls2

Cage/race inner diametral clearance (m or in).

bPocCls1

Cage pocket clearance I (m or in) defined as follows:

1. For ball bearings with cylindrical, spherical or rectangular pockets (kPocType = 0, 1 or 3 on Record 7.0), and for all roller bearings (except when kPocType > 0, in which case is not used), bPocCls1 is the diametral pocket clearance (m or in) in the circumferential direction.
2. For elongated pockets in ball bearings (kPocType = 2 on Record 7.0), bPocCls1 is the diametral pocket clearance (m or in) in the axial direction.
3. For conical pockets in ball bearings (kPocType = 4 on Record 7.0), bPocCls1 is the difference (m or in) between the inner pocket diameter and the nominal ball diameter.

bPocCls2

Cage pocket clearance II (m or in) defined as follows:

1. For ball bearings with cylindrical or spherical pockets (kPocType = 0 or 1 on Record 7.0), or for all roller bearings, bPocCls2 is zero.
2. For elongated pockets in ball bearings (kPocType = 2 on Record 7.0), bPocCls2 is the offset (m or in) between the two pocket centers.
3. For rectangular pockets (kPocType = 3 on Record 7.0), bPocCls2 is the diametral clearance (m or in) in the axial direction.
4. For conical pockets in ball bearings (kPocType = 4 on Record 7.0), bPocCls2 is the difference (m or in) between the inner pocket diameter and the nominal ball diameter.

cageAngCut

Angular width of cut (deg), as defined figure 27, when the cage is segmented.

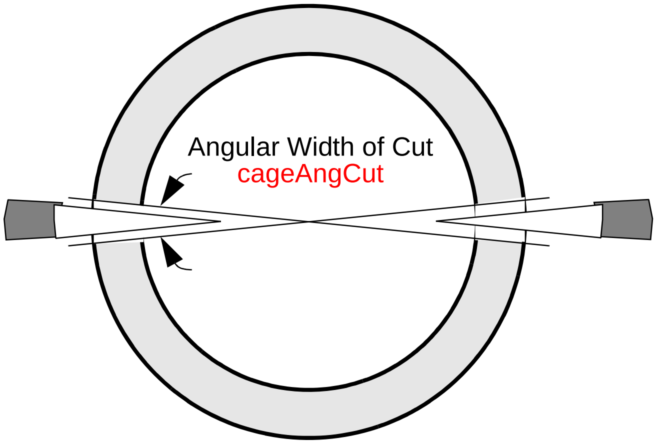


Figure 27. Angular width of cut in case of a segmented cage.

cageConeAng

Cage semi cone angle (deg), as shown in figure 28, when cage is conical, generally in tapered roller bearings.

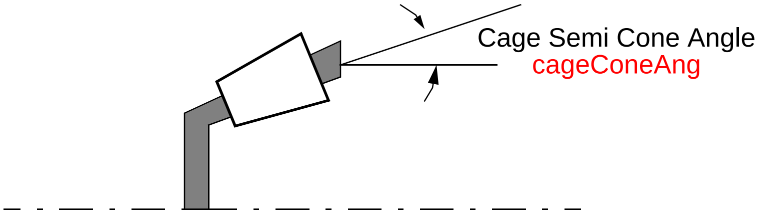


Figure 28. Cage semi cone angle in case of a tapered roller bearing.

Record 7.2.i, i=1,nGL

Cage/Race Guide Land Geometry

This record is required only when a cage is present in the bearing, nCseg > 0 on Record 3.2 and the cage is guided on the race, nGL > 0 on Record 7.0, nGL = number of cage/race guide lands.

This data record is repeated independently for each guide land. Thus the geometry at each guide land may be different. The type of guidance at each land is specified in the array iCageGuide on Record 7.0. As an example figure 29 shows two guide lands one guided on the outer race while the other is guided on the inner race. Such a configuration, is simply for illustrative purpose it does not represent any specific practical application. In total there may be a maximum of four guide lands, two on the negative x-axis, interacting with the outer and inner races, and two on the positive x-axis, again interacting with the outer and inner races. In most practical applications cage guidance is either on the outer or inner race. However, if either the cage or the race surface at the guide lands is not circular the cage may interact with both races. In such cases the options for guidance on both races must be turned on simulate a potential problem.

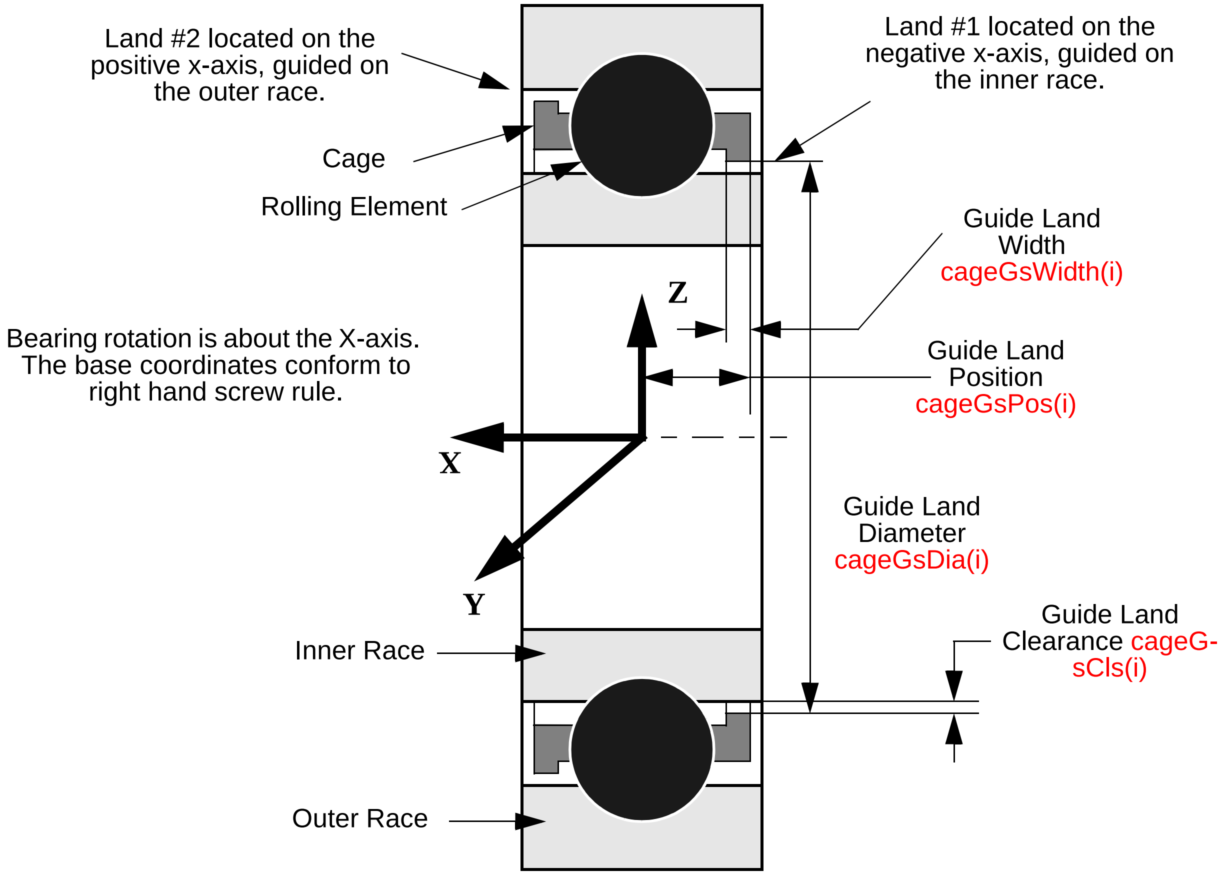

Land #2 located on the positive x-axis, guided on the outer race.
Land #1 located on the negative x-axis, guided on the inner race.
Cage
Rolling Element
Guide Land Width cageGsWidth(i)
Z
Bearing rotation is about the X-axis. The base coordinates conform to right hand screw rule.
Guide Land Position cageGsPos(i)
X
Guide Land Diameter cageGsDia(i)
Y
Guide Land Clearance cageG-sCls(i)
Inner Race
Outer Race

Figure 29. Cage/Race guide land definitions.

recID

Record identifier - maximum 12 characters in single quotes.

cageGsDia(i)

Cage guide land diameter (m or in) for land #i.

cageGsWidth(i)

Land width (m or in) for the land #i.

cageGsPos(i)

Distance (m or in) of outer edge of land #i from the geometric center of cage.

cageGsCls(i)

Diametral clearance (m or in) on land #i.

Record 7.3.i, i=1,kPocType

Geometry of Cage Pocket Surfaces for Roller Bearings

This record is required only roller bearings with cage, nCseg > 0 and kBrg > 1 on Record 3.2, when arbitrary guide surfaces have to be prescribed, kPocType > 0 on Rec 7.0

The data on this record is prescribed for each pair of pocket guide surfaces. The number of guide surface pairs is defined by value of kPocType on rec 7.0, kPocType > 0. Thus i varies for 1 to n. The data is supplied on the guide surface located on the positive y-axis of the pocket frame, as shown in figure 30. A corresponding surface on the negative x-axis, to form a pair, is internally defined by symmetry.

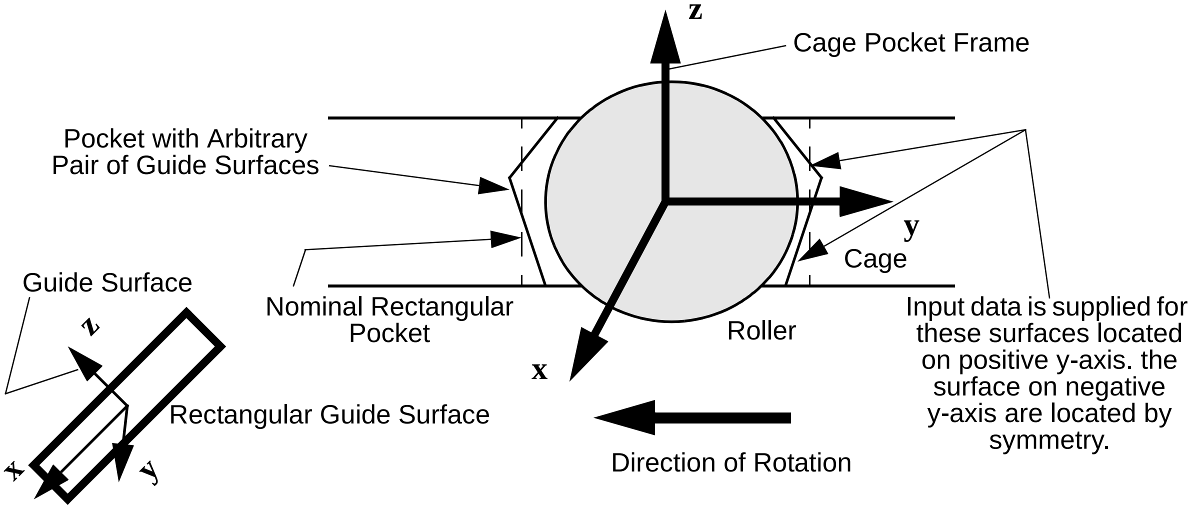


Figure 30. Definition of cage pocket guide surfaces for roller bearings.

The data record is repeated for each surface pair. In the event the surfaces are not symmetric about the x-axis of the pocket frame, then surface definition is accomplished in the designated user programmable subroutine.

recID

Record identifier - maximum 12 characters in single quotes.

bPocGsAng1(i)

Pocket guide surface transformation angle-x (deg), located the guide surface frame relative to the pocket frame. The angle is defined as rotation about the x-axis.

bPocGsAng2(i)

Pocket guide surface transformation angle-y (deg), locating the guide surface frame relative to the pocket frame. The angle is defined as rotation about the y-axis.

bPocGsAng3(i)

Pocket guide surface transformation angle-z (deg), locating the guide surface frame relative to the pocket frame. The angle is defined as rotation about the z-axis.

bPocGsCen1(i)

X-coordinate of guide surface (m or in) center relative to the pocket center bPocGsCen2(i)

Y-coordinate of guide surface (m or in) center relative to the pocket center.

bPocGsCen3(i)

Z-coordinate of guide surface (m or in) center relative to the pocket center.

bPocGsLen1(i)

Guide surface width (m or in), surface dimension along the z-axis as shown above.

bPocGsLen2(i)

Guide surface length (m or in), surface dimension along the x-axis.

Record 7.4

Cage Pocket Geometric Imperfections

This record is required only when a cage is present, nCseg > 0 on Rec 3.2, and the cage pocket geometric imperfection flag kCagePocImp on Record 7.0 has a value between 1 and 3, 0 < kCagePocImp < 4.

The data contains deviation of the various geometrical parameters from their nominal values specified on Record 7.1 and the actual type of variations are defined by the value of kCagePocImp as follows:

kCagePocImp = 1

The specified data represents actual deviation of the various dimensions from their nominal value on Record 7.1 for pocket #1 only. All other pockets have no imperfections.

kCagePocImp = 2

The specified data represents actual deviation of the various dimensions from their nominal value on Record 7.1 for all pockets.

kCagePocImp = 3

The specified data represents an rms deviation of the various dimensions from their nominal value on Record 7.1 and the actual imperfections in individual pockets are computed from a normal distribution.

For kCagePocImp = 4 arbitrary geometric imperfections may be programmed in user subroutine Adrx8 and this data record is not required.

recID

Record identifier - maximum 12 characters in single quotes.

bPocClsVar1

Deviation in cage pocket clearance I (m or in).

bPocClsVar2

Deviation in cage pocket clearance II (m or in).

bPocThknsVar

Deviation of pocket thickness (m or in) from the nominal value, which is equal to the difference between the outer and inner radii of the cage.

bPocCenVar1

Axial position (m or in) of pocket center relative to the ideally centered position.

bPocCenVar2

Angular position (deg) of pocket center relative to the geometrically ideal location.

bPocAngVar1

Variation in first transformation angle (deg) for pocket frame.

bPocAngVar2

Second transformation angle (deg) for pocket frame.

bPocAngVar3

Third transformation angle (deg) for pocket frame.

Record 7.5.i, i=1,nGL

Cage Guide Land Geometric Imperfections

This record is required only when a cage is present, nCseg > 0 on Record 3.2, it is guided on the races, nGL > 0 on Record 7.0, and cage guide land geometric imperfection flag kCageGsImp has a value of 1 or 2, 0 < kCageGsImp < 3 on Record 7.0.

The data record is repeated for each guide land.

recID

Record identifier - maximum 12 characters in single quotes.

cageGsRadVar1

First cage land radius variation parameter defined as:

kCageGsImp=1: Elliptical cage guide land:
(semi Y-axis - nominal radius) (m or in).

kCageGsImp=2: Sinusoidal variation in guide land radius:
Amplitude of radius variation (m or in).

cageGsRadVar2

Second cage land radius variation parameter defined as:

kCageGsImp=1: Elliptical cage guide land:
(semi Z-axis - nominal radius) (m or in).

kCageGsImp=2: Sinusoidal variation in guide land radius:
Frequency of radius variation defined as number of peaks in the radius profile.

cageGsRadVar3

Third cage land radius variation parameter defined as:

kCageGsImp=1: Elliptical cage guide land:
This parameter is not used, it may be left at a value of 0.

kCageGsImp=2: Sinusoidal variation in guide land radius:
Phase shift (deg) of radius variation.

Record 7.6.i, i=1.nGL

Race Land Geometric Imperfections

This record is required only when a cage is present, nCseg > 0 on Record 3.2, it is guided on the races, nGL > 0 on Record 7.0, and race guide land geometric imperfection flag kRaceGsImp has a value of 1 or 2, 0 < kRaceGsImp < 3 on Record 7.0.

The data record is repeated for each guide land.

recID

Record identifier - maximum 12 characters in single quotes.

raceGsRadVar1

First race land radius variation parameter defined as:

kRaceGsImp=1: Elliptical race guide land:
(semi Y-axis - nominal radius) (m or in).

kRaceGsImp=2: Sinusoidal variation in guide land radius:
Amplitude of radius variation (m or in).

raceGsRadVar2

Second race land radius variation parameter defined as:

kRaceGsImp=1: Elliptical race guide land:
(semi Z-axis - nominal radius) (m or in).

kRaceGsImp=2: Sinusoidal variation in guide land radius:
Frequency of radius variation defined as number of peaks in the radius profile.

raceGsRadVar3

Third race land radius variation parameter defined as:

kRaceGsImp=1: Elliptical race guide land:
This parameter is not used, it may be left at a value of 0.

kRaceGsImp=2: Sinusoidal variation in guide land radius:
Phase shift (deg) of radius variation.

Record 7.7

Arbitrary Inertial parameters for the Cage

This record is required when arbitrary inertial parameters for the cage have to be prescribed. Only for arbitrary inertial parameters for the cage, nCseg > 0 on Record 3.2 and kCageIP > 0 on Rec 7.0

recID

Record identifier - maximum 12 characters in single quotes.

cageMass

Cage mass (kgm or lbm).

cageMIx

Moment of inertia (kgm*m^2 or lbm*in^2) of the cage about the polar x-axis.

cageMIy

Moment of inertia (kgm*m^2 or lbm*in^2) of the cage about the transverse y-axis.

cageMIz

Moment of inertia (kgm*m^2 or lbm*in^2) of the cage about the transverse z-axis.

cageGeoCenX

X-coordinate of cage geometric center relative to its mass center in cage fixed frame.

cageGeoCenY

Y-coordinate of cage geometric center relative to its mass center in cage fixed frame.

cageGeoCenZ

Z-coordinate of cage geometric center relative to its mass center in cage fixed frame.

cageFrameX

X transformation angle defining the cage fixed geometrical reference frame relative to principal frame.

cageFrameY

Y transformation angle defining the cage fixed geometrical reference frame relative to principal frame.

cageFrameZ

Z transformation angle defining the cage fixed geometrical reference frame relative to principal frame.

3.8 Material Properties

Record 8.1

Rolling Element Material Properties

Data on this record is required for arbitrary rolling element material, kReMat > 0 Rec 3.3

recID

Record identifier - maximum 12 characters in single quotes.

reDen

Rolling element density (kgm/m^3 or lbm/in^3).

reEM

Rolling element elastic modulus (N/m^2 or lbf/in^2).

rePR

Rolling element Poisson’s ratio.

reCTE

Coefficient of thermal expansion of rolling element (m/m/K or in/in/R).

reHC

Heat capacity of rolling elements (J/kg/K or lbf.in/lbm/R).

reTC

Thermal conductivity of rolling elements (W/m/K or lbf.in/in/R/s)

reESL

Elastic strain limit for the rolling element.

reH

Rolling element hardness (Rockwell-C).

reWC

Rolling element wear coefficient.

Record 8.2.1

Outer Race Material Properties

Required for arbitrary material properties of the outer race, kRaceMat1 > 0 Rec 3.3

recID

Record identifier - maximum 12 characters in single quotes.

raceDen1

Material density (kgm/m^3 or lbm/in^3) for outer race.

raceEM1

Elastic modulus (N/m^2 or lbf/in^2) for outer race.

racePR1

Poisson’s ratio for outer race.

raceCTE1

Coefficient of thermal expansion (m/m/K or in/in/R) for outer race.

raceHC1

Heat capacity of outer race (J/kg/K or lbf.in/lbm/R).

raceTC1

Thermal conductivity of outer race (W/m/K or lbf.in/in/R/s).

raceESL1

Elastic strain limit for the outer race.

raceH1

Hardness (Rockwell-C) for outer race.

raceWC1

Wear coefficient for outer race.

Record 8.2.2

Inner Race Material Properties

Required for arbitrary material properties of the inner race, kRaceMat2 > 0 Rec 3.3

recID

Record identifier - maximum 12 characters in single quotes.

raceDen2

Material density (kgm/m^3 or lbm/in^3) for inner race.

raceEM2

Elastic modulus (N/m^2 or lbf/in^2) for inner race.

racePR2

Poisson’s ratio for inner race.

raceCTE2

Coefficient of thermal expansion (m/m/K or in/in/R) for inner race.

raceHC2

Heat capacity of inner race (J/kg/K or lbf.in/lbm/R).

raceTC2

Thermal conductivity of inner race (W/m/K or lbf.in/in/R/s).

raceESL2

Elastic strain limit for the inner race.

raceH2

Hardness (Rockwell-C) for inner race.

raceWC2

Wear coefficient for inner race.

Record 8.3

Shaft Material Properties

Data required for arbitrary shaft material, kShftMat > 0 on Rec 3.3

recID

Record identifier - maximum 12 characters in single quotes.

shftDen

Material density (kgm/m^3 or lbm/in^3) for the shaft.

shftEM

Elastic modulus (N/m^2 or lbf/in^2) for the shaft.

shftPR

Poisson's ratio for the shaft.

shftCTE

Coefficient of thermal expansion (m/m/K or in/in/R) for the shaft.

shftTC

Thermal conductivity of shaft (W/m/K or lbf.in/in/R/s)

Record 8.4

Housing Material Properties

Data required for arbitrary housing material, kHsngMat > 0 on Rec 3.3

recID

Record identifier - maximum 12 characters in single quotes.

hsngDen

Material density (kgm/m^3 or lbm/in^3) for the housing.

hsngEM

Elastic modulus (N/m^2 or lbf/in^2) for the housing.

hsngPR

Poisson's ratio for the housing.

hsngCTE

Coefficient of thermal expansion (m/m/K or in/in/R) for the housing.

hsngTC

Thermal conductivity of housing (W/m/K or lbf.in/in/R/s)

Record 8.5

Cage Material Properties

Data record required for arbitrary cage material, kCageMat > 0 on Rec 7.0

recID

Record identifier - maximum 12 characters in single quotes.

cageDen

Cage density (kgm/m^3 or lbm/in^3).

cageEM

Cage elastic modulus (N/m^2 or lbf/in^2).

cagePR

Cage Poisson's ratio.

cageCTE

Coefficient of thermal expansion of cage (m/m/K or in/in/R).

cageHC

Heat capacity of cages (J/kg/K or lbf.in/lbm/R).

cageTC

Thermal conductivity of cage (W/m/K or lbf.in/in/R/s)

cageESL

Elastic strain limit for the cage.

cageH

Cage hardness (Rockwell-C).

cageWC

Cage wear coefficient.

Record 8.6.0

Bearing Weibull Dispersion

Data Record required for arbitrary fatigue life parameters, kLifeCons=1 on Rec 3.3.

For definition of various constants in the fatigue life model see the following references, which document all the life formulae used in ADORE:

Gupta, P.K. and Tallian, T.E., "Rolling Bearing Life Prediction - Correction for Materials and Operating Conditions - Part III: Implementation in Bearing Dynamics Computer Code", ASME Journal of Tribology, vol 112, pp 23-26, January 1990.

Tallian, T.E., "A Data-Fitted Rolling Bearing Life Prediction Model - Part IV: Model Implementation for Current Engineering Use", STLE Tribology Transactions, Vol 39, 1996, pp 957-963.

Tallian, T.E., "Data Fitted Bearing Life Prediction Model for Variable Operating Conditions", STLE Transactions, Vol 42, 1999, pp 241-249.

Gupta, P.K., Oswald, F.B. and Zaretsky, E.V., “Comparison of Models Rolling Bearing Dynamic Capacity and Life”, to be published STLE Transactions.

recID

Record identifier - maximum 12 characters in single quotes.

sProb

Survival probability for the bearing.

brgWbDis

Weibull dispersion slope for the bearing.

This may be different from the values prescribed for the races on records below.

Record 8.6.1

Fatigue Life Parameters for Outer Race

Data Record required for arbitrary fatigue life parameters, kLifeCons=1 on Rec 3.3

For definition of various constants in the fatigue life model see the following references, which document all the life formulae used in ADORE:

Gupta, P.K. and Tallian, T.E., "Rolling Bearing Life Prediction - Correction for Materials and Operating Conditions - Part III: Implementation in Bearing Dynamics Computer Code", ASME Journal of Tribology, vol 112, pp 23-26, January 1990.

Tallian, T.E., "A Data-Fitted Rolling Bearing Life Prediction Model - Part IV: Model Implementation for Current Engineering Use", STLE Tribology Transactions, Vol 39, 1996, pp 957-963.

Tallian, T.E., "Data Fitted Bearing Life Prediction Model for Variable Operating Conditions", STLE Transactions, Vol 42, 1999, pp 241-249.

Gupta, P.K., Oswald, F.B. and Zaretsky, E.V., “Comparison of Models Rolling Bearing Dynamic Capacity and Life”, to be published STLE Transactions.

Data on this record specifies the parameters for the outer race.

recID

Record identifier - maximum 12 characters in single quotes.

fco1

Factor which modifies the default fatigue constant for the original Lundberg-Palmgen model for the outer race. Default value is 1.0.

fcLP1

Factor which modifies the default fatigue constant for the updated Lundberg-Palmgen model for the outer race. Default value is 1.0.

shExLP1

Shear stress exponent in the updated Lundberg-Palmgren model

for the outer race.

depExLP1

Shear stress depth exponent in the updated Lundberg-Palmgren model

for the outer race.

fcZ1

Factor which modifies the default fatigue constant for the Zaretsky model. for the outer race. Default value is 1.0.

shExZ1

Shear stress exponent for the Zaretsky model for the outer race.

shearLmtIH1

Ioannides-Harris (I-H) shear stress for infinite life (Pa or lbf/in^2). Default value is:

1.00E+08 Pa or 1.45E+04 lbf/in^2.

wbDis1

Weibull dispersion exponent for outer race.

Record 8.6.2

Fatigue Life Parameters for Inner Race

Data Record required for arbitrary fatigue life parameters, kLifeCons=1 on Rec 3.3. The specified data corresponds to the inner race.

recID

Record identifier - maximum 12 characters in single quotes.

fco2

Factor which modifies the default fatigue constant for the original Lundberg-Palmgen model for the inner race. Default value is 1.0.

fcLP2

Factor which modifies the default fatigue constant for the updated Lundberg-Palmgen model for the inner racer. Default value is 1.0.

shExLP2

Shear stress exponent in the updated Lundberg-Palmgren model

for the inner race.

depExLP2

Shear stress depth exponent in the updated Lundberg-Palmgren model

for the inner race.

fcZ2

Factor which modifies the default fatigue constant for the Zaretsky model, for the inner race. Default value is 1.0.

shearLmtIH2

Ioannides-Harris (I-H) shear stress for infinite life (Pa or lbf/in^2). Default value is:

1.00E+08 Pa or 1.45E+04 lbf/in^2.

wbDis2

Weibull dispersion exponent for inner race.

Record 8.6.3

Life Modification Parameters for Outer Race

Data record required for arbitrary life modification parameters, kLifeMod = 99 on Rec 3.3.

The data on this record corresponds to the outer race.

recID

Record identifier - maximum 12 characters in single quotes.

rmsAspSlope1

Composite rms asperity slope (rad) for outer race.

shearLmt1

Limit shear stress (Pa or lbf/in^2) for outer race.

aspTrac1

Asperity traction coefficient for outer race.

resStress1

Residual stress (Pa or lbf/in^2) in the outer race.

facMat1

Material factor for the outer race.

Suggested values:

52100 Steel	1.197
8620 Steel	1.773
M50 Steel	2.267

facCont1

Contamination for the outer race.

Suggested default 1.0

For VIMVAR process, for aerospace applications, a factor as low as 0.10 may be used.

facProc1

Materials processing factor for the outer race.

Suggested values:

CVD old: Carbon vacuum deoxidation, through-hardening steel (groups pre-dating 1975) = 2.58.
CVD new: Carbon vacuum deoxidation, through-hardening steel (groups dating 1975 and later) = 0.077 (Default).
CVD carb: Carbon vacuum deoxidation, carburizing steel (all dates) = 4.85.
VIMVAR: Vacuum induction melt, vacuum arc remelt = 0.003, established with contamination factor, facCont1 = 0.10 for aerospace applications.

Record 8.6.4

STLE Life Modification Parameters for Outer Race

Data record required for arbitrary life modification parameters for the STLE model.

kLifeMod = 99 on Rec 3.3.

The data on this record corresponds to the outer race.

recID

Record identifier - maximum 12 characters in single quotes.

facMatLF1

STLE materials factor of the outer race.

facProcLF1

STLE materials processing factor of the outer race.

hardnessLF1

STLE hardness factor of the outer race.

Record 8.6.5

Life Modification Parameters for Inner Race

Data record required for arbitrary life modification parameters, kLifeMod = 99 on Rec 3.3.

For definition of various constants in the fatigue life model see the following references, which document all the life formulae used in ADORE:

Gupta, P.K. and Tallian, T.E., "Rolling Bearing Life Prediction - Correction for Materials and Operating Conditions - Part III: Implementation in Bearing Dynamics Computer Code", ASME Journal of Tribology, vol 112, pp 23-26, January 1990.

Tallian, T.E., "A Data-Fitted Rolling Bearing Life Prediction Model - Part IV: Model Implementation for Current Engineering Use", STLE Tribology Transactions, Vol 39, 1996, pp 957-963.

Tallian, T.E., "Data Fitted Bearing Life Prediction Model for Variable Operating Conditions", STLE Transactions, Vol 42, 1999, pp 241-249.

The data on this record corresponds to the inner race.

recID

Record identifier - maximum 12 characters in single quotes.

rmsAspSlope2

Composite rms asperity slope (rad) for inner race.

shearLmt2

Limit shear stress (Pa or lbf/in^2) for inner race.

aspTrac2

Asperity traction coefficient for inner race.

resStress2

Residual stress (Pa or lbf/in^2) in the inner race.

facMat2

Material factor for the inner race.

Suggested values:

52100 Steel	1.197
8620 Steel	1.773
M50 Steel	2.267

facCont2

Contamination for the inner race.

Suggested default 1.0

For VIMVAR process, for aerospace applications, a factor as low as 0.10 may be used.

facProc2

Materials processing factor for the inner race.

Suggested values:

CVD old: Carbon vacuum deoxidation, through-hardening steel (groups pre-dating 1975) = 2.58.
CVD new: Carbon vacuum deoxidation, through-hardening steel (groups dating 1975 and later) = 0.077 (Default).
CVD carb: Carbon vacuum deoxidation, carburizing steel (all dates) = 4.85.
VIMVAR: Vacuum induction melt, vacuum are remelt = 0.003, established with contamination factor, facCont2 = 0.10 for aerospace applications.

Record 8.6.6

STLE Life Modification Parameters for Inner Race

Data record required for arbitrary life modification parameters for the STLE model.

kLifeMod = 99 on Rec 3.3.

The data on this record corresponds to the inner race.

recID

Record identifier - maximum 12 characters in single quotes.

facMatLF1

STLE materials factor of the inner race.

facProcLF1

STLE materials processing factor of the inner race.

hardnessLF1

STLE hardness factor of the inner race.

3.9 Operating Conditions

Record 9.0

Mounted Race Fits

recID

Record identifier - maximum 12 characters in single quotes.

roomTemp

Room temperature (K or R).

raceFit1

Diametral mounted shrink fit allowance on outer race (m or in) at room temperature.

raceFit2

Diametral mounted shrink fit allowance on inner race (m or in) at room temperature.

For an interference fit the shrink fit allowance is positive while a negative value indicates a loose fit.

Record 9.1.1

Applied Loads or Displacements

The operating data supplied on records 9.1.1 and 9.1.2 is used for computing the quasi-static equilibrium solution, which may be used for computing the initial conditions for the dynamic solutions. Any time-dependent operating conditions must be programmed in the optional subroutine Adrx1.

recID

Record identifier - maximum 12 characters in single quotes.

appLoadX

Applied force (N or lbf) along the x-axis when kFS1=0 on record 3.3.

appLoadY

Applied force (N or lbf) along the y-axis when kFS2=0 on record 3.3.

appLoadZ

Applied force (N or lbf) along the z-axis when kFS3=0 on record 3.3.

appDispX

Relative race displacement (m or in) along the x-axis when kFS1=1 on record 3.3, or initial guess for relative race displacement, along the x-axis, when kFS1=0 on record 3.3.

When kFS1=0 and appDispX is set to zero, the initial guess for relative race displacement is estimated from the default stiffness values available in the internal data base.

appDispY

Relative race displacement (m or in) along the y-axis when kFS2=1 on record 3.3, or initial guess for relative race displacement, along the y-axis, when kFS2=0 on record 3.3.

When kFS2=0 and appDispY is set to zero, the initial guess for relative race displacement is estimated from the default stiffness values available in the internal data base.

appDispZ

Relative race displacement (m or in) along the z-axis when kFS3=1 on record 3.3, or initial guess for relative race displacement, along the x-axis, when kFS3=0 on record 3.3.

When kFS3=0 and appDispZ is set to zero, the initial guess for relative race displacement is estimated from the default stiffness values available in the internal data base.

Record 9.1.2

Applied Moments, Misalignments and Operating Speeds

The operating data supplied on records 9.1.1 and 9.1.2 is used for computing the quasi-static equilibrium solution, which may be used for computing the initial conditions for the dynamic solutions. Any time-dependent operating conditions must be programmed in the optional subroutine Adrx1.

recID

Record identifier - maximum 12 characters in single quotes.

appMomY

Applied moment (N.m or lbf.in) along y-axis when kFS4=0 on Record 3.3.

appMomZ

Applied moment (N.m or lbf.in) along z-axis when kFS5=0 on Record 3.3.

appMis11

Misalignment-y on reference race, rotation about y-axis (rad). Outer race is normally the reference race.

appMis21

Misalignment-z on reference race, rotation about z-axis (rad). Outer race is normally the reference race.

appMis12

Misalignment-y on the moving race, rotation about y-axis (rad), when kFS4=1 on Record 3.3, or initial guess for computing race misalignment when solving race moment equilibrium equation, when kFS4=0 on Record 3.3.

Normally the inner race is displaced relative to the outer for obtaining the equilibrium solution, it is therefore labeled as the moving race.

appMis22

Misalignment-z on the moving race, rotation about z-axis (rad), when kFS5=1 on Record 3.3, or initial guess for computing race misalignment when solving race moment equilibrium equation, when kFS5=0 on Record 3.3.

Normally the inner race is displaced relative to the outer for obtaining the equilibrium solution, it is therefore labeled as the moving race.

rpm1

Angular velocity of outer race (rpm).

rpm2

Angular velocity of the inner race (rpm)

Record 9.2

Parameters for Stiffness Computations

The data record is required only for stiffness computations, kStif > 0 on Record 3.3.

recID

Record identifier - maximum 12 characters in single quotes.

pctDisp

Percent displacement increment for stiffness computation.

rpmRange11

Initial outer race velocity (rpm) in stiffness-speed table.

rpmRange21

Final outer race velocity (rpm) in stiffness-speed table.

rpmRange12

Initial inner race velocity (rpm) in stiffness-speed table.

rpmRange22

Final inner race velocity (rpm) in stiffness-speed table.

Record 9.3

Rotating Loads

This data record is required only for rotating loads, kRotLoad > 0 on Rec 3.4.

Rotating radial loads are simulated by applying a whirl motion to the races, where the race center rotates relative to a fixed point in space with a prescribed velocity. The radius of the whirl orbit is specified as a fraction of the maximum radial displacement resulting from the sum of stationary and rotating load. Thus the initial radial load on Record 9.1.1 must be set equal to the sum of fixed and rotating loads. The figure 31 schematically shows the whirl orbits and the related parameters.

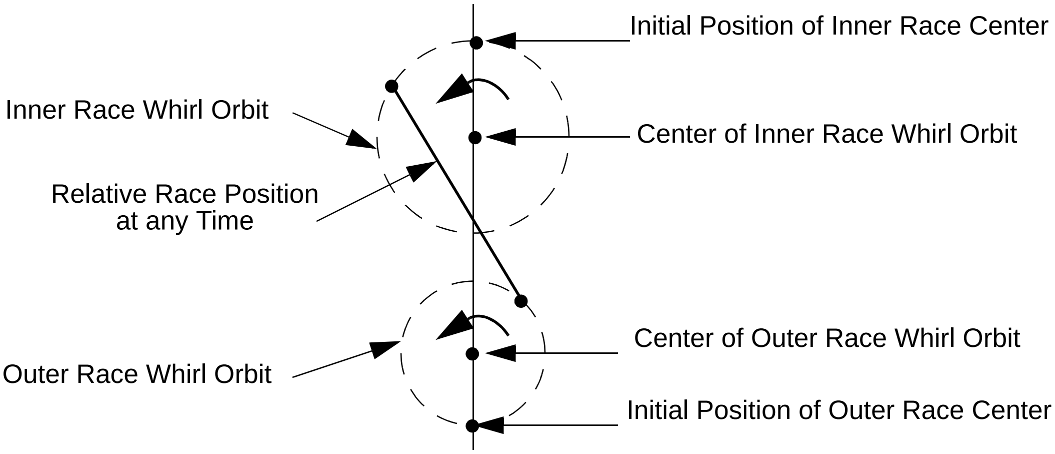

Initial Position of Inner Race Center
Inner Race Whirl Orbit
Center of Inner Race Whirl Orbit
Relative Race Position
at any Time
Center of Outer Race Whirl Orbit
Outer Race Whirl Orbit
Initial Position of Outer Race Center

Figure 31. Rotating load simulation in terms of race mass center orbits.

recID

Record identifier - maximum 12 characters in single quotes.

rotLoadFrac1

Ratio of outer race orbit radius to relative radial deflection the bearing, when a fraction of radial load rotates with the outer race.

rotLoadFrac2

Ratio of inner race orbit radius to relative radial deflection the bearing, when a fraction of radial load rotates with the inner race.

rotLoadRpm1

Rotational speed (rpm) of load rotating with outer race

rotLoadRpm2

Rotational speed (rpm) of load rotating with inner race

Record 9.4

Rotating Reference Frame

This record is required only for rotating reference frames, kRotFrame=1 on Record 3.4.

Normally all equations of motion are written in a space fixed coordinate frame located at the bearing center. However, if the bearing as a whole rotates in space, such as in a crank shaft or a planetary gear, then additional transport and Corioliss components must be added to the equations of motion. Under the rotating reference frame option a simple orbital motion with a constant velocity is simulated as schematically shown below in figure 32. More complicated motion at variable orbit radius and rotating speed may be modeled in the user programmable subroutine Adrx1.

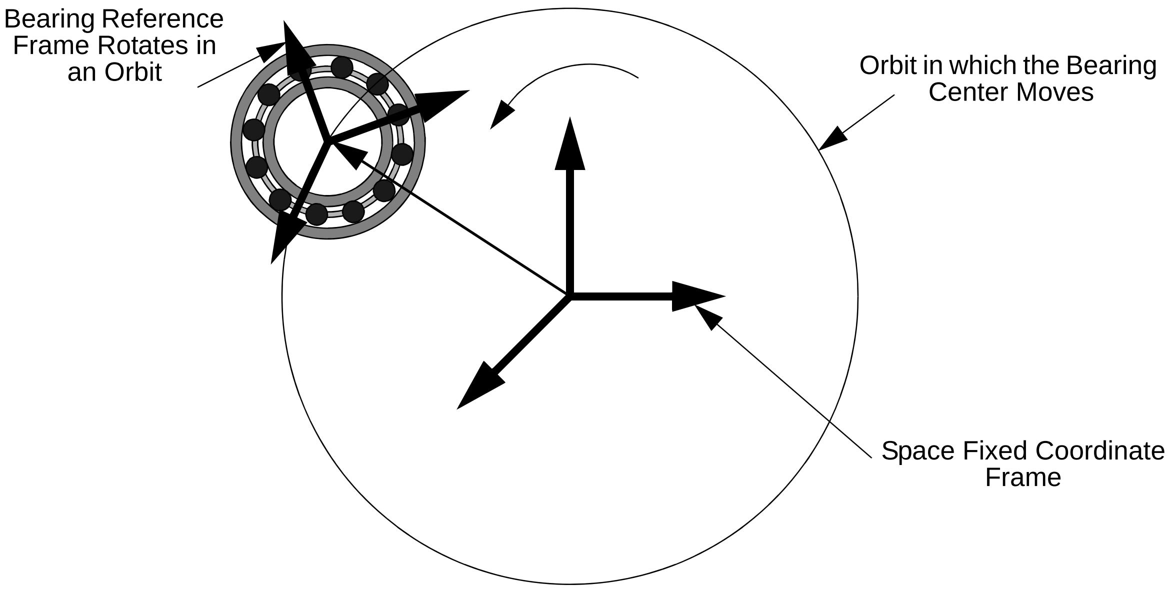

Bearing Reference Frame Rotates in an Orbit
Orbit in which the Bearing Center Moves
Space Fixed Coordinate Frame

Figure 32. Simulation of rotating reference frames.

recID

Record identifier - maximum 12 characters in single quotes.

brgOrbitRad

Radius of orbit (m or in) in which the bearing center travels.

brgAngPos

Initial angular position (deg) of bearing center.

brgAngVel

Angular velocity (rpm) at which the bearing center rotates

brgLoadFrac1

Fraction on the inertial load exerted on the outer race to be supported by the bearing.

brgLoadFrac2

Fraction on the inertial load exerted on the inner race to be supported by the bearing.

Record 9.5.k, k=1,nCseg

Cage Initial Position

Data on this record is required only for bearings with cage, nCseg > 0 on Record 3.2.

This record contains the initial parameters for the cage or cage segment in case of a segmented cage. The data record is repeated for each cage segment.

recID

Record identifier - maximum 12 characters in single quotes.

pocLubFilm(k)

Maximum lubricant film (m or in) in cage pocket.

wvRatio(k)

Ratio of initial cage mass center whirl velocity to cage angular velocity.

avRatio(k)

Ratio of initial cage angular velocity to the epicyclic value.

cageGcPos(k)

Vector of length three containing the initial position (x, y, z, coordinates) of the cage (m or in) mass center relative to the locus of the centers of rolling elements.

cageAngPos(k)

Vector of length three containing three transformation angles (deg) which define initial angular position of the cage relative to the inertial frame.

3.10 Traction and Friction Parameters

Record 10.0

Traction Model Options

This data record is always required.

There are three types of traction models used in ADORE:

1. A hypothetical traction curve defined by four empirical coefficients A, B, C, D:

$$\kappa = (A+Bu)e^{(-C)u} + D$$

where κ is the traction coefficient at slip velocity u.

Normally the traction coefficient at zero slip velocity is zero. Thus, $D = -A$ and the above general equation may be reduced to:

$$\kappa = (A+Bu)e^{(-C)u} - A$$

Thus the model is based on three constitutive constants, A, B, and C, which may be computed by the three conditions shown below in the graphical representation of the traction-slip equation in figure 33.

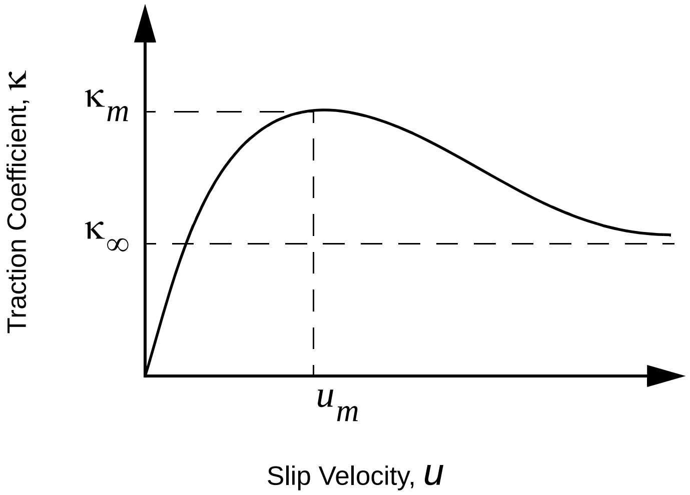


Figure 33. Hypothetical traction-slip relation.

In general, the hypothetical relationship, stated above, may be prescribed in two ways:

(1) Actual values of the coefficients A, B, C, D.

(2) Four conditions which may be used to compute the coefficients.

As an additional simplification, when the coefficient C is set to zero, traction becomes linearly dependent on slip with a slope B. Such a model may be valid under low slip conditions, but traction has to bounded at high slip rates; in other words a continued increase of traction with increasing slip velocities may not be practical. For this practical reason two traction slopes may be used to define the simplified model:

$\kappa = A + Bu$, $u < u_o$, and

$\kappa = A + Bu_o + Cu$, $u > u_o$

Such a simplified model reduces the curve in figure 33 to two straight lines as shown below in figure 34.

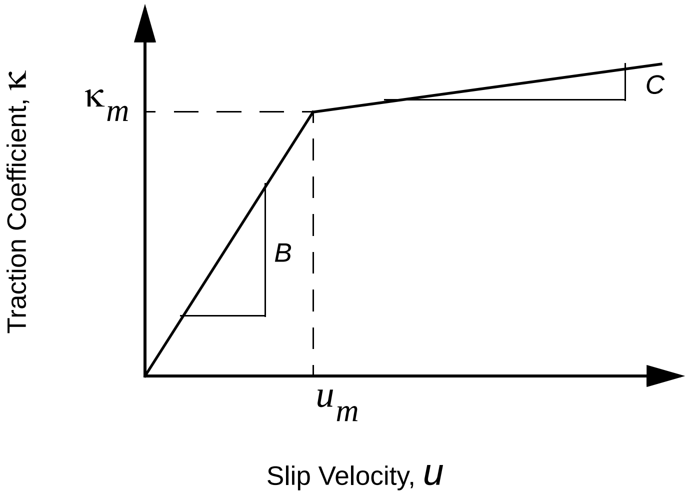


Figure 34. Simplified two-slope traction-slip model.

Note that the constant C, here is simply a slope and it is different from the one discussed earlier. Generally $C « B$. In fact, C may be set to zero when traction is constant at high slip velocities. In addition if B is also set to zero the model reduces to a simple constant traction coefficient.

When traction slope at zero slip is defined and the traction coefficient asymptotes to to a maximum value, the coefficient B may be set to zero, and A, C, D may be computed by three conditions, e.g., traction at zero slip, maximum asymptotic traction at infinite slip, and traction slope at zero slip.

Based on the above discussion, a model type variable, may be associated with the hypothetical traction-slip relation. This model type variable may be assigned three different values to define the following three prescriptions for a hypothetical traction-slip relation:

0 The simplified two slopes model.

1 Four conditions to compute coefficients A, B, C, D.

-1 Coefficients A, B, C, D are directly prescribed.

2 Traction asymptotes to a maximum value with defined slope at zero slip.

2. An elastohydrodynamic model based on the energy equation through the lubricant film and Newtonian behavior of the lubricant:

Energy Equation: $K\frac{\partial^2 T}{\partial z^2} = -\tau\dot{s}$

where K, T, τ and $\dot{s}$ are respectively the thermal conductivity, temperature, shear stress and strain rate, while z is the coordinate direction through the film.

Geometric Compatibility: $\dfrac{\partial u}{\partial z} = \dot{s}(\tau, p, T)$

where u is the slip velocity and the strain rate is a function of shear stress, pressure p, and temperature T.

Constitutive Equation: $\dot{s}(\tau, p, T) = \dfrac{\tau}{\mu(p, T)}$

where the viscosity $\mu(p, T)$ as a function of pressure p, and temperature T, may assume one of the following types of relations:

Type I Relation: $\mu = \mu_o \exp[\alpha p + \beta(T_o - T)]$

Type II Relation: $\mu = \mu_o \exp\left[\alpha p + \beta\left(\dfrac{1}{T} - \dfrac{1}{T_o}\right)\right]$

where α, β and μ_o are respectively the pressure-viscosity coefficient, temperature-viscosity coefficient and reference viscosity at a reference temperature T_o.

At any point in the contact, the energy, geometric compatibility and constitutive equations are solved simultaneously through the film with the prescribed velocities and temperatures at the interacting surfaces and at a given pressure. The slip distribution through the slip provides the strain rate which then leads to computation of temperature and shear stress distribution through the film. The shear stress is noted at the mid plane and the computation is repeated incrementally along the contact length. The computed mid plane shear stress is then integrated to compute overall traction force.

It is once again seen that the model is based on three constitutive constants, α, β and μ_o, which are generally computed by curve fitting experimental traction data to the model described above. When the slip variation along the minor axis of the contact ellipse is ignored, it may be seen that the above model may be implemented essentially in closed form. Thus from computational stand point implementation of this model may be fairly efficient.

3. An elastohydrodynamic model based on visco-elastic behavior of the lubricant:

Shear stress/strain rate equation: $\dot{s} = \dfrac{1}{G}\dfrac{\partial\tau}{\partial t} + \dfrac{\tau_o}{\mu}f\left(\dfrac{\tau}{\tau_o}\right)$

where G, μ and τ_o are respectively the shear modulus, viscosity and critical shear stress of the lubricant. Again, there are three constitutive parameters which define the model.

The shear stress function may either be one of the following two types:

Type I Relation: $f\left(\frac{\tau}{\tau_o}\right) = \operatorname{asinh}\left(\frac{\tau}{\tau_o}\right)$

Type II Relation: $f\left(\frac{\tau}{\tau_o}\right) = \operatorname{atanh}\left(\frac{\tau}{\tau_o}\right)$

Similar to the Newtonian model, lubricant viscosity can again be expressed as a function of pressure and temperature by one of the following two types of relation:

Type I Relation: $\mu = \mu_o \exp[\alpha p + \beta(T_o - T)]$

Type II Relation: $\mu = \mu_o \exp\left[\alpha p + \beta\left(\frac{1}{T} - \frac{1}{T_o}\right)\right]$

Similar to the viscosity variation as a function of pressure and temperature the other two constitutive constants, e.g., G, μ and τ_o may also be functions of pressure and temperature. Again these constitutive constants, and their variation as a function of pressure and temperature, have to be determined experimentally. However, implementation of this model is substantially more complicated since a differential equation has to be solved to compute the shear stress distribution.

Complete analytical details of the elastohydrodynamic models are contained in the following references:

Kannel, J.F. and Walowit, J.A., "Simplified Analysis for Traction Between Rolling-Sliding EHD Contact," ASME Journal of Lubrication Technology, vol 93, 1971, pp 39-46.

Gupta, P.K., Flamand, L., Berthe, D. and Godet, M., "On the Traction Behavior of Several Lubricants," ASME Journal of Lubrication Technology, vol 103, 1981, pp 55-64.

Johnson, K.L. and Tevaarwerk, J.L., ‘Shear Behavior of EHD Oil Films,’ Proceedings of the Royal Society, London, A356, 1977, pp 215.

Bair, S. and Winer, W.O., ‘A Rheological Model for EHD Contacts based on Primary Laboratory Data,’ ASME Journal of Lubrication Technology, vol 101, #3, 1979, pp 258.

Gupta, P.K., Cheng, H.S., Zhu, D., Forster, N.H. and Schrand, J.B., ‘Visco-Elastic Effects in MIL-L-7808 Type Lubricant, Part I: Analytical Formulation,’ STLE Tribology Transactions, vol 35, #2, 1992, pp 269-274.

Forster, N.H., Schrand, J.B., and Gupta, P.K., "Visco-Elastic Effects in MIL-L-7808 Type Lubricant, Part II: Experimental Data Correlations," STLE Tribology Transactions, vol 35, #2, 1992, pp 275-280.

Gupta, P.K., "Visco-Elastic Effects in MIL-L-7808 Type Lubricant, Part III: Model Implementation in Bearing Dynamics Computer Code," STLE Tribology Transactions, vol 35, #4, 1992, pp 724-730.

Hamrock, B.J. and Dowson, D., "Isothermal Elastohydrodynamic Lubrication of Point Contacts, Part III: Fully Flooded Results", ASME Journal of Lubrication Technology, vol 99, #2, 1977, pp 264-276.

Hamrock, B.J. and Dowson, D., Ball Bearing Lubrication: The Elastohydrodynamics of Elliptical Contacts, John Wiley & Sons, 1981.

Dowson, D. and Higginson, G.R., Elastohydrodynamic Lubrication, Paragon Press, 1966.

Wilson, W.R.D. and Sheu, S., "Effect of Inlet Shear Heating Due to Sliding on Elastohydrodynamic Film Thickness," ASME Journal of Lubrication Technology, vol105, 1983, pp 187-188.

Wolveridge, P.E., Baglin, K.P. and Archard, J.F., “The Starved Lubrication of Cylinders in Line Contact,” Proceedings of Institution of Mechanical Engineers, London, Vol 185 81/71, pp 1159-1169.

In addition to any of the above three types of models, an arbitrary traction-slip relation may be programmed in user subroutine Adrx7.

For most oil lubricated bearing the Newtonian model is the most recommended option. For a number of lubricant, the model coefficients are available in the data base built in within ADORE. Thus the task of traction modeling simply reduces to specification of a model code. For the visco-elastic model the user is expected to prescribe all the model coefficients.

For the rolling element to race contact either one of the above model types may be used. However, if an elastohydrodynamic model is selected, a hypothetical model is also prescribed for computing traction when the elastohydrodynamic model breaks down due to the lubricant film thickness being less than the critical value. For all other interactions, such as, rolling element to cage contact, cage/race contact, and contact between roller ends and guide flanges, only a prescribed traction/slip relation may be used to compute traction at a given slip velocity.

The options on Record 10.0 define the model type at the at the various interactions.

recID

Record identifier - maximum 12 characters in single quotes.

kTrac

Traction code at rolling element/race interaction:

-1	Arbitrary traction model in subroutine Adrx7
0	Hypothetical traction-slip model.
1	Mineral oil, SAE 30 or mobil dte.
2	5p4e Polyphenyl ether.

3 MIL-L-7808 type oil.

4 MIL-L-23699 or mobil jet ii.

5 MIL-L-27502 or mcs 1780, a high temp version of 23699.

6 Traction fluid Santotrac 30.

7 Traction fluid Santotrac 50.

8 Visco-Elastic model for the MIL-L-7808 lubricant.

9 Traction model with user defined coefficients.

This case is different from the case kTrac =0 in the sense that the traction/slip behavior is computed by the Newtonian model used under kTrac = 1 to 7, and a visco-elastic model for kTrac = 8, however, the various coefficients of the constitutive equation of the lubricant are supplied by the user on records 10.4.k.

Note that even if kTrac > 0, data for kTrac =0 is still required for use when the elastohydrodynamic traction model breaks down.

kTracType

Hypothetical traction model type at rolling element to race contact:

-1 Coefficients A, B, C, D are directly prescribed.

0 The simplified two slopes model.

1 Four conditions to compute coefficients A, B, C, D.

2 Traction asymptotes to a maximum value with defined slope at zero slip.

See discussion above.

kCPTrac

Rolling element to cage traction model type:

-1 Arbitrary traction model in user subroutine ADRX7.

0 Hypothetical model.

kCPTracType

Hypothetical model type at rolling element to cage contact, when kCPTrac =0:

-1 Coefficients A, B, C, D are directly prescribed.

0 The simplified two slopes model.

1 Four conditions to compute coefficients A, B, C, D.

2 Traction asymptotes to a maximum value with defined slope at zero slip.

See discussion above.

kCRTrac

Cage to race traction model type:

-1 Arbitrary traction model in user subroutine ADRX7.

0 Hypothetical model.

kCRTracType

Hypothetical model type at cage to race contact, when kCRTrac =0:

-1 Coefficients A, B, C, D are directly prescribed.

0 The simplified two slopes model.

1 Four conditions to compute coefficients A, B, C, D.

2 Traction asymptotes to a maximum value with defined slope at zero slip.
See discussion above.

kRFTrac

Race flange to roller traction model type for roller bearings:

-1 Arbitrary traction model in user subroutine ADRX7.

0 Hypothetical model.

kRFTracType

Hypothetical model type at race flange to roller contact, when kRFTrac =0:

0 The simplified two slopes model.

-1 Coefficients A, B, C, D are directly prescribed.

1 Four conditions to compute coefficients A, B, C, D.

2 Traction asymptotes to a maximum value with defined slope at zero slip.
See discussion above.

kRRTrac

Rolling element to rolling element traction model type:

-1 Arbitrary traction model in user subroutine ADRX7.

0 Hypothetical model.

kRRTracType

Hypothetical model type at rolling element to rolling element contact, when kRRTrac =0:

0 The simplified two slopes model.

-1 Coefficients A, B, C, D are directly prescribed.

1 Four conditions to compute coefficients A, B, C, D.

2 Traction asymptotes to a maximum value with defined slope at zero slip.
See discussion above.

Record 10.1A

Rolling Element to Race Contact:
Hypothetical Traction Model Coefficients

This data is required when kTracType = -1 on Record 10.0.

The data specifies the four coefficients, A, B, C, D, of the hypothetical traction-slip relation:

$$= (A + Bu)e^{(-C)u} + I$$

as shown below in figure 35.

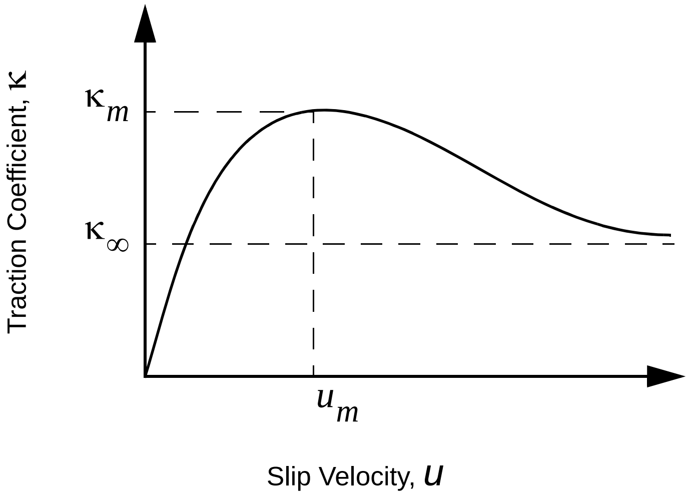


Figure 35. Hypothetical traction-slip relation.

recID

Record identifier - maximum 12 characters in single quotes.

reRaceTC1

Coefficient A in the hypothetical traction relation for rolling element to race contact.

reRaceTC2

Coefficient B (s/m or s/in) in the hypothetical traction relation for the rolling element to race contact.

reRaceTC3

Coefficient C (s/m or s/in) in the hypothetical traction relation for the rolling element to race contact.

reRaceTC4

Coefficient D in the hypothetical traction relation for the rolling element to race contact.

Record 10.1B

Rolling Element to Race Contact:
Coefficients of the Two Slopes Hypothetical Traction Model

This data record is required when kTracType = 0 on Record 10.0.

The data specifies the two slopes and the transition point of the two slopes model, as shown below in figure 36, for the rolling element to race contact.

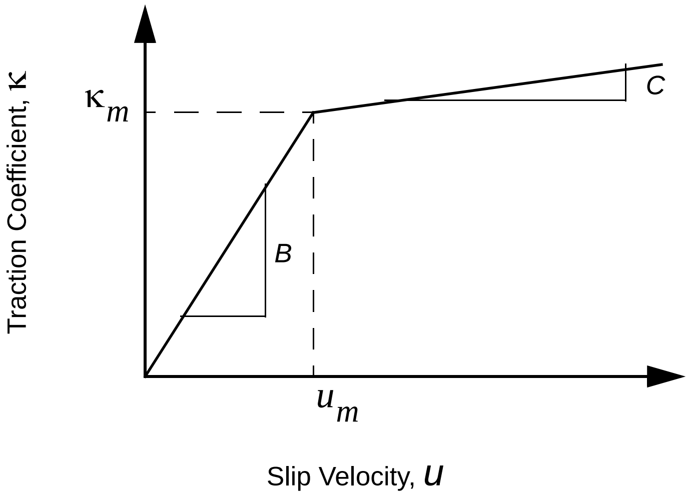


Figure 36. Simplified two slopes traction model.

recID

Record identifier - maximum 12 characters in single quotes.

reRaceTC1

Traction coefficient at zero slip at the rolling element to race contact.

reRaceTC2

Traction/slip slope (s/m or s/in) for $\text{slip} \leq$ reRaceTC4. Slope B in figure 36 above. The transition velocity u_m is specified in variable reRaceTC4 below.

reRaceTC3

Traction/slip slope (s/m or s/in) for $\text{slip} >$ reRaceTC4. Slope C in figure 36 above. The transition velocity u_m is specified in variable reRaceTC4 below.

reRaceTC4

Slip velocity (m/s or in/s) separating the two slopes. Shown as u_m in figure 36 above.

Record 10.1C

Rolling Element to Race Contact:
Conditions for Computing Coefficients of the Hypothetical Traction Model

This data is required when kTracType = 1 on Record 10.0.

The data specifies four conditions from which the coefficients, A, B, C, D, of the hypothetical traction-slip relation may be computed:

$$ = (A + Bu)e^{(-C)u} + I$$

as shown below in figure 37

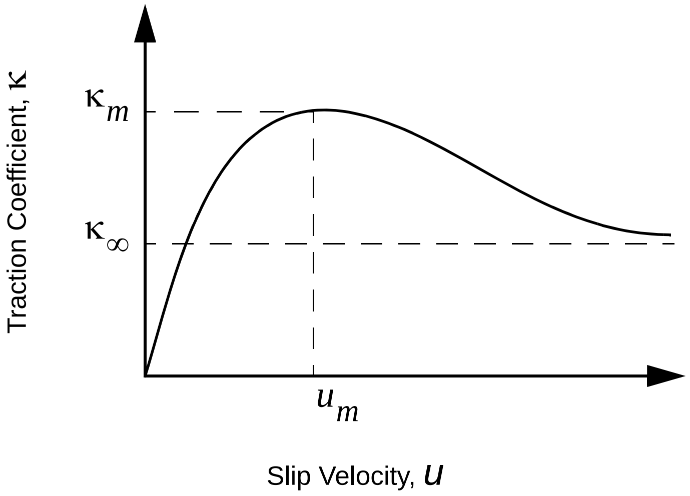


Figure 37. Hypothetical traction-slip relation.

recID

Record identifier - maximum 12 characters in single quotes.

reRaceTC1

Traction coefficient at zero slip for the rolling element to race contact.

reRaceTC2

Maximum traction coefficient at the rolling element to race contact. Labeled as κ_m in figure 37 above.

reRaceTC3

Traction coefficient at infinite slip at the rolling element to race contact. Labeled as κ_∞ in figure 37 above.

reRaceTC4

Slip velocity (m/s or in/s) corresponding to maximum traction. Labeled as u_m in figure 37 above.

Record 10.1D

Rolling Element to Race Contact:
Conditions for Computing Coefficients of the Hypothetical Traction Model

This data is required when kTracType = 2 on Record 10.0.

The data specifies four conditions from which the coefficients, A, C, D , of the hypothetical traction-slip relation may be computed:

$$\kappa = Ae^{(-C)u} + D$$

recID

Record identifier - maximum 12 characters in single quotes.

reRaceTC1

Traction coefficient at zero slip for the rolling element to race contact.

reRaceTC2

Maximum asymptotic traction coefficient at infinite slip for the rolling element to race contact.

reRaceTC3

Traction slope at zero slip at the rolling element to race contact.

reRaceTC4

Presently not used.

Record 10.2A

Rolling Element to Flange Contact:
Hypothetical Traction Model Coefficients

This data record is required for roller bearing with guide flanges, kFlngIndxx > 0 Rec 3.2 and kRFTracType = -1 on Rec 10.0

The data specifies the four coefficients, A, B, C, D, of the hypothetical traction-slip relation:

$$ = (A + Bu)e^{(-C)u} + I$$

as shown below in figure 38.

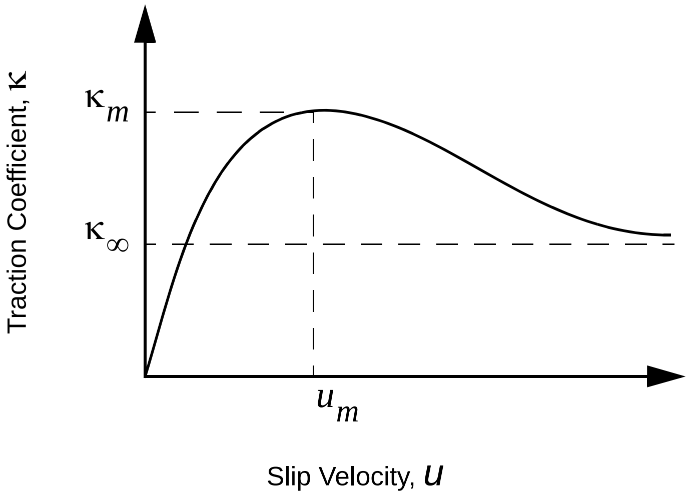


Figure 38. Hypothetical traction-slip relation.

recID

Record identifier - maximum 12 characters in single quotes.

reFlngTC1

Coefficient A in the hypothetical traction relation for rolling element to flange contact.

reFlngTC2

Coefficient B (s/m or s/in) in the hypothetical traction relation for the rolling element to flange contact.

reFlngTC3

Coefficient C (s/m or s/in) in the hypothetical traction relation for the rolling element to flange contact.

reFlngTC4

Coefficient D in the hypothetical traction relation for the rolling element to flange contact.

Record 10.2B

Rolling Element to Flange Contact:
Coefficients of the Two Slopes Hypothetical Traction Model

This data record is required for roller bearing with guide flanges, kFlngIndxx > 0 Rec 3.2 and kRFTracType = 0 on Rec 10.0

The data specifies the two slopes and the transition point of the two slopes model, as shown below in figure 39, for the rolling element to flange contact.

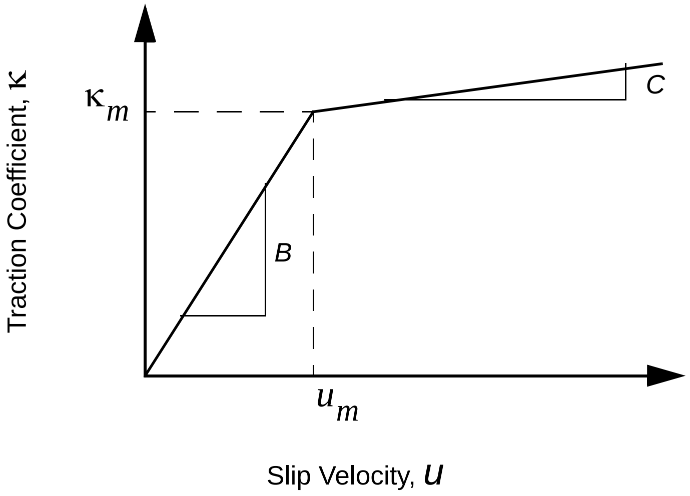


Figure 39. Simplified two-slopes traction model.

recID

Record identifier - maximum 12 characters in single quotes.

reFlngTC1

Traction coefficient at zero slip at the rolling element to flange contact.

reFlngTC2

Traction/slip slope (s/m or s/in) for slip $\leq$ reFlngTC4. Slope *B* in figure 39 above. The transition velocity u_m is specified in variable reFlngTC4 below.

reFlngTC3

Traction/slip slope (s/m or s/in) for slip $>$ reFlngTC4. Slope *C* in figure 39 above. The transition velocity u_m is specified in variable reFlngTC4 below.

reFlngTC4

Slip velocity (m/s or in/s) separating the two slopes. Shown as u_m in figure 39 above.

Record 10.2C

Rolling Element to Flange Contact: Conditions for Computing Coefficients of the Hypothetical Traction Model

This data record is required for roller bearing with guide flanges, kFlngIndxx > 0 Rec 3.2 and kRFTracType = 1 on Rec 10.0

The data specifies four conditions from which the coefficients, A, B, C, D, of the hypothetical traction-slip relation may be computed:

$$= (A + Bu)e^{(-C)u} + I$$

as shown below in figure 40.

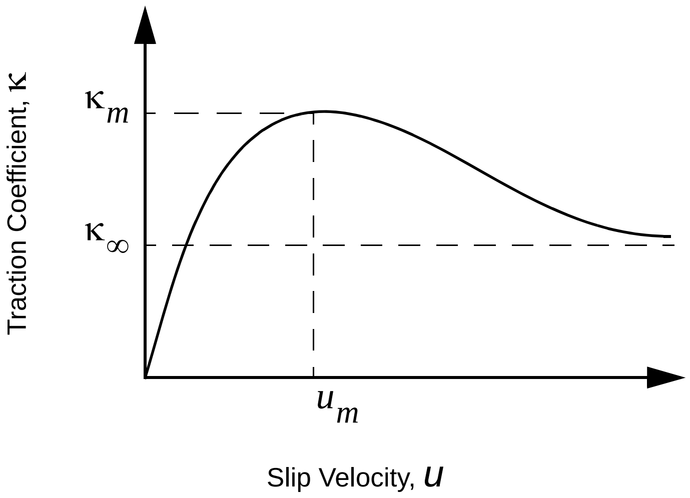


Figure 40. Hypothetical traction-slip relation.

recID

Record identifier - maximum 12 characters in single quotes.

reFlngTC1

Traction coefficient at zero slip for the rolling element to flange contact.

reFlngTC2

Maximum traction coefficient at the rolling element to flange contact. Labeled as κ_m in figure 40 above.

reFlngTC3

Traction coefficient at infinite slip at the rolling element to flange contact. Labeled as κ_∞ in figure 40 above.

reFlngTC4

Slip velocity (m/s or in/s) corresponding to maximum traction. Labeled as u_m in figure 40 above.

Record 10.2D

Rolling Element to Race Flange Contact:
Conditions for Computing Coefficients of the Hypothetical Traction Model

This data is required when kRFTracType = 2 on Record 10.0.

The data specifies four conditions from which the coefficients, A, C, D , of the hypothetical traction-slip relation may be computed:

$$\kappa = Ae^{(-C)u} + D$$

recID

Record identifier - maximum 12 characters in single quotes.

reFlngTC1

Traction coefficient at zero slip for the rolling element to race flange contact.

reFlngTC2

Maximum asymptotic traction coefficient at infinite slip for the rolling element to race flange contact.

reFlngTC3

Traction slope at zero slip at the rolling element to race flange contact.

reFlngTC4

Presently not used.

Record 10.3

Critical Film Thickness and Lubricant Starvation

This data record is required for elastohydrodynamic traction models only, kTrac > 0 on Record 10.0

For lubricated contacts a critical value of film thickness is defined on this record. When the computed actual film thickness is less than this critical value then a metal contact is assumed and the elastohydrodynamic traction model is replaced by a hypothetical model prescribed on record 10.1. Normally this critical film thickness may be set equal to the composite surface roughness of the interacting rolling element and race surfaces, since a majority of surface asperities will be in contact when the film thickness approaches such a value.

Lubricant starvation is modeled by apply a film thickness reduction factor based on semi-empirical formula stated. It is assumed that rather than the whole inlet zone filled with lubricant, the lubricant adheres to the interacting surfaces and forms a meniscus at a definite distance from the contact zone, as shown below in figure 41. The primary input is, therefore, the distance of this meniscus from edge of the contact zone. It is specified as a ratio of actual distance to the contact half width. Normally the contact is fully flooded when this ratio has a value of 10 or more, while it is heavily starved for values 1 or less.

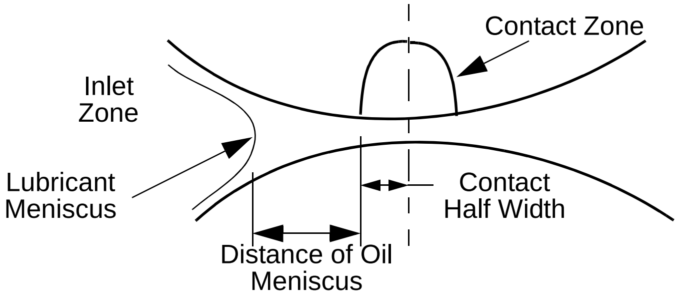


Figure 41. Schematic of an elastohydrodynamic contact.

recID

Record identifier - maximum 12 characters in single quotes.

reRaceFilm

Critical film thickness (m or in) for lubricant model breakdown at rolling element to race interface.

strParam

Starvation parameter. Ratio of the lubricant meniscus distance from the edge of contact to the contact half width.

Record 10.4.0

User Defined Lubricant

This data record is required to prescribed an elastohydrodynamic model for a lubricant which is not present in ADORE data base, kTrac > 8 on Record 10.0.

recID

Record identifier - maximum 12 characters in single quotes.

lubName

Text string (maximum 36 chars) defining lubricant name. This text string is used for documentation purpose only.

Record 10.4.1

Options for Elastohydrodynamic Traction Model

This record is required when an elastohydrodynamic model has to defined for an arbitrary lubricant, kTrac > 8 on Rec 10.0.

Generally there are two types of elastohydrodynamic models:

Newtonian Models: An elastohydrodynamic contact basically consists to two regions: a low pressure region or the inlet zone, and a high-pressure region, where the lubricant shear results in traction. In a Newtonian model the lubricant behavior is defined by a viscosity- pressure-temperature relation. This relation is prescribed for both the low and high pressure regions. The low pressure relation used to compute film thickness; while the high pressure relation used to compute traction.

Visco-elastic Models: Here both the viscous and elastic behaviors of the lubricant are considered. The model is based on three fundamental properties: viscosity, shear modulus, and a critical shear stress which defines the onset of viscous behavior. All these properties may vary with pressure and temperature. This variations must be prescribed for this type of model.

The data on this record defines the desired model and constitutive relation types.

recID

Record identifier - maximum 12 characters in single quotes.

kTracClass

Traction model class defined as:

1 Newtonian elastohydrodynamic model.

2 Visco-elastic model.

3 Traction computations in Adrx7.
Under this option Adrx7 is called to compute traction after computing the film thickness, which is also passed to Adrx7 for use in traction modeling, if necessary.

kMType

Elastohydrodynamic model type:

1 Viscosity varies exponentially with temperature for Newtonian model, kTracClass =1; or the viscous term assumes a sinh function in the visco-elastic model, kTracClass =2.

2 Viscosity varies exponentially with inverse of temperature in the Newtonian model, kTracClass =1; or the viscous term assumes an arctanh function in the visco-elastic model, kTracClass =2.

kVType

Viscosity relation type: For the Newtonian model, this relation type applies in the low-pressure region, which is used to compute the lubricant film thickness, while the relation type entered above, via kMType, is used in the high-pressure region to compute traction.

1 Viscosity varies exponentially with temperature.

2 Viscosity varies exponentially with inverse of temperature.

kGType

The shear modulus relation type. This is applicable only for the visco-elastic models:

1 Shear modulus varies exponentially with temperature.
2 Shear modulus varies exponentially with inverse of temperature.

kSType

Critical shear stress relation type: This is also applicable only for visco-elastic models:

1 Critical shear varies exponentially with temperature.
2 Critical shear varies exponentially with inverse of temperature.

Record 10.4.2

Lubricant Base Properties

The data record is required only for user defined lubricant, kTrac > 8 on Record 10.0.

The base properties specified on this record are used for computing lubricant film thickness for both Newtonian and visco-elastic models, and for establishing the viscosity relation for the visco-elastic models.

For computation of isothermal film thickness the formulae for point and line contact are contained in the following references:

Hamrock, B.J. and Dowson, D., Ball Bearing Lubrication: The Elastohydrodynamics of Elliptical Contacts, John Wiley & Sons, 1981.

Dowson, D. and Higginson, G.R., Elastohydrodynamic Lubrication, Paragon Press, 1966.

After computing the isothermal film thickness a thermal reduction factor is applied to allow for thermal effects. These factors are contained in the following references:

Gupta, P.K., Cheng, H.S., Zhu, D., Forster, N.H. and Schrand, J.B., “Visco-Elastic Effects in MIL-L-7808 Type Lubricant, Part I: Analytical Formulation,” STLE Tribology Transactions, vol 35, #2, 1992, pp 269-274.

Wilson, W.R.D. and Sheu, S., "Effect of Inlet Shear Heating Due to Sliding on Elastohydrodynamic Film Thickness," ASME Journal of Lubrication Technology, vol 105, 1983, pp 187-188.

In addition to thermal reduction factors the film thickness may also be reduced for starvation effects, a factor for which is determined from the following reference:

Wolveridge, P.E., Baglin, K.P. and Archard, J.F., “The Starved Lubrication of Cylinders in Line Contact,” Proceedings of Institution of Mechanical Engineers, London, Vol 185 81/71, pp 1159-1169.

recID

Record identifier - maximum 12 characters in single quotes.

refTemp

Reference temperature (K or R).

refVis

Reference viscosity (Pa.s or lbf.s/in^2).

lubTherCond

Lubricant thermal conductivity (W/m/K or lbf.in/in/R/s).

Record 10.4.3

Coefficients of the Viscosity-Pressure-Temperature Relation

This data record is required to prescribe arbitrary lubricant properties under kTrac > 8 on Record 10.0.

The generalized form of the viscosity-pressure-temperature relation is:

$$\mu = \mu_o \exp[\alpha_1 p + \alpha_2 p^2 + \alpha_3 p^3 + \beta_1 \Delta T + \beta_2 \Delta T^2 + \beta_3 \Delta T^3$$

$$+ \gamma_1 p \Delta T + \gamma_2 p \Delta T^2 + \gamma_3 p \Delta T^3]$$

where ΔT is equal to $(T_o - T)$ or $\left(\frac{1}{T} - \frac{1}{T_o}\right)$ respectively for the Type I (kVType = 1 on Record 10.4.1) or Type II (kVType = 2 on Record 10.4.1)viscosity relation; T_o being respectively the reference temperature. In addition μ_o, α, β, γ are reference viscosity, viscosity-pressure, viscosity-temperature and viscosity-pressure-temperature coefficients respectively.

Normally only one term is used in the above relation. The generalized polynomial relation is retained only for more rigorous modeling if necessary.

The various coefficients are specified on this data record.

recID

Record identifier - maximum 12 characters in single quotes.

visCoeff11

First viscosity-pressure coefficient, α_1 in the above equation (m^2/N or in^2/lbf).

visCoeff21

Second viscosity-pressure coefficient, α_2 in the above equation (m^2/N or in^2/lbf$)^2$.

visCoeff31

Third viscosity-pressure coefficient, α_3 in the above equation (m^2/N or in^2/lbf$)^3$.

visCoeff12

First viscosity-temperature coefficient, β_1 in the above equation, (1/K or 1/R) if kVType =1 or (K or R) if kVType =2.

visCoeff22

Second viscosity-temperature coefficient, β_2 in the above equation, (1/K^2 or 1/R^2) if kVType =1 or (K^2 or R^2) if kVType =2.

visCoeff32

Third viscosity-temperature coefficient, β_3 in the above equation, (1/K^3 or 1/R^3) if kVType =1 or (K^3 or R^3) if kVType =2.

visCoeff13

First viscosity-pressure-temperature coefficient, γ_1 in the above equation, (m^2/N/K or in^2/lbf/R) if kVType =1 or (K.m^2/N or R.in^2/lbf) if kVType =2.

visCoeff23

Second viscosity-pressure-temperature coefficient, γ_2 in the above equation, [(m^2/N/K)^2 or (in^2/lbf/R)^2] if kVType =1 or [(K.m^2/N)^2 or (R.in^2/lbf)^2] if kVType =2.

visCoeff33

Third viscosity-pressure-temperature coefficient, γ_3 in the above equation, [(m^2/N/K)^3 or (in^2/lbf/R)^3] if kVType =1 or [(K.m^2/N)^3 or (R.in^2/lbf)^3] if kVType =2.

Record 10.4.4

Coefficients for Newtonian Model

This data record is required to prescribe arbitrary coefficients for the Newtonian traction model, kVTrac > 8 on Rec 10.0 and kTracClass =1 on Rec 10.4.1

The generalized form of the viscosity-pressure-temperature for traction computation may be written in a form similar to the viscosity relation for ambient pressure conditions discussed earlier:

$$\mu = \mu_o \exp[\alpha_1 p + \alpha_2 p^2 + \alpha_3 p^3 + \beta_1 \Delta T + \beta_2 \Delta T^2 + \beta_3 \Delta T^3$$

$$+ \gamma_1 p \Delta T + \gamma_2 p \Delta T^2 + \gamma_3 p \Delta T^3]$$

where ΔT is equal to $(T_o - T)$ or $\left(\frac{1}{T} - \frac{1}{T_o}\right)$ respectively for the Type I (kVType = 1 on Record 10.4.1) or Type II (kVType = 2 on Record 10.4.1) viscosity relation; T_o being respectively the reference temperature. In addition μ_o, α, β, γ are reference viscosity, viscosity-pressure, viscosity-temperature and viscosity-pressure-temperature coefficients respectively.

Once again, only one term is generally used in the above relation. The generalized polynomial relation is only retained for more rigorous modeling if necessary. Also, note that although the symbols in the above equation are identical to that used in the low pressure viscosity relation, the actual coefficient here are quite different.

The various coefficients are specified on this data record.

recID

Record identifier - maximum 12 characters in single quotes.

refTracVis

Reference viscosity for traction computation (N.s/m^2 or lbf.s/in^2).

tracVisCoeff11

First Viscosity-pressure coefficient, α_1 in the above equation (m^2/N or in^2/lbf).

tracVisCoeff21

Second Viscosity-pressure coefficient, α_2 in the above equation (m^2/N or in^2/lbf)^2.

tracVisCoeff31

Third Viscosity-pressure coefficient, α_3 in the above equation (m^2/N or in^2/lbf)^3.

tracVisCoeff12

First Viscosity-temperature coefficient, β_1 in the above equation, (1/K or 1/R) if kVType =1 or (K or R) if kVType =2.

tracVisCoeff22

Second Viscosity-temperature coefficient, β_2 in the above equation, (1/K^2 or 1/R^2) if kVType =1 or (K^2 or R^2) if kVType =2.

tracVisCoeff32

Third Viscosity-temperature coefficient, β_3 in the above equation, (1/K^3 or 1/R^3) if kVType =1 or (K^3 or R^3) if kVType =2.

tracVisCoeff13

First Viscosity-pressure-temperature coefficient, γ_1 in the above equation, (m^2/N/K or in^2/lbf/R) if kVType =1 or (K.m^2/N or R.in^2/lbf) if kVType =2.

tracVisCoeff23

Second Viscosity-pressure-temperature coefficient, γ_2 in the above equation, [(m^2/N/K)^2 or (in^2/lbf/R)^2] if kVType =1 or [(K.m^2/N)^2 or (R.in^2/lbf)^2] if kVType =2.

tracVisCoeff33

Third Viscosity-pressure-temperature coefficient, γ_3 in the above equation, [(m^2/N/K)^3 or (in^2/lbf/R)^3] if kVType =1 or [(K.m^2/N)^3 or (R.in^2/lbf)^3] if kVType =2.

Record 10.4.5

Shear Modulus Relation for Visco-elastic Model

This data record is required for a visco-elastic traction model, kTrac > 8 on Rec 10.0 and kTracClass =2 on Rec 10.4.1

The three basic lubricant properties used to model visco-elastic effects in a lubricant are viscosity, shear modulus and a critical shear stress. These properties may, in general. vary with pressure and temperature. The viscosity relations have already been prescribed on record 10.4.3. A relation for shear modulus is the subject of this data record.

A general equation for the variation of shear modulus with pressure and temperature is written as a polynomial:

$$G = G_o\left(1 + \sum_{i=1}^{2} \alpha_{gi}\Delta P^i\right)\left(1 + \sum_{i=1}^{2} \beta_{gi}\Delta T^i\right)$$

where G is the shear modulus at any pressure P and temperature, T, while α_{gi} and β_{gi} are the pressure and temperature coefficients for the shear modulus variation. $\Delta P = (P - P_o)$ is a pressure differential measured relative to a reference pressure, P_o. where ΔT is equal to $(T_o - T)$ or $\left(\frac{1}{T} - \frac{1}{T_o}\right)$ respectively for the Type I (kGType = 1 on Record 10.4.1) or Type II (kGType = 2 on Record 10.4.1) viscosity relation; T_o being respectively the reference temperature and G_o being the shear modulus at the reference pressure and temperature.

recID

Record identifier - maximum 12 characters in single quotes.

refTG

Reference temperature, T_o (K or R).

refPG

Reference pressure, P_o (N/m^2 or lbf/in^2).

refShearMod

Reference shear modulus G_o (N/m^2 or lbf/in^2).

shearModCoeff11

First shear modulus-pressure coefficient, α_{g1} (m^2/N or in^2/lbf).

shearModCoeff21

Second shear modulus-pressure coefficient α_{g2} [(m^2/N)^2 or (in^2/lbf)^2].

shearModCoeff12

First shear modulus-temperature coefficient, β_{g1}, (1/K or 1/R) when kGType = 1 or (K or R) when kGType = 2 on Record 10.4.1

shearModCoeff22

Second shear modulus-temperature coefficient, β_{g2}, (1/K^2 or 1/R^2) when kGType = 1 or (K^2 or R^2) when kGType = 2 on Record 10.4.1

Record 10.4.6

Critical Shear Stress Relation for Visco-elastic Model

This data record is required for a visco-elastic traction model, kTrac > 8 on Rec 10.0 and kTracClass =2 on Rec 10.4.1

The three basic lubricant properties used to model visco-elastic effects in a lubricant are viscosity, shear modulus and a critical shear stress. These properties may, in general. vary with pressure and temperature. The viscosity and shear modulus relations have already been prescribed on records 10.4.3 and 10.4.5 respectively. A relation for critical shear stress is the subject of this data record.

A general equation for the variation of critical shear stress with pressure and temperature is written as a polynomial:

$$S = S_o\left(1 + \sum_{i=1}^{2} \alpha_{si}\Delta P^i\right)\left(1 + \sum_{i=1}^{2} \beta_{si}\Delta T^i\right)$$

where S is the critical shear stress at any pressure P and temperature, T, while α_{si} and β_{si} are the pressure and temperature coefficients for the critical shear stress variation.

$\Delta P = (P - P_o)$ is a pressure differential measured relative to a reference pressure, P_o.

where ΔT is equal to $(T_o - T)$ or $\left(\frac{1}{T} - \frac{1}{T_o}\right)$ respectively for the Type I (kSType = 1 on Record 10.4.1) or Type II (kSType = 2 on Record 10.4.1) viscosity relation; T_o being respectively the reference temperature and S_o being the critical shear stress at the reference pressure and temperature.

recID

Record identifier - maximum 12 characters in single quotes.

refTG

Reference temperature, T_o (K or R).

refPG

Reference pressure, P_o (N/m^2 or lbf/in^2).

refCritShear

Reference critical shear stress S_o (N/m^2 or lbf/in^2).

critShearCoeff11

First critical shear stress-pressure coefficient, α_{s1} (m^2/N or in^2/lbf).

critShearCoeff21

Second critical shear stress-pressure coefficient α_{s2} [(m^2/N)^2 or (in^2/lbf)^2].

critShearCoeff12

First critical shear stress-temperature coefficient, β_{s1}, (1/K or 1/R) when kSType = 1 or (K or R) when kSType = 2 on Record 10.4.1

critShearCoeff22

Second critical shear stress-temperature coefficient, β_{s2}, (1/K^2 or 1/R^2) when kSType = 1 or (K^2 or R^2) when kSType = 2 on Record 10.4.1.

Record 10.5.1A

Rolling Element to Cage Contact:
Hypothetical Traction Model Coefficients

This data record is required when a cage is present in the bearing, nCseg > 0 Rec 3.2 and kCPTracType = -1 on Record 10.0

The data specifies the four coefficients, A, B, C, D, of the hypothetical traction-slip relation:

$$= (A + Bu)e^{(-C)u} + I$$

as shown below in figure 42.

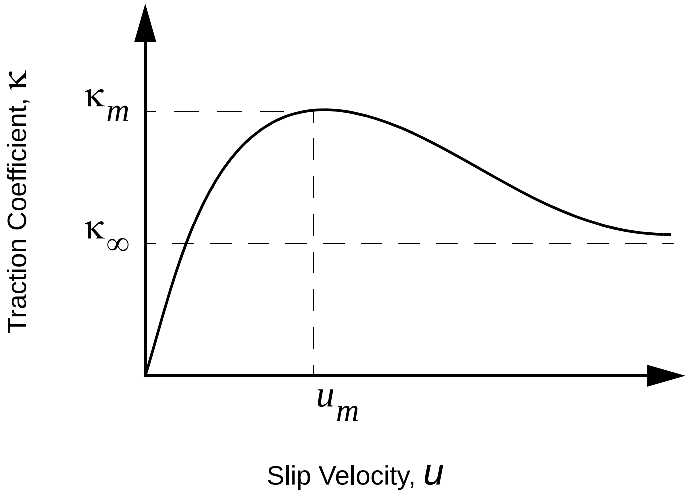


Figure 42. Hypothetical traction-slip relation.

recID

Record identifier - maximum 12 characters in single quotes.

reCageTC1

Coefficient A in the hypothetical traction relation for rolling element to cage contact.

reCageTC2

Coefficient B (s/m or s/in) in the hypothetical traction relation for the rolling element to cage contact.

reCageTC3

Coefficient C (s/m or s/in) in the hypothetical traction relation for the rolling element to cage contact.

reCageTC4

Coefficient D in the hypothetical traction relation for the rolling element to cage contact.

Record 10.5.1B

Rolling Element to Cage Contact:
Coefficients of the Two Slopes Hypothetical Traction Model

This data record is required when a cage is present in the bearing, nCseg > 0 Rec 3.2 and kCPTracType = 0 on Record 10.0

The data specifies the two slopes and the transition point of the two slopes model, as shown below in figure 43, for the rolling element to cage contact.

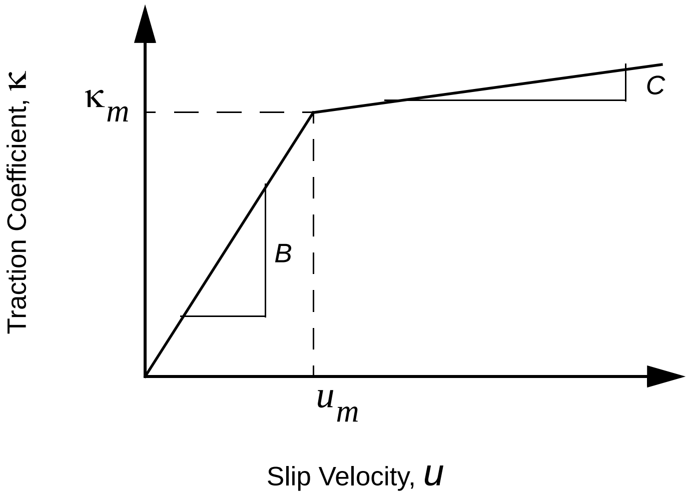


Figure 43. Simplified two-slopes traction model.

recID

Record identifier - maximum 12 characters in single quotes.

reCageTC1

Traction coefficient at zero slip at the rolling element to cage contact.

reCageTC2

Traction/slip slope (s/m or s/in) for $\mathrm{slip} \leq$ reCageTC4. Slope B in figure 43 above. The transition velocity u_m is specified in variable reCageTC4 below.

reCageTC3

Traction/slip slope (s/m or s/in) for $\mathrm{slip} >$ reCageTC4. Slope C in figure 43 above. The transition velocity u_m is specified in variable reCageTC4 below.

reCageTC4

Slip velocity (m/s or in/s) separating the two slopes. Shown as u_m in figure 43 above.

Record 10.5.1C

Rolling Element to Cage Contact:
Conditions for Computing Coefficients of the Hypothetical Traction Model

This data record is required when a cage is present in the bearing, nCseg > 0 Rec 3.2 and kCPTracType = 1 on Record 10.0.

The data specifies four conditions from which the coefficients, A, B, C, D, of the hypothetical traction-slip relation may be computed:

$$= (A + Bu)e^{(-C)u} + I$$

as shown below in figure 44.

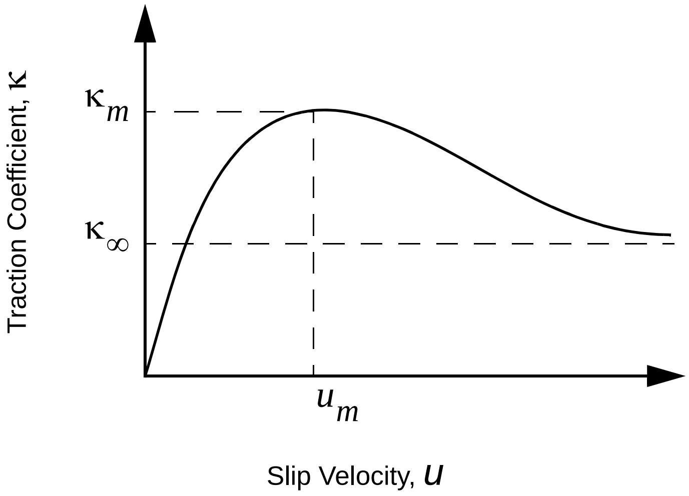


Figure 44. Hypothetical traction-slip relation.

recID

Record identifier - maximum 12 characters in single quotes.

reCageTC1

Traction coefficient at zero slip for the rolling element to cage contact.

reCageTC2

Maximum traction coefficient at the rolling element to cage contact. Labeled as κ_m in figure 44 above.

reCageTC3

Traction coefficient at infinite slip at the rolling element to cage contact. Labeled as κ_∞ in figure 44 above.

reCageTC4

Slip velocity (m/s or in/s) corresponding to maximum traction. Labeled as u_m in figure 44 above.

Record 10.5.1D

Rolling Element to Cage Contact:
Conditions for Computing Coefficients of the Hypothetical Traction Model

This data record is required when a cage is present in the bearing, nCseg > 0 Rec 3.2 and kCPTracType = 1 on Record 10.0.

The data specifies four conditions from which the coefficients, A, C, D, of the hypothetical traction-slip relation may be computed:

$$\kappa = Ae^{(-C)u} + D$$

recID

Record identifier - maximum 12 characters in single quotes.

reCageTC1

Traction coefficient at zero slip for the rolling element to cage contact.

reCageTC2

Maximum asymptotic traction coefficient at infinite slip for the rolling element to race flange contact.

reCageTC3

Traction slope at zero slip at the rolling element to race flange contact.

reCageTC4

Presently not used.

Record 10.5.2A

Cage to Race Contact:
Hypothetical Traction Model Coefficients

This data record is required when a race guided cage is present in the bearing, nCseg > 0 Record 3.2, iCageGuide(i) > 0 on Record 7.0 and kCRTracType = -1 on Record 10.0

The data specifies the four coefficients, A, B, C, D, of the hypothetical traction-slip relation:

$$= (A + Bu)e^{(-C)u} + I$$

as shown below in figure 45.

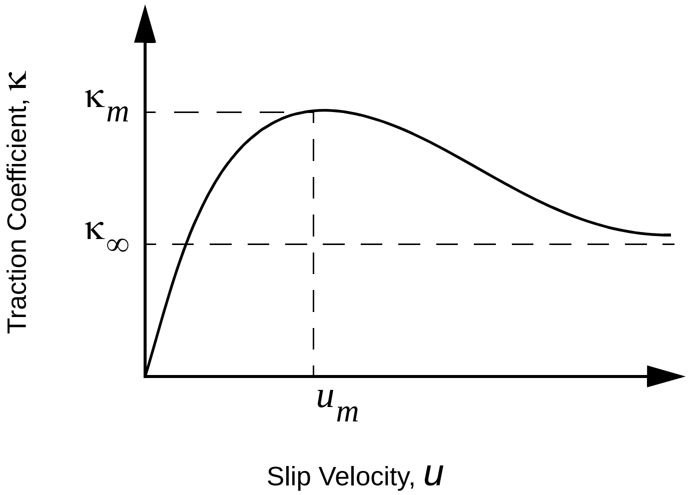


Figure 45. Hypothetical traction-slip relation.

recID

Record identifier - maximum 12 characters in single quotes.

cageRaceTC1

Coefficient A in the hypothetical traction relation for cage to race contact.

cageRaceTC2

Coefficient B (s/m or s/in) in the hypothetical traction relation for the cage to race contact.

cageRaceTC3

Coefficient C (s/m or s/in) in the hypothetical traction relation for the cage to race contact.

cageRaceTC4

Coefficient D in the hypothetical traction relation for the cage to race contact.

Record 10.5.2B

Cage to Race Contact: Coefficients of the Two Slopes Hypothetical Traction Model

This data record is required when a race guided cage is present in the bearing, nCseg > 0 Record 3.2, iCageGuide(i) > 0 on Record 7.0 and kCRTracType = 0 on Record 10.0

The data specifies the two slopes and the transition point of the two slopes model, as shown below in figure 46, for the Cage to Race contact.

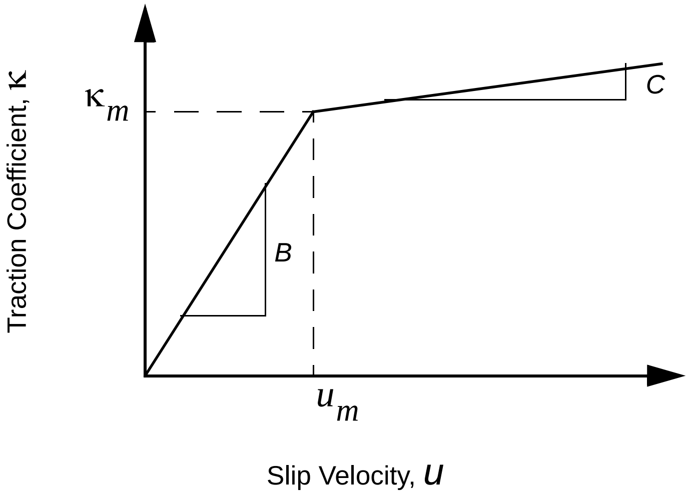


Figure 46. Simplified two-slopes traction model.

recID

Record identifier - maximum 12 characters in single quotes.

cageRaceTC1

Traction coefficient at zero slip at the cage to race contact.

cageRaceTC2

Traction/slip slope (s/m or s/in) for slip ≤ cageRaceTC4. Slope *B* in figure 46 above. The transition velocity u_m is specified in variable cageRaceTC4 below.

cageRaceTC3

Traction/slip slope (s/m or s/in) for slip > cageRaceTC4. Slope *C* in figure 46 above. The transition velocity u_m is specified in variable cageRaceTC4 below.

cageRaceTC4

Slip velocity (m/s or in/s) separating the two slopes. Shown as u_m in figure 46 above.

Record 10.5.2C

Cage to Race Contact:
Conditions for Computing Coefficients of the Hypothetical Traction Model

This data record is required when a race guided cage is present in the bearing, nCseg > 0 Record 3.2, iCageGuide(i) > 0 on Record 7.0 and kCRTracType = 1 on Record 10.0.

The data specifies four conditions from which the coefficients, A, B, C, D, of the hypothetical traction-slip relation may be computed:

$$ = (A + Bu)e^{(-C)u} + I$$

as shown below in figure 47.

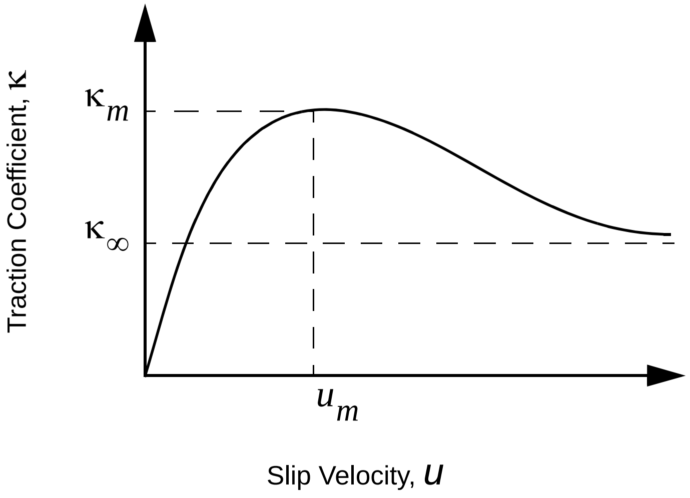


Figure 47. Hypothetical traction-slip relation.

recID

Record identifier - maximum 12 characters in single quotes.

cageRaceTC1

Traction coefficient at zero slip for the cage to race contact.

cageRaceTC2

Maximum traction coefficient at the cage to race contact. Labeled as κ_m in figure 47 above.

cageRaceTC3

Traction coefficient at infinite slip at cage to race contact. Labeled as κ_∞ in figure 47 above.

cageRaceTC4

Slip velocity (m/s or in/s) corresponding to maximum traction. Labeled as u_m in figure 47 above.

Record 10.5.2D

Cage to Race Contact:
Conditions for Computing Coefficients of the Hypothetical Traction Model

This data record is required when a cage is present in the bearing, nCseg > 0 Rec 3.2 and kCRTracType = 1 on Record 10.0.

The data specifies four conditions from which the coefficients, A, C, D, of the hypothetical traction-slip relation may be computed:

$$\kappa = Ae^{(-C)u} + D$$

recID

Record identifier - maximum 12 characters in single quotes.

cageRaceTC1

Traction coefficient at zero slip for the cage to race flange contact.

cageRaceTC2

Maximum asymptotic traction coefficient at infinite slip for the cage to race contact.

cageRaceTC3

Traction slope at zero slip at the cage to race contact.

cageRaceTC4

Presently not used.

Record 10.5.3A

Rolling Element to Rolling Element Contact:
Hypothetical Traction Model Coefficients

Data on this record is presently used only for ball bearings.

This data record is required for cageless bearings, nCseg = 0 Record 3.2 and kRRTracType = -1 on Record 10.0

The data specifies the four coefficients, A, B, C, D, of the hypothetical traction-slip relation:

$$= (A + Bu)e^{(-C)u} + I$$

as shown below in figure 48.

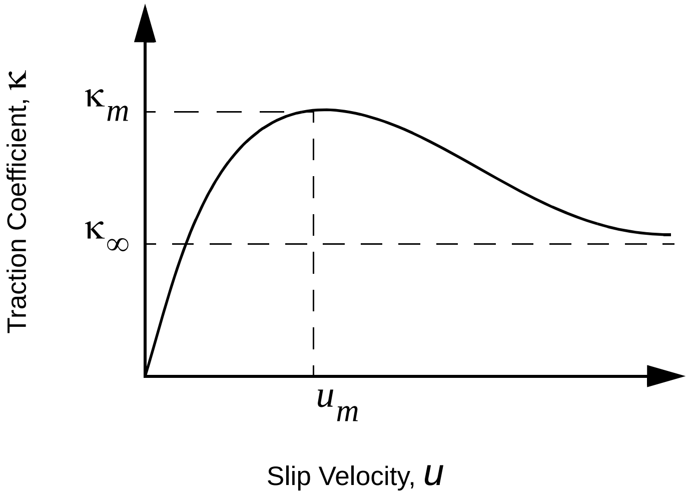


Figure 48. Hypothetical traction-slip relation.

recID

Record identifier - maximum 12 characters in single quotes.

reReTC1

Coefficient A in the hypothetical traction relation for rolling element to rolling element contact.

reReTC2

Coefficient B (s/m or s/in) in the hypothetical traction relation for the rolling element to rolling element contact.

reReTC3

Coefficient C (s/m or s/in) in the hypothetical traction relation for the rolling element to rolling element contact.

reReTC4

Coefficient D in the hypothetical traction relation for the rolling element to rolling element contact.

Record 10.5.3B

Rolling Element to Rolling Element Contact:
Hypothetical Traction Model Coefficients

Data on this record is presently used only for ball bearings.

This data record is required for cageless bearings, nCseg = 0 Record 3.2 and kRRTracType = 0 on Record 10.0

The data specifies the two slopes and the transition point of the two slopes model, as shown below in figure 49, for the rollling element to rolling element contact.

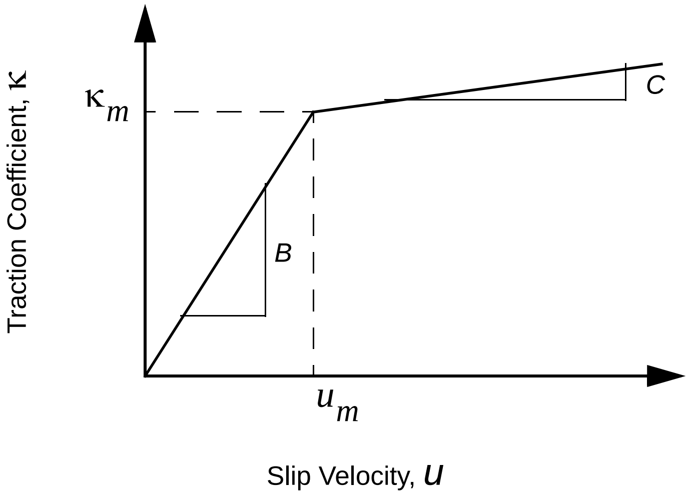


Figure 49. Simplified two-slopes traction model.

recID

Record identifier - maximum 12 characters in single quotes.

reReTC1

Traction coefficient at zero slip at the rolling element to rolling element contact.

reReTC2

Traction/slip slope (s/m or s/in) for $\text{slip} \leq$ reReTC4. Slope *B* in figure 49 above. The transition velocity u_m is specified in variable reReTC4 below.

reReTC3

Traction/slip slope (s/m or s/in) for $\text{slip} >$ reReTC4. Slope *C* in figure 49 above. The transition velocity u_m is specified in variable reReTC4 below.

reReTC4

Slip velocity (m/s or in/s) separating the two slopes. Shown as u_m in figure 49 above.

Record 10.5.3C

Rolling Element to Rolling Element Contact: Hypothetical Traction Model Coefficients

Data on this record is presently used only for ball bearings.

This data record is required for cageless bearings, nCseg = 0 Record 3.2 and kRRTracType = 1 on Record 10.0

The data specifies four conditions from which the coefficients, A, B, C, D, of the hypothetical traction-slip relation may be computed:

$$= (A + Bu)e^{(-C)u} + I$$

as shown below in figure 50.

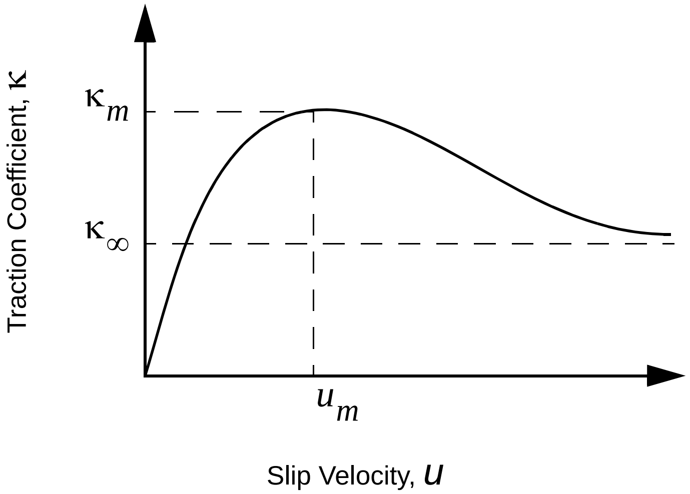


Figure 50. Hypothetical traction-slip relation.

recID

Record identifier - maximum 12 characters in single quotes.

reReTC1

Traction coefficient at zero slip for the rolling element to rolling element contact.

reReTC2

Maximum traction coefficient at the rolling element to rolling element contact. Labeled as κ_m in figure 50 above.

reReTC3

Traction coefficient at infinite slip at rolling element to rolling element contact. Labeled as κ_∞ in figure 50 above.

reReTC4

Slip velocity (m/s or in/s) corresponding to maximum traction. Labeled as u_m in figure 50 above.

Record 10.5.3D

Rolling Element to Rolling Element Contact:
Conditions for Computing Coefficients of the Hypothetical Traction Model

This data record is required for cageless bearings, nCseg = 0 Record 3.2 and kRRTracType = 1 on Record 10.0

The data specifies four conditions from which the coefficients, A, C, D, of the hypothetical traction-slip relation may be computed:

$$\kappa = Ae^{(-C)u} + D$$

recID

Record identifier - maximum 12 characters in single quotes.

reReTC1

Traction coefficient at zero slip for the rolling element to rolling element contact.

reReTC2

Maximum asymptotic traction coefficient at infinite slip for the rolling element to rolling element contact.

reReTC3

Traction slope at zero slip at the rolling element to rolling element contact.

reReTC4

Presently not used.

Record 10.6

Cage Pocket and/or Land Hydrodynamics

This record is required when a hypothetical traction model is prescribed at the rolling element to race contact, kTrac ≤ 0 on Record 10.0, and modeling of hydrodynamic effects in either the cage pocket or the cage/race guide lands is required, kPocHydro or kGsHydro $\neq 0$ on Record 7.0.

In absence of an elastohydrodynamic model there is no lubricant property data available. Thus oil properties are required to model hydrodynamics. This record specifies these required properties.

recID

Record identifier - maximum 12 characters in single quotes.

pocVis

Effective lubricant viscosity (N.s/m^2 or lbf.s/in^2) for hydrodynamic interaction in cage pockets.

gsVis

Effective lubricant viscosity (N.s/m^2 or lbf.s/in^2) for hydrodynamic interaction at the cage/race interface.

Record 10.7A

Churning and Drag Parameters

This record is required when modeling of churning and drag effects is required with kChrn = 2 on Record 3.3.

Very simple models based on conventional laminar and turbulent flows are used in ADORE to model churning and drag models effects. When the bearing is only partly filled with oil, it is assumed that the actual media is a uniform mixture of oil and air. The effective density is the volume average density. Since density of oil is negligible compared to that of the oil, the effective density is simply equal to oil density multiplied by the fraction of bearing cavity filled with oil. For shearing effects the effective viscosity may simply be set equal to viscosity of oil.

The models used and the various churning and drag coefficient are contained in the following references:

Rumbarger, J.H., Filetti, E.G. and Gubernick, D., “Gas turbine engine main shaft roller bearing system analysis.”, ASME Journal of Lubrication Technology, vol 95, pp 401-416, 1973.

Schlichtig, H., BOUNDARY LAYER THEORY, MCGRAW HILL, PP 15-19, 606-108, 1968.

The required effective density and viscosity are prescribed on this record.

recID

Record identifier - maximum 12 characters in single quotes.

chrnDen

Effective churning media density (kgm/m3 or lbm/in3) for churning effects.

chrnVis

Effective churning media viscosity (N.s/m^2 or lbf.s/in^2) for churning.

Record 10.7B

Churning and Drag Parameters

This record is required when modeling of churning and drag effects is required with kChrn = 1 or > 2 on Record 3.3.

Very simple models based on conventional laminar and turbulent flows are used in ADORE to model churning and drag models effects. When the bearing is only partly filled with oil, it is assumed that the actual media is a uniform mixture of oil and air. The effective density is the volume average density. Since density of oil is negligible compared to that of the oil, the effective density is simply equal to oil density multiplied by the fraction of bearing cavity filled with oil. For shearing effects the effective viscosity may simply be set equal to viscosity of oil.

The models used and the various churning and drag coefficient are contained in the following references:

Rumbarger, J.H., Filetti, E.G. and Gubernick, D., “Gas turbine engine main shaft roller bearing system analysis.”, ASME Journal of Lubrication Technology, vol 95, pp 401-416, 1973.

Schlichtig, H., BOUNDARY LAYER THEORY, MCGRAW HILL, PP 15-19, 606-108, 1968.

The required effective density and viscosity are prescribed on this record as a ratio of the base values contained in the ADORE data base for the selected churning media by the parameter kChrn specified on record 3.3.

recID

Record identifier - maximum 12 characters in single quotes.

denRatio

Ratio of effective density to base density of churning media, as specified by value of kChrn on Record 3.3.

visRatio

Ratio of effective viscosity to base viscosity of churning media, as specified by value of kChrn on Record 3.3.

3.11 Gravity Effects

Record 11

Gravity Effects

This record is only required for dynamic simulations, mode ≥ 0 on Record 1.

Gravity effects are modeled by simply adding the weights of the various bearing elements to the applied force vectors in the prescribed direction.

This record prescribes the acceleration due to gravity vector in the inertial frame of reference shown below in figure 48.

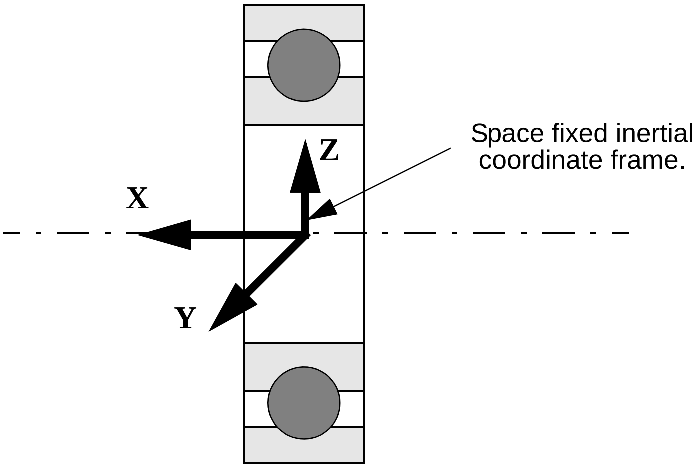


Figure 51. Base inertial coordinate system.

recID

Record identifier - maximum 12 characters in single quotes.

gravityVecX

Component of gravity vector (m/s^2 or in/s^2) in X direction.

gravityVecY

Component of gravity vector (m/s^2 or in/s^2) in Y direction.

gravityVecZ

Component of gravity vector (m/s^2 or in/s^2) in Z direction.

3.12 Inputs for User Programmable Routines

Records 12.1 to 12.n

Inputs for User Programmable Subroutines

These records are required when optional inputs are programmed in the user subroutines.The data format must conform to the optional codes in user programmable subroutines Adrx1 to Adrx9.

4. ADORE OUTPUT

Due to the extensive amount of data a significant effort is devoted to the organization and control of the output from ADORE. Both print and plot outputs are provided and the size of the output can be greatly controlled by exercising the output control options in the input to ADORE.

4.1 Print Output

Typical print outputs from ADORE for ball and cylindrical roller bearings are contained in the software media under subdirectory Disk1 (see Media Contents in Chapter 2 of this manual). The first few pages of the output consists of a listing of all the input data records, bearing geometry, material properties, inertial parameters, lubrication parameters, initial operating conditions, scale factors and output controls. Most of this data is essentially input to ADORE. The translational and rotational constraints listed under initial operating conditions correspond to the specification of either a force or an acceleration, as discussed in the preceding section. The six components listed under the translational constraints represent the outer and inner race constraints along the (X,Y,Z) axes. The first three components are for the outer race while the latter three are for the inner race. Rotational constraints are specified only along the X-axis and the two components printed correspond to the outer and inner races. Along the Y and Z axes the constraint switch is always set to one, meaning that only angular accelerations resulting in rotating or time-varying misalignments can be prescribed along these axes. The data control parameters listed under output control just denote the variables kPrtOpt and kPrtFreq specified on input record 1, while the auto plot codes denote the bearing elements (array kPltElemInd of input record 3.4) for which the plot data is stored.

The print output, at each time step, is divided into four sections:

1. Rolling Element Parameters
2. Race and Cage Parameters
3. Applied Parameters
4. Time Step Summary

Any or all of the output sections may be printed at any preselected time steps by appropriately exercising the output control options. Although most of the output is self-explanatory, a brief discussion of some of the parameters may be helpful to the user.

4.1.1 Angular Velocities

All angular velocity vectors are printed in terms of an amplitude and the orientation θ and ϕ. The two angles define the orientation of the angular velocity vectors as shown in figure 49. The (X,Y,Z) coordinate frame shown in this figure corresponds to the rolling element azimuth frame (with Z axis pointed radially outward, X axis along the bearing axis and Y axis determined by the right hand screw rule) in the case of rolling element velocities and it represents the inertial frame for the cage and race angular velocities.

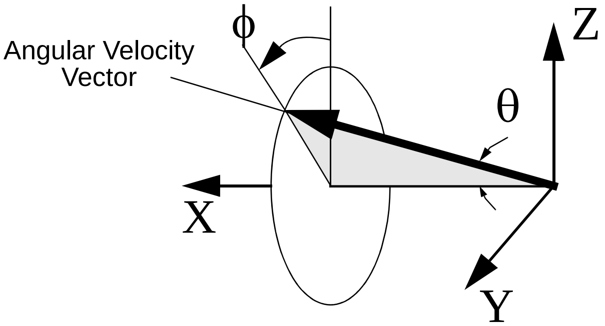


Figure 52. Rolling element angular velocity vector in the azimuth coordinate frame.

4.1.2 Angular Positions

The angular position of any bearing element is defined as the orientation of the principal axis of inertia (X) in a certain coordinate frame. The coordinate frame used is the azimuth frame for rolling elements and the inertial frame for the cage and races. Similar to the angular velocity vector the body fixed principal axis of inertia (X) is located by the two angles θ and ϕ, as shown below in figure 50.

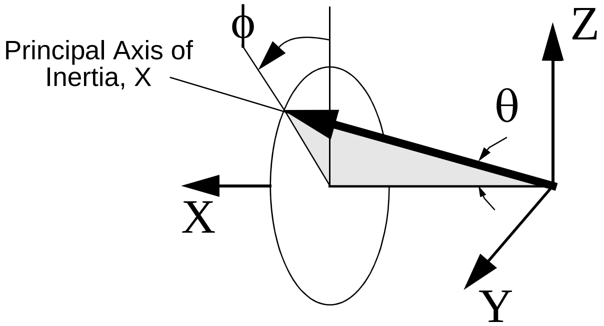


Figure 53. Rolling element orientation in the azimuth coordinate frame.

4.1.3 Rolling Element Contact Depth & Chordal Distance

For ball bearings the extent of contact on the race is defined by locating the depth of outer contact edge relative to race shoulder, *s*, and the semi-chordal distance of inner edge of contact, *t*, as shown below in figure 54. These parameters are derived by simple geometrical relation between race geometry and contact angle. If r_g is the radius of the race groove curvature center locus, r_s is the shoulder radius, α is the contact angle, f is the race curvature ratio, *D* is the ball diameter and *a* is the major contact half width and the relations for *s* and *t* are simply written as:

$$s = fD\left(\cos\alpha + \frac{a}{fD}\right) + r_g - r_s$$

$$t = fD\sin\left(\alpha - \frac{a}{fD}\right)$$

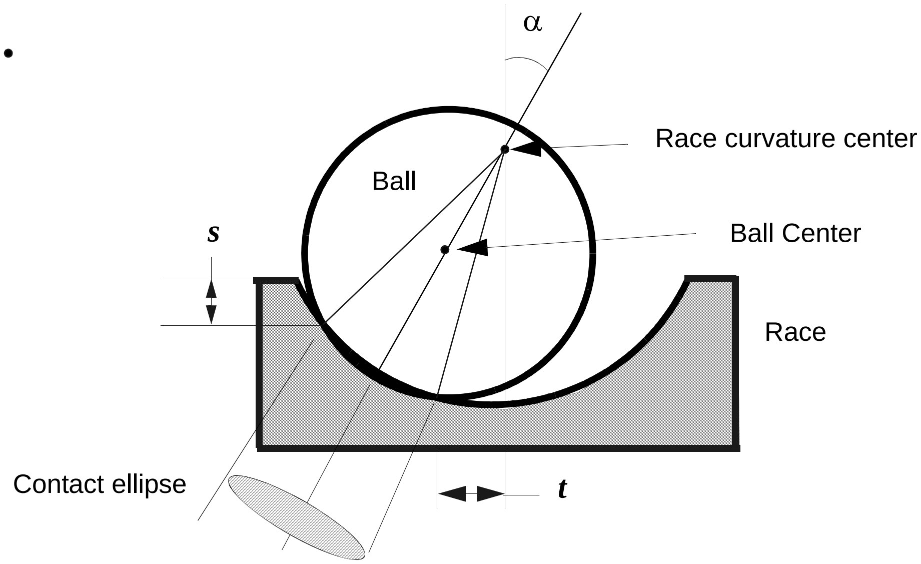


Figure 54. Position of contact edge relative to race geometry.

4.1.4 Time Averaged Wear Rates

The wear rate at any interaction is computed by the well known Archard's wear equation:

$$w(t) = K\frac{Q(t)V(t)}{H}$$

where $w(t)$ is the volumetric wear rate at any instant of time *t*, *Q(t)* and *V(t)* are respectively the instantaneous contact load and sliding velocity functions, *K* is the wear coefficient, and *H* is hardness of the material.

Since for rolling element/race contacts the sliding velocities and loads may greatly vary over the contact zone, the product *QV* is replaced by an integral of the load-slip product over the contact zone. Also, for these interactions the wear is divided between the races and the rolling elements according to the prescribed wear coefficients.

For the rolling element/cage and cage/race interactions all the wear is assumed to occur on the cage. This is quite reasonable since in most cases the material of the cage will be softer than that of the rolling elements or the races.

Since all loads and sliding velocities are functions of time, the wear rates also vary with time and any instantaneous value of the wear rate has little practical significance. The wear rates are, therefore, time-averaged over the time of bearing performance simulation. As the bearing reaches a steady-state condition, these time-averaged wear rates tend to assume fairly constant values. Thus, subject to the uncertainty in the wear coefficients, these average rates may be used to compute wear in a bearing over extended times. Also, if any mechanical interactions in the bearing progressively increase with time, as in the case of gross instabilities, these time-averaged quantities develop a definite positive gradient with respect to time. These rates are, therefore, also useful in identification of instabilities of bearing elements. Clearly, such an interpretation of the results is completely insensitive to the actual value of the wear coefficient used, since the wear coefficient is simply a multiplier in the equation of time-averaged wear rate which is written as

$$W(T) = \frac{K}{H}\int_{0}^{T} Q(t)V(t)dt$$

where T is the time of performance simulation.

4.1.5 Rolling Element/Cage Contact Angle

This output variable denotes the angular position of rolling element/cage interaction in a cage pocket coordinate frame, as shown in figure 52. The rolling element drives the cage when the contact angle is 180 degrees and the cage drives the rolling element if the contact angle is zero. Clearly, the contact angle can be anywhere from zero to 360 degrees for a ball bearing but for a roller bearing it will only be either zero or 180 degrees.

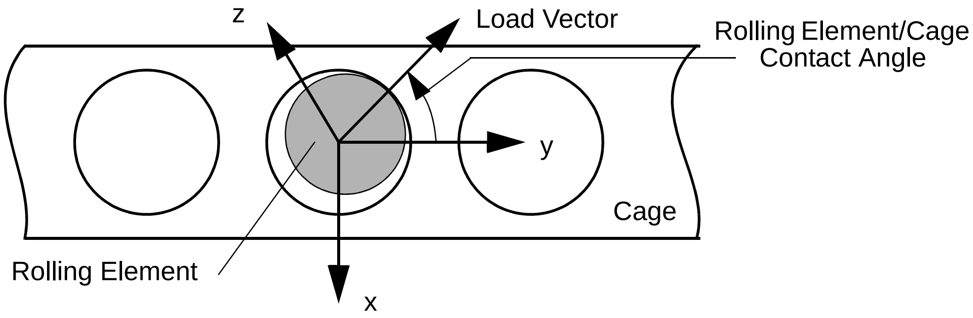


Figure 55. Ball/Cage contact angle for cylindrical pockets.

In the case of spherical pockets the contact position is defined by two angles, q and f, as defined below in figure 53.

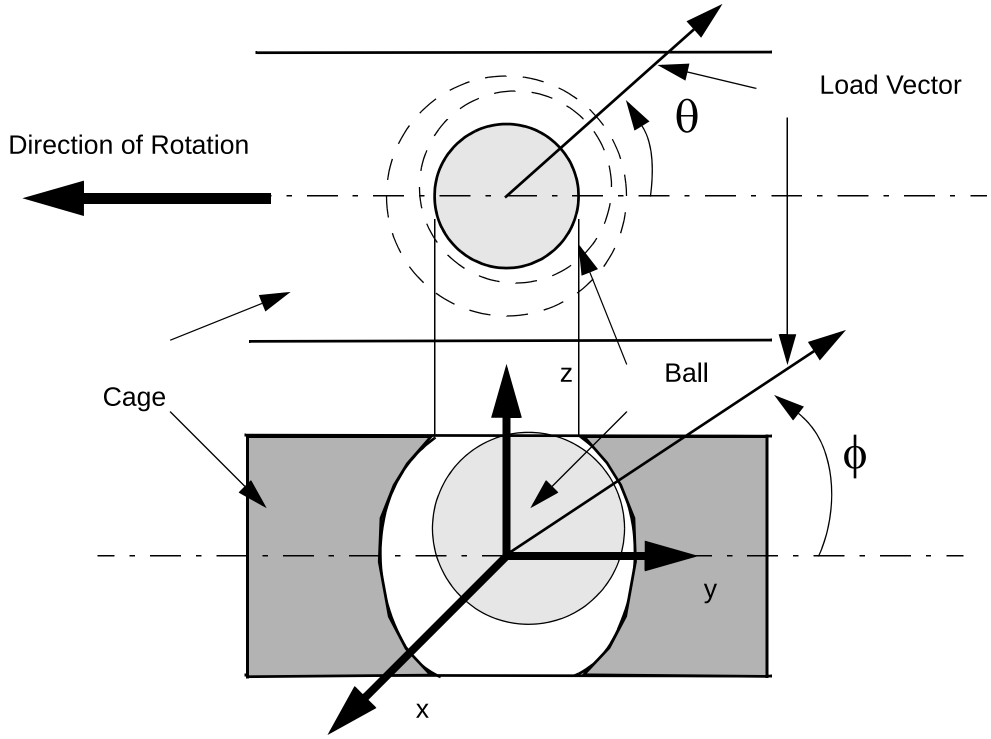


Figure 56. Ball/Cage contact angles for spherical pockets.

The contact angles for roller bearings are significantly easier to define the pocket surfaces on which the rollers contact are generally flat. For example, for a cylindrical roller bearing with rectangular pockets, the contact angle will either be zero or 180^{o}.

4.1.6 Cage/Race Contact and Attitude Angles

The cage/race contact angle defines the angular position of the cage/race contact in the cage fixed coordinate frame as shown below in figure 54. The attitude angle is only relevant when the hydrodynamics at the cage/race interface is considered. It essentially denotes the angle between the line of minimum clearance and the hydrodynamic load.

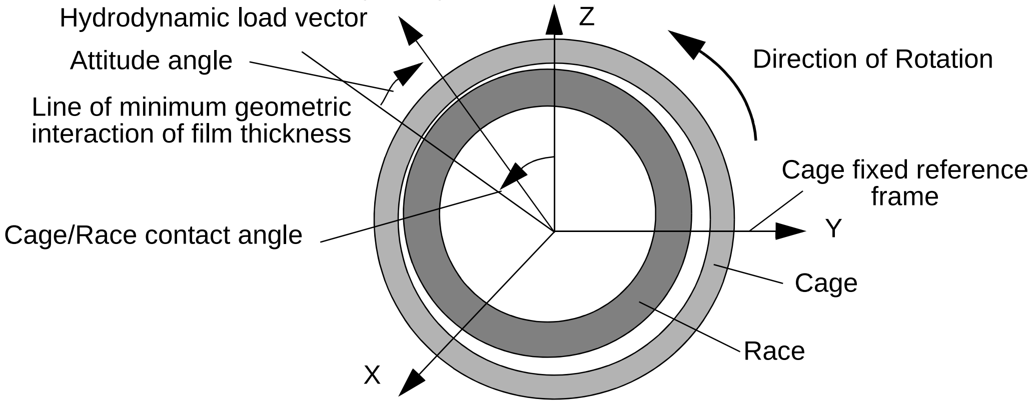


Figure 57. Schematic of cage/race contact and attitude angles.

4.1.7 Power Loss

The frictional dissipation at each interaction is computed and printed in the print output. A sum of all these losses and loss due to lubricant churning and drag is printed out as the total power loss. The fraction of this loss due to churning and drag effects is indicated as the churning loss fraction.

4.1.8 Internal Clearance and Operating Fits

The internal bearing clearance, outer and inner race fits and the cage/race effective diametral play denote actual operating values after allowing for thermal and centrifugal growths.

4.1.9 Fatigue Life

ADORE provides two values for bearing fatigue life; a basic life and a modified life. The basic life is computed from the actual load distribution and contact geometry at the various rolling element/race contacts. using the well accepted fatigue life constants for conventional bearing steels. The modified life results after application of various life modifying factors for both subsurface and surface effects. Some of the effects considered include, bulk material defects, hardness factors, surface roughness, asperity traction, and lubrication effects. Again, the various factors are computed by certain default values of the pertinent parameters. For special materials, the use has the option to specify all material properties and parameters used in the life calculation algorithm.

4.1.10 Rolling Element Orbital Velocity Ratio

This variable is essentially a ratio of the rolling element orbital velocity to the angular velocity of the inner race relative to the outer. The value printed in section 4 of the print output denotes an average over all rolling elements and the time over which the performance simulation is obtained.

4.1.11 Cage Angular Velocity Ratio

This parameter is the ratio of the angular velocity of the cage to the angular velocity of the inner race relative to the outer. In the case of a segmented cage the value printed in the step summary represents an average over all segments and over the time of integration.

4.1.12 Cage Whirl Ratio

The ratio of the mass center whirl velocity to the angular velocity of the inner race relative to the outer race is denoted by this output variable. Again an average is computed over the time of integration and all the cage segments if the cage is segmented.

4.2 Plot Output

In view of the large amount of output generated by ADORE, the plot output is essential in determining the general dynamic behavior of the bearing. Normally the output data is stored in pertinent data files during the run and later input to available plotting programs to display the plots. ADORE plot facility is a platform independent Java based application. The plot output is divided into four sets.

1. Power Dissipation and Life
2. Rolling Element Motion

3. Cage Motion
4. Race Motion

There are a number of plots in each set and under default conditions all the plots in the dta set are displayed over 2,500 steps. This maximum number of steps can be interactively changed if the number of steps in the simulation is larger or if the plots are required over a smaller number of steps to see the solutions in more detail. Like wise the desired plots can be interactively selected. The various plots and variables plotted in each data set are discussed in the following sections:

4.2.1 Power Dissipation and Life

There are four plots in this set:

Plot #1 - Overall Power Loss and Life

Power Loss: The total heat generated in the bearing at all interactions in included in this variable. In addition to all concentrated contacts, such as rolling element to race and cage contacts with the rolling elements and race, the energy dissipated in churning and drag is also included.

Churning Contribution: Fraction of total power loss dissipated in churning and drag in included in this variable.

Fatigue Life: Algorithms used in ADORE for computing fatigue life assume that the applied loads at the various contacts exist indefinitely. Thus variations in fatigue life do not have any physical significance. The life value plotted at any instant of time represents the computed life at that instant with the assumption and the load conditions are static and they exists indefinitely.

Plot #2 - Applied Moments

Applied Moment -> X-Comp: Sum of all moment exerted on the outer and inner races along the bearing axis is included in this variable. Note this variable is simply one component of exerted moment, and a multiplication of this moment by the race angular velocity may not give the total power loss in the bearing.

Applied Moment -> Y-Comp: Similar to the first variable on this plot, this variable contains the moment component along the transverse y axis.

Applied Moment -> Z-Comp: This variable represents the applied moment component along z axis. Normally the z-axis is along the applied radial load. is included in this variable.

Plot #3 - Time-Averaged Wear Rates

Time averaged wear rates of the form

$$W(T) = \frac{K}{H}\int_0^T Q(t)V(t)dt$$

are included in this plot for each of the bearing elements. If the values for wear coefficient, K and material hardness are realistic, then these average rates may be used to estimate wear over a given time. Note that the wear coefficient and hardness are simply constants, thus the plotted results may be prorated to make adjustments of other wear coefficient and hardness ratios. The quantity under the integral sign has additional practical significance in the sense that if either the loads and sliding velocity at any individual contacts increase in an unbounded fashion then these integrals will demonstrate a positive slope and they will not converge to a well defined steady-state value as a function of time. Thus the plotted rates are good indicator of catastrophic instabilities.

Rolling Element #1: Total time-averaged wear rate of rolling element #1 due to contact with the outer and inner races and the cage.

Races: Time-averaged wear rates of the outer and inner races due to contacts with rolling elements and the cage.

Cage: Cage time-averaged wear rates due to all contacts with the rolling elements and the races.

Plot #4 - Bulk Temperatures

The estimated bulk temperatures of the bearing elements resulting from all thermal interactions are included in this plot. Since ADORE does not model thermal transients changes to the geometric dimensions as a function of changing temperatures are applied in a stepwise fashion. Thus the temperature variation show a step-wise pattern. Under stable conditions, however, this step wise pattern will normally converge to a steady value. A divergent pattern, on the other hand, will represent a thermal instability.

Rolling Element: Estimated bulk temperature of the rolling elements.

Races: Estimated bulk temperature of the outer and inner races.

Cage: Estimated bulk temperature of the cage or cage segment.

4.2.2 Rolling Element Motion

Plot #1 - Rolling Element Accelerations

ORBITAL: Orbital angular acceleration of rolling element

RADIAL: Radial acceleration of rolling element mass center. Under constrained mode, mode > 0 or input Record 1, this component is set to zero.

AXIAL: Axial acceleration of rolling element mass center. Under constrained mode, mode > 0 or input Record 1, this component is set to zero.

Plot #2 - Rolling Element Velocity

ORBITAL: Orbital angular velocity of rolling element

RADIAL: Radial velocity of rolling element mass center. Under constrained mode, mode > 0 or input Record 1, this component is set to zero.

AXIAL: Axial velocity of rolling element mass center. Under constrained mode, mode > 0 or input Record 1, this component is set to zero.

Plot #3 - Rolling Element Position

ORBITAL: Orbital angular position of rolling element

RADIAL: Radial position of rolling element mass center. Under constrained mode, mode > 0 or input Record 1, this component is constant.

AXIAL: Axial position of rolling element mass center. Under constrained mode, mode > 0 or input Record 1, this component is constant.

Plot #4 - Rolling Element Angular Orientation

Angular orientation of the rolling element is defined by three angles, rotation about the principal polar axis of inertia (axis X) and orientation of this axis in the rolling element azimuth frame, defined by two angles, θ and ϕ, as follows:

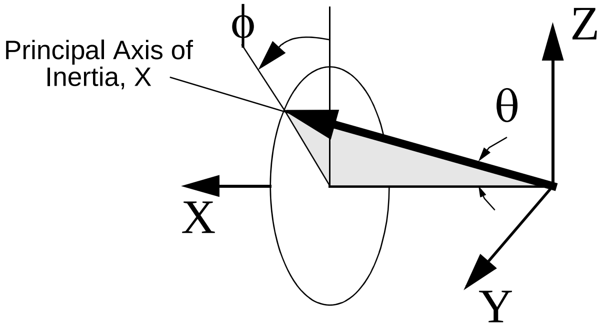


Figure 58. Rolling element orientation in the azimuth coordinate frame.

THETA: Angle θ defining orientation of rolling element principal axis X.

PHI: Angle ϕ defining orientation of rolling element principal axis X.

ROTATION: Rotation of rolling element about the principal X-axis.

Plot #5 - Rolling Element Angular Velocity

Angular velocity of the rolling element is defined by its magnitude and orientation of the angular velocity vector in the rolling element azimuth frame, defined by two angles, θ and ϕ, as follows:

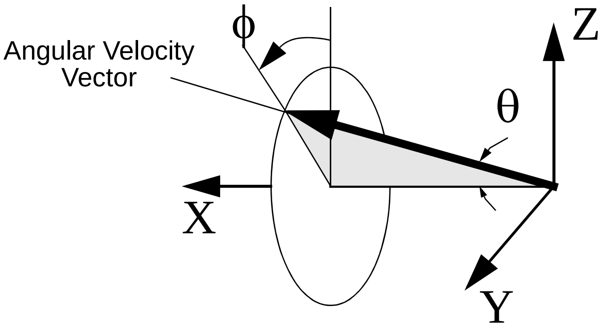


Figure 59. Rolling element angular velocity vector in the azimuth coordinate

MAGNITUDE: Magnitude of rolling element angular velocity vector.

THETA: Angle θ defining orientation of rolling element angular velocity vector.

PHI: Angle ϕ defining orientation of rolling element angular velocity vector.

Plot #6 - Rolling Element / Race Interactions - Set #1

Rolling element to race contact loads, contact angles and spin-to-roll ratios are the subject of this plot.

CCONTACT LOAD: Contact loads at the outer and inner race contacts.

CONTACT ANGLE: Contact angles at the outer and inner race contacts.

SPIN/ROLL: Spin-to-Roll ratios at the outer and inner races. Spin velocity is defined as the component of the rolling element angular velocity vector, relative to the race and normal to the contact plane, while roll velocity is the relative angular velocity component in the plane of contact.

Plot #7 - Rolling Element / Race Interactions - Set #2

SLIP VEL: Maximum slip velocity in the rolling element to race contact. Slip velocity is defined as the relative sliding between the rolling element and race.

Q*V: Integral of the product of load and slip velocity in the contact.

LUB FILM: Lubricant film thickness in the rolling element to race contact.

Plot #8 - Rolling Element / Race Interactions - Set #3

HEAT GEN: Contact heat generation is the integral of the product of slip velocity and traction force in the rolling element to race contact.

CON TEMP RISE: Rise in temperate in the contact as a result of thermal interaction.

RACE CON TEMP: Contact temperature at the rolling element to race contact.

Plot #9 - Rolling Element / Outer Race Flange Interactions

This plot is only active for roller bearings with guide flanges on the outer race.

NOR LOAD: Normal contact load between the roller corner and the outer race flange.

GEO INT: Geometric interaction between the roller corner and the outer race flange. Geometric interaction is defined as clearance between the interacting roller and flange contact. A negative value of this clearance indicates contact.

HEAT GEN: Local heat generated at the roller and flange interface at the outer race contact.

Plot #10 - Rolling Element / Inner Race Flange Interactions

This plot is only active for roller bearings with guide flanges on the inner race.

NOR LOAD: Normal contact load between the roller corner and the inner race flange.

GEO INT: Geometric interaction between the roller corner and the inner race flange. Geometric interaction is defined as clearance between the interacting roller and flange contact. A negative value of this clearance indicates contact.

HEAT GEN: Local heat generated at the roller and flange interface at the inner race contact.

4.2.3 Cage Motion

Plot #1 - Cage Mass Center Velocities

WHIRL RATIO: Whirl velocity represents the angular velocity of cage mass center about the bearing center. The WHIRL RATIO is ratio of this angular velocity to the angular velocity of the rotating race. In the event both races are rotating then the higher of the two velocities is used as the base velocity.

RADIAL: Radial component of the cage mass center velocity.

AXIAL: Axial component of the cage mass center velocity.

Plot #2 - Cage/Race Interaction at Guide Land #1

NOR FORCE: Cage/Race normal contact force at guide land #1.

GEO INT: Geometric interaction at guide land #1. Geometric interaction represents the clearance on contact deflection at the interacting cage and race surfaces. A negative value of GEO INT represents contact while a positive value represents clearance.

CONTACT ANGLE: Angular position of cage/race contact or geometric interaction, on guide land #1, in a cage fixed coordinate frame as shown below:

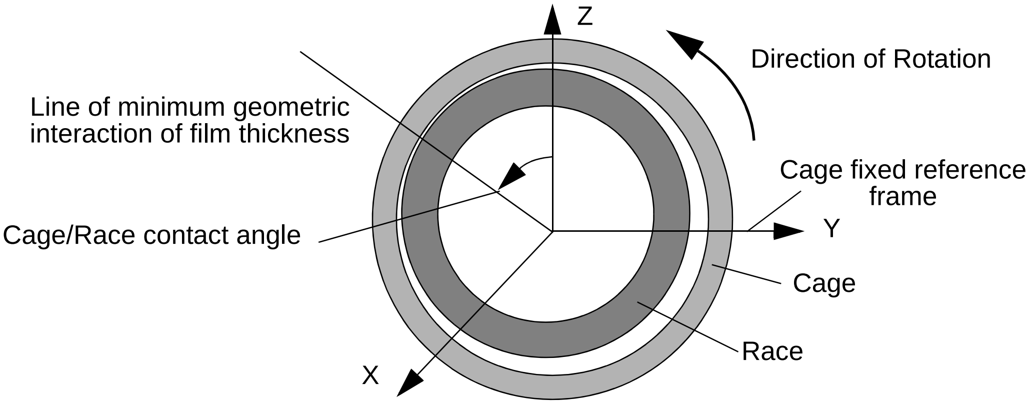

Z
Direction of Rotation
Line of minimum geometric interaction of film thickness
Cage fixed reference frame
Y
Cage/Race contact angle
Cage
Race
X

Figure 60. Schematic of cage/race contact angle.

Plot #3 - Cage/Race Interaction at Guide Land #2

NOR FORCE: Cage/Race normal contact force at guide land #2.

GEO INT: Geometric interaction at guide land #2. Geometric interaction represents the clearance on contact deflection at the interacting cage and race surfaces. A negative value of GEO INT represents contact while a positive value represents clearance.

CONTACT ANGLE: Angular position of cage/race contact or geometric interaction, at guide land #2, in a cage fixed coordinate frame as shown below:

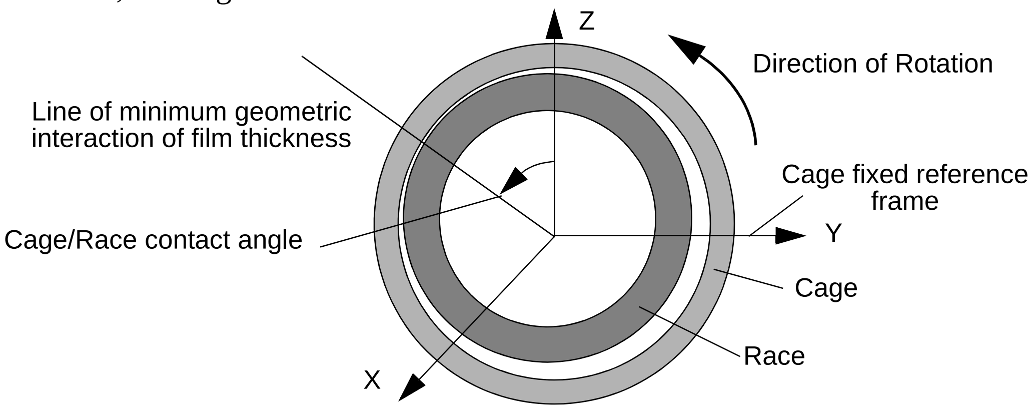

Z
Direction of Rotation
Line of minimum geometric interaction of film thickness
Cage fixed reference frame
Y
Cage/Race contact angle
Cage
Race
X

Figure 61. Schematic of cage/race contact angle.

Plot #4 - Cage Mass Center Acceleration

ORBITAL: Orbital angular acceleration of the cage mass center.

RADIAL: Radial acceleration of cage mass center.

AXIAL: Axial acceleration of cage mass center.

Plot #5 - Cage Mass Center Whirl Orbit

Generally the cage mass center whirl orbit is plotted in a plane normal to the bearing axis, which is the X-axis. Thus the Y component of mass center position is plotted as a function of the X component. Optionally, under program input control, any of the two components may be plotted against each other to obtain a whirl orbit in any plane.

Y-POSITION/CLEARANCE: Y component of the cage mass center position divided by the average cage/race guide land clearance.

Z-POSITION/CLEARANCE: Z component of the cage mass center position divided by the average cage/race guide land clearance.

Plot #6 - Cage Mass Center Position

ORBITAL: Angular position of cage mass center about the bearing axis.

RADIAL: Radial position of cage mass center

AXIAL: Axial position of cage mass center.

Plot #7 - Cage Angular Orientation

Angular orientation of the cage is defined by three angles, rotation about the principal polar axis of inertia (axis X) and orientation of this axis in the rolling element azimuth frame, defined by two angles, θ and ϕ, as follows:

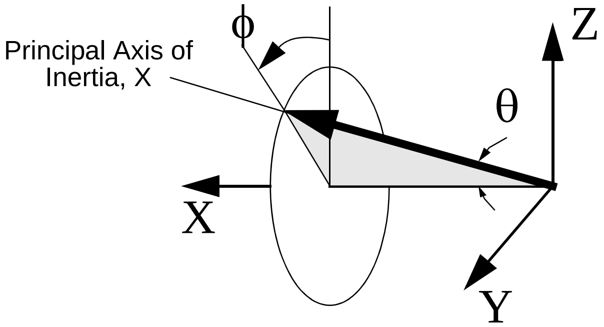


Figure 62. Cage orientation in the inertial coordinate frame.

THETA: Angle θ defining orientation of cage principal axis X.

PHI: Angle ϕ defining orientation of cage principal axis X.

ROTATION: Rotation of cage about the principal X-axis.

Plot #8 - Cage Angular Velocity

Angular velocity of the cage is defined by its magnitude and orientation of the angular velocity vector in the rolling element azimuth frame, defined by two angles, θ and ϕ, as follows:

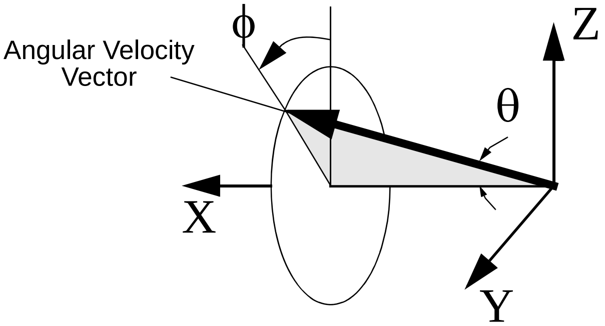


Figure 63. Cage angular velocity vector in the inertial coordinate frame.

OMEGA: Magnitude of Cage angular velocity vector.

THETA: Angle θ defining orientation of Cage angular velocity vector.

PHI: Angle ϕ defining orientation of Cage angular velocity vector.

Plot #9 to N-1 - Cage Pocket Interactions

Following the above 8 plots a number of plots are produced to display the cage pocket interactions. In each plot the results are plotted for a maximum of two guide surfaces in each pocket. Thus the number of pocket interaction plots depend on the number of cage pockets, or rolling elements, and the number of guide surfaces in each pocket.

NOR FORCE: Cage pocket normal contact force.

GEO INT: Geometric interaction in the cage pocket. Geometric interaction represents the clearance on contact deflection at the interacting cage and rolling element surfaces. A negative value of GEO INT represents contact while a positive value represents clearance.

CONTACT ANGLE: Angular position of cage-to-ball contact for ball bearings.

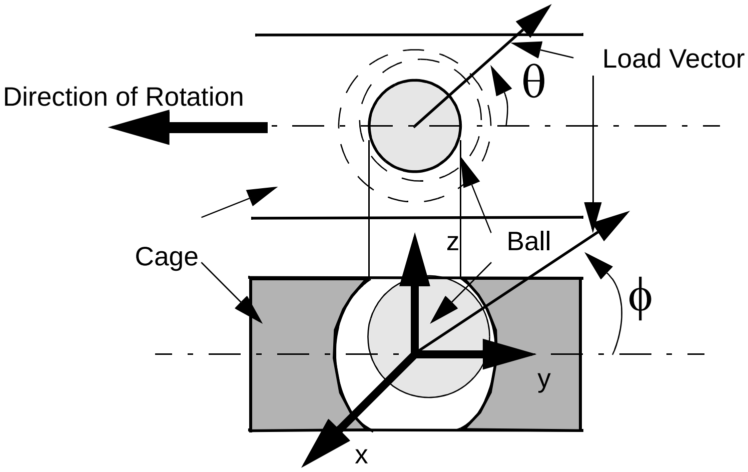


Figure 64. Ball/Cage contact angles for spherical pockets.

For ball bearings with spherical pocket there may be two components of contact angle, θ and ϕ, as defined above in figure 61. In the event of cylindrical pocket, the angle ϕ is zero and θ defines the contact position completely.

CONTACT POS: For roller bearings the guide surfaces are generally flat and the contact takes place normal to the guide surface. Thus, the contact angle is already defined from pocket geometry. In such cases the contact angle solutions are replaced of contact position values, which define the axial position of roller/cage contact along the roller axis.

4.2.3 Race Motion

Plot #1 - Race Mass Center Velocities

ORBITAL: Whirl or orbital angular velocity of race center about the bearing center.

RADIAL: Radial component of the race mass center velocity.

AXIAL: Axial component of the race mass center velocity.

Plot #2 - Applied Forces

The applied forces on the race are displayed in the base coordinate system as shown below in figure 62.

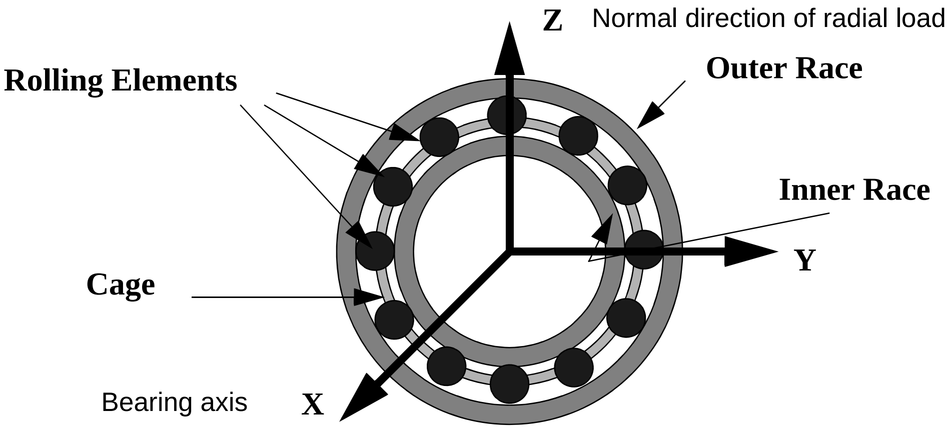


Figure 65. Base coordinate system.

X-COMP: X component of the applied force vector; X is the bearing axis.

Y-COMP: Y component of the applied force vector; Y is one of the transverse axes. Normally the bending moments are exerted about the Y axis when radial load is applied along the Z axis.

Z-COMP: Z component of the applied force vector; Z axis is normally along the radial load.

Plot #3 - Applied Moments

The applied moments on the race are displayed in the base bearing coordinate frame shown below:

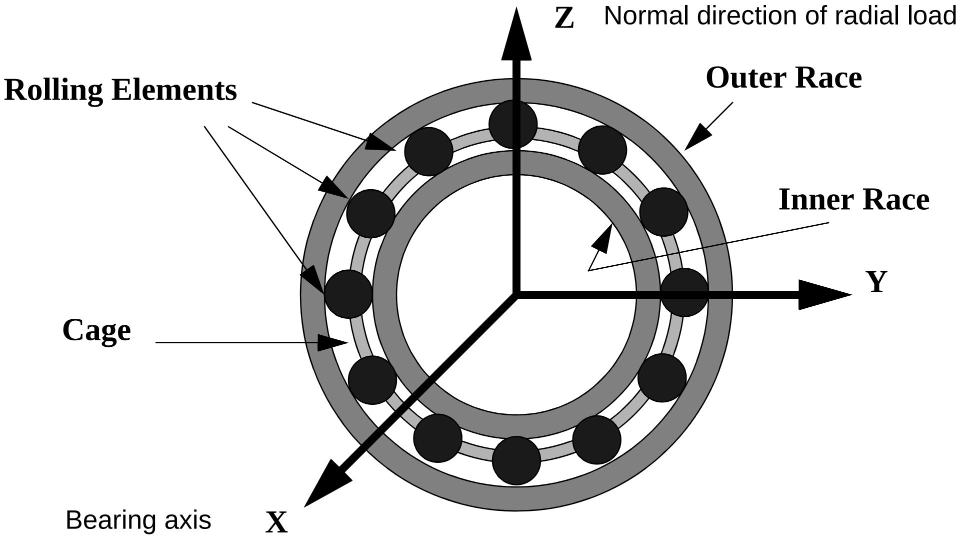


Figure 66. Base coordinate system.

X-COMP: X component of the applied moment vector; X is the bearing axis.

Y-COMP: Y component of the applied moment vector; Y is one of the transverse axes. Normally the bending moments are exerted about the Y axis when radial load is applied along the Z axis.

Z-COMP: Z component of the applied moment vector; Z axis is normally along the radial load.

Plot #4 - Race Mass Center Acceleration

ORBITAL: Orbital angular acceleration of the cage mass center.

RADIAL: Radial acceleration of cage mass center.

AXIAL: Axial acceleration of cage mass center.

Plot #5 - Race Mass Center Whirl Orbit

Similar to the cage the race mass center whirl orbit is generally plotted in a plane normal to the bearing axis, which is the X-axis. Thus the Y component of mass center position is plotted as a function of the X component. Optionally, under program input control, any of the two components may be plotted against each other to obtain a whirl orbit in any plane.

Y-POS: Y component of the race mass center position.

Z-POS: Z component of the race mass center position.

Plot #6 - Race Mass Center Position

X-POS: Axial position of race mass center.

Y-POS: Y position of race mass center

Z-POS: Z position of race mass center.

Plot #7 - Race Angular Orientation

Angular orientation of the race is defined by three angles, rotation about the principal polar axis of inertia (axis X) and orientation of this axis in the rolling element azimuth frame, defined by two angles, θ and ϕ, as follows:

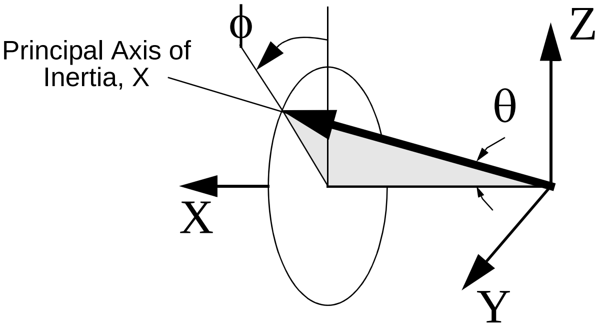


Figure 67. Race orientation in the inertial coordinate frame.

THETA: Angle θ defining orientation of race principal axis X.

PHI: Angle ϕ defining orientation of race principal axis X.

ROTATION: Rotation of race about the principal X-axis.

Plot #8 - Race Angular Velocity

Angular velocity of the race is defined by its magnitude and orientation of the angular velocity vector in the inertial frame, defined by two angles, θ and ϕ, as follows:

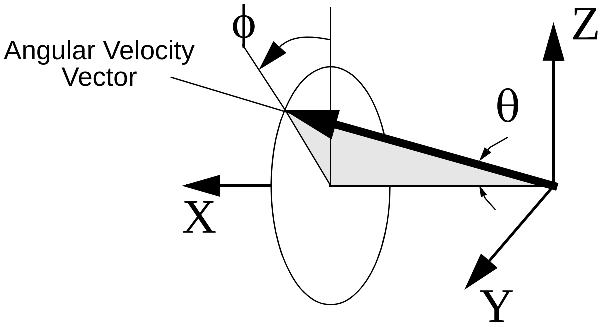


Figure 68. Race angular velocity vector in the inertial coordinate frame.

MAGNITUDE: Magnitude of race angular velocity vector.

THETA: Angle θ defining orientation of race angular velocity vector.

PHI: Angle ϕ defining orientation of race angular velocity vector.
plot is, therefore, only generated when the step size is constant. The plot displays relative amplitude as a function of frequency.

4.3 Graphics Animation Output

In addition to the plot output discussed above, ADORE, under user input control, may generate a data set which stores all key features of bearing element motion as a function of time. This data set may then be input to the optional graphics animation facility AGORE (Animated Graphics Of Rolling Elements) to display an animated view of bearing motion. Unlike the plot output, these animated displays permit the user to comprehend fairly sophisticated motion of bearing elements with very little or no imagination.

Typical overall bearing view is shown in figure 66, where all the ball, the cage and races are shown. In the central part of the diagram, the two blue coordinate frames correspond to the outer and inner races, which rotate with the race, while the green coordinate frame rotates with the cage. The rotating red arrow points to the location of race/cage contact. The dashed red circle seen just below the cage inner diameter corresponds to the inner race guide surface in this example. when the cage contact the race the resulting guide land force variations are displayed in the data area to the right of the graphic window. A time bar is seen in the lower part of the display. As the bearing rotates this bar fills indicating the extent of simulation completed. Anytime the balls make contact in the cage pockets, a red asterisk appears in the pocket, as seen in pocket numbers 1 and 18 in figure 66.

The animated display can be controlled by the option button displayed to the right of the graphic area, while the various views are controlled by the menu options as discussed earlier in Section 1 of this manual.

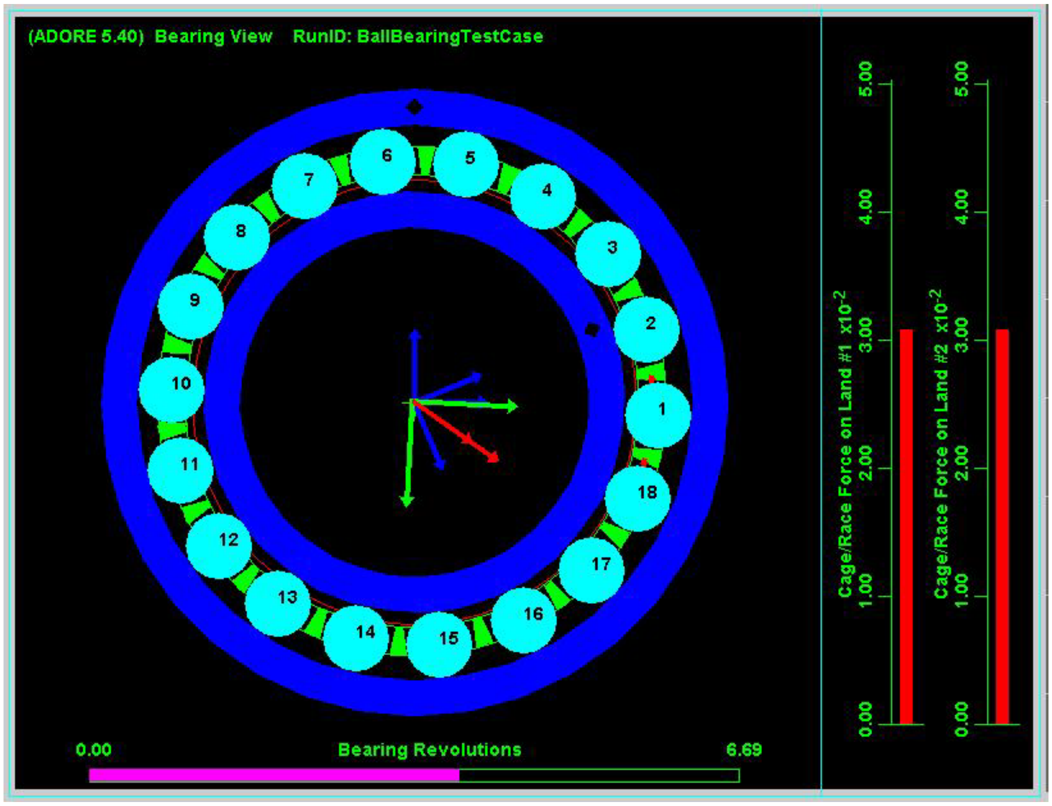

(ADORE 5.40) Bearing View RunID: BallBearingTestCase
6
5
7
4
8
3
9
2
10
1
11
18
12
17
13
16
14
15
0.00
Bearing Revolutions
6.69
Cage/Race Force on Land #1 x10-2
Cage/Race Force on Land #2 x10-2
0.00
1.00
2.00
3.00
4.00
5.00

Figure 69. Typical bearing view as provided by the animation facility, AGORE.

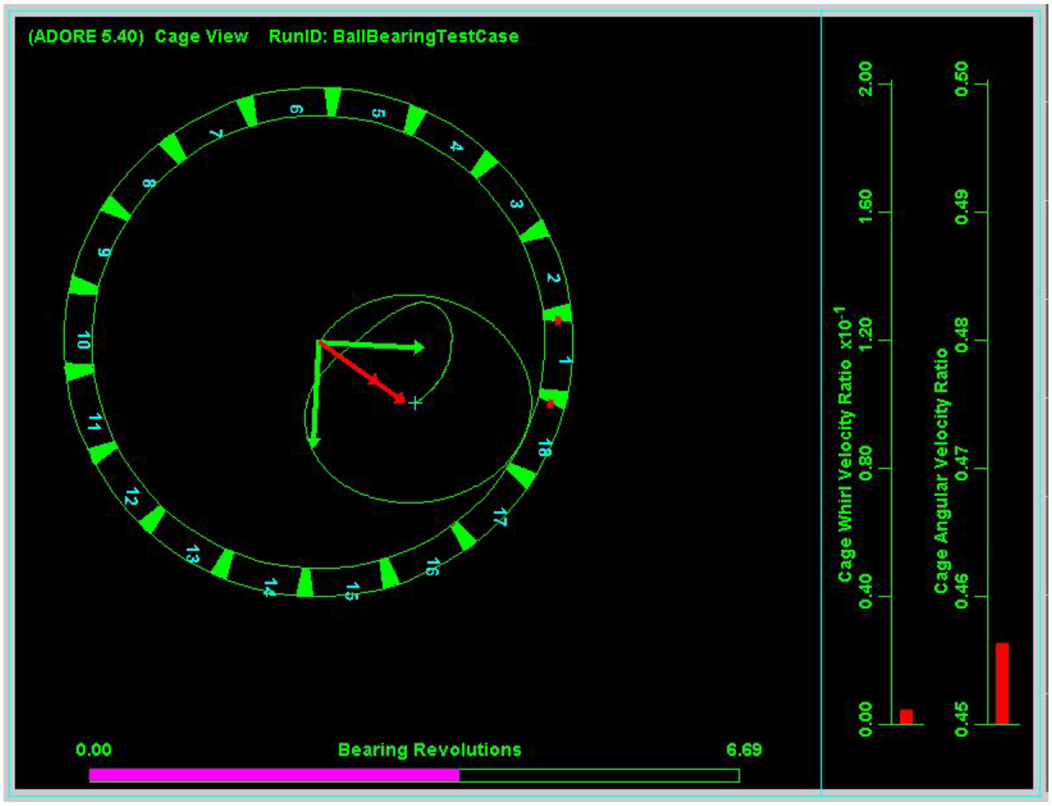

(ADORE 5.40) Cage View RunID: BallBearingTestCase
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
0.00
Bearing Revolutions
6.69
Cage Whirl Velocity Ratio x10^{-1}
0.00
0.40
0.80
1.20
1.60
2.00
Cage Angular Velocity Ratio
0.45
0.46
0.47
0.48
0.49
0.50

Figure 70. Typical cage view as provided by the animation facility, AGORE.

By selecting the cage view from the view menu, the cage motion is displayed in a two dimensional plane as shown above in figure 67. Again the pockets in which the rolling elements are contacting are highlighted with a red asterisk (pocket numbers 18 and 1 in figure 67). In the central part of the display cage whirl orbit is plotted at an enlarged scale, as the cage mass center whirl around the bearing center. The red arrow, again points to the direction of cage/race contact. Since the green coordinate shown in the central part of the display is fixed in the cage, orientation of the red arrow relative to the green coordinate frame has substantial practical significance. For a well behaved cage/race contact, the red arrow should be constantly moving relative to the green coordinate frame indicating the cage/race contact is uniformly distributed around the cage surface. Fixed orientation of this red arrow relative to green coordinate frame will imply that a fixed point on the cage is interacting with the race, indicating a potential wear of the cage surface.

The data area to the right of the display plots the cage whirl and angular velocity variations, while the time bar in the bottom shows the extent of simulation.

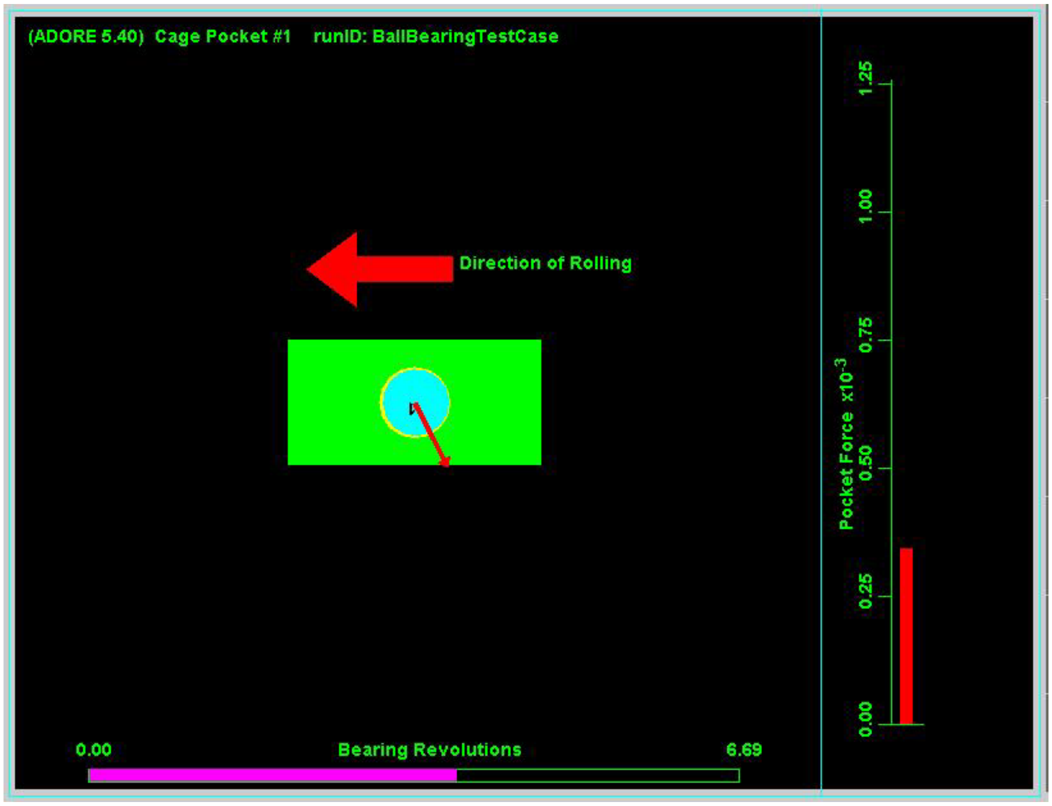
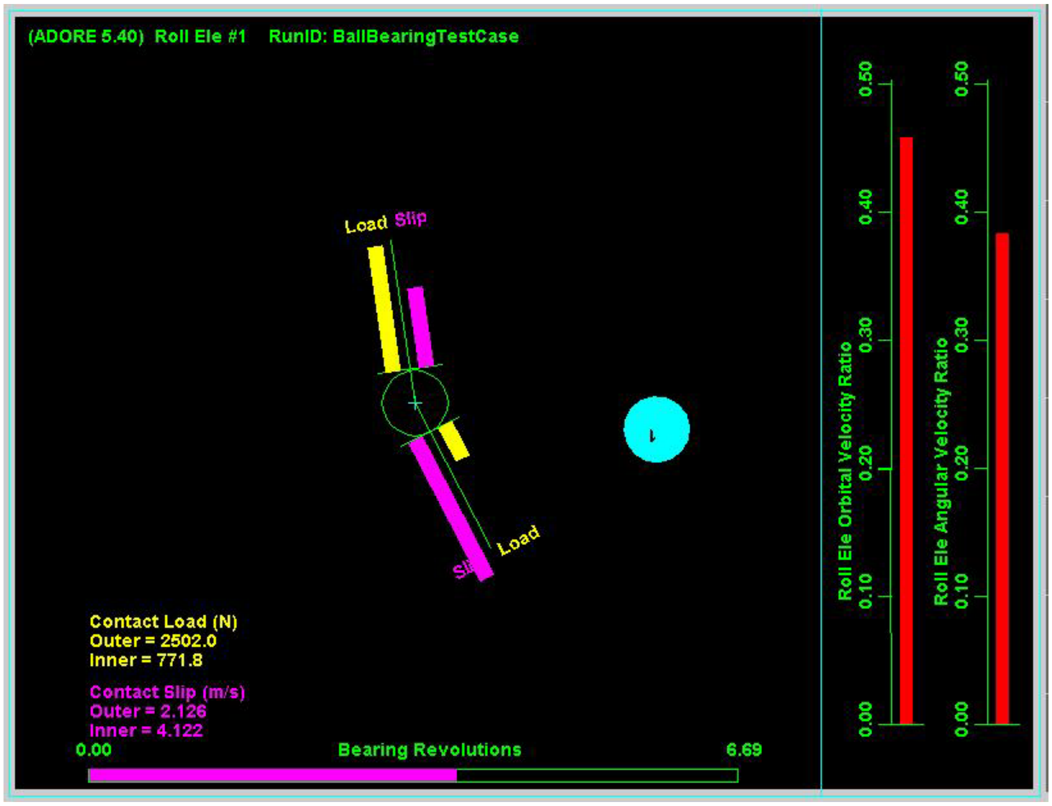


Figure 71. Typical cage pocket view as provided by AGORE.

Typical cage pocket interaction is shown above in figure 68. Now the cage pocket is stationary while the rolling element moves in the pocket. The direction of cage rotation is shown above by the thick red arrow. The thin red arrow at the center of the rolling element indicated the direction of cage pocket contact, while the contact force is displayed the right in the data area. By using the frame advance buttons the rolling element to cage collisions can be interactively tracked.



(ADORE 5.40) Roll Ele #1 RunID: BallBearingTestCase

Contact Load (N)
Outer = 2502.0
Inner = 771.8

Contact Slip (m/s)
Outer = 2.126
Inner = 4.122

Figure 72. Typical rolling element view as provided by AGORE.

Figure 69 above shows the typical rolling element motion view. As the rolling element moves around the bearing, the contact loads and maximum slip in the contact are displayed in this view in an animated fashion. The data area contains the rolling element orbital and angular velocities, which are plotted as ratios to the shaft angular velocity. A large variation in these ratios will represent bearing skid.

The above example represents a ball bearing example. Similar animations may be obtained with a roller bearing, where the roller/flange interaction is also included. In addition, the race motion may also be seen. These view may be useful when the race is subjected to motion due to rotating load, external vibrations, or other more complicated conditions.

5. DATA MANAGEMENT IN ADORE

Since ADORE provides a time-transient analysis, the output for a typical run containing several thousands of time steps may be prohibitively large. Also, after making a run for a definite number of time steps and after reviewing the results, it may be found that the performance simulations must be further advanced in time in order to achieve reasonable steady-state solutions. In order to efficiently fulfill such a need it is necessary to restart the integration from the time at which the previous run was terminated. Also, for easy interpretation of the results it may be essential to plot the entire output, generated during all the runs, on the same graph. It is for these reasons that some type of data management is necessary.

ADORE employs several sequential data files which are opened during execution. The list of default file names and fortran unit codes used are documented in the following table.

Table 2: ADORE Data Sets

File Name	FORTRAN Device Code	Device Code Variable	File Contents
DATA.txt	2	input	ADORE input data
PRINT.txt	3	output	Print output
MASTER	7	master	Master data file which stores all program inputs
FINAL	8	final	Final solution vector
SOL1	11	pfile(1)	Plot solutions for selected bearing element #1
SOL2	12	pfile(2)	Plot solutions for selected bearing element #2
SOL3	13	pfile(3)	Plot solutions for selected bearing element #3
SOL4	14	pfile(4)	Plot solutions for selected bearing element #4
SOL5	15	pfile(5)	Plot solutions for selected bearing element #5
SOL6	16	pfile(6)	Plot solutions for selected bearing element #6
SOL7	17	pfile(7)	Powerloss and life data
SOL8	18	pfile(8)	Graphic animation data
SOL9	19	pfile(9)	User selected data

All devices are defined in program module “Devices”. If on a given computer system, any of the above device codes are used for other system data sets, then the above defaults must be appropriately changes. The default file names may be changed to any user defined names by exercising the designated program option: kFnOpt = 1 on input data Record 1 and then defining the file names on Record 2.3. Typical examples of the various data sets are included in program media under subdirectory Disk1 (see Media Contents in Chapter 2 of this manual).

A detailed description of each of the data sets, including the pertinent data, is presented below.

5.1 File DATA.txt

This is the user supplied input file which contains all the input data required to execute a run. This file may be prepared in accordance to ADORE input instructions described in section 4 of this manual. Either any text editor or the ADORE input facility, AdrInput, may also be used to prepare this file. See examples in Appendix B for typical listings of this file.

5.2 File PRINT.txt

All the print output goes to this file. At the end of the run the file may either be printed or viewed with any text editor. Typical output is contained in the program media under subdirectory Disk1 (see Media contents in Chapter 2 of this manual).

5.3 File MASTER

This file contains all the bearing data and certain solutions at the final time step, which are required for restarting the simulation in a subsequent run. The file is created during the first run and it is updated at each subsequent run. Note that in the event of an abnormal termination, this file may not be updated properly. It is therefore necessary, that the files created by a preceding run be safely kept until the following run is successfully completed and the properly updated files become available.

5.4 File FINAL

The file FINAL contains the last solution vector. The data in this file is written at the end of each run. The purpose of this file is to provide the initial conditions under the following circumstances:

1. After making an initial run if the solutions demonstrate that the simulations have to be continued over more time in order to ascertain steady-state, then the final solution vector written in file FINAL by the prior run may be used as initial condition for a subsequent continuation run. This is accomplished by setting kIcOpt = 1 on Record 1 of ADORE input. In such an instance no input data after Record 2 is required, since it is read from the file MASTER created by the initial run.

2. In the case of simulating the bearing performance over very large time domain, the transient solutions may be produced over several thousand time steps; and, even after a reasonable control of the amount of data, the data files may become very large and they may exceed the permissible mass storage limits on the available computer system. Under such a condition, it may be necessary to divide the simulation into several “batches”, where each batch is independent of the other. The file FINAL created by the last run of batch #1, may be used to specify the initial conditions for the first run of batch #2, and thus, the continuity between the two batches is maintained. This is accomplished by executing ADORE with the arbitrary initial condition option kIcOpt = -1 on ADORE input record 1. Note that under such a mode of operation it may not be possible to plot the data of the two or batches together.

3. After a steady-state solution has been obtained for a certain bearing application, it may often be desired to investigate the influence of a small perturbation in one of the bearing design or operating parameters. This is easily accomplished by using the file FINAL, which may contain the steady-state solution, to prescribe the initial conditions for the perturbed condition. This is also done with kIcOpt = -1 on Record 1 of ADORE input.

5.5 Files SOL1 to SOL6

These files contain the plot data for a maximum of six bearing elements for which the plot output may be generated. Again, the files are created during the first run and updated during subsequent runs. In the event the plot data is monitored for less than the maximum permissible number of bearing elements, some of these files may remain unused. ADORE assigns the files to the required bearing elements starting with SOL1.

All of these files are ASCII formatted text files. The first two columns are always blank followed by a maximum of 130 columns of text. The files may, therefore, be printed on any 132 columns printer. There are two types information, which is recorded in these files:

5.5.1 Header Information

The first line contains the program version and the bearing specification code supplied by the user on input record 3.1, in format (2x,a12,5x,a36).

On the second line a title for the specific bearing element is included in a character string. Depending on the bearing element, the length of this string may vary. However, the string is terminated by the “$” character.

The third line contains a number of integer variables in format (2x,20i6). A description of these variables is as follows:

Variable # Description

1 Number of data values in the solution record, discussed later in this section.

2 Number of rolling elements in the bearing.

3 Number of rolling elements contained in a cage segment, when the cage is segmented. For a one piece cage this variable is equal to the number of rolling elements.

4 Number of cage segments in the bearing.

5 Index of the bearing element, as defined in input data record 3.4, associated with the data file.

6 Flange indicator flag for the outer race. When the race flanges exist on the outer race (either kFlngInd11 or kFlngInd21 on Record 3.2 is nonzero) this flag has a value of 1 otherwise it is set to zero.

7 Flange indicator flag for the inner race. When the race flanges exist on the inner race (either kFlngInd12 or kFlngInd22 on Record 3.2 is nonzero) this flag has a value of 1 otherwise it is set to zero.

8-17 A vector of length 10, containing the length of character strings in each component of the units vector described later.

18 Bearing type, as defined in input data record 3.2.

19 Cage pocket code, as defined in input data record 7.0.

20 Number of active surfaces in the cage pocket. This depends on the pocket shape. For example, for a rectangular pocket in a roller bearing, there are two active surface, while for a ball bearing with cylindrical pocket, there is one continuous surface.

The fourth, and last line in the file header contains the units vector, which is a character string array of length 10, in format (2x,10(A10,2x)). The components of this array contain the various units used in the plots. The number of characters in each unit components in contained in variables 8-17, as discussed above. The last component is blank, and this is used in place of units when the variable plotted is dimensionless.

5.5.2 Solution Records

After the above header information, the solution records are stored in the files at each selected time step (see description of input variable kPltFreq on Record 1). The first line in the solution record contains five variables, one integer and four floating point numbers, in format (2x,i16,1p,6e16.7). The variables are:

Variable # Description

1 Time step number.

2 Bearing rotation in revolutions.

3 Angular position of rolling element, if the file belongs to a rolling element, in revolutions.

4 The last step size in real time (seconds).

5 Current value of real time (seconds).

Subsequent lines in the solution record contain the different variables plotted in a given data set. Most variables have appropriate dimensions. The units conform to the unit system description in the program input section. The units used for various output variables in the available SI or English system of units are defined as follows:

Length:	Meter (m) or inch (in).
Force:	Newton (N) or pound force (lbf).
Time	Second (s)
Pressure:	Pascal (Pa) or pound per square inch (lbf/in^2).
Temperature:	Degrees Kelvin (K) or degrees Rankine (R)
Velocity:	Meter per second (m/s) or inch per second (in/s).
Acceleration:	Meter per second square (m/s^2) or inch square per second (in/s^2).

Angular Position:	Degrees (deg).
Angular velocity:	Revolutions per minute (rpm)
Angular Acceleration:	Revolutions per minute per second (rpm/s).
Wear Rate:	Cubic meter per second (m^3/s) or cubic inch per second (in^3/s).
Heat Generation:	Watts (w) or inch-pound per second (in.lbf/s).

The contents of the record depend on the specific bearing element assigned to the data file. There are, of course, three types of bearing elements, rolling element (ball or roller), cage, and the race. For each of these elements, the variables in the solution record are discussed below.

Solution Record for Rolling Element

The number of components in the solution vector are different for ball and roller elements. The actual number of components is recorded in variable #1 on third line of the header information discussed above. The variables in the rolling element solution file are list below sequentially:

1 Orbital acceleration of the rolling element (rpm/s).

2 Radial acceleration of the rolling element (m/s^2 or in/s^2).

3 Axial acceleration of the rolling element (m/s^2 or in/s^2).

4 Mass center orbital angular velocity of the rolling element (rpm).

5 Radial velocity of rolling element mass center (m/s or in/s).

6 Axial velocity of rolling element mass center (m/s or in/s).

7 Orbital position of the rolling element (deg).

8 Radial position of the rolling element (m or in).

9 Axial position of the rolling element (m or in).

10-11 Angular orientation, the angles (deg) theta (θ) and phi (ϕ), of the rolling element, defined as follows:

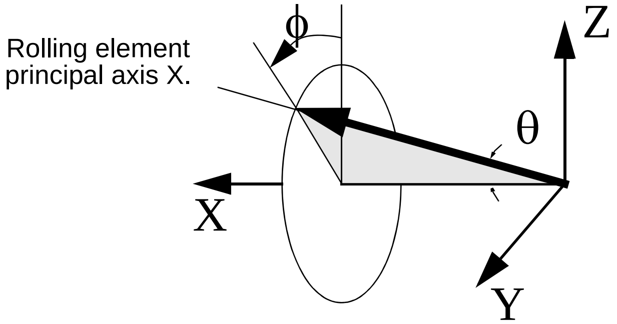


Figure 73. Rolling element orientation in the azimuth frame.

12 Total rotation of the rolling element (deg).

13 Magnitude of the rolling element angular velocity vector (rpm).

14-15 Orientation of the angular velocity vector, the angles (deg) theta (θ) and phi (ϕ), which are defined similar to the angles shown above for rolling element orientation.

16-17 Contact loads (N or lbf) at the outer and inner races.

18-19 Contact angles (deg) at the outer and inner races.

20-21 Spin-to-Roll ratios at the outer and inner race contacts.

22-23 Maximum slip velocity (m/s or in/s) in the outer and inner race contacts.

24-25 Heat generation (W or in.lbf/s) in the outer and inner race contacts.

26-27 Lubricant film thickness (m or in) at the outer and inner race contacts.

28-29 Roller guide flange forces (N or lbf) on the two possible outer race guide flanges (roller bearings only).

30-31 Roller guide flange geometric interaction (m or in) at the two possible guide flanges on the outer race (roller bearings only).

32-33 Heat generation (W or in.lbf/s) at the two outer race flange contacts (roller bearings only).

34-35 Roller guide flange forces (N or lbf) on the two possible inner race guide flanges (roller bearings only).

36-37 Roller guide flange geometric interaction (m or in) at the two possible guide flanges on the inner race (roller bearings only).

38-39 Heat generation (W or in.lbf/s) at the two inner race flange contacts (roller bearings only).

Solution Record for the Cage or Cage Segment

The actual number of elements in the cage or cage segment solution vector depends on the number of pockets in the cage segment, and the number of active surfaces in each pocket. The total number of applicable variables are again recorded in variable #1 on third line of the header information. The variable sequence in the cage motion solution file is as follows:

1 Cage mass center whirl velocity ratio (whirl angular velocity/race angular velocity).

2 Radial velocity of cage mass center (m/s or in/s).

3 Axial velocity of cage mass center (m/s or in/s).

4 Cage/Race force (N or lbf) at guide land #1.

5 Geometric interaction (m or in) at the cage/race guide land #1.

6 Contact angle (deg) at the cage/race guide land #1.

7 Cage/Race force (N or lbf) at guide land #2.

8 Geometric interaction (m or in) at the cage/race guide land #2.

9 Contact angle (deg) at the cage/race guide land #2.

10 Orbital angular acceleration (rpm/s) of cage mass center.

11 Radial acceleration (m/s^2 or in/s^2) of cage mass center.

12 Axial acceleration (m/s^2 or in/s^2) of cage mass center.

13-15 Cartesian (X,Y,Z) components of cage mass center position divided by the average guide land clearance. If cage/race guidance is present only at one land, then the average clearance is equal to the clearance at this land.

16 Orbital position (deg) of the cage mass center.

17-18 Radial and axial position (m or in) of cage mass center.

19-20 Angular orientation of the cage, the angles, the angles (deg) theta (θ) and phi (φ), of the rolling element, defined as follows:

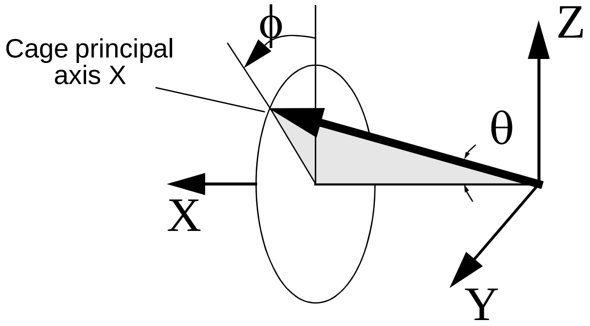


Figure 74. Cage orientation in inertial frame.

21 Total rotation (deg) of the cage.

22 Angular velocity ratio (angular velocity/shaft velocity) of the cage.

23-24 Orientation of the cage angular velocity vector, the angles (deg) theta (θ) and phi (ϕ), which are defined similar to the angles shown above for cage angular orientation.

Following the above basic solution vector, the solutions in each cage pocket are recorded for each guide surface. In general there are four solutions for each pocket guide surface, pocket force (N or lbf), geometric interaction (m or in) and two components of contact angle (deg) or contact position (m or in).

For ball bearings with spherical pockets the two components of contact angels, θ and ϕ, are shown in the figure 72 below. For cylindrical pockets the angle ϕ will be zero, while for conical pockets it is defined by the cone angle. For rectangular or square pockets ϕ will once again be zero, and θ will define the orientation of pocket guide surface relative to the pocket center. The number of guide surfaces for ball bearings is essentially one for most pockets except for square or rectangular pockets where it is 4. Thus is general there are four solution values for each guide surface in each pocket.

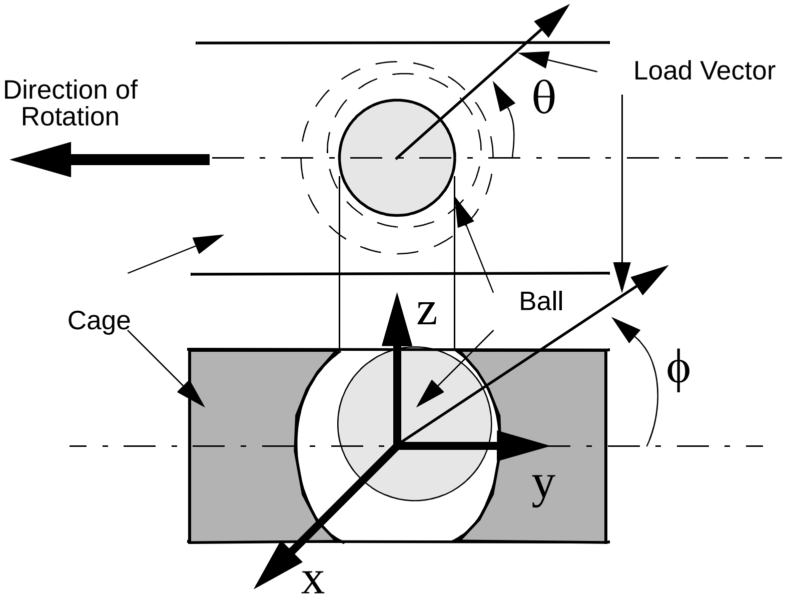


Figure 75. Ball/Cage contact angles for spherical pockets.

With the pocket denoted as i (i=1,n, n being the number of pockets), and guide surface denoted as j (j=1,m, where m=1 for all pockets, except square and rectangular, in which case m=4), the cage pocket solutions for ball bearings are documented as follows:

Variable #	Description
24+(i-1)*4m+(j-1)*4+1	Contact force (N or lbf) in pocket i on guide surface j.
24+(i-1)*4m+(j-1)*4+2	Geometric interaction (m or in) in pocket i on guide surface j.
24+(i-1)*4m+(j-1)*4+3	Contact angle θ (deg) in pocket i on guide surface j.
24+(i-1)*4m+(j-1)*4+4	Contact angle ϕ (deg) in pocket i on guide surface j.

For roller bearings there are always multiple guide surfaces and the contact angle, θ, as defined above for ball bearings, will either be zero or 180^o, respectively for the guide surfaces which drive or get driven by the rolling elements. Since the surfaces are flat the contact angle, ϕ, is always zero. Except for roller bearings with cylindrical pockets where q will define the angular position of roller/cage contact, similar to ball bearings with spherical pockets. Thus for each guide surface there are three solutions recorded for roller bearings. For cylindrical pockets these solutions are contact force, geometric interaction and contact angle ϕ. For all other pocket shapes the contact angle solution is replaced by axial position of contact on the guide surface. Thus for roller bearings, once again with the pocket denoted as i (i=1,n, n being the number of pockets), and guide surface denoted as j (j=1,m, where m=2 for most pockets except for customized pockets), the cage pocket solutions are documented as follows:

Variable #	Description
24+(i-1)*4m+(j-1)*4+1	Contact force (N or lbf) in pocket i on guide surface j.

24+(i-1)*4m+(j-1)*4+2 Geometric interaction (m or in) in pocket i on guide surface j.

24+(i-1)*4m+(j-1)*4+3 Contact angle ϕ (deg) in pocket i on guide surface j for roller bearings with cylindrical pockets and axial position of contact (m or in) or all other pockets.

Solution Record for the Races

The solution vector of the races is quite similar to the basic record of the cage. There are a total of 24 variables in the solution record.

Variable # Description

1 Race mass center whirl velocity (rpm).

2 Radial velocity of race mass center (m/s or in/s).

3 Axial velocity of race mass center (m/s or in/s).

4-6 Applied forces (N or lbf) in the X,Y,Z directions on the outer and inner races.

7-9 Applied moments (N.m or lbf.in) in the X,Y,Z directions on the outer and inner races.

10 Orbital angular acceleration (rpm/s) of race mass center.

11-12 Radial and axial acceleration (m/s^2 or in/s^2) of race mass center.

13-15 Cartesian (X,Y,Z) components of race mass center position (m or in).

16 Orbital position (deg) of the race mass center.

17-18 Radial and axial position (m or in) of race mass center.

19-20 Angular orientation of the race, the angles, the angles (deg) theta (θ) and phi (φ), for race orientation, defined as follows:

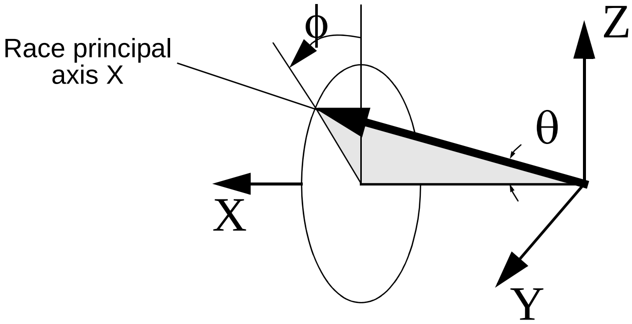


Figure 76. Race angular orientation in inertial frame.

21 Total rotation (deg) of the race.

22 Angular velocity (rpm) of the race.

23-24 Orientation of the race angular velocity vector, the angles (deg) theta (θ) and phi (ϕ), which are defined similar to the angles shown above for race angular orientation.

5.6 File SOL7

This file contains data for the power dissipation and life plots. This file is always active, it is again created during the first run and updated in subsequent runs.

Similar to the SOL1 to SOL6 files this files contains a header and a solution record.

5.6.1 Header Information

Format of the header information contained in the first four lines of the data file is identical to that discussed above for files SOL1 to SOL6.

The first line contains the program version and the bearing specification code supplied by the user on input record 3.1, in format (2x,a12,5x,a36).

On the second line a plot title “Power Dissipation and Life$” is included. Note that the character string is terminated with “$”.

The third line contains a number of integer variables in format (2x,20i6). A description of these variables is as follows:

Variable # Description

1 Number of data values in the solution record, discussed later in this section.

2 Number of rolling elements in the bearing.

3 Number of rolling elements contained in a cage segment, when the cage is segmented. For a one piece cage this variable is equal to the number of rolling elements.

4 Number of cage segments in the bearing.

5 Index of the bearing element, as defined in input data record 3.4, associated with the data file.

6 Flange indicator flag for the outer race. When the race flanges exist on the outer race (either kFlngInd11 or kFlngInd21 on Record 3.2 is nonzero) this flag has a value of 1 otherwise it is set to zero.

7 Flange indicator flag for the inner race. When the race flanges exist on the inner race (either kFlngInd12 or kFlngInd22 on Record 3.2 is nonzero) this flag has a value of 1 otherwise it is set to zero.

8-17 A vector of length 10, containing the length of character strings in each component of the units vector described later.

18 Bearing type, as defined in input data record 3.2.

19 Cage pocket code, as defined in input data record 7.0.

20 Number of active surfaces in the cage pocket. This depends on the pocket shape. For example, for a rectangular pocket in a roller bearing, there are two active surface, while for a ball bearing with cylindrical pocket, there is one continuous surface.

The fourth, and last line in the file header contains the units vector, which is a character string array of length 10, in format (2x,10(a10,2x)). The components of this array contain the various units used in the plots. The number of characters in each unit components in contained in variables 8-17, as discussed above. The last component is blank, and this is used in place of units when the variable plotted is dimensionless.

5.6.2 Solution Record

The first line in the solution record is identical to that in other plot files. The solution records are stored in the files at each selected time step (see description of input variable kPltFreq on Record 1). The first line in the solution record contains five variables, one integer and four floating point numbers, in format (2x,i16,1p,6e16.7). The variables are:

Variable # Description

1 Time step number.

2 Bearing rotation in revolutions.

3 Angular position of rolling element, if the file belongs to a rolling element, in revolutions.

4 The last step size in real time (seconds).

5 Current value of real time (seconds).

Subsequent lines in the solution record contain the various solutions at the selected time step. Most quantities are dimensional and the units conform to the unit system description in the program input section. The solution variables are:

Variable # Description

1 Total power dissipation (W or lbf.in/s) in the bearing.

2 Fraction of total power consumed in churning and drag.

3 Fatigue life (Hours).

4-5 Applied moment (N.m or lbf.in) about the X-axis on the outer and inner races.

6-7 Applied moment (N.m or lbf.in) about the Y-axis on the outer and inner races.

8-9 Applied moment (N.m or lbf.in) about the Z-axis on the outer and inner races.

10 Time averaged wear rate (m^3/s or in^3/s) for rolling element #1.

11-12 Time averaged wear rate (m^3/s or in^3/s) for the outer and inner races.

13	Time averaged wear rate (m^3/s or in^3/s) for the cage.
14	Rolling element bulk temperature (K or R)
15-16	Bulk temperature of the outer and inner races (K or R).
17	Cage bulk temperature (K or R).

5.7 File SOL8

Similar to SOL7, this file is also created at the first run and updated in subsequent continuation runs. The file is only active when the graphics animation option, kAGraf on ADORE input Record 1 is nonzero.The data contained here is used by the graphics animation code, which displays an animated pictorial view of the bearing, based on the dynamic solutions generated by ADORE. Again, the file has two parts, the header and solution record.

5.7.1 Header Information

In addition to the information contained in the other plot files the header in this file also contains some geometrical information.

The first line contains the program version and the bearing specification code supplied by the user on input Record 3.1, in format (2x,a10,5x,a36).

The second line is similar to third line in the other plot files. There are a number of integer variables in format (2x,40i3).

Variable #	Description
1	Bearing type, as defined in input data record 3.2.
2	Number of rolling elements in the bearing.
3-12	A vector of length 10, containing the length of character strings in each component of the units vector, as in the other plot files.
13	Number of geometrical variables included in the header after the units strings. The actual value is 12.
14	Number of variables in the solution vector. This depends on number of rolling elements in the bearing. The actual value is 6*(n+3)+5*n+4, where n is the number of rolling elements.

The third line in the file header contains the units vector, which is a character string array of length 10, in format (2x,10(a10,2x)). The components of this array contain the various units used in the plots. The number of characters in each unit components in contained in variables 3-12, as discussed above. The last component is blank, and this is used in place of units when the variable plotted is dimensionless.

Following the above three lines, the header also includes 12 geometrical variables in format (2x,13e10.3). Since the all graphics are processed to some scale, and all geometrical variables have a length scale, all the quantities are in dimensionless form.

Variable #	Description
1-2	Cage outer and inner radii (m or in).

3-4	Cage outer and inner radial clearances (m or in).
5-6	Cage pocket clearances I and II (m or in) as defined in input data record 7.3.
7	Rolling element radius (m or in).
8	Pitch diameter of the bearing (m or in).
9-10	Outer race outer and inner radii (m or in).
11-12	Inner race outer and inner radii (m or in).

5.7.2 Solution Record

The first line in the solution record contains three variables in format (2x,i16,6e16.7). The variables are:

Variable #	Description
1	Time step number.
2	Bearing rotation in revolutions.
3	Current value of real time (seconds).

Subsequent lines in the solution record, which is composed of 11 variables for each rolling element, 10 variables for the cage and 6 variables for each of the races. For a bearing with n rolling elements, first n sets of 11 variables each are assembled for the rolling elements, then the 10 variables for the cage are added, and finally the two sets of 6 variables each are added for the two races. The data is written in format (2x,13e10.3). Notation for the units are identical to that used earlier for other plots files. In addition, a notation B for rolling elements, C for cage and R for race is used in the following description of the different variables:

Variable	Description
1-3	Rolling element mass center coordinates [axial (m or in), radial (m or in) and orbital (rad)].
4-6	The transformation angles (rad) which define the angular orientation of the rolling element.
7-9	Position vector (m or in) which locates rolling element center relative to the cage pocket center.
10	Cage pocket force on the rolling element (N or lbf).
11	Cage pocket contact angle (rad).
1-3	Cartesian (X,Y,Z) coordinates (m or in) of cage mass center.
4-6	Three transformation angles (rad) which define angular orientation of the cage.
7-8	Cage/Race force (N or lbf) and contact angles (rad) for guide land #1.

9-10 Cage/Race force (N or lbf) and contact angles (rad) for guide land #2.

1-3 Cartesian (X,Y,Z) coordinates (m or in) of race mass center.

4-6 Three transformation angles (RAD) which define angular orientation of the race.

5.8 File SOL9

This file is for user output. Using the optional subroutine Adrx9, any of the solutions of interest may be output to this file at given time step. The data may then be used as input into other modeling software or post processing procedures, such as plotting.

6. USER PROGRAMMABLE FUNCTIONS AND SUBROUTINES

In addition to the flexibility in the input data, several user-programmable subroutine in the ADRXn module allow a number of special effects to be very easily programmed. Access to data internal to ADORE is provided by attaching appropriate data module to user codes. Complete documentation of each variable in the various data modules is included in the source listing. Considerable care must be exercised while using the data modules to avoid any unintentional change of the values set for any of the variables.

In addition to optional programming the user also has access to certain parameters which are used to set up ADORE. For example, by default the maximum number of rolling elements is set to 40. In the event the bearing to be modeled has more than this maximum limit of rolling elements then this parameter can be increased. Likewise if done of the user applications will require this maximum number of rolling elements, then the limit can be reduced to save memory and possibly speed up the computation. The module “Parameters” contains such parameters. The source listing provides complete documentation of each parameters and the values set are clearly shown.

As the user makes changes to the ADORE source code and/or adds code to the user programmable subroutines it is often desirable to track the modified version for documentation purposes. To facilitate this ADORE version 5.60 introduces a new variable to define “user version”. This variable is simply a string of characters which is appends the main ADORE version included in all print and plot output and ADORE data sets. The character string variable “jver” included in data module “Constants” is used to set the user version. After making any changes to ADORE source code and/or attaching any user subroutines it is recommended that the user set an appropriate character string in this data module to track the modified version of the code.

The objective behind user access to source codes and permitting user programming is to permit customization of the model to meet the user needs as closely as possible. ADORE is structured and modularized in such a way that simple programming in the user programmable functions and subroutines will permit modeling of most sophisticated applications.

The purpose and the programming instructions for each of these routines are documented in the source listings. A brief overview of the scope of each subroutine is presented below.

6.1 Subroutine ADRX0

This subroutine just provides the user with an interface to access a materials property data base. For given bearing element the materials properties may be extracted from the data base and passed on the relevant subroutines in ADORE.

6.2 Subroutine ADRX1

Any time variations in the applied loads and race speeds can be programmed in this subroutine to any degree of complexity. Often experimental data available from laboratory tests of the system can be incorporated to obtain bearing performance simulations under actual laboratory conditions. Under default conditions this subroutine is basically empty as seen in the source listing presented below:

```
      subroutine Adrx1
!
!-----------------------------------------------------------------------
```

```
!
!    Adrx1   optional functions
!    -----   ------------------
!
!  this subroutine permits modeling of the following optional
!  functons in adore --
!
!  1.  applied loads and moments.
!  2.  race accelerations.
!  3.  additional external loads and moments on rolling elements.
!  4.  additional external loads and moments on cage segments.
!  5.  appropriate functions for moving coordinate frames.
!  6.  temperature of various elements.
!  7.  gravity vector.
!  8.  arbitrary suppression of degrees-of-freedom and symetry.
!
!
!  general instructions
!  --------------------
!
!  1. the subroutine operates in four modes, as defined by the value of
!     an incoming flag icm(1) --
!
!     mode 1  icm(1) = -1 the very first call for reading any required
!                         input data and for setting tolerance values
!                         and certain program options. any write
!                         statments in this mode produce print output
!                         under the heading -input from user programmable
!                         subroutines- immediately following the listing
!                         of main data records.
!
!     mode 2  icm(1) = 0  second call to print any data under the initial
!                         output section, -output from user programmable
!                         subroutines-, following the main input data
!                         documentation.
!
!     mode 3  icm(1) = 1  main computing mode. the code for this mode
!                         is executed in the inner most loop of the
!                         of the program. the code should, therefore, be
!                         free of any input/output statements. also,
!                         complexity of this code shall directly affect
!                         the overall computing effort.
!
!     mode 4  icm(1) = 2  certain output can be printed in this mode,
!                         following the nominal adore print output
!                         produced at selected time step. the code under
!                         this mode is executed only when the print output
!                         is required, as defined by options on adore
!                         input record 1.
!
!     mode 5  icm(1) = 3  certain output can be stored in data files
!                         created by the user at first call to this
!                         subroutine, called with icm(1)=-1.
!
!  2. the output flag jcm(1) must be set to >=1 at the first call,
!     initiated with icm(1) = -1. the subroutine is never called after
!     the very first call unless this flag is set to >=1. significance
!     of jcm(1) is as follows:
!     1 = compute model only, no output is controlled
!     2 = computations with output control
!
!  3. all computations are perfomed in dimensionless form. the
!     force, length and time scale values, as available in the
!     subroutine, must be used for dimensional organization of all
!     computed functions.
!
!  4. all force vectors are prescribed in an inertial coordinate frame
!     which is firxed in space with origin at the bearing center.
!
!  5. all moments are prescribed in body fixed coordinate frames.
!     appropriate transformation matrices from inertial to body fixed
!     coordinates are available within the subroutine.
!
!  6. all input/output variables are transmitted via common blocks.
!
!
      use Parameters                                                  ! base parameters
      use Devices                                                       ! input output
and other fortran devices
      use SubX                                                        ! primary module
to provide interface this subroutine
     use Constants                                                 ! module containing
various constants
      implicit none
!------------------------------------
!  mode 1    icm(1) = -1
!------------------------------------
!

      select case (icm(1))                                              ! icm,jcm->SubX
      case (-1)
         jcm(1)=0                                                     ! jcm(1) must be
set to 1 when using this subroutine
                                                                      ! the subroutine
will never be called if jcm(1)=0
!
!        --------------------------------
!        insert any read/write statements for optional
!        input data
!
!        use fortran read device code = input,
!        and write device code = output,
!        both defined in module Devices
!
!        also set any arbitrary suppression
!        of degrees of freedom and symetry considerations
!        variables kReDOF, kCageDOF, kRaceDOF, kSymetry
!        in module SubX
!        --------------------------------
!
         continue
      case (0)
!-------------------------
!     mode 2    icm(1) = 0
!     --------------------
!
!
!        --------------------------------
!        insert any output to be documented with the
!        initial data output
!
!        use fortran output device code = output,
!        defined in module Devices
!        --------------------------------
!
!
!        --------------------------------
!        perform other one time computations, such as
!        dimensional organization and/or setting values
!        for any constants
!        --------------------------------
!
!        --------------------------------
!        set initial velocities for arbitrary
!        accelerations mode
!        use appropriate variables in module SubX
!        race angular velocity is already set
!        to initial race rpm specified
!        in the main input data
!        --------------------------------
!
         continue
      case (1)
!-------------------------
!     mode 3    icm(1) = 1
!     --------------------
!
!
!        --------------------------------
!        insert coding for appropriate model
!
!        this is the main computing area it
!        must be free of any input/output
!        statements
!        --------------------------------
!
         continue
      case (2)
!-------------------------
!     mode 4    icm(1) = 2
!     --------------------
!
!
!        --------------------------------
!        insert any write statements for
!        documenting output with the time
!        step solutions
!
!        use fortran output device code = output,
!        defined in module Devices
!        --------------------------------
!
         continue
      case (3)
!-------------------------
!     mode 5    icm(1) = 3
!     --------------------
!
!
!        ----------------------------------
!        insert any write statement for
!        any data to be stored in optional
!        data files created by the user
!        at first call (icm(1)=-1)
!
!        this data may be used later by the
!        user to perform additional analysis
!        or to generate additional plots
!        -----------------------------------
!
         continue
      end select
      return
      end
```

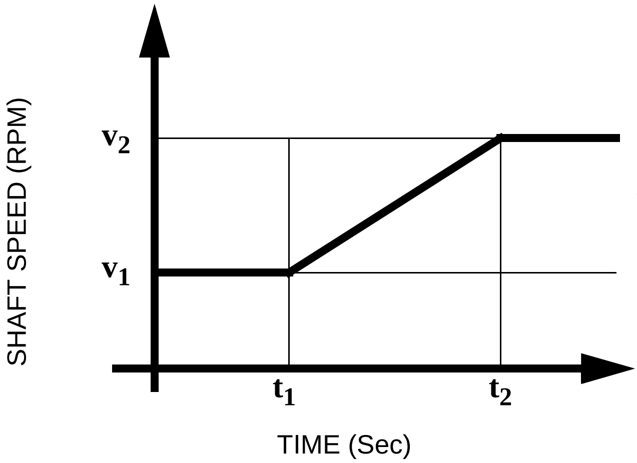
As documented by the comment statements, the procedure basically works in five modes controlled by the flag `icm(1)`, which is set by the calling routine in ADORE. At the time of first call, `icm(1)=-1` and the procedure executed all statements under mode 1 in the above listing. Here the user flag `jcm(1)` must be set equal to 1 if Adrx1 is to be used; in addition all user inputs may be read in this mode. At the second call the flag `icm(1)=0` and any statements under mode 2 are executed. Here any output documentation one time computations may be performed. Examples are nondimensionalizing the variables and computations of certain constants. Variables to used in the later calls must of course be appropriately saved. In subsequent calls the flag `icm(1)=1` and the procedure will execute the statements under mode 3. This is the actual computation mode. Since this part may be called thousands of times, all computations must be coded in the most efficient manner. In addition, the code must be free of any input/output statements. Whenever ADORE documents any print output it will also call Adrx1, if it is being used, with the control flag `icm(1)=2`. Thus any print statements inserted under mode 2 may be executed and the data may be documented with the main print output. In addition to these modes, ADORE also calls Adrx1 with `icm(1)=3` at the end of each time step. The purpose here is to documented any time varying data in a user created data set, which could be set up at the time of first call under mode 1. The purpose of this data set may be to either plot certain variables as a function of time or input the time-varying data to other applications for further modeling.

Actual use of Adrx1 may be best illustrated by the examples presented below.

6.2.1 Adrx1 Example 1: Angular Acceleration on Inner Race

In this example the inner race of a bearing accelerates from a rotation speed v_1 at time t_1 to speed v_2 at time t_2. As shown schematically in figure 74 below, the speed changes linearly; in other words the angular acceleration is constant.

Such an acceleration may be easily programmed in subroutine Adrx1. In the first segment of code listed below first the variables are declared; note that rpm1 and rpm2 are used in place of v_1 and v_2. Then the flag `jcm(1)` is set equal to 1 to trigger use of this subroutine.



Angular Acceleration:

$$acc = \frac{rpm2 - rpm1}{t2 - t1} = \frac{v_2 - v_1}{t_2 - t_1}$$

$$t_1 \le t \le t_2$$

Figure 77. Modeling race acceleration over a prescribed time interval.

Note that all variable declarations include the keyword, “save” to save the variables for future calls to this routine. After the variable declaration the speed and time variables are read in at first call to Adrx1. Thus the code under “mode 1” is as follows:

```
      ....
      ....
      use Parameters
      use Devices
      use SubX
      use Constants
      implicit none
      character*12                      :: jrec    ! record identifier
      real(r8),save                     :: rpm1    ! initial speed (rpm)
      real(r8),save                     :: rpm2    ! final speed (rpm)
       real(r8),save                    :: t1      ! initial time (seconds)
      real(r8),save                     :: t2      ! final time (seconds)
      real(r8),save                     :: acc     ! local variable for acceleration
!------------------------------------
!  mode 1    icm(1) = -1
!------------------------------------
!

      select case (icm(1))                         ! icm,jcm->SubX
      case (-1)
         jcm(1)=1                                   ! set switch to call this subroutine
!
!        --------------------------------
!        insert any read/write statements for optional
!        input data
!
!        use fortran read device code = input,
!        and write device code = output,
!        both defined in module Devices
!        --------------------------------
!
         read(input,*) jrec,rpm1,rpm2,t1,t2
         write(output,101) jrec,rpm1,rpm2,t1,t2
```

```
  101    format(3x,a12,1p,4e11.4)
         continue
       .....
       .....
```

In the next code segment, under “mode 2”, the model is documented in the print output, the variables are nondimensionalized and the angular acceleration is computed. In addition, the initial angular velocity is set and the accelerations are initialized. The initial angular velocity must also be set in the ADORE input data set on Record 9.1. This sets all initial conditions in the bearing corresponding to this initial velocity. The resulting code in Adrx1 will be as follows:

```
       .....
       .....
      case (0)
!-------------------------
!     mode 2    icm(1) = 0
!     --------------------
!
!
!        -------------------------------
!        insert any output to be documented with the
!        initial data output
!
!        use fortran output device code = output,
!        defined in module Devices
!        -------------------------------
!
         write(output,102) rpm1,t1,rpm2,t2
  102    format(/5x,’Race acceleration:’//5x,                          &
     &   ’Initial speed = ’,1p,e11.4,2x,’at time = ’,e11.4/5x,          &
     &   ’Final speed   = ’,e11.4,2x,’at time = ’,e11.4)
!
!        -------------------------------
!        perform other one time computations, such as
!        dimensional organization and/or setting values
!        for any constants
!        -------------------------------
!
         rpm1=rpm1*pi*sTime/30.0_r8          !  pi,sTime->Constants
         rpm2=rpm2*pi*sTime/30.0_r8
         t1=t1/sTime                         ! dimensionless times
         t2=t2/sTime
         acc=(rpm2-rpm1)/(t2-t1)             ! constant race acceleration
!        -------------------------------
!        set initial velocities for arbitrary
!        accelerations mode
!        use appropriate variables in module SubX
!        race angular velocity is already set
!        to initial race rpm specified
!        in the main input data
!        -------------------------------
!
         raceInitAngVel(1,2)=rpm1            ! set initial speed of inner race
         raceAngAcc=zero                     ! initialize race ang acceleration
         continue                            ! raceInitAngVel,raceAngAcc->SubX
      .....
      .....
```

Now in the next code segment, under “mode 3”, the race angular acceleration is simply set when the current time is between t_1 and t_2. The code segment in Adrx1 will simply be:

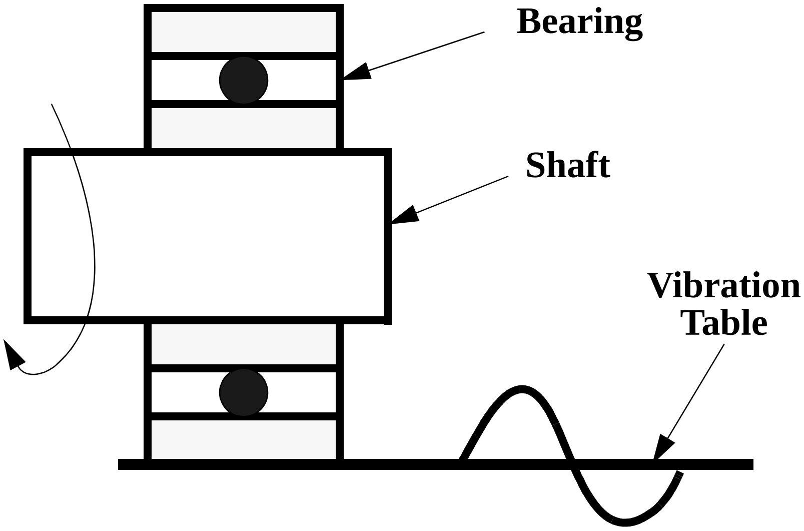
```
      .....
      .....

      case (1)
!-------------------------
!     mode 3    icm(1) = 1
!     --------------------
!
!
!        --------------------------------
!        insert coding for appropriate model
!
!        this is the main computing area it
!        must be free of any input/output
!        statements
!        --------------------------------
!
         if (dimLessTime <= t1) then                    ! dimLessTime->SubX
            raceAngAcc(1,2)=zero
         else if (dimLessTime >= t2) then
            raceAngAcc(1,2)=zero
         else
            raceAngAcc(1,2)=acc
         end if
         continue
      .....
      .....
```

The rest of segments in Adrx1 may not be used in this example. Note that certain variables from modules SubX and Constants are used in the above code.

6.2.2 Adrx1 Example 2: Vibrational Loading

In this example the bearing housing is actually mounted on a vibrating platform. Thus the bearing is subjected to a sinusoidal vibration, as shown schematically in figure 75.



Position:

$$A = A_o \sin \omega t$$

Velocity:

$$\dot{A} = A_o \omega \cos \omega t$$

Acceleration:

$$\ddot{A} = -A_o \omega^2 \sin \omega t$$

Figure 78. Modeling of vibrational loading on the outer race.

Again in the first segment of the code, the variables are declared, the flag jcm(1) is set to 1 and the variables are read in from the input stream:

```
      .....
      .....
      implicit none
      character*12                :: jrec         ! record identifier
      real(r8),save               :: a0           ! amplitude of vibratory motion (m)
      real(r8),save               :: omega        ! vibration frequency (Hz)
!------------------------------------
!  mode 1    icm(1) = -1
!------------------------------------
!

      select case (icm(1))                                              ! icm,jcm->SubX
      case (-1)
         jcm(1)=1                                 ! set switch to call this routine
!
!        --------------------------------
!        insert any read/write statements for optional
!        input data
!
!        use fortran read device code = input,
!        and write device code = output,
!        both defined in module Devices
!        --------------------------------
!
         read(input,*) jrec,a0,omega
         write(output,101) jrec,a0,omega
  101    format(3x,a12,1p,2e11.4)
         continue
      .....
      .....
```

In the second code segment, again the model is documented, variables are nondimensionalized and the initial conditions are set:

```
      .....
      .....
      case (0)
!-------------------------
!     mode 2    icm(1) = 0
!     --------------------
!
!
!        --------------------------------
!        insert any output to be documented with the
!        initial data output
!
!        use fortran output device code = output,
!        defined in module Devices
!        --------------------------------
!
         write(output,102) a0,omega
  102    format(//5x,’Vibrational loading on outer race:’/5x,            &
     &      ’Amplitude of vibration = ’,1p,e11.4,’ m’/5x,                &
     &      ’Frequency of vibration = ’,e11.4,’ Hz’/)
!
!        --------------------------------
!        perform other one time computations, such as
!        dimensional organization and/or setting values
!        for any constants
!        --------------------------------
!
```

```
         a0=a0/sLen                        ! nondimensionalize vibration amplitude
         omega=omega*two*pi*sTime          ! nondimensionalize vibration frequency
!        --------------------------------
!        set initial velocities for arbitrary
!        accelerations mode
!        use appropriate variables in module SubX
!        race angular velocity is already set
!        to initial race rpm specified
!        in the main input data
!        --------------------------------
!
         raceAcc=zero                      ! initialize race acceleration
         raceInitVel(3,1)=a0*omega         ! initial velocity of the outer race
         a0=-a0*omega**2                   ! acceleration amplitude
         continue
      .....
      .....
```

Finally, in the next segment of code in Adrx1 radial acceleration, about the z-axis is applied on the outer race:

```
      .....
      .....

      case (1)
!-------------------------
!     mode 3    icm(1) = 1
!     --------------------
!
!
!        --------------------------------
!        insert coding for appropriate model
!
!        this is the main computing area it
!        must be free of any input/output
!        statements
!        --------------------------------
!
         raceAcc(3,1)=a0*dsin(omega*dimLessTime)       ! dimLessTime->SubX
         continue
      .....
      .....
```

General programming procedures are identical in all user subroutines. Thus the general format used in the above example is also applicable to rest of the user subroutines.

6.3 Subroutine ADRX2

The roller/race-flange contact behavior can be incorporated here in terms of a load-deflection relation. If any such data is available, then the simplified treatment of equivalent Hertzian contact may be replaced by more realistic constitutive relations. Thus the roller flange interactions may be more precisely modeled.

6.4 Subroutine ADRX3

The purpose of this subroutine is to prescribe any force deflection relation for rolling-element-to-cage contact in the cage pocket. Such a relation is often obtained experimentally and, if available, it should replace the simplified Hertz contact analysis used in ADORE.

6.5 Subroutine ADRX4

This subroutine is similar to ADRX3, but it applies to cage/race interactions. Since the load-deflection relation for line contact is often determined experimentally, this subroutine will help implement any available semi-empirical constitutive equation for the cage/race contact.

6.6 Subroutine ADRX5

Variation in roller radius as a function of the axial and circumferential position on the roller surface can be programmed in this subroutine. Thus, roller out-of-roundness, roller coning and similar effects can be very easily programmed.

6.7 Subroutine ADRX6

This subroutine is identical in scope to ADRX5 except that it provides the variation in the radius of the interacting surface of the race. Also, for ball and spherical roller bearings, the variation in curvature across the groove may be programmed in this subroutine.

6.8 Subroutine ADRX7

Any arbitrary traction-slip relation for the rolling element to race contact may be prescribed in this subroutine. Aside from prescribing an equation, actual traction-slip data may be inserted in a tabular form and the data may be interpolated for appropriate conditions in the rolling element to race contact. When this subroutine is activated all standard traction models for the rolling element to race contact are bypassed and the data prescribed herein is used.

6.9 Subroutine ADRX8

This subroutine is called only once after all the input data is read in from the data file DATA.txt in a start-up run. The purpose of the routine is to prescribe arbitrary geometrical imperfections on rolling elements and in the cage pockets. Since the number of variables here is quite large, this data is collected from this subroutine, while providing the user with the freedom of reading in only the variables of interest. After the initial run, the data is stored in file MASTER. In a continuation run, therefore, this data is not required.

6.10 Subroutine ADRX9

Time-varying output data may be stored in the user data set SOL9 in this subroutine. Most solutions generated in ADORE are defined in data module “Solutions”.

6.10.1 Adrx9 Example: Arbitrary Output in File SOL9

The objective of this example is to extract all local heat generations in the bearing as computed in ADORE. The data is to used subsequently in finite element for the races and cage to compute overall temperature distribution as a result of heat generated in the bearing. Thus all heat generations, contact size and contact locations, as required by the finite element model, are collected from the module “Solutions” and written in the data set SOL9 as a function of time. Note that all solutions are generally dimensionless. Hence appropriate scale factors, available in data module “Constants” are applied before writing the data to the data set. Following is the listing of Adrx9 for this example:

```
      subroutine Adrx9(kStep)
!
!-------------------------------------------------------
!  Adrx9   arbitrary output file set up by the user to
!          extract any number of output variables in the
!          optional output file SOL9
!
!-------------------------------------------------------
      use Devices
      use Constants
      use BrgGeom
      use Solutions
      implicit none
      integer        :: kStep               ! input call identifier defined as follows:
                                           !  =-1, value at first call.
                                           ! for writing any header info before data
                                           !  >= 0, time step number
      integer       :: i,j,k               ! loop indices

      if (kStep < 0) then
!------------------------------------
!        process output at first call
!        ----------------------------
         continue
      else
!-------------------------------------------------
!        write desired output to file SOL9
!        device code is pfile(9) in module Devices
!        all output variables in module Solutions
!        -----------------------------------------
         aList(1)=powerLoss*cPV
         aList(2)=pLt*cPV
         write(pfile(9),100) kStep,aList(1:2)  ! step #, total power loss
                                               ! & time-averaged power loss
  100    format(2x,’Step ’,i6,1p,2e14.5))
!------------------------
!        re/race contacts
!        ----------------
         do i=1,2                               ! start race loop
            do j=1,nRe                            ! nRe->BrgGeom, rolling element loop
               aList(1)=reSV(i,j)*cPV           ! re/race intantaneous power loss,
                                                ! cPV->Constants
               aList(2)=reSVt(i,j)*cPV          ! re/race time-averaged power loss
               aList(3)=conWidthA(i,j)*sLen     ! major contact half width,
                                                ! sLen->Constants
               aList(4)=conWidthB(i,j)*sLen     ! minor contact half width
               aList(5:7)=conPosR(1:3,i,j)*sLen    ! contact position (x,y,z)
                                                  ! rel to race ctr in race frame
               write(pfile(9),101) i,j,aList(1:7)
  101          format(2x,2i3,1p,8e14.5)
            end do
         end do
!------------------------
!        re/cage contacts
!        ----------------
         do i=1,nCseg                                       ! start cage segment loop
                                                            ! nCseg->BrgGeom
            do j=1,nPoc(i)                                  ! start cage pocket loop,
                                                            ! nPoc->BrgGeom
               do k=1,nPocSur(j,i)                 ! start pocket guide
                                                   ! surface loop, nPocSur->BrgGeom
```

```
                  aList(1)=pocSV(k,j,i)*cPV        ! instantaneous power loss
                                                   ! at pocket contact
                  aList(2)=pocSVt(k,j,i)*cPV         ! time-averaged power loss
                                                     ! at pocket contact
                  aList(3:4)=pocConWidth(1:2,k,j,i)*sLen ! major & minor half width
                  aList(5:6)=pocConPos(1:2,k,j,i)*sLen  ! contact pos rel to poc ctr
                                                        ! or guide land ctr
                  aList(7:8)=pocConAng(1:2,k,j,i)*cAng  ! contact angle
                  write(pfile(9),102) i,j,k,aList(1:8)
  102             format(2x,3i3,1p,10e14.5)
               end do
            end do
         end do
!-------------------------
!        cage/race contact
!        -----------------
         do i=1,nCseg                        ! cage segment loop, nCseg->BrgGeom
            do j=1,nGL                       ! guide surface loop, nGL->BrgGeom
              aList(1)=cLandSV(j,i)*cPV       ! cage/race instantaneous heat generation
              aList(2)=cLandSVt(j,i)*cPV      ! cage/race time-averaged heat generation
               aList(3:4)=cLandConWidth(1:2,1,j,i)*sLen  ! major & minor half widths
               aList(5:7)=cLandConPosC(1:3,1,j,i)*sLen  ! contact pos (x,y,z) rel to
                                                        ! cage ctr in cage frame
               aList(8:10)=cLandConPosR(1:3,1,j,i)*sLen ! contact pos (x,y,z) rel to
                                                        ! race ctr in race frame
               write(pfile(9),102) i,j,iCageGuide(j,i),aList(1:10)
                                          ! iCageGuide->BrgGeom, cage guidance type
            end do
         end do
      end if
      return
      end
```