Generation and Tooth Contact Analysis of Spiral Bevel Gears With Predesigned Parabolic Functions of Transmission Errors

Faydor L. Litvin and Hong-Tao Lee

GRANT NAG3-783
NOVEMBER 1989
Generation and Tooth Contact Analysis of Spiral Bevel Gears With Predesigned Parabolic Functions of Transmission Errors

Faydor L. Litvin and Hong-Tao Lee
University of Illinois at Chicago
Chicago, Illinois

Prepared for
Propulsion Directorate
USAARTA-AVSCOM and
NASA Lewis Research Center
under Grant NAG3-783

NASA
National Aeronautics and Space Administration
Office of Management
Scientific and Technical Information Division
1989
# Contents

1 INTRODUCTION                                      1
1.1 Introduction ......................................... 1
1.2 Transmission Errors And Its Compensation .......... 4

2 GLEASON'S SPIRAL BEVEL GEARS                    13
2.1 Gleason System ....................................... 13
2.2 Head Cutters ........................................ 16
2.3 Coordinate Systems and Sign Conventions .......... 24
2.4 Generated Tooth Surfaces                      27

3 SYNTHESIS OF SPIRAL BEVEL GEARS                 37
3.1 Gear Machine-Tool Settings ......................... 37
3.1.1 Preliminary Considerations ...................... 38
3.1.2 Gear Cutting Ratio ................................ 38
3.1.3 Cutter Tip Radius, Radial Setting, and Cradle Angle .... 40
3.2 Determination of the Mean Contact Point on the Gear Tooth Surfaces ..... 40
3.3 Local Synthesis ...................................... 49
3.3.1 Preliminary Considerations ....................... 50
3.3.2 Relations Between Directions of the Paths of the Mean Contact Point in its Motion over the Gear and Pinion Tooth Surfaces .......... 54
3.3.3 Principal Curvatures and Directions of the Pinion Tooth Surface at the Mean Contact Point ............................................. 58
3.3.4 First Order Characteristics ..................... 63
3.3.5 Principal Curvatures and Directions of the Pinion Cutter Surface at the Mean Contact Point ......................................... 66
3.4 Pinion Machine-Tool Settings ......................... 70
3.4.1 Determination of Pinion Cutting Ratio ............. 70
3.4.2 Determination of parameters $E_m$ and $L_m$ ........ 79
3.4.3 Determination of Pinion Radial Setting and Cradle Angle .......... 80

4 CONCLUSION                                      82

A GEOMETRY AND KINEMATICS OF GEARS IN THREE DIMENSIONS 84
A.1 Concept of Surfaces .................................. 84
A.2 Relations Between Principal Curvatures and Directions for Mating Surfaces .... 94
NOMENCLATURE

Index

Upper case English characters except "T" and digit numbers indicate surfaces.

Lower case English characters indicate coordinate systems.

\(c\)  tool surface \((c = G, P)\)
\(w\)  work surface \((w = 1, 2)\)
\(r, q\)  tool surfaces or work surfaces
\(G\)  gear tool surface
\(P\)  pinion tool surface
1  pinion surface
2  gear surface
\(I\)  first principal
\(II\)  second principal

Matrix

\([A]\)  3 by 4 symmetric augmented matrix which relates principal curvatures and directions for mating surfaces
\([B]\)  4 by 1 matrix representing homogenous coordinates of point \(B\)
\([L_{ab}]\)  3 by 3 matrix describing the transformation of vector from the \(S_b\) coordinate system to \(S_a\) coordinate system
\([M_{ab}]\)  4 by 4 matrix describing the transformation of coordinates from the \(S_b\) coordinate system to \(S_a\) coordinate system
\([N]\)  3 by 1 matrix representing components of normal vector \(\vec{N}\)
\( [n] \) 3 by 1 matrix representing components of unit normal vector \( \vec{n} \)

\( [\omega] \) 3 by 1 matrix representing components of angular velocity vector \( \vec{\omega} \)

**Vector**

\( \vec{B} \) position vector of point \( B \) on a surface

\( \vec{B}_u \) \( \partial \vec{B}/\partial u \)

\( \vec{B}_v \) \( \partial \vec{B}/\partial v \)

\( \vec{e}_i, \vec{e}_n \) unit vectors along the principal directions of the surface at the contact point

\( \vec{i}, \vec{j}, \vec{k} \) base vectors along axes \( X, Y, \) and \( Z, \) respectively

\( \vec{N} \) normal vector of point \( B \) on a surface

\( \vec{n} \) unit normal vector of point \( B \) on a surface

\( \vec{n}_u \) \( \partial \vec{n}/\partial u \)

\( \vec{n}_v \) \( \partial \vec{n}/\partial v \)

\( \vec{v}^{(C,W)} \) slide velocity of surfaces \( \Sigma_C \) and \( \Sigma_W \)

\( \vec{v} \) transfer velocity

\( r_{1} \vec{v}^{(1)}, r_{2} \vec{v}^{(2)} \) velocity vectors of contact point in its motion over the pinion and gear surfaces, respectively

\( \vec{\omega} \) angular velocity

\( \vec{\omega}^{(F,Q)} \) relative angular velocity of surface \( F \) with respect to surface \( Q \)

\( \vec{r} \) tangent vector

**English Upper Case**

\( A \) mean pitch cone distance
$A_0, A_1, A_2$  coefficient of a quadratic equation

$A, B$  auxiliary parameters

$B$  point on a surface

$C^n$  class of a function

$E, F, G$  auxiliary parameters for first fundamental form

$E_m$  machining offset

$E^3$  three-dimensional space

$F$  zero function

$I$  first fundamental form

$II$  second fundamental form

$I$  Interval

$L$  generating planar curve for a sphere

$L, M, N$  auxiliary parameters for second fundamental form

$L_m$  vector sum of machine center to back and sliding base

$M$  middle point on the gear surface

$N$  number of teeth

$P$  plane

$R$  radius of a circle

$R_{cx}, R_{cz}$  $x$ and $z$ coordinates, respectively, of the center of a circle in the $S_c$ coordinate system

$S$  coordinate system

$T$  the smaller absolute value of $A$ and $B$

$V_{c_i}^{(w,c)}$  the projection of $V^{(w,c)}$ on the $e_{c_i}$

$V_{c_H}^{(w,c)}$  the projection of $V^{(w,c)}$ on the $e_{c_H}$

$W$  point width

$V_{2_I}^{(1)}, V_{2_H}^{(1)}$  the projections of vector $V^{(1)}$ on vectors $e_{2_I}$ and $e_{2_H}$, respectively
$X_MCB$  machine center to back

$X_{SB}$  sliding base

**English Lower Case**

$a$  constant  (Chapter 1)

$a$  semimajor axis of the contact ellipse  (Chapter 3 and Appendix A)

$a_{ij}$  element of matrix $[A]$  $(i = 1, 2, 3  j = 1, 2, 3)$

$b$  constant  (Chapter 1)

$b$  semiminor axis of the contact ellipse  (Chapter 3 and Appendix A)

$b_1, b_2$  auxiliary variables

$c$  clearance

$c_{11}, c_{12}, c_{13}$  auxiliary variables

$d_a$  average diameter of gear cutter

$d_1, d_2, d_3$  auxiliary variables

$f_1, f_2$  auxiliary variables

$m_{x_Q}$  gear ratio

$m'_{x_Q}$  derivative of gear ratio with respective to $\phi_Q$

$q$  cradle angle

$r$  tip radius of the cutter

$s$  radial setting

$t$  semimajor axis of the contact ellipse

$t_1, t_2, t_4$  auxiliary variables

$u$  surface coordinates of a cone surface

$u_{11}, u_{12}, u_{21}$  auxiliary variables
auxiliary variables

Non-English Upper Case

$\Sigma$ surface

$\Gamma$ shaft angle

$\Upsilon$ angle measured counterclockwise from the root to the tangent of the path on the gear surface

$\Re$ ratio constant

$\Re$ open rectangle

$\Delta$ discriminant of an equation

Non-English Lower Case

$\alpha$ orientation angle of ellipse

$\beta$ mean spiral angle

$\delta$ dedendum angle

$\epsilon$ specified tolerance value

$\gamma$ root angle

$\kappa$ principal curvature

$\kappa_\Delta \kappa_{2\Sigma} - \kappa_{1\Sigma}$

$\kappa_\Sigma \kappa_1 + \kappa_\Pi$

$\kappa_\Delta \kappa_1 - \kappa_\Pi$

$\kappa_n$ normal curvature

$\kappa_r$ relative normal curvature of the mating surface
<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\lambda$</td>
<td>surface coordinate of a surface of revolution</td>
</tr>
<tr>
<td>$\mu$</td>
<td>pitch angle</td>
</tr>
<tr>
<td>$\nu_1, \nu_2$</td>
<td>angles formed between vectors $\vec{r}^{(1)}$ and $\vec{e}_1$, and $\vec{r}^{(2)}$ and $\vec{e}_2$, respectively</td>
</tr>
<tr>
<td>$\omega$</td>
<td>angular velocity</td>
</tr>
<tr>
<td>$\phi_c$</td>
<td>turn angle of the cradle when the work is being cut</td>
</tr>
<tr>
<td>$\phi_w$</td>
<td>rotation angle of the work while it is being cut</td>
</tr>
<tr>
<td>$\phi'_w$</td>
<td>rotation angle of one member while it is being in meshing with another member of a pair of gears</td>
</tr>
<tr>
<td>$\phi'_2(\phi'_1)$</td>
<td>transmission function, the rotation angle of the gear in terms of that of the pinion in a pair of meshing gears</td>
</tr>
<tr>
<td>$\dot{\phi}'_2(\phi'_1)$</td>
<td>transmission function of a pair conjugate gear</td>
</tr>
<tr>
<td>$\Delta \phi'_2(\phi'_1)$</td>
<td>transmission error function</td>
</tr>
<tr>
<td>$(\Delta \phi'_2)^{(1)}$</td>
<td>predesigned parabolic function of transmission errors</td>
</tr>
<tr>
<td>$(\Delta \phi'_2)^{(2)}$</td>
<td>linear function of transmission errors induced by misalignment</td>
</tr>
<tr>
<td>$\psi'_1, \Delta \psi'_2$</td>
<td>expressions of $\phi'_1$ and $\Delta \phi'_2$ in a new coordinate system</td>
</tr>
<tr>
<td>$\psi$</td>
<td>blade angle</td>
</tr>
<tr>
<td>$\theta$</td>
<td>surface coordinate of a cone surface and a surface of revolution</td>
</tr>
<tr>
<td>$\sigma_{x\phi}$</td>
<td>angle measured counterclockwise from $\vec{e}<em>{x_1}$ to $\vec{e}</em>{\phi_1}$</td>
</tr>
<tr>
<td>$\tau$</td>
<td>auxiliary variable, $\theta \pm q \pm \phi_c$</td>
</tr>
<tr>
<td>$\varepsilon$</td>
<td>elastic approach</td>
</tr>
<tr>
<td>$\varpi$</td>
<td>angle formed by the tangent to the curvature and first principal curvature</td>
</tr>
</tbody>
</table>
SUMMARY

A new approach for determination of machine-tool settings for spiral bevel gears is proposed. The proposed settings provide a predesigned parabolic function of transmission errors and the desired location and orientation of the bearing contact. The predesigned parabolic function of transmission errors is able to absorb piece-wise linear functions of transmission errors that are caused by the gear misalignment and reduce the gear noise. The gears are face-milled by head cutters with conical surfaces or surfaces of revolution.

A computer program for simulation of meshing, bearing contact and determination of transmission errors for misaligned gear has been developed.
CHAPTER 1

INTRODUCTION

1.1 Introduction

The most important criteria of quality of meshing and contact of gears are the low level of noise and the sufficient dimensions and location of the bearing contact. Sometimes these requirements are contradictory and can be achieved by a compromise in the process of gear synthesis. Such a method of synthesis for spiral bevel gears has been developed in this report.

Traditionally, Gleason's spiral bevel gears are designed and manufactured with non-conjugate tooth surfaces. By varying machine-tool settings the transmission errors can be of different forms, which included a piece-wise linear function, an "S" curve, and a parabolic function, symmetrical or otherwise. Only a parabolic function with gear lagging is preferred. The problem encountered is that it is very difficult to reduce the level of a parabolic function of transmission errors with gear lagging.

Litvin et al. [1] proposed a method for generation of spiral bevel gears with conjugate tooth surfaces. Ideally such conjugate pair provides zero transmission errors. In practice, spiral bevel gears are frequently required to operate under misalignment caused by mounting tolerances and deflections. Using the Tooth Contact Analysis (TCA) programs we have found that the conjugate spiral bevel gears cause lead functions of transmission errors — strong monotonous increasing or decreasing functions for a cycle of meshing. These functions may be considered as linear functions
or almost linear functions (Figure 1). Due to gear misalignment the bearing contact can be shifted from the desired location even to the tooth edge. For this reason it is necessary to control also the location and dimensions of the bearing contact.

There is an opportunity to reach these goals if the gears will be designed as non-conjugate pairs that transform rotation with a predesigned parabolic function of transmission errors. Then, as it will be proven in the next section, a linear function of transmission errors will be absorbed and the sensitivity of the gears to misalignment will be reduced.

The determination of pinion machine-tool settings is based on the local synthesis of the gears proposed by Litvin [2, 3, 4]. The local synthesis must satisfy the following requirements:

1. The gear surfaces are in tangency at the chosen mean contact point.
2. The tangent to the path of contact has the prescribed direction at the mean contact point.
3. The contact ellipse for the tooth surfaces has the desired dimensions at the mean contact point.
4. The transmission function $\phi_2(\phi_1)$ has the prescribed value at the mean contact point and its second derivative is negative on gear convex side and positive on gear concave side. Here, $\phi_1$ and $\phi_2$ are the rotation angles of the pinion and gear while they are being cut, respectively.

Requirement 4 means that the function of transmission errors is a parabolic one with gear lagging within the neighborhood of the mean contact point.

Traditionally, a pair of Gleason's gears is generated by two cones. In some cases the pinion is generated by a surface of revolution instead of a cone surface to obtain better bearing contact and to avoid an edge contact. Both cases are investigated and the machine-tool settings are determined according to the local synthesis and predesigned function of transmission errors.
transmission errors

rotation angle of pinion

Figure 1: Transmission errors of conjugate gears caused by misalignment.
1.2 Transmission Errors And Its Compensation

In theory a pair of mating gears transforms rotation with a constant gear ratio

\[ m_{21} = \frac{\omega_2}{\omega_1} = \frac{N_1}{N_2} \quad (1.1) \]

where \( \omega_1 \) and \( \omega_2 \) are the angular velocities of the gears

\( N_1 \) and \( N_2 \) are the numbers of teeth of pinion and gear, respectively

Therefore, the transmission function is expected to be linear for ideal gears, i.e.,

\[ \tilde{\phi}_2'(\phi'_1) = \frac{N_1}{N_2} \phi'_1 \quad (1.2) \]

However, the actual function \( \phi_2'(\phi'_1) \) is always different from \( \tilde{\phi}_2'(\phi'_1) \) except at the mean contact point. The transmission errors are defined as the difference of theoretical and actual functions of transmission functions, i.e.,

\[ \Delta \phi_2'(\phi'_1) = \phi_2'(\phi'_1) - \tilde{\phi}_2'(\phi'_1) = \phi_2'(\phi'_1) - \frac{N_2}{N_1} \phi'_1 \quad (1.3) \]

In general the transmission errors of gears may occur due to the following four reasons [5]:

1. The gears cannot exactly transform rotation described by equation (1.2) because of the method of their generation. Spiral bevel gears and hypoid gears that are generated by Gleason methods are good examples for this case.
2. The gear axes are misaligned or the gear shafts are deflected. Zhang, in his dissertation [5], has proved that the deflected gear shafts can be modeled as misaligned gear axes. Spur gears, helical gears, and conjugate spiral bevel gears are very sensitive to misalignment.

3. Heat treatment deviation of the real gear surface is one of the most important factors in surface distortion.

4. The elastic deformation of gear tooth surfaces under applied load.

Cases 1 and 2 among the above-mentioned are the main sources of transmission errors. They will be discussed later. The topics of 3 and 4 are beyond the scope of this report and will not be discussed.

For a pair of conjugate gears under misalignment, the investigation results in that the transmission function \( \phi'_2(\phi'_1) \) becomes a discontinuous piece-wise function that is linear or almost linear for each cycle of meshing as shown in Figure 2. The corresponding transmission errors determined by equation (1.3) are also an approximately piece-wise linear function as shown in Figure 3. Such functions cause a discontinuity in the regular tooth meshing and usually impact at the transfer point.

There is another type of function of transmission errors that is a piece-wise parabolic function as shown in Figure 4. This type of transmission errors does not cause a discontinuity of regular tooth meshing at transfer points. Gears with this type of transmission errors are not so sensitive to misalignment. This statement is based on an investigation into the interaction of a parabolic function with a linear function.

Consider that a pair of gears is predesigned with a parabolic function of transmission errors. This function may be represented by

\[
(\Delta \phi'_2)^{(1)} = a(\phi'_1)^2
\]  

(1.4)
Figure 2: Transmission functions of gears under misalignment.
Figure 3: Transmission errors caused by gear Misalignment.
Figure 4: A piece-wise parabolic function of transmission errors.
The level of transmission errors is $a(2\pi/N_1)^2$.

Misalignment induces a linear function of transmission errors. It may be represented by

$$\left(\triangle \phi'_2\right)^{(2)} = b\phi'_1$$  \hspace{1cm} (1.5)

Since $\left(\triangle \phi'_2\right)^{(1)}$ and $\left(\triangle \phi'_2\right)^{(2)}$ are very small, the principle of superposition can be applied for the interaction of functions $\left(\triangle \phi'_2\right)^{(1)}$ and $\left(\triangle \phi'_2\right)^{(2)}$. Therefore, the resulting function is

$$\triangle \phi'_2 = \left(\triangle \phi'_2\right)^{(1)} + \left(\triangle \phi'_2\right)^{(2)} = a(\phi'_1)^2 + b\phi'_1$$  \hspace{1cm} (1.6)

Equation (1.6) can be rewritten in a new coordinate system by (Figure 5)

$$\triangle \psi'_2 = a(\psi'_1)^2$$  \hspace{1cm} (1.7)

where

$$\triangle \psi'_2 = \triangle \phi'_2 + \frac{b^2}{4a} \quad \psi'_1 = \phi'_1 + \frac{b}{2a}$$  \hspace{1cm} (1.8)

From equation (1.7) we know that although the misalignment occurs, the resulting function of transmission errors represents the same parabolic function that has been translated with respect to the given parabolic function. This means that the predesigned parabolic function $\left(\triangle \phi'_2\right)^{(1)}$ will absorb the linear function $\left(\triangle \phi'_2\right)^{(2)}$ induced by misalignment. The level of transmission errors remains the same since the parabolic function of each tooth translates the same amount.
Figure 5: Interaction of parabolic and linear functions.
Misalignment changes the path of contact. The locations of transfer points are shifted to an edge. The amount of the shift is determined by $b/2a$. In general, the absolute value of $b$ increases if the amount of misalignment increases. It is possible that an unfavorable ratio $b/2a$ may cause one of the transfer points to be off the tooth surface and that the function of transmission errors, $\Delta \psi'$, will become a discontinuous function for every cycle of meshing (Figure 6). To avoid this, the level of predesigned function of transmission error, or the absolute value of $a$, should be chosen with the expected level of transmission errors caused by misalignment.
Figure 6: Discontinued parabolic function of transmission errors.
CHAPTER 2

GLEASON'S SPIRAL BEVEL GEARS

2.1 Gleason System

The Gleason Works, Rochester, New York, is one of the leading companies that produces equipment for manufacture of bevel and hypoid gears. William Gleason built the first machine in 1874 to cut bevel gears with straight teeth [6]. During the following years, the Gleason Works has developed a set of machines to generate spiral bevel gears. The basic construction (Figure 7) of a cutting machine consists of three major parts: the frame, the cradle, and the sliding base [7, 8].

When cutting starts, the work is plunged into the cutter. As the cutter rotates through the blank, a relative rolling motion is produced between the cradle and the work spindle to generate the tooth surface. While the cutter rolls out of engagement with the work, the cradle reverses rapidly, the sliding base on which the work is mounted is translated with respect to the cutter, and the work is indexed ahead for cutting the next tooth. This sequence of operations is repeated until the last tooth is cut.

In the process of cutting, the head-cutter rotates about its axis, and the axis generates in the cradle coordinate system a cylindrical surface. We may imagine that the cutter generates a tooth of crown gear as shown in Figure 8. Therefore, the cutting process corresponds to the motion of the gear rolling on a crown rack. The angular velocity of the head-cutter about its axis is not related with the generating motions and depends only on the desired velocity of cutting. This is
Figure 7: An isometric view for a gear generator.
Figure 8: Cutting spiral gear teeth on the basic crown rack.
an important advantage of the Gleason methods of manufacture. Another advantage is that the same method for generation can be used as well for grinding. Grinding is essential for producing gears with hardened tooth surfaces and of high quality.

### 2.2 Head Cutters

Traditionally straight-sided blades have been applied in practice. The blades of the cutter generate cone surfaces while the cutter rotates its axis. Figure 9 shows these two cones. A current point $B$ on the cone surface is represented in the coordinate system $S_c$ as follows:

\[
    \vec{B}_c = \begin{bmatrix}
        B_{cx} \\
        B_{cy} \\
        B_{cz} \\
        1
    \end{bmatrix} = \begin{bmatrix}
        r \cot \psi - u \cos \psi \\
        u \sin \psi \sin \theta \\
        u \sin \psi \cos \theta \\
        1
    \end{bmatrix}
\]  

(2.1)

where $u = \overline{AB}$ and $\theta$ are the surface coordinates, $r$ is the tip radius of the cutter, and $\psi$ is the blade angle. For the inside blade of the cutter, $\psi$ is an acute angle. For the outside blade of the cutter, $\psi$ is an obtuse angle.

Using equations (A.5) and (2.1) (provided $u \sin \psi \neq 0$), we obtain the equations of the unit normal to the cone surface.

\[
    \vec{n}_c = \begin{bmatrix}
        n_{cx} \\
        n_{cy} \\
        n_{cz}
    \end{bmatrix} = \pm \begin{bmatrix}
        \sin \psi \\
        \cos \psi \sin \theta \\
        \cos \psi \cos \theta
    \end{bmatrix}
\]  

(2.2)

The total differential of vector $\vec{B}_c$ is
Figure 9: Generated cone surfaces of the head-cutter.
The total differential of vector $\vec{n}_c$

\[
[dB_c] = \begin{bmatrix}
-\cos \psi \, du \\
\sin \psi (\sin \theta \, du + u \cos \theta \, d\theta) \\
\sin \psi (\cos \theta \, du - u \sin \theta \, d\theta)
\end{bmatrix}
\] (2.3)

Equations (A.26), (2.3), and (2.4) yield

\[
\frac{0}{-\cos \psi \, du} = \frac{\pm \cos \psi \cos \theta \, d\theta}{\sin \psi (\sin \theta \, du + u \cos \theta \, d\theta)} = \frac{\mp \cos \psi \sin \theta \, d\theta}{\sin \psi (\cos \theta \, du - u \sin \theta \, d\theta)} = -\kappa_{1,2}
\] (2.5)

Equation (2.5) is satisfied if

\[
du \, d\theta = 0
\] (2.6)

One of the principal directions corresponds to $du = 0$; the other one to $d\theta = 0$. They can be represented by equations

\[
\vec{\varepsilon}_{1c} = \frac{\delta \vec{B}_c}{\delta \theta},
\]

\[
\vec{\varepsilon}_{2c} = \frac{\partial \vec{B}_c}{\partial u},
\] (2.7) (2.8)
Equations (2.3) and (2.7) yield

\[ \vec{e}_{\tau} = \pm \begin{bmatrix} 0 \\ \cos \theta \\ - \sin \theta \end{bmatrix} \]  

(2.9)

Plugging \( du = 0 \) into equations (2.5), we have

\[ \kappa_{\tau} = \mp \frac{1}{u \tan \psi} \]  

(2.10)

The sense of the principal curvature relies on the chosen direction of the normal.

Similarly, the unit vector of the second principal direction is

\[ \vec{e}_{\eta'} = \pm \begin{bmatrix} - \cos \psi \\ \sin \psi \sin \theta \\ \sin \psi \cos \theta \end{bmatrix} \]  

(2.11)

The principal curvature is

\[ \kappa_{\eta'} = 0 \]  

(2.12)

In addition to the cone surface, a tool provided by a surface of revolution is considered here. This surface of revolution is generated by an circular arc that rotates about the cutter axis. Such a surface can be applied as a grinding wheel or as a surface of a tool with curved blades.

Suppose the generating planar curve \( L \) (Figure 10) is an arc of a circle of radius \( R \) centered at point \((R_{c\tau}, 0, R_{c\psi})\). The spherical surface is generated by the circle in the rotational motion.
Figure 10: Generating arc circle for the curved edge of the head cutter.
about the $Z_c$-axis. Consider an auxiliary coordinate system $S_{c'}$ which is rigidly connected to the generating circle. Initially $S_{c'}$ and $S_c$ coincide. The generating curve may be represented in the coordinate system $S_{c'}$ with the matrix equation

$$\begin{bmatrix} B'_{c_x} \\ B'_{c_y} \\ B'_{c_z} \\ 1 \end{bmatrix} = \begin{bmatrix} R_{c_x} + R \cos \lambda \\ 0 \\ R_{c_z} + R \sin \lambda \\ 1 \end{bmatrix}$$ (2.13)

where $\lambda$ is the varied parameter for planar curve $L$. The parameter $\lambda$ lies within the following intervals:

Inside blade \( \begin{cases} 0 < \lambda < \pi/2, & \text{if } L \text{ is concave down;} \\
\pi < \lambda < 3\pi/2, & \text{if } L \text{ is concave up;} \end{cases} \)

Outside blade \( \begin{cases} 3\pi/2 < \lambda < 2\pi, & \text{if } L \text{ is concave down;} \\
\pi/2 < \lambda < \pi, & \text{if } L \text{ is concave up.} \end{cases} \)

The auxiliary coordinate system $S_{c'}$ rotates about the $Z_c$ axis and the coordinate transformation from $S_{c'}$ to $S_c$ is (Figure 11)

$$\bar{B}_c = \begin{bmatrix} B_{c_x} \\ B_{c_y} \\ B_{c_z} \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \theta & \sin \theta & 0 \\ 0 & -\sin \theta & \cos \theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} B'_{c_x} \\ B'_{c_y} \\ B'_{c_z} \\ 1 \end{bmatrix}$$ (2.14)

Equations (2.13) and (2.14) yield
Figure 11: Coordinate transformations to generate spherical surfaces.
Using equations (A.5) and (2.15), the unit normal to this spherical surface may be represented by

\[
\tilde{\mathbf{n}}_c = \begin{bmatrix}
    n_{c_x} \\
    n_{c_y} \\
    n_{c_z}
\end{bmatrix} = \pm \begin{bmatrix}
    \cos \lambda \\
    \sin \lambda \sin \theta \\
    \sin \lambda \cos \theta
\end{bmatrix}
\]  

(2.16)

According to Rodrigues' formula, the principal directions on the generating surface correspond to \( \delta \lambda = 0 \) and \( \delta \theta = 0 \), respectively. The unit vector of the principal direction corresponding to \( \delta \lambda = 0 \) is

\[
\tilde{\imath}_c = \pm \begin{bmatrix}
    0 \\
    \cos \theta \\
    -\sin \theta
\end{bmatrix}
\]  

(2.17)

The principal curvature is

\[
\kappa_1 = \pm \frac{\sin \lambda}{R_{c_x} + R \sin \lambda}
\]  

(2.18)

The unit vector of the principal direction corresponding to \( \delta \theta = 0 \) is

\[
\tilde{\jmath}_c = \pm \begin{bmatrix}
    -\sin \lambda \\
    \cos \lambda \sin \theta \\
    \cos \lambda \cos \theta
\end{bmatrix}
\]  

(2.19)
The principal curvature is

\[ \kappa_n = \pm \frac{1}{R} \]  

(2.20)

2.3 Coordinate Systems and Sign Conventions

Left-hand gear-members are usually cut by the counterclockwise motion of the cradle that carries the head-cutter. This motion is viewed from the front of the cradle and from the back of the work spindle. Cutting is performed from the toe to the heel. Figure 12 shows the top and front views of the machine when a left-hand gear-member is cut.

Right-hand gear-members are usually cut by motions that are opposite to the motions of the left-hand members being cut. Cutting is performed from the heel to the toe. Figure 13 shows the top and front views of the machine for this case.

We set up five coordinate systems in either case. Coordinate system \( S_c \) is rigidly connected to the head cutter, coordinate system \( S_w \) is rigidly connected to the work, and coordinate systems \( S_m, S_p \) and \( S_a \) are rigidly connected to the frame. Axes \( Z_m \) and \( Z_p \) coincide with the root line and pitch line, respectively. Axis \( X_m \) is perpendicular to the generatrix of the root cone of the work. Axis \( X_p \) is perpendicular to the generatrix of the pitch cone of the work. Axes \( Z_a \) and \( Z_w \) coincide. Origin \( O_m \) is located at the machine center, and origins \( O_a \) and \( O_p \) are located at the apex of the pitch cone of the work.

Three special machine-tool settings, which are the machining offset, machine center to back, and the sliding base, are used only for the generation of pinions. The machining offset, denoted by \( E_m \), is the shortest distance between the cradle axis and pinion axis. In figures 12 and 13, \( L_m \) represents a vector sum of machine center to back, \( X_{MCB} \), and the sliding base, \( X_{SB} \). The change
Figure 12: Top and front views of a left-hand gear generator.
Figure 13: Top and front views of a right-hand gear generator.
of machine center to back is directed parallel to the pinion axis and the direction of the sliding base is pointed parallel to the cradle axis.

The sign conventions for machine-tool settings are given in Table 1.

2.4 Generated Tooth Surfaces

The generated surface $\Sigma_W$ is an envelope of the family of the tool surface $\Sigma_C$. Surfaces $\Sigma_W$ and $\Sigma_C$ contact each other at every instant along a line which is a spatial curve. Surface $\Sigma_W$ is conjugate with $\Sigma_C$. In mathematical sense the determination of a conjugate surface is based on the theory of an envelope of a family of given surfaces. In differential geometry, to determine $\Sigma_W$ we must find:

(a) the family of surfaces $\Sigma_\phi$ generated by the given surface $\Sigma_C$ in the $S_w$ coordinate system
and

(b) the envelope $\Sigma_W$ of the family of surfaces $\Sigma_{\phi}$. 

The matrix representation of the family of surfaces $\Sigma_{\phi}$ may be represented by the matrix equation

$$[B_w] = [M_{wc}] [B_c] \quad (2.21)$$

where $[M_{wc}]$ is a matrix which describes the transformation of coordinates from the "old" coordinate system $S_c$ to the "new" coordinate system $S_w$. From Figures 12 and 13, we obtain

$$[M_{wc}] = [M_{wa}] [M_{ap}] [M_{pm}] [M_{mc}] \quad (2.22)$$

We can obtain $[M_{wa}]$ from Figure 14 as

$$[M_{wa}] = \begin{bmatrix}
\cos \phi_w & \pm \sin \phi_w & 0 & 0 \\
\mp \sin \phi_w & \cos \phi_w & 0 & 0 \\
0 & 0 & 1 & 0 \\
0 & 0 & 0 & 1
\end{bmatrix} \quad (2.23)$$

where $\phi_w$ is the rotation angle of the work while it is being cut. Here the upper sign corresponds to the generation of a left-hand spiral bevel gear that is shown in Figure 12, and the lower sign corresponds to the generation of a right-hand spiral bevel gear shown in Figure 13. Henceforth we will obey this notation.

The transformation matrices $[M_{ap}]$ and $[M_{pm}]$ can be obtained from Figures 12 and 13 as
Figure 14: The rotation angle of the work while it is being cut.
\[
[M_{ap}] = \begin{bmatrix}
\cos \mu & 0 & \sin \mu & 0 \\
0 & 1 & 0 & 0 \\
-\sin \mu & 0 & \cos \mu & 0 \\
0 & 0 & 0 & 1
\end{bmatrix}
\] (2.24)

\[
[M_{pm}] = \begin{bmatrix}
\cos \delta & 0 & -\sin \delta & -L_m \sin \delta \\
0 & 1 & 0 & \pm E_m \\
\sin \delta & 0 & \cos \delta & L_m \cos \delta \\
0 & 0 & 0 & 1
\end{bmatrix}
\] (2.25)

where \( \mu \) and \( \delta \) are the pitch angle and dedendum angle of the work, respectively. To derive the transformation matrix \([M_{mc}]\), let us apply an auxiliary coordinate system \(S_x\) rigidly connected to the tool (Figure 15). Thus

\[
[M_{mc}] = [M_{ms}] [M_{sc}]
\]

\[
= \begin{bmatrix}
1 & 0 & 0 & 0 \\
0 & \cos \phi_c & \pm \sin \phi_c & 0 \\
0 & \mp \sin \phi_c & \cos \phi_c & 0 \\
0 & 0 & 0 & 1
\end{bmatrix}
\begin{bmatrix}
1 & 0 & 0 & 0 \\
0 & \cos q & \mp \sin q & \mp s \sin q \\
0 & \pm \sin q & \cos q & s \cos q \\
0 & 0 & 0 & 1
\end{bmatrix}
\] (2.26)

where \( \phi_c \) is the turn angle of the cradle while the work is cut, and \( s \) is the radial setting. The determination of the envelope \( \Sigma_W \) of the locus of surfaces \( \Sigma_4 \) is based on necessary and sufficient conditions of envelope existence that have been developed in the classical Differential Geometry. A simpler method representation for determination of necessary condition of \( \Sigma_W \) existence is based
Figure 15: Auxiliary coordinate system $S_s$. 

Left-Hand Member

Right-Hand Member
on the geometric property of conjugate surfaces: at points of tangency of the generating surface \( \Sigma_C \) and the generated surface \( \Sigma_W \) the common unit normal \( \vec{n} \) to the surfaces is perpendicular to the slide velocity \( \vec{V}^{(CW)} \) of these two surfaces \([9, 10]\). This is given by the scalar product

\[
\vec{n} \cdot \vec{V}^{(CW)} = 0
\]  

(2.27)

In the modern theory of gearing, equation (2.27) is called the equation of meshing. This equation is of fundamental importance in the kinematics of gearing. Since equation (2.27) is valid in any reference system, we will derive the equation of meshing in the \( S_m \) coordinate system. Let us designate \( \tau r V_m^{(C)} \) and \( \tau r V_m^{(W)} \) the transfer velocities of common contact points \( B_m \) on the cutter and the work, respectively. Thus

\[
\vec{V}_m^{(CW)} = \tau r V_m^{(C)} - \tau r V_m^{(W)}
\]  

(2.28)

The cradle rotates about the \( Z_m \) axis with the angular velocity \( \vec{\omega}_m^{(C)} \) (Figures 12 and 13); therefore, the transfer velocity \( \tau r V_m^{(C)} \) is represented by the equation

\[
\tau r V_m^{(C)} = \vec{\omega}_m^{(C)} \times \vec{B}_m
\]  

(2.29)

The work rotates about the \( Z_w \) axis with the angular velocity \( \vec{\omega}_m^{(W)} \) (Figures 12 and 13) which does not pass through the origin \( O_m \) of the \( S_m \) coordinate system. It is known from the theoretical mechanics that the angular velocity \( \vec{\omega}_m^{(W)} \) may be substituted by an equal vector \( \vec{\omega}_m^{(W)} \) which passes through \( O_m \) and a vector-moment

\[
O_m \vec{O}_a \times \vec{\omega}_m^{(W)}
\]  

(2.30)
Note that the moment has the same unit and physical meaning as linear velocity. Thus

\[ \tau_r v_m^{(w)} = \omega_m^{(w)} \times \bar{B}_m + \overrightarrow{O_mO_a} \times \bar{\omega}_m^{(w)} \]  \hspace{1cm} (2.31)

It is evident from Figures 12 and 13 that

\[ [\omega_m^{(\gamma)}] = \mp \omega^{(\gamma)} \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \]  \hspace{1cm} (2.32)

\[ [\omega_m^{(w)}] = \pm \omega^{(w)} \begin{bmatrix} -\sin \gamma \\ 0 \\ \cos \gamma \end{bmatrix} \]  \hspace{1cm} (2.33)

\[ \overrightarrow{O_mO_a} = \begin{bmatrix} 0 \\ \pm E_m \\ -L_m \end{bmatrix} \]  \hspace{1cm} (2.34)

In equation (2.33) \( \gamma \) is the root angle of the work. Substituting equation (2.29)–(2.34) into equation (2.28), we obtain

\[ \overrightarrow{V}_m^{(cw)} = \begin{bmatrix} \omega^{(w)}(E_m \pm B_{my}) \cos \gamma \\ \pm \omega^{(c)B_{mx}} + \omega^{(w)}[(B_{mx} \mp L_m) \sin \gamma \mp B_{mx} \cos \gamma] \\ \mp \omega^{(c)B_{my}} \pm \omega^{(w)}(B_{my} - E_m) \sin \gamma \end{bmatrix} \]  \hspace{1cm} (2.35)
The coordinates of the common contact points $B_m$ may be obtained from equations of the generating surface $\Sigma_C$. Then we get

$$[B_m] = [M_{mc}] [B_c] \quad (2.36)$$

The common unit normals $n_m$ may be represented by the unit normals to $\Sigma_C$. Therefore

$$[n_m] = [L_{mc}] [n_c] \quad (2.37)$$

where $[L_{mc}]$ is the rotation matrix obtained by eliminating of the last row and last column of the corresponding matrix $[M_{mc}]$.

Hence, if $\Sigma_C$ is a cone surface, substituting equations (2.1) and (2.26) into equation (2.36) we obtain

$$\begin{bmatrix} B_{m_x} \\ B_{m_y} \\ B_{m_z} \end{bmatrix} = \begin{bmatrix} \tau \cot \psi - u \cos \psi \\ u \sin \psi \sin \tau \mp s \sin(q - \phi_c) \\ u \sin \psi \cos \tau + s \cos(q - \phi_c) \end{bmatrix} \quad (2.38)$$

where $\tau = \theta \mp q \pm \phi_c$. Substituting equations (2.2) and (2.26) into equation (2.37) the unit normals may be represented as

$$\begin{bmatrix} n_{m_x} \\ n_{m_y} \\ n_{m_z} \end{bmatrix} = \pm \begin{bmatrix} \sin \psi \\ \cos \psi \sin \tau \\ \cos \psi \cos \tau \end{bmatrix} \quad (2.39)$$

Similarly if $\Sigma_C$ is a spherical surface, substituting equations (2.15) and (2.26) into equation (2.36), we obtain
Substituting equations (2.16) and (2.26) into equation (2.37) the unit normals may be represented as

\[
\begin{bmatrix}
B_{m_x} \\
B_{m_y} \\
B_{m_z} \\
1
\end{bmatrix} = \begin{bmatrix}
R_{c_x} + R \cos \lambda \\
(R_{c_x} + R \sin \lambda) \sin \tau \mp s \sin(q - \phi_c) \\
(R_{c_x} + R \sin \lambda) \cos \tau + s \cos(q - \phi_c) \\
1
\end{bmatrix} (2.40)
\]

Using equations (2.27), (2.35), (2.38), and (2.39), we may obtain the equation of meshing for the case that \( \Sigma_C \) is a cone surface by

\[
(u - r \cot \psi \cos \psi) \cos \gamma \sin \tau + s [(m_{cw} - \sin \gamma) \cos \psi \sin \theta \mp \cos \gamma \sin \psi \sin(q - \phi_c)]
\]

\[
\mp E_m (\cos \gamma \sin \psi + \sin \gamma \cos \psi \cos \tau) - L_m \sin \gamma \cos \psi \sin \tau = 0
\]

(2.43)

For \( \Sigma_C \) being a spherical surface, using equations (2.27), (2.35), (2.40), and (2.41), the equation of meshing is represented by

\[
(R_{c_x} \cos \lambda - R_{c_x} \sin \lambda) \cos \gamma \sin \tau + s [(m_{cw} - \sin \gamma) \sin \lambda \sin \theta \mp \cos \gamma \cos \lambda \sin(q - \phi_c)]
\]

\[
\pm E_m (\cos \gamma \cos \lambda + \sin \gamma \sin \lambda \cos \tau) - L_m \sin \gamma \sin \lambda \sin \tau = 0
\]

(2.44)
Equations (2.43) and (2.44) relate the generating surface coordinates \((u \text{ and } \theta)\) for a cone surface or \((\lambda \text{ and } \theta)\) for a surface of revolution with the turn angle \(\phi_c\).
CHAPTER 3

SYNTHESIS OF SPIRAL BEVEL GEARS

3.1 Gear Machine-Tool Settings

We designate for the following discussions the gear-generating tool surface by $\Sigma_G$, the generated gear surface by $\Sigma_2$, the pinion-generating tool surface by $\Sigma_P$, and the generated pinion surface $\Sigma_1$. A parameter with the subscript $i$ indicates that it is related to surface $\Sigma_i$. To set up the gear machine-tool settings, the following data are considered as given:

- $\Gamma$: shaft angle
- $N_2$: gear tooth number
- $N_1$: pinion tooth number
- $\gamma_2$: gear root angle
- $A$: mean pitch cone distance
- $\beta$: mean spiral angle
- $\psi_o$: blade angle for gear cutter
- $d_o$: average diameter of gear cutter
- $w_o$: point width
3.1.1 Preliminary Considerations

We prefer to calculate the values of pitch angles and dedendum angles rather than obtain them from the blank design summary because the data in the summary are not accurate enough for computer calculations.

The gear pitch angle is represented by

\[ \mu_2 = \arctan \frac{\sin \Gamma}{\frac{N_1}{N_2} + \cos \Gamma} \]  

The pinion pitch angle is

\[ \mu_1 = \Gamma - \mu_2 \]  

The dedendum angles are

\[ \delta_1 = \mu_1 - \gamma_1, \quad \delta_2 = \mu_2 - \gamma_2 \]  

3.1.2 Gear Cutting Ratio

The process of gear generation is based on the imaginary meshing of a crown gear with the member-gear. The instantaneous axis of rotation by such meshing coincides with the pitch line, axis \( Z_p \), that is shown in Figures 12 and 13. The generating surface \( \Sigma_G \), which may be imagined as the surface of the crown gear, and the to be generated gear surface \( \Sigma_2 \) contact each other at a line.
at every instant. The ratio of angular velocities of the crown gear and the being generated gear (the cutting ratio) remains constant while the spatial line of contact moves over surfaces Σ₁ and Σ₂.

The determination of cutting ratio is based on following consideration. The angular velocity in relative motion is

\[ \omega^{(2)} = \omega^{(1)} - \omega^{(3)} = a\vec{k}_p \]  \hspace{1cm} (3.4)

This means that vectors \( \omega^{(2)} \) and \( \vec{k}_p \) are collinear. Since equation (3.4) is valid in any reference frame, let us derive it in the \( S_m \) coordinate system. From Figures 12 and 13 we have

\[
\vec{k}_p = \begin{bmatrix}
\sin \delta_2 \\
0 \\
\cos \delta_2
\end{bmatrix} \hspace{1cm} (3.5)
\]

By replacing the superscript 'c' by 'c' in equation (2.32) and 'w' by '2' in equation (2.33), we may represent in matrix form angular velocities \( \omega^{(c)} \) and \( \omega^{(2)} \). Consequently, we obtain the following equation

\[
\frac{\pm \omega^{(c)} \pm \omega^{(2)} \sin \gamma_2}{\sin \delta_2} = \frac{\pm \omega^{(2)} \cos \gamma_2}{\cos \delta_2} \hspace{1cm} (3.6)
\]

Equation (3.6) results in that

\[
m_{c2} = \frac{\omega^{(c)}}{\omega^{(2)}} = \frac{\sin \mu_2}{\cos \delta_2} \hspace{1cm} (3.7)
\]
3.1.3 Cutter Tip Radius, Radial Setting, and Cradle Angle

Figure 16 shows that the inside and outside tip radii of the head-cutter are represented by

\[ r_o = \frac{1}{2}(d_o + w_o) \]  \hspace{1cm} (3.8)

Figure 16 shows the front view of the installation of the head cutter. From the relations between the lengths and angles of the triangle \( O_mO_cM_o \), we may express the radial setting \( s_o \) and cradle angle \( q_o \) as follows:

\[ s_o = \sqrt{\frac{d_o^2}{4} + A^2 \cos^2 \delta - d_o A \cos \delta \sin \beta} \]  \hspace{1cm} (3.9)

and

\[ q_o = \arccos \frac{A^2 \cos^2 \delta + s_o^2 - \frac{d_o^2}{4}}{2A s_o \cos \delta} \]  \hspace{1cm} (3.10)

3.2 Determination of the Mean Contact Point on the Gear Tooth Surfaces

The gear and pinion surfaces of spiral bevel gears are in point contact at every instant. The mean contact point is the center of the bearing contact and its location is selected generally at the middle of the working depth on the gear tooth. Figure 17 shows a gear tooth surface. Section \( AD \) is the gear tip and it is parallel to the generatrix of the root cone of the pinion. Section \( BC \) is the pinion tip and it is parallel to the root line of the gear. The working area is within \( ABCD \). In the \( S_p \) coordinate system, line \( AD \) may be represented by
Figure 16: The front view of the installation of the head cutter
Figure 17: Gear tooth surface.
\[ B_{p_x} = B_{p_x} \tan \delta_1 - c \quad (3.11) \]

where \( c \) is the clearance and \( \delta_1 \) is the pinion dedendum angle. Line \( \overline{BC} \) is represented by

\[ B_{p_x} = -B_{p_x} \tan \delta_2 + c \quad (3.12) \]

The mean contact point is located on a line which passes through the middle point of the two points at which the normal section of the gear surface intersects line \( \overline{AD} \) and line \( \overline{BC} \), respectively. In addition, the mean contact point must be on the gear surface. This means that it must satisfy the equation of meshing for the gear being generated by the tool. We will use these two requirements to determine the location of the mean contact point and represent the procedure of derivations as follows:

**STEP 1:** The initial guess for \( \theta_\alpha \) is

\[ \theta_\alpha = \pm(\phi_\alpha - \beta + \pi/2) \]

**STEP 2:** Determination of \( u_\alpha \) based on the given \( \theta_\alpha \)

Equation (2.43) determines parameter \( u_\alpha \). The turn angle \( \phi_\alpha \) is set to zero when equation (2.43) is applied.

**STEP 3:** Representation of gear tooth surface in coordinate systems \( S_c \) and \( S_p \)

Equation (2.1) determines the gear tooth surface in coordinate system \( S_c \). The gear tooth surface may be represented in the \( S_p \) coordinate system as follows:

\[ [B_p] = [M_{pm}][M_{mc}][B_c] \quad (3.13) \]
Transformation matrix \([M_{pm}]\) may be obtained from equation (2.25) by setting \(E_m\) and \(L_m\) to zero. Equation (2.26) determines matrix \([M_{mc}]\).

**STEP 4: Determination of middle point**

The \(z\) coordinate of the middle point \(M\) of the two points, which are the intersections of the normal section of the gear tooth surface and the gear tooth tips, may be obtained by

\[
M_{pz} = \frac{B_{pz}(\tan \delta_1 - \tan \delta_2)}{2}
\]  

(3.14)

The above equation is derived by dividing the sum of equations (3.11) and (3.12) by 2.

**STEP 5: Judgement of \(u_G\)**

The acceptable value of \(u_G\) is determined by the following criterion:

\[
|B_{pz} - M_{pz}| < \epsilon
\]

where \(\epsilon\) is a specified tolerance value. If the above criterion is satisfied, parameters \(u_G\) and \(\theta_G\) of the mean contact point are determined. Otherwise, repeat **STEP 2 to STEP 5** by a new value of \(\theta_G\) until the criterion is satisfied.

As a matter of fact, the determination of the location of the mean contact point is the same as that of a root of equation

\[
B_{pz} - M_{pz} = 0
\]

(3.15)
The new value of the $\theta_G$ in Step 5 depends on which method is used to solve this equation. In this study Newton’s method was used.

So far we have already determined parameters $u_\sigma$ and $\theta_G$ of the mean contact point. Repeating the task done in Step 3, we have the coordinates of the mean contact point $B$. The common unit normal $\vec{n}$ to surfaces $\Sigma_G$ and $\Sigma_2$ at the mean contact point $B$ is

$$[n_p] = [L_{pm}]{[L_{mc}][n_c]}$$

(3.16)

where matrix $[L_{pm}]$ is obtained by deleting the fourth row and column from matrix $[M_{pm}]$ given by equation (2.25). Similarly, we may obtain rotation matrix $[L_{mc}]$ from matrix $[M_{mc}]$ by equation (2.26). Although the unit normal has two directions, we choose the direction corresponding to the positive sign in equation (2.2) regardless of the hand of the gear. The principal directions at the mean contact point $B$ on the gear tool surface $\Sigma_G$ in the $S_p$ coordinate system may be obtained by the following coordinate transformation:

$$[e_{a,II_p}] = [L_{pm}][L_{mc}][e_{a,II_c}]$$

(3.17)

Here we choose positive sign in equation (2.9) as the direction of the first principal direction. The second principal direction is determined by rotating of the first principal direction about unit normal by $90^\circ$.

The principal curvatures and directions at the mean contact point $B$ on the gear surface $\Sigma_2$ may be derived according to the formula expressed in Section A.2. Note that surfaces $\Sigma_G$ and $\Sigma_2$ are in line contact. To apply these formula, we may consider that surfaces $\Sigma_2$ and $\Sigma_G$ are equivalent to surfaces $\Sigma_T$ and $\Sigma_Q$, respectively, in Section A.2.
The derivation of the principal curvatures and directions at the mean contact point $B$ is performed as follows:

**STEP 1:** We represent the angular velocity $\omega^{(2)}_p$ in the $S_v$ coordinate system as follows:

$$\begin{bmatrix} \omega^{(2)}_p \end{bmatrix} = \pm \begin{bmatrix} -\sin \mu_2 \\ 0 \\ \cos \mu_2 \end{bmatrix}$$

This is a direct result from drawings of Figures 12 and 13.

**STEP 2:** We represent the angular velocity $\omega^{(G)}_p$ in the $S_p$ coordinate system as follows:

$$\begin{bmatrix} \omega^{(G)}_p \end{bmatrix} = \pm \begin{bmatrix} \cos \delta_2 \\ 0 \\ \sin \delta_2 \end{bmatrix}$$

This is also a direct result from Figures 12 and 13.

**STEP 3:** The relative angular velocity $\omega^{(2G)}_p$ is represented by

$$\omega^{(2G)}_p = \omega^{(2)}_p - \omega^{(G)}_p = \pm \omega^{(2)}$$

$$= \begin{bmatrix} -\sin \mu_2 + m_{\alpha 2} \cos \delta_2 \\ 0 \\ \cos \mu_2 + m_{\alpha 2} \sin \delta_2 \end{bmatrix}$$

**STEP 4:** The transfer velocity of the mean point $B$ on surface $\Sigma_\mathring{G}$ is

$$\mathbf{tr} \mathbf{V}_p^{(G)} = \omega^{(G)}_p \times \mathbf{B}_p$$

$$= (3.21)$$
**Step 5:** The transfer velocity of the mean point $B$ on surface $\Sigma_2$ is

$$t_s T_p^{(2)} = \omega_p^{(2)} \times \vec{B}_p$$

(3.22)

**Step 6:** The relative velocity of the mean point $B$ is

$$V_p^{(2G)} = t_s T_p^{(2)} - t_r T_p^{(G)}$$

(3.23)

**Step 7:** The projection of $V_p^{(2G)}$ on the $\varepsilon_{G,I_p}$ is

$$V_{G,I}^{(2G)} = V_p^{(2G)} \cdot \varepsilon_{G,I_p}$$

(3.24)

**Step 8:** The projection of $V_p^{(2G)}$ on the $\varepsilon_{G,II_p}$ is

$$V_{G,II}^{(2G)} = V_p^{(2G)} \cdot \varepsilon_{G,II_p}$$

(3.25)

**Step 9:** Using equation (A.33), we obtain

$$a_{13} = -\kappa_{G,I} V_{G,I}^{(2G)} - [\omega_p^{(2G)} \cdot \vec{n}_p \varepsilon_{G,I_p}]$$

(3.26)
STEP 10: Using equation (A.35), we have

\[ a_{23} = -\kappa_{\alpha \Pi} V_{\alpha \Pi}^{(2G)} - [\omega_p^{(2G)} \bar{n}_P \epsilon_{\alpha \Pi \Pi}] \]  \hspace{1cm} (3.27)

STEP 11: Using equation (A.36), we obtain

\[ a_{33} = \kappa_{\alpha I} \left(V_{\alpha I}^{(2G)} \right)^2 + \kappa_{\alpha \Pi} \left(V_{\alpha \Pi}^{(2G)} \right)^2 - [\bar{n}_P \omega_p^{(2G)} - V_p^{(2G)}] - \bar{n}_P \cdot \left(\omega_p^{(2)} \times \sigma V_p^{(G)} - \omega_p^{(G)} \times \sigma V_p^{(2)} \right) \]  \hspace{1cm} (3.28)

Note that \( m_{\alpha 2} = 0 \)

STEP 12: To determine the principal directions at point B on gear surface, we first use equation (A.40). Thus

\[ \tan 2\sigma_{\alpha I} = \frac{2a_{11} a_{23}}{a_{23}^2 - a_{13}^2 + (\kappa_{\alpha I} - \kappa_{\alpha \Pi}) a_{33}} \]  \hspace{1cm} (3.29)

Rotating unit vector \( \bar{e}_{\alpha I} \) about the unit normal vector \( \bar{n} \) by \(-\sigma_{\alpha I}\), we may obtain unit vector \( \bar{e}_{\alpha I} \). Rotating unit vector \( \bar{e}_{\alpha I} \) about the unit normal vector \( \bar{n} \) by \( \pi/2 \), we may have unit vector \( \bar{e}_{\alpha \Pi} \).

STEP 13: Using equations (A.41) and (A.42), we may determine the principal curvatures on the gear surfaces as follows:

\[ \kappa_{\alpha I} - \kappa_{\alpha \Pi} = \frac{a_{23}^2 - a_{13}^2 + (\kappa_{\alpha I} - \kappa_{\alpha \Pi}) a_{33}}{a_{33} \cos 2\sigma_{\alpha I}} \]  \hspace{1cm} (3.30)

\[ \kappa_{\alpha I} + \kappa_{\alpha \Pi} = (\kappa_{\alpha I} + \kappa_{\alpha \Pi}) - \frac{a_{13}^2 + a_{23}^2}{a_{33}} \]  \hspace{1cm} (3.31)
STEP 14: Eliminating $\kappa_{2n}$ by considering the sum of equations (3.30) and (3.31) together and then dividing the sum by 2, we can determine $\kappa_{2i}$. Eliminating $\kappa_{2i}$ by dividing the difference of equations (3.31) and (3.30) by 2, we can determine $\kappa_{2n}$.

3.3 Local Synthesis

The determination of pinion machine-tool settings is based on the idea of local synthesis of gear tooth surfaces proposed by Litvin [2, 3, 4]. The goal of local synthesis for meshing of spiral bevel gears is to satisfy the following requirements:

1. The gear tooth surfaces must contact each other at the prescribed mean contact point $B$.
2. The contact ellipse for the gear tooth surface must have the desired dimensions at point $B$.
3. The tangent to the contact path must have the prescribed direction at point $B$.
4. The instant gear ratio $m_{z1}(\phi)$ and its derivative $m'_{z1}(\phi)$ must have the prescribed values at point $B$.

The local synthesis for the gear tooth surfaces connects the concept of meshing and the concept of bearing contact. It provides the optimal conditions of meshing for the gear tooth surfaces being in mesh at, and within the neighborhood of, the mean contact point $B$. The local synthesis needs the information on the characteristics of the tooth surfaces of the zero, first, and second orders.

Starting the local synthesis we already know the location of the mean contact point $B$ on the gear surface, the unit normal to the gear tooth surface at point $B$, the principal curvatures and directions at point $B$ on the gear tooth surface.

We will consider the local synthesis in a fixed coordinate system $S_f$. Figure 18 shows the relations among $S_f$, fixed coordinate systems which are attached to the frame of the gear generator,
and fixed coordinate systems connected to the frame of the pinion generator. From Figure 18 we know that $S_f$ and $S_{p(G)}$ coincide with each other. Therefore, the coordinates of the mean contact point $B$, the orientation of the surface unit normal, and the principal directions at the point $B$ on the gear surface are known since they have been determined in the $S_{p(G)}$ system.

3.3.1 Preliminary Considerations

Spiral bevel gears transform rotation motion between intersecting shafts with an instantaneous point contact of surfaces. It corresponds to the second case discussed in Section A.2. Some elements of matrix $[A]$ shown in equation (A.30) are not related with the principal curvatures and directions of the pinion surface; therefore, they may be derived at the stage where the principal curvatures and directions are not known yet. We will consider that all derivations are performed in $S_f$ coordinate system. Throughout the rest of the report, we will drop the subscript if it is considered in the $S_f$ coordinate system.

The following representation is the result of direct observation of drawings of Figure 18.

\[
\omega^{(1)} = \pm \omega^{(1)} \begin{bmatrix}
\sin \mu_1 \\
0 \\
\cos \mu_1
\end{bmatrix}
\] (3.32)

\[
\omega^{(2)} = \pm m_2 \omega^{(1)} \begin{bmatrix}
-\sin \mu_2 \\
0 \\
\cos \mu_2
\end{bmatrix}
\] (3.33)

Recall that the upper sign in the equations corresponds to a left-hand member. As far as a pair of spiral bevel gears is concerned, the hands of the spiral must be opposite; a left-hand gear (pinion) and a right-hand pinion (gear) constitute a pair. Therefore, if we take the upper sign in
\{ \text{Left-Hand Gear} \}
\{ \text{Right-Hand Pinion} \}

\{ \text{Right-Hand Gear} \}
\{ \text{Left-Hand Pinion} \}

\text{Figure 18: Coordinate systems for local synthesis.}
equation (3.32), we must pick up the lower sign in equation (3.33). The relative angular velocity is

$$\omega^{(12)} = \omega^{(1)} - \omega^{(2)}$$  \hfill (3.34)

The transfer velocity of the mean contact point $B$ on surface $\Sigma_1$ is

$$t_{\text{tr}} \vec{V}^{(1)} = \vec{\omega}^{(1)} \times \vec{B}$$  \hfill (3.35)

The transfer velocity of the mean contact point $B$ on surface $\Sigma_2$ is

$$t_{\text{tr}} \vec{V}^{(2)} = \vec{\omega}^{(2)} \times \vec{B}$$  \hfill (3.36)

The relative velocity of the mean point $B$ is

$$\vec{V}^{(12)} = t_{\text{tr}} \vec{V}^{(1)} - t_{\text{tr}} \vec{V}^{(2)}$$  \hfill (3.37)

The projection of $\vec{V}^{(12)}$ on the vector $\vec{e}_{2_1}$ is

$$V_{2_1}^{(12)} = \vec{V}^{(12)} \cdot \vec{e}_{2_1}$$  \hfill (3.38)

The projection of $\vec{V}^{(12)}$ on the vector $\vec{e}_{2_II}$ is

$$V_{2_II}^{(12)} = \vec{V}^{(12)} \cdot \vec{e}_{2_II}$$  \hfill (3.39)
Let surfaces $\Sigma_1$ and $\Sigma_\tau$, $\Sigma_2$ and $\Sigma_Q$, be equivalent, respectively. Equation (A.33) yields

$$a_{31} = -\kappa_2, V_{2r}^{(12)} - [\omega(12) \tilde{n} \tilde{e}_{2r}]$$  \hspace{1cm} (3.40)

Using equation (A.35), we obtain

$$a_{32} = -\kappa_2, V_{2n}^{(12)} - [\omega(12) \tilde{n} \tilde{e}_{2n}]$$  \hspace{1cm} (3.41)

Equation (A.36) yields

$$a_{33} = \kappa_2, \left( V_{2r}^{(12)} \right)^2 + \kappa_2, \left( V_{2n}^{(12)} \right)^2 - [\tilde{n} \omega(12) \tilde{V}^{(12)}]$$

\[-\tilde{n} \cdot \left( \omega^{(1)} \times \hat{t} \tilde{V}^{(2)} - \omega^{(2)} \times \hat{t} \tilde{V}^{(1)} \right) + \left( \omega^{(1)} \right)^2 m_{21} \left( \tilde{n} \times \tilde{k}_2 \right) \cdot \tilde{B}\]  \hspace{1cm} (3.42)

where $\tilde{k}_2$ is the unit vector along the axis of rotation of the gear. It is represented by (Figure 18)

$$[\tilde{k}_2] = \begin{bmatrix} -\sin \mu_2 \\ 0 \\ \cos \mu_2 \end{bmatrix}$$  \hspace{1cm} (3.43)

In general, spiral bevel gears are designed and manufactured with non-conjugate tooth surfaces. Varying the machine-tool settings it is possible to obtain a lead function of transmission errors, a parabolic function with pinion lagging, or a parabolic function with gear lagging. Only a parabolic
function with gear lagging is good for applications. Therefore, for the convex side of gear tooth $m_{21}'$ we must provide a negative value, and for the concave side of gear tooth $m_{21}'$ must be positive. The absolute value of $m_{21}'$ controls the level of the transmission errors. We consider $m_{21}'$ as an input.

On the gear surface a path of contact that appears almost straight and substantially vertical to the root may fully satisfy the operating requirements in many cases; however, it should not be assumed that this is true for all cases. Sometimes a different direction or shape may be preferable [11]. The tendency of the direction of the contact path may be determined by the relative velocity $\tau_{21}'$ at the mean contact point on the gear surface. Let $\nu_z$ denote the angle between the unit vector $e_z$ at the mean contact point on the gear surface and the direction of tangent at the same point to the path of contact. The relation between the principal directions and the direction of the contact path may be represented as follows (Figure 19):

$$\nu_z = \Theta + \sigma_2$$

The angle $\Theta$ is measured counterclockwise from the root to the tangent of the path. This angle is considered as an input.

3.3.2 Relations Between Directions of the Paths of the Mean Contact Point in its Motion over the Gear and Pinion Tooth Surfaces

Figure 20 shows the common tangent plane to the gear and pinion surfaces at the mean contact point $B$. The notations in Figure 20 are as follows:

- $e_{11}$ and $e_{22}$: unit vectors of the principal directions on the gear surface
- $\vec{V}^{(12)}$: sliding velocity at point $B$
- $\vec{r}_V^{(1)}$: velocity vector of contact point $B$ in its motion over the pinion surface
- $\vec{r}_V^{(2)}$: velocity vector of contact point $B$ in its motion over the gear surface
Figure 19: The direction of the contact path.
Figure 20: Common plane at the mean contact point.
\[ r_{\text{V}_2}^{(1)} \] and \[ r_{\text{V}_2}^{(1)} \] are the projections of vector \( r_{\text{V}}^{(1)} \) on vectors \( \vec{e}_2 \) and \( \vec{e}_{2i} \).

\( \nu_1 \) and \( \nu_2 \): angles formed between vectors \( r_{\text{V}}^{(1)} \) and \( \vec{e}_2 \), \( r_{\text{V}}^{(2)} \) and \( \vec{e}_{2i} \), respectively.

The relation between angles \( \nu_1 \) and \( \nu_2 \) depends on parameters in motion and the principal curvatures of the gear tooth surface. For derivations we will use the following equations:

\[ r_{\text{V}}^{(2)} = r_{\text{V}}^{(1)} + r_{\text{V}}^{(12)} \] (3.45)

that yields

\[ r_{\text{V}_2}^{(2)} = r_{\text{V}_2}^{(1)} + r_{\text{V}_2}^{(12)} \] (3.46)

\[ r_{\text{V}_{2i}}^{(2)} = r_{\text{V}_{2i}}^{(1)} + r_{\text{V}_{2i}}^{(12)} \] (3.47)

From the geometric relations shown in Figure 20 we have

\[ r_{\text{V}_{2i}}^{(2)} = r_{\text{V}_{2i}}^{(12)} \tan \nu_2 \] (3.48)

\[ r_{\text{V}_{2i}}^{(1)} = r_{\text{V}_{2i}}^{(12)} \tan \nu_1 \] (3.49)

Substituting equation (3.48) and (3.49) into equation (3.47), and then substituting equation (3.46) into (3.47), we obtain an expression for \( r_{\text{V}_2}^{(1)} \) in terms of \( r_{\text{V}_2}^{(12)} \), \( r_{\text{V}_{2i}}^{(12)} \), \( \nu_1 \), and \( \nu_2 \) as follows

\[ r_{\text{V}_2}^{(1)} = \frac{r_{\text{V}_{2i}}^{(12)} - r_{\text{V}_{2i}}^{(12)} \tan \nu_2}{\tan \nu_2 - \tan \nu_1} \] (3.50)
According to equation (A.29), \( r_{V_{2}^{(1)}} \) and \( r_{V_{2}^{(1)}} \) are related as follows:

\[
a_{31} r_{V_{2}^{(1)}} + a_{32} r_{V_{2}^{(1)}} = a_{33}
\]

(3.51)

Here surface \( \Sigma_2 \) is equivalent to surface \( \Sigma_2 \); surface \( \Sigma_1 \) to surface \( \Sigma_2 \). Substituting equation (3.49) into equation (3.51), we have

\[
(a_{31} + a_{32} \tan \nu_1) r_{V_{2}^{(1)}} = a_{33}
\]

(3.52)

Finally, combining equations (3.50) and (3.52), we have the relation between angles \( \nu_1 \) and \( \nu_2 \)

\[
\tan \nu_1 = \frac{(a_{32} + a_{31} r_{V_{2}^{(1)}}) \tan \nu_2 - a_{31} r_{V_{2}^{(1)}}}{a_{32} (r_{V_{2}^{(1)}} - r_{V_{2}^{(1)}} \tan \nu_2) + a_{33}}
\]

(3.53)

3.3.3 Principal Curvatures and Directions of the Pinion Tooth Surface at the Mean Contact Point

The derivation of principal curvatures and directions of the pinion tooth surface at the mean contact point is based on the following procedure.

**Step 1:** Representation of \( A \) and \( B \) in terms of coefficients \( a_{11}, a_{12}, \) and \( a_{22} \)

We recall that the lengths of semiaxes of the contact ellipse, \( a \) and \( b \), are determined by parameters \( A, B, \) and \( \varepsilon \) (see Section A.4).

The sum of equations (A.31) and (A.34) yields
\[ a_{11} + a_{22} = \kappa_{2\Delta} - \kappa_{1\Sigma} \quad (3.54) \]

Substituting equation (A.31) by equation (A.34) we obtain

\[ a_{11} - a_{22} = \kappa_{2\Delta} - \kappa_{1\Delta} \cos 2\sigma_{12} \quad (3.55) \]

We may represent parameter \( \mathcal{A} \) in equation (A.54) in terms of \( a_{11} \), \( a_{12} \), and \( a_{22} \) as follows:

\[ \mathcal{A} = -\frac{1}{4} \left[ (a_{11} + a_{22}) + \sqrt{(a_{11} - a_{22})^2 + 4a_{12}^2} \right] \quad (3.56) \]

Also, the representation of parameter \( \mathcal{B} \) in equation (A.55) in terms of \( a_{11} \), \( a_{12} \), and \( a_{22} \) gives

\[ \mathcal{B} = -\frac{1}{4} \left[ (a_{11} + a_{22}) - \sqrt{(a_{11} - a_{22})^2 + 4a_{12}^2} \right] \quad (3.57) \]

Furthermore, equations (3.56) and (3.57) yield

\[ [(a_{11} + a_{22}) + 4\mathcal{A}]^2 = (a_{11} - a_{22})^2 + 4a_{12}^2 = [(a_{11} - a_{22}) + 4\mathcal{B}]^2 \quad (3.58) \]

Let \( T \) denote the smaller absolute value of \( \mathcal{A} \) and \( \mathcal{B} \). Therefore, equation (3.57) can be written as

\[ [(a_{11} + a_{22}) + 4T]^2 = (a_{11} - a_{22})^2 + 4a_{12}^2 \quad (3.59) \]
STEP 2: Representation of coefficients $a_{11}$, $a_{12}$, and $a_{22}$ in terms of $rV_2^{(1)}$ and $rV_{2II}^{(1)}$

Using the first two equations in (A.29) and equation (3.54), we may derive a system of three linear equations in unknowns $a_{11}$, $a_{12}$, and $a_{22}$

\[
\begin{align*}
    rV_2^{(1)} a_{11} &+ rV_{2II}^{(1)} a_{12} = a_{13} \\
    rV_2^{(1)} a_{12} &+ rV_{2II}^{(1)} a_{22} = a_{23} \\
    a_{11} &+ a_{22} = \kappa_\lambda
\end{align*}
\]

where $\kappa_\lambda = \kappa_{II} - \kappa_{I}$. Using Cramer's rule we may solve equations (3.60) as follows:

\[
\begin{align*}
    a_{11} &= \frac{a_{13} rV_2^{(1)} - a_{23} rV_{2II}^{(1)} + \kappa_\lambda (rV_{2II}^{(1)})^2}{(rV_2^{(1)})^2 + (rV_{2II}^{(1)})^2} \\
    a_{12} &= \frac{a_{13} rV_2^{(1)} + a_{23} rV_{2II}^{(1)} - \kappa_\lambda rV_2^{(1)} rV_{2II}^{(1)}}{(rV_2^{(1)})^2 + (rV_{2II}^{(1)})^2} \\
    a_{22} &= \frac{-a_{13} rV_2^{(1)} + a_{23} rV_{2II}^{(1)} + \kappa_\lambda (rV_{2II}^{(1)})^2}{(rV_2^{(1)})^2 + (rV_{2II}^{(1)})^2}
\end{align*}
\]

The third equation in (A.29) is

\[
a_{31} rV_2^{(1)} + a_{32} rV_{2II}^{(1)} = a_{33}
\]

Substituting equation (3.49) into equation (3.64), we have

\[
rV_2^{(1)} = \frac{a_{33}}{a_{13} + a_{23} \tan r_1}
\]
Plugging equations (3.49) and (3.65) into equations (3.61)-(3.63), we obtain the following results

\[ a_{11} = d_1 \kappa_A + b_1 \]  

(3.66)

\[ a_{12} = d_2 \kappa_A + b_2 \]  

(3.67)

\[ a_{13} = d_3 \kappa_A + b_3 \]  

(3.68)

where

\[ d_1 = \frac{\tan^2 \nu_1}{1 + \tan^2 \nu_1} \]  

(3.69)

\[ d_2 = \frac{-\tan \nu_1}{1 + \tan^2 \nu_1} \]  

(3.70)

\[ d_3 = \frac{1}{1 + \tan^2 \nu_1} \]  

(3.71)

\[ b_1 = \frac{a_{13} - a_{23} \tan^2 \nu_1}{a_{33}(1 + \tan^2 \nu_1)} \]  

(3.72)

\[ b_2 = \frac{(a_{33} + a_{13} \tan \nu_1)(a_{13} + a_{23} \tan \nu_1)}{a_{33}(1 + \tan^2 \nu_1)} \]  

(3.73)
STEP 3: Determination of $\kappa_\Delta$

Equations (3.59) and (3.66)–(3.71) lead to

$$\kappa_\Delta = -\frac{\left[4T^2 - (b_1^2 + b_2^2)\right] (1 + \tan^2 \nu_1)}{2T(1 + \tan^2 \nu_1) + b_1(1 - \tan^2 \nu_1) + 2b_2 \tan \nu_1}$$  \hspace{1cm} (3.74)

Since $\kappa_\Delta = \kappa_{2\Sigma} - \kappa_{1\Sigma}$, equation (3.74) becomes

$$\kappa_{1\Sigma} = \kappa_{2\Sigma} + \frac{\left[4T^2 - (b_1^2 + b_2^2)\right] (1 + \tan^2 \nu_1)}{2T(1 + \tan^2 \nu_1) + b_1(1 - \tan^2 \nu_1) + 2b_2 \tan \nu_1}$$  \hspace{1cm} (3.75)

Note that

$$\mathcal{T} = \frac{\varepsilon}{t^2}$$  \hspace{1cm} (3.76)

where $t$ is the semimajor axis of the contact ellipse. This is an input datum. In general, it is about one sixth of the width of the gear tooth. Gleason Works suggests that the elastic approach $\varepsilon$ is 0.00025 inches [11].

STEP 4: Determination of $a_{11}$, $a_{12}$, and $a_{22}$

Substituting equation (3.74) into equations (3.66)–(3.68), we obtain $a_{11}$, $a_{12}$, and $a_{22}$.

STEP 5: Determination of $\sigma_{12}$

Using equations (A.32) and (3.55), we obtain

$$\tan 2\sigma_{12} = \frac{2a_{12}}{\kappa_{1\Delta} - a_{11} + a_{22}}$$  \hspace{1cm} (3.77)
It provides two solutions for $\sigma_{12}$, and we will choose the smaller value. Rotating unit vector $\vec{e}_{2j}$ about the unit normal vector $\vec{n}$ by $-\sigma_{12}$, we may obtain unit vector $\vec{e}_{1j}$. Rotating unit vector $\vec{e}_{1j}$ about the unit normal vector $\vec{n}$ by $\pi/2$, we may obtain unit vector $\vec{e}_{2H}$.

**STEP 6: Determination of $\kappa_{1\Delta}$**

Using equation (A.32), we obtain

$$\kappa_{1\Delta} = \frac{2\sigma_{12}}{\sin 2\sigma_{12}}$$

**STEP 7: Determination of $\kappa_{1I}$ and $\kappa_{1II}$**

The principal curvatures of the pinion surface at the mean contact point $B$ are determined by

$$\kappa_{1I} = \frac{\kappa_{1E} + \kappa_{1\Delta}}{2}, \quad \kappa_{1II} = \frac{\kappa_{1E} + \kappa_{1\Delta}}{2}$$

3.3.4 First Order Characteristics

Four surfaces, the gear head-cutter surface $\Sigma_G$, the gear surface $\Sigma_2$, the pinion head-cutter surface $\Sigma_P$, and the pinion surface $\Sigma_1$, are in tangency simultaneously at the mean contact point $B$. It implies that these four surfaces have a common normal at the mean contact point. We can use this information to determine pinion blade angle $\psi_p$ and parameter $\tau_p$.

The representation of the unit normal to the pinion head-cutter surface in the $S_f$ coordinate system is
\[
\begin{align*}
[n_f] &= [L_{f,f}][L_{f,m}][L_{m,m}][L_{m,c}][n_c] \\
&= \begin{bmatrix}
-1 & 0 & 0 \\
0 & -1 & 0 \\
0 & 0 & 1 \\
\end{bmatrix}
\begin{bmatrix}
\cos \delta & 0 & -\sin \delta \\
0 & 1 & 0 \\
\sin \delta & 0 & \cos \delta \\
\end{bmatrix}
\begin{bmatrix}
1 & 0 & 0 \\
0 & \cos \phi & \pm \sin \phi \\
0 & \mp \sin \phi & \cos \phi \\
\end{bmatrix}
\begin{bmatrix}
1 & 0 & 0 \\
0 & \cos \phi_p & \mp \sin \phi_p \\
0 & \mp \sin \phi_p & \cos \phi_p \\
\end{bmatrix}
\begin{bmatrix}
\frac{n_{c_x}}{r} \\
\frac{n_{c_y}}{r} \\
\frac{n_{c_z}}{r} \\
\end{bmatrix}
\end{align*}
\]

(3.80)

Let us consider the straight-edged blade first. Equation (2.2) describes the unit normal in the \( S_z \) coordinate system. Before plugging equation (2.2) into equation (3.80), we must investigate the sense of equation (2.2). From Figure 18 we know we must choose the minus sign for the unit normal. Therefore, equations (2.2) and (3.80) yield (subscript ‘f’ is dropped)

\[
\begin{align*}
\begin{bmatrix}
\cos \delta \sin \psi - \sin \delta \cos \psi \cos \tau \\
\cos \psi \sin \tau \\
-\sin \delta \sin \psi - \cos \delta \cos \psi \cos \tau \\
\end{bmatrix}
\begin{bmatrix}
\cos \delta & 0 & -\sin \delta \\
0 & 1 & 0 \\
\sin \delta & 0 & \cos \delta \\
\end{bmatrix}
\begin{bmatrix}
1 & 0 & 0 \\
0 & \cos \phi & \pm \sin \phi \\
0 & \mp \sin \phi & \cos \phi \\
\end{bmatrix}
\begin{bmatrix}
\frac{n_{c_x}}{r} \\
\frac{n_{c_y}}{r} \\
\frac{n_{c_z}}{r} \\
\end{bmatrix}
\end{align*}
\]

(3.81)

Multiplying \( n_x \) by \( \cos \delta_1 \), \( n_z \) by \( -\sin \delta_1 \), and then considering their sum, we obtain

\[
n_x \cos \delta_1 - n_z \sin \delta_1 = \sin \psi_p
\]

(3.82)

Obviously, the pinion blade angle is

\[
\psi_p = \begin{cases}
\arcsin(n_x \cos \delta_1 - n_z \sin \delta_1) & \text{Gear Concave Side} \\
(\pi - \psi_p) & \text{Gear Convex Side}
\end{cases}
\]

(3.83)
The $x$ component in equation (3.81) may be rewritten as

$$
\cos \tau_p = \frac{n_x - \cos \delta \sin \psi_p}{-\sin \delta \cos \psi_p}
$$

(3.84)

The $y$ component in equation (3.81) may be rewritten as

$$
\sin \tau_p = \frac{n_y}{\cos \psi_p}
$$

(3.85)

The parameter $\tau_p$ may be obtained by

$$
\tau_p = 2 \arctan \frac{\sin \tau_p}{1 + \cos \tau_p}
$$

(3.86)

Let us now consider the curve-edged blade. Equations (2.16) and (3.80) yield

$$
\begin{bmatrix}
  n_x \\
  n_y \\
  n_z
\end{bmatrix} =
\begin{bmatrix}
  \cos \delta \cos \lambda_p - \sin \delta \sin \lambda_p \cos \tau_p \\
  \sin \lambda_p \sin \tau_p \\
  -\sin \delta \cos \lambda_p - \cos \delta \sin \lambda_p \cos \tau_p
\end{bmatrix}
$$

(3.87)

Multiplying $n_x$ by $\cos \delta$, $n_z$ by $-\sin \delta$, and then considering their sum, we obtain

$$
\cos \lambda_p = n_x \cos \delta - n_z \sin \delta
$$

(3.88)
The quadrant in which the parameter $\lambda_p$ locates may be determined by the discussion stated in Section 2.2.

The blade angle is the angle formed by a line tangent to the blade surface at the mean contact point and a line perpendicular to the cutter head face. Thus we have

$$\psi_p = \begin{cases} \frac{5}{2}\pi - \lambda_p & \text{pinion concave side, blade concave down;} \\ \frac{3}{2}\pi - \lambda_p & \text{pinion concave side, blade concave up;} \\ \frac{1}{2}\pi - \lambda_p & \text{pinion convex side, blade concave down;} \\ \frac{3}{2}\pi - \lambda_p & \text{pinion convex side, blade concave up.} \end{cases}$$

Rewriting the $x$ component in equation (3.87), we have

$$\cos \tau_p = \frac{n_x - \cos \delta_1 \cos \lambda_p}{- \sin \delta_1 \sin \lambda_p} \quad (3.89)$$

The $y$ component in equation (3.87) may be rewritten as

$$\sin \tau_p = \frac{n_y}{\sin \lambda_p} \quad (3.90)$$

Substituting equations (3.89) and (3.90) into equations (3.86), we may obtain $\tau_p$.

3.3.5 Principal Curvatures and Directions of the Pinion Cutter Surface at the Mean Contact Point

The first principal direction of the pinion cutter surface at the mean contact point may be represented in the $S_p$ coordinate system as follows:
Using equation (2.9) and (3.91), we may obtain the first principal direction for the straight-edged cutter. It is

\[
\left[ \varepsilon_{P_f} \right] = \left[ L_{f_p(p)} \right] \left[ L_{pl(p_m(p))} \right] \left[ L_{m(p)\tilde{n}_p} \right] \left[ \varepsilon_{P_f} \right]
\]

(3.91)

Using equations (2.17) and (3.91), we may obtain the first principal direction for the curve-edged cutter. The result is the same as for the straight-edged cutter, that is described in equation (3.92).

In above equation, there are two senses. Only the direction which forms the smaller angle with the gear cutter first principal direction can be chosen. From the first order information we have already determined the parameter \( \tau_p \); therefore, the first principal direction of the pinion cutter is also determined. The unit vector of the second principal direction of the pinion cutter surface may be obtained by rotating the unit vector of the first principal direction of the pinion cutter surface, \( \varepsilon_{P_f} \), about the common normal, \( \tilde{n} \), by an angle \( \pi/2 \).

We use the concept discussed in Section A.2 to derive the principal curvatures of the pinion cutter surface at the mean contact point. We recall that surfaces \( \Sigma_P \) and \( \Sigma_1 \) are in line contact in the process of generation. Hence, using equation (A.37), we obtain

\[
a_{11} a_{22} - a_{12}^2 = 0
\]

(3.93)

Substituting equations (A.31), (A.32), and (A.34) into (3.93), we obtain the first principal curvature of the pinion cutter.
\[ \kappa_{PI} = \frac{\kappa_{PI} (\kappa_{III} \cos^2 \sigma_{P1} + \kappa_{II} \sin^2 \sigma_{P1}) - \kappa_{II} \kappa_{III}}{\kappa_{PI} - \kappa_{II} \sin^2 \sigma_{P1} - \kappa_{III} \cos^2 \sigma_{P1}} \]  

(3.94)

The second curvature of the pinion cutter is zero for a straight-edged cutter (see equation (2.12)) and \( \mp 1/R \) for a curve-edged cutter (see equation (2.20)). Since the principal curvatures and directions of the pinion cutter surface at the mean contact point have been determined, some data relating to pinion machine-tool settings may be obtained without any difficulty.

Let us consider a straight-edged cutter first. Rewriting equation (2.10), we may obtain

\[ u_p = \frac{1}{\kappa_{PI} \tan \psi} \]  

(3.95)

We choose only the positive sign in equation (2.10) since we have specified the direction of the unit normal \( \vec{n} \). We may represent the mean contact point \( B \) in the \( S_{m(p)} \) coordinate system as follows:

\[ [B_{m(p)}] = [M_{m(p)p(p)}] [M_{p(p)f}] [B_f] \]  

(3.96)

where

\[ [M_{m(p)p(p)}] = \begin{bmatrix} \cos \delta_i & 0 & \sin \delta_i & 0 \\ 0 & 1 & 0 & \mp E_m \\ -\sin \delta_i & 0 & \cos \delta_i & -L_m \\ 0 & 0 & 0 & 1 \end{bmatrix} \]  

(3.97)

and
\[
\begin{bmatrix}
-1 & 0 & 0 & 0 \\
0 & -1 & 0 & 0 \\
0 & 0 & 1 & 0 \\
0 & 0 & 0 & 1
\end{bmatrix} = M_{p(P)f}
\] (3.98)

Considering only the \( x \) component of the above equation, we obtain

\[ B_{mx} = -B_{f_x} \cos \delta + B_{f_x} \sin \delta \] (3.99)

Using equation (2.38), we obtain \( r_p \) as follows:

\[ r_p = (B_{m_x} + u_p \cos \psi_p) \tan \psi_p \] (3.100)

Let us now consider the curve-edged cutter. Using equation (2.18), we obtain

\[ R_{c_s} = \frac{-\sin \lambda_p}{\kappa_p} - R \sin \lambda_p \] (3.101)

The parameter \( R_{c_s} \) may be obtained by equations (2.40) and (3.99). That is

\[ R_{c_s} = B_{m_x} - R \cos \lambda_p \] (3.102)

The cutter tip radius may be represented by

\[ r_p = R_{c_s} \pm \sqrt{|R^2 - R_{c_s}^2|} \] (3.103)
3.4 Pinion Machine-Tool Settings

There are five machine-tool settings $m_{p_1}$, $E_m$, $L_m$, $s_p$, and $q_p$ to be determined. The key to the solution of this problem is the determination of the cutting ratio $m_{p_1}$. Let us consider this problem first.

3.4.1 Determination of Pinion Cutting Ratio

We will use the relations between principal curvatures and directions for the pinion cutter surface and the pinion surface to derive the pinion cutting ratio $m_{p_1}$. To apply the equations described in Section A.2, we consider that surfaces $\Sigma_1$ and $\Sigma_2$ are equivalent, and that surfaces $\Sigma_P$ and $\Sigma_Q$ are equivalent. Also, the following data are considered as given: (1) the principal curvatures of the pinion surface at the mean contact point, $\kappa_{11}$ and $\kappa_{12}$; (2) the principal directions of the pinion surface at the mean contact point, $\vec{e}_{11}$ and $\vec{e}_{12}$; (3) the coordinates of the mean contact point; (4) the unit normal at the mean contact point; (5) the coefficients $a_{11}$, $a_{12}$, and $a_{22}$.

The procedure to determine $m_{p_1}$ is as follows:

Step 1: Representation of $\omega^{(1p)}$

The angular velocity of the pinion is represented by

\[
\omega^{(1)} = \pm \omega^{(1)} \begin{bmatrix} \sin \mu_1 \\ 0 \\ \cos \mu_1 \end{bmatrix}
\]  \hspace{1cm} (3.104)

The angular velocity of the pinion cutter is represented by
\[
\omega^{(1)} = \pm m_{p_1} \omega^{(1)}
\]

\[
\begin{bmatrix}
\begin{array}{c}
\cos \delta_1 \\
0 \\
-\sin \delta_1 \\
\end{array}
\end{bmatrix}
\]

(3.105)

Therefore, we may obtain the relative angular velocity \( \omega^{(1)} \) as follows:

\[
\dot{\omega}^{(1)} = \pm \omega^{(1)}
\]

\[
\begin{bmatrix}
\begin{array}{c}
\sin \mu_1 - m_{p_1} \cos \delta_1 \\
0 \\
\cos \mu_1 + m_{p_1} \sin \delta_1 \\
\end{array}
\end{bmatrix}
\]

(3.106)

STEP 2: Representation of \( [\omega^{(1)} \vec{n} \vec{e}_{p_1}] \)

The scalar \( [\omega^{(1)} \vec{n} \vec{e}_{p_1}] \) is represented by

\[
\begin{bmatrix}
\begin{array}{c}
\pm (\sin \mu_1 - m_{p_1} \cos \delta_1) \\
0 \\
\pm (\cos \mu_1 + m_{p_1} \sin \delta_1) \\
\end{array}
\end{bmatrix}
\begin{bmatrix}
\begin{array}{c}
\vec{n}_x \\
\vec{n}_y \\
\vec{n}_z \\
\end{array}
\end{bmatrix}
\begin{bmatrix}
\begin{array}{c}
\vec{e}_{p_1x} \\
\vec{e}_{p_1y} \\
\vec{e}_{p_1z} \\
\end{array}
\end{bmatrix}
\]

(3.107)

\[
= \pm \left\{ \left( n_x \vec{e}_{p_1x} - n_y \vec{e}_{p_1y} \cos \delta_1 \right) \sin \delta_1 + \left( n_z \vec{e}_{p_1z} - n_y \vec{e}_{p_1y} \sin \delta_1 \right) \cos \mu_1 \right\} m_{p_1}
\]

+ \left( n_y \vec{e}_{p_1y} - n_z \vec{e}_{p_1z} \right) \sin \mu_1 + \left( n_x \vec{e}_{p_1x} - n_z \vec{e}_{p_1z} \right) \cos \mu_1 \right\} \omega^{(1)}
\]

\[
= (c_{11} m_{p_1} + c_{12}) \omega^{(1)}
\]

STEP 3: Representation of \( [\omega^{(1)} \vec{n} \vec{e}_{p_{11}}] \)

The scalar \( [\omega^{(1)} \vec{n} \vec{e}_{p_{11}}] \) is represented by
\[
\begin{align*}
\mathbf{\omega}^{(1)} &= \begin{bmatrix} (\sin \mu_1 - m_{p_1} \cos \delta_1) & 0 & (\cos \mu_1 + m_{p_1} \sin \delta_1) \\ n_x & n_y & n_z \\ \epsilon_{P_{II_x}} & \epsilon_{P_{II_y}} & \epsilon_{P_{II_z}} \end{bmatrix} \\
&= \pm \left( n_y \epsilon_{P_{II_y}} - n_z \epsilon_{P_{II_z}} \right) \sin \mu_1 + \left( n_x \epsilon_{P_{II_z}} - n_y \epsilon_{P_{II_x}} \right) \cos \mu_1 \mathbf{\omega}^{(1)} \\
&= c_{22} \mathbf{\omega}^{(1)}
\end{align*}
\]

**STEP 4: Representation of \( \mathbf{\tilde{V}}^{(1)} \)**

The velocity \( \mathbf{t}_r \mathbf{\tilde{V}}^{(1)} \) may be obtained by

\[
\mathbf{t}_r \mathbf{\tilde{V}}^{(1)} = \mathbf{\omega}^{(1)} \times \vec{B}
\]

\[
= \pm \mathbf{\omega}^{(1)} \begin{bmatrix} -B_y \cos \mu_1 \\ B_x \cos \mu_1 - B_z \sin \mu_1 \\ B_y \sin \mu_1 \end{bmatrix}
\]

The velocity \( \mathbf{t}_r \mathbf{\tilde{V}}^{(1)} \) may be obtained by

\[
\begin{align*}
\mathbf{t}_r \mathbf{\tilde{V}}^{(1)} &= \mathbf{\omega}^{(P)} \times \vec{B} + \dot{\mathbf{O}}_{fOm} \times \mathbf{\dot{\omega}}^{(P)} \\
&= \mathbf{\omega}^{(3)} m_{p_1} \begin{bmatrix} (E_m \pm B_y) \sin \delta_1 \\ \pm (L_m - B_x \sin \delta_1 - B_y \cos \delta_1) \\ (E_m \pm B_y) \cos \delta_1 \end{bmatrix}
\end{align*}
\]

The sliding velocity \( \mathbf{\tilde{V}}^{(1)} \) is described by
\[ V^{(1P)}_I = \tau V^{(1)} - \tau V^{(P)} \]

\[ = \omega^{(1)} \left[ \begin{array}{l} \pm B_y \cos \mu_1 - m_{p_1} (E_m \pm B_y \sin \delta_1) \\ \pm (B_x \cos \mu_1 - B_z \sin \mu_1) \mp m_{p_1} [L_m - (B_x \sin \delta_1 + B_z \cos \delta_1)] \\ \pm B_y \sin \mu_1 - m_{p_1} (E_m \pm B_y \cos \delta_1) \end{array} \right] \] (3.111)

**STEP 5: Representation of \( V^{(1P)}_{P_I} \) and \( V^{(1P)}_{P_{II}} \)**

Using equations (A.33) and (3.107), we have

\[ a_{13} = -\kappa_{P_I} V^{(1P)}_{P_I} - (c_{11} m_{p_1} + c_{12}) \omega^{(1)} \] (3.112)

Equations (A.35) and (3.108) yield

\[ a_{23} = -\kappa_{P_{II}} V^{(1P)}_{P_{II}} - c_{22} \omega^{(1)} \] (3.113)

From equation (A.37) we have

\[ a_{11} a_{23} - a_{13} a_{23} = 0 \] (3.114)

Using equations (3.112) - (3.114), we obtain

\[ a_{12} \kappa_{P_I} V^{(1P)}_{P_I} - a_{11} \kappa_{P_{II}} V^{(1P)}_{P_{II}} = [-a_{14} c_{11} m_{p_1} + (a_{11} c_{22} - a_{13} c_{12})] \omega^{(1)} \] (3.115)

Moreover, we know that
Considering only the \( z \) component in equations (3.111) and (3.116), we obtain

\[
V_j^{(1P)} = V_{P_i}^{(1P)} \varepsilon_{P_i} + V_{P_{II}}^{(1P)} \varepsilon_{P_{II}}
\]  
(3.116)

Considering only the \( z \) component in equations (3.111) and (3.116), we receive

\[
V_{P_i}^{(1P)} e_{P_{Iz}} + V_{P_{II}}^{(1P)} e_{P_{IIz}} = \left[ \mp B_y \cos \mu, - m_{P_1} (E_m \pm B_y) \sin \delta_1 \right] \omega^{(1)}
\]  
(3.117)

Considering only the \( z \) component in equations (3.111) and (3.116), we receive

\[
V_{P_i}^{(1P)} e_{P_{Iz}} + V_{P_{II}}^{(1P)} e_{P_{IIz}} = \left[ \pm B_y \sin \mu, - m_{P_1} (E_m \pm B_y) \cos \delta_1 \right] \omega^{(1)}
\]  
(3.118)

Multiplying equation (3.117) by \( \cos \delta_1 \) and equation (3.118) by \( \sin \delta_1 \), and adding the resulting equations, we obtain

\[
V_{P_{II}}^{(1P)} = \mp \frac{B_y \cos \gamma_1}{\varepsilon_{P_{IIz}} \cos \delta_1 - \varepsilon_{P_{IIz}} \sin \delta_1} \omega^{(1)} = t_1 \omega^{(1)}
\]  
(3.119)

Substituting equation (3.119) into equation (3.117), we obtain

\[
V_{P_i}^{(1P)} = \left( - \frac{c_{11}}{\kappa_{P_i}} m_{P_1} + \frac{a_{11} \kappa_{P_{II}} t_4 + a_{11} c_{22} - a_{12} c_{12}}{a_{12} \kappa_{P_i}} \right) \omega^{(1)} = (t_1 m_{P_1} + t_2) \omega^{(1)}
\]  
(3.120)

**STEP 6: Representation of \( V_j^{(1P)} \)**

The matrix form of equation (3.116) may be represented by

74
\[ V^{(1P)} = \begin{bmatrix} V_{P1}^{(1P)} e_{Pr}^{(1P)} + V_{H}^{(1P)} e_{P_{IIz}} \\ V_{P1}^{(1P)} e_{Pr}^{(1P)} + V_{H}^{(1P)} e_{P_{IIy}} \\ V_{P1}^{(1P)} e_{Pr}^{(1P)} + V_{H}^{(1P)} e_{P_{IIz}} \end{bmatrix} \]

Substituting equations (3.119) and (3.120) into equation (3.121), we have

\[ \begin{align*}
V^{(1P)} &= \omega^{(1)} \\
&= \begin{bmatrix} t_{1} m_{p1} + t_{2} e_{Pr} + t_{4} e_{P_{IIz}} \\ t_{1} m_{p1} + t_{2} e_{Pr} + t_{4} e_{P_{IIy}} \\ t_{1} m_{p1} + t_{2} e_{Pr} + t_{4} e_{P_{IIz}} \end{bmatrix} \\
&= \omega^{(1)} \\
&= \begin{bmatrix} u_{11} m_{p1} + u_{12} \\ u_{21} m_{p1} + u_{22} \\ u_{31} m_{p1} + u_{32} \end{bmatrix} \end{align*} \]

STEP 7: Representation of \( \tilde{n} \omega^{(1P)} V^{(1P)} \)

The scalar \( \tilde{n} \omega^{(1P)} V^{(1P)} \) may be represented by

\[ \begin{bmatrix} \tilde{n} \omega^{(1P)} V^{(1P)} \end{bmatrix} = \begin{bmatrix} \omega^{(1)} \end{bmatrix}^{2} \begin{bmatrix} n_{x} \\ n_{y} \\ n_{z} \end{bmatrix} = \begin{bmatrix} \omega^{(1)} \end{bmatrix}^{2} \begin{bmatrix} \pm (\sin \mu_{1} + m_{p1} \cos \delta_{1}) \\ u_{11} m_{p1} + u_{12} \\ u_{21} m_{p1} + u_{22} \\ u_{31} m_{p1} + u_{32} \end{bmatrix} \]

\[ = \left( v_{1} m_{p1}^{2} + v_{2} m_{p1} + v_{3} \right) \begin{bmatrix} \omega^{(1)} \end{bmatrix}^{2} \]

where

\[ v_{1} = \pm [(u_{11} \sin \delta_{1} - u_{21} \cos \delta_{1}) n_{y} - (n_{x} \cos \delta_{1} + n_{z} \sin \delta_{1}) u_{21}] \]

75
\[ v_2 = \mp [(u_{21} \cos \mu_1 + u_{22} \sin \delta_1)n_x - (u_{21} \sin \mu_1 - u_{22} \cos \delta_1)n_z \]
- \[(u_{11} \cos \mu_1 + u_{12} \sin \delta_1 + u_{32} \cos \delta_1 - u_{11} \sin \mu_1)n_y] \]

(3.125)

\[ v_3 = \pm [(u_{22} n_x \cos \mu_1 - (u_{12} \cos \mu_1 - u_{32} \sin \mu_1)n_y - u_{22} n_z \sin \mu_1] \]

(3.126)

**STEP 8: Representation of**

\[ \tilde{\mathbf{V}}(1) \times \mathbf{t_XV}(P) - \tilde{\mathbf{V}}(P) \times \mathbf{t_XV}(1) \]

**The velocity** \( \mathbf{t_XV}(P) \) **may be described by**

\[ \mathbf{t_XV}(P) = \mathbf{t_XV}(1) - \tilde{\mathbf{V}}(1) \]

(3.127)

Substituting equations (3.109) and (3.122) into equation (3.127), we have

\[ \mathbf{t_XV}(P) = \omega^{(1)} \left[ \begin{array}{c}
-u_{11} m_{P_1} \mp B_y \cos \mu_1 - u_{12} \\
-u_{21} m_{P_1} \mp B_z \sin \mu_1 \pm B_x \cos \mu_1 - u_{22} \\
-u_{31} m_{P_1} \pm B_y \sin \mu_1 - u_{32}
\end{array} \right] \]

(3.128)

**Vector** \( \omega^{(1)} \times \mathbf{t_XV}(P) \) **is represented by**

\[ \omega^{(1)} \times \mathbf{t_XV}(P) = \left[ \omega^{(1)} \right]^2 \left[ \begin{array}{c}
\pm [u_{21} m_{P_1} - (B_z \sin \mu_1 - B_x \cos \mu_1 \pm u_{22})] \cos \mu_1 \\
(\mp u_{11} \cos \mu_1 \pm u_{11} \sin \mu_1) m_{P_1} - B_y \mp u_{12} \cos \mu_1 \pm u_{32} \sin \mu_1 \\
\mp [u_{22} m_{P_1} - (B_x \sin \mu_1 - B_z \cos \mu_1 \pm u_{22})] \sin \mu_1
\end{array} \right] \]

(3.129)
Vector \( \omega^{(l)} \times r \vec{v}^{(1)} \) is represented by

\[
\omega^{(l)} \times r \vec{v}^{(1)} = \left[ \omega^{(1)} \right]^2 \begin{bmatrix}
-(B_z \sin \mu_1 - B_x \cos \mu_1)m_{p_1} \sin \delta_1 \\
-B y m_{p_1} \sin \gamma_1 \\
-(B_z \sin \mu_1 - B_x \cos \mu_1)m_{p_1} \cos \delta_1
\end{bmatrix}
\]  

(3.130)

Subtracting equation (3.130) from equation (3.129), we obtain

\[
\omega^{(l)} \times r \vec{v}^{(1)} - \omega^{(l)} \times r \vec{v}^{(1)} = \left[ \omega^{(1)} \right]^2 \begin{bmatrix}
h_{11}m_{p_1} + h_{12} \\
h_{21}m_{p_1} + h_{22} \\
h_{31}m_{p_1} + h_{32}
\end{bmatrix}
\]  

(3.131)

where

\[
h_{11} = \pm u_{21} \cos \mu_1 - (B_x \cos \mu_1 - B_z \sin \mu_1) \sin \delta_1
\]  

(3.132)

\[
h_{12} = (B_z \sin \mu_1 - B_x \cos \mu_1 \pm u_{22}) \cos \mu_1
\]  

(3.133)

\[
h_{21} = B_y \sin \gamma_1 \mp u_{11} \cos \mu_1 \pm u_{31} \sin \mu_1
\]  

(3.134)

\[
h_{22} = -(B_y \pm u_{12} \cos \mu_1 \mp u_{32} \sin \mu_1)
\]  

(3.135)

\[
h_{31} = \mp u_{21} \sin \mu_1 - (B_x \cos \mu_1 - B_z \sin \mu_1) \cos \delta_1
\]  

(3.136)

\[
h_{32} = -(B_z \sin \mu_1 - B_x \cos \mu_1 \pm u_{22}) \sin \mu_1
\]  

(3.137)
Therefore, we may obtain \( \vec{n} \cdot (\vec{\omega}^{(1)} \times \text{tr} \vec{V}^{(p)} - \vec{\omega}^{(p)} \times \text{tr} \vec{V}^{(1)}) \) as follows:

\[
\vec{n} \cdot (\vec{\omega}^{(1)} \times \text{tr} \vec{V}^{(p)} - \vec{\omega}^{(p)} \times \text{tr} \vec{V}^{(1)}) = (f_1 m_p + f_2) [\omega^{(1)}]^2
\]  

(3.138)

where

\[
f_1 = n_x h_{11} + n_y h_{21} + n_z h_{31}
\]  

(3.139)

\[
f_2 = n_x h_{12} + n_y h_{22} + n_z h_{32}
\]  

(3.140)

**STEP 9: Representation of \( m_p \)**

Using equations (A.33), (3.107), and (3.120), the equation for \( a_{13} \) may be represented by

\[
a_{13} = - \left( \kappa_{p_l} t_2 + c_{12} \right) \omega^{(1)}
\]  

(3.141)

Using equations (A.35), (3.108), and (3.119), \( a_{23} \) may be described by

\[
a_{23} = - \left( \kappa_{p_l} t_4 + c_{22} \right) \omega^{(1)}
\]  

(3.142)

Using equations (A.36), (3.119), (3.120), (3.123), and (3.138), the expression for \( a_{33} \) may be represented by
From equation (A.39) we know that

$$a_{12}a_{33} - a_{13}a_{23} = 0$$  \hspace{1cm} (3.144)

Equations (3.141)-(3.144) yield

$$m_{p_1} = \frac{a_{12} (\kappa_{P_1} t_2^2 + \kappa_{P_2} t_4^2 - v_3 - f_3) - (\kappa_{P_1} t_2 + c_{12})(\kappa_{P_2} t_4 + c_{22})}{a_{12} (2\kappa_{P_1} t_1 t_2 - v_2 - f_1)}$$  \hspace{1cm} (3.145)

3.4.2 Determination of parameters $E_m$ and $L_m$

Parameters $E_m$ and $L_m$ of the pinion machine-tool settings have been shown in Figures 12 and 13. Since the pinion cutting ratio $m_{p_1}$ has been determined, it is very easy to find these two parameters. We may determine vector $\vec{V}^{(1p)}$ from equation (3.122). Applying equation (3.111), then, we obtain

$$E_m = \frac{\mp B_y \cos \mu_1 - V_x^{(1p)}}{m_{p_1} \sin \delta_1} \mp B_y$$  \hspace{1cm} (3.146)

$$L_m = \frac{B_y \cos \mu_1 - B_z \sin \mu_1 \mp V_y^{(1p)}}{m_{p_1}} + B_z \sin \delta_1 + B_z \cos \delta_1$$  \hspace{1cm} (3.147)
3.4.3 Determination of Pinion Radial Setting and Cradle Angle

The determination of the pinion radial setting and the cradle angle is based on the consideration that the position vectors of the pinion tooth surface and head-cutter surface must coincide at the mean contact point. Equation (3.96) describes the mean contact point $B$ in the $S_{m(p)}$ coordinate system. Considering the $y$ and $z$ components in equation (3.96), we obtain

$$B_{m(y)} = -B_{f_y} + E_m$$

$$B_{m(z)} = B_{f_z} \sin \delta + B_{f_z} \cos \delta - L_m$$

For a straight-edged cutter, by using equations (2.38), (3.148), and (3.149), we have

$$s_p \sin q_p = \pm B_{f_y} + E_m \pm u_p \sin \psi_p \sin \tau_p$$

$$s_p \cos q_p = B_{f_z} \sin \delta + B_{f_z} \cos \delta - L_m - u_p \sin \psi_p \cos \tau_p$$

For a curved-edged cutter, by using equations (2.40), (3.148), and (3.149), we have

$$s_p \sin q_p = \pm B_{f_y} + E_m \pm \frac{\cos \lambda_p \sin \tau_p}{\kappa_p}$$

$$s_p \cos q_p = B_{f_z} \sin \delta + B_{f_z} \cos \delta - L_m \pm \frac{\cos \lambda_p \cos \tau_p}{\kappa_p}$$
Using $\sin^2 q_p + \cos^2 q_p = 1$, we eliminate $q_p$ and solve for pinion radius $s_p$. Eliminating $s_p$, we may determine the pinion cradle angle $q_p$. 
CHAPTER 4

CONCLUSION

As it was mentioned in Chapter 1, the reduction of transmission errors of spiral bevel gears is a difficult problem. Although it is possible to generate conjugate spiral bevel gears, with zero transmission errors, we have to take into account that the gear are very sensitive to misalignment. Using the TCA programs we have found that even a small misalignment of gears results in discontinuity of functions of transmission errors that is accompanied with the jump of the function at the transfer points. Thus the idea of gears with non-zero transmission has to be complemented with the modification of the process for their generation that allows to reduce the sensitivity of gears to their misalignment.

From the result of computation by TCA programs we know that gear misalignment causes a linear or almost linear function of transmission errors. Litvin has discovered that a sum of a parabolic function and a linear function represents again a parabolic function that is just translated with respect to the initial parabolic function. Then, if a parabolic function is predesigned, it becomes possible to keep the same level of transmission errors for aligned as well as misaligned gears.

Gear misalignment is also accompanied with the shift of the bearing contact to the edge of gear tooth surface. To keep the shift of bearing contact in reasonable limits, it is necessary to limit the tolerances for gear misalignment and the respective value of predesigned parabolic function.

In Chapter 2 the basic concept and methods of Gleason systems have been presented. Equations that describe the surface of the head cutter, which is either a cone surface or a surface of revolution,
have been derived. These equations cover the determination of position vectors, surface unit normal vectors, principal curvatures, and principal directions.

Mathematical models for geometry of spiral bevel gears have been also proposed in Chapter 2. The gear surface is represented as an envelope of the family of the tool surfaces. The tool surface and being generated gear surface are considered conjugate ones. Based on the geometric properties of conjugate surfaces, the equation of meshing has been derived.

The determination of pinion machine-tool settings is based on the method of local synthesis. The first derivative of gear ratio, the tangent to the contact path, and the dimensions of the contact ellipse of the gear surface at the mean contact point are considered as input to local synthesis. Thus the level of transmission errors and the bearing contact are under control. It provides the optimal conditions of meshing for the gear surfaces being in meshing at, and within the neighborhood of, the mean contact point.

Equations that determine the principal curvatures and directions at the mean contact point on the pinion surface have been derived. They are functions of the principal curvatures and directions at the mean contact point on the gear surface and the input of local synthesis. Based on the information on the characteristics of the pinion surface of the zero (position), first (normal), and second (principal curvatures and directions) orders, equations that determine the pinion basic machine-tool settings have been derived.

In Appendix A the basic concept and methods of theory of gearing that have been used in this work have been presented. Numerical examples are given in Appendix B. These examples include determination of machine-tool settings and results of computation by TCA programs. Computer programs have been developed, that include machine-tool settings and TCA. They are listed in Appendix C. The computer programs cover determination of machine-tool settings for straight-lined as well as curved blades. The developed TCA programs allow to simulate the meshing of aligned and misaligned gears.
APPENDIX A

GEOMETRY AND KINEMATICS OF GEARS IN THREE DIMENSIONS

A.1 Concept of Surfaces

Most of the ideas underlying gear theory are based on strict definitions proposed in the field of differential geometry. In what follows we introduce the concept that is applied in this report.

All in all we require that our functions can be differentiated at least once and usually more times. Accordingly we say a function $F$ belongs to class $C^n$ on an interval $I$ if the $n$th order derivative of $F$ exists and is continuous on $I$. In addition, we denote the class of continuous functions by $C^0$.

A parametric representation of a surface $\Sigma$ is a continuous mapping of an open rectangle $\mathcal{R}$, given in the plane $P$ of the parameters $(u, v)$, onto a three-dimensional space $E^3$ such that

$$\bar{B}(u, v) \in C^0, \quad (u, v) \in \mathcal{R} \tag{A.1}$$

where $\bar{B}$ is the position vector which determines the point surface (Figure 21). The vector function $\bar{B}(u, v)$ may be represented by

$$\bar{B}(u, v) = B_x(u, v) \hat{i} + B_y(u, v) \hat{j} + B_z(u, v) \hat{k} \tag{A.2}$$

1Adopted from the manuscript of the book “Theory of Gearing” by Litvin, in press by NASA.
Figure 21: A parametric representation of a surface.
where \( \mathbf{i}, \mathbf{j}, \) and \( \mathbf{k} \) are unit vectors of the coordinate axes.

We call a surface point \( \mathbf{B}(u, v) \) a regular point if at this point

\[
\mathbf{B}_u \times \mathbf{B}_v \neq 0 \tag{A.3}
\]

where

\[
\mathbf{B}_u = \frac{\partial \mathbf{B}}{\partial u}, \quad \mathbf{B}_v = \frac{\partial \mathbf{B}}{\partial v}
\]

A surface is called a regular one if each point on it is a regular point.

A regular surface has the following properties:

- It is at least class of \( C^1 \).

- There is a one-to-one correspondence between the points of plane \( P \) (of the parameters \( (u, v) \)) and the three-dimensional space \( E^3 \).

- A regular surface has a tangent plane at all its points.

The normal vector \( \mathbf{N} \) to the surface at a point \( \mathbf{B} \) is

\[
\mathbf{N} = \mathbf{B}_u \times \mathbf{B}_v \tag{A.4}
\]

and its unit normal is represented by

\[
\mathbf{n} = \frac{\mathbf{N}}{|\mathbf{N}|} = \frac{\mathbf{B}_u \times \mathbf{B}_v}{|\mathbf{B}_u \times \mathbf{B}_v|} \tag{A.5}
\]
The direction of the surface normal \( \vec{N} \) and unit normal \( \vec{n} \), with respect to the surface, depends on the order of the factors of the cross product (equation (A.4)). By changing the order of the factors, we may change the direction of the normal to the opposite direction.

A surface is uniquely determined by certain local invariant quantities called the first and second fundamental forms. The first fundamental form of a surface is defined by

\[
I = d\vec{B} \cdot d\vec{B} = (\vec{B}_u \, du + \vec{B}_v \, dv) \cdot (\vec{B}_u \, du + \vec{B}_v \, dv) = (\vec{B}_u \cdot \vec{B}_u) du^2 + 2(\vec{B}_u \cdot \vec{B}_v) du \, dv + (\vec{B}_v \cdot \vec{B}_v) dv^2 \tag{A.6}
\]

where we set

\[
E = \vec{B}_u \cdot \vec{B}_u, \quad F = \vec{B}_u \cdot \vec{B}_v, \quad G = \vec{B}_v \cdot \vec{B}_v
\]

The second fundamental form is

\[
II = -d\vec{B} \cdot d\vec{n} = -(\vec{B}_u \, du + \vec{B}_v \, dv) \cdot (\vec{n}_u \, du + \vec{n}_v \, dv) = -(\vec{B}_u \cdot \vec{n}_u) du^2 - (\vec{B}_u \cdot \vec{n}_v + \vec{B}_v \cdot \vec{n}_u) du \, dv - (\vec{B}_v \cdot \vec{n}_v) dv^2 \tag{A.7}
\]

where we have

\[
L = -\vec{B}_u \cdot \vec{n}_u, \quad M = \frac{1}{2}(\vec{B}_u \cdot \vec{n}_v + \vec{B}_v \cdot \vec{n}_u), \quad N = -\vec{B}_v \cdot \vec{n}_v
\]

The second fundamental form exists only if the surface is at least class \( C^2 \). In this report we will consider all the gear tooth surfaces as regular surfaces with class at least \( C^2 \).
On a given surface various curves pass through a common point $B$ and have the same unit tangent vector $\hat{\tau}$ at $B$ (Figure 22). One of these curves (designated by $L_0$) is located on the plane $P$, which is drawn through the unit tangent vector $\hat{\tau}$ and the surface unit normal $\hat{n}$. The curvature of curve $L_0$ is called normal curvature. Since the unit tangent vector $\hat{\tau}$ of the surface may have different directions on the surface, for each direction there is a normal curvature. The normal curvature is a function of the first and second fundamental forms:

$$\kappa_n = \frac{II}{I} = \frac{L \, du^2 + 2M \, du \, dv + N \, dv^2}{E \, du^2 + 2F \, du \, dv + G \, dv^2} \quad (A.8)$$

The extreme value of the normal curvature taken at a certain point of the surface are called the principal curvatures. The directions of the normal sections of the surface with the extreme normal curvatures are called the principal directions. Equation (A.8) yields

$$\mathcal{F} = \kappa_n (E \, du^2 + 2F \, du \, dv + G \, dv^2) - (L \, du^2 + 2M \, du \, dv + N \, dv^2) = 0 \quad (A.9)$$

For a given point on the surface, $E$, $F$, $G$, $L$, $M$, and $N$ are constant. The normal curvature $\kappa_n$ is a function of the ratio $du$ and $dv$. Therefore, equation (A.9) is an identity of $du$ and $dv$. From calculus, the partial derivative

$$\frac{\partial \mathcal{F}}{\partial du} = 0 \quad (A.10)$$

Substituting equation (A.9) into equation (A.10), it yields

$$\kappa_n (E \, du + F \, dv) - (L \, du + M \, dv) + \frac{\partial \kappa_n}{\partial du} (E \, du^2 + 2F \, du \, dv + G \, dv^2) = 0 \quad (A.11)$$
Figure 22: The normal curvature.
Also, the partial derivative

\[ \frac{\partial F}{\partial dv} = 0 \]  \hspace{1cm} (A.12)

Substituting equation (A.9) into equation (A.12), it yields

\[ \kappa_n(F \, du + G \, dv) - (M \, du + N \, dv) + \frac{\partial \kappa_n}{\partial dv} E \, du^2 + 2F \, du \, dv + G \, dv^2 = 0 \]  \hspace{1cm} (A.13)

Recall that the principal curvatures are the extreme values of the normal curvature \( \kappa_n \). Thus \( \partial \kappa_n/\partial du = 0 \) and \( \partial \kappa_n/\partial dv = 0 \) if \( \kappa_n \) is the principal curvature. Equations (A.11) and (A.13) yield

\[ (\kappa_n E - L) \, du + (\kappa_n F - M) \, dv = 0 \]  \hspace{1cm} (A.14)

and

\[ (\kappa_n F - M) \, du + (\kappa_n G - N) \, dv = 0, \]  \hspace{1cm} (A.15)

respectively. Solving the homogeneous system of equation (A.14) and (A.15) by eliminating \( du \) and \( dv \), we obtain

\[ (EG - F^2)\kappa_n^2 - (EN - 2FM + GL)\kappa_n + (LN - M^2) = 0 \]  \hspace{1cm} (A.16)

The discriminant of equation (A.16) is
\[ \begin{align*}
\Delta &= (EN - 2FM + GL)^2 - 4(EG - F^2)(LN - M^2) \\
&= \left[ (EN - GL) - \frac{2E}{F}(EM - FL) \right]^2 + \frac{4(EG - F^2)}{E^2}(EM - FL)^2
\end{align*} \] (A.17)

Equation (A.17) shows that the discriminant is greater than or equal to zero. Thus the equation has either two distinct real roots—the principal curvatures at a nonumbilical point, or a single real root with multiplicity two—the curvature at an umbilical point. The discriminant is equal to zero if and only if

\[ EN - GL = EM - FL = 0 \] (A.18)

Since \( E \neq 0 \) and \( G \neq 0 \), equation (A.18) can be shown to be identically equal to

\[ \frac{L}{E} = \frac{M}{F} = \frac{N}{G} = \kappa \] (A.19)

Considering equations (A.8) and (A.19), we obtain

\[ \kappa_n = \kappa \] (A.20)

This means that the principal curvature is the same as the normal curvature at any direction. Thus each direction may be considered as a principal direction. Any point which is on a plane or at which a surface turns into a plane\(^2\) is an umbilical point. Any point on a spherical surface\(^3\) is also an umbilical point.

Two distinct principal curvatures can always be obtained at a nonumbilical point. These two curvatures correspond to two distinct principal directions. By canceling \( \kappa_n \) from equations (A.14)

\(^2\)The normal curvature on each direction is zero.
\(^3\)The normal curvature on each direction is the inverse of the radius.
and (A.15), we have the following equation for principal directions

\[(EM - FL) du^2 - (GL - EN) du dv + (FN - GM) dv^2 = 0 \quad (A.21)\]

The discriminant of the above equation is identical to equation (A.17). At a nonumbilical point equation (A.21) can be represented as a product of two cofactors \((A_i \, du_i + B_i \, dv_i)(i = 1, 2)\) since the discriminant is larger than zero. This means that it represents two perpendicular directions. Thus we may conclude that at a nonumbilical point there exist two distinct principal curvatures in two perpendicular directions.

After representing of \(E, F, G, L, M,\) and \(N\) in the form of \(B, \xi, \iota, \phi,\) and \(\psi\) by equations (A.6) and (A.7), equations (A.14) and (A.15) will yield

\[B_u \cdot (\kappa_n \, d\vec{B} + d\vec{n}) = 0 \quad (A.22)\]

\[B_v \cdot (\kappa_n \, d\vec{B} + d\vec{n}) = 0 \quad (A.23)\]

Obviously,

\[\vec{n} \cdot (\kappa_n \, d\vec{B} + d\vec{n}) = 0 \quad (A.24)\]

Therefore, \(\kappa_n \, d\vec{B} + d\vec{n}\) is a zero vector since it is orthogonal to \(B_u, B_v,\) and \(\vec{n}\.\) In short, we have

\[d\vec{n} = -\kappa_n \, d\vec{B} \quad (A.25)\]
The above equation, which completely characterizes the principal curvatures and directions, is called Rodrigues' formula. This formula simplifies the calculations to obtain principal curvatures and principal directions. The matrix form of Rodrigues' formula is

\[
\begin{bmatrix}
\frac{\partial n_x}{\partial u} du + \frac{\partial n_x}{\partial v} dv \\
\frac{\partial n_y}{\partial u} du + \frac{\partial n_y}{\partial v} dv \\
\frac{\partial n_z}{\partial u} du + \frac{\partial n_z}{\partial v} dv
\end{bmatrix} = -\kappa_{i,j}
\begin{bmatrix}
\frac{\partial B_x}{\partial u} du + \frac{\partial B_x}{\partial v} dv \\
\frac{\partial B_y}{\partial u} du + \frac{\partial B_y}{\partial v} dv \\
\frac{\partial B_z}{\partial u} du + \frac{\partial B_z}{\partial v} dv
\end{bmatrix}
\] (A.26)

Matrix equation (A.26) yields three scalar equations in three unknowns, the ratio \(du/dv\), and the principal curvatures \(\kappa_i\) and \(\kappa_j\). Using any two of the scalar equations we may develop a quadratic equation (provided \(dv \neq 0\))

\[A_2 \left( \frac{du}{dv} \right)^2 + A_1 \frac{du}{dv} + A_0 = 0\] (A.27)

The two roots of this equation correspond to two principal directions on the surface. By putting both roots into the third scalar equation, we may determine the principal curvatures \(\kappa_i\) and \(\kappa_j\).

It is possible to have either positive or negative principal curvatures. The sense of the principal curvature depends on the location of the center of curvature on the normal. The principal curvature is positive if the center of curvature is located on the positive normal.

The normal curvature on each direction may be expressed in terms of principal curvatures. This is so called Euler's Theorem. That states

\[\kappa_n = \kappa_i \cos^2 \varphi + \kappa_j \sin^2 \varphi\] (A.28)

where \(\varphi\) is the angle formed by the tangent to the normal curvature and principal direction with curvature \(\kappa_i\).
A.2 Relations Between Principal Curvatures and Directions for Mating Surfaces

Consider two gear surfaces $\Sigma_F$ and $\Sigma_Q$ which are in meshing. Moreover, we have the following assumptions:

1. The rotation angles, $\phi_F$ and $\phi_Q$, of both gears are given;

2. The function $\phi_Q(\phi_F)$ has continuous derivatives of second order;

3. The angular velocity $\omega^{(F)}$ of gear $F$ is constant.

Then relations between principal curvatures and directions of these mating surfaces may be determined. Such relations were first proposed by Litvin [12] and then extended for the case $m_{FQ} \neq 0$ by Litvin and Gutman [3], where $m_{FQ} = \omega^{(F)}/\omega^{(Q)}$ is the gear ratio.

The relations may be expressed by a system of three linear equations in two unknowns $rQ'_{F}$ and $rQ'_{Q}$:

$$a_{j1} rQ'_{F} + a_{j2} rQ'_{Q} = a_{j3} \quad (j = 1, 2, 3) \tag{A.29}$$

where $rQ'_{F}$ and $rQ'_{Q}$ are the projections of the relative velocity $rQ'_{F}$ at the contact point $B$ on the principal directions on surface $\Sigma_Q$. The equation may be represented by a symmetric augmented matrix $[A]$. That is

$$[A] = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \tag{A.30}$$

Here
\[ a_{11} = \kappa_{\varphi_I} - \kappa_{\varphi_I} \cos^2 \sigma - \kappa_{\varphi_{II}} \sin^2 \sigma = \kappa_{\varphi_I} - \frac{\kappa_{\varphi_I} + \kappa_{\varphi_{II}}}{2} - \frac{\kappa_{\varphi_I} - \kappa_{\varphi_{II}}}{2} \cos 2\sigma \quad (A.31) \]

\[ a_{12} = a_{21} = \frac{\kappa_{\varphi_I} - \kappa_{\varphi_{II}}}{2} \sin 2\sigma \quad (A.32) \]

\[ a_{13} = a_{31} = -\kappa_{\varphi_I} V^{(\varphi_Q)}_{\varphi_I} - [\omega^{(\varphi_Q)} \vec{n} \vec{e}_{\varphi_I}] \quad (A.33) \]

\[ a_{22} = \kappa_{\varphi_{II}} - \kappa_{\varphi_I} \sin^2 \sigma - \kappa_{\varphi_{II}} \cos^2 \sigma = \kappa_{\varphi_{II}} - \frac{\kappa_{\varphi_I} + \kappa_{\varphi_{II}}}{2} - \frac{\kappa_{\varphi_I} - \kappa_{\varphi_{II}}}{2} \cos 2\sigma \quad (A.34) \]

\[ a_{23} = a_{32} = -\kappa_{\varphi_{II}} V^{(\varphi_Q)}_{\varphi_{II}} - [\omega^{(\varphi_Q)} \vec{n} \vec{e}_{\varphi_{II}}] \quad (A.35) \]

\[ a_{33} = \kappa_{\varphi_I} \left(V^{(\varphi_Q)}_{\varphi_I}\right)^2 + \kappa_{\varphi_{II}} \left(V^{(\varphi_Q)}_{\varphi_{II}}\right)^2 - [\vec{n} \omega^{(\varphi_Q)} V^{(\varphi_Q)}] \]

\[ - \vec{n} \cdot \left(\omega^{(\varphi)} \times t_v V^{(\varphi_Q)} - \omega^{(\varphi)} \times t_v V^{(\varphi_Q)}\right) + \left(\omega^{(\varphi)}\right)^2 m'_Q \left(\vec{n} \times \vec{e}_Q\right) \cdot (\vec{B} - \vec{O}_Q \vec{O}_P) \quad (A.36) \]

\( \kappa_{\varphi_I} \) and \( \kappa_{\varphi_{II}} \) are the principal curvatures at the contact point \( B \) of gear \( F \),

\( \kappa_{\varphi_I} \) and \( \kappa_{\varphi_{II}} \) are the principal curvatures at the contact point \( B \) of gear \( Q \),

\( \vec{e}_{\varphi_I} \) and \( \vec{e}_{\varphi_{II}} \) are the unit vectors of the principal directions at the contact point \( B \) of gear \( Q \),
\( \sigma \) is the angle measured counterclockwise from \( \hat{e}_{\varphi_1} \), the unit vector of the principal direction at the contact point \( B \) of gear \( \mathcal{F} \), to \( \hat{e}_{\varphi_1} \).

\( \omega^{(\mathcal{F})} \) and \( \omega^{(\mathcal{Q})} \) are the angular velocities of gears \( \mathcal{F} \) and \( \mathcal{Q} \), respectively,
\( \omega^{(\mathcal{F} \mathcal{Q})} \) is the relative angular velocity of gear \( \mathcal{F} \) with respect to gear \( \mathcal{Q} \),
\( \hat{n} \) is the common unit normal vector,
\( \vec{V}^{(\mathcal{F} \mathcal{Q})} \) is the relative velocity of the contact point on gear \( \mathcal{F} \) with respect to the same contact point on gear \( \mathcal{Q} \),
\( \vec{V}^{(\mathcal{F} \mathcal{Q})}_{\mathcal{Q}_I} \) and \( \vec{V}^{(\mathcal{F} \mathcal{Q})}_{\mathcal{Q}_H} \) are the projections of \( \vec{V}^{(\mathcal{F} \mathcal{Q})} \) on the \( \hat{e}_{\varphi_1} \) and \( \hat{e}_{\varphi_H} \), respectively,
\( t_{r,\mathcal{F}} \) and \( t_{r,\mathcal{Q}} \) are the transfer velocities of contact point \( B \) on gear \( \mathcal{F} \) and gear \( \mathcal{Q} \), respectively,
\( \vec{B} \) is the position vector of the common contact point \( B \),
\( \overrightarrow{O_{\mathcal{F}}O_{\mathcal{Q}}} \) is the position vector from \( O_{\mathcal{Q}} \) to \( O_{\mathcal{F}} \),
\( \hat{k}_{\mathcal{Q}} \) is the unit vector of the axis of rotation of gear \( \mathcal{Q} \), and
\( m'_{\mathcal{Q} \mathcal{F}} \) is the derivative of the rotation ratio of gear \( \mathcal{Q} \) to gear \( \mathcal{F} \). It is represented as

\[
m'_{\mathcal{Q} \mathcal{F}} = \frac{d}{d\phi_{\mathcal{F}}} m_{\mathcal{Q} \mathcal{F}}(\phi_{\mathcal{F}})
\]

where \( \phi_{\mathcal{F}} \) is the rotation angle of gear \( \mathcal{F} \), and

\[
m_{\mathcal{Q} \mathcal{F}}(\phi_{\mathcal{F}}) = \frac{\omega^{(\mathcal{Q})}}{\omega^{(\mathcal{F})}}
\]

Totally, there are two cases of tangency of gear tooth surfaces:

1. The surfaces \( \Sigma_{\mathcal{F}} \) and \( \Sigma_{\mathcal{Q}} \) are in line contact and \( B \) is just a point of the instantaneous line of contact.
2. The surfaces $\Sigma_\mathcal{F}$ and $\Sigma_\mathcal{Q}$ are in point contact and $B$ is the single point of tangency at the considered instant.

In the case of line contact of mating surfaces, the rank of matrix $[A]$ is equal to one. Thus all determinants of the second order formed from the elements of $[A]$ are zero. This yields

\[
\begin{pmatrix}
\frac{a_{11}}{a_{12}} &=& \frac{a_{12}}{a_{22}} &=& \frac{a_{13}}{a_{23}} \\
\end{pmatrix}
\]  

(A.37)

\[
\begin{pmatrix}
\frac{a_{11}}{a_{13}} &=& \frac{a_{12}}{a_{23}} &=& \frac{a_{13}}{a_{23}} \\
\end{pmatrix}
\]  

(A.38)

\[
\begin{pmatrix}
\frac{a_{12}}{a_{13}} &=& \frac{a_{22}}{a_{23}} &=& \frac{a_{23}}{a_{23}} \\
\end{pmatrix}
\]  

(A.39)

Using equations (A.31)-(A.39) we obtain

\[
\tan 2\sigma = \frac{2a_{13}a_{23}}{a_{23}^2 - a_{13}^2 + (\kappa_{Q_1} - \kappa_{Q_2})a_{33}}
\]

(A.40)

\[
\kappa_{Q_1} - \kappa_{Q_2} = \frac{a_{23}^2 - a_{13}^2 + (\kappa_{Q_1} - \kappa_{Q_2})a_{33}}{a_{33} \cos 2\sigma}
\]

(A.41)

\[
\kappa_{Q_1} + \kappa_{Q_2} = (\kappa_{Q_1} + \kappa_{Q_2}) - \frac{a_{13}^2 + a_{23}^2}{a_{33}}
\]

(A.42)

For the case when surfaces $\Sigma_\mathcal{F}$ and $\Sigma_\mathcal{Q}$ are in point contact, the rank of matrix $[A]$ described by equation (A.30) is two. Consequently,
Equation (A.43) yields the relation

\[ f(\kappa_x, \kappa_y, \kappa_{\alpha_x}, \kappa_{\alpha_y}, \sigma) = 0 \] (A.44)

In general, equations of the generated surface are evidently much more complicated than those of the generating one. Therefore, a direct way to obtain the principal curvatures and directions of the generated surface is a very difficult task. This work can be significantly simplified if we apply the relations, described in this section, between principal curvatures and directions of meshing surfaces.

### A.3 Relative Normal Curvature

The relative normal curvature, \( \kappa_r \), of two mating surfaces, \( \Sigma_x \) and \( \Sigma_Q \), at the contact point \( B \) is defined as the difference of the normal curvatures of both surfaces taken in a common normal section of surfaces and represented as

\[ \kappa_r = \kappa^{(Q)}_n - \kappa^{(x)}_n \] (A.45)

Suppose the common normal section form an angle \( \varpi \) with the unit vector \( \vec{e}_{\alpha_x} \) and an angle \( (\varpi + \sigma) \) with the unit vector \( \vec{e}_{\alpha_y} \) (Figure 23). According to Euler's Theorem (equation (A.28)), we obtain

\[ \kappa^{(Q)}_n = \kappa_{\alpha_x} \cos^2 \varpi + \kappa_{\alpha_y} \sin^2 \varpi \]
\[ \kappa^{(x)}_n = \kappa_{x_x} \cos^2 (\varpi + \sigma) + \kappa_{x_y} \sin^2 (\varpi + \sigma) \] (A.46)
Figure 23: A tangent plane to a surface
Substituting equation (A.46) into (A.45), after simple transformations we get

\[
\kappa_r = (\kappa_{\varphi_I} - \kappa_{\varphi_H} \cos^2 \sigma - \kappa_{\varphi_H} \sin^2 \sigma) \cos^2 \varpi + (\kappa_{\varphi_H} - \kappa_{\varphi_H} \sin^2 \sigma - \kappa_{\varphi_H} \cos^2 \sigma) \sin^2 \varpi \\
+ \frac{1}{2}(\kappa_{\varphi_I} - \kappa_{\varphi_H}) \sin 2\sigma \sin 2\varpi
\]  

(A.47)

Equation (A.47) and expressions for \(a_{11}, a_{12}, \text{ and } a_{22}\) in equations (A.31), (A.32), and (A.34) yield

\[
\kappa_r = \frac{1}{2} [a_{11} + a_{22} + (a_{11} - a_{22}) \cos 2\varpi] + a_{12} \sin 2\varpi
\]  

(A.48)

The extreme values of function \(\kappa_r(\varpi)\) may be determined by

\[
\frac{d}{d\varpi}(\kappa_r) = 0
\]  

(A.49)

Thus we obtain

\[
\tan 2\varpi = \frac{2a_{12}}{a_{11} - a_{22}}
\]  

(A.50)

This equation has two solutions \(\varpi_1\) and \(\varpi_2\). Moreover, \(|\varpi_1 - \varpi_2| = \frac{\pi}{2}\). This means that there are two perpendicular directions for the extreme relative normal curvatures. Using equations (A.48) and (A.50) the extreme values of the relative normal curvatures are represented by

\[
\kappa_r = \frac{1}{2} \left[ (a_{11} + a_{22}) \pm \sqrt{(a_{11} - a_{22})^2 + 4a_{12}^2} \right]
\]  

(A.51)

We may determine whether or not two surfaces interfere each other by the concept of relative normal curvatures. If two surfaces contact at a point with any interference, the sign of the relative
normal curvature in each direction must remain the same. In other words, the product of two extreme values of the relative normal curvatures is positive. Equations discussed in this section were first proposed by Litvin [9].

A.4 Contact Ellipse

In theory the tooth surfaces of a pair of spiral bevel gears are in contact at a single point at every instant. In practice the surface of the solids is deformed elastically over a region surrounding the initial point of contact, thereby bring the two bodies into contact over a small area in the neighborhood of the initial contact point [13, 14]. Such an area is an ellipse whose center of symmetry is the theoretical point of contact and the dimensions depend on the elastic approach and principal curvatures and directions of the contacting surfaces. If the approach of surfaces under the action of load is given, the size and orientation of the contact ellipse can be defined as a result of a geometric solution. Litvin[9, 15] has investigated the mathematical modeling of the contact ellipse.

Let us now consider that two surfaces $\Sigma_1$ and $\Sigma_2$ are contact at a single point $B$. The principal curvatures, $\kappa_1$ and $\kappa_2$ of $\Sigma_1$ and $\kappa_1'$ and $\kappa_2'$ of $\Sigma_2$, at point $B$ are known. Also known are unit vectors $\hat{e}_1$ and $\hat{e}_2'$, which are directed along the principal directions of $\Sigma_1$ at point $B$, and $\hat{e}_1'$ and $\hat{e}_2''$, which are directed along the principal directions of $\Sigma_2$ at point $B$. Unit vectors $\hat{e}_1$ and $\hat{e}_2'$ determine the tangent plane (Figure 24). Angle $\sigma_{12}$, which is measured counterclockwise from $\hat{e}_1$ to $\hat{e}_2'$, is also determined since $\hat{e}_1$ and $\hat{e}_2'$ have already been known. Then the contact ellipse may be described as

$$\frac{\zeta^2}{a^2} + \frac{\eta^2}{b^2} = 1 \quad \text{(A.52)}$$

in which $\zeta$ and $\eta$ are coordinates with respect to the $\zeta$ and $\eta$ axes with origin at the contact point $B$. The lengths of semiaxes $a$ and $b$ are
Figure 24: Contact ellipse on the tangent plane.
\[
a = \sqrt{\frac{\varepsilon}{A}}, \quad b = \sqrt{\frac{\varepsilon}{B}} \tag{A.53}
\]

where \( \varepsilon \) is the approach, and

\[
A = \frac{1}{4} \left( \kappa_{1\Sigma} - \kappa_{2\Sigma} - \sqrt{\kappa^{2}_{1\Delta} - 2\kappa_{1\Delta} \kappa_{2\Delta} \cos 2\sigma_{12} + \kappa^{2}_{2\Delta}} \right) \tag{A.54}
\]

\[
B = \frac{1}{4} \left( \kappa_{1\Sigma} - \kappa_{2\Sigma} + \sqrt{\kappa^{2}_{1\Delta} - 2\kappa_{1\Delta} \kappa_{2\Delta} \cos 2\sigma_{12} + \kappa^{2}_{2\Delta}} \right) \tag{A.55}
\]

where

\[
\kappa_{1\Sigma} = \kappa_{1I} + \kappa_{1II}, \quad \kappa_{2\Sigma} = \kappa_{2I} + \kappa_{2II} \tag{A.56}
\]

\[
\kappa_{1\Delta} = \kappa_{1I} - \kappa_{1II}, \quad \kappa_{2\Delta} = \kappa_{2I} - \kappa_{2II} \tag{A.57}
\]

The angle \( \alpha_1 \) which determines the orientation of the ellipse may be obtained by equations

\[
\cos 2\alpha_1 = \frac{\kappa_{1\Delta} - \kappa_{2\Delta} \cos 2\sigma_{12}}{\sqrt{\kappa^{2}_{1\Delta} - 2\kappa_{1\Delta} \kappa_{2\Delta} \cos 2\sigma_{12} + \kappa^{2}_{2\Delta}}} \tag{A.58}
\]

and

\[
\sin 2\alpha_1 = \frac{2\kappa_{2\Delta} \sin 2\sigma_{12}}{\sqrt{\kappa^{2}_{1\Delta} - 2\kappa_{1\Delta} \kappa_{2\Delta} \cos 2\sigma_{12} + \kappa^{2}_{2\Delta}}} \tag{A.59}
\]
\[
\alpha_1 = \arctan \frac{\sin 2\alpha_1}{1 + \cos 2\alpha_1} \tag{A.60}
\]

Note that the angle \( \alpha_1 \) is measured counterclockwise from the \( \eta \) axis to the unit vector \( \vec{e}_1 \).

Since

\[
A^2 - B^2 = \frac{1}{4}(\kappa_{2\Sigma} - \kappa_{1\Sigma}) \sqrt{\kappa_{1\Delta}^2 - 2\kappa_{1\Delta} \kappa_{2\Delta} \cos 2\tau_{12} + \kappa_{2\Delta}^2}
\]

the semimajor axis of the contact ellipse may be determined by the following conditions:

- The length of the semimajor axis, which is along the \( \eta \) axis, is \( b \)
  
  if \( \kappa_{2\Sigma} > \kappa_{1\Sigma} \) or \( |A| > |B| \).

- The length of the semimajor axis, which is along the \( \zeta \) axis, is \( a \)
  
  if \( \kappa_{1\Sigma} > \kappa_{2\Sigma} \) or \( |B| > |A| \).
APPENDIX B

NUMERICAL EXAMPLES

In this section, we will use the synthesis method discussed in Chapter 3 to determine the machine-tool settings for a pair of spiral bevel gear drive, and then we will use the TCA to simulate the meshing of this pair under alignment and misalignment conditions. Two cases are considered here. Both cases use straight blades to cut gears, but for the pinion, case 1 uses straight blades, and case 2 uses curved blades.

The major blank data is represented in Table 2. Table 3 shows the input for case 1, and Table 4 shows the input for case 2. The output for the gear machine-tool settings is shown in Table 5, which is the same for both cases. For the pinion machine-tool settings, case 1 is shown in Table 6, and case 2 is shown in Table 7.

Two conditions of misalignment are considered when the TCA is applied to simulate the meshing. They are the shift of pinion along its axis, which is denoted by $\Delta A$, and the error of pinion shaft offset, which denoted by $\Delta V$. We consider that $\Delta A$ is positive when the mounting distance of pinion is increased. The sense of $\Delta V$ is the same as $y$, shown in Figure 18. The output of the TCA is shown from Figure 25 to Figure 34 for case 1, and from Figure 35 to Figure 44 for case 2, respectively.
### Table 2: BLANK DATA.

<table>
<thead>
<tr>
<th>Pinion</th>
<th>Gear</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Teeth</td>
<td>10</td>
</tr>
<tr>
<td>Diametral Pitch</td>
<td>5.559</td>
</tr>
<tr>
<td>Shaft Angle</td>
<td>90°</td>
</tr>
<tr>
<td>Mean Cone Distance</td>
<td>3.226</td>
</tr>
<tr>
<td>Outer Cone Distance</td>
<td>3.796</td>
</tr>
<tr>
<td>Whole Depth</td>
<td>0.335</td>
</tr>
<tr>
<td>Working Depth</td>
<td>0.302</td>
</tr>
<tr>
<td>Clearance</td>
<td>0.033</td>
</tr>
<tr>
<td>Face Width</td>
<td>1.139</td>
</tr>
<tr>
<td>Root Cone Angle</td>
<td>12°1'</td>
</tr>
<tr>
<td>Mean Spiral Angle</td>
<td>—</td>
</tr>
<tr>
<td>Hand of Spiral</td>
<td>R.H.</td>
</tr>
</tbody>
</table>

### Table 3: INPUT DATA FOR CASE 1.

<table>
<thead>
<tr>
<th>Gear Convex Side</th>
<th>Gear Concave Side</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gear Blade Angle</td>
<td>20°</td>
</tr>
<tr>
<td>Gear Cutter Average Diameter</td>
<td>6</td>
</tr>
<tr>
<td>Gear Cutter Point Width</td>
<td>0.08</td>
</tr>
<tr>
<td>First Derivative of Gear Ratio</td>
<td>-0.0035</td>
</tr>
<tr>
<td>Semimajor Axis of Contact Ellipse</td>
<td>0.171</td>
</tr>
<tr>
<td>Contact Path Direction Angle</td>
<td>90°</td>
</tr>
</tbody>
</table>
Table 4: INPUT DATA FOR CASE 2.

<table>
<thead>
<tr>
<th></th>
<th>Gear Convex Side</th>
<th>Gear Concave Side</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gear Blade Angle</td>
<td>20°</td>
<td></td>
</tr>
<tr>
<td>Gear Cutter Average Diameter</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>Gear Cutter Point Width</td>
<td>0.08</td>
<td></td>
</tr>
<tr>
<td>First Derivative of Gear Ratio</td>
<td>-0.0037</td>
<td>0.0055</td>
</tr>
<tr>
<td>Semimajor Axis of Contact Ellipse</td>
<td>0.171</td>
<td>0.171</td>
</tr>
<tr>
<td>Contact Path Direction Angle</td>
<td>90°</td>
<td>75°</td>
</tr>
<tr>
<td>Radius of Blade</td>
<td>40.</td>
<td>50.</td>
</tr>
</tbody>
</table>

Table 5: GEAR MACHINE-TOOL SETTINGS.

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Radial</td>
<td>2.87798</td>
</tr>
<tr>
<td>Cradle Angle</td>
<td>58.6365</td>
</tr>
<tr>
<td>Ratio of Roll</td>
<td>0.973748</td>
</tr>
</tbody>
</table>
Table 6: PINION MACHINE-TOOL SETTINGS WITH STRAIGHT BLADE.

<table>
<thead>
<tr>
<th></th>
<th>Pinion Concave Side</th>
<th>Pinion Convex Side</th>
</tr>
</thead>
<tbody>
<tr>
<td>Blade Angle</td>
<td>16.5561°</td>
<td>22.9907°</td>
</tr>
<tr>
<td>Tip Radius of Cutter</td>
<td>2.96469</td>
<td>3.07037</td>
</tr>
<tr>
<td>Radial</td>
<td>2.99331</td>
<td>2.69783</td>
</tr>
<tr>
<td>Cradle Angle</td>
<td>63.1869°</td>
<td>54.1910°</td>
</tr>
<tr>
<td>Ratio of Roll</td>
<td>0.22900</td>
<td>0.25348</td>
</tr>
<tr>
<td>Machining Offset</td>
<td>0.17404</td>
<td>-0.24459</td>
</tr>
<tr>
<td>Machine Center to Back + Sliding Base</td>
<td>0.021231</td>
<td>0.052118</td>
</tr>
</tbody>
</table>

Table 7: PINION MACHINE-TOOL SETTINGS WITH CURVED BLADE.

<table>
<thead>
<tr>
<th></th>
<th>Pinion Concave Side</th>
<th>Pinion Convex Side</th>
</tr>
</thead>
<tbody>
<tr>
<td>Blade Angle</td>
<td>16.5561°</td>
<td>22.9907°</td>
</tr>
<tr>
<td>Blade Center</td>
<td>(11.557, 0., -35.309)</td>
<td>(19.685, 0., 49.006)</td>
</tr>
<tr>
<td>Tip Radius of Cutter</td>
<td>2.98467</td>
<td>3.04386</td>
</tr>
<tr>
<td>Radial</td>
<td>2.95578</td>
<td>2.74261</td>
</tr>
<tr>
<td>Cradle Angle</td>
<td>63.0025°</td>
<td>54.0900°</td>
</tr>
<tr>
<td>Ratio of Roll</td>
<td>0.23157</td>
<td>0.24915</td>
</tr>
<tr>
<td>Machining Offset</td>
<td>0.12042</td>
<td>-0.18825</td>
</tr>
<tr>
<td>Machine Center to Back + Sliding Base</td>
<td>0.01690</td>
<td>0.03605</td>
</tr>
</tbody>
</table>
Figure 25: Straight-edged blade, gear convex side, alignment.
Figure 26: Straight-edged blade, gear convex side, $\Delta A = +0.002$ inches.
Figure 27: Straight-edged blade, gear convex side, $\Delta A = -0.002$ inches.
Figure 28: Straight-edged blade, gear convex side, $\Delta V = +0.002$ inches.
Figure 29: Straight-edged blade, gear convex side, $\Delta V = -0.002$ inches.
Figure 30: Straight-edged blade, gear concave side, alignment.
Figure 31: Straight-edged blade, gear concave side, $\Delta A = +0.002$ inches.
Figure 32: Straight-edged blade, gear concave side, \( \Delta A = -0.002 \) inches.
Figure 33: Straight-edged blade, gear concave side, $\Delta V = +0.002$ inches.
Figure 34: Straight-edged blade, gear concave side, $\Delta V = -0.002$ inches.
Figure 35: Curved-edged blade, gear convex side, alignment.
Figure 36: Curved-edged blade, gear convex side, $\Delta A = +0.002$ inches.
Figure 37: Curved-edged blade, gear convex side, $\Delta A = -0.002$ inches.
Figure 38: Curved-edged blade, gear convex side, \( \Delta V = +0.002 \) inches.
Figure 39: Curved-edged blade, gear convex side, $\Delta V = -0.002$ inches.
Figure 40: Curved-edged blade, gear concave side, alignment.
Figure 41: Curved-edged blade, gear concave side, $\Delta A = +0.002$ inches.
Figure 42: Curved-edged blade, gear concave side, $\Delta A = -0.002$ inches.
Figure 43: Curved-edged blade, gear concave side, $\Delta V = +0.002$ inches.
Figure 44: Curved-edged blade, gear concave side, $\Delta V = -0.002$ inches.
APPENDIX C

LISTING OF COMPUTER PROGRAMS

*******************************************************************************
* Gleason's Spiral Bevel Gears
* Basic Machine-Tool Settings and Tooth Contact Analysis
* Straight Blade to Cut the Pinion
*******************************************************************************

IMPLICIT REAL*8 (A-H,K,M-Z)
REAL*8 X(1),F(1),FI(1),PAR(6),LM,TX(5),TF(5),TF1(5),TPAR(19),
   AZSP(1,1),WORKP(1),AZS(5,5),WORK(5)
CHARACTER*8 HG,HNGR
DIMENSION IPVTP(1),IPVT(5)
EXTERNAL PCN,TCN
COMMON/P1/PAR
COMMON/T1/TPAR
COMMON/A0/HG
COMMON/A1/p11,p12,p13,p21,p22,p31,p32,p33,p1,p2,p3
COMMON/A2/SG,CSRT2,QG,SNPSIG,CSPSIG
COMMON/A3/TND1,TND2,RITAG
COMMON/A4/CSD2,SND2,CSPIT2,SNPIT2
COMMON/A5/CSQG,SNQG,THETAG
COMMON/B1/CSPH11,SNPH11,SP,EM,LM,CSRT1,CSD1,SND1,CSPSIP,SNPSIP
COMMON/B2/CSPIT1,SNPIT1,MP1,MG2,QP
COMMON/B3/B2fx,B2fy,B2ez
COMMON/B4/CSPH2,SNPH2,CSPH21,SNPH21
COMMON/C1/UG,CSTAUG,SNTAUG
COMMON/C2/N2fx,N2fy,N2ez
COMMON/D1/UP,CSTAUP,SNTAUP
COMMON/E1/XBF,YBF,ZBF
COMMON/F1/PHIG0
COMMON/G1/DAL,DV1
*
* INPUT THE DESIGN DATA
*  
* TNI : number of pinion teeth
* ---- sec. 3.1

129
* TN2 : number of gear teeth
  ----- sec. 3.1
* RT1dg, RT1min : root angle of pinion (degree and arc minute, respectively)
  ----- sec. 3.1
* RT2dg, RT2min : root angle of gear (degree and arc minute, respectively)
  ----- sec. 3.1
* SHAFdg : shaft angle (degree)
  ----- sec. 3.1
* BETAdg : mean spiral angle (degree)
  ----- sec. 3.1
* ADIA : average gear cutter diameter
  ----- sec. 3.1
* W : point width of gear cutter
  ----- sec. 3.1
* A : mean cone distance
  ----- sec. 3.1
* ALPHdg : blade angle of gear cutter (degree)
  ----- sec. 3.1
* DLTXdg : angle measured counterclockwise from root of gear to
  the tangent of the contact path (degree)
  gear convex side
  ----- fig. 19
* DLTVdg : angle measured counterclockwise from root of gear to
  the tangent of the contact path (degree)
  gear concave side
  ----- fig. 19
* M21XPR : first derivative of gear ratio
  gear convex side
  ----- sec. 3.1.1
* M21VPR : first derivative of gear ratio
  gear concave side
  ----- sec. 3.1.1
* AXILX : semimajor axis of contact ellipse
  gear convex side
  ----- eq. (3.76)
* AXILV : semimajor axis of contact ellipse
  gear concave side
  ----- eq. (3.76)
* HNGR : hand of gear ('L' or 'R')
* DA : amount of shift along pinion axis
  + : pinion mounting distance being increased
  - : pinion mounting distance being decreased
* DV : amount of pinion shaft offset
  the same sense as yf shown in fig. 18
* DEF : elastic approach
  ----- eq. (3.76)
* EPS : amount to control calculation accuracy

* OUTPUT OF THE BASIC MACHINE-TOOL SETTINGS

* PSIGdg : gear blade angle
* PSIPdg  : pinion blade angle
* RG      : tip radius of gear cutter
* RP      : tip radius of pinion cutter
* SG      : gear radial
* SP      : pinion radial
* QGdg    : gear cradle angle
* QPdg    : pinion cradle angle
* MG2     : gear cutting ratio
* MP1     : pinion cutting ratio
* EM      : machining offset
* LM      : machine center to back + sliding base
*
DATA TN1,TN2/10.D00,41.D00/
DATA RT1dg,RT1min/12.D00,1.D00/
DATA RT2dg,RT2min/72.D00,25.D00/
DATA SHAFdg,BETAdg/90.D00,35.D00/
DATA AD1A/6.0D00/
DATA W/0.08D00/
DATA A/3.226D00/
DATA ALPHdg/20.D00/
DATA DLTXdg/ 90.D00/
DATA DLTVdg/ 75.D00/
DATA M21XPR/-3.5D-03/
DATA M21VPR/5.2D-03/
DATA AXILX/0.1710D00/
DATA AXILV/0.1810D00/
DATA HNGR/'L'/
DATA DV,DA/0.D00,0.D00/
DATA DEF/0.00025D00/
DATA EPS/1.D-12/
*
*
DA1=DA
DV1=DV
HG=HNGR
*
* CONVERT DEGREES TO RADIANS
*
CNST=4.D00*DATAN(1.D00)/180.D00
RITAG=90.D00*CNST
DLTX=DLTXdg*CNST
DLTV=DLTVdg*CNST
RT1=(RT1dg+RT1min/60.D00)*CNST
RT2=(RT2dg+RT2min/60.D00)*CNST
BETA=BETAdg*CNST
PSIG=ALPHdg*CNST
SHAFT=SHAFdg*CNST
CSRT2=DCOS(RT2)
SNRT2=DSIN(RT2)
CSRT1=DCOS(RT1)
SNRT1=DSIN(RT1)
*
* CALCULATE PITCH ANGLES

\[
MM21 = \frac{TN1}{TN2}
\]

\[\text{eq. (3.1)}\]

\[
PITCH2 = \text{Datan} \left( \frac{DSIN(Shaft)}{MM21 + DCOS(Shaft)} \right)
\]

\[
\text{IF}(PITCH2 .LT. 0.D00) \text{THEN}
\]

\[
PITCH2 = PITCH2 + 180.D00
\]

\[
\text{END IF}
\]

\[
CSPIT2 = DCOS(PITCH2)
\]

\[
SNPIT2 = DSIN(PITCH2)
\]

\[\text{eq. (3.2)}\]

\[
PITCH1 = SHAFT - PITCH2
\]

\[
CSPIT1 = DCOS(PITCH1)
\]

\[
SNPIT1 = DSIN(PITCH1)
\]

* CALCULATE DEENDUM ANGLES

\[\text{eq. (3.3)}\]

\[
D1 = PITCH1 - RT1
\]

\[
D2 = PITCH2 - RT2
\]

\[
CSD1 = DCOS(D1)
\]

\[
SND1 = DSIN(D1)
\]

\[
TND1 = SND1/CSD1
\]

\[
CSD2 = DCOS(D2)
\]

\[
SND2 = DSIN(D2)
\]

\[
TND2 = SND2/CSD2
\]

* CALCULATE GEAR CUTTING RATIO

\[\text{eq. (3.7)}\]

\[
MG2 = DSIN(PITCH2)/CSD2
\]

* FOR GEAR CONVEX SIDE I = 1, FOR GEAR CONCAVE SIDE I = 2.

\[
\text{DO 99999 I=1,2}
\]

\[
\text{IF(I .EQ. 1)THEN}
\]

\[
\text{WRITE(72,*) 'GEAR CONVEX SIDE'}
\]

\[
DLTA = DLTX
\]

\[
M21PRM = M21XPR
\]

\[
AXIL = AXILX
\]

\[
\text{ELSE}
\]

\[
\text{WRITE(72,*) 'GEAR CONCAVE SIDE'}
\]

\[
DLTA = DLTV
\]

\[
M21PRM = M21VPR
\]

\[
AXIL = AXILV
\]

\[
\text{END IF}
\]

\[
\text{WRITE(72,*)}
\]

\[\text{eq. (3.76)}\]

\[
AXIA = DEF/(AXIL*AXIL)
\]

* CALCULATE GEAR BLADE ANGLE

* sec. 2.2
IF(I .EQ. 2) THEN
  PSIG = 180.0 * CNST - PSIG
END IF
CPSIG = DCOS(PSIG)
SNPSIG = DSIN(PSIG)
CTPSIG = CPSIG / SNPSIG

* CALCULATE CUTTER TIP RADIUS
*

* ----- eq. (3.8)
  IF(I .EQ. 1) THEN
    RG = (ADIA - W) / 2.00
  ELSE
    RG = (ADIA + W) / 2.00
  END IF
*

* CALCULATE RADIAL
*

* ----- eq. (3.9)
  IF(I .EQ. 1) THEN
    SG = DSQURT((ADIA * ADIA / 4.00 - A * A * CSD2 * ADIA * ADIA - A * ADIA * CSD2 * DSIN(BETA))
  END IF
*

* CALCULATE CRADLE ANGLE
*

* ----- eq. (3.10)
  QG = DACOS((A * A * CSD2 * CSD2 + SG * SG - ADIA * ADIA / 4.00) / (2.00 * A * SG * CSD2))
  CSG = DCOS(QG)
  SNG = DSIN(QG)
END IF
*

PAR(1) = RG * CTPSIG * CPSIG
PAR(4) = RG * CTPSIG

* CALCULATE PHIG
*

PHIG = 0.00
PHIGO = PHIG
CSPHIG = DCOS(PHIG)
SNPHIG = DSIN(PHIG)

* IF(HG .EQ. 'L') THEN
  IF(I .EQ. 1) THEN
  * Mmc = Mms * Msc
  * ----- eq. (2.26)
    CALL COMBI(m11, m12, m13, m21, m22, m23, m31, m32, m33, m1, m2, m3,
    1.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00,
    1.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00,
    0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00,
    END IF
  * Mpc = Hpm * Mme
  * ----- eqs. (2.25), (3.13)
    CALL COMBI(p11, p12, p13, p21, p22, p23, p31, p32, p33, p1, p2, p3,
CSD2, 0.D00, -SND2, 0.D00, 1.D00, 0.D00, SND2, 0.D00, CSD2,
0.D00, 0.D00, 0.D00,
ml1, ml2, ml3, m21, m22, m23, m31, m32, m33, ml, m2, m3)

* ELSE *

IF (I .EQ. 1) THEN

Mmc = Mms * Msc

CALL COMBI (mll, m12, m13, m21, m22, m23, m31, m32, m33, ml, m2, m3,
1.D00, 0.D00, 0.D00, 0.D00, CPHIG, -SNPHIG, 0.D00, SNPHIG, CPHIG,
0.D00, 0.D00, 0.D00,
1.D00, 0.D00, 0.D00, 0.D00, CSQG, SQG, 0.D00, -SQG, CSQG,
0.D00, SQG * SQG, SQG * CSQG)
END IF

Mpc = Mpm * Mmc

CALL COMBI (pll, p12, p13, p21, p22, p23, p31, p32, p33, pl, p2, p3,
CSD2, 0.D00, -SND2, 0.D00, 1.D00, 0.D00, SND2, 0.D00, CSD2,
0.D00, 0.D00, 0.D00,
ml1, ml2, ml3, m21, m22, m23, m31, m32, m33, ml, m2, m3)
END IF

* DETERMINE MAIN CONTACT POINT *

* *

* CALCULATE THETAG *

* X(1) represents THETAG

PAR (2) = (MG2 - SNRT2) * CPSIG

IF (HG .EQ. 'L') THEN

PAR (3) = -SNQG * CSRT2 * SNPSIG

STEP 1 IN SEC. 3.2
X(1) = (QG - BETA + RITAG)
ELSE

PAR (3) = SNQG * CSRT2 * SNPSIG

STEP 1 IN SEC. 3.2
X(1) = -(QG - BETA + RITAG)
END IF

CALL NONLIN (PCN, 14, 1, 100, X, F, FI, 1.D-5, AZSP, IPVTP, WORKP)
THETAG = X(1)
CSTHEG = DCOS (THETAG)
SNTHEG = DSIN (THETAG)

* *

* CALCULATE TAUG *

* eq. (2.38)

IF (HG .EQ. 'L') THEN

TAUG = THETAG - QG + PHIG
ELSE

TAUG = THETAG + QG - PHIG
END IF

CSTAUG = DCOS (TAUG)
\[ SNTAUG = DSIN(TAUG) \]

* * * CALCULATE UG * * *

\[
\text{IF}(HG \, \text{EQ.} \, 'L') \text{THEN} \\
UG = \frac{RG \times CTPSIG \times CSPSIG \times SG \times ((MG2 - SNRT2) \times CSPSIG \times SNTHEG - DSIN(QG - PHIG) \times CSRT2 \times SNPSIG)}{CSRT2 \times SNTAUG} \\
\text{ELSE} \\
UG = \frac{RG \times CTPSIG \times CSPSIG \times SG \times ((MG2 - SNRT2) \times CSPSIG \times SNTHEG + DSIN(QG - PHIG) \times CSRT2 \times SNPSIG)}{CSRT2 \times SNTAUG} \\
\text{END IF}
\]

* * * CALCULATE MAIN CONTACT POINT * * *

\[
Bcx = RG \times CTPSIG - UG \times CSPSIG \\
Bcy = UG \times SNPSIG \times SNTHEG \\
Bcz = UG \times SNPSIG \times CSTHEG
\]

\[
\text{eq. (2.1)} \\
Ncx = SNPSIG \\
Ncy = CSPSIG \times SNTHEG \\
Ncz = CSPSIG \times CSTHEG
\]

\[
\text{eq. (2.2)} \\
\text{eq. (2.9)} \\
\text{CALL TRCOOR} (Bpx, Bpy, Bpz, \\
\quad p11, p12, p13, p21, p22, p23, p31, p32, p33, p1, p2, p3, \\
\quad Bcx, Bcy, Bcz)
\]

\[
\text{CALL TRCOOR} (Npx, Npy, Npz, \\
\quad p11, p12, p13, p21, p22, p23, p31, p32, p33, 0.D00, 0.D00, 0.D00, \\
\quad Ncx, Ncy, Ncz)
\]

\[
\text{CALL TRCOOR} (EGIpx, EGIpy, EGIpz, \\
\quad p11, p12, p13, p21, p22, p23, p31, p32, p33, 0.D00, 0.D00, 0.D00, \\
\quad EGIcx, EGIcy, EGIcz)
\]

\[
\text{fig. 18} \, \& \, \text{sec. 3.3} \\
Bfx = Bpx \\
Bfy = Bpy \\
Bfz = Bpz \\
Nfx = Npx \\
Nfy = Npy \\
Nfz = Npz \\
EGIfx = EGIpx \\
EGIfy = EGIpy \\
EGIfz = EGIpz
\]

* * * CALCULATE PSIP * * *

\[
\text{eq. (3.83)}
\]
PSIP = DASIN(CSD1*Nfx - SND1*Nfz)
IF (I .EQ. 1) THEN
    PSIP = -PSIP + 180.00*CNST
END IF
CSPSIP = DCOS(PSIP)
SNPSIP = DSIN(PSIP)

*CALCULATE TAUP*

c ----- eqs. (3.84)-(3.86)
TAUP = DATAN2(Nfy/CSPSIP, (Nfx - CSD1*SNPSIP)/(-SND1*CSPSIP))
CSTAUP = DCOS(TAUP)
SNTAUP = DSIN(TAUP)

*CALCULATE PRINCIPAL CURVATURES AND DIRECTIONS OF THE GEAR CUTTER*

c ----- eq. (2.10)
KGI = -CTPSIG/UG
c ----- eq. (2.12)
KGI1 = 0.00

The second principal direction is determined by rotating of
the first principal direction about the unit normal by 90 degrees
CALL ROTATE(EGIIfx, EGIIfy, EGIIfz, EGIIfx, EGIIfy, EGIIfz, RITAG,
. Nfx, Nfy, Nfz)

*CALCULATE W2G*

c ----- eqs. (3.18)-(3.20)
IF(HG .EQ. 'L') THEN
    W2fx = SNPIT2
    WGFx = -MG2*CSD2
    W2fy = 0.00
    WGFy = 0.00
    W2fz = CSPIT2
    WGFz = -MG2*SND2
ELSE
    W2fx = SNPIT2
    WGFx = MG2*CSD2
    W2fy = 0.00
    WGFy = 0.00
    W2fz = -CSPIT2
    WGFz = MG2*SND2
END IF

*CALCULATE VT2, VTG, AND VT2G*

c ----- eq. (3.22)
CALL CROSS(VT2fx, VT2fy, VT2fz, W2fx, W2fy, W2fz, Bfx, Bfy, Bfz)
c ----- eq. (3.21)
CALL CROSS(VTGfx,VTGfy,VTGfz,WGfx,WGfy,WGfz,Bfx,Bfy,Bfz)
c ------ eq. (3.23)
VT2Gfx=VT2fx-VTGfx
VT2Gfy=VT2fy-VTGfy
VT2Gfz=VT2fz-VTGfz

* CALCULATE V(2G) GI AND V(2G) GII

C ------ eq. (3.24)
CALL DOT(VGI,EGIfx,EGIfy,EGIfz,VT2Gfx,VT2Gfy,VT2Gfz)
c ------ eq. (3.25)
CALL DOT(VGII,EGIIfx,EGIIfy,EGIIfz,VT2Gfx,VT2Gfy,VT2Gfz)

* CALCULATE A13, A23, A33

C ------ eq. (3.26)
CALL DET(DETI,W2Gfx,W2Gfy,W2Gfz,Nfx,Nfy,Nfz,EGIfx,EGIfy,EGIfz)
A13=-KGI*VGI-DETI
c ------ eq. (3.27)
CALL DET(DETII,W2Gfx,W2Gfy,W2Gfz,Nfx,Nfy,Nfz,EGIIfx,EGIIfy,EGIIfz)
A23=-KGII*VGII-DETII
c ------ eq. (3.28)
CALL DET(DET3,Nfx,Nfy,Nfz,W2Gfx,W2Gfy,W2Gfz,VT2Gfx,VT2Gfy,VT2Gfz)
CALL CROSS(Cx,Cy,Cz,W2fx,W2fy,W2fz,VTGfx,VTGfy,VTGfz)
CALL CROSS(Dx,Dy,Dz,WGfx,WGfy,WGfz,VT2fx,VT2fy,VT2fz)
CALL DOT(DET45,Nfx,Nfy,Nfz,Cx-Dx,Cy-Dy,Cz-Dz)
A33=KGI*VGI*VGI+KGII*VGII*VGII-DET3-DET45

* CALCULATE SIGMA

C ------ eq. (3.29)
P=A23*A23-A13*A13+(KGI-KGII)*A33
SIGDBL=DATAN(2.DO0*A13*A23/P)
SIGMA=0.5DO0*SIGDBL

* CALCULATE K21 AND K2II

C ------ eqs. (3.30)-(3.31)
T1=P/(A33*DCOS(SIGDBL))
T2=KGI+KGII-(A13*A13+A23*A23)/A33
K2I=(T1+T2)/2.DO0
K2II=(T2-T1)/2.DO0

* CALCULATE E2I AND E2II

C ------ description after eq. (3.29)
CALL ROTATE(E2Ifx,E2Ify,E2Ifz,EGIfx,EGIfy,EGIfz,-SIGMA,Nfx,Nfy,
.Nfz)
CALL ROTATE(E2IIfx,E2IIfy,E2IIfz,E2Ifx,E2Ify,E2Ifz,RITAG,
.Nfx,Nfy,Nfz)
c ------ eq. (3.44)
TNETAG=DSIN(DLTA+SIGMA)/DCOS(DLTA+SIGMA)
* CALCULATE W2

* 

\[
\text{----- eq. (3.33)}
\]

\[
\begin{align*}
\text{IF}(HG \ .EQ. 'L') & \text{THEN} \\
W2fx & =-MM21*SNPIT2 \\
W2fy & =0.D00 \\
W2fz & =MM21*CSPLIT2 \\
\text{ELSE} \\
W2fx & =MM21*SNPIT2 \\
W2fy & =0.D00 \\
W2fz & =-MM21*CSPLIT2 \\
\end{align*}
\]

END IF

* 

* CALCULATE W1

* 

\[
\text{----- eq. (3.32)}
\]

\[
\begin{align*}
\text{IF}(HG \ .EQ. 'L') & \text{THEN} \\
W1fx & =-SNPIT1 \\
W1fy & =0.D00 \\
W1fz & =-CSPLIT1 \\
\text{ELSE} \\
W1fx & =SNPIT1 \\
W1fy & =0.D00 \\
W1fz & =CSPLIT1 \\
\end{align*}
\]

END IF

* 

* CALCULATE W12

* 

\[
\text{----- eq. (3.34)}
\]

\[
\begin{align*}
W12fx & =W1fx-W2fx \\
W12fy & =W1fy-W2fy \\
W12fz & =W1fz-W2fz \\
\end{align*}
\]

* 

* CALCULATE VT2

* 

\[
\text{----- eq. (3.36)}
\]

CALL CROSS(VT2fx,VT2fy,VT2fz,W2fx,W2fy,W2fz,Bfx,Bfy,Bfz)

* 

* CALCULATE VT1

* 

\[
\text{----- eq. (3.35)}
\]

CALL CROSS(VT1fx,VT1fy,VT1fz,W1fx,W1fy,W1fz,Bfx,Bfy,Bfz)

* 

* CALCULATE VT12

* 

\[
\text{----- eq. (3.37)}
\]

\[
\begin{align*}
VT12fx & =VT1fx-VT2fx \\
VT12fy & =VT1fy-VT2fy \\
VT12fz & =VT1fz-VT2fz \\
\end{align*}
\]

* 

* CALCULATE V2

* 

\[
\text{----- eq. (3.38)}
\]
CALL DOT(V21,VT12fx,VT12fy,VT12fz,E21fx,E21fy,E21fz)
c ---- eq. (3.39)
 CALL DOT(V211,VT12fx,VT12fy,VT12fz,E211fx,E211fy,E211fz)
* 
* CALCULATE A31
* 
c ---- eq. (3.40)
 CALL DET(DET1,W12fx,W12fy,W12fz,Nfx,Nfy,Nfz,E21fx,E21fy,E21fz)
A31=-K2I*V2I-DET1
c ---- eq. (A.33)
A13=A31
* 
* CALCULATE A32
* 
c ---- eq. (3.41)
 CALL DET(DET2,W12fx,W12fy,W12fz,Nfx,Nfy,Nfz,E211fx,E211fy,E211fz)
A32=-K2II*V211-DET2
c ---- eq. (A.35)
A23=A32
* 
* CALCULATE A33
* 
c ---- eq. (3.42)
 CALL DET(DET3,Nfx,Nfy,Nfz,W12fx,W12fy,W12fz,VT12fx,VT12fy,VT12fz)
 CALL CROSS(Cx,Cy,Cz,W1fx,W1fy,W1fz,VT2fx,VT2fy,VT2fz)
 CALL CROSS(Dx,Dy,Dz,W2fx,W2fy,W2fz,VT1fx,VT1fy,VT1fz)
 CALL DOT(DOT1,Nfx,Nfy,Nfz,Cx-Dx,Cy-Dy,Cz-Dz)
 CALL DET(DET4,Nfx,Nfy,Nfz,W2fx,W2fy,W2fz,Bfx,Bfy,Bfz)
A33=K21*V2I*V2I+K2II*V2II*V2II-DET3-DOTI+M21PRM*DET4
* 
* CALCULATE ETAP
* 
c ---- eq. (3.53)
 ETAP=DATAN(( (A33+A31*V2I)*TNETAG-A31*V2I)/ (A33+A32*(V211-V2I*TNETAG)))
 TNETAP=DSIN(ETAP)/DCOS(ETAP)
* 
* CALCULATE A11, A12, AND A22
* 
 N3=(1.D00+TNETAP*TNETAP)*A33
c ---- eq. (3.72)
 N1=(A13*A13-(A23*TNETAP)**2)/N3
c ---- eq. (3.73)
 N2=(A23+A13*TNETAP)*(A13+A23*TNETAP)/N3
 KS2=K21+K2II
 G2=K21-K2II
c ---- eqs. (3.74), (3.75)
 KS1=KS2-((4.D00*AXIA*AXIA-N1*N1-N2*N2)*(1.D00+TNETAP*TNETAP)/
 (-2.D00*AXIA*(1.D00+TNETAP*TNETAP)+N1*(TNETAP*TNETAP-1.D00)
 -2.D00*N2*TNETAP))
c ---- eqs. (3.66), (3.69) & description after eq. (3.60)
 A11=TNETAP*TNETAP/(1.D00+TNETAP*TNETAP)*(KS2-KS1)+N1
c ---- eqs. (3.67), (3.70) & description after eq. (3.60)
\[ A_{12} = -\frac{T_{NETAP}}{(1.00 + T_{NETAP} \cdot T_{NETAP})} \cdot (K_{S2} - K_{S1}) + N_{2} \]

\[ A_{22} = \frac{1.00}{(1.00 + T_{NETAP} \cdot T_{NETAP})} \cdot (K_{S2} - K_{S1}) - N_{1} \]

\[ A_{21} = A_{12} \]

\[ \text{CALCULATE } \Sigma(12) \]

\[ \text{eq. (A.32)} \]

\[ \text{eq. (3.77)} \]

\[ \text{SIGDBL} = \text{DATA}\text{TAN}(2.00 \cdot A_{12} / (K_{21} - K_{211} - A_{11} + A_{22})) \]

\[ \text{SIGM}_{12} = 0.500 \cdot \text{SIGDBL} \]

\[ \text{CALCULATE } K_{11} \text{ AND } K_{111} \]

\[ \text{eq. (3.78)} \]

\[ G_{1} = \frac{2.00 \cdot A_{12}}{\sin(\text{SIGDBL})} \]

\[ \text{eq. (3.79)} \]

\[ K_{11} = 0.500 \cdot (K_{S1} + G_{1}) \]

\[ K_{111} = 0.500 \cdot (K_{S1} - G_{1}) \]

\[ \text{CALCULATE } E_{11} \text{ AND } E_{111} \]

\[ \text{similar to description after eq. (3.29)} \]

\[ \text{CALL ROTATE}(E_{111x}, E_{111y}, E_{111z}, E_{111x}, E_{111y}, E_{111z}, RITAG, N_{fx}, N_{fy}, N_{fz}) \]

\[ \text{CALCULATE PRINCIPAL DIRECTIONS OF THE PINION CUTTER} \]

\[ \text{eq. (3.92)} \]

\[ \text{IF}(H_{G} \cdot \text{EQ. 'L'}) \text{THEN} \]

\[ E_{PIx} = \text{SN}_D1 \cdot \text{ST}_AUP \]
\[ E_{PIy} = \text{CST}_AUP \]
\[ E_{PIz} = \text{CS}_D1 \cdot \text{ST}_AUP \]

\[ \text{ELSE} \]
\[ E_{PIx} = -\text{SN}_D1 \cdot \text{ST}_AUP \]
\[ E_{PIy} = -\text{CST}_AUP \]
\[ E_{PIz} = -\text{CS}_D1 \cdot \text{ST}_AUP \]

\[ \text{END IF} \]

\[ \text{IF}(\text{DACOS}(E_{GIx} \cdot E_{PIx} + E_{GIy} \cdot E_{PIy} + E_{GIz} \cdot E_{PIz}) / \text{CNST} \cdot \text{GT.} \ 45.00) \]

\[ \text{THEN} \]

\[ E_{PIx} = -E_{PIx} \]
\[ E_{PIy} = -E_{PIy} \]
\[ E_{PIz} = -E_{PIz} \]

\[ \text{END IF} \]

\[ \text{CALL ROTATE}(E_{II1x}, E_{II1y}, E_{II1z}, E_{II1x}, E_{II1y}, E_{II1z}, RITAG, N_{fx}, N_{fy}, N_{fz}) \]

\[ \text{CALCULATE THE ANGLE BETWEEN PRINCIPAL DIRECTIONS OF PINION AND CUTTER} \]
* cross product of eli and epi
  SNSIGM=(ELIfy*EPIfz-ELIfz*EPIfy)/Nfx
* dot product of eli and epi
  CSSIGM=ELIfx*EPIfx+ELIfy*EPIfy+ELIfz*EPIfz
  CS2SIG=2.D00*CSSIGM*CSSIGM-1.D00
  TN2SIG=2.D00*SNSIGM*CSSIGM/CS2SIG

* CALCULATE PRINCIPAL CURVATURES OF PINION CUTTER

* eq. (2.12)
  KPII=0.D00
* eq. (3.94)
  KPI=KII*KIII/(KII*SNSIGM*SNSIGM+KIII_CSSIGM*CSSIGM)

* CALCULATE A11, A12, AND A22

* eq. (A.31)
  A11=KPI-KII*CSSIGM*CSSIGM-KIII*SNSIGM*SNSIGM
* eq. (A.32)
  A12=(KII-KIII)*SNSIGM*CSSIGM
* eq. (A.34)
  A22=KPII-KII*SNSIGM*SNSIGM-KIII*CSSIGM*CSSIGM

* CALCULATE UP

* eq. (3.95)
  UP=1.D00/(KPI*SNPSIP/CSPSIP)

* CALCULATE RP

* eq. (3.99)
  Bmx=-Bfx*CSD1+Bfz*SND1
* eq. (3.100)
  RP=(Bmx+UP*CSPSIP)*SNPSIP/CSPSIP

* CALCULATE MPI

* eq. (3.107)
  C11=(Nfy*EPIfz-Nfz*EPIfy)*CSD1+(Nfy*EPIfx-Nfx*EPIfy)*SND1
  C12=(Nfz*EPIfz-Nfy*EPIfy)*SNPIT1+(Nfy*EPIfx-Nfx*EPIfy)*CSPIT1
* eq. (3.108)
  C22=-(Nfy*EPIIfz-Nfz*EPIIfy)*SNPIT1+(Nfy*EPIIfx-Nfx*EPIIfy)*CSPIT1
  IF(HG .EQ. 'R')THEN
    C11=-C11
    C12=-C12
    C22=-C22
  END IF
* eq. (3.119)
  T4=(Bfy*CSR1)/(EPIIfx*CSD1-EPIIfz*SND1)
  IF(HG .EQ. 'R')THEN
    T4=-T4
  END IF
c eq. (3.120)
T1=-C11/KPI
T2=(A11*KPI*I*T4+A11*C22-A12*C12)/(A12*KPI)
c eq. (3.122)
U11=T1*EP1fx
U12=T2*EP1fx+T4*EP1fx
U21=T1*EP1fy
U22=T2*EP1fy+T4*EP1fy
U31=T1*EP1fz
U32=T2*EP1fz+T4*EP1fz
c eq. (3.124)
V1=U21*Nfz*CSD1+U21*Nfx*SND1-Nfy*(U11*SND1+U31*CSD1)
c eq. (3.125)
V2=(U22*CSD1-U12*SNPIT1)*Nfz-(U11*CSPIT1+U12*SND1+U32*CSD1-U31
*SNPIT1)*Nfy+(U21*CSPIT1+U22*SND1)*Nfx
IF(HG.EQ. 'R') THEN
V1=-V1
V2=-V2
V3=-V3
END IF
c eq. (3.132)
H11=-U21*CSPIT1+SND1*(Bfz*SNPIT1-Bfx*CSPIT1)
c eq. (3.134)
H21=U11*CSPIT1-U31*SNPIT1+Bfy*SNRTI
c eq. (3.136)
H31=U21*SNPIT1+CSD1*(Bfz*SNPIT1-Bfx*CSPIT1)
c eq. (3.133)
H12=(Bfz*SNPIT1-Bfx*CSPIT1-U22)*CSPIT1
c eq. (3.135)
H22=-Bfy-U12*CSPIT1+U32*SNPIT1
c eq. (3.137)
H32=-(Bfz*SNPIT1-Bfx*CSPIT1-U22)*SNPIT1
IF(HG.EQ. 'R') THEN
H11=U21*CSPIT1+SND1*(Bfz*SNPIT1-Bfx*CSPIT1)
H21=U11*CSPIT1-U31*SNPIT1+Bfy*SNRTI
H31=U21*SNPIT1+CSD1*(Bfz*SNPIT1-Bfx*CSPIT1)
H12=(Bfz*SNPIT1-Bfx*CSPIT1-U22)*CSPIT1
H22=-Bfy-U12*CSPIT1+U32*SNPIT1
H32=-(Bfz*SNPIT1-Bfx*CSPIT1-U22)*SNPIT1
END IF
c eq. (3.139)
F1=Nfx*H11+Nfy*H21+Nfz*H31
c eq. (3.140)
F2=Nfx*H12+Nfy*H22+Nfz*H32
c eq. (3.145)
Y2=A12*(2.00*KPI*I*T2-V2-F1)
Y3=A12*(KPI*I*T2+KPI*I*T4-V3-F2)-(KPI*T2+C12)*(KPI*I*T4+C22)
MP1=Y3/Y2
* CALCULATE EM AND LM
VTIPfx = U11*MP1 + U12
VTIPfy = U21*MP1 + U22
VTIPfz = U31*MP1 + U32

IF (HG .EQ. 'L') THEN
    EM = (Bfy*CSPIT1 - VTIPfx)/(MP1*SNDI) + Bfy
    LM = (Bfx*CSPIT1 - Bfz*SNPIT1 + VTIPfy)/MP1 + Bfx*SNDI + Bfz*CSDI
ELSE
    EM = (-Bfy*CSPIT1 - VTIPfx)/(MP1*SNDI) - Bfy
    LM = (Bfx*CSPIT1 - Bfz*SNPIT1 - VTIPfy)/MP1 + Bfx*SNDI + Bfz*CSDI
END IF

* CALCULATE SP AND QP
*

ZI = -Bfy + EM - UP*SNPSIP*SNTAUP
Z2 = Bfx*SNDI + Bfz*CSDI - LM - UP*SNPSIP*CSTAUP
SP = DSQRT(Z1*Z1 + Z2*Z2)
QP = DATAN(Z1/Z2)
IF (HG .EQ. 'L') THEN
    THETAP = TAUP - QP
ELSE
    THETAP = TAUP + QP
END IF

* CONVERT RADIUS TO DEGREE
*
PSIGDG = PSIG/CNST
PSIPDG = PSIP/CNST
TAUDG = TAUG/CNST
TAUPDG = TAUP/CNST
QGDG = QG/CNST
QPDG = QP/CNST
THEGDG = THEG/CNST
THEPDG = THEP/CNST
PHIGDG = PHIGO/CNST
*
* OUTPUT
*
WRITE (72, 10000) PSIGDG, PSIPDG, RG, RP, TAUDG, TAUPDG, SG, SP, QGDG, QPDG,
   MG2, MP1, EM, LM, UC, UP, THEGDG, THEPDG, PHIGDG
IF (I .EQ. 1) THEN
TPAR(1) = RG * CSPSIG / SNPSIG * CSPSIG
TPAR(2) = (MG2 - SNRT2) * CSPSIG
TPAR(3) = CSRT2 * SNPSIG
TPAR(4) = RG * CSPSIG / SNPSIG
TPAR(5) = CSD2 * SNPSIG
TPAR(6) = SND2 * CSPSIG
TPAR(7) = SND2 * SNPSIG
TPAR(8) = CSD2 * CSPSIG
TPAR(9) = RP * CSPSIG / SNPSIG * CSPSIG
TPAR(10) = (MPI - SNRT1) * CSPSIG
TPAR(11) = CSRT1 * SNPSIG
TPAR(12) = SNRT1 * CSPSIG
TPAR(13) = RP * CSPSIG / SNPSIG
TPAR(14) = CSD1 * SNPSIG
TPAR(15) = SND1 * CSPSIG
TPAR(16) = SND1 * SNPSIG
TPAR(17) = CSD1 * CSPSIG
TPAR(18) = LM * SND1
TPAR(19) = LM * CSD1

PHIP = 0.D00
PHI21 = 0.D00
PHI11 = 0.D00
CSPH11 = DCOS(PHI11)
SNPH11 = DSIN(PHI11)

TX(1) = PHIP
TX(2) = THETAP
TX(3) = PHI21
TX(4) = PHIG0
TX(5) = THETAG
CALL NONLIN(TCN, 14, 5, 100, TX, TF, TF1, 1.D-5, AZS, IPVT, WORK)

PHIPO = TX(1)
THEPO = TX(2)
PHI210 = TX(3)
PHIG0 = TX(4)
THEG0 = TX(5)

TX(1) = PHIPO
TX(2) = THEPO
TX(3) = PHI210
TX(4) = PHIG0
TX(5) = THEG0
D1H11 = 18.D00 / 36.D00 * CNST

DO 100 IJ = 1, 60
CSPH11=DCOS(PHI11)
SNPH11=DSIN(PHI11)
CALL NONLIN(TCN,14,5,100,TX,TF,TF1,1.D-5,AZS,IPVT,WORK)
PHIP=TX(1)
THETAP=TX(2)
PHI21=TX(3)
PHIG=TX(4)
THETAG=TX(5)
*
CALL PRING2(KS2,G2,E2Ifx,E2Ify,E2Iffz,E2IIfx,E2IIfz)
CALL PRINPl(KS1,G1,EIfx,Elfy,Elffz,ElIfx,ElIffz)
CALL SIGAN2(E2Ifx,E2Ify,E2Iffz,E2IIfx,E2IIfz,ElIfx,ElIffz)
CALL EULER(KS2,G2,KSI,G1,CS2SIG,SN2SIG,IEU)
IF(IEU .EQ. I) THEN
   WRITE(72,*) 'THERE IS INTERFERENCE'
   GO TO 88888
END IF
*
CALL ELLIPS(KS2,G2,KSI,G1,CS2SIG,SN2SIG,DEF,ALFAI,
   AXISSL,AXISS,ElIfx,ElIffz,CS2SIG,SN2SIG,IEU)
*

* XBf, YBf, and ZBf is the direction of the long axis of the ellipse
* CALL PF(XBp,YBp,ZBp, XBf,YBf,ZBf)
ELB1px=B2px+XBp
ELB1pz=B2pz+ZBp
ELB2px=B2px-XBp
ELB2pz=B2pz-ZBp
*
IF(I .EQ. 1) THEN
   WRITE(79,9000) IJ,PHI11/CNST,IJ,ERROR
   WRITE(78,8000) IJ,B2pz,IJ,B2px
   WRITE(77,7000)ELB1pz,ELB1px,ELB2pz,ELB2px
ELSE
   WRITE(89,9000) IJ,PHI11/CNST,IJ,ERROR
   WRITE(88,8000) IJ,B2pz,IJ,B2px
   WRITE(87,7000)ELB1pz,ELB1px,ELB2pz,ELB2px
END IF
*
PHI11=PHI11+DIHI11
*
100 CONTINUE
*
*
PHI11=0.D00
CSPH11=DCOS(PHI11)
SNPH11=DSIN(PHI11)
\[ TX(1) = \text{PHI0} \]
\[ TX(2) = \text{THE0} \]
\[ TX(3) = \text{PHI210} \]
\[ TX(4) = \text{PHIG0} \]
\[ TX(5) = \text{THEG0} \]
\[ \text{D1HI11} = 18 \cdot 360 / 36.00 \cdot \text{CNST} \]

\[ \text{DO 200 IJ} = 1,60 \]
\[ \text{CSPHI11} = \text{DCOS} \text{(PHI11)} \]
\[ \text{SNPHI11} = \text{DSIN} \text{(PHI11)} \]
\[ \text{CALL NONLIN(TCN,14,5,100,TX,TF,TF1,1.D-5,AZS,IPVT,WORK)} \]
\[ \text{PHIP} = TX(1) \]
\[ \text{THETAP} = TX(2) \]
\[ \text{PHI21} = TX(3) \]
\[ \text{PHIG} = TX(4) \]
\[ \text{THETAG} = TX(5) \]
\[ \text{ERROR} = ((\text{PHI21} \cdot 36.02 - \text{PHI210} \cdot 36.02) - \text{PHI11} \cdot 36.02 \cdot \text{TNI/TN2}) / \text{CNST} \]

\[ \text{CALL PRING2(KS2,G2,E2Ifx,E2Ify,E2Ifz,E2IIfx,E2IIfy,E2IIfz)} \]
\[ \text{CALL PRINP1(KS1,G1,E1Ifx,E1Ify,E1Ifz,E1IIfx,E1IIfy,E1IIfz)} \]
\[ \text{CALL SIGAN2(E2Ifx,E2Ify,E2Ifz,E2IIfx,E2IIfy,E2IIfz,E1Ifx,E1Ify,} \]
\[ \text{E1IIfx,E1IIfy,E1IIfz,CS2SIG,SN2SIG,SI} \]
\[ \text{M12)} \]

\[ \text{CALL EULER(KS2,G2,KSI,GI,CS2SIG,SN2SIG,IEU)} \]
\[ \text{IF(IEU .EQ. 1) THEN} \]
\[ \text{WRITE(72,*)'THERE IS INTERFERENCE'} \]
\[ \text{GO TO 88888} \]
\[ \text{END IF} \]

\[ \text{CALL ELLIPS(KS2,G2,KSI,GI,CS2SIG,SN2SIG,DEF,ALFAI,} \]
\[ \text{AXISL,AXISS,E1Ifx,E1Ify,E1IIfz)} \]

\[ \text{CALL PF(B2px,B2py,B2pz,B2fx,B2fy,B2fz)} \]

\[ \text{XBF}, \text{YBF}, \text{and ZBF is the direction of the long axis of the ellipse} \]

\[ \text{CALL PF(XBp,YBp,ZBp,XBF,YBF,ZBF)} \]
\[ \text{ELB1px} = \text{B2px} + \text{XBp} \]
\[ \text{ELB1pz} = \text{B2pz} + \text{ZBp} \]
\[ \text{ELB2px} = \text{B2px} - \text{XBp} \]
\[ \text{ELB2pz} = \text{B2pz} - \text{ZBp} \]

\[ \text{IF(I .EQ. 1) THEN} \]
\[ \text{WRITE(79,9001)IJ,PHI11/CNST,IJ,ERROR} \]
\[ \text{WRITE(78,8001)IJ,B2pz,IJ,B2px} \]
\[ \text{WRITE(77,7000)ELB1px,ELB1pz,ELB2px,ELB2pz} \]
\[ \text{ELSE} \]
\[ \text{WRITE(89,9001)IJ,PHI11/CNST,IJ,ERROR} \]
\[ \text{WRITE(88,8001)IJ,B2pz,IJ,B2px} \]
\[ \text{WRITE(87,7000)ELB1px,ELB1pz,ELB2px,ELB2pz} \]
\[ \text{END IF} \]

\[ \text{PHI11} = \text{PHI11-D1HI11} \]
* FOR THE DETERMINATION OF MEAN CONTACT POINT
*
SUBROUTINE PCN(X,F,NE)
IMPLICIT REAL*8(A-H,K,M-Z)
CHARACTER*8 HG
INTEGER NE
REAL*8 X(NE),F(NE),PAR(6)
COMMON/P1/PAR
COMMON/A0/HG
COMMON/A1/p11,p12,p13,p21,p22,p23,p31,p32,p33,p1,p2,p3
COMMON/A2/SG,CSR2,QTG,SNPSIG,CPSIG
COMMON/A3/ITG1,ITG2,RTG
THETAG=X(1)
CSTHEG=DCOS(THETAG)
SNHTHEG=DSIN(THETAG)
IF(HG .EQ. 'L') THEN
UG=PAR(1)+SG*(PAR(2)*SNHTHEG+PAR(3))/(CSR2*DSIN(THETAG+QTG))
ELSE
UG=PAR(1)+SG*(PAR(2)*SNHTHEG+PAR(3))/(CSR2*DSIN(THETAG+QTG))
END IF
Bcx=PAR(4)+UG*CSIG
Bcy=UG*SNPSIG*SNHTHEG
Bcz=UG*SNPSIG*CSTHEG
CALL TRCOPR(Bpx,Bpy,Bpz,
. p11,p12,p13,p21,p22,p23,p31,p32,p33,p1,p2,p3,
. Bcx,Bcy,Bcz)
XM=Bpz*(ITG1-ITG2)/2.D00
F(1)=Bpx-XM
END
*
* FOR THE DETERMINATION OF COORDINATES AND NORMALS OF CONTACT POINTS
*
SUBROUTINE TCN(TX,TF,NE)
IMPLICIT REAL*8(A-H,K,M-Z)
INTEGER NE
CHARACTER*8 HG
REAL*8 TX(NE),TF(NE),TPAR*(19),LM
COMMON/A0/HG
COMMON/T1/TPAR
COMMON/A2/SG,CSRT2,QG,SNPSIG,CSPSIG
COMMON/A4/CSD2,SND2,CSPIT2,SNPIT2
COMMON/B1/CSPHI11,SNPHI11,SP,EM,LM,CSRT1,CSD1,SND1,CSPSIG,SNPSIG
COMMON/B2/CSPIT1,SNPIT1,MP1,MP2,QP
COMMON/B3/B2fx,B2fy,B2fz
COMMON/B4/CSPH2,SNPH2,CSPH21,SNPH21
COMMON/C1/UG,CSTAUG,SNTAUG
COMMON/C2/N2fx,N2fy,N2fz
COMMON/D1/UP,CSTAU1,SNTAUG
COMMON/F1/PHIGO
COMMON/G1/DA,DV
PHIP=TX(1)
THETAP=TX(2)
PHI21=TX(3)
PHIG=TX(4)
THETAG=TX(5)
CSPHIP=DCOS(PHIP)
SNPHIP=DSIN(PHIP)
CSTHEP=DCOS(THETAP)
SNTHEP=DSIN(THETAP)
CSPH21=DCOS(PHI21)
SNPH21=DSIN(PHI21)
CSPHIG=DCOS(PHIG)
SNPHIG=DSIN(PHIG)
CSTHEG=DCOS(THETAG)
SNTHEG=DSIN(THETAG)
PHI2=(PHIG-PHIGO)/MG2
PHI1=PHIP/MP1
CSPH2=DCOS(PHI2)
SNPH2=DSIN(PHI2)
CSPH1=DCOS(PHI1)
SNPH1=DSIN(PHI1)
IF(HG .EQ. 'L') THEN
  TAUP=THETAP+QP-PHIP
ELSE
  TAUP=THETAP-QP+PHIP
END IF
CSTAU1=DCOS(TAUP)
SNTAUG=DSIN(TAUP)
IF(HG .EQ. 'L') THEN
  TAUG=THETAG-QG-PHIP
ELSE
  TAUG=THETAG+QG-PHIP
END IF
CSTAU1=DCOS(TAUG)
SNTAUG=DSIN(TAUG)
CSQPHP=DCOS(QP-PHIP)
SNQPHP=DSIN(QP-PHIP)
CSQPHG=DCOS(QG-PHIP)
SNQPHG=DSIN(QG-PHIP)
GEAR

SURFACE EQUATIONS

IF(HG .EQ. 'L') THEN
  UG = TPAR(1) - SG* (TPAR(2) * SNTHEG - SNQPHG*TPAR(3)) / (CSRT2*SNTAUG)
  B2py = UG*SNPSIG*SNTAUG - SG*SNQPHG
ELSE
  UG = TPAR(1) - SG* (TPAR(2) * SNTHEG + SNQPHG*TPAR(3)) / (CSRT2*SNTAUG)
  B2py = UG*SNPSIG*SNTAUG + SG*SNQPHG
END IF

B2px = CSD2*(TPAR(4) - UG*CSPSIG) - SND2*(UG*SNPSIG*CSTAUG + SG*CSQPHG)
B2pz = SND2*(TPAR(4) - UG*CSPSIG) + CSD2*(UG*SNPSIG*CSTAUG + SG*CSQPHG)
N2px = TPAR(5) - TPAR(6)*CSQPHG
N2py = CSPSIG*CSTAUG
N2pz = TPAR(7) + TPAR(8)*CSQPHG

[**Mwp**] = [**Mwa**] [**Map**]

IF(HG .EQ. 'L') THEN
  CALL COMBI(wp11, wp12, wp13, wp21, wp22, wp23, wp31, wp32, wp33, wp1, wp2, wp3,
      . CSPH2, SNPH2, 0.D00, -SNPH2, CSPH2, 0.D00, 0.D00, 1.D00,
      . 0.D00, 0.D00, 0.D00, 0.D00, 0.D00, 0.D00, -SNPH2, CSPH2, 0.D00, 0.D00,
      . CSPIT2, 0.D00, SNPIT2, 0.D00, 1.D00, 0.D00, -SNPIT2, 0.D00, CSPIT2,
      . 0.D00, 0.D00, 0.D00)
ELSE
  CALL COMBI(wp11, wp12, wp13, wp21, wp22, wp23, wp31, wp32, wp33, wp1, wp2, wp3,
      . CSPH2, -SNPH2, 0.D00, SNPH2, CSPH2, 0.D00, 0.D00, 1.D00,
      . 0.D00, 0.D00, 0.D00, 0.D00, 0.D00, 0.D00, -SNPH2, CSPH2, 0.D00, 0.D00,
      . CSPIT2, 0.D00, SNPIT2, 0.D00, 1.D00, 0.D00, -SNPIT2, 0.D00, CSPIT2,
      . 0.D00, 0.D00, 0.D00)
END IF
CALL TRCOOR(B2wx, B2wy, B2wz,
      . wp11, wp12, wp13, wp21, wp22, wp23, wp31, wp32, wp33, wp1, wp2, wp3,
CALL TRCOOR(N2wx, N2wy, N2wz,
      . wp11, wp12, wp13, wp21, wp22, wp23, wp31, wp32, wp33, 0.D00, 0.D00, 0.D00,
      . N2px, N2py, N2pz)

[**Mfw**] = [**Mfa**] [**Maw**]

fa1 = CSPIT2
fa12 = 0.D00
fa13 = -SNPIT2
fa21 = 0.D00
fa22 = 1.D00
fa23 = 0.D00
fa31 = SNPIT2

149
fa32=0.D00
fa33=CSPIT2
fa1=0.d00
fa2=0.d00
fa3=0.d00

IF(HG .EQ. 'L') THEN
    CALL COMBI(fw11,fw12,fw13,fw21,fw22,fw23,fw31,fw32,fw33,
                fw1,fw2,fw3,
                CSPIT2,0.D00,-SNPIT2,0.D00,1.D00,0.D00,SNPIT2,0.D00,CSPIT2,
                0.D00,0.D00,0.D00,
                CSPH21,-SNPH21,0.D00,SNPH21,CSPH21,0.D00,0.D00,0.D00,0.D00,1.D00,
                0.D00,0.D00,0.D00)
ELSE
    CALL COMBI(fw11,fw12,fw13,fw21,fw22,fw23,fw31,fw32,fw33,
                fw1,fw2,fw3,
                CSPIT2,0.D00,-SNPIT2,0.D00,1.D00,0.D00,SNPIT2,0.D00,CSPIT2,
                0.D00,0.D00,0.D00,
                CSPH21,SNPH21,0.D00,-SNPH21,CSPH21,0.D00,0.D00,0.D00,1.D00,
                0.D00,0.D00,0.D00)
END IF

CALL TRCOOR(B2fx,B2fy,B2fz,
            fw11,fw12,fw13,fw21,fw22,fw23,fw31,fw32,fw33,fwl,fw2,fw3,
            B2wx,B2wy,B2wz)
CALL TRCOOR(N2fx,N2fy,N2fz,
            fw11,fw12,fw13,fw21,fw22,fw23,fw31,fw32,fw33,0.D00,0.D00,0.D00,
            N2wx,N2wy,N2wz)

* * PINION
* SURFACE EQUATIONS *

IF(HG .EQ. 'L') THEN
    UP=TPAR(9) - (SP*(TPAR(10)*SNTHEP+SNQPHP*TPAR(11)) - EM*(TPAR(11) +
                TPAR(12)*CSTAUP) - LM*TPAR(12)*SNTAUP)/(CSRTI*SNTAUP)
    Blpy=UP*SNPSIP*SNTAUP+SP*SNQPHP-EM
ELSE
    UP=TPAR(9) - (SP*(TPAR(10)*SNTHEP-SNQPHP*TPAR(11)) + EM*(TPAR(11) +
                TPAR(12)*CSTAUP) - LM*TPAR(12)*SNTAUP)/(CSRTI*SNTAUP)
    Blpy=UP*SNPSIP*SNTAUP-SP*SNQPHP+EM
END IF

Blpx=CSDI*(TPAR(13)-UP*CSPSIP)-SNDI*(UP*SNPSIP*CSTAUP+SP*
            CSQPHP)-LM*SNDI
Blpz=SNDI*(TPAR(13)-UP*CSPSIP)+CSDI*(UP*SNPSIP*CSTAUP+SP*
            CSQPHP)+LM*CSDI

Nlpx=-(TPAR(14)-TPAR(15)*CSTAUP)
Nlpy=-CSPSIP*CSTAUP
Nlpz=-(TPAR(16)+TPAR(17)*CSTAUP)

* [Mwp]=[Mwa][Map]

IF(HG .EQ. 'L') THEN
    CALL COMBI(wp11,wp12,wp13,wp21,wp22,wp23,wp31,wp32,wp33,
                wp1,wp2,wp3,
ELSE
CALL COMBI(wp11, wp12, wp13, wp21, wp22, wp23, wp31, wp32, wp33,
wp1, wp2, wp3,
CSPH1, SNPH1, 0.D00, -SNPH1, CSPH1, 0.D00, 0.D00, 1.D00,
0.D00, 0.D00, 0.D00,
CSPIT1, 0.D00, SNPIT1, 0.D00, 0.D00, 1.D00, 0.D00, -SNPIT1, 0.D00, CSPIT1,
0.D00, 0.D00, 0.D00)
END IF
CALL TRCOOR(Blwx, Blwy, Blwz,
wp11, wp12, wp13, wp21, wp22, wp23, wp31, wp32, wp33,
Blpx, Blpy, Blpz)
CALL TRCOOR(Nlwx, Nlwy, Nlwz,
wp11, wp12, wp13, wp21, wp22, wp23, wp31, wp32, wp33, 0.D00, 0.D00, 0.D00,
Nlpx, Nlpy, Nlpz)
*
*******************************************************************************
*
* [Mpw] = [Mpa] [Maw]
*
*
IF(HG .EQ. 'L') THEN
CALL COMBI(pw11, pw12, pw13, pw21, pw22, pw23, pw31, pw32, pw33,
pw1, pw2, pw3,
CSPIT1, 0.D00, -SNPIT1, 0.D00, 1.D00, 0.D00, SNPIT1, 0.D00, CSPIT1,
0.D00, 0.D00, 0.D00,
CSPHII, SNPHII, 0.D00, -SNPHII, CSPHII, 0.D00, 0.D00, 0.D00, 1.D00,
0.D00, 0.D00, 0.D00)
ELSE
CALL COMBI(pw11, pw12, pw13, pw21, pw22, pw23, pw31, pw32, pw33,
pw1, pw2, pw3,
CSPIT1, 0.D00, -SNPIT1, 0.D00, 1.D00, 0.D00, SNPIT1, 0.D00, CSPIT1,
0.D00, 0.D00, 0.D00,
CSPHII, -SNPHII, 0.D00, SNPHII, CSPHII, 0.D00, 0.D00, 0.D00, 1.D00,
0.D00, 0.D00, 0.D00)
END IF
CALL TRCOOR(Blpx, Blpy, Blpz,
pw11, pw12, pw13, pw21, pw22, pw23, pw31, pw32, pw33, 0.D00, 0.D00, 0.D00,
Blwx, Blwy, Blwz)
CALL TRCOOR(Nlpx, Nlpy, Nlpz,
pw11, pw12, pw13, pw21, pw22, pw23, pw31, pw32, pw33, 0.D00, 0.D00, 0.D00,
Nlwx, Nlwy, Nlwz)
Blfx=-Blpx+DA*SNPIT1
Blfy=-Blpy+DV
Blfz=Blpz+DA*CSPIT1
Nlfx=-Nlpx
Nlfy=-Nlpy
Nlfz=Nlpz
TF(1)=B2fx-Blfx
TF(2)=B2fy-Blfy
TF(3)=B2fz-Blfz
151
TF(4) = N2fx - N1fx
TF(5) = N2fy - N1fy
END

* * FOR THE DETERMINATION OF GEAR PRINCIPAL CURVATURES AND DIRECTIONS *

SUBROUTINE PRING2(KS2,G2,E2Ifx,E2Ify,E2Ifz,E2IIfx,E2IIfy,E2IIfz)
IMPLICIT REAL*8(A-H,K,M-Z)
CHARACTER*8 HG
COMMON/A0/HG
COMMON/A2/SG,CSRT2,QG,SNPSIG,CPSIG
COMMON/A3/TND1,TND2,RITAG
COMMON/A4/CSD2,SNP2,CPIT2,SNPIT2
COMMON/B2/CPIT1,SNPIT1,MP1,MP2,QQ
COMMON/B3/Bfx,Bfy,Bfz
COMMON/B4/CSHP2,SNPH2,CSHP21,SNPH21
COMMON/C1/UG,STAUG,STNUG
COMMON/C2/Nfx,Nfy,Nfz
KGI=-CPSIG/(UG*SNPSIG)
KGII=0.D00
EGIfx=SND2*SNTAUG
EGIfy=CSAUG
EGIfz=-CSD2*SNTAUG
CALL ROTATE(EGIIfx,EGIIfy,EGIIfz,EGIfx,EGIfy,EGIIfz,RITAG,
. Nfx,Nfy,Nfz)

* * CALCULATE W2G *
* *
IF (HG .EQ. 'L') THEN
W2fx=-SNPIT2
WGfx=-MG2*CSD2
W2fy=0.D00
WGfy=0.D00
W2fz=CPIT2
WGfz=-MG2*SND2
ELSE
W2fx=SNPIT2
WGfx=MG2*CSD2
W2fy=0.D00
WGfy=0.D00
W2fz=CPIT2
WGfz=MG2*SND2
END IF

W2Gfx=W2fx-WGfx
W2Gfy=W2fy-WGfy
W2Gfz=W2fz-WGfz

* * CALCULATE VT2, VTG, AND VT2G *
*
CALL CROSS(VT2fx,VT2fy,VT2fz,W2fx,W2fy,W2fz,Bfx,Bfy,Bfz)
CALL CROSS(VTGfx,VTGfy,VTGfz,WGfx,WGfy,WGfz,Bfx,Bfy,Bfz)
VT2Gfx=VT2fx-VTGfx
VT2Gfy=VT2fy-VTGfy
VT2Gfz=VT2fz-VTGfz
*
* CALCULATE V(2G)GI AND V(2G)GII
* CALL DOT(VGI,EGIfx,EGIfy,EGIfz,VT2Gfx,VT2Gfy,VT2Gfz)
CALL DOT(VGII,EGIIfx,EGIIfy,EGIIfz,VT2Gfx,VT2Gfy,VT2Gfz)
* CALCULATE A13,A23,A33
* CALL DET(DETI,W2Gfx,W2Gfy,W2Gfz,Nfx,Nfy,Nfz,EGIfx,EGIfy,EGIfz)
A13=-KGI*VGI-DETI
CALL DET(DETII,W2Gfx,W2Gfy,W2Gfz,Nfx,Nfy,Nfz,EGIIfx,EGIIfy,EGIIfz)
A23=-KGII*VGII-DETII
CALL DET(DET3,Nfx,Nfy,Nfz,W2Gfx,W2Gfy,W2Gfz,VT2Gfx,VT2Gfy,VT2Gfz)
CALL CROSS(Cx,Cy,Cz,W2fx,W2fy,W2fz,VTGfx,VTGfy,VTGfz)
CALL CROSS(Dx,Dy,Dz,WGfx,WGfy,WGfz,VT2fx,VT2fy,VT2fz)
CALL DOT(DET45,Nfx,Nfy,Nfz,Cx-Dx,Cy-Dy,Cz-Dz)
A33=KGI*VGI*VGI+KGII*VGII*VGII-DET3-DET45
*
* CALCULATE SIGMA
* P=A23*A23-AI3*AI3+(KGI-KGII)*A33
SIGDBL=DATAN(2.D00*A13*A23/P)
SIGMA=0.5D00*SIGDBL
* CALCULATE K21 AND K211
* T1=KGI+KGII-(A13*A13+A23*A23)/A33
K2I=(T1+T2)/2.D00
K2II=(T2-T1)/2.D00
* CALCULATE E2I AND E2II
* CALL ROTATE(E2Ifx,E2Ify,E2Ifz,EGIfx,EGIfy,EGIfz,-SIGMA,Nfx,Nfy,
. Nfz)
CALL ROTATE(E2IIfx,E2IIfy,E2IIfz,E2Ifx,E2Ify,E2Ifz,RITAG,
. Nfx,Nfy,Nfz)
END
* FOR THE DETERMINATION OF PINION PRINCIPAL CURVATURES AND DIRECTIONS
* SUBROUTINE PRINPI(KSI,G1,ELIfx,ELIfy,ELIfz,E11Ifx,E11Ify,E11Ifz)
IMPLICIT REAL*8(A-H,K,M-Z)
REAL*8 TPAR(19),LM
CHARACTER*8 HG
COMMON/T1/TPAR
COMMON/A0/HG
COMMON/A3/TND1,TND2,RITAG
COMMON/B1/CSPH11,SNPH11,SP,EM,LM,CSRT1,CSD1,SND1,CSPSIP,SNPSIP
153
COMMON/B2/CSPIT1,SNPIT1,MP1,MP2,QP
COMMON/B3/Bfx,Bfy,Bfz
COMMON/C2/Nfx,Nfy,Nfz
COMMON/D1/UP,CSTAUP,SNTAUP
KPI=CSPSIP/(UP*SNPSIP)
KPII=0.00
EPIfx=SNDI*SNTAUP
EPIfy=CSTAUP
EPIfz=CSL1*SNTAUP
CALL ROTATE(EPIIfx,EPIIfy,EPIIfz,EPIfx,EPIfy,EPIfz,RITAG,
. Nfx,Nfy,Nfz)
*
* CALCULATE WIP
*
IF(HG .EQ. 'L')THEN
   Wlfx=-SNPIT1
   WPfx=-MP1*CSDL
   Wlfy=0.00
   WPfy=0.00
   Wlfz=-CSPIT1
   WPfz=MP1*SND1
ELSE
   Wlfx=SNPIT1
   WPfx=MP1*CSDL
   Wlfy=0.00
   WPfy=0.00
   Wlfz=CSPIT1
   WPfz=-MP1*SND1
END IF
WIPfx=Wlfx-WPfx
WIPfy=Wlfy-WPfy
WIPfz=Wlfz-WPfz
*
* CALCULATE VT2, VTG, AND VT2G
*
*
CALL CROSS(VTIfx,VTIfy,VTIfz,Wlfx,Wlfy,Wlfz,Bfx,Bfy,Bfz)
CALL CROSS(VTPIfx,VTPIfy,VTPIfz,WPfx,WPfy,WPfz,Bfx,Bfy,Bfz)
IF(HG .EQ. 'L')THEN
   CALL CROSS(VTP2fx,VTP2fy,VTP2fz,TPAR(18),EM,TPAR(19),
. WPfx,WPfy,WPfz)
ELSE
   CALL CROSS(VTP2fx,VTP2fy,VTP2fz,TPAR(18),-EM,TPAR(19),
. WPfx,WPfy,WPfz)
END IF
VTPfx=VTP1fx+VTP2fx
VTPfy=VTP1fy+VTP2fy
VTPfz=VTP1fz+VTP2fz
VTIfx=VTIfx-VTPfx
VTIfy=VTIfy-VTPfy
VTIfz=VTIfz-VTPfz
CALL DOT(VPI,EPIfx,EPIfy,EPIfz,VTIfx,VTIfy,VTIfz)
CALL DOT(VPII,EPIIfx,EPIIfy,EPIIifz,VTIfx,VTIfy,VTIfz)
* CALCULATE A13, A23, A33
* 
* CALL DET(DETI, WIPfx, WIPfy, WIPfz, Nfx, Nfy, Nfz, EPIfx, EPIfy, EPIfz)
A13 = -KPI*VPI - DETI
CALL DET(DETII, WIPfx, WIPfy, WIPfz, Nfx, Nfy, Nfz,
. EPIIfx, EPIIfy, EPIIfz)
A23 = -KPII*VPII - DETII
CALL DET(DET3, Nfx, Nfy, Nfz, WIPfx, WIPfy, WIPfz,
. VT1Pfx, VT1Pfy, VT1Pfz)
CALL CROSS(Cx, Cy, Cz, W1fx, 0.000, W1fz, VTPfx, VTPfy, VTPfz)
CALL CROSS(Dx, Dy, Dz, WPfx, 0.000, WPfz, VTIfx, VTIfy, VTIfz)
CALL DOT(DET45, Nfx, Nfy, Nfz, Cx-Dx, Cy-Dy, Cz-Dz)
A33 = KPI*VPI + KPII*VPII - DET3 - DET45

* CALCULATE SIGMA
*
* P = A23*A23 - A13*A13 + (KPI - KPII)*A33
SIGDBL = DATAN(2.000*A13*A23/P)
SIGMA = 0.5000*SIGDBL

* CALCULATE KII AND KIII
*
* G1 = P / (A33*DCOS(SIGDBL))
KS1 = KPI + KPII - (A13*A13 + A23*A23)/A33
KII = (KS1 + G1)/2.000
KIII = (KS1 - G1)/2.000

* CALCULATE E1I AND E1II
*
* CALL ROTATE(E1Ifx, E1Ify, E1Ifz, EPIfx, EPIfy, EPIfz, -SIGMA, Nfx, Nfy,
. Nfz)
CALL ROTATE(E1IIIfx, E1IIIfy, E1IIIfz, E1Ifx, E1Ify, E1Ifz, RITAG,
. Nfx, Nfy, Nfz)
END

* FOR THE DETERMINATION OF THE ANGLE BETWEEN GEAR PRINCIPAL DIRECTIONS
* AND PINION PRINCIPAL DIRECTIONS
*
* SUBROUTINE SIGAN2(E21fx, E21fy, E21fz, E21Ifx, E21Ify, E21Ifz, CS2SIG, SN2SIG, SIGM12)
IMPLICIT REAL*8(A-H, K-M-Z)
CALL DOT(CSSIG, E1Ifx, E1Ify, E1Ifz, E21Ifx, E21Ify, E21Ifz)
CALL DOT(SNSIG, E1Ifx, E1Ify, E1Ifz, -E21Ifx, -E21Ify, -E21Ifz)
SIGM2 = 4.000*DATAN(SNSIG/(1.000 + CSSIG))
SIGM12 = 0.5000*SIGM2
CS2SIG = DCOS(SIGM2)
SN2SIG = DSIN(SIGM2)
END

* FOR THE DETERMINATION OF CONTACT ELLIPS
*
* SUBROUTINE ELLIPS(KS2, G2, KS1, G1, CS2SIG, SN2SIG, DEF, ALFAI,
AXISL, AXISS, EI fx, EI fy, EI fz

IMPLICIT REAL*8 (A-H, K-M, Z)
COMMON/A3/TND1, TND2, RITAG
COMMON/C2/Nfx, Nfy, Nfz
COMMON/E1/XBf, YBf, ZBf
D=DSQRT (GI *GI-2. DOO*G1 *G2*CS2SIG+G2*G2)
CS2AF1=(G1-G2*CS2SIG)/D
SN2AF1=G2*SN2SIG/D
ALFA1=DATAN (SN2AF1/(1. DOO+CS2AF1))
A= .25D00*DABS (KS1-KS2-D)
B= .25D00*DABS (KS1-KS2+D)
IF(KS2 LT. KS1) THEN
AXISL=DSQRT (DEF/A)
AXISS=DSQRT (DEF/B)
CALL ROTATE (XBf, YBf, ZBf, EI fx, EI fy, EI fz, RITAG-ALFA1, Nfx, .
. Nfy, Nfz)
ELSE
AXISL=DSQRT (DEF/B)
AXISS=DSQRT (DEF/A)
CALL ROTATE (XBf, YBf, ZBf, EI fx, EI fy, EI fz, -ALFA1, Nfx, Nfy, .
. Nfz)
END IF
XBf=AXISL*XBf
YBf=AXISL*YBf
ZBf=AXISL*ZBf
END

COORDINATE TRANSFORMATION FOR F TO P

SUBROUTINE PF(B2px, B2py, B2pz, Bfx, Bfy, Bfz)
IMPLICIT REAL*8 (A-H, K-M, Z)
COMMON/A4/CSD2, SND2, CSPH2, SNPIT2
COMMON/B4/CSPH2, SNPH2, CSPH21, SNPH21

[Mwf] = [Mwa] [Maf]

CALL COMBI(w11, w12, w13, w21, w22, w23, w31, w32, w33, w1, w2, w3, .
. CSPH21, SNPH21, 0. DOO, -SNPH21, CSPH21, 0. DOO, 0. DOO, 0. DOO, 1. DOO, .
. 0. DOO, 0. DOO, 0. DOO,
. CSPIT2, 0. DOO, SNPIT2, 0. DOO, 1. DOO, 0. DOO, 0. DOO, -SNPIT2, 0. DOO, CSPIT2, .
. 0. DOO, 0. DOO, 0. DOO)
CALL TRCOORD(B2wx, B2wy, B2wz, .
. w11, w12, w13, w21, w22, w23, w31, w32, w33, w1, w2, w3, .
. Bfx, Bfy, Bfz)

[Mpw] = [Mpa] [Maw]

CALL COMBI(p11, p12, p13, p21, p22, p23, p31, p32, p33, p1, p2, p3, .
. CSPIT2, 0. DOO, -SNPIT2, 0. DOO, 1. DOO, 0. DOO, SNPIT2, 0. DOO, CSPIT2, .
. 0. DOO, 0. DOO, 0. DOO,
. CSPH2, -SNPH2, 0. DOO, SNPH2, CSPH2, 0. DOO, 0. DOO, 0. DOO, 1. DOO, .
. 0. DOO, 0. DOO, 0. DOO)
CALL TRCOORD(B2px, B2py, B2pz,
SUBROUTINE EULER(KS2,G2,KSI,GI,CS2SIG,SN2SIG,IEU)
IMPLICIT REAL*8(A-H,K,M-Z)
T=KS2-KSI
U=DSQRT((G2-GI*CS2SIG)**2+(GI*SN2SIG)**2)
KR1=(T+U)/2.DO0
KR2=(T-U)/2.DO0
IF(KR1*KR2 .LT. 0.DO0) THEN
  IEU=1
ELSE
  IEU=0
END IF
END

SUBROUTINE DET(S,A,B,C,D,E,F,G,H,P)
IMPLICIT REAL*8(A-H,K,M-Z)
S=A*E*P+D*H*C+G*B*F-A*H*F-D*B*P-G*E*C
RETURN
END

SUBROUTINE TRCOOR(XN,YN,ZN,RII,RI2,RI3,R21,R22,R23,R31,R32,R33,
                    TI,T2,T3,XP,YP,ZP)
IMPLICIT REAL*8(A-H,O-Z)
XN=RII*XP+RI2*YP+RI3*ZP+TI
YN=R21*XP+R22*YP+R23*ZP+T2
ZN=R31*XP+R32*YP+R33*ZP+T3
RETURN
END

SUBROUTINE COMBI(C11,C12,C13,C21,C22,C23,C31,C32,C33,CI,C2,C3,
                    A11,A12,A13,A21,A22,A23,A31,A32,A33,A1,A2,A3,
                    B11,B12,B13,B21,B22,B23,B31,B32,B33,B1,B2,B3)
IMPLICIT REAL*8(A-H,M,N,O-Z)
C11=B31*A13+B21*A12+B11*A11
C12=B32*A13+B22*A12+B12*A11
C13=B33*A13+B23*A12+B13*A11
C21=B31*A23+B21*A22+B11*A21
C22=B32*A23+B22*A22+B12*A21
C23=B33*A23+B23*A22+B13*A21
C31=B31*A33+B21*A32+B11*A31
C32=B32*A33+B22*A32+B12*A31
C33=B33*A33+B23*A32+B13*A31
C1 = B3*A13 + B2*A12 + B1*A11 + A1
C3 = B3*A33 + B2*A32 + B1*A31 + A3
RETURN
END

* DOT OF TWO VECTORS *

SUBROUTINE DOT(V, X1, Y1, Z1, X2, Y2, Z2)
IMPLICIT REAL*8(A-H, O-Z)
V = X1*X2 + Y1*Y2 + Z1*Z2
RETURN
END

* CROSS OF TWO VECTORS *

SUBROUTINE CROSS(X, Y, Z, A, B, C, D, E, F)
IMPLICIT REAL*8(A-H, O-Z)
X = B*F - C*E
Y = C*D - A*F
Z = A*E - B*D
RETURN
END

* ROTATION A VECTOR ABOUT ANOTHER VECTOR *

SUBROUTINE ROTATE(XN, YN, ZN, XP, YP, ZP, THETA, UX, UY, UZ)
IMPLICIT REAL*8(A-H, O-Z)
CT = DCOS(THETA)
ST = DSIN(THETA)
VT = 1.0 - CT
R11 = UX*UX*VT + CT
R12 = UX*UY*VT - UZ*ST
R13 = UX*UZ*VT + UY*ST
R21 = UX*UY*VT + UZ*ST
R22 = UY*UY*VT + CT
R23 = UY*UZ*VT - UX*ST
R31 = UX*UZ*VT - UY*ST
R32 = UY*UZ*VT + UX*ST
R33 = UZ*UZ*VT + CT
CALL TRCOORD(XN, YN, ZN, R11, R12, R13, R21, R22, R23, R31, R32, R33, 0.0, 0.0, 0.0, XP, YP, ZP)
RETURN
END

* **** SUBROUTINE NOLIN ****

SUBROUTINE NOLIN(FUNC, NSIG, NE, NC, X, Y, YI, DELTA, A, IPVT, WORK)
IMPLICIT REAL*8(A-H, O-Z)
DIMENSION X(NE), Y(NE), YI(NE), A(NE, NE), IPVT(NE), WORK(NE)
EXTERNAL FUNC
NDIM = NE
EPSI=1.D00/10.D00**NSIG
CALL NONLIO(FUNC,EPSI,NE,NC,X,DELTA,NDIM,A,Y,YI,WORK,IPVT)
RETURN
END

***** SUBROUTINE NOLINO *****

SUBROUTINE NOLINO(FUNC,EPSI,NE,NC,X,DELTA,NDIM,A,Y,YI,WORK,IPVT)
IMPLICIT REAL*8(A-H,O-Z)
DIMENSION X(NE),Y(NE),YI(NE),IPVT(NE),WORK(NE),A(NDIM,NE)
EXTERNAL FUNC

* NC: # OF COUNT TIMES
DO 5 I=1,NC
CALL FUNC(X,Y,NE)
END 5

* NE: # OF EQUATIONS
DO 15 J=1,NE
IF (DABS(Y(J)).GT.EPSI) GO TO 25
15 CONTINUE
GO TO 105

25 DO 35 J=1,NE
35 YI(J)=Y(J)
DO 45 J=1,NE
DIFF=DABS(X(J))*DELTA
IF (DABS(X(J)).LT.1.D-12) DIFF=DELTA
XMAM=X(J)
X(J)=X(J)-DIFF
CALL FUNC(X,Y,NE)
X(J)=XMAM
DO 55 K=1,NE
A(K,J)=(YI(K)-Y(K))/DIFF
55 CONTINUE
45 CONTINUE
DO 65 J=1,NE
65 Y(J)=-YI(J)
CALL DECOMP(NDIM,NE,A,COND,IPVT,WORK)
CALL SOLVE(NDIM,NE,A,Y,IPVT)
DO 75 J=1,NE
X(J)=X(J)+Y(J)
75 CONTINUE
5 CONTINUE
105 RETURN
END

***** SUBROUTINE DECOMP *****

SUBROUTINE DECOMP(NDIM,N,A,COND,IPVT,WORK)
IMPLICIT REAL*8(A-H,O-Z)
DIMENSION A(NDIM,N),WORK(N),IPVT(N)

DECOMPOSES A REAL MATRIX BY GAUSSIAN ELIMINATION,
AND ESTIMATES THE CONDITION OF THE MATRIX.

159
USE SUBROUTINE SOLVE TO COMPUTE SOLUTIONS TO LINEAR SYSTEM.

INPUT..

NDIM = DECLARED ROW DIMENSION OF THE ARRAY CONTAINING A
N = ORDER OF THE MATRIX
A = MATRIX TO BE TRIANGULARIZED

OUTPUT..

A CONTAINS AN UPPER TRIANGULAR MATRIX U AND A PREMUTED
VERSION OF A LOWER TRIANGULAR MATRIX I-L SO THAT
(PERMUTATION MATRIX)*A=L*U

COND = AN ESTIMATE OF THE CONDITION OF A.
FOR THE LINEAR SYSTEM A*X = B, CHANGES IN A AND B
MAY CAUSE CHANGES COND TIMES AS LARGE IN X.
IF COND+1.0 .EQ. COND, A IS SINGULAR TO WORKING
PRECISION. COND IS SET TO 1.0D+32 IF EXACT
SINGULARITY IS DETECTED.

IPVT = THE PIVOT VECTOR
IPVT(K) = THE INDEX OF THE K-TH PIVOT ROW
IPVT(N) = (-1)**(NUMBER OF INTERCHANGES)

WORK SPACE.. THE VECTOR WORK MUST BE DECLARED AND INCLUDED
IN THE CALL. ITS INPUT CONTENTS ARE IGNORED.
ITS OUTPUT CONTENTS ARE USUALLY UNIMPORTANT.

THE DETERMINANT OF A CAN BE OBTAINED ON OUTPUT BY
DET(A) = IPVT(N) * A(1,1) * A(2,2) * ... * A(N,N).

IPVT(N)=1
IF (N.EQ.1) GO TO 150
NM1=N-1

ANORM=0.D0
DO 20 J=1,N
T=0.D0
DO 10 I=1,N
10 T=T+DABS(A(I,J))
IF (T.GT.ANORM) ANORM=T
20 CONTINUE

DO 70 K=1,NM1
KP1=K+1

M=K
DO 30 I=KP1,N

COMPUTE THE 1-NORM OF A.

DO 20 J=1,N
T=0.D0
DO 10 I=1,N
10 T=T+DABS(A(I,J))
IF (T.GT.ANORM) ANORM=T
20 CONTINUE

DO GAUSSIAN ELIMINATION WITH PARTIAL
PIVOTING.

DO 70 K=1,NM1
KP1=K+1

M=K
DO 30 I=KP1,N
IF (DABS(A(I,K)).GT.DABS(A(M,K))) M=I

30 CONTINUE
IPVT(K)=M
IF (M.NE.K) IPVT(N)=-IPVT(N)
T=A(M,K)
A(M,K)=A(K,K)
A(K,K)=T

* SKIP THE ELIMINATION STEP IF PIVOT IS ZERO.
* IF (T.EQ.0.D0) GO TO 70
*
* COMPUTE THE MULTIPLIERS.
* DO 40 I=KP1,N
A(I,K)=-A(I,K)/T
*
* INTERCHANGE AND ELIMINATE BY COLUMNS.
* DO 60 J=KP1,N
T=A(M,J)
A(M,J)=A(K,J)
A(K,J)=T
IF (T.EQ.0.D0) GO TO 60
DO 50 I=KP1,N
50 A(I,J)=A(I,J)+A(I,K)*T
60 CONTINUE
70 CONTINUE

* COND = (1-NORM OF A)*(AN ESTIMATE OF THE 1-NORM OF A-INV)
* THE ESTIMATE IS OBTAINED BY ONE STEP OF INVERSE ITERATION FOR THE
* SMALL SINGULAR VECTOR. THIS INVOLVES SOLVING TWO SYSTEMS
* OF EQUATIONS, (A-TRANSPOSE)*Y = E AND A*Z = Y WHERE E
* IS A VECTOR OF +1 OR -1 COMPONENTS CHOSEN TO CAUSS GROWTH IN Y.
* ESTIMATE = (1-NORM OF Z)/(1-NORM OF Y)
*
* SOLVE (A-TRANSPOSE)*Y = E.
* DO 100 K=1,N
T=0. DO
IF (K.EQ.1) GO TO 90
KM1=K-1
DO 80 I=1,KM1
80 T=T+A(I,K)*WORK(I)
90 EK=1.D0
IF (T.LT.0.D0) EK=-1.D0
IF (A(K,K).EQ.0.D0) GO TO 160
A11=A(1,1)
WORK(K)=-(EK+T)/A(1,1)
100 CONTINUE
* DO 120 KB=1,NM1
K=N-KB
T=0. DO
KP1=K+1
DO 110 I=KP1,N
110 T=T+A(I,K)*WORK(K)
WORK(K) = T
M = IPVT(K)
IF (M.EQ.K) GO TO 120
T = WORK(M)
WORK(M) = WORK(K)
WORK(K) = T
120 CONTINUE
*
YNORM = 0.D0
DO 130 I = 1, N
130 YNORM = YNORM + DABS(WORK(I))
*
SOLVE A*X = Y
CALL SOLVE (NDIM, N, A, WORK, IPVT)
*
ZNORM = 0.D0
DO 140 I = 1, N
140 ZNORM = ZNORM + DABS(WORK(I))
*
ESTIMATE THE CONDITION.
COND = ANORM * ZNORM / YNORM
IF (COND.LT.1.D0) COND = 1.D0
RETURN
*
150 COND = 1.D0
IF (A(1,1).NE.0.D0) RETURN
*
EXACT SINGULARITY
160 COND = 1.D032
RETURN
END
SUBROUTINE SOLVE (NDIM, N, A, B, IPVT)
*
IMPLICIT REAL*8 (A-H, O-Z)
DIMENSION A(NDIM, N), B(N), IPVT(N)
*
SOLVES A LINEAR SYSTEM, A*X = B
DO NOT SOLVE THE SYSTEM IF DECOMP HAS DETECTED SINGULARITY.
*
~COMPUTER METHODS FOR MATHEMATICAL COMPUTATIONS~, BY G. E. FORSYTHE,
~ M. A. MALCOLM, AND C. B. MOLER (PRENTICE-HALL, 1977)
*
INPUT..
*
NDIM = DECLARED ROW DIMENSION OF ARRAY CONTAINING A
N = ORDER OF MATRIX
A = TRIANGULARIZED MATRIX OBTAINED FROM SUBROUTINE DECOMP
B = RIGHT HAND SIDE VECTOR
IPVT = PIVOT VECTOR OBTAINED FROM DECOMP
*
OUTPUT..
*
* B = SOLUTION VECTOR, X

* DO THE FORWARD ELIMINATION.

IF (N.EQ.1) GO TO 50
NM1=N-1
DO 20 K=1,NM1
   KP1=K+1
   M=IPVT(K)
   T=B(M)
   B(M)=B(K)
   B(K)=T
   DO 10 I=KP1,N
      B(I)=B(I)+A(I,K)*T
  10 CONTINUE
20 CONTINUE

* NOW DO THE BACK SUBSTUTION.

DO 40 KB=1,NM1
   KM1=N-KB
   K=KM1+1
   B(K)=B(K)/A(K,K)
   T=-B(K)
   DO 30 I=1,KM1
      B(I)=B(I)+A(I,K)*T
  30 CONTINUE
40 CONTINUE
50 B(1)=B(1)/A(1,1)
RETURN
END
IMPLICIT REAL*8(A-H,K,M-Z)
REAL*8 X(1), F(1), FI(1), PAR(6), LM, TX(6), TF(6), TF1(6), TPAR(19),
      AZSP(1,1), WORKP(1), AZS(6,6), WORK(6), LANDAP, LANPDG, LANDP0
CHARACTER*8 HG, HNGR
DIMENSION IPVT(6), IPVTP(1)
EXTERNAL PCNI, PCN2, TCN
COMMON/P1/PAR
COMMON/T1/TPAR
COMMON/A0/HG
COMMON/A1/p11, p12, p13, p21, p22, p23, p31, p32, p33, p1, p2, p3
COMMON/A2/SG, CSRT2, QG, SNPSIG, CSPSIG
COMMON/A3/TND1, TND2, RITAG
COMMON/A4/CSD2, SND2, CSPIT2, SNPIT2
COMMON/A5/CSPQ, SQNQ, THETAG
COMMON/B1/CSPH11, SP, EM, LM, CSRT1, CSD1, SND1, CSLANP, SNLANP
COMMON/B2/CSPIT1, SNPIT1, MP1, MG2, QP
COMMON/B3/B2fx, B2fy, B2fz
COMMON/B4/CSPH2, SNPH2, CSPH21, SNPH21
COMMON/B5/XCR, ZCR
COMMON/C1/UG, CSTAUG, SNTAUG
COMMON/C2/N2fx, N2fy, N2fz
COMMON/D1/CSTAUP, SNTAUP
COMMON/E1/XBF, YBF, ZBF
COMMON/F1/PHIGO
COMMON/G1/DA1, DV1

* INPUT THE DESIGN DATA

* TN1 : number of pinion teeth ------ sec. 3.1
* TN2 : number of gear teeth ------- sec. 3.1
* RT1dg, RT1min : root angle of pinion (degree and arc minute, respectively)
  ------ sec. 3.1
* RT2dg, RT2min : root angle of gear (degree and arc minute, respectively)
  ------ sec. 3.1
* SHAFdg : shaft angle (degree) ------ sec. 3.1
* BETAdg : mean spiral angle (degree) ------ sec. 3.1
* ADIA : average gear cutter diameter
sec. 3.1: point width of gear cutter
sec. 3.1: mean cone distance
sec. 3.1: blade angle of gear cutter (degree)
fig. 10: radius of blade
gear convex side
fig. 10: radius of blade
gear concave side
fig. 10: angle measured counterclockwise from root of gear to the tangent of the contact path (degree)
gear convex side
fig. 10: angle measured counterclockwise from root of gear to the tangent of the contact path (degree)
gear concave side
fig. 19: first derivative of gear ratio
gear convex side
fig. 19: first derivative of gear ratio
gear concave side
eq. (3.76): semimajor axis of contact ellipse
gear convex side
eq. (3.76): semimajor axis of contact ellipse
gear concave side
eq. (3.76): hand of gear ('L' or 'R')
+ : pinion mounting distance being increased
- : pinion mounting distance being decreased

fig. 18: amount of pinion shaft offset
the same sense as yf shown in
eq. (3.76): elastic approach

sec. 3.1.1: gear blade angle
sec. 3.1.1: pinion blade angle
tip radius of gear cutter
tip radius of pinion cutter
gear radial
pinion radial
gear cradle angle
pinion cradle angle

OUTPUT OF THE BASIC MACHINE-TOOL SETTINGS

PSIGdg: gear blade angle
PSIPdg: pinion blade angle
RG: tip radius of gear cutter
RP: tip radius of pinion cutter
SG: gear radial
SP: pinion radial
QGdg: gear cradle angle
QPdg: pinion cradle angle
* MG2 : gear cutting ratio
* MPI : pinion cutting ratio
* EM : machining offset
* LM : machine center to back + sliding base
* XCR, ZCR : x and z coordinates of center of blade

DATA TN1,TN2/10.D00,41.D00/
DATA RT1dg,RT1min/12.D00,1.D00/
DATA RT2dg,RT2min/72.D00,25.D00/
DATA SHAFdg,BETAAdg/90.D00,35.D00/
DATA ADIA/6.0D00/
DATA W/0.08D00/
DATA A/3.226D00/
DATA ALPHAdg/20.D00/
DATA DLTXdg/ 90.D00/
DATA DLTVdg/ 75.D00/
DATA M21XPR/-3.5D-03/
DATA M21VPR/5.2D-03/
DATA AXILX/0.1710D00/
DATA AXILV/0.1810D00/
DATA RX/40.00D00/
DATA RV/50.00D00/
DATA HNGR/'L'/
DATA DA,DV/0.D00,0.D00/
DATA DEF/0.00025D00/
DATA EPS/1.D-12/

DA1=DA
DV1=DV
HG=HNGR

* CONVERT DEGREES TO RADIANS

CNST=4.D00*DATAN(1.D00)/180.D00
RITAG=90.D00*CNST
DLTX=DLTXdg*CNST
DLTV=DLTVdg*CNST
RT1=(RT1dg+RT1min/60.D00)*CNST
RT2=(RT2dg+RT2min/60.D00)*CNST
BETA=BETAAdg*CNST
PSIG=ALPHAdg*CNST
SHAFT=SHAFdg*CNST
CSRT2=DCOS(RT2)
SNRT2=DSIN(RT2)
CSRT1=DCOS(RT1)
SNRT1=DSIN(RT1)

* CALCULATE PITCH ANGLES

MM21=TN1/TN2
c ------ eq. (3.1)

166
PITCH2 = \text{DATAN}\left(\frac{\text{DSIN}(\text{SHAFT})}{(\text{MM21} + \text{DCOS}(\text{SHAFT}))}\right)

\text{IF}(\text{PITCH2} < 0.0) \text{THEN}
\quad \text{PITCH2} = \text{PITCH2} + 180.00
\text{END IF}

\text{CSPIT2} = \text{DCOS}(\text{PITCH2})
\text{SNPIT2} = \text{DSIN}(\text{PITCH2})

\text{c eq. (3.2)}
\text{PITCH1} = \text{SHAFT} - \text{PITCH2}
\text{CSPIT1} = \text{DCOS}(\text{PITCH1})
\text{SNPIT1} = \text{DSIN}(\text{PITCH1})

\text{*}
\text{CALCULATE DEEDENDUM ANGLES}
\text{*}

\text{c eq. (3.3)}
\text{D1} = \text{PITCH1} - \text{RT1}
\text{D2} = \text{PITCH2} - \text{RT2}
\text{CSD1} = \text{DCOS}(\text{D1})
\text{SND1} = \text{DSIN}(\text{D1})
\text{TND1} = \text{SND1}/\text{CSD1}
\text{CSD2} = \text{DCOS}(\text{D2})
\text{SND2} = \text{DSIN}(\text{D2})
\text{TND2} = \text{SND2}/\text{CSD2}

\text{*}
\text{CALCULATE GEAR CUTTING RATIO}
\text{*}

\text{c eq. (3.7)}
\text{MG2} = \text{DSIN}(\text{PITCH2})/\text{CSD2}

\text{*}
\text{FOR GEAR CONVEX SIDE I = 1, FOR GEAR CONCAVE SIDE I = 2.}
\text{*}
\text{DO 99999 I=1,2}
\text{IF(I .EQ. 1) THEN}
\quad \text{WRITE(72,*)'GEAR CONVEX SIDE'}
\quad \text{DLTA} = \text{DLTX}
\quad \text{M21PRM} = \text{M21XPR}
\quad \text{AXIL} = \text{AXILX}
\quad \text{R} = \text{RX}
\text{ELSE}
\quad \text{WRITE(72,*)'GEAR CONCAVE SIDE'}
\quad \text{DLTA} = \text{DLTV}
\quad \text{M21PRM} = \text{M21VPR}
\quad \text{AXIL} = \text{AXILV}
\quad \text{R} = \text{RV}
\text{END IF}
\text{WRITE(72,*)}
\text{c eq. (3.76)}
\quad \text{AXIA} = \text{DEF}/(\text{AXIL} \times \text{AXIL})

\text{*}
\text{CALCULATE GEAR BLADE ANGLE}
\text{*}

\text{c sec. 2.2}
\text{IF(I .EQ. 2) THEN}
\quad \text{PSIG} = 180.00 \times \text{CNST} - \text{PSIG}
END IF
CSPSIG=DCOS(PSIG)
SNPSIG=DSIN(PSIG)
CTPSIG=CSPSIG/SNPSIG

* CALCULATE CUTTER TIP RADIUS
*
c ---- eq. (3.8)
IF(I .EQ. 1) THEN
   RG=(ADIA-W)/2.D00
ELSE
   RG=(ADIA+W)/2.D00
END IF

* CALCULATE RADIAL
*
c ---- eq. (3.9)
IF(I .EQ. 1) THEN
   SG=DSQRT(ADIA*ADIA/4.D00+A*A*CSD2*CSD2-A*ADIA*CSD2*DSIN(BETA))
END IF

* CALCULATE CRADLE ANGLE
*
c ---- eq. (3.10)
QG=DCOS((A*A*CSD2*CSD2+SG*SG-ADIA*ADIA/4.D00)/(2.D00*A*SG*CSD2))
CSQG=DCOS(QG)
SNQG=DSIN(QG)
END IF

PAR(1)=RG*CTPSIG*CSPSIG
PAR(4)=RG*CTPSIG

* CALCULATE PHIG
*
PHIG=0.D00
PHIGO=PHIG
CSPHIG=DCOS(PHIG)
SNPHIG=DSIN(PHIG)

* IF(HG .EQ. 'L') THEN
   IF(I .EQ. 1) THEN
   Mmc=Mms*Msc
   CALL COMBI(mll,m12,m13,m21,m22,m23,m31,m32,m33,m1,m2,m3,
   1.D00,0.D00,0.D00,0.D00,0.D00,0.D00,0.D00,0.D00,0.D00,-SNPHIG,CSPHIG,
   0.D00,0.D00,0.D00,0.D00,0.D00,0.D00,-SNQG,CSPHIG,SNQG,
   0.D00,0.D00,0.D00,0.D00,0.D00,0.D00,0.D00,0.D00,0.D00,0.D00,0.D00,0.D00,0.D00,0.D00,0.D00,0.D00)
   END IF
   * Mpc=Mpm*Mmc
   CALL COMBI(p11,p12,p13,p21,p22,p23,p31,p32,p33,p1,p2,p3,
   CSD2,0.D00,-SND2,0.D00,1.D00,0.D00,0.D00,0.D00,0.D00,0.D00,0.D00,0.D00,0.D00,0.D00,0.D00,0.D00,0.D00,
   0.D00,0.D00,0.D00,0.D00,
ELSE

IF(I .EQ. 1) THEN

Mmc = Mms * Ms

END IF

Mpc = Mpm * Mmc

END IF

* DETERMINE MAIN CONTACT POINT

* CALCULATE THETAG

X(1) represents THETAG

PAR(2) = (MG2 - SNRT2) * CSPSIG

IF(HG .EQ. 'L') THEN

PAR(3) = -SNQG * CSRT2 * SNPSIG

END IF

step 1 in sec. 3.2

X(1) = QG - BETA + RITAG

ELSE

PAR(3) = SNQG * CSRT2 * SNPSIG

END IF

step 1 in sec. 3.2

X(1) = -(QG - BETA + RITAG)

END IF

CALL NONLIN(PCNI, 14, 1, 100, X, FI, 1.D-5, AZSP, IPVTP, WORKP)

THETAG = X(1)

CSTHEG = DCOS(THETAG)

SNTHEG = DSIN(THETAG)

* CALCULATE TAUG

* eq. (2.38)

IF(HG .EQ. 'L') THEN

TAUG = THETAG - QG + PHIG

ELSE

TAUG = THETAG + QG - PHIG

END IF

CSTAUG = DCOS(TAUG)

SNIAUG = DSIN(TAUG)

*
* CALCULATE UG

* 

\[
\begin{align*}
\text{IF}(\text{HG} \ . \ EQ. \ 'L') & \text{THEN} \\
\text{UG} & = \text{RG} \times \text{CTPSIG} \times \text{CSPSIG} - \text{SG} \times ((\text{MG2} - \text{SNRT2}) \times \text{CSPSIG} \times \text{SNTHEG} - \text{DSIN}((\text{QG} - \text{PHIG}) \times \text{CSRT2} \times \text{SNPSIG}) / (\text{CSRT2} \times \text{SNTAUG}) \\
\text{ELSE} & \\
\text{UG} & = \text{RG} \times \text{CTPSIG} \times \text{CSPSIG} - \text{SG} \times ((\text{MG2} - \text{SNRT2}) \times \text{CSPSIG} \times \text{SNTHEG} + \text{DSIN}((\text{QG} - \text{PHIG}) \times \text{CSRT2} \times \text{SNPSIG}) / (\text{CSRT2} \times \text{SNTAUG}) \\
\end{align*}
\]

* CONVERT RADIANS TO DEGREES

* 

\[
\begin{align*}
\text{PSIGDG} & = \text{PSIG} / \text{CNST} \\
\text{TAUGDG} & = \text{TAUG} / \text{CNST} \\
\text{QGDG} & = \text{QG} / \text{CNST} \\
\text{THEGDG} & = \text{THETAG} / \text{CNST} \\
\text{PHIGDG} & = \text{PHIGO} / \text{CNST} \\
\end{align*}
\]

* OUTPUT OF GEAR SETTINGS

* 

\[
\begin{align*}
\text{WRITE}(72,10000) & \text{PSIGDG, QGDG, RG, SG, MG2, TAUGDG, UG, THEGDG, PHIGDG} \\
\end{align*}
\]

* CALCULATE MAIN CONTACT POINT

* 

\[
\begin{align*}
\text{eq. (2.1)} & \\
\text{Bcx} & = \text{RG} \times \text{CTPSIG} - \text{UG} \times \text{CSPSIG} \\
\text{Bcy} & = \text{UG} \times \text{SNPSIG} \times \text{SNTHEG} \\
\text{Bcz} & = \text{UG} \times \text{SNPSIG} \times \text{CSTHEG} \\
\text{eq. (2.2)} & \\
\text{Ncx} & = \text{SNPSIG} \\
\text{Ncy} & = \text{CSPSIG} \times \text{SNTHEG} \\
\text{Ncz} & = \text{CSPSIG} \times \text{CSTHEG} \\
\text{eq. (2.9)} & \\
\text{EGIcx} & = 0.D00 \\
\text{EGIcy} & = \text{CSTHEG} \\
\text{EGIcz} & = -\text{SNTHEG} \\
\text{eq. (3.13)} & \\
\text{CALL TRCOOR}(\text{Bpx}, \text{Bpy}, \text{Bpz}, \\
\text{. p11, p12, p13, p21, p22, p23, p31, p32, p33, p1, p2, p3}, \\
\text{. Bcx, Bcy, Bcz}) \\
\text{eq. (3.16)} & \\
\text{CALL TRCOOR}(\text{Npx}, \text{Npy}, \text{Npz}, \\
\text{. p11, p12, p13, p21, p22, p23, p31, p32, p33, 0.D00, 0.D00, 0.D00}, \\
\text{. Ncx, Ncy, Ncz}) \\
\text{eq. (3.17)} & \\
\text{CALL TRCOOR}(\text{EGIpx}, \text{EGIpy}, \text{EGIpz}, \\
\text{. p11, p12, p13, p21, p22, p23, p31, p32, p33, 0.D00, 0.D00, 0.D00}, \\
\text{. EGIcx, EGIcy, EGIcz}) \\
\text{fig. 18 & sec. 3.3} & \\
\text{Bfx} & = \text{Bpx} \\
\text{Bfy} & = \text{Bpy} \\
\text{Bfz} & = \text{Bpz}
\end{align*}
\]
Nfx=NPx
Nfy=Npy
Nfz=Npz
EGIfx=EGIpx
EGIfy=EGIpy
EGIfz=EGIpz

* CALCULATE LANDAP
*
UG0=UG
CSTAG0=CSTAG
SNTAG0=SNTAG
PHIGO=PHIG
THETG0=THETAG

DO 99999 J=1,2
*
CSTAG=CSTAG0
SNTAG=SNTAG0
UG=UG0
PHIG=PHIG0
THETAG=THETAG0

IF(J .EQ. 1)THEN
   WRITE(72,*)'BLADE CONCAVE DOWN'
ELSE
   WRITE(72,*)'BLADE CONCAVE UP'
END IF
LANDAP = DACOS (CSD1 * Nfx - SND1 * Nfz)
IF (I .EQ. 1) THEN
   IF (J .EQ. 1) THEN
      LANDAP = 360.00 * CNST - LANDAP
      PSIP = 450.00 * CNST - LANDAP
   ELSE
      LANDAP = 180.00 * CNST - LANDAP
      PSIP = 270.00 * CNST - LANDAP
   END IF
ELSE
   IF (J .EQ. 1) THEN
      PSIP = 90.00 * CNST - LANDAP
   ELSE
      LANDAP = 180.00 * CNST + LANDAP
      PSIP = 270.00 * CNST - LANDAP
   END IF
END IF

END IF
CSLANP = DCOS (LANDAP)
SNLANP = DSIN (LANDAP)

* CALCULATE TAUP
*
TAUP = DATAN2 (Nfy/SNLANP, (Nfx-CSD1*CSLANP)/(-SND1*SNLANP))
IF (J .EQ. 2) THEN
   TAUP = DATAN2 (-Nfy/SNLANP, (-Nfx-CSD1*CSLANP)/(-SND1*SNLANP))
**END IF**
CSTAUP=DCOS(TAUP)
SNTAUP=DSIN(TAUP)

* * CALCULATE PRINCIPAL CURVATURES AND DIRECTIONS OF THE GEAR CUTTER *
* *
c ----- eq. (2.10)
   KG1=-CTPSIG/UG
c ----- eq. (2.12)
   KGII=0.D00
c ----- the second principal direction is determined by rotating of
   the first principal direction about unit normal by 90 degrees
   CALL ROTATE(EGIIfx,EGIIfy,EGIIfz,EGIffx,EGIffy,EGIffz,RITAG,
           . Nfx,Nfy,Nfz)
* *
* CALCULATE W2G *
* *
c ----- eqs. (3.18)-(3.20)
   IF(HG .EQ. 'L') THEN
      W2fx=-SNPIT2
      WGfx=-MG2*CSD2
      W2fy=0.D00
      WGfy=0.D00
      W2fz=CSPIT2
      WGfz=-MG2*SND2
   ELSE
      W2fx=SNPIT2
      WGfx=MG2*CSD2
      W2fy=0.D00
      WGfy=0.D00
      W2fz=-CSPIT2
      WGfz=MG2*SND2
   END IF
* *
   W2Gfx=W2fx-WGfx
   W2Gfy=W2fy-WGfy
   W2Gfz=W2fz-WGfz
* *
* CALCULATE VT2, VTG, AND VT2G *
* *
c ----- eq. (3.22)
   CALL CROSS(VT2fx,VT2fy,VT2fz,W2fx,W2fy,W2fz,Bfx,Bfy,Bfz)
c ----- eq. (3.21)
   CALL CROSS(VTGFx,VTGfy,VTGFz,WGfx,WGfy,WGfz,Bfx,Bfy,Bfz)
c ----- eq. (3.23)
   VT2Gfx=VT2fx-VTGFx
   VT2Gfy=VT2fy-VTGFy
   VT2Gfz=VT2fz-VTGFz
* *
* CALCULATE V(2G)GI AND V(2G)GII *
* *
c ----- eq. (3.24)
   CALL DOT(VGI,EGIfx,EGIfy,EGIfz,VT2Gfx,VT2Gfy,VT2Gfz)
CALL DOT(VGII,EGIIfx,EGIIfy,EGIIfz,VT2Gfx,VT2Gfy,VT2Gfz)

* CALCULATE A13, A23, A33

CALL DET(DETI,W2Gfx,W2Gfy,W2Gfz,Nfx,Nfy,Nfz,EGIfx,EGIfy,EGIfz)
A13=-KGI*VGI-DETI

CALL DET(DETII,W2Gfx,W2Gfy,W2Gfz,Nfx,Nfy,Nfz,EGIIfx,EGIIfy,EGIIfz)
A23=-KGI*VGI-DETI

CALL DET(DET3,Nfx,Nfy,Nfz,W2Gfx,W2Gfy,W2Gfz,VT2Gfx,VT2Gfy,VT2Gfz)
CALL CROSS(Cx,Cy,Cz,W2fx,W2fy,W2fz,VTGfx,VTGfy,VTGfz)
CALL CROSS(Dx,Dy,Dz,WGfx,WGfy,WGfz,VT2fx,VT2fy,VT2fz)
CALL DOT(DET45,Nfx,Nfy,Nfz,Cx-Dx,Cy-Dy,Cz-Dz)
A33=KGI*VGI*VGI*VGI+KGII*VGII*VGII-DET3-DET45

* CALCULATE SIGMA

P=A23*A23-AI3*AI3+(KGI-KGII)*A33
SIGDBL=DATAN(2.D00*A13*A23/P)
SIGMA=0.5DO0*SIGDBL

* CALCULATE K21 AND K2II

T1=P/(A33*DCOS(SIGDBL))
T2=KGI+KGII-(A13*A13+A23*A23)/A33
K21=(T1+T2)/2.D00
K2II=(T2-T1)/2.DO0

* CALCULATE E21 AND E211

CALL ROTATE(E21fx,E21fy,E21fz,EGIfx,EGIfy,EGIfz,-SIGMA,Nfx,Nfy
.Nfz)
CALL ROTATE(E211fx,E211fy,E211fz,E21fx,E21fy,E21fz,RITAG
.Nfx,Nfy,Nfz)

TNETAG=DSIN(DLTA+SIGMA)/DCOS(DLTA+SIGMA)

* CALCULATE W2

IF(HG .EQ. 'L') THEN
    W2fx=-MM21*SNPIT2
    W2fy=0.DO0
    W2fz=MM21*CSPIT2
ELSE
    W2fx=MM21*SNPIT2
    W2fy=0.DO0

173
W2fz=-MM21*CSPIT2
END IF
*
* CALCULATE W1
*
  c ----- eq. (3.32)
  IF (HG .EQ. 'L') THEN
    W1fx=-SNPIT1
    W1fy=0.D00
    W1fz=-CSPIT1
  ELSE
    W1fx=SNPIT1
    W1fy=0.D00
    W1fz=CSPIT1
  END IF
*
* CALCULATE W12
*
  c ----- eq. (3.34)
    W12fx=W1fx-W2fx
    W12fy=W1fy-W2fy
    W12fz=W1fz-W2fz
*
* CALCULATE VT2
*
  c ----- eq. (3.36)
    CALL CROSS(VT2fx,VT2fy,VT2fz,W2fx,W2fy,W2fz,Bfx,Bfy,Bfz)
*
* CALCULATE VT1
*
  c ----- eq. (3.35)
    CALL CROSS(VT1fx,VT1fy,VT1fz,W1fx,W1fy,W1fz,Bfx,Bfy,Bfz)
*
* CALCULATE VT12
*
  c ----- eq. (3.37)
    VT12fx=VT1fx-VT2fx
    VT12fy=VT1fy-VT2fy
    VT12fz=VT1fz-VT2fz
*
* CALCULATE V2
*
  c ----- eq. (3.38)
    CALL DOT(V2I,VT12fx,VT12fy,VT12fz,E2Ifx,E2Ify,E2Ifz)
  c ----- eq. (3.39)
    CALL DOT(V2II,VT12fx,VT12fy,VT12fz,E2IIfx,E2IIfy,E2IIfz)
*
* CALCULATE A31
*
  c ----- eq. (3.40)
    CALL DET(DET1,W12fx,W12fy,W12fz,Nfx,Nfy,Nfz,E2Ifx,E2Ify,E2Ifz)
    A31=-K2I*V2I-DET1
  c ----- eq. (A.33)
A13 = A31

* CALCULATE A32
* eq. (3.41)
CALL DET(DET2,W12Fx,W12Fy,W12Fz,Nfx,Nfy,Nfz,E2Ifx,E2Ify,E2Ifz)
A32 = -K2II*V2II - DET2

* CALCULATE A33
* eq. (3.42)
CALL DET(DET3,Nfx,Nfy,Nfz,W12Fx,W12Fy,W12Fz,VT12Fx,VT12Fy,VT12Fz)
CALL CROSS(Cx,Cy,Cz,WlFx,WlFy,WlFz,VT2Fx,VT2Fy,VT2Fz)
CALL CROSS(Dx,Dy,Dz,W2Fx,W2Fy,W2Fz,VT1Fx,VT1Fy,VT1Fz)
CALL DOT(DOT1,Nfx,Nfy,Nfz,Cx-Dx,Cy-Dy,Cz-Dz)
CALL DET(DET4,Nfx,Nfy,Nfz,W2Fx,W2Fy,W2Fz,Bfx,Bfy,Bfz)
A33 = K2I*V2I*V2I + K2II*V2II*V2II - DET3 - DOT1 + M21PRM*DET4

* CALCULATE ETAP
* eq. (3.53)
ETAP = DATAN(((A33 + A31*V2I)*TNETAG - A31*V2I) / (A33 + A32*V2I - V2I*TNETAG))
TNETAP = DSIN(ETAP) / DCOS(ETAP)

* CALCULATE A11, A12, AND A22
* eq. (3.72)
N3 = (1.00 + TNETAP*TNETAP)*A33

* CALCULATE N1
* eq. (3.73)
N1 = (A13*A13 - (A23*TNETAP)**2) / N3

* CALCULATE N2
* eq. (3.74)
N2 = (A23 + A13*TNETAP) * (A13 + A23*TNETAP) / N3

* CALCULATE KS2
* eq. (3.75)
KS2 = K2I + K2II
G2 = K2I - K2II

* Calculate KS1
* eqs. (3.66), (3.69) & description after eq. (3.60)
A11 = TNETAP*TNETAP / (1.00 + TNETAP*TNETAP)*(KS2 - KS1) + N1

* Calculate A12
* eqs. (3.67), (3.70) & description after eq. (3.60)
A12 = TNETAP / (1.00 + TNETAP*TNETAP)*(KS2 - KS1) + N2

* Calculate A22
* eqs. (3.68), (3.71) & description after eq. (3.60)
A22 = 1.00 / (1.00 + TNETAP*TNETAP)*(KS2 - KS1) - N1

* Calculate A21
* eq. (A.32)
A21 = A12

* CALCULATE SIGMA(12)
* eq. (3.77)
SIGDBL = DATAN((2.00*A12 / (K2I - K2II - A11 + A22)))
SIGM12 = .5D00*SIGDBL

* CALCULATE KII AND KIII
* c ----- eq. (3.78)
   G1 = 2. D00*A12/DSIN(SIGDBL)
c ----- eq. (3.79)
   KII = .5D00*(KS1+G1)
   KIII = .5D00*(KS1-G1)

* CALCULATE EII AND EIII
* c ----- similar to description after eq. (3.29)
   CALL ROTATE(E1Ifx,E1Ify,E1Ifz,E2Ifx,E2Ify,E2Ifz,-SIGM12,Nfx,Nfy,
              ,Nfz)
   CALL ROTATE(E2Ifx,E1Ify,E2Ifz,E1Ifx,E1Ify,E1Ifz,RITAG,
              ,Nfx,Nfy,Nfz)

* PINION
*
*
* CALCULATE PRINCIPAL DIRECTIONS OF THE PINION CUTTER
* c ----- eq. (3.92)
   IF(HG .EQ. 'L') THEN
      EPIfx = SND1*SNTAUP
      EPIfy = CSTAUP
      EPIfz = CSD1*SNTAUP
   ELSE
      EPIfx = -SND1*SNTAUP
      EPIfy = CSTAUP
      EPIfz = -CSD1*SNTAUP
   END IF
   IF(DACOS(EGIfx*EPIfx+EGIfy*EPIfy+EGIfz*EPIfz)/CNST .GT. 45.D00)
      THEN
         EPIfx = -EPIfx
         EPIfy = -EPIfy
         EPIfz = -EPIfz
      END IF
*
   CALL ROTATE(EPIIfx,EPIIfy,EPIIfz,EPIfx,EPIfy,EPIfz,RITAG,
              ,Nfx,Nfy,Nfz)
*
* CALCULATE THE ANGLE BETWEEN PRINCIPAL DIRECTIONS OF PINION AND CUTTER
* c ----- cross product of eII and ePI
   SNSIGM = (E1Ify*EPIfz-E1Ifz*EPIfy)/Nfx
   c ----- dot product of eII and ePI
   CSSIGM = E1Ifx*EPIfx+E1Ify*EPIfy+E1Ifz*EPIfz
   CS2SIG = 2. D00*CSSIGM*CSSIGM-1. D00
   TN2SIG = 2. D00*SNSIGM*CSSIGM/CS2SIG
*
* CALCULATE PRINCIPAL CURVATURES OF PINION CUTTER
c ----- eq. (2.20)
    KPII=1.000/R
    IF(J .EQ. 2) THEN
    KPII=-KPII
    END IF

c ----- eq. (3.94)
    KPI=(KPII*(KIII*CSS1G*CSS1G+KIII*SNS1G*SNS1G)-(KIII*KIII)/
    (KPII-KIII*SNS1G*SNS1G-KIII*CSS1G*CSS1G)

* CALCULATE A11, A12, AND A22
* c ----- eq. (A.31)
    A11=KPI-KIII*CSS1G*CSS1G-KIII*SNS1G*SNS1G
    c ----- eq. (A.32)
    A12=(KIII-KIII)*SNS1G*CSS1G
    c ----- eq. (A.34)
    A22=KPII-KIII*SNS1G*SNS1G-KIII*CSS1G*CSS1G

* CALCULATE ZCR
* c ----- eq. (3.101)
    IF(J .EQ. 1) THEN
    ZCR=(SNLANP/KPI)-R*SNLANP
    ELSE
    ZCR=-(SNLANP/KPI)-R*SNLANP
    END IF

* CALCULATE XCR
* c ----- eq. (3.99)
    Bmx=-Bfx*CSDI+Bfz*SNDI
    XCR=Bmx-R*CSLANP

* CALCULATE RP
* c ----- eq. (3.103)
    IF(I*J .EQ. 2) THEN
    RP=ZCR+DSQRT(DABS(R*R-XCR*XCR))
    ELSE
    RP=ZCR-DSQRT(DABS(R*R-XCR*XCR))
    END IF

* CALCULATE MCP
*    Z11=Nfy*EPI1fz-Nfz*EPI1fy
    Z12=Nfy*EPI1fx-Nfx*EPI1fy
    Z21=Nfy*EPI2fz-Nfz*EPI2fy
    Z22=Nfy*EPI2fx-Nfx*EPI2fy
    c ----- eqs. (3.107), (3.108)
    C11=Z11*CSDI+Z12*SNDI
    C12=-Z11*SNPITI+Z12*CSPITI
    C22=-Z21*SNPITI+Z22*CSPITI
IF(HG .EQ. 'R') THEN
   C11 = -C11
   C12 = -C12
   C22 = -C22
END IF

\[ T4 = \frac{(Bfy*CSRTI)}{(EPIIfx*CSD1-EPIIfz*SND1)} \]
IF(HG .EQ. 'R') THEN
   T4 = -T4
END IF

\[ T4 = \frac{(Bfy*CSRTI)}{(EPIIfx*CSD1-EPIIfz*SND1)} \]
\[ T1 = -CII/KPI \]
\[ T2 = \frac{(AII1*KPII*T4+AII1*C22-AI2*C12)}{(AII2*KPI)} \]

\[ U11 = T1*EPIIfx \]
\[ U12 = T2*EPIIfx+T4*EPIIfx \]
\[ U21 = T1*EPIIfy \]
\[ U22 = T2*EPIIfy+T4*EPIIfy \]
\[ U31 = T1*EPIIfz \]
\[ U32 = T2*EPIIfz+T4*EPIIfz \]

\[ V1 = U21*(Nfz*CSD1+Nfx*SND1)-Nfy*(U11*SND1+U31*CSD1) \]
\[ V2 = (U22*CSD1-U21*SNPIT1)*Nfz-(U11*CSPI1+U12*SND1+U32*CSD1-U31)*Nfy+\]
\[ *(SNPIT1)*Nfy+(U21*CSPI1+U22*SND1)*Nfx \]
\[ V3 = U22*CSPI1+Nfx+(U32*SNPIT1-U12*CSPI1)*Nfy-U22*SNPIT1*Nfz \]
IF(HG .EQ. 'R') THEN
   V1 = -V1
   V2 = -V2
   V3 = -V3
END IF

\[ H11 = -U21*CSPI1+SND1*(Bfz*SNPIT1-Bfx*CSPI1) \]
\[ H21 = U11*CSPI1-U31*SNPIT1+Bfy*SNRT1 \]
\[ H31 = U21*SNPIT1+CSD1*(Bfz*SNPIT1-Bfx*CSPI1) \]
\[ H12 = (Bfz*SNPIT1-Bfx*CSPI1-U22)*CSPI1 \]
\[ H22 = -(Bfy+U12*CSPI1-U32*SNPIT1) \]
\[ H32 = -(Bfz*SNPIT1-Bfx*CSPI1+U22)*SNPIT1 \]
IF(HG .EQ. 'R') THEN
   H11 = U21*CSPI1+SND1*(Bfz*SNPIT1-Bfx*CSPI1)
   H21 = U11*CSPI1+U31*SNPIT1+Bfy*SNRT1
   H31 = U21*SNPIT1+CSD1*(Bfz*SNPIT1-Bfx*CSPI1)
   H12 = (Bfz*SNPIT1-Bfx*CSPI1+U22)*CSPI1
   H22 = -(Bfy+U12*CSPI1-U32*SNPIT1)
   H32 = -(Bfz*SNPIT1-Bfx*CSPI1+U22)*SNPIT1
END IF

\[ V1 = U21*(Nfz*CSD1+Nfx*SND1)-Nfy*(U11*SND1+U31*CSD1) \]
FI = Nfx*H11 + Nfy*H21 + Nfz*H31
eq (3.140)
F2 = Nfx*H12 + Nfy*H22 + Nfz*H32
eq (3.145)

Y2 = A12*(2.00*KP1*T1*T2-V2-F1)
Y3 = A12*(KPI*T2*T2+KPI*T4*T4-V3-F2)-(KPI*T2+C12)*(KPI*T4+C22)
MP1 = -Y3/Y2

* CALCULATE EM AND LM
*

VTIPfx = U11*MP1 + U12
VTIPfy = U21*MP1 + U22
VTIPfz = U31*MP1 + U32
eq (3.122)

IF (HG .EQ. 'L') THEN
  EM = (Bfy*CSPIT1-VTIPfx)/(MPI*SND1)+Bfy
  LM = (Bfx*CSPIT1-Bfz*SNPIT1+VTIPfy)/MP1+Bfx*SND1+Bfz*CSD1
ELSE
  EM = (-Bfy*CSPIT1-VTIPfx)/(MP1*SND1)-Bfy
  LM = (Bfx*CSPIT1-Bfz*SNPIT1-VTIPfy)/MP1+Bfx*SND1+Bfz*CSD1
END IF

* CALCULATE SP AND QP
*

IF (HG .EQ. 'L') THEN
  IF (J .EQ. 1) THEN
    Z1 = -Bfy+EM-SNLANP/KPI*SNTAUP
    Z2 = Bfx*SND1+Bfz*CSD1-LM-SNLANP/KPI*CSTAUP
  ELSE
    Z1 = -Bfy+EM+SNLANP/KPI*SNTAUP
    Z2 = Bfx*SND1+Bfz*CSD1-LM-SNLANP/KPI*CSTAUP
  END IF
ELSE
  IF (J .EQ. 1) THEN
    Z1 = Bfy+EM+SNLANP/KPI*SNTAUP
    Z2 = Bfx*SND1+Bfz*CSD1-LM-SNLANP/KPI*CSTAUP
  ELSE
    Z1 = Bfy+EM-SNLANP/KPI*SNTAUP
    Z2 = Bfx*SND1+Bfz*CSD1-LM-SNLANP/KPI*CSTAUP
  END IF
END IF

SP = DSQRT(Z1*Z1+Z2*Z2)
QP = DATAN(Z1/Z2)

IF (HG .EQ. 'L') THEN
  THETAP = TAUP-QP
ELSE
  THETAP = TAUP+QP
END IF

* CONVERT RADIANS TO DEGREES

179
PSIPDG=PSIP/CNST
TAUPDG=TAUP/CNST
QPDG=QP/CNST
THEPDG=THETAP/CNST
LANPDG=LANDAP/CNST

* OUTPUT
* WRITE(72,10001) PSIPDG, QPDG, RP, SP, MPI, LANPDG, XCR, ZCR, EM, LM, TAUPDG, THEPDG

10000 FORMAT(1X, 'GEAR SETTINGS:', /
      1X, 'PSIGDG =', G20.12, 12X, 'QGDG =', G20.12, /
      1X, 'RG =', G20.12, 12X, 'SG =', G20.12, /
      1X, 'MG2 =', G20.12, 12X, 'TAUGDG =', G20.12, /
      1X, 'UG =', G20.12, 12X, 'THETAGDG =', G20.12, /
      1X, 'PHIGODG =', G20.12, //
      1X, 'PINION SETTINGS:', /)

10001 FORMAT(1X, 'PSIPDG =', G20.12, 12X, 'QPDG =', G20.12, /
      1X, 'RP =', G20.12, 12X, 'SP =', G20.12, /
      1X, 'MPI =', G20.12, 12X, 'LANPADP =', G20.12, /
      1X, 'XCR =', G20.12, 12X, 'ZCR =', G20.12, /
      1X, 'EM =', G20.12, 12X, 'LM =', G20.12, /
      1X, 'TAUPDG =', G20.12, 12X, 'THETAPDG =', G20.12, /)

* TCA
*
TPAR(1)=RG*CSPSIG/SNPSIG*CPSIG
TPAR(2)= (MG2-SNRT2)*CPSIG
TPAR(3)=CSRT2*SNPSIG
TPAR(4)=RG*CPSIG/SNPSIG
TPAR(5)=CSD2*SNPSIG
TPAR(6)=SND2*CPSIG
TPAR(7)=SND2*SNPSIG
TPAR(8)=CSD2*SNPSIG
TPAR(9)=ZCR*CSRT1
TPAR(10)=SP*CSRT1
TPAR(11)=EM*CSRT1
TPAR(12)=XCR*CSRT1
TPAR(13)=SP*(MPI-SNRT1)
TPAR(14)=EM*SNRT1
TPAR(15)=LM*SNRT1
TPAR(16)=J
TPAR(17)=R

* PHIP=0.D00
PHII1=0.D00
PHII2=0.D00
CSPHII=DCOS(PHII1)
SNPHII=DSIN(PHII1)

* TX(1)=PHIP
TX(2)=THETAP
TX(3)=LANDAP

180
TX(4) = PHI21
TX(5) = PHIG
TX(6) = THETAG
CALL NONLIN(TCN, 14, 6, 200, TX, TF, TF1, 1.D-5, AZS, IPVT, WORK)
PHIPO = TX(1)
THEPO = TX(2)
LANDPO = TX(3)
PHI210 = TX(4)
PHIG0 = TX(5)
THEG0 = TX(6)

TX(1) = PHIPO
TX(2) = THEPO
TX(3) = LANDPO
TX(4) = PHI210
TX(5) = PHIG0
TX(6) = THEG0

DPHI11 = 18.D00/36.D00*CNST

DO 100 IJ = 1, 60
CSPHI11 = DCOS(PHI11)
SNPHI11 = DSIN(PHI11)
CALL NONLIN(TCN, 14, 6, 200, TX, TF, TF1, 1.D-5, AZS, IPVT, WORK)
PHIP = TX(1)
THETAP = TX(2)
LANDAP = TX(3)
PHI21 = TX(4)
PHIG = TX(5)
THETAG = TX(6)
ERROR = ((PHI21*36.D02-PHI210*36.D02)-PHI11*36.D02*TN1/TN2)/CNST

CALL PRING2(KS2, G2, E21fx, E21fy, E21fz, E21Ifx, E21Ify, E21Ifz)
CALL PRINPI(KS1, G1, EIIfx, EIIfy, EIIfz, EIIfx, EIIfy, EIIfz, EIIfx, EIIfy, .
              EIIfx, CS2SIG, SN2SIG, SIGM12)
CALL SIGAN2(E21fx, E21fy, E21fz, E21Ifx, E21Ify, E21Ifz, EIIfx, EIIfy, .
              EIIfx, CS2SIG, SN2SIG, SIGM12)
CALL EULER(KS2, G2, KS1, G1, CS2SIG, SN2SIG, IEU)
IF(IEU .EQ. 1) THEN
  WRITE(72, *) 'THERE IS INTERFERENCE'
  GO TO 88888
END IF

CALL ELLIPS(KS2, G2, KS1, G1, CS2SIG, SN2SIG, DEF, ALFA1, .
           AXISL, AXISS, EIIfx, EIIfy, EIIfz)


* XBf, YBf, and ZBf is the direction of the long axis of the ellipse

* CALL PF(XBp, YBp, ZBp, XBf, YBf, ZBf)
ELB1px = B2px + XBp
ELB1pz = B2pz + ZBp
ELB2px = B2px - XBp
ELB2pz = B2pz - ZBp

181
IF(I .EQ. 1 .AND. J .EQ. 1)THEN
  WRITE(9,9000) IJ,PHII/CONST, IJ,ERROR
  IF(IJ .LE. 37)THEN
    WRITE(8,8000) IJ,B2pz, IJ,B2px
    WRITE(7,7000) ELB1pz, ELB1px, ELB2pz, ELB2px
  END IF
ELSE IF(I .EQ. 1 .AND. J .EQ. 2)THEN
  WRITE(29,9000) IJ,PHII/CONST, IJ,ERROR
  IF(IJ .LE. 37)THEN
    WRITE(28,8000) IJ,B2pz, IJ,B2px
    WRITE(27,7000) ELB1pz, ELB1px, ELB2pz, ELB2px
  END IF
ELSE IF(I .EQ. 2 .AND. J .EQ. 1)THEN
  WRITE(29,9000) IJ,PHII/CONST, IJ,ERROR
  IF(IJ .LE. 37)THEN
    WRITE(28,8000) IJ,B2pz, IJ,B2px
    WRITE(27,7000) ELB1pz, ELB1px, ELB2pz, ELB2px
  END IF
ELSE
  WRITE(89,9000) IJ,PHII/CONST, IJ,ERROR
  IF(IJ .LE. 37)THEN
    WRITE(88,8000) IJ,B2pz, IJ,B2px
    WRITE(87,7000) ELB1pz, ELB1px, ELB2pz, ELB2px
  END IF
END IF

PHII=PHII+DPHII

100 CONTINUE

PHII=0.D0

DO 200 IJ=1,60
  CPHII=DCOS(PHII)
  SPhii=DSIN(PHII)
  CALL NONLIN(TCN,14,6,200,TX,TF,TF1,0.1.D-5,AZS,IPVT,WORK)
  PHIP=T(1)
  THEP =T(2)
  LANDA =T(3)
  PHII1 =T(4)
  PHIG =T(5)
  THETAG =T(6)
ERROR=((PHII1*36.D02-PHI21*36.D02)-PHII*36.D02*TN1/TN2)/CONST

182
CALL PRING2(KS2,G2,E21fx,E21fy,E21fz,E211fx,E211fy,E211fz)
CALL PRINP1(KS1,G1,E11fx,E11fy,E11fz,E111fx,E111fy,E111fz)
CALL SIGAN2(E21fx,E21fy,E21fz,E211fx,E211fy,E211fz,E11fx,E11fy,
E11fz,CS2SIG,SN2SIG,SIGM12)
CALL EULER(KS2,G2,KS1,G1,CS2SIG,SN2SIG,IEU)
IF(IEU .EQ. I) THEN
   WRITE(72,*) 'THERE IS INTERFERENCE'
   GO TO 88888
END IF
CALL ELLIPS(KS2,G2,KS1,G1,CS2SIG,SN2SIG,DEF,ALFA1,
AXISL,AXISS,E11fx,E11fy,E11fz)

** XBf, Ybf, and ZBf is the direction of the long axis of the ellipse **

CALL PF(XBp,YBp,ZBp,XBf,YBf,ZBf)
ELB1px=B2px+XBp
ELB1px=B2pz+ZBp
ELB2px=XBp
ELB2pz=ZBp

IF(I .EQ. 1 .AND. J .EQ. 1) THEN
   WRITE(9,9001) IJ,PHII1/CNST, IJ,ERROR
   IF(IJ .LE. 37) THEN
      WRITE(8,8001) IJ,B2pz, IJ,B2px
      WRITE(7,7000) ELBIpz, ELBIpx, ELB2pz, ELB2px
   END IF
ELSE IF(I .EQ. 1 .AND. J .EQ. 2) THEN
   WRITE(79,9001) IJ,PHII1/CNST, IJ,ERROR
   IF(IJ .LE. 37) THEN
      WRITE(78,8001) IJ,B2pz, IJ,B2px
      WRITE(77,7000) ELBIpz, ELBIpx, ELB2pz, ELB2px
   END IF
ELSE IF(I .EQ. 2 .AND. J .EQ. 1) THEN
   WRITE(29,9001) IJ,PHII1/CNST, IJ,ERROR
   IF(IJ .LE. 37) THEN
      WRITE(28,8001) IJ,B2pz, IJ,B2px
      WRITE(27,7000) ELBIpz, ELBIpx, ELB2pz, ELB2px
   END IF
ELSE
   WRITE(89,9001) IJ,PHII1/CNST, IJ,ERROR
   IF(IJ .LE. 37) THEN
      WRITE(88,8001) IJ,B2pz, IJ,B2px
      WRITE(87,7000) ELBIpz, ELBIpx, ELB2pz, ELB2px
   END IF
END IF

PHII1=PHII1-DPHII1
* 200 CONTINUE
* 
99999 CONTINUE
88888 CONTINUE
* 
7000 FORMAT(6X,'EX(1)=',F9.6,/,6X,'EY(1)='F9.6,/,
       6X,'EX(2)=',F9.6,/,6X,'EY(2)='F9.6,/,
       6X,'CALL CURVE(EX,EY,2,0)'
8000 FORMAT(6X,'X0(',I2,')='F9.6,/,6X,'Y0(',I2,')='F15.6)
8001 FORMAT(6X,'X1(',I2,')='F9.6,/,6X,'Y1(',I2,')='F15.6)
9000 FORMAT(6X,'X0(',I2,')='F7.3,/,6X,'Y0(',I2,')='F16.4)
9001 FORMAT(6X,'X1(',I2,')='F7.3,/,6X,'Y1(',I2,')='F16.4)
END
*
* FOR THE DETERMINATION OF MEAN CONTACT POINT
*
SUBROUTINE PCN1(X,F,NE)
IMPLICIT REAL*8(A-H,K-M-Z)
CHARACTER*8 HG
INTEGER NE
REAL*8 X(NE),F(NE),PAR(6)
COMMON/P1/PAR
COMMON/A0/HG
COMMON/A1/p11,p12,p13,p21,p22,p23,p31,p32,p33,p1,p2,p3
COMMON/A2/SG,CSR2,OG,SNPSIG,CSPSIG
COMMON/A3/TND1,TND2,RITAG
THETAG=X(1)
CSTHEG=DCOS(THETAG)
SNTHEG=DSIN(THETAG)
IF(HG .EQ. 'L')THEN
   UG=PAR(1)-SG*(PAR(2)*SNTHEG+PAR(3))/(CSR2*DSIN(THETAG-QG))
ELSE
   UG=PAR(1)-SG*(PAR(2)*SNTHEG+PAR(3))/(CSR2*DSIN(THETAG+QG))
END IF
Bcx=PAR(4)-UG*CSPSIG
Bcy=UG*SNPSIG*SNTHEG
Bcz=UG*SNPSIG*CSTHEG
CALL TRCOOR(Bpx,Bpy,Bpz,
   p11,p12,p13,p21,p22,p23,p31,p32,p33,p1,p2,p3,
   Bcx,Bcy,Bcz)
XM=Bpz*(TND1-TND2)/2.D00
F(1)=Bpx-XM
END
*
* FOR THE DETERMINATION OF MEAN CONTACT POINT
*
SUBROUTINE PCN2(X,F,NE)
IMPLICIT REAL*8(A-H,K-M-Z)
CHARACTER*8 HG
INTEGER NE
REAL*8 X(NE),F(NE),PAR(6)
COMMON/P1/PAR
COMMON/A0/HG
COMMON/A1/p11,p12,p13,p21,p22,p23,p31,p32,p33,p1,p2,p3
COMMON/A2/SG,CSRT2,QG,SNPSIG,CSPSIG
COMMON/A3/TND1,TND2,RITAG
COMMON/A4/CSD2,SND2,CSPIT2,SNPIT2
COMMON/A5/CSQG,SNQG,THETAG

PHIG=X(1)
CSPHIG=DCOS(PHIG)
SNPHIG=DSIN(PHIG)

IF(HG .EQ. 'L') THEN
UG=PAR(1)-SG*(PAR(2)+PAR(3)*DSIN(QG-PHIG)) /
• (CSRT2*DSIN(THETAG-QG+PHIG))
ELSE
UG=PAR(1)-SG*(PAR(2)+PAR(3)*DSIN(QG-PHIG)) /
• (CSRT2*DSIN(THETAG+QG-PHIG))
END IF

Bcx=PAR(4)-UG*CSPSIG
Bcy=UG*PAR(5)
Bcz=UG*PAR(6)

* Mmc=Mms*Msc

* Mpc=Mpm*Mmc

CALL COMBI(m11,m12,m13,m21,m22,m23,m31,m32,m33,m1,m2,m3)
. 1.D00,0.D00,0.D00,0.D00,CSPHIG,SNPHIG,0.D00,-SNPHIG,CSPHIG,
. 0.D00,0.D00,0.D00,
. 1.D00,0.D00,0.D00,0.D00,CSD2,-SNQG,0.D00,SNQG,CSQG,
. 0.D00,-SG*SNQG,SG*CSQG)

* Mmc=Mms*Msc

CALL COMBI(p11,p12,p13,p21,p22,p23,p31,p32,p33,p1,p2,p3)
. CSD2,0.D00,-SND2,0.D00,1.D00,0.D00,SNQG,0.D00,0.D00,
. 0.D00,0.D00,0.D00,
. m11,m12,m13,m21,m22,m23,m31,m32,m33,m1,m2,m3)

* Mpc=Mpm*Mmc

CALL COMBI(p11,p12,p13,p21,p22,p23,p31,p32,p33,p1,p2,p3)
. CSD2,0.D00,-SND2,0.D00,1.D00,0.D00,SNQG,0.D00,0.D00,
. 0.D00,0.D00,0.D00,
. m11,m12,m13,m21,m22,m23,m31,m32,m33,m1,m2,m3)
END IF

CALL TRCOOR(Bpx,Bpy,Bpz,
. p11,p12,p13,p21,p22,p23,p31,p32,p33,p1,p2,p3,
. Bcx,Bcy,Bcz)
Xm = Bpz \ast (TND1 - TND2) / 2.00
F(1) = Bpx - Xm
RETURN
END

* FOR THE DETERMINATION OF COORDINATES AND Normals OF CONTACT POINTS

SUBROUTINE TCN (TX, TF, NE)
IMPLICIT REAL*8 (A-H,K,M-Z)
REAL*8 LANDAP, LM
CHARACTER*8 HG
INTEGER NE
DIMENSION TX(NE), TF(NE), TPAR(19)
COMMON/T1/TPAR
COMMON/A0/HG
COMMON/A2/SG, CSRT2, QG, SNPSIG, CSPSIG
COMMON/A4/CSD2, SND2, CSPIT2, SNPIT2
COMMON/B1/CSPH11, SNPH11, SP, EM, LM, CSRT1, CSD1, SND1, CSLANP, SNLANP
COMMON/B2/CSPIT1, SNPIT1, MP1, MG2, QP
COMMON/B3/B2fx, B2fy, B2fz
COMMON/B4/CSPH2, SNPH2, CSPH21, SNPH21
COMMON/B5/XCR, ZCR
COMMON/C1/UG, CSTAUG, SNTAUG
COMMON/C2/N2fx, N2fy, N2fz
COMMON/D1/CSTAUP, SNTAUP
COMMON/F1/PHIGO
COMMON/G1/DA, DV
J = IDINT (TPAR(16))
PHIP = TX(1)
THETAP = TX(2)
LANDAP = TX(3)
PHI21 = TX(4)
PHIG = TX(5)
THETAG = TX(6)
CSPHIP = DCOS (PHIP)
SNPHIP = DSIN (PHIP)
CSTHEP = DCOS (THETAP)
SNTHEP = DSIN (THETAP)
CSLANP = DCOS (LANDAP)
SNLANP = DSIN (LANDAP)
CSPH21 = DCOS (PHI21)
SNPH21 = DSIN (PHI21)
CSPHIG = DCOS (PHIG)
SNPHIG = DSIN (PHIG)
CSTHEG = DCOS (THETAG)
SNTHEG = DSIN (THETAG)
PHI2 = (PHIG - PHIGO) / MG2
PHI1 = PHIP / MP1
CSPH2 = DCOS (PHI2)
SNPH2 = DSIN (PHI2)
CSPH1 = DCOS (PHI1)
SNPH1 = DSIN (PHI1)
IF (HG .EQ. 'L') THEN
  TAUP = THETAP + QP - PHIP
ELSE
  TAUP = THETAP - QP + PHIP
END IF

CSTAUP = DCOS (TAUP)
SNTAUP = DSIN (TAUP)

IF (HG .EQ. 'L') THEN
  TAUG = THETAG - QG + PHIG
ELSE
  TAUG = THETAG + QG - PHIG
END IF

CSTAUG = DCOS (TAUG)
SNTAUG = DSIN (TAUG)

CSQPHP = DCOS (QP - PHIP)
SNQPHG = DSIN (QP - PHIP)
CSQPHG = DCOS (QG - PHIG)
SNQPHG = DSIN (QG - PHIG)

* * * LEFT-HAND GEAR * * *

IF (HG .EQ. 'L') THEN
  UG = TPAR (1) - SG * (TPAR (2) * SNTHEG - SNQPHG * TPAR (3)) / (CSRT2 * SNTAUG)
  B2py = UG * SNPSIG * SNTAUG - SG * SNQPHG
ELSE
  UG = TPAR (1) - SG * (TPAR (2) * SNTHEG + SNQPHG * TPAR (3)) / (CSRT2 * SNTAUG)
  B2py = UG * SNPSIG * SNTAUG + SG * SNQPHG
END IF

B2px = CSD2 * (TPAR (4) - UC * CSPSIG) - SND2 * (UG * SNPSIG * CSTAUG + SG * CSQPHG)
B2pz = SND2 * (TPAR (4) - UC * CSPSIG) + CSD2 * (UG * SNPSIG * CSTAUG + SG * CSQPHG)
N2px = TPAR (5) - TPAR (6) * CSTAUG
N2py = CSPSIG * SNTAUG
N2pz = TPAR (7) + TPAR (8) * CSTAUG

* [Mwp] = [Mwa] [Map]

* * *
END IF
CALL TRCOORD(B2wx,B2wy,B2wz,
    wp11, wp12, wp13, wp21, wp22, wp23, wp31, wp32, wp33, wp1, wp2, wp3,
    B2px, B2py, B2pz)
CALL TRCOORD(N2wx,N2wy,N2wz,
    wp11, wp12, wp13, wp21, wp22, wp23, wp31, wp32, wp33, 0.00, 0.00, 0.00,
    N2px, N2py, N2pz)

* [Mfw] = [Mfa] [Maw]

* IF(HG .EQ. 'L') THEN
    CALL COMBI(fw11, fw12, fw13, fw21, fw22, fw23, fw31, fw32, fw33,
        fw1, fw2, fw3,
        CSPIT2, 0.00, -SNPIT2, 0.00, 1.00, 0.00, SNPIT2, 0.00, CSPIT2,
        0.00, 0.00, 0.00,
        CSPH21, -SNPH21, 0.00, SNPH21, CSPH21, 0.00, 0.00, 0.00, 1.00,
        0.00, 0.00, 0.00)
ELSE
    CALL COMBI(fw11, fw12, fw13, fw21, fw22, fw23, fw31, fw32, fw33,
        fw1, fw2, fw3,
        CSPIT2, 0.00, -SNPIT2, 0.00, 1.00, 0.00, SNPIT2, 0.00, CSPIT2,
        0.00, 0.00, 0.00,
        CSPH21, SNPH21, 0.00, -SNPH21, CSPH21, 0.00, 0.00, 0.00, 1.00,
        0.00, 0.00, 0.00)
END IF
CALL TRCOORD(B2fx,B2fy,B2fz,
    fw11, fw12, fw13, fw21, fw22, fw23, fw31, fw32, fw33, fw1, fw2, fw3,
    B2wx, B2wy, B2wz)
CALL TRCOORD(N2fx,N2fy,N2fz,
    fw11, fw12, fw13, fw21, fw22, fw23, fw31, fw32, fw33, 0.00, 0.00, 0.00,
    N2wx, N2wy, N2wz)

* PINION

* IF(HG .EQ. 'L') THEN
    Blpy=(ZCR+TPAR(17)*SNLANP)*SNTAUP+SP*SNQPHP-EM
ELSE
    Blpy=(ZCR+TPAR(17)*SNLANP)*SNTAUP-SP*SNQPHP+EM
END IF
Blpx=(XCR+TPAR(17)*CSLANP)*CSDI-((ZCR+TPAR(17)*SNLANP)*CSTAUP+
        SP*CSQPHP+LM)*SN1
Blpz=(XCR+TPAR(17)*CSLANP)*SN1+((ZCR+TPAR(17)*SNLANP)*CSTAUP+
        SP*CSQPHP+LM)*CSD1
IF(J .EQ. 2) THEN
    NIpx=CSLANP*CSD1-SNLANP*SN1*CSTAUP
    NIpy=SNLANP*SNTAUP
    NIpz=CSLANP*SN1+SNLANP*CSD1*CSTAUP
ELSE
    NIpx=-CSLANP*CSD1-SNLANP*SN1*CSTAUP
    NIpy=-SNLANP*SNTAUP
    NIpz=-CSLANP*SN1-SNLANP*CSD1*CSTAUP
END IF
* [Mwp] = [Mwa] [Map]

* IF(HG .EQ. 'L') THEN
  CALL COMBI(wp11, wp12, wp13, wp21, wp22, wp23, wp31, wp32, wp33,
             wp1, wp2, wp3,
             CSPHI, -SNPHI, 0.D00, SNPH1, CSPH1, 0.D00, 0.D00, 1.D00,
             0.D00, 0.D00, 0.D00,
             CSPIT1, 0.D00, -SNPIT1, 0.D00, 1.D00, 0.D00, -SNPIT1, 0.D00, CSPIT1,
             0.D00, 0.D00, 0.D00)
ELSE
  CALL COMBI(wp11, wp12, wp13, wp21, wp22, wp23, wp31, wp32, wp33,
             wp1, wp2, wp3,
             CSPHI, SNPHI, 0.D00, -SNPHI, CSPH1, 0.D00, 0.D00, 1.D00,
             0.D00, 0.D00, 0.D00,
             CSPIT1, 0.D00, SNPIT1, 0.D00, 1.D00, 0.D00, -SNPIT1, 0.D00, CSPIT1,
             0.D00, 0.D00, 0.D00)
END IF

CALL TRCOOR(B1wx, B1wy, B1wz,
             wp11, wp12, wp13, wp21, wp22, wp23, wp31, wp32, wp33, wp1, wp2, wp3,
             B1px, B1py, B1pz)

CALL TRCOOR(N1wx, N1wy, N1wz,
             wp11, wp12, wp13, wp21, wp22, wp23, wp31, wp32, wp33, 0.D00, 0.D00, 0.D00,
             N1px, N1py, N1pz)

* [Mpw] = [Mpa] [Maw]

* IF(HG .EQ. 'L') THEN
  CALL COMBI(pw11, pw12, pw13, pw21, pw22, pw23, pw31, pw32, pw33,
             pw1, pw2, pw3,
             CSPIT1, 0.D00, -SNPIT1, 0.D00, 1.D00, 0.D00, SNPIT1, 0.D00, CSPIT1,
             0.D00, 0.D00, 0.D00,
             CSPHI1, SNPH11, 0.D00, -SNPHII, CSPHII, 0.D00, 0.D00, 1.D00,
             0.D00, 0.D00, 0.D00)
ELSE
  CALL COMBI(pw11, pw12, pw13, pw21, pw22, pw23, pw31, pw32, pw33,
             pw1, pw2, pw3,
             CSPIT1, 0.D00, -SNPIT1, 0.D00, 1.D00, 0.D00, SNPIT1, 0.D00, CSPIT1,
             0.D00, 0.D00, 0.D00,
             CSPHI1, -SNPH11, 0.D00, SNPHII, CSPHII, 0.D00, 0.D00, 1.D00,
             0.D00, 0.D00, 0.D00)
END IF

CALL TRCOOR(B1px, B1py, B1pz,
             pw11, pw12, pw13, pw21, pw22, pw23, pw31, pw32, pw33, pw1, pw2, pw3,
             B1wx, B1wy, B1wz)

CALL TRCOOR(N1px, N1py, N1pz,
             pw11, pw12, pw13, pw21, pw22, pw23, pw31, pw32, pw33, 0.D00, 0.D00, 0.D00,
             N1wx, N1wy, N1wz)

B1fx = -B1px + DA_SNPIT1
B1fy = -B1py + DV
B1fz = B1pz + DA*CSPIT1
N1fx = -N1px
N1fy = -N1py
N1fz = N1pz

189
IF (HG .EQ. 'L') THEN
  !
  TF (1) = (TPAR (9) * SNTAUP + TPAR (10) * SNQPHP - TPAR (11)) * CSLANP -
        (TPAR (12) * SNTAUP - TPAR (13) * SNTHEP + TPAR (14) * CSTAUP + TPAR (15) *
        SNTAUP) * SNLANP
ELSE
  TF (1) = (TPAR (9) * SNTAUP - TPAR (10) * SNQPHP + TPAR (11)) * CSLANP -
        (TPAR (12) * SNTAUP - TPAR (13) * SNTHEP - TPAR (14) * CSTAUP + TPAR (15) *
        SNTAUP) * SNLANP
END IF

TF (2) = B2fx - B1fx
TF (3) = B2fy - B1fy
TF (4) = B2fz - B1fz
TF (5) = N2fx - N1fx
TF (6) = N2fy - N1fy
END

* FOR THE DETERMINATION OF GEAR PRINCIPAL CURVATURES AND DIRECTIONS *

SUBROUTINE PRING2(KSG, GG, EGIfx, EGIfy, EGIfz, EGIIfx, EGIIfy, EGIIfz)
IMPLICIT REAL*8(A-H, K, M-Z)
COMMON/A2/SG, CSRT2, QG, SNPSIG, CSPSIG
COMMON/A3/TND1, TND2, RITAG
COMMON/A4/CSD2, SND2, CSPIT2, SNPIT2
COMMON/B2/CSPIT1, SNPIT1, MP1, MG2, QP
COMMON/B3/B2fx, B2fy, B2fz
COMMON/B4/CSPH2, SNPH2, CSPH21, SNPH21
COMMON/C1/UG, CSTAUG, SNTAUG
COMMON/C2/N2fx, N2fy, N2fz
COMMON/F1/PHIGO
KCI = -CSPSIG / (UG * SNPSIG)
KCI I = 0. DO0
ECIfx = SND2 * SNTAUG
ECIfy = CSTAUG
ECIfz = -CSD2 * SNTAUG
ECIIfx = -CSD2 * CSPSIG - SND2 * SNPSIG * CSTAUG
ECIIfy = SNPSIG * SNTAUG
ECIIfz = -SNPSIG * CSPSIG + CSD2 * SNPSIG * CSTAUG
WGfx = -SNPIT2
WGfy = 0. DO0
WGfz = CSPIT2
WCfx = -MG2 * CSD2
WCfy = 0. DO0
WCfz = -MG2 * SND2
WGCfx = WGfx - WCfx
WGCfy = WGfy - WCfy
WGCfz = WGfz - WCfz
CALL CROSS(VTGfx, VTGfy, VTGfz, WGfx, WGfy, WGfz, B2fx, B2fy, B2fz)
CALL CROSS(VTCEFx, VTCEFy, VTCEFz, WCfx, WCfy, WCfz, B2fx, B2fy, B2fz)
VTGCEFx = VTGfx - VTCEFx
VTGCEFy = VTGfy - VTCEFy
VTGCEFz = VTGfz - VTCEFz
CALL DOT(VCI, ECIfx, ECIfy, ECIfz, VTGCEFx, VTGCEFy, VTGCEFz)
CALL DOT(VCI I, ECIIfx, ECIIfy, ECIIfz, VTGCEFx, VTGCEFy, VTGCEFz)
CALCULATE A13, A23, A33

CALL DET(DETI,WGCfx,WGCfy,WGCfz,N2fx,N2fy,N2fz,ECIfx,ECIfy,ECIfz)
A13=-KCI*VC1-DETI
CALL DET(DETII,WGCfx,WGCfy,WGCfz,N2fx,N2fy,N2fz,ECIIfx,ECIIfy,ECIIfz)
A23=-KCI*VC1-DETI
CALL DET(DET3,N2fx,N2fy,N2fz,WGCfx,WGCfy,WGCfz,VTGCfx,VTGCfy,VTGCfz)
CALL CROSS(Cx,Cy,Cz,WGfx,0.D00,WGfz,VTcfy,VTcfy,VTcfz)
CALL CROSS(Dx,Dy,Dz,WCfx,0.D00,WCfz,VTgfx,VTgfy,VTgfz)
CALL DOT(DET45,N2fx,N2fy,N2fz,Cx-Dx,Cy-Dy,Cz-Dz)
A33=KCI*VC1*VC1+KCI*VC1*VC1-DET3-DET45

CALCULATE SIGMA

P=A23*A23-A13*A13+(KCI-KCI)*A33
SIGMA2=DATAN(2.D00*A13*A23/P)
SIGMA=0.5D00*SIGMA2

CALCULATE KGI AND KGII

GG=P/(A33*DCOS(SIGMA2))
KSG=KCI+KCI-(A13*A13+A23*A23)/A33
KGI=(KSG+GG)/2.D00
KGII=(KSG-GG)/2.D00

CALCULATE EGI AND EGII

CALL ROTATE(EGIfx,EGIfy,EGIfz,ECIfx,ECIfy,ECIfz,-SIGMA,N2fx,N2fy,
.N2fz)
CALL ROTATE(EGIIfx,EGIIfy,EGIIfz,EGIfx,EGIfy,EGIfz,RITAG,
.N2fx,N2fy,N2fz)
END

FOR THE DETERMINATION OF PINION PRINCIPAL CURVATURES AND DIRECTIONS

SUBROUTINE PRINPI(KSP,GP,EPIfx,EPIfy,EPIfz,EPIIfx,EPIIfy,EPIIfz)
IMPLICIT REAL*8(A-H,K,M-Z)
REAL*8 LM,TPAR(19)
COMMON/T1/TPAR
COMMON/A3/TND1,TND2,RITAG
COMMON/B1/CSPHI1,SNPHI1,SP,EM,LM,CSRT1,CSD1,SND1,CSLANP,SNLANP
COMMON/B2/CSPIT1,SNPIT1,MP1,MG2,QP
COMMON/B3/B2fx,B2fy,B2fz
COMMON/B5/XCR,ZCR
COMMON/C2/N2fx,N2fy,N2fz
COMMON/D1/CSTAUP,SNTAUP
J=IDINT(TPAR(16))
R=TPAR(17)
IF(J .EQ. 1)THEN
KCI=SNLANP/(ZCR+R*SNLANP)
KCII = 1.00/R
ELSE
KCI = -SNLANP/(ZCR + R*SNLANP)
KCII = -1.00/R
END IF

ECIfx = SND1*SNTAUP
ECIfy = CSTAUP
ECIfz = CSD1*SNTAUP
ECIIfx = CSD1*SNLANP + SND1*CSLANP*CSTAUP
ECIIfy = -CSLANP*SNTAUP
ECIIfz = -SND1*SNLANP + CSD1*CSLANP*CSTAUP
IF (J .EQ. 2) THEN
ECIIfx = -ECIIfx
ECIIfy = -ECIIfy
ECIIfz = -ECIIfz
END IF

WPfx = -SNPIT1
WPfy = 0.00
WPfz = -CSPIT1
WCfx = -MPI*CSD1
WCfy = 0.00
WCfz = MPI*SND1
WPCfx = WPfx - WCfx
WPCfy = WPfy - WCfy
WPCfz = WPfz - WCfz
CALL CROSS(VTPfx, VTPfy, VTPfz, WPfx, WPfy, WPfz, B2fx, B2fy, B2fz)
CALL CROSS(VTCIfx, VTCIfy, VTCIfz, WCfx, WCfy, WCfz, B2fx, B2fy, B2fz)
CALL CROSS(VTC2fx, VTC2fy, VTC2fz, LM*SND1, EM, LM*CSD1, WCfx, WCfy, WCfz)
VTCfx = VTCIfx + VTC2fx
VTCfy = VTCIfy + VTC2fy
VTCfz = VTCIfz + VTC2fz
VTPCfx = VTPfx - VTCfx
VTPCfy = VTPfy - VTCfy
VTPCfz = VTPfz - VTCfz
CALL DOT(VCI, ECIfx, ECIfy, ECIfz, VTPCfx, VTPCfy, VTPCfz)
CALL DOT(VCI, ECIIfx, ECIIfy, D2Ifz, VTPCfx, VTPCfy, VTPCfz)

* * CALCULATE A13, A23, A33 *
* *
CALL DET(DETI, WPCfx, WPCfy, WPCfz, N2fx, N2fy, N2fz, ECIfx, ECIfy, ECIfz)
A13 = -KCI*VCI-DETI
CALL DET(DETII, WPCfx, WPCfy, WPCfz, N2fx, N2fy, N2fz, ECIIfx, ECIIfy, ECIIfz)
A23 = -KCII*VCI1-DETI
CALL DET(DET3, N2fx, N2fy, N2fz, WPCfx, WPCfy, WPCfz, VTPCfx, VTPCfy, VTPCfz)
CALL CROSS(Cx, Cy, Cz, WPfx, 0.00, WPfz, VTCfx, VTCfy, VTCfz)
CALL CROSS(Dx, Dy, Dz, WCfx, 0.00, WCfz, VTPfx, VTPfy, VTPfz)
CALL DOT(DET45, N2fx, N2fy, N2fz, Cx - Dx, Cy - Dy, Cz - Dz)
A33 = KCI*VCI1*VCI + KCII*VCI1*VCI1-DETI3-DETI45

192
* CALCULATE SIGMA

\[
P = A_{23}^2 - A_{13}^2 A_{13} + (KCI - KCI) A_{33}
\]
\[
SIGMA_2 = \text{DATAN}(2.00 \times A_{13} A_{23} / P)
\]
\[
SIGMA = 0.5 \times SIGMA_2
\]

* CALCULATE KPI AND KPII

\[
GP = P / (A_{33} \times \text{DCOS}(SIGMA_2))
\]
\[
KSP = KCI + KCI - (A_{13}^2 A_{13} + A_{23}^2 A_{23}) / A_{33}
\]
\[
KPI = (KSP + GP) / 2.00
\]
\[
KPII = (KSP - GP) / 2.00
\]

* CALCULATE EPI AND EPII

\[
\text{CALL ROTATE}(EPI_{fx}, EPI_{fy}, EPI_{fz}, ECI_{fx}, ECI_{fy}, ECI_{fz}, -SIGMA, N_2fx, N_2fy, N_2fz)
\]
\[
\text{CALL ROTATE}(EPIII_{fx}, EPIII_{fy}, EPIII_{fz}, EPI_{fx}, EPI_{fy}, EPI_{fz}, \text{RITAG}, N_2fx, N_2fy, N_2fz)
\]
* FOR THE DETERMINATION OF THE ANGLE BETWEEN GEAR PRINCIPAL DIRECTIONS
* AND PINION PRINCIPAL DIRECTIONS

* SUBROUTINE SIGAN2(EGIfx, EGIfy, EGIIfx, EGIIfy, EGIIfz, EPIfx, EPIfy, EPIfz, CS2SIG, SN2SIG, SIGMPG)

IMPLICIT REAL*8(A-H,K,M-Z)
CALL DOT(CSSIG, EPIfx, EPIfy, EPIfz, EGIfx, EGIfy, EGIIfz)
CALL DOT(SNSIG, EPIfx, EPIfy, EPIfz, -EGIIfx, -EGIIfy, -EGIIfz)
SIGM2 = 4.00 \times \text{DATAN}(SNSIG / (1.00 + CSSIG))
SIGMPG = 0.5 \times SIGM2
CS2SIG = \text{DCOS}(SIGM2)
SN2SIG = \text{DSIN}(SIGM2)
END

* FOR THE DETERMINATION OF CONTACT ELLIPS

* SUBROUTINE ELLIPS(KSG, GG, KSP, GP, CS2SIG, SN2SIG, DEF, ALFAP, AXISL, AXISS, EPIfx, EPIfy, EPIfz)

IMPLICIT REAL*8(A-H,K,M-Z)
COMMON/A3/TND1, TND2, RITAG
COMMON/C2/N2fx, N2fy, N2fz
COMMON/EI/XBf, Ybf, Zbf
D = DSQRT(GP*GP - 2.00*GP*GG*CS2SIG + GG*GG)
CS2AFP = (GP - GG*CS2SIG) / D
SN2AFP = GG*SN2SIG / D
ALFAP = \text{DATAN}(SN2AFP / (1.00 + CS2AFP))
A = 25.00 \times DABS(KSP - KSG - D)
B = 25.00 \times DABS(KSP - KSG + D)
IF(KSG .LT. KSP) THEN
AXISL = DSQRT(DEF/A)
AXISS = DSQRT(DEF/B)
\text{CALL ROTATE}(XBf, Ybf, Zbf, EPIfx, EPIfy, EPIfz, \text{RITAG} - ALFAP, N_2fx,
. N2fy,N2fz)
ELSE
AXISL=DSQRT(DEF/B)
AXISS=DSQRT(DEF/A)
CALL ROTATE(XBf,YBf,ZBf,EPIfx,EPIfy,EPIfz,-ALFAP,N2fx,N2fy,
. N2fz)
END IF
XBf=AXISL*XBF
YBf=AXISL*YBF
ZBf=AXISL*ZBF
END

* COORDINATE TRANSFORMATION FOR F TO P
*
IMPLICIT REAL*8(A-H,K,M-Z)
COMMON/A4/CSD2,SND2,CSPIT2,SNPIT2
COMMON/B4/CSPH2,SNPH2,CSPH21,SNPH21

* [Mt] = [Mta] [MaF]
*
CALL COMBI(t11,t12,t13,t21,t22,t23,t31,t32,t33,t1,t2,t3,
. CSPH21,SNPH21,0.D00,-SNPH21,CSPH21,0.D00,0.D00,0.D00,1.D00,
. 0.D00,0.D00,0.D00,
. CSPIT2,0.D00,SNPIT2,0.D00,1.D00,0.D00,-SNPIT2,0.D00,CSPIT2,
. 0.D00,0.D00,0.D00)
CALL TRCOOR(B2wx,B2wy,B2wz,
. t11,t12,t13,t21,t22,t23,t31,t32,t33,t1,t2,t3,

* [Mpt] = [Mpa] [Mpt]
*
CALL COMBI(p11,p12,p13,p21,p22,p23,p31,p32,p33,p1,p2,p3,
. CSPIT2,0.D00,-SNPIT2,0.D00,1.D00,0.D00,SNPIT2,0.D00,CSPIT2,
. 0.D00,0.D00,0.D00,
. CSPH2,-SNPH2,0.D00,SNPH2,CSPH2,0.D00,0.D00,0.D00,1.D00,
. 0.D00,0.D00,0.D00)
CALL TRCOOR(B2px,B2py,B2pz,
. p11,p12,p13,p21,p22,p23,p31,p32,p33,p1,p2,p3,
END

* USING EULER FORMULA TO DETERMINATION SURFACE INTERFERENCE
*
SUBROUTINE EULER(KSG,GG,KSP,GP,CS2SIG,SN2SIG,IEU)
IMPLICIT REAL*8(A-H,K,M-Z)
A=KSG-KSP
B=DSQRT((GG-GP*CS2SIG)**2+(GP*SN2SIG)**2)
KR1=(A+B)/2.D00
KR2=(A-B)/2.D00
IF(KR1*KR2 .LT. 0.D00) THEN
IEU=1
ELSE
```
IEU=0
END IF
END

* DETERMINANT

SUBROUTINE DET(S, A, B, C, D, E, F, G, H, P)
IMPLICIT REAL*8(A-H, K, M-Z)
RETURN
END

* COORDINATE TRANSFORMATION

SUBROUTINE TRCOOR(XN, YN, ZN, R11, R12, R13, R21, R22, R23, R31, R32, R33,
T1, T2, T3, XP, YP, ZP)
IMPLICIT REAL*8(A-H, O-Z)
XN=R11*XP+R12*YP+R13*ZP+T1
YN=R21*XP+R22*YP+R23*ZP+T2
ZN=R31*XP+R32*YP+R33*ZP+T3
RETURN
END

* MULTIPLICATION OF TWO TRANSFORMATION MATRICES

SUBROUTINE COMBI(C11, C12, C13, C21, C22, C23, C31, C32, C33, C1, C2, C3,
A11, A12, A13, A21, A22, A23, A31, A32, A33, A1, A2, A3,
B11, B12, B13, B21, B22, B23, B31, B32, B33, B1, B2, B3)
IMPLICIT REAL*8(A-H, M, N, O-Z)
C11=B31*A13+B21*A12+B11*A11
C12=B32*A13+B22*A12+B12*A11
C13=B33*A13+B23*A12+B13*A11
C21=B31*A23+B21*A22+B11*A21
C22=B32*A23+B22*A22+B12*A21
C23=B33*A23+B23*A22+B13*A21
C31=B31*A33+B21*A32+B11*A31
C32=B32*A33+B22*A32+B12*A31
C33=B33*A33+B23*A32+B13*A31
C1=B3*A13+B2*A12+B1*A11+A1
C3=B3*A33+B2*A32+B1*A31+A3
RETURN
END

* DOT OF TWO VECTORS

SUBROUTINE DOT(V, X1, Y1, Z1, X2, Y2, Z2)
IMPLICIT REAL*8(A-H, O-Z)
V=X1*X2+Y1*Y2+Z1*Z2
RETURN
END

* CROSS OF TWO VECTORS
```
SUBROUTINE CROSS(X, Y, Z, A, B, C, D, E, F)
IMPLICIT REAL*8(A-H, O-Z)
X = B*F - C*E
Y = C*D - A*F
Z = A*E - B*D
RETURN
END

* ROTATION A VECTOR ABOUT ANOTHER VECTOR *

SUBROUTINE ROTATE(XN, YN, ZN, XP, YP, ZP, THETA, UX, UY, UZ)
IMPLICIT REAL*8(A-H, O-Z)
CT = DCOS(THETA)
ST = DSIN(THETA)
VT = 1.D00 - CT
RI1 = UX*UX*VT + CT
RI2 = UX*UY*VT - UZ*ST
RI3 = UX*UZ*VT + UY*ST
R21 = UX*UY*VT + UZ*ST
R22 = UY*UY*VT + CT
R23 = UY*UZ*VT - UX*ST
R31 = UX*UZ*VT - UY*ST
R32 = UY*UZ*VT + UX*ST
R33 = UZ*UZ*VT + CT
CALL TRCOORD(XN, YN, ZN, RI1, RI2, RI3, R21, R22, R23, R31, R32, R33,
. 0.D00, 0.D00, 0.D00,
. XP, YP, ZP)
RETURN
END

***** SUBROUTINE NOLIN *****

SUBROUTINE NONLIN(FUNC, NSIG, NE, NC, X, Y, YI, DELTA, A, IPVT, WORK)
IMPLICIT REAL*8(A-H, O-Z)
DIMENSION X(NE), Y(NE), YI(NE), A(NE, NE), IPVT(NE), WORK(NE)
EXTERNAL FUNC

NDIM = NE
EPSI = 1.D00/10.D00**NSIG
CALL NONLIO(FUNC, EPSI, NE, NC, X, DELTA, NDIM, A, Y, YI, WORK, IPVT)
RETURN
END

***** SUBROUTINE NOLINO *****

SUBROUTINE NONLIO(FUNC, EPSI, NE, NC, X, DELTA, NDIM, A, Y, YI, WORK, IPVT)
IMPLICIT REAL*8(A-H, O-Z)
DIMENSION X(NE), Y(NE), YI(NE), A(NDIM, NE)
EXTERNAL FUNC

* NC: # OF COUNT TIMES
DO 5 I = 1, NC
CALL FUNC(X, Y, NE)
* NE: # OF EQUATIONS

196
DO 15 J=1,NE
   IF (DABS(Y(J)).GT.EPSI) GO TO 25
15 CONTINUE
   GO TO 105
25 DO 35 J=1,NE
35 Y1(J)=Y(J)
   DO 45 J=1,NE
      DIFF=DABS(X(J))*DELTA
      IF (DABS(X(J)).LT.1.D-12) DIFF=DELTA
      XMAM=X(J)
      X(J)=X(J)-DIFF
      CALL FUNC(X,Y,NE)
      X(J)=XMAM
   DO 55 K=1,NE
      A(K,J)=(Y1(K)-Y(K))/DIFF
55 CONTINUE
   CONTINUE
45 CONTINUE
   DO 65 J=1,NE
65 Y(J)=-Y1(J)
   CALL DECOMP (NDIM,NE,A,COND,IPVT,WORK)
   CALL SOLVE (NDIM,NE,A,Y,IPVT)
   DO 75 J=1,NE
      X(J)=X(J)+Y(J)
75 CONTINUE
5 CONTINUE
105 RETURN
END

***** SUBROUTINE DECOMP *****

SUBROUTINE DECOMP (NDIM,N,A,COND,IPVT,WORK)
IMPLICIT REAL*8(A-H,O-Z)
DIMENSION A(NDIM,N) ,WORK (N), IPVT (N)

* DECOMPOSES A REAL MATRIX BY GAUSSIAN ELIMINATION,
AND ESTIMATES THE CONDITION OF THE MATRIX.

* COMPUTER METHODS FOR MATHEMATICAL COMPUTATIONS-, BY G. E. FORSYTHE,
   M. A. MALCOLM, AND C. B. MOLER (PRENTICE-HALL, 1977)

* USE SUBROUTINE SOLVE TO COMPUTE SOLUTIONS TO LINEAR SYSTEM.

* INPUT..
* NDIM = DECLARED ROW DIMENSION OF THE ARRAY CONTAINING  A
* N    = ORDER OF THE MATRIX
* A    = MATRIX TO BE TRIANGULARIZED

* OUTPUT..
* A CONTAINS AN UPPER TRIANGULAR MATRIX  U AND A PREMUTED
   VERSION OF A LOWER TRIANGULAR MATRIX  I-L  SO THAT
(PERMUTATION MATRIX) * A = L * U

COND = AN ESTIMATE OF THE CONDITION OF A.
FOR THE LINEAR SYSTEM A \* X = B, CHANGES IN A AND B
MAY CAUSE CHANGES COND TIMES AS LARGE IN X.
IF COND + 1.0 .EQ. COND, A IS SINGULAR TO WORKING
PRECISION. COND IS SET TO 1.0D+32 IF EXACT
SINGULARITY IS DETECTED.

IPVT = THE PIVOT VECTOR
IPVT(K) = THE INDEX OF THE K-TH PIVOT ROW
IPVT(N) = (-1)**(NUMBER OF INTERCHANGES)

WORK SPACE. THE VECTOR WORK MUST BE DECLARED AND INCLUDED
IN THE CALL. ITS INPUT CONTENTS ARE IGNORED.
ITS OUTPUT CONTENTS ARE USUALLY UNIMPORTANT.

THE DETERMINANT OF A CAN BE OBTAINED ON OUTPUT BY
DET(A) = IPVT(N) * A(1,1) * A(2,2) * ... * A(N,N)

IPVT(N) = 1
IF (N .EQ. 1) GO TO 150
NM1 = N - 1

ANORM = 0.0
DO 20 J = 1, N
   T = 0.0
   DO 10 I = 1, N
   10   T = T + DABS(A(I, J))
   IF (T .GT. ANORM) ANORM = T
   20 CONTINUE

DO 70 K = 1, NM1
   KP1 = K + 1

M = K
DO 30 I = KP1, N
   IF (DABS(A(I, K)) .GT. DABS(A(M, K))) M = I
   30 CONTINUE

IPVT(K) = M
IF (M .NE. K) IPVT(N) = -IPVT(N)
T = A(M, K)
A(M, K) = A(K, K)
A(K, K) = T

SKIP THE ELIMINATION STEP IF PIVOT IS ZERO.
IF (T .EQ. 0.0) GO TO 70

DO 40 I = KP1, N
   A(I, K) = -A(I, K) / T
   40 CONTINUE

INTERCHANGE AND ELIMINATE BY COLUMNS.
T = A(M, J)
A(M, J) = A(K, J)
A(K, J) = T
IF (T.EQ.0.D0) GO TO 60
DO 50 I=KPI,N
50 A(I, J) = A(I, J) + A(I, K) * T
60 CONTINUE
70 CONTINUE

CON D = (1-NORM OF A) * (AN ESTIMATE OF THE 1-NORM OF A INVERSE)
* THE ESTIMATE IS OBTAINED BY ONE STEP OF INVERSE ITERATION FOR THE
* SMALL SINGULAR VECTOR. THIS INVOLVES SOLVING TWO SYSTEMS
* IS A VECTOR OF +1 OR -1 COMPONENTS CHOSEN TO CAUSS GROWTH IN Y.
* ESTIMATE = (1-NORM OF Z) / (1-NORM OF Y)
*
* DO 100 K=1,N
T=0.DO
IF (K.EQ.1) GO TO 90
KM1=K-1
DO 80 I=1,KM1
80 T = T + A(I, K) * WORK(I)
90 EK=1.DO
IF (T.LT.0.DO) EK=-1.DO
IF (A(K,K).EQ.0.DO) GO TO 160
A11=A(1,1)
WORK(K) = -(EK+T)/A(1,1)
100 CONTINUE
DO 120 KB=1,NM1
K=N-KB
T=0.DO
KP1=K+1
DO 110 I=KP1,N
110 T = T + A(I, K) * WORK(K)
WORK(K) = T
M=IPVT(K)
IF (M.EQ.K) GO TO 120
T=WORK(M)
WORK(M) = WORK(K)
WORK(K) = T
120 CONTINUE
*
YNORM=0.DO
DO 130 I=1,N
130 YNORM=YNORM+DABS(WORK(I))
*
* SOLVE A*Z = Y
* CALL SOLVE (NDIM, N, A, WORK, IPVT)
*
ZNORM=0.0D0
DO 140 I=1,N
140 ZNORM=ZNORM+DABS(WORK(I))

* ESTIMATE THE CONDITION.
COND=ANORM*ZNORM/YNORM
IF (COND.LT.1.0D0) COND=1.0D0
RETURN

* 1-BY-1 CASE..
150 COND=1.0D0
IF (A(1,1).NE.0.0D0) RETURN

* EXACT SINGULARITY
160 COND=1.0D32
RETURN
END

SUBROUTINE SOLVE (NDIM,N,A,B,IPVT)
IMPLICIT REAL*8(A-H,O-Z)
DIMENSION A(NDIM,N),B(N),IPVT(N)

* SOLVES A LINEAR SYSTEM, A*X = B
* DO NOT SOLVE THE SYSTEM IF DECOMP HAS DETECTED SINGULARITY.
* COMPUTER METHODS FOR MATHEMATICAL COMPUTATIONS-, BY G. E. FORSYTHE,
* M. A. MALCOLM, AND C. B. MOLER (PRENTICE-HALL, 1977)
* INPUT..
* NDIM = DECLARED ROW DIMENSION OF ARRAY CONTAINING A
* N = ORDER OF MATRIX
* A = TRIANGULARIZED MATRIX OBTAINED FROM SUBROUTINE DECOMP
* B = RIGHT HAND SIDE VECTOR
* IPVT = PIVOT VECTOR OBTAINED FROM DECOMP
* OUTPUT..
* B = SOLUTION VECTOR, X

IF (N.EQ.1) GO TO 50
NM1=N-1
DO 20 K=1,NM1
   KP1=K+1
   M=IPVT(K)
   T=B(M)
   B(M)=B(K)
   B(K)=T
   DO 10 I=KP1,N
      10   B(I)=B(I)+A(I,K)*T
   CONTINUE

DO 40 KB=1,NM1

DO THE FORWARD ELIMINATION.
DO THE BACK SUBSTITUTION.
KM1=N-KB
K=KM1+1
B(K)=B(K)/A(K,K)
T=-B(K)
DO 30 I=1,KM1
30   B(I)=B(I)+A(I,K)*T
CONTINUE
B(1)=B(1)/A(1,1)
RETURN
END
Bibliography


# Report Documentation Page

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>4. Title and Subtitle</td>
<td>Generation and Tooth Contact Analysis of Spiral Bevel Gears With Predesigned Parabolic Functions of Transmission Errors</td>
<td></td>
<td></td>
</tr>
<tr>
<td>7. Author(s)</td>
<td>Faydor L. Litvin and Hong-Tao Lee</td>
<td></td>
<td></td>
</tr>
<tr>
<td>9. Performing Organization Name and Address</td>
<td>University of Illinois at Chicago \Department of Mechanical Engineering \Chicago, Illinois</td>
<td></td>
<td></td>
</tr>
<tr>
<td>12. Sponsoring Agency Name and Address</td>
<td>Propulsion Directorate \U.S. Army Aviation Research and Technology Activity—AVSCOM \Cleveland, Ohio 44135-3127 \and \NASA Lewis Research Center \Cleveland, Ohio 44135-3191</td>
<td></td>
<td></td>
</tr>
<tr>
<td>15. Supplementary Notes</td>
<td>Project Manager, Robert F. Handschuh, Propulsion Directorate, U.S. Army Aviation Research and Technology Activity—AVSCOM.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>16. Abstract</td>
<td>A new approach for determination of machine-tool settings for spiral bevel gears is proposed. The proposed settings provide a predesigned parabolic function of transmission errors and the desired location and orientation of the bearing contact. The predesigned parabolic function of transmission errors is able to absorb piece-wise linear functions of transmission errors that are caused by the gear misalignment and reduce gear noise. The gears are face-milled by head cutters with conical surfaces or surfaces of revolution. A computer program for simulation of meshing, bearing contact and determination of transmission errors for misaligned gear has been developed.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>17. Key Words (Suggested by Author(s))</td>
<td>Gear geometry \Spiral bevel gears \Tooth contact analysis \Gear</td>
<td></td>
<td></td>
</tr>
<tr>
<td>18. Distribution Statement</td>
<td>Unclassified—Unlimited \Subject Category 37</td>
<td></td>
<td></td>
</tr>
<tr>
<td>19. Security Classif. (of this report)</td>
<td>Unclassified</td>
<td></td>
<td></td>
</tr>
<tr>
<td>20. Security Classif. (of this page)</td>
<td>Unclassified</td>
<td></td>
<td></td>
</tr>
<tr>
<td>21. No of pages</td>
<td>215</td>
<td></td>
<td></td>
</tr>
<tr>
<td>22. Price*</td>
<td>A10</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>