T.C.

UNIVERSITY OF GAZİANTEP

DEPARTMENT OF MECHANICAL ENGINEERING

ME499 GRADUATION PROJECT

STUDENT: ONUR ELDEŞ

NUMBER: 201442790

SUBMITTED TO : PROF.DR.NİHAT YILDIRIM

DATE OF SUBMISSION : JANUARY 2019

GENERATING A USER FRIENDLY INTERFACE SOFTWARE FOR OPTIMIZATION OF HELICAL&SPUR GEARS GEOMETRIES TO REDUCE THE NOISE AND VIBRATION

UNDERGRADUATION PROJECT IN MECHANICAL ENGINEERING UNIVERSITY OF GAZİANTEP

TURKEY

SUPERVISOR

PROF.DR.NİHAT YILDIRIM

BY ONUR ELDEŞ

JANUARY 2019

ACKNOWLEDGEMENTS

I would like to express my appreciation to my supervisor Prof. Dr. Nihat YILDIRIM for his support for my graduation project study and immense knowledge.

I would like to thank to Reseaech Assist. Burak ŞAHİN for his valuable support.

I would like to thank to my family and my friend Fatih HAKAN. They always support and encouraged me.

ABSTRACT

Gears are playing a critical role in power transmission systems especially in defence industry which transmit torque and motion between shafts. The gear noise problem that widely occurs in power transmission systems ,and it is originates from the vibration of the gear pair system. Transmission error theory is giving us a useful perspective to reduce the main causes of vibration such as pressure angle error, picth error ,misalignment , crowning ,relief and shape of relief . The main purpose of this project is generate a user friendly interface software program to serving to reduce the vibration and noise with these mentioned causes. Software algorithms are based on gathering previous researchs and using them to construct a reliable fea. This software is designed for lcr gear only. The main purposes of this program are reaching the better transmission error curve under design load for statıc contion and getting minimum peak-to-peak transmission error curve, so elımınate the main causes of vibration and noise for low contact helical or spur gear. This porgram based on finite element analysis's working theory,at this program user can decide the mesh number along the involute profile,thin slice number along facewidth and force increments.

LIST OF CONTENT

LIST OF FIGURES

1 INTRODUCTION

Before explaining the software , there is a necessary to giving theoretical background . So, the first 5 chapter serving to this purpose.

1.1 INTRODUCTION

1.1.1 History of gears

Gears are considered as one of the oldest equipment known to mankind. The origin of gears goes down to the Chinese South-Pointing Chariot in the 27th Century B.C. This chariot was known to pointing to the south no matter how it was turned.

Aristotle has the credit to his name of giving the earliest description of gears in the 4th century B.C. According to his definition, the direction of rotation is reversed when one gear wheel drives another gear wheel. Gears have been used by the Greek Inventors in water wheels and clocks. The sketches of various types of gears of this time can be found in the notebooks of Leonardo da Vinci.

Even after these ground-breaking discoveries, no major development concerning wheels was made until the 17th Century. In this time, first attempts were made to provide constant velocity ratios. These attempts utilized the involute curves. This was just the beginning of something that changed the world for all the good reasons. However, in the 19th century, form cutters and rotating cutters were first used and it was then in 1835 when the English inventor Whitworth patented the first gear hobbing process.

Various other patents followed until 1897 when Herman Pfauter of Germany invented the first hobbing machine capable of cutting both spur and helical gears. Through the 20th century, various types of machines developed. But, the next major step came in 1975 when the Pfauter Company in Germany introduced the first NC hobbing machine and in 1982 the Full 6 axis machine was introduced.

1.1.2 Introduction about gears

A gear is a kind of machine element in which teeth are cut around cylindrical or cone shaped surfaces with equal spacing. By meshing a pair of these elements, they are used to transmit rotations and forces from the driving shaft to the driven shaft. Gears can be classified by shape as involute, cycloidal and trochoidal gears. Also, they can be classified by shaft positions as parallel shaft gears, intersecting shaft gears, and non-parallel and non-intersecting shaft gears.

Their practical usage in the present day modern engineering system is enormous. In accordance with a contemporary development of mechanical engineering techniques ever growing requirements and working specifications. Along with modern high speed manufacturing industry development, gear are used widely in many applications ranging from automotive transmission to robot and aerospace engines. Different kinds of metallic gears are currently being manufactured for various industrial purposes. 74% of them are spur gears, 15% helical, 5% worm, 4% bevel, and the others are either epicyclical or internal gears.

The main purpose of gear mechanisms is to transmit rotation and torque between axes. The gear is a machine element that has intrigued many engineers because of numerous technological problems arises in a complete mesh cycle. If the gears were perfectly rigid and no geometrical errors or modifications were present, the gears would result in a constant speed at the output shaft. The assumption of no friction leads to that the gears would transmit the torque perfectly, which means that a constant torque at the output shaft. No force variations would exist and hence no vibrations and no noise could be created.

Of course, in reality, there are geometrical errors, deflections and friction present, and accordingly, gears sometimes create noise to such an extent that it becomes a problem. Transmission error occurs when a traditional non-modified gear drive is operated under assembly errors.

The definition of transmission error is “the difference between the actual position of the output gear and the position it occupy if the gear drive were perfectly conjugate”. Transmission error is the rotation delay between driving and driven gear caused by the disturbances of inevitable random noise factors such as elastic deformation, manufacturing error, alignment error in assembly.

1.1.3 Gears and Noise

Noise in the environment has been recognised as one of the main problems which reduce the "quality of life", and it is the subject of an increasing number of complaints from the general public.

Noise from transportation has been the major contributor, and consequently, governments are under increasing pressure to introduce legislation to restrict noise emissions from vehicles and other machines, and thus steadily reduce the permissible limits . Combined with ever-stringent gaseous emissions regulations, engine manufacturers have been forced to increase fuel injection pressure, which has led to increased engine noise levels and deteriorated engine sound quality . Growing public awareness of noise pollution and an increasing number of noise sources, which are the result of increasing traffic density in urban areas, bring about increasingly stringent noise limits. This is especially true for the commercial and passenger car industries. When compared to other forms of power generation, combustion engines tend to be the prime source of noise emission .

Zhao and Reinhart have investigated the noise generated by the diesel engine, and conclude that this is influenced by the forcing functions which drive the structure of the engine, which in turn drives the radiating surfaces, which actually produce noise. The basic forcing functions causing the noise are cylinder pressure, bearing and gear impacts, piston slap, valve and overhead clearances. These forces act within the engine, causing the structure to vibrate. The structure in turn forces the radiating surfaces to vibrate and radiate noise.

Much of the noise reduction work carried out in the past tended to focus on the structure and radiating surfaces of the engine. Combustion in the engine has received substantial attention over the last decade, with other systems given slightly less priority. However, as most forcing functions are engine performance related and thus impossible to change dramatically without making serious compromises in engine performance, emissions, and fuel economy, this leaves the workable aspects of the forcing functions such as the timing gear systems, comprising of the injection timing gear, fuel pump gear, camshaft gear and transmission systems to be examined more closely.

Impacts between gears have long been identified as one of the main contributors of noise within the transmission systems. Most of these gear train impacts are caused by alternating torque fluctuations produced by combustion and inertia forces acting on the main running gear. The alternating torque accelerates and decelerates individual gears, which results in the excitation of gears. This excitation is then transmitted to the surrounding structures which include the gear transmission system, the engine mounting struts and engine panels.

Gear transmissions are an integral part of automotives and other industrial machineries. In keeping with the current trend towards high mechanical efficiency, the pursuit of compact and

Figure 1.1 Noise transmission path in gear transmissions

lightweight transmission systems cause an increasing amount of elastic deformation of the gears. The study and understanding of gear dynamics is fundamentally important for the monitoring, control and design of better gear transmission systems.

Although a lot of work has been carried out in the field of gear dynamics, there is still scope to investigate thoroughly certain areas that were not well developed before. In the past, the computational limitations were a barrier to certain methods of investigation, and as a result theoretical and numerical methods were prevalent in trying to understand the dynamic behaviour of gears.

Recent advancements in computational software, development of numerous finite element analysis (FEA) packages along with faster computers have aided in some innovative approaches to investigating vibration in gears.

As mentioned earlier in this chapter, various factors like torque and geometrical imperfections have an effect on the vibration and noise generated from a geared system, and in order to better understand this, one needs to focus on the contact mechanics of the geared system.

In essence, if a gear pair is taken to represent an idealised system with all the necessary constraints and operating conditions accurately modelled, it would be possible to analyse the behaviour of the gears under operation.

Most of the research carried out concentrated on the reduction of noise in gear contact and the hypothesis that transmission error (TE) was the main cause of most of the noise generated by gear pairs in contact, was developed and established.

Considering that transmission error is a major *source of noise*, the *actual noise* does not come directly from the angular speed variations. As mentioned earlier, the torsional accelerations cause vibratory bearing reactions that excite the gearbox casing, which then propagates the noise through the pulsation of the casing walls.

One of the main aims of this project is to develop an FEM procedure which is capable of modelling gear pairs and gear systems accurately, simulating their respective working conditions and allowing automated design changes, such as profile modification, within suitable optimisation algorithms.

1.2 LAYOUT OF THESIS

This thesis is consisting of 9 chapters.

Chapter 1; This sections is aimed to give introductory informations about this thesis concepts and basic for gears.

Chapter 2; consists of all gears background and basics for spur and helical gears geometry formulations and properties to serve the scopes of thesis.

Chapter 3; literature researches and theoretical base of this thesis is explained.

Chapter 4; TE theory and spur and helical gear calculations are defined. This chapter is the main background chapter for the this thesis.

Chapter 5; After chapter 4 , detailed describtions of tooth profiles of spur helical gears are explained in this section. Theoretical background explainations are finished with this section.

Chapter 6; Algorithms , explanations of them and nomenclature for algorithms are represented to the target group of this project.

Chapter 7; Some approved tests for spur gears are studied to prove the correctness of this project.

Chapter 8; Some approved tests for helical gears are studied to prove the correctness of this project.

Chapter 9; Discussions,conclusions are done based on last researches . Tips for the future works are suggested .

2 FUNDAMENTALS OF GEARS

2.1 TYPES OF GEARS

Spur gears, illustrated in Fig.2.1, have teeth parallel to the axis of rotation and are used to transmit motion from one shaft to another, parallel, shaft. Of all types, the spur gear is the simplest and, for this reason, will be used to develop the primary kinematic relationships of the tooth form.

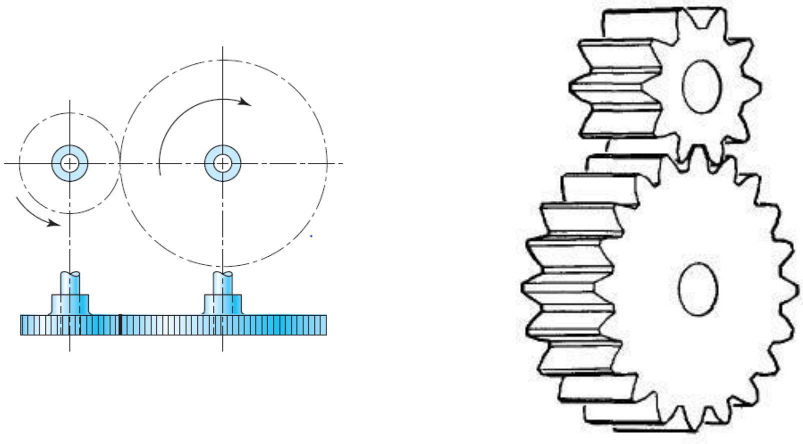


Figure 2.1 Spur gear pairs

Helical gears, shown in Fig. 2.2, have teeth inclined to the axis of rotation. Helical gears can be used for the same applications as spur gears and, when so used, are not as noisy, because of the more gradual engagement of the teeth during meshing. The inclined tooth also develops thrust loads and bending couples, which are not present with spur gearing. Sometimes helical gears are used to transmit motion betwee n nonparallel shafts.

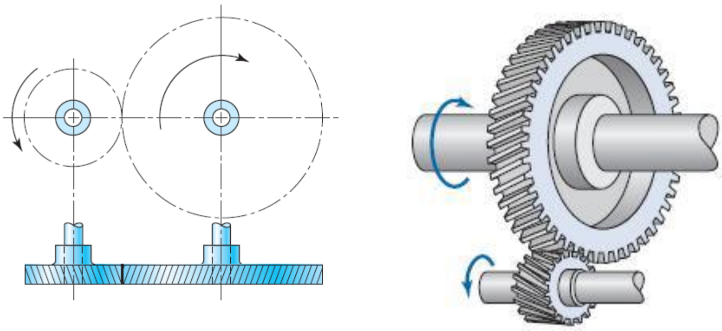


Figure 2.2 Helical gear pair

Bevel gears, shown in Fig. 2.3, have teeth formed on conical surfaces and are used mostly for transmitting motion between intersecting shafts. The figure actually illustrates *straight-tooth bevel gears. Spiral bevel gears* are cut so the tooth is no longer straight, but forms a circular arc. *Hypoid gears* are quite similar to spiral bevel gears except that the shafts are offset and nonintersecting.

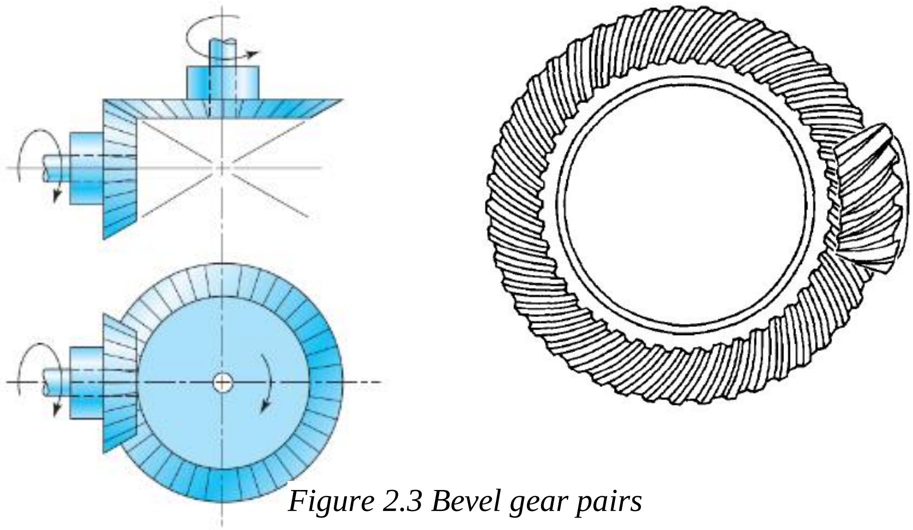


Figure 2.3 Bevel gear pairs

Worm gears, represent the screw type of gear set. The worm is a derivation of the helical gear with quite a large helix angle, usually around 90° (Figure2.4) and its body is usually long in the axial direction and these attributes are what gives the worm its screw like qualities. The worm is usually meshed with a normal disc type gear which is called the “worm gear” or the “wheel”. The main advantage of the worm gear drive is that it can achieve a high gear ratio with very few parts. When compared to the gear ratios of helical gears being limited to less than 10:1, worm gear drives can operate with gear ratios ranging from 10:1 to 100:1. Due to the relatively large helix of the worm (gear), there is considerable sliding action between the teeth resulting in significant frictional losses reducing the efficiency of the drive to almost 50% in some applications .

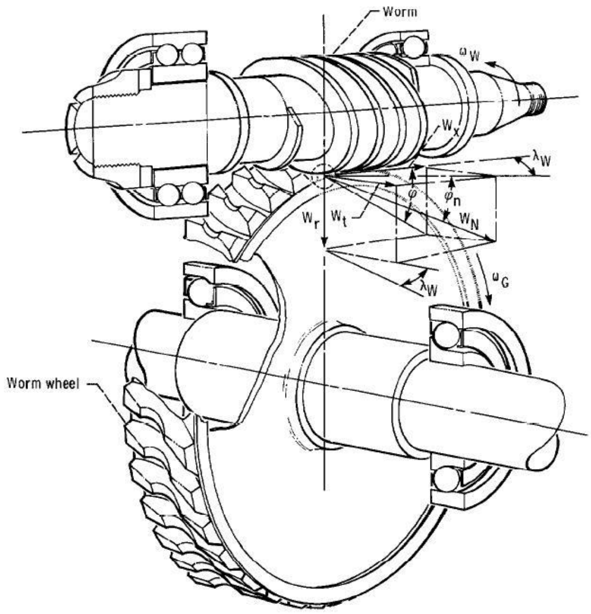


Figure 2.4 Worm gear pair

2.2 GEAR SELECTION CRITERIA

Since there are countless types of machines that have applications for gears, choosing the right type of gear for the suitable application is quite an elaborate task. In most cases the geometric arrangement of the apparatus that needs the gear drive will dictate the gear selection. If the gears are to be on parallel axes, then spur or helical gears are the ones to be used. Bevel and worm gears can be used if the axes are at right angles but are not suitable for parallel axes drives. The general gear selection criteria can be summarised as shown in the Table 2.1.

Categories of Gears	Types of Gears	Efficiency (%)
Parallel Axes Gears	Spur Gear Spur Rack Internal Gear Helical Gear Helical Rack Double Helical Gear	98 ... 99.5
Intersecting Axes Gears	Straight Bevel Gear Spiral Bevel Gear Zerol Gear	98 ... 99
Nonparallel and Nonintersecting Axes Gears	Worm Gear Screw Gear Hypoid Gear	30 ... 90 70 ... 95 96 ... 98

Figure 2.5 Types of gears in common use

2.3 BASICS OF SPUR GEARS

The fundamentals of gearing are illustrated through the spur gear tooth, both because it is the simplest, and hence most comprehensible, and because it is the form most widely used, particularly for instruments and control systems.

The basic geometry and nomenclature of a spur gear mesh is shownin **Figure 2.6**. The essential features of a gear mesh are:

1. Center distance.
2. The pitch circle diameters (or pitch diameters).
3. Sizeof teeth (or module).
4. Number of teeth.
5. Pressure angle of the contacting involutes.

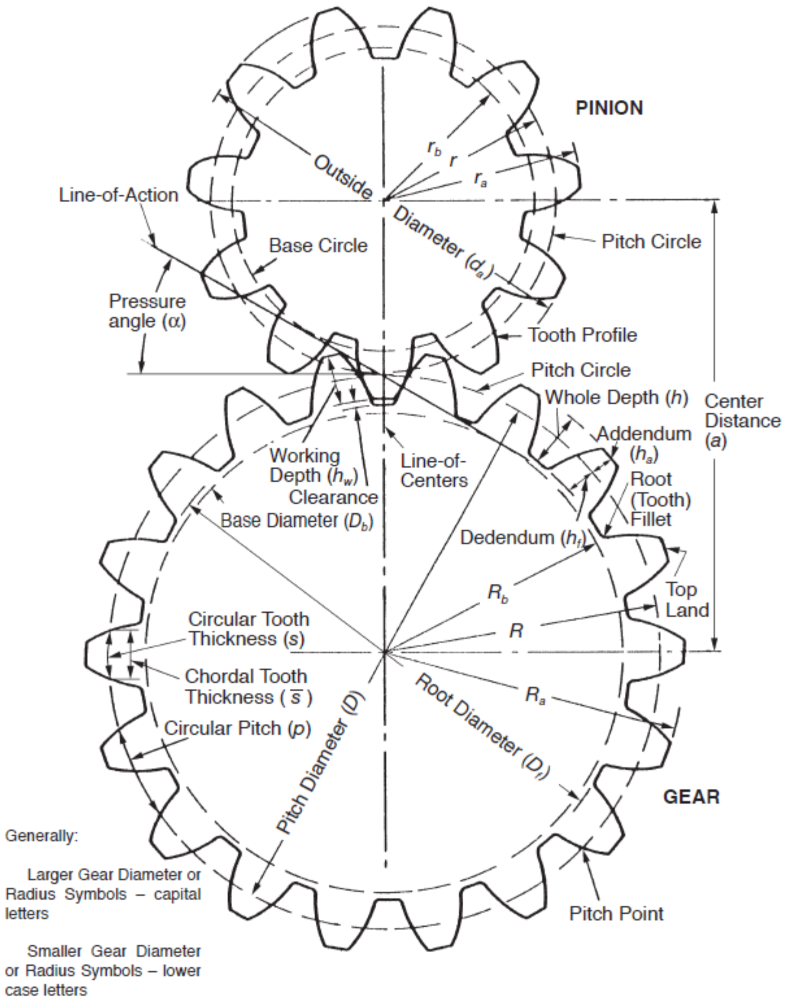


Figure 2.6 Basic Gear Geometry

2.3.1 ***The Law Of Gearing***

A primary requirement of gears is the constancy of angular velocities or proportionality of position transmission. Precision instruments require positioning fidelity. High-speed and/or high-power gear trains also require transmission at constant angular velocities in order to avoid severe dynamic problems. Constant velocity (i.e., constant ratio) motion transmission is defined as "conjugate action" of the gear tooth profiles. A geometric relationship for the form of the tooth profiles to provide conjugate action which is summarized as the Law of Gearing as follows:

"A common normal to the tooth profiles at their point of contact must, in all positions of the contacting teeth, pass through a fixed point on the lineof-centers called the pitch point."

Any two curves or profiles engaging each other and satisfying the lawof gearing are conjugate curves.

2.3.2 ***Conjugate action***

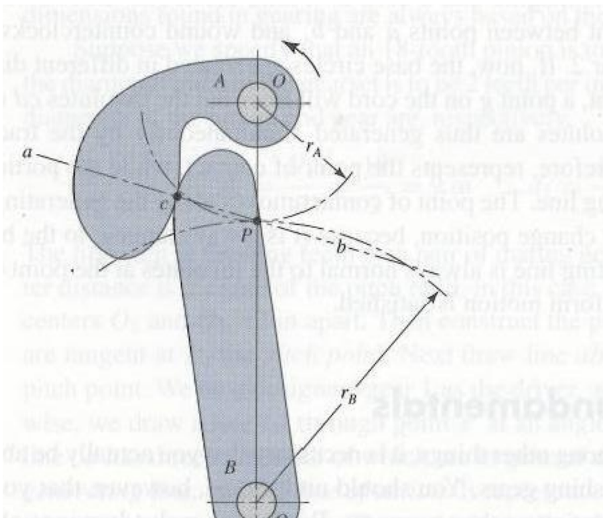


Figure 2.7 Conjugate action between cam A and follower B

The mating teeth of gears acting against each other are like cams and produce rotary motion as a result. Considering the gear teeth to be perfectly formed, smooth and rigid, although highly unrealistic, helps demonstrate the principle of conjugate action.

When gear teeth have been so designed to produce a constant angular velocity ratio between the gear pair on meshing, they are said to have conjugate action. This action can be further analysed using the figure 2.7 shown below.

2.3.3 Pressure Angle

The pressure angle is defined as the angle between the line-of action (common tangent to the base circles in **Figures 2.9** and **2.10**) and a perpendicular to the line-of-centers. See **Figure 2.8**.

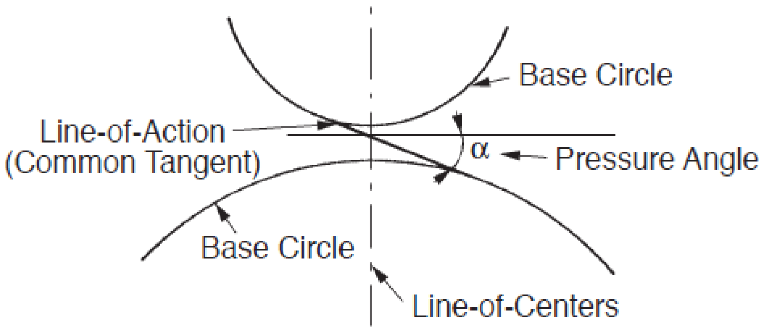


Figure 2.8 Definition of Pressure Angle

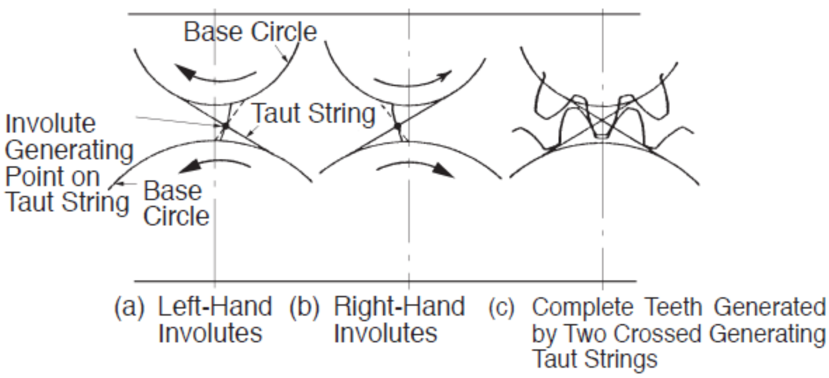


Figure 2.9 Generation and Action of Gear Teeth

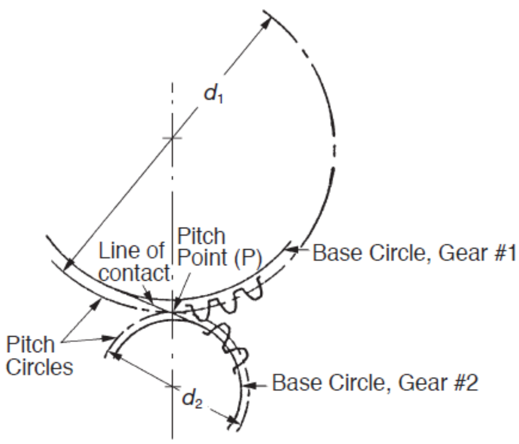


Figure 2.10 Definition of Pitch Circle and

From the geometry of these figures, it is obvious that the pressure angle varies (slightly) as the center distance of a gear pair is altered. The base circle is related to the pressure angle and pitch diameter by the equation:

db=dcosα (2-1)

where *d* and α are the standard values, or alternately:

db= *d*'cos *d*' (2-2)

where *d*' and- α' are the exact operating values.

The basic formula shows that the larger the pressure angle the smaller the base circle. Thus, for standard gears, 14.5° pressure angle gears have base circles much nearer to the roots of teeth than 20° gears. It is for this reason that 14.5° gears encounter greater undercutting problems than 20° gears.

2.3.4 Proper Meshing And Contact Ratio

Figure 2.11 shows a pair of standard gears meshing together. The contact point of the two involutes, as **Figure 2.11** shows, slides along the common tangent of the two base circles as rotation occurs. The common tangent is called the line-of-contact, or line-of-action.

A pair of gears can only mesh correctly if the pitches and the pressure angles are the same. Pitch comparison can be module *m*, circular *p*, or base *pb*.

That the pressure angles must be identical becomes obvious from the following equation for base pitch:

pb = π m cosα (2-3)

Thus, if the pressure angles are different, the base pitches cannot be identical.

The length of the line-of-action is shown as *ab* in **Figure 2.11**.

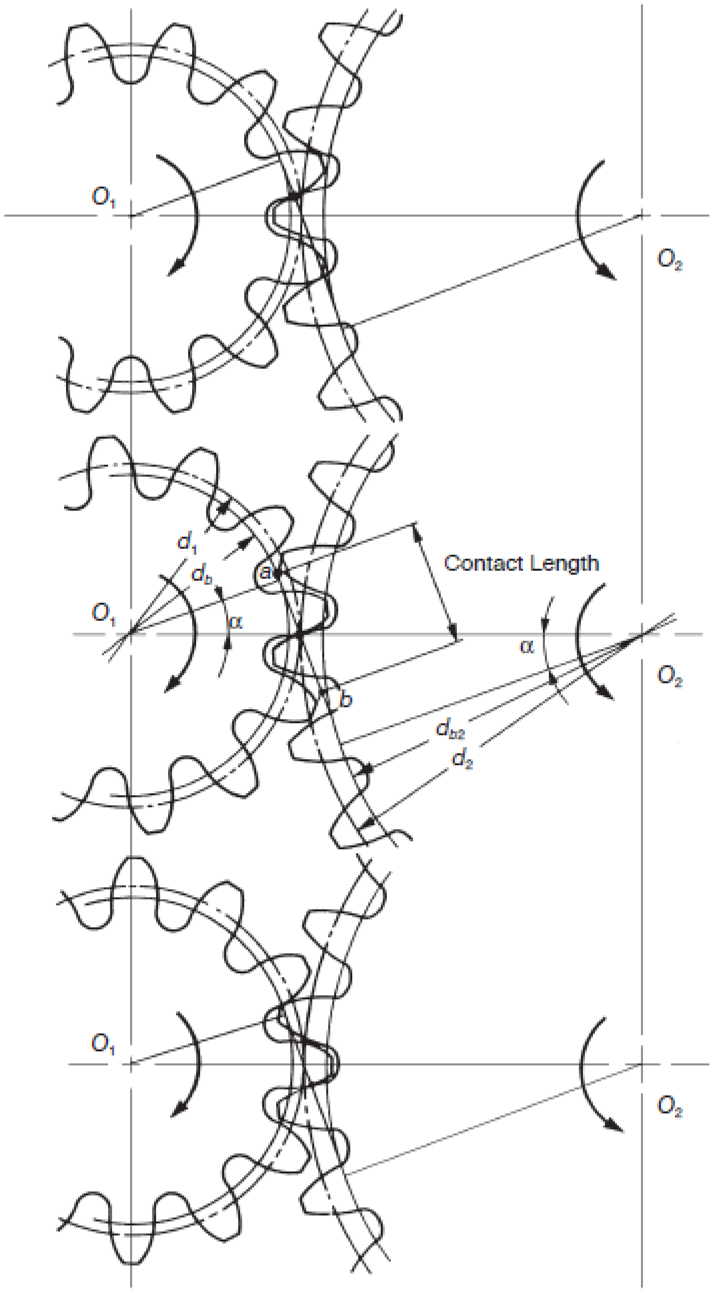


Figure 2.11 The Meshing of Involute Gear

2.3.5 <u>Contact Ratio</u>

To assure smooth continuous tooth action, as one pair of teeth ceases contact a succeeding pair of teeth must already have come into engagement. It is desirable to have as much overlap as possible. The measure of this overlapping is the contact ratio. This is a ratio of the length of the line-of-action to the base pitch. **Figure 2.12** shows the geometry. The length-of-action is determined from the intersection of the line-of-action and the outside radii. For the simple case of a pair of spur gears, the ratio of the length-of-action to the base pitch is determined from:

$$\varepsilon_\gamma = \frac{\sqrt{(R_a^2 - R_b^2)} + \sqrt{(r_a^2 - r_b^2)} - a \sin\alpha}{p \cos\alpha} \qquad (2\text{-}4)$$

It is good practice to maintain a contact ratio of 1.2 or greater. Under no circumstances should the ratio drop below 1.1, calculated for all tolerances at their worst-case values. A contact ratio between 1 and 2 means that part of the time two pairs of teeth are in contact and during the remaining time one pair is in contact. A ratio between 2 and 3 means 2 or 3 pairs of teeth are always in contact. Such a high contact ratio generally is not obtained with external spur gears, but can be developed in the meshing of an internal and external spur gear pair or specially designed nonstandard external spur gears.

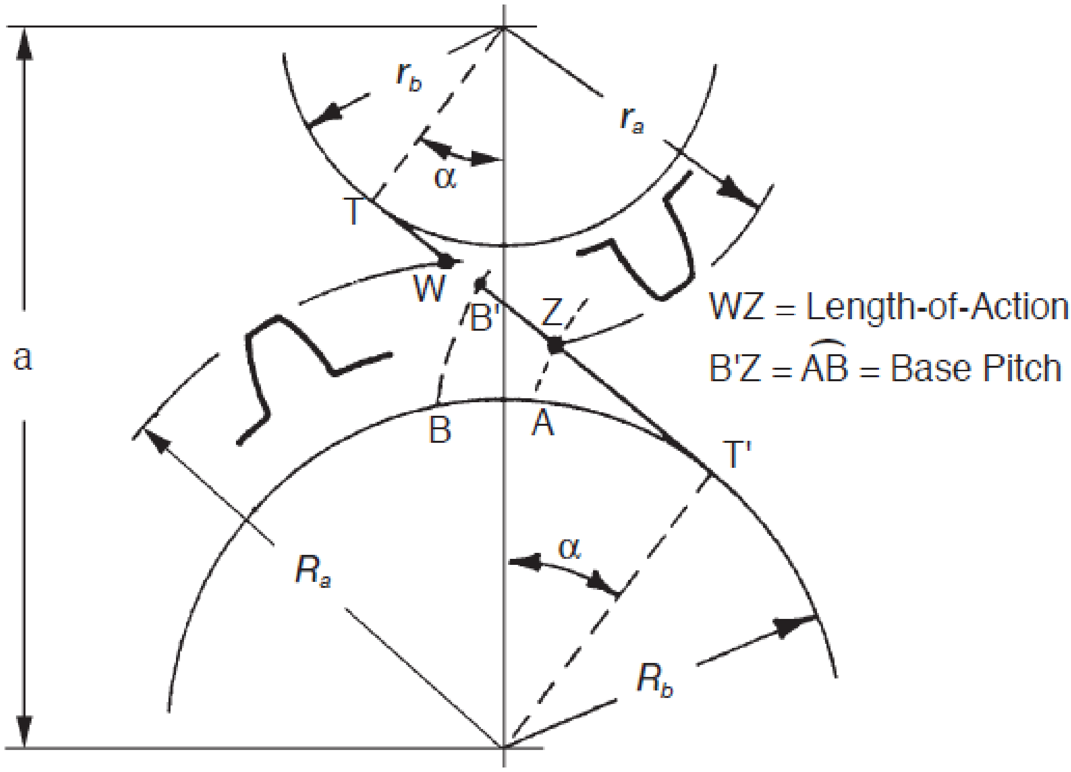


Figure 2.12 Geometry of Contact Ratio

2.3.6 <u>The Involute Function</u>

There is almost an infinite number of curves that can be developed to satisfy the law of gearing, and many different curve forms have been tried in the past. Modern gearing (except for clock gears) is based on involute teeth. This is due to three major advantages of the involute curve:

1. Conjugate action is independent of changes in centerdistance.
2. The form of the basic rack tooth is straight-sided, and therefore is relatively simple and can be accurately made; as a generating tool it imparts high accuracy to the cut gear tooth.

3. One cutter can generate all gear teeth numbers of the same pitch. The involute curve is most easily understood as the trace of a point at the end of a taut string that unwinds from a cylinder. It is imagined that a point on a string, which is pulled taut in a fixed direction, projects its trace onto a plane that rotates with the base circle.

Figure 2.13 shows an element of involute curve. The definition of involute curve is the curve traced by a point on a straight line which rolls without slipping on the circle. The circle is called the base circle of the involutes. Two opposite hand involute curves meeting at a cusp form a gear tooth curve. We can see, from **Figure 2.12**, the length of base circle arc ac equals the length of straight line bc.

$$\tan\alpha = \frac{bc}{Oc} = \frac{r_b\theta}{r_b} = \theta \text{ (radian)} \qquad (2\text{-}5)$$

The θ in **Figure 2.12** can be expressed as invα + α, then **Formula (2-5)** will become:

$$\mathrm{inv}\alpha = \tan\alpha - \alpha \qquad (2\text{-}6)$$

Function of α, or invα, is known as involute function. Involute function is very important in gear design. Involute function values can be obtained from appropriate tables. With the center of the base circle *O* at the origin of a coordinate system, the involute curve can be expressed by values of *x* and *y* as follows:

$$x = r\cos(\mathrm{inv}\alpha) = \frac{r_b}{\cos\alpha}\cos(\mathrm{inv}\alpha) \qquad (2\text{-}7)$$

$$y = r\sin(\mathrm{inv}\alpha) = \frac{r_b}{\cos\alpha}\sin(\mathrm{inv}\alpha)$$

where, $r = \dfrac{r_b}{\cos\alpha}$.

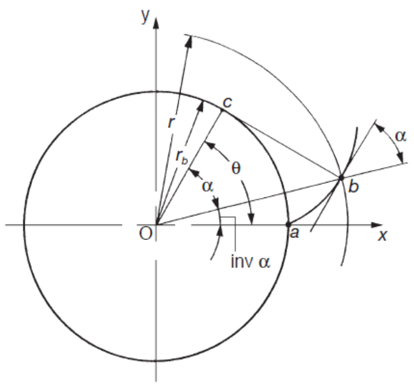


Figure 2.13 The Involute Curve

2.3.7 Spur Gear Nomenclature

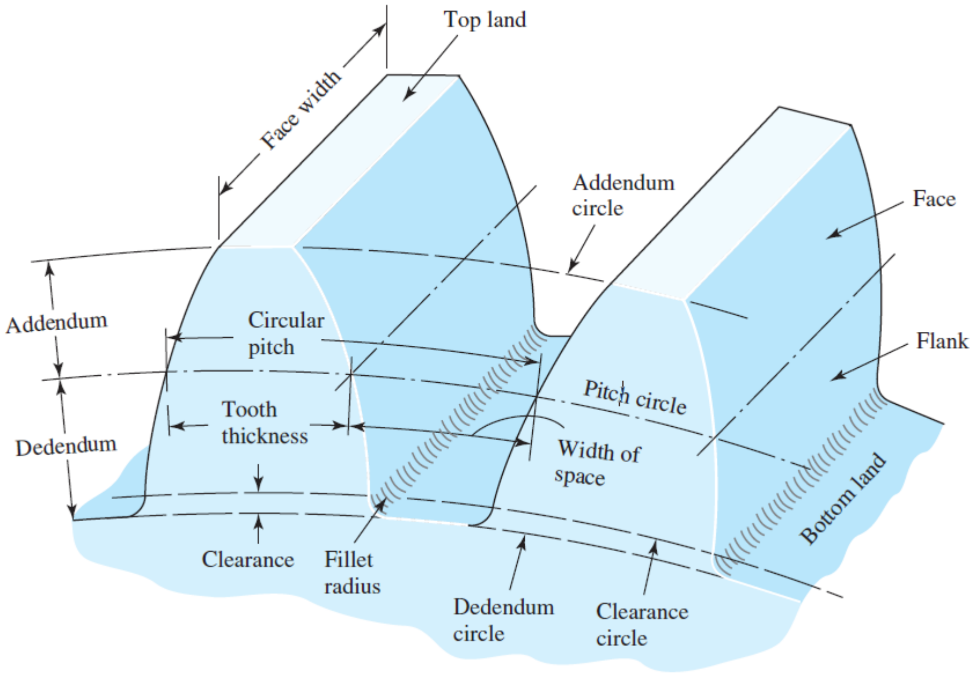


Figure 2.14 Nomenclature of spur-gear teeth.

Pitch Circles Referring to **Figure 2-15**, the tangent to the two base circles is the line of contact, or line-of-action in gear vernacular. Where this line crosses the line-of-centers establishes the pitch point, P. This in turn sets the size of the pitch circles, or as commonly called, the pitch diameters. The ratio of the pitch diameters gives the velocity ratio: Velocity ratio of gear 2 to gear 1 is: d1/d2.

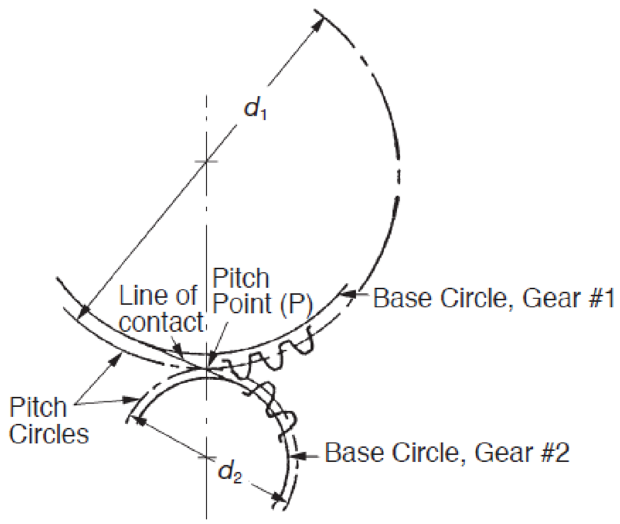


Figure 2.15 Definition of Pitch Circle and Pitch Point

Essential to prescribing gear geometry is the size, or spacing of the teeth along the pitch circle. This is termed pitch, and there are two basic forms.

Circular pitch — A naturally conceived linear measure along the pitch circle of the tooth spacing. Referring to **Figure 2-16**, it is the linear distance (measured along the pitch circle arc) between corresponding points of adjacent teeth. It is equal to the pitch-circle circumference divided by the number of teeth:

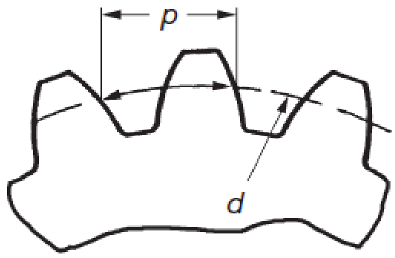


Figure 2.16 Definition of Circular Pitch

Module — Metric gearing uses the quantity module *m* in place of the American inch unit, diametral pitch. The module is the length of pitchdiameter per tooth. Thus:

Relation of pitches: From the geometry that defines the two pitches, it can be shown that module and circular pitch are related by the expression:

This relationship is simple to remember and permits an easy transformation from one to the other. Diametral pitch *Pd* is widely used in England and America to represent the tooth size. The relation between diametral pitch and module is as follows:

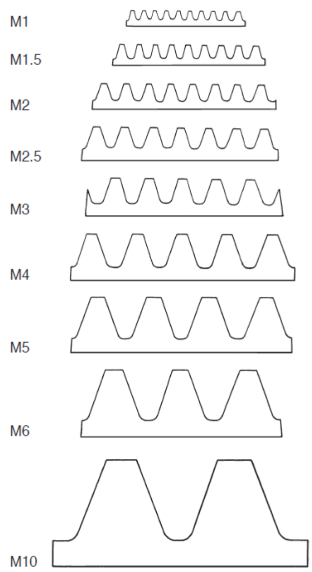


Figure 2.17 Comparative Size of Various Rack Teeth

The ***<u>addendum</u>*** *a* is the radial distance between the *top land* and the pitch circle.

The ***<u>dedendum</u>*** *b* is the radial distance from the *bottom land* to the pitch circle. The *whole depth ht* is the sum of the addendum and the dedendum.

The ***<u>clearance circle</u>*** is a circle that is tangent to the addendum circle of the mating gear. The ***<u>clearance c</u>*** is the amount by which the dedendum in a given gear exceeds the addendum of its mating gear. The *backlash* is the amount by which the width of a tooth space exceeds the thickness of the engaging tooth measured on the pitch circles.

<u>Backlash</u> is the tangential space between teeth of mating gears at pitch circles.

<u>Full depth</u> is sum of the addendum and the dedendum.

<u>Face width</u> is length of tooth parallel to axes.

<u>Pressure line</u> is the common normal at the point of contact of mating gears along which the driving tooth exerts force on the driven tooth.

<u>Pressure angle</u> is the angle between the pressure line and common tangent to pitch circles. It is also called angle of obliquity. high pressure angle requires wider base and stronger teeth.

<u>Contact ratio</u> is angle of angle of action and pitch angle.

<u>Path of approach</u> is the distance along the pressure line traveled by the contact point from the point of engagement to the pitch point.

<u>Path of recess</u> is the distance traveled along the pressure line by the contact point from the pitch point to the path of disengagement.

<u>Patch of contact</u> is the sum of path of approach and path of recess.

<u>Arc of approach</u> is the distance traveled by a point on either pitch circle of the two wheels from the point of engagement to the pitch.

<u>Arc of recess</u> is the distance traveled by a point on either pitch circle of the two wheels from the point to the point of disengagement.

<u>Arc of contact</u> is the distance traveled by a point on either pitch circle of the two wheels during the period of contact of a pair of teeth.

<u>Angle of action</u> is the angle turned by a gear during arc of contact.

To Obtain	From Known	Use This Formula
Pitch Diameter	Module	$D = mN$
Circular Pitch	Module	$p_c = m\pi = \frac{D}{N}\pi$
Module	Diametral Pitch	$m = \frac{25.4}{P_d}$
Number of Teeth	Module and Pitch Diameter	$N = \frac{D}{m}$
Addendum	Module	$a = m$
Dedendum	Module	$b = 1.25m$
Outside Diameter	Module and Pitch Diameter or Number of Teeth	$D_o = D + 2m = m(N + 2)$
Root Diameter	Pitch Diameter and Module	$D_R = D - 2.5m$
Base Circle Diameter	Pitch Diameter and Pressure Angle	$D_b = D \cos\phi$
Base Pitch	Module and Pressure Angle	$p_b = m\,\pi \cos\phi$
Tooth Thickness at Standard Pitch Diameter	Module	$T_{std} = \frac{\pi}{2} m$
Center Distance	Module and Number of Teeth	$C = \frac{m(N_1 + N_2)}{2}$
Contact Ratio	Outside Radii, Base Circle Radii, Center Distance, Pressure Angle	$m_p = \frac{\sqrt{{}_1R_o - {}_1R_b} + \sqrt{{}_2R_o - {}_2R_b} - C\sin\phi}{m\,\pi\cos\phi}$
Backlash (linear)	Change in Center Distance	$B = 2(\Delta C)\tan\phi$
Backlash (linear)	Change in Tooth Thickness	$B = \Delta T$
Backlash (linear) Along Line-of-action	Linear Backlash Along Pitch Circle	$B_{LA} = B\cos\phi$
Backlash, Angular	Linear Backlash	${}_aB = 6880\,\frac{B}{D}$ (arc minutes)
Min. No. of Teeth for No Undercutting	Pressure Angle	$N_c = \frac{2}{\sin^2\phi}$

Figure 2.18 Spur Gear Design Formulas

2.4 BASICS OF HELICAL GEARS

The helical gear differs from the spur gear in that its teeth are twisted along a helical path in the axial direction. It resembles the spur gear in the plane of rotation, but in the axial direction it is as if there were a series of staggered spur gears. This design brings forth a numberof different features relative to the spur gear, two of the most important being as follows:

1. Tooth strength is improved because of the elongated helical wraparound tooth base support.
2. Contact ratio is increased due to the axial tooth overlap. Helical gears thus tend to have greater load carrying capacity than spur gears of the same size. Spur gears, on the other hand, have a somewhat higher efficiency.

Helical gears are used in two forms:
1. Parallel shaft applications, which is the largest usage.
2. Crossed-helicals (also called spiral or screw gears) for connecting skew shafts, usually at right angles.

2.4.1 Generation Of The Helical Tooth

The helical tooth form is involute in the plane of rotation and can be developed in a manner similar to that of the spur gear. However, unlike the spur gear which can be viewed essentially as two dimensional, the helical gear must be portrayed in three dimensions to show changing axial features.

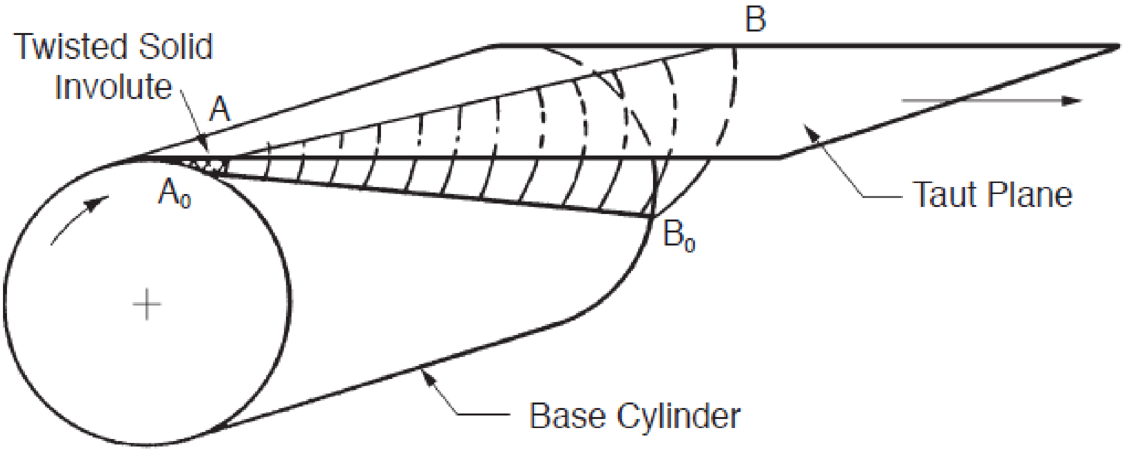


Figure 2.19 Generation of the Helical Tooth Profile

Referring to **Figure 2.19 ,** there is a base cylinder from which a taut plane is unwrapped, analogous to the unwinding taut string of the spur gear in **Figure 2.20.**

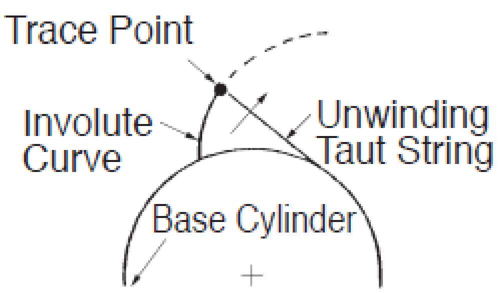


Figure 2.20 Generation of an Involute by a Taut String

On the plane there is a straight line AB, which when wrapped on the base cylinder has a helical trace AoBo. As the taut plane is unwrapped, any point on the line AB can be visualized as tracing an involute from the base cylinder. Thus, there is an infinite series of involutes generated by line AB, all alike, but displaced in phase along a helix on the base cylinder.

Again, a concept analogous to the spur gear tooth development is to imagine the taut plane being wound from one base cylinder on to another as the base cylinders rotate in opposite directions. The result is the generation of a pair of conjugate helical involutes. If a reversedirection of rotation is assumed and a second tangent plane is arranged so that it crosses the first,a complete involute helicoid tooth is formed.

2.4.2 <u>Helical Teeth</u>

In the plane of rotation, the helical gear tooth is involute and all of the relationships governing spur gears apply to the helical. However, the axial twist of the teeth introduces a helix angle. Since the helix angle varies from the base of the tooth to the outside radius, the helix angle β is defined as the angle between the tangent to the helicoidal tooth at the intersection of the pitch cylinder and the tooth profile, and an element of the pitch cylinder. See **Figure 2.20** .

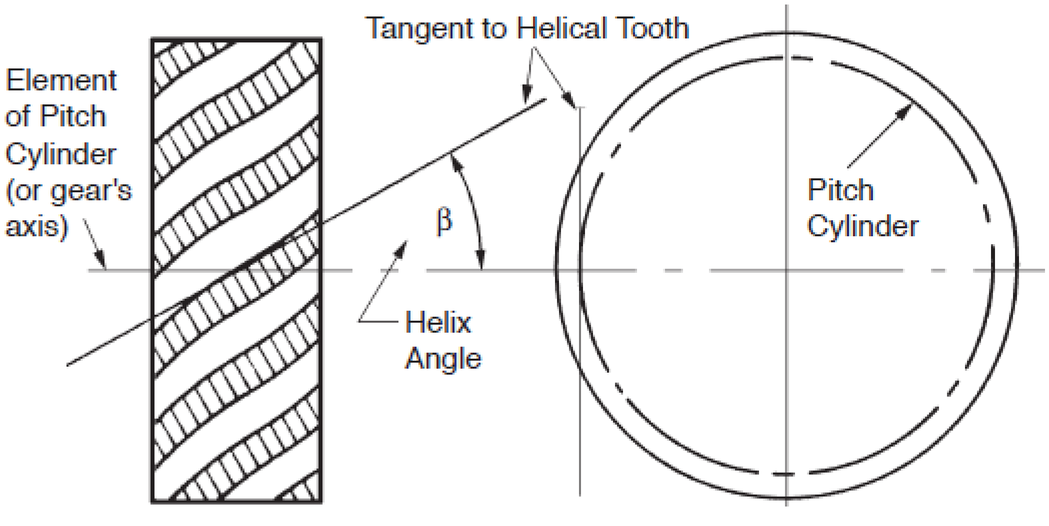


Figure 2.21 Definition of Helix Angle

The direction of the helical twist is designated as either left or right. The direction is defined by the right-hand rule. For helical gears, there are two related pitches – one in the plane of rotation and the other in a plane normal to the tooth. In addition, there is an axial pitch. Referring to **Figure 2.22**, the two circular pitches are defined and related as follows:

$$p_x = p_t \cot\beta = \frac{p_n}{\sin\beta} = \text{axial pitch} \qquad (2\text{-}8)$$

$$p_n = p_t \cos\beta = \text{normal circular pitch} \qquad (2\text{-}9)$$

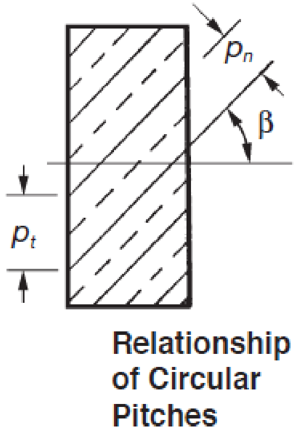


Figure 2.22 Relationship of Circular Pitches

A helical gear such as shown in **Figure 2.23** is a cylindrical gear in which the teeth flank are helicoid. The helix angle in standard pitch circle cylinder is β, and the displacement of one rotation is the lead, *L*. The tooth profile of a helical gear is an involute curve from an axial view, or in the plane perpendicular to the axis. The helical gear has two kinds of tooth profiles – one is based on a normal system, the other is based on an axial system. Circular pitch

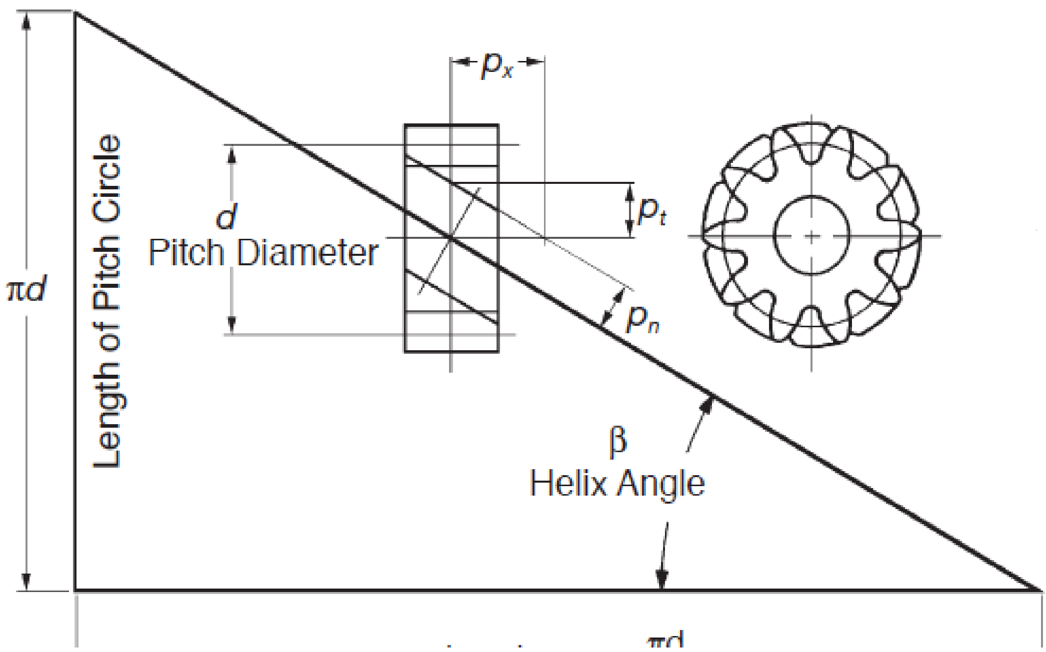


Figure 2.23 Fundamental Relationship of a Helical Gear (Right-Hand)

measured perpendicular to teeth is called normal circularpitch, *pn*. And *pn* divided by π is then a normal module, *mn*.

$$m_n = \frac{p_n}{\pi} \qquad (2\text{-}10)$$

The tooth profile of a helical gear with applied normal module, *mn*, and normal pressure angle *αn* belongs to a normal system. In the axial view, the circular pitch on the standard pitch circle is called the radial circular pitch, *pt*. And *pt* divided by π is the radial module, *mt*.

$$m_t = \frac{p_t}{\pi} \qquad (2\text{-}11)$$

2.4.3 <u>Equivalent Spur Gear</u>

The true involute pitch and involute geometry of a helical gear is in the plane of rotation. However, in the normal plane, looking at one tooth, there is a resemblance to an involute tooth of a pitch corresponding to the normal pitch. However, the shape of the tooth corresponds to a spur gear of a larger number of teeth, the exact value depending on the magnitude of the helix angle.

The geometric basis of deriving the number of teeth in this equivalent tooth form spur gear is given in **Figure 2.24**. The result of the transposed geometry is an equivalent number of teeth, given as:

$$z_v = \frac{z}{\cos^3\beta} \qquad (2\text{-}12)$$

This equivalent number is also called a virtual number because this spur gear is imaginary. The value of this number is used in determining helical tooth strength.

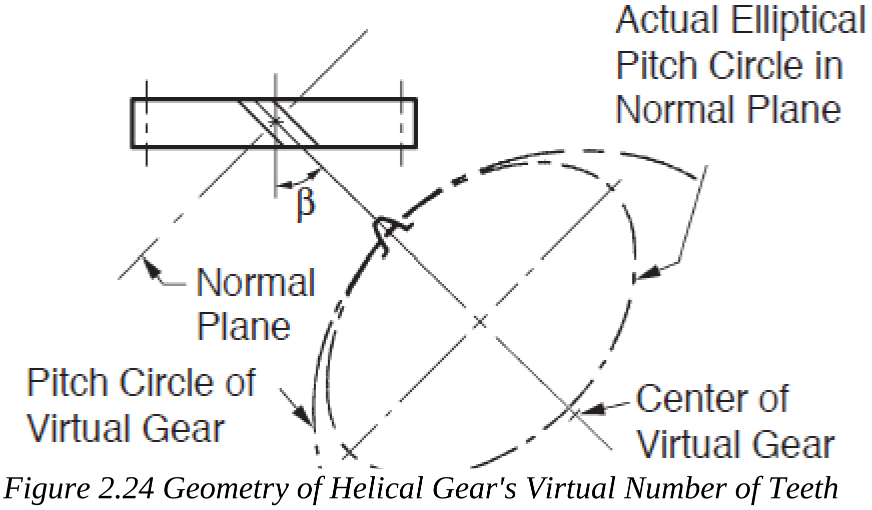


Figure 2.24 Geometry of Helical Gear's Virtual Number of Teeth

2.4.4 Helical Gear Pressure Angle

Although, strictly speaking, pressure angle exists only for a gear pair, a nominal pressure angle can be considered for an individual gear. For the helical gear there is a normal pressure, αn, angle as well as the usual pressure angle in the plane of rotation, α. **Figure 2.25** shows their relationship, which is expressed as:

$$\tan\alpha = \frac{\tan\alpha_n}{\cos\beta} \qquad (2\text{-}13)$$

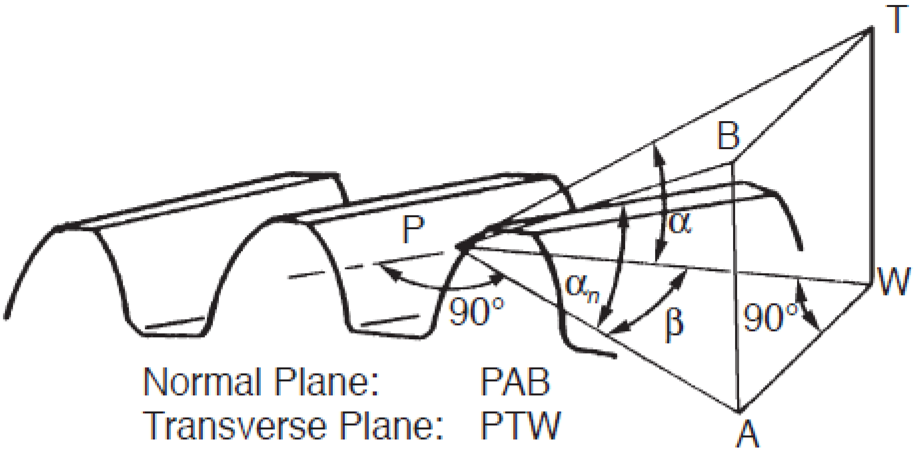


Figure 2.25 Geometry of Two Pressure Angles

2.4.5 Importance Of Normal Plane Geometry

Because of the nature of tooth generation with a rack-type hob, a single tool can generate helical gears at all helix angles as well as spur gears. However, this means the normal pitch is the common denominator, and usually is taken as a standard value. Since the true involute features are in the transverse plane, they will differ from the standard normal values. Hence, there is a real need for relating parameters in the two reference planes.

2.4.6 Helical Tooth Proportions

These follow the same standards as those for spur gears. Addendum, dedendum, whole depth and clearance are the same regardless of whether measured in the plane of rotation or the normal plane. Pressure angle and pitch are usually specified as standard values in the normal plane, but there are times when they are specified as standard in the transverse plane.

2.4.7 Parallel Shaft Helical Gear Meshes

Fundamental information for the design of gear meshes is as follows:

Helix angle – Both gears of a meshed pair must have the same helix angle. However, the helix direction must be opposite; i.e., a left-hand mates with a right-hand helix.

Pitch diameter – This is given by the same expression as for spur gears, but if the normal module is involved it is a function of the helix angle. The expressions are:

$$d = z\, m_t = \frac{z}{m_n \cos\beta} \qquad (2\text{-}14)$$

Center distance – Utilizing **Equation (2-14)**, the center distanceof a helical gear mesh is:

$$a = \frac{z_1 + z_2}{2\, m_n \cos\beta} \qquad (2\text{-}15)$$

Note that for standard parameters in the normal plane, the center distance will not be a standard value compared to standard spur gears. Further, by manipulating the helix angle, β, the center distance can be adjusted over a wide range of values. Conversely, it is possible:

1. to compensate for significant center distance changes (or errors) without changing the speed ratio between parallel geared shafts; and
2. to alter the speed ratio between parallel geared shafts, without changing the center distance, by manipulating the helix angle along with the numbers of teeth.

2.4.8 Helical Gear Contact Ratio

The contact ratio of helical gears is enhanced by the axial overlap of the teeth. Thus, the contact ratio is the sum of the transverse contact ratio, calculated in the same manner as for spur gears, and a term involving the axial pitch.

$$(\varepsilon)_{total} = (\varepsilon)_{trans} + (\varepsilon)_{axial} \qquad (2\text{-}16)$$

or

$$\varepsilon_r = \varepsilon_\alpha + \varepsilon_\beta$$

2.5 MANUFACTURING OF GEAR TEETH

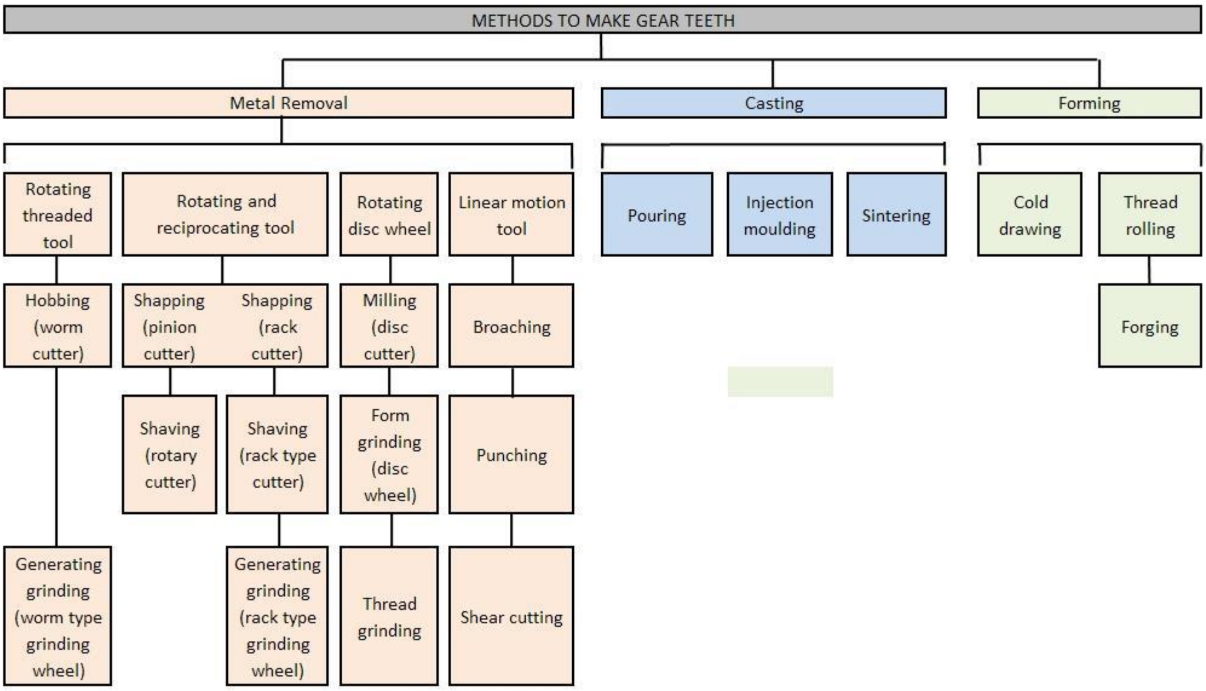


Figure 2.26 Gear manufacturing methods

Depending on the application of the gears, the manufacturing method used for the forming of gear teeth can be identified. There are usually a large number of ways that gear teeth can be manufactured falling broadly into the three categories of casting, forming and metal removal (Figure 2.26).

Gears can be produced by various casting processes, and mostly these gears will have a rough surface finish and will not be very accurate dimensionally. These types of gears are usually used in machines where additional noise and loss of accuracy of motion can be tolerated, such as in farm machines, handheld appliances and toys.

Gears made by forming are more accurate than the ones made by casting. In roll forming, the gear blank is mounted on a shaft and pressed against a rolling die usually made of hardened steel. The rolls are gradually fed inwards for several rotations until all the gear teeth are formed. Roll forming of gear teeth can be carried out using hot- or cold-rolling methods. Cold rolling is one of the new methods used to obtain high quality generated profiles with improved mechanical properties. Gear teeth are usually machined by one of the above mentioned methods to leave a toothed wheel after the process.

Highly stressed gears are usually made of steel and then cut with form or generating cutters. The shape of the form cutter is the exact tooth space between gear teeth. In generating cutters, the tool has the different shape compared to the tooth profile, and this is moved relative to the gear blank, so that the required gear tooth shape is obtained after a full revolution of the gear blank. Generating gears from gear blanks is one of the most accurate ways of obtaining high-precision gears; this is mainly due to the stiffness between the gear blank and the cutter during the generating procedure.

2.5.1 Milling

Some gear teeth are cut with a form miller cutting tool shaped to fit the exact tooth space between gear teeth. Using this method, requires different tools for different sets of gears, and is hence expensive compared to other forms of cutters. The cutting tool is usually a toothed disc with the "gear tooth space" contour ground into the sides of the teeth.

2.5.2 Shaping

This is the most common type of generating cutter where the teeth is generated either with a pinion or rack cutter. The pinion cutter is used to cut along the vertical axis by cutting the gear blank to the required depth. Each tooth of the pinion cutter is a cutting tool, so the gear geometry is complete after a complete rotation of the blank.

2.5.3 Hobbing

Almost all types of gears including spur, helical and worm gears can be produced by the hobbing technique, with the only exceptions being internal gears. The hobbing machine is essentially a specialised version of a milling tool, where a cutting tool (the hob), progressively makes a series of cuts into the gear blank to generate the required teeth shape. The hobbing machine works with two non-parallel spindles, one carrying the gear blank and the other mounted with the hob. In the hobbing process, the cutting tool is usually cylindrical with helical cutting teeth that have grooves running the length of the hob in order to aid in cutting and chip removal. The hob is thus used to create spur and helical gears by being fed hob across the face width of the gear blank and thus cutting the required teeth profile. In the case of the worm gear, the hob is generally passed tangentially past the blank or radially into the blank.

The important factor, that decides which type of manufacturing method is chosen in the manufacture of the gear, depends on the intended application of the gear and the resources available for the application. There is in effect a trade-off between cost and accuracy in the manufacturing methods discussed above. The cheaper options of casting and forming will provide relatively good gears that can perform the desired tasks but always with a slight loss of accuracy, whereas, the more accurate generating processes will provide high-precision gears at a cost.

3 LITERATURE SURVEY AND BACKGROUND

All informations in this chapter are based on the book Gear Noise and Vibration Second Edition,Revised and Expanded J. Derek Smith. To reach more detail about this topic it is strongly recommended that see the book's related chapter/chapters.[24]

3.1 CAUSES OF NOISE

3.1.1 Possible causes of gear noise

To generate noise from gears the primary cause must be a force variation which generates a vibration (in the components), which is then transmitted to the surrounding structure. It is only when the vibration excites external panels that airborne noise is produced. Inside a normal sealed gearbox there are high noise levels but this does not usually matter since the air pressure fluctuations are not powerful enough to excite the gearcase significantly. Occasionally in equipment such as knitting machinery there are gears which are not sealed in oiltight cases and direct generated noise can then be a major problem. There are slight problems in terminology because a given oscillation at, for example, 600 Hz is called a vibration while it is still inside the steel but is called noise as soon as it reaches the air. Vibrations can be thought of as either variations of force or of movement, though, in reality, both must ocur together. Also, unfortunately, mechanical and electrical engineers often talk about "noise" when they mean the background random vibrations or voltages which are not the signal of interest. Thus we can sometimes encounter something being described as the signal-to-noise ratio of the (audible) noise.An additional complication can arise with very large structures especially at high frequencies because force and displacement variations no longer behave as conventional vibrations but act more as shock or pressure waves radiating through the system but this type of problem is rare. In general it is possible to reduce gear noise by:

(a) Reducing the excitation at the gear teeth. Normally for any system, less amplitude of input gives less output (noise) though this is not necessarily true for some non-linear systems.

(b) Reducing the dynamic transmission of vibration from the gear teeth to the sound radiating panels and out of the panels often by inserting vibration isolators in the path or by altering the sound radiation properties of the external panels.

(c) Absorbing the noise after it has been generated or enclosing the whole system in a soundproof box.

(d) Using anti-noise to cancel the noise in a particular position or limited number of positions, or using cancellation methods to increase the effectiveness of vibration isolators.

Of these approaches, (c) and (d) are very expensive and tend to be temperamental and delicate or impracticable so this book concentrates on (a) and (b) as the important approaches, from the economic viewpoint.

Sometimes initial development work has been done by development engineers on the gear resonant frequencies or the gear casing or sound radiating structure so (b) may have been tackled in part, leaving (a) as the prime target. However, it is most important to determine first whether (a) or (b) is the major cause of trouble. A possible alternative cause of noise in a spur gearbox can ocur with an overgenerous oil supply if oil is trapped in the roots of the meshing teeth. If the oil cannot escape fast through the backlash gap, it will be expelled forcibly axially from the tooth roots and, at once-per-tooth frequency, can impact on the end walls of the gearcase. This effect is rare and does not occur with helical teeth or with mist lubrication.

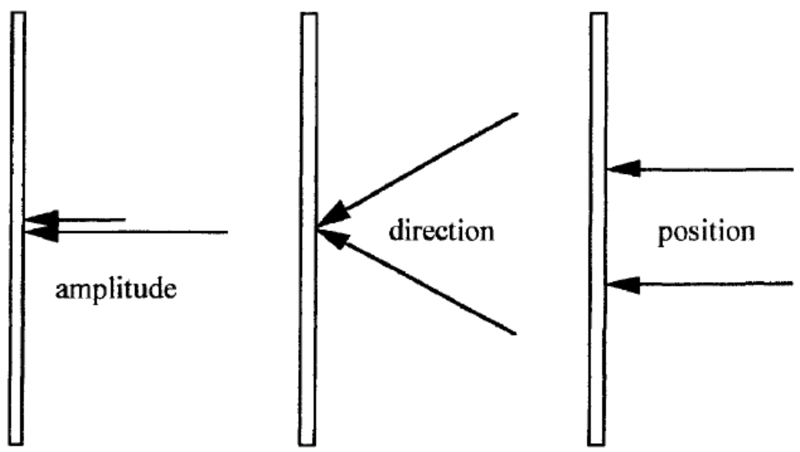


Figure 3.1 Types of vibration excitation due to change in amplitude (a),direction (b), or position (c).

The excitation is generally due to a force varying either in amplitude, direction or position as indicated in Fig. 3.1. Wildhaber-Novikov or Circ-Arc gears [1] . produce a strong vibration excitation due to the resultant force varying in position [Fig3.l(c)] as the contact areas move axially along the pitch line of the gears, so this type of drive is inherently noisier than an involute design.

Variation of direction of the contact force between the gears [Fig3. l(b)] can occur with unusual gear designs such as cycloidal and hypocycloidal gears [2] but, with involute gears, the direction variation is only due to friction effects. The effect is small and can be neglected for normal industrial gears as it is at worst a variation of $\pm 3°$ when the coefficient of friction is 0.05 with spur gears but is negligibly small with helical gears. For involute gears of normal attainable accuracy it is variation of the amplitude of the contact force [Fig..l(a)] that gives the dominant vibration excitation. The inherent properties of the involute give a constant force direction and a tolerance of centre distance variation as well as, in theory, a constant velocity ratio.

The source of the force variation in involute gears is a variation in the smoothness of the drive and is due to a combination of small variations of the form of the tooth from a true involute and varying elastic deflection of the teeth. This relative variation in displacement between the gears acts via the system dynamic response to give a force variation and resulting vibration. This thesis deals mainly with parallel shaft involute gears since this type of drive dominates the field of power transmission. Fundamentally the same ideas apply in the other types of drive such as chains, toothed belts, bevels, hypoids, or worm and wheel drives but they are of much less economic importance. The approach to problems is the same.

3.1.2 The Basic İdea Of Transmission Error

The fundamental concept of operation of involute (spur) gears is that shown in Fig. 3.2 where an imaginary string unwraps from one (pinion) base circle and reels onto a second (wheel) base circle. Any point fixed on the string generates an involute relative to base circle 1 and so maps out an involute tooth profile on gear 1 and at the same time maps out an involute relative to gear 2. (An involute is defined as the path mapped out by the end of an unwrapping string.) This theoretical string is the "line of action" or the pressure line and gives the

direction and position of the normal force between the gear teeth. Of course it is a rather peculiar mathematical string that
pushes instead of pulls, but this does not affect the geometry. In the literature on gearing geometry there is a tremendous amount of jargon with much discussion of pitch diameters, reference diameters, addendum size, dedendum size, positive and negative corrections (of the reference radius), undercutting limits, pressure angle variation, etc., together with a host of arcane rules about what can or cannot be done.

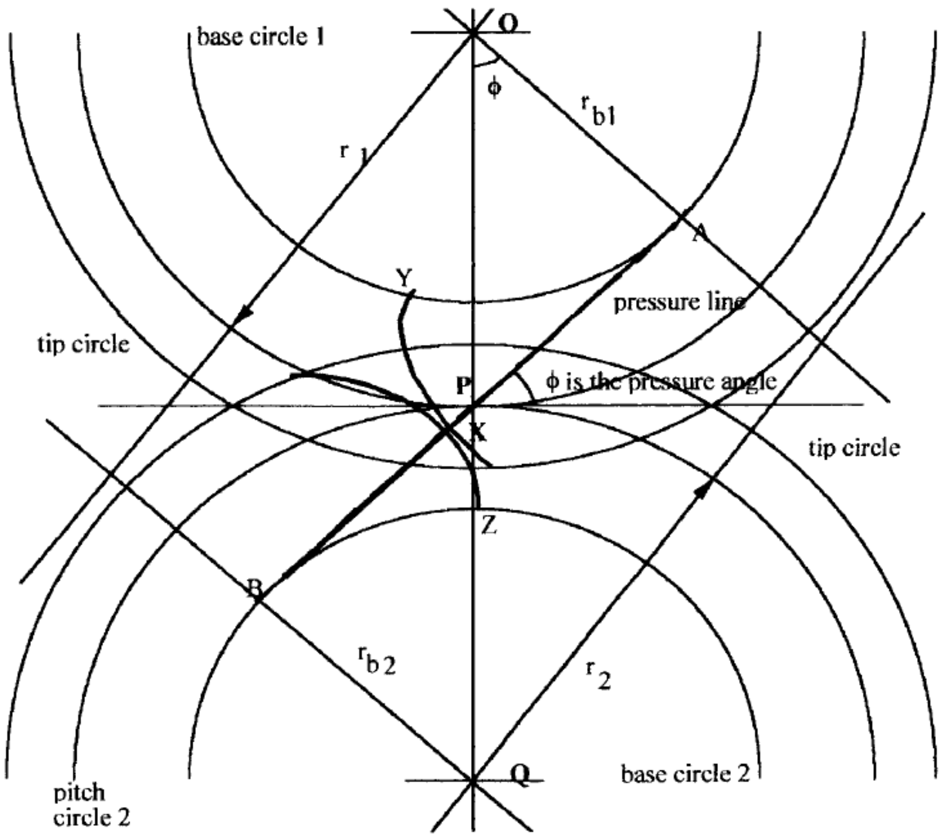


Figure 3.2 Involute operation modelled on unwrapping string.

All this is irrelevant as far as noise is concerned and it is important to remember that the involute is very, very simply defined and much jargon merely specifies where on an involute we work. There is, in reality, only one true dimension on a spur gear and that
is the base circle radius (and the number of teeth). Any one involute should mate with another to give a constant velocity ratio while they are in contact. It is possible to have two gears of slightly different nominal pressure angle meshing satisfactorily since pressure angle is not a fundamental property of a flank and depends on the centre distance at which the gears happen to be set. The only relevant criteria are:

(a) Both gears must be (nearly) involutes.

(b) Before one pair of teeth finish their contact the next pair must be ready to take over (contact ratio greater than 1.00).
(c) The base pitches of both gears must be the same (except for tip relief) so that there is a smooth handover from one pair to the next. (The base pitch of a gear is the distance from one tooth's flank to the next tooth's flank along the line of action and so tangential to the base circle.) If gears were perfect involutes, absolutely rigid and correctly spaced, there would be no vibration generated when meshing. In practice, for a variety of reasons, this does not occur and the idea of Transmission Error (T.E.) came into existence. Classic work on this was carried out by **Gregory, Harris and Munro** [3,4] at Cambridge in the late 1950s.

We define T.E. [5] by imagining that the input gear is being driven at an absolutely steady angular velocity and we would then hope that the output gear was rotating at a steady angular velocity. Any variation from this steady velocity gives a variation from the "correct position" of the output and this is the T.E. which will subsequently generate vibration. More formally, "T.E. is the difference between the angular position that the output shaft of a drive would occupy if the drive were perfect and the actual position of the output." In practical terms, we take successive angular positions of the input, calculate where the output should be, and subtract this from the measured output position to give the "error" in position. Measurements are made by measuring angular displacements and so the answers appear initially in units of seconds of arc. It is possible to measure T.E. semi-statically by using dividing heads and theodolites on input and output and indexing a degree at a time but this is extremely slow and laborious though it can be the only possible way for some very large gears. Although the measurements are made as angular movements the errors are rarely given as angles as it is much more informative to multiply the error angle (in radians) by the pitch circle radius to turn the error into microns of displacement. Such errors are rather small typically only a micron or two even for mass produced gears such as those in cars. There is, unfortunately, some uncertainty as to whether we should multiply by pitch circle radius to get tangential movement at pitch radius or multiply by base circle radius to get movement along the pressure line, i.e., normal to the involute surfaces. Either is legitimate but we usually use the former since it ties in with the standard way of defining pitch and helix errors between teeth. However, from a geometric aspect, to correspond with profile error measurements (which are normal to the involute), the latter is preferable. The great advantage of specifying T.E. as a linear measurement (typically less than 5 um) is that all gears of a given quality, regardless of size of tooth module or pitch diameter, have about the same sizes of error so comparisons are relatively easy.

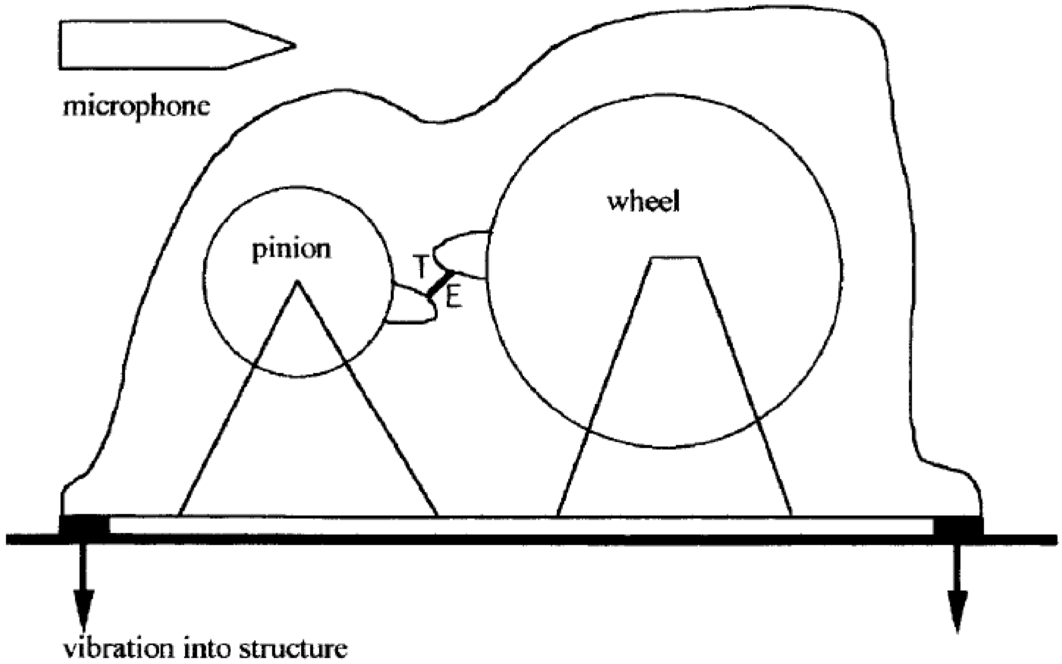


Figure 3.3 Transmission error excitation between gears.

It seems utterly ridiculous that a 1 mm module (25DP) gear less than an inch diameter will have roughly the same I.E. as a 25 mm module (1DP) wheel some 3 metres diameter of the same quality, but this is surprisingly close to what happens in practice (the module is the pitch circle diameter of the gear in millimetres divided by the number of teeth). This unexpected constant size of errors is liable to cause problems in the future with the current trend towards "micromechanics". If a gear tooth is only 20 u.m tall, the base pitch is about 20 ^im but errors of 2 fim in pitch or profile are still likely with corresponding T.E. errors so that a speed variation of 10% becomes possible.

Having defined T.E., we are left with a mental picture either of the “unwrapping string” varying in length or, as sketched in Fig. 3.3, of a smallbut energetic demon between the gear teeth surfaces imposing a relative vibration. For most noise purposes it is only the vibrating part of the T.E. that is important so any steady (elastic) deflections are ignored.

3.1.3 Gearbox İnternal Responses

T.E. is the error between the gear teeth. This idea of a relative displacement (microns) being the cause of a force variation and hencevibration is unusual since traditionally we excite with an external force such as an out of balance or vibrate the supporting ground to produce a vibration.

In gearing we have a relative displacement (the T.E.) between the mating gears generating the forces between the teeth and the subsequent vibrations through the system.
The relative displacement between the teeth is generated by equal and opposite vibrating forces on the two gear teeth surfaces, moving them apart and deflecting them a sufficient distance to accommodate the T.E. When we consider the internal responses of the gearbox, the input is the relative vibration between the gear teeth and the outputs (as far as noise is concerned) are the vibration forces transmitted through the bearings to the gearcase. In general the "output" force through each bearing should have six components: three forces and three moments, but we usually ignore the moments as they are very small and the axial forces will be negligible if there are spur gears, double helicals, or thrust cones. Single helical gears (and right angle drives) give axial forces and, unfortunately, the end panels of gearcases are often flat and are rather flexible. The resulting end panel vibrations are important if it is the gearcase which is producing noise, but of little importance if it is vibration through the mounting feet that is the principal cause.
Occasionally vibrating forces will transmit along the shafts to outside components and radiate noise. A ship's propeller will act as a good loudspeaker if directly coupled to a gearbox, but insertion of a flexible elastomeric coupling will usually block the vibration effectively, provided it has been correctly designed for the right frequency range. Similarly, in wind turbines, the propellors can act as surprisingly effective loudspeakers so it is necessary to have good isolation between blades and gears. In a car, the trouble path can be upstream or downstream, as vibration from the gearbox travels to the engine and radiates from engine panels, or escapes through the engine mounts to the body shell, or travels to the rear axle and through its supports to the body. At one time the vibration also travelled directly via gear levers and clutch cables into the body shell.
The assumption usually made is that, when modelling internal resonances and responses, the bearing housings can be taken as rigid. This is usually a reasonable idealisation of the situation since bearing housing movements are typically less than 10% of gear movements. Occasionally a flexible casing, or one where masses are moving in antiphase, will give the effect of reducing or increasing the apparent stiffness of supporting shafts or bearings.
Gears are sometimes assumed to vibrate only torsionally but this assumption is wildly incorrect due to bearings and to shaft deflections so any model of gears must allow for lateral movement (i.e., movement perpendicular to the gear axis). Masses are known accurately and stiffnesses can be predicted or measured with reasonable precision, but there are majör problems with damping which cannot be designed or predicted reliably.

3.1.4 External Responses

The path of the vibration from the bearing housings to the final radiating panels on either the gearcase or external structure is usually complex. Fortunately, although prediction is difficult and unreliable due to damping uncertainties it is relatively easy to test experimentally so this part of the path rarely gives much trouble in development. One of the first requirements is to establish whether it is the gearcase itself which is the dominant noise source or, more commonly, whether the vibration is transmitted into the main structure to generate the noise. Transmission to the structure is greatly affected by the isolators fitted between the gearbox and the structure. There is liable to be a large number of parallel paths for the vibration through the structure and an extremely large number of resonances which are so closely packed in frequency that they overlap. A statistical energy approach [6] with the emphasis on energy transmission and losses over a broad frequency band can give a clearer description than the conventional transfer function approach when frequencies are high and there are multiple inputs and resonances. In a very large structure the conventional ideas of resonant systems are no longer so relevant and the transmission of energy has more in common with ideas of propagation of stress waves.

3.1.5 Overall path to noise

The complete vibration transmission path is shown in Fig. 3.4. It starts from the combination of manufacturing errors, design errors and tooth and gear deflections to generate the T.E. Though manufacturing errors are usually blamed it is more commonly design that is at fault. The T.E. is then the source of the vibration and it drives the internal dynamics of the gears to give vibration forces through the bearing supports. In turn, these bearing forces drive the external gearcase vibrations or, via any isolation mounts, drive the external structure to find "loudspeaker" panels. In a vehicle, after the vibration has travelled from the gearbox through the engine main casting to the support mounts and hence to the structure, it may travel several metres in the body before exciting a panel to emit sound that annoys the occupants. Vibration travelling along te input and output shafts to cause trouble can aalso occur but is less common.

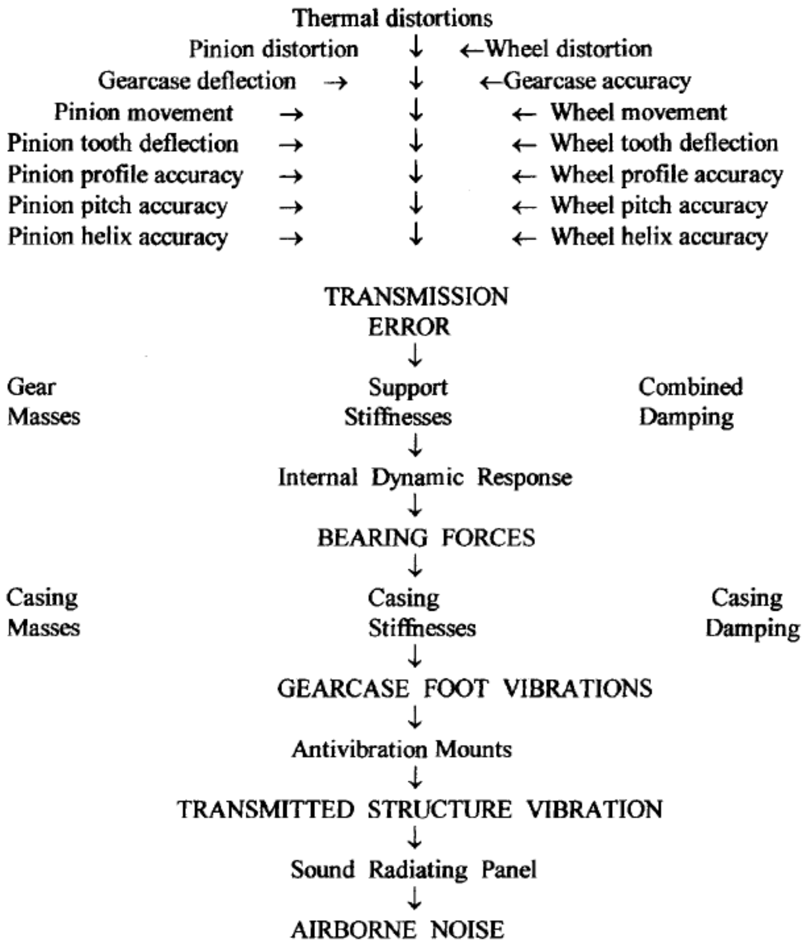


Figure 3.4 Vibration excitation and transmission path.

3.1.6 T.E. - noise relationship

It is very difficult for a traditional gear engineer trained to think in terms of pitch, profile, and helix measurements to change over to ideas of single flank checking, i.e., T.E., especially as T.E. is not relevant for gear strength. The change is not helped by the difference that the traditional methods are methods where the gears are stationary on expensive machines in the metrology lab whereas T.E. is measured when the gears are rotating and can be done on a test rig out in the main works or sometimes even on the equipment while running normally.

However, the basic idea is that pitch, profile and helix errors may combine with tooth bending, gear body distortions and whole gear body deflections to give an overall relative deflection (from smooth running) at the meshpoint between the gears. It is also difficult to convince gear engineers that there is a very big difference between roll (double flank) checking, which is extremely cheap and easy, and T.E. (single flank) checking since they give rather similar looking results. Unfortunately, there are a large number of important gear errors which are missed completely by roll checking so this method should be discouraged except for routine control of backlash. The problems with double flank measurement arise from the basic averaging effect that occurs. Any production process or axis error in transfer from machine to machine may produce errors which give +ve errors on one flank which effectively cancel -ve errors on the facing flank. The resulting centre distance variation is negligible but there may be large (cancelling) errors on the drive and overrun flanks. Shavers and certain types of gear grinders are prone to this type of fault which is worse with high helix angle gears.

The question then arises as to the connection between T.E. and final noise. Few practising engineers initially believe the academics' claim that noise is proportional to T.E., although the system normally behaves (except under light load) as a linear system. For any linear system the output should be proportional to input. Doubling the T.E. should give 6dB increase in noise level or, with a target reduction of 10 dB on noise, the T.E. should be reduced by VlO, i.e., roughly 3. This only applies at a single frequency and different frequencies encounter high or low responses en route so a major visible frequency component in the T.E. may be minor in the final noise because it could not find a convenient resonance.

Tests over 20 years ago [7,8] established the link, and recent accurate work by Palmer and Munro [9] has confirmed the exact relationship by direct testing and shown how the noise corresponds exactly to the T.E. Since most companies flatly refuse to believe that there is a direct link between noise and T.E., it is common for companies to re-invent the wheel by testing T.E. and cross-checking against testbed noise checks. This is apparently very wasteful but has the great advantage of establishing what T.E. levels are permissible on production, as well as giving people faith that the test is relevant. For this learning stage of the process it is simplest to borrow or hire a set of equipment to establish relevance before tackling a capital requisition or to take sets of gears for test to the nearest set of equipment. Unfortunately, those few firms who have T.E. equipment usually use it very heavily so it may be better to ask a university if equipment can be hired. Newcastle [10] ,Huddersfield [11] and Cambridge [12] in the U.K., Ohio State University [13] and other researchers [14, 15, 16] have developed their own T.E. equipment and are usually happy to provide experience as well as a foil range of equipment and analysis techniques.

Academic equipment based on off-line analysis is often, however, not suited to high speeds or mass production.

3.2 HARRIS MAPPING FOR SPUR GEARS

3.2.1 Elastic deflections of gears

The basic geometric theory for spur gears assumes the "unwrapping string" generation of a perfect involute. We can then replace the two mating involute curves with a string unwrapping from one base circle and coiling onto the other base circle as in Fig. 3.5.

A contact between one pair of mating teeth should then travel along the "string," the "pressure line" or "line of contact" until it reaches the tip of the driving gear tooth. To achieve a smooth take-over, before one contact reaches the tip there must be another contact coming into action, one tooth space behind. For the theoretical ideal of a rigid gear the only requirement for a smooth take-over is that the base pitch, the distance between two successive teeth along the pressure line, should be exactly the same for both gears.

Unfortunately, although gear teeth are short and stubby, they have elasticity and there are significant deflections. The deflection between two teeth is partly due to Hertzian contact deflections, which are non-linear, but mainly due to bulk tooth movement because the tooth acts as a rather short cantilever with a very complex stress distribution and some rotation occurs at the tooth root. A generally accepted Figure for the mesh stiffness of normal teeth is 1.4 x 10 N/m/m or 2 x 10 IbFin/in, a Figure used by Gregory, Harris and Munro [17] in the late 1950s but one which has stood the test of time. As a rough rule of thumb we can load gears to 100N per mm of face width per mm module so a 4 mm module gear 25 mm wide might be loaded to 10,OOON (1 ton). This load infers a deflection of the order of 400/1.4 x 107 m or 28.6 pm (1.1 mil).

Experimental measurement of this rather high stiffness has proved extremely difficult both statically and dynamically even with spur gears so that we are mainly dependent on finite element stressing software packages to give an answer. There is a significant effect at the ends of gears since the ability to expand axially reduces the effective Young's modulus and high angle helical gears have reduced contact support at one end and additional buttressing at the other end.

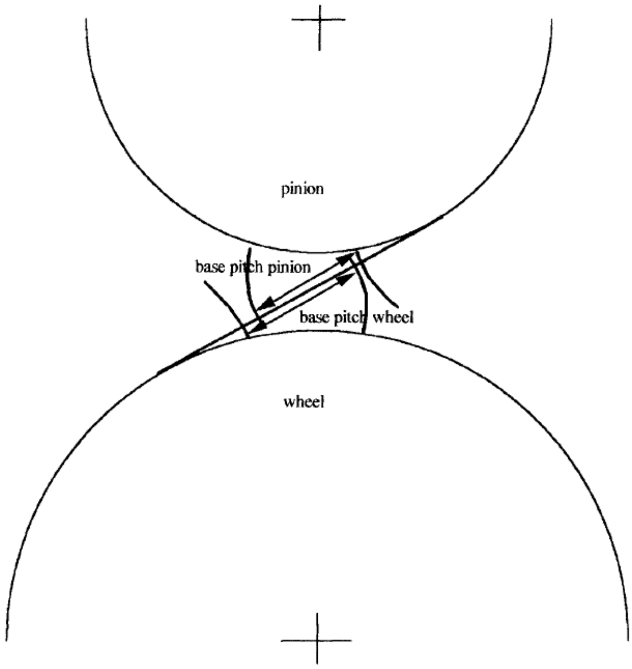


Figure 3.5 Handover of contact betweeen successive teeth.

Different manufacturing methods produce different root shapes and affect stiffness, but the main variations arise from variation of pressure angle or undercutting and, to a lesser extent, from low tooth numbers.
The stiffness of each tooth varies considerably from root to tip, but with two teeth the effects mainly cancel. The highest combined stiffness occurs with contact at the pitch points and the stiffness decreases about 30% toward the limits of travel but the decrease is highly dependent on the contact ratio and gear details. In practice it is unusual for the applied load to be completely even across the face width as this implies that helix and alignment accuracies, and gear body deflections, must sum to less than a few fim.

As a result, we have to allow for typically up to 100% overload and deflection at either end of the tooth, or in the middle if crowned, so deflections can be large. Using the ruleof thumb that conventional surface-hardened teeth may be loaded to 100 N/mm facewidth/mm module, the above 4mm module gear (6 DP) loaded to 400 N/mm would deflect 400/14, i.e., 28 urn, nominally but, allowing for load concentrations, this could rise to 50 um (2 mil).

3.2.2 Reasons for tip relief

Since there is deflection of the mating pair of teeth under load, it is not possible to have the next tip enter contact in the pure involute position because there would be sudden interference corresponding to the elastic deflection and the corner of the tooth tip would gouge into the mating surface. Manufacturing errors can add to this effect so that it is necessary to relieve the tooth tip (Fig. 3.6) to ensure that the corner does not dig in. Correspondingly, at the end of the contact, the (other) tooth tip is relieved to give a gradual removal of force. High loads on the unsupported corner of a tooth tip would give high stresses and rapid failure, especially with casehardened gears which might spall (crack their case). In addition a sharp corner plays havoc with the oil film locally as the oil squeezes out too easily allowing metal to metal contact and accelerated failure. Tip relief design was traditionally a black art but can be determined logically.

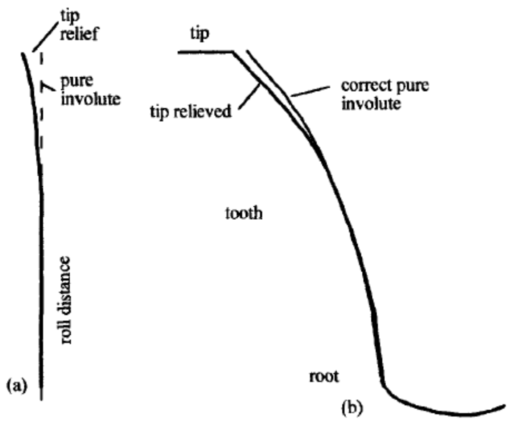


Figure 3.6 Picture of tip relief showing deviation from an involute in (a) and typical tooth shape (b).

The amount of "tip relief needed in the example above can be estimated by adding the worst case elastic deflection, for example, 28.6um + 70% (to allow for misalignment), to the possible base (adjacent) pitch errors of 3 um on each gear and to the possible profile errors of 3 um on each gear.

The total tip relief needed is then 61 jim (2.5 mil). There can be some extra tip relief correction required if there is a large temperature differential between two mating gears, as one base pitch grows more than the other due to thermal expansion, but the effect is usually very small [18]. This "tip relief can be achieved by removing metal from the tip or the root of the teeth or from both. There are two main schools of thought. The traditional approach was to give tip and root relief, as indicated in Fig.3.7, with a rather arbitrary division between the two and with the tip and root relief meeting roughly at the pitch point. The actual shape of the relief, as a function of roll angle, which is directly proportional to roll distance, tends to be almost parabolic.

There are two problems with this approach. It is not immediately clear where the tip of the mating tooth will meet the lower part of the working flank so it is more difficult to work out how much the effective root relief is at the point where the mating tip meets the flank. Rather more important is the fact that this parabolic shape of relief is not desirable from either noise aspects and for helical gears is undesirable from stressing aspects.

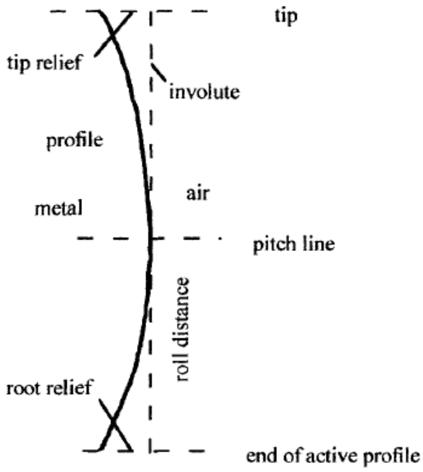


Figure 3.7 Tip and root relief applied on a gear.

In practice, we usually wish to have relief varying linearly with roll angle, starting at a point on the flank well above the pitch point so that there is a significant part of a tooth pair meshing cycle where two "correct" involutes are meeting.

When discussing profile corrections there are initially two uncertainties about the specifications. The first is whether the relief quoted is in the tangential direction or whether in the direction of the line of action. As the difference is normally only 6% on standard gears it is not important but most traditional profile measuring machines measure normal to the involute (i.e., in the direction of the line of action) and it is the movement or error in this direction that gives the vibration excitation so we usually specify this. When using a 3-D coordinate measuring machine it is again better to work in the direction of the line of action. The other possible uncertainty is determining the position of a point up the tooth flank. The obvious choices of distance from root or tip are irrelevant as the profile ends are not accurate.

Specifying actual radius is of little help in locating the correct points and referencing them to gear rotation. What is done in practice is to work in terms of roll distance. See Fig. 3.8. As the gear rotates and the "unwrapping string" leaves one gear base circle and transfers to the other there is a linear relationship between rotation and the distance that the common point of contact moves along the line of action. Roll distance is simply roll angle in radians times base circle radius. We measure and specify position in the tooth mesh cycle by giving the distance that the point of contact has travelled. Tooth flank starting and finishing points are unclear so design works in roll distance measured from the pitch point.

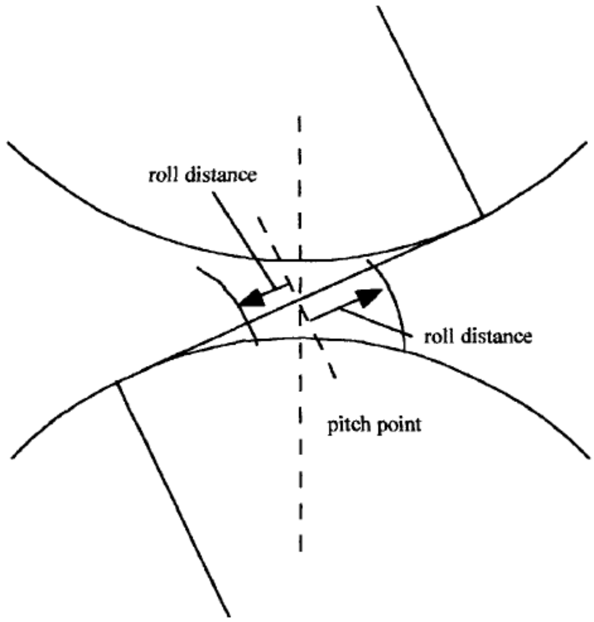


Figure 3.8 Unwrapping string model.

There is not a linear connection between roll distance and distance up the flank as can be seen from Fig. 3.9 which shows the "string" unwrapped at equal angular intervals and so equal distances along the line of action. Up the flank the distance intervals (between arrow tips) steadily increase. When giving experimental measurements of profile or of the design on a single gear of a pair it is usual to show the reliefs relative to a perfect involute which is a straight vertical line up the page. Roll distance is vertical and the reliefs (to large scale) are shown horizontally as in Fig. 3.7. However when we are looking at the meshing of a pair of teeth the picture is turned onits side as in Fig. 3.10 so that roll distances are horizontal and reliefs are vertical. There can be problems locating exactly where on an experimental profile measurement the pitch point occurs as it can only be located by an accurate knowledge of the pitch radius and this depends on the centre distance at which the pair of gears will run. The main choice in profile design is between giving both tip and root relief on the pinion so that the wheel (or annulus) stays pure involute for easy production or giving tip relief, but no root relief, on both, which is easier to assess and control. This choice can be controlled by production constraints of availability of suitable gear machines and cutters. In this book it is assumed that tip relief is given on both gears but there is no root relief to complicate the geometry.

A very special case arises for very large slow gears which have been in service for a while so

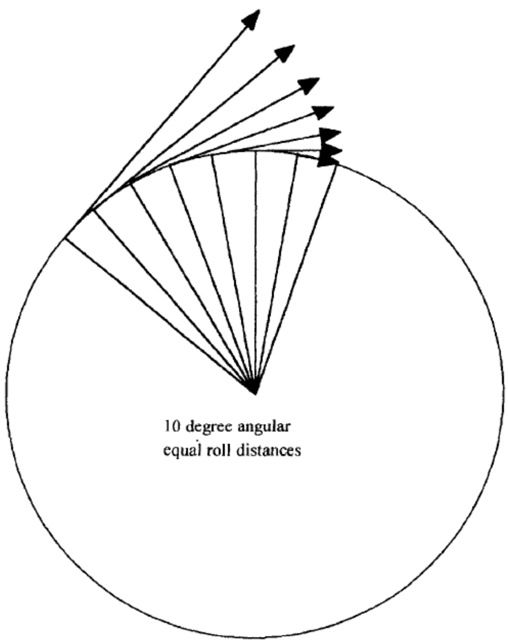


Figure 3.9 Effect of equal steps of roll on involute.

that both pinion and wheel have worn away from their original (involute) profile. The most economical repair is then to leave the wheel as it is and adjust the profile of the pinion to suit the now incorrect wheel.

3.2.3 Unloaded T.E. for spur gears

Fig. 3.10(a) shows diagrammatically what happens when we take two mating spur gear teeth, each with tip relief extending a third of the way down (but no root relief), and mesh them. All distances along the profile are in terms of roll distance, not actual distance, and so are proportional to gear rotation (multiplied by base circle radius).The horizontal line represents the pure involute and the two tooth profiles, shown slightly apart for clarity, follow the involute profile to above their pitch line where they are relieved. In this case the tip reliefs are linear, as is modern custom. The combination of two teeth with perfect involutes in the centre is to give zero T.E. for this part of the mesh. Where there is tip relief it is irrelevant which gear has it as either gives a drop in the T.E. trace for the combination.

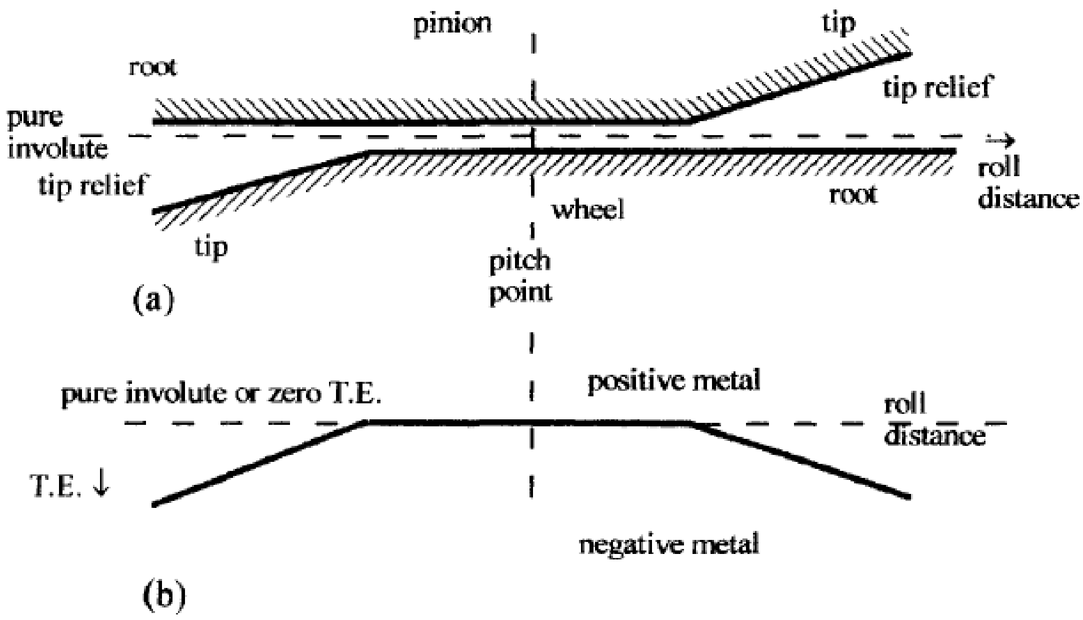


Figure 3.10 Effects of mating two spur gear profiles, each with tip relief.

T.E. traces are conventionally drawn with positive metal giving an upward movement but when testing experimentally the results can correspond to positive metal either way so it is advisable to check polarity. In the metrology lab this can simplest be done by passing a piece of paper or hair though the mesh.

The combined effect of one pair of teeth meshing under no load would be to give a T.E. of the shape shown in Fig. 3.10(b) with about one third of the total span following the involute for both profiles and generating no error. The tip reliefs then give a drop (negative metal) at both ends. The same effect is obtained if the relief is solely on the pinion at tip and root.

However, the geometry is more complex at the root as the mating tip does not penetrate to the bottom of the machined flank. Putting several pairs of teeth in mesh in succession gives the effect shown in Fig. 3.11 (a). If there are no pitch or profile errors and no load applied so no elastic deflections, the central involute sections will be at the same level (of "zero" T.E.) and part way down the tip relief there will be a handover to the next contacting pair of teeth. One base pitch is then the distance from handover to handover. When we measure T.E. under no-load conditions we cannot see the parts shown dashed since handover to the next pair of teeth has occurred.

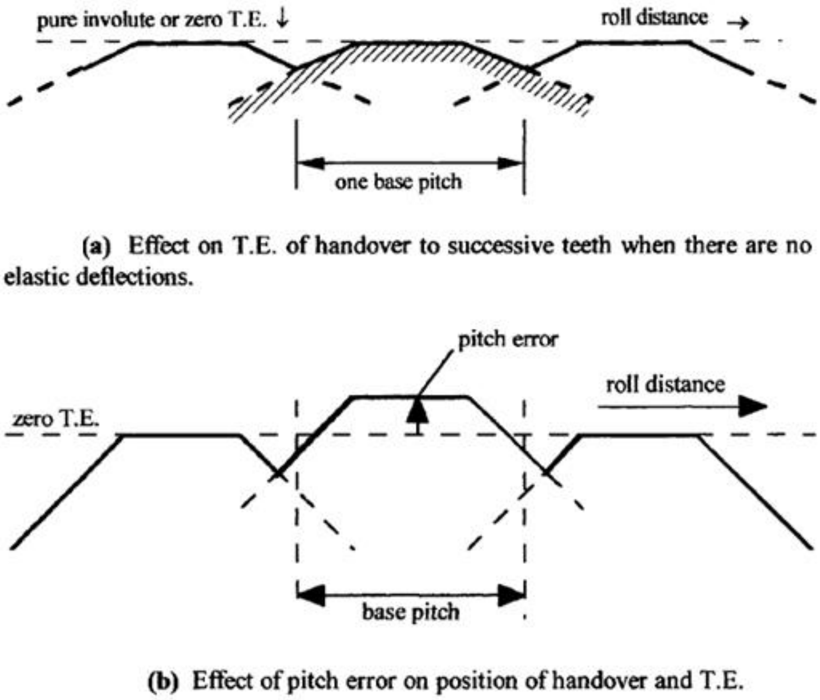


Figure 3.11

Fig. 3.11(b) shows the effect of a pitch error which will not only give a raised section but will alter the position at which the handover from one pair to the *next* occurs,

3.2.4 <u>Effect of load on T.E.</u>

We wish to predict the T.E. under load as this is the excitation which will determine the vibration levels in operation. As soon as load is applied there are two regimes, one around the pitch point where only one pair of teeth are in contact and one near the handover points where there are two pairs in contact, sharing the load but not, in general, equally. The total load remains constant so, as we are taking the simplifying assumption that stiffness is constant, the combined deflection of the two pairs in contact must equal the deflection when just one pair is in contact. In particular, exactly at the changeover points, the loads and deflections are equal if there are no pitch errors so each contact deflection should be half the "single pair" value.

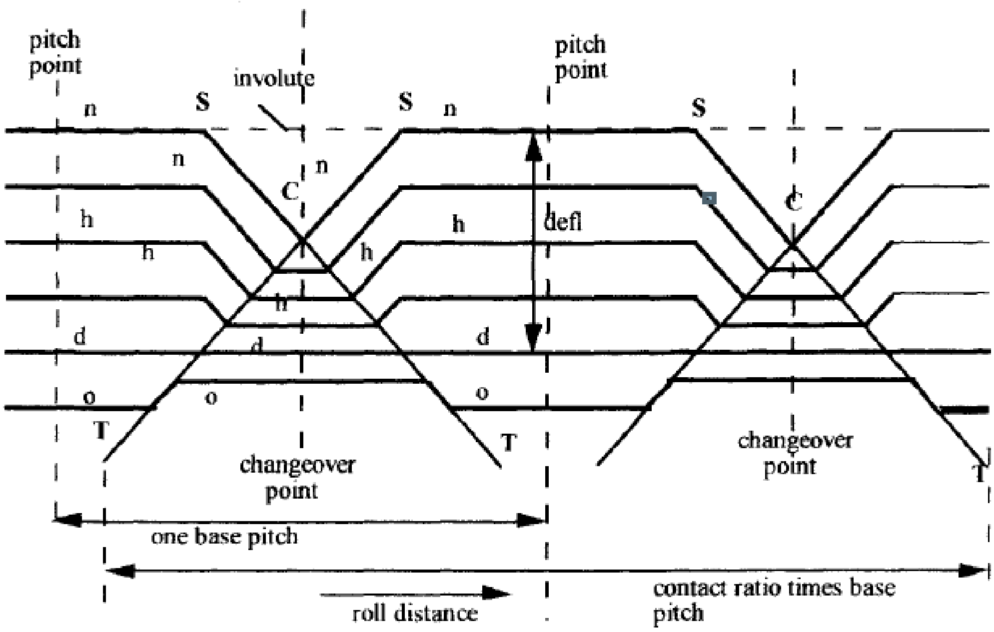


Figure 3.12 Harris map of interaction of elastic deflections and long tip relief.

This explanation of the handover process was developed by Harris [19] and the diagrams of the effects of varying load are termed "**Harris maps**" .Fig. 3.12 shows the effect. The top curve (n) is the T.E. under no load and then as load is applied the double contact regime steadily expands around the changeover point. Curve (h) is the curve for half "design" load. *At* aparticular "design load" the effects of tip relief are exactly cancelled by the elastic deflections (curve d) so there is no T.E. There is a downward deflection (defl) away from the "rigid pure involute" position but, as the sum of tip relief and deflection is constant, it does not cause vibration.

Above the "design" load the single contact deflections are greater than the combined double contact plus tip relief deflections. The result is as shown by curve (o) with a "positive metal" effect at changeover. Varying stiffness throughout the mesh alters the effects slightly, but the principle remains. In this approach it should be emphasised that "design" load is the load at which minimum T.E. is required, not the maximum applied load which may be much greater. Since the eventual objective is to achieve minimum T.E. when the drive is running under load, there will normally be a desired design T.E. under (test) no-load. This leads to the curious phraseology of the "error in the transmission error," meaning the change from the desired no-load T.E. which has been estimated to give zero-loaded T.E.

3.2.5 Long, short, or intermediate relief

In 1970, Neimann in Germany [20] and Munro in the U.K. introduced and developed the ideas of "long" and "short" relief designs for the two extreme load cases where the "design" load is full load or is zero load. Fig. 3.12 shows the variation of T.E. with load for a "long relief design" which is aimed at producing minimum noise in the "design load" condition. Specifying the tip relief profile begins with determining the tip relief at the extreme tip points T, making the normal assumptions about overload due to misalignment and manufacturing errors. The necessary relief at the crossover points C (where contact hands over to the next pair of teeth at no-load) is half the mean elastic deflection and here we do not take manufacturing errors into account. Typically the relief at T may be 3 to 4 times that at C. The crossover points C are spaced one base pitch apart and the tip points T are spaced apart the contact ratio times a base pitch. It is, of course, simplest if the tip reliefs (which should be equal) are symmetrical. The start of (linear) tip relief is then found by extending TC backwards till it meets the püre involute at the point S.

An alternative requirement is to have a design which is quiet at no load or a very light load since this is likely to occur for the final drive motorway cruising condition or when industrial machinery is running light, as often happens.

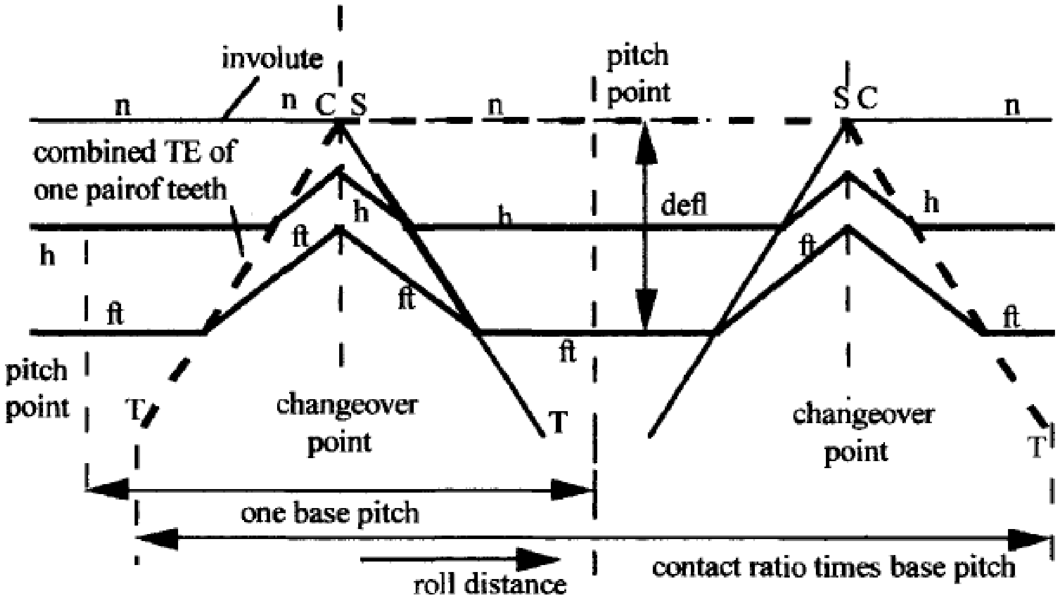


Figure 3.13 Harris map of deflections with a "short" tip relief design.

The "design" condition is zero load so we require "short relief as shown in Fig. 3.13 , which shows the variation of T.E. with load for "short" tip relief.
The pure involute extends for the whole of a base pitch so there is no tip relief encountered at all at light load (n). The tip relief at T must, however, still allow for all deflections and errors. As load is applied we are then exceeding "design" load of zero and there will be considerable T.E. with high sections at the changeover points. Curve "ft" is the full torque curve where there is a section at changeover with double contact and hence half the deflection (defl) from the pure involute that occurs near the pitch points. Palmer and Munro [9] succeeded in getting very good agreement between predicted and measured T.E. under varying load in a test rig to confirm these predictions.
Care must be taken when discussing "design load" in gearing to define exactly what is meant because one designer may be thinking purely in terms of strength so his "design" load will be the maximum that the drive can take. If, however, noise is the critical factor, "design load" may refer to the condition where noise has to be a minimum and may be only 10% of the permitted maximum load. If the requirement is for minimum noise at, for instance, half load, then the relief should correspondingly be a "medium" relief. The short or long descriptions refer to the starting position of the relief, but the amount of relief at the tip of each tooth remains constant.

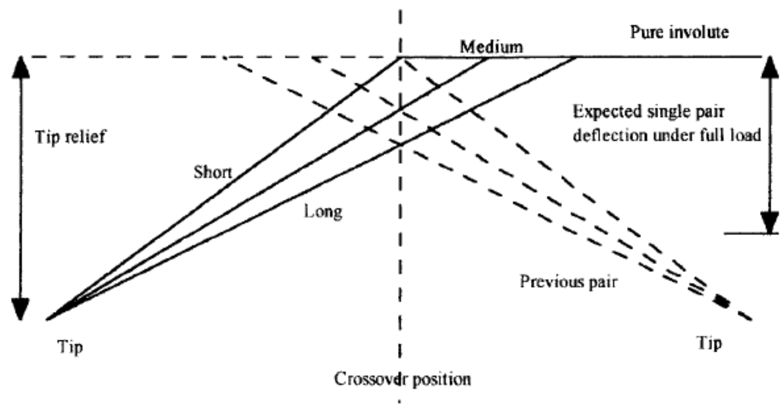


Figure 3.14 Tooth relief shapes near crossover for low, medium, and high values of design quiet load in relation to maximum load.

Fig. 3.14 shows for comparison the three shapes of relief near the crossover point for the conditions of the design quiet condition being zero, half and full load. For standard gears with a contact ratio well below 2 it is only possible to optimise for one "design" condition but as soon as the contact ratio exceeds 2 then there can be two conditions in which zero T.E. is theoretically attainable.

4 TRANSMISSION ERROR THEORY AND CONSTRUCTION

Spur gear design criteria include two main points of static and dynamic performances. Static performance usually deals with global kinematic requirements, geometrical requirements and strength analysis of gear materials under load. Dynamic performance of gears is just as important as performance under static conditions during meshing. It deals with gear noise and vibration which are key factors for smooth meshing and quietness. Any non-uniformity in motion transfer cause gear noise and vibration under load.

Theoretically a pair of gear with the perfect involute tooth profiles and tooth spacing transmits a uniform motion between the shafts. But in reality motion is not transmitted uniformly. Due to deflection of the teeth under load, interference, in other words corner contact occurs between the teeth of the incoming pair and this causes impact loads on the teeth and instantaneous non uniform motion transfer between the gears. When the gears run at high speed these impact loads and non uniform motion transfer causes high dynamic tooth loads, vibration and the generation of noise. The instantaneous error in uniform motion transfer is called the **Transmission Error(TE) (**Figure 4.1a) and it is defined as "*the difference between the actual position of the output gear and the position it would occupy if the gears were perfectly conjugate"*. This definition is valid for both loaded and unloaded gears. It is the TE under load which causes the problem at high speed whether it is due only to tooth deflection, manufacturing errors or both [22]. TE curves are constructed for different design loads (Figure 4.1b).

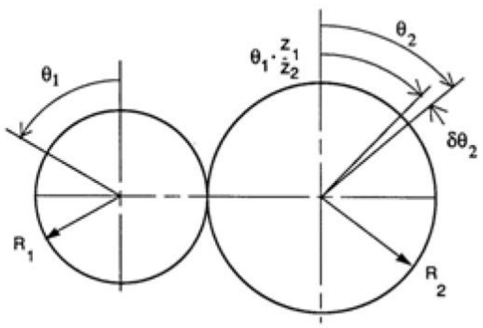


Figure 4.1a TE

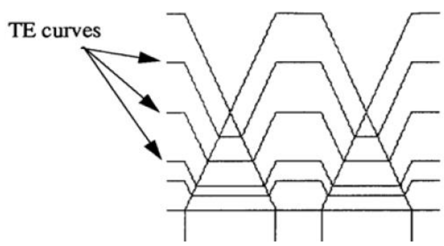


Figure 4.1b TE curves

The mathematical formulation of TE in linear units is as follows:

$$TE = r_{b2} X (\theta_2 - \frac{T_1}{T_2} X \theta_1) \quad (4\text{-}1)$$

θ_1: Angular rotation of the input gear

θ_2: Angular rotation of the output gear

T_1: Tooth number of the driving gear

T_2: Tooth number of the driven gear

r_{b2}: Base radius of the output gear

The static transmission error is strongly related to the kinematic accuracy of gear trains, dynamic load of gear tooth, and gear noise. It is defined as the difference between the theoretical position of the output gear, with perfect geometric accuracy and rigid drive, and the actual output position at a low enough speed [22]. In a gear set, TE is defined as the difference between the effective and the ideal position of the output shaft with reference to the input shaft. The ideal position represents a condition of perfect gear box, without geometrical errors and deflections. TE can be expressed either by an angular displacement or, more conveniently, as a linear displacement measured along a line of action at base circle [22]. Transmission error is one of main excitation source of gears and it has possible negative effects on gear operation namely high vibration levels transmitted via gear body, shaft, bearing, housing and casing, high noise levels of both air borne and structure borne types, kinematic inaccuracy causing problems in accurate positioning applications. TE under design load is the main cause of all the negative effects and has to be minimized in amplitude and optimized in shape to get a smooth TE curve. For a smooth meshing during gear operation, the production of gears with minimum geometrical errors is first step and very critical to reduce TE value. But loading effect or gear tooth flexibility under load cause to TE for an even error free gear pair. Adjacent pitch error which and tooth deflection both cause corner contact (CC) or can be called as interference It is effective at the start and end of mesh. In addition, both adjacent PE and the teeth deflection will cause a position variation of the output gear hence a TE of gears. Corner contact causes an instantaneous variation in TE curve as well as the physicaldamage of tooth surface under heavy loads. An intentional profile modification is very vital to avoid interferences. For this purpose, the tip of the driven flank or the root of the driving flank or both has to be modified. A definite amount of material is removed to provide a smooth meshing of gear pair. Intentional profile modifications of gears at tooth tips of both driving and driven gears are illustrated in Figure 4.1a and b respectively. Profile modification for linear type can be characterized by two parameters namely amount of relief and extent of relief (Figure 4.4). Amount of relief is the thickness of material removed from tip of the tooth in a plane normal to the involute curve. Extent of relief shows how far the relief extends down the tooth from the tip (Figure 4.4). Extent of relief, although can be measured in roll angle, is usually given as linear dimension between start and end points of the relief along the path of contact. Extent of relief in linear units is calculated by multiplying the required roll angle difference between start and end of relief by the base radius (rb) in equation 4.1. Gear operation at speeds away from the system's resonant speeds is important. However, systems will generally be multiple degree of freedom systems with multiples of coupled and uncoupled resonant frequencies. In addition, the resonance is not hit by only the main excitation frequency but also by its harmonics and sub harmonics. There is a direct relationship between static TE values and vibration and noise levels. Smoothing the static TE curve under operating load, either by profile relief or some other means, comes front to avoid harmonics hitting problem and high vibration and noise levels.

Construction of loaded and unloaded TE curves in linear units for a pair of gears with profile relief is shown in Figure 4.1 e. For ease understand of topic, it is assumed that the gears in

$$\text{Extent of relief} = r_b * (roll\ angle_{end} - roll\ angle_{start}) \tag{4-2}$$

mesh have no adjacent pitch errors, tooth pair stiffness is constant and transmitted load is also constant. The vertical axis refers to TE in linear units while horizontal axis represents the nominal positions of contact along the line of action. When the contact point is in single tooth pair region (SC) all the load is carried by that one pair and when the contact point is in double tooth pair region (DC) the total load is shared by two pairs (regardless of whether equally or not). Applying a proper relief on the gear teeth and also constructing the expected static TE curve under load to check the suitability of that relief is very essential. Loaded and no load static TE curve is constructed for a constant tooth load and constant tooth pair stiffness with a linear type of profile relief (Figure 4.1 e).

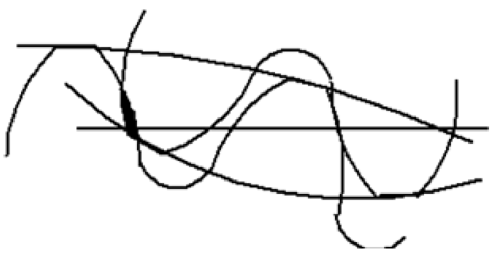


Figure 4-1 c

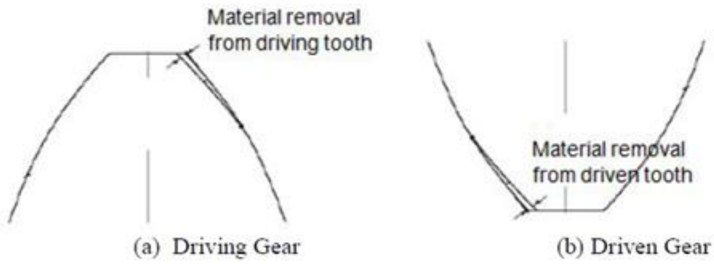


Figure 4.1 d

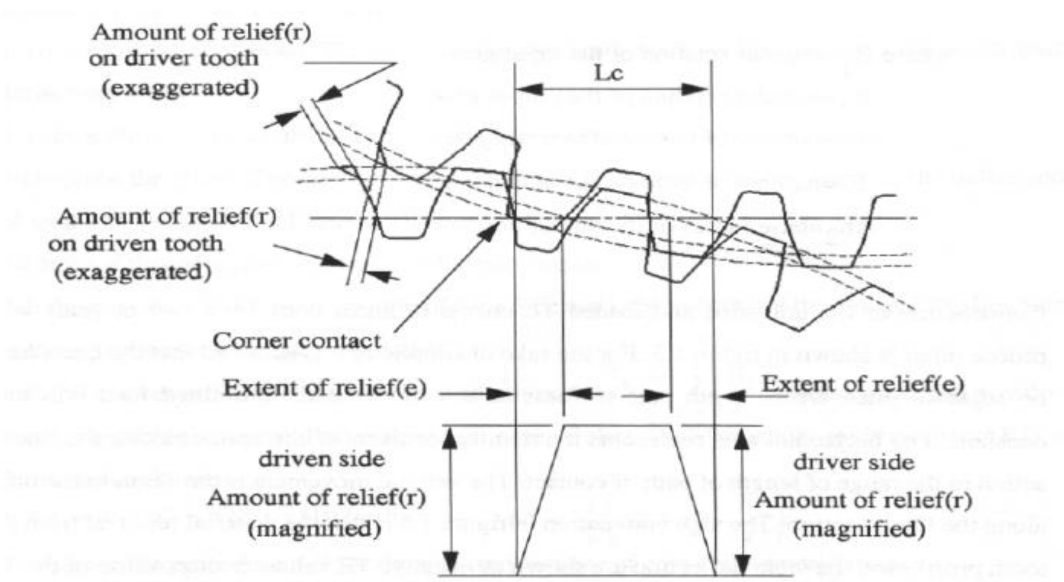


Figure 4.1 e

4.1 SPUR GEAR TE THEORY AND QUASI STATIC TE CURVES

4.1.1 Isolated TE Curve

Before going any further we must look at the basic geometry of tooth contact, as shown in figure 4.2. The base radii of the gears are O1T1 and O2T2 respectively, and the T1T2 is the common tangent to the base circles. The tip radius of the driving gear is O1D and that of the driven gear O2A.If the gears rotate in the directions shown, then tooth contact starts at point A and finishes at point D. The line AD is called line of contact Lc.

Path of contact a perfect involute tooth pair can be represented by a horizontal line, line AD, and TE is the deviation from this line. To obtain the TE curve of a gear pair, isolated TE curve of each tooth pair of the meshing gear pair must be used.

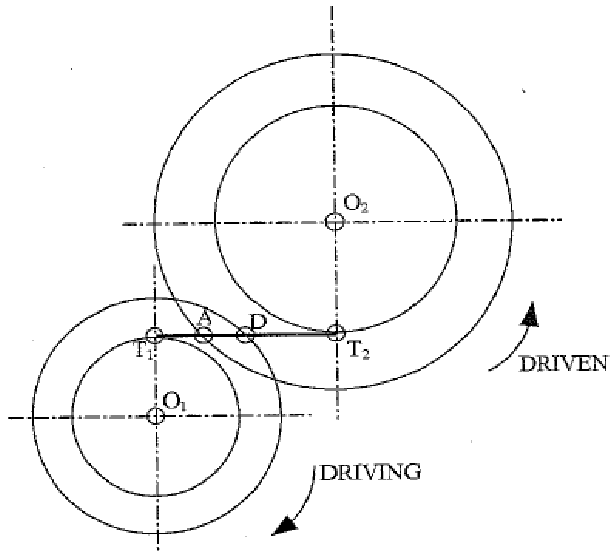


Figure 4.2 Defining the path of contact AD.

Consider the gear pair to have only one tooth on each gear , all the others are temporarily removed. Then the TE curve for that pair will be as shown in figure 4.3 for the case of no-load.

Figure 4.3 Isolated TE curve for a Tooth pair.

Next step is to draw the isolated TE curves of other tooth pairs of the gears to the same scale by displacing each one base pitch(pb) apart along the horizontal axis as seen in figure 4.4. In this figure each isolated TE curve must be at vertically same level, but here one of two consecutive line is drawn to be shifted down for ease of understand and observe.

There are four important points during mesh of a pair of LCR spur gear. These points are marked to be A,B, C, and D on figure 4.4. Point A is the initiation point of contact for tooth pair 2. Between points A and B, load is shared between first and second tooth pairs, means this region is Double Contact region. At point B contact of first tooth pairs ends up. From point B, to point C, load is carried by only second tooth pair, so this region is called as Single Contact region. At point C contact initiation for third pair takes place and a new DC region starts, which will continue until point D. At this point contact of second tooth pair vanishes.[22]

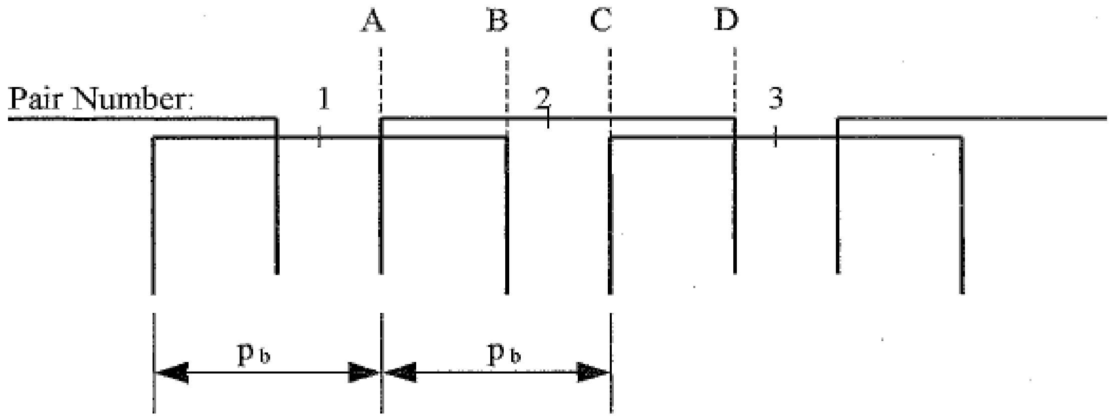


Figure 4.4 Tooth Boundary Profiles.

4.1.2 <u>Construction Continuous TE Curve for Spur Gear</u>

Isolated TE curve gives the TE curve of the mating gears if other tooth pairs were to be removed temporarily thus leaving that particular tooth pair in mesh only. For angular positions inside the path of contact of that particular tooth pair the TE is finite but outside the path of contact the TE gets infinite in negative direction. This means the driven gear does not follow the driving gear. However, in practice there is the next pair of tooth to take over the drive thus preventing the large negative TE. The path of contact of every neighbouring tooth pair overlap each other by about the distance of A-B and C-D. Thus , the isolated TE curves of the neighbouring tooth pairs have to be overlapped as well.

The first step of constructing continuous TE curve is to produce the isolated TE curves of each tooth pair. By doing that , a series of isolated TE curves will be collected and given the name of the tooth order numbers to prevent any confusion. The next step is to bring these collected TE curves together, spaced one base pitch apart as shown in figures 4.4 and 4.5 ,to simulate the overlapping action of the meshing tooth pairs along the path of contact. In figure 4.5 , s_i represents start of engagement of ith tooth pair while ei represents end of engagement of ith pair. The no load TE curve now is the combination of uppermost part of isollated TE curves, shown as thick line in figure 4.5 with label “No Load”. [22]

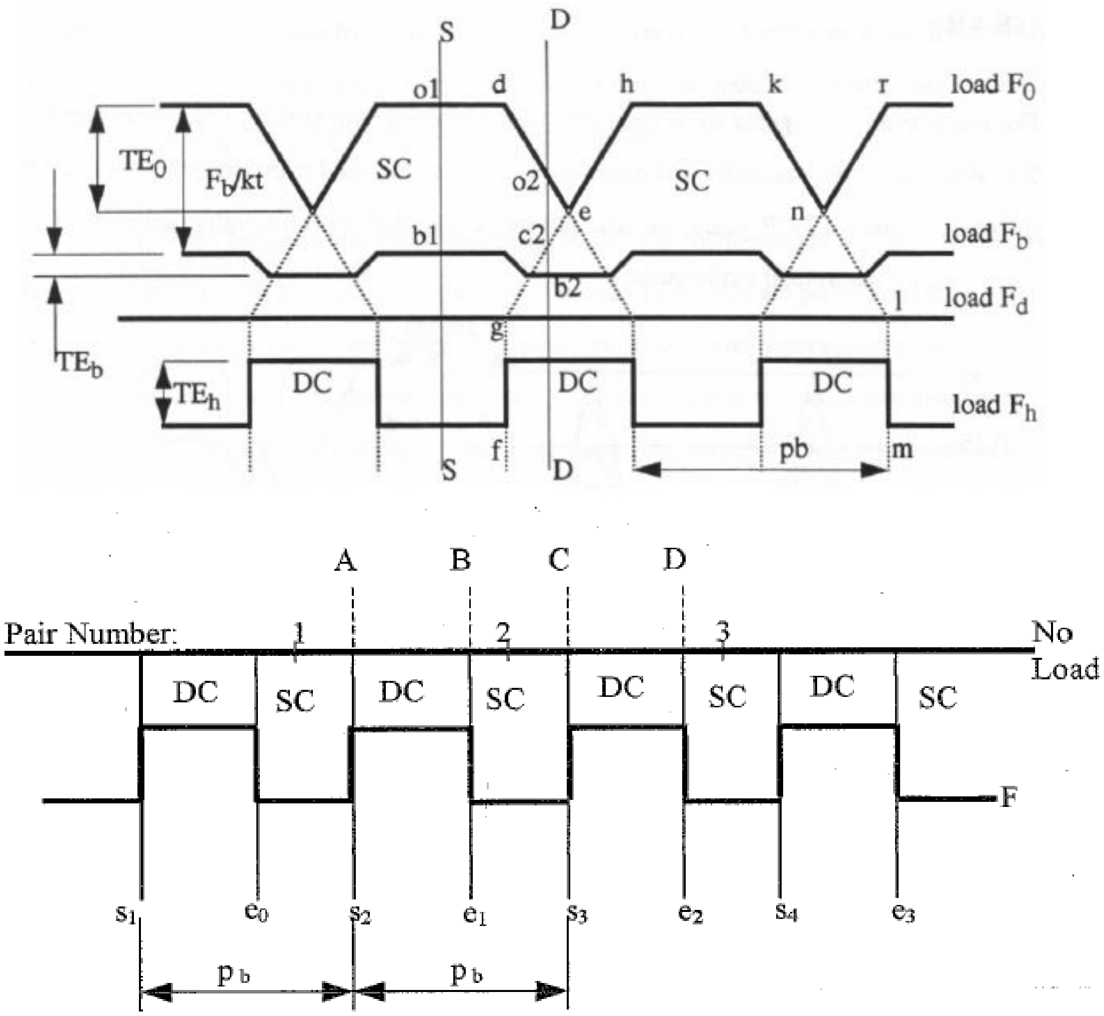


Figure 4.5 Continuous TE curves for spur gears.

The procedure for constructing the unloaded and loaded TE curves as follows [23]:

a) Construct the boundary profiles of the mating tooth pairs such as curves f-gh-k-l-m.

b) Displace them by the base pitch of the gears along the horizontal axis.

c) Construct the no load TE curve by following the top boundary borders of the each tooth boundary profile, curve d-e-h-k-n-r.

d) To construct the loaded TE curve under any load, such as design load Fb, calculate the tooth pair deflection, $\frac{F_b}{k_t}$ **(kt=k*b)**

e) Take a slice, for example S-S, downwards representing the deflection of teeth) by the amount of teeth deflection, $\frac{F_b}{k_t}$

f) Check if this new point is in the double pair region

g) If answer is NO as in the case of slice S-S, mark the position of the new point b1 which is the actual position of the output gear

h) If answer is YES then use the load constraint $F_b = \sum_{i=1}^{2} F_i$ to find the actual position in the double pair region.

For instance, in case of slice D-D the actual position (point b2) is founded by equations:

$$F_b = k_t(o_2 - b_2) + k_t(c_2 - b_2) \quad (4\text{-}3)$$

i) Repeat the procedure 5-9 for the required number of contact positions (slices)

j) Connect the new marked points (b1,b2,etc.) to construct the loaded TE curve There are two cases of profile relief such as long and short relief.They define the useful limits beyond which is not profitable to go in terms of corner contact, tooth loading and TE. The amount of relief at teeth tip is same for short and long relief cases and it is equal to sum of the tooth pair deflection. There is difference for extent of relief values for both cases.[22]

4.2 HELICAL GEAR TE THEORY AND QUASISTATIC TE CURVES

Spur gear is special type of helical gear with zero helix angles. Spur gears are regarded as two dimensional objects, while helical gears are three dimensional objects.

Helical gears are analogous to a set of stepped gears which consist of a number of identical spur gears so arranged that the teeth of each individual member are slightly out of phase relative to each other. In such an arrangement, there is an overlap during successive engagement of teeth, that is, when two teeth are in mesh at the pitch line, other mating pairs of teeth are in different phases of contact including approach and recess contacts. A helical gear construction is approximated if a composite body is made up of an infinite number of such stepped gears, each of which is a lamination of infinitesimal thickness, placed side by side successively with a slight phase difference. This has been illustrated in Figure 4.6.[22]

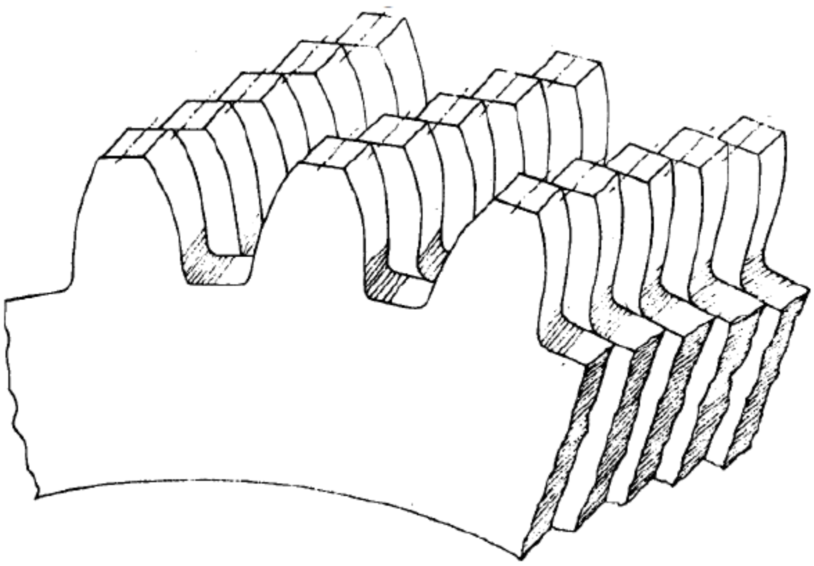


Figure 4.6 Formation of helical gears

This approximation is called as thin slice theory which allows accepting helical gears as stepped spur gears based on helix angle and an amount of slice shift. In this case each slice of helical gears can be thought as spur gear and modeled in 2D. Thin slice theory helps to simplify helical gear analysis which is more difficult than spur gear analysis due to helical gear's complexity. Helical gears can be considered as a combination of thin spur gear slices and these slices are brought together side by side with a slice shift (ss) based on helix angle (β), face width (b), and slice number (n). Slice shift can be defined difference between slices (Figure 4.7). It can be calculated as in following equation.

$$ss = \frac{b}{n}\tan\beta \quad (4\text{-}4)$$

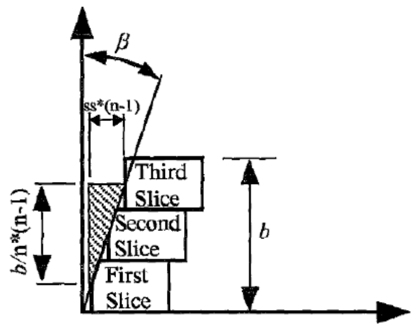


Figure 4.7 Top view of slice

The procedure of constructing of helical gear transmission error curve is based on same principle with spur gears. TE curve of helical gear for no relief and relief situations is illustrated in Figures 4.8 – 4.9 respectively.

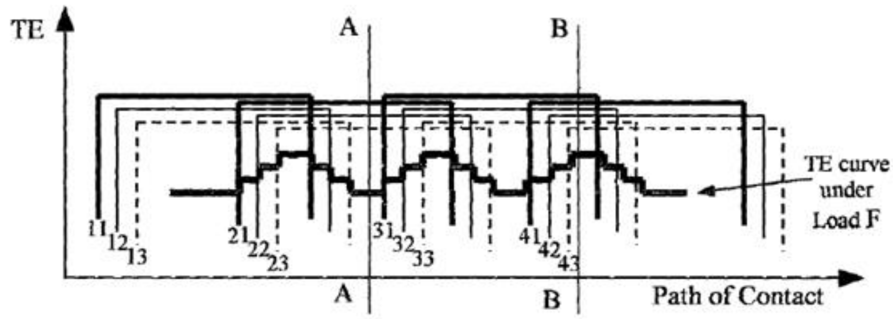


Figure 4.9 Formation of helical gear TE curve(no relief)

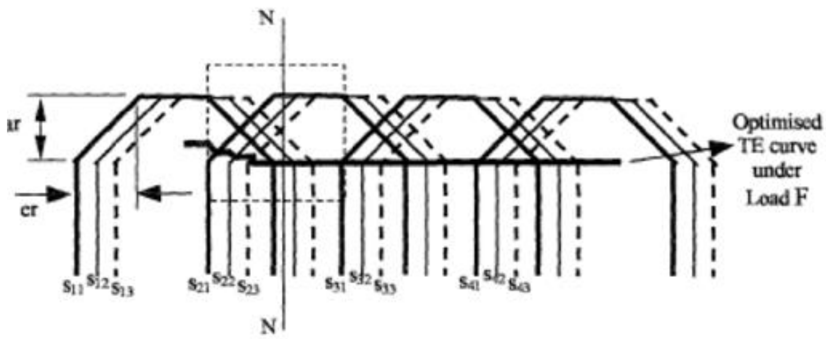


Figure 4.8 Formation of helical gear TE curve(with relief)

The procedure of constructing transmission error curves for no relief and relief situations [22] are given below:

a) Construct the boundary profiles of the each slice and place them with the distance (ss) in between, which is determined from equation 4.3.

b) Displace curves by the transverse base pitch of the gears along the horizontal axis.

c) No load TE curve is the upper most borders of each slice of Figure 4.9.

d) Take a section, for example N-N, to represent an instant of contact.

e) Determine which slices may be in contact, for section N-N.

f) Determine amount of relief values of all slices for any instant.

g) Put the relief values in ascending order ($r_{21}=0<r_{22}<r_{23}<r_{13}<r_{12}$)

h) Consider that all of the load (F) is carried by the slice with minimum relief value, determine deflection value of slice, $d_{21} = \frac{F}{(k*\frac{b}{n})}$

Where d_{21} is deflection of first slice of second tooth, k is stiffness value for the slice, b is face width and n is number of slices.

i) If $d_{21} > r_{22}$ then consider the slice 22 is in contact too

j) Write the load constraints to recalculate the deflection value

$$F = \left(k * \frac{b}{n}\right)[(r_{22} - d_{21+22}) + (r_{21} - d_{21+22})] \qquad (4\text{-}5)$$

k) Repeat the procedure 9-10 for the required number of slices for instant N-N

l) Connect the marked points to obtain the Te curve for the load [22]

5 TOOTH PROFILE DESIGN

This section is mainly based on related study of Kütük M.A.[22], Palmer, D. and Munro, R.G.[9]. Software's algortihm is writed with these mentioned studies . This chapter is the last chapter for giving the theoratical information about bacground . In all the chapter after here, I will explain software as section by section with details. I mean the software's theoretical background is very powerful ,it was a challange for gathering all them in together.

5.1 INTRODUCTION

The key point of the study is the performance of both helical and spur gears, under the effect of both geometric tooth errors and assembly errors. Since the gear geometry is based on sound mathematical relations it should also theoretically be possible to design both nominal sizes and the detailed tooth profile relief of the helical gears by using some derived analytical formulas. The same analytical formulas should also be employed in manufacturing of the gears since the gear teeth are cut based on the mathematical relation of involute curve.However, derivation of such analytical formulas reprea-senting the variation of tooth surfaces in 3D and also of other parameters such mesh stiffness would involve rather complicated mathematical relations. This requires a potential knowledge of mathematics and its manipulation. Because of difficulties, which could have been faced in further steps, the analytical methos of dealing with the helical gear profile design is left aside and an easier but sligthly less accurate way of approximate method is adopted.

Approximate methods could also be named as the numerical methods. They would include the method of finite element and others, such as dividing the helical gear to thin slicesalong the axis so that the helical gear is made up by the thin spur gears each of which is slightly offset angularly in accordance with the helix angle. Either of the two methods would approximate to the actual case of helical gear behaviou but with different accuraciess in results.

Increasing #of elements in FEM and incresing # of slices in thin slice theory will increse the accuracy of results with all other gear related parameters are correctly involved in the procedure. The FEM method , however, requires either a commercial package program or a home written FE program that requires an extensive FE knowledge.

Thin slice theory, on the other hand, requires the knowledge about TE theory of spur gears since the helical gears are represented as combination of a set of spur gears.

To construct a reliable analysis and design program, which can handle or simulate real operating conditions, parameters affecting the operating conditions of gears must be considered and included in the study. The most important ones of these parameeters:

- Existence of relief
- Shape of relief
- Crowning
- Misalingment
- Pitch error
- Pressure angle error
- Tooth pair stiffness [22]

5.2 PRESSURE ANGLE ERROR

Deviation of pressure angle of a rack from the ideal value causes profile deviation of the gear produced. For plus value of deeviation on the rack, a negative material occurance is observed on the tooth profile above the pitch point down to root of the gear. This profile deviation reverses for a negative value of deviation.

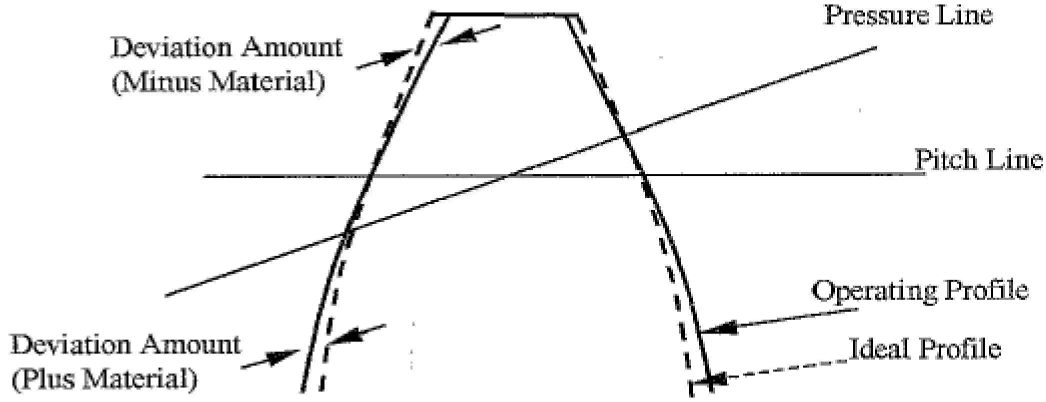


Figure 5.1 Effect of positive pressure angle error on involute profile

The value of deviation from the ideal tooth profile is determined as:

- Determine the coordinates of the ideal profile using the standart involute equations.
- Determine the actual coordinates of the operating profile fort he existence of PEA.
- Find the amount of deviation of the operating profile from the ideal value in the direciton of path of contact

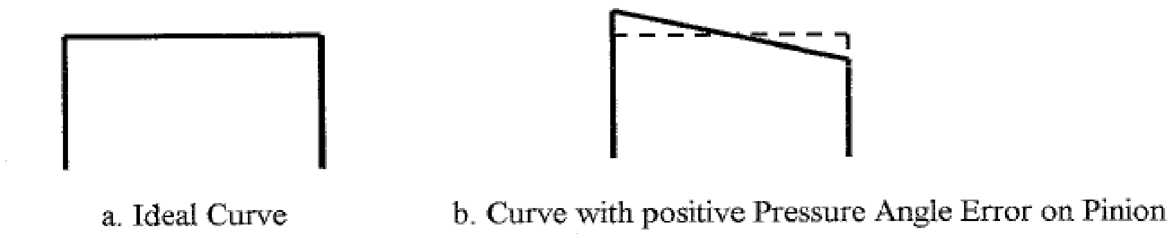


Figure 5.2 Effect of PEA on the isolated TE curve

- For the helical gears, the change in the isolated TE curves due to PEA affects all the slices of the teeth pair.[22]

5.2.1 ***Start Of Active Profile And TIF***

SAP is defined as the lowest point on the gear tooth where contact with the mating gear tooth tip can occur.On gears without tooth tip chamfers, it will be determined by the maximum outside diameter of the mating gear. It is usually expressed in degrees of roll above the base diameter.There is a diameter called the form diameter (Df), associated with this roll angle, which can be calculated.

True Involute Form diameter (TIF) is defined as being the point on the gear tooth where the involute form must begin.It is not necessarily the same diameter as the form diameter determined by the SAP. For example,it is possible for a SAP to occur in an undercut area , but the TIF diameter to be at a higher point on the tooth profile.

SAP is dependent on the mating gear outside diameter and the operating center distance.[23]

Terms and Definitions

N	Number of teeth in gear (specify as minus if an internal gear)
N_m	Number of teeth in mate (specify as minus if an internal gear)
P_n	Normal diametral pitch
D_o	Outside diameter of gear (inside dia. if gear is internal)
D_{om}	Outside diameter of mate (inside dia. if mate is internal)
D_b	Base circle diameter of gear
D_{bm}	Base circle diameter of mate
D_f	Form diameter
OD/ID	Outside diameter/Inside diameter
SAP	Start of active profile
TIF	True involute form
m_c	Profile contact ratio
ϕ_n	Normal pressure angle
ϕ_t	Transverse pressure angle
ϕ'_t	Operating transverse pressure angle
ψ	Helix angle (zero if spur gear)
θ_o	Roll angle at gear OD/ID
θ_{om}	Roll angle at mating gear OD/ID

Figure 5.3 .Terms and definitions about SAP & TIF

Calculating Transverse Pressure Angle

$$\phi_t = \operatorname{atan}\left(\frac{\tan(\phi_n)}{\cos(\psi)}\right) \tag{1}$$

Calculating Operating Transverse Pressure Angle

$$\phi_t' = \operatorname{acos}\left[\frac{(N + N_m) \cdot \cos(\phi_t)}{2 \cdot P_n \cdot CD \cdot \cos(\psi)}\right] \tag{2}$$

Calculating Base Circle Diameter of Gear

$$D_b = \left[\frac{N \cdot \cos(\phi_t)}{P_n \cdot \cos(\psi)}\right] \tag{3}$$

Calculating Base Circle Diameter of Mating Gear

$$D_{bm} = \left[\frac{N_m \cdot \cos(\phi_t)}{P_n \cdot \cos(\psi)}\right] \tag{4}$$

Calculating Pressure Angle at OD/ID of Gear if Mating Gear OD/ID is Given

$$\phi_{om} = \operatorname{acos}\left(\frac{D_{bm}}{D_{om}}\right) \tag{5}$$

Calculating SAP if Form Diameter is Given

$$SAP = 180 \cdot \frac{\sqrt{\frac{D_f^2}{D_b^2} - 1}}{\pi} \tag{6}$$

Calculating Pressure Angle at OD/ID of Mating Gear if SAP is Given

$$\phi_{om} = \operatorname{atan}\left[\frac{(N + N_m) \cdot \tan(\phi_t') - \pi \cdot N \cdot \frac{SAP}{180}}{N_m}\right] \tag{7}$$

Calculating Start of Active Profile

$$SAP = 180 \cdot \frac{(N + N_m) \cdot \tan(\phi_t') - N_m \cdot \tan(\phi_{om})}{\pi \cdot N} \tag{8}$$

Calculating OD/ID of Mating Gear

$$D_{om} = \frac{D_{bm}}{\cos(\phi_{om})} \tag{9}$$

Calculating Pressure Angle at SAP of Gear

$$\phi_f = \operatorname{atan}\left(\frac{\pi \cdot SAP}{180}\right) \tag{10}$$

Calculating Form Diameter at SAP of Gear

$$D_f = \frac{D_b}{\cos(\phi_f)} \tag{11}$$

Calculating Pressure Angle at OD/ID of Gear

$$\phi_o = \operatorname{acos}\left(\frac{D_b}{D_o}\right) \tag{12}$$

Calculating Roll Angle at OD/ID of Gear

$$\theta_o = 180 \cdot \frac{\tan(\phi_o)}{\pi} \tag{13}$$

Calculating Profile Contact Ratio

$$m_c = \frac{|\theta_o - SAP| \cdot |N|}{360} \tag{14}$$

* **Note: quantities enclosed in vertical bars are absolute values.**

Figure 5.4 Equaitons of Figure 5.3

5.3 PITCH ERROR

Pitch error is defined as the unequal spacing between the neighbouring teeth. This causes both corner contact and an increase in TE value during operation of gears.

Existance of an adjacent PE between the teeth of gears causes a positive TE when plus material condition happens and a negative TE when material removal happens on the teeth. This is simulated on TE diagram by shifting the TE boundary curve of the tooth pair upwards or downward as much the value of PE(in microns) depending on whether positive or negative cases occur. [22]

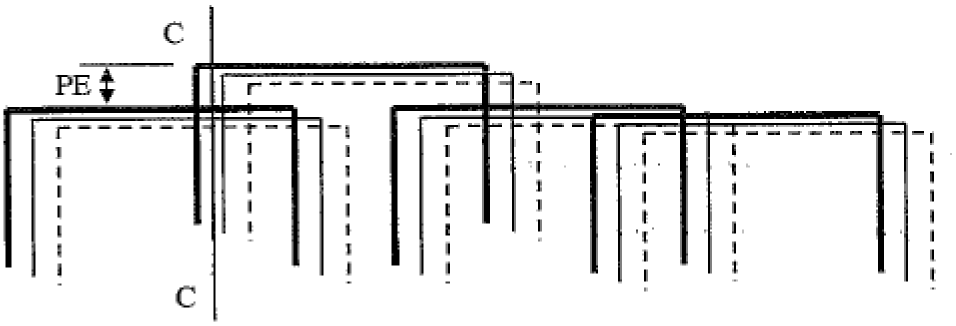


Figure 5.3 Effect of PE on the isolated TE boundaries of tooth pairs

5.4 MISALINGMENT

Parallel axis gears, even if designed and manufactured well, are liable to be misaligned due to small manufacturing errors.Misalignment of gears also occurs due to dimensional variations in machining housing or due to the errors in mounting gears on shafts. This misalignment gives rise to TE hence vibration and noise.[22]

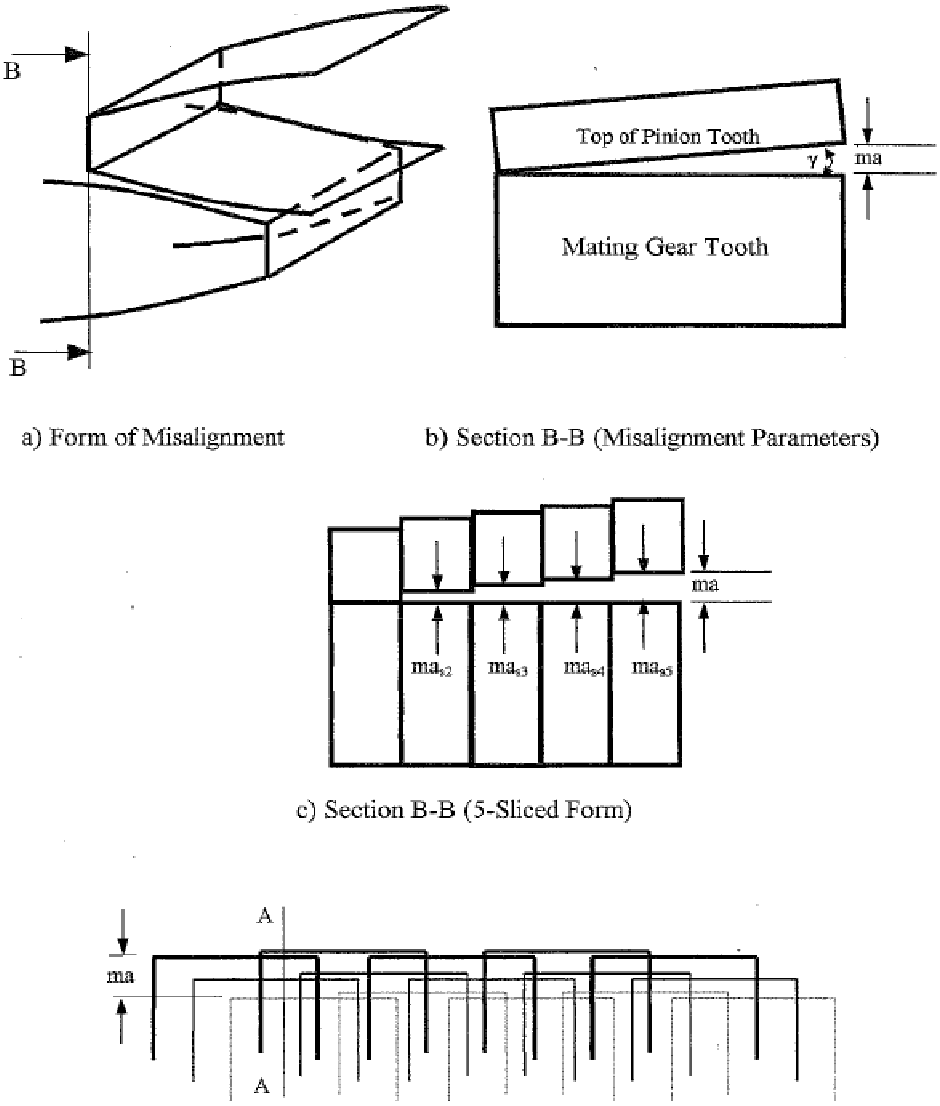


Figure 5.4 Misaligned gear tooth pair and construcion of the isolated TE boundaries.

5.5 CROWNING

Crowning process of gears involves taking a slight deeper ut at the face and/or end of tooth than at the midflank. This has the effect of providing end relief to the teeth and thereby reducing the suddenness of load application as each tooth comes in mesh. Crowning reduces the overstressing effect in misalignment. Calculation procedure for crowning is the same as the misalignment , that is crowning value for each slice is converted to a PE value on this slice.[22]

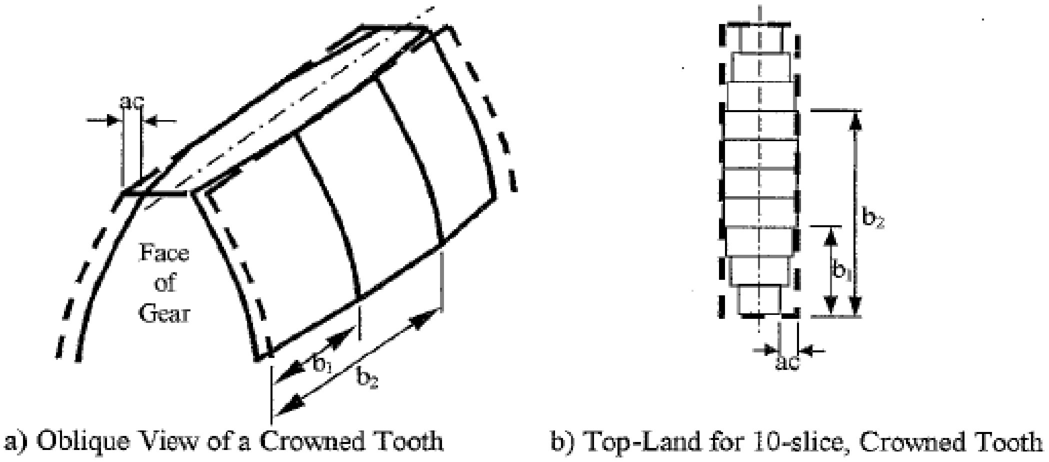


Figure 5.5 Crowning parameters

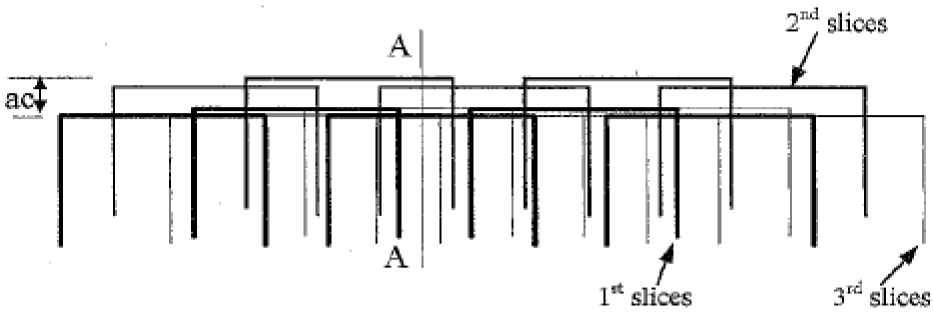


Figure 5.6 Effect of crowning on the isolated TE boundaries

5.6 RELIEF AND SHAPE OF RELIEF

Profile relief may be characterized by two parameters:

- Amount of relief(ar)
- Extent of relief(er)

The amount of relief is usually estimated as the deflection of the gear mesh at the maximum load for spur gears. For helical gears, it is not easy to obtain a genaral rule for the amount of relief, as the number of tooth pairs in contact cannot be determined so easily for an instant.[22]

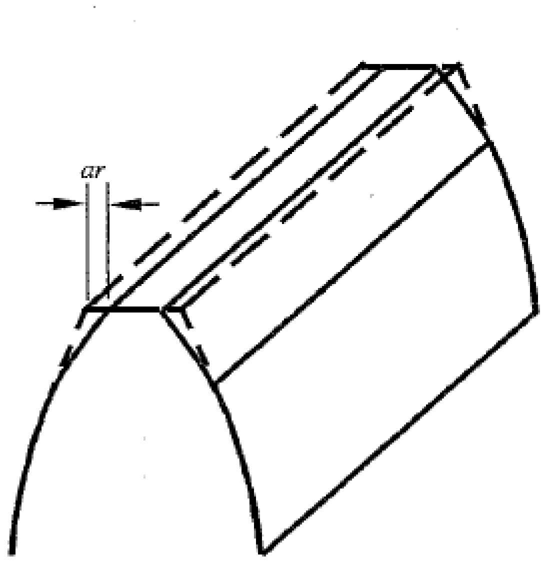


Figure 5.7 Simple tip relief

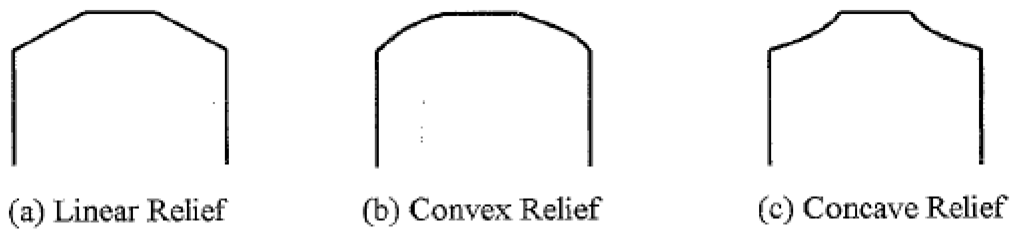


Figure 5.8 Different relief shapes

6 EXPLANATION OF SOFTWARE

6.1 INTRODUCTION

Gear design and optimazing the tooth profile difficult and long-lasting calculation process due to complexity of gear and it involute profile. Generally,most of the results of calculations is not satifying at the beginning of the desingning. Then calculations repated again and again to reach the desired results. Changing or cheching the paramaters is also slow and prone to erros.Because of this causes, this software is aimed to eliminate this factors and getting better results with saving time.In addition to all this this software's main objects are:

- Reach the minimum TEpp at the design load(Fd)
- Better tooth load diagram

This software was developed by using Dev-C++ is a full-featured integrated development environment(ide) for the C/C++ programming language.Then the results of this algorithms are plotted on MATlab.

6.2 DEV-C++

Dev-C++ is a free full-featured integrated development environment (IDE) distributed under the GNU General Public License for programming in C and C++. It is written in Delphi.

It is bundled with, and uses, the MinGW or TDM-GCC 64bit port of the GCC as its compiler. Dev-C++ can also be used in combination with Cygwin or any other GCC-based compiler.

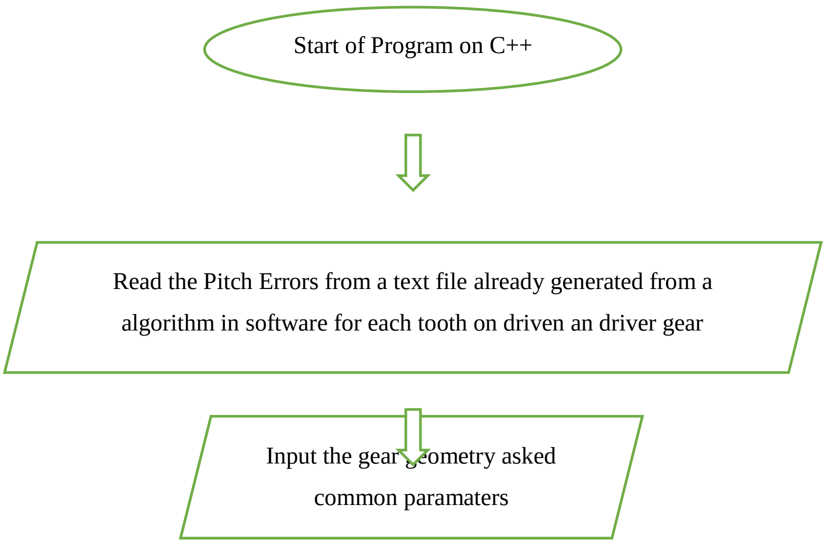
Dev-C++ is generally considered a Windows-only program, but there are attempts to create a Linux version: header files and path delimiters are switchable between platforms.

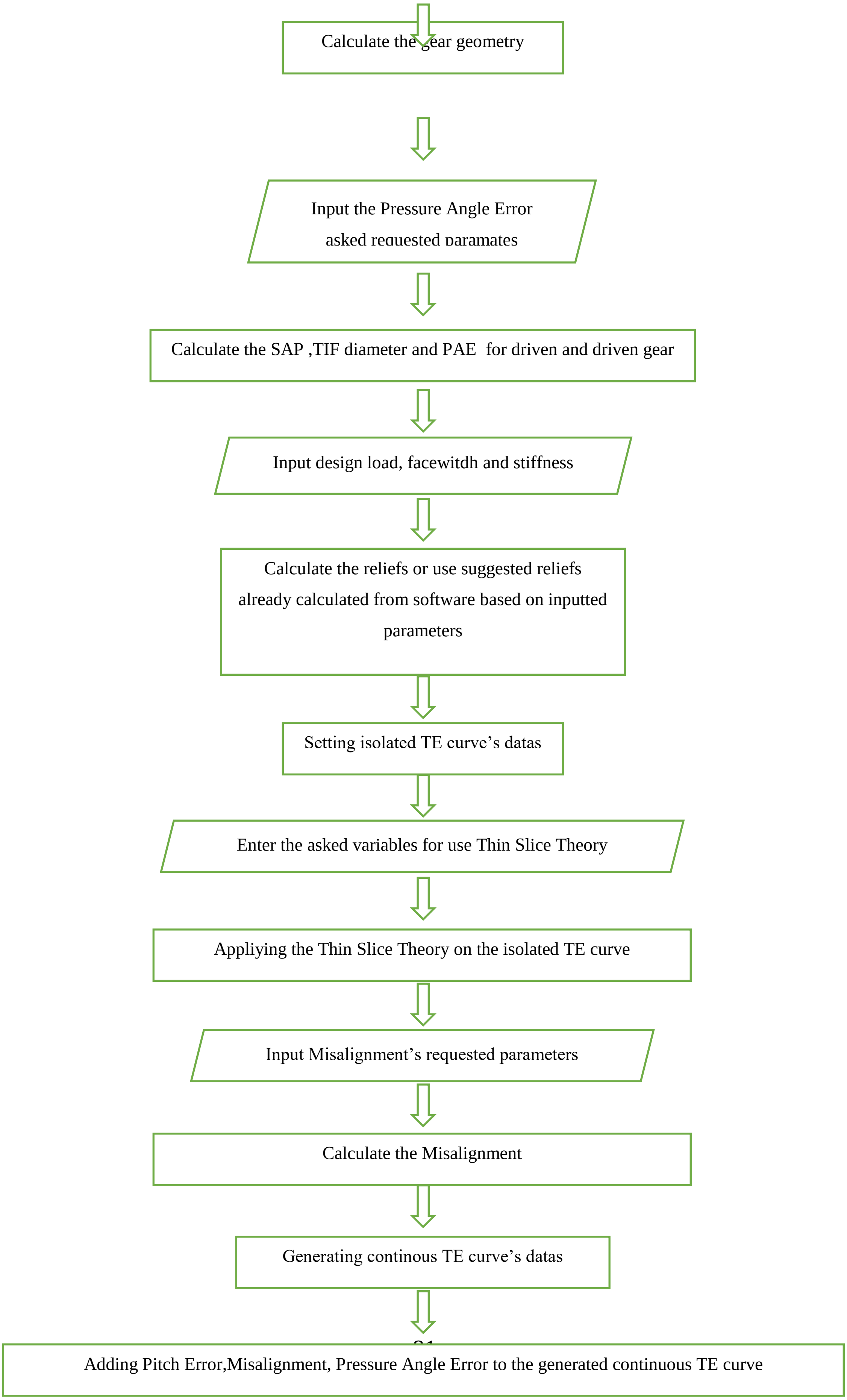
6.3 MATLAB

MATLAB (*matrix laboratory*) is a multi-paradigm numerical computing environment and proprietary programming language developed by MathWorks. MATLAB allows matrix manipulations, plotting of functions and data, implementation of algorithms, creation of user interfaces, and interfacing with programs written in other languages, including C, C++, C#, Java, Fortran and Python. Although MATLAB is intended primarily for numerical computing, an optional toolbox uses the MuPAD symbolic engine, allowing access to symbolic computing abilities. An additional package, Simulink, adds graphical multi-domain simulation and model-based design for dynamic and embedded systems.

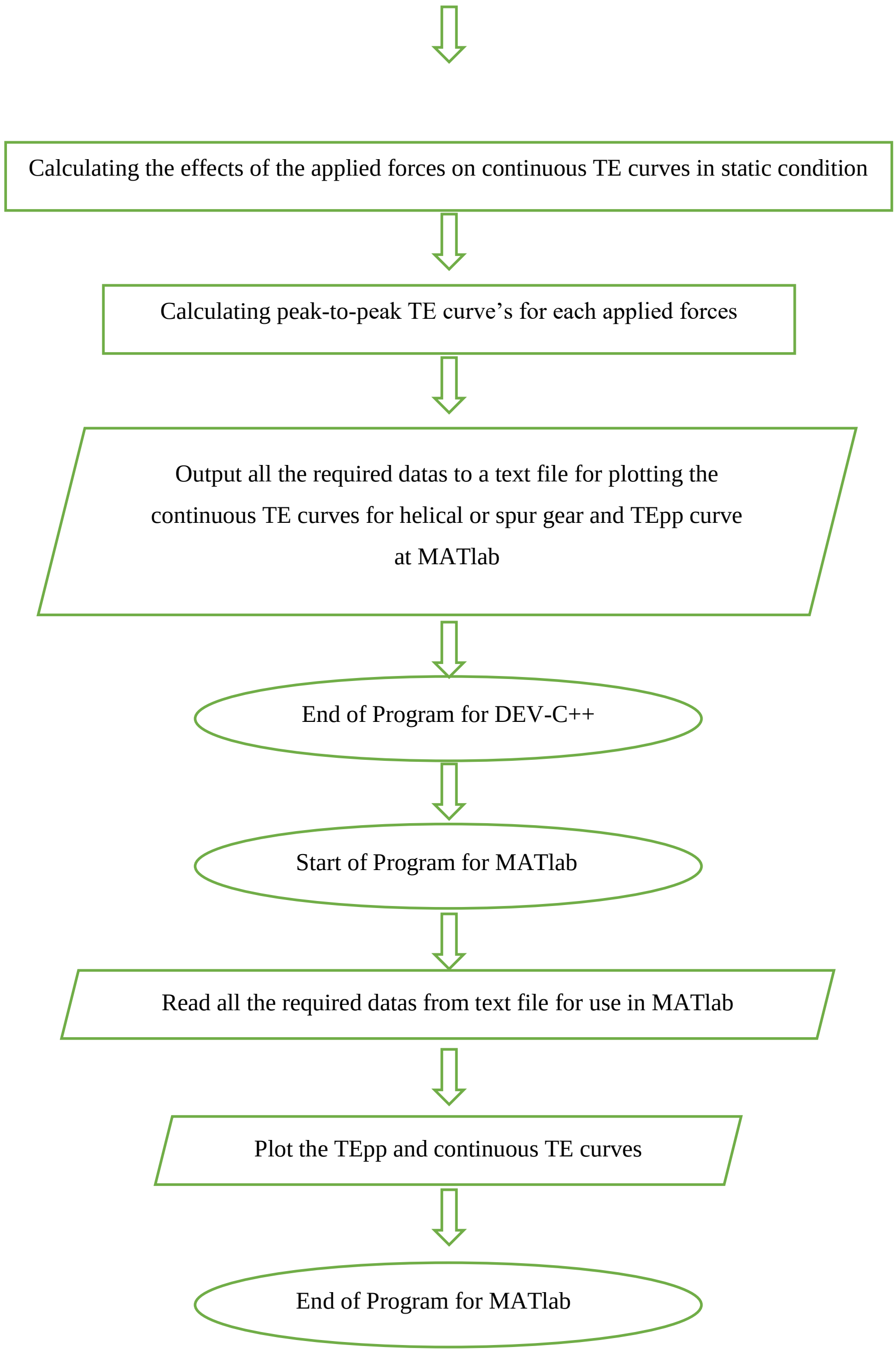
As of 2018, MATLAB has more than 3 million users worldwide. MATLAB users come from various backgrounds of engineering, science, and economics.

6.4 FLOWCHART OF THE SOFTWARE





Calculate the gear geometry
Input the Pressure Angle Error
asked requested paramates
Calculate the SAP ,TIF diameter and PAE for driven and driven gear
Input design load, facewitdh and stiffness
Calculate the reliefs or use suggested reliefs
already calculated from software based on inputted
parameters
Setting isolated TE curve’s datas
Enter the asked variables for use Thin Slice Theory
Appliying the Thin Slice Theory on the isolated TE curve
Input Misalignment’s requested parameters
Calculate the Misalignment
Generating continous TE curve’s datas
Adding Pitch Error,Misalignment, Pressure Angle Error to the generated continuous TE curve


Calculating the effects of the applied forces on continuous TE curves in static condition
Calculating peak-to-peak TE curve's for each applied forces
Output all the required datas to a text file for plotting the continuous TE curves for helical or spur gear and TEpp curve at MATlab
End of Program for DEV-C++
Start of Program for MATlab
Read all the required datas from text file for use in MATlab
Plot the TEpp and continuous TE curves
End of Program for MATlab

6.5 C++ ALGORITHMS AND EXPLANATIONS

/*DONE BY ONUR ELDES (JANUARY 2019)

6.1.1 LAYOUT OF THE SOFTWARE:

*THE MAIN PURPOSES OF THIS PROGRAM ARE REACHING THE BETTER TRANSMISSION ERROR CURVE UNDER DESIGN LOAD FOR STATIC CONTION AND GETTING MINIMUM PEAK-TO-PEAK TRANSMISSION ERROR CURVE, SO ELIMINATE THE MAIN CAUSES OF VIBRATION AND NOISE FOR LOW CONTACT HELICAL OR SPUR GEAR.THIS PORGRAM BASED ON FINITE ELEMENT ANALYSIS'S WORKING THEORY.

*RANDOM PITCH ERROR VALUES FOR EACH TOOTH OF PNION AND GEAR ARE CREATED AND WRITED AT A OUTPUT FILE FOR ADD TO CALCULATIONS WITH THE START OF PROGRAM

*COMMON GEOMETRICAL PARAMATERS OF SPUR AND HELICAL GEAR ARE CALCULATED AT THE BEGINNIG SECTIONS OF THE SOFTWARE WITH SOME ASKED INPUTS TO THE USER.

*THEN,PRESSURE ANGLE ERRORS ARE CALCULATED BASED ON USER INPUTS, MOST CORRECT TIF AND SAP VALUES ARE FOUND AT THIS CALCULATIONS.

*BEST AMOUNT OF RELIEFS AND EXTENT OF RELIEFS ARE CALCULATED TO GET BETTER TE CURVE AND SUGGESTED TO THE USER. USER CAN BE INPUT DESIRED AR AND ER FOR RIGHTSIDE AND LEFTSIDE OF TOOTH PROFILE, IF HE/SHE REJECTS USING THE RECOMMENDED ONES.

*THEN, ACCORDING THE REACHED INFOS UNDEFORMED ISOLATED TE CURVE IS GENERATED.

*AFTER PREVIOUS STAGE, THIN SLICE THEORY CALCULATIONS ARE STARTING. IF USER WANTS TO DESIGN FOR SPUR GEAR ,USER MUST INPUT THIN SLICE NUMBER AS 1 SLICE.

*MISALIGNMENT AND CROWNING ARE CALCULATED ACCORDING TO USER'S INPUT FOLLOWING SECTIONS.

*NOW PROGRAM CREATING THE CONTINOUS TE CURVE TO LOADING AFTER ALL THE PPREVIOUS SECTIONS.

*LOADED CONTINOUS TE CURVE'S DATAS ARE CALCULATED AFTER USER ENTERED THE LOAD INCREMENTS.

*AT THE LAST CALCULATIONS SECTION,THE SOFTWARE IS PREPARING THE PEAK-TO-PEAK TE CURVE DATAS FOR EACH FORCE INCREMENT.

*ALL THE NEEDED DATAS ARE WRITED A EXTERNAL FILE TO PLOT IN MATLAB AT THE END OF PROGRAM.

*/

```
/*AT THIS PROGRAM USER CAN DECIDE THE MESH NUMBER ALONG THE INVOLUTE PROFILE,THIN SLICE NUMBER ALONG FACEWIDTH AND FORCE INCREMENTS*/

#include <iostream>

#include <cmath>

#include <fstream>

#include <iomanip>

using namespace std;

//HEAVY MATRICES USED AT THE FOLLOWING STAGES.
```

```
double RFminus[5000];

double RFplus[5000];

double RFx2[5000];

double RFy2[5000];

double nearRF[5000];

double RFy2minus[5000];

double RFy2plus[5000];

double LLi[1000];

double LRi[1000];

double L[1000];

double LTS[1000][1000];

double sTS[1000][1000];

double PAE[1000];

double box[5000][5000];

double boxleak[5000][5000];

double boxleakf[5000][5000];

double boxf[5000][5000];

double tboxf[5000][5000];

double Q[5000];

double K[5000][5000];

double def[5000][5000];

double TEmax_min[1000];


int main()

{

/*
```

6.1.2 LIST OF CONTENT OF SOFTWARE:

0.STARTING OF PROGRAM

1.DESCRIBING OUTPUT FILES

2.PITCH ERROR

3.GEOMETRY OF GEAR

4.PRESSURE ANGLE ERROR

4.1 FINDING SAP and form diameter at SAP

4.2 EVALUATING INVOLUTE PROFILE

4.3 FINDING PAE

5.ADDING RELIEFS

5.1. AMOUNT OF RELIEF

5.2. EXTEND OF RELIEF

6.SETTING ISOLATED TE CURVE

7.THIN SLICE THEORY

7.1 MISALIGNMENT

7.2 CROWNING

7.3 GENERATING STS

7.4 ADDITION OF PITCH ERROR,MISALIGNMENT,CROWNING TO sTS

7.5 FINDING INFINITE

7.6 boxleak[][]

7.7 PREPARING boxf[][] PROPER TO DEFORMATION CALCULATIONS

8.CALCULATIONS OF DEFORMATIONS

8.1 STIFFNESS VALUES

8.2 DEFORMATION

9.TEmax_min

10.OUTPUT FILES FOR MATLAB

11.END OF PROGRAM

*/

/////

6.1.3 ***0.Starting Of Program***

/////

top:;//IF CR>=2,USER WILL REDIRECTED TO HERE, FOR INPUTTING REQUESTED PARAMATERS AGAIN.THIS PORGRAM IS DESIGNED JUST FOR LCR

/////

6.1.4 ***1.Describing Output Files***

/////

//AT THIS SECTION,ALL THE EXTERNAL OUTPUT FILES ARE DEFINED.NAMES OF THESE FILE ARE DESCRIBED BY THEM GENERATING DATES.

```
    ofstream allin("ste_out_03_09_18.txt");  //GENERAL OUTPUT FILE.YOU CAN FIND ALL THE OUTPUT DATAS IN THIS FILE

    ofstream nomen("nomenclature_03_09_18.txt");

    ofstream Li("a1_1_Li_out_03_09_18.txt");

    ofstream LL("a1_2_LLi_out_03_09_18.txt");

    ofstream LR("a1_3_LRi_out_03_09_18.txt");

    ofstream BOX("a2box_out_03_09_18.txt");

    ofstream DEF("a3def_out_03_09_18.txt");

    ofstream TE("a4TEmax_min_out_03_09_18.txt");


//THIS OUTPUTS ARE USED FOR CONTROLS TO SEE THERE IS FAILURE OR NOT.IF YOU SEE THIS WORDS BEGINNING AND END OF PROGRAM, IT MEANS THERE IS NO FAILURE

            allin<<"ONURELDES"<<endl;

            cout<<"ONURELDES"<<endl;

            cout<<endl;



///////////////
```

6.1.5 2.Pitch Error

```
///////////////

//PITCH ERROR ARE CREATED RANDOMLY.IF YOU DESIRED ,YOU CAN READ THE SPECIFIC PE FROM A EXTERNAL FILE.

//THERE ARE PITCH ERRORS FOR EACH TOOTH OF DRIVEN AND DRIVER GEAR.AT THIS SECTION WE GETTING TOTAL PE FOR EACH TOOTH MATE BY ADDING

//THEM TO EACHOTHER.SO PE[TG] MATRIX IS DEFINED WHICH IS INCLUDING ALL THE PE DATAS FOR ONE REVOLUTION OF GEAR.


//THIS SECTION IS CREATED DUE TO THERE IS NO PE DATAS ALREADY PREPARED FOR READING.SO ,THERE IS A NECESSARY TO

//CREATE THEM MANUALLY AND THEN READ THEM. IF THERE IS A PE DATAS FILE EXIST ,YOU CAN CANCEL THIS SECTIONS INITIAL PARTS.


/*

Pitch error is defined as the unequal spacing between the neighbouring teeth.

This causes both corner contack and an increase in TE value during operation of gears.

Existence of an adjacent PE between teeth of gears causes a positive TE when plus material condition

happens and a negative TE when material removal happens on the teeth.
```

This is simulated on TE diagram by shifting the TE boundary curve of tooth pair upwards or downwards as much the value of PE(in microns) depending on whether positive or negative material cases occur.

```
*/

//TP=tooth number of pinion(SMALLER)
//TG=tooth number of gear(BIGGER)
   ofstream peTsmaller("a2peTsmaller_out_03_09_18.txt");//pressure error for smaller teeth gear
   ofstream peTbigger("a2peTbigger_out_03_09_18.txt");//pressure error for bigger teeth gear
   ofstream pe("a2pe_out_03_09_18.txt");//pressure error for sum of bigger and smaller gear

          int Tdriven,Tdriver;//T means tooth number
          cout<<"enter tooth numbers of driver and driven gears respectively:"<<endl;
          cin>>Tdriver>>Tdriven;

   int Tbigger;//DECIDED BY PROGRAM BY PICKING UP BETWEEN DRIVEN AND DRIVER
   int Tsmaller;

if(Tdriven>=Tdriver)
{Tsmaller=Tdriver;Tbigger=Tdriven;}
if(Tdriven<Tdriver)
{Tsmaller=Tdriven;Tbigger=Tdriver;}

//cout<<"Tsmaller="<<Tsmaller<<endl;
//cout<<"Tbigger="<<Tbigger<<endl;

//MANUEL PE DATAS ARE CREATED FOR PINION AND WRITING TO OUTPUT FILE
          for(int i=0;i<Tsmaller;i++){
             if(i%2==0){peTsmaller<<0<<endl;}
             else peTsmaller<<0<<endl;}

//MANUEL PE DATAS ARE CREATED FOR GEAR AND WRITING TO OUTPUT FILE
          for(int i=0;i<Tbigger;i++){
            if(i%2==0){peTbigger<<0<<endl;}
```

```
        else peTbigger<<0<<endl;}



double smaller[Tsmaller];

double bigger[Tbigger];

double PE[Tbigger];//THE MATRIX INCULEDS TOTAL PE DATAS FOR EACH TOOTHMATE AT ONE
REVOLOTION

//READS THE PE FOR PINION

ifstream pes("a2peTsmaller_out_03_09_18.txt");


if ( pes.is_open() ) {

for (int i=0; i<Tsmaller; i++) {pes>>smaller[i];}

pes.close();}

else cout << "Unable to open file!";


//for (int i=0; i<Tsmaller; i++) {cout << "smaller[" << i << "]="<< smaller[i] << endl;}


//READS THE PE FOR GEAR

ifstream peb("a2peTbigger_out_03_09_18.txt");


if ( peb.is_open() ) {

for (int i=0; i<Tbigger; i++) {peb>>bigger[i];}

peb.close();}

else cout << "Unable to open file!";


//for (int i=0; i<Tbigger; i++) {cout << "bigger[" << i << "]="<< bigger[i] << endl;}


//CREATING PE[TPE]


for(int i=0;i<Tbigger;i++){

      if(i<Tsmaller){PE[i]=bigger[i]+smaller[i];}

      if(i>=Tsmaller){PE[i]=bigger[i]+smaller[i-Tsmaller];}}


//for (int i=0; i<Tbigger; i++) {cout << "PE[" << i << "]="<< PE[i] << endl;}
```

```
double largest=0;;//LARGEST PE
double smallest=0;//SMALLEST PE
//FINDING LARGEST
  for(int i = 0;i <Tbigger; i++) {
    if(largest <= PE[i])
      largest = PE[i];
}
cout<<"largest PE = "<<largest<<endl;
//FINDING SMALLEST
  for(int i = 0;i <Tbigger; i++) {
    if(smallest >= PE[i])
      smallest= PE[i];
}
cout<<"smallest PE = "<<smallest<<endl;
cout<<endl;


//////////////////////////////
```

6.1.6 3.Geometry Of Gear

```
//////////////////////////////
        double m,mt,mn,DP,aopn,aopt,helix,pt,px,pn,Y,X,bb,add,ded;
        double pc,pb,Lc,pcr,acr,rtdriven,rtdriver,rbdriven,rbdriver,rpdriven,rpdriver,CD;
        /*
        2.nomenclature
        m=module;
        mn=normal module;!! mn=m;
        mt=tranverse module;
        aopn=normal angle of pessure,
        aopt=transverse angle of pressure;
        helix=helix angle;
        pt=transverse pitch;
        px=axial pitch;
        pn=normal pitch;
        Y=multiplier for addendum;
```

```
		X=multiplier for dedendum;
		add=addendum;
		ded=dedendum;
		bb=facewidth;
		pc=circular pitch;
		pb=base pitch;
		Lc=contactlength;
		pcr=profile contact ratio;
		acr=axial contact ratio;//totalCR=pcr+acr;
		rt=tip_radii;
		rb=base_radii;
		rp=pitch_radii
		*/
cout<<"enter normal module"<<endl;
cin>>m;
allin<<"m=   "<<m<<endl;
cout<<"enter angle of pressure in normal plane(in degrees)"<<endl;
cin>>aopn;
allin<<"aopn= "<<aopn<<endl;
cout<<"input helix angle(in degrees)"<<endl;
cin>>helix;
cout<<endl;
allin<<"helix="<<helix<<endl;

//calculation of aopt from aopn
		aopt=atan(tan(aopn*M_PI/180.0)/cos(helix*M_PI/180.0));
//		cout<<"angle of pressure in transverse plane(in degrees)= "<<aopt*180.0/M_PI<<endl;

//calculation of mn,mt,pt,pn,px
		mn=m;
		mt=mn/cos(helix*M_PI/180.0);
		pt=M_PI*mt;
		pn=M_PI*mn;
		px=pt/tan(helix*M_PI/180.0);
```

```
        cout<<"transverse module= "<<mt<<endl;
        cout<<"normal module= "<<mn<<endl;
        cout<<"transverse pitch= "<<pt<<endl;
        cout<<"normal pitch= "<<pn<<endl;
        cout<<"axial pitch= "<<px<<endl;

        allin<<"transverse module= "<<mt<<endl;
        allin<<"normal module= "<<mn<<endl;
        allin<<"transverse pitch= "<<pt<<endl;
        allin<<"normal pitch= "<<pn<<endl;
        allin<<"axial pitch= "<<px<<endl;

//cout<<"cos(angle_of_pressure transverse )="<<cos(aopt)<<endl;
  cout<<endl;
cout<<"enter multiplier Y of addendum as Y*module=addendum"<<endl;
cin>>Y;
  cout<<"enter multiplier X of dedendum as X*module=dedendum"<<endl;
        cin>>X;
  cout<<"enter facewidth(mm)"<<endl;
        cin>>bb;
  cout<<endl;

        add=m*Y;
        ded=m*X;
        pc=M_PI*mt;
        pb=pc*cos(aopt);
        rpdriven=mt*Tdriven/2;
        rpdriver=mt*Tdriver/2;
        rbdriven=rpdriven*cos(aopt);
        rbdriver=rpdriver*cos(aopt);
        rtdriven=rpdriven+add;
        rtdriver=rpdriver+add;
        CD=rpdriven+rpdriver;
```

```
DP=25.4/mn;

cout<<fixed<<setprecision(6)<<"rpdriven="<<rpdriven<<endl;
cout<<fixed<<setprecision(6)<<"rpdriver="<<rpdriver<<endl;
cout<<"CD= "<<CD<<endl;
cout<<"diametral pitch= "<<DP<<endl;
cout<<"pc ="<<pc<<endl;
cout<<"pb ="<<pb<<endl;
cout<<fixed<<setprecision(6)<<"rbdriven="<<rbdriven<<endl;
cout<<fixed<<setprecision(6)<<"rbdriver="<<rbdriver<<endl;
cout<<fixed<<setprecision(6)<<"rtdriven="<<rtdriven<<endl;
cout<<fixed<<setprecision(6)<<"rtdriver="<<rtdriver<<endl;

allin<<fixed<<setprecision(6)<<"rpdriven="<<rpdriven<<endl;
allin<<fixed<<setprecision(6)<<"rpdriver="<<rpdriver<<endl;
allin<<"CD= "<<CD<<endl;
allin<<"diametral pitch= "<<DP<<endl;
allin<<"pc ="<<pc<<endl;
allin<<"pb ="<<pb<<endl;
allin<<fixed<<setprecision(6)<<"rbdriven="<<rbdriven<<endl;
allin<<fixed<<setprecision(6)<<"rbdriver="<<rbdriver<<endl;
allin<<fixed<<setprecision(6)<<"rtdriven="<<rtdriven<<endl;
allin<<fixed<<setprecision(6)<<"rtdriver="<<rtdriver<<endl;

//CALCULATION OF CR

Lc=pow((rtdriven*rtdriven-rbdriven*rbdriven),0.5)+pow((rtdriver*rtdriver-rbdriver*rbdriver),0.5)-
  (pow((rpdriven*rpdriven-rbdriven*rbdriven),0.5)+pow((rpdriver*rpdriver-rbdriver*rbdriver),0.5));
pcr=Lc/pb;
acr=bb/px;
cout<<"Lc ="<<Lc<<endl;
cout<<"pcr= "<<pcr<<endl;
cout<<"acr= "<<acr<<endl;
cout<<"totalCR= "<<pcr+acr<<endl;
```

```
allin<<"Lc=  "<<Lc<<endl;

allin<<"pcr=  "<<pcr<<endl;

allin<<"acr=  "<<acr<<endl;

allin<<"totalCR= "<<pcr+acr<<endl;


while(1) //LOOP FOR CONTROLLING LCR CASE

{if(pcr< 2){cout<<"pcr ="<<pcr<<endl;break;}//CR=contact_ratio

 if(pcr>=2){cout<<"pcr ="<<pcr<<"\nyour pcr>=2,these program is not for high contact ratios,\nit is for low contact ratios"<<endl;

        cout<<"enter new values!!"<<endl;

 goto top;}}//GO TO TOP IF CR>=2 AND INPUT AGAIN

cout<<endl;


/////////
```

6.1.7 ***4.Pressure Angle Error***

```
////////

double totalpae1;

double totalpae2;

while (1){

        char anspae;

        cout<<"is there any pressure angle error Y or N "<<endl;

        cin>>anspae;

        if(anspae=='y'||anspae=='Y'){

/*
```

Deviation of pressure angle of a rack from the ideal values causes profile deviation of the gear produced.For plus value of deviation on the rack, a negative material occurance is observed on the tooth profile above the pich point up to tip point;and positive material exists below the pitch point down to the root of gear.This profile deviation reverses for a negative value of deviation.The value of deviation from the ideal tooth profile is determined as:

-> Determine the coordinates of the ideal profile using standard involute equations

-> Determine the actual coordinates of the operating profile for the existence of pressure error

-> Find the amount of deviation of the operating profile from the ideal profile value in the direction of patch of contactFor the case of helical gears, the change in isolated TE curves due to pressure angle error affects all the slices of the teeth of pair.

*/

/////////

6.1.7.1 4.1 Finding Sap And Form Diameter At Sap

////////

/*

Start of active profile(SAP) is defined as the lowest point on the gear tooth where contact with the mating gear tooth tip

can occur.On gears without tooth tip chamfers, it will be determined by the max. outside diameter of the mating gear.

it is usually expressed in degrees of rool above base diameter.There is a diameter called the form diameter(Df), associated

with this roll angle which can be calculated.

True involute Form diameter(TIF) is defined as being the point on the gear tooth where the involute form must begin.it is not

necessarily the same diameter as the form diameter determined by the SAP.For example, it is possible for a SAP to occur in an

undercut area,but the TIF diameter to be at a higher point on the tooth profile.

SAP is dependent on the mating gear outside diameter and the operating center distance.

by DAN THURMAN

*/

/*

Terms and Definitions

Df= form diameter in inches

SAP= start of active profile

tpa= transverse pressure angle

otpa= operating transverse pressure angle

paom= pressure angle at OD/ID of gear if mating gear OD/ID is given

paSAP= pressure angle at SAP of gear

*/

//all calculations on this section based on inches

/*calculating Qt= transverse pressure angle*/

cout<<"FINDING RF AT SAP"<<endl;

double tpa;

tpa=atan(tan(aopn*M_PI/180.)/cos(helix*M_PI/180.));//cout<<"tpa= "<<tpa*180/M_PI<<endl;

```
/*calculating Qto= operating transverse pressure angle*/

double otpa;

otpa=acos(((Tdriver+Tdriven)*cos(tpa))/(2*DP*(CD/25.4)*cos(helix*M_PI/180.)));//cout<<"otpa=
"<<otpa*180/M_PI<<endl;


/*calculating pressure angle at OD/ID of gear if mating gear OD/ID is given*/

double paom;

paom=acos((rbdriven)/(rtdriven));//cout<<"paom= "<<paom*180/M_PI<<endl;

///////////////////////////////////////////////driver

/*calculating SAP*/

double SAPdriver;//in degrees

SAPdriver=180*((Tdriver+Tdriven)*tan(otpa)-Tdriven*tan(paom))/M_PI/Tdriver;

cout<<"SAPdriver= "<<SAPdriver<<endl;


/*calculating Qf*/

double paSAP;

paSAP=atan(M_PI*SAPdriver/180);//cout<<"paSAP= "<<paSAP*180/M_PI<<endl;


/*calculating form dia at SAP  of gear*/

double rfdriver;

rfdriver=(rbdriver*2/25.4)/cos(paSAP)*25.4/2;cout<<"rf driver= "<<rfdriver<<endl;


///////////////////////////////////////////////driven


/*calculating SAP*/

double SAPdriven;//in degrees

SAPdriven=180*((Tdriver+Tdriven)*tan(otpa)-Tdriver*tan(paom))/M_PI/Tdriven;cout<<"SAPdriven=
"<<SAPdriven<<endl;


/*calculating Qf*/

paSAP;

paSAP=atan(M_PI*SAPdriven/180);//cout<<"paSAP= "<<paSAP*180/M_PI<<endl;
```

```
/*calculating form dia at SAP  of gear*/

double rfdriven;

rfdriven=(rbdriven*2/25.4)/cos(paSAP)*25.4/2;cout<<"rf driven= "<<rfdriven<<endl;

/////////
```

6.1.7.2 4.2 Evaluating Involute Profile

```
(EFE BERKAY GÜVEN(07.18.2018))

////////

/*THIS SECTION BASED ON A SOFTWARE PREPARED BY EFE BERKAY GUVEN WHO IS
DESIGNED SOURCE CODES OF THIS SECTION.THIS SORTWARE'S DUTY IS

GENERATING COORDINAATES OF INVOLUTE PROFILE OF ASYMMETRIC GEARS*/


cout<<"\nFINDING PRESSURE ANGLE ERROR "<<endl;


double paedn;

cout<<"enter pressure angle error on driven in degrees:"<<endl;

cin>>paedn;

double paedr;

cout<<"enter pressure angle error on driver in degrees:"<<endl;

cin>>paedr;

ofstream dosya("coordinates.txt");

            long double r,bc,r2,w,st,tk;

            long double w11,w22,wa;

            long double x1,x2,y1,y2;

            long double fip,seg;


            //FORMULAS THAT USED TO  FIND COORDINATES

            r=0.38*mn;

   bc=M_PI*mn/4.;


/////////////////////////////////////////////////////////////////// DRIVER'S PAE

/////////

//aopn için tip ve bottomda x2 degerlerinin bulunması

////////
```

```
            st=(-rpdriver*sin(aopn*M_PI/180.))+sqrt(pow(rtdriver,2.0)-
pow((rpdriver*cos(aopn*M_PI/180.)),2.0));//ara formül

   tk=st*sin(aopn*M_PI/180.);//ara formül


            //eg region

            dosya<<"eg region "<<endl;

            w11=-(mn/cos(aopn*M_PI/180.));//eg aralıgını tanımlamak için parametre

   w22=(tk/cos(aopn*M_PI/180.));//eg aralıgını tanımlamak için parametre

   int stepeg;

   cout<<"input stepnumber for involute curve"<<endl;

   cin>>stepeg;

   wa=(w22-w11)/stepeg;//eg aralıgını tanımlamak için parametre


   int temprf=-1;

   double RFtemp;

      dosya<<"x2        "<<"y2        "<<"RFtemp"<<endl;

   for(double j=w11;j<=w22;j=j+wa)

                  {temprf++;

                  x1=(j*cos(aopn*M_PI/180.));

      y1=((bc)-(j*sin(aopn*M_PI/180.)));

      fip=((-x1)+(y1*tan(aopn*M_PI/180.)))/(rpdriver*tan(aopn*M_PI/180.));//ara formül

      seg=rpdriver*(fip);//ara formül

      y2=x1*cos(fip)-y1*sin(fip)+rpdriver*cos(fip)+seg*sin(fip);

      x2=x1*sin(fip)+y1*cos(fip)+rpdriver*sin(fip)-seg*cos(fip);

            RFtemp=pow(x2,2.)+pow(y2,2.);

                  nearRF[temprf]=pow(RFtemp,0.5);

                  dosya<<fixed<<setprecision(6)<<x2<<"  "<<fixed<<setprecision(6)<<y2<<"
"<<nearRF[temprf]<<"  "<<temprf<<endl;

                  }

                  if(temprf !=stepeg){

                  temprf++;

                  double j=w22;

                  x1=(j*cos(aopn*M_PI/180.));

      y1=((bc)-(j*sin(aopn*M_PI/180.)));
```

```
fip=((-x1)+(y1*tan(aopn*M_PI/180.)))/(rpdriver*tan(aopn*M_PI/180.));//ara formül

seg=rpdriver*(fip);//ara formül

y2=x1*cos(fip)-y1*sin(fip)+rpdriver*cos(fip)+seg*sin(fip);

x2=x1*sin(fip)+y1*cos(fip)+rpdriver*sin(fip)-seg*cos(fip);

    RFtemp=pow(x2,2.)+pow(y2,2.);

        nearRF[temprf]=pow(RFtemp,0.5);

        dosya<<fixed<<setprecision(6)<<x2<<"  "<<fixed<<setprecision(6)<<y2<<"
"<<nearRF[temprf]<<"  "<<temprf<<endl;

        }

/////////
```

FINDING PAE

```
////////

    int temptip=temprf;

    dosya<<"temptip= "<<temptip<<endl;

    double tipx2=x2;

for(int j=0;j<=temptip;j++){

dosya<<nearRF[j]<<"  "<<j<<endl;

    }

    int tempminus=0;

    int tempplus=0;

    dosya<<"\nfarkları= "<<endl;

for(int i=0;i<=temptip;i++)

    {

if(nearRF[i]-rfdriver<0){RFminus[tempminus]=nearRF[i]-rfdriver;

        dosya<<RFminus[tempminus]<<"   "<<i<<endl;tempminus++;}

if(nearRF[i]-rfdriver>=0){RFplus[tempplus]=nearRF[i]-rfdriver;

        dosya<<RFplus[tempplus]<<"   "<<i<<endl;tempplus++;}

    }

    int found;

if(-1*RFminus[tempminus-1]<=RFplus[0]){
```

```
            found=tempminus-1;
}
if(-1*RFminus[tempminus-1]>RFplus[0]){
            found=tempminus;
}

            //cout<<"found= "<<found<<endl;
            dosya<<"found= "<<found<<endl;

                        x1=((w11+wa*(found))*cos(aopn*M_PI/180.));
       y1=((bc)-((w11+wa*(found))*sin(aopn*M_PI/180.)));
       fip=((-x1)+(y1*tan(aopn*M_PI/180.)))/(rpdriver*tan(aopn*M_PI/180.));//ara formül
       seg=rpdriver*(fip);//ara formül
       y2=x1*cos(fip)-y1*sin(fip)+rpdriver*cos(fip)+seg*sin(fip);
       x2=x1*sin(fip)+y1*cos(fip)+rpdriver*sin(fip)-seg*cos(fip);
            double nearestRF= pow((pow(x2,2.)+pow(y2,2.)),0.5);
   double bottomx2=x2;
   double bottomy2=y2;
            dosya<<endl;

   cout<<"nearestRF = "<<nearestRF<<endl;
   dosya<<"nearestRF = "<<nearestRF<<endl;

   //cout<<"bottomx2= "<<bottomx2<<endl;
   //cout<<"bottomy2= "<<bottomy2<<endl;
            //cout<<"tipx2= "<<tipx2<<endl;
   dosya<<"bottomx2= "<<bottomx2<<endl;
   dosya<<"bottomy2= "<<bottomy2<<endl;
            dosya<<"tipx2= "<<tipx2<<endl;


/////////
//aopn+pae için tip ve bottomda x2 degerlerinin bulunması
///////
```

```
st=(-rpdriver*sin((aopn+paedr)*M_PI/180.))+sqrt(pow(rtdriver,2.0)-
pow((rpdriver*cos((aopn+paedr)*M_PI/180.)),2.0));//ara formül

tk=st*sin((aopn+paedr)*M_PI/180.);//ara formül


        //eg region

        dosya<<"\neg region pae "<<endl;

        w11=-(mn/cos((aopn+paedr)*M_PI/180.));//eg aralıgını tanımlamak için parametre

w22=(tk/cos((aopn+paedr)*M_PI/180.));//eg aralıgını tanımlamak için parametre

wa=(w22-w11)/stepeg;//eg aralıgını tanımlamak için parametre


temprf=-1;

RFtemp;

  dosya<<"x2        "<<"y2        "<<"RFtemp"<<endl;

for(double j=w11;j<=w22;j=j+wa)

            {temprf++;

            x1=(j*cos((aopn+paedr)*M_PI/180.));

  y1=((bc)-(j*sin((aopn+paedr)*M_PI/180.)));

  fip=((-x1)+(y1*tan((aopn+paedr)*M_PI/180.)))/(rpdriver*tan((aopn+paedr)*M_PI/180.));//ara formül

  seg=rpdriver*(fip);//ara formül

  y2=x1*cos(fip)-y1*sin(fip)+rpdriver*cos(fip)+seg*sin(fip);

  x2=x1*sin(fip)+y1*cos(fip)+rpdriver*sin(fip)-seg*cos(fip);

            RFx2[temprf]=x2;

            RFy2[temprf]=y2;

        RFtemp=pow(x2,2.)+pow(y2,2.);

            nearRF[temprf]=pow(RFtemp,0.5);

            dosya<<fixed<<setprecision(6)<<x2<<"  "<<fixed<<setprecision(6)<<y2<<"
"<<nearRF[temprf]<<"  "<<temprf<<endl;


            }

            if(temprf !=stepeg){

            temprf++;

            double j=w22;

            x1=(j*cos((aopn+paedr)*M_PI/180.));

  y1=((bc)-(j*sin((aopn+paedr)*M_PI/180.)));
```

```
	fip=((-x1)+(y1*tan((aopn+paedr)*M_PI/180.)))/(rpdriver*tan((aopn+paedr)*M_PI/180.));//ara formül
	seg=rpdriver*(fip);//ara formül
	y2=x1*cos(fip)-y1*sin(fip)+rpdriver*cos(fip)+seg*sin(fip);
	x2=x1*sin(fip)+y1*cos(fip)+rpdriver*sin(fip)-seg*cos(fip);
			RFx2[temprf]=x2;
			RFy2[temprf]=y2;
		RFtemp=pow(x2,2.)+pow(y2,2.);
			nearRF[temprf]=pow(RFtemp,0.5);
			dosya<<fixed<<setprecision(6)<<x2<<"  "<<fixed<<setprecision(6)<<y2<<"
"<<nearRF[temprf]<<"  "<<temprf<<endl;
			}


		double paetemptip=temprf;
		dosya<<"paetemptip= "<<paetemptip<<endl;
		double tipx2pae=x2;
		tempminus=0;
		tempplus=0;


		dosya<<"\nfarkları=RFy2[i]-bottomy2 "<<endl;
		dosya<<"bottomy20 "<<bottomy2<<endl;
   for(int i=0;i<=paetemptip;i++)
		{
   if(RFy2[i]-bottomy2<=0){RFy2minus[tempminus]=RFy2[i]-bottomy2;
			dosya<<RFy2minus[tempminus]<<"   "<<i<<endl;tempminus++;}
   if(RFy2[i]-bottomy2>0){RFy2plus[tempplus]=RFy2[i]-bottomy2;
			dosya<<RFy2plus[tempplus]<<"   "<<i<<endl;tempplus++;}
   }
		int foundpae;
if(-1*RFy2minus[tempminus-1]<=RFy2plus[0]){
		foundpae=tempminus-1;
}
if(-1*RFy2minus[tempminus-1]>RFy2plus[0]){
		foundpae=tempminus;
}
```

```
            //cout<<"found= "<<foundpae<<endl;
            dosya<<"found= "<<foundpae<<endl;
   double bottomx2pae=RFx2[foundpae];
   double bottomy2pae=RFy2[foundpae];
            dosya<<endl;



   //cout<<"bottomx2pae= "<<bottomx2pae<<endl;
   //cout<<"bottomy2pae= "<<bottomy2pae<<endl;
            //cout<<"tipx2pae= "<<tipx2pae<<endl;
   dosya<<"bottomx2= "<<bottomx2pae<<endl;
   dosya<<"bottomy2= "<<bottomy2pae<<endl;
            dosya<<"tipx2= "<<tipx2pae<<endl;






            cout<<"PAE at tip= "<<(tipx2pae-tipx2)*1000<<endl;
            cout<<"PAE at bottom(fromSAP)= "<<(bottomx2pae-bottomx2)*1000<<endl;
            double PAEtipdriver=(tipx2pae-tipx2)*1000;
            double PAESAPdriver=(bottomx2pae-bottomx2)*1000;



//////////////////////////////////////////////////////////DRIVEN


/////////
//aopn için tip ve bottomda x2 degerlerinin bulunması
////////


            st=(-rpdriven*sin(aopn*M_PI/180.))+sqrt(pow(rtdriven,2.0)-
pow((rpdriven*cos(aopn*M_PI/180.)),2.0));//ara formül
   tk=st*sin(aopn*M_PI/180.);//ara formül


            //eg region
```

```
dosya<<"eg region "<<endl;
w11=-(mn/cos(aopn*M_PI/180.));//eg aralıgını tanımlamak için parametre
w22=(tk/cos(aopn*M_PI/180.));//eg aralıgını tanımlamak için parametre

wa=(w22-w11)/stepeg;//eg aralıgını tanımlamak için parametre

temprf=-1;
dosya<<"x2      "<<"y2      "<<"RFtemp"<<endl;
for(double j=w11;j<=w22;j=j+wa)
{temprf++;
x1=(j*cos(aopn*M_PI/180.));
y1=((bc)-(j*sin(aopn*M_PI/180.)));
fip=((-x1)+(y1*tan(aopn*M_PI/180.)))/(rpdriven*tan(aopn*M_PI/180.));//ara formül
seg=rpdriven*(fip);//ara formül
y2=x1*cos(fip)-y1*sin(fip)+rpdriven*cos(fip)+seg*sin(fip);
x2=x1*sin(fip)+y1*cos(fip)+rpdriven*sin(fip)-seg*cos(fip);
RFtemp=pow(x2,2.)+pow(y2,2.);
nearRF[temprf]=pow(RFtemp,0.5);
dosya<<fixed<<setprecision(6)<<x2<<"  "<<fixed<<setprecision(6)<<y2<<"
"<<nearRF[temprf]<<"  "<<temprf<<endl;
}
if(temprf !=stepeg){
temprf++;
double j=w22;
x1=(j*cos(aopn*M_PI/180.));
y1=((bc)-(j*sin(aopn*M_PI/180.)));
fip=((-x1)+(y1*tan(aopn*M_PI/180.)))/(rpdriven*tan(aopn*M_PI/180.));//ara formül
seg=rpdriven*(fip);//ara formül
y2=x1*cos(fip)-y1*sin(fip)+rpdriven*cos(fip)+seg*sin(fip);
x2=x1*sin(fip)+y1*cos(fip)+rpdriven*sin(fip)-seg*cos(fip);
RFtemp=pow(x2,2.)+pow(y2,2.);
nearRF[temprf]=pow(RFtemp,0.5);
dosya<<fixed<<setprecision(6)<<x2<<"  "<<fixed<<setprecision(6)<<y2<<"
"<<nearRF[temprf]<<"  "<<temprf<<endl;
```

```
                        }
/////////
```

6.1.7.3 4.3 Finding PAE

```
////////


            temptip=temprf;
            dosya<<"temptip= "<<temptip<<endl;
            tipx2=x2;
   for(int j=0;j<=temptip;j++){
   dosya<<nearRF[j]<<"  "<<j<<endl;
            }


            tempminus=0;
            tempplus=0;
            dosya<<"\nfarkları= "<<endl;
   for(int i=0;i<=temptip;i++)
            {
   if(nearRF[i]-rfdriven<0){RFminus[tempminus]=nearRF[i]-rfdriven;
                           dosya<<RFminus[tempminus]<<"   "<<i<<endl;tempminus++;}
   if(nearRF[i]-rfdriven>=0){RFplus[tempplus]=nearRF[i]-rfdriven;
                           dosya<<RFplus[tempplus]<<"   "<<i<<endl;tempplus++;}


            }


if(-1*RFminus[tempminus-1]<=RFplus[0]){
            found=tempminus-1;
}
if(-1*RFminus[tempminus-1]>RFplus[0]){
            found=tempminus;
}


            //cout<<"found= "<<found<<endl;
            dosya<<"found= "<<found<<endl;
```

```
                    x1=((w11+wa*(found))*cos(aopn*M_PI/180.));
    y1=((bc)-((w11+wa*(found))*sin(aopn*M_PI/180.)));
    fip=((-x1)+(y1*tan(aopn*M_PI/180.)))/(rpdriven*tan(aopn*M_PI/180.));//ara formül
    seg=rpdriven*(fip);//ara formül
    y2=x1*cos(fip)-y1*sin(fip)+rpdriven*cos(fip)+seg*sin(fip);
    x2=x1*sin(fip)+y1*cos(fip)+rpdriven*sin(fip)-seg*cos(fip);
        nearestRF= pow((pow(x2,2.)+pow(y2,2.)),0.5);
  bottomx2=x2;
  bottomy2=y2;
        dosya<<endl;

  cout<<"nearestRF = "<<nearestRF<<endl;
  dosya<<"nearestRF = "<<nearestRF<<endl;

  //cout<<"bottomx2= "<<bottomx2<<endl;
  //cout<<"bottomy2= "<<bottomy2<<endl;
        //cout<<"tipx2= "<<tipx2<<endl;
  dosya<<"bottomx2= "<<bottomx2<<endl;
  dosya<<"bottomy2= "<<bottomy2<<endl;
        dosya<<"tipx2= "<<tipx2<<endl;



/////////
//aopn+pae için tip ve bottomda x2 degerlerinin bulunması
////////
  st=(-rpdriven*sin((aopn+paedn)*M_PI/180.))+sqrt(pow(rtdriven,2.0)-
pow((rpdriven*cos((aopn+paedn)*M_PI/180.)),2.0));//ara formül
  tk=st*sin((aopn+paedn)*M_PI/180.);//ara formül

        //eg region
        dosya<<"\neg region pae "<<endl;
        w11=-(mn/cos((aopn+paedn)*M_PI/180.));//eg aralıgını tanımlamak için parametre
  w22=(tk/cos((aopn+paedn)*M_PI/180.));//eg aralıgını tanımlamak için parametre
```

```
wa=(w22-w11)/stepeg;//eg aralıgını tanımlamak için parametre


temprf=-1;
RFtemp;
   dosya<<"x2        "<<"y2        "<<"RFtemp"<<endl;
for(double j=w11;j<=w22;j=j+wa)
            {temprf++;
            x1=(j*cos((aopn+paedn)*M_PI/180.));
   y1=((bc)-(j*sin((aopn+paedn)*M_PI/180.)));
   fip=((-x1)+(y1*tan((aopn+paedn)*M_PI/180.)))/(rpdriven*tan((aopn+paedn)*M_PI/180.));//ara formül
   seg=rpdriven*(fip);//ara formül
   y2=x1*cos(fip)-y1*sin(fip)+rpdriven*cos(fip)+seg*sin(fip);
   x2=x1*sin(fip)+y1*cos(fip)+rpdriven*sin(fip)-seg*cos(fip);
            RFx2[temprf]=x2;
            RFy2[temprf]=y2;
        RFtemp=pow(x2,2.)+pow(y2,2.);
            nearRF[temprf]=pow(RFtemp,0.5);
            dosya<<fixed<<setprecision(6)<<x2<<"  "<<fixed<<setprecision(6)<<y2<<"
"<<nearRF[temprf]<<"  "<<temprf<<endl;


            }
            if(temprf !=stepeg){
            temprf++;
            double j=w22;
            x1=(j*cos((aopn+paedn)*M_PI/180.));
   y1=((bc)-(j*sin((aopn+paedn)*M_PI/180.)));
   fip=((-x1)+(y1*tan((aopn+paedn)*M_PI/180.)))/(rpdriven*tan((aopn+paedn)*M_PI/180.));//ara formül
   seg=rpdriven*(fip);//ara formül
   y2=x1*cos(fip)-y1*sin(fip)+rpdriven*cos(fip)+seg*sin(fip);
   x2=x1*sin(fip)+y1*cos(fip)+rpdriven*sin(fip)-seg*cos(fip);
            RFx2[temprf]=x2;
            RFy2[temprf]=y2;
        RFtemp=pow(x2,2.)+pow(y2,2.);
            nearRF[temprf]=pow(RFtemp,0.5);
```

```
                        dosya<<fixed<<setprecision(6)<<x2<<"  "<<fixed<<setprecision(6)<<y2<<"
"<<nearRF[temprf]<<"  "<<temprf<<endl;
                        }


            paetemptip=temprf;
            dosya<<"paetemptip= "<<paetemptip<<endl;
            tipx2pae=x2;
            tempminus=0;
            tempplus=0;


            dosya<<"\nfarkları=RFy2[i]-bottomy2 "<<endl;
            dosya<<"bottomy20 "<<bottomy2<<endl;
   for(int i=0;i<=paetemptip;i++)
            {
   if(RFy2[i]-bottomy2<=0){RFy2minus[tempminus]=RFy2[i]-bottomy2;
                          dosya<<RFy2minus[tempminus]<<"   "<<i<<endl;tempminus++;}
   if(RFy2[i]-bottomy2>0){RFy2plus[tempplus]=RFy2[i]-bottomy2;
                          dosya<<RFy2plus[tempplus]<<"   "<<i<<endl;tempplus++;}
   }
            foundpae;
if(-1*RFy2minus[tempminus-1]<=RFy2plus[0]){
            foundpae=tempminus-1;
}
if(-1*RFy2minus[tempminus-1]>RFy2plus[0]){
            foundpae=tempminus;
}
            //cout<<"found= "<<foundpae<<endl;
            dosya<<"found= "<<foundpae<<endl;
   bottomx2pae=RFx2[foundpae];
   bottomy2pae=RFy2[foundpae];
            dosya<<endl;



   //cout<<"bottomx2pae= "<<bottomx2pae<<endl;
```

```
    //cout<<"bottomy2pae= "<<bottomy2pae<<endl;
            //cout<<"tipx2pae= "<<tipx2pae<<endl;
    dosya<<"bottomx2= "<<bottomx2pae<<endl;
    dosya<<"bottomy2= "<<bottomy2pae<<endl;
            dosya<<"tipx2= "<<tipx2pae<<endl;






            cout<<"PAE at tip= "<<(tipx2pae-tipx2)*1000<<endl;
            cout<<"PAE at bottom(fromSAP)= "<<(bottomx2pae-bottomx2)*1000<<endl;

            double PAEtipdriven=(tipx2pae-tipx2)*1000;
            double PAESAPdriven=(bottomx2pae-bottomx2)*1000;

// total presurre angle error hangi dişlinin driver olmasına veya driven olmasına göre değişir.
//deformasyon hesapları dişli sayısı büyük olana göre yapıldıgı için(pitch error u tamamıyla görebilmek için)
//burada total pressure angle error u bulurken hangi dişli daha cok dişe sahipse ona göre yapılır.
//total pae driver a göre bulunuyor.
double totalpaeSAPdriver=PAESAPdriver+PAEtipdriven;
double totalpaetipdriver=PAEtipdriver+PAESAPdriven;
//total pae driven a göre bulunuyor.
double totalpaetipdriven=PAEtipdriven+PAESAPdriver;
double totalpaeSAPdriven=PAESAPdriven+PAEtipdriver;

//driven ve driver arasında hangisinin diş sayısı daha cok ise ona göre pae belirleniyor.
cout<<"total PAE bulunurken disli sayisi buyuk olanın driven yada driver olmasina dikkat edilmistir."<<endl;

while(1){
    if(Tbigger=Tdriven){
            cout<<"totalpae of driven"<<endl;
    totalpae1=totalpaetipdriven;cout<<"totalpaeSAP1="<<totalpae1<<endl;
    totalpae2=totalpaeSAPdriven;cout<<"totalpaetip2="<<totalpae2<<endl;}
            if(Tbigger=Tdriver){
```

```
            cout<<"totalpae of driver"<<endl;
    totalpae1=totalpaeSAPdriver;cout<<"totalpaeSAP1="<<totalpae1<<endl;
    totalpae2=totalpaetipdriver;cout<<"totalpaetip2="<<totalpae2<<endl;}
    break;
}
//PAE hatalarını içeren matrix in olusturulması
//matrix olusturulurken linear dogru denklemi kullanıldı.
//          cout<<"Sonuclar coordinates.txt dosyasi olarak programin bulundugu klasore aktarildi."<<endl;
    cout<<endl;
  break;}
if(anspae=='n'||anspae=='N'){totalpae1=0;totalpae2=0;break;}
}
//pressure angle error larını içeren matrixin olusturulması


///////////////
```

6.1.8 5.Adding Reliefs

```
///////////////


/////////
```

6.1.8.1 5.1.Amount Of Relief

```
////////
//WE FIRST SUGGESTING A AR AND ER VALUE TO THE USER FOR GETTING MINIMUM TEPP.IF USER REJECTS USE THEM, HE/SHE CAN ENTER A SPECIFIC VALUE.
/*
Amount of relief is usually estimated as the deflection of the gear mesh at the maximum load for spur gears
Linear relief was the simplest shape to employ at the beginning of the analysis since it,
with other assumptions,made the analysis easy to understand and proceed quickly during desing.
However,there are lots of different mathematical shapes or forms to employ if the feasibility and
cost of manufacturing them is not taken into consideration.
*/
/*In this program we only handle with Linear case.*/


            double Fd,k,arL,arR;
/*
```

2.2.kısım için nomenclature

Fd=design load;

k=stiffness;

arL=leftside amount of relief

arR=right side amount of relief

*/

```
// WE SUGGESTING A AR VALUE TO GET MINIMUM TEPP
        cout<<"enter Design Load(N),stiffness(N/mm/micron) respectively"<<endl;
        cin>>Fd>>k;
cout<<endl;
        while(1){
            cout<<"suggested ar(micron) for both side="<<Fd/k/bb<<endl;

            char answ;//amount of relief(ar)in önerilen olup olmaması soruluyor.
                        cout<<"For change ar enter 'Y'"<<endl;cin>>answ;

//Eger ar önerilen olmayacak ve kullanıcı tarafından girilecek ise arLeft ve arRight isteniyor.
            if(answ=='y'||answ=='Y'){cout<<"enter ar(micron)left: ";cin>>arL;
cout<<"enter ar(micron) right:";
 cin>>arR;
break;}
//Eger ar önerilen olacaksa sağ ve sol tarafın ar si F/k/b olacak.

            else{arL=Fd/k/bb;arR=Fd/k/bb;break;}}
               cout<<endl;

        cout<<"arLeft= "<<arL<<endl;
        cout<<"arRigth= "<<arR<<endl;
cout<<endl;
        allin<<"arLeft=  "<<arL<<endl;
        allin<<"arRigth=  "<<arR<<endl;
```

//////////////

6.1.8.2 5.2.Extend Of Relief

//////////////

```
//WE SUGGESTING A ER TO GET MINIMUM TEPP

        double er1,er2;//extend of relief LEFT=1 ve RIGHT=2;
        cout<<"recommended er: "<<Lc-pb<<"  for the TEpp=0."<<endl;



while(1){char answe;
            cout<<"are you want to use recommended er?('Y' or 'N')"<<endl;
                    cin>>answe;

              if(answe=='y'||answe=='Y'){er1=Lc-pb;er2=Lc-pb;
                                        cout<<"er1(left)= "<<er1<<"\n"<<"er2(right)=
"<<er2<<endl;
                                                    break;}
              if(answe=='n'||answe=='N'){cout<<"enter er1 & er2\n"<<(Lc-pb)/2<<"<=er>="<<Lc/2<<endl;
                                        cin>>er1>>er2;
                                                    break;}}
        cout<<"er1 ="<<er1<<endl;
        cout<<"er2 ="<<er2<<endl;

        allin<<"er1 ="<<er1<<endl;
        allin<<"er2 ="<<er2<<endl;
```

//////////////

6.1.9 <u>6.Setting Isolated Te Curve</u>

//////////////

```
//WE FIRST DIVIDING Pb INTO XPB NODES WHICH IS PLACED AT DELX INTERVALS.
```

```
    double delx;//delx=MESH NUMBER
    int xpb;//xpb=number of points for pb;
            cout<<endl;
            cout<<"input number of points as power of 2 which is for pb(base pitch)"<<endl;
            cin>>xpb;

            delx=pb/xpb;

//          cout<<"xpb= "<<xpb<<endl;
//          cout<<"delx="<<delx<<endl;
            allin<<"xpb= "<<xpb<<endl;
            allin<<"delx="<<delx<<endl;

//LLi olusturulması
int xB=Lc/2/delx;
if(xB*delx!=Lc/2){xB++;}
int xer2=er2/delx;
if(xer2*delx!=er2){xer2++;}
int tempLLiB=1;
//LLi olusturulması
for (int i=0;i<=xpb;i++)
            {
                   if(i<=xB-xer2){LLi[i]=0;
                                        //cout<<"LLi["<<i<<"]a="<<LLi[i]<<endl;
                                                            }
                   if(i>xB-xer2&&i<xB){LLi[i]=LLi[i-1]+arR/(er2/delx);
                                             // cout<<"LLi["<<i<<"]b="<<LLi[i]<<endl;
                                                                              }
                   if(i>=xB&&delx<=xpb){LLi[i]=arR;
                                          //cout<<"LLi["<<i<<"]c="<<LLi[i]<<endl;
                                                                    }}
```

```
//LRi olusturulması

int xer1=er1/delx;

if(xer1*delx!=er1){xer1++;}

int tempLRiB=1;

            for (int i=0;i<=xpb;i++)

            {       double dist=delx*i;

                  if(i<=xpb-xB){LRi[i]=arL;

                                          // cout<<"LRi["<<i<<"]a="<<LRi[i]<<endl;

                                                                  }

                  if(i>xpb-xB&&i<xpb-xB+xer1){LRi[i]=LRi[i-1]-arL/(er1/delx);


                        //cout<<"LRi["<<i<<"]b="<<LRi[i]<<endl;


                        }

                  if(i>=xpb-xB+xer1&&i<=xpb){LRi[i]=0.;

                                                //cout<<"LRi["<<i<<"]c="<<LRi[i]<<endl;

                                                                                    }}


//LLi cıktı verilmesi

            for (int i=0;i<xpb+1;i++){allin<<"LLi["<<setw(6)<<setfill(' ')<<i<<"]= "<<fixed<<setprecision(4)<<-
1*LLi[i]<<endl;}

            allin<<endl;

            for (int i=0;i<xpb+1;i++){LL<<-1*LLi[i]<<endl;}

//LRi cıktı verilmesi

            for (int i=0;i<xpb+1;i++){allin<<"LRi["<<setw(6)<<setfill(' ')<<i<<"]= "<<fixed<<setprecision(4)<<-
1*LRi[i]<<endl;}

            allin<<endl;

            for (int i=0;i<xpb+1;i++){LR<<-1*LRi[i]<<endl;}


//Li olusturulması

            int xLc=2*xB;


//          cout<<"xLc= "<<xLc<<endl;

            allin<<"xLc="<<xLc<<endl;
```

```
            int temp=0;

    for(int i=xpb-xB;i<=xpb;i++){L[temp]=LRi[i];temp++;}//1.döngü

    for(int i=1;i<=xB;i++){L[temp]=LLi[i];temp++;}  //2.döngü

/*1.döngünün son noktası 2.döngünün ilk noktasına denk geldiği için ,bu iki noktanın çakışmasını engellemek
için

2. döngünün indexi 1 den başlatıldı.*/

//Li cıktı verilmesi

double mpae=(totalpae2-totalpae1)/xLc;

double bpae=totalpae1;

for(int i=0;i<=xLc;i++){PAE[i]=mpae*i+totalpae1;}

for(int i=0;i<=xLc;i++){//cout<<"PAE["<<i<<"]= "<<PAE[i]<<endl;

}

            for(int i=0;i<=xLc;i++){L[i]=L[i]-PAE[i];

                        //cout<<"Li["<<i<<"]="<<-1*L[i]<<endl;

                                                }

double lureleft=L[0];

double lureright=L[xLc];

            for(int i=0;i<=xLc;i++){allin<<"Li["<<i<<"]="<<-1*L[i]<<endl;}

    for(int i=0;i<=xLc;i++){Li<<-1*L[i]<<endl;}

            Li<<endl;

            allin<<endl;



//BURAYA KADAR OLAN KISIMDA ELDE EDİLEN DEGERLER SPUR GEAR MANTIGIYLA
OLUSTULMUSTUR.



//////////////////////////////
```

6.1.10 <u>7.Thin Slice Theory</u>

```
//////////////////////////////


/*

Helical gear is 3D object.A spur gear is in fact a special case of helical gears when the helix angle
```

becomes ZERO. However,a spur gear is conventionally regarded as 2D , as the basic geometry can be clearly reperesented by its transverse section.
On this ground,helical gears are analogous to a set of stepped spur gears which consist of a number of identical spur gears so arranged that teeth of each individual meember are slightly out of phase relative to each other.
A helical gear can be thought of as an infinite number of such stepped gears, each a lamination of infinitesimal thickness,placed side by side,and each with a slight phase diffrence.
The same approximation is called the 'THIN SLICE THEORY' and used for modelling helical gears.
*/

```
/*
7.kısım için nomenclature
tsn=thin slice number
ss:shift of each slice
ss/delx=dogruluk
xss=ss için işlem(mesh)sayısı
xtsn=bir helix dişli için slicelar eklendikten somraki mesh sayısı
//yani xLc=spur gear için xtsn=helical gear için aynı anlamı ifade ediyor
*/
   int tsn,xss;
if(helix!=0){
   check:;
        cout<<"input the thin slice numbers= "<<endl;
        cin>>tsn;
        cout<<endl;
        allin<<"tsn= "<<tsn<<endl;

//ss in hesaolanması
        double ss=bb*tan(M_PI*helix/180.0)/tsn;//True
//      cout<<"ss="<<ss<<endl;
        allin<<"ss=  "<<ss<<endl;
//dogruluğun hesaplanması
        allin<<"ss/delx="<<ss/delx<<endl;//True
```

```
//xss in;programın dogrulugunu arttırmak için ona en yakın sayıya yuvarlanması
   if(ss/delx<.5){
            cout<<"girilen tsn degeri sonucu olusan ss/delx 1'ten kucuktur.TSN degeri
<="<<bb*tan(M_PI*helix/180.0)/delx<<endl;
   goto check;
            }
            xss=round(ss/delx);
}


if(helix==0){
   xss=0;
            cout<<"input the thin slice numbers= "<<endl;
            cin>>tsn;
            cout<<endl;
            allin<<"tsn= "<<tsn<<endl;
}
//xtsn in hesaplanması
            int xtsn=xLc+(tsn-1)*xss;
//          cout<<"xss="<<xss<<endl;
//          cout<<"xtsn="<<xtsn<<endl;
            allin<<"xss="<<xss<<endl;
            allin<<"xtsn="<<xtsn<<endl;


/////////
```

6.1.10.1 7.1.Misalignment

```
////////


/*
Mısalıgnment of each slice is simulated with appropriate amount of PE on the corresponding slice.
Each slice will have a separate value of PE due to crowning.
*/


double aomis;
cout<<"enter angle of misalignmet in degrees: "<<endl;
```

```
cin>>aomis;
double MA[tsn];

for(int i=0;i<tsn;i++){
        MA[i]=i*(bb/tsn)*aomis*M_PI/180;
//      cout<<"MA["<<i<<"]= "<<MA[i]<<endl;
}

/////////
```

6.1.10.2 7.2.Crowning

```
////////
/*
Crowning reduces the overstressing effect in misalignment.On the other hand,over crowning may cause reduced effecive facewidth and high tooth stresses.
Calculatıon procedure for crowning is the same as the misalıgnment.That is crowning value for each slice is converted to a PE value on this slice.
Each slice will have a separate value of PE due to crowning.
*/
double maxcro;
double CRO[tsn];
while(1){
char anscro;
cout<<"is there any crowning input Y or N "<<endl;
cin>>anscro;

if(anscro=='y'||anscro=='Y')
{
double ac,b1,b2,A,B,xb2,xb1;

double tsndelx=bb/tsn;
while(1){char answe;
            cout<<"what is the type of crowning\nLinear(input L) or Parabolic(input P)"<<endl;
                    cin>>answe;
```

```
if(answe=='l'||answe=='L'){
        cout<<"input AC(amount of crowning at face) "<<endl;
          cin>>ac;
        cout<<"input b1<="<<bb/2<<" distance as extent of first region:"<<endl;
          cin>>b1;

        xb1=b1/tsndelx;
        xb1=(int)round(xb1);
        b2=bb-b1;
        xb2=tsn-xb1;
        //cout<<"b2 distance:"<<b2<<endl;
        //cout<<"xb1: "<<xb1<<endl;
        //cout<<"xb2: "<<xb2<<endl;

        for(int i=0;i<tsn;i++){
              if(i<=xb1){CRO[i]=(xb1-i)*ac/xb1;}
              if(i>xb1&&i<xb2){CRO[i]=0;}
              if(i>=xb2){CRO[i]=(i-xb2+1)*ac/xb1;}
                                                            }
    break;}

if(answe=='p'||answe=='P'){

                        cout<<"input AC(amount of crowning at face) "<<endl;
          cin>>ac;
        cout<<"if y=A*x^B "<<endl;
                        cout<<"input A and B respectively "<<endl;
                          cin>>A>>B;


        double exp=1/B;
                                                            double
b1=pow(ac,exp);
```

```
cout<<"b1= "<<b1<<endl;
                xb1=b1/tsndelx;
                int xb11=xb1;
                b2=bb-b1;
                xb2=tsn-xb11;
                cout<<"b2 distance:"<<b2<<endl;
                cout<<"xb11: "<<xb11<<endl;
                cout<<"xb2: "<<xb2<<endl;

                        for(int i=0;i<tsn;i++){
                    if(i<=xb11){CRO[i]=A*pow(xb1-(i*tsndelx),B);}
                    if(i>xb11&&i<xb2){CRO[i]=0;}
                    if(i>=xb2){CRO[i]=A*pow((i-xb2)*tsndelx,B);}}
                        break;}}
        cout<<endl;
    for(int i=0;i<tsn;i++){
        allin<<"CRO["<<i<<"]= "<<CRO[i]<<endl;
        }
cout<<endl;

for(int i=0;i<tsn;i++){
        if(i<xb2){
                if(CRO[i]<arR){CRO[i]=0.;}
                if(CRO[i]>=arR){CRO[i]=CRO[i]-arR;}
        }
        if(i>=xb2){
                if(CRO[i]<arL){CRO[i]=0.;}
                if(CRO[i]>=arL){CRO[i]=CRO[i]-arL;}
        }
}
for(int i=0;i<tsn;i++){
        //cout<<"CRO["<<i<<"]= "<<CRO[i]<<endl;
        }
```

```
cout<<endl;

while(1){
            if(CRO[0]>=CRO[tsn-1]){maxcro=CRO[0];}
            else maxcro=CRO[tsn-1];
            break;}
break;}




if(anscro=='n'||anscro=='N'){maxcro=0;
  for(int i=0;i<tsn;i++){
  CRO[i]=0;
  //cout<<"CRO["<<i<<"]= "<<CRO[i]<<endl;
            }break;}
}
//cout<<"maxcro= "<<maxcro<<endl;



//////////////
```

6.1.10.3 7.3.Generating sTS

```
//////////////

//sTS= thin slice theory nin spur gear hesaplanması sonucu olusturulan Li[] TE curvene uygulanmasıyla elde
//edilen ss mesafeyle thin slice number kadar ötelenmiş Li lerden olusur.

double AR;//arL ve arR arasında büyük olanıdır.
while(1){
            if(arL<=arR){AR=arR;break;}
            else AR=arL;break;}

//slice bitimini ayrırt edebilmek için öncelikle bütün degerler (-1)e eşitleniyor.
//daha sonra xtsn içinde xLc kadar kısımlar ss kadar peşi sıra öteleniyor.
```

//üstteki döngü sonucu acıkta kalan(-1 olan noktalar) herbir slice için acıkta_kalan=leak(xtsn-xLc) kadar kısım AR+lure'a eşitleniyor.

//bu işlemler herbir slice için xtsn in tam ortasının önce solu(j=0;j<=xLc/2+i*xss;j++)

//ve sonra sagı(j=xLc/2+i*xss+1;j<=xtsn;j++) için yapılmıstır.

```
//lure'ın bulunması

double luremax=abs(largest)+abs(smallest)+MA[tsn-1]+maxcro+1+abs(totalpae1)+abs(totalpae2);

//cout<<"lure= "<<lure<<endl;



////////////


        for(int i=0;i<tsn;i++){for(int j=0;j<=xtsn;j++){sTS[i][j]=-1.;}}


        for(int i=0;i<tsn;i++){for(int j=0;j<=xB;j++){sTS[i][j+xss*i]=L[j];}}


        for(int i=0;i<tsn;i++){
           for(int j=0;j<=xB+i*xss;j++){if(sTS[i][j]==-1.){sTS[i][j]=AR+luremax;}}}


        for(int i=0;i<tsn;i++){for(int j=xB+1;j<=xLc;j++){sTS[i][j+xss*i]=L[j];}}


        for(int i=0;i<tsn;i++){
          for(int j=xB+i*xss+1;j<=xtsn;j++){if(sTS[i][j]==-1.){sTS[i][j]=AR+luremax;}}}



//sTS in yazdırılması
        allin<<"mesh";
        for(int i=0;i<tsn;i++){allin<<" sTS["<<left<<setw(3)<<i<<"] ";}
        allin<<endl;


        for(int j=0;j<=xtsn;j++)
        {  allin<<setw(6)<<setfill(' ')<<left<<j;
          for(int i=0;i<tsn;i++){
```

```
		allin<<setw(10)<<setfill(' ')<<left<<fixed<<setprecision(3)<<-1*sTS[i][j];}
			allin<<endl;}
	allin<<endl;
```

//////////////

6.1.10.4 7.4. Addition Of Pitch Error,Misalignment,Crowning To sTS

//////////////

```
//bu hatalar bütün dişlilerin bütün slicelarına eklenip box[][] potasında oluşturulmustur.

double leak=xtsn-xpb;//acıkta kalan kısım
//cout<<"leak= "<<leak<<endl;
	cout<<endl;
	for(int i=0;i<Tbigger*tsn;i++){
	  int temp3=0;
		for(int k=0;k<=xpb*Tbigger+leak;k++){

			int temp1=i%tsn;
			int temp2=i/tsn;

			if(k<temp2*xpb){box[i][k]=AR+luremax;}

			if(k>=temp2*xpb&&k<=temp2*xpb+xtsn) {
					box[i][k]=sTS[temp1][temp3]-
PE[temp2]+MA[temp1]+CRO[temp1];temp3++;
					if(k<temp2*xpb+xss*temp1){box[i][k]=AR+luremax;}
					if(k>temp2*xpb+xLc+xss*temp1){box[i][k]=AR+luremax;}

			}

		  if(k>temp2*xpb+xtsn){box[i][k]=AR+luremax;}
```

```
                                      }
    }
    /*BOX<<"after STS"<<endl;
BOX<<endl;BOX<<endl;
              for(int j=0;j<=xpb*Tbigger+leak;j++)
              {  BOX<<setw(6)<<setfill(' ')<<left<<j;
                 for(int i=0;i<Tbigger*tsn;i++){
                      BOX<<setw(10)<<setfill(' ')<<left<<fixed<<setprecision(3)<<-1*box[i][j];}
                            BOX<<endl;}
BOX<<endl;BOX<<endl;BOX<<endl;*/
```

//herbir slice'ın TE bitim ve baslangıc kısımları AR+lure a eşilenmişti sTS'te.

//PE eklenmesiyle bu kısımların degerleri artar veya azalabilir,bunu önlemek ve farkını anlamak için

//bu kısımlar AR+1.2*lure'a eşitleniyor.

/////////

6.1.10.5 7.5.Finding Infinite

////////

//infinite ın bulunması için Fmax in kullanıcıdan istenmesi

```
    double y,max,min,Fmax,Fmin,inc,percent;
              cout<<"input multipliers min and max for Fmin=Fd*min&Fmax=Fd*max"<<endl;
              cin>>min>>max;
              cout<<endl;

              Fmax=Fd*max;
```

```
            Fmin=Fd*min;

//          cout<<"Fmin="<<Fmin<<endl;
//          cout<<"Fmax="<<Fmax<<endl;
            allin<<"Fmin="<<Fmin<<endl;
            allin<<"Fmax="<<Fmax<<endl;

double infinite=Fmax/k/bb+luremax+AR;

for(int i=0;i<Tbigger*tsn;i++)
{for(int k=0;k<=xpb*Tbigger+leak;k++){
            if(box[i][k]>=AR+luremax){box[i][k]=infinite;}
            }
}
/*BOX<<"after infinite"<<endl;
BOX<<endl;BOX<<endl;
            for(int j=0;j<=xpb*TG+leak;j++)
            {  BOX<<setw(6)<<setfill(' ')<<left<<j;
               for(int i=0;i<TG*tsn;i++){
                    BOX<<setw(10)<<setfill(' ')<<left<<fixed<<setprecision(3)<<-1*box[i][j];}
                         BOX<<endl;}
BOX<<endl;BOX<<endl;BOX<<endl;*/

/////////
```

6.1.10.6 7.6.boxkleak[][]

```
////////

int temp5=0;
            for(int i=0;i<Tbigger*tsn;i++){int temp6=0;
                        for(int k=xpb*Tbigger;k<=xpb*Tbigger+leak;k++){
                        boxleak[temp5][temp6]=box[i][k];temp6++;}temp5++;}

int tsni;
```

```
// acıkta kalan kısmın kacıncı thin slice da oldugunun bulunması//tsni

for(int k=0;k<1;k++){
        for(int i=1;i<Tbigger*tsn;i++){
                if(boxleak[i][k]!=infinite){tsni=i;
                break;
                } }}

//cout<<"tsni= "<<tsni<<endl;
allin<<"tsni= "<<tsni<<endl;

int tsnleak=Tbigger*tsn-tsni;//kaç thin slice in acıkta kaldıgı=tsnleak

//cout<<"tsnleak= "<<tsnleak<<endl;
allin<<"tsnleak= "<<tsnleak<<endl;

// boxkleak[][] matrixine son hali veriliyor

for(int i=0;i<tsnleak;i++)
{for(int k=0;k<=xpb*Tbigger;k++){
        if(k<=leak){boxleakf[i][k]=boxleak[i+tsni][k];}
        if(k>leak){boxleakf[i][k]=infinite;}
        }
}
//olusturulan boxf[][] potasına ilk dişli ile leak kadar son kısımdaki tsn lerin temasının eklenmesi
```

```
        for(int k=0;k<=xpb*Tbigger;k++){int tempf=0;
          for(int i=0;i<Tbigger*tsn+tsnleak;i++){
                  if(i<tsnleak){boxf[i][k]=boxleakf[i][k];}


                  if(i>=tsnleak){boxf[i][k]=box[tempf][k];tempf++;}


                }
        }
////
/*
BOX<<endl;BOX<<endl;
        for(int j=0;j<=xpb*TG+leak;j++)
        {  BOX<<setw(6)<<setfill(' ')<<left<<j;
           for(int i=0;i<TG*tsn;i++){
              BOX<<setw(10)<<setfill(' ')<<left<<fixed<<setprecision(3)<<-1*box[i][j];}
                 BOX<<endl;}
BOX<<endl;BOX<<endl;BOX<<endl;

 for(int j=0;j<=leak;j++)
        {  BOX<<setw(6)<<setfill(' ')<<left<<j;
           for(int i=0;i<TG*tsn;i++){
              BOX<<setw(10)<<setfill(' ')<<left<<fixed<<setprecision(3)<<-1*boxleak[i][j];}
                 BOX<<endl;}
BOX<<endl;BOX<<endl;BOX<<endl;

for(int j=0;j<=xpb*TG;j++)
        {  BOX<<setw(6)<<setfill(' ')<<left<<j;
           for(int i=0;i<tsnleak;i++){
              BOX<<setw(10)<<setfill(' ')<<left<<fixed<<setprecision(3)<<-1*boxleakf[i][j];}
                 BOX<<endl;}
BOX<<endl;BOX<<endl;BOX<<endl;
*/


////
```

```
double TG=5;//hocanın isteği üzerine sadece matlab görseli için edğistirildi

//boxf[][] yazdırılıyor

  for(int j=0;j<=xpb*TG;j++)

            {  BOX<<setw(6)<<setfill(' ')<<left<<j;

               for(int i=0;i<Tbigger*tsn+tsnleak;i++){

                   BOX<<setw(10)<<setfill(' ')<<left<<fixed<<setprecision(3)<<-1*boxf[i][j];}

                        BOX<<endl;}
```

/////////////

6.1.10.7 7.7.Preparing boxf[][] Proper To Deformation Calculations

//////////////

```
// boxf[][]'IN transposunun alınası=tboxf[i][j]


for(int i=0;i<=xpb*Tbigger;i++){

  for(int j=0;j<Tbigger*tsn+tsnleak;j++){tboxf[i][j]=boxf[j][i];}}
```

```
// tboxf[i][j] elemanlarını küçükten büyüğe sıralama


for(int i=0;i<=xpb*Tbigger;i++)

{   for(int j=0;j<Tbigger*tsn+tsnleak;j++){Q[j]=tboxf[i][j];}

            for(int k=0;k<Tbigger*tsn+tsnleak;k++){for(int l=0;l<Tbigger*tsn+tsnleak;l++){if(Q[k]<Q[l]){double
swap=Q[k];Q[k]=Q[l];Q[l]=swap;}}}

            for(int q=0;q<Tbigger*tsn+tsnleak;q++) {tboxf[i][q]=Q[q];}}
```

////////////////////////////

6.1.11 *8.Calculations Of Deformations*

////////////////////////////

//////////////

6.1.11.1 8.1.Stiffness Values

//////////////

```
for(int j=0;j<Tbigger*tsn+tsnleak*tsn;j++){for(int i=0;i<=xpb*Tbigger;i++){K[j][i]=k;}}
```

//////////////

6.1.11.2 8.2.Deformations

//////////////

```
/*
8.2.kısım için nomenclature
max=multiplier for Fmax;
min=multiplier for Fmin;
Fmax=Fd*max;
Fmin=Fd*min;
percent=%increment;
inc=force increment=percent*Fd;
*/
	cout<<"input percent increase in load as Fd*%X"<<endl;
	cin>>percent;
	inc=Fd*percent/100;
		cout<<endl;
	cout<<"force_increment= "<<inc<<endl;
	allin<<"inc= "<<inc<<endl;




//defoormasyonların hesaplandığı döngü
temp=0;
int last=(Fmax-Fmin)/inc;
```

```
for(int f=Fmin;f<=Fmax;f=f+inc)
    {for(int i=0;i<=xpb*Tbigger;i++)
      {double sum1=0;
                    double sum2=0;
          for(int j=0;j<Tbigger*tsn+tsnleak;j++){  if(tboxf[i][j]==infinite)break;
                              sum1=sum1+K[j][i];
                                                 sum2=sum2+tboxf[i][j]*K[j][i];
                              y=(f*tsn/bb+sum2)/sum1;
                                                           def[temp][i]=y;
                              if(y<=tboxf[i][j+1])break;}}
            temp++;}


//deformasyonlar yazdırılıyor
        for(int j=0;j<=xpb*TG;j++)
        {DEF<<setw(6)<<setfill(' ')<<left<<j;
          for(int i=0;i<=last;i++)
           {DEF<<setw(10)<<setfill(' ')<<left<<fixed<<setprecision(3)<<-1*def[i][j];}
        DEF<<endl;}
```

//////////////////////////////

6.1.12 9.TEmax_min

//////////////////////////////

```
//TEmax_min-TORQUE curvenin olusturulması için def[last+1][xpb*TG+1]; nın elemanlarının küçükten
büyüğe sıralanması


double Q1[xpb*Tbigger+1];//a temporary array
for(int i=0;i<=last;i++)
{   for(int j=0;j<=xpb*Tbigger;j++){Q1[j]=def[i][j];}
        for(int k=0;k<=xpb*Tbigger;k++){
          for(int l=0;l<xpb*Tbigger+1;l++){
                 if(Q1[k]<Q1[l]){double swap=Q1[k];
```

```
                                        Q1[k]=Q1[l];
                                                                Q1[l]=swap;}}}
        for(int q=0;q<xpb*Tbigger+1;q++){def[i][q]=Q1[q];}
}
```

//elemanları küçükten büyüğe sıralanan def in control için yeniden yazdırılması

```
        allin<<"defswap*********"<<endl;
  allin<<"mesh";for(int i=0;i<=last;i++){allin<<" def["<<left<<setw(3)<<i<<"] ";}allin<<endl;
        for(int j=0;j<xpb*Tbigger+1;j++)
        {allin<<setw(6)<<setfill(' ')<<left<<j;
          for(int i=0;i<=last;i++)
           {       allin<<setw(10)<<setfill(' ')<<left<<fixed<<setprecision(3)<<-1*def[i][j];}allin<<endl;}

        allin<<"def*********"<<endl;
```

//elemanları küçükten büyüğe sıralanan def in ilk(en küçük=minima) ve son elemanının(en büyük=maxima) farkından olusan TEmax_min[last+1] olusturulması

```
        double TEmax_min[last+1];
        for(int i=0;i<=last;i++){TEmax_min[i]=def[i][xpb*Tbigger]-def[i][0];}

        allin<<"TEmax_min*********"<<endl;
  allin<<"force     ";
        for(int i=0;i<1;i++){allin<<" TEmax_min["<<left<<setw(3)<<"] ";}
        allin<<endl;
temp=0;
for(int f=Fmin;f<=Fmax;f=f+inc)
        {
         allin<<setw(10)<<setfill(' ')<<left<<fixed<<setprecision(3)<<f;
         allin<<setw(10)<<setfill('
')<<left<<fixed<<setprecision(3)<<TEmax_min[temp];allin<<endl;temp++;}

        allin<<"TEmax_min*********"<<endl;
//
temp=0;
```

```
for(int f=Fmin;f<=Fmax;f=f+inc)
	{
	 TE<<setw(10)<<setfill(' ')<<left<<fixed<<setprecision(3)<<f;
	 TE<<setw(10)<<setfill(' ')<<left<<fixed<<setprecision(3)<<TEmax_min[temp];TE<<endl;temp++;}
//////////////////////////////
```

6.1.13 __*10.Output Files For MATlab*__

```
//////////////////////////////
nomen<<"xpb "<<xpb<<endl;
nomen<<"xLc "<<xLc<<endl;
nomen<<"tsn "<<tsn<<endl;
nomen<<"xtsn "<<xtsn<<endl;
nomen<<"last "<<last<<endl;
nomen<<"tsnleak= "<<tsnleak<<endl;
nomen<<"Tbigger "<<Tbigger<<endl;
//////////////////////////////
```

6.1.14 __*11.End Of Program*__

```
//////////////////////////////
allin<<endl;
cout<<endl;
allin<<"ONURELDES"<<endl;
cout<<"ONUR_ELDES"<<endl;
//goto top
return 0;
}
```

6.6 C++ NOMENCLATURE FOR SOFTWARE

- top: if cr>=2,user wıll redirected to here, for inputting requested paramaters again. This program ıs designed just for LCR gears.
- ste_out_03_09_18.txt: general output file,you can find all the output files in it.
- nomenclature_03_09_18.txt: nomenclature file.
- a1_1_Li_out_03_09_18.txt: isolated TE curve's datas output file.
- a1_2_LLi_out_03_09_18.txt: Leftside of the from middle point of isolated TE curve's datas output file.
- a1_3_LRi_out_03_09_18.txt: Rightside of the from middle point of isolated TE curve's datas output file.
- a2box_out_03_09_18.txt: contains all the datas of continuous TE curve after addition of PE,PAE,AR,ER, CROWNING,MISALIGNMENT effects.
- a3def_out_03_09_18.txt: continous TE curve which is deformed under each load increments datas output file.
- a4TEmax_min_out_03_09_18.txt: TEpp datas output file.
- a2peTsmaller_out_03_09_18.txt: output file contains datas of pitch errors for pinion.
- a2peTbigger_out_03_09_18.txt: output file contains datas of pitch errors for gear.
- a2pe_out_03_09_18.txt: output file contains datas of pressure error for sum of pinion and gear.
- Tdriven: tooth number of driven gear.
- Tdriver: tooth number of driver gear.
- Tbigger: tooth number of a gear which have bigger value between pinion and gear.
- Tsmaller: tooth number of a gear which have smaller value between pinion and gear.
- smaller[Tsmaller]: A matrix contains pitch errors of smaller tooth gear.
- bigger[Tbigger]: A matrix contains pitch errors of bigger tooth gear.
- PE[Tbigger];the matrıx ınculeds total pe datas for each toothmate at one revolotıon.
- largest: largest pe of pe[i].
- smallest: smallest pe of pe[i].
- m:modüle.
- mn:normal module; mn=m.
- mt:tranverse modüle.
- aopn:normal angle of pessure.
- aopt:transverse angle of pressure.

- helix:helix angle.
- pt:transverse pitch.
- px:axial pitch.
- pn:normal pitch.
- Y:multiplier for addendum.
- X:multiplier for dedendum.
- add:addendum.
- ded:dedendum.
- bb:facewidth.
- pc:circular pitch.
- pb:base pitch.
- Lc:contactlength.
- pcr:profile contact ratio.
- acr:axial contact ratio;//totalCR=pcr+acr.
- rt:tip_radii.
- rb:base_radii.
- rp:pitch_radii.
- CD: center distance .
- DP: diametral pitch.
- totalCR: pcr+acr.
- totalpae1: total pressure angle error 1 //see related codes
- totalpae2: total pressure angle error 2 //see related codes
- Df= form diameter in inches.
- SAP= start of active profile .
- tpa= transverse pressure angle.
- otpa= operating transverse pressure angle.
- paom= pressure angle at OD/ID of gear if mating gear OD/ID is given.
- paSAP= pressure angle at SAP of gear.
- rf: form diameter at SAP for related gear.
- paedn: pressure angle error on driven in degrees.
- paedr: pressure angle error on driver in degrees.
- temprf: temporary value paramates used to find rf.
- RFtemp: pow(x2,2.)+pow(y2,2.);

- nearRF[temprf]: a matrix includes all the RFtemp values.
- temptip:seeked value of RF at tip of tooth.
- foundpae: node number of seeked RF at SAP.
- bottomx2pae=RFx2[foundpae].
- bottomy2pae=RFy2[foundpae].
- PAEtipdriver=(tipx2pae-tipx2)*1000.
- PAESAPdriver=(bottomx2pae-bottomx2)*1000.
- PAEtipdriven=(tipx2pae-tipx2)*1000.
- PAESAPdriven=(bottomx2pae-bottomx2)*1000.
- totalpaeSAPdriver=PAESAPdriver+PAEtipdriven.
- totalpaetipdriver=PAEtipdriver+PAESAPdriven.
- totalpaetipdriven=PAEtipdriven+PAESAPdriver.
- totalpaeSAPdriven=PAESAPdriven+PAEtipdriver.
- Fd=design load.
- k=stiffness.
- arL=leftside amount of relief.
- arR=right side amount of relief.
- suggested ar: Fd/k/bb.
- er1,er2: extend of relief LEFT=1 ve RIGHT=2.
- recommended er: Lc-pb for the TEpp=0.
- delx: distance between nodes which are used to divide tooth profile into meshes. delx=pb/xpb.
- xpb: mesh number.
- xB=Lc/2/delx.
- xer2=er2/delx.
- xer1=er1/delx.
- LLi[i]: Leftside of the from middle point of isolated TE curve’s datas matrix.
- LRi[i]: Rightside of the from middle point of isolated TE curve’s datas matrix.
- mpae=(totalpae2-totalpae1)/xLc.
- bpae=totalpae1.
- PAE[i]=mpae*i+totalpae1.
- L[i]: isolated TE curve’s datas matrix.
- lureleft: isolated TE curve’s leftside possible maximum distance form x-axis.

- lureright: isolated TE curve’s rightside possible maximum distance form x-axis.
- tsn=thin slice number.
- ss:shift of each slice.
- ss/delx=correctness factor.
- xss=node number in ss.
- xtsn=node number after adding thin slices to isolated TE curve . It must be bigger than xLc.
- aomis: angle of misalignment.
- MA[tsn]:misalignments for each thin slice matrix.
- maxcro: maximum crowning value.
- CRO[tsn]:crowning values for each thin slice matrix.
- tsndelx=bb/tsn.
- ac:amount of crowning at face.
- b1: distance as extent of first region.
- xb1=b1/tsndelx.
- b2:distance as extent of 2nd region.
- xb2=tsn-xb1.
- A: y=A*x^B of crowning formula if it is a parabolic curve.
- B: y=A*x^B of crowning formula if it is a parabolic curve.
- exp=1/B.
- AR: biggest one between ar Land arR.
- luremax=abs(largest)+abs(smallest)+MA[tsn-1]+maxcro+1+abs(totalpae1)+abs(totalpae2).
- sTS[i][j]:isolated TE curve after adding thin slices, pıtch error,mısalıgnment,crownıng and pressure angle error.
- leak=xtsn-xpb.
- box[i][k]:continous TE curve for one revolution.
- min:multiplier min for Fmin=Fd*min.
- max:multipliers for Fmax=Fd*max.
- Fmax=Fd*max.
- Fmin=Fd*min.
- infinite=Fmax/k/bb+luremax+AR.
- tsnleak=Tbigger*tsn-tsni.
- boxleak[i][k]: leak section from box.
- boxleakf: final shape of boxleak.

- boxf:final shape of box.
- TG=5. It is tooth mate number as decided for plotting.you can change it if you desire.
- tboxf[i][j]=boxf[j][i].
- K[j][i]=k:stiffness matrix.
- percent=%increment of force.
- inc=force increment=percent*Fd.
- last=(Fmax-Fmin)/inc.
- Q1[xpb*Tbigger+1]:a temporary array used in calculations of TEpp.
- def[last+1][xpb*TG+1]:deformaiton matrix.
- TEmax_min[last+1]:TEpp matrix.

6.7 MATlab SOFTWARE

```
clc;
clear;
clear all;

fid=fopen('nomenclature_03_09_18.txt','r');
nomenc=textscan(fid,'%s%f');
nomenc{1}
b=nomenc{2}
   xpb=b(1)
   xLc=b(2)
   tsn=b(3)
   xtsn=b(4)
   last=b(5)
   tsnleak=b(6)
   TG=b(7)
XPB=0:1:xpb
XLC=0:1:xLc
load a2box_out_03_09_18.txt;
edit a2box_out_03_09_18.txt;
box=a2box_out_03_09_18
```

```
figure
for i=2:tsn*TG+tsnleak+1
plot(box(:,1),box(:,i))
hold on
end
grid on
xlabel('step number')
ylabel('length')
title('box through pb')
hold on
load a3def_out_03_09_18.txt;
edit a3def_out_03_09_18.txt;
def=a3def_out_03_09_18
for i=2:last+2
plot(def(:,1),def(:,i),'LineWidth',2)
hold on
end
grid on
xlabel('step number')
ylabel('deformation')
title(' loading effects')

figure
load a4TEmax_min_out_03_09_18.txt;
edit a4TEmax_min_out_03_09_18.txt;
TE=a4TEmax_min_out_03_09_18
TE(:,1)
TE(:,2)
plot(TE(:,1),TE(:,2))
grid on
xlabel('force increment')
ylabel('TEpp')
title('TEmax-min')
%program sonu
```

7 CASE STUDIES FOR SPUR GEAR

7.1 SUGGESTED AR, ER ;NO PE, PAE, CROWNING, MISALIGNMENT

Table 7.1

DRIVER GEAR TOOTH NUMBER	20
DRIVEN GEAR TOOTH NUMBER	20
PICTH ERROR	NO
NORMAL MODULE	2
NORMAL PRESSURE ANGLE(DEGREE)	20
HELIX ANGLE(DEGREE)	0
ADDENDUM MULTIPLIER(Y*m=add)	1
DEDENDUM MULTIPLIER(X*m=add)	1.25
FACEWITDH(mm)	10
PRESSURE ANGLE ERROR	NO
DESIGN LOAD(N)	1000
STIFFNESS(N/mm/micron)	10
SUGGESTED AR OF LEFTSIDE	YES => 10micron
SUGGESTED AR OF LEFTSIDE	YES => 10micron
SUGGESTED ER OF LEFTSIDE	YES => 3.2877
SUGGESTED ER OF LEFTSIDE	YES => 3.2877
NUMBER OF NODES AT BASE PITCH	128
THIN SLICE NUMBER	1
MISALIGNMENT(DEGREE)	0
CROWNING	NO
MULTIPLIER FOR Fmin=min*Fd	0
MULTIPLIER FOR Fmax=min*Fd	2
% INCREASE IN LOADING	20%

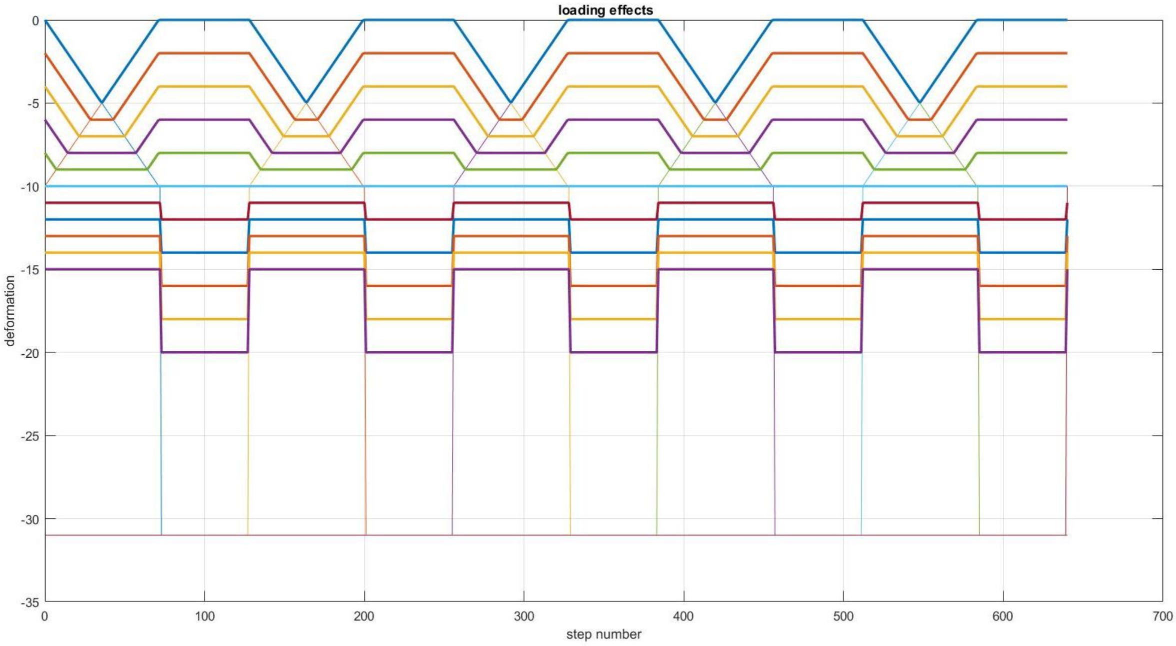


Figure 7.1

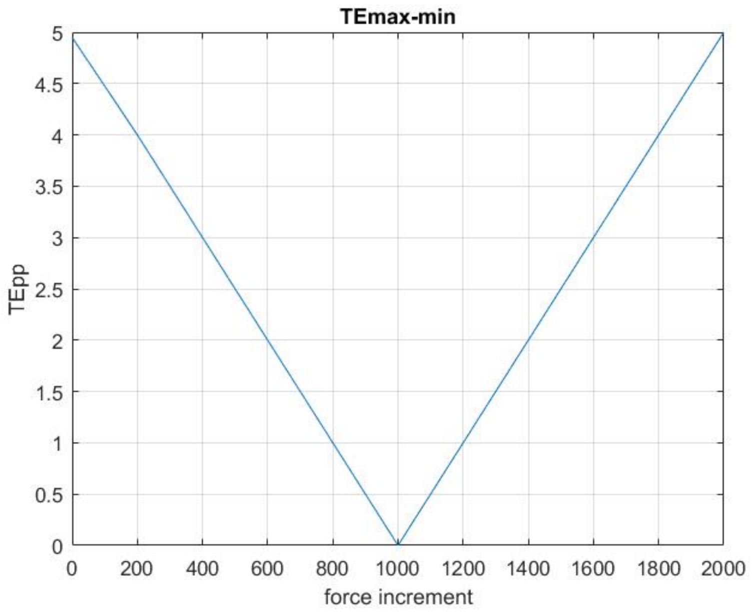


Figure 7.2

7.2 NO AR, ER ;NO PE, PAE, CROWNING, MISALIGNMENT

Table 7.2

DRIVER GEAR TOOTH NUMBER	20
DRIVEN GEAR TOOTH NUMBER	20
PITCH ERROR	NO
NORMAL MODULE	2
NORMAL PRESSURE ANGLE(DEGREE)	20
HELIX ANGLE(DEGREE)	0
ADDENDUM MULTIPLIER(Y*m=add)	1
DEDENDUM MULTIPLIER(X*m=add)	1.25
FACEWITDH(mm)	10
PRESSURE ANGLE ERROR	NO
DESIGN LOAD(N)	1000
STIFFNESS(N/mm/micron)	10
SUGGESTED AR OF LEFTSIDE	NO =>0 micron
SUGGESTED AR OF LEFTSIDE	NO =>0 micron
SUGGESTED ER OF LEFTSIDE	NO =>0 micron
SUGGESTED ER OF LEFTSIDE	NO =>0 micron
NUMBER OF NODES AT BASE PITCH	128
THIN SLICE NUMBER	1
MISALIGNMENT(DEGREE)	0
CROWNING	NO
MULTIPLIER FOR Fmin=min*Fd	0
MULTIPLIER FOR Fmax=min*Fd	2
% INCREASE IN LOADING	20%

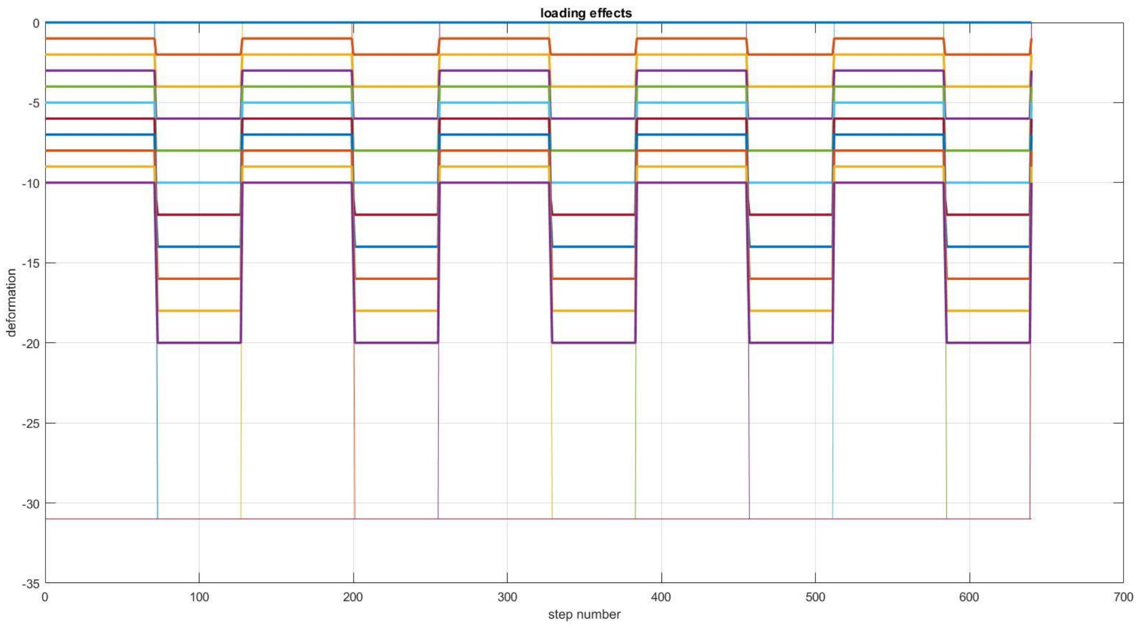


Figure 7.3

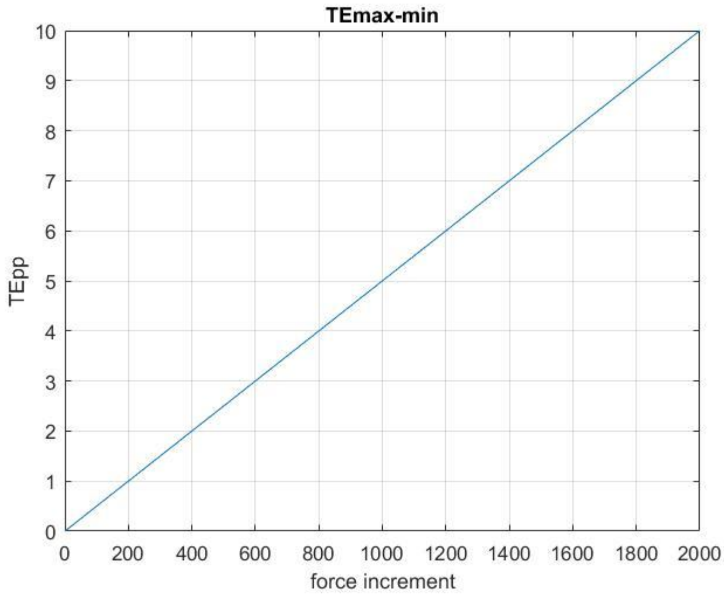


Figure7.4

7.3 RANDOM AR, ER;NO PE, PAE, CROWNING, MISALIGNMENT

Table 7.3

DRIVER GEAR TOOTH NUMBER	20
DRIVEN GEAR TOOTH NUMBER	20
PITCH ERROR	NO
NORMAL MODULE	2
NORMAL PRESSURE ANGLE(DEGREE)	20
HELIX ANGLE(DEGREE)	0
ADDENDUM MULTIPLIER(Y*m=add)	1
DEDENDUM MULTIPLIER(X*m=add)	1.25
FACEWITDH(mm)	10
PRESSURE ANGLE ERROR	NO
DESIGN LOAD(N)	1000
STIFFNESS(N/mm/micron)	10
SUGGESTED AR OF LEFTSIDE	NO =>5 micron
SUGGESTED AR OF LEFTSIDE	NO =>8 micron
SUGGESTED ER OF LEFTSIDE	NO =>1.75 micron
SUGGESTED ER OF LEFTSIDE	NO =>3.5 micron
NUMBER OF NODES AT BASE PITCH	128
THIN SLICE NUMBER	1
MISALIGNMENT(DEGREE)	0
CROWNING	NO
MULTIPLIER FOR Fmin=min*Fd	0
MULTIPLIER FOR Fmax=min*Fd	2
% INCREASE IN LOADING	20%

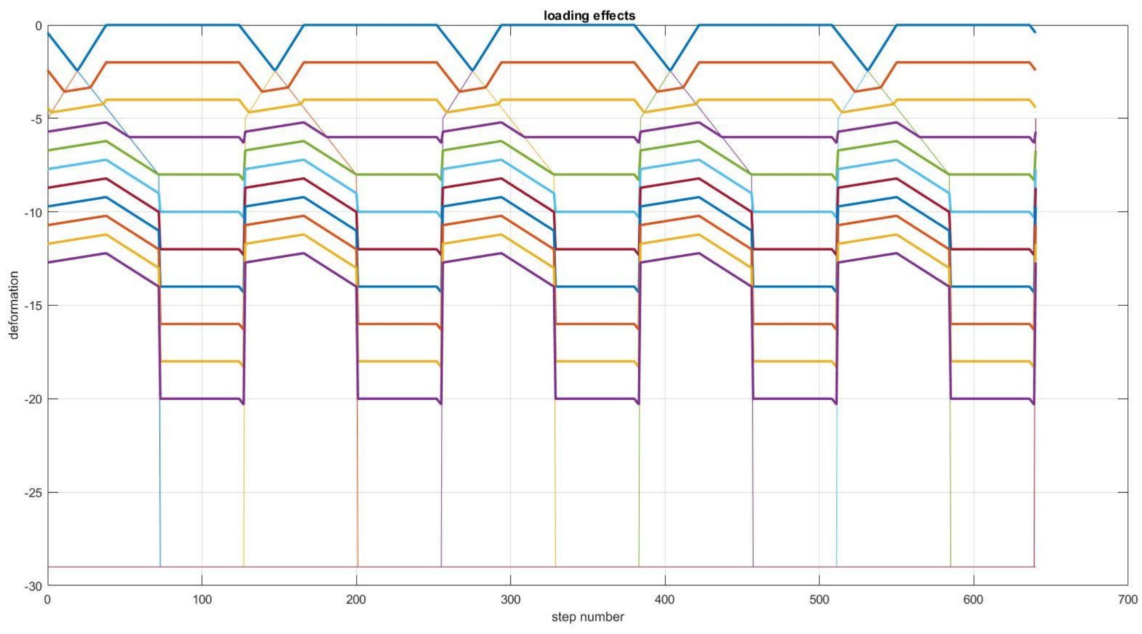


Figure 7.5

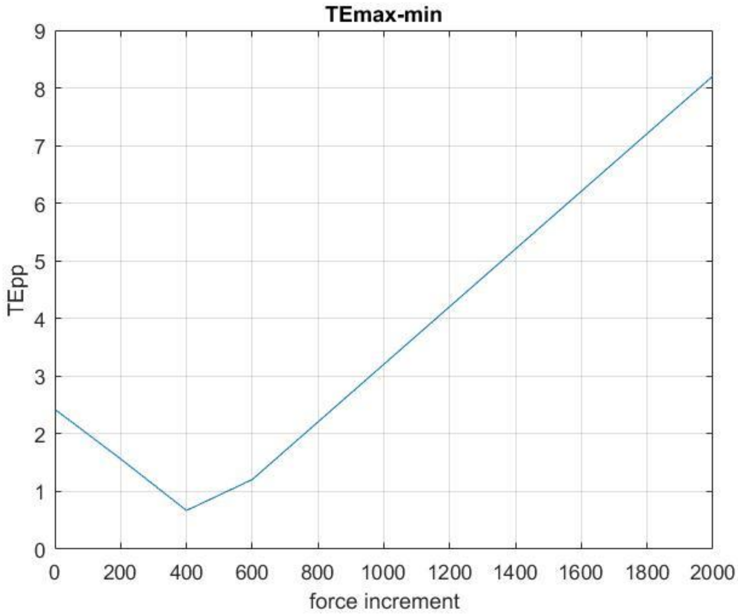


Figure 7.6

7.4 SUGGESTED AR, ER; RANDOM PE;NO PAE, CROWNING, MISALIGNMENT

Table 7.4

DRIVER GEAR TOOTH NUMBER	20
DRIVEN GEAR TOOTH NUMBER	20
PITCH ERROR	YES
NORMAL MODULE	2
NORMAL PRESSURE ANGLE(DEGREE)	20
HELIX ANGLE(DEGREE)	0
ADDENDUM MULTIPLIER(Y*m=add)	1
DEDENDUM MULTIPLIER(X*m=add)	1.25
FACEWITDH(mm)	10
PRESSURE ANGLE ERROR	NO
DESIGN LOAD(N)	1000
STIFFNESS(N/mm/micron)	10
SUGGESTED AR OF LEFTSIDE	YES => 10micron
SUGGESTED AR OF LEFTSIDE	YES => 10micron
SUGGESTED ER OF LEFTSIDE	YES => 3.2877
SUGGESTED ER OF LEFTSIDE	YES => 3.2877
NUMBER OF NODES AT BASE PITCH	128
THIN SLICE NUMBER	1
MISALIGNMENT(DEGREE)	0
CROWNING	NO
MULTIPLIER FOR Fmin=min*Fd	0
MULTIPLIER FOR Fmax=min*Fd	2
% INCREASE IN LOADING	20%

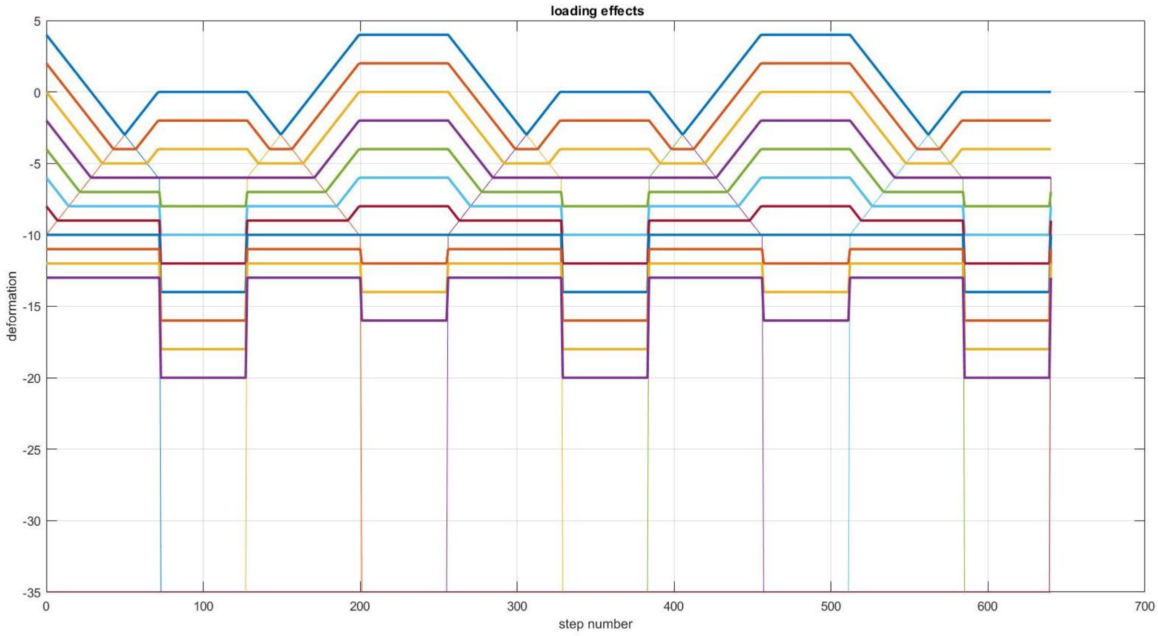


Figure 7.7

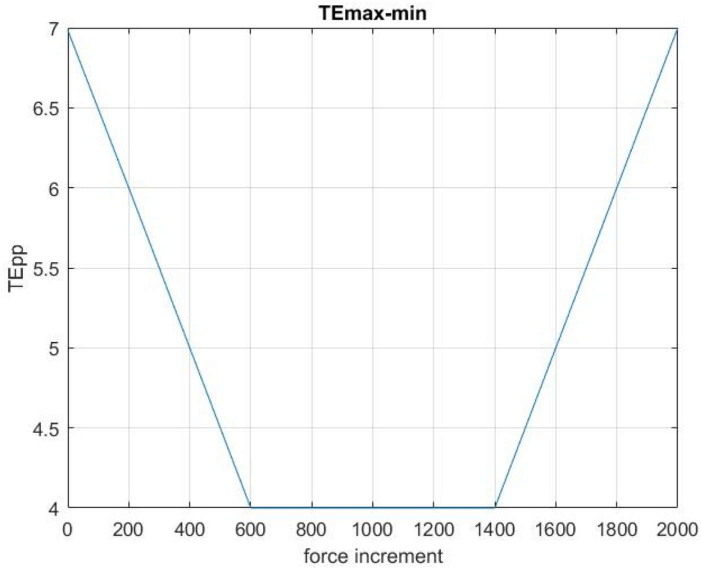


Figure 7.8

7.5 SUGGESTED AR, ER; NO PE;INPUTTED PAE; NO CROWNING, MISALIGNMENT

Table 7.5

DRIVER GEAR TOOTH NUMBER	20
DRIVEN GEAR TOOTH NUMBER	20
PITCH ERROR	NO
NORMAL MODULE	2
NORMAL PRESSURE ANGLE(DEGREE)	20
HELIX ANGLE(DEGREE)	0
ADDENDUM MULTIPLIER(Y*m=add)	1
DEDENDUM MULTIPLIER(X*m=add)	1.25
FACEWITDH(mm)	10
PRESSURE ANGLE ERROR	YES
PAE OF DRIVEN GEAR(DEGREE)	0.05
PAE OF DRIVER GEAR(DEGREE)	0
#OF MESHFOR INVOLUTE CURVE	100
DESIGN LOAD(N)	1000
STIFFNESS(N/mm/micron)	10
SUGGESTED AR OF LEFTSIDE	YES => 10micron
SUGGESTED AR OF LEFTSIDE	YES => 10micron
SUGGESTED ER OF LEFTSIDE	YES => 3.2877
SUGGESTED ER OF LEFTSIDE	YES => 3.2877
NUMBER OF NODES AT BASE PITCH	128
THIN SLICE NUMBER	10
MISALIGNMENT(DEGREE)	0
CROWNING	NO
MULTIPLIER FOR Fmin=min*Fd	0
MULTIPLIER FOR Fmax=min*Fd	2
% INCREASE IN LOADING	20%

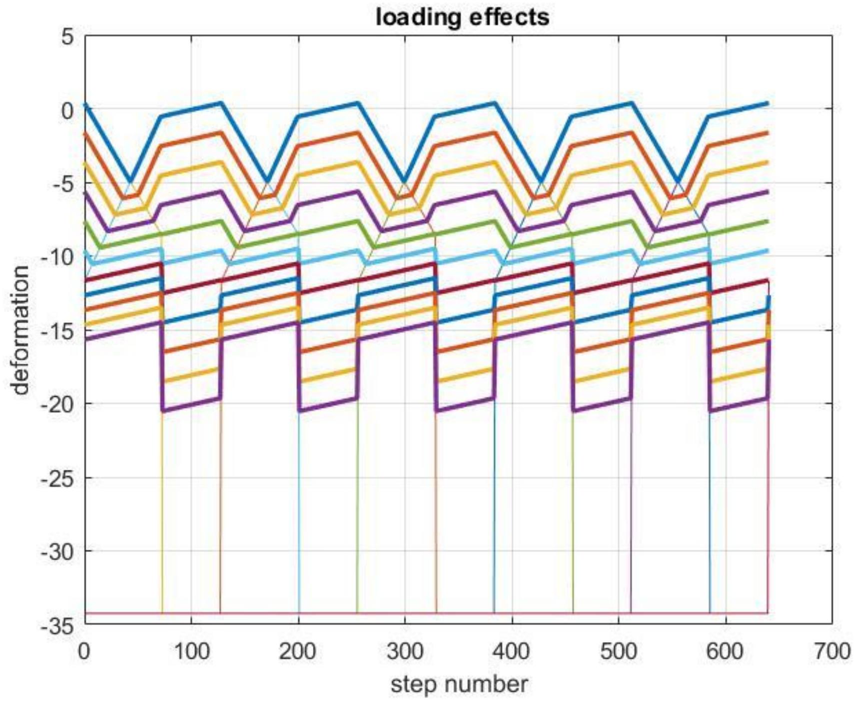


Figure 7.9

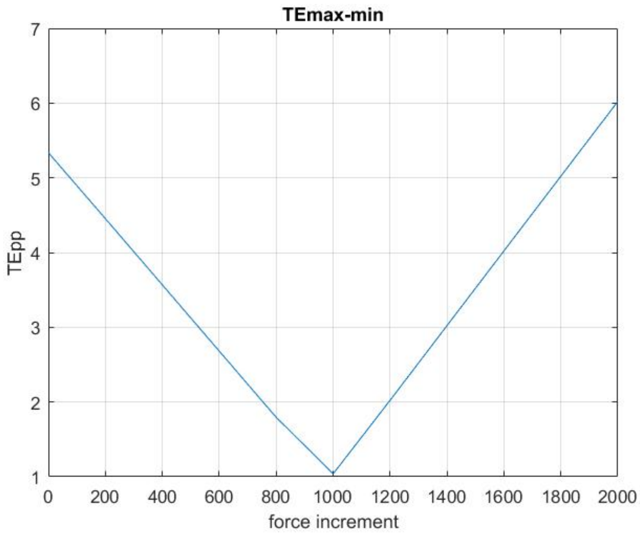


Figure 7.10

7.6 SUGGESTED AR, ER; NO PE; NO PAE; INPUTTED LINEAR CROWNING; NO MISALIGNMENT

Table 7.6

DRIVER GEAR TOOTH NUMBER	20
DRIVEN GEAR TOOTH NUMBER	20
PITCH ERROR	NO
NORMAL MODULE	2
NORMAL PRESSURE ANGLE(DEGREE)	20
HELIX ANGLE(DEGREE)	0
ADDENDUM MULTIPLIER(Y*m=add)	1
DEDENDUM MULTIPLIER(X*m=add)	1.25
FACEWITDH(mm)	10
PRESSURE ANGLE ERROR	NO
DESIGN LOAD(N)	1000
STIFFNESS(N/mm/micron)	10
SUGGESTED AR OF LEFTSIDE	YES => 10microns
SUGGESTED AR OF LEFTSIDE	YES => 10microns
SUGGESTED ER OF LEFTSIDE	YES => 3.2877
SUGGESTED ER OF LEFTSIDE	YES => 3.2877
NUMBER OF NODES AT BASE PITCH	128
THIN SLICE NUMBER	10
MISALIGNMENT(DEGREE)	0
CROWNING	YES
TYPE OF CROWNING	LINEAR
AMOUNT OF CROWNING AT FACE	14 microns
EXTENT OF FIRST REGION	4microns
MULTIPLIER FOR Fmin=min*Fd	0
MULTIPLIER FOR Fmax=min*Fd	2
% INCREASE IN LOADING	20%

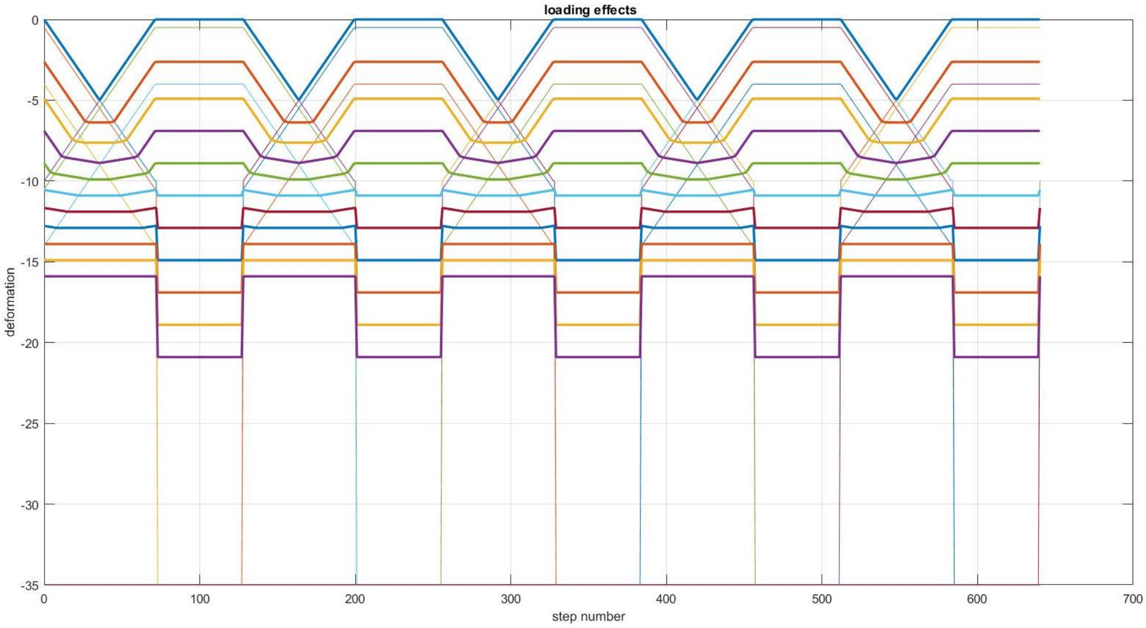


Figure 7.11

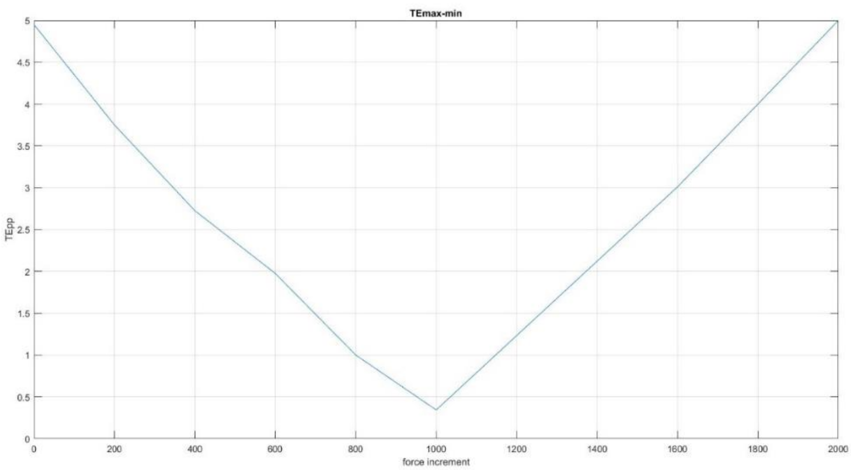


Figure 7.12

7.7 SUGGESTED AR, ER; NO PE; NO PAE; INPUTTED PARABOLIC CROWNING; NO MISALIGNMENT

Table 7.7

DRIVER GEAR TOOTH NUMBER	20
DRIVEN GEAR TOOTH NUMBER	20
PITCH ERROR	NO
NORMAL MODULE	2
NORMAL PRESSURE ANGLE(DEGREE)	20
HELIX ANGLE(DEGREE)	0
ADDENDUM MULTIPLIER(Y*m=add)	1
DEDENDUM MULTIPLIER(X*m=add)	1.25
FACEWITDH(mm)	10
PRESSURE ANGLE ERROR	NO
DESIGN LOAD(N)	1000
STIFFNESS(N/mm/micron)	10
SUGGESTED AR OF LEFTSIDE	YES => 10microns
SUGGESTED AR OF LEFTSIDE	YES => 10microns
SUGGESTED ER OF LEFTSIDE	YES => 3.2877
SUGGESTED ER OF LEFTSIDE	YES => 3.2877
NUMBER OF NODES AT BASE PITCH	128
THIN SLICE NUMBER	10
MISALIGNMENT(DEGREE)	0
CROWNING	YES
TYPE OF CROWNING	PARABOLIC
AMOUNT OF CROWNING AT FACE	14 microns
Y=A*X^B=> A=?	1
Y=A*X^B=> B=?	3
MULTIPLIER FOR Fmin=min*Fd	0
MULTIPLIER FOR Fmax=min*Fd	2
% INCREASE IN LOADING	20%

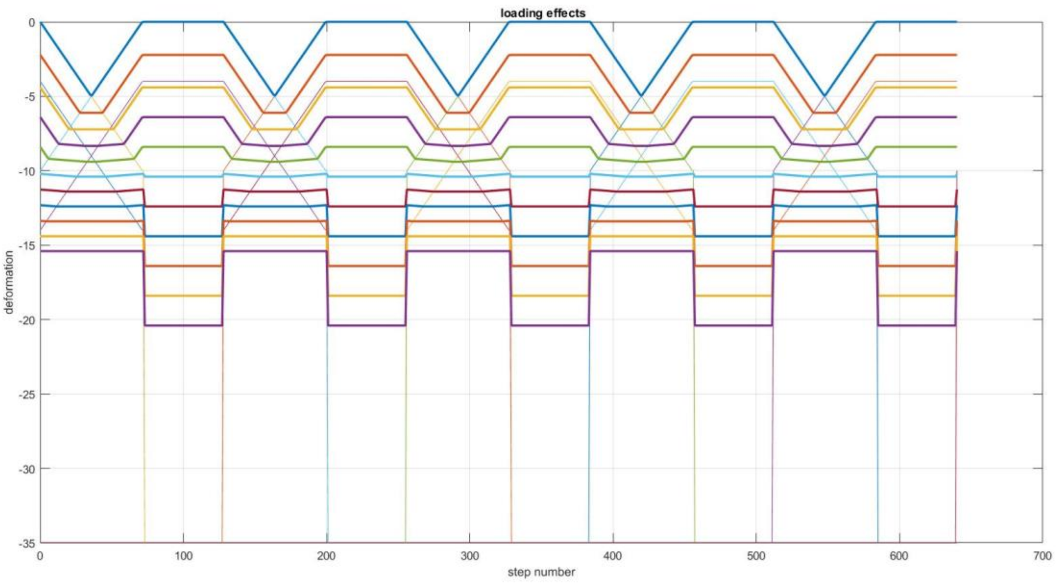


Figure 7.13

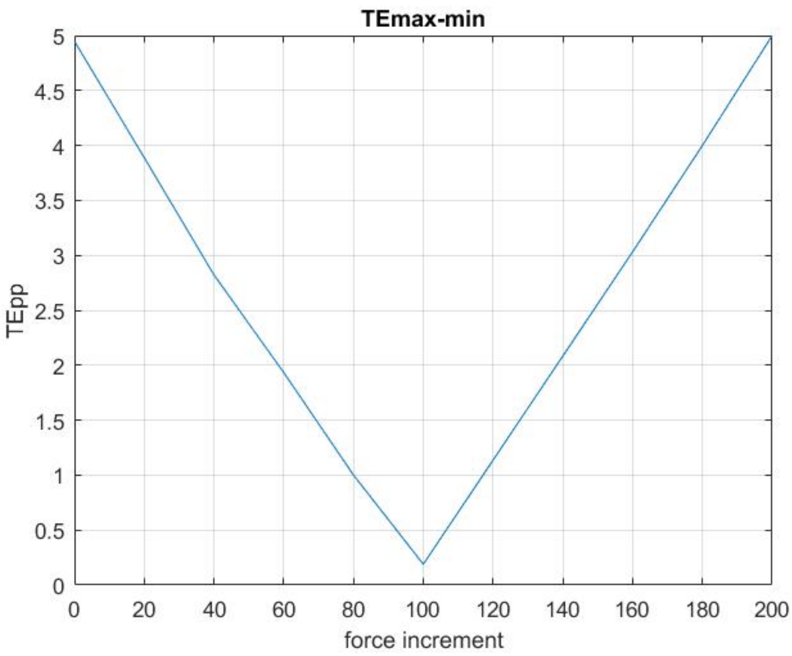


Figure 7.14

7.8 SUGGESTED AR, ER; NO PE; NO PAE, CROWNING; INPUTTED MISALIGNMENT

Table 7.8

DRIVER GEAR TOOTH NUMBER	20
DRIVEN GEAR TOOTH NUMBER	20
PITCH ERROR	NO
NORMAL MODULE	2
NORMAL PRESSURE ANGLE(DEGREE)	20
HELIX ANGLE(DEGREE)	0
ADDENDUM MULTIPLIER(Y*m=add)	1
DEDENDUM MULTIPLIER(X*m=add)	1.25
FACEWITDH(mm)	10
PRESSURE ANGLE ERROR	NO
DESIGN LOAD(N)	1000
STIFFNESS(N/mm/micron)	10
SUGGESTED AR OF LEFTSIDE	YES => 10microns
SUGGESTED AR OF LEFTSIDE	YES => 10microns
SUGGESTED ER OF LEFTSIDE	YES => 3.2877
SUGGESTED ER OF LEFTSIDE	YES => 3.2877
NUMBER OF NODES AT BASE PITCH	128
THIN SLICE NUMBER	10
MISALIGNMENT(DEGREE)	10
CROWNING	NO
MULTIPLIER FOR Fmin=min*Fd	0
MULTIPLIER FOR Fmax=min*Fd	2
% INCREASE IN LOADING	20%

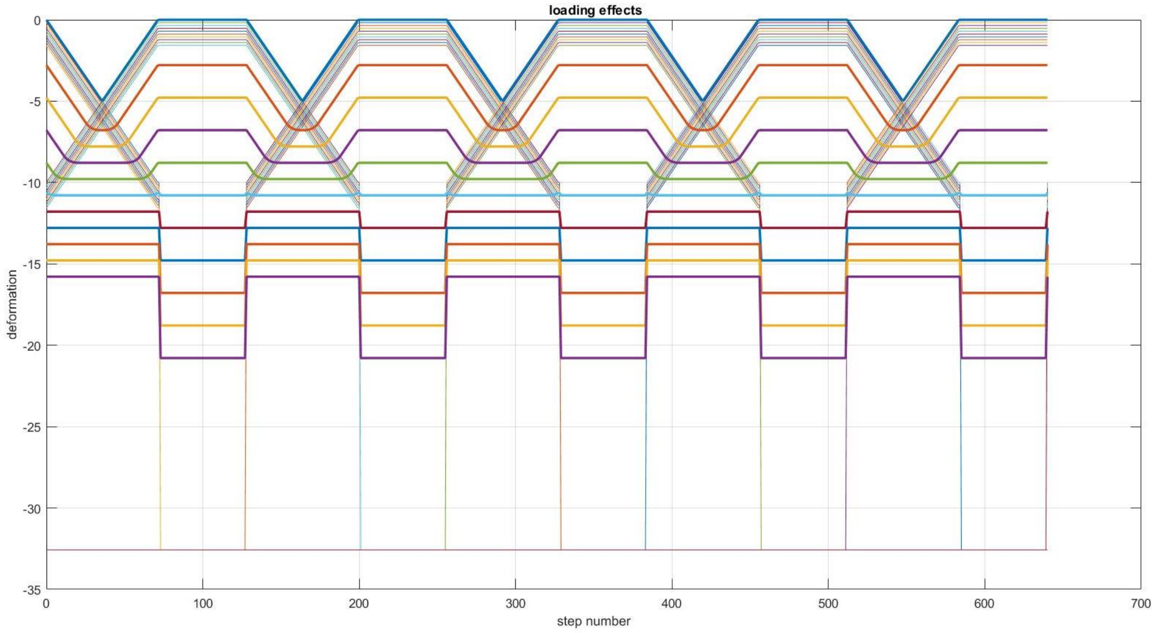


Figure 7.15

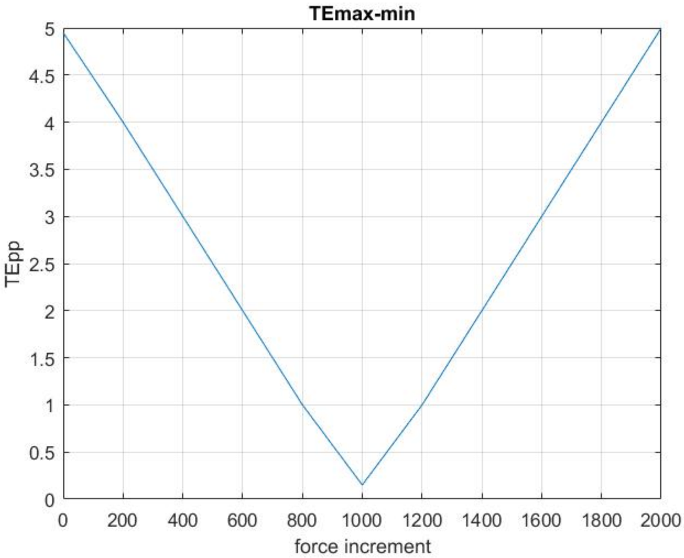


Figure 7.16

7.9 RANDOM AR, ER; INPUTTED PE,PAE, CROWNING, MISALIGNMENT

Table 7.9

DRIVER GEAR TOOTH NUMBER	20
DRIVEN GEAR TOOTH NUMBER	20
PITCH ERROR	YES
NORMAL MODULE	2
NORMAL PRESSURE ANGLE(DEGREE)	20
HELIX ANGLE(DEGREE)	0
ADDENDUM MULTIPLIER(Y*m=add)	1
DEDENDUM MULTIPLIER(X*m=add)	1.25
FACEWITDH(mm)	10
PRESSURE ANGLE ERROR	YES
PAE OF DRIVEN GEAR(DEGREE)	0.05
PAE OF DRIVER GEAR(DEGREE)	0
#OF MESHFOR INVOLUTE CURVE	100
DESIGN LOAD(N)	1000
STIFFNESS(N/mm/micron)	10
SUGGESTED AR OF LEFTSIDE	NO =>5 micron
SUGGESTED AR OF LEFTSIDE	NO =>8 micron
SUGGESTED ER OF LEFTSIDE	NO =>1.75 micron
SUGGESTED ER OF LEFTSIDE	NO =>3.5 micron
NUMBER OF NODES AT BASE PITCH	128
THIN SLICE NUMBER	10
MISALIGNMENT(DEGREE)	10
CROWNING	YES
TYPE OF CROWNING	LINEAR
AMOUNT OF CROWNING AT FACE	14 microns
EXTENT OF FIRST REGION	4microns
MULTIPLIER FOR Fmin=min*Fd	0
MULTIPLIER FOR Fmax=min*Fd	2
% INCREASE IN LOADING	20%

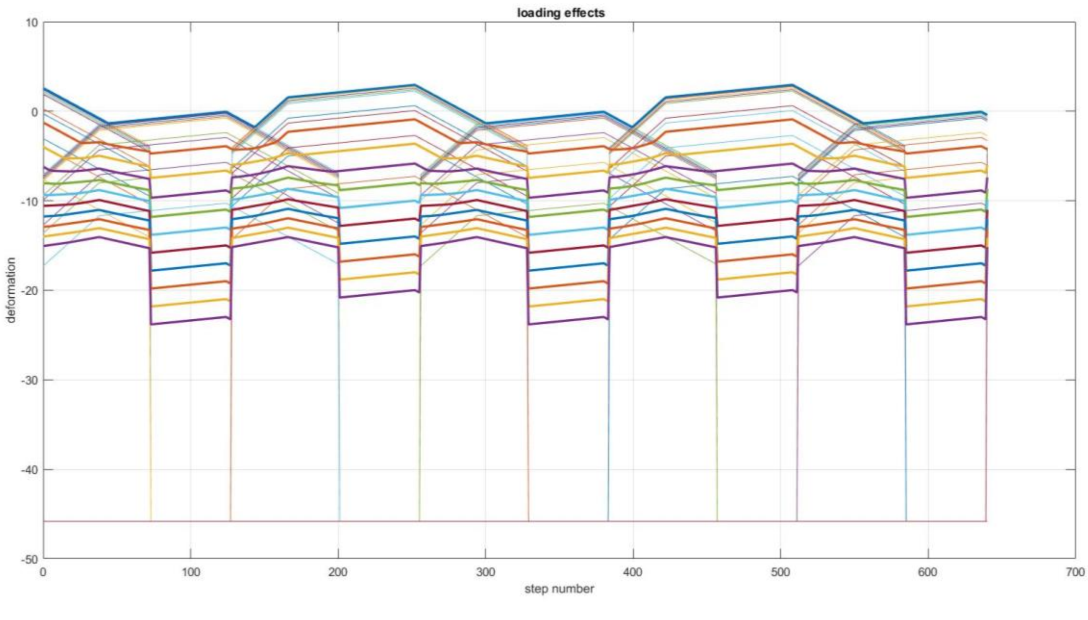


Figure 7.17

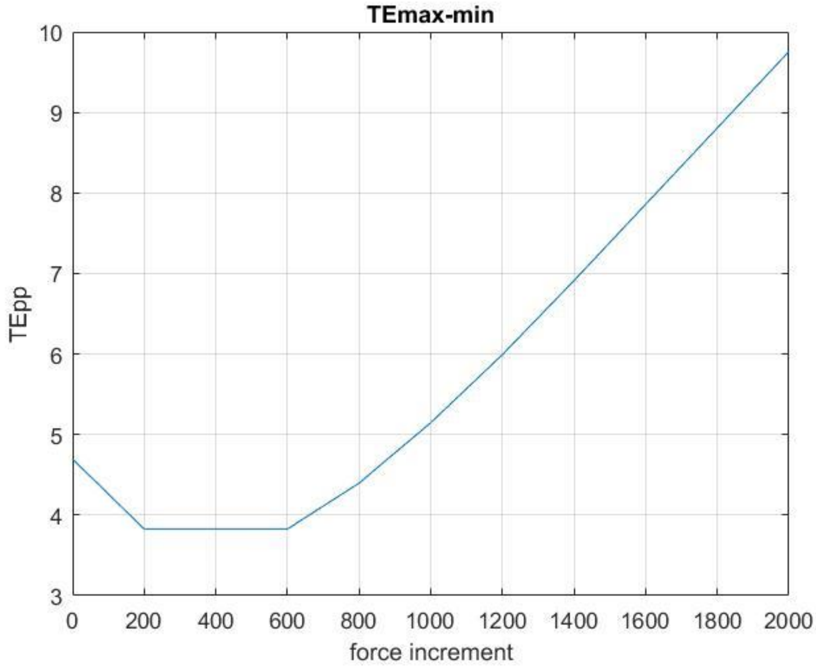


Figure 7.18

8 CASE STUDIES FOR HELICAL GEAR

8.1 SUGGESTED AR, ER ;NO PE, PAE, CROWNING, MISALIGNMENT

Table 8.1

DRIVER GEAR TOOTH NUMBER	20
DRIVEN GEAR TOOTH NUMBER	20
PICTH ERROR	NO
NORMAL MODULE	2
NORMAL PRESSURE ANGLE(DEGREE)	20
HELIX ANGLE(DEGREE)	14
ADDENDUM MULTIPLIER(Y*m=add)	1
DEDENDUM MULTIPLIER(X*m=add)	1.25
FACEWITDH(mm)	10
PRESSURE ANGLE ERROR	NO
DESIGN LOAD(N)	1000
STIFFNESS(N/mm/micron)	10
SUGGESTED AR OF LEFTSIDE	YES => 10micron
SUGGESTED AR OF LEFTSIDE	YES => 10micron
SUGGESTED ER OF LEFTSIDE	YES => 3.2877
SUGGESTED ER OF LEFTSIDE	YES => 3.2877
NUMBER OF NODES AT BASE PITCH	128
THIN SLICE NUMBER	10
MISALIGNMENT(DEGREE)	0
CROWNING	NO
MULTIPLIER FOR Fmin=min*Fd	0
MULTIPLIER FOR Fmax=min*Fd	2
% INCREASE IN LOADING	20%

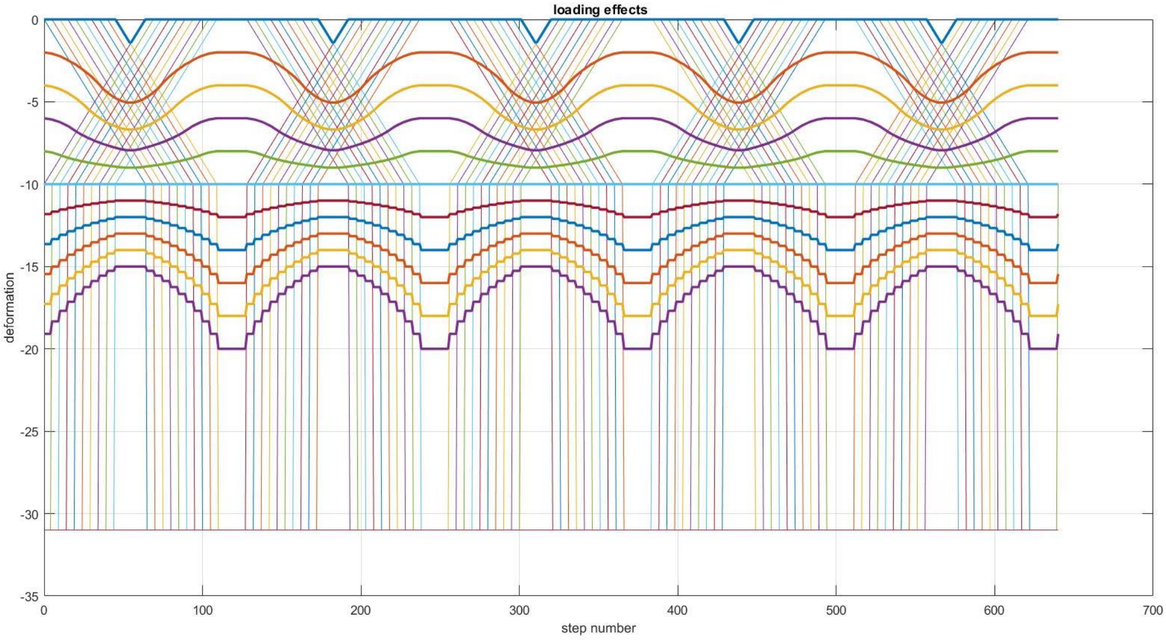


Figure 8.1

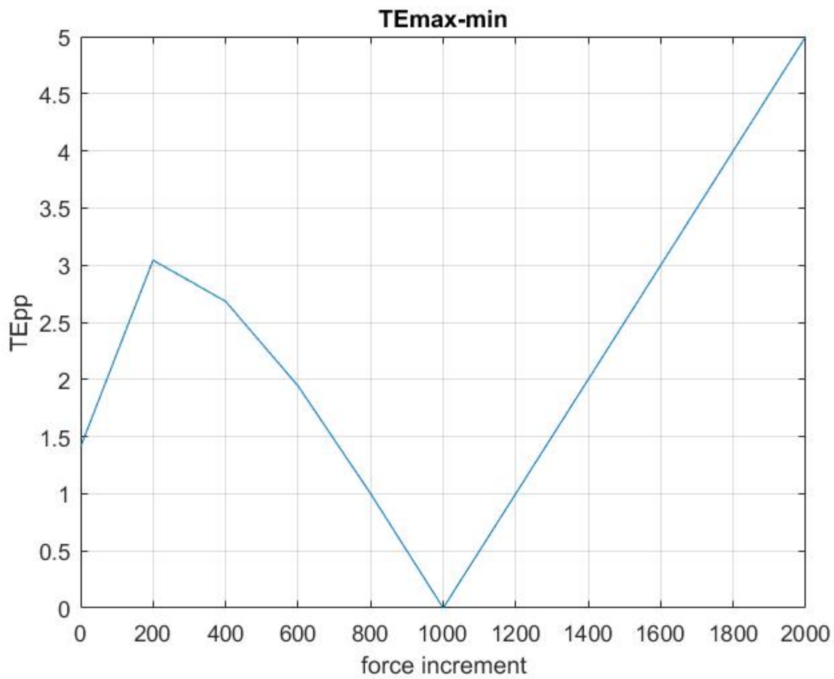


Figure 8.2

8.2 RANDOM AR, ER;NO PE, PAE, CROWNING, MISALIGNMENT

Table 8.2

DRIVER GEAR TOOTH NUMBER	20
DRIVEN GEAR TOOTH NUMBER	20
PITCH ERROR	NO
NORMAL MODULE	2
NORMAL PRESSURE ANGLE(DEGREE)	20
HELIX ANGLE(DEGREE)	14
ADDENDUM MULTIPLIER(Y*m=add)	1
DEDENDUM MULTIPLIER(X*m=add)	1.25
FACEWITDH(mm)	10
PRESSURE ANGLE ERROR	NO
DESIGN LOAD(N)	1000
STIFFNESS(N/mm/micron)	10
SUGGESTED AR OF LEFTSIDE	NO =>5 micron
SUGGESTED AR OF LEFTSIDE	NO =>8 micron
SUGGESTED ER OF LEFTSIDE	NO =>1.75 micron
SUGGESTED ER OF LEFTSIDE	NO =>3.5 micron
NUMBER OF NODES AT BASE PITCH	128
THIN SLICE NUMBER	1
MISALIGNMENT(DEGREE)	0
CROWNING	NO
MULTIPLIER FOR Fmin=min*Fd	0
MULTIPLIER FOR Fmax=min*Fd	2
% INCREASE IN LOADING	20%

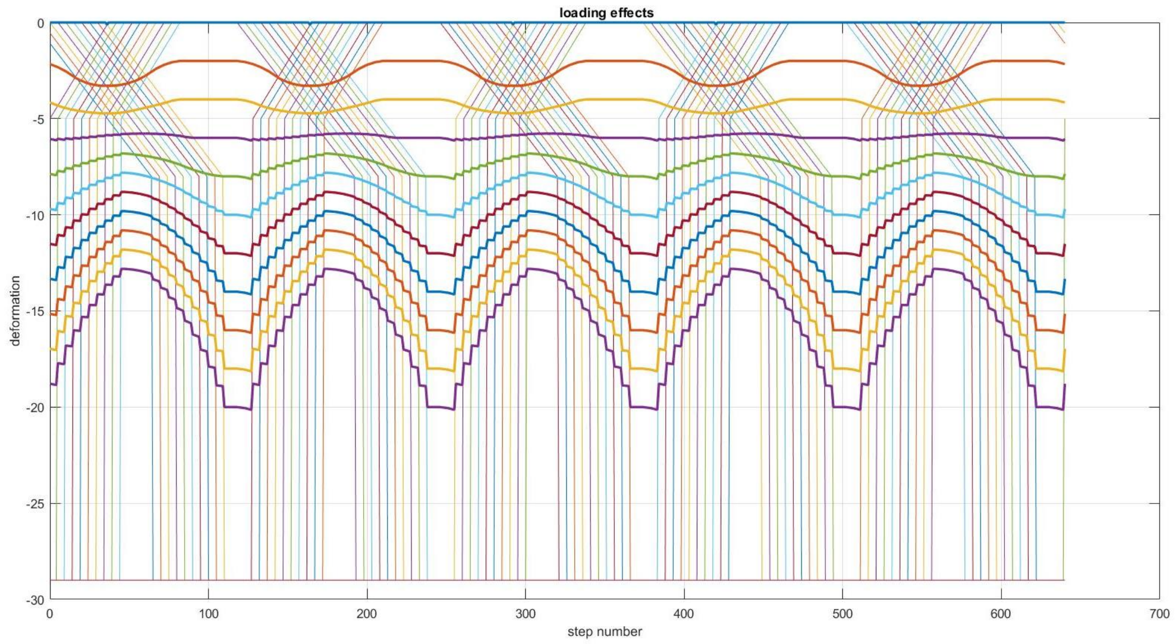


Figure 8.3

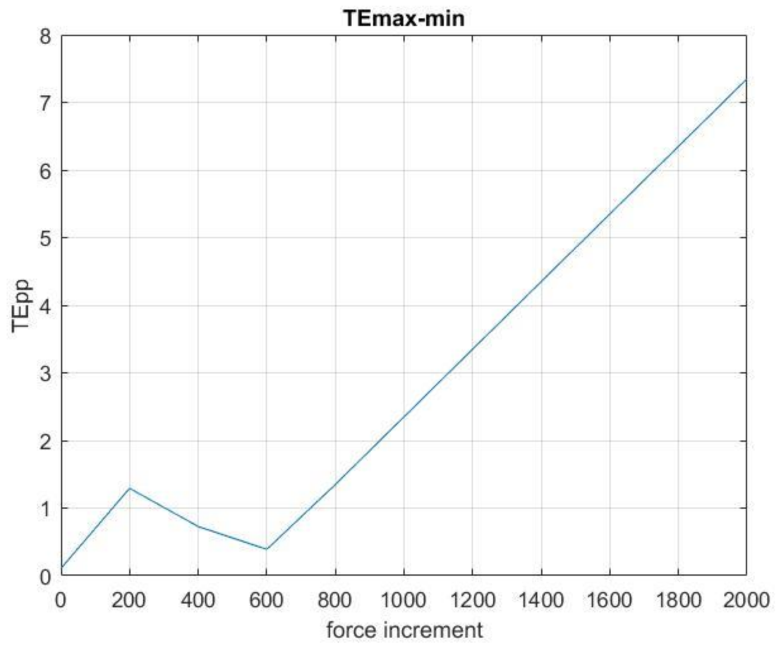


Figure 8.4

8.3 SUGGESTED AR, ER; RANDOM PE;NO PAE, CROWNING, MISALIGNMENT

Table 8.3

DRIVER GEAR TOOTH NUMBER	20
DRIVEN GEAR TOOTH NUMBER	20
PITCH ERROR	YES
NORMAL MODULE	2
NORMAL PRESSURE ANGLE(DEGREE)	20
HELIX ANGLE(DEGREE)	14
ADDENDUM MULTIPLIER(Y*m=add)	1
DEDENDUM MULTIPLIER(X*m=add)	1.25
FACEWITDH(mm)	10
PRESSURE ANGLE ERROR	NO
DESIGN LOAD(N)	1000
STIFFNESS(N/mm/micron)	10
SUGGESTED AR OF LEFTSIDE	YES => 10micron
SUGGESTED AR OF LEFTSIDE	YES => 10micron
SUGGESTED ER OF LEFTSIDE	YES => 3.2877
SUGGESTED ER OF LEFTSIDE	YES => 3.2877
NUMBER OF NODES AT BASE PITCH	128
THIN SLICE NUMBER	1
MISALIGNMENT(DEGREE)	0
CROWNING	NO
MULTIPLIER FOR Fmin=min*Fd	0
MULTIPLIER FOR Fmax=min*Fd	2
% INCREASE IN LOADING	20%

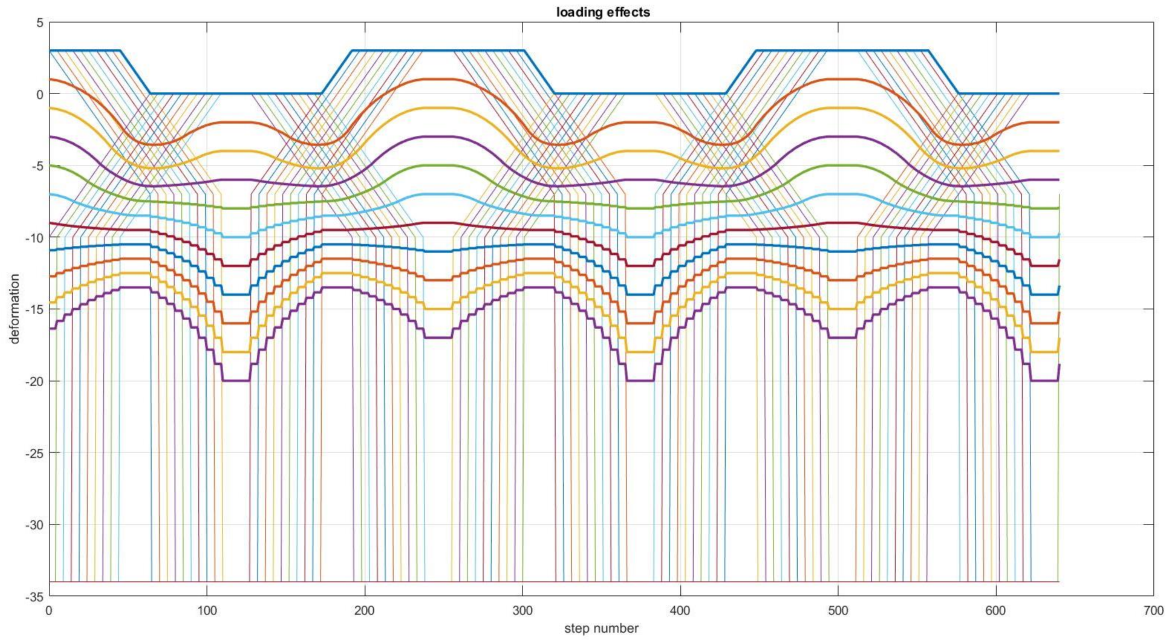


Figure 8.5

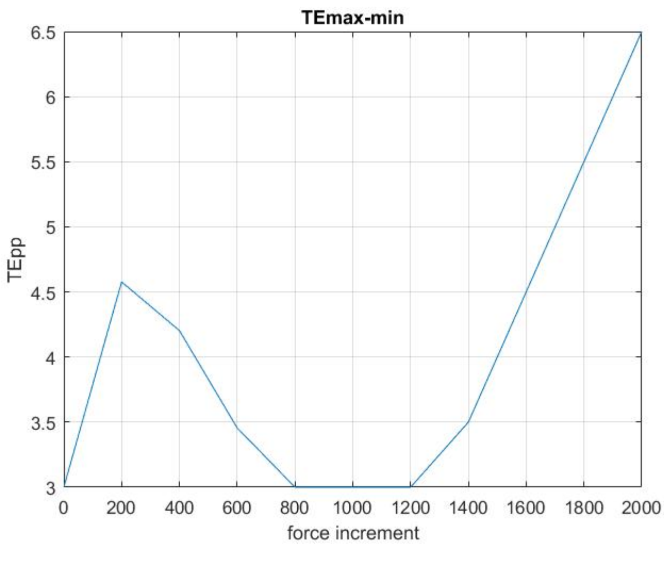


Figure 8.6

8.4 HELICAL GEAR SUGGESTED AR, ER; NO PE;INPUTTED PAE; NO CROWNING, MISALIGNMENT

Table 8.4

DRIVER GEAR TOOTH NUMBER	20
DRIVEN GEAR TOOTH NUMBER	20
PITCH ERROR	NO
NORMAL MODULE	2
NORMAL PRESSURE ANGLE(DEGREE)	20
HELIX ANGLE(DEGREE)	14
ADDENDUM MULTIPLIER(Y*m=add)	1
DEDENDUM MULTIPLIER(X*m=add)	1.25
FACEWITDH(mm)	10
PRESSURE ANGLE ERROR	YES
PAE OF DRIVEN GEAR(DEGREE)	0.05
PAE OF DRIVER GEAR(DEGREE)	0
#OF MESHFOR INVOLUTE CURVE	100
DESIGN LOAD(N)	1000
STIFFNESS(N/mm/micron)	10
SUGGESTED AR OF LEFTSIDE	YES => 10micron
SUGGESTED AR OF LEFTSIDE	YES => 10micron
SUGGESTED ER OF LEFTSIDE	YES => 3.2877
SUGGESTED ER OF LEFTSIDE	YES => 3.2877
NUMBER OF NODES AT BASE PITCH	128
THIN SLICE NUMBER	10
MISALIGNMENT(DEGREE)	0
CROWNING	NO
MULTIPLIER FOR Fmin=min*Fd	0
MULTIPLIER FOR Fmax=min*Fd	2
% INCREASE IN LOADING	20%

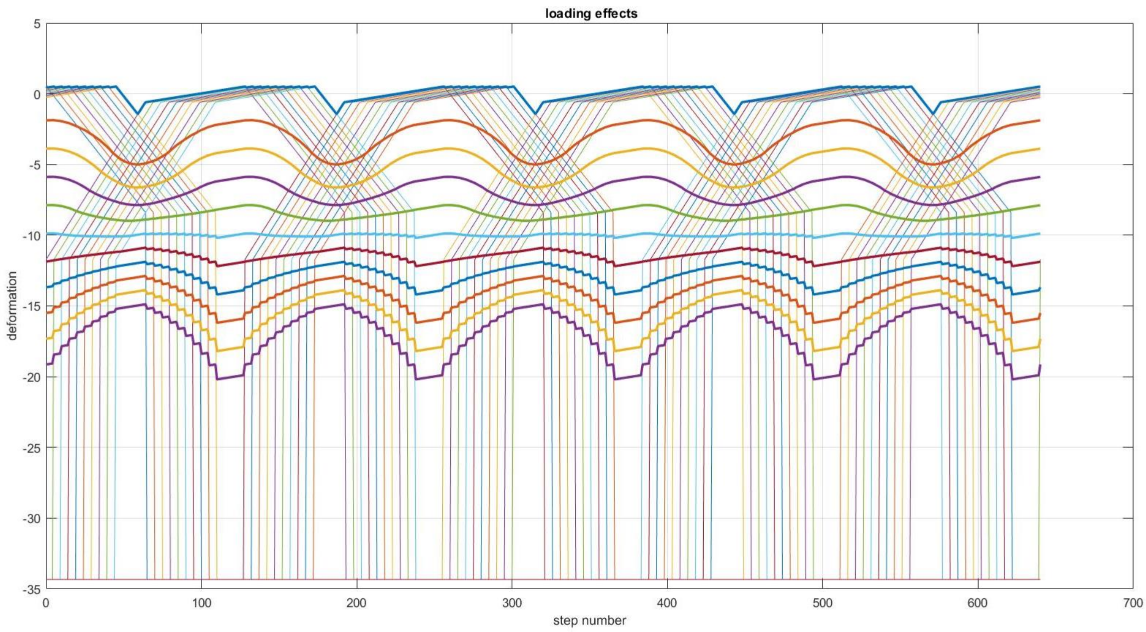


Figure 8.7

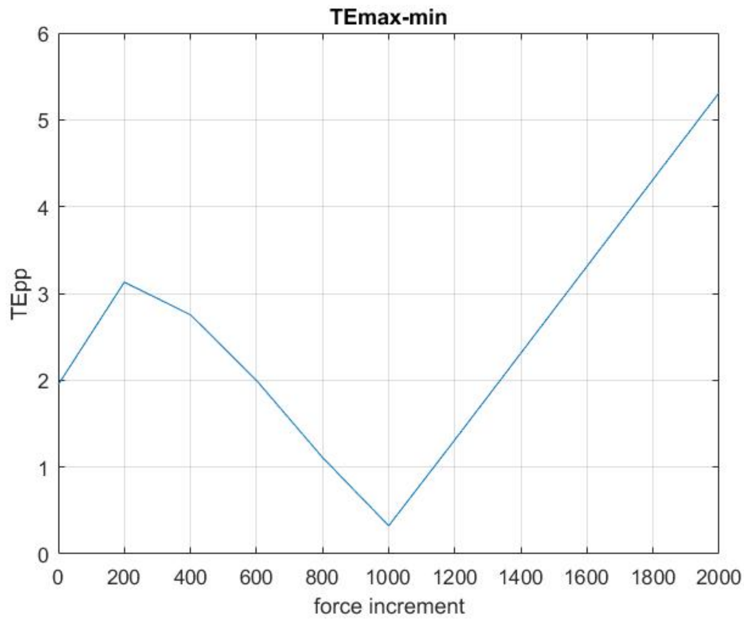


Figure 8.8

8.5 HELICAL GEAR SUGGESTED AR, ER; NO PE; NO PAE; INPUTTED LINEAR CROWNING; NO MISALIGNMENT

Table 8.5

DRIVER GEAR TOOTH NUMBER	20
DRIVEN GEAR TOOTH NUMBER	20
PITCH ERROR	NO
NORMAL MODULE	2
NORMAL PRESSURE ANGLE(DEGREE)	20
HELIX ANGLE(DEGREE)	14
ADDENDUM MULTIPLIER(Y*m=add)	1
DEDENDUM MULTIPLIER(X*m=add)	1.25
FACEWITDH(mm)	10
PRESSURE ANGLE ERROR	NO
DESIGN LOAD(N)	1000
STIFFNESS(N/mm/micron)	10
SUGGESTED AR OF LEFTSIDE	YES => 10microns
SUGGESTED AR OF LEFTSIDE	YES => 10microns
SUGGESTED ER OF LEFTSIDE	YES => 3.2877
SUGGESTED ER OF LEFTSIDE	YES => 3.2877
NUMBER OF NODES AT BASE PITCH	128
THIN SLICE NUMBER	10
MISALIGNMENT(DEGREE)	0
CROWNING	YES
TYPE OF CROWNING	LINEAR
AMOUNT OF CROWNING AT FACE	14 microns
EXTENT OF FIRST REGION	4microns
MULTIPLIER FOR Fmin=min*Fd	0
MULTIPLIER FOR Fmax=min*Fd	2
% INCREASE IN LOADING	20%

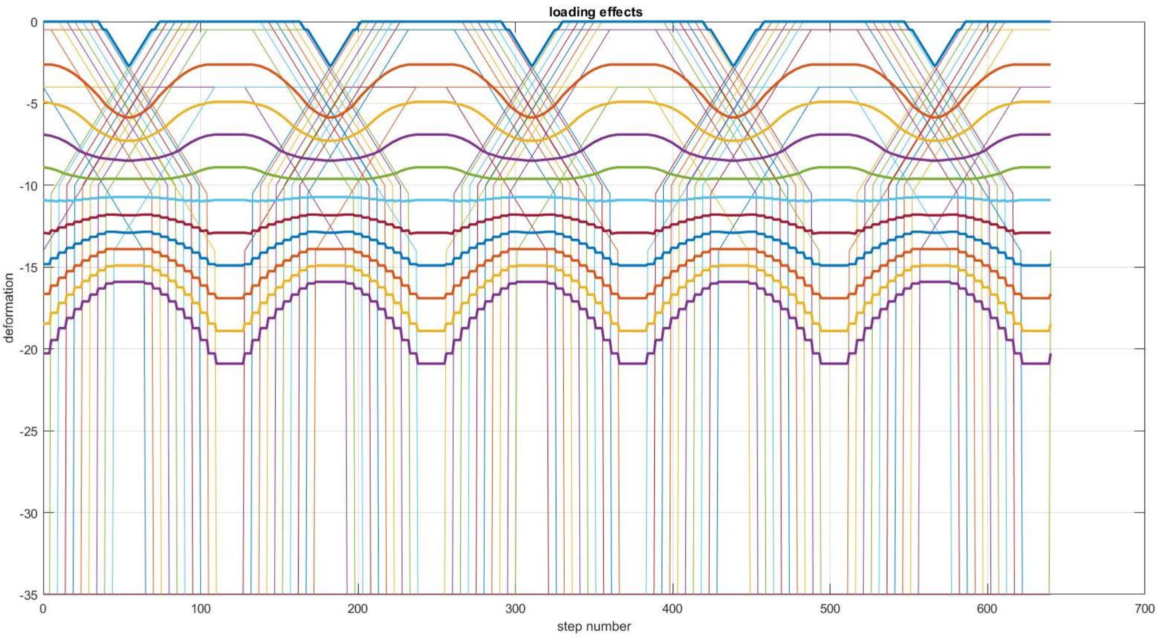


Figure 8.9

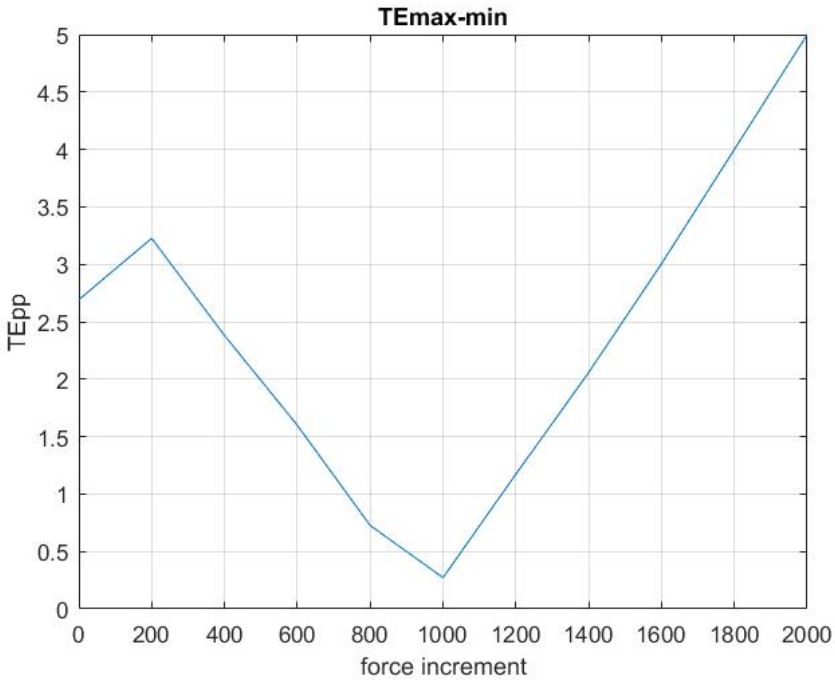


Figure 8.10

8.6 HELICAL GEAR SUGGESTED AR, ER; NO PE; NO PAE; INPUTTED PARABOLIC CROWNING; NO MISALIGNMENT

Table 8.6

DRIVER GEAR TOOTH NUMBER	20
DRIVEN GEAR TOOTH NUMBER	20
PITCH ERROR	NO
NORMAL MODULE	2
NORMAL PRESSURE ANGLE(DEGREE)	20
HELIX ANGLE(DEGREE)	0
ADDENDUM MULTIPLIER(Y*m=add)	1
DEDENDUM MULTIPLIER(X*m=add)	1.25
FACEWITDH(mm)	10
PRESSURE ANGLE ERROR	NO
DESIGN LOAD(N)	1000
STIFFNESS(N/mm/micron)	10
SUGGESTED AR OF LEFTSIDE	YES => 10microns
SUGGESTED AR OF LEFTSIDE	YES => 10microns
SUGGESTED ER OF LEFTSIDE	YES => 3.2877
SUGGESTED ER OF LEFTSIDE	YES => 3.2877
NUMBER OF NODES AT BASE PITCH	128
THIN SLICE NUMBER	10
MISALIGNMENT(DEGREE)	0
CROWNING	YES
TYPE OF CROWNING	PARABOLIC
AMOUNT OF CROWNING AT FACE	14 microns
Y=A*X^B=> A=?	1
Y=A*X^B=> B=?	3
MULTIPLIER FOR Fmin=min*Fd	0
MULTIPLIER FOR Fmax=min*Fd	2
% INCREASE IN LOADING	20%

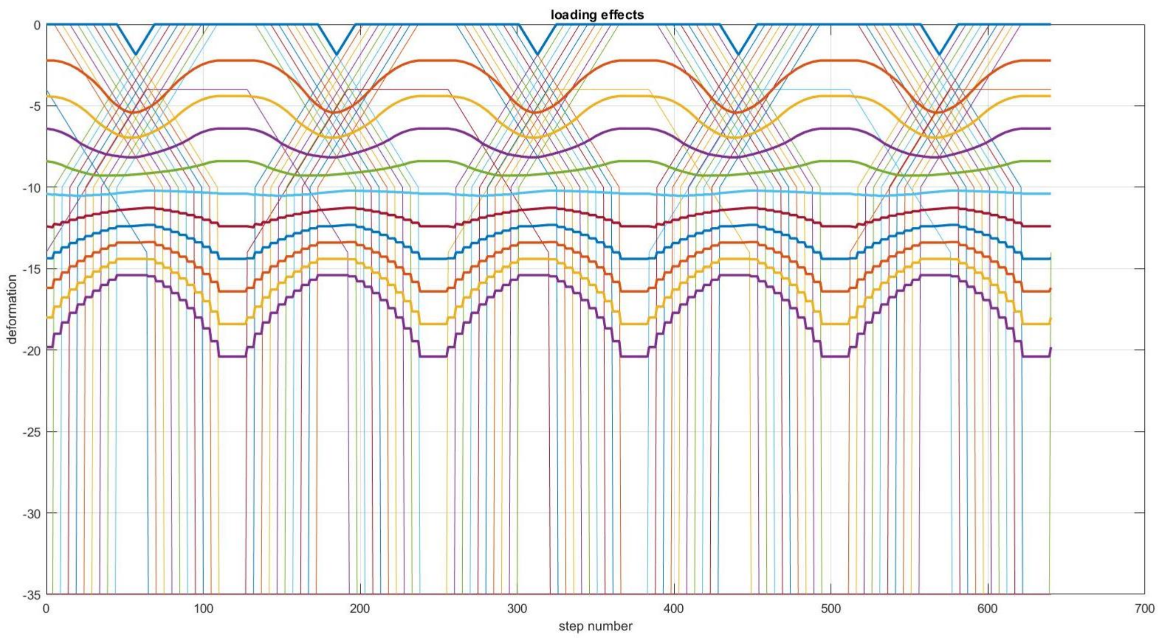


Figure 8.11

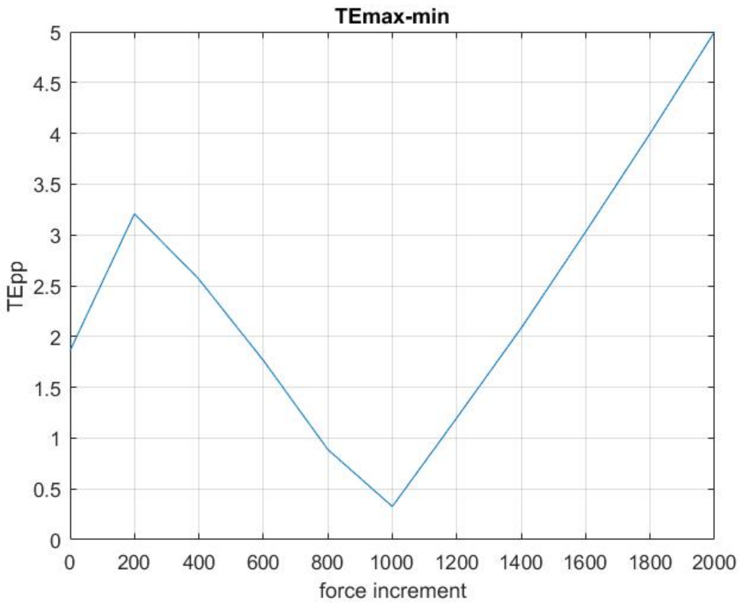


Figure 8.12

8.7 HELICAL GEAR SUGGESTED AR, ER; NO PE; NO PAE, CROWNING; INPUTTED MISALIGNMENT

Table 8.7

DRIVER GEAR TOOTH NUMBER	20
DRIVEN GEAR TOOTH NUMBER	20
PITCH ERROR	NO
NORMAL MODULE	2
NORMAL PRESSURE ANGLE(DEGREE)	20
HELIX ANGLE(DEGREE)	0
ADDENDUM MULTIPLIER(Y*m=add)	1
DEDENDUM MULTIPLIER(X*m=add)	1.25
FACEWITDH(mm)	10
PRESSURE ANGLE ERROR	NO
DESIGN LOAD(N)	1000
STIFFNESS(N/mm/micron)	10
SUGGESTED AR OF LEFTSIDE	YES => 10microns
SUGGESTED AR OF LEFTSIDE	YES => 10microns
SUGGESTED ER OF LEFTSIDE	YES => 3.2877
SUGGESTED ER OF LEFTSIDE	YES => 3.2877
NUMBER OF NODES AT BASE PITCH	128
THIN SLICE NUMBER	10
MISALIGNMENT(DEGREE)	10
CROWNING	NO
MULTIPLIER FOR Fmin=min*Fd	0
MULTIPLIER FOR Fmax=min*Fd	2
% INCREASE IN LOADING	20%

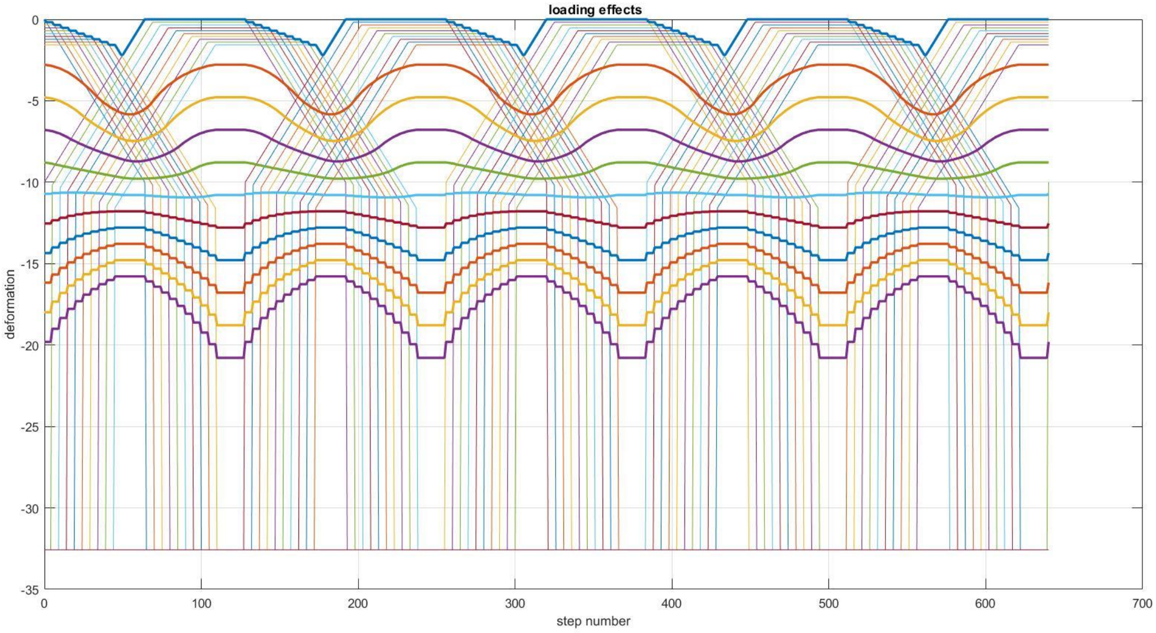


Figure 8.13

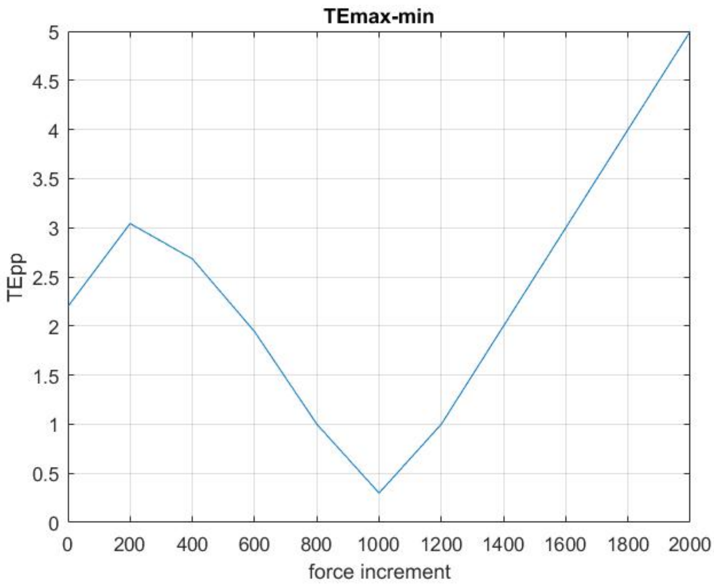


Figure 8.14

8.8 HELICAL GEAR RANDOM AR, ER; INPUTTED PE,PAE, CROWNING, MISALIGNMENT

Table 8.8

DRIVER GEAR TOOTH NUMBER	20
DRIVEN GEAR TOOTH NUMBER	20
PITCH ERROR	YES
NORMAL MODULE	2
NORMAL PRESSURE ANGLE(DEGREE)	20
HELIX ANGLE(DEGREE)	14
ADDENDUM MULTIPLIER(Y*m=add)	1
DEDENDUM MULTIPLIER(X*m=add)	1.25
FACEWITDH(mm)	10
PRESSURE ANGLE ERROR	YES
PAE OF DRIVEN GEAR(DEGREE)	0.05
PAE OF DRIVER GEAR(DEGREE)	0
#OF MESHFOR INVOLUTE CURVE	100
DESIGN LOAD(N)	1000
STIFFNESS(N/mm/micron)	10
SUGGESTED AR OF LEFTSIDE	NO =>5 micron
SUGGESTED AR OF LEFTSIDE	NO =>8 micron
SUGGESTED ER OF LEFTSIDE	NO =>1.75 micron
SUGGESTED ER OF LEFTSIDE	NO =>3.5 micron
NUMBER OF NODES AT BASE PITCH	128
THIN SLICE NUMBER	10
MISALIGNMENT(DEGREE)	10
CROWNING	YES
TYPE OF CROWNING	LINEAR
AMOUNT OF CROWNING AT FACE	14 microns
EXTENT OF FIRST REGION	4microns
MULTIPLIER FOR Fmin=min*Fd	0
MULTIPLIER FOR Fmax=min*Fd	2
% INCREASE IN LOADING	20%

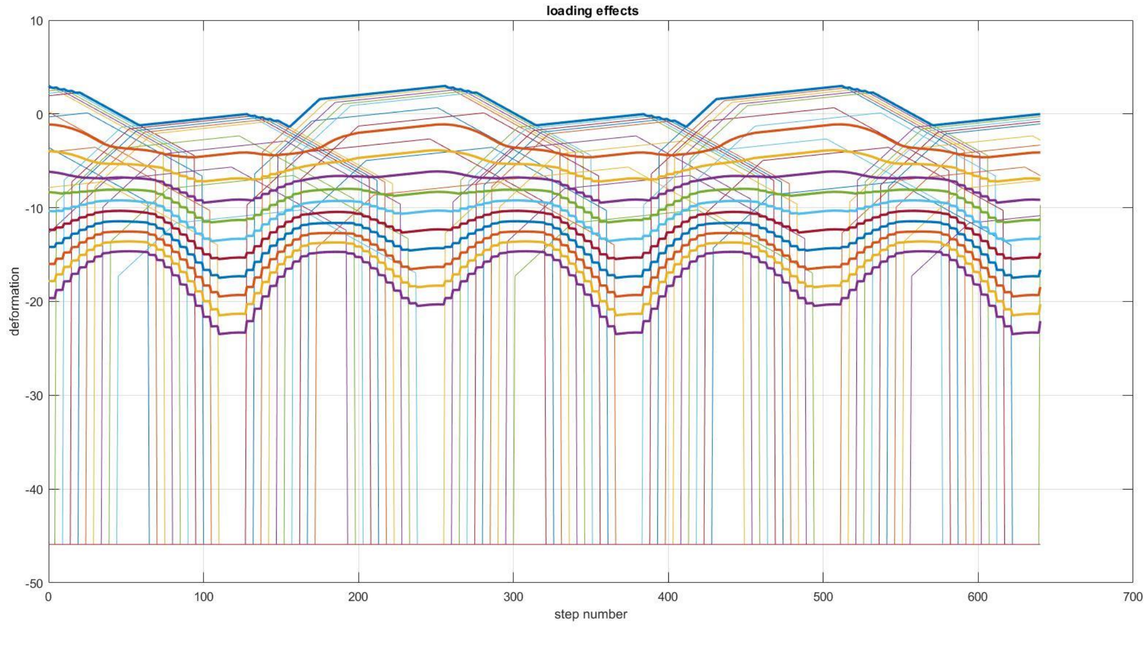


Figure 8.15

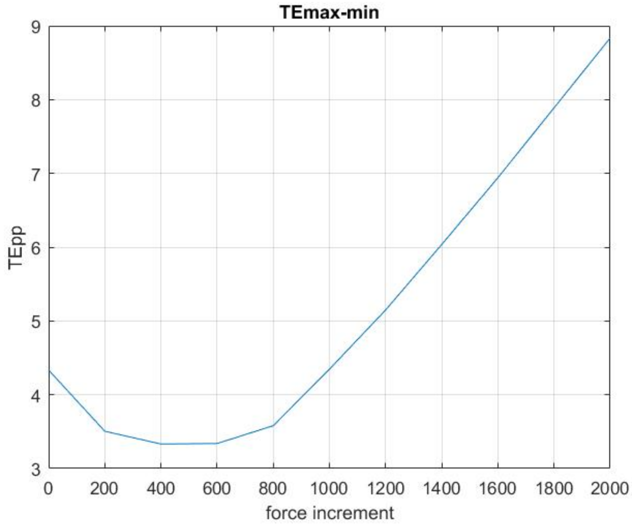


Figure 8.16

9 DISCUSSION AND CONCLUSION

9.1 DISCUSSION AND CONCLUSION

We develop a user interface software for calculate the isolated and continous TE curves. After calculating these we applied some paramaters which are amount of relief, extent of relief, pitch error , pressure angle error ,SAP and TIF values , crowning and misalignments. Then TE curves are loaded and better TE curves and TEpp values are found.All the results are meeting with references [1],[4],[5],[20],[22],[23] and [24]. The program accomplish it's mission by reach the purposes of software. The software offers the user time-saving calculations,iterations and best result for reach the minimum peak-to-peak TE values.With these program user have opportunity to see the actual TE values for actual gear or gears systems with chance of change and see effects of mentioned parameters.

As a conclusion , this program is aimed to help user to get better design and analysis results for LCR spur or helical gear systems in a possible best correctness.

9.2 TIPS FOR FUTURE WORK

This program satisfied expectations from itself, but nothing is perfect. Some possible extensions/updates can be made for future. These are may be follwing ones:

- Including HCR gears.
- Adding other gear thypes.
- Increasing nodes and thin slice numbers.
- Increasing correctness near the perfect.
- Adding stress calculations
- Modelling TE curves results on gear as 3D.
- Checking results with International Gear Standarts.
- Guessing the possible failure locations on gear teeth.

LIST OF REFERENCES

[1] Lemanski, A. J., Gear Design, S.A.E., Warrendale 1990. Ch 3

[2] Buckingham, Earle, Analytical mechanics of spur gears, Dover, New York. 1988.

[3] Harris, S.L., 'Dynamic loads on the teeth of spur gears.' Proc. Inst. Mech. Eng., Vol 172, 1958, pp 87-112.

[4] Gregory, R.W., Harris, S.L. and Munro, R.G., 'Dynamic behaviour of spur gears.' Proc. Inst. Mech. Eng., Vol 178, 1963-64, Part I, pp 207-226.

[5] Munro, R.G., 'The Effect of Geometrical Errors on the Transmission of Motion Between Gears.' I. Mech. E. Conf. Gearing in 1970, Sept. 1970, p79.

[6] Cremer, L., Heckl, M., and Ungar, E.E., Structure-borne sound. Springer-Verlag, 1973, Berlin.

[7] Kohler, H.K., Pratt, A., Thompson, A.M. Dynamics and noise of parallel axis gearing. Inst. Mech. Eng. Conf. Gearing in 1970, Sept, pp 111-121.

[8] Furley, A.J.D., Jeffries, J.A. and Smith, J.D., 'Drive Trains in Printing Machines', Inst. Mech. Eng. Conference, Vibrations in Rotating Machinery, Cambridge, 1980, pp.239-245

[9] Palmer, D. and Munro, R.G., 'Measurements of transmission error,vibration and noise in spur gears.' British Gear Association Congress,1995, Suite 45, IMEX Park, Shobnall Rd., Burton on Trent.

[10] The Design Unit, Stephenson Building, Claremont Rd, Newcastle upon TyneNEl 7RU, U.K. D.A. Hofrnann.

[11] Dept. of Mechanical Eng., Queensgate, Huddersfield, HD1 3DH, U.K. Prof R.G. Munro.

[12] University Eng. Dept., Trumpington St., Cambridge CB2 1PZ, U.K. Dr J.Derek Smith.

[13] Ohio State Univ., Mech. Eng. Dept., 206 West 18th Ave., Columbus, Ohio, 43210-1107. Prof D. R. Houser.

[14] INS A de Lyon, Villeurbane, Cedex, France. Mr D. Remond.

[15] University of New South Wales, Australia. Mr R.B. Randall.

[16] Tech. Univ. of Ostrava, CZ - 703 88 Ostrava, Czech Republic. Mr. Jiri Tuma.

[17] Gregory, R.W., Harris, S.L. and Munro, R.G., 'Dynamic behaviour of spur gears.' Proc. Inst. Mech. Eng., Vol 178, 1963-64, Part I, pp 207-226.

[18] Maag Gear Handbook (English version) Maag, CH8023, Zurich,Switzerland.

[19] Harris, S.L., 'Dynamic loads on the teeth of spur gears.1 Proc. Inst. Mech. Eng., Vol 172, 1958, pp 87-112.

[20]Niemann, G. and Baethge, J., 'Transmission error, tooth stiffness, and noise of parallel axis gears.' VDI-Z, Vol 2, 1970, No 4 and No8.

[21] Yıldırım, N. (1994). Theoretical and Experimental Research in High contact Ratio Spur Gearing. Ph.D. Thesis in School of Engineering. The University of Huddersfield School of Engineering.

[22](Kütük M.A. (2003). A Theoretical Research into Effects of Tooth Profile and Assembly Errors on the Helical Gear Performance. Ph.D. Thesis in Mechanical Engineering. University of Gaziantep.

[23] Calculating SAP and TIF by Dan Thurman Gear Technology pg.65-68 relese September/ October 1999

[24] Gear Noise and Vibration Second Edition,Revised and Expanded J. Derek Smith.