iJRASET
International Journal For Research in
Applied Science and Engineering Technology

INTERNATIONAL JOURNAL
FOR RESEARCH
IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 9 Issue: VIII Month of publication: August 2021

DOI: https://doi.org/10.22214/ijraset.2021.37588

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429
Volume 9 Issue VIII Aug 2021- Available at www.ijraset.com

Gear FE Modelling Process Automation Using TCL Script

Amardeep Kamble[1], Siddhesh Bandekar[2]
[1, 2]Department of Mechanical Engineering, JSPM'S Jayawantrao Sawant College of Engineering, Pune, Maharashtra, India

Abstract: Paper describes the automation in design process and its effect on the overall process. The objective of automation is to reduce the effort and time taken by any design personnel while working on product. For any gearbox designing gear train is a crucial part and it should run at optimum level throughout its life span. Gear designing a one-step while working on gear train and analysing these gears in FEA is an important step of design process. While working on multiple iterations of gear designs it is always required to change parameters of gear and run FEA fast and efficiently. Paper describes automation process done to model a gear in FE software which ultimately reduces time for design processes by eliminating few steps during design phase.

Keywords: Gear automation, TCL script, Involute profile generation, Finite Element Analysis.

I. INTRODUCTION

With the industrial revolution, early gears made of wood were transformed into modern gears, offering remarkable tooth shapes. Nowadays, gear design is largely standardized. Organizations such as AGMA or ISO supply standards that establish accuracy classification, rating formulas, etc. Gears are the backbone & one of the most critical components in mechanical power transmission systems. Common gear practice also offers common design techniques that are largely employed. But as gear applications are getting precise, accuracy in designing and manufacturing is expected to increase thus this becomes challenging in terms of design process. Gear design and manufacturing is very intricate and complex process which if not carried correctly can lead to failure of the system i.e. Gear noise, loss of power in transmission, breaking of teeth while torque transmission. Generalized gear design and manufacturing process is standardized by organization like ISO & AGMA but some parameters can be varied by organizations to improve performance based on years of experience. The assembly precision also has a strong influence on gear behaviour. To reduce the effects of assembly deviation such as misalignment, flank crowning and tip relief are often recommended. Apart from conventional requirements being imposed on the load carrying capacity and fatigue life of heavy-duty gears, modern gear developments are characterized by a growing environmental awareness. This means not only the increase of power density but also the consideration of efficiency ratio, natural resources, and noise emission. Furthermore, the expanding techniques in the field of gear manufacturing are creating new possibilities to realize gears which are suitable for modern requirements in terms of load capacity and running behaviour. The dynamic effects within the mesh are essentially determined by the engagement shock, the parametric excitation. With all factors affecting in gear and gear train it is always good to analyse the system before creating any problem in actual field working. Thus, proper automated tools will help us in future to work on these processes and target the possible issue with system.

II. CONVENTIONAL PROCESS OF GEAR MESHING

While talking about conventional process of finite element analysis, pre-processing is done in various software packages such as Altair HyperMesh, Ansys Spaceclaim etc. Amongst all packages Altair HyperMesh is well known and preferred by many organizations because of its handling and user-friendly capability. Conventional process of gear modelling for finite element analysis purpose consists of various steps. These steps are discussed in below flow chart: As shown in above figure, all these processes are user controlled and depends skill of individual based on experience. For novice person he/she might need more time as compared to expert person. Thus, there is variation in time taken to follow the process and complete the task based on person to person skill.

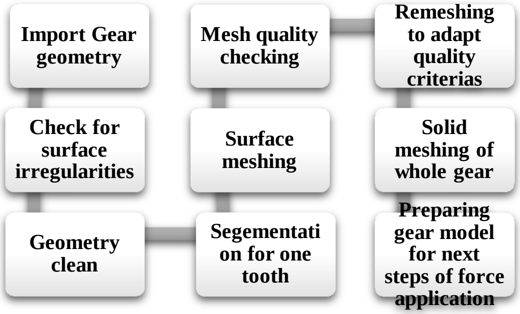

Import Gear geometry
Check for surface irregularities
Geometry clean
Segementati on for one tooth
Surface meshing
Mesh quality checking
Remeshing to adapt quality criterias
Solid meshing of whole gear
Preparing gear model for next steps of force application

Figure 1 : Process flow for gear modelling in pre-processor

III.GEAR FE MODEL CREATION: AUTOMATED PROCESS

Automation process for gear FE modelling is sub-divided in following parts:

A. Understanding Macro Geometry of Gear

Automation of gear generation is latter process but understanding basics about gear tooth is necessary to start constructing body or structure. Below are some of the basic nomenclature described with their descriptions because these are necessary not only while creating final mesh in software but also while writing code in TCL/TK language.

1) *Tooth module (m): This is the height of the addendum in mm. Also, module m can be defined as the number of mm in the pitch circle diameter for each tooth in the gear. This takes us to our first gear calculation formula:*

m = d/z

where m = the gear module, d = operating pitch diameter of the gear, z = the number of teeth the gear has.

2) *Pitch (p):* If you multiply Module by Pi, you can obtain Pitch (p). Pitch is the distance between corresponding points on adjacent teeth.
3) *Circular Pitch (CP):* Circular Pitch (CP) denotes the reference pitch (p). For instance, gears can be produced at an exact integral value, such as CP5/CP10/CP15/CP20. Transformation from CP to Module: m = CP / π
4) *Diametrical Pitch (DP):* DP stands for Diametrical Pitch. By ISO standards, the unit Millimeter (mm) is designated to express length, however, the unit inch is used in the USA, the UK and other countries; Diametrical Pitch is also used in these countries. Transformation from DP to Module: m = 25.4 / DP
5) *Pressure angle (a):* Pressure angle is the leaning angle of a gear tooth, an element determining the tooth profile. Recently, the pressure angle (α) is usually set to 20°, however, 14.5° gears were prevalent. Below figure shows some of the basics about gear tooth.

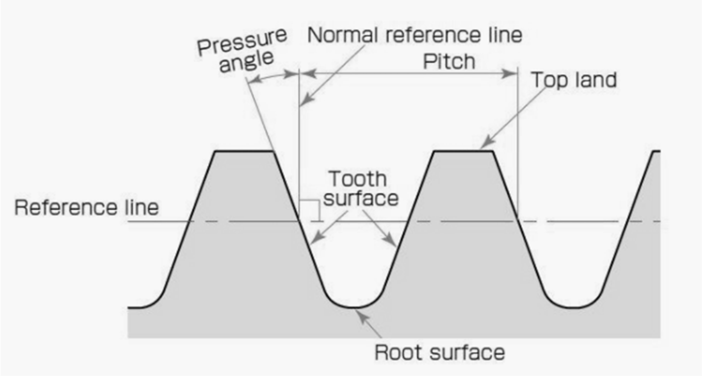


Figure 2: Pitch (p) & Pressure angle(α) of gear tooth

6) *Number of teeth (z):* As name suggests it is the total number of teeth in a gear.
7) *Tooth depth (h):* Tooth depth is determined from the size of the module (m). Introduced here are Tooth Profiles (Full depth) specified by ISO and JIS (Japan Industrial Standards) standards. Please see Figure 4 below for explanations for Tooth depth (h) / Addendum (ha) / Dedendum (hf).

Tooth depth (h) is the distance between tooth tip and the tooth root.

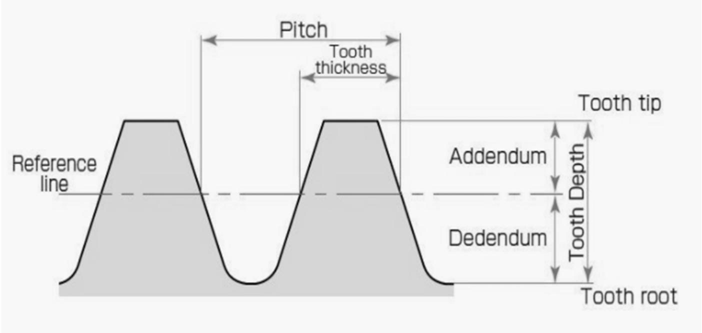


Figure 3: Tooth depth (h), Addendum (ha), Dedendum (hf) of a gear

8) Addendum (ha)

Addendum (ha) is the distance between the reference line and the tooth tip.

ha = 1.00 x m

9) Dedendum (hf)

Dedendum (hf) is the distance between the reference line and the tooth root.

hf = 1.25 x m

10) Tooth thickness (b)

Tooth thickness (b) is basically half the value of pitch (p).

b = π x m/2

11) Diameter of Gears

Reference diameter (d) as name suggests is reference diameter for construction of gear. Sometimes it is also called as pitch circle diameter as it is the circle through which line of action passes.

Reference diameter (d) = z x m

Tip diameter (da) is the addendum diameter of the circle, so basically it is reference diameter in addition of two times addendum of gear tooth.

Tip diameter (da) = d + (2 x m)

Root diameter (df) is dedendum diameter of circle. Thus, it is reference diameter minus two times dedendum of gear tooth.

Root diameter (df) = d – (2.5 x m)

Below figure shows the different diameters of gear discussed in earlier point.

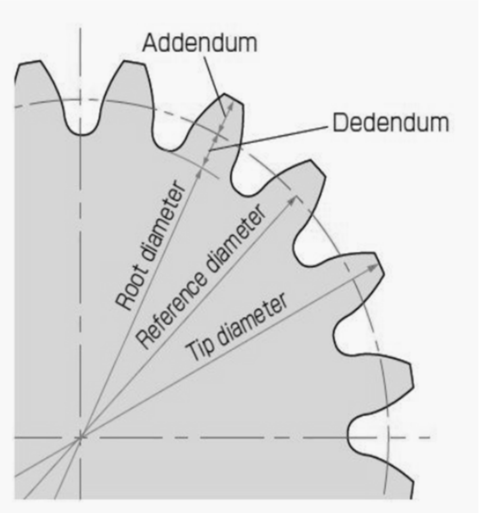


Figure 4: Reference, Tip & Root diameters of a gear

B. Choosing automation process language

With rapidly evolving world, automation is nothing new. Automation with complex languages might serve the purpose but is not recommended as in aftereffects, in case of modifying source code can be troublesome. To avoid this, it is always recommended to work on language which easy to learn and has a huge user base so in case of any issues person can ask the questions to user community and get the answers easily.

There are widely available language packages such Python, JAVA, HTML but for this task Tcl/Tk was chosen. The reason behind is the software which is being used to generate mesh and assembly uses Tcl/Tk as base framework thus it is always to use native language of software and we can modify as per our need and test it rigorously till we get desired results. Apart from that below are some of the advantages mentioned for Tcl/Tk language:

1) *Rapid Development:* The most important reason why people use Tcl is that it gets their job done faster. In many cases user can implement applications 5-10x faster with Tcl than with other languages. Once an application is built in Tcl, it can also be evolved rapidly to meet changing needs.
2) *Graphical user Interfaces:* With its Tk toolkit, Tcl provides facilities for creating GUIs that are incredibly simple yet remarkably powerful. For example, the Tk canvas widget makes it easy to create displays with graphics, yet it also provides powerful facilities such as bindings and tags.
3) *Easy to Learn:* Tcl is a very simple language. Experienced programmers can learn Tcl and produce their first interesting application in just a few hours or days. Casual programmers can also learn Tcl quickly. Tcl is often used in situations where experienced programmers create a base set of facilities, and more casual programmers write Tcl scripts to customize those facilities, create business rules, etc.
4) *Mature but Evolving:* Because Tcl is constantly evolving, new cutting-edge features are being added all the time, yet always with an eye to the consistent quality and concern for backwards compatibility that Tcl has always been known for.
5) *Deployment:* Dynamic languages often make deployment harder, because user needs to get both the language interpreter and the application scripts onto the target machine. Most dynamic languages provide tools to "compile" everything into a single executable (Tcl has had that too, since about 1993). But Tcl/Tk goes way beyond those simple solutions, using technologies like the Tcl Virtual File System, and Starkits and Starpacks to make deployment more flexible, powerful, and transparent. Other options allow commercial applications to protect their intellectual property, a rare capability in dynamic languages.
6) *Testing*: Tcl is an ideal language to use for automated hardware and software testing, and it may well be the dominant language used for this purpose. With Tcl user can easily connect to testing hardware or internal APIs of an application, invoke test functions, check the results, and report errors. Tcl's interpreted implementation allows tests to be created rapidly, and the tests can be saved as Tcl script files to reuse for regression testing. If users are testing a software application, Tcl allows to connect directly to lower-level APIs within the application, which provides much more precise and complete testing.
7) *The Tcl community*: Another attractive reason for using Tcl is the large and helpful community of Tcl users and developers. The Tcl community is a constant source of ideas, free extensions, applications, and technical support.
8) *It's free so no Cost Involved*: Tcl is open source — freely available, meaning you can do anything you want with it, including modifying it to suit your own needs or incorporating it into commercial products, no strings attached.

Apart from these advantages one of the most useful and necessary quality of Tcl is user can write arithmetic expressions and language can execute it perfectly without any issues. This is very useful for this project since there are few formulae required to calculate macro geometry of gear.

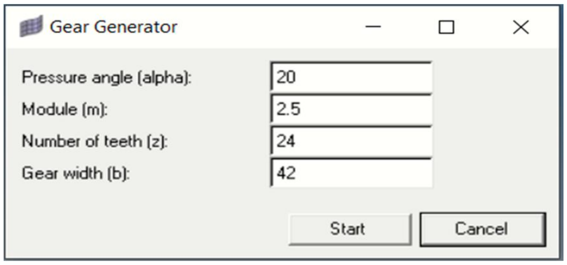
C. Implementation of Automation

With overview of how macro geometry works based on formulae and looking at the effect of number of elements in terms of mesh convergence now automation part can be carried out.

Automation consists of three phases:

1) Graphic User Interface (GUI) for user inputs
2) Tcl script generation for spur gear tooth generation
3) User inputs for gear center & axis of rotation

Graphic User Interface or GUI is basically an accessory provided to user where variable data can be input based on requirement and then this data can be linked with Tcl script so to get desired output. For this exercise of gear generation, though macro geometry consists of many variable dimensions it is possible to link the data with few basic requirements such as gear module (m), number of teeth (z), pressure angle (α), gear tooth width. Using the Tk which is GUI generation toolkit provided along with Tcl is used and GUI is prepared. Below is the image from Altair's HyperMesh software showing GUI with labels.


Gear Generator

Pressure angle (alpha): 20
Module (m): 2.5
Number of teeth (z): 24
Gear width (b): 42

Figure 5: Graphic User Interface for gear generator

After user inputs, user needs to click on ‘Start’ button which will trigger the Tcl script and following steps will be generated. The steps include creation of basic gear diameters and based on gear diameters involute profile is generated using the input data.
Once user clicks start button GUI asks to specify gear center and axis of rotation which can be provided in terms of two nodes.
After this, creation for various geometric entities will start which include reference diameter, base diameter, tip & root diameter. Based on this following construction is generated as shown. These diameters provide base for creation of involute profile. Now based on tooth thickness (b) center line is transformed and tooth thickness is determined by formula b = π x m/2. Using involute profile construction method based on tooth thickness, circular pitch & base diameter following construction is generated.

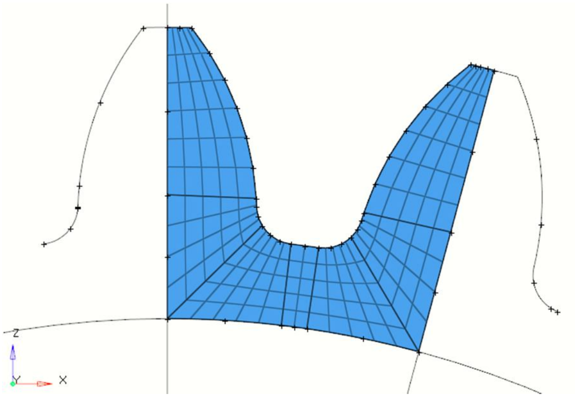


Figure 6: Mesh creation based on tooth profile (2 Dimensional)

Once these curves are generated half tooth can be used to mesh the tooth face and then can be extruded along width direction. On creation of one tooth then it can be rotated about axis of rotation and multiple teeth are created to complete the gear mesh creation.

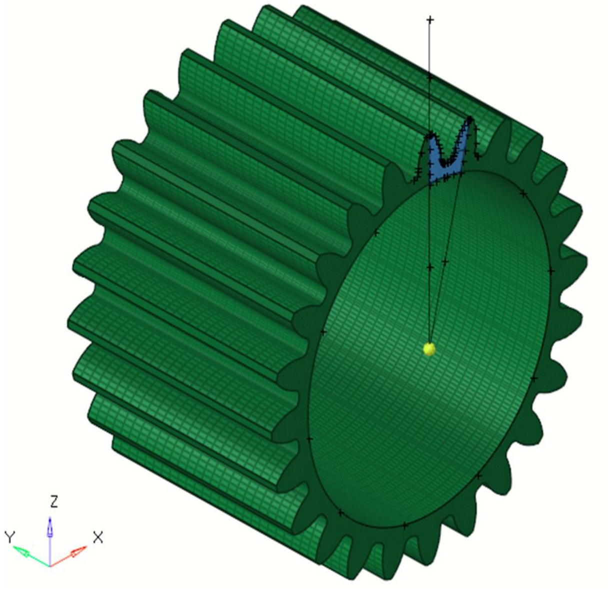


Figure 7: 3D Spur Gear mesh creation

So, all the steps are automated once user inputs are executed and final output is as shown in the above figures. Thus, fulfilling the requirements from this study. Apart from this process automation main goal of this process was to reduce manual intervention in the creation process to save time.

IV.RESULTS & DISCUSSION

Conventional process took around 30-45 minutes whereas automated process completed the same task within 4-6 minutes thus reducing time consumption by significant amount around 80%. Thus, here conclusive results prove the effective use of automation and can be extended for further studies also such as simple shafts or splined shafts etc.

V. NOMENCLATURES

m	Module of gear
p	Pitch of gear
CP	Circular Pitch
DP	Diametrical Pitch
a	Pressure Angle
z	Number of teeth
h	Gear tooth depth
ha	Gear tooth addedndum
hf	Gear tooth dedendum
b	Gear tooth thickness
d	Reference diameter
da	Addendum diameter of gear
df	Dedendum diameter of gear

REFERENCES

[1] Veerasamy Suresh Babu, Abubacker K. M., Development of Involute Profiled Spur Gear Model with Excel Spreadsheet, Solidworks and CAD Technique, SSRG International Journal of Mechanical Engineering (SSRG - IJME) – Volume 5 Issue 5–May 2018

[2] Edward E. Osakue, Lucky Anetor, A method for constructing standard involute gear tooth profile, International Mechanical Engineering Congress and Exposition IMECE2018, November 9-15, 2018.

[3] Dong Liang, Tianhong Luo, Rulong Tan, Generation and Meshing Analysis of a New Type of Double Helical Gear Transmission, Hindawi Mathematical Problems in Engineering, Volume 2018, Article ID 1539849.

[4] Jiaxing Zhan, Mohammad Fard, Effects of helix angle, mechanical errors, and coefficient of friction on the time-varying tooth-root stress of helical gears, March 2018.

[5] Clif Flynt, Chapter 15 - Programming Tools - The Morgan Kaufmann Series in Software Engineering and Programming 2003,

[6] John K. Ousterhout, TCL and the TK toolkit, First Edition – 1994.