

DEVELOPMENT APPLICATIONS FOR CAD - CAE SYSTEMS

Senad Rahimić

**University of "Džemal Bijedić" Mostar, Faculty of Mechanical Engineering,
Univerzitetski Kampus, 88104 Mostar
Bosnia and Herzegovina**

Amar Kukuruzović

**University of "Džemal Bijedić" Mostar, Faculty of Mechanical Engineering,
Univerzitetski Kampus, 88104 Mostar
Bosnia and Herzegovina**

ABSTRACT

The core of a CAD system is the, which makes use of graphics for product representation, databases for storing the product model and drives the peripherals for product presentation. The possibility of improvement CAD - CAE systems using the application based on object oriented method paper is shown in this paper. The application is linked to the CAD-CAE systems using the API commands. Originally the technique was aiming at automating a number of tasks a designer is performing and in particular the modelling of the product. Today CAD systems are covering most of the activities in the design cycle, they are recording all product data, and they are used as a platform for collaboration between remotely placed design teams.

Keywords: CAD-CAE system, application, drills

1. INTRODUCTION

This section of the report discusses the development of an automated drill modeling tool in Solidworks. The tool uses Solidworks to generate a 3D model of a drill based on manufacturing parameters of the drill supplied by the user. The need to use manufacturing parameters to model drills will be established. The applet uses a GUI to accept these parameters from the user and generates the model "on-the-fly". The applet allows the user to save the model in a variety of formats which can be them imported into a meshing program or into the meshing module of an FEA package for subsequent use in FE-based simulations. The applet was written using Visual Basic APIs in SolidWorks 2012 and is forward-compatible with SolidWorks 2012. The current version of the applet is restricted to designing two-flute twist drills. The subsequent section discusses the geometry of conventional two-flute drills and their design and manufacturing. This is followed by a discussion of the 3D analytical formulation of drill geometry. After this, an algorithm to model drills using commercial CAD tools is discussed. Finally, an implementation of this algorithm in Solidworks to create an automatic drill geometry modeling tool is presented.

2. DESIGN OF DRILLS

The geometry of two-flute twist drills is shown in Figure 1. For more information on the standard description of features and geometry of drills, the reader is referred to Galloway [1] and the ASM Handbooks [2]. The geometric parameters of conventional two-flute twist drills are determined by their manufacturing parameters. Drill manufacturing consists primarily of two grinding steps, namely grinding the flute faces and grinding the flank faces. The parameters of these grinding operations determines the geometric parameters of the drill. Parameters such as point angle and web thickness are implicit functions of the drill's manufacturing parameters.

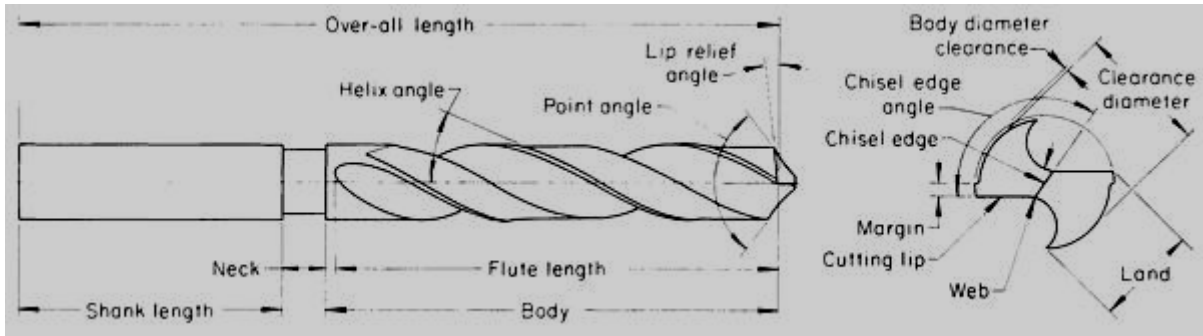


Figure 1: Standard Geometry of Drill

The next section will discuss how the manufacturing parameters such as grinding wheel cross-section and axis of rotation determine the geometric parameters of the drill. We will also see that the same design features can be generated with multiple sets of manufacturing parameters. Before we take a look at an analytical formulation of the drill geometry, past work in characterizing the drill geometry - based on which much of the work in this report has been developed - is discussed.

3. ANALYTICAL FORMULATION

The cross-section the flute is dependent on the shape of the grinding wheel used for grinding it. The cross-section of the drill has to be designed such that it generates a straight secondary cutting edge when the flanks are ground. Hence, the shape of the flute grinding wheel is dependent on the specifications of the drill that is being ground. Instead of describing the grinding wheels, we can directly consider the cross sectional profile of the flute. The cross-section of the flute can be divided into 8 sections as shown in Figure 4. Sections 1, 2, 3, 4 are unground parts of the drill-blank and are arcs which make up a circle. Sections 5 and 6 can be described by the following polar equation:

$$\psi = \sin^{-1} + \frac{\sqrt{r^2 - (\frac{W}{2})^2}}{r} \tan h \cot p \quad (1)$$

where, W is the web thickness, r is the radius of the drill, h is the helix angle and p is the half-point angle.

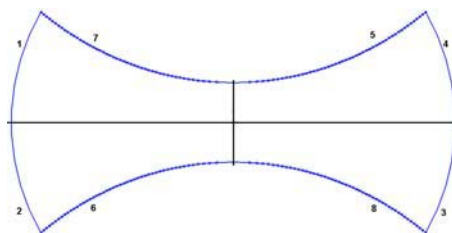


Figure 2: Flute Cross-Section

Here, r is varied from W/2 to R. This polar equation makes sure that the flank section produces a drill with a straight cutting edge. Sections 7 and 8 do not contribute much to the cutting performance of the drill and only need to be optimized to provide rigidity. For simplicity, they can be modeled as symmetric to sections 5 and 6 respectively. To convert the flute profile from 2D to a 3D boundary surface, a z-component term can be appended to the equation to capture the helical profile. The z-component term is as follows (for a drill of radius r:

$$z_{flute} = \left| \frac{\tanh h}{r} z \right| \quad (2)$$

Here, h is the peripheral helix angle, and is (for a drill of length l):

$$h = \tan^{-1} \frac{2\pi r}{l} \quad (3)$$

From the above analysis, we can see that the following parameters (geometric and manufacturing) are needed to completely describe a drill:

- *Geometric* - R Radius of drill, w Web thickness, h Helix angle, p Half-point angle.
- *Manufacturing* -d x-shift of cone (in cone-coordinate system), S y-shift of cone (in cone-coordinate system), _ Cone angle

4. IMPLEMENTATION USING SOLIDWORKS

Solidworks is a popular 3D modeling CAD package. Solidworks uses a feature-based parametric approach for 3D drawings. Features are defined to create volume and modifications to sketches and these features can be rolled-back or modified to create multiple configurations of the same part. The program uses a feature hierarchy to determine “child” and “parent” features. Solidworks allows models to be saved in many different graphical formats and is hence very useful in ensuring that the model is portable. Solidworks is also integrated with an API (Application Programming Interface) which contains many functions that can be called from programming languages such as Visual Basic and C++. These functions provide access to Solidworks’ graphical engine and can be used to create solid models. Programs can also be written in these programming languages that can accept input to generate user-defined solid models.

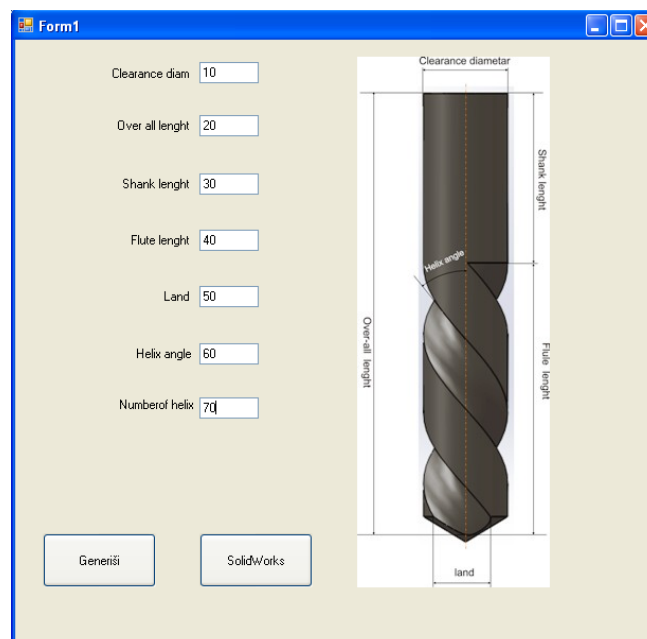


Figure 3: Drill Form

Currently, the software supports saving the drill in the .SLDPRT (Solidworks), .SAT (ACIS) and .IGS (International Graphics Exchange) formats. The user interface is shown in Figure 3.

```

Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click
    System.Threading.Thread.CurrentThread.CurrentCulture = _
        System.Globalization.CultureInfo.CreateSpecificCulture("en-US")

    Dim objExcel As New Microsoft.Office.Interop.Excel.Application
    Dim objWorkbook As Microsoft.Office.Interop.Excel.Workbook
    Dim objWorksheet As Microsoft.Office.Interop.Excel.Worksheet

    objWorkbook = objExcel.Workbooks.Open("E:\Documents and Settings\Amar\Desktop\burgija1.xlsx")
    objWorksheet = CType(objWorkbook.Worksheets.Item("Sheet1"), Microsoft.Office.Interop.Excel.Worksheet)

    objWorksheet.Cells(7, 2) = TextBox1.Text
    objWorksheet.Cells(7, 3) = TextBox2.Text
    objWorksheet.Cells(7, 4) = TextBox3.Text
    objWorksheet.Cells(7, 6) = TextBox4.Text
    objWorksheet.Cells(7, 15) = TextBox5.Text
    objWorksheet.Cells(7, 16) = TextBox6.Text
    objWorksheet.Cells(7, 17) = TextBox7.Text

    objWorkbook.Save()
    objWorkbook.Close(False)
    objExcel.Quit()
End Sub

```

Figure 4: Visual Basic code

Visual Basics APIs was used to design an ergonomic and accessible user interface show on Figure 4. The interface accepts inputs from the user regarding the geometric and manufacturing specifications of the drill in SI units. A schematic of a drill with all the dimensions marked is also shown alongside for ease-of-use. After entering the data, the user can click the “Generate Drill” in order to invoke the necessary Solidworks commands to execute the program. Following a brief wait, the drill is generated in Solidworks.

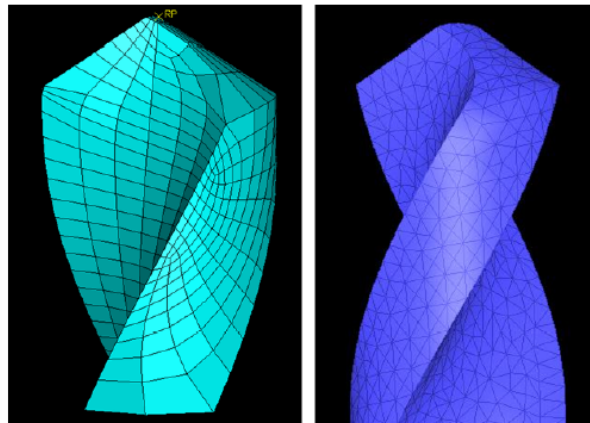


Figure 5: Meshed in SolidWorks Simulation model.

5. CONCLUSION

Future work includes expanding the package to model different drill types. A module that permits the description of arbitrary ground geometries can also be included. Physical prototypes of the drill also have to be created and compared with regular drills to validate the accuracy of the modeling technique. Using the capability of the software to save in different formats, a drill was generated and exported into Simulation modul of SolidWorlks, respectively. The drills were then meshed using the meshing module of these two packages.

6. REFERENCES

- [1] Choi, J., Min, S., Dornfeld, D. A., Alam, M., and Tzong, T., 2003. “Modeling of inter-layer gap formation in drilling of a multilayered material”. In CIRP International workshop on modeling of machining operations.,
- [2] BHsu, B., 2002. “Computer simulations for burr formation study”. PhD thesis, University of California, Berkeley.
- [3] Committee, A. I. H., 1999. ASM Metals Handbook. ASM International.