

## Original Research Article

# Research on Parametric Design Technology of cycloidal pin wheel Reducer for Robot based on SolidWorks

## ABSTRACT

SolidWorks software and VB.net programming language are used to accurately design the cycloidal wheel of cycloidal pin wheel reducer for robot and to automatically draw the three-dimensional model of cycloidal wheel, which is conducive to the preservation and modification of the drawings. In batch design, the design model can be automatically updated by modifying the parameters of cycloidal wheel 3D model or drawing, which can reduce a lot of work for designers.

*Keywords: SolidWorks; Parameterization; Cycloidal wheel; Two-dimensional graph*

## 1. INTRODUCTION

Robot cycloidal pin wheel (Japan called RV) reducer is developed by the Japanese Emperor company in 1986, and the first application in industrial robot joints on a new type of high precision reduction transmission mechanism. Robot cycloid reducer adopts the combination form of involute gear planetary transmission and cycloid pin gear planetary transmission in structure, which has the advantages of compact structure, small volume, light weight, high transmission accuracy and efficiency, and large transmission ratio range [1]. As an important part of industrial robot, the end positioning accuracy of industrial robot depends on the transmission accuracy of cycloidal reducer with robot to a great extent, and as the core parts of cycloidal reducer with robot, the design of cycloidal wheel is more important. As the key parameters of the cycloid wheel are highly coupled with the target of the designed product, the result makes the calculation in the design process huge, and the relevant parameters need to be constantly adjusted to achieve the rated transmission target [2], so the parametric design becomes an important subject. In this paper, VB.net language and SolidWorks secondary development technology are used to complete the parametric design of cycloidal wheel and realize the two-dimensional drawing of cycloidal wheel.

Comment [H1]: Incomplete sentence

Comment [H2]: Separate

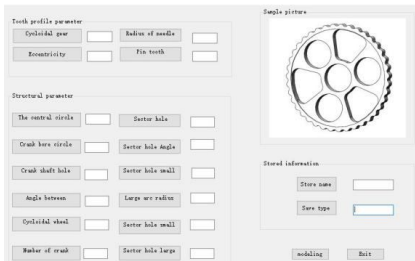
## 2. PARAMETRIC DESIGN OF CYCLOIDAL WHEEL

### 2.1 Parametric design operation interface of cycloidal wheel

The most important part of cycloidal pin wheel reducer for robot is cycloidal wheel, and most current researches are also on cycloidal wheel [3]. The parametric design interface of cycloidal wheel is divided into four parts: tooth shape parameter, structure parameter, sample picture and storage information. The tooth profile parameter is to set the teeth of the cycloidal wheel, the structural parameter is to set the overall structure and internal hole structure of the cycloidal wheel, and the stored information can be used to name the cycloidal wheel and set the saving path. After the correct input of the parameters, the three-dimensional model of

Comment [H3]: Separate; and others

cycloidal wheel can be automatically created through SolidWorks by clicking the modeling button, and the program can be finished by clicking the exit program button, as shown in Figure 1.

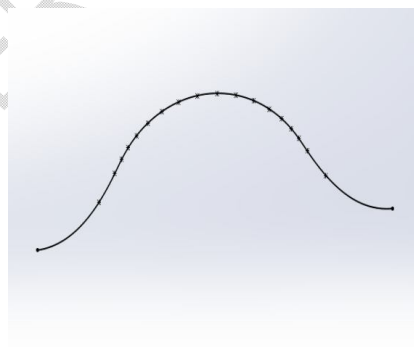


**Fig 1 Parametric design and operation interface of cycloidal wheel**

## 2.2 Cycloidal wheel parametric design program

Cycloidal gear is the core part of cycloidal reducer used by robot. The shape of ectoid directly determines the performance of cycloidal reducer used by robot. When cycloidal gear engages with standard needle teeth, the tooth shape is standard tooth shape, and Its equation is shown as formula 2-1[4].Where:  $r_p$  is the radius of the center circle of the needle tooth, mm;  $r_p$  is the radius of needle teeth, mm;  $a$  is eccentricity, mm;  $iH$  is the transmission ratio between cycloidal pin wheel and pin teeth;  $K_1$  is the short amplitude coefficient;  $\varphi$  is the meshing phase Angle,  $^\circ$

When modeling manually, you can directly draw cycloidal Outlines by entering formulas in SolidWorks. But with the program control, it will be too many parameters to achieve. To solve this problem, we can draw 20 coordinate points in SolidWorks by the equation, and then simulate the single tooth shape of cycloidal wheel through the coordinate points, as shown in Figure 2.



**Fig 2 Single gear profile fitting of cycloid wheel**

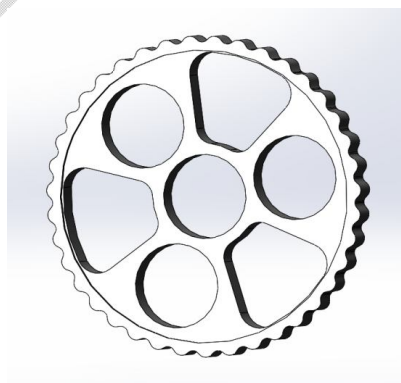
Relevant core code is as follows:  
DimSwappAsSldWorks.SldWorks  
DimPartAsSldWorks.ModelDoc2

```
Swapp=CreateObject("Sldworks.application")
Part=Swapp.ActiveDoc
```

This is a reference for creating Sldworks objects. Since objects and methods modeled later have a parent-child relationship with Sldworks objects, Sldworks objects can be referenced only after they are referenced.

$$\begin{cases} x_c = [r_p - r_p \phi^{-1}(K_1, \varphi)] \cos(1 - i^H) \varphi + [a - K_1 r_p \phi^{-1}(K_1, \varphi)] \cos i^H \varphi \\ y_c = [r_p - r_p \phi^{-1}(K_1, \varphi)] \sin(1 - i^H) \varphi - [a - K_1 r_p \phi^{-1}(K_1, \varphi)] \cos i^H \varphi \end{cases} \quad 2-1$$

```
Dim u, t, i As Double
Dim X(20) As Double
Dim Y(20) As Double
u = 2 * pi / 19
t = 0
For i = 0 To 19
If t <= 2 * pi Then
X(i)=(Rz-rz*(1+K1^2-2*K1*Cos(t))^(1/2))*Sin((1-(Zz/Zc))*t)+(A-K1*rz*(1 +K1^2-2 * K1*
Cos(t))^(0.5))*Sin(Zz/Zc*t)
Y(i)=(Rz-rz*(1+K1^2-2*K1*Cos(t))^(1/2))*Cos((1-(Zz/Zc))*t)-(A-K1*rz*(1+K1^2-
2*K1*Cos(t))^(0.5))*Cos(Zz/Zc*t)
t=t+u
End If
Next i
DimskSegment As SldWorks.SketchSegment
Part.SketchSpline(19,0.001*X(0),0.001*Y(0), 0)
.....
Part.SketchSpline(0,0.001*X(19), 0.001*Y(19), 0)
Through the linkage of code and solidworks, a single tooth shape spline curve of cycloidal
wheel is synthesized, and then the overall drawing of three-dimensional modeling of
cycloidal wheel is completed by creating gear teeth, array gear teeth, stretching and cutting,
as shown in Figure 3.
```



**Fig 3 SolidWorks automatically draws cycloidal wheel models**

The relevant code is as follows:

```

Part.SketchManager.CreateArc(0, 0, 0, 0, 0.001 * Y(0), 0, 0.001 * X(19), 0.001 * Y(19),
0, -1)
Part.Extension.SelectByID2("Sketch 1", "SKETCHREGION", 0,0, 0, True, 4,Nothing, 0)
Part.FeatureManager.FeatureExtrusion2(True,False,False,0,0,T/1000,T/1000,False, Fal
se,False,False,0,0,False,False, False,False,True,True,True,0,0,False)
Part.Extension.SelectByID2("Boss-Extrude1", "BODYFEATURE", 0,0,0,False,4,Nothing,
0)
Part.FeatureManager.FeatureCircularPattern4(Zb, 4*Pi/2, False, "NULL", False,True,
False)
Part.Extension.SelectByID2("Front", "PLANE", 0, 0, 0, False, 0, Nothing,0)'
Part.SketchManager.CreateCircle(0, 0, 0, d1 / 2 / 1000, 0, 0)
Part.SketchManager.CreateCircleByRadius(d3/2/1000*Sin(a1),-
d3/2/1000*Cos(a1),0,d2/2/1000)
Part.Extension.SelectByID2("Sketch 2", "SKETCHSEGMENT", 0,0,0,False,0,Nothing, 0)
Part.FeatureManager.FeatureCut3(True, False, True, 1, 0, 0 0, False, False,
False, False, 0, 0, False, False, False,False,False,True,True,True,True,False,0,0,False)
.....

```

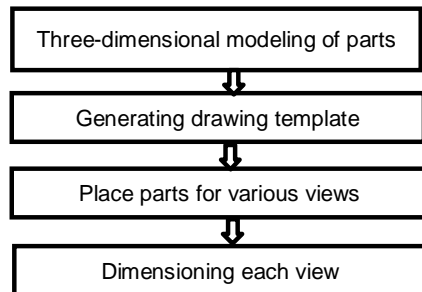
**3. SECONDARY DEVELOPMENT AND DESIGN OF CYCLOIDAL WIRE WHEEL TWO-DIMENSIONAL DIAGRAM**

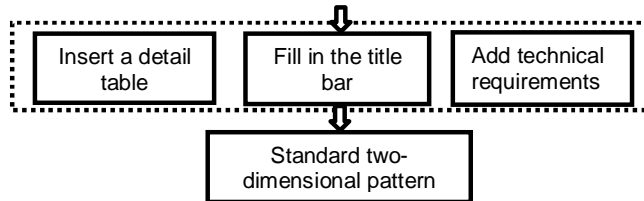
Two-dimensional drawings of mechanical parts generally go through the following five steps: (1) three-dimensional modeling of parts;(2) Create and generate drawing templates;(3) Place all kinds of views of parts;(4) Mark the size of each view;(5) Insert the list, fill in the title bar, etc.At present, various SolidWorks secondary development technologies are widely used in various fields of machinery manufacturing, such as automotive, aviation and other fields [5]. The two-dimensional drawing process of SolidWorks secondary development is shown in Figure 4:

```

Else
Part=Swapp.NewDocument("C:\ProgramData\SolidWorks\SOLIDWORKS
2016\templates\1a4-7.drwdot", 0, 0, 0)
End If

```





**Fig 4 Engineering drawing process of secondary development**

### 3.1 Cycloidal wheel parametric design program

Before writing 2D graphics programs through VB.NET, add references to CommandsTypeLibrary, ConstantTypeLibrary, and TypeLibrary to link SolidWorks to VisualStudio. Then write a program to create a two-dimensional graph document. Before creating a cycloidal wheel two-dimensional diagram, you need to first invoke the created two-dimensional diagram template. The result is shown in Figure 5, and its program code is as follows:

```

Dim myfilename As String
myfilename="C:\ProgramData\SolidWorks\SOLIDWORKS2016\templates\A4-7.drwdot"
If Dir(myfilename) = "" Then
MsgBox("File does not exist")
  
```



**Fig 5 Engineering drawing template**

After the 2D drawing template is created, the main view and section view of parts need to be created. When creating the section view, a section line needs to be created first, and then the corresponding section view is generated. The relevant program code is as follows:

```

Dim myView As Object
myView=Part.CreateDrawViewFromModelView3("C:\Users\Administrator\Desktop\parts
8.SLDPRT", "*Forward vision", 0.088, 0.129, 0)
Part = Swapp.ActiveDoc
Part.ActivateView("Two-dimensional graph view 1")
Dim skSegment As Object
skSegment=Part.SketchManager.CreateLine(0#, 0.088, 0#, 0#, -0.091, 0#)
Part.Extension.SelectByID2("Line1", "SKETCHSEGMENT", 0, 0, 0, False, 0, Nothing, 0)
Dim excludedComponents As Object
excludedComponents = Nothing
myView=Part.CreateSectionViewAt5(0.194,0.129,0,"A", 0,(excludedComponents), 0)
  
```

```
Part.ClearSelection2(True)
```

Dimension annotation, taking the cycloid wheel thickness dimension of 13.6mm as an example, SelectByID2 command first to select two sides, and then AddHorizontalDimension2 command to make annotations. Finally, dimension tolerances are added by the EditDimensionProperties2 command. The associated program code is shown below:

```
Part.ActivateView("Two-dimensional diagram view 2")
Part.Extension.SelectByID2("", "EDGE",0.194-L1/2/2/1000,0.129,-499.96002351577, False,
0, Nothing, 0)
Part.Extension.SelectByID2("", "EDGE",0.194+L1/2/2/1000,0.129,-499.96002351577, True, 0,
Nothing, 0)
Part.AddHorizontalDimension2(0.205202026460628, 0.0842934923041332, 0)
Part.EditDimensionProperties2(2, 0, -0.00002, "", "", True, 9, 2, True, 12, 12, "", "", True, "",
"", False)
Part.ClearSelection2 (True)
```

Add a comment to insert text through the InsertNote command, set the text Angle by the myNote Angle property, and set the text style and size by the SetBalloon property. To set the lead of the text using the SetLeader3 property of myAnnotation, the core code is as follows:

```
myAnnotation=mySFSymbol.GetAnnotation()
Dim myNote As Object
Dim myTextFormat As Object
myTextFormat = Nothing
myNote=Part.InsertNote("Uniform distribution")
If Not myNote Is Nothing Then
myNote.LockPosition = False
myNote.Angle = 0
myNote.SetBalloon(0, 0)
myAnnotation = myNote.GetAnnotation()
If Not myAnnotation Is Nothing Then
myAnnotation.SetLeader3(SwConst.swLeaderStyle_e.swNO_LEADER, 0, True,
False, False, False)
myAnnotation.SetPosition(0.121954231993969, 0.0963539281900556,
0)
End If
End If
Part.ClearSelection2 (True)
```

To insert the specification table, you can use the InsertTableAnnotation2 command to set the coordinate position of the specification table insertion, set the number of rows and columns of the specification table, and obtain the template path. The BorderLineWeight command is used to set the boundary width, and the GridLineWeight command is used to set the grid line width. The code is as follows:

```
Dim swTableAnnotation As Object
swTableAnnotation=Part.InsertTableAnnotation2(False, 0.229347788134974,
0.136902169659606, 1, "C:\Users\Administrator\Desktop\Specification template.sldtbt", 20,
3)
If Not swTableAnnotation Is Nothing Then
swTableAnnotation.BorderLineWeight = 0
```

