DotDyn

Xie Yu

1 介绍

RotDyn类主要用于处理单个轴的转子动力学问题，通常需要借助Commonshaft类一块使用。

2 原理

2.1 转动动力学方程[1]

考虑转动效应的动力方程如下:

$$[M]\{\ddot{u}\}+([G]+[C])\{\dot{u}\}+([K]-[K_c])\{u\}=\{F\} \tag{1}$$

式中: $[M],[C],[K]$一一分别为结构质量、阻尼和刚度矩阵;

$[K_c]$一结构转动引起的旋转软化矩阵，它使旋转坐标系下的结构刚度降低; $[G]$一结构转动对“阻尼”矩阵的贡献，在旋转坐标系下通常称为科氏矩阵或科氏阻尼矩阵，而在固定坐标系下称为回转矩阵或回转阻尼矩阵; $[F]$一—固定坐标系下的外荷载向量。在旋转坐标系下，为外荷载向量与科氏力向量之和，科氏力向量$\langle F_c\rangle=[G]\langle\dot{u}\rangle_0$, $\langle\dot{u}\rangle_0$为命令IC定义的节点速度向量。

从上式中，不难看出影响一个结构的响应主要影响在于**质量**、**阻尼**、**刚度**，如何获得准确的结构的相关信息，是得到准确计算的关键。

在旋转坐标系下，转动效应称为科氏效应，即生成科氏矩阵和科氏力。科氏矩阵通过转速和单元形函数求得各单元的科氏矩阵，然后累加形成结构的科氏矩阵。科氏力则为科氏矩阵与节点速度之积。

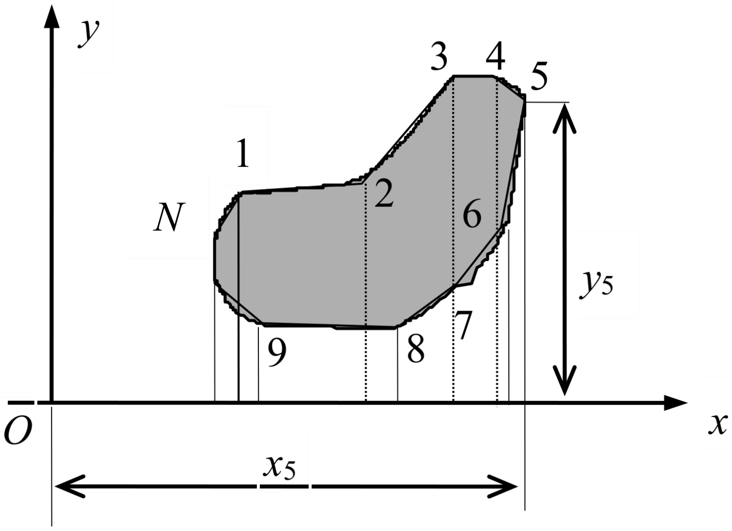
在固定坐标系下，转动效应称为回转效应，即生成回转矩阵，它通过单元回转动能和单元形函数求得。在转子动力学中，将回转效应分为两部分，即与转速相关的回转矩阵和与位移相关的回转矩阵，只是动力学方程表达形式不同而已。

在ANSYS中，无论考虑回转效应或是考虑科氏效应均采用命令CORIOLIS定义。

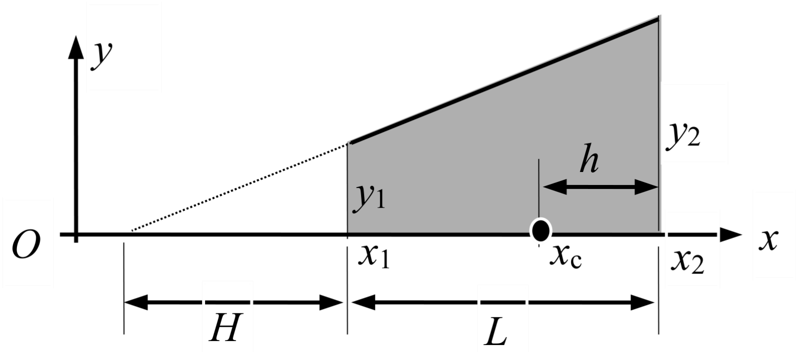
2.1.1 质量参数[2]

首先结构的质量参数，应该在网格构建完成后便知晓，下文介绍了如何通过网格来计算结构的质量，转动惯量等参数。

设有一个旋转对称体，Ox 轴为其旋转对称轴。 图 1 为通过旋转轴 Ox 的半平面截取此物体得到的截面轮廓图。我们在轮廓线上取 N 个点, 使得连接这些点形成的封闭折线，尽可能地逼近轮廓线。从这些点中任选一点为起点，按顺时针方向绕轮廓线顺次把这些点编号为 1、2，直到 N。



折线中的每一线段，以Ox为轴旋转一周，构成一个截头圆锥台。我们先分别计算每一圆锥台的惯性参量；然后，再按一定的规则，把各个圆锥台的惯性参数综合在一起，就得到整个旋转体的惯性参量。



一个圆锥台是计算惯性参数中的基本单元，设其母线两端点的坐标为$x1$,$y1$和$x2$,$y2$。由理论力学中知识，不难得到，该圆锥台的惯性参量为:

质量:

$$M = \frac{1}{3}\pi\rho L y_1^2 \frac{\alpha^3 - 1}{\alpha - 1} \tag{2}$$

质量中心:

$$x_c = x_2 - \frac{L}{4}\frac{\alpha^2 + 2\alpha + 3}{\alpha^2 + \alpha + 1} \tag{3}$$

绕x轴的转动惯量:

$$J_x = \frac{1}{10}\pi\rho L y_1^4 \frac{\alpha^5 - 1}{\alpha - 1} \tag{4}$$

绕中心直径的转动惯量:

$$J_{xc} = \pi\rho(L+H)y_2^2[\frac{y_2^2}{20} + \frac{(L+H)^2}{80} + \frac{(\frac{L+H}{4} - h)^2}{3}] - \pi\rho H y_1^2[\frac{y_1^2}{20} + \frac{H^2}{80} + \frac{(L + \frac{H}{4} - h)^2}{3}] \tag{5}$$

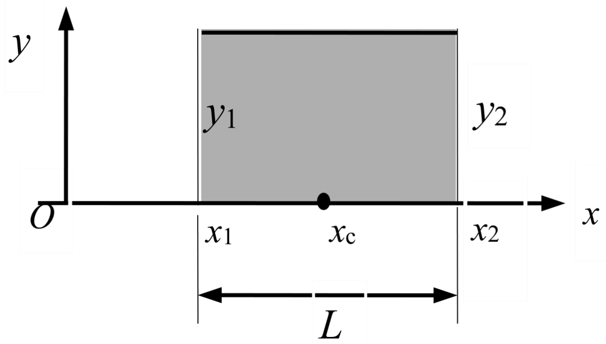
式中,

ρ — 密度,

α — y_2/y_1,

$h — \frac{L}{4}\frac{\alpha^2+2\alpha+3}{\alpha^2+\alpha+1}$,

圆柱是一个特殊的圆锥台，有$y1$=$y2$，它相应的惯性参量为



质量:

$$M=\pi\rho Ly_1^2 \tag{6}$$

质量中心:

$$x_c=0.5(x_1+x_2) \tag{7}$$

绕x轴的转动惯量:

$$J_x=\frac{1}{2}\pi\rho Ly_1^4 \tag{8}$$

绕中心直径的转动惯量:

$$J_{xc}=0.25\pi\rho Ly_1^2(y_1^2+\frac{L^2}{3}) \tag{9}$$

对于编程，首先，我们可以得到任意形状的截面网格，从而通过上述方法得到结构的转动惯量参数。

- 整体的质量和极转动惯量取各个圆锥台的相应量的代数和;
- 将各圆锥台质心，平行力系合成规则得到整体的质心;
- 用移轴定理求出各个圆锥台对过公共质心的直径转动惯量，然后代数相加得整体的直径转动惯量。

以一个半径50mm的钢球举例($\rho=7850kg/m^3$) , 其理论计算的公式如下 :

质量:

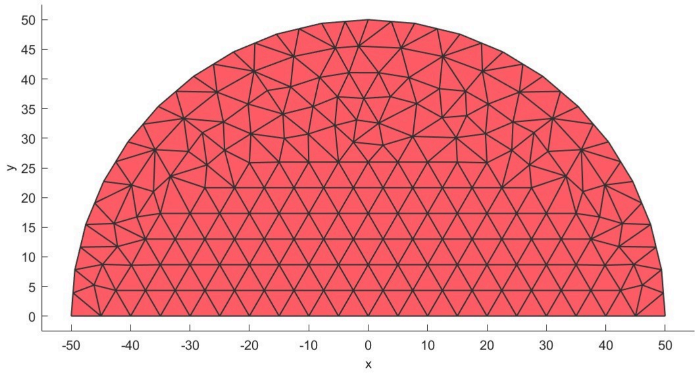
$$M=\frac{4}{3}\pi\rho R^3=4.11kg \tag{10}$$

绕x轴的转动惯量:

$$J_x=\frac{2}{5}MR^2=4.11kg\cdot m^2 \tag{11}$$

绕y轴的转动惯量:

$$J_y=\frac{2}{5}MR^2=4.11kg\cdot m^2 \tag{12}$$



理论计算和程序计算对比结果如下:

Items	Theory	Program	Error
Mass	4.11	4.06	1.02%
J_x	4.11	4.06	1.02%
J_y	4.11	4.06	1.02%

有一个半径50 mm， 长度200mm的轴体,其理论计算公式如下:

质量:

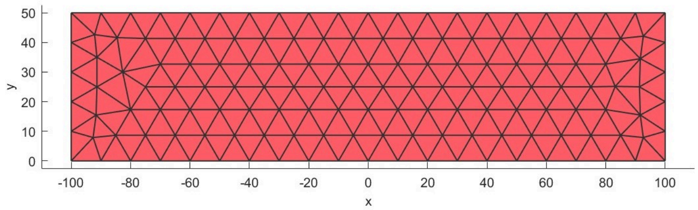
$$M = \pi \rho R^2 L = 12.3kg \tag{13}$$

绕x轴的转动惯量:

$$J_x = \frac{1}{2} M R^2 = 15.374t \cdot mm^2 \tag{14}$$

绕y轴的转动惯量:

$$J_y = \frac{1}{4} M R^2 + \frac{1}{12} M L^2 = 48.685kg \cdot m^2 \tag{15}$$



理论计算和程序计算如下，两者基本一致。

Items	Theory	Program	Error
Mass	12.3	12.3	0%
J_x	15.374	15.374	0%
J_y	48.685	48.685	0%

2.1.2 阻尼参数

在ANSYS中可使用7种阻尼，即Rayleigh阻尼、材料阻尼、常阻尼比、模态阻尼、单元阻尼、材料常阻尼系数、材料结构阻尼系数。在ANSYS结构动力分析中，仅在VT法谐响应分析中可采用滞变阻尼模型，库仑阻尼模型可用于瞬态动力分析，其他均采用黏性阻尼模型。在ANSYS中可同时指定多种阻尼，程序自动形成阻尼矩阵，但不是所有阻尼均可用于各种动力分析。各种分析类型可用阻尼见下表。从表中可知，动力分析类型不同，其所能够定义的阻尼类型也不同，而不同的阻尼类型在ANSYS中的输人命令亦不相同，因此在定义阻尼时应引起足够的重视。

分析类型		α、β阻尼	材料阻尼	常阻尼比	模态阻尼	单元阻尼②	材料常阻尼系数	材料结构阻尼系数
命令		ALPHAD BETAD	MP，DAMP	DMPRAT	MDAMP		MP，DMPR	TB，SDAMP
静态分析		/	/	/	/	/	/	/
模态分析	无阻尼	—	—③	—	—	—	—	/
	有阻尼	√	√	—	—	√	—	/
谐响应分析	完全法	√	√	√⑤	—	√	√⑤	√⑥
	缩减法	√	√	√⑤	—	√	—	/
	模态叠加法	√	√③	√	√	—	—	/
	QRDAMP 模态叠加法	√④	√④	√	√	√	√⑤	/
瞬态分析	完全法	√	√	—	—	√	—	/
	缩减法	√	√	—	—	√	—	/
	模态叠加法	√	√③	√	√	—	—	/
	QRDAMP 模态叠加法	√④	√④	√	√	√	—	/
谱分析	SPRS，MPRS	√①	√③	√	√	—	—	/
	DDAM	√①	/	/	/	/	/	/
	PSD	√	√③	√	√	—	—	/
子结构分析		√	√	—	—	√	—	/
屈曲分析		/	/	/	/	/	/	/

注：1.表中“/”表示不需要，“”不考虑，“√”可考虑。2.①仅可用β阻尼，,α阻尼不可用；②包括超单元阻尼矩阵；③“MP，DAMP"定义等效材料阻尼比，用于后续的谱分析或模态叠加法分析；④在模态叠加法中，必须采用QRDAMP法提取模态才可考虑α阻尼$\mathcal{S}$阻尼和材料阻尼（MP，MADP）；⑤在谐响应分析中，命令DMPRAT和“MP，DMPR"定义的是结构阻尼比，而非模态阻尼比；⑥必须用命令“TB，ELAS"定义材料性质。

- 在 Baffalo RotDyn中设置params.Damping为常阻尼比。

  常阻尼比（ConstantDampingRatio）是定义结构阻尼的最简单方法，也就是通常简称的阻尼比ξ一一实际阻尼与临界阻尼之比，采用命令DMPRAT定义。常阻尼比仅适用于谐响应分析、谱分析和模态叠加法瞬态动力分析。

- 对于材料阻尼DAMP、材料常阻尼系数DMPR和材料结构阻尼系数SDAMP可在对应的材料里设置。

  材料常阻尼系数（ConstantMaterialDampingCoefficient）仅适用于谐响应分析的完全法和QRDAMP模态叠加法，采用命令“MP，DMPR"定义。

  材料结构阻尼系数是频变阻尼，采用命令“TB，SDAMP"和TBFIELD定义，仅用于变分法谐响应分析，并可选择黏性阻尼模型或滞变阻尼模型。

- 针对单元阻尼，可以在Support中设置，RotDyn会将其转换为combin14单元和combin214单元来计算。

  Support'% Spring connected to ground

  % Node number,kx,K11,K22,K12,K21,Cx,C11,C22,C12,C21

2.1.3 刚度参数

部件的刚度比如齿轮、轴承、联轴器均有相关的规范和计算方法，可以参考其他部件Baffalo说明。

3 类结构

4 案例

4.2 Shaft modal analysis (Flag=1)

下图为一个等截面转子，长 l 为 1m，圆形截面的直径 d=0.04m，两端为简支支承。转子材料的质量密度 为 7800kg/m 3 ，弹性模量 E 为 2.0×10 11 N/m 2 。求此转子在静止状态下的固有频率。

此转子的各阶固有圆频率 ωi 有理论解如下：

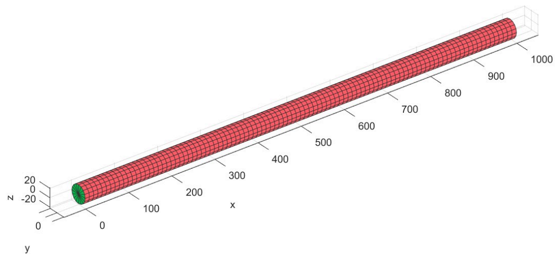
$$\omega_i = i^2\pi^2\sqrt{\frac{EJ}{\gamma Al^4}} \tag{16}$$

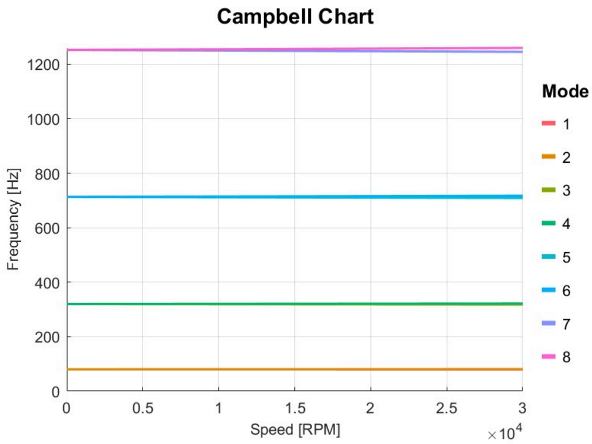
以截面惯性矩 $J = \pi d^4/64$，截面积 $A = \pi d^2/4$，固有频率 $f = \omega/2\pi$ 代入，得到转子的各阶固有频率 f_i为

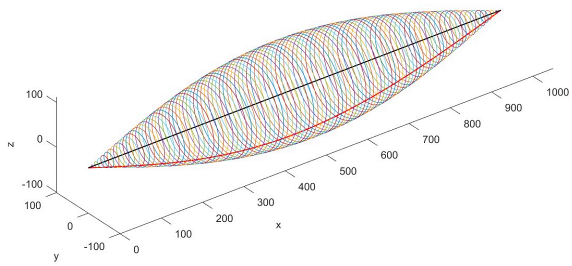
$$f_i = \frac{i^2\pi}{8}\frac{d}{l^2}\sqrt{\frac{E}{\gamma}} \tag{17}$$

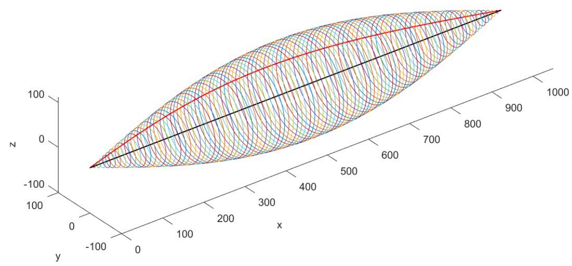
代入具体数据，得到前 3 阶固有频率为 79.5403、318.1612、715.8627Hz，相应的振型为半个周期、一个周期和一个半周期的正弦曲线。

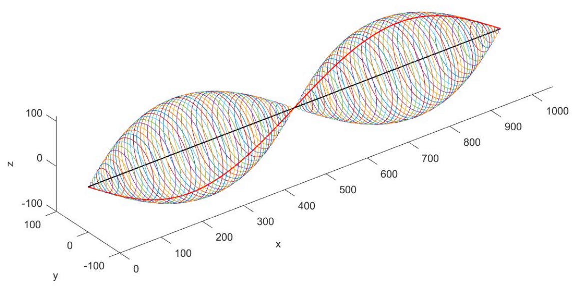
```
% Shaft 1
inputshaft1.Length = 1000;
inputshaft1.ID = [0,0];
inputshaft1.OD = [40,40];
paramsshaft1 = struct();
obj1 = shaft.Commonshaft(paramsshaft1, inputshaft1);
obj1 = obj1.solve();
mat{1,1}=obj1.params.Material;
inputRotDyn.Shaft=obj1.output.BeamMesh;
inputRotDyn.Speed=[0,15000,30000];
inputRotDyn.MaterialNum=1;
inputRotDyn.BCNode=[1,1,1,1,0,0,0;size(obj1.output.Node,1),1,1,1,0,0,0];
paramsRotDyn.Material=mat;
paramsRotDyn.PrintCampbell=1;
paramsRotDyn.PrintORB=1;
obj2 = solve.RotDyn(paramsRotDyn,inputRotDyn);
Plot(obj2);
obj2 = obj2.solve();
ANSYSSolve(obj2.output.Assembly)
PlotCampbell(obj2,'NMode',8);
for i=1:10
    PlotORB(obj2,2,i,'scale',10)
end
```

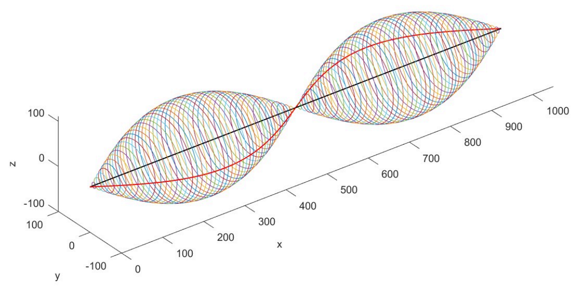

20
0
-20
z
0
y
0
100
200
300
400
500
600
700
800
900
1000
x

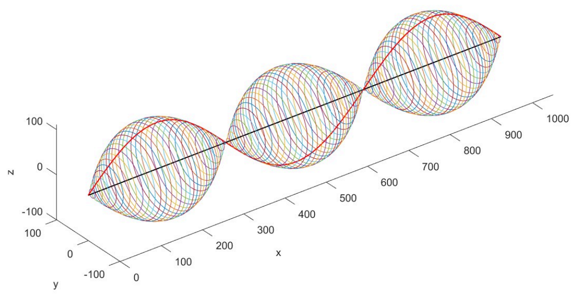

Campbell Chart
Frequency [Hz]
0
200
400
600
800
1000
1200
0
0.5
1
1.5
2
2.5
3
Speed [RPM]
×10^4
Mode
1
2
3
4
5
6
7
8


100
0
-100
z
100
0
-100
y
0
100
200
300
400
500
600
700
800
900
1000
x

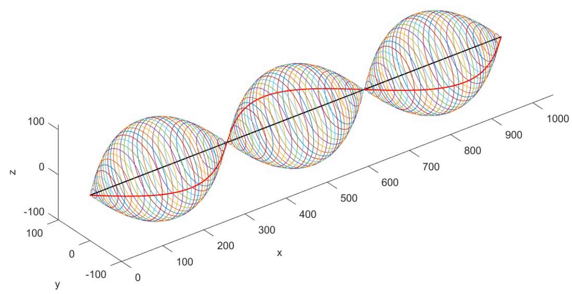

100
0
-100
z
100
0
-100
y
0
100
200
300
400
500
600
700
800
900
1000
x


100
0
-100
z
100
0
-100
y
0
100
200
300
400
500
600
700
800
900
1000
x

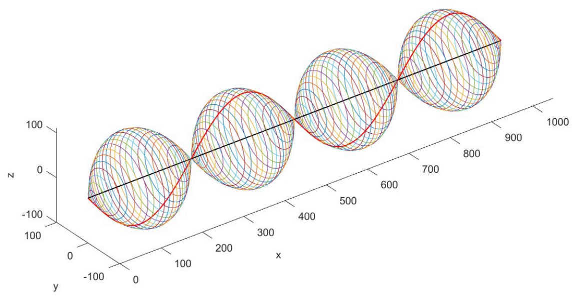

100
0
-100
z
100
0
-100
y
0
100
200
300
400
500
600
700
800
900
1000
x



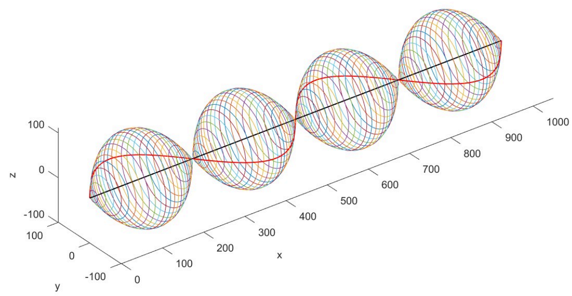
100
0
-100
z
100
0
-100
y
0
100
200
300
400
500
600
700
800
900
1000
x



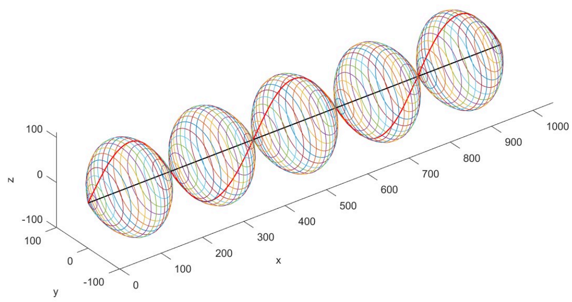
100
0
-100
z
100
0
-100
y
0
100
200
300
400
500
600
700
800
900
1000
x



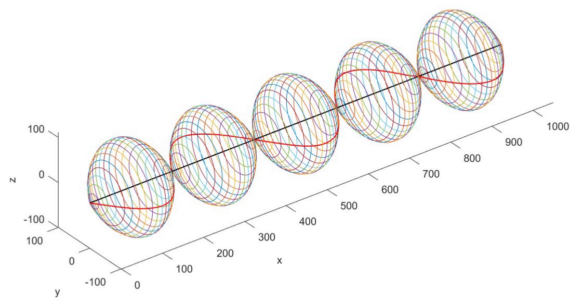
100
0
-100
z
100
0
-100
y
0
100
200
300
400
500
600
700
800
900
1000
x



100
0
-100
z
100
0
-100
y
0
100
200
300
400
500
600
700
800
900
1000
x



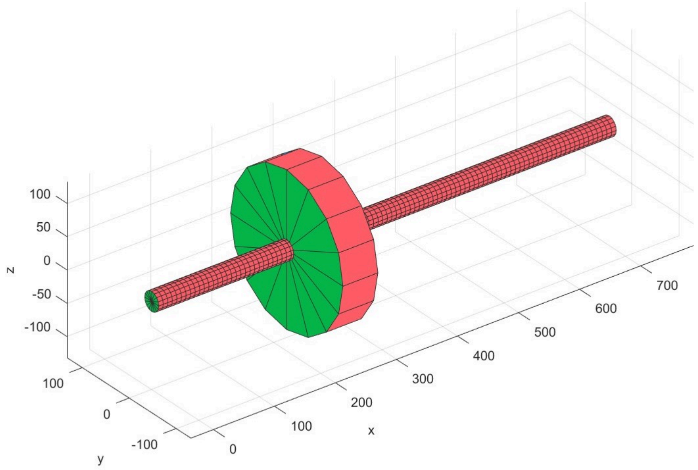
100
0
-100
z
100
0
-100
y
0
100
200
300
400
500
600
700
800
900
1000
x



100
0
-100
z
100
0
-100
y
0
100
200
300
400
500
600
700
800
900
1000
x

4.3 Add Discs (Flag=2)

如图所示的刚性支承单圆盘转子，圆盘的质量$m=20\,\mathrm{kg}$，半径$R=120\mathrm{mm}$,转轴的跨度$l=750\mathrm{mm}$，直径$d=30\mathrm{mm}$。圆盘到左支点的距离$a=l/3=250\mathrm{mm}$。

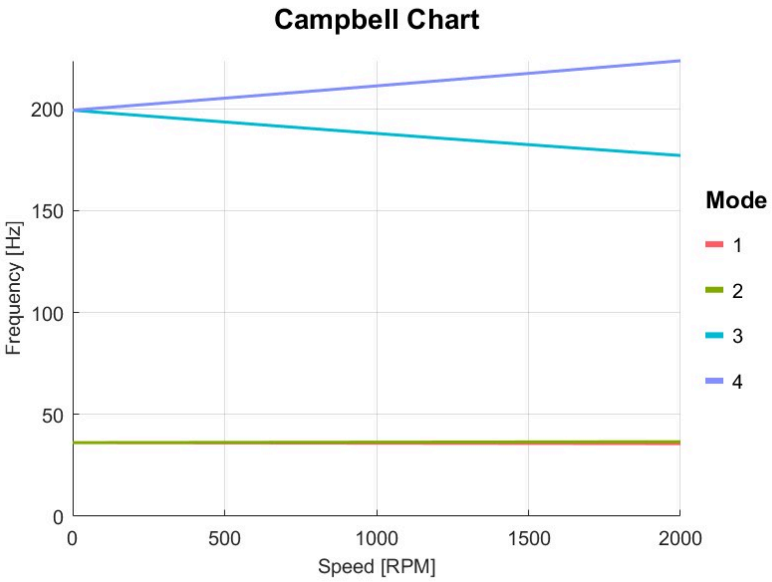


仅考虑轴的弯曲但不计轴的质量，考虑回转效应时的频 率方程为:

$$\omega^4-2\Omega\omega^3-2.1340661\times10^6\omega^2+1.7674781\times10^6\Omega\omega+1.2052387\times10^{11}=0 \tag{18}$$

其中，Ω为转速（即激励），ω为待求涡动频率，两者单位均为rad/s。

```
ID=30;
inputshaft1.Length =750;
inputshaft1.ID = [0,0];
inputshaft1.OD = [ID,ID];
paramsshaft1.Beam_N = 30;
obj1 = shaft.Commonshaft(paramsshaft1, inputshaft1);
obj1 = obj1.solve();
mat{1,1}=obj1.params.Material;
inputRotDyn.Shaft=obj1.output.BeamMesh;
inputRotDyn.Speed=[0,1500,3000];% Unit: RPM
inputRotDyn.MaterialNum=1;
inputRotDyn.BCNode=[1,1,1,1,0,0,0;size(obj1.output.Node,1),1,1,1,0,0,0];
paramsRotDyn.Material=mat;
paramsRotDyn.PrintCampbell=1;
paramsRotDyn.Position=[0,0,0,0,0,0];
Dyn = solve.RotDyn(paramsRotDyn,inputRotDyn);
[Dyn,Num1]= AddCnode(Dyn,250);
DiscMass=20*10^-3;
rou=obj1.params.Material.Dens;
OD=240;
L=DiscMass/((OD^2-ID^2)*pi/4*rou);
Dyn= AddDisc(Dyn,Num1,OD,ID,L,1);
Plot(Dyn);
Dyn = Dyn.solve();
ANSYSSolve(Dyn.output.Assembly)
PlotCampbell(Dyn,'NMode',6);
```



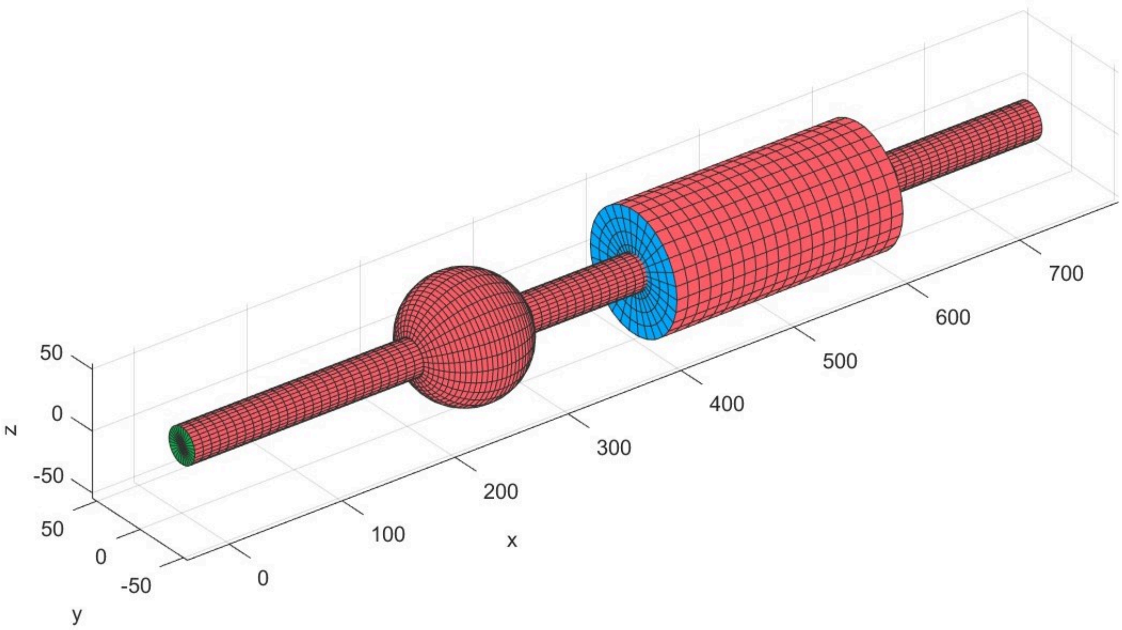
4.4 Add Userdefined Disc, output moment of inertia (Flag=3)

```
OD=30;
inputshaft1.Length =750;
inputshaft1.ID = [0,0];
inputshaft1.OD = [OD,OD];
paramsshaft1.Beam_N = 30;
obj1 = shaft.Commonshaft(paramsshaft1, inputshaft1);
obj1 = obj1.solve();
mat{1,1}=obj1.params.Material;
inputRotDyn.Shaft=obj1.output.BeamMesh;
inputRotDyn.Speed=[0,1500,3000];% Unit: RPM
inputRotDyn.MaterialNum=1;
inputRotDyn.BCNode=[1,1,1,1,0,0,0;size(obj1.output.Node,1),1,1,1,0,0,0];
paramsRotDyn.Material=mat;
paramsRotDyn.PrintCampbell=1;
paramsRotDyn.Position=[0,0,0,0,0,0];
Dyn = solve.RotDyn(paramsRotDyn,inputRotDyn);
[Dyn,Num1]= AddCnode(Dyn,250);
[Dyn,Num2]= AddCnode(Dyn,500);
a1=Point2D('Circle center');
a1=AddPoint(a1,0,0);
a1=AddPoint(a1,[-50;50],[0;0]);
b1=Line2D('Semi circle');
b1=AddCircle(b1,50,a1,1,'ang',180);
b1=AddLine(b1,a1,2);
S1=Surface2D(b1);
a2=Point2D('Square corner');
a2=AddPoint(a2,[-100;-100;100;100],[0;50;50;0]);
b2=Line2D('Square');
b2=AddCurve(b2,a2,1);
S2=Surface2D(b2);
Dyn= AddUserdefinedDisc(Dyn,Num1,S1,1);
Dyn= AddUserdefinedDisc(Dyn,Num2,S2,1);
Plot(Dyn);
disp(Dyn.input.PointMass)
```

加入自定义的截面并计算其质量、转动惯量和弯曲惯量

Successfully revolve to solid mesh .

35.0000	0.0041	4.0578	4.0580
69.0000	0.0123	15.3742	48.6848

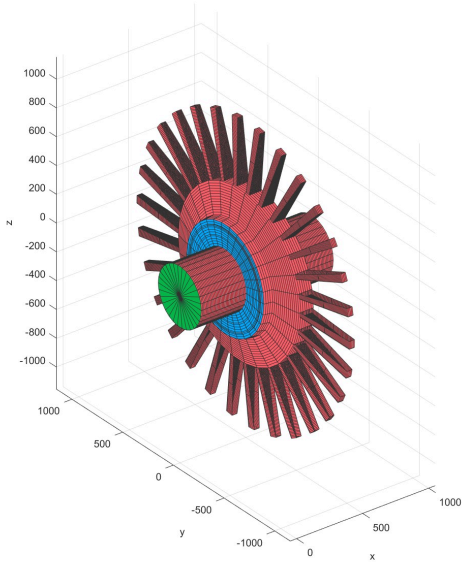


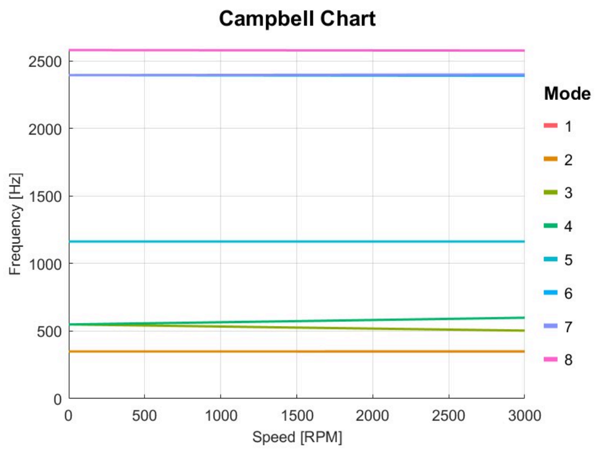
4.5 Add Blade (Flag=4)

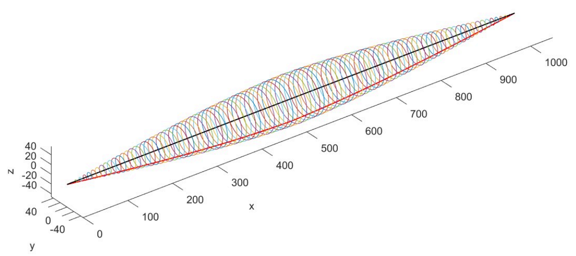
```
OD=425;
inputshaft1.Length =1000;
inputshaft1.ID = [0,0];
inputshaft1.OD = [OD,OD];
paramsshaft1.Beam_N = 30;
obj1 = shaft.Commonshaft(paramsshaft1, inputshaft1);
obj1 = obj1.solve();
mat{1,1}=obj1.params.Material;
inputRotDyn.Shaft=obj1.output.BeamMesh;
inputRotDyn.Speed=[0,1500,3000];% Unit: RPM
inputRotDyn.MaterialNum=1;
inputRotDyn.BCNode=[1,1,1,1,0,0,0;size(obj1.output.Node,1),1,1,1,0,0,0];
paramsRotDyn.Material=mat;
paramsRotDyn.NMode=15;
paramsRotDyn.Freq=[0,3000];
paramsRotDyn.PrintCampbell=1;
paramsRotDyn.PrintORB=1;
paramsRotDyn.Position=[0,0,0,0,0,0];
Dyn = solve.RotDyn(paramsRotDyn,inputRotDyn);
[Dyn,Num1]= AddCnode(Dyn,500);
a=Point2D('Circle center');
a=AddPoint(a,[-160;-160;-155;-155;-95;-55;55;90;155;155;160;160;-160],...
[212.5;350;350;380;390;595;595;385;355;320;320;212.5;212.5]);
a=AddPoint(a,[-45;-20;18;45],...
[595;1050;1050;595]);
b1=Line2D('Section1');
b1=AddCurve(b1,a,1);
S1=Surface2D(b1);
b2=Line2D('Section2');
```

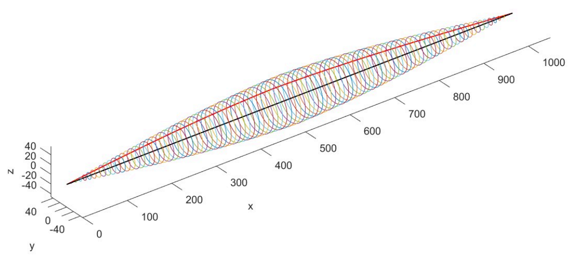
```
b2=AddCurve(b2,a,2);
S2=Surface2D(b2);
Dyn= AddUserdefinedDisc(Dyn,Num1,S1,1);
Dyn= AddBlade(Dyn,Num1,S2,1,30,3);
Plot(Dyn);
Dyn = Dyn.solve();
ANSYSSolve(Dyn.output.Assembly);
PlotCampbell(Dyn,'NMode',8);
for i=1:8
PlotORB(Dyn,2,i,'scale',100)
end
```

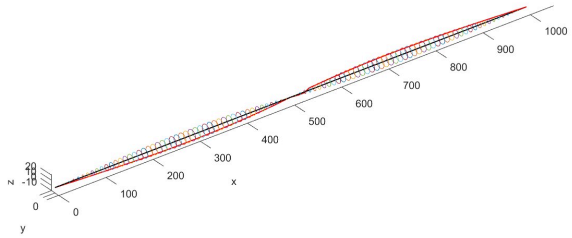
增加叶片，RotDyn会换算叶片的质量的转动惯量到轴中心。



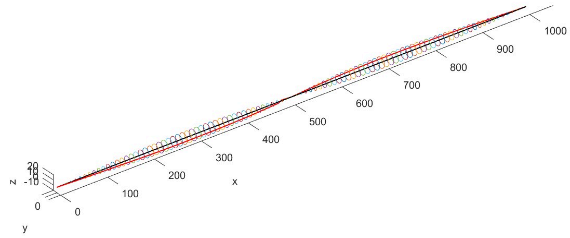




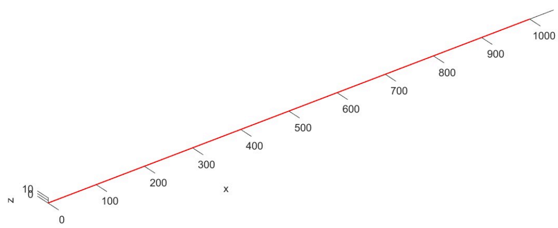




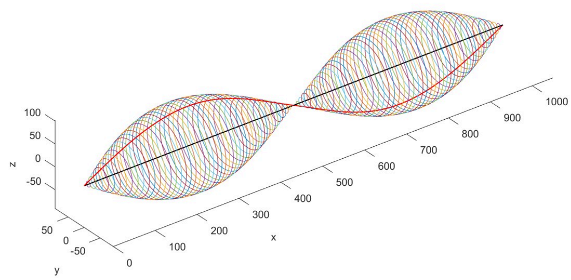
z
20
10
0
-10
x
0
100
200
300
400
500
600
700
800
900
1000
y
0



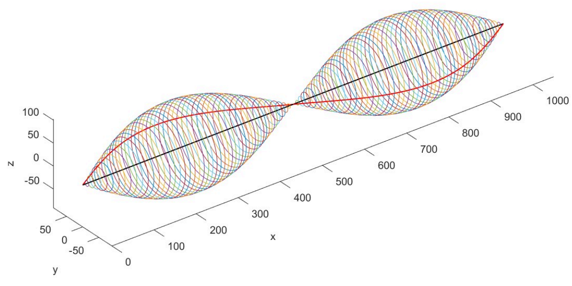
z
20
10
0
-10
x
0
100
200
300
400
500
600
700
800
900
1000
y
0



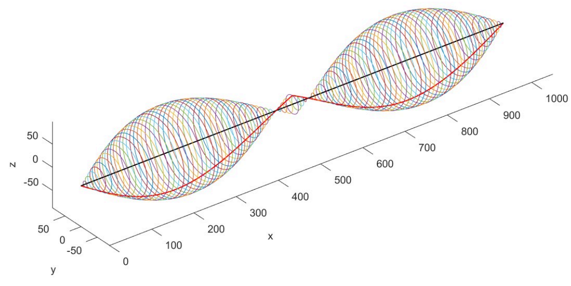
z
10
0
x
0
100
200
300
400
500
600
700
800
900
1000



z
100
50
0
-50
x
0
100
200
300
400
500
600
700
800
900
1000
y
50
0
-50



z
100
50
0
-50
x
0
100
200
300
400
500
600
700
800
900
1000
y
50
0
-50

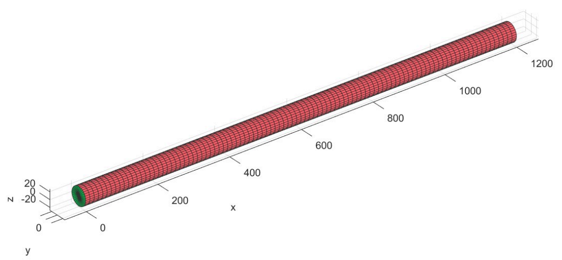


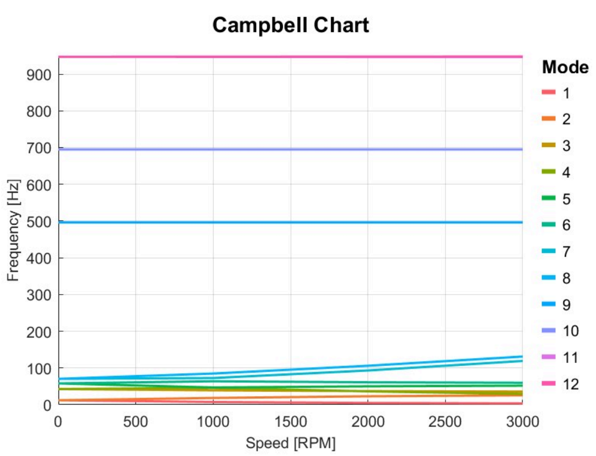
z
50
0
-50
x
0
100
200
300
400
500
600
700
800
900
1000
y
50
0
-50

4.6 Multi Discs & Calculate critical speed (Flag=5)

```
m1=0.102;% unit:ton
JT1=6377*2;% unit: ton/mm2
JD1=6377;% unit: ton/mm2
% Shaft1
OD=50;% Unit: mm
inputshaft1.Length =1200;
inputshaft1.ID = [0,0];
inputshaft1.OD = [OD,OD];
paramsshaft1.Beam_N = 30;
obj1 = shaft.Commonshaft(paramsshaft1, inputshaft1);
obj1 = obj1.solve();
mat{1,1}=obj1.params.Material;
inputRotDyn.Shaft=obj1.output.BeamMesh;
inputRotDyn.Speed=[0,1000,2000,3000];% Unit: RPM
inputRotDyn.MaterialNum=1;
inputRotDyn.BCNode=[1,1,1,1,0,0,0];
paramsRotDyn.Material=mat;
paramsRotDyn.PrintCampbell=1;
Dyn = solve.RotDyn(paramsRotDyn,inputRotDyn);
[Dyn,Num1]= AddCnode(Dyn,400);
Dyn=AddPointMass(Dyn,Num1,m1,JT1,JD1);
[Dyn,Num2]= AddCnode(Dyn,800);
Bound=[1,1,1,0,0,0];
Dyn= AddBCNode(Dyn,Num2,Bound);
[Dyn,Num3]= AddCnode(Dyn,1200);
Dyn=AddPointMass(Dyn,Num3,m1,JT1,JD1);
Plot(Dyn);
Dyn = Dyn.solve();
ANSYSSolve(Dyn.output.Assembly);
PlotCampbell(Dyn,'NMode',12);
Dyn=CalculateCriticalSpeed(Dyn);
disp(Dyn.output.CriticalSpeed);
```

计算临界转速



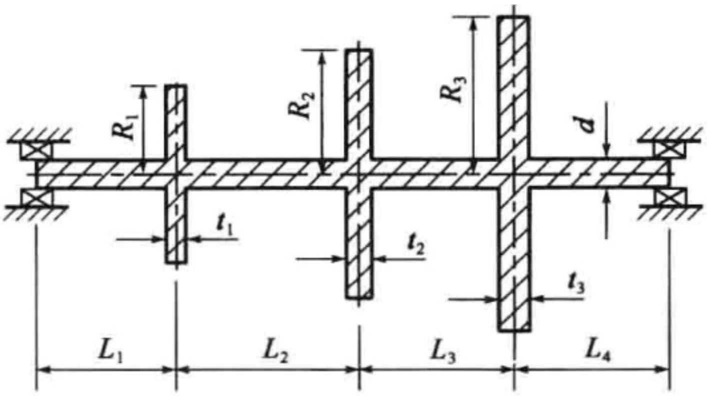


1.0e+03 *

```
0.5661
1.1379
2.2070
2.1237
```

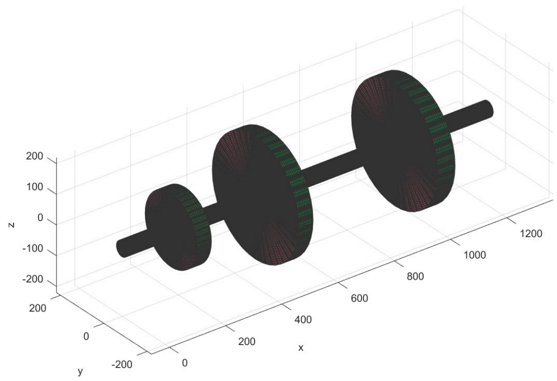
4.7 Add Elastic Support (Flag=6)

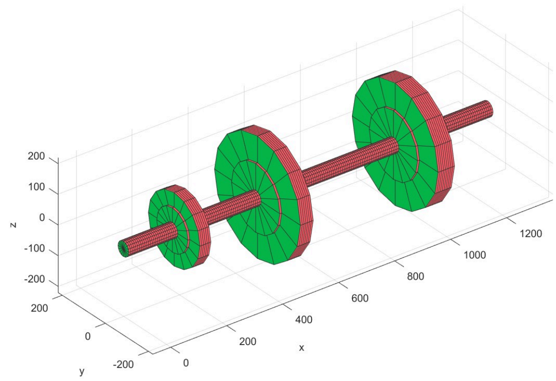
如下图所示钢质圆盘转子结构，材料的弹性模量为200GPa，密度7830 kg/m^3，泊松系数为0.3，长度分别为 $L_1 = 0.2m$，$L_2 = 0.3m$，$L_3 = 0.5m$，$L_4 = 0.3m$，轴径$d = 0.05m$，圆盘半径分别为 $R_1 = 0.12m$ ，$R_2 = 0.2m$ ，$R_3 = 0.2m$,厚 度分别为 $t_1 = 0, 05m$ ，$t_2 = 0.05m$ ，$t_3 = 0.06m$ 。轴两端为简支边界条件，图中支承仅为示意。

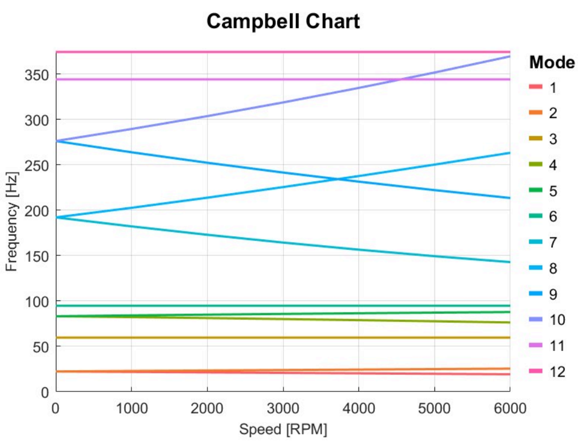


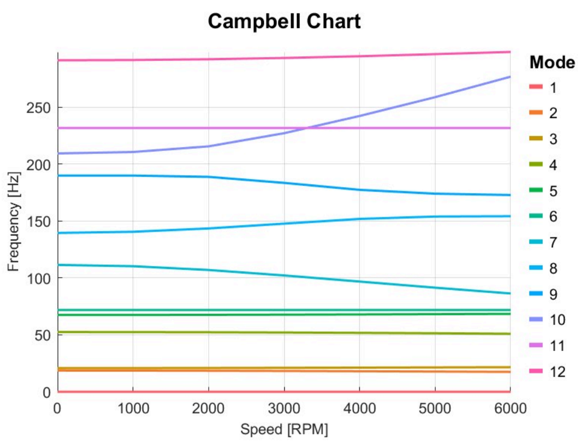

```
L1=200;
L2=300;
L3=500;
L4=300;
t1=50;t2=50;t3=60;
R1=120;
R2=200;
R3=200;
d=50;
% Shaft1
inputshaft1.Length = [L1-t1/2;L1+t1/2;L1+L2-t2/2;L1+L2+t2/2;L1+L2+L3-t3/2;L1+L2+L3+t3/2;L1+L2+L3+L4];
inputshaft1.ID = [[0,0];[0,0];[0,0];[0,0];[0,0];[0,0];[0,0]];
inputshaft1.OD = [[d,d];[R1*2,R1*2];[d,d];[R2*2,R2*2];[d,d];[R3*2,R3*2];[d,d]];
paramsshaft1.Beam_N = 16;
paramsshaft1.N_Slice=201;
obj1 = shaft.Commonshaft(paramsshaft1, inputshaft1);
obj1 = obj1.solve();
Plot3D(obj1)
% Rigid Boundary
mat{1,1}=obj1.params.Material;
inputRotDyn.Shaft=obj1.output.BeamMesh;
inputRotDyn.Speed=[0,1000,2000,3000,4000,5000,6000];
inputRotDyn.MaterialNum=1;
inputRotDyn.BCNode=[1,1,1,1,1,0,0;size(obj1.output.Node,1),0,1,1,1,0,0];
paramsRotDyn.Material=mat;
paramsRotDyn.PrintCampbel=1;
paramsRotDyn.ShaftTorsion=1;
paramsRotDyn.PrintORB=1;
Dyn1 = solve.RotDyn(paramsRotDyn,inputRotDyn);
Dyn1 = Dyn1.solve();
Plot(Dyn1);
```

```
ANSYSSolve(Dyn1.output.Assembly);
PlotCampbell(Dyn1,'NMode',12);
% Elastic support
mat{1,1}=obj1.params.Material;
inputRotDyn.Shaft=obj1.output.BeamMesh;
inputRotDyn.Speed=[0,1000,2000,3000,4000,5000,6000];
inputRotDyn.MaterialNum=1;
inputRotDyn.Support=[1,1e8,8e3,3e3,0,0,0,1e-3,2e-4,0,0;size(obj1.output.Node,1),0,8e3,3e3,0,0,0,1e-3,2e-4,0,0];
paramsRotDyn.Material=mat;
paramsRotDyn.PrintCampbell=1;
paramsRotDyn.ShaftTorsion=1;
paramsRotDyn.PrintORB=1;
inputRotDyn.BCNode=[];
Dyn2 = solve.RotDyn(paramsRotDyn,inputRotDyn);
Dyn2 = Dyn2.solve();
Plot(Dyn2);
ANSYSSolve(Dyn2.output.Assembly);
PlotCampbell(Dyn2,'NMode',12);
```









Num	Status	rpm1	rpm2	rpm3	rpm4	rpm5	rpm6	rpm7

1	BW	22.406	21.884	21.362	20.842	20.325	19.813	19.306
2	FW	22.406	22.927	23.447	23.963	24.476	24.983	25.484
3		59.648	59.648	59.648	59.648	59.648	59.648	59.648
4	BW	83.198	82.229	81.192	80.086	78.909	77.662	76.348
5	FW	83.198	84.104	84.95	85.739	86.477	87.167	87.812
6		94.803	94.803	94.803	94.803	94.803	94.803	94.803
7	BW	192.02	182.11	172.88	164.33	156.47	149.3	142.8
8	FW	192.02	202.57	213.71	225.39	237.56	250.16	263.13
9	BW	276.11	263.73	252.17	241.41	231.38	222.05	213.37
10	FW	276.11	289.36	303.5	318.57	334.58	351.53	369.4
11		343.94	343.94	343.94	343.94	343.94	343.94	343.94
12		374.18	374.18	374.18	374.18	374.18	374.18	374.18

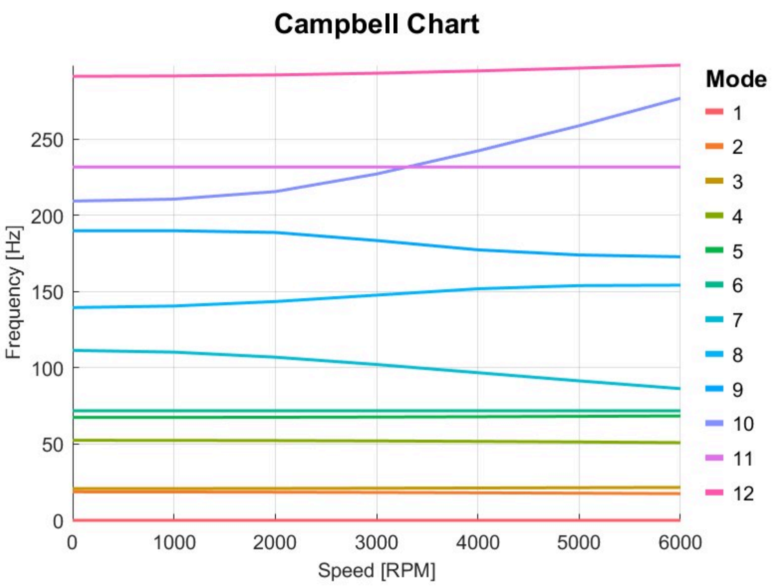
Num	Status	rpm1	rpm2	rpm3	rpm4	rpm5	rpm6	rpm7
1		0	0	0	0	0	0	0
2	BW	18.664	18.619	18.493	18.303	18.068	17.8	17.509
3	FW	20.791	20.826	20.923	21.064	21.232	21.415	21.604
4	BW	52.486	52.442	52.309	52.091	51.792	51.417	50.975
5	FW	67.5	67.53	67.619	67.765	67.962	68.203	68.474
6		71.896	71.896	71.896	71.896	71.896	71.896	71.896
7	BW	111.47	110.32	107.02	102.22	96.835	91.461	86.372
8	FW	139.56	140.57	143.51	147.72	151.88	153.96	154.25
9	FW	189.93	189.95	188.82	183.53	177.41	174.04	172.86
10	FW	209.4	210.66	215.61	227.17	242.31	258.84	276.84
11		231.75	231.75	231.75	231.75	231.75	231.75	231.75
12	BW	291.24	291.46	292.14	293.25	294.77	296.63	298.55

4.8 Consider prestress (Flag=7)

```
L1=200;
L2=300;
L3=500;
L4=300;
t1=50;t2=50;t3=60;
R1=120;
R2=200;
R3=200;
d=50;
% Shaft1
inputshaft1.Length = [L1-t1/2;L1+t1/2;L1+L2-t2/2;L1+L2+t2/2;L1+L2+L3-t3/2;L1+L2+L3+t3/2;L1+L2+L3+L4];
inputshaft1.ID = [[0,0];[0,0];[0,0];[0,0];[0,0];[0,0];[0,0]];
inputshaft1.OD = [[d,d];[R1*2,R1*2];[d,d];[R2*2,R2*2];[d,d];[R3*2,R3*2];[d,d]];
paramsshaft1.Beam_N = 16;
paramsshaft1.N_Slice=201;
obj1 = shaft.Commonshaft(paramsshaft1, inputshaft1);
obj1 = obj1.solve();
Plot3D(obj1)
% Elastic support
```

```
mat{1,1}=obj1.params.Material;
inputRotDyn.Shaft=obj1.output.BeamMesh;
inputRotDyn.Speed=[0,1000,2000,3000,4000,5000,6000];
inputRotDyn.MaterialNum=1;
inputRotDyn.Support=[1,1e8,8e3,3e3,0,0,0,1e-3,2e-4,0,0;size(obj1.output.Node,1),0,8e3,3e3,0,0,0,1e-3,2e-4,0,0];
paramsRotDyn.Material=mat;
paramsRotDyn.PrintCampbell=1;
paramsRotDyn.ShaftTorsion=1;
paramsRotDyn.Material=mat;
paramsRotDyn.PStress=1;
Dyn2 = solve.RotDyn(paramsRotDyn,inputRotDyn);
Dyn2 = Dyn2.solve();
Plot(Dyn2);
ANSYSSolve(Dyn2.output.Assembly);
Dyn2=PlotCampbell(Dyn2,'NMode',12);
disp(Dyn2.output.Campbell);
```

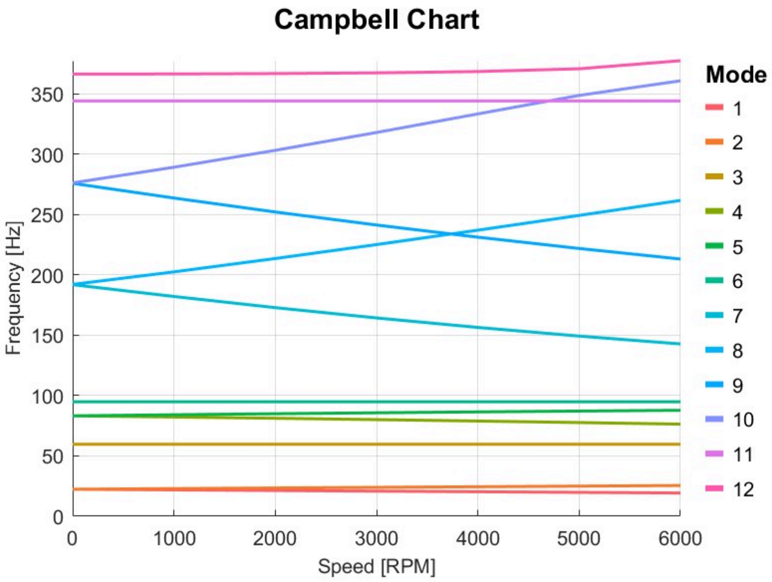
Num	Status	rpm1	rpm2	rpm3	rpm4	rpm5	rpm6	rpm7
1		0	0	0	0	0	0	0
2	BW	18.664	18.619	18.493	18.303	18.068	17.8	17.509
3	FW	20.791	20.826	20.923	21.064	21.232	21.415	21.604
4	BW	52.486	52.442	52.309	52.091	51.792	51.417	50.975
5	FW	67.5	67.53	67.619	67.765	67.962	68.203	68.474
6		71.896	71.896	71.896	71.896	71.896	71.896	71.896
7	BW	111.47	110.32	107.02	102.22	96.835	91.461	86.372
8	FW	139.56	140.57	143.51	147.72	151.88	153.96	154.25
9	FW	189.93	189.95	188.82	183.53	177.41	174.04	172.86
10	FW	209.4	210.66	215.61	227.17	242.31	258.84	276.84
11		231.75	231.75	231.75	231.75	231.75	231.75	231.75
12	BW	291.24	291.46	292.14	293.25	294.77	296.63	298.55



4.9 Eccentricity of shaft (Flag=8)

```
L1=200;
L2=300;
L3=500;
L4=300;
t1=50;t2=50;t3=60;
R1=120;
R2=200;
R3=200;
d=50;
% Shaft1
inputshaft1.Length = [L1-t1/2;L1+t1/2;L1+L2-t2/2;L1+L2+t2/2;L1+L2+L3-
t3/2;L1+L2+L3+t3/2;L1+L2+L3+L4];
inputshaft1.ID = [[0,0];[0,0];[0,0];[0,0];[0,0];[0,0];[0,0]];
inputshaft1.OD = [[d,d];[R1*2,R1*2];[d,d];[R2*2,R2*2];[d,d];[R3*2,R3*2];[d,d]];
paramsshaft1.Beam_N = 16;
paramsshaft1.N_Slice=201;
obj1 = shaft.Commonshaft(paramsshaft1, inputshaft1);
obj1 = obj1.solve();
% Rigid Boundary
mat{1,1}=obj1.params.Material;
inputRotDyn.Shaft=obj1.output.BeamMesh;
inputRotDyn.Speed=[0,1000,2000,3000,4000,5000,6000];
inputRotDyn.MaterialNum=1;
inputRotDyn.BCNode=[1,1,1,1,1,0,0;size(obj1.output.Node,1),0,1,1,1,0,0];
paramsRotDyn.Material=mat;
paramsRotDyn.PrintCampbell=1;
paramsRotDyn.ShaftTorsion=1;
paramsRotDyn.PrintORB=1;
paramsRotDyn.ey=[5,-5];
Dyn1 = solve.RotDyn(paramsRotDyn,inputRotDyn);
Dyn1 = Dyn1.solve();
Plot(Dyn1);
ANSYSSolve(Dyn1.output.Assembly);
Dyn1=PlotCampbell(Dyn1,'NMode',12);
disp(Dyn1.output.Campbell);
```

Num	Status	rpm1	rpm2	rpm3	rpm4	rpm5	rpm6	rpm7
1	BW	22.405	21.883	21.362	20.842	20.325	19.812	19.305
2	FW	22.406	22.927	23.446	23.963	24.475	24.982	25.483
3	BW	59.647	59.647	59.647	59.647	59.647	59.647	59.647
4	BW	83.165	82.211	81.173	80.064	78.884	77.632	76.311
5	FW	83.198	84.087	84.931	85.719	86.455	87.142	87.785
6	BW	94.803	94.803	94.803	94.803	94.803	94.803	94.803
7	BW	191.83	182.03	172.8	164.26	156.4	149.23	142.72
8	FW	192.02	202.43	213.49	225.03	236.96	249.19	261.6
9	BW	275.76	263.58	252.03	241.24	231.17	221.79	213.07
10	FW	276.1	289.1	303.09	317.86	333.25	348.56	360.66
11	FW	343.94	343.94	343.94	343.94	343.94	343.93	343.94
12	FW	366.24	366.34	366.65	367.25	368.33	370.62	377.31



4.10 High speed rotor example (Flag=9)

下图为一个工业用的高速机械的转子，工作转速为50000 r/min，转子尺寸如图中所示（单位毫米）。转子由两个滑动轴承B1、B2支承。两端有两个铝制叶轮D1、D2。

已知有关参数为:

轴的材质为钢，其弹性模量为2.0E11N/m^2，质量密度为7800kg/m^3，泊桑比为0.3;

B1的油膜刚度系数为 K_{xx} =8E7N/m，K_{xy} =1E7N/m，K_{yx} =6E7N/m，K_{yy} =1E8N/m;

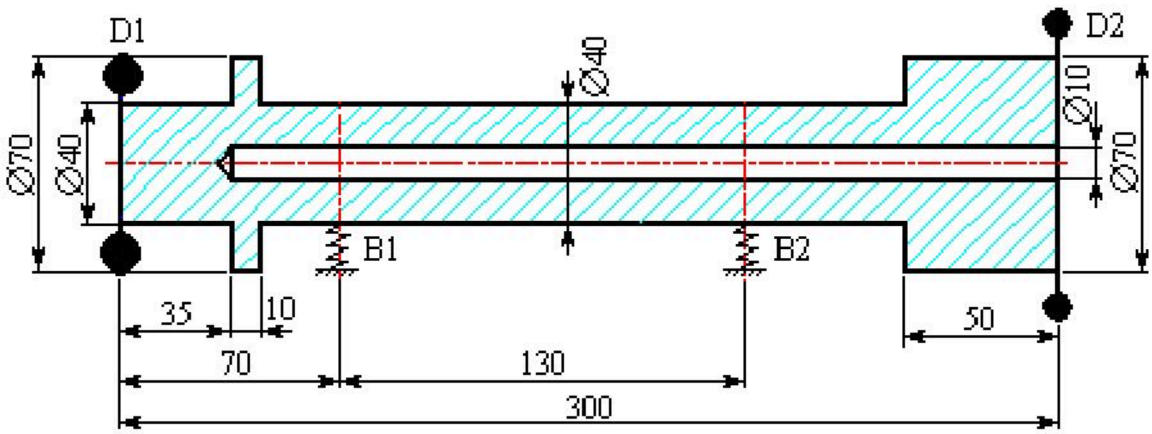
B1的油膜阻尼系数为 C_{xx} =8E3N s/m，C_{xy} =3E3N s/m，C_{yx} =3E3N s/m，C_{yy} =1.2E4N s/m;

B2的油膜刚度系数为 K_{xx} =5E7N/m，K_{xy} =2E7N/m，K_{yx} =4E7N/m，K_{yy} =7E7N/m;

B2的油膜阻尼系数为 C_{xx} =6E3N s/m，C_{xy} =1.5E3N s/m，C_{yx} =1.5E3N s/m，C_{yy} =8E3N s/m;

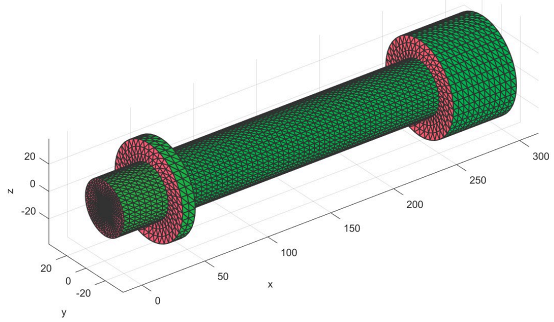
D1的质量为1.2kg，极转动惯量为2.4 E-3kg m^2，直径转动惯量为1.2E-3kg m^2；

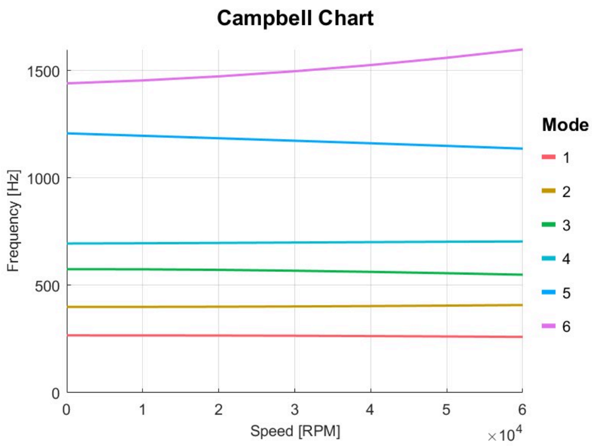
D2的质量为1.0kg，极转动惯量为2.0E-3kg m^2，直径转动惯量为1.0E-3kg m^2。

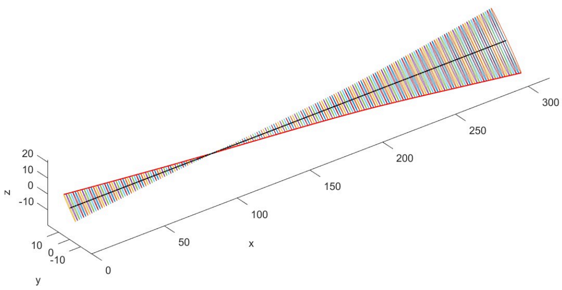



```
% Shaft
inputshaft1.Length = [35;45;70;200;250;300];
inputshaft1.ID = [[0,0];[10,10];[10,10];[10,10];[10,10];[10,10]];
inputshaft1.OD = [[40,40];[70,70];[40,40];[40,40];[40,40];[70,70]];
paramsshaft1.Beam_N = 16;
paramsshaft1.N_Slice=201;
obj1 = shaft.Commonshaft(paramsshaft1, inputshaft1);
obj1 = obj1.solve();
Plot3D(obj1)
mat{1,1}=obj1.params.Material;
```

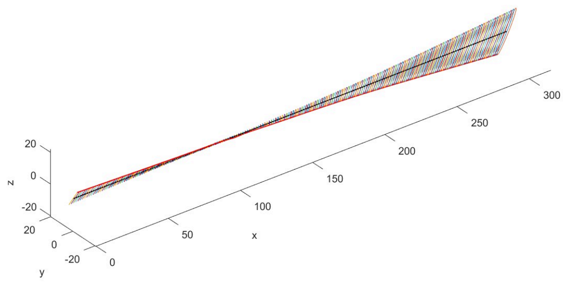
```
inputRotDyn.Shaft=obj1.output.BeamMesh;
inputRotDyn.Speed=[0,10000,20000,30000,40000,50000,60000];
inputRotDyn.PointMass=[1,1.2e-3,2.4,1.2;size(obj1.output.Node,1),1e-3,2,1];
inputRotDyn.MaterialNum=1;
paramsRotDyn.Material=mat;
paramsRotDyn.PrintCampbell=1;
paramsRotDyn.PrintORB=1;
Dyn1 = solve.RotDyn(paramsRotDyn,inputRotDyn);
[Dyn1,Num1]= AddCnode(Dyn1,70);
Dyn1=AddSupport(Dyn1,Num1,[1e10,8e4,1e5,1e4,6e4,0,8,12,3,3]);
[Dyn1,Num2]= AddCnode(Dyn1,200);
Dyn1=AddSupport(Dyn1,Num2,[0,5e4,7e4,2e4,4e4,0,6,8,1.5,1.5]);
Dyn1 = Dyn1.solve();
ANSYSSolve(Dyn1.output.Assembly);
Dyn1=PlotCampbell(Dyn1,'NMode',6);
disp(Dyn1.output.Campbell);
for i=1:6
PlotORB(Dyn1,2,i,'scale',1)
end
Dyn1=CalculateCriticalSpeed(Dyn1);
disp(Dyn1.output.CriticalSpeed);
```



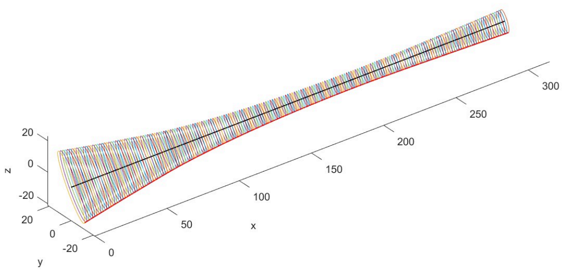




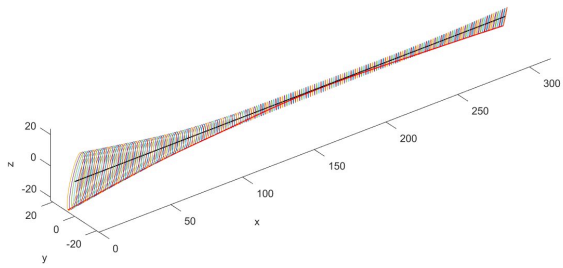
20
10
0
-10
z
10
0
-10
y
0
50
100
150
200
250
300
x



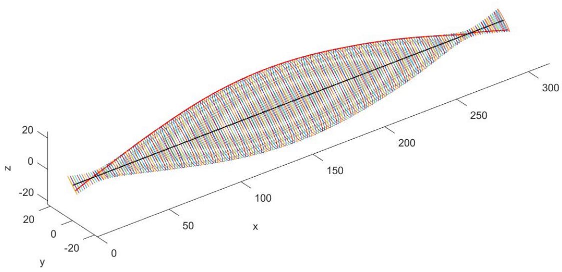
20
0
-20
z
20
0
-20
y
0
50
100
150
200
250
300
x



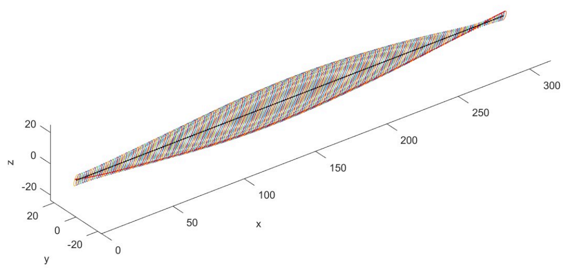
20
0
-20
z
20
0
-20
y
0
50
100
150
200
250
300
x



20
0
-20
z
20
0
-20
y
0
50
100
150
200
250
300
x



20
0
-20
z
20
0
-20
y
0
50
100
150
200
250
300
x

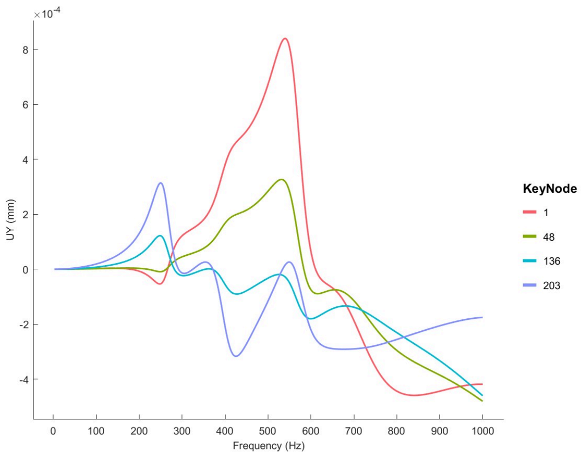


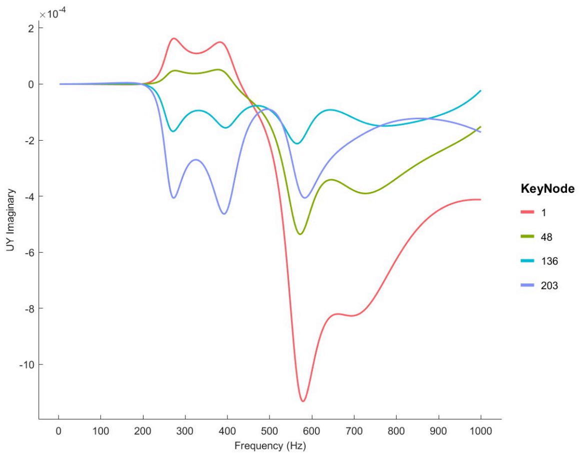
20
0
-20
z
20
0
-20
y
0
50
100
150
200
250
300
x

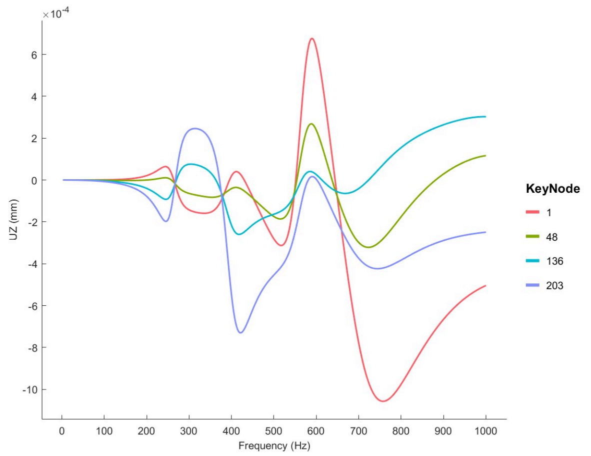
4.11 High speed rotor Harmonic analysis (Flag=10)

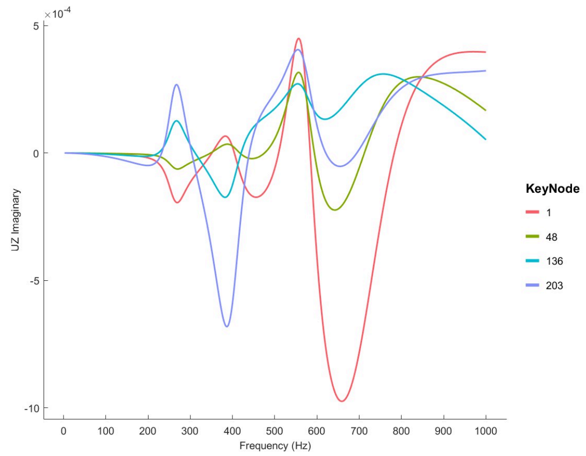
在高速轴上施加me=0.5e-6 Nmm的不平衡量，进行谐响应分析，可以观察此时轴节点的谐响应。

```
% Shaft
inputshaft1.Length = [35;45;70;200;250;300];
inputshaft1.ID = [[0,0];[10,10];[10,10];[10,10];[10,10];[10,10]];
inputshaft1.OD = [[40,40];[70,70];[40,40];[40,40];[40,40];[70,70]];
paramsshaft1.Beam_N = 16;
paramsshaft1.N_Slice=201;
obj1 = shaft.Commonshaft(paramsshaft1, inputshaft1);
obj1 = obj1.solve();
Plot3D(obj1)
mat{1,1}=obj1.params.Material;
inputRotDyn.Shaft=obj1.output.BeamMesh;
inputRotDyn.Speed=[0,10000,20000,30000,40000,50000,60000];
inputRotDyn.PointMass=[1,1.2e-3,2.4,1.2;size(obj1.output.Node,1),1e-3,2,1];
inputRotDyn.UnBalanceForce=[1,0.5e-6;size(obj1.output.Node,1),0.5e-6];
inputRotDyn.MaterialNum=1;
paramsRotDyn.Material=mat;
paramsRotDyn.Type=3;
Dyn1 = solve.RotDyn(paramsRotDyn,inputRotDyn);
[Dyn1,Num1]= AddCnode(Dyn1,70);
Dyn1=AddSupport(Dyn1,Num1,[1e10,8e4,1e5,1e4,6e4,0,8,12,3,3]);
[Dyn1,Num2]= AddCnode(Dyn1,200);
Dyn1=AddSupport(Dyn1,Num2,[0,5e4,7e4,2e4,4e4,0,6,8,1.5,1.5]);
Dyn1 = Dyn1.solve();
ANSYSSolve(Dyn1.output.Assembly);
Dyn1=PlotSpeedup(Dyn1);
```





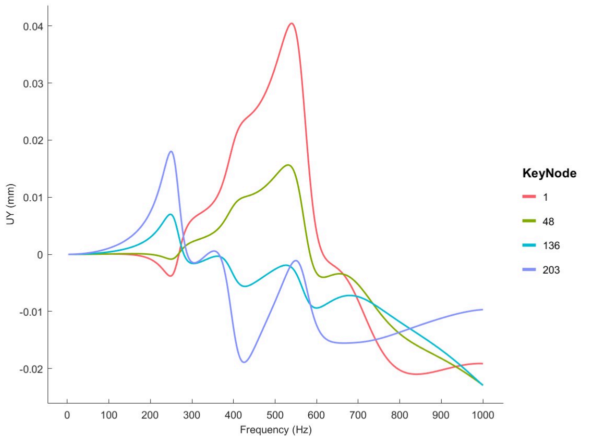


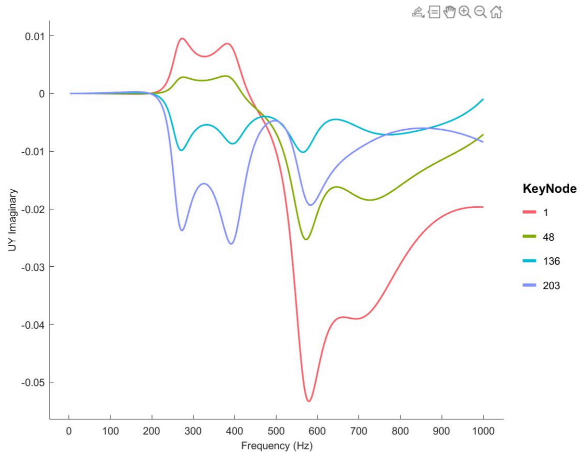


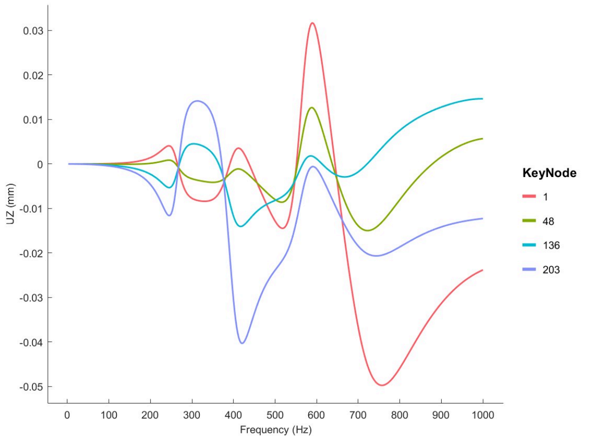
4.12 High speed unbalance (Flag=11)

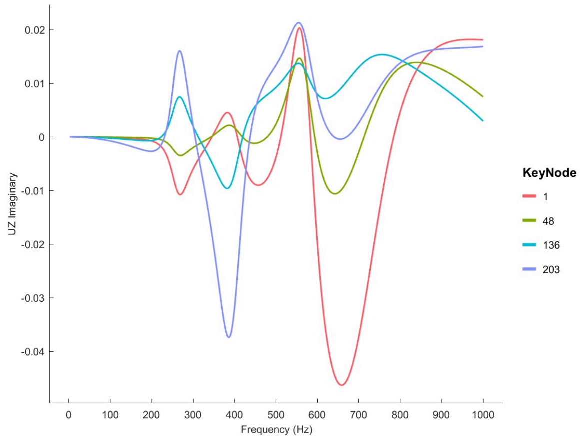
如果输入平衡质量，RotDyn会计算轴的重心，根据ISO1940标准计算平衡盘上的不平衡载荷，如下所示，设置平衡质量G=2.5,平衡转速为3000RPM,得到的谐响应如下。

```
% Shaft
inputshaft1.Length = [35;45;70;200;250;300];
inputshaft1.ID = [[0,0];[10,10];[10,10];[10,10];[10,10];[10,10]];
inputshaft1.OD = [[40,40];[70,70];[40,40];[40,40];[40,40];[70,70]];
paramsshaft1.Beam_N = 16;
paramsshaft1.N_Slice=201;
obj1 = shaft.Commonshaft(paramsshaft1, inputshaft1);
obj1 = obj1.solve();
Plot3D(obj1)
mat{1,1}=obj1.params.Material;
inputRotDyn.Shaft=obj1.output.BeamMesh;
inputRotDyn.Speed=[0,10000,20000,30000,40000,50000,60000];
inputRotDyn.PointMass=[1,1.2e-3,2.4,1.2;size(obj1.output.Node,1),1e-3,2,1];
inputRotDyn.MaterialNum=1;
inputRotDyn.BalanceQuality=[2.5,3000,0,300,0];
paramsRotDyn.Material=mat;
paramsRotDyn.Type=3;
Dyn1 = solve.RotDyn(paramsRotDyn,inputRotDyn);
[Dyn1,Num1]= AddCnode(Dyn1,70);
Dyn1=AddSupport(Dyn1,Num1,[1e10,8e4,1e5,1e4,6e4,0,8,12,3,3]);
[Dyn1,Num2]= AddCnode(Dyn1,200);
Dyn1=AddSupport(Dyn1,Num2,[0,5e4,7e4,2e4,4e4,0,6,8,1.5,1.5]);
Dyn1 = Dyn1.solve();
ANSYSSolve(Dyn1.output.Assembly);
PlotSpeedup(Dyn1);
```









5 参考文献

[1] ANSYS结构动力分析与应用

[2] 转动机械的转子动力学设计