Imagen que contiene negro, tabla, pájaro, búho

Descripción generada automáticamente



**UNIVERSIDAD AUTONOMA DE COAHUILA**

**FACULTAD DE SISTEMAS**

**ALUMNO: RICARDO GABRIEL RODRIGUEZ GONZALEZ**

**MATRICULA: 17001433**

**METODOS NUMERICOS**

**EXAMEN ORDINARIO**

**PROFESOR: Miguel Ángel Mendoza Zamora**

**8 DE JUNIO DE 2022**

VOLUMENES DE SÓLIDOS DE REVOLUCIÓN

Los sólidos de revolución son sólidos que se generan al girar una región alrededor de un eje.

Gráfico

Descripción generada automáticamenteGráfico, Gráfico de líneas

Descripción generada automáticamentePor ejemplo: La figura siguiente se genera como un sólido que resulta al girar el conjunto de puntos que se obtienen usando como intervalo de evaluación en la función , alrededor del eje .

Un sólido de revolución es, desde otra perspectiva, una figura tridimensional que se caracteriza porque su superficie no es plana, sino que es curva.

Cabe señalar que los sólidos de revolución pueden tomar distintas formas, incluso irregulares.

Otro punto a tener en cuenta es que la superficie plana que gira para formar el sólido puede, o no, cruzarse con el eje de revolución.

El código hecho en OCTAVE, que se utiliza para generar el volumen del sólido de revolución es muy sencillo, se basa en el cálculo del volumen de un cilindro.

clear

clc

clf

format long

close all

xx=linspace(**0**,**25**,**100**);

ff=sqrt(xx);

plot(xx,ff,'-k')

hold on

grid

n=input('Dame la cantidad de puntos : ');

x=linspace(**0**,**25**,n);

plot(x,zeros(**1**,n),'.r')

y=sqrt(x);

**for** i=**1**:n,

plot([x(i),x(i)],[**0**,y(i)],'-k') %se traza una recta vertical

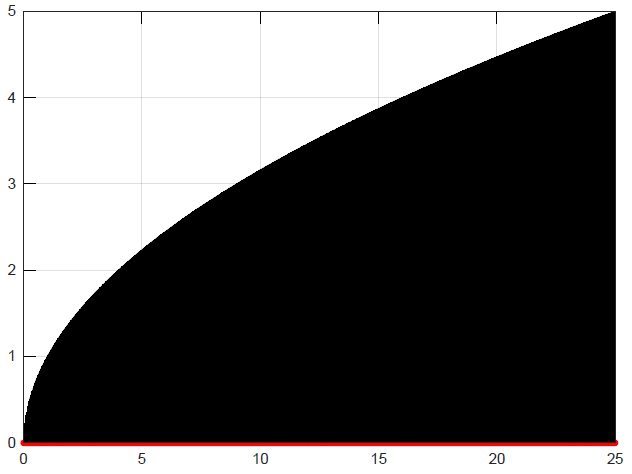
**end**

area\_circulo=pi\*y.^**2**;

base=abs(x(**1**)-x(**2**));

vol=sum(base\*area\_circulo)

Dame la cantidad de puntos :1000

vol = 982.7304346814919

display('El valor exacto del volumen es: ')

**625**\*pi/**2**

El valor exacto del volumen es:

ans = **981.7477042468104**

El siguiente código se utiliza para calcular el volumen de un sólido de revolución dando la función y el intervalo . El programa se llama **grafica\_vol\_020622.m**

clear

clf % limpia ventana de graficacion

clc % limpia ventana de comandos

close all % cierra ventana de graficacion

clear % limpia memoria

format long

g1=' Dame la Función ';

g2=' F(x)= ';

disp(g1); %despliega la cadena de

caracteres en ventana de comandos

funcion=input(g2,'s');

vf=vectize(funcion); % se usa la funcion vectize

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

k=**25**; % necesario para graficar circunferencias

intervalo=input('Dame el intervalo de graficacion [a

b] :');

a=intervalo(**1**);

b=intervalo(**2**);

x=linspace(a,b,k);

f=eval(vf); % funcion 1

figure

n=length(x);

plot(x,f,'\*r')

title(vf)

hold on

grid

xx=**0**;

ff=**0**;

% se suavizan los datos usando para disminuir la variabilidad

%se usa promedios moviles

ff=f;

xx=x;

yy=ff;

plot(xx,ff,'.') %se grafican los nuevos puntos

%%%%%%% spline cubico %%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

hipo=**0**;

mm=**1**;

ang=**0**;

**for** i=**1**:length(xx),

hipo(i)=sqrt(xx(i)^**2**+yy(i)^**2**);

ang(i)=acos(xx(i)/hipo(i));

**end**

figure % se crea otra ventana de graficacion

t=linspace(**0**,**2**\*pi,length(xx));

A=[xx' yy' ang', hipo', t'];

%%%%%%%%%%% graficando puntos %%%%%%%%%%%%%

plot3(A(:,**1**),A(:,**2**),zeros(length(xx),**1**),'\*r') %puntos rojos

hold on

plot3(A(:,**1**),-A(:,**2**),zeros(length(xx),**1**),'\*r')

%%%

plot3(A(:,**1**),zeros(length(xx),**1**),A(:,**2**),'\*k') %puntos negros

plot3(A(:,**1**),zeros(length(xx),**1**),-A(:,**2**),'\*k')

grid

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

n=length(xx);

**for** i=**1**:length(xx),

% se grafica la lineas verticales color negro

plot3([xx(i),xx(i)],[**0**,yy(i)],'-k')

plot3([xx(i),xx(i)],[-yy(i),yy(i)],'-k')

x=xx(i);y=yy(i);

h=sqrt(y^**2**);

teta=atan(y/x);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%

hold on

ang=pi/length(xx);

A=[cos(ang) sin(ang);

-sin(ang) cos(ang)];

xn=[h\*cos(teta);h\*sin(teta)];

j=**1**;

**while** j<**2**\*k,

x0=A\*xn; % se generan ptos para graficar circulos

plot3(xx(i),x0(**1**),x0(**2**),'.b');

xn=x0;

j=j+**1**;

**end**

**end**

title('Calcular volumen de pieza')

disp(' ')

disp('FIN DEL PROGRAMA: Calcular volumen de pieza')

disp(' ')

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

figure

plot(xx,ff,'-k')

hold on

grid

n=input('Dame la cantidad de puntos : ');

x=linspace(a,b,n);

plot(x,zeros(**1**,n),'\*r')

y=eval(vf);

**for** i=**1**:n,

plot([x(i),x(i)],[**0**,y(i)],'-k') %se traza una recta vertical

**end**

area\_circulo=pi\*y.^**2**;

base=abs(x(**1**)-x(**2**));

vol=sum(base\*area\_circulo)

disp('')

title({' Vol ',num2str(vol)})

PROYECTO ORDINARIO

1. Utilizando el programa **grafica\_vol\_020622.m**. Seleccione una de las siguientes opciones para calcular el volumen señalado por la gráfica.

Utilizamos la opción a)

clc

clear

clf

close all

format long

funcion1='x^2\*cos(x)\*atan(x)';

funcion=funcion1;

vf=vectize(funcion);

funcion2='x^2\*sin(x)';

funcion=funcion2;

vf2=vectize(funcion2)

%intervalo=input('Dame el intervalo de graficacion [a b] :');

%a=intervalo(1);

%b=intervalo(2);

a=pi/**32**;

b=pi;

x=a:(b-a)/**100**:b;

f=eval(vf);

f2=eval(vf2);

n=length(x);

figure

plot(x,f,'k')

hold on

plot(x,f2,'b')

grid

**for** i=**51**:length(x),

plot([x(i),x(i)],[f(i),f2(i)],'-r')

hold on

title({[' f1(x) = ', funcion1, ' --- f2(x) = ',funcion2 ]})

xlabel('[(a+b)/2 , b]')

**end**

disp('-------------------------------------------------------------------')

disp('Calcule el volumen generado por la dos graficas, ');

disp('Usando como referencia las rectas verticales rojas');

1. Grafique el sólido de revolución de la opción seleccionada.

clear

clf *% limpia ventana de graficacion*

clc *% limpia ventana de comandos*

close all *% cierra ventana de graficacion*

clear *% limpia memoria*

format long

*%%%%%%%%FUNCION 1%%%%%%%%%%%%%%%%%%%%%%*

funcion='x^2\*cos(x)\*atan(x)';

vf=vectize(funcion); *% se usa la funcion vectize*

*%%%%%%%%FUNCION 2%%%%%%%%%%%%%%%%%%%%%%*

funcion2='x^2\*sin(x)';

vf2=vectize(funcion2); *% se usa la funcion vectize*

*%%%%%%%%VARIABLES%%%%%%%%%%%%%%%%%%%%%%*

k = 25; *% necesario para graficar circunferencias*

a = pi/32;

b = pi;

x = a:(b-a)/100:b;

f = eval(vf);

f2 = eval(vf2);

n=length(x);

figure

plot(x,f,'k')

hold on

plot(x,f2,'b')

grid

**for** i=51:length(x),

plot([x(i),x(i)],[f(i),f2(i)],'-r')

hold on

title({[' f1(x) = ', funcion, ' --- f2(x) = ',funcion2 ]})

xlabel('[(a+b)/2 , b]')

**end**

xx = 0;

ff = 0;

ff2 = 0;

*% se suavizan los datos usando para disminuir la variabilidad*

*%se usa promedios moviles*

ff=f;

ff2 = f2;

xx = x;

yy = ff;

yy2 = ff2;

plot(xx,ff,'.k') *%se grafican los nuevos puntos*

plot(xx,ff2,'.b') *%se grafican los nuevos puntos*

*%[pi/32 pi]*

*%%%%%%% SPLINE CUBICO F1 %%%%%%%%%%%%*

*%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%*

hipo=0;

mm=1;

ang=0;

**for** i=1:length(xx),

hipo(i)=sqrt(xx(i)^2+yy(i)^2);

ang(i)=acos(xx(i)/hipo(i));

**end**

hipo2=0;

mm2=1;

ang2=0;

**for** i=1:length(xx),

hipo2(i)=sqrt(xx(i)^2+yy2(i)^2);

ang2(i)=acos(xx(i)/hipo2(i));

**end**

figure *% se crea otra ventana de graficacion*

t=linspace(0,2\*pi,length(xx));

A=[xx' yy' ang', hipo', t'];

A2=[xx' yy2' ang2', hipo2', t'];

*%%%%%%%%%%% graficando puntos %%%%%%%%%%%%%*

plot3(A(:,1),A(:,2),zeros(length(xx),1),'-r') *%puntos rojos*

hold on

plot3(A(:,1),-A(:,2),zeros(length(xx),1),'-r')

plot3(A2(:,1),A2(:,2),zeros(length(xx),1),'-r') *%puntos rojos*

hold on

plot3(A2(:,1),-A2(:,2),zeros(length(xx),1),'-r')

*%%%*

plot3(A(:,1),zeros(length(xx),1),A(:,2),'-k') *%puntos negros*

plot3(A(:,1),zeros(length(xx),1),-A(:,2),'-k')

plot3(A2(:,1),zeros(length(xx),1),A2(:,2),'-k') *%puntos negros*

plot3(A2(:,1),zeros(length(xx),1),-A2(:,2),'-k')

grid

*%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%*

n=length(xx);

**for** i=1:length(xx),

*% se grafica la lineas verticales color negro*

plot3([xx(i),xx(i)],[0,yy(i)],'-k')

plot3([xx(i),xx(i)],[-yy(i),yy(i)],'-k')

plot3([xx(i),xx(i)],[0,yy2(i)],'-k')

plot3([xx(i),xx(i)],[-yy2(i),yy2(i)],'-k')

x=xx(i);y=yy(i);

x2=xx(i);y2=yy2(i);

h=sqrt(y^2);

h2=sqrt(y2^2);

teta=atan(y/x);

teta2 =atan(y2/x2);

*%%%%%%%%%%%%%%%%%%%%%%%%%%%%*

hold on

ang=pi/length(xx);

A=[cos(ang) sin(ang);

-sin(ang) cos(ang)];

xn=[h\*cos(teta);h\*sin(teta)];

ang2= pi/length(xx);

A2=[cos(ang2) sin(ang2)

-sin(ang2) cos(ang2)];

xn2=[h2\*cos(teta2);h2\*sin(teta2)];

j=1;

**while** j<2\*k,

x0=A\*xn; *% se generan ptos para graficar circulos*

plot3(xx(i),x0(1),x0(2),'.b');

xn=x0;

x02=A2\*xn2;

plot3(xx(i),x02(1),x02(2),'.b');

xn2=x02;

j=j+1;

**end**

**end**

*%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%*

figure

plot(xx,ff,'-k')

hold on

plot(xx,ff2,'-k')

grid

n=input('Dame la cantidad de puntos : ');

x=linspace(a,b,n);

plot(x,zeros(1,n),'\*r')

y = eval(vf);

y2 = eval(vf2);

**for** i=1:n,

plot([x(i),x(i)],[y(i),y2(i)],'-k') *%se traza una recta vertical*

**end**

area\_circulo=pi\*(y+y2).^2;

base=abs(x(1)-x(2));

vol=sum(base\*area\_circulo)

disp('')

title({' Vol ',num2str(vol)})

Gráfico

Descripción generada automáticamenteInterfaz de usuario gráfica, Gráfico

Descripción generada automáticamente

Gráfico

Descripción generada automáticamente

Gráfico, Gráfico de superficie

Descripción generada automáticamente

Gráfico, Gráfico de superficie

Descripción generada automáticamente

Conclusión

En este proyecto estuvimos investigando como podíamos juntar 2 funciones para que fueran graficadas en el spline cubico el cual se nos fue otorgado para poder calcular y visualizar graficas en solidos de revolución, así mismo, también fue posible hacer el cálculo del volumen de esta figura, en este caso elegimos la opción “a” la cual elegimos no por ningún motivo en específico.