Capítulo 4.
Implementación

En este capítulo describimos los algoritmos utilizados para el trazado de las líneas tangentes exteriores a dos círculos para la construcción visual de los cilindros generalizados. Se describen también los pasos para la obtención de primitivas geométricas a partir de dichos cilindros. Además se muestran las acciones requeridas para rotar las figuras geométricas propuestas.

4.1 LÍNEAS TANGENTES A 2 CIRCUNFERENCIAS

Estas últimas fórmulas arrojan los puntos extremos de la circunferencia menor. Así se trazan las líneas tangentes una de (xT1, yT1) a (x T2 , y T2) y otra de (xT1´,y T1´) a (x T2´ , y T2´).

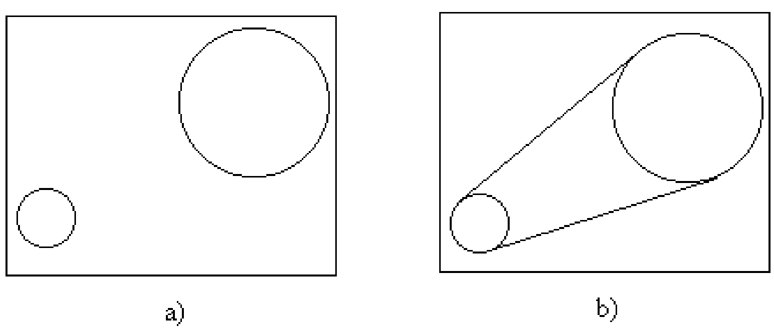

Figura 4.1 a)Círculos que conforman en Cilindro Generalizado en nuestra técnica
b)Líneas tangentes que muestran el espacio contemplado en esta técnica.

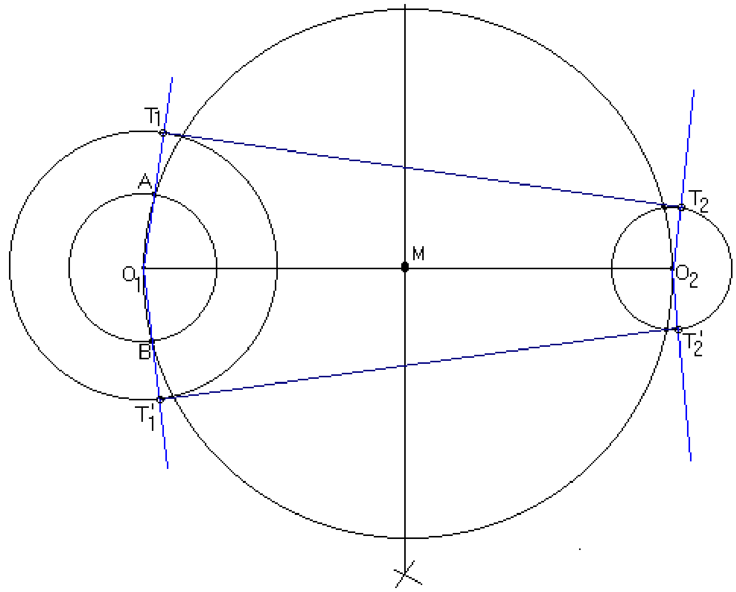

Figura 4.2 Rectas tangentes exteriores a 2 circunferencias. Adaptado de [Coll89].

$$x = \frac{x_2 + x_1}{2} + \frac{(x_2 - x_1)(r_1^2 - r_2^2)}{2d^2}$$
$$\pm \frac{y_2 - y_1}{2d^2}\sqrt{((r_1 + r_2)^2 - d^2)(d^2 - (r_2 - r_1)^2)}$$

y

$$y = \frac{y_2 + y_1}{2} + \frac{(y_2 - y_1)(r_1^2 - r_2^2)}{2d^2}$$
$$\pm \frac{x_2 - x_1}{2d^2}\sqrt{((r_1 + r_2)^2 - d^2)(d^2 - (r_2 - r_1)^2)}$$

:

$$x_{T1} = x_1 * (1 - \alpha_d) + x_A * (\alpha_d)$$
$$y_{T1} = y_1 * (1 - \alpha_d) + y_A * (\alpha_d)$$

y

$$x_{T1'} = x_1 * (1 - \alpha_d) + x_B * (\alpha_d)$$
$$y_{T1'} = y_1 * (1 - \alpha_d) + y_B * (\alpha_d)$$

$$x_{T2} = x_2 *(1-\alpha_d) + (x_A + (x_2 - x_1)) *(\alpha_d)$$
$$y_{T2} = y_2 *(1-\alpha_d) + (y_A + (y_2 - y_1)) *(\alpha_d)$$

y

$$x_{T2'} = x_2 *(1-\alpha_d) + (x_B + (x_2 - x_1)) *(\alpha_d)$$
$$y_{T2'} = y_2 *(1-\alpha_d) + (y_B + (y_2 - y_1)) *(\alpha_d)$$

4.2 Construcción de primitivas geométricas con cilindros generalizados

Para realizar estas figuras se uso una clase llamada cilindros que incluye los métodos para obtener cada uno de sus datos. La clase cilindros contiene lo siguiente:

```
public class V_cilindros
{
int x1,y1,r1,x2,y2,r2;
public V_cilindros(int x1,int y1,int r1,int x2,int
{
this.x1=x1;
this.y1=y1;
this.r1=r1;
this.x2=x2;
this.y2=y2;
this.r2=r2;
}
public int getX1() { return x1;}
public int getY1() { return y1;}
public int getR1() { return r1;}
public int getX2() { return x2;}
public int getY2() { return y2;}
public int getR2() { return r2;}
}
```

La obtención de los cilindros para cada primitiva geométricas se puede observar a continuación.

```
public static void cilindros_cuadrado(int ccx,int
{
int cx0,cy0,cr0,cx1,cy1,cx2,cy2,cx3,cy3,cx4,cy4;
cx0=ccx+(cclado/2);//circulo central
cy0=ccy+(cclado/2);
cr0=(int)cclado/2;
```

```
cx1=ccx;//circulo de la orilla inferior izquierda
cy1=ccy;
cx2=ccx;//superior izquierda
cy2=ccy+cclado;
cx3=ccx+cclado;
cy3=ccy+cclado;
cx4=ccx+cclado;
cy4=ccy;
V_cilindros cil1=new V_cilindros(cx0,cy0,cr0,cx1,c
cil_cuad.addElement(cil1);
V_cilindros cil2=new V_cilindros(cx0,cy0,cr0,cx2,c
cil_cuad.addElement(cil2);
V_cilindros cil3=new V_cilindros(cx0,cy0,cr0,cx3,c
cil_cuad.addElement(cil3);
V_cilindros cil4=new V_cilindros(cx0,cy0,cr0,cx4,c
cil_cuad.addElement(cil4);
}

public static void cilindros_triangulo(int ttx,int
{
int tx0,ty0,tr0,tx1,ty1,tx2,ty2,tx3,ty3,co;
float altura;
tx1=ttx; ty1=tty;
altura=(float).87*(ttlado);
tx2=ttx+(ttlado/2);
ty2=tty+(int)Math.floor(altura);
tx3=ttx+ttlado;
ty3=tty; //circulo central
co=(int)Math.floor(.5773*(ttlado/2));
tx0=ttx+(ttlado/2);
ty0=tty+(int)Math.floor(co);
tr0=co;
V_cilindros cil5=new V_cilindros(tx0,ty0,tr0,tx1,t
cil_tria.addElement(cil5);
V_cilindros cil6=new V_cilindros(tx0,ty0,tr0,tx2,t
cil_tria.addElement(cil6);
V_cilindros cil7=new V_cilindros(tx0,ty0,tr0,tx3,t
cil_tria.addElement(cil7);
}
public static void cilindros_rectangulo(int rrx,ir
{
int rx0,ry0,rr0,rx1,ry1,rr1,rx2,ry2,rx3,ry3,rx4,ry
rx0=rrx+(rralto/2);
ry0=rry+(rralto/2);
rr0=rralto/2;
rx1=rrx+rrancho-(rralto/2);
ry1=rry+(rralto/2);
rr1=rralto/2-2;
//Circulos de las esquinas
rx2=rrx;
ry2=rry;
rx3=rrx;
```

```
ry3=rry+rralto;
rx4=rrx+rrancho;
ry4=rry+rralto;
rx5=rrx+rrancho;
ry5=rry; //cilindro central
V_cilindros cil8=new V_cilindros(rx0,ry0,rr0,rx1,r
cil_rect.addElement(cil8);
V_cilindros cil9=new V_cilindros(rx0,ry0,rr0,rx2,r
cil_rect.addElement(cil9);
V_cilindros cil10=new V_cilindros(rx0,ry0,rr0,rx3,
cil_rect.addElement(cil10);
V_cilindros cil11=new V_cilindros(rx1,ry1,rr1,rx4,
cil_rect.addElement(cil11);
V_cilindros cil12=new V_cilindros(rx1,ry1,rr1,rx5,
cil_rect.addElement(cil12);
}
```

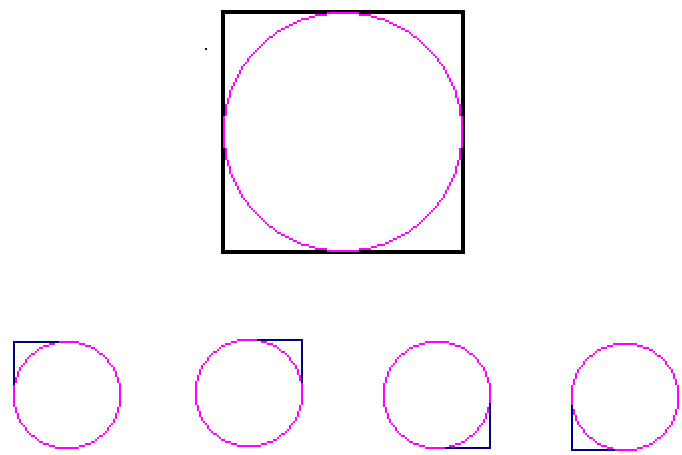

Figura 4.3 Formación de un cuadrado con cilindros generalizados

Tabla 4.1 Cilindros que representan un cuadrado

NUMERO DE CILINDRO (sentido manecillas reloj)	Circulo inicial (pequeño)	Circulo final (grande)
Cilindro 1	origen=(x,y), r=0	origen=(x+(lado/2),y+(lado/2) r=lado/2

Cilindro 2	origen=(x,y+lado), r=0	origen=(x+(lado/2),y+(lado/2) r=lado/2
Cilindro 3	origen=(x+lado,y+lado), r=0	origen=(x+(lado/2),y+(lado/2) r=lado/2
Cilindro 4	origen=(x+lado,y), r=0	origen= (x+(lado/2),y+(lado/2) r=lado/2

$$distancia\_a\_incentro = \frac{1}{\sqrt{3}} * \frac{lado}{2}$$

$$altura = \frac{\sqrt{3}}{2} * lado$$

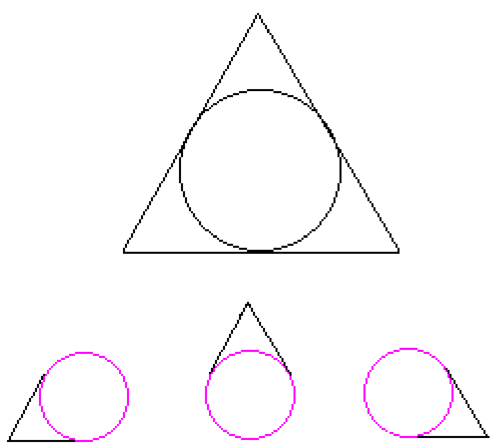

Figura 4.4 Formación de un triángulo con cilindros generalizados

Tabla 4.2 Cilindros que representan un triángulo

NUMERO DE CILINDRO (sentido manecillas reloj)	Circulo inicial (pequeño)	Circulo final (grande)
Cilindro 1	origen=(x,y), r=0	origen=(x+(lado/2), ycentral r=distancia_a_incentro

Cilindro 2	origen=(x+(lado/2), yaltura), r=0	origen=(x+(lado/2), ycentral r=distancia_a_incentro
Cilindro 3	origen=(x+lado,y), r=0	origen=(x+(lado/2), ycentral r=distancia_a_incentro

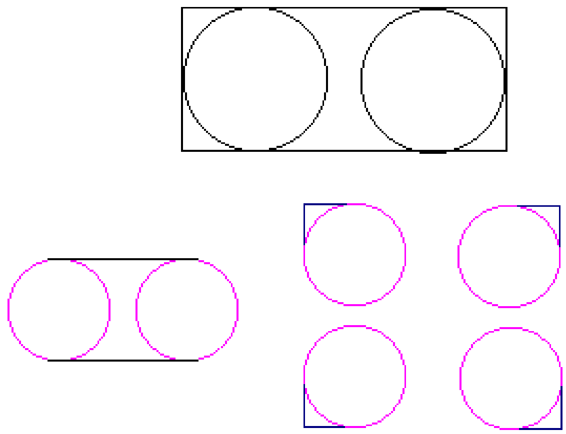

Figura 4.5 Formación de un rectángulo con cilindros generalizados

Tabla 4.3 Cilindros que representan un rectángulo

NUMERO DE CILINDRO(sentido manecillas reloj)	Circulo inicial	Círculo final
Cilindro 1(central)	origen=(x+(alto/2),y+(alto/2)), r=alto/2	origen=(x+ancho-(alto/2),y+(lado/2)), r=alto/2
Cilindro 2	origen=(x,y), r=0	origen=(x+(alto/2),y+(alto/2)), r=alto/2
Cilindro 3	origen=(x,y+alto), r=0	origen=(x+(alto/2),y+(alto/2)),

		r=alto/2
Cilindro 4	origen= (x+ancho,y+alto), r=0	origen=(x+ancho-(alto/2),y+(lado/2)), r=alto/2
Cilindro 5	origen=(x+lado,y), r=0	origen=(x+ancho-(alto/2),y+(lado/2)), r=alto/2

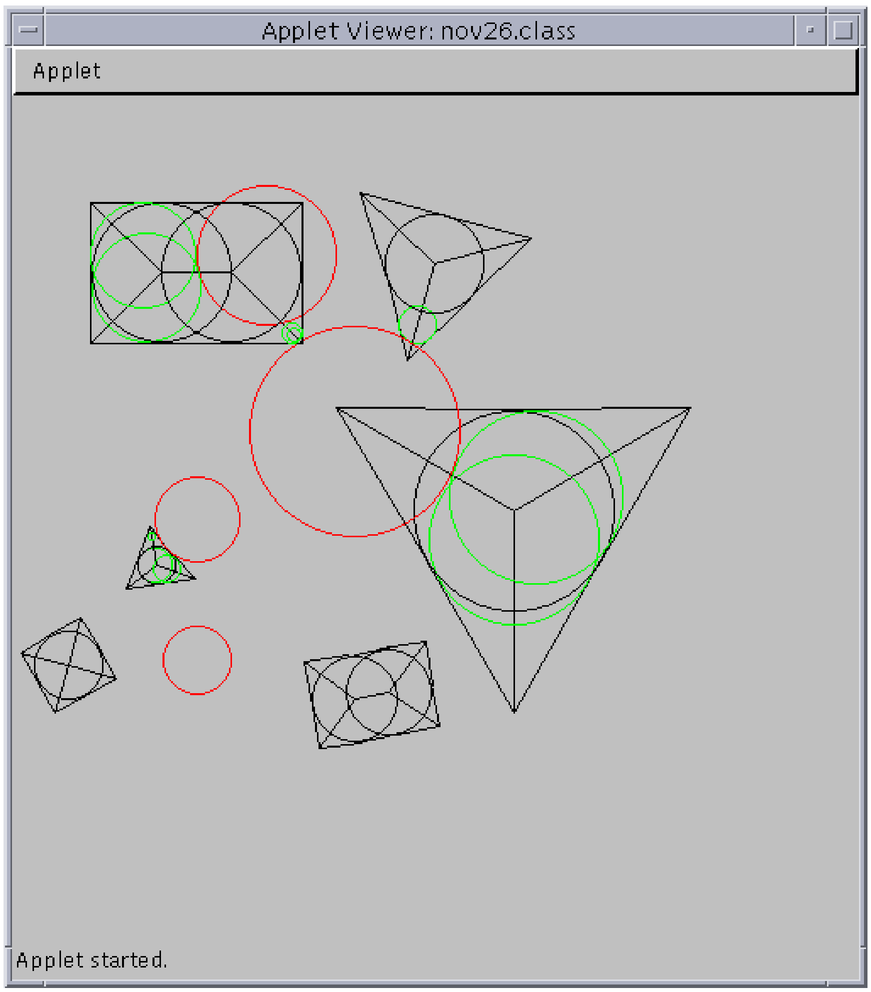

Figura 4.6 Primitivas geométricas en pruebas de colisión con obstáculos(foto del sistema)

4.3 Procedimiento para rotación de figuras

De forma que obtenemos la siguiente expresión:

x´ = xCos(teta) - ySen(teta)

y´ = xSen(teta) - yCos(teta)

donde (teta) es el ángulo de rotación.

3. Sumar xorigen, yorigen a los valores x´, y´ obtenidos del paso anterior. Estas nuevas coordenadas serán las esquinas rotadas sobre las cuales se vuelen a calcular los cilindros que conforman la figura. Para el triángulo se rotan dos esquinas a partir de una esquina origen y para el cuadrado y el rectángulo se rotan tres esquinas a partir de una esquina origen.

Cos(teta)	-Sen(teta)
Sen(teta)	Cos (teta)