

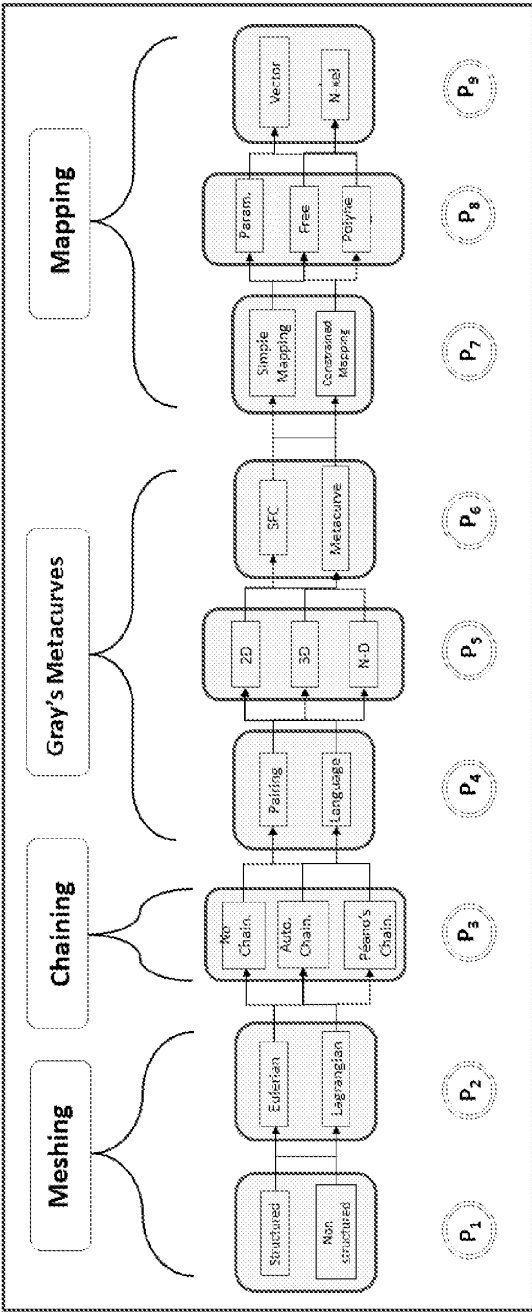


FIGURE 1

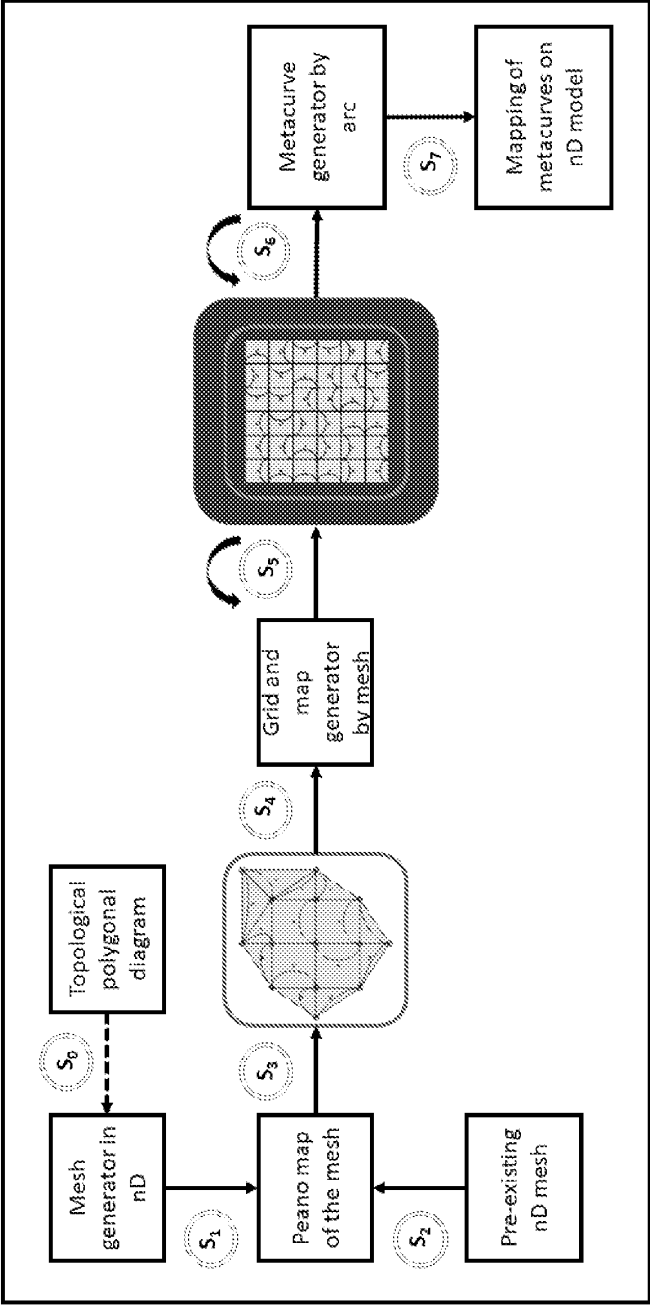


FIGURE 2

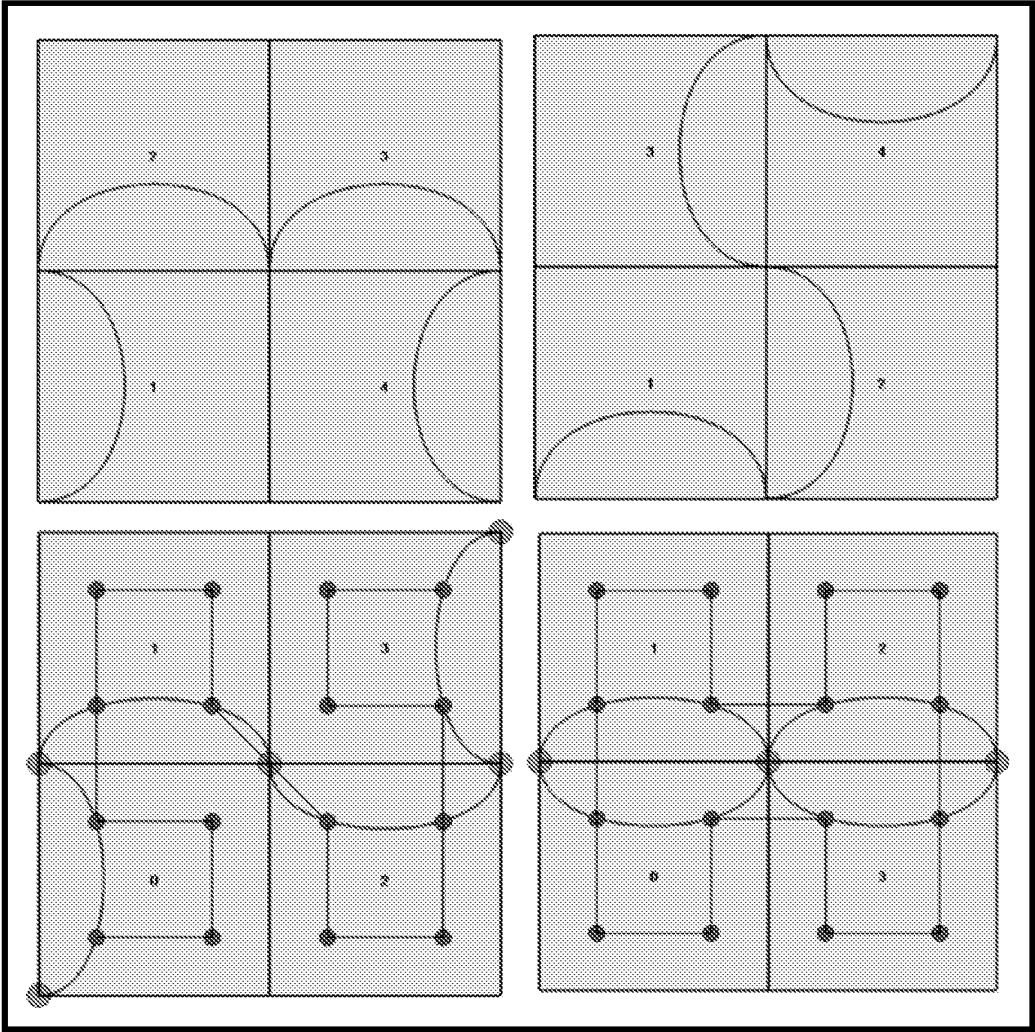


FIGURE 3

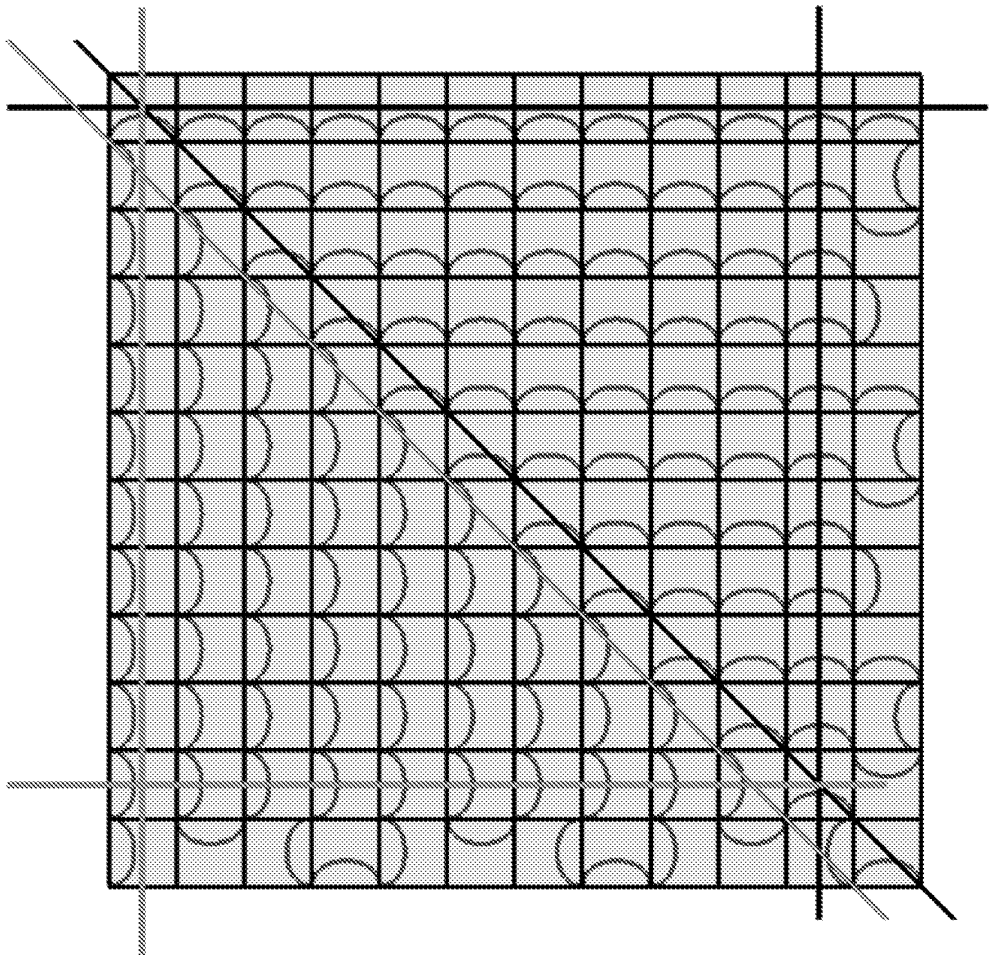


FIGURE 4

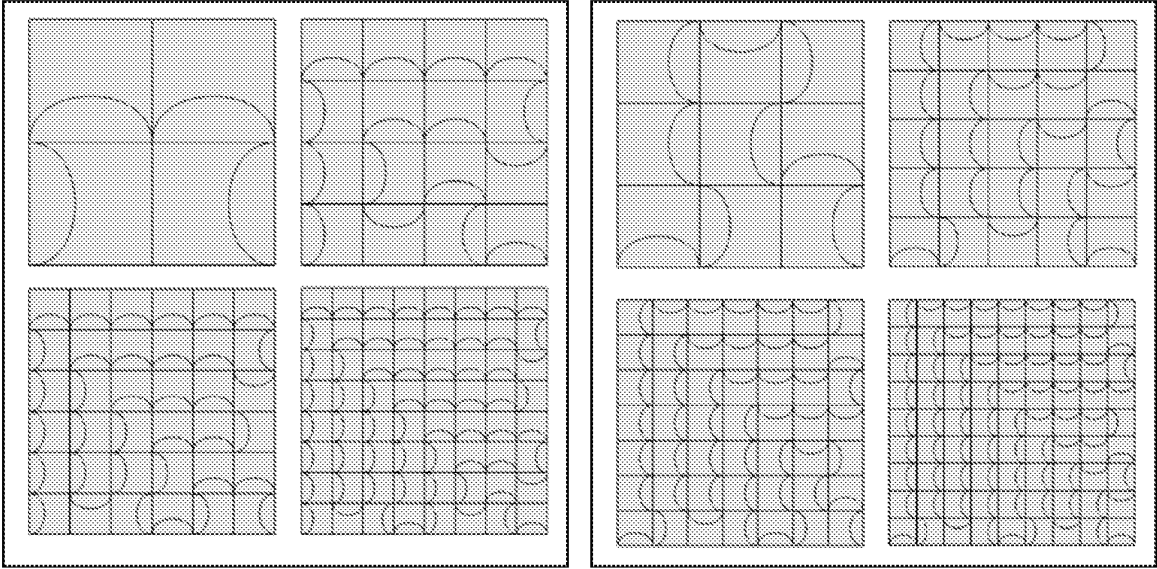


FIGURE 5

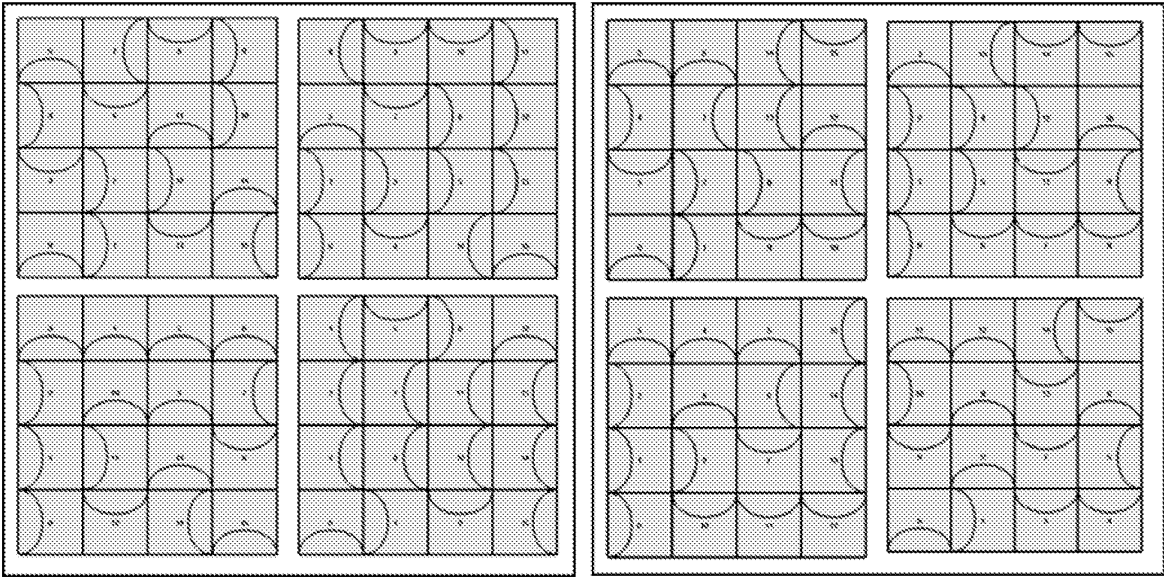


FIGURE 6

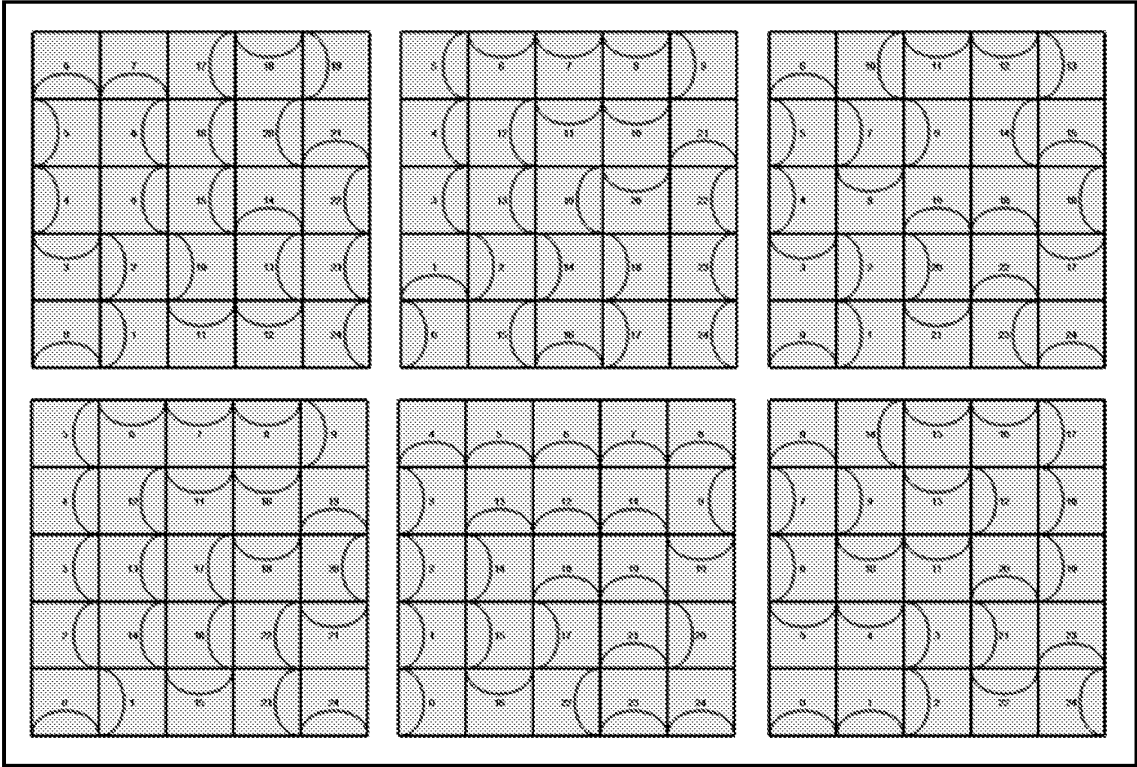


FIGURE 7

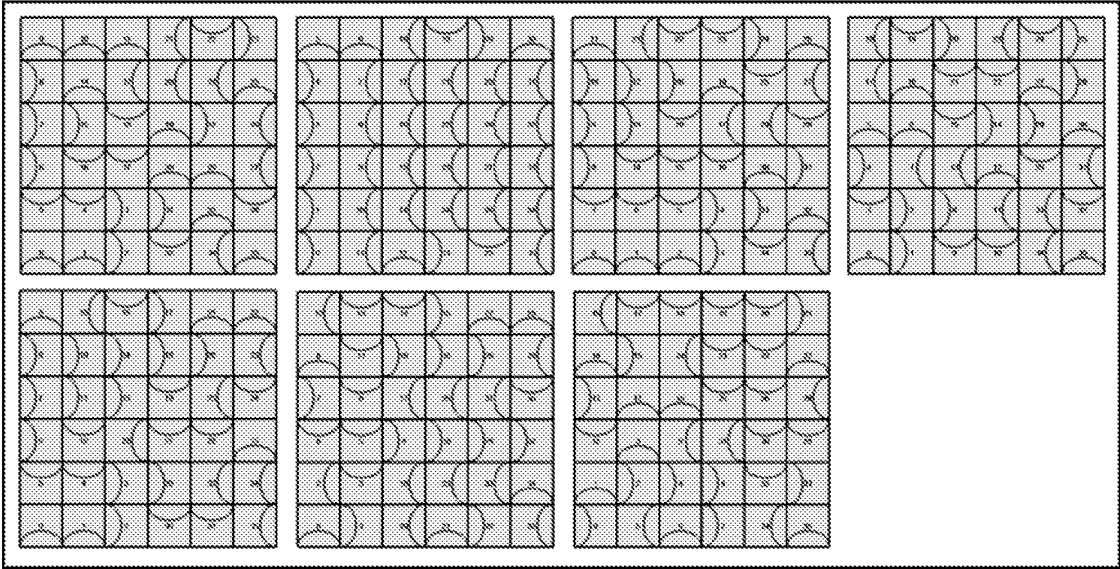


FIGURE 8

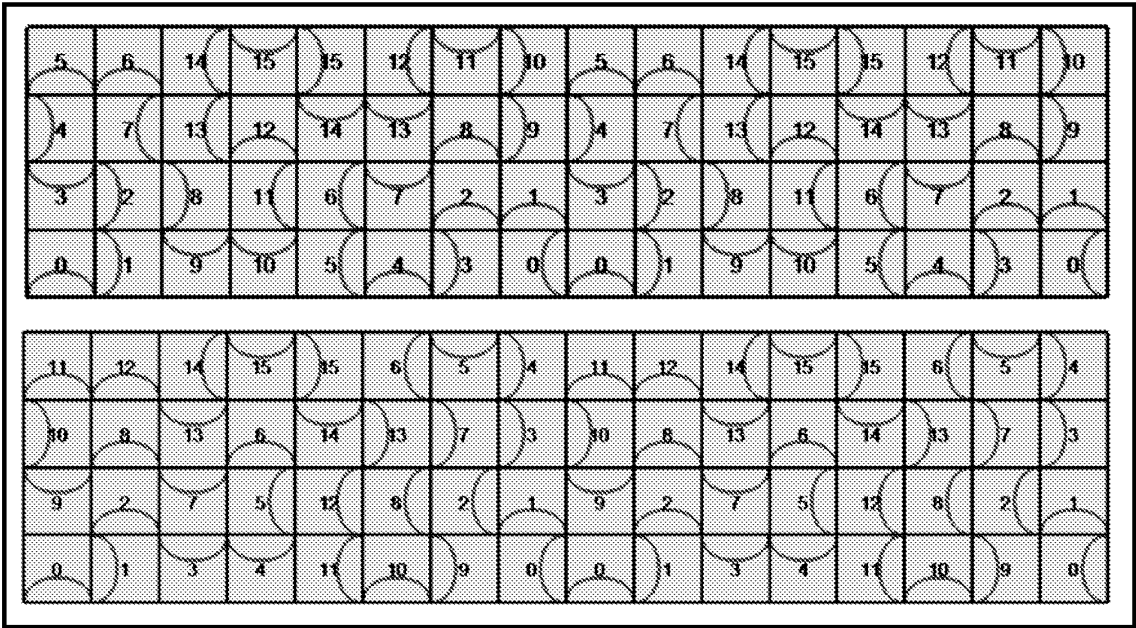


FIGURE 9

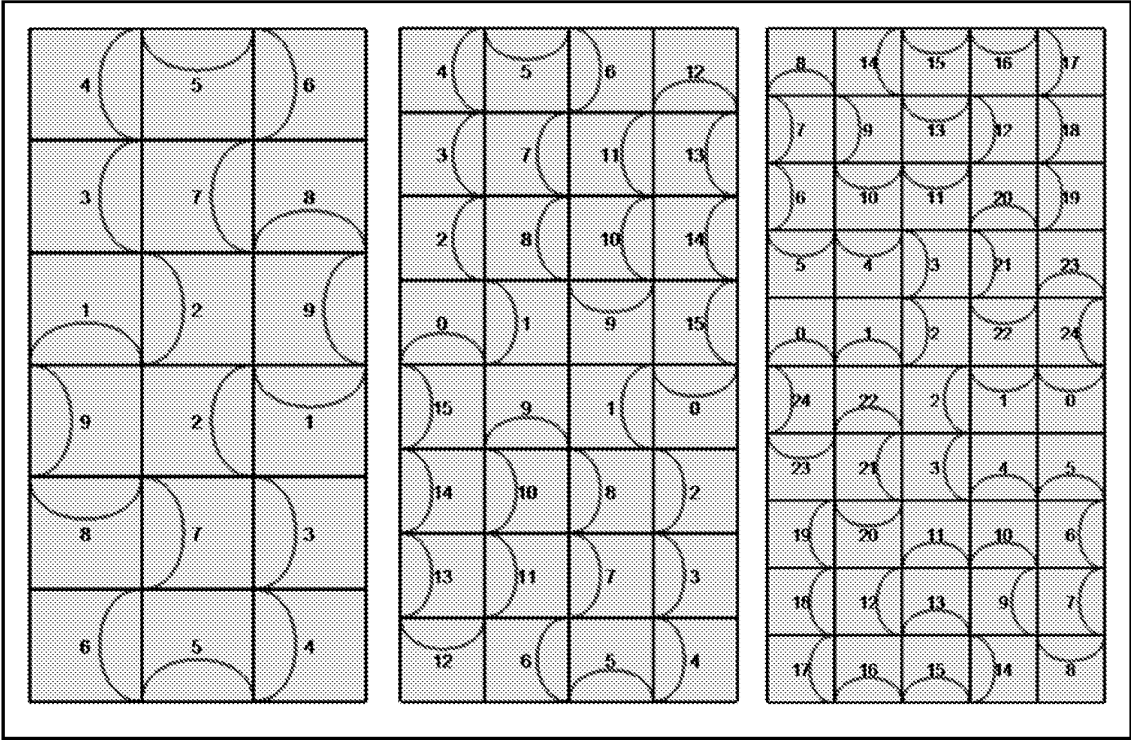


FIGURE 10

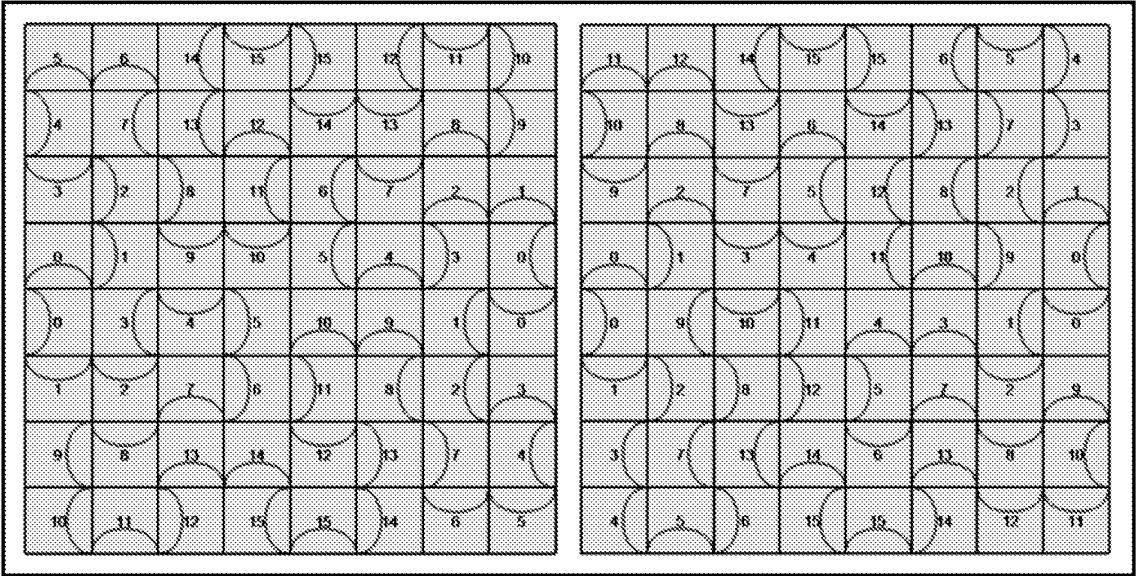


FIGURE 11

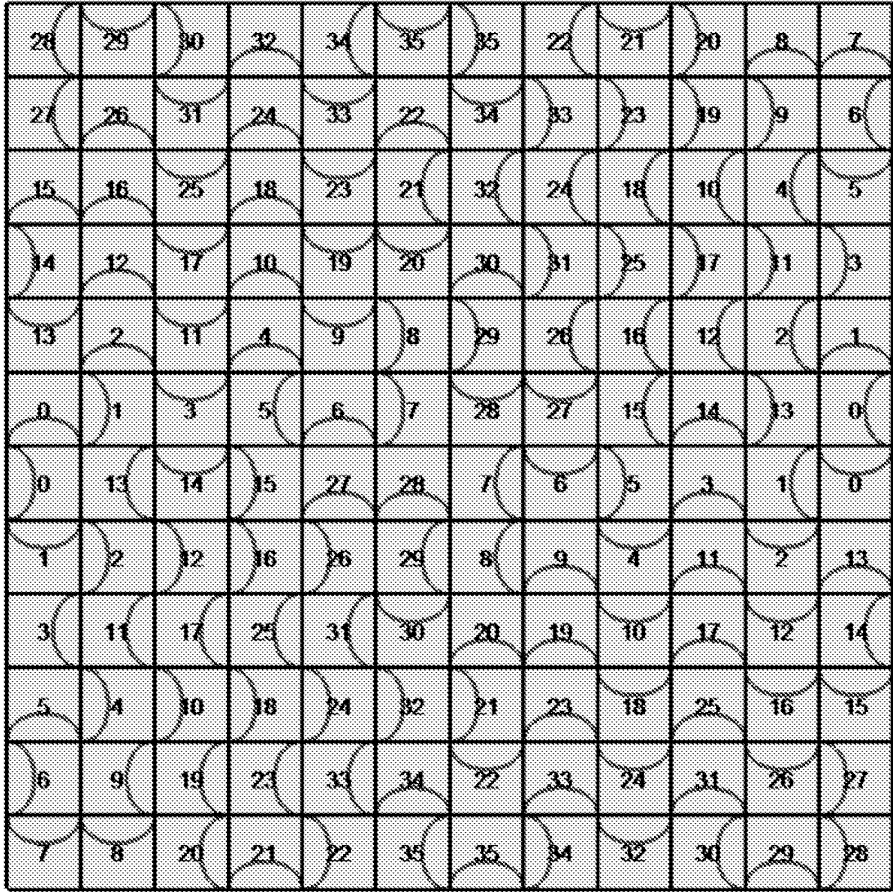


FIGURE 12

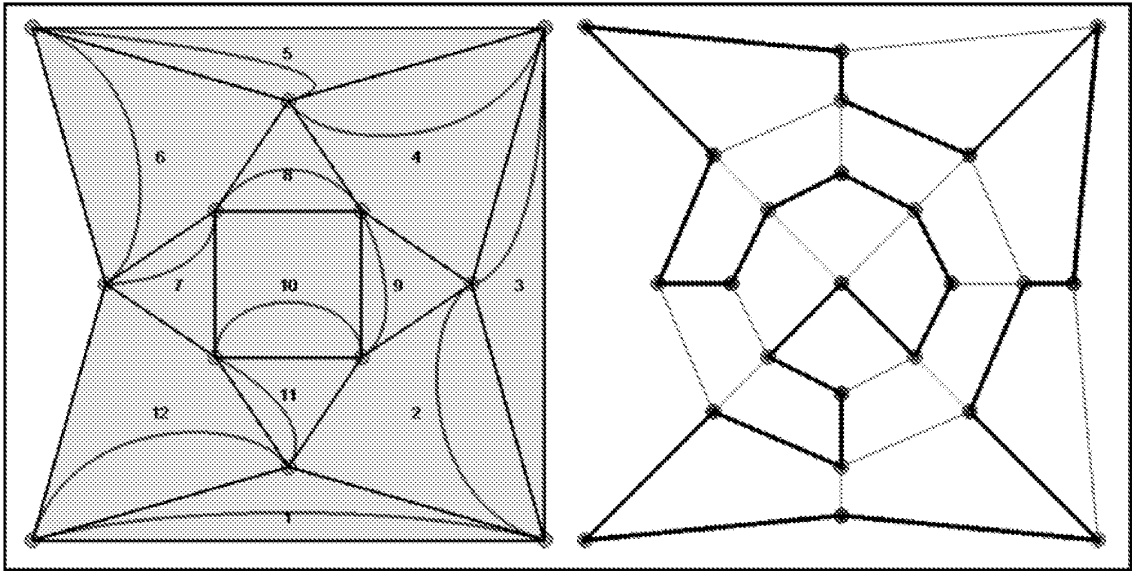


FIGURE 13

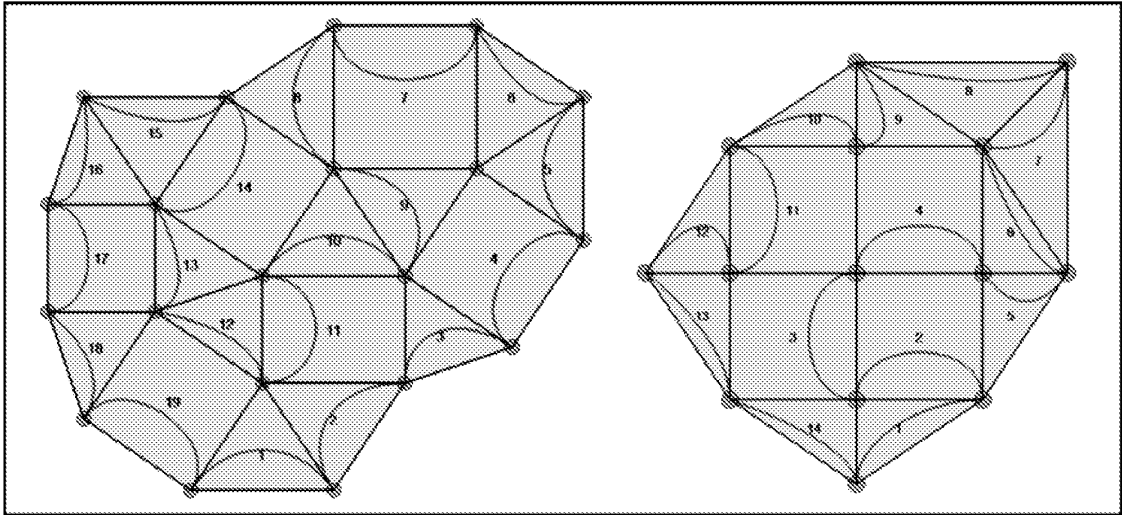


FIGURE 14

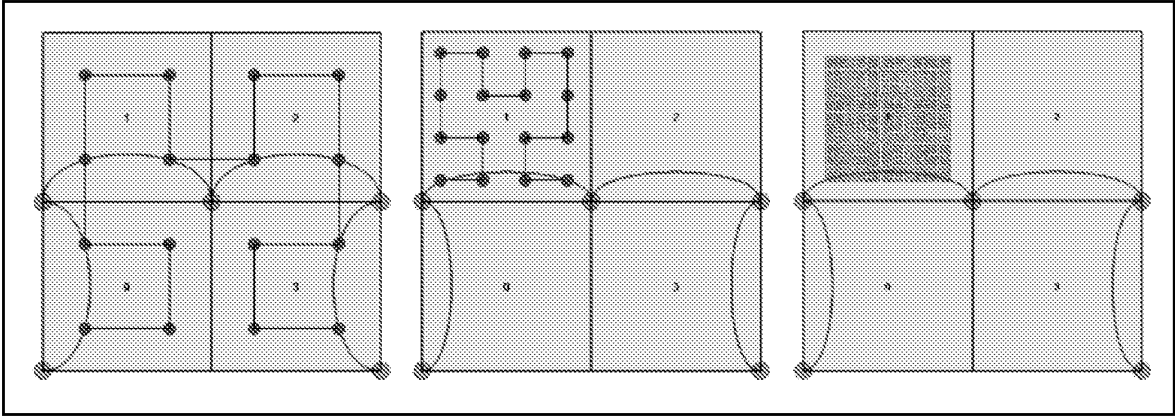


FIGURE 15

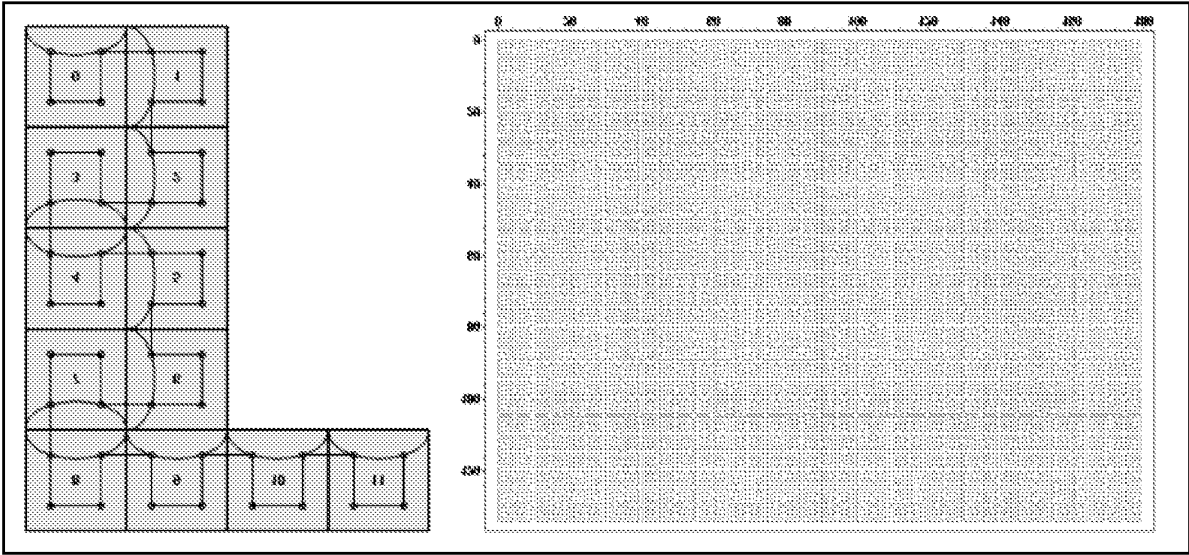


FIGURE 16

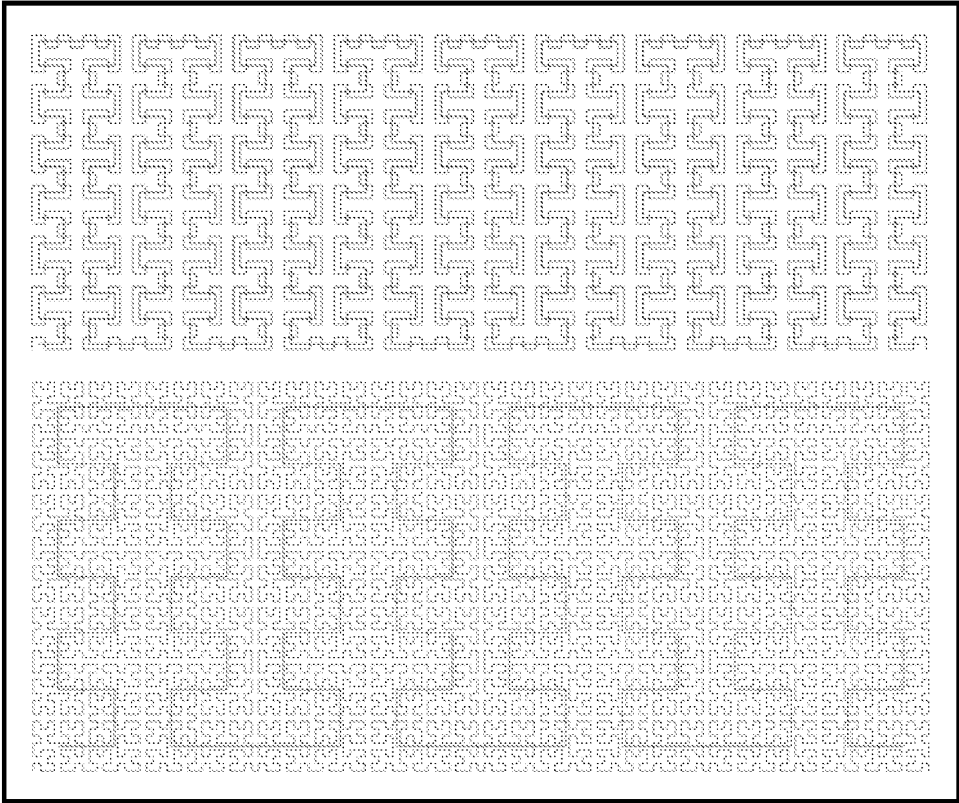


FIGURE 17

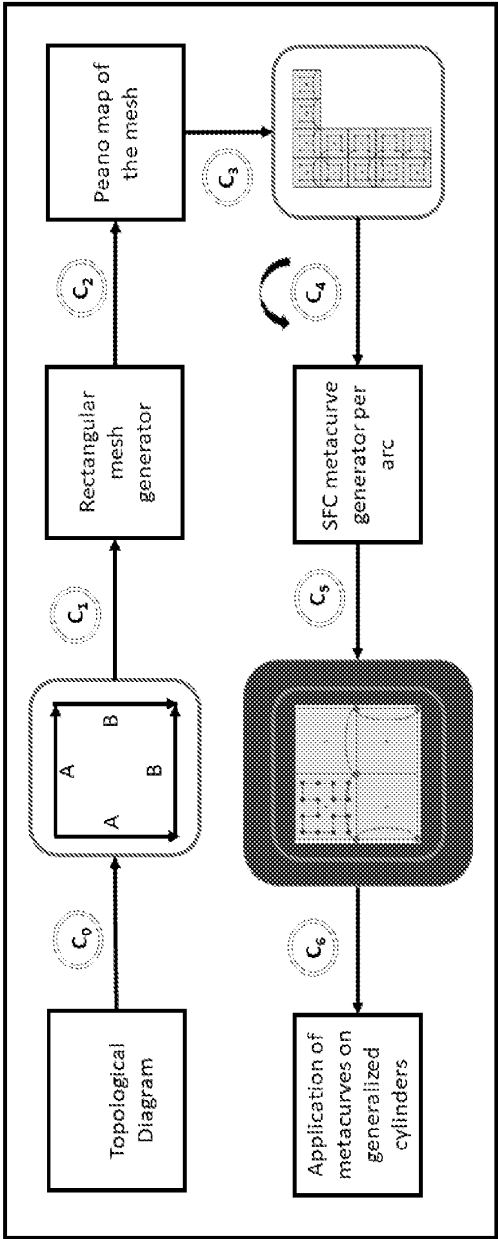


FIGURE 18

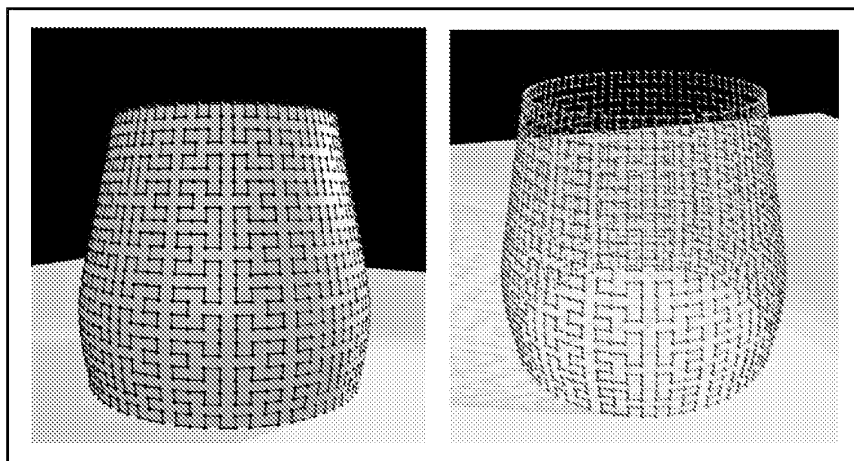


FIGURE 19

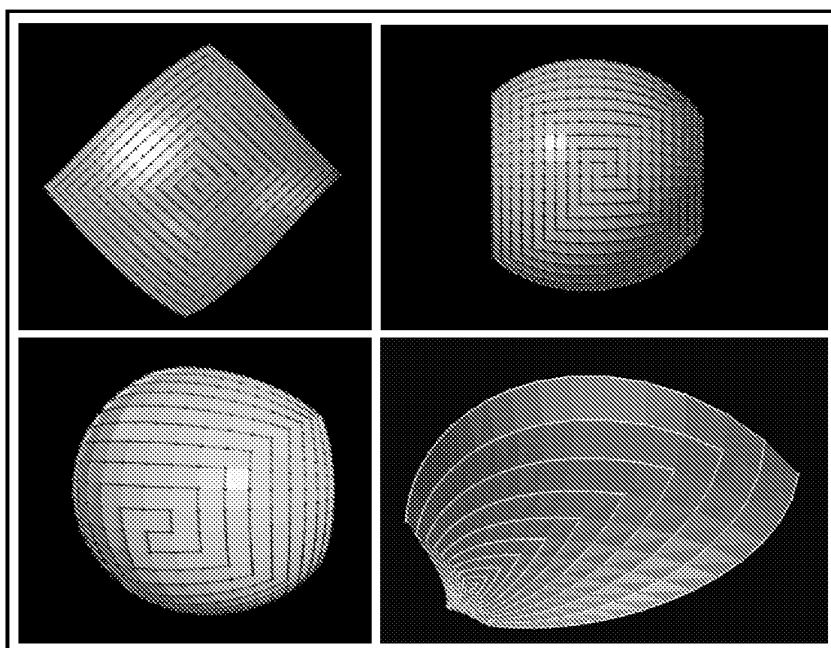


FIGURE 20

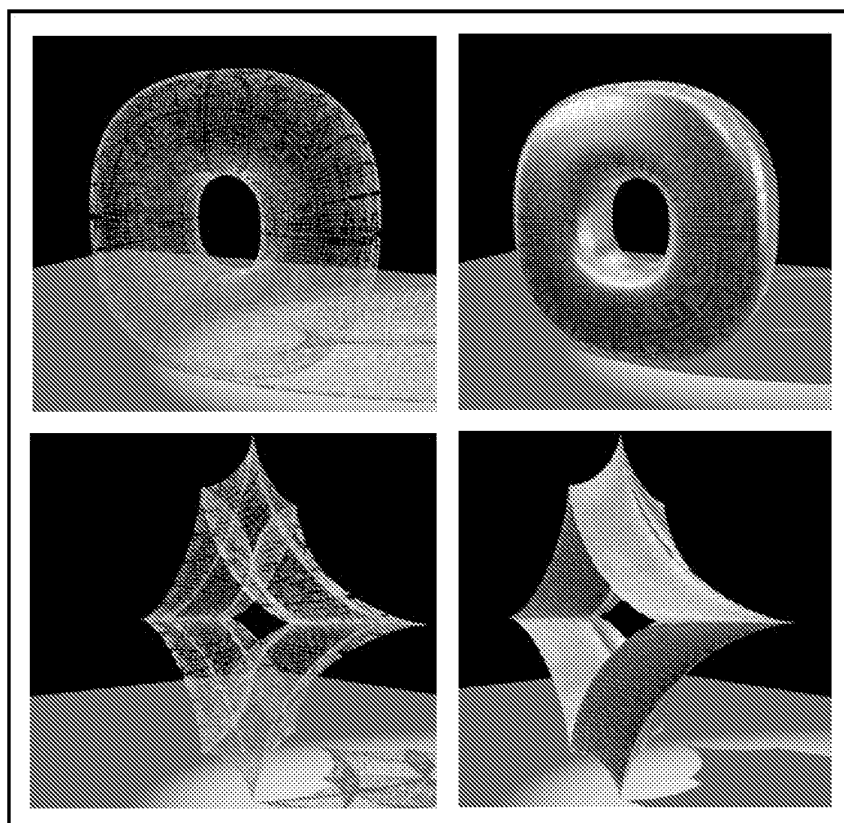


FIGURE 21

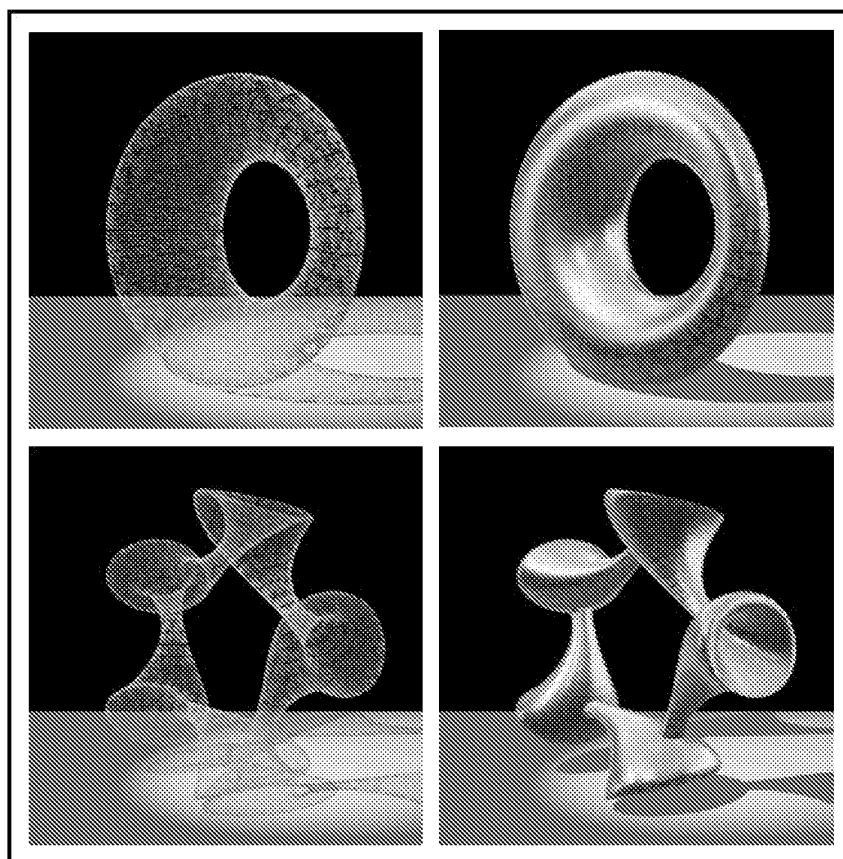


FIGURE 22

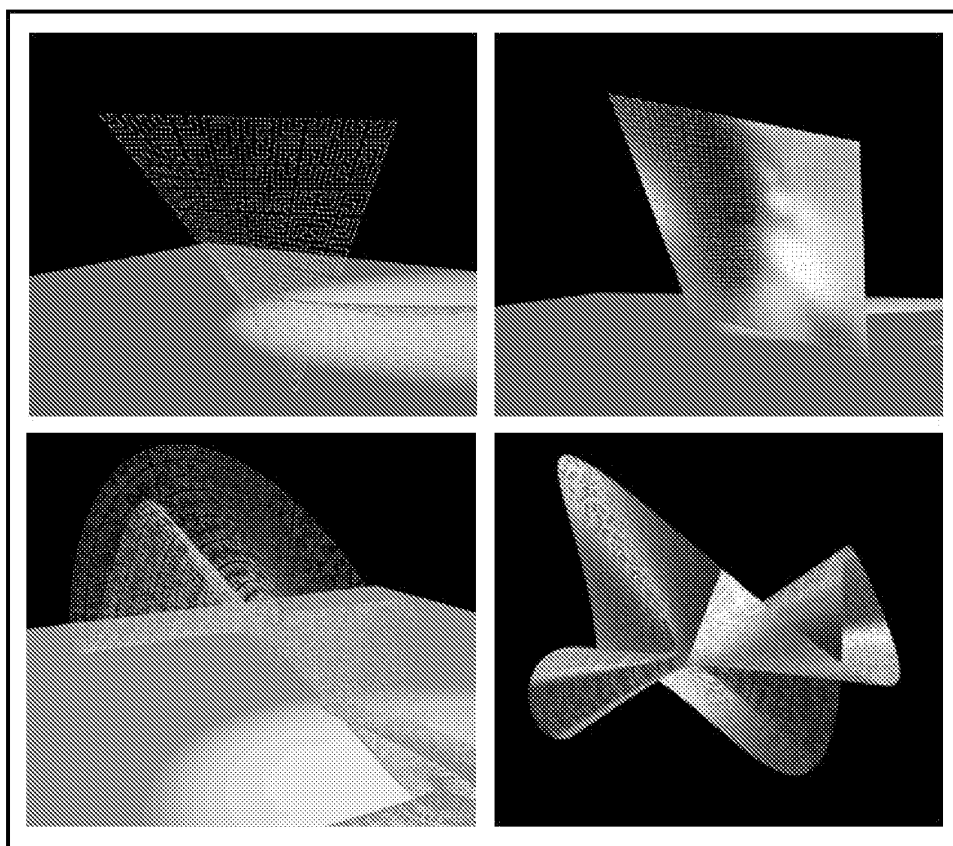


FIGURE 23

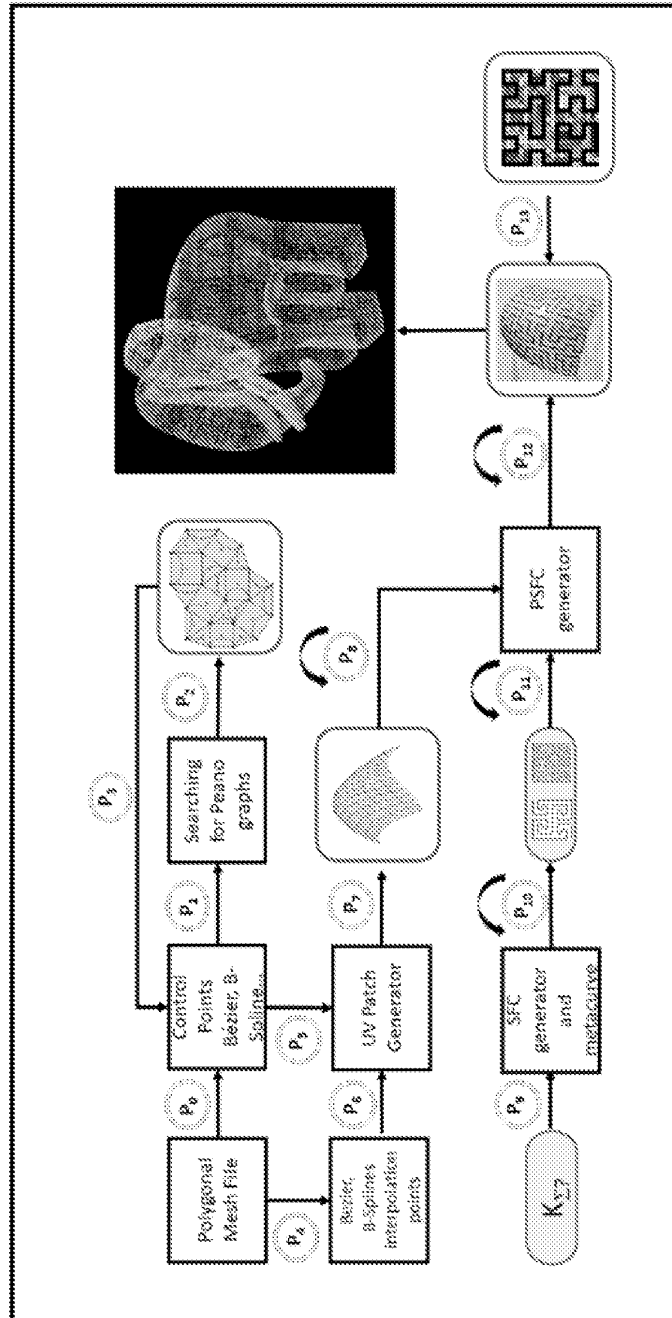


FIGURE 24

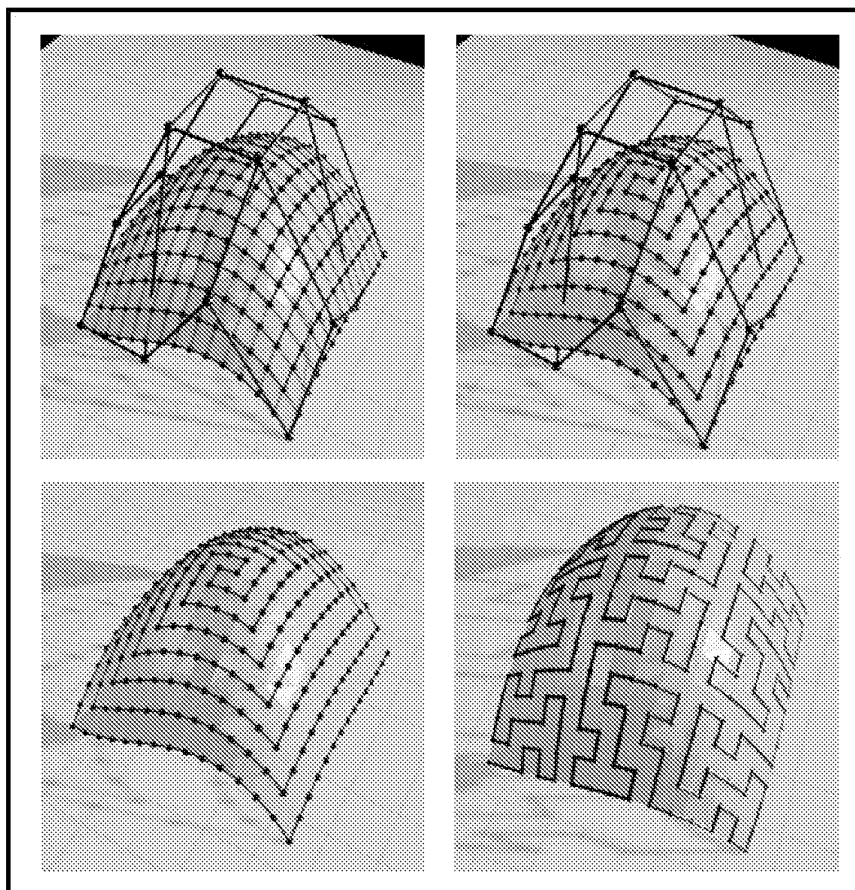


FIGURE 25

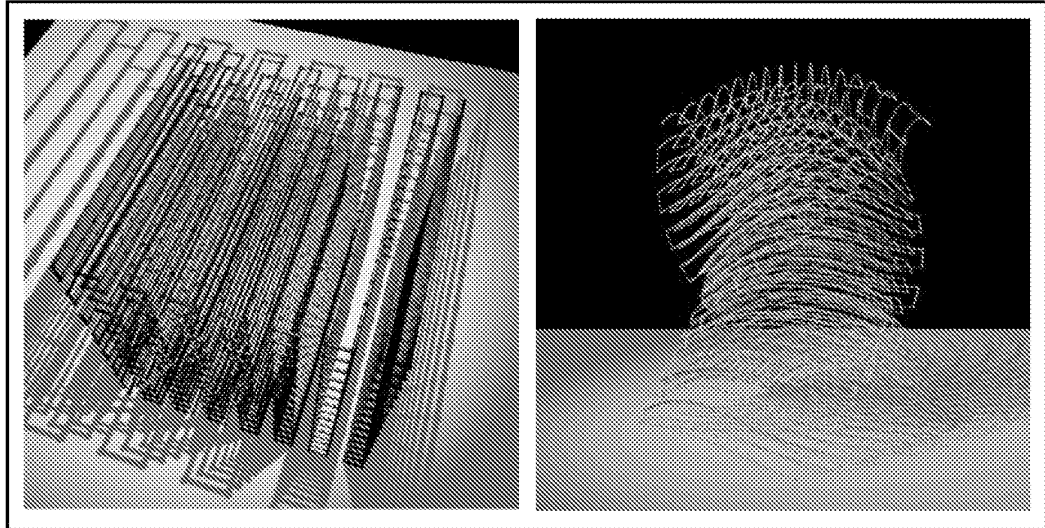


FIGURE 26

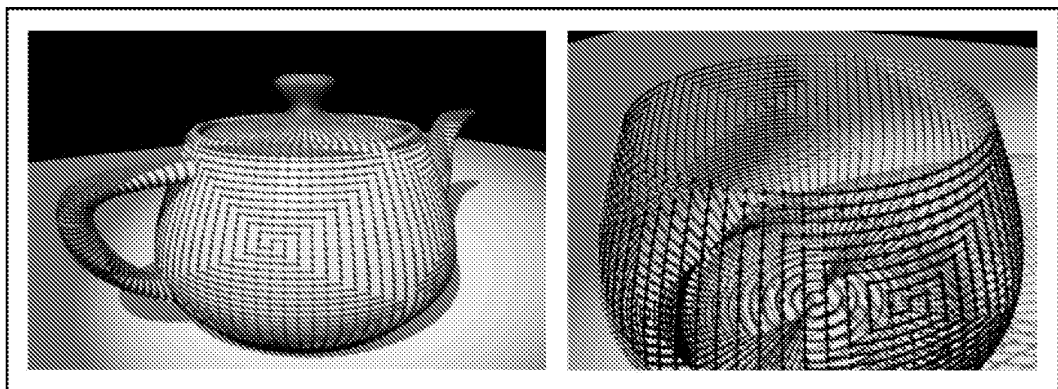


FIGURE 27

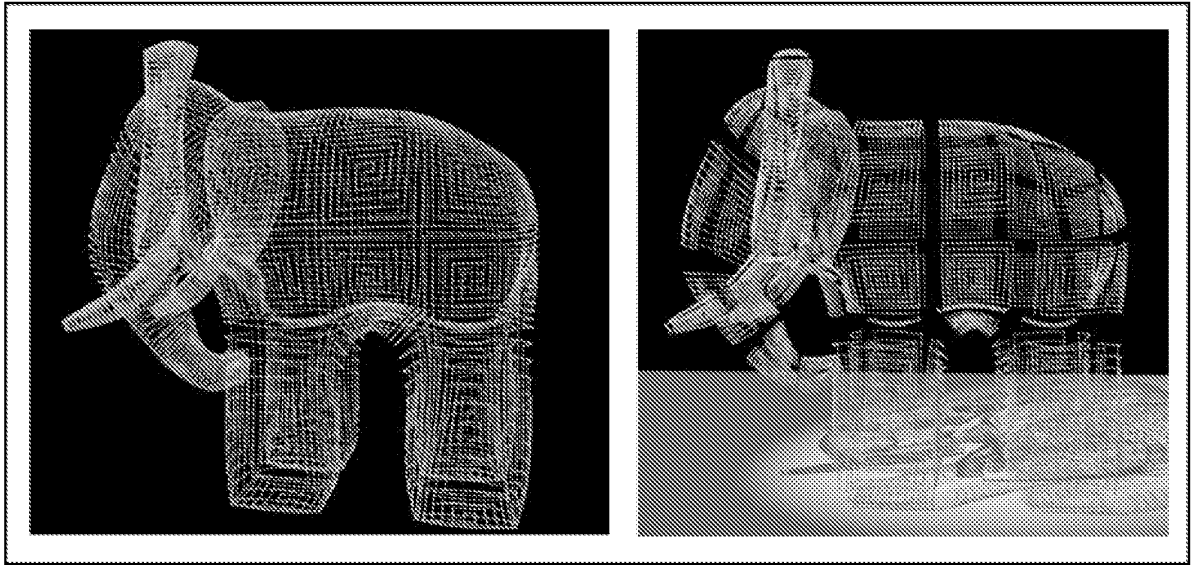


FIGURE 28

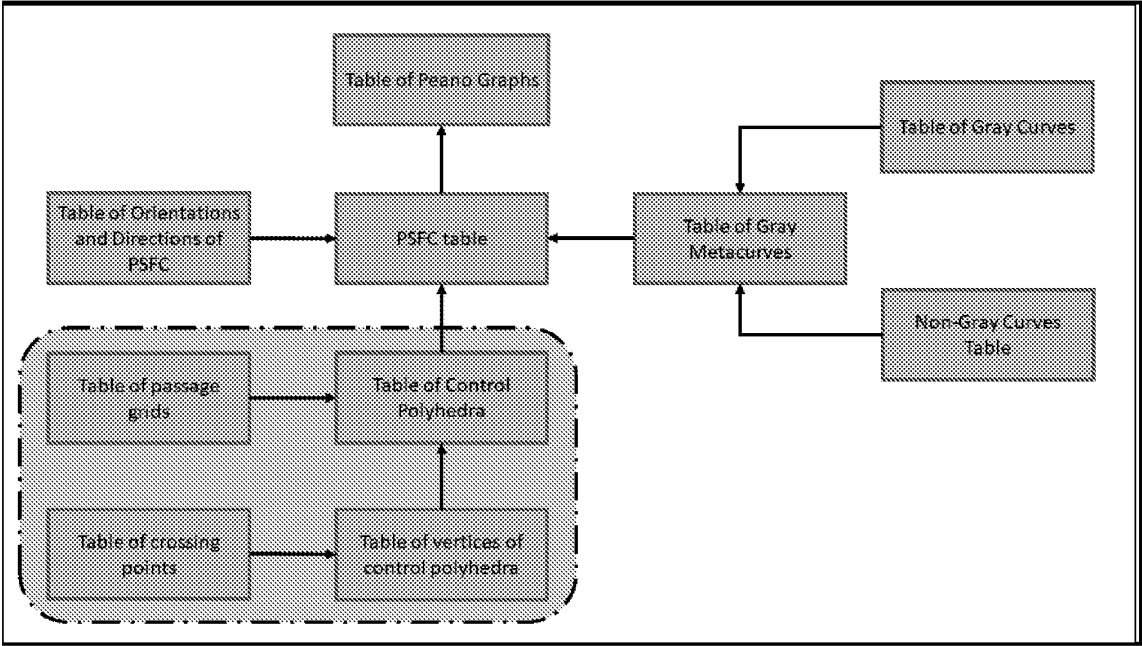


FIGURE 29

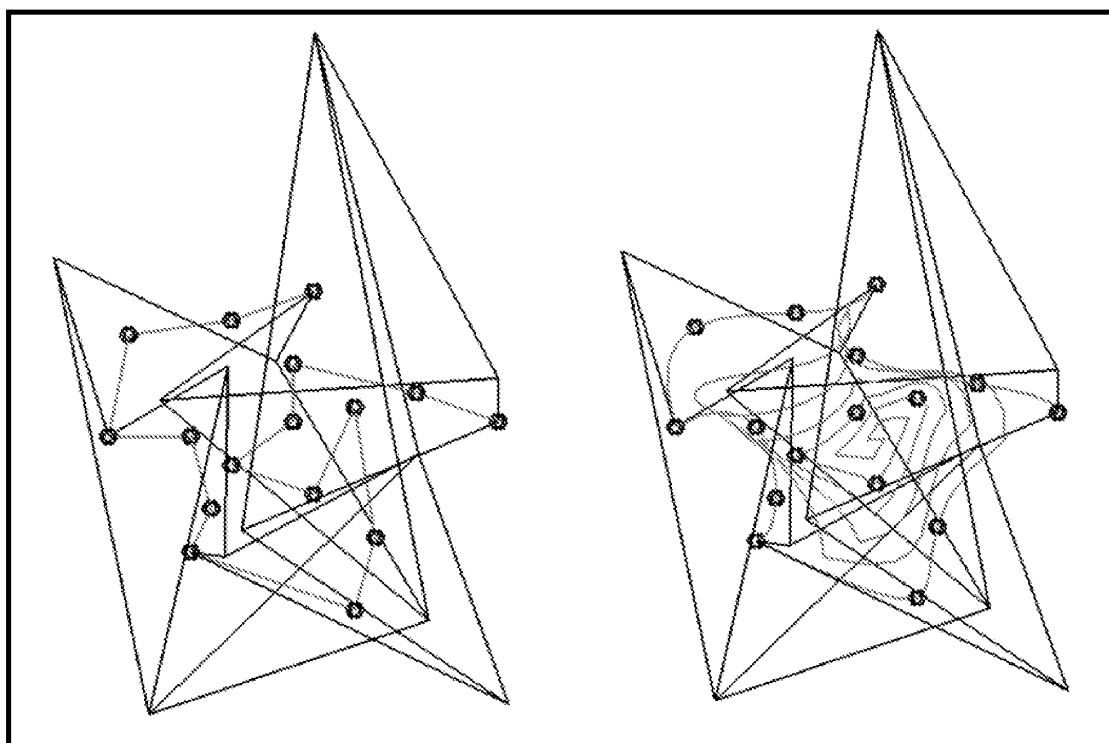


FIGURE 30

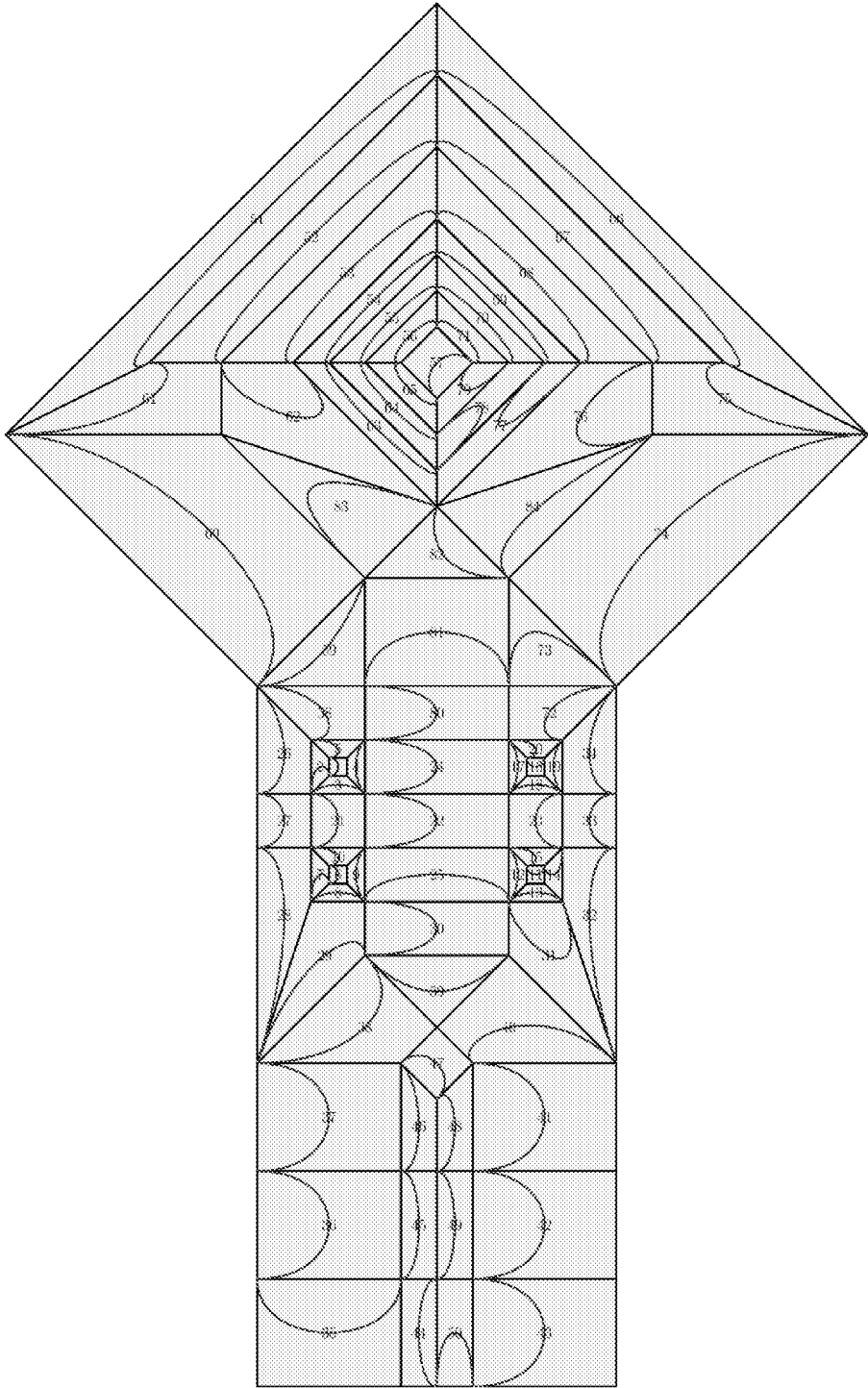


FIGURE 31

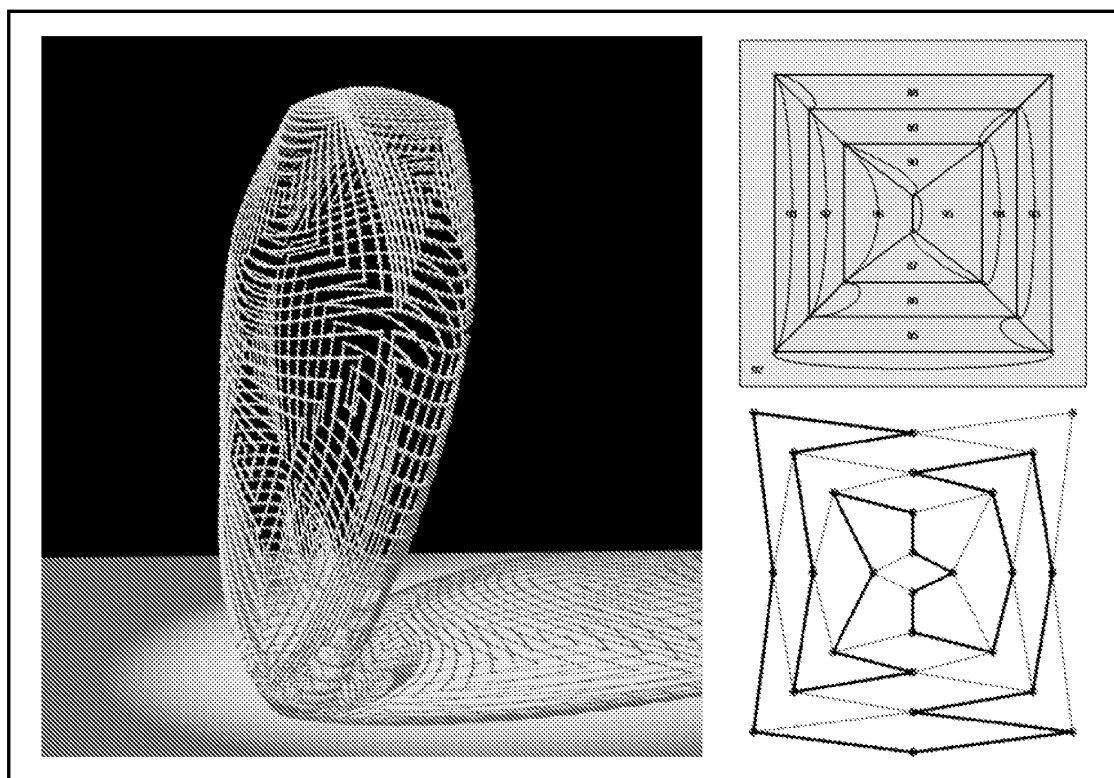


FIGURE 32

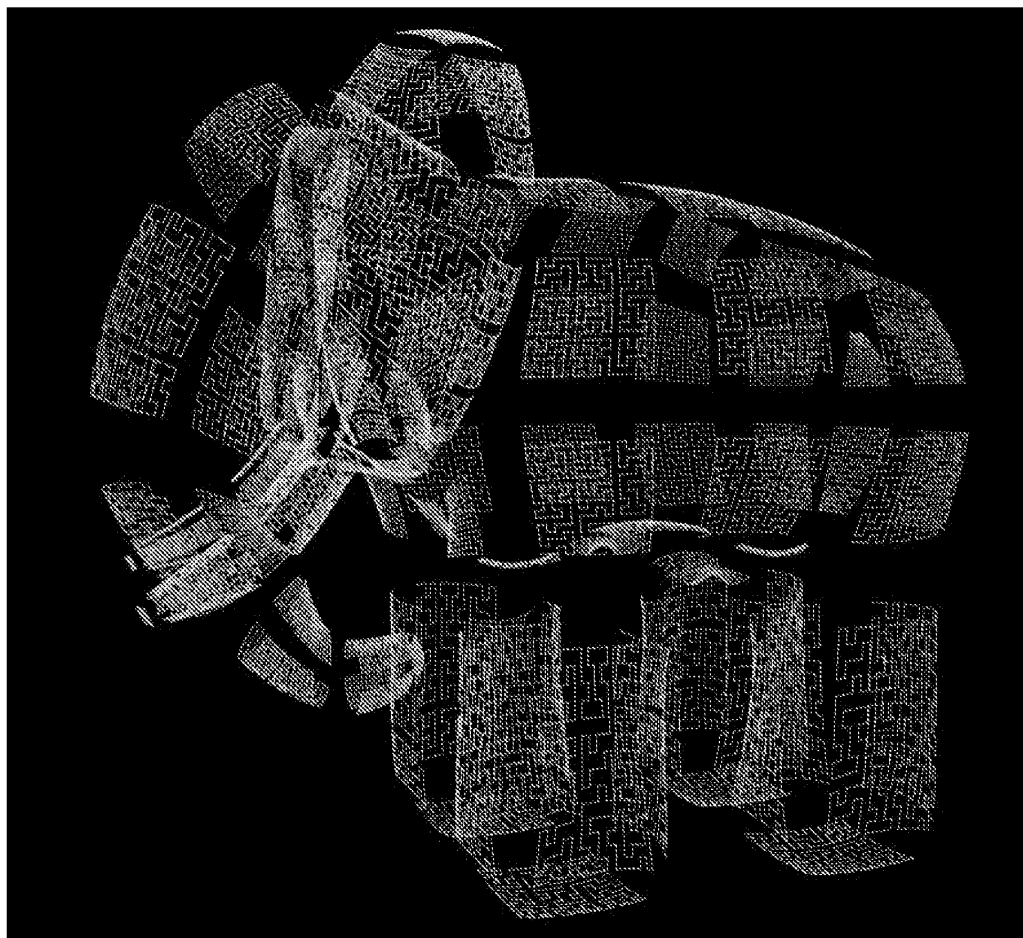


FIGURE 33

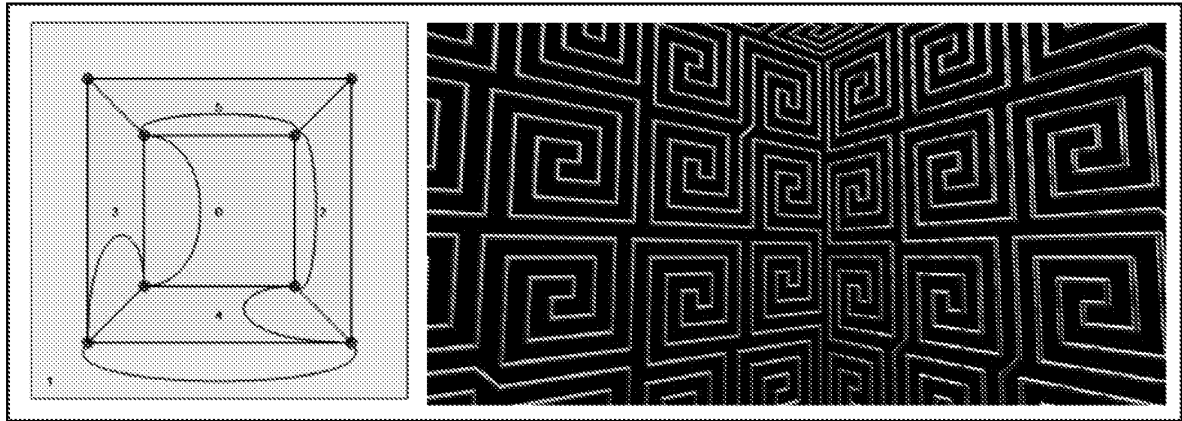


FIGURE 34

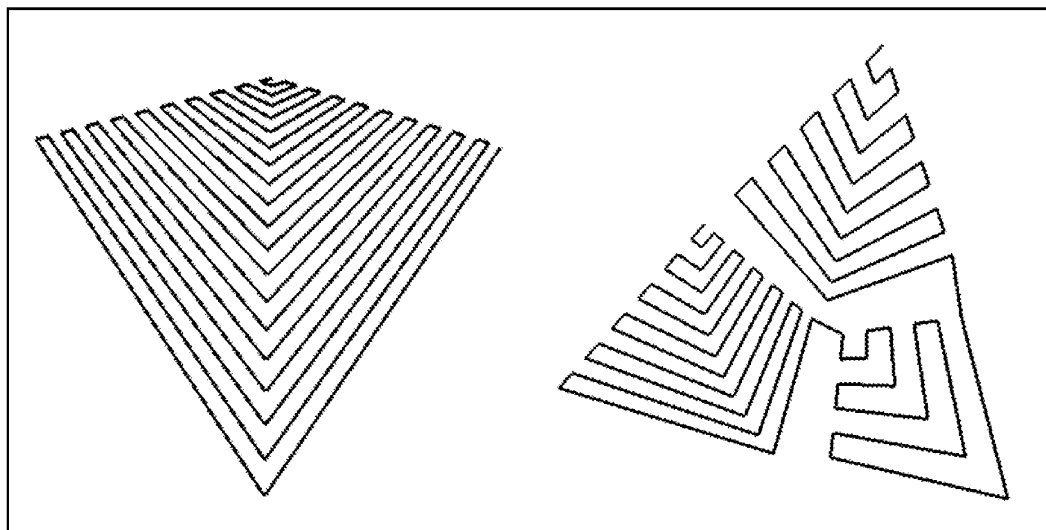


FIGURE 35

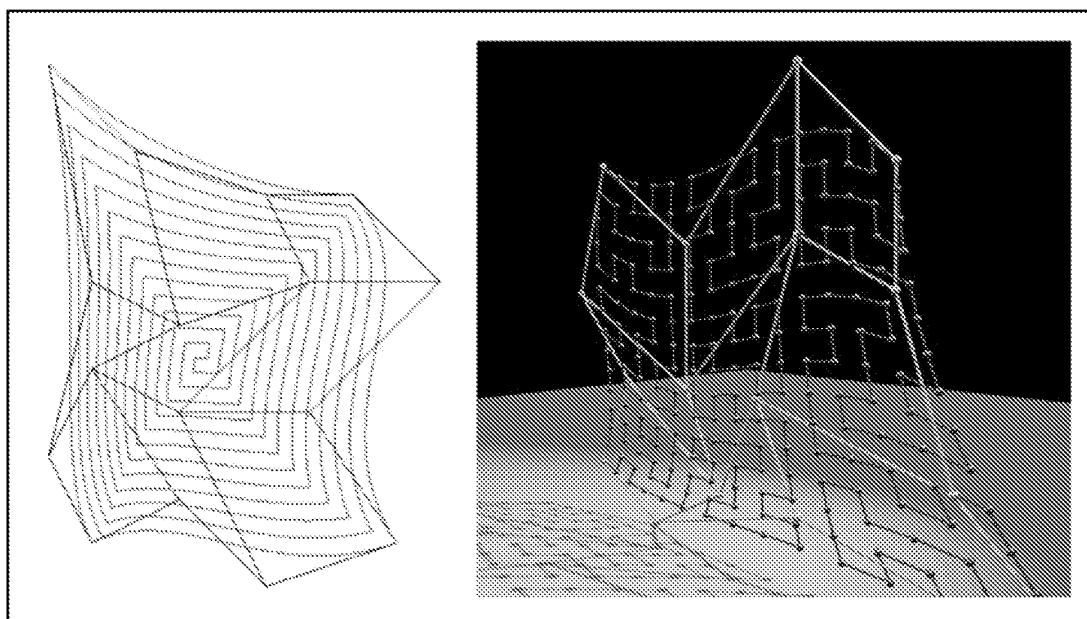


FIGURE 36

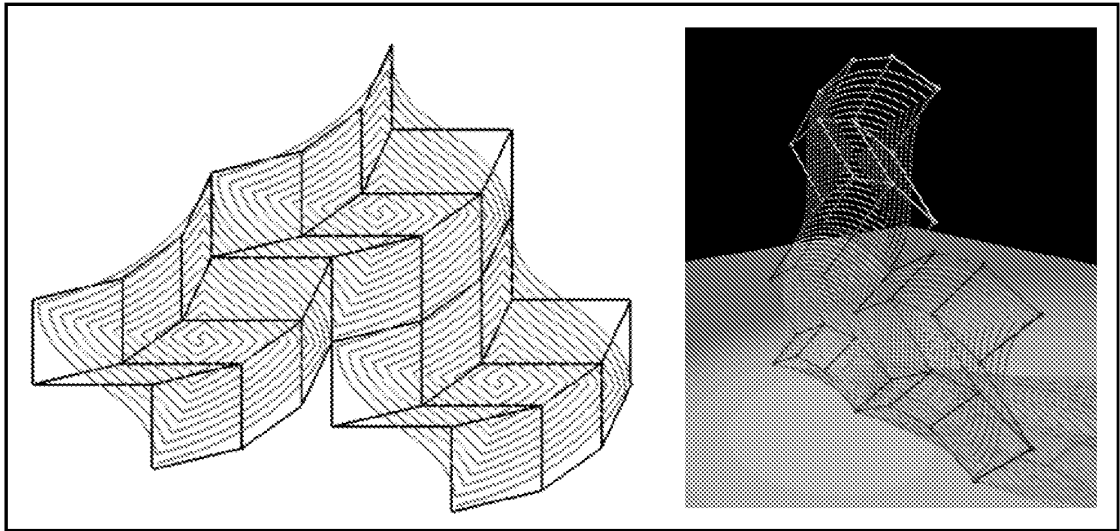


FIGURE 37

```
0 > parSPH:=proc(f,R,a,uMxMn,vMxMn)
1 > local ls,ms,uMx,vMx,uMn,vMn,
2 > flg,flt,i,j,v3;
3 > ls:=[ ];ms:=[ ];
4 > uMx:=uMxMn[1];
5 > uMn:=uMxMn[2];
6 > vMx:=vMxMn[1];
7 > vMn:=vMxMn[2];
8 > flg:=(uMx-uMn)/a;
9 > flt:=(vMx-vMn)/a;
10 > for i from 0 to a do
11 >   ms:=[ ];
12 >   for j from 0 to a do
13 >     v3:=tabSPH(f,R,[i*flg+uMn,j*flt+vMn]);
14 >     ms:=[op(ms),v3];
15 >   end do;
16 > ls:=[op(ls),op(ms)];
17 > end do;
18 > return(ls);
19 > end proc;
```

FIGURE 38

```
0 > sfcSPH:=proc(f,R,a,uMxMn,vMxMn)
1 > local ms,uMx,vMx,uMn,vMn,
2 > flg,flt,k,v2,i,j,v3;
3 > ms:=[];
4 > uMx:=uMxMn[1];
5 > uMn:=uMxMn[2];
6 > vMx:=vMxMn[1];
7 > vMn:=vMxMn[2];
8 > flg:=(uMx-uMn)/a;
9 > flt:=(vMx-vMn)/a;
10 > for k from 0 to (a+1)^2-1 do
11 > v2:=DCOmC(k,a);
12 > i:=v2[1];
13 > j:=v2[2];
14 > v3:=tabSPH(f,R,[i*flg+uMn,j*flt+vMn]);
15 > ms:=[op(ms),v3];
16 > end do;
17 > return(ms);
18 > end proc;
```

FIGURE 39

```
0 > tabSPH:=proc(f,R,v2)
1 > local v3;
2 > if f=0
3 > then v3:=SPH0(R,v2);
4 > return(v3);
5 > end if;
6 > if f=1
7 > then v3:=SPH1(R,v2);
8 > return(v3);
9 > end if;
10 > if f=2
11 > then v3:=SPH2(R,v2);
12 > return(v3);
13 > end if;
14 > if f=3
15 > then v3:=SPH3(R,v2);
16 > return(v3);
17 > end if;
18 > end proc;
```

FIGURE 40

```
0  > SPH0:=proc(r,v2)
1  > local u,v,cosU,sinU,cosV,sinV,x,y,z;
2  > u:=v2[1];
3  > v:=v2[2];
4  > cosU:=cos(u);
5  > sinU:=sin(u);
6  > cosV:=cos(v);
7  > sinV:=sin(v);
8  > x:=r*cosU*cosV;
9  > y:=r*sinU*cosV;
10 > z:=r*sinV;
11 > return([x,y,z]);
12 > end proc;
```

FIGURE 41

```
0 > SPH1:=proc(R,v2)
1 > local u,v,cosU,sinU,cosV,sinV,x,y,z;
2 > u:=v2[1];
3 > v:=v2[2];
4 > cosU:=cos(u);
5 > sinU:=sin(u);
6 > cosV:=cos(v);
7 > sinV:=sin(v);
8 > x:=R/2*(cosU+cosV);
9 > y:=R/2*(sinU+sinV);
10 > z:=R*sin((u-v)/2);
11 > return([x,y,z]);
12 > end proc;
```

FIGURE 42

```
0 > SPH2:=proc(R,v2)
1 > local u,v,x,y,z;
2 > u:=v2[1];
3 > v:=v2[2];
4 > x:=R*2*u/(1+u^2+v^2);
5 > y:=R*2*v/(1+u^2+v^2);
6 > z:=R*(1-u^2-v^2)/(1+u^2+v^2);
7 > return([x,y,z]);
8 > end proc;
```

FIGURE 43

```
0 > SPH3:=proc(R,v2)
1 > local u,v,x,y,z,sc;
2 > u:=v2[1];
3 > v:=v2[2];
4 > x:=R*(u^2-v^2);
5 > y:=R*2*u*v;
6 > z:=R*(u+v)^5;
7 > sc:=R/sqrt(x^2+y^2+z^2);
8 > return([x*sc,y*sc,z*sc]);
9 > end proc;
```

FIGURE 44

```
0  > DC0uwLL:=proc(ind,v2)
1  > local a,b,nbrCol,indCol,p2,
2  > ind0,indSeg,p1,p0,s,x,y;
3  > a:=v2[1];
4  > b:=v2[2];
5  > nbrCol:=(b+1)*2;
6  > indCol:=iquo(ind,nbrCol);
7  > p2:=indCol mod 2;
8  > ind0:=irem(ind,nbrCol);
9  > if p2 <> 0
10 > then ind0:=nbrCol-ind0-1;
11 > end if;
12 > indSeg:=iquo(ind0,2);
13 > p1:=indSeg mod 2;
14 > p0:=ind0 mod 2;
15 > if (p0+p1+p2) mod 2 = 0
16 > then s:=0;
17 > else s:=1;
18 > end if;
19 > x:=indCol*2+s;
20 > y:=iquo(ind0,2);
21 > return([x,y]);
22 > end proc;
```

FIGURE 45

```
0  > CD0uwLL:=proc(v2,v2ab)
1  > local x,y,a,b,nbrCol,indCol,
2  > p2,s,ind0,ind;
3  > x:=v2[1];
4  > y:=v2[2];
5  > a:=v2ab[1];
6  > b:=v2ab[2];
7  > nbrCol:=(b+1)*2;
8  > indCol:=iquo(x,2);
9  > p2:=indCol mod 2;
10 > if (x+y+p2) mod 2 = 0
11 >   then s:=0;
12 >   else s:=1;
13 > end if;
14 > ind0:=2*y+s;
15 > if p2 <> 0
16 >   then ind0:=nbrCol-ind0-1;
17 > end if;
18 > ind:=indCol*nbrCol+ind0;
19 > return(ind);
20 > end proc;
```

FIGURE 46

```

0  > PATCHuv:=proc(ls,n)
1  > local ms,ns,nuv,nvu,nbr,i,PAT,
2  > Gx,Gy,Gz,j,u,v,X,Y,Z;
3  > ms:=[];ns:=[];nuv:=[];nvu:=[];
4  > nbr:=nops(ls);
5  > for i from 0 to nbr-1 do
6  >   PAT:=ls[i+1];
7  >   ms:=[];
8  >   Gx:=[]; Gy:=[]; Gz:=[];
9  >   for j from 0 to 15 do
10 >    Gx:=[op(Gx),PAT[j+1][1]];
11 >    Gy:=[op(Gy),PAT[j+1][2]];
12 >    Gz:=[op(Gz),PAT[j+1][3]];
13 >   end do;
14 >   for u from 0 to n do
15 >    ms:=[];
16 >    for v from 0 to n do
17 >     X:=evalf(BEZeval(u/n,v/n,Gx));
18 >     Y:=evalf(BEZeval(u/n,v/n,Gy));
19 >     Z:=evalf(BEZeval(u/n,v/n,Gz));
20 >     ms:=[op(ms),[X,Y,Z]];
21 >    end do;
22 >    nuv:=[op(nuv),ms];
23 >   end do;
24 >   for v from 0 to n do
25 >    ns:=[];
26 >    for u from 0 to n do
27 >     X:=evalf(BEZeval(u/n,v/n,Gx));
28 >     Y:=evalf(BEZeval(u/n,v/n,Gy));
29 >     Z:=evalf(BEZeval(u/n,v/n,Gz));
30 >     ns:=[op(ns),[X,Y,Z]];
31 >    end do;
32 >    nvu:=[op(nvu),ns];
33 >   end do;
34 >   return([op(nuv),op(nvu)]);
35 > end do;
36 > end proc:

```

FIGURE 47

```

0  > PATCHsfc:=proc(ls , a , b , lsOR , lsINV)
1  > local ns , c , cc , nbr , i , r , inv , PAT , ms ,
2  > Gx , Gy , Gz , j , k , ind , v2 , ut , vt , X , Y , Z ;
3  > ns := [] ;
4  > c := a + 1 ;
5  > cc := c * c ;
6  > nbr := nops ( ls ) ;
7  > for i from 0 to nbr - 1 do
8  >   r := lsOR [ i + 1 ] ;
9  >   inv := lsINV [ i + 1 ] ;
10 >   PAT := ls [ i + 1 ] ;
11 >   ms := [] ;
12 >   Gx := [] ; Gy := [] ; Gz := [] ;
13 >   for j from 0 to 15 do
14 >     Gx := [ op ( Gx ) , PAT [ j + 1 ] [ 1 ] ] ;
15 >     Gy := [ op ( Gy ) , PAT [ j + 1 ] [ 2 ] ] ;
16 >     Gz := [ op ( Gz ) , PAT [ j + 1 ] [ 3 ] ] ;
17 >   end do ;
18 >   for k from 0 to cc - 1 do
19 >     if inv = 0
20 >       then ind := k ;
21 >       else ind := cc - 1 - k ;
22 >     end if ;
23 >     v2 := PTsfc ( ind , a , b , r ) ;
24 >     ut := v2 [ 1 ] ;
25 >     vt := v2 [ 2 ] ;
26 >     X := BEZeval ( ut , vt , Gx ) ;
27 >     Y := BEZeval ( ut , vt , Gy ) ;
28 >     Z := BEZeval ( ut , vt , Gz ) ;
29 >     ms := [ op ( ms ) , [ X , Y , Z ] ] ;
30 >   end do ;
31 >   ns := [ op ( ns ) , ms ] ;
32 > end do ;
33 > end proc ;

```

FIGURE 48

```
0 > PTsfc:=proc(ind,a,b,r)
1 > local uv,x,y,ur,vr,u,v,ut,vt,X,Y,Z,v2;
2 > uv:=DC0mC(ind,a);
3 > x:=uv[1];y:=uv[2];
4 > if r=0 then ur:=x; vr:=y; end if;
5 > if r=1 then ur:=a-y; vr:=x; end if;
6 > if r=2 then ur:=a-x; vr:=a-y; end if;
7 > if r=3 then ur:=y; vr:=a-x; end if;
8 > u:=evalf(ur/a);
9 > v:=evalf(vr/a);
10 > ut:=evalf(1/2*(2*b*u-b+a)/a);
11 > vt:=evalf(1/2*(a-b+2*b*v)/a);
12 > v2:=[ut,vt];
13 > return(v2);
14 > end proc;
```

FIGURE 49

```
0 > INTERPOL:=proc(lsPTSpas)
1 > local ms,i,lsX,lsY,lsZ,Xls,Yls,Zls,x;
2 > ms:=[];
3 > lsX:=[];
4 > lsY:=[];
5 > lsZ:=[];
6 > for i from 1 to 16 do
7 >   lsX:=[op(lsX),lsPTSpas[i][1]];
8 >   lsY:=[op(lsY),lsPTSpas[i][2]];
9 >   lsZ:=[op(lsZ),lsPTSpas[i][3]];
10 > end do;
11 > Xls:=SOLUTana(lsX);
12 > Yls:=SOLUTana(lsY);
13 > Zls:=SOLUTana(lsZ);
14 > for i from 1 to 16 do
15 >   x:=[Xls[i],Yls[i],Zls[i]];
16 >   ms:=[op(ms),x];
17 > end do;
18 > return(ms);
19 > end proc;
```

FIGURE 50

```

0  > SOLUTana:=proc(l s)
1  > local l sQ,Q0,Q1,Q2,Q3,Q4,Q5,Q6,Q7,
2  > Q8,Q9,Q10,Q11,Q12,Q13,Q14,Q15,
3  > Q5a,Q5b,Q5c,Q5d,Q6a,Q6b,Q6c,Q6d,
4  > Q9a,Q9b,Q9c,Q9d,Q10a,Q10b,Q10c,Q10d;
5  > l sQ:=[];
6  > Q0:=l s[1];
7  > Q1:=-5/6*l s[1]+3*l s[2]-3/2*l s[3]+1/3*l s[4];
8  > Q2:=1/3*l s[1]-3/2*l s[2]+3*l s[3]-5/6*l s[4];
9  > Q3:=l s[4];
10 > Q4:=-5/6*l s[1]+3*l s[5]-3/2*l s[9]+1/3*l s[13];
11 > Q5a:=25/36*l s[1]-5/2*l s[2]+5/4*l s[3]-5/18*l s[4];
12 > Q5b:=-5/2*l s[5]+9*l s[6]-9/2*l s[7]+l s[8];
13 > Q5c:=5/4*l s[9]-9/2*l s[10]+9/4*l s[11]-1/2*l s[12];
14 > Q5d:=-5/18*l s[13]+l s[14]-1/2*l s[15]+1/9*l s[16];
15 > Q5:=Q5a+Q5b+Q5c+Q5d;
16 > Q6a:=-5/18*l s[1]+5/4*l s[2]-5/2*l s[3]+25/36*l s[4];
17 > Q6b:=l s[5]-9/2*l s[6]+9*l s[7]-5/2*l s[8];
18 > Q6c:=-1/2*l s[9]+9/4*l s[10]-9/2*l s[11]+5/4*l s[12];
19 > Q6d:=1/9*l s[13]-1/2*l s[14]+l s[15]-5/18*l s[16];
20 > Q6:=Q6a+Q6b+Q6c+Q6d;
21 > Q7:=-5/6*l s[4]+3*l s[8]-3/2*l s[12]+1/3*l s[16];
22 > Q8:=1/3*l s[1]-3/2*l s[5]+3*l s[9]-5/6*l s[13];
23 > Q9a:=-5/18*l s[1]+l s[2]-1/2*l s[3]+1/9*l s[4];
24 > Q9b:=5/4*l s[5]-9/2*l s[6]+9/4*l s[7]-1/2*l s[8];
25 > Q9c:=-5/2*l s[9]+9*l s[10]-9/2*l s[11]+l s[12];
26 > Q9d:=25/36*l s[13]-5/2*l s[14]+5/4*l s[15]-5/18*l s[16];
27 > Q9:=Q9a+Q9b+Q9c+Q9d;
28 > Q10a:=1/9*l s[1]-1/2*l s[2]+l s[3]-5/18*l s[4];
29 > Q10b:=-1/2*l s[5]+9/4*l s[6]-9/2*l s[7]+5/4*l s[8];
30 > Q10c:=l s[9]-9/2*l s[10]+9*l s[11]-5/2*l s[12];
31 > Q10d:=-5/18*l s[13]+5/4*l s[14]-5/2*l s[15]+25/36*l s[16];
32 > Q10:=Q10a+Q10b+Q10c+Q10d;
33 > Q11:=1/3*l s[4]-3/2*l s[8]+3*l s[12]-5/6*l s[16];
34 > Q12:=l s[13];
35 > Q13:=-5/6*l s[13]+3*l s[14]-3/2*l s[15]+1/3*l s[16];
36 > Q14:=1/3*l s[13]-3/2*l s[14]+3*l s[15]-5/6*l s[16];
37 > Q15:=l s[16];
38 > l sQ:=[Q0,Q1,Q2,Q3,Q4,Q5,Q6,Q7,
39 > Q8,Q9,Q10,Q11,Q12,Q13,Q14,Q15];
40 > end proc;

```

FIGURE 51

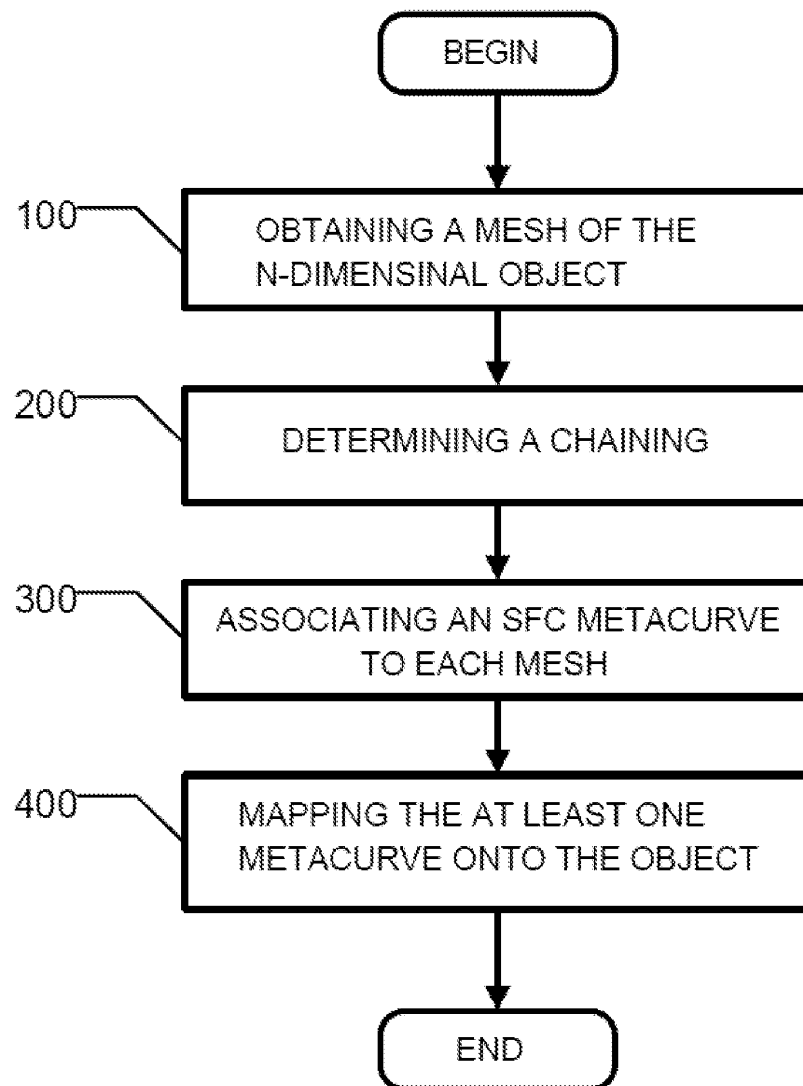


FIGURE 52

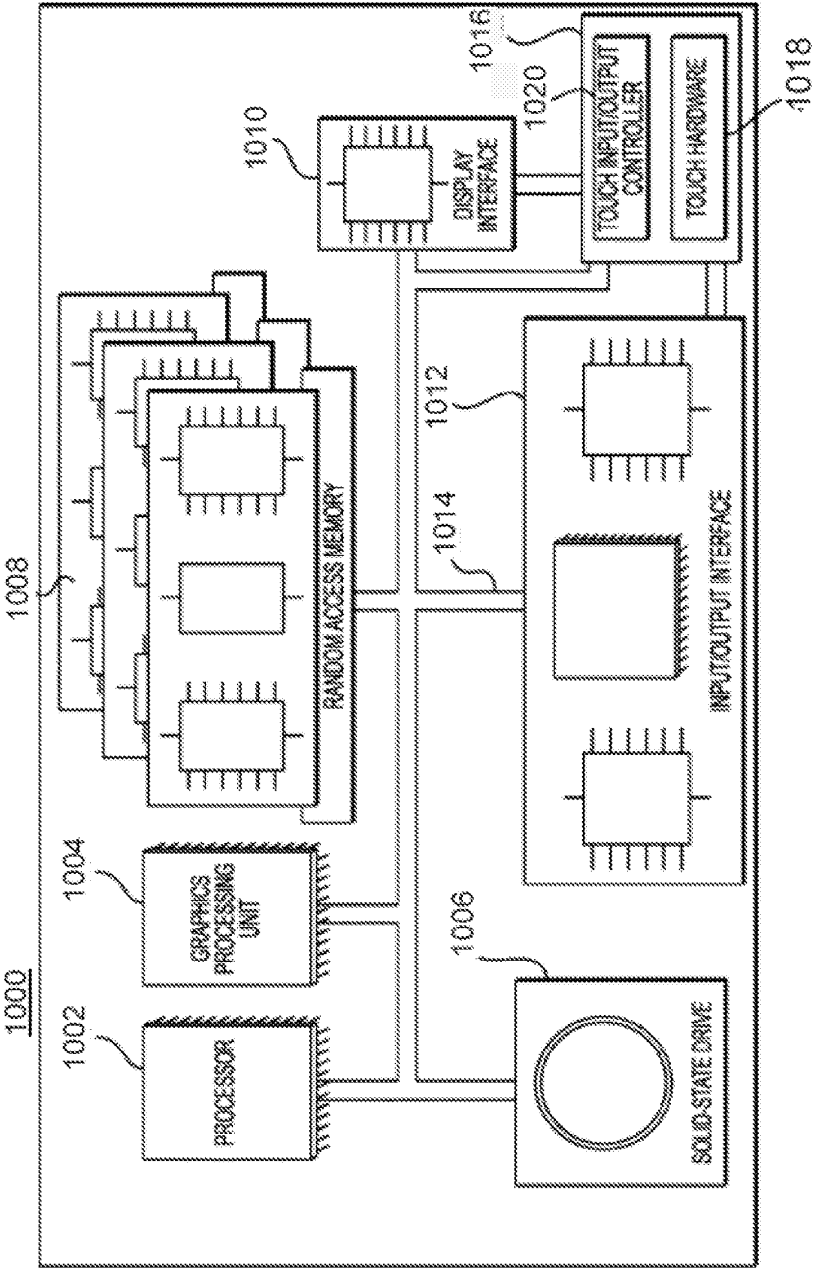


FIGURE 53