

2.6.2 - MODELOS DE ALAMBRE CON LINEAS OCULTAS

Así como en la perspectiva axonométrica podíamos prescindir de la coordenada $Z2(I,J)$, para la perspectiva cónica no podemos prescindir puesto que va a ser utilizada para hallar las coordenadas definitivas.

Veamos el algoritmo de cálculo a utilizar desarrollado a partir de las ecuaciones analíticas:

Primer giro

$$\begin{aligned}X1(I, J) &= X0(I, J) * \text{COS}(A) - Y0(I, J) * \text{SIN}(A) \\Y1(I, J) &= X0(I, J) * \text{SIN}(A) + Y0(I, J) * \text{COS}(A) \\Z1(I, J) &= Z0(I, J)\end{aligned}$$

Segundo giro

$$\begin{aligned}X2(I, J) &= X1(I, J) \\Y2(I, J) &= Z1(I, J) * \text{SIN}(M / 2) + Y1(I, J) * \text{COS}(M / 2) \\Z2(I, J) &= Z1(I, J) * \text{COS}(M / 2) - Y1(I, J) * \text{SIN}(M / 2)\end{aligned}$$

Ecuación de la recta proyectante que pasa por $V(AX, BX, CX)$ y por $[X2(I, J), Y2(I, J), Z2(I, J)]$.

Vectores directores de la recta en el espacio:

$$\begin{aligned}V1 &= X2(I, J) - AX \\V2 &= Y2(I, J) - BX \\V3 &= Z2(I, J) - CX\end{aligned}$$

Ecuación continua en el espacio:

$$(X - AX) / V1 = (Y - BX) / V2 = (Z - CX) / V3$$

Intersección con el plano del cuadro J de altura ZX:

$$\begin{aligned}X3(I, J) &= (V1 / V3) * (ZX - CS) + AX \\Y3(I, J) &= (V2 / V3) * (ZX - CS) + BX \\Z3 &= ZX\end{aligned}$$

Para que la perspectiva cónica no sea deformada, tenemos que asegurarnos que:

- La altura del punto de vista sea mayor que la de todos los puntos de la malla una vez girada (fácil, pues todas las alturas serán menores que cero, véase figura 2-6-1-5).

- El ángulo de visión con que abarcamos la figura con respecto a la perpendicular desde el punto de vista al cuadro debe ser inferior a 30° (cono de visión óptima en perspectivas cónicas).

Esta segunda condición podemos alterarla si deseamos ver la malla deformada como sucede a veces para conseguir determinados efectos.

Si para conservar memoria queremos usar sólo dos variables indexadas, en lugar de las cuatro utilizadas para este menester tendríamos:

$$X2(I,J) = \left[\frac{[X0(I,J) * \cos(A) - Y0(I,J) * \sin(A) - AX]}{[Z0(I,J) * \cos(M/2) - X0(I,J) * \sin(A) * \sin(M/2) - Y0(I,J) * \cos(A) * \sin(M/2) - CX]} \right] * (ZX - CX) + AX$$

$$Y2(I,J) = \left[\frac{[Z0(I,J) * \sin(M/2) + X0(I,J) * \sin(A) * \cos(M/2) + Y0(I,J) * \cos(A) * \sin(M/2) - BX]}{[Z0(I,J) * \cos(M/2) - X0(I,J) * \sin(A) * \sin(M/2) - Y0(I,J) * \cos(A) * \sin(M/2) - CX]} \right] * (ZX - CX) + BX$$

Y utilizando variables intermedias:

$$A = X0(I,J) \quad B = Y0(I,J) \quad C = Z0(I,J)$$

$$SI = \sin(A) \quad CO = \cos(A) \quad SS = \sin(M/2) \quad CC = \cos(M/2)$$

$$X1 = A * CO - B * SI$$

$$Y1 = A * SI + B * CO$$

$$Y2 = C * SS + Y1 * CC$$

$$Z2 = C * CC - Y1 * SS$$

$$V1 = X1 - AX \quad V2 = Y2 - BX \quad V3 = Z2 - CX$$

$$X2(I,J) = (V1/V3) * (ZX - CX) + AX$$

$$Y2(I,J) = (V2/V3) * (ZX - CX) + BX$$

Para obtener el dibujo centrado en pantalla bastaría utilizar como en axonometría el vector posición para el centrado (CE,DE) con lo que las nuevas coordenadas serían:

$$X2(I,J) = X2(I,J) + CE$$

$$Y2(I,J) = Y2(I,J) + DE$$

Que a la hora de dibujar pasarían a ser:

$$X = X2(I,J)$$

$$Y = 200 - Y2(I,J)$$

Debido al sentido inverso de las coordenadas en pantalla.

En estas circunstancias el algoritmo de cálculo de las coordenadas sería:

```

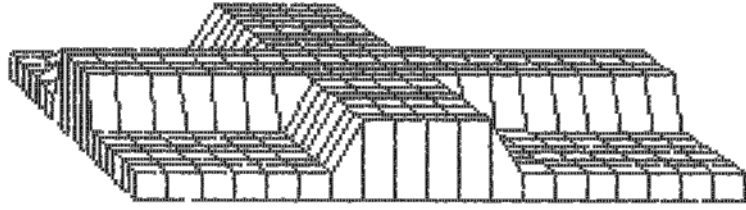
1000 REM SEAN X0(I,J), Y0(I,J), Z0(I,J) LAS COORDENADAS PRIMITIVAS DE LOS
      VÉRTICES DE LA MALLA, (AX, BX, CX) COORD. PTO DE VISTA, ZX ALTURA
      CUADRO
1010 REM SEA A EL ANGULO ADOPTADO PARA EL PRIMER GIRO Y M/2 EL ANGULO
      ADOPTADO PARA EL SEGUNDO GIRO (EN GRADOS SEXAGESIMALES)
1020 A=A*M7180 : B=M/2
1030 CO=COS(A) : SI=SIN(A) : CC=COS(B) : SS=SIN(B)
1040 XC = ((X0(10,10)*CO-Y0(10,10)*SI*SS-AX)/(Z0(10,10))*CC-X0(10,10)*SI*SS-Y0(10,10)
      *CO*SS-CX))*(ZX-CX)+AX : CE=150-XC
1050 YC = ((Z0(20,20) * SS+X0(20,20) * SI * CC+Y0(20,20) * CO * SS-BX)/(Z0(20,20) * CC-
      X0(20,20)*SI*SS-Y0(20,20)*CO*SS-CX))*(ZX-CX)+BX
1060 DE=100-YC
1070 FOR I=1 TO 20 : FOR J=1 TO 20
1080 A=X0(I,J) : B=Y0(I,J) : C=Z0(I,J)
1090 X1=A*CO-B*SI : Y1=A*SI+B*CO
1100 Y2=C*SS+Y1*CC : Z2=C*CC-Y1*SS
1110 V1=X1-AX : V2=Y2-BX : V3=Z2-CX
1120 X2(I,J)=(V1/V3)*(ZX-CX)+AX+CE
1130 Y2(I,J)=(V2/V3)*(ZX-CX)+AX+CE
1140 NEXT J : NEXT I

```

Para dibujar la malla sin considerar líneas ocultas, sirven los mismos algoritmos utilizados en axonometría, no así en la eliminación de líneas ocultas, donde este proceso resulta más laborioso.

10	10	10	10	10	10	10	10	30	30	30	30	30	10	10	10	10	10	10	10
10	10	10	10	10	10	10	10	30	30	30	30	30	10	10	10	10	10	10	10
10	10	10	10	10	10	10	10	30	30	30	30	30	10	10	10	10	10	10	10
10	10	10	10	10	10	10	10	30	30	30	30	30	10	10	10	10	10	10	10
10	10	10	10	10	10	10	10	30	30	30	30	30	10	10	10	10	10	10	10
10	10	10	10	10	10	10	10	30	30	30	30	30	10	10	10	10	10	10	10
10	10	10	10	10	10	10	10	30	30	30	30	30	10	10	10	10	10	10	10
30	30	30	30	30	30	30	30	30	30	30	30	30	30	30	30	30	30	30	30
30	30	30	30	30	30	30	30	30	30	30	30	30	30	30	30	30	30	30	30
30	30	30	30	30	30	30	30	30	30	30	30	30	30	30	30	30	30	30	30
30	30	30	30	30	30	30	30	30	30	30	30	30	30	30	30	30	30	30	30
30	30	30	30	30	30	30	30	30	30	30	30	30	30	30	30	30	30	30	30
10	10	10	10	10	10	10	10	30	30	30	30	30	10	10	10	10	10	10	10
10	10	10	10	10	10	10	10	30	30	30	30	30	10	10	10	10	10	10	10
10	10	10	10	10	10	10	10	30	30	30	30	30	10	10	10	10	10	10	10
10	10	10	10	10	10	10	10	30	30	30	30	30	10	10	10	10	10	10	10
10	10	10	10	10	10	10	10	30	30	30	30	30	10	10	10	10	10	10	10
10	10	10	10	10	10	10	10	30	30	30	30	30	10	10	10	10	10	10	10

MATRIZ DE RESULTADOS ANALÍTICOS CORRESPONDIENTE A UNA CRUZ GENERADA MEDIANTE PERFIL DESLIZANTE



SIMULACRO DE PERSPECTIVA CABALLERA

DATOS DE LA PERSPECTIVA:

GIRO 0°

COORDENADAS DEL PUNTO DE VISTA (-400,600,2000)

ALTURA DEL PLANO DEL CUADRO (0)

Es interesante observar que si el primer giro es de cero grados, la perspectiva cónica resultante es central (de un solo punto de fuga) y si el punto de vista se elige muy elevado, el resultado puede considerarse una perspectiva caballera cuyos aspectos visuales son los de la figura 2-6-1-6.

Cuando el punto de vista se elige muy elevado, la proyección que se obtiene es en general una perspectiva paralela oblicua (en rigor esto sucedería cuando el punto de vista estuviese en el infinito).

2.6.3 - ELIMINACIÓN DE LÍNEAS OCULTAS

Así como en axonometría no existe dificultad en determinar cual es la zona lateral visible desde el observador (siempre son los laterales más lejanos al plano del cuadro), en el sistema central o cónico no es tan sencillo, pues depende de la posición en que estemos situados como observador (posición del punto de vista), además de la posición de la malla.

Veamos un ejemplo gráfico que ilustra muy bien la afirmación anterior.

Sea la figura 2-6-3-1 en la que aparecen las posiciones relativas de la malla, plano del cuadro PC y punto de vista (el plano del cuadro aparece vertical y la malla horizontal).

Desde el punto de vista V1 veríamos los laterales 1 y 2. Desde el V2 los laterales 1 y 4. Desde el V3 los laterales 2 y 3. Desde le V4 sólo el lateral 1 y, desde el V5, el lateral 2.

Naturalmente estos puntos de vista no cumplen la condición de mantener la malla dentro del cono de visión óptima

A medida que aumentásemos la distancia del punto de vista al plano del cuadro, la tendencia a penetrar en la zona limitada por los laterales 1 y 2 de la malla sería mayor, por lo que en las mallas no deformadas podemos considerar siempre esa zona o como máximo las colindantes (franjas desde las que se ve solamente un lateral).

Tendremos pues que asegurarnos de que el punto de vista se encuentra en esas zonas y dibujar o no el lateral en función de cual sea la zona en que nos encontramos.

Como algoritmo básico para eliminación de líneas ocultas podemos utilizar el mismo que ya hemos empleado en axonometría partiendo del vértice con que lo hacíamos para ese tipo de perspectivas.

Veamos la forma de determinar el lateral que no se ve en función de las coordenadas del punto de vista y de la malla (antes de centrar).

Los puntos que vamos a utilizar para hacer esa discusión son el vértice (1,1), el vértice (20,1) y el vértice (1,20) de la malla una vez realizados los giros de ángulo A y B (recordemos que $B=M/2$).

La figura 2-6-3-2 ilustra la posición de esos puntos con respecto a los ejes de pantalla.

En caso que V esté por encima de la recta que pasa por (1,20) y (1,1) y también por encima de la recta que pasa por (1,1) y (20,1) estaremos en el caso de visión de los dos laterales.

Si está por encima de una recta y por debajo de otra, sólo veremos un lateral. Bastará considerar las coordenadas X y Z de cada punto ya que de otra manera manejaríamos planos (en realidad cada recta representará un plano que divide el espacio en dos partes).

Las coordenadas del punto serán:

Para el punto V

$$(AX, CX)$$

Para el punto (1,1)

$$X(1,1)=0Z(1,1)=0$$

Para el punto (20,1)

$$\begin{aligned} X(20,1) &= X0(20,1)*CO - Y0(20,1)*SI \\ Z(20,1) &= Z0(20,1)*CC - X0(20,1)*SI*SS - Y0(20,1)*CO*SS \end{aligned}$$

Para simplificar vamos a cambiar de variables:

$$\begin{aligned} A1 &= X(1,1) & A2 &= Z(1,1) \\ B1 &= X(20,1) & B2 &= Z(20,1) \\ C1 &= X(1,20) & C2 &= Z(1,20) \end{aligned}$$

PLANO DEL CUADRO

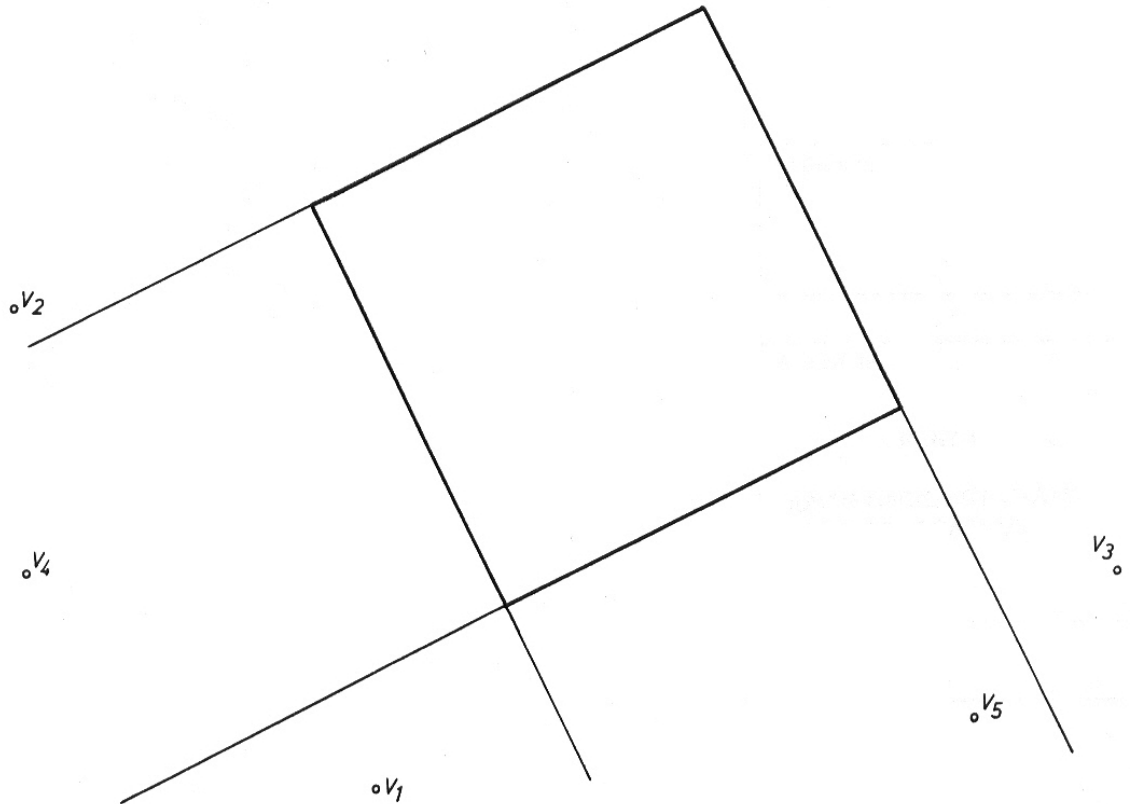


FIGURA 2-6-3-1

Las ecuaciones de las dos rectas serán:

$$V1=A1-B1 : V2=A2-B2 : V3=V2/V1 : V4=V3*A1 : V5=A2-V4$$

$$Z(T)=V3*T+V5$$

$$U1=A1-C1 : U2=A2-C2 : U3=U2/U1 : U4=U3*A1 : U5=A2-U4$$

$$Z8T)=U3*T+U5$$

Sustituyendo T por AX en las dos ecuaciones se obtiene la posición que debemos comparar con CX. Vamos a darle nombre a esas variables.

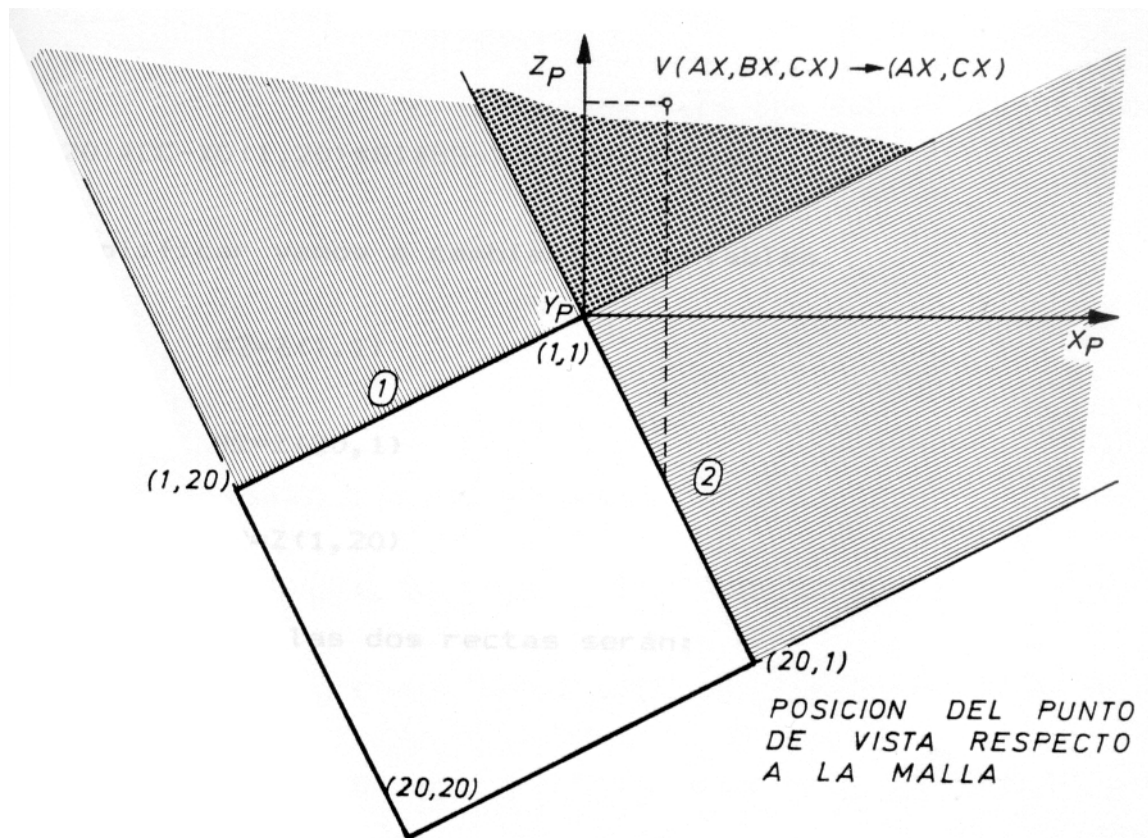


FIGURA 2-6-3-2

$$CV=V3*AX+V5$$

$$CU=U3*AX+U5$$

Si $CV > CX$ el punto de vista está por debajo de la recta $[(A1, A2)(B1, B2)]$.

Si $CU > CX$ el punto de vista está por debajo de la recta $[(A1, A2)(C1, C2)]$.

En función de esta circunstancia tomaremos la decisión de dibujar o no el lateral.

Veamos el algoritmo que corresponde:

```

700 INPUT "COORDENADAS DEL PUNTO DE VISTA";AX,BX,CX
800 REM SEAN AX,BX,CX LAS COORDENADAS DEL PUNTO DE VISTA, A EL
ANGULO DEL PRIMER GIRO Y M/2 EL ANGULO DEL SEGUNDO GIRO DE
NOMBRE B
810 CO=COS(A) : SI=SIN(A) : CC=COS(B) : SS=SIN(B) : A1=0 : A2=0
820 X(20,1)=X0(20,1)*CO-Y0(20,1)*SI : B1=X(20,1)
830 Z(20,1)=Z0(20,1)*CC-X0(20,1)*SI*SS-Y0(20,1)*CO*SS : B2=Z(20,1)
840 X(1,20)=X0(1,20)*CO-Y0(1,20)*SI : C1=X(1,20)
850 Z(1,20)=Z0(1,20)*CC-X0(1,20)*SI*SS-Y0(1,20)*CO*SS : C2=Z(1,20)
860 V1=A1-B1 : V2=A2-B2 : U1=A1-C1 : U2=A2-C2
    
```

```
870 IF V1=0 THEN 1000
880 V3=V2/V1 : V4=V3*A1 : V5=A2-V4 : CV=V3*AX+V5
890 IF U1=0 THEN 1100
900 U3=U2/U1 : U4=U3*A1 : U5=A2-U4 : CU=U3*AX+U5
920 IF (CX>CV)*(CX>CU) THEN 1600
930 IF (CX>CV) THEN SW=1 : GOTO 970
940 IF (CX>CU) THEN SW=2 : GOTO 970
950 PRINT "IMPOSIBLE CAMBIAR COORDENADAS DEL PUNTO DE VISTA" : GOTO
700
960 REM EL PUNTO DE VISTA DEBE ESTAR POR ENCIMA DE LAS RECTAS
965 PARALELAS QUE PASAN POR EL PUNTO (20,20) DE LA MALLA
970 X(20,20)=C1-B1 : D1=X(20,20)
975 Z(20,20)=B2+C2 : D2=Z(20,20)
980 V4=V3*D1 : V5=D2-V4 : DV=V3*AX+V5
985 U4=U3*D1 : U5=D2-U4 : DU=U3*AX+U5
990 IF (CX<DV)*(CX<DU) THEN 950
995 ON SW GOTO 1200,1400
1000 IF (AX<0)*(CX>0) THEN 1400
1020 GOTO 950
1100 IF (AX>0)*(CX>0) THEN 1600
1120 GOTO 950
1200 REM DIBUJO LATERAL IZQUIERDO : HACER ... : GOTO 2000
1400 REM DIBUJO LATERAL DERECHO : HACER ... : GOTO 2000
1600 REM DIBUJO DE LOS DOS LATERALES
.
1900
2000 PROCESO DE DIBUJO DE LA MALLA CON ELIMINACIÓN DE LÍNEAS
OCULTAS
```

El programa CONICAPERS ilustra estos algoritmos junto con los ya vistos. Los números de instrucción sólo se han colocado aquí a modo de ejemplo para poder seguir con claridad las instrucciones múltiples, la separación de las cuales se ha hecho mediante dos puntos como si se tratara de un programa.

Para seguir estos algoritmos hay que tener en cuenta que el signo (*) de las instrucciones condicionales equivale a un (y) copulativo, es decir, "si se cumple la primera condición (entre paréntesis) y también se cumple la segunda, entonces hacer". El "de lo contrario hacer", está en la línea siguiente pues se ha utilizado el léxico informático para mostrar estos algoritmos.

PROGRAMA CONICAPERS

```
10 CLR
20 REM DIBUJO ORDENADO DE LA CUADRICULA TRIDIMENSIONAL
30 DIM X(22,22),Y(22,22),Z(22,22),X2(22,22),Y2(22,22),
    Z2(22,22),Z3(255),Z4(100),Z5(22,22)
37 PRINT"COLOCAR DISCO CON ARCHIVO EN BOCA 2"
40 INPUT"INTRODUCIR NOMBRE DEL ARCHIVO ";B#
41 REM ■■■■PROCESO DE RECOGIDA DE DATOS DEL ARCHIVO B# "
42 ROPEN#1,FD2,B#
43 FOR I1=1TO20:FOR J1=1TO20
44 INPUT#1,X(I1,J1),Y(I1,J1),Z(I1,J1)
45 NEXT:NEXT
46 CLOSE#1
60 REM ■■■■LA POSICION PRIMITIVA DE LOS PUNTOS ES (X,Y,Z)
61 X(0,0)=X(1,1):Y(0,0)=Y(1,1):Z(0,0)=0:X(21,21)=X(20,20):
    Y(21,21)=Y(20,20):Z(21,21)=0
62 X(0,21)=X(1,20):Y(0,21)=Y(1,20):Z(0,21)=0:X(21,0)=X(20,1):
    Y(21,0)=Y(20,1):Z(21,0)=0
63 FORJ=1TO20
64 X(0,J)=X(1,J):X(21,J)=X(20,J)
65 X(J,0)=X(J,1):X(J,21)=X(J,20)
66 Y(0,J)=Y(1,J):Y(21,J)=Y(20,J)
67 Y(J,0)=Y(J,1):Y(J,21)=Y(J,20)
68 Z(0,J)=0:Z(21,J)=0:Z(J,0)=0:Z(J,21)=0
69 NEXT
80 REM ■■■■LA POSICION FINAL DE LOS PUNTOS ES (X2,Y2,Z2)
100 PRINTTAB(15);"ANGULO DEL PRIMER GIRO":INPUT A
105 IF(A>=0)*(A<90)THENFORI=0TO21:FORJ=0TO21:Z5(I,J)=Z(I,J):
    NEXT:NEXT:GOTO155
110 IF(A>=90)*(A<180)THEN A=A-90:FOR I=0TO21:FORJ=0TO21:
    Z2(I,J)=Z(J,21-I):NEXT:NEXT:GOTO150
120 IF(A>=180)*(A<270)THEN A=A-180:FOR I=0TO21:FORJ=0TO21:
    Z2(I,J)=Z(21-I,21-J):NEXT:NEXT:GOTO150
130 IF(A>270)*(A<=360)THEN A=A-270:FOR I=0TO21:FORJ=0TO21:
    Z2(I,J)=Z(21-J,I):NEXT:NEXT:GOTO150
140 PRINT"EL ANGULO DEBE SER MENOR QUE 360":GOTO100
150 FOR I=0TO21:FORJ=0TO21:Z5(I,J)=Z2(I,J):NEXT:NEXT
155 A=π*A/180
200 B=π/2
250 PRINTCHR$(6):PRINT"!!ESTOY CALCULANDO!!"
380 REM ■■■■PROCESO DE CALCULO DE LAS NUEVAS COORDENADAS
390 SS=SIN(B):CC=COS(B):CD=COS(A):SI=SIN(A)
400 FOR I=0TO21:FOR J=0TO21
440 X1=X(I,J)*CD-Y(I,J)*SI
460 Y1=X(I,J)*SI+Y(I,J)*CD
480 Z1=Z5(I,J)
520 Y2(I,J)=INT(Z1*SS+Y1*CC)
560 X2(I,J)=INT(X1)
570 Z2(I,J)=INT(Z1*CC-Y1*SS)
580 NEXTJ:NEXTI:PRINTCHR$(6)
595 GOSUB20000
596 PRINT"VOY A CENTRAR":GOSUB 8000
600 PRINTCHR$(6)
605 REM ELIMINACION DE LINEAS OCULTAS
```

PROGRAMA CONICAPERS (1)

```

605 REM ELIMINACION DE LINEAS OCULTAS
607 PRINT"VOY A ELIMINAR LINEAS OCULTAS"
610 GRAPHI1,C,01
620 S1=X2(0,21):S2=X2(0,0):S3=X2(21,0):L1=S1:L2=S2:L3=S3
625 IFS1>S2THENS1=S2:Y2(0,20)=Y2(1,0)
627 IFS3<S2THENS3=S2:Y2(20,0)=Y2(1,0)
630 LINE S1,200-Y2(0,20),S2,200-Y2(1,0),S3,200-Y2(20,0)
670 GOSUB 10000
680 REM ████████PROCESO DE DIBUJO████████
682 FORI=1TO255:Z3(I)=200:NEXT
684 FORI=1TO100:Z4(I)=200:NEXT
700 FOR J=1 TO 20
710 FOR I=0 TO19:A=0:B=0:C=0:D=0
720 S1=X2(I,J):S2=X2(I+1,J):S3=X2(I+1,J-1)
730 R1=200-Y2(I,J):R2=200-Y2(I+1,J):R3=200-Y2(I+1,J-1)
735 IF (S1>255)+(S2>255)+(S3>255)GOTO1000
740 IF (R1<=Z3(S1))*(R2<=Z3(S2))*(R3<=Z3(S3))THEN LINE S1,R1,
    S2,R2,S3,R3:GOTO7500
750 D=0:A=0:B=0:C=0:IF R1<=Z3(S1)THEN A=1:D=1
760 IF R2<=Z3(S2)THEN B=1:D=D+1
770 IF R3<=Z3(S3)THEN C=1:D=D+1
780 IF D=1 GOTO3000
790 IF D=0 GOTO7590
800 IF A=0 GOTO 2500
810 IF B=0 GOTO 2000
900 REM UN PUNTO OCULTO DE LOS TRES
910 REM TERCER PUNTO OCULTO
911 PRINTCHR#(6):CURSOR 1,1
920 IF (S2=S3)THEN KX=S2:GOTO959
925 Q1=R2:Q2=R3:Q1=S2:Q2=S3
930 P1=(Q2-Q1)/(Q2-Q1):P2=P1*Q1:P3=Q1-P2
935 FOR KX=S2+1TO83
940 KY=INT(P3+P1*KX)
950 IF KX<=255THENIFKY>=Z3(KX)THENKY=Z3(KX):GOTO965
955 IF KX>255THENXX=KX-255:IFKY>=Z4(XX)THENKY=Z4(XX):GOTO965
957 NEXT KX
959 FOR KY=R2TOR3
960 IF KX<=255THENIFKY=Z3(KX)GOTO965
961 IF KX>255THENXX=KX-255:IFKY=Z4(XX)GOTO965
962 NEXT KY
965 S3=KX:R3=KY
970 LINE S1,R1,S2,R2,S3,R3:GOTO7500
1000 W1=0:W2=0:W3=0
1005 IF (S1>255)THEN 1500
1010 IF (S2>255)THEN 1300
1012 REM
1015 W3=S3-255
1020 D=0:IF R1<=Z3(S1)THEN A=1:D=1
1030 IF R2<=Z3(S2)THEN B=1:D=D+1
1040 IF R3<=Z4(W3)THEN C=1:D=D+1
1050 IF D=3 THEN LINE S1,R1,S2,R2,S3,R3:GOTO 7500
1060 GOTO 780
1300 REM ████████
1305 D=0:IF R1<=Z3(S1)THEN A=1:D=1
1310 W2=S2-255:IF R2<=Z4(W2)THEN B=1:D=D+1
1315 W3=S3-255:IF R3<=Z4(W3)THEN C=1:D=D+1

```

PROGRAMA CONICAPERS (2)

```
1315 W3=S3-255: IF R3<=Z4(W3) THEN C=1: D=D+1
1320 GOTO1050
1500 REM SEGUNDO PUNTO OCULTO
1505 D=0: W1=S1-255: IF R1<=Z4(W1) THEN A=1: D=1
1510 W2=S2-255: IF R2<=Z4(W2) THEN B=1: D=D+1
1515 W3=S3-255: IF R3<=Z4(W3) THEN C=1: D=D+1
1520 GOTO1050
2000 REM SEGUNDO PUNTO OCULTO
2001 PRINTCHR$(6): CURSOR1,1
2004 IF (S1=S2) THEN KX=S1: GOTO2041
2005 O1=R1: O2=R2: Q1=S1: Q2=S2
2007 P1=(O2-O1)/(Q2-Q1): P2=P1*Q1: P3=O1-P2
2010 FOR KX=S1+1 TO S2
2020 KY=INT(P3+P1*KX)
2030 IF KX<=255 THEN IF KY>=Z3(KX) THEN KY=Z3(KX): GOTO2048
2032 IF KX>255 THEN XX=KX-255: IF KY>=Z4(XX) THEN KY=Z4(XX): GOTO2048
2040 NEXTKX: GOTO2048
2041 FOR KY=R1 TO R2
2042 IF KX<=255 THEN IF KY=Z3(KX) GOTO2048
2043 IF KX>255 THEN XX=KX-255: IF KY=Z4(XX) GOTO2048
2044 NEXT KY
2048 S2=KX: R2=KY: WW=1: LINE S1, R1, S2, R2: GOTO 7500
2050 IF (S2=S3) THEN KX=S2: GOTO2091
2055 O1=R2: O2=R3: Q1=S2: Q2=S3: E1=-1: E2=-1: IF S3<S2 THEN E1=1: E2=1
2057 P1=(O2-O1)/(Q2-Q1): P2=P1*Q1: P3=O1-P2
2060 FOR KX=S3+E1 TO S2 STEP E2
2070 KY=INT(P3+P1*KX)
2080 IF KX<=255 THEN IF KY>=Z3(KX) THEN KY=Z3(KX): GOTO2098
2085 IF KX>255 THEN XX=KX-255: IF KY>=Z4(XX) THEN KY=Z4(XX): GOTO2098
2090 NEXT KX
2091 FOR KY=R3 TO R2
2092 IF KX<=255 THEN IF KY=Z3(KX) GOTO2098
2093 IF KX>255 THEN XX=KX-255: IF KY=Z4(XX) GOTO2098
2094 NEXT KY
2098 S2=KX: R2=KY: WW=0
2100 LINE S2, R2, S3, R3: IF E1=-1 GOTO7550
2110 GOTO7590
2500 REM PRIMER PUNTO OCULTO
2501 PRINTCHR$(6): CURSOR 1,1
2510 IF (S1=S2) THEN KX=S1: GOTO2551
2515 O1=R1: O2=R2: Q1=S1: Q2=S2
2517 P1=(O2-O1)/(Q2-Q1): P2=P1*Q1: P3=O1-P2
2525 FOR KX=S2-1 TO S1 STEP -1
2530 KY=INT(P3+P1*KX)
2540 IF KX<=255 THEN IF KY>=Z3(KX) THEN KY=Z3(KX): GOTO 2555
2545 IF KX>255 THEN XX=KX-255: IF KY>=Z4(XX) THEN KY=Z4(XX): GOTO2555
2550 NEXTKX
2551 FOR KY=R2 TO R1
2552 IF KX<=255 THEN IF KY=Z3(KX) GOTO2555
2553 IF KX>255 THEN XX=KX-255: KY=Z4(XX): GOTO2555
2554 NEXT KY
2555 R1=KY: S1=KX: IF (I=0)*(S3<S2) THEN S1=S2: R1=R2
2560 LINE S1, R1, S2, R2, S3, R3: GOTO7500
3000 REM SEGUNDO PUNTO OCULTO HAY DOS PUNTOS OCULTOS
3001 PRINTCHR$(6): CURSOR 1,1
3010 IF C=1 GOTO 3800
```

PROGRAMA CONICAPERS (3)

```

3010 IF C=1 GOTO 3800
3020 IF B=1 GOTO 3400
3025 PRINTCHR$(6)
3030 REM ■■■■ES EL 1er PUNTO EL QUE SE VE
3031 PRINTCHR$(6):CURSOR 1,1
3040 IF (S1=S2) THEN KX=S1:GOTO3081
3045 O1=R1:O2=R2:Q1=S1:Q2=S2
3047 P1=(O2-O1)/(Q2-Q1):P2=P1*Q1:P3=O1-P2
3050 FOR KX=S1+1 TO S2
3060 KY=INT(P3+P1*KX)
3070 IF KX<=255 THEN IF KY>=Z3(KX) THEN KY=Z3(KX):GOTO3088
3075 IF KX>255 THEN XX=KX-255: IF KY>=Z4(XX) THEN KY=Z4(XX):GOTO 3088
3080 NEXT KX:GOTO3088
3081 FOR KY=R1 TO R2
3082 IF KX<=255 THEN IF KY=Z3(KX) GOTO3088
3083 IF KX>255 THEN XX=KX-255: IF KY=Z4(XX) GOTO3088
3084 NEXT KY
3088 S2=KX:R2=KY:S3=KX:R3=KY
3090 LINE S1,R1,S2,R2:GOTO7500
3400 REM ■■■■EL 2o PUNTO ES EL QUE SE VE
3401 PRINTCHR$(6):CURSOR 1,1
3410 IF (S1=S2) THEN KX=S1:GOTO3451
3415 O1=R1:O2=R2:Q1=S1:Q2=S2:E1=-1:E2=-1
3417 P1=(O2-O1)/(Q2-Q1):P2=P1*Q1:P3=O1-P2: IF S2<S1 THEN E1=1:E2=1
3420 FOR KX=S2+E1 TO S1 STEP E2
3430 KY=INT(P3+P1*KX)
3440 IF KX<=255 THEN IF KY>=Z3(KX) THEN KY=Z3(KX):GOTO3458
3445 IF KX>255 THEN XX=KX-255: IF KY>=Z4(XX) THEN KY=Z4(XX):GOTO3458
3450 NEXT KX:STOP
3451 FOR KY=R2 TO R1
3452 IF KX<=255 THEN IF KY=Z3(KX) GOTO3458
3453 IF KX>255 THEN XX=KX-255: IF KY=Z4(XX) GOTO3458
3454 NEXT KY:STOP: IF (KX=S1)*(KY=R1) THEN 7590
3458 S1=KX:R1=KY
3460 IF (S2=S3) THEN KX=S2:GOTO3499
3465 O1=R2:O2=R3:Q1=S2:Q2=S3:E1=1:E2=1
3467 P1=(O2-O1)/(Q2-Q1):P2=P1*Q1:P3=O1-P2: IF S3<S2 THEN E1=-1:E2=-1
3470 FOR KX=S2+E1 TO S3 STEP E2
3480 KY=INT(P3+P1*KX)
3490 IF KX<=255 THEN IF KY>=Z3(KX) THEN KY=Z3(KX):GOTO 3503
3493 IF KX>255 THEN XX=KX-255: IF KY>=Z4(XX) THEN KY=Z4(XX):GOTO3503
3495 NEXT KX:GOTO3503
3499 FOR KY=R2 TO R3
3500 IF KX<=255 THEN IF KY=Z3(KX) GOTO3503
3501 IF KX>255 THEN XX=KX-255: IF KY=Z4(XX) GOTO3503
3502 NEXT KY
3503 S3=KX:R3=KY
3510 LINE S1,R1,S2,R2,S3,R3:GOTO7500
3800 REM ■■■■EL 3er PUNTO ES EL QUE SE VE
3801 PRINTCHR$(6):CURSOR 1,1
3810 IF (S3=S2) THEN KX=S2:GOTO3851
3815 O1=R2:O2=R3:Q1=S2:Q2=S3
3817 P1=(O2-O1)/(Q2-Q1):P2=P1*Q1:P3=O1-P2
3820 FOR KX=S3-1 TO S2 STEP -1
3830 KY=INT(P3+P1*KX)

```

PROGRAMA CONICAPERS (4)

```
3830 KY=INT(P3+P1*KX)
3840 IF KX<=255THENIFKY>=Z3(KX)THEN KY=Z3(KX):GOTO 3858
3845 IF KX>255THENXX=KX-255:IFKY>=Z4(XX)THEN KY=Z4(XX):GOTO3858
3850 NEXT KX:GOTO7590
3851 FOR KY=R3TOR2
3852 IF KX<=255THENIFKY=Z3(KX)GOTO3858
3853 IF KX>255THENXX=KX-255:IFKY=Z4(XX)GOTO3858
3854 NEXT KY
3858 S2=KX:R2=KY:S1=KX:R1=KY
3859 REM LINE S2,R2,S3,R3
3860 GOTO7500
7500 REM ALTURA EN PANTALLA
7510 IF (S1=S2)THEN KX=S1:GOTO7542
7515 O1=R1:O2=R2:Q1=S1:Q2=S2
7517 P1=(O2-O1)/(Q2-Q1):P2=P1*Q1:P3=O1-P2
7520 FOR D4=S1TO82
7525 IF D4>255THEND5=D4-255:Z4(D5)=INT(P3+P1*D4):GOTO7540
7530 Z3(D4)=INT(P3+P1*D4)
7540 NEXTD4:GOTO7548
7542 IF R1<R2 THEN AL=R1:GOTO7544
7543 AL=R2
7544 IFB1>255 THEN D5=S1-255:Z4(D5)=AL:GOTO7548
7546 Z3(S1)=AL
7548 IF WW=1GOTO2050
7549 IFH9=1THENH9=0:GOTO3460
7550 IF (S2=S3)THEN KX=S2:GOTO7582
7555 O1=R2:O2=R3:Q1=S2:Q2=S3
7557 P1=(O2-O1)/(Q2-Q1):P2=P1*Q1:P3=O1-P2
7560 FOR D8=92TO83:E3=INT(P3+P1*D8):E4=0
7565 IF D8>255THEND6=D8-255:E4=1:IFE3<Z4(D6)THEN Z4(D6)=E3:GOTO7580
7567 IFE4=1GOTO7580
7570 IFE3<Z3(D8)THENZ3(D8)=E3
7580 NEXT D8:GOTO 7590
7582 IF R2<R3 THEN AL=R2:GOTO7584
7583 AL=R3
7584 IF S2>255 THEN D6=S2-255:Z4(D6)=AL:GOTO7590
7586 Z3(S2)=AL
7590 NEXT I:NEXT J
7600 END
8000 REM■■■■■PROCESO DE CENTRADO
8005 XA=0:XI=10000:YA=0:YI=10000:ZA=0:ZI=10000
8010 FOR I=0 TO21:FOR J=0TO21
8020 IF X2(I,J)>XA THEN XA=X2(I,J)
8030 IF X2(I,J)<XI THEN XI=X2(I,J)
8040 IF Y2(I,J)>YA THEN YA=Y2(I,J)
8050 IF Y2(I,J)<YI THEN YI=Y2(I,J)
8060 IF Z2(I,J)>ZA THEN ZA=Z2(I,J)
8070 IF Z2(I,J)<ZI THEN ZI=Z2(I,J)
8075 NEXT:NEXT:PRINT"PROCESO DE CENTRADO"
8080 REM FX,FY,FZ SON LOS FACTORES DE CONVERSION DE COORDENADAS
      PARA EL CENTRADO
8090 FX=INT((XA+XI)/2-150):FY=INT((YA+YI)/2-100):FZ=INT((ZA+ZI)/2)
8100 FOR I=0TO 21:FOR J=0TO21
8110 X2(I,J)=X2(I,J)-FX:Y2(I,J)=Y2(I,J)-FY:Z2(I,J)=Z2(I,J)-FZ
```

PROGRAMA CONICAPERS (5)

```
B110 X2(I,J)=X2(I,J)-FX:Y2(I,J)=Y2(I,J)-FY:Z2(I,J)=Z2(I,J)-FZ
B120 NEXT: NEXT: RETURN
10000 FORAB=1 TO 255
10010 Z3(AB)=0: NEXT
10020 FORCD=1 TO 100
10030 Z4(CD)=0: NEXT
10040 RETURN
20000 REM OBTENCION DE LAS COORDENADAS CONICAS DE LOS PUNTOS
20001 PRINT "INTRODUCIR COORD. PUNTO DE VISTA": INPUT "-50<AX>350 " : AX
20002 INPUT "50<BX>400 " : BX
20003 INPUT "300<CX>2000 " : CX
20005 IF BX<100 THEN PRINT "BX DEBE SER MAYOR QUE 100": GOTO 20005
20007 IF CX<200 THEN PRINT "CX DEBE SER MAYOR QUE 200": GOTO 20005
20010 INPUT "ENTRAR ALTURA Z DEL PLANO DEL CUADRO": ZX
20020 FOR I=0 TO 21: FOR J=0 TO 21
20030 IF CX=Z2(I,J) THEN PRINT "PTO EN PLANO DESVANECIMIENTO": STOP
20040 KX=(ZX-CX)/(CX-Z2(I,J))
20050 X2(I,J)=INT((AX-X2(I,J))*KX+AX)
20060 Y2(I,J)=INT((BX-Y2(I,J))*KX+BX)
20070 NEXT J: NEXT I: TT=1: RETURN
30000 REM COMPROBACION: CONSOLEC80: FOR J=20 TO 1 STEP -1: FOR I=1 TO 20
30005 B$=STR$(ABS(INT(Z(I,J)))): B$=STRING$( "0", 2-LEN(B$))+B$+" "
30010 IF Z(I,J)<0 THEN B$="-"+B$: PRINT CHR$(%04);
30020 PRINT B$; : NEXT: PRINT: NEXT: STOP: CONSOLEC40: END
```

PROGRAMA CONICAPERS (6)

2.7 - GENERACIÓN DE UN SISTEMA CONJUNTO DE OBTENCIÓN DE PROYECCIONES CÓNICAS Y AXONOMÉTRICAS

2.7.1 - COMPARACIÓN DE ALGORITMOS UTILIZADOS EN AMBOS MÉTODOS Y COORDINACIÓN ENTRE ELLOS

El procedimiento empleado para visualizar la perspectiva axonométrica y la cónica difieren realmente en pocos puntos. En ambos se han utilizado los dos giros consecutivos antes de proyectar (aunque la forma de proyectar sea distinta). En ambos se ha utilizado el mismo sistema de centrado aproximado. En los dos hemos dibujado solamente la parte superior de la superficie junto con los laterales visibles. Y, por último, también en ambos hemos señalado un mismo procedimiento para la supresión de líneas ocultas.

Todo ello hace posible que podamos formar un conjunto que posibilite la obtención de cualquier tipo de perspectiva de la malla sin tener que duplicar instrucciones.

Veamos pues los algoritmos anteriores integrados en uno solo que va a permitirnos realizar todas las opciones.

Requerimientos previos.

Supongamos que las coordenadas de los vértices de la malla son: $(X0(I,J), Y0(I,J), Z0(I,J))$ para valores de I y J que van desde 1 hasta 20.

Llamaremos A al primer ángulo de giro según la figura 2-5-2-2 (sentido antihorario).

Llamaremos B al segundo ángulo de giro según la figura 2-5-2-3 (sentido antihorario).

Ambos giros vamos a suponerlos en grados sexagesimales.

Llamaremos (CE,DE) al vector posición de centrado (coordenadas normales, es decir, segunda coordenada dirección vertical hacia arriba).

Las coordenadas del centro de la pantalla las llamaremos PX, PY con dirección de la segunda coordenada vertical y hacia abajo.

Llamaremos (AX, BX, CX) a las coordenadas del punto de vista (coordenadas de pantalla antes de centrar) (figura 2-6-1-5).

Llamaremos ZX a la altura del cuadro con respecto a la pantalla o bien con respecto al punto vértice de la malla de subíndices (1,1) (figura 2-6-1-5).

Las variables A, B, C, X1, Y1, Y2, Z2, V1, V2, V3, V4,V5, U1, U2, U3, U4, U5 son auxiliares y van cambiando de valor para cada punto.

En esas circunstancias tendremos:

```
1000 REM LASA COORDENADAS X0(I,J), Y0(I,J), Z0(I,J) SON CONOCIDAS
1010 INPUT "DESEA LA PERSPECTIVA CONICA?"; A$
1020 INPUT "ANGULO DEL PRIMER GIRO (GRADOS SEXAGESIMALES)?"; A
1030 REM CAMBIO DE VARIABLES PARA SITUAR CON SUBINDICES (1,1) EL
VERTICE MAS CERCANO AL OBSERVADOR [ X1(I, J), Y1(I, J), Z1(I, J) ]
1040 IF (A>=0) * (A<90) THEN FOR I=1 TO 20 : FOR J=1 TO 20 : X1(I,J)=X0(I,J) :
Y1(I,J)=Y0(I,J) : Z1(I,J)=Z0(I,J) : NEXT J : NEXT I : REM A NO VARIA : GOTO 1100
1050 IF (A>=90)*(A<180) THEN FOR I=1 TO 20 : FOR J=1 TO 20 : X1(I,J)=X0(I,J) :
Y1(I,J)=Y0(I,J) : Z1(I,J) = Z0(U,20-I) : NEXT J : NEXT I : A=A-90 : GOTO 1100
1060 IF (A>=180)*(A<270) THEN FOR I=1 TO 20 : FOR J01 TO 20 : X1(I,J)=X0(I,J) :
Y1(I,J)=Y0(I,J) : Z1(I,J)=Z0(20-I,20-I) : NEXT J : NEXT I : A=A-180 : GOTO 1100
1070 IF (A>=270)*(A<360) THEN FOR I=1 TO 20 : FOR J=1 TO 20 : X1(I,J)=X0(I,J) :
Y1(I,J)=Y0(I,J) : Z1(I,J)=Z0(20-J,I) : NEXT J : NEXT I : A=A-270 : GOTO 1100
1080 PRINT "EL ANGULO DEBE SER MENOR DE 360°" : GOTO 1020
1100 A=A*π/180
1110 IF A$="SI" THEN B=π/2 : GOTO 1150
1120 INPUT "ANGULO DEL SEGUNDO GIRO?" (GRADOS SEXAGESIMALES"; B
1130 IF (B>90) THEN "DEBE SER MENOR DE 90°" : GOTO 1120
1140 B=B*M/180
1150 CO=COS(A) : SI=SIN(A) : CC=COS(B) : SS=SIN(B)
1160 IF A$="SI" THEN INPUT "COORD. DEL PUNTO DE VISTA";AX,BX,CX : GOTO
1180
1170 GOTO 1800
1180 X2(20,1)=X0(20,1) * CO-Y0(20,1) * SI : B1=X2(20,1)
1190 Z2(20,1)=Z0(20,1) * CC-X0(20,1) * SI * SS-Y0(20,1) * CO* SS : B2=Z2(20,1)
1200 X2(1,20)=X0(1,20) * CO-Y0(1,20) * SI : C1=X2(1,20)
1210 Z2(1,20)=Z0(1,20) * CC-X0(1,20) * SI * SS-Y0(1,20) *CO * SS :
C2=Z2(1,20)
1220 X2(20,20)=C1-B1 : D1=X2(20,20) : A1=0 : A2=0
1230 Z2(20,20)=B2+C2 : D2=Z2(20,20)
1240 V1=A1-B1 : V2=A2-B2 : U1=A1-C1 : U2=A2-C2
1250 IF V1=0 THEN 1370
1260 V3=V2/V1 : V4=V3 * A1 : V5=A2-V4 : CV=V3 * AX+V5
1270 IF U1=0 THEN 1400
1280 U3=U2/U1 : U4=U3 *A1 : U5=A2-U4 : CU=U3 *AX+U5
1290 IF (CX>CV) *(CX>CU) THEN LT00 : REM DOS LATERALES : GOTO 2000
1300 IF (CX>CV) THEN SW=1 : GOTO 1330
1310 IF (CX>CU) THEN SW=2 : GOTO 1330
1320 PRINT "FUERA DEL CONO DE VISION OPTIMO" : GOTO 1160
1330 V4=V3 *D1 : V5=D2-V4 : DV=V3 *AX+V5
1340 U4=U3 *D1 : U5=D2-U4 : DU=U3 *AX+U5
1350 IF (CX<DV) *(CX<DU) THEN 1320
1360 ON SW GOTO
1370 IF (AX<0) *(CX<0) THEN LT=0 : GOTO 2000
```

```
1380 IF (AX<200) *(CX>0) THEN LT=1 : REM LATERAL DERECHO : GOTO 2000
1390 GOTO 1320
1400 IF (AX>0) *(CX>0) THEN LT=0 : GOTO 2000
1410 IF (AX>200) *(CX>0) THEN LT=2 : REM LATERAL IZQUIERDO : GOTO 2000
1420 GOTO 1320
1800 REM VECTOR POSICION DE CENTRADO EN AXONOMETRÍA
1810 XC=X1(10,10)*CO-Y1(10,10)*SI : CE=PE-XC
1820 YC=(X1(10,10) *SI+Y1(10,10)*CO)*CC+Z1(10,10)*SS : DE=PY-YC
1830 GOTO 2020
2000 REM CALCULO DE LAS COORDENADAS TRAS EL SEGUNDO GIRO
(CENTRADAS PARA AXONOMETRIA Y SIN CENTRAR PARA CONICA)
2010 CE=0 : DE=0
2020 FOR I=1 TO 20 : FOR J=1 TO 20
2030 X2(I,J)=X1(I,J)*CO-Y1(I,J)*SI+CE
2040 Y2(I,J)=(X1(I,J)*SI+Y1(I,J)*CO)*CC+Z1(I,J)*SS+DE
2050 Z2(I,J)=Z1(I,J)*CC-(X1(I,J)*SI+Y1(I,J)*CO)SS
2060 NEXT J : NEXT I
2070 IF (A$<>"SI") THEN 4000
2080 INPUT "INTRUDICIR ALTURA DEL PLANO DEL CUADRO";ZX
2090 REM VECTOR POSICION DE CENTRADO DE CONICA
2100 XC=(X2(10,10)-AX)*(ZX-CX)/(Z2(10,10)-CX)+AX : CE=PX-XC
2110 YC=(Y2(10,10)-BX)*(ZX-CX)/(Z2(10,10)-CX)+BX : DE=PY-YC
2120 REM CALCULO DE LAS COORDENADAS CONICAS
3000 FOR I01 TO 20 : FOR J=1 TO 20
3010 A=X2(I,J) : B=Y2(I,J) : C=Z2(I,J)
3020 V1=A-AX : V2=B-BX : V3=C-CX
3030 X2(I,J):(V1/V3)*(ZX-CX)+AX+CE
3040 Y2(I,J)=(V2/V3)*(ZX-CX)+BX+DE
3050 NEXT J : NEXT I
4000 REM PROCESO DE DIBUJO CON ELIMINACION DE LINEAS OCULTAS COMUN
A LOS DOS TIPOS DE PERSPECTIVAS
```

La parte que continua es común a ambos métodos por lo que se puede prescindir en este apartado de su tratamiento.

El programa PROYECCIÓN MALLA responde a lo exigido en este apartado.

La nomenclatura correspondiente a este programa es la siguiente:

$X(22,22)$: Coordenada horizontal primitiva en el sentido de las X, que corresponde a cada uno de los vértices de la malla y también a los laterales.

$Y(22,22)$: Coordenada horizontal primitiva en el sentido de las Y, que corresponde a cada uno de los vértices de la malla y también a los laterales.

$Z(22,22)$: Tercera coordenada correspondiente a cada uno de los vértices de la malla y que representa la altura que tienen con respecto al plano de referencia antes de efectuar movimientos.

X2(22,22), Y2(22,22), Z2(22,22) : Coordenadas de los puntos de la malla una vez ejecutados los movimientos y la proyección. También se utiliza para ubicar los resultados intermedios, por esa razón se necesita también la tercera coordenada.

Z3(255) : Variable utilizada para contener los valores de la frontera en la eliminación de líneas ocultas.

Z4(100) : Complementa a la variable Z3(255) y tiene el mismo objetivo.

El resto de variables utilizadas responde a los mismos valores que les hemos dado en este apartado.

En la SÍNTESIS podemos ver el programa completo y algunos de los resultados obtenidos con él junto a los datos utilizados.

2.7.2 - CENTRADO DE LAS PERSPECTIVAS OBTENIDAS

Si nos interesa un centrado exacto y no aproximado como lo tenemos hasta ahora, hay que encontrar el valor exacto del vector posición de centrado. Para ello lo más efectivo es tomar las posiciones más alta y más baja así como la que resulta situada más hacia la derecha y más hacia la izquierda.

He aquí una subrutina que produce el centrado partiendo de las coordenadas de los vértices una vez aplicados los dos giros en axonometría o el proceso equivalente en cónica.

```
8000 REM PROCESO DE CENTRADO DE COORDENADAS
8010 XA=0 : XI=10.000 : YA=0 : YI=10.000
8020 FOR I=1 TO 20 : FOR J=1 TO 20
8030 IF X2(I,J)>XA THEN XA=X2(I,J)
8040 IF X2(I,J)<XI THEN XI=X2(I,J)
8050 IF Y2(I,J)>YA THEN YA=Y2(I,J)
8060 IF Y2(I,J)<YI THEN YI=Y2(I,J)
8070 NEXT J : NEXT I
8080 REM FX SERA LA PRIMERA COORD. DEL VECTOR CENTRADO
8090 REM FY SERA LA SEGUNDA COORD. DEL VECTOR CENTRADO
8100 FX=PX-(XA+XI)/2
8110 FY=PY-(YA+YI)/2
8120 FOR I=1 TO 20 . FOR J=1 TO 20
8130 X2(I,J)=X2(I,J)+FX : Y2(I,J)=Y2(I,J)+FY
8140 NEXT J : NEXT I
```

Este proceso aunque más exacto es más lento que los anteriores pues requiere procesar al menos una vez más todos los puntos de la malla. El programa PROYECCIÓN MALLA CENTRADA incorpora este procedimiento de centrado.

2.7.3 - LATERALES DE LA MALLA

Para dibujar los laterales de la malla se han incorporado al total de puntos vértice de cuadrícula, otra serie de puntos cuyas dos primeras coordenadas son las mismas que tienen los puntos del borde de la cuadrícula, mientras que su altura se ha hecho igual a una constante.

El tratamiento de estos puntos puede ser el mismo que se da al resto de puntos de la malla.

El algoritmo que utilizaríamos para conseguir sus coordenadas sería el siguiente:

```
FOR i=1 TO 20 : X0(i,21)=X0(i,20) : X0(21,i)=X0(20,i):  
X0(0,i)=X0(1,i) : X0(i,0)=X0(i,1) : Y0(i,21)=Y0(i,20)  
Y0(21,i)=Y0(20,i) : Y0(0,i)=Y0(1,i) : Y0(i,0)=Y0(i,1)  
NEXT I : X0(0,0)=X0(1,1) : X0(0,21)=X0(1,20)  
X0(21,0)=X0(20,1) : X0(0,21)=Y0(1,20)  
Y0(21,0)=Y0(20,1) : Y0(21,21)=Y0(20,20)  
FOR I=0 TO 21 : Z(I,21)=K : Z(21,1)=K : Z(I,0)=K : Z(0,I)=K : NEXT I
```

De esta manera convertimos la malla de 20 por 20 puntos en una malla de 22 por 22 puntos.

Con este tipo de mallas se han realizado los programas de visualización en perspectiva.

En el caso que no quisiéramos ver el lateral, bastaría con no procesar en los bucles más que de I=1 hasta I=20 y lo mismo para el otro índice.

2.8 - OBTENCIÓN DE PERFILES EN LAS SUPERFICIES GENERADAS MEDIANTE NUESTRO MODELO

2.8.1 - PERFILES LONGITUDINALES Y TRANSVERSALES

Una de las utilidades con más aplicaciones en las superficies definidas mediante mallas de alambre, es la obtención de perfiles producidos por planos.

Los más fáciles de obtener son los que se producen al cortar dicha superficie por planos verticales paralelos a los bordes de la malla y mejor aún, que pasen por vértices de dicha malla.

Para resolver esos casos es suficiente un algoritmo muy simple que recorre la malla de un lado a otro. De cada punto (vértices de la malla) guarda su altura y la distancia horizontal que le separa del anterior, para luego poder dibujar un perfil.

Los algoritmos que dan perfiles longitudinales y transversales son los que siguen:

```
INPUT "ENTRAR LA COORDENADA X POR LA QUE INTERESA HALLAR
EL PERFIL TRANSVERSAL";X
I = INT(X/10) : D(1) = 1 : Z(1) = Z(I,1) : DT = D(1)
FOR J=2 TO 20 : D(J)=DT+Y(I,J+1)-Y(I,J) : Z(J)=Z(I,J) : NEXT J
FOR J=1 TO 19
LINE D(J), Z(J), D(J+1), Z(J+1)
NEXT J
```

Y para el longitudinal:

```
INPUT "ENTRAR LA COORDENADA Y POR LA QUE INTERESA HALLAR
EL PERFIL LONGITUDINAL";Y
J=INT(Y/10) : D(1)=1 : Z(1)=Z(1,J) : DT=D(1)
FOR I=2 TO 20 : D(I)=DT+X(I+1,J)-X(I,J) : Z(I)=Z(I,J) : NEXT I
FOR I=1 TO 10
LINE D(I), Z(I), D(I+1), Z(I+1)
NEXT I
```

2.8.2 - PERFILES OBLICUOS

Los perfiles longitudinales y transversales son útiles sobre todo para hacer comprobaciones por su gran rapidez, pero sin embargo en la práctica se hacen necesarios otros tipos de perfiles conseguidos también a través de planos verticales que no son paralelos a los laterales.

Se hace necesario el ir de un punto a otro de la malla de forma oblicua y no necesariamente a través de vértices.

En este caso hay que encontrar la intersección del plano vertical con todas las aristas que atraviesa, hallar sus coordenadas y posteriormente proceder como en el caso anterior.

La secuencia a seguir será la siguiente:

- Entrar las coordenadas de los puntos que determinarán la dirección del plano de corte (solamente dos coordenadas puesto que la altura se obtendrá automáticamente sobre la superficie).
- Determinación del vector director de la recta que representa el plano vertical proyectado sobre el plano horizontal.
- Determinación de la ecuación de la misma recta que luego utilizaremos para ir encontrando puntos del perfil.
- Determinación (figura 2-8-2-1) de los puntos de intersección del plano vertical con las cuadrículas de la malla.
- Hallar las distancias en proyección horizontal que existen entre los puntos de intersección obtenidos.
- Crear una variable que contenga las distancias acumuladas desde el primer hasta el último punto y otra de igual subíndice que contenga las alturas de cada punto en el mismo orden.
- Dibujar el perfil utilizando las coordenadas del apartado anterior.

Resolución analítica que corresponde:

Sean (X_A, Y_A) y (X_B, Y_B) las coordenadas de los puntos que sitúan el plano.

El vector director de la recta AB será:

$$V_1 = X_B - X_A \quad V_2 = Y_B - Y_A$$

La ecuación de la recta será en función de las dos variables:

$$(X - X_A)/V_1 = (Y - Y_A)/V_2 \quad Y = (V_2/V_1) * (X - X_A) + Y_A$$

$$X = (V_1/V_2) * (Y - Y_A) + X_A$$

Nos iremos desplazando desde el primer punto (X_A, Y_A) hacia el punto (X_B, Y_B) determinando todas las intersecciones con la malla.

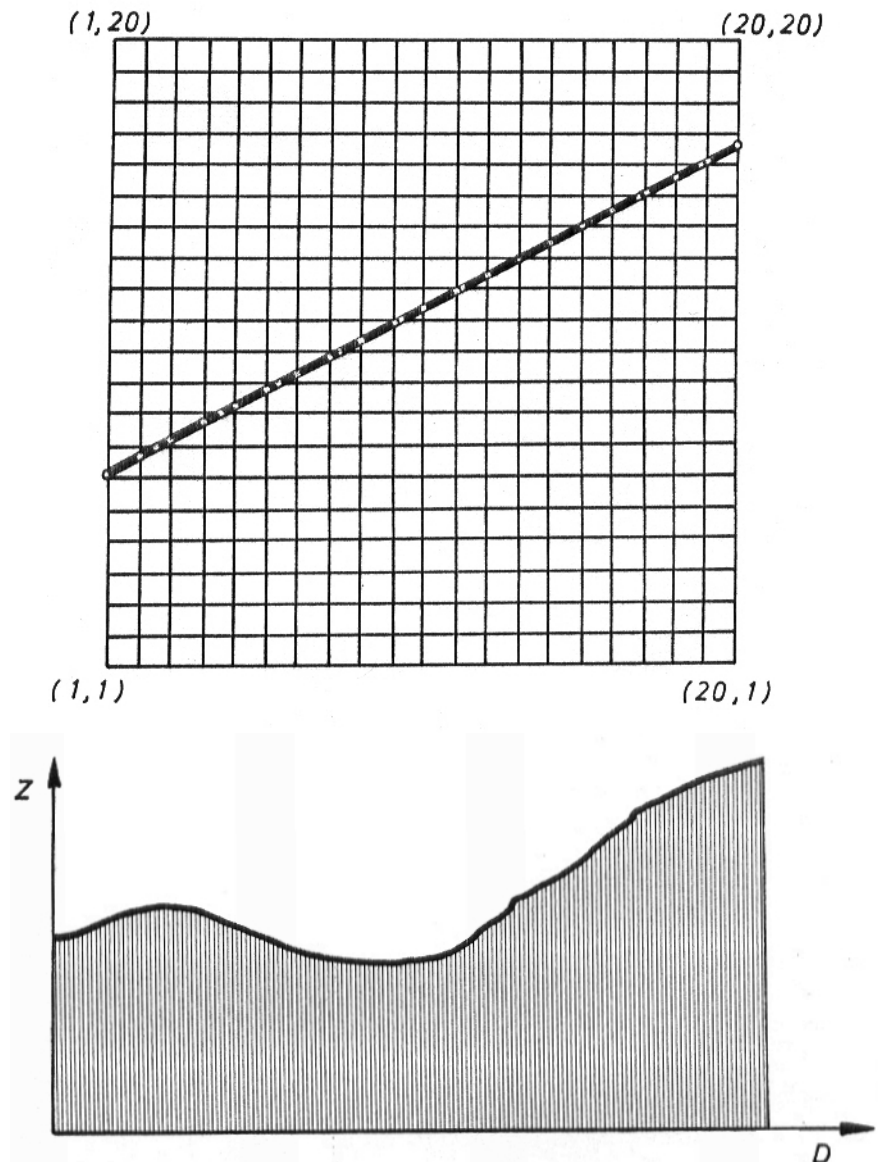


FIGURA 2-8-2-1

El programa PERFIL SOBRE PLANO en sus instrucciones 200 a 3100 resuelve este problema a base de rastrear según las dos variables (de aquí que interesen las dos ecuaciones de la recta).

Una vez tengamos las coordenadas de esos puntos de intersección, el trazado es muy similar al visto en el apartado anterior, sólo que las distancias hay que hallarlas por Pitágoras.

Esta subrutina de trazado podemos verla en el mismo programa de utilidad en las instrucciones 4000 en adelante.

En la SÍNTESIS se muestran los diferentes resultados obtenidos en la aplicación de este programa a ejemplos concretos además del programa que resuelve ya sintetizados todos los casos de perfiles y cuyo nombre es PERFIL POLIGONAL.

2.8.3 - PERFILES SEGÚN ITINERARIO

Si el desplazamiento a través de la superficie de la malla se efectúa siguiendo un camino que no es perteneciente a un solo plano vertical, sino que responde a una poligonal en planta, el problema se complica algo más pues hay que determinar un perfil oblicuo para cada uno de los segmentos de la poligonal y posteriormente enlazarlos entre sí de forma que el perfil sea continuo.

En el programa PERFIL POLIGONAL se efectúan las siguientes funciones:

- Elección de la malla guardada en disco con el número de puntos (20*20) mediante la variable A\$ (nombre del archivo) instrucciones 50 a 90.
- Determinación del camino a recorrer mediante la introducción de las coordenadas horizontales de la poligonal (instrucciones 100 a 120).
- Determinación de los puntos de corte de la poligonal con las rectas de la malla.
- Determinación el parámetro distancia al primer punto a través de la poligonal y la altura para cada uno de los puntos de intersección obtenidos.
- Creación de un archivo ordenado que contiene los puntos de paso por la malla en función de su distancia al primer punto sobre la poligonal en planta y alturas de los mismos cuyo nombre guardamos con la variable B\$ (instrucciones 4650 a 4800).
- Dibujo del perfil poligonal en pantalla así como de los ejes horizontal y vertical para poder visualizarlos de forma ordenada.

Veamos la nomenclatura específica que se ha empleado en la elaboración de este programa:

X(21,21) : Coordenadas horizontales en la dirección X de los puntos de la malla incluidos los laterales.

Y(21,21) : Coordenadas horizontales en la dirección Y de los puntos de la malla incluidos los laterales.

Z(21,21) : Coordenada vertical que señala las alturas de los puntos de la malla incluidos los laterales.

A(100,3) : Variable indexada que contiene las coordenadas de los puntos de intersección entre la poligonal itinerario y la malla (A(100,1) coordenada X, A(100,2) coordenada Y, A(100,3) coordenada Z).

B(50,2) : Variable indexada que contiene las coordenadas horizontales de los puntos de la poligonal itinerario (G(50,1) coordenada horizontal X, G(50,2) coordenada horizontal Y).

H(150,2) : Coordenadas paramétricas de los puntos de intersección entre la poligonal itinerario y la malla puestas de forma ordenada. (H(150,1) señala la distancia en proyección y siguiendo el itinerario entre cada punto y el primero, H(150,2) señala la altura del punto).

XA, YA : Coordenadas provisionales del punto del segmento que se esté tratando en su extremo más cercano al comienzo de la poligonal.

XB, YB : Coordenadas del punto más lejano del segmento que se esté considerando.

V1, V2 : Vector director del segmento que va desde el punto (XA,YA) hasta el punto (XB,YB).

C : Tangente de la recta que pasa por (XA,YA) y (XB,YB).

B* : Variable en la que se ubicará el nombre del archivo que va a contener a los puntos H(150,2).

A* : Variable que ubicará el nombre del archivo de puntos de la malla para su lectura en disco.

El resto de variables son intermedias y contienen valores de tránsito.

En la SÍNTESIS podemos ver el programa completo y algunos resultados obtenidos mediante el mismo (perfiles poligonales).

Si aplicamos la técnica de trazado de curvas mediante SPLINE que vimos en el apartado 2.3 podemos hacer que el perfil resulte una curva continua con las características ya expuestas (ver en SÍNTESIS los resultados obtenidos al trazar los perfiles mediante SPLINE).

2.9 - MALLAS FICTICIAS

2.9.1 - DESCRIPCIÓN Y CARACTERÍSTICAS

Hemos dado el nombre de mallas ficticias a las mallas de alambre cuyas coordenadas horizontales no están dispuestas en forma de rejilla.

Vamos a utilizar este tipo de mallas sobre todo en visualización, aunque también pueden usarse en otras aplicaciones.

En cierto modo ya utilizamos una parte de estas técnicas para visualizar las mallas topográficas puesto que en ellas los laterales están compuestos por puntos cuyas coordenadas horizontales son coincidentes.

Hasta ahora, una de las condiciones que exigíamos a las mallas que utilizábamos era el hecho que un mismo no punto podía tener dos alturas diferentes. Esta restricción vamos a hacerla desaparecer de la forma siguiente:

- Cuando nos interese tener un punto con dos alturas distintas tendremos que crear toda una fila o columna de puntos cuyas coordenadas sean coincidentes, a excepción de la ya mencionada, aunque en ocasiones será posible aprovechar la misma fila o columna para varios puntos que tengan las mismas características.

- Cuando se añada una fila o columna deberemos tener en cuenta que el número de columnas (o filas) también aumentará en una unidad.

Veamos cómo se efectuaría el proceso de realce a una forma plana con elevación (figura 2-9-1-1):

Nº de filas antes de dar realce a esta figura en la malla, 6.

Nº de columnas antes de dar realce a la figura en la malla, 6.

Nº de filas después de dar realce a la figura, 9.

Nº de columnas después de dar realce a la figura, 9.

Veamos cómo se realiza este aumento de filas y columnas paso a paso; para ello representaremos con 0 (cero) los puntos bajos y con 1 (uno) los altos.

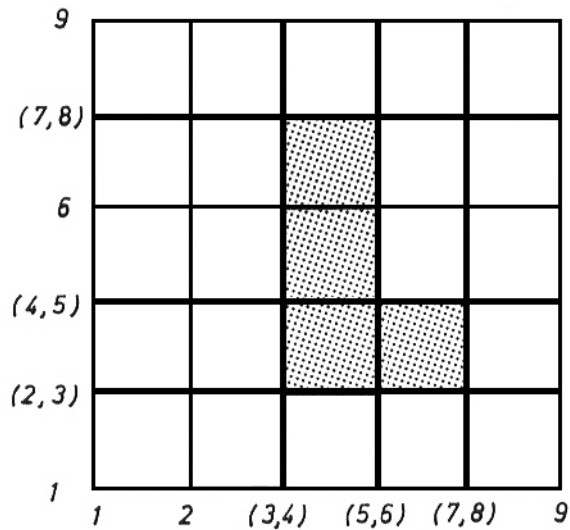


FIGURA 2-9-1-1

Malla original:

```

0 0 0 0 0 0
0 0 1 1 0 0
0 0 1 1 0 0
0 0 1 1 1 0
0 0 1 1 1 0
0 0 0 0 0 0
    
```

Malla tras añadir una columna:

```

0 0 0 - 0 0 0 0
0 0 0 - 1 1 0 0
0 0 0 - 1 1 0 0
0 0 0 - 1 1 1 0
0 0 0 - 1 1 1 0
0 0 0 - 0 0 0 0
    
```

Malla tras añadir la segunda columna:

```

0 0 0 - 0 0 - 0 0 0
0 0 0 - 1 1 - 0 0 0
0 0 0 - 1 1 - 0 0 0
0 0 0 - 1 1 - 1 1 0
0 0 0 - 1 1 - 1 1 0
0 0 0 - 0 0 - 0 0 0
    
```

Malla tras añadir la tercera columna:

0	0	0	-	0	0	-	0	0	-	0	0
0	0	0	-	1	1	-	0	0	-	0	0
0	0	0	-	1	1	-	0	0	-	0	0
0	0	0	-	1	1	-	1	1	-	0	0
0	0	0	-	1	1	-	1	1	-	0	0
0	0	0	-	0	0	-	0	0	-	0	0

Malla tras añadir la primera fila:

0	0	0	-	0	0	-	0	0	-	0	0
0	0	0	-	0	0	-	0	0	-	0	0
			-			-			-		
0	0	0	-	1	1	-	0	0	-	0	0
0	0	0	-	1	1	-	0	0	-	0	0
0	0	0	-	1	1	-	1	1	-	0	0
0	0	0	-	1	1	-	1	1	-	0	0
0	0	0	-	0	0	-	0	0	-	0	0

Malla tras añadir la segunda fila:

0	0	0	-	0	0	-	0	0	-	0	0
0	0	0	-	0	0	-	0	0	-	0	0
			-			-			-		
0	0	0	-	1	1	-	0	0	-	0	0
0	0	0	-	1	1	-	0	0	-	0	0
0	0	0	-	1	1	-	0	0	-	0	0
			-			-			-		
0	0	0	-	1	1	-	1	1	-	0	0
0	0	0	-	1	1	-	1	1	-	0	0
0	0	0	-	0	0	-	0	0	-	0	0

Malla tras añadir la tercera fila:

0	0	0	-	0	0	-	0	0	-	0	0
0	0	0	-	0	0	-	0	0	-	0	0
			-			-			-		
0	0	0	-	1	1	-	0	0	-	0	0
0	0	0	-	1	1	-	0	0	-	0	0
0	0	0	-	1	1	-	0	0	-	0	0
			-			-			-		
0	0	0	-	1	1	-	1	1	-	0	0
0	0	0	-	1	1	-	1	1	-	0	0
			-			-			-		
0	0	0	-	0	0	-	0	0	-	0	0
0	0	0	-	0	0	-	0	0	-	0	0

Donde las cifras unidas por guión tiene las mismas coordenadas horizontales aunque pueden tener o no distintas alturas.

La posición de cada signo 0 ó 1 indica los subíndices de la variable correspondiente, de manera que en este caso habría 14 puntos elevados aunque estarían situados sobre 10 vértices distintos en planta.

Véanse en las figuras que siguen a continuación los resultados obtenidos en cada paso con el programa MALLAS FICTICIAS.

Veamos como queda afectada la malla al introducir una fila o columna:

- Tipos de fila a añadir:

- Fila por debajo de una ya existente con las mismas posiciones horizontales y alturas inferiores o iguales.
- Fila por encima de una ya existente con las mismas posiciones horizontales y alturas inferiores o iguales.

- Tipos de columna a añadir:

- Columna por la izquierda de una ya existente con las mismas posiciones horizontales y alturas iguales o inferiores.
- Columna por la derecha de una ya existente con las mismas posiciones horizontales y alturas iguales o inferiores.

En el primer caso el efecto es el siguiente:

- Las filas que están por encima deben elevarse un subíndice vertical (si su subíndice era J, pasará a ser J+1).
- Lo mismo ocurre con la propia fila a repetir.
- El número subíndices verticales aumenta en una unidad.
- La relación entre las nuevas coordenadas X1, Y1, Z1 y las antiguas X, Y, Z en función de sus subíndices será:

- * Desde la fila 1 hasta la fila a reproducir se conservarán exactamente iguales $X1(I,J)=X(I,J)$, $Y1(I,J)=Y(I,J)$, $Z1(I,J)=Z(I,J)$.

- * Desde la fila a reproducir hasta la última (excluida la línea a reproducir), serán $X1(I,J)=X(I,J-1)$, $Y1(I,J)=Y(I,J-1)$ y $Z1(I,J)=Z(I,J-1)$.

- El número de columnas permanecerá invariable.
- La fila a reproducir estará repetida hasta que variemos sus alturas, y esta variación se hará en la fila del mismo número de la malla primitiva.
- Vemos claramente que el cambio sólo afecta a las filas superiores.

En el segundo caso se reproducen todas las circunstancias anteriores a excepción de la siguiente:

- La variación de la fila repetida debemos hacerla en la fila de número inmediato superior al que tenía en la malla original.

Cuando lo que queremos es intercalar una columna, el procedimiento es semejante. Veamos los efectos que producen en el tercer caso:

- Las columnas que están a la derecha deben elevarse un subíndice vertical (si su subíndice era I ahora pasaría a ser $I+1$).
- De igual modo ocurre con la columna a reproducir.
- El número de subíndices horizontales aumenta en una unidad.
- La relación entre las nuevas coordenadas $X1$, $Y1$, $Z1$ y las antiguas en función de los tres subíndices será:

* Desde la columna 1 hasta la columna a reproducir se conservarán exactamente iguales $X1(I,J)=X(I,J)$, $Y1(I,J)=Y(I,J)$, $Z1(I,J)=Z(I,J)$.

* Desde la columna a reproducir hasta la última (excluida la línea a reproducir) serán $X1(I,J)=X(I-1,J)$, $Y1(I,J)=Y(I-1,J)$, $Z1(I,J)=Z(I-1,J)$.

- El número de filas permanecerá invariable.
- La columna a reproducir estará repetida hasta que variemos sus alturas y esta variación se hará en la columna del mismo número de la malla primitiva.
- El cambio sólo afecta a las columnas de la derecha, pero no a las de la izquierda.

En el cuarto caso los efectos que se producen son los mismos a excepción del siguiente:

- La variación de la columna repetida debemos hacerla en la columna de número inmediato superior al que tenía en la malla original.

Hechas estas consideraciones, es posible sistematizar la inclusión de una fila o columna diferenciando entre los cuatro casos mediante un algoritmo que parte de los datos siguientes:

- Malla inicial definida por filas y columnas a modo de retícula espacial necesariamente cuadrada cuyos puntos tienen coordenadas de nombre $X(I,J)$, $Y(I,J)$, $Z(I,J)$ para valores de I que oscilan entre la unidad y su valor máximo IT y valores de J que oscilan entre la unidad y su valor máximo JT .

Las dos primeras coordenadas responden a la posición de los vértices sobre un plano horizontal y están situadas a igual distancia a modo de retícula regular.

```
1000 REM ALGORITMO QUE INTERCALA FILAS O COLUMNAS FICTICIAS
1010 PRINT "QUE TIPO DE LINEA DESEA INTERCALAR?"
1020 PRINT "1 COLUMNA VERTICAL POR LA IZQUIERDA"
```



```
1030 PRINT "2 COLUMNA VERTICAL POR LA DERECHA"
1040 PRINT "3 FILA HORIZONTAL POR ABAJO"
1050 PRINT "4 FILA HORIZONTAL POR ARRIBA"
1060 PRINT "EL ORDEN DE ENTRADA DE LÍNEAS DEBE SER"
1070 PRINT "DE DERECHA A IZQUIERDA Y DE ARRIBA HACIA ABAJO"
1080 GET A : IF A=0 GOTO 1080
1090 ON A GOSUB 2000,2000,4000,4000
1100 PRINT "DESEA INTERCALAR OTRA LÍNEA?"
1110 GET A$: IF A$="" GOTO 1110
1120 IF A$="S" GOTO 1010
2000 REM INTERCALADO DE COLUMNAS
2010 INPUT "POSICION DE LA COLUMNA A INTERCALAR CON REPETICION
RESPECTO A LA MALLA DE ORIGEN";IP
2020 FOR J=1 TO JT
2030 FOR I=(IT+A) TO (IP+1) STEP-1 : X(I,J)=X(I-1,J) : Y(I,J)=Y(I-1,J)
2040 Z(I,J)=Z(I-1,J) : NEXT I : NEXT J
2050 IT=IT+1 : IP=IP+A-1
2060 PRINT "DESEA TODA LA COLUMNA DE ALTURA BAJA?"
2070 GET A$: IF A$="" GOTO 2050
2080 IF A$="S" THEN FOR J=1 TO JT : Z(IP,J)=Z(IP,1) : NEXT J
2090 PRINT "NUMERO DE VERTICE A COLOCAR EN ALTURA BAJA?"
2100 PRINT "PULSAR 0 (CERO) PARA FINALIZAR"
2110 INPUT J : IF J=0 THEN RETURN
2120 Z(IP,J)=Z(IP,1) : GOTO 2090
4000 REM INTERCALADO DE FILAS
4010 INPUT "POSICIÓN DE LA FILA A REPRODUCIR RESPECTO A LA MÁS BAJA EN
LA MALLA DE ORIGEN";JP
4020 FOR I=1 TO IT
4030 FOR J=(JT+1) TO (JP+1) STEP-1 : X(I,J)=X(I,J-1)
4040 Y(I,J)=Y(I,J+A) : Z(I,J)=Z(I,J-1) : NEXT J : NEXT I
4050 JT=JT+1 : JP=JP-3+A
4060 PRINT "DESEA TODA LA FILA DE ALTURA BAJA?"
4070 GET A$: IF A$="" GOTO 4070
4080 IF A$="S" THEN FOR I=1 TO IT : Z(I,JP)=Z(1,JP) : NEXT I
4090 PRINT "Nº DE VÉRTICE A COLOCAR EN ALTURA BAJA?"
4100 INPUT I : IF I=0 THEN RETURN
4110 Z(I,JP)=Z(1,JP) : GOTO 4090
```

2.9.2 - TRATAMIENTO DE MALLAS FICTICIAS

Un tratamiento completo de mallas ficticias debe comprender los siguientes apartados:

- Lectura de datos de la malla original a partir del soporte donde se encuentre y considerando que el número de filas y columnas sea posible variarlo para leer datos de mallas de distinto número de puntos incluso cuando no son cuadradas.

- Menú para poder intercalar líneas de los cuatro tipos que hemos definido en función de las necesidades.

- Indicaciones referentes al orden de entrada de estas líneas intercalables para evitar que la introducción de unas líneas afecten a otras. En la definición que nosotros hemos hecho el orden debe ser de mayor a menor índice, tanto para filas como para columnas.

- Proceso de intercalado con adaptación de las coordenadas a los nuevos subíndices, que es en realidad lo que hemos desarrollado hasta ahora.

- Posibilidad de visualizar la malla una vez hechos todos los cambios y grabarla en un soporte en caso de ser satisfactorio.

Los puntos segundo, tercero y cuarto ya los hemos desarrollado; veamos un algoritmo que servirá para el primer punto:

```
100 REM LECTURA DE LOS DATOS DE UNA MALLA RECTANGULAR DE N° DE
FILAS Y COLUMNAS CONOCIDOS
110 PRINT "NOMBRE DE ARCHIVO ORIGEN?"
120 INPUT C$: PRINT "NUMERO DE COLUMNAS?" : INPUT IT
130 PRINT "NUMERO DE FILAS?,JT
140 REM LECTURA
150 ROPEN # 1, C$
160 FOR I01 TO IP : FOR J=1 TO JP
170 INPUT # 1, X(I,J), Y(I,J), Z(I,J) : NEXT J : NEXT I
```

2.9.3 - OBTENCIÓN DE LA PERSPECTIVA EN ESTE TIPO DE MALLA

El último punto que es en realidad el objetivo final, podemos realizarlo mediante dos procedimientos como vamos a ver seguidamente:

- Crear un sistema de visualización adecuado al número de filas y de columnas variables.

- Convertir la malla resultante en otra malla con el número de filas y columnas adecuados para poder utilizar el sistema de visualización que disponemos.

Aunque no sería excesivamente complicado realizar la primera opción, hemos creído conveniente efectuar la segunda para de esta forma utilizar el programa PROYECCIÓN MALLA. Veamos el algoritmo que convierte una malla con un número menor de filas y columnas al que nosotros hemos utilizado (20*20) en una malla cuadrada que cumpla todas las restricciones.

```
5000 REM AJUSTE MALLA A DIMENSIONES ADECUADAS
5005 IF IT=20 GOTO 5030
5010 IF IT<20 THEN FOR J01 TO JT : FOR I=IT TO 20 : X(I,J)=X(IT,J) : Y(I,J)=Y(IT,J) :
Z(I,J)=Z(IT,J) : NEXT I : NEXT J : GOTO 5030
5020 PRINT "NO ES POSIBLE CON MALLA DE 20*20 : END
5030 IF JT=20 THEN RETURN
5040 IF JT<20 THEN FOR I=1 TO IT : FOR J=JT TO 20 : X(I,J)=X(I,JT) : Y(I,J)=Y(I,JT) :
Z(I,J)=Z(I,JT) : NEXT J : NEXT I : RETURN
5050 GOTO 5020
```

El proceso de grabación sería:

```
6000 PRINT "DESEA GRABAR EL NUEVO ARCHIVO?"
6010 GET A$ : IF A$="" GOTO 6010
6020 IF A$="S" GOSUB 6500
6030 END
6500 REM GRABACION EN DISCO
6510 INPUT "NOMBRE QUE ASIGNAMOS AL ARCHIVO?";B$
6520 WOPEN # 2, B$
6530 FOR I=1 TO 20 : FOR J=1 TO 20
6540 PRINT # 2, X(I,J), Y(I,J), Z(I,J)
6550 CLOSE # 2
6560 END
```

Tras efectuarse este algoritmo podíamos utilizar el programa PROYECCIÓN MALLA PARA OBTENER LA PERSPECTIVA.

Todo lo expuesto y con suficientes ejemplos, se ha incorporado en el programa MALLAS FICTICIAS que se encuentra en el apartado correspondiente de la SÍNTESIS.