CLIPPING USING HOMOGENEOUS COORDINATES

James F. Blinn
Caltech/JPL

Martin E. Newell
XEROX/PARC

Abstract

Clipping is the process of determining how much of a given line segment lies within the boundaries of the display screen. Homogeneous coordinates are a convenient mathematical device for representing and transforming objects. The space represented by homogeneous coordinates is not, however, a simple Euclidean 3-space. It is, in fact, analagous to a topological shape called a "projective plane". The clipping problem is usually solved without consideration for the differences between Euclidean space and the space represented by homogeneous coordinates. For some constructions, this leads to errors in picture generation which show up as lines marked invisible when they should be visible. This paper will examine these cases and present techniques for correctly clipping the line segments.

1. INTRODUCTION

Homogeneous coordinates have long been used in computer graphics as a convenient mathematical device for representing and transforming objects [3]. However, in spite of the uniformity of representation and operation afforded by homogeneous coordinates, they are not often exploited to the full. This is probably due to a lack of publications explicitly directed at clarifying the use of these techniques. Sutherland and Hodgman [4] provide one of the very few discussions on this topic as an appendix to their paper on Polygon Clipping. The present paper presents techniques for using homogeneous coordinates to represent three dimensional objects, and shows how the homogeneous representation can be carried through transformation and clipping in a consistent way. It is largely a reiteration of the appendix of [4] and an expansion of the ideas presented there.

While it is assumed that the reader has some knowledge of homogeneous coordinate representations, the following sections are included both as a review and to introduce basic techniques and terminology used in the remainder of the paper.

1.1 Homogeneous Coordinates

The representation and transformation of objects in 3 dimensions is usually performed in analysis in a Cartesian coordinate system. Thus three coordinates (X,Y,Z) are sufficient to represent a point in three dimensions. A transformation such as rotation or scale is then represented by a 3x3 matrix. Multiplication of the position vector by this matrix yields a transformed position vector. Certain points (notably points at infinity) and certain transformations (notably translations and perspective projection) are not representable in this scheme. The notation called "homogeneous coordinates" has been devised which will encompass all points and transformations of interest. In this scheme, each point is represented in a redundant manner by 4 coordinates. These four coordinates will be named, in this discussion, as lower case letters (x, y, z, w). The redundancy is expressed in the convention that any (non-zero) multiple of all components of the homogeneous representation of a point is another homogeneous representation of the same point. To get from the homogeneous representation to the more conventional representation the redundancy is removed by dividing each component by w, unless $w=\emptyset$. This yields a vector which, by the homogeneous convention, still represents the same point but has a w component of 1. The first 3 components are then the conventional components of the point, named with upper case letters (X,Y,Z).

$$(x,y,z,w) \rightarrow (x/w,\ y/w,\ z/w,\ 1) \rightarrow (X,Y,Z)$$

All homogeneous points with w=1 are already in this conventional form. In fact, unless there is a good reason to do so, points on objects to be modelled are usually initially specified with w=1. Certain transformations performed on these objects might, however, generate points with $w \neq 1$. The division operation can be thought of as a projection of a point in 4-space onto the plane w=1 by a line through the origin. We show this by examining a section of the (x,y,z,w) space where $y=z=\emptyset$. The remaining x and w coordinates appear as in figure 1.

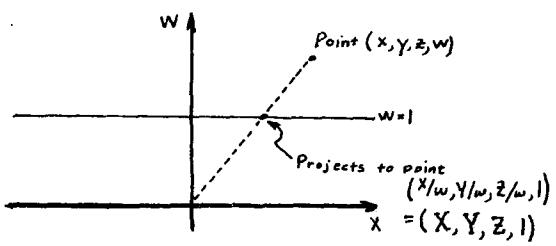

Figure 1 - A Homogeneous Point

All homogeneous points (x,y,z,w) which represent the same real point (X,Y,Z) lie on the line through the origin and (X,Y,Z,1).

1.2 Homogeneous Transformations

A point, P, represented in homogeneous coordinates (x,y,z,w) is linearly transformed into an image point, P'=(x',y',z',w') by multiplication by a 4x4 matrix M.

$$P' = P\ M$$

To interpret the effect of this matrix on the 3D image of the point let M be partitioned as:

$$M = \begin{bmatrix} r & r & r & p \\ r & r & r & p \\ r & r & r & p \\ t & t & t & s \end{bmatrix}$$

The 3x3 partition, denoted by r, represents rotation and scaling. The 1x3 partition, denoted by t, represents translation. The 3x1 partition, denoted by p, represents perspective. As with points, any (non-zero) multiple of a homogeneous transformation matrix represents the same transformation. Therefore the 1x1 partition, denoted by s, has the same meaning for transformations as the w component does for points.

1.3 Line Segments

Line segments will be represented here in a parametric form as the weighted sum of the two endpoints $P_1=(x_1,y_1,z_1,w_1)$ and $P_2=(x_2,y_2,z_2,w_2)$.

$$P = (1-a)\ P_1 + a\ P_2$$

$$\emptyset <= a <= 1$$

As the parameter "a" varies from $\emptyset$ to 1 the generated point moves linearly from P_1 to P_2. To find the conventional coordinates of points on this segment, each point on the line is projected onto the w=1 plane. This is illustrated in figure 2.

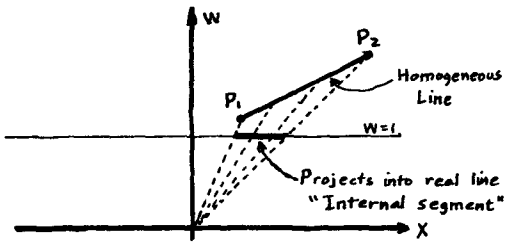

Figure 2 - A Homogeneous Line Segment

For segments defined with both endpoints having w=1 all the interpolated w values are 1 and the line segment and its projection are the same. For segments defined with each endpoint having a positive value of w (as in figure 2) a similar line appears. However for segments defined with opposite signs of w on each endpoint a more unusual situation occurs. The segment generated by the linear interpolation in 4-space is quite ordinary. The segment generated by projecting each point of this onto w=1 must pass through infinity at the point where the 4-space segment passes through w=$\emptyset$. This is illustrated below with the points P_1 and $-P_2$. The endpoints represent the same projected points as those in figure 2. The projected line segment, however, starts at P_1 and goes in the direction away from P_2, passes through infinity and comes back to meet P_1 from the other side. This is illustrated in figure 3.

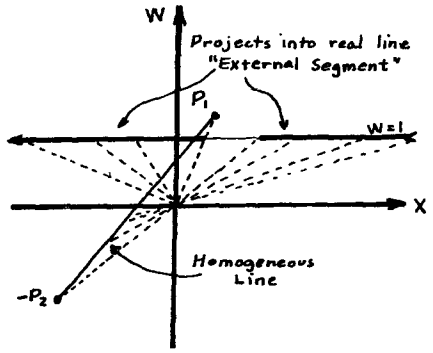

Figure 3 - An External Line Segment

This line segment consists of the complement of th set of points in the example of figure 2. It il lustrates an alternative way of connecting point (X_1,Y_1,Z_1) and (X_2,Y_2,Z_2). It will be called a "external line segment" in contrast to the "inter nal line segment" of figure 2. These types o lines can show up in practical applications as result of perspective transformations and as a re sult of some commonly used methods for definin curves. It is these external line segments whic can cause trouble in clipping algorithms.

2. CLIPPING

Clipping is the operation of removing portion of a line segment which are outside the scree boundaries. We will begin by examining a simpl clipping algorithm. This algorithm will work cor rectly only for the region w>$\emptyset$ so we will initiall concern ourselves with this region.

To simplify the arithmetic, it is convenier to clip to the boundaries -1<X<+1 and -1<Y<+1. Th viewing transformation can be adjusted to map ar desired object window to this region. The clippir boundaries are thus the planes

X=-1 (left)
X=+1 (right)
Y=-1 (bottom)
Y=+1 (top)

In the homogeneous representation these become:

x/w=-1
x/w=+1
y/w=-1
y/w=+1

or the four homogeneous planes

w+x=$\emptyset$
w-x=$\emptyset$
w+y=$\emptyset$
w-y=$\emptyset$

Looking at the left and right boundaries in the xw plane we have figure 4.

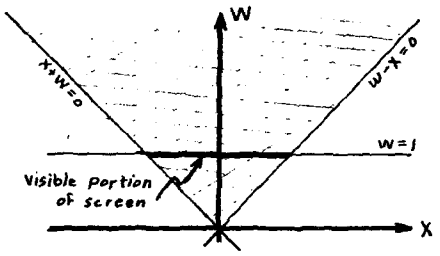

Figure 4 - The Visible Screen Region

The region on the w=1 plane between x=-1 and x=+1 represents the visible region in X after the homogeneous division has been performed. Any points in the cross hatched area will project onto this region and are thus visible. A point is visible if

$$w+x > 0$$
$$\text{and } w-x > 0$$

Note that all points within the cross-hatched area satisfy this condition.

If a line segment lies partly inside and partly outside the screen it will penetrate one of the homogeneous clipping planes. We need to find the point of intersection. This can be expressed as a value for "a" in the parametric definition of the line segment.

$$P = (1-a)\ P_1 + a\ P_2$$

Suppose the line segment crosses the w+x=0 plane. The value of "a" at which this occurs is

$$[(1-a)w_1+(a)w_2] + [(1-a)x_1+(a)x_2] = 0$$

or

$$a = (w_1+x_1) \ / \ ((w_1+x_1) - (w_2+x_2))$$

The quantity w_1+x_1 is proportional to the distance from the point P_1 to the plane x+w=0. Therefore it may be interpreted as a transformed coordinate of P_1 relative to the boundary x+w=0. For this reason it will be called a "Boundary Coordinate". For any point there is a boundary coordinate for each clipping boundary.

BL = w+x (left)
BR = w-x (right)
BB = w+y (bottom)
BT = w-y (top)

These are defined so that a positive value indicates that a point is on the visible side of the clipping plane. If a line $(1-a)P_1+aP_2$ crosses, for example, the left boundary it does so at $a=BL_1/(BL_1-BL_2)$. A similar expression holds for the other boundaries.

3. THE HOMOGENEOUS PERSPECTIVE TRANSFORM

External line segments first appear when using the perspective transformation. A perspective projection essentially causes division of X and Y by the Z distance in front of the eye. The homogeneous perspective transformation makes clever use of the homogeneous division (which must be done anyway) by merging the Z division with it. The simplest form models the eye at (0,0,-1). To achieve a perspective projection the X and Y should then be divided by Z+1. In homogeneous terms x/w and y/w should be divided by z/w+1=(z+w)/w, becoming x/(z+w) and y/(z+w). This can be expressed in matrix form as

$$\begin{pmatrix} x & y & z & w \end{pmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{pmatrix} x & y & z & z+w \end{pmatrix}$$

The effect of thistransformation is to change the boundaries of a "view cone" radiating from the eye into the same parallel clipping boundaries that were used for orthographic projections. Points with z/w=0 are unchanged. Points with z/w>0 are scaled down while points with -1<z/w<0 are scaled up. See figure 5.

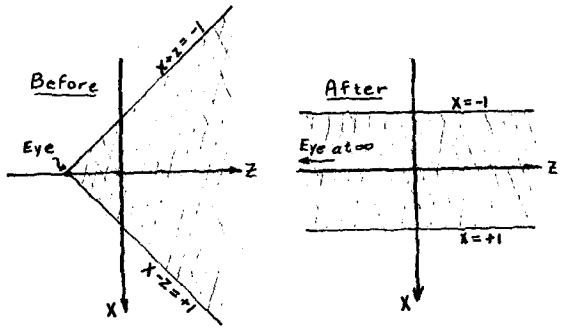

Figure 5 - Perspective Transform in XZ

The process of effectively clipping to the view cone for perspective pictures is performed by first applying the homogeneous perspective transformation and then clipping to the same boundaries defined previously. Indeed, this interpretation of the perspective transformation explains the use of post-perspective transformations to achieve special viewing effects such as projection onto an oblique viewing plane. Such a projection can be considered as a conventional perspective projection followed by a translation of the required part of the projection to the clipping region. It may be verified that composition of two such transformations yields one which correctly maps the boundaries of the oblique viewing cone into the same clipping planes used above.

Although the Z coordinate of a point might not seem immediately useful after perspective projection, it is necessary for hidden line/surface computations and for depth cueing. This transform has the important property that it includes z such that straight lines remain straight. Examining the zw plane we see that the transformation is merely a skew along the w axis, figure 6.

Note that objects originally defined in the w=1 plane become distorted when, after transformation, they are projected back into the new w=1 plane. The eyepoint transforms into (0 0 -1 0), a point infinitely distant in the minus z direction. Points infinitely far away in the positive z direc-

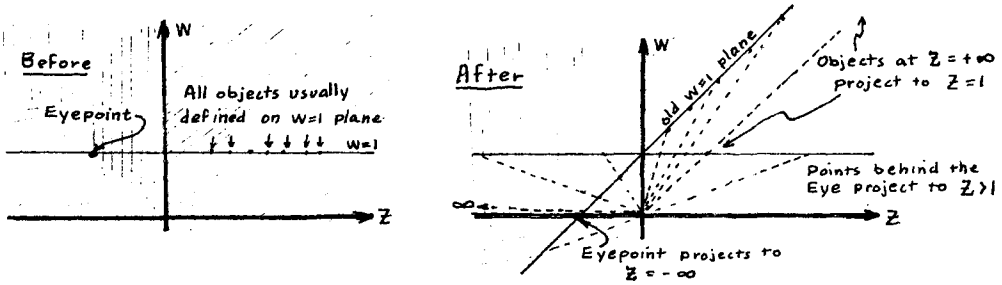

Figure 6 - Perspective Transform in zw

tion transform into z/w=1. Note the effect of the transformation on the three cross-hatched regions of figure 6. In general the following transformations of regions in z have taken place

before	after
$-\infty < z/w < -1$	$1 < z/w < +\infty$
$-1 < z/w < 0$	$-\infty < z/w < 0$
$0 < z/w < +\infty$	$0 < z/w < +1$

Points that were behind the eye have "wrapped around" through - and are now at z/w > 1.

Let us now consider the effect of the perspective transformation on line segments and how they are clipped. For segments which are totally in front of the eye nothing very unusual happens. Consider, however, a segment from a point in front of the eye to a point behind the eye (a perfectly reasonable occurrence when arbitrary viewing positions are allowed). After the perspective transformation this becomes an external line segment. It starts out at some Z<1, proceeds in the negative Z direction past the eye (at $-\infty$ in Z), wraps around to positive Z and ends at some Z>1, see figure 7.

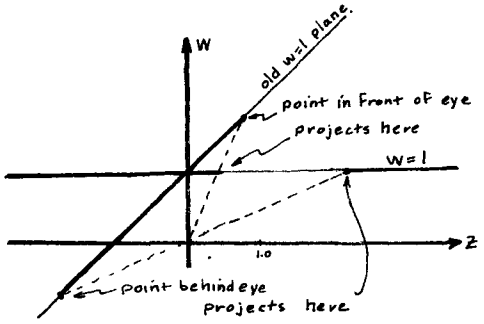

Figure 7 - Perspective Transform of Line Segment

If the two endpoints are projected back onto the w=1 plane both endpoints could quite possibly lie within the visible region of the screen in X and Y. This is despite the fact that the point behind the eye is quite obviously invisible. Furthermore, these points should be connected by an external line segment rather than an internal one. This case is difficult to distinguish from the case of two ordinary visible points which started out in front of the eye. (It can be detected by noting that the endpoints of external line segments straddle the z/w=1 plane). These problems can be resolved by clipping all segments in the homogeneous space prior to projecting back onto w=1. In the present case, the line passing by the eye would be clipped at the X edge of the screen before it even passed the eye, see figure 8.

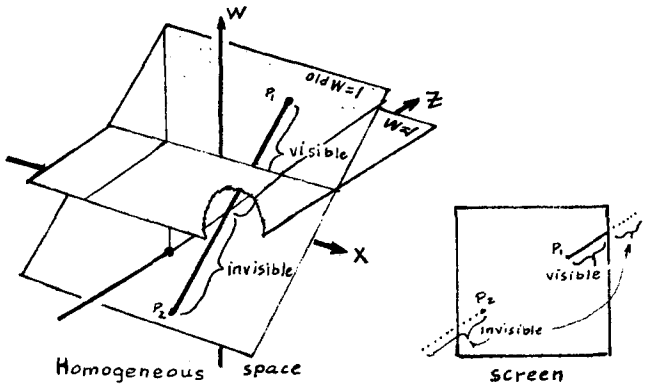

Figure 8 - Clipping of External Segment

The portions of the external segment which give trouble are clipped off by one of the left, right, bottom or top planes.

The X,Y clipping process is sometimes augmented by a clipping operation in Z. This is done primarily for the purpose of restricting the range of the Z coordinate. Points between the eye and the Z=0 plane (i.e. the screen) take up a small portion of the real world, but after perspective transformation they stretch from $-\infty$ to 0. To avoid the need to represent points with infinite Z coordinates we can clip away those with z/w less than some amount using a "near" clipping plane. In addition, to further restrict the range of Z values a "far" clipping plane is sometimes included. Z clipping can be standardized just as X,Y clipping by defining the visible region in Z to be 0<Z<+1. The actually desired locations of the near and far boundaries can be incorporated into the transformation matrix as a scale and translation of the z coordinate. Thus we have two new clipping planes and two boundary coordinates.

$$BN = z \quad \text{(near)}$$
$$BF = w - z \quad \text{(far)}$$

A point is visible with respect to the z clipping planes if both these quantities are positive.

The main point of this section is, then, that for our first exposure to external line segments, those formed by the perspective transform, the original clipping algorithm still works. We have seen that the clipping algorithm works correctly for lines which remain in the w>0 region and for those which dip into the w<0 region due to the perspective transformation. The clipping should be per-

formed before the homogeneous division, however. The addition of Z clipping is useful to restrict the range of Z values after the perspective transform, but is not necessary to the correct elimination of line segments behind the eye.

5. RATIONAL PARAMETRIC CURVES

This section introduces a standard modelling technique which happens to generate lines which are not correctly clipped. This is the technique of modelling curved lines parametrically with rational polynomial functions. To illustrate the problem it is only necessary to consider two dimensional planar curves. We will therefore assume the Z coordinate is always zero and not include it in subsequent matrix equations.

The simplest, non-linear, parametric curves are the conic sections. It is possible to represent any conic section by

$$X = P(t)/R(t)$$
$$Y = Q(t)/R(t)$$

or

$$x = P(t)$$
$$y = Q(t)$$
$$w = R(t)$$

where P(t), R(t) and S(t) are quadratic polynomials. To prove this we start with the simple parabola X=Y . This can be represented parametrically in homogeneous coordinates as

$$(x\ y\ w) = (t^2\ t\ 1)$$

By the homogeneous convention, any non-zero scalar multiple of each point on the parabola also lies on the parabola. The locus of all such points in homogeneous space is the cone $(x/w)=(y/w)^2$. The parabola is the intersection of this cone with the w=1 plane. This cone, incidentally, is an elliptic cone rather than a circular cone, as shown in figure 9.

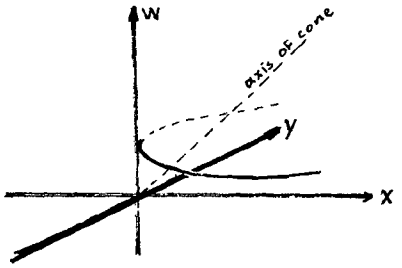

Figure 9 - Elliptic Cone

Note that this matches the classical defenition of a parabola as the intersection of a cone with a plane parallel to one of its sides. Now by various rotations of the parabola (and thus the cone) about the origin (representable by a 3x3 matrix) we can change the relative orientation of the plane of intersection and thus generate different conic sections. The multiplication of the vector $(t^2\ t\ 1)$ by a 3x3 matrix yields 3 general quadratic polynomials.

$$(x\ y\ w) = (t^2\ t\ 1)\ M = (P(t)\ Q(t)\ R(t))$$

For example, by rotating 45 degrees around the y axis we can make the axis of the cone perpendicular to the intersecting plane (w=1) and generate an ellipse (since the cone is elliptic). Then, by scaling by $\sqrt{2}$ in y the ellipse turns into a circle.

$$(x\ y\ w) = (t^2\ t\ 1)\begin{bmatrix} 1/\sqrt{2} & 0 & 1/\sqrt{2} \\ 0 & \sqrt{2} & 0 \\ -1/\sqrt{2} & 0 & 1/\sqrt{2} \end{bmatrix}$$

$$X = (t^2-1)/(t^2+1)$$
$$Y = (\ 2t\ )/(t^2+1)$$

The reader can verify that $X^2+Y^2=1$ independent of the value of t.

Alternatively, by rotating -45 degrees around y, the axis of the cone will become parallel to w=1. Such an intersection yields a hyperbola.

$$(x\ y\ w) = (t^2\ t\ 1)\begin{bmatrix} 1/\sqrt{2} & 0 & -1/\sqrt{2} \\ 0 & 1 & 0 \\ 1/\sqrt{2} & 0 & 1/\sqrt{2} \end{bmatrix}$$

$$X = (t^2 + 1)/(1-t^2)$$
$$Y = (t\ \sqrt{2}\ )/(1-t^2)$$

How does the clipping process work when applied to this hyperbola? This curve happens to lie wholly outside the standard clipping region so we will scale it down by a factor of 2 to make the problem interesting. When projected onto the w=1 plane, it appears as in figure 10.

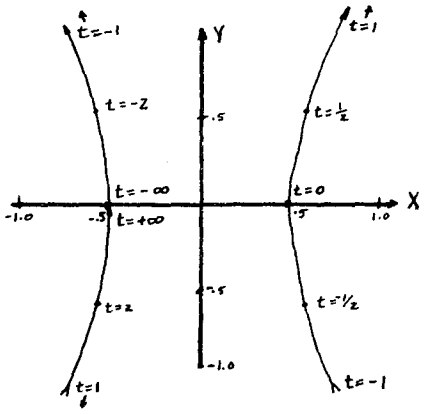

Figure 10 - Parametric Hyperbola

Thus, two branches should be visible on the resultant display. To draw the curve we evaluate the x, y, and w functions at equal steps in t and connect the points with straight line segments. This traces out the rotated and scaled parabola in homogeneous space. This parabola appears as in figure 11.

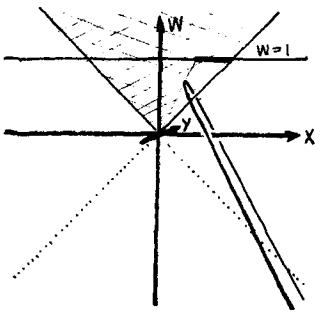

Figure 11 - Side View of Rotated Parabola

Note that only one branch falls within the visible region in the sense we have defined it so far. The clipping algorithm will remove an entire branch of the hyperbola which should have been visible.

6. CLIPPING WITH NEGATIVE W

To properly clip such objects we must re-examine the definition of the "visible" region. For example, for the left clipping boundary a point is visible if $X = x/w > -1$. This implies two possible sets of conditions for visibility. That is

$$w>0 \text{ and } w+x>0$$

or

$$w<0 \text{ and } w+x<0$$

A point must be tested against two planes, $w+x=0$ and $w=0$. Two planes are necessary because of the topological properties of the space represented by homogeneous coordinates. Riesenfeld [2] points out that this space is not the usual Euclidean 3-Space. It is, in fact, a space whose properites are analagous to a shape known as a "projective plane". The important difference is that, for a projective space, a single (infinitely extended) flat plane does not divide space into two distinct regions. Just placing a plane between two points does not separate them. There is always an alternate path (perhaps through infinity) connecting them. In order to separate two regions in homogeneous space it is necessary to use two dividing planes. This is shown very nicely in Barr [1].

The complete visible region in the x dimension is formed by the intersection of the newly defined left and right visible regions. Note that each point in the new visible region, when projected onto the w=1 plane falls within the $-1<X<+1$ region. The two visible regions are shown labelled A and C in figure 13. Points in region C would all be marked invisible by the original algorithm because they occur on the "invisible" side of both the $w-x=0$ and $w+x=0$ planes. It is in just this region that the points lie for the missing branch of the hyperbola, see figure 12.

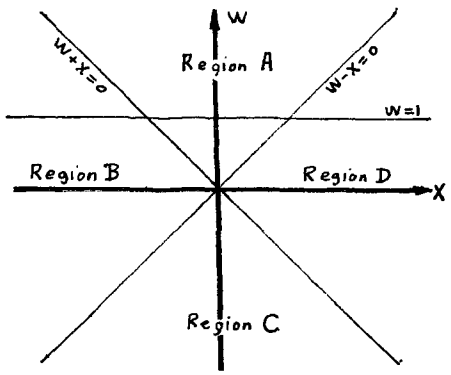

Figure 12 - Complete Visible Region In xw

One interesting case that can occur is when an external line segment has one endpoint in region A and the other in region C, as shown in figure 13.

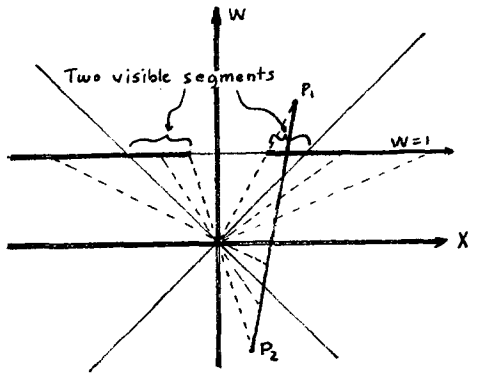

Figure 13 - Multiple Visible Segments

The segment in homogeneous space intersects the x clipping surface at two points. Two disjoint portions of the segment will be mapped into the visible screen region. This is precisely what happens with the hyperbola for the line segment between parameter values t=.9 and t=1.1. At t=.9 one branch is just approaching its asymptote going to infinity towards the upper right. At t=1 the point on the curve passes through infinity. At t=1.1 it has wrapped around and is coming in from the lower left. See figure 14. Any complete clipping algorithm which works in homogeneous coordinates must, therefore, be able to generate two output segments for one input segment.

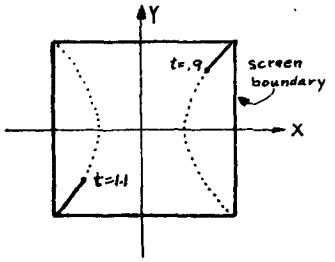

Figure 14 - Multiple Segments of Hyperbola

One effect of the inclusion of the visible region at negative w is that the z clip is no longer optional in order to remove objects behind the eye. Before, when using only the region for positive w, a line passing behind the eye was correctly clipped off where it left the screen. With the complete visible region, if it extends far enough behind the eye it may penetrate the negative w visible region and re-appear on the screen in the same manner as multiple segments of a hyperbola. This problem is solved by including a "far" clip bounded by the planes w=0 and w-z=0, keeping points with Z<1. This just states that the perspective view cone must be closed off by the plane at infinity, which transforms to the plane z/w=1 after the perspective transform. See figure 15.

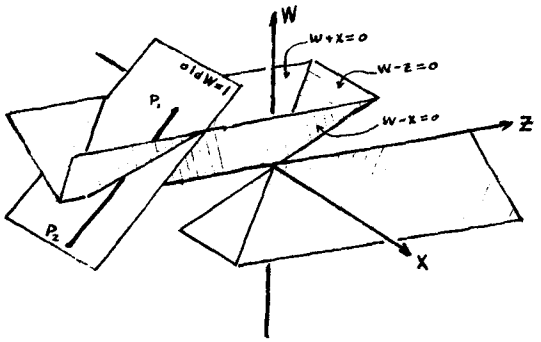

Figure 15 - Complete Visible Region In xzw

7. ALGORITHMS

Given the new definition of the visible region, we turn to the question of how, algorithmically, we can clip to this region. The big problem is that two output segments may be generated from one input segment. The basic control structure of most clipping algorithms allows only one output segment. Sutherland and Hodgman solved the problem of two disjoint clipping regions by transforming and clipping each object twice, once the transformation matrix intact and a second time after multiplying the matrix by -1. This effectively uses only the positive w clipping region but mirrors the line segments about the w=0 plane to generate the missing lines. This approach, while not complicating the control structure, requires each point to be transformed twice. We would like to avoid this.

The next step up from this would not require doubly transforming the object but would place a processor between the transformation and clipping stages. It would do a simple test against the w=0 plane. If a segment was totally above it would be passed directly to the clipper. If it was completely below it would be mirrored and passed to the clipper. If it straddled the w=0 plane it would be passed to the clipper twice, once mirrored and once not. In this case also, the clipper itself works only with the positive w portion of the clipping region. Double output segments come from two calls to the clipper.

A general solution to the problem would be to invent a clipper which actually handles disjoint clipping regions. The first step necessary is to redefine the boundary coordinates to accurately reflect the visibility of points. Each is formed as the product of the equations of the two planes defining that boundary. These would be

$$BL = w(w+x)$$
$$BR = w(w+x)$$
$$BB = w(w+y)$$
$$BT = w(w-y)$$
$$BN = w\ z$$
$$BF = w(w-z)$$

They are, again, chosen so that they are positive in both visible regions. Clipping is then performed one boundary at a time with the surviving portions of the segment being passed on to the next boundary. This requires two plane intersection tests for each boundary. For reasons of economy we can merge the left and right clipping regions into one global x clipping surface by defininining

$$BX = (w+x)(w-x)$$

An incoming line segment would be tested against each of these component planes. If it intersected only one, the visible portion would be retained and passed to the y clip. If it intersected both, the signs of (w+x) and (w-x) would determine if it was the center section or the two end sections that was visible. The y and z visible regions are similarly defined by

$$BY = (w+y)(w-y)$$
$$BZ = w\ \ (w-z)$$

It is not clear whether a disjoint clipper would be superior to the simpler mechanism of mirroring the line segments.

8. CONCLUSIONS

This paper has shown that, under certain circumstances the clipping region traditionally used in computer graphics is incomplete. There is an additonal such region mirrored about the w=0 plane. Points are generated in this region usually only for certain modelling techniques, such as rational polynomial parameteric curves. In order to properly clip such curves the clipper must be capable of generating two output segments for one input segment.

REFERENCES

[1] Barr, S., Experiments in Topology, Thomas Y. Crowell Co., New York, 1964.

[2] Riesenfeld, R. F., "Homogeneous Coordinates and Projective Planes in Computer Graphics", JACM, to appear.

[3] Roberts, L. G., "Homogeneous Matrix Representation and Manipulation of N-Dimensional Constructs", MIT Lincoln Laboratory, MS 1405, May 1965.

[4] Sutherland, I. E., and Hodgman, G. W., "Reentrant Polygon Clipping", CACM, Vol 17, No 1 (Jan 1974), pg 40.