Additional mathematical results

In this appendix we consider some useful mathematical results that are relevant to three-dimensional computer graphics but were not considered in the book itself.

0.1 The intersection of a line with non-planar primitive shapes

In this section we consider the calculation of the point of intersection between a line and some non-planar shapes that often occur in 3DCG work.

0.1.1 Intersection of a line with a sphere

A sphere is specified by two parameters: a position vector,$\mathbf{P}_c$, giving the center of the sphere and a scalar r, its radius. A line $\mathbf{p} = \mathbf{P}_l + \mu\mathbf{d}$ intersects the sphere once, twice or not at all.

Figure 1 shows two paths from the origin O to the point $\mathbf{p}$ on the surface of the sphere where the line pierces it.

Equating the two vector paths to the point $\mathbf{p}$ gives:

$$\mathbf{P}_c + \mathbf{q} = \mathbf{P}_l + \alpha\mathbf{d} \tag{1}$$

α is the length along the line from $\mathbf{P}_l$ to the point of intersection between line and sphere. Since $\mathbf{q}$ is a vector that extends from the center, to the surface, of the sphere its length is r and thus $\mathbf{q} \cdot \mathbf{q} = r^2$. Writing equation 1 as:

$$\alpha\mathbf{d} + (\mathbf{P}_l - \mathbf{P}_c) = \mathbf{q}$$

and taking the dot product of each side with itself gives:

$$(\mathbf{d} \cdot \mathbf{d})\alpha^2 + 2\mathbf{d} \cdot (\mathbf{P}_l - \mathbf{P}_c)\alpha + (\mathbf{P}_l - \mathbf{P}_c) \cdot (\mathbf{P}_l - \mathbf{P}_c) = \mathbf{q} \cdot \mathbf{q} = r^2 \tag{2}$$

Eliminating $\mathbf{q}$ reveals a quadratic equation of the form $(a\alpha^2 + b\alpha + c = 0)$ in α. In this equation the coefficients are given by:

$$\begin{aligned} a &= \mathbf{d} \cdot \mathbf{d} \\ b &= 2\mathbf{d} \cdot (\mathbf{P}_l - \mathbf{P}_c) \\ c &= (\mathbf{P}_l - \mathbf{P}_c) \cdot (\mathbf{P}_l - \mathbf{P}_c) - r^2 \end{aligned}$$

Once α has been determined we can locate $\mathbf{p}$ from the equation of the line $\mathbf{p} = \mathbf{P}_l + \alpha\mathbf{d}$. The procedure is illustrated in the form of the algorithm given in Figure 2

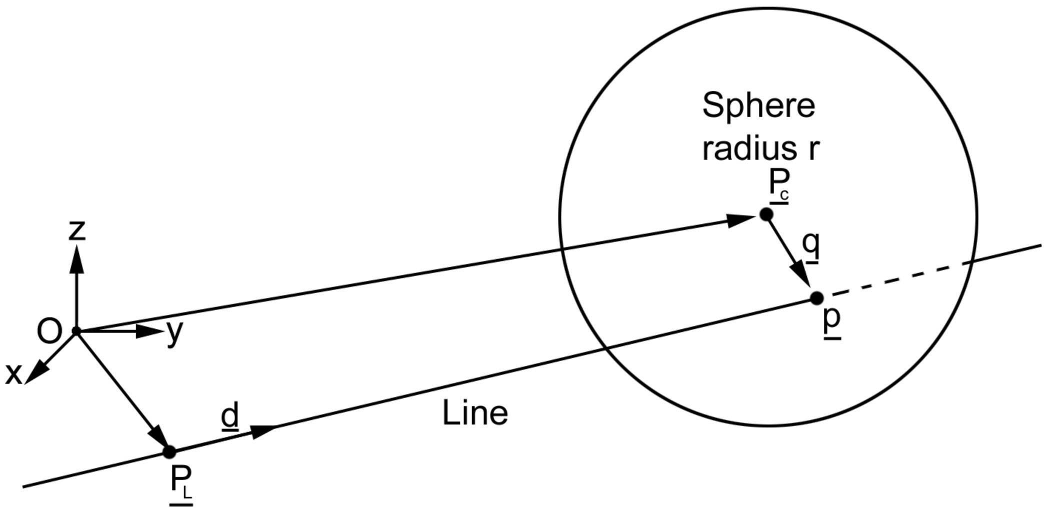

Figure 1: Geometry of the intersection between line and sphere.

$\mathbf{P}_{lc} = (\mathbf{P}_l - \mathbf{P}_c)$
$a = \mathbf{d} \cdot \mathbf{d}$
$b = 2\mathbf{d} \cdot \mathbf{P}_{lc}$
$c = \mathbf{P}_{lc} \cdot \mathbf{P}_{lc} - r^2$
$b_4 = b^2 - 4ac$
if $b_4 < 0$ then no intersection
else {
    if $b_4 < \epsilon$ {
        $$\alpha_1 = \alpha_2 = \frac{-b}{2a}$$
    }
    else {
        $b_4 = \sqrt{b_4}$
        $$\alpha_1 = \frac{(-b + b_4)}{2a}$$
        $$\alpha_2 = \frac{(-b - b_4)}{2a}$$
    }
    $\mathbf{p}_1 = \mathbf{p}_l + \alpha_1 \mathbf{d}$
    $\mathbf{p}_2 = \mathbf{p}_l + \alpha_2 \mathbf{d}$
}

Figure 2: Algorithm for determining the intersection of a line and sphere.

0.1.2 Intersection of a line and a cylinder

A similar argument to that used to calculate the points(s) of intersection between a line and sphere will produce an algorithm that:

- determines whether there are one, two or no points of intersection, one point of intersection is quite rare.
- determines whether the intersection is on the side (curved part) or at the end (flat disks) of the cylinder.
- determines the position vector of the intersection point(s), when they exist.

The geometry of the problem is shown in Figure 3. The cylinder is specified by position vectors $\mathbf{P}_1$ and $\mathbf{P}_2$ which define points lying at each end of the axis of a finite cylinder. The cylinder has radius r and the vector $\mathbf{s} = (\mathbf{P}_2 - \mathbf{P}_1)$ determines the axis of the cylinder. As before we will use the following equation for the line: $\mathbf{p} = \mathbf{P}_l + \mu\mathbf{d}$.

To develop an algorithm to solve this problem we proceed in the following steps:

1. Find the intersection point(s) between an **infinitely long cylinder** of radius r and central axis passing through $\mathbf{P}_1$ and $\mathbf{P}_2$.

   As before, we find two alternate paths leading from the origin to the point of intersection. They give rise to the equation:

   $$\mathbf{P}_l + \mu\mathbf{d} = \mathbf{P}_1 + \lambda\mathbf{s} + \mathbf{q} \tag{3}$$

   where λ is the proportion of $\mathbf{s}$ that leads from $\mathbf{P}_1$ to a point on the cylinders axis where the vector $\mathbf{q}$ originates. (The argument now proceeds to obtain a quadratic in μ and also to eliminate λ.)

   The vectors $\mathbf{s}$ and $\mathbf{q}$ are perpendicular, therefore:

   $$\mathbf{s} \cdot \mathbf{q} = 0 \tag{4}$$

   Since $\mathbf{q}$ runs from the cylinder's axis to its surface:

   $$|\mathbf{q}| = r \tag{5}$$

   Eliminating $\mathbf{q}$ and λ from 3 results in a quadratic in μ from which the point(s) of intersection can be determined.

   Taking the dot product of both sides of 3 with $\mathbf{s}$ and using 4 gives:

   $$(\mathbf{P}_l - \mathbf{P}_1) \cdot \mathbf{s} + \mu\mathbf{d} \cdot \mathbf{s} = \lambda\mathbf{s} \cdot \mathbf{s}$$

   which may be rearranged to give:

   $$\lambda = \frac{(\mathbf{P}_l - \mathbf{P}_1) \cdot \mathbf{s} + \mu\mathbf{d} \cdot \mathbf{s}}{\mathbf{s} \cdot \mathbf{s}} \tag{6}$$

   Substituting 6 into 3 gives:

   $$(\mathbf{P}_l - \mathbf{P}_1) + \mu\mathbf{d} - \left(\frac{(\mathbf{P}_l - \mathbf{P}_1) \cdot \mathbf{s} + \mu\mathbf{d} \cdot \mathbf{s}}{\mathbf{s} \cdot \mathbf{s}}\right)\mathbf{s} = \mathbf{q} \tag{7}$$

   Taking the dot product of both sides of 7 with themselves and using 5 to eliminate $\mathbf{q}$ a quadratic equation in μ is obtained which may be written as:

   $$a\mu^2 + 2b\mu + c = 0$$

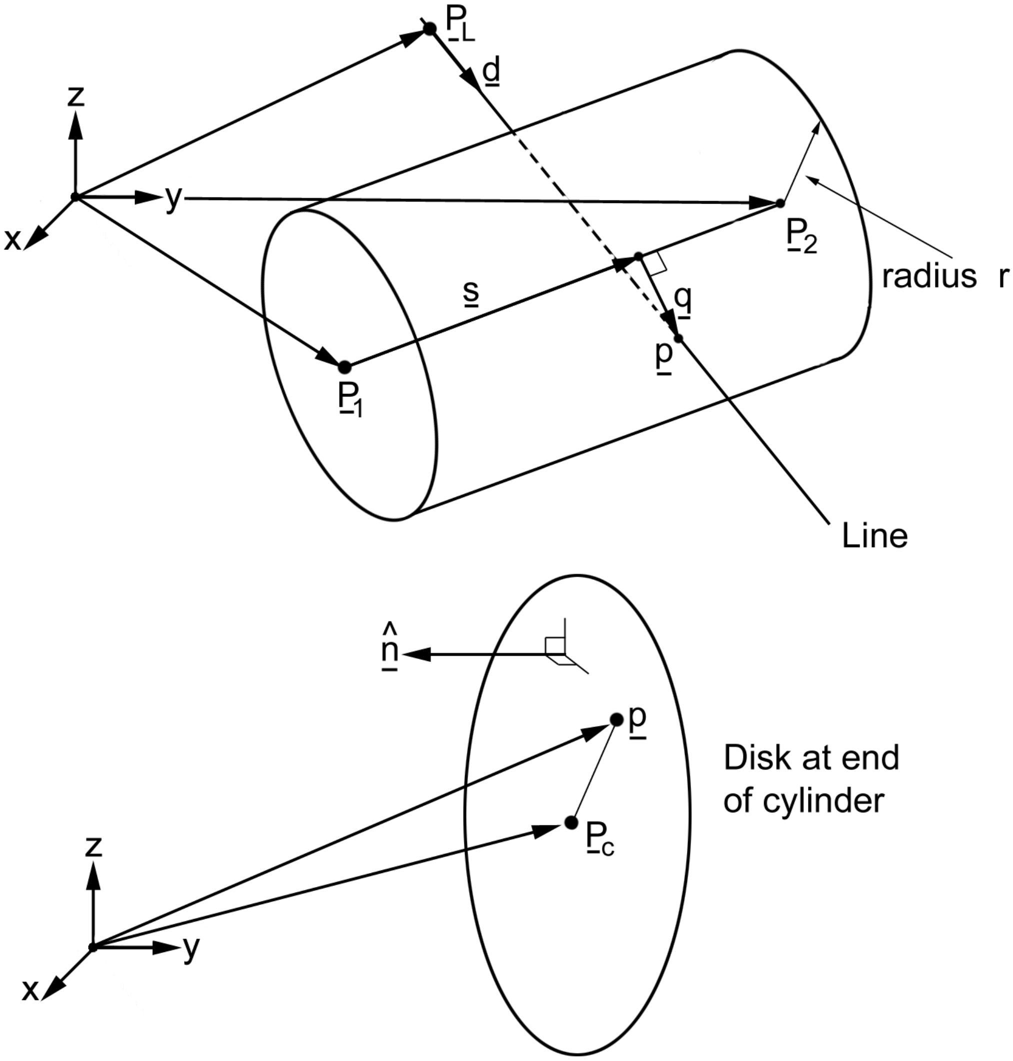

Figure 3: Geometry of an intersection between a line and a cylinder.

where (with $\mathbf{t} = (\mathbf{P}_l - \mathbf{P}_1)$, for clarity):

$$\begin{aligned}
a &= (\mathbf{d} - \left(\frac{\mathbf{d}\cdot\mathbf{s}}{\mathbf{s}\cdot\mathbf{s}}\right)\mathbf{s}) \cdot (\mathbf{d} - \left(\frac{\mathbf{d}\cdot\mathbf{s}}{\mathbf{s}\cdot\mathbf{s}}\right)\mathbf{s}) \\
b &= (\mathbf{d} - \left(\frac{\mathbf{d}\cdot\mathbf{s}}{\mathbf{s}\cdot\mathbf{s}}\right)\mathbf{s}) \cdot (\mathbf{t} - \left(\frac{\mathbf{t}\cdot\mathbf{s}}{\mathbf{s}\cdot\mathbf{s}}\right)\mathbf{s}) \\
c &= (\mathbf{t} - \left(\frac{\mathbf{t}\cdot\mathbf{s}}{\mathbf{s}\cdot\mathbf{s}}\right)\mathbf{s}) \cdot (\mathbf{t} - \left(\frac{\mathbf{t}\cdot\mathbf{s}}{\mathbf{s}\cdot\mathbf{s}}\right)\mathbf{s}) - r^2
\end{aligned}$$

No real roots to this quadratic implies that the line does not intersect the infinitely long cylinder and we need proceed no further.

2. If there is at least one real root we must investigate further to determine if the intersection occurs between the points $\mathbf{P}_1$ and $\mathbf{P}_2$. For this we need values for λ corresponding to the root(s)of the quadratic in μ. Starting with the smaller root ($\mu = \mu_1$) a λ is calculated and if $0 \le \lambda \le 1$ there is an intersection on the outside surface of the cylinder at the point given by $\mathbf{P}_l + \mu_1\mathbf{d}$. If this is not the case a λ corresponding to the larger root($\mu = \mu_2$) is determined and if $0 \le \lambda \le 1$ there is an intersection on the inside surface.

3. If the point of intersection is on the inside surface of the cylinder it will be necessary to check the intersection with a pair of disks, radius r, and surface normal in the direction $\mathbf{P}_2 - \mathbf{P}_1$. One of the disks contains the point $\mathbf{P}_2$ and the other the point $\mathbf{P}_1$.

   To obtain the point of intersection $\mathbf{P}_i$ with the disks use the algorithm described in chapter 1. And if the intersection is within the disk $(\mathbf{P}_i - \mathbf{P}_c) \cdot (\mathbf{P}_i - \mathbf{P}_c) < r^2$.

A complete algorithm for the intersection of a line and a cylinder is given in two steps, shown in Figures 4 and 5

0.1.3 Intersection between a line and a regular cone

The argument used in determining the intersection of a line and a cylinder may be developed further to enable the points of intersection between a line and cone to be obtained. As in the case of a sphere and a cylinder there may be either none, one or two intersection points. Figure 6 illustrates the location of the point of intersection $\mathbf{P}_i$ between a line specified by:

$$\mathbf{p} = \mathbf{P}_0 + \mu\hat{\mathbf{d}}$$

and a cone that is defined by the points $\mathbf{P}_1$, the apex of the cone, $\mathbf{P}_2$ at the base of the cone and the half angle θ of the cone. To determine the point of intersection $\mathbf{P}_i$, a four step algorithm, is as follows:

1. For the infinite double cone, with axis AA', determine any points of intersection with the line $\mathbf{P}_0 + \mu\hat{\mathbf{d}}$ and their location.
2. For points of intersection determine which, if any, intersect the half cone between $\mathbf{P_1}$ and $\mathbf{P_2}$, reject any other intersections.
3. Determine whether and where the line intersects a disk centered on $\mathbf{P_2}$ and perpendicular to $\mathbf{P_2} - \mathbf{P_1}$
4. Depth sort the intersections of cone and disk to make the final choice for $\mathbf{P}_i$

   Note:
   An important application of this calculation occurs when rendering atmospheric effects associated with spotlights, for example dust scattering, volume shadowing and volumetric fog.

Step 1: Calculate the intersection point or points

$$\mathbf{s} = (\mathbf{P}_2 - \mathbf{P}_1)$$
$$\mathbf{P}_{l1} = (\mathbf{P}_l - \mathbf{P}_1)$$
$$\mathbf{A} = \mathbf{d} - \left(\frac{\mathbf{d}\cdot\mathbf{s}}{\mathbf{s}\cdot\mathbf{s}}\right)\mathbf{s}$$
$$\mathbf{B} = \mathbf{P}_{l1} - \left(\frac{\mathbf{P}_{l1}\cdot\mathbf{s}}{\mathbf{s}\cdot\mathbf{s}}\right)\mathbf{s}$$
$$a = \mathbf{A}\cdot\mathbf{A}$$
$$b = 2\mathbf{A}\cdot\mathbf{B}$$
$$c = \mathbf{B}\cdot\mathbf{B} - r^2$$
$$b_4 = b^2 - 4ac$$

if $b_4 < 0$ no intersection points
else {
  if $b_4 < \epsilon$ {
    thus one intersection point, so set
    $\mu_1 = \mu_2 = \dfrac{-b}{2a}$
  }
  else {
    two intersection points determined by
    $b_4 = \sqrt{b_4}$
    $\mu_1 = \dfrac{(-b + b_4)}{2a}$
    $\mu_2 = \dfrac{(-b - b_4)}{2a}$
  }
  $\mathbf{p}_1 = \mathbf{p}_l + \mu_1\mathbf{d}$
  $\mathbf{p}_2 = \mathbf{p}_l + \mu_2\mathbf{d}$
}

Figure 4: Step 1 in determining the intersection between a line and a cylinder, the geometry of the problem is shown in Figure 3.

Step 2: Consider the values μ_1 and μ_2 in turn

$\mu = \max(\mu_1, \mu_2)$
$\lambda = \dfrac{\mathbf{P}_{l1}\cdot\mathbf{s} + \mu\mathbf{d}\cdot\mathbf{s}}{\mathbf{s}\cdot\mathbf{s}}$
if $0 \le \lambda \le 1$ {
  return intersection on front face
}
$\mu = \min(\mu_1, \mu_2)$
$\lambda = \dfrac{\mathbf{P}_{l1}\cdot\mathbf{s} + \mu\mathbf{d}\cdot\mathbf{s}}{\mathbf{s}\cdot\mathbf{s}}$
if $0 \le \lambda \le 1$ {
  intersection on back face
  check disks at ends of cylinders
}

Figure 5: Step 2 in determining the intersection between a line and a cylinder.

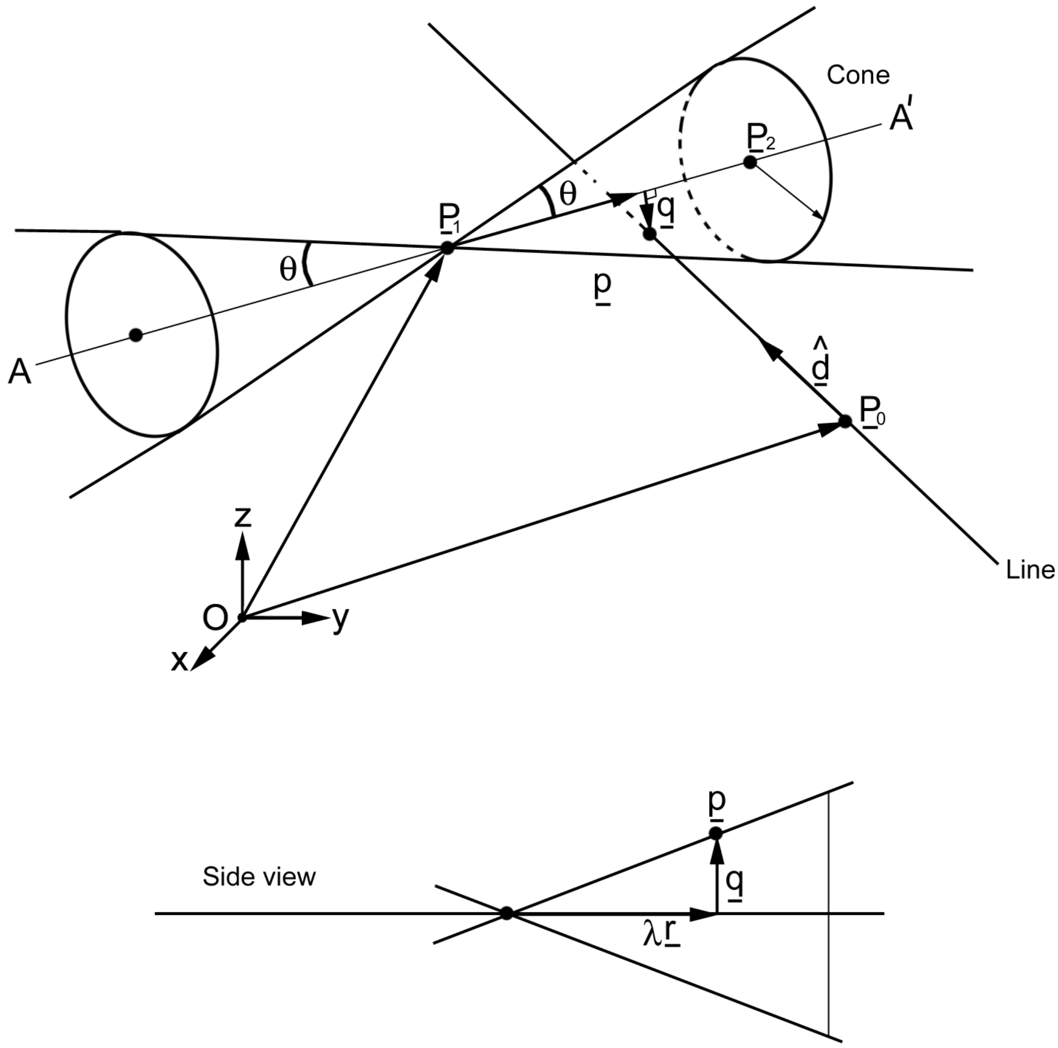

Figure 6: Geometry for determining the intersection between a line and a cone.

One point worth making in connection with these scenarios is that the line might originate inside the cone and this will necessitate some other special tests to determine whether the line is passing from outside to inside or inside to outside at $\mathbf{P}_i$

For completeness the algebra of the first step is now discussed. Consider the intersection of a line with the double cone as illustrated in Figure 6. The cone extends to infinity in both directions. The key step in the procedure is to equate two different paths from the coordinate origin O to $\mathbf{P}_i$

First obtain the unit vector $\hat{\mathbf{r}}$:

$$\hat{\mathbf{r}} = \frac{\mathbf{P}_2 - \mathbf{P}_1}{|\mathbf{P}_2 - \mathbf{P}_1|}$$

Using two alternate paths $\mathbf{P}_i$ may be expressed as:

$$\begin{aligned} \mathbf{P}_i &= \mathbf{P}_0 + \mu\hat{\mathbf{d}} \text{ or} \\ \mathbf{P}_i &= \mathbf{P}_1 + \lambda\hat{\mathbf{r}} + \mathbf{q} \end{aligned}$$

for appropriate λ and μ. These can be equated:

$$\mathbf{P}_0 + \mu\hat{\mathbf{d}} = \mathbf{P}_1 + \lambda\hat{\mathbf{r}} + \mathbf{q} \tag{8}$$

The vector $\mathbf{q}$ is directed radially from the axis of the cone thus $\mathbf{q} \cdot \hat{\mathbf{r}} = 0$. The vector $\lambda\hat{\mathbf{r}}$ joins $\mathbf{P}_1$ to the base of $\mathbf{q}$ The length of $\mathbf{q}$ is $l = \lambda \tan\theta$ and since $l = |\mathbf{q}|$ we can write: $\mathbf{q} \cdot \mathbf{q} = (\lambda \tan\theta)^2$

To determine λ from equation 8 take the dot product of both sides with $\hat{\mathbf{r}}$:

$$\lambda = \mu(\hat{\mathbf{d}} \cdot \hat{\mathbf{r}}) + (\mathbf{P}_0 - \mathbf{P}_1) \cdot \hat{\mathbf{r}} \tag{9}$$

Substituting 9 in 8 eliminates λ and therefore:

$$\mu\hat{\mathbf{d}} + (\mathbf{P}_0 - \mathbf{P}_1) - (\mu(\hat{\mathbf{d}} \cdot \hat{\mathbf{r}}) + (\mathbf{P}_0 - \mathbf{P}_1) \cdot \hat{\mathbf{r}})\hat{\mathbf{r}} = \mathbf{q} \tag{10}$$

Writing 10as a polynomial in μ:

$$\mathbf{A}\mu + \mathbf{B} = \mathbf{q} \tag{11}$$

with:

$$\begin{aligned} \mathbf{A} &= \hat{\mathbf{d}} - (\hat{\mathbf{d}} \cdot \hat{\mathbf{r}})\hat{\mathbf{r}} \\ \mathbf{B} &= (\mathbf{P}_0 - \mathbf{P}_1) - ((\mathbf{P}_0 - \mathbf{P}_1) \cdot \hat{\mathbf{r}})\hat{\mathbf{r}} \end{aligned}$$

taking the dot product of each side of 11 with itself gives a quadratic equation in μ:

$$(\mathbf{A} \cdot \mathbf{A})\mu^2 + 2(\mathbf{A} \cdot \mathbf{B})\mu + (\mathbf{B} \cdot \mathbf{B}) = \mathbf{q} \cdot \mathbf{q} = (\lambda \tan\theta)^2 \tag{12}$$

Substituting λ from 9 and rearranging gives a quadratic with μ as the only unknown:

$$a\mu^2 + 2b\mu + c = 0 \tag{13}$$

where:

$$\begin{aligned} a &= (\mathbf{A} \cdot \mathbf{A}) - (\hat{\mathbf{d}} \cdot \hat{\mathbf{r}})^2 \tan^2\theta \\ b &= (\mathbf{A} \cdot \mathbf{B}) - (\hat{\mathbf{d}} \cdot \hat{\mathbf{r}})((\mathbf{P}_0 - \mathbf{P}_1) \cdot \hat{\mathbf{r}}) \tan^2\theta \\ c &= (\mathbf{B} \cdot \mathbf{B}) - ((\mathbf{P}_0 - \mathbf{P}_1) \cdot \hat{\mathbf{r}})^2 \tan^2\theta \end{aligned}$$

Once the values of μ satisfying 13 are known a similar argument to that outlined in Step 2 of section 0.1.2 may be used to complete the determination of the intersection points $\mathbf{P}_i$.

0.2 Some other useful non geometric mathematical results

Chapter 1 dealt primarily with geometrical results necessary for developing algorithms used in 3D computer graphics. There are a few other mathematical topics that have utility when developing these algorithms and they are dealt with in this section.

0.2.1 Random number generation

Computer languages, particularly **C** and **C++** normally have quite poor random number generation. They are limited to returning pseudo random integers in the range $0 - 32767$. Some computers now offer the `drand48()` function, which is good but, for systems that don't offer it the short function below which generates a pseudo random double in the interval $[0, 1]$.

```
long IX=9123,IY=8844,IZ=20846; /* initial state */

void dSeedRandom(long i){
 /* use internal integer random number generator to seed the algorithm */
 srand(i); IX=rand(); IY=rand(); IZ=rand();
}

double dRandom(void){
 IX=imod(171*IX,30269);
 IY=imod(172*IY,20207);
 IZ=imod(170*IZ,30323);
 return (double)fmod((double)IX/30269.0+
                     (double)IY/30307.0+
                     (double)IZ/30323.0,1.0);
}
```

0.2.2 Solving a set of n simultaneous equations

A mathematical task that occurs quite often in computer graphics is finding the solution to a set of n linear simultaneous equations. Gauss elimination and a host of other techniques have been developed to perform this task efficiently. Many excellent algorithms have been published on this subject,. In general the problem can be stated in matrix form as $[A][x] = [y]$, the matrix $[A]$ is of dimension $n \times n$ and the matrices $[x]$ and $[y]$ are of size $n \times 1$. The matrix $[x]$ is the unknown. One method of solution is to obtain the inverse of $[A]$ and write $[x] = [A]^{-1}[y]$. This approach is very inefficient. However, sometimes the equations show a particular structure when shortcuts may be taken.

For example the problem of finding the parameters for the cubic spline curves gives rise to a set of linear equations where many elements of the $[A]$ matrix are zero. In fact all the non-zero elements lie on the three central diagonals of $[A]$. Matrices with this special property are solved very rapidly and the algorithm for their solution is presented here.

Given the known matrix A of dimension $n \times n$, the known vector y of dimension $n\ times 1$ and the unknown vector of dimension $n \times 1$ satisfying $[A][x] = [y]$ the problem may be expressed as:

$$\begin{bmatrix} b_0 & c_0 & 0 & 0 & .. & 0 \\ a_1 & b_1 & c_1 & 0 & .. & 0 \\ 0 & a_2 & b_2 & c_2 & & \\ .. & .. & .. & .. & .. & .. \\ 0 & .. & 0 & a_{n-2} & b_{n-2} & c_{n-2} \\ 0 & .. & 0 & 0 & a_{n-1} & b_{n-1} \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ .. \\ .. \\ x_{n-1} \end{bmatrix} = \begin{bmatrix} y_0 \\ y_1 \\ y_2 \\ .. \\ .. \\ y_{n-1} \end{bmatrix}$$

where a_i, b_i and c_i are the non-zero entries of $[A]$. To solve for the x_i the following efficient two step algorithm is used:

1. Work through all the i from 1 to $n-1$ replacing the b_i with:

$$b_i - \frac{a_i}{b_{i-1}} c_{i-1}$$

   At the same time replace the y_i with:

$$y_i - \frac{a_i}{b_{i-1}} y_{i-1}$$

2. Obtain the results, first set :

$$x_{n-1} = \frac{y_{n-1}}{b_{n-1}}$$

   Then work through all the i from $n-2$ back to 0 to calculate the remaining x_i:

$$x_i = \frac{y_i - c_i y_{i+1}}{b_i}$$

The execution time of this algorithm is near linear with respect to n.