I
Introductory Mathematics

Do not worry too much about your difficulties in mathematics, I can assure you that mine are still greater.

— Albert Einstein.

I.1 Useful Sums

1. The sum of a geometric series is

$$\sum_{i=0}^{n} a^i = \begin{cases} 0, & \text{if } a = 0, \\ n+1, & \text{if } a = 1, \\ \frac{1-a^{n+1}}{1-a}, & \text{otherwise.} \end{cases} \tag{I.1}$$

A simple corollary is

$$\sum_{i=0}^{\infty} a^i = \frac{1}{1-a} \quad \text{for } |a| < 1. \tag{I.2}$$

Differentiating Equation (I.2) yields

$$\sum_{i=0}^{\infty} ia^i = \frac{a}{(1-a)^2} \quad \text{for } |a| < 1.$$

2. The binomial theorem

$$(a+b)^n = \sum_{i=0}^{n} \binom{n}{i} a^i b^{n-i}.$$

3. Series expansion of an exponential (Taylor series)

$$e^x = \sum_{i=0}^{\infty} \frac{x^i}{i!}.$$

4. The sum of the first n integers

$$\sum_{i=1}^{n} i = \frac{n(n+1)}{2}.$$

5. The sum of the first n integers squared

$$\sum_{i=1}^{n} i^2 = \frac{n(n+1)(2n+1)}{6}.$$

6. Sum of powers of 2

$$\sum_{i=0}^{n} 2^i = 2^0 + 2^1 + 2^2 + \cdots + 2^n = 2^{n+1} - 1.$$

I.2 Matrices

A matrix $\mathbf{T}$ is a rectangular array of numbers, where each element a_{ij} is identified by its row and column. Matrix $\mathbf{T}_1$ below is "generic," with m rows and n columns. Notice how elements a_{ii} constitute the main diagonal of the matrix. Matrix $\mathbf{T}_2$ is diagonal ($a_{ij} = 0$ for $i \neq j$), matrix $\mathbf{T}_3$, is symmetric ($a_{ij} = a_{ji}$), and $\mathbf{T}_4$ is an identity matrix.

$$\mathbf{T}_1 = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \dots & a_{mn} \end{pmatrix}, \quad \mathbf{T}_2 = \begin{pmatrix} a_{11} & 0 & 0 & 0 \\ 0 & a_{22} & 0 & 0 \\ 0 & 0 & a_{33} & 0 \\ 0 & 0 & 0 & a_{44} \end{pmatrix},$$

$$\mathbf{T}_3 = \begin{pmatrix} 33 & -17 & 201 & -5 \\ -17 & 66 & 26 & -68 \\ 201 & 26 & 21 & -9 \\ -5 & -68 & -9 & 0 \end{pmatrix}, \quad \mathbf{T}_4 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}.$$

The transpose of matrix $\mathbf{A}$ (denoted by $\mathbf{A}^T$) is obtained from $\mathbf{A}$ by reflecting all the elements with respect to the main diagonal. A symmetric matrix equals its transpose.

> All problems in computer graphics can be solved with a matrix inversion.
>
> James F. Blinn, 1993.

I.2.1 Matrix Operations

The rule for matrix addition/subtraction is $c_{ij} = a_{ij} \pm b_{ij}$, where $\mathbf{C} = \mathbf{A} \pm \mathbf{B}$. The rule for matrix multiplication is slightly more complex: $c_{ij} = \sum_{k=1}^{n} a_{ik} b_{kj}$. Each element of $\mathbf{C}$ is the *dot product* of a row of $\mathbf{A}$ and a column of $\mathbf{B}$. In the dot product, corresponding elements from $\mathbf{A}$ and $\mathbf{B}$ are multiplied, and the products summed. In order for the multiplication to be well defined, each row of $\mathbf{A}$ must have the same size as a column of $\mathbf{B}$. Matrices $\mathbf{A}$ and $\mathbf{B}$ can therefore be multiplied only if the number of columns of $\mathbf{A}$ equals the number of rows of $\mathbf{B}$. Note that matrix multiplication is not commutative, i.e., $\mathbf{AB} \neq \mathbf{BA}$ in general.

An example of matrix multiplication is the product of the 1×3 and 3×1 matrices

$$(1, -1, 5) \begin{pmatrix} 4 \\ -2 \\ 3 \end{pmatrix},$$

which yields the 1×1 matrix 21.

Tensor products. This is a special case of matrix multiplication. If $\mathbf{A}$ is a column vector and $\mathbf{B}$ is a row vector (each with n elements), then their tensor product $\mathbf{C}$ is defined by $\mathbf{C}_{ij} = \mathbf{A}_i \mathbf{B}_j$. Example:

$$\begin{pmatrix} 4 \\ -2 \\ 3 \end{pmatrix} (1, -1, 5) = \begin{pmatrix} 4 & -4 & 20 \\ -2 & 2 & -10 \\ 3 & -3 & 15 \end{pmatrix}.$$

A square matrix has a determinant, denoted by either "det $\mathbf{A}$" or $|\mathbf{A}|$, that is a number. The determinant of the 2×2 matrix $\left(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right)$ is defined as $ad - bc$. The determinant of a larger matrix can be calculated by the rule (note the alternating signs):

$$\begin{vmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{vmatrix} = a_{11} \begin{vmatrix} a_{22} & a_{23} \\ a_{32} & a_{33} \end{vmatrix} - a_{12} \begin{vmatrix} a_{21} & a_{23} \\ a_{31} & a_{33} \end{vmatrix} + a_{13} \begin{vmatrix} a_{21} & a_{22} \\ a_{31} & a_{32} \end{vmatrix}.$$

Matrix division is not defined, but certain matrices have an *inverse*. The inverse of $\mathbf{A}$ is denoted $\mathbf{A}^{-\mathbf{1}}$, and has the property that $\mathbf{A}\mathbf{A}^{-1} = \mathbf{A}^{-1}\mathbf{A} = \mathbf{I}$, where $\mathbf{I}$ is the *identity matrix* (with ones in the diagonal and zeros elsewhere). The inverse of a matrix is used, e.g., to solve systems of linear algebraic equations. Such a system can be denoted $\mathbf{Ax} = \mathbf{b}$ where $\mathbf{A}$ is the matrix of coefficients, $\mathbf{x}$ is the column of unknowns, and $\mathbf{b}$ is the column of the right-hand side coefficients. The solution is $\mathbf{x} = \mathbf{A}^{-1}\mathbf{b}$.

Example: The following system of three equations with three unknowns x, y, and z

$$\begin{aligned} x - y &= 1, \\ -x + y &= 2, \\ 25x + 2y + z &= 3, \end{aligned} \tag{I.3}$$

can be written

$$\begin{pmatrix}1 & -1 & 0\\ -1 & 1 & 0\\ 25 & 2 & 1\end{pmatrix}\begin{pmatrix}x\\ y\\ z\end{pmatrix}=\begin{pmatrix}1\\ 2\\ 3\end{pmatrix}.$$

The inverse of the 3×3 transformation matrix (used in Section 4.32.1)

$$\mathbf{T}=\begin{pmatrix}a & b & 0\\ c & d & 0\\ m & n & 1\end{pmatrix}\quad\text{is}\quad\mathbf{T}^{-1}=\frac{1}{ad-bc}\begin{pmatrix}d & -b & 0\\ -c & a & 0\\ cn-dm & bm-an & 1\end{pmatrix}. \tag{I.4}$$

In general, however, the calculation of the inverse is not trivial and can be found in any text on Linear Algebra, and also in [Press et al. 88]. Page 227 has an interesting example of the inverse of a matrix.

Here is a summary of the properties of matrix operations:

$$\begin{gathered}\mathbf{A}+\mathbf{B}=\mathbf{B}+\mathbf{A},\quad \mathbf{A}+(\mathbf{B}+\mathbf{C})=(\mathbf{A}+\mathbf{B})+\mathbf{C},\\ k(\mathbf{A}+\mathbf{B})=k\mathbf{A}+k\mathbf{B},\quad (k+m)\mathbf{A}=k\mathbf{A}+m\mathbf{A},\quad k(m\mathbf{A})=(km)\mathbf{A}=m(k\mathbf{A}),\\ \mathbf{A}(\mathbf{BC})=(\mathbf{AB})\mathbf{C},\quad \mathbf{A}(\mathbf{B}+\mathbf{C})=\mathbf{AB}+\mathbf{AC},\\ (\mathbf{A}+\mathbf{B})\mathbf{C}=\mathbf{AB}+\mathbf{AC},\quad \mathbf{A}(k\mathbf{B})=k(\mathbf{AB})=(k\mathbf{A})\mathbf{B},\\ (\mathbf{A}+\mathbf{B})^T=\mathbf{A}^T+\mathbf{B}^T,\quad (k\mathbf{A})^T=k^T\mathbf{A}^T,\quad (\mathbf{AB})^T=\mathbf{B}^T\mathbf{A}^T.\end{gathered}$$

Information on the history of matrices and determinants can be found at URL `http://www-groups.dcs.st-and.ac.uk/~history/HistTopics/`, file `Matrices_and_determinants.html`.

◇ **Exercise I.1:** Add, subtract, and multiply the two matrices

$$\mathbf{A}=\begin{pmatrix}1 & 2 & 3\\ 4 & 5 & 6\\ 7 & 8 & 9\end{pmatrix},\qquad \mathbf{B}=\begin{pmatrix}7 & 8 & 9\\ 4 & 5 & 6\\ 1 & 2 & 3\end{pmatrix}.$$

◇ **Exercise I.2:** Calculate the inverse of

$$\mathbf{T}=\begin{pmatrix}1 & -1 & 0\\ -1 & 1 & 0\\ 25 & 2 & 1\end{pmatrix}.$$

A matrix is *orthogonal* if the dot product of any two different rows is zero (and the same for columns. A matrix is *orthonormal* if it is orthogonal and the dot product of a row with itself is one (and the same for columns). Imagine a square matrix $\mathbf{A}$. When it is transposed, its rows and columns change roles. A general element (i,j) of the product $\mathbf{AA}^T$ is thus the dot product of row i of $\mathbf{A}$ and row j of the same $\mathbf{A}$. Therefore, if $\mathbf{A}$ is orthonormal, then $\mathbf{AA}^T$ is the identity matrix $\mathbf{I}$. However, the product $\mathbf{BB}^{-1}$ for any matrix $\mathbf{B}$ is $\mathbf{I}$ (if $\mathbf{B}$ has an inverse), so we conclude that the transpose $\mathbf{A}^T$ of an orthonormal matrix $\mathbf{A}$ equals its inverse $\mathbf{A}^{-1}$.

The opposite is also true. If $\mathbf{A}^T = \mathbf{A}^{-1}$ for some matrix $\mathbf{A}$, then $\mathbf{A}$ is orthonormal. It can be shown that an orthonormal matrix is always a rotation matrix, and that any rotation matrix [Equation (4.48)] is orthonormal.

Eigenvalues and eigenvectors (from the German word for "own") are useful mathematical quantities associated with matrices. They are defined as follows: If $\mathbf{A}$ is an $n\times n$ matrix and if there exist vectors $\mathbf{x}$ and scalars λ such that $\mathbf{Ax} = \lambda\mathbf{x}$ [or $(\mathbf{A} - \lambda I)\mathbf{x} = 0$], then λ is called an *eigenvalue* of $\mathbf{A}$, and $\mathbf{x}$ is the *eigenvector* associated with λ. The eigenvectors of a symmetric matrix are orthogonal.

In principle, calculating the eigenvalues of an $n\times n$ matrix involves solving an nth-degree polynomial equation. Therefore, for $n \geq 5$, the results cannot in general be expressed purely in terms of explicit radicals. Even for the simple matrix

$$\begin{pmatrix} a & b \\ -b & 2a \end{pmatrix},$$

the eigenvalues have the two complicated expressions

$$\frac{1}{2}\left(3a - \sqrt{a^2 - 4b^2}\right), \text{ and } \frac{1}{2}\left(3a + \sqrt{a^2 - 4b^2}\right).$$

This is why mathematical software is used in practice to obtain approximate values (real and complex) of the eigenvalues and eigenvectors of a given matrix.

Eigenvalues and eigenvectors are mentioned in Section 4.4.8.

Bibliography

Press, W. H., B. P. Flannery, et al. (1988) *Numerical Recipes in C: The Art of Scientific Computing*, Cambridge University Press.
(Also available on-line from `http://www.nr.com/`.)

> We [he and Halmos] share a philosophy about linear algebra: we think basis-free, we write basis-free, but when the chips are down we close the office door and compute with matrices like fury.
>
> Irving Kaplansky.

I.3 Trigonometric Identities

Many of the identities listed here can be derived with the help of DeMoivre's theorem [Equation (Ans.3)].

Basic Identities

$$\tan\alpha = \frac{\sin\alpha}{\cos\alpha}, \quad \cot\alpha = \frac{\cos\alpha}{\sin\alpha} = \frac{1}{\tan\alpha}, \quad \csc\alpha = \frac{1}{\sin\alpha}, \quad \sec\alpha = \frac{1}{\cos\alpha}.$$

$$\sin(-\alpha) = -\sin\alpha, \quad \cos(-\alpha) = \cos\alpha, \quad \tan(-\alpha) = -\tan\alpha.$$

$$\sin^2\alpha + \cos^2\alpha = 1, \quad \tan^2\alpha + 1 = \sec^2\alpha, \quad \cot^2\alpha + 1 = \csc^2\alpha.$$

Sum and Difference Identities

$$\cos(\alpha \pm \beta) = \cos\alpha\cos\beta \mp \sin\alpha\sin\beta, \quad \sin(\alpha \pm \beta) = \sin\alpha\cos\beta \pm \cos\alpha\sin\beta,$$

$$\tan(\alpha \pm \beta) = \frac{\tan\alpha \pm \tan\beta}{1 \mp \tan\alpha\tan\beta}.$$

Cofunction Identities

$$\sin(\pi/2 - \alpha) = \cos\alpha, \quad \cos(\pi/2 - \alpha) = \sin\alpha, \quad \tan(\pi/2 - \alpha) = \cot\alpha.$$

Multiple Angle and Half Angle Identities

$$\cos 2\alpha = \cos^2\alpha - \sin^2\alpha = 1 - 2\sin^2\alpha = 2\cos^2\alpha - 1, \quad \sin 2\alpha = 2\sin\alpha\cos\alpha,$$

$$\tan 2\alpha = \frac{2\tan\alpha}{1 - \tan^2\alpha}.$$

$$\cos(\alpha/2) = \pm\sqrt{(1 + \cos\alpha)/2}, \quad \sin(\alpha/2) = \pm\sqrt{(1 + \cos\alpha)/2}.$$

$$\tan(\alpha/2) = \pm\sqrt{\frac{1 - \cos\alpha}{1 + \cos\alpha}} = \frac{\sin\alpha}{1 + \cos\alpha} = \frac{1 - \cos\alpha}{\sin\alpha}.$$

Sum and Product Identities

$$\sin\alpha + \sin\beta = 2\sin\left(\frac{\alpha+\beta}{2}\right)\cos\left(\frac{\alpha-\beta}{2}\right),$$

$$\sin\alpha - \sin\beta = 2\cos\left(\frac{\alpha+\beta}{2}\right)\sin\left(\frac{\alpha-\beta}{2}\right).$$

$$\cos\alpha + \cos\beta = 2\cos\left(\frac{\alpha+\beta}{2}\right)\cos\left(\frac{\alpha-\beta}{2}\right),$$

$$\cos\alpha - \cos\beta = -2\sin\left(\frac{\alpha+\beta}{2}\right)\sin\left(\frac{\alpha-\beta}{2}\right).$$

$$\sin\alpha\cos\beta = \frac{1}{2}\left[\sin(\alpha+\beta) + \sin(\alpha-\beta)\right],$$

$$\cos\alpha\sin\beta = \frac{1}{2}\left[\sin(\alpha+\beta) - \sin(\alpha-\beta)\right].$$

$$\cos\alpha\cos\beta = \frac{1}{2}\left[\cos(\alpha+\beta) + \cos(\alpha-\beta)\right],$$

$$\sin\alpha\sin\beta = -\frac{1}{2}\left[\cos(\alpha+\beta) - \cos(\alpha-\beta)\right].$$

Note that the line above also implies

$$\cos^2\alpha = \frac{1}{2}\left(\cos(2\alpha) + 1\right), \quad \sin^2\alpha = \frac{1}{2}\left(1 - \cos(2\alpha)\right).$$

Laws of Sines and Cosines: any triangle with sides a, b, c and angles α, β, γ satisfies the law of sines $a/\sin\alpha = b/\sin\beta = c/\sin\gamma$ and the law of cosines

$$a^2 = b^2 + c^2 - 2bc\,\cos\alpha, \quad b^2 = a^2 + c^2 - 2ac\,\cos\beta, \quad c^2 = a^2 + b^2 - 2ab\,\cos\gamma.$$

Mathematics is the only universal language there is, senator!
— Jodie Foster (as Ellie Arroway) in *Contact* (1977).

I.4 Vector Algebra

A vector is a mathematical entity with two attributes, direction and magnitude (notice that a vector has no spatial position). The magnitude of vector $\mathbf{P} = (x, y, z)$ (also called its *absolute value*) is $|\mathbf{P}| = \sqrt{x^2 + y^2 + z^2}$. The direction of a vector can be expressed by the cosines of the angles it makes with the coordinate axes $x/|\mathbf{P}|$, $y/|\mathbf{P}|$, and $z/|\mathbf{P}|$. Note that the vector $(x/|\mathbf{P}|, y/|\mathbf{P}|, z/|\mathbf{P}|)$ has a magnitude of 1 (it is a *unit vector*).

The three unit vectors in the directions of the coordinate axes are traditionally denoted $\mathbf{i} = (1, 0, 0)$, $\mathbf{j} = (0, 1, 0)$, and $\mathbf{k} = (0, 0, 1)$.

I.4.1 Operations on Vectors

Vector addition is defined by adding the individual elements of the vectors being added. Thus, $\mathbf{P} + \mathbf{Q} = (P_x, P_y, P_z) + (Q_x, Q_y, Q_z) = (P_x + Q_x, P_y + Q_y, P_z + Q_z)$. This operation is both commutative ($\mathbf{P}+\mathbf{Q} = \mathbf{Q}+\mathbf{P}$) and associative $\mathbf{P}+(\mathbf{Q}+\mathbf{T}) = (\mathbf{P} + \mathbf{Q}) + \mathbf{T}$. Subtraction of vectors $\mathbf{P} - \mathbf{Q}$ is done similarly and results in the vector from $\mathbf{Q}$ to $\mathbf{P}$.

Vectors can be multiplied in three different ways:

1. The multiplication of a scalar by a vector is defined by $\alpha\mathbf{P} = (\alpha x, \alpha y, \alpha z)$. It changes the magnitude of the vector (by a factor α), but not its direction. This operation is distributive with respect to vector addition or subtraction, $\alpha(\mathbf{P} \pm \mathbf{Q}) = \alpha\mathbf{P} \pm \alpha\mathbf{Q}$.

2. The dot product of two vectors is denoted by $\mathbf{P} \bullet \mathbf{Q}$ and is defined as the scalar

$$(P_x, P_y, P_z)(Q_x, Q_y, Q_z)^T = \mathbf{P}\mathbf{Q}^T = P_xQ_x + P_yQ_y + P_zQ_z.$$

This also equals $|\mathbf{P}|\,|\mathbf{Q}|\cos\theta$, where θ is the angle between the vectors. The dot product of perpendicular vectors (also called *orthogonal vectors*) is thus zero. The dot product is commutative, $\mathbf{P} \bullet \mathbf{Q} = \mathbf{Q} \bullet \mathbf{P}$ and is also distributive with respect to vector addition or subtraction $\mathbf{P} \bullet (\mathbf{Q} \pm \mathbf{T}) = \mathbf{P} \bullet \mathbf{Q} \pm \mathbf{P} \bullet \mathbf{T}$.

The triple product $(\mathbf{P} \bullet \mathbf{Q})\mathbf{R}$ is sometimes useful. It can be represented as

$$\begin{aligned}
&(\mathbf{P} \bullet \mathbf{Q})\mathbf{R}\\
&= (P_xQ_x + P_yQ_y + P_zQ_z)(R_x, R_y, R_z)\\
&= \big((P_xQ_x + P_yQ_y + P_zQ_z)R_x, (P_xQ_x + P_yQ_y + P_zQ_z)R_y, (P_xQ_x + P_yQ_y + P_zQ_z)\big)R_z\\
&= (Q_x, Q_y, Q_z)\begin{pmatrix} P_xR_x & P_yR_x & P_zR_x\\ P_xR_y & P_yR_y & P_zR_y\\ P_xR_z & P_yR_z & P_zR_z \end{pmatrix}\\
&= \mathbf{Q}(\mathbf{PR}), \qquad (I.5)
\end{aligned}$$

where the notation $(\mathbf{PR})$ stands for the 3×3 matrix above.

3. The cross product of two vectors (also called the *vector product*) is denoted $\mathbf{P}\times\mathbf{Q}$ and is defined as the vector

$$(P_2Q_3 - P_3Q_2, -P_1Q_3 + P_3Q_1, P_1Q_2 - P_2Q_1). \tag{I.6}$$

It is easy to show that $\mathbf{P}\times\mathbf{Q}$ is perpendicular to both $\mathbf{P}$ and $\mathbf{Q}$.

◇ **Exercise I.3:** Prove it!

The following expressions show how $\mathbf{P}\times\mathbf{Q}$ can be expressed by means of a determinant.

$$\begin{aligned}\mathbf{P}\times\mathbf{Q} &= \begin{vmatrix} \mathbf{i} & \mathbf{j} & \mathbf{k} \\ P_1 & P_2 & P_3 \\ Q_1 & Q_2 & Q_3 \end{vmatrix} = \mathbf{i}\begin{vmatrix} P_2 & P_3 \\ Q_2 & Q_3 \end{vmatrix} - \mathbf{j}\begin{vmatrix} P_1 & P_3 \\ Q_1 & Q_3 \end{vmatrix} + \mathbf{k}\begin{vmatrix} P_1 & P_2 \\ Q_1 & Q_2 \end{vmatrix} \\ &= (P_2Q_3 - P_3Q_2, -P_1Q_3 + P_3Q_1, P_1Q_2 - P_2Q_1),\end{aligned}$$

or, alternatively, by means of a matrix

$$= (Q_1, Q_2, Q_3)\begin{pmatrix} 0 & P_3 & -P_2 \\ -P_3 & 0 & P_1 \\ P_2 & -P_1 & 0 \end{pmatrix}. \tag{I.7}$$

◇ **Exercise I.4:** The cross product $\mathbf{P}\times\mathbf{Q}$ is perpendicular to both $\mathbf{P}$ and $\mathbf{Q}$. In what direction does it point?

The cross product is not commutative and is not associative. It is, however, distributive with respect to addition or subtraction of vectors. Hence $\mathbf{P}\times(\mathbf{Q}\pm\mathbf{T}) = \mathbf{P}\times\mathbf{Q}\pm\mathbf{P}\times\mathbf{T}$.

The magnitude of $\mathbf{P}\times\mathbf{Q}$ equals $|\mathbf{P}|\,|\mathbf{Q}|\sin\theta$, where θ is the angle between the two vectors. The cross product therefore has a simple geometric interpretation. Its magnitude equals the area of the parallelogram defined by the two vectors.

◇ **Exercise I.5:** Given that $\mathbf{P}\times\mathbf{Q} = 0$, what does it tell us about the vectors involved?

As an example, the vector equation of a straight line is shown below for the case where the direction of the line and one point on the line are known. Assume that $\mathbf{d}$ is a unit vector in the direction of the line and $\mathbf{P}_1$ is a given point on the line. The equation of the entire line is

$$\mathbf{P}(t) = \mathbf{P}_1 + t\mathbf{d}, \text{ for any real } t. \tag{I.8}$$

◇ **Exercise I.6:** Derive the vector line equation for the straight segment between two given points $\mathbf{P}_1$ and $\mathbf{P}_2$.

> What if angry vectors veer
> Round your sleeping head, and form.
> There's never need to fear
> Violence of the poor world's abstract storm.
>
> — Robert Penn Warren, *Lullaby in Encounter*, 1957.

I.4.2 The Scalar Triple Product

The scalar triple product of three vectors $\mathbf{P}$, $\mathbf{Q}$, and $\mathbf{R}$ is defined as

$$S = \mathbf{P} \bullet (\mathbf{Q} \times \mathbf{R}) = P_1(Q_2R_3 - Q_3R_2) + P_2(Q_3R_1 - Q_1R_3) + P_3(Q_1R_2 - Q_2R_1)$$
$$= \begin{vmatrix} P_1 & P_2 & P_3 \\ Q_1 & Q_2 & Q_3 \\ R_1 & R_2 & R_3 \end{vmatrix}. \tag{I.9}$$

Interchanging two rows in a determinant changes its sign, so interchanging rows twice leaves the determinant unchanged. This is why the triple product is not affected by a cyclic permutation of its three components. We can therefore write

$$S = \mathbf{P} \bullet (\mathbf{Q}\times\mathbf{R}) = \mathbf{Q} \bullet (\mathbf{R}\times\mathbf{P}) = \mathbf{R} \bullet (\mathbf{P}\times\mathbf{Q}).$$

The triple product has a simple geometric interpretation. It equals the volume of the parallelepiped defined by the three vectors. An important corollary is: If the three vectors are coplanar, then the parallelepiped defined by them has volume zero, implying that their scalar triple product is zero.

I.4.3 Projecting a Vector

A common and useful operation on vectors is projecting a vector $\mathbf{a}$ on another vector $\mathbf{b}$. The idea is to break vector $\mathbf{a}$ up into two perpendicular components $\mathbf{c}$ and $\mathbf{d}$, such that $\mathbf{c}$ is in the direction of $\mathbf{b}$.

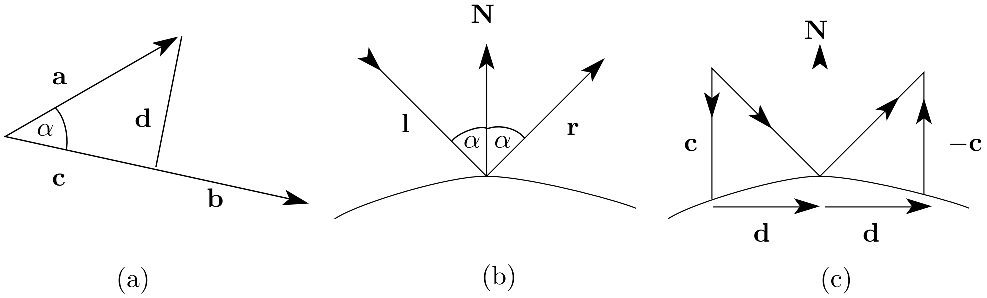


Figure I.1: Projecting a Vector.

Figure I.1a shows that $\mathbf{a} = \mathbf{c} + \mathbf{d}$ and $|\mathbf{c}| = |\mathbf{a}| \cos\alpha$. On the other hand $\mathbf{a} \bullet \mathbf{b} = |\mathbf{a}|\,|\mathbf{b}| \cos\alpha$, yielding the magnitude of $\mathbf{c}$

$$|\mathbf{c}| = |\mathbf{a}|\frac{(\mathbf{a} \bullet \mathbf{b})}{|\mathbf{a}|\,|\mathbf{b}|} = \frac{(\mathbf{a} \bullet \mathbf{b})}{|\mathbf{b}|}. \tag{I.10}$$

The direction of $\mathbf{c}$ is identical to the direction of $\mathbf{b}$, so we can write vector $\mathbf{c}$ as

$$\mathbf{c} = |\mathbf{c}|\frac{\mathbf{b}}{|\mathbf{b}|} = \frac{(\mathbf{a} \bullet \mathbf{b})}{|\mathbf{b}|^2}\mathbf{b}. \tag{I.11}$$

Example: Given vectors $\mathbf{a} = (2, 1)$ and $\mathbf{b} = (1, 0)$ it is easy to calculate

$$\mathbf{c} = \frac{(\mathbf{a} \bullet \mathbf{b})}{|\mathbf{b}|^2}\mathbf{b} = \frac{2\times 1 + 1\times 0}{1^2 + 0^2}(2, 0) = (4, 0), \qquad \mathbf{d} = \mathbf{a} - \mathbf{c} = (-2, 1).$$

◇ **Exercise I.7:** The projection method above works also for three-dimensional vectors. Given vectors $\mathbf{a} = (2, 1, 3)$ and $\mathbf{b} = (1, 0, -1)$, calculate the projection of $\mathbf{a}$ on $\mathbf{b}$.

◇ **Exercise I.8:** Vectors and their operations have been known for a long time. Explain why they have become important in the last few decades, since the introduction of the digital computer.

I.5 Complex Numbers

Complex numbers are expressed in terms of the special number i that is defined as $\sqrt{-1}$ and, hence, satisfies $i\times i = i^2 = -1$. Any complex number z can be represented either as the sum $a+bi$ or as the pair (a, b), where a and b are real. The *conjugate* of z is denoted by z^* and is defined as $a - bi$. Complex conjugates roughly correspond to negative real numbers. The sum of the real numbers a and $-a$ is zero and the sum $z+z^*$ is $2a$, which is real. The *magnitude* or *absolute value* of a complex number is denoted by $|z|$ and is defined as $\sqrt{z \cdot z^*} = \sqrt{a^2 + b^2}$. The sum and the difference of the complex numbers $a+bi$, $c+di$ are the obvious $(a+b)\pm(c+d)i$. The product makes use of the relation $i^2 = -1$ and is $(a + bi)(c + di) = (ac - bd) + (ad + bc)i$. The inverse, z^{-1}, of z is defined as $z^*/|z|$. It corresponds to the reciprocal $1/a$ of a real number a. The division z_1/z_2 is easy to perform for $|z_2| \neq 0$:

$$\frac{z_1}{z_2} = \frac{z_1 z_2^*}{z_2 z_2^*} = \frac{(a + bi)(c - di)}{c^2 + d^2} = \left(\frac{ac + bd}{c^2 + d^2}, \frac{bc - ad}{c^2 + d^2}\right).$$

The multiplication rule of complex numbers can be interpreted as a rotation in two dimensions. This is easy to see if we consider the product of the two complex numbers (x, y) and $(\cos\theta, \sin\theta)$.

$$\begin{aligned}(x, y) \cdot (\cos\theta, \sin\theta) &= (x\cos\theta - y\sin\theta, x\sin\theta + y\cos\theta)\\ &= (x, y)\begin{pmatrix}\cos\theta & \sin\theta\\ -\sin\theta & \cos\theta\end{pmatrix}.\end{aligned} \tag{I.12}$$

This product rotates the two-dimensional point (x, y) through an angle θ about the origin.

◇ **Exercise I.9:** Use the rule of complex number multiplication to multiply the complex number $(0, 1)$ by itself.

◇ **Exercise I.10:** Use the rule of complex number multiplication to multiply the complex number (a, b) by the number $(-a, -b)$.

> The shortest path between two truths in the real domain passes through the complex domain.
>
> —Jacques Hadamard.

Complex numbers can be represented graphically as two-dimensional points where the real part is the x coordinate and the imaginary part is the y coordinate. Such a representation is called an *Argand diagram*.

We normally use the *Cartesian coordinates* (a, b) of a point $\mathbf{P}$. The *polar coordinates* of $\mathbf{P}$ are (r, θ), where $r = \sqrt{a^2 + b^2}$ is the distance of the point from the origin and $\theta = \arctan(b/a)$ is the angle between the x axis and vector r. Given a complex number $z = (x, y)$, r is its absolute value and θ is called its *argument* (`arg` for short). The polar coordinates can be obtained from the Cartesian ones by $(a, b) = (r\cos\theta, r\sin\theta)$. Since the complex number $z = (a, b)$ can be interpreted as a two-dimensional point, it has the polar representation $z = (r\cos\theta, r\sin\theta) = r(\cos\theta + i\sin\theta)$. This representation is useful in many applications.

The famous Euler formula

$$e^{i\theta} = \cos\theta + i\sin\theta$$

allows us to write $z = re^{i\theta}$, a representation that makes it easy to multiply and divide complex numbers

$$z_1 z_2 = r_1 r_2 e^{i(\theta_1+\theta_2)}, \quad \frac{z_1}{z_2} = r_1 r_2 e^{i(\theta_1-\theta_2)},$$

and even extract roots

$$\sqrt[n]{z} = \sqrt[n]{r}\left[\cos\left(\frac{\theta + 2k\pi}{n}\right) + i\sin\left(\frac{\theta + 2k\pi}{n}\right)\right], \quad k = 0, 1, \ldots, n-1.$$

The n roots of z can be visualized as equally-spaced points lying on the circumference of a circle of radius $\sqrt[n]{r}$ whose center is at the origin. Connecting them produces an n-sided regular polygon.

◇ **Exercise I.11:** (Mathematical.) We know that $i = \sqrt{-1}$. What is $\sqrt{i}$?

◇ **Exercise I.12:** While we are at it, what are i^i and $\ln i$?

Bibliography

Nahin, Paul J., (1998) *An Imaginary Tale: The Story of* $\sqrt{-1}$, Princeton, NJ, Princeton University Press.

The Complex Number Song

(Tune: John Brown's Body)

Mine eyes have seen the glory of the Argand diagram
They have seen the i's and thetas of De Moivre's mighty plan
Now I can find the complex roots with consummate elan
With the root of minus one

Complex numbers are so easy
Complex numbers are so easy
Complex numbers are so easy
With the root of minus one

In Cartesian co-ordinates the complex plane is fine,
But the grandeur of the polar form this beauty doth outshine
You be raising i+40 to the power of 99
With the root of minus one

You'll realise your understanding was just second rate
When you see the power and magic of the complex conjugate
Drawing vectors corresponding to the roots of minus eight
With the root of minus one

(Attributed to Mrs P. E. Perella.)

I.6 Convolution

This is an important quantity that has several practical applications. It is used in Sections 5.6.1 and 5.8. We start with the simple, intuitive concept of a *system*. This is anything that receives input and generates output in response. The input and output can be one-dimensional (a function of the time), two-dimensional (a function of two spatial variables), or can have any number of dimensions. We will be concerned with the relation of the output to the input, not with the internal operation of the system. We will also concentrate on *linear systems*, since they are both simple and important. A linear system is defined as follows: If input $x_1(t)$ produces output $y_1(t)$ [we denote this by $x_1(t) \to y_1(t)$] and if $x_2(t) \to y_2(t)$, then $x_1(t) + x_2(t) \to y_1(t) + y_2(t)$. Any system that does not satisfy this condition is considered nonlinear.

This definition implies that $2x_1(t) = x_1(t) + x_1(t) \to y_1(t) + y_1(t) = 2y_1(t)$ or, in general, $a\,x_1(t) \to a\,y_1(t)$ for any real a.

Some linear systems are *shift invariant*. If such a linear system satisfies $x(t) \to y(t)$, then $x(t-T) \to y(t-T)$, i.e., shifting the input by an amount T shifts the output by the same amount, but does not otherwise affect the output. In the discussion of convolution, we assume that the systems in question are linear and shift-invariant. This is true (or true to a very good approximation) for electrical networks and optical systems, the main pieces of hardware used in image processing and compression.

It is useful to have a general relation between the input and output of a linear, System. Frequently used without need.

Dayton has adopted the commission system of government.	Dayton has adopted government by commission.
The dormitory system	Dormitories

— Strunk and White, *The Elements of Style*.

shift-invariant system. It turns out that the expression

$$y(t) = \int_{-\infty}^{+\infty} f(t,\tau)x(\tau)\,d\tau, \tag{I.13}$$

is general enough for this purpose. In other words, there is always a two-parameter function $f(t,\tau)$ that can be used to predict the output $y(t)$ if the input $x(\tau)$ is known. However, we want to express this relation with a one-parameter function, and we use the shift-invariance of the system for this purpose. For a linear, shift-invariant system we can write

$$y(t-T) = \int_{-\infty}^{+\infty} f(t,\tau)x(\tau-T)\,d\tau.$$

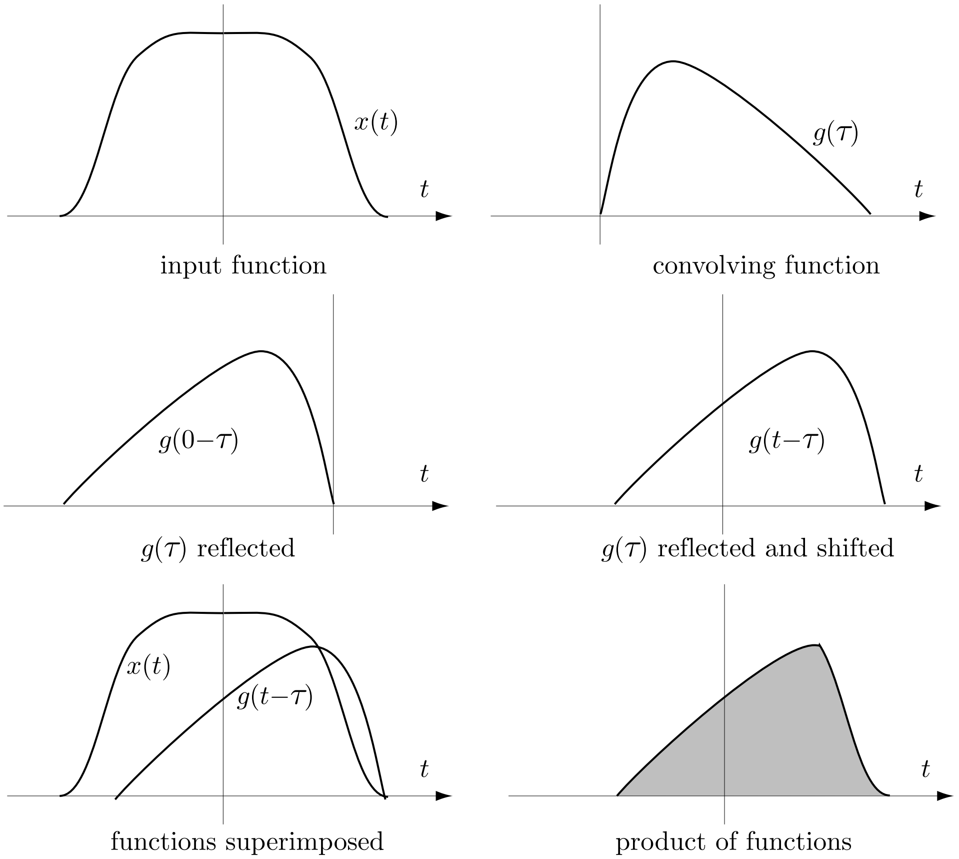
If we change variables by adding T to both t and τ, we get

$$y(t) = \int_{-\infty}^{+\infty} f(t+T,\tau+T)x(\tau)\,d\tau. \tag{I.14}$$

Comparing Equations (I.13) and (I.14) shows that $f(t,\tau) = f(t+T,\tau+T)$. Thus, function f has the property that if we add T to both its parameters, it does not change. The function is constant as long as the difference between its parameters is constant. Function f depends only on the difference of its parameters, so it is essentially a single parameter function. We can therefore write $g(t-\tau) = f(t,\tau)$, which changes Equation (I.13) to

$$y(t) = \int_{-\infty}^{+\infty} g(t-\tau)x(\tau)\,d\tau. \tag{I.15}$$

This is the *convolution integral*, an important relation between $x(t)$ and $y(t)$ or between $x(t)$ and $g(t)$. This relation is denoted $y = g \star x$ and it says that the output of a linear, shift-invariant system is given by the convolution of its input with a certain function $g(t)$ (or by *convolving* x with g). Function g, which is characteristic of the system, is called the *impulse response* of the system. Figure I.2 shows a graphical description of a convolution, where the final result (the integral) is the gray area under the curve.



input function

convolving function

$g(\tau)$ reflected

$g(\tau)$ reflected and shifted

functions superimposed

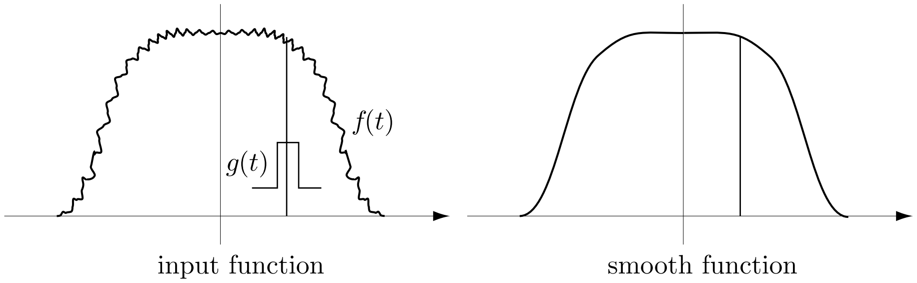
product of functions

Figure I.2: The Convolution of $x(t)$ and $g(t)$.

"Oh no," George said. "It was more than money."

He leaned his forehead in his hand and tried to remember what else more than money. The darkness inside his head was full of convolutions. His eardrums were too tight. Only the higher registers of sound were getting through.

—Paul Scott, *The Bender*



input function

smooth function

Figure I.3: Applying Convolution to Denoising a Function.

The convolution has a number of important properties. It is commutative, associative, and distributive over addition. These properties are shown in Equation (I.16)

$$\begin{aligned} f\star g &= g\star f,\\ f\star(g\star h) &= (f\star g)\star h,\\ f\star(g+h) &= f\star g+f\star h. \end{aligned}\tag{I.16}$$

Practical problems normally involve discrete sequences of numbers, rather than continuous functions, so the *discrete convolution* is useful. The discrete convolution of the two sequences $f(i)$ and $g(i)$ is defined as

$$h(i)=f(i)\star g(i)=\sum_j f(j)\,g(i-j).\tag{I.17}$$

If the lengths of $f(i)$ and $g(i)$ are m and n, respectively, then $h(i)$ has length $m+n-1$.

Example: Given the two sequences $f=\big(f(0),f(1),\ldots,f(5)\big)$ (six elements) and $g=\big(g(0),g(1),\ldots,g(4)\big)$ (five elements), Equation (I.17) yields the ten elements of the convolution $h=f\star g$

$$\begin{aligned}
h(0)&=\sum_{j=0}^{0}f(j)g(0-j)=f(0)g(0)\\
h(1)&=\sum_{j=0}^{1}f(j)g(1-j)=f(0)g(1)+f(1)g(0)\\
h(2)&=\sum_{j=0}^{2}f(j)g(2-j)=f(0)g(2)+f(1)g(1)+f(2)g(0)\\
h(3)&=\sum_{j=0}^{3}f(j)g(3-j)=f(0)g(3)+f(1)g(2)+f(2)g(1)+f(3)g(0)\\
h(4)&=\sum_{j=0}^{4}f(j)g(4-j)=f(0)g(4)+f(1)g(3)+f(2)g(2)+f(3)g(1)+f(4)g(0)\\
h(5)&=\sum_{j=1}^{5}f(j)g(5-j)=f(1)g(4)+f(2)g(3)+f(3)g(2)+f(4)g(1)+f(5)g(0)\\
h(6)&=\sum_{j=2}^{5}f(j)g(6-j)=f(2)g(4)+f(3)g(3)+f(4)g(2)+f(5)g(1)\\
h(7)&=\sum_{j=3}^{5}f(j)g(7-j)=f(3)g(4)+f(4)g(3)+f(5)g(2)
\end{aligned}$$

$$h(8) = \sum_{j=4}^{5} f(j)g(8-j) = f(4)g(4) + f(5)g(3)$$

$$h(9) = \sum_{j=5}^{5} f(j)g(9-j) = f(5)g(4)$$

A simple example of the use of a convolution is smoothing (or denoising). This shows how convolution can be used as a filter. Given a noisy function $f(t)$ (Figure (I.16)), we select a rectangular pulse as the convolving function $g(t)$. It is defined as

$$g(t) = \begin{cases} 1, & -a/2 < t < a/2, \\ \frac{1}{2}, & t = \pm a/2, \\ 0, & \text{elsewhere}, \end{cases}$$

where a is a suitably small value (typically 1, but could be anything). As the convolution proceeds, the pulse is moved from left to right and is multiplied by $f(t)$. The result of the product is a local average of $f(t)$ over an interval of width a. This has the effect of suppressing the high frequency fluctuations of $f(t)$.

From the Dictionary

convolution: coiling together

convolve: roll together

I.7 Voronoi Diagrams

Imagine a petri dish ready for growing bacteria. Four bacteria of different types are simultaneously placed in it at different points and immediately start multiplying. We assume that their colonies grow at the same rate. Initially, each colony consists of a growing circle around one of the starting points. After a while some of them meet and stop growing in the meeting area due to lack of food. The final result is that the entire dish gets divided into four areas, one around each of the four starting points, such that all the points within area i are closer to starting point i than to any other start point. Such areas are called *Voronoi regions* or *Dirichlet tessellations*. Figure I.4a shows the Voronoi regions for four points placed approximately at the four corners of a dashed rectangle. The regions are close to the four quadrants of the rectangle. Figure I.4b,c shows how the regions change when the points are moved.

At the time of writing there are on the web several Java applets that demonstrate the concepts discussed here. A typical example is [Zhao 98].

Bibliography

Zhao, Zhiyuan (1998) is an applet at
`http://ra.cfm.ohio-state.edu/~zhao/algorithms/algorithms.html`.

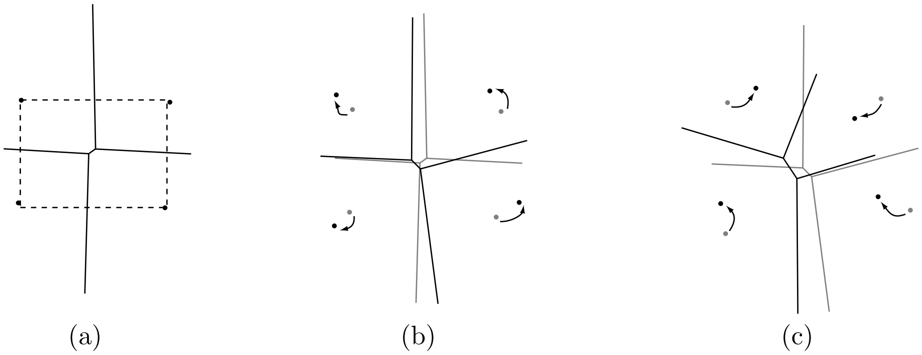


Figure I.4: Three Voronoi Diagrams of Four Points.

I.8 L Systems

Lindenmayer Systems (or L-systems for short) were developed by the biologist Aristid Lindenmayer in 1968 as a tool [Lindenmayer 68] to describe the morphology of plants. They were initially used in computer science, in the 1970s, as a tool to define formal languages, but have become really popular only after 1984, when Alvy Ray Smith pointed out [Smith 84] that L-systems can be used to draw many types of fractals, in addition to their use in botany. Today L-systems are also used to generate tilings, geometric art, and even musical scores.

The main idea of L-systems is to define a complex object by (1) defining an initial simple object, called the *axiom*, and (2) giving rules that show how to replace parts of the axiom.

> The following true story is an example of Aristid's modesty. At one of the American conferences somebody asked him what the L in "L-systems" stands for. Aristid's answer was "Languages."
>
> —Grzegorz Rozenberg.

The rules are applied successively, creating parts that get more and more complex, thereby transforming the simple axiom closer to the final, complex goal. The rules are called *rewriting* or *production* rules, and are an extension of Chomsky's work on formal grammars, and also of the BNF notation. N. Chomsky showed, in the 1950s, how to describe the syntax of a natural language by means of production rules. At about the same time Backus and Naur developed the BNF notation, which is based on rewriting rules, specifically to provide a formal definition [Naur 60] of the syntax of ALGOL 60.

Figure I.5 shows how a fractal, the Koch snowflake curve, is constructed, in several steps, out of an axiom that is a simple triangle (I.5a) and a rewriting rule that says: Replace each straight segment with the curve of I.5b. Figure I.5c is the

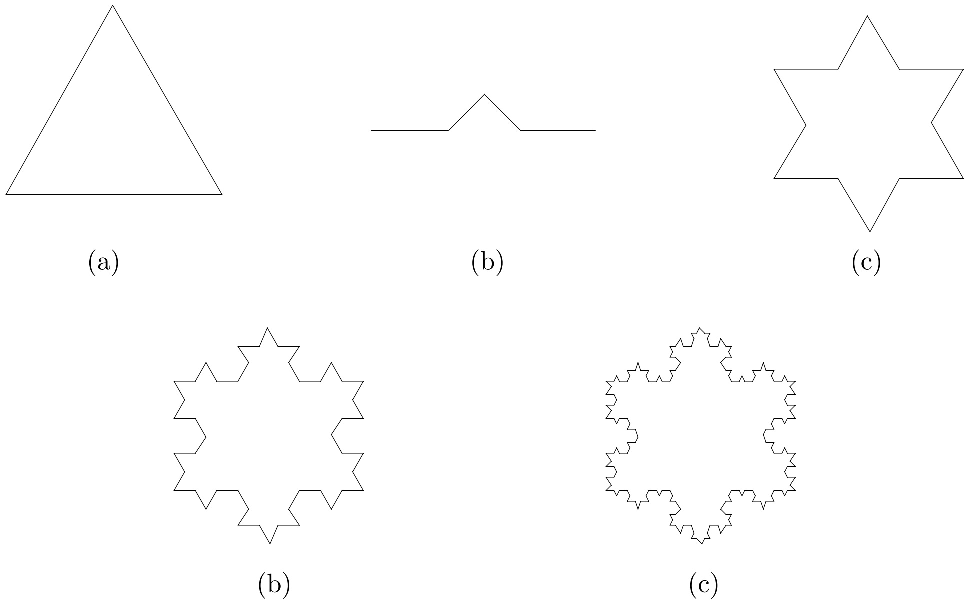


Figure I.5: Successive Generations of the Koch Snowflake.

result of applying the rule on **all** three triangle sides. Figure I.5d is the result of applying the same rule on all 12 sides of I.5c, and so on.

Notice that in order to construct iteration $i+1$ of an object, the rule has to be applied to **all** parts of iteration i of the object. This is the main difference between L-systems and Chomsky grammars, and this is also one reason why L-systems are so powerful. Another reason is the notation used in modern L-systems, a notation introduced in 1979 by A. Szilard and R. E. Quinton and improved by P. Prusinkiewicz in 1986. It is based on the LOGO language [Abelson 82] and the concept of turtle moves. These two differences are illustrated below.

Example: An L-system dealing with the two letters x and y. The axiom is y and the two rewriting rules are: $x \to xy$ (every occurrence of x should be replaced by xy) and $y \to x$ (every occurrence of y should be replaced by x). The first iteration starts with the axiom y, and applies **both** rules to it. The first rule does not apply, and the second yields x. The result of the first iteration is thus x. Iteration 2 applies both rules to x. The first rule replaces x by xy and the second rule does not apply, since the original string x did not have any y in it. Iteration 3 replaces the x of xy by xy and the y of xy by x. The result is xyx. Successive iterations produce the strings

$$y \to x \to xy \to xyx \to xyxxy \to xyxxyxyx.$$

(This does not seem useful, but wait until this method is applied to geometric shapes.) The two parts on the left and right of a production rule are called its *predecessor* and *successor*, respectively. An L-system such as the one above is called a D0L-system. (D0L stands for Deterministic, Context-Free L-system. Notice that most texts on L-systems corrupt this name and spell it “DOL” instead of “D0L.”)

Turtle Moves: It is possible to define geometric shapes by imagining a turtle moving in the two-dimensional plane, sometimes leaving marks behind. The LOGO programming language supports drawing commands that “move” the turtle from point to point and cause it to turn at an angle when it reaches a point. The production rules of L-systems also use this notation. Mathematically, the state of the turtle is represented by a triplet (x, y, α) where (x, y) are the present coordinates of the turtle and α is its heading. The basic notation used in such a rule employs the following characters:

F : The turtle moves forward a distance d, drawing a straight line of a given thickness W. The state of the turtle changes from (x, y, α) to $(x + d\cos\alpha, y + d\sin\alpha, \alpha)$.

f : The turtle moves forward as above, but without drawing anything.

$+$: The turtle turns to the right (clockwise) by a given angle δ. Its new state is thus $(x, y, \alpha + \delta)$.

$-$: The turtle turns to the left (counterclockwise) by the same angle δ. Its new state is $(x, y, \alpha - \delta)$.

Table I.7 shows several more character commands that have traditionally been used in L-systems. As more research is done in this field, the number of turtle commands will grow, but the reader has to keep one important convention in mind: When a rewriting rule contains a command that the turtle (i.e., the computer implementation of L-systems) does not understand, *that command is ignored*; no error message is issued. This convention is useful and is commonly used in drawing complex shapes.

Table I.7 implies that several more parameters, such as C, sl, and Δ, are needed to completely specify the shape being drawn. These parameters should be supported by any computer implementation of L-systems; they should have default values, and should be easy for the user to modify. These parameters are listed in Table I.8.

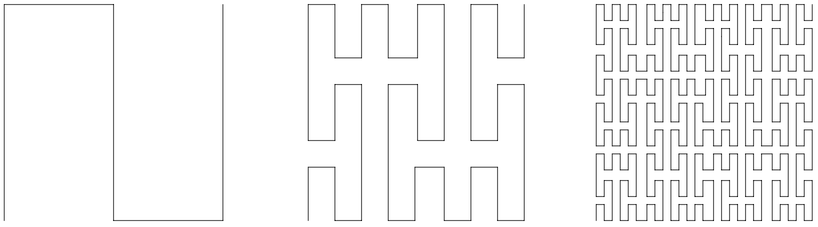
The string `F+F+F+F` is a command to move forward one line length, turn right, and repeat three more times. If the turn angle is 90°, the result is a square of size d. If the initial turtle heading is $\alpha = 90°$, then the start/end point is the bottom left corner of the square (Figure I.9a). The string `FFF+FF+F+F-F-F-FF+F+FFF` draws the shape of Figure I.9b.

The Koch snowflake of Figure I.5 was generated by an L-system with an axiom `F++F++F` and the single production rule `F->F-F++F-F`. The initial heading was 0° and the turn angle 60°. Figure I.6 shows three iterations of the Peano space-filling curve drawn with both an initial heading and a turn angle of 90°.

The L-system for this curve consists of the axiom `X` and the two production rules

`X->XFYFX+F+YFXFY-F-XFYFX` and `Y->YFXFY-F-XFYFX+F+YFXFY`.

The key to understanding this L-system is the rule that any unknown turtle commands (in this case the characters `X` and `Y`) should be ignored. The first iteration draws the axiom `X`, which is unknown, causing nothing to be drawn. The next iteration executes the two rewriting rules. The first rule replaces the axiom `X` with



(a) (b) (c)

Figure I.6: Three Iterations of the Peano Curve.

XFYFX+F+YFXFY-F-XFYFX, which is plotted (since X and Y are unknown) as FF+F+FF-F-FF. The second rule looks for a Y in the axiom, but finds none. The iteration thus draws FF+F+FF-F-FF, which results in Figure I.6a. The next iteration starts with XFYFX+F+YFXFY-F-XFYFX, replaces each X with the successor of rule 1, and replaces each Y with the successor of rule 2. The result is a very long string, that, when drawn, produces the curve of Figure I.6b.

The L-system for the Hilbert curve is similarly defined by the axiom X and the 2 production rules

X->-YF+XFX+FY- and Y->+XF-YFY-FX+.

◇ **Exercise I.13:** Show how to get the 4 orientations of the Hilbert curve out of the L-system above.

Abelson, H. and A. A. diSessa (1982) *Turtle Geometry*, Cambridge, MA, MIT Press.

Prusinkiewicz, Przemysław (1986) *Graphical Applications of L-systems*, in Proc. of Graphics Interface '86—Vision Interface '86, pp .247–253.

Prusinkiewicz, P., and A. Lindenmayer (1990) *The Algorithmic Beauty of Plants*, New York, Springer Verlag.

Prusinkiewicz, P., A. Lindenmayer, and F. D. Fracchia (1991) "Synthesis of Space-Filling Curves on the Square Grid," in *Fractals in the Fundamental and Applied Sciences*, edited by Peitgen, H.-O. et al., Amsterdam, Elsevier Science Publishers, pp. 341–366.

Smith, Alvy Ray (1984) "Plants, Fractals and Formal Languages," *Computer Graphics* **18**(3):1–10.

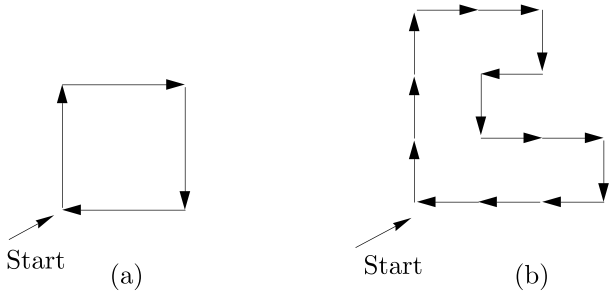
Szilard, A. L. and R. E. Quinton (1979) "An Interpretation for D0L Systems by Computer Graphics," *The Science Terrapin* **4**:8–13.

F	Move forward d units and draw a line.
f	Move forward d units without drawing.
+	Turn clockwise by an angle δ.
-	Turn counterclockwise by an angle δ.
|	Reverse direction (rotate by 180°).
[	Push current turtle state into the stack.
]	Pop current turtle state from the stack.
#	Increment the line width W by an amount w.
!	Decrement the line width W by an amount w.
@	Draw a dot with radius W.
{	Open a polygon.
}	Close a polygon and fill it with color C.
<	Divide line length d by scale factor sl.
>	Multiply line length d by scale factor sl.
&	Swap meaning of $+$ and $-$.
(	Decrement turn angle δ by Δ.
)	Increment turn angle δ by Δ.
*	Match any character (used in context-sensitive L-systems only).
...	Ignore rule (used in context-sensitive L-systems only).

Table I.7: L-system Conventions for Turtle Commands.

d	The line length.
sl	Scale factor for line length d.
W	The line width.
w	The line width increment.
α	The initial turtle heading.
δ	Turn angle.
Δ	Increment/decrement the turn angle δ.
C	Default color for polygon fill.

Table I.8: Additional Turtle Parameters.



(a) (b)

Figure I.9: Examples of Turtle Movements.

I.9 The Greek Alphabet

A	α	alpha	I	ι		iota	P	ρ	ϱ	rho
B	β	beta	K	κ		kappa	Σ	σ	ς	sigma
Γ	γ	gamma	Λ	λ		lambda	T	τ		tau
Δ	δ	delta	M	μ		mu	Υ	υ		upsilon
E	ϵ	epsilon	N	ν		nu	Φ	ϕ	φ	phi
Z	ζ	zeta	Ξ	ξ		xi	X	ξ		xi
H	η	eta	O	o		omicron	Ψ	ψ		psi
Θ	θ	theta	Π	π	ϖ	pi	Ω	ω		omega

I.10 Interpolating Polynomials

This section shows how to predict the value of a pixel from those of 16 of its near neighbors by means of a two-dimensional interpolating polynomial. The results are used in Table 4.118.

We start with an intuitive discussion of the term *interpolation*. Given two numbers a and b, their average $(a+b)/2$ is always located midway between them, so we can use the average to interpolate them. However, given four numbers a, b, c, and d, their average $(a+b+c+d)/4$ is not a good interpolation, since it is not located "midway" between the four. A simple example is the four numbers 1, 1, 1, and 100. Their average is close to 25, so it is nowhere "in the middle" of the four numbers. Interpolating four numbers is therefore done by (1) converting the numbers to two-dimensional points, (2) calculating a smooth curve that passes through the points, and (3) finding the midpoint of the curve.

Any numbers a, b, c, and d can be converted to the points $(1,a)$, $(2,b)$, $(3,c)$, and $(4,d)$. It is intuitively clear that the midpoint (x,y) of a smooth curve that passes through those points is a good candidate for the title "the interpolation of the four points." The y coordinate becomes the interpolation of the four numbers, and the x coordinate is ignored.

This method is called one-dimensional interpolation. It can be extended to more than four numbers, and also to pixels, where it becomes two-dimensional interpolation. As mentioned before, we want to use a group of 16 neighboring pixels to predict the value of a pixel at the center of the group. The main idea is to consider the 16 neighbor pixels a 4×4 equally-spaced points on a surface (where the value of a pixel is interpreted as the height of the surface) and to derive a polynomial function $\mathbf{P}(u,w)$ that passes through all 16 points. Graphically, $\mathbf{P}(u,w)$ can be thought of as a surface. The value of the pixel at the center of the 4×4 group can then be predicted by calculating the height of the center point $\mathbf{P}(.5,.5)$ of the surface. Mathematically, this surface is the two-dimensional polynomial interpolation of the 16 points.

I.10.1 One-Dimensional Interpolation

A surface can be viewed as an extension of a curve, so we start by deriving a one-dimensional polynomial (a curve) that interpolates four points, then extend it to a two-dimensional polynomial (a surface) that interpolates a grid of 4×4 points.

Given four points $\mathbf{P}_1$, $\mathbf{P}_2$, $\mathbf{P}_3$, and $\mathbf{P}_4$ we look for a polynomial that will pass through them. In general, a polynomial of degree n in x is defined (Section 3.23) as the function

$$P_n(x) = \sum_{i=0}^{n} a_i x^i = a_0 + a_1 x + a_2 x^2 + \cdots + a_n x^n, \qquad \text{(I.18)}$$

where a_i are the $n+1$ coefficients of the polynomial and the parameter x is a real number. The one-dimensional interpolating polynomial that is of interest to us is special, and differs from the definition above in two respects

1. This polynomial goes from point $\mathbf{P}_1$ to point $\mathbf{P}_4$. Its length is finite, and it is therefore better to describe it as the function

$$P_n(t) = \sum_{i=0}^{n} a_i t^i = a_0 + a_1 t + a_2 t^2 + \cdots + a_n t^n;\ \text{where } 0 \le t \le 1.$$

This is the *parametric representation* of a polynomial. We want this polynomial to go from $\mathbf{P}_1$ to $\mathbf{P}_4$ when the parameter t is varied from 0 to 1.

2. The only given data are the four points and we have to use them to calculate all $n+1$ coefficients of the polynomial. This suggests the value $n = 3$ (a polynomial of degree 3, a cubic polynomial; one which has four coefficients). The idea is to set up and solve four equations, with the four coefficients as the unknowns, and with the four points as known quantities. Thus, we use the notation (T indicates transpose)

$$\mathbf{P}(t) = \mathbf{a}t^3 + \mathbf{b}t^2 + \mathbf{c}t + \mathbf{d} = (t^3, t^2, t, 1)(\mathbf{a}, \mathbf{b}, \mathbf{c}, \mathbf{d})^T = \mathbf{T}(t) \cdot \mathbf{A}. \qquad \text{(I.19)}$$

The four coefficients $\mathbf{a}, \mathbf{b}, \mathbf{c}, \mathbf{d}$ are shown in boldface because they are not numbers. Keep in mind that the polynomial has to pass through the given points, so the value of $\mathbf{P}(t)$ for any t must be the three coordinates of a point. Each coefficient should therefore be a triplet. $\mathbf{T}(t)$ is the row vector $(t^3, t^2, t, 1)$, and $\mathbf{A}$ is the column vector $(\mathbf{a}, \mathbf{b}, \mathbf{c}, \mathbf{d})^T$. Calculating the curve therefore involves finding the values of the four unknowns $\mathbf{a}$, $\mathbf{b}$, $\mathbf{c}$, and $\mathbf{d}$. $\mathbf{P}(t)$ is called a *parametric cubic* (or PC) polynomial.

It turns out that degree 3 is the smallest one that is still useful for an interpolating polynomial. A polynomial of degree 1 has the form $\mathbf{P}_1(t) = \mathbf{c}t + \mathbf{d}$ and is, therefore, a straight line, so it can only be used in special cases. A polynomial of degree two (quadratic) has the form $\mathbf{P}_2(t) = \mathbf{b}t^2 + \mathbf{c}t + \mathbf{d}$ and is a conic section, so it can only take a few different shapes. A polynomial of degree 3 (cubic) is thus the simplest one that can take on complex shapes, and can also be a space curve.

◇ **Exercise I.14:** Prove that a quadratic polynomial must be a plane curve.

Our ultimate problem is to interpolate pixels. Pixels are always equally-spaced, so we assume that the two interior points $\mathbf{P}_2$ and $\mathbf{P}_3$ are equally spaced between $\mathbf{P}_1$ and $\mathbf{P}_4$. The first point $\mathbf{P}_1$ is the start point $\mathbf{P}(0)$ of the polynomial, the last point, $\mathbf{P}_4$ is the endpoint $\mathbf{P}(1)$, and the two interior points $\mathbf{P}_2$ and $\mathbf{P}_3$ are the two equally-spaced interior points $\mathbf{P}(1/3)$ and $\mathbf{P}(2/3)$ of the polynomial.

We thus write $\mathbf{P}(0) = \mathbf{P}_1$, $\mathbf{P}(1/3) = \mathbf{P}_2$, $\mathbf{P}(2/3) = \mathbf{P}_3$, $\mathbf{P}(1) = \mathbf{P}_4$, or

$$\begin{aligned}
\mathbf{a}(0)^3 + \mathbf{b}(0)^2 + \mathbf{c}(0) + \mathbf{d} &= \mathbf{P}_1,\\
\mathbf{a}(1/3)^3 + \mathbf{b}(1/3)^2 + \mathbf{c}(1/3) + \mathbf{d} &= \mathbf{P}_2,\\
\mathbf{a}(2/3)^3 + \mathbf{b}(2/3)^2 + \mathbf{c}(2/3) + \mathbf{d} &= \mathbf{P}_3,\\
\mathbf{a}(1)^3 + \mathbf{b}(1)^2 + \mathbf{c}(1) + \mathbf{d} &= \mathbf{P}_4.
\end{aligned}$$

These equations are easy to solve and the solutions are:

$$\begin{aligned}
\mathbf{a} &= -9/2\mathbf{P}_1 + 27/2\mathbf{P}_2 - 27/2\mathbf{P}_3 + 9/2\mathbf{P}_4,\\
\mathbf{b} &= 9\mathbf{P}_1 - 45/2\mathbf{P}_2 + 18\mathbf{P}_3 - 9/2\mathbf{P}_4,\\
\mathbf{c} &= -11/2\mathbf{P}_1 + 9\mathbf{P}_2 - 9/2\mathbf{P}_3 + \mathbf{P}_4,\\
\mathbf{d} &= \mathbf{P}_1.
\end{aligned}$$

Substituting into Equation (I.19) gives

$$\begin{aligned}
\mathbf{P}(t) =&(-9/2\mathbf{P}_1 + 27/2\mathbf{P}_2 - 27/2\mathbf{P}_3 + 9/2\mathbf{P}_4)t^3\\
&+ (9\mathbf{P}_1 - 45/2\mathbf{P}_2 + 18\mathbf{P}_3 - 9/2\mathbf{P}_4)t^2\\
&+ (-11/2\mathbf{P}_1 + 9\mathbf{P}_2 - 9/2\mathbf{P}_3 + \mathbf{P}_4)t + \mathbf{P}_1.
\end{aligned}$$

Which, after rearranging, becomes

$$\begin{aligned}
\mathbf{P}(t) =&(-4.5t^3 + 9t^2 - 5.5t + 1)\mathbf{P}_1 + (13.5t^3 - 22.5t^2 + 9t)\mathbf{P}_2\\
&+ (-13.5t^3 + 18t^2 - 4.5t)\mathbf{P}_3 + (4.5t^3 - 4.5t^2 + t)\mathbf{P}_4\\
=&G_1(t)\mathbf{P}_1 + G_2(t)\mathbf{P}_2 + G_3(t)\mathbf{P}_3 + G_4(t)\mathbf{P}_4\\
=&\mathbf{G}(t)\cdot\mathbf{P}, \qquad\qquad \text{(I.20)}
\end{aligned}$$

where

$$\begin{aligned}
G_1(t) &= (-4.5t^3 + 9t^2 - 5.5t + 1), & G_2(t) &= (13.5t^3 - 22.5t^2 + 9t),\\
G_3(t) &= (-13.5t^3 + 18t^2 - 4.5t), & G_4(t) &= (4.5t^3 - 4.5t^2 + t);
\end{aligned} \qquad \text{(I.21)}$$

$\mathbf{P}$ is the column $(\mathbf{P}_1, \mathbf{P}_2, \mathbf{P}_3, \mathbf{P}_4)^T$, and $\mathbf{G}(t)$ is the row vector

$$\bigl(G_1(t), G_2(t), G_3(t), G_4(t)\bigr).$$

The functions $G_i(t)$ are called *blending functions*, since they create any point on the curve as a blend of the four given points. Note that they add up to 1 for any value of t. This property must be satisfied by any set of blending functions, and such functions are called *barycentric*. We can also write

$$G_1(t) = (t^3, t^2, t, 1)(-4.5, 9, -5.5, 1)^T$$

and, similarly, for $G_2(t)$, $G_3(t)$, and $G_4(t)$. In matrix notation this becomes

$$\mathbf{G}(t) = (t^3, t^2, t, 1) \begin{pmatrix} -4.5 & 13.5 & -13.5 & 4.5 \\ 9.0 & -22.5 & 18 & -4.5 \\ -5.5 & 9.0 & -4.5 & 1.0 \\ 1.0 & 0 & 0 & 0 \end{pmatrix} = \mathbf{T}(t) \cdot \mathbf{N}. \tag{I.22}$$

The curve can now be written $\mathbf{P}(t) = \mathbf{G}(t) \cdot \mathbf{P} = \mathbf{T}(t) \cdot \mathbf{N} \cdot \mathbf{P}$. $\mathbf{N}$ is called the basis matrix and $\mathbf{P}$ is the geometry vector. From Equation (I.19) we know that $\mathbf{P}(t) = \mathbf{T}(t) \cdot \mathbf{A}$, so we can write $\mathbf{A} = \mathbf{N} \cdot \mathbf{P}$. Equation (5.17) illustrates an application of this interpolating polynomial for image compression.

> The word *barycentric* is derived from *barycenter*, meaning "center of gravity," because such weights are used to calculate the center of gravity of an object. Barycentric weights have many uses in geometry in general, and in curve and surface design in particular.

Given the four points, the interpolating polynomial can be calculated in two steps:

1. Set-up the equation $\mathbf{A} = \mathbf{N} \cdot \mathbf{P}$ and solve it for $\mathbf{A} = (\mathbf{a}, \mathbf{b}, \mathbf{c}, \mathbf{d})^T$.
2. The polynomial is $\mathbf{P}(t) = \mathbf{T}(t) \cdot \mathbf{A}$.

I.10.2 Example

(This example is in two dimensions, each of the four points $\mathbf{P}_i$ and each of the four coefficients $\mathbf{a}$, $\mathbf{b}$, $\mathbf{c}$, and $\mathbf{d}$ is a pair. For three-dimensional curves, the method is the same, except that triplets should be used, instead of pairs.) Given the four two-dimensional points $\mathbf{P}_1 = (0,0)$, $\mathbf{P}_2 = (1,0)$, $\mathbf{P}_3 = (1,1)$, and $\mathbf{P}_4 = (0,1)$, we set up the equation

$$\begin{pmatrix} \mathbf{a} \\ \mathbf{b} \\ \mathbf{c} \\ \mathbf{d} \end{pmatrix} = \mathbf{A} = \mathbf{N} \cdot \mathbf{P} = \begin{pmatrix} -4.5 & 13.5 & -13.5 & 4.5 \\ 9.0 & -22.5 & 18 & -4.5 \\ -5.5 & 9.0 & -4.5 & 1.0 \\ 1.0 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} (0,0) \\ (1,0) \\ (1,1) \\ (0,1) \end{pmatrix},$$

which is easy to solve

$$\begin{aligned} \mathbf{a} &= -4.5(0,0) + 13.5(1,0) - 13.5(1,1) + 4.5(0,1) = (0,-9), \\ \mathbf{b} &= 19(0,0) - 22.5(1,0) + 18(1,1) - 4.5(0,1) = (-4.5, 13.5), \\ \mathbf{c} &= -5.5(0,0) + 9(1,0) - 4.5(1,1) + 1(0,1) = (4.5, -3.5), \\ \mathbf{d} &= 1(0,0) - 0(1,0) + 0(1,1) - 0(0,1) = (0,0). \end{aligned}$$

$$\text{Thus} \qquad \mathbf{P}(t) = \mathbf{T} \cdot \mathbf{A} = (0,-9)t^3 + (-4.5, 13.5)t^2 + (4.5, -3.5)t.$$

It is now easy to calculate and verify that $\mathbf{P}(0) = (0,0) = \mathbf{P}_1$, and

$$\mathbf{P}(1/3) = (0,-9)1/27 + (-4.5, 13.5)1/9 + (4.5, -3.5)1/3 = (1,0) = \mathbf{P}_2,$$

$$\mathbf{P}(1) = (0,-9)1^3 + (-4.5, 13.5)1^2 + (4.5, -3.5)1 = (0,1) = \mathbf{P}_4.$$

◇ **Exercise I.15:** Calculate $\mathbf{P}(2/3)$ and verify that it is equal to $\mathbf{P}_3$.

◇ **Exercise I.16:** Imagine the circular arc of radius one in the first quadrant (a quarter circle). Write the coordinates of the four points that are equally spaced on this arc. Use the coordinates to calculate a PC interpolating polynomial approximating this arc. Calculate point $\mathbf{P}(1/2)$. How far does it deviate from the midpoint of the true quarter circle?

The main advantage of this method is its simplicity. Given the four points, it is easy to calculate the PC polynomial that passes through them.

◇ **Exercise I.17:** This method makes sense if the four points are (at least approximately) equally spaced along the curve. If they are not, the following may be done: Instead of using 1/3 and 2/3 as the intermediate values, the user may specify values α, β such that $\mathbf{P}_2 = \mathbf{P}(\alpha)$ and $\mathbf{P}_3 = \mathbf{P}(\beta)$. Generalize Equation (I.22) such that it depends on α and β.

I.10.3 Two-Dimensional Interpolation

The PC polynomial, Equation (I.19), can easily be extended to two dimensions by means of a technique called *Cartesian product.* The polynomial is generalized from a cubic curve to a *bicubic* surface.

A one-dimensional PC polynomial has the form $\mathbf{P}(t) = \sum_{i=0}^{3} \mathbf{a}_i t^i$. Two such curves, $\mathbf{P}(u)$ and $\mathbf{P}(w)$, can be combined by means of this technique to form the surface:

$$\begin{aligned}\mathbf{P}(u,w) &= \sum_{i=0}^{3}\sum_{j=0}^{3} \mathbf{a}_{ij} u^i w^j\\ &= \mathbf{a}_{33}u^3w^3 + \mathbf{a}_{32}u^3w^2 + \mathbf{a}_{31}u^3w + \mathbf{a}_{30}u^3 + \mathbf{a}_{23}u^2w^3 + \mathbf{a}_{22}u^2w^2 + \mathbf{a}_{21}u^2w + \mathbf{a}_{20}u^2\\ &+ \mathbf{a}_{13}uw^3 + \mathbf{a}_{12}uw^2 + \mathbf{a}_{11}uw + \mathbf{a}_{10}u + \mathbf{a}_{03}w^3 + \mathbf{a}_{02}w^2 + \mathbf{a}_{01}w + \mathbf{a}_{00}\\ &= (u^3, u^2, u, 1)\begin{pmatrix}\mathbf{a}_{33} & \mathbf{a}_{32} & \mathbf{a}_{31} & \mathbf{a}_{30}\\ \mathbf{a}_{23} & \mathbf{a}_{22} & \mathbf{a}_{21} & \mathbf{a}_{20}\\ \mathbf{a}_{13} & \mathbf{a}_{12} & \mathbf{a}_{11} & \mathbf{a}_{10}\\ \mathbf{a}_{03} & \mathbf{a}_{02} & \mathbf{a}_{01} & \mathbf{a}_{00}\end{pmatrix}\begin{pmatrix}w^3\\ w^2\\ w\\ 1\end{pmatrix}, \quad \text{where } 0 \le u, w \le 1. \end{aligned} \tag{I.23}$$

This is a double cubic polynomial (hence the name *bicubic*) with 16 terms, where each of the 16 coefficients $\mathbf{a}_{ij}$ is a triplet. Note that the surface depends on all 16 coefficients. Any change in any of them produces a different surface. Equation (I.23) is the *algebraic representation* of a bicubic surface. In order to use it in practice, the 16 unknown coefficients have to be expressed in terms of the 16 known, equally-spaced points. We denote these points

$$\begin{matrix}\mathbf{P}_{03} & \mathbf{P}_{13} & \mathbf{P}_{23} & \mathbf{P}_{33}\\ \mathbf{P}_{02} & \mathbf{P}_{12} & \mathbf{P}_{22} & \mathbf{P}_{32}\\ \mathbf{P}_{01} & \mathbf{P}_{11} & \mathbf{P}_{21} & \mathbf{P}_{31}\\ \mathbf{P}_{00} & \mathbf{P}_{10} & \mathbf{P}_{20} & \mathbf{P}_{30}.\end{matrix}$$

To calculate the 16 unknown coefficients, we write 16 equations, each based on one of the given points:

$$\begin{matrix}\mathbf{P}(0,0)=\mathbf{P}_{00} & \mathbf{P}(0,1/3)=\mathbf{P}_{01} & \mathbf{P}(0,2/3)=\mathbf{P}_{02} & \mathbf{P}(0,1)=\mathbf{P}_{03}\\ \mathbf{P}(1/3,0)=\mathbf{P}_{10} & \mathbf{P}(1/3,1/3)=\mathbf{P}_{11} & \mathbf{P}(1/3,2/3)=\mathbf{P}_{12} & \mathbf{P}(1/3,1)=\mathbf{P}_{13}\\ \mathbf{P}(2/3,0)=\mathbf{P}_{20} & \mathbf{P}(2/3,1/3)=\mathbf{P}_{21} & \mathbf{P}(2/3,2/3)=\mathbf{P}_{22} & \mathbf{P}(2/3,1)=\mathbf{P}_{23}\\ \mathbf{P}(1,0)=\mathbf{P}_{30} & \mathbf{P}(1,1/3)=\mathbf{P}_{31} & \mathbf{P}(1,2/3)=\mathbf{P}_{32} & \mathbf{P}(1,1)=\mathbf{P}_{33}.\end{matrix}$$

Solving, substituting the solutions in Equation (I.23), and simplifying produces the *geometric representation* of the bicubic surface

$$\mathbf{P}(u,w)=(u^3,u^2,u,1)\mathbf{N}\begin{pmatrix}\mathbf{P}_{33} & \mathbf{P}_{32} & \mathbf{P}_{31} & \mathbf{P}_{30}\\ \mathbf{P}_{23} & \mathbf{P}_{22} & \mathbf{P}_{21} & \mathbf{P}_{20}\\ \mathbf{P}_{13} & \mathbf{P}_{12} & \mathbf{P}_{11} & \mathbf{P}_{10}\\ \mathbf{P}_{03} & \mathbf{P}_{02} & \mathbf{P}_{01} & \mathbf{P}_{00}\end{pmatrix}\mathbf{N}^T\begin{pmatrix}w^3\\ w^2\\ w\\ 1\end{pmatrix}, \tag{I.24}$$

where $\mathbf{N}$ is the Hermite matrix of Equation (I.22).

The surface of Equation (I.24) can now be used to predict the value of a pixel as a polynomial interpolation of 16 of its near neighbors. All that is necessary is to substitute $u = 0.5$ and $w = 0.5$. The following *Mathematica* code

```
Clear[Nh,P,U,W];
Nh={{-4.5,13.5,-13.5,4.5},{9,-22.5,18,-4.5},
 {-5.5,9,-4.5,1},{1,0,0,0}};
P={{p33,p32,p31,p30},{p23,p22,p21,p20},
 {p13,p12,p11,p10},{p03,p02,p01,p00}};
U={u^3,u^2,u,1};
W={w^3,w^2,w,1};
u:=0.5;
w:=0.5;
Expand[U.Nh.P.Transpose[Nh].Transpose[W]]
```

does that and produces

$$\begin{aligned}&\mathbf{P}(.5,.5)\\ &\quad=0.00390625\mathbf{P}_{00}-0.0351563\mathbf{P}_{01}-0.0351563\mathbf{P}_{02}+0.00390625\mathbf{P}_{03}\\ &\quad-0.0351563\mathbf{P}_{10}+0.316406\mathbf{P}_{11}+0.316406\mathbf{P}_{12}-0.0351563\mathbf{P}_{13}\\ &\quad-0.0351563\mathbf{P}_{20}+0.316406\mathbf{P}_{21}+0.316406\mathbf{P}_{22}-0.0351563\mathbf{P}_{23}\\ &\quad+0.00390625\mathbf{P}_{30}-0.0351563\mathbf{P}_{31}-0.0351563\mathbf{P}_{32}+0.00390625\mathbf{P}_{33},\end{aligned}$$

where the 16 coefficients are the ones used in Table 4.118.

◇ **Exercise I.18:** How can this method be used in cases where not all 16 points are known?

◇ **Exercise I.19:** The center point of the surface is calculated as a weighted sum of the 16 equally-spaced data points. It makes sense to assign small weights to points located away from the center, but our result assigns *negative* weights to eight of the 16 points. Explain the meaning of negative weights and show what role they play in interpolating the center of the surface.

Readers who find it hard to follow the details above should compare the way two-dimensional polynomial interpolation is presented here to the way it is discussed by [Press et al. 88]. The following quotation is from page 125: "... The formulas that obtain the c's from the function and derivative values are just a complicated linear transformation, with coefficients which, having been determined once, in the mists of numerical history, can be tabulated and forgotten."

> Seated at his disorderly desk, caressed by a counterpane of drifting tobacco haze, he would pore over the manuscript, crossing out, interpolating, re-arguing, and then referring to volumes on his shelves.
>
> Christopher Morley, *The Haunted Bookshop*