Appendix A

Mathematical background

A.1 Vectors and matrices

A brief summary of vector and matrix conventions and operations is given here. An excellent handbook for vector and matrix computation is (Barnett, 1990) and readers should refer to it for details. An alternative that may also be found helpful is (Golub and van Loan, 1989). The vector and matrix operations described should be available in appropriate programming languages such as **matlab**.

Vectors Throughout the book, vectors are denoted in bold, for example

$$\mathbf{r} = \begin{pmatrix} x \\ y \end{pmatrix} \quad \text{or} \quad \mathbf{a} = \begin{pmatrix} a_1 \\ a_2 \\ a_3 \end{pmatrix}.$$

Scalar product and vector product take their usual meanings and are denoted

$$\mathbf{a} \cdot \mathbf{b} \quad \text{and} \quad \mathbf{a} \times \mathbf{b}$$

respectively. In three dimensions, for instance,

$$\mathbf{a} \cdot \mathbf{b} = a_1 b_1 + a_2 b_2 + a_3 b_3.$$

and

$$\mathbf{a} \times \mathbf{b} = (a_2 b_3 - a_3 b_2,\ a_3 b_1 - a_1 b_3,\ a_3 b_1 - a_1 b_3)^T.$$

The magnitude of a vector may be measured via its **Euclidean norm**:

$$|\mathbf{r}| = \sqrt{\mathbf{r} \cdot \mathbf{r}}$$

and a vector $\mathbf{r}$ for which $|\mathbf{r}| = 1$ is said to be normalised, or a "unit" vector.

Matrices Matrices are generally non-bold capitals, for example A, with components denoted A_{ij}. The transpose A^T is defined by

$$A^T_{ij} = A_{ji}.$$

The **rank** of a matrix A is the number of linearly independent vectors that comprise its columns.

A matrix operation that is frequently useful is the **Kronecker product**

$$A \otimes B = \begin{pmatrix} A_{11}B & A_{12}B & \dots \\ A_{21}B & A_{22}B & \dots \\ \dots & \dots & \dots \end{pmatrix} \tag{A.1}$$

which combines two arrays of dimension $M_1 \times N_1$ and $M_2 \times N_2$ to make a larger one of dimension $M_1 M_2 \times N_1 N_2$.

Linear equations The linear simultaneous equations

$$A\mathbf{x} = \mathbf{b}$$

have a unique solution when A is square and is non-singular — that is, $\det A \neq 0$, where $\det A$ is the **determinant** of A. Then a solution can be found using the standard inverse

$$\mathbf{x} = A^{-1}\mathbf{b}.$$

If there is no solution, as may happen when A is not square, there may nonetheless be a unique, optimal, approximate solution which is expressed using a **pseudo-inverse** A^+:

$$\mathbf{x} = A^+\mathbf{b} \quad \text{where} \quad A^+ = (A^T A)^{-1} A^T.$$

(More general definitions of pseudo-inverse can be made, but are not used in this book.)

Rotation matrices Matrices for rotation about x, y and z axes are respectively denoted R_x, R_y and R_z where, for example,

$$R_z(\theta) = \begin{pmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{pmatrix} \tag{A.2}$$

so that a point in three dimensions given by a vector $\mathbf{r} = (x, y, z)^T$ is transformed to a rotated point

$$\mathbf{r}' = R_z(\theta)\mathbf{r}.$$

In two dimensions, a rotation is a 2×2 matrix

$$R(\theta) = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix}.$$

Rotation matrices have the property that they are **orthogonal**, satisfying $R^T R = I$. Generally, an orthogonal matrix U satisfies $U^T U = I$ and, in three dimensions, can be interpreted as a rotation (when $\det U = 1$) or a combination of rotation and reflection ($\det U = -1$).

Eigenvalues and eigenvectors The eigenvalues λ_n and eigenvectors $\mathbf{u}_n$ of a square $N \times N$ matrix A are defined as satisfying

$$A\mathbf{u}_n = \lambda_n \mathbf{u}_n, \ n = 1, \ldots, N.$$

Eigenvalues and eigenvectors may have complex values, unless A is **symmetric** ($A^T = A$) in which case they are guaranteed to be real.

The **trace** of the matrix is defined to be

$$\mathrm{tr}(A) = \sum_{n=1}^{N} A_{nn}$$

and has the property that

$$\mathrm{tr}(A) = \sum_{n=1}^{N} \lambda_n.$$

Diagonalisation Eigenvalues and eigenvectors can be used to decompose a square matrix A as

$$A = UDU^{-1}$$

where D is the "diagonal" matrix

$$D = \text{diag}(\lambda_1, \ldots, \lambda_N)$$

with the eigenvalues along the diagonal and zeros elsewhere. The matrix U consists of columns which are normalised eigenvectors of A. One important application of the diagonal form is in computing powers of A:

$$A^p = UD^pU^{-1} \quad \text{where} \quad D^p = \text{diag}(\lambda_1^p, \ldots, \lambda_N^p).$$

Setting $p = \frac{1}{2}$ allows a **square root** of A to be computed.

Singular value decomposition (SVD) An alternative form of decomposition of a matrix A is the SVD, which applies not only to square matrices but also to rectangular ones of size $M \times N$. It has the form

$$A = UDV$$

where U is an $M \times M$ matrix, D is a diagonal $M \times N$ matrix and V is an $N \times N$ matrix. Both U and V are orthogonal matrices. The diagonal values of D are $D_{nn} = \sigma_n$ where $\lambda_n = \sigma_n^2$ are eigenvalues of the symmetric matrix A^TA and are hence guaranteed to be positive.

A measure of the "size" of A is its **spectral radius** $\|A\|_2 = \sqrt{\lambda_1}$ where λ_1 is the largest eigenvalue of A^TA. The condition for an iterative process involving A, in which arbitrarily large powers A^n of A are applied to vectors, to be stable is that $\|A\|_2 < 1$.

A.2 B-spline basis functions

Useful introductory reference books on splines are (Faux and Pratt, 1979; Foley et al., 1990). An excellent, comprehensive reference is (Bartels et al., 1987).

In chapter 3, spline functions are written as a linear combination of a number of spline "basis functions." Basis functions are constructed using the following general rule which can be used to define any arbitrary set of polynomial splines. Let $B_{n,d}$ be the nth basis function for a spline of order d. Then for a spline with single knots of unit spacing, the following recursive rule applies:

Ground instance

$$B_{n,1}(s) = \begin{cases} 1 & \text{if } n \leq s < n+1 \\ 0 & \text{otherwise} \end{cases}$$

Inductive step

$$B_{n,d}(s) = \frac{(s-n)B_{n,d-1}(s) + (n+d-s)B_{n+1,d-1}(s)}{d-1}$$

and some examples are shown in figure A.1. These functions satisfy the following conditions:

Support $B_{n,d}(s) = 0$ for $s \notin [n, n+d)$

Positivity $B_{n,d}(s) \geq 0$ for all s

Normalisation $\sum_{-\infty}^{\infty} B_{n,d}(s) = 1$ for all s

Translational invariance $B_{n+1,d}(s) = B_{n,d}(s-1)$ for all s

and further, there is a smoothness constraint for $d > 1$, namely that $B_{n,d}$ has continuous $(d-2)$th derivative for all s and all $d > 1$.

Non-uniform B-spline functions

The spline basis functions generated above, for which the knots are uniformly spaced at unit intervals, can be generalised to produce spline functions with arbitrary knot spacing. Consider a spline with N_K knots at positions $k_0 \leq k_1 \leq \ldots \leq k_{N_K-1}$, then the recursive rule becomes

Ground instance

$$B_{n,1}(s) = \begin{cases} 1 & \text{if } k_n \leq s < k_{n+1} \\ 0 & \text{otherwise} \end{cases}$$

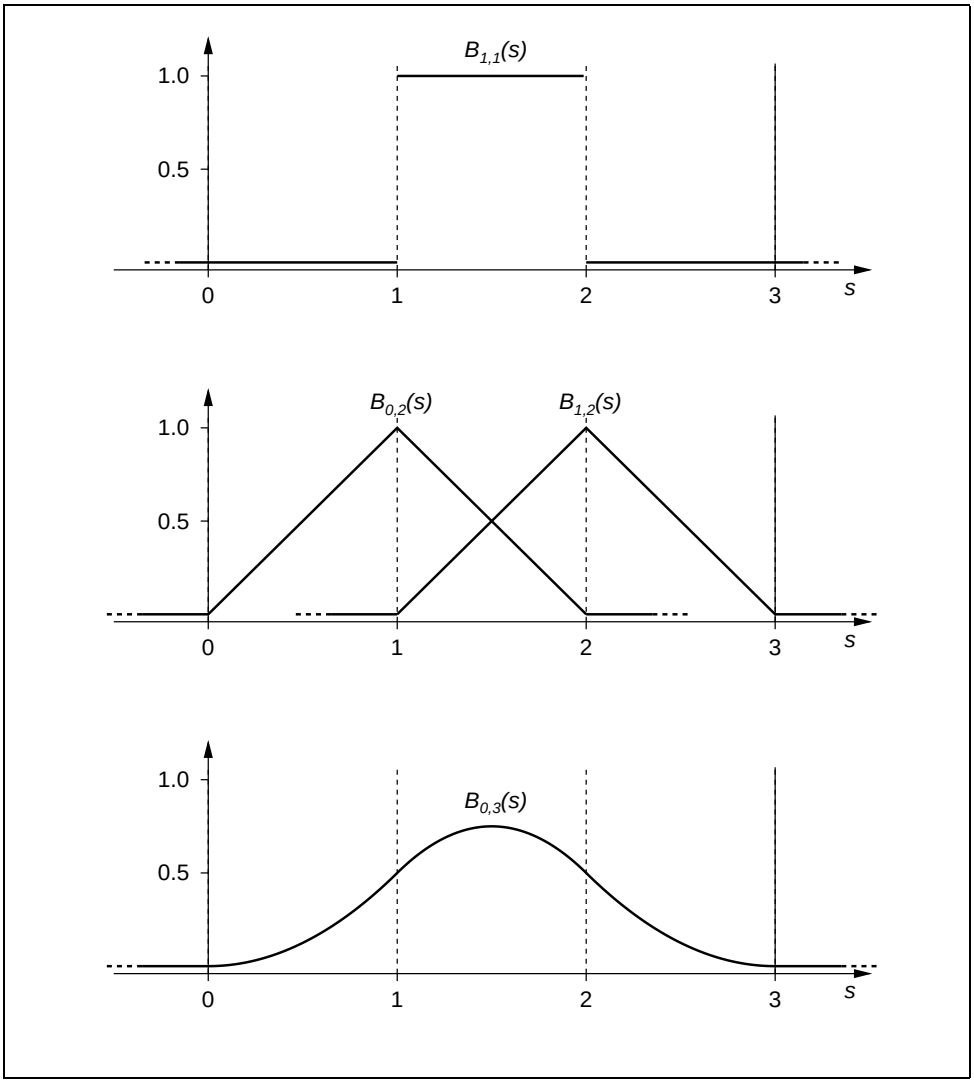

Figure A.1: A spline basis function $B_{n,d}$ of order d is built up recursively from basis functions of lower order.

Inductive step

$$B_{n,d}(s) = \frac{(s - k_n)B_{n,d-1}(s)}{k_{n+d-1} - k_n} + \frac{(k_{n+d} - s)B_{n+1,d-1}(s)}{k_{n+d} - k_{n+1}}$$

which reduces to the uniform case when $k_n = n$. The rule can be used to generate spline functions with knots of multiplicity m by setting m consecutive k_n to be equal. (Terms in the inductive step are zero when the denominator is zero. The validity of this can be shown by taking a limit as the knots approach one another — it is easy to see by induction that the basis function in the numerator is identically zero whenever the denominator is zero). The conditions of positivity and normalisation still hold in general, and now the support of the basis function $B_{n,d}$ is $[k_n, k_{n+d})$. The basis functions are clearly no longer necessarily translated copies of each other, however, and the introduction of a multiple knot reduces the smoothness of a basis function; the function is C^{d-1-m} at a knot of multiplicity m. This weakening of the smoothness property is the motivation for using multiple knots; it permits B-spline functions, and therefore curves, with sharp corners and discontinuities.

An implementation of B-spline functions

The recursive rule for generating $B_{n,d}$ can be converted into an algorithm by expressing each basis function as a sequence of polynomials $p_n(s)$ defined over the intervals $[k_n, k_{n+1})$. Since the support[1] of $B_{n,d}$ is $[k_n, k_{n+d})$, any spline basis function of order d can be represented using just d polynomials $B^{\sigma}_{n,d}$, one for each of the d spans S_σ in the support of $B_{n,d}$. Now the inductive step of the rule can be applied, over each interval in turn, to obtain each of the $B^{\sigma}_{n,d}$.

Where a B-spline contains multiple knots, some of the inter-knot intervals have zero length, so it is convenient to introduce the concept of "spans". These correspond to the non-empty inter-knot intervals above, and the span ends are called "breakpoints". A B-spline function, therefore, is a piecewise polynomial curve made up of a series of L spans $S_0 \ldots S_{L-1}$ connected at breakpoints $s_0 < s_1 < \ldots < s_L$. We adopt the convention that all spans are unit length, $(s_i = i)$, so the basis functions making up a spline are uniquely determined by the knot multiplicities $m_0 \ldots m_L$ at the breakpoints. A periodic B-spline function is constructed by considering the basis functions to be periodic over the interval $[0, L]$. A periodic B-spline function must have $m_0 = m_L$ and

[1]The support of a function is the interval over which it is non-zero.

it has "multiple knot count"

$$M = \sum_{1}^{L} (m_i - 1)$$

while for an aperiodic spline $m_0 = m_L = d$ to control the boundary conditions of the spline, and

$$M = \sum_{0}^{L} (m_i - 1).$$

An L span B-spline is a linear combination of N_B basis functions, where

$$N_B = L + M = N_K - m_0$$

for a periodic spline, and

$$N_B = L + M + 1 - d = N_K - d$$

for an aperiodic spline. Thus, for example, a simple L span aperiodic quadratic B-spline is a linear combination of $N_B = L+2$ basis functions (see figure 3.6 on page 48). The relationship between spans, knots and basis functions is illustrated for two cases in figures A.2 and A.3.

Building a spline function from basis functions

B-spline functions can be evaluated efficiently using "span matrices." Over the span S_σ, any spline function is a linear combination of the basis functions $B_{b_\sigma,d} \ldots B_{b_\sigma+d-1,d}$ where

$$b_\sigma = \left(\sum_{i=0}^{\sigma} m_i \right) - d$$

so

$$x(s)^{[\sigma,\sigma+1)} = x(s)^\sigma = \sum_{b_\sigma}^{b_\sigma+d-1} x_i B_{i,d}(s)$$

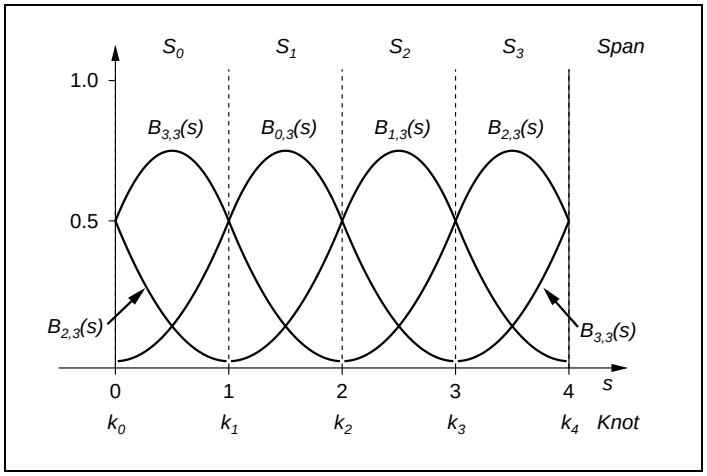

Figure A.2: A simple periodic B-spline with no multiple knots has $L = 4$ spans, $N_K = 5$ knots, and is a combination of $N_B = 4$ (periodic) basis functions.

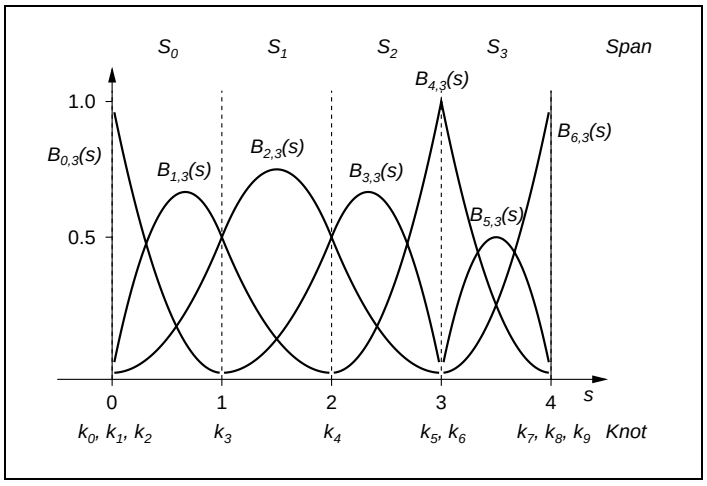

Figure A.3: An aperiodic spline must have knots of multiplicity d at its endpoints. Here there is also a double knot between the third and fourth spans, leading to a discontinuity in the first derivative of the function. Here $L = 4$, $N_K = 10$, $N_B = 7$.

(with obvious variations for periodic splines). For each span, therefore, we can compute a $d \times d$ span matrix B_σ^S such that

$$x(s+\sigma)^\sigma = (1 \; s \; \ldots \; s^{d-1}) B_\sigma^S \begin{pmatrix} x_{b_\sigma} \\ x_{b_\sigma+1} \\ \vdots \\ x_{b_\sigma+d-1} \end{pmatrix}$$

where the ith column of the span matrix corresponds to the polynomial coefficients of the basis function $B_{b_\sigma+i-1,d}$ over the interval of that span (in practice it is convenient to define each span matrix over the interval $[0, 1)$). The span matrices for the spline in figure A.3 are as follows:

$$B_0^S = \begin{pmatrix} 1.00 & 0.00 & 0.00 \\ -2.00 & 2.00 & 0.00 \\ 1.00 & -1.50 & 0.50 \end{pmatrix}$$

$$B_1^S = \begin{pmatrix} 0.50 & 0.50 & 0.00 \\ -1.00 & 1.00 & 0.00 \\ 0.50 & -1.00 & 0.50 \end{pmatrix}$$

$$B_2^S = \begin{pmatrix} 0.50 & 0.50 & 0.00 \\ -1.00 & 1.00 & 0.00 \\ 0.50 & -1.50 & 1.00 \end{pmatrix}$$

$$B_3^S = \begin{pmatrix} 1.00 & 0.00 & 0.00 \\ -2.00 & 2.00 & 0.00 \\ 1.00 & -2.00 & 1.00 \end{pmatrix}$$

and the algorithm used to calculate them is given in figure A.4. Once span matrices have been computed off-line, the spline can be evaluated efficiently at any values of s

To calculate span matrices for a non-periodic B-spline of order d with knot multiplicities $m_i,\ 0 \leq i \leq L$.

1. **Calculate** the knot values k_i:
   (a) **Initialise**: $p = 0,\ q = 0$
   (b) **For** $i = 0 \ldots L$
      i. **For** $j = 1 \ldots m_i$
         $k_p = q,\ p = p + 1$
      ii. $q = q + 1$
2. **For** each span $\sigma = 0 \ldots L - 1$:
   (a) **Find** the index b_σ of the first basis function whose support includes the span.

$$b_\sigma = \left(\sum_0^\sigma m_i \right) - d$$

   (b) **For** $i = 1 \ldots d$ recursively calculate the basis polynomial $B^\sigma_{b_\sigma+i-1,d}$ for span σ using the following rule
      i. **Ground instance**

$$B^\sigma_{n,1}(s) = \begin{cases} 1 & \text{if } k_n \leq \sigma < k_{n+1} \\ 0 & \text{otherwise} \end{cases}$$

      ii. **Recursive rule**

$$B^\sigma_{n,d}(s) = \frac{(s + \sigma - k_n) B^\sigma_{n,d-1}(s)}{k_{n+d-1} - k_n} + \frac{(k_{n+d} - s - \sigma) B^\sigma_{n+1,d-1}(s)}{k_{n+d} - k_{n+1}}$$

      where terms are zero when the denominator is zero.
      iii. **Store** the coefficients of $B^\sigma_{b_\sigma+i-1,d}$ as the ith column of the $d \times d$ span matrix B^S_σ, where the top row corresponds to the constant polynomial coefficient.

Figure A.4: Algorithm to calculate span matrices for aperiodic B-splines. *Obvious modifications must be made for the periodic case.*

and $x_0 \ldots x_{N_B-1}$. For notational purposes it is convenient also to define the $d \times N_B$ "placement matrices" G_σ:

$$(G_\sigma)_{ij} = \begin{cases} 1 & \text{if } i - b_\sigma = j \\ 0 & \text{otherwise} \end{cases} \tag{A.3}$$

so that

$$x(s+\sigma) = (1 \; s \; \ldots \; s^{d-1}) B_\sigma^S G_\sigma \mathbf{Q}$$

where $0 \leq s < 1$. The derivative of the function can be calculated as

$$x'(s+\sigma) = (0 \; 1 \; \ldots \; (d-1)s^{d-2}) B_\sigma^S G_\sigma \mathbf{Q}$$

and so when considering a spline curve,

$$\begin{pmatrix} x \\ y \end{pmatrix} = (1 \; s \; \ldots \; s^{d-1} \; 1 \; s \; \ldots \; s^{d-1}) \begin{pmatrix} B_\sigma^S G_\sigma & 0 \\ 0 & B_\sigma^S G_\sigma \end{pmatrix} \begin{pmatrix} \mathbf{Q}^x \\ \mathbf{Q}^y \end{pmatrix},$$

the tangent to the curve is given by

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = (0 \; 1 \; \ldots \; (d-1)s^{d-2} \; 0 \; 1 \; \ldots \; (d-1)s^{d-2}) \begin{pmatrix} B_\sigma^S G_\sigma & 0 \\ 0 & B_\sigma^S G_\sigma \end{pmatrix} \begin{pmatrix} \mathbf{Q}^x \\ \mathbf{Q}^y \end{pmatrix},$$

and the normal is given by

$$\begin{pmatrix} n_x \\ n_y \end{pmatrix} = \begin{pmatrix} -y' \\ x' \end{pmatrix}.$$

Calculating the spline metric matrix

Using the span matrices it is straightforward to compute the spline metric matrix $\mathcal{B}$, where

$$\mathcal{B} = \frac{1}{L} \int_0^L \mathbf{B}(s) \mathbf{B}(s)^T \, ds \tag{A.4}$$

as described on page 50.

$$\begin{aligned}\mathcal{B} &= \frac{1}{L}\sum_{\sigma=0}^{L-1}\left(\int_0^1 \mathbf{B}(s+\sigma)\mathbf{B}(s+\sigma)^T\,ds\right)\\ &= \frac{1}{L}\sum_{\sigma=0}^{L-1} G_\sigma^T (B_\sigma^S)^T \mathcal{P} B_\sigma^S G_\sigma\end{aligned}$$

where

$$\mathcal{P} = \int_0^1 \begin{pmatrix} 1 \\ \vdots \\ s^{d-1} \end{pmatrix} \begin{pmatrix} 1 & \ldots & s^{d-1} \end{pmatrix} ds,$$

the "Hilbert" matrix (Barnett, 1990) whose coefficients are

$$\mathcal{P}_{ij} = \frac{1}{i+j-1}.$$

Similarly, the matrix $\mathcal{B}'$ used on page 65 to define the area coefficients $\mathcal{A}$, was defined as

$$\mathcal{B}' = \frac{1}{L}\int_0^L \mathbf{B}(s)\mathbf{B'}^T(s)\,ds$$

and may be calculated as follows:

$$\mathcal{B} = \frac{1}{L}\sum_{\sigma=0}^{L-1} G_\sigma^T (B_\sigma^S)^T \mathcal{P}' B_\sigma^S G_\sigma$$

where

$$\begin{aligned}\mathcal{P}' &= \int_0^1 \begin{pmatrix} 1 \\ \vdots \\ s^{d-1} \end{pmatrix} \begin{pmatrix} 0 & \ldots & (d-1)s^{d-2} \end{pmatrix} ds, \quad \text{so}\\ \mathcal{P}'_{ij} &= \begin{cases} 0 & \text{if } i=j=1 \\ \frac{j-1}{i+j-2} & \text{otherwise} \end{cases}.\end{aligned}$$

A.3 Probability

An excellent introductory text on probability is (Papoulis, 1990). It is impossible to cover the necessary ground here, but since much of the argument in the book is probabilistic, a few basic concepts are reviewed here.

Probability distributions A continuous random variable x taking real values $x \in \mathbb{R}$ has a probability distribution defined by its **density function** $p(x) \geq 0$. Its interpretation is that, for an interval $\mathcal{I} = [a, b]$:

$$P(x \in \mathcal{I}) = \int_a^b p(x)\, dx.$$

This definition extends to a multi-dimensional random variable $\mathbf{X} \in \mathbb{R}^{N_X}$ so that, for a subset $\mathcal{I} \in \mathbb{R}^{N_X}$:

$$P(\mathbf{X} \in \mathcal{I}) = \int_{\mathcal{I}} p(\mathbf{X})\, d\mathbf{X}.$$

Since $\mathbf{X}$ has to take some value, p must satisfy the normalisation property that

$$\int_{\mathbb{R}^{N_X}} p(\mathbf{X})\, d\mathbf{X} = 1.$$

A **conditional** distribution for X specifies the probable values of $\mathbf{X}$ *given* that the value of some related variable $\mathbf{Y}$ is known and is defined by the density $p(X|Y)$. This is interpreted, as before, via integration:

$$P(\mathbf{X} \in \mathcal{I}|\mathbf{Y}) = \int_{\mathcal{I}} p(\mathbf{X}|\mathbf{Y})\, d\mathbf{X}.$$

The associated normalisation property is

$$\int_{\mathbb{R}^{N_X}} p(\mathbf{X}|\mathbf{Y})\, d\mathbf{X} = 1.$$

Mean and variance The expectation or mean of the random variable $\mathbf{X}$, denoted $\mathcal{E}[\mathbf{X}]$, is

$$\mathcal{E}[\mathbf{X}] = \int_{\mathbb{R}^{N_X}} p(\mathbf{X})\, d\mathbf{X}$$

which is a linear operation so that

$$\mathcal{E}[A\mathbf{X} + \mathbf{b}] = A\mathcal{E}[\mathbf{X}] + \mathbf{b}.$$

The variance of $\mathbf{X}$, denoted $\mathcal{V}[\mathbf{X}]$ is defined as an expectation:

$$\mathcal{V}[\mathbf{X}] = \mathcal{E}[(\mathbf{X} - \overline{\mathbf{X}})(\mathbf{X} - \overline{\mathbf{X}})^T]$$

where $\overline{\mathbf{X}} = \mathcal{E}[\mathbf{X}]$. It scales quadratically, as $\mathcal{V}[A\mathbf{X} + \mathbf{b}] = A\mathcal{V}[\mathbf{X}]A^T$, and is invariant to the additive constant $\mathbf{b}$, naturally enough since it is a measure of "spread" about the mean. It is also known as the "covariance matrix" of $\mathbf{X}$ and must be symmetric and "positive semi-definite" (all eigenvalues positive or zero).

Bayes' rule Suppose a density $p(\mathbf{X})$ is given, based on prior knowledge of the state $\mathbf{X}$ of some system and its likely values. Then suppose that observations $\mathbf{Z}$ are made from an imperfect sensing device which is characterised by its **observation density** $p(\mathbf{Z}|\mathbf{X})$, specifying the likely range of observations *given* a particular system state $\mathbf{X}$. Then Bayes' rule gives the **posterior density** $p(\mathbf{X}|\mathbf{Z})$:

$$p(\mathbf{X}|\mathbf{Z}) = kp(\mathbf{Z}|\mathbf{X})p(\mathbf{X}),$$

where k is a constant, not dependent on $\mathbf{X}$, whose value can be determined if need be by insisting that the posterior be normalised. Note that $p(\mathbf{Z}|\mathbf{X})$ is also known as a **likelihood** function for $\mathbf{X}$.

Estimation When both the prior and observation density are available, a common way to estimate the value of $\mathbf{X}$ given observations $\mathbf{Z}$ is simply to find the $\mathbf{X}$ that maximises the posterior:

$$\hat{\mathbf{X}} = \arg\max_{\mathbf{X}} p(\mathbf{X}|\mathbf{Z}).$$

This is known as the MAP (Maximum A Posteriori) estimate. Alternatively, if no prior is available, an estimator can be defined by

$$\hat{\mathbf{X}} = \arg\max_{\mathbf{X}} p(\mathbf{Z}|\mathbf{X}),$$

the MLE (Maximum Likelihood Estimator).

Normal distribution Much use is made in the book of multi-variate normal or Gaussian distributions. A vector variable $\mathbf{X}$ distributed as a Gaussian is denoted

$$\mathbf{X} \sim \mathcal{N}(\overline{\mathbf{X}}, P) \tag{A.5}$$

where $\overline{\mathbf{X}}$ is the mean of the distribution and P is its covariance matrix, assumed non-singular. The density function for $\mathbf{X}$ is

$$p(\mathbf{X}) = \frac{1}{\sqrt{2\pi}^{N_X}} \frac{1}{\sqrt{\det P}} \exp -\frac{1}{2}(\mathbf{X} - \overline{\mathbf{X}})^T S(\mathbf{X} - \overline{\mathbf{X}})$$

where $S = P^{-1}$, the **information matrix**.

Alternatively, given a vector $\mathbf{w}$ of N_X independent standard normal distributions, so that each $w_n \sim \mathcal{N}(0,1)$ and $\mathbf{w} \sim \mathcal{N}(\mathbf{0}, I_{N_X})$, $\mathbf{X}$ can be described as a linear transformation of $\mathbf{w}$:

$$\mathbf{X} = B\mathbf{w} + \overline{\mathbf{X}}$$

where $B = \sqrt{P}$. Circles $|\mathbf{w}| < c$ map to **confidence** ellipsoids in $\mathbf{X}$ space, regions which contain the value of $\mathbf{X}$ with probability $\chi^2_{N_X}(c)$, where χ^2_ν is the "chi-squared distribution function" for ν degrees of freedom, and can be found in statistical tables. For example, for $N_X = 2$,

$$P(|\mathbf{w}| < 2) = 86\% \quad \text{and} \quad P(|\mathbf{w}| < 3) = 99\%.$$

Appendix B

Stochastic dynamical systems

B.1 Continuous-time first-order dynamics

A first-order AR process (9.7) can be regarded as a first-order "stochastic differential equation" (SDE) in continuous time that has been sampled at regular intervals. If the sampling interval is τ so that $t_k = k\tau$, then the AR process is obtained by integrating the SDE over successive sampling intervals. The SDE is expressed as

$$\dot{\mathbf{X}} = F(\mathbf{X} - \overline{\mathbf{X}}) + G\dot{\mathbf{w}} \tag{B.1}$$

where $\mathbf{X}(t)$ is a vector in shape-space, F and G are $N_X \times N_X$ matrices and $\mathbf{w}(t)$ is a N_X-dimensional vector of independent, univariate Brownian processes in continuous time. A univariate Brownian process w has the property that the value $w(t)$ has a Gaussian distribution with $\mathcal{E}[w(t)] = 0$ and $\mathcal{V}[w(t)] = t$. The derivative $\dot{w}(t)$ is a "white noise" signal, that is one with equal power at all frequencies. The coefficients F are the deterministic parameters of the process, in the sense that its eigenvalues λ_i are the so-called "poles" of the AR process, constants with units of inverse time that represent the rates of decay of the various characteristic motions of the system. (This applies to the case that all poles are real-valued and negative. Any real positive pole will cause the process to be unstable. There is also the possibility of complex poles, representing oscillations or damped oscillations.) The matrix G represents a coupling to the multi-dimensional white noise $\dot{\mathbf{w}}$ that is driving the dynamical system. As in the discrete case, there is a mean-state and a Riccati equation for continuous time:

$$\dot{\hat{\mathbf{X}}} = \mathbf{F}(\hat{\mathbf{X}} - \overline{\mathbf{X}}) \quad \text{and} \quad \dot{\mathbf{P}} = \mathbf{FP} + \mathbf{PF^T} + \mathbf{Q} \tag{B.2}$$

where the "covariance coefficient" $Q = GG^T$.

Conversion between continuous and discrete time

A continuous-time SDE can be converted to a discrete-time form (Gelb, 1974; Astrom and Wittenmark, 1984) by computing A and C directly from F and Q:

$$A = \exp F\tau \quad \text{and} \quad C = \int_0^\tau (\exp Ft)\, Q(\exp F^T t)\, dt. \tag{B.3}$$

It is possible to evaluate the integral for C exactly by diagonalising F but in practice the following approximation, to lowest order in τ, is convenient (and particularly so for the second-order process):

$$A = (I - F\tau)^{-1} \quad \text{and} \quad C = Q\tau \quad \text{so that} \quad B = G\sqrt{\tau}. \tag{B.4}$$

The approximation for A is known as the "backward difference" approximation and is preferable to the more obvious "forward difference" $A = F\tau$ because it preserves stability: that is, any SDE that is stable is approximated as a stable AR process, with $\|A\|_2 < 1$.

Power spectrum

In chapter 9, a form (9.12) on page 199 for the power spectrum of a first-order ARP is used. That form is derived briefly here. Restricting $\mathbf{X}(t)$ in the continuous process above to be one-dimensional $X(t)$, so that F and G are scalar coefficients, suppose that

$$X(t) \propto \exp 2\pi i f t,$$

and set its mean to zero for simplicity. Then (B.1) becomes

$$2\pi i f X = FX + G\dot{w}$$

so that

$$X = \frac{G\dot{w}}{2\pi i f - F}$$

and the power spectrum

$$S_{XX}(f) = \left| \frac{G}{2\pi i f - F} \right|^2 S_{\dot{w}\dot{w}}(f),$$

where $|\cdot|$ denotes complex modulus. Now the power spectrum $S_{\dot{w}\dot{w}}(f)$ of white noise is constant, so

$$S_{XX}(f) \propto \frac{G^2}{4\pi^2 f^2 + F^2}$$

which can be rewritten directly in the required form.

B.2 Second-order dynamics in continuous time

A second-order SDE can be written, as it was in the discrete case, as a first-order one in a suitable state-space:

$$\dot{\mathcal{X}} = F(\mathcal{X} - \overline{\mathcal{X}}) + G\dot{\mathbf{w}} \tag{B.5}$$

where now

$$F = \begin{pmatrix} 0 & I \\ F_1 & F_2 \end{pmatrix} \quad \text{and} \quad G = \begin{pmatrix} 0 \\ G_0 \end{pmatrix}. \tag{B.6}$$

This normal form is consistent with a state-space representation in terms of position and velocity:

$$\mathcal{X} = \begin{pmatrix} \mathbf{X} \\ \dot{\mathbf{X}} \end{pmatrix}.$$

The white noise $\dot{\mathbf{w}}$ can be interpreted mechanically as a (generalised) force applied to a particle whose configuration is $\mathbf{X}$. Now, from (B.3),

$$\mathcal{X}(t_k) - \overline{\mathcal{X}} = A'(\mathcal{X}(t_{k-1}) - \overline{\mathcal{X}}) + B'\mathbf{w}_k$$

$$\text{where} \quad A' = \exp F\tau,$$

which can be written in terms of its submatrices as

$$A' = \begin{pmatrix} A'_{11} & A'_{12} \\ A'_{21} & A'_{22} \end{pmatrix}.$$

The matrix A' does not yet conform to the normal form (9.18) on page 204 for discrete coefficients A, which would require $A'_{11} = 0$ and $A'_{12} = I$. To reach the normal form,

a coordinate transformation $\mathcal{X} \to M\mathcal{X}$ with

$$M = \begin{pmatrix} I & 0 \\ A'_{11} & A'_{12} \end{pmatrix}$$

must be applied. The transformed process has $A = M^{-1}A'M$ which is in normal form, and $B = M^{-1}B'$ which is only approximately (for small τ) in the normal form for B in (9.18) on page 204.

The inverse transformation, obtaining continuous parameters F and G from discrete ones A and B requires a matrix logarithm

$$F' = \frac{1}{\tau} \log A$$

followed by a coordinate change, similar to the one above, to reach the normal form.

Power spectrum

The expression (9.13) on page 200 for a second-order power spectrum is obtained from (B.5) using a frequency analysis similar to the one in the first-order case.

B.3 Accuracy of learning

The claims concerning accuracy of learning, stated in chapter 11 on page 241, are justified here. First the proportional error in the discrete dynamical parameters a_1, a_2 and b_0 is obtained. Then this is used to derive the proportional error of the underlying continuous parameters f, β and $\overline{\rho}$.

Discrete analysis

The error of estimators $\hat{a}_1$, $\hat{a}_2$ and $\hat{b}_0$ is derived from the Fisher information measure for a maximum likelihood estimator (Kendall and Stuart, 1979). Asymptotically, for large M,

$$\mathcal{V}[b_0|\hat{b}_0]^{-1} = -\mathcal{E}\left[\partial^2 L/\partial b_0^2\right],$$

where L is the log-likelihood function (11.1) on page 237. Using (11.2) on page 237, this gives

$$\mathcal{V}[b_0|\hat{b}_0] = \frac{\hat{b}_0^2}{2(M-2)}$$

so that the proportional error in $\hat{b}_0$, denoted $\Delta\hat{b}_0$, is

$$\Delta\hat{b}_0 = \frac{\sqrt{\mathcal{V}[b_0|\hat{b}_0]}}{\hat{b}_0} = \frac{1}{\sqrt{2(M-2)}}. \tag{B.7}$$

Applying a similar analysis to the vector $\mathbf{a} = (a_1, a_2)^T$ obtains error variances and covariances for a_1 and a_2. This gives

$$\begin{aligned}
\mathcal{V}[\mathbf{a}|\hat{\mathbf{a}}]^{-1} &= -\mathcal{E}\left[\partial^2 L/\partial \mathbf{a}^2\right] \\
&= \mathcal{E}\left[\frac{1}{\hat{b}_0^2}\begin{pmatrix} r_{11} & r_{12} \\ r_{21} & r_{22} \end{pmatrix}\right] \\
&= (M-2)\mathcal{E}[\frac{1}{\hat{b}_0^2}\mathcal{X}(t)\mathcal{X}(t)^T] \\
&= (M-2)\frac{1}{\hat{b}_0^2}\mathcal{P}_\infty
\end{aligned}$$

where r_{ij} are the auto-correlation coefficients from the learning algorithm of figure 11.2 on page 238. Covariance $\mathcal{P}_\infty$ is obtained as the steady-state solution for $\mathcal{P}$ in the state equation (9.19) on page 204 for the process. After some manipulation, this gives

$$\mathcal{V}[\mathbf{a}|\hat{\mathbf{a}}] = \frac{1+\hat{a}_2}{M-2}\begin{pmatrix} 1-\hat{a}_2 & -\hat{a}_1 \\ -\hat{a}_1 & 1-\hat{a}_2 \end{pmatrix}. \tag{B.8}$$

For all 3 parameters a_1, a_2 and b_0 it seems that estimator error $\propto 1/\sqrt{M-2}$ so that a typical training sequence of 1000 video fields should lead to error of just a few percent. This is a little misleading however because small changes in a_1 and a_2 can have a substantial effect on the ARP model. Looking at continuous parameters β, f and $\overline{\rho}$ gives a clearer picture.

Continuous analysis

Making the assumption that $\beta\tau \ll 1$ then, from (9.25) on page 206,

$$\beta\tau \approx \frac{1}{2}(1+a_2)$$

so that, using (B.8),

$$\mathcal{V}[\beta\tau] \approx \frac{1}{4}\mathcal{V}[a_2] \approx \frac{1}{M-2}\hat{\beta}\tau$$

and finally

$$\Delta\hat{\beta} = \frac{\sqrt{\mathcal{V}[\beta]}}{\hat{\beta}} = \frac{1}{\sqrt{\hat{\beta}\tau}}\frac{1}{\sqrt{M-2}} \approx \frac{1}{\sqrt{\hat{\beta}T}},$$

as claimed in chapter 11. A similar analysis for $\hat{f}$ shows that

$$\Delta\hat{f} \equiv \frac{\sqrt{\mathcal{V}[f]}}{\hat{\beta}} = \frac{1}{2\pi}\frac{1}{\sqrt{\hat{\beta}T}}$$

Note that the error $\Delta\hat{f}$ in the estimated frequency is defined here relative to $\hat{\beta}$. Finally, it remains to establish an error bound for the estimated value of $\overline{\rho}$. From (9.27) on page 206, and given that $\Delta\hat{b}_0$ (see above) can be neglected,

$$\Delta\hat{\overline{\rho}} = \frac{1}{2}\Delta\hat{\beta}\left(1 + \frac{\hat{\beta}}{\pi\hat{f}\sin 2\pi\hat{f}\tau}\right).$$

Appendix C

Further shape-space models

C.1 Recursive synthesis of shape-spaces

Chapter 4 ended with a discussion about shape-spaces for articulated motion and a summary comparing, for various hinged objects, the number of degrees of freedom of the object and the dimension of linear shape-space needed to represent its motion. There is a powerful general rule, presented here, for building up the dimension of a shape-space as articulated components are tacked onto a body. The dimension is built up recursively, tacking one component on at a time. It will be convenient to write the equation of the curve in *homogeneous coordinates*, a standard geometric tool in graphics and computer vision (Faugeras, 1993; Foley et al., 1990), giving

$$\mathbf{r}_h(s) = \begin{pmatrix} x(s) \\ y(s) \\ 1 \end{pmatrix}$$

so that the curve ranges over the shape-space swept out by

$$\mathbf{r}_h(s) = T\overline{\mathbf{r}}_h(s)$$

where the template $\overline{\mathbf{r}}_h(s)$ (also in homogeneous coordinates) may be either two-dimensional or three-dimensional as appropriate and T is a linear transformation. For example, the Euclidean similarities discussed earlier are represented by the following

transformation T:

$$T\mathbf{r} = \begin{pmatrix} X_1 \\ X_2 \end{pmatrix} + \begin{pmatrix} X_3 & -X_4 \\ X_4 & X_3 \end{pmatrix} \mathbf{r}$$

which can conveniently be written as a 3×3 matrix in homogeneous coordinates:

$$T = \begin{pmatrix} X_3 & -X_4 & X_1 \\ X_4 & X_3 & X_2 \\ 0 & 0 & 1 \end{pmatrix}. \tag{C.1}$$

This is appropriate, of course, for a body moving rigidly in the plane. More generally, it applies to the end link of a series of hinged links attached to a base that moves rigidly in the plane.

The new component is attached so that it is free to be acted on by transformations T' relative to the end body. For example, a simple planar hinge is represented in homogeneous coordinates by

$$T'(\theta) = \begin{pmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

Incremental rule: now the general rule can be stated, that when the hinged component is added, the dimension of the shape-space is increased by:

$$\dim\{T, TT'\}^+ - \dim\{T\}^+, \tag{C.2}$$

where $\{T\}^+$ is the vector space of transformations T and $\{T_1, T_2\}^+$ denotes the vector space spanned by the two transformations taken jointly (simply concatenating the elements of T_1 and T_2 into one vector). To make this clear, we will work through the rule using the examples for T and T' given above of a planar base element moving rigidly in the plane with a single hinged component.

Planar rigid body with hinged appendage

First of all, $\dim\{T\}^+ = 4$, clearly, since T has 4 independent linear parameters $X_1, \ldots, X_4$. Next we compute TT':

$$TT' = \begin{pmatrix} \gamma & -\delta & X_1 \\ \delta & \gamma & X_2 \\ 0 & 0 & 1 \end{pmatrix} \tag{C.3}$$

where $\gamma = X_3 \cos\theta - x_4 \sin\theta$ and $\delta = X_4 \cos\theta + X_3 \sin\theta$ so now

$$\dim\{T, TT'\}^+ = \dim\{X_1, X_2, X_3, X_4, \gamma, \delta\}^+ = 6$$

so that adding the hinge increases the dimension of shape-space by

$$\dim\{T, TT'\}^+ - \dim\{T\}^+ = 6 - 4 = 2$$

and the dimension of the new space is increased from 4 to 6.

Further hinged appendages

Now suppose we want to add a further hinged appendage. If it is added to the main body (figure C.1a), the argument above is unchanged (it is not in the least affected by the existence of the previous appendage) and the increase in dimension is still 2. Now the total dimension of the shape-space increases to 8. If instead the new appendage is tacked onto the end of the previous appendage (figure C.1b) we can apply the general method as follows. The matrix TT' in (C.3) above becomes the new T, and now

$$T' \equiv T'(\phi) = \begin{pmatrix} \cos\phi & -\sin\phi & 0 \\ -\sin\phi & \cos\phi & 0 \\ 0 & 0 & 1 \end{pmatrix},$$

where ϕ is the angle of the latest appendage. Now the argument proceeds exactly as for the addition of the first appendage except that we have γ, δ and ϕ where before we had X_3, X_4 and θ, so again the subspace dimension is increased by 2. Alternatively, a quicker way to get to the same conclusion, is simply to note that the shape-space of the

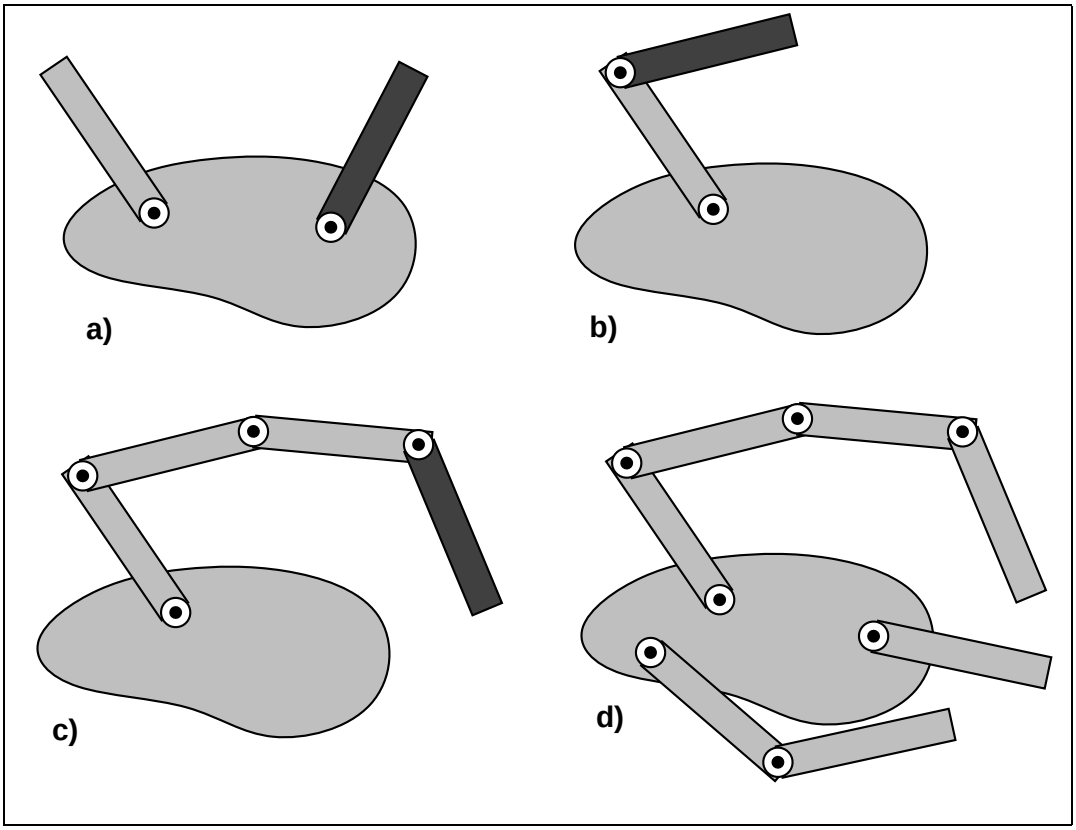

Figure C.1: Hinged appendages *Hinging an appendage onto: a) the main body; b) the end of another appendage; c) the end of a chain of appendages. d) A general planar articulated body consisting of a base in rigid motion with n hinged bodies.*

first appendage, considered in isolation, is the space of Euclidean similarities. (It can execute any rigid motion given that it is hinged to a base that can execute any rigid motion.) Therefore we are simply solving again the problem of computing the increase in shape-space dimension when an appendage is hinged to a base whose shape-space is the Euclidean similarities. Clearly we could continue to add appendages to a chain (figure C.1c), adding 2 to the shape-space dimension each time.

Now a simple inductive argument shows the following rule. A rigid planar body with n hinged appendages (figure C.1d) has a shape-space with dimension $4 + 2n$.

This is true regardless of how the hinges are arranged, provided there is no *closed* kinematic chain (sequence of hinged bodies forming a loop).

Adding telescopic appendages

How is the dimension of shape-space affected if an appendage is added with a "prismatic" or telescopic joint? In that case

$$T' \equiv T'(d) = \begin{pmatrix} 1 & 0 & d \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix},$$

where d is the variable length by which the joint is extended and taking T for Euclidean similarities of the base object as in (C.1), gives

$$TT' = \begin{pmatrix} X_3 & -X_4 & X_3 d + X_1 \\ X_4 & X_3 & X_4 d + X_2 \\ 0 & 0 & 1 \end{pmatrix},$$

so that

$$\dim(\{T, TT'\}^+) = \dim(\{X_1, X_2, X_3, X_4, X_3 d, X_4 d\}^+) = 6$$

(since $X_3 d, X_4 d$ extend the basis by two elements — even though there is only one new degree of freedom d, it appears non-linearly and requires two degrees of freedom to represent linearly). Just as in the hinged case therefore, each telescopic appendage to a rigid body also raises the dimension of shape-space by two.

Planar body with co-planar appendage, in three dimensions

From earlier discussion in chapter 4, we know that images of a planar body in three dimensions form an affine space, so that $\{T\}^+$ is the usual 6-dimensional planar affine space. Unfortunately, unlike rigid planar bodies in which hinged appendages cost only 2 degrees of freedom each, in the planar affine case they come relatively expensively. Each requires 4 degrees of freedom so that insisting on linear parameterisation is relatively costly.

The argument, applying the incremental rule (C.2) is as follows. Transformations are

$$T = \begin{pmatrix} X_3 & X_6 & X_1 \\ X_5 & X_4 & X_2 \\ 0 & 0 & 1 \end{pmatrix} \quad \text{and} \quad T' = \begin{pmatrix} \cos\phi & -\sin\phi & 0 \\ -\sin\phi & \cos\phi & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

and after a little calculation

$$\dim(\{T, TT'\}^+) = \dim(\{X_1, X_2, ((X_n \cos\phi, X_n \sin\phi), n = 3, \ldots, 6)\}^+) = 10,$$

an increase of 4 over the 6 affine degrees of freedom.

Proof of incremental rule

For completeness, a proof of the rule (C.2) for incrementing the dimension of shape-space is included here. Consider a base shape γ which we would normally represent by a template either as a parameterised curve $\overline{\mathbf{r}}(s)$ in two dimensions or $\overline{\mathbf{R}}(s)$ in three dimensions, or as a control point vector $\overline{\mathbf{Q}}$ in two or three dimensions. It is subject to linear transformations T onto the image plane, parameterised (not necessarily linearly) by a parameter set λ, giving a set of image shapes $\{T(\lambda)\gamma,\ \lambda \in \Lambda\}$. This set spans a vector space denoted by the closure $\{T(\lambda)\gamma,\ \lambda \in \Lambda\}^+$ which generally (for non-degenerate γ) is isomorphic to the closure of the space of transformations $\{T(\lambda),\ \lambda \in \Lambda\}^+$, regardless of the particular shape γ.

Next, it is assumed that the body to which an appendage is about to be added is already articulated so that, already attached to the base, are a set of components transformed relative to the base by a set of linear transformations $T_n(\lambda_n),\ \lambda_n \in \Lambda_n$ for $n = 1 \ldots N$. The n^{th} component is thus transformed into the image plane by the transformation $TT_n(\lambda_n)$. Finally, the appended component is attached via $T'(\lambda'),\ \lambda' \in \Lambda'$. It is assumed that the parameters $\lambda_1, \ldots, \lambda_N, \lambda'$ are all independent — the hinging/telescopic actions of the individual components are not coupled and this is where closed kinematic chains are excluded. We also assume that all components are non-degenerate so that we can continue to consider the transformations only and drop any reference to the component shapes themselves.

The independence of parameters for components means that

$$\dim\{T, TT'\}^+ + \dim\{T, TT_1, \ldots, TT_N\}^+ = \dim\{T, TT_1, \ldots, TT_N, TT'\}^+ + \dim\{T\}^+$$

(omitting for simplicity explicit reference to parameters λ, $\lambda_1 \ldots \lambda_n$ and λ') and this is simply rearranged into a formula for the increase in shape-space dimension:

$$\dim\{T, TT_1, \ldots, TT_N, TT'\}^+ - \dim\{T, TT_1, \ldots, TT_N\}^+ = \dim\{T, TT'\}^+ - \dim\{T\}^+,$$

the right hand side of which is the required formula (C.2), simplified in that it involves only the base and the new component.

Silhouettes

For smooth silhouette curves, it can be shown that a shape-space of dimension 11 is appropriate. This shape-space representation of the curve is an approximation, valid for sufficiently small changes of viewpoint. The proof of this result follows from results in the computer vision literature about the projection of silhouettes into images (Giblin and Weiss, 1987; Blake and Cipolla, 1990; Vaillant, 1990; Koenderink, 1990; Cipolla and Blake, 1992b).