

# Approximating and Distinguishing Ostracoda by the Use of B-Splines

Walter Neubauer<sup>1</sup>, Johann Linhart<sup>2</sup>

<sup>1</sup> Unterfeldstraße 13/10, A-5101 Bergheim (E-Mail: mathstud@gmx.at).

<sup>2</sup> Department of Mathematics, University of Salzburg, Hellbrunnerstraße 34, A-5020 Salzburg  
(E-Mail: johann.linhart@sbg.ac.at).

## 1. Preface

Biological outlines may be investigated by various mathematical models. There is a well-developed theory of shape (Zelditch 2004) for the case of sufficiently many good landmarks. For the morphometrics of outlines with few or no landmarks sometimes simple polygonal curves (Bézier curves) have been used (see for instance Loy 2000), but this may lead to polynomials of rather high degree, which tend to oscillate in an undesirable way. This is a well-known problem in many technical applications, especially in computer aided geometric design (CAGD), where it is usually overcome by the use of splines (Hoschek and Lasser 1993, Farin 1990).

Splines consist of several polynomial curves of low degree that smoothly fit together. Two of the most widely used splines are Bézier splines and B-splines. These are not different curves, but only different representations of the same curves; they may be transformed into one another (Hoschek and Lasser 1993). In both cases the shape of the curve is determined by a list of geometrically meaningful control points, but for B-splines fewer control points are needed.

The name B-spline was coined by Isaac Jacob Schoenberg and it is used as an abbreviation for basic spline (de Boor 1978, Farin 2001). The authoritative person for the development of the theory of B-spline curves and surfaces was Carl de Boor due to his researches at General Motors.

In the 50s and 60s of the last century the automobile industry in particular had to face the difficulty that freeform curves and surfaces couldn't be exactly reproduced owing to a lack of a proper mathematical description. Carl de Boor solved this problem by depicting the shape of component parts as parametric curves, defined piecewise by polynomials.

As a result, the curve's shape is determined by its so-called control points or de-Boor points, which are the vertices of the control polygon. It turned out that this way of characterizing curves and shapes was advantageous for other fields and generalized versions and enhancements of B-splines were developed shortly after. Representing curves and shapes using B-splines and their further development is still employed in Computer-Aided-Design systems. Nowadays the areas of application are manifold and diversified and reach far beyond technical mould design and construction (Bartels 1987, Farin 2001, Hoschek and Lasser 1993).

For our task, the approximation of outline data, B-spline curves offer several advantages. B-spline curves are invariant under affine transformations, the pixel data can be approximated by a numerically stable and accurate algorithm and, we obtain an enormous data reduction. The primary outline data, consisting of approximately 1000 to 1400 pixels, can be excellently depicted by a B-spline curve determined by just 16 control points. A further important fact to mention in advance is a property called "local control". Thereby the shifting of a single control point of the B-spline curve does not cause the change of the entire curve progression but just a deviation in the surrounding of the concerning control point. This characteristic has proved very useful for the examination of morphological structures just having an effect on single parts of the outline.

This article gives an overview of the theoretical background of approximating B-spline curves, following the elaborations in Bartels (1987), de Boor (1978), Hoschek and Lasser (1993), Piegl (1995). A detailed description of the algorithm for the approximation of ostracods' outlines is presented in Bayer et al. (2002) and Neubauer (2007). The shape of a carapace offers the possibility of a fast and stable computation. Moreover, the area deviation provides us with demonstrative results for the application in palaeontology since a perspicuous graphical representation of the stated area is feasible and the output is in units of square micrometers.

The last section deals with a method of distinguishing two populations of ostracods by using the data of a distance matrix, which frequently finds application in biological research.

## 2. B-Spline Representation

### 2.1. B-Spline Curves

**Definition 2.1.** A  $p$ th-degree B-spline curve is defined by

$$\mathbf{C}(t) = \sum_{i=0}^n N_{i,p}(t) \mathbf{P}_i \quad a \leq t \leq b.$$

The  $\mathbf{P}_i$  are the control points, and the  $N_{i,p}(t)$  are the  $p$ th-degree B-spline basis functions defined by

$$N_{i,0}(t) = \begin{cases} 1 & \text{if } u_i \leq t < u_{i+1} \\ 0 & \text{otherwise} \end{cases}$$

$$N_{i,p}(t) = \frac{t - u_i}{u_{i+p} - u_i} N_{i,p-1}(t) + \frac{u_{i+p+1} - t}{u_{i+p+1} - u_{i+1}} N_{i+1,p-1}(t),$$

where  $U = (u_0, \dots, u_m)$  is a nondecreasing sequence of real numbers, i.e.,  $u_i \leq u_{i+1}$ ,  $i = 0, \dots, m-1$ , called knot vector. The items  $u_i$  are called knots and the polygon formed by the  $\mathbf{P}_i$  is the control polygon.

Particularly we use B-spline curves of degree  $p = 2$ , also called quadratic B-spline curves, for the approximation of ostracods' outlines. Their basis functions are, already calculated, given by

$$N_{i,2}(t) = \begin{cases} \frac{(t-u_i)^2}{(u_{i+2}-u_i)(u_{i+1}-u_i)} & \text{for } u_i \leq t < u_{i+1} \\ \frac{(u-u_i)(u_{i+2}-t)}{(u_{i+2}-u_i)(u_{i+2}-u_{i+1})} + \frac{(u-u_{i+1})(u_{i+3}-t)}{(u_{i+3}-u_{i+1})(u_{i+2}-u_{i+1})} & \text{for } u_{i+1} \leq t < u_{i+2} \\ \frac{(u_{i+3}-t)^2}{(u_{i+3}-u_{i+1})(u_{i+3}-u_{i+2})} & \text{for } u_{i+2} \leq t < u_{i+3} \end{cases}$$

and  $N_{i,2}(t) = 0$  for  $t$  outside the interval  $[u_i, u_{i+3})$ . In these formulas, division by zero may occur. When this is the case, the result of the division is set equal to zero.

A common choice for the knot vector of  $p$ th-degree basis functions is setting the first  $p+1$  knots 0, the last  $p+1$  knots 1 and the interior knots equally spaced;

$$u_0 = u_1 = \dots = u_p = 0$$

$$u_j = \frac{j-p}{m-2p} \quad \text{for } j = p+1, \dots, m-p-1$$

$$u_{m-p} = \dots = u_m = 1.$$

This so-called uniform knot vector has, referring to our application with  $p = 2$ , the form

$$U = \{0, 0, 0, \frac{1}{m-4}, \dots, \frac{m-5}{m-4}, 1, 1, 1\}.$$

As an example, for  $n = 6$  the basis functions of degree 2 generated by an uniform knot vector are depicted below (fig. 1).

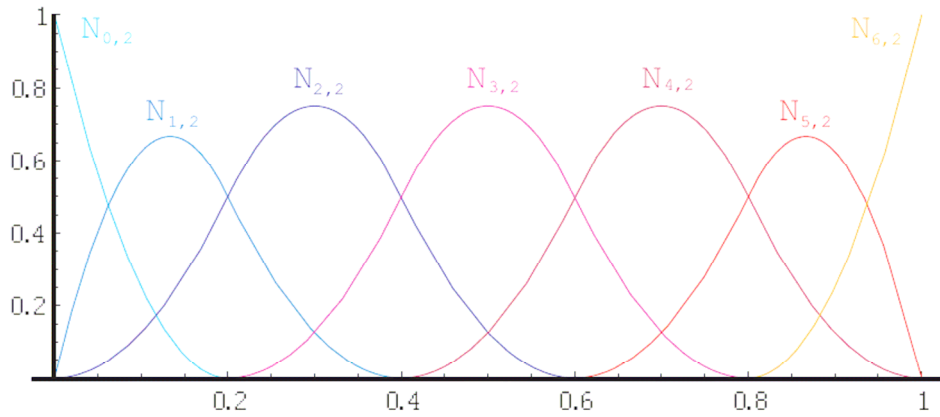


Figure 1: The nonzero second-degree basis functions generated by a knot vector  $U = (0,0,0,1/5,2/5,3/5,4/5,1,1,1)$ .

To give an introductory idea of B-spline techniques we contemplate the following example. Let  $U$  be a uniform knot vector as defined in the equation above,  $p = 2$  and the set of control points

$$\{\mathbf{P}_i\} = \{(-1, 0), (-0.9, 1), (-0.2, 1.3), (-0.5, -1), (0.7, -1), (0.1, 0.5), (1, 1)\}.$$

Figure 2 shows the resulting B-spline curve.

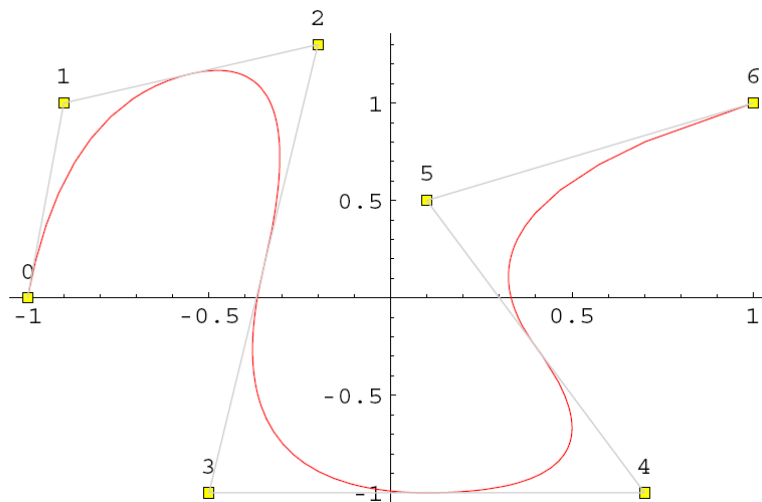


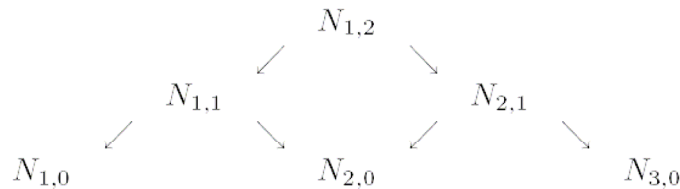
Figure 2: B-spline curve using the conditions of the example above.

Below a number of properties of B-spline curves following mostly from the definition 2.1 are given.

**Property 2.2. Endpoint interpolation:**  $C(0) = \mathbf{P}_0$  and  $C(1) = \mathbf{P}_n$ .

**Property 2.3. Affine invariance:** An affine transformation, including translation, rotation and scaling, is applied to the curve by applying it to the control points.

**Property 2.4. Local support:** Moving  $\mathbf{P}_i$  changes  $C(t)$  only in the interval  $[u_i, u_{i+p+1})$  (fig. 3). This follows from the fact that  $N_{i,p}(t) = 0$  for  $t \notin [u_i, u_{i+p+1})$ , since  $N_{i,p}(t)$  is a just linear combination of  $N_{i,0}(t), \dots, N_{i+p,0}(t)$  (see definition 2.1) and those zero-degree basis functions are 0 outside  $[u_{i+m}, u_{i+m+1})$ ,  $m = 0, \dots, p$ . The triangular scheme illustrates this fact for the basis function  $N_{1,2}(t)$ .



$N_{1,2}(t)$  is a combination of  $N_{1,0}(t)$ ,  $N_{2,0}(t)$ , and  $N_{3,0}(t)$ . Thus,  $N_{1,2}(t)$  is nonzero only for  $t \in [u_1, u_4)$ .

Conversely, in any given knot span  $[u_j, u_{j+1})$  at most  $p+1$  of the  $N_{i,p}(t)$  are nonzero, namely the functions  $N_{j-p,p}, \dots, N_{j,p}$ .

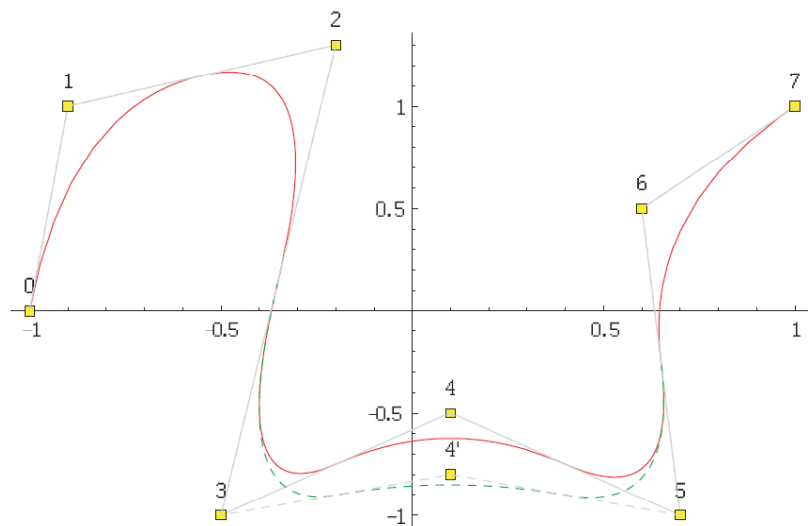


Figure 3: A curve with degree  $p = 2$  on  $U = (0, 0, 0, 1/6, 2/6, 3/6, 4/6, 5/6, 1, 1, 1)$ ; moving  $\mathbf{P}_4$  to  $\mathbf{P}_4'$  changes the curve in the interval  $[2/6, 5/6)$ .

Moving along the curve from  $t = 0$  to  $t = 1$ , the functions  $N_{i,p}(t)$  act like switches. As the parameter  $t$  moves past a knot, one  $N_{i,p}(t)$  and the corresponding  $\mathbf{P}_i$  switches off and  $\mathbf{P}_{i+p+1}$  switches on.

**Property 2.5.** The B-spline curve tangents each segment of the control polygon.

## 2.2. Approximation to Outline Pixel Data with B-Spline Curves

In this section we study the construction of B-spline curves that should fit a rather arbitrary set of geometric data, such as pixels of an ostracod's outline, following the elaborations in Hoschek and Lasser (1993), Piegler (1995), Bayer et al. (2002), Deuffhard and Hohmann (2002). The aim is to construct curves that do not necessarily satisfy the given data in an exact way, but only approximately. In some applications - such as ours - a large number of points is generated, which can contain measurement errors or computational noise. In this case, it is important for the curve to capture the “shape” of the data, but not to “wiggle” its way through each single point.

Given is an array of sequenced pixels  $\{\mathbf{Q}_k\}$ ,  $k = 0, \dots, l$ , which we want to approximate with a second-degree B-spline curve. If we assign a parameter value  $t_k$  to each pixel  $\mathbf{Q}_k$ , and select the knot vector  $U = (u_0, \dots, u_m)$  to be uniform, we can set up a system of linear equations

$$\mathbf{Q}_k = \mathbf{C}(t_k) = \sum_{i=0}^n N_{i,p}(t_k) \mathbf{P}_i$$

with  $l+1$  equations, where the control points  $\mathbf{P}_i$  are the  $n+1$  unknowns. The choice of the parameter values  $t_k$  enormously affects the shape of the curve. An adequate method provides the chordal parameterisation. It assigns the parameter values  $t_k$  proportionally to the total length of the outline. Let  $d$  be the total chord length

$$d = \sum_{k=1}^l \|\mathbf{Q}_k - \mathbf{Q}_{k-1}\|.$$

The parameter values are given by

$$\begin{aligned} t_0 &= 0 \\ t_k &= t_{k-1} + \frac{\|\mathbf{Q}_k - \mathbf{Q}_{k-1}\|}{d} \quad k = 1, \dots, l-1 \\ t_l &= 1. \end{aligned}$$

Typically, it is necessary to find a B-spline curve approximating a large number of points. In general, the number of pixels  $l$  to approximate is much higher than the number of desired control points  $n$ . Hence, the system of equations is overdetermined and we will not get an exact solution.

The best way to overcome this difficulty is to solve the system in the sense of minimizing the sum of the squared differences between the given set of pixels  $\{\mathbf{Q}_k\}$  and the appropriate values of the B-spline curve  $\mathbf{C}(t_k)$ ,

$$\sum_{k=0}^l \|\mathbf{C}(t_k) - \mathbf{Q}_k\|^2 = \sum_{k=0}^l \left\| \sum_{i=0}^n N_{i,p}(t_k) \mathbf{P}_i - \mathbf{Q}_k \right\|^2 \longrightarrow \min.$$

There are manifold possibilities to get a solution for this least square problem. A numerical stable computation of the minimizing problem results from the pseudo-inverse matrix gathered from the singular value decomposition of the system matrix (see Deuffhard and Hohmann 2002, Hogben 2007, Strang 1998). This method is implemented in MORPHOMATICA to approximate outlines.

### 3. Implementation of the Formal Methods to Ostracod Outlines

#### 3.1. Data Structure

A photograph of an ostracod's valve is taken under a microscope. Afterwards, the so-called Tps-dig (Rohlf 2001) saves the outline in a data set. This program creates a file with the pairs of coordinates of the outlining pixels of the picture. Also other information, such as potential landmarks and the file name of the picture, is saved in this file. The illustration below schematically shows the structure of such a file.

```

LM=0
OUTLINES=1

POINTS=1387
630 78
631 79
631 80
632 81
633 82
...
627 74
627 75
628 76
629 77

IMAGE=J001.tif
ID=J001

```

The file describes an outline with no specified landmarks ( $LM = 0$ ) and one contour ( $OUTLINES = 1$ ) with 1387 pixels ( $POINTS = 1387$ ).

### 3.2. Centre of Gravity and Axes of Inertia

The positions of the valves on the pictures vary widely and the tps-files do not tag any basing points. Nevertheless, a meaningful comparison of two valves should be independent of alignment and position in the picture. Therefore, to compare the shape of two outlines, they first have to be superimposed. Ideally, this should be done in such a way that the “difference” between them is as small as possible. But this would be a very difficult task, so we choose to position the two outlines in such a way that the centroids and the main axes of inertia coincide.

The structure of the tps-data suggests computing the centroid based on the pixels using the arithmetic mean of the coordinate vectors of the points  $\mathbf{Q}_k$ , that is

$$\mathbf{S} = \frac{1}{l} \sum_{k=0}^l \mathbf{Q}_k.$$

But this is the centroid of these points and not of the whole outline. Consequently, this only makes sense if the points are distributed very uniformly. If parts of the outline are a little bit rugged or jagged, there will be relatively many points  $\mathbf{Q}_k$  concentrated in these parts and  $\mathbf{S}$  tends to move towards them. So the centroids of two very similar outlines may be rather different, if only one has some rugged parts. Figure 4 illustrates this difficulty.



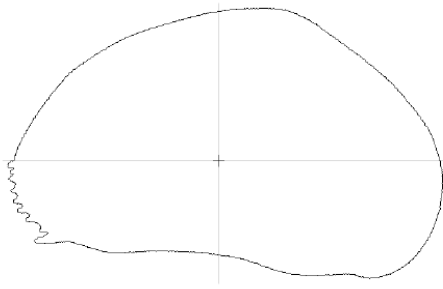


Figure 4a: Centroid and main axes computed with the outline points.

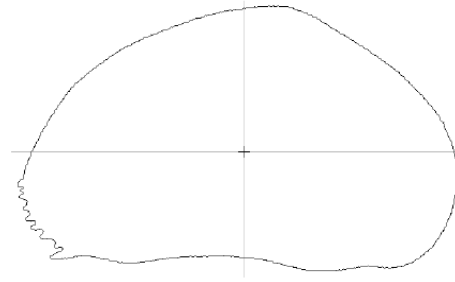


Figure 4b: The origin now is the centre of gravity of the domain and the axes correspond with the main axes of inertia.

This is why the above centroid should be replaced by the centre of gravity of the domain  $A$  surrounded by the given outline. This centre is defined by

$$\mathbf{S} = (s_x, s_y) = \frac{1}{a} \int_A (x, y) dx dy,$$

where  $s_x$  and  $s_y$  denote the coordinates of  $\mathbf{S}$ , and  $a$  is the area of the domain  $A$ . It turns out that  $\mathbf{S}$  can be computed in the following rather simple way, if  $x_k$  and  $y_k$  denote the coordinates of  $\mathbf{Q}_k$ , and the points are given in counterclockwise order,

$$s_x = -\frac{1}{6a} \sum_{i=0}^n (y_{i+1} - y_i) (x_{i+1}^2 + x_i x_{i+1} + x_i^2).$$

$s_y$  is computed in a similar way with  $x$  and  $y$  interchanged and the whole expression multiplied by  $-1$ . Of course,  $(x_{l+1}, y_{l+1})$  is understood to be equal to  $(x_1, y_1)$ .

Analogical to the centroid, the axes of inertia should also be calculated not only for the points, but for the whole outline or, more precisely, for the domain surrounded by the outline. The moment of inertia with regard to a certain axis (passing through the origin) is defined to be the integral of the squared distance from this axis, taken over the considered domain. It may be computed in a similar, but somewhat more complicated way as the centre of gravity above.

If the moment of inertia is not equal for all directions of axes, there is a unique direction, which yields the minimum moment, and this is taken to be the new  $x$ -axis. This direction is given by an eigenvector of a certain 2 by 2 matrix and thus it is not difficult to compute.

The new  $y$ -axis is, of course, perpendicular to the new  $x$ -axis and corresponds to the maximum moment of inertia.

Finally, we have to define a congruence transformation  $f$ , which moves the points of the contour as mentioned above. Shifting  $\mathbf{S}$  to the point  $\mathbf{Y} = (0,0)$  and rotating the vectors  $\mathbf{E}_1 = (e_{11}, e_{12})$ , specifying the axis with minimum moment of inertia, and  $\mathbf{E}_2 = (e_{21}, e_{22})$ , specifying the axis with maximum moment, to  $(1,0)$  resp.  $(0,1)$  can be done by a transformation

$$f(\mathbf{X}) = \begin{pmatrix} e_{11} & e_{12} \\ e_{21} & e_{22} \end{pmatrix} (\mathbf{X} - \mathbf{S}).$$

Applying this transformation to each point moves the contour into the desired position.

### 3.3. Approximation to Contour Data

To prepare the point data for a good and meaningful approximation, it is necessary to divide the contour into two halves. If the B-spline approximation is applied in a straightforward way to an outline, it may happen that two very similar outlines lead to rather different control points. This is the case, for instance, with the two artificial elliptical outlines of figure 5. One should note that this phenomenon occurs due to the fact that moving the control points simultaneously in a suitable way around the curve has only little influence on the shape of the curve.

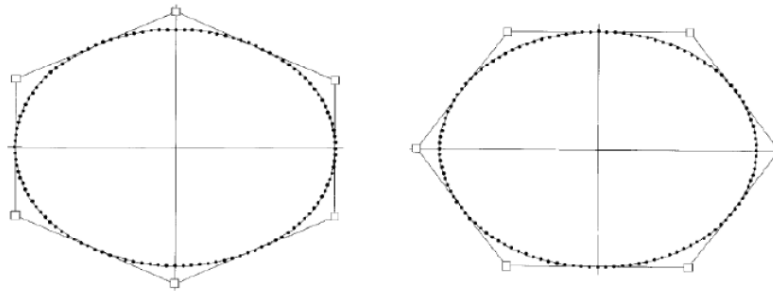


Figure 5: Two rather similar elliptical outlines with different control points.

To avoid this problem, we cut the outline in two pieces and approximate each half separately under the condition that the resulting curves fit together.

After the standardisation this can easily be done by using the  $x$ -axis as dividing line. Concretely, all points  $\mathbf{P}_i = (x_i, y_i)$  with  $x_i > 0$  will be assigned to the dorsal region and all points with  $x_i < 0$  to the ventral region. Therefore we must assume that the contour crosses the  $x$ -axis

at most at two points, what is usually the case for ostracods' outlines (fig. 6). Hence there are only two pairs of consecutive points in the contour where  $x_i \geq 0$  and  $x_{i+1} < 0$ .

We pass a line through these points, determine the intersection points with the  $x$ -axis and add them to the contour. These intersections will be the starting and end points of the approximating B-spline curves.

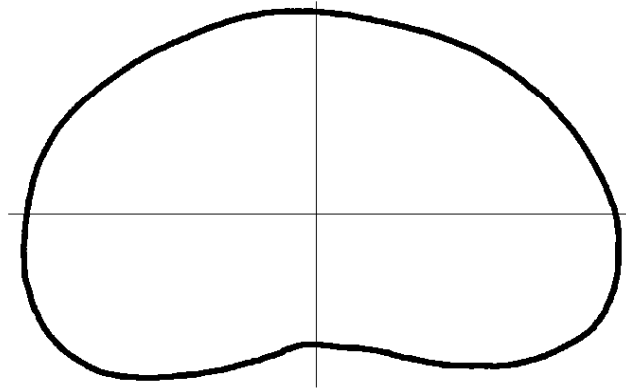


Figure 6: Ostracod contour after standardisation.

The outlining points of the respective region get numbered consecutively clockwise and are approximated by a B-spline curve of degree  $p = 2$  with a uniform knot vector and parameter value gathered by a chordal parameterisation. The procedure is explained in sections 2.1 and 2.2. Figure 7 plots the contour data of an ostracod with its approximating B-spline curve.

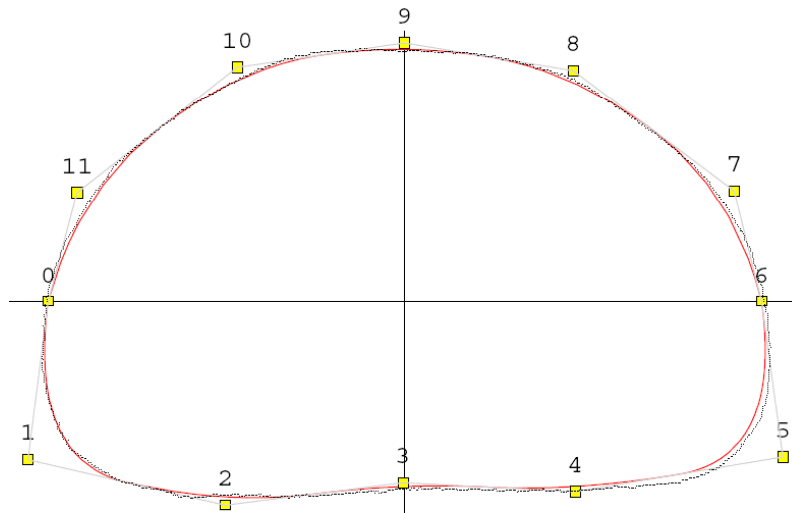


Figure 7: Ostracod contour approximated by a second-degree B-spline curve and its control polygon.

However, this approximation scheme does not bring an optimum result. A better solution can be achieved if the vector  $\mathbf{C}(t_i) - \mathbf{Q}_i$ ,  $t_i$  indicating the corresponding chord length parameter of  $\mathbf{Q}_i$ , is perpendicular to the curve. Therefore we alter the parameter  $t_i$  iteratively by adding a value  $\lambda_i$  until  $\mathbf{C}(t_i) - \mathbf{Q}_i$  is roughly perpendicular to the tangent  $\mathbf{C}'(t_i)$ .

The comparison in figure 8 shows the desired result. After 6 iterations the approximating B-spline curve approaches the contour data in an observable better way.

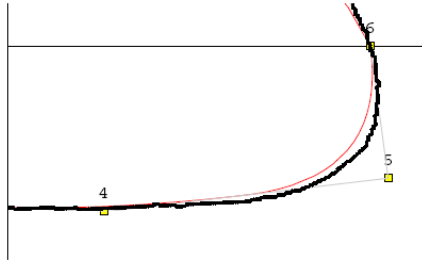


Figure 8a: Before adjusting the parameter.

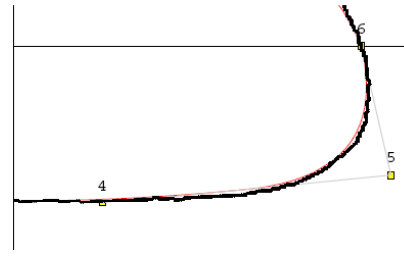


Figure 8b: After 6 iterative steps a better approach is guaranteed.

To compare biological aspects of ostracoda it is sometimes useful to adjust the sizes of the standardized outlines. This is essential for comparing, e.g., ostracods of different age.

Transforming the control points of the B-spline curve in such a way that the end points  $\mathbf{P}_0 = (x_0, 0)$ ,  $\mathbf{P}_n = (x_n, 0)$ , lying on the  $x$ -axis, get the coordinates  $\mathbf{P}_0 = (-1, 0)$ ,  $\mathbf{P}_n = (1, 0)$  is one possibility. This method has one slight shortcoming. The centre of gravity shifts out of the origin, what can cause difficulties if the contour is rather anomalous or pear-shaped.

We obtain a better solution if we use a transformation

$$s(\mathbf{X}) = \sqrt{a}\mathbf{X},$$

where  $a$  is the ratio of the areas  $a_1$ ,  $a_2$  enclosed by the B-spline curves,

$$a = \frac{a_1}{a_2}.$$

This method keeps the centre of gravity in the origin and guarantees a better comparability.

## 4. Distinguishing Outlines

### 4.1. Distance of the Corresponding Control Points

At first sight, the following concept of a distance between two B-spline curves seems to be quite natural.

**Definition 4.1.** Let  $\mathbf{C}$ ,  $\mathbf{D}$  be two B-spline curves with control point sequences  $\mathbf{P}_0, \dots, \mathbf{P}_n$  resp.  $\mathbf{Q}_0, \dots, \mathbf{Q}_n$ . We define the distance between  $\mathbf{C}$  and  $\mathbf{D}$  as the square root of the sum of all squared Euclidean distances between the corresponding control points divided by the number of control points

$$d(\mathbf{C}, \mathbf{D}) := \frac{1}{n} \sqrt{\sum_{i=0}^n \|\mathbf{P}_i - \mathbf{Q}_i\|^2}.$$

Defining the difference between two B-spline curves in this way has a number of coherent reasons. First of all, two superimposed B-spline curves are identical if their corresponding control points coincide. Property 2.3 (Affine invariance) implies that a translation or rotation is applied to the curve by applying it to the control points. Both indicate a reasonable measure. A further evidence provides the attribute that the control polygon represents a kind of approximation to the B-spline curve. So, to some extent, the control points describe the shape of the curve (cf. Baltanás et al. 2003).

A further advantage of distinguishing B-spline curves by using the distance of the corresponding control points arises from property 2.4 (Local support). A single control point takes effect just on a part of the B-spline curve. By measuring the Euclidean distance of two corresponding control points we should be able to determine whether the respective regions differ significantly or not.

Moreover, our measure  $d$  is of special interest for users from the field of biology or palaeontology, since it is a tangible, intelligible and easy to visualize tool. The computing time is extremely short and the approximation of the outline pixel data generates a unique sequence of control points. Their distances may have an explanatory power of the difference in the curves shape.

Nevertheless, the computation of the difference of two superimposed B-spline curves with the measure defined above has some shortcomings, which are discussed in this section. At first, this distance function does not induce a metric. This is easy to comprehend by imaging a B-spline curve whose control points are located on a straight line.

The resulting B-spline curve corresponds with this line. After moving one or several control points along the line, our measurement  $d$  gets positive, but the B-spline curve is still the same line.

Additionally, the measure  $d$  only considers the magnitude of the difference vector between two corresponding points, but not the direction. To comprehend the effect of shifting a control point, we examine the following testing arrangement. Let  $\mathbf{P}_0, \dots, \mathbf{P}_4$  be control points put on a semicircle of radius 1 in uniformly distributed angles, that is

$$\mathbf{P}_0 = (-1, 0) \quad \mathbf{P}_1 = \left(-\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}\right) \quad \mathbf{P}_2 = (0, 1) \quad \mathbf{P}_3 = \left(\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}\right) \quad \mathbf{P}_4 = (1, 0).$$

By selecting various vectors  $\mathbf{V}_j$  with the same length  $\|\mathbf{V}_j\| = a$  for all  $j$  and adding them to control point  $\mathbf{P}_2$ , we obtain a B-spline curve  $\mathbf{C}_j(t)$  for each vector. Owing to the measure defined above,

$$\begin{aligned} d(\mathbf{C}, \mathbf{C}^j) &= \frac{1}{n} \sqrt{\sum_{i=0}^n \|\mathbf{P}_i - \mathbf{P}_i^j\|^2} \\ &= \frac{1}{5} \sqrt{\|\mathbf{P}_2 - \mathbf{P}_2^j\|^2} \\ &= \frac{a}{5} \end{aligned}$$

for any vector of length  $a$ , although, intuitively speaking, some of the curves deviate much more from  $\mathbf{C}(t)$  than others (fig. 9).

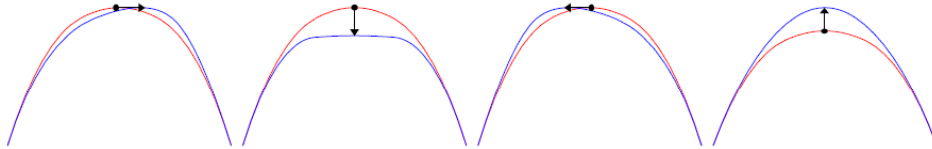


Figure 9: The primal B-spline curve (red) and various B-spline curves (blue) generated by adding different vectors with the same length to a control point.

This difficulty is of particular importance if we adjust several control points. Each point of the curve is determined by at least 3 control points (for degree  $\geq 2$ ), see property 2.4. Thereby, neighbouring control points can be positioned in such a way that an almost identical curve emerges. Figure 5 shows this effect. The left B-spline curve results from the right curve by rotating the control points by  $30^\circ$ . It is obvious, that both curves are nearly similar, but our difference measure yields a high value of dissimilarity.

## 4.2. Area Deviation

The B-spline curves we use to approximate ostracods' outlines feature some “nice” characteristics. We want to use these characteristics for computing a demonstrative and tangible measure to distinguish contours. Our approximating B-spline curves have no loops, self-intersections or other anomalies. We obtain solely so-called simply-closed curves given by

**Definition 4.2.** A curve  $\mathbf{C}(t)$ ,  $t \in [a, b]$ , in the plane is called simply-closed if it has no self-intersections,

$$\mathbf{C}(t_1) \neq \mathbf{C}(t_2) \text{ for } t_1 \neq t_2, t_1, t_2 \in [a, b],$$

and the endpoints coincide,

$$\mathbf{C}(a) = \mathbf{C}(b).$$

Accordingly, a simply-closed curve  $\mathbf{C}$  constitutes a bounded part of the plane  $A$  surrounded by the curve.  $A$  can also be seen as the interior of  $\mathbf{C}$ . To distinguish two simply-closed curves  $\mathbf{C}$  and  $\mathbf{D}$  with their interiors  $A$  and  $B$ , we consider the area of the part of the plane, which is contained in exactly one of the domains  $A$  and  $B$  and may be viewed as the area “between” the outlines. This corresponds to the area of the symmetric difference  $\mathbf{A} \Delta \mathbf{B}$ , which is called the area deviation of  $A$  and  $B$ , illustrated in figure 10.

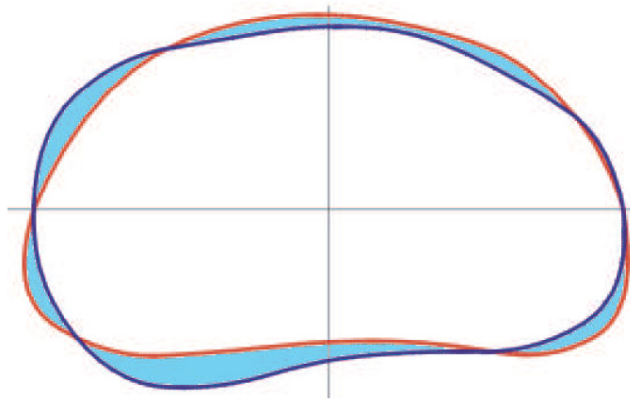


Figure 10: The light blue area is the area deviation of two superimposed simple-closed B-spline curves.

The area deviation offers several advantages for the use in ostracodology. This common and very natural measure is demonstrative and tangible; the resulting differences are in square micrometers, what doubtless contributes to a better conceivability. Furthermore, the measure is rather inured to possible data errors and measuring faults.

As mentioned above, ostracods' outlines feature a good characteristic for a fast computation without difficulties. In general, the approximating polygons are not convex, but the angles between the  $x$ -axis and a straight line from the origin to the vertices are in ascending order. This fact makes it possible to reduce the number of computations for possible points of intersection substantially.

Every specimen is approximated by two open B-spline curves describing two regions, a dorsal and a ventral one, fixed by the main axes of inertia with the minimum moment. To calculate an approximative value for the enclosed area "between" the B-spline curves of two specimen-halves, in the present implementation 51 points on each curve, corresponding to equally spaced parameter values, are computed. We denote them with  $\mathbf{C}_i$  for the first curve and  $\mathbf{D}_j$  for the second ( $i, j = 0, \dots, 51$ ). This yields a polygon with 50 line segments substituting each B-spline curve, with  $S_i = \mathbf{C}_i\mathbf{C}_{i+1}$  denoting the segments of the first polygon and  $T_j = \mathbf{D}_j\mathbf{D}_{j+1}$  denoting the segments of the second.

To evaluate the area deviation, the points of intersection of the two superimposed polygons are of importance. In principle, it would be necessary to determine the points of intersection of each segment of the first polygon with each segment of the second polygon, what requires  $50 \times 50 = 2500$  comparisons for each specimen-half. This goes beyond the scope of a tolerable computing time.

To accelerate the process, the cotangent  $\cot(\varphi) = x/y$  is assigned to every vertex  $\mathbf{C}_i$  resp.  $\mathbf{D}_j$ . For this purpose, let  $\varphi_i$  be the angle between the  $x$ -axis and the vector  $\mathbf{OC}_i$ , pointing from the origin to the vertex  $\mathbf{C}_i$ , and  $\psi_j$  be the angle between the  $x$ -axis and the vector  $\mathbf{OD}_j$ . If the cotangent values are in ascending order, this means

$$\cot(\varphi_i) < \cot(\varphi_{i+1})$$

and

$$\cot(\psi_j) < \cot(\psi_{j+1}),$$

we can confidently assume that points of intersection on a segment  $S_i$  can only be possible for segments  $T_j$  where

$$\cot(\psi_{j+1}) > \cot(\varphi_i)$$

and

$$\cot(\psi_j) < \cot(\varphi_{i+1}),$$

see figure 11.



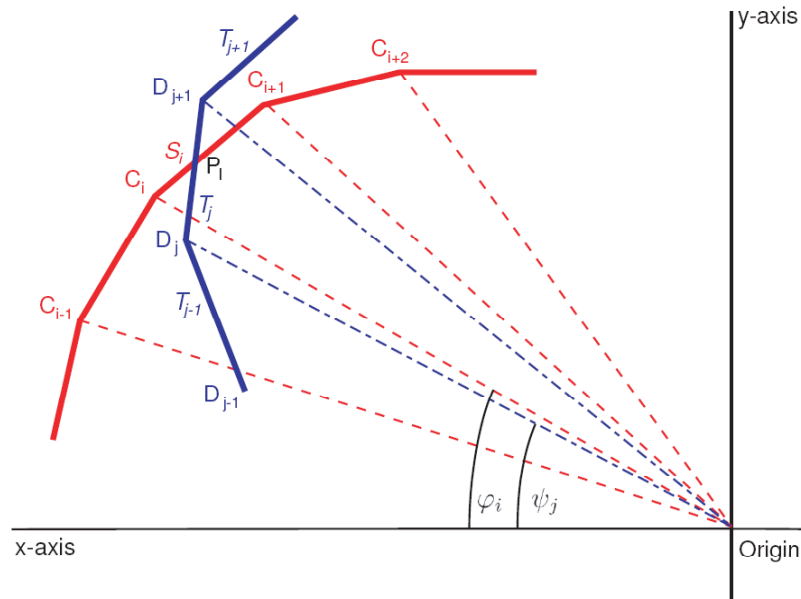


Figure 11: For possible points of intersection of the segment  $S_i$  only the segments  $T_j$  and  $T_{j+1}$  of the other polygon are worth considering.

Usually, a segment of a polygon has to be compared with 2 or 3 segments of the other polygon. This makes it possible to reduce the number of comparisons to about 150 as opposed to 2500 mentioned above.

Due to the structure of the algorithm certain outlines of specimens cannot be treated. As described above the cotangent-values of the successive vertices of the spline-approximating polygon must be in ascending order. In other words, the B-spline considered as a clockwise directed curve must not change its direction (viewed from the origin). As a rule, such shapes occur only in faulty datasets. To get a general idea of such uncalculable shapes, a couple of exemplary specimens with their approximating B-splines are indicated below.

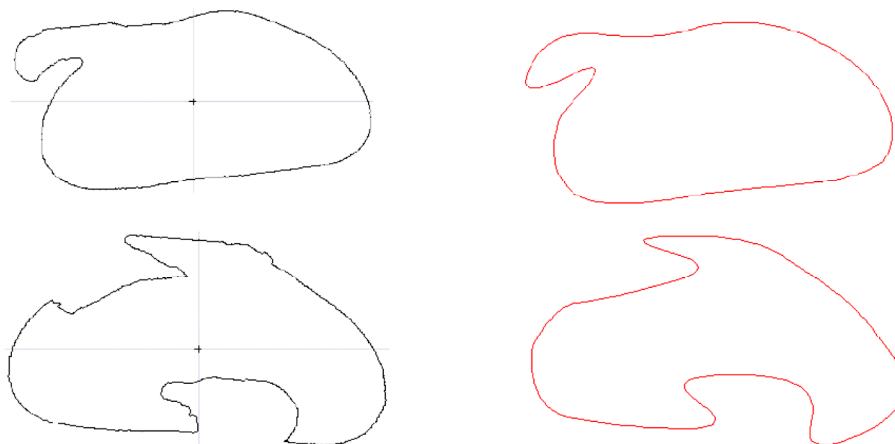


Figure 12: Examples of uncalculable outlines (black) and their approximating B-spline curves (red).

The distinction of ostracods using the area deviation is implemented in the current version of the program MORPHOMATICA (Linhart et al. 2006) and a brief documentation of the algorithm was introduced for the first time by us in Minati et al. (2008).

The program additionally offers the possibility to distinguish the valves by examining only the dorsal resp. ventral region.

## 5. Classification of Populations

In many fields of ostracodology a comparison of whole groups or populations of ostracods is wanted and necessary. In the following a measure for the dissimilarity of two populations is explained, which is proposed by one of us (J. L.). It is supposed that for any two individuals a kind of “distance” is available. So let  $P = p_1, \dots, p_n$  and  $Q = q_1, \dots, q_m$  be two populations and  $d$  a distance defined on  $P \cup Q$ . The *dissimilarity index* of  $P$  and  $Q$  is then defined by

$$\text{diss}(P, Q) := \frac{\frac{1}{nm} s(P, Q)}{\frac{1}{n^2} s(P) + \frac{1}{m^2} s(Q)},$$

where

$$s(P, Q) := \sum_{i=1}^n \sum_{k=1}^m d(p_i, q_k)$$

and

$$s(P) := \sum_{i=1}^{n-1} \sum_{k=i+1}^n d(p_i, p_k), \quad s(Q) := \sum_{i=1}^{m-1} \sum_{k=i+1}^m d(q_i, q_k).$$

Some essential properties of this dissimilarity index are:

1.  $\text{diss}(P, Q)$  only depends on the distances  $d(x, y)$  with  $x, y \in P \cup Q$ .
2.  $\text{diss}(P, Q)$  is invariant to scaling transformations.
3.  $\text{diss}(P, Q) = 1$  if  $P = Q$ .
4. If the underlying distance  $d$  is a so-called hypermetric (see Kelly 1970),  $\text{diss}(P, Q)$  is always  $\geq 1$ . The area deviation (cf. chapter 4.2) is a typical example for a hypermetric (see also Kelly 1970).

The dissimilarity index of  $P$  and  $Q$  may be viewed as the ratio of the average distance between individuals of different populations to the average distance between individuals of the same population. Perhaps this becomes more clear if the denominator of the expression defining  $\text{diss}(P, Q)$  is written in the following form:

$$\frac{\frac{1}{n^2/2}s(P) + \frac{1}{m^2/2}s(Q)}{2}.$$

At first sight, it seems that here  $n^2/2$  should be replaced by  $n(n-1)/2$ , since this is the number of distances within the population  $P$  (and analogously for  $Q$ ). But then the value of  $\text{diss}(P, Q)$  would be smaller than 1 for  $P = Q$ , which does not make sense. Intuitively, one might think that to a certain extent also the distances between identical individuals (which are of course equal to zero) should be taken into account.

For several purposes it will be more convenient to consider the natural logarithm of the dissimilarity index,

$$\ln \text{diss}(P, Q),$$

which yields a value in the range  $[0, \infty)$  instead of  $[1, \infty)$ .

The current version (1.6) of “Morphomatica” creates a “resemblance matrix” with entries representing the pairwise distances given by the area deviation. Table 1 shows a resemblance matrix comparing the valves of respectively 8 specimens of the species *Pseudocandona danubialis* from Ada-Kaleh (fig. 13) and *Pseudocandona eremita* from Aștileu (fig. 14), both localities in Romania. The data and pictures originate from a comparative study of Iepure et al. 2007, investigating the morphology of valves belonging to populations from Romania.

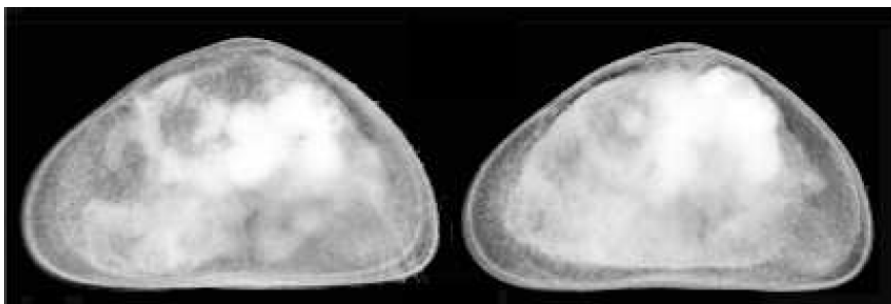


Figure 13: Lateral view of female valves belonging to the species *Pseudocandona danubialis* from Ada-Kaleh.

	AK1	AK2	AK3	AK4	AK5	AK6	AK7	AK8	Ast1	Ast2	Ast3	Ast4	Ast5	Ast6	Ast7	Ast8
AK1	–															
AK2	50.71	–														
AK3	25.11	38.77	–													
AK4	25.03	38.26	0.71	–												
AK5	9.47	51.62	25.54	25.42	–											
AK6	30.89	23.25	25.6	25.24	34.16	–										
AK7	24.45	48.31	16.59	16.65	23.44	35.46	–									
AK8	42.94	29.36	43.08	42.67	47.45	18.52	50.5	–								
Ast1	64.12	46.03	67.33	66.89	66.35	43.68	72.62	29.96	–							
Ast2	60.38	40.56	62.21	61.78	63.82	38.48	68.47	28.28	12.81	–						
Ast3	47.58	49.18	59.68	59.39	48.81	39.46	58.66	33.03	22.38	21.1	–					
Ast4	55.3	35.71	56.31	55.86	57.99	32.2	63.33	28.5	15.9	10.25	21.82	–				
Ast5	51.62	36.57	54.24	53.86	53.57	31.47	58.85	29.97	16.89	11.91	18.69	6.35	–			
Ast6	59.79	49.27	66.54	66.22	62.89	43.41	69.25	32.05	12.84	11.92	17.27	15.47	15.8	–		
Ast7	58.18	41.82	61.3	60.94	60.58	38.13	65.92	34.11	14.41	10.13	23.31	7.6	8.33	13.51	–	
Ast8	73.64	44.79	73.77	73.39	75.73	49.97	81.57	37.58	17.31	16.65	31.63	19.31	23.35	19.25	18.37	–

Table 1: Area deviation between valves of the Ada-Kaleh and Aştileu populations.

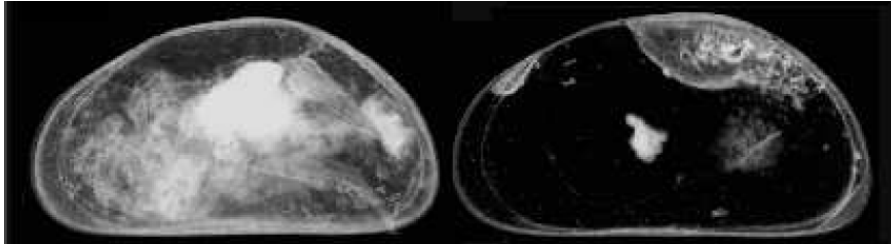


Figure 14: Lateral view of female valves belonging to the species *Pseudocandona eremita* from Aștileu.

Computing the above dissimilarity index for the Ada-Kaleh and Aștileu populations (resemblance matrix table 1) yields

$$\begin{array}{ll} n = 8 & m = 8 \\ s(P) = 869.2 & s(Q) = 454.56 \\ s(P, Q) = 3412.94 & \end{array}$$

and

$$\begin{array}{ll} \text{diss}(P, Q) & = 2.58, \\ \ln \text{diss}(P, Q) & = 0.947. \end{array}$$

## References

- Baltanás, A., Brauneis, W., Danielopol, D. L. and Linhart, J. 2003. Morphometric methods for applied ostracodology: tools for outline analysis of nonmarine ostracodes. In: L. E. Park and A. J. Smith (Eds.) Bridging the gap: trends in the ostracode biological and geological sciences. Paleontol. Soc. Papers, 9: 101-118.
- Bartels, R. H., Beatty, J. C. and Barsky; B. A. 1987. An Introduction to Splines for use in Computer Graphics and Geometric Modeling. Morgan Kaufmann Publishers, Los Altos.
- Bayer, S., Brauneis, W. and Trischitz, U. 2002. Approximierende B-Splines. Bachelor Thesis, Department of Mathematics, University of Salzburg.
- De Boor, C. 1978. A Practical Guide to Splines. Springer Verlag, New York.
- Deuflhard, P. and Hohmann, A. 1995. Numerical analysis. A first course in scientific computation. de Gruyter, Berlin.
- Farin, G. 1990. Curves and Surfaces for Computer Aided Geometric Design. A Practical Guide (2<sup>nd</sup> ed.) Academic Press, San Diego.

- Farin, G. 2001. Shape. In: B. Enquist (Ed.), *Mathematics Unlimited - 2001 and Beyond*: 463-477. Springer Verlag, Berlin.
- Hogben, L. 2007. *Handbook of Linear Algebra*. Chapman and Hall/CRC, Boca Raton.
- Hoschek, J. and Lasser, D. 1993. *Fundamentals of Computer Aided Geometric Design*. A. K. Peters, Wellesley, MA.
- Iepure, S., Namiotko, T. and Danielopol, D. L. 2007. Evolutionary and taxonomic aspects within the species group *Pseudocandona eremita*. *Hydrobiologia*, 585: 159-180.
- Kelly, J. B. 1970. Metric Inequalities and Symmetric Differences. In O. Shisha (Ed.), *Inequalities 2*: 193-212. Academic Press, New York.
- Linhart, J., Brauneis, W. Neubauer, W., Danielopol, D. L. 2006. *Morphomatica*, Computer Program, version 1.6. [http://palstrat.uni-graz.at/morphomatica/morphomatica\\\_e.htm](http://palstrat.uni-graz.at/morphomatica/morphomatica\_e.htm).
- Loy A., Busilacchi, S., Costa, C., Ferlin, L. and Cataudella, S. 2000. Comparing geometric morphometrics and outline fitting to monitor fish shape variability of *Diplodus puntazzo* (Teleostea: Sparidae). *Aquacultural Engineering* 21: 271-283.
- Minati, K., Cabral, M. C., Pipik, R., Danielopol, D. L., Linhart, J. and Neubauer W. 2008. Morphological variability among European populations of *Vestalenula cylindrica* (Straub). *Palaeogeogr., Palaeoclimat., Palaeoecol.*, 264: 296-305.
- Neubauer, W. 2007. Measuring the Difference of Approximating B-Spline Curves with Application in Distinguishing Ostracoda. Master thesis, Dept. of Mathematics, Univ. of Salzburg (download at [http://palstrat.uni-graz.at/morphomatica/morphomatica\\_e.htm](http://palstrat.uni-graz.at/morphomatica/morphomatica_e.htm)).
- Piegl, N. and Tiller, W. 1995. *The NURBS book*; Springer Verlag, Berlin.
- Rohlf, F. J. 2001. tpsDIG, Program version 1.43. Department of Ecology and Evolution, State University of New York, Stony Brook, NY.: <http://life.bio.sunysb.edu/morph/soft-dataacq.html> (1/17/04).
- Strang, G. 1998. *Lineare Algebra*. Springer Verlag, Berlin.
- Veltkamp, R. C. 2001. Shape Matching: Similarity Measures and Algorithms. In SMI 2001 International Conference, *Shape Modeling and Applications*: (download at: <http://people.cs.uu.nl/marc/asci/smi2001.pdf>: 188-197).
- Zelditch, M. L., Swiderski, D. L., Sheets, H. D. and Fink, W. L. 2004. *Geometric morphometrics for biologists, a primer*. Elsevier (Academic Press), Amsterdam.