Universität
Stuttgart

Fachbereich
Mathematik

Collocation with WEB–Splines

Christian Apprich, Klaus Höllig, Jörg Hörner, Ulrich Reif

Preprint 2015/009

Universität Stuttgart

Fachbereich Mathematik

Collocation with WEB–Splines

Christian Apprich, Klaus Höllig, Jörg Hörner, Ulrich Reif

Preprint 2015/009

Fachbereich Mathematik
Fakultät Mathematik und Physik
Universität Stuttgart
Pfaffenwaldring 57
D-70 569 Stuttgart

E-Mail: preprints@mathematik.uni-stuttgart.de
WWW: http://www.mathematik.uni-stuttgart.de/preprints

ISSN **1613-8309**

COLLOCATION WITH WEB–SPLINES

Christian Apprich, Klaus Höllig, Jörg Hörner and Ulrich Reif

May 8, 2015

Abstract

We describe a collocation method with weighted extended B–splines (WEB–splines) for arbitrary bounded multidimensional domains, considering Poisson's equation as a typical model problem. By slightly modifying the B–spline classification for the WEB–basis, the centers of the supports of inner B–splines can be used as collocation points. This resolves the mismatch between the number of basis functions and interpolation conditions, already present in classical univariate schemes, in a simple fashion.

Collocation with WEB–splines is very easy to implement; sample programs are available on the website `www.web-spline.de`. In contrast to standard finite element methods, no mesh generation and numerical integration is required, regardless of the geometric shape of the domain. As a consequence, computing times are considerably shorter than for Ritz-Galerkin schemes. Moreover, numerical tests confirm that increasing the B–spline degree yields highly accurate approximations already on relatively coarse grids.

Keywords: Collocation · WEB–spline · Boundary Value Problem · Interpolation

2010 Mathematics Subject Classification: 65L60 · 65D07

1 Introduction

B–splines play an important role in many branches of applied mathematics and engineering. Numerical methods, in particular approximation and data fitting, were among the first applications. Later on, the potential for geometric modeling and computer graphics has been realized, leading to a very fruitful synthesis of mathematical and engineering methods. The application to the numerical solution of partial differential equations is fairly recent. For a long time, the regular structure of tensor product grids prevented a systematic use of B–splines as finite elements. However, the geometric limitations could be overcome in a simple and elegant fashion. Essentially, two different techniques have been developed. Weighted extended B–splines (WEB–splines), introduced by Höllig, Reif, and Wipper [17], employ implicit descriptions of simulation domains; isogeometric elements, proposed by Hughes, Cottrell, and Bazilevs [18], use NURBS parametrizations for geometry representation. Both approaches, which are described in detail in the textbooks [13, 9], have two major advantages over classical mesh-based finite element techniques:

- regular grid with one trial function per grid point;

- arbitrary choice of order and smoothness.

As a consequence, finite element methods with B–splines can be implemented very efficiently and yield highly accurate numerical solutions with relatively few parameters.

In principle, isogeometric and weighted techniques can both be applied to all problems admitting a finite element discretization. Which method is the optimal choice primarily depends on the available geometry description. Isogeometric methods are best suited for domains which can be smoothly parametrized over rectangles or cuboids, or which can be expressed as unions of few such parametrizations. Often, the necessary domain parametrizations are available from CAD/CAM models using NURBS [26]. Weighted methods can handle domains well which have a convenient implicit description, or problems with natural boundary conditions. Implicit boundary representations naturally arise in constructive solid geometry and can be constructed, e.g., with Rvachev's R-function method [30]. Moreover, general methods for defining weight functions are available [4, 13]. Figure 1 illustrates typical isogeometric (left) and weighted (right) discretizations. The domain in the middle is an example where a combination of both techniques is appropriate [16].

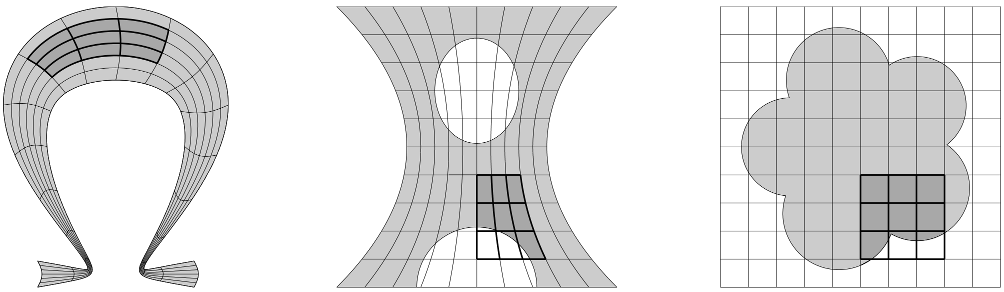


Figure 1: Isogeometric (left), mixed (middle), and weighted (right) finite element discretizations

In addition to substantial parameter savings, the smoothness of B–splines has other important advantages compared to classical, mesh-based C^0-elements. For a partial differential equation $\mathcal{L}u = f$ given on some domain $D \subset \mathbb{R}^d$, the pointwise residual of the finite element approximation $u^h \approx u$ can be computed if the chosen smoothness of the spline space is sufficiently high:

$$R^h(x) = f(x) - (\mathcal{L}u^h)(x), \quad x \in D\,.$$

Providing a convenient local a posteriori error measure, this feature can be used, for instance, to guide adaptive refinement. Moreover, turning to the subject of this article, B–splines can serve as basis functions for collocation. This straightforward discretization technique determines a numerical approximation u^h by demanding a vanishing residual at a grid of points:

$$R^h(\xi_i) = 0, \quad i \in I\,.$$

For isogeometric elements, collocation has been very successfully applied to a variety of problems [2, 3, 32]. So far, however, isogeometric collocation is limited to domains which are smooth images of rectangles or cuboids. For domains, modeled by a union of several isogeometric patches, two problems arise:

- there is no longer a canonical choice of the collocation points like for a tensor product grid of a single parameter rectangle or cuboid;
- for tesselations of planar domains, it is possible to adapt some highly specialized techniques from geometric modeling to maintain C^1- or C^2-continuity at extraordinary points [12], but nothing similar is known for higher dimensional cases.

With WEB–splines B_i, both difficulties can be overcome in a simple and elegant fashion. Deviating slightly from the original definition given in [17], a one-to-one correspondence can be established between the basis functions B_i, $i \in I$, and uniform B–splines b_i having the centers ξ_i of their supports inside the domain D. Hence, regardless of the shape of D, we can use the points ξ_i, $i \in I$, for collocation, a canonical choice in view of the fundamental Schoenberg-Whitney conditions (cf., e.g., [33]). Moreover, since WEB–splines share all essential approximation properties of uniform B–splines, we obtain stable and highly accurate numerical solutions. Perhaps the most significant advantage is the simplicity of the implementation. Compared with Ritz-Galerkin methods, no numerical integration is necessary, leading to substantial savings in computing time. The resulting MATLAB[1] code for Poisson's problem in three dimensions consists of less than 130 lines and can be downloaded from the website `www.web-spline.de`.

While WEB–collocation obviously has a broad range of applications, only Poisson's problem will be considered in this article. This typical model problem allows us to describe our new approach in a simple setting. After reviewing the definition and properties of uniform B–splines in Section 2, we give a slightly modified definition of WEB–splines, suited for collocation methods, in Section 3. We illustrate this definition in Section 4 for univariate interpolation, which serves as a motivating example for our collocation scheme. Sections 5 and 6 describe the collocation algorithm for Poisson's equation with Dirichlet boundary conditions and illustrate its performance. In the concluding Section 7 we summarize the main features of WEB–collocation and outline topics for future research.

2 B-Splines

Univariate B–splines can be defined for arbitrary knot sequences. Unfortunately, the resulting local flexibility for univariate approximation methods does not persist in several variables. For tensor product B–splines, knot placement has a global effect. Therefore, adaptive methods employ hierarchical bases with uniform B–splines (cf. e.g., [23, 10, 24, 1]) rather than global knot insertion techniques. The definition of WEB–splines is based on uniform B–splines as well. In light of the remarks just made, this should not be considered

[1] MATLAB® is a registered trademark of The MathWorks, Inc., Natick, MA, U.S.A.

a limitation or a potential drawback. Quite the contrary, using uniform knot sequences has obvious computational advantages. In particular, values, derivatives, and scalar products can be precomputed.

Definition 2.1 (B–Spline) *A d-variate uniform B–spline b_k of degree n, grid width h, and support $kh + [0, n+1]^d h$ is a product of univariate B–splines:*

$$b_k(x) = b_{(k_1,\ldots,k_d)}(x_1, \ldots, x_d) = b(x_1/h - k_1) \cdots b(x_d/h - k_d)\,,$$

where b is the standard cardinal B–spline of degree n with knots $0, \ldots, n+1$.

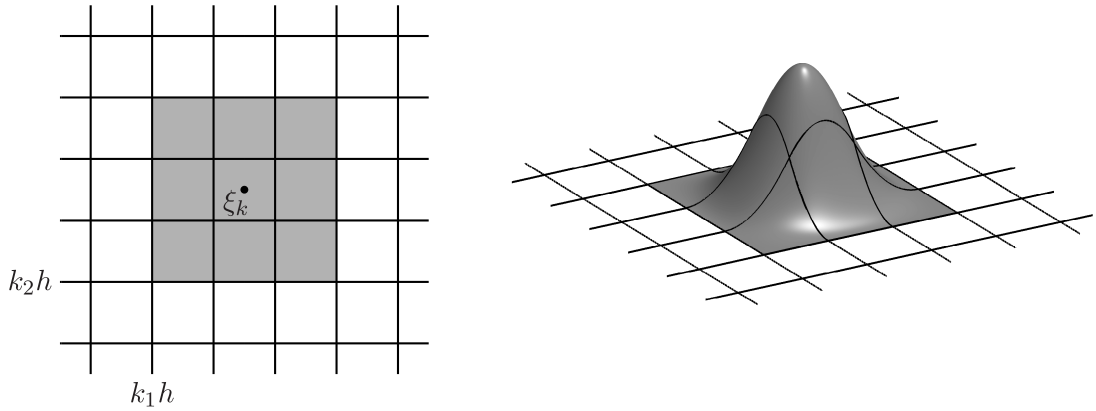


Figure 2: Support and graph of a biquadratic uniform B–spline b_k

As is illustrated in Figure 2, the uniform d-variate B–spline is a bell-shaped function corresponding to a regular grid. On lines parallel to the coordinate axes, b_k coincides with a multiple of a uniform univariate B–spline. The center of the support,

$$\xi_k = (k_1, \ldots, k_d)h + (n+1, \ldots, n+1)h/2\,,$$

is marked with a dot in the figure and is often used to identify the position of a B–spline on the grid.

The properties of B–splines are familiar [6]:

- b_k is nonnegative with support $kh + [0, n+1]^d h$;
- b_k coincides with a polynomial of coordinate degree n on each grid cell $\ell h + (0,1)^d h$;
- b_k is $(n-1)$-times continuously differentiable across grid cell boundaries;
- for any open set $\Omega \subseteq D$, the B–splines with some support in Ω are linearly independent.

Based on the last property, we can define splines for arbitrary domains in the usual way.

Definition 2.2 (Splines) *The spline space corresponding to a domain $D \subset \mathbb{R}^d$ is spanned by all relevant B–splines, i.e., by those B–splines b_k with some support in D (denoted by $k \sim D$):*

$$\mathbb{B}^h = \bigoplus_{k \sim D} b_k \, .$$

For computations it is convenient to keep also irrelevant indices, i.e., to represent splines as linear combinations

$$u^h = \sum_{k \in K} u_k b_k \, ,$$

where $K \subset \mathbb{Z}^d$ is the smallest rectangular array containing all relevant k, and u_k is set to zero for $k \not\sim D$.

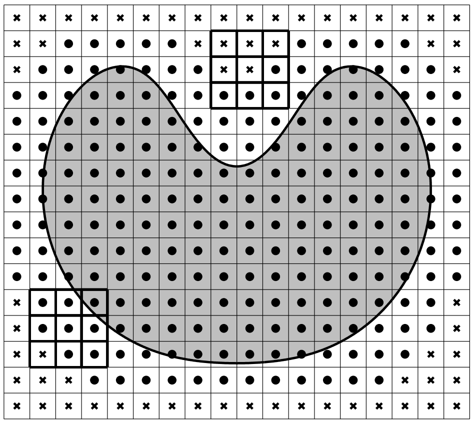


Figure 3: Relevant and irrelevant biquadratic B–splines, marked with dots and crosses, for a bounded domain D

Figure 3 shows the relevant and irrelevant biquadratic B–splines for a kidney-shaped domain. As mentioned before, we use dots at the relevant B–spline centers ξ_k to visualize the free parameters. Stars mark the irrelevant B–splines having no support inside of D.

The example reveals two principal difficulties, not present in the univariate theory:

- B–splines near the domain boundary with small support in D can cause severe instabilities, even if the grid width h is not small.
- Due to the rectangular B–spline support, it appears difficult to incorporate boundary conditions for general domains D.

WEB–splines, defined in the next section, resolve both problems in an elegant fashion. For their construction, the following beautiful explicit formula for representing polynomials plays a crucial role (cf., e.g., [14]).

Theorem 2.1 (Marsden's Identity) *For any $x, y \in \mathbb{R}^d$,*

$$(x-y)^n := \prod_{\nu=1}^{d} (x_\nu - y_\nu)^n = \sum_{k \in \mathbb{Z}^d} h^{nd} \psi(k - y/h)\, b_k(x)\,,$$

where $\psi(z) = \prod_{\nu=1}^{d} (z_\nu + 1) \cdots (z_\nu + n)$.

We note that $\psi(k - y/h)$ is a polynomial of coordinate degree n in $k = (k_1, \ldots, k_d)$. Since any polynomial p of coordinate degree $\leq n$ can be written as a linear combination of polynomials of the form $(\cdot - y_\ell)^n$, it follows that the coefficients u_k in the B–spline representation $p = \sum_k u_k b_k$ are a polynomial of coordinate degree $\leq n$ in k.

3 WEB–Splines

WEB–splines eliminate the problems of the spline space $\mathbb{B}^h$ mentioned in the previous section. They provide a stable basis and incorporate homogeneous boundary conditions. We discuss each aspect in turn.

To stabilize the B–spline basis for the spline space $\mathbb{B}^h$, we partition the relevant B–splines for a domain into two types. With the application to collocation methods in mind, we deviate slightly from the original splitting criterion.

Definition 3.1 (Classification) *The inner B–splines b_i, $i \in I$, for a domain D are those relevant B–splines with centers $\xi_i \in D$. The remaining relevant B–splines b_j, $j \in J$, are referred to as outer B–splines.*

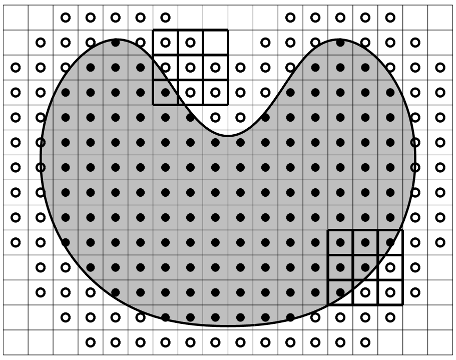


Figure 4: Inner and outer biquadratic B–splines, marked with dots and circles, respectively

An example of this classification is shown in Figure 4. We see that the inner B–splines b_i generally have a large portion or all of their support in D. If D is a Lipschitz domain, there exist boxes s_i and S_i with $s_i \subset D \cap \operatorname{supp} b_i \subset S_i$ whose side lengths are bounded from below, respectively above, by a constant times h^d. Hence, one can show by constructing appropriate dual functionals that this part of the B–spline basis is stable. On the other hand, the outer B–splines b_j are close to the boundary, and some of them have very small support in D. If we form appropriate linear combinations with neighboring inner B–splines,

$$ {}^{\mathrm{e}}b_i = b_i + \sum_{j \in J(i)} e_{i,j} b_j \,, $$

we can avoid instabilities while maintaining the approximation power of the spline space.

The extension of inner B–splines b_i by adjoining outer B–splines b_j is described in detail in [17]. Here, we merely sketch the key argument which is based on Marsden's identity; the extension coefficients $e_{i,j}$ are determined so that the representation of polynomials is preserved. Since the Marsden coefficients $h^{nd}\psi(k - y/h)$ are polynomials of coordinate degree n in $k_1, \ldots, k_d$, we can recover their values for an outer index j by interpolating at an array

$$ I(j) = \ell + \{0, \ldots, n\}^d, \quad \ell = \ell(j)\,, $$

of inner indices i. We write the interpolant in Lagrange form

$$ \psi(j - y/h) = \sum_{i \in I(j)} e_{i,j} \psi(i - y/h)\,, $$

where

$$ e_{i,j} = \prod_{\nu=1}^{d} \prod_{\substack{\alpha_\nu = \ell_\nu \\ \alpha_\nu \neq i_\nu}}^{\ell_\nu + n} \frac{j_\nu - \alpha_\nu}{i_\nu - \alpha_\nu} \tag{1} $$

are the values of the Lagrange polynomials corresponding to the indices $i \in I(j)$, evaluated at j. Substituting into Marsden's identity, we arrive at

$$ \prod_{\nu=1}^{d} (x_\nu - y_\nu)^n = \sum_{i \in I} h^{nd} \psi(i - y/h) b_i(x) + \sum_{j \in J} \sum_{i \in I(j)} e_{i,j} h^{nd} \psi(i - y/h) b_j(x)\,. $$

Interchanging the order of summation in the double sum leads to

$$ \sum_{i \in I} h^{nd} \psi(i - y/h) \left[b_i(x) + \sum_{j \in J(i)} e_{i,j} b_j(x) \right]\,, $$

where $J(i) = \{j \in J : i \in I(j)\}$. This shows that Marsden's identity remains valid for the extended B–splines ${}^{\mathrm{e}}b_i = [\ldots]$. As a consequence, the approximation power of the stabilized spline space

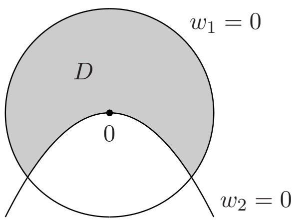
$$ {}^{\mathrm{e}}\mathbb{B}^h = \bigoplus_{i \in I} {}^{\mathrm{e}}b_i \subset \mathbb{B}^h $$

has not deteriorated.

We now turn to the representation of homogeneous Dirichlet boundary conditions. In hindsight, the key idea is quite simple. If the domain is represented in implicit form by means of some *weight function* $w : \mathbb{R}^d \to \mathbb{R}$,

$$D : \ w(x) > 0 \,,$$

multiplying the relevant B–splines b_k by w yields weighted B–splines wb_k which vanish on ∂D. This method has already been suggested by Kantorovich and Krylov [21], and was extensively analyzed by Rvachev and his collaborators in the decades after 1970 [29, 30, 31, 34]. In order to preserve optimal approximation properties, it is essential that w is positive and smooth on $\overline{D}$ and vanishes on ∂D in such a way that the ratio of w and the boundary distance as well as its reciprocal remain bounded on D. However, even weight functions which are not differentiable at the boundary turn out to be useful in practice. Such weight functions can be constructed in various ways. Rvachev's R-function method combines weight functions for elementary domains according to Boolean operations; cf. Figure 5 for an example. General purpose constructions are based on smoothed distance functions [13] and spline approximations [4].



$$\begin{aligned} w_1(x) &= 1 - x_1^2 - x_2^2 \\ w_2(x) &= x_1^2 + x_2 \\ w &= w_1 + w_2 - \sqrt{w_1^2 + w_2^2} \end{aligned}$$

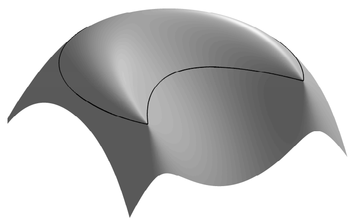


Figure 5: Combining elementary weight functions via Rvachev's method

Combining the extension procedure with the domain representation via weight functions leads to the following definition:

Definition 3.2 (WEB–Splines) *The weighted extended B–splines, corresponding to an implicitly defined domain $D : w > 0$, are given by*

$$B_i = \frac{1}{\gamma_i} \, w \, {}^{\mathrm{e}}b_i, \quad i \in I \,,$$

where the normalizing constants γ_i are chosen proportional to the maximum of w on $D \cap \operatorname{supp} b_i$. The WEB–splines form a stable basis for a subspace $w^{\mathrm{e}}\mathbb{B}^h$ of the weighted spline space $w\mathbb{B}^h$.

Figure 6 shows several WEB–splines on a two–dimensional domain. Qualitatively, there is little difference to standard uniform B–splines, except that the support is adapted to the curved boundary, and WEB–splines near the boundary can be negative on small subsets of their support.

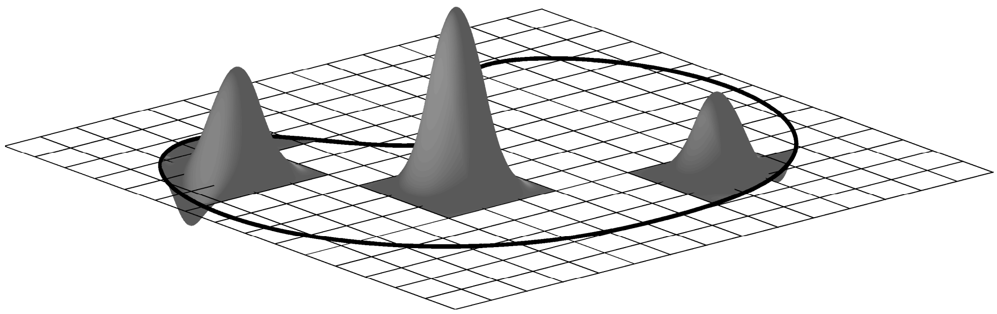


Figure 6: Biquadratic WEB–splines

As was shown in [17], WEB–splines inherit all properties of uniform B–splines which are essential for finite element methods:

- The diameter of $\operatorname{supp} B_i$ is proportional to the grid width h.
- At any point $x \in D$ at most $O(1)$ WEB–splines are nonzero.
- The WEB–basis is stable, i.e.,

$$h^{d/2}\,|U| \asymp \left\| \sum_i u_i B_i \right\|,$$

  where $|\cdot|$, $\|\cdot\|$ denote the 2-norm of vectors and the L_2-norm of functions on D, respectively, and the symbol $\asymp$ denotes inequalities in both directions with constants independent of h.
- Linear combinations of WEB–splines approximate functions u, for which u/w is smooth on $\overline{D}$, with optimal order $O(h^{n+1})$.

The proofs of the last two properties are not straightforward. However, the arguments of [17] are easily adapted to the slightly modified WEB–spline concept used in this article.

4 Univariate Interpolation

Collocation for boundary value problems is closely related to interpolation methods. Therefore, we consider this simpler application first, as a motivating example for our new discretization technique. We begin by recalling the fundamental criterion for the well-posedness of univariate spline interpolation.

Theorem 4.1 (Schoenberg-Whitney Conditions) *Any data f_k can be interpolated uniquely at an increasing sequence of points t_k by a continuous spline $\sum_k u_k b_k$ if and only if $b_k(t_k) > 0$ for all k.*

Intuitively, for uniform B–splines b_k, choosing t_k to be the center ξ_k of $\operatorname{supp} b_k$ appears to be the optimal choice. Unfortunately, the number of relevant B–splines for a bounded interval D is larger than the number of interpolation conditions. As is illustrated in Figure 7 for degree 3, some of the points $t_k = \xi_k$ lie outside of D. There are several possible remedies:

- Place additional interpolation points near the left and right endpoints of D.
- Impose boundary conditions at the endpoints of D (typically derivative values).
- Remove knots near the endpoints of D (not-a-knot condition).

Unfortunately, none of these choices has a natural multivariate analogue for domains which are topologically not of tensor product type.

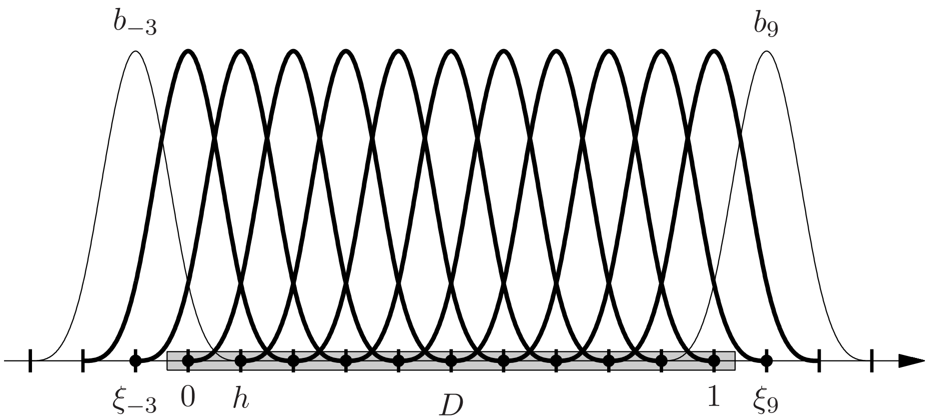


Figure 7: Cubic interpolation at B–spline centers

With the definitions of the previous section in mind, a simple alternative suggests itself:

Definition 4.1 (Interpolation with Extended B–Splines) *With* ${}^{\mathrm{e}}b_i$*,* $i \in I$*, the extended B–splines with respect to an interval* D*, an extended spline interpolant*

$$u^h = \sum_{i \in I} u_i \, {}^{\mathrm{e}}b_i$$

matches given data f_i *at the B–spline centers* ξ_i*. The coefficients* u_i *are determined by solving the linear system*

$$AU = F, \quad a_{i,i'} = {}^{\mathrm{e}}b_{i'}(\xi_i)\,.$$

A multivariate generalization of this new interpolation variant is straightforward. By definition, the centers ξ_i of the inner B–splines lie inside the domain D and can be used as interpolation points, corresponding one-to-one to the stabilized basis functions ${}^{\mathrm{e}}b_i$.

For the example in Figure 7, there are two outer B–splines with centers at $\xi_{-3} = -h$ and $\xi_9 = 1 + h$, i.e., $I = \{-2, \ldots, 8\}$ and $J = \{-3, 9\}$. According to definition (1) (with $d = 1$, $n = 3$), the extension coefficients corresponding to $I(-3) = \{-2, -1, 0, 1\}$ and $I(9) = \{5, 6, 7, 8\}$ are

$$e_{\cdot,-3} : \; 4, -6, 4, -1, \qquad e_{\cdot,9} : \; -1, 4, -6, 4\,.$$

For example,

$$e_{-1,-3} = \frac{-3-(-2)}{-1-(-2)} \cdot \frac{-3-0}{-1-0} \cdot \frac{-3-1}{-1-1} = -6\,.$$

This yields the extended B–splines

$${}^{\mathrm{e}}b_{-2} = b_{-2} + 4b_{-3},\ {}^{\mathrm{e}}b_{-1} = b_{-1} - 6b_{-3},\ {}^{\mathrm{e}}b_0 = b_0 + 4b_{-3},\ {}^{\mathrm{e}}b_1 = b_1 - b_{-3},\ {}^{\mathrm{e}}b_2 = b_2, \ldots$$

The corresponding interpolation matrix is easily computed from the B–spline values 1/6, 2/3, 1/6 at the grid points:

$$A = \begin{pmatrix} \frac{2}{3}+\frac{4}{6} & \frac{1}{6}-1 & 0+\frac{4}{6} & 0-\frac{1}{6} & & & & \\ \frac{1}{6} & \frac{2}{3} & \frac{1}{6} & 0 & & & & \\ & \frac{1}{6} & \frac{2}{3} & \frac{1}{6} & & & & \\ & & \ddots & \ddots & \ddots & & & \\ & & & \frac{1}{6} & \frac{2}{3} & \frac{1}{6} & & \\ & & & 0 & \frac{1}{6} & \frac{2}{3} & \frac{1}{6} & \\ & & & 0-\frac{1}{6} & 0+\frac{4}{6} & \frac{1}{6}-1 & \frac{2}{3}+\frac{4}{6} & \end{pmatrix}.$$

For example, the second entry in the first row is the value of the second extended B–spline at the first inner B–spline center:

$${}^{\mathrm{e}}b_{-1}(\xi_{-2}) \;=\; b_{-1}(\xi_{-2}) - 6b_{-3}(\xi_{-2}) \;=\; \frac{1}{6} - 6 \cdot \frac{1}{6} \;=\; -\frac{5}{6}\,.$$

Since each B–spline is nonzero only at three evaluation points, the extension procedure only influences the first and last row of A. The other rows coincide with the standard cubic interpolation matrix.

Figure 8 illustrates the convergence of extended spline interpolation. We choose $f(x) = \exp(x)\sin(2\pi x)$ as a test function and obtain the standard order of convergence $O(h^{n+1})$. The small MATLAB program, which generates the figure, can be downloaded from the website `www.web-spline.de` and is easily modified to experiment with our interpolation method.

5 Collocation for Poisson's Problem

As already indicated in the introduction, the concept of collocation is as simple as convincing. To approximate the solution u of Poisson's problem

$$-\Delta u \;=\; -\sum_{\nu=1}^{d} \frac{\partial^2 u}{\partial x_\nu^2} \;=\; f \quad \text{in } D, \qquad u \;=\; 0 \quad \text{on } \partial D, \tag{2}$$

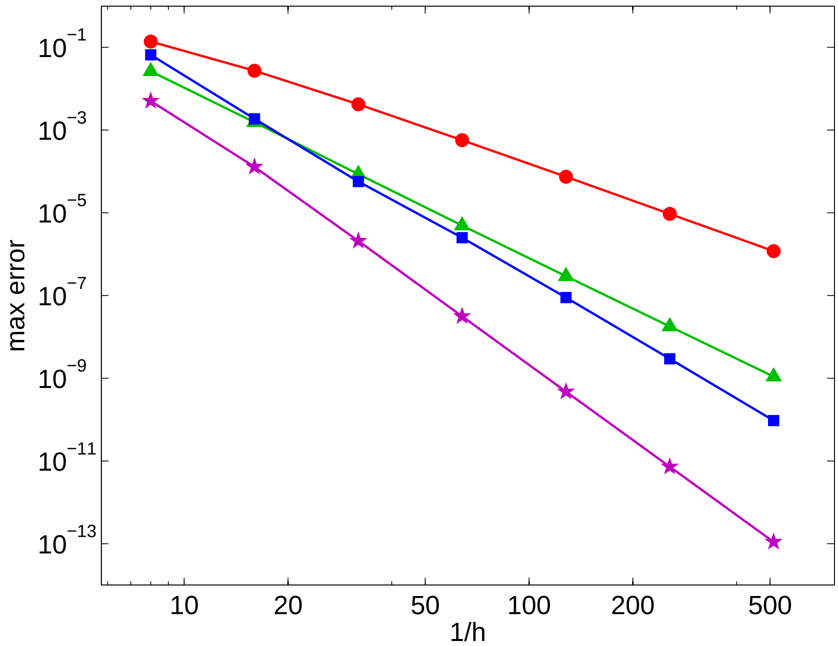


Figure 8: Error of extended spline interpolation for degrees $n = 2$ (●), 3 (▲), 4 (■) and 5 (★) as a function of the grid width h

on a bounded domain $D \subset \mathbb{R}^d,\ d \in \mathbb{N},$ by a linear combination

$$u^h \;=\; \sum_{i \in I} u_i\, B_i \;\in\; w^{\mathrm{e}}\mathbb{B}^h$$

of WEB–splines of degree $n \geq 2$, vanishing on ∂D by construction, we require u^h to satisfy the governing differential equation at all centers ξ_i of B–spline supports for which i is an inner index, i. e.

$$-\,\Delta u^h(\xi_i) \;=\; -\sum_{i' \in I} u_{i'} \Delta B_{i'}(\xi_i) \;=\; f(\xi_i) \qquad \text{for all}\ \ i \in I, \tag{3}$$

cf. Figures 9 (left) and 11 (right). Due to the one–to–one correspondence established above between the basis functions B_i and the chosen collocation points, (3) forms a linear system

$$CU \;=\; F, \qquad c_{i,i'} \;=\; -\Delta B_{i'}(\xi_i),$$

with the same number of equations given and coefficients $u_{i'}$ to be determined. Since the ξ_i are located at the centers of the grid cells for even n, quadratic WEB–splines are admissible as basis functions despite having discontinuous second derivatives across the grid lines.

Compared with current techniques for solving boundary value problems like finite element or finite volume methods, which are typically restricted to a specific class of problems, collocation is a much more general concept. Any differential operator $\mathcal{L}$, linear or nonlinear and of arbitrary order, could replace the Laplacian in (2). We only need to ensure that the functional basis is smooth enough to be inserted into the resulting differential equation.

Collocation techniques appear in the literature yet before 1940 [20, 11]. Progress in analyzing the method was first made in the case of univariate problems and polynomial bases, leading to convergence and stability results for different choices of collocation points in the interval domain [22, 36, 35]. From the late 1960s on, spline functions were preferred to global polynomials as collocation bases for solving ordinary differential equations [28, 7]. In contrast, multivariate collocation developed much slower, restrained in particular by the lack of any obvious concept of how to incorporate essential boundary conditions into the collocation space for boundary value problems on general domains in two or more dimensions. Research thus focused on approximation on rectangular domains, using splines of third [8, 19, 27] or higher degree [25]. The principal class of curved domains, for which the method was successfully adopted, comprised all regions that are given as images of rectangles or cuboids under smooth coordinate transforms [5, 2], or can be decomposed into a finite number of such curvilinear patches or blocks, respectively [3, 32].

Our implementation of WEB–collocation for Poisson's problem on plane or spatial domains is fully self–contained, though sharing many concepts and structures with the FEMB finite element package presented in [15]. For the sake of consistency, the domain D is supposed to be scaled to fit into the d–dimensional unit cube $Q = [0, 1]^d$. A weight function w, implicitly defining D, is specified by the user, along with the right–hand side f of (2), the spline degree n and the number $H = 1/h$ of grid cells per coordinate direction. The main steps of the algorithm are then the following:

Step 1: Precomputing Spline Values and Extension Coefficients

As all B–splines entering the WEB–spline basis, together with their partial derivatives of order ≤ 2, need to be evaluated only at the collocation points, located at the centers ξ_i of inner B–spline supports, we can determine and store the values

$$\partial^\alpha b_{k'}(\xi_k), \qquad \alpha \in \mathbb{N}_0^d, \quad |\alpha| \leq 2, \quad k, k' \in K,$$

right in advance before starting the actual computation. Due to translatorial symmetry of B–splines and boundedness of their supports, the number of nonzero values, that actually need to be stored per order of derivative, is limited to n^2 if n is odd, or $(n+1)^2$ if n is even. In a similar preprocessing step, we can compute and tabulate all extension coefficients $e_{i,j}$ potentially appearing in any of the WEB–splines B_i. Namely, according to (1), the value of $e_{i,j}$ only depends on the relative position of the indices i, j and $\ell(j)$.

Step 2: Classification of B–Splines

Given the domain $D : w > 0$, we determine a box $S \supseteq D$ and define the array $\tilde{K} \subset \mathbb{Z}^d$ as the set of indices of B–splines which do not vanish on S. Hence, all relevant indices are contained in $\tilde{K}$. In order to split the elements of $\tilde{K}$ into inner, outer, and irrelevant B–splines, we evaluate the weight function w at all centers ξ_i of B–spline supports. Applying

Definition 3.1, we include b_i in the set of inner B–splines, if $w(\xi_i) > 0$. In the sense of Definition 2.2, the remaining B–splines must be checked for an overlap of their support with D. However, as a slight modification, we suggest to regard b_j as outer if and only if $b_j(\xi_i) > 0$ for some inner index $i \in I$. This criterion is much easier to check and does not alter the linear system to be solved. The two approaches reveal differences only when evaluating the approximate solution u^h at points which are very close to the boundary of the domain, and typically, these differences are significantly smaller than the overall approximation error.

Step 3: Constructing the WEB–Spline Basis

By traversing the set I of inner indices, we can identify all sub-arrays of size $n+1$ and store their centers η_i. Then, for each $j \in J$, we determine the nearest of these centers to obtain the array $I(j)$. To store extension coefficients and other data in standard matrix format for further processing, we need to convert multi–indices to sequences of consecutive univariate indices. Applying, for instance, lexicographic ordering to the set of relevant indices $K \subset \mathbb{Z}^d$, we obtain the index set $K_* = \{1, 2, \ldots, \#K\} \subset \mathbb{N}$. The subscript star also identifies corresponding indices, i.e., k_* is the index in K_* corresponding to $k \in K$. In an analogous way, we derive I_* from I and identify $i_* \in I_*$ with $i \in I$. Now, the *extension matrix* E of size $\#K \times \#I$ is defined by

$$E(k_*, i_*) = \begin{cases} 1 & \text{if } k = i \in I \\ e_{i,j} & \text{if } k = j \in J \text{ and } i \in I(j) \\ 0 & \text{else.} \end{cases}$$

That is, rows of E corresponding to an inner index just refer to the index itself, while rows corresponding to an outer index contain the extension coefficients. The $e_{i,j}$ can be read from the table of precomputed values, according to Step 1.

Step 4: Assembly of the Collocation System

The assembly of the collocation system corresponding to the WEB–basis is fairly easy. First, we apply the differential operator to the relevant weighted B–splines and evaluate at the centers ξ_i to obtain a $\#I \times \#K$-matrix L:

$$L(i_*, k_*) = -\Delta(w b_k)(\xi_i), \quad i_* \in I_*,\ k_* \in K_*.$$

This step can also be made efficient by using the precomputed values according to Step 1. Second, multiplication with the extension matrix yields the *collocation matrix* $C = LE$. This matrix is square and contains, as requested, the entries $C(i_*, \ell_*) = -\Delta(w\, {}^{\mathrm{e}}b_\ell)(\xi_i)$. Third, we define the vector F by evaluating the function f at the centers ξ_i, i.e., $F_{i_*} = f(\xi_i), i_* \in I_*$. Now, the linear system to be solved reads $CU = F$.

Step 5: Solving and Postprocessing

For the purpose of keeping our code short and comprehensible, we solve the linear system

$CU = F$ by applying MATLAB's standard backslash command, which is sufficiently fast and stable for dealing with the sample problems presented in this paper. It is, though, evident that a high–performance solver, suited for general non–symmetric systems with high sparsity, should be employed instead when moving to more complex computations. In a final step, the coefficients u_i in terms of the WEB–spline basis are recombined according to

$$u^h \;=\; \sum_{i\in I} u_i\, B_i \;=\; \sum_{i\in I} \frac{u_i}{\gamma_i}\, wb_i \;+\; \sum_{j\in J} \sum_{i\in I(j)} \frac{u_i}{\gamma_i}\, e_{i,j}\, wb_j \;=\; \sum_{k\in K} \tilde{u}_k\, wb_k,$$

yielding a representation of u^h with respect to the regular B–spline basis of $\mathbb{B}^h$. Again, this can be implemented through a multiplication with the extension matrix E.

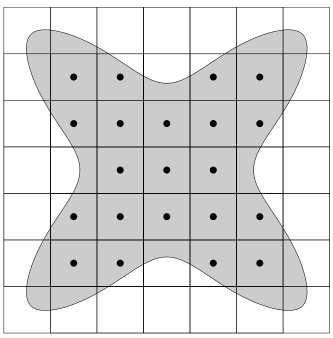
For a rough visualization and estimation of approximation errors and rates of convergence, we use the values of w and b_k at the centers of all B–spline supports, which have already been computed in the course of the algorithm, so that no additional evaluation of functions is necessary.

6 Convergence Rates and Computing Times

As a first illustration of the performance of our collocation algorithm, we consider a planar free–form domain described by a weight function in Bézier form,

$$w(x) \;=\; \sum_{k_1=0}^{m} \sum_{k_2=0}^{m} w_k\, b_k^m(x_1, x_2)\,,$$

where $b_k^m(x) = b_{k_1}^m(x_1)\, b_{k_2}^m(x_2)$ are the bivariate Bernstein polynomials of degree m, cf. [14]. The example shown in Figure 9 was generated using $m = 4$ and the matrix $W = (w_k)$ of Bernstein coefficients displayed at the right side of the figure.



$$W \;=\; \begin{pmatrix} -1 & 0 & -8 & 0 & -1 \\ 0 & 16 & -10 & 16 & 0 \\ -8 & -10 & 4 & -10 & -8 \\ 0 & 16 & -10 & 16 & 0 \\ -1 & 0 & -8 & 0 & -1 \end{pmatrix}$$

Figure 9: Bézier domain with grid and collocation points ξ_i, $i \in I$ (left), and matrix of coefficients (right)

More generally, we could use bivariate splines for defining weight functions. In this case, the collocation basis functions are products of linear combinations of B–splines. The grid widths need not be the same; usually a coarser mesh is used for the function w than for the collocation grid. If a common knot spacing is desired, e.g., for a more efficient implementation, this can be accomplished via subdivision.

Returning to our example, we chose $u(x) = \mathrm{e}^{w(x)} - 1$ as a "manufactured" solution to Poisson's problem. Setting $f = -\Delta u$, we computed approximations to u using WEB–splines of degree $n = 2, \ldots, 5$ on grids with up to 3200 cells per coordinate, resulting in linear systems of about 5.3 million unknowns. The maximal pointwise errors and accumulated computation times are shown in the two diagrams of Figure 10. We observe convergence at order $O(h^n)$ for splines of even degree n, and $O(h^{n-1})$ for odd n, in full conformance with the L^∞ rates obtained by isogeometric collocation for boundary value problems on a plane domain [2, 3, 32]. Although not yet fully optimized with respect to the linear solver, our program returned the results, even for highly refined discretizations, within a few minutes at most.

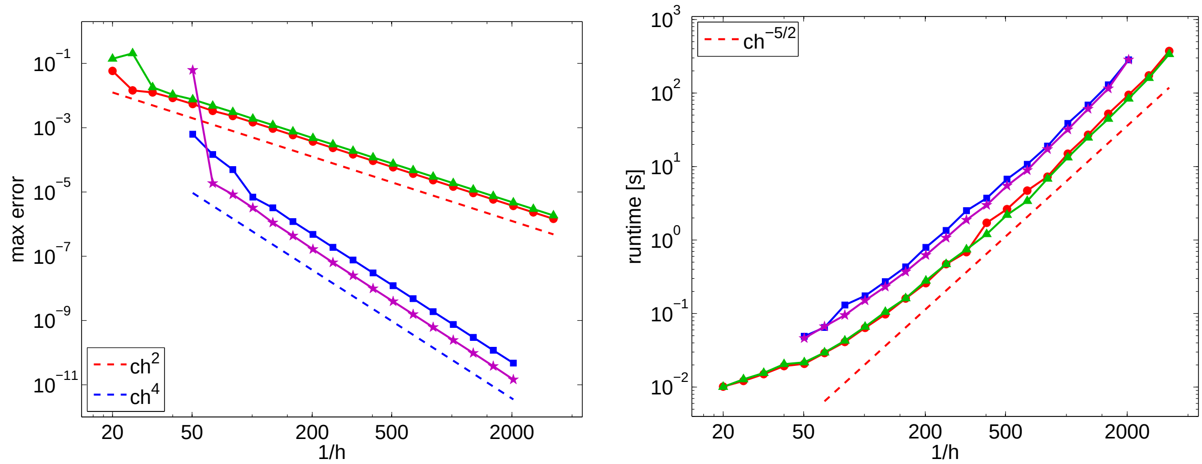


Figure 10: Error (left) and computation time (right) of 2D WEB–spline collocation for degrees $n = 2$ (●), 3 (▲), 4 (■) and 5 (★) as functions of the grid width h

It is one of the most convenient features of WEB–spline collocation that, due to the regular pattern of the underlying grids, the two–dimensional algorithm is easily adapted to three dimensions. The only action to be taken is to replace doubly by triply indexed array structures, expand double to triple loops, and provide trivariate instead of bivariate weight and force functions. Hence, our MATLAB codes for plane and for spatial problems do not differ by a single line in length.

A three–dimensional test domain D is given by the smooth weight function[2]

$$w(x) = 1/4 - (y_1^2 + y_2^2/3 - 1)^2(y_1^2/4 + y_2^2/4 - 1)^2 - y_3^2, \qquad y_\nu = 5(x_\nu - 1/2),$$

cf. Figure 11. We define $u(x) = w(x)\,\mathrm{e}^{x_1+x_2x_3}$, for $x \in D$, as true solution to (2) and, as before, recover u by entering $f = -\Delta u$ as right–hand side into our algorithm.

[2]Many other beautiful examples of domains bounded by compact algebraic surfaces can be found at `http://virtualmathmuseum.org/Surface/gallery_o.html#AlgebraicSurfaces`.

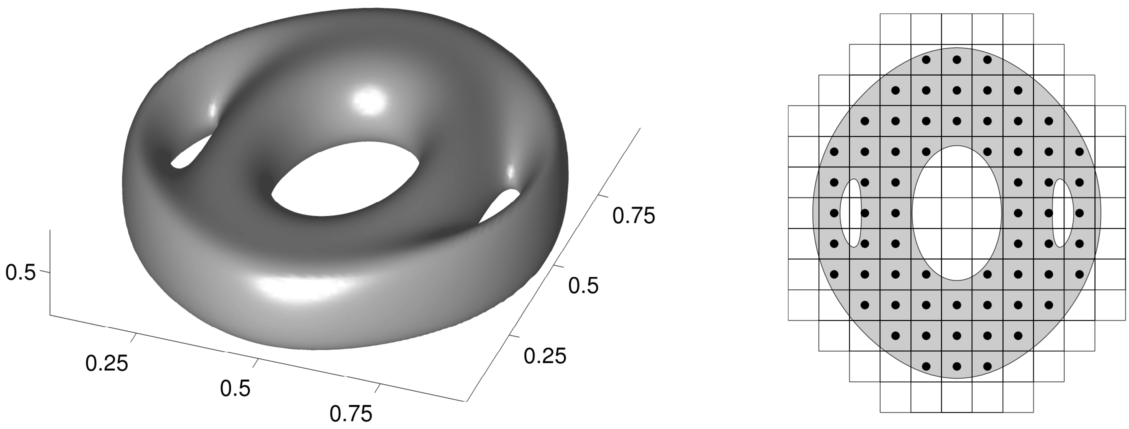


Figure 11: Spatial domain D, defined by $w(x) > 0$ (left), and cross section of grid with collocation points ξ_i, $i \in I$ (right)

Figure 12 shows the decrease of the relative L^2 error, together with the increase in computation time, for a sequence of progressively refined grids ($H \leq 140$) and WEB–splines of degrees 2 to 5. Like in the two–dimensional case, we observe significantly higher accuracies for WEB–splines of degree $n \in \{4, 5\}$, and the obvious pairing of subsequent even and odd orders, as is characteristic for collocation methods. The runtimes, to end with, were again short enough to await the results in front of the computer. For example, a tricubic approximation with 224 376 B–spline coefficients was computed in less than 82 seconds on an Intel Core i7 − 3770 machine (2.8 GHz, 16 GB memory).

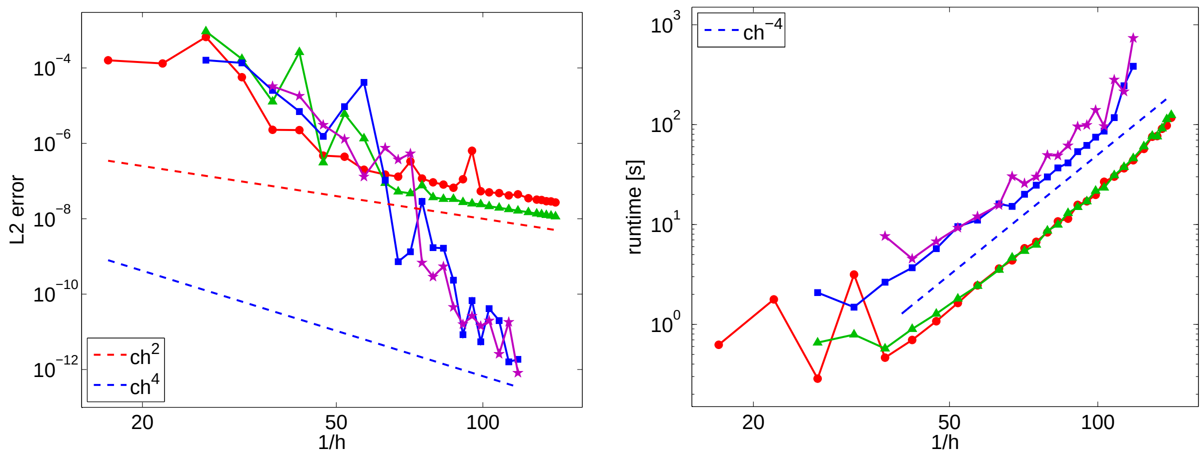


Figure 12: Error (left) and computation time (right) of 3D WEB–spline collocation for degrees $n = 2$ (●), 3 (▲), 4 (■) and 5 (★) as functions of the grid width h

The small runtimes are partly due to the sparseness of the collocation system. For example, for tricubic B–splines, Ritz–Galerkin matrices have an average bandwidth of $(2 \cdot 3 + 1)^3 = 343$, while the bandwidth for WEB–collocation is $3^3 = 27$. As a consequence, the total computation time t grows only moderately with the dimension, which is proportional to h^{-3}. The dashed line in the right diagram of Figure 12 suggests that $t \approx O(h^{-4})$, which is only slightly worse than the optimal linear growth $O(h^{-3})$.

7 Conclusion

We have described a collocation method with WEB–splines for Poisson's equation with homogeneous Dirichlet boundary conditions on arbitrary multidimensional domains. Regardless of the shape of the domain, there is a canonical unique correspondence between the basis functions and the collocation points which are the B–spline centers inside the domain.

The method is simple to implement, and a sample MATLAB code can be downloaded from the website `www.web-spline.de`. Since no numerical integration is involved, the discretization is considerably less time-consuming than for standard Ritz-Galerkin methods.

Numerical tests confirm the error behavior expected from the classical univariate theory. Theoretical derivation seems out of reach at present and remains an open question to our as well as to any other multivariate collocation method. We recall, however, that for WEB–spline approximations, the residual can be evaluated at arbitrary points, which, in conjunction with bounds on derivatives, provides a convenient a posteriori error bound.

The generalization to other elliptic boundary value problems (e.g., biharmonic equation, system of linear elasticity, eigenvalue problems, ...) with homogeneous essential boundary conditions is straightforward. Moreover, from the univariate example in Section 4, it is apparent that the method can also be used for interpolating functions and data on arbitrary free–form domains.

The implementation of natural and mixed boundary conditions is less obvious. It is conceivable that the solution structures developed by Rvachev and his collaborators can be adapted to suit the collocation approach. Another topic of current research is collocation for hierarchical bases. This generalization is essential for treating singular problems efficiently.

We are optimistic that progress on the variety of open problems will be possible, making WEB–collocation a very competitive alternative to standard Ritz-Galerkin approximation with B–splines.

References

[1] C. Apprich, K. Höllig, J. Hörner, A. Keller and E. Nava–Yazdani: *Finite Element Approximation with Hierarchical B–Splines*, to appear in: Curves and Surfaces 2014, J.D. Boissonnat et al. (eds.), LNCS, Springer (2015).

[2] F. Auricchio, L. Beirão da Veiga, T.J.R. Hughes, A. Reali, G. Sangalli: *Isogeometric collocation methods*, Math. Mod. Meth. Appl. Sci. 20 (2010), 2075-2107.

[3] F. Auricchio, L. Beirão da Veiga, T.J.R. Hughes, A. Reali, G. Sangalli: *Isogeometric collocation for elastostatics and explicit dynamics*, Comput. Methods Appl. Mech. Engrg. 249/252 (2012), 2-14.

[4] C. Bajaj, R. Bettadapura, N. Lei, A. Mollere, C. Peng and A. Rand: *Constructing A-spline weight functions for Stable WEB–spline finite element methods*, in: Proc. ACM Symposium on Solid and Physical Modeling, SPM 2010, G. Elber et al. (eds.), ACM (2010), 153–158.

[5] J.J. van Blerk, J.F. Botha: *Numerical solution of partial differential equations on curved domains by collocation*, Numer. Methods Partial Differential Equations 9 (1993), 357–371.

[6] C. de Boor: *A Practical Guide to Splines*, Springer-Verlag, New York, 1978.

[7] C. de Boor, B. Swartz: *Collocation at Gaussian Points*, SIAM J. Numer. Anal. 10 (1973), 582–606.

[8] J.C. Cavendish: *A collocation method for elliptic and parabolic boundary–value problems using cubic splines*, Dissertation, University of Pittsburgh, 1972.

[9] J.A. Cottrell, T.J.R. Hughes, and Y. Bazilevs: *Isogeometric Analysis: Toward Integration of CAD and FEA*, Wiley, 2009.

[10] T. Dokken, T. Lyche, K.F. Pettersen: *Polynomial splines over locally refined box-partitions*, Comput. Aided Geom. Design 30 (2013), 331–356.

[11] R.A. Fraser, W.N.P. Jones and S.W. Skan: *Approximations to functions and to the solutions of differential equations*, ARC Technical Report, GB Aeronautical Research Committee, 1937.

[12] D. Groisser, J. Peters: *Matched G^k-constructions always yield C^k-continuous isogeometric elements*, CAGD, to appear.

[13] K. Höllig: *Finite Element Methods with B–Splines*, SIAM, 2003.

[14] K. Höllig and J. Hörner: *Approximation and Modeling with B–Splines*, SIAM, 2013.

[15] K. Höllig and J. Hörner: *Programming Finite Element Methods with B–Splines*, to appear in: Comput. Math. Appl. (2015).

[16] K. Höllig, J. Hörner, and A. Hoffacker: *Finite element analysis with B–splines: weighted and isogeometric methods*, in: Curves and Surfaces 2010, J.D. Boissonnat et al. (eds.), LNCS 6920, Springer (2012), 330–350.

[17] K. Höllig, U. Reif, and J. Wipper: *Weighted extended B–spline approximation of Dirichlet problems*, SIAM J. Numer. Anal. 39 (2001), 442–462.

[18] T.J.R. Hughes, J.A. Cottrell, and Y. Bazilevs: *Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement*, Comput. Methods Appl. Mech. Engrg. 194 (2005), 4135–4195.

[19] T. Ito: *A collocation method for boundary–value problems using spline functions*, Dissertation, Brown University, 1972.

[20] L.V. Kantorovich: *Sur une méthode de resolution approchée d'equations differentielles aux derivées partielles*, C. R. Acad. Sci. URSS 2 (1934), 534-536.

[21] L.V. Kantorovich and V.I. Krylov: *Approximate Methods of Higher Analysis*, Interscience Publishers, 1958.

[22] E.B. Karpilovskaya: *О сходимости интерполяционного метода для обыкновенных дифференциальных уравнений*, Uspekhi Mat. Nauk 8 (1953), 111–118.

[23] R. Kraft: *Adaptive und linear unabhängige Multilevel B–Splines und ihre Anwendungen*, Dissertation, University of Stuttgart, 1998.

[24] D. Mokris and B. Jüttler: *TDHB–splines: The truncated decoupled basis of hierarchical tensor–product splines*, CAGD 31 (2014), 531–544.

[25] P. Percell, M.F. Wheeler: *A C^1 finite element collocation method for elliptic equations*, SIAM J. Numer. Anal. 17 (1980), 605–622.

[26] L. Piegl and W. Tiller: *The NURBS Book*, Springer, 1997.

[27] P.M. Prenter, R.D. Russell: *Orthogonal collocation for elliptic partial differential equations*, SIAM J. Numer. Anal. 13 (1976), 923–939.

[28] R. D. Russell, L. F. Shampine: *A collocation method for boundary value problems*, Numer. Math. 19 (1972), 1-28.

[29] V.L. Rvachev: *Method of R-functions in boundary-value problems*, Int. Appl. Mech. 11 (1975), 345–354.

[30] V.L. Rvachev and T.I. Sheiko: *R-functions in boundary value problems in mechanics*, Appl. Mech. Rev. 48 (1995), 151–188.

[31] V.L. Rvachev, T.I. Sheiko, V. Shapiro and I. Tsukanov: *On completeness of RFM solution structures*, Comput. Mech. 25 (2000), 305–316.

[32] D. Schillinger, J.A. Evans, A. Reali, M.A. Scott, T.J.R. Hughes: *Isogeometric collocation: Cost comparison with Galerkin methods and extension to adaptive hierarchical NURBS discretizations*, Comput. Meth. Appl. Mech. Engrg. 267 (2013), 170-232.

[33] L.L. Schumaker: *Spline Functions: Basic Theory*, Wiley–Interscience, 1980.

[34] V. Shapiro: *Semi–analytic geometry with R–functions*, Acta Numer. 16 (2007), 239–303.

[35] G.M. Vainikko: *On convergence and stability of the collocation method*, Differential Equations 1 (1965), 186–194.

[36] K. Wright: *Chebyshev collocation methods for ordinary differential equations*, Comput. J. 6 (1963/1964), 358–365.

Christian Apprich
IMNG, Fachbereich Mathematik, Universität Stuttgart
Pfaffenwaldring 57, 70569 Stuttgart, Germany
E-Mail: apprich@mathematik.uni-stuttgart.de

Klaus Höllig
IMNG, Fachbereich Mathematik, Universität Stuttgart
Pfaffenwaldring 57, 70569 Stuttgart, Germany
E-Mail: hoellig@mathematik.uni-stuttgart.de

Jörg Hörner
IMNG, Fachbereich Mathematik, Universität Stuttgart
Pfaffenwaldring 57, 70569 Stuttgart, Germany
E-Mail: hoerner@mathematik.uni-stuttgart.de

Ulrich Reif
AG Geometrie & Approximation, Technische Universität Darmstadt
Schlossgartenstr. 7, 64289 Darmstadt, Germany
E-Mail: reif@mathematik.tu-darmstadt.de

Erschienene Preprints ab Nummer 2012-001

Komplette Liste: `http://www.mathematik.uni-stuttgart.de/preprints`

2015-009 *Apprich, C.; Höllig, K.; Hörner, J.; Reif, U.:* Collocation with WEB–Splines

2015-008 *Kabil, B.; Rodrigues, M.:* Spectral Validation of the Whitham Equations for Periodic Waves of Lattice Dynamical Systems

2015-007 *Kollross, A.:* Hyperpolar actions on reducible symmetric spaces

2015-006 *Schmid, J.; Griesemer, M.:* Well-posedness of Non-autonomous Linear Evolution Equations in Uniformly Convex Spaces

2015-005 *Hinrichs, A.; Markhasin, L.; Oettershagen, J.; Ullrich, T.:* Optimal quasi-Monte Carlo rules on higher order digital nets for the numerical integration of multivariate periodic functions

2015-004 *Kutter, M.; Rohde, C.; Sändig, A.-M.:* Well-Posedness of a Two Scale Model for Liquid Phase Epitaxy with Elasticity

2015-003 *Rossi, E.; Schleper, V.:* Convergence of a numerical scheme for a mixed hyperbolic-parabolic system in two space dimensions

2015-002 *Döring, M.; Györfi, L.; Walk, H.:* Exact rate of convergence of kernel-based classification rule

2015-001 *Kohler, M.; Müller, F.; Walk, H.:* Estimation of a regression function corresponding to latent variables

2014-021 *Neusser, J.; Rohde, C.; Schleper, V.:* Relaxed Navier-Stokes-Korteweg Equations for Compressible Two-Phase Flow with Phase Transition

2014-020 *Kabil, B.; Rohde, C.:* Persistence of undercompressive phase boundaries for isothermal Euler equations including configurational forces and surface tension

2014-019 *Bilyk, D.; Markhasin, L.:* BMO and exponential Orlicz space estimates of the discrepancy function in arbitrary dimension

2014-018 *Schmid, J.:* Well-posedness of non-autonomous linear evolution equations for generators whose commutators are scalar

2014-017 *Margolis, L.:* A Sylow theorem for the integral group ring of $PSL(2,q)$

2014-016 *Rybak, I.; Magiera, J.; Helmig, R.; Rohde, C.:* Multirate time integration for coupled saturated/unsaturated porous medium and free flow systems

2014-015 *Gaspoz, F.D.; Heine, C.-J.; Siebert, K.G.:* Optimal Grading of the Newest Vertex Bisection and H^1-Stability of the L_2-Projection

2014-014 *Kohler, M.; Krzyżak, A.; Walk, H.:* Nonparametric recursive quantile estimation

2014-013 *Kohler, M.; Krzyżak, A.; Tent, R.; Walk, H.:* Nonparametric quantile estimation using importance sampling

2014-012 *Györfi, L.; Ottucsák, G.; Walk, H.:* The growth optimal investment strategy is secure, too.

2014-011 *Györfi, L.; Walk, H.:* Strongly consistent detection for nonparametric hypotheses

2014-010 *Köster, I.:* Finite Groups with Sylow numbers $\{q^x, a, b\}$

2014-009 *Kahnert, D.:* Hausdorff Dimension of Rings

2014-008 *Steinwart, I.:* Measuring the Capacity of Sets of Functions in the Analysis of ERM

2014-007 *Steinwart, I.:* Convergence Types and Rates in Generic Karhunen-Loève Expansions with Applications to Sample Path Properties

2014-006 *Steinwart, I.; Pasin, C.; Williamson, R.; Zhang, S.:* Elicitation and Identification of Properties

2014-005 *Schmid, J.; Griesemer, M.:* Integration of Non-Autonomous Linear Evolution Equations

2014-004 *Markhasin, L.:* L_2- and $S^r_{p,q}B$-discrepancy of (order 2) digital nets

2014-003 *Markhasin, L.:* Discrepancy and integration in function spaces with dominating mixed smoothness

2014-002 *Eberts, M.; Steinwart, I.:* Optimal Learning Rates for Localized SVMs

2014-001 *Giesselmann, J.:* A relative entropy approach to convergence of a low order approximation to a nonlinear elasticity model with viscosity and capillarity

2013-016 *Steinwart, I.:* Fully Adaptive Density-Based Clustering

2013-015 *Steinwart, I.:* Some Remarks on the Statistical Analysis of SVMs and Related Methods

2013-014 *Rohde, C.; Zeiler, C.:* A Relaxation Riemann Solver for Compressible Two-Phase Flow with Phase Transition and Surface Tension

2013-013 *Moroianu, A.; Semmelmann, U.:* Generalized Killling spinors on Einstein manifolds

2013-012 *Moroianu, A.; Semmelmann, U.:* Generalized Killing Spinors on Spheres

2013-011 *Kohls, K; Rösch, A.; Siebert, K.G.:* Convergence of Adaptive Finite Elements for Control Constrained Optimal Control Problems

2013-010 *Corli, A.; Rohde, C.; Schleper, V.:* Parabolic Approximations of Diffusive-Dispersive Equations

2013-009 *Nava-Yazdani, E.; Polthier, K.:* De Casteljau's Algorithm on Manifolds

2013-008 *Bächle, A.; Margolis, L.:* Rational conjugacy of torsion units in integral group rings of non-solvable groups

2013-007 *Knarr, N.; Stroppel, M.J.:* Heisenberg groups over composition algebras

2013-006 *Knarr, N.; Stroppel, M.J.:* Heisenberg groups, semifields, and translation planes

2013-005 *Eck, C.; Kutter, M.; Sändig, A.-M.; Rohde, C.:* A Two Scale Model for Liquid Phase Epitaxy with Elasticity: An Iterative Procedure

2013-004 *Griesemer, M.; Wellig, D.:* The Strong-Coupling Polaron in Electromagnetic Fields

2013-003 *Kabil, B.; Rohde, C.:* The Influence of Surface Tension and Configurational Forces on the Stability of Liquid-Vapor Interfaces

2013-002 *Devroye, L.; Ferrario, P.G.; Györfi, L.; Walk, H.:* Strong universal consistent estimate of the minimum mean squared error

2013-001 *Kohls, K.; Rösch, A.; Siebert, K.G.:* A Posteriori Error Analysis of Optimal Control Problems with Control Constraints

2012-013 *Diaz Ramos, J.C.; Dominguez Vázquez, M.; Kollross, A.:* Polar actions on complex hyperbolic spaces

2012-012 *Moroianu; A.; Semmelmann, U.:* Weakly complex homogeneous spaces

2012-011 *Moroianu; A.; Semmelmann, U.:* Invariant four-forms and symmetric pairs

2012-010 *Hamilton, M.J.D.:* The closure of the symplectic cone of elliptic surfaces

2012-009 *Hamilton, M.J.D.:* Iterated fibre sums of algebraic Lefschetz fibrations

2012-008 *Hamilton, M.J.D.:* The minimal genus problem for elliptic surfaces

2012-007 *Ferrario, P.:* Partitioning estimation of local variance based on nearest neighbors under censoring

2012-006 *Stroppel, M.:* Buttons, Holes and Loops of String: Lacing the Doily

2012-005 *Hantsch, F.:* Existence of Minimizers in Restricted Hartree-Fock Theory

2012-004 *Grundhöfer, T.; Stroppel, M.; Van Maldeghem, H.:* Unitals admitting all translations

2012-003 *Hamilton, M.J.D.:* Representing homology classes by symplectic surfaces

2012-002 *Hamilton, M.J.D.:* On certain exotic 4-manifolds of Akhmedov and Park

2012-001 *Jentsch, T.:* Parallel submanifolds of the real 2-Grassmannian