Oden Institute REPORT 19-08

May 2019

A Tchebycheffian Extension of Multi-Degree B-Splines: Algorithmic Computation and Properties

by

Rene R. Hiemstra, Thomas J.R. Hughes, Carla Manni, Hendrik Speleers, and Deepesh Toshniwal

Oden Institute for Computational Engineering and Sciences
The University of Texas at Austin
Austin, Texas 78712

Reference: Rene R. Hiemstra, Thomas J.R. Hughes, Carla Manni, Hendrik Speleers, and Deepesh Toshniwal, "A Tchebycheffian Extension of Multi-Degree B-Splines: Algorithmic Computation and Properties," Oden Institute REPORT 19-08, Oden Institute for Computational Engineering and Sciences, The University of Texas at Austin, May 2019.

A TCHEBYCHEFFIAN EXTENSION OF MULTI-DEGREE B-SPLINES: ALGORITHMIC COMPUTATION AND PROPERTIES*

RENÉ R. HIEMSTRA†, THOMAS J. R. HUGHES†, CARLA MANNI‡, HENDRIK SPELEERS‡, AND DEEPESH TOSHNIWAL†§

Abstract. In this paper we present an efficient and robust approach to compute a normalized B-spline-like basis for spline spaces with pieces drawn from extended Tchebycheff spaces. The extended Tchebycheff spaces and their dimensions are allowed to change from interval to interval. The approach works by constructing a matrix that maps a generalized Bernstein-like basis to the B-spline-like basis of interest. The B-spline-like basis shares many characterizing properties with classical univariate B-splines and may easily be incorporated in existing spline codes. This may contribute to the full exploitation of Tchebycheffian splines in applications, freeing them from the restricted role of an elegant theoretical extension of polynomial splines. Numerical examples are provided that illustrate the procedure described.

Key words. Tchebycheffian splines, Multi-degree B-splines, Generalized B-splines, Extraction algorithms

AMS subject classifications. 41A15, 41A50, 65D07, 65D15

1. Introduction.

In the classical polynomial setting, univariate multi-degree splines are piecewise polynomial functions that are glued together in a certain smooth way and where the various pieces can have different degrees [2, 34]. This multi-degree formulation offers significant advantages with respect to the classical uniform-degree case, allowing for the modeling of complex geometries with fewer control points and more versatile adaptive schemes in numerical simulation [33, 37].

Polynomial splines, in both the uniform-degree or multi-degree version, can be seen as a special case of Tchebycheffian splines [9, 26, 29, 30], i.e., smooth piecewise functions whose pieces are drawn from extended Tchebycheff spaces (ET-spaces). ET-spaces are natural generalizations of algebraic polynomial spaces [14, 30] because they satisfy the same bounds on the number of zeros of non-trivial elements. Relevant examples of ET-spaces are nullspaces of linear differential operators on suitable intervals [10, 30]. Tchebycheffian splines share many properties with the classical polynomial splines but also offer a much more flexible framework, due to the wide diversity of ET-spaces. Multivariate extensions of Tchebycheffian splines can be easily obtained via (local) tensor-product structures [6, 7, 8].

The rich variety of parameters in Tchebycheffian spline spaces has been explored in free-form design and constrained interpolation/approximation; see [11, 18, 27, 31, 39] and references therein. In addition, Tchebycheffian splines emerge as a natural tool in several engineering contexts. Among others, Tchebycheffian splines based on trigonometric and/or exponential functions allow for an exact representation of conic sections with (almost) arc-length parameterization, without the need for a rational form [19]. As a consequence, their elegant behavior with respect to differentiation and integration makes them an appealing substitute for the rational NURBS model in the framework of both Galerkin and collocation isogeometric methods [1, 20, 22, 23].

*Submitted to the editors DATE.

†Oden Institute for Computational Engineering and Sciences, University of Texas at Austin, USA (rene@oden.utexas.edu, hughes@oden.utexas.edu, deepesh@utexas.edu).

‡Department of Mathematics, University of Rome Tor Vergata, Italy (manni@mat.uniroma2.it, speleers@mat.uniroma2.it).

§Delft Institute of Applied Mathematics, Delft University of Technology, The Netherlands

When the geometry is not an issue, Tchebycheffian splines can still provide an interesting problem-dependent alternative to classical polynomial B-splines/NURBS for solving differential problems: they allow for an efficient treatment of sharp gradients and thin layers [21, 22] and are able to outperform classical polynomial B-splines in the spectral approximation of differential operators [22, 23].

The success of polynomial splines greatly relies on the famous B-spline basis which can also be defined in the multi-degree setting [2, 29, 34, 35, 37]. Most of the results known for polynomial splines extend in a natural way to Tchebycheffian splines. However, the possibility of representing the space in terms of a basis with similar properties to polynomial B-splines is not always guaranteed, even for pieces taken from ET-spaces of the same dimension. More precisely, there are two main categories of Tchebycheffian splines: the various pieces are drawn either from the same ET-space — see [29] for a proper meaning in case of different local dimensions — or from different ET-spaces. In the former case, Tchebycheffian splines always admit a representation in terms of basis functions with similar properties to polynomial B-splines. The latter offers a much more general framework — sometimes referred to in the literature as piecewise Tchebycheffian splines [27] — and allows us to optimally benefit from the great diversity of ET-spaces, but the existence of a B-spline-like basis requires constraints on the various ET-spaces.

Tchebycheffian splines can be easily incorporated in existing spline codes because the corresponding B-spline-like basis, whenever it exists, is compatible with classical B-splines as it enjoys the same structural properties. When all the ET-spaces have the same dimension, various approaches have been used in the Tchebycheffian setting to construct such a B-spline-like basis: generalized divided differences [28, 30], Hermite interpolation [9, 29], integral recurrence relations [3, 17], de Boor-like recurrence relations [12, 15], and blossoming [26]. Each of these definitions has advantages according to the problem one has to face or to the properties to be proved. All these constructions lead to the same functions, up to a proper scaling. For non-uniform local dimensions, the literature is much less developed and B-spline-like bases have been constructed via Hermite interpolation [9, 29].

Unfortunately, none of the currently available constructions for B-spline-like bases of Tchebycheffian spline spaces is very well suited for their efficient and robust numerical evaluation and manipulation, due to computational complexity and/or numerical instabilities. This drawback has seriously penalized Tchebycheffian splines, so far, in practical applications despite their great potential, and has confined them mostly to the role of an elegant theoretical extension of the polynomial case.

This paper focuses on the construction, properties, and evaluation of a basis with similar properties to polynomial B-splines for Tchebycheffian splines with pieces drawn from different ET-spaces of possibly different dimension, whenever it exists. Following [29], we refer to these spaces as *generalized Tchebycheffian splines (GT-splines)* and to the corresponding basis as *generalized Tchebycheffian B-splines (GTB-splines)*.

The main goal of the paper is to present an efficient and robust algorithm for evaluation of GTB-splines. The algorithm proceeds by incrementally increasing the smoothness at the breakpoints starting from the space of piecewise discontinuous functions obtained by collecting the various ET-spaces which are represented in terms of a Bernstein-like basis. At each step, the smoothness constraints are represented in the form of a matrix whose nullspace identifies the basis elements. The algorithm explicitly constructs this nullspace without solving any linear systems. In other words, at each step, the algorithm explicitly constructs a matrix that specifies how GTB-splines that are C^r at some breakpoint can be linearly combined to form GTB-splines

that are C^{r+1}. The product of all such matrices is called an *extraction operator* and it maps the Bernstein-like basis to GTB-splines. In fact, we prove that the output of the algorithm is exactly the entire set of GTB-splines that span the considered GT-spline space. The algorithm can be seen as a Tchebycheffian extension of the one proposed in [36, 37, 38] for multi-degree polynomial splines. In order to ensure existence of a GTB-spline basis, we consider the sufficient conditions proposed in [9] which can be easily checked and are satisfied for a wide class of GT-splines of interest in applications. However, this is not a limitation of the algorithm we are proposing; the algorithm produces the required GTB-spline basis whenever it exists.

The main goal is complemented by additional results: we provide a knot insertion formula and a global integral recurrence relation for GTB-splines. While the former is in complete analogy with the one known for the multi-degree polynomial case [2, 38], the latter is a new contribution also for the multi-degree polynomial case, where only local integral recurrence relations have been proposed so far in the literature [2, 34]. The provided global integral recurrence relation completely mimics the one known for polynomial/Tchebycheffian splines of uniform degree/local dimension and is expressed in an elegant way by using an extension of the concept of weights.

The remainder of the paper is organized as follows. Section 2 recalls several properties of ET-spaces, introduces notation, and defines the space of GT-splines. The existence, under proper assumptions, of GTB-splines is summarized in Section 3; it basically collects in a homogeneous and self-contained presentation results from [9, 29]. Section 4 presents local and global integral recurrence relations for GTB-splines, while Section 5 is devoted to the knot insertion formula which is the main ingredient for the evaluation algorithm described and analyzed in Section 6. An interesting case study is detailed in Section 7, and some numerical examples are collected in Section 8. We end with some concluding remarks in Section 9.

2. Preliminaries. We are interested in piecewise functions, whose pieces belong to ET-spaces and are glued together in a certain smooth way. We first define ET-spaces on a real interval J (see, e.g., [30]).

DEFINITION 2.1 (Extended Tchebycheff space). *Given an interval J, a space $\mathbb{T}_p(J) \subset C^p(J)$ of dimension $p+1$ is an extended Tchebycheff (ET-) space on J if any Hermite interpolation problem with $p+1$ data on J has a unique solution in $\mathbb{T}_p(J)$. In other words, for any integer $m \geq 1$, let $\bar{x}_1, \ldots, \bar{x}_m$ be distinct points in J and let $d_1, \ldots, d_m$ be nonnegative integers such that $p+1 = \sum_{i=1}^{m}(d_i+1)$. Then, for any set $\{f_{i,j} \in \mathbb{R}\}_{i=1,\ldots,m,\, j=0,\ldots,d_i}$ there exists a unique $q \in \mathbb{T}_p(J)$ such that*

$$D^j q(\bar{x}_i) = f_{i,j}, \quad i = 1, \ldots, m, \quad j = 0, \ldots, d_i.$$

If J is a bounded closed interval, then any ET-space of dimension $p+1$ on J is an extended complete Tchebycheff (ECT-) space on J, i.e., it is spanned by the following functions (see [24, 30]):

$$\begin{cases} u_0(x) := w_0(x), \\ u_1(x) := w_0(x) \int_z^x w_1(y_1)\mathrm{d}y_1, \\ \qquad \vdots \\ u_p(x) := w_0(x) \int_z^x w_1(y_1) \int_z^{y_1} \cdots \int_z^{y_p} w_p(y_p)\mathrm{d}y_p \cdots \mathrm{d}y_1, \end{cases} \tag{2.1}$$

where z is any point in J and $w_j \in C^{p-j}(J)$, $j = 0, 1, \ldots, p$ are positive functions called *weights*. The functions $u_0, \ldots, u_p$ are called *generalized powers*.

Remark 2.2. A given ECT-space can be identified by different sets of weights; see [17] for details and examples. In particular, it is clear that the two weight systems

$$w_0, \dots, w_p \quad \text{and} \quad K_0 w_0, \dots, K_p w_p,$$

where $K_0, \dots, K_p$ are positive constants, identify the same ECT-space.

Remark 2.3. A very important case for applications is $w_0 = 1$ with $p \geq 1$. Under such assumptions, it can be directly checked that

$$\begin{cases} Du_0(x) = 0, \\ Du_1(x) = w_1(x), \\ \qquad \vdots \\ Du_p(x) = w_1(x) \int_z^x \cdots \int_z^{y_p} w_p(y_p) \mathrm{d}y_p \cdots \mathrm{d}y_2, \end{cases} \tag{2.2}$$

i.e., the space spanned by the derivatives of the functions in (2.1) is an ECT-space of dimension p on J and it is identified by the weights $w_1, \dots, w_p$.

Remark 2.4. The polynomial space of degree p fits in this framework by taking $w_0 = \cdots = w_p = 1$. In this case, the functions in (2.1) become $u_j = (x-z)^j/(j!)$, which are the standard (polynomial) power functions.

Pieces of our splines shall be drawn from arbitrary ECT-spaces of possibly different dimensions. Consider a partitioning, Δ, of the interval $[a, b] \subset \mathbb{R}$ into a sequence of breakpoints,

$$\Delta := \{\, a =: x_0 < x_1 < \cdots < x_{m-1} < x_m := b \,\}.$$

Furthermore, we set $J_i := [x_{i-1}, x_i)$, $i = 1, \dots, m-1$, and $J_m := [x_{m-1}, x_m]$. We also define an ECT-space of dimension p_i+1 on each closed interval $[x_{i-1}, x_i]$, $i = 1, \dots, m$:

$$\mathbb{T}_{p_i}^{(i)} := \operatorname{span}\left\{u_0^{(i)}, \dots, u_{p_i}^{(i)}\right\}, \quad u_j^{(i)} \in C^{p_i}([x_{i-1}, x_i]), \quad j = 0, \dots, p_i,$$

where $u_0^{(i)}, \dots, u_{p_i}^{(i)}$ are generalized powers defined in terms of positive weights $w_j^{(i)} \in C^{p_i - j}([x_{i-1}, x_i])$, $j = 0 \dots, p_i$ as in (2.1). Collectively, these functions span the following space:

$$\mathbb{S}^{\boldsymbol{p}}(\Delta) := \left\{[a, b] \xrightarrow{s} \mathbb{R} : \, s|_{J_i} \in \mathbb{T}_{p_i}^{(i)}, \; i = 1, \dots, m\right\}.$$

In order to measure smoothness at the breakpoints we define the following jump operator for a given $s \in \mathbb{S}^{\boldsymbol{p}}(\Delta)$,

$$\operatorname{Jump}_{x_i, k}(s) := D_-^k s(x_i) - D_+^k s(x_i).$$

Then, we can define the space of generalized Tchebycheffian splines as follows.

Definition 2.5 (Generalized Tchebycheffian splines). *Given the sets of integers* $\boldsymbol{p} := \{p_1, \dots, p_m\}$ *and*

$$\boldsymbol{r} := \{r_i \in \mathbb{Z} : -1 \leq r_i \leq \min\{p_i, p_{i+1}\}, \; i = 1, \dots, m-1, \; r_0 = r_m = -1\}, \tag{2.3}$$

we define

$$\mathbb{S}_{\boldsymbol{r}}^{\boldsymbol{p}}(\Delta) := \left\{s \in \mathbb{S}^{\boldsymbol{p}}(\Delta) : \operatorname{Jump}_{x_i, j}(s) = 0, \; j = 0, \dots, r_i \text{ and } i = 1, \dots, m-1\right\}. \tag{2.4}$$

The value r_i represents the smoothness at breakpoint x_i, $i = 1, \dots, m-1$. The set of linear constraints, encoded in $\mathrm{Jump}_{x_i,j}(s) = 0$, enforce the prescribed smoothness in between adjoining elements of the partition. This leads to a system of $\phi := \sum_{i=1}^{m-1}(r_i+1)$ equations in $\theta := \sum_{i=1}^{m}(p_i+1)$ unknowns. All smoothness conditions are linearly independent because the functions $\{u_0^{(i)}(x), \dots, u_{p_i}^{(i)}(x)\}$ on each interval J_i form an ECT-system [9, 29]. This leads to the following dimension result; see also [9, Theorem 1.1].

COROLLARY 2.6. *The dimension of* $\mathbb{S}_{\boldsymbol{r}}^{\boldsymbol{p}}(\Delta)$ *is*

$$n := \theta - \phi = p_1 + 1 + \sum_{i=1}^{m-1}(p_{i+1} - r_i) = p_m + 1 + \sum_{i=1}^{m-1}(p_i - r_i).$$

Whenever we deal with a single weight system $w_0, \dots, w_p$, $p := \max_{1\le i\le m} p_i$, defined on the entire interval $[a, b]$, all the pieces of the spline functions are basically taken from the "same" ECT-space, possibly allowing different local dimensions. In this case, the spline space in (2.4) is quite well understood and it enjoys all the nice properties of standard polynomial splines; see [16, 26, 30] and references therein for the case where all the local spaces have the same dimension, and [29] for non-uniform dimensions. On the other hand, in order to fully exploit the richness and the variety of ECT-spaces, it is of interest to consider different ECT-spaces on different intervals. In this much more general framework, obtaining spline spaces equipped with the same properties as standard polynomial splines, including a B-spline-like basis, entails constraints on the various ECT-spaces which can be described in terms of reciprocal smoothness of the associated weight systems. From this perspective, we consider the following definition, which is equivalent to the requirement on the weights in [9, Lemma 2.7] taking into account Remark 2.2.

DEFINITION 2.7 (Admissible weights). *The weight systems* $\{w_j^{(i)},\ j = 0, \dots, p_i\}$ *generating the ECT-spaces* $\mathbb{T}_{p_i}^{(i)}$, $i = 1, \dots, m$, *are admissible for the space* $\mathbb{S}_{\boldsymbol{r}}^{\boldsymbol{p}}(\Delta)$ *if for* $i = 1, \dots, m-1$ *and* $j = 0, \dots, r_i$ *we have*

$$D_-^l w_j^{(i)}(x_i) = D_+^l w_j^{(i+1)}(x_i), \quad l = 0, \dots, r_i - j.$$

Remark 2.8. How to construct the weights is well known for a single ECT-space [13, 26] but it can be an issue whenever different ECT-spaces are considered [27]. However, there are cases where admissible weights, according to Definition 2.7, can be easily constructed. For example, they can be obviously extracted from a single weight system such that all the pieces are drawn from the same ECT-space. Furthermore, they can be easily deduced for an interesting class of Tchebycheffian splines which allows for the use of different ECT-spaces, the so-called *generalized polynomial splines*; see section 7. For handling more general settings, one could apply the constructive procedure for finding all weight systems associated with a given ECT-space in a bounded closed interval presented in [25].

Remark 2.9. Dealing with admissible weights gives only a sufficient condition for obtaining Tchebycheffian splines equipped with a B-spline-like basis; see [9] and also the next section. The simplicity of this condition and the fact that it embraces relevant classes of Tchebycheffian splines motivate our choice. We refer the reader to [27] for explicit necessary and sufficient conditions for smoothly gluing together ECT-spaces of dimension 5.

In the next section we summarize from the literature some properties of GT-spline spaces. In particular, we show that, under certain assumptions on the weights, a GT-spline space admits a B-spline-like basis and we present some properties of these basis functions.

3. Generalized Tchebycheffian B-splines. In this section we introduce basis functions for the GT-spline space $\mathbb{S}_{\boldsymbol{r}}^{\boldsymbol{p}}(\Delta)$ that possess all the characterizing properties of standard polynomial B-splines. B-spline-like bases can be defined under different normalizations. Here, in this paper, we mainly focus on the partition-of-unity normalization, and we call the corresponding functions *Generalized Tchebycheffian B-splines (GTB-splines)*. The material we are going to present in the current section is greatly inspired by the results in [9, 29]. We provide a concise summary, aiming for a self-contained presentation and a unified notation tailored for the subsequent sections. Furthermore, we detail the proofs of results which are not explicitly presented in the literature in the form we need.

We begin by introducing some notation and several results that assist in characterizing GTB-splines. Similar to polynomial B-splines, GTB-splines can be defined using certain knot vectors. To allow for ECT-spaces of varying dimension, it is convenient to consider two knot vectors,

$$\boldsymbol{u} := (u_k)_{k=1}^{n} := (\underbrace{x_0,\ \ldots,\ x_0}_{p_1-r_0 \text{ times}},\ \ldots,\ \underbrace{x_i,\ \ldots,\ x_i}_{p_{i+1}-r_i \text{ times}},\ \ldots,\ \underbrace{x_{m-1},\ \ldots,\ x_{m-1}}_{p_m-r_{m-1} \text{ times}}), \tag{3.1a}$$

$$\boldsymbol{v} := (v_k)_{k=1}^{n} := (\underbrace{x_1,\ \ldots,\ x_1}_{p_1-r_1 \text{ times}},\ \ldots,\ \underbrace{x_i,\ \ldots,\ x_i}_{p_i-r_i \text{ times}},\ \ldots,\ \underbrace{x_m,\ \ldots,\ x_m}_{p_m-r_m \text{ times}}). \tag{3.1b}$$

Remark 3.1. With respect to standard polynomial splines of uniform degree p, the above vectors are related to the standard B-spline knot vector, $\{\xi_k,\ k = 1, \ldots, n+p\}$, via the relationship, $u_k = \xi_k$ and $v_k = \xi_{k+p+1}$, $k = 1, \ldots, n$.

Remark 3.2. The use of the two knot vectors (3.1a) and (3.1b) greatly simplifies the presentation of the properties of GTB-splines. These two knot vectors have been introduced in [9] to characterize properly posed Hermite interpolation problems in GT-spline spaces and have been recently exploited to describe B-spline-like bases for multi-degree polynomial splines in [2, 38].

Mimicking the polynomial spline setting, each of the intervals

$$[u_k, v_k], \quad k = 1, \ldots, n,$$

corresponds to the support of a B-spline-like basis function in $\mathbb{S}_{\boldsymbol{r}}^{\boldsymbol{p}}(\Delta)$. Lemma 3.3 implies that these intervals are non-empty and satisfy $[u_k, v_k] \cup [u_{k+1}, v_{k+1}] = [u_k, v_{k+1}]$ and $[u_k, v_k] \cap [u_{k+1}, v_{k+1}] = [u_{k+1}, v_k]$. Lemma 3.4 shows that there are $p_i + 1$ of such intervals that intersect with element $[x_{i-1}, x_i)$. Before proving these lemmas, we define two types of quantities,

$$\mu_{\boldsymbol{u}}(i) := \sum_{j=0}^{i-1} (p_{j+1} - r_j), \quad \mu_{\boldsymbol{v}}(i) := \sum_{j=1}^{i} (p_j - r_j), \quad i = 0, \ldots, m,$$

where an empty sum is assumed to be zero.

LEMMA 3.3. *Let $k \in \{1, \ldots, n\}$ be arbitrary. It holds that $u_k \leq v_{k-1}$ and $u_k < v_k$. Additionally, if $r_i \geq 0$ for all $i \in \{1, \ldots, m-1\}$, then $u_k < v_{k-1} \leq v_k$.*

Proof. Let $k = \mu_{\boldsymbol{u}}(i) + l$, where $1 \le l \le p_{i+1} - r_i$. By inspection, $u_k = x_i$. On the other hand,

$$k = \mu_{\boldsymbol{u}}(i) + l = \sum_{j=0}^{i-1}(p_{j+1} - r_j) + l = \sum_{j=1}^{i}(p_j - r_j) + \sum_{j=0}^{i-1}(r_{j+1} - r_j) + l$$
$$= \mu_{\boldsymbol{v}}(i) + r_i - r_0 + l \ge \begin{cases} \mu_{\boldsymbol{v}}(i) + 2, & r_i \ge 0, \\ \mu_{\boldsymbol{v}}(i) + 1, & r_i \ge -1, \end{cases}$$

since $r_0 = -1$ and $l \ge 1$. By inspection, $v_k \ge x_{i+1}$. Similarly,

$$v_{k-1} \ge \begin{cases} x_{i+1}, & r_i \ge 0, \\ x_i, & r_i \ge -1. \end{cases}$$

The above proves the lemma since $x_{i+1} > x_i$. ∎

LEMMA 3.4.

$$(u_k, v_k) \cap (x_{i-1}, x_i) = \begin{cases} (x_{i-1}, x_i), & k = \mu_{\boldsymbol{u}}(i) - p_i, \dots, \mu_{\boldsymbol{u}}(i), \\ \emptyset, & \textit{otherwise.} \end{cases}$$

Proof. It follows from the previous lemma that the intervals (u_k, v_k), $k = 1, \dots, n$, are non-empty and satisfy

$$\bigcap_{k=k_1}^{k_2} (u_k, v_k) = (u_{k_2}, v_{k_1}).$$

The minimum k_1 and maximum k_2 at which $(u_{k_2}, v_{k_1}) = (x_{i-1}, x_i)$ can be found by inspecting the knot vectors (3.1a) and (3.1b). It follows that $k_2 = \mu_{\boldsymbol{u}}(i)$ and $k_1 = \mu_{\boldsymbol{v}}(i-1) + 1 = \mu_{\boldsymbol{u}}(i) - p_i$. ∎

In order to measure the local continuity of B-spline-like basis functions at break points, we define the following quantities. For $k = 1, \dots, n$, let $u_k = x_i$ and $v_k = x_j$, and set

$$\begin{aligned} r_{\boldsymbol{u}}(k) &:= p_{i+1} - 1 - \max\{l \ge 0 : u_k = u_{k+l}\}, \\ r_{\boldsymbol{v}}(k) &:= p_j - 1 - \max\{l \ge 0 : v_k = v_{k-l}\}. \end{aligned} \tag{3.2}$$

Note that from the knot vector definitions (3.1a) and (3.1b) it can be deduced that

$$\begin{aligned} r_{\boldsymbol{u}}(k) &= r_i + \max\{l \ge 0 : u_k = u_{k-l}\} \ge r_i, \\ r_{\boldsymbol{v}}(k) &= r_j + \max\{l \ge 0 : v_k = v_{k+l}\} \ge r_j. \end{aligned} \tag{3.3}$$

With this notation in place, we can give a local dimension formula.

LEMMA 3.5. *The restriction of the spline space $\mathbb{S}^{\boldsymbol{p}}_{\boldsymbol{r}}(\Delta)$ to the interval $[u_k, v_k]$ has dimension $r_{\boldsymbol{u}}(k) + r_{\boldsymbol{v}}(k) + 3$.*

Proof. Let $u_k = x_i$ and $v_k = x_j$, $j > i$. The restriction of the spline space $\mathbb{S}^{\boldsymbol{p}}_{\boldsymbol{r}}(\Delta)$ to the interval $[u_k, v_k]$ has dimension

$$\sum_{l=i+1}^{j}(p_l + 1) - \sum_{l=i+1}^{j-1}(r_l + 1) = p_j + 1 + \sum_{l=i+1}^{j-1}(p_l - r_l).$$

On the other hand, from (3.1a), (3.1b), and (3.2) we get

$$k + p_{i+1} - 1 - r_{\boldsymbol{u}}(k) = k + \max\{l \geq 0 : u_k = u_{k+l}\} = \sum_{l=0}^{i}(p_{l+1} - r_l),$$

$$k - p_j + 1 + r_{\boldsymbol{v}}(k) = k - \max\{l \geq 0 : v_k = v_{k-l}\} = 1 + \sum_{l=1}^{j-1}(p_l - r_l),$$

thus

$$k - r_{\boldsymbol{u}}(k) = 2 + \sum_{l=1}^{i}(p_l - r_l), \quad k + r_{\boldsymbol{v}}(k) = \sum_{l=1}^{i}(p_l - r_l) + \sum_{l=i+1}^{j-1}(p_l - r_l) + p_j.$$

Subtracting the above expressions gives the result. ∎

We now show the existence of a basis of the space $\mathbb{S}^{\boldsymbol{p}}_{\boldsymbol{r}}(\Delta)$ with several nice properties.

THEOREM 3.6 (Unit-integral B-splines). *Assume there exist admissible weights for the space $\mathbb{S}^{\boldsymbol{p}}_{\boldsymbol{r}}(\Delta)$. Then, there exists a basis of the space $\mathbb{S}^{\boldsymbol{p}}_{\boldsymbol{r}}(\Delta)$ consisting of the functions $\{M_k,\ k = 1, \dots, n\}$, with the following properties:*

$$M_k(x) > 0, \quad \textit{for all } x \in (u_k, v_k), \tag{3.4}$$ *(Non-negativity)*

$$M_k(x) = 0, \quad \textit{for all } x \notin [u_k, v_k], \tag{3.5}$$ *(Local support)*

$$\int_{u_k}^{v_k} M_k(x)\mathrm{d}x = 1, \tag{3.6}$$ *(Unit integral)*

$$M_k \textit{ is exactly } C^{r_{\boldsymbol{u}}(k)} \textit{ at } x = u_k, \tag{3.7}$$ *(Start-point smoothness)*

$$M_k \textit{ is exactly } C^{r_{\boldsymbol{v}}(k)} \textit{ at } x = v_k, \tag{3.8}$$ *(End-point smoothness)*

$$\operatorname{span}\left\{ M_k|_{J_i} \right\}_{k=\mu_{\boldsymbol{u}}(i)-p_i}^{\mu_{\boldsymbol{u}}(i)} \equiv \mathbb{T}^{(i)}_{p_i}. \tag{3.9}$$ *(Local linear independence)*

Proof. Consider the restriction of the spline space $\mathbb{S}^{\boldsymbol{p}}_{\boldsymbol{r}}(\Delta)$ to the interval $[u_k, v_k]$. Since there exist admissible weights for $\mathbb{S}^{\boldsymbol{p}}_{\boldsymbol{r}}(\Delta)$, we know that [9, Theorem 3.1] ensures the existence of a unique function s_k in such space satisfying the following Hermite interpolation problem:

$$\begin{aligned} D^j_+ s_k(u_k) &= 0, \quad j = 0, \dots, r_{\boldsymbol{u}}(k), \\ D^{r_{\boldsymbol{u}}(k)+1}_+ s_k(u_k) &= 1, \\ D^j_- s_k(v_k) &= 0, \quad j = 0, \dots, r_{\boldsymbol{v}}(k). \end{aligned}$$

Moreover, from Lemma 3.5 and [9, Theorem 2.5] it follows that the function s_k has no additional zeros (counting multiplicity) in $[u_k, v_k]$; see also [29, Theorem 4.2]. Therefore, taking

$$M_k(x) := \begin{cases} \dfrac{s_k(x)}{\int_{u_k}^{v_k} s_k(y)\mathrm{d}y}, & u_k \leq x < v_k, \\ 0, & \text{otherwise}, \end{cases}$$

it can be easily checked that M_k satisfies (3.4)–(3.8). Note that (3.3) implies that M_k can have a higher smoothness at the end points of its support (u_k and v_k) than required by the space $\mathbb{S}^{\boldsymbol{p}}_{\boldsymbol{r}}(\Delta)$.

With the same line of arguments of [29, Theorem 4.4] we can show that the functions in the set $\mathcal{M} := \{M_k,\ k = 1, \dots, n\}$ are linearly independent and thus span the space $\mathbb{S}^{\boldsymbol{p}}_{\boldsymbol{r}}(\Delta)$. Finally, Lemma 3.4 shows that $\{M_k,\ k = \mu_{\boldsymbol{u}}(i) - p_i, \dots, \mu_{\boldsymbol{u}}(i)\}$ are the only functions in $\mathcal{M}$ which are non-zero functions on the interval $[x_{i-1}, x_i)$. Consequently, these $p_i + 1$ functions span $\mathbb{T}^{(i)}_{p_i}$ and thus they are locally linearly independent. ∎

Under a proper assumption on the weights related to the local ECT-spaces, one can also define a basis that forms a partition of unity, so mimicking all properties of the standard polynomial B-spline basis obtained by the Cox–de Boor recursion formula [4].

Theorem 3.7 (GTB-splines). *Assume there exist admissible weights for the space $\mathbb{S}^{\boldsymbol{p}}_{\boldsymbol{r}}(\Delta)$ and for all the $\mathbb{T}^{(i)}_{p_i}$ we have*

$$w^{(i)}_0 = 1, \quad i = 1, \dots, m. \tag{3.10}$$

Then, there exists a basis of the space $\mathbb{S}^{\boldsymbol{p}}_{\boldsymbol{r}}(\Delta)$ consisting of the functions $\{N_k,\ k = 1, \dots, n\}$, with the following properties:

$$N_k(x) > 0, \quad \textit{for all } x \in (u_k, v_k), \qquad \textit{(Non-negativity)} \tag{3.11}$$

$$N_k(x) = 0, \quad \textit{for all } x \notin [u_k, v_k], \qquad \textit{(Local support)} \tag{3.12}$$

$$\sum_{k=1}^{n} N_k(x) = 1, \quad \textit{for all } x \in [a, b], \qquad \textit{(Partition of unity)} \tag{3.13}$$

$$N_k \textit{ is exactly } C^{r_{\boldsymbol{u}}(k)} \textit{ at } x = u_k, \qquad \textit{(Start-point smoothness)} \tag{3.14}$$

$$N_k \textit{ is exactly } C^{r_{\boldsymbol{v}}(k)} \textit{ at } x = v_k, \qquad \textit{(End-point smoothness)} \tag{3.15}$$

$$\operatorname{span}\left\{ N_k|_{J_i} \right\}_{k=\mu_{\boldsymbol{u}}(i)-p_i}^{\mu_{\boldsymbol{u}}(i)} \equiv \mathbb{T}^{(i)}_{p_i}. \qquad \textit{(Local linear independence)} \tag{3.16}$$

Proof. Without loss of generality, we can assume $r_i \geq 0$, $i = 1, \dots, m-1$. Indeed, if $r_l = -1$ for some $1 \leq l \leq m-1$ then the spline space $\mathbb{S}^{\boldsymbol{p}}_{\boldsymbol{r}}(\Delta)$ can be decomposed into two disconnected spaces defined on $[x_0, x_l)$ and $[x_l, x_m]$, respectively. Recall that $r_i \geq 0$, $i = 1, \dots, m-1$, implies that $u_k < v_{k-1}$, $k = 1, \dots, n$ (see Lemma 3.3). Furthermore, if $p_i = 0$ then $N_k(x) = 1$ for $x \in [x_{i-1}, x_i) \subseteq [u_k, v_k)$. For $i = 1, \dots, m$, let $\widehat{\mathbb{T}}^{(i)}_{p_i}$ denote the space of the derivatives of the functions belonging to $\mathbb{T}^{(i)}_{p_i}$. For $p_i > 0$, from (2.2) and (3.10) it follows that $\widehat{\mathbb{T}}^{(i)}_{p_i}$ is an ECT-space of dimension p_i on J_i identified by the weights

$$\widehat{w}^{(i)}_j := w^{(i)}_{j+1} \in C^{p_i-1-j}, \quad j = 0 \dots, p_i - 1. \tag{3.17}$$

Consider now the spline space

$$\begin{aligned} \widehat{\mathbb{S}^{\boldsymbol{p}}_{\boldsymbol{r}}}(\Delta) := \Big\{ [a, b] \xrightarrow{s} \mathbb{R} : s|_{J_i} \in \widehat{\mathbb{T}}^{(i)}_{p_i},\ i = 1, \dots, m \ \text{ and} \\ \operatorname{Jump}_{x_i, j}(s) = 0,\ j = 0, \dots, r_i - 1,\ i = 1, \dots, m-1 \Big\}, \end{aligned} \tag{3.18}$$

which has dimension $n - 1$. Since the weights defined in (3.17) are admissible for $\widehat{\mathbb{S}^{\boldsymbol{p}}_{\boldsymbol{r}}}(\Delta)$, we can apply Theorem 3.6 to all (non-trivial) disconnected parts of $\widehat{\mathbb{S}^{\boldsymbol{p}}_{\boldsymbol{r}}}(\Delta)$,

and so the space admits a basis of functions $\{\widehat{M}_k,\ k = 2, \dots, n\}$ such that

$$\widehat{M}_k(x) > 0, \quad \text{for all } x \in (u_k, v_{k-1}), \tag{3.19}$$

$$\widehat{M}_k(x) = 0, \quad \text{for all } x \notin [u_k, v_{k-1}], \tag{3.20}$$

$$\int_{u_k}^{v_{k-1}} \widehat{M}_k(x)\mathrm{d}x = 1, \tag{3.21}$$

$$\widehat{M}_k \text{ is exactly } C^{r_{\boldsymbol{u}}(k)-1} \text{ at } x = u_k, \tag{3.22}$$

$$\widehat{M}_k \text{ is exactly } C^{r_{\boldsymbol{v}}(k-1)-1} \text{ at } x = v_{k-1}. \tag{3.23}$$

We then define the following functions belonging to $\mathbb{S}^{\boldsymbol{p}}_{\boldsymbol{r}}(\Delta)$:

$$\begin{aligned} N_1(x) &:= 1 - \int_a^x \widehat{M}_2(y)\mathrm{d}y, \\ N_k(x) &:= \int_a^x \widehat{M}_k(y)\mathrm{d}y - \int_a^x \widehat{M}_{k+1}(y)\mathrm{d}y, \quad k = 2, \dots, n-1, \\ N_n(x) &:= \int_a^x \widehat{M}_n(y)\mathrm{d}y. \end{aligned}$$

A direct inspection shows that the above functions satisfy (3.13). From (3.20), (3.22), and (3.23), we deduce (3.14) and (3.15). Moreover, (3.20) and (3.22) imply

$$D_+^{r_{\boldsymbol{u}}(k)+1} N_k(u_k) > 0. \tag{3.24}$$

For $k = 1, \dots, n$, from [9, Theorem 2.5] it follows that the function N_k has no additional zeros (counting multiplicity) in $[u_k, v_k]$. Then, (3.24) gives (3.11). Linear independence of the functions $\{N_k,\ k = 1, \dots, n\}$ follows by applying the same line of arguments as in the proof of Theorem 3.6. ∎

Remark 3.8. The assumption in (3.10) is equivalent to the fact that each $\mathbb{T}^{(i)}_{p_i}$ contains the constants for $i = 1, \dots, m$.

Remark 3.9. Each of the splines M_k and N_k, $k = 1, \dots n$, in the space $\mathbb{S}^{\boldsymbol{p}}_{\boldsymbol{r}}(\Delta)$ are uniquely defined, up to a constant multiple, by a triple $([u_k, v_k], r_{\boldsymbol{u}}(k), r_{\boldsymbol{v}}(k))$. Indeed, from Lemma 3.5 it follows that the end-point smoothness conditions (see (3.7)–(3.8) and (3.14)–(3.15)) uniquely determine the spline function, up to a constant multiple. The additional constraint of unit integral (3.6) or partition of unity (3.13), respectively, specifies this constant.

Properties (3.11)–(3.16) are very important in both geometric modeling and isogeometric analysis; they make the set of GTB-splines $\{N_k,\ k = 1, \dots, n\}$ the basis of choice for the space $\mathbb{S}^{\boldsymbol{p}}_{\boldsymbol{r}}(\Delta)$ in those applications. Although the theory of GTB-splines has been established for many years [9, 29], a stable and efficient way for computing the GTB-spline basis functions, and performing fundamental operations such as knot insertion, has been lacking. The next sections, containing the original contribution of the paper, focus on these issues. We start by describing some integral recurrence relations that are suited for symbolic computation, and afterwards we develop a procedure based on knot insertion that is suited for numerical evaluation.

4. Integral recurrence relations. In this section we present some integral recurrence relations which could be used for symbolic computation of the GTB-spline basis $\{N_k,\ k = 1, \dots, n\}$. From the proof of Theorem 3.7 we know that they can be

obtained by integrating certain unit-integral B-splines of lower degree and smoothness $\{\widehat{M}_k,\ k=2,\dots,n\}$. We also note that any set $\{\widehat{N}_k,\ k=2,\dots,n\}$ of functions in the same space satisfying (3.19), (3.20), (3.22), and (3.23) can be easily converted into the unit-integral B-splines by

$$\widehat{M}_k(x) = \frac{\widehat{N}_k(x)}{\int_a^b \widehat{N}_k(y)\,\mathrm{d}y}.$$

These form the ingredients for the considered integral recurrence relations. With $p := \max_{1\le i\le m} p_i$, we will discuss in the following how to recursively construct the GTB-splines $N_k = N_{k,p}$, $k=1,\dots,n$, under the same assumptions as in Theorem 3.7. We will consider a local and a global recursive construction.

We start with a pointwise recurrence. It is a direct extension of the integral relation presented in [2, 38] for polynomial splines of non-uniform degree. For $q = 0,\dots,p$ and $k = p-q+1,\dots,n$, the spline $N_{k,q}$ is supported on the interval $[u_k, v_{k-p+q}]$ and defined at $x \in [x_{i-1}, x_i) \subset [u_k, v_{k-p+q}]$ as follows:

$$N_{k,q}(x) := \begin{cases} w_{p_i}^{(i)}(x), & q = p - p_i, \\ w_{p-q}^{(i)}(x) \displaystyle\int_a^x \left[\frac{N_{k,q-1}(y)}{d_{k,q-1}} - \frac{N_{k+1,q-1}(y)}{d_{k+1,q-1}} \right] \mathrm{d}y, & q > p - p_i, \\ 0, & \text{otherwise}, \end{cases} \tag{4.1}$$

where

$$d_{j,q-1} := \int_a^b N_{j,q-1}(y)\,\mathrm{d}y. \tag{4.2}$$

In the above we assumed that any undefined $N_{j,q-1}$ with $j < p-q+2$ or $j > n$ must be regarded as the zero function, and we used the convention that if $d_{j,q-1} = 0$ then

$$\int_a^x \frac{N_{j,q-1}(y)}{d_{j,q-1}}\,\mathrm{d}y := \begin{cases} 1, & x \ge u_j \text{ and } j \le n, \\ 0, & \text{otherwise}. \end{cases} \tag{4.3}$$

The relation in (4.1) builds up the GTB-splines on each of the intervals $[x_{i-1}, x_i)$ separately; hence, it is a *local recurrence*. When all the degrees p_i are uniform, GTB-splines are usually defined by means of a global recurrence formula; see, e.g., [17]. In order to produce a *global recurrence*, we first define a global set of weight functions $\{w_j,\ j=0,\dots,p\}$ by

$$w_j(x) := \begin{cases} w_j^{(i)}(x), & j \le p_i, \\ 0, & \text{otherwise}, \end{cases} \qquad x \in [x_{i-1}, x_i), \quad i = 1,\dots,m. \tag{4.4}$$

This allows us to redefine $N_{k,q}$ globally. For $q = 0,\dots,p$ and $k = p-q+1,\dots,n$, the spline $N_{k,q}$ can be evaluated at $x \in [a,b)$ as follows:

$$N_{k,0}(x) := \begin{cases} w_p(x), & x \in [x_{i-1}, x_i), \\ 0, & \text{otherwise}, \end{cases} \tag{4.5}$$

and

$$N_{k,q}(x) := w_{p-q}(x) \int_a^x \left[\frac{N_{k,q-1}(y)}{d_{k,q-1}} - \frac{N_{k+1,q-1}(y)}{d_{k+1,q-1}} \right] \mathrm{d}y, \quad q > 0, \tag{4.6}$$

where $d_{j,q-1}$ is defined in (4.2). Again, we assumed that any undefined $N_{j,q-1}$ with $j < p-q+2$ or $j > n$ must be regarded as the zero function, and we used the convention that if $d_{j,q-1} = 0$ then (4.3) is taken. At the right end point b, the spline $N_{k,q}$ is defined by taking the limit from the left, i.e., $N_{k,q}(b) := \lim_{x\to b, x<b} N_{k,q}(x)$.

Remark 4.1. The global recurrence (4.5)–(4.6) is exactly the same as the definition known for Tchebycheffian B-splines of uniform local dimensions; see [17, Definition 7]. This was possible thanks to the enrichment of the local weights with zero functions so that we have $p+1$ functions associated with each local interval $[x_{i-1}, x_i)$, and we can glue them together into the global weights in (4.4).

Remark 4.2. The above recurrence relations are stated for the GTB-splines N_k but they do not require partition of unity. Actually, they can be used to construct locally supported functions enjoying properties (3.11), (3.12), and (3.14)–(3.16) that sum up to w_0; see [17] for Tchebycheffian B-splines of uniform local dimensions.

We note that for $m = 1$ the space $\mathbb{S}^{\boldsymbol{p}}_{\boldsymbol{r}}(\Delta)$ reduces to a single ECT-space and the GTB-spline basis always exists, provided that (3.10) holds. In analogy with the polynomial case, this basis is called the *Bernstein basis* corresponding to the considered ECT-space. We end this section by considering the special spline space $\mathbb{S}^{\boldsymbol{p}}_{\boldsymbol{r}}(\Delta) = \mathbb{S}^{\boldsymbol{p}}(\Delta)$ of discontinuous GT-splines. In this case, under assumptions (3.10), the GTB-spline basis always exists because all weight systems identifying the various ECT-spaces $\mathbb{T}^{(i)}_{p_i}$ are admissible for $\mathbb{S}^{\boldsymbol{p}}(\Delta)$. This basis is nothing else than the global Bernstein basis.

DEFINITION 4.3 (Global Bernstein basis). *Let $\{B^{(i)}_j,\ j = 0, \dots, p_i\}$ be the Bernstein basis corresponding to the ECT-space $\mathbb{T}^{(i)}_{p_i}$, $i = 1, \dots, m$. Let $l := l(i,j) := \sum_{k=1}^{i-1}(p_k+1) + j$ and define*

$$B_l(x) := \begin{cases} B^{(i)}_j(x), & x \in J_i, \\ 0, & \textit{otherwise.} \end{cases}$$

We recall from [17, Example 16] that the local Bernstein functions $B^{(i)}_j := B_{j,p_i}$, $j = 0, \dots, p_i$ are defined recursively as follows. For $q = 0, \dots, p_i$ and $j = 0, \dots, q$, the function $B_{j,q}$ is defined at $x \in [x_{i-1}, x_i]$ as

$$B_{0,0}(x) := w^{(i)}_{p_i}(x), \tag{4.7}$$

and

$$B_{j,q}(x) := w^{(i)}_{p_i-q}(x) \begin{cases} 1 - \displaystyle\int_{x_{i-1}}^{x} \frac{B_{0,q-1}(y)}{b_{0,q-1}}\mathrm{d}y, & j = 0, \\ \displaystyle\int_{x_{i-1}}^{x} \left[\frac{B_{j-1,q-1}(y)}{b_{j-1,q-1}} - \frac{B_{j,q-1}(y)}{b_{j,q-1}}\right]\mathrm{d}y, & 0 < j < q, \quad q > 0, \\ \displaystyle\int_{x_{i-1}}^{x} \frac{B_{q-1,q-1}(y)}{b_{q-1,q-1}}\mathrm{d}y, & j = q, \end{cases} \tag{4.8}$$

where

$$b_{j,q-1} := \int_{x_{i-1}}^{x_i} B_{j,q-1}(y)\,\mathrm{d}y. \tag{4.9}$$

Remark 4.4. Instead of using the recurrence relation (4.7)–(4.8), each Bernstein function $B_j^{(i)}$ can also be computed by solving the following Hermite interpolation problem in the space $\mathbb{T}_{p_i}^{(i)}$: for $j = 0$,

$$B_0^{(i)}(x_{i-1}) = 1, \quad D^l B_0^{(i)}(x_i) = 0, \quad l = 0, \dots, p_i - 1,$$

and for $j = 1, \dots, p_i$,

$$D^l B_j^{(i)}(x_{i-1}) = 0, \quad l = 0, \dots, j-1, \quad D^l B_j^{(i)}(x_i) = 0, \quad l = 0, \dots, p_i - j - 1,$$
$$D^j B_j^{(i)}(x_{i-1}) = -\sum_{l=0}^{j-1} D^j B_l^{(i)}(x_{i-1}).$$

Since $\mathbb{T}_{p_i}^{(i)}$ is an ECT-space, this interpolation problem has a unique solution; see Definition 2.1. Any convenient basis in $\mathbb{T}_{p_i}^{(i)}$ can be used to represent the Bernstein functions.

The presented integral recurrence relations, in particular the global one in (4.5)–(4.6), are suited for symbolic computation. However, they might lack stability in numerical computation. In the next section we provide a knot insertion procedure which is an important ingredient to produce an efficient and stable numerical evaluation algorithm for the basis functions we are interested in.

5. Knot insertion. Knot insertion is the fundamental operation of inserting a new knot into an existing knot vector while maintaining the shape of a spline curve. If the new knot is already present in the initial knot vector, then the continuity is reduced at the corresponding breakpoint. Otherwise, a new breakpoint is inserted in the partition.

LEMMA 5.1. *If $r_i < \min\{p_i, p_{i+1}\}$, then there are exactly $r_i + 3$ successive GTB-splines that have a jump in their (r_i+1)-th order derivative at $x = x_i$:*

$$\begin{aligned}
&\mathrm{Jump}_{x_i, r_i+1}(N_k) = 0, \quad \text{for } k = 1, \dots, \mu_{\boldsymbol{v}}(i) - 1,\\
&\mathrm{Jump}_{x_i, r_i+1}(N_k) \neq 0, \quad \text{for } k = \mu_{\boldsymbol{v}}(i), \dots, \mu_{\boldsymbol{u}}(i) + 1,\\
&\mathrm{Jump}_{x_i, r_i+1}(N_k) = 0, \quad \text{for } k = \mu_{\boldsymbol{u}}(i) + 2, \dots, n.
\end{aligned}$$

Proof. The local support (3.12) implies that all GTB-splines vanish in a neighborhood of x_i except for N_k, $k = \mu_{\boldsymbol{v}}(i-1)+1, \dots, \mu_{\boldsymbol{u}}(i+1)$. From (3.15) it follows that the GTB-splines $N_{\mu_{\boldsymbol{v}}(i)-l}(x)$, with $l > 0$, are at least C^{r_i+1}-smooth at $x = x_i$. Similarly, from (3.14) it follows that the GTB-splines $N_{\mu_{\boldsymbol{u}}(i)+1+l}(x)$, with $l > 0$, are at least C^{r_i+1}-smooth at $x = x_i$. The remaining GTB-splines $N_k(x)$, $k = \mu_{\boldsymbol{v}}(i), \dots, \mu_{\boldsymbol{u}}(i)+1$ are C^{r_i}-smooth at $x = x_i$. From [9, Theorem 4.2] and [29, Theorem 4.3] it follows that these functions have minimal support, thus they cannot be smoother at $x = x_i$. Since $\mu_{\boldsymbol{u}}(i) + 1 - (\mu_{\boldsymbol{v}}(i) - 1) = r_i + 2 - r_0 = r_i + 3$ the result follows. ∎

Suppose we remove a knot $u = v = x_i \in (a, b)$ from $\boldsymbol{u}$ and $\boldsymbol{v}$, respectively, resulting in a new spline space $\mathbb{S}_{\tilde{\boldsymbol{r}}}^{\boldsymbol{p}}(\Delta)$ with corresponding knot vectors,

$$\tilde{\boldsymbol{u}} := (\tilde{u}_k)_{k=1}^{n-1} := (\underbrace{x_0, \dots, x_0}_{p_1 - r_0 \text{ times}}, \dots, \underbrace{x_{i-1}, \dots, x_{i-1}}_{p_i - r_{i-1} \text{ times}}, \underbrace{x_i, \dots, x_i}_{p_{i+1} - r_i - 1 \text{ times}},$$
$$\underbrace{x_{i+1}, \dots, x_{i+1}}_{p_{i+2} - r_{i+1} \text{ times}}, \dots, \underbrace{x_{m-1}, \dots, x_{m-1}}_{p_m - r_{m-1} \text{ times}}),$$

and

$$\tilde{\boldsymbol{v}} := (\tilde{v}_k)_{k=1}^{n-1} := (\underbrace{x_1,\ \ldots,\ x_1}_{p_1-r_1 \text{ times}},\ \ldots,\ \underbrace{x_{i-1},\ \ldots,\ x_{i-1}}_{p_{i-1}-r_{i-1} \text{ times}},\ \underbrace{x_i,\ \ldots,\ x_i}_{p_i-r_i-1 \text{ times}},$$
$$\underbrace{x_{i+1},\ \ldots,\ x_{i+1}}_{p_{i+1}-r_{i+1} \text{ times}},\ \ldots,\ \underbrace{x_m,\ \ldots,\ x_m}_{p_m-r_m \text{ times}}).$$

The smoothness vector is easily deduced to be $\tilde{\boldsymbol{r}} = (r_0, \ldots, r_{i-1}, r_i + 1, r_{i+1}, \ldots, r_m)$; it is assumed to satisfy the same restrictions as in (2.3), and so $r_i + 1 \leq \min\{p_i, p_{i+1}\}$. Consequently, the spline space $\mathbb{S}^{\boldsymbol{p}}_{\tilde{\boldsymbol{r}}}(\Delta)$ is a subspace of $\mathbb{S}^{\boldsymbol{p}}_{\boldsymbol{r}}(\Delta)$ with one additional continuous derivative at $x = x_i$.

From Remark 3.9 we recall that each of the GTB-splines $\tilde{N}_k$, $k = 1, \ldots n-1$, that form a basis for $\mathbb{S}^{\boldsymbol{p}}_{\tilde{\boldsymbol{r}}}(\Delta)$, are uniquely defined, up to a constant multiple, by a triple $([\tilde{u}_k, \tilde{v}_k], r_{\tilde{\boldsymbol{u}}}(k), r_{\tilde{\boldsymbol{v}}}(k))$. It can directly be verified that the following relation is consistent with the definition of the knot vectors $\tilde{\boldsymbol{u}}$ and $\tilde{\boldsymbol{v}}$.

LEMMA 5.2. *The following relationship holds*

$$([\tilde{u}_k, \tilde{v}_k], r_{\tilde{\boldsymbol{u}}}(k), r_{\tilde{\boldsymbol{v}}}(k)) = \begin{cases} ([u_k, v_k], r_{\boldsymbol{u}}(k), r_{\boldsymbol{v}}(k)), & \text{if } 1 \leq k < \mu_{\boldsymbol{v}}(i), \\ ([u_k, v_{k+1}], r_{\boldsymbol{u}}(k), r_{\boldsymbol{v}}(k+1)), & \text{if } \mu_{\boldsymbol{v}}(i) \leq k \leq \mu_{\boldsymbol{u}}(i), \\ ([u_{k+1}, v_{k+1}], r_{\boldsymbol{u}}(k+1), r_{\boldsymbol{v}}(k+1)), & \text{if } \mu_{\boldsymbol{u}}(i) < k < n. \end{cases}$$

An alternative, yet, equivalent perspective is that $\boldsymbol{u}$ and $\boldsymbol{v}$ are obtained from $\tilde{\boldsymbol{u}}$ and $\tilde{\boldsymbol{v}}$ by the process of knot insertion.

PROPOSITION 5.3. *Let $\boldsymbol{u}$ and $\boldsymbol{v}$ be obtained from $\tilde{\boldsymbol{u}}$ and $\tilde{\boldsymbol{v}}$ by inserting a single knot $u = v = x_i \in (a, b)$, respectively. Then,*

$$\tilde{N}_k(x) = \alpha_k N_k(x) + \beta_{k+1} N_{k+1}(x), \tag{5.1}$$

where

(i) $\alpha_k = 1$ *and* $\beta_{k+1} = 0$ *if* $1 \leq k < \mu_{\boldsymbol{v}}(i)$;

(ii) $\alpha_k > 0$ *and* $\beta_{k+1} = -\alpha_k \dfrac{\text{Jump}_{x_i, r_i+1}(N_k)}{\text{Jump}_{x_i, r_i+1}(N_{k+1})} > 0$ *if* $\mu_{\boldsymbol{v}}(i) \leq k \leq \mu_{\boldsymbol{u}}(i)$;

(iii) $\alpha_k = 0$ *and* $\beta_{k+1} = 1$ *if* $\mu_{\boldsymbol{u}}(i) < k < n$.

Proof. Because $\mathbb{S}^{\boldsymbol{p}}_{\tilde{\boldsymbol{r}}}(\Delta) \subset \mathbb{S}^{\boldsymbol{p}}_{\boldsymbol{r}}(\Delta)$, every $\tilde{N}_k \in \mathbb{S}^{\boldsymbol{p}}_{\tilde{\boldsymbol{r}}}(\Delta)$ can be uniquely written as a linear combination of the GTB-splines that form a basis for $\mathbb{S}^{\boldsymbol{p}}_{\boldsymbol{r}}(\Delta)$. The particular functions involved in the linear combination in (5.1) follow from Lemma 5.2. Case (i) and (iii) follow directly. Case (ii) follows from Lemma 5.1 and

$$0 = \text{Jump}_{x_i, r_i+1}(\tilde{N}_k) = \alpha_k \text{Jump}_{x_i, r_i+1}(N_k) + \beta_{k+1} \text{Jump}_{x_i, r_i+1}(N_{k+1}). \tag{5.2}$$

Finally, the start-point smoothness (3.14) implies that

$$D^j_+ \tilde{N}_k(u_k) = D^j_+ N_k(u_k) = 0, \quad j = 0, \ldots, r_{\boldsymbol{u}}(k),$$

and

$$D^{r_{\boldsymbol{u}}(k)+1}_+ \tilde{N}_k(u_k) = \alpha_k D^{r_{\boldsymbol{u}}(k)+1}_+ N_k(u_k) \neq 0. \tag{5.3}$$

Then, the positivity (3.11) implies that the derivatives on both sides of (5.3) have the same sign. Consequently, α_k must be positive. A similar argument, involving the end-point smoothness in (3.15), shows that β_{k+1} is positive. ∎

We can also write (5.1) in matrix notation,

$$\tilde{N}_k(x) = \sum_{l=1}^{n} \tilde{\mathsf{C}}_{kl} N_l(x), \quad k = 1, \dots, n-1 \quad \Longleftrightarrow \quad \tilde{\mathbf{N}}(x) = \tilde{\mathsf{C}}\, \mathbf{N}(x). \tag{5.4}$$

Here, $\tilde{\mathsf{C}} \in \mathbb{R}^{(n-1)\times n}$ and the entries $\tilde{\mathsf{C}}_{kl}$ are determined by the three cases (i)–(iii). The matrix $\tilde{\mathsf{C}}$ has the following sparsity structure:

$$\tilde{\mathsf{C}} = \begin{bmatrix} \mathsf{I}_A & & \\ & \hat{\mathsf{C}} & \\ & & \mathsf{I}_B \end{bmatrix}, \quad \hat{\mathsf{C}} = \begin{bmatrix} \alpha_{\mu_{\boldsymbol{v}}(i)} & \beta_{\mu_{\boldsymbol{v}}(i)+1} & & & \\ & \alpha_{\mu_{\boldsymbol{v}}(i)+1} & \ddots & & \\ & & \ddots & \beta_{\mu_{\boldsymbol{u}}(i)} & \\ & & & \alpha_{\mu_{\boldsymbol{u}}(i)} & \beta_{\mu_{\boldsymbol{u}}(i)+1} \end{bmatrix}. \tag{5.5}$$

Here, I_A and I_B are identity matrices of size $(\mu_{\boldsymbol{v}}(i)-1)$ and $(n-1-\mu_{\boldsymbol{u}}(i))$, respectively, and $\hat{\mathsf{C}} \in \mathbb{R}^{(r_i+2)\times(r_i+3)}$.

THEOREM 5.4 (Knot insertion). *Let $\boldsymbol{u}$ and $\boldsymbol{v}$ be obtained from $\tilde{\boldsymbol{u}}$ and $\tilde{\boldsymbol{v}}$ by inserting a single knot $u = v = x_i \in (a, b)$, respectively. Then,*

$$s(x) = \sum_{k=1}^{n-1} \tilde{d}_k \tilde{N}_k(x) = \sum_{k=1}^{n} d_k N_k(x), \tag{5.6}$$

where

$$d_k = \begin{cases} \tilde{d}_k, & 1 \le k \le \mu_{\boldsymbol{v}}(i), \\ \beta_k \tilde{d}_{k-1} + \alpha_k \tilde{d}_k, & \mu_{\boldsymbol{v}}(i) < k \le \mu_{\boldsymbol{u}}(i), \\ \tilde{d}_{k-1}, & \mu_{\boldsymbol{u}}(i) < k \le n, \end{cases} \tag{5.7}$$

with $\alpha_k + \beta_k = 1$.

Proof. We write (5.6) as,

$$\tilde{\mathbf{d}}^T \tilde{\mathbf{N}}(x) = \tilde{\mathbf{d}}^T \big(\tilde{\mathsf{C}}\, \mathbf{N}(x)\big) = \big(\tilde{\mathbf{d}}^T \tilde{\mathsf{C}}\big) \mathbf{N}(x) = \mathbf{d}^T \mathbf{N}(x).$$

It follows that

$$\mathbf{d}^T = \tilde{\mathbf{d}}^T \tilde{\mathsf{C}} \quad \Longleftrightarrow \quad d_l = \sum_{k=1}^{n-1} \tilde{d}_k \tilde{\mathsf{C}}_{kl}, \quad l = 1, \dots, n.$$

The structure of $\tilde{\mathsf{C}}$, given in (5.5), leads to the result in (5.7). Finally, the partition of unity (3.13) of both bases $\{N_k,\ k = 1, \dots, n\}$ and $\{\tilde{N}_k,\ k = 1, \dots, n-1\}$ implies that the column sum of matrix $\tilde{\mathsf{C}}$ is one. Hence, $\alpha_{\mu_{\boldsymbol{v}}(i)} = \beta_{\mu_{\boldsymbol{u}}(i)+1} = 1$ and $\alpha_k + \beta_k = 1$ for $k = \mu_{\boldsymbol{v}}(i) + 1, \dots, \mu_{\boldsymbol{u}}(i)$. ∎

Besides its intrinsic interest, the knot insertion procedure can be applied recursively in order to compute a Bézier extraction operator which allows for efficient and stable evaluation of GTB-splines. This will be shown in the next section.

6. Algorithmic evaluation. In this section we present an algorithm that computes the GTB-spline basis $\{N_k,\ k = 1, \dots, n\}$, whenever it exists, for the spline space $\mathbb{S}^{\boldsymbol{p}}_{\boldsymbol{r}}(\Delta)$ using Bézier extraction, i.e., representing each basis element in the form

$$N_k(x) = \sum_{l=1}^{\theta} \mathsf{C}_{kl} B_l(x), \quad k = 1, \dots, n \quad \Longleftrightarrow \quad \mathbf{N}(x) = \mathsf{C}\, \mathbf{B}(x). \tag{6.1}$$

Here, $\{B_l,\ l=1,\dots,\theta := \sum_{i=1}^{m}(p_i+1)\}$ denotes the global Bernstein basis for $\mathbb{S}^{\boldsymbol{p}}(\Delta)$, see Definition 4.3, and $\mathsf{C} \in \mathbb{R}^{n\times\theta}$ is the extraction operator that maps functions from $\mathbb{S}^{\boldsymbol{p}}(\Delta)$ to $\mathbb{S}^{\boldsymbol{p}}_{\boldsymbol{r}}(\Delta)$.

By construction, $\{B_l,\ l=1,\dots,\theta\}$ forms a global, locally supported basis for the space $\mathbb{S}^{\boldsymbol{p}}(\Delta)$ that has the properties listed in Theorem 3.7. Since $\mathbb{S}^{\boldsymbol{p}}_{\boldsymbol{r}}(\Delta)$ is a subspace of $\mathbb{S}^{\boldsymbol{p}}(\Delta)$, we can use the knot insertion procedure to convert from the global Bernstein basis $\{B_l,\ l=1,\dots,\theta\}$ to the smooth GTB-spline basis $\{N_k,\ k=1,\dots,n\}$.

As already observed in [38] for the polynomial setting, the knot insertion procedure in (5.1) can be regarded as a nullspace computation of the smoothness constraints (5.2) for all k at the breakpoint x_i. Let $\mathbf{a} \in \mathbb{R}^n$ denote the vector with entries

(6.2)
$$\mathbf{a} := \begin{bmatrix} 0 & \cdots & 0 & \mathrm{Jump}_{x_i,r_i+1}(N_{\mu_{\boldsymbol{v}}(i)}) & \cdots & \mathrm{Jump}_{x_i,r_i+1}(N_{\mu_{\boldsymbol{u}}(i)+1}) & 0 & \cdots & 0 \end{bmatrix}^T.$$

Then, the knot insertion matrix $\tilde{\mathsf{C}} \in \mathbb{R}^{(n-1)\times n}$ in (5.4) and (5.5) satisfies $\tilde{\mathsf{C}}\mathbf{a} = \mathbf{0}$. This matrix can be computed by Algorithm 6.1 using the vector $\mathbf{a}$ as input.

PROPOSITION 6.1. *Given the vector* $\mathbf{a}$ *in* (6.2) *as input, Algorithm* 6.1 *computes the matrix* $\tilde{\mathsf{C}}$ *in* (5.5).

Proof. The partition of unity (3.13) implies that the column sum of $\tilde{\mathsf{C}}$ is equal to one. In combination with (5.2), it can be observed that the non-trivial entries of $\tilde{\mathsf{C}}$ can be computed in succession as follows:

$$\begin{aligned}
\alpha_{\mu_{\boldsymbol{v}}(i)} &= 1 \\
&\downarrow \\
\beta_{\mu_{\boldsymbol{v}}(i)+1} &= -\alpha_{\mu_{\boldsymbol{v}}(i)} \cdot \mathbf{a}_{\mu_{\boldsymbol{v}}(i)} / \mathbf{a}_{\mu_{\boldsymbol{v}}(i)+1} \\
&\downarrow \\
\alpha_{\mu_{\boldsymbol{v}}(i)+1} &= 1 - \beta_{\mu_{\boldsymbol{v}}(i)+1} \\
&\downarrow \\
\beta_{\mu_{\boldsymbol{v}}(i)+2} &= -\alpha_{\mu_{\boldsymbol{v}}(i)+1} \cdot \mathbf{a}_{\mu_{\boldsymbol{v}}(i)+1} / \mathbf{a}_{\mu_{\boldsymbol{v}}(i)+2} \\
&\vdots \\
\beta_{\mu_{\boldsymbol{u}}(i)+1} &= 1.
\end{aligned}$$

It can be directly verified that $\tilde{\mathsf{C}}\mathbf{a} = \mathbf{0}$. This logic is encoded in Algorithm 6.1. ∎

By applying Algorithm 6.1 repeatedly the global Bernstein basis $\{B_l,\ l=1,\dots,\theta\}$ can be mapped to the smooth GTB-spline basis $\{N_k,\ k=1,\dots,n\}$. This procedure is called *Bézier extraction* following terminology introduced in the polynomial spline context [5, 32]; we follow suit.

THEOREM 6.2 (Bézier extraction). *Let* $\rho := \rho(i,j) := \sum_{k=1}^{i-1}(r_k+1)+j+1$, *and consider the following linear indexing of the smoothness constraints*

$$A_\rho(\cdot) = \mathrm{Jump}_{x_i,j}(\cdot), \quad j=0,\dots,r_i, \quad i=1,\dots,m-1.$$

Let the input into Algorithm 6.2 *be given by the matrix* A *with matrix columns*

$$A_\rho(\mathbf{B}), \quad \rho=1,\dots,\phi,$$

where the vector $\mathbf{B}$ *collects the global Bernstein functions* $\{B_l,\ l=1,\dots,\theta\}$; *see* (6.1). *Then, Algorithm* 6.2 *produces a Bézier extraction operator* $\mathsf{C} \in \mathbb{R}^{n\times\theta}$ *that reproduces*

Algorithm 6.1 Nullspace computation of a smoothness constraint based on knot insertion

function NULLSPACE($\mathbf{a} \in \mathbb{R}^n$)
  $\tilde{\mathsf{C}} \leftarrow$ zero matrix (size: $(n-1) \times n$)
  $k \leftarrow 1$
  while $k < n$ & $\mathbf{a}(k) = 0$ **do**
    $\tilde{\mathsf{C}}(k,k) \leftarrow 1$
    $k \leftarrow k+1$
  end while
  $\tilde{\mathsf{C}}(k,k) \leftarrow 1$
  while $k+1 < n$ & $\mathbf{a}(k+1) \neq 0$ **do**
    $\tilde{\mathsf{C}}(k,k+1) \leftarrow -\mathbf{a}(k) \,/\, \mathbf{a}(k+1)\, \tilde{\mathsf{C}}(k,k)$
    $\tilde{\mathsf{C}}(k+1,k+1) \leftarrow 1 - \tilde{\mathsf{C}}(k,k+1)$
    $k \leftarrow k+1$
  end while
  while $k < n$ **do**
    $\tilde{\mathsf{C}}(k,k+1) \leftarrow 1$
    $k \leftarrow k+1$
  end while
  return $\tilde{\mathsf{C}}$
end function

Algorithm 6.2 Generalized Bézier extraction

function EXTRACTION_OPERATOR($\mathsf{A} \in \mathbb{R}^{\theta \times \phi}$)
  $\mathsf{C} \leftarrow$ identity matrix (size: $\theta \times \theta$) ▷ Initialize extraction operator
  for $\rho = 1 : \phi$ **do** ▷ Loop over smoothness constraints
    $\tilde{\mathsf{C}} \leftarrow$ NULLSPACE($\mathsf{A}(:,\rho)$) ▷ Compute nullspace of ρ-th column of A
    $\mathsf{C} \leftarrow \tilde{\mathsf{C}} * \mathsf{C}$ ▷ update C
    $\mathsf{A} \leftarrow \tilde{\mathsf{C}} * \mathsf{A}$ ▷ update A
  end for
  return C
end function

a GTB-spline basis $\{N_k,\ k = 1, \dots, n\}$ *for the spline space* $\mathbb{S}^{\boldsymbol{p}}_{\boldsymbol{r}}(\Delta)$ *according to*

$$\mathbf{N}(x) = \mathsf{C}\, \mathbf{B}(x).$$

Proof. Let $\mathbb{S}^{\boldsymbol{p}}_{(\rho)}$ denote the GT-spline space that satisfies the first ρ linear smoothness constraints $A_k(\cdot)$, $k = 1, \dots, \rho$, in Definition 2.5, and let $\mathbf{N}_{(\rho)}(x)$ denote its corresponding GTB-spline basis. Note that $\mathbb{S}^{\boldsymbol{p}}_{(0)} \equiv \mathbb{S}^{\boldsymbol{p}}(\Delta)$, $\mathbb{S}^{\boldsymbol{p}}_{(\phi)} \equiv \mathbb{S}^{\boldsymbol{p}}_{\boldsymbol{r}}(\Delta)$ and $\mathbb{S}^{\boldsymbol{p}}_{(\rho)} \subset \mathbb{S}^{\boldsymbol{p}}_{(\rho-1)}$.

The space $\mathbb{S}^{\boldsymbol{p}}_{(\rho-1)}$ can be obtained from $\mathbb{S}^{\boldsymbol{p}}_{(\rho)}$ by inserting a single knot into its corresponding knot vectors. Consequently, there exists $\tilde{\mathsf{C}}_\rho \in \mathbb{R}^{(\theta-\rho)\times(\theta-\rho+1)}$ with its structure given by (5.5) such that $\mathbf{N}_{(\rho)}(x) = \tilde{\mathsf{C}}_\rho\, \mathbf{N}_{(\rho-1)}(x)$. Repeating this argument we observe that

$$\mathsf{C} = \tilde{\mathsf{C}}_\phi \cdots \tilde{\mathsf{C}}_2 \tilde{\mathsf{C}}_1.$$

Algorithm 6.2 implements this recursion in line 5. It remains to show that at each step, ρ, the input into Algorithm 6.1 is such that it reproduces operator $\tilde{\mathsf{C}}_\rho$. The correct input is given by the vector, $\mathbf{a} = A_\rho(\mathbf{N}_{(\rho-1)})$, such that

$$\tilde{\mathsf{C}}_\rho\, \mathbf{a} = \tilde{\mathsf{C}}_\rho\, A_\rho(\mathbf{N}_{(\rho-1)}) = A_\rho(\tilde{\mathsf{C}}_\rho\, \mathbf{N}_{(\rho-1)}) = A_\rho(\mathbf{N}_{(\rho)}) = \mathbf{0}.$$

We have that

(i) at step 1 the input into Algorithm 6.1 is

$$A_1(\mathbf{B}) = A_1(\mathbf{N}_{(0)});$$

(ii) the update in line 6 shows that at step ρ the input into Algorithm 6.1 is

$$\tilde{\mathsf{C}}_{\rho-1} \cdots \tilde{\mathsf{C}}_1 A_\rho(\mathbf{B}) = \tilde{\mathsf{C}}_{\rho-1} \cdots \tilde{\mathsf{C}}_2 A_\rho(\mathbf{N}_{(1)}) = \tilde{\mathsf{C}}_{\rho-1} A_\rho(\mathbf{N}_{(\rho-2)}) = A_\rho(\mathbf{N}_{(\rho-1)}).$$

Hence, by induction, the input into Algorithm 6.1 is correct at every step of the recursion. Consequently, Algorithm 6.2 produces the expected output. ∎

Remark 6.3. Because the coefficients α_k and β_k are positive and sum to 1, Algorithm 6.1 is numerically stable. Hence, the computation of the matrix $\tilde{\mathsf{C}}$ will be accurate as long as the vector $\mathbf{a}$ is known to sufficient precision. In practice this means that we require accurate and stable evaluation of Bernstein functions and their higher-order derivatives at the breakpoints.

Remark 6.4. For polynomial B-splines of non-uniform degree, so-called *multi-degree B-splines*, Bézier extraction has been analyzed and successfully applied in [37, 38]. An efficient MATLAB toolbox implementation illustrating Algorithms 6.1 and 6.2 can be found in [36].

Remark 6.5. Whenever the space $\mathbb{S}^{\boldsymbol{p}}_{\boldsymbol{r}}(\Delta)$ admits a basis $\{N_k,\ k = 1, \ldots, n\}$ with the properties listed in Theorems 3.7 and 5.4, Algorithm 6.2 (using Algorithm 6.1) can be applied for an efficient evaluation of these basis functions. In other words, the algorithm does not require that the space $\mathbb{S}^{\boldsymbol{p}}_{\boldsymbol{r}}(\Delta)$ is identified by an admissible weight system (see Definition 2.7).

7. Case study: generalized polynomial B-splines. In this section we consider a special class of GTB-splines, the so-called *generalized polynomial B-splines (GPB-splines)*; see [23] and references therein. They can be seen as the minimal extension of polynomial splines of non-uniform degree still offering a wide variety of additional flexibility.

Given $p_i \geq 2$, let $\mathfrak{u}^{(i)}, \mathfrak{v}^{(i)} \in C^{p_i}([x_{i-1}, x_i])$, and

$$U^{(i)} := D^{p_i-1}\mathfrak{u}^{(i)}, \quad V^{(i)} := D^{p_i-1}\mathfrak{v}^{(i)},$$

such that $\mathbb{G}^{(i)} := \operatorname{span}\{U^{(i)}, V^{(i)}\}$ is an ECT-space on $[x_{i-1}, x_i]$. There exists a unique couple of functions $\widetilde{U}^{(i)}, \widetilde{V}^{(i)} \in \mathbb{G}^{(i)}$ such that

$$\widetilde{U}^{(i)}(x_{i-1}) = 1, \quad \widetilde{U}^{(i)}(x_i) = 0, \qquad \widetilde{V}^{(i)}(x_{i-1}) = 0, \quad \widetilde{V}^{(i)}(x_i) = 1.$$

The generalized polynomial space of degree $p_i \geq 2$ on the closed interval $[x_{i-1}, x_i]$ is then defined by

$$\mathbb{G}^{(i)}_{p_i} := \operatorname{span}\left\{1, x, x^2, \ldots, x^{p_i-2}, \mathfrak{u}^{(i)}(x), \mathfrak{v}^{(i)}(x)\right\}. \tag{7.1}$$

We refer to [11] for more details on such spaces and their properties.

Example 7.1. Popular choices for $\mathfrak{u}^{(i)}$ and $\mathfrak{v}^{(i)}$ are given by
- $\mathfrak{u}^{(i)}(x) = x^{p_i-1}, \quad \mathfrak{v}^{(i)}(x) = x^{p_i}$,
- $\mathfrak{u}^{(i)}(x) = \sinh(\omega x), \quad \mathfrak{v}^{(i)}(x) = \cosh(\omega x), \quad 0 < \omega$,
- $\mathfrak{u}^{(i)}(x) = \sin(\omega x), \quad \mathfrak{v}^{(i)}(x) = \cos(\omega x), \quad 0 < \omega(x_i - x_{i-1}) < \pi$,

which correspond to the classical polynomial, exponential and trigonometric spaces, respectively.

Remark 7.2. Once $\mathfrak{u}^{(i)}$ and $\mathfrak{v}^{(i)}$ are chosen they uniquely define $U^{(i)}$ and $V^{(i)}$. Conversely, if $U^{(i)}$ and $V^{(i)}$ are chosen they uniquely define the space $\mathbb{G}^{(i)}_{p_i}$.

Remark 7.3. From [17, Example 10] we know that the space $\mathbb{G}^{(i)}_{p_i}$ is an ECT-space generated by the weights

$$\begin{aligned} w^{(i)}_0(x) &= \cdots = w^{(i)}_{p_i-2}(x) = 1, \\ w^{(i)}_{p_i-1}(x) &= \widetilde{U}^{(i)}(x) + \widetilde{V}^{(i)}(x), \\ w^{(i)}_{p_i}(x) &= \frac{\widetilde{U}^{(i)}(x) D\widetilde{V}^{(i)}(x) - \widetilde{V}^{(i)}(x) D\widetilde{U}^{(i)}(x)}{\left(\widetilde{U}^{(i)}(x) + \widetilde{V}^{(i)}(x)\right)^2}. \end{aligned}$$

It can be easily checked that these local weights are admissible for the spline space $\mathbb{S}^{\boldsymbol{p}}_{\boldsymbol{r}}(\Delta)$, where the different pieces belong to $\mathbb{G}^{(i)}_{p_i}$, $i = 1, \dots, m$ (see Definition 2.7), whenever $r_i < \min\{p_i, p_{i+1}\}$. Moreover, they fulfill the partition-of-unity assumption in (3.10).

From Remark 7.3 and Theorem 3.7 it follows that there exist GTB-splines for spline spaces composed of generalized polynomial spaces as in (7.1) of possibly different dimensions. These GTB-splines can be computed by the algorithmic procedure described in section 6 starting from the local Bernstein bases. In the remainder of the section we discuss and illustrate the Bernstein basis in case of generalized polynomial spaces.

Using the explicit expressions of the weights provided in Remark 7.3, we can simplify the recurrence relation in (4.7)–(4.8) of the local Bernstein functions $B^{(i)}_j := B_{j,p_i}$, $j = 0, \dots, p_i$ as follows; see also [17, Section 4]. For $q = 1, \dots, p_i$ and $j = 0, \dots, q$, the function $B_{j,q}$ can be evaluated at $x \in [x_{i-1}, x_i]$ as

$$B_{0,1}(x) := \widetilde{U}^{(i)}(x), \quad B_{1,1}(x) := \widetilde{V}^{(i)}(x), \tag{7.2}$$

and

$$B_{j,q}(x) := \begin{cases} 1 - \displaystyle\int_{x_{i-1}}^{x} \frac{B_{0,q-1}(y)}{b_{0,q-1}} \mathrm{d}y, & j = 0, \\ \displaystyle\int_{x_{i-1}}^{x} \left[\frac{B_{j-1,q-1}(y)}{b_{j-1,q-1}} - \frac{B_{j,q-1}(y)}{b_{j,q-1}} \right] \mathrm{d}y, & 0 < j < q, \quad q > 1, \\ \displaystyle\int_{x_{i-1}}^{x} \frac{B_{q-1,q-1}(y)}{b_{q-1,q-1}} \mathrm{d}y, & j = q, \end{cases} \tag{7.3}$$

where $b_{j,q-1}$ is defined in (4.9).

Example 7.4. The classical (polynomial) Bernstein basis of degree $p_i = q$ on $[x_{i-1}, x_i] = [0, 1]$ can be expressed as

$$B_{j,q}(x) = \binom{q}{j} x^j (1-x)^{q-j}, \quad j = 0, \dots, q.$$

Example 7.5. The generalized Bernstein basis for the exponential space in Example 7.1 defined on $[x_{i-1}, x_i] = [0, 1]$ reads for degree $q = 1$,

$$B_{0,1}(x) = \frac{\sinh(\omega(1-x))}{\sinh(\omega)}, \quad B_{1,1}(x) = \frac{\sinh(\omega x)}{\sinh(\omega)},$$

and for degree $q = 2$,

$$B_{0,2}(x) = \frac{1-\cosh(\omega(1-x))}{1-\cosh(\omega)}, \quad B_{2,2}(x) = \frac{1-\cosh(\omega x)}{1-\cosh(\omega)},$$
$$B_{1,2}(x) = \frac{\cosh(\omega(1-x)) + \cosh(\omega x) - \cosh(\omega) - 1}{1-\cosh(\omega)}.$$

Example 7.6. The Bernstein basis for the trigonometric space in Example 7.1 defined on $[x_{i-1}, x_i] = [0, 1]$ reads for degree $q = 1$,

$$B_{0,1}(x) = \frac{\sin(\omega(1-x))}{\sin(\omega)}, \quad B_{1,1}(x) = \frac{\sin(\omega x)}{\sin(\omega)},$$

and for degree $q = 2$,

$$B_{0,2}(x) = \frac{1-\cos(\omega(1-x))}{1-\cos(\omega)}, \quad B_{2,2}(x) = \frac{1-\cos(\omega x)}{1-\cos(\omega)},$$
$$B_{1,2}(x) = \frac{\cos(\omega(1-x)) + \cos(\omega x) - \cos(\omega) - 1}{1-\cos(\omega)}.$$

Remark 7.7. Instead of using the recurrence relation (7.2)–(7.3), each Bernstein function $B_j^{(i)}$ can also be computed by solving in the space $\mathbb{G}_{p_i}^{(i)}$ the Hermite interpolation problem stated in Remark 4.4.

GT-spline spaces with pieces drawn from (different) generalized polynomial spaces containing polynomial, exponential or trigonometric functions (see, e.g., Examples 7.4 and 7.6) are of particular interest both in geometric design and numerical simulation because they offer a valid alternative to NURBS. Indeed, they allow for a locally exact representation of conic sections with respect to (almost) arc length, and moreover, the derivative spaces belong to the same class, exactly as for polynomial splines; see [23] and references therein for further details.

8. Numerical examples. In this section we present two numerical examples to illustrate the algorithmic procedure in section 6 and a simple application of GTB-splines for exact smooth representation of profiles containing conic section segments.

Example 8.1. Consider a GT-spline space $\mathbb{S}_{\boldsymbol{r}}^{\boldsymbol{p}}(\Delta)$ defined by

$$\Delta = \{0, 1, 5/2, 5\}, \quad \boldsymbol{p} = \{2, 3, 4\}, \quad \boldsymbol{r} = \{-1, 2, 2, -1\},$$

and

$$\mathbb{T}_2^{(1)} = \operatorname{span}\left\{1, x, x^2\right\}, \quad \mathbb{T}_3^{(2)} = \operatorname{span}\left\{1, x, \cos(\pi x/2), \sin(\pi x/2)\right\},$$
$$\mathbb{T}_4^{(3)} = \operatorname{span}\left\{1, x, x^2, \sinh(10x), \cosh(10x)\right\}.$$

All these spaces are special instances of generalized polynomial spaces discussed in section 7. Sequences of admissible weights for $\mathbb{S}_{\boldsymbol{r}}^{\boldsymbol{p}}(\Delta)$ can be computed as described in

TABLE 8.1
The triples $([u_k, v_k], r_{\boldsymbol{u}}(k), r_{\boldsymbol{v}}(k))$ *defining* $N_k \in \mathbb{S}^{\boldsymbol{p}}_{\boldsymbol{r}}(\Delta)$ *for* $k = 1, \ldots, n$ *as in Example 8.1.*

k	1	2	3	4	5	6
u_k	0	0	0	1	5/2	5/2
v_k	5/2	5	5	5	5	5
$r_{\boldsymbol{u}}(k)$	-1	0	1	2	2	3
$r_{\boldsymbol{v}}(k)$	2	3	2	1	0	-1

Remark 7.3 by taking into account the explicit expressions for $\widetilde{U}^{(i)}$ and $\widetilde{V}^{(i)}$, $i = 1, 2, 3$:

$$\widetilde{U}^{(1)}(x) = 1 - x, \quad \widetilde{U}^{(2)} = -\sqrt{2}\cos(\pi/4 + \pi x/2), \quad \widetilde{U}^{(3)} = \frac{\sinh(50 - 10x)}{\sinh(25)},$$
$$\widetilde{V}^{(1)}(x) = x, \qquad \widetilde{V}^{(2)} = -\sqrt{2}\cos(\pi x/2), \qquad \widetilde{V}^{(3)} = -\frac{\sinh(25 - 10x)}{\sinh(25)}.$$

The resulting GT-spline space $\mathbb{S}^{\boldsymbol{p}}_{\boldsymbol{r}}(\Delta)$ has dimension 6 and, in view of Theorem 3.7, possesses a GTB-spline basis. For this space, the knot vectors, $\boldsymbol{u}$ and $\boldsymbol{v}$, and the start-point and end-point smoothness, $r_{\boldsymbol{u}}$ and $r_{\boldsymbol{v}}$, are depicted in Table 8.1. Figure 8.1 shows the spline functions constructed at different iterations in Algorithm 6.2, together with their first and second derivatives. Starting from the 12 global Bernstein basis functions of $\mathbb{S}^{\boldsymbol{p}}(\Delta)$ (Figure 8.1, first row), Algorithm 6.2 incrementally increases the smoothness at the breakpoints until the 6 final GTB-splines (Figure 8.1, last row) are obtained.

Example 8.2. The profile depicted in Figure 8.2 (left) consists of one circular arc, with center $(2, 0)$ and radius 1, connected by a straight line segment to another circular arc, with center $(0, 3)$ and radius 2. More precisely, we are considering the profile described by the parametric curve

$$(X(x),\ Y(x)) = \begin{cases} (2 - \sin(x),\ \cos(x)), & x \in [-3\pi/4, 0), \\ (2 - x,\ 1), & x \in [0, 2), \\ (-2\sin(x/2 - 1),\ 3 - 2\cos(x/2 - 1)), & x \in [2, 2 + \pi]. \end{cases}$$

One can easily verify that this parameterization is C^1 in both components. This profile can be exactly represented as a parametric C^1 GT-spline curve whose components belong to the 4-dimensional GT-spline space $\mathbb{S}^{\boldsymbol{p}}_{\boldsymbol{r}}(\Delta)$ defined by

$$\Delta = \{-3\pi/4, 0, 2, 2 + \pi\}, \quad \boldsymbol{p} = \{2, 1, 2\}, \quad \boldsymbol{r} = \{-1, 1, 1, -1\},$$

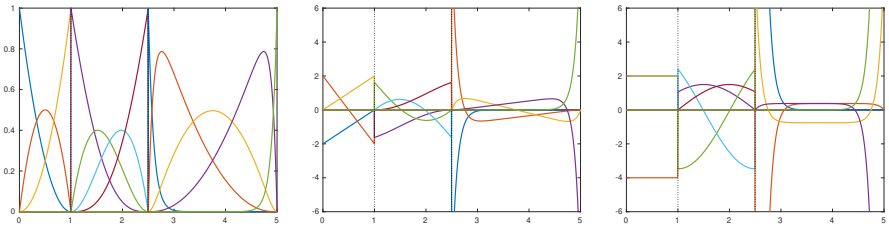
and

$$\mathbb{T}^{(1)}_2 = \operatorname{span}\{1, \cos(x), \sin(x)\}, \quad \mathbb{T}^{(2)}_1 = \operatorname{span}\{1, x\},$$
$$\mathbb{T}^{(3)}_2 = \operatorname{span}\{1, \cos(x/2), \sin(x/2)\}.$$

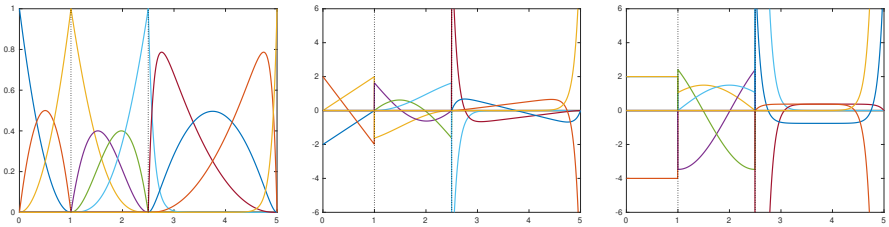
The parametric coefficients (control points) are given by

$$(2 + \sqrt{2}/2,\ -\sqrt{2}/2), \quad (3 + \sqrt{2},\ 1), \quad (-2,\ 1), \quad (-2,\ 3).$$

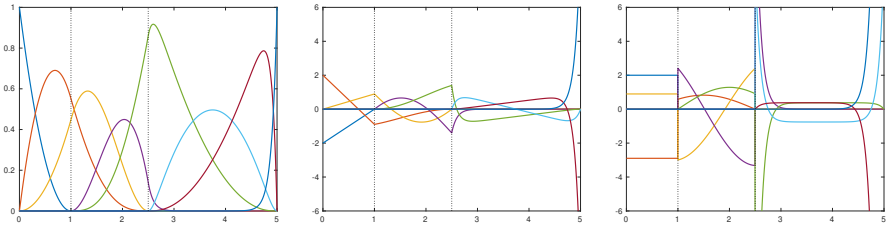
The representation in terms of GTB-splines and the corresponding control polygon is visualized in Figure 8.2 (left). We see that the control polygon nicely encapsulates the profile. The C^1 basis functions used in the representation are shown in Figure 8.2 (middle), and their first derivatives in Figure 8.2 (right).



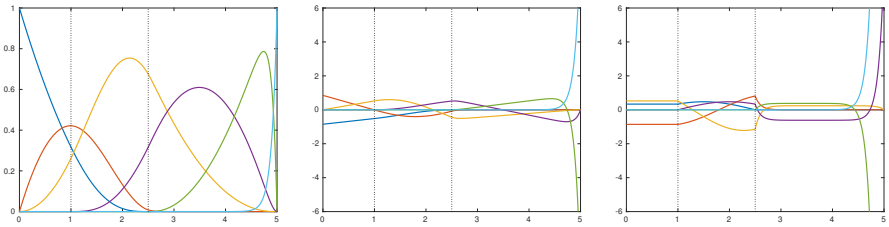
(a) $\boldsymbol{r} = \{-1, -1, -1, -1\}$



(b) $\boldsymbol{r} = \{-1, 0, 0, -1\}$



(c) $\boldsymbol{r} = \{-1, 1, 1, -1\}$



(d) $\boldsymbol{r} = \{-1, 2, 2, -1\}$

FIG. 8.1. *Sets of GTB-splines for the GT-spline spaces* $\mathbb{S}^{\boldsymbol{p}}_{\boldsymbol{r}}(\Delta)$ *built from the ECT-spaces* $\mathbb{T}^{(i)}_{p_i}$, $i = 1, 2, 3$ *defined in Example 8.1 and different smoothness classes* $\boldsymbol{r}$ *(left column), together with their first derivatives (middle column) and second derivatives (right column). Knot positions are visualized by vertical dotted lines.*

Remark 8.3. The nice behavior of the control polygon in Example 8.2 (and corresponding Figure 8.2) is not a coincidence. The control polygon is easy to construct for any C^1 GTB-spline curve whose components belong to a GT-spline space consisting of ECT-spaces of dimension 3 and 2 in an alternating sequence. The control points are given by: (a) the two end points of the curve, and (b) the ordered intersections of the two end point tangent lines with the tangent lines corresponding to the linear segments.

FIG. 8.2. *A profile consisting of two circular arcs connected by a straight line defined in Example 8.2 (left column), together with the C^1 GTB-splines used in the representation (middle column) and their first derivatives (right column). Knot positions are visualized by vertical dotted lines.*

9. Conclusion. GT-spline spaces are smooth function spaces where the pieces are drawn from different ET-spaces of possibly different dimensions. Under quite mild assumptions, they offer the possibility of exploiting the wide flexibility of ET-spaces while retaining the nice properties of classical polynomial spline spaces, including a B-spline-like basis, the so-called GTB-splines.

Besides their common application in constrained interpolation/approximation and computer-aided geometric design — the richness of ET-spaces offers a huge universe of shapes for modeling — GTB-splines can be a powerful tool for numerical simulation as well. Since a relevant class of ET-spaces can be specified as the nullspaces of linear differential operators, it is clear that GT-splines and GTB-splines can provide an appealing problem-dependent alternative to classical polynomial splines for the numerical treatment of differential and integral problems. Heretofore, this great potential has been thwarted by the lack of efficient and reliable evaluation procedures for GTB-splines, even in the simpler case where ET-spaces of the same dimension are glued together.

We have presented an efficient and robust algorithm for evaluation of GTB-splines whenever they exist. The algorithm proceeds by incrementally increasing the smoothness starting from the space of piecewise discontinuous functions obtained by collecting the various ET-spaces. It requires as input the local Bernstein-like bases and produces as output the entire set of GTB-splines that span the considered GT-spline space. The algorithm recursively constructs the nullspace of a suitable matrix in a numerically stable way without solving a linear system. In contrast with the current available methods for evaluation of GTB-splines, the proposed strategy does not require any (numerical) integration. Indeed, integration can be avoided also to produce the starting Bernstein-like bases as they can be obtained by solving suitable local Hermite interpolation problems and this can be done in a pre-processing step. The provided algorithm is a Tchebycheffian extension of the procedure recently developed and analyzed in [36, 37, 38] for multi-degree polynomial splines.

The considered ET-spaces, defined on the bounded and closed intervals identified by the breakpoints, are represented in terms of weights, possibly constrained by some admissibility conditions. It should be noted, however, that this is merely for the sake of presentation, so as to have a framework where GTB-splines exist. The proposed algorithm does not require any weights and always produces the GTB-spline basis whenever it exists. Therefore, the end-user can completely ignore this representation in terms of weights when just interested in the computation of GTB-splines.

The majority of works on Tchebycheffian splines deals with spline spaces obtained by gluing together ET-spaces of the same dimension. In this context, mainly motivated by computer-aided geometric design as application, the concept of geometric

continuity is often considered instead of classical continuity. Geometric continuity offers additional shape parameters for design. However, this flexibility comes at a price of increased complexity and can be of practical interest only when equipped with proper, preferably automatic, strategy for parameter selection. In this paper, we have deliberately confined ourselves to classical continuity with the aim of promoting the use of GT-splines in the wider context of numerical simulation and more precisely in isogeometric methods, where the choice of the ET-spaces has to be driven by the character of the problem under consideration. Nevertheless, the presented procedure has the potential to construct an efficient evaluation algorithm for geometrically continuous Tchebycheffian splines as well. Future research efforts will also focus on multivariate extensions of the algorithmic evaluation approach; a particularly interesting topic in this direction is the construction of a B-spline-like basis for GT-splines on T-meshes.

Acknowledgments. R. R. Hiemstra, T. J. R. Hughes, and D. Toshniwal were partially supported by the Office of Naval Research (N00014-17-1-2119, N00014-17-1-2039, and N00014-13-1-0500), by the Army Research Office (W911NF-13-1-0220), by the National Institutes of Health (5R01HL129077-02,03), and by the National Science Foundation Industry/University Cooperative Research Center (IUCRC) for Efficient Vehicles and Sustainable Transportation Systems (EV-STS), and the United States Army CCDC Ground Vehicle Systems Center (TARDEC/NSF Project 1650483 AMD 2). C. Manni and H. Speleers were partially supported by the Mission Sustainability Programme of the University of Rome Tor Vergata through the project IDEAS (CUP E81I18000060005) and by the MIUR Excellence Department Project awarded to the Department of Mathematics, University of Rome Tor Vergata (CUP E83C18000100006); they are members of Gruppo Nazionale per il Calcolo Scientifico – Istituto Nazionale di Alta Matematica. This support is gratefully acknowledged.

REFERENCES

[1] A. Aimi, M. Diligenti, M. L. Sampoli, and A. Sestini, *Non-polynomial spline alternatives in isogeometric symmetric Galerkin BEM*, Appl. Numer. Math., 116 (2017), pp. 10–23.
[2] C. V. Beccari, G. Casciola, and S. Morigi, *On multi-degree splines*, Comput. Aided Geom. Design, 58 (2017), pp. 8–23.
[3] D. Bister and H. Prautzsch, *A new approach to Tchebycheffian B-splines*, in Curves and Surfaces with Applications in CAGD, A. Le Méhauté, C. Rabut, and L. L. Schumaker, eds., Vanderbilt Univ. Press, 1997, pp. 387–394.
[4] C. d. Boor, *A Practical Guide to Splines, Revised Edition*, Springer–Verlag, 2001.
[5] M. J. Borden, M. A. Scott, J. A. Evans, and T. J. R. Hughes, *Isogeometric finite element data structures based on Bézier extraction of NURBS*, Int. J. Numer. Meth. Eng., 87 (2011), pp. 15–47.
[6] C. Bracco, T. Lyche, C. Manni, F. Roman, and H. Speleers, *Generalized spline spaces over T-meshes: Dimension formula and locally refined generalized B-splines*, Appl. Math. Comput., 272 (2016), pp. 187–198.
[7] C. Bracco, T. Lyche, C. Manni, F. Roman, and H. Speleers, *On the dimension of Tchebycheffian spline spaces over planar T-meshes*, Comput. Aided Geom. Design, 45 (2016), pp. 151–173.
[8] C. Bracco, T. Lyche, C. Manni, and H. Speleers, *Tchebycheffian spline spaces over planar T-meshes: Dimension bounds and dimension instabilities*, J. Comput. Appl. Math., 349 (2019), pp. 265–278.
[9] B. Buchwald and G. Mühlbach, *Construction of B-splines for generalized spline spaces generated from local ECT-systems*, J. Comput. Appl. Math., 159 (2003), pp. 249–267.
[10] W. A. Coppel, *Disconjugacy*, Springer–Verlag, 1971.
[11] P. Costantini, T. Lyche, and C. Manni, *On a class of weak Tchebycheff systems*, Numer. Math., 101 (2005), pp. 333–354.

[12] N. Dyn and A. Ron, *Recurrence relation for Tchebycheffian B-splines*, J. Anal. Math., 51 (1988), pp. 118–138.
[13] S. Karlin, *Total Positivity*, Stanford Univ. Press, 1968.
[14] S. Karlin and W. J. Studden, *Tchebycheff Systems: With Applications in Analysis and Statistics*, Interscience Publishers, 1966.
[15] T. Lyche, *A recurrence relation for Chebyshevian B-splines*, Constr. Approx., 1 (1985), pp. 155–173.
[16] T. Lyche, C. Manni, and H. Speleers, *Foundations of spline theory: B-splines, spline approximation, and hierarchical refinement*, in Splines and PDEs: From Approximation Theory to Numerical Linear Algebra, T. Lyche et al., eds., vol. 2219 of Lecture Notes in Mathematics, Springer International Publishing AG, 2018, pp. 1–76.
[17] T. Lyche, C. Manni, and H. Speleers, *Tchebycheffian B-splines revisited: An introductory exposition*, in Advanced Methods for Geometric Modeling and Numerical Simulation, C. Giannelli and H. Speleers, eds., Springer INdAM Series, In press.
[18] T. Lyche and L. L. Schumaker, *A multiresolution tensor spline method for fitting functions on the sphere*, SIAM J. Sci. Comput., 22 (2000), pp. 724–746.
[19] E. Mainar, J. M. Peña, and J. Sánchez-Reyes, *Shape preserving alternatives to the rational Bézier model*, Comput. Aided Geom. Design, 18 (2001), pp. 37–60.
[20] C. Manni, F. Pelosi, and M. L. Sampoli, *Generalized B-splines as a tool in isogeometric analysis*, Comput. Methods Appl. Mech. Eng., 200 (2011), pp. 867–881.
[21] C. Manni, F. Pelosi, and M. L. Sampoli, *Isogeometric analysis in advection-diffusion problems: Tension splines approximation*, J. Comput. Appl. Math., 236 (2011), pp. 511–528.
[22] C. Manni, A. Reali, and H. Speleers, *Isogeometric collocation methods with generalized B-splines*, Comput. Math. Appl., 70 (2015), pp. 1659–1675.
[23] C. Manni, F. Roman, and H. Speleers, *Generalized B-splines in isogeometric analysis*, in Approximation Theory XV: San Antonio 2016, G. E. Fasshauer and L. L. Schumaker, eds., vol. 201 of Springer Proceedings in Mathematics & Statistics, Springer International Publishing AG, 2017, pp. 239–267.
[24] M.-L. Mazure, *Extended Chebyshev piecewise spaces characterised via weight functions*, J. Approx. Theory, 145 (2007), pp. 33–54.
[25] M.-L. Mazure, *Finding all systems of weight functions associated with a given extended Chebyshev space*, J. Approx. Theory, 163 (2011), pp. 363–376.
[26] M.-L. Mazure, *How to build all Chebyshevian spline spaces good for geometric design?*, Numer. Math., 119 (2011), pp. 517–556.
[27] M.-L. Mazure, *Constructing totally positive piecewise Chebyshevian B-spline bases*, J. Comput. Appl. Math., 342 (2018), pp. 550–586.
[28] G. Mühlbach, *ECT-B-splines defined by generalized divided differences*, J. Comput. Appl. Math., 187 (2006), pp. 96–122.
[29] G. Nürnberger, L. L. Schumaker, M. Sommer, and H. Strauss, *Generalized Chebyshevian splines*, SIAM J. Math. Anal., 15 (1984), pp. 790–804.
[30] L. L. Schumaker, *Spline Functions: Basic Theory, 3rd Edition*, Cambridge Univ. Press, 2007.
[31] D. G. Schweikert, *An interpolation curve using a spline in tension*, J. Math. Phys., 45 (1966), pp. 312–317.
[32] M. A. Scott, M. J. Borden, C. V. Verhoosel, T. W. Sederberg, and T. J. R. Hughes, *Isogeometric finite element data structures based on Bézier extraction of T-splines*, Int. J. Numer. Meth. Eng., 88 (2011), pp. 126–156.
[33] T. W. Sederberg, J. Zheng, and X. Song, *Knot intervals and multi-degree splines*, Comput. Aided Geom. Design, 20 (2003), pp. 455–468.
[34] W. Shen and G. Wang, *A basis of multi-degree splines*, Comput. Aided Geom. Design, 27 (2010), pp. 23–35.
[35] W. Shen and G. Wang, *Changeable degree spline basis functions*, J. Comput. Appl. Math., 234 (2010), pp. 2516–2529.
[36] H. Speleers, *Algorithm xxx: Computation of multi-degree B-splines*, ACM Trans. Math. Soft., (In press).
[37] D. Toshniwal, H. Speleers, R. R. Hiemstra, and T. J. R. Hughes, *Multi-degree smooth polar splines: A framework for geometric modeling and isogeometric analysis*, Comput. Methods Appl. Mech. Eng., 316 (2017), pp. 1005–1061.
[38] D. Toshniwal, H. Speleers, R. R. Hiemstra, C. Manni, and T. J. R. Hughes, *Multi-degree B-splines: Algorithmic computation and properties*, Tech. Report ICES 18-22, UT Austin, 2018.
[39] G. Wang and M. Fang, *Unified and extended form of three types of splines*, J. Comput. Appl. Math., 216 (2008), pp. 498–508.