SIAM J. NUMER. ANAL.
Vol. xx, No. x, pp. x–x

NON-UNIFORM DISCONTINUOUS GALERKIN FILTERS VIA SHIFT AND SCALE*

DANG-MANH NGUYEN† AND JÖRG PETERS†

Abstract. Convolving the output of Discontinuous Galerkin computations with symmetric Smoothness-Increasing Accuracy-Conserving (SIAC) filters can improve both smoothness and accuracy. To extend convolution to the boundaries, several one-sided spline filters have recently been developed. This paper interprets these filters as instances of a general class of position-dependent (PSIAC) spline filters that can have non-uniform knot sequences and skip B-splines of the sequence.

PSIAC filters with rational knot sequences have rational coefficients. For prototype knot sequences, such as integer sequences that may have repeated entries, PSIAC filters can be expressed in symbolic form. Based on the insight that filters for shifted or scaled knot sequences are easily derived by non-uniform scaling of one prototype filter, a single filter can be re-used in different locations and at different scales. Computing a value of the convolution then simplifies to forming a scalar product of a short vector with the local output data. Restating one-sided filters in this form improves both stability and efficiency compared to their original formulation via numerical integration. PSIAC filtering is demonstrated for several established and one new boundary filter.

Key words. Discontinuous Galerkin; spline filter; shifted convolution; SIAC filtering; boundary filter; symbolic representation

AMS subject classifications. 65M12; 65D07

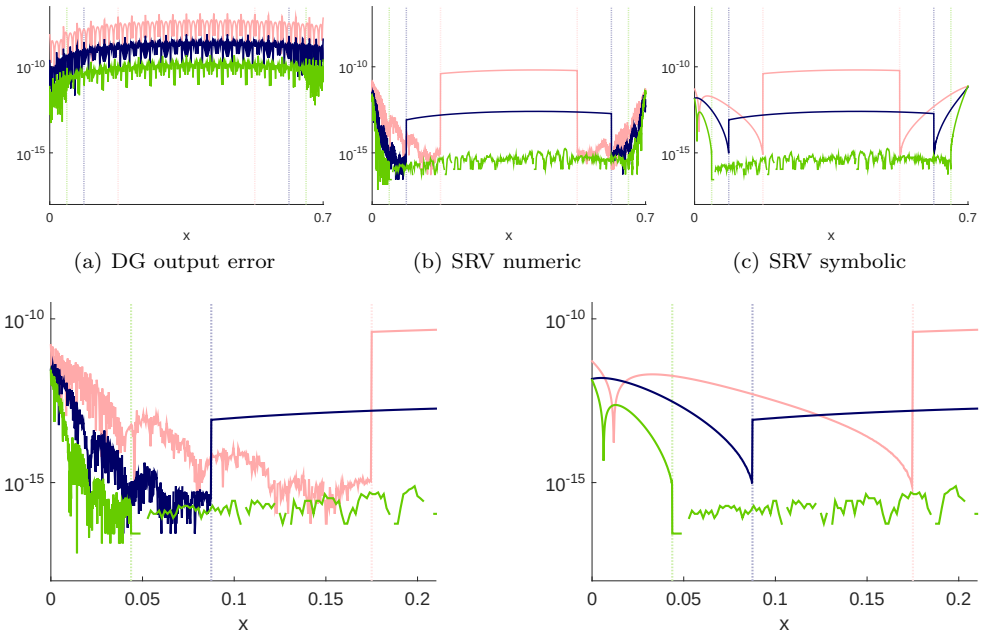
1. Introduction.

Since the output of Discontinuous Galerkin (DG) computations often captures higher order moments of the true solution [ML78], post-processing DG output by convolution can improve both smoothness and accuracy [BS77, CLSS03]. In the interior of the domain of computation, symmetric smoothness-increasing accuracy-conserving (SIAC) filters have been demonstrated to provide optimal accuracy [CLSS03]. However their symmetric footprint precludes using these filters near boundaries of the computational domain.

To filter near boundaries, Ryan and Shu [RS03] pioneered the use of one-sided spline filters. The Ryan-Shu filters improve the L^2 error, but not necessarily the point-wise errors. In fact, the filters are observed to increase the pointwise error near the boundary. This motivated the design of the SRV filter [SRV11], a filter kernel of increased support. Due to near-singular calculations, a stable numerical derivation of the SRV filter requires computing partly in quadruple precision. Indeed, the coefficients of all the boundary filters [RS03, SRV11, MRK12, RLKV15, MRK15] are derived by inverting a matrix whose entries are determined by Gaussian quadrature; and, as pointed out in [RLKV15], SRV filter matrices are close to singular. [RLKV15] therefore introduced the RLKV filter, that augments the filter of [RS03] by a single additional B-spline. This improves stability, retains the support-size and has the boundary filter join the symmetric interior filter without jump. However, on canonical test problems, RLKV filter errors are higher than those of the corresponding symmetric filters and they have sub-optimal L^2 and L^∞ superconvergence rates [RLKV15]. Moreover, RLKV yields poorer derivative approximation than SRV filters [LRKV16]. The goal of this paper is to reformulate boundary filters in a framework that eliminates the need for inverting near-singular matrices and thereby arrive at stable filters that can obtain the optimal accuracy.

*This work was supported in part by NSF grant CCF-1117695 and NIH R01 LM011300-01
†Department of Computer & Information Science & Engineering, University of Florida. (dmNguyen@cise.ufl.edu, jorg@cise.ufl.edu).

In particular, one contribution of this paper is to reinterpret the published one-sided filters in an explicit, symbolic form as position-dependent spline filters. Symbolic expression of coefficients for spline filters have recently been developed in [MRK15] for uniform knot sequences and in [Pet15] for general knot sequences. Reinterpretation of the published filters in symbolic form mitigates their instability and allows them to reach their full potential.



(a) DG output error (b) SRV numeric (c) SRV symbolic

(d) Left zoom of (b) numeric (e) Left zoom of (c) symbolic

FIG. 1. *Point-wise error (y-axis) over the x-domain for* $u(x,0) := .7\sin(\pi\sqrt{10x/7})$. *The graphs in each figure correspond, from top to bottom, to refining the interval width h splitting the domain by* $m = 20, 40, 80$ *break points. The vertical scales of (a), (b) and (c) agree. (a) The pointwise error of* L^2*-projection of* $u(x,0)$*, interpreted as a DG approximation at time* $\tau := 0$*, onto the space of piecewise* cubic *Bernstein-Bézier polynomials. (b,c) The point-wise error after double precision convolution. The vertical dotted lines separate the boundary DG output from the interior data region. In the interior, the graphs of (b) and (c) agree since the same* symmetric SIAC filter *is applied; comparison with (a) illustrates the error reduction.* At the boundaries*, the filter of [SRV11] is computed (b) by the numerical approach vs. (c) as PSIAC filter using the symbolic approach. (d) and (e) enlarge the portion of the filtered output at the left boundary to show the stabilizing effect of the symbolic approach.*

Specifically, this paper

- ▷ proves general properties of SIAC filters when their knots are shifted or scaled;
- ▷ uses these properties to express the filter kernels in a factored, semi-explicit form that becomes explicit for given knot patterns and yields rational coefficients for rational knot sequences;
- ▷ characterizes a class of position-dependent SIAC spline filters (PSIAC filters) whose coefficients turn out to be polynomial expressions in the position for general knot sequences;
- ▷ shows that SRV and RLKV [RS03, SRV11, MRK12, RLKV15, MRK15] are PSIAC filters;
- ▷ illustrates the general framework by comparing the established numerical ap-

proach with the new symbolic filter derivation and by adding a new effective, stably-computed filter of lower degree than SRV or RLKV.

The new characterization allows, in standard double precision, to replace the current three-step *numerical approach* of approximate computation of the matrix, its inversion and application to the data by Gauss quadrature, by a single-step *symbolic approach* derived in Theorem 4.2. Fig. 1 contrasts, for standard double precision, the noisy error of numerical SRV filtering with the error of the new symbolic SRV formulation.

We will demonstrate that the symbolic approach is more stable and show that

- ▷ scaled and shifted versions of the filter are easily obtained from one symbolic prototype filter;
- ▷ computation is more efficient: filtering the DG output reduces to a single dot product of two vectors of small size;
- ▷ computing approximate derivatives of the filtered DG output [RSA05] (see also [Tho77, RC09, LRKV16]) simplifies to differentiating the polynomial representation of the filtered output;
- ▷ the smoothness of the so-filtered DG output is shown to be C^∞.

The last point is of interest since [RS03, SRV11] observed and conjectured that the smoothness of the filtered DG computation is the same as the smoothness in the domain interior where the symmetric filter applied. Not only does this conjecture hold, but the smoothness will be shown to be C^∞.

Organization. Section 2 introduces the canonical test equation, B-splines, convolution, and a generalization of the formula of [Pet15] for convolution with splines based on arbitrary knot sequences. Section 3 reformulates the generalized convolution formula and derives the convolution coefficients when the knot sequence is a scaled or shifted copy of a prototype sequence. Section 4, in particular Theorem 4.2, summarizes the resulting efficient convolution with PSIAC filters based on scaled and shifted copies of a prototype sequence. Section 5 shows that RS, SRV and RLKV are PSIAC filters, and that the symbolic approach improves stability and leads to a broader class of new filters.

2. Notation, Canonical Problem and Filters. This section establishes the notation for filters and DG output, exhibits the canonical test problem, the DG method, B-splines and reproducing filters and reviews one-sided and position-dependent SIAC filters in the literature.

2.1. Convolution, Sequences and Notation. We denote by $f * g$ the convolution of a function f with a function g, i.e.

$$(f * g)(x) := \int_{\mathbb{R}} f(t)\, g(x-t)\, \mathrm{d}t = (g * f)(x),$$

for every x where the integral exists. Filtering means convolving a function with a kernel. Here and in the following, we abbreviate (the ascending or descending sequences)

(2.1)

$$i:j \;:=\; \begin{cases} (i, i+1, \ldots, j-1, j), \text{if } i \le j, \\ (i, i-1, \ldots, j+1, j), \text{if } i > j, \end{cases} \qquad \text{and } s_{i:j} := \begin{cases} (s_i, s_{i+1}, \ldots, s_{j-1}, s_j), \text{if } i \le j, \\ (s_i, s_{i-1}, \ldots, s_{j+1}, s_j), \text{if } i > j. \end{cases}$$

Sequences are also used as summation indices: $\sum_{i:j} := \sum_{\ell=i}^{j}$. We reserve the following symbols:

d	degree of the DG output;
m	number of intervals of the DG output;
$s_{0:m}$	*prototype* increasing break point sequence, typically integers; the break sequence of the DG output is $hs_{0:m}$;
k	degree of the filter kernel;
$r+1$	number of filter coefficients for reproduction of polynomials up to degree r;
$\mathcal{J} := 0 : j_r$	index sequence; if the B-splines of the filter are consecutive, then $j_r = r$;
n	number of knot intervals spanned by the filter; $n = j_r + k + 1$;
$\mathbf{t} := t_{0:n}$	prototype (integer) knot sequence of the filter; the input knot sequence of the filter is $ht_{0:n} + \xi$ where ξ is the shift and h scales.

The notation is illustrated by the following example.

EXAMPLE 2.1. *A linear DG output sequence on 200 uniform segments of the interval* $[-1..1]$ *implies* $d = 1$, $m = 200$, $h = \frac{1}{100}$ *and* $s_{0:m} = -100 : 100$ *a sequence of* $m+1$ *consecutive integers. A degree-one spline filter defined over the knot sequence* $\mathbf{t} := 0 : 6$ *and associated with the index sequence* $\mathcal{J} := (0, 3, 4)$ *corresponds to* $k = 1$, $n = 6$, $r = 2$ *and* $j_r = 4$. *That is, the filter uses the B-splines (defined in Eq.* (2.7)*)* $B(x|(0,1,2))$ $B(x|(3,4,5))$ $B(x|(4,5,6))$ *but omits, or sets to zero the coefficients of the two B-splines* $B(x|(1,2,3))$ $B(x|(2,3,4))$ *defined over the knot sequences* $1:3$ *and* $2:4$.

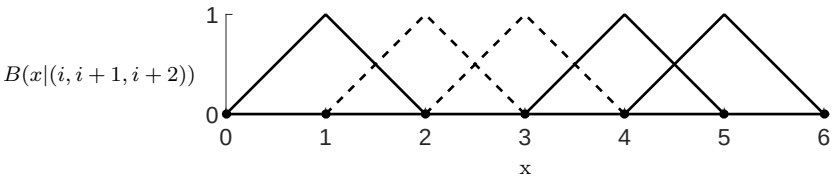


FIG. 2. *Illustration of the index sequence of* $\mathcal{J}$ *of Example. 2.1. The B-splines selected by* $\mathcal{J}$ *have* solid *lines; the skipped B-splines not in* $\mathcal{J}$ *have* dashed *lines.*

2.2. The canonical test problem, the Discontinuous Galerkin method and B-splines. To demonstrate the performance of the filters on a concrete example, [RS03] used the following univariate hyperbolic partial differential wave equation:

$$\begin{aligned} \frac{du}{d\tau} + \frac{d}{dx}\Big(\kappa(x,\tau)\, u\Big) &= \rho(x,\tau), & x \in (a..b), \tau \in (0..\widetilde{\tau}) \\ u(x,0) &= u_0(x), & x \in [a..b] \end{aligned} \tag{2.2}$$

subject to periodic boundary conditions, $u(a,\tau) = u(b,\tau)$, or Dirichlet boundary conditions $u(e,\tau) = u_0(\tau)$ where, depending on the sign of $\kappa(x,\tau)$, e is either a or b. Subsequent work [RS03, SRV11, RLKV15] adopted the same differential equation to test their new one-sided filters and to compare to the earlier work. Eq. (2.2) is therefore considered the *canonical test problem.* We note, however, that SIAC filters apply more widely, for example to FEM and elliptic equations [BS77].

In the DG method, the domain $[a..b]$ is partitioned into intervals by a sequence $hs_{0:m}$ of break points $a =: hs_0, \ldots, hs_m := b$. Assuming that the sequence is rational, scaling by h will later allow us to consider a prototype sequence $s_{0:m}$ of integers. Let $\mathbb{P}_h^d$ be the linear space of all piecewise polynomials with break points $hs_{0:m}$ and of degree less than or equal to d. We use modal or nodal scalar-valued basis functions $\phi_i(.\,;\, hs_{0:m})$ $0 \le i \le m$ of $\mathbb{P}_h^d$ that are linearly independent and satisfy the scaling relations

$$\phi_i(hx\,;\, hs_{0:m}) = \phi_i(x\,;\, s_{0:m}). \tag{2.3}$$

Relation (2.3) is typically used for refinement in FEM, DG or Iso-geometric PDE solvers. Examples of basis functions ϕ_i are Bézier-Bernstein basis functions [dB05], Lagrange polynomials that are defined based on Legendre-Gauss-Lobatto quadrature points [HW07], and Legendre polynomials.

The DG method approximates the time-dependent solution to Eq. (2.2) by

$$u(x,\tau) := \sum_{i=0}^{m} u_i(\tau)\, \phi_i(x\,;\, hs_{0:m}), \quad \phi_i \in \mathbb{P}_h^d. \tag{2.4}$$

Multiplying two sides of Eq. (2.4) with a test function v and integrating by parts yields the weak form of Eq. (2.2):

$$\int_a^b \big(\frac{du}{d\tau}\, v - \kappa(x,\tau) u\, \frac{dv}{dx}\big)\, \mathrm{d}x = \int_a^b \rho(x,\tau)\, v\, \mathrm{d}x - \Big(\kappa(x,\tau) u(x,\tau) v(x)\Big)\Big|_{x=a}^{x=b}. \tag{2.5}$$

Substituting u on the left of Eq. (2.5) by (2.4), treating the rightmost, non-integral term of Eq. (2.5) as a numerical flux, and choosing $v(x) := \phi_j(x\,;\, hs_{0:m})$, yields a system of ordinary differential equations in τ with the coefficients $u_i(\tau)$, $0 \le i \le m$, as unknowns. This system can be solved by, e.g., a standard fourth-order four stage explicit Runge-Kutta method (ERK) [HW07, Section 3.4].

The goal of *SIAC filtering* is to spatially smooth $u(x,\tau)$ by convolution in x with a linear combination of B-splines. Typically filtering is applied after the last time step when $\tau = \widetilde{\tau}$. We define B-splines in terms of divided differences [CS66]. For a sufficiently smooth univariate real-valued function g with kth derivative $g^{(k)}$, divided differences are defined by (cf. [dB05])

$$\begin{aligned} &\Delta_{t_i} g := g(t_i), && \text{and for } j > i \\ &\Delta_{t_{i:j}} g := \begin{cases} (\Delta_{t_{i+1:j}} g - \Delta_{t_{i:j-1}} g)/(t_j - t_i), & \text{if } t_i \ne t_j, \\ \frac{1}{(j-i)!}\, g^{(j-i)}(t_i), & \text{if } t_i = t_j. \end{cases} \end{aligned} \tag{2.6}$$

If $t_{i:j}$ is a non-decreasing sequence, we call its elements t_ℓ *knots*. Repeated knots $t_i = t_j$ equate divided differences with derivatives. The classical definition of the *B-spline of degree k with knot sequence $t_{i:j}$*, $j := i + k + 1$ is

$$B(x|t_{i:j}) := (t_j - t_i)\ \Delta_{t_{i:j}}(\max\{(\cdot - x), 0\})^k. \tag{2.7}$$

Here $\Delta_{t_{i:j}}$ acts on the function $g : t \to (\max\{(t-x), 0\})^k$ for a given $x \in \mathbb{R}$. Consequently, a B-spline is a non-negative piecewise polynomial function in x with support on the interval $[t_i..t_j)$. If ν is the multiplicity of the number t_ℓ in the sequence $t_{i:j}$, then $B(x|t_{i:j})$ is at least $k - \nu$ times continuously differentiable at t_ℓ. This definition

of $B(t\,|\,t_{i:i+k+1})$ agrees, after scaling, with the definition $N(t\,|\,t_{i:i+k+1})$ of the B-spline by recurrence [dB02]:

$$N(t\,|\,t_{i:i+k+1}) = \frac{t_{i+k+1} - t_i}{k+1}\, B(t\,|\,t_{i:i+k+1}). \tag{2.8}$$

2.3. SIAC filter kernel coefficients. A piecewise polynomial $f : \mathbb{R} \to \mathbb{R}$ is said to be a *SIAC spline kernel of reproduction degree* r if convolution of f with monomials reproduces the monomials up to degree r, i.e., if

$$(f * (\cdot)^\delta)(x) = x^\delta, \qquad \delta = 0..r. \tag{2.9}$$

Mirzargar et al. [MRK15] derived semi-explicit formulas when the filter has uniform knots while [Pet15] gives semi-explicit formulas for the coefficients of spline kernels over general knot sequences. The following definition further generalizes these formulas by allowing to skip some B-splines when constructing the kernel.

DEFINITION 2.1 (SIAC spline filter kernel). *Let* $\mathcal{J} := (0, \ldots, j_r)$ *be a sequence of strictly increasing integers between* 0 *and* j_r. *A* SIAC spline kernel *of degree* k *and reproduction degree* r *with index sequence* $\mathcal{J}$ *and knot sequence* $t_{0:n}$ *is a spline*

$$f(x) := \sum_{j\in\mathcal{J}} f_j B(x|t_{j:j+k+1}),$$

of degree k *with coefficients* f_j *chosen so that*

$$\Big(\sum_{j\in\mathcal{J}} f_j B(\cdot|t_{j:j+k+1}) * (-\cdot)^\delta\Big)(x) = (-x)^\delta, \qquad \delta = 0..r. \tag{2.9'}$$

When $\mathcal{J} = 0 : r$ then Definition 2.1 replicates the definition of [Pet15].

LEMMA 2.2 (SIAC coefficients). *The vector* $\mathbf{f} := [f_0, \ldots, f_r]^\mathsf{t} \in \mathbb{R}^{r+1}$ *of B-spline coefficients of the SIAC filter with index sequence* $\mathcal{J} := (0, \ldots, j_r)$ *and knot sequence* $\mathbf{t} := t_{0:n}$ *is*

$$\mathbf{f} := \textit{first column of } M_{0,\mathbf{t},\mathcal{J}}^{-1}, \qquad M_{0,\mathbf{t},\mathcal{J}} := \left[\Delta_{t_{j:j+k+1}} x^{k+1+\delta}\right]_{\delta=0:r,\, j\in\mathcal{J}}. \tag{2.10}$$

Proof. We need only prove invertibility of $M_{0,\mathbf{t},\mathcal{J}}$ since the remaining claims of the lemma then follow as in [Pet15].

Let ℓ be a left null vector of $M_{0,\mathbf{t},\mathcal{J}}$, i.e. for all $j \in \mathcal{J}$, $\kappa := k+1$ and $p(x) := x^\kappa \sum_\delta \ell_\delta x^\delta$

$$0 = \Delta_{t_{j:j+\kappa}} \sum_{\delta=0}^{r} \ell_\delta x^{\kappa+\delta} = \Delta_{t_{j:j+\kappa}} x^\kappa \sum_{\delta=0}^{r} \ell_\delta x^\delta = \Delta_{t_{j:j+\kappa}} p(x). \tag{2.11}$$

Let q be the interpolant of p at $t_{j:j+\kappa+1}$, i.e. spanning two consecutive hence overlapping knot sequences. If knots repeat, q is a Hermite interpolant. By Rolle's theorem, the derivative $D(p-q)$, vanishes at a set of knots $t^1_{j:j+\kappa}$ interlaced with $t_{j:j+\kappa+1}$, i.e. $t_i \le t_i^1 \le t_{i+1}$. The inequality is strict unless $t_i = t_i^1$ represents a multiple root. Then Dq (Hermite) interpolates Dp at $t^1_{j:j+\kappa}$ and by the relation between divided difference and derivatives,

$$\kappa \Delta_{t_{j:j+\kappa}} p = \Delta_{t^1_{j:j+\kappa-1}} Dp, \text{ and } \kappa \Delta_{t_{j+1:j+\kappa+1}} p = \Delta_{t^1_{j+1:j+\kappa}} Dp.$$

Induction in κ yields

$$\kappa!\mathbb{\Delta}_{t_{j:j+\kappa}}p = 2\mathbb{\Delta}_{t_{j:j+1}^{\kappa-1}}D^{\kappa-1}p, \text{ and } \kappa!\mathbb{\Delta}_{t_{j+1:j+\kappa+1}}p = 2\mathbb{\Delta}_{t_{j+1:j+2}^{\kappa-1}}D^{\kappa-1}p$$

and finally

$$\kappa!\mathbb{\Delta}_{t_{j:j+\kappa}}p = D^{\kappa}p(t_j^{\kappa}), \text{ and } \kappa!\mathbb{\Delta}_{t_{j+1:j+\kappa+1}}p = D^{\kappa}p(t_{j+1}^{\kappa})$$

for $t_j^{\kappa} \le t_{j+1}^{\kappa}$. That is, the κth divided difference of each sequence equals the κth derivative at points t_j^{κ} respectively t_{j+1}^{κ}; and the shift of the subsequence of knots from $t_{j:j+\kappa}$ to $t_{j+1:j+\kappa+1}$ implies that where t_{j+1}^{κ} is either strictly to the right of t_j^{κ} or t_j^{κ} is a multiple root. Then (2.11) implies that $D^{\kappa}p$, a polynomial of degree at most r, has $r+1$ roots counting multiplicity, hence is the zero polynomial. Given the factor t^{κ} of p, this can only hold if $\ell = 0$, i.e. $M_{0,\mathbf{t},\mathcal{J}}$ has no non-trivial left null vector and, as a square matrix, $M_{0,\mathbf{t},\mathcal{J}}$ is invertible.

Since all sequences $t_{j:j+\kappa}$, $j \in \mathcal{J}$ can be obtained by repeated shifts to the right, the conclusion $\ell = 0$ holds for $j \in \mathcal{J}$ and $M_{0,\mathbf{t},\mathcal{J}}^{-1}$ is well-defined. □

2.4. Review of Symmetric and Boundary SIAC filters. We split the DG data at any known discontinuities and treat the domains separately. Then convolution can be applied throughout a given closed interval $[a..b]$.

A SIAC spline kernel with knot sequence $t_{0:r+k+1}$ is *symmetric* (about the origin in $\mathbb{R}$) if

$$t_\ell + t_{r+k+1-\ell} = 0, \text{ for } \ell = 0 : \lceil (r+k+1)/2 \rceil. \tag{2.12}$$

Convolution with a symmetric SIAC kernel of a function g at x then requires g to be defined in a two-sided neighborhood of x. Near boundaries, Ryan and Shu [RS03] therefore suggested convolving the DG output with a one-sided kernel whose support is shifted to one side of the origin: for x near the left domain endpoint a, the one-sided SIAC kernel is defined over $(x-a) + h\big(-(3d+1) : 0\big)$ where d is the degree of the DG output. The Ryan-Shu x-position-dependent one-sided kernel yields optimal L^2-convergence, but its point-wise error near a can be larger than that of the DG output.

In [SRV11], Ryan et al. improved the one-sided kernel by increasing its monomial reproduction from degree $r = 2d$ to degree $r = 4d$. This one-sided kernel reduces the boundary error when $d = 1$ but the kernel support is increased by $2d$ additional knot intervals and numerical roundoff requires high precision calculations to determine the kernel's coefficients. ([SRV11] did not draw conclusions for degrees $d > 1$.)

Ryan-Li-Kirby-Vuik [RLKV15] suggested an alternative position-dependent one-sided kernel that has the same support size as the symmetric kernel and its reproduction degree is only higher by one. The new idea is that the spline space defining the kernel is enriched by one B-spline. The new kernel computation is stable up to degree $d = 4$ in double precision. When $d = 1$ the RLKV kernel's point-wise error on the canonical test problem is as low as that of the symmetric SIAC kernel that is applied in the interior. However, when $d > 1$, the RLKV error is higher than that of the symmetric kernel.

In [RS03, SRV11, MRK12, RLKV15] convolution with position-dependent one-sided kernels is computed as follows. For each domain position x,

- ▷ *Calculate kernel coefficients*:
  compute the entries of the position-dependent reproduction matrix M by

Gaussian quadrature; solve a corresponding linear system $M\mathbf{f} = \mathbf{p}$ for the kernel coefficients $\mathbf{f}$ to match the monomials to be reproduced that have been collected in the vector $\mathbf{p}$. As pointed out in [RLKV15], M may be close to singular (for example, when $d \geq 3$ for the SRV kernels) so that higher numerical precision (e.g. quadruple precision) is required to assemble and solve the linear system.

▷ *Convolve the kernel with the DG output* by Gaussian quadrature.

Note that, unlike the (position-independent) classical symmetric SIAC filter, the position-dependent boundary kernel coefficients have to be determined afresh for each point x.

3. Coefficients of shifted and scaled filters. To calculate filters for DG output more efficiently and stably, we formulate Lemma 2.2 in *multi-index notation*:

$$t_{0:n}^{\boldsymbol{\omega}} := t_0^{\omega_0} \dots t_n^{\omega_n} \quad \text{and} \quad |\boldsymbol{\omega}| := \sum_{j=0}^{n} |\omega_j|$$

as follows.

LEMMA 3.1 (SIAC reproduction matrix). *The matrix Eq.* (2.10) *has the alternative form*

$$M := M_{0,t_{0:n},\mathcal{J}} = \left[\sum_{|\boldsymbol{\omega}|=\delta} t_{j:j+k+1}^{\boldsymbol{\omega}} \right]_{\delta=0:r,\, j\in\mathcal{J}}. \tag{3.1}$$

Proof. Applying Steffensen's formula [dB05, Eq. (27)][1], the divided differences of monomials in (2.10) can be rewritten as (3.1). □

Denote the reproduction matrix associated with the shifted knot sequence $t_{j:j+k+1} + \xi$, $\xi \in \mathbb{R}$ as

$$M_\xi = \big[M_\xi(\delta, j)\big]_{\delta=0:r,\, j\in\mathcal{J}} := \left[\sum_{|\boldsymbol{\omega}|=\delta} (t_{j:j+k+1} + \xi)^{\boldsymbol{\omega}} \right]_{\delta=0:r,\, j\in\mathcal{J}}. \tag{3.2}$$

Since each entry $M_\xi(\delta, j)$ is a polynomial of degree δ in ξ, the determinant of M_ξ is a polynomial of degree $r(r+1)/2$ in ξ. Using for example Cramer's rule, the entries of M_ξ^{-1} are rational functions in ξ whose numerator and denominator are polynomials of degree $r(r+1)/2$ in ξ. Since the convolution coefficients are the entries of the first column of M_ξ^{-1}, it is remarkable, that we can show that the coefficients are not rational but polynomial and of degree r rather than $r(r+1)/2$ in ξ.

To prove this claim, we employ the following technical result that generalizes a binomial identity to multiple indices.

LEMMA 3.2 (A binomial identity). *We abbreviate* $\binom{\mathbf{b}}{\mathbf{a}} := \binom{b_0}{a_0} \dots \binom{b_{k+1}}{a_{k+1}}$ *and write* $\mathbf{a} \leq \mathbf{b}$ *to indicate that, for each* kth *component,* $a_k \leq b_k$. *Then for* $\delta > |\mathbf{a}|$,

$$\sum_{\boldsymbol{\alpha}\geq\mathbf{a},\, |\boldsymbol{\alpha}|=\delta} \binom{\boldsymbol{\alpha}}{\mathbf{a}} = \binom{\delta+k+1}{|\mathbf{a}|+k+1} = \binom{\delta+k+1}{\delta-|\mathbf{a}|}. \tag{3.3}$$

[1] The index α in [dB05, Eq. (27)] is misprinted. It should be: $|\alpha| = n - k + 1$.

Proof. By the Maclaurin expansion: $\frac{1}{(1-x)^{k+1}} = \sum_{\ell=0}^{\infty} \binom{k+\ell}{k} x^\ell$, $|x| < 1$, we see that

$$\frac{x^k}{(1-x)^{k+1}} = \sum_{\ell \geq k} \binom{\ell}{k} x^\ell, \quad |x| < 1. \tag{3.4}$$

Applying Eq. (3.4) to both sides of the following identity

$$x^{k+1} \frac{x^{a_0}}{(1-x)^{a_0+1}} \cdots \frac{x^{a_{k+1}}}{(1-x)^{a_{k+1}+1}} = \frac{x^{|\mathbf{a}|+k+1}}{(1-x)^{|\mathbf{a}|+k+2}}, \tag{3.5}$$

and dividing both sides by x^{k+1}, we see that

$$\sum_{\boldsymbol{\alpha} \geq \mathbf{a}} \binom{\boldsymbol{\alpha}}{\mathbf{a}} x^{|\boldsymbol{\alpha}|} = \sum_{\ell \geq |\mathbf{a}|+k+1} \binom{\ell}{|\mathbf{a}|+k+1} x^{\ell-k-1}. \tag{3.6}$$

That is, multiplying both sides of Eq. (3.6) by x^{k+1} yields Eq. (3.5). Selecting the coefficients of x^δ from both sides of Eq. (3.6) yields Eq. (3.3). □

LEMMA 3.3 (Reproduction matrix for shifted knots). *The matrices M_ξ and $M := M_{0,t_{0:n},\mathcal{J}}$ are related by*

$$M_\xi = P_{k,r}(\xi)\, M := \begin{bmatrix} 1 & 0 & \cdots & 0 \\ \binom{1+k+1}{1}\xi & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \binom{r+k+1}{r}\xi^r & \cdots & \binom{r+k+1}{1}\xi & 1 \end{bmatrix} M. \tag{3.7}$$

Note that $P_{k,r}(1)$ is the result of deleting the first $k+1$ rows and $k+1$ columns of the lower triangular Pascal matrix of order $r+k+1$, a fact that will help in deriving a symbolic inverse of $P_{k,r}(\xi)$.

Proof. Abbreviating $\mathbf{t}_j := t_{j:j+k+1}$, the entry (δ, j) of M_ξ defined by Eq. (3.2) is

(3.8)

$$\begin{aligned} M_\xi(\delta, j) &= \sum_{|\boldsymbol{\omega}|=\delta} \sum_{0 \leq \boldsymbol{\ell} \leq \boldsymbol{\omega}} \binom{\boldsymbol{\omega}}{\boldsymbol{\ell}} \xi^{|\boldsymbol{\ell}|}\, \mathbf{t}_j^{\boldsymbol{\omega}-\boldsymbol{\ell}} \\ &= \sum_{|\boldsymbol{\omega}|=\delta} \sum_{\beta=0}^{\delta} \xi^\beta \sum_{|\boldsymbol{\ell}|=\beta,\, 0 \leq \boldsymbol{\ell} \leq \boldsymbol{\omega}} \binom{\boldsymbol{\omega}}{\boldsymbol{\ell}} \mathbf{t}_j^{\boldsymbol{\omega}-\boldsymbol{\ell}} = \sum_{\beta=0}^{\delta} \xi^\beta \sum_{|\mathbf{a}|=\delta-\beta} \mathbf{t}_j^{\mathbf{a}} \sum_{|\boldsymbol{\omega}|=\delta,\, \boldsymbol{\omega} \geq \mathbf{a}} \binom{\boldsymbol{\omega}}{\mathbf{a}}. \end{aligned}$$

The last equality of Eq. (3.8) follows by substituting $\mathbf{a} = \boldsymbol{\omega} - \boldsymbol{\ell}$. Applying Lemma 3.2 to Eq. (3.8) and noting that $|\mathbf{a}| = \delta - \beta$, we see that

$$M_\xi(\delta, j) = \sum_{\beta=0}^{\delta} \binom{\delta+k+1}{\beta} \xi^\beta M(\delta-\beta, j). \tag{3.9}$$

Eq. (3.9) is the expanded form of Eq. (3.7). □

Next we consider the effect of scaling knots, as might be done to refine a DG computation.

LEMMA 3.4 (Reproduction matrix for scaled knots). *Let* $\mathrm{diag}(\mathbf{v})$ *denote the square matrix with diagonal* $\mathbf{v}$ *and zero otherwise. Then for* $\mathbf{t} := t_{0:n}$

$$M_{0,h\mathbf{t},\mathcal{J}}^{-1} = M_{0,\mathbf{t},\mathcal{J}}^{-1}\, \mathrm{diag}(h^{-(0:r)}), \qquad \textit{where } h^{-(0:r)} := [1, h^{-1}, \ldots, h^{-r}]. \tag{3.10}$$

Proof. By Lemma 3.1, multiplying the $(\delta+1)$-th row of $M_{0,\mathbf{t},\mathcal{J}}$ by h^δ yields $M_{0,h\mathbf{t},\mathcal{J}}$, $\delta = 0,\dots,r$ and hence

$$M_{0,h\mathbf{t},\mathcal{J}} = \operatorname{diag}([1,h,\dots,h^r])\, M_{0,\mathbf{t},\mathcal{J}}$$

which is equivalent to (3.10). ∎

Together, we obtain the following semi-explicit formula for the filter coefficients.

THEOREM 3.5 (Scaled and shifted SIAC coefficients). *The SIAC filter coefficients $f_{\xi;\ell}$ associated with the knot sequence $h\mathbf{t}+\xi$ are polynomials of degree r in ξ:*

$$\mathbf{f}_\xi := [f_{\xi,\ell}]_{\ell=0:r} = \; M_{0,\mathbf{t},\mathcal{J}}^{-1}\, \operatorname{diag}\Big(\big[(-1)^\ell \tbinom{\ell+k+1}{\ell}\big]_{\ell=0:r} \Big) \Big[\big(\frac{\xi}{h}\big)^{0:r} \Big]^{\mathtt{t}}. \tag{3.11}$$

Proof. By Lemma 3.3 and Lemma 3.4, the matrix M_ξ corresponding to the scaled and shifted knot sequence $h\mathbf{t}+\xi$ has the inverse

$$M_\xi^{-1} = M_{0,\mathbf{t},\mathcal{J}}^{-1} \operatorname{diag}(h^{-(0:r)})(P_{k,r}(\xi))^{-1}. \tag{3.12}$$

According to [ES04], $P_{k,r}(\xi) = (P_{k,r}(1))^\xi$ and hence $(P_{k,r}(\xi))^{-1} = P_{k,r}^{-\xi}(1) = P_{k,r}(-\xi)$. Since Theorem 2.2 requires only the first column of M_ξ^{-1}, we replace, in (3.12), $P_{k,r}(-\xi)$ by its first column and obtain

$$\mathbf{f}_\xi = M_{0,\mathbf{t},\mathcal{J}}^{-1} \operatorname{diag}(h^{-(0:r)}) \operatorname{diag}\big(\big[(-1)^\ell \tbinom{\ell+k+1}{\ell}\big]_{\ell=0:r}\big) \big(\xi^{0:r}\big)^{\mathtt{t}}. \tag{3.13}$$

Since the diagonal matrices in Eq. (3.13) commute, we have proven Eq. (3.11). ∎

Compared to (3.13), formula (3.11) is advantageous in that two matrices are grouped together that can be pre-computed independent of h and ξ.

The following corollary implies that the kernel coefficients $f_{\xi,\ell}$, can be computed stably, as scaled integers.

COROLLARY 3.6 (Rational SIAC filter coefficients $f_{\xi,\ell}$). *If the knots $t_{0:n}$ are rational, then the filter coefficients $f_{\xi,\ell}$ are polynomials in ξ/h with rational coefficients.*

Proof. Lemma 3.1 implies that the entries of M are rational if the knots are rational. Since the determinant of a matrix with rational entries is rational, for example Cramer's rule implies that the convolution coefficients are rational. ∎

4. Position-dependent (PSIAC) filtering. In this section, we first derive a general factored expression for the convolution of PSIAC filters with DG data. Then we specialize the setup to one-sided PSIAC filters when the DG breakpoint sequence is uniform.

4.1. PSIAC filters. When symmetric SIAC filtering increases the smoothness of the DG output, the result is in general a piecemeal function. One may expect the same of any one-sided kernel. However, this section proves that convolution with PSIAC filters yields a single polynomial piece over their interval of application. We start by defining position-dependent kernels.

DEFINITION 4.1 (PSIAC kernel). *A PSIAC kernel at position x has the form*

$$f_x(s) := \sum_{j\in\mathcal{J}} f_{x;j} B(s \,|\, ht_{j:j+k+1} + x), \quad s \in h[t_0..t_n] + x. \tag{4.1}$$

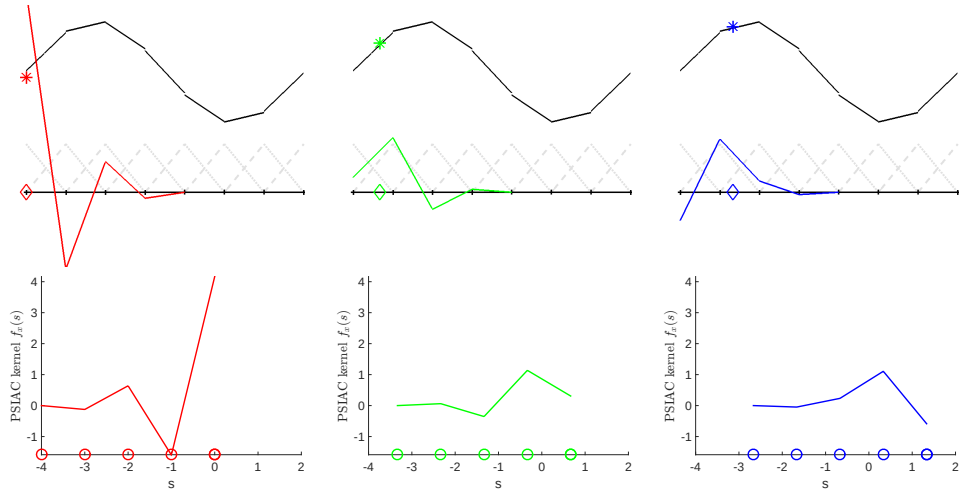


FIG. 3. *Position-dependent filtering at three locations x (marked as a diamond) near the left endpoint a (red diamond) of the* discontinuous *DG output. Note the small gaps between the segments. (Top) The resulting convolved values are marked by a star. (Bottom) The rightmost knot of the kernel is at location $x - a$.*

Fig. 3 illustrates position-dependent filtering when $d = k = 1$. Here we introduced the scaling h in anticipation of using a prototype knot sequence, typically a subset of integers, to create one filter whose shifts and scaled versions can be efficiently computed as explained in the previous section. That is, the DG output will be convolved with a PSIAC kernel $f_{x-h\lambda}(s)$ of reproduction degree r, associated with an index sequence $\mathcal{J}$ and defined over the scaled and shifted knot sequence $ht_{0:n} + x - h\lambda$ where the constant $h\lambda$ adjusts to the left or right boundary. Substituting $x - h\lambda$ for x in Eq. (4.1), changing variable from s to $x - s$ and recalling that $n = j_r + k + 1$, we obtain an alternative spline representation of $f_{x-h\lambda}(s)$ with x-dependent coefficients $f_{j_r-j} := f_{x-h\lambda;j_r-j}$ over the shifted and reversed knot sequence $h(\lambda - t_{n:0})$:

$$f_{x-h\lambda}(s) = \sum_{j\in\mathcal{J}} f_{j_r-j}\, B(x - s \,|\, h\lambda - ht_{n-j:j_r-j}), \quad s \in h[t_0..t_n] + x - h\lambda. \tag{4.2}$$

Now consider the DG output $u(s,\tau)$ with break point sequence $s_{0:m}$ as in Eq. (2.4). Convolving $u(s,\tau)$ with $f_{x-h\lambda}(s)$ yields, after a change of variable $t := x - s$, the *filtered DG output*

$$\begin{aligned}\big(u{*}f_{x-h\lambda}\big)(x) &= \sum_{j\in\mathcal{J}} f_{j_r-j} \int_{ht_0+x-h\lambda}^{ht_n+x-h\lambda} u(x-s,\tau)\, B(x - s \,|\, h\lambda - ht_{n-j:j_r-j})\,\mathrm{d}s \\ &= \sum_{i\in\mathcal{I},\, j\in\mathcal{J}} u_i(\tau)\, f_{j_r-j} \int_{h\lambda-ht_n}^{h\lambda-ht_0} \phi_i(t\,;\, hs_{0:m})\, B(t \,|\, h\lambda - ht_{n-j:j_r-j})\,\mathrm{d}t.\end{aligned} \tag{4.3}$$

We note in the last expression that the integral no longer depends on x. Now the coefficient $f_{j_r-j} := f_{x-h\lambda;j_r-j}$ depends on x: by Theorem 3.5 f_{j_r-j} is a polynomial

in x. This yields the following factored representation of the convolution.

THEOREM 4.2 (Efficient PSIAC filtering of DG output). *Let $f_x(s)$ be a PSIAC kernel of reproduction degree r with index sequence $\mathcal{J} = 0 : j_r$ and knot sequence $ht_{0:n} + x - h\lambda$. Let $u(x,\tau) := \sum_{i=0}^{m} u_i(\tau)\,\phi_i(x\,;\, hs_{0:m})$, $x \in [a..b]$ and $\tau \geq 0$, be the DG output. Let $\mathcal{I}$ be the set of indices of basis functions $\phi_i(.\,;\, hs_{0:m})$ with support overlapping $h[\lambda - t_n..\lambda - t_0]$. Then the* filtered DG approximation *is a polynomial in x of degree r:*

$$\big(u * f_x\big)(x) = \mathbf{u}_{\mathcal{I}}\, Q_\lambda \Big[\big(\frac{x}{h} - \lambda\big)^{0:r}\Big]^{\mathtt{t}}. \tag{4.4}$$

$$\mathbf{u}_{\mathcal{I}} := \big[u_i(\tau)\big]_{i\in\mathcal{I}},$$

$$Q_\lambda := G_\lambda\, A\, M^{-1}_{0,\mathbf{t},\mathcal{J}}\ \mathrm{diag}(\big[(-1)^\ell \tbinom{\ell+k+1}{\ell}\big]_{\ell=0:r}),$$

$$G_\lambda := \left[\int_{\lambda-t_n}^{\lambda-t_0} \phi_i(s\,;\, s_{0:m})\, B(s \,|\, \lambda - t_{n-j:j_r-j})\,\mathrm{d}s\right]_{i\in\mathcal{I},\, j\in\mathcal{J}}, \tag{4.5}$$

where A is the reversal matrix (1 on the anti-diagonal and zero else).

Proof. Let $f_{r:0} := f_{x-h\lambda;r:0}$ be the position-dependent coefficients of the parametrized kernel $f_{x-h\lambda}$ arranged in reverse order. We rewrite Eq. (4.3) as

$$\big(u{*}f_{x-h\lambda}\big)(x) = \mathbf{u}_{\mathcal{I}}\, G_{h\lambda}\, f_{r:0}, \tag{4.6}$$

$$G_{h\lambda} := \left[\int_{h(\lambda-t_n)}^{h(\lambda-t_0)} \phi_i(t\,;\, hs_{0:m})\, B(t \,|\, h\lambda - ht_{n-j:j_r-j})\,\mathrm{d}t\right]_{i\in\mathcal{I},\, j\in\mathcal{J}}.$$

It follows from Eq. (2.3) that

$$B(hs \,|\, h\lambda - ht_{n-j:j_r-j}) = \frac{1}{h} B(s \,|\, \lambda - t_{n-j:j_r-j}).$$

Then the change of variable $s := t/h$ yields

$$G_{h\lambda}(i,j) = \int_{\lambda-t_n}^{\lambda-t_0} \phi_i(hs\,;\, hs_{i:i+d+1})\, B(hs \,|\, h\lambda - ht_{n-j:j_r-j})\, h\,\mathrm{d}s = G_\lambda(i,j).$$

By Theorem 3.5

$$f_{r:0} = A f_{0:r} = A\, M^{-1}_{0,t_{0:n},\mathcal{J}}\ \mathrm{diag}\big(\,\big[(-1)^\ell \tbinom{\ell+k+1}{\ell}\big]_{\ell=0:r}\,\big)\,\Big(\big(\frac{x}{h} - \lambda\big)^{0:r}\Big)^{\mathtt{t}}. \tag{4.7}$$

Substituting Eq. (4.7) into Eq. (4.6) yields Eq. (4.4). □

The factored representation implies that instead of recomputing the filter coefficients afresh for each point x of the convolved output as in the established numerical approach, we simply compute the coefficients corresponding to one prototype knot sequence $\mathbf{t}$, scale by h as needed and at runtime pre-multiply with the data and post-multiply with the vector of shifted monomials as stated in Eq. (4.4).

Increased multiplicity of an inner knot of the symmetric, position-independent SIAC kernel reduces its smoothness, and this, in turn, reduces the smoothness of the filtered output. By contrast, Theorem 4.2 shows that when the PSIAC knots are shifted along evaluation points x then PSIAC convolution yields a polynomial, i.e. infinite smoothness regardless of the knot multiplicity. That is, we may view position-dependent filtering as a form of polynomial approximation.

The polynomial characterization directly provides a symbolic expression for the derivatives of the convolved DG output. By differentiating ℓ times both sides of Eq. (4.4), we arrive at the following result.

COROLLARY 4.3 (Derivatives of PSIAC-filtered DG output).

$$\frac{\mathrm{d}^\ell}{\mathrm{d}x^\ell}\big(u * f_x\big)(x) = \mathbf{u}_{\mathcal{I}}\, Q_\lambda\, \mathrm{diag}(h^{-(0:r)})\, \big(\frac{\mathrm{d}^\ell}{\mathrm{d}x^\ell}(x - h\lambda)^{0:r}\big)^{\mathtt{t}}. \tag{4.8}$$

The following Corollary shows that for rational DG break points and rational kernel knots, we can pre-compute and store the prototype matrix Q_λ stably in terms of integer fractions.

COROLLARY 4.4 (Rational PSIAC convolution coefficients). *With the assumptions and notation of Theorem 4.2, if the basis functions $\phi_i(.\,; s_{0:m})$ are piecewise polynomials with rational coefficients, the shift λ is rational and the sequences $t_{0:n}$ and $s_{0:m}$ are rational then the matrix Q_λ has rational entries.*

Proof. By Lemma 3.1, the reproduction matrix $M^{-1}_{0,\mathbf{t},\mathcal{J}}$ has rational entries. Since the integral of a polynomial with rational coefficients over an interval with rational end points is rational, Eq. (4.5) implies that G_λ also has rational entries. This implies that the entries of Q_λ are rational. ∎

4.2. Application to filtering at boundaries. We now derive explicit forms of the matrix G_λ of Theorem 4.2 when the DG break points $s_{0:m}$ are uniform and the PSIAC filters are one-sided. For one-sided filters, λ is replaced by λ_L and λ_R for the left-sided and right-sided kernels respectively. Recalling that convolution reverses the direction of the filter kernel (cf. Fig. 3), it is natural to assume that, for the left boundary kernel, the right-most knot is zero when evaluating at the left endpoint $x = a$, i.e. $(ht_n + x - h\lambda_L)|_{x=a} = 0$. An analogous assumption applies at the right boundary. This yields

$$\lambda_L = t_n + \frac{a}{h}, \qquad \lambda_R = t_0 + \frac{b}{h}. \tag{4.9}$$

COROLLARY 4.5 (G_λ matrices for uniform DG spacing). *Assume that the DG break point sequence $s_{0:m}$ is uniform, hence after scaling consists of consecutive integers. Without loss of generality, the DG output on each interval $[s_i..s_{i+1}]$ is defined in terms of Bernstein-Bézier polynomials B^i_ℓ of degree d, where the superscript i indicates the interval and $\ell = 0..d$, i.e.*

$$B^i_\ell(x) := \begin{cases} \binom{d}{\ell}(x - s_i)^\ell (s_{i+1} - x)^{d-\ell} & \text{if } x \in [s_i..s_{i+1}] \\ 0 & \text{otherwise.} \end{cases}$$

Let n_0 be the smallest integer greater than or equal to $t_n - t_0$. Then, for $i = 0..(n_0 - 1)$, $\ell = 0..d$, and $j \in \mathcal{J}$

$$\begin{aligned} G_{\lambda_L}\big((d+1)i + \ell, j\big) &= \int_i^{i+1} B^i_\ell(t)\, B(t \,|\, t_n - t_{n-j:j_r-j)})\, \mathrm{d}t, \\ G_{\lambda_R}\big((d+1)i + \ell, j\big) &= \int_{n_0-i}^{n_0-i+1} B^{n_0-i}_{d-\ell}(t)\, B(t \,|\, t_{j:j+k+1} - t_0)\, \mathrm{d}t. \end{aligned} \tag{4.10}$$

Proof. First we consider G_{λ_L}. In Eq. (4.5), we change to the variable $t = s - \lambda_L + t_n$. Abbreviating $\widehat{t_j} := t_n - t_{n-j}$, since the B-splines are translation invariant,

for $0 \le \widehat{t}_j = t_n - t_{n-j} \le t_n - t_0 \le n_0$,

$$G_{\lambda_L}(i,j) = \int_0^{t_n - t_0} \phi_i(t\,;\, s_{0:m} - \lambda_L + t_n)\, B(t \,|\, \widehat{t}_{j:j+k+1})\, \mathrm{d}t. \tag{4.11}$$

Since $s_0 = \frac{a}{h}$, Eq. (4.9) implies that the first point of the sequence of translated DG break points $s_{0:m} - \lambda_L + t_n$ equals 0, i.e., $s_0 - \lambda_L + t_n = 0$. Since the break points are consecutive integers starting from 0 and $0 \le \widehat{t}_j \le n_0$ the relevant DG break points are $0 : n_0$. We re-index the basis functions $\phi_i(s\,;\, s_{0:m} - \lambda_L + t_n)$ in terms of the Bernstein-Bézier basis functions B^i_ℓ. Since B^i_ℓ are non-zero on any interval $[i..i+1]$, Eq. (4.10) for G_{λ_L} follows from Eq. (4.11).

Now consider G_{λ_R}. The change of variable $t := -(s - \lambda_R + t_0)$ together with the translation invariance of B-splines imply that

$$\begin{aligned} G_{\lambda_R}(i,j) &= \int_0^{t_n - t_0} \phi_i(-t\,;\, s_{0:m} - \lambda_R + t_0)\, B(-t \,|\, t_0 - t_{n-j:j_r-j})\, \mathrm{d}t. \\ &= \int_0^{t_n - t_0} \phi_{m-i}(t\,;\, \lambda_R - t_0 - s_{m:0})\, B(t \,|\, t_{j:j+k+1} - t_0)\, \mathrm{d}t. \end{aligned} \tag{4.12}$$

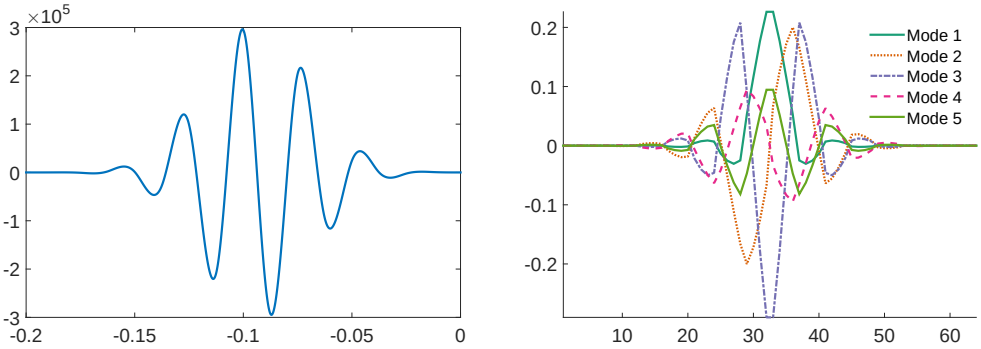
Because of Eq. (4.9) and $s_m = \frac{b}{h}$, $\lambda_R - t_0 - s_m = 0$. Eq. (4.10) is derived by re-indexing the basis functions ϕ_{m-i} as for the left boundary except that we have to count the functions B^i_ℓ backward from the last interval $[t_{n_0-1}..t_{n_0}]$ to the first one $[t_0..t_1]$ when i and j increase. □

The explicit form of G_λ in Corollary 4.5 affords stable pre-computation of the entries of Q_λ. Representing the DG output in Bernstein-Bézier form is for convenience only: the DG output can be represented in any piecewise polynomial form to arrive at similar formulas.

According to Theorem 4.2, the superposition of h-scaled columns of Q_λ multiplying the input data yields the filtered output. For each column of Q_λ, we can plot each entry as ordinate using its row index as abscissa. Linearly connecting consecutive points yields a piecewise linear function (see e.g. Fig. 4b) that we call 'mode function'. Fig. 4 illustrates for [SRV11] the difference in magnitude between the kernel and the mode functions: The kernel, shown in Fig. 4a, is six orders of magnitude larger than the largest mode function in Fig. 4b. (The remaining seven mode functions come in groups that are 2 or 4 orders of magnitude smaller.) So, even while the filter coefficients are very large and alternating, they can be stably combined in a one-time symbolic computation to form the entries of the mode matrix G_λ. And since G_λ has small entries compared to the kernel coefficients, applying G_λ to the data requires no increased precision.

5. PSIAC kernels. In this section, we first restate the SRV and the RLKV filters as PSIAC filters and compare convolution using the numerical approach to using the symbolic approach. Then, to illustrate the generality of the setup, we define a new multiple-knot linear filter and compare it to the SRV and the RLKV filters.

5.1. An alternative view of the published boundary filters. We now recast several published boundary filters as special cases of PSIAC kernels defined over shifted knots in the sense of Theorem 4.2. The published filters considered in this section all chose $d = k$, i.e. the filter has the same degree as the DG output.



(a) [SRV11] kernel (scale is 10^5) (b) Leading [SRV11] mode functions

FIG. 4. *(a) Left-sided [SRV11] kernel of degree 3 over uniform knots (interval size $h = 1/80$). The coefficients are of size 10^5 and their sign alternates. (b) The leading h-independent [SRV11] mode functions defined by the columns of Q_λ are six orders of magnitude smaller.*

RS and SRV filters. The boundary SIAC filters RS [RS03] and SRV [SRV11] of reproduction degree r are both defined over the following shifted knots:

$$\mu := \frac{r+d+1}{2}, \quad \lambda_{L,d} := a + \mu, \quad \lambda_{R,d} := b - \mu$$
$$\boldsymbol{t}_{*,d}(x) := \Big(-\mu, -\mu+1, \ldots, \mu \Big) + x - \lambda_{*,d}. \tag{5.1}$$

Here L or R are substituted for $*$, we obtain the left-side and the right-side filters respectively. Both types of filter are associated with a consecutive index sequence $\mathcal{J}$. However, the two kernels have different reproduction degrees: $r(\mathrm{RS}) = 2d$ and $r(\mathrm{RV}) = 4d$. The prototype knot sequences $\boldsymbol{t}_{*,d}(\lambda_{*,d})$ are chosen symmetrically supported about the origin.

In Proposition 5.1 below we denote as $\mathbf{b}_k$ the vectors of Bernstein-Bézier coefficients of the $d+1$ polynomial pieces of the uniform B-spline of degree d defined over the knots $0 : (d+1)$. For example,

$$\begin{aligned} d = 1: &\quad [\mathbf{b}_0\, \mathbf{b}_1] = \left[\begin{smallmatrix} 0 & 1 \\ 1 & 0 \end{smallmatrix}\right] \\ d = 2: &\quad [\mathbf{b}_0\, \mathbf{b}_1\, \mathbf{b}_2] = \frac{1}{2}\left[\begin{smallmatrix} 0 & 1 & 1 \\ 0 & 2 & 0 \\ 1 & 1 & 0 \end{smallmatrix}\right] \\ d = 3: &\quad [\mathbf{b}_0\, \mathbf{b}_1\, \mathbf{b}_2\, \mathbf{b}_3] = \frac{1}{6}\left[\begin{smallmatrix} 0 & 1 & 4 & 1 \\ 0 & 2 & 4 & 0 \\ 0 & 4 & 2 & 0 \\ 1 & 4 & 1 & 0 \end{smallmatrix}\right]. \end{aligned} \tag{5.2}$$

PROPOSITION 5.1. *Let $\mathbf{b}_k$ denote the Bernstein-Bézier coefficients of polynomial pieces of a uniform B-spline of degree d defined over the knots $0 : (d+1)$ and $\mathcal{M}_d$ the*

matrix with entries $\mathcal{M}_d(\ell, j) := \frac{1}{2d+1}\binom{d}{\ell}\binom{d}{j}\binom{2d}{\ell+j}^{-1}$, $\ell, j = 0..d$. *Then*

$$G_{\lambda_L} := G_{\lambda_{L,d}} = \begin{bmatrix} \mathcal{M}_d\mathbf{b}_0 & 0 & \cdots & 0 \\ \mathcal{M}_d\mathbf{b}_1 & \mathcal{M}_d\mathbf{b}_0 & \cdots & 0 \\ \vdots & \vdots & \cdots & \vdots \\ \mathcal{M}_d\mathbf{b}_d & \mathcal{M}_d\mathbf{b}_{d-1} & \cdots & 0 \\ 0 & \mathcal{M}_d\mathbf{b}_d & \cdots & 0 \\ 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \cdots & \vdots \\ 0 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & \mathcal{M}_d\mathbf{b}_0 \\ \vdots & \vdots & \cdots & \vdots \\ 0 & 0 & \cdots & \mathcal{M}_d\mathbf{b}_d \end{bmatrix} \tag{5.3}$$

and G_{λ_R} *is obtained from* G_{λ_L} *by reversing the order of the columns and of the rows.*

Proof. Using the notation of Corollary 4.5, $n_0 = n = r+d$, since $\widehat{t}_{0:n} = t_n - t_{n:0} = 0 : (r+d)$,

$$G_{\lambda_L}\big((d+1)i+\ell, j\big) = \int_i^{i+1} B_\ell^i(t)\, B(t \,|\, j : j+d+1)\,\mathrm{d}t, \tag{5.4}$$

When i and ℓ vary, Eq. (5.4) gives the jth column of G_{λ_L}. The support of $B(t \,|\, j : j+d+1)$ contains $d+1$ intervals $[j+\rho..j+\rho+1]$, $\rho = 0..d$. Restricted to the interval $[j+\rho..j+\rho+1]$, $\ell = 0..d$, we can write $B(\cdot \,|\, j : j+d+1)$ in terms of the Bernstein-Bézier basis functions $B_\ell^{j+\rho}$

$$B(\cdot \,|\, j : j+d+1)\big|_{[j+\rho, j+\rho+1]} = \left[B_0^{j+\rho} \cdots B_d^{j+\rho}\right] \mathbf{b}_\rho.$$

When $[i..i+1] \equiv [j+\rho..j+\rho+1]$, i.e. when $i = j+\rho$, we can therefore rewrite Eq. (5.4) as

$$\begin{aligned} \big[G_{\lambda_L}\big((d+1)i+\ell, j\big)\big]_{\ell=0:d} &= \int_i^{i+1} \big[B_\ell^i(t)\big]_{\ell=0:d} \, \big[B_j^i(t)\big]_{j=0:d}^{\mathtt{t}} \mathbf{b}_\rho \,\mathrm{d}t, \\ &= \left[\int_0^1 B_\ell^0(t)\, B_j^0(t)\,\mathrm{d}t\right]_{\ell=0:d, j=0:d} \mathbf{b}_\rho =: \mathcal{M}_d\mathbf{b}_\rho, \end{aligned} \tag{5.5}$$

where

$$\begin{aligned} \mathcal{M}_d(\ell, j) &= \int_0^1 B_\ell^0(t)\, B_j^0(t)\,\mathrm{d}t \\ &= \binom{d}{\ell}\binom{d}{j}\binom{2d}{\ell+j}^{-1} \int_0^1 \binom{2d}{\ell+j} x^{\ell+j}(1-x)^{2d-\ell-j}\,\mathrm{d}t. \end{aligned} \tag{5.6}$$

Since the integral in Eq. (5.6) is the integral of a Bernstein-Bézier basis function of degree $2d$ that equals $\frac{1}{2d+1}$ [dB02], we have derived the formula for $\mathcal{M}_d(\ell, j)$ in Eq. (5.3). The formula for G_{λ_L} follows, because the entries of the jth column vanish when the two intervals $[i..i+1]$ and $[j+\rho..j+\rho+1]$ do not overlap.

The formula for G_{λ_R} in Eq. (4.10) is that of G_{λ_L} except that the B-splines appear in reverse order. Therefore, reversing the column order of G_{λ_L} and then the row order of the result yields G_{λ_R}. □

RLKV filter. The index sequence $\jmath$ of the boundary kernel RLKV is non-consecutive. The left and right kernels are of degree $2d+1$ and are defined over the following shifted knots

$$\boldsymbol{t}_{L,d}(x) := \Big(-\mu, \ldots, \mu-1, \underbrace{\mu, \ldots, \mu}_{d+1 \text{ times}} \Big) + x - \lambda_{L,d}, \tag{5.7}$$

$$\jmath_L := \{1 : (2d+1), 3d+1\};$$

$$\boldsymbol{t}_{R,d}(x) := \Big(\underbrace{-\mu, \ldots, -\mu}_{d+1 \text{ times}}, -\mu+1, \ldots, \mu, \Big) + x - \lambda_{R,d}, \tag{5.8}$$

$$\jmath_R := \{1, d : (3d+1)\}.$$

The prototype knot sequence $\boldsymbol{t}_{L,d}(\lambda_{L,d})$ is chosen to have symmetric support about the origin.

PROPOSITION 5.2. *Let $\mathcal{M}_d$ be defined as in Proposition 5.1 and let $\mathcal{M}_d^{(1)}$ denote its first column. Then*

$$G_{\lambda_{L,d}} = \begin{bmatrix} (d+1)\mathcal{M}_d^{(1)} & \mathcal{M}_d\mathbf{b}_0 & 0 & \cdots & 0 \\ 0 & \mathcal{M}_d\mathbf{b}_1 & \mathcal{M}_d\mathbf{b}_0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ 0 & \mathcal{M}_d\mathbf{b}_d & \mathcal{M}_d\mathbf{b}_{d-1} & \cdots & 0 \\ 0 & 0 & \mathcal{M}_d\mathbf{b}_d & \cdots & 0 \\ 0 & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ 0 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 0 & \cdots & \mathcal{M}_d\mathbf{b}_0 \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ 0 & 0 & 0 & \cdots & \mathcal{M}_d\mathbf{b}_d \end{bmatrix} \tag{5.9}$$

G_{λ_R} is obtained from G_{λ_L} by reversing the order of the columns and of the rows.

Proof. The proof is the same as that of Proposition 5.1 except that the additional B-spline of the kernel contributes the first column of (5.9). ∎

Another use of the explicit polynomial form of PSIAC coefficients is to determine the smoothness of the filtered output across the transition between the boundary region, where boundary filters are applied, and the interior region, where the symmetric filter is applied. The RLKV filter of degree $d = 1$ consists of the symmetric SIAC filter plus a B-spline whose position-dependent coefficient can be written out explicitly. The Taylor expansion of this coefficient at $h\lambda$ is

$$\frac{15}{24}(x - h\lambda) - \frac{10}{24}(x - h\lambda)^3. \tag{5.10}$$

Consequently, the only term that distinguishes the RLKV filter from the symmetric filter vanishes, to first order, at the point of transition between the symmetric SIAC kernel and the RLKV kernel. Since the next term $\frac{15}{24} \neq 0$, the kernels join exactly C^0.

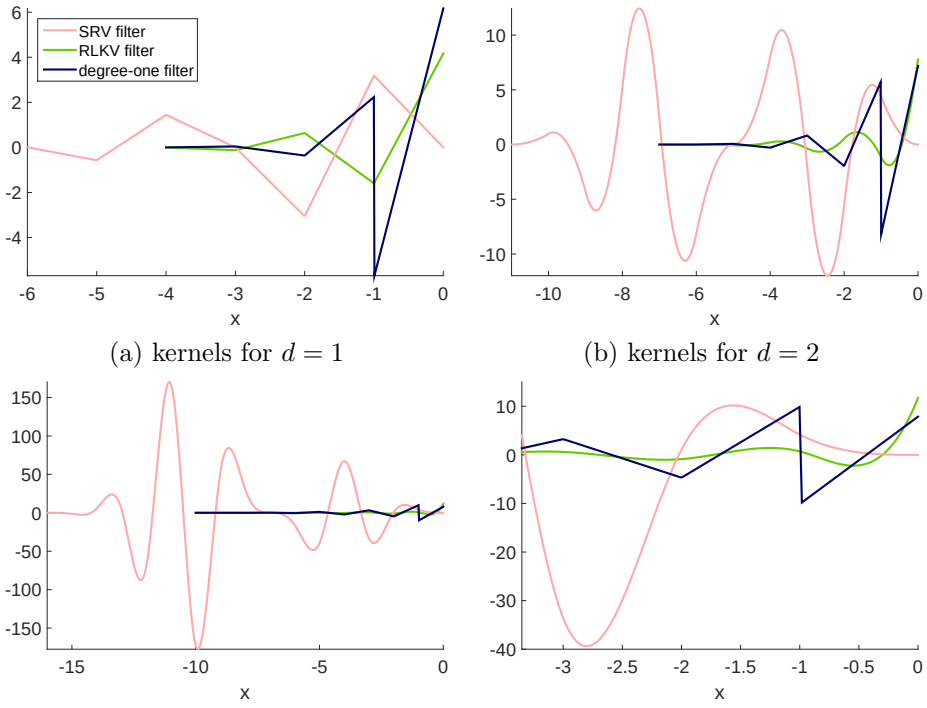
5.2. A multiple-knot boundary PSIAC kernel. Even though [CLSS03] and [MRK15] discuss varying the number of kernel coefficients $r+1$ and the degree k of symmetric kernels, ever since their introduction in the seminal paper [RS03] boundary kernels have been given the same degree as the symmetric kernel [RLKV15] – perhaps to guarantee the same smoothness near the boundary as in domain interior. Indeed, the authors of [SRV11] numerically observed and predicted that SRV-filtered DG outputs would be as smooth as the SRV kernel. Theorem 4.2 implies not only that a PSIAC kernel need not have the same degree as the symmetric kernel, but confirms

smoothness of the output. Theorem 4.2 shows that PSIAC kernels may have multiple knots without reducing the infinite smoothness of the filtered DG output.

We define a new *multiple-knot left-sided kernel* f_L and a right-sided kernel f_R each of degree $k = 1$ (hence double-knot filters) respectively over the knot sequences

$$\begin{aligned} \boldsymbol{t}_L &:= x - \lambda_L + \big(-\mu, \ldots, \mu-3, \mu-2, \mu-1, \mu-1, \mu, \mu \big), \;\; \mu := \frac{3d+1}{2}, \\ \boldsymbol{t}_R &:= x - \lambda_R + \big(-\mu, -\mu, -\mu+1, -\mu+1, -\mu+2, -\mu+3, \ldots, \mu \big). \end{aligned} \tag{5.11}$$

Since the degree of the kernels is 1, their reproduction degree is $3d + 2$ even though they have the same support size as the symmetric SIAC kernel that one might apply in the interior of the *DG* domain. The two double-knots increase the reproduction degree but not the support.



(a) kernels for $d = 1$ (b) kernels for $d = 2$

(c) kernels for $d = 3$ (c') right zoom of (c)

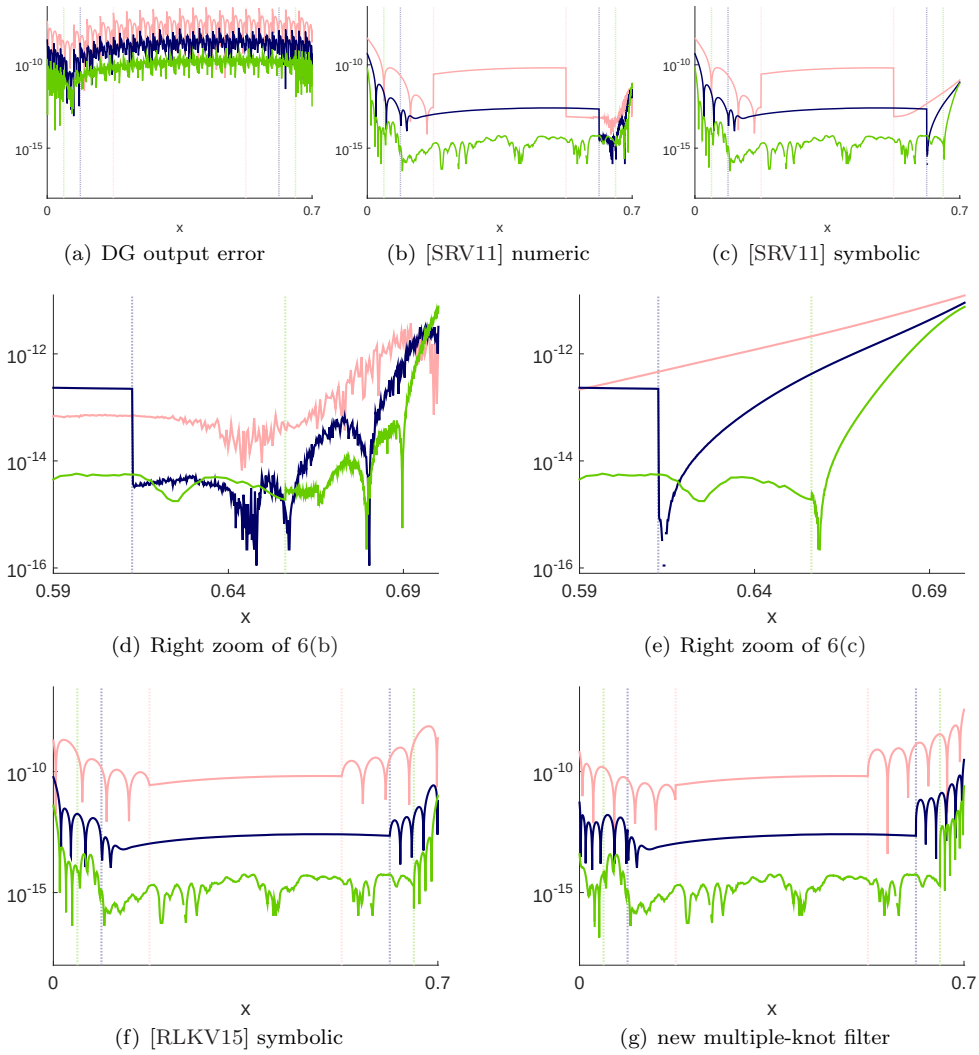
FIG. 5. *Graphs of the three kernels defined at the left boundary $x = a$. Note that the degree of [SRV11] and [RLKV15] increases with d while the degree of the multiple-knot kernel stays linear.*

5.3. Examples. Fig. 5 graphs instances of the [SRV11], [RLKV15] and of the new multiple-knot kernel of degree-one. With the help of the following two examples we numerically verify the symbolic expression (4.4). We demonstrate improved stability of the symbolic approach over the numerical approach and we illustrate a possible use of a kernel of degree $k \neq d$ (we will choose $k = 1 < d$) and illustrate a possible use of inner knots with higher multiplicity. Both test examples are special cases of the canonical Eq. (2.2), $\frac{du}{d\tau} + \frac{d}{dx}\Big(\kappa(x,\tau)\, u\Big) = \rho(x,\tau)$ for $x \in (a..b), \tau \in (0..\widetilde{\tau})$, with

$$\kappa(x,\tau) \equiv 1, \quad \rho(x,\tau) \equiv 0, \quad 0 \leq \tau \leq \widetilde{\tau}. \tag{5.12}$$

Example 1 Consider Eq. (2.2) with specialization (5.12), aperiodic Dirichlet boundary conditions and $\widetilde{\tau} := \frac{1}{16}$. The exact solution is $u(x, \widetilde{\tau}) = \frac{7}{10}\sin(\pi\sqrt{\frac{10}{7}}(x - \widetilde{\tau}))$.

Example 2 Consider Eq. (2.2) with specialization (5.12), periodic boundary conditions and $\widetilde{\tau} := 1$, i.e. after a sequence of time steps. The exact solution is $u(x, \widetilde{\tau}) = \sin(2\pi(x - \widetilde{\tau}))$.



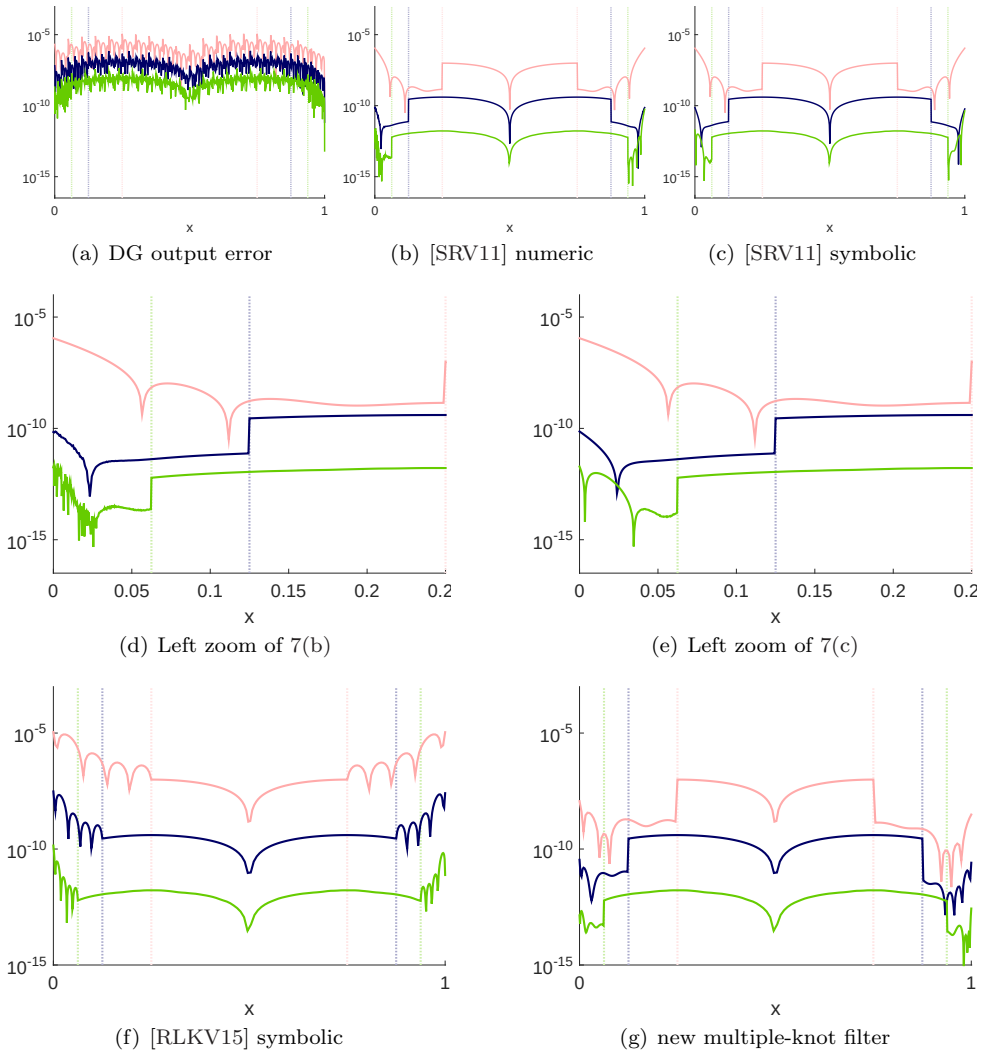
(a) DG output error (b) [SRV11] numeric (c) [SRV11] symbolic

(d) Right zoom of 6(b) (e) Right zoom of 6(c)

(f) [RLKV15] symbolic (g) new multiple-knot filter

FIG. 6. Example 1, $d = 2$, aperiodic boundary conditions*: Point-wise errors (y-axis) of DG approximations. The three graphs in each subfigure correspond, from top to bottom, (red, blue, green) to* $m = 20, 40$ *and* 80 *DG break points.*

We used upwind numerical flux [HW07] and solved the resulting ordinary differential equations with the help of a standard fourth-order four stage explicit Runge-Kutta method (ERK) [HW07, Section 3.4] and a time step small enough to attain superconvergence. Figures 6 and 7 show the error of the DG approximations of Example 1 and Example 2, respectively, when using different post-filters. Since in the interior



(a) DG output error (b) [SRV11] numeric (c) [SRV11] symbolic

(d) Left zoom of 7(b) (e) Left zoom of 7(c)

(f) [RLKV15] symbolic (g) new multiple-knot filter

FIG. 7. Example 2, $d = 3$, periodic boundary conditions*: Point-wise errors (y-axis) of DG approximations. The three graphs in each subfigure correspond, from top to bottom, (red, blue, green) to* $m = 20, 40$ *and* 80 *DG break points.*

the same symmetric SIAC filter is applied, we focus on the error near the endpoints.

For [SRV11], the numerical approach introduces high oscillations in the point-wise errors near the boundaries for both example problems. The increase of the oscillations with decreasing mesh size is the result of near-singular matrices: when $m = 80$ and $d = 3$, the matrices M have condition numbers near the limit where MATLAB declares them singular. While the numerically computed filter wildly oscillates when $m = 80$ (see the bottom-most graph of Fig. 7d), the symbolic inversion yields stable results regardless of mesh size, see Fig. 7e.

For the RLKV kernels we only show the result of the symbolic approach. The difference between the numerical approach and the symbolic approach is less than 10^{-13} confirming on one hand the stability of the RLKV filter and on the other

hand the correctness of reformulation according to Theorem 4.2. The RLKV filter is juxtaposed with our degree-one multiple-knot kernel. There is no indication that its multiple interior knot prevents superconvergence. For Example 1, displayed in Fig. 6, the multiple-knot filter shows errors on par with RLKV (a slightly higher right-sided error and a smaller left-sided one). In Example 2, Fig. 7, the multiple-knot kernel has a clearly lower error near the endpoints than both the RLKV filter and the symmetric SIAC filter.

6. Conclusion. The state-of-the-art approach for computing a filtered DG output [RS03, SRV11, MRK12, RLKV15] consists of, at each evaluation point $x \in [a..b]$, (i) assembling followed by inverting the reproduction matrix to obtain the coefficients of the position-dependent boundary kernel and then (ii) calculating the convolution integral by Gauss-Legendre quadrature to obtain the filtered DG output. Compared to that numerical approach, the PSIAC filters presented in this paper provide a more stable, flexible, versatile and efficient approach:

- ▷ *Stability*: The knots $t_{0:n}$ can be chosen freely, e.g. so that the reproduction matrix M is sufficiently regular. When the filter knots are rational, the entries of the inverse of M can be pre-computed exactly as fractions of integers. This avoids the need for repeatedly inverting near-singular constraint matrices at run-time.
- ▷ *Flexibility*: The PSIAC-filtered DG output is a single polynomial piece for the length of application of the PSIAC filter. The PSIAC kernel can be of any degree and it can be defined over a sequence of knots of any multiplicity.
- ▷ *Versatility*: The polynomial characterization of the PSIAC-filtered DG output directly yields, for example, an explicit expression of the derivatives of the filtered DG output.
- ▷ *Efficiency*: Given the prototype knot sequence and subset of B-splines on that knot sequence chosen to define a filter, the convolution matrix Q_λ can be pre-computed once and stored. Scaling columns by powers of h yields the matrix for a filter of the h-scaled knot sequence.
  Given the vector $\mathbf{u}_\mathcal{I}$ of coefficients of the DG output, the vector of polynomial coefficients $\mathbf{u}_\mathcal{I}\, Q_\lambda$ of the filtered output (a vector of size $r+1$) can be computed per data set, for all convolution points. The convolution for a point x near the boundary then simplifies to a dot product of two vectors of size $r+1$.

Acknowledgement. We thank Xiaozhou Li for pointing out the motivations for devising the RLKV boundary filter.

REFERENCES

[BS77] J.H. Bramble and A.H. Schatz. Higher order local accuracy by averaging in the finite element method. *Math Comput*, 31(137):94–111, January 1977.

[CLSS03] B. Cockburn, M. l. Luskin, C.-W. Shu, and E. Süli. Enhanced accuracy by post-processing for finite element methods for hyperbolic equations. *Math Comput*, 72(242):577–606, 2003.

[CS66] H. B. Curry and I. J. Schoenberg. On Pólya frequency functions iv: the fundamental spline functions and their limits. *J Anal Math*, 17:71107, 1966.

[dB02] C.W. de Boor. B-spline Basics. In M. Kim G. Farin, J. Hoschek, editor, *Handbook of Computer Aided Geometric Design*. Elsevier, 2002.

[dB05] C. W. de Boor. Divided differences. *Surveys in Approximation Theory*, 1:46–69, 2005.

[ES04] A. Edelman and G. Strang. Pascal matrices. *Am Math Mon*, 111:189–197, 2004.

[HW07] J.S. Hesthaven and T. Warburton. *Nodal Discontinuous Galerkin methods: algorithms, analysis, and applications*. Springer Science & Business Media, 2007.

[LRKV16] X. Li, J.K. Ryan, R.M. Kirby, and C. Vuik. Smoothness-increasing accuracy-conserving (SIAC) filters for derivative approximations of Discontinuous Galerkin (DG) solutions over nonuniform meshes and near boundaries. *J Comput Appl Math*, 294:275 – 296, 2016.

[ML78] M.S. Mock and P.D. Lax. The computation of discontinuous solutions of linear hyperbolic equations. *Commun Pur Appl Math*, 31(4):423–430, 1978.

[MRK12] H. Mirzaee, J.K. Ryan, and R. M. Kirby. Efficient implementation of smoothness-increasing accuracy-conserving (SIAC) filters for Discontinuous Galerkin solutions. *SIAM J Sci Comput*, 52(1):85–112, 2012.

[MRK15] M. Mirzargar, J.K. Ryan, and R.M. Kirby. Smoothness-increasing accuracy-conserving (SIAC) filtering and quasi-interpolation: A unified view. *SIAM J Sci Comput*, pages 1–25, 2015.

[Pet15] J. Peters. General spline filters for Discontinuous Galerkin solutions. *Comp Math Appl*, 70(5):1046 – 1050, 2015.

[RC09] J.K. Ryan and B. Cockburn. Local derivative post-processing for the Discontinuous Galerkin method. *J Comput Phys*, 228(23):8642 – 8664, 2009.

[RLKV15] J.K. Ryan, X. Li, R.M. Kirby, and K. Vuik. One-sided position-dependent smoothness-increasing accuracy-conserving (SIAC) filtering over uniform and non-uniform meshes. *SIAM J Sci Comput*, 64(3):773–817, 2015.

[RS03] J.K. Ryan and C.-W. Shu. On a one-sided post-processing technique for the Discontinuous Galerkin methods. *Methods and Applications of Analysis*, 10(2):295–308, 2003.

[RSA05] J.K. Ryan, C.-W. Shu, and H. Atkins. Extension of a post processing technique for the Discontinuous Galerkin method for hyperbolic equations with application to an aeroacoustic problem. *SIAM J Sci Comput*, 26(3):821–843, 2005.

[SRV11] P.v. Slingerland, J.K. Ryan, and C. Vuik. Position-dependent smoothness-increasing accuracy-conserving (SIAC) filtering for improving Discontinuous Galerkin solutions. *SIAM J Sci Comput*, 33(2):802–825, 2011.

[Tho77] V. Thomée. High order local approximations to derivatives in the finite element method. *Math Comput*, 31(139):652–660, 1977.