EXPLICIT LEAST-DEGREE BOUNDARY FILTERS FOR DISCONTINUOUS GALERKIN[*]

DANG-MANH NGUYEN[†] AND JÖRG PETERS[†]

Abstract. Convolving the output of Discontinuous Galerkin (DG) computations using spline filters can improve both smoothness and accuracy of the output. At domain boundaries, these filters have to be one-sided for non-periodic boundary conditions. Recently, position-dependent smoothness-increasing accuracy-preserving (PSIAC) filters were shown to be a superset of the well-known one-sided RLKV and SRV filters. Since PSIAC filters can be formulated symbolically, PSIAC filtering amounts to forming linear products with local DG output and so offers a more stable and efficient implementation.

The paper introduces a new class of PSIAC filters NP_0 that have small support and are piecewise constant. Extensive numerical experiments for the canonical hyperbolic test equation show NP_0 filters outperform the more complex known boundary filters. NP_0 filters typically reduce the L^∞ error in the boundary region below that of the interior where optimally superconvergent symmetric filters of the same support are applied. NP_0 filtering can be implemented as forming linear combinations of the data with short rational weights. Exact derivatives of the convolved output are easy to compute.

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

AMS subject classifications. 65M12; 65D07

1. Introduction. The output of Discontinuous Galerkin (DG) computations often captures higher order moments of the true solution [ML78]. Therefore post-processing DG output by convolution with splines can improve both smoothness and accuracy [BS77, CLSS03, JvRV14]. In the interior of the domain of computation, symmetric smoothness increasing accuracy conserving (SIAC) spline filters have been demonstrated to provide optimal accuracy [CLSS03]. To accommodate non-periodic boundary data, such symmetric filters need to be complemented by one-sided filters. The point-wise error and stability, of the pioneering Ryan-Shu boundary filters [RS03] have been noticeably improved upon, during the last decade [SRV11, MRK12, RLKV15, LRKV16].

Most recently these boundary filters have been simplified and improved by replacing numerical approximation with symbolic formulas, both in the uniform symmetric case [MRK15] and in the general case [Pet15]. For general knot sequences, [NP16] introduced a factored symbolic characterization of spline filters that identified the existing boundary filters as position-dependent SIAC spline filters (PSIAC filters). While in earlier formulations, the rule for each convolved value had to be derived numerically by Gauss quadrature, PSIAC prototype filters can be precomputed and then easily scaled and shifted to fit the boundary region as required. PSIAC filtering then amounts to forming a linear combination of the local DG output with simple weights that are polynomial in the position. The coefficients of these polynomial expressions are rational numbers for rational knot sequences. Remarkably [NP16, Theorem 4.2] PSIAC-filtered DG output lies on a single polynomial over the boundary region, where the filter is deployed. RLKV-filtered and SRV-filtered DG output is therefore infinitely differentiable and derivatives of PSIAC-filtered DG output need not be approximated [Tho77, RSA05, RC09, LRKV16] but have an explicit, easily-evaluated expression.

[†]Department of Computer & Information Science & Engineering, University of Florida. (nguyendm@ufl.edu, jorg@cise.ufl.edu).

(a) Error filtered by RLKV in the boundary region at final time $T = 2/3$

(b) Error filtered by NP_0 in the boundary region at final time $T = 2/3$

(c) DG error at final time $T = 2/3$

(d) Maximal errors restricted to the left $\triangleright$ and right $\triangleleft$ boundary region(new=NP_0)
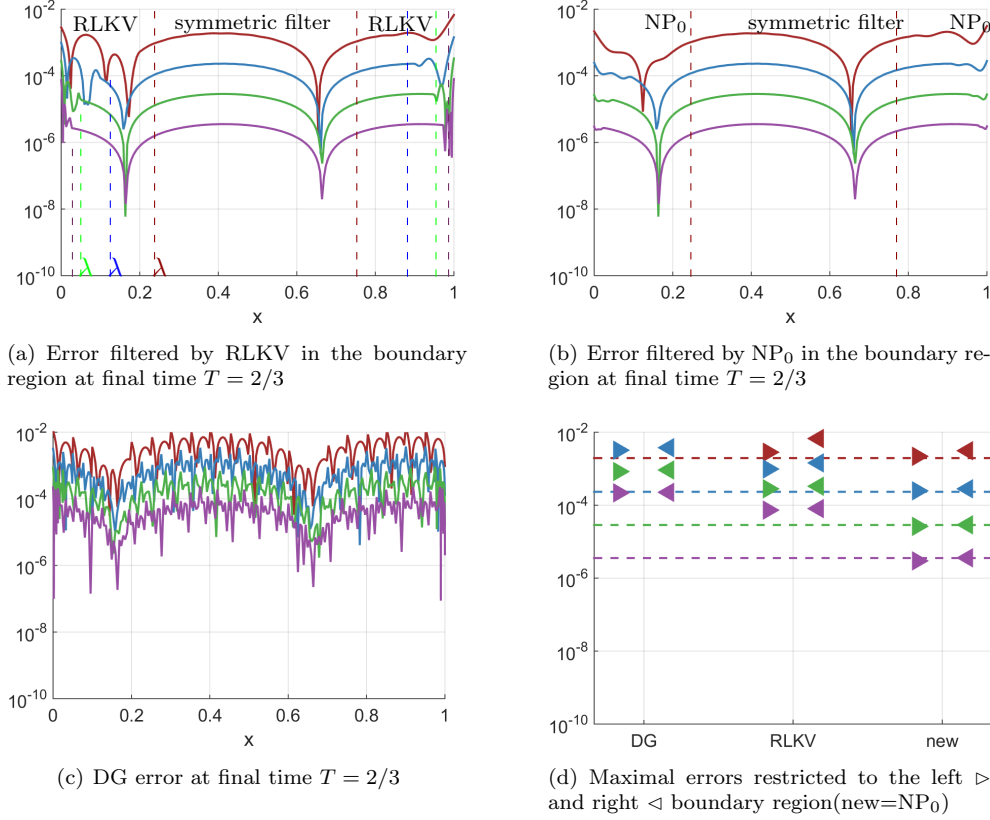
FIG. 1. *Pointwise errors $e(x, T = 2/3)$ of the filtered output (a),(b) of the DG solution of degree $d = 3$ for the canonical partial differential equation $u_\tau + u_x = 0$ computed up to fixed final time $T = 2/3$ (cf. (2.2)). The unfiltered error is shown in (c) and (d) summarizes the result. Each subfigure shows in order, from top to bottom, the graph for mesh-size $h = 20^{-1}$ (red), $h = 40^{-1}$ (blue), $h = 80^{-1}$ (green) and $h = 160^{-1}$ (purple). (a,b) The growing interior, bordered by vertical dashed lines of the color of the graph, is post-processed by the symmetric SIAC filter. The graph for the interior is therefore identical in (a) and (b). The left boundary region $[0..\lambda]$ is to the left of the left dashed line and and the right boundary region is to the right of the right line. The RLKV and NP_0 error are computed on ever smaller boundary regions $[0..\lambda]$ and $[1 - \lambda..1]$ since λ shrinks with h (see (a)). (a) Errors after convolving the data of the boundary regions with the one-sided RLKV filters [RLKV15] and the interior with the symmetric SIAC filter. This graph is identical to [RLKV15, Figure 5,top-right] but is computed with the symbolic formulation. (b) Errors after convolving the data of the boundary regions with the new least-degree PSIAC filters NP_0 and the interior with the symmetric SIAC filter. (c) Original errors of the degree $d = 3$ DG output. (d) The error in $[0..\lambda^+]$, i.e. the left boundary region augmented by the transition to the inner symmetric SIAC filter region, is indicated by $\triangleright$. The error on the right by $\triangleleft$. The error of the symmetric SIAC filter applied in the interior is indicated by $--$. The error of the new PSIAC filter NP_0 is lower or on par with the optimally superconvergent symmetric SIAC filter in the interior.*

This paper introduces a new piecewise constant PSIAC filter, that we will refer to as the NP_0 filter. On variants of the canonical hyperbolic test equation (2.2), the simple NP_0 filter outperforms, on different aspects, both the SRV and the RLKV filter. Applied to the DG output, NP_0 is more stable than the superconvergent, but larger-support SRV filter [SRV11] – SRV requires quadruple precision to produce reliable results when the DG polynomials are of degree $d > 2$ [RLKV15]; and NP_0-filtered

output converges faster than that of the small-support, stable but suboptimal RLKV filter [RLKV15]. Fig. 1 shows results for four different x-mesh spacings. The RLKV-error in the boundary region exceeds the error in the interior where the symmetric SIAC filter is applied; and for large mesh spacing, the pointwise RLKV error in Fig. 1a even exceeds that of the unfiltered DG error in Fig. 1c. By contrast Fig. 1b shows NP_0 consistently reducing the error in $[0..\lambda^+]$, the boundary regionaugmented by the transition region to symmetric SIAC filter, even below the error of the symmetric SIAC filter. This message is condensed in Fig. 1d. Here the maximal error of the symmetric SIAC filter of degree d in the interior is displayed as dashed lines and the error near the left and right boundary by $\triangleright$, respectively $\triangleleft$.

Let d be the polynomial degree of the DG output and k the degree of the spline filters. The theory of [JvRV14] guarantees a convergence order of $d+1+k$ for the NP_k filter. However, extensive numerical experiments, presented in detail in the Appendix B, indicate optimal super-convergence of order $2d+1$ for the NP_0 filter ($k = 0$) applied to the canonical hyperbolic equation $u_\tau + u_x = 0$ for a range of final times T of the DG computation. Yet, this paper does not aim to establish this super-convergence or sharp theoretical convergence bounds. Rather it establishes properties and practical formulas that make it easy to apply the NP_0 filter, e.g. explicit formulas for stably computing the convolution with the data based on a symbolic formula provided by [NP16]. PSIAC-filtering of DG output at a point x in the left boundary region $[0..\lambda]$ amounts to forming a linear combination of the vector of DG coefficients $\mathbf{u}_{[x]}$ whose functions have support at x:

$$(1.1) \qquad u_{\text{filtered}}(x) := \mathbf{u}_{[x]} \cdot \boldsymbol{\mathcal{V}}_x.$$

By Eq. (2.11) of Theorem 2.5 of this paper,

$$(1.2) \qquad \boldsymbol{\mathcal{V}}_x := Q_\lambda^h \boldsymbol{\lambda}_x, \qquad \boldsymbol{\lambda}_x := \left[(\lambda - x)^0 \cdots (\lambda - x)^{3d} \right]^t,$$

where Q_λ^h is a matrix with rational entries that depend only on the space of DG polynomials and on the space of filter kernels, not on x. For efficient implementation, the local DG coefficient vector is therefore multiplied with the matrix Q_λ^h in advance yielding a vector of size $3d + 1$ to be multiplied with $\boldsymbol{\lambda}_x$. The central four entries $\overline{\boldsymbol{\mathcal{V}}}_x$ of $\boldsymbol{\mathcal{V}}_x$ are its largest in absolute value. For the NP_0 and the SRV filter (in PSIAC form), $d = 3$ and $x = 0$, the four entries are respectively

$$(1.3) \qquad \overline{\boldsymbol{\mathcal{V}}}_0(\mathrm{NP}_0) = 10080^{-1} \begin{bmatrix} 70381 & 70381 & -56627 & -56627 \end{bmatrix}$$

$$\overline{\boldsymbol{\mathcal{V}}}_0(\mathrm{SRV}) = 15256200960000^{-1} \big[3549982809648204 \quad 9809076669570393$$

$$11473452075703833 \quad 6592494198365004 \big].$$

While the fractions of integers have at most 5 digits for NP_0, the fractions of the SRV filters have up to 17 digits and their absolute value is much larger: Fig. 2 shows that for $d = 3$ the alternating coefficients of the SRV filter are two orders of magnitude larger than those of the NP_0 filter.

In summary,

   $\triangleright$ applying the PSIAC boundary filter NP_0 is efficient: one multiplies the boundary DG data with the matrix Q_λ and then forms, for each x, hence $\boldsymbol{\lambda}_x$, the dot product between the two short vectors;

   $\triangleright$ numerical experiments in double floating point precision on the canonical wave equation indicate that NP_0 has a convergence rate of $2d+1$, where d is the piecewise degree of the DG output.
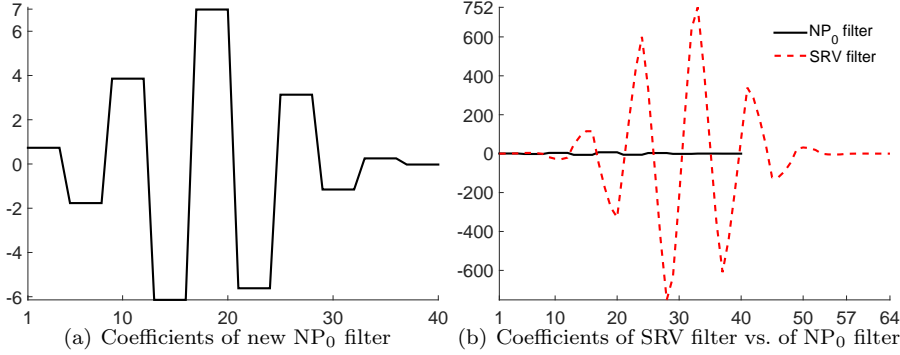
(a) Coefficients of new NP$_0$ filter   (b) Coefficients of SRV filter vs. of NP$_0$ filter

FIG. 2. *Size of the entries of* $\boldsymbol{\mathcal{V}} = Q_\lambda^{h=1}\boldsymbol{\lambda}$ *for filtering* $d = 3$ *DG output at a boundary point* $x = 0$. *Note the 100-fold difference in scale between (a) and (b).*

Organization. Section 2 introduces notation, spline convolution and the canonical test equations, followed by a review of the literature in more detail. Section 3 focuses on the new NP$_0$ filter and compares it to the existing SRV and RLKV filters in their improved symbolic PSIAC form. Equation (3.11) and Appendix A provide explicit Q_λ for degree $d \in \{1, 2, 3\}$. Section 4 compares the SRV, RLKV, NP$_0$ and the symmetric SIAC filters numerically on three variants of the canonical hyperbolic equation.

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

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 f with a kernel g.

2.1. Sequences, Splines, and Reproduction. The goal of *SIAC filtering* is to spatially smooth out the DG output $u(x, \tau)$ by convolution in x with a linear combination of B-splines. Typically filtering is applied after the last time step when $\tau = T$. Specifically, we will focus on piecewise polynomial SIAC spline kernels f : $\mathbb{R} \to \mathbb{R}$ such that convolution of f with monomials $(\cdot)^\delta$ reproduces the monomials up to degree r. Let $\mathcal{J} := (0, \ldots, j_r)$ be a sequence of strictly increasing integers between 0 and j_r, abbreviate the sequences of consecutive integers as

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

Let $B(x|t_{j:j+k+1})$ denote the unit integral B-spline with (non-decreasing) knot sequence $t_{j:j+k+1}$ (see [dB02]) related to the recursively defined B-spline $N(t\,|\,t_{i:i+k+1})$ by $N(t\,|\,t_{i:i+k+1}) = \frac{t_{i+k+1} - t_i}{k+1} B(t\,|\,t_{i:i+k+1})$. Then 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

$$(2.1) \qquad \left(\sum_{j \in \mathcal{J}} f_j B(\cdot | t_{j:j+k+1}) * (-\cdot)^\delta \right)(x) = (-x)^\delta, \qquad \delta = 0 : r.$$

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;
$\mathcal{J} := (0, \ldots, j_r)$	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.

This 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 degree-one spline filter defined over the knot sequence $\mathbf{t} := 0 : 6$ and associated with the index set $\mathcal{J} := \{0, 3, 4\}$ corresponds to $k = 1$, $n = 6$, $r = 2$ and $j_r = 4$. The two B-splines defined over the knot sequences $1 : 3$ and $2 : 4$ are skipped.*

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

$$(2.2) \qquad \frac{du}{d\tau} + \frac{d}{dx} \Big(\kappa(x, \tau) \, u \Big) = \rho(x, \tau), \qquad x \in (a..b), \tau \in (0..T)$$

$$u(x, 0) = u_0(x), \qquad x \in [a..b]$$

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

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

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

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

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

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

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

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].

2.3. A synopsis of DG filtering. Since convolution with a symmetric SIAC kernel of a function g at x requires g to be defined in a two-sided neighborhood of x, near boundaries, Ryan and Shu [RS03] proposed convolving the DG output with a 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 $(x-a)+h\big(-(3d+1), -3d, \ldots, 0\big)$ 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], Slingerland-Ryan-Vuik 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 quadruple precision calculations to determine the kernel's coefficients. ([SRV11] additionally required quadruple precision for computing the DG output.) Indeed, the coefficients of the boundary filters [RS03, SRV11, MRK12, RLKV15, MRK15] are computed by inverting a matrix whose entries are determined by Gaussian quadrature; and, as pointed out in [RLKV15], SRV filter matrices are close to singular.

Ryan-Li-Kirby-Vuik [RLKV15] therefore suggested an alternative one-sided position-dependent kernel that has the same support size as the symmetric kernel and has reproduction degree higher by one, enriching the spline space by one B-spline. This RLKV kernel is stably computed, as has been verified numerically, in double precision, up to input data degree $d = 4$ and joins the symmetric SIAC filter, applied in the interior, without a jump in error. However, the error of the RLKV kernel at the boundaries can be higher than that of the symmetric SIAC filter and the L^2 and L^∞ superconvergence rates are sub-optimal [RLKV15] (c.f. Fig. 9,10,11). [LRKV16] additionally states that RLKV has a poorer derivative approximation than SRV filters.

[NP16] reinterprets the published one-sided filters in an explicit, symbolic form as position-dependent PSIAC 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 improves their numerical stability.

2.4. Symbolic formulation of the 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 $t_\ell + t_{r+k+1-\ell} = 0$ for $\ell = 0 : \lceil (r+k+1)/2 \rceil$. 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.

LEMMA 2.1 (SIAC coefficients [NP16]). *The vector* $\mathbf{f} := [f_0, \ldots, f_r]^t \in \mathbb{R}^{r+1}$ *of B-spline coefficients of the SIAC filter with index sequence* $\mathcal{J} := (0, \ldots, j_r)$ *and knot sequence* $t_{0:n}$ *is*

$$(2.6) \qquad \mathbf{f} := \textit{first column of } M^{-1}, \quad M := M_{t_{0:n}, \mathcal{J}} = \left[\sum_{|\boldsymbol{\omega}| = \delta} t_{j:j+k+1}^{\boldsymbol{\omega}} \right]_{\delta = 0:r, \, j \in \mathcal{J}}$$

$$\textit{where} \quad t_{0:p}^{\boldsymbol{\omega}} := t_0^{\omega_0} \ldots t_p^{\omega_p} \quad \textit{and} \quad |\boldsymbol{\omega}| := \sum_{j=0}^{n} |\omega_j|$$

This characterization yields the following formula for the filter coefficients.

THEOREM 2.2 (Scaled and shifted SIAC coefficients are polynomial [NP16]). *The SIAC filter coefficients* $f_{\xi;\ell}$ *associated with the knot sequence* $h t_{0:n} + \xi$ *are polynomials of degree* r *in* ξ:

$$(2.7) \qquad \mathbf{f}_\xi := [f_{\xi;\ell}]_{\ell=0:r} = M_{t_{0:n}, \mathcal{J}}^{-1} \, \text{diag} \left(\left[(-1)^\ell \binom{\ell+k+1}{\ell} \right]_{\ell=0:r} \right) \left[\left(\frac{\xi}{h} \right)^{0:r} \right]^t.$$

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

COROLLARY 2.3 (Coefficient polynomials $f_{\xi,\ell}$ have rational coefficients [NP16]). *If the knots* $t_{0:n}$ *are rational, then the filter coefficients* $f_{\xi,\ell}$ *are polynomials in* ξ *and* h *with rational coefficients.*

We can now define the PSIAC kernel.

DEFINITION 2.4 (PSIAC kernel). *A PSIAC kernel with index sequence* $\mathcal{J} = (0, \ldots, j_r)$ *and knot sequence* $h t_{0:n} + x$ *has the form*

$$(2.8) \qquad f_x(s) := \sum_{j \in \mathcal{J}} f_{x;j} B(s \,|\, h t_{j:j+k+1} + x), \quad s \in h[t_0, t_n] + x.$$

The DG output is convolved with a PSIAC kernel $f_{x-h\lambda}(s)$ of reproduction degree r, associated with an index sequence $\mathcal{J}$ and defined over shifted knots $h t_{0:n} + x - h\lambda$ – where the constant $h\lambda$ adjusts the filter kernel to the left or right boundary. Example 2.2 illustrates the simple explicit form of the PSIAC coefficients according to Theorem 2.2 and verifies that the corresponding PSIAC filter reproduces as predicted by the derivation.

EXAMPLE 2.2 (Reproduction by PSIAC filtering). *Let* $h = 1$, $k = 0$ *and* f_x *be the least degree PSIAC filter with* $t_{0:n} = \{-2, -1, 0\}$, $r = 1$, $\mathcal{J} = (0, 1)$. *According to* (2.8) *of Definition 2.4:*

$$(2.9) \quad f_x(s) := f_{x;0} B(s \,|\, \{-2, -1\} + x) + f_{x;1} B(s \,|\, \{-1, 0\} + x)$$

$$= f_{x;0} \, \chi_{[-2,-1]+x} + f_{x;1} \, \chi_{[-1,0]+x}, \qquad \chi_{[\alpha,\beta]}(s) := \begin{cases} 1, & \textit{if } s \in [\alpha, \beta]; \\ 0, & \textit{else} \end{cases}$$

where $\chi_{[\alpha,\beta]}$ denotes the indicator function of the domain $[\alpha,\beta]$. Equation (2.7) of Theorem 2.2 provides the formula

$$(2.10) \qquad \begin{bmatrix} f_{x;0} \\ f_{x;1} \end{bmatrix} := \begin{bmatrix} 1 & 1 \\ -2-1 & -1+0 \end{bmatrix}^{-1} \begin{bmatrix} 1 & 0 \\ 0 & -2 \end{bmatrix} \begin{bmatrix} 1 \\ x \end{bmatrix} = \frac{1}{2} \begin{bmatrix} 2x-1 \\ 3-2x \end{bmatrix}.$$

This choice of filter coefficients $f_{x;0}$ and $f_{x;1}$ satisfies Eq. (2.1). For $\delta = 0$:

$$(f_x * 1)(x) = \int_{\mathbb{R}} f_x(s)\mathrm{d}s = f_{x;0} \int_{\mathbb{R}} \chi_{[-2,-1]+x}(s)\,\mathrm{d}s + f_{x;1} \int_{\mathbb{R}} \chi_{[-1,0]+x}(s)\,\mathrm{d}s = 1.$$

For $\delta = 1$: $\quad (f_x * (-\cdot))(x) = \int_{\mathbb{R}} f_x(s)\,(s-x)\,\mathrm{d}s$

$$= f_{x;0} \int_{\mathbb{R}} \chi_{[-2,-1]+x}(s)\,(s-x)\,\mathrm{d}s + f_{x;1} \int_{\mathbb{R}} \chi_{[-1,0]+x}(s)\,(s-x)\,\mathrm{d}s$$

$$= f_{x;0} \int_{-2+x}^{-1+x} (s-x)\mathrm{d}s + f_{x;1} \int_{-1+x}^{0+x} (s-x)\mathrm{d}s$$

$$= f_{x;0} \left(\frac{s^2}{2}\Big|_{s=-2+x}^{s=-1+x} - x \right) + f_{x;1} \left(\frac{s^2}{2}\Big|_{s=-1+x}^{s=0+x} - x \right)$$

$$= f_{x;0}(-f_{x;1} - x) + f_{x;1}(f_{x;0} - x) = -x.$$

Leveraging Theorem 2.2, we can efficiently compute the convolution as follows.

THEOREM 2.5 (Efficient PSIAC filtering of DG output [NP16]). Let $f_x(s)$ be a PSIAC kernel of reproduction degree r with index sequence $\mathcal{J} = (0,\ldots,j_r)$ 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:

$$(2.11) \qquad (u * f_x)(x) = \mathbf{u}_{\mathcal{I}}\, Q_\lambda \left[\left(\frac{x}{h} - \lambda \right)^{0:r} \right]^t.$$

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

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

$$(2.12) \qquad G_\lambda := \left[\int_{\lambda-t_n}^{\lambda-t_0} \phi_i(s\,;\,s_{0:m})\, B(s\,|\,\lambda - t_{n-j:j_r-j})\,\mathrm{d}s \right]_{i \in \mathcal{I},\, j \in \mathcal{J}}.$$

A is the reversal matrix with 1 on the antidiagonal and zero else.

The factored representation implies that instead of recomputing the filter coefficients afresh for each point x of the convolved output as was the practice prior to [NP16], we simply pre-compute the coefficients corresponding to one prototype knot sequence $\mathbf{t}$ and, at runtime, pre-multiply with the data and post-multiply with the vector of shifted monomials scaled by h according to Eq. (2.11).

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 2.5 shows that when the PSIAC knots are shifted along evaluation points x then PSIAC convolution yields a polynomial, i.e. the

representation near the boundary is infinitely smooth regardless of the knot multiplicity. *That is, we may view position-dependent filtering as a form of polynomial approximation.* For example, the RLKV-filtered output is a single polynomial over the boundary region where it applies.

EXAMPLE 2.3 (Coefficients of the RLKV-filtered DG output polynomial). *Let $d = k = 3$. Consider the canonical partial differential equation $u_\tau + u_x = 0$ of (2.2) for $x \in [0,1]$ at final time $T = 1$ for mesh-sizes $h_i := 2^{-i}/10$, $i = 1, 2, 3$. The analytical solution of this equation is $u_e(x) = \sin 2\pi(x - T)$. Let $\lambda_{L,i} := 5h_i$, $\lambda_{R,i} := 1 - 5h_i$ and $f_{L,i,x}$ be the RLKV filters with respect to the left boundary regions. Applying the RLKV-filter to the DG output u_{h_i} computed for mesh size h_i yields a polynomial in x as predicted by Theorem 2.5:*

$$(2.13) \qquad P_{L,i}(x) := (f_{L,i,x} * u_{h_i})(x) = \sum_{k=0}^{7} a_{k,i}(x - \lambda_{L,i})^k, \qquad x \in [0, \lambda_{L,i}],$$

where

$$\begin{bmatrix} a_{0,1} & \cdots & a_{3,1} \\ a_{4,1} & \cdots & a_{7,1} \end{bmatrix} = \begin{bmatrix} 0.999999901374753 & 0.000021494468508 & -19.738996791744032 & -0.008053345774016 \\ 64.88630724285224 & 0.623484670536888 & -82.011046997856752 & -11.875898409510508 \end{bmatrix}$$

$$\begin{bmatrix} a_{0,2} & \cdots & a_{3,2} \\ a_{4,2} & \cdots & a_{7,2} \end{bmatrix} = \begin{bmatrix} 0.707106780904271 & 4.442883051171333 & -13.95772600673645 & -29.233165736223864 \\ 45.9166940244231 & 57.754323004960845 & -59.8089096658096 & -57.790782092166637 \end{bmatrix}$$

$$\begin{bmatrix} a_{0,3} & \cdots & a_{3,3} \\ a_{4,3} & \cdots & a_{7,3} \end{bmatrix} = \begin{bmatrix} 0.382683432364482 & 5.804906304724222 & -7.553868156289703 & -38.194755175826380 \\ 24.85114895244633 & 75.39658368210462 & -32.618972790201198 & -71.729391408995241 \end{bmatrix}.$$

Analogously, by symmetry, the filtered data of the right boundary region is

$$P_{R,i}(x) := (f_{R,i,x} * u_{h_i})(x) = \sum_{k=0}^{7} (-1)^{k+1} a_{k,i}(x - \lambda_{R,i})^k, \quad x \in [\lambda_{R,i}, 1].$$

Fig. 3a plots the polynomials $P_{L,i}$ and $P_{R,i}$, the difference between the polynomials and the exact solution (note the scale 10^{-5} in Fig. 3b) and the error in log scale Fig. 3c. Fig. 3c matches the error graphs of [RLKV15, Figure 5,top-right] that were pointwise computed numerically.

The polynomial characterization directly provides a symbolic expression for the derivatives of the convolved DG output.

COROLLARY 2.6 (Derivatives of PSIAC-filtered DG output [NP16]).

$$(2.14) \qquad \frac{d^\ell}{dx^\ell}\big(u * f_x\big)(x) = \mathbf{u}_\mathcal{I}\, Q_\lambda\, \mathrm{diag}(h^{-(0:r)})\, \Big(\frac{d^\ell}{dx^\ell}(x - h\lambda)^{0:r}\Big)^t.$$

2.5. Boundary filters as PSIAC filters. The symmetric knot sequence of the symmetric kernel of degree d is

$$(2.15) \qquad \mathbf{t} := h(-\mu, -\mu + 1, \ldots, \mu), \quad \mu := \frac{r + d + 1}{2}, \; r := 2d.$$

On $[a..b]$, this symmetric kernel can only be applied at evaluation points x where

$$(2.16) \qquad \lambda_{L,d} := a + \mu, \leq x \leq b - \mu =: \lambda_{R,d}.$$

The boundary SIAC kernels RS [RS03] and SRV [SRV11] of reproduction degree $r + 1$ are of degree $k = d$, the degree of the DG output. Their index sequence $\mathcal{J}$ is consecutive, and they are defined over the shifted knots

$$(2.17) \qquad \boldsymbol{t}_{*,d}(\xi) := \Big(-\mu, -\mu + 1, \ldots, \mu\Big) + \xi - \lambda_{*,d}, \quad * \in \{L, R\}$$

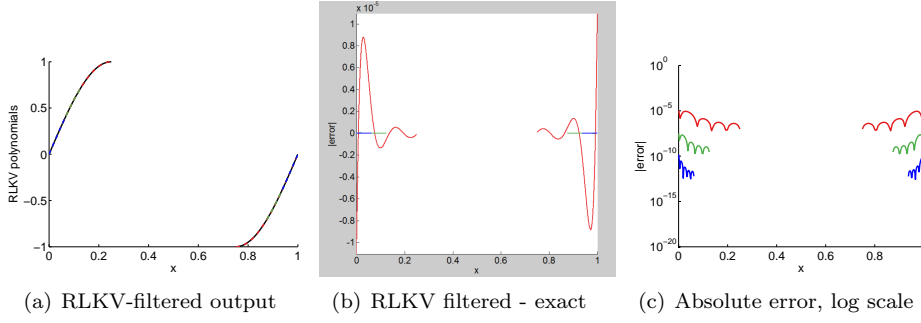(a) RLKV-filtered output   (b) RLKV filtered - exact   (c) Absolute error, log scale

FIG. 3.   *RLKV-filtered boundary region of the DG output for d=3 of Example 2.3. The graphs in red, green, and blue correspond to $i=1,2,3$ respectively, i.e. to $N=20,40,80$ DG segments. (a) Three segments of degree 7 partially cover one another, representing the RLKV-filtered output, superimposed with the exact solution (black dashed). At this resolution all four are visually indistinguishable where their domain overlaps (the x-domain of the boundary filter halves with doubling N). (b) Fine sampling of difference between the filtered output and the exact solution. (c) The canonical log scale display of the absolute error left: $|P_{L,i}(x) - u_e(x)|$, $x \in [0, \lambda_{L,i}]$ and right: $|P_{R,i}(x) - u_e(x)|$, $x \in [\lambda_{R,i}, 1]$ at computational resolution $6N$. If the sampling were as fine as (b), the arches would reach down to zero.*

that form a symmetric support about the origin when $\xi = \lambda_{*,d}$. The two kernels differ in their degree: $r(\mathrm{RS}) = 2d$ and $r(\mathrm{RV}) = 4d$. Explicit forms of the matrix G_λ, that is defined in Theorem 2.5 to efficiently construct the filter, are presented in [NP16].

The index sequence $\mathcal{J}$ of the boundary kernel RLKV [RLKV15] is non-consecutive. The left and right kernels are of degree $2d+1$ and are defined over the shifted knots, symmetric about the origin:

(2.18)
$$\boldsymbol{t}_{L,d}(\xi) := \Big(-\mu, \ldots, \mu-1, \underbrace{\mu, \ldots, \mu}_{d+1 \text{ times}} \Big) + \xi - \lambda_{L,d},$$

$$\mathcal{J}_L := \{1 : (2d+1), 3d+1\};$$

(2.19)
$$\boldsymbol{t}_{R,d}(\xi) := \Big(\underbrace{-\mu, \ldots, -\mu}_{d+1 \text{ times}}, -\mu+1, \ldots, \mu, \Big) + \xi - \lambda_{R,d},$$

$$\mathcal{J}_R := \{1, d : (3d+1)\}.$$

Explicit forms of the matrix G_λ, that is defined in Theorem 2.5 to efficiently construct the filter, are presented in [NP16].

Theorem 2.5 shows that a PSIAC filter need not have the same degree as the symmetric SIAC filter and it shows that PSIAC filters may have multiple knots without reducing the continuity of the filtered DG output. To illustrate this, [NP16] introduced filters with multiple interior knots. This class of filters is denoted NP_k and has knot sequences

$$\boldsymbol{t}_L := x - \lambda_L + \Big(-\mu, \ldots, \mu-3, \mu-2, \underbrace{\mu-1, \cdots, \mu-1}_{k+1 \text{ times}}, \underbrace{\mu, \cdots, \mu}_{k+1 \text{ times}} \Big),$$

(2.20)
$$\boldsymbol{t}_R := x - \lambda_R + \Big(\underbrace{-\mu, \cdots, -\mu}_{k+1 \text{ times}}, \underbrace{-\mu+1, \cdots, -\mu+1}_{k+1 \text{ times}}, -\mu+2, -\mu+3 \ldots, \mu \Big).$$

3. New least-degree DG filters. The symbolic formulation (2.11) applies to kernels of degree different from the degree d of the DG output. We may therefore

consider a piecewise-constant $(k = 0)$ PSIAC filter. The NP_0 filter has a consecutive index sequence $\mathcal{J}$ and is defined over the shifted knots

$$(3.1) \qquad \boldsymbol{t}_{*,d}(\xi) := \Big(-\mu, -\mu+1, \ldots, \mu \Big) + \xi - \lambda_{*,d}, \quad \mu = \frac{3d+1}{2}, \quad * \in \{L, R\}.$$

Therefore the piecewise constant NP_0 filter has reproduction degree $r+1 = 3d+1$ and the same support size as the symmetric kernel. (Numerical experiments show that a filter with $2d+1$ constant pieces still achieves optimal superconvergence – albeit with a larger error than using $3d+1$ pieces. We choose $3d+1$ since this is the number of pieces that the symmetric SIAC filter of the interior uses).

While the smoothness of the filtered output of position-*independent* filters, e.g. symmetric SIAC filters, depends on the filter degree, position-*dependent* PSIAC filters yield maximally smooth output regardless of their degree: by Theorem 2.5, the DG output filtered by a PSIAC filter is a polynomial over the respective boundary region independent of the degree or smoothness of the PSIAC filter. That is, even our piecewise constant NP_0 PSIAC filter increases the smoothness to infinity in the boundary region. Example 3.1 illustrates the remarkable fact that PSIAC filtering yields a single polynomial. This is in contrast to the finite smoothness at break points of data filtered with position-independent SIAC filters.

EXAMPLE 3.1 (PSIAC-filtering yields polynomial output). *Let $h = 1$, $\chi_{[\alpha,\beta]}$ be the indicator function of $[\alpha, \beta]$ and*

$$u_{h_0}(x) := \chi_{[0,1]}(x) + \chi_{[3,4]}(x), \ x \in [0,7] =: \Omega.$$

*the discontinuous DG output. Convolving $u_{h_0}(x)$ at x in the interior region $[2, 5]$ of Ω with the symmetric (position-*independent*) SIAC filter of reproduction degree $r = 1$,*

$$\mathcal{K}(s) := \frac{1}{2}B(s \,|\, -1 : 0) + \frac{1}{2}B(s \,|\, 0 : 1) = \frac{1}{2}\chi_{[-1,1]}(s),$$

yields

$$(3.2) \qquad (\mathcal{K} * u_{h_0})(x) = \frac{1}{2}B(s \,|\, 2 : 4) + \frac{1}{2}B(s \,|\, 3 : 5), \quad x \in [2, 5].$$

That is, convolving with $\mathcal{K}$ yields a C^0 output, as predicted by the SIAC theory developed by Ryan et al. [RS03, SRV11].

By contrast, at x in the left boundary region $[0, 2]$ of Ω, convolving $u_{h_0}(x)$ with the left-sided least-degree position-dependent PSIAC filter f_x defined in Example 2.2 yields

$$\text{for } x \in [0,2]: \quad (f_x * u_{h_0})(x) = \int_{\mathbb{R}} f_x(s)u_{h_0}(x-s)\mathrm{d}s$$

$$\xupequal{\text{by Eq. (2.9)}} f_{x;0} \int_{-2+x}^{-1+x} u_{h_0}(x-s)\mathrm{d}s + f_{x;1}\int_{-1+x}^{0+x} u_{h_0}(x-s)\mathrm{d}s$$

$$(3.3) \qquad \xupequal{0 \le x - s \le 2} f_{x;0}\int_{-2+x}^{-1+x}\chi_{[0,1]}(x-s)\mathrm{d}s + f_{x;1}\int_{-1+x}^{0+x}\chi_{[0,1]}(x-s)\mathrm{d}s$$

$$\xupequal{\text{change: } t = x-s} f_{x;0}\int_1^2 \chi_{[0,1]}(t)\mathrm{d}t + f_{x;1}\int_0^1 \chi_{[0,1]}(t)\mathrm{d}t$$

$$= f_{x;1} \xupequal{\text{by Eq. (2.10)}} \frac{1}{2}(3 - 2x).$$

*The equality Eq. (3.3) holds because when $0 \leq x \leq 2$ and $-2 + x \leq s \leq 0 + x$
then $0 \leq x - s \leq 2$. Hence $u_{h_0}(x - s) = \chi_{[0,1]}(x - s)$. As Theorem 2.5 predicts
$(f_x * u_{h_0})(x) = \frac{1}{2}(3 - 2x)$, $x \in [0, 2]$, is a polynomial over the boundary region $[0, 2]$.*

Fig. 4 graphs instances of the SRV, RLKV and NP_0 kernels. Note that the NP_0
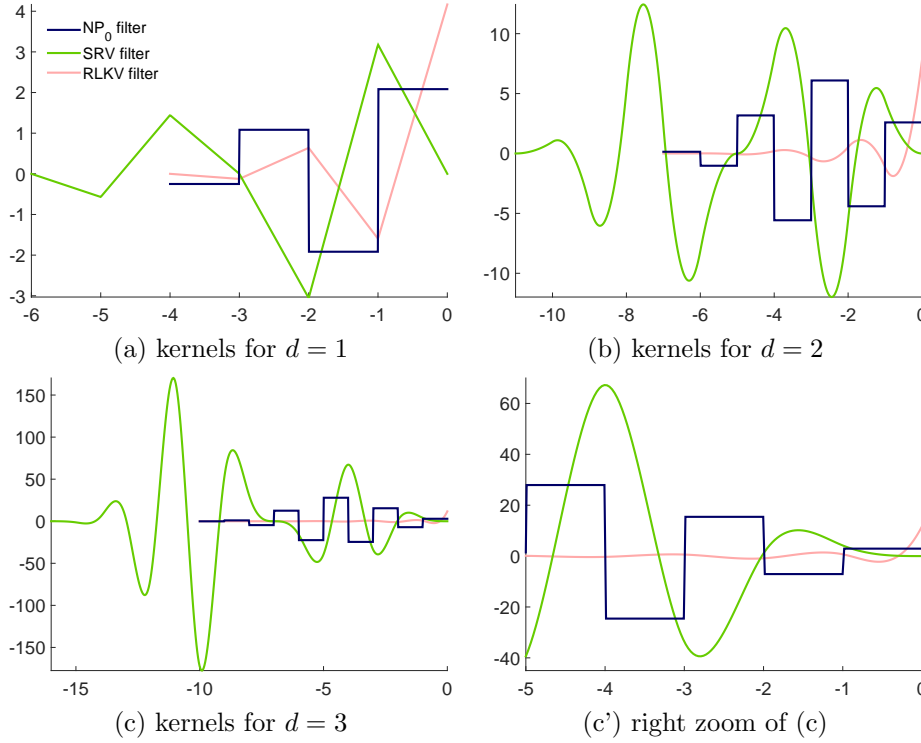filter remains piecewise constant, while the degree of the other two filters increases
with the degree d of the DG data.

(a) kernels for $d = 1$

(b) kernels for $d = 2$

(c) kernels for $d = 3$

(c') right zoom of (c)

FIG. 4. *Graphs of the three kernels defined at the left boundary $x = a$. By construction the
degree of [SRV11] and [RLKV15] increases with d while the degree of the new NP_0 kernel remains
piecewise constant. The NP_0 kernel (blue) has the same support as the RLKV kernel (red), shorter
than the SRV kernel (green).*

3.1. Symbolic form. We reduce the convolution of the DG data with the NP_0
kernels to an inner product of two short vectors. The inverse of the SIAC reproduction
matrix $M_{\mathbf{t}, \mathcal{J}}$ and the matrix G_{λ_*}, of the formulation (2.11) in Theorem 2.5, are
explicitly derived for NP_0 based on the following two propositions.

PROPOSITION 3.1. *(SIAC reproduction matrix for degree $k = 0$) The SIAC
reproduction matrix for least degree filters with index sequence $\mathcal{J} := (0, \ldots, j_r)$ and
knot sequence $t_{0:n}$ is*

$$(3.4) \qquad M_{t_{0:n}, \mathcal{J}} = \left[\frac{t_{j+1}^{\delta+1} - t_j^{\delta+1}}{t_{j+1} - t_j} \right]_{\delta = 0:r, \, j \in \mathcal{J}}.$$

If the knot sequence $t_{0:n}$ is uniform, i.e. $h_1 := t_{j+1} - t_j$, then

$$(3.5) \qquad M_{t_{0:n}, \mathcal{J}} = h_1^{-1} \left[(t_j + h_1)^{\delta+1} - t_j^{\delta+1} \right]_{\delta = 0:r, \, j \in \mathcal{J}}.$$

Proof. Since $k = 0$, each entry of $M_{t_{0:n}, \mathcal{J}}$ given by Eq. (2.6) is the sum of all monomials in t_{j+1} and t_j of total degree δ, and hence of the form (3.4). Eq. (3.5) is a direct consequence of Eq. (3.4). □

PROPOSITION 3.2 (G_{λ_*} for degree $k = 0$ and uniform DG intervals). *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.*

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

Let I be the $3d + 1$ identity matrix and $\mathbf{1}$ the $(d + 1)$ column vector of ones. The matrix G_{λ_L} for the left-sided kernel and the matrix G_{λ_R} for the right-sided kernel defined by Eq. (2.11) are
(3.7)

$$
G_{\lambda_L} = G_{\lambda_R} = I \otimes \mathbf{1} = \frac{1}{d+1} \begin{bmatrix} 1 & \cdots & 1 & 0 & \cdots & 0 & \cdots & 0 & \cdots & 0 & 0 & \cdots & 0 \\ 0 & \cdots & 0 & 1 & \cdots & 1 & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & 0 & \cdots & 0 & \cdots & 0 & 1 & \cdots & 1 \end{bmatrix}^t \in \mathbb{N}^{(d+1)(3d+1)} \times \mathbb{N}^{(3d+1)}.
$$

Proof.   First we derive Eq. (3.7) for G_{λ_L}. In Eq. (2.12), we change to the variable $t = s - \lambda_L + t_n$. Since the B-splines are translation invariant and $k = 0$ (hence $n = j_r + 1$)

$$
(3.8) \qquad B(s \mid \lambda - t_{n-j:j_r-j}) = B(t \mid t_n - t_{n-j:n-j-1}) = B(t \mid j : j + 1).
$$

Consequently, Eq. (2.12) can be rewritten as

$$
(3.9) \qquad G_{\lambda_L}(i, j) = \int_0^{3d+1} \phi_i(t \,;\, s_{0:m} - \lambda_L + t_n) \, B(t \mid j : j + 1) \, \mathrm{d}t.
$$

Since $s_0 = \frac{a}{h}$, the one-sided condition $\lambda_L = t_n + \frac{a}{h}$ 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 the B-splines $B(t \mid j : j+1)$ are supported over $[0, 3d+1]$, the relevant DG break points are $0 : 3d+1$.

We re-write the basis functions $\phi_j(s \,;\, s_{0:m} - \lambda_L + t_n)$, that are supported on an interval $[i..i+1]$, in terms of DG output Bernstein-Bézier basis functions B^i_ℓ, $\ell = 0..d$. Since each B-spline $B(t \mid j : j + 1)$ is supported on $[j, j + 1]$ and each B^i_ℓ is supported on $[i, i + 1]$, the entries of G_{λ_L} are non-trivial only if $i = j$. Therefore

$$
(3.10) \qquad G_{\lambda_L}\big((d+1)i + \ell, i\big) = \int_i^{i+1} B^i_\ell(t) \, B(t \mid i : i + 1) \, \mathrm{d}t = \frac{1}{h} \times \frac{h}{d+1} = \frac{1}{d+1}.
$$

The second last equality in Eq. (3.10) holds since $B(t \mid i : i + 1)|_{[i,i+1]} \equiv \frac{1}{h}$ and the integral of $B^i_\ell(t)$ over $[i, i + 1]$ equals $\frac{h}{d+1}$ [dB02]. Eq. (3.10) shares the entries of Eq. (3.7) for G_{λ_L}.

A similar argument for deriving G_{λ_R} but starting with the substitution $t := -(s - \lambda_R + t_0)$ proves Eq. (3.7) for G_{λ_R}. □

We note that $G_{\lambda_L} = G_{\lambda_R}$ uniquely holds for the piecewise constant one-sided filter and not for any of the higher-degree filters in the literature.

Formula (3.7) of Proposition 3.2 makes it easy to explicitly list the matrix $Q_\lambda^{d,h}$ for computing the filtered DG output $\mathbf{u}_{[x]} \cdot Q_\lambda^{d,h} \boldsymbol{\lambda}_x$ for meshsize h and DG-degree d at x (cf. Eq. (1.1)). For $d = 1$,

$$(3.11) \qquad Q_\lambda^{1,h} = Q_\lambda^{1,1} \operatorname{diag}([1, h^{-1}, \ldots, h^{-r}]), \qquad Q_\lambda^{1,1} = \frac{1}{24} \begin{bmatrix} -1 & 1 & 3 & -2 \\ -1 & 1 & 3 & -2 \\ 7 & -15 & -3 & 6 \\ 7 & -15 & -3 & 6 \\ 7 & 15 & -3 & -6 \\ 7 & 15 & -3 & -6 \\ -1 & -1 & 3 & 2 \\ -1 & -1 & 3 & 2 \end{bmatrix}.$$

The Appendix A lists $Q_\lambda^{d,1}$ for $d = 2$ and $d = 3$.

3.2. Filter transition. Let $\mathcal{L}_x * u_h$ denote the DG output u_h filtered by the left boundary PSIAC filter $\mathcal{L}_x$ and $\mathcal{K} * u_h$ the DG output filtered by the symmetric SIAC filter $\mathcal{K}$ of the interior. These two filtered outputs overlap on the interval $[a_1, a_2] = a + c_L + [0..2h]$ where c_L separates the interior and left boundary. Without loss of generality, after substituting $z := (x - a_1)/(a_2 - a_1)$, we may assume that $a_1 := 0$, $a_2 := 1$. [SRV11] suggests a smoothness-preserving transition filtering scheme that we state succinctly as

$$(3.12) \qquad u_h^\star(x) := (1 - \alpha(x)) \left(\mathcal{L}_x * u_h\right)(x) + \alpha(x) \left(\mathcal{K} * u_h\right)(x),$$

$$\alpha(x) := \sum_{i=0}^{2\rho} \alpha_i B_i(x), \quad \alpha_i := \begin{cases} 0 & \text{if } i \leq \rho; \\ 1 & \text{else.} \end{cases}$$

where $B_i(x) = \binom{2\rho}{i}(1-x)^{2\rho-i}x^i$ are Bernstein-Bézier polynomials of degree 2ρ defined over the unit interval $[0..1]$. The Bernstein-Bézier representation guarantees that $u_h^\star$ Hermite-interpolates both filtered DG output up to order ρ. The degree 2ρ need not be twice the minimum degree of the boundary filter and the symmetric SIAC filter but can be chosen to further smooth out the transition. That is, one may choose $\rho = 2$ even though $d = k = d_s = 1$. The transition for $\rho = 2$ is less abrupt at 0 and 1 than for $\rho = 1$.

4. Numerical Comparison of the SRV, RLKV, NP$_0$ and the symmetric SIAC filter. This section compares numerical results on post-filtering DG output for (2.2) using the new NP$_0$ filters as well as the state-of-the-art filters, SRV [SRV11] and RLKV [RLKV15]. We implemented SRV and RLKV in their PSIAC form [NP16] and verified that the output of our RLKV implementation matches that of a numerical code provided by the authors [RLKV15]. Our measurements compare the absolute error $e(x, T)$ as a function of domain points x and final times T. DG output is computed until time T before the filters are applied. Fig. 1a.b.c showed graphs for T-slices of $e(x, T)$ for fixed $T = 0.7$ and $x \in X = [0..1]$. The graphs of the Appendix display, as functions of T, the L^p norm of the slices

$$e_X(T) := \|e(x, T)\|_{p,X}, \quad p \in \{2, \infty\}, x \in X.$$

That is, each point on the graphs succinctly represents the max-norm or L^2-norm error over the support X of the filter. Denote by $[0..\lambda^+]$ the left boundary region $[0..\lambda]$ plus the transition region. Then for each boundary filter the graphs show $e_{[0..\lambda^+]}(T)$ on the left and $e_{[1-\lambda^+..1]}(T)$ on the right and replicate in both left and

right $e_{[\lambda^+..1-\lambda^+]}(T)$ for the symmetric SIAC filter and $e_{[0..1]}(T)$ for DG. The graphs in the Appendix B summarize ca. 15,000 measurements, some of which are graphed as T-slices in the technical addendum [NPdf]. (Showing all T-slices would fill 100 pages in the compressed display format of [NPdf]).

The graphs plot errors for three instances of the canonical problem (2.2),

$$\frac{du}{d\tau} + \frac{d}{dx}\Big(\kappa(x,\tau)\,u\Big) = \rho(x,\tau) \quad x \in (a..b),\, \tau \in (0..T).$$

The filters are applied at time T for final times $T \in [0..1]$ for Test 4.1 and for final times $T \in [0..2\pi]$ for the other two Test problems.

Test 4.1 (Constant wave-speed, *periodic* boundary conditions). *Consider the specializations of Eq. (2.2):*

(4.1) $$\kappa(x,\tau) \equiv 1, \quad \rho(x,\tau) \equiv 0, \quad 0 \le \tau \le T,$$

$a := 0$, $b := 1$, *periodic boundary conditions,* $u_0(x) := \sin(2\pi x)$. *The exact solution at the final time T is* $u(x,T) = \sin(2\pi(x-T))$.

Test 4.2 (Constant wave-speed, *Dirichlet* boundary conditions). *Consider Eq. (2.2) with specializations* (4.1), *but* $a := 0$, $b := 2\pi$, *Dirichlet boundary conditions,* $u_0(x) := \sin(x)$, $u(0,\tau) := -\sin(\tau)$. *The exact solution is at the final time T is* $u(x,T) = \sin(x-T)$.

Test 4.3 (*Variable* wave-speed, periodic boundary conditions). *Consider Eq.* (2.2) *with the specializations*

(4.2) $$\kappa(x,\tau) := 2 + \sin(x+\tau), \quad \rho(x,\tau) := \cos(x-\tau) + \sin(2x), \quad 0 \le \tau \le T,$$

$a := 0$, $b := 2\pi$, *and periodic boundary conditions,* $u_0(x) := \sin(x)$. *The exact solution at the final time T is* $u(x,T) = \sin(x-T)$.

The comparison of the filters yields broadly the same qualitative results for all three test scenarios.

Fig. 6, 7 and 8 juxtapose the maximal point-wise errors $\|e(x,T)\|_{\infty,X}$ for each final time T on the abcissa and resolutions of $[0..2\pi]$ into $N = 80$ segments (dashed) and $N = 160$ segments (solid) for $d = 1, 2$ and half the N-resolution for $d = 3$. The region X for the DG output is the whole domain, the left, respectively right boundary and transition regions for the boundary filters SRV, RLKV, NP$_0$, and the interior $[\lambda^+..1-\lambda^+]$ for symmetric SIAC filter. In Fig. 5 (a copy of Fig. 6a,b) the filter error for Test 4.1 and the left boundary regions is plotted in (a) and the right boundary regions in (b). For example, for $N = 160$ and Test 4.2 solved up to final time $T = 1/3$ before filters were applied, we can read off in (a) the maximal absolute error of the left RLKV-filter as approximately 10^{-4}, less that the top grey dotted DG-error and the error of the SRV-filter, the NP$_0$ and the symmetric SIAC filter as approximately 10^{-6}. (The numbers for DG, RLKV and NP$_0$ match those of of the graphs in Section 1.1, Ex.1 of [NPdf] for fixed $T = 1/3$.).

Fig. 9, 10 and 11 display, for various final times T on the abscissa, the convergence rate

(4.3) $$\rho = \rho(T,h) := \ln\Big(\frac{e_X(T) \text{ for mesh size } 2h}{e_X(T) \text{ for mesh size } h}\Big)\Big/\ln 2.$$

for the L^2 and the L^∞ norm, for the left boundary region and the right boundary region, and for several $h = 1/N$ where N is the sampling of the spatial domain. Fig. 9
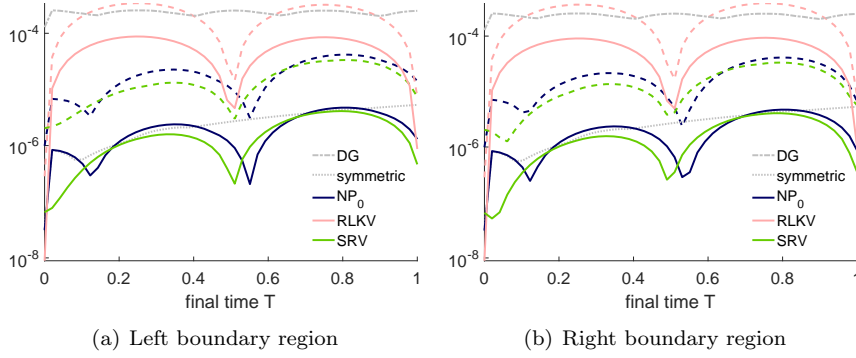
(a) Left boundary region   (b) Right boundary region

FIG. 5. **Example graphs** *from Fig. 6a,b for L^∞ error $\|e(x,T)\|_{\infty,X}$ vs final time for Test 4.1 (constant wave-speed, periodic boundary condition), $d = 1$, $T = [0..1]$. The L^∞-error of PSIAC filters RLKV (red), SRV (green), NP_0 (blue) is computed only over the respective region: $[0..\lambda^+]$ in (a) and $[1 - \lambda^+, 1]$ in (b). The top-most grey dash-dotted graphs in (a) and (b) are identical and represent $e_{[0..1]}(T)$ for the raw DG output for $N = 160$ x-segments. The dotted grey graph in the middle is $e_{[\lambda^+..1-\lambda^+]}(T)$ for symmetric SIAC filter over the interior. Graphs for boundary filter errors are dashed when the overall partition is $N = 80$ and solid for $N = 160$. Each point on the abscissa therefore represents one final time $0 \leq T \leq 1$ and each ordinate represents a maximal absolute error over the domain for that final time.*

and Fig. 10 each show #(degrees×norms×boundary regions×final times×filter types or DG output ×refinements) = $(3\times2\times2\times50\times5\times2)$ = 6000 convergence rates. Fig. 11, with 30 final times, shows 3600 convergence rates. The graphs help us spot that convergence rates are higher for some isolated T than in general.

For $d = 1$, Fig. 6 of the Appendix B shows for all final times T that both SRV and NP_0 filtering result in lower errors than RLKV-filtering. For $d = 2, 3$, Fig. 7 and for $d = 3$, Fig. 8 point to the instability of the SRV filter at double precision: for certain final times T, the SRV-error even increases above that of the raw DG output. Instability also causes the SRV-error to exceed that of the NP_0 filter in Fig. 6. For $d = 1, 2$, the maximal point-wise errors of the RLKV filter are noticeably larger than those of the SRV filter and the NP_0 filter. The point-wise errors of the NP_0 filter are on par with the symmetric SIAC filter (in gray).

We note the consistency between L^2 and L^∞ convergence and therefore clearly higher rates for SRV, NP_0 and the symmetric SIAC filters over RLKV. When $d = 1$ and $d = 2$, the NP_0 and SRV filters show optimal L^2 and L^∞ super-convergence rates of order $2d + 1$ in the boundary region. The convergence rate of 2 for the boundary regions for RLKV and $d = 1$ has been verified also with the original RLKV code. Fig. 12 in Appendix B compares the convergence orders for different end times T and DG input of degree $d = 1, 2, 3$.

For $d = 3$ the point-wise errors and convergence rates of all filters oscillate since calculations are close to machine precision. However, the point-wise errors and convergence rates of the NP_0 filter are on par with that of the symmetric SIAC filter, while the errors of the SRV filters are notably higher and convergence rates become sub-optimal, and even lower than $d + 1$ for some T when $d = 3$. Fig. 2 helps explain the instability of the SRV filters. Plotting the alternating entries of the convolution vector $\boldsymbol{\mathcal{V}} = Q_\lambda \boldsymbol{\lambda}$ at a boundary point x (e.g., $x = 0$ when $a = 0$), we find the entries of the SRV filters to be two orders of magnitude larger than those of NP_0.

5. Conclusion. PSIAC NP_0 filters enjoy three positive properties. First, explicit tabulated integer-based formulas make the computation of the NP_0 filters and their application for convolution more stable and efficient. Second, on canonical test equations, applying NP_0 filters reduces the errors of the DG output to that of optimal symmetric SIAC filters. Third, the support of the NP_0 filters matches that of the symmetric SIAC filter, making them naturally compatible.

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.

[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.

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

[JvRV14]   Liangyue Ji, Paulien van Slingerland, Jennifer Ryan, and Kees Vuik. Superconvergent error estimates for position-dependent smoothness-increasing accuracy-conserving (SIAC) post-processing of Discontinuous Galerkin solutions. *Math Comput*, 83(289):2239–2262, 2014.

[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.

[NP16]   Dang-Manh Nguyen and Jörg Peters. Nonuniform discontinuous galerkin filters via shift and scale. *SIAM Journal on Numerical Analysis*, 54(3):1401–1422, 2016.

[NPdf]   Dang-Manh Nguyen and Jörg Peters. More measurements: Effective least-degree boundary kernels for Discontinuous Galerkin, https://www.cise.ufl.edu/research/SurfLab/15lsiac.pdf.

[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.

Appendix A: Explicit filter matrix Q_λ for $\mathbf{NP}_0$. For $d = 1$, $Q_\lambda^{1,1}$ is listed in (3.11). For $d = 2$,

$$Q_\lambda^{2,1} = \frac{1}{967680} \begin{bmatrix}
-225 & -7252 & 3108 & 7840 & -2800 & -1344 & 448 \\
-225 & -7252 & 3108 & 7840 & -2800 & -1344 & 448 \\
-225 & -7252 & 3108 & 7840 & -2800 & -1344 & 448 \\
2862 & 62608 & -38808 & -58240 & 30240 & 5376 & -2688 \\
2862 & 62608 & -38808 & -58240 & 30240 & 5376 & -2688 \\
2862 & 62608 & -38808 & -58240 & 30240 & 5376 & -2688 \\
-22863 & -264740 & 288540 & 92960 & -95760 & -6720 & 6720 \\
-22863 & -264740 & 288540 & 92960 & -95760 & -6720 & 6720 \\
-22863 & -264740 & 288540 & 92960 & -95760 & -6720 & 6720 \\
363012 & 0 & -505680 & 0 & 136640 & 0 & -8960 \\
363012 & 0 & -505680 & 0 & 136640 & 0 & -8960 \\
363012 & 0 & -505680 & 0 & 136640 & 0 & -8960 \\
-22863 & 264740 & 288540 & -92960 & -95760 & 6720 & 6720 \\
-22863 & 264740 & 288540 & -92960 & -95760 & 6720 & 6720 \\
-22863 & 264740 & 288540 & -92960 & -95760 & 6720 & 6720 \\
2862 & -62608 & -38808 & 58240 & 30240 & -5376 & -2688 \\
2862 & -62608 & -38808 & 58240 & 30240 & -5376 & -2688 \\
2862 & -62608 & -38808 & 58240 & 30240 & -5376 & -2688 \\
-225 & 7252 & 3108 & -7840 & -2800 & 1344 & 448 \\
-225 & 7252 & 3108 & -7840 & -2800 & 1344 & 448 \\
-225 & 7252 & 3108 & -7840 & -2800 & 1344 & 448
\end{bmatrix}.$$

For $d = 3$, $Q_\lambda^{3,1} = \frac{1}{14515200}$

$$\begin{bmatrix}
2880 & -1152 & -12300 & 3280 & 6825 & -1638 & -1050 & 240 & 45 & -10 \\
2880 & -1152 & -12300 & 3280 & 6825 & -1638 & -1050 & 240 & 45 & -10 \\
2880 & -1152 & -12300 & 3280 & 6825 & -1638 & -1050 & 240 & 45 & -10 \\
2880 & -1152 & -12300 & 3280 & 6825 & -1638 & -1050 & 240 & 45 & -10 \\
-33120 & 16848 & 139020 & -47160 & -72975 & 22302 & 9870 & -2880 & -315 & 90 \\
-33120 & 16848 & 139020 & -47160 & -72975 & 22302 & 9870 & -2880 & -315 & 90 \\
-33120 & 16848 & 139020 & -47160 & -72975 & 22302 & 9870 & -2880 & -315 & 90 \\
-33120 & 16848 & 139020 & -47160 & -72975 & 22302 & 9870 & -2880 & -315 & 90 \\
182880 & -127152 & -737400 & 342360 & 338100 & -142128 & -33600 & 13680 & 900 & -360 \\
182880 & -127152 & -737400 & 342360 & 338100 & -142128 & -33600 & 13680 & 900 & -360 \\
182880 & -127152 & -737400 & 342360 & 338100 & -142128 & -33600 & 13680 & 900 & -360 \\
182880 & -127152 & -737400 & 342360 & 338100 & -142128 & -33600 & 13680 & 900 & -360 \\
-681120 & 736848 & 2408280 & -1754760 & -644700 & 447552 & 52080 & -35280 & -1260 & 840 \\
-681120 & 736848 & 2408280 & -1754760 & -644700 & 447552 & 52080 & -35280 & -1260 & 840 \\
-681120 & 736848 & 2408280 & -1754760 & -644700 & 447552 & 52080 & -35280 & -1260 & 840 \\
-681120 & 736848 & 2408280 & -1754760 & -644700 & 447552 & 52080 & -35280 & -1260 & 840 \\
2342880 & -5311152 & -1797600 & 3853080 & 372750 & -773388 & -27300 & 55440 & 630 & -1260 \\
2342880 & -5311152 & -1797600 & 3853080 & 372750 & -773388 & -27300 & 55440 & 630 & -1260 \\
2342880 & -5311152 & -1797600 & 3853080 & 372750 & -773388 & -27300 & 55440 & 630 & -1260 \\
2342880 & -5311152 & -1797600 & 3853080 & 372750 & -773388 & -27300 & 55440 & 630 & -1260 \\
2342880 & 5311152 & -1797600 & -3853080 & 372750 & 773388 & -27300 & -55440 & 630 & 1260 \\
2342880 & 5311152 & -1797600 & -3853080 & 372750 & 773388 & -27300 & -55440 & 630 & 1260 \\
2342880 & 5311152 & -1797600 & -3853080 & 372750 & 773388 & -27300 & -55440 & 630 & 1260 \\
2342880 & 5311152 & -1797600 & -3853080 & 372750 & 773388 & -27300 & -55440 & 630 & 1260 \\
-681120 & -736848 & 2408280 & 1754760 & -644700 & -447552 & 52080 & 35280 & -1260 & -840 \\
-681120 & -736848 & 2408280 & 1754760 & -644700 & -447552 & 52080 & 35280 & -1260 & -840 \\
-681120 & -736848 & 2408280 & 1754760 & -644700 & -447552 & 52080 & 35280 & -1260 & -840 \\
-681120 & -736848 & 2408280 & 1754760 & -644700 & -447552 & 52080 & 35280 & -1260 & -840 \\
182880 & 127152 & -737400 & -342360 & 338100 & 142128 & -33600 & -13680 & 900 & 360 \\
182880 & 127152 & -737400 & -342360 & 338100 & 142128 & -33600 & -13680 & 900 & 360 \\
182880 & 127152 & -737400 & -342360 & 338100 & 142128 & -33600 & -13680 & 900 & 360 \\
182880 & 127152 & -737400 & -342360 & 338100 & 142128 & -33600 & -13680 & 900 & 360 \\
-33120 & -16848 & 139020 & 47160 & -72975 & -22302 & 9870 & 2880 & -315 & -90 \\
-33120 & -16848 & 139020 & 47160 & -72975 & -22302 & 9870 & 2880 & -315 & -90 \\
-33120 & -16848 & 139020 & 47160 & -72975 & -22302 & 9870 & 2880 & -315 & -90 \\
-33120 & -16848 & 139020 & 47160 & -72975 & -22302 & 9870 & 2880 & -315 & -90 \\
2880 & 1152 & -12300 & -3280 & 6825 & 1638 & -1050 & -240 & 45 & 10 \\
2880 & 1152 & -12300 & -3280 & 6825 & 1638 & -1050 & -240 & 45 & 10 \\
2880 & 1152 & -12300 & -3280 & 6825 & 1638 & -1050 & -240 & 45 & 10 \\
2880 & 1152 & -12300 & -3280 & 6825 & 1638 & -1050 & -240 & 45 & 10
\end{bmatrix}.$$

Appendix B: Numerical experiments - Error Graphs and Convergence Rates. Our measurements compare the absolute error $e(x, T)$ as a function of domain points x and final times T. DG output is computed until time T before the filters are applied. Fig. 1a.b.c showed graphs for T-slices of $e(x, T)$ for fixed $T = 0.7$ and $x \in X = [0..1]$. The graphs of the Appendix display, as functions of T, the L^p norm of the slices

$$e_X(T) := \|e(x, T)\|_{p,X}, \quad p \in \{2, \infty\}, x \in X.$$

That is, each point on the graphs succinctly represents the max-norm or L^2-norm error over the support X of the filter. Denote by $[0..\lambda^+]$ the left boundary region $[0..\lambda]$ plus the transition region. Then for each boundary filter the graphs show $e_{[0..\lambda^+]}(T)$ on the left and $e_{[1-\lambda^+..1]}(T)$ on the right and replicate in both left and right $e_{[\lambda^+..1-\lambda^+]}(T)$ for the symmetric SIAC filter and $e_{[0..1]}(T)$ for DG. Fig. 9, 10 and 11 show L^∞ and L^2 convergence rates of filtered DG output plotted against the final time T.

The errors in Fig. 6, 7, and 8 are computed for DG output of degree $d = 1, 2, 3$ and for uniform partitions of the x-domain into intervals of length $h := 1/N$. A fixed number of points, depending on d but not N, in the boundary regions are the result of PSIAC-filtering with one of SRV, RLKV, NP_0. (We verified correctness of RLKV by comparison with a certified numerical RLKV implementation.) We emphasize that all boundary filters are measured only in the boundary regions inclusive of the transition region. Errors of the symmetric filter are computed only for the interior exclusive of the transition region. With increasing N the interior converges to the full domain and symmetric SIAC filter contributes an increasing number of values. The number of values generated by boundary filters does not increase with N but stays fixed. Regardless of the choice of boundary filter, the L^2 error over the full domain is dominated by the symmetric SIAC filter. Therefore errors and rates for the boundary filters are not combined with those of the symmetric filter.

The L^∞ errors in (Test 4.1, Test 4.2, Test 4.3) have been computed for $(50, 50, 30)$ different final times T in $([0, 1], [0, 2\pi], [0, 2\pi])$ respectively. (Due to scaling by 2π, the test problems have different final times T.) For example, the value of the RLKV error for Test 4.1 for $N = 80$ and DG element degree $d = 1$ (dashed red graph in Fig. 6a) at final time $T = 0.4$ is ca. 0.0002. This value is obtained by solving Test 4.1 using DG of degree $d = 1$ up to time $T = 0.4$ and then convolving with the RLKV filter to obtain $s = 6$ samples per boundary region interval for a total of $3s$ error samples. The maximum absolute sampled value is 0.0002. The error of the symmetric SIAC filter is computed from the remaining interior $N - 2 * 3s$ samples. The s samples within the intervals are uniformly distributed. Sampling instead at Gauss-Legendre points does not noticeably change the outcome.

All graphs share the same color and style assignments. In Fig. 6, 7, and 8, the raw DG output for $N = 160$ in (a–d) and $N = 80$ in (e,f) is the top-most dash-dotted graph (largest error). In Fig. 9, 10 and 11 the DG convergence-rate is at the bottom. The dotted grey graph represents symmetric SIAC filter for $N = 160$ in (a–d) and $N = 80$ in (e,f). The boundary filters are RLKV (red), SRV(green) and NP_0 filter (blue). Dashed graphs correspond to $N = 80$ in (a–d) and $N = 40$ in (e,f) and *solid graphs* to $N = 160$ in (a–d) and $N = 80$ in (e,f), halving the interval length h.

Fig. 9 for Test 4.1 and Fig. 10 for Test 4.2, and Fig. 11 for Test 4.3 display the L^∞ *and the* L^2 *convergence rates* of the filtered DG outputs. All computations are performed in double precision.

(a) Left boundary, $d = 1$

(b) Right boundary, $d = 1$

(c) Left boundary, $d = 2$

(d) Right boundary, $d = 2$

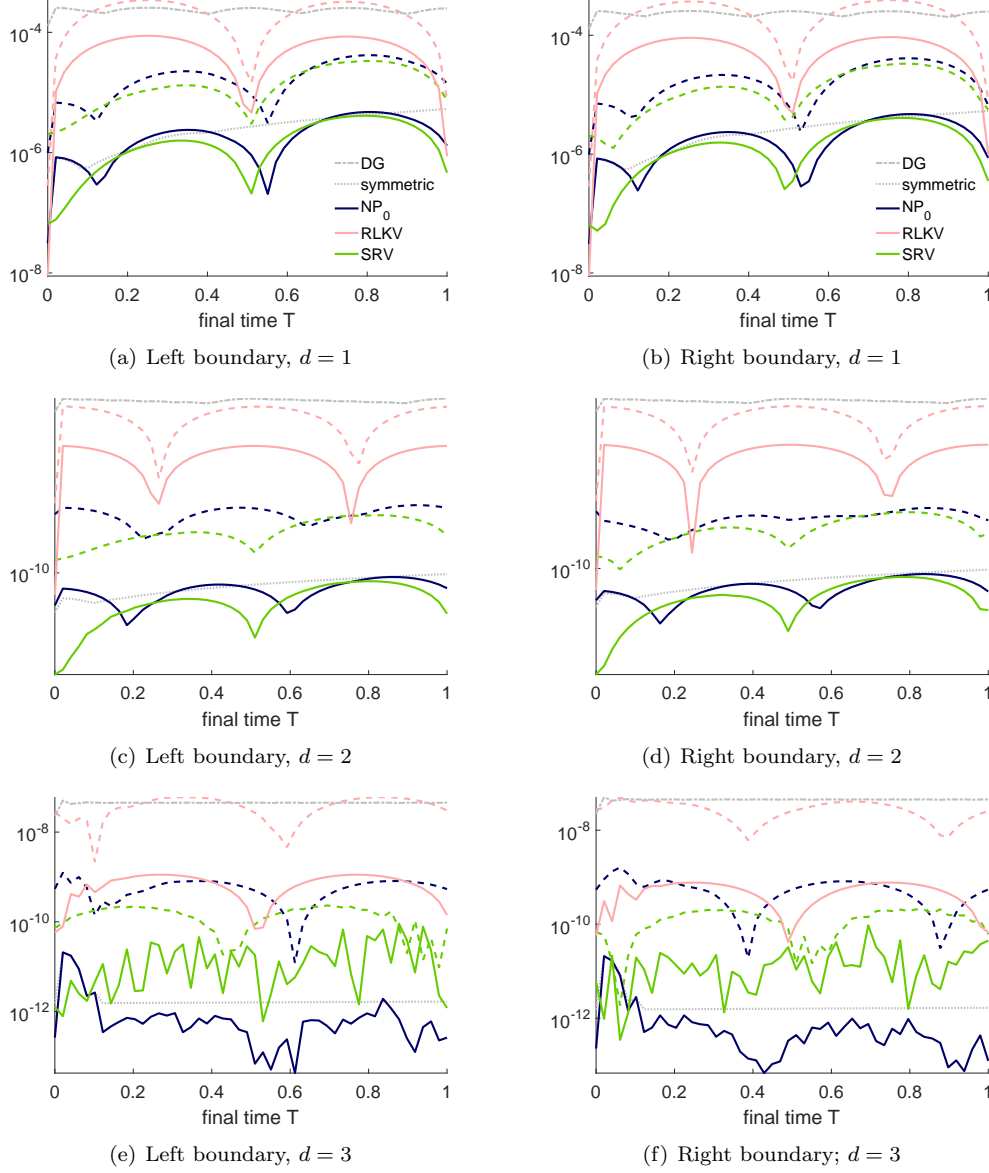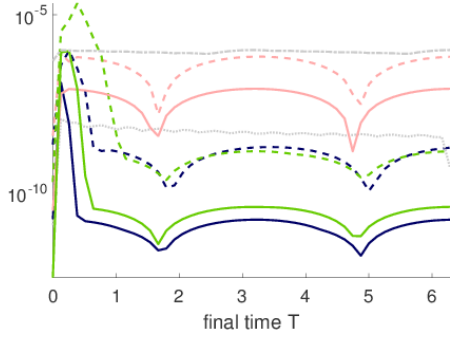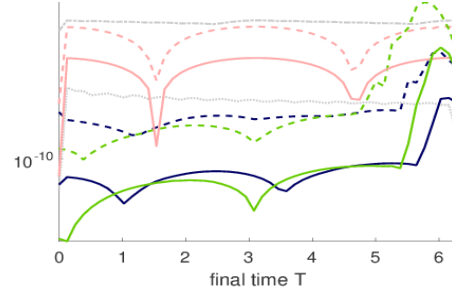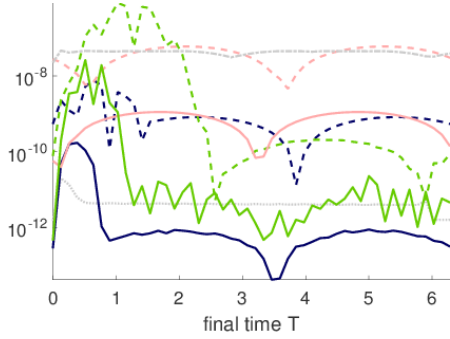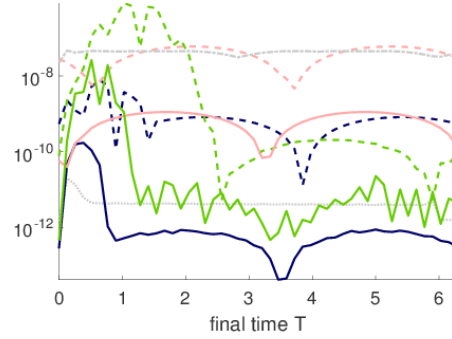(e) Left boundary, $d = 3$

(f) Right boundary; $d = 3$

FIG. 6. L^∞ *error vs final time* Test 4.1 (constant wave-speed, periodic boundary condition). Abscissa: final time T at which the DG output was computed. Ordinate: maximum point-wise errors. Red (RLKV), green (SRV), blue (NP_0) dashed graphs correspond to $N = 80$ in (a–d) and to $N = 40$ in (e,f). All other graphs correspond to $N = 160$ in (a–d) and to $N = 80$ in (e,f), with top dot-dashed grey (DG output) and lower dotted grey (symmetric SIAC filter).
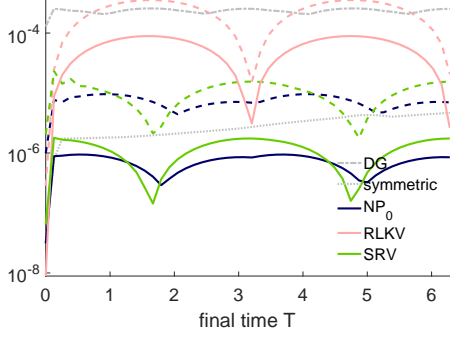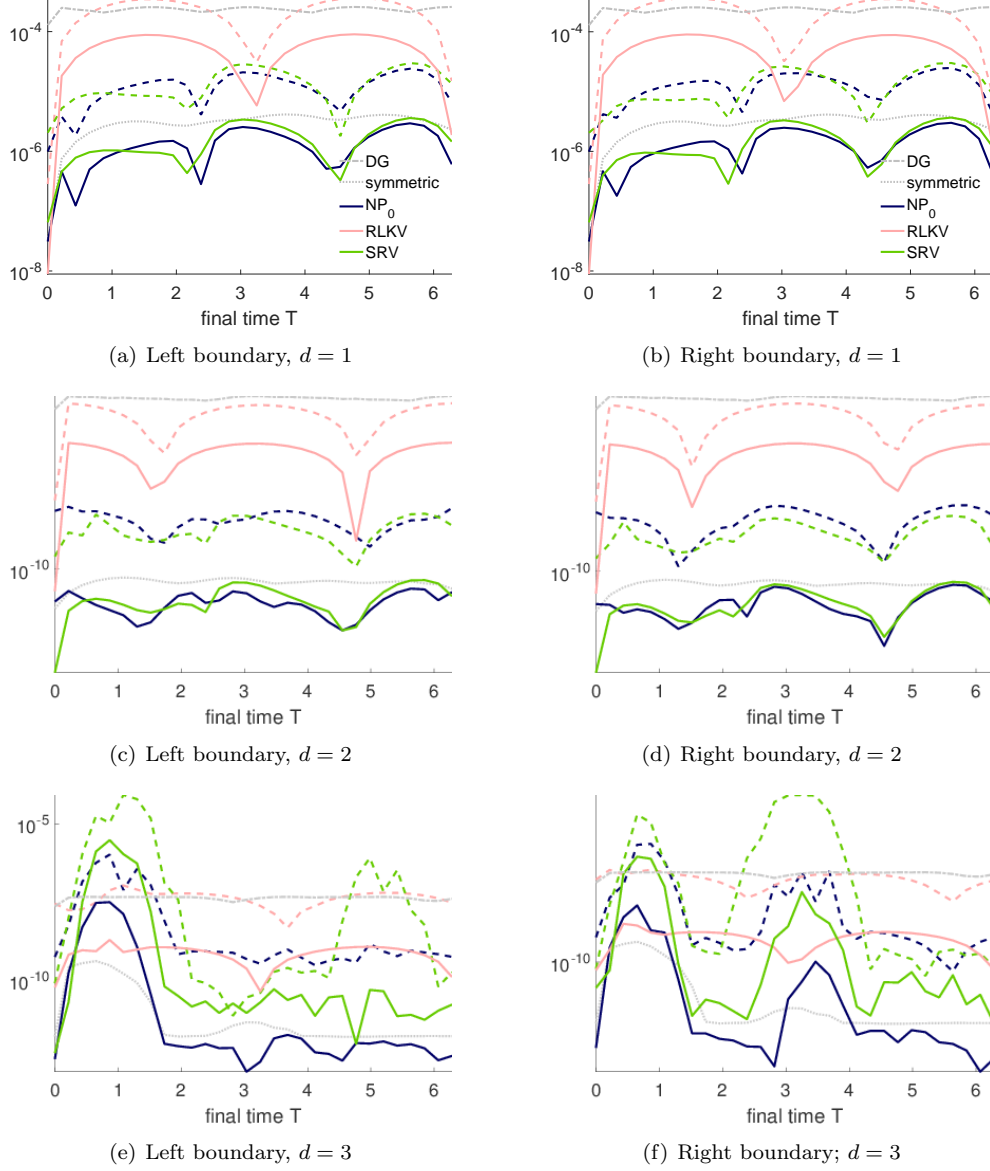
FIG. 7. L^∞ **error vs final time** Test 4.2 (constant wave-speed, Dirichlet boundary condition). Abscissa: final time T at which the DG output was computed. Ordinate: maximum point-wise errors. Red (RLKV), green (SRV), blue (NP_0) dashed graphs correspond to $N = 80$ in (a–d) and to $N = 40$ in (e,f). All other graphs correspond to $N = 160$ in (a–d) and to $N = 80$ in (e,f), with top dot-dashed grey (DG output) and lower dotted grey (symmetric SIAC filter).

(a) Left boundary, $d = 1$

(b) Right boundary, $d = 1$

(c) Left boundary, $d = 2$

(d) Right boundary, $d = 2$

(e) Left boundary, $d = 3$

(f) Right boundary; $d = 3$

Fig. 8. L^∞ **error vs final time** *Test 4.3 (variable wave-speed, periodic boundary condition). Abscissa: final time T at which the DG output was computed. Ordinate: maximum point-wise errors. Red (RLKV), green (SRV), blue (NP$_0$) dashed graphs correspond to $N = 80$ in (a–d) and to $N = 40$ in (e,f). All other graphs correspond to $N = 160$ in (a–d) and to $N = 80$ in (e,f), with top dot-dashed grey (DG output) and lower dotted grey (symmetric SIAC filter).*
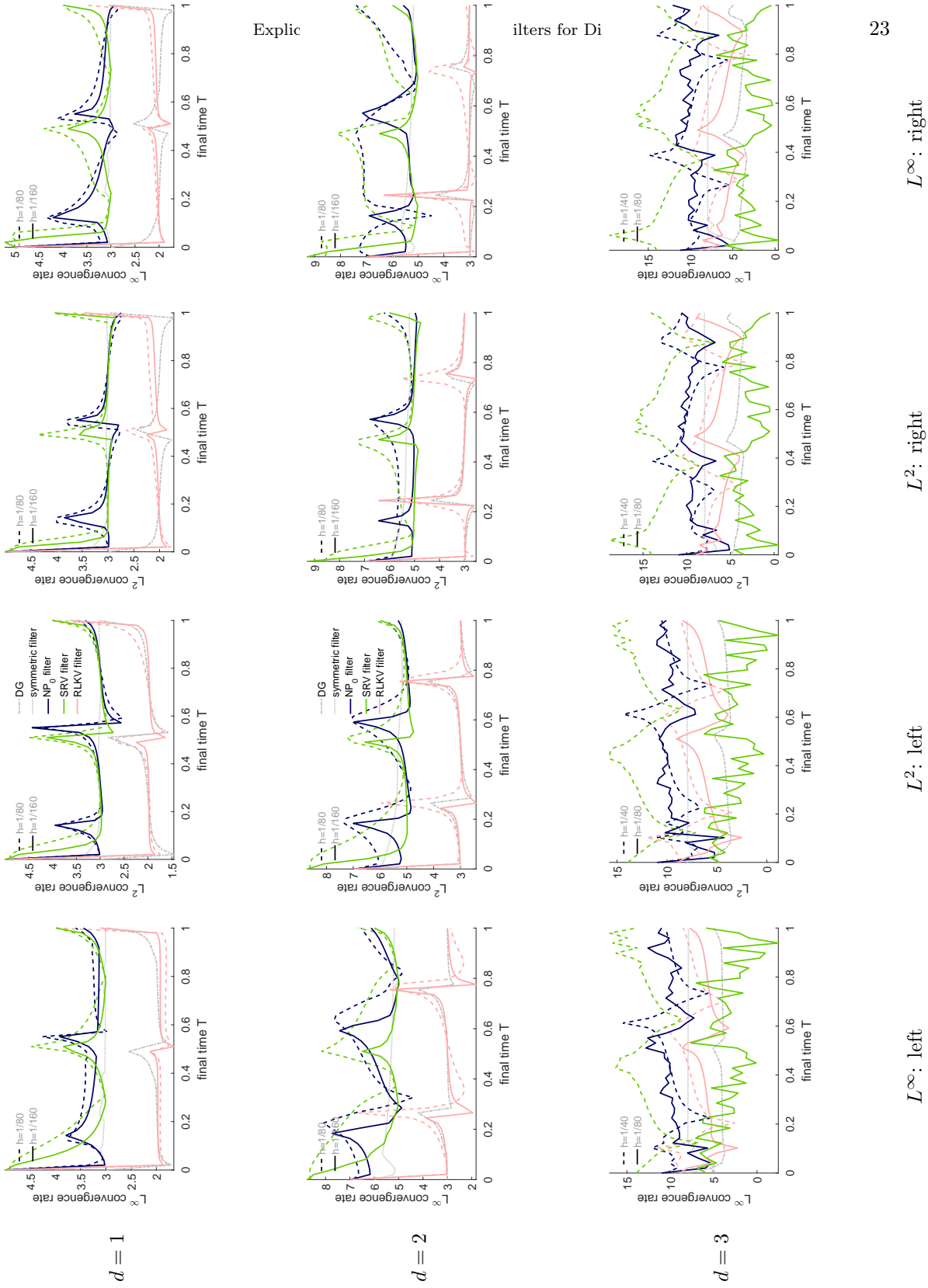
FIG. 9. **Convergence rates** Test 4.1 (constant wave-speed, periodic boundary condition). L^2- and L^∞ convergence of the convolved DG data in the left, respectively right boundary region. Rows 1, 2, 3 show rates for degree $d = 1, 2, 3$. Upper, dotted grey: symmetric SIAC filter; lower dash-dotted: DG rate. The graphs have the same indicative colors and styles as those in Fig. 6, 7 and 8.
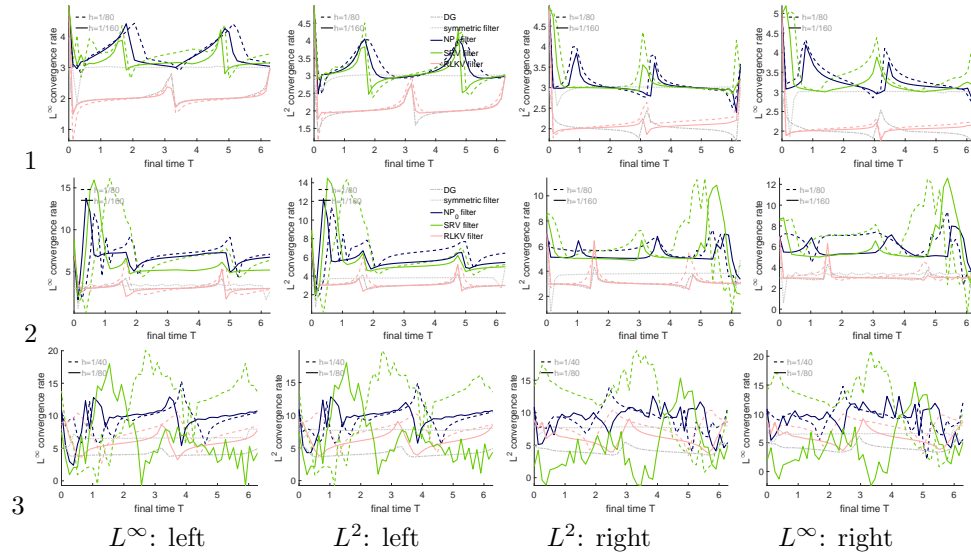
FIG. 10.   **Convergence rates** *Test 4.2 (constant wave-speed, Dirichlet boundary condition).*
L^2- and L^∞ convergence of the convolved DG data in the left, respectively right boundary region.
Rows 1, 2, 3 show rates for degree $d = 1, 2, 3$. Upper, dotted grey: symmetric SIAC filter; lower
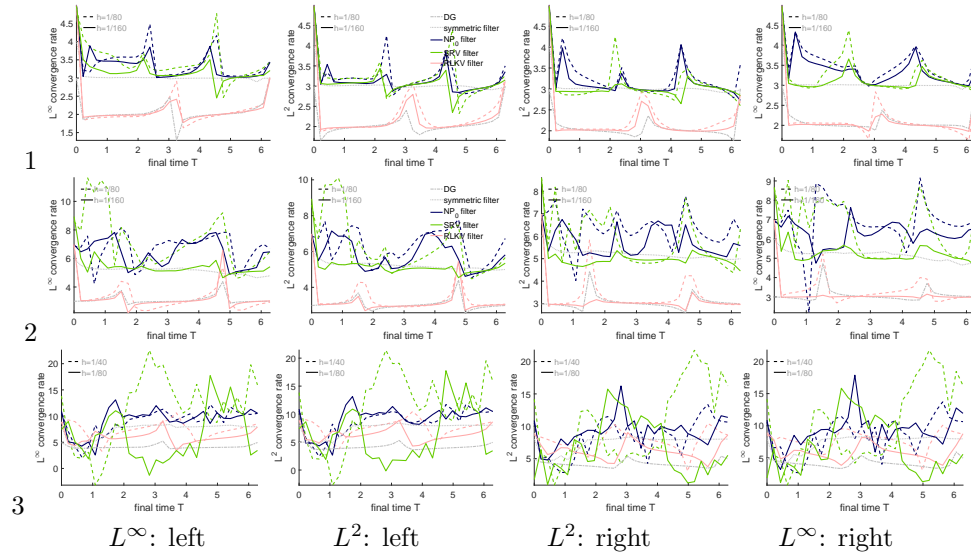*dash-dotted: DG rate.*

FIG. 11.   **Convergence rates** *Test 4.3 (variable wave-speed, periodic boundary condition).*
L^2- and L^∞ convergence of the convolved DG data in the left, respectively right boundary region.
Rows 1, 2, 3 show rates for degree $d = 1, 2, 3$. Upper, dotted grey: symmetric SIAC filter; lower
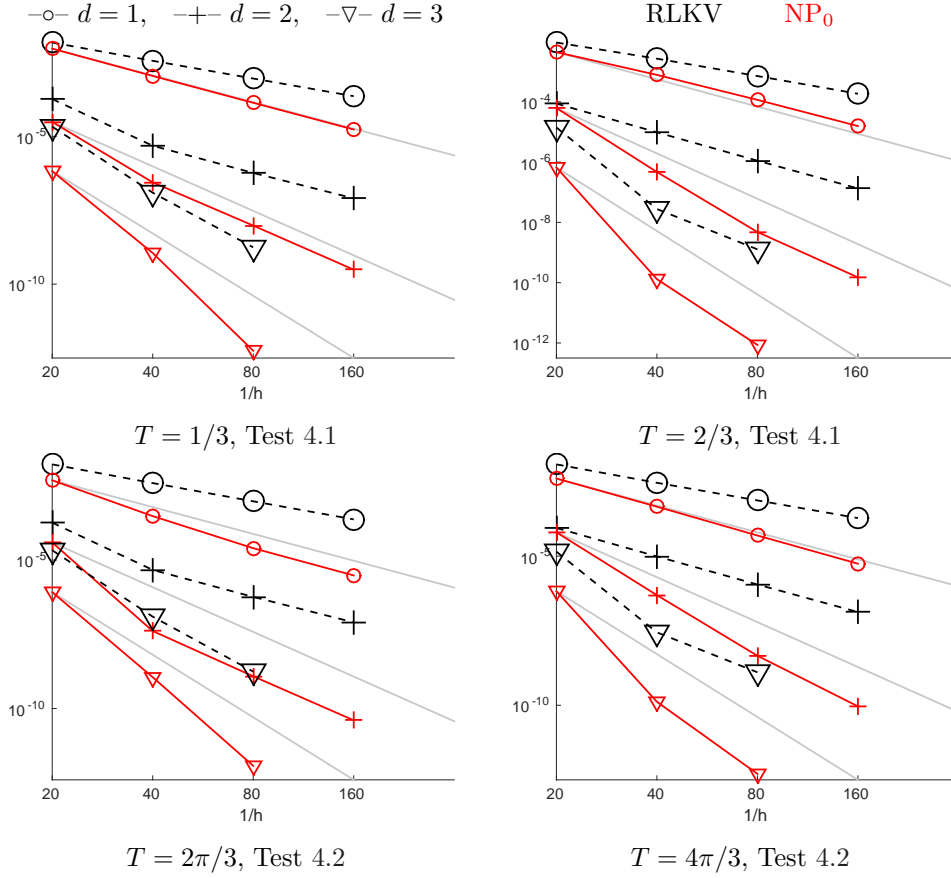*dash-dotted: DG rate.*

Fig. 12. L^2 error and convergence rates for the left boundary region for (top) end times $T \in \{1/3, 2/3\}$ and periodic boundary constraints Test 4.1 and (bottom) $T \in \{2\pi/3, 4\pi/3\}$ for Dirichlet constraints Test 4.2. The ordinate measures the L^2 error on the transition and boundary region, the abcissa counts the number of input DG pieces for the total interval $[0..1]$. Only a constant fraction of these pieces is used by the boundary filters and hence to compute the L^2 norm. The grey downward lines in each graph indicate the slopes for optimal super-convergence order $2d+1$ that are attained by symmetric SIAC filter of DG degree d that is applied in the interior: from top to bottom they correspond to $d = 1, 2, 3$. The larger, black symbols indicate the (typically larger) error for RLKV: The smaller, red symbols indicate the error and rate for NP_0. The symbols are for $d = 1$ '∘', for $d = 2$ '+' and for $d = 3$ '∇'. The graphs for the L^∞ error look alike. Individual pointwise error, i.e. error-vs-x graphs are available at [NPdf].