Max-Planck-Institut für Mathematik Bonn

Discrete least squares quadrature rules on equidistant and arbitrary points

by

Jan Glaubitz

Max-Planck-Institut für Mathematik
Preprint Series 2018 (23)

Discrete least squares quadrature rules on equidistant and arbitrary points

Jan Glaubitz

Max-Planck-Institut für Mathematik
Vivatsgasse 7
53111 Bonn
Germany

Institut Computational Mathematics
TU Braunschweig
Universitätsplatz 2
38106 Braunschweig
Germany

MPIM 18-23

Discrete least squares quadrature rules on equidistant and arbitrary points

Jan Glaubitz*

April, 2018

Quadrature rules are omnipresent in all fields of numerical analysis. Gauss-Lobatto and Gauss-Legendre quadrature rules, for instance, provide stable high-order methods for numerical integration when data is known at the corresponding sets of quadrature points. In most applications, however, it may be impractical - if not even impossible - to obtain data to fit known quadrature rules. In this work, we thus revisit the principle of discrete least squares in order to construct stable high-order rules on equidistant as well as arbitrary sets of points. Further, a sufficient condition for stability to hold for these rules is proven for the whole class of integrators $\alpha \in BV\big([a,b]\big)$ and a comparative study for different choices in the construction of *discrete least squares quadrature rules* is provided. For the first time, several inner products yielding different rules are investigated in a way that allows us to give clear recommendation what kind of discrete least squares quadrature rules should be used in a given situation.

MSC2010: 65D30, 65D32, 93E24, 41A55
Keywords: Numerical integration, stable high-order quadrature rules, least squares approximation, discrete orthogonal polynomials, equidistant points, arbitrary points

1 Introduction

Let us consider a linear functional L on $C[a,b]$, i.e. the set of continuous functions defined on a compact interval $[a,b]$. Then, there exists a function $\alpha \in BV\big([a,b]\big)$ of bounded variation, such that the values of L are given by the Riemann-Stieltjes integral

$$L(f) = \int_a^b f(x) \, \mathrm{d}\,\alpha(x), \quad \forall f \in C([a,b]), \tag{1}$$

where α is referred to as the *integrator of* L, see page 195 in [25].[1] If α is a step function with a finite number of discontinuities, the resulting Riemann-Stieltjes integral reduces to a finite sum

$$L_N(f) = \sum_{n=1}^{N} \omega_n f(x_n), \tag{2}$$

which we call an N*-point quadrature rule for* L. Further note that for a continuously differentiable integrator α the Riemann-Stieltjes integral reduces to

$$\int_a^b f(x) \, \mathrm{d}\,\alpha(x) = \int_a^b f(x)\alpha'(x) \, \mathrm{d}\,x, \tag{3}$$

* *Corresponding author:* Jan Glaubitz
Max-Planck-Institut für Mathematik, Vivatsgasse 7, 53111 Bonn, Germany.
Institut Computational Mathematics, TU Braunschweig, Universitätsplatz 2, 38106 Braunschweig, Germany.
(j.glaubitz@tu-bs.de)

[1] This is the original formulation of F. Riesz's representation theorem which represents the dual space of the Banach space $C\big([a,b]\big)$ of continuous functions on $[a,b]$ as Riemann–Stieltjes integrals against functions of bounded variation. Later, that theorem was reformulated in terms of measures [19].

and numerical integration for weight functions $\omega(x) = \alpha'(x)$ simply arises as another special case.

Often, especially in practical applications, we are faced with the problem to recover $L(f)$ from a finite set of measurements $\{f(x_n)\}_{n=1}^N$ at points $\{x_n\}_{n=1}^N$. In this situation, the idea is to approximate $L(f)$ by the value of an N-point quadrature rule $L_N(f)$, where the so called *weights* $\{\omega_n\}_{n=1}^N$ are chosen such that

$$L_N(f) \to L(f) \quad \text{for} \quad N \to \infty. \tag{4}$$

We say that a sequence of quadrature rules $(L_N)_{N\in\mathbb{N}}$ *converges* to the linear functional L if (4) holds for all sufficiently smooth functions f. Typically, the function f has to be at least Lipschitz-continuous for convergence to hold [9].

Yet, in many situations quadrature rules are not just desired to approximate a linear functional L sufficiently accurate, but are furthermore required to be exact, at least for a certain finite dimensional subspace $V \subset C[a,b]$. A typical choice is $V = \mathbb{P}_{d-1}$, i.e. the space of polynomials of degree at most $d-1$. We say that a quadrature rule L_N has *order of exactness* d if it is exact for all polynomials of degree $d-1$ or less, i.e. if

$$L_N(p) = L(p) \quad \forall p \in \mathbb{P}_{d-1} \tag{5}$$

holds.

If one can freely chose the points $\{x_n\}_{n=1}^N$, there are many classes of quadrature rules which are able to provide high orders of convergence as well as high orders of exactness, for instance Gauss type methods for numerical integration. In realistic applications, however, measurements are typically performed at equidistant or even arbitrary points. Here, the usual quadrature rules break down. While composite quadrature rules, for instance, are easy to implement and easy to apply, they are detained from high orders of exactness and converge slowly. Interpolatory quadrature rules, on the other hand, can provide any order of exactness d if $N = d$ points are used, but might not converge as $N \to \infty$ and are often numerically unstable.

This problem is due to a sequence of quadrature rules $(L_N)_{N\in\mathbb{N}}$ with

$$L_N(f) = \sum_{n=1}^N \omega_{n,N} f(x_{n,N}) \tag{6}$$

to feature weights $\{\omega_{n,N}\}_{n=1}^N$ of mixed signs. Once weights with negative signs arise in a approximating series of quadrature rules $(L_N)_{N\in\mathbb{N}}$, convergence breaks down and they become numerically unstable. At least for usual classes of quadrature rules on equidistant or arbitrary sets of points, such as interpolatory and composite quadrature rules, stability in the sense of nonnegative-only weights and high orders of exactness seem to be orthogonal properties.

In this work, we thus aim to construct nonstandard stable quadrature rules on equidistant or even arbitrary points which are also able to provide high orders of exactness. The idea is to replace the usual interpolation polynomial in interpolatory quadrature rules by more general discrete least squares (DLS) approximations. To the resulting quadrature rules we will thus refer to as *discrete least squares quadrature rules (DLS-QR)*. This approach originates from the 1970 works [27, 28] of Wilson. To the best of the author's knowledge, the connection between discrete least squares approximations and quadrature rules for numerical integration was not further explored since then, and just revisited in the work [18] of Huhybrechs.

Since then, the present work is the first to carefully investigate different existing and novel approaches for the discrete least squares approximations, i.e. for the underlying discrete inner product. A comparison of these allows us to give clear recommendations what kind of discrete least squares quadrature rules should be used in a given situation.

The rest of this work is organised as follows. In section 2, we revisit the concept of interpolatory quadrature rules and in particular investigate numerical stability - or rather its absence - for Newton-Cotes rules on equidistant points. This serves as both a motivation of stability in the framework of numerical integration as well as for this work to be self-contained. In section 3, discrete least squares quadrature rules are introduced as certain least squares solutions of an underdetermined linear system. Further, the concept of discrete orthogonal polynomials and their stable construction will be incorporated in order to provide a stable and efficient algorithm for the construction of discrete least squares quadrature rules. Stability of discrete least squares quadrature rules themselves is investigated in section 4. A sufficient condition for stability to hold for a sufficiently number of quadrature points is derived and proven. Moreover, a comparative study for different discrete least squares quadrature rules is started in this section. This study is continued and extended in section 5. Besides comparing the accuracy of different discrete least squares quadrature rules to interpolatory and composite rules we demonstrate that stability for discrete least squares quadrature rules essentially is a $N \approx Cd^2$ process for commonly used integrators. We close this work with some concluding thoughts in section 6.

2 Interpolatory quadrature rules

Interpolatory quadrature rules are based on polynomial interpolation, where the quadrature points $\{x_n\}_{n=1}^N$ are used as the interpolation points. The idea is to replace f by its interpolation polynomial f_N, for instance in the Lagrange basis given by $f_N(x) = \sum_{n=1}^N f(x_n) l_n(x)$, and to perform integration over f_n instead of f then, yielding

$$L(f) = \int_a^b f(x) \, \mathrm{d}\,\alpha(x) \approx \int_a^b f_N(x) \, \mathrm{d}\,\alpha(x) = \sum_{n=1}^N \underbrace{\left(\int_a^b l_n(x) \, \mathrm{d}\,\alpha(x) \right)}_{=\omega_n} f(x_n) = L_N(f). \tag{7}$$

While the Lagrange basis has the advantage of providing an explicit formula for the weights, other bases are possible. Especially from a computational point of view, these offer more robust alternatives for the computation of the weights.

2.1 Newton-Cotes rules

In applications, f is often known only at equidistant points

$$x_n = a + (b-a)\frac{n-1}{N-1}, \quad n = 1, \dots, N, \tag{8}$$

on $[a,b]$. For $N = 1$, we set $x_1 = (a+b)/2$, i.e. the midpoint of the interval. Equidistant points often arise in finite difference and collocation methods for ordinary as well as partial differential equations or integral equations. See for instance [3, 5, 14, 21] and references therein.

When using equidistant points, interpolatory quadrature rules defined by (7) are called *Newton-Cotes rules*. Other popular quadrature rules for equidistant points are composite Newton-Cotes rules, such as the composite trapezoidal rule.

It is well-known that Newton-Cotes rules for the set of N points are exact for all polynomials of degree $N-1$ or less, due to polynomial interpolation to be exact in this case. Note that Newton-Cotes quadrature rules might also be derived directly from exactness condition (5) by formulating it for every element of a basis $\{\varphi_n\}_{n=1}^N$ of $\mathbb{P}_{N-1}$. Then, for a given vector of N quadrature points $\underline{x}$, the vector of weights $\underline{\omega}$ can be obtained by solving the linear system

$$\underline{\underline{V}}\,\underline{\omega} = \underline{m}, \tag{9}$$

where the so-called *Vandermonde matrix* $\underline{\underline{V}} = \big(\varphi_m(x_n)\big)_{m,n=1}^N$ contains the function evaluations of the basis elements φ_m and the right-hand side vector $\underline{m} = \big(L(\varphi_m)\big)_{m=1}^N$ contains the *moments* of the basis elements. Again, the quadrature rule constructed in this way interpolates the function f in the quadrature points $\underline{x}$ and integrates the resulting interpolation function f_N exactly.

Table 1 shows the weights of the first nine Newton-Cotes rules for $[a,b] = [0,1]$ and integrator $\alpha(x) = x$, i.e. the common Riemann integral. For a general interval $[a,b]$, the weights just have to be multiplied by $(b-a)$.

Table 1 further lists the errors for the (composite) Newton-Cotes rules, where

$$h = h_N = \begin{cases} (b-a), & \text{if } \ N = 1, \\ \frac{(b-a)}{N-1}, & \text{if } \ N > 1. \end{cases} \tag{10}$$

See for instance chapter 2.5 of [8]. Note the feature of the Newton-Cotes rules that if the number of points is odd, i.e. $N = 2k-1$ for $k \in \mathbb{N}$, the error is of the form

$$E(f) = c_k h^{2k+1} f^{(2k)}(\xi). \tag{11}$$

If the number of points is even, i.e. $N = 2k$ for $k \in \mathbb{N}$, the error is of the form

$$E(f) = d_k h^{2k+1} f^{(2k)}(\xi) \tag{12}$$

as well. A similar behaviour can be observed for the order of exactness.

N	weights $\underline{\omega}$	E(f)
1	1	$\frac{1}{24}h^3 f^{(2)}(\xi)$
2	$\frac{1}{2}\ \frac{1}{2}$	$-\frac{1}{12}h^3 f^{(2)}(\xi)$
3	$\frac{1}{6}\ \frac{4}{6}\ \frac{1}{6}$	$-\frac{1}{90}h^5 f^{(4)}(\xi)$
4	$\frac{1}{8}\ \frac{3}{8}\ \frac{3}{8}\ \frac{1}{8}$	$-\frac{3}{80}h^5 f^{(4)}(\xi)$
5	$\frac{7}{90}\ \frac{32}{90}\ \frac{12}{90}\ \frac{32}{90}\ \frac{7}{90}$	$-\frac{8}{945}h^7 f^{(6)}(\xi)$
6	$\frac{19}{288}\ \frac{75}{288}\ \frac{50}{288}\ \frac{50}{288}\ \frac{75}{288}\ \frac{19}{288}$	$-\frac{275}{12096}h^7 f^{(6)}(\xi)$
7	$\frac{41}{840}\ \frac{216}{840}\ \frac{27}{840}\ \frac{272}{840}\ \frac{27}{840}\ \frac{216}{840}\ \frac{41}{840}$	$-\frac{9}{1400}h^9 f^{(8)}(\xi)$
8	$\frac{751}{17280}\ \frac{3577}{17280}\ \frac{1223}{17280}\ \frac{2989}{17280}\ \frac{2989}{17280}\ \frac{1223}{17280}\ \frac{3577}{17280}\ \frac{751}{17280}$	$-\frac{8183}{518400}h^9 f^{(8)}(\xi)$
9	$\frac{989}{28350}\ \frac{5888}{28350}\ -\frac{928}{28350}\ \frac{10496}{28350}\ -\frac{454}{28350}\ \frac{10496}{28350}\ -\frac{928}{28350}\ \frac{5888}{28350}\ \frac{989}{28350}$	$-\frac{2368}{467775}h^{11} f^{(10)}(\xi)$

Table 1: Quadrature weights and errors for the (composite) Newton-Cotes rules up to $N = 9$.

2.2 Stability of Newton-Cotes rules

For a variety of integrators α, Newton-Cotes rules are easy to apply and to implement. Yet, the low-order (composite) rules converge fairly slowly, while the high-order rules are known to be numerically unstable. Stability in the framework of quadrature rules follows from having only positive weights $\omega_{n,N}$. Such quadrature rules are known to converge for all continuous functions $f \in C([a, b])$.

A common measure of stability of quadrature rules is given by

$$\kappa(\underline{\omega}) := \sum_{n=1}^{N} |\omega_n|. \tag{13}$$

We call a quadrature rule L_N with weights $\underline{\omega}$ *stable* if

$$\kappa(\underline{\omega}) = L(1) \tag{14}$$

holds. The idea behind this concept is that for a perturbed input $\tilde{f} \in C([a, b])$ with $|\tilde{f}(x) - f(x)| \leq \varepsilon$, the error of the quadrature rule can be estimated by

$$\left| L_N(\tilde{f}) - L_N(f) \right| \leq \sum_{n=1}^{N} \left| \omega_n \left(\tilde{f}(x_n) - f(x_n) \right) \right| \leq \varepsilon \kappa(\underline{\omega}). \tag{15}$$

Round-off errors due to a inexact arithmetic are hence bounded by the factor $\kappa(\underline{\omega})$. Note that for positive weights $\kappa(\underline{\omega}) = L(1)$, while otherwise $\kappa(\underline{\omega}) > L(1)$. It is obvious that a quadrature rule is stable if and only if all its weights are non-negative. This is summarised in

Lemma 2.1. *Let L_N be a quadrature rule for the linear functional L with weights $\underline{\omega} = (\omega_1, \ldots, \omega_N)$ and points $\{x_n\}_{n=1}^N \subset [a, b]$, i.e.*

$$L_N(f) = \sum_{n=1}^{N} \omega_n f(x_n), \quad \text{for} \quad f \in C([a, b]). \tag{16}$$

Then L_N is stable if and only if all weights are non-negative.

Unfortunately, as it is listed in Table 1, Newton-Cotes rules feature negative weights starting from $N = 9$. The instability intensifies for higher orders $N \geq 11$, where Newton-Cotes rules have large weights with differing signs. Figure 1 illustrates the rising instability of Newton-Cotes rules as the number of equidistant quadrature points N increases. Here, integration is performed on $[0, 1]$ and for integrator $\alpha(x) = x$.

As a result, Newton-Cotes rules suffer from a major loss of precision when inexact arithmetics are used, due to round-off errors to heavily pollute the computations then.

Note that even in exact arithmetcis, Newton-Cotes rules are not guaranteed to converge. This problem is related to the failure of the underlying interpolation polynomials f_N to converge. As Runge showed in

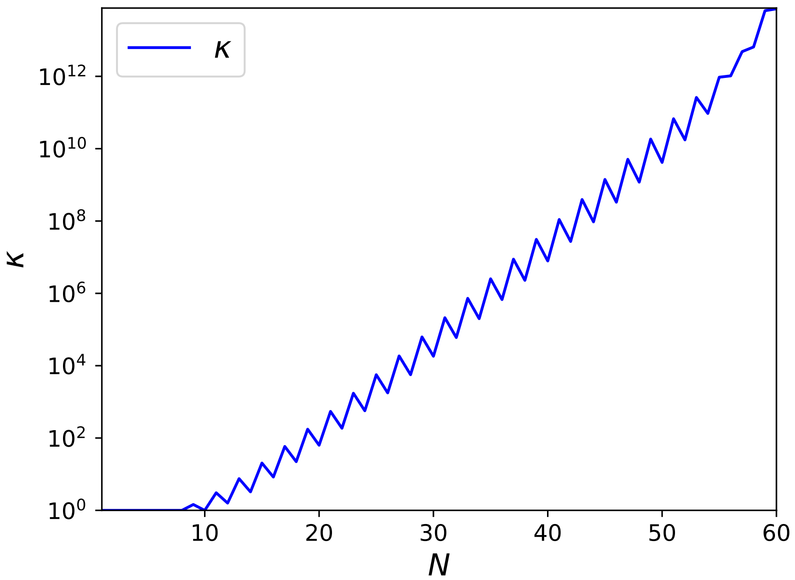


Figure 1: Stability values $\kappa(\underline{\omega})$ for Newton-Cotes rules.

1901 [23], polynomial interpolation in equidistant points may diverge exponentially, even if the underlying function f is arbitrarily smooth. A sufficient condition for convergence to hold is that f is analytic in an ellipse in the complex plane centred at $\frac{1}{2}(a+b)$ with major axis of length $\frac{10}{8}(b-a)$ along the real axis and minor axis of length $\frac{6}{8}(b-a)$ along the imaginary axis [7].

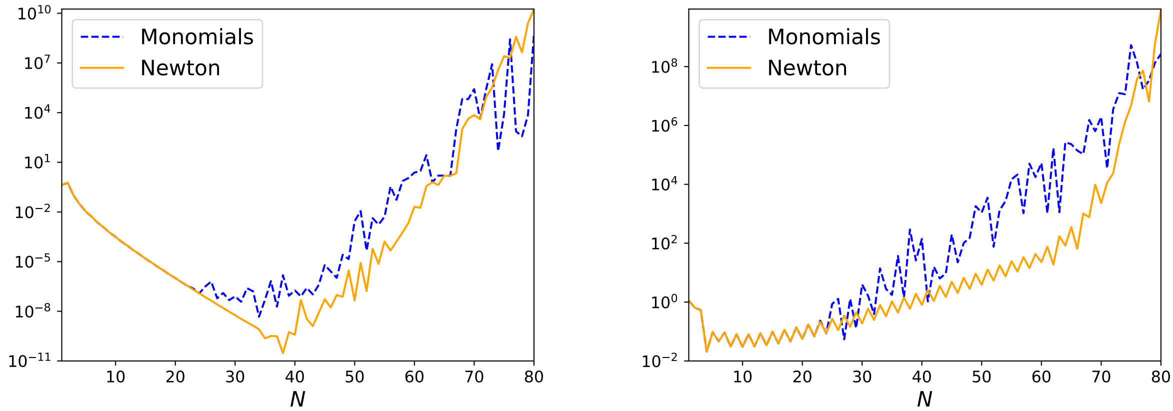
2.3 Numerical tests for Newton-Cotes rules

In this subsection, we demonstrate the numerical performance of Newton-Cotes rules for the functions $f(x) = \frac{1}{1+x^2}$ and $f(x) = \frac{1}{1+8x^2}$ on $[-1,1]$ for integrator $\alpha(x) = x$. Figure 2a shows the $|\cdot|$-errors for $f(x) = \frac{1}{1+x^2}$, while Figure 2b shows the $|\cdot|$-errors for $f(x) = \frac{1}{1+8x^2}$. Both times, the errors are plotted against an increasing number of equidistant points N. The (blue) dashed line thereby corresponds to the Newton-Cotes rules where the weights have been determined by solving the linear system (9) for the basis of monomials $\{x^m\}_{m=0}^{N-1}$. The (orange) straight line, on the other hand, corresponds to the Newton-Cotes rules where the weights have been determined by solving the linear system (9) for the basis of Newton polynomials

$$N_1(x) = 1, \quad N_m(x) = (x-x_1)\cdot(x-x_2)\cdot\dots\cdot(x-x_m), \text{ for } m = 2,\dots,N. \tag{17}$$

We use this opportunity to point out the importance of robustly solving the linear system (9) for the weights. It is well-known that the basis of monomials yield the Vandermonde matrix to become very ill-conditioned for equidistant points. For the basis of Newton polynomials, however, the Vandermonde matrix $\underline{\underline{V}}$ becomes an upper triangular matrix and the linear system (9) can be easily solved iteratively. As a result, the performance of the Newton-Cotes rule in Figure 2a is superior for the Newton basis compared to the basis of monomials.

Further note that $f(x) = \frac{1}{1+x^2}$ has its poles at $\pm i$ and thus is analytic in a sufficiently large region for polynomial interpolation to converge. In accordance to this, Figure 2a illustrates convergence of the Newton-Cotes rules until $N \approx 40$. Starting at $N \approx 40$, convergence starts to fail since the numerical instability of the Newton-Cotes rules becomes a dominating factor. For $f(x) = \frac{1}{1+8x^2}$, on the other hand, convergence fails from the start, as can be observed in Figure 2b. This is due to f having its poles $\pm\frac{i}{2\sqrt{2}}$ too close to the real axis and thus being not analytic in a large enough region for polynomial interpolation to converge. This time starting at $N \approx 70$ for the basis of Newton polynomials, rounding-off errors due to the numerical instability of the Newton-Cotes rules fully pollute the computations. The jagged patterns in Figure 2 as well as in Figure 1 are due to the above mentioned difference in the order of convergence for even and odd numbers of points N. All tests were computed in the programming language *Julia* [2] using double precision (16 digits).



(a) Errors for $f(x) = \frac{1}{1+x^2}$ on $[-1, 1]$.

(b) Errors for $f(x) = \frac{1}{1+8x^2}$ on $[-1, 1]$.

Figure 2: Errors for Newton-Cotes rules with respect to the number of points N. Both tests demonstrate divergence due to numerical instabilty.

3 Discrete least squares quadrature rules

In section 2, we have demonstrated that the use of (composite) Newton-Cotes rules is not recommended when high orders of convergence or of exactness are required. An important issue in numerical integration is therefore the construction of other stable high-order quadrature rules. In this section we follow the works [27, 28] of Wilson and do so by replacing the function f by its least squares approximation

$$f_{d,N} \in \mathbb{P}_{d-1} \quad \text{such that} \quad \sum_{n=1}^{N} \left| f(x_n) - f_{d,N}(x_n) \right|^2 = \min_{p \in \mathbb{P}_{d-1}} \sum_{n=1}^{N} \left| f(x_n) - p(x_n) \right|^2 . \tag{18}$$

with respect to a greater number of quadrature points N than technically needed. In Analogy to interpolatory quadrature rules, integration is performed over the least squares approximation then, yielding

$$L(f) = \int_a^b f(x) \,\mathrm{d}\,\alpha(x) \approx \int_a^b f_{d,N}(x) \,\mathrm{d}\,\alpha(x) = L_{d,N}(f). \tag{19}$$

After introducing more general (discrete) least squares approximations $f_{d,N}$ than the standard one above, an efficient approach to compute the coefficients of $f_{d,N}$ by bases of discrete orthogonal polynomials will be presented. In analogy to (9), we will characterise the weights of the arising discrete least squares quadrature rules (DLS-QRs) as certain solutions of an underdetermined linear system. Here, the weights will correspond to solutions which minimise certain weighted discrete norms.

In contrast to former works, the influence of different weighted inner products will be investigated in great detail. It turns out that by this procedure, rules with positive weights can be constructed on equidistant or any other grid of points by letting the number of quadrature points N be sufficiently large.

3.1 Discrete least squares approximations

We now introduce the discrete least squares (DLS) approximation of a function f in order to replace the usual interpolation polynomial f_N in the approach for interpolatory quadrature rules by them. In the last section, two crucial problems with interpolatory quadrature rules, and in particular Newton-Cotes rules have already been addressed: Numerical instability, and thus possibly heavy rounding errors in inexact arithmetics, and divergence even in exact arithmetics and already for some analytic functions f. Yet, both problems can be overcome by the introduction of discrete least squares quadrature rules. Stability for the resulting quadrature rules will be proven in section 4, while we refer to [4, 13, 15, 22] for studies on convergence of discrete least squares approximations.

The idea behind discrete least squares approximations is to define a discrete inner product

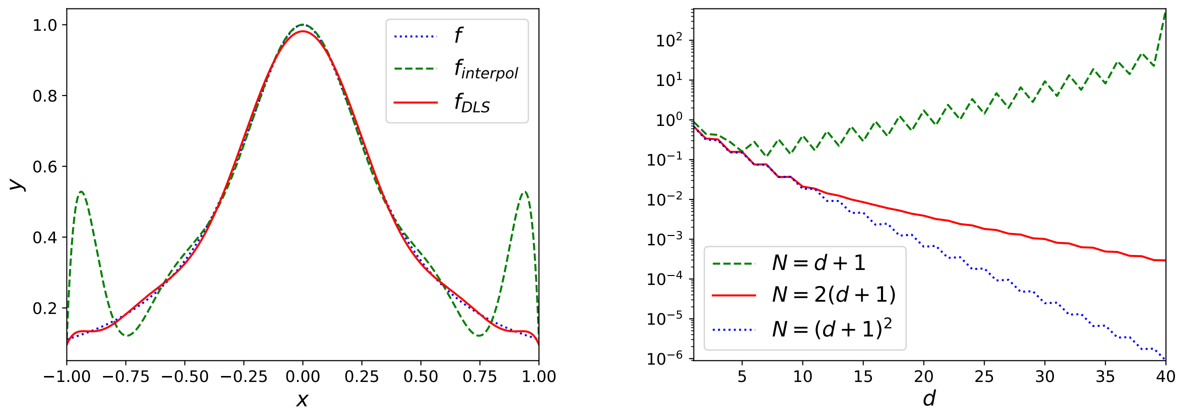
$$\langle f, g \rangle_{\underline{r}} = \sum_{n=1}^{N} r_n f(x_n) g(x_n) \tag{20}$$

on a set of N points $\{x_n\}_{n=1}^N$ and to determine the polynomial best approximation $f_{d,N} \in \mathbb{P}_{d-1}$ then. This best approximation $f_{d,N}$ is referred to as the *discrete least squares approximation* of f with respect to the discrete inner product $\langle \cdot, \cdot \rangle_{\underline{r}}$. Thus, $f_{d,N}$ can also be characterised by

$$\left\| f - f_{d,N} \right\|_{\underline{r}} = \min_{p \in \mathbb{P}_{d-1}} \| f - p \|_{\underline{r}}, \tag{21}$$

where $\|p\|_{\underline{r}}^2 = \langle p, p \rangle_{\underline{r}}$ denotes the norm induced by the inner product. The concept of (discrete) least squares approximations dates back to Gauss and Legendre [1, 16, 20]. First famous examples for the least squares method include the analysis of survey data and astronomical calculations, such as the successfully predicted orbit of the asteroid Ceres by Gauss in 1801 [12].

Figure 3 illustrates the enhanced accuracy of discrete least squares approximations for $f(x) = \frac{1}{1+8x^2}$ from section 2.



(a) Polynomial interpolation and a discrete least squares approximation of $f(x) = \frac{1}{1+8x^2}$ on $[-1, 1]$.

(b) $|| \cdot ||_\infty$-errors for different discrete least squares approximations of $f(x) = \frac{1}{1+8x^2}$ on $[-1, 1]$.

Figure 3: Polynomial approximations of $f(x) = \frac{1}{1+8x^2}$ on $[-1, 1]$ by polynomial interpolation and different discrete least squares approximations. The choice $N = d$ corresponds to polynomial interpolation.

In particular, Figure 3a provides a comparison of polynomial interpolation and a discrete least squares approximation for polynomial degree $d = 11$. The discrete least squares approximation was computed using $N = d^2 = 121$ equidistant points with constant weights $r_n = \frac{2}{N}$. Even though a slightly oscillatory behaviour is still present at the element boundaries, the discrete least squares approximation is demonstrated to be significantly more accurate than polynomial interpolation. This behaviour is further expanded in Figure 3b, where polynomial interpolation and different least squares approximations are compared with respect to the $|| \cdot ||_\infty$-norm. The discrete least squares approximations are performed for $N = d, 2d$, and d^2 equidistant points and constant weights $r_n = \frac{2}{N}$ again. Note that the discrete least squares approximation $N = d$ is nothing more than the interpolation polynomial.

3.2 Discrete orthogonal polynomials

We now address the problem of efficiently and robustly computing discrete least squares approximations $f_{d,N}$ characterised by (21) or equivalently by

$$\left\langle f - f_{d,N}, p \right\rangle_{\underline{r}} = 0 \quad \forall p \in \mathbb{P}_{d-1}. \tag{22}$$

Expressing the discrete least squares approximation $f_{d,N}$ with respect to a basis $\{\varphi_k\}_{k=1}^d$ of $\mathbb{P}_{d-1}$, i.e. $f_{d,N}(x) = \sum_{k=1}^d \hat{f}_{d,N} \varphi_k(x)$, relation (22) result in a linear system of equations

$$\sum_{k=1}^{d} \hat{f}_{k,N} \langle \varphi_k, \varphi_i \rangle_{\underline{r}} = \langle f, \varphi_i \rangle_{\underline{r}} \quad \text{for } i = 1, \ldots, d, \tag{23}$$

which might be solved by Gaussian elimination. Things get far more pleasant, however, once $f_{d,N}$ is expressed in an orthogonal basis, for which $\langle \varphi_k, \varphi_i \rangle_{\underline{r}} = \delta_{i,k} \|\varphi_k\|_{\underline{r}}^2$ holds. Then, the coefficients are directly given by

$$\hat{f}_{k,N} = \frac{\langle f, \varphi_k \rangle_{\underline{r}}}{\|\varphi_k\|_{\underline{r}}^2}. \tag{24}$$

It should be stressed that for non-classical orthogonal polynomials, i.e. non-classical inner products, often no explicit formula is known. Here, we are in need of a basis $\{\varphi_k\}_{k=1}^d$ of $\mathbb{P}_{d-1}$ which is orthogonal with respect to a discrete inner product $\langle f, g \rangle_{\underline{r}} = \sum_{n=1}^N r_n f(x_n) g(x_n)$. Such bases are called *discrete orthogonal polynomials* [13]. Fortunately, the literature offers a few construction algorithms for bases of discrete orthogonal polynomials, for instance the Stieltjes procedure [13] and Gram-Schmidt process [26]. Yet, it should be stressed heavily that neither the *Stieltjes procedure*,

$$\begin{aligned} \varphi_0 &= 0, \ \varphi_1 = 1, \\ \varphi_{k+1} &= (x - \alpha_k)\, \varphi_k - \beta_k \varphi_{k-1}, \end{aligned} \tag{25}$$

with recursion coefficients

$$\alpha_k = \frac{\langle x\varphi_k, \varphi_k \rangle_{\underline{r}}}{\langle \varphi_k, \varphi_k \rangle_{\underline{r}}}, \qquad \beta_k = \frac{\langle \varphi_k, \varphi_k \rangle_{\underline{r}}}{\langle \varphi_{k-1}, \varphi_{k-1} \rangle_{\underline{r}}}, \tag{26}$$

nor the *classical Gram-Schmidt process* applied to an initial basis $\{v_k\}_{k=1}^d$ of (typically non-orthogonal) polynomials,

$$\begin{aligned} \tilde{\varphi}_1 &= v_1, \quad \varphi_1 = \frac{\tilde{\varphi}_1}{\|\tilde{\varphi}_1\|_{\underline{r}}}, \\ \tilde{\varphi}_{k+1} &= v_{k+1} - \sum_{i=1}^{k} \langle \varphi_i, v_{k+1} \rangle_{\underline{r}}\, \varphi_i, \quad \varphi_{k+1} = \frac{\tilde{\varphi}_{k+1}}{\|\tilde{\varphi}_{k+1}\|_{\underline{r}}}, \end{aligned} \tag{27}$$

are recommend, since they are numerically unstable. This is also demonstrated in Figure 4. In our implementation we thus constructed all bases of discrete orthogonal polynomials by the numerical stable *modified Gram-Schmidt process* [26], where $\tilde{\varphi}_{k+1}$ is computed by

$$\begin{aligned} \tilde{\varphi}_{k+1}^{(1)} &= v_{k+1} - \langle \varphi_1, v_{k+1} \rangle_{\underline{r}}\, \varphi_1, \\ \tilde{\varphi}_{k+1}^{(i)} &= \tilde{\varphi}_{k+1}^{(i-1)} - \left\langle \varphi_i, \tilde{\varphi}_{k+1}^{(i-1)} \right\rangle_{\underline{r}} \varphi_i, \quad \text{for } i = 2, \dots, k \\ \tilde{\varphi}_{k+1} &= \tilde{\varphi}_{k+1}^{(k)}, \end{aligned} \tag{28}$$

instead of (27).

A comparative study of the above construction algorithms is provided by Figure 4. There, the orthonormality-error E_{ON} is plotted against an increasing number of elements d. Thereby, the *orthonormality-error* $E_{ON}\left(\{\varphi_k\}_{k=1}^d\right)$ for a basis $\{\varphi_k\}_{k=1}^d$ is defined as

$$E_{ON}\left(\{\varphi_k\}_{k=1}^d\right) := \sum_{i,k=1}^{d} \left| \delta_{i,k} - \langle \varphi_i, \varphi_k \rangle \right|, \tag{29}$$

and thus measures how well the basis $\{\varphi_k\}_{k=1}^d$ (typically produced by one of the procedures above) approximates the orthonormal-property $\langle \varphi_i, \varphi_k \rangle = \delta_{i,k}$. Figure 4 illustrates that the modified Gram-Schmidt (mod G-S) process (28) provides better approximations of orthonormal bases then the classic Gram-Schmidt (cla G-S) process (27) as well as the Stieltjes (Stieltjes) procedure (25). In this test, the discrete inner product $\langle a, b \rangle_{\underline{r}} = \frac{2}{N} \sum_{n=1}^N a(x_n) b(x_n)$ on equidistant points x_n in $[-1, 1]$ has been used. Further, both Gram-Schmidt process have been applied to an initial basis of Legendre polynomials.

3.3 Discrete least squares quadrature rules

We now introduce discrete least squares quadrature rules, which will provide stable high-order rules, even on equidistant and arbitrary points. There are two ways of thinking about least squares quadrature rules. The first way has already been scratched in (7) and is to replace f by a discrete least squares approximation $f_{d,N}$, performing integration over $f_{d,N}$ then.

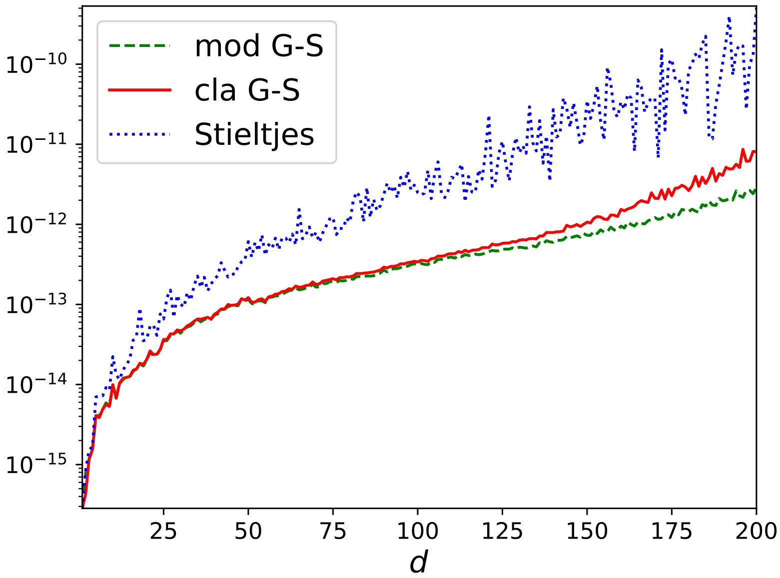


Figure 4: Orthonormality-errors E_{ON} for the modified as well as classical Gram-Schmidt process and the Stieltjes procedure.

This approach, however, won't directly reveal the quadrature weights of the resulting quadrature rule. As a second way, one might determine the weights of a discrete least squares quadrature rule by 'solving' a linear system analogously to (9). Yet, by allowing the number of quadrature points N to be larger than the order of exactness d, the linear system

$$\underline{\underline{V}}\,\underline{\omega} = \underline{m} \quad \text{with} \quad \underline{\underline{V}} = \left(\varphi_k(x_n)\right)_{k,n=1}^{d,N} \tag{30}$$

becomes underdetermined. The $(N-d)$-dimensional affine subspace of solutions

$$\Omega = \{\underline{\omega} \in \mathbb{R}^N \mid \underline{\underline{V}}\,\underline{\omega} = \underline{m}\} \tag{31}$$

is, for instance, generated by $(N-d)$ vectors $\underline{\omega}$ corresponding to distinct interpolatory quadrature rules utilising only d of the N points. Note that for every $\underline{\omega} \in \Omega$, the corresponding quadrature rule

$$L_N(f) = \sum_{n=1}^{N} \omega_n f(x_n) \tag{32}$$

provides order of exactness d. To furthermore achieve stability, we now seek for the quadrature rule with the best stability properties among all the possible quadrature rules contained in Ω. Ideally, we would do so by determining $\underline{\omega}^* \in \Omega$ such that

$$\kappa\left(\underline{\omega}^*\right) \leq \kappa\left(\underline{\omega}\right) \quad \forall \underline{\omega} \in \Omega, \tag{33}$$

which would be l^1-minimisation. In this work, however, we will go over to the more practical task to do this minimisation with respect to certain *weighted least squares stability measures*

$$\kappa_{\underline{r}}\left(\underline{\omega}\right) := \sum_{n=1}^{N} r_n \left|\omega_n\right|^2. \tag{34}$$

In section 4, we will prove that this also yields stability, i.e. $\kappa\left(\underline{\omega}^*\right) = L(1)$, for sufficiently large N and appropriate weights $\underline{r}$.

Going over to this more suitable class of stability measures, the unique[2] vector $\underline{\omega}^* \in \Omega$ such that $\kappa_{\underline{r}}\left(\underline{\omega}^*\right) \leq \kappa_{\underline{r}}\left(\underline{\omega}\right)$ for all $\underline{\omega} \in \Omega$ is given by

$$\underline{\omega}^* = \arg\min_{\underline{\omega}\in\Omega} \left\|\underline{\omega}\right\|_{\underline{r}} \tag{35}$$

[2] Uniqueness follows from the approximation to be performed w.r.t. an inner product $\langle \underline{u}, \underline{v}\rangle_{\underline{r}} = \sum_{n=1}^{N} r_n u_n v_n$.

then, i.e. as the least squares solution of the underdetermined system (30). For the computation of $\underline{\omega}^*$ note the connection

$$\|\underline{\omega}\|_{\underline{r}} = \left\| \sqrt{\underline{\underline{R}}}\,\underline{\omega} \right\|_2, \quad \text{with} \quad \underline{\underline{R}} = \operatorname{diag}(r_1, \ldots, r_n) \tag{36}$$

between the different discrete inner products. Thereby, $\|\underline{\omega}\|_2$ denotes the norm induced by the *Euclidean inner product* $\langle \underline{u}, \underline{v} \rangle_2 = \sum_{n=1}^N u_n v_n$ on $\mathbb{R}^N$. Following [17], the vector $\underline{\omega}^* \in \Omega$ is thus given by

$$\underline{\omega}^* = \sqrt{\underline{\underline{R}}}\,\underline{\omega}^{LS}, \tag{37}$$

where the vector $\underline{\omega}^{LS}$ is the common[3] least squares solution of the (modified) underdetermined system

$$\underline{\underline{V}}\sqrt{\underline{\underline{R}}}\,\underline{\omega} = \underline{m}. \tag{38}$$

Since this is a common least squares problem, the vector $\underline{\omega}^{LS}$ is obtained by

$$\underline{\omega}^{LS} = \left(\underline{\underline{V}}\sqrt{\underline{\underline{R}}} \right)^T \underline{u}, \tag{39}$$

where $\underline{u}$ is the unique solution of the (modified) normal equation

$$\left(\underline{\underline{V}}\sqrt{\underline{\underline{R}}} \right)^T \left(\underline{\underline{V}}\sqrt{\underline{\underline{R}}} \right) \underline{u} = \underline{m}. \tag{40}$$

The weighted least squares solution $\underline{\omega}^*$ might now be determined by solving the linear system (40) and computing

$$\underline{\omega}^* = \sqrt{\underline{\underline{R}}} \left(\underline{\underline{V}}\sqrt{\underline{\underline{R}}} \right)^T \underline{u} = \underline{\underline{R}}\,\underline{\underline{V}}^T \underline{u}. \tag{41}$$

The real advantage of this approach reveals, however, when incorporating the concept of discrete orthogonal polynomials from the previous subsection. If the Vandermonde matrix $\underline{\underline{V}} = \left(\varphi_k(x_n) \right)_{k,n=1}^{d,N}$ is formulated for a basis of discrete orthonormal polynomials $\{\varphi_k\}_{k=1}^d$ with respect to the discrete inner product $\langle \cdot, \cdot \rangle_{\underline{r}}$, we get $\left(\underline{\underline{V}}\sqrt{\underline{\underline{R}}} \right)^T \left(\underline{\underline{V}}\sqrt{\underline{\underline{R}}} \right) = \underline{\underline{I}}$ and the (modified) normal equation (41) reduces to

$$\underline{\omega}^* = \underline{\underline{R}}\,\underline{\underline{V}}^T \underline{m}. \tag{42}$$

The weights are hence given by

$$\omega_n^* = r_n \sum_{k=1}^d \varphi_k(x_n) \left(\int_a^b \varphi_k(x) \, \mathrm{d}\,\alpha(x) \right), \quad \text{for} \quad n = 1, \ldots, N. \tag{43}$$

Finally, we define the *discrete least squares quadrature rule* $L_{d,N}$ with order of exactness d and with respect to the discrete inner product $\langle f, g \rangle_{\underline{r}} = \sum_{n=1}^N r_n f(x_n) g(x_n)$ by

$$L_{d,N}(f) := \sum_{n=1}^N \omega_n^* f(x_n). \tag{44}$$

Note that the weights ω_n^* have been chosen such that the weighted least squares stability measure $\kappa_{\underline{r}}(\cdot)$ is minimal while maintaining order of exactness d. Stability for discrete least squares quadrature rules, in the sense of the original stability measure $\kappa(\underline{\omega}^*) = L(1)$, is not automatically guaranteed. In the next section, however, we will show that stability follows for appropriate inner products $\langle f, g \rangle_{\underline{r}} = \sum_{n=1}^N r_n f(x_n) g(x_n)$ once N is sufficiently large.

[3] I.e. with respect to the Euclidean inner product $\langle \cdot, \cdot \rangle_2$.

4 Stability of discrete least squares quadrature rules

In this section we address stability of discrete least squares quadrature rules. Our main result is Theorem 4.3, which provides a simple sufficient condition for discrete least squares quadrature rules to be stable.

Until now, such a result is only provided in Wilsons work [28], where it is shown that the construction of a stable discrete least squares quadrature rule[4] in the case of equidistant points, integrator $\alpha(x) = x$, and discrete inner product $\langle f, g \rangle = \frac{1}{N} \sum_{n=0}^{N-1} f(n)g(n)$ on $\mathbb{P}_{d-1}\left([0, N-1]\right)$ is essentially an $N \approx Cd^2$ process. Also in [18] stability of discrete least squares quadrature rules has been addressed, yet by somewhat questionable arguments.

We will investigate some of the conjectures formulated in [18] and utilise the insights from this investigation to derive new results for the whole class of discrete least squares quadrature rules. Furthermore, we investigate and compare different options for the discrete inner product $\langle \cdot, \cdot \rangle_{\underline{r}}$ in order to identify discrete least squares quadrature rules with enhanced stability properties, i.e. the discrete least squares quadrature rule to require a smaller number of quadrature points N to provide a fixed order of exactness d. It was already conjectured by Huybrechs [18] that 'convergence' might be increased by creating discrete inner products that converge faster to the continuous inner product. Yet, this work is the first to investigate this conjecture. In particular, subsection 4.2 will provide a comparative study of different discrete least squares quadrature rules on grids of equidistant as well as arbitrary points. As a result, we will be able to give clear recommendation what discrete least squares quadrature rule should be applied in a given situation.

4.1 A sufficient condition for stability

In this subsection, we will give sufficient conditions for discrete inner product $\langle f, g \rangle_{\underline{r}} = \sum_{n=1}^{N} r_n f(x_n) g(x_n)$ to yield a stable quadrature rule for sufficiently large N. It should be stressed once more that discrete least squares quadrature rules have been constructed in subsection 3.3 in order to provide a minimal weighted least squares stability measure $\kappa_{\underline{r}}(\cdot)$. Thus, stability of the quadrature rule - in the sense of $\kappa(\underline{\omega}) = L(1)$ to hold - is not automatically guaranteed.

Referring to Lemma 2.1, stability of a quadrature rule is equivalent to the weights to be non-negative. Our aim thus is to investigate under what conditions the weights ω_n^*, see (43), of a discrete least squares quadrature rule $L_{d,N}$ are ensured to become non-negative. By denoting

$$\varepsilon_k = \left(\sum_{n=1}^{N} r_n \varphi_k(x_n) - \int_a^b \varphi_k(x) \, \mathrm{d}\, \alpha(x) \right), \quad k = 1, \ldots, d, \tag{45}$$

the n-th discrete least squares weight ω_n^* can be written as

$$\omega_n^* = r_n \sum_{k=1}^{d} \varphi_k(x_n) \left(\langle 1, \varphi_k(x) \rangle_{\underline{r}} - \varepsilon_k \right) = r_n \left(\varphi_1(x_n) - \sum_{k=1}^{d} \varphi_k(x_n) \varepsilon_k \right). \tag{46}$$

Here, the basis $\{\varphi_k\}_{k=1}^d$ consists of discrete orthogonal polynomials with respect to the discrete inner product $\langle \cdot, \cdot \rangle_{\underline{r}}$. Positivity of the weights is thereby equivalent to the condition

$$\varphi_1(x_n) > \sum_{k=1}^{d} \varphi_k(x_n) \varepsilon_k \tag{47}$$

to hold for $n = 1, \ldots, N$. Further note that φ_1 is a constant function, which however might depend on N due to the discrete inner product to vary with N. In general, the constant value of the first discrete orthonormal polynomial is given by

$$\varphi_1(x) = \frac{1}{\|1\|_{\underline{r}}}. \tag{48}$$

We summarise this in

Lemma 4.1. *Let L be a linear functional on $C[a,b]$ with integrator $\alpha \in BV[a,b]$ and let $\left(\langle \cdot, \cdot \rangle_{\underline{r}} \right)_{N \in \mathbb{N}}$ be a sequence of discrete inner products with weights $\underline{r} \in \mathbb{R}^N$ at points $\underline{x} \in [a,b]^N$ respectively. Let $\left(L_{d,N} \right)$ further*

[4] Note that Wilson refers to discrete least squares quadrature rules as *nearest point formulas* in [28]

be the sequence of corresponding discrete least squares quadrature rules with weights $(\underline{\omega}^*)_{N\in\mathbb{N}}$ *given by* (43). *Then, the discrete least squares quadrature* $L_{d,N}$ *is stable if and only if*

$$\|1\|_{\underline{r}} \left(\sum_{k=1}^{d} \varphi_k(x_n)\varepsilon_k \right) < 1 \tag{49}$$

holds for $n = 1, \ldots, N$.

Remark 4.2. If every inner product $\langle \cdot, \cdot \rangle_{\underline{r}}$ induces a quadrature rule with order of exactness $\tilde{d}$, condition (49) becomes

$$\sqrt{L(1)} \left(\sum_{k=\tilde{d}+1}^{d} \varphi_k(x_n)\varepsilon_k \right) < 1, \tag{50}$$

since $\varepsilon_k = 0$ for $k = 1, \ldots, \tilde{d}$ in this case.

More general, we can also observe that the left hand side of (49) converges to zero once the discrete inner products converge to the continuous inner product[5] , due to

$$\varepsilon_k = \langle 1, \varphi_k \rangle_{\underline{r}} - \langle 1, \varphi_k \rangle_{\alpha} \to 0 \tag{51}$$

to hold as well as $\|1\|_{\underline{r}}$ and $\varphi_k(x_n)$ to be uniformly bounded. This observation yields our main result, which provides a sufficient condition for discrete least squares quadrature rules to be stable for a sufficiently large number of quadrature points N.

Theorem 4.3. *Let* L *be a linear functional on* $C[a,b]$ *with integrator* $\alpha \in BV[a,b]$ *and let* $\left(\langle \cdot, \cdot \rangle_{\underline{r}} \right)_{N\in\mathbb{N}}$ *be a sequence of discrete inner products with weights* $\underline{r} \in \mathbb{R}^N$ *at points* $\underline{x} \in [a,b]^N$ *respectively such that*

$$\sum_{n=1}^{N} r_n f(x_n) g(x_n) = \langle f, g \rangle_{\underline{r}} \to \langle f, g \rangle_{\alpha} = \int_a^b f(x)g(x) \,\mathrm{d}\,\alpha(x) \quad \forall f, g \in \mathbb{P}_{d-1}. \tag{52}$$

Let $\left(L_{d,N} \right)$ *further be the sequence of corresponding discrete least squares quadrature rules with weights* $(\underline{\omega}^*)_{N\in\mathbb{N}}$ *given by* (43). *Then there exists an* $N_{min} \in \mathbb{N}$ *such that all discrete least squares quadrature* $L_{d,N}$ *with* $N \geq N_{min}$ *are stable.*

Proof. First note that by Lemma 4.1 it is sufficient to show the existence of an $N_{\min} \in \mathbb{N}$ for which

$$\|1\|_{\underline{r}} \left(\sum_{k=1}^{d} \varphi_k(x_n)\varepsilon_k \right) < 1 \tag{53}$$

holds for all $N \geq N_{\min}$. Therefor let $\varepsilon > 0$. By condition (52), then there exists an $N_{\min} = N_{\min}(\varepsilon)$ such that

$$\varepsilon_k < \varepsilon \tag{54}$$

for $k = 1, \ldots, d$. At the same time, we have

$$\|1\|_{\underline{r}}^2 = \langle 1, 1 \rangle_{\underline{r}} \to \langle 1, 1 \rangle_{\alpha} = L(1), \quad \text{and} \quad \left|\varphi_k(x)\right| \to \left|\varphi_k^{\alpha}(x)\right| \ \forall x \in [a,b], \tag{55}$$

where $\{\varphi_k^{\alpha}\}$ denotes the corresponding basis of (continuous) orthogonal polynomials with respect to the (continuous) inner product $\langle \cdot, \cdot \rangle_{\alpha}$. In particular, there exist constants $B_1, B_2 > 0$ such that $\|1\|_{\underline{r}} < B_1$ and $\left|\varphi_k(x)\right| < B_2$ for all $N \in \mathbb{N}$, $k = 1, \ldots, d$, and $x \in [a,b]$. To sum up, we thus have

$$\|1\|_{\underline{r}} \left(\sum_{k=1}^{d} \varphi_k(x_n)\varepsilon_k \right) < B_1 d B_2 \varepsilon \tag{56}$$

for $N \geq N_{\min}$. The assertion follows from choosing $\varepsilon = \frac{1}{B_1 d B_2}$. □

[5] Please note that, strictly speaking, $\langle f, g \rangle_{\alpha} = \int_a^b f(x)g(x) \,\mathrm{d}\,\alpha(x)$ just induces an inner product when α is strictly monotonically increasing almost everywhere.

We end this subsection by a first demonstration on how large N has to be chosen for discrete least squares quadrature rules to be stable in practical applications. In Figure 5, the stability value κ is illustrated for a fixed order of exactness $d = 50$ and an increasing number of quadrature points N on $[0, 1]$. Thereby, the straight (blue) line corresponds to $\kappa(\underline{\omega}^*)$ for a discrete least squares rule resulting from the standard discrete inner product with weights $r_n = \frac{1}{N}$ on equidistant points. The dashed (red) line, on the other hand, corresponds to $\kappa(\underline{\omega}^*)$ for a discrete least squares rule resulting from the standard discrete inner product with weights $r_n = \frac{1}{N}$ on a set of randomly distributed points. In our implementation, this set of random points is constructed by iteratively adding a new randomly chosen point to the vector of already existing points.

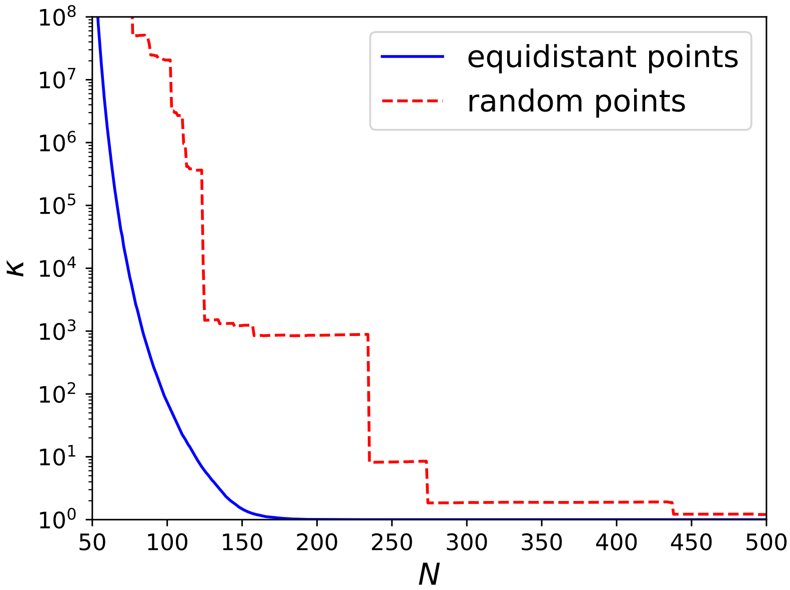


Figure 5: Stability value κ for order of exactness $d = 50$ and an increasing number of equidistant as well as randomly distributed points N.

Note that the discrete inner product $\langle \cdot, \cdot \rangle_{\underline{r}}$ for equidistant points converges to the continuous one. In accordance to Theorem 4.3, the resulting stability values κ thus show a formidable rate of convergence. Starting from $N = 157$, the corresponding least squares quadrature rules are all stable. Note the much slower convergence for randomly chosen points as displayed in Figure 5. It is important to note that condition (52) in Theorem 4.3 is not fulfilled by a standard discrete inner product on randomly distributed points; condition (52) is at most satisfied in a probabilistic sense. Yet, in practicle computations we are still able to handle this case. This is further demonstrated by Figure 6.

In Figure 6, the minimal values of N for stability to hold for the above discrete least squares quadrature rules are plotted against the desired order of exactness d. It is demonstrated by Figure 6 that the minimal value of N needed for a stable quadrature rule might be quite large, especially for randomly chosen points. In the next subsection, we thus aim to investigate the possibility to further reduce N by going over to other discrete inner products.

4.2 A comparative study

We now aim to reduce the number of quadrature points N needed for the discrete least squares quadrature rule $L_{d,N}$ to be stable. This will be achieved by comparing rules resulting from different discrete inner products, i.e. weights $\underline{r}$. We will first do so in the case of equidistant points and then for the case of randomly distributed points.

Taking a look back at the proof of Theorem 4.3, we note that the aim essentially has to be to reduce the bound ε in (54) for the errors

$$\varepsilon_k = \langle 1, \varphi_k \rangle_{\underline{r}} - \langle 1, \varphi_k \rangle, \quad k = 1, \dots, d. \tag{57}$$

Thus, the intend has to be to increase the convergence $\varepsilon_k \to 0$. There are essentially two ways to do so:

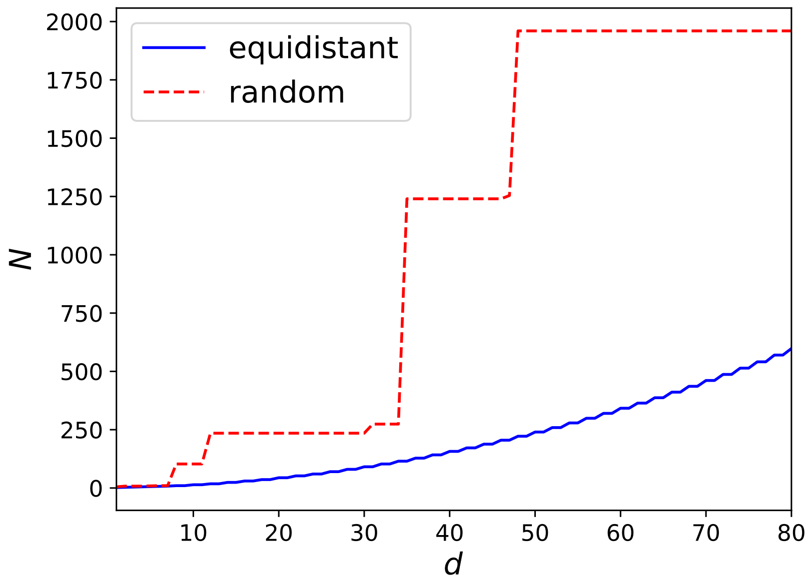


Figure 6: Minimal number of equidistant as well as randomly distributed points N for the discrete least squares quadrature rule corresponding to the standard discrete inner product to be stable.

1. Consulting Remark 4.2, the first $\tilde{d}$ errors $\varepsilon_1, \ldots, \varepsilon_{\tilde{d}}$ vanish completely if the discrete inner product induces a quadrature rule with order of exactness $\tilde{d}$.

2. The order of convergence for $\varepsilon_k \to 0$ can be increased by utilising discrete inner products with faster convergence to the continuous inner product.

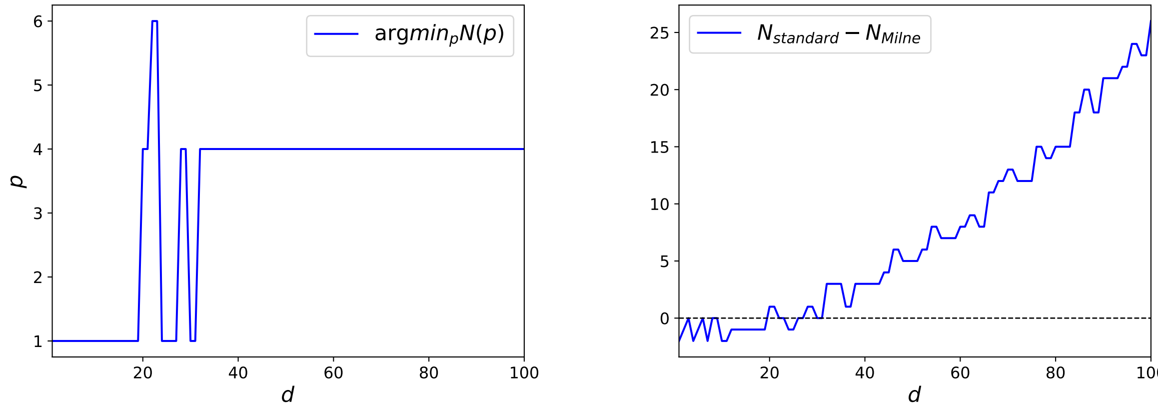
Both mechanisms are met by composite interpolatory quadrature rules [8], at least up to a certain degree. A comparison of the composite rules for the first eight Newton-Cotes rules on equidistant points is provided by Figure 7. Note that higher composite Newton-Cotes are not reasonable, since negative weights arise then. Here, the resulting discrete least squares quadrature rules are compared with respect to the minimal number of quadrature points that are required for the rule to be stable. In Figure 7a, the degree of the composite Newton-Cotes rule which provides the smallest number of points N in order to be stable is plotted against an increasing order of exactness d. It should be stressed that in our implementation, we set p to be the smallest degree, if multiple composite Newton-Cotes rules provide the same number N. As shown by Figure 7b, this is for instance the case for the composite Newton-Cotes rules of degree one and four. When having the choice between different discrete inner product providing the same advantage, we think it is preferable to decide for the most simple one.

We note from Figure 7a that not just the pure order of convergence of the discrete inner product is vital, but also the constants $c_k f^{2k}(\xi)$ and $d_k f^{2k}(\xi)$ in (11) and (12), which might become fairly large for higher order polynomials in the bases $\{\varphi_k\}_{k=1}^d$. As a result, the composite 1-point Newton-Cotes rule (i.e. the usual Riemann sum or standard discrete inner product) as well as the composite 4-point Newton-Cotes rule (also *Simpson's* 3/8 *rule*) are demonstrated to provide the best results, for almost all degrees d. The minimal numbers of quadrature points N required for the rules to be stable are further compared in Figure 7b. Here, the difference in N between the two rules is plotted against an increasing order of exactness d. A positive value means that the composite 4-point Newton-Cotes rule provides a smaller N, while a negative value means that the composite 1-point Newton-Cotes rule provides a smaller N. Lumping things together, we recommend to

- utilise the composite 1-point Newton-Cotes rule for the underlying discrete inner product for orders of exactness $d \leq 20$,

- utilise the composite 4-point Newton-Cotes rule for the underlying discrete inner product for all orders of exactness $d > 20$.

This comparison will be further investigated by numerical tests in section 5.



(a) Number of points p used in the composite Newton-Cotes rule which provides the smallest number of quadrature points N.

(b) Difference in the minimal number of points N between the standard discrete inner product and the composite 4-point Newton-Cotes rule.

Figure 7: A comparative study of discrete least squares quadrature rules resulting from different composite Newton-Cotes rules as discrete inner products.

We end this section by noting that for a grid of randomly distributed points only the standard inner product (composite 1-point Newton-Cotes rule) and the composite trapezoidal rule (composite 2-point Newton-Cotes rule) seem reasonable. Note that both rules just differ slightly at the end points. In accordance to this, Figure 8 illustrates only a slight difference between the minimal numbers of points N needed for the discrete least squares quadrature rules to be stable.

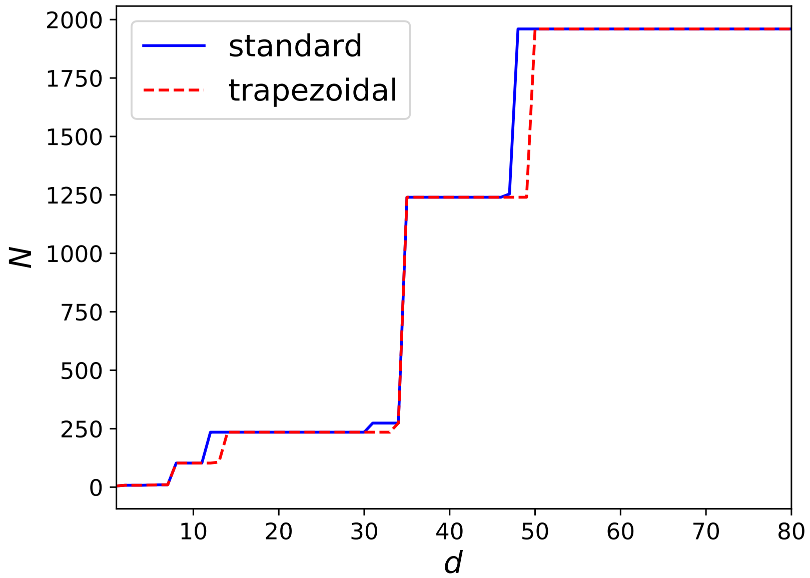


Figure 8: Minimal number of randomly distributed points N needed for the discrete least squares quadrature rules corresponding to the standard discrete inner product as well as the trapezoidal rule to be stable.

Yet, a small advantage can be observed for the trapezoidal rule. The advantage of using the composite trapezoidal rule will become more wide-ranging when comparing the accuracy in subsection 5.3.

5 Numerical tests

In this section, we demonstrate the prior theoretical investigations in several numerical tests. In subsection 5.1, we illustrate that stability essentially holds for $N \approx Cd^2$, as it was proven in [28] for the standard discrete inner product corresponding to the intergrator $\alpha(x) = x$. We demonstrate the same behaviour for further important classes of integrators and numerically determine the respective constants C. In subsection 5.2 and 5.3, we then investigate accuracy of discrete least squares quadrature rules on equidistnat as well as arbitrary points.

5.1 Minimal number of quadrature points for different integrators

We begin with demonstrating for some widely used integrators that stability of discrete least squares quadrature rules essentially is an d^2 process. In fact, in our numerical tests, we found stability to hold already for $N \approx Cd^2$ with fairly small constants $C < 1$ for all tested integrators. Figure 9 shows the number of points N needed for stability to hold for the standard discrete least squares quadrature rule and different integrators.

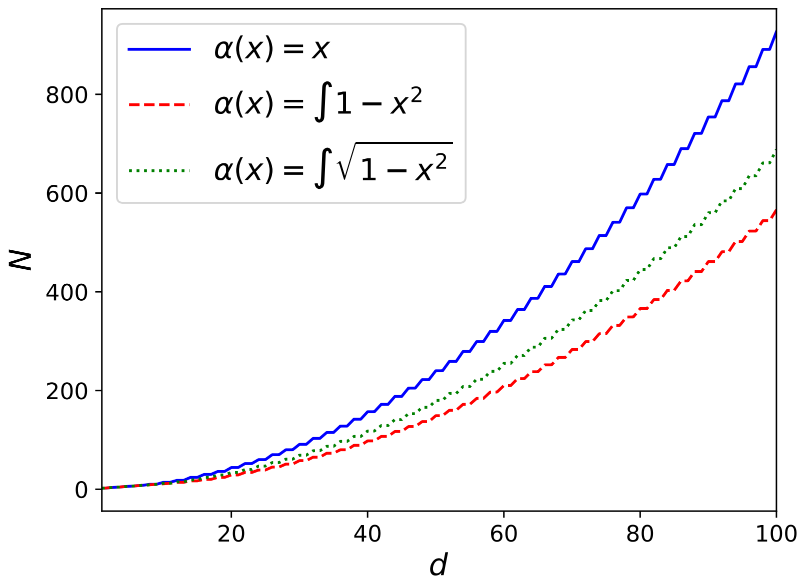


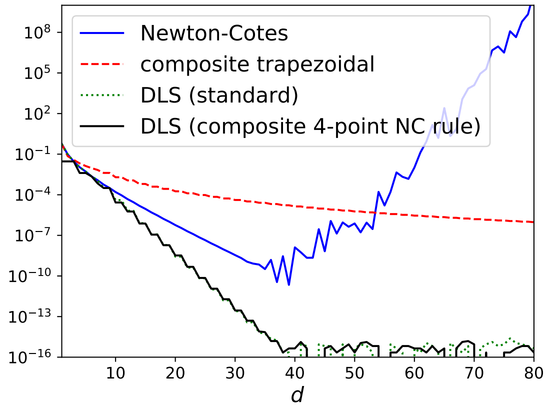
Figure 9: Minimal number of equidistant points N for the rules corresponding to the standard discrete inner products to be stable. Different integrators α are illustrated.

Thereby, the (blue) straight line corresponds to the usual integrator $\alpha(x) = x$ on $[-1, 1]$, i.e. the linear functional $L(f) = \int_{-1}^{1} f(x) \,\mathrm{d}\,x$. The (red) dashed line corresponds to the integrator $\alpha(x) = \int 1 - x^2 \,\mathrm{d}\,x$ on $[-1, 1]$, i.e. the linear functional $L(f) = \int_{-1}^{1} f(x) \left(1 - x^2\right) \mathrm{d}\,x$, and the (green) dotted line corresponds to the integrator $\alpha(x) = \int \sqrt{1 - x^2} \,\mathrm{d}\,x$ on $[-1, 1]$, i.e. the linear functional $L(f) = \int_{-1}^{1} f(x)\sqrt{1 - x^2} \,\mathrm{d}\,x$.

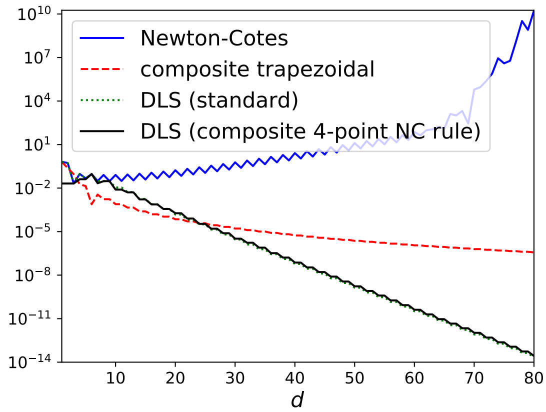
All three functions indicate that stability of the corresponding discrete least squares quadrature rule is a d^2 process, i.e. $N \approx Cd^2$ is sufficient for stability to hold. This can be examined by fitting the parameters C and s in the model $N = Cd^s$ in the sense of least squares. For integrator $\alpha(x) = x$ this least squares fit yields $C \approx 0.1, s \approx 1.95$ for $\alpha(x) = x$, while for $\alpha(x) = \int 1 - x^2$ and $\alpha(x) = \int \sqrt{1 - x^2}$ we have $C \approx 0.07, s \approx 1.93$ and $C \approx 0.8, s \approx 1.94$, respectively. Note that this is in accordance with Wilson's work [28], where he has shown that at least stability for the standard discrete least squares quadrature rule for integrator $\alpha(x) = x$ essentially is an d^2 process. By the present work, we gather strong evidence for this rate to also hold for a significant broader class of integrators. Here, a field of open problems for the analytical investigation of (non-standard) discrete least squares rules for more general integrators arises. We look forward to further research on this.

5.2 Accuracy on equidistant points

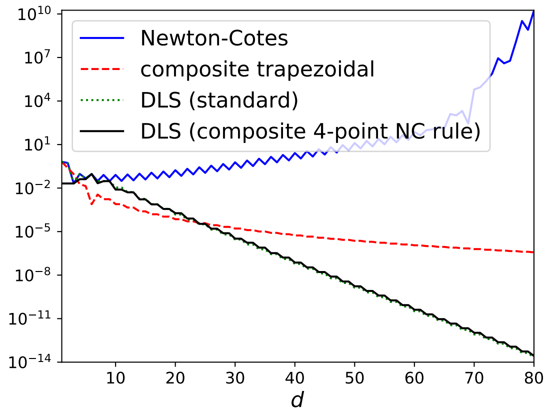
We now investigate accuracy of discrete least squares quadrature rules on equidistant points for $\alpha(x) = x$ on $[-1, 1]$. The errors of the standard discrete least squares quadrature rule and the rule for the composite 3/8 rule of Simpson are compared to the errors of the Newton-Cotes rule as well as the usual composite trapezoidal rule. Figure 10 shows the corresponding errors as a function with respect to d for a set of different functions f.



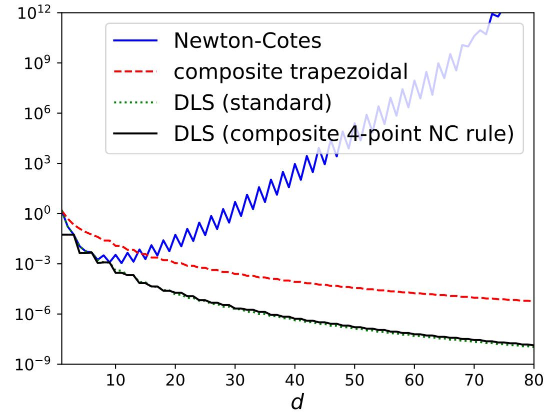
(a) $f(x) = \frac{1}{1+x^2}$



(b) $f(x) = \frac{1}{1+8x^2}$



(c) $f(x) = \cos(20x)$



(d) $f(x) = |x|^3$

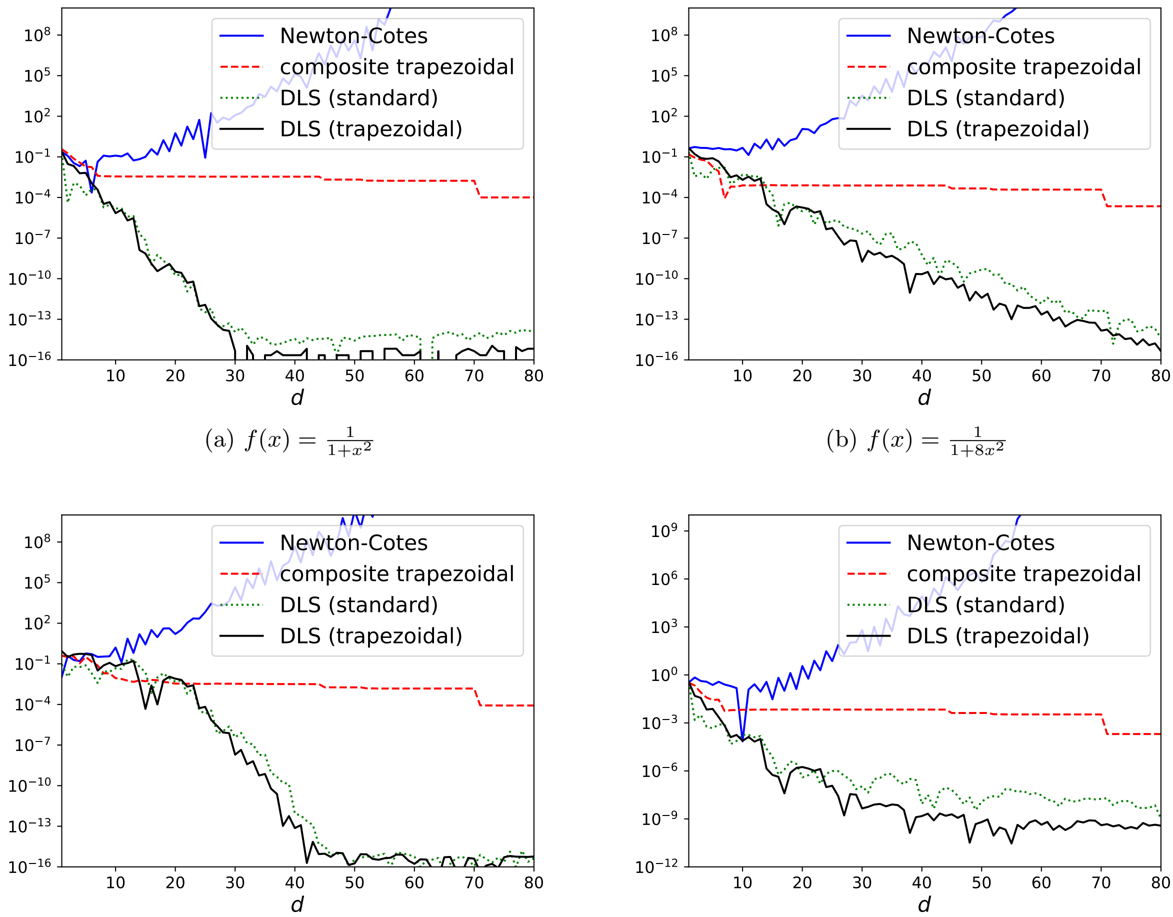
Figure 10: Errors for the Newton-Cotes and composite trapezoidal rule as well as for the standard and composite trapezoidal discrete least squares quadrature rules. Equidistant points on $[-1, 1]$

Here, the errors for the Newton-Cotes rule are illustrated by the (blue) straight line, while the (red) dashed line corresponds to composite trapezoidal rule. The errors of the standard discrete least squares quadrature rule are further illustrated by the (green) dotted line and the errors of the discrete least squares quadrature rule for the composite 3/8-rule of Simpson are illustrated by the (black) straight line. For the number of equidistant quadrature points we set $N = d$ for the Newton-Cotes rule and the minimal number of points needed for stability for both discrete least squares quadrature rules, respectively. To allow a fair comparison with the discrete least squares quadrature rules, we have chosen the number of quadrature points for the composite trapezoidal rule to be equal to the number of points for the standard discrete least squares quadrature rule.

We start by once more noting that the Newton-Cotes rule diverges in all tests, either from the start due to divergence of the polynomial interpolation, such as in Figure 10b and 10c, or due to numerical instabilities, such as in Figure 10a and 10d. The composite trapezoidal rule is demonstrated to converge in all four cases, even though the convergence is fairly slow. At the same time, both discrete least squares quadrature rules show formidably fast convergence in all test cases. Furthermore, they provide the smallest errors from the start in Figure 10a and 10a, and at least the smallest errors for degrees $d \geq 25$ in 10b and 10d. Note that in Figure 10a both discrete least squares quadrature rules reach machine precision, i.e. $2^{-53} \approx 1.1 \cdot 10^{-16}$ for double precision, already for $d = 40$. In our tests we were not able to observe noticeable differences in the accuracy of the standard and 3/8-rule discrete least squares quadrature rule. But note that by Figure 7 in subsection 4.2 the 3/8-rule discrete least squares quadrature rule has been shown to provide the same order of exactness as well as accuracy for a smaller number of quadrature points. In fact, it should be stressed once more that both discrete least squares quadrature rule do not just provide a remarkably faster convergence in all test cases, but are also able - in contrast to the composite trapezoidal rule - to exactly integrate polynomials up to degree $d-1$.

5.3 Accuracy on arbitrary points

We now perform the same analysis as in subsection 5.2 for a grid of arbitrary quadrature points. Figure 11 illustrates the errors in the same manner as Figure 10. This time, the discrete least squares quadrature rule for the composite trapezoidal rule on arbitrary points is included instead of the 3/8-rule discrete least squares quadrature rule.



(a) $f(x) = \frac{1}{1+x^2}$ (b) $f(x) = \frac{1}{1+8x^2}$

(c) $f(x) = \cos(20x)$ (d) $f(x) = |x|^3$

Figure 11: Errors for the Newton-Cotes and composite trapezoidal rule as well as for the standard and composite trapezoidal discrete least squares quadrature rules. Arbitrary points on $[-1, 1]$.

For a fair comparison, all rules are applied to the same number of $N = d^2$ randomly chosen points in $[-1, 1]$. As demonstrated in Figure 11, the Newton-Cotes rule again diverges in all test cases. The composite trapezoidal rule still converges, but now even slower than in the case of equidistant points. We note that both discrete least squares quadrature rules still show an impressive rate of convergence, even though they are now applied to a set of randomly chosen points. In Figure 11a they even reach machine precision again, now already at $d = 30$.

This time using a more challenging grid of arbitrary points, we are further able to note a clear advantage for the (non-standard) trapezoidal discrete least squares quadrature rule compared to the standard discrete least squares quadrature rule. In all tests on grids of randomly chosen points, the trapezoidal discrete least squares quadrature rule provides higher orders of accuracy. Figure 11, for instance, demonstrates well that the trapezoidal discrete least squares quadrature rule provides results which are up to 10^{-3} more accurate.

6 Concluding thoughts

In this work, we revisited discrete least squares quadrature rules (DLS-QRs) introduced 1970 by Wilson in [27, 28]. Thereby, a sufficient condition for stability was proven for a significantly broader class of linear functionals and more general discrete least squares quadrature rules. We further compared different discrete inner products resulting in different rules and were able to demonstrate an advantage for building the discrete inner product on certain composite rules. This has been shown to be especially beneficial for grids of arbitrary points.

The resulting method has applications in creating stable high-order quadrature rules for data arising, for instance, from experimentation. Here, it may be impracticable - if not even impossible - to obtain measurements to fit known quadrature rules. An additional advantage is that known errors or uncertainties of the data can be incorporated by adapting the weights in the underlying discrete inner product.

Another promising field of application, which has not been fully explored yet, is the construction of stable numerical methods for partial differential equations [14]. Especially in Finite Difference and some recent Discontinuous Galerkin methods, high orders of exactness for quadrature rules are vital to mimic integration by parts on a discrete level, a property also known as *summation by parts*. The discrete least squares quadrature rules constructed in this work can thus as well be seen as a method to construct so called *summation by parts operators* - a hot topic in FD and DG methods [6, 10, 11, 24] - on any set of collocation points.

Acknowledgement. I would like to thank Heike Faßbender (TU Braunschweig, Germany) for helpful advice. Further, I would like to thank the Max Planck Institute for Mathematics (MPIM) Bonn for wonderful working conditions.

References

[1] A. Abdulle and G. Wanner. 200 years of least squares method. *Elemente der Mathematik*, 57(2):45–60, 2002.

[2] J. Bezanson, A. Edelman, S. Karpinski, and V. B. Shah. Julia: A fresh approach to numerical computing. *SIAM Review*, 59(1):65–98, 2017.

[3] J. P. Boyd. *Chebyshev and Fourier spectral methods*. Courier Corporation, 2001.

[4] H. Brass. Error estimates for least squares approximation by polynomials. *Journal of approximation theory*, 41(4):345–349, 1984.

[5] H. Brunner. *Collocation methods for Volterra integral and related functional differential equations*, volume 15. Cambridge University Press, 2004.

[6] T. Chen and C.-W. Shu. Entropy stable high order discontinuous galerkin methods with suitable quadrature rules for hyperbolic conservation laws. *Journal of Computational Physics*, 345:427–461, 2017.

[7] P. Davis. On a problem in the theory of mechanical quadrature. *Pacific Journal of Mathematics*, 5(5):669–674, 1955.

[8] P. J. Davis and P. Rabinowitz. *Methods of numerical integration*. Courier Corporation, 2007.

[9] G. Faber. Über die interpolatorische darstellung stetiger funktionen. *Jber. Deutsch. Math. Verein*, 23:192–210, 1914.

[10] D. C. D. R. Fernández, J. E. Hicken, and D. W. Zingg. Review of summation-by-parts operators with simultaneous approximation terms for the numerical solution of partial differential equations. *Computers & Fluids*, 95:171–196, 2014.

[11] G. J. Gassner. A skew-symmetric discontinuous galerkin spectral element discretization and its relation to sbp-sat finite difference methods. *SIAM Journal on Scientific Computing*, 35(3):A1233–A1253, 2013.

[12] C. F. Gauss and C. H. Davis. *Theory of the motion of the heavenly bodies moving about the sun in conic sections*. Courier Corporation, 2004.

[13] W. Gautschi. *Orthogonal polynomials: computation and approximation*. Oxford University Press on Demand, 2004.

[14] J. Glaubitz and P. Öffner. A novel discontinuous galerkin method using the principle of discrete least squares. *MPIM preprint series: 2017-63*, 2017. Submitted.

[15] R. Goertz and P. Öffner. Spectral accuracy for the hahn polynomials. *arXiv preprint arXiv:1609.07291*, 2016.

[16] H. H. Goldstine. *A History of Numerical Analysis from the 16th through the 19th Century*, volume 2. Springer Science & Business Media, 2012.

[17] G. H. Golub and C. F. Van Loan. *Matrix computations*, volume 3. JHU Press, 2012.

[18] D. Huybrechs. Stable high-order quadrature rules with equidistant points. *Journal of computational and applied mathematics*, 231(2):933–947, 2009.

[19] P. Lax. *Functional analysis*. Pure and applied mathematics. Wiley, 2002.

[20] M. Merriman. On the history of the method of least squares. *The Analyst*, 4(2):33–36, 1877.

[21] K. Morton and D. Mayers. *Numerical Solution of Partial Differential Equations: An Introduction*. Cambridge University Press, 2005.

[22] L. Reichel. On polynomial approximation in the uniform norm by the discrete least squares method. *BIT Numerical Mathematics*, 26(3):349–368, 1986.

[23] C. Runge. Über empirische funktionen und die interpolation zwischen äquidistanten ordinaten. *Zeitschrift für Mathematik und Physik*, 46(224-243):20, 1901.

[24] M. Svärd and J. Nordström. Review of summation-by-parts schemes for initial–boundary-value problems. *Journal of Computational Physics*, 268:17–38, 2014.

[25] A. E. Taylor and D. C. Lay. *Introduction to functional analysis*, volume 2. Wiley New York, 1958.

[26] L. N. Trefethen and D. Bau III. *Numerical linear algebra*, volume 50. Siam, 1997.

[27] M. W. Wilson. Discrete least squares and quadrature formulas. *Mathematics of Computation*, 24(110):271–282, 1970.

[28] M. W. Wilson. Necessary and sufficient conditions for equidistant quadrature formula. *SIAM Journal on Numerical Analysis*, 7(1):134–141, 1970.