

## Parametric integration by magic point empirical interpolation

MAXIMILIAN GAß

Center for Mathematics, Technische Universität München, Munich, Germany

KATHRIN GLAU\*

Center for Mathematics, Technische Universität München, Munich, Germany and School of Mathematical Sciences, Queen Mary University of London, London, UK

\*Corresponding author: k.glau@qmul.ac.uk

[Received on 30 November 2015; revised on 13 October 2017]

We derive analyticity criteria for explicit error bounds and an exponential rate of convergence of the magic point empirical interpolation method introduced by Barrault *et al.* (2004, An empirical interpolation method: application to efficient reduced-basis discretization of partial differential equations. *C. R. Math.*, **339**, 667–672). Furthermore, we investigate its application to parametric integration. We find that the method is well-suited to Fourier transforms and has a wide range of applications in such diverse fields as probability and statistics, signal and image processing, physics, chemistry and mathematical finance. To illustrate the method, we apply it to the evaluation of probability densities by parametric Fourier inversion. Our numerical experiments display convergence of exponential order, even in cases where the theoretical results do not apply. The magic point integration performs considerably well, when compared with Clenshaw–Curtis quadrature and with Chebyshev interpolation in the parameter space.

**Keywords:** parametric integration; Fourier transform; parametric Fourier inversion; magic point interpolation; empirical interpolation; online–offline decomposition.

### 1. Introduction

At the basis of a large variety of mathematical applications lies the computation of parametric integrals of the form

$$\mathcal{I}(h_p) := \int_{\Omega} h_p(z) \mu(dz) \quad \text{for all } p \in \mathcal{P}, \quad (1.1)$$

where  $\Omega \subset \mathbb{C}^d$  is a compact integration domain,  $\mathcal{P}$  a compact parameter set,  $(\Omega, \mathfrak{A}, \mu)$  a measure space with  $\mu(\Omega) < \infty$  and  $h : \mathcal{P} \times \Omega \rightarrow \mathbb{C}$  a bounded function.

A prominent class of such integrals are parametric *Fourier transforms*

$$\widehat{f}_q(z) = \int_{\Omega} e^{i\langle z, x \rangle} f_q(x) dx \quad \text{for all } p = (q, z) \in \mathcal{P}, \quad (1.2)$$

with a parametric family of complex functions  $f_q$  defined on the compact set  $\Omega$ . Today, Fourier transforms lie at the heart of applications in optics, electrical engineering, chemistry, probability, partial differential equations, statistics and finance. The application of Fourier transforms even sparked many advancements in those disciplines, underlining its impressive power. The introduction of Fourier analysis to image

formation theory in Duffieux (1946) marked a turning point in optical image processing, as outlined, for instance, in Stark (1982). The application of Fourier transform to nuclear magnetic resonance (NMR) in Ernst & Anderson (1966) was a major breakthrough in increasing the sensitivity of NMR, for which Richard R. Ernst was awarded the Nobel prize in Chemistry in 1991. Fourier analysis also plays a fundamental role in statistics, in particular in the spectral representation of time series of stochastic data (see Brockwell & Davis, 2002). For an elucidation of other applications impacted by Fourier theory, we recommend (Kammler, 2007, Appendix 1).

For parametric Fourier integrals of form (1.2) with fixed value of  $q$  and  $z$  being the only varying parameter, efficient numerical methods have been developed based on discrete Fourier transform (DFT) and fast Fourier transform (FFT). The latter was developed by Cooley & Tukey (1965) to obtain the Fourier transform  $\hat{f}(z)$  for a large set of values  $z$  simultaneously. With regard to (1.2), this is the special case of a parametric Fourier transform, where  $q$  is fixed and  $z$  varies over a specific set. The immense impact of FFT highlights the exceptional usefulness of efficient methods for parametric Fourier integration.

Shifting the focus to other examples, parametric integrals arise as generalized moments of parametric distributions in probability, statistics, engineering and computational finance. These expressions often appear in optimization routines, where they have to be computed for a large set of different parameter constellations. Prominent applications are model calibration and statistical learning algorithms based on moment estimation, regression and expectation–maximization (see, e.g., Hastie *et al.*, 2009).

The efficient computation of parametric integrals is also a cornerstone of the quantification of parameter uncertainty in generalized moments of form (1.1), which leads to an integral of the form

$$\int_{\mathcal{P}} \int_{\Omega} h_p(z) \mu(dz) P(dp), \quad (1.3)$$

where  $P$  is a probability distribution on the parameter space  $\mathcal{P}$ .

In all of these cases, parametric integrals of the form (1.1) have to be determined for a large set of parameter values. Therefore, efficient algorithms for parametric integration have a wide range of applicability. In the context of reduced basis methods for parametric partial differential equations, a magic point empirical interpolation method has been developed by Barrault *et al.* (2004) to treat nonlinearities that are expressed by parametric integrals. The applicability of this method to the interpolation of parametric functions has been demonstrated by Maday *et al.* (2009).

In this article, we focus on the approximation of parametric integrals by magic point empirical interpolation in general, a method that we call *magic point integration* and

- provide sufficient conditions for an *exponential rate of convergence* of magic point interpolation and integration in its degrees of freedom, accompanied by *explicit error bounds*;
- translate these conditions to the special case of parametric Fourier transforms, which shows the broad scope of the results;
- empirically demonstrate efficiency of the method in a numerical case study;<sup>1</sup> and
- compare its performance with the Clenshaw–Curtis quadrature and with Chebyshev interpolation in the parameter space.

<sup>1</sup> An in-depth case study related to finance is presented in Gaß *et al.* (2017) and supports our empirical findings.

### 1.1 Sketch of the main idea

To sketch the idea, let us start with an observation that is as simple as it is crucial. If the interpolation  $I_M(f) := \sum_{m=1}^M \beta_m^M e_m^M(z) \approx f(z)$  with coefficients  $\beta_m^M \in \mathbb{C}$  and functions  $e_m^M : \Omega \rightarrow \mathbb{C}$  is a good approximation of  $f$  in  $L^\infty(\Omega)$  in terms of stability and order of convergence, then

$$\mathcal{I}_M(f)(z) := \sum_{m=1}^M \beta_m^M \int_{\Omega} e_m^M(z) \mu(dz) \quad (1.4)$$

is a good approximation for  $\int_{\Omega} f(z) \mu(dz)$  in the sense that few number of summands  $M$  are sufficient to obtain a high accuracy. An approximation of type (1.4), however, is only *numerically efficient*, if this is the case for the computation of both the coefficients  $\beta_m^M$  and the weights  $w_m^M := \int_{\Omega} e_m^M(z) \mu(dz)$ .

In classical quadrature rules, the interpolation  $I_M(f)$  is a polynomial interpolation, for instance, the Chebyshev interpolation of  $f$ , which results in the well-known Clenshaw–Curtis quadrature rule. In this case, the Chebyshev coefficients can be easily and efficiently computed, and, thirdly the basis functions  $e_m$  are polynomials whence the weights  $w_m^M$  are known explicitly. We notice that classical quadrature rules yield the same quadrature nodes and basis functions for all integrands  $f$  on a given domain  $\Omega$ .

Yet as we often have more insight into our integration problem, namely that the integrand stems from a particular set  $\{h_p | p \in \mathcal{P}\}$ , it is natural to ask: *How can we systematically tailor a quadrature rule to a specific set of integrands?* The conceptual difficulty arises from the fact that we want to obtain a method that autonomously finds and exploits the specific features of the given set of integrands. Therefore, we propose a two-step procedure in which the first step is a learning algorithm.

Here, we choose the magic point interpolation method of [Barraut et al. \(2004\)](#). In a recursive greedy procedure, it delivers a nodal interpolation of functions tailored a given set  $\{h_p | p \in \mathcal{P}\}$ . The learning procedure after  $M \in \mathbb{N}$  steps delivers both the *magic points*  $z_1^*, \dots, z_M^*$  and the functions  $\theta_1^M, \dots, \theta_M^M$  on the basis of which the *magic point interpolation* operator

$$I_M(h)(p, z) := \sum_{m=1}^M h_p(z_m^*) \theta_m^M(z) \quad (1.5)$$

is constructed to approximate all functions from the set  $\{h_p | p \in \mathcal{P}\}$  particularly well in the  $L^\infty$ -norm. This allows us to define the **magic point integration** operator

$$\mathcal{I}_M(h)(p) := \sum_{m=1}^M h_p(z_m^*) \int_{\Omega} \theta_m^M(z) \mu(dz). \quad (1.6)$$

This integration routine inherits its favourable numerical properties from the respective properties of the magic point interpolation. We will discuss convergence results in detail. To efficiently employ operator (2.2) for numerical integration, we split the procedure in an *offline* and an *online phase*.

In the offline phase, we iteratively construct the magic points  $z_1^*, \dots, z_M^*$  along with the functions  $\theta_1^M, \dots, \theta_M^M$ , thus the operator (1.5). Moreover, we precompute all weights  $w_m^M := \int_{\Omega} \theta_m^M(z) dz$ .

The online phase consist in evaluating the sum

$$\sum_{m=1}^M h_p(z_m^*) w_m^M. \quad (1.7)$$

## 1.2 Related literature

Another algorithm for parametric integration has been proposed by Antil *et al.* (2013). Their two-step greedy algorithm for reduced order quadratures is similar to the algorithm we propose, since it also divides the problem into an offline and an online phase, where an essential part of the offline phase consists in an application of an empirical interpolation. In contrast to the method we propose here, the authors explicitly abstain from applying the original magic point interpolation method of Barrault *et al.* (2004), and the resulting algorithm of Antil *et al.* (2013) significantly differs from magic point integration. The main reason is that they consider parametric scalar products of functions in a Hilbert space to obtain a reduced basis. Their starting point thus is the representation of the integrands in an orthonormal basis of a Hilbert space. Following the spirit of reduced order modelling, they first consider the projection to a large but finite dimensional subspace. In the first step of the offline phase, a reduced basis of the approximation space is generated. The second step of the offline phase is a greedy procedure, where the discrete empirical interpolation (DEIM) extracts empirical integration points from the reduced basis.

Philosophically, both approaches differ as well: Antil *et al.* (2013) employ a two-stage procedure. First, the integration problem is discretized. Secondly, the discrete problem is reduced by exploiting additional knowledge. In contrast, the magic point integration that we propose systematically exploits insight into the infinite dimensional integration problem directly.

Similarly to the procedure of Antil *et al.* (2013), low-rank tensor decompositions for parameter-dependent integrals are constructed in two steps, first employing a discretization of choice, and second reducing the tensor structure (see Ballani, 2012; the survey article by Grasedyck *et al.*, 2013 and the further references therein). The two approaches, empirical interpolation and low-rank tensor decompositions, are closely interrelated as discussed in Bebendorf *et al.* (2014). In contrast to these approaches, magic point integration uses the actual function set of interest as an input into the learning procedure. No *a priori* discretization method is involved. Instead, it is the learning procedure itself that delivers the discretization. Moreover, the article Bebendorf *et al.* (2014) discusses an interesting and close relation between magic point interpolation and the adaptive cross approximation. This relation can be directly transferred to our framework, i.e., when we compare our ansatz relying on the magic point interpolation of the parametric integrands to the use of the adaptive cross approximation of the integrands instead.

Monte Carlo methods for parametric integrals have been developed and investigated in Heinrich (1998), Heinrich & Sindambiwe (1999), by introducing the multilevel Monte Carlo method. The key idea of this approach is to combine interpolation of the integrand in the parameter space with Monte Carlo integration in a hierarchical way. In contrast to our approach, the interpolation is known and fixed beforehand and, thus, is not tailored to the parametric family of integrands, which is the key feature of magic point integration.

The remainder of the article is organized as follows. In Section 2, we revisit the magic point empirical interpolation method from Barrault *et al.* (2004), related error bounds and present the related integral approximation, which can be perceived from two different perspectives. On the one hand, it delivers a *quadrature rule for parametric integrals* and, on the other, an *interpolation method for parametric integrals* in the parameter space. In Section 3, we provide analyticity conditions on the parametric integrals that imply exponential order of convergence of the method. We focus on the case of parametric Fourier transforms in the subsequent Section 4. In a case study, we discuss the numerical implementation and its results in Section 5. Here, we use the magic point integration method to evaluate densities of a parametric class of distributions that are defined through their Fourier transforms.

## 2. Magic point empirical interpolation for integration

We now introduce the *magic point empirical interpolation method for parametric integration* to approximate parametric integrals of the form (1.1). Before we closely follow Maday *et al.* (2009) to describe the interpolation method, let us state our basic assumptions that ensure the well-definedness of the iterative procedure.

**ASSUMPTION 2.1** Let  $(\Omega, \|\cdot\|_\infty)$  and  $(\mathcal{P}, \|\cdot\|_\infty)$  be compact,  $\mathcal{P} \times \Omega \ni (p, z) \mapsto h_p(z)$  bounded and  $p \mapsto h_p$  be continuous, i.e., for every sequence  $p_i \rightarrow p$  we have  $\|h_{p_i} - h_p\|_\infty \rightarrow 0$ . Moreover, there exists  $p \in \mathcal{P}$  such that the function  $h_p$  is not constantly zero.

For  $M \in \mathbb{N}$ , the method delivers iteratively the *magic points*  $z_1^*, \dots, z_M^*$ , functions  $\theta_1^M, \dots, \theta_M^M$  and constructs the *magic point interpolation operator*

$$I_M(h)(p, z) := \sum_{m=1}^M h_p(z_m^*) \theta_m^M(z), \quad (2.1)$$

which allows us to define the **magic point integration operator**

$$\mathcal{I}_M(h)(p) := \sum_{m=1}^M h_p(z_m^*) \int_{\Omega} \theta_m^M(z) \, dz. \quad (2.2)$$

In the first step,  $M = 1$ , we define the first *magic parameter*  $p_1^*$ , the first *magic point*  $z_1^*$  and the first *basis function*  $q_1$  by

$$p_1^* := \arg \max_{p \in \mathcal{P}} \|h_p\|_{L^\infty}, \quad (2.3)$$

$$z_1^* := \arg \max_{z \in \Omega} |h_{p_1^*}(z)|, \quad (2.4)$$

$$q_1(\cdot) := \frac{h_{p_1^*}(\cdot)}{h_{p_1^*}(z_1^*)}. \quad (2.5)$$

Notice that, thanks to Assumption 2.1, these operations are well-defined.

For  $M \geq 1$  and  $1 \leq m \leq M$  we define

$$\theta_m^M(z) := \sum_{j=1}^M (B^M)_{jm}^{-1} q_j(z), \quad B_{jm}^M := q_m(z_j^*), \quad (2.6)$$

where we denote by  $(B^M)_{jm}^{-1}$  the entry in the  $j$ th line and  $m$ th column of the inverse of matrix  $B^M$ . By construction,  $B^M$  is a lower triangular matrix with unity diagonal and is thus invertible (see Barrault *et al.*, 2004).

Then, recursively, as long as there are at least  $M$  linearly independent functions in  $\{h_p \mid p \in \mathcal{P}\}$ , the algorithm chooses the next magic parameter  $p_M^*$  according to a greedy procedure. It selects the parameter

so that  $h_{p_M^*}$  is the function in the set  $\{h_p \mid p \in \mathcal{P}\}$  which is worst represented by its approximation with the previously identified  $M - 1$  magic points and basis functions. So the  $M$ th magic parameter is

$$p_M^* := \arg \max_{p \in \mathcal{P}} \|h_p - I_{M-1}(h)(p, \cdot)\|_{L^\infty}. \quad (2.7)$$

In the same spirit, select the  $M$ th magic point as

$$z_M^* := \arg \max_{z \in \Omega} |h_{p_M^*}(z) - I_{M-1}(h)(p_M^*, z)|. \quad (2.8)$$

The  $M$ th basis function  $q_M$  is the residual  $r_M$ , normed to 1 when evaluated at the new magic point, i.e.,

$$r_M(z) := h_{p_M^*}(z) - I_{M-1}(h)(p_M^*, z), \quad (2.9)$$

$$q_M(\cdot) := \frac{r_M(\cdot)}{r_M(z_M^*)}. \quad (2.10)$$

Notice the well-definedness of the operations in the iterative step, thanks to Assumption 2.1, and the fact that the denominator in (2.10) is zero only if all functions in  $\{h_p \mid p \in \mathcal{P}\}$  are perfectly represented by the interpolation  $I_{M-1}$ , in which case they span a linear space of dimension  $M - 1$  or less.

Let us understand the approach from two different perspectives. On the one hand, it is an interpolation method:

**REMARK 2.2** (Interpolation of parametric integrals in the parameters) For each  $m = 1, \dots, M$ , the function  $\theta_m^M$  is a linear combination of snapshots  $h_{p_j^*}$  for  $j = 1, \dots, M$  with coefficients  $\beta_j^m$ , which may be iteratively computed. We thus may rewrite formula (2.2) as

$$\mathcal{I}_M(h)(p) = \sum_{m=1}^M h_p(z_m^*) \sum_{j=1}^M \beta_j^m \int_{\Omega} h_{p_j^*}(z) \, dz. \quad (2.11)$$

Thus, for an arbitrary parameter  $p \in \mathcal{P}$ , the integral  $\int_{\Omega} h_p(z) \, dz$  is approximated by a linear combination of integrals  $\int_{\Omega} h_{p_j^*}(z) \, dz$  for the magic parameters  $p_j^*$ . In other words,  $\mathcal{I}_M(h)$  is an interpolation of the function  $p \mapsto \int_{\Omega} h_p(z) \, dz$ . Here, the interpolation nodes are the magic parameters  $p_j^*$  and the coefficients are given by  $\sum_{m=1}^M h_p(z_m^*) \beta_j^m$ . In contrast to classic interpolation methods, the ‘magic nodes’ are tailored to the parametric set of integrands.

On the other hand, magic point integration is an approximate integration rule:

**REMARK 2.3** (Empirical quadrature rule for integrating parametric functions) Magic point integration is an interpolation method for integrating a parametric family of integrands over a compact domain. From this point of view, the magic point empirical interpolation of Barrault *et al.* (2004) provides a *quadrature rule for integrating parametric functions*. The weights are  $\int_{\Omega} \theta_m^M(z) \, dz$  and the nodes are the magic points  $z_m^*$  for  $m = 1, \dots, M$ . As in the adaptive quadrature rules, these quadrature nodes are not fixed in advance. Although the adaptive quadrature rules iteratively determine the next quadrature node in view

of integrating a single function, the *empirical integration method* (2.2) tailors itself to the integration of a given *parametrized family of integrands*.

A discrete approach to empirical interpolation has been introduced by Chaturantabut & Sorensen (2010). Although the empirical interpolation is designed for parametric functions, the input data for DEIM is discrete. A canonical way to use DEIM for integration is to first choose a fixed grid in  $\Omega$  for a discrete integration of all parametric integrands. Then, DEIM can be performed on the integrands evaluated on this grid.

In contrast, using magic point empirical interpolation for parametric integration separates the choice of the integration grid from the selection of nodes  $p_m^*$ . For each function, a different integration discretization may be used. Indeed, in our numerical study, we leave the discretization choices regarding integration to Matlab's `quadgk` routine. Fixing a grid for discrete integration beforehand might, e.g., become advantageous when the domain of integration  $\Omega$  is high dimensional.

### 3. Convergence analysis of magic point integration

We therefore summarize the results the convergence results for the magic point interpolation in the  $L^\infty$ -norm, which clearly extend to the convergence results for the magic point integration method. Interestingly, these results relate the convergence of the magic point interpolation to the best linear  $n$ -term approximation that is formally expressed by the Kolmogorov  $n$ -width, consider the monograph of Pinkus (1985) for a general reference. For a real or complex normed linear space  $(\mathcal{X}, \|\cdot\|)$  and  $\mathcal{U} \subset \mathcal{X}$ , the *Kolmogorov  $n$ -width* is given by

$$d_n(\mathcal{U}, \mathcal{X}) = \inf_{\mathcal{U}_n \in \mathcal{E}(\mathcal{X}, n)} \sup_{g \in \mathcal{U}} \inf_{f \in \mathcal{U}_n} \|g - f\|, \quad (3.1)$$

where  $\mathcal{E}(\mathcal{X}, n)$  is the set of all  $n$ -dimensional subspaces of  $\mathcal{X}$ .

This concept allows to express the convergence behaviour of the interpolation subject to the ‘true complexity’ of set of functions of interest, if one agrees that the Kolmogorov  $n$ -width is a good measure of the intrinsic complexity. The approach is especially worthwhile when tackling the curse of dimensionality in the parameter or in the integration space.

In the following section, we present *a posteriori* and *a priori* error bounds, mostly from the literature, which are relevant for the integration method proposed. To obtain precise convergence rates, the Kolmogorov  $n$ -width needs to be estimated. We devote Section 3.2 below to this task and obtain widely applicable sufficient conditions for an exponential convergence of the magic point integration method. For univariate parametric Fourier integrals, the conditions boil down to exponential moment conditions.

#### 3.1 Error bounds for the magic point interpolation

We first present an *a posteriori error bound*, which follows as a corollary from the results in Maday *et al.* (2016). Let  $\mathcal{U} := \{h_p \mid p \in \mathcal{P}\}$  and  $(\mathcal{X}, \|\cdot\|) = (L^\infty(\Omega), \|\cdot\|_\infty)$ . Moreover, the convergence statement involves the Lebesgue constant

$$\Lambda_M := \sup_{z \in \Omega} \sum_{m=1}^M |\theta_m^M(z)|, \quad (3.2)$$

with  $\theta_m^M$  from equation (2.6).

PROPOSITION 3.1 Let  $\Omega \subset \mathbb{C}^d$ ,  $\mathcal{P}$  and  $h : \mathcal{P} \times \Omega \rightarrow \mathbb{C}$  be such that Assumption 2.1 is satisfied and that the sequence  $(\Lambda_M)_{M \geq 1}$  is monotonically increasing. Then, the following assertions hold:

- (a) If there exist constants  $c_0 > 0$  and  $\alpha > 1$ , such that  $d_n(\mathcal{U}, L^\infty(\Omega)) \leq c_0 n^{-\alpha}$  for all  $n \geq 1$ , then for all  $M \geq 1$  and all  $h \in \mathcal{U}$ , we have

$$\|h - I_M(h)\|_\infty \leq 3 \cdot 2^{3\alpha} c_0 (1 + \Lambda_M)^3 M^{1-\alpha}.$$

- (b) If there exist constants  $c_0, c_1 > 0$  and  $\alpha > 0$ , such that  $d_n(\mathcal{U}, L^\infty(\Omega)) \leq c_0 e^{-c_1 n^\alpha}$  for all  $n \geq 1$ , then for all  $M \geq 1$  and all  $h \in \mathcal{U}$ , we have

$$\|h - I_M(h)\|_\infty \leq \sqrt{2} c_0 (1 + \Lambda_M)^2 \sqrt{M} e^{-c_2 M^\alpha},$$

where  $c_2 = c_1 2^{-2\alpha-1}$ .

*Proof.* The generalized empirical interpolation method (GEIM) has been introduced in (Maday *et al.*, 2016, Section 2) to generalize the concept of the magic point interpolation to general Banach spaces instead of  $L^\infty(\Omega)$ . Indeed, it is elementary to verify that the magic point interpolation can be interpreted as a GEIM and that the assumptions P1–P3 in Maday *et al.* (2016, Section 2) are satisfied in our case. Moreover, the inequalities (9) in Maday *et al.* (2016) are satisfied for  $\eta = 1$  and  $\mathcal{F}_n^\eta = \mathcal{F}_n = \text{span}\{q_1, \dots, q_n\}$  for all  $n \in \mathbb{N}$ . We therefore can apply in Maday *et al.* (2016, Theorem 13). We notice that the Lebesgue constant  $\Lambda_n$  appearing in the latter theorem is differently defined as the operator norm of  $I_n$ , namely by  $\|I_n\| := \sup_{\varphi \in L^\infty(\Omega)} \frac{\|I_n(\varphi)\|_\infty}{\|\varphi\|_\infty}$ . It is, however, elementary to verify that  $\|I_n\| \leq \Lambda_n = \sup_{z \in \Omega} \sum_{m=1}^n |\theta_m^n(z)|$ . Therefore, we can replace the constant  $\Lambda_n$  in the result of Maday *et al.* (2016, Theorem 13) with the Lebesgue constant from equation (3.2). Since  $\Lambda_n$  is monotonically increasing by assumption, we can apply Corollary 11 and Lemma 12 from Maday *et al.* (2016) and obtain the analogue of Corollary 14 in Maday *et al.* (2016). As the last step of our proof, we estimate the constants derived in the latter corollary to simplify the expression. This is an elementary step, namely writing  $n = 4\ell + k$ , we see that  $\ell_2 = 2(\ell + \lceil \frac{k}{4} \rceil) \leq n/2 + 2 \leq 3/2n$ .  $\square$

Next, we present a variant of Maday *et al.* (2009, Theorem 2.4), which is an *a priori error bound* for the magic point empirical interpolation method:

PROPOSITION 3.2 Let  $\Omega \subset \mathbb{C}^d$ ,  $\mathcal{P}$  and  $h : \mathcal{P} \times \Omega \rightarrow \mathbb{C}$  be such that Assumption 2.1 is satisfied. If there exist constants  $\alpha > \log(4)$  and  $c > 0$  such that

$$d_M(\mathcal{U}, L^\infty(\Omega)) \leq c e^{-\alpha M},$$

then for arbitrary  $\varepsilon > 0$  and  $C := \frac{c}{4} e^\alpha + \varepsilon$  we have for all  $u \in \mathcal{U}$  that

$$\|u - I_M(u)\|_\infty \leq C M e^{-(\alpha - \log(4))M}. \quad (3.3)$$

The proof follows analogous to the proof of Maday *et al.* (2009, Theorem 2.4) and is in detail provided in Gaß *et al.* (2015a). In contrast to Maday *et al.* (2009, Theorem 2.4), Proposition 3.2 expresses the result directly in terms of the Kolmogorov  $n$ -width, which we prefer. We emphasize, however, that this is only a minor difference and the scientific contribution originates from Maday *et al.* (2009).



### 3.2 A priori error bounds under analyticity conditions

On the basis of in Maday *et al.* (2009, Theorem 2.4), we identify two generic cases for which this result implies exponential convergence of magic point interpolation and integration. In the first case, the set of parametric functions consists of univariate functions that are analytic on a specific Bernstein ellipse. In the second case, the functions may be multivariate and do not need to satisfy higher-order regularity. The parameter set now is a compact subset of  $\mathbb{R}$ , and the regularity of the parameter dependence allows an analytic extension to a specific Bernstein ellipse in the parameter space.

To formulate our analyticity assumptions, we define the *Bernstein ellipse*  $B([-1, 1], \varrho)$  with parameter  $\varrho > 1$  as the open region in the complex plane bounded by the ellipse with foci  $\pm 1$  and semiminor and semimajor axis lengths summing up to  $\varrho$ . Moreover, we define for  $\underline{b} < \bar{b} \in \mathbb{R}$  the *generalized Bernstein ellipse* by

$$B([\underline{b}, \bar{b}], \varrho) := \tau_{[\underline{b}, \bar{b}]} \circ B([-1, 1], \varrho), \quad (3.4)$$

where the transform  $\tau_{[\underline{b}, \bar{b}]} : \mathbb{C} \rightarrow \mathbb{C}$  is given by

$$\tau_{\underline{b}, \bar{b}}(x) := \bar{b} + \frac{b - \bar{b}}{2}(1 - \Re(x)) + i \frac{\bar{b} - b}{2}\Im(x) \quad \text{for all } x \in \mathbb{C}. \quad (3.5)$$

For  $\Omega \subset \mathbb{R}$  we set

$$B(\Omega, \varrho) := B([\inf \Omega, \sup \Omega], \varrho). \quad (3.6)$$

**DEFINITION 3.3** Let  $f : X_1 \times X_2 \rightarrow \mathbb{C}$  with  $X_m \subset \mathbb{C}^{n_m}$  for  $m = 1, 2$ . We say  $f$  has the *analytic property with respect to  $\mathcal{X}_1 \subset \mathbb{C}$  in the first argument*, if  $X_1 \subset \mathcal{X}_1$  and  $f$  has an extension  $f_1 : \mathcal{X}_1 \times X_2 \rightarrow \mathbb{C}$  such that for all fixed  $x_2 \in X_2$  the mapping  $x_1 \mapsto f_1(x_1, x_2)$  is analytic in the inner of  $\mathcal{X}_1$  and

$$\sup_{(x_1, x_2) \in \mathcal{X}_1 \times X_2} |f_1(x_1, x_2)| < \infty. \quad (3.7)$$

We define the analytic property of  $f$  with respect to  $\mathcal{X}_2$  in the second argument analogously.

We denote

$$\mathcal{I}(h_p, \mu) := \int_{\Omega} h_p(z) \mu(dz) \quad (3.8)$$

and for all  $p \in \mathcal{P}$  and  $M \in \mathbb{N}$ ,

$$\mathcal{I}_M(h, \mu)(p) := \int_{\Omega} I_M(h)(p, z) \mu(dz), \quad (3.9)$$

where  $I_M$  is the magic point interpolation operator given by the iterative procedure from Section 2. Hence,

$$\mathcal{I}_M(h, \mu)(p) = \sum_{m=1}^M h_p(z_m^*) \int_{\Omega} \theta_m^M(z) \mu(dz), \quad (3.10)$$

where  $z_m^*$  are the magic points and  $\theta_m^M$  are given by (2.6) for every  $m = 1, \dots, M$ .

**THEOREM 3.4** Let  $\Omega \subset \mathbb{C}^d$ ,  $\mathcal{P}$  and  $h : \mathcal{P} \times \Omega \rightarrow \mathbb{C}$  be such that Assumption 2.1 holds and one of the following conditions is satisfied for some  $\varrho > 4$ ,

- (a)  $d = 1$ , i.e.,  $\Omega$  is a compact subset of  $\mathbb{R}$ , and  $h$  has the analytic property with respect to  $B(\Omega, \varrho)$  in the second argument,
- (b)  $\mathcal{P}$  is a compact subset of  $\mathbb{R}$ , and  $h$  has the analytic property with respect to  $B(\mathcal{P}, \varrho)$  in the first argument.

Then, there exists a constant  $C > 0$  such that for all  $p \in \mathcal{P}$  and  $M \in \mathbb{N}$ ,

$$\|h - I_M(h)\|_{\infty} \leq CM(\varrho/4)^{-M}, \quad (3.11)$$

$$\sup_{p \in \mathcal{P}} |\mathcal{I}(h_p, \mu) - \mathcal{I}_M(h, \mu)(p)| \leq C\mu(\Omega)M(\varrho/4)^{-M}. \quad (3.12)$$

*Proof.* Assume (a). Thanks to an affine transformation, we may without loss of generality assume  $\Omega = [-1, 1]$ . As assumed,  $(p, z) \mapsto h_p(z)$  has an extension  $f_1 : \mathcal{P} \times B(\Omega, \varrho) \rightarrow \mathbb{C}$  that is bounded and for every fixed  $p \in \mathcal{P}$  is analytic in the Bernstein ellipse  $B(\Omega, \varrho)$ . We exploit the analyticity property to relate the approximation error to the best  $n$ -term approximation of the set  $\mathcal{U} := \{h_p \mid p \in \mathcal{P}\}$ . This can conveniently be achieved by inserting an example of an interpolation method that is equipped with exact error bounds, and we choose Chebyshev projection for this task. From Trefethen (2013, Theorem 8.2) we obtain for each  $N \in \mathbb{N}$  and the interpolation  $I_N^{\text{Cheby}}$  defined in the appendix as the explicit error bound,

$$\sup_{p \in \mathcal{P}} \|f_1(p, \cdot) - I_N^{\text{Cheby}}(f_1(p, \cdot))\|_{\infty} \leq c(f_1)\varrho^{-N}, \quad (3.13)$$

with constant  $c(f_1) := \frac{4}{\varrho-1} \sup_{(p,z) \in \mathcal{P} \times B(\Omega, \varrho)} |f_1(p, z)|$ . Next, we can apply the general convergence result from Maday *et al.* (2009, Theorem 2.4). Consulting their proof, we realize that

$$\sup_{p \in \mathcal{P}} \|h_p - I_M(h)(p, \cdot)\|_{\infty} \leq CM(\varrho/4)^{-M},$$

with  $C = \frac{c(f_1)\varrho}{4}$ . Equation (3.12) follows by integration.

The proof follows analogously under assumption (b). □

REMARK 3.5 The proof makes the constant  $C$  explicit. Under assumption (a) it is given by

$$C = \frac{\varrho}{2(\varrho - 1)} \max_{(p,z) \in \mathcal{P} \times B(\Omega, \varrho)} |f_1(p, z)|. \quad (3.14)$$

Under assumption (b), an analogous constant can be derived.

REMARK 3.6 Consider a multivariate integral of the form

$$\mathcal{I}_M(h, \mu_1 \otimes \mu_2) := \int_{\mathcal{P}} \int_{\Omega} h(p, z) \mu_1(dz) \mu_2(dp), \quad (3.15)$$

with compact sets  $\mathcal{P} \subset \mathbb{R}^D$  and  $\Omega \subset \mathbb{R}^d$  equipped with finite Borel-measures  $\mu_1$  and  $\mu_2$ . Then, the application of the magic point empirical interpolation as presented in Section 2 yields

$$\mathcal{I}_M(h, \mu_1 \otimes \mu_2) := \sum_{m=1}^M \int_{\mathcal{P}} h_p(z_m^*) \mu_2(dp) \int_{\Omega} \theta_m^M(z) \mu_1(dz), \quad (3.16)$$

and, under the assumptions of Theorem 3.4, we obtain

$$|\mathcal{I}(h, \mu_1 \otimes \mu_2) - \mathcal{I}_M(h, \mu_1 \otimes \mu_2)| \leq C \mu_1(\Omega) \mu_2(\mathcal{P}) M(\varrho/4)^{-M}. \quad (3.17)$$

REMARK 3.7 Theorem 3.4 has an obvious extension to functions  $(p, z) \mapsto h_p(z)$  that are piecewise analytic. To be more precise, we can assume that  $\Omega = \cup_{i=1}^n \Omega_i$  with piecewise disjunct and compact sets  $\Omega_i \subset \mathbb{C}$  such that for each  $i = 1, \dots, n$  the mapping  $\mathcal{P} \times \Omega_i \ni (p, z) \mapsto h_p(z)$  has the analytic property w.r.t.  $B(\Omega_i, \varrho_i)$  in the second argument. The proof can be generalized to this setting, and the error bound needs to be adapted in the obvious way.

If the domain  $\Omega$  is one-dimensional and  $z \mapsto h_p(z)$  enjoys desirable analyticity properties, the *approximation error* of magic point integration *decays exponentially* in the number  $M$  of interpolation nodes, *independently of the dimension of the parameter space  $\mathcal{P}$* . Vice versa, if the parameter space is one-dimensional and  $p \mapsto h_p(z)$  enjoys desirable analyticity properties, the *approximation error decays exponentially* in the number  $M$  of interpolation nodes, *independently of the dimension of the integration space  $\Omega$* . If the function set  $\{h_p, p \in \mathcal{P}\}$  has a small Kolmogorov  $n$ -width, magic point interpolation and integration do not suffer from the curse of dimensionality—in contrast to standard polynomial approximation. This is owed to the fact that classic interpolation methods are adapted to very large function classes, whereas magic point interpolation is adapted to a parametrized class of functions.

In return, this highly appealing convergence comes with an additional cost: The offline phase involves a loop over optimizations to determine the magic parameters and the magic points. Although the online phase is independent of the dimensionality, the offline phase is thus affected. Let us further point out that Theorem 3.4 is a result on the *theoretical iterative procedure* as described in Section 2. As we will discuss in Section 5.1 below, the implementation inevitably involves additional problem simplifications and approximations to perform the necessary optimizations. In particular, instead of the whole parameter space a training set is fixed in advance. For this more realistic setting, rigorous *a posteriori* error bounds have been developed for the empirical interpolation method (see Eftang *et al.*, 2010). These bounds

rely on derivatives in the parameters and straightforwardly translate to an error bound for magic point integration.

In our implementation, we obtain the discrete training set by simulating according to the uniform distribution on the parameter space. For high-dimensional parameter spaces, simulation of pseudo-random numbers with quasi-Monte Carlo methods is highly promising. These methods are well-established and often reduce the computational complexity considerably. Also, other discretization techniques such as sparse grids can be employed. Moreover, the optimization problem (2.7), (2.8) itself can be implemented using more advanced techniques. In particular, for the applications that we consider in Section 5 and in Gaß *et al.* (2015a), the derivatives of the parametric integrands are explicitly available, which enables the use of gradient-based optimization techniques.

#### 4. Magic points for Fourier transforms

In various fields of applied mathematics, Fourier transforms play a crucial role and parametric Fourier inverse transforms of the form

$$f_q(x) = \frac{1}{2\pi} \int_{\mathbb{R}} e^{-izx} \widehat{f}_q(z) \, dz \quad (4.1)$$

need to be computed for different values  $p = (q, x) \in \mathcal{P} = \mathcal{Q} \times \mathcal{X} \subset \mathbb{R}^{D+1}$ , where the function  $z \mapsto \widehat{f}_q(z) := \int_{\mathbb{R}} e^{izy} f_q(y) \, dy$  is well-defined and integrable for all  $q \in \mathcal{Q}$ .

A prominent example arises in the context of signal processing, where signals are filtered with the help of *short-time Fourier transform* (STFT) that is of the form

$$\Phi_b(f)(z) := \int_{\mathbb{R}} f(t) w(t - b) e^{-izt} \, dt, \quad (4.2)$$

with a *window function*  $w$  and parameter  $b$ . One typical example for windows are the Gaussian windows,  $w_\sigma(t) = \frac{1}{\sqrt{2\pi}\sigma} e^{-t^2/(2\sigma^2)}$ , where an additional parameter appears, namely the standard deviation  $\sigma$  of the normal distribution. The STFT with a Gaussian window has been introduced by Gabor (1946). His pioneering approach has become indispensable for time–frequency signal analysis. We refer to Debnath & Bhatta (2014) for historical backgrounds.

We consider the truncation of the integral in (4.1) to a compact integration domain  $\Omega = [\underline{\Omega}, \overline{\Omega}] \subset \mathbb{R}$  and choose the same domain  $\Omega$  for all  $p = (q, x) \in \mathcal{P} = \mathcal{Q} \times \mathcal{X}$ . Thus, we are left to approximate for all  $p = (q, x) \in \mathcal{P}$  the integral

$$\mathcal{I}(h_p) := \int_{\Omega} h_p(z) \, dz, \text{ where } h_p(z) = h_{(q,x)}(z) = \frac{1}{2\pi} e^{-izx} \widehat{f}_q(z). \quad (4.3)$$

Then, according to Section 2, we consider the approximation of  $\mathcal{I}(h_p)$  by magic point integration, i.e., by

$$\mathcal{I}_M(h)(p) := \sum_{m=1}^M h_p(z_m^*) \int_{\Omega} \theta_m^M(z) \, dz, \quad (4.4)$$

where  $z_m^*$  are the magic points and  $\theta_m^M$  are given by (2.6) for every  $m = 1, \dots, M$ . In such cases where the analyticity properties of  $q \mapsto \widehat{f}_q(z)$  or of  $z \mapsto \widehat{f}_q(z)$  are directly accessible, Theorem 3.4 can be applied to

estimate the error  $\sup_{(q,x) \in \mathcal{Q} \times \mathcal{X}} |\mathcal{I}(h_{(q,x)}) - \mathcal{I}_M(h)(q,x)|$ . The following corollary offers a set of conditions in terms of existence of exponential moments of the functions  $f_q$ .

**COROLLARY 4.1** For  $\eta > \sqrt{15/8} (\overline{\Omega} - \underline{\Omega})$  and every parameter  $q \in \mathcal{Q}$  assume  $\int e^{(\eta+\varepsilon)|x|} |f_q(x)| dx < \infty$  for some  $\varepsilon > 0$  and assume further

$$\sup_{q \in \mathcal{Q}} \int_{\Omega} e^{\eta|x|} |f_q(x)| dx < \infty.$$

Then,

$$\|h - I_M(h)\|_{\infty} \leq C(\eta) M(\varrho(\eta)/4)^{-M}, \quad (4.5)$$

$$\sup_{(q,x) \in \mathcal{Q} \times \mathcal{X}} |\mathcal{I}(h_{(q,x)}) - \mathcal{I}_M(h)(q,x)| \leq C(\eta) |\Omega| M(\varrho(\eta)/4)^{-M}, \quad (4.6)$$

with

$$\begin{aligned} \varrho(\eta) &= \frac{2\eta}{|\Omega|} + \sqrt{\left(\frac{2\eta}{|\Omega|}\right)^2 + 1}, \\ C(\eta) &= \frac{\varrho(\eta)}{4\pi(\varrho(\eta) - 1)} e^{\eta(-\underline{\mathcal{X}} \vee \overline{\mathcal{X}})} \sup_{q \in \mathcal{Q}} \int_{\Omega} e^{\eta|x|} |f_q(x)| dx, \end{aligned}$$

where  $|\Omega| := \overline{\Omega} - \underline{\Omega}$ ,  $\underline{\mathcal{X}} = \inf \mathcal{X}$  and  $\overline{\mathcal{X}} = \sup \mathcal{X}$ .

*Proof.* From the theorem of Fubini and the lemma of Morera, we obtain that the mappings  $z \mapsto \widehat{f}_q(z)$  have analytic extensions to the complex strip  $\mathbb{R} + i[-\eta, \eta]$ . We determine the value of  $\varrho(\eta)$ . It has to be chosen such that the associated semiminor  $b_{\varrho}$  of the Bernstein ellipse  $B([-1, 1], \varrho)$  is mapped by  $\tau_{[\underline{\Omega}, \overline{\Omega}]}$  onto the semiminor of the generalized Bernstein ellipse  $B([\underline{\Omega}, \overline{\Omega}], \varrho)$  that is of length  $\eta$ . Using the relation

$$b_{\varrho} = \frac{\varrho - \frac{1}{\varrho}}{2} \quad (4.7)$$

between the semiminor  $b_{\varrho}$  of a Bernstein ellipse and its ellipse parameter  $\varrho$  we solve

$$\Im(\tau_{[\underline{\Omega}, \overline{\Omega}]}(i b_{\varrho})) = \frac{\overline{\Omega} - \underline{\Omega}}{2} b_{\varrho} = \eta \quad (4.8)$$

for  $\varrho$  and find that for

$$\varrho(\eta) = \frac{2\eta}{\overline{\Omega} - \underline{\Omega}} + \sqrt{\left(\frac{2\eta}{\overline{\Omega} - \underline{\Omega}}\right)^2 + 1} \quad (4.9)$$

the Bernstein ellipse  $B(\Omega, \varrho(\eta))$  is contained in the strip of analyticity of  $\widehat{f}_q$  and by the choice of  $\eta$  we have  $\varrho(\eta) > 4$ . Hence, assumption (a) of Theorem 3.4 is satisfied. With regard to Remark 3.5, we also obtain the explicit form of the constant  $C(\eta)$ , which proves the claim.  $\square$

Similarly, we consider the case where  $\mathcal{Q}$  is a singleton and

$$h_p(z) = h_x(z) = \frac{1}{2\pi} e^{-izx} \widehat{f}(z). \quad (4.10)$$

Under this assumption we obtain an interesting additional assertion if the integration domain  $\Omega$  is rather small. This case occurs, e.g., for approximations of STFT.

COROLLARY 4.2 For  $\eta > \sqrt{15/8} (\overline{\mathcal{X}} - \underline{\mathcal{X}})$ , we have

$$\|h - I_M(h)\|_\infty \leq C(\eta) M(\varrho(\eta)/4)^{-M}, \quad (4.11)$$

$$\sup_{x \in \mathcal{X}} |\mathcal{I}(h_x) - \mathcal{I}_M(h)(x)| \leq C(\eta) |\Omega| M(\varrho(\eta)/4)^{-M}, \quad (4.12)$$

with

$$\begin{aligned} \varrho(\eta) &= \frac{2\eta}{|\mathcal{X}|} + \sqrt{\left(\frac{2\eta}{|\mathcal{X}|}\right)^2 + 1}, \\ C(\eta) &= \frac{\varrho(\eta)}{4\pi(\varrho(\eta) - 1)} e^{\eta(-\underline{\Omega} \vee \overline{\Omega})} \sup_{z \in \Omega} |\widehat{f}(z)|, \end{aligned}$$

where  $|\mathcal{X}| := \overline{\mathcal{X}} - \underline{\mathcal{X}}$ ,  $\underline{\mathcal{X}} = \inf \mathcal{X}$  and  $\overline{\mathcal{X}} = \sup \mathcal{X}$ .

The proof of the corollary follows similarly to the proof of Corollary 4.1.

## 5. Case study

### 5.1 Implementation and complexity

The implementation of the algorithm requires further discretizations. For the parameter selection, we replace the continuous parameter space  $\mathcal{P}$  by a discrete parameter cloud  $\mathcal{P}^{\text{disc}}$ . Additionally, for the selection of magic points we replace  $\Omega$  by a discrete set  $\Omega^{\text{disc}}$ , instead of considering the whole continuous domain. Each function  $h_p$  is then represented by its evaluation on this discrete  $\Omega^{\text{disc}}$  and is thus represented by a finite-dimensional vector, numerically.

The crucial step during the offline phase is finding the solution to the optimization problem

$$(p_{M+1}, z_{M+1}) := \arg \max_{(p,z) \in \mathcal{P}^{\text{disc}} \times \Omega^{\text{disc}}} \left| h_p(z) - \sum_{m=1}^M h_p(z_m^*) \theta_m^M(z) \right|. \quad (5.1)$$

In a continuous setup, problem (5.1) is solved by optimization routines. In our discrete implementation, however, we are able to consider *all* magic parameter candidates in the discrete parameter space  $\mathcal{P}^{\text{disc}}$  and *all* magic point candidates in the discrete domain  $\Omega^{\text{disc}}$  to solve problem (5.1). This results in a complexity of  $O(M \cdot |\mathcal{P}^{\text{disc}}| \cdot |\Omega^{\text{disc}}|)$  during the offline phase for identifying the basis functions and magic points of the algorithm. We comment on our choices for the dimensionality of the involved discrete sets in the numerical section, later. Before magic point integration can be performed, the quantities  $\int_{\Omega} \theta_m^M(z) dz$  need to be computed for  $m = 1, \dots, M$ . The complexity of this final step during the offline phase depends on the number of integration points that the integration method uses.

## 5.2 Tempered stable distribution

We test the parametric integration approach on the evaluation of the density of a tempered stable distribution as an example. This is a parametric class of infinitely divisible probability distributions. A random variable  $X$  is called infinitely divisible, if for each  $n \in \mathbb{N}$  there exist  $n$  independent and identically distributed random variables whose sum coincides with  $X$  in distribution. The rich class of infinitely divisible distributions, containing, for instance, the Normal and the Poisson distribution, is characterized through their Fourier transforms by the Lévy–Khinchine representation. Using this link, a variety of infinitely divisible distributions is specified by Fourier transforms. The class of tempered stable distributions is an example of this sort, namely it is defined by a parametric class of functions, which by the theorem of Lévy–Khinchine are known to be Fourier transforms of infinitely divisible probability distributions. Its density function, however, is not explicitly available. To evaluate it, a Fourier inversion has to be performed numerically. As we show below, magic point integration can be an adequate way to efficiently compute this Fourier inversion for different parameter values and at different points on the density's domain.

Following Küchler & Tappe (2014), parameters

$$\alpha^+, \lambda^+, \alpha^-, \lambda^- \in (0, \infty), \quad \beta^+, \beta^- \in (0, 1) \quad (5.2)$$

define the distribution  $\eta_q$  on  $(\mathbb{R}, \mathcal{B}(\mathbb{R}))$  that we call a *tempered stable distribution* and write

$$\eta_q = \text{TS}(q) = \text{TS}(\alpha^+, \beta^+, \lambda^+; \alpha^-, \beta^-, \lambda^-), \quad (5.3)$$

if its characteristic function  $\varphi_q(z) := \int_{\mathbb{R}} e^{izx} \eta_q(dx)$  is given by

$$\varphi_q(z) = \exp \left( \int_{\mathbb{R}} (e^{izx} - 1) F_q(dx) \right), \quad z \in \mathbb{R}, \quad (5.4)$$

with Lévy measure

$$F_q(dx) = \left( \frac{\alpha^+}{x^{1+\beta^+}} e^{-\lambda^+ x} \mathbb{1}_{x \in (0, \infty)} + \frac{\alpha^-}{x^{1+\beta^-}} e^{-\lambda^- x} \mathbb{1}_{x \in (-\infty, 0)} \right) dx. \quad (5.5)$$

The tempered stable distribution is also defined for  $\beta^+, \beta^- \in (1, 2)$ , where the characteristic function is given by a similar expression as in (5.4). We consider the tempered stable distribution in a special case by introducing the parameter space

$$\mathcal{Q} = \{(C, G, M, Y) \mid C > 0, G > 0, M > 0, Y \in (1, 2)\} \quad (5.6)$$

and setting

$$\alpha^+ = \alpha^- = C, \quad \lambda^- = G, \quad \lambda^+ = M, \quad \beta^+ = \beta^- = Y. \quad (5.7)$$

The resulting distribution  $\gamma = \text{CGMY}(C, G, M, Y)$  is also known as CGMY distribution and is well established in finance (see Carr *et al.*, 2002).<sup>2</sup> The density function of a tempered stable or a CGMY

<sup>2</sup> As usual, we denote the parameters of the CGMY distribution with  $C, M, G, Y$  as a reference to the authors of Carr *et al.* (2002). It will be clear from the context whether  $M$  denotes the number of magic points or the CGMY parameter.

TABLE 1 *Parameter intervals for the numerical study. The parameter  $Y$  is kept constant*

	$C$	$G$	$M$	$Y$	$x$
interval	$[1, 5]$	$[1, 8]$	$[1, 8]$	1.1	$[-1, 1]$

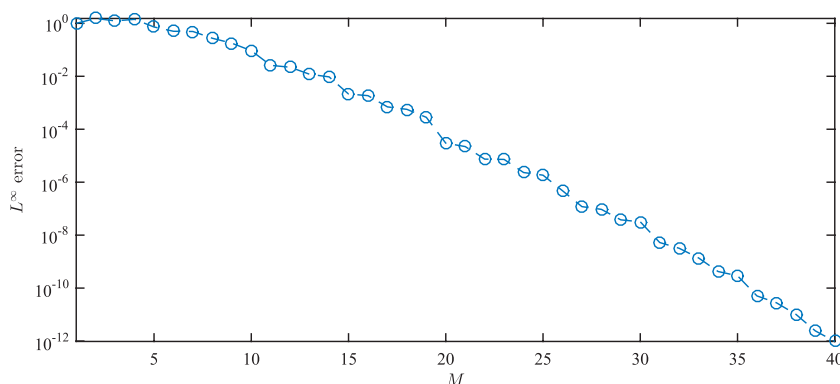


FIG. 1. Decay of the error (2.8) on  $\mathcal{P}^{\text{disc}}$  and  $\Omega^{\text{disc}}$  during the offline phase of the algorithm.

distribution, respectively, is not known in the closed form. With  $q = (C, G, M, Y) \in \mathcal{Q}$  of (5.6) and  $z \in \mathbb{R}$ , its Fourier transform, however, is explicitly given by

$$\varphi_q(z) = \exp\left(C\Gamma(-Y)((M - iz)^Y - M^Y + (G + iz)^Y - G^Y)\right), \quad (5.8)$$

wherein  $\Gamma$  denotes the gamma function. By Fourier inversion, the density  $f_q$  can be evaluated,

$$f_q(x) = \frac{1}{2\pi} \int_{\mathbb{R}} e^{-izx} \varphi_q(z) dz = \frac{1}{\pi} \int_0^\infty \Re(e^{-izx} \varphi_q(z)) dz \quad \text{for all } x \in \mathbb{R}. \quad (5.9)$$

### 5.3 Numerical results

We restrict the integration domain of (5.9) to  $\Omega = [0, 65]$  and apply the parametric integration algorithm on a subset of

$$\mathcal{P} = \mathcal{Q} \times \mathbb{R}, \quad (5.10)$$

specified in Table 1. We draw 4000 random samples from  $\mathcal{P}$  respecting the intervals bounds of Table 1 and run the offline phase until an  $L^\infty$  accuracy of  $10^{-12}$  is reached. Here, we compute the benchmark values by numerical integration using Matlab's `quadgk` routine with a tolerance  $10^{-14}$ . The error decay during the offline phase is displayed by Fig. 1. We observe exponential error decay reaching the prescribed accuracy threshold at  $M = 40$ .

Let us have a closer look at some of the basis functions  $q_m$  that are generated during the offline phase. Figure 2 depicts five basis functions that have been created at an early stage of the offline phase (left)



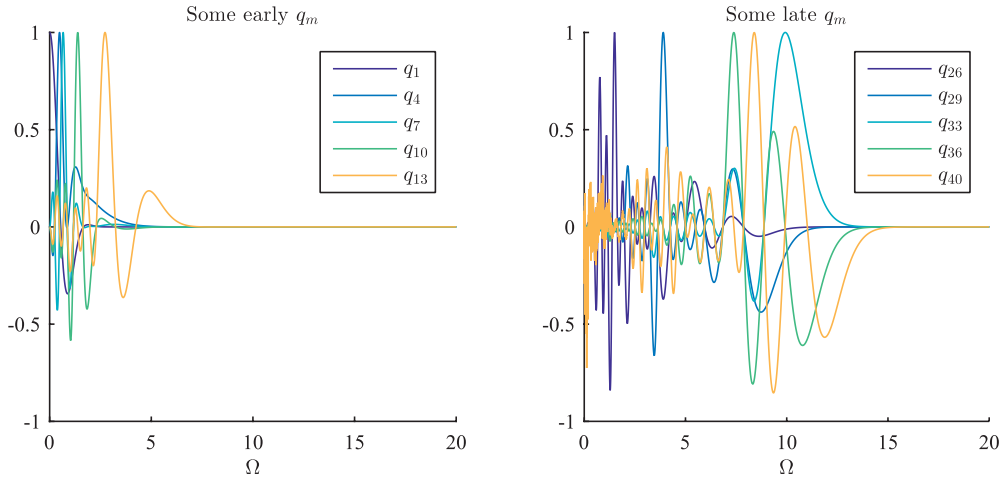


FIG. 2. Some basis functions  $q_m$  constructed early (left) and late (right) during the offline phase of the algorithm.

together with five basis functions that have been identified towards the end of it (right). Note that all plotted basis functions are supported on a relatively small subinterval of  $\Omega$ . They are numerically zero outside of it. Interestingly, the functions in both sets have intersections with the  $\Omega$  axis rather close to the origin. Such intersections mark the location of magic points. Areas on  $\Omega$  where these intersections accumulate thus reveal regions where the approximation accuracy of the algorithm is low. In these regions, the shapes across all parametrized integrands seem to be most diverse.

In the right picture, we observe in comparison that at a later stage in the offline phase, magic points  $z_m^*$  are picked further away from the origin, as well. We interpret this observation by assuming that the more magic points are chosen close to the origin, the better the algorithm is capable of approximating integrands more precisely there. Thus, its focus shifts towards the right for increasing values of  $M$  where now smaller variations attract its attention.

We now test the algorithm on parameters not contained in the training set. To this extent, we randomly draw 1000 parameter sets  $p_i = (C_i, G_i, M_i, Y_i, x_i)$ ,  $i = 1, \dots, 1000$ , uniformly distributed from the intervals given by Table 1. For each  $p_i$ ,  $i = 1, \dots, 1000$ , we evaluate  $f_{(C_i, G_i, M_i, Y_i)}(x_i)$  by Fourier inversion using Matlab's `quadgk` routine. Additionally, we approximate  $f_{(C_i, G_i, M_i, Y_i)}(x_i)$  using the interpolation operator  $\mathcal{I}_m$  for all values  $m = 1, \dots, M$ . For each such  $m$  we compute the absolute and the relative error.

The maximal absolute error over the 1000 samples, i.e., the  $L^\infty((p_i)_{i=1, \dots, 1000})$ -error for each  $m = 1, \dots, M$ , is illustrated by Fig. 3. To be precise, we compute the benchmark values by numerical Fourier inversion, restrict the resulting integration domain to  $\Omega = [0, 65]$  and integrate numerically with Matlab's `quadgk` routine with a tolerance of  $10^{-14}$ . The set-up of the numerical experiment does not fall in the scope of our theoretical result in several respects. We discretized both the parameter space and the set of magic point candidates. While Table 1 ensures a joint strip of analyticity  $\mathbb{R} + i(-1, 1)$  that all integrands  $h_p$  share, a Bernstein ellipse  $B(\Omega, \varrho)$  with  $\varrho > 4$  on which those integrands are analytic does not exist. In spite of all these limitations, we still observe that the exponential error decay of the offline phase is maintained.<sup>3</sup> From this, we draw two conclusions. First, both  $\Omega^{\text{disc}}$  and  $\mathcal{P}^{\text{disc}}$  appear

<sup>3</sup> This observation is confirmed by our empirical studies in Gaß *et al.* (2017), where the existence of such a shared strip of analyticity was sufficient for empirical exponential convergence.

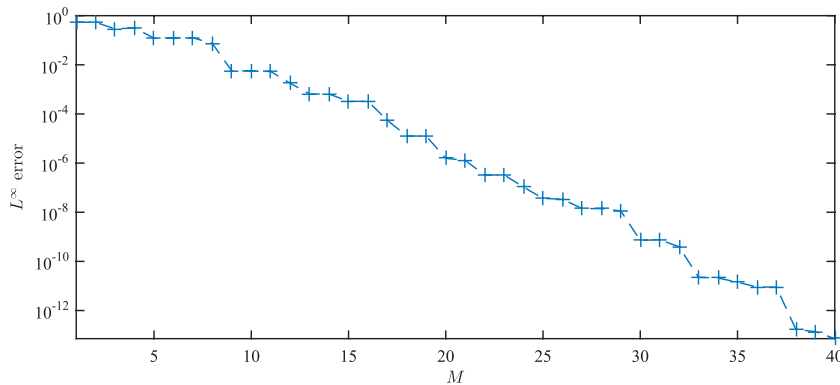


FIG. 3. Out of sample error decay. For 1000 randomly drawn parameters  $(C_i, G_i, M_i, Y_i, x_i)$ ,  $i = 1, \dots, 1000$  from the set prescribed in Table 1, the CGMY density is evaluated by the parametric integration method and by numerical Fourier inversion via Matlab's `quadgk` routine. For each number  $M$  of magic points, the maximal absolute difference between both is displayed.

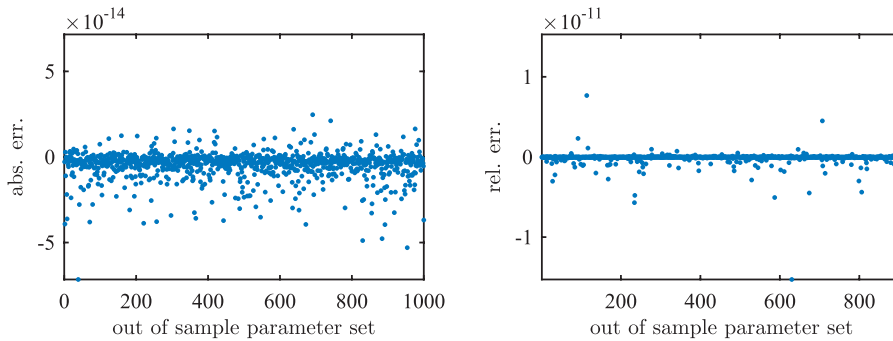


FIG. 4. All out of sample errors in detail. Left: The absolute errors achieved for each of the 1000 randomly drawn parameter sets. Right: The relative errors for density values above  $10^{-3}$ .

sufficiently rich to represent their continuous counterparts. Secondly, the practical use of the algorithm extends beyond the analyticity conditions imposed by Theorem 3.4.

Figure 4 presents the absolute errors and the relative errors for each parameter set  $p_i$ , individually, for the maximal number  $M$  magic points. Note that only relative errors for CGMY density values larger than  $10^{-3}$  have been plotted to exclude the influence of numerical noise. While individual outliers are not present among the absolute deviations, they dominate the relative deviations in contrast. This result is in line with the objective function that the algorithm minimizes.

Finally, Fig. 5 displays all magic parameters identified during the offline phase, together with those randomly drawn parameter constellations that resulted in the ten smallest absolute errors (crosses) and the 10 largest absolute errors (stars). We observe that orange parameter constellations are found in areas densely populated by magic parameters, whereas green parameter constellations often do not have magic parameters in their immediate neighbourhood, at all. This result may surprise at first. On second thought, however, we understand that areas where magic parameters accumulate mark precisely those locations in the parameter space where approximation is especially challenging for the algorithm. Integrands associated with parameter constellations from there exhibit the largest variation. Approximation for white

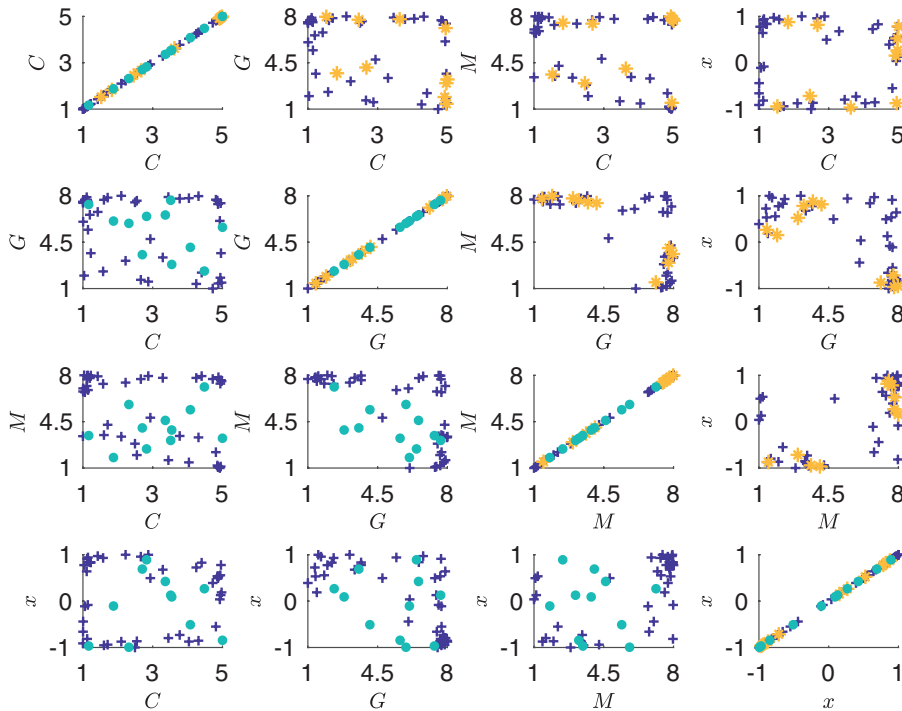


FIG. 5. An illustration of those randomly drawn parameter constellations that resulted in the 10 worst (stars) and the 10 best (circles) absolute errors. The crosses ("+" signs) the magic parameters selected during the offline phase.

areas, on the other hand, is already covered by the previously chosen magic parameters. Consequently, crosses are very likely to be found there.

#### 5.4 Comparison of magic point integration and Clenshaw–Curtis

Since in our numerical experiments, we compute univariate analytic integrands over a finite domain, it is interesting to compare the performance of the new integration routine to the Clenshaw–Curtis method. The latter is well-known for its excellence performance, particularly in case of analyticity.

To compare both methods, we use again the setting introduced in Section 5.2. We randomly sample 1000 parameter constellations according to the uniform distribution in the set specified in Table 1. For these, we compute the benchmark integrals with Matlab's `quadgk` routine for an absolute tolerance of  $10^{-14}$  and the maximal difference with respect to the results obtained by both the Clenshaw–Curtis quadrature rules and the magic point integration, for varying numbers of quadrature nodes, respectively, magic points,  $M$ .

Figure 6 displays the resulting error decays. Here, the magic point integration method clearly outperforms the Clenshaw–Curtis quadrature rule. For less than 35 magic points, the accuracy of magic point integration reaches already  $10^{-10}$ , while the error of the Clenshaw–Curtis quadrature is still in the second digit. The range of machine precision, i.e., an error of  $10^{-12}$ , is achieved with about 40 magic points while about 200 Chebyshev nodes are required in the Clenshaw–Curtis routine.

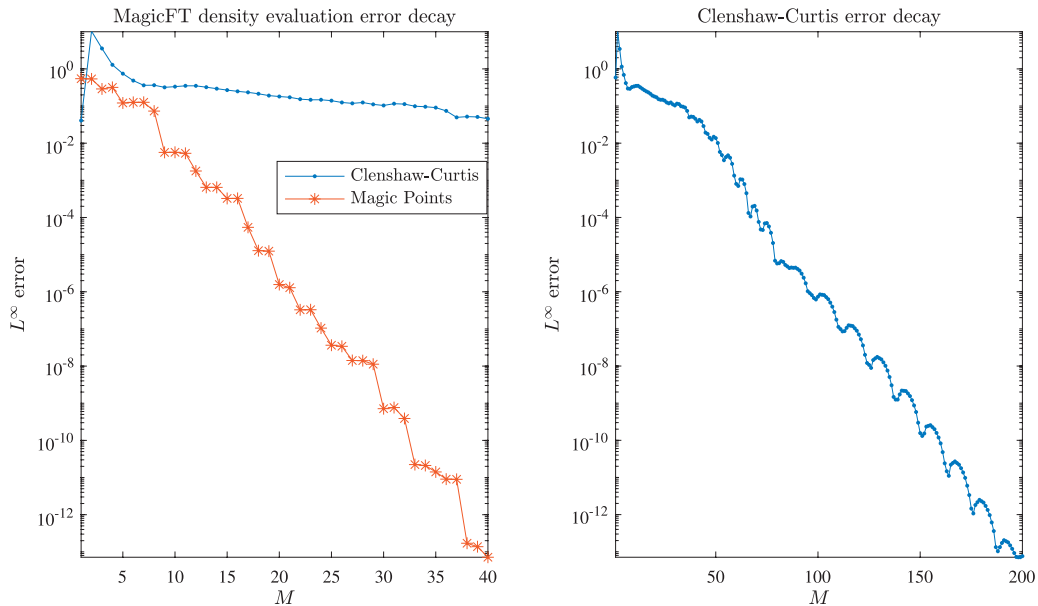


FIG. 6. Comparison of the error decay of magic point integration versus Clenshaw–Curtis. The maximal absolute errors achieved for each of the 1000 randomly drawn parameter sets for varying numbers  $M$  of quadrature nodes. Left: Performance of both magic point integration and Clenshaw–Curtis for  $M = 1, \dots, 41$  quadrature nodes. Right: Performance of Clenshaw–Curtis for  $M = 1, \dots, 200$  quadrature nodes. Here, the magic point integration clearly outperforms the Clenshaw–Curtis quadrature.

In accordance with the theoretical results, we observe an exponential error decay for both methods. For magic point integration, the rate of convergence, however, is much higher. This is not reflected by the current theoretical results. Indeed, in the proof of Theorem 3.4, we estimate the Kolmogorov  $n$ -width by the Chebyshev interpolation, i.e., by the Clenshaw–Curtis quadrature. The theoretical bound that we obtained is therefore larger than the error bound for the Clenshaw–Curtis quadrature.

The numerical experiments reveal that the magic point interpolation is actually able to reduce the computational complexity by adapting a quadrature rule to the parametric family of integrands in the offline phase. Moreover, this gives evidence that the Kolmogorov  $n$ -width of the parametric family of integrands is indeed significantly smaller than the error obtained by the interpolation with Chebyshev polynomials. Figure 7 displays the magic points versus the Chebyshev nodes, scaled to the unit interval. It is interesting to see that there is not even a qualitative similarity between the two different sets of points.

### 5.5 Comparison of magic point integration and Chebyshev interpolation in the parameter space

As highlighted in Remark 2.3 and Remark 2.2, magic point integration can be seen both as a quadrature rule and as an interpolation method in the parameter space. Therefore, we also compare the performance of the magic point integration with the Chebyshev interpolation in the parameter space. In our numerical experiments reported up to this point, we considered a five-dimensional parameter space as laid in Table 1. Tensorized Chebyshev interpolation, however, is exposed to the curse of dimensionality, and therefore magic point integration will clearly outperform this method. To gain more insight, we reduce the parameter space to a univariate, a bivariate and a three-dimensional setting, below.

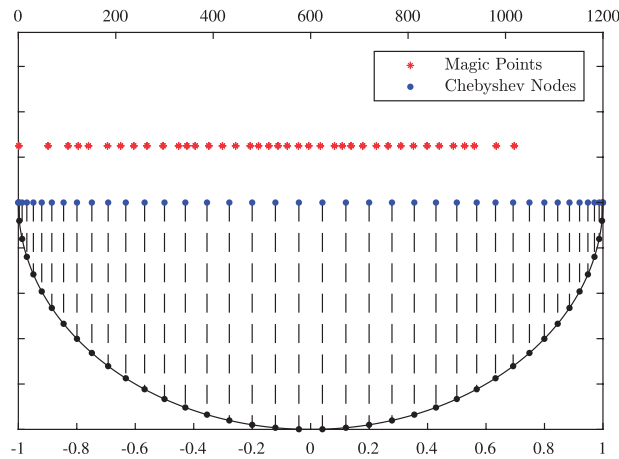


Fig. 7. Magic points versus Clenshaw–Curtis points for  $M = 40$ .

**5.5.1 Comparison for a single free parameter.** We fix the parameters  $C = 1$ ,  $M = 4$ ,  $Y = 1.1$ ,  $x = -1$ , and only vary  $G \in [1, 8]$ . For a uniform test grid of 100 parameters in  $[1, 8]$ , we compute the benchmark integrals with Matlab's `quadgk` routine for an absolute tolerance of  $10^{-14}$ , and the maximal difference with respect to the results obtained by both Chebyshev interpolation in the parameter  $G$  and magic point integration for a varying number of magic points and Chebyshev points  $M$ , respectively. We refer to the appendix for a precise definition of the Chebyshev interpolation that we implemented.

Figure 8 displays the resulting error decays. Both methods perform comparably well. As expected, the convergence is exponential in the number of nodes. The magic point integration shows a slightly higher rate of decay as the Chebyshev interpolation. Figure 9 displays the first 10 magic parameters versus the first 10 Chebyshev nodes in  $G$ , scaled to the unit interval. Remarkably, we recognize similarities in the choices of points, as both sets cluster at the boundaries. We also recognize significant differences. The magic points first of all are not located at the Chebyshev points. Secondly, there is an asymmetry in their distribution with more points on the left side than on the right.

**5.5.2 Comparison for two free parameters.** We fix the parameters  $C = 1$ ,  $M = 4$ ,  $Y = 1.1$ , and vary  $G \in [1, 8]$  and  $x \in [-1, 1]$ . For a uniform test grid of  $100 \times 100$  parameters in  $[1, 8] \times [-1, 1]$ , we compute the benchmark integrals with Matlab's `quadgk` routine for an absolute tolerance of  $10^{-14}$  and the maximal difference with respect to the results obtained by both tensorized Chebyshev interpolation in  $G, x$ , and magic point integration for a varying number of magic points and Chebyshev points  $M$ , respectively. For the precise definition of the tensorized Chebyshev interpolation, we again refer to the appendix. Since we are using the tensorized Chebyshev interpolation, we have a grid of Chebyshev nodes. We specify  $N$  number of nodes in both directions,  $G$  and  $x$ , and obtain a total number of  $M = N^2$  number of nodes.

Figure 10 displays the error decay of both the magic point integration and the tensorized Chebyshev interpolation in  $G, x$ . The magic point integration outperforms the Chebyshev interpolation by far. For an accuracy in the range of machine precision, i.e.,  $10^{-12}$ , less than 25 magic parameters versus over 700 Chebyshev nodes are required. This is an empirical indicator that magic point integration has the potential to tackle the curse of dimensionality.

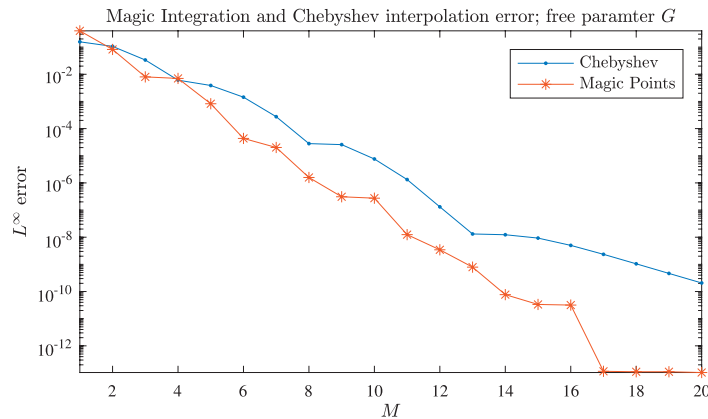


FIG. 8. Comparison of the error decay of magic point integration versus Chebyshev interpolation in the parameter space. The maximal absolute errors achieved for each of the 100 equidistant grid points in the parameter set  $G \in [1, 8]$ , for varying numbers  $M$  of interpolation nodes.

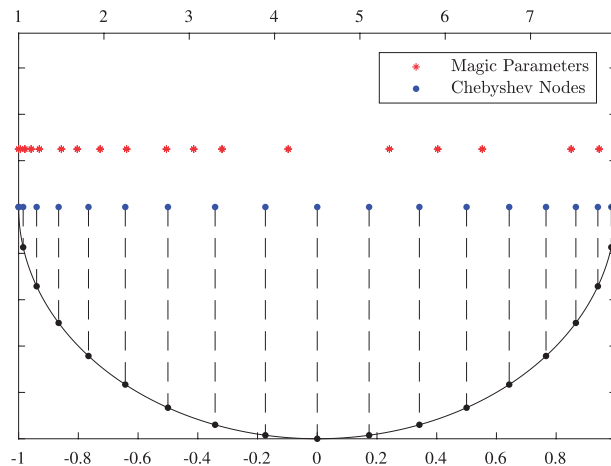


FIG. 9. Magic parameters versus Chebyshev interpolation nodes for  $M = 19$  interpolation nodes.

Next, we investigate the performance in relation to the dimensionality of the parameter space in some more depth. Figure 10 shows that the convergence rate of magic point integration is considerably higher. Yet, it is not clear how much the method is still exposed to the curse of dimensionality in the parameter space. From Fig. 8, we may conclude that both methods perform comparably in the univariate setting. Therefore, we next display the error decays of the Chebyshev interpolation in a different scale. While for magic point integration we show the error for the number of magic points  $M$ , we show the error of the Chebyshev interpolation for  $N$ , the number of nodes chosen in each of the interpolation axes. The result is provided in Fig. 11.

The similarity of both curves is striking. For  $M = N = 15$ , we observe roughly the same error of about  $10^{-8}$ . To be clear, this means that 15 summands have been used for the magic point integration, to obtain

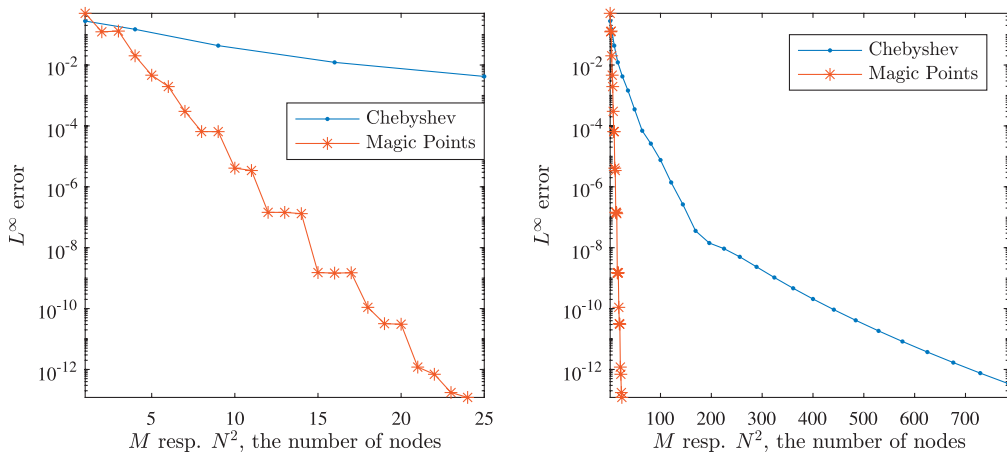


FIG. 10. Comparison of the error decay of magic point integration versus Chebyshev interpolation in the parameter space. The maximal absolute errors achieved for each of the  $100 \times 100$  equidistant grid points in the parameter set  $(G, x) \in [1, 8] \times [-1, 1]$ , for varying numbers  $M$  of interpolation nodes.

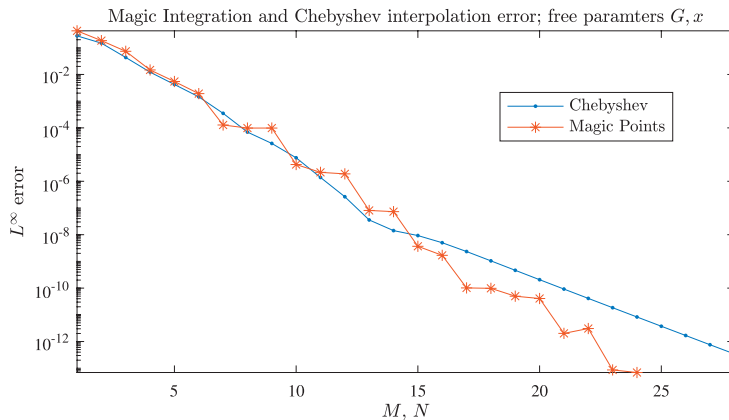


FIG. 11. Comparison of the error decay of magic point integration versus Chebyshev interpolation in the parameter space; specifications as in Fig. 10. For the Chebyshev interpolation the number of summands equals  $(N + 1)^2$ , while the number of summands in the magic point interpolation is  $M$ . This figure visualizes that the magic point integration seen as interpolation in the parameter space is not exposed to the curse of dimensionality—in contrast to the tensorized Chebyshev interpolation.

a maximal error of  $10^{-8}$  over the  $10^4$  test parameter. For the same precision,  $(N + 1)^2 = 256$  summands are required for the tensorized Chebyshev interpolation in  $G, x$ . In other words, in these experiments, the magic point interpolation exhibits indeed *no curse of dimensionality* in the parameter space.

Figure 12 displays the grid of tensorized Chebyshev nodes and of magic parameters for which roughly the same accuracy of  $10^{-12}$  has been achieved. This visualizes the complexity reduction of magic points versus Chebyshev interpolation. In contrast to the example of the univariate parameter space, the distribution of the magic parameters is totally different from the Chebyshev points. It is, moreover, apparent that they only appear at the boundaries of the domain.

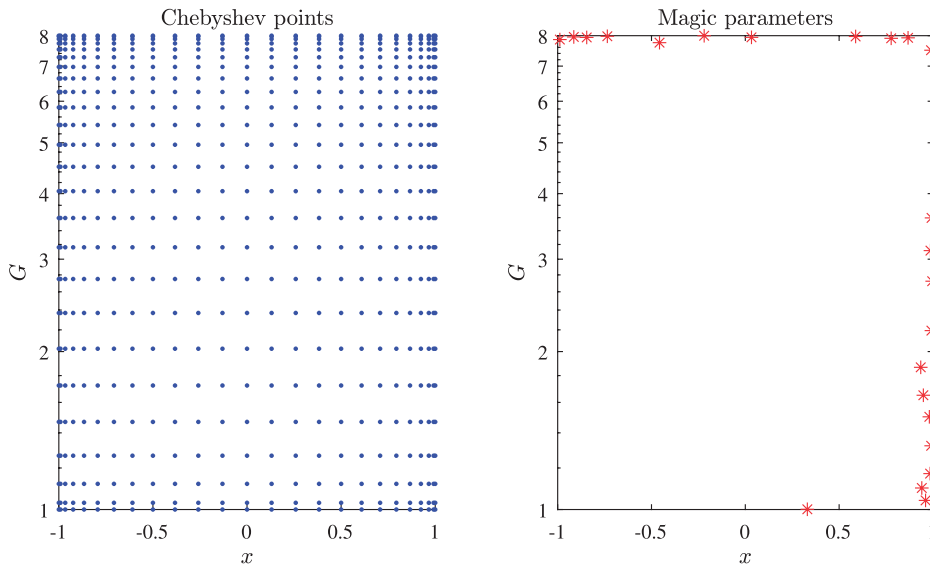


FIG. 12.  $M = 23$  Magic parameters versus  $(N + 1)^2$  Chebyshev interpolation nodes for  $N = 28$ , for which both methods achieve roughly the same accuracy.

**5.5.3 Comparison for three free parameters with low-rank tensor techniques.** In this section, we consider three varying parameters. We have already seen for the bivariate parameter space that the tensorized Chebyshev interpolation is exposed to the curse of dimensionality, in contrast to the magic point empirical interpolation. Low-rank tensor techniques start from the tensorized Chebyshev interpolation and exploit its compressible structure by a greedy procedure, which is similar to the offline phase in the magic point interpolation. It is, therefore, very interesting to investigate the performance of low-rank tensor techniques in this example. To do so, we use the `chebfun` toolbox for Matlab, see <http://www.chebfun.org> (accessed 16 November 2017), developed by Prof. Trefethen and his working group, particularly we use `chebfun3` written by Benham Hashemi (see also Hashemi & Trefethen, 2017).

We fix the parameters  $C = 1$ ,  $Y = 1.1$  and vary  $(G, M, x) \in [1, 8] \times [1, 8] \times [-1, 1]$ . We uniformly sample a set of 1000 test parameters from  $[1, 8] \times [1, 8] \times [-1, 1]$ , compute the benchmark integrals with Matlab's `quadgk` routine for an absolute tolerance of  $10^{-14}$  and the maximal difference with respect to the results obtained by `chebfun3` from in  $G, M, x$  for a number of absolute tolerances, namely for  $(10^{-1-k/2})_{k=0,\dots,22}$ , and the magic point integration for varying numbers of magic points  $M$ . To compare the complexity of both interpolation methods, we compute the number  $M$  of summands of each `chebfun3` object. Moreover, `chebfun3` allows to extract the number  $M$  of summands required for the same precision without the use low-rank compression techniques. Figure 13 displays the error achieved by `chebfun3`, magic point integration for the number of summands  $M$  each. The third line, which is dashed, shows the error for `chebfun3` for the number of summands that would be required for the tensorized Chebyshev interpolation.

The figure shows that the low-rank tensor technique performs considerably better compared to the tensorized Chebyshev interpolation. For instance, `chebfun3` achieves an accuracy of about  $10^{-2}$  with less than 200 summands, while the interpolation based on the full tensor structure requires about 500 summands. The performance of magic point integration outperforms both methods by far. This can be



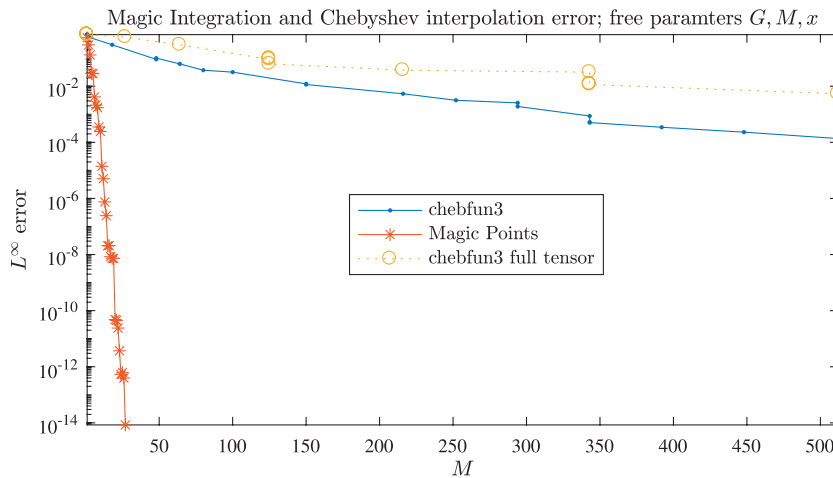


FIG. 13. Magic integration and Chebyshev interpolation three varying parameters  $G$ ,  $M$ ,  $x$ .

explained by the fact that the accuracy of the tensorized Chebyshev interpolation is determined by the resolution along the axes, which carries over to the low-rank approximations. This *a priori* choice of axes, however, is not present for the magic point interpolation. Instead, the set of basis functions is iteratively produced for the set of functions of interest.

### Acknowledgements

The authors thank Bernard Haasdonk, Laura Iapichino, Daniel Wirtz and Barbara Wohlmuth for fruitful discussions, and two anonymous referees for their valuable comments and suggestions. Maximilian Gaß thanks the KPMG Center of Excellence in Risk Management and Kathrin Glau acknowledges the TUM Junior Fellow Fund for financial support.

### REFERENCES

- ANTIL, H., FIELD, S. E., HERRMANN, F., NOCHETTO, R. H. & TIGLIO, M. (2013) Two-step greedy algorithm for reduced order quadratures. *J. Sci. Comput.*, **57**, 604–637.
- BALLANI, J. (2012) Fast evaluation of singular BEM integrals based on tensor approximations. *Numer. Math.*, **121**, 433–460.
- BARRAULT, M., MADAY, Y., NGUYEN, N. C. & PATERA, A. T. (2004) An empirical interpolation method: application to efficient reduced-basis discretization of partial differential equations. *C. R. Math.*, **339**, 667–672.
- BEBENDORF M., MADAY Y. & STAMM B. (2014) *Comparison of Some Reduced Representation Approximations*. In: A. Quarteroni & G. Rozza (eds) *Reduced Order Methods for Modeling and Computational Reduction*. MS&A - Modeling, Simulation and Applications, vol 9. Cham: Springer, pp. 67–100.
- BROCKWELL, P. J. & DAVIS, R. A. (2002) *Introduction to Time Series and Forecasting*, 2nd edn. New York: Springer.
- CARR, P., GEMAN, H., MADAN, D. B. & YOR, M. (2002) The fine structure of asset returns: An empirical investigation. *J. Bus.*, **75**, 305–333.
- CHATURANTABUT, S. & SORESENSEN, D. C. (2010) Nonlinear model reduction via discrete empirical interpolation. *SIAM J. Sci. Comput.*, **32**, 2737–2764.
- COOLEY, J. W. & TUKEY, J. W. (1965) An algorithm for the machine calculation of complex Fourier series. *Math. Comput.*, **19**, 297–301.

- DEBNATH, L. & BHATTA, D. (2014) *Integral Transforms and Their Applications*, 3rd edn. Boca Raton, USA: Apple Academic Press Inc.
- DUFFIEUX, P.-M. (1946) *L'Intégrale de Fourier et ses Applications à l'Optique*. Rennes: Oberthur.
- EFTANG, J. L., GREPL, M. A. & PATERA, A. T. (2010) A posteriori error bounds for the empirical interpolation method. *Ser. I*, **348**, 575–579.
- ERNST, R. R. & ANDERSON, W. A. (1966) Application of Fourier transform spectroscopy to magnetic resonance. *Rev. Sci. Instrum.*, **37**, 93–102.
- GABOR, D. (1946) Theory of communication. *J. Inst. Electr. Eng. III Radio Commun. Eng.*, **93**, 429–457.
- GAB, M., GLAU, K., MAHLSTEDT, M. & MAIR, M. (2015a) Chebyshev interpolation for parametrized option pricing. *Working Paper*. Available at URL <http://arxiv.org/abs/1505.04648>. Accessed 16 November 2017.
- GAB, M., GLAU, K. & MAIKR, M. (2017) Magic points in finance: empirical integration for parametrized option pricing. *SIAM J. Finan. Math.*, **8**, 766–803.
- GRASEDYCK, L., KRESSNER, D. & TOBLER, C. (2013) A literature survey of low-rank tensor approximation technique. *GAMM-Mitt.*, **36**, 53–78.
- HASHEMI, B. & TREFETHEN, L. N. (2017) Chebfun in three dimensions. *SIAM J. Sci. Comput.*, **39**, C341–C363.
- HASTIE, T., TIBSHIRANI, R. & FRIEDMAN, J. (2009) *The Elements of Statistical Learning*, 2nd edn. Heidelberg: Springer.
- HEINRICH, S. (1998) Monte Carlo complexity of global solution of integral equations. *J. Complex.*, **14**, 151–175.
- HEINRICH, S. & SINDAMBIWE, E. (1999) Monte Carlo complexity of parametric integration. *J. Complex.*, **15**, 317–341.
- KAMMLER, D. W. (2007) *A First Course in Fourier Analysis*, 2nd edn. New York: Cambridge University Press.
- KÜCHLER, U. & TAPPE, S. (2014) Exponential stock models driven by tempered stable processes. *J. Econometrics*, **181**, 53–63.
- MADAY, Y., MULA, O. & TURINICI, G. (2016) Convergence analysis of the generalized empirical interpolation method. *SIAM J. Numer. Anal.*, **54**, 1713–1731.
- MADAY, Y., NGUYEN, C. N., PATERA, A. T. & PAU, G. S. H. (2009) A general multipurpose interpolation procedure: the magic points. *Comm. Pure Appl. Anal.*, **8**, 383–404.
- PINKUS, A. (1985)  $n$ -widths in approximation theory. *Ergebnisse der Mathematik und ihrer Grenzgebiete [Results in Mathematics and Related Areas (3)]*, Vol. 7. Heidelberg: Springer.
- STARK, H. (1982) Theory and measurement of the optical Fourier transform. *Applications of Optical Fourier Transforms* (H. Stark ed). New York: Academic Press, pp. 2–40.
- TREFETHEN, L. N. (2013) *Approximation Theory and Approximation Practice*. New York: SIAM Books.

## Appendix

Since different versions of the Chebyshev interpolation are used in the literature, we introduce the version that we use. For interpolation univariate functions  $f : [-1, 1] \rightarrow \mathbb{R}$ , the interpolation with Chebyshev polynomials of degree  $N$  is of the form

$$I_N^{\text{Cheby}}(f)(x) := \sum_{j=0}^N c_j T_j(x) \quad (\text{A.1})$$

with coefficients  $c_j := \frac{2^{1_{0 < j < N}}}{N} \sum_{k=0}^N {}'' f(x_k) \cos(j\pi \frac{k}{N})$ , for  $j \leq N$ , basis functions  $T_j(x) := \cos(j \arccos(x))$  for  $x \in [-1, 1]$  and  $j \leq N$ , where  $\sum {}''$  indicates that the first and last summands are halved. The Chebyshev nodes  $x_k = \cos(\pi \frac{k}{N})$  are displayed in Figs 7 and 9. For an arbitrary compact parameter interval  $[\underline{x}, \bar{x}]$ , interpolation (A.1) needs to be adjusted by the appropriate linear transformation.

We consider the tensorized Chebyshev interpolation of functions  $f : [-1, 1]^D \rightarrow \mathbb{R}$ . For a more general hyperrectangular domain  $[\underline{x}_1, \bar{x}_1] \times \dots \times [\underline{x}_D, \bar{x}_D]$ , appropriate linear transformations need to be applied. Let  $\bar{N} := (N_1, \dots, N_D)$  with  $N_i \in \mathbb{N}_0$  for  $i = 1, \dots, D$ . The interpolation, which has  $\prod_{i=1}^D (N_i + 1)$

summands, is given by

$$I_{\overline{N}}^{\text{Cheby}}(f)(x) := \sum_{j \in J} c_j T_j(x), \quad (\text{A.2})$$

where the summation index  $j$  is a multi-index ranging over  $J := \{(j_1, \dots, j_D) \in \mathbb{N}_0^D : j_i \leq N_i \text{ for } i = 1, \dots, D\}$ . For  $j = (j_1, \dots, j_D) \in J$  the basis function is  $T_j(x_1, \dots, x_D) = \prod_{i=1}^D T_{j_i}(x_i)$ . The coefficients  $c_j$  for  $j = (j_1, \dots, j_D) \in J$  are given by

$$c_j = \left( \prod_{i=1}^D \frac{2^{\mathbb{1}_{\{0 < j_i < N_i\}}}}{N_i} \right) \sum_{k_1=0}^{N_1} \dots \sum_{k_D=0}^{N_D} f(x^{(k_1, \dots, k_D)}) \prod_{i=1}^D \cos\left(j_i \pi \frac{k_i}{N_i}\right), \quad (\text{A.3})$$

where the Chebyshev nodes  $x^k$  for the multi-index  $k = (k_1, \dots, k_D) \in J$  are given by  $x^k = (x_{k_1}, \dots, x_{k_D})$  with the univariate Chebyshev nodes  $x_{k_i} = \cos\left(\pi \frac{k_i}{N_i}\right)$  for  $k_i = 0, \dots, N_i$  and  $i = 1, \dots, D$ .

For the precise definition of the approximation with `chebfun3` we refer to [Hashemi & Trefethen \(2017\)](#).