

## **A parametrized barycentric approximation for inverse problems with application to the Black–Scholes formula**

OLIVER SALAZAR CELIS\*

*Department of Mathematics and Computer Science, University of Antwerp,  
Middelheimlaan 1, 2020 Antwerpen, Belgium and  
EY, EMEA Financial Services Organisation (FSO), Quantitative Advisory Services,  
De Kleetlaan 2, 1831 Diegem, Belgium*

\*Corresponding author: oliver.salazarcelis@uantwerpen.be oliver.salazar.celis@be.ey.com

[Received on 9 August 2016; revised on 25 January 2017]

We introduce a method to construct a bivariate rational approximation particularly suited to accurately and compactly represent the inverse of a bivariate function. At the core of the method is a parametrized barycentric form of which the involved unknowns are determined from a sequence of univariate approximations. Our exposition focuses on the inversion of the Black–Scholes formula, yielding an accurate expression for the implied volatility. We demonstrate that our result significantly improves the accuracy of existing bivariate approximations, and, based on S&P 500 Index option data, we show that the accuracy gain proves to be practically relevant. Using the obtained coefficients, included in this article, the approach can easily be implemented.

*Keywords:* implied volatility; Black–Scholes; inverse function; analytical approximation; rational functions; barycentric form; parametrized model.

### **1. Motivation**

Implied volatility (IV) is of great practical importance and it is ubiquitous in any application of market participants: it is a measure of potential future stock movements, it serves to quote options and calibrate stochastic volatility models such as the Heston model (Heston, 1993), it is hedged against by market models and drives risk management tools for the allocation of economical capital.

Because a closed-form solution for the IV is not known, iterative root-finding techniques (Newton–Raphson, Dekker–Brent, . . .) are commonly used by scholars and practitioners (see, e.g., Manaster & Koehler, 1982; Jäckel, 2015). In real-time situations, large data providers often need to invert millions of options as fast as possible (see Baumeister, 2013). Analytical approximations that can deliver IVs in constant time are therefore an attractive alternative to such iterative techniques.

More so than polynomials (see, e.g., Brenner & Subrahmanyam, 1988; Chance, 1996; Corrado & Miller, 1996; Chambers & Nawalkha, 2001; Lorig *et al.*, 2014), rational functions are particularly suited for accurate and low-complexity analytical IV approximations. Their use in this context has been opted before (among others by Li, 2008; Pistorius & Stolte, 2012; Jäckel, 2015). To the best of our knowledge, only Li (2008) proposes a truly bivariate rational approximation. However, based on S&P 500 Index (SPX) option data, we find that several claims are repriced outside the quoted bid–ask spread when using this approximation. In addition, the approximation turns out to be challenging to obtain (see, e.g., Li, 2008, section 2.3), not to mention replicate or improve its accuracy.

Our main contribution is a method to construct a bivariate rational approximation that can be used to represent the inverse of a bivariate function. This approximation relies on a parametrized barycentric form

of which the involved unknowns are determined from a sequence of (simpler) univariate approximations. Although our focus is on the inversion of the Black–Scholes formula, the essence of the method applies to similar inverse approximation problems in general.

The article is organized as follows. First, the initial notational requisites regarding the Black–Scholes and the inverse problem are presented in Section 2. The approximation details of our approach are explained in Section 3, although they may be skipped in a first reading. The obtained improvements in terms of storage and accuracy are presented in Section 4. The practical implications thereof are illustrated by considering SPX options in Section 5. Finally, Section 6 concludes with a discussion.

## 2. Black–Scholes and the inverse problem

In what follows, we adopt the approach of Li (2008) and Pistorius & Stolte (2012). Let  $S$  be the current stock price,  $\sigma$  the volatility of the stock,  $r$  the risk-free interest rate and  $\delta$  the dividend rate. Let  $C$  be the price of a call option at time  $t$  with expiration date  $T$  and strike price  $K$ , then the Black–Scholes formula expresses  $C$  in closed form as follows:

$$C(S, t; r, \sigma, T, K, \delta) = S e^{-\delta(T-t)} \mathcal{N}(d_1) - K e^{-r(T-t)} \mathcal{N}(d_2),$$

where

$$d_1 = \frac{\log(S e^{(r-\delta)(T-t)}/K)}{\sigma\sqrt{T-t}} + \frac{1}{2}\sigma\sqrt{T-t}, \quad d_2 = d_1 - \sigma\sqrt{T-t}$$

and  $\mathcal{N}(\cdot)$  is the cumulative normal distribution function.

Alternatively, the Black–Scholes formula can also be expressed in terms of the normalized call option price  $c(\cdot, \cdot)$  (see, e.g., Higham, 2004) by

$$C(S, t; r, \sigma, T, K, \delta) = S e^{-\delta(T-t)} c\left(\log(S e^{(r-\delta)(T-t)}/K), \sigma\sqrt{T-t}\right), \quad (2.1)$$

where

$$c(x, v) = \mathcal{N}\left(\frac{x}{v} + \frac{v}{2}\right) - e^{-x} \mathcal{N}\left(\frac{x}{v} - \frac{v}{2}\right), \quad (2.2)$$

with moneyness  $x$  and integrated volatility  $v$  equal to

$$\begin{aligned} x &= \log(S e^{(r-\delta)(T-t)}/K) = \log(F/K), \\ v &= \sigma\sqrt{T-t} \end{aligned}$$

and  $F = S e^{(r-\delta)(T-t)}$  is the forward price. For ease of notation, we often denote  $c(x, v)$  simply by  $c$ .

The Black–Scholes formula implied that volatility  $\sigma$  is then obtained from inverting (2.2) with respect to  $v$  and dividing by  $\sqrt{T-t}$ . It is known that one merely needs to focus on call prices due to the put-call parity

$$c(x, v) = p(x, v) + 1 - e^{-x}. \quad (2.3)$$

In analogy to (2.1), the normalized put price  $p(\cdot, \cdot)$  is obtained from the actual put-price  $P$  after dividing by  $S e^{-\delta(T-t)}$ .

## 2.1 Domain

As explained in Li (2008), the domain

$$D = \{-1/2 \leq x \leq 1/2, 0 < v \leq 1, -2 \leq x/v \leq 2\}$$

is sufficient for most realistic practical needs. The condition  $-2 \leq x/v \leq 2$  excludes regions where option prices are very insensitive to changes in  $v$  and inversion is not very meaningful. This condition is equivalent to the condition for the normalized call price  $c(x, -x/2) \leq c(x, v) \leq c(x, 1)$ , which can be checked *a priori* without the need for the unknown implied integrated volatility  $v$ .

The domain can further be restricted to either  $x \geq 0$  or  $x \leq 0$  due to the following duality

$$c(-x, v) = e^x c(x, v) + 1 - e^x, \quad (2.4)$$

which implies

$$v(x, c) = v(-x, e^x c + 1 - e^x). \quad (2.5)$$

Hence, an option with moneyness  $x \geq 0$  and normalized call price  $c(x, v)$  can be inverted using  $-x \leq 0$  and  $c(-x, v)$  from (2.4). We adopt the approximation domain where  $x \leq 0$ , since the error in (2.4) from a normalized call price, obtained after plugging in an approximated volatility, is not magnified by a factor  $e^x$  with  $x \geq 0$ , but rather multiplied by a factor less than one. The domain  $\{-1/2 \leq x \leq 0, 0 < v \leq 1, -2 \leq x/v \leq 2\}$  is illustrated in Fig. 1a. Note that for each fixed  $x$ -value, a different number of scattered  $c(x, v)$  values are available, i.e., the domain for the inverse problem is typically not gridded.

It is further convenient to consider  $v$  in function of the square root of the normalized call price  $\sqrt{c}$  rather than  $c$  directly. As shown in Fig. 1c, this stems from the square root-like behaviour of  $v(x, c)$  near  $c \approx 0$  (with  $x$  fixed). Figure 1d illustrates that  $v$  becomes more linear near  $c \approx 0$  when considered in function of  $\sqrt{c}$ , which eases the approximation effort in this region. On the other hand, this also transforms the original domain, as shown in Fig. 1a and 1b. Ultimately, the approximations of Li (2008) and Pistorius & Stolte (2012) are also expressed in terms of  $\sqrt{c}$ .

We remark that, if desired, the domain can be enlarged in the  $x$ - and  $v$ -directions, although those regions are typically associated with less liquid options. Also, a separate, dedicated approximation can be constructed for the excluded region using the same techniques outlined in this article, for instance, based on  $\log(c)$ . To not overload the current exposition and maintain comparable results as in the literature, our main focus is on the domain above.

## 3. Approximation

At the core of our approach is the bivariate parametrized barycentric form (3.1), which is introduced and motivated in Section 3.2. As explained in Sections 3.3 and 3.4, the involved unknowns can be determined from a sequence of univariate rational approximations. Before that, we first detail how the underlying data are generated in Section 3.1.

### 3.1 Data

The normalized call option price  $c(x, v)$  is calculated from (2.2), but recall that our goal is to approximate  $v(x, c)$ . To that end, we evaluate values  $(x_i, v_j)$  on an  $(N, M) \in \mathbb{N}_0^2$  grid with  $i = 1, \dots, N$  and

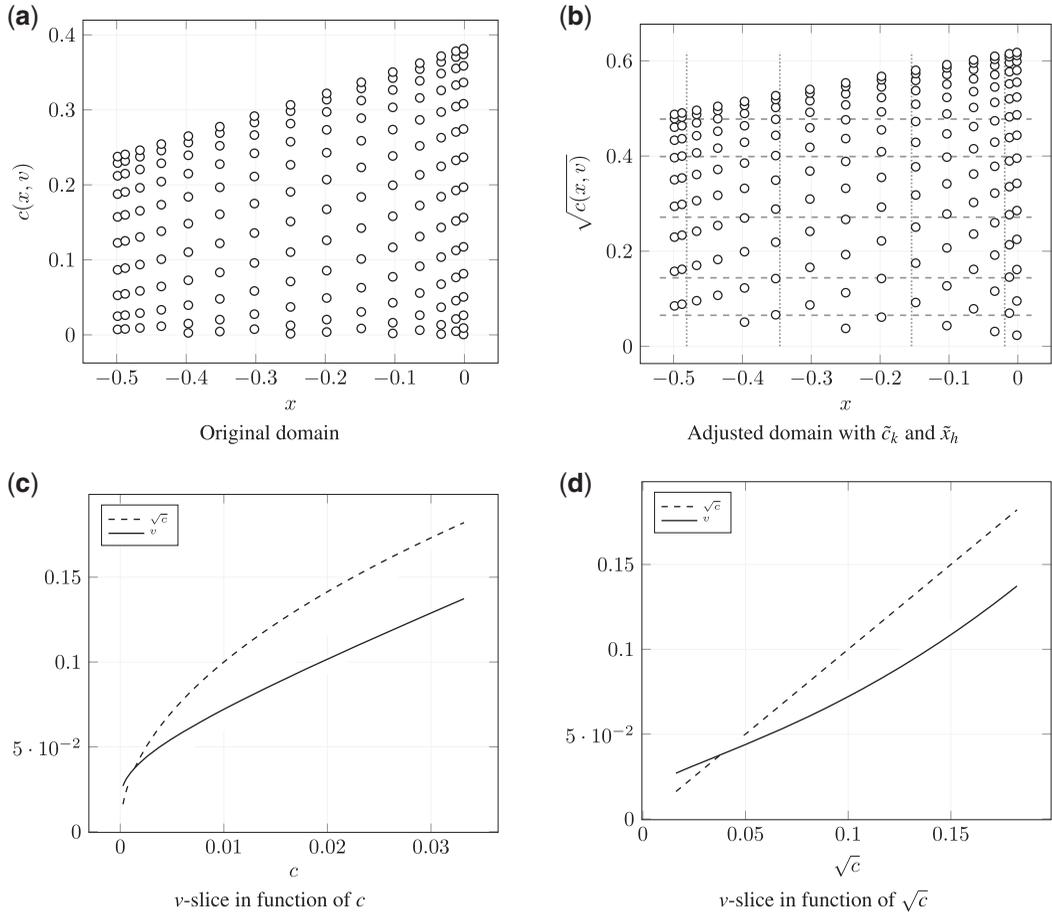


FIG. 1. (Top) Domain(s) of approximation illustrated using a sparse  $(x, v)$  grid. (Bottom) Illustration of square root transformation ( $x = -0.05286307$ ).

$j = 1, \dots, M$  to obtain function values  $c_{i,j} := c(x_i, v_j)$ . Rather than equidistantly, we sample  $(x_i, v_j)$  on a dense Chebyshev grid of size  $N \times M$ . Hence, the  $x_i$  result from  $N$  Chebyshev nodes

$$-\cos(\pi(2i-1)/(2N)), \quad i = 1 \dots, N,$$

which are mapped from the interval  $] -1, 1[$  to the interval  $]a, b[$  through the affine transformation  $1/2(a+b) - 1/2(b-a)\cos(\pi(2i-1)/(2N))$ . In an analogous way, the  $v_j$  are obtained in the interval  $[0, 1]$ . This way, we sample more densely near the corners of the rectangular domain, as illustrated in Fig. 2. We ultimately start from a fairly dense grid with  $N = 300$  and  $M = 600$ , with twice as many samples in  $v$  to ensure that sufficient values  $c(x_i, v_j)$  per fixed  $x_i$  remain after eliminating combinations where  $x_i/v_j < -2$ . From the initial 180 000 grid values, 37 577 values are discarded due to the domain restrictions. Next, we interchange the roles of  $c_{i,j}$  and  $v_j$  and take the square root of  $c_{i,j}$  as illustrated in

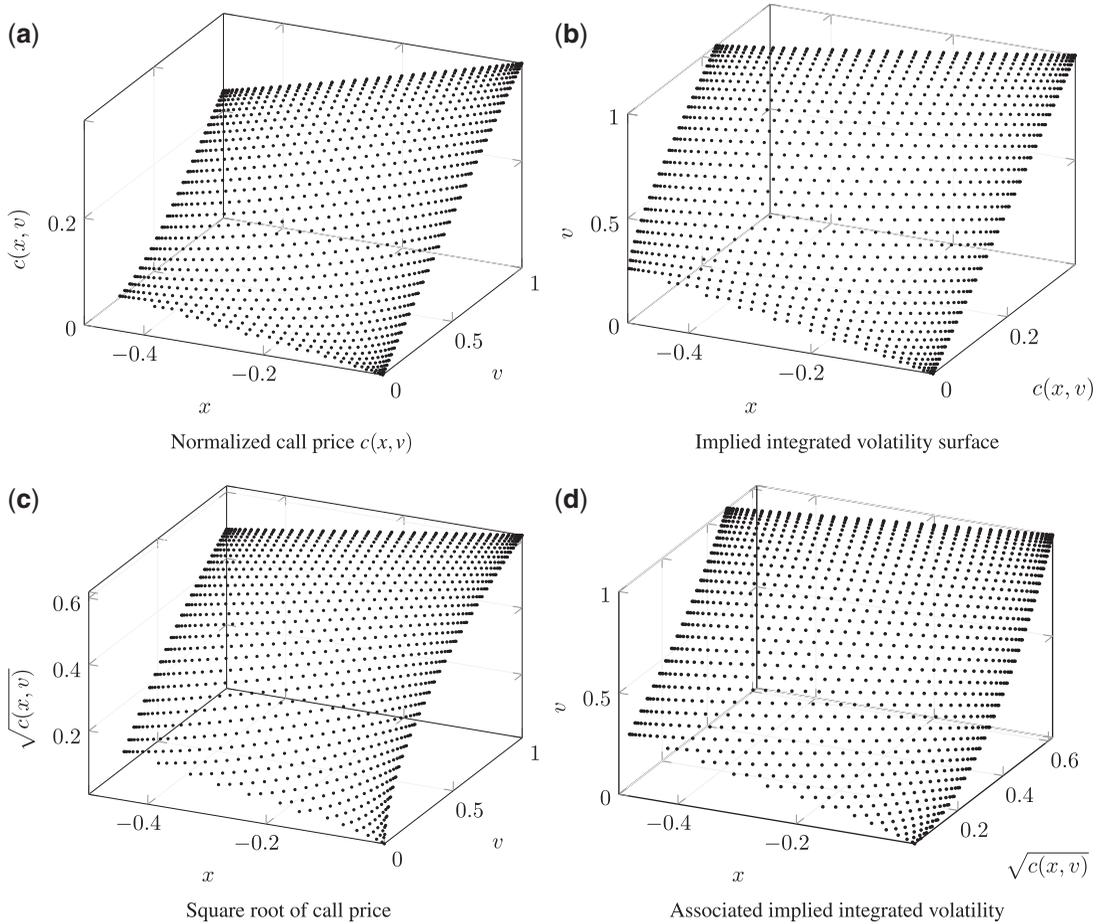


FIG. 2. Normalized call price and implied integrated volatility on a sparse grid.

Fig. 2. As a result, the final grid is only structured in the  $x$ -direction and scattered in the  $c$ -direction. This is illustrated in Figs 1a and 1b.

Note that most of our illustrations are often down-sampled to retain visual clarity.

Although our domain is in fact not exactly rectangular, for ease of notation, we adopt the convention that if a certain  $(x_i, v_j)$  combination lies outside of the domain, then it is discarded and its contribution to a sum or product is ignored.

### 3.2 Functional form

The underlying approximation problem that we consider is the following: determine a bivariate rational function  $R(x, c)$  of the form

$$R(x, c) \equiv \sum_{k=1}^n \frac{p_k^{(\ell)}(x)}{\sqrt{c} - \tilde{c}_k} \tilde{r}_k^{(m)}(x; \tilde{c}_k) \Big/ \sum_{k=1}^n \frac{p_k^{(\ell)}(x)}{\sqrt{c} - \tilde{c}_k}, \quad (3.1)$$

such that

$$R(x_i, c_{ij}) \approx v(x_i, c_{ij}), \quad i = 1, \dots, N, j = 1, \dots, M. \quad (3.2)$$

In (3.1), the  $\tilde{c}_1 < \dots < \tilde{c}_n$  are fixed, but arbitrary chosen distinct values in the domain of  $\sqrt{c}$ , the  $p_k^{(\ell)}$  are univariate polynomials of degree  $\ell$  and the  $\tilde{r}_k^{(m)}$  are univariate rational functions of degree at most  $m$  in numerator and denominator.

One way to consider (3.1) is as a blend of  $n$  univariate rational functions  $\tilde{r}_1^{(m)}, \dots, \tilde{r}_n^{(m)}$  with the  $n$  bivariate rational functions

$$\frac{p_1^{(\ell)}(x)}{\sqrt{c} - \tilde{c}_1}, \dots, \frac{p_n^{(\ell)}(x)}{\sqrt{c} - \tilde{c}_n},$$

acting as blending functions (an interpretation like the one introduced in Floater & Hormann, 2007). After multiplying the numerator and denominator of (3.1) with the monic polynomial

$$(\sqrt{c} - \tilde{c}_1) \cdots (\sqrt{c} - \tilde{c}_n)$$

of degree  $n$  in terms of  $\sqrt{c}$ , one can rewrite the blending functions as

$$p_1^{(\ell)}(x) \prod_{k=1, k \neq 1}^n (\sqrt{c} - \tilde{c}_k), \dots, p_n^{(\ell)}(x) \prod_{k=1, k \neq n}^n (\sqrt{c} - \tilde{c}_k).$$

Hence, it becomes clear that the numerator and denominator of  $R(x, c)$  can be written as bivariate polynomials in terms of  $x$  and  $\sqrt{c}$ .

Another way to consider the bivariate rational function (3.1) is directly as a univariate barycentric rational function in terms of  $\sqrt{c}$  with parametrized weights  $p_k^{(\ell)}(x)$ . For an introduction to barycentric rational and Lagrange interpolation, we refer to Schneider & Werner (1986); Berrut & Trefethen (2004); Trefethen (2013) and the references therein. The barycentric basis has some advantages over the monomial basis, particularly in the parametric context that we envisage here, where approximation errors may accumulate in each step.

First, the weights  $p_k^{(\ell)}$  appear in the denominator exactly as in the numerator, except without the data factors  $\tilde{r}_k^{(m)}$ . This means that any common factor in all the weights  $p_k^{(\ell)}$  may be cancelled without affecting the value of  $R(x, c)$  (see Berrut & Trefethen, 2004). This property is used in Section 3.4.2 and serves to keep the overall complexity manageable. Secondly, it is well known that small perturbations of the weights  $p_k^{(\ell)}$  do not affect the interpolation property at  $\tilde{r}_k^{(m)}$ . This property allows us to maintain a high degree of accuracy when recombining the different components of (3.1) after having computed them separately.

In the remainder of this section, we determine the unknowns occurring in (3.1) progressively. First, the values  $\tilde{c}_1 < \dots < \tilde{c}_n$  are fixed and function values leading to the construction of the  $p_k^{(\ell)}(x)$  and  $\tilde{r}_k^{(m)}(x; \tilde{c}_k)$  are produced in Section 3.3. Section 3.4 is devoted to the actual construction of these functions: the  $\tilde{r}_k^{(m)}(x; \tilde{c}_k)$  are treated in Section 3.4.1, while the  $p_k^{(\ell)}(x)$  are dealt with in Section 3.4.2.

### 3.3 Dependency on $c$

In this section, we fix the values  $\tilde{c}_1 < \dots < \tilde{c}_n$  occurring in (3.1), and calculate coefficients serving as function values for the construction of the functions  $p_k^{(\ell)}(x)$  and  $\tilde{r}_k^{(m)}(x; \tilde{c}_k)$ .

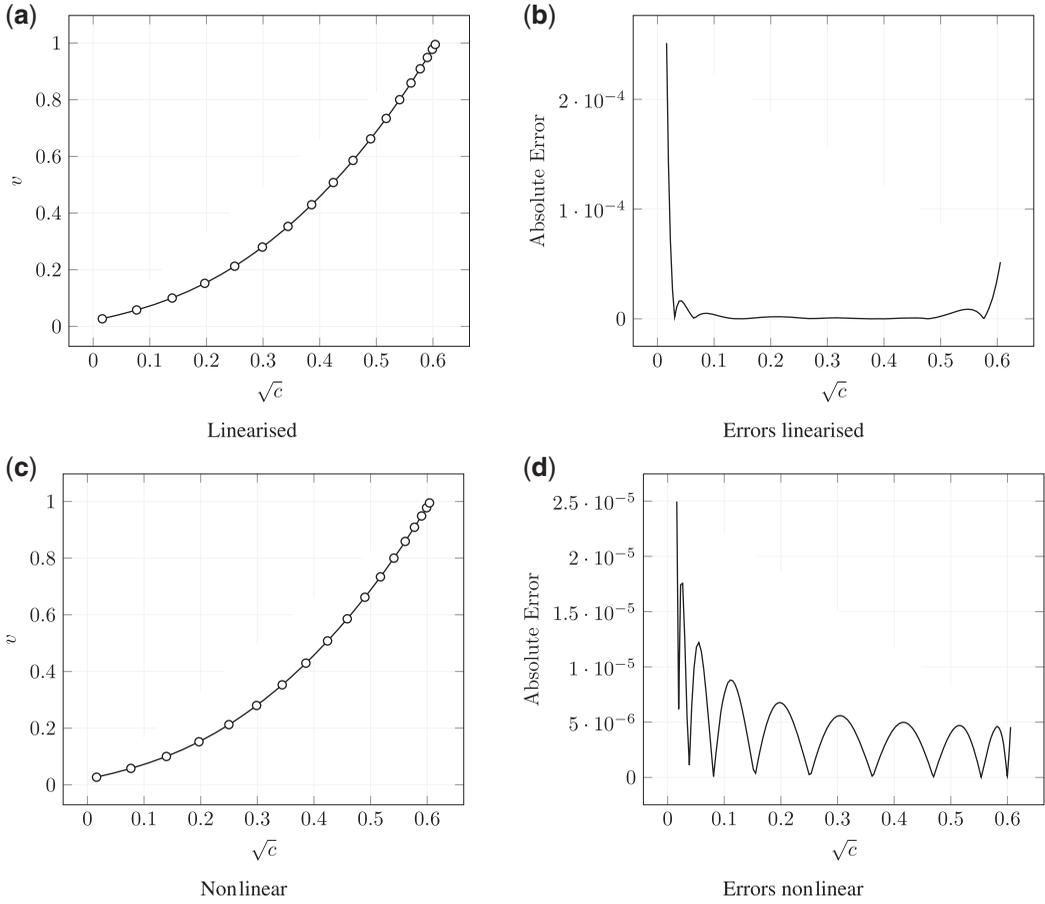


FIG. 3. Rational least squares approximations from (3.4) ( $x = -0.05286307$ ).

The rationale is as follows. Because the  $x_i$ 's can be sampled on a grid, we can fix their unique values and consider the projections of  $v_j$  in terms of  $c_{i,j}$ . Hence, we fix  $x$  and start by approximating the resulting univariate slices in function of  $c$ . An example of a slice near  $x \approx 0$  is shown in Fig. 3. Note that  $v_j$  becomes more nonlinear in  $\sqrt{c}$ , as  $x$  approaches zero and in fact it is known that

$$v(0, c) = \sqrt{8} \operatorname{erf}^{-1}(c), \quad (3.3)$$

where  $\operatorname{erf}^{(-1)}(\cdot)$  denotes the inverse error function.

**3.3.1 Univariate rational approximation.** For each unique  $x_i$  ( $i = 1, \dots, N$ ), we construct a univariate rational approximation for  $v$  in terms of  $c$ . We represent this univariate rational function in barycentric form

$$r_i^{(n)}(c; x_i) \equiv \sum_{k=1}^n \frac{u_{k,i}}{\sqrt{c} - \tilde{c}_k} \Big/ \sum_{k=1}^n \frac{w_{k,i}}{\sqrt{c} - \tilde{c}_k} \approx v(x_i, c), \quad (3.4)$$

where the  $\tilde{c}_1 < \dots < \tilde{c}_n$  are  $n$  Chebychev nodes mapped to the interval  $[0.053, 0.488]$ . Their location is illustrated in Fig. 1b by the horizontal dashed lines. Figure 1b further illustrates that this interval covers the largest common  $\sqrt{c}$  range across different  $x$ -values in our data set. Note that the nodes  $\tilde{c}_k$  merely serve to represent the rational function in a Lagrange basis and their choice is irrelevant. On the other hand, it is well known that the Chebyshev nodes enjoy several attractive properties. For instance, with respect to the numerical stability of evaluating (3.4), they are nearly optimal due to their slow-growing Lebesgue constant (see Higham, 2004).

Like in Knockaert (2008) or Pachón *et al.* (2012) and the related Gonnet *et al.* (2011), our main computational tool is the singular value decomposition (SVD). These methods typically start from rational interpolation and add (linearized) least squares conditions. Knockaert (2008) relies on the barycentric basis and the associated Löwner matrix, while Pachón *et al.* (2012) exploit the structure of the chosen interpolation nodes. Unlike in Knockaert (2008) and Pachón *et al.* (2012), however, we do not have function values at the nodes  $\tilde{c}_k$  at our disposal. As can be seen from Fig. 1b, such values are not available for each fixed  $x_i$ . Hence, we cannot start from interpolation. Notwithstanding, by considering the form (3.4) with weights  $u_{k,i}$  and  $w_{k,i}$ , such function values are not required as we show next.

Given that typically  $M \gg n$ , i.e., the number of function values is much larger than the total degree  $n - 1$  of the rational function, we obtain nonzero weights  $u_{k,i}$  and  $w_{k,i}$  from solving the *linearized* rational least squares problem of  $r_i^{(n)}(c_{i,j}; x_i) \approx v_{i,j}$  for  $j = 1, \dots, M$ . We compute an approximation to the resulting homogeneous problem from the SVD of the following Cauchy-like matrix of size  $M \times 2n$

$$\left( \begin{array}{ccc|ccc} \frac{1}{\sqrt{c_{i,1}} - \tilde{c}_1} & \dots & \frac{1}{\sqrt{c_{i,1}} - \tilde{c}_n} & -\frac{v_{i,1}}{\sqrt{c_{i,1}} - \tilde{c}_1} & \dots & -\frac{v_{i,1}}{\sqrt{c_{i,1}} - \tilde{c}_n} \\ \vdots & \dots & \vdots & \vdots & \dots & \vdots \\ \frac{1}{\sqrt{c_{i,M}} - \tilde{c}_1} & \dots & \frac{1}{\sqrt{c_{i,M}} - \tilde{c}_n} & -\frac{v_{i,M}}{\sqrt{c_{i,M}} - \tilde{c}_1} & \dots & -\frac{v_{i,M}}{\sqrt{c_{i,M}} - \tilde{c}_n} \end{array} \right). \quad (3.5)$$

Hence, we take the right singular vector associated with the smallest singular value of the above matrix. This vector is typically normalized to have Euclidean norm equal to 1. Note that the weights  $u_{k,i}$  and  $w_{k,i}$  obtained this way are only determined up to a common multiple.

In a second step, to improve the accuracy further, we use the obtained weights as a starting point for the truly nonlinear optimization problem that minimizes the Euclidean norm of the residual vector with entries  $r_i^{(n)}(c_{i,j}; x_i) - v_{i,j}$ . To decrease the number of variables, we renormalize by keeping one of the weights fixed, for instance,  $w_{1,i} = 1$ . Hence, we divide by  $w_{1,i} \neq 0$  and optimize the remaining weights. For more information about the employed solver we refer to Section 4. Provided that the initial denominator is free of poles, it is remarkable that the initial solution obtained from the SVD proves a good starting point for this highly nonlinear optimization problem, and typically improves the accuracy considerably after only a few iterations. Although we do not claim to find the actual global optimum, the little overhead due the use of an iterative solver is greatly compensated by the substantial gain in accuracy from the nonlinear least squares solution compared with the linearized solution. This is, for instance, illustrated in Fig. 3. For reasons further explained in Section 3.4.2, we renormalize the optimized weights again to have  $\ell_1$ -norm equal to 1.

With  $n = 5$ , both the SVD and the nonlinear optimization remain numerically manageable, and sufficiently accurate approximations are obtained. An overview of the realized errors across each  $x_i$  is shown in Fig. 4. This figure is obtained after solving  $N$  homogeneous problems, each calculating the SVD of the matrix (3.5) of size  $M \times 2n$  followed by a nonlinear optimization step.

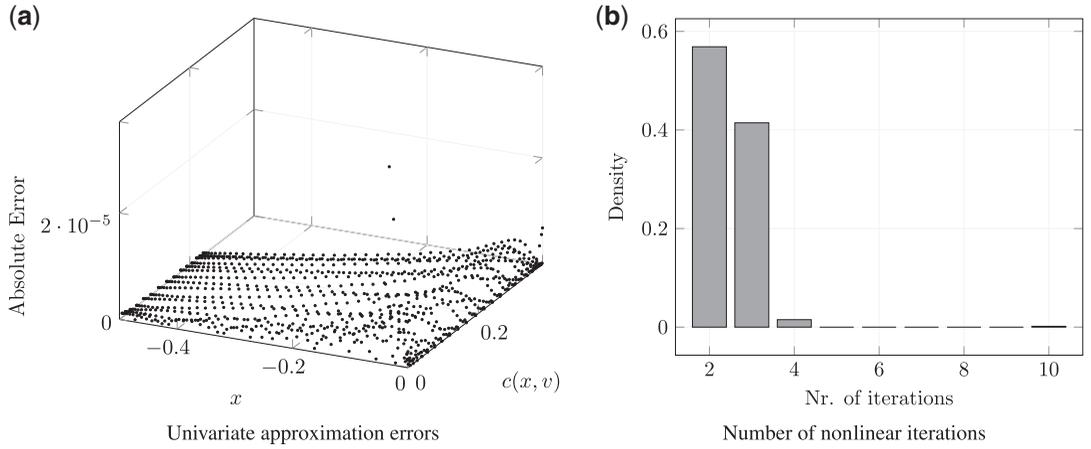


FIG. 4. (Left panel) Errors of the obtained univariate approximations (3.4) with  $n = 5$ . (Right panel) Number of iterations for the truly nonlinear optimization problem.

### 3.4 Dependency on $x$

In this section, we construct the functions  $p_k^{(\ell)}(x)$  and  $\tilde{r}_k^{(m)}(x; \tilde{c}_k)$  occurring in (3.1). To this end, we rely on the previously obtained weights  $u_{k,i}$  and  $w_{k,i}$ .

When we explicitly consider the weights as functions of  $x$ , then we denote them by  $u_k(x)$  and  $w_k(x)$ . One way to proceed would then be to rewrite (3.1) directly as

$$R(x, c) \equiv \sum_{k=1}^n \frac{u_k(x)}{\sqrt{c - \tilde{c}_k}} \bigg/ \sum_{k=1}^n \frac{w_k(x)}{\sqrt{c - \tilde{c}_k}} \quad (3.6)$$

and determine  $u_k(x)$  and  $w_k(x)$  from the values  $u_{k,i}$  and  $w_{k,i}$ . However, this is undesirable for two reasons. First, it ignores the relation that exists between  $u_k(x)$  and  $w_k(x)$ , and may unnecessarily increase the needed complexity to represent  $u_k(x)$ . Indeed, we know that evaluation of (3.6) at  $\tilde{c}_k^2$  gives

$$R(x, \tilde{c}_k^2) = \frac{u_k(x)}{w_k(x)}, \quad k = 1 \dots, n.$$

Since we aim for  $R(x, \tilde{c}_k^2) \approx v(x, \tilde{c}_k^2)$  and, provided that  $w_k(x) \neq 0$ , we may also write

$$u_k(x) \approx v(x, \tilde{c}_k^2) w_k(x), \quad k = 1 \dots, n. \quad (3.7)$$

Hence, each numerator weight function  $u_k(x)$  of (3.6) can be considered as the product of  $v(x, \tilde{c}_k^2)$  with its associated denominator weight function  $w_k(x)$ . This merely reflects the fact that an approximating rational function is fully determined by its denominator. In other words, if denominator weight functions  $w_k(x)$  can be obtained, then it is sufficient to approximate the slices  $v(x, \tilde{c}_k^2)$  in order to recover the numerator weights  $u_k(x)$  for  $k = 1 \dots, n$ . This task is considered in Section 3.4.1.

Secondly, recall that the weights in (3.4) are only determined up to a common multiple. This implies that any  $u_k(x)$  and  $w_k(x)$  obtained from the values  $u_{k,i}$  and  $w_{k,i}$  are most likely only determined up to a

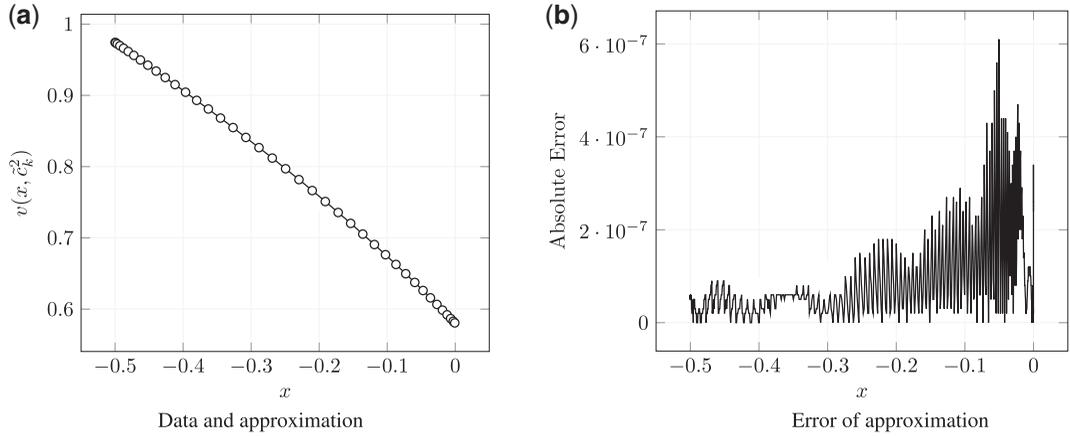


FIG. 5. Example univariate rational approximation (3.8) of degree  $m - 1 = 3$  in numerator and denominator for  $v(x, \tilde{c}_k^2)$  with  $k = 5$  and  $\tilde{c}_k = 0.4776719$ .

common function of  $x$  that can be cancelled out. We explain in Section 3.4.2 how we circumvent this issue.

**3.4.1 Numerator.** Based on the preceding discussion, we are now able to construct the rational functions  $\tilde{r}_k^{(m)}(x; \tilde{c}_k)$  occurring in the numerator of (3.1).

Provided that each previously obtained  $r_i^{(n)}(c; x_i)$  from (3.4) is continuous and a sufficiently good approximations for  $v(x_i, c)$  ( $i = 1, \dots, N$ ), then we have for each fixed  $\tilde{c}_k$  ( $k = 1, \dots, n$ )

$$r_i^{(n)}(\tilde{c}_k^2; x_i) \equiv \frac{u_{k,i}}{w_{k,i}} := \frac{u_k(x_i)}{w_k(x_i)} \approx v(x_i, \tilde{c}_k^2), \quad i = 1 \dots, N.$$

Function values for these slices are available for each  $x_i$  ( $i = 1, \dots, N$ ) from the ratios  $u_k(x_i)/w_k(x_i) \approx v(x_i, \tilde{c}_k^2)$ . We again rely on rational approximations using the same technique as described in Section 3.3.1 to arrive at

$$\tilde{r}_k^{(m)}(x; \tilde{c}_k) \equiv \sum_{h=1}^m \frac{\tilde{u}_{h,k}}{x - \tilde{x}_h} \Big/ \sum_{h=1}^m \frac{\tilde{w}_{h,k}}{x - \tilde{x}_h} \approx v(x, \tilde{c}_k^2). \quad (3.8)$$

Here the  $\tilde{x}_1 < \dots < \tilde{x}_m$  are  $m$  Chebychev nodes mapped to the interval  $[-0.5, 0]$ , the domain of  $x$ . Their location is illustrated in Fig. 1b by the vertical dotted lines. An example of one of the obtained approximations is shown in Fig. 5.

Hence, in this step, we solve  $n$  homogeneous problems, each calculating the SVD of a matrix like (3.5) of size  $N \times 2m$  followed by a nonlinear optimization. Combining then the approximations (3.8) with the factorizations (3.7) and substituting into (3.6) brings us almost to (3.1). The final relation between the polynomials  $p_k^{(l)}(x)$  and the weight functions  $w_k(x)$  is discussed next.

3.4.2 *Denominator.* In this section, we determine the polynomials  $p_k^{(\ell)}(x)$  occurring in (3.1). Their construction depends on the denominator weights  $w_{k,i}$  obtained from (3.4).

Recall that after the factorization (3.7) of the  $u_k(x)$ , the weight functions  $w_k(x)$  appear both in the numerator and the denominator of (3.6). The functions  $w_k(x)$  are therefore only determined up to a common function of  $x$ , which can be cancelled out of (3.6). We would like to isolate and capture only the *true* dependency on  $x$  and  $c$  across the different  $w_k(x)$ . To this end, we express the weight functions  $w_k(x)$  as irreducible rational functions with one *common* denominator. We then determine the coefficients of the different numerators and common denominator simultaneously from the weights  $w_{k,i}$  obtained from (3.4). In this way, we aim to capture the redundant common information in this common denominator and capture the *true* dependency between  $x$  and  $c$  in the numerators—implicitly assuming that it is polynomial. Provided then that the obtained denominator is strictly nonzero, it ultimately cancels out.

For simplicity, we assume that the numerator and denominator degrees are equal and write

$$w_k(x) \equiv \frac{p_k^{(\ell)}(x)}{q^{(\ell)}(x)} = \sum_{l=0}^{\ell} \alpha_{l,k} x^l \bigg/ \sum_{l=0}^{\ell} \beta_l x^l, \quad k = 1, \dots, n. \quad (3.9)$$

The  $n(\ell + 1)$  numerator coefficients  $\alpha_{l,k}$  and  $\ell + 1$  denominator coefficients  $\beta_l$  are in a first step obtained from the simultaneous linearized rational least squares conditions

$$\begin{cases} p_1^{(\ell)}(x_i) &= w_1(x_i) q^{(\ell)}(x_i) \\ \vdots & \vdots \\ p_n^{(\ell)}(x_i) &= w_n(x_i) q^{(\ell)}(x_i) \end{cases}, \quad i = 1, \dots, N.$$

These conditions give rise to a matrix of the form

$$\begin{pmatrix} \mathbf{P} & -\mathbf{W}\mathbf{Q} \end{pmatrix}.$$

Here,  $\mathbf{P} = \text{diag}(\mathbf{V}, \dots, \mathbf{V})$  is a block diagonal-like matrix of total size  $nN \times n(\ell + 1)$ , with  $n$  diagonal entries  $\mathbf{V}$  equal to the Vandermonde matrix of size  $N \times (\ell + 1)$ .  $\mathbf{W} = \text{diag}(\mathbf{W}_1, \dots, \mathbf{W}_n)$  is a block diagonal matrix of total size  $nN \times nN$ , where each  $\mathbf{W}_k$  ( $k = 1, \dots, n$ ) is an  $N \times N$  diagonal matrix with diagonal entries  $w_k(x_i)$  ( $i = 1, \dots, N$ ).  $\mathbf{Q}$  is a block column matrix of total size  $nN \times (\ell + 1)$  of the form

$$\mathbf{Q} = \begin{pmatrix} \mathbf{V} & \dots & \mathbf{V} \end{pmatrix}^T,$$

where T denotes the transpose of a matrix and  $\mathbf{V}$  denotes the Vandermonde matrix of size  $N \times (\ell + 1)$  like before. After multiplication, the matrix  $\mathbf{W}\mathbf{Q}$  is of size  $nN \times (\ell + 1)$ . So that the total size of the resulting homogeneous system is  $nN \times (n + 1)(\ell + 1)$ . Note that this system contains many zero entries and needs to be solved only once.

In line with the previous steps, we add a second nonlinear optimization step to improve the final accuracy. The obtained rational approximations are illustrated in Fig. 6. Since all the approximation errors are essentially accumulated in these coefficients, we did apply a spline smoothing step to obtain a pole-free solution to the simultaneous linearized rational least squares problem. We used the original data again in the nonlinear optimization step. We verified *a posteriori*, through calculating the roots, that the obtained denominator  $q^{(\ell)}(x)$  is strictly positive. A similar check for constant signs has been done

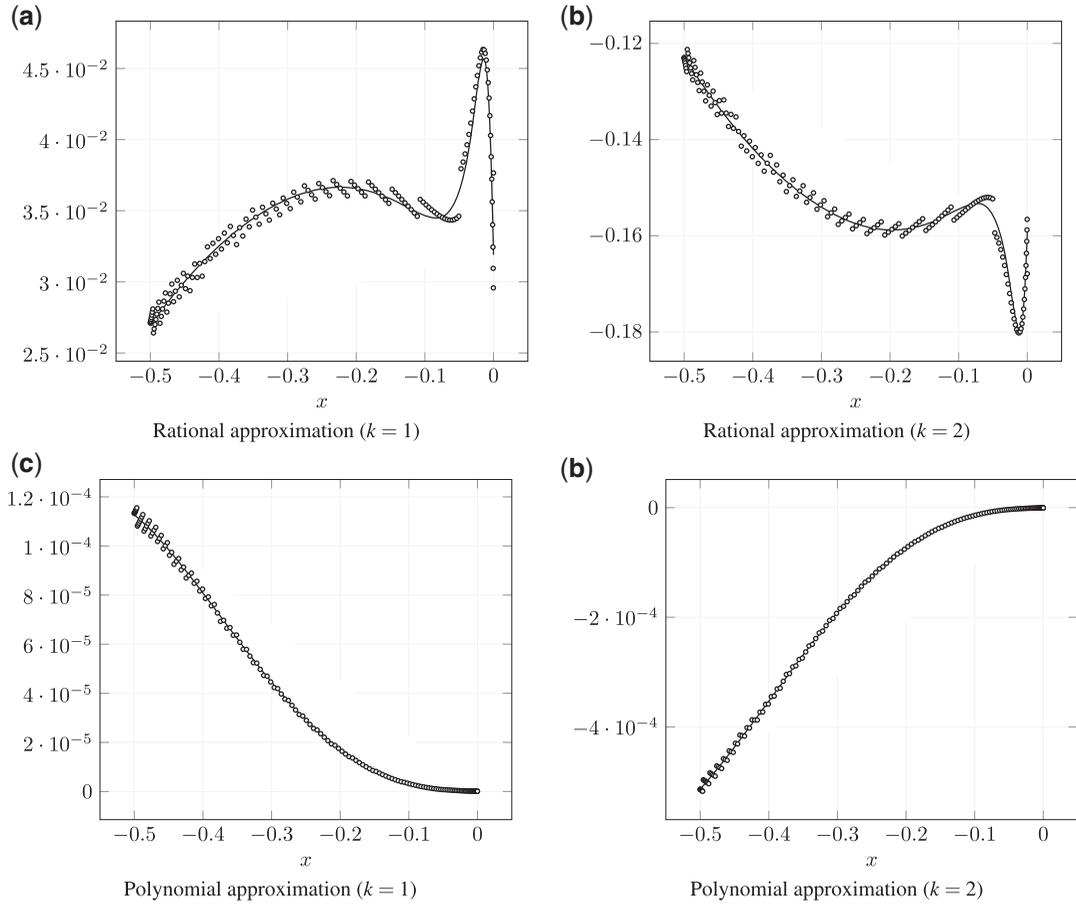


FIG. 6. (Top panels) Example data and rational approximations  $p_k^{(\ell)}(x)/q^{(\ell)}(x)$  of degree  $\ell = 4$  for  $w_k(x)$  with  $k = 1$  and  $k = 2$ . (Bottom panels) Data and final polynomial approximations  $p_k^{(\ell)}(x) \approx w_k(x)q^{(\ell)}(x)$  of degree  $\ell = 4$  with  $k = 1$  and  $k = 2$ , respectively.

for the obtained numerator polynomials  $p_k^{(\ell)}(x)$ . Note that alternating signs for the  $p_k^{(\ell)}(x)$  is a necessary condition for (3.1) to be free of poles in terms of  $x$ .

We further remark that the normalization of the weight values  $w_{k,i}$ , chosen in Section 3.3.1, plays an important role for the interpretation. If, for instance, we kept all  $w_{1,i} = 1$ , then necessarily  $w_1(x) \equiv 1$ . The remaining  $w_k(x)$  with  $k \neq 1$  could still be considered as irreducible rational functions with one common denominator. However, this denominator would not represent common redundant information, but information related to  $w_1(x)$ , which is divided through in the  $w_k(x)$ . Multiplying the numerator and denominator of (3.6) with this common denominator would reintroduce the true  $x$  dependency back to  $w_1(x)$ . From an approximation point of view, it is somewhat awkward to determine a function for  $w_1(x)$  without using explicitly available data for it. Therefore, we adopted a normalization that does not leave any of the weight functions constant.

#### 4. Storage and accuracy comparison

We have implemented the approach in R (see [R Core Team, 2015](#)) on a dense  $N \times M = 300 \times 600$  Chebyshev grid using the `nls` (nonlinear least squares) function. For the latter, the default Gauss–Newton algorithm is used.

We compare our results with the bivariate rational approximation of [Li \(2008\)](#) and the rational approximation method of [Pistorius & Stolte \(2012\)](#). The latter suggests to interpolate in terms of  $x$ , using for instance splines, between pre-calculated univariate rational functions in terms of  $\sqrt{c}$ . Although the philosophy is clearly similar to ours, such an approach is not truly bivariate and does not describe the whole function with one set of parameters.

##### 4.1 Storage comparison

Whereas storage itself is probably less of an issue in modern days, the obtained number of coefficients relates directly to the required number of operations and therefore reflects the speed of evaluation. Although a detailed speed comparison is beyond the scope of the current article, the storage requirements discussed here do give a basic indication on how the considered methods relate in that respect.

With  $n = m - 1 = \ell + 1 = 5$ , our approximation of the form (3.1) requires the storage of 70 coefficients. The obtained parameters for each of the unknowns are detailed in the appendix. Compared with the bivariate rational approximation of [Li \(2008\)](#) with only 31 parameters, we require about twice the number of coefficients. Provided that excess storage is compensated by increased accuracy, such a difference is still moderate and acceptable.

In [Pistorius & Stolte \(2012\)](#), a total of 105 univariate rational functions in terms of  $\sqrt{c}$ , with varying degree from 7 till 9, are used for different values of  $x$ . Not only is additional interpolation required in terms of  $x$ , also more than 1500 coefficients (105 times at least 8 coefficients in the numerator and 7 in the denominator) need to be stored. Compared with truly bivariate approximations, such storage requirements are clearly considerably higher.

##### 4.2 Accuracy comparison

Regarding accuracy, a first aspect is the error of the obtained IV approximation. As illustrated in [Fig. 7](#), our IV approximation is about a factor 5 more accurate in terms of the maximal absolute error compared with the bivariate rational approximation of [Li \(2008\)](#). When comparing the quantiles of the errors in [Table 1](#), then it is seen that for most of the input values across the domain, our IV approximation is nearly 90 times more accurate.

In the approach of [Pistorius & Stolte \(2012\)](#), absolute errors smaller than  $10^{-8}$  for most of the input values are observed, but errors increase to about  $10^{-5}$  near  $x = 0$ . Such behaviour of the absolute error is similar to the one observed in our approach. However, recall that for their method IVs are only available on a grid of given  $x$  values and additional interpolation is needed in between. This property, combined with the excessive storage requirements compared with truly bivariate approximations, leads us not to consider the approach further.

A second aspect regarding accuracy is the error of the inversion itself. Meaning the accuracy of the obtained option price after repricing with (2.2) using an IV approximation. Here we consider the full range  $-1/2 \leq x \leq 1/2$ , where the positive part is priced using the duality (2.4). [Figure 8](#) (left panels) illustrates that repricing errors in this area are not magnified, but rather become smaller when using duality in this way. Compared with the bivariate approximation of [Li \(2008\)](#), our repricing error is about a factor 3 more accurate in terms of the maximal absolute error. When comparing the quantiles of the

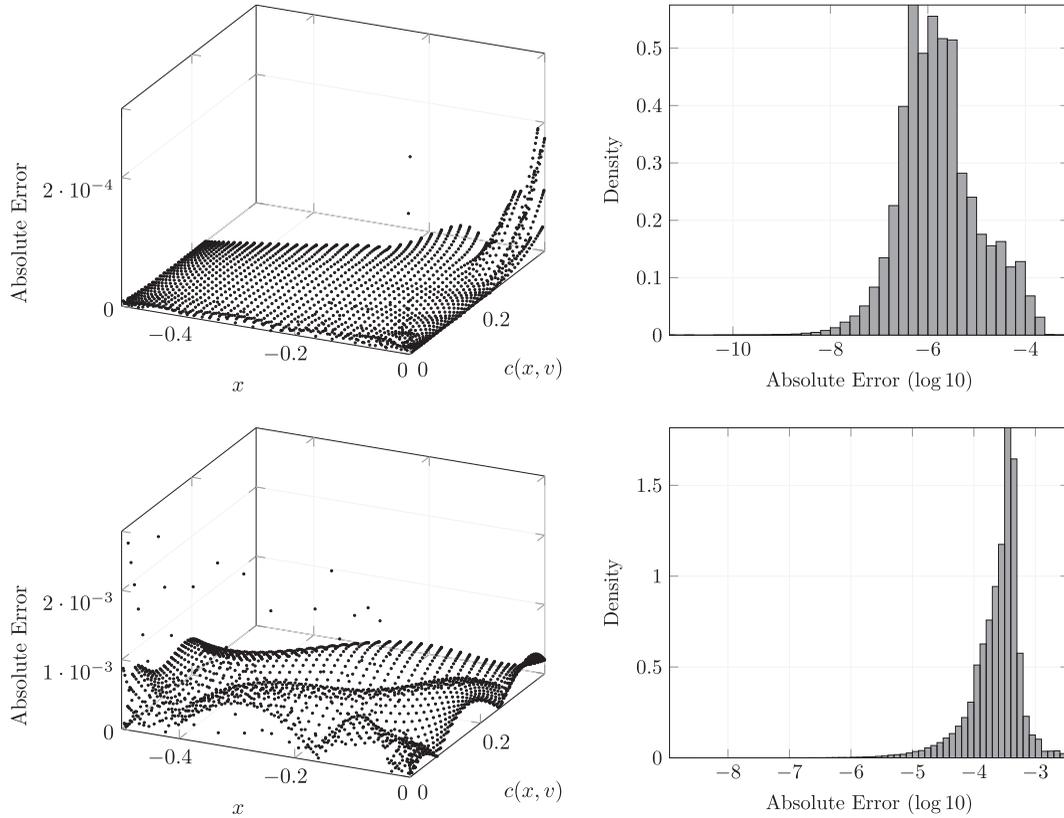


FIG. 7. Bivariate IV error surface and IV error distribution of our final approximation (top panels) vs. the bivariate approximation of Li (2008) (bottom panels).

TABLE 1 *Comparisons of absolute errors for the IV approximations per quantile*

Quantile	Absolute error of (3.1)	Absolute error of Li (2008)
0%	9.673595e-12	1.174385e-09
25%	4.918475e-07	1.588056e-04
50%	1.444778e-06	2.964778e-04
75%	4.721971e-06	4.090227e-04
100%	6.888483e-04	3.242556e-03

errors in Table 2, then it is seen that for most of the input values across the domain our IV approximation is nearly 175 times more accurate.

It is worth noting that an absolute repricing error of the order  $10^{-4}$ , as in the approximation of Li (2008), may be substantial, depending on the magnitude of the underlying. After all, from (2.1) follows that the stock price  $S$  together with a short time to maturity may magnify any inaccuracies present in the approximation for  $c$ . The implication thereof is further illustrated in Section 5.

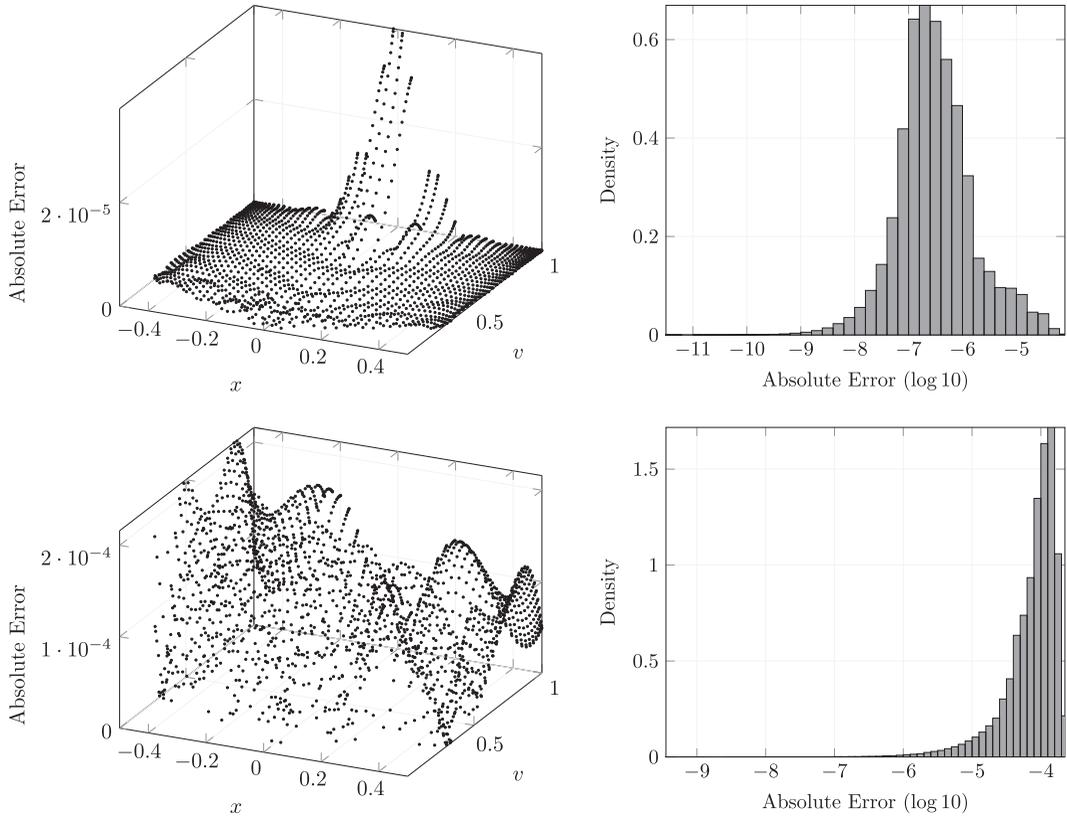


FIG. 8. Bivariate error surface and error distribution after repricing using our final approximation (top panels) vs. the bivariate approximation of Li (2008) (bottom panels).

TABLE 2 *Comparisons of absolute errors after repricing per quantile*

Quantile	Absolute error using (3.1)	Absolute error using Li (2008)
0%	3.048561e-12	3.927062e-10
25%	1.179762e-07	5.255702e-05
50%	2.847269e-07	9.394444e-05
75%	7.601848e-07	1.331565e-04
100%	6.828597e-05	2.166093e-04

## 5. Application to market data

To underline the practical relevance of our approximation, we use SPX option data. We verify whether the revaluation of option prices using the obtained implied volatility yields a value inside the given bid-ask range. Our sample consists of option prices quoted from 1 July 2013 till 31 July 2013 (22 trading days) with several different times to maturity and strikes. For illustration purposes, we only keep the most reliable data: out-of-the-money call (when  $K > F$ ) and put (when  $K < F$ ) prices with nonzero bid-ask

TABLE 3 *Summary statistics for the sample of SPX options. There are 16916 observations in total, of which 6101 are call prices and 10815 are put prices*

Put options (10815)	Min.	Mean	Median	Max.
Price (\$)	0.02500	21.79415	10.15000	186.75000
Time to maturity (years)	0.00392	0.69351	0.33725	3.53725
Bid-ask spread (\$)	0.05000	2.04638	1.30000	15.20000
Norm. price ( $p$ )	0.00001	0.01319	0.00609	0.11711
Moneyness ( $x$ )	0.00001	0.12567	0.08710	0.49992
Call options (6101)	Min.	Mean	Median	Max.
Price (\$)	0.02500	24.08561	12.10000	184.45000
Time to maturity (years)	0.00392	0.75856	0.37255	3.53725
Bid-ask spread (\$)	0.05000	2.09945	1.20000	15.30000
Norm. price ( $c$ )	0.00001	0.01458	0.00729	0.11390
Moneyness ( $x$ )	-0.47383	-0.06316	-0.04413	-0.00000

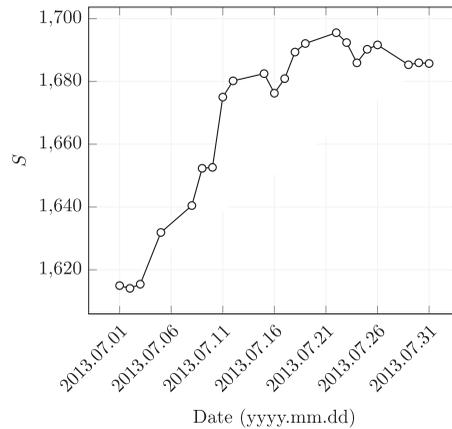


FIG. 9. Daily price levels  $S$  of the S&P 500 Index for July 2013.

spreads. For instance, also the Chicago Board Options Exchange (CBOE) only considers out-of-the-money contracts to calculate the VIX index (see [Chicago Board Options Exchange, 2009](#)). Summary statistics regarding our sample, restricted to the domain of approximation, are given in Table 3. The magnitude of the daily closing values of the SPX are illustrated in Fig. 9.

We proceed as follows. For each call option, we first calculate an implied volatility from the moneyness and mid-price. Using this obtained implied volatility, we then evaluate the call price from (2.2) and check whether that obtained price lies inside the given bid-ask range. For put options, we do the same, but rely on the put-call parity (2.3) and the duality (2.4).

When using the bivariate approximation of Li (2008), we find 864 (about 14.2%) bid-ask violations for call prices and 919 (about 8.49%) for put prices. The nonzero distances to the bid-ask range in function

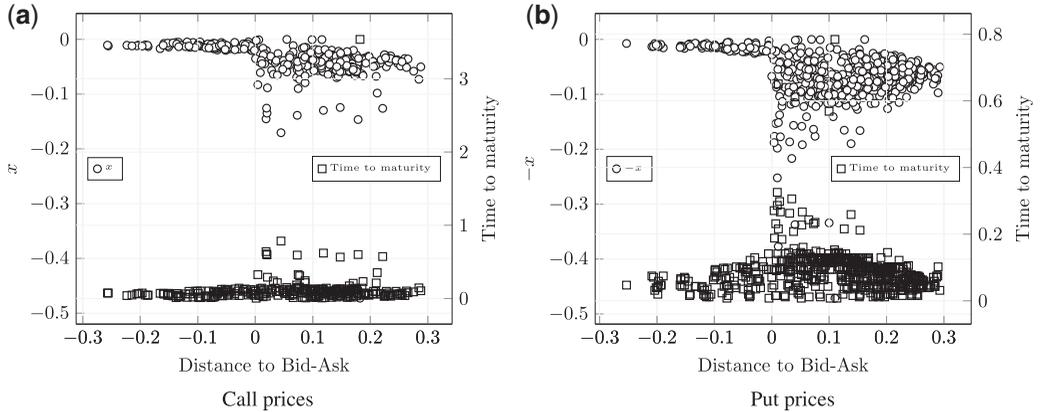


FIG. 10. Nonzero distances to the bid-ask ranges with respect to time to maturity (squares) and moneyness  $x$  (dots). Negative (positive) values on the horizontal axis indicate values below (above) the bid (ask) range.

of  $x$  and time to maturity are illustrated in Fig. 10. Given the error surface of Fig. 7, it is not surprising that violations occur for small (absolute)  $x$  values. It is also not surprising that violations occur for short times to maturity ( $T - t$ ) due to the factor  $S e^{-\delta(T-t)}$ . As mentioned at the end of Section 4.2, from (2.1) follows that a short time to maturity, and a sufficiently large value  $S$  may magnify the repricing error substantially. With a fairly uniform absolute repricing error of the order  $10^{-4}$  and a value  $S$  of the order  $10^3$  (as is the case here), one arrives quickly at a repricing error that is larger than the smallest observed bid-ask spread of 0.05 (cf. Table 3).

The results for our approximation with coefficient given in the appendix are fairly brief: all obtained values lie within the given bid-ask range. From the error surface in Fig. 8, it is seen that the maximal absolute repricing errors of our approximation occur at a particular location: for small absolute  $x$  (near the money) and large  $v$ . From the price surface in Fig. 2a, it is seen that this combination implies higher normalized option prices  $c$ . In practice, such options typically trade with larger tick-size and thus larger bid-ask spread: the CBOE fixes the ticks for all options trading below 3 at 0.05 and for all others at 0.10 (see [Chicago Board Options Exchange, 2016](#)). For a short time to maturity, it would take an underlying at a price  $S = 1464.45$  before the maximal error would become material at a bid-ask spread of 0.1. However, as seen from the statistics of our sample in Table 3, a bid-ask spread of 1 is not uncommon (even for the most actively traded index option in the USA). Moreover, short maturity together with large  $v$  implies high volatility and bid-ask spreads tend to widen even further during high volatility regimes and for less traded options. Hence, from the above together with the fact that only few underlying trade at a price  $S$  above the order  $10^4$ , we may consider our approximation sufficiently accurate for most practical cases.

## 6. Conclusion and discussion

Based on a parametrized barycentric representation, we have constructed an accurate closed-form approximation for the implied volatility. This approximation not only is truly bivariate and strikes a fair balance between accuracy and the required number of coefficients, it can also be evaluated in real time for any set of parameters within the domain. Furthermore, given the pre-calculated coefficients in the appendix, it is very easy to implement. Based on real SPX option data, we find that the existing approximation of Li

(2008) may lead to bid-ask violations when repricing. This adds further weight to the practical relevance of our new approximation.

Although our focus was on the inversion of the Black–Scholes formula, the methodology presented here provides ideas that apply to other (multivariate) inverse approximation problems. Starting from a sequence of univariate rational approximations for a scattered variable, the obtained coefficients in barycentric form serve as data to parametrize in terms of the remaining gridded variable(s). The barycentric form is particularly suited for recombining the results, because small perturbations in the coefficients do not affect its interpolation property at the given nodes.

This does not necessarily imply that the initial univariate approximations need to be solved directly in barycentric form like in (3.5). One could opt for another, possibly better conditioned basis, and recast the obtained rational approximation in barycentric form in a second step. This allows for much more advanced implementations, such as the recent work of [Berljafa & Güttel \(2015\)](#) based on rational Krylov techniques. It is also worth noting that further accuracy can typically be gained by increasing the degrees ( $n$ ,  $m$  and  $\ell$ ) of the involved rational functions. However, degree reductions must be employed to avoid the so-called spurious poles or Froissart doublets (see for instance [Beckermann \*et al.\*, 2016](#), and the references therein). Such phenomena are inherent to the rational approximation problem and their treatment is beyond the scope of this article. We merely bring to mind that overshooting the degree may produce redundant artefacts such as (nearly) common factors that carry no genuine information about the underlying function. These artefacts can then cause spurious discontinuities such as jumps or piecewise behaviour in the evolution of the coefficients. Despite such intrinsic complications, we have introduced several ideas that, when combined, may turn the parametrized barycentric form into an interesting tool to solve practical multivariate approximation problems.

## Acknowledgements

The author thanks Brent Oeyen, Prof. Dr Annie Cuyt, Dr Wen-Shin Lee and Dr İrem Yaman for their encouragements and constructive discussions during the preparation of this article.

## Disclaimer

This material has been prepared for general informational purposes only, and is not intended to be relied upon as accounting, tax, or other professional advice. Please refer to your advisors for specific advice.

## REFERENCES

- BAUMEISTER, J. (2013) Inverse problems in finance. *Recent Developments in Computational Finance: Foundations, Algorithms and Applications* (T. Gerstner & P. Kloeden eds). Interdisciplinary Mathematical Sciences, vol. 14. Singapore: World Scientific Publishing, pp. 81–158.
- BECKERMANN, B., LABAHN, G. & MATOS, A. C. (2016) On rational functions without Froissart doublets. arXiv:1605.00506.
- BERLJAJA, M. & GÜTTEL, S. (2015) The RKFIT algorithm for nonlinear rational approximation. *Manchester Institute for Mathematics Sciences EPrints*, **2015.38**.
- BERRUT, J.-P. & TREFETHEN, L. N. (2004) Barycentric Lagrange interpolation. *SIAM Rev.*, **46**, 501–517.
- BRENNER, M. & SUBRAHMANYAN, M. G. (1988) A simple formula to compute the implied standard deviation. *Financial Analysts Journal*, **44**, 80–83.

- CHAMBERS, D. R. & NAWALKHA, S. K. (2001) An improved approach to computing implied volatility. *Financial Rev.*, **36**, 89–100.
- CHANCE, D. M. (1996) A generalized simple formula to compute the implied volatility. *Financial Rev.*, **31**, 859–867.
- CHICAGO BOARD OPTIONS EXCHANGE (2009) The CBOE Volatility Index - VIX.
- CHICAGO BOARD OPTIONS EXCHANGE (2016) Equity options product specifications.
- CORRADO, C. J. & MILLER, T. W. J. (1996) A note on a simple, accurate formula to compute implied standard deviations. *J. Banking Finance*, **20**, 595–603.
- FLOATER, M. S. & HORMANN, K. (2007) Barycentric rational interpolation with no poles and high rates of approximation. *Numerische Mathematik*, **107**, 315–331.
- GONNET, P., PACHÓN, R. & TREFETHEN, L. N. (2011) Robust rational interpolation and least-squares. *Electron. Trans. Numer. Anal.*, **38**, 146–167.
- HESTON, S. L. (1993) A closed-form solution for options with stochastic volatility with applications to bond and currency options. *Rev. Financ. Stud.*, **6**, 327–343.
- HIGHAM, N. J. (2004) The numerical stability of barycentric Lagrange interpolation. *IMA J. Numer. Anal.*, **24**, 547–556.
- HIGHAM, D. J. (2004) *An Introduction to Financial Option Valuation: Mathematics, Stochastics and Computation*. Cambridge, UK: Cambridge University Press.
- JÄCKEL, P. (2015) Let’s be rational. *Wilmott*, **2015**, 40–53.
- KNOCKAERT, L. (2008) A simple and accurate algorithm for barycentric rational interpolation. *IEEE Signal Process. Lett.*, **15**, 154–157.
- LI, M. (2008) Approximate inversion of the Black–Scholes formula using rational functions. *Eur. J. Oper. Res.*, **185**, 743–759.
- LORIG, M., PAGLIARANI, S. & PASCUCCI, A. (2014) A Taylor series approach to pricing and implied vol for LSV models. *J. Risk*, **17**, 1–17.
- MANASTER, S. & KOEHLER, G. (1982) The calculation of implied variances from the Black–Scholes model: a note. *J. Finance*, **37**, 227–230.
- PACHÓN, R., GONNET, P. & VAN DEUN, J. (2012) Fast and stable rational interpolation in roots of unity and Chebychev points. *SIAM J. Numer. Anal.*, **50**, 1713–1734.
- PISTORIUS, M. & STOLTE, J. (2012) Fast computation of vanilla prices in time–changed models and implied volatilities using rational approximations. *Int. J. Theor. Appl. Finance*, **15**, 1250031.
- R CORE TEAM (2015) *R: A Language and Environment for Statistical Computing*. Vienna, Austria: R Foundation for Statistical Computing.
- SCHNEIDER, C. & WERNER, W. (1986) Some new aspects of rational interpolation. *Math. Comput.*, **47**, 285–299.
- TREFETHEN, L. N. (2013) *Approximation Theory and Approximation Practice*. Philadelphia, USA: Society for Industrial and Applied Mathematics.

## Appendix A. Obtained coefficients

Our approximation (3.4) is of the form

$$R(x, c) \equiv \sum_{k=1}^n \frac{p_k^{(\ell)}(x)}{\sqrt{c} - \tilde{c}_k} \tilde{r}_k^{(m)}(x; \tilde{c}_k) \Big/ \sum_{k=1}^n \frac{p_k^{(\ell)}(x)}{\sqrt{c} - \tilde{c}_k}.$$

### A.1. Dependency on $c$

The values  $\tilde{c}_1 < \dots < \tilde{c}_n$  with  $n = 5$  are given in Table A1.

TABLE A1 Values  $\tilde{c}_1 < \dots < \tilde{c}_n$ 

$k$	$\tilde{c}_k$
1	0.065332896439701332
2	0.144082646599642455
3	0.271502418963989633
4	0.398922191328336839
5	0.477671941488277962

## A.2. Dependency on $x$

A.2.1. *Numerator.* The obtained parameters for the barycentric rational functions (3.8) of the form

$$\tilde{r}_k^{(m)}(x; \tilde{c}_k) \equiv \sum_{h=1}^m \frac{\tilde{u}_{h,k}}{x - \tilde{x}_h} \bigg/ \sum_{h=1}^m \frac{\tilde{w}_{h,k}}{x - \tilde{x}_h}, \quad k = 1, \dots, n,$$

where  $m = 4$  and  $n = 5$  are given in Tables A2 and A3.

TABLE A2 Values  $\tilde{x}_1 < \dots < \tilde{x}_m$ 

$h$	$\tilde{x}_h$
1	-0.480969883127821685
2	-0.345670858091272404
3	-0.154329141908727541
4	-0.019030116872178315

With the normalization  $\tilde{w}_{1,k} = 1$ , it requires the storage of 40 coefficients. If desired, this can be reduced to 36 coefficients when writing the numerators and denominators of (3.8) in monomial form for  $k = 1, \dots, n$

$$\tilde{r}_k^{(m)}(x; \tilde{c}_k) \equiv \frac{\sum_{h=1}^m \frac{\tilde{u}_{h,k}}{x - \tilde{x}_h}}{\sum_{h=1}^m \frac{\tilde{w}_{h,k}}{x - \tilde{x}_h}} \equiv \frac{\tilde{p}_k^{(m)}(x)}{\tilde{q}_k^{(m)}(x)} = \frac{\sum_{l=0}^m \tilde{\alpha}_{l,k} x^l}{\sum_{l=0}^m \tilde{\beta}_{l,k} x^l},$$

with coefficient values as in Table A4.

TABLE A3 *Weights  $\tilde{u}_{h,k}$  and  $\tilde{w}_{h,k}$* 

$k$	$\tilde{u}_{1,k}$	$\tilde{u}_{2,k}$	$\tilde{u}_{3,k}$	$\tilde{u}_{4,k}$
1	0.2614787548	-0.2761063691	0.0433222165	-0.0005589997
2	0.3680452668	-0.4250280929	0.0878671762	-0.0031866801
3	0.5553284558	-0.7433208602	0.2328763965	-0.0240908987
4	0.7873068196	-1.2065629413	0.5267842738	-0.0946549001
5	0.9617625469	-1.5436931441	0.7523959252	-0.1583119694
$k$	$\tilde{w}_{1,k}$	$\tilde{w}_{2,k}$	$\tilde{w}_{3,k}$	$\tilde{w}_{4,k}$
1	1.0000000000	-1.3661012546	0.3950301839	-0.0197530550
2	1.0000000000	-1.4393159661	0.4908071459	-0.0436874644
3	1.0000000000	-1.5838703489	0.7052132116	-0.1167445964
4	1.0000000000	-1.7360002657	0.9626898518	-0.2244914104
5	1.0000000000	-1.7773173191	1.0437551467	-0.2643429903

TABLE A4 *Coefficients  $\tilde{\alpha}_{l,k}$  and  $\tilde{\beta}_{l,k}$* 

$k$	$\tilde{\alpha}_{0,k}$	$\tilde{\alpha}_{1,k}$	$\tilde{\alpha}_{2,k}$	$\tilde{\alpha}_{3,k}$
1	0.01072688	-1.89038287	51.66929061	-164.05900046
2	0.05204818	-2.56518231	27.13455783	-47.26515657
3	0.18503755	-3.04559569	12.50524291	-10.46791405
4	0.40158350	-3.22802264	6.79818305	-3.10106719
5	0.57996554	-3.54143625	6.01322745	-2.44257239
$k$	$\tilde{\beta}_{0,k}$	$\tilde{\beta}_{1,k}$	$\tilde{\beta}_{2,k}$	$\tilde{\beta}_{3,k}$
1	1.00000000	-64.31913976	350.80166680	-53.50462150
2	1.00000000	-25.78200971	86.26369389	-13.31678198
3	1.00000000	-10.16113754	17.86534126	-2.31491563
4	1.00000000	-5.35998208	5.53715189	-0.52952359
5	1.00000000	-4.36675918	4.13205235	-0.42101873

### A.2.2. Denominator

The coefficients obtained from (3.9) for the polynomials  $p_k^{(\ell)}(x)$  of the form

$$p_k^{(\ell)}(x) = \sum_{l=0}^{\ell} \alpha_{l,k} x^l$$

with  $\ell = 4$  for  $k = 1, \dots, n$  and  $n = 5$  are given in Table A5.

TABLE A5 *Coefficients*  $\alpha_{l,k}$ 

$k$	$\alpha_{0,k}$	$\alpha_{1,k}$	$\alpha_{2,k}$	$\alpha_{3,k}$	$\alpha_{4,k}$
1	9.093088e-08	-7.716775e-06	6.574232e-06	-2.558192e-03	-3.402852e-03
2	-4.541491e-07	1.855219e-05	-3.770090e-04	9.112664e-03	1.167112e-02
3	9.250607e-07	-1.648322e-05	1.360401e-03	-1.319014e-02	-1.312207e-02
4	-9.714459e-07	3.969999e-06	-2.049026e-03	9.822552e-03	2.594871e-03
5	4.090481e-07	1.543441e-06	1.049118e-03	-3.183720e-03	2.153617e-03