Mathematical Modelling and Analysis
Volume 12 Number 4, 2007, pages 515–538
 ISSN 1392-6292 print, ISSN 1648-3510 online

FAST SPLINE QUASICOLLOCATION SOLVERS OF INTEGRAL EQUATIONS

G. VAINIKKO and I. ZOLK

University of Tartu, Institute of Mathematics

J. Liivi 2, 50409 Tartu, Estonia

E-mail: gennadi.vainikko@ut.ee; indrek.zolk@ut.ee

Received September 1, 2005; revised October 15, 2005; published online December 15, 2007

Abstract. A fast (C, C^m) solver for linear Fredholm integral equations $u = Tu + f$ with smooth data is constructed on the basis of a discrete version of the spline quasicollocation method. By a fast (C, C^m) solver we mean a discrete method that meets the optimal accuracy for $f \in C^m$ with minimal arithmetic work.

Key words: fast solvers, Fredholm integral equation, complexity, quasicollocation method, two grid iterations, splines, interpolation, quasi-interpolation

1. Introduction

Consider the Fredholm integral equation

$$u(x) = \int_0^1 K(x, y)u(y)\, dy + f(x), \quad 0 \le x \le 1, \tag{1.1}$$

where $f \in C^m[0, 1]$ and $K \in C^{2m}([0, 1] \times [0, 1])$, $m \in \mathbb{N}$. Assume that the corresponding homogeneous equation $u = Tu$ has in $C[0, 1]$ only the trivial solution $u = 0$; here $T = T_K$ denotes the integral operator of equation (1.1).

We are interested in fast solvers for equation (1.1). In literature, the meaning of a fast solver often varies. In the present paper, we use a notion of a fast solver which meets the optimal accuracy of the solver with the minimal amount of the arithmetical work (we mean the order optimalities), cf. [17, 19, 20, 21]. Below we assume that the information about f and K is restricted to $n_\star$ sample values; the evaluation points for f and K depend on the solver of equation (1.1). A solver is called *information optimal* on a class of problems (1.1) if (disregarding the amount of the arithmetical work) its accuracy on the class is of the same order as the lower bound of the error over all solvers with $n_\star$ sample values of f and K. We proclaim a solver to be *fast* if it is information optimal and its implementation costs $b_m n_\star$ flops where the constant b_m is independent of $n_\star$. In addition, we require a fast evaluation of the approximate solution at a given point. More precisely, by a *fast* (C, C^m) *solver* we mean a solver that produces approximate solutions u_n, $n \in \mathbb{N}$, such that

- given the values of f at not more than $n_\star$ points and the values K at not more than $n_\star$ points (determined by the solver, with the property that $n_\star = n_\star(n) \to \infty$ as $n \to \infty$), the parameters of u_n can be determined at the cost of $b_m n_\star$ flops, and an accuracy

$$\|u - u_n\|_\infty \le c_{m,n} n_\star^{-m} \|f\|_{m,\infty}, \qquad c_m := \limsup_{n\to\infty} c_{m,n} < \infty \tag{1.2}$$

  is achieved where $u = (I - T)^{-1} f$ is the solution of (1.1) and the constant c_m is independent of f (it may depend only on m and K);
- having the parameters of u_n in hand, the value of u_n at any point $x \in [0,1]$ is available at the cost of b'_m flops (b'_m depends only on m).

Here $\|u\|_\infty = \sup_{0\le x\le 1} |u(x)|$ and $\|u\|_{m,\infty} = \max_{0\le k\le m} \left\|u^{(k)}\right\|_\infty$ are the norms in $C[0,1]$ and $C^m[0,1]$, respectively. We have not set a condition that in (1.2) also $c_{m,n}$ are independent of f; setting this condition we obtain a more strict notion of a fast (C, C^m) solver.

Estimate (1.2) can be rewritten with respect to the complexity $n_{\star\star} := b_m n_\star$ of the fast (C, C^m) solver in the form

$$\begin{aligned} &\|u - u_n\|_\infty \le \overline{c}_{m,n} n_{\star\star}^{-m} \|f\|_{m,\infty}, \qquad \overline{c}_{m,n} = b_m^m c_{m,n}, \\ &\overline{c}_m := \limsup_{n\to\infty} \overline{c}_{m,n} < \infty. \end{aligned} \tag{1.3}$$

This form of the estimate enables a comparison of fast (C, C^m) solvers with different complexity parameters b_m, i.e. the smaller $\overline{c}_m$ corresponds to the more effective solver.

It is not known what are the smallest values of c_m and $\overline{c}_m$ in (1.2) and (1.3) over all fast (C, C^m) solvers. On the other hand, it is relatively easy to prove (cf. [16, 17]) that for any solver using the sample values of f on a uniform grid in $[0,1]$ consisting of $n_\star$ points (whereas the information about K and the arithmetical work may be unrestricted), there is a "bad" nonzero function $f \in C^m[0,1]$ such that

$$\|u - u_n\|_\infty \ge \frac{\Phi_{m+1}}{\|I - T_K\|_{C\to C}} \pi^{-m} n_\star^{-m} \|f\|_{m,\infty}. \tag{1.4}$$

where

$$\Phi_m = \frac{4}{\pi} \sum_{k=0}^{\infty} \frac{(-1)^{km}}{(2k+1)^m}, \quad m \in \mathbb{N}, \tag{1.5}$$

is the Favard constant,

$$1 = \Phi_1 < \Phi_3 < \Phi_5 < \cdots < \frac{4}{\pi} < \cdots < \Phi_6 < \Phi_4 < \Phi_2 = \frac{\pi}{2}, \qquad \lim_{m\to\infty} \Phi_m = \frac{4}{\pi}.$$

Thus $c_m \geq \Phi_{m+1}\pi^{-m}/\|I - T_K\|_{C\to C}$ in (1.2) if uniform grids are used for f. When we restrict the information about K and the arithmetical work in accordance to the definition of the fast (C, C^m) solver, the value of the constant c_m in (1.2) is expected to increase. In Sections 3–5 we succeed in construction of a spline quasicollocation solver with $c_m = \Phi_{m+1}\pi^{-m}d_{K,m}$ where the constant $d_{K,m}$ is determined by the kernel K through the quantities $\|K\|_{2m,\infty}$ and $\left\|(I - T_K)^{-1}\right\|_{C\to C}$, and $d_{K,m} \to 1$ as $\|K\|_{2m,\infty} \to 0$. Thus for small $\|K\|_{2m,\infty}$, upper estimate (1.2) for the proposed solver and lower estimate (1.4) for any solver are close to one another.

Estimate (1.4) holds true disregarding the smoothness of the kernel K. The role of the smoothness of K becomes clear due to the following result of Werschultz [21] published in 2003 but having a long prehistory: for any solver of (1.1) with $n_\star$ evaluation points for f and K, disregarding the amount of the arithmetical work, in any class of kernels

$$\mathcal{K}^{m'}_{\gamma,\kappa} = \left\{K \in C^{m'}([0,1]\times[0,1]) \,:\, \|K\|_{C^{m'}} \leq \gamma, \quad \left\|(I - T_K)^{-1}\right\|_{C\to C} \leq \kappa\right\},$$

$m' \geq 1,\ \ \gamma > 0,\ \ \kappa > 1,$

there is a "bad" K such that even for $f \equiv 1$, it holds

$$\|u - u_n\|_\infty \geq c^0_m n_\star^{-\min\{m,m'/2\}}\|f\|_{m,\infty}i, \tag{1.6}$$

where c^0_m is a positive constant depending only on m, m', γ and κ; a self-contained proof can be found also in lecture notes [16]. According to (1.6), under the traditional assumptions $f \in C^m[0,1]$, $K \in C^m([0,1]\times[0,1])$, i.e., for $m' = m$, the accuracy $O(n_\star^{-m/2})$ and not more can be achieved by any solver. This partial result is relatively simple, it has been established already in 1967 by Emel'yanov and Il'in [3]. A further consequence of (1.6) is that accuracy (1.2) is possible only if $m' \geq 2m$. This explains the constellation $m' = 2m$ of our smoothness assumptions $f \in C^m[0,1]$, $K \in C^{2m}([0,1]\times[0,1])$.

It is of common interest to rearrange known methods into fast (C, C^m) solvers and characterise their effectiveness by the constant $\overline{c}_m$ in (1.3). Some results in this direction can be found in [21] (Galerkin method with discontinuous piecewise polynomial approximation of the solution), [19] (Nyström method), [17, 20] (wavelet approximations). The constant c_m in (1.2) and the effectiveness constant $\overline{c}_m$ in (1.3) have not been characterised explicitly in these works. In the present paper we introduce a new fast (C, C^m) solver based on the quasi-interpolation by smooth splines, and we expose the parameters c_m, b_m and $\overline{c}_m$ of this solver explicitly. It can be shown that the effectiviness constant $\overline{c}_m$ of the constructed solver is smaller than those for the solvers treated in [17, 19, 20, 21] but we cannot present corresponding arguments here – they need a revisiting of long considerations of previous works in a more detailed level; an interested reader can find the details and a theoretical comparison of the mentioned methods and some further methods in the lecture notes [16].

Perhaps the proposed quasicollocation method is of interest not only in the complexity analysis but also in the practical solving of integral equations with smooth kernels. Numerical examples are presented in Section 6.

2. Approximation of Functions by Splines

2.1. The father B-spline

The *father* B*-spline* B_m *of order* m in the terminology of [2, 12], or of degree $m-1$ in the terminology of [5, 13, 22] can be defined by the formula

$$B_m(x) = \frac{1}{(m-1)!} \sum_{i=0}^{m} (-1)^i \binom{m}{i} (x-i)_+^{m-1}, \quad x \in \mathbb{R}, \quad m \in \mathbb{N}. \tag{2.1}$$

Here, as usual, $0! = 1$, $0^0 := \lim_{x \downarrow 0} x^x = 1$,

$$\binom{m}{i} = \frac{m!}{i!\,(m-i)!}, \qquad (x-i)_+^{m-1} = \begin{cases} (x-i)^{m-1}, & x-i \geq 0, \\ 0, & x-i < 0. \end{cases}$$

Let us recall some properties of B_m:

$$\operatorname{supp} B_m = [0, m], \quad B_m(x) = B_m(m-x) > 0 \quad \text{for } 0 < x < m, \tag{2.2}$$

$$B_m \in C^{(m-2)}(\mathbb{R}), \quad B_m^{(m-1)}(x) = (-1)^l \binom{m-1}{l}, \quad \text{for } l < x < l+1, \ \ l = 0, \ldots, m-1,$$

$$\int_{\mathbb{R}} B_m(x)\, dx = 1, \quad \sum_{j \in \mathbb{Z}} B_m(x-j) = 1, \quad x \in \mathbb{R}.$$

2.2. Spline interpolation on the uniform grid in $\mathbb{R}$

Introduce in $\mathbb{R}$ the uniform grid $h\mathbb{Z} = \{ih : i \in \mathbb{Z}\}$ of the step size $h > 0$. Denote by $S_{h,m}$, $m \in \mathbb{N}$, the space of splines of order m (of degree $m-1$) and defect 1 with the knot set $h\mathbb{Z}$. The family of B-splines $B_m\left(h^{-1}x - j\right)$, $j \in \mathbb{Z}$, belong to $S_{h,m}$, and the same is true for $\sum_{j \in \mathbb{Z}} d_j B_m\left(h^{-1}x - j\right)$ with arbitrary coefficients d_j; note that the series is locally finite: it follows from (2.2) that

$$\sum_{j \in \mathbb{Z}} d_j B_m\left(h^{-1}x - j\right) = \sum_{j=i-m+1}^{i} d_j B_m\left(h^{-1}x - j\right) \text{ for } x \in [ih, (i+1)h), \ i \in \mathbb{Z}.$$

Given a function $f \in C(\mathbb{R})$, bounded or of at most polynomial growth as $|x| \to \infty$, we determine the interpolant $Q_{h,m}f \in S_{h,m}$ by the conditions

$$(Q_{h,m}f)(x) = \sum_{j \in \mathbb{Z}} d_j B_m\left(h^{-1}x - j\right),$$

$$(Q_{h,m}f)\left(\left(k + \frac{m}{2}\right)h\right) = f\left(\left(k + \frac{m}{2}\right)h\right), \quad k \in \mathbb{Z}. \tag{2.3}$$

For $m = 1$ and $m = 2$, $Q_{h,m}f$ is the usual piecewise constant, respectively, piecewise linear interpolant which can be determined on every subinterval $[ih, (i+1)h)$, $i \in \mathbb{Z}$, independently of other subintervals. For $m \geq 3$, the value of $Q_{h,m}f$ at a given point $x \in \mathbb{R}$ depends on the values of f at all interpolation knots $\left(k + \frac{m}{2}\right) h$, $k \in \mathbb{Z}$. It occurs (see [11, 13]) that for $m \geq 3$ conditions (2.3) really determine d_j, $j \in \mathbb{Z}$, uniquely in the space of bounded or polynomially growing bisequences (d_j), namely,

$$d_j = \sum_{k \in \mathbb{Z}} \alpha_{j-k,m} f\left(\left(k + \frac{m}{2}\right) h\right) = \sum_{k \in \mathbb{Z}} \alpha_{k,m} f\left(\left(j - k + \frac{m}{2}\right) h\right), \; j \in \mathbb{Z}, \quad (2.4)$$

where

$$\alpha_{k,m} = \sum_{l=1}^{m_0} \frac{z_{l,m}^{m_0-1}}{P_m'(z_{l,m})} z_{l,m}^{|k|}, \quad k \in \mathbb{Z}, \quad m_0 = \begin{cases} (m-2)/2 & \text{if } m \text{ is even}, \\ (m-1)/2 & \text{if } m \text{ is odd}, \end{cases}$$

and $z_{l,m} \in (-1, 0)$, $l = 1, \ldots, m_0$, are the roots of the characteristic polynomial

$$P_m(z) = \sum_{|k| \leq m_0} B_m\left(k + \frac{m}{2}\right) z^{k+m_0}$$

(it is a polynomial of degree $2m_0$). It occurs that P_m has exactly m_0 simple roots $z_{l,m}$, $l = 1, \ldots, m_0$, in the interval $(-1, 0)$, and the remaining m_0 roots are of the form $z_{l+m_0,m} = 1/z_{l,m} \in (-\infty, -1)$, $l = 1, \ldots, m_0$.

Denote by $BC(\mathbb{R})$ the space of bounded continuous functions on $\mathbb{R}$ equipped with the norm $\|f\|_\infty = \sup_{x \in \mathbb{R}} |f(x)|$, by $V^{m,\infty}(\mathbb{R})$ the space of functions having bounded mth (distribution) derivative in $\mathbb{R}$, by $W^{m,\infty}(\mathbb{R})$ the standard Sobolev space of functions on $\mathbb{R}$ having bounded derivatives of order $\leq m$, and by $W^{m,\infty}_{(0,1)}(\mathbb{R})$ the space of functions $f \in W^{m,\infty}(\mathbb{R})$ with supports in $(0, 1)$.

Lemma 1. *For $f \in V^{m,\infty}(\mathbb{R})$, it holds $f - Q_{h,m}f \in BC(\mathbb{R})$ and*

$$\|f - Q_{h,m}f\|_\infty \leq \Phi_{m+1} \pi^{-m} h^m \left\| f^{(m)} \right\|_\infty, \quad (2.5)$$

where Φ_m is the Favard constant (1.5).

Estimate (2.5) is established in [5] for 1-periodic f, $h = 1/n$ with even $n \in \mathbb{N}$, and in [15] in general case. The following lemma proved in [15] tells that, in some sense, the spline interpolation yields the best approximation of the function classes $W^{m,\infty}(\mathbb{R})$ and $V^{m,\infty}(\mathbb{R})$, asymptotically also of $W^{m,\infty}_{(0,1)}(\mathbb{R})$, compared with other methods that use the same information as $Q_{h,m}f$ – the values $f|_{\mathbb{Z}_{h,m}}$ of f on the grid $\mathbb{Z}_{h,m} = \left\{\left(j + \frac{m}{2}\right) h : j \in \mathbb{Z}\right\}$. Denote by $C(\mathbb{Z}_{h,m})$ the vector space of all (grid) functions defined on $\mathbb{Z}_{h,m}$.

Lemma 2. *For given $\gamma > 0$, we have by Lemma* 1

$$\sup_{\substack{f \in V^{m,\infty}(\mathbb{R}) \\ \|f^{(m)}\|_\infty \leq \gamma}} \|f - Q_{h,m}f\|_\infty \leq \Phi_{m+1} \pi^{-m} h^m \gamma,$$

whereas for any mapping $M_h : C(\mathbb{Z}_{h,m}) \to C(\mathbb{R})$ (linear or nonlinear, continuous or discontinuous), it holds

$$\sup_{\substack{f\in W^{m,\infty}(\mathbb{R})\\ \|f^{(m)}\|_\infty\le\gamma}} \left\|f - M_h\left(f|_{\mathbb{Z}_{h,m}}\right)\right\|_\infty \ge \Phi_{m+1}\pi^{-m}h^m\gamma,$$

$$\liminf_{h\to 0} \sup_{\substack{f\in W^{m,\infty}_{(0,1)}(\mathbb{R})\\ \|f^{(m)}\|_\infty\le\gamma}} \frac{\|f - M_h\left(f|_{\Delta_h}\right)\|_\infty}{\Phi_{m+1}\pi^{-m}h^m\gamma} \ge 1.$$

2.3. Quasi-interpolation by splines

Let $m \ge 3$. In a quasi-interpolant $Q^{(p)}_{h,m}f$, $p \in \mathbb{N}$, the infinite sum (2.4) defining the coefficients d_j of the spline interpolant (2.3) is replaced by a finite sum:

$$\left(Q^{(p)}_{h,m}f\right)(x) = \sum_{j\in\mathbb{Z}} d^{(p)}_j B_m\left(h^{-1}x - j\right),$$

$$d^{(p)}_j = \sum_{|k|\le p-1} \alpha^{(p)}_{k,m} f\left(\left(j - k + \tfrac{m}{2}\right)h\right).$$

A simple truncation of the series in (2.4) does not give acceptable results. Using a special difference calculus for fast decaying bisequences, the following formulae for $\alpha^{(p)}_{k,m}$ have been proposed in [6]:

$$\alpha^{(p)}_{k,m} = \sum_{q=|k|}^{p-1} (-1)^{k+q}\binom{2q}{k+q}\gamma_{q,m}, \quad |k| \le p-1,$$

$$\gamma_{0,m} = 1, \qquad \gamma_{q,m} = \sum_{l=1}^{m_0} \frac{(1+z_{l,m})z_{l,m}^{m_0+q-1}}{(1-z_{l,m})^{2q+1}P'_m(z_{l,m})}, \quad q \ge 1.$$

Then it occurs that

$$\left\|Q_{h,m}f - Q^{(p)}_{h,m}f\right\|_\infty \le c_{m,p}h^{2p}\left\|f^{(2p)}\right\|_\infty \quad \text{for } f \in V^{2p,\infty}(\mathbb{R})$$

with a constant $c_{m,p}$ that can be described explicitly. A consequence is that for $f \in V^{m,\infty}(\mathbb{R})$ with uniformly continuous $f^{(m)}$ and $2p > m$, it holds $\left\|Q_{h,m}f - Q^{(p)}_{h,m}f\right\|_\infty h^{-m} \to 0$ as $h \to 0$, i.e., the quasi-interpolant $Q^{(p)}_{h,m}f$ is asymptotically of the same accuracy as the interpolant $Q_{h,m}f$. It is reasonable to take the smallest $p \in \mathbb{N}$ for which $2p > m$, denote it by m_1,

$$m_1 = \left\{\begin{array}{l} \dfrac{m}{2} + 1,\ m \text{ even} \\ \dfrac{m+1}{2},\ m \text{ odd} \end{array}\right\} = \left\lfloor\frac{m}{2}\right\rfloor + 1 = m - m_0, \quad Q'_{h,m} = Q^{(m_1)}_{h,m},$$

$$\alpha'_{k,m} := \alpha^{(m_1)}_{k,m}, \quad |k| < m_1. \tag{2.6}$$

Note that $(Q'_{h,m}f)(x)$ is well defined for $x \in [ih,(i+1)h]$ with an $i \in \mathbb{Z}$ if f is given on $[(i-m+1)h,(i+m)h] \cap \mathbb{Z}_{h,m}$.

We assumed that $m \geq 3$. For $m = 1$ and $m = 2$, we may put $Q'_{h,m} = Q_{h,m}$. The following results are proved in [6].

Lemma 3. *For* $i \in \mathbb{Z}$, $f \in C^m[(i-m+1)h,\ (i+m)h]$, *it holds*

$$\max_{ih \leq x \leq (i+1)h} \left|f(x) - \left(Q'_{h,m}f\right)(x)\right| \\ \leq \left(\Phi_{m+1}\pi^{-m} + q_m c'_m\right) h^m \max_{(i-m+1)h \leq x \leq (i+m)h} \left|f^{(m)}(x)\right|. \quad (2.7)$$

For $f \in C^m[-\delta, 1+\delta]$, $\delta > 0$, *it holds*

$$\limsup_{h \to 0} h^{-m} \max_{0 \leq x \leq 1} \left|f(x) - \left(Q'_{h,m}f\right)(x)\right| \leq \Phi_{m+1}\pi^{-m} \max_{0 \leq x \leq 1} \left|f^{(m)}(x)\right|.$$

For a relatively compact set $\mathcal{M}$ *in* $C[-\delta, 1+\delta]$, $\delta > 0$, *it holds*

$$\sup_{f \in \mathcal{M}} \max_{0 \leq x \leq 1} \left|f(x) - \left(Q'_{h,m}f\right)(x)\right| \to 0 \text{ as } h \to 0.$$

In [6], formulae for $q_m := \|Q_{h,m}\|_{BC(\mathbb{R}) \to BC(\mathbb{R})}$, $q'_m := \left\|Q'_{h,m}\right\|_{BC(\mathbb{R}) \to BC(\mathbb{R})}$, c'_m from (2.7) and their numerical values are presented.

3. Spline Quasi-Collocation Solvers

3.1. Introducing remarks

Fast (C, C^m) solvers can be designed directly for integral equation (1.1) or for the equivalent equation

$$v = Tv + g, \qquad g = Tf. \quad (3.1)$$

We choose the latter way. The solutions of (1.1) and (3.1) are in the relations

$$v = Tu, \qquad u = v + f. \quad (3.2)$$

Assuming $f \in C^m[0,1]$, $K \in C^{2m}([0,1] \times [0,1])$, we have $u \in C^m[0,1]$, $v \in C^{2m}[0,1]$ for the solutions of (1.1) and (3.1), and the higher smoothness of v simplifies a fast solving of (3.1) compared with (1.1). On the other hand, a new problem arrives, how to approximate $g = Tf$ in a fast and sufficiently precise way. Nevertheless, fast solvers using equation (3.1) occur to be more effective and algorithmically more simple than the solvers using (1.1) directly.

For $n \in \mathbb{N}$, $h = 1/n$, the spline quasicollocation method for equation (3.1) can be defined as the solving the equation

$$v_n = Q'_{h,2m}Tv_n + Q'_{h,2m}Tf,$$

where $Q'_{h,2m}$ is the quasi-interpolation operator introduced in Section 2.3. Note that $Q'_{h,2m}u$ is well defined in $[0,1]$ if u is defined on $[-2mh, 1+2mh]$. Thus a realization of the quasi-collocation method needs the values of K on $[-2mh, 1+2mh] \times [0,1]$. In the problem setting, the kernel K of the integral operator T is given only on $[0,1]\times[0,1]$. So we have somehow to extend $K(x,y)$ with respect to the argument x. The quasi-collocation method is not discrete. To build a fully discrete method, we need an extension of K with respect to both arguments x and y. Also f must be extended for numerical algorithms. We are interested in extensions that preserve the smoothness of f and K. In Section 3.2 we discuss one possible extension.

3.2. A smoothness preserving extension of functions

Let $u \in C^m[0,1]$. It is possible to extend u to $[-\delta, 1]$, $0 < \delta \le 1/m$, by the well-known reflection formula (see, e.g., [7])

$$u(x) = \sum_{j=0}^{m} c_j u(-jx) \quad \text{for } -\delta \le x < 0, \tag{3.3}$$

where c_j are chosen so that a C^m-smooth joining takes place at $x = 0$. Namely, differentiating (3.3) k times we have

$$u^{(k)}(x) = \sum_{j=0}^{m} (-j)^k c_j u^{(k)}(-jx), \quad -\delta \le x < 0,$$

the C^m-smooth joining at $x = 0$ happens if $\lim_{x\uparrow 0} u^{(k)}(x) = u^{(k)}(0)$, i.e., if

$$\sum_{j=0}^{m} (-j)^k c_j = 1, \quad k = 0, \dots, m.$$

This is a uniquely solvable $(m+1)\times(m+1)$ Vandermonde system to determine $c_0, \dots, c_m$. Moreover, the Cramer rule enables to present the solution of the system in a closed form. Let us recall the formula for a Vandermonde determinant:

$$V(z_0, z_1, \dots z_m) := \begin{vmatrix} 1 & 1 & \dots & 1 \\ z_0 & z_1 & \dots & z_m \\ z_0^2 & z_1^2 & \dots & z_m^2 \\ \vdots & \vdots & \ddots & \vdots \\ z_0^m & z_1^m & \dots & z_m^m \end{vmatrix} = \prod_{0 \le l < k \le m} (z_k - z_l).$$

If we replace the jth column by the column consisting of 1's, we again obtain a Vandermonde determinant. The Cramer rule yields after reductions

$$\begin{aligned} c_j &= \frac{V(-m, \dots, -m+j-1,\, 1,\, -m+j+1, \dots, 0)}{V(-m, -m+1, \dots, 0)} \\ &= (-1)^j \binom{m+1}{j+1}, \quad j = 0, \dots, m. \end{aligned}$$

Same c_j suit to extend u onto $[0, 1+\delta]$:

$$u(x) = \sum_{j=0}^{m} c_j u(1 - j(x-1)) \quad \text{for } 1 < x \le 1+\delta. \tag{3.4}$$

As the result we obtain an extended function $u \in C^m[-\delta, 1+\delta]$. It holds

$$\max_{-\delta \le x \le 1+\delta} |u(x)| \le \sum_{j=0}^{m} |c_j| \max_{0 \le x \le 1} |u(x)| = \left(2^{m+1} - 1\right) \max_{0 \le x \le 1} |u(x)|.$$

If u is given only on the grid $\{ih : i = 0, 1, \dots, n\}$, $h = 1/n$, $n \ge m^2$, (3.3)–(3.4) still enable to compute $u(ih)$ for $i = -m, \dots, -1$ and for $i = n+1, \dots, n+m$. One must be careful with truncation errors when the values $u(jh)$ for $j = 0, \dots, n$, are computed, since these errors may be magnified $2^{m+1} - 1$ times extending $u(jh)$ by (3.3) for $j < 0$ and by (3.4) for $j > n$.

3.3. Discrete quasi-collocation method

Assume that $f \in C^m\left[-\frac{\delta^2}{m}, 1 + \frac{\delta^2}{m}\right]$, $K \in C^{2m}([-2\delta, 1+2\delta] \times [-2\delta, 1+2\delta])$, $\delta > 0$, and that the knot values of f and K exploited below are given. Put $h = 1/n$ where $n \in \mathbb{N}$, $n \ge m/\delta$ (then $mh^2 \le \delta^2/m$, $2mh \le 2\delta$). Using the spline quasi-interpolation operators (see Section 2.3)

$$Q'_{h^2,m} : C[-mh^2, 1+mh^2] \to C[0,1], \quad Q'_{h,2m} : C[-2mh, 1+2mh] \to C[0,1],$$

introduce the quasi-interpolant approximations of f and K: for $0 \le x \le 1$,

$$\begin{aligned} f_n(x) &:= \left(Q'_{h^2,m} f\right)(x) \\ &= \sum_{l=-m+1}^{n^2-1} \Big(\sum_{|q|<m_1} \alpha'_{q,m} f\Big(\Big(l - q + \frac{m}{2}\Big) h^2\Big) \Big) B_m\left(n^2 x - l\right); \end{aligned} \tag{3.5}$$

for $-2mh \le x \le 1+2mh$, $0 \le y \le 1$,

$$\begin{aligned} K_n(x,y) &:= Q'_{h,2m,y} K(x,y) \\ &= \sum_{k=-2m+1}^{n-1} \Big(\sum_{|q| \le m} \alpha'_{q,2m} K(x, (k - q + m)h)\Big) B_{2m}(ny - k); \end{aligned} \tag{3.6}$$

recall (2.6) for the definition of m_1; the subindex y in $Q'_{h,2m,y}K(x,y)$ indicates that $Q'_{h,2m}$ is applied to $K(x,y)$ as to a function of y treating x as a parameter. According to Lemma 3, we have the estimates

$$\max_{0 \le x \le 1} |f(x) - f_n(x)| \le \Phi_{m+1} \pi^{-m} n^{-2m} \max_{0 \le x \le 1} \left|f^{(m)}(x)\right| + n^{-2m} \varepsilon_{n,m,f}, \tag{3.7}$$

$$\begin{aligned} &\max_{\substack{-2mh \le x \le 1+2mh \\ 0 \le y \le 1}} |K(x,y) - K_n(x,y)| \\ &\quad \le \Phi_{2m+1} \pi^{-2m} n^{-2m} \max_{0 \le x,y \le 1} \left|\partial_y^{2m} K(x,y)\right| + n^{-2m} \varepsilon_{n,m,K}, \end{aligned} \tag{3.8}$$

where $\varepsilon_{n,m,f},\ \varepsilon_{n,m,K} \to 0$ as $n \to \infty$. Introduce the integral operator

$$T_n : C[0,1] \to C[-2mh, 1+2mh], \quad (T_n v)(x) = \int_0^1 K_n(x,y)v(y)\,dy.$$

Due to (3.8),

$$\|T - T_n\|_{C[0,1]\to C[-2mh,1+2mh]} \le \Phi_{2m+1}\pi^{-2m}n^{-2m} \max_{0\le x,y\le 1} \left|\partial_y^{2m} K(x,y)\right| + n^{-2m}\varepsilon_{n,m,K}. \tag{3.9}$$

Instead of the pure quasi-collocation method $v_n = Q'_{h,2m}Tv_n + Q'_{h,2m}Tf$, we introduce its fully discrete modification

$$v_n = Q'_{h,2m}T_n v_n + Q'_{h,2m}T_n f_n. \tag{3.10}$$

Theorem 1. *Let $f \in C^m\left[-\frac{\delta^2}{m}, 1+\frac{\delta^2}{m}\right]$, $K \in C^{2m}([-2\delta, 1+2\delta]\times[-2\delta, 1+2\delta])$, $\delta > 0$, and let $\mathcal{N}(I-T) = \{0\}$. Then equation (3.1) has in $C[0,1]$ a unique solution $v \in C^{2m}[0,1]$, for all sufficiently large n equation (3.10) has in $C[0,1]$ a unique solution v_n, and it holds*

$$\max_{0\le x\le 1}|v(x) - v_n(x)| \le \kappa_n n^{-2m}\Big(\Big(\Phi_{m+1}\pi^{-m}\left\|f^{(m)}\right\|_\infty + \varepsilon_{n,m,f}\Big)\|K\|_\infty + \Big(\Phi_{2m+1}\pi^{-2m}\Big(\kappa\left\|\partial_x^{2m}K\right\|_\infty + (\kappa q'_{2m}+1)\left\|\partial_y^{2m}K\right\|_\infty\Big) + \varepsilon_{n,m,K}\Big)\|f\|_\infty\Big), \tag{3.11}$$

where $\varepsilon_{n,m,K} \to 0$, $\varepsilon_{n,m,f} \to 0$ as $n \to \infty$,

$$\kappa_n := \left\|\left(I - Q'_{h,2m}T_n\right)^{-1}\right\|_{C[0,1]\to C[0,1]} \to \left\|(I-T)^{-1}\right\|_{C[0,1]\to C[0,1]} =: \kappa,$$

Φ_m is the Favard constant, $q'_{2m} = \left\|Q'_{h,m}\right\|_{BC(\mathbb{R})\to BC(\mathbb{R})}$ (see Section 2.3),

$$\|f\|_\infty := \max_{0\le x\le 1}|f(x)|, \qquad \|K\|_\infty := \max_{0\le x,y\le 1}|K(x,y)|$$

and similar sense have the norms $\left\|f^{(m)}\right\|_\infty$, $\left\|\partial_x^{2m}K\right\|_\infty$, $\left\|\partial_y^{2m}K\right\|_\infty$.

Proof. Consider the difference of operators in equations (3.1) and (3.10):

$$T - Q'_{h,2m}T_n = \left(I - Q'_{h,2m}\right)T + Q'_{h,2m}(T - T_n).$$

The operator $T : C[0,1] \to C^{2m}[-\delta, 1+\delta]$ is compact. The image $M \subset C^{2m}[-\delta, 1+\delta]$ of the unit ball of $C[0,1]$ by T is relatively compact in $C^{2m}[-\delta, 1+\delta]$, and Lemma 3 yields

$$\left\|\left(I - Q'_{h,2m}\right)T\right\|_{C[0,1]\to C[0,1]} = \sup_{\|u\|_\infty\le 1}\max_{0\le x\le 1}\left|(Tu)(x) - \left(Q'_{h,2m}Tu\right)(x)\right| \le \Phi_{2m+1}\pi^{-2m}n^{-2m}\max_{0\le x,y\le 1}\left|\partial_x^{2m}K(x,y)\right| + n^{-2m}\varepsilon_{n,m,M},$$

where $\varepsilon_{n,m,M} \to 0$ as $n \to \infty$. The norm $\left\| Q'_{h,2m}(T - T_n) \right\|_{C[0,1]\to C[0,1]}$ can be estimated with the help of (3.9), and we obtain

$$\left\| T - Q'_{h,2m}T_n \right\|_{C[0,1]\to C[0,1]} \leq \Phi_{2m+1}\pi^{-2m}n^{-2m} \tag{3.12}$$
$$\times \left(\max_{0\leq x,y\leq 1} \left| \partial_x^{2m} K(x,y) \right| + q'_{2m} \max_{0\leq x,y\leq 1} \left| \partial_y^{2m} K(x,y) \right| \right) + n^{-2m}\varepsilon'_{n,m,K},$$

where $\varepsilon'_{n,m,K} \to 0$ as $n \to \infty$. Further, for the difference of the free terms of equations (3.1) and (3.10) we have

$$Tf - Q'_{h,2m}T_n f_n = T(f - f_n) + \left(T - Q'_{h,2m}T_n \right) f_n$$

that together with (3.7) and (3.12) yields

$$\left\| Tf - Q'_{h,2m}T_n f_n \right\|_\infty \leq \|K\|_\infty \left(\Phi_{m+1}\pi^{-m} \left\| f^{(m)} \right\|_\infty + \varepsilon_{n,m,f} \right) n^{-2m}$$
$$+ \left(\frac{\Phi_{2m+1}}{\pi^{2m}} \left(\left\| \partial_x^{2m} K \right\|_\infty + q'_{2m} \left\| \partial_y^{2m} K \right\|_\infty \right) + \varepsilon_{n,m,K} \right) \max_{0\leq x\leq 1} \frac{|f_n(x)|}{n^{2m}}. \tag{3.13}$$

Here $\max_{0\leq x\leq 1} |f_n(x)| \leq \max_{0\leq x\leq 1} |f(x)| + O\left(n^{-2m}\right)$ as $n \to \infty$. The claims of the theorem follow from estimates (3.12) and (3.13). ■

Remark 1. According to (2.7), for the quantity or $\varepsilon_{n,m,f}$ in (3.13) and hence in (3.11), we have the estimate

$$\varepsilon_{n,m,f} \leq q_m c'_m \max_{-(m-1)h^2 \leq x \leq 1+(m-1)h^2} |f(x)|.$$

3.4. Two grid iterations

Equation (3.10) can be solved by the two grid iteration method (cf. [1, 3, 19]): take a $\nu \in \mathbb{N}$ such that $\nu \sim n^\theta$, $\frac{1}{2} < \theta < 1$, rewrite equation (3.10) in the form

$$v_n = Q'_{1/\nu,2m}T_\nu v_n + \left(Q'_{h,2m}T_n - Q'_{1/\nu,2m}T_\nu \right) v_n + Q'_{h,2m}T_n f_n, \tag{3.14}$$

and starting from $v_n^0 = 0$, compute the iterations for $k = 1, 2, \cdots$:

$$v_n^k = Q'_{1/\nu,2m}T_\nu v_n^k + \left(Q'_{h,2m}T_n - Q'_{1/\nu,2m}T_\nu \right) v_n^{k-1} + Q'_{h,2m}T_n f_n. \tag{3.15}$$

The operator $I - Q'_{1/\nu,2m}T_\nu$ is invertible for sufficiently large n, cf. Theorem 1. It occurs that two iterations (3.15) is enough for our accuracy needs.

Theorem 2. *Let the conditions of Theorem* 1 *be fulfilled, and let* $\nu \sim n^\theta$, $\frac{1}{2} < \theta < 1$. *Then*

$$\max_{0\leq x\leq 1} \left| v_n(x) - v_n^2(x) \right| \leq d_m n^{-4\theta m} \|f\|_{\infty,[0,1]} = o\left(n^{-2m}\right) \|f\|_{\infty,[0,1]}, \tag{3.16}$$

where v_n *is the solution of equation* (3.10) *and* v_n^2 *is the second iteration defined by* (3.15). *The constant* d_m *can be reproduced explicitly following the proof.*

Proof. Denoting $S_{n,\nu} := \left(I - Q'_{1/\nu,2m}T_\nu\right)^{-1}\left(Q'_{h,2m}T_n - Q'_{1/\nu,2m}T_\nu\right)$, we rewrite (3.14) and (3.15) in the form

$$v_n = S_{n,\nu}v_n + \left(I - Q'_{1/\nu,2m}T_\nu\right)^{-1} Q'_{h,2m}T_n f_n, \qquad v_n^0 = 0,$$
$$v_n^k = S_{n,\nu}v_n^{k-1} + \left(I - Q'_{1/\nu,2m}T_\nu\right)^{-1} Q'_{h,2m}T_n f_n, \quad k = 1, 2.$$

Subtracting we obtain

$$v_n - v_n^2 = S_{n,\nu}\left(v_n - v_n^1\right) = S_{n,\nu}^2\left(v_n - v_n^0\right) = S_{n,\nu}^2 v_n,$$
$$\|v_n - v_n^2\|_{\infty,[0,1]} \le \|S_{n,\nu}\|^2_{C[0,1]\to C[0,1]}\|v_n\|_{\infty,[0,1]} \le \kappa_\nu^2$$
$$\times \left\|Q'_{h,2m}T_n - Q'_{1/\nu,2m}T_\nu\right\|^2_{C[0,1]\to C[0,1]} \kappa_n \left\|Q'_{h,2m}T_n\right\|_{C[0,1]\to C[0,1]}\|f_n\|_{\infty,[0,1]}.$$

Here (see Theorem 1 and (3.12))

$$\left\|Q'_{h,2m}T_n\right\|_{C[0,1]\to C[0,1]} \to \|T\|_{C[0,1]\to C[0,1]},$$
$$\|f_n\|_{\infty,[0,1]} \to \|f\|_{\infty,[0,1]}, \quad \kappa_n, \kappa_\nu \to \kappa \text{ as } n \to \infty,$$
$$\left\|Q'_{h,2m}T_n - Q'_{1/\nu,2m}T_\nu\right\|_{C[0,1]\to C[0,1]}$$
$$\le \left\|Q'_{h,2m}T_n - T\right\|_{C[0,1]\to C[0,1]} + \left\|T - Q'_{1/\nu,2m}T_\nu\right\|_{C[0,1]\to C[0,1]}$$
$$\le 2\varPhi_{2m+1}\pi^{-2m}\nu^{-2m}\left(\left\|\partial_x^{2m}K\right\|_\infty + q'_{2m}\left\|\partial_y^{2m}K\right\|_\infty\right) + 2\nu^{-2m}\varepsilon_{\nu,m,K},$$

and we arrive at (3.16). ■

Due to (3.2) and Theorems 1, 2, $u_n = v_n^2 + f$ is a suitable approximation to the solution u of equation (1.1) in the practice but not for the construction of a fast (C, C^m) solver since too rich information about f is used. Therefore we set

$$u_n = v_n^2 + f_n, \tag{3.17}$$

where f_n is defined in (3.5) and satisfies (3.7). For u_n defined by (3.17) Theorems 1 and 2 still imply the estimate

$$\|u - u_n\|_\infty \le \frac{c_{m,n}}{n^{2m}}\left(\|f\|_\infty + \left\|f^{(m)}\right\|_\infty\right), \quad c_m := \limsup_{n\to\infty} c_{m,n} < \infty. \tag{3.18}$$

In Section 5 we present (3.18) in a more detailed form (5.2), (5.3) with an explicit formula for c_m.

Treating (3.18) as the estimate (1.2) with $n_\star = n^2$, our main task now will be to show, how the parameters of v_n^2 can be computed in $b_m n_\star = b_m n^2$ flops (see the definition of the fast (C, C^m) solver, Section 1). The $n^2 + 2m - 1$ parameters of f_n – the knot values $f\left(\left(l + \frac{m}{2}\right)h^2\right)$, $l = -m+1, \ldots, n^2+m-1$, involved in (3.5) – are assumed to be given, similarly as suitable $\sim n^2$ knot values of $K(x, y)$. To be formally precise, we should rewrite (3.18) with respect to $n_\star = n^2 + 2m - 1$ and take the care that also K is evaluated not more than in $n^2 + 2m - 1$ points. Fortunately, moderate shifts in $n_\star$ do not influence on asymptotics as $n \to \infty$, in particular, on c_m in (3.18), so we do not need to follow the formalism so precisely.

4. Implementation of the two Grid Iterations

4.1. Reformulation of the iteration formula and further prospects

The two grid iterations (3.15) occur to be more flexible and convenient if we rewrite them with respect to $w_n^k := T_n v_n^k$ and $w_\nu^k := T_\nu v_n^k$, $k = 1, 2, \dots$ Applying T_ν to (3.15), then T_n to (3.15), the iteration formula (3.15) yields for $k = 1, 2, \dots$

$$w_\nu^k = T_\nu Q'_{1/\nu,2m} w_\nu^k + \left(T_\nu Q'_{h,2m} w_n^{k-1} - T_\nu Q'_{1/\nu,2m} w_\nu^{k-1}\right) + T_\nu Q'_{h,2m} T_n f_n,$$
$$w_n^k = T_n Q'_{1/\nu,2m} w_\nu^k + \left(T_n Q'_{h,2m} w_n^{k-1} - T_n Q'_{1/\nu,2m} w_\nu^{k-1}\right) + T_n Q'_{h,2m} T_n f_n.$$

With $w_n^0 = 0$, $w_\nu^0 = 0$, the first iteration approximations can be represented as

$$w_\nu^1 = T_\nu Q'_{1/\nu,2m} w_\nu^1 + T_\nu Q'_{h,2m} T_n f_n, \tag{4.1}$$

$$w_n^1 = T_n Q'_{1/\nu,2m} w_\nu^1 + T_n Q'_{h,2m} T_n f_n, \tag{4.2}$$

$$w_\nu^2 = T_\nu Q'_{1/\nu,2m} w_\nu^2 - w_\nu^1 + T_\nu Q'_{h,2m} \left(w_n^1 + 2T_n f_n\right), \tag{4.3}$$

after that (3.15) yields

$$v_n^2 = Q'_{1/\nu,2m} \left(w_\nu^2 - w_\nu^1\right) + Q'_{h,2m} \left(w_n^1 + T_n f_n\right). \tag{4.4}$$

Note that (4.1) and (4.3) are equations on the coarse level ν w.r.t. w_ν^1 and w_ν^2, respectively, whereas in (4.2), only certain linear operators are applied to known functions. The computations can be performed using only the grid values of functions: having $(T_n f_n)(ih)$, $i = -2m + 1, \dots, n + 2m - 1$, at our disposal, we can compute $w_\nu^1(i/\nu)$, $i = -2m + 1, \dots, \nu + 2m - 1$, by solving (4.1), after that $w_n^1(ih)$, $i = -2m + 1, \dots, n + 2m - 1$, by (4.2) and $w_\nu^2(i/\nu)$, $i = -2m + 1, \dots, \nu + 2m - 1$, by solving (4.3); with these grid values, (4.4) recovers v_n^2 for $0 \le x \le 1$. The computation of $(T_n f_n)(ih)$, $i = -2m + 1, \dots, n + 2m - 1$, occurs to be the most labour consuming part of the solver and we postpone the algorithmic details to Section 4.5. In Sections 4.2–4.4 we comment on the computation of the grid values of $T_n Q'_{h,2m} w$, $T_\nu Q'_{1/\nu,2m} w$, $T_n Q'_{1/\nu,2m} w$, $T_\nu Q'_{h,2m} w$ through the grid values of a given function w. This enables an implementation of (4.1)–(4.3) without matrix representations of the operators in (4.1)–(4.3) so far as we solve equations (4.1) and (4.3) by a suitable iteration method, for instance by GMRES. About algorithmic aspects of GMRES, see [8, 9, 10, 14]; the complexity of GMRES for discretisations of integral equations has been analyzed in [10, 18]. Applying GMRES to equation (3.10) directly, accuracy $O\left(n^{-r}\right)$ with any given r is achieved in $o(\log n)$ GMRES iterations that results to a complexity $o\left(n^2 \log n\right)$ flops of the method due to the dimension $\sim n$ of problem (3.10). This is quite acceptable for practical purposes but slightly too expensive for the construction of a fast (C, C^m) solver where only $O\left(n^2\right)$ flops are allowed (see Section 3). When we solve equation (3.10) via two grid iterations (4.1)–(4.4) and apply GMRES to equations (4.1) and (4.3), we achieve an accuracy $O\left(n^{-r}\right)$ still in $o(\log n)$ GMRES iterations, and due to the dimension $\nu \ll n$ of problems (4.1), (4.3), the complexity of method (4.1)–(4.4) reduces to desired $O\left(n^2\right)$ flops.

When solving (4.1) and (4.3) by the Gauss elimination or by other direct methods of the complexity $O\left(\nu^3\right)$, we have to strengthen the condition on ν so that $\nu^3 \ll n^2$ – we choose the coarse level so that $\nu \asymp n^\theta$, $\frac{1}{2} < \theta < \frac{2}{3}$; moreover, we need the matrix representation of the coarse level equations (4.1) and (4.3). The matrix representation of the fine level equation (3.10) is also useful in the practice allowing to solve it directly (without two grid iterations) by standard codes of the Gauss elimination, GMRES, conjugate gradients or other methods of linear algebra. The matrix representations of equation (3.10) is treated in Section 4.6; for (4.1) and (4.3) the matrix representation is similar.

4.2. Application of the operator $T_n Q'_{h,2m}$

For $w \in C[-2mh, 1+2mh]$, we have

$$\left(Q'_{h,2m} w\right)(y) = \sum_{j=-2m+1}^{n-1} \sum_{|p|\le m} a'_{p,2m} w((j-p+m)h) B_{2m}(ny-j), \quad 0 \le y \le 1,$$

and by (3.6), for $-2mh \le x \le 1+2mh$,

$$(T_n Q'_{h,2m} w)(x) = \int_0^1 K_n(x,y)(Q'_{h,2m} w)(y)\,dy = \sum_{k=-2m+1}^{n-1} \sum_{|q|\le m} \alpha'_{q,2m}$$
$$\times K(x,(k-q+m)h) \sum_{j=-2m+1}^{n-1} \beta^{n,n}_{k,j} \sum_{|p|\le m} \alpha'_{p,2m} w((j-p+m)h), \qquad (4.5)$$

where we denoted

$$\beta^{n,n}_{k,j} = \int_0^1 B_{2m}(ny-k) B_{2m}(ny-j)\,dy, \quad k,j = -2m+1,\dots,n-1. \quad (4.6)$$

From (2.2) we observe that $\beta^{n,n}_{k,j} = 0$ for $|k-j| \ge 2m$, thus the summing up over p and j in (4.5) is cheap, it is sufficient $\sim (2m+1)n$ flops for the computation of

$$\xi_j := \sum_{|p|\le m} \alpha'_{p,2m} w((j-p+m)h), \quad j = -2m+1,\dots,n-1, \qquad (4.7)$$

after that $\sim 4mn$ flops for the computation of

$$\eta_k := \sum_{j=-2m+1}^{n-1} \beta_{k,j}^{n,n} \xi_j, \quad k = -2m+1, \ldots, n-1. \tag{4.8}$$

Now (4.5) takes the form

$$\begin{aligned} \left(T_n Q'_{h,2m} w\right)(x) &= \sum_{|q| \le m} \sum_{k=-2m+1}^{n-1} \alpha'_{q,2m} K(x, (k-q+m)h) \eta_k \\ &= \sum_{|q| \le m} \sum_{k'=-m-q+1}^{n+m-q-1} \alpha'_{q,2m} K(x, k'h) \eta_{k'+q-m} \end{aligned}$$

and after the change of summation ordering

$$\left(T_n Q'_{h,2m} w\right)(x) = \sum_{k'=-2m+1}^{n+2m-1} K\left(x, k'h\right) \sum_{q=\max\{-m,-k'-m+1\}}^{\min\{m,-k'+n+m-1\}} \alpha'_{q,2m} \eta_{k'+q-m}.$$

Thus we can continue (4.7), (4.8) by

$$\zeta_{k'} = \sum_{q=\max\{-m,-k'-m+1\}}^{\min\{m,-k'+n+m-1\}} \alpha'_{q,2m} \eta_{k'+q-m}, \quad k' = -2m+1, \ldots, n+2m-1, \tag{4.9}$$

$$\left(T_n Q'_{h,2m} w\right)(ih) = \sum_{k'=-2m+1}^{n+2m-1} K\left(ih, k'h\right) \zeta_{k'}, \quad i = -2m+1, \ldots, n+2m-1, \tag{4.10}$$

that cost, respectively, $\sim (2m+1)n$ and $\sim n^2$ flops.

Let us comment on the computation of nonzero ones of the quantities $\beta_{k,j}^{n,n}$, $k, j = -2m+1, \ldots, n-1$, defined in (4.6). Everybody of them is a sum of not more than $2m$ copies of the following $m(2m+1)$ "brick" integrals

$$\int_{ih}^{(i+1)h} B_{2m}(ny) B_{2m}(ny - l)\, dy, \quad l = 0, \ldots, 2m-1, \quad i = l, \ldots, 2m-1$$

The integrand of a "brick" integral is a polynomial of degree $4m-2$, and a "brick" integral can be evaluated exactly by the $2m$ point Gauss quadrature formula in $2m$ flops provided that we already have evaluated $B_{2m}(ny)$ at the Gauss knots of the subintervals $(ih, (i+1)h)$, $i = 0, \ldots, 2m$ (the evaluations can be done in $2m \cdot 2m \cdot 2m$ flops). Thus all "brick" integrals are available in $\sim 12m^3$ flops, and also all $\beta_{k,j}^{n,n}$, $k, j = -2m+1, \ldots, n-1$, are available in $O\left(m^3\right)$ flops independently of n.

The computation of the knot values $\left(T_n Q'_{h,2m} w\right)(ih)$, $i = -2m+1, \ldots, n+2m-1$, from (4.7)–(4.10) costs $\sim (2m+1)n + 2mn + (2m+1)n + n^2 \sim n^2$ flops.

The computation of the knot values $\left(T_\nu Q'_{1/\nu,2m} w\right)(i/\nu s)$, $i = -2m+1, \ldots, \nu+2m-1$, is similar to (4.6)–(4.10) and costs $\sim \nu^2$ flops.

4.3. Application of the operator $T_\nu Q'_{h,2m}$

Only slight modifications occur in formulae (4.6)–(4.10) when we compute $T_\nu Q'_{h,2m} w$. Namely, for $-2m/\nu \le x \le 1 + 2m/\nu$,

$$\left(T_\nu Q'_{h,2m} w\right)(x) = \int_0^1 K_\nu(x,y)(Q'_{h,2m} w)(y)\,dy = \sum_{k=-2m+1}^{\nu-1} \sum_{|q|\le m}$$

$$\times\, \alpha'_{q,2m} K(x,(k-q+m)/\nu) \sum_{j=-2m+1}^{n-1} \beta^{\nu,n}_{k,j} \sum_{|p|\le m} \alpha'_{p,2m} w((j-p+m)h),$$

where

$$\beta^{\nu,n}_{k,j} = \int_0^1 B_{2m}(\nu y - k) B_{2m}(ny - j)\,dy,$$
$$k = -2m+1,\dots,\nu-1, \quad j = -2m+1,\dots,n-1. \tag{4.11}$$

Having computed

$$\xi_j := \sum_{|p|\le m} \alpha'_{p,2m} w((j-p+m)h), \quad j = -2m+1,\dots,n-1, \tag{4.12}$$

$$\eta_k := \sum_{j=-2m+1}^{n-1} \beta^{\nu,n}_{k,j} \xi_j, \quad k = -2m+1,\dots,\nu-1, \tag{4.13}$$

$$\zeta_{k'} = \sum_{q=\max\{-m,-k'-m+1\}}^{\min\{m,-k'+\nu+m-1\}} \alpha'_{q,2m} \eta_{k'+q-m}, \quad k' = -2m+1,\dots,\nu+2m-1, \tag{4.14}$$

we obtain

$$\left(T_\nu Q'_{h,2m} w\right)\left(\frac{i}{\nu}\right) = \sum_{k'=-2m+1}^{\nu+2m-1} K\left(\frac{i}{\nu},\frac{k'}{\nu}\right)\zeta_{k'}, \quad i = -2m+1,\dots,\nu+2m-1. \tag{4.15}$$

Dividing the integrals in (4.11) into elementary "bricks" similarly as in Section 4.2, we can compute $\beta^{\nu,n}_{k,j}$, $k = -2m+1,\dots,\nu-1$, $j = -2m+1,\dots,n-1$, in $O(m^2 n/\nu)$. After that the computation of $\left(T_\nu Q'_{h,2m} w\right)(i/\nu)$, $i = -2m+1,\dots,\nu-1$, by (4.12)–(4.15) costs $\sim (2m+1)n + 2\nu^2 + (2m+1)\nu + \nu^2 \sim 3\nu^2$ flops; recall that $\nu \asymp n^\theta$, $\frac{1}{2} < \theta < 1$.

4.4. Application of the operator $T_n Q'_{1/\nu,2m}$

For the computation of $T_n Q'_{1/\nu,2m} w$ we obtain the following formulae: for $-2mh \le x \le 1 + 2mh$,

$$\left(T_n Q'_{i/\nu,2m} w\right)(x) = \int_0^1 K_n(x,y)\left(Q'_{1/\nu,2m} w\right)(y)\,dy = \sum_{k=-2m+1}^{n-1} \sum_{|q|\le m}$$

$$\times\, a'_{q,2m} K(x,(k-q+m)h) \sum_{j=-2m+1}^{\nu-1} \beta_{k,j}^{n,\nu} \sum_{|p|\le m} \alpha'_{p,2m} w((j-p+m)/\nu),$$

where (cf. (4.11))

$$\beta_{k,j}^{n,\nu} = \int_0^1 B_{2m}(ny-k) B_{2m}(\nu y - j)\, dy = \beta_{j,k}^{\nu,n},$$
$$k=-2m+1,\ldots,n-1, \quad j=-2m+1,\ldots,\nu-1.$$

Having computed

$$\xi_j := \sum_{|p|\le m} \alpha'_{p,2m} w((j-p+m)/\nu), \quad j=-2m+1,\ldots,\nu-1, \tag{4.16}$$

$$\eta_k := \sum_{j=-2m+1}^{\nu-1} \beta_{k,j}^{n,\nu} \xi_j, \quad k=-2m+1,\ldots,n-1, \tag{4.17}$$

$$\zeta_{k'} = \sum_{q=\max\{-m,-k'-m+1\}}^{\min\{m,-k'+n+m-1\}} \alpha'_{q,2m} \eta_{k'+q-m}, \quad k'=-2m+1,\ldots,n+2m-1, \tag{4.18}$$

we obtain

$$\left(T_n Q'_{1/\nu,2m} w\right)(ih) = \sum_{k'=-2m+1}^{n+2m-1} K\left(ih, k'h\right) \zeta_{k'}, \quad i=-2m+1,\ldots,n+2m-1. \tag{4.19}$$

The computation of $\left(T_n Q'_{1/\nu,2m} w\right)(ih)$, $i=-2m+1,\ldots,n-1$, by formulae (4.16)–(4.19) costs $\sim (2m+1)\nu + 4mn + (2m+1)n + n^2 \sim n^2$ flops.

4.5. The computation of $T_n f_n$

By (3.5) and (3.6),

$$\left(T_n f_n\right)(x) = \int_0^1 K_n(x,y) \sum_{l=-m+1}^{n^2-1} \Big(\sum_{|p|\le m_1-1} \alpha'_{p,m} f\left((l-p+\frac{m}{2})h^2\right)\Big)$$
$$\times\, B_m(n^2 y - l)\, dy = \sum_{k=-2m+1}^{n-1} \sum_{|q|\le m} \alpha'_{q,2m} K(x,(k-q+m)h)$$
$$\times \sum_{l=-m+1}^{n^2-1} \gamma_{k,l} \sum_{|p|\le m_1-1} \alpha'_{p,m} f\left(\left(l-p+\frac{m}{2}\right)h^2\right) \tag{4.20}$$

where we denoted for $k=-2m+1,\ldots,n-1$, $l=-m+1,\ldots,n^2-1$

$$\gamma_{k,l} = \gamma_{k,l,m,n} = \int_0^1 B_{2m}(ny-k) B_m(n^2 y - l)\, dy. \tag{4.21}$$

Let us comment on the computation of these integrals.

First of all, $\gamma_{k,l} = 0$ if $(0,1) \cap \big(knh^2, (k+2m)nh^2\big) \cap \big(lh^2, (l+m)h^2\big) = \emptyset$, thus for fixed k, $-2m+1 \le k \le n-1$, the summation over l in (4.20) is actually restricted to the index set

$$\mathcal{I}_k = \mathcal{I}_{k,m,n} := \{l \in \mathbb{Z}:\ l_k^\star \le l \le l_k^{\star\star}\}, \quad \mathrm{card}\,(\mathcal{I}_k) < m(2n+1),$$
$$l_k^\star := \max\{-m+1, -m+kn+1\}, \qquad l_k^{\star\star} := \min\big\{n^2-1, 2mn+kn-1\big\}.$$

There is a kind of periodicity in $\gamma_{k,l}$ for "central" k, $0 \le k \le n-2m-1$: the change of variables $y \mapsto z$, $ny = nz - 1$ in (4.21) yields $\gamma_{k,l} = \gamma_{k+1,l+n}$.

Further, each of integrals $\gamma_{k,l}$, $k = -2m+1, \ldots, n-1$, $l \in \mathcal{I}_k$, is a sum of not more than $2mn + 2m - 2$ copies of the $\sim 2m^2n$ "brick" integrals

$$\int_{ih^2}^{(i+1)h^2} B_{2m}(ny) B_m\big(n^2y - j\big)\, dy, \ \ j = -m+1, \ldots, 2mn-1, \ \ j \le i \le j+m-1,$$

with a polynomial integrand of degree $3m-2$; due to symmetry, actually the number of bricks can be reduced to $\sim m^2n$. A "brick" integral can be exactly computed by a $\sim \frac{3}{2}m$ point Gauss quadrature formula at the cost of $\sim \frac{3}{2}m$ flops provided that we already have evaluated $B_{2m}(ny)$ at the Gauss knots of the subintervals $\big(ih^2, (i+1)h^2\big)$, $i = 0, \ldots, 2mn-1$; due to symmetry, these evaluations can be done in $\sim mn \cdot \frac{3}{2}m \cdot 3m$ flops (number of intervals $\times$ number of Gauss knots in one interval $\times$ evaluation cost of polynomial integrand at one Gauss knot). Thus all "brick" integrals can be computed in $\sim \frac{3}{2}m^3n + \frac{9}{2}m^3n = 6m^3n$ flops, and the same asymptotics holds for the price of all $\gamma_{k,l}$, $k = -2m+1, \ldots, n-1$, $l \in \mathcal{I}_k$.

A direct computation of the intermediate quantities (see (4.20))

$$\varphi_k := \sum_{l \in \mathcal{I}_k} \gamma_{k,l} \sum_{|p| \le m_1 - 1} \alpha'_{p,m} f\left(\left(l - p + \frac{m}{2}\right)h^2\right), \quad k = -2m+1, \ldots, n-1,$$

costs $\sim 3mn^2$ flops ($\sim mn^2$ for the summations over p and $\sim 2mn^2$ for the summations over l). By changing the ordering of summations the computation cost can be reduced to $\sim 2mn^2$ flops. Indeed, with $l' = l - p$ and the change of summation ordering we obtain for $k = -2m+1, \ldots, n-1$:

$$\begin{aligned}
\varphi_k &= \sum_{l=l_k^\star}^{l_k^{\star\star}} \gamma_{k,l} \sum_{p=-m_1+1}^{m_1-1} \alpha'_{p,m} f\left(\left(l - p + \frac{m}{2}\right)h^2\right) \\
&= \sum_{l=l_k^\star}^{l_k^{\star\star}} \gamma_{k,l} \sum_{l'=l-m_1+1}^{l+m_1-1} \alpha'_{l-l',m} f\left(\left(l' + \frac{m}{2}\right)h^2\right) \\
&= \sum_{l'=l_k^\star - m_1 + 1}^{l_k^{\star\star} + m_1 - 1} \left(\sum_{l=\max\{l_k^\star, l'-m_! + 1\}}^{\min\{l_k^{\star\star}, l'+m_1-1\}} \gamma_{k,l}\alpha'_{l-l',m} \right) f\left(\left(l' + \frac{m}{2}\right)h^2\right),
\end{aligned}$$

For a fixed k, $-2m+1 \le k \le n-1$, the computation of the occurring here coefficients

$$\bar{\gamma}_{k,l'} := \sum_{l=\max\{l_k^\star, l'-m_1+1\}}^{\min\{l_k^{\star\star}, l'+m_1-1\}} \gamma_{k,l}\alpha'_{l-l',m}, \quad l' \in \mathcal{I}'_k := \{l_k^\star - m_1 + 1, \ldots, l_k^{\star\star} + m_1 + 1\},$$

costs $\sim(m+1)2mn$ flops (the number of terms in the sum is $\leq m+1$, and $\operatorname{card}(\mathcal{I}'_k) \leq m(2n+2)$). Due to the periodicity property $\gamma_{k,l} = \gamma_{k+1,l+n}$ for $0 \leq k \leq n-2m-1$, $l \in \mathcal{I}_k$, it holds $\bar{\gamma}_{k,l'} = \bar{\gamma}_{k+1,l'+n}$ for $0 \leq k \leq n-2m-1$. So it is sufficient to compute the coefficients $\bar{\gamma}_{k,l'}$ for $4m+1$ different values of k, and all $\bar{\gamma}_{k,l'}$, $k = -2m+1, \ldots, n-1$, $l' \in \mathcal{I}'_k$, are available in $\sim 8m^3 n$ flops (plus $\sim 6m^3 n$ flops for the computation of $\gamma_{k,l}$, $k = -2m+1, \ldots, n-1$, $l \in \mathcal{I}_k$, established above).

Having $\bar{\gamma}_{k,l'}$ in hand, the computation of

$$\varphi_k = \sum_{l'=l_k^\star - m_1+1}^{l_k^{\star\star}+m_1-1} \bar{\gamma}_{k,l'} f\left(\left(l' + \frac{m}{2}\right)h^2\right), \quad k = -2m+1, \ldots, n-1, \tag{4.22}$$

costs $\sim 2mn \cdot n$ flops (the number of terms in the sum times the number of k). Thus the full cost of φ_k, $k = -2m+1, \ldots, n-1$, is $\sim 2mn^2 + 14m^3 n \sim 2mn^2$ flops as asserted. After that for $i = -2m+1, \ldots, n+2m-1$

$$(T_n f_n)(ih) = \sum_{k=-2m+1}^{n-1} \sum_{|q| \leq m} \alpha'_{q,2m} K(ih, (k-q-m)h)\varphi_k,$$

can be computed in $\sim n^2$ flops similarly as in Section 4.2, replacing η_k in (4.9) by φ_k:

$$\psi_{k'} = \sum_{q=\max\{-m, -k'-m+1\}}^{\min\{m, -k'+n+m-1\}} \alpha'_{q,2m}\varphi_{k'+q-m}, \quad k' = -2m+1, \ldots, n+2m-1, \tag{4.23}$$

$$(T_n f_n)(ih) = \sum_{k'=-2m+1}^{n+2m-1} K(ih, k'h)\psi_{k'}, \quad i = -2m+1, \ldots, n+2m-1, \tag{4.24}$$

The summary is that the knot values $(T_n f_n)(ih)$, $i = -2m+1, \ldots, n+2m-1$, are available by (4.22)–(4.24) in $\sim(2m+1)n^2$ flops.

4.6. The matrix representation of the systems

Consider the counterpart of equation (3.10)

$$w_n = T_n Q'_{h,2m} w_n + g \tag{4.25}$$

with an arbitrary force term g. The function $T_n Q'_{h,2m} w_n$ is uniquely determined by the knot values $w_n(ih)$, $i = -2m+1, \ldots, 1+2m-1$. Collocating (4.25) at points ih we obtain a certain system of linear algebraic equations

$$w_n(ih) = \sum_{j=-2m+1}^{n+2m-1} \tau_{i,j} w_n(jh) + g(ih), \quad i = -2m+1, \ldots, n+2m-1. \tag{4.26}$$

The matrix entries $\tau_{i,j}$ can be determined knowing from Section 4.2 the response of the operator $T_n Q'_{h,2m} w$ to a given function w. Namely, let functions $e_j \in C[-2mh, 1+2mh]$, $j = -2m+1, \dots, n+2m-1$, be such that

$$e_j(ih) = \delta_{i,j}, \quad i,j = -2m+1, \dots, n+2m-1, \tag{4.27}$$

then

$$\tau_{i,j} = \left(T_n Q'_{h,2m} e_j\right)(ih), \quad i,j = -2m+1, \dots, n+2m-1. \tag{4.28}$$

Revisiting Section 4.2 and using (4.27) we observe that the matrix of system (4.26) (the knot values (4.28)) is available in $\sim(6m+3)n^2$ flops.

Respectively, the computation of the matrix for equations (4.1) and (4.3) costs $\sim(6m+3)\nu^2 = o\left(n^2\right)$ flops.

5. Summary: the Spline Quasicollocation Fast (C, C^m) Solver

The integrals $\beta^{n,n}_{k,j}$, $\beta^{\nu,n}_{k,j}$, $\beta^{n,\nu}_{k,j}$, $\beta^{n,n}_{k,j}$, $\gamma_{k,l}$, $\bar{\gamma}_{k,l'}$ (see Section 4) are available at the cost of $O\left(m^3 n\right)$ flops. Computing v_n^2 via iterations (4.1)–(4.4), we apply either GMRES or the Gauss elimination for the solving of equations (4.1) and (4.3), and so we actually obtain two different fast (C, C^m) solvers. In both cases we define the approximate solution u_n of (1.1) via the formula (see (3.5), (3.17) and (4.4))

$$u_n = v_n^2 + f_n = Q'_{1/\nu,2m}\left(w_\nu^2 - w_\nu^1\right) + Q'_{h,2m}\left(w_n^1 + T_n f_n\right) + Q'_{h^2,m} f. \tag{5.1}$$

The approximation u_n is well defined on $[0,1]$ provided that the $\sim n^2$ knot values $f\left(\left(l + \frac{m}{2}\right)h^2\right)$, $l = -m+1, \dots, n^2+m-1$, are given and and the parameters $(T_n f_n)(ih)$, $w_n^1(ih)$, $i = -2m+1, \dots, n+2m-1$, and $w_\nu^k(i/\nu)$, $i = -2m+1, \dots, \nu+2m-1$, $k = 1,2$, have been computed; in the computations, the $\sim n^2$ knot values $K(ih, jh)$, $i,j = -2m+1, \dots, n+2m-1$, of K have been involved. The considerations of Sections 3 and 4 can be interpreted as a construction and justification of the fast (C, C^m) spline quasicollocation solver $\{(4.1)–(4.3), (5.1)\}$. For simplicity, let us confine ourselves to the case of Gauss elimination for solving (4.1) and (4.3). First of all, (3.7), (3.11) and (3.16) yield

$$\begin{aligned} \max_{0\le x\le 1} |u(x) - u_n(x)| \le \kappa_n n^{-2m} \Big(&\Phi_{m+1}\pi^{-m}\left(\|K\|_\infty + 1\right)\left\|f^{(m)}\right\|_\infty \\ + \Big(\Phi_{2m+1}\pi^{-2m}&\left(\kappa\left\|\partial_x^{2m}K\right\|_\infty + (\kappa q'_{2m}+1)\left\|\partial_y^{2m}K\right\|_\infty\right)\Big)\|f\|_\infty\Big) + \frac{\varepsilon_{n,f,K}}{n^{2m}}, \end{aligned} \tag{5.2}$$

where $\varepsilon_{n,f,K} \to 0$ as $n \to \infty$. This is estimate (1.2) with $n_\star = n^2$ and

$$c_m = \lim_{n\to\infty} c_{m,n} = \Phi_{m+1}\pi^{-m} d_{K,m}, \tag{5.3}$$

$$d_{K,m} = \left(\frac{\Phi_{2m+1}}{\Phi_{m+1}}\pi^{-m}\left(\kappa\left\|\partial_x^{2m}K\right\|_\infty + (\kappa q'_{2m}+1)\left\|\partial_y^{2m}K\right\|_\infty\right)\right) + \kappa\left(\|K\|_\infty + 1\right).$$

Further, according to Section 4.5, $(T_n f_n)(ih)$, $i = -2m+1, \dots, n+2m-1$, are available in $\sim(2m+1)n^2$ flops. The solution of (4.1) and (4.3) by the Gauss elimination costs $O\left(\nu^3\right) = o\left(n^2\right)$ flops (recall the condition $\nu \sim n^\theta$, $\frac{1}{2} < \theta < \frac{2}{3}$ in the case of Gauss elimination), whereas (4.2) can be implemented in $\sim 2n^2$ flops (see Sections 4.2 and 4.4). Thus all parameters of u_n are available at the cost of $\sim(2m+3)n^2 = b_m n_\star$ flops where $b_m = 2m+3$. For the effectiviness constant $\overline{c}_m$ (see (1.3)) of the solver $\{(4.1)–(4.3), (5.1)\}$ we obtain the formula $\overline{c}_m = (2m+3)^m c_m$ with c_m defined in (5.3).

Finally, the value $u_n(x)$ at any $x \in [0,1]$ can be computed in b'_m flops determined by the price of the evaluation of the quasi-interpolants in (5.1) when the knot values of corresponding functions are given.

For solver $\{(4.1)–(4.4), (5.1)\}$, the upper estimate (1.2) and the lower estimate (1.4) (which is true for any solver), are of the same order w.r.t. n; moreover, $d_{K,m} \to 1$ as $\|K\|_{2m,\infty} \to 0$, and the limits of the r.h.s. of estimates (1.2) and (1.4) coincide.

Estimate (5.2) contains the term $\varepsilon_{n,m,f,K}$ which converges to 0 as $n \to \infty$ but non-uniformly with respect to f from the unit ball of $C^m[0,1]$. With the help of Remark 3.1 we have an estimate of type (5.2) also with $c_{m,n}$ independent of f, resulting to (1.2) with

$$c_m = \limsup_{n\to\infty} c_{m,n} = \Phi_{m+1}\pi^{-m} d_{K,m} + (1 + \kappa\|K\|_\infty) q_m c'_m,$$

see Lemma 3 for the quantities q_m and c'_m and [6] for their numerical values.

6. Numerical Example

The results of the present paper are mainly of theoretical character but since we have introduced a new method for the solving of integral equations (the discrete quasicollocation method (3.10)), it is of interest to examine its accuracy also in numerical examples. Below we consider the model problem (1.1) with $K(x,y) = e^{-\alpha xy}$ and the solution $u(x) = e^{-\beta x}$ where α and β are positive parameters; the free term $f(x)$ can be easily computed. Note that for large α and β, an acceptable accuracy of the solution of (3.10) needs large n so far as uniform grids are used, and fast solvers, in particular, $\{(4.1)–(4.3), (5.1)\}$ will be useful; numerically, GMRES applied directly to (3.10) (without two grid iterations) is preferable due to the simplicity of its algorithm compared with (4.1)–(4.4) and since the computational cost of GMRES exceeds the complexity of (4.1)–(4.4) only by a factor $o(\log n)$.

The accuracy of the computed approximate solution $u_n = v_n + f$ is presented graphically on Fig. 1 – Fig. 7. For the parameter values $\alpha = 1, 10$, $\beta = \pi$ and $\alpha = 1$, $\beta = 10\pi$, the solution of (3.10) is of a high accuracy already for relatively small n, and corresponding system of linear equations was solved by the Gauss elimination neglecting the two grid iterations. On the other hand, for $\alpha = 1$, $\beta = 100\pi$ and for $\alpha = 30$, $\beta = \pi$, an acceptable accuracy of the solution of (3.10) is achieved for relatively large n, so we tackled (3.10) by GMRES still neglecting the two grid iterations.

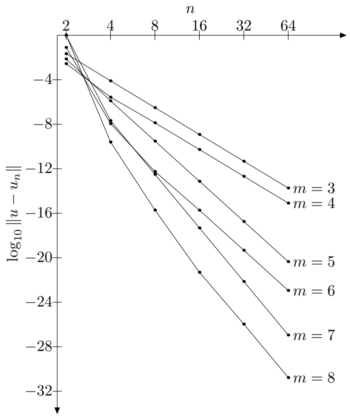

Figure 1. $\alpha = 1$, $\beta = \pi$, 256-bit real numbers, Gauss elimination.

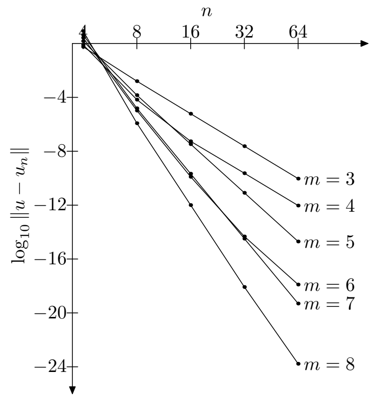

Figure 2. $\alpha = 1$, $\beta = 10\pi$, 256-bit real numbers, Gauss elimination.

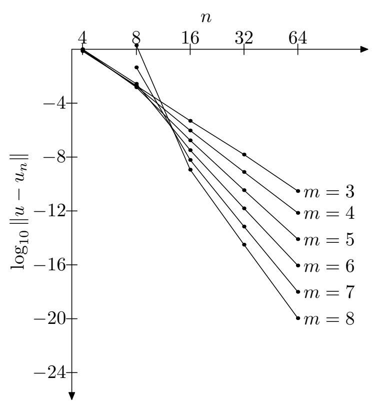

Figure 3. $\alpha = 10$, $\beta = \pi$, 256-bit real numbers, Gauss elimination.

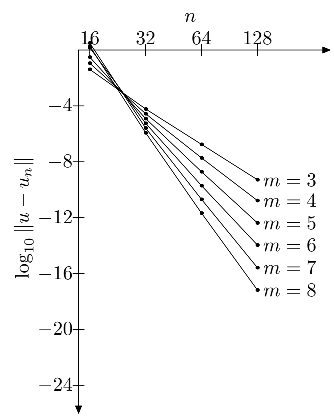

Figure 4. $\alpha = 30$, $\beta = \pi$, 256-bit real numbers, GMRES.

The computations were performed in C++ using 256-bit real numbers (type `qd_real`) made available by library `qd` version 2.2.4 (see [4]). For stability and simplicity, the evaluations of $B_m(x)$ were done in a way that the coefficients of the father B-spline B_m in (2.1) were computed beforehand as overhead.

For comparison, in the case $\alpha = 1$, $\beta = \pi$ computations using 64-bit real numbers (type `double`) were also performed as well as the problem (3.10) was also solved by GMRES together with measuring the running time of the program (see Fig. 8) (the source code was compiled under GNU C++ compiler 4.2.1 and run on a 1460 MHz Athlon XP, the host operating system being FreeBSD 6.2). Note that the low accuracy due to only 64-bit numbers slightly improves already when the values of $B_m(x)$ by formula (2.1) are computed

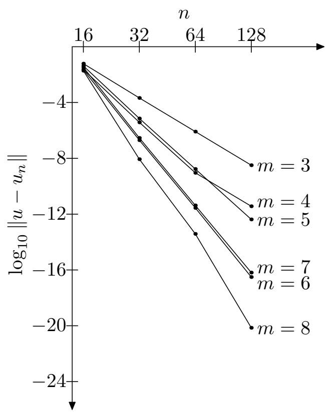

Figure 5. $\alpha = 1$, $\beta = 100\pi$, 256-bit real numbers, GMRES.

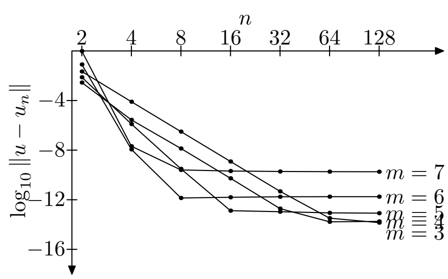

Figure 6. $\alpha = 1$, $\beta = \pi$, 64-bit real numbers, Gauss elimination.

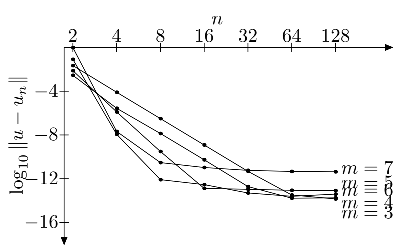

Figure 7. $\alpha = 1$, $\beta = \pi$, 64-bit real numbers ($B_m(x)$ is evaluated in 256-bit arithmetics), Gauss elimination.

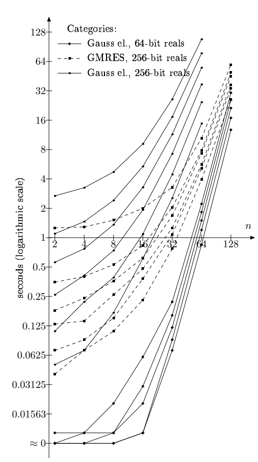

Figure 8. Program run time comparison for the case $\alpha = 1$, $\beta = \pi$. The line of lowest times in every category corresponds to $m = 3$; the greater m, the more time is consumed.

in 256-bit arithmetics (as overhead) and other calculations are still done in 64-bit arithmetics (see Fig. 6 and Fig. 7).

Acknowledgement

This work was supported by the Estonian Science Foundation, grants No 5859 and 5704.

References

[1] K. E. Atkinson. *The Numerical Solution of Integral Equations of the Second Kind*. Cambridge University Press, Cambridge, 1997.

[2] C. de Boor. *A practical Guide to Splines*. Springer, New York, 2001.

[3] K. V. Emel'yanov and A. M. Il'in. The number of the arithmetic operations necessary for the approximate solution of Fredholm integral equations of the second kind. *USSR Comput. Math. Phys*, **7**(4), 259–267, 1967. (In Russian)

[4] Y. Hida, X. S. Li and D. H. Bailey. *Library for Double-Double and Quad-Double Arithmetic*. NERSC Division, Lawrence Berkeley National Laboratory, 2007. http://www.cs.berkeley.edu/~yozo

[5] N. P. Korneychuk. *Splines in the Approximation Theory*. Nauka, Moscow, 1984. (In Russian)

[6] E. Leetma and G. Vainikko. Quasi-interpolation by splines on the uniform knot sets. *Math. Mod. Anal.*, **12**, 107–120, 2007.

[7] J. L. Lions and E. Magenes. *Non-Homogenous Boundary Value Problems and Applications, Vol. 1*. Springer, Berlin, 1972.

[8] O. Nevanlinna. *Convergence of Iterations for Linear Equations*. Birkhäuser, Basel, 1993.

[9] R. Plato. *Numerische Mathematik Kompakt*. Vieweg, Braunschweig, 2000. (In German)

[10] J. Saranen and G. Vainikko. *Periodic Integral and Pseudodifferential Equations with Numerical Approximation*. Springer, Berlin, 2002.

[11] I. J. Schoenberg. *Cardinal Interpolation*. SIAM, Philadelphia, 1973.

[12] L. L. Schumaker. *Spline Functions: Basic Theory*. Krieger Publ., Malabar, Florida, 1993.

[13] S. B. Stechkin and Yu. N. Subbotin. *Splines in Numerical Mathematics*. Nauka, Moscow, 1976. (In Russian)

[14] L. N. Trefethen and D. Bau. *Numerical Linear Algebra*. SIAM, Philadelphia, 1997.

[15] G. Vainikko. Error estimates for the cardinal spline interpolation. *ZAA*. (to appear)

[16] G. Vainikko. *Fast Solvers of Integral Equations*. Lecture notes, University of Tartu, Sept.–Dec. 2006. http://www.ut.ee/~gen/FASTlecturesSIAM.pdf

[17] G. Vainikko. Fast wavelet solvers of periodic integral equations. In: *Proc. Int. Conf. on Topics in Functional and Numerical Analysis (TOFNA-2005)*. (to appear)

[18] G. Vainikko. GMRES and discrete approximation of operators. *Proc. Estonian Acad Sci. Phys. Math*, **53**, 124–131, 2004.

[19] G. Vainikko. Fast solvers of integral equations of the second kind: quadrature methods. *J. Integr. Eq. Appl*, **17**, 91–120, 2005.

[20] G. Vainikko, A. Kivinukk and J. Lippus. Fast solvers of integral equations of the second kind: wavelet methods. *J. Complexity*, **21**, 243–273, 2005.

[21] A. G. Werschulz. Where does smoothness count the most for Fredholm equations of the second kind with noisy information? *J. Complexity*, **19**, 758–798, 2003.

[22] Yu. S. Zav'yalov, B. I. Kvasov and V. L. Miroshnichenko. *Methods of Spline Functions*. Nauka, Moscow, 1980. (In Russian)