



**University of
Zurich**^{UZH}

**Zurich Open Repository and
Archive**

University of Zurich
University Library
Strickhofstrasse 39
CH-8057 Zurich
www.zora.uzh.ch

Year: 2022

Numerical approximation of poisson problems in long domains

Chipot, Michel ; Hackbusch, Wolfgang ; Sauter, Stefan ; Veit, Alexander

DOI: <https://doi.org/10.1007/s10013-021-00512-9>

Posted at the Zurich Open Repository and Archive, University of Zurich

ZORA URL: <https://doi.org/10.5167/uzh-205956>

Journal Article

Published Version



The following work is licensed under a Creative Commons: Attribution 4.0 International (CC BY 4.0) License.


Originally published at:

Chipot, Michel; Hackbusch, Wolfgang; Sauter, Stefan; Veit, Alexander (2022). Numerical approximation of poisson problems in long domains. *Vietnam Journal of Mathematics*, 50(2):375-393.

DOI: <https://doi.org/10.1007/s10013-021-00512-9>



Numerical Approximation of Poisson Problems in Long Domains

Michel Chipot¹ · Wolfgang Hackbusch²  · Stefan Sauter¹ · Alexander Veit¹

Received: 19 October 2020 / Accepted: 6 January 2021 / Published online: 18 June 2021
© The Author(s) 2021

Abstract

In this paper, we consider the Poisson equation on a “long” domain which is the Cartesian product of a one-dimensional long interval with a $(d - 1)$ -dimensional domain. The right-hand side is assumed to have a rank-1 tensor structure. We will present and compare methods to construct approximations of the solution which have tensor structure and the computational effort is governed by only solving elliptic problems on lower-dimensional domains. A zero-th order tensor approximation is derived by using tools from asymptotic analysis (method 1). The resulting approximation is an elementary tensor and, hence has a fixed error which turns out to be very close to the best possible approximation of zero-th order. This approximation can be used as a starting guess for the derivation of higher-order tensor approximations by a greedy-type method (method 2). Numerical experiments show that this method is converging towards the exact solution. Method 3 is based on the derivation of a tensor approximation via exponential sums applied to discretized differential operators and their inverses. It can be proved that this method converges exponentially with respect to the tensor rank. We present numerical experiments which compare the performance and sensitivity of these three methods.

Keywords Poisson problem · Long domain · Asymptotic analysis · Tensor approximation · Alternating least squares

Mathematics Subject Classification (2010) 15A69 · 35B40 · 35J2 · 65K05

1 Introduction

In this paper, we consider elliptic partial differential equations on domains which are the Cartesian product of a “long” interval $I_\ell = (-\ell, \ell)$ with a $(d - 1)$ -dimensional domain ω , the *cross section* - a typical application is the modelling of a flow in long cylinders. We will investigate and compare methods which reduce the problem on the original d -dimensional

Dedicated to Bernd Sturmfels on the occasion of his 60th birthday.

✉ Wolfgang Hackbusch
wh@mis.mpg.de

Extended author information available on the last page of the article.

domain to a problem on the cross section of the domain. Such type of dimension reduction methods have important practical applications in fluid and continuum mechanics in order to reduce the computational complexity. We refer, e.g., to [27] for different kinds of application. As a model problem we consider the Poisson equation with homogeneous Dirichlet boundary conditions and a right-hand side which is an *elementary tensor*; i.e., the product of a univariate function (on the long interval) and a $(d - 1)$ -variate function on the cross section. Such problems have been studied by using asymptotic analysis, see., e.g., [9]. Our first approximation (method 1) is based on this technique and approximates the solution by an elementary tensor where the function on the cross section is the solution of a Poisson-problem on the cross section and the corresponding univariate function is determined afterwards as the best approximation in the Sobolev space H_0^1 on the long interval. In Lemma 2 below, it is shown that this approximation converges exponentially with respect to the length of the cylinder for any subdomain $I_{\ell_0} \times \omega$ for fixed $\ell_0 < \ell$. However, for fixed ℓ this is a one-term approximation with a fixed error.

Method 2 uses the result of method 1 as the initial guess for an iterative procedure which results in a *greedy*-type method. Recursively, one assumes that a rank- k tensor approximation of the solution has already been derived and then starts an iteration to compute the $k + 1$ term: a) one chooses an univariate function on I_ℓ as an initial guess for this iteration and determines the function on the cross section as the best approximation in H_0^1 of the cross-section. In step b) the iteration is flipped and one fixes the new function on the cross section and determines the corresponding best approximation in H_0^1 of the interval. Steps a) and b) are iterated until a stopping criterion is reached and this gives the $k + 1$ term in the tensor approximation. In the literature this approach is also known as *Proper Generalized Decomposition* (PGD) [7]. We have performed numerical experiments which are reported in Section 4 which show that this method leads to a convergent approximation also for fixed ℓ as the tensor rank of the approximation increases. However, it turns out that this method is quite sensitive and requires that the inner iteration a), b) leads to an accurate approximation of the $(k + 1)$ term in order to ensure that the outer iteration is converging. This greedy algorithm has similarities with *alternating least squares* (ALS) methods which would alternately optimize the low-rank factors of the approximate solution. While the greedy method has the advantage that it can be combined with method 1 in a natural fashion, the numerical experiments which we have performed indicate that the convergence speed can slow down as the number of outer iterations increases. This is well known and there are various strategies to accelerate the convergence speed such as Galerkin projection or one half-step of ALS on the low-rank factors. The former is discussed in [19], the latter in the context of PGD applied to uncertainty quantification. If the structure allows to conveniently perform one half-step or even a full step of ALS, the convergence gets substantially accelerated. We have considered here the original greedy method because its popularity related to its straightforward implementation; it is well suited when only a moderate approximation accuracy of the Poisson problem is required.

Method 3 is based on a different approach which employs numerical tensor calculus (see [14]). First one defines an exponential sum approximation of the function $1/x$. Since the differential operator $-\Delta$ is of tensor form, the exponential sum, applied to the inverse of a discretisation of the Laplacian by a matrix which must preserve the tensor format, directly leads to a tensor approximation of the solution u . We emphasize that the explicit computation of the inverse of the discretisation matrix can be avoided by using the *hierarchical format* for their representation (see [15]). An advantage of this method is that a full theory is available which applies to our application and allows us to choose the tensor rank

via an a priori error estimate. It also can be shown that the tensor approximation converges exponentially with respect to the tensor rank (see [14]).

The goal of this paper is to compare three popular approaches for the numerical approximation of Poisson problems in domains of the form $I_\ell \times \omega$ and to assess their performance with respect to the length ℓ via numerical experiments. These methods exhibit different computational complexities and our results can be used to determine a suitable method given a desired accuracy range. For an in-depth theoretical analysis of the presented methods we refer to the existing literature.

There exist various other approaches in the literature such as DMRG [24], AMEN [10], and QTT [17, 18, 23], and hierarchical local model reduction by domain decomposition [26] which also have in principle the potential to solve Poisson-type problems on long domains. We also mention here that the discretized equation (see eq. (19)) takes the form of a Sylvester matrix equation. The numerical solution of Sylvester equations has been discussed at length in the literature; see, e.g., [28]. In particular, an ADI method or rational Krylov subspace method with optimally chosen shifts (which constructs the rational approximation directly and not via exponential sums) will be more efficient than methods 2 and 3. We have chosen the proposed method 1-3 for a systematic comparison since they were developed in different areas of mathematics and all three are very popular. In the conclusions, we will summarize the behaviour of these methods depending on the kind of application and the accuracy requirements.

The paper is structured as follows. In Section 2 we formulate the problem on the long product domain and introduce the assumptions on the *tensor format* of the right-hand side. The three different methods for constructing a tensor approximation of the solution are presented in Section 3. The results of numerical experiments are presented in Section 4 where the convergence and sensitivity of the different methods is investigated and compared. For the experiments we consider first the case that the cross section is the one-dimensional unit interval and then the more complicated case that the cross section is an L-shaped polygonal domain. Finally, in the concluding section we summarize the results and give an outlook.

2 Setting

Let ω be an open, bounded and connected Lipschitz domain in \mathbb{R}^{d-1} , $d \geq 1$. In the following we consider Poisson problems on domains of the form

$$\Omega_\ell := I_\ell \times \omega \quad \text{with} \quad I_\ell := (-\ell, \ell),$$

where ℓ is large. We are interested in Dirichlet boundary value problems of the form

$$\begin{aligned} -\Delta u_\ell &= F && \text{in } \Omega_\ell, \\ u_\ell &= 0 && \text{on } \partial\Omega_\ell \end{aligned} \tag{1}$$

with weak formulation

$$\begin{cases} \text{find } u_\ell \in H_0^1(\Omega_\ell) \text{ s.t.} \\ (\nabla u_\ell, \nabla v)_{L_2(\Omega_\ell)} = (F, v)_{L_2(\Omega_\ell)} \quad \forall v \in H_0^1(\Omega_\ell). \end{cases}$$

Specifically we are interested in right-hand sides f which have a tensor structure of the form $F = 1 \otimes f$ or more generally $F = \sum_{k=0}^n g_k \otimes f_k$, $n \in \mathbb{N}$, where g_k is a univariate function and f, f_k are functions which depend only on the $(d - 1)$ -dimensional variable $x' \in \omega$. Here, we use the standard tensor notation $(g \otimes f)(x) = g(x_1) f(x')$ with $x' = (x_k)_{k=2}^d$. In this paper, we will present and compare methods to approximate u_ℓ in tensor form.

We consider a right-hand side of the form

$$F = 1 \otimes f \quad \text{for some } f \in L^2(\omega) \quad (2)$$

and derive a first approximation of u_ℓ as the solution of the $(d - 1)$ -dimensional problem on ω :

$$\begin{aligned} -\Delta' u_\infty(x') &= f(x') & \text{in } \omega, \\ u_\infty &= 0 & \text{on } \partial\omega \end{aligned} \quad (3)$$

with weak form

$$\begin{cases} \text{find } u_\infty \in H_0^1(\omega) \text{ s.t.} \\ (\nabla' u_\infty, \nabla' v)_{L_2(\omega)} = (f, v)_{L_2(\omega)} \quad \forall v \in H_0^1(\omega). \end{cases}$$

3 Numerical Approximation

In this section we derive three different methods to approximate problem (1). In all three methods we exploit the special structure of the domain Ω_ℓ and the right-hand side F . Our goal is to reduce the original d -dimensional problem on Ω_ℓ to one or more $(d - 1)$ -dimensional problems on ω . Compared to standard methods like finite elements methods or finite difference methods, which solve the equations on Ω_ℓ , this strategy can significantly reduce the computational cost since ℓ is considered large and the discretisation in the x_1 direction can be avoided.

3.1 Method 1: A One-term Approximation Based on an Asymptotic Analysis of Problem (1)

Although the right-hand side F in (1) is independent of x_1 , it is easy to see that this is not the case for the solution u_ℓ , i.e., due to the homogeneous Dirichlet boundary conditions it is clear that u_ℓ depends on x_1 . However, if ℓ is large one can expect that u_ℓ is approximately constant with respect to x_1 in a subdomain Ω_{ℓ_0} , where $0 < \ell_0 \ll \ell$ and thus converges locally to a function independent of x_1 for $\ell \rightarrow \infty$. The asymptotic behaviour of the solution u_ℓ when $\ell \rightarrow \infty$ has been investigated in [8]. It can be shown that

$$u_\ell \longrightarrow 1 \otimes u_\infty \quad \text{in } \Omega_{\ell_0},$$

where u_∞ is the solution of (3), with an exponential rate of convergence. More precisely, the following theorem holds:

Theorem 1 *There exist constants $c, \alpha > 0$ independent of ℓ s.t.*

$$\int_{\Omega_{\ell/2}} |\nabla(u_\ell - 1 \otimes u_\infty)|^2 dx \leq c e^{-\alpha\ell} \|f\|_{2,\omega}^2,$$

where $\|\cdot\|_{2,\omega}$ refers to the $L^2(\omega)$ -norm.

For a proof we refer to [8, Theorem 6.6].

Theorem 1 shows that $1 \otimes u_\infty$ is a good approximation of u_ℓ in $\Omega_{\ell/2}$ when ℓ is large. This motivates to seek approximations of u_ℓ in Ω_ℓ which are of the form

$$u_\ell \approx u_\ell^{M_1} := \psi_\ell \otimes u_\infty,$$

where $\psi_\ell \in H_0^1(-\ell, \ell)$. Here, we choose ψ_ℓ to be the solution of the following best approximation problem: Given $u_\ell \in H_0^1(\Omega_\ell)$ and $u_\infty \in H_0^1(\omega)$, find $\psi \in H_0^1(-\ell, \ell)$ s.t.

$$\|\nabla(u_\ell - \psi \otimes u_\infty)\|_2 = \inf_{\theta \in H_0^1(-\ell, \ell)} \|\nabla(u_\ell - \theta \otimes u_\infty)\|_2. \tag{4}$$

In order to solve problem (4) we define the functional

$$J(u_\ell, u_\infty)(\theta) := \|\nabla(u_\ell - \theta \otimes u_\infty)\|_2^2$$

and consider the variational problem of minimizing it with respect to $\theta \in H_0^1(-\ell, \ell)$.

A simple computation shows that this is equivalent to finding $\tilde{\theta} \in H_0^1(I_\ell)$ such that

$$\left. \begin{aligned} & \left(\nabla(\theta \otimes u_\infty), \nabla(\tilde{\theta} \otimes u_\infty) \right)_2 = \left(\nabla u_\ell, \nabla(\tilde{\theta} \otimes u_\infty) \right)_2 \\ \iff & \left(\begin{pmatrix} \theta' \otimes u_\infty \\ \theta \otimes \nabla' u_\infty \end{pmatrix}, \begin{pmatrix} \tilde{\theta}' \otimes u_\infty \\ \tilde{\theta} \otimes \nabla' u_\infty \end{pmatrix} \right)_2 = \left(-\Delta u_\ell, \tilde{\theta} \otimes u_\infty \right)_2 \\ \iff & \alpha_\infty^2 \left(\theta', \tilde{\theta}' \right)_{2, I_\ell} + \beta_\infty^2 \left(\theta, \tilde{\theta} \right)_{2, I_\ell} = \left(1 \otimes f, \tilde{\theta} \otimes u_\infty \right)_2 \\ \iff & \alpha_\infty^2 \left(\theta', \tilde{\theta}' \right)_{2, I_\ell} + \beta_\infty^2 \left(\theta, \tilde{\theta} \right)_{2, I_\ell} = \left(1 \otimes (-\Delta' u_\infty), \tilde{\theta} \otimes u_\infty \right)_2 \\ \iff & \alpha_\infty^2 \left(\theta', \tilde{\theta}' \right)_{2, I_\ell} + \beta_\infty^2 \left(\theta, \tilde{\theta} \right)_{2, I_\ell} = \beta_\infty^2 \int_{I_\ell} \tilde{\theta}. \end{aligned} \right\} \quad \forall \tilde{\theta} \in H_0^1(I_\ell)$$

with

$$\alpha_\infty := \|u_\infty\|_{2, \omega}, \quad \beta_\infty := \|\nabla' u_\infty\|_{2, \omega}.$$

The strong form of the resulting equation is

$$\begin{aligned} -\alpha_\infty^2 \theta'' + \beta_\infty^2 \theta &= \beta_\infty^2 \quad \text{on } (-\ell, \ell), \\ \theta(-\ell) = \theta(\ell) &= 0. \end{aligned}$$

The solution of this one-dimensional boundary value problem is given by

$$\theta(x_1) := 1 - \frac{\cosh\left(\frac{\beta_\infty}{\alpha_\infty} x_1\right)}{\cosh\left(\frac{\beta_\infty}{\alpha_\infty} \ell\right)}.$$

This shows that an approximation of our original problem (1) is given by

$$u_\ell^{M_1} := \psi_\ell(\lambda_\infty, \cdot) \otimes u_\infty, \quad \text{with } \psi_\ell(a, x_1) := 1 - \frac{\cosh(ax_1)}{\cosh(a\ell)} \tag{5}$$

and

$$\lambda_\infty = \frac{\beta_\infty}{\alpha_\infty} = \frac{\sqrt{(f, u_\infty)_{2, \omega}}}{\alpha_\infty}.$$

Note that $\psi_\ell(a, \cdot)$ satisfies

$$-\psi_\ell''(a, \cdot) + a^2 \psi_\ell(a, \cdot) = a^2 \quad \text{and} \quad \psi_\ell(\pm\ell) = 0. \tag{6}$$

In Section 4 we report on various numerical experiments that show the approximation properties of this rather simple one-term approximation.

Figure 1 shows a plot of $\psi_\ell(\lambda_\infty, \cdot)$ for $\ell = 20$ and $\lambda_\infty = 2$. Since ψ_ℓ approaches 1 with an exponential rate as x_1 moves away from $\pm\ell$ towards the origin, an analogous result to Theorem 1 can be shown for $u_\ell^{M_1}$.

Lemma 2 *There exist constants $c, \tilde{c} > 0$ independent of ℓ such that, for $\delta_\ell < \ell$,*

$$\|\nabla(u_\ell - \psi_\ell(\lambda_\infty, \cdot) \otimes u_\infty)\|_{2, \Omega_{\ell-\delta_\ell}}^2 \leq C_{\omega, \delta_\ell}^{(1)} \|u_\infty\|_{2, \omega}^2 + C_{\omega, \delta_\ell}^{(2)} \|\nabla' u_\infty\|_{2, \omega}^2, \tag{7}$$

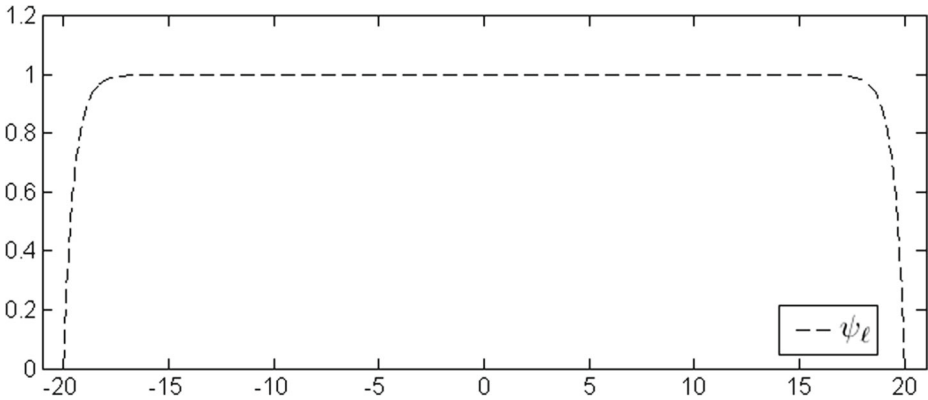


Fig. 1 Plot of $\psi_\ell(\lambda_\infty, \cdot)$ for $\ell = 20$ and $\lambda_\infty = 2$

with

$$C_{\omega, \delta_\ell}^{(1)} := 4e^{-2\lambda_1 \delta_\ell}, \quad C_{\omega, \delta_\ell}^{(2)} := 4 \left(\frac{1}{\lambda_1} e^{-2\lambda_1 \delta_\ell} + 6(\ell - \delta_\ell) e^{-2\lambda_1 \ell} \right)$$

and

$$\lambda_1 := \inf_{v \in H_0^1(\omega) \setminus \{0\}} \frac{\|\nabla' v\|_{2, \omega}}{\|v\|_{2, \omega}}. \tag{8}$$

The right-hand side in (7) goes to 0 with an exponential rate of convergence if δ_ℓ is bounded from below when $\ell \rightarrow \infty$.

Proof For $i = 1, 2, \dots$, let w_i be the i -th eigenfunction of $-\Delta'$, i.e., $w_i \in H_0^1(\omega)$ is a solution of

$$(\nabla' w_i, \nabla' v)_{2, \omega} = \lambda_i^2 (w_i, v)_{2, \omega} \quad \forall v \in H_0^1(\omega) \tag{9}$$

and we normalize the eigenfunctions such that $(w_i, w_j)_{2, \omega} = \delta_{i, j}$ and order them such that $(\lambda_i)_i$ is increasing monotonously. Furthermore let $u_{\ell, i} \in H_0^1(\Omega_\ell)$ be the solution of

$$(\nabla u_{\ell, i}, \nabla v)_{2, \Omega_\ell} = \lambda_i^2 (1 \otimes w_i, v)_{2, \Omega_\ell} \quad \forall v \in H_0^1(\Omega_\ell).$$

Then one concludes from (6) and (9) that

$$u_{\ell, i} = \psi_\ell(\lambda_i, \cdot) \otimes w_i.$$

If $f \in L^2(\omega)$ it holds

$$f = \sum_{i=1}^{\infty} (f, w_i)_{2, \omega} w_i.$$

This shows that the solutions of (3) and (1) can be expressed as

$$u_\infty = \sum_{i=1}^{\infty} \frac{(f, w_i)_{2, \omega}}{\lambda_i^2} w_i,$$

$$u_\ell = \sum_{i=1}^{\infty} \frac{(f, w_i)_{2, \omega}}{\lambda_i^2} u_{\ell, i} = \sum_{i=1}^{\infty} \frac{(f, w_i)_{2, \omega}}{\lambda_i} \psi_\ell(\lambda_i, \cdot) \otimes w_i.$$

With ψ_ℓ as in (5) we get

$$u_\ell - \psi_\ell(\lambda_\infty, \cdot) \otimes u_\infty = \sum_{i=1}^{\infty} \frac{(f, w_i)_{2, \omega}}{\lambda_i^2} \phi_{\ell, i} \otimes w_i, \quad \phi_{\ell, i}(x_1) := \frac{\cosh(\lambda_\infty x_1)}{\cosh(\lambda_\infty \ell)} - \frac{\cosh(\lambda_i x_1)}{\cosh(\lambda_i \ell)}.$$

Let $\delta_\ell < \ell$. Then, since $\int_\omega w_i w_j dx' = \delta_{i,j}$, we get

$$\begin{aligned} \|\nabla(u_\ell - \psi_\ell(\lambda_\infty, \cdot) \otimes u_\infty)\|_{2, \Omega_{\ell-\delta_\ell}}^2 &= \int_{-\ell+\delta_\ell}^{\ell-\delta_\ell} \int_\omega |\nabla(u_\ell - \psi_\ell(\lambda_\infty, \cdot) \otimes u_\infty)|^2 dx \\ &= \sum_{i=1}^\infty \frac{(f, w_i)^2}{\lambda_i^4} \int_{-\ell+\delta_\ell}^{\ell-\delta_\ell} \left((\phi'_{\ell,i})^2 + \lambda_i^2 \phi_{\ell,i}^2 \right). \end{aligned} \tag{10}$$

One has for any $\alpha > 0$

$$\begin{aligned} \int_{-\ell+\delta_\ell}^{\ell-\delta_\ell} \left(\frac{\cosh(\alpha x_1)}{\cosh(\alpha \ell)} \right)^2 dx_1 &= 2 \int_0^{\ell-\delta_\ell} \left(\frac{\cosh(\alpha x_1)}{\cosh(\alpha \ell)} \right)^2 dx_1 \\ &= \frac{1}{2} \int_0^{\ell-\delta_\ell} \frac{e^{2\alpha x_1} + 2 + e^{-2\alpha x_1}}{\cosh(\alpha \ell)^2} dx_1 \\ &\leq 2 \int_0^{\ell-\delta_\ell} \frac{e^{2\alpha x_1} + 3}{e^{2\alpha \ell}} dx_1 \\ &\leq \frac{1}{\alpha} e^{-2\alpha \delta_\ell} + 6(\ell - \delta_\ell) e^{-2\alpha \ell} \end{aligned}$$

and similarly

$$\int_{-\ell+\delta_\ell}^{\ell-\delta_\ell} \left(\frac{\sinh(\alpha x_1)}{\cosh(\alpha \ell)} \right)^2 dx_1 \leq \frac{1}{\alpha} e^{-2\alpha \delta_\ell}.$$

Since $\lambda_1 \leq \lambda_i$ for all $i \in \mathbb{N}$ and

$$\lambda_1 = \inf_{v \in H_0^1(\omega) \setminus \{0\}} \frac{\|\nabla' v\|_{2,\omega}}{\|v\|_{2,\omega}} \leq \frac{\|\nabla' u_\infty\|_{2,\omega}}{\|u_\infty\|_{2,\omega}} = \lambda_\infty,$$

we get

$$\int_{-\ell+\delta_\ell}^{\ell-\delta_\ell} (\phi'_{\ell,i})^2 \leq 2e^{-2\lambda_\infty \delta_\ell} + 2e^{-2\lambda_1 \delta_\ell} \leq 4e^{-2\lambda_1 \delta_\ell} = C_{\omega, \delta_\ell}^{(1)} \tag{11}$$

and

$$\begin{aligned} \int_{-\ell+\delta_\ell}^{\ell-\delta_\ell} \phi_{\ell,i}^2 &\leq 2 \int_{-\ell+\delta_\ell}^{\ell-\delta_\ell} \left(\left| \frac{\cosh(\lambda_\infty x_1)}{\cosh(\lambda_\infty \ell)} \right|^2 + \left| \frac{\cosh(\lambda_i x_1)}{\cosh(\lambda_i \ell)} \right|^2 \right) \\ &\leq 2 \left(\frac{1}{\lambda_\infty} e^{-2\lambda_\infty \delta_\ell} + \frac{1}{\lambda_i} e^{-2\lambda_i \delta_\ell} + 6(\ell - \delta_\ell) e^{-2\lambda_\infty \ell} + 6(\ell - \delta_\ell) e^{-2\lambda_i \ell} \right) \\ &\leq 4 \left(\frac{1}{\lambda_1} e^{-2\lambda_1 \delta_\ell} + 6(\ell - \delta_\ell) e^{-2\lambda_1 \ell} \right) = C_{\omega, \delta_\ell}^{(2)}. \end{aligned} \tag{12}$$

We employ the estimates (11) and (12) in (10) and obtain

$$\begin{aligned} \|\nabla(u_\ell - \psi_\ell(\lambda_\infty, \cdot) \otimes u_\infty)\|_{2, \Omega_{\ell-\delta_\ell}}^2 &\leq C_{\omega, \delta_\ell}^{(1)} \sum_{i=1}^\infty \frac{(f, w_i)_{2,\omega}^2}{\lambda_i^4} + C_{\omega, \delta_\ell}^{(2)} \sum_{i=1}^\infty \frac{(f, w_i)_{2,\omega}^2}{\lambda_i^2} \\ &= C_{\omega, \delta_\ell}^{(1)} \|u_\infty\|_{2,\omega}^2 + C_{\omega, \delta_\ell}^{(2)} \|\nabla' u_\infty\|_{2,\omega}^2, \end{aligned}$$

which shows the assertion. □

Lemma 2 suggests that one cannot expect convergence of the approximation $\psi_\ell(\lambda_\infty, \cdot) \otimes u_\infty$ on the whole domain Ω_ℓ . Indeed it can be shown that, in general, $\|\nabla(u_\ell - \psi_\ell(\lambda_\infty, \cdot) \otimes u_\infty)\|_{2, \Omega_\ell} \not\rightarrow 0$ as $\ell \rightarrow \infty$. Setting $\delta_\ell = 0$ in Lemma 2 shows that the error on Ω_ℓ can be estimated as follows:

Corollary 3 *It holds*

$$\|\nabla(u_\ell - \psi_\ell(\lambda_\infty, \cdot) \otimes u_\infty)\|_{2,\Omega_\ell}^2 \leq 4 \left(\|u_\infty\|_{2,\omega}^2 + 6\ell e^{-2\lambda_1\ell} \|\nabla' u_\infty\|_{2,\omega}^2 \right),$$

where λ_1 is as in (8).

3.2 Method 2: An Alternating Least Squares Type Iteration

Method 1 can be interpreted as a 2-step algorithm to obtain an approximation $u_\ell^{M_1}$ of u_ℓ .

- Step 1: Solve (3) in order to obtain an approximation of the form $1 \otimes u_\infty$ which is *non-conforming*, i.e., does not belong to $H_0^1(\Omega_\ell)$.
- Step 2: Using u_∞ , find a function ψ_ℓ that satisfies (4) in order to obtain the *conforming* approximation $u_\ell^{M_1} := \psi_\ell(\lambda_\infty, \cdot) \otimes u_\infty \in H_0^1(\Omega_\ell)$.

In this section we extend this idea and seek approximations of the form

$$u_{\ell,m}^{M_2} = \sum_{j=0}^m p^{(j)} \otimes q^{(j)} \tag{13}$$

by iteratively solving least squares problems similar to (4). We denote by

$$\text{Res}_m = u_\ell - u_{\ell,m}^{M_2} = u_\ell - \sum_{j=0}^m p^{(j)} \otimes q^{(j)}$$

the residual of the approximation and suggest the following iteration to obtain $u_{\ell,m}^{M_2}$:

- $m = 0$: Set $q^{(0)} = u_\infty$ and $p^{(0)} = \psi_\ell(\lambda_\infty, \cdot)$.
- $m > 0$: Find $q^{(m)} \in H_0^1(\omega)$ s.t.

$$q^{(m)} = \arg \min_{q \in H_0^1(\omega)} \left\| \nabla \left(\text{Res}_{m-1} - p^{(m-1)} \otimes q \right) \right\|_2. \tag{14}$$

Then, given $q^{(m)}$, find $p^{(m)} \in H_0^1(I_\ell)$ s.t.

$$p^{(m)} = \arg \min_{p \in H_0^1(I_\ell)} \left\| \nabla \left(\text{Res}_{m-1} - p \otimes q^{(m)} \right) \right\|_2. \tag{15}$$

Iterate (14) and (15) until a stopping criterion is reached (inner iteration). Then set $\text{Res}_m = \text{Res}_{m-1} - p^{(m)} \otimes q^{(m)}$.

The algorithm exhibits properties of a greedy algorithm. It is easy to see that in each step of the (outer) iteration the error decreases or stays constant. We focus here on its accuracy in comparison with the two other methods via numerical experiments. We emphasize that for tensors of order at least 3, convergence can be shown for the (inner) iteration (see [11, 25, 29, 30]). This limit, however, is not a global minimum in general. The outer iteration can be shown to converge as well against the true solution u_ℓ under the condition that we find the best rank-1 approximation in the inner iteration (see [12]).

The idea of computing approximations in the separated form (13) by iteratively enriching the current solution with rank-1 terms is known in the literature as *Proper Generalized Decomposition* (PGD). The PGD has been applied to various problems in computational mechanics (e.g. [1, 4, 5, 21]), computational rheology [6], quantum chemistry (e.g. [2, 3]) and others.

An extensive review of the method can be found in [7]. For an error and convergence analysis of the (outer) iteration in the case of the Poisson equation we also refer to [20], where a similar (but not identical) approach as ours is considered.

In each step of the (outer) iteration above we need to solve at least two minimization problems (14) and (15). In the following we derive the strong formulations of these problems.

3.2.1 Resolution of (14)

As before an investigation of the functional

$$J(q^{(m)}) := \|\nabla(\text{Res}_{m-1} - p^{(m-1)} \otimes q^{(m)})\|_2^2$$

shows that $q^{(m)}$ needs to satisfy

$$\begin{aligned} & \left(\nabla \left(p^{(m-1)} \otimes q^{(m)} \right), \nabla \left(p^{(m-1)} \otimes q \right) \right)_2 = \left(\nabla \text{Res}_{m-1}, \nabla \left(p^{(m-1)} \otimes q \right) \right)_2 \\ \iff & p_{0,m-1} \left(-\Delta' q^{(m)}, q \right)_{2,\omega} + p_{1,m-1} \left(q^{(m)}, q \right)_{2,\omega} = \left(-\Delta \text{Res}_{m-1}, p^{(m-1)} \otimes q \right)_2 \end{aligned}$$

for all $q \in H_0^1(\omega)$, where

$$p_{0,m-1} := \left\| p^{(m-1)} \right\|_{2,I_\ell}^2, \quad p_{1,m-1} := \left\| \left(p^{(m-1)} \right)' \right\|_{2,I_\ell}^2.$$

For the right-hand side we obtain

$$\begin{aligned} \left(-\Delta \text{Res}_{m-1}, p^{(m-1)} \otimes q \right)_2 &= \left(-\Delta \left(u_\ell - \sum_{j=0}^{m-1} p^{(j)} \otimes q^{(j)} \right), p^{(m-1)} \otimes q \right)_2 \\ &= \left(1 \otimes f, p^{(m-1)} \otimes q \right)_2 + \left(\sum_{j=0}^{m-1} \left(p^{(j)} \right)'' \otimes q^{(j)} + p^{(j)} \otimes \Delta' q^{(j)}, p^{(m-1)} \otimes q \right)_2 \\ &= \tilde{p}_{m-1}(f, q)_{2,\omega} + \sum_{j=0}^{m-1} \left(\tilde{p}_{2,j,m-1} \left(q^{(j)}, q \right)_{2,\omega} + \tilde{p}_{0,j,m-1} \left(\Delta' q^{(j)}, q \right)_{2,\omega} \right), \end{aligned}$$

where

$$\tilde{p}_{m-1} := \int_{-\ell}^\ell p^{(m-1)}, \quad \tilde{p}_{2,j,m-1} := \left(\left(p^{(j)} \right)'' , p^{(m-1)} \right)_{2,I_\ell}, \quad \tilde{p}_{0,j,m-1} := \left(p^{(j)}, p^{(m-1)} \right)_{2,I_\ell}.$$

In order to compute (14) we therefore have to solve in ω

$$- p_{0,m-1} \Delta' q^{(m)} + p_{1,m-1} q^{(m)} = \tilde{p}_{m-1} f + \sum_{j=0}^{m-1} \left(\tilde{p}_{2,j,m-1} q^{(j)} + \tilde{p}_{0,j,m-1} \Delta' q^{(j)} \right). \tag{16}$$

3.2.2 Resolution of (15)

Setting the derivative of the functional

$$J(p^{(m)}) := \|\nabla(\text{Res}_{m-1} - p^{(m)} \otimes q^{(m)})\|_2^2$$

to zero, shows that $p^{(m)}$ needs to satisfy

$$\begin{aligned} & \left(\nabla \left(p^{(m)} \otimes q^{(m)} \right), \nabla \left(p \otimes q^{(m)} \right) \right)_2 = \left(\nabla \text{Res}_m, \nabla \left(p \otimes q^{(m)} \right) \right)_2 \\ \iff & -q_{0,m} \left(\left(p^{(m)} \right)'', p \right)_{2,I_\ell} + q_{1,m} \left(p^{(m)}, p \right)_{2,I_\ell} = \left(-\Delta \text{Res}_m, p \otimes q^{(m)} \right)_2 \end{aligned}$$

for all $p \in H_0^1(-\ell, \ell)$, where

$$q_{0,m} = \|q^{(m)}\|_{2,\omega}^2, \quad q_{1,m} = \|\nabla' q^{(m)}\|_{2,\omega}^2.$$

For the right-hand side we obtain

$$\begin{aligned} \left(-\Delta \text{Res}_{m-1}, p \otimes q^{(m)} \right)_2 &= \left(-\Delta \left(u_\ell - \sum_{j=0}^{m-1} p^{(j)} \otimes q^{(j)} \right), p \otimes q^{(m)} \right)_2 \\ &= \left(1 \otimes f, p \otimes q^{(m)} \right)_2 + \left(\sum_{j=0}^{m-1} \left(\left(p^{(j)} \right)'' \otimes q^{(j)} + p^{(j)} \otimes \Delta' q^{(j)} \right), p \otimes q^{(m)} \right)_2 \\ &= \tilde{q}_m \int_{-\ell}^\ell p + \sum_{j=0}^{m-1} \left(\tilde{q}_{0,j,m} \left(\left(p^{(j)} \right)'', p \right)_{2,I_\ell} + \tilde{q}_{2,j,m} \left(p^{(j)}, p \right)_{2,I_\ell} \right), \end{aligned}$$

where

$$\tilde{q}_m := \left(f, q^{(m)} \right)_{2,\omega}, \quad \tilde{q}_{2,j,m} := \left(\Delta' q^{(j)}, q^{(m)} \right)_{2,\omega}, \quad \tilde{q}_{0,j,m} := \left(q^{(j)}, q^{(m)} \right)_{2,\omega}.$$

In order to obtain the solution of (15) we therefore have to solve in I_ℓ

$$-q_{0,m} \left(p^{(m)} \right)'' + q_{1,m} p^{(m)} = \tilde{q}_m + \sum_{j=0}^{m-1} \left(\tilde{q}_{2,j,m} p^{(j)} + \tilde{q}_{0,j,m} \left(p^{(j)} \right)'' \right). \tag{17}$$

Remark 4 The constants $p_{1,m-1}$, $\tilde{p}_{2,j,m-1}$, $q_{1,m}$ and $\tilde{q}_{2,j,m}$ involve derivatives and Laplace-operators. Note that after solving (16) and (17) for $q^{(m)}$ and $p^{(m)}$, discrete versions of $\Delta' q^{(m)}$ and $(p^{(m)})''$ can be easily obtained via the same equations. Furthermore, since

$$q_{1,m} = \|\nabla' q^{(m)}\|_{2,\omega}^2 = \left(-\Delta' q^{(m)}, q^{(m)} \right)_{2,\omega} = -\tilde{q}_{2,m,m}$$

and

$$p_{1,m-1} = \left\| \left(p^{(m-1)} \right)' \right\|_{2,I_\ell}^2 = \left(- \left(p^{(m-1)} \right)'', p^{(m-1)} \right)_{2,I_\ell} = -\tilde{p}_{2,m-1,m-1}$$

a numerical computation of the gradients can be avoided.

3.3 Method 3: Exploiting the Tensor Product Structure of the Operator

In this section we exploit the tensor product structure of the Laplace operator and the domain Ω_ℓ . Recall that

$$\Omega_\ell = I_\ell \times \omega.$$

Note that we do not assume that ω has a tensor product structure. Furthermore the Laplace operator in our original problem (1) can be written as

$$-\Delta = -\partial_1^2 - \Delta'. \tag{18}$$

We discretize (1) with F as in (2) on a mesh \mathcal{G} , e.g., by finite elements or finite differences on a tensor mesh, i.e., each mesh cell has the form $(x_{i-1}, x_i) \times \tau_j$, where τ_j is an element of the mesh for ω . The essential assumption is that the system matrix for the discrete version of $-\Delta$ in (18) is of the tensor form

$$A = A_1 \otimes M' + M_1 \otimes A' \tag{19}$$

If we discretize with a finite difference scheme on an equidistant grid for I_ℓ with step size h , then A_1 is the tridiagonal matrix $h^{-2}\text{tridiag}[-1, 2, -1]$ and M_1 is the identity matrix. A finite element discretisation with piecewise linear elements leads as well to $A_1 = h^{-2}\text{tridiag}[-1, 2, -1]$, while $M_{x_1} = \text{tridiag}[1/6, 2/3, 1/6]$. It can be shown that the inverse of the matrix A can be efficiently approximated with a sum of matrix exponentials. More precisely the following Theorem holds which is proved in [14, Proposition 9.34].

Theorem 5 *Let $M^{(j)}, A^{(j)}$ be positive definite matrices with $\lambda_{\min}^{(j)}$ and $\lambda_{\max}^{(j)}$ being the extreme eigenvalues of the generalized eigenvalue problem $A^{(j)}x = \lambda M^{(j)}x$ and set*

$$A = A^{(1)} \otimes M^{(2)} \otimes \dots \otimes M^{(n)} + M^{(1)} \otimes A^{(2)} \otimes \dots \otimes M^{(n)} + \dots + M^{(1)} \otimes \dots \otimes M^{(n-1)} \otimes A^{(n)}.$$

Then A^{-1} can be approximated by

$$B := \left(\sum_{v=1}^r a_{v,[a,b]} \bigotimes_{j=1}^n \exp \left(-\alpha_{v,[a,b]} \left(M^{(j)} \right)^{-1} A^{(j)} \right) \right) \left(\bigotimes_{j=1}^n \left(M^{(j)} \right)^{-1} \right),$$

where the coefficients $a_v, \alpha_v > 0$ are such that

$$\begin{aligned} \varepsilon \left(\frac{1}{x}, [a, b], r \right) &:= \left\| \frac{1}{x} - \sum_{v=1}^r a_{v,[a,b]} e^{-\alpha_{v,[a,b]}x} \right\|_{[a,b]} \\ &= \inf \left\{ \left\| \frac{1}{x} - \sum_{v=1}^r b_v e^{-\beta_v x} \right\|_{[a,b],\infty} : b_v, \beta_v \in \mathbb{R} \right\} \end{aligned}$$

with $a := \sum_{j=1}^n \lambda_{\min}^{(j)}$ and $b := \sum_{j=1}^n \lambda_{\max}^{(j)}$. The error can be estimated by

$$\|A^{-1} - B\|_2 \leq \varepsilon \left(\frac{1}{x}, [a, b], r \right) \|M^{-1}\|_2,$$

where $M = \bigotimes_{j=1}^n M^{(j)}$.

Theorem 5 shows how the inverse of matrices of the form (19) can be approximated by sums of matrix exponentials. It is based on the approximability of the function $1/x$ by sums of exponentials in the interval $[a, b]$. We refer to [14, 16] for details how to choose r and the coefficients $a_{v,[a,b]}, \alpha_{v,[a,b]}$ in order to reach a given error tolerance $\varepsilon \left(\frac{1}{x}, [a, b], r \right)$. Note that the interval $[a, b]$ where $1/x$ needs to be approximated depends on the matrices $A^{(j)}$ and $M^{(j)}$. Thus, if A changes a and b need to be recomputed which in turn has an influence on the optimal choice of the parameters $a_{v,[a,b]}$ and $\alpha_{v,[a,b]}$.

Numerical methods based on Theorem 5 can only be efficient if the occurring matrix exponential can be evaluated at low cost. In our setting we will need to compute the matrices $\exp(-\alpha_{v,[a,b]} M_1^{-1} A_1)$ and $\exp(-\alpha_{v,[a,b]} (M')^{-1} A')$. The evaluation of the first matrix will typically be simpler. In the case where a finite difference scheme is employed and

A_1 is a tridiagonal Toeplitz matrix while M_1 is the identity, the matrix exponential can be computed by diagonalizing A_1 , e.g., $A_1 = SD_1S^{-1}$, and using $\exp(-\alpha_{v,[a,b]}M_1^{-1}A_1) = S\exp(-\alpha_{v,[a,b]}D_1)S^{-1}$. The computation of exponentials for general matrices is more involved. We refer to [22] for an overview of different numerical methods. Here, we will make use of the Dunford–Cauchy integral (see [15]). For a matrix \tilde{M} we can write

$$\exp(-\tilde{M}) = \frac{1}{2\pi i} \oint_{\mathcal{C}} (\zeta I - \tilde{M})^{-1} e^{-\zeta} d\zeta$$

for a contour $\mathcal{C} = \partial D$ which encircles all eigenvalues of \tilde{M} . We assume here that \tilde{M} is positive definite. Then the spectrum of \tilde{M} satisfies $\sigma(\tilde{M}) \subset (0, \|\tilde{M}\|)$ and the following (infinite) parabola

$$\left\{ \zeta(s) = x(s) + iy(s) : x(s) := s^2, y(s) := -s \text{ for } s \in \mathbb{R} \right\}$$

can be used as integration curve \mathcal{C} . The substitution $\zeta \rightarrow s^2 - is$ then leads to

$$\exp(-\tilde{M}) = \int_{-\infty}^{\infty} \underbrace{\left(\frac{1}{2\pi i} (s^2 - is)I - \tilde{M} \right)^{-1}}_{=:G(s)} e^{-s^2+is} (2s - i) ds. \tag{20}$$

The integrand decays exponentially for $s \rightarrow \pm\infty$. Therefore (20) can be efficiently approximated by sinc quadrature, i.e.,

$$\exp(-\tilde{M}) = \int_{-\infty}^{\infty} G(s) ds \approx \mathfrak{h} \sum_{v=-N}^N G(v\mathfrak{h}), \tag{21}$$

where $\mathfrak{h} > 0$ and should be chosen s.t. $\mathfrak{h} = \mathcal{O}((N + 1)^{-2/3})$. We refer to [15] for an introduction to sinc quadrature and for error estimates for the approximation in (21). The parameters \mathfrak{h} and N in our implementation have been chosen such that quadrature errors become negligible compared to the overall discretisation error. For practical computations, the *halving rule* (see [15, §14.2.2.2]) could be faster while the Dunford–Schwartz representation with sinc quadrature is more suited for an error analysis.

Table 1 Relative $L^2(\Omega_{\ell}^{2D})$ -errors of the approximations $u_{2D,\ell}^{M_1}$ for different values of ℓ and f

	$f(x') = 1$	$f(x') = \sin(2x' + 0.5)$	$f(x') = \tanh(4x' + 1)$	$f(x') = x' $
$\ell = 1$	$3.30 \cdot 10^{-2}$	$2.44 \cdot 10^{-1}$	$2.48 \cdot 10^{-1}$	$1.15 \cdot 10^{-1}$
$\ell = 5$	$6.34 \cdot 10^{-3}$	$5.24 \cdot 10^{-2}$	$5.39 \cdot 10^{-2}$	$2.22 \cdot 10^{-2}$
$\ell = 10$	$4.24 \cdot 10^{-3}$	$3.53 \cdot 10^{-2}$	$3.63 \cdot 10^{-2}$	$1.51 \cdot 10^{-2}$
$\ell = 20$	$2.92 \cdot 10^{-3}$	$2.44 \cdot 10^{-2}$	$2.51 \cdot 10^{-2}$	$1.03 \cdot 10^{-2}$
$\ell = 50$	$1.82 \cdot 10^{-3}$	$1.52 \cdot 10^{-2}$	$1.56 \cdot 10^{-2}$	$6.45 \cdot 10^{-3}$

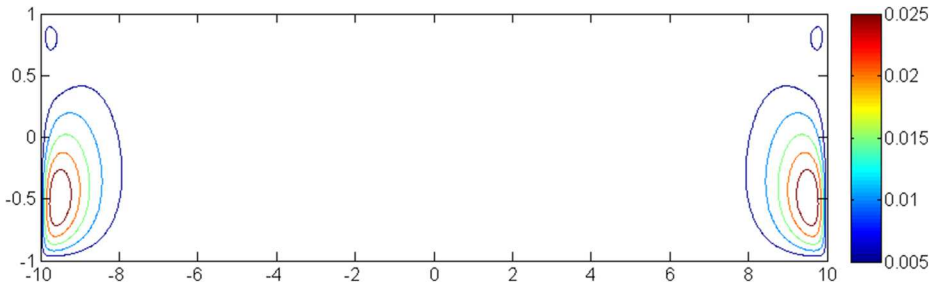


Fig. 2 Absolute error $|u_{2D,\ell}^{M_1} - u_{2D,\ell}^{ref}|$ for $\ell = 10$ and $f(x') = \tanh(4x' + 1)$

4 Numerical Experiments

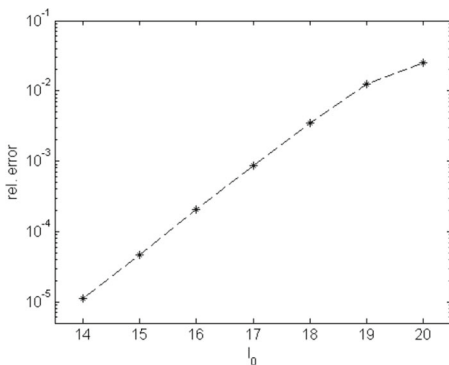
4.1 The Case of a Planar Cylinder

In this subsection we apply the methods derived in Section 3 to a simple model problem in two dimensions. We consider the planar cylinder

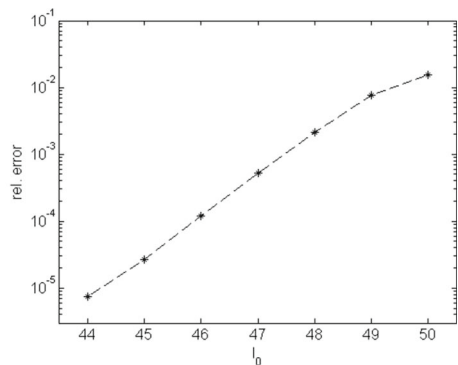
$$\Omega_\ell^{2D} = I_\ell \times (-1, 1)$$

and solve (1) for different right-hand sides $F = 1 \otimes f$ (see (2)) and different lengths ℓ . The reduced problem (3) on $\omega = (-1, 1)$ is solved using a standard finite difference scheme with a mesh width of $h = 10^{-2}$. We compare the approximations of (1) to a reference solution $u_{2D,\ell}^{ref}$ that is computed using a finite difference method on a sufficiently refined two-dimensional grid (mesh width $h = 10^{-2}$).

In Table 1 we state the $L^2(\Omega_\ell^{2D})$ -errors of the approximations $u_{2D,\ell}^{M_1}$ for various values of ℓ and right-hand sides f . Having in mind that $u_{2D,\ell}^{M_1}$ is a rather simple one-term approximation that only requires the solution of one $(d - 1)$ -dimensional problem (plus some postprocessing), the accuracy of the approximation is satisfactory especially for larger values of ℓ .



(a) $\ell = 20$



(b) $\ell = 50$

Fig. 3 Relative L^2 -errors of the approximation $u_{2D,\ell}^{M_1}$ in Ω_{l_0} for $f(x') = \tanh(4x' + 1)$

Table 2 Relative L^2 -errors of the approximations $u_{2D,\ell,m}^{M_2}$ for different values of ℓ and iterations m . We used $f(x') = \tanh(4x' + 1)$ throughout

	$\ell = 1$	$\ell = 5$	$\ell = 10$	$\ell = 20$	$\ell = 50$
$m = 1$	$2.48 \cdot 10^{-1}$	$5.39 \cdot 10^{-2}$	$3.63 \cdot 10^{-2}$	$2.51 \cdot 10^{-2}$	$1.57 \cdot 10^{-2}$
$m = 2$	$1.96 \cdot 10^{-2}$	$1.32 \cdot 10^{-2}$	$9.33 \cdot 10^{-3}$	$6.58 \cdot 10^{-3}$	$8.16 \cdot 10^{-3}$
$m = 3$	$7.44 \cdot 10^{-3}$	$2.66 \cdot 10^{-3}$	$1.85 \cdot 10^{-3}$	$1.29 \cdot 10^{-3}$	$8.16 \cdot 10^{-4}$
$m = 4$	$1.18 \cdot 10^{-3}$	$7.73 \cdot 10^{-4}$	$5.46 \cdot 10^{-4}$	$3.87 \cdot 10^{-4}$	$2.44 \cdot 10^{-4}$
$m = 5$	$3.80 \cdot 10^{-4}$	$3.74 \cdot 10^{-4}$	$2.71 \cdot 10^{-4}$	$1.96 \cdot 10^{-4}$	$1.25 \cdot 10^{-4}$
$m = 6$	$1.68 \cdot 10^{-4}$	$3.05 \cdot 10^{-4}$	$2.23 \cdot 10^{-4}$	$1.63 \cdot 10^{-4}$	$1.04 \cdot 10^{-4}$
$m = 7$	$1.33 \cdot 10^{-4}$	$2.90 \cdot 10^{-4}$	$2.12 \cdot 10^{-4}$	$1.55 \cdot 10^{-4}$	$9.95 \cdot 10^{-5}$

Figure 2 shows the pointwise, absolute error $|u_{2D,\ell}^{M_1} - u_{2D,\ell}^{ref}|$ in Ω_ℓ for $\ell = 10$ and $f(x') = \tanh(4x' + 1)$. As expected the accuracy of the approximation is very high in the interior of the planar cylinder (away from $\pm\ell$).

Lemma 2 (and Fig. 2) suggests that the approximation in the interior of the cylinder is significantly better than on the whole domain Ω_ℓ . Indeed, if the region of interest is only a subdomain $\Omega_{\ell_0} \subset \Omega_\ell$, where $\ell_0 < \ell$, the error decreases exponentially as $\ell_0 \rightarrow 0$. Figure 3 shows the relative error $\|u_{2D,\ell}^{M_1} - u_{2D,\ell}^{ref}\|_{L^2(\Omega_{\ell_0})} / \|u_{2D,\ell}^{ref}\|_{L^2(\Omega_{\ell_0})}$ with respect to ℓ_0 for $\ell = 20, 50$ and the right-hand side $f(x') = \tanh(4x' + 1)$. We can see that the exponential convergence sets in almost immediately as ℓ_0 moves away from ℓ .

To conclude, method 1 can be used in applications where

- only a limited approximation accuracy is required,
- a good starting point for more accurate methods is needed,
- the region of interest is a subdomain Ω_{ℓ_0} of Ω_ℓ with $\ell_0 < \ell$.

In method 2 we use $u_{2D,\ell}^{M_1}$ as starting value of the iteration which is then successively refined by approximating the residual in each step with a series of L^2 best approximations. In Table 2 we state the relative errors of this approach in the case $f(x') = \tanh(4x' + 1)$ for different values of ℓ and iteration steps. We can see that five iterations are sufficient to reduce the error of the initial approximation $u_{2D,\ell}^{M_1}$ by a factor 100 for all considered values of ℓ . However, in this case more iterations do not lead to significantly better results and the convergence seems to flatten. One explanation for this is that the residuals are increasingly difficult to approximate with each step of the iteration. After a few iterations a one-term

Table 3 Relative L^2 -errors of the approximations $u_{2D,\ell,r}^{M_3}$ for different values of ℓ and r . We used $f(x') = \tanh(4x' + 1)$ throughout

	$\ell = 1$	$\ell = 5$	$\ell = 10$	$\ell = 20$
$r = 1$	$1.48 \cdot 10^{-1}$	$1.02 \cdot 10^{-1}$	$9.76 \cdot 10^{-2}$	$9.60 \cdot 10^{-2}$
$r = 2$	$2.87 \cdot 10^{-2}$	$3.32 \cdot 10^{-2}$	$3.14 \cdot 10^{-2}$	$3.09 \cdot 10^{-3}$
$r = 3$	$9.06 \cdot 10^{-3}$	$9.78 \cdot 10^{-3}$	$9.88 \cdot 10^{-3}$	$9.79 \cdot 10^{-3}$
$r = 4$	$2.07 \cdot 10^{-3}$	$3.06 \cdot 10^{-3}$	$2.81 \cdot 10^{-3}$	$2.70 \cdot 10^{-3}$
$r = 5$	$1.24 \cdot 10^{-3}$	$1.11 \cdot 10^{-3}$	$1.16 \cdot 10^{-3}$	$1.16 \cdot 10^{-3}$

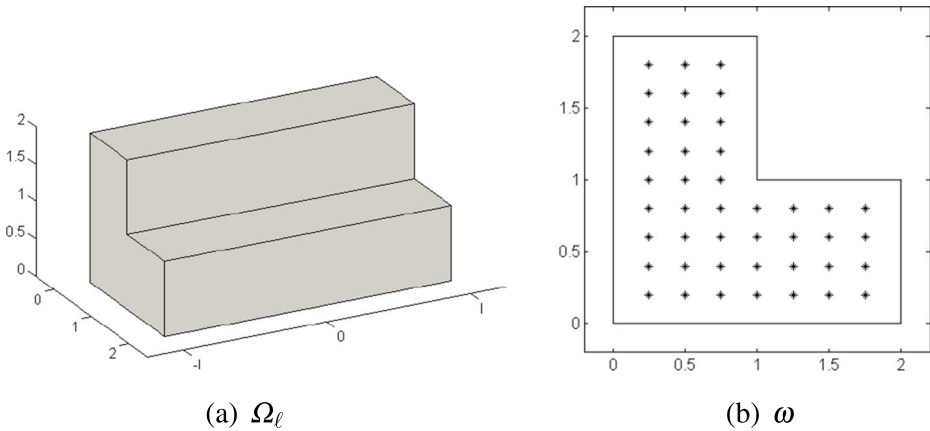


Fig. 4 Plot of domain Ω_ℓ and cross-section ω

approximation of these residuals of the form $p^{(m)} \otimes q^{(m)}$ therefore is not sufficiently accurate which leads to reduced decay of the error in the overall scheme.

Note that in the case $\ell = 1$, Ω_ℓ cannot be considered as a “long” domain. Therefore, the initial approximation $u_{2D,\ell}^{M_1}$ only exhibits a low accuracy. Nevertheless the error of $u_{2D,\ell,m}^{M_2}$ decays quickly as m increases and reaches a similar level of accuracy as for larger ℓ . This shows that method 2 can be used for more general domains than considered here (e.g. [13]).

In Table 3 we show the relative errors of the approximations $u_{2D,\ell,r}^{M_3}$ for $f(x') = \tanh(4x' + 1)$ and different values of ℓ and r . As the theory predicts the error decays exponentially in r and is governed by the approximability of the function $1/x$ by exponential sums. Note that in this two-dimensional example the arising matrix exponentials could be computed via diagonalization of the involved finite difference matrices. An approximation of the Dunford–Cauchy integral was not necessary in this case.

4.2 A Three-dimensional Domain with a Non-rectangular Cross Section

In this section we consider the three-dimensional domain

$$\Omega_\ell = (-\ell, \ell) \times \underbrace{[(0, 2) \times (0, 1) \cup (0, 1) \times (1, 2)]}_{=\omega},$$

where ω is an “L-shaped” domain (see Fig. 4). As before we solve problem (1) for different right-hand sides f and different values of ℓ . The reduced problem (3) on ω is solved using a

Table 4 Relative L^2 -errors of the approximations $u_{3D,\ell}^{M_1}$ for different values of ℓ and f

	$f(x') = 1$	$f(x') = \sin(x_2 + 0.5)x_3$	$f(x') = \tanh(x_2x_3)$
$\ell = 1$	$3.49 \cdot 10^{-2}$	$5.52 \cdot 10^{-2}$	$5.59 \cdot 10^{-2}$
$\ell = 5$	$1.12 \cdot 10^{-2}$	$1.77 \cdot 10^{-2}$	$1.79 \cdot 10^{-2}$
$\ell = 10$	$7.73 \cdot 10^{-3}$	$1.22 \cdot 10^{-2}$	$1.23 \cdot 10^{-2}$
$\ell = 20$	$5.37 \cdot 10^{-3}$	$8.51 \cdot 10^{-3}$	$8.86 \cdot 10^{-2}$
$\ell = 50$	$3.37 \cdot 10^{-3}$	$5.34 \cdot 10^{-3}$	$5.54 \cdot 10^{-2}$

Table 5 Relative L^2 -errors of the approximations $u_{3D,\ell,m}^{M_2}$ for different values of ℓ and iterations m . We used $f(x') = \tanh(x_2x_3)$ throughout

	$\ell = 1$	$\ell = 5$	$\ell = 10$	$\ell = 20$	$\ell = 50$
$m = 1$	$5.59 \cdot 10^{-2}$	$1.79 \cdot 10^{-2}$	$1.24 \cdot 10^{-2}$	$8.63 \cdot 10^{-3}$	$5.42 \cdot 10^{-3}$
$m = 2$	$9.82 \cdot 10^{-3}$	$4.38 \cdot 10^{-3}$	$3.08 \cdot 10^{-3}$	$2.18 \cdot 10^{-3}$	$1.37 \cdot 10^{-3}$
$m = 3$	$2.86 \cdot 10^{-3}$	$1.09 \cdot 10^{-3}$	$7.64 \cdot 10^{-4}$	$5.37 \cdot 10^{-4}$	$3.39 \cdot 10^{-4}$
$m = 4$	$8.03 \cdot 10^{-4}$	$3.37 \cdot 10^{-4}$	$2.37 \cdot 10^{-4}$	$1.63 \cdot 10^{-4}$	$1.03 \cdot 10^{-4}$
$m = 5$	$3.46 \cdot 10^{-4}$	$1.38 \cdot 10^{-4}$	$9.93 \cdot 10^{-5}$	$7.08 \cdot 10^{-5}$	$4.50 \cdot 10^{-5}$
$m = 6$	$2.73 \cdot 10^{-4}$	$1.03 \cdot 10^{-4}$	$7.50 \cdot 10^{-5}$	$5.37 \cdot 10^{-5}$	$3.42 \cdot 10^{-5}$
$m = 7$	$2.59 \cdot 10^{-4}$	$9.51 \cdot 10^{-5}$	$6.92 \cdot 10^{-5}$	$4.96 \cdot 10^{-5}$	$3.16 \cdot 10^{-5}$

standard 2D finite difference scheme. As 3D reference solution we use an accurate approximation using method 3, i.e., $u_{3D,\ell,r}^{M_3}$ for $r = 30$, which is known to converge exponentially in r . The reduced problem and the reference solution are computed with a mesh width of 0.02.

Table 4 shows the relative errors of the approximations $u_{3D,\ell}^{M_1}$ for different values of ℓ and right-hand sides f . As the theory predicts we cannot observe an exponentially decreasing error as ℓ gets large, since we measure the error on the whole domain Ω_ℓ and not only a subdomain $\Omega_{\ell-\delta_\ell}$. As before we only have to solve one two-dimensional problem on ω in order to obtain the approximation $u_{3D,\ell}^{M_1}$.

In Table 5 we show the relative errors of the approximations $u_{3D,\ell,m}^{M_2}$ for $f(x') = \tanh(x_2x_3)$ and different values of ℓ and m (number of iterations). As in the 2D case this method significantly improves the initial approximation $u_{3D,\ell,1}^{M_2} = u_{3D,\ell}^{M_1}$ using the alternating least squares type iteration. However, also here we observe that the convergence slows down when a certain accuracy is reached. We remark that a good starting point for the iteration is crucial for this method. In all our experiments $u_{3D,\ell}^{M_1}$ was a good choice which leads to a convergence behaviour similar to the ones in Table 5. Other choices often did not lead to satisfactory results.

In Table 6 we show the relative errors of the approximations $u_{3D,\ell,r}^{M_3}$ again for $f(x') = \tanh(x_2x_3)$ and different values of ℓ and r . As before the error decays exponentially with respect to r . The arising matrix exponentials $\exp(-\alpha_{v,[a,b]}A_{x'})$ in these experiments were computed using the sinc quadrature approximation (21). The number of quadrature points N was chosen such that the corresponding quadrature error had a negligible effect on the overall approximation.

Table 6 Relative L^2 -errors of the approximations $u_{3D,\ell,r}^{M_3}$ for different values of ℓ and r . We used $f(x') = \tanh(x_2x_3)$ throughout

	$\ell = 1$	$\ell = 5$	$\ell = 10$	$\ell = 20$
$r = 1$	$1.10 \cdot 10^{-1}$	$1.06 \cdot 10^{-1}$	$1.06 \cdot 10^{-1}$	$1.06 \cdot 10^{-1}$
$r = 2$	$2.18 \cdot 10^{-2}$	$1.94 \cdot 10^{-2}$	$1.92 \cdot 10^{-2}$	$1.92 \cdot 10^{-2}$
$r = 3$	$6.21 \cdot 10^{-3}$	$6.63 \cdot 10^{-3}$	$6.63 \cdot 10^{-3}$	$6.62 \cdot 10^{-3}$
$r = 4$	$2.36 \cdot 10^{-3}$	$2.02 \cdot 10^{-3}$	$2.02 \cdot 10^{-3}$	$2.01 \cdot 10^{-3}$
$r = 5$	$7.91 \cdot 10^{-4}$	$7.39 \cdot 10^{-4}$	$7.15 \cdot 10^{-4}$	$7.15 \cdot 10^{-4}$

5 Conclusion

We have compared three different methods for constructing tensor approximations to the solution of a Poisson equation on a long product domain for a right-hand side which is an elementary tensor.

The construction of a one-term tensor approximation is based on asymptotic analysis. The approximation converges exponentially (on a fixed subdomain) as the length of the cylinder goes to infinity. However, the error is fixed for fixed length since the approximation consists of only one term. The cost for computing this approximation is very low—it consists of solving a Poisson-type problem on the cross section and a cheap post-processing step to find the univariate function in the one-term tensor approximation.

The greedy-type method uses this elementary tensor and generates step-by-step a rank- k approximation. The computation of the m -th term in the tensor approximation itself requires an inner iteration. If one is interested in only a moderate accuracy (but improved accuracy compared to the initial approximation) this method is still relatively cheap and significantly improves the accuracy. However, the theory for this application is not fully developed and the definition of a good stopping criterion is based on heuristics and experiments.

Finally the approximation which is based on exponential sums is the method of choice among these three methods if a higher accuracy is required. A well developed a priori error analysis allows us to choose the tensor rank in the approximation in a very economic way. Since the method is converging exponentially with respect to the tensor rank, the method is also very efficient (but more expensive than the first two methods for the very first terms in the tensor representation). However, its implementation requires the realization of inverses of discretisation matrices in a sparse \mathcal{H} -matrix format and a contour quadrature approximation of the Cauchy–Dunford integral by sinc quadrature by using a non-trivial parametrisation of the contour.

We expect that these methods can be further developed and an error analysis which takes into account all error sources (contour quadrature, discretisation, iteration error, asymptotics with respect to the length of the cylinder, \mathcal{H} -matrix approximation) seems to be feasible. Also the methods are interesting in the context of a-posteriori error analysis to estimate the error due to the truncation of the tensor representation at a cost which is proportional to the solution of problems on the cross sections. We further expect that more general product domains of the form $\times_{m=1}^d \omega_m$ for some $\omega_m \in \mathbb{R}^{d_m}$ with dimensions $1 \leq d_m \leq d$ such that $\sum_{m=1}^d d_m = d$ and domains with outlets can be handled by our methods since also in this case zero-th order tensor approximation can be derived by asymptotic analysis (see [9]).

Acknowledgements This work was performed, in part, when the first author was visiting the USTC in Hefei and during his part time employment at the S. M. Nikolskii Mathematical Institute of RUDN University, 6 Miklukho-Maklay St, Moscow, 117198. The publication was supported by the Ministry of Education and Science of the Russian Federation.

Funding Open Access funding enabled and organized by Projekt DEAL.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.


References

1. Aghighi, M.S., Ammar, A., Metivier, C., Normandin, M., Chinesta, F.: Non-incremental transient solution of the Rayleigh–Bénard convection model by using the PGD. *J. Non-Newtonian Fluid Mech.* **200**, 65–78 (2013)
2. Ammar, A., Cueto, E., Chinesta, F.: Reduction of the chemical master equation for gene regulatory networks using proper generalized decompositions. *Int. J. Numer. Methods Biomed. Eng.* **28**, 960–973 (2012)
3. Ammar, A., Joyot, P.: The nanometric and micrometric scales of the structure and mechanics of materials revisited; an introduction to the challenges of fully deterministic numerical descriptions. *Int. J. Multiscale Comput. Eng.* **6**, 191–213 (2008)
4. Ammar, A., Mokdad, B., Chinesta, F., Keunings, R.: A new family of solvers for some classes of multidimensional partial differential equations encountered in kinetic theory modelling of complex fluids: Part II: Transient simulation using space-time separated representations. *J. Non-Newtonian Fluid Mech.* **144**, 98–121 (2007)
5. Chinesta, F., Ammar, A., Falco, A., Laso, M.: On the reduction of stochastic kinetic theory models of complex fluids. *Model. Simul. Mater. Sci. Eng.* **15**, 639 (2007)
6. Chinesta, F., Ammar, A., Leygue, A., Keunings, R.: An overview of the proper generalized decomposition with applications in computational rheology. *J. Non-Newtonian Fluid Mech.* **166**, 578–592 (2011)
7. Chinesta, F., Keunings, R., Leygue, A.: *The Proper Generalized Decomposition for Advanced Numerical Simulations: A Primer*. Springer, Cham (2014)
8. Chipot, M.: *Elliptic Equations: An Introductory Course*. Birkhäuser, Basel (2009)
9. Chipot, M.: *Asymptotic Issues for Some Partial Differential Equations*. Imperial College Press, New Jersey (2016)
10. Dolgov, S.V., Savostyanov, D.V.: Alternating minimal energy methods for linear systems in higher dimensions. *SIAM J. Sci. Comput.* **36**, A2248–A2271 (2014)
11. Espig, M., Hackbusch, W., Khachatryan, A.: On the convergence of alternating least squares optimisation in tensor format representations. arXiv:1506.00062 (2015)
12. Falcó, A., Nouy, A.: Proper generalized decomposition for nonlinear convex problems in tensor Banach spaces. *Numer. Math.* **121**, 503–530 (2012)
13. Giner, E., Bognet, B., Ródenas, J.J., Leygue, A., Fuenmayor, F.J., Chinesta, F.: The proper generalized decomposition (pgd) as a numerical procedure to solve 3d cracked plates in linear elastic fracture mechanics. *Int. J. Solids Struct.* **50**, 1710–1720 (2013)
14. Hackbusch, W.: *Tensor Spaces and Numerical Tensor Calculus*, 2nd edn. Springer, Cham (2019)
15. Hackbusch, W.: *Hierarchical Matrices: Algorithms and Analysis*. Springer Series in Computational Mathematics, vol. 49. Springer, Berlin (2015)
16. Hackbusch, W.: Computation of best L^∞ exponential sums for $1/x$ by Remez’ algorithm. *Comput. Vis. Sci.* **20**, 1–11 (2019)
17. Kazeev, V., Schwab, C.: Quantized tensor-structured finite elements for second-order elliptic PDEs in two dimensions. *Numer. Math.* **138**, 133–190 (2018)
18. Khoromskij, B.N.: $O(d \log N)$ -quantics approximation of N - d tensors in high-dimensional numerical modeling. *Constr. Approx.* **34**, 257–280 (2011)
19. Kressner, D., Sirković, P.: Truncated low-rank methods for solving general linear matrix equations. *Numer. Linear Algebra Appl.* **22**, 564–583 (2015)
20. Le Bris, C., Lelièvre, T., Maday, Y.: Results and questions on a nonlinear approximation approach for solving high-dimensional partial differential equations. *Constr. Approx.* **30**, 621–651 (2009)
21. Mokdad, B., Pruliere, E., Ammar, A., Chinesta, F.: On the simulation of kinetic theory models of complex fluids using the Fokker-Planck approach. *Appl. Rheol.* **17**, 26494-1–26494-14 (2007)
22. Moler, C., Van Loan, C.: Nineteen dubious ways to compute the exponential of a matrix, twenty-five years later. *SIAM Rev.* **45**, 3–49 (2003)
23. Oseledets, I.V.: Approximation of $2^d \times 2^d$ matrices using tensor decomposition. *SIAM J. Matrix Anal. Appl.* **31**, 2130–2145 (2010)
24. Oseledets, I.V., Dolgov, S.V.: Solution of linear systems and matrix inversion in the TT-format. *SIAM J. Sci. Comput.* **34**, A2718–A2739 (2012)
25. Oseledets, I.V., Rakhuba, M.V., Uschmajew, A.: Alternating least squares as moving subspace correction. *SIAM J. Numer. Anal.* **56**, 3459–3479 (2018)
26. Perotto, S., Ern, A., Veneziani, A.: Hierarchical local model reduction for elliptic problems: a domain decomposition approach. *Multiscale Model Simul.* **8**, 1102–1127 (2010)

27. Repin, S., Sauter, S.A.: Accuracy of Mathematical Models. European Mathematical Society, Berlin (2020)
28. Simoncini, V.: Computational methods for linear matrix equations. *SIAM Rev.* **58**, 377–441 (2016)
29. Uschmajew, A.: Local convergence of the alternating least squares algorithm for canonical tensor approximation. *SIAM J. Matrix Anal. Appl.* **33**, 639–652 (2012)
30. Uschmajew, A.: A new convergence proof for the higher-order power method and generalizations. *Pac. J. Optim.* **11**, 309–321 (2015)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Affiliations

Michel Chipot¹ · Wolfgang Hackbusch²  · Stefan Sauter¹ · Alexander Veit¹

Michel Chipot
m.m.chipot@math.uzh.ch

Stefan Sauter
stas@math.uzh.ch

Alexander Veit
alexander.veit@math.uzh.ch

¹ Institute for Mathematics, University of Zürich, Winterthurerstr. 190, CH-8057, Zürich, Switzerland

² Max Planck Institute for Mathematics in the Sciences, Inselstr. 22, D-04103, Leipzig, Germany