

# On the shape of the fundamental sloshing mode in axisymmetric containers

Tadeusz Kulczycki · Mateusz Kwaśnicki ·  
Bartłomiej Siudeja

the date of receipt and acceptance should be inserted later

**Abstract** In the paper we numerically study positions of high spots (extrema) of the fundamental sloshing mode of liquid in an axisymmetric tank. Our approach is based on a linear model reducing the problem to appropriate Steklov eigenvalue problem. We propose a numerical scheme for calculating sloshing modes and a novel method of making images of oscillating fluid.

## 1 Introduction

We consider the linearized sloshing problem in axisymmetric containers for perfect fluid. The aim of the paper is to numerically study the positions of the *high spots*, the extremal points of the fundamental sloshing mode  $\varphi$ . Their locations coincide with the extrema of the free surface elevation of liquid (see [Figure 1](#)), under the assumption that the liquid oscillates freely with a fundamental frequency according to  $\varphi$ . We work with a simplified linear model (see [section 2](#)), studied by many authors, including Faltinsen-Timokha [\[9, 10\]](#), Fox-Kuttler [\[13\]](#), Kozlov-Kuznetsov [\[19\]](#) and McIver [\[28\]](#). Linear sloshing in simple moving containers was also considered by Wu-Chen [\[33\]](#), Ardakani-Bridges [\[2\]](#) and Herczyński-Weidmann [\[16\]](#). More accurate physical models can also be considered, but nonlinear phenomena force a severely restricted

---

T. Kulczycki  
Institute of Mathematics and Computer Science, Wrocław University of Technology, Wybrzeże  
Wyspiańskiego 27, 50-370 Wrocław, Poland  
E-mail: Tadeusz.Kulczycki@pwr.wroc.pl

M. Kwaśnicki  
Institute of Mathematics and Computer Science, Wrocław University of Technology, Wybrzeże  
Wyspiańskiego 27, 50-370 Wrocław, Poland  
E-mail: Mateusz.Kwasnicki@pwr.wroc.pl

B. Siudeja  
Department of Mathematics, University of Oregon, Eugene, OR 97403, USA  
E-mail: siudeja@uoregon.edu

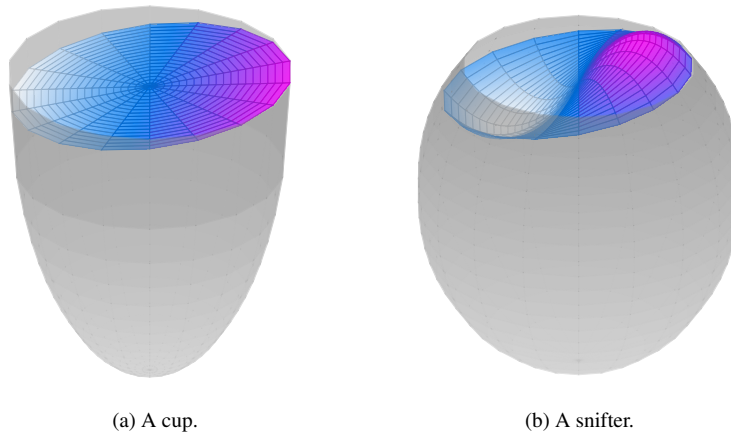


Fig. 1: High spots in a container with straight sides (a cup) compared to a bulbous one (a snifter).

choice of containers (see a review of the numerical methods by Rebouillat and Liksonov [31]). For general overview of both linear and nonlinear models of sloshing see monographs by Ibrahim [17], Kopachevsky-Krein [18], Lamb [25], Moiseev [30] or Troesch [32].

In a bulbous container high spots are located inside a container (see Figure 1b). On the other hand in a convex container, such that the angle between a wall of the container and a free surface is smaller or equal to  $\pi/2$ , high spots are located on the boundary of the container (see Figure 1a). This effect was proved rigorously by Kulczycki and Kwaśnicki in [23]. It has also been studied for spherical tanks by Faltinsen and Timokha [9], using analytic approximations.

In this paper we propose a new numerical scheme for calculating sloshing modes for solids of revolution. This scheme, applied to the fundamental mode, allows us to find positions of high spots with great accuracy. However, the same scheme can also be used to study higher sloshing modes. In section 4 we present numerical results for the first few modes of various containers, we inspect convergence of the numerical scheme and compare to known results. We would like to emphasize, that our numerical method matches analytic results of McIver [28] and Faltinsen-Timokha [9], which are designed specifically for spherical tanks. In contrast, our numerical approach can be applied to arbitrary solid of revolution.

We also describe an ingenious method of obtaining images of oscillating liquid showing positions of high spots. Our method is based on photographing a reflection of a dotted piece of paper on a very slightly disturbed surface of the liquid (see section 5). Note that this approach is designed for the small oscillations appropriate for the linear approximations, and it cannot be applied to turbulent, highly nonlinear sloshing. We experimented with 3 different containers and positions of the high

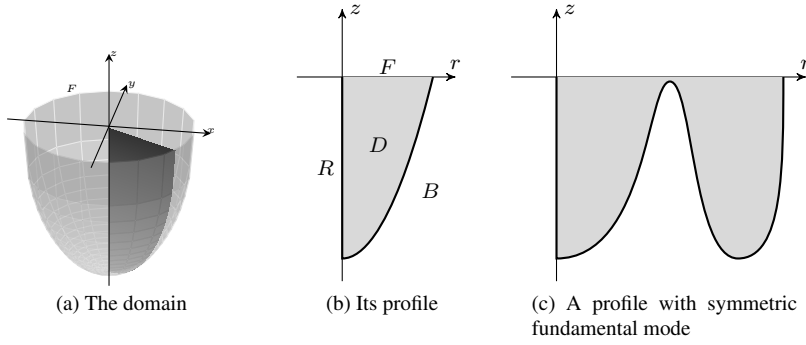


Fig. 2: Solid of revolution and the generating profiles.

spots agree with our numerical results and/or known results. Our experiment was first briefly described in the Steklov memoirs published in Notices of AMS [24].

## 2 Mathematical model

We assume that an inviscid, incompressible, heavy liquid occupies a three-dimensional container bounded from above by a free surface, which in its mean position is a simply connected two-dimensional domain of finite diameter. Let Cartesian coordinates  $(x, y, z)$  be chosen so that the mean free surface lies in the  $(x, y)$ -plane and the  $z$ -axis is directed upwards (see Figure 2a). The surface tension is neglected on the free surface, and we assume the liquid motion to be irrotational and of small amplitude. These assumptions allow us to linearise boundary conditions on the free surface and get the following boundary value problem for  $\varphi(x, y, z)$  (the velocity potential of the flow with a time-harmonic factor removed):

$$\Delta\varphi = 0 \text{ in } W, \quad (1)$$

$$\frac{\partial\varphi}{\partial z} = \nu\varphi \text{ on } F, \quad (2)$$

$$\frac{\partial\varphi}{\partial\mathbf{n}} = 0 \text{ on } B, \quad (3)$$

where  $W$  is the part of the container filled with liquid (in its mean position),  $B$  is the wetted part of its boundary, and  $F$  is the free surface of the liquid (see Figure 2a). The outer normal derivative on  $B$  is denoted by  $\frac{\partial}{\partial\mathbf{n}}$ .

The zero eigenvalue obviously exists for the problem (1–3), but we exclude it with the help of the following orthogonality condition:

$$\int_F \varphi = 0, \quad (4)$$

This condition also ensures that the volume of the fluid is preserved throughout the sloshing process.

Let us reiterate that we are working with a linear model. We are not concerned with the interactions between the fluid and the environment. As such, the model is well suited for infinitesimal sloshing of a perfect liquid (cf. [13, 19, 9, 10]). In other words, our approximations only apply to a nearly still water surface, as opposed to turbulent phenomena in strong sloshing (as in [31]).

For sufficiently regular domains it is known that the mixed Steklov problem (1–4) has a discrete sequence of eigenvalues

$$0 < \nu_1 \leq \nu_2 \leq \nu_3 \leq \dots \rightarrow \infty,$$

and the corresponding modes  $\varphi_n \in H^1(W)$ ,  $n = 1, 2, 3, \dots$ , restricted to the free surface  $F$ , form (together with a constant function eliminated by (4)) a complete orthogonal set in  $L^2(F)$ .

The following observation plays a key role in our study: if liquid oscillates freely with the fundamental frequency  $\nu_1$  then at every moment the free-surface elevation of liquid is proportional to the fundamental eigenfunction  $\varphi(x, y, 0)$  (see e.g. Lamb [25]).

Let us point out, that general containers are very hard to work with, even with the linearized model. Axisymmetric domains described below encompass many practical shapes and are somewhat easier to handle. Another simplification leading to manageable calculations involves two-dimensional containers, or long troughs with fluid sloshing only sideways. These were studied by Kulczycki-Kuznetsov [21, 22] and Faltinsen-Timokha [10], among others.

## 2.1 Axisymmetric containers

Now we turn to the problem of sloshing in axisymmetric containers. It is convenient to introduce the cylindrical coordinates  $(r, \theta, z)$  so that

$$x = r \cos \theta, \quad y = r \sin \theta.$$

and to take the  $z$ -axis as the axis of symmetry for  $W$ . In this case  $F$  is typically a disc on  $xy$ -plane (Figure 2). Moreover, we will consider  $W$  as obtained by rotation of a domain  $D$  adjacent to both axes in the  $rz$ -plane (Figure 2b). It is convenient to think of  $D$  as the cross-sections of  $W$  along the half-plane  $\theta = 0$ .

It is clear that the ansatz

$$\varphi = \psi_m(r, z) \cos(m\theta), \quad \text{or} \quad \varphi = \psi_m(r, z) \sin(m\theta), \quad m = 0, 1, 2, \dots, \quad (5)$$

where  $\psi$  is bounded near  $R = \{r = 0\} \cap D$ , reduces the eigenvalue problem (1–3) in  $W$  to the following sequence of boundary value problems:

$$(\psi_m)_{zz} + (\psi_m)_{rr} + \frac{1}{r}(\psi_m)_r - \frac{m^2}{r^2}\psi_m = 0, \quad \text{on } D \quad (6)$$

$$(\psi_m)_z = \lambda_m \psi_m, \quad \text{on } F \quad (7)$$

$$\frac{\partial \psi_m}{\partial \mathbf{n}} = 0, \quad \text{on } B \quad (8)$$



These relations must hold for all  $m$ , and by (4), for  $m = 0$  we also need

$$\int_F \psi(r, 0) r \, dr = 0. \quad (9)$$

The above reduction was applied by many authors (e.g. [23, 15, 27]). Note that there is no boundary condition on  $R$ , but if  $m > 0$  we need continuity of the three dimensional solution, while for  $m = 0$  we need continuous derivative (three dimensional solutions are harmonic, hence smooth). Therefore we have either Dirichlet ( $m > 0$ ) or Neumann ( $m = 0$ ) condition implied on  $R$ .

The variational methods (e.g. [8]) guarantee that for every  $m = 0, 1, 2, \dots$ , the spectral problem (6–9) has a sequence of eigenvalues

$$0 < \lambda_{m,1} \leq \lambda_{m,2} \leq \lambda_{m,3} \leq \dots, \quad m = 0, 1, 2, \dots, \quad (10)$$

and corresponding eigenfunctions  $\psi_{m,k}$ ,  $m \geq 0$ ,  $k \geq 1$ . Every eigenvalue in (10) has a finite multiplicity. Moreover, for every  $m \geq 0$  we have  $\lambda_{m,k} \rightarrow \infty$  as  $k \rightarrow \infty$ .

It is clear that, the sequence of eigenvalues  $\{\nu_n\}_{n=1}^{\infty}$  of problem (1)–(4) for  $W$  coincides with the double sequence  $\{\lambda_{m,k}\}_{m=0,k=1}^{\infty}$ , with every number  $\lambda_{m,k}$  repeated twice when  $m \geq 1$ . Thus the sequence of problems (6)–(9) is equivalent to the original sloshing problem.

### 2.1.1 Known rigorous results

Usually, the modes corresponding to  $\nu_1$  are antisymmetric (that is  $\nu_1 = \lambda_{1,1}$ ). Nevertheless, some rather strange containers (profiles from Figure 2c and Figure 4c) have rotationally symmetric fundamental eigenfunctions. For the latter we show numerically that  $\nu_1 \neq \lambda_{1,1}$ , and  $\nu_1 = \lambda_{0,1}$  instead.

Under the assumption  $\nu_1 = \lambda_{1,1}$ , there are two linearly independent antisymmetric fundamental sloshing modes corresponding to the least eigenvalue  $\nu_1$  ( $= \nu_2$ , has double multiplicity). In cylindrical coordinates they have the form (5) with  $m = 1$ .

By [23, Theorems 1.1 and 1.2], if  $W$  is a convex solid of revolution such that the angle between  $B$  and  $F$  is smaller or equal to  $\pi/2$ , then indeed  $\nu_1 = \nu_2 = \lambda_{1,1}$  correspond to antisymmetric eigenfunctions  $\varphi_1$  and  $\varphi_2$ , and the high spots of the modes described by  $\varphi_1$  and  $\varphi_2$  are located on the boundary, e.g. containers from Figure 1a and Figure 2. On the other hand, [23, Proposition 1.3] (see also [9]) asserts that if  $B$  and  $F$  form an obtuse angle then  $\varphi_1(x, y, 0)$ ,  $\varphi_2(x, y, 0)$  attain their maxima inside  $F$ , as shown on Figure 1b. We check our numerical methods described below against these theoretical results in section 4.

## 3 Numerical approximations

In this section we design a two dimensional numerical FEM scheme for calculating sloshing eigenvalues (and eigenfunctions) for solids of revolution described in section 2. Three dimensional FEM can also be used (see e.g. [24]), however one should be able to improve accuracy by exploiting decomposition (5) (see also Figure 2).

Unfortunately, this leads to a singular weak formulation, due to singular form of the gradient in polar coordinates. We introduce two modifications, leading to nonsingular schemes. The first is more natural, but leads to badly conditioned discrete problems for higher modes  $m$ , due to fast decaying weights near  $r = 0$ . The second modification is designed to keep a balance between singularity removal and decay of the weights. We use the second approach to recover, essentially exactly, the results for partially filled tanks obtained by McIver [28], for modes  $m = 0, 1, 2, 3$ . The same approach can be applied to arbitrary axisymmetric containers.

Our derivations are based on classical weak formulations of eigenvalue problems (see e.g. Babuška-Osborn [4] and Blanchard-Brüning [8]). One can also consult Bernardi, Dauge and Maday [6] for axisymmetric domains.

### 3.1 Modified decompositions

#### 3.1.1 Harmonic polynomials

We introduce the first modified, but equivalent decomposition by putting  $\psi_m(r, z) = r^m \phi_m(r, z)$ . The solutions of the original three dimensional problem take form (cf. (5))

$$\varphi_m = \phi_m(r, z)r^m \cos(m\theta), \quad \text{or} \quad \varphi_m = \phi_m(r, z)r^m \sin(m\theta). \quad (11)$$

Note that  $c_m = r^m \cos(m\theta)$  and  $s_m = r^m \sin(m\theta)$  are polar forms of the harmonic polynomials, e.g.  $c_1 = x$ ,  $c_2 = x^2 - y^2$ ,  $s_2 = 2xy$ . Indeed, if we interpret  $(x, y) = (r, \theta)$  as a complex number  $w = x + iy$ , then  $c_m = \Re w^m$  and  $s_m = \Im w^m$ .

Furthermore, note that  $\varphi_m$  is analytic inside  $W$  (it is in fact harmonic). In particular, the first  $m$   $r$ -derivatives must be continuous, forcing  $\varphi_m$  to decay at least as fast as  $r^m$ , hence our modified decomposition can be used to solve the sloshing problem. We will show that this new decomposition leads to a nonsingular weak formulation. Note that Neumann boundary condition on  $B$  must be imposed on  $r^m \phi_m$ . This is however irrelevant, since Neumann boundary condition is naturally incorporated into weak formulations.

Troesch [32] used a similar approach (family of harmonic polynomials, solid spherical harmonics) to find shapes of the domain on which polynomials are eigenfunctions for the sloshing problem. He found that the lowest  $m = 1$  mode for a cone with aperture angle  $\pi/2$  has eigenfunction with  $\psi_1 = r + rz/2$ . Our modification gives  $u = 1 + z/2$ , and even the simplest linear numerical approximation provides the exact solution. In fact for any  $m > 0$ , Troesch found a cone such that  $r^m(1 + cz)$  is the lowest mode.

#### 3.1.2 Linear approach

The above approach gives the best possible accuracy near  $r = 0$ , effectively removing the decay at  $r = 0$  from all eigenfunctions. However, in order to remove the singularity of the angular derivative, it is enough to introduce  $\psi_m(r, z) = ru_m(r, z)$  for  $m > 0$ . This means that low degree approximations (linear or quadratic) cannot

recover the exact decay around  $r = 0$  for  $m$  larger than 2. However, this limitation does not influence overall performance of the numerical method, especially that the high spots for the fundamental mode occur close to the boundary (far from  $r = 0$ ). All the numerical results described in [section 4](#) were obtained using this approach.

### 3.2 General weak formulations.

We will derive appropriate weak formulations based on weighted Sobolev spaces described in [\[20\]](#) and [\[6\]](#). Let

$$H_\alpha^1(D) = \{u : D \rightarrow \mathbb{R} : \int_D |\nabla u|^2 r^\alpha dr dz + \int_D u^2 r^\alpha dr dz < \infty\},$$

$$V_\alpha^1(D) = \{u : D \rightarrow \mathbb{R} : \int_D |\nabla u|^2 r^\alpha dr dz + \int_D u^2 r^{\alpha-2} dr dz < \infty\}.$$

The former is the classical weighted Sobolev space, while latter is more appropriate for  $m > 0$ , as we show below.

Start with the differential equation [\(1\)](#) written in a divergence form

$$L\psi := \nabla \cdot (r\nabla\psi) - \frac{m^2}{r}\psi = 0. \quad (12)$$

First we find the weak formulation for the classical decomposition [\(5\)](#) and the Steklov problem [\(6–9\)](#). We want to construct a symmetric bilinear form using  $L^2$  inner product of  $L\psi$  and smooth test functions. Then we can formally close the form in appropriate Sobolev space to get a valid weak formulation. For smooth functions, the equation  $\langle L\psi, v \rangle = 0$  is equivalent to

$$\begin{aligned} \int_D \frac{m^2}{r} \psi v dr dz &= \int_D [\nabla \cdot (r\nabla\psi)] v dr dz = - \int_D \nabla\psi \cdot \nabla v r dr dz + \int_{\partial D} r \frac{\partial\psi}{\partial n} v dS \\ &= - \int_D r \nabla\psi \cdot \nabla v dr dz + \int_F \lambda_m \psi v r dr \end{aligned} \quad (13)$$

In the last equality used the fact that the boundary integral equals 0 on  $B$  (due to Neumann BC for  $\psi$ ) and on  $R$  (since  $r = 0$ ). Note that we did not need to know the boundary condition for  $R$ , as long as  $\psi$  is bounded there. Note that the above weak formulation is consistent with the reduction of the integral  $\langle \Delta\phi, v \rangle$  from the three dimensional axisymmetric container to its two dimensional profile. In particular we get  $r dr$ , the polar Jacobian. Therefore, it is natural to consider the reduced problem on  $H_1^1(D)$ .

When  $m = 0$  (axisymmetric eigenfunctions), formulation [\(13\)](#) reduces to

$$\int_D \nabla\psi \cdot \nabla v r dr dy = \lambda_m \int_F \psi v r dr. \quad (14)$$

The left side naturally extends to  $H_1^1(D)$ , and the space of all axisymmetric functions  $H^1(W)$  (on the 3D container  $W$ ) is isomorphic to  $H_1^1(D)$  [\[6, Section II\]](#). Furthermore, traces of such functions on  $F$  are well defined, and belong to  $H_1^{1/2}(F)$ . We get a compact embedding of  $H_1^1(D)$  into  $L_1^2(F)$  and the standard spectral argument (e.g. [\[8, Section 6.3\]](#)) leads to discreteness of the spectrum of the quadratic form [\(14\)](#).

### 3.3 Non-singular weak formulations.

When  $m > 0$ , the weak formulation (13) contains singular  $L^2_{-1}$  inner product. Therefore it is natural to consider the space  $V_1^1(D)$  as its domain. Functions in this space have null trace on  $R$  (see [29]), and the space is isomorphic to  $H^1(W)$  with zero condition on the axis of symmetry [6, Section II]. This last condition exactly characterizes the sloshing modes  $m > 0$ . However, the presence of the singular  $L^2_{-1}$  norm hinders applicability of that weak formulation.

The first modified decomposition (11) solves this issue, by introducing  $\psi(r, \theta) = r^m \phi(r, \theta)$ , effectively eliminating the singularity. Replacing  $\psi$  with  $r^m \phi$  and test function  $v$  with  $r^m v$  (to keep symmetry of the formulation) leads to

$$\begin{aligned} \lambda_m \int_F \phi v r^{2m+1} dr &= \int_D \nabla(r^m \phi) \cdot \nabla(r^m v) r dr dy + m^2 \int_D \phi v r^{2m-1} dr dy \\ &= \int_D r^{2m-2} (m\phi + r\phi_r, r\phi_z) \cdot (mv + rv_r, rv_z) r dr dy + \int_D m^2 \phi v r^{2m-1} dr dy \\ &= \int_D \nabla \phi \cdot \nabla v r^{2m+1} dr + 2m^2 \int_D \phi v r^{2m-1} dr + m \int_D (v\phi_r + \phi v_r) r^{2m} dr. \end{aligned}$$

The last expression defines a symmetric quadratic form that can be extended to the Sobolev space  $V_{2m+1}^1(D)$ . Again, one gets appropriate compact embeddings into  $L^2_{2m+1}(F)$ , and

$$\begin{aligned} \lambda_m \int_F uv r^{2m+1} dr &= \int_D \nabla u \cdot \nabla v r^{2m+1} dr + 2m^2 \int_D uv r^{2m-1} dr \\ &\quad + m \int_D (vu_r + uv_r) r^{2m} dr. \end{aligned} \quad (15)$$

can be used to numerically approximate the eigenvalues and the eigenfunctions for  $m > 0$ .

Unfortunately, the presence of  $r^{2m+1}$  in the weak formulation requires high degree quadratures for accurate numerical integration near  $r = 0$  when  $m$  is large. Furthermore, fast decay of  $r^{2m+1}$  leads to discrete matrix eigenvalue problems with nearly singular matrices, for very fine meshes. This degrades the convergence of the numerical method, or even leads to numerically singular matrices.

These problems are mitigated by a linear approach  $\psi(r, \theta) = ru(r, \theta)$  (described in section 3.1.2). We get

$$\begin{aligned} \lambda_m \int_F uv r^3 dr &= \int_D \nabla(ru) \cdot \nabla(rv) r dr dy + m^2 \int_D uv r dr dy \\ &= \int_D (u + ru_r, ru_z) \cdot (v + rv_r, rv_z) r dr dy + \int_D m^2 \phi v r dr dy \\ &= \int_D \nabla u \cdot \nabla v r^3 dr + (m^2 + 1) \int_D uv r dr + \int_D (vu_r + uv_r) r^2 dr. \end{aligned} \quad (16)$$

Note that when  $m = 1$  we recover the first modification, while for  $m > 1$  we still use the Sobolev space  $V_3^1(D)$ , hence avoiding high powers of  $r$ , high degree quadratures in numerical integration, and nearly singular matrices in the discrete approximation.

### 3.4 Numerical method

The simplified weak formulations (15) and (16) allow us to use commonly available numerical method suitable for Laplace eigenvalue problems. However, we only present the latter approach due to numerical instability of (15) we experienced for  $m > 1$ . Note that formulations (15) and (16) are the same for  $m = 1$ , usually containing the fundamental mode.

We triangulate the domain and use conforming  $P1$  (first-order) or  $P2$  (second-order) finite elements to discretize the weak formulation (16). Resulting piecewise linear (or quadratic) functions give approximate shapes of the eigenfunctions, in particular the sloshing profiles on  $F$ , showing high-spots.

We also design an adaptive mesh refining strategy to decrease errors in approximations, and fit the possibly curved boundary. We use classical local residual error a posteriori estimators (see e.g. Ainsworth and Oden [1]), that were recently studied for pure Steklov problems by Armentano-Padra [3], Garau-Morin [14]. We modify their approach to fit the weighted weak formulation and variable boundary conditions. In each step we choose a small fraction of triangles to be refined, based on the largest local error estimators [14, Definitions 2.5 and 2.7]). Even though we are using (16) to find approximation  $\tilde{u}$ , we evaluate the error estimators on  $\tilde{\psi} = r\tilde{u}$  and the original operator  $L$  given by (12). More precisely, for any triangle  $T$  and its edge  $E$  in the triangulation of the domain we find

$$R(T) = \int_T \left( \nabla \cdot (r \nabla \tilde{\psi}) - \frac{m^2}{r} \tilde{\psi} \right)^2 dr dz, \quad (\text{element residual}) \quad (17)$$

$$S(T, E) = \int_E \left( \tilde{\psi}_z - \lambda \tilde{\psi} \right)^2 r dr \quad \text{if } E \subset F, \quad (\text{boundary residual}) \quad (18)$$

$$J(T, E) = \int_E \left( \frac{\partial_n \tilde{\psi} + \partial_{\bar{n}} \tilde{\psi}}{2} \right)^2 r ds \quad \text{if } E \subset D, \quad (\text{jump residual}) \quad (19)$$

where  $\partial_n$  is the inner, and  $\partial_{\bar{n}}$  is the outer normal derivative for the edge  $E$ . Note that  $\partial_{\bar{n}}$  is evaluated outside of  $T$ , on an adjacent piece of the triangulation. The element residual measures deviation of the approximation from harmonicity. The jump residual is responsible for the discontinuity of the derivative of the approximation across triangles. Finally, the boundary residual measures deviation from the Steklov eigenvalue condition on  $F$ .

We add all errors corresponding to the same triangle and refine the least accurate 10% of the triangles. Finally, we add all errors and take square root to find a global residual norm, used in the numerical results section 4 to control convergence. It turns out that this norm correlates almost exactly with the eigenvalue error in known cases. In cases where the eigenvalues are not known, the plots of the squared norm and the differences to the best approximation are also nearly identical. This indicates that we have a reliable error estimator.

In order to get more than just the first eigenvalue, we considered the following strategies. Execute the adaptive loop just for the first eigenvalue and find all eigenvalues based on fully refined mesh. This leads to rather poor approximations for

higher eigenvalues. It is better to add residual errors from all eigenfunctions and refine worst triangles based on the accumulated estimators. This gives much better results for higher eigenvalues, but slightly degrades the accuracy of the lowest eigenvalue estimation. Finally, we tried to run the adaptive algorithm for each eigenvalue separately. This is obviously optimal, but one must first know fair approximations for these eigenvalues in order to target the specific one. In practice, the optimal separate calculations must be preceded by a collective adaptive calculation yielding good starting points for targeted searches.

### 3.5 Implementation

We used FEniCS [26] to implement the above numerical scheme. It provides intuitive mathematical interface for defining weak forms and solving differential equations using Finite Element Methods. Adaptive methods are also easy to implement, including residual errors and snapping mesh boundary to a given theoretical domain.

#### 3.5.1 Eigenvalue problem

Assuming that variable *mesh* contains an already created triangulation, we define appropriate function space (quadratic Continuous-Galerkin method), test and trial functions

```
V = FunctionSpace(mesh, "CG", 2) # second order method
u = TrialFunction(V)
v = TestFunction(V)
```

We do not need to specify any boundary conditions, but we need to create subdomain  $F$ , since we integrate over it later. We assume that  $F \subset \{z = 0\}$  and the mesh is contained in  $\{z \leq 0\}$ . We can use variable  $s$  to limit  $F$  to  $r \leq s$ . We use this facility to find the eigenvalues of the ice-fishing problem [19].

```
# x[0]=z, x[1]=r
s = 1 # F contained r<=s
class Steklov(SubDomain): # subdomain F
    def __init__(self, s = 1):
        self.size = s
        super(Steklov, self).__init__()
    def inside(self, x, on_boundary):
        return on_boundary and near(x[1], 0) and x[0] <= self.size
steklov = Steklov(s)
boundary = MeshFunction("size_t", mesh, 1) # 1D elements
boundary.set_all(0)
steklov.mark(boundary, 1) # mark F for later integration
```

Next we create both sides of the weak formulation (16)

```
ds = Measure("ds")[boundary]
r = Expression('x[0]', cell=triangle, degree=1)
# right side
a = ( r**3*inner(grad(u), grad(v)) + r**2*(u*Dx(v, 0)+v*Dx(u, 0))
      + Constant(m**2+1)*r*u*v ) * dx
# left side
b = r**3*u*v*ds(1) # integration over surface (F marked with 1)
A = PETScMatrix()
B = PETScMatrix()
assemble(a, tensor=A)
assemble(b, tensor=B)
```

Finally, we configure and run the eigenvalue solver

```
eigsolver = SLEPcEigenSolver(A,B)
eigsolver.parameters["spectral_transform"] = "shift-and-invert"
eigsolver.parameters["problem_type"] = "gen_hermitian"
eigsolver.parameters["spectrum"] = "smallest real"
eigsolver.parameters["spectral_shift"] = 1E-10
eigsolver.solve(3) # find 3 eigenvalues
eig = eigsolver.get_eigenpair(0) # get the first
```

The variable *eig* contains a tuple of four elements:  $\Re\lambda$ ,  $\Im\lambda$ ,  $\Re f$ ,  $\Im f$ . Imaginary parts are obviously zero in our case, while  $\Re f$  can be turned into the eigenfunction, but it needs to be multiplied by  $r$  (if  $m > 0$ )

```
u = Function(V)
u.vector()[:] = eig[2]
u = project(u*r, V)
```

Now function  $u$  can be evaluated at any point in the domain using standard Python evaluation, e.g.  $u(r, 0)$  will give a value at point  $(r, 0)$  in  $F$ . We used this to generate plots of the expected heights of the sloshing fluid.

Note that one can find an eigenvalue near a target number  $T$  by changing two solver parameters

```
eigsolver.parameters["spectrum"] = "target real"
eigsolver.parameters["spectral_shift"] = T
```

This allows us to optimize the adaptive loop described below for a specific eigenvalue.

### 3.5.2 Adaptive mesh refinement

We choose a small fraction of cells, e.g. 10% to be refined. The refinement procedure is built into FEniCS, we only need to find the local errors. To achieve this we find indicators that are constant on triangles and equals to the local error for each cell. Then largest 10% gets marked and refined.

We declare the space of discontinuous, piecewise constant functions, a test function and few helper functions.

```
C = FunctionSpace(mesh, "DG", 0) # piecewise constant function
w = TestFunction(C)
h = CellSize(mesh)
n = FacetNormal(mesh) # normal vectors to edges
r = Expression("x[0]", cell=triangle, degree=1)
ds = Measure("ds")(boundary)
```

Here *boundary* is the object marked for integration over  $F$  in the eigenvalue solving code. Assuming  $u$  and  $e$  are the just found eigenfunction and eigenvalue we calculate local residual error indicators (17–19)

```
# three kinds of error
# dS inner edges, ds(1) Steklov part
errform = ( h**2*(div(r*grad(u))-Constant(m**2)*u)**2*w*dx
            + h*(Dx(u,1)-Constant(e)*u)**2*r*w*ds(1)
            + avg(h)*jump(grad(u),n)**2*avg(w)*r*dS )
indicators = assemble(errform) # errors for each cell
```

Finally we find the cut-off value of the error for the 10% of worst cells, mark those cells and refine the mesh.

```
fraction=0.1
cutoff = sorted(indicators, reverse=True)[int(len(indicators)*fraction)-1]
marker = CellFunction("bool", mesh)
marker.array()[:] = indicators>cutoff # mark worst errors
mesh = refine(mesh, marker)
```

We can perform the refinement based on a single eigenfunction, or add indicators for all eigenfunctions to get the collective refinement. We use both methods in [section 4](#). The collective method is faster, but less accurate. While the separate refinements give better results, but we still need to perform a collective search to determine the targets for separate searches.

### 3.5.3 Perimeter refinement and mesh snapping

Assuming that the left boundary  $R$  is inside  $\{r = 0\}$ , we can select all boundary cells except those touching  $\{r = 0\}$  and refine them. This increases number of mesh vertices on  $B$  (potentially curved boundary) as well as on the Steklov part. The refinement on the Steklov part  $F$  of the boundary is beneficial, since we need to integrate over  $F$ . While the refinement on curved part  $B$  allows us to snap new boundary vertices to the theoretical boundary  $B$ , getting better shape approximation.

First we extract all triangles with an edge on the boundary.

```
perimeter = [c for c in cells(mesh) if any([f.exterior() for f in facets(c)])]
perimeter = [c for c in perimeter if min([v.x(0) for v in vertices(c)])>0]
```

Next we find 25% triangles with largest diameter and refine those.

```
marker = CellFunction('bool', mesh, False)
diameters = [c.diameter() for c in perimeter]
cutoff = sorted(diameters, reverse=True)[int(len(diameters)*0.25)-1]
for c in perimeter:
    marker[c] = c.diameter()>cutoff
mesh = refine(mesh, marker)
```

Whenever we have new boundary vertices, either via adaptive eigenvalue refinement, or perimeter refinement, we can use boundary snapping to achieve better boundary approximation on  $B$ . The appropriate algorithms are already available in FEniCS, we only need to specify the function describing curved boundary  $B$ . Assuming  $r = f(z)$  describes  $B$ , we define

```
class Curved(SubDomain):
    def snap(self, x):
        # snap only on B, not on R or F
        if not near(x[0]*x[1], 0):
            x[0]=f(x[1])
```

Using this class we can snap all boundary vertices (not satisfying  $r = 0$  or  $z = 0$ ) to the prescribed curve, by simply redefining their  $r$  coordinates using  $f(z)$ . And we apply the snapping function to existing *mesh*.

```
curve = Curved()
mesh.snap_boundary(curve)
```

Before we start solving for eigenvalues on a curved domain (e.g. spherical containers), we perform a few loops containing perimeter refinement and boundary snapping, to improve the initial shape of the domain. Later, perimeter refinement and boundary snapping are also incorporated into each adaptive step.

### 3.5.4 Mesh generation

FEniCS has many ready to use meshes to choose from, but it is also possible to build a mesh from vertices of a polygon, as long as those vertices are given in a counter-clockwise order. This can be used to build a mesh with arbitrary “curved” boundary



	exact	P2 separate	P2 collective	P1 separate	P1 collective
$m = 0$	3.8281081396	3.8281081405	3.8281081418	3.8281387227	3.8281483323
	7.0155753606	7.0155753657	7.0155753669	7.0156844394	7.0156809670
	10.173468105	10.173468124	10.173468126	10.173695372	10.173706920
$m = 1$	1.7507975745	1.7507975746	1.7507975746	1.7507989922	1.7508023457
	5.3311932955	5.3311932970	5.3311932972	5.3312426129	5.3312472275
	8.5363157091	8.5363157164	8.5363157170	8.5364660207	8.5364704653
$m = 2$	3.0406821799	3.0406821800	3.0406821802	3.0406864963	3.0406919358
	6.7061131204	6.7061131223	6.7061131229	6.7061691529	6.7061730312
	9.9694677794	9.9694677892	9.9694677900	9.9696239668	9.9696306520

Table 1: Numerical approximations of the three smallest eigenvalues for modes  $m = 0, 1, 2$ , for a cylinder with radius 1 and height 1 compared to true eigenvalues, using either linear  $P1$  (400k triangles) or quadratic  $P2$  elements (100k triangles). Finally we use a collective adaptive loop, or each eigenvalue separately.

$B$ . Say  $B$  is described by  $r = f(z)$ . Then we can take a mesh with vertices  $(0, 0)$ ,  $(0, -d)$  (where  $d$  is the depth of the tank), and a number  $n$  of vertices of the form  $(f(dk/n), -dk/n)$ , with  $k = n, \dots, 0$ . One can choose a any parametrization, but this one is consistent with the boundary snapping described in the previous section. Having all the points, we create the mesh.

```
# points = list of tuples (points)
# size = initial mesh size parameter
polygon = Polygon([Point(*p) for p in points])
mesh = Mesh(polygon, size)
```

We can also add a few extra points on the top of the profile  $\{z = 0\}$  in order to ensure there are enough boundary intervals in  $F$  to perform integration over  $F$ . This is however not strictly necessary, as the mesh created by FEniCS is well balanced, and further refinements will be applied to that part of the boundary.

## 4 Numerical results

In this section we apply our numerical scheme to various shapes, including spherical and cylindrical tanks. We chose to limit the calculations to around 100 000 triangles with the quadratic method, and 400 000 triangles with the linear method (about 0.5GB memory usage for both methods).

We recover the exact solutions for cylindrical tanks with great accuracy. We also get nearly the same results as McIver [28] in the numerically challenging spherical case (curved boundary). Finally we study quite general domains end use residual norms to check stability and convergence of our approximations.

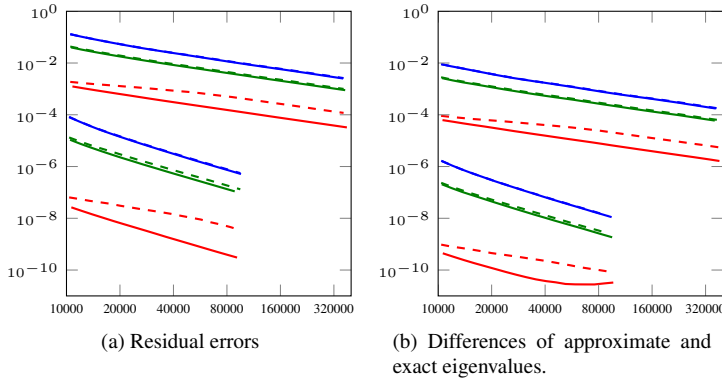


Fig. 3: Convergence of the adaptive method for  $m = 1$ . Dashed lines indicate one adaptive loop for all eigenvalues, while solid lines show results for each eigenvalue separately optimized. Longer lines show the first order, piecewise linear approximations, shorter the quadratic method. Finally red, green and blue stand for the first, second and third eigenvalues.

#### 4.1 Cylindrical tanks - comparisons to exact solutions.

The sloshing eigenvalues for the cylindrical tank (coffee cup) of radius 1 and depth  $d$  can be calculated using

$$\lambda_{m,k} = j'_{m,k} \tanh(d j'_{m,k}),$$

where  $j'_{m,k}$  is the  $k$ -th zero of the derivative of the Bessel  $J_m$  function. The exact formulas allow us to test our numerical method.

We use a square mesh, with side 1 to get the results from Table 1. The solid of revolution obtained from this profile is a coffee cup with radius 1 and height 1. In each approximation we either apply the adaptive procedure to all eigenvalues together, or each one separately. Table 1 shows almost no difference these strategies for very fine meshes, but the individual approach gives better results for the lowest eigenvalues, as evidenced by Figure 3. The slopes of the lines on that figure clearly differentiate the orders of the used methods, while being virtually the same in the scope of each method. We get consistent rates of convergence, obviously faster for the quadratic method. Also, note that the quadratic method seems to reach the limits of the floating point arithmetic for the lowest eigenvalue. Finally, squared residual error norm obtained from (17–19) correlates nearly perfectly with the actual approximation error (even the scales on the plots are identical).

Let us also point out that one could simply refine the mesh uniformly to around 100 000 triangles for quadratic method, without using any adaptive strategy. The results one would get are the same as starting with 5000 triangles and performing a few adaptive steps, until just 10 000 triangles are reached. Therefore adaptive approach saves a lot of memory, allowing for a more accurate results. Also, the advantage of the

	McIver	$\varepsilon = 1/64$	$\varepsilon = 1/16$	$\varepsilon = 1/4$
$m = 1$	2.75475	2.754750	2.754465	2.736291
	5.89215	5.892146	5.892041	5.885388
	9.03285	9.032852	9.032790	9.028844
	12.1741	12.17412	12.17407	12.17130

Table 2: Quadratic approximations of the eigenvalues of the approximate ice-fishing problem: cylinder with height and radius one, and small hole of radius  $\varepsilon$ . Note that for modes  $m = 0$  and  $m = 2$  we obtained the same values as McIver (even for  $\varepsilon = 1/16$ ) hence we do not list the results.

adaptive method should increase with complexity of the domain (e.g. corners). The adaptive approach is even more important for domains that cannot be exactly triangulated, e.g. spherical tanks (see [subsection 4.3](#)), due to the necessity of the boundary snapping algorithm (improving perimeter approximation).

It is clear that the linear method performs poorly compared to the quadratic. The latter gives nearly exact answers for the lowest eigenvalues of each mode, and only slightly worse approximations for higher eigenvalues. However, one should expect great performance in this case, due to a very basic shape of the container. In the next section we study convergence for more irregular shapes, with no known eigenvalues. There, we will use residual errors to check convergence.

#### 4.1.1 Ice-fishing problem.

Before we proceed, we can use the same mesh to find a numerical approximation of the lowest eigenvalue for the ice-fishing problem. Formally, we should consider an infinitely wide and deep ocean covered with ice, with a small round fishing hole. Sloshing in such container was studied by McIver [28] and Kozlov-Kuznetsov [19].

We will approximate this infinite case using a cylindrical tank with small round hole on top. The only difference from the cylindrical case discussed above is that we need to impose the Steklov condition on a part of the top boundary. More precisely we can take  $F = \{(r, z) : z = 0 \text{ and } r < \varepsilon\}$  to get a good approximation of the ice-fishing problem. Now we have a flat top part of the boundary with changing boundary condition, leading to a slower convergence. Nevertheless, for a fixed  $\varepsilon$  the approximations stabilize rather quickly, and the domain approximation seems the main source of errors (see [Table 2](#) for a comparison with the results due to McIver [28]). Interestingly, only the  $m = 1$  mode seems hard to estimate, with  $m = 0$  and  $m = 2$  very quickly approaching McIver's values, even for relatively large  $\varepsilon$ .

The profile of the fundamental mode of the sloshing liquid is shown on [Figure 11](#), together with the profiles for spherical tanks filled to various levels. We chose to postpone the plot, since McIver [28] treats the ice-fishing problem as a limiting case of almost full spherical tanks.

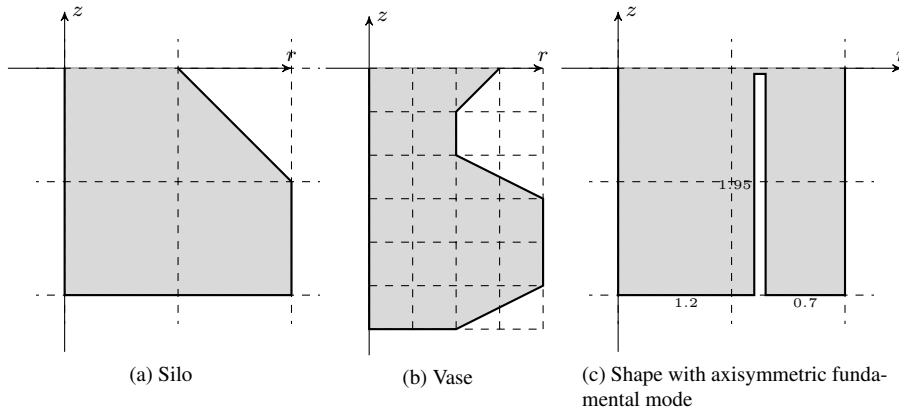


Fig. 4: Profiles of the tanks.

#### 4.2 Other polygonal shapes, and adaptive loop convergence.

For further testing, we choose a few polygonal shapes from Figure 4, so that the boundary can be exactly represented using triangulations. Exact eigenvalues are not known, but we can still check stability and convergence of the residual errors. We will use only the optimal, separate adaptive loops for each eigenvalue.

##### 4.2.1 The silo and the vase.

Figure 5 shows the residual errors and the differences to the best found approximation for adaptive loops on the silo (Figure 4a) and the vase (Figure 4b). The scales on the plots are identical to the cylindrical data from Figure 3. The lines for the silo are nearly identical to the ones for the cylindrical tank, with generally slightly higher errors. The vase also shows similar performance, but with the lowest eigenvalue showing higher initial residual errors. This seems to indicate that the reentrant corners (over 180 degrees) inside of the vase are a significant challenge for the numerical method. This is not unexpected, since smoothness of the theoretical solutions of any Laplace-type eigenvalue problem deteriorates significantly near reentrant corners. See Fox-Henrici-Moler [12] and Betcke-Trefethen [7] for the significance of the reentrant corners and a different numerical method of handling these corners.

Note that the plots of the differences are almost straight, as expected. Except for the ends of the lines which show unwarranted improved convergence. The difference to the best approximation is nearly the same as the difference to the exact value when the errors are large. But once we reach the limits of the approximation scheme, we can see false convergence to overestimated best approximation. This problem does not apply to the residual errors, which exhibit consistent decay rates throughout the whole adaptive loop.

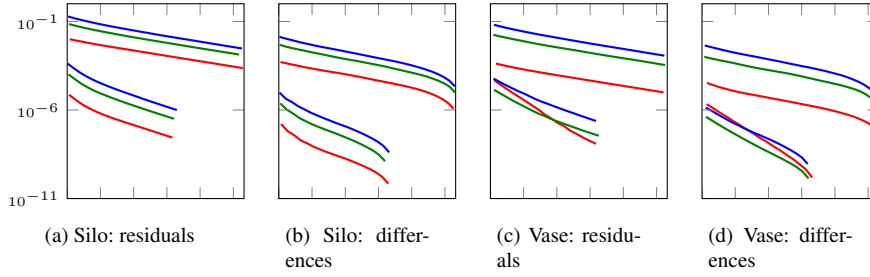


Fig. 5: Convergence for the three lowest eigenvalues of the mode  $m = 1$  (linear and quadratic methods). Axis ranges for all plots are the same as for cylindrical tanks, Figure 3.

	linear P1	quadratic P2
$m = 0$	1.046054, 6.366773	1.046038, 6.366661
$m = 1$	1.238800, 3.782383	1.238794, 3.782350

Table 3: Approximations of the eigenvalues of the shape from Figure 6c. The smallest eigenvalue belongs to  $m = 0$ !

#### 4.2.2 A shape with the axisymmetric fundamental sloshing mode.

The last shape (from Figure 4c) is the most interesting, both numerically and theoretically. Due to very narrow passage and reentrant corners we might expect slower convergence (compared to a cylindrical tank). Instead, Figure 6a and Figure 6b show unchanged convergence rates, but with much higher initial errors (especially in the quadratic method). Notice also that the errors for the quadratic method are almost independent of the eigenvalue. This shows that the adaptive loop tries to refine the narrow passage (and reentrant corners) before tackling the Steklov boundary  $F$ . We still see that the residual error estimators correlate almost perfectly with the differences.

The profiles of the sloshing fluid for the last shape (Figure 6c) are the most interesting aspect of the tank. The lowest eigenvalue belongs to the mode  $m = 0$ , not to  $m = 1$ , as was the case in all other shapes (see Table 3). Therefore, we found an example for which  $\nu_1 \neq \lambda_{1,1}$  and the results of [23] do not apply (see section 2.1.1). However, our example is rather strange, and almost all practical shapes should satisfy the conditions imposed in [23].

We see that the eigenfunction for the lowest eigenvalue attains its maximum at the center of the tank, and the global high spot is almost twice the height of the local boundary maximum.

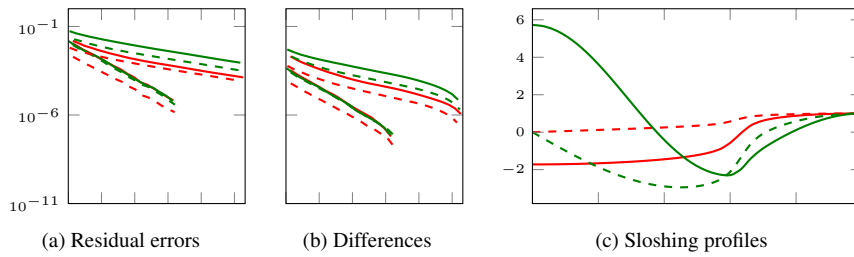


Fig. 6: The two lowest eigenvalues of the modes  $m = 0$  (solid lines) and  $m = 1$  (dashed lines) found using linear and quadratic approximations, with separate adaptive loops. Note that the lowest eigenvalue belongs to  $m = 0$ , solid red line on the rightmost plot.

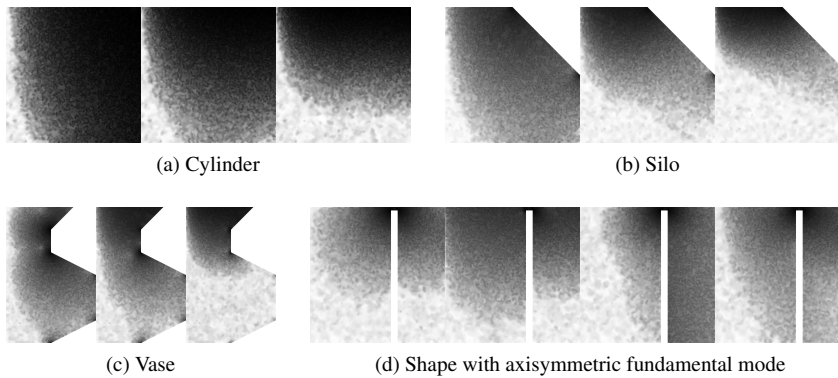


Fig. 7: Sizes of the mesh triangles (darker color indicates smaller triangles). Adaptive loop refines mostly triangles near any corner and near  $F$ . For each domain we show the adaptively refined mesh for the smallest three eigenvalues of the mode  $m = 1$  (except for the last shape with two lowest eigenvalues for  $m = 0, 1$ ).

#### 4.2.3 Sizes of triangles created by the adaptive loop.

We end our investigations with triangle density plots, [Figure 7](#). The adaptive loop creates highly nonuniform meshes, with many small triangles in most important parts of the domain. It is clear that the free surface  $F$  is the key to good approximation. However, any corners in the boundary of the domain are also heavily refined by the adaptive loop. This is especially noticeable on the last two shapes ([Figure 7c](#) and [Figure 7d](#)).

Note also, that the higher the eigenvalue, the higher the concentration of triangles near the free surface  $F$ . This can be observed on any of the subplots of [Figure 7](#). As

	$d = 0.2$	$d = 0.6$	$d = 1.0$	$d = 1.4$	$d = 1.8$
$m = 0$	3.8261 <sub>3</sub> <sup>18</sup>	3.6501 <sub>4</sub> <sup>44</sup>	3.7451 <sub>7</sub> <sup>68</sup>	4.3010 <sub>2</sub> <sup>22</sup>	6.7641 <sub>8</sub> <sup>80</sup>
	9.2561 <sub>3</sub> <sup>27</sup>	7.2659 <sub>6</sub> <sup>63</sup>	6.9763 <sub>6</sub> <sup>60</sup>	7.8005 <sub>5</sub> <sup>48</sup>	12.113 <sub>9</sub> <sup>930</sup>
	14.7556 <sub>1</sub> <sup>07</sup>	10.7449 <sub>7</sub> <sup>76</sup>	10.1474 <sub>8</sub> <sup>73</sup>	11.2558 <sub>7</sub> <sup>77</sup>	17.39 <sub>5</sub> <sup>5915</sup>
	20.118 <sub>8</sub> <sup>787</sup>	14.196 <sub>4</sub> <sup>333</sup>	13.304 <sub>2</sub> <sup>170</sup>	14.698 <sub>4</sub> <sup>295</sup>	22.65 <sub>7</sub> <sup>6823</sup>
$m = 1$	1.0723 <sub>3</sub> <sup>24</sup>	1.2625 <sub>1</sub> <sup>06</sup>	1.5601 <sub>6</sub> <sup>57</sup>	2.1232 <sub>0</sub> <sup>02</sup>	3.9593 <sub>0</sub> <sup>02</sup>
	6.2008 <sub>0</sub> <sup>06</sup>	5.3683 <sub>2</sub> <sup>22</sup>	5.2755 <sub>5</sub> <sup>44</sup>	5.9728 <sub>3</sub> <sup>26</sup>	9.4534 <sub>8</sub> <sup>66</sup>
	11.8821 <sub>2</sub> <sup>18</sup>	8.9418 <sub>1</sub> <sup>05</sup>	8.5044 <sub>4</sub> <sup>40</sup>	9.4762 <sub>2</sub> <sup>22</sup>	14.7548 <sub>4</sub> <sup>34</sup>
	17.358 <sub>9</sub> <sup>856</sup>	12.423 <sub>3</sub> <sup>266</sup>	11.683 <sub>5</sub> <sup>41</sup>	12.938 <sub>0</sub> <sup>03</sup>	20.022 <sub>4</sub> <sup>340</sup>
$m = 2$	2.1079 <sub>2</sub> <sup>16</sup>	2.3876 <sub>7</sub> <sup>70</sup>	2.8196 <sub>9</sub> <sup>93</sup>	3.6335 <sub>8</sub> <sup>73</sup>	6.3154 <sub>1</sub> <sup>63</sup>
	8.3952 <sub>3</sub> <sup>30</sup>	6.8866 <sub>6</sub> <sup>86</sup>	6.6594 <sub>1</sub> <sup>10</sup>	7.5087 <sub>0</sub> <sup>09</sup>	11.8582 <sub>2</sub> <sup>19</sup>
	14.2944 <sub>4</sub> <sup>11</sup>	10.5081 <sub>8</sub> <sup>76</sup>	9.9412 <sub>9</sub> <sup>89</sup>	11.058 <sub>3</sub> <sup>298</sup>	17.2101 <sub>3</sub> <sup>17</sup>
	19.80 <sub>9</sub> <sup>8901</sup>	14.021 <sub>7</sub> <sup>632</sup>	13.149 <sub>9</sub> <sup>899</sup>	14.547 <sub>4</sub> <sup>385</sup>	22.509 <sub>3</sub> <sup>191</sup>
$m = 3$	3.1294 <sub>9</sub> <sup>80</sup>	3.4664 <sub>2</sub> <sup>16</sup>	3.9941 <sub>6</sub> <sup>61</sup>	5.0122 <sub>5</sub> <sup>47</sup>	8.4631 <sub>0</sub> <sup>03</sup>
	10.4883 <sub>2</sub> <sup>17</sup>	8.3121 <sub>4</sub> <sup>40</sup>	7.9728 <sub>0</sub> <sup>06</sup>	8.9708 <sub>9</sub> <sup>90</sup>	14.1373 <sub>9</sub> <sup>89</sup>
	16.5802 <sub>1</sub> <sup>04</sup>	12.0063 <sub>7</sub> <sup>64</sup>	11.3199 <sub>6</sub> <sup>57</sup>	12.578 <sub>9</sub> <sup>861</sup>	19.565 <sub>3</sub> <sup>290</sup>
	22.15 <sub>5</sub> <sup>469</sup>	15.563 <sub>6</sub> <sup>479</sup>	14.566 <sub>9</sub> <sup>780</sup>	16.10 <sub>4</sub> <sup>206</sup>	24.91 <sub>2</sub> <sup>2201</sup>

Table 4: Spherical tanks filled to the depth  $d$ . Numerical results for the lowest four eigenvalues of modes  $m = 0, 1, 2, 3$ , compared to McIver’s analytic approximations. Using quadratic approximations with separate adaptive loops for each eigenvalue. Differences showed as stacked numbers, with McIver’s results below. In most cases rounding gives exactly the same results.

the eigenfunctions become more complicated (change sign more often), the shape of the domain (corners) become less important for the residual error estimators.

### 4.3 Spherical tank and shape approximation.

Now we turn to the shapes which cannot be exactly triangulated. A perfect example is a spherical tank, which has rounded profile. We will use mesh snapping technique described in [section 3.5.3](#) to improve our approximations. For a spherical container that is filled to an arbitrary level, the profile is a semicircle with a piece cut out at the top. Note that by approximating that shape with an inscribed polygon, we shrink the domain, hence lower the eigenvalue (this follows from the somewhat counterintuitive domain monotonicity [[5](#), Proposition 3.2.1], that works backwards compared to other Laplace eigenvalue problems). At the same time, finite element methods generally overestimate the eigenvalues. Therefore we get additional weak error cancellation. Similar cancellation is possible for any convex profile.

We consider a sphere of radius 1, filled to a depth  $d < 2$ . Note that the same parametrization was already considered by McIver [[28](#)] and Faltinsen-Timokha [[9](#)]. In both papers authors use the geometry of the sphere to either numerically or analytically approximate the sloshing eigenvalues of the spherical tank. Our finite element approach gives nearly the same values as the other authors. In [Table 4](#) we compare our approximations to McIver’s [Table 2](#).

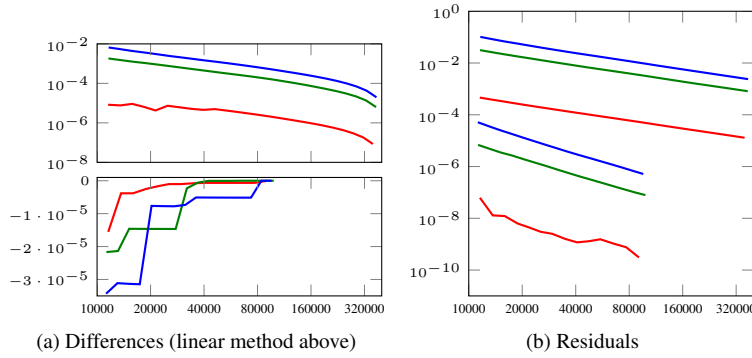


Fig. 8: Convergence of the adaptive loop for a half-full spherical tank and  $m = 1$ . Using linear and quadratic methods, but with one adaptive loop per eigenvalue. Note that the differences to best result are now negative for the quadratic method, with sharp jumps indicating successful perimeter refinement. At the same time residual norms behave almost exactly as in earlier, exact boundary cases. In the linear method, perimeter refinement seems insignificant.

The decay of the differences of the approximations is no longer easy to check, due to perimeter approximation having significant impact on eigenvalues (especially in the quadratic method). We illustrate this on [Figure 8a](#) for a half-full tank. The large jumps on the lower subplot (quadratic method) indicate steps with successful perimeter refinement. The almost flat parts of the graphs (in fact slightly decreasing) show steps without perimeter refinement, only with the residual norm based mesh refinement. As mentioned before, the perimeter snapping enlarges the domain, hence increases the eigenvalue. This effect dwarfs any decreases caused by finer mesh eigenvalue calculations on the quadratic method.

Crucially, the residual norms still converge in a predictable manner (see [Figure 8b](#)). Only the lowest eigenvalue, with the simplest eigenfunction is slightly influenced by the changing shape of the domain. This confirms our observation from previous section, that the lower the eigenvalue, the more important the boundary for the adaptive loop. The linear approach seems unaffected by the perimeter approximation, with positive differences to the best result.

One might try to further improve the quadratic approach by trying to balance the perimeter and the eigenvalue approximations. However, extreme density of triangles near the curved boundary will eventually lead to significant rounding errors, singular matrices.

For our adaptive loop we obtained the triangle densities from [Figure 9](#) (fill levels  $d = 0.2, 1.0$  and  $1.8$ ). Note that the deeper the tank, the less relevant its bottom, and the higher the eigenvalue, the more triangles near the free boundary  $F$ . At the same time, the curved boundary is very heavily refined due to perimeter snapping.



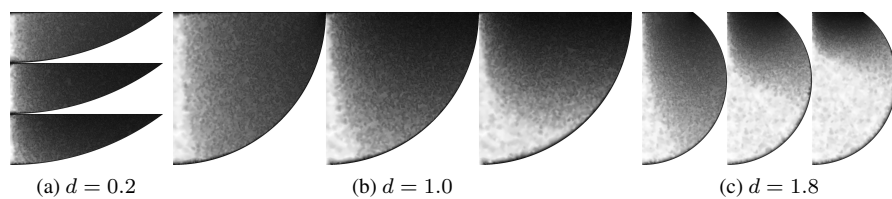


Fig. 9: Sizes of the mesh triangles (darker color indicates smaller triangles) for the mode  $m = 1$  of a half-full spherical tank. Note very small triangle on the curved boundary.

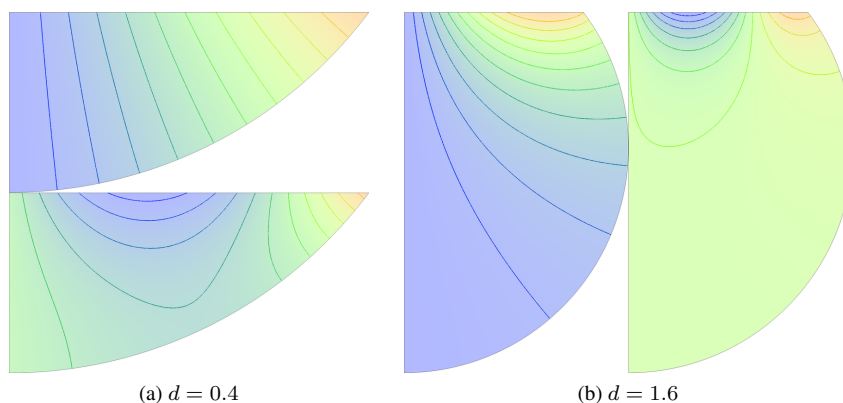


Fig. 10: Plots of the first two eigenfunctions of the mode  $m = 1$  for spherical tanks.

Finally we make density plots (with a few level curves) of two eigenfunctions for the spherical tanks filled to the levels  $d = 0.4$  and  $d = 1.6$  on [Figure 10](#). We again notice that the higher eigenfunctions fluctuate more near  $F$ , and that the deep parts of the containers are nearly insignificant.

We also provide sloshing profiles for the fundamental modes for various depths on [Figure 11](#), where we also mark the high-spots (cf. Figure 2 in Faltinsen-Timokha [9]). We include the ice-fishing problem (discussed and solved numerically in [section 4.1.1](#)), since it can be understood as the limiting case  $d = 2$  (full tank), see McIver [28]. Note that numerical approximation of the ice-fishing using cylindrical tank is much easier than using spherical tanks.

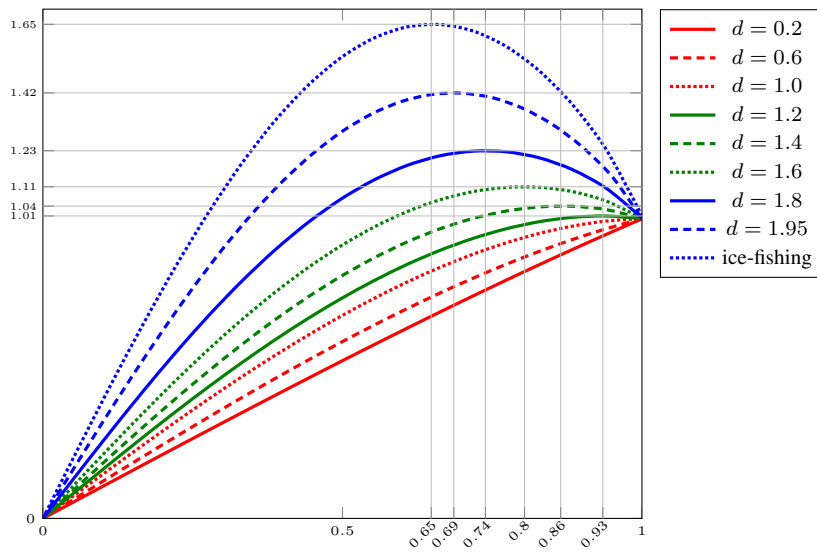


Fig. 11: Sloshing profiles for the lowest eigenvalues of spherical tanks, with high-spots marked. The ice-fishing problem is equivalent to  $d = 2.0$ .

## 5 Images of high spots

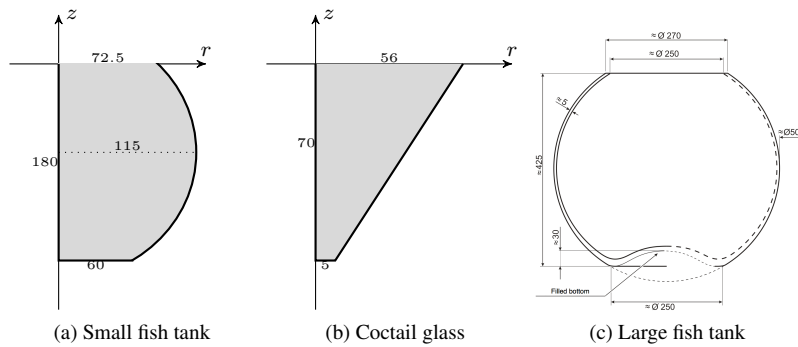


Fig. 12: Shapes of the tanks we used to take photographs (lengths in millimeters). The first two profiles were also used in Kuznetsov et. al. [24].

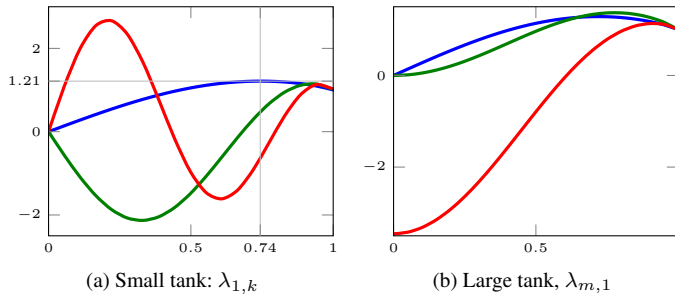


Fig. 13: Profiles of the sloshing water in a fish tank.

To further check our numerical methods, and the mathematical model itself, we attempted to photograph a sloshing liquid in various containers (tank profiles shown on Figure 12, numerical sloshing profiles on Figure 13). Note that throughout this paper we used the simplified linear model of sloshing. Clearly, it cannot be applied to strong turbulent sloshing. But we visually check that it is very accurate for infinitesimal sloshing. To reiterate, we did not attempt to eliminate any nonlinearities intrinsic to strong sloshing from our experiment, opting instead for capturing long time behavior. After a long time, all nonlinear phenomena (with higher order of decay) are insignificant in comparison to linear wave modes we study.

In order to get accurate experimental results for infinitesimal sloshing we needed a nearly still water surface. We achieve this by first nudging a container several times in roughly equal intervals, corresponding to theoretical sloshing frequency. After that, we allow the water waves to dampen to a point of being barely visible.

Next, we photograph a reflection of a dotted piece of paper on the nearly still surface of the water, using a few seconds long exposure to blur the dots. The setup of the experiment is shown on Figure 14, together with the results on the first two containers from Figure 12. This figure is reproduced from [24], but we overlay the numerical results on the boubous container images Figure 14a and Figure 14b. The dotted red line shows the numerically calculated distance from the center of the free surface to the high spot, assuming the solid line is the boundary of the surface. We used our finite element method to find the radius of the dotted circle, using container measurements from Figure 12a.

The results of the experiment on the small fishbowl (profile from Figure 12a) agree with the infinitesimal, linear model. Most of the dots trace a path caused by the changing tangent planes of the moving surface. Only the dots corresponding to the local extrema of the sloshing amplitude remain still, hence sharp on our photographs. Note that the lowest sloshing eigenvalue has multiplicity two, hence it is possible to get phase-shifted combination of the two corresponding modes, giving an illusion of whirling from Figure 14b. The sloshing is irrotational, so the wave is a standing wave, but the extrema travel around the red dotted circle.

The cocktail glass (profile from Figure 12b) also gives the results (Figure 14c), that are consistent with the model. The high spot seems to be on the boundary and all

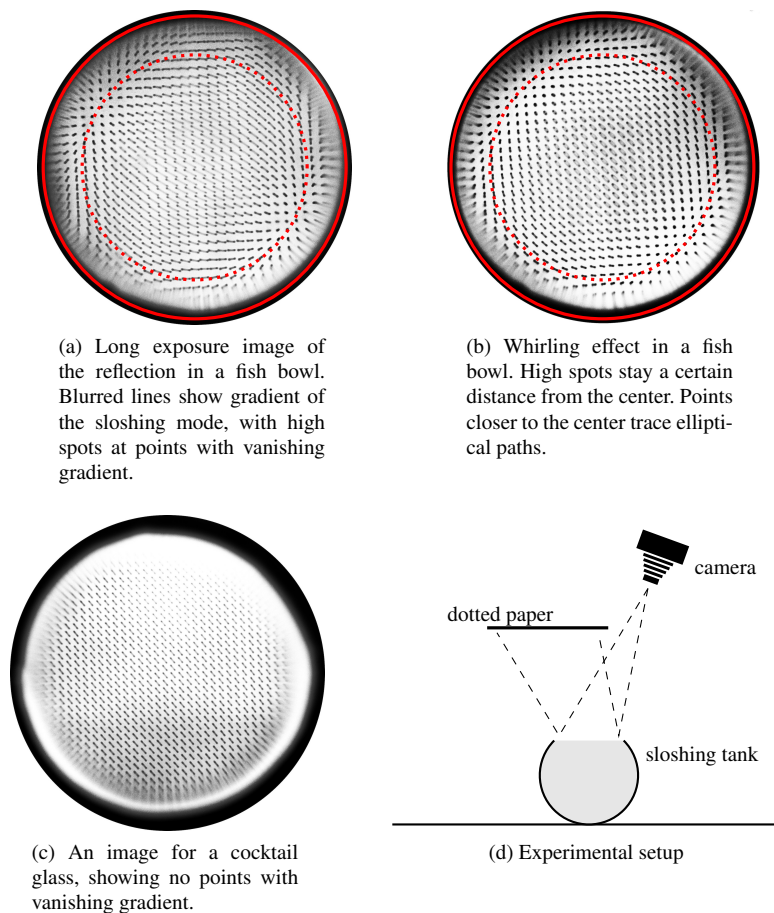


Fig. 14: Photographic experiment

dots trace roughly the same path. This suggests that the sloshing profile is nearly a flat plane. In fact, certain cones do have exactly planar sloshing profile (see Troesch [32] or a brief discussion in section 3.1.1).

### 5.1 Larger tank and higher modes

Our numerical results seem to almost exactly match the experiments. However, alert reader may notice very significant blur (on all photographs) very near the edges of the free surface. This is most likely caused by the interaction of the water surface and the container. As noted earlier, we did not try to eliminate such nonlinear phenomena. Instead, we performed the same experiment in a larger fishbowl with the shape from Figure 12c. Much larger free surface should yield less significant boundary in-

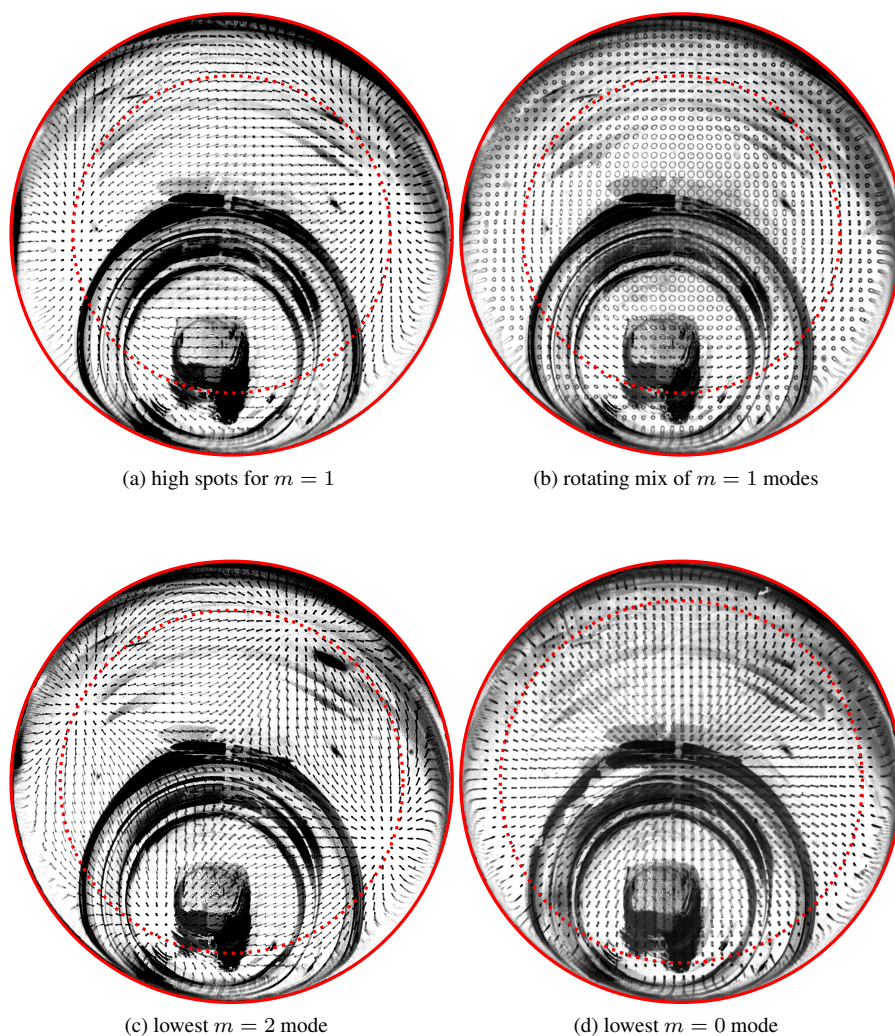


Fig. 15: Large fishbowl. Pictures of the eigenfunctions for the lowest three modes. Note the reflection caused by the protruding, convex bottom of the tank, showing a slightly off-center perspective of our camera setup from [Figure 14d](#).

teraction. Indeed, [Figure 15](#) still shows great accuracy. [Figure 15a](#) and [Figure 15b](#) show the modes for the lowest eigenvalue (cf. [Figure 14a](#) and [Figure 14b](#)). The latter emphasizes the whirling effect we first noticed on the smaller tank.

The larger tank allowed us to capture the eigenfunctions for the lowest eigenvalues of higher modes of oscillations:  $m = 2$  on [Figure 15c](#) and  $m = 0$  on [Figure 15d](#). To capture the second mode we changed the frequency of nudging to match the sec-

	eigenvalue	high spot position	high spot height
$m = 1$	$2.453^{5073}_{0649}$	$0.7^{19}_{20}$	$1.28^{504}_{486}$
$m = 2$	$3.86169^{97}_{23}$	0.777	1.3693
$m = 0$	$4.07333^{33}_{29}$	0	3.4562
		0.912	1.1354

Table 5: Upper and lower bounds for the eigenvalues of the large fish tank from Figure 12c as well as positions of the high spots (extrema) and their relative height compared to boundary value. The bounds result from replacing the curved bottom with flat one, so that the new tank is either contained (lower bound) or contains (upper bound) the original tank. The accuracy of the position of the high spot is limited by the sampling of the numerical solution on the free surface. Heights do differ but on too high digits to be of any significance, especially given low position accuracy.

ond natural frequency. As a result we have four local maximal positioned on the numerically calculated dotted circle (consistent with  $\cos(2\theta)$  in the eigenfunction decomposition (5)). To capture the lowest radial mode  $m = 0$  we repeatedly dipped a tip of a pencil (with appropriate frequency). The extrema form a circle that is not far from the numerically predicted position. Nevertheless, that last circle is already quite close to the boundary, so the interactions of the water and the container seem to influence accuracy of the experiment.

A protruding bottom of the large fish tank (see Figure 12c) allows us to also test the dependence of the eigenvalues on the shape of the bottom of the tank. Clearly, in a deep container the dependence should be nearly insignificant. The eigenvalue of the tank (assuming it is perfectly spherical) is contained between the eigenvalues of the spherical tanks with 125mm radius cut at the top and either 125mm or 155mm straight cuts at the bottom (the shape from Figure 12a with slightly different proportions). We can use domain monotonicity argument (see [5, Proposition 3.2.1]) to give upper and lower bounds for the eigenvalues of the original large tank (note that somewhat counterintuitively the smaller the tank, the smaller the eigenvalue). Table 5 shows the bounds for the eigenvalues as well as the high spots for the lowest three eigenfunctions (corresponding to the lowest eigenvalues of modes  $m = 1, 2$  and 0). Note that the high spot positions and heights are not bounds, but merely approximations.

## Acknowledgements

Bartłomiej Siudeja was partially supported by the Polish National Science Centre grant 2012/07/B/ST1/03356. T. Kulczycki and M. Kwanicki were supported in part by Project S30011/I-18 of the Institute of Mathematics and Computer Science of Wrocław University of Technology.

The authors would like to thank N. Kuznetsov for many useful discussions on the subject of the paper and for sharing his knowledge about the sloshing problem.



## References

1. **M. Ainsworth and J. T. Oden**, *A posteriori error estimation in finite element analysis*, Pure and Applied Mathematics (New York), Wiley-Interscience [John Wiley & Sons], New York, 2000. [MR1885308](#)
2. **H. A. Ardakani and T. J. Bridges**, *Shallow-water sloshing in vessels undergoing prescribed rigid-body motion in three dimensions*, *J. Fluid Mech.* **667** (2011), 474–519.
3. **M. G. Armentano and C. Padra**, *A posteriori error estimates for the Steklov eigenvalue problem*, *Appl. Numer. Math.* **58** (2008), no. 5, 593–601. [MR2407734](#)
4. **I. Babuška and J. Osborn**, *Eigenvalue problems*, Handbook of numerical analysis, Vol. II, Handb. Numer. Anal., II, North-Holland, Amsterdam, 1991, pp. 641–787. [MR1115240](#)
5. **R. Bañuelos, T. Kulczycki, I. Polterovich, and B. Siudeja**, *Eigenvalue inequalities for mixed Steklov problems*, Operator theory and its applications, Amer. Math. Soc. Transl. Ser. 2, vol. 231, Amer. Math. Soc., Providence, RI, 2010, pp. 19–34. [MR2758960](#)
6. **C. Bernardi, M. Dauge, and Y. Maday**, *Spectral methods for axisymmetric domains*, Series in Applied Mathematics (Paris), vol. 3, Gauthier-Villars, Éditions Scientifiques et Médicales Elsevier, Paris; North-Holland, Amsterdam, 1999, Numerical algorithms and tests due to Mejdí Azañez. [MR1693480](#)
7. **T. Betcke and L. N. Trefethen**, *Reviving the method of particular solutions*, *SIAM Rev.* **47** (2005), no. 3, 469–491 (electronic). [MR2178637](#)
8. **P. Blanchard and E. Brüning**, *Variational methods in mathematical physics*, Texts and Monographs in Physics, Springer-Verlag, Berlin, 1992, A unified approach, Translated from the German by Gillian M. Hayes. [MR1230382](#)
9. **O. M. Faltinsen and A. N. Timokha**, *Analytically approximate natural sloshing modes for a spherical tank shape*, *J. Fluid Mech.* **703** (2012), 391–301. [MR2949919](#)
10. **O. M. Faltinsen and A. N. Timokha**, *Analytically approximate natural sloshing modes and frequencies in two-dimensional tanks*, *European Journal of Mechanics B/Fluids* **47** (2014), 176–187.
11. **N. Filonov**, *On an inequality for the eigenvalues of the Dirichlet and Neumann problems for the Laplace operator*, *Algebra i Analiz* **16** (2004), no. 2, 172–176. [MR2068346](#)
12. **L. Fox, P. Henrici, and C. Moler**, *Approximations and bounds for eigenvalues of elliptic operators*, *SIAM J. Numer. Anal.* **4** (1967), 89–102. [MR0215542](#)
13. **D. W. Fox and J. R. Kuttler**, *Sloshing frequencies*, *Z. Angew. Math. Phys.* **34** (1983), no. 5, 668–696. [MR723140](#)
14. **E. M. Garau and P. Morin**, *Convergence and quasi-optimality of adaptive FEM for Steklov eigenvalue problems*, *IMA J. Numer. Anal.* **31** (2011), no. 3, 914–946. [MR2832785](#)
15. **I. Gavriluk, M. Hermann, I. Lukovsky, O. Solodun and A. Timokha**, *Natural sloshing frequencies in rigid truncated conical tanks*, *Eng. Comput.* **25** (2008), 518–540.
16. **A. Herczyński and P. D. Weidman**, *Experiments on the Periodic Oscillations of Free Containers Driven by Liquid Sloshing*, *J. Fluid Mech.* **693** (2012).
17. **R. Ibrahim**, *Liquid sloshing dynamics*, Theory and applications (Cambridge University Press), Cambridge, UK, 2005.
18. **N. D. Kopachevsky and S. G. Krein**, *Operator approach to linear problems of hydrodynamics. Vol. 1*, Operator Theory: Advances and Applications, vol. 128, Birkhäuser Verlag, Basel, 2001, Self-adjoint problems for an ideal fluid. [MR1860016](#)
19. **V. Kozlov and N. Kuznetsov**, *The ice-fishing problem: the fundamental sloshing frequency versus geometry of holes*, *Math. Methods Appl. Sci.* **27** (2004), no. 3, 289–312. [MR2023011](#)
20. **A. Kufner**, *Weighted Sobolev spaces*, Teubner-Texte zur Mathematik [Teubner Texts in Mathematics], vol. 31, BSB B. G. Teubner Verlagsgesellschaft, Leipzig, 1980, With German, French and Russian summaries. [MR664599](#)
21. **T. Kulczycki and N. Kuznetsov**, *‘High spots’ theorems for sloshing problems*, *Bull. Lond. Math. Soc.* **41** (2009), no. 3, 495–505. [MR2506833](#)
22. **T. Kulczycki and N. Kuznetsov**, *On the ‘high spots’ of fundamental sloshing modes in a trough*, *Proc. R. Soc. Lond. Ser. A Math. Phys. Eng. Sci.* **467** (2011), no. 2129, 1491–1502. [MR2782167](#)
23. **T. Kulczycki and M. Kwaśnicki**, *On high spots of the fundamental sloshing eigenfunctions in axially symmetric domains*, *Proc. Lond. Math. Soc. (3)* **105** (2012), no. 5, 921–952. [MR2997042](#)
24. **N. Kuznetsov, T. Kulczycki, M. Kwaśnicki, A. Nazarov, S. Poborchi, I. Polterovich and B. Siudeja**, *The Legacy of Vladimir Andreevich Steklov*, *Notices of AMS* **61** (2014), no. 1, 9–22.
25. **H. Lamb**, *Hydrodynamics*, Cambridge University Press (1932).

26. **A. Logg, K.-A. Mardal, and G. Wells**, *Automated solution of differential equations by the finite element method. The FEniCS book.*, Lecture Notes in Computational Science and Engineering 84. Berlin: Springer. xiii, 723 p. (2012).
27. **I. A. Lukovskii, M. Y. Barnyak, and A. N. Komarenko**, *Approximate methods for solving problems of the dynamics of a bounded volume of fluid*, “Naukova Dumka”, Kiev, 1984. [MR780750](#)
28. **P. McIver**, *Sloshing frequencies for cylindrical and spherical containers filled to an arbitrary depth*, *J. Fluid Mech.* **201** (1989), 243–257. [MR993197](#)
29. **B. Mercier and G. Raugel**, *Résolution d'un problème aux limites dans un ouvert axisymétrique par éléments finis en  $r, z$  et séries de Fourier en  $\theta$* , *RAIRO Anal. Numér.* **16** (1982), no. 4, 405–461. [MR684832](#)
30. **N. N. Moiseev**, *Introduction to the theory of oscillations of liquid-containing bodies*, *Advances in Applied Mechanics*, Vol. 8, Academic Press, New York, 1964, pp. 233–289. [MR0167074](#)
31. **S. Rebouillat and D. Liksonov**, *Fluidstructure interaction in partially filled liquid containers: A comparative review of numerical approaches*, *Computers & Fluids* **39** (2010), 739–746.
32. **B. A. Troesch**, *Free oscillations of a fluid in a container*, *Boundary problems in differential equations*, Univ. of Wisconsin Press, Madison, Wis., 1960, pp. 279–299. [MR0114437](#)
33. **C-H. Wu and B-F. Chen**, *Sloshing waves and resonance modes of fluid in a 3D tank by a time-independent finite difference method*, *Ocean Engineering* **26** (2009), no. 6–7, 500–510.