UDC 519.632.4

R. H. Drebotiy, H. A. Shynkarenko✉

HEURISTIC CHOICE OF THE REGULARIZATION PARAMETER FOR OPTIMAL STABILIZATION OF THE FINITE ELEMENT APPROXIMATIONS

The problem of the optimal choice of the regularization parameter in the stabilization scheme of the finite element method for the singularly perturbed diffusion – advection – reaction problem is considered. Stabilization is based on the combination of Tikhonov-type regularization with the auxiliary Cauchy problem. The behavior of perturbations in the approximate solution with respect to change in the regularization parameter is studied. On the basis of performed analysis the heuristic criterion for the optimal choice of the regularization parameter is constructed. The criterion is formulated as a local problem of minimization for corresponding function constructed in a view of composition of a linear functional and the obtained finite element approximation. The proposed approach is developed for one-dimensional problems and then generalized for 2D problems. The possibility of using the Harrow – Hassidim – Lloyd quantum algorithm in combination with the swap test to implement the computation of the obtained loss function on quantum computers also is discussed.

Key words: *finite element method, diffusion – advection – reaction model, stabilization schemes, Cauchy problem, discrete Laplace operator, Harrow – Hassidim – Lloyd algorithm.*

Introduction. The diffusion – advection – reaction (DAR) model plays an important role in simulating air pollution migration [2, 14]. There are cases when advection is much faster than diffusion. Problems of this kind are called singularly perturbed. The efficient application of the finite element method (FEM) to the DAR model in this situation is complicated due to the special structure of the solution of this model. Advection-dominated problems always lead to the existence of thin boundary layers in which the gradient of the solution is very large. For such a solution structure, it is impossible to achieve adequate accuracy using the FEM on uniform coarse meshes due to the high rate of parasitic oscillations in the obtained approximations on the entire domain of the problem. The classical a priori error estimates for FEM [2, 4, 16, 18] always depend on the diameter of the mesh elements and some norms of the derivatives of the exact solution. This fact gives us the chance to obtain an accurate approximation of singularly perturbed problems by using uniformly refined meshes with the elements of smaller diameter. Actually, the count of elements which we need in this case is very large, and this makes the practical algorithm very computationally intensive.

There are currently two general approaches to solving the advection-dominated problems. The first of them is to use mesh adaptivity [1, 2, 4, 7]. In this approach we use *nonuniform* meshes, which are adapted to the exact solution to represent its structure: they use smaller elements around the boundary layer and larger elements in the rest of the domain (or even elements with a different polynomial order as in p- or hp-adaptivity [7]). The second approach is to apply stabilization schemes [1, 6, 18]. In this case we modify the Galerkin FEM scheme with additional penalty terms to add some dummy artificial diffusion component [3]. This approach often leads to quite smooth and accurate approximate solutions. Special words can be said about discontinuous finite elements [3, 4, 9], which also add some penalty terms to the Galerkin scheme, but the role of these penalty terms is more related to the connectivity between the elements, rather than to overcoming singular perturbations.

In [8] we proposed the new FEM scheme which is somewhat similar to

✉roman.drebotiy@gmail.com

known stabilization procedures, but on the other hand, it tries to incorporate more a posteriori knowledge about the exact solution. The proposed scheme combines a regularization procedure similar to the standard Tikhonov-type [13] regularization with an auxiliary Cauchy problem corresponding to the initial DAR model. The proposed scheme has one real positive regularization parameter λ, which plays a crucial role in obtaining an accurate approximation. The seminal article [8] does not provide any guidance on how we should choose this parameter.

In this article, we present a simple optimization-based heuristic strategy for choosing the regularization parameter based on the analysis of numerical experiments and some theoretical observations. The proposed algorithm leads to a fully automatic finite element stabilization procedure.

The proposed approach has certain benefits compared to adaptive schemes, which we will discuss later. One of them is the fact, that it is suitable for implementation on quantum computers, which will be discussed later in the article.

The paper is structured as follows: first, we define the DAR model problem; then we present and review the algorithm from [8], after which we discuss the main problem of this article and introduce a new algorithm; then we provide some numerical results for the one-dimensional DAR problem; next, we provide generalization for *2D* problems and a corresponding numerical example; and finally, we discuss the possibility of using quantum computers to automatically choice of the regularization parameter.

1. Model problem. We consider the following boundary value problem (BVP) for the diffusion − advection − reaction equation:

$$\begin{aligned} &\textit{find function:} \quad u : \bar{\Omega} \to \mathbb{R} \quad \textit{such that} \\ &\quad -\mu\Delta u + \boldsymbol{\beta}\cdot\nabla u + \sigma u = f \quad \textit{in} \quad \Omega \subset \mathbb{R}^2, \\ &\quad u = 0 \qquad\qquad\qquad\quad \textit{on} \quad \Gamma = \partial\Omega. \end{aligned} \tag{1}$$

Here Ω is a bounded domain with a Lipschitz boundary $\Gamma = \partial\Omega$, $\mu = \text{const} > 0$ and $\sigma = \text{const} > 0$ are coefficients of diffusion and reaction, respectively, function $f = f(x)$ and vector $\boldsymbol{\beta} = \{\beta_i(x)\}_{i=1}^2$ represent the sources and advection flow velocity, respectively. We will consider an incompressible flow, i.e., $\nabla\cdot\boldsymbol{\beta} = \dfrac{\partial}{\partial x_1}\beta_1 + \dfrac{\partial}{\partial x_2}\beta_2 = 0$ in Ω.

The boundary value problem (1) admits the following variational formulation:

$$\begin{aligned} &\textit{find function:} \quad u \in V := H_0^1(\Omega) \quad \textit{such that} \\ &\quad a(u,v) = \langle \ell, v\rangle \qquad \forall v \in V, \end{aligned} \tag{2}$$

where

$$a(u,v) = \int_\Omega (\mu\nabla u\cdot\nabla v + \boldsymbol{\beta} v\cdot\nabla u + \sigma uv)\,dx \qquad \forall u, v \in V,$$

$$\langle \ell, v\rangle = \int_\Omega fv\,dx \qquad \forall v \in V.$$

Here and below we assume that the problem data are quite regular and satisfy the hypotheses of the Lax − Milgram lemma. Under these conditions, problem (2) has a unique weak solution $u \in V$, moreover, its bilinear form generates a new (energy) norm in the space of admissible functions V that is equivalent to the standard norm of the Sobolev space $H^1(\Omega)$.

It is well-known that the large values of the Péclet number

$$\mathrm{Pe} := \mu^{-1} \|\boldsymbol{\beta}\|_\infty \operatorname{diam} \Omega$$

or/and the Strouhal number $\mathrm{St} := \sigma \|\boldsymbol{\beta}\|_\infty^{-1} \operatorname{diam} \Omega$, where $\|\boldsymbol{\beta}\|_\infty = \left(\sum_{i=1}^{2} \operatorname{ess\,sup}_{x \in \Omega} |\beta_i(x)|^2 \right)^{1/2}$, indicate, that problem (2) may be singularly perturbed [2, 4, 19]. This fact shows a small margin of stability of the sought solution and, as consequence, warns of excessive computational costs when finding high-quality finite element approximations of this solution. Actually, a number of adaptive and stabilized finite element schemes have been developed for the successful solution of the singularly perturbed problems, for details see [1–3, 7, 8, 19]. Below we present one of the stabilized schemes recently proposed by the authors [8] for advection-dominated problems, i.e. problems with large values of Péclet number.

2. Finite element regularization and auxiliary Cauchy problem. To deal with large Péclet numbers, in [8] we proposed the following stabilization procedure. Let $\Gamma_0 := \{x \in \partial\Omega \mid \mathbf{n}(x) \cdot \boldsymbol{\beta}(x) < 0\}$, and $\mathbf{n}$ is a unit vector of outward normal to the boundary of the domain Ω.

Let us consider the following *reduced problem*:

find function: $u_0 \in C^1(\Omega)$ *such that*

$$\boldsymbol{\beta} \cdot \nabla u_0 + \sigma u_0 = f \quad \text{in} \quad \Omega,$$

$$u_0|_{\Gamma_0} = 0, \tag{3}$$

and replace the original variational problem (2) with the following one:

given: *parameter* $\lambda = \text{const} \geq 0$ *and solution of* (3) u_0,

find: *function* $u^* \in V$ *such that*

$$a(u^*, v) + \lambda (u^*, v)_V = \langle \ell, v \rangle + \lambda (u_0, v)_V \quad \forall v \in V. \tag{4}$$

Problem (3) is actually Cauchy problem restricted to the domain Ω. We suppose that there are no closed integral curves of the vector field $\boldsymbol{\beta}$ that are entirely contained in domain Ω.

For the numerical solution of (3), in [8] it is proposed to use the method of characteristics in combination with some methods such as Runge – Kutta to solve the obtained system of ordinary differential equations. Let us briefly describe the algorithm from [8]. Consider the case when $\boldsymbol{\beta} \neq 0$ at any point on Ω. Let define a function $[0, \operatorname{mes} \Gamma_0] \ni \eta \mapsto \rho(\eta) = (\rho_1(\eta), \rho_2(\eta)) \in \overline{\Gamma}_0$ that maps a parameter η bijectively onto the $\overline{\Gamma}_0$, that is, we constructed a parametrization of the set Γ_0. Let assume that the direction of increase of the parameter η corresponds to the movement along the Γ_0 in the counterclockwise direction with respect to Ω. For each value of η, we can find the integral curve $x(t, \eta) = (x_1(t, \eta), x_2(t, \eta)) \in \Omega$ of the vector field $\boldsymbol{\beta}$ as a solution of the following Cauchy problem for the system of two ordinary differential equations (ODE) written in vector form:

$$\frac{\partial x(t, \eta)}{\partial t} = \boldsymbol{\beta}(x(t, \eta)),$$

$$x(0, \eta) = \rho(\eta). \tag{5}$$

Let us define the function $z(t, \eta) = u_0(x(t, \eta))$. Taking into account (5) and

using the chain rule for total derivatives, we can derive the following:

$$\frac{\partial z(t,\eta)}{\partial t} = \boldsymbol{\beta}(x(t,\eta)) \cdot \nabla_x u_0(x(t,\eta)).$$

Using the last equality and (5), we can rewrite problem (3) as a Cauchy problem for a system of three scalar ODEs:

$$\begin{aligned}
&\frac{\partial x(t,\eta)}{\partial t} = \boldsymbol{\beta}(x(t,\eta)),\\
&\frac{\partial z(t,\eta)}{\partial t} = f(x(t,\eta)) - \sigma z(t,\eta),\\
&x(0,\eta) = \rho(\eta),\\
&z(0,\eta) = 0.
\end{aligned} \tag{6}$$

In the literature, the method used by us to reduce the first-order PDE to the system of ODEs is called the method of characteristics [10]. Let us summarize the steps.

Algorithm 1. *Solving reduced problem.*

1°) Generate some one-dimensional mesh of points on the curve Γ_0.

2°) For each of the generated points, construct an approximate solution of (6) using some methods such as Runge − Kutta or similar.

3°) Interpolate the solutions to obtain a single approximation (for example, we can use splines for this. A more efficient interpolation can also be constructed for this case, but that is beyond the scope of this article).

Taking into account the previous algorithm, we can introduce the following finite element regularization scheme.

Algorithm 2. *FEM regularization.*

1°) Solve Cauchy problem (3) to obtain an approximation u_0 using **Algorithm 1**.

2°) Choose a parameter λ in some way.

3°) Apply the FEM to obtain the numerical solution of (4) and obtain the numerical approximation u_h^*.

Note, that in this paper we consider *only linear* finite elements.

The choice of the regularization parameter on step ***2°*** is a key step to successful application of the described algorithm. The Péclet number for problem (4) can be calculated as

$$\mathrm{Pe}_{\mathrm{reg}} = (\mu + \lambda)^{-1} \|\boldsymbol{\beta}\|_\infty \operatorname{diam} \Omega. \tag{7}$$

By adding positive λ we will reduce the effect of high-frequency oscillations and this will allow us to use coarse meshes in **Algorithm 2**. Using a small number of elements will not ensure the ability to precisely reproduce the boundary layer of the computed solution. In practice, when we model the distribution of air pollution, we are much more interested in having an enough accurate approximation that cover the structure of exact solution, and it is not critical not to have a precise representation of the boundary layer, since it will be very thin and concentrated only on those finite elements, which touch a certain part of the boundary.

Problem (4) represents the necessary balance between using the solution of the original problem (1) and the solution for its limit case as $\mu \to 0$, that is, the reduced problem (3). Later, we will construct an example, where neither the computed u_0 nor the FEM approximation on a coarse mesh, nor the regularized approximation with set $u_0 \equiv 0$ on a coarse mesh are able to appro-

ximate a sufficiently accurate exact solution in contrast to **Algorithm 2** with a specially chosen λ.

3. Optimal choice of regularization parameter. In this section we establish some heuristic rules based mostly on the numerical experimental observations used by us to construct the loss function and the appropriate optimization problem for regularization parameter optimal choice procedure. Next, we present a practical algorithm for solving this optimization problem.

First, we consider the one-dimensional DAR model problem:

$$\begin{aligned} &\textit{find:} \quad \textit{function} \quad u : \bar{\Omega} \to \mathbb{R} \quad \textit{such that} \\ &-(\mu u')' + \beta u' + \sigma u = f \quad \textit{in} \quad \Omega = (0,1), \\ &u(0) = u(1) = 0, \end{aligned} \tag{8}$$

where μ, β, σ, f are some functions. The corresponding variational formulation (2) of problem (8) has the following bilinear and linear forms:

$$a(u,v) = \int_0^1 (\mu u'v' + \beta u'v + \sigma uv)\, dx,$$

$$\langle \ell, v \rangle = \int_0^1 fv\, dx.$$

Having analyzed many examples of numerical solutions of equation (4) for different singular perturbed problems and different λ on coarse meshes, we arrive to the observation that the dependence of u_h^* on λ always has the same properties. Let us demonstrate this on one case of an advection-dominated problem. Consider problem (8) with constant coefficients: $\mu = \sigma = 1$, $\beta = f = 10^3$. The exact solution has a boundary layer at the right end of the domain $[0,1]$. Equation (4) we solved using the FEM based on the Galerkin method with linear basis functions. All approximations are obtained on a uniform mesh of 20 finite elements. Fig. 1 shows the plots of FEM approximation for an increasing sequence of the values of λ (vertical scaling may be different for each subplot). The exact solution of the problem is shown by a dashed line on each of subplots. The topmost approximation (for $\lambda = 0$) corresponds to the classical FEM approximation without regularization. The basic property of this approximation is that it has high-frequency oscillations spread far from the boundary layer – in this particular case, in the entire domain of definition of the problem, i.e., a sign of the approximation derivative changes on each consecutive finite element.

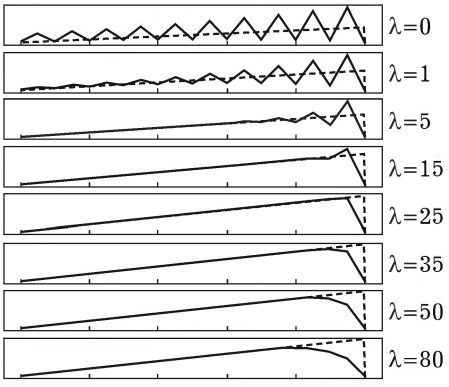


Fig. 1. Finite element approximations of problem (6) with $\mu = \sigma = 1$, $\beta = f = 10^3$ for different $\lambda = 0, 1, 5, 15, 25, 35, 50, 80$ (top to down). The dashed lines represent the exact solutions.

Evolution of u_h^*, when λ increases, may be considered as a certain smoothing. Consider part of the graphs in Fig. 1 without the last finite element, containing the boundary layer with high value of gradient. As we can observe, the smoothing process has three phases:

I – uniform smoothening of the high-frequency component. During this phase, gradient jumps decrease almost uniformly to 0 over all elements (not taking to account last element);

II – further smoothening starts: some increasing gradient jumps near the boundary layer to smooth corner-like graph rightmost part;

III – further smoothening by spreading this process from the boundary layer to the entire domain. This results in reducing of gradient jumps over all elements.

The phases described above can be characterized as the consecutive elimination of oscillation frequency. The smoothing process removes the high-frequency components first, followed by the lower and progressively lower frequencies. Note, that such frequency elimination behavior can be described theoretically in terms of singular value decomposition for some cases of classical Tikhonov regularization [11].

To obtain an adequate and valuable approximation we actually need to somehow catch the point between phases **I** and **II**. At this point all parasitic high-frequency per-element oscillations are eliminated, but further smoothing does not yet take effect. To find this point, we construct some quantity that will represent phases **I**–**III** and it will measure per-element oscillations which we want to eliminate from approximation.

Assume that we have n linear finite elements. Let $[x_{i-1}, x_i]$ be the domain of the ith element. Suppose, we computed the approximation u_h^* of problem (4) on this mesh with some regularization parameter. Denote by $g_i := u_h^*(x_i) - u_h^*(x_{i-1})$ the derivative (gradient) of the approximation on ith element. Consider the vector of gradient jumps

$$\mathbf{j} = (j_1, j_2, \ldots, j_{n-2}), \tag{9}$$

where $j_k := g_{k+1} - g_k$ is actual jump of derivative between the elements. Note that we excluded the last element as in our description of smoothing process above. This will be clarified later. Also note, that jump can be reformulated as $j_k := \left(g_{k+1} - \frac{g_k + g_{k+1}}{2}\right) - \left(g_k - \frac{g_k + g_{k+1}}{2}\right)$, showing that it measures the change of gradient on element crossing with respect to average tangent coefficient.

The vector $\mathbf{j}$ measures gradient jumps over all elements except the last one. We need to somehow extract from this vector the integral quantity which will indicate how much these jumps oscillate with the change of sign. To do this, we first define the reference oscillating vector:

$$\mathbf{e} := (\underbrace{1, -1, 1, -1, \ldots, (-1)^{n-3}}_{n-2}). \tag{10}$$

This vector includes per-element oscillating pattern. Now we simply measure the similarity between jumps vector $\mathbf{j}$ and $\mathbf{e}$ by computing some kind of "correlation" in the form of inner product:

$$F(\lambda) := |(\mathbf{j}, \mathbf{e})| \tag{11}$$

By expanding the last expression, we get

$$F(\lambda) = \left| \sum_{k=1}^{n-2} (-1)^{k-1} \delta^2 u_h^*(x_k; \lambda) \right| \tag{12}$$

where by $\delta^2 u_h^*(x_k;\lambda)$ we denote the second-order central difference in appropriate point:

$$\delta^2 u_h^*(x_k;\lambda) = u_h^*(x_{k+1};\lambda) - 2u_h^*(x_k;\lambda) + u_h^*(x_{k-1};\lambda) \tag{13}$$

and by $u_h^*(x;\lambda)$ we denoted the approximation $u_h^*(x)$ computed for some λ. Note, that formula (12) can be used also for non-uniform meshes and thus allows to combine adaptive schemes with the proposed approach in a certain way.

Fig. 2 demonstrates the general structure of the graph of $F(\lambda)$. The graph always consists of three parts that directly correspond to the smoothing phases described above and they are marked with the corresponding numbers. So, the curve $F(\lambda)$ consists of the following three parts:

1) high-gradient descending part, which corresponds to the elimination of the parasitic oscillations. Such a high descending rate is explained by the uniform decrease of oscillations over entire domain;

2) ascending part, which corresponds to the smoothening of the rightmost corner-like part of FEM approximation (i.e., the smoothening will eliminate the representation of the boundary layer in the approximation structure);

3) final descending part, which corresponds to the final smoothing of the entire approximation.

Due to the special structure of the curve $F(\lambda)$, we can choose optimal value of λ as the point, in which $F(\lambda)$ has its local minimum immediately after phase **I**. The corresponding minimizing value is denoted in Fig. 2 via λ^*.

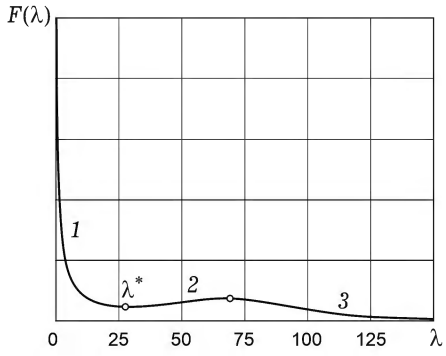


Fig. 2. Schematic graph of typical $F(\lambda)$. Graph segments are marked with smoothening phase numbers.

Remark 1. We excluded the last gradient jump in (12). Note that the boundary layer was assumed to be located inside the last element. If it is inside the first element, we have to exclude the first jump j_1. In general, such an exclusion is necessary to ensure the special shape of the curve $F(\lambda)$, which we actually described above. Excluding the last gradient jump will add the described phase **II** to the smoothing process. Without such an exclusion we will have more shallow drop in $F(\lambda)$ during phase **I**, since the last gradient is relatively large due to the presence of boundary layer. Also phase **II** of the smoothening process will be not present, as if this last gradient will decrease and compensate the other increasing gradients during smoothening the right part of the graph of $u_h^*(x;\lambda)$. The described behavior will change the graph of $F(\lambda)$ to L-shaped, without a clearly identifiable corner, which will make the choice of optimal λ more ambiguous and, therefore, complicated.

Let us define some enough large maximum value $\lambda_{\max} > 0$ for the regularization parameter λ. We can consider the following optimization problem:

given: $\lambda_{\max} > 0$,

find: $\lambda = \lambda^*$ *such that*

$$F(\lambda) \to \text{local}\min \quad \text{on} \quad (0, \lambda_{\max}). \tag{14}$$

The last component we need to have a fully formulated optimization problem is a method for choosing $\lambda_{\max}$. It is well known from standard FEM a priori estimates [18] that to obtain an accurate approximation using uniform meshes in one-dimensional domains we need to have the following relationship between the Péclet number Pe and the number of finite elements n:

$$\text{Pe} = O(n), \tag{15}$$

that is, we may consider $\text{Pe} \approx C \cdot n$ for some constant $C > 0$. Most often, in practice, $C > 1$. Applying this relation to the regularized problem (4) and using expression (7) we get

$$\text{Pe}_{\text{reg}} = (\mu + \lambda)^{-1} \|\boldsymbol{\beta}\|_\infty \operatorname{diam}\Omega \approx C\, n,$$

and, therefore, we can intentionally get a slightly rough estimate:

$$\lambda = \frac{\|\boldsymbol{\beta}\|_\infty \operatorname{diam}\Omega}{Cn} - \mu < \frac{2\|\boldsymbol{\beta}\|_\infty \operatorname{diam}\Omega}{n}.$$

So, we suggest to use

$$\lambda_{\max} = \frac{2\|\boldsymbol{\beta}\|_\infty \operatorname{diam}\Omega}{n}. \tag{16}$$

In the next section we propose a practical algorithm for solving (14) with $\lambda_{\max}$ defined as (16).

It can be noted that the presented approach is somewhat similar to the well-known approach using the so-called L-curve [11] for Tikhonov regularization. The last one is not applicable to our case because it is directly related to problems, when we have classical Tikhonov regularization problem based on least squares distance minimization. Here, for advection-dominated problems we will always have a nonsymmetric problem, so the variational problem will not be related to the corresponding minimization problem.

4. Solving the optimization problem. In this section we present simple algorithm based on bisections method. Still, this algorithm uses some heuristic rules, since the structure of $F(\lambda)$ does not allow to localize the zone near the minimum, in which it is a unique extremum point. The algorithm consists of two steps.

Algorithm 3. *Solving optimization problem.*

1°) Find a point $\lambda_{\text{loc}} \in (0, \lambda_{\max}]$ such that $F'(\lambda_{\text{loc}}) > 0$.

2°) Solve optimization problem (14) on the interval $(0, \lambda_{\text{loc}})$.

Step ***1°*** tries to localize zone around the point of local minimum, in which this minimum is the only point of extremum. This is necessary, because as can be seen from Fig. 2, the local maximum is between zones ***2*** and ***3***. For this step, we use two heuristic procedures. The step ***1°*** can be detailed as follows:

Algorithm 4. *Solving optimization problem* (**Algorithm 3 – Step 1°**).

1°) $\lambda_{\text{loc}} :=$ **ProcSimple**$(\lambda_{\max})$

2°) **if** $\lambda_{\text{loc}} > 0$ **then**

3°) **return** λ_{loc}

4°) **else**

5°) **return ProcRefined**$(\lambda_{\max})$.

In "simple" procedure we use bisections with choosing left intervals to try to find the ascending part of $F(\lambda)$:

Algorithm 5. *Procedure* **ProcSimple** $(\lambda_{\max})$.

Require: $\delta > 0$; **MaxIter** $\in \mathbb{N}$;

1°) **initialization:** $\lambda_{\text{loc}} := \lambda_{\max}$;

2°) **while** $F(\lambda_{\text{loc}} - \delta) > F(\lambda_{\text{loc}} + \delta)$ **and** **MaxIter** > 0

3°) $\lambda_{\text{loc}} := \lambda_{\text{loc}}/2$;

4°) **MaxIter** := **MaxIter** $- 1$;

5°) **if** **MaxIter** ≤ 0 **then**

6°) **return** 0

7°) **return** λ_{loc}.

The refined procedure includes the same steps, but it extends step ***3°*** to take additional "back steps" to the right of current parameter value:

Algorithm 6. *Procedure* **ProcRefined** $(\lambda_{\max})$.

Require: $\delta > 0$; **MaxIter** $\in \mathbb{N}$;

1°) **initialization:** $\lambda_{\text{loc}} := \lambda_{\max}$;

2°) **while** $F(\lambda_{\text{loc}} - \delta) > F(\lambda_{\text{loc}} + \delta)$ **and** **MaxIter** > 0

3°) $\lambda_{\text{back}} :=$ **ProcRefinedBacksteps** (λ_{loc})

4°) **if** $\lambda_{\text{back}} > 0$, **then**

5°) **return** λ_{back}

6°) $\lambda_{\text{loc}} := \lambda_{\text{loc}}/2$;

7°) **MaxIter** := **MaxIter** $- 1$;

8°) **if** **MaxIter** ≤ 0, **then**

9°) **return** 0

10°) **return** λ_{loc}.

Algorithm 7. *Procedure* **ProcRefinedBacksteps** (λ_{loc}).

Require: $\delta > 0$; **MaxIter** $\in \mathbb{N}$;

1°) **initialization:** $k := 1$;

2°) **initialization:** $\lambda_{\text{back}} := \lambda_{\text{loc}}/2$;

3°) **while** $F(\lambda_{\text{back}} - \delta) \ge F(\lambda_{\text{back}} + \delta)$ **and** $k <$ **MaxIter**

4°) $\lambda_{\text{back}} := \lambda_{\text{loc}}\left(1 - \frac{1}{2}\left(\frac{2}{3}\right)^k\right)$;

5°) $k := k + 1$;

6°) **if** $k \ge$ **MaxIter**, **then**

7°) **return** 0

8°) **return** λ_{back}.

Remark 2. Formula $\lambda_{\text{back}} := \lambda_{\text{loc}}\left(1 - \frac{1}{2}\left(\frac{2}{3}\right)^k\right)$ represents bisection, with the right section always chosen for the next step and a midpoint always shifted to the left.

For step ***2°*** of **Algorithm 3** we use standard bisections to find the minimum with a given precision:

Algorithm 8. *Solving optimization problem* (**Algorithm 3 – Step *2°***).

Require: $\delta > 0$; $\text{Tol} > 0$; **MaxIter** $\in \mathbb{N}$;

1°) **initialization:** $a := 0$;

2°) **initialization:** $b := \lambda_{\text{loc}}$;

3°) **while** $b - a \ge \text{Tol}$

4°) $\lambda_{\text{mid}} := (a + b) / 2$;

5°) **if** $F(\lambda_{\text{mid}} - \delta) < F(\lambda_{\text{mid}} + \delta)$, **then**

6°) $b := \lambda_{\text{mid}}$

7°) **else**

8°) $a := \lambda_{\text{mid}}$

9°) **return** 0

10°) **return** λ_{mid} **.**

In the next chapter we demonstrate some numerical results.

5. Numerical experiment. Let us consider a one-dimensional singular perturbed DAR BVP with the following data:

$$\mu = 1, \qquad \beta = 10^4, \qquad \sigma = 10^5, \qquad f = 10^8 \cos 4.5\pi x . \tag{17}$$

Note, that in this problem we have $\text{Pe} = 10^4$.

To solve this problem, we used **Algorithm 3** with 40 linear finite elements and obtained the optimal regularization parameter $\lambda = 105.713$. In Fig. 3 we present four plots. The first (upper) curve corresponds to the solution of the auxiliary Cauchy problem (Fig. 1***a***), the next one to the classical finite element approximation without any regularization used (Fig. 1***b***). The last two plots represent the regularized solutions with the optimal λ with the function $u_0 \equiv 0$ (Fig. 1***c***) and one calculated from the Cauchy problem (Fig. 1***d***), respectively. We can clearly see that the bottommost graph is the most accurate. This example demonstrates the case also mentioned in the article, when for a large value of Pe it may happen that the solution of the Cauchy problem cannot be used as an adequate approximation to the original BVP solution. The third and fourth subplots demonstrate the importance of using a special function u_0, computed as a solution of the auxiliary Cauchy problem compared to the classical Tikhonov regularization.

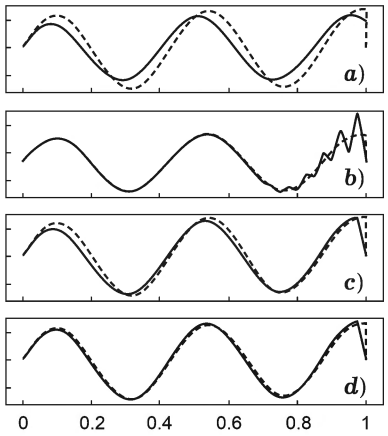


Fig. 3. Results of **Algorithm 3** for DAR model with data (17)**:** ***a***) function u_0 **;** ***b***) the finite element approximations without regularization; ***c***) the finite element approximations with optimal λ and $u_0 = 0$; ***d***) the finite element approximations with optimal λ and u_0 calculated as the solution of the Cauchy problem (3). By dashed lines are denoted the exact solutions.

Remark 3. In this article we focus on qualitative estimation of finite element approximations and do not consider any a priori or a posteriori error estimates. Note, that the stabilized problem (4) actually has a different exact solution than the original variational problem (2). This lead to the conclusion

that to derive some residual-based error estimates [19], we should combine the obtained approximation for (4) with the residual of equation (2).

Remark 4. It is interesting to compare the proposed scheme with typical adaptive algorithms. For example, in h-adaptive schemes [1, 2] we construct a sequence of approximations, starting from one built on some coarse mesh. At each iteration, we make some refinements by adding new degrees of freedom. At each iteration we need to find a finite element approximation, compute a posteriori error estimates for each element based on this, and then refine the mesh. Thus, computational power is mainly consumed by solving a linear system of equations at each iteration on more and more refined mesh. Also, for such an approach, we need an efficient way to store and rearrange the matrix of the linear system and the right-hand vector. In our stabilization technique, we need to solve the linear system several times, but the size of this system will not grow. Also, the matrix at each step will be of the form $A + \lambda M$, where A and M are the matrices of the same sparse structure, so we can compute them only once and simply construct a storage scheme based on some known schemes for sparse matrices to store each pair of elements (A_{ij}, M_{ij}) as one cell in this scheme that provides an efficient procedure for accessing this cell.

6. Computational complexity. Let us estimate the number of atomic operations that **Algorithm 3** can achieve in the average case where a simple localization procedure is successful. In this case, we can consider our algorithm as a sequence of subsequent bisections of the interval $(0,\lambda_{\max})$. Therefore, if we define some tolerance $\varepsilon > 0$ for finding λ, the algorithm stops when iteration number m satisfies the inequality $\lambda_{\max} / 2^m < \varepsilon$. Taking into account that the solution by the FEM in one-dimensional case has the complexity $O(n)$, where n is an element count (and the same complexity of the numerical solution of the Cauchy problem (3) can also be taken into account), we can derive that the total number of elementary operations would be:

$$O\left(n \log_2 \frac{2\|\boldsymbol{\beta}\|_\infty \operatorname{diam}\Omega}{n\varepsilon} \right).$$

7. Generalization to *2D* problems. In order to implement the constructed algorithm for two-dimensional problems we can not simply use the provided framework directly. First of all, we need to provide an appropriate generalization to formula (12). The second thing to consider is the appropriate estimation of $\lambda_{\max}$ for the optimization problem (14) for *2D* problems. The given estimate (16) was derived specifically for one-dimensional problems, since the asymptotic relation (15) for the Péclet number holds only for the one-dimensional case.

We keep the general structure (12) in the form (11) for *2D* problems, that is, we consider $F(\lambda)$ to be an absolute value of the scalar product between some vector representing the oscillating pattern of the non-regularized solution and a vector consisting of some quantities that approximate the Laplacian of the original solution at mesh nodes. As in the one-dimensional case, we exclude from the calculation those nodes that are adjacent to a certain boundary part of the mesh.

To construct a discrete Laplacian using a (generally) irregular mesh we use the cotangent formula [6] which can be derived by manually computing the piecewise-linear solution of Poisson's equation using the FEM. Let us consider a triangular mesh. Consider the indexing of the nodes: $x_1, x_2, \ldots, x_N$. For the ith node and a function u, the approximation of the value $(\Delta u)(x_i)$ of the Laplace operator can be calculated as:

$$\tilde{\Delta}(u,i) := \frac{1}{2A_i} \sum_{j \in N_{\text{mesh}}(i)} (\cot \alpha_{ij} + \cot \beta_{ij})(u(x_j) - u(x_i)),$$

where A_i is one third of the sum of the areas of the adjacent triangles, $N_{\text{mesh}}(i)$ is the set of all adjacent nodes for the ith node, α_{ij} and β_{ij} are the angles opposite to the edge (i,j) (see Fig. 4).

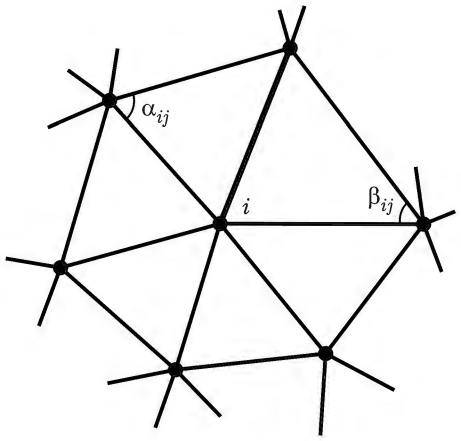


Fig. 4. Components of cotangent formula.

Similar to (16), denote by $u_h^*(x;\lambda)$ the approximation $u_h^*(x)$ computed for some λ. Denote by $M \subset \Omega$ the set of all nodes (points) of the mesh and by $\Gamma_{\text{out}} := \partial\Omega \setminus \Gamma_0$. For some set of node indices S, we define $N_{\text{mesh}}(S) :=$ $:= \bigcup_{j \in S} N_{\text{mesh}}(j)$. If $R \subset \Omega$, then we can extend the last definition to the form:

$N_{\text{mesh}}(R) := x\left[\bigcup_{j:\, x_j \in R} N_{\text{mesh}}(j)\right]$, where $x[S] := \{x_j \mid j \in S\}$.

Let us define a set of nodes $Q := M \setminus (\Gamma_0 \cup N_{\text{mesh}}(\Gamma_{\text{out}}))$, that is, Q contains all internal nodes, except for those nodes that are adjacent to any nodes from Γ_{out}. The concept of excluding these nodes is a direct generalization of the same from one-dimensional case, since we predict that the boundary layer is located near Γ_{out}.

Let us now define the quantity $F_{2D}(\lambda)$, which is a generalization of $F(\lambda)$ to the *2D* case:

$$F_{2D}(\lambda) := \left| \sum_{j \in Q} \operatorname{sgn}\left(\tilde{\Delta}(u_h^*(\,\cdot\,;0), j)\right) \tilde{\Delta}(u_h^*(\,\cdot\,;\lambda), j) \right|$$

where $u_h^*(\,\cdot\,;\lambda)$ is a function of one (first) independent argument and fixed second argument.

$F_{2D}(\lambda)$ has the same form as $F(\lambda)$, that is, it is constructed as a scalar product of the vector $\mathbf{e}_{2D} := \left[\operatorname{sgn}\left(\tilde{\Delta}(u_h^*(\,\cdot\,;0), j)\right)\right]_{j \in Q}$, which represents the oscillating pattern of the original non-regularized solution in the similar manner to the vector $\mathbf{e}$ from (10) and the vector $\mathbf{j}_{2D}(\lambda) := \left[\tilde{\Delta}(u_h^*(\,\cdot\,;\lambda), j)\right]_{j \in Q}$, which represents a measure of local "curvature" and gradient jumps similar to $\mathbf{j}$ in (9).

Experiments have shown that the behavior of the quantity $F_{2D}(\lambda)$ is completely the same as in the one-dimensional case, that is, it has the same

structure as shown in Fig. 2. Due to this fact, we can now use the algorithms described for one-dimensional problems to find the optimal value of λ.

The last thing left is to find $\lambda_{\max}$. For the *2D* case, we use a less accurate estimate. From formula (9) we can get

$$\lambda = \frac{\|\boldsymbol{\beta}\|_\infty \operatorname{diam}\Omega}{\mathrm{Pe}_{\mathrm{reg}}} - \mu < \frac{\|\boldsymbol{\beta}\|_\infty \operatorname{diam}\Omega}{\mathrm{Pe}_{\mathrm{reg}}}. \tag{18}$$

Now we can simply substitute to (18) some (user-defined) fixed small number $\mathrm{Pe}_{\mathrm{user}}$ instead of $\mathrm{Pe}_{\mathrm{reg}}$ for which we know, that FEM with a coarse mesh will give us an adequate non-oscillating approximation. For example, in our experiments we used $\mathrm{Pe}_{\mathrm{user}} = 10$. So, we will have

$$\lambda_{\max} := \frac{\|\boldsymbol{\beta}\|_\infty \operatorname{diam}\Omega}{\mathrm{Pe}_{\mathrm{user}}}.$$

8. Numerical experiment (*2D* problem). Let us consider the following model problem data:

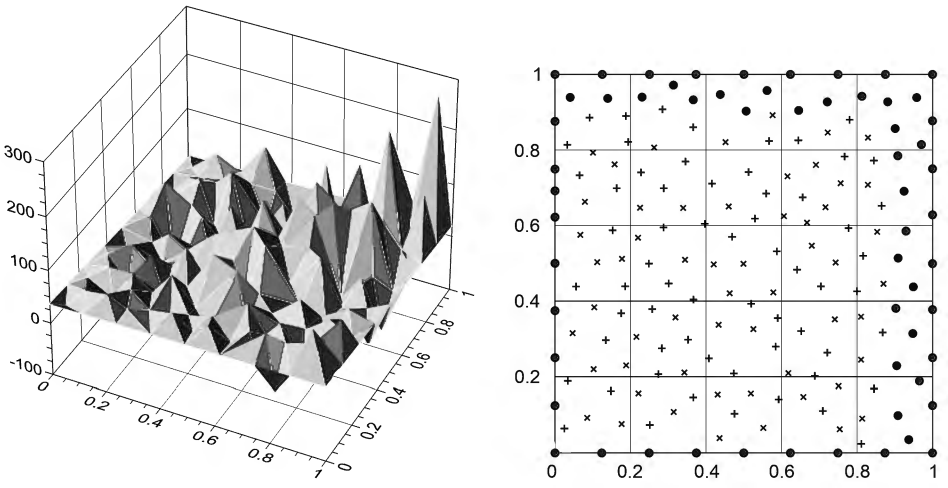
$$\Omega = [0,1]^2, \qquad \mu = 1, \qquad \boldsymbol{\beta}(x,y) \equiv (10^3, 10^3), \qquad \sigma = 10^2,$$

$$f(x,y) = 10^5 \cos(4.5\pi x/2)\cos(4.5\pi y/2).$$

To construct Delaunay triangulations, we used J. R. Shewchuk's library. For this experiment, we construct a mesh of 319 triangles. Fig. 5 shows the non-regularized approximation (Fig. 5***a***) and the corresponding distribution of the components of the vector $\mathbf{e}_{2D}$ (Fig. 5***b***). Dots denote the nodes which are not contained in $\mathbf{e}_{2D}$, "plus" and "cross" markers denote the nodes which correspond to 1 and -1, respectively, in the corresponding components of $\mathbf{e}_{2D}$.

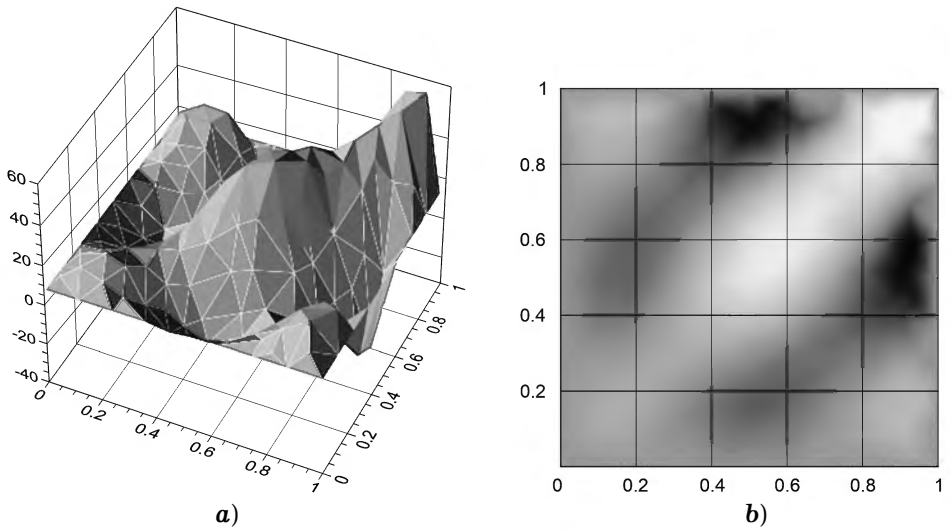
Fig. 6 depicts the regularized approximation after finding the optimal λ according to our algorithm.

Remark 5. The computational complexity in the *2D* case will depend to a greater extent on the algorithm for solving the system of linear equations (SLEs) obtained from the Galerkin method. The number of iterations required to find λ will be $O(\log(\operatorname{diam}\Omega\ \mathrm{Pe}))$.



a) ***b***)

Fig. 5: ***a***) non-regularized approximation; ***b***) corresponding distribution of the components of the vector $\mathbf{e}_{2D}$.



a) *b*)

Fig. 6: *a*) regularized approximation graph; *b*) corresponding heatmap plot. Computed optimal $\lambda = 20.0195$.

9. Possibility of using quantum computing to estimate regularization parameter. The most computationally intensive part of the proposed regularization algorithm is the solution of SLE. We need to solve the SLE several times, but in fact we only need the entire solution at the last step – when we have already calculated the optimal regularization parameter and want to finally solve our problem with this parameter. All previous calls to the linear system solver procedure will be used only to calculate the scalar product of the obtained solution and some fixed vector to obtain the quantity of interest – $F(\lambda)$. Previous observations lead to the conclusion that we can apply the Harrow – Hassidim – Lloyd (HHL) quantum algorithm [12] to solve the SLE and calculate $F(\lambda)$ (with some modification).

The important thing, which we need to modify (to be suitable for certain quantum qomputations) is actually the expression for $F(\lambda)$ (12). Since on quantum register we will operate with quantum state $\left|u_h^*(\lambda)\right\rangle$ which will encode in its amplitudes the actual nodal values of the FEM approximation, it is clear that we need to introduce normalization to $F(\lambda)$. We can rearrange the terms of the sum in the expression for $F(\lambda)$ and rewrite it as $F(\lambda) = \left|(\mathbf{p}, \mathbf{u}_n^*)\right|$ for some constant vector $\mathbf{p}$. Since overall scaling does not affect the distribution of F, we can assume that $|\mathbf{p}| = 1$. For real quantum computing we can estimate the quantity $F_q(\lambda) = \left|\left\langle \mathbf{p} \middle| \mathbf{u}_h^*(\lambda) \right\rangle\right|$ using, for example, the swap test [5]. This quantity does not correspond to F but to the modified quantity $\left|(\mathbf{p}, \mathbf{u}_h^*)\right| / \left|\mathbf{u}_h^*\right|$. Due to the oscillatory nature of the obtained approximations and the described evolution under changing of λ, we can suppose, that this quantity will have the same form, which is also observed in numerical experiments, that is, we can use it in the same manner for our parameter estimation algorithm.

The actual implementation of the described approach for a real quantum computer will not be effective yet, but the general concept is very important for the future. It shows the possibility of using quantum computing and obtaining a potentially exponential increase in the performance of the regularization parameter estimation procedure.

The matrices in our SLEs are sparse, and it is known that we can probably simulate the corresponding unitary evolution operators efficiently, but

even this does not guarantee an efficient implementation with currently available hardware. Also, we need to efficiently prepare arbitrary states for the right-hand side of the SLEs, which requires the generation of the appropriate quantum circuits. In [17] it was shown that a real implementation of the full procedure of solving SLE obtained from FEM for Maxwell's equation lead to a very large number of used quantum gates. For example, the authors constructed a circuit to solve the SLE of size $N = 332\,020\,680$ and the actual total width and depth of the quantum circuit were of order 10^8 and 10^{29}, respectively. In real quantum computing you will also need to include some schemas for error correction to ensure stable error-tolerant computations. Also, taking into account the probabilistic nature of quantum computing, we will need to run the computations several times to estimate $\left\langle \mathbf{p} \middle| u_h^*(\lambda) \right\rangle$ using the swap test. These facts show that an actually usable implementation can not be constructed at least for now, but due to asymptotic estimates showing an exponential speedup of HHL compared to classical algorithms for solving SLEs, we may be able to exploit this power in the future.

Remark 6. One note regarding implementations of the unitary operators e^{iAt} (which we need in the HHL algorithm), where A is the (extended) matrix of the system (we assume that we have already extended the matrix to be Hermitian, as suggested in [12]). For our SLEs we will have a bit special case, because all matrices are of the form $A + \lambda M$. The matrices A and M are fixed. In general, $[A, M] \neq 0$. In such a case, we can probably generate circuits for simulating e^{iAt} and e^{iMt} only once, and then, taking into account Trotter's formula [15], combine both circuits in the actual computation.

Conclusions. In this article we constructed a heuristic procedure for finding the optimal value of the regularization parameter for a stabilized finite element scheme, based on a combination of Tikhonov-type regularization and the auxiliary Cauchy problem. We analyzed the behavior of perturbations in the approximate solution with respect to the change of the regularization parameter. Based on this, we derived a local minimization problem for the suitable loss function constructed as a composition of a certain linear functional and the corresponding finite element approximation. The obtained function acts as an indicator of high-frequency parasitic oscillations, which we want to eliminate from the final approximation. We proposed a practical heuristic method for finding the local minimum point of the constructed loss function. In this work we did not consider any a priori and a posteriori error estimates. The construction and investigation of such estimates is the topic for further research. The proposed approach is developed for one-dimensional problems and then is generalized for *2D* problems. For both cases numerical results are presented. We also note that the proposed approach is suitable for partial implementation on quantum computers by using the Harrow – Hassidim – Lloyd quantum algorithm together with the swap test to implement the computation of the obtained loss function.

1. *Козаревська Ю., Шинкаренко Г.* Регуляризація чисельних розв'язків варіаційних задач міграції домішок: h-адаптивний метод скінченних елементів. Ч. I // Вісн. Львів. ун-ту. Сер. Прикл. математика та інформатика. – 2002. – Вип. 5. – С. 153–164.
2. *Трушевський В. М., Шинкаренко Г. А., Щербина Н. М.* Метод скінченних елементів і штучні нейронні мережі. Теоретичні аспекти і застосування. – Львів, Вид. центр Львів. нац. ун-ту ім. І. Франка, 2014. – 396 с.
3. *Bartels S.* Numerical approximation of partial differential equations. – Cham: Springer, 2016. – xv+535 p.
4. *Brenner S. C., Scott L. R.* The mathematical theory of finite element methods. – New York: Springer, 2008. – xviii+400 p.
5. *Buhrman H., Cleve R., Watrous J., de Wolf R.* Quantum fingerprinting // Phys. Rev. Lett. – **87**, No. 16. – Art. 167902.

–https://doi.org/10.1103/PhysRevLett.87.167902.
6. *Crane K., de Goes F., Desbrun M., Schröder P.* Digital geometry processing with discrete exterior calculus // Proc. of SIGGRAPH '13: ACM SIGGRAPH 2013 Courses. – New York, 2013. – Article No. 7. – P. 1–126. – https://doi.org/10.1145/2504435.2504442.
7. *Drebotiy R. G., Shynkarenko H. A.* On the application of the one hp-adaptive finite element strategy for nonsymmetric convection – diffusion – reaction problems // Журн. обчисл. прикл. математики. – 2017. – № 3(126). – С. 48–60.
8. *Drebotiy R., Shynkarenko H.* Regularized finite element method for singular perturbed convection–diffusion–reaction models with nonuniform sources // Вісн. Львів. ун-ту. Сер. Прикл. математика та інформатика. – 2021. – Вип. 29. – С. 27–36. – http://doi.org/10.30970/vam.2021.29.11330.
9. *Feng X., Karakashian O., Xing Y.* Recent developments in discontinuous Galerkin finite element methods for partial differential equations. – Ser.: The IMA Volumes in Mathematics and its Applications. – Vol. 157. – Springer, 2014. – 279 p.
10. *Fritz J.* Partial differential equations. – New York: Springer, 2012. – ix+198 p.
11. *Hansen P. C.* The L-curve and its use in the numerical treatment of inverse problems // Computational inverse problems in electrocardiology / Johnston P. R. (Ed.). – WIT Press, 2001. – P. 119–142.
12. *Harrow A. W., Hassidim A., Lloyd S.* Quantum algorithm for linear systems of equations // Phys. Rev. Lett. – 2009. – **103**, No. 15. – Art. 150502. – https://doi.org/10.1103/PhysRevLett.103.150502.
13. *Lenzen F., Scherzer O.* Tikhonov type regularization methods: History and recent progress // Proc. of ECCOMAS 2004: European Congress on Computational Methods in Applied Sciences and Engineering, 2004. – 21 p.
14. *Logan J. D.* Transport modeling in hydrogeochemical systems. – New York: Springer, 2001. – 226 p.
15. *Nielsen M. A., Chuang I. L.* Quantum computation and quantum information. – Cambridge: Cambr. Univ. Press, 2010. – 702 p. – https://doi.org/10.1017/CBO9780511976667.
16. *Rektorys K.* Variational methods in mathematics, science and engineering. – Dordrecht: D. Reidel Publ. Co., 1980. – 589 p.
17. *Scherer A., Valiron B., Mau S.-C., Alexander S., van den Berg E., Chapuran T. E.* Concrete resource analysis of the quantum linear system algorithm used to compute the electromagnetic scattering cross section of a 2D target // Quantum Inf. Process. – 2017. – **16**, No. 60. – 65 p. – https://doi.org/10.1007/s11128-016-1495-5.
18. *Stewart J. R., Hughes T. J. R.* A tutorial in elementary finite element error analysis: a systematic presentation of a priori and a posteriori error estimates // Comput. Meth. Appl. Mech. Eng. – 1998. – **158**, No. 1-2. – P. 1–22. – https://doi.org/10.1016/S0045-7825(97)00230-2.
19. *Verfürth R.* Adaptive finite element methods. – Ruhr-Universität Bochum: Lecture Notes Winter Term 2018/19. – 129 p.

ЕВРИСТИЧНИЙ ВИБІР ПАРАМЕТРА РЕГУЛЯРИЗАЦІЇ ДЛЯ ОПТИМАЛЬНОЇ СТАБІЛІЗАЦІЇ АПРОКСИМАЦІЙ МЕТОДУ СКІНЧЕННИХ ЕЛЕМЕНТІВ

Розглянуто задачу оптимального вибору параметра регуляризації у схемі стабілізації методу скінченних елементів для сингулярно збурених задач дифузії – адвекції – реакції. Стабілізація базується на поєднанні регуляризації Тихонова з допоміжною задачею Коші. Проаналізовано поведінку збурень наближеного розв'язку щодо зміни параметра регуляризації. На основі проведеного аналізу побудовано евристичний критерій оптимального вибору параметра регуляризації. Критерій формулюється як локальна задача мінімізації відповідної функції, побудованої у вигляді композиції лінійного функціонала та отриманої скінченно-елементної апроксимації. Запропонований підхід розроблено для одновимірних задач, а потім узагальнено для двовимірних. Також показано можливість використання квантового алгоритму Гарроу – Гассидима – Ллойда у поєднанні зі swap-тестом для реалізації обчислення отриманої функції втрат на квантовому комп'ютері.

Ключові слова: *метод скінченних елементів, модель дифузії – адвекції – реакції, схеми стабілізації, задача Коші, дискретний оператор Лапласа, алгоритм Гарроу – Гассидима – Ллойда.*

Ivan Franko National University of Lviv, Lviv

Received 01.02.23