

Algorithms Composition Approach Based on Difference Potentials Method for Parabolic Problems

Yekaterina Epshteyn¹

Abstract. In this work we develop an efficient and flexible *Algorithms Composition Approach* based on the idea of the difference potentials method (DPM) for parabolic problems in composite and complex domains. Here, the parabolic equation serves both as the simplified model, and as the first step towards future development of the proposed framework for more realistic systems of materials, fluids, or chemicals with different properties in the different domains. Some examples of such models include the ocean-atmosphere models, chemotaxis models in biology, and blood flow models. Very often, such models are heterogeneous systems - described by different types of partial differential equations (PDEs) in different domains, and they have to take into consideration the complex structure of the computational subdomains. The major challenge here is to design an efficient and flexible numerical method that can capture certain properties of analytical solutions in different domains, while handling the arbitrary geometries and complex structures of the subdomains. The Algorithms Compositions principle, as well as the Domain Decomposition idea, is one way to overcome these difficulties while developing very efficient and accurate numerical schemes for the problems. The *Algorithms Composition Approach* proposed here can handle the complex geometries of the domains without the use of unstructured meshes, and can be employed with fast Poisson solvers. Our method combines the simplicity of the finite difference methods on Cartesian meshes with the flexibility of the Difference Potentials method. The developed method is very well suited for parallel computations as well, since most of the computations in each domain are performed independently of the others.

AMS subject classification: 65M06, 65M22, 65M55, 65M70, 35K05, 35K20

Key words: Parabolic equation, heat equation, Calderon's potentials, Calderon's boundary equations with projections, difference potentials methods, finite difference, Cartesian meshes, complex and composite domains, curvilinear boundaries, algorithms composition, and domain decomposition.

1. Introduction

In this work we develop an efficient and flexible *Algorithms Composition Approach* based on the idea of the Difference Potentials Method (DPM) for parabolic problems in composite and complex domains.

Here, a parabolic equation (2.1) serves both as the simplified model, and as the first step towards future development of the proposed scheme for more realistic systems of materials, fluids, or chemicals with different properties in the different domains (or in the different parts of the domains). Some examples of such models include the ocean-atmosphere models, chemotaxis models in biology, and blood flow models (see for example [2, 36, 53, 10, 9, 43, 42]).

Numerical approximations and modeling of many physical, biological, and biomedical problems often deal with heterogeneous models (described by different types of partial differential equations (PDEs) in different domains), and/or they have to take into consideration the complex structure of the computational subdomains. The major challenge here is to design an efficient and flexible numerical method that can capture certain properties of analytical solutions in different domains/subdomains (such as positivity, different regularity/smoothness of the solutions in the domains/subdomains, etc), while handling the arbitrary geometries and complex structures of the domains. *The Algorithms Compositions principle, as well as the Domain Decomposition idea, is one way to overcome these difficulties while developing very efficient and accurate numerical schemes for the problems.* This methodology can be used within any discretization for PDEs (such as finite differences, finite volumes, finite elements, or spectral methods). It provides great opportunities to subdivide problems into subproblems, and to design the most suitable numerical approximation for each of them independently. After that, one can compose the problems and algorithms together by imposing some interface conditions. Such schemes can be used for parallel computations as well, since most of the computations in each subdomain are performed independently of the others (see for example, [41, 51]).

There is an extensive literature that addresses problems in domains with complex geometries and interface problems. We will briefly discuss below some established finite difference methods for such problems. For more detailed review on the subject the reader can consult for example [25]. The immersed boundary method (IB) was originally proposed by Peskin to model blood flow in a human heart (see for example [38, 39]). One of the essential ideas of the IB method is to employ a discrete delta function to place/spread a singular source to neighboring mesh points. The IB method is simple and efficient but in most cases is a low-order (first-order) method. The higher-order (second order) version of the IB method has been recently proposed in [19]. The IB method has been applied to many problems in computational fluid dynamics and mathematical biology (see for example [37, 40, 12, 54]), and it has been parallelized [33].

¹Department of Mathematics, The University of Utah, Salt Lake City, UT, 84112, epshteyn@math.utah.edu

The immersed interface method (IIM) is designed for interface problems and problems defined on irregular domains [23, 24, 25]. This method is a sharp interface method for PDEs that can have discontinuities in the coefficients, the solution and its derivatives, and it can handle Dirac singularities in the source terms. The IIM is based on Cartesian meshes with second order or fourth order (for some problems) accuracy. Standard finite difference or some standard finite element schemes can be employed as the core discretization in IIM. The IIM modifies these schemes near or at the interfaces/boundaries through the interface relations so that second order/or fourth order accuracy can be achieved in the whole domain. However, some of the difficulties with IIM schemes are the requirement for the explicit knowledge of the jump conditions at the interfaces and boundaries for the development of IIM (in order to derive the correction terms for the schemes at the mesh points near the interface), and that the parallelization of IIM is not a trivial task due to global coupling of the solutions in the subdomains. The IIM is used for many problems such as Stefan problems, incompressible Stokes and Navier-Stokes flow problems, etc, (see for example [25]). The other sharp interface method is the ghost fluid method (GFM). This method was originally proposed for the accurate approximation of the boundary conditions for hyperbolic systems [13], and was later developed in [26] (the convergence was proved in [27]) for the elliptic interface problems. The GFM is based on a Cartesian grid finite-difference method. Similar to IIM, the main idea of GFM [26] is to incorporate the jump conditions into the finite difference scheme. However, the idea of GFM is to decompose the flux jump in dimension by dimension fashion which may decrease the accuracy of the method in some cases. The main advantages of GFM are its second-order accuracy, simplicity, and ease of implementation. Some shortcomings of GFM are that fast Poisson solvers cannot be employed with the method, since the coefficients of the finite difference discretization are modified at the mesh points near the boundaries; the GFM may produce only a first-order accurate solution for more general boundary conditions (for example for mixed boundary conditions). The GFM has been applied to several problems including the simulation of incompressible flame, incompressible multiphase flows, etc, and it has been applied to complex domain problems with Dirichlet boundary conditions (see for example [35, 20, 15]). Let us comment that in [15], the ghost values across the interface were defined using dimension by dimension extrapolation, and the overall second order accuracy was achieved for the complex domain problems with Dirichlet boundary conditions (instead of explicit information about the jump conditions across the interface). Also, based on the ideas from [15], a fourth-order scheme has been developed in [14] for Dirichlet boundary conditions on domains with complex geometry.

Let us now mention the method based on the integral equations approach. In [30], the fast and high-order methods were developed for solving Laplace's and biharmonic equations on complex domains with smooth boundaries in 2D. The idea of the method proposed in [30] is to combine integral equations based on the single and double layer theory with finite difference method to solve a Laplace's equation on a complex domain. In this method the domain is embedded in a computationally simple region (auxiliary domain) where a fast Poisson solver can be used on a uniform mesh. The right-hand side of the original equation is modified appropriately in the auxiliary domain using the information about the values of the extended solution at the mesh points near (inside and outside) the original continuous boundary (at the irregular mesh points). The approximations to the unknown values of the extended solution at these irregular mesh points are constructed through Taylor expansion and the solution of the Fredholm integral equation of the second kind. Finally, the approximation of the solution in the original region is obtained by the use of the fast Poisson solver in the simple auxiliary domain. Based on these ideas, the method introduced in [30] avoids the common difficulty with the solution of the integral boundary equations (for example problems near the original boundary). This approach has been parallelized in [29, 17]. The method can be accelerated if coupled with a fast multipole method [32]. The possibility of extension to elliptic problems with variable coefficients is mentioned in [31]. Note that, in general, the methods based on integral equations approaches are very efficient for homogeneous source terms and for the specific type of boundary conditions (see also discussion below on Difference Potentials Method and Boundary Element Method).

Similar to the method in [30], the idea of our method here and in [50, 48] is to first introduce a computationally simple auxiliary domains. After that, the original domains/subdomains are embedded into simple auxiliary domains (and the auxiliary domains are discretized using Cartesian meshes). However, compared to the integral approach in [30], we construct discrete *Boundary Equations* with projections (discrete generalized Calderon's boundary equations with projections) to obtain the values of the solutions at the points near the continuous boundaries of the original domains (at the points of the discrete grid boundaries which approximate the continuous boundaries from the inside and outside of the domains). Using the obtained values of

the solutions at the discrete grid boundaries, the approximation to the solutions in each domain/subdomain is constructed through the discrete generalized Green's formulas. The main complexity of our approach reduces to the several solutions of simple auxiliary problems on structured Cartesian grids, and similar to [30] the solutions of these auxiliary problems can be combined with fast Poisson solvers. Like method in [30], and IIM and GFM, our method (and in general, methods based on Difference Potentials Method [49, 34]) preserve the underlying accuracy of the schemes being used for the space discretization of the continuous PDEs in each domain/subdomain (here, and in [50, 48] we considered second-order finite difference scheme for the space approximation). But compared to [30], and to IIM and GFM, our approach is not restricted by the type of the boundary or interface conditions (as long as the continuous problems are well-posed). Let us mention that the accuracy of our approach is confirmed by several numerical experiments in the current paper (see Section 5) and in papers [50, 48]. The reader can consult [49] for the theoretical convergence study of the methods based on Difference Potentials.

In this work, we will consider the heat equation (2.5) in a composite domain with curvilinear smooth boundaries in 2D (even though, the proposed framework is general and can be extended in the future to the arbitrary 1D, 2D and 3D domains). We will further develop, as well as numerically test the *Algorithms Composition Scheme* proposed originally in [50, 48] for linear elliptic problems. The *Algorithms Composition Approach* developed in this paper is an accurate, simple, and robust scheme of algorithms composition for the numerical approximations of the boundary value problems in composite and complex domains. The proposed method can handle complex geometries without the use of unstructured meshes (with the consideration of only regular Cartesian grids), and can be employed with fast Poisson solvers. Our method combines the simplicity of the finite difference methods on Cartesian meshes with the flexibility of the Difference Potentials method [49].

Difference Potentials Method (DPM) can be understood as the discrete version of the method of generalized Calderon's potentials and Calderon's boundary equations with projections in the theory of PDEs. The DPM on its own, or in combination with other numerical methods, is an efficient tool for the numerical solution of the interior and exterior boundary value problems in arbitrary domains (see for example, [49, 46, 28, 47, 52, 34, 50, 48, 10, 9]). Viktor S. Ryaben'kii originally introduced DPM in his Doctor of Science thesis (Habilitation thesis) in 1969. The DPM allows one to reduce uniquely solvable and well-posed boundary value problems into pseudo-differential boundary equations. In some respect, the difference potentials method (DPM) is related in spirit to the boundary element method (BEM). The idea of the BEM (see for example [1, 8]) is to reduce the boundary value problem to Fredholm-type integral equations with respect to equivalent boundary sources, and these equations are discretized accordingly. One shortcoming of the BEM is the full structure of the resulting systems/matrices (as opposed to the sparse nature of the systems/matrices produced by the finite differences (FD), or finite element methods (FEM)). Recent progress on fast multipole methods considerably accelerated solutions of such full systems [18]. However, the most serious drawback of BEM methods is the requirement for the explicit knowledge of the fundamental solution of given differential operators. This can impose several restrictions on the practical applications of BEM methods. The essence of the DPM is to transform uniquely solvable and well-posed boundary value problems into so-called pseudo-differential boundary equations that do not employ a fundamental solution. Hence, on one hand, the DPM enjoys geometric flexibility of the BEM. On the other hand, it can be applied to a wider class of problems than the BEM, and it can have several advantages over the BEM. We will make a few comments about this below:

Main features and advantages of the DPM (see also Section 2 in this paper and [49, 34] for more detailed discussion):

- (i) The original PDEs (without imposed boundary conditions) are reduced to an *equivalent generalized Calderon's boundary equations with projections*. These equations are supplemented by the given boundary conditions. Compared to BEM, DPM does not employ Fredholm equations of the first or second kind;
- (ii) The derived Calderon's problem can be discretized on structured Cartesian grids. Discrete inverse operators which are introduced in the DPM for the approximation of the Calderon's potentials and projections do not contain any singularities or convolutions. These inverse operators can be obtained using fast numerical calculations as the solution of the simple and computationally efficient auxiliary problems [49, 34, 50, 48, 10, 9];
- (iii) DPM can treat arbitrary smooth boundaries of the domains, and the boundaries do not need to align/conform with the grid - this does not produce any loss of accuracy. The DPM provides flexibility to

handle general boundary conditions in an efficient and universal way, and the method always produces a well-posed discrete version of the problem (if the original continuous boundary value problem is well-posed) [49, 34, 3]. Note that well posedness for BEM may not be a trivial task for problems with general boundary conditions;

(iv) DPM gives flexibility to construct high-order schemes on regular Cartesian grids for problems with complex geometries [49, 34, 3]. Some of the advantages of numerical methods on Cartesian grids are that the grid generation for these schemes is trivial, the design of the high-order numerical methods that satisfy certain stability properties are usually much more straightforward on regular structured grids, and the numerical methods on Cartesian meshes are more robust than those of body-fitted grids;

(v) DPM can approximate both variable coefficients and constant coefficients problems: the main steps in the construction of the Calderon's potentials and projections for variable coefficients and constant coefficients stay essentially the same [49, 34, 50, 48, 10, 9].

The developed Algorithms Composition Framework combines the above advantages of the DPM and offers novel flexibility to the DPM (for a more detailed discussion, see Sections 3 - 6 in this paper, as well as [50, 48]):

(i) the Algorithms Composition Framework is well-suited for heterogenous problems and complex interface problems, as well as for the development of the adaptive schemes and domain decomposition approaches;

(ii) our method provides the flexibility to consider a non-difference approximation of the general boundary and interface (/or matching) conditions, which automatically takes into account the smoothness of the solution. Such self-tuning is impossible, for example in the difference or finite-element approximations of these conditions;

(iii) the proposed framework can be further developed for the efficient computation of the solutions in arbitrary domains with arbitrary boundaries and interfaces;

(iv) the numerical schemes, as well as meshes can be chosen totally independently for each subdomain/domain; the boundaries of the subdomains and interfaces do not need to conform/align with the grids;

(v) the main complexity of the developed algorithm reduces to the several solutions of simple auxiliary problems on structured Cartesian grids;

(vi) *the proposed approach is general and can be developed as a high-order method (both in time and space: higher than first order in time and higher than second order in space).* For example, by considering high-order methods such as high-order finite difference, finite element, or spectral methods for the construction of the discrete parts of generalized Calderon's potentials, as well as for the approximations of the particular solutions to the inhomogeneous equations;

(vii) since the schemes for constructing the solutions in each domain/subdomain are independent of each other, our approach is very well suited for parallel computations;

(viii) the proposed method is not restricted to 2D and can be applied to variable coefficients problems. The method can be generalized to the equations not necessarily of the elliptic/parabolic type as well.

The paper is organized as follows. First, in Section 2 we give some preliminaries that will be helpful for the introduction of the proposed algorithm. Then, we introduce the idea of DPM for the parabolic equation in the single domain, and we make an overview of the important properties of the DPM for the model under consideration in Section 2.1. In Section 3, we develop the *Algorithms Composition Scheme* for the parabolic model in a composite domain using the flexibility of the DPM. In Section 4, we state the main steps of the proposed algorithm. Finally, we illustrate the flexibility and performance of the proposed scheme in several numerical experiments in Section 5. Some concluding remarks are given in Section 6, and some technical details about the algorithm are provided in Appendix Section 7.

2. Preliminaries

In this section, we will introduce some preliminaries that will be helpful for the discussions in the following sections. We are concerned in this paper with the parabolic initial and boundary value problem (IVP) in some bounded domain $\bar{\Omega} \subset \Omega^0 \subset \mathbb{R}^2$, and its neighborhood Ω^0 over the time interval $[0, T]$.

$$\frac{\partial u}{\partial t} + Lu = q \text{ in } \Omega \times (0, T), \quad (2.1)$$

$$l(u) = \psi \text{ on } \partial\Omega \times (0, T), \quad (2.2)$$

$$u|_{t=0} = u_0 \text{ in } \Omega. \quad (2.3)$$

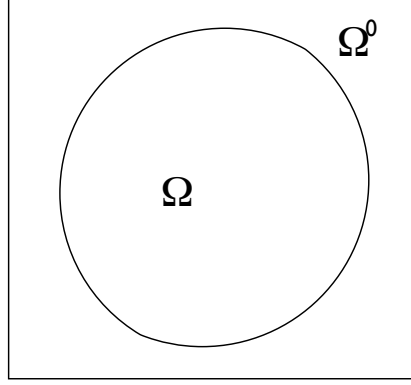


Fig. 2.1: Example (a sketch) of the original domain Ω and auxiliary domain Ω^0

Here, $q(x_1, x_2, t)$ is the sufficiently regular source function in the domain Ω^0 (function q is originally given $\bar{\Omega}$, and extended to the larger domain Ω^0), and L is the second order linear symmetric elliptic differential operator

$$Lu := - \sum_{k,j=1}^2 \frac{\partial}{\partial x_k} (a_{kj}(\mathbf{x}) \frac{\partial u}{\partial x_j}) + a_0 u, \quad (2.4)$$

where coefficients a_0 and a_{kj} are assumed to be sufficiently smooth functions in Ω^0 , $a_{kj}(\mathbf{x}) = a_{jk}(\mathbf{x})$, $(k, j) = 1, 2$ for almost every $\mathbf{x} := (x_1, x_2) := (x, y) \in \Omega^0$. Moreover, there exists a constant $\alpha_0 > 0$ such that $\sum_{k,j=1}^2 a_{kj}(\mathbf{x}) \xi_k \xi_j \geq \alpha_0 |\xi|^2$ for each $\xi \in \mathbb{R}^2$, and for almost every $\mathbf{x} \in \Omega^0$. The classical example of the above parabolic problem (2.1) is the heat equation

$$\frac{\partial u}{\partial t} - \Delta u = q. \quad (2.5)$$

To simplify the presentation of the ideas, we will be concerned below with the heat equation (2.5) subject to (2.2) - (2.3). However, let us mention that the same idea can be extended in a straightforward way to a general parabolic equation (2.1).

Next, at the given time t , we denote by \mathbf{v}_Γ the Cauchy data of an arbitrary continuous piecewise smooth function $v(x, y, t)$ defined on Γ and in some of its neighborhoods:

$$\mathbf{v}_\Gamma := \left(v \Big|_\Gamma, \frac{\partial v}{\partial n} \Big|_\Gamma \right). \quad (2.6)$$

Here, $\frac{\partial}{\partial n}$ is the inward (with respect to Ω) normal derivative to Γ , and $\Gamma := \partial\Omega$ is a piecewise smooth boundary of Ω .

Now, in order to introduce our ideas, let us consider model problem (2.5) in its time-discrete form. Towards this end, we subdivide time interval $[0, T]$ into N_t time steps and denote by $t^i := i\Delta t$, $\Delta t > 0$. Setting $m := \frac{1}{\Delta t}$ and considering the Backward Euler method, we obtain below the following time discrete reformulation of the parabolic equation (2.5) in some domain Ω^0

Let us introduce

$$L_{\Delta t}[u^{i+1}] := \Delta u^{i+1} - m u^{i+1}, \quad (x, y) \in \Omega^0, \quad (2.7)$$

where $L_{\Delta t}$ denotes the linear elliptic operator applied to u^{i+1} . We also denote the right-hand side as:

$$G_{\Delta t}^u := -q^{i+1} - m u^i, \quad (x, y) \in \Omega^0, \quad (2.8)$$

where $q^{i+1} := q(x, y, t^{i+1})$.

Then the semi-discrete formulation of the model (2.5) is stated as follows:
Find some $u^{i+1} := u^{i+1}(x, y) \approx u(x, y, t^{i+1})$ such that

$$L_{\Delta t}[u^{i+1}] = G_{\Delta t}^u, \quad (x, y) \in \Omega^0, \quad (2.9)$$

Remark: For each i this is a time-independent elliptic problem.

Let us now introduce the auxiliary problem. For the purposes of the discussion below, we will suppress for now the explicit dependence on the time level i . We place the original domain Ω in the auxiliary domain $\Omega^0 \subset \mathbb{R}^2$: $\bar{\Omega} \subset \Omega^0$. Next, we will formulate a discrete in time and continuous in space *Auxiliary Problem (AP)*:

DEFINITION 2.1. *For any given sufficiently regular right-hand side $G_{\Delta t}$, find F such that,*

$$L_{\Delta t}[F] = G_{\Delta t}, \quad (x, y) \in \Omega^0, \quad (2.10)$$

$$F = 0, \quad (x, y) \in \partial\Omega^0, \quad (2.11)$$

where $L_{\Delta t}$ is the same linear elliptic operator as in (2.9) and is applied here to F . Since the auxiliary domain Ω^0 can be arbitrary, we can choose it to be a square. The above (AP) Dirichlet problem is uniquely solvable.

Remark: This is a continuous in space Dirichlet problem for each fixed time level.

Let us now construct a potential with a density \mathbf{v}_Γ . Define the vector function

$$\mathbf{v}_\Gamma := (\phi^0(s), \phi^1(s)), \quad (2.12)$$

where $\phi^0(s)$ and $\phi^1(s)$ are two piecewise smooth continuous functions on $|\Gamma|$ that are s -periodic with a period of $|\Gamma|$, s is the arc length along $|\Gamma|$, and $|\Gamma|$ is the length of the boundary. Here, the arc length is chosen as a parameter only for definiteness.

Let $v(x, y) = v_{\Omega^0}$ be an arbitrary sufficiently smooth function on Ω^0 that satisfies condition (2.11) on $\partial\Omega^0$: $v|_{\partial\Omega^0} = 0$. Assume that its Cauchy data \mathbf{v}_Γ is defined as in (2.6) and is equal to the vector function \mathbf{v}_Γ in (2.12). Then, we can recall the following definition of the potential $P_{\Omega\Gamma}\mathbf{v}_\Gamma$ below [48, 49].

DEFINITION 2.2. *A Potential $u_\Omega := P_{\Omega\Gamma}\mathbf{v}_\Gamma$ defined on Ω with density \mathbf{v}_Γ is equal on Ω to the solution of (AP, Def. 2.1), with the right hand-side $G_{\Delta t}$ defined as follows:*

$$G_{\Delta t} := \begin{cases} 0, & (x, y) \in \Omega, \\ L_{\Delta t}[v], & (x, y) \in \Omega^0 \setminus \Omega \end{cases} \quad (2.13)$$

It can be shown that at each time level the above potential $u_\Omega := P_{\Omega\Gamma}\mathbf{v}_\Gamma$ is well-defined: it depends only on Cauchy data (2.12), but is independent of the choice of a particular function $v(x, y)$ satisfying (2.11) on $\partial\Omega^0$ whose Cauchy data coincides with (2.12). For a more detailed discussion on the potentials with projectors, see [48, 49].

Remark: The potential $u_\Omega := P_{\Omega\Gamma}\mathbf{v}_\Gamma$ can be viewed as the modification [45] of the Calderon potential [4]. However, in comparison to the Calderon potential, the potential $P_{\Omega\Gamma}\mathbf{v}_\Gamma$ admits a finite-dimensional constructive approximation by Difference Potentials [49, 10, 9], as will be illustrated below in Section 2.1.

2.1. Scheme Based on the Difference Potentials

We will develop our *Algorithms Composition Approach* based on the idea of the Difference Potentials Method (DPM) [49, 48, 50, 10, 9]. Difference Potentials Method (DPM) can be viewed as the method of building and computing the discrete parts of the modified Calderon's potentials (see for example, [49, 28, 47, 52, 50, 48, 10, 9] and Introduction Section 1).

We will present in this Section the necessary preliminaries and some overview of the numerical scheme based on the difference potentials for the single arbitrary domain $\Omega \in \mathbb{R}^2$. This discussion will be important for the development of our approach for the composite/complex domain $\Omega \in \mathbb{R}^2$, which we will present in Sections 3 - 5.

At this point, let us assume that we consider (2.5) in some domain Ω - an arbitrary bounded domain in \mathbb{R}^2 with the boundary $\partial\Omega$. First, let us introduce some preliminary notations and definitions that will

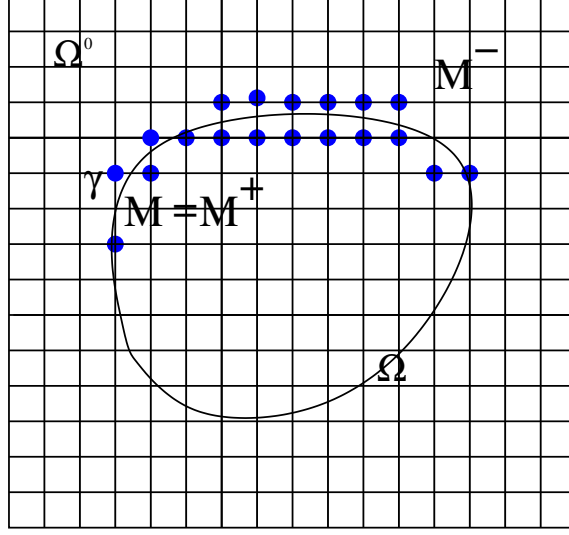


Fig. 2.2: Example (a sketch) of the auxiliary domain Ω^0 , original domain Ω ; the example of some points (x_j, y_k) in the set γ : the points which are outside Ω are from γ_{ex} , the points which are inside Ω are from $\gamma_{in} \in M$

be used in this section. Denote Π_{SRVR} as the extension operator of function v from set/domain R to the set/domain S , π_S as the restriction operator to the set/domain S , $w_S := \pi_S w$ as the restriction of function w to the set/domain S , and χ_S as the characteristic function of the set S .

Next, we introduce here the auxiliary fully discrete problem. Let us place the original domain Ω in the auxiliary domain $\Omega^0 \subset \mathbb{R}^2$. The choice of the domain Ω^0 should be convenient for the computations, so we will choose it to be a square, and we will introduce a Cartesian mesh for Ω^0 , with points $x_j = j\Delta x, y_k = k\Delta y$ ($k, j = 0, \pm 1, \dots$). Let us assume for simplicity that $\Delta x = \Delta y := h$. Let us also define a five-point stencil $N_{j,k}$ with its center placed at (x_j, y_k) to be the set of the following points: $N_{j,k} := \{(x_j, y_k), (x_{j\pm 1}, y_k), (x_j, y_{k\pm 1})\}$.

In addition, we also introduce point sets $M := M^+ := (x_j, y_k) \in \Omega$ - the sets of all the points (x_j, y_k) that belong to the interior of the original domain Ω . We now define $N := N^+ := \{\bigcup_{j,k} N_{j,k} | (x_j, y_k) \in M\}$ - the set of all points covered by five-point stencils when center point (x_j, y_k) of the stencil goes through all the points of the set M . Note that the points in the set N will be both inside and outside of the original domain Ω .

Now, let us introduce the grid boundaries $\gamma_{ex} := \gamma_{ex} = N \setminus M$ is the exterior grid boundary layer for domain Ω . $\gamma_{in} := \{(x_j, y_k) | (x_j, y_k) \in M : N_{j,k} \not\subset M\}$ is the interior grid boundary layer for domain Ω (in other words, this is the set of all the nodes (x_j, y_k) in M for which stencil $N_{j,k}$ is not the subset of M : $N_{j,k}$ contains nodes outside of M , but the center point (x_j, y_k) of the stencil $N_{j,k}$ belongs to M). Define $\gamma := \gamma_{ex} \cup \gamma_{in}$ - a narrow set of nodes that surrounds the continuous boundary $\partial\Omega$, Figure 2.2.

Next, we construct the auxiliary set M^1 by completing the set N to a rectangle, and adding one extra layer of grid points to each side of the rectangle, hence $N \subset M^1$. Also, as before, define $N^1 := \{\bigcup_{j,k} N_{j,k} | (x_j, y_k) \in M^1\}$, and finally, let us introduce $\gamma^1 := N^1 \setminus M^1$.

We can now introduce the space approximation of (2.9) and consider the fully discrete version of equation (2.5). Therefore, the computed quantity will be the point values $u_{j,k}(t) \approx u(x_j, y_k, t)$. We denote by $u_{j,k}^i$ the computed $u_{j,k}(t^i)$: $u_{j,k}^i \approx u_{j,k}(t^i)$ at the discrete time level $t^i := i\Delta t$, with time step Δt . Additionally, we denote by $\Delta_{j,k}$ the discrete Laplacian obtained using second order central difference formulas for the x and y variables, and by

$$L_{\Delta t, h}[u^{i+1}] := \Delta_{j,k} u^{i+1} - m u_{j,k}^{i+1}, \quad (x_j, y_k) \in \Omega^0. \quad (2.14)$$

Finally, we denote by g^u ,

$$g^u := g_{j,k}^u = -q_{j,k}^{i+1} - m u_{j,k}^i, \quad (2.15)$$

where, as before, $q_{j,k}^{i+1} \equiv q(x_j, y_k, t^{i+1})$ is the value of the source function $q(x_j, y_k, t)$ at t^{i+1} .

Thus, the fully discrete finite-difference based version of the parabolic equation (2.5) is:
Find some u^{i+1} such that it satisfies

$$L_{\Delta t, h}[u^{i+1}] := g^u, \quad (x_j, y_k) \in \Omega^0. \quad (2.16)$$

Again, as in the ‘‘Preliminaries’’ in Section 2, we will suppress for now the explicit dependence on the time level i for the clarity of the discussion. Based on a central finite difference approximation (2.16), we will now formulate the fully discrete analog of the auxiliary problem (AP), Def. 2.1 in Section 2 - the *Discrete Auxiliary Problem (DAP)*:

DEFINITION 2.3. *For the given grid function g , find the solution of the scheme f such that:*

$$L_{\Delta t, h}[f] = g, \quad (x_j, y_k) \in M_1, \quad (2.17)$$

$$f = 0, \quad (x_j, y_k) \in \gamma^1, \quad (2.18)$$

where, as before in (2.14), $L_{\Delta t, h}[f] \equiv \Delta_{j,k} f - m f_{j,k}$. We note that the ((DAP), Def. 2.3) is well defined for any right hand side g - it has a unique solution f defined on the set N^1 .

Also, it should be noted that the solution of ((DAP), Def. 2.3) can be efficiently computed using the Fast Fourier Transform (FFT) with the appropriate choice of the auxiliary set M^1 .

We now introduce a linear space V_γ of all the grid functions v_γ defined on γ , similar to [48, 10, 9, 49]. We will extend by zero the value of v_γ to other points of the grid N^1 . Let us now recall that,

DEFINITION 2.4. *A Difference Potential [48, 10, 9, 49] with the given density $v_\gamma \in V_\gamma$ is the grid function $u := \mathbf{P}_{N_\gamma} v_\gamma$ defined on the set N , which coincides (on the set N) with the solution of ((DAP), Def. 2.3) when the right hand-side is defined as follows:*

$$g := \begin{cases} 0, & (x_j, y_k) \in M, \\ L_{\Delta t, h}[v_\gamma], & (x_j, y_k) \in M^1 \setminus M. \end{cases} \quad (2.19)$$

The *Difference Potential* can be viewed as the discrete analog of the modified potential of Calderon’s type, or as the discrete analog of the space-continuous potential, Def. 2.2 in Section 2 (similar to Def. 2.4, the definition of *Difference Potential* can be extended by considering the set $M^1 \setminus M$ as the ‘‘interior’’ set, and set M as the ‘‘exterior’’ one, [49]). Here, \mathbf{P}_{N_γ} denotes the operator that constructs the difference potential $u = \mathbf{P}_{N_\gamma} v_\gamma$ from the given density $v_\gamma \in V_\gamma$. The operator \mathbf{P}_{N_γ} is the linear operator of density v_γ , and it can be easily constructed (see for example [10, 9, 49]). Again, for our problem, $L_{\Delta t, h}[v_\gamma] \equiv \Delta_{j,k} v_\gamma - m(v_\gamma)_{j,k}$.

As in the space-continuous case, (Def. 2.2, Section 2), the concept of the difference potential is well-defined at each time level due to the following statement:

THEOREM 2.5. *The difference potential $\mathbf{P}_{N_\gamma} v_\gamma$ depends only on $v_\gamma \in V_\gamma$, but is independent of the choice of the function v defined on N^1 (satisfying condition (2.18) on $\gamma^1 : v = 0, (x_j, y_k) \in \gamma^1$), and coinciding with v_γ on $\gamma : v|_\gamma \equiv Tr_\gamma v = v_\gamma$.*

Let us recall the proof for the reader’s convenience (for the general proof and discussion, see [49, 50, 48]).

Proof: Let us define the sets $N^+ := N$, $\Omega^- := \Omega^0 \setminus \Omega$, $M^- := (x_j, y_k) \in \Omega^-$, and $N^- := \{\bigcup_{j,k} N_{j,k} | (x_j, y_k) \in M^-\}$. Recall that, π_{N^+} denotes the restriction operator to the set $N := N^+$, $\Pi_{N^1 \setminus \gamma} v_\gamma$ is the arbitrary extension v_γ to N^1 , and χ_{M^-} is the characteristic function of set M^- .

Let us notice now that the difference potential, Def. 2.4, can be represented in the following operator form:

$$u = \mathbf{P}_{N^+ \setminus \gamma} v_\gamma = \pi_{N^+} L_{\Delta t, h}^{-1} [\chi_{M^-} L_{\Delta t, h}[v_\gamma]],$$

where v_γ is extended by zero outside of set γ to $N^1 \setminus \gamma$. Now, let us represent the arbitrary function z defined on N^1 , such that: $z = \Pi_{N^1 \setminus \gamma} v_\gamma$, as the sum of the three terms

$$z = \chi_{N^1 \setminus N^+} z + \chi_{N^1 \setminus N^-} z + v_\gamma.$$

Notice that, for the first term $\chi_M - L_{\Delta t, h}[\chi_{N^1 \setminus N} + z] = L_{\Delta t, h}[\chi_{N^1 \setminus N} + z]$, and for the second term $\chi_M - L_{\Delta t, h}[\chi_{N^1 \setminus N} - z] = 0$. From this we obtain that

$$\pi_{N+} L_{\Delta t, h}^{-1} [\chi_M - L_{\Delta t, h}[\chi_{N^1 \setminus N} + z]] = \pi_{N+} L_{\Delta t, h}^{-1} [L_{\Delta t, h}[\chi_{N^1 \setminus N} + z]] = 0.$$

Hence, the only remaining contribution to the potential u is the contribution from the third term :

$$u \equiv \mathbf{P}_{N+\gamma} v_\gamma = \pi_{N+} L_{\Delta t, h}^{-1} [\chi_M - L_{\Delta t, h}[v_\gamma]].$$

□

Next, let us recall [10, 9, 49] and define another operator $\mathbf{P}_\gamma : V_\gamma \rightarrow V_\gamma$ as the trace (or restriction) of the Difference Potentials $\mathbf{P}_{N\gamma} v_\gamma$ on the grid boundary γ : $\mathbf{P}_\gamma v_\gamma := \text{Tr}_\gamma \mathbf{P}_{N\gamma} v_\gamma = \mathbf{P}_{N\gamma} v_\gamma|_\gamma$. We now state an important theorem for the further development of our scheme (see for example [49, 48].) The details of the general proof can be found in [49]:

THEOREM 2.6. *At each time level t^{i+1} , density $v_\gamma \in V_\gamma$ is the trace of some solution $u := u_N \in N$: $v_\gamma = \text{Tr}_\gamma u_N \equiv u_\gamma$ to the homogeneous equation*

$$L_{\Delta t, h}[u] = 0, \quad (x_j, y_k) \in M \quad (2.20)$$

if and only if we have

$$Q_\gamma v_\gamma := v_\gamma - \mathbf{P}_\gamma v_\gamma = 0, \quad (x_j, y_k) \in \gamma. \quad (2.21)$$

Moreover, the solution u_N defined on the set N can be reconstructed from its boundary value v_γ using the formula $u := \mathbf{P}_{N\gamma} v_\gamma$.

Remark: It can be shown that \mathbf{P}_γ is the projector, hence $V_\gamma = \text{Im } \mathbf{P}_\gamma \oplus \text{Ker } \mathbf{P}_\gamma$.

Thus, Theorem 2.6 implies that the problem of finding a unique solution to (2.20) subject to the appropriate approximation of the boundary conditions on $\partial\Omega$ denoted by $l(u) = \psi$, in other words the problem:

$$L_{\Delta t, h}[u] = 0, \quad (2.22)$$

$$l(u) = \psi, \quad (2.23)$$

is equivalent to the problem of finding the unique density function $v_\gamma \equiv u_\gamma$ from the system of the *Boundary Equations*:

$$Q_\gamma v_\gamma = 0, \quad (2.24)$$

$$l(\mathbf{P}_{N\gamma} v_\gamma) = \psi. \quad (2.25)$$

After that, at each time level the solution u_N to (2.22)-(2.23) is reconstructed from $u_N = \mathbf{P}_{N\gamma} v_\gamma$.

Remark: Let us note that the equation (2.24) can be viewed as the generalized Poincaré-Steklov interface equation.

From the above discussion and, in particular, from Theorem 2.6 and from (2.24) - (2.25), the next result follows:

PROPOSITION 2.7. *At each time level t^{i+1} , the approximation u_N to the solution u of the (IVP) problem (2.5), (2.2) - (2.3) in domain $\bar{\Omega}$ can be obtained as*

$$u_N = \mathbf{P}_{N\gamma} v_\gamma + \bar{u}_N, \quad (2.26)$$

$$l(v + \bar{u}) = \psi, \quad (2.27)$$

where \bar{u}_N is the approximation of any particular solution to the inhomogeneous equation (2.5) in $\bar{\Omega}$, at the given time level t^{i+1} . $v_N := \mathbf{P}_{N\gamma} v_\gamma$ is the difference potential in $\bar{\Omega}$ at the same time level t^{i+1} , with a density v_γ that satisfies equation (2.24): $Q_\gamma v_\gamma = 0$. Equation (2.27) denotes the approximation of the boundary condition $l(u) = \psi$ on $\partial\Omega$, and the representation of the boundary conditions in the form (2.27) can be viewed as the consequence of the equality (2.26) and (2.24) (projected on the continuous boundary). Finally, define $\bar{u}_\gamma := \text{Tr}_\gamma \bar{u}_N$.

Remark: Let us mention that in (2.26), we can consider any particular solution: Let \bar{u}_N be some solution to $L_{\Delta t, h}[\bar{u}_N] = g^u, (x_j, y_k) \in M$. We extend the solution arbitrarily to N^1 , and we will apply operator $L_{\Delta t, h}$ to the extended solution \bar{u}_{N^1} . Hence, we will get the right-hand side of the equation $L_{\Delta t, h}[\bar{u}_{N^1}] = g$, which coincides on M with g^u . For example, we can construct the particular solution \bar{u}_N as follows:

DEFINITION 2.8. Define, \bar{u} to be the solution f of the auxiliary problem (DAP, Def. 2.3) with the right hand-side defined as

$$g := \begin{cases} g^u, & (x_j, y_k) \in M, \\ \tilde{g}^u, & (x_j, y_k) \in M^1 \setminus M \end{cases} \quad (2.28)$$

where g^u is given in (2.15) and \tilde{g}^u is some extension of g^u to $M^1 \setminus M$. Set $\bar{u}_N := \pi_N \bar{u}$. This point will be discussed in more details in Section 5.

Remark: Finally, let us comment briefly about the accuracy in space of the approximation $u_N = \mathbf{P}_{N_\gamma} v_\gamma + \bar{u}_N$ in (2.26)-(2.27), to the solution u of (2.5), (2.2) - (2.3). One would expect that the solution u_N will converge to the continuous solution u in the discrete Hölder norm of order $q + \epsilon$ (with the arbitrary $0 < \epsilon < 1$), with the rate $O(h^{p-\epsilon})$ as $h \rightarrow 0$. Here, p is the order of the accuracy of the approximation of the continuous differential operator $L_{\Delta t}$ by the discrete operator $L_{\Delta t, h}$ (we assume that the boundary condition (2.2) is approximated by (2.27) with the same or a higher order than p). Hence, for the central finite difference scheme that is discussed here, we will expect the $O(h^2)$ rate in the maximum norm in space. For a more detailed and general discussion, and the proof of the accuracy of the (DPM) method, the reader can consult [49, 34].

Remark: Let us emphasize that one could develop any other numerical approximation in space (such as high-order finite difference, finite-volume, finite element, or spectral methods) for (2.1) within the presented framework of the potentials and of the difference potentials. But for our goals in this paper, we will consider the second order finite-difference approximation for the model equation (2.5).

3. Algorithms Composition Approach based on the Difference Potentials Method

We will now develop, as well as numerically test the *Algorithms Composition Approach* for a parabolic model under consideration (2.5), (2.2) - (2.3).

We start the introduction of the scheme and the illustration of the ideas by considering the system (2.5), (2.2) - (2.3) (or formulation (2.9) in its time-discrete form) in some composite domain $\bar{\Omega}$. $\bar{\Omega}$ is an arbitrary bounded domain in \mathbb{R}^2 with the boundary $\partial\Omega$, and such that it consists of two disjoint subdomains Ω_1 and Ω_2 : $\bar{\Omega} = \bar{\Omega}_1 \cup \bar{\Omega}_2$, $\Gamma = \bar{\Omega}_1 \cap \bar{\Omega}_2$, with piecewise smooth boundaries $\Gamma_1 := \partial\Omega_1 \cup \Gamma$ and $\Gamma_2 := \partial\Omega_2 \cup \Gamma$ (see Figure 3.1). We would like to also emphasize that the two subdomains are considered here only for the

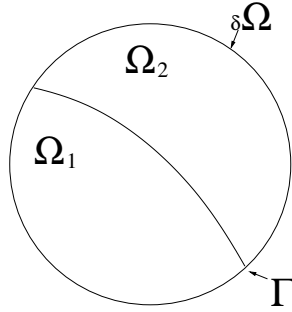


Fig. 3.1: Sketch of domain Ω with boundary $\partial\Omega$, two subdomains Ω_1, Ω_2 , and the interface Γ .

simplicity of the presentation. The discussed idea can be extended in a straightforward way to multiple subdomains or composite domains.

Our goal at each time level t^{i+1} is to find an approximations to u^{i+1} in domain Ω :

$$u_\Omega^{i+1} := \begin{cases} u_{\Omega_1}^{i+1}, & (x, y) \in \Omega_1, \\ u_{\Omega_2}^{i+1}, & (x, y) \in \Omega_2, \end{cases} \quad (3.1)$$

where $u_{\Omega_p}^{i+1}$ (here $p = 1, 2$) are the solutions to (2.9) in each subdomain Ω_p :

$$L_{\Delta t}[u_{\Omega_p}^{i+1}] = G_{\Delta t}^{u_{\Omega_p}}, \quad (x, y) \in \Omega_p, \text{ and } (p = 1, 2). \quad (3.2)$$

Solution u_{Ω}^{i+1} is subject to the appropriate boundary conditions on the boundary $\partial\Omega$ of the original domain

$$l(u_{\Omega}^{i+1}) = \psi^{i+1}, \quad (3.3)$$

and to the interface conditions on the interface boundary Γ , which we will select to be the following:

$$l_{\Gamma}(u_{\Omega_1}^{i+1}, u_{\Omega_2}^{i+1}) := \begin{cases} u_{\Omega_1}^{i+1} = u_{\Omega_2}^{i+1}, & (x, y) \in \Gamma, \\ \nabla u_{\Omega_1}^{i+1} \cdot n_{\Gamma} = \beta \nabla u_{\Omega_2}^{i+1} \cdot n_{\Gamma}, & (x, y) \in \Gamma. \end{cases} \quad (3.4)$$

Here, n_{Γ} is a unit outward normal vector to the interface boundary Γ (with respect to Ω_1). To develop our idea, we will consider the difference potentials scheme (2.26) - (2.27) from Section 2.1 (which uses Backward Euler in time and central finite difference space discretization) as the fully discrete approximation of (2.5), (2.2) - (2.3) in each subdomain Ω_p , and we will build an algorithm for the approximation of u_{Ω}^{i+1} in (3.1) based on the algorithms composition idea. As before, we will remove below the explicit dependence on time for the clarity of the presentation.

First, as we have done in (Section 2.1) for the single domain Ω , we will introduce auxiliary difference problems for each subdomain Ω_p , ($p = 1, 2$): we will place each of the original subdomains Ω_p in the auxiliary domains $\Omega_p^0 \subset \mathbb{R}^2$, ($p = 1, 2$). As before, the choice of each domain Ω_p^0 should be convenient for computations, and the choice of these auxiliary domains do not need to depend on each other. Again, for each subdomain, we will proceed in a similar way as we did in (Section 2.1). For each Ω_p^0 we will introduce, for example, a Cartesian mesh (again the choice of the meshes for the auxiliary problems in each subdomain can be totally independent. The choice for each subdomain is based on the considerations of the efficiency and simplicity of the resulting discrete problems.) After that, all the definitions, notations, and properties introduced in (Section 2.1) extends to each subdomain Ω_p in a direct and straightforward way: we will use index p , ($p = 1, 2$) to distinguish each subdomain.

The cornerstone of our approach is the following proposition, which is a consequence of Proposition 2.7 in Section 2.1 for a single domain.

PROPOSITION 3.1. *At each time level t^{i+1} , the fully discrete approximation u_{N_p} to the solution u_{Ω_p} , ($p = 1, 2$) in (3.2) - (3.4) is obtained as*

$$u_{N_p} = \mathbf{P}_{N_p \gamma_p} v_{\gamma_p} + \bar{u}_{N_p}, \quad (3.5)$$

$$l(\mathbf{v}_{\Gamma_1} + \bar{\mathbf{u}}_{\Gamma_1}, \mathbf{v}_{\Gamma_2} + \bar{\mathbf{u}}_{\Gamma_2}) = \psi, \quad (3.6)$$

$$l_{\Gamma}(\mathbf{v}_{\Gamma_1} + \bar{\mathbf{u}}_{\Gamma_1}, \mathbf{v}_{\Gamma_2} + \bar{\mathbf{u}}_{\Gamma_2}) = \phi, \quad (3.7)$$

where \bar{u}_{N_p} is the approximation of the particular solution to the inhomogeneous equation (2.5) in each subdomain $\bar{\Omega}_p$ at the given time level t^{i+1} . $\mathbf{P}_{N_p \gamma_p} v_{\gamma_p}$ is the difference potential with a density v_{γ_p} in each domain $\bar{\Omega}_p$ at the same time level t^{i+1} . Expressions in (3.6)-(3.7) denote the approximations of the boundary and interface conditions, respectively, on the continuous boundaries $\partial\Omega_1, \partial\Omega_2$, and interface boundary Γ (in other words on the boundaries Γ_1 and Γ_2). Denote $\bar{u}_{\gamma_p} = Tr_{\gamma_p} \bar{u}_N$. The construction of (3.6)-(3.7) will be discussed in more detail in Section 3.2. Density/trace v_{γ_p} in the equation (3.5) ranges over the solution of the *Boundary Equation* on each discrete grid boundary set γ_p :

$$Q_{\gamma_p} v_{\gamma_p} \equiv v_{\gamma_p} - \mathbf{P}_{\gamma_p} v_{\gamma_p} = 0, \quad (3.8)$$

see (2.24) in Section 2.1. However, there are multiple solutions $(v_{\gamma_1}, v_{\gamma_2})$ to (3.8), and as before, the unique pair of the densities/traces $v_{\gamma_1} \in V_{\gamma_1}$ and $v_{\gamma_2} \in V_{\gamma_2}$ will satisfy the above *Boundary Equation* (3.8), as well as the boundary and interface conditions (3.6) - (3.7).

3.1. System of Boundary Equations: Weak Formulation

At each time level, the unique pair of the densities/traces $(v_{\gamma_1}, v_{\gamma_2}) \equiv (Tr_{\gamma_1} v_{N_1}, Tr_{\gamma_2} v_{N_2}) \in V_{\gamma_1} \times V_{\gamma_2}$ of the approximation (3.5)-(3.7) is the unique solution to the *Boundary Equation* (3.8), subject to the boundary

and interface conditions (3.6) - (3.7). There are different ways to solve (3.8) subject to (3.6) - (3.7). One possibility is to directly consider the original formulation (3.8) and employ finite difference approximation of the boundary and interface conditions (3.6) - (3.7) (see for example [10, 49] for more details on this approach). However, to avoid the difficulties associated with the finite difference approximations of (3.6) - (3.7) in arbitrary domains, we will take advantage of the weak formulation of (3.8) and the spectral approximation of the Cauchy data.

To define a weak formulation, let us first introduce the discrete norm for the space of grid functions $v_{\gamma_p} \in V_{\gamma_p}$. Similar to [48], we will consider the following norm:

$$\|v_{\gamma_p}\|_{V_{\gamma_p}}^2 := h \left[\sum |v_\nu|^2 + \alpha \sum \left| \frac{v_{\nu_1+1, \nu_2} - v_\nu}{h} \right|^2 + \alpha \sum \left| \frac{v_{\nu_1, \nu_2+1} - v_\nu}{h} \right|^2 \right], (p = 1, 2). \quad (3.9)$$

Here, the sum is extended over all $\nu := (\nu_1, \nu_2) \in \gamma_p$ for the first term in (3.9), over all $\nu := (\nu_1, \nu_2)$ that, together with $(\nu_1 + 1, \nu_2)$ belong to γ_p in the second term of (3.9), and over all $\nu := (\nu_1, \nu_2)$ that, together with $(\nu_1, \nu_2 + 1)$ belong to γ_p in the third term of (3.9). α is a nonnegative coefficient which will be defined in the numerical experiments.

Remark: The norm (3.9) is the discrete analog of the continuous norm for the Cauchy data $\mathbf{v}_{\Gamma_p} \in V_{\Gamma_p}$:

$$\|\mathbf{v}_{\Gamma_p}\|_{V_{\Gamma_p}}^2 := \int_{\Gamma_p} \left[|v_p^0(s)|^2 + \alpha \left(\left| \frac{dv_p^0(s)}{ds} \right|^2 + |v_p^1(s)|^2 \right) \right] ds \quad (3.10)$$

Let us introduce $\tilde{\mathbf{v}}_{\Gamma_p} \in \tilde{V}_{\Gamma_p}$ as a finite dimensional approximation of the continuous Cauchy data $\mathbf{v}_{\Gamma_p} \in V_{\Gamma_p}$: $\tilde{\mathbf{v}}_{\Gamma_p} \approx \mathbf{v}_{\Gamma_p}$, and \tilde{V}_{Γ_p} as the finite dimensional subspace of V_{Γ_p} . The details on the construction of the approximation $\tilde{\mathbf{v}}_{\Gamma_p}$ will be presented in Section 3.2. Finally, let us also introduce

DEFINITION 3.2. *Let $\Pi_{\gamma_p \Gamma_p}$ be the operator that, to every $\mathbf{v}_{\Gamma_p} \equiv (v_p^0(s), v_p^1(s))$ from the space of continuous Cauchy data V_{Γ_p} , assigns v_{γ_p} from the space of the discrete densities V_{γ_p} , ($p = 1, 2$):*

$$v_{\gamma_p} = \Pi_{\gamma_p \Gamma_p} \mathbf{v}_{\Gamma_p}, \quad (3.11)$$

Moreover, we have from Def. 3.2 that for $\tilde{\mathbf{v}}_{\Gamma_p} \in \tilde{V}_{\Gamma_p}$, we can define $\tilde{v}_{\gamma_p} \in V_{\gamma_p}$ to be:

$$\tilde{v}_{\gamma_p} := \Pi_{\gamma_p \Gamma_p} \tilde{\mathbf{v}}_{\Gamma_p} \quad (3.12)$$

The exact form of the operator $\Pi_{\gamma_p \Gamma_p}$ will be given in (3.22) - (3.23) in Section 3.2 as well.

Therefore, the weak formulation of the *Boundary Equations* (3.8) subject to (3.6)-(3.7) is formulated as follows:

DEFINITION 3.3.

At each time level, find $(\tilde{\mathbf{v}}_{\Gamma_1}, \tilde{\mathbf{v}}_{\Gamma_2}) \equiv (\tilde{u}_{\Gamma_1}, \tilde{u}_{\Gamma_2}) \in \tilde{V}_{\Gamma_1} \times \tilde{V}_{\Gamma_2}$ that minimizes the functional

$$\|l(\tilde{\mathbf{v}}_{\Gamma_1} + \tilde{\mathbf{u}}_{\Gamma_1}, \tilde{\mathbf{v}}_{\Gamma_2} + \tilde{\mathbf{u}}_{\Gamma_2}) - \psi\|_{\Phi}^2 + \|l_{\Gamma}(\tilde{\mathbf{v}}_{\Gamma_1} + \tilde{\mathbf{u}}_{\Gamma_1}, \tilde{\mathbf{v}}_{\Gamma_2} + \tilde{\mathbf{u}}_{\Gamma_2}) - \phi\|_{\Phi}^2 + \sum_{p=1}^2 \|Q_{\gamma_p} \Pi_{\gamma_p \Gamma_p} \tilde{\mathbf{v}}_{\Gamma_p}\|_{V_{\gamma_p}}^2, \quad (3.13)$$

where $\|\cdot\|_{\Phi}$ is the Hilbert norm, \tilde{V}_{Γ_1} , and \tilde{V}_{Γ_2} are finite dimensional subspaces of V_{Γ_1} and V_{Γ_2} , respectively, and $(\tilde{\mathbf{u}}_{\Gamma_1}, \tilde{\mathbf{u}}_{\Gamma_2})$ are the finite dimensional approximations of $(\mathbf{u}_{\Gamma_1}, \mathbf{u}_{\Gamma_2})$ (it is a pair of the Cauchy Data of the particular solution $(\bar{u}_{N_1}, \bar{u}_{N_2})$). This finite dimensional approximation is discussed in Section 3.2.

Therefore, at each time level, the unique pair of the densities/traces $(v_{\gamma_1}, v_{\gamma_2}) \in V_{\gamma_1} \times V_{\gamma_2}$ of the approximation (3.5)-(3.7) to the solution (3.1) of (2.5), (2.2)-(2.3) in the composite domain Ω is obtained as the approximation:

$$(v_{\gamma_1}, v_{\gamma_2}) \approx (\tilde{v}_{\gamma_1}, \tilde{v}_{\gamma_2}) = (\Pi_{\gamma_1 \Gamma_1} \tilde{\mathbf{v}}_{\Gamma_1}, \Pi_{\gamma_2 \Gamma_2} \tilde{\mathbf{v}}_{\Gamma_2}). \quad (3.14)$$

3.2. System of Boundary Equations: Discretization of the Cauchy Data

As before, we will assume that the solution is given at a fixed time level and we will suppress the explicit dependence on time for the clarity of the presentation.

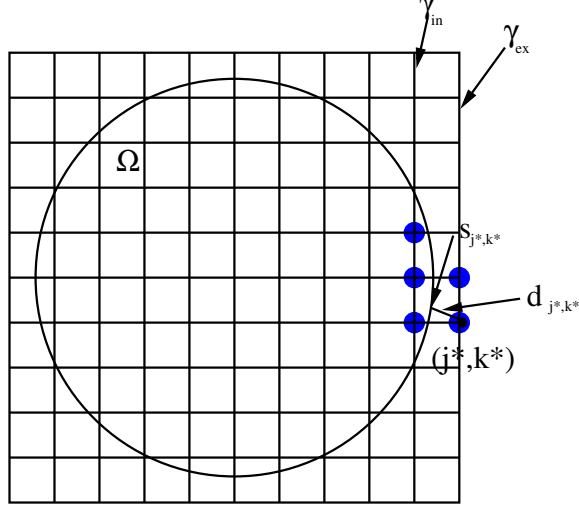


Fig. 3.2: Example (a sketch) of the geometry with the curve boundary Ω ; point (x_j^*, y_k^*) is in the set $\gamma_{ex} \in M$, d_{j^*, k^*} is the distance from this point to the boundary of the domain Ω , and s_{j^*, k^*} is the corresponding arc length.

1. As we already showed in Sections 3 - 3.1, at each time level, the problem of finding the approximation to the unique solution $u_\Omega := (u_{\Omega_1}, u_{\Omega_2})$ of the model (2.5), (2.2) - (2.3) in composite domain $\bar{\Omega} \equiv \bar{\Omega}_1 \cup \bar{\Omega}_2$ reduces to the problem of finding the unique pair of the densities/traces $(v_{\gamma_1}, v_{\gamma_2}) \equiv (u_{\gamma_1}, u_{\gamma_2}) \in V_{\gamma_1} \times V_{\gamma_2}$ (3.14). This pair of the densities/traces satisfies the generalized Poincaré-Steklov interface equations (3.8), as well as the boundary and interface conditions (3.6) - (3.7).
2. After that, the approximation of the solution $(u_{\Omega_1}, u_{\Omega_2})$ of (2.5), (2.2)-(2.3) in $\bar{\Omega}$ is reconstructed from the approximated densities $(\tilde{v}_{\gamma_1}, \tilde{v}_{\gamma_2}) \in V_{\gamma_1} \times V_{\gamma_2}$ using the pair of difference potentials $(\mathbf{P}_{N_1\gamma_1} \tilde{v}_{\gamma_1}, \mathbf{P}_{N_2\gamma_2} \tilde{v}_{\gamma_2})$, see equation (3.5).

In order to solve for the unknown densities/traces in Step 1 above, we will consider a weak formulation Def. 3.3, Section 3.1, and we will employ a spectral approach for the approximation (3.14) of $(v_{\gamma_1}, v_{\gamma_2})$ as elaborated below.

Consider a set of basis functions on the curve boundaries Γ_1 and Γ_2 :

$$\phi_1(s), \dots, \phi_{\mathcal{L}}(s), \text{ and } \phi_1^*(s), \dots, \phi_{\mathcal{L}}^*(s) \quad (3.15)$$

where s is the arc length and \mathcal{L} is the total number of the basis functions. We will consider here the same basis functions on all parts of the boundaries, although in general, the basis functions can be selected differently for the different parts of the boundaries. In general, this choice will depend on the smoothness of the boundary of the domains and on the smoothness of the solution.

Next, we will assume that for every sufficiently smooth single-valued periodic function $f(s)$ defined on the boundaries Γ_1 and Γ_2 , the sequence

$$\epsilon_{\mathcal{L}} := \min_{u_{p\ell}^0, u_{p\ell}^1} \int [|f(s) - \sum_{\ell=1}^{\mathcal{L}} u_{p\ell}^0 \phi_{\ell}(s)|^2 + |f'(s) - \sum_{\ell=1}^{\mathcal{L}} u_{p\ell}^1 \phi_{\ell}^*(s)|^2] ds \quad (3.16)$$

tends to zero with increasing \mathcal{L} : $\lim_{\mathcal{L}} \epsilon_{\mathcal{L}} = 0$ as $\mathcal{L} \rightarrow \infty$.

Therefore, we will employ the following approximation to construct finite-dimensional $\tilde{\mathbf{v}}_{\Gamma_p} \in \tilde{V}_{\Gamma_p}$:

$$\tilde{\mathbf{v}}_{\Gamma_p} := \left(\sum_{\ell=1}^{\mathcal{L}} u_{p\ell}^0 \phi_{\ell}, \sum_{\ell=1}^{\mathcal{L}} u_{p\ell}^1 \phi_{\ell}^* \right), \quad (3.17)$$

that discretize the elements $\mathbf{v}_{\Gamma_p} \in V_{\Gamma_p}$ from the space of continuous Cauchy Data on the boundary Γ_p ,

$p = 1, 2$:

$$\mathbf{v}_{\Gamma_p} := (v_p^0(s), v_p^1(s)), \quad (3.18)$$

and we have

$$\tilde{\mathbf{v}}_{\Gamma_p} \approx \mathbf{v}_{\Gamma_p}. \quad (3.19)$$

In (3.17), $\phi_\ell \equiv (\phi_\ell, 0)$ is the set of basis functions for the first component of the Cauchy data $v^0(s)$, and $\phi_\ell^* \equiv (0, \phi_\ell^*)$ is the set of basis functions for the second component of the Cauchy data $v^1(s)$, $\ell = 1, \dots, \mathcal{L}$. In our numerical experiments, we will consider the same set of basis functions for both components $\phi_\ell \equiv \phi_\ell^*$. The coefficients $(u_{p\ell}^0, u_{p\ell}^1)$ with $p = 1, 2$ and $\ell = 1, \dots, \mathcal{L}$, are the unknown expansion coefficients that need to be determined.

To obtain the approximation of the discrete densities/traces v_{γ_p} , ($p = 1, 2$) at the points $(x_j, y_k) \in \gamma_p$, we will consider the following Taylor expansion:

$$v_{\gamma_p} \equiv v_{N_p}(x_j, y_k)|_{\gamma_p} = v_p^0(s_{j,k}) + d_{j,k}v_p^1(s_{j,k}) + O(d_{j,k}^2). \quad (3.20)$$

Here, $s_{j,k}$ is the value of the arc length s at the point where the continuous boundary Γ_p intersects the normal constructed from the point $(x_j, y_k) \in \gamma_p$ to Γ_p . The parameter $d_{j,k}$ is the shortest distance from $(x_j, y_k) \in \gamma_p$ to the intersection point $s_{j,k}$ of the normal with Γ_p . $d_{j,k}$ is taken with the plus sign if $(x_j, y_k) \in \Omega_p$, and with a minus sign if $(x_j, y_k) \notin \Omega_p$, see Figure 3.2. Thus, we have

$$v_{\gamma_p} \approx v_p^0(s_{j,k}) + d_{j,k}v_p^1(s_{j,k}). \quad (3.21)$$

Let us recall the operator $\Pi_{\gamma_p\Gamma_p}$, Def. 3.2, Section 3.1 that assigns v_{γ_p} , ($p = 1, 2$) to every $\mathbf{v}_{\Gamma_p} \equiv (v_p^0(s), v_p^1(s))$ from the space of continuous Cauchy data. We will construct such an operator $\Pi_{\gamma_p\Gamma_p}$ according to the above Taylor formula (3.21), hence:

$$v_{\gamma_p} \approx \Pi_{\gamma_p\Gamma_p} \mathbf{v}_{\Gamma_p}, \quad (3.22)$$

$$\Pi_{\gamma_p\Gamma_p} \mathbf{v}_{\Gamma_p} := v_p^0(s_{j,k}) + d_{j,k}v_p^1(s_{j,k}). \quad (3.23)$$

For example, if $\mathbf{v}_{\Gamma_p} = (v_p^0(s), 0)$, then we have that $\Pi_{\gamma_p\Gamma_p} \mathbf{v}_{\Gamma_p} = v_p^0(s_{j,k})$. Similarly, if $\mathbf{v}_{\Gamma_p} = (0, v_p^1(s))$, then the action of the operator $\Pi_{\gamma_p\Gamma_p}$ is given as $\Pi_{\gamma_p\Gamma_p} \mathbf{v}_{\Gamma_p} = d_{j,k}v_p^1(s_{j,k})$. Thus, we have the following finite-dimensional approximation for the discrete density $v_{\gamma_p} \in V_{\gamma_p}$:

$$v_{\gamma_p} \approx \tilde{v}_{\gamma_p} = \Pi_{\gamma_p\Gamma_p} \tilde{\mathbf{v}}_{\Gamma_p} = \sum_{\ell=1}^{\mathcal{L}} (u_{p\ell}^0(\Pi_{\gamma_p\Gamma_p} \phi_\ell) + u_{p\ell}^1(\Pi_{\gamma_p\Gamma_p} \phi_\ell^*)), \quad (p = 1, 2) \quad (3.24)$$

where \tilde{v}_{γ_p} is as defined previously in (3.12), Section 3.1.

It follows that the approximation of the density/trace v_{γ_p} is the function of the unknown coefficients $u_{p\ell}^0$ and $u_{p\ell}^1$ which need to be determined. Once the expansion coefficients $u_{p\ell}^0$ and $u_{p\ell}^1$ are obtained, the density $\tilde{v}_{\gamma_p} \approx v_{\gamma_p} \equiv u_{\gamma_p}$ is reconstructed using formula (3.24).

3.3. System of Boundary Equations: Discrete Variational Formulation

As we showed in Section 3, densities/traces $(v_{\gamma_1}, v_{\gamma_2})$ range over the solutions of the *Boundary Equation* (3.8) on each discrete grid boundary set (γ_1, γ_2) . Therefore, using the above approximation (3.24) for v_{γ_p} in the system of *Boundary Equations* (3.8), we will obtain the system of linear equations with $|\gamma_p|$ equations for $2\mathcal{L}$ unknowns $\mathbf{u}_p^0 := (u_{p1}^0, \dots, u_{p\mathcal{L}}^0)$, $\mathbf{u}_p^1 := (u_{p1}^1, \dots, u_{p\mathcal{L}}^1)$:

$$B_p^0 \mathbf{u}_p^0 + B_p^1 \mathbf{u}_p^1 = 0, \quad (p = 1, 2). \quad (3.25)$$

Here, $|\gamma_p|$ is the total number of points in the set γ_p ($p = 1, 2$) and $(u_{1\ell}^0, u_{1\ell}^1)$, $(u_{2\ell}^0, u_{2\ell}^1)$ are the unknown expansion coefficients of $(\tilde{\mathbf{v}}_{\Gamma_1}, \tilde{\mathbf{v}}_{\Gamma_2})$ (3.19). Matrices B_p^0 and B_p^1 are defined as $B_p^0 := (\mathbf{b}_{p1}^0, \dots, \mathbf{b}_{p\mathcal{L}}^0)$ and $B_p^1 := (\mathbf{b}_{p1}^1, \dots, \mathbf{b}_{p\mathcal{L}}^1)$. Here, columns $\mathbf{b}_{p\ell}^0$ and $\mathbf{b}_{p\ell}^1$ are $|\gamma_p|$ dimensional vectors whose components are computed as the values at the points of the set $|\gamma_p|$ of the grid functions $Q_{\gamma_p}[\Pi_{\gamma_p\Gamma_p} \phi_\ell]$ and $Q_{\gamma_p}[\Pi_{\gamma_p\Gamma_p} \phi_\ell^*]$, respectively. For more a detailed discussion on the computation of these matrices please see Appendix Section 7.

Notice that the linear system (3.25) will be the overdetermined linear system since we have to have $|\gamma_p| > \mathcal{L}$ for the accurate resolution of the density v_{γ_p} . Finally, let us define

$$G_p(\tilde{\mathbf{v}}_{\Gamma_p}) := \|Q_{\gamma_p}[\Pi_{\gamma_p \Gamma_p} \tilde{\mathbf{v}}_{\Gamma_p}]\|_{V_{\gamma_p}}^2 = \|B_p^0 \mathbf{u}_p^0 + B_p^1 \mathbf{u}_p^1\|_{V_{\gamma_p}}^2, \quad (p = 1, 2) \quad (3.26)$$

Therefore, at each time level t^{i+1} we have the following discrete variational formulation of (3.13):

DEFINITION 3.4. *Find the unique weak solution pair $(\tilde{\mathbf{v}}_{\Gamma_1}, \tilde{\mathbf{v}}_{\Gamma_2}) \in \tilde{V}_{\Gamma_1} \times \tilde{V}_{\Gamma_2}$ for which the constants $(u_{1\ell}^0, u_{1\ell}^1)$ and $(u_{2\ell}^0, u_{2\ell}^1)$ with $\ell = 1, \dots, \mathcal{L}$ minimize the functional:*

$$\|l(\tilde{\mathbf{v}}_{\Gamma_1} + \tilde{\mathbf{u}}_{\Gamma_1}, \tilde{\mathbf{v}}_{\Gamma_2} + \tilde{\mathbf{u}}_{\Gamma_2}) - \psi\|_{\Phi}^2 + \|l_{\Gamma}(\tilde{\mathbf{v}}_{\Gamma_1} + \tilde{\mathbf{u}}_{\Gamma_1}, \tilde{\mathbf{v}}_{\Gamma_2} + \tilde{\mathbf{u}}_{\Gamma_2}) - \phi\|_{\Phi}^2 + \sum_{p=1}^2 G_p(\tilde{\mathbf{v}}_{\Gamma_p}). \quad (3.27)$$

Here, $\tilde{\mathbf{u}}_{\Gamma_p}$ is equal to $\tilde{\mathbf{u}}_{\Gamma_p} := \left(\sum_{\ell=1}^{\mathcal{L}} \bar{u}_{p\ell}^0 \phi_{\ell}, \sum_{\ell=1}^{\mathcal{L}} \bar{u}_{p\ell}^1 \phi_{\ell}^* \right)$, and the coefficients $(\bar{u}_{p\ell}^0, \bar{u}_{p\ell}^1)$ are obtained using the values of the particular solution \bar{u}_{N_p} at the points of the set γ_p , in other words using $\bar{u}_{\gamma_p} \approx \Pi_{\gamma_p \Gamma_p} \tilde{\mathbf{u}}_{\Gamma_p}$. The values \bar{u}_{γ_p} are known values since the particular solution \bar{u}_{N_p} is constructed as the solution of the simple auxiliary problem ((DAP, Def. 2.3), Section 2.1) (for more details see the Remark and Def. 2.8 after Proposition 2.7 in Section 2.1, as well as discussion in Section 5). Formulation (3.27) is the well-known least-square problem.

4. Algorithm

In this section, we will give brief summary of the important steps of the *Algorithms Composition Scheme* for the model problem (2.5), (2.2) - (2.3) in the composite domain $\bar{\Omega} = \bar{\Omega}_1 \cup \bar{\Omega}_2$. As it was derived in Sections 3.1 - 3.3, our algorithm will be based on the discrete variational formulation (3.27). Let us note a few important points of the method we developed. Firstly, the order of the operations of the proposed framework does not increase with the choice of the numerical discretization used with it. Moreover, the overall complexity of the method reduces to the several solutions of simple auxiliary problems on regular Cartesian meshes (no need for the generation and storage of the unstructured meshes; no need for the design of the schemes on the unstructured meshes). Finally, the selection of the auxiliary problems and meshes for each domain Ω_p is totally independent of each other, and is done based on the idea of the simplicity and efficiency of the resulting numerical scheme.

We have the following steps at each time level:

1. Construct a discrete functional $G_p(\tilde{\mathbf{v}}_{\Gamma_p}) \equiv \|B_p^0 \mathbf{u}_p^0 + B_p^1 \mathbf{u}_p^1\|_{V_{\gamma_p}}^2$ for each domain $\Omega_p, (p = 1, 2)$.

This functional is the weak formulation of the *Boundary Equation* (3.8) (note that the minimum of $G_p(\tilde{\mathbf{v}}_{\Gamma_p})$ gives an approximation to the Cauchy data of the general solution to the homogeneous equation in each subdomain).

To build $G_p(\tilde{\mathbf{v}}_{\Gamma_p})$:

- Select the auxiliary domain denoted here as $\Omega_{G_p}^0$, and place the original domain Ω_p into the auxiliary domain for the computation and construction of $G_p(\tilde{\mathbf{v}}_{\Gamma_p})$, $(p = 1, 2)$. The choice of the auxiliary domain $\Omega_{G_p}^0$ should be convenient for computation, so we will choose it to be a square. Select Cartesian mesh $2^{n_p} \times 2^{n_p}$ for each of the auxiliary domains $\Omega_{G_p}^0$, with n_p being a positive integer and $(p = 1, 2)$.
- Recall that $\tilde{\mathbf{v}}_{\Gamma_p} \in \tilde{V}_{\Gamma_p}$ is the finite dimensional approximation of the continuous Cauchy data $\mathbf{v}_{\Gamma_p} \in V_{\Gamma_p}$ (see formula (2.6) in Section 2, as well as (3.19) in Section 3.2), and it is constructed using the set of the basis functions $\phi_{\ell}(s)$ for the first component of the Cauchy data, as well as the set of the basis functions $\phi_{\ell}^*(s)$ for the second component of the Cauchy data (formula (3.17), Section 3.1) : $\tilde{\mathbf{v}}_{\Gamma_p} := \left(\sum_{\ell=1}^{\mathcal{L}} u_{p\ell}^0 \phi_{\ell}, \sum_{\ell=1}^{\mathcal{L}} u_{p\ell}^1 \phi_{\ell}^* \right)$.
- Construct matrices B_p^0 and B_p^1 for each domain. This construction reduces to the computation of the difference potentials (or to the solution of the simple auxiliary problems).

Remark: However if time step will not change during the simulations, and if the geometry, the grid, and the system of basis functions (3.15) will not depend on time, then it will suffice to precompute matrices B_p^0 and B_p^1 once and to store them.

See details in Section 3.3 and in Appendix Section 7.

2. Next, choose the new auxiliary domain denoted as $\Omega_{\tilde{u}_p}^0$ and place the original domain Ω_p into the auxiliary domain for the computation of the particular solution \bar{u}_{N_p} , ($p = 1, 2$). As before, the choice of the auxiliary domain $\Omega_{\tilde{u}_p}^0$ should be convenient for computation, so we will choose it to be a square. Select Cartesian mesh $2^{m_p} \times 2^{m_p}$ for the auxiliary domain $\Omega_{\tilde{u}_p}^0$, where m_p is a positive integer. Find any particular solution \bar{u}_{N_p} , ($p = 1, 2$) as the solution of the simple auxiliary problem (see *Remark* and see Def. 2.8 after Proposition 2.7 in Section 2.1, as well as the discussion at the beginning of Section 5). After that, calculate their Cauchy data on Γ_p , ($p = 1, 2$).
3. Solve the joint discrete variational problem: find Cauchy data $\tilde{\mathbf{v}}_{\Gamma_p} \in \tilde{V}_{\Gamma_p}$, ($p = 1, 2$) that minimizes the functional (3.27), Section 3.3:

$$\|l(\tilde{\mathbf{v}}_{\Gamma_1} + \tilde{\mathbf{u}}_{\Gamma_1}, \tilde{\mathbf{v}}_{\Gamma_2} + \tilde{\mathbf{u}}_{\Gamma_2}) - \psi\|_{\Phi}^2 + \|l_{\Gamma}(\tilde{\mathbf{v}}_{\Gamma_1} + \tilde{\mathbf{u}}_{\Gamma_1}, \tilde{\mathbf{v}}_{\Gamma_2} + \tilde{\mathbf{u}}_{\Gamma_2}) - \phi\|_{\Phi}^2 + \sum_{p=1}^2 G_p(\tilde{\mathbf{v}}_{\Gamma_p}).$$

Remark: Let us note that if the Dirichlet to Neumann map is well defined for the problem, then one can solve the overdetermined linear system (3.25) for the unknowns $\mathbf{u}_p^1 := (u_{p1}^1, \dots, u_{p\mathcal{L}}^1)$ in terms of $\mathbf{u}_p^0 := (u_{p1}^0, \dots, u_{p\mathcal{L}}^0)$ using the least squares method:

$$B_p^0 \mathbf{u}_p^0 + B_p^1 \mathbf{u}_p^1 = 0, \quad (p = 1, 2),$$

This step can be viewed as the construction of the Dirichlet to Neumann map: to each set \mathbf{u}_p^0 assign a set \mathbf{u}_p^1 that minimizes $G_p(\tilde{\mathbf{v}}_{\Gamma_p})$. In this case, the general solution in each Ω_p will be approximated by a difference potential whose density will depend only on \mathbf{u}_p^0 .

4. Using the obtained Cauchy data $\tilde{\mathbf{v}}_{\Gamma_p}$, construct $\tilde{v}_{\gamma_p} \approx \Pi_{\gamma_p \Gamma_p} \tilde{\mathbf{v}}_{\Gamma_p}$, Section 3.2. After that, extend the computed densities \tilde{v}_{γ_p} from the grid boundaries γ_p to the interior of each domain Ω_p by computing difference potentials $\mathbf{P}_{N_p} \tilde{v}_{\gamma_p}$ in each domain Ω_p .

At each time level reconstruct the desired approximation to the model (2.5), (2.2) - (2.3) in the composite domain $\bar{\Omega}$ using the discrete generalized Green's formula:

$$u_{N_p} = \mathbf{P}_{N_p \gamma_p} \tilde{v}_{\gamma_p} + \bar{u}_{N_p}, \quad (p = 1, 2)$$

(see Proposition 3.1 and equation (3.5) in Section 3.)

Remark: Let us emphasize the important flexibility of the algorithm to choose different auxiliary domains $\Omega_{G_p}^0$ and $\Omega_{\tilde{u}_p}^0$ for the computations of the projection and of the particular solution. This provides an opportunity to consider different grids and combine different discretizations if necessary.

5. Numerical Examples

In this section, we demonstrate the performance of the proposed *Algorithms Composition Approach* on several test problems. In the numerical experiments below, we consider the set of the basis functions on Γ_p , ($p = 1, 2$) that is defined as:

$$\begin{aligned} \phi_{p,1}(s) &= 1, \quad \phi_{p,2}(s) = \cos\left(\frac{2\pi}{|\Gamma|}s\right), \quad \phi_{p,3} = \sin\left(\frac{2\pi}{|\Gamma|}s\right), \dots \\ \phi_{p,2\mathcal{N}}(s) &= \cos\left(\frac{2\pi}{|\Gamma|}\mathcal{N}s\right), \quad \phi_{p,2\mathcal{N}+1} = \sin\left(\frac{2\pi}{|\Gamma|}\mathcal{N}s\right) \end{aligned} \quad (5.1)$$

We assume that the sets $\phi \equiv \phi^*$ in the numerical examples below. In all of the tests, we will consider the heat equation (2.5) in the composite domain $\bar{\Omega} \equiv \bar{\Omega}_1 \cup \bar{\Omega}_2$ as our model problem. Here, the heat equation serves as a simplified model for more realistic systems of the materials, fluids, or chemicals with different properties in different parts of the domains (for example, the ocean-atmosphere models, chemotaxis models, or blood flow models [2, 36, 43, 42]). We consider equation (2.5) with the known analytical expressions for the exact solutions. This allows us to study the errors in the approximate solutions that will depend on the size of the auxiliary domains, mesh sizes, total number of the basis functions on Γ_p , etc.

Similar to [48], we will first define three functions

$$u^{(1)}(x, y) := \cos(x) \cos(y), \quad (5.2)$$

$$u^{(2)}(x, y) := \sin(c_1 x) \sin(c_2 y) P_5(\rho/0.9), \quad (5.3)$$

and

$$u^{(3)}(x, y) := \max\left(\frac{1 - \rho^2}{1 + 4\rho^2}, 0\right). \quad (5.4)$$

Here, the Cartesian coordinates (x, y) of points are related to their polar coordinates (r, θ) as $(x, y) = (r \cos(\theta), r \sin(\theta))$. The interface boundary $\Gamma \equiv \Gamma_2$ of the domain Ω_2 is parametrically defined in polar coordinates (r, θ) by the relation $r(\theta) \equiv r_\Gamma(\theta) = 1 + 0.22 \sin(k_\theta \theta)$, where k_θ is the parameter. $\rho = r/r_\Gamma(\theta)$ in (5.3) - (5.4). See Figures 5.1 - 5.4 for the examples of the domains that are used in the numerical tests below.

The function $P_5(x)$ is continuous: identically equal to 1 for $x \leq 0$; vanishes for $x \geq 1$ and, on the interval $0 \leq x \leq 1$, it is a unique ninth-degree polynomial whose derivatives up to the fourth order vanish at the endpoints of $0 \leq x \leq 1$:

$$P_5(x) := \begin{cases} 1, & x \leq 0, \\ 1 - 126x^5 + 420x^6 - 540x^7 + 315x^8 - 70x^9, & 0 \leq x \leq 1, \\ 0, & x \geq 1. \end{cases} \quad (5.5)$$

Let us note that due to the multiplier $P_5(\rho/0.9)$, the function $u^{(2)}$ (5.3) vanishes outside of Ω_2 and in a neighborhood of $\Gamma \equiv \Gamma_2$. At the same time, $u^{(2)}$ exhibits strong oscillations deep inside Ω_2 .

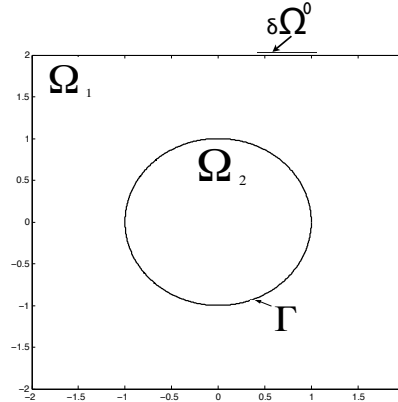


Fig. 5.1: Test Problem 1: Example of the auxiliary domain Ω^0 , domains Ω_1 and Ω_2 and the boundary Γ of the domain Ω_2 ; parameter $k_\theta = 0$

5.1. First Example

In the first example, we will construct our test problem with the analytical solution that is given as,

$$u(x, y, t) = \begin{cases} e^{-t} u_{\Omega_1}(x, y) = e^{-t} u^{(1)}, & (x, y) \in \Omega_1, \\ e^{-t} u_{\Omega_2}(x, y) = e^{-t} (u^{(1)} + u^{(2)}), & (x, y) \in \Omega_2, \end{cases} \quad (5.6)$$

with $c_1 = c_2 = 16$ in the formula (5.3). Thus, the solution $u(x, y, t)$ will exhibit strong oscillations inside the domain Ω_2 , but none in the domain Ω_1 . We consider four test problems, Figures 5.1 - 5.4 with $k_\theta = 0$ in the Test Problem 1, Figure 5.1; $k_\theta = 2$ in the Test Problem 2, Figure 5.2; with $k_\theta = 3$ in the Test Problem 3, Figure 5.3; and with $k_\theta = 5$ in the Test Problem 4, Figure 5.4.

We select $u(x, y, t)$ in (5.6) to be the exact solution for the heat equation (2.5), we set time step $dt = 1.e-6$ for all tests in Section 5.1, we use discrete norm (3.9) with $\alpha = 0.5$, and we consider time interval $[0, 0.01]$. In all experiments in this Section, we will choose the auxiliary domains

$$\Omega_{G_1}^0 \equiv \Omega_{u_{N_1}}^0 := [-2, 2] \times [-2, 2],$$

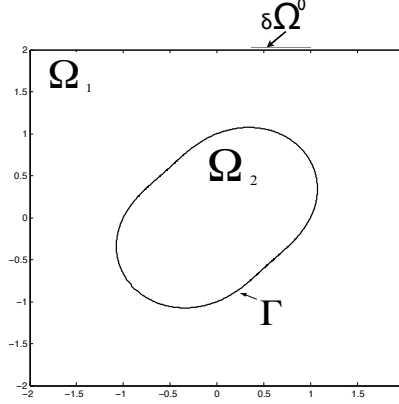


Fig. 5.2: Test Problem 2: Example of the auxiliary domain Ω^0 , domains Ω_1 and Ω_2 and the boundary Γ of the domain Ω_2 ; $k_\theta = 2$

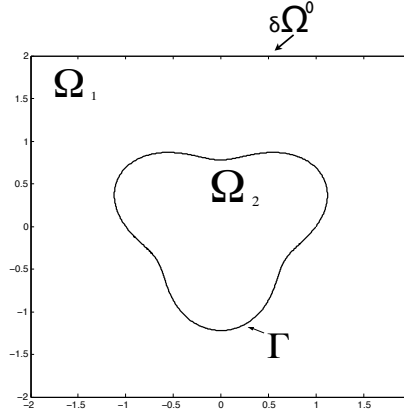


Fig. 5.3: Test Problem 3: Example of the auxiliary domain Ω^0 , domains Ω_1 and Ω_2 and the boundary Γ of the domain Ω_2 ; $k_\theta = 3$

for the construction of $G_1(\tilde{\mathbf{v}}_{\Gamma_1})$, as well as for the computation of the particular solution \bar{u}_{N_1} in Ω_1 (note that these auxiliary domains $\Omega_{G_1}^0$ and $\Omega_{\bar{u}_{N_1}}^0$ will coincide with the boundaries of the domain Ω_1 , see Figures 5.1 - 5.4). At the same time, we will select the auxiliary domain

$$\Omega_{G_2}^0 = [-1.6, 1.6] \times [-1.6, 1.6]$$

for the construction of the $G_2(\tilde{\mathbf{v}}_{\Gamma_2})$, and we will choose the auxiliary domain

$$\Omega_{\bar{u}_{N_2}}^0 = [-1.7, 1.7] \times [-1.7, 1.7]$$

for the approximation of the particular solution \bar{u}_{N_2} in Ω_2 . We will use the same Cartesian meshes

$$2^{n_1} \times 2^{n_1} \equiv 2^{n_2} \times 2^{n_2} \text{ with } n_1 \equiv n_2 = 9,$$

for the auxiliary domains $\Omega_{G_1}^0$ and $\Omega_{G_2}^0$ in the Tables 5.2 - 5.9. We will select Cartesian mesh

$$2^{m_1} \times 2^{m_1}$$

(hence, $h_1 := 4/2^{m_1}$) for the approximation of the particular solution \bar{u}_{N_1} in $\Omega_{\bar{u}_{N_1}}^0$, as well as the Cartesian mesh

$$2^{m_2} \times 2^{m_2}$$

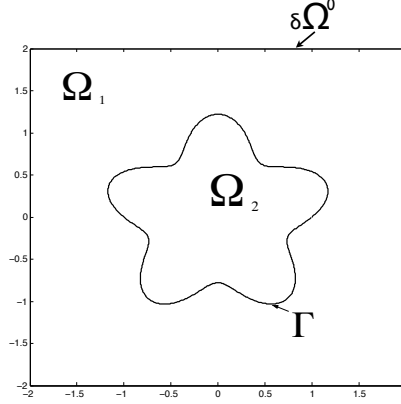


Fig. 5.4: Test Problem 4: Example of the auxiliary domain Ω^0 , domains Ω_1 and Ω_2 and the boundary Γ of the domain Ω_2 ; $k_\theta = 5$

(hence, $h_2 := 3.4/2^{m_2}$) for the computation of the particular solution \bar{u}_{N_2} in $\Omega_{\bar{u}_{N_2}}^0$, see Tables 5.1 - 5.9.

To compute the particular solution \bar{u}_{N_1} in $\Omega_{\bar{u}_{N_1}}^0$, we make the smooth extension $\tilde{g}^u \in \Omega_{\bar{u}_{N_1}}^0 \setminus \Omega_1$ of the right-hand side $g^u = -q^{i+1} - u_{\Omega_1}^i$ (see Def. 2.8) outside of the domain Ω_1 (in other words we make a smooth extension to the entire auxiliary domain/square Ω^0 , see Figures 5.1 - 5.4). Similarly, to compute the particular solution \bar{u}_{N_2} in $\Omega_{\bar{u}_{N_2}}^0$, we make smooth extension of the right-hand side $g^u = -q^{i+1} - u_{\Omega_2}^i$ outside of the domain Ω_2 to the auxiliary domain $\Omega_{\bar{u}_{N_2}}^0$. We compute the $L_2(L_\infty)$ (Tables 5.2 - 5.10) and $L_\infty(L_\infty)$ (Tables 5.1 - 5.10) errors respectively:

$$L_2(L_\infty) := \left(\sum_{i=0}^{N_t} \Delta t (\max_{(x,y)} |u_{calc}^i - u_{exact}^i|)^2 \right)^{1/2},$$

and

$$L_\infty(L_\infty) := \max_{(i=0, \dots, N_t)} \left(\max_{(x,y)} |u_{calc}^i - u_{exact}^i| \right).$$

Let us first make a few important remarks about the accuracy of the *Algorithms Composition Framework* and validate the developed approach for problems with complex interfaces. *We will select a suitable problem and test the accuracy of the developed approach against the accuracy of the numerical method on the single and geometrically simple domain.* In the current paper we combined the proposed *Algorithms Composition Framework* with second-order central finite difference scheme in space and first order Backward Euler scheme in time: the backward-centered scheme. (However, the developed algorithms composition approach is general. Future research will include the development of the higher-order space and time discretization within the proposed *Algorithms Composition Framework*, see Conclusion Section 6 for a more detailed discussion). Therefore, the obtained space accuracy is limited by the accuracy of the second order finite difference scheme. Let us illustrate this point, first using theoretical estimates and then numerically. The complex interface test problem (5.6) which we consider in this Section can be viewed as the classical solution (without complex interface) to the heat equation (2.5) in the square domain $\Omega \equiv [-2, 2] \times [-2, 2]$. Hence, the solution (5.6) to (2.5) can be computed approximately using standard backward-centered finite difference scheme for the heat equation on the single and simple square domain $\Omega \equiv [-2, 2] \times [-2, 2]$. Thus we can test the accuracy of our *Algorithms Composition Framework* against the accuracy of the “single domain computations”. Moreover, one would expect that the accuracy of the problem (5.6) with the complex interface will be limited by the accuracy of “single domain computations”. Let us recall as well, that the theoretical error $L_\infty(L_\infty)$ of the backward-centered finite difference scheme in the square domain with Dirichlet boundary conditions is given as

$$\max_{0 \leq i \leq N_t} \max_{0 \leq j, k \leq N_s} |u((jh, kh), i\Delta t) - U_{jk}^i| \leq cT(\Delta t + h^2), \quad (5.7)$$

where $c = \max\{(\|\frac{\partial^4 u}{\partial x^4}\|_{L^\infty(\bar{\Omega} \times [0, T])} + \|\frac{\partial^4 u}{\partial y^4}\|_{L^\infty(\bar{\Omega} \times [0, T])})/12, \|\frac{\partial^2 u}{\partial t^2}\|_{L^\infty(\bar{\Omega} \times [0, T])}/2\}$, where $u((jh, kh), i\Delta t)$ is the exact solution at the discrete points of the space and time mesh, and U_{jk}^i is the approximate solution obtained by the standard backward-centered scheme. If we now take the smallest mesh size $h \approx 0.0033$ (which was used as the smallest mesh size in the problems with complex interfaces below), then by performing rough estimates using formula (5.7) and expression for the analytical solution (5.6), we can show that the best accuracy that can be achieved by the backward-centered finite difference scheme on the single and geometrically simple domain $\Omega \equiv [-2, 2] \times [-2, 2]$ is about $O(10^{-4}) - O(10^{-5})$ (note that the second order continuous piecewise-linear finite element method will result in an error with a similar range). Therefore, our results in Tables 5.1 - 5.10 (as well as the results in Tables 5.11 - 5.12, and Table 5.13) below will be limited by the second order space accuracy. We illustrate this point numerically below as well, see Table 5.1.

For the experiment in Table 5.1 we used the settings as described above except in the last two rows of the table: $2^{n_1} \times 2^{n_1} \equiv 2^{n_2} \times 2^{n_2}$ with $n_1 \equiv n_2 = 10$ for the auxiliary domains $\Omega_{G_1}^0$ and $\Omega_{G_2}^0$, and the time step was set to $dt = 8.e - 7$ (instead of $1.e - 6$) to avoid the influence of the time discretization error. Notations in Table 5.1: “Finite Difference” stands for the “single domain computations” approach when solution to the problem (5.6) viewed as the classical solution (without complex interface) to the heat equation (2.5) in the square domain $\Omega \equiv [-2, 2] \times [-2, 2]$, and backward-centered finite difference scheme on the single and geometrically simple domain $\Omega \equiv [-2, 2] \times [-2, 2]$ is used to approximate the solution numerically. Mesh h_0 is used everywhere in the domain $\Omega \equiv [-2, 2] \times [-2, 2]$ for the computation in this “single domain computations”. “Algorithms Composition Framework” in Table 5.1 is used when we consider (5.6) as the complex interface problem. Meshes with h_1 and h_2 are used for the computations of the particular solutions $\bar{u}_{N_1} \in \Omega_1$ and $\bar{u}_{N_2} \in \Omega_2$, respectively, in the Algorithms Composition Framework. We consider test Problem 3 with $2\mathcal{L} = 42$ in this experiment with complex interface problem. Notation “—//—” in Table 5.1 is used to denote the same value as in the above row.

We see that the accuracy of our Algorithms Composition Framework for complex interface problem (5.6) does not differ from the accuracy of the “single domain computations”. Moreover, Table 5.1 illustrates an important flexibility of Algorithms Composition Framework - different meshes can be used in different subdomains, and the same accuracy is achieved using a much coarser mesh (with bigger size of h) in the regions/subdomains of the problems where solutions exhibit smaller gradients. This feature of our method is very important for the future development of the adaptive schemes/simulations for heterogenous problems, as well as for the parallel computations.

Next, let us continue with more accuracy tests of Algorithms Composition Framework alone using different curves as the interfaces below (see Tables 5.2 - 5.9). We again observe a second order convergence for the space discretization for all four test problems from Tables 5.2 - 5.9. Also, there is not much difference in the results when we change the total number $2\mathcal{L}$ of the basis functions for the approximation of the Cauchy data from $2\mathcal{L} = 42$, Table 5.1 to $2\mathcal{L} = 22$ and $2\mathcal{L} = 30$, Tables 5.2 - 5.9. This is explained by the low dimension of the spaces of the Cauchy data \mathbf{u}_Γ (2.6) of the exact solution u .

Finally, in the last Table 5.10 in this Section, we again consider the test Problem 3 with $2\mathcal{L} = 22$, but we fix the mesh size $m_1 = 7$ for the construction of the particular solution \bar{u}_{N_1} in the domain $\Omega_{\bar{u}_{N_1}}^0$, and we only vary the mesh size m_2 . The accuracy of the results is not affected in comparison to the results presented in Tables 5.2-5.9 with $m_1 = m_2$ (this is similar to the performance of Algorithms Composition Framework that is illustrated in Table 5.1). Again, this is expected due to the highly oscillatory behavior of the solution (5.6) in the subdomain Ω_2 , and hence computations of the solution in the subdomain Ω_2 requires finer/smaller mesh sizes. This example again illustrates the advantage of our algorithms composition approach since we have flexibility to use different meshes in the different parts of the domain, and to solve problems independently in each domain. This makes the numerical scheme much more computationally efficient and very suitable for adaptive and parallel computations.

5.2. Second Example

As the second test problem we will again consider the heat equation (2.5), but with the exact solution given below:

$$u(x, y, t) = \begin{cases} 100tu_{\Omega_1}(x, y) = 100tu^{(1)}, & (x, y) \in \Omega_1, \\ 100tu_{\Omega_2}(x, y) = 100t(u^{(1)} + u^{(2)}), & (x, y) \in \Omega_2. \end{cases} \quad (5.8)$$

We consider here $c_1 = c_2 = 4$ in (5.3). The function $u^{(2)}$ is much less oscillatory now than the one in the

“Algorithms Composition Framework”				“Finite Difference”		
m_1	m_2	h_1	h_2	$L_\infty(L_\infty)$ error of $u(x, y)$	h_0	$L_\infty(L_\infty)$ error of $u(x, y)$
6	8	0.0625	0.0133	0.00387	0.0133	0.00388
8	8	0.0156	0.0133	0.00387	0.0133	—//—
7	9	0.0312	0.0066	0.00097	0.0066	0.00097
9	9	0.0078	0.0066	0.00097	0.0066	—//—
8	10	0.0156	0.0033	0.00024	0.0033	0.00024
10	10	0.0039	0.0033	0.00024	0.0033	—//—

TABLE 5.1. Comparison of the $L_\infty(L_\infty)$ Errors for the “Algorithms Composition Framework” (with complex interface: test problem 3, $k_\theta = 3$, number of the basis functions $2\mathcal{L} = 42$;) and “Classical Approach” (without complex interface) as functions of the mesh size;

	$L_2(L_\infty)$ error of $u(x, y)$	Ratio	$L_\infty(L_\infty)$ error of $u(x, y)$	Ratio
7	0.00130		0.01523	
8	0.00032	4.06	0.00382	3.99
9	0.00008	4.00	0.00096	3.98

TABLE 5.2. Errors as functions of the mesh size $m_1 = m_2$; number of the basis functions $2\mathcal{L} = 22$; $k_\theta = 0$, Test problem 1

previous Section 5.1.

As before, we set time step $dt = 1.e - 6$ for all tests in Section 5.2, and we consider the same time interval $[0, 0.01]$. Here, for the construction of the $G_1(\tilde{\mathbf{v}}_{\Gamma_1})$, as well as for the computation of the particular solution \bar{u}_{N_1} , we will choose the same auxiliary domains (as in the Section 5.1)

$$\Omega_{G_1}^0 \equiv \Omega_{\bar{u}_{N_1}}^0 := [-2, 2] \times [-2, 2].$$

However, we will select the auxiliary domain

$$\Omega_{G_2}^0 = [-1.5, 1.5] \times [-1.5, 1.5],$$

for the construction of the $G_2(\tilde{\mathbf{v}}_{\Gamma_2})$, and we will consider the auxiliary domain

$$\Omega_{\bar{u}_{N_2}}^0 = [-1.6, 1.6] \times [-1.6, 1.6],$$

for the computation of the particular solution \bar{u}_{N_2} . We will use the Cartesian meshes

$$2^{n_1} \times 2^{n_1} \equiv 2^{n_2} \times 2^{n_2} \text{ with } n_1 \equiv n_2 = 10$$

for the domains $\Omega_{G_1}^0$ and $\Omega_{G_2}^0$, and we will select two different discrete norms (3.9) with $\alpha = 0.5$ and $\alpha = 1.0$ for the construction of the $G_1(\tilde{\mathbf{v}}_{\Gamma_1})$ and $G_2(\tilde{\mathbf{v}}_{\Gamma_2})$. The results are reported in Table 5.11 ($\alpha = 0.5$) and in Table 5.12 ($\alpha = 1.0$). As demonstrated, the results are not affected by the choice of α . The errors that are reported in Tables 5.11 - 5.12 are smaller than the ones in Tables 5.2 - 5.9. This is again expected due to the less oscillatory behavior of the function $u^{(2)}$ in (5.8), and due to the choice of the auxiliary problems.

Remark: In the numerical experiments presented in Sections 5.1 and 5.2 we observed the overall second order convergence of the scheme in space. Let us note that according to the general theoretical results for a second order differential operator approximated by a discrete one with a second order accuracy [44] (see also [49]), one would need to consider Taylor expansion in (3.20) with the derivative of order $2 + 2 = 4$ to maintain a second order accuracy of the approximation of the continuous potential by a difference potential. However, it is also established that in reality this condition can be relaxed (see for example [34] and formula (3.20) in this paper). We believe that the choice of the norm in the variational formulation (3.13) plays an important role for the obtained convergence of our method in the considered numerical tests.

	$L_2(L_\infty)$ error of $u(x, y)$	Ratio	$L_\infty(L_\infty)$ error of $u(x, y)$	Ratio
7	0.00130		0.01523	
8	0.00032	4.06	0.00382	3.99
9	0.00008	4.00	0.00096	3.98

TABLE 5.3. Errors as functions of the mesh size $m_1 = m_2$; number of the basis functions $2\mathcal{L} = 30$; $k_\theta = 0$, Test problem 1

	$L_2(L_\infty)$ error of $u(x, y)$	Ratio	$L_\infty(L_\infty)$ error of $u(x, y)$	Ratio
7	0.00131		0.01534	
8	0.00033	3.97	0.00389	3.94
9	0.00008	4.13	0.00097	4.01

TABLE 5.4. Errors as functions of the mesh size $m_1 = m_2$; number of the basis functions $2\mathcal{L} = 22$; $k_\theta = 2$, Test problem 2

	$L_2(L_\infty)$ error of $u(x, y)$	Ratio	$L_\infty(L_\infty)$ error of $u(x, y)$	Ratio
7	0.00131		0.01534	
8	0.00033	3.97	0.00389	3.94
9	0.00008	4.13	0.00097	4.01

TABLE 5.5. Errors as functions of the mesh size $m_1 = m_2$; number of the basis functions $2\mathcal{L} = 30$; $k_\theta = 2$, Test problem 2

	$L_2(L_\infty)$ error of $u(x, y)$	Ratio	$L_\infty(L_\infty)$ error of $u(x, y)$	Ratio
7	0.00131		0.01540	
8	0.00033	3.97	0.00387	3.98
9	0.00008	4.13	0.00097	3.99

TABLE 5.6. Errors as functions of the mesh size $m_1 = m_2$; number of the basis functions $2\mathcal{L} = 22$; $k_\theta = 3$, Test problem 3

	$L_2(L_\infty)$ error of $u(x, y)$	Ratio	$L_\infty(L_\infty)$ error of $u(x, y)$	Ratio
7	0.00131		0.01540	
8	0.00033	3.97	0.00387	3.98
9	0.00008	4.13	0.00097	3.99

TABLE 5.7. Errors as functions of the mesh size $m_1 = m_2$; number of the basis functions $2\mathcal{L} = 30$; $k_\theta = 3$, Test problem 3

	$L_2(L_\infty)$ error of $u(x, y)$	Ratio	$L_\infty(L_\infty)$ error of $u(x, y)$	Ratio
7	0.00138		0.01588	
8	0.00035	3.94	0.00398	3.99
9	0.00009	3.89	0.0010	3.98

TABLE 5.8. Errors as functions of the mesh size $m_1 = m_2$; number of the basis functions $2\mathcal{L} = 22$; $k_\theta = 5$, Test problem 4

	$L_2(L_\infty)$ error of $u(x, y)$	Ratio	$L_\infty(L_\infty)$ error of $u(x, y)$	Ratio
7	0.00138		0.01588	
8	0.00035	3.94	0.00398	3.99
9	0.00009	3.89	0.0010	3.98

TABLE 5.9. Errors as functions of the mesh size $m_1 = m_2$; number of the basis functions $2\mathcal{L} = 30$; $k_\theta = 5$, Test problem 4

	$L_2(L_\infty)$ error of $u(x, y)$	Ratio	$L_\infty(L_\infty)$ error of $u(x, y)$	Ratio
7	0.00131		0.01540	
8	0.00033	3.97	0.00387	3.98
9	0.00008	4.13	0.00097	3.99

TABLE 5.10. Errors as functions of the mesh size m_2 ; $m_1 = 7$ is fixed. number of the basis functions $2\mathcal{L} = 22$; $k_\theta = 3$, Test problem 3

	$L_2(L_\infty)$ error of $u(x, y)$	Ratio	$L_\infty(L_\infty)$ error of $u(x, y)$	Ratio
8	0.00001		0.00020	
9	$2.5e - 6$	4.0	0.00005	4.0

TABLE 5.11. Errors as functions of the mesh size $m_1 = m_2$; number of the basis functions $2\mathcal{L} = 22$; $k_\theta = 3$, $\alpha = 0.5$, Test problem 3

5.3. Third Example

As the third test problem we will again consider the heat equation (2.5), but with the exact solution given below:

$$u(x, y, t) = \begin{cases} e^{-t}u_{\Omega_1}(x, y) = e^{-t}u^{(1)}, & (x, y) \in \Omega_1, \\ e^{-t}u_{\Omega_2}(x, y) = e^{-t}(u^{(1)} + u^{(2)} + 1.5u^{(3)}), & (x, y) \in \Omega_2. \end{cases} \quad (5.9)$$

We consider here $c_1 = c_2 = 1$ in (5.3). This test problem (5.9) is more challenging since $\beta(s) \neq 1$ in the interface conditions (3.4) and it will be calculated as

$$\beta(s) = \frac{\partial u_{\Omega_1}}{\partial n} \left(\frac{\partial u_{\Omega_2}}{\partial n} \right)^{-1} \quad (5.10)$$

As before, we set time step $dt = 1.e - 6$ for the test in Section 5.3, and we consider the same time interval $[0, 0.01]$. Here, for the construction of the $G_1(\tilde{\mathbf{v}}_{\Gamma_1})$, as well as for the computation of the particular solution \bar{u}_{N_1} we will choose the same auxiliary domains (as in the Sections 5.1 - 5.2)

$$\Omega_{G_1}^0 \equiv \Omega_{\bar{u}_{N_1}}^0 := [-2, 2] \times [-2, 2].$$

Also, for the construction of the $G_2(\tilde{\mathbf{v}}_{\Gamma_2})$ we will select the auxiliary domain

$$\Omega_{G_2}^0 = [-1.6, 1.6] \times [-1.6, 1.6],$$

and for the computation of the particular solution \bar{u}_{N_2} we will consider the auxiliary domain

$$\Omega_{\bar{u}_{N_2}}^0 = [-1.7, 1.7] \times [-1.7, 1.7].$$

For the domains $\Omega_{G_1}^0$ and $\Omega_{G_2}^0$, we will use the Cartesian meshes

$$2^{n_1} \times 2^{n_1} \equiv 2^{n_2} \times 2^{n_2} \text{ with } n_1 \equiv n_2 = 10,$$

and we will select the discrete norm (3.9) with $\alpha = 0.5$ for the construction of the $G_1(\tilde{\mathbf{v}}_{\Gamma_1})$ and $G_2(\tilde{\mathbf{v}}_{\Gamma_2})$. The results are reported in Table 5.13. For this test problem (5.9) we observe that the convergence of our scheme drops. This could be explained by the jump in the normal derivative of the solution (5.9) at the interface, by the accuracy limitations of the considered second order finite difference space discretization (used away from the interface), by the choice of the extension operator (formula (3.23)) , as well as by the first order time discretization scheme used in this paper within the algorithms composition framework. As the part of the future research, this result will be improved by considering different time and space discretization within the developed approach, as well as different choices of the extension operators.

	$L_2(L_\infty)$ error of $u(x, y)$	Ratio	$L_\infty(L_\infty)$ error of $u(x, y)$	Ratio
8	0.00001		0.00020	
9	$2.5e - 6$	4.0	0.00005	4.0

TABLE 5.12. Errors as functions of the mesh size $m_1 = m_2$; number of the basis functions $2\mathcal{L} = 22$; $k_\theta = 3$, $\alpha = 1.0$, Test problem 3

	$L_2(L_\infty)$ error of $u(x, y)$	Ratio	$L_\infty(L_\infty)$ error of $u(x, y)$	Ratio
7	$3.3e - 5$		0.00051	
8	$8.3e - 6$	3.97	0.00013	3.92
9	$3.1e - 6$	2.68	$4.2e - 5$	3.10

TABLE 5.13. Errors as functions of the mesh size $m_1 = m_2$; number of the basis functions $2\mathcal{L} = 22$; $k_\theta = 0$, $\alpha = 0.5$, Test problem 1

6. Concluding Remarks

In this work, we developed an efficient and flexible *Algorithms Composition Framework* based on the idea of the difference potentials method (DPM) for parabolic problems in composite domains. We illustrated the accuracy, efficiency, and flexibility of our method with several numerical examples. Here, the parabolic equation served as the simplified model, and the first step towards future development of the proposed scheme for more realistic models of materials, fluids, or chemicals with different properties in different domains.

In this work we considered and tested our approach only on the geometries with smooth curvilinear boundaries. In the future, we plan to extend the proposed method to problems in domains with arbitrary smooth boundaries and with boundaries with corners. In this respect, the choice of the different basis functions (3.15) for different parts of the boundaries, as well as the question of the efficient linear solvers for the scheme (3.13) will be investigated with the goal of designing even more efficient and accurate methods.

Other future investigations will include the extension and further development of the proposed scheme to problems in physics and biology (see for example [10, 9, 53, 43, 42], [2, 5, 6, 21]), as well as the development of the space discretization based on the high-order finite difference (see for example [34]), finite-volume (see for example [22], [7, 10, 9]), finite element methods (see for example [11]), and spectral methods [16] within the proposed algorithms composition framework. Higher-order time discretization schemes will be studied as well.

Acknowledgment: The author would like to express her gratitude to Professors Viktor Ryaben’kii and Viktor Turchaninov for introducing and teaching her the idea of the Difference Potentials Method many years ago, and for the warm and encouraging atmosphere of their collaboration. The author also would like to thank very much Professor Viktor Ryaben’kii for the inspiring discussions about future research.

The author would like to thank Professor Fengyan Li for the helpful discussions. Finally, the author is grateful to the anonymous referees for the careful reading of the paper and for their most valuable comments about the method, suggestions that helped to improve the manuscript, as well as questions inspiring future directions. The research is supported in part by the National Science Foundation Grant # DMS-1112984.

REFERENCES

- [1] C. A. Brebbia. *The boundary element method for engineers*. Halsted Press [John Wiley & Sons], New York, 1978.
- [2] Didier Bresch and Jonas Koko. Operator-splitting and Lagrange multiplier domain decomposition methods for numerical simulation of two coupled Navier-Stokes fluids. *Int. J. Appl. Math. Comput. Sci.*, 16(4):419–429, 2006.
- [3] S. Britt, S. Tsynkov, and E. Turkel. A high order numerical method for the helmholtz equation with non-standard boundary conditions. *SISC*, 2013. submitted for publication.
- [4] A.-P. Calderón. Boundary value problems for elliptic equations. In *Outlines Joint Sympos. Partial Differential Equations (Novosibirsk, 1963)*, pages 303–304. Acad. Sci. USSR Siberian Branch, Moscow, 1963.
- [5] Claire Chainais-Hillairet and Francis Filbet. Asymptotic behaviour of a finite-volume scheme for the transient drift-diffusion model. *IMA J. Numer. Anal.*, 27(4):689–716, 2007.
- [6] Claire Chainais-Hillairet, Jian-Guo Liu, and Yue-Jun Peng. Finite volume scheme for multi-dimensional drift-diffusion equations and convergence analysis. *M2AN Math. Model. Numer. Anal.*, 37(2):319–338, 2003.

- [7] A. Chertock, Y. Epshteyn, and A. Kurganov. High-order finite-difference and finite-volume methods for chemotaxis models. 2010. in preparation.
- [8] Robert Dautray and Jacques-Louis Lions. *Mathematical analysis and numerical methods for science and technology. Vol. 4*. Springer-Verlag, Berlin, 1990. Integral equations and numerical methods, With the collaboration of Michel Artola, Philippe Bénilan, Michel Bernadou, Michel Cessenat, Jean-Claude Nédélec, Jacques Planchard and Bruno Scheurer, Translated from the French by John C. Amson.
- [9] Y. Epshteyn. Upwind-difference potentials method for chemotaxis models. 2012. to appear in the proceedings of "European Congress on Computational Methods in Applied Sciences and Engineering (ECCOMAS 2012)".
- [10] Y. Epshteyn. Upwind-difference potentials method for Patlak-Keller-Segel chemotaxis model. *Journal of Scientific Computing*, 53(3):689 – 713, 2012. DOI: 10.1007/s10915-012-9599-2.
- [11] Yekaterina Epshteyn and Alexander Kurganov. New interior penalty discontinuous Galerkin methods for the Keller-Segel chemotaxis model. *SIAM J. Numer. Anal.*, 47(1):386–408, 2008/09.
- [12] Lisa J. Fauci and Charles S. Peskin. A computational model of aquatic animal locomotion. *J. Comput. Phys.*, 77(1):85–108, 1988.
- [13] Ronald P. Fedkiw, Tariq Aslam, Barry Merriman, and Stanley Osher. A non-oscillatory Eulerian approach to interfaces in multimaterial flows (the ghost fluid method). *J. Comput. Phys.*, 152(2):457–492, 1999.
- [14] Frédéric Gibou and Ronald Fedkiw. A fourth order accurate discretization for the Laplace and heat equations on arbitrary domains, with applications to the Stefan problem. *J. Comput. Phys.*, 202(2):577–601, 2005.
- [15] Frederic Gibou, Ronald P. Fedkiw, Li-Tien Cheng, and Myungjoo Kang. A second-order-accurate symmetric discretization of the Poisson equation on irregular domains. *J. Comput. Phys.*, 176(1):205–227, 2002.
- [16] David Gottlieb and Steven A. Orszag. *Numerical analysis of spectral methods: theory and applications*. Society for Industrial and Applied Mathematics, Philadelphia, Pa., 1977. CBMS-NSF Regional Conference Series in Applied Mathematics, No. 26.
- [17] A. Greenbaum and A. Mayo. Rapid parallel evaluation of integrals in potential theory on general three-dimensional regions. *J. Comput. Phys.*, 145(2):731–742, 1998.
- [18] L. Greengard and V. Rokhlin. A fast algorithm for particle simulations. *J. Comput. Phys.*, 73(2):325–348, 1987.
- [19] Boyce E. Griffith and Charles S. Peskin. On the order of accuracy of the immersed boundary method: higher order convergence rates for sufficiently smooth problems. *J. Comput. Phys.*, 208(1):75–105, 2005.
- [20] Myungjoo Kang, Ronald P. Fedkiw, and Xu-Dong Liu. A boundary condition capturing method for multiphase incompressible flow. *J. Sci. Comput.*, 15(3):323–360, 2000.
- [21] Heinz-Otto Kreiss and N. Anders Petersson. An embedded boundary method for the wave equation with discontinuous coefficients. *SIAM J. Sci. Comput.*, 28(6):2054–2074 (electronic), 2006.
- [22] Alexander Kurganov and Eitan Tadmor. New high-resolution central schemes for nonlinear conservation laws and convection-diffusion equations. *J. Comput. Phys.*, 160(1):241–282, 2000.
- [23] Randall J. LeVeque and Zhi Lin Li. The immersed interface method for elliptic equations with discontinuous coefficients and singular sources. *SIAM J. Numer. Anal.*, 31(4):1019–1044, 1994.
- [24] Randall J. LeVeque and Zhilin Li. Immersed interface methods for Stokes flow with elastic boundaries or surface tension. *SIAM J. Sci. Comput.*, 18(3):709–735, 1997.
- [25] Zhilin Li and Kazufumi Ito. *The immersed interface method*, volume 33 of *Frontiers in Applied Mathematics*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2006. Numerical solutions of PDEs involving interfaces and irregular domains.
- [26] Xu-Dong Liu, Ronald P. Fedkiw, and Myungjoo Kang. A boundary condition capturing method for Poisson’s equation on irregular domains. *J. Comput. Phys.*, 160(1):151–178, 2000.
- [27] Xu-Dong Liu and Thomas C. Sideris. Convergence of the ghost fluid method for elliptic equations with interfaces. *Math. Comp.*, 72(244):1731–1746 (electronic), 2003.
- [28] J. Loncaric, V. S. Ryaben’kii, and S. V. Tsynkov. Active shielding and control of noise. *SIAM J. Appl. Math.*, 62(2):563–596 (electronic), 2001.
- [29] A. Mayo and A. Greenbaum. Fast parallel iterative solution of Poisson’s and the biharmonic equations on irregular regions. *SIAM J. Sci. Statist. Comput.*, 13(1):101–118, 1992.
- [30] Anita Mayo. The fast solution of Poisson’s and the biharmonic equations on irregular regions. *SIAM J. Numer. Anal.*, 21(2):285–299, 1984.
- [31] Anita Mayo. The rapid evaluation of volume integrals of potential theory on general regions. *J. Comput. Phys.*, 100(2):236–245, 1992.
- [32] A. McKenney, L. Greengard, and A. Mayo. A fast Poisson solver for complex geometries. *J. Comput. Phys.*, 118(2):348–355, 1995.
- [33] David McQueen and Charles Peskin. Shared-memory parallel vector implementation of the immersed boundary method for the computation of blood flow in the beating mammalian heart. *The Journal of Supercomputing*, 11(3):213–236, 1997.
- [34] M. Medvinsky, S. Tsynkov, and E. Turkel. The method of difference potentials for the helmholtz equation using compact high order schemes. *Journal of Scientific Computing*, pages 1–44, May 2012.
- [35] Duc Q. Nguyen, Ronald P. Fedkiw, and Myungjoo Kang. A boundary condition capturing method for incompressible flame discontinuities. *J. Comput. Phys.*, 172(1):71–98, 2001.
- [36] Benoît Perthame. *Transport equations in biology*. Frontiers in Mathematics. Birkhäuser Verlag, Basel, 2007.
- [37] Charles S. Peskin. *Mathematical aspects of heart physiology*. Courant Institute of Mathematical Sciences New York University, New York, 1975. Notes based on a course given at New York University during the year 1973/74.
- [38] Charles S. Peskin. Numerical analysis of blood flow in the heart. *J. Computational Phys.*, 25(3):220–252, 1977.
- [39] Charles S. Peskin. The immersed boundary method. *Acta Numer.*, 11:479–517, 2002.
- [40] Charles S. Peskin and David M. McQueen. A three-dimensional computational method for blood flow in the heart. I.

- Immersed elastic fibers in a viscous incompressible fluid. *J. Comput. Phys.*, 81(2):372–405, 1989.
- [41] Alfio Quarteroni and Alberto Valli. *Domain decomposition methods for partial differential equations*. Numerical Mathematics and Scientific Computation. The Clarendon Press Oxford University Press, New York, 1999. Oxford Science Publications.
- [42] Alfio Quarteroni, Alessandro Veneziani, and Paolo Zunino. Mathematical and numerical modeling of solute dynamics in blood flow and arterial walls. *SIAM J. Numer. Anal.*, 39(5):1488–1511 (electronic), 2001/02.
- [43] Alfio Quarteroni, Alessandro Veneziani, and Paolo Zunino. A domain decomposition method for advection-diffusion processes with application to blood solutes. *SIAM J. Sci. Comput.*, 23(6):1959–1980 (electronic), 2002.
- [44] A. A. Reznik. Approximation of surface potentials of elliptic operators by difference potentials. *Dokl. Akad. Nauk SSSR*, 263(6):1318–1321, 1982.
- [45] V. S. Ryaben'kiĭ. Boundary equations with projectors. *Uspekhi Mat. Nauk*, 40(2(242)):121–149, 238, 1985.
- [46] V. S. Ryaben'kiĭ. Difference potentials analogous to Cauchy integrals. *Uspekhi Mat. Nauk*, 67(3(405)):147–172, 2012.
- [47] V. S. Ryaben'kii and S. V. Tsynkov. Artificial boundary conditions for the numerical solution of external viscous flow problems. *SIAM J. Numer. Anal.*, 32(5):1355–1389, 1995.
- [48] V. S. Ryaben'kiĭ, V. I. Turchaninov, and E. Yu. Èpshteĭn. An algorithm composition scheme for problems in composite domains based on the method of difference potentials. *Zh. Vychisl. Mat. Mat. Fiz.*, 46(10):1853–1870, 2006.
- [49] V.S. Ryaben'kii. Method of difference potentials and its applications. *Springer-Verlag*, 2001.
- [50] V.S. Ryaben'kii, V. I. Turchaninov, and Ye. Yu. Epshteyn. The numerical example of algorithms composition for solution of the boundary-value problems on compound domain based on difference potential method. *Moscow, Keldysh Institute for Applied Mathematics, Russia Academy of Sciences*, (3), 2003.
- [51] Andrea Toselli and Olof Widlund. *Domain decomposition methods—algorithms and theory*, volume 34 of *Springer Series in Computational Mathematics*. Springer-Verlag, Berlin, 2005.
- [52] S. Tsynkov. Numerical solution of problems on unbounded domains. A review. *Appl. Numer. Math.*, 27(4):465–532, 1998. Absorbing boundary conditions.
- [53] Idan Tuval, Luis Cisneros, Christopher Dombrowski, Charles W. Wolgemuth, John O. Kessler, and Raymond E. Goldstein. Bacterial swimming and oxygen transport near contact lines. *PNAS*, 102 (7):2277–2282, 2005.
- [54] Luoding Zhu and Charles S. Peskin. Simulation of a flapping flexible filament in a flowing soap film by the immersed boundary method. *J. Comput. Phys.*, 179(2):452–468, 2002.

7. Appendix: Matrix Computation for System of Boundary Equations As we discussed in Section 3.3 using the approximation (3.24) for v_{γ_p} in the system of *Boundary Equations* (3.8), we obtain the system of linear equations (3.25) with $|\gamma_p|$ equations for $2\mathcal{L}$ unknowns $\mathbf{u}_p^0 := (u_{p1}^0, \dots, u_{p\mathcal{L}}^0)$, $\mathbf{u}_p^1 := (u_{p1}^1, \dots, u_{p\mathcal{L}}^1)$:

$$B_p^0 \mathbf{u}_p^0 + B_p^1 \mathbf{u}_p^1 = 0, \quad (p = 1, 2).$$

Once again, here, $|\gamma_p|$ is the total number of points in the set γ_p ($p = 1, 2$) and $(u_{1\ell}^0, u_{1\ell}^1)$, $(u_{2\ell}^0, u_{2\ell}^1)$ are the unknown expansion coefficients of $(\tilde{\mathbf{v}}_{\Gamma_1}, \tilde{\mathbf{v}}_{\Gamma_2})$ (3.19). Matrices B_p^0 and B_p^1 are defined as $B_p^0 := (\mathbf{b}_{p1}^0, \dots, \mathbf{b}_{p\mathcal{L}}^0)$ and $B_p^1 := (\mathbf{b}_{p1}^1, \dots, \mathbf{b}_{p\mathcal{L}}^1)$. Columns $\mathbf{b}_{p\ell}^0$ and $\mathbf{b}_{p\ell}^1$ of these matrices are $|\gamma_p|$ dimensional vectors. The components of the vectors are computed as the values at the points of the set $|\gamma_p|$ of the grid functions $Q_{\gamma_p}[\Pi_{\gamma_p \Gamma_p} \phi_\ell]$ and $Q_{\gamma_p}[\Pi_{\gamma_p \Gamma_p} \phi_\ell^*]$, respectively. These values of the grid functions

$$Q_{\gamma_p}[\Pi_{\gamma_p \Gamma_p} \phi_\ell] \equiv \Pi_{\gamma_p \Gamma_p} \phi_\ell - \mathbf{P}_{N_{\gamma_p}}[\Pi_{\gamma_p \Gamma_p} \phi_\ell]$$

and

$$Q_{\gamma_p}[\Pi_{\gamma_p \Gamma_p} \phi_\ell^*] \equiv \Pi_{\gamma_p \Gamma_p} \phi_\ell^* - \mathbf{P}_{N_{\gamma_p}}[\Pi_{\gamma_p \Gamma_p} \phi_\ell^*]$$

are known and obtained by constructing $2\mathcal{L}$ difference potentials: let us recall that the difference potential $u_{N_p} = \mathbf{P}_{N_{\gamma_p}} v_{\gamma_p}$ (Def. 2.4, Section 2.1) can be easily constructed in general. The operator $\mathbf{P}_{N_{\gamma_p}}$ is the linear operator of the density v_{γ_p} . Here, the difference potential

$$\mathbf{P}_{N_{\gamma_p}}[\Pi_{\gamma_p \Gamma_p} \phi_\ell], \text{ or } \mathbf{P}_{N_{\gamma_p}}[\Pi_{\gamma_p \Gamma_p} \phi_\ell^*]$$

is constructed by solving the simple auxiliary problem ((DAP), Def. 2.3), with the right-hand side given in (2.19), where density v_{γ_p} is set to:

$$v_{\gamma_p} := \Pi_{\gamma_p \Gamma_p} \phi_{\ell^*}, \text{ for } (x_j, y_k) \in \gamma_p, \quad (7.1)$$

or to

$$v_{\gamma_p} := \Pi_{\gamma_p \Gamma_p} \phi_{\ell^*}^*, \text{ for } (x_j, y_k) \in \gamma_p, \quad (7.2)$$

with fixed ℓ^* that takes the values from 1 to \mathcal{L} . Let us note that for the detailed discussion on the general construction of the difference potential, one may refer to [49, 10].