

DEEP ADAPTIVE BASIS GALERKIN METHOD FOR HIGH-DIMENSIONAL EVOLUTION EQUATIONS WITH OSCILLATORY SOLUTIONS*

YIQI GU[†] AND MICHAEL K. NG[†]

Abstract. In this paper, we study deep neural networks (DNNs) for solving high-dimensional evolution equations with oscillatory solutions. Different from deep least-squares methods that deal with time and space variables simultaneously, we propose a deep adaptive basis Galerkin (DABG) method which employs the spectral-Galerkin method for time variable by a tensor-product basis for oscillatory solutions and the deep neural network method for high-dimensional space variables. The proposed method can lead to a linear system of differential equations having unknown DNNs that can be trained via the loss function. We establish a posteriori estimates of the solution error which is bounded by the minimal loss function and the term $O(N^{-m})$, where N is the number of basis functions and m characterizes the regularity of the equation, and show that if the true solution is a Barron-type function, the error bound converges to zero as $M = O(N^p)$ approaches to infinity where M is the width of the used networks and p is a positive constant. Numerical examples, including high-dimensional linear parabolic and hyperbolic equations, and a nonlinear Allen–Cahn equation are presented to demonstrate that the performance of the proposed DABG method is better than that of existing DNNs.

Key words. deep learning, Galerkin method, parabolic equation, hyperbolic equation, Legendre polynomials

MSC codes. 65M60, 65M15, 68T07, 41A25

DOI. 10.1137/21M1468383

1. Introduction. The partial differential equations (PDEs) that involve time widely appear in areas such as fluid dynamics, biological population, and chemical diffusion. Numerical methods for solving evolution equations have been developed extensively in the past several decades. In most cases, the equations are discretized in time by numerical schemes (e.g., linear multistep methods and Runge–Kutta methods), and the derived semidiscrete equations in space are thereafter tackled by traditional PDE solvers such as finite difference methods and finite element methods. Despite high accuracy, these traditional solvers suffer from the curse of dimensionality in storage and hence are generally incapable of solving problems with very high dimensions.

In recent years, a series of methods based on deep neural networks (DNNs) have been proposed to solve high-dimensional PDEs (e.g., [6, 9, 15, 17, 27, 4, 29, 19, 3, 28, 20]). Basically, the unknown solution is approximated by a DNN, which is trained through the minimization of specific loss functions. Thanks to the special nonlinear structure, the number of free parameters in a DNN increases mildly with the dimension, making high-dimensional approximations feasible with limited storage. On the other hand, the approximation errors of neural networks have dimension-independent rates for special functions (e.g., continuous functions [22], Hölder functions [24, 23, 16], and Barron functions [1, 2, 18, 8, 7, 25, 26, 5]), hence overcoming or lessening the

*Submitted to the journal's Methods and Algorithms for Scientific Computing section December 29, 2021; accepted for publication (in revised form) June 24, 2022; published electronically September 26, 2022.

<https://doi.org/10.1137/21M1468383>

Funding: This work is supported by HKRGC GRF 12300519, 17201020 and 17300021, C1013-21GF, C7004-21GF and joint NSFC-RGC N-HKU76921.

[†]Department of Mathematics, The University of Hong Kong, Pokfulam, Hong Kong (yiqigu@hku.hk, mng@maths.hku.hk).

curse of dimensionality.

1.1. Problem description. We focus on the development of DNN-based methods for two typical categories of evolution equations, the parabolic and hyperbolic equations. As an example, we first consider the linear parabolic equation. Given a bounded domain $\Omega \subset \mathbb{R}^d$ with $d \in \mathbb{N}^+$ and $T > 0$, we aim to solve

$$(1.1) \quad \begin{cases} \frac{\partial u(x,t)}{\partial t} + \mathcal{L}_x u(x,t) = f(x,t) & \text{in } \Omega \times (0, T], \\ u(x,0) = g(x) & \text{in } \Omega, \quad u(x,t) = h(x,t) & \text{on } \partial\Omega \times [0, T], \end{cases}$$

where \mathcal{L}_x is a uniformly elliptic operator only involving x ; f , g , and h are given functions. Several methods such as physics-informed neural networks [20] and the deep Galerkin method [27] are developed for similar problems. Overall, these methods can be generalized as the DNN-based least-squares method. More precisely, let \mathcal{N} be a generic class of DNNs; it searches for the solution of (1.1) in \mathcal{N} by the following minimization problem,

$$(1.2) \quad \min_{\hat{u} \in \mathcal{N}} J[\hat{u}] := \left\| \frac{\partial \hat{u}}{\partial t} + \mathcal{L}_x \hat{u} - f \right\|_{L^2(\Omega \times (0, T])}^2 + \lambda_1 \|\hat{u}(\cdot, 0) - g\|_{L^2(\Omega)}^2 + \lambda_2 \|\hat{u} - h\|_{L^2(\partial\Omega \times [0, T])}^2$$

with $\lambda_1, \lambda_2 > 0$, where the loss function is defined as the square sum of equation error and condition errors, and the minimizer \hat{u} is taken as an approximate solution. Note that if (1.1) is a well-posed problem, then $\|u\|$ is bounded by $O(\|f\| + \|g\| + \|h\|)$. By the linearity, the error $\|\hat{u} - u\|$ is bounded by the loss function $J[\hat{u}]$. It implies that if a network \hat{u} has a small value of $J[\hat{u}]$, it is close to the true solution u .

In practice, the L^2 integrals in J are usually discretized and computed by the Monte Carlo method (especially when d is large), and the minimization can be solved under systematic deep learning frameworks. The deep least-squares (DLS) method enjoys the advantages that it is friendly in high dimensions and flexible for various kinds of equations and domains. However, the main difficulty lies in solving the optimization (1.2) numerically. Due to the high nonconvexity of the loss function, usual optimizers (e.g., stochastic gradient descent) might converge to bad local minimizers if the initialization or hyper-parameters are not configured well. For this issue, one approach is to simplify the loss function. In [14], Gu, Wang, and Yang constructed special DNN structures according to the initial/boundary conditions. More precisely, for the problem (1.1), one can design a DNN-based class \mathcal{N}' such that $\hat{u}(x, 0) = g(x)$ in Ω and $\hat{u}(x, t) = h(x, t)$ on $\partial\Omega \times [0, T]$ for all $\hat{u} \in \mathcal{N}'$. With hypothesis space \mathcal{N}' , the condition errors in J automatically vanish. Then the optimization (1.2) can be simplified as follows:

$$(1.3) \quad \min_{\hat{u} \in \mathcal{N}'} J'[\hat{u}] := \left\| \frac{\partial \hat{u}}{\partial t} + \mathcal{L}_x \hat{u} - f \right\|_{L^2(\Omega \times (0, T])}^2.$$

This design leads to an easier implementation and facilitates the convergence of optimizers to some extent. Nevertheless, a number of experiments show that if the true solution is fairly oscillatory, then optimizers are still likely to find bad local minimizers. For example, the least-squares method fails in the two-dimensional (2-D) heat equation with a solution having eight trigonometric waves in time (see section 5.2.1). The drawback of the least-squares method inspires us to look for more accurate deep methods for underlying oscillatory solutions.

1.2. Contribution. In evolution equations, more caution is required for the discretization of the time variable than the space variable, because the stability of the solution is usually sensitive to the approximation scheme in time. Therefore, different from the DLS method that deals with time and space simultaneously in (1.2), we consider addressing the numerical solution in time and space separately by two distinct techniques, respectively. On one hand, the time varies merely in one dimension, which allows us to employ traditional numerical techniques for the time discretization. Among all available techniques, the spectral-Galerkin method [21] is well-known for its high accuracy for high-regularity solutions. Especially, if the true solution is C^∞ , then the approximate solution obtained by the spectral-Galerkin method has an exponential decaying error order in terms of the degree of freedom, which is much more accurate than other techniques such as linear multistep methods and Runge–Kutta methods. On the other hand, the space variable (possibly of high dimensions) that is intractable by traditional techniques can be handled appropriately using DNN-based techniques. Consequently, it is promising to develop a hybrid approach combining the traditional high-accuracy Galerkin method and the recent dimension-friendly deep methods.

In this work, we put forward a novel deep adaptive basis Galerkin (DABG) method to solve parabolic and hyperbolic equations. Instead of using a single DNN to approximate the solution as in the previous work, we propose a special adaptive basis structure for approximation. Specifically, the structure is formulated as a tensor product of a sequence of fixed 1-D orthogonal polynomials in time and a sequence of unknown DNNs in space. According to this structure, two computational steps are performed to solve the PDE. First, we apply the Galerkin method to the original equation to eliminate the time variable, leading to a linear system of differential equations having unknown DNNs. Thanks to the orthogonality of the polynomials, the system is sparse with unknown DNNs. Next, we solve the resulting DNN system by minimizing its residual under a DLS framework.

The theoretical analysis is also conducted for the proposed numerical method. We first develop a posteriori estimates that implies the solution error is bounded by the minimal loss function and the term $O(N^{-m})$, where N is the number of basis functions and m is used to characterize the regularity of right hand side function f . Moreover, given that the true solution is a Barron-type function, we derive a priori estimates quantifying the error bound by M and N , where M is the width of the used networks. We show that the solution converges to zero as long as $M = O(N^p)$ and N tends to infinity for some constant power p .

In the numerical experiments, we implement a DABG method on a series of problems to validate its performance. First, for the low-dimensional heat equation, DABG obtains numerical solutions with errors as small as $O(10^{-8})$, which is much more accurate than the compared DLS method. To the best of our knowledge, in the experiments of recently published literature, DNN-based methods can only achieve the best error $O(10^{-4})$. Furthermore, for the high-dimensional problems with oscillatory solutions, DABG can still preserve high accuracy while the DLS method performs quite worse. Besides, an example of a high-dimensional Allen–Cahn equation is considered and solved by DABG, implying the proposed method can be generalized for nonlinear equations. The performance of the proposed method is quite good.

1.3. Organization. This paper is organized as follows. In section 2, we review a Galerkin method with orthogonal polynomial basis for the first- and second-order ordinary differential equations. In section 3, we introduce the adaptive basis structure, followed by the DABG method for parabolic equations being presented. We also

conduct error estimates for the proposed method. In section 4, we derive a similar DABG method for hyperbolic equations as well as the corresponding error estimates. Several numerical examples are demonstrated in section 5 to test the performance of the proposed method. Conclusions and a discussion about further research works are provided in section 6.

2. Preliminaries. In this section, we mainly introduce the Galerkin method for first- and second-order ordinary differential equations, which provides a foundation for the DABG method discussed in the following sections. We specifically use orthogonal polynomials to construct the basis functions because of the high accuracy of the approximation and the sparsity of the derived linear system. Such a method is also referred to as a spectral-Galerkin method. We remark that other types of basis functions can be also applied similarly, such as the finite element method [10]. Throughout this paper, we use (\cdot, \cdot) to denote the L^2 -inner product over $[0, T]$, namely, $(p, q) = \int_0^T p(t)q(t)dt$.

2.1. Orthogonal polynomials. We first introduce the orthogonal polynomials, which will be adopted to construct the basis functions in the following Galerkin method. Let $I := [a, b] \subset \mathbb{R}$ be an interval and N be a positive integer; we use $P^N(I)$ to denote the space of all polynomials in I of degree no greater than N . We also define $P_{0-}^N(I) := \{p \in P^N(I), p(a) = 0\}$ and $P_{0+}^N(I) := \{p \in P^N(I), p(b) = 0\}$ as subspaces of $P^N(I)$.

In the special case $I = [-1, 1]$, Legendre polynomials [12] form an orthogonal basis for $P^N([-1, 1])$ in the sense that

$$(2.1) \quad \int_{-1}^1 L_n(z)L_m(z)dz = \frac{2}{2n+1}\delta_{mn} \quad \forall m, n \geq 0.$$

Also, we have the boundary values

$$(2.2) \quad L_n(\pm 1) = (\pm 1)^n, \quad L'_n(\pm 1) = (\pm 1)^{n-1}n(n+1)/2 \quad \forall n \geq 0.$$

2.2. Spectral-Galerkin methods for 1-D problems. For completeness, we present the spectral-Galerkin methods by solving the following 1-D model problems:

$$(2.3) \quad \begin{cases} \partial_t^k u(t) + \alpha u(t) = f(t), & t \in (0, T], \\ u(0) = 0, & u'(0) = g_0 \text{ if } k = 2 \end{cases}$$

for $k = 1$ or 2 , where $\alpha \in \mathbb{R}$ is a given constant.

2.2.1. First-order equation. First, let us consider the first-order equation (2.3) with $k = 1$. Due to the initial condition $u(0) = 0$, we let $X^{(1)} := \{u \in H^1([0, T]), u(0) = 0\}$ be the trial space, where $H^1([0, T])$ is the first-order Sobolev space in $[0, T]$. Also, we let the space $Y^{(1)} := L^2([0, T])$ be the test space. Multiplying $v \in Y$ on both sides of (2.3) leads to the variational formulation: find $u \in X^{(1)}$ such that

$$(2.4) \quad (u', v) + \alpha(u, v) = (f, v) \quad \forall v \in Y^{(1)}.$$

In the sense of approximation, we let $X_N^{(1)} := P_{0-}^N([0, T]) \subset X^{(1)}$ be the trial space and $Y_N^{(1)} := P^{N-1}([0, T]) \subset Y^{(1)}$ be the test space (note that $\dim(X_N^{(1)}) = \dim(Y_N^{(1)})$), and then we have the Galerkin formulation: find $u_N \in X_N^{(1)}$ such that

$$(2.5) \quad (u'_N, v_N) + \alpha(u_N, v_N) = (f, v_N) \quad \forall v_N \in Y_N^{(1)}.$$

Since the test space is not the same as the trial space, (2.5) is actually a Petrov–Galerkin formulation. Although (2.5) can be solved by various methods such as the finite element method, we proposed the spectral method with orthogonal polynomial-type basis, which has higher accuracy for smooth solutions. Specifically, define

$$(2.6) \quad \phi_n^{(1)}(t) := L_n \left(\frac{2t}{T} - 1 \right) + L_{n-1} \left(\frac{2t}{T} - 1 \right) \quad \forall n \geq 1.$$

Using the property (2.2), it can be verified that $\{\phi_n^{(1)}\}_{n=1}^N$ forms a basis for $X_N^{(1)}$. On the other hand, define

$$(2.7) \quad \psi_n^{(1)}(t) := L_{n-1} \left(\frac{2t}{T} - 1 \right) - L_{n-2} \left(\frac{2t}{T} - 1 \right) \quad \forall n \geq 1$$

with $L_{-1} := 0$, and then $\{\psi_n^{(1)}\}_{n=1}^N$ forms a basis for $Y_N^{(1)}$. The special design of basis functions contributes to the sparsity of the derived linear system (see Proposition 2.1).

Next, we let $u_N(t) = \sum_{n=1}^N \tilde{u}_n \phi_n^{(1)}(t) \in X_N^{(1)}$ be the approximate solution with undetermined coefficients \tilde{u}_n . Taking $v_N = \psi_j^{(1)}$ for $j = 1, \dots, N$ in (2.5) leads to the following linear system:

$$(2.8) \quad \sum_{n=1}^N \left(a_{jn}^{(1)} + \alpha b_{jn}^{(1)} \right) \tilde{u}_n = (f, \psi_j^{(1)}), \quad j = 1, \dots, N,$$

where

$$(2.9) \quad a_{jn}^{(1)} := ((\phi_n^{(1)})', \psi_j^{(1)}), \quad b_{jn}^{(1)} := (\phi_n^{(1)}, \psi_j^{(1)}).$$

The matrix in (2.8) is sparse with band structures due to the following result.

PROPOSITION 2.1. *It satisfies*

$$(2.10) \quad a_{jn}^{(1)} = \begin{cases} 2, & j = 1, \\ -2, & j = n + 1 \geq 2, \\ 0 & \text{otherwise,} \end{cases} \quad b_{jn}^{(1)} = \begin{cases} \frac{T}{2n-1}, & j = n, \\ -\frac{2T}{(2n-1)(2n+1)}, & j = n + 1, \\ -\frac{T}{2n+1}, & j = n + 2, \\ 0 & \text{otherwise.} \end{cases}$$

Proof. From the orthogonality (2.1), note that $\int_{-1}^1 L_n(z)p(z)dz = 0$ for all $p \in P^m([-1, 1])$ if $m < n$. Let us first consider the integral $a_{jn}^{(1)} = \int_0^T (\phi_n^{(1)})' \psi_j^{(1)} dt$. The property (2.2) and integration by parts will be used repeatedly.

In the case $j \geq 2$, if $n < j - 1$, by the transformation $t = T(z + 1)/2$,

$$\int_0^T (\phi_n^{(1)})' \psi_j^{(1)} dt = \int_{-1}^1 (L'_n(z) + L'_{n-1}(z))(L_{j-1}(z) - L_{j-2}(z))dz = 0$$

since $L'_n(z) + L'_{n-1}(z) \in P^{n-1}([-1, 1])$. Similarly, if $n = j - 1$,

$$\begin{aligned} \int_0^T (\phi_{j-1}^{(1)})' \psi_n^{(1)} dt &= \int_{-1}^1 (L'_{j-1}(z) + L'_{j-2}(z))(L_{j-1}(z) - L_{j-2}(z))dz \\ &= - \int_{-1}^1 L'_{j-1}(z)L_{j-2}(z)dz \\ &= \int_{-1}^1 L_{j-1}(z)L'_{j-2}(z)dz - L_{j-1}(1)L_{j-2}(1) + L_{j-1}(-1)L_{j-2}(-1) = -2; \end{aligned}$$

if $n > j - 1$,

$$\begin{aligned} \int_0^T (\phi_n^{(1)})' \psi_j^{(1)} dt &= - \int_0^T \phi_n^{(1)} (\psi_j^{(1)})' dt + \phi_n^{(1)}(T) \psi_j^{(1)}(T) - \phi_n^{(1)}(0) \psi_j^{(1)}(0) \\ &= - \int_0^T \phi_n^{(1)} (\psi_j^{(1)})' dt = \int_{-1}^1 (L_{n-1}(z) + L_n(z))(L'_{j-1}(z) - L'_j(z)) dz = 0. \end{aligned}$$

In the case $j = 1$,

$$\begin{aligned} \int_0^T (\phi_n^{(1)})' \psi_1^{(1)} dt &= \int_{-1}^1 L'_n(z) + L'_{n-1}(z) dz \\ &= L_n(1) + L_{n-1}(1) - L_n(-1) - L_{n-1}(-1) = 2. \end{aligned}$$

The result for $b_{jn}^{(1)}$ can be proved by a similar argument. □

2.2.2. Second-order equation. Next, we briefly present the variational and Galerkin formulations for the second-order equation (2.3) with $k = 2$ derived by a similar argument as above. Define $X^{(2)} := \{u \in H^1([0, T]), u(0) = 0\}$, $Y^{(2)} := \{v \in H^1([0, T]), v(T) = 0\}$, and their approximate spaces $X_N^{(2)} := P_{0-}^N([0, T]) \subset X^{(2)}$, $Y_N^{(2)} := P_{0+}^N([0, T]) \subset Y^{(2)}$. Note that $X^{(2)}$ and $X_N^{(2)}$ are identical to $X^{(1)}$ and $X_N^{(1)}$, respectively, while we use the labels “(1)” and “(2)” to distinguish them in the types of the equation. However, different from $Y^{(1)} = L^2([0, T])$ in the first-order case, $Y^{(2)}$ is taken as the H^1 space with homogeneous Dirichlet condition on the right end point (same for $Y_N^{(2)}$).

We remark that even though the essential condition $u(0) = 0$ is enforced in the trial space $X^{(2)}$, the natural condition $u'(0) = g_0$ is not enforced explicitly and will be absorbed in the bilinear form. The variational formulation is finding $u \in X^{(2)}$ such that

$$(2.11) \quad (u', v') - \alpha(u, v) = -(f, v) - g_0 v(0) \quad \forall v \in Y^{(2)}$$

and its Galerkin formulation is finding $u_N \in X_N^{(2)}$ such that

$$(2.12) \quad (u'_N, v'_N) - \alpha(u_N, v_N) = -(f, v_N) - g_0 v_N(0) \quad \forall v_N \in Y_N^{(2)}.$$

Moreover, define

$$(2.13) \quad \phi_n^{(2)}(t) := \begin{cases} L_1(\frac{2t}{T} - 1) + L_0(\frac{2t}{T} - 1), & n = 1, \\ (1 - \frac{1}{n})L_n(\frac{2t}{T} - 1) + (2 - \frac{1}{n})L_{n-1}(\frac{2t}{T} - 1) + L_{n-2}(\frac{2t}{T} - 1), & n \geq 2, \end{cases}$$

and

$$(2.14) \quad \psi_n^{(2)}(t) := \begin{cases} L_1(\frac{2t}{T} - 1) - L_0(\frac{2t}{T} - 1), & n = 1, \\ (1 - \frac{1}{n})L_n(\frac{2t}{T} - 1) - (2 - \frac{1}{n})L_{n-1}(\frac{2t}{T} - 1) + L_{n-2}(\frac{2t}{T} - 1), & n \geq 2. \end{cases}$$

Using the property (2.2), it can be verified that $\{\phi_n^{(2)}\}_{n=1}^N$ and $\{\psi_n^{(2)}\}_{n=1}^N$ form a basis for $X_N^{(2)}$ and $Y_N^{(2)}$, respectively.

Finally, the approximate solution $u_N = \sum_{n=1}^N \tilde{u}_n \phi_n^{(2)}$ can be determined by solving

$$(2.15) \quad \sum_{n=1}^N \left(a_{jn}^{(2)} - \alpha b_{jn}^{(2)} \right) \tilde{u}_n = -(f, \psi_j^{(2)}) - g_0 \psi_j^{(2)}(0), \quad j = 1, \dots, N,$$

where

$$(2.16) \quad a_{jn}^{(2)} := ((\phi_n^{(2)})', (\psi_j^{(2)})'), \quad b_{jn}^{(2)} := (\phi_n^{(2)}, \psi_j^{(2)}).$$

By the fact that $\partial_t \phi_n^{(2)}(0) = \partial_t \psi_n^{(2)}(T) = 0$ for all $n \geq 2$ and a similar proof as in Proposition 2.1, we can deduce the following result, which implies that the matrix in (2.15) is also sparse with band structures.

PROPOSITION 2.2. $a_{jn}^{(2)} \neq 0$ if and only if $j = 1$ or $n = 1$ or $j = n$; $b_{jn}^{(2)} \neq 0$ if and only if $j = n, n \pm 1, n \pm 2$.

3. Parabolic equations. In this section, we present the deep adaptive basis Galerkin method for linear parabolic equation (1.1) in detail. For simplicity, the inhomogeneous initial and boundary conditions in (1.1) can be converted to homogeneous ones by lifting techniques. Specifically, we can construct a smooth function $v(x, t)$ such that $v(x, 0) = g(x)$ in Ω and $v = h$ on $\partial\Omega \times (0, T)$, and then the function $w := u - v$ satisfies $\frac{\partial w}{\partial t} + \mathcal{L}_x w = f - \frac{\partial v}{\partial t} - \mathcal{L}_x v$ in $\Omega \times (0, T)$ with homogeneous conditions. Therefore, without loss of generality, we consider the following equation:

$$(3.1) \quad \begin{cases} \frac{\partial u(x, t)}{\partial t} + \mathcal{L}_x u(x, t) = f(x, t) & \text{in } \Omega \times (0, T], \\ u(x, 0) = 0 \text{ in } \Omega, \quad u(x, t) = 0 & \text{on } \partial\Omega \times [0, T]. \end{cases}$$

For the readers' convenience, we use the notation $\hat{\cdot}$ to denote functions which involve neural networks.

3.1. Fully connected neural network. We first introduce the fully connected neural network (FNN), which is one of the most widely used neural networks and will be adopted to construct the following adaptive basis. Mathematically speaking, let $\sigma(x)$ be an activation function which is applied entrywise to a vector x to obtain another vector of the same size. Letting $L \in \mathbb{N}^+$ and $M_\ell \in \mathbb{N}^+$ for $\ell = 1, \dots, L-1$, an FNN $\hat{\phi}$ is the composition of $L-1$ simple nonlinear functions given by

$$(3.2) \quad \hat{\phi}(x; \theta) := a^T h_{L-1} \circ h_{L-2} \circ \dots \circ h_1(x) \quad \text{for } x \in \mathbb{R}^d,$$

where $a \in \mathbb{R}^{M_{L-1}}$; $h_\ell(x_\ell) := \sigma(W_\ell x_\ell + b_\ell)$ with $W_\ell \in \mathbb{R}^{M_\ell \times M_{\ell-1}}$ ($M_0 := d$) and $b_\ell \in \mathbb{R}^{M_\ell}$ for $\ell = 1, \dots, L-1$. Here M_ℓ is called the width of the ℓ th layer and L is called the depth of the FNN. $\theta := \{a, W_\ell, b_\ell : 1 \leq \ell \leq L-1\}$ is the set of parameters determining the underlying neural network $\hat{\phi}$. Popular activation functions in solving PDEs include the sigmoidal function $(1 + e^{-x})^{-1}$ and k th power of rectified linear unit (ReLU ^{k}) $\max\{0, x^k\}$ for $k \in \mathbb{N}^+$.

Denote $M := \max\{M_\ell, 1 \leq \ell \leq L-1\}$, and then the number of free parameters $|\theta| = O(M^2 L + Md)$. Comparatively, the degree of freedom of linear structures such as finite elements and tensor product polynomials increases exponentially with d . Hence FNNs are more practicable in high-dimensional approximations. Given an activation function, we consider the architecture with fixed width $M = M_\ell$ for all ℓ and denote $\mathcal{F}_{L, M}$ as the set consisting of all FNNs with depth L and width M .

TABLE 3.1
Possible choices of \mathcal{S} for various types of the problem domain Ω .

Domain type	Formulation of Ω	\mathcal{S}
d -box	$\prod_{i=1}^d (a_i, b_i)$	$\{\hat{w} : \hat{w}(x) = \prod_{i=1}^d (x_i - a_i)(x_i - b_i) \cdot \hat{\phi}, \hat{\phi} \in \mathcal{F}_{L,M}\}$
d -ball	$\{x \in \mathbb{R}^d : x < r\}$	$\{\hat{w} : \hat{w}(x) = (x ^2 - r^2)\hat{\phi}, \hat{\phi} \in \mathcal{F}_{L,M}\}$
Interior of a level set	$\{x \in \mathbb{R}^d : \rho(x) < a\}$	$\{\hat{w} : \hat{w}(x) = (\rho(x) - a)\hat{\phi}, \hat{\phi} \in \mathcal{F}_{L,M}\}$
General Lipschitz	$\Omega \subset \mathbb{R}^d, \partial\Omega$ is $C^{0,1}$	$\{\hat{w} : \hat{w}(x) = \text{dist}(x, \partial\Omega)\hat{\phi}, \hat{\phi} \in \mathcal{F}_{L,M}\}$

3.2. Adaptive basis structures. Recall that for 1-D problems in section 2.2, we use the linear expansion of $\{\phi_n^{(1)}\}_{n=1}^N$ for approximation. Based on that, for multidimensional parabolic equations, we replace the scalar coefficients \tilde{u}_n by space-dependent basis functions. Considering the spatial dimension d might be moderately large, these bases can be constructed based on FNNs, thanks to their capability of high dimensions. Specifically, we aim to look for approximate solutions of the following form:

$$(3.3) \quad \hat{u}_N(x, t) = \sum_{n=1}^N \hat{w}_n(x) \phi_n^{(1)}(t),$$

where $\hat{w}_n : \mathbb{R}^d \rightarrow \mathbb{R}$ are functions belonging to a specified FNN-based class \mathcal{S} . Clearly, \mathcal{S} should have regularity such that $\mathcal{L}_x \hat{w}$ is meaningful for all $\hat{w} \in \mathcal{S}$.

We expect the expansion (3.3) to satisfy the homogeneous boundary condition $\hat{u}_N(x, t) = 0$ on $\partial\Omega \times (0, T)$. A convenient way is to enforce the boundary condition on \hat{w}_n . That is, we expect that $\hat{w}_n(x) = 0$ on $\partial\Omega$ for all n . By this setting, the boundary condition is automatically satisfied by \hat{u}_N , and we do not have to deal with the boundary condition explicitly in the formulation. Consequently, we require that all functions in \mathcal{S} vanish on $\partial\Omega$.

On the other hand, we need to ensure that the functions in \mathcal{S} have FNN structures fundamentally. Therefore, we design \mathcal{S} as $\mathcal{S} = \{\hat{w} : \hat{w}(x) = \nu(x)\hat{\phi}(x), \hat{\phi} \in \mathcal{F}_{L,M}\}$, where ν is a prescribed smooth function governing the boundary condition, namely, $\nu|_{\partial\Omega} = 0$. Possible choices of \mathcal{S} for various types of Ω are listed in Table 3.1. In section 3.5, we will show that if Ω is smooth, there is always a boundary governing function ν with the same smoothness (Lemma 3.8).

To sum up, we define the hypothesis space

$$(3.4) \quad \mathcal{U}_N^P := \left\{ \hat{u}_N(x, t) = \sum_{n=1}^N \hat{w}_n(x) \phi_n^{(1)}(t), \hat{w}_n \in \mathcal{S} \right\},$$

where the label ‘‘P’’ is short for the parabolic equation. We can formally regard \mathcal{U}_N^P as a tensor-product class, i.e., $\mathcal{U}_N^P = \mathcal{S} \times X_N^{(1)}$. Since the basis functions of the structure \hat{u}_N in (3.4) include neural networks, which are nonlinear structures with the free parameters, we can view \hat{u}_N as an adaptive basis structure. Moreover, all functions in \mathcal{U}_N^P always satisfy the initial and boundary conditions in (3.1). It suffices to look for approximate solutions in \mathcal{U}_N^P that satisfy the differential equation.

We remark that even though we propose to take N independent FNNs in the adaptive basis structure (3.4), one can alternatively use a vector-valued FNN with N -dimensional output in the structure. More precisely, an N -dimensional vector-valued FNN $\hat{\phi}$ is defined as (3.2) by specifying $a \in \mathbb{R}^{M_{L-1} \times N}$, and the n th component of

$\hat{\phi}$ can be taken to build $\hat{w}_n(x)$. It is clear that using one FNN with N -dimensional output is equivalent to using N scalar FNNs sharing the same input/hidden layers h_ℓ but having different output layers a . Therefore, the hypothesis space constructed by one FNN with N -dimensional output is a subspace of the one with N scalar FNNs. For a more general discussion, we choose to use the latter case in this paper.

3.3. Galerkin formulation. Following the method for the first-order problem in section 2.2.1, we can formally derive the variational form for the parabolic problem (3.1). Recalling $X^{(1)}$, $Y^{(1)}$ and $Y_N^{(1)}$ are defined in section 2.2.1, we define the trial space

$$(3.5) \quad \mathcal{U}^P := \left\{ u(x, t) : u(x, \cdot) \in X^{(1)} \quad \forall x \in \Omega, \quad u|_{\partial\Omega \times [0, T]} = 0 \right\},$$

and then the variational form is finding $u \in \mathcal{U}^P$ such that

$$(3.6) \quad (\partial_t u(x, \cdot), v) + (\mathcal{L}_x u(x, \cdot), v) = (f(x, \cdot), v) \quad \forall v \in Y^{(1)}, \quad \forall x \in \Omega.$$

Accordingly, the Galerkin formulation is finding $\hat{u}_N \in \mathcal{U}_N^P$ such that

$$(3.7) \quad (\partial_t \hat{u}_N(x, \cdot), v_N) + (\mathcal{L}_x \hat{u}_N(x, \cdot), v_N) = (f(x, \cdot), v_N) \quad \forall v_N \in Y_N^{(1)}, \quad \forall x \in \Omega.$$

Taking \hat{u}_N as the expression (3.3) and $v_N = \psi_j^{(1)}$ for $j = 1, \dots, N$, we obtain

$$(3.8) \quad \sum_{n=1}^N \left(a_{jn}^{(1)} \hat{w}_n(x) + b_{jn}^{(1)} \mathcal{L}_x \hat{w}_n(x) \right) = \left(f(x, \cdot), \psi_j^{(1)} \right), \quad j = 1, \dots, N, \quad \forall x \in \Omega,$$

with unknowns $\{\hat{w}_n\} \subset \mathcal{S}$, where $a_{jn}^{(1)}$ and $b_{jn}^{(1)}$ are defined in (2.9). Noting that (3.8) holds for all $x \in \Omega$, we may not find a set of $\{\hat{w}_n\}$ exactly making (3.8) satisfied. Therefore, in a practical way, we solve (3.8) by minimizing its residual in the least-squared sense,

$$(3.9) \quad \min_{\hat{w}_n \in \mathcal{S}} J^P[\hat{w}_1, \dots, \hat{w}_N] := \sum_{j=1}^N r_j \left\| R_j^{(1)}[\hat{w}_1, \dots, \hat{w}_N] \right\|_{L^2(\Omega)}^2 + \lambda N^{-4} \|\mathcal{L}_x \hat{w}_N\|_{L^2(\Omega)}^2,$$

where $R_j^{(1)}[\hat{w}_1, \dots, \hat{w}_N] := \sum_{n=1}^N (a_{jn}^{(1)} \hat{w}_n(x) + b_{jn}^{(1)} \mathcal{L}_x \hat{w}_n(x)) - (f(x, \cdot), \psi_j^{(1)})$ is the residual of the j th equation in (3.8); $r_j := N^{-3} \sum_{k=j}^N k(2k-1)$ between $O(N^{-1})$ and $O(1)$ is a special weight of $R_j^{(1)}$; and $\lambda > 0$ is a given regularization parameter. Since the loss function is nonnegative, the optimization (3.9) always admits a solution. We remark that in (3.9) the weights $\{r_j\}$ and the regularization term $\lambda N^{-4} \|\mathcal{L}_x \hat{w}_N\|_{L^2(\Omega)}^2$ are specifically chosen such that a decent a posteriori estimate can be derived (see section 3.4). Also, by Proposition 2.1, the sum $\sum_{n=1}^N (a_{jn}^{(1)} \hat{w}_n(x) + b_{jn}^{(1)} \mathcal{L}_x \hat{w}_n(x))$ in $R_j^{(1)}$ only includes at most four nonzero terms if $j \geq 2$.

It is worth noting that even though the regularization term $\lambda N^{-4} \|\mathcal{L}_x \hat{w}_N\|_{L^2(\Omega)}^2$ is essential in the error analysis, it is less important in practice because it is much smaller than the residual part of the loss. More precisely, using (2.6) we obtain

$$(3.10) \quad \begin{aligned} \mathcal{L}_x \hat{u}_N = & \mathcal{L}_x \hat{w}_1(x) L_0 \left(\frac{2t}{T} - 1 \right) + \sum_{n=1}^{N-1} (\mathcal{L}_x \hat{w}_{n+1}(x) \\ & + \mathcal{L}_x \hat{w}_n(x)) L_n \left(\frac{2t}{T} - 1 \right) + \mathcal{L}_x \hat{w}_N(x) L_N \left(\frac{2t}{T} - 1 \right). \end{aligned}$$

Namely, $\mathcal{L}_x \hat{w}_N(x)$ is the coefficient associated with $L_N(\frac{2t}{T} - 1)$ in the orthogonal expansion of $\mathcal{L}_x \hat{u}_N$ with the basis $\{L_n(\frac{2t}{T} - 1)\}_{n=0}^N$. Due to the orthogonality, it holds that for each $x \in \Omega$,

$$(3.11) \quad \mathcal{L}_x \hat{w}_N(x) = \frac{\int_0^T \mathcal{L}_x \hat{u}_N(x, t) \cdot L_N(\frac{2t}{T} - 1) dt}{\int_0^T |L_N(\frac{2t}{T} - 1)|^2 dt} = \frac{2N + 1}{T} \int_0^T \mathcal{L}_x \hat{u}_N(x, t) \cdot L_N\left(\frac{2t}{T} - 1\right) dt.$$

Using the identity of Legendre polynomials

$$(3.12) \quad L_n(x) = (2n + 1)^{-1}(L'_{n+1}(x) - L'_{n-1}(x))$$

and integration by parts, we further obtain

$$(3.13) \quad \begin{aligned} & \int_0^T \mathcal{L}_x \hat{u}_N(x, t) \cdot L_N\left(\frac{2t}{T} - 1\right) dt \\ &= -\frac{T}{4N + 2} \left(\int_0^T \mathcal{L}_x \partial_t \hat{u}_N(x, t) L_{N+1}\left(\frac{2t}{T} - 1\right) dt \right. \\ & \quad \left. - \int_0^T \mathcal{L}_x \partial_t \hat{u}_N(x, t) L_{N-1}\left(\frac{2t}{T} - 1\right) dt \right). \end{aligned}$$

Since $|L_n(x)| \leq 1$ in $[-1, 1]$ for all n , it follows (3.13) that

$$(3.14) \quad \left| \int_0^T \mathcal{L}_x \hat{u}_N(x, t) \cdot L_N\left(\frac{2t}{T} - 1\right) dt \right| \leq CN^{-1} \|\mathcal{L}_x \hat{u}_N(x, \cdot)\|_{H^1([0, T])}$$

for some constant C only depending on T . Note that \hat{u}_N is infinitely differentiable in t , so we can do integration by parts on the resulting terms of (3.13) recursively for any number of times, and it finally leads to

$$(3.15) \quad \begin{aligned} |\mathcal{L}_x \hat{w}_N(x)| &= \frac{2N + 1}{T} \left| \int_0^T \mathcal{L}_x \hat{u}_N(x, t) \cdot L_N\left(\frac{2t}{T} - 1\right) dt \right| \\ &\leq CN^{1-p} \|\mathcal{L}_x \hat{u}_N(\cdot, x)\|_{H^p([0, T])}, \end{aligned}$$

where the integer p can be arbitrarily large. This implies the regularization term $\lambda N^{-4} \|\mathcal{L}_x \hat{w}_N\|_{L^2(\Omega)}^2$ decays exponentially to zero as $N \rightarrow \infty$. Therefore, if N is moderately large, the regularization term can be neglected compared with the residual part of the loss that decays to zero in an algebraic rate as $M, N \rightarrow \infty$ (in the proof of Theorem 3.9).

3.4. A posteriori estimates. We conduct error analysis for the proposed DABG formulation (3.9). First, given an interval $I = [a, b]$, we introduce the weighted L^2 -space with weight function $(b - z)^\alpha(z - a)^\alpha$, namely,

$$(3.16) \quad L_\alpha^2(I) := \left\{ u : I \rightarrow \mathbb{R} : \int_I |u|^2 (b - z)^\alpha (z - a)^\alpha dz < \infty \right\}, \quad \alpha \in \mathbb{R},$$

equipped with the norm $\|u\|_{L_\alpha^2(I)} := (\int_I |u|^2 (b - z)^\alpha (z - a)^\alpha dz)^{\frac{1}{2}}$, and the non-uniformly Jacobi-weighted Sobolev space

$$(3.17) \quad \mathcal{W}^m(I) := \left\{ u : I \rightarrow \mathbb{R} : \partial_z^k u \in L_k^2(I), \quad 0 \leq k \leq m \right\}, \quad m \in \mathbb{N},$$

equipped with the norm $\|u\|_{\mathcal{W}^m(I)} := (\sum_{k=0}^m \|\partial_z^k u\|_{L_k^2(I)}^2)^{\frac{1}{2}}$. Also, we define the L^2 -orthogonal projection $\pi_I^N : L^2(I) \rightarrow P^N(I)$ such that

$$(3.18) \quad \int_I (\pi_I^N u - u) v dz = 0 \quad \forall v \in P^N(I)$$

given $u \in L^2(I)$. In the case $I = [-1, 1]$, noting that the Legendre polynomials $\{L_n\}_{n=0}^\infty$ forms an orthogonal basis for $L^2([-1, 1])$, if we expand $u(z) = \sum_{n=0}^\infty \tilde{u}_n L_n(z)$, then $\pi_{[-1,1]}^N u(z) = \sum_{n=0}^N \tilde{u}_n L_n(z)$. The following lemmas show the approximation error between u and $\pi_{[-1,1]}^N u$.

LEMMA 3.1 (Theorem 3.35 [21]). *For any $u \in \mathcal{W}^m([-1, 1])$, if $N \gg m$, we have*

$$(3.19) \quad \|\pi_{[-1,1]}^N u - u\|_{L^2([-1,1])} \leq cN^{-m} \|\partial_z^m u\|_{L_m^2([-1,1])},$$

where c is a constant number.

By rescaling the interval, we directly have the following.

COROLLARY 3.2. *For any $u \in \mathcal{W}^m([0, T])$, if $N \gg m$, we have*

$$(3.20) \quad \|\pi_{[0,T]}^N u - u\|_{L^2([0,T])} \leq cN^{-m} \|\partial_t^m u\|_{L_m^2([0,T])},$$

where c is a constant number only depending on m and T .

In the following passage, we will apply $\pi_{[0,T]}^N$ to functions of variables x and t , which means we apply the orthogonal projection $\pi_{[0,T]}^N$ with respect to t for each x pointwise.

Now we derive the a posteriori estimate for the DABG method. For simplicity, we only consider the heat equation with $\mathcal{L}_x = -\Delta_x$. It can be derived that the error is bounded by the final minimized loss J^P and $O(N^{-m})$ if f has $\mathcal{W}^m([0, T])$ regularity in t . The analysis can be easily generalized for other types of parabolic equations.

THEOREM 3.3. *Let u be the classical solution of (3.1) with $\mathcal{L}_x = -\Delta_x$. Let $m \geq 0$ be some integer. Suppose that $f \in \mathcal{W}^m(0, T; L^2(\Omega))$, $u \in L^2(0, T; H_0^1(\Omega))$, and $\partial_t u \in L^2(0, T; H^{-1}(\Omega))$. Let $\hat{u}_N(x, t) = \sum_{n=1}^N \hat{w}_n(x) \phi_n^{(1)}(t)$, where $\{\hat{w}_n\}_{n=1}^N$ solves the minimization (3.9). Suppose that $\hat{w}_n \in H_0^1(\Omega) \cap H^{-1}(\Omega)$ is twice differentiable for $n = 1, \dots, N$. Denote $\delta = J^P[\hat{w}_1, \dots, \hat{w}_N]$ as the minimized loss. Then*

$$(3.21) \quad \begin{aligned} & \sup_{0 \leq t \leq T} \|u(\cdot, t) - \hat{u}_N(\cdot, t)\|_{H_0^1(\Omega)} + \|u - \hat{u}_N\|_{L^2(0,T;H^2(\Omega))} + \|\partial_t(u - \hat{u}_N)\|_{L^2(0,T;L^2(\Omega))} \\ & \leq c \left(N^{\frac{3}{2}} \sqrt{\delta} + N^{-m} \|\partial_t^m f\|_{L_m^2(0,T;L^2(\Omega))} \right) \end{aligned}$$

provided that $N \gg m$, where c is a constant only depending on Ω , m , T , and λ .

Proof. Let $\hat{e}_N := \hat{u}_N - u$, and then it satisfies

$$(3.22) \quad \begin{cases} \partial_t \hat{e}_N(x, t) - \Delta_x \hat{e}_N(x, t) = \hat{p}(x, t) := \partial_t \hat{u}_N(x, t) - \Delta_x \hat{u}_N(x, t) - f(x, t) & \text{in } \Omega \times (0, T], \\ \hat{e}_N(x, 0) = 0 & \text{in } \Omega, \quad \hat{e}_N(x, t) = 0 & \text{on } \partial\Omega \times [0, T]. \end{cases}$$

By the hypothesis, $\hat{p} \in L^2(0, T; L^2(\Omega))$, $\hat{e}_N \in L^2(0, T; H_0^1(\Omega))$, and $\partial_t \hat{e}_N \in L^2(0, T; H^{-1}(\Omega))$. Then using a priori estimates for heat equations (e.g., Theorem 5, p. 382, in [11]), we can obtain

$$(3.23) \quad \sup_{0 \leq t \leq T} \|\hat{e}_N(\cdot, t)\|_{H_0^1(\Omega)} + \|\hat{e}_N\|_{L^2(0,T;H^2(\Omega))} + \|\partial_t \hat{e}_N\|_{L^2(0,T;L^2(\Omega))} \leq c_1 \|\hat{p}\|_{L^2(0,T;L^2(\Omega))},$$

where c_1 only depends on Ω and T . It suffices to estimate the right hand side.

Denote $\tilde{L}_n(t) := L_n(\frac{2t}{T} - 1)$; then $\{\tilde{L}_n\}_{n=0}^N$ is an orthogonal basis for $P^N([0, T])$, and $\{\tilde{L}_n\}_{n=0}^\infty$ is an orthogonal Schauder basis for $L^2([0, T])$. Especially, by (2.1) we have

$$(3.24) \quad \int_0^T \tilde{L}_n(t)\tilde{L}_m(t)dt = \frac{T}{2n+1} \delta_{mn} \quad \forall m, n \in \mathbb{N}.$$

Note that

$$(3.25) \quad \begin{aligned} \hat{p}(x, t) &= \sum_{n=1}^N \left(\hat{w}_n(x) \partial_t \phi_n^{(1)}(t) - \Delta_x \hat{w}_n(x) \phi_n^{(1)}(t) \right) - \pi_{[0,T]^{N-1}} f + (\pi_{[0,T]^{N-1}} f - f) \\ &= \left(\sum_{n=1}^N \hat{w}_n(x) \partial_t \phi_n^{(1)}(t) - \sum_{n=1}^{N-1} \Delta_x \hat{w}_n(x) \phi_n^{(1)}(t) - \Delta_x \hat{w}_N(x) \tilde{L}_{N-1}(t) - \pi_{[0,T]^{N-1}} f \right) \\ &\quad + \left(\pi_{[0,T]^{N-1}} f - f - \Delta_x \hat{w}_N(x) \tilde{L}_N(t) \right) := I_1(x, t) + I_2(x, t). \end{aligned}$$

Clearly, (3.25) is an orthogonal decomposition on the subspace $P^{N-1}([0, T])$, namely, $I_1 = \pi_{[0,T]^{N-1}} \hat{p}$ and $I_2 = \hat{p} - \pi_{[0,T]^{N-1}} \hat{p}$. Then we have

$$(3.26) \quad \begin{aligned} \|\hat{p}\|_{L^2(0,T;L^2(\Omega))} &\leq \|\pi_{[0,T]^{N-1}} \hat{p}\|_{L^2(0,T;L^2(\Omega))} \\ &\quad + \|\pi_{[0,T]^{N-1}} f - f\|_{L^2(0,T;L^2(\Omega))} + \|\Delta_x \hat{w}_N(x) \tilde{L}_N(t)\|_{L^2(0,T;L^2(\Omega))}. \end{aligned}$$

For the first term in (3.26), note that $\pi_{[0,T]^{N-1}} \hat{p}(x, t) = \sum_{n=0}^{N-1} \tilde{p}_n(x) \tilde{L}_n(t)$ with

$$(3.27) \quad \tilde{p}_n(x) = \frac{\int_0^T \hat{p}(x, t) \tilde{L}_n(t) dt}{\int_0^T |\tilde{L}_n(t)|^2 dt} = \frac{2n+1}{T} \int_0^T \hat{p}(x, t) \tilde{L}_n(t) dt.$$

Also, noting that $\tilde{L}_n = \sum_{j=1}^{n+1} \psi_j^{(1)}$ for $0, \dots, N-1$, then for every $x \in \Omega$,

$$(3.28) \quad \begin{aligned} \sum_{n=0}^{N-1} (2n+1) \left(\hat{p}(x, \cdot), \tilde{L}_n \right)^2 &= \sum_{n=0}^{N-1} \left(\sum_{j=1}^{n+1} \sqrt{2n+1} \left(\hat{p}(x, \cdot), \psi_j^{(1)} \right) \right)^2 \\ &\leq \sum_{n=0}^{N-1} (n+1) \sum_{j=1}^{n+1} (2n+1) \left(\hat{p}(x, \cdot), \psi_j^{(1)} \right)^2 \\ &= \sum_{j=1}^N \sum_{n=j-1}^{N-1} (n+1)(2n+1) \left(\hat{p}(x, \cdot), \psi_j^{(1)} \right)^2 \\ &= N^3 \sum_{j=1}^N r_j \left(\hat{p}(x, \cdot), \psi_j^{(1)} \right)^2 = N^3 \sum_{j=1}^N r_j \left| R_j^{(1)}[\hat{w}_1(x), \dots, \hat{w}_N(x)] \right|^2. \end{aligned}$$

Therefore, by (3.28),

(3.29)

$$\begin{aligned} \|\pi_{[0,T]}^{N-1} \hat{p}\|_{L^2(0,T;L^2(\Omega))}^2 &= \int_{\Omega} \left\| \sum_{n=0}^{N-1} \tilde{p}_n(x) \tilde{L}_n(\cdot) \right\|_{L^2([0,T])}^2 dx \\ &= \int_{\Omega} \int_0^T \sum_{n=0}^{N-1} \tilde{p}_n^2(x) \tilde{L}_n^2(t) dt dx = \sum_{n=0}^{N-1} \int_{\Omega} \tilde{p}_n^2(x) dx \int_0^T \tilde{L}_n^2(t) dt \\ &\leq \sum_{n=0}^{N-1} \frac{2n+1}{T} \left\| \int_0^T \hat{p}(x,t) \tilde{L}_n(t) dt \right\|_{L^2(\Omega)}^2 \leq \frac{N^3}{T} \sum_{j=1}^N r_j \|R_j[\hat{w}_1, \dots, \hat{w}_N]\|_{L^2(\Omega)}^2. \end{aligned}$$

For the second term in (3.26), using Corollary 3.2 we have if $N \gg m$,

(3.30)
$$\|\pi_{[0,T]}^{N-1} f - f\|_{L^2(0,T;L^2(\Omega))}^2 \leq c_2 N^{-2m} \|\partial_t^m f\|_{L_m^2(0,T;L^2(\Omega))}^2,$$

where c_2 only depends on m and T .

For the third term in (3.26), using (3.24) we have

(3.31)
$$\begin{aligned} \|\Delta_x \hat{w}_N(x) \tilde{L}_N(t)\|_{L^2(0,T;L^2(\Omega))} &= \|\Delta_x \hat{w}_N\|_{L^2(\Omega)} \|\tilde{L}_N\|_{L^2([0,T])} \\ &= \sqrt{\frac{T}{2N+1}} \|\Delta_x \hat{w}_N\|_{L^2(\Omega)}. \end{aligned}$$

Combining (3.29), (3.30), and (3.31) completes the proof. □

Remark 3.1. In Theorem 3.3, the derived error bound has a quantity $O(N^{\frac{3}{2}})$ multiplied to the minimal loss $\sqrt{\delta}$. This quantity can also be interpreted as the condition number of the transformation matrix between the used basis $\{\psi_j^{(1)}\}_{j=1}^N$ and the orthonormal basis $\{\sqrt{\frac{2n+1}{T}} \tilde{L}_n\}_{n=0}^{N-1}$ of the test space $Y_N^{(1)}$. More precisely, we have $[\sqrt{\frac{1}{T}} \tilde{L}_0 \dots \sqrt{\frac{2N-1}{T}} \tilde{L}_{N-1}]^\top = D[\psi_1^{(1)} \dots \psi_N^{(1)}]^\top$ with $D_{ij} = \sqrt{\frac{2i-1}{T}}$ if $j \leq i$ and $D_{ij} = 0$ if $j > i$. The 2-condition number of D is estimated as $O(N^{\frac{3}{2}})$. A slightly different proof can be derived from this fact.

3.5. A priori error estimates. Now let us investigate how small the loss function J^P in (3.9) can be minimized if we use special hypothesis space \mathcal{S} . Recalling that $\mathcal{S} = \{\hat{w} = \nu \hat{\phi}, \hat{\phi} \in \mathcal{F}_{L,M}\}$ has an FNN-based structure, it is expected that \mathcal{S} with larger depth L or width W leads to smaller optimal J^P . More precisely, we expect the convergence $\min_{\hat{w}_n \in \mathcal{S}} J^P \rightarrow 0$ if $L, M \rightarrow \infty$. If this holds, then Theorem 3.3 directly shows that the solution error converges to zero as $L, M, N \rightarrow \infty$ in some sense.

3.5.1. Barron function and its approximation. The analysis is essentially based on the approximation property of FNNs for special functions. Although neural network approximation theory has been widely developed in recent years, we will especially consider in this paper the approximation error for Barron-type functions in H^2 norm. It has been studied that the approximation errors of two-layer FNNs for Barron-type functions have dimension-independent rates with respect to the width [1, 2, 8, 7, 25, 26, 5]. We remark that stronger error bounds that overcome the curse of dimensionality can be achieved by using FNNs with advanced activation functions [24, 23, 22, 16].

Consider the following class of two-layer FNNs with M neurons:

$$(3.32) \quad \mathcal{F}_{2,M} = \left\{ \hat{\phi}(x; \theta) = \sum_{m=1}^M a_m \sigma(w_m^\top x + b_m) : w_m \in \mathbb{R}^d, a_m, b_m \in \mathbb{R} \right\},$$

where $\theta := \{a_m, w_m, b_m\}_{1 \leq m \leq M}$ is the set of parameters. Note the definition (3.32) coincides with the generic FNN (3.2) when $L = 2$.

Also, given a function $f : \mathbb{R}^d \rightarrow \mathbb{R}$, its s th-order Barron norm is defined as $\|f\|_{\mathcal{B}^s} = \int_{\mathbb{R}^d} (1 + |\omega|)^s |\mathcal{F}[f]| d\omega$, where $\mathcal{F}[f](\omega) := (2\pi)^{-d/2} \int_{\mathbb{R}^d} f(x) e^{-i\omega^\top x} dx$ is the Fourier transform of f . We use \mathcal{B}^s to denote the Banach space of functions having bounded Barron norm, namely, the Barron space. It is shown that \mathcal{B}^s is continuously embedded in $H^s(\Omega)$ if Ω is bounded [25].

The following lemma shows differentiable functions with compact support, and sufficiently high-order derivatives are in \mathcal{B}^s .

LEMMA 3.4. *Let $s \geq 1$ and $f \in C_c^\gamma(\mathbb{R}^d)$ with $\gamma > d/2 + s$; then $f \in \mathcal{B}^s$ in the sense that*

$$(3.33) \quad \|f\|_{\mathcal{B}^s}^2 \leq c \int_{\mathbb{R}^d} |f|^2 + |D^\gamma f|^2 dx < \infty$$

for some constant c only depending on γ and s .

Proof.

$$(3.34) \quad \begin{aligned} \|f\|_{\mathcal{B}^s} &= \int_{\mathbb{R}^d} (1 + |\omega|)^s |\mathcal{F}[f]| d\omega \\ &= \int_{\mathbb{R}^d} (1 + |\omega|^{2(\gamma-s)})^{-1/2} \cdot (1 + |\omega|^{2(\gamma-s)})^{1/2} (1 + |\omega|)^s |\mathcal{F}[f]| d\omega. \end{aligned}$$

Since $\gamma > d/2 + s$, $\int_{\mathbb{R}^d} (1 + |\omega|^{2(\gamma-s)})^{-1} d\omega$ is a finite number depending on γ and s . So using a Cauchy-Schwarz inequality on (3.34) leads to

$$(3.35) \quad \|f\|_{\mathcal{B}^s}^2 \leq c \int_{\mathbb{R}^d} (1 + |\omega|^{2\gamma}) |\mathcal{F}[f]|^2 d\omega.$$

Using the identity $\mathcal{F}[\partial_x^\alpha f] = (i\omega)^\alpha \mathcal{F}[f]$ for each multi-index α satisfying $|\alpha| \leq \gamma$ and Parseval's identity, it follows (3.35) that

$$(3.36) \quad \|f\|_{\mathcal{B}^s}^2 \leq c \int_{\mathbb{R}^d} |\mathcal{F}[f]|^2 + |\mathcal{F}[D^\gamma f]|^2 d\omega = c \int_{\mathbb{R}^d} |f|^2 + |D^\gamma f|^2 dx,$$

which is finite since $f \in C_c^\gamma(\mathbb{R}^d)$. □

Next, we introduce the approximation property of FNNs, which implies any function in the Barron space can be efficiently approximated by two-layer FNNs in Sobolev norms.

LEMMA 3.5 (Corollary 1 [25]). *Let $s \in \mathbb{N}$. Suppose the activation function of $\mathcal{F}_{2,M}$ is set as the sigmoidal function $\sigma(x) = (1 + e^{-x})^{-1}$, arctan $\sigma(x) = \arctan(x)$, hyperbolic tangent $\sigma(x) = \tanh(x)$, or SoftPlus $\sigma(x) = \log(1 + e^x)$. Then for any $f \in \mathcal{B}^{s+1}$, we have*

$$(3.37) \quad \inf_{\hat{\phi} \in \mathcal{F}_{2,M}} \|\hat{\phi} - f\|_{H^s(\Omega)} \leq |\Omega|^{\frac{1}{2}} C(s, \text{diam}(\Omega), \sigma) \|f\|_{\mathcal{B}^{s+1}} M^{-\frac{1}{2}}.$$

3.5.2. Continuous extension in \mathbb{R}^d . We should be careful that the PDE solution is spatially defined in a bounded domain Ω rather than \mathbb{R}^d . To apply the preceding approximation property, we choose to use extension approaches to extend the continuous solutions from $\bar{\Omega} \times [0, T]$ to $\mathbb{R}^d \times [0, T]$. Denote $C_{0-}^k([0, T]) := \{f \in C^k([0, T]) : f(0) = 0\}$ (we omit k if $k = 0$). The existence of a continuous extension is given by the following lemma.

LEMMA 3.6. *Letting $f(x, t) \in C_{0-}^k(0, T; C^\gamma(\bar{\Omega}))$ with some $\gamma \geq 0$, there exists some extension $f_{\text{ext}} \in C_{0-}^k(0, T; C_c^\gamma(\mathbb{R}^d))$ in the sense that $f_{\text{ext}}(x, t) = f(x, t)$ if $(x, t) \in \bar{\Omega} \times [0, T]$.*

Proof. Note that $\{\phi_n^{(1)}\}_{n=1}^\infty$ forms a Schauder basis for $C_{0-}([0, T])$, so f has an expansion $f(x, t) = \sum_{n=1}^\infty \tilde{f}_n(x)\phi_n^{(1)}(t)$ with $\tilde{f}_n(x) \in C^\gamma(\bar{\Omega})$. Now given a closed subset $\Lambda \subset \mathbb{R}^d$ satisfying $\bar{\Omega} \subset \subset \Lambda$, using continuous extension (e.g., Whitney extension theorem) we can always find some $\tilde{f}_{n,\text{ext}} \in C_c^\gamma(\mathbb{R}^d)$ such that $\tilde{f}_{n,\text{ext}} = \tilde{f}_n$ in $\bar{\Omega}$ and $\tilde{f}_{n,\text{ext}} = 0$ in $\mathbb{R}^d \setminus \Lambda$ for all n . Then the desired extension is formed by $f_{\text{ext}}(x, t) := \sum_{n=1}^\infty \tilde{f}_{n,\text{ext}}(x)\phi_n^{(1)}(t)$. It is clear that $f_{\text{ext}}(x, t) = f(x, t)$ in $\bar{\Omega} \times [0, T]$. Moreover, $f_{\text{ext}}(x, 0) = 0$ for all $x \in \bar{\Omega}$, and f_{ext} is C^k in t follows that $f(x, t)$ is C^k in t . \square

Next, let us define a special norm for functions in $C(0, T; C_c^\gamma(\mathbb{R}^d))$. Recall that $\{\tilde{L}_n := L(\frac{2t}{T} - 1)\}_{n=0}^\infty$ forms a Schauder basis for $C([0, T])$. So, given $f \in C(0, T; C_c^\gamma(\mathbb{R}^d))$, f has an expansion $f(x, t) = \sum_{n=0}^\infty \tilde{f}_n(x)\tilde{L}_n(t)$. We define $\|f\|_{L_*^2} := (\sum_{n=0}^\infty \int_{\mathbb{R}^d} |\tilde{f}_n|^2 dx)^{1/2}$. Note that $\{\tilde{f}_n\}$ are the coefficients associated with the L^2 -orthogonal basis $\{\tilde{L}_n\}$ in time variable, so $\|\cdot\|_{L_*^2}$ is actually a weighted L^2 norm over $\mathbb{R}^d \times [0, T]$ (but not the standard L^2 norm because $\{\tilde{L}_n\}$ is not orthonormal). We show that $\|f\|_{L_*^2}$ is always bounded by the standard L^2 norm of $\partial_t^2 f$ if f is C^2 smooth in t by the following lemma.

LEMMA 3.7. *Supposing $f \in C^2(0, T; C_c(\mathbb{R}^d))$ with $\gamma \geq 0$, it holds that*

$$(3.38) \quad \|f\|_{L_*^2}^2 \leq \frac{5T^3}{4} \|\partial_t^2 f\|_{L^2(\mathbb{R}^d \times [0, T])}^2 < \infty.$$

Proof. By the orthogonality relation (3.24), we have

$$(3.39) \quad \tilde{f}_n = \frac{\int_0^T f(x, t)\tilde{L}_n(t)dt}{\int_0^T |\tilde{L}_n(t)|^2 dt} = \frac{2n+1}{T} \int_0^T f(x, t)L_n\left(\frac{2t}{T} - 1\right) dt.$$

Using the identity (3.12) and integration by parts, we can obtain

$$(3.40) \quad \begin{aligned} & \left| \int_0^T f(x, t)L_n\left(\frac{2t}{T} - 1\right) dt \right| \\ &= \left| \frac{T}{4n+2} \int_0^T \partial_t f(x, t) \left(L_{n+1}\left(\frac{2t}{T} - 1\right) - L_{n-1}\left(\frac{2t}{T} - 1\right) \right) dt \right| \\ &\leq \frac{T}{4n+2} \left(\left| \int_0^T \partial_t f(x, t)L_{n+1}\left(\frac{2t}{T} - 1\right) dt \right| + \left| \int_0^T \partial_t f(x, t)L_{n-1}\left(\frac{2t}{T} - 1\right) dt \right| \right). \end{aligned}$$

Following (3.40), we repeat doing the preceding step with integration by parts on the last two terms, and then it leads to

$$(3.41) \quad \left| \int_0^T f(x, t) L_n \left(\frac{2t}{T} - 1 \right) dt \right| \leq 4 \left(\frac{T}{4n+2} \right)^2 \int_0^T |\partial_t^2 f(x, t)| dt,$$

in which we use the boundedness $|L_n(x)| \leq 1$ for all $x \in [-1, 1]$ and all n . Combining (3.39) and (3.41) and using Cauchy–Schwarz inequality, we have $|\tilde{f}_n|^2 \leq \frac{T^3}{(2n+1)^2} \int_0^T |\partial_t^2 f(x, t)|^2 dt$, which completes the proof with the relation $\sum_{n=0}^\infty (2n+1)^{-2} < 5/4$. \square

Using the concept of continuous extension and the L_*^2 norm, we can define another norm for functions in $C_{0-}(0, T; C^\gamma(\bar{\Omega}))$ that will be used in later analysis. Specifically, for each $f \in C_{0-}(0, T; C^\gamma(\bar{\Omega}))$, we define

$$(3.42) \quad \|f\|_{\dagger} := \inf_{f_{\text{ext}}} (\|f_{\text{ext}}\|_{L_*^2}^2 + \|D^\gamma f_{\text{ext}}\|_{L_*^2}^2)^{1/2},$$

where the infimum is taken over all possible extensions f_{ext} of f in the space $C_{0-}(0, T; C_c^\gamma(\mathbb{R}^d))$. By Lemma 3.6, such extension always exists, so $\|f\|_{\dagger}$ is well-defined. And by Lemma 3.7, $\|f\|_{\dagger}$ is finite if $f \in C^2(0, T; C^\gamma(\bar{\Omega}))$.

3.5.3. Error estimates. Now let us estimate how small J^P can be minimized in the hypothesis space $\mathcal{S} = \{\hat{w} = \nu \hat{\phi}, \hat{\phi} \in \mathcal{F}_{2,M}\}$, and thereafter derive the final solution error. Before presenting the estimate, we discuss the existence of the boundary governing function $\nu(x)$. The following result shows we can always find some ν that is positive everywhere and vanishes on the boundary if the domain is smooth enough.

LEMMA 3.8. *Let Ω be a C^γ domain in \mathbb{R}^d with $\gamma \geq 2$. Then there exists some $\nu \in C_0^\gamma(\bar{\Omega})$ such that $\nu > 0$ in Ω .*

Proof. We give a constructive proof. Consider the elliptic equation

$$(3.43) \quad -\Delta_x \nu(x) = 1 \text{ in } \Omega, \quad \nu(x) = 0 \text{ on } \partial\Omega,$$

which admits a unique solution since Ω is C^γ smooth. By the global regularity theorem for strictly elliptic equations (e.g., Theorem 6.19, p. 111, in [13]), the solution ν is C^γ smooth in $\bar{\Omega}$. And by the maximum principle of strictly elliptic operators, ν only takes its minimum on $\partial\Omega$, so $\nu > 0$ in Ω . \square

For simplicity, we can normalize ν by multiplying some positive constant so that

$$(3.44) \quad \max\{\|\nu\|_\infty, \|\nabla_x \nu\|_\infty, \|\Delta_x \nu\|_\infty\} = 1.$$

Now we give the following error estimates.

THEOREM 3.9. *Let $\gamma > 2/d + 3$ be an integer. Let u be the classical solution of (3.1) with $\mathcal{L}_x = -\Delta_x$ and Ω being C^γ smooth. Let $\nu \in C_0^\gamma(\bar{\Omega})$ satisfy $\nu > 0$ in Ω and the normalization condition (3.44) (Lemma 3.8 implies the existence of ν). Moreover, let $m, m' \geq 0$ be two integers. Suppose $f \in \mathcal{W}^m(0, T; L^2(\Omega))$, $u \in C^{\max(2, m')}(0, T; C_0^\gamma(\bar{\Omega}))$, and $u/\nu < \infty$ on $\partial\Omega$ for each $t \in [0, T]$. If $\{\hat{w}_n\}_{n=1}^N$ solves the minimization (3.9) with hypothesis space $\mathcal{S} = \{\hat{w} = \nu \hat{\phi}, \hat{\phi} \in \mathcal{F}_{2,M}\}$, where the activation function σ of $\mathcal{F}_{2,M}$ satisfies the hypothesis of Lemma 3.5, then $\hat{u}_N(x, t) := \sum_{n=1}^N \hat{w}_n(x) \phi_n^{(1)}(t)$ satisfies the estimate*

$$\begin{aligned}
 (3.45) \quad & \sup_{0 \leq t \leq T} \|u(\cdot, t) - \hat{u}_N(\cdot, t)\|_{H^1_0(\Omega)} \\
 & + \|u - \hat{u}_N\|_{L^2(0,T;H^2(\Omega))} + \|\partial_t(u - \hat{u}_N)\|_{L^2(0,T;L^2(\Omega))} \\
 & \leq c(d^{\frac{1}{2}}N^2M^{-\frac{1}{2}}\|u/\nu\|_{\dagger} + d^{\frac{1}{2}}N^{-\frac{3}{2}-m'}\|\partial_t^{m'}(u/\nu)\|_{L^2_{m'}(0,T;H^2(\Omega))}) \\
 & + N^{-m}\|\partial_t^m f\|_{L^2_m(0,T;L^2(\Omega))}
 \end{aligned}$$

provided that $N \gg m, m'$, where c only depends on $\Omega, T, \gamma, \sigma, \lambda, m$, and m' .

Proof. We use the notation $A \lesssim B$ to mean $A \leq cB$ for some constant c only depending on $\Omega, T, \gamma, \sigma, \lambda, m$, and m' .

Denote $Q_T := \bar{\Omega} \times [0, T]$. Since $u(x, 0) = 0$ in $\bar{\Omega}$ and $\{\phi_n^{(1)}\}_{n=1}^\infty$ forms a Schauder basis for $C^2_{0-}([0, T])$, we can expand u in Q_T as $u = \sum_{n=1}^\infty \tilde{u}_n(x)\phi_n^{(1)}(t)$ with $\tilde{u}_n \in C^\gamma(\bar{\Omega})$. Denote $\tilde{u}_n^*(x) := \tilde{u}_n(x)/\nu(x)$; then the division $u/\nu = \sum_{n=1}^\infty \tilde{u}_n^*(x)\phi_n^{(1)}(t)$ is well-defined and continuous in Q_T since u/ν is finite on $\partial\Omega$ for each t and $\nu \neq 0$ in Ω . By the hypothesis $\nu \in C^\gamma_0(\bar{\Omega})$, we have $\tilde{u}_n^* \in C^\gamma(\bar{\Omega})$ and hence $u/\nu \in C^2_{0-}(0, T; C^\gamma(\bar{\Omega}))$.

Now from (3.42), let $u^*_{\text{ext}} \in C^2_{0-}(0, T; C^\gamma(\mathbb{R}^d))$ be some extension of u/ν such that

$$(3.46) \quad \|u/\nu\|_{\dagger} = \left(\|u^*_{\text{ext}}\|_{L^2_*}^2 + \|D^\gamma u^*_{\text{ext}}\|_{L^2_*}^2 \right)^{1/2},$$

which is finite by Lemma 3.7. We expand $u^*_{\text{ext}} = \sum_{n=1}^\infty \tilde{u}_{n,\text{ext}}^*(x)\phi_n^{(1)}(t)$ with $\tilde{u}_{n,\text{ext}}^* \in C^\gamma(\mathbb{R}^d)$. It is clear that $\tilde{u}_{n,\text{ext}}^*$ is an extension of \tilde{u}_n^* from $\bar{\Omega}$ to \mathbb{R}^d .

Next by Lemmas 3.4 and 3.5, there exists $\hat{\psi}_n \in \mathcal{F}_{2,M}$ such that

$$\begin{aligned}
 (3.47) \quad & \|\hat{\psi}_n - \tilde{u}_n^*\|_{H^2(\Omega)}^2 = \|\hat{\psi}_n - \tilde{u}_{n,\text{ext}}^*\|_{H^2(\Omega)}^2 \\
 & \lesssim M^{-1}\|\tilde{u}_{n,\text{ext}}^*\|_{\mathcal{B}^3}^2 \lesssim M^{-1} \int_{\mathbb{R}^d} |\tilde{u}_{n,\text{ext}}^*|^2 + |D^\gamma \tilde{u}_{n,\text{ext}}^*|^2 dx
 \end{aligned}$$

for $n = 1, \dots, N$. Then it follows (3.47) that

$$\begin{aligned}
 (3.48) \quad & \sum_{n=1}^N \|\hat{\psi}_n - \tilde{u}_n^*\|_{H^2(\Omega)}^2 \lesssim M^{-1} \sum_{n=1}^\infty \int_{\mathbb{R}^d} |\tilde{u}_{n,\text{ext}}^*|^2 + |D^\gamma \tilde{u}_{n,\text{ext}}^*|^2 dx \\
 & \lesssim M^{-1} \sum_{n=1}^\infty \int_{\mathbb{R}^d} |\tilde{u}_{n,\text{ext}}^* + \tilde{u}_{n+1,\text{ext}}^*|^2 + |D^\gamma \tilde{u}_{n,\text{ext}}^* + D^\gamma \tilde{u}_{n+1,\text{ext}}^*|^2 dx,
 \end{aligned}$$

where the last inequality follows the fact $\sum_{n=1}^\infty |r_n + r_{n+1}|^2 - \sum_{n=1}^\infty |r_n|^2 = |\sum_{n=1}^\infty r_n|^2 \geq 0$ for all real r_n .

On the other hand, using (2.6) we have

$$(3.49) \quad u^*_{\text{ext}} = \tilde{u}_{1,\text{ext}}^*(x)\tilde{L}_0(t) + \sum_{n=1}^\infty (\tilde{u}_{n,\text{ext}}^*(x) + \tilde{u}_{n+1,\text{ext}}^*(x))\tilde{L}_n(t),$$

so it follows (3.48) that

$$(3.50) \quad \sum_{n=1}^N \|\hat{\psi}_n - \tilde{u}_n^*\|_{H^2(\Omega)}^2 \lesssim M^{-1} \left(\|u^*_{\text{ext}}\|_{L^2_*}^2 + \|D^\gamma u^*_{\text{ext}}\|_{L^2_*}^2 \right) = M^{-1}\|u/\nu\|_{\dagger}^2.$$

Since $\nu\hat{\psi}_n \in \mathcal{S}$, substituting $f = u_t - \Delta_x u = \sum_{n=1}^\infty \nu\tilde{u}_n^*(x)\partial_t\phi_n^{(1)}(t) - \Delta_x(\nu\tilde{u}_n^*(x))\phi_n^{(1)}(t)$ in J^P and using Proposition 2.1, we obtain

(3.51)

$$\begin{aligned} J^P[\hat{w}_1, \dots, \hat{w}_N] &\leq J^P[\nu\hat{\psi}_1, \dots, \nu\hat{\psi}_N] \\ &= \sum_{j=1}^N r_j \left\| \sum_{n=1}^N a_{jn}^{(1)}(\nu\hat{\psi}_n - \nu\tilde{u}_n^*) - \sum_{n=1}^N b_{jn}^{(1)}(\Delta_x(\nu\hat{\psi}_n) - \Delta_x(\nu\tilde{u}_n^*)) \right\|_{L^2(\Omega)}^2 \\ &\quad + \lambda N^{-4} \|\Delta_x(\nu\hat{\psi}_N)\|_{L^2(\Omega)}^2 \\ &\lesssim N \sum_{n=1}^N \|\nu\hat{\psi}_n - \nu\tilde{u}_n^*\|_{L^2(\Omega)}^2 + N \|\Delta_x(\nu\hat{\psi}_1) - \Delta_x(\nu\tilde{u}_1^*)\|_{L^2(\Omega)}^2 + \sum_{n=1}^N \|\Delta_x(\nu\hat{\psi}_n) \\ &\quad - \Delta_x(\nu\tilde{u}_n^*)\|_{L^2(\Omega)}^2 + N^{-4} \left(\|\Delta_x(\nu\hat{\psi}_N) - \Delta_x(\nu\tilde{u}_N^*)\|_{L^2(\Omega)} + \|\Delta_x(\nu\tilde{u}_N^*)\|_{L^2(\Omega)} \right)^2 \\ &\lesssim N \sum_{n=1}^N \|\nu\hat{\psi}_n - \nu\tilde{u}_n^*\|_{L^2(\Omega)}^2 + N \|\Delta_x(\nu\hat{\psi}_1) - \Delta_x(\nu\tilde{u}_1^*)\|_{L^2(\Omega)}^2 \\ &\quad + \sum_{n=1}^N \|\Delta_x(\nu\hat{\psi}_n) - \Delta_x(\nu\tilde{u}_n^*)\|_{L^2(\Omega)}^2 + N^{-4} \|\Delta_x(\nu\tilde{u}_N^*)\|_{L^2(\Omega)}^2. \end{aligned}$$

On one hand, (3.44) implies that $\|\nu\hat{\psi}_n - \nu\tilde{u}_n^*\|_{L^2(\Omega)}^2 \leq \|\hat{\psi}_n - \tilde{u}_n^*\|_{L^2(\Omega)}^2$ and $\|\Delta_x(\nu\hat{\psi}_n) - \Delta_x(\nu\tilde{u}_n^*)\|_{L^2(\Omega)}^2 \leq d\|\hat{\psi}_n - \tilde{u}_n^*\|_{H^2(\Omega)}^2$, and then using (3.47) we have

(3.52)

$$\begin{aligned} N \sum_{n=1}^N \|\nu\hat{\psi}_n - \nu\tilde{u}_n^*\|_{L^2(\Omega)}^2 + N \|\Delta_x(\nu\hat{\psi}_1) - \Delta_x(\nu\tilde{u}_1^*)\|_{L^2(\Omega)}^2 \\ + \sum_{n=1}^N \|\Delta_x(\nu\hat{\psi}_n) - \Delta_x(\nu\tilde{u}_n^*)\|_{L^2(\Omega)}^2 \lesssim dN \sum_{n=1}^N \|\hat{\psi}_n - \tilde{u}_n^*\|_{H^2(\Omega)}^2 \lesssim dNM^{-1} \|u/\nu\|_{\dagger}^2. \end{aligned}$$

On the other hand, we note that for $k = 0, 1, 2$,

(3.53)

$$\partial_x^k \left(\frac{u}{\nu} \right) = \sum_{n=1}^\infty \partial_x^k \tilde{u}_n^*(x) \phi_n^{(1)}(t) = \partial_x^k \tilde{u}_1^*(x) \tilde{L}_0(t) + \sum_{n=1}^\infty (\partial_x^k \tilde{u}_n^*(x) + \partial_x^k \tilde{u}_{n+1}^*(x)) \tilde{L}_n(t).$$

So for every $x \in \Omega$,

$$\begin{aligned} (3.54) \quad &\|\pi_{[0,T]}^{N-1} \partial_x^k \left(\frac{u(x, \cdot)}{\nu(x)} \right) - \partial_x^k \left(\frac{u(x, \cdot)}{\nu(x)} \right)\|_{L^2([0,T])}^2 \\ &= \int_0^T \sum_{n=N}^\infty |\partial_x^k \tilde{u}_n^*(x) + \partial_x^k \tilde{u}_{n+1}^*(x)|^2 \cdot |\tilde{L}_n(t)|^2 dt \\ &= \sum_{n=N}^\infty \frac{T}{2n+1} |\partial_x^k \tilde{u}_n^*(x) + \partial_x^k \tilde{u}_{n+1}^*(x)|^2 \geq \frac{T}{2N+1} |\partial_x^k \tilde{u}_N^*(x)|^2. \end{aligned}$$

Since $u \in C^{m'}(0, T; C_0^\infty(\bar{\Omega})) \subset \mathcal{W}^{m'}(0, T; H^2(\bar{\Omega}))$, by Corollary 3.2, if $N \gg m'$,

$$(3.55) \quad \begin{aligned} |\partial_x^k \tilde{u}_N^*(x)|^2 &\lesssim N \|\pi_{[0,T]}^{N-1} \partial_x^k \left(\frac{u(x, \cdot)}{\nu(x)}\right) - \partial_x^k \left(\frac{u(x, \cdot)}{\nu(x)}\right)\|_{L^2([0,T])}^2 \\ &\lesssim N^{1-2m'} \|\partial_t^{m'} \partial_x^k \left(\frac{u(x, \cdot)}{\nu(x)}\right)\|_{L_{m'}^2([0,T])}^2, \end{aligned}$$

and therefore using (3.44) we have

$$(3.56) \quad \begin{aligned} \|\Delta_x(\nu \tilde{u}_N^*)\|_{L^2(\Omega)}^2 &\leq \|\tilde{u}_N^*\|_{L^2(\Omega)}^2 + \|\nabla_x \tilde{u}_N^*\|_{L^2(\Omega)}^2 + \|\Delta_x \tilde{u}_N^*\|_{L^2(\Omega)}^2 \\ &\lesssim \left(\|\partial_t^{m'}(u/\nu)\|_{L_{m'}^2(0,T;L^2(\Omega))}^2 + \|\nabla_x \partial_t^{m'}(u/\nu)\|_{L_{m'}^2(0,T;L^2(\Omega))}^2 \right. \\ &\quad \left. + d \sum_{i=1}^d \|\partial_{x_i}^2 \partial_t^{m'}(u/\nu)\|_{L_{m'}^2(0,T;L^2(\Omega))}^2 \right) N^{1-2m'} \\ &\lesssim dN^{1-2m'} \|\partial_t^{m'}(u/\nu)\|_{L_{m'}^2(0,T;H^2(\Omega))}^2. \end{aligned}$$

Finally, combining (3.51), (3.52), (3.56) and using Theorem 3.3 completes the proof. \square

Theorem 3.1 implies that the solution error approaches to zero if $M^{1/2}N^{-2} \rightarrow \infty$ and $N \rightarrow \infty$ given that $m \geq 1$ and $m' \geq 0$. It requires that $M = O(N^p)$ with $p > 4$. So in theory the width of the neural network should be much larger than the number of bases in the Galerkin formulation. However, in practical examples (e.g., in section 5), the width getting small errors is much less than the theoretical requirement.

3.5.4. Long time behavior estimates. Let us further investigate how the solution error behaves as T varies. We do rescale the time variable by taking $t' = t/T$ in (3.1), and then the problem becomes

$$(3.57) \quad \begin{cases} \frac{1}{T} \frac{\partial u(x, Tt')}{\partial t'} + \mathcal{L}_x u(x, Tt') = f(x, Tt') & \text{in } \Omega \times (0, 1], \\ u(x, 0) = 0 \text{ in } \Omega, \quad u(x, t') = 0 \text{ on } \partial\Omega \times [0, 1]. \end{cases}$$

Hence it suffices to consider the problem (3.57) in standard time interval with t multiplied by T in the equation. For wave-like solutions oscillatory in time, it is equivalent to increase their frequency by T times. Therefore, we specify the solution as the sine Fourier mode $u = \nu \sin(Tt)$ and investigate how the frequency T influences the error bound in (3.45) for the standard heat equation in $\Omega \times [0, 1]$.

For the first term, by the estimate (3.38) we have $\|u/\nu\|_{\dagger} = \|\sin(Tt)\|_{\dagger} \sim O(T^{7/2})$. For the second term, since $|\partial_t^{m'}(u/\nu)| = T^{m'} |\cos(Tt)|$ if m' is odd and $|\partial_t^{m'}(u/\nu)| = T^{m'} |\sin(Tt)|$ if m' is even, we have $\|\partial_t^{m'}(u/\nu)\|_{L_{m'}^2(0,T;H^2(\Omega))} \sim O(T^{m'})$. Similarly, for the third term, noting $f = T\nu \cdot \cos(Tt) - \Delta_x \nu \cdot \sin(Tt)$, we have $\|\partial_t^m f\|_{L_m^2(0,T;L^2(\Omega))^2} \sim O(T^{m+1})$. Combining the preceding results shows that the error bound in (3.45) is dominated by $O(d^{\frac{1}{2}}N^2M^{-\frac{1}{2}}T^{\frac{7}{2}} + d^{\frac{1}{2}}N^{-\frac{3}{2}}(\frac{T}{N})^{m'} + T(\frac{T}{N})^m)$.

Consequently, to make the error bound small, it is necessary to set $\frac{T}{N} < 1$; namely, the number of temporal basis functions should exceed the frequency of the solution. This conclusion is trivial if we note the fact that if the solution has high-frequency modes in t , the trial space $X_N^{(1)}$ should be large enough to contain them. Moreover, in the case u is C^∞ smooth in t , the regularity orders m' and m are infinitely large, and hence the error bound grows exponentially with T if M and N are fixed.

4. Hyperbolic equations. Similar to the DABG method for parabolic equations introduced in section 3, we also develop the method for the following type of hyperbolic equations:

$$(4.1) \quad \begin{cases} \frac{\partial^2 u(x,t)}{\partial t^2} + \mathcal{L}_x u(x,t) = f(x,t) & \text{in } \Omega \times (0, T], \\ u(x,0) = 0, \partial_t u(x,0) = g_0(x) & \text{in } \Omega, \quad u(x,t) = 0 \text{ on } \partial\Omega \times [0, T]. \end{cases}$$

Since all discussions inherit from section 3 directly, we briefly present the definitions and method without many explanations. Also, we exhibit the error estimates that can be proved by the same argument as in Theorems 3.3 and 3.9.

Recalling $X^{(2)}, Y^{(2)}, Y_N^{(2)}$ as defined in section 2.2.2, we define the trial space

$$(4.2) \quad \mathcal{U}^H := \left\{ u(x,t) : u(x,\cdot) \in X^{(2)} \ \forall x \in \Omega, \ u|_{\partial\Omega \times [0,T]} = 0 \right\},$$

where the label ‘‘H’’ is short for the hyperbolic equation. Following the method for the second-order problem in section 2.2.2, we derive the variational form: find $u \in \mathcal{U}^H$ such that

$$(4.3) \quad (\partial_t u(x,\cdot), \partial_t v) - \alpha (\mathcal{L}_x u(x,\cdot), v) = - (f(x,\cdot), v) - g_0(x)v(0) \ \forall v \in Y^{(2)}, \ \forall x \in \Omega.$$

Similarly, define the approximate trial space

$$(4.4) \quad \mathcal{U}_N^H := \left\{ \hat{u}_N(x,t) = \sum_{n=1}^{N-1} \hat{w}_n(x)\phi_n^{(2)}(t), \ \hat{w}_n \in \mathcal{S} \right\}.$$

Then the Galerkin formulation for (4.1) is finding $\hat{u}_N \in \mathcal{U}_N^H$ such that

$$(4.5) \quad \begin{aligned} (\partial_t \hat{u}_N(x,\cdot), \partial_t v_N) - \alpha (\mathcal{L}_x \hat{u}_N(x,\cdot), v_N) \\ = - (f(x,\cdot), v_N) - g_0(x)v_N(0) \ \forall v_N \in Y_N^{(2)}, \ \forall x \in \Omega, \end{aligned}$$

where the integration by parts is used. Taking $\hat{u}_N = \sum_{n=1}^N \hat{w}_n \phi_n^{(2)}$ and $v_N = \psi_j^{(2)}$ for $j = 1, \dots, N$, we obtain

$$(4.6) \quad \sum_{n=1}^N \left(a_{jn}^{(2)} \hat{w}_n(x) - b_{jn}^{(2)} \mathcal{L}_x \hat{w}_n(x) \right) = - \left(f(x,\cdot), \psi_j^{(2)} \right) - g_0(x)\psi_j^{(2)}(0), \quad j = 1, \dots, N,$$

with unknowns $\{\hat{w}_n\} \subset \mathcal{S}$, where $a_{jn}^{(2)}$ and $b_{jn}^{(2)}$ are defined in (2.16). Practically, we solve (4.6) by the following least-squares optimization:

$$(4.7) \quad \min_{\hat{w}_n \in \mathcal{S}} J^H[\hat{w}_1, \dots, \hat{w}_N] := \sum_{j=1}^N \left\| R_j^{(2)}[\hat{w}_1, \dots, \hat{w}_N] \right\|_{L^2(\Omega)}^2 + \lambda N^{-5} I[\hat{w}_N],$$

where $R_j^{(2)}[\hat{w}_1, \dots, \hat{w}_N] := \sum_{n=1}^N (a_{jn}^{(2)} \hat{w}_n(x) - b_{jn}^{(2)} \mathcal{L}_x \hat{w}_n(x)) + (f(x,\cdot), \psi_j^{(2)}) + g_0(x)\psi_j^{(2)}(0)$ is the residual of the j th equation in (4.6), and

$$(4.8) \quad \begin{aligned} I[\hat{w}_N] &:= \left(\hat{u}_N''(x,\cdot) + \mathcal{L}_x \hat{u}_N(x,\cdot) - f(x,\cdot), \tilde{L}_N \right) \\ &= \frac{T(N-1)}{N(2N+1)} \mathcal{L}_x \hat{w}_N(x) - \int_0^T f(x,t) L_N \left(\frac{2t}{T} - 1 \right) dt \end{aligned}$$

is the inner product of the differential equation residual and the orthogonal polynomial $\tilde{L}_N(t) := L_N(\frac{2t}{T} - 1)$ in $L^2([0, T])$. The term $\lambda N^{-5}I[\hat{w}_N]$ can be viewed as a regularization of the last coefficient \hat{w}_N , which is specifically constructed according to the a posteriori estimate. By Proposition 2.2, $R_j^{(2)}[\hat{w}_1, \dots, \hat{w}_N]$ only includes at most seven nonzero terms if $j \geq 2$. Similar to the parabolic case, the regularization term $\lambda N^{-5}I[\hat{w}_N]$ are specially chosen such that the a posteriori estimate holds. More precisely, we have the following a posteriori estimate for the wave equation.

THEOREM 4.1. *Let u be the classical solution of (4.1) with $\mathcal{L}_x = -\Delta_x$. Suppose that $f \in \mathcal{W}^m(0, T; L^2(\Omega))$, $u \in L^2(0, T; H_0^1(\Omega))$, $\partial_t u \in L^2(0, T; L^2(\Omega))$, and $\partial_{tt} u \in L^2(0, T; H^{-1}(\Omega))$. Let $\hat{u}_N(x, t) = \sum_{n=1}^N \hat{w}_n(x) \phi_n^{(2)}(t)$, where $\{\hat{w}_n\}_{n=1}^N$ solves the minimization (4.7). Suppose that $\hat{w}_n \in H_0^1(\Omega)$. Denote $\delta = J^H[\hat{w}_1, \dots, \hat{w}_N]$ as the minimized loss. Then*

$$(4.9) \quad \sup_{0 \leq t \leq T} \left(\|u(\cdot, t) - \hat{u}_N(\cdot, t)\|_{H_0^1(\Omega)} + \|\partial_t u(\cdot, t) - \partial_t \hat{u}_N(\cdot, t)\|_{L^2(\Omega)} \right) \\ \leq c \left(N^3 \sqrt{\delta} + N^{-m} \|\partial_t^m f\|_{L_m(0, T; L^2(\Omega))} \right),$$

where c is a constant only depending on Ω , m , T , and λ .

Remark 4.1. In Theorem 4.1, the error bound has a quantity $O(N^3)$ appended to the minimal loss $\sqrt{\delta}$, which is larger than the parabolic case. It is due to the usage of a three-term combination in (2.14) to construct $\psi_n^{(2)}$, and the transformation matrix has a larger 2-condition number $O(N^3)$.

Similar to Theorem 3.9, we also have the error estimate.

THEOREM 4.2. *Let $\gamma > 2/d + 3$ be an integer. Let u be the classical solution of (4.1) with $\mathcal{L}_x = -\Delta_x$ and Ω being C^γ smooth. Let $\nu \in C_0^\gamma(\bar{\Omega})$ satisfy $\nu > 0$ in Ω and the normalization condition (3.44). (Lemma 3.8 implies the existence of ν .) Moreover, let $m \geq 0$ be an integer. Suppose $f \in \mathcal{W}^m(0, T; L^2(\Omega))$, $u \in C^2(0, T; C_0^\gamma(\bar{\Omega}))$, and $u/\nu < \infty$ on $\partial\Omega$ for each $t \in [0, T]$. If $\{\hat{w}_n\}_{n=1}^N$ solves the minimization (4.7) with hypothesis space $\mathcal{S} = \{\hat{w} = \nu \hat{\phi}, \hat{\phi} \in \mathcal{F}_{2, M}\}$, where the activation function σ of $\mathcal{F}_{2, M}$ satisfies the hypothesis of Lemma 3.5, then $\hat{u}_N(x, t) := \sum_{n=1}^N \hat{w}_n(x) \phi_n^{(2)}(t)$ satisfies the estimate*

$$(4.10) \quad \sup_{0 \leq t \leq T} \left(\|u(\cdot, t) - \hat{u}_N(\cdot, t)\|_{H_0^1(\Omega)} + \|\partial_t u(\cdot, t) - \partial_t \hat{u}_N(\cdot, t)\|_{L^2(\Omega)} \right) \\ \leq c \left(d^{\frac{1}{2}} N^{\frac{7}{2}} M^{-\frac{1}{2}} \|u/\nu\|_{\dagger} + N^{-m} \|\partial_t^m f\|_{L_m(0, T; L^2(\Omega))^2} \right)$$

provided that $N \gg m$, where c only depends on Ω , T , γ , σ , λ , and m .

As mentioned in section 3.5, the error bound given in Theorem 4.2 is merely derived from a theoretical perspective. The width M used in practice leading to convergence could be much smaller than the theoretical quantity N^p with $p > 7$. A hyperbolic example is presented in section 5.2.5 to demonstrate it.

5. Numerical examples.

5.1. Overview. The proposed DABG method is tested by several examples including parabolic and hyperbolic equations. Meanwhile, the popular DLS method (1.3) is implemented for comparison. To be consistent with the hypothesis of the pre-

ceding theory, we set homogeneous initial and boundary conditions for all examples. Therefore, we set the hypothesis space \mathcal{N}' of the DLS method as

$$(5.1) \quad \mathcal{N}' = \left\{ \hat{u}(x, t) = t^\beta \nu(x) \hat{\phi}(x, t), \hat{\phi} \in \mathcal{F}_{L, M} : \mathbb{R}^{d+1} \rightarrow \mathbb{R} \right\}$$

with $\beta = 1$ for parabolic equations and $\beta = 2$ for hyperbolic equations, where the variables x and t are concatenated as one input variable of the FNN $\hat{\phi}$. In (5.1), ν is the same boundary governing function as DABG that is chosen corresponding to the domain shape. In our experiments, we choose $\nu(x) = (x_1^2 - 1)(x_2^2 - 1)$ for Cases 1–2, where Ω is set as the square $[-1, 1]^2$, and choose $\nu(x) = |x|^2 - 1$ for Cases 3–5, where Ω is set as the unit sphere. It is clear that $\hat{u}|_{\partial\Omega \times [0, T]} = 0$ and $\partial_t^{\beta-1} \hat{u}|_{\Omega \times \{t=0\}} = 0$ for all $\hat{u} \in \mathcal{N}'$, so the functions in \mathcal{N}' always satisfy the homogeneous initial and boundary conditions and hence \mathcal{N}' can be taken as the hypothesis space of the DLS formulation (1.3). Other settings are summarized as follows:

- *Environment.* The methods are implemented mainly in a Python environment. The PyTorch library with CUDA toolkit is utilized for neural network implementation and GPU-based parallel computing.
- *Optimizer and hyper-parameters.* The network-based optimizations problems (3.9) and (4.7) are solved by the batch gradient descent as well as the compared DLS method. For both methods, the gradient descent is implemented for totally 20000 iterations with the same decaying learning rates.
- *Network setting.* For DABG, we fix $L = 3$ in the class S and test the method for various width M and basis numbers N . For DLS, we implement it with various depth L and width M . For both methods, we choose the sigmoidal function as the activation function of the involved FNNs. And their parameters are initialized by

$$(5.2) \quad a, W_l, b_l \sim U(-\sqrt{M}, \sqrt{M}), \quad l = 1, \dots, L.$$

- *Numerical integration.* For the evaluation of L^2 norm over Ω , we adopt the quasi-Monte Carlo method with Halton sequence. In every iteration, the integral is approximately evaluated as the mean of the integrand at 10^4 uniform random points.
- *Testing set and error evaluation.* We generate a spatial testing set \mathcal{X} consisting of $N_{\mathcal{X}}$ uniform random points in Ω . Also, denote $\mathcal{T} = \{kT/N_{\mathcal{T}}\}_{k=1}^{N_{\mathcal{T}}}$ as the set of $N_{\mathcal{T}}$ equidistant grid points in $[T/N_{\mathcal{T}}, T]$. We compute the following relative ℓ^2 error over $\mathcal{X} \times \mathcal{T}$:

$$(5.3) \quad e_{\ell^2(\mathcal{X} \times \mathcal{T})} := \left(\frac{\sum_{t \in \mathcal{T}} \sum_{x \in \mathcal{X}} |\hat{u}(x, t) - u(x, t)|^2}{\sum_{t \in \mathcal{T}} \sum_{x \in \mathcal{X}} |u(x, t)|^2} \right)^{\frac{1}{2}},$$

where \hat{u} and u are the numerical solution and true solution, respectively.

5.2. Examples.

5.2.1. Case 1: 2-D heat equation oscillatory in time. We consider the heat equation (3.1) with $\mathcal{L}_x = -\Delta_x$. In this case, the domain is set to be the 2-D square $\Omega = [-1, 1]^2$, and the true solution is set to be

$$u_1(x, t) = \exp(\sin(2\pi wt)(x_1^2 - 1)(x_2^2 - 1)) - 1,$$

where w is the wave number in the time variable.

TABLE 5.1
Relative ℓ^2 errors of DABG and DLS methods in Case 1 with $w = 1$.

(a) DABG

N	$M = 10$	$M = 20$	$M = 40$	$M = 80$
4	7.145e-02	7.145e-02	7.145e-02	7.144e-02
6	3.604e-03	3.603e-03	3.597e-03	3.685e-03
8	5.015e-04	5.016e-04	5.000e-04	4.999e-04
10	4.531e-05	4.545e-05	4.539e-05	1.314e-04
12	1.749e-06	1.712e-06	1.804e-06	2.229e-06
14	1.813e-07	1.157e-07	5.795e-07	4.839e-07
16	1.517e-07	8.218e-08	1.840e-06	1.690e-07
18	1.607e-07	8.531e-08	3.663e-07	5.854e-07

(b) DLS

L	$M = 10$	$M = 20$	$M = 40$	$M = 80$
3	5.907e-03	1.266e-03	1.911e-03	5.256e-03
4	2.898e-03	4.643e-04	1.246e-03	1.281e-03
5	2.571e-03	9.885e-01	4.881e-04	9.885e-01

TABLE 5.2
Relative ℓ^2 errors of DABG and DLS methods in Case 1 with $w = 8$.

(a) DABG

N	$M = 10$	$M = 20$	$M = 40$	$M = 80$
40	1.124e-01	1.124e-01	1.113e-01	1.076e-01
50	7.045e-03	7.045e-03	6.932e-03	2.397e-02
60	4.783e-04	4.791e-04	4.787e-04	1.628e-02
70	4.328e-04	4.321e-04	4.322e-04	1.526e-02

(b) DLS

L	$M = 10$	$M = 20$	$M = 40$	$M = 80$
3	9.607e-01	8.574e-01	4.019e-01	9.898e-01
4	8.675e-01	5.152e-01	5.973e-01	5.786e-01
5	9.960e-01	9.960e-01	9.960e-01	9.960e-01

First, we solve the equation with $T = 1$ and $w = 1, 8$. For $w = 1$, u_1 has only one wave in time and is hence less oscillatory. It is observed in Table 5.1 that both DABG and DLS can find solutions with small errors, but DABG with errors $O(10^{-8})$ is clearly more accurate than DLS with errors $O(10^{-4})$. For $w = 8$, the solution is highly oscillatory in time. It is observed in Table 5.2 that DABG can still obtain errors $O(10^{-4})$, yet DLS obtains larger errors $O(10^{-1})$. For DABG, the error curves versus N are presented in Figure 5.1, and the exponential decaying rate is observed for small N . When N is relatively large, the error decaying of DABG is limited due to the optimization error of deep learning. Besides, it is notable that when $w = 8$ the largest width choice $M = 80$ leads to the worst result, probably because solving the optimization with wider networks is more difficult. We also show the temporal profiles of the true and numerical solutions at $(x_1, x_2) = (0, 0)$ in Figure 5.2. When $w = 8$, DABG can capture all eight waves, while DLS can only recover six waves. This implies DABG is more capable of finding solutions with high frequencies in time.

Next, we investigate the error behavior versus time by solving the problem with $T = 2, 4, 6, 8, 10$ for the fixed solution with $w = 1$. We implement the method with $M = 20$ and $N = 20, 40, 60$. The errors are listed in Table 5.4 and the error curves

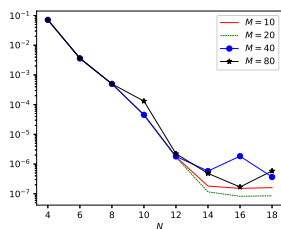
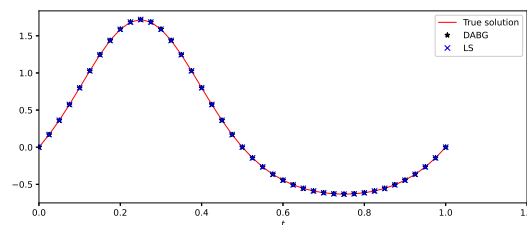
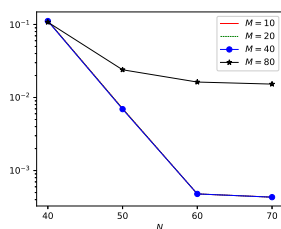
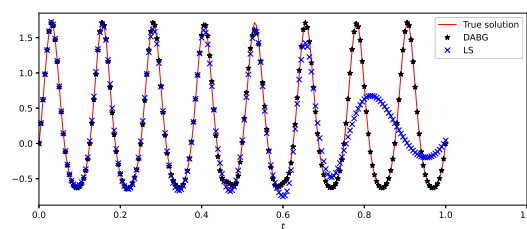
(a) $w = 1$ (a) $w = 1$ (b) $w = 8$ (b) $w = 8$ FIG. 5.1. Relative ℓ^2 errors versus N of DABG in Case 1.

FIG. 5.2. Temporal profiles of the true solution and numerical solutions obtained by DABG and DLS.

versus time are shown in Figure 5.4. We observe that for each fixed N , the error basically grows in exponential rate with T . Note the numerical result is consistent with the theory for C^∞ solutions discussed in section 3.5.4.

5.2.2. Case 2: 2-D heat equation oscillatory in time and space. We continue solving the preceding heat equation, and the true solution is now set as

$$u_2(x, t) = \exp(\sin(2\pi wt(x_1^2 - 1)(x_2^2 - 1))) - 1.$$

Note the wave number w is coupled with both t and x . We set $w = 3$, and hence u_2 is oscillatory in both time and space. It is shown in Table 5.3 that DABG with errors $O(10^{-3})$ still performs better than DLS with errors $O(10^{-1})$. However, compared with the preceding case, it shows DABG is less accurate for solutions oscillatory in both time and space than those only oscillatory in time.

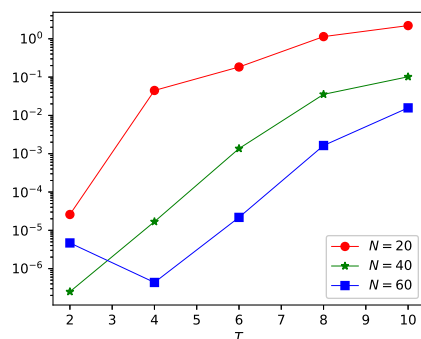
5.2.3. Case 3: 20-D parabolic equation. In this case, we test the capability of the methods in high dimensions. Let us consider the parabolic equation (3.1) with $\mathcal{L}_x u = -\nabla_x \cdot (a(x)\nabla_x u)$, where $a(x) = 1 + |x|^2/2$ is nonconstant. The domain is set to be the 20-D unit ball $\Omega = \{x \in \mathbb{R}^{20} : |x| < 1\}$, and the true solution is set to be

$$u_3(x, t) = \sin((\sin(2\pi wt)(|x|^2 - 1)).$$

We set $w = 3$, and obtained errors are shown in Table 5.4. It shows that DABG still obtains smaller errors ($O(10^{-3})$) than DLS ($O(10^{-2})$).

5.2.4. Case 4: 20-D Allen–Cahn equation. In this case, we solve the Allen–Cahn equation

$$(5.4) \quad \begin{cases} u_t - \Delta u + u^3 - u = f_{AC} & \text{in } \Omega \times (0, T], \\ u(x, 0) = 0 & \text{in } \Omega, \quad u(x, t) = 0 & \text{on } \partial\Omega \times [0, T]. \end{cases}$$

FIG. 5.3. Relative ℓ^2 errors versus T of DABG in Case 1.

N	$T = 2$	$T = 4$	$T = 6$	$T = 8$	$T = 10$
10	2.393e-02	7.501e-01	2.533e+00	3.727e+00	2.632e+00
20	2.590e-05	4.482e-02	1.836e-01	1.141e+00	2.213e+00
30	2.219e-06	1.394e-03	7.172e-02	1.179e-01	4.590e-01
40	2.497e-07	1.682e-05	1.362e-03	3.546e-02	1.020e-01
50	8.236e-06	1.379e-06	5.531e-04	3.683e-03	5.696e-02
60	4.682e-06	4.353e-07	2.176e-05	1.636e-03	1.574e-02

FIG. 5.4. Relative ℓ^2 errors of DABG methods in Case 1 for various T .

TABLE 5.3

Relative ℓ^2 errors of DABG and DLS methods in Case 2 with $w = 3$.

(a) DABG

N	$M = 20$	$M = 40$	$M = 80$
5	6.436e-01	5.612e-01	5.590e-01
10	5.184e-01	4.214e-02	3.742e-02
15	4.954e-01	2.647e-02	1.419e-02
20	5.902e-01	1.667e-02	3.095e-02
25	3.005e-01	1.577e-02	8.192e-03

(b) DLS

L	$M = 20$	$M = 40$	$M = 80$
3	8.103e-01	5.304e-01	7.790e-01
4	5.367e-01	2.288e+00	2.285e+00
5	2.291e+00	2.289e+00	2.285e+00

TABLE 5.4

Relative ℓ^2 errors of DABG and DLS methods in Case 3 with $w = 3$.

(a) DABG

N	$M = 40$	$M = 80$	$M = 160$
10	3.615e-01	3.615e-01	3.616e-01
15	3.077e-03	3.077e-03	3.078e-03
20	3.078e-03	3.078e-03	3.077e-03
25	3.075e-03	3.075e-03	3.074e-03

(b) DLS

L	$M = 40$	$M = 80$	$M = 160$
3	6.844e-01	9.935e-01	4.681e-02
4	4.542e-02	5.292e-01	9.935e-01
5	9.935e-01	9.935e-01	9.935e-01

The domain is set as $\Omega = \{x \in \mathbb{R}^{20} : |x| < 1\}$, and the true solution is set to be

$$(5.5) \quad u_4(x, t) = \sin(\sin(2\pi wt)(|x|^2 - 1)).$$

Due to the nonlinearity of the equation, we propose the following fixed point scheme,

$$(5.6) \quad u_t^{(n)} - \Delta u^{(n)} - u^{(n)} = f_{AC} - (u^{(n-1)})^3, \quad n = 1, 2, \dots,$$

where $u^{(0)}$ is an initial guess, and $u^{(n)}$ is the numerical solution obtained in the n th step. In practice, we use DABG to solve (3.52). In each SGD iteration, we update

TABLE 5.5
Relative ℓ^2 errors of DABG and DLS methods in Case 4 with $w = 3$.

(a) DABG

N	$M = 40$	$M = 80$	$M = 160$
10	3.080e-01	3.080e-01	3.080e-01
14	2.300e-03	2.300e-03	2.300e-03
18	2.069e-03	2.069e-03	2.069e-03
22	2.070e-03	2.070e-03	2.070e-03

TABLE 5.6
Relative ℓ^2 errors of DABG and DLS methods in Case 5 with $w = 3$.

(a) DABG

N	$M = 40$	$M = 80$	$M = 160$
10	3.055e-01	3.057e-01	3.056e-01
14	3.505e-03	3.505e-03	3.513e-03
18	1.907e-03	1.907e-03	1.913e-03
22	2.114e-03	2.114e-03	2.121e-03

(b) DLS

L	$M = 40$	$M = 80$	$M = 160$
3	5.312e-01	9.937e-01	9.394e-01
4	7.791e-02	1.329e-01	9.937e-01
5	9.937e-01	9.937e-01	9.937e-01

(b) DLS

L	$M = 40$	$M = 80$	$M = 160$
3	1.141e-01	1.142e+00	9.744e-01
4	8.465e-02	9.745e-01	9.744e-01
5	9.745e-01	9.745e-01	9.745e-01

the loss function according to (3.52); namely, in the n th iteration, we compute the loss function with $f = f_{AC} - (u^{(n-1)})^3$ and update the solution by gradient descent. Obtained errors for $w = 3$ are shown in Table 5.5. It shows that DABG (with errors $O(10^{-3})$) performs better than least square (with errors $O(10^{-2})$).

5.2.5. Case 5: 20-D hyperbolic equation. In this case, we solve the hyperbolic equation (4.1) with $\mathcal{L}_x u = -\nabla_x \cdot (a(x)\nabla_x u)$ and $g_0 = 0$, where $a(x) = 1 + |x|^2/2$ is nonconstant. The domain $\Omega = \{x \in \mathbb{R}^{20} : |x| < 1\}$ and the true solution is

$$(5.7) \quad u_5(x, t) = \sin(t \sin(2\pi wt)(|x|^2 - 1)).$$

We report the errors for $w = 3$ in Table 5.6. The same as in the previous parabolic cases, DABG obtains smaller errors than DLS in this hyperbolic example.

6. Conclusion. We develop a novel and highly accurate DABG method for solving high-dimensional evolution equations. In this method, the solution is approximated by the adaptive basis structure, which is formulated as a tensor-product of a set of orthogonal polynomials and a class of DNNs. The first step is the time discretization of the equation using traditional Galerkin formulation, leading to a linear system with unknown DNNs. The second step is determining these DNNs via the minimization of the residual of the linear system. This method enjoys not only the high accuracy of Galerkin formulation, but also the capability of high-dimensional approximation of DNNs. For problems with oscillatory solutions, the numerical results show a significant advantage of the proposed method over the popular DLS method.

Note that the technique for time discretization is not limited to the Galerkin method with polynomial-type basis used in this work. For some special time-dependent problems (e.g., porous medium equation and Navier–Stokes equations), it is promising to consider other techniques such as finite element approximation and energy stable numerical schemes, which can be further studied. Another future direction is to consider better deep solvers for the spatial equation after time discretization. In this work, we formulate the loss function following the idea of a minimal residual, which is straightforward but sometimes less efficient. The simultaneous training of all the unknown DNNs is also computationally expensive. Since the coefficient matrix has a sparse banded structure, faster learning relying on this property is expected to be developed.

REFERENCES

- [1] A. R. BARRON, *Neural net approximation*, in Proceedings of the 7th Yale Workshop on Adaptive and Learning Systems, 1992.
- [2] A. R. BARRON, *Universal approximation bounds for superpositions of a sigmoidal function*, IEEE Trans. Inform. Theory, 39 (1993), pp. 930–945.
- [3] C. BECK, S. BECKER, P. CHERIDITO, A. JENTZEN, AND A. NEUFELD, *Deep splitting method for parabolic PDEs*, SIAM J. Sci. Comput., 43 (2021), pp. A3135–A3154.
- [4] J. BERG AND K. NYSTRÖM, *A unified deep artificial neural network approach to partial differential equations in complex geometries*, Neurocomputing, 317 (2018), pp. 28–41.
- [5] A. CARAGEA, P. PETERSEN, AND F. VOIGTLAENDER, *Neural Network Approximation and Estimation of Classifiers with Classification Boundary in a Barron Class*, <https://arxiv.org/abs/2011.09363>, 2020.
- [6] W. E, J. HAN, AND A. JENTZEN, *Deep learning-based numerical methods for high-dimensional parabolic partial differential equations and backward stochastic differential equations*, Commun. Math. Stat., 5 (2017), pp. 349–380.
- [7] W. E, C. MA, S. WOJTOWYTSCH, AND L. WU, *Towards a Mathematical Understanding of Neural Network-Based Machine Learning: What We Know and What We Don't*, <https://arxiv.org/abs/2009.10713>, 2020.
- [8] W. E, C. MA, AND L. WU, *The Barron space and the flow-induced function spaces for neural network models*, Constr. Approx., 55 (2022), pp. 369–406.
- [9] W. E AND B. YU, *The deep Ritz method: A deep learning-based numerical algorithm for solving variational problems*, Commun. Math. Stat., 6 (2018).
- [10] A. ERN AND J.-L. GUERMOND, *Theory and Practice of Finite Elements*, Springer, New York, 2004.
- [11] L. C. EVANS, *Partial Differential Equations*, 2nd ed., American Mathematical Society, Providence, RI, 2010.
- [12] W. GAUTSCHI, *Orthogonal Polynomials: Computation and Approximation*, Oxford University Press, Oxford, 2004.
- [13] D. GILBARG AND N. S. TRUDINGER, *Elliptic Partial Differential Equations of Second Order*, Springer, New York, 2015.
- [14] Y. GU, C. WANG, AND H. YANG, *Structure probing neural network deflation*, J. Comput. Phys., 434 (2021), 110231.
- [15] J. HAN, A. JENTZEN, AND W. E, *Solving high-dimensional partial differential equations using deep learning*, Proc. Natl. Acad. Sci. USA, 115 (2018), pp. 8505–8510.
- [16] Y. JIAO, Y. LAI, X. LU, F. WANG, J. Z. YANG, AND Y. YANG, *Deep Neural Networks with ReLU-Sine-Exponential Activations Break Curse of Dimensionality on Hölder Class*, <https://arxiv.org/abs/2103.00542>, 2021.
- [17] Y. KHOO, J. LU, AND L. YING, *Solving parametric PDE problems with artificial neural networks*, European J. Appl. Math., 32 (2021), pp. 421–435.
- [18] J. M. KLUSOWSKI AND A. R. BARRON, *Approximation by combinations of ReLU and squared ReLU ridge functions with ℓ^1 and ℓ^0 controls*, IEEE Trans. Inform. Theory, 64 (2018), pp. 7649–7656.
- [19] K. LI, K. TANG, T. WU, AND Q. LIAO, *D3M: A deep domain decomposition method for partial differential equations*, IEEE Access, 8 (2019), pp. 5283–5294.
- [20] M. RAISSI, P. PERDIKARIS, AND G. E. KARNIADAKIS, *Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations*, J. Comput. Phys., 378 (2019), pp. 686–707.
- [21] J. SHEN, T. TANG, AND L.-L. WANG, *Spectral Methods: Algorithms, Analysis and Applications*, Springer, New York, 2011.
- [22] Z. SHEN, H. YANG, AND S. ZHANG, *Deep Network Approximation: Achieving Arbitrary Accuracy with Fixed Number of Neurons*, <https://arxiv.org/abs/2107.02397>, 2021.
- [23] Z. SHEN, H. YANG, AND S. ZHANG, *Deep network with approximation error being reciprocal of width to power of square root of depth*, Neural Comput., 33 (2021), pp. 1005–1036.
- [24] Z. SHEN, H. YANG, AND S. ZHANG, *Neural network approximation: Three hidden layers are enough*, Neural Netw., 141 (2021), pp. 160–173.
- [25] J. W. SIEGEL AND J. XU, *Approximation rates for neural networks with general activation functions*, Neural Netw., 128 (2020), pp. 313–321.
- [26] J. W. SIEGEL AND J. XU, *High-Order Approximation Rates for Neural Networks with ReLU^k Activation Functions*, <https://arxiv.org/abs/2012.07205>, 2020.
- [27] J. SIRIGNANO AND K. SPILIOPOULOS, *DGM: A deep learning algorithm for solving partial differential equations*, J. Comput. Phys., 375 (2018), pp. 1339–1364.

- [28] Z.-Q. J. X. W. CAI, *Multi-Scale Deep Neural Networks for Solving High Dimensional PDEs*, <https://arxiv.org/abs/1910.11710>, 2019.
- [29] Y. ZANG, G. BAO, X. YE, AND H. ZHOU, *Weak adversarial networks for high-dimensional partial differential equations*, *J. Comput. Phys.*, 411 (2020), 109409.