

Neural network enhanced computations on coarse grids

Jan Nordström and Oskar Ålund

The self-archived postprint version of this journal article is available at Linköping University Institutional Repository (DiVA):

<http://urn.kb.se/resolve?urn=urn:nbn:se:liu:diva-170823>

N.B.: When citing this work, cite the original publication.

Nordström, J., Ålund, O., (2021), Neural network enhanced computations on coarse grids, *Journal of Computational Physics*, 425, 109821. <https://doi.org/10.1016/j.jcp.2020.109821>

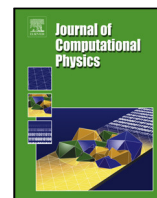
Original publication available at:

<https://doi.org/10.1016/j.jcp.2020.109821>

Copyright: Elsevier

<http://www.elsevier.com/>





Neural Network Enhanced Computations on Coarse Grids

Jan Nordström^{a,b,*}, Oskar Ålund^a

^aDepartment of Mathematics, Computational Mathematics, Linköping University, SE-581 83 Linköping, Sweden

^bDepartment of Mathematics and Applied Mathematics, University of Johannesburg, P.O. Box 524 Auckland Park 2006, South Africa

ARTICLE INFO

Article history:

Keywords: boundary layer, numerical oscillations, neural network, summation-by-parts, penalty terms, coarse grids

ABSTRACT

Unresolved gradients produce numerical oscillations and inaccurate results. The most straightforward solution to such a problem is to increase the resolution of the computational grid. However, this is often prohibitively expensive and may lead to excessive execution times. By training a neural network to predict the shape of the solution, we show that it is possible to reduce numerical oscillations and increase both accuracy and efficiency. Data from the neural network prediction is imposed using multiple penalty terms inside the domain.

© 2020 Elsevier Inc. All rights reserved.

1. Introduction

Unphysical oscillations due to steep gradients are common in numerical solutions of partial differential equations (see Figure 1a). Such oscillations are particularly prevalent in connection with so called boundary layers [1, 2, 3]. A boundary layer is essentially a quick decline or increase in the solution in close proximity of a boundary, such as a temperature drop near a heat source, or a reduction of speed close to a solid surface. If the boundary layer is not properly resolved, gradients will be over- or underestimated and produce oscillations, see Figure 1a.

The most straightforward solution to such a problem is to increase the resolution of the computational grid until the boundary layer is accurately predicted. However, this is often prohibitively expensive and may lead to infeasible execution times. Other approaches include adding artificial dissipation [4, 5], or employing filtering techniques [6], which may reduce the oscillations but do not improve the accuracy, rather reduce it.

A fairly recent idea that could potentially be applied in this context, called the multiple penalty technique (MPT), is outlined in [7] and expanded upon in [8, 9]. The authors study the effects of imposing redundant (from a continuum theoretical perspective) but accurate data in the interior of the domain. It was found that, if properly implemented, such imposition can improve both the accuracy and efficiency of the prediction.

In this paper we investigate the MPT in conjunction with data prediction using an artificial neural network (ANN) [10]. Suppose some limited knowledge of the shape of a boundary layer exists, either from measurement data, or from high resolution simulations. This knowledge can be used to train an ANN to predict the shape of the boundary layer under perturbed circumstances (i.e. for parameters of the problem where no prior knowledge of the boundary layer

*Corresponding author: e-mail: jan.nordstrom@liu.se

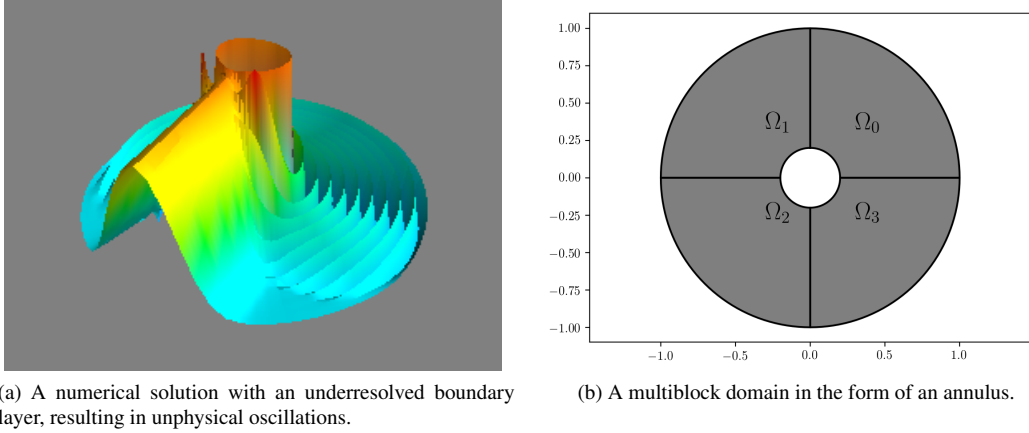


Fig. 1. A numerical solution to a linear advection–diffusion problem with a steep boundary layer.

exists). The key question to be answered in this note is: can a low resolution simulation be improved by imposing predictions using a trained ANN and the MPT?

The paper is structured as follows. In Section 2 we briefly discuss the discretization framework which merges discrete differential operators on Summation-By-Parts (SBP) form, with a penalty technique called the Simultaneous Approximation Term (SAT) for imposing boundary conditions [11, 12]. In Section 3 we outline the basic idea of the MPT in the context of a simple one-dimensional advection problem. Section 4 introduces an application with a boundary layer in a heat distribution problem that will serve as our model problem. Section 5 gives a brief overview of the theory of ANNs, followed by numerical experiments in Section 6. Finally, conclusions are drawn in Section 7.

2. Summation-By-Parts Operators and Weak Boundary Conditions

The numerical scheme employed in this paper is based on the method agnostic SBP-SAT technique [13]. In short, SBP operators are discrete derivative operators satisfying a discrete version of integration by parts, and SATs are penalty terms that push the solution toward compliance with the boundary conditions. This technique can be used for a multitude of methods, such as finite difference (FD) methods, finite volume (FV) methods, and discontinuous Galerkin (DG) methods [14, 15, 13, 16, 17, 18]. In this paper we will use FDs.

Consider an equidistant grid $x_i = i\Delta x$, $i = 0, 1, \dots, N$, on the unit interval, and let P be a diagonal, positive definite quadrature matrix. Then P defines an inner product on \mathbb{R}^{N+1} : $(\mathbf{u}, \mathbf{v}) := \mathbf{u}^\top P \mathbf{v}$ for all $\mathbf{u}, \mathbf{v} \in \mathbb{R}^{N+1}$, as well as an associated norm: $\|\mathbf{u}\| := \sqrt{(\mathbf{u}, \mathbf{u})}$. A matrix D is a discrete derivative operator if for any smooth function $u : [0, 1] \rightarrow \mathbb{R}$ and vector \mathbf{u} , we have $(D\mathbf{u})_i = u'(x_i) + \mathcal{O}(\Delta x^p)$, where p is an integer defining the order of the operator (p may vary with i —for FD based operators, p is lower close to the boundary). D is called an SBP operator if it satisfies

$$(\mathbf{u}, D\mathbf{v}) = u_N v_N - u_0 v_0 - (D\mathbf{u}, \mathbf{v}). \quad (1)$$

To illustrate the SBP-SAT technique, we consider the following one-dimensional linear advection problem:

$$u_t + u_x = 0, \quad x \in (0, 1), \quad t > 0, \quad (2)$$

where $u(t, 0) = g(t)$ and $u(0, x) = f(x)$. Let $\mathbf{e}_0 = (1, 0, 0, \dots, 0)^\top$ and consider the SBP-SAT approximation of (2):

$$\mathbf{u}_t + D\mathbf{u} = \alpha P^{-1}(u_0 - g)\mathbf{e}_0. \quad (3)$$

The left-hand side of (3) should be self-explanatory. The right-hand side is the SAT, whose purpose is to push the solution to satisfy the boundary condition. By deriving an expression for the time evolution of the P -norm of \mathbf{u} , we can choose α such that (3) is stable. Consider for simplicity the case when $g \equiv 0$, and multiply (3) by $2\mathbf{u}^\top P$. It follows from the SBP property (1) that

$$\|\mathbf{u}\|_t^2 = u_0^2 - u_N^2 + 2\alpha u_0^2 = -u_N^2 + (2\alpha + 1)u_0^2. \quad (4)$$

Hence, if $\alpha \leq -1/2$, the energy of the solution cannot grow with time.

The derivation above illustrates the basic structure of all SBP-SAT based schemes. The SBP-SAT technique for FD can be extended to multiple dimensions and curvilinear meshes, using Kronecker products and metric transformations [19]. Hence, with multiblock formulations and stable interface couplings [20], it is possible to treat a fairly general set of geometries. This fact will be leveraged in order to simulate heat distribution around a rod, see Figure 1b.

3. The Multiple Penalty Technique

To compute the solution to a PDE, one typically only impose a minimal number of conditions, so as to not overdetermine the problem (i.e. remove the existence of a solution). However, it has been shown that imposing additional solution data can actually aid numerical simulations if done properly [7, 8, 9]. Suppose that for some reason we know the solution value $h(t) := u(t, x_i)$ in the node x_i . Adding an additional internal penalty to (3) yields:

$$\mathbf{u}_t + D\mathbf{u} = \alpha P^{-1}(u_0 - g)\mathbf{e}_0 + \beta P^{-1}(u_i - h)\mathbf{e}_i. \quad (5)$$

We multiply (5) by $2\mathbf{u}^\top P$ and disregard the boundary terms. It follows that

$$\|\mathbf{u}\|_t^2 = 2\beta u_i^2 - 2\beta u_i h = \beta u_i^2 + \beta(u_i - h)^2 - \beta h^2. \quad (6)$$

Hence, if $\beta < 0$, the internal penalty does not introduce any instability in the scheme, i.e. the solution is bounded by data. The MPT is used in this paper to force low resolution simulations to conform to a highly resolved boundary layer, and to mitigate numerical oscillations. Other ways of using the MPT (also in time) are illustrated in [7, 9].

Remark 3.1. The function h is considered to be known data to the problem, and (6) implies a bound on \mathbf{u} in terms of h . In practice, h will approximate the exact solution in the node x_i and the internal penalty will guide the numerical solution toward h . If the approximation of the exact solution is bounded in x_i for all times, (e.g. if h is extracted from a stable approximation as in the application below) then (6) guarantees that also \mathbf{u} stays bounded for all times. Regarding accuracy, there are a number of factors that can affect the quality of the data. If data is available at points that do not coincide with the computational grid, one must interpolate to the grid, which results in loss of accuracy. One might also get inaccurate data from poor predictions (in our case due to inappropriate network architecture for example). Finally, if data is given by experiments, measurement errors are normally present.

4. An application

To illustrate our technique, we consider a heat distribution problem in incompressible flow around a rod. The governing equation is

$$T_t + \mathbf{a} \cdot \nabla T = \epsilon \Delta T, \quad (7)$$

where T is the temperature and \mathbf{a} is a given velocity field. Equation (7) is posed on the annulus shaped domain seen in Figure 1b. Along the inner circle we impose a constant Dirichlet condition, $T = 1$. Along the outer circle we impose a homogeneous Neumann condition ($\nabla T \cdot \mathbf{n} = 0$) at outflow, and a homogeneous Robin condition ($\mathbf{a} \cdot \mathbf{n}T - \epsilon \nabla T \cdot \mathbf{n} = 0$) at inflow. It can be shown that, with these boundary conditions, the problem is well-posed.

Various choices of ϵ and \mathbf{a} result in boundary layers of varying shapes forming along the inner boundary, as seen in Figure 2. With decreasing ϵ , the sharpness of the boundary layer increases. In Figure 1a we have used the SBP-SAT technique outlined in Section 2 to compute a steady state solution for $\epsilon = 0.01$ with the boundary layer deliberately underresolved, resulting in unphysical oscillations permeating the domain. This behavior is a common nuisance in computational fluid dynamics, which leads to computational inefficiency by requiring massive amounts of gridpoints.

5. Artificial Neural Networks

An artificial neural network (ANN) is a non-linear function loosely based on a biological neural network. ANNs are perhaps best illustrated using diagrams such as the one seen in Figure 3a. The network consists of a series of connected layers of computational neurons. The output of any particular layer is a non-linear function \mathbf{f} , called an activation function, applied to an affine transformation $W\mathbf{u} + \mathbf{b}$ of the input \mathbf{u} to the layer. The matrix W is called

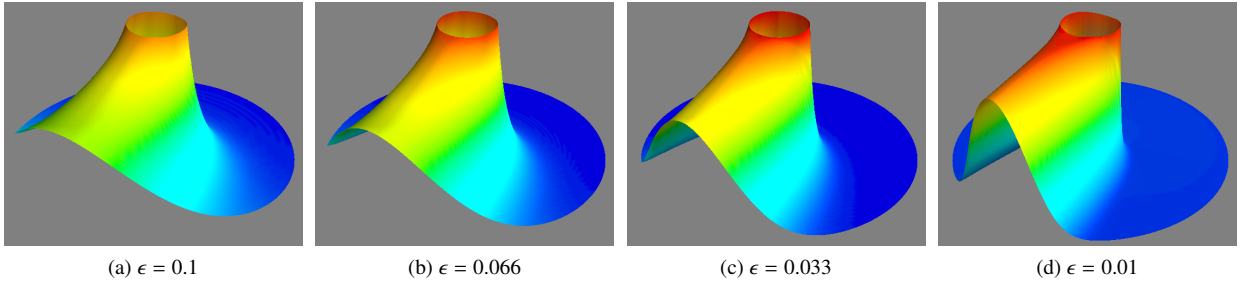


Fig. 2. Boundary layers for $\mathbf{a} = (1, 1)/\sqrt{2}$ and decreasing ϵ .

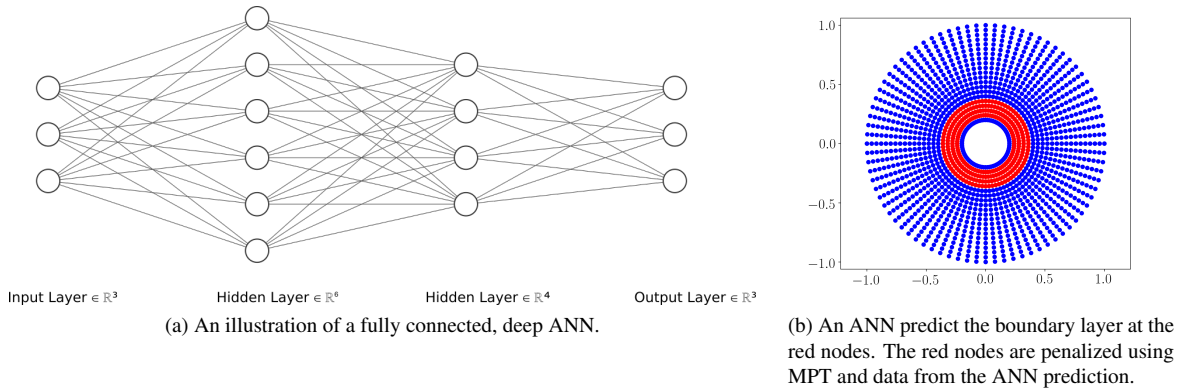


Fig. 3. A neural network (left panel) is used to predict the steady state near the inner circle (right panel).

a weight matrix and it represents the strengths of the neural connections between the current layer and the directly preceding layer. The vector \mathbf{b} consists of values called biases. Biases represent threshold potentials of the neurons.

Given that we know the correct outputs for some finite set of inputs, we can adjust the weights in the network such that the network conforms to this data. With proper training it will generalize to unseen inputs and reasonably accurately predict the correct outputs. The training of the ANN is typically done using stochastic gradient descent methods that minimize a loss function with respect to the network parameters, i.e. the weights and biases [10].

6. Boundary Layer Shape Inference

Suppose we know the shape of the boundary layer for a finite set of parameters, for example via measurements or from high resolution simulations. A neural network can then be trained to predict the shape of the boundary layer for

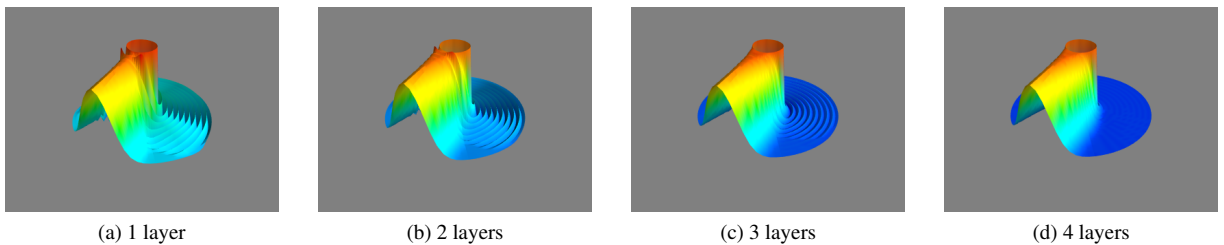


Fig. 4. A low resolution (21×21 per block) simulation with increasing layers of internal penalties (cf. Figure 3b which has four layers of internal penalties surrounding the inner circle of the domain). As the internal penalties capture more and more of the boundary layer, the oscillations are reduced.

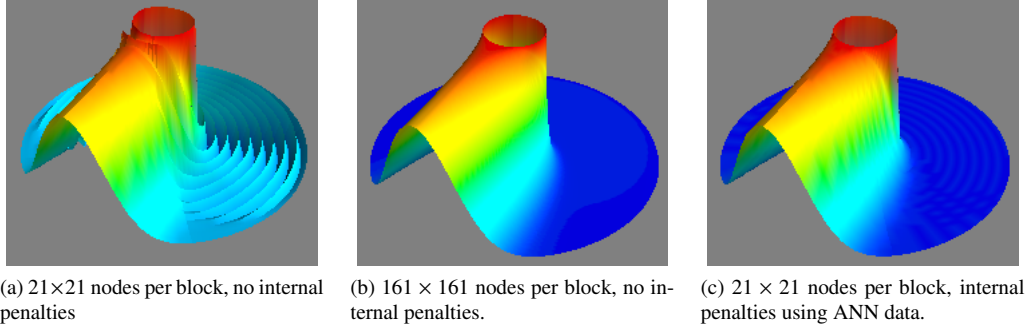


Fig. 5. Three steady state calculations of problem (7) with $\mathbf{a} = (1, 1)/\sqrt{2}$ and $\epsilon = 0.016$.

unknown parameters. Below we consider a deep, fully connected neural network and fix for simplicity $\mathbf{a} = (1, 1)/\sqrt{2}$. The network takes the diffusion parameter ϵ as input, and produces a prediction of the steady state solution at nodes in close proximity to the inner circle of the domain (see Figure 3b). All calculations are done using a third order accurate SBP-SAT scheme. The training data consists of a limited set of diffusion coefficients (as input) together with their corresponding steady state solution evaluations in nodes close to the inner circle (as output). More precisely we consider a fine grid of 161×161 nodes in each of the 4 blocks, and a coarse grid of 21×21 nodes per block, see Figure 1b. Steady state solutions are computed on the fine grid. Subsets of the solutions, consisting of $21 \times 21 \times 4$ node values at nodes that coincide with the coarse grid, are used as network outputs. We have chosen 4 layers of interior grid nodes as in Figure 3b because this is enough to enclose the whole boundary layer. It is possible to use fewer internal penalties, but this decreases the mitigating effect on numerical oscillations as shown in Figure 4.

We train a fully connected, deep ANN with seven hidden layers of 64 neurons each, based on high resolution solutions for a set of arbitrarily chosen diffusion coefficients listed below

$$\epsilon \in \{0.0100, 0.0224, 0.0348, 0.0472, 0.0596, 0.0720, 0.0845, 0.0969\}.$$

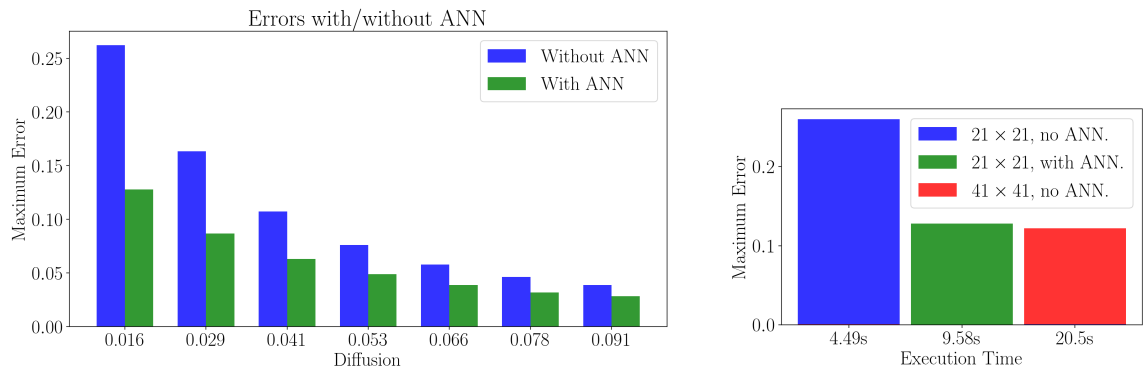
The hidden layers use ReLU [21] activation functions while the output layer uses linear activation functions. A mean squared loss function was used and the loss at the end of the training was 8.58×10^{-6} .

Remark 6.1. Selecting an appropriate network architecture can be challenging. If the network is not sufficiently rich, it will not be able to capture the details of the boundary layer, and if it is too rich it can lead to overfitting issues. The network architecture used here consistently captures the boundary layer over multiple training runs, but was arrived at through trial and error. Attempts to formalize architecture selection has been made however, see for example [22].

Next we compare coarse grid solutions, with and without internal penalties, against fine grid solutions for diffusion coefficients unseen by the ANN during training. Figure 5 illustrates the difference between a coarse grid simulation aided by internal penalties, and a standard coarse grid simulation enforcing only boundary data. Note that the oscillations permeating the domain in the simulation without internal penalties are mitigated by imposing the ANN prediction. The effect on the maximum error compared to the high resolution solution can be seen in Figure 6a. Naturally, the greatest effect of the ANN prediction can be seen for smaller ϵ , since that is when the boundary layer is steepest. For $\epsilon = 0.016$, a comparable error reduction can be achieved by grid refinement, as seen in Figure 6b, but with double the execution time compared to that of the ANN supported low resolution simulation.

7. Conclusions

We have shown that imposing artificial neural network predictions using multiple interior penalty terms can be a viable alternative to grid refinement for reducing numerical oscillations as well as increasing accuracy and efficiency for problems with steep gradients. This assumes that we have access to data for some set of problem parameters with which we can train the ANN. This technique can for example be used in design settings where multiple runs must be executed in a limited time. Further research is needed to assess the viability of this method in more complicated settings, such as for example steady/unsteady flows modelled by the Euler or Navier-Stokes equations.



(a) Errors for ANN aided simulations versus conventional simulations. The errors are computed against a high resolution numerical solution.

(b) Execution times and errors comparing the efficiency of ANN support versus grid refinement with $\epsilon = 0.016$.

Fig. 6. Numerical results showing performance increase due to ANN support.

References

- [1] C. Canuto, M. Y. Hussaini, A. Quarteroni, T. A. Zang, Spectral methods: fundamentals in single domains, Springer Science & Business Media, 2007.
- [2] B. Cockburn, G. E. Karniadakis, C.-W. Shu, Discontinuous Galerkin methods: theory, computation and applications, Vol. 11, Springer Science & Business Media, 2012.
- [3] Y. Bazilevs, C. Michler, V. Calo, T. Hughes, Weak dirichlet boundary conditions for wall-bounded turbulent flows, *Computer Methods in Applied Mechanics and Engineering* 196 (49-52) (2007) 4853–4862.
- [4] J. Blazek, Computational fluid dynamics: principles and applications, Butterworth-Heinemann, 2015.
- [5] K. Mattsson, M. Svård, J. Nordström, Stable and accurate artificial dissipation, *Journal of Scientific Computing* 21 (1) (2004) 57–79.
- [6] T. Lundquist, J. Nordström, Stable and accurate filtering procedures, *Journal of Scientific Computing* 82 (1) (2020) 1–21.
- [7] J. Nordström, Q. Abbas, B. A. Erickson, H. Frenander, A flexible boundary procedure for hyperbolic problems: multiple penalty terms applied in a domain, *Communications in Computational Physics* 16 (2) (2014) 345–358.
- [8] H. Frenander, J. Nordström, A stable and accurate davies-like relaxation procedure using multiple penalty terms for lateral boundary conditions, *Dynamics of Atmospheres and Oceans* 73 (2016) 34–46.
- [9] H. Frenander, J. Nordström, A stable and accurate relaxation technique using multiple penalty terms in space and time, *Dynamics of Atmospheres and Oceans* 79 (2017) 56–65.
- [10] G. Daniel, Principles of artificial neural networks, Vol. 7, World Scientific, 2013.
- [11] D. Funaro, D. Gottlieb, A new method of imposing boundary conditions in pseudospectral approximations of hyperbolic equations, *Mathematics of computation* 51 (184) (1988) 599–613.
- [12] M. H. Carpenter, D. Gottlieb, S. Abarbanel, Time-stable boundary conditions for finite-difference schemes solving hyperbolic systems: methodology and application to high-order compact schemes, *Journal of Computational Physics* 111 (2) (1994) 220–236.
- [13] M. Svård, J. Nordström, Review of summation-by-parts schemes for initial-boundary-value problems, *Journal of Computational Physics* 268 (2014) 17–38.
- [14] D. C. D. R. Fernández, P. D. Boom, D. W. Zingg, A generalized framework for nodal first derivative summation-by-parts operators, *Journal of Computational Physics* 266 (2014) 214–239.
- [15] J. Nordström, K. Forsberg, C. Adamsson, P. Eliasson, Finite volume methods, unstructured meshes and strict stability for hyperbolic problems, *Applied Numerical Mathematics* 45 (4) (2003) 453–473.
- [16] G. J. Gassner, A skew-symmetric discontinuous galerkin spectral element discretization and its relation to sbp-sat finite difference methods, *SIAM Journal on Scientific Computing* 35 (3) (2013) A1233–A1253.
- [17] D. C. D. R. Fernández, J. E. Hicken, D. W. Zingg, Review of summation-by-parts operators with simultaneous approximation terms for the numerical solution of partial differential equations, *Computers & Fluids* 95 (2014) 171–196.
- [18] K. Mattsson, F. Ham, G. Iaccarino, Stable and accurate wave-propagation in discontinuous media, *Journal of Computational Physics* 227 (19) (2008) 8753–8767.
- [19] O. Ålund, J. Nordström, Encapsulated high order difference operators on curvilinear non-conforming grids, *Journal of Computational Physics* 385 (2019) 209–224.
- [20] M. H. Carpenter, J. Nordström, D. Gottlieb, A stable and conservative interface treatment of arbitrary spatial accuracy, *Journal of Computational Physics* 148 (1999) 341–365.
- [21] G. E. Dahl, T. N. Sainath, G. E. Hinton, Improving deep neural networks for lvsr using rectified linear units and dropout, in: 2013 IEEE international conference on acoustics, speech and signal processing, IEEE, 2013, pp. 8609–8613.
- [22] W. C. Carpenter, M. E. Hoffman, Selecting the architecture of a class of back-propagation neural networks used as approximators, *AI EDAM* 11 (1) (1997) 33–44.