Accelerating Multivariate Functional Approximation Computation with Domain Decomposition Techniques

Vijay Mahadevan(✉), David Lenz, Iulian Grindeanu, and Thomas Peterka

Argonne National Laboratory, Lemont, IL 60439, USA
{mahadevan,dlenz,iulian,peterka}@anl.gov

Abstract. Modeling large datasets through Multivariate Functional Approximations (MFA) plays a critical role in scientific analysis and visualization workflows. However, this requires scalable data partitioning approaches to compute MFA representations in a reasonable amount of time. We propose a fully parallel and efficient method for computing MFA with B-spline bases without sacrificing the reconstructed solution accuracy or continuity. Our approach reduces the total work per task and uses a restricted Additive Schwarz (RAS) method to converge control point data across subdomain boundaries. We also provide a detailed analysis of the parallel approach with domain decomposition solvers to minimize subdomain error residuals and recover high-order continuity with optimal communication cost determined by the overlap regions in the RAS implementation. In contrast to previous methods that generally only recover C^1 continuity for arbitrary B-spline order p or required post-processing to blend discontinuities in the reconstructed data, the accuracy of the MFA remains bounded as the number of subdomains is increased. We demonstrate the effectiveness of our approach using analytical and scientific datasets in 1, 2, and 3 dimensions and show that it is highly scalable (due to bounded outer iteration counts) and that the parallel performance at scale is directly proportional to the nearest-neighbor communication implementations.

Keywords: functional approximation · domain decomposition · B-spline representations · additive Schwarz solvers

1 Introduction

Large-scale discrete data analysis of various scientific computational simulations often require high-order continuous functional representations that have to be evaluated anywhere in the domain. Such expansions described as *Multivariate Functional Approximations* (MFA) [4] in arbitrary dimensions allow the original

This work is supported by the U.S. Department of Energy, Office of Science, Advanced Scientific Computing Research under Contract DE-AC02-06CH11357, Program Manager Margaret Lentz.

J. Mikyška et al. (Eds.): ICCS 2023, LNCS 14073, pp. 89–103, 2023.
https://doi.org/10.1007/978-3-031-35995-8_7

discrete data to be modeled, and expressed in a compact form, in addition to supporting higher-order derivative queries (without further approximations such as finite differences) for complex data analysis tasks. MFA utilizes approximations of the raw discrete data using a hypervolume of piecewise continuous functions. One particular option is to use the variations of the B-Spline or NURBS bases [23,24] for the MFA *encoding* of scientific data. The reconstructed data in MFA retains the spatiotemporal contiguity, and statistical distributions, with lesser storage requirements. Due to the potentially large datasets that need to be encoded into MFA, the need for computationally efficient algorithms (in both time and memory) to parallelize the work is critically important. It is also essential to guarantee that the solution smoothness in the reconstructed (or *decoded*) dataset is consistently preserved when transitioning from a single MFA domain to multiple domains during parallelization.

Achieving improved performance without sacrificing discretization accuracy requires an infrastructure that is consistent in the error metrics of the decoded data and an algorithm that remains efficient in the limit of large number of parallel tasks. In this paper, we will utilize domain decomposition (DD) techniques [26] with data partitioning strategies to produce scalable MFA computation algorithms that minimize the reconstruction error when reproducing a given dataset. In such partitioned analysis, it is imperative to ensure that the continuity of the encoded and decoded data across subdomain interfaces is maintained, and remains consistent with the degree of underlying expansion bases used in MFA [23]. This is due to the fact that independently computing MFA approximations in individual subdomains do not guarantee even C^0 regularity in either the MFA space or in the reconstructed data. In order to tackle this issue, we rely on an iterative Schwarz-type DD scheme to ensure that continuity is enforced, and the overall error stays bounded as the number of subdomains are increased (or as the subdomain size decreases).

In addition to remaining efficient, we also require the devised algorithms to extend naturally to arbitrary dimensional settings and to handle large datasets. We next discuss some of the related work in the literature that have been explored for reconstruction of scattered data, and approaches to make these algorithms scalable in order to motivate the ideas presented in the paper.

1.1 Related Work

Domain decomposition (DD) techniques in general rely on the idea of splitting a larger domain of interest into smaller partitions or subdomains, which results in coupled Degrees-of-Freedom (DoF) at their common interfaces. Combining the application of DD schemes and NURBS bases with isogeometric analysis (IGA) [6,7] for high-fidelity modeling of nonlinear Partial Differential Equations (PDEs) [8,10,19] has enjoyed recent success at scale. However, many of these implementations lack full support to handle multiple geometric patches in a distributed memory setting due to non-trivial requirements on continuity constraints at patch boundaries. Directly imposing higher-order geometric continuity in IGA requires specialized parameterizations in order to preserve the approximation properties [16], which can be difficult to parallelize [15] generally.

To overcome some of these issues with discontinuities along NURBS or B-spline patches, Zhang et al. [28] proposed to use a gradient projection scheme to constrain the value (C^0), the gradient (C^1), and the Hessian (C^2) at a small number of test points for optimal shape recovery. Such a constrained projection yields coupled systems of equations for control point data for local patches, and results in a global minimization problem that needs to be solved.

Alternatively, it is possible to create a constrained recovery during the actual post-processing stage i.e., during the decoding stage of the MFA through standard blending techniques [14], in order to recover continuity in the decoded data. However, the underlying MFA representation remains discontinuous, and would become more so with increasing number of subdomains without the ability to recover higher-order derivatives along these boundaries. Moreover, selecting the amount of overlap and resulting width of the blending region relies strongly on a heuristic, which can be problematic for general problem settings.

In contrast, we propose extensions to the constrained solvers used by Zhang et al. [28], and introduce a two-level, DD-based, parallel iterative scheme to enforce the true degree of continuity, independent of the basis function polynomial degree p, unlike the low-order constraints used previously. The outer iteration utilizes the restricted Additive Schwarz (RAS) method [13], with efficient inner subdomain solvers that can handle linear Least-Squares systems to minimize the decoded residual within acceptable error tolerances. Such an iterative solver has low memory requirements that scales with growing number of subdomains, and necessitates only nearest-neighbor communication of the interface data once per outer iteration to converge towards consistent MFA solutions.

2 Approach

With motivations to accelerate the computation of an accurate MFA representation scalably, we utilize a data decomposition approach with overlapping subdomains to create shared layers of piecewise accurate functional reconstructions. This is similar to a multipatch approach typically taken in IGA computations [6,8]. However, in order to ensure that higher-order continuity across domain boundaries are preserved, an outer iteration loop is inevitable to converge the shared unknowns across the interfaces. These global iterations guarantee consistent MFA encodings in parallel, without which the representations will not even ensure C^0 regularity.

In this section, we first provide an illustrative example by formulating the constrained minimization problem to be solved in each subdomain and explain the iterative methodology used in the current work to converge the shared DoFs.

2.1 Numerical Background

A p-th degree NURBS or B-spline curve [24] is defined using the Cox-deBoor functions for each subdomain as

$$\boldsymbol{C}(u) = \sum_{i=0}^{n} R_{i,p}(u)\boldsymbol{P}(i), \quad \forall u \in \Omega \tag{1}$$

$$R_{i,p}(u) = \frac{N_{i,p}(u)W_i}{\sum_{i=0}^{n} N_{i,p}(u)W_i} \tag{2}$$

where $R_{i,p}(u)$ are the piecewise rational functions with $\boldsymbol{P}$ control points of size n, W_i are the control point weights, with the p-th degree B-spline bases $N_{i,p}(u)$ defined on a knot vector u.

Given a set of input points $\boldsymbol{Q}$ that need to be encoded into a MFA, with the weights $W = 1$ (B-spline representations) for simplicity, the unconstrained minimization problem to compute the optimal set of control point locations within a subdomain can be posed as a solution to a linear Least-SQuares (LSQ) system that minimizes the net error of the B-spline approximation.

$$\begin{aligned} &\arg\min_{\boldsymbol{P} \in \mathbb{R}^n} E = \|\boldsymbol{Q} - R\boldsymbol{P}\|_{L_2}, \\ &\text{where } R \in \mathbb{R}^{m \times n} \ , \ \boldsymbol{Q} \in \mathbb{R}^m \end{aligned} \tag{3}$$

An appropriate LSQ solver such as the one based on Cholesky decomposition or the more efficient ℓ-BFGS scheme [29] can compute the control point solution $\boldsymbol{P}$ that minimizes the residual error $\boldsymbol{E}$ for the given input data $\boldsymbol{Q}$ and MFA representation of degree p. Note that the minimization procedure can be performed independently on each subdomain without dependencies as there are no constraints explicitly specified in Eq. (3). However, in order to recover high-order continuity across subdomain interfaces, computing the unconstrained solution is insufficient. At the minimum, the shared DoFs on subdomain boundaries have to match to recover C^0 continuity in the decoded data ($R\boldsymbol{P}$).

A straightforward approach to achieve C^0 continuity in the recovered solution is by ensuring that the common control point data $\boldsymbol{P}$ at subdomain interfaces are clamped with repeated knots, in addition to using clamping at the global domain boundaries. In this scheme, the control points exactly interpolate (are clamped to) input data points at the subdomain interface boundaries. Such an approach requires in general a good spatial distribution of $\boldsymbol{Q}$, and yields only low-order continuous approximations (C^0) when the solution remains smooth across the subdomain interfaces. It should also be noted that as the number of subdomains increases, the global solution being computed becomes further constrained, and more interpolatory due to clamped DoFs. Moreover, the MFA solution computed becomes dependent on the number of subdomains used to decompose the problem; i.e., the global control point data $\boldsymbol{P}$ recovers different reconstructions as a function of number of subdomains ($\mathcal{N}$) used.

While the implementation of the domain decomposed MFA can be much simpler with clamped knots on all subdomain boundaries, ensuring higher-order continuity would require that all $p - 1$ derivatives of the approximation match

as well. As a continuous extension, one could relax the interpolatory behavior of clamped knot boundaries by reducing the number of repeated knots, and instead use unclamped knots at internal subdomain boundary interfaces by sharing knot spans between subdomains.

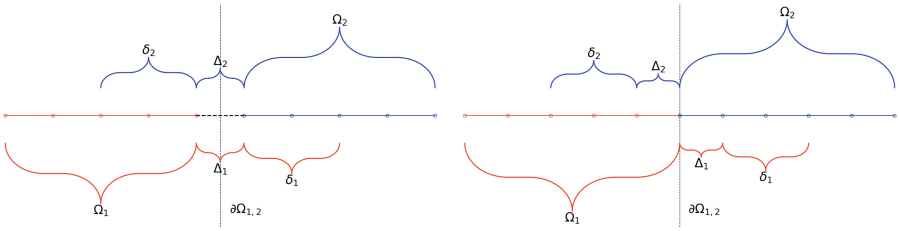
More generally, the constrained minimization problem to recover continuity [24] can be formulated as

$$R\boldsymbol{P} = \boldsymbol{Q} \quad | \quad \mathcal{C}\boldsymbol{P} = \boldsymbol{G}, \tag{4}$$

where $\mathcal{C}$ is the constraint matrix imposing continuity restrictions on the control points $\boldsymbol{P}$ along with its derivatives, with data exchanged from neighboring domains stored in $\boldsymbol{G}$, around the neighborhood of the interface $\Omega_{i,j}$ shared by subdomains i and j. With the use of penalized constraints ($\mathcal{C}$) and Lagrange multipliers [12,22], the solution to the constrained LSQ problem can recover optimal control point values. This modification allows us to recover fully consistent (C^0 to C^{p-1}) continuous MFA reconstructions using the solution procedure detailed for the global constrained minimization problem Eq. (4).

2.2 Shared Knot Spans at Subdomain Interfaces

In the current work, in contrast to using clamped knots, we utilize unclamped (floating), shared knot spans at all interior subdomain boundaries such that the high-order continuity and consistency of the decoded solution are recovered, independent of $\mathcal{N}$ (Fig. 1).



(a) Even degree $p = 2$ (b) Odd degree $p = 3$

Fig. 1. Illustration: 1D parallel partitioned domain with unclamped interior knots and augmented spans ($|\delta| = 2$)

For the purpose of illustration and to explain the proposed solver methodology, let us consider a simple one dimensional domain (Ω) with two subdomains ($\mathcal{N} = 2$) as shown in Fig. (1), where Ω_1 and Ω_2 represent the subdomains that share an interface $\partial\Omega_{1,2}$. In Fig. (1), the layout of the knot spans for both an even degree ($p = 2$) and odd degree ($p = 3$) are shown. For generality, we also introduce here an overlap layer Δ_1 and Δ_2 on each subdomain that represents the set of shared knot spans with its adjacent subdomain (for internal boundaries), and an optional augmented layer δ_1 and δ_2 that has a connotation similar

to that of an overlap region in traditional DD schemes [26]. Note that in order to reconstruct the input data in $\Omega_i, \forall i \in [1,2]$, the knot spans must mandatorily include Δ_i regions. This Δ_i overlap region is required by definition to maintain partition of unity of a B-spline curve in order to evaluate Eq. (2). For generality, Δ_i represents the repeated knots along clamped global domain boundaries, and the shared knots between two subdomains in the unclamped interior boundaries. For arbitrary degree p, the number of knot spans in Δ_i is given by $\lfloor \frac{p}{2} \rfloor$, where $\lfloor . \rfloor$ represents the floor operator. In multidimensional tensor product expansions, these shared spans are replaced by shared layers of knot spans along the subdomain interfaces. The δ_i regions are additional, and optional, shared knot spans that can help improve error convergence in a manner similar to overlap regions in DD methods used for PDE solvers [13,26].

Note that the control point DoF vector can be represented by three separate parts based on the local support of the basis expansion. The control point vector is in general given as $\boldsymbol{P}(\Omega \cup \Delta \cup \delta) = [\boldsymbol{P}(\Omega); \boldsymbol{P}(\Delta); \boldsymbol{P}(\delta)]$. However, the owned DoFs represented by $\boldsymbol{P}(\Omega)$ is the only component computed through the LSQ subdomain solver, while the other components, $\boldsymbol{P}(\Delta)$ and $\boldsymbol{P}(\delta)$, are only used to impose the constraints to recover continuity.

Now, the constrained minimization problem for the two subdomain case can be written as

$$\left[\begin{array}{c|c} R_1(\Omega_1) & \lambda_{1,2}(\Delta_1 \cup \delta_1) \\ \hline \lambda_{2,1}(\Delta_2 \cup \delta_2) & R_2(\Omega_2) \end{array}\right] \left[\begin{array}{c} \boldsymbol{P}_1(\Omega_1 \cup \Delta_1 \cup \delta_1) \\ \boldsymbol{P}_2(\Omega_2 \cup \Delta_2 \cup \delta_2) \end{array}\right] = \left[\begin{array}{c} \boldsymbol{Q}_1 \\ \boldsymbol{Q}_2 \end{array}\right] \tag{5}$$

where the diagonal operators R_1 and R_2 are the piecewise rational functions that minimize the local subdomain residuals in $\Omega_j, \forall j \in [1,2]$, while the off-diagonal blocks $\lambda_{1,2}$ and $\lambda_{2,1}$ represent the coupling terms between the subdomains near the interface $\partial\Omega_{1,2}$. This coupling term provides the constraints on the shared control point data, and higher-order derivatives as needed to recover smoothness and enforce continuity along subdomain boundaries. For higher dimensional problems, the constraints on the control points must include both face neighbor and diagonal neighbor contributions to accurately determine the globally consistent minimization problem.

The coupling blocks $\lambda_{i,j}$ can be viewed as Lagrange multipliers that explicitly couple the control point DoFs across a subdomain interface $(\boldsymbol{P}_1(\Omega_1 \cup \Delta_1 \cup \delta_1) \cap \boldsymbol{P}_2(\Omega_2 \cup \Delta_2 \cup \delta_2))$ such that continuity is preserved in a weak sense [24]. Using appropriate Schur complements to eliminate the coupled DoF contributions in each subdomain, with $\lambda_{i,j}$ evaluated at *lagged* iterates of adjacent subdomains, the set of coupled constrained equations in Eq. (5) can be completely decoupled for each subdomain. This modified system resembles a block-Jacobi operator of the global system. The scheme illustrated in this section follows ideas similar to the Jacobi-Schwarz method [13] and the overlapping RAS method [27].

In the above description, the coupled control point patches, $\boldsymbol{P}_1(\Delta_1)$ and $\boldsymbol{P}_2(\Delta_2)$ belonging to adjacent subdomains near $\partial\Omega_{1,2}$ are exchanged simultaneously between Ω_1 and Ω_2 before the local domain solves are computed independently. One key advantage with such a DD scheme is that only nearest neighbor

exchange of data is required, which keeps communication costs bounded as the number of subdomains $\mathcal{N}$ increase [27], while providing opportunities to interlace recomputation of the constrained control point solution. Note that in such an iterative scheme, nearest neighbor exchanges can be performed compactly per dimension and direction, thereby minimizing communication costs and eliminating expensive global collectives.

Augmenting Knot Spans with Overlap: One of the key metrics of interest is that the parallel solver infrastructure does not amplify any approximation errors unresolved by the tensor product B-spline mesh. Since the local decoupled subdomain solution is encoded accurately to satisfy Eq. (3) in each individual subdomain without any data communication (i.e., embarassingly parallel), imposing the constraints for the shared DoFs in Δ should ensure the error change is bounded. However, as the control point data across subdomains become synchronized, numerical artifacts, especially for high-degree ($p > 2$) basis reconstructions at subdomain interfaces can become dominant sources of error. A key validation metric is to compare the error profiles from the multiple subdomain cases to the single subdomain case, in order to ensure convergence of the solvers to the same unique solution, independent of $\mathcal{N}$.

For many problem domains, overlapping Schwarz solvers [13,18] have been proven to be more stable, efficient and scalable compared to non-overlapping variants [3,27]. We utilize the concept of overlap regions by sharing additional knot spans between subdomains in order to produce better MFA reconstructions of the underlying data. This user-specified, additional overlap is described by $\delta_j, \forall j \in [1, 2]$ in Fig. (1). The amount of data overlap utilized for computing the functional approximation can directly affect the accuracy of the subdomain solver, and the scalability of the algorithm. Additionally, the presence of the augmented knot spans in δ keep the residual errors $\boldsymbol{E}$ bounded as the number of subdomain increase with appropriate overlap regions. Note that the optimal range of δ depends on the gradient of the input data near the subdomain interfaces, i.e., $|\delta| = 0$ may suffice for reconstructing smoother solution profiles, while strong gradient reconstructions may require $|\delta| = p$ or $|\delta| = 2p$ in general.

2.3 Solver Workflow

Computing the functional approximation of large datasets require efficient solvers at two levels: first, the local decoupled subdomain problem in Eq. (3), and next, the constrained minimization problem in Eq. (4). Hence, the global problem reduces to a series of minimization problems in each subdomain.

Subdomain Solvers: For the linear LSQ solvers that can be used to compute local subdomain control point solution $\boldsymbol{P}$, there are a variety of choices available. Direct methods like Singular Value decomposition or Cholesky decomposition operating on the normal equations [2] can compute optimal values. Alternatively, the iterative LSQ solvers such as orthogonal decomposition methods based on

QR and QZ factorizations are more stable, especially when the normal form of the operator, $R^T R$, is ill-conditioned.

Restricted Additive-Schwarz Solvers: The outer RAS iterations work together with nearest neighbor communication procedures to exchange shared DoF data between adjacent subdomains. This is an important step to ensure that $\boldsymbol{P}$ computed through the LSQ procedure is consistent and high-order continuous across subdomain boundaries. The final minimized control point solution is achieved when the interface solutions $\boldsymbol{P}(\Delta)$ match on all $\partial\Omega_{i,j} \in \Omega$.

It is also important to note that unlike the blending approaches that can be directly applied on decoded data [14], the numerical error with the constrained iterative scheme is not bounded by the original partitioned, unconstrained least-squares solution; i,e., imposing subdomain boundary constraints can create artificial numerical peaks (non-monotonic) in reconstructed data as we converge towards continuity recovery. To address this issue, we can increase the subdomain overlap δ to ensure uniform convergence to the true single-subdomain solution error, in the limit of $\mathcal{N} \to \infty$.

2.4 Implementation

The DD techniques presented here for MFA computation are primarily implemented in Python-3, with main dependencies on `SciPy` for B-spline bases evaluations and linear algebra routines. Additionally, the drivers utilize Python bindings (`PyDIY`) for the `DIY` [20] C++ library. `DIY` is a programming model and runtime for block-parallel analytics on distributed-memory machines, built on `MPI-3`. Rather than programming for process parallelism directly in `MPI`, the programming model in `DIY` is based on block parallelism. In `DIY`, data are decomposed into subdomains called blocks. One or more of these blocks are assigned to processing elements (processes or threads) and the computation is described over these blocks, and communication between blocks is defined by reusable patterns. `PyDIY` utilizes `PyBind11`[1] to expose the C++ library in a Pythonic way. In our implementation, `PyDIY` is exclusively used to manage the data decomposition, including specifications to share an interface $\partial\Omega_{i,j}$ and ghost layers that represent the $\Delta \cup \delta$ overlapping domains.

The implementation of non-overlapping and overlapping RAS schemes applied to the computation of MFA exhibits scalable convergence properties in the limit of decreasing subdomain size (i.e., as $\mathcal{N} \to \infty$). This is a favorable property for strong scaling, especially when tackling large datasets, as the net computational cost always remains bounded. This behavior can be explained by the nature of how the RAS iterative procedure resolves the shared DoFs.

[1] PyBind11: https://github.com/pybind/pybind11 (last accessed 01/24/2023).

3 Results

To demonstrate the effectiveness of the iterative algorithm for MFA computation, we devised a series of analytical closed form functionals and utilized real-world scientific datasets in both 2 and 3 dimensions obtained from high-fidelity simulations. We also define a *compression ratio* (η), which gives the ratio of total input points in the dataset ($dim\ \boldsymbol{Q}$) to the total control points ($dim\ \boldsymbol{P}$) used in the MFA B-spline representation. Note that as $\eta \to 1$, one can achieve smaller error residuals compared with the reference data for a given degree p, while $\eta \gg 1$ produces smoother approximations with potentially larger pointwise errors.

3.1 Error Convergence Analysis

For a domain $\Omega(x, y, z) = [-4, 4]^3$, we define closed-form synthetic datasets shown in Eq. (6), Eq. (7) and Eq. (8) for 1D, 2D and 3D respectively to verify error convergence and to demonstrate the parallel scalability of MFA computation.

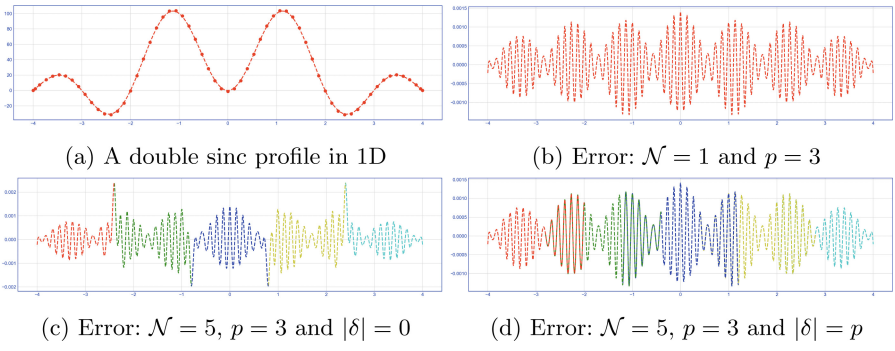
$$F(x) = \mathcal{S}(x+1) + \mathcal{S}(x-1), \forall x \in \Omega = [-4, 4], \tag{6}$$

$$F(x, y) = \mathcal{S}(\sqrt{x^2 + y^2}) + \mathcal{S}(2(x-2)^2 + 2(y+2)^2), \tag{7}$$

$$F(x, y, z) = \mathcal{S}(\sqrt{x^2 + y^2 + z^2}) + \mathcal{S}(2(x-2)^2 + (y+2)^2 + (z-2)^2), \tag{8}$$

where $\mathcal{S}(x) = \frac{sin(x)}{x}$.

First, to determine the effect of using augmented or overlapped knot span regions (δ) as the number of subdomains $\mathcal{N}$ are increased, we use the 1D function in Eq. (6) on a single subdomain as the reference solution (Fig. (2a)). The error profiles for $\mathcal{N} = 5$ using non-overlapping spans ($|\delta| = 0$) and augmented spans ($|\delta| = 3$) are also shown in Fig. (2c) and Fig. (2d) respectively.



(a) A double sinc profile in 1D

(b) Error: $\mathcal{N} = 1$ and $p = 3$

(c) Error: $\mathcal{N} = 5$, $p = 3$ and $|\delta| = 0$

(d) Error: $\mathcal{N} = 5$, $p = 3$ and $|\delta| = p$

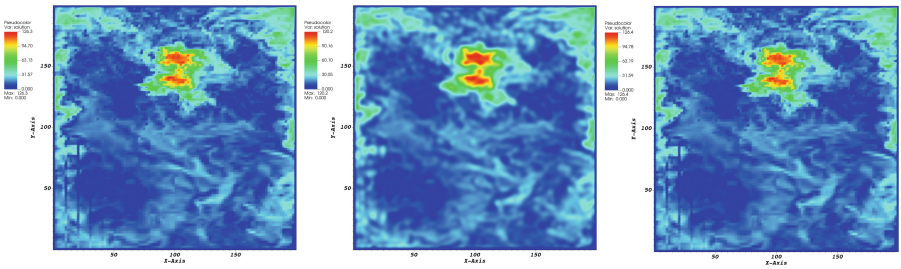
Fig. 2. Demonstration of error convergence, and effect of the overlapping spans to minimize numerical artifacts

It is evident from Fig. (2c) that in the non-overlapping case, the reconstructed data at subdomain boundaries are influenced by contributions from both adjacent domain DoFs, which are enforced to be C^{p-1} continuous by the constrained minimization solver. However, as we increase the number of overlap regions in terms of both the underlying data and the local bases support spans, the error profiles as shown in Fig. (2d) approaches the reference profile (with $\mathcal{N} = 1$) shown in Fig. (2b). Heuristically, for many of the problems tested, using $|\delta| = p$ provides optimal error convergence as number of subdomains increase, even though increasing this parameter to $|\delta| = 2p$ or higher will in general always improve the numerical accuracy at the cost of higher communication costs between neighboring subdomains.

3.2 Real Simulation Datasets

Next, we apply the parallel MFA algorithm to some real world examples cases.

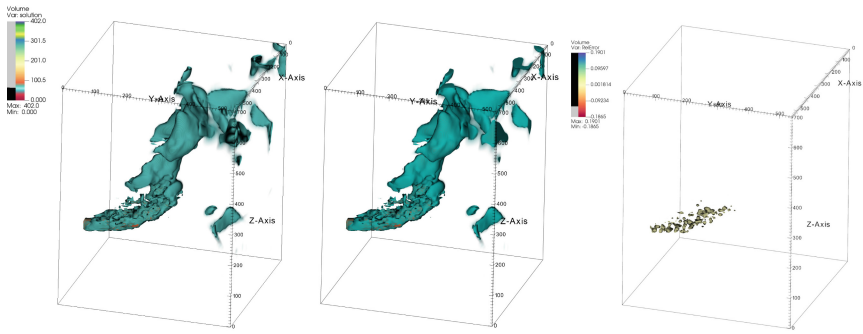
Computational Fluid Dynamics Dataset: In this study, we utilized a 3D dataset from the large-eddy simulation of Navier-Stokes equations for validation of MAX experiments [25] using Nek5000 [11]. The velocity field data is representative of turbulent mixing and thermal striping that occurs in the upper plenum of liquid sodium fast reactors. For the 2D analysis, a slice of the velocity magnitude was resampled onto a 200×200 regular grid [23]. The reference solution and the converged, reconstructed solution with $\mathcal{N} = 5 \times 5 = 25$ subdomains with $p = 6$ and $|\delta| = 2p$ is shown in Fig. (3) for different η. Depending on the use case for MFA reconstruction, the converged error norms with 20 floating control points per subdomain yielding $\eta = 4$ is sufficient to evaluate continuous derivatives everywhere in the domain Ω. The decoded MFA representation (with $\eta = 1$) is also shown in Fig. (3c), which can fully reconstruct the sharp features in the input dataset in contrast to a lossy smoothing shown in Fig. (3b) ($\eta = 4$).



(a) Reference solution (b) MFA Solution: $\eta = 4$ (c) MFA Solution: $\eta = 1$

Fig. 3. Velocity profile: reference solution (left), B-spline MFA with $p = 6$, $\mathcal{N} = 5 \times 5$, $|\delta| = p$ for $\eta = 4$ (middle), and $\eta = 1$ (right).

Turbulent Combustion Dataset: S3D is a turbulent fuel jet combustion dataset generated from a simulation in the presence of an external cross-flow [5]. The 3D domain has the span $|\Omega| = 704 \times 540 \times 550$, containing three components of the vector field. The magnitude of the velocity field is shown below in Fig. (4a) with 209M points. The converged MFA reconstruction is shown in Fig. (4b) with $8^3 = 512$ subdomains and $n = 62$ per direction in each subdomain yields $\eta = \frac{(704 \times 540 \times 550)}{(8 \times 62)^3} \approx 1.72$. While uniform knot refinement does yield sufficient error reductions in most subdomains, adaptive knot insertion with the DeCasteljau algorithm [24] can reduce the relatively large errors (as shown in Fig. (4c)) to provide better reconstructions everywhere. This was demonstrated on a single subdomain previously [21], but with minor modifications to the communication routines, we can naturally extend the algorithm to handle adaptive knot spans with high-order continuity.



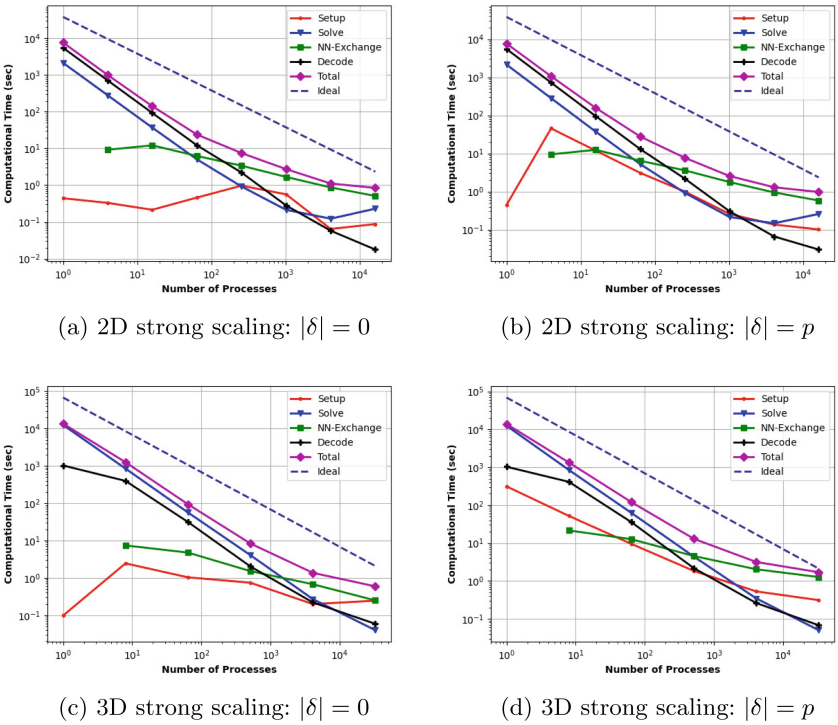
(a) S3D dataset profile (b) Converged solution (c) Final error profile

Fig. 4. Volume rendered S3D dataset with $\eta = 1.72$: reference profile (left), converged MFA decoded profile (middle) and the corresponding reconstruction error (right) with $\mathcal{N} = 8 \times 8 \times 8 = 512$, $p = 3$ and $|\delta| = 2p = 6$

3.3 Parallel Scalability

The strong scaling studies using Eq. (7) for 2D and Eq. (8) for 3D were performed on the Theta Cray XC40 supercomputer operated by the Argonne Leadership Computing Facility (ALCF), which provides 4,392 KNL compute nodes with 64 compute cores and 192 GB DDR4 RAM per node. The strong scaling tests in Fig. (5) were performed on 1 to 16,384 tasks in 2D, increasing by a factor of $2^2 = 4$, and the 3D tests were executed on 1 to 32,768 tasks, increasing by a factor of $2^3 = 8$. Note that the 2D studies used 400M input points with $\eta = 4$ and 3D cases used 1.331B points with $\eta = 1.25$ for all runs. In order to better understand the effects of using augmented overlaps (δ) on scalability, two cases with $|\delta| = 0$ and $|\delta| = p$ were also tested. A breakdown of the timings for each

task in computing the parallel MFA representation is plotted for all cases. This task-wise breakdown helps us clearly visualize the operations that scale optimally and the ones that do not (Fig. 5).



(a) 2D strong scaling: $|\delta| = 0$

(b) 2D strong scaling: $|\delta| = p$

(c) 3D strong scaling: $|\delta| = 0$

(d) 3D strong scaling: $|\delta| = p$

Fig. 5. Strong scaling performance with $p = 3$ and one subdomain per task for non-overlapping (left) and overrlapping (right) RAS schemes

The data partitioned RAS iterative scheme shows good strong scalability for the large dataset experiments, and overall time to compute the MFA in parallel is reduced at a nearly ideal rate up to 8192 MPI tasks in 3D as $\mathcal{N}$ increases, while ensuring C^{p-1} continuity in the subdomain interfaces. It is important to note that the dominant computational time is usually driven by the decoupled LSQ solution computation and the decoding operations, which are embarrassingly parallel as the size of the subdomains decreases in direct proportion to the tasks. Given that the scalability of the linear algebraic LSQ solvers [1,9] and Sparse Matrix Vector (SpMV) products used in the decode tasks are well understood, the bottlenecks potentially occur primarily due to nearest neighbor communication for constraint data exchange between adjacent subdomains. The overall scalability of the algorithm becomes sub-linear when the cost of nearest neighbor exchange cross over the cost of the subdomain solve.

The overall strong scaling efficiency of RAS for MFA remains around 50% for both overlapping and non-overlapping 2D problem cases at 16,384 tasks. However, the added setup cost and sub-optimally scaling nearest neighbor communications reduce the 3D problem efficiency for the overlapping subdomain cases to 25% at 32,768 tasks from 70% in comparison to the non-overlapping cases. The results confirm that the communication cost at scale is driven by the size of the messages being transferred (determined by $|\delta|$) between subdomains.

4 Summary

We have presented a scalable DD approach to tackle the issue of discontinuous B-spline based MFA representations when performing the computations in parallel. The Restricted Additive Schwarz (RAS) method is a natural algorithmic fit for data analysis problems to reconstruct MFA representations in a scalable workflow. Through the use of overlapping Schwarz-based iterative schemes, combined with constrained local subdomain solvers, the two-level iterative technique has been shown to be robust in converging to the functional representation of the given data, without deviating from the single subdomain accuracy metrics.

We have demonstrated that the use of overlap layers δ does preserve the overall MFA accuracy in comparison to the single subdomain case. We determined that for all the problems tested, including real datasets, $|\delta| = p$ to $|\delta| = 2p$ is nearly optimal in terms of error recovery and computational cost even for 3D problems up to 32,768 tasks. A stronger theory for deriving an optimal $|\delta|$ measure will be part of the future work. The MFA-DD scheme applied to both 2D and 3D problems show good parallel scalability and degrades only when the cost of nearest neighbor subdomain data exchanges start to creep up beyond the cost of the locally constrained subdomain solves. Given that scaling characteristics of these sub-processes are well understood in the literature, the parallel speedups behave predictably well at scale on the large computing machines tested.

Another natural extension to improve the accuracy is to utilize a hierarchy of control lattices (multilevel B-spline approximations [17]) to generate a sequence of MFA, the sum of which produces better approximations to datasets with strong gradients. Note that Replacing B-spline bases with NURBS bases ($W \neq 1$) only requires imposing the constraints on the $\boldsymbol{P}_i W_i$ data instead of $\boldsymbol{P}_i$, and will be explored in the future.

Acknowledgements. This work is supported by Advanced Scientific Computing Research, Office of Science, U.S. Department of Energy, under Contract DE-AC02-06CH11357, program manager Margaret Lentz. This research used the Theta supercomputer operated by the Argonne Leadership Computing Facility (ALCF), a DOE Office of Science User Facility supported under Contract No. DE-AC02-06CH11357, and the Bebop cluster operated by the Argonne Laboratory Computing Resource Center (LCRC) at Argonne National Laboratory.

References

1. Benzi, M.: Preconditioning techniques for large linear systems: a survey. J. Comput. Phys. **182**(2), 418–477 (2002)
2. Björck, Å.: Numerical Methods for Least Squares Problems. SIAM (1996)
3. Bjørstad, P.E., Widlund, O.B.: To overlap or not to overlap: a note on a domain decomposition method for elliptic problems. SIAM J. Sci. Stat. Comput. **10**(5), 1053–1061 (1989)
4. de Boor, C., DeVore, R.: Approximation by smooth multivariate splines. Trans. Am. Math. Soc. **276**(2), 775–788 (1983)
5. Chen, J.H., et al.: Terascale direct numerical simulations of turbulent combustion using s3d. Comput. Sci. Discov. **2**(1), 015001 (2009)
6. Cottrell, J.A., Hughes, T.J., Bazilevs, Y.: Isogeometric Analysis: Toward Integration of CAD and FEA. Wiley, Hoboken (2009)
7. Da Veiga, L.B., Cho, D., Pavarino, L.F., Scacchi, S.: Overlapping Schwarz methods for isogeometric analysis. SIAM J. Numer. Anal. **50**(3), 1394–1416 (2012)
8. Dalcin, L., Collier, N., Vignal, P., Côrtes, A., Calo, V.M.: PetIGA: a framework for high-performance isogeometric analysis. Comput. Methods Appl. Mech. Eng. **308**, 151–181 (2016)
9. Dalcin, L.D., Paz, R.R., Kler, P.A., Cosimo, A.: Parallel distributed computing using python. Adv. Water Resour. **34**(9), 1124–1139 (2011)
10. Dedè, L., Quarteroni, A.: Isogeometric analysis for second order partial differential equations on surfaces. Comput. Methods Appl. Mech. Eng. **284**, 807–834 (2015)
11. Deville, M.O., Fischer, P.F., Mund, E.H.: High-Order Methods for Incompressible Fluid Flow, vol. 9. Cambridge University Press, Cambridge (2002)
12. Dornisch, W., Klinkel, S.: Boundary conditions and multi-patch connections in isogeometric analysis. PAMM **11**(1), 207–208 (2011)
13. Efstathiou, E., Gander, M.J.: Why restricted additive Schwarz converges faster than additive Schwarz. BIT Numer. Math. **43**(5), 945–959 (2003)
14. Grindeanu, I., Peterka, T., Mahadevan, V.S., Nashed, Y.S.: Scalable, high-order continuity across block boundaries of functional approximations computed in parallel. In: 2019 IEEE International Conference on Cluster Computing (CLUSTER), pp. 1–9. IEEE (2019)
15. Hofer, C., Langer, U.: Fast Multipatch Isogeometric Analysis Solvers. Ph.D. thesis, Johannes Kepler University Linz (2018)
16. Kapl, M., Sangalli, G., Takacs, T.: Construction of analysis-suitable G1 planar multi-patch parameterizations. Comput. Aided Des. **97**, 41–55 (2018)
17. Lee, S., Wolberg, G., Shin, S.Y.: Scattered data interpolation with multilevel B-splines. IEEE Trans. Visual Comput. Graph. **3**(3), 228–244 (1997)
18. Lions, P.L.: On the schwarz alternating method. i. In: First International Symposium on Domain Decomposition Methods for Partial Differential Equations, vol. 1, p. 42. Paris, France (1988)
19. Marini, F.: Parallel Additive Schwarz Preconditioning for Isogeometric Analysis. Ph.D. thesis, Università degli Studi di Milano (2015)
20. Morozov, D., Peterka, T.: Block-Parallel Data Analysis with DIY2. In: Proceedings of the 2016 IEEE Large Data Analysis and Visualization Symposium LDAV 2016. Baltimore (2016)
21. Nashed, Y.S.G., Peterka, T., Mahadevan, V., Grindeanu, I.: Rational approximation of scientific data. In: Rodrigues, J.M.F., et al. (eds.) ICCS 2019. LNCS, vol. 11536, pp. 18–31. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-22734-0_2

22. Paul, K., Zimmermann, C., Duong, T.X., Sauer, R.A.: Isogeometric continuity constraints for multi-patch shells governed by fourth-order deformation and phase field models. Comput. Methods Appl. Mech. Eng. **370**, 113219 (2020)
23. Peterka, T., Youssef, S., Grindeanu, I., Mahadevan, V.S., Yeh, R., Tricoche, X., et al.: Foundations of multivariate functional approximation for scientific data. In: 2018 IEEE 8th Symposium on Large Data Analysis and Visualization (LDAV), pp. 61–71 (2018)
24. Piegl, L., Tiller, W.: The NURBS Book. Springer, Heidelberg (2012)
25. Pointer, W., Lomperski, S., Fischer, P.: Validation of CFD methods for advanced SFR design: upper plenum thermal striping and stratification. In: Proc. International Conference on Nuclear Engineering ICONE17, Brussels, Belgium (2009)
26. Smith, B., Bjorstad, P., Gropp, W.: Domain Decomposition: Parallel Multilevel Methods for Elliptic Partial Differential Equations. Cambridge University Press, Cambridge (2004)
27. St-Cyr, A., Gander, M.J., Thomas, S.J.: Optimized multiplicative, additive, and restricted additive Schwarz preconditioning. SIAM J. Sci. Comput. **29**(6), 2402–2425 (2007)
28. Zhang, X., Wang, Y., Gugala, M., Müller, J.D.: Geometric continuity constraints for adjacent NURBS patches in shape optimisation. In: ECCOMAS Congress, vol. 2, p. 9316 (2016)
29. Zheng, W., Bo, P., Liu, Y., Wang, W.: Fast B-spline curve fitting by L-BFGS. Comput. Aided Geom. Des. **29**(7), 448–462 (2012)