

Local Surrogate Models with Reduced Dimensionality via Overlapping Domain Decomposition and Proper Generalized Decomposition

Marco Discacciati^[0000-0001-8343-8953], Ben J. Evans^[0009-0006-7246-1262], and Matteo Giacomini^[0000-0001-6094-5944]

1 Introduction

The combination of domain decomposition (DD) methods and model order reduction (MOR) techniques can lead to efficient algorithms to solve parametric partial differential equations for challenging applications such as, e.g., the construction of digital twins involving multi-physics models. Indeed, DD can split the target problem into local subproblems characterized by a single physics and by a reduced number of parameters and degrees of freedom that can be more easily handled by MOR techniques (see, e.g., [1, 5]).

In [4], we proposed a computational framework based on a substructured overlapping DD algorithm to couple local surrogate models generated by proper generalized decomposition (PGD). This is a physics-based a priori MOR method (see, e.g., [2]) that does not require any sampling of the parametric space and that can be very conveniently implemented in a completely non-intrusive way [3]. By exploiting the linearity of the underlying parametric partial differential equation, in the offline phase, we construct local surrogate PGD models that incorporate arbitrary traces of the solution at the subdomain interfaces, to be used in the online phase to perform alternating Schwarz iterations. However, the construction of such local surrogate models requires to suitably cluster the interface nodes, to appropriately define the parametric values of the traces, and to solve parametric subproblems with combinations of boundary conditions that could be not physically relevant.

Marco Discacciati, Ben J. Evans
Department of Mathematical Sciences, Loughborough University, Epinal Way, LE11 3TU, Loughborough, United Kingdom, e-mail: {m.discacciati, b.j.evans}@lboro.ac.uk

Matteo Giacomini
Laboratori de Càlcul Numèric (LaCàN), E.T.S. de Ingenieria de Caminos, Canales y Puertos, Universitat Politècnica de Catalunya, Barcelona, Spain, and Centre Internacional de Mètodes Numèrics en Enginyeria (CIMNE), Barcelona, Spain, e-mail: matteo.giacomini@upc.edu

For the purpose of open access, the authors have applied a Creative Commons Attribution (CC BY) licence to any Author Accepted Manuscript version arising from this submission.

To overcome these drawbacks, in this work, we improve the computational framework of [4] by reducing the parametric dimension of the offline local problems. This avoids the burden of the implementation of the clustering procedure, and it allows to compute local *PGD basis* functions that can be easily linearly combined during the online phase to obtain the local surrogate models with the interface trace needed by the Schwarz algorithm. This novel approach simplifies the overall implementation of the DD-PGD procedure and it results in significantly lower computational costs both offline and online.

After introducing the model parametric problem in Sect. 2, we present the novel approach to construct the local surrogate models in Sect. 3, and we show how to perform the online phase in Sect. 4. Finally, numerical results in Sect. 5 assess the performance of the algorithm compared to its counterpart in [4].

2 Problem formulation and PGD-accelerated Schwarz method

In this section, we formulate the alternating Schwarz method for a parametric model problem, and we explain how the algorithm can be split into an offline and an online phase to exploit the benefits of MOR. For the sake of simplicity, we consider a parametric Poisson problem with homogeneous Dirichlet boundary conditions, we assume that the domain is parameter-independent, and we introduce a domain decomposition into two overlapping subdomains only. The extension to more general linear elliptic operators with arbitrary boundary conditions and decompositions into multiple subdomains (without cross-points) can be found in [4].

Let $\Omega \subset \mathbb{R}^d$ ($d = 1, 2, 3$) be an open bounded domain with Lipschitz boundary $\partial\Omega$, and let $\boldsymbol{\mu} = (\mu^1, \dots, \mu^P) \in \mathcal{P}$ be a tuple of $P \in \mathbb{N}$ problem parameters with $\mathcal{P} = \mathcal{I}^1 \times \dots \times \mathcal{I}^P \subset \mathbb{R}^P$ with \mathcal{I}^p compact ($p = 1, \dots, P$). Moreover, for all $\boldsymbol{\mu} \in \mathcal{P}$, let $\nu(\boldsymbol{\mu}) > 0$ be a positive parametric diffusion coefficient, and $s(\boldsymbol{\mu}) \in L^2(\Omega)$ a parametric source term. Then, consider the well-posed linear elliptic parametric boundary value problem: for all $\boldsymbol{\mu} \in \mathcal{P}$, find $u(\boldsymbol{\mu})$ such that

$$\begin{aligned} -\nabla \cdot (\nu(\boldsymbol{\mu}) \nabla u(\boldsymbol{\mu})) &= s(\boldsymbol{\mu}) && \text{in } \Omega, \\ u(\boldsymbol{\mu}) &= 0 && \text{on } \partial\Omega. \end{aligned} \quad (1)$$

The domain Ω is split into two overlapping subdomains $\Omega_i \subset \Omega$ ($i = 1, 2$) such that $\Omega_1 \cup \Omega_2 = \Omega$ and $\Omega_1 \cap \Omega_2 \neq \emptyset$, and, for $i = 1, 2$, let $\Gamma_i = \partial\Omega_i \setminus \partial\Omega$. Considering this decomposition, problem (1) can be equivalently reformulated in the multi-domain form: for all $\boldsymbol{\mu} \in \mathcal{P}$, find $u_i(\boldsymbol{\mu})$ ($i = 1, 2$) such that

$$\begin{aligned} -\nabla \cdot (\nu(\boldsymbol{\mu}) \nabla u_i(\boldsymbol{\mu})) &= s_i(\boldsymbol{\mu}) && \text{in } \Omega_i, \\ u_i(\boldsymbol{\mu}) &= 0 && \text{on } \partial\Omega_i \cap \partial\Omega, \\ u_1(\boldsymbol{\mu}) &= u_2(\boldsymbol{\mu}) && \text{on } \Gamma_1 \cup \Gamma_2, \end{aligned} \quad (2)$$

where $s_i(\boldsymbol{\mu})$ denotes the restriction of $s(\boldsymbol{\mu})$ to Ω_i , and $(2)_3$ ensures the continuity of the local solutions $u_1(\boldsymbol{\mu})$ and $u_2(\boldsymbol{\mu})$ across the interfaces Γ_1 and Γ_2 .

2.1 Parametric overlapping Schwarz method

Using the multi-domain formulation (2), we can adapt the classical overlapping alternating Schwarz method to the case of a parametric problem. More precisely, let $\bar{\boldsymbol{\mu}} \in \mathcal{P}$ be a fixed set of parameters, and let $\lambda_1^{(0)}$ be a suitable trace on the interface Γ_1 . Then, for $k \geq 1$ until convergence when $\|u_1^{(k)}(\bar{\boldsymbol{\mu}}) - u_2^{(k)}(\bar{\boldsymbol{\mu}})\|_{\Gamma} < \text{tol}$, for a suitable tolerance tol and a suitable norm $\|\cdot\|_{\Gamma}$ on $\Gamma_1 \cup \Gamma_2$,

- Find $u_1^{(k)}(\bar{\boldsymbol{\mu}})$ such that

$$\begin{aligned} -\nabla \cdot (\nu(\bar{\boldsymbol{\mu}}) \nabla u_1^{(k)}(\bar{\boldsymbol{\mu}})) &= s_1(\bar{\boldsymbol{\mu}}) && \text{in } \Omega_1, \\ u_1^{(k)}(\bar{\boldsymbol{\mu}}) &= 0 && \text{on } \partial\Omega_1 \cap \partial\Omega, \\ u_1^{(k)}(\bar{\boldsymbol{\mu}}) &= \lambda_1^{(k-1)} && \text{on } \Gamma_1. \end{aligned} \quad (3)$$

- Set $\lambda_2^{(k)} = u_1^{(k)}(\bar{\boldsymbol{\mu}})|_{\Gamma_2}$.
- Find $u_2^{(k)}(\bar{\boldsymbol{\mu}})$ such that

$$\begin{aligned} -\nabla \cdot (\nu(\bar{\boldsymbol{\mu}}) \nabla u_2^{(k)}(\bar{\boldsymbol{\mu}})) &= s_2(\bar{\boldsymbol{\mu}}) && \text{in } \Omega_2, \\ u_2^{(k)}(\bar{\boldsymbol{\mu}}) &= 0 && \text{on } \partial\Omega_2 \cap \partial\Omega, \\ u_2^{(k)}(\bar{\boldsymbol{\mu}}) &= \lambda_2^{(k)} && \text{on } \Gamma_2. \end{aligned} \quad (4)$$

- Set $\lambda_1^{(k)} = u_2^{(k)}(\bar{\boldsymbol{\mu}})|_{\Gamma_1}$.

To efficiently obtain the solution of the parametric problem (1) for any set of parameters $\boldsymbol{\mu} \in \mathcal{P}$, the execution of the alternating Schwarz algorithm can be split into an offline phase and an online phase. In the former, PGD is used to precompute local surrogate models that can handle arbitrary Dirichlet boundary conditions at the interfaces, thus reducing the computational cost of the online Schwarz iterations.

More precisely, in the *offline* phase, the local parametric problems (3) and (4) are solved by PGD so that, at the end of this phase, local surrogate models u_i^{PGD} are available. The local surrogate models u_i^{PGD} feature arbitrary traces λ_i at the interfaces, and this is achieved by parametrizing them through a set of auxiliary parameters, say $\boldsymbol{\Lambda}_i$. As we will see in Sect. 3, the linearity of the local problems is exploited to avoid incurring the curse of dimensionality after introducing such additional parameters.

In the *online* phase, the alternating Schwarz iterations are performed for a fixed set of parametric values $\bar{\boldsymbol{\mu}} \in \mathcal{P}$, and the solution of the local problems (3) and (4) is replaced by the *evaluation* of the precomputed local surrogate models u_i^{PGD} at specific instances of the interface parameters $\boldsymbol{\Lambda}_i$. Since no problems are solved at this stage, the Schwarz algorithm can be executed in real time. Details are provided in Sect. 4.

3 Offline phase: Computation of the local surrogate models

In the offline phase of the algorithm, PGD is used to solve the following local parametric problem: for all $\boldsymbol{\mu} \in \mathcal{P}$, find $u_i(\boldsymbol{\mu})$ ($i = 1, 2$) such that

$$\begin{aligned}
-\nabla \cdot (\nu(\boldsymbol{\mu}) \nabla u_i(\boldsymbol{\mu})) &= s_i(\boldsymbol{\mu}) && \text{in } \Omega_i, \\
u_i(\boldsymbol{\mu}) &= 0 && \text{on } \partial\Omega_i \cap \partial\Omega, \\
u_i(\boldsymbol{\mu}) &= \lambda_i && \text{on } \Gamma_i,
\end{aligned} \tag{5}$$

where λ_i is a space-dependent function that represents an arbitrary Dirichlet boundary condition at the interface Γ_i . A suitable parametrization is introduced to handle the arbitrariness of the trace λ_i . More precisely, considering that (5) is solved, e.g., by the finite element method, we express λ_i as a linear combination of suitable finite element basis functions on Γ_i :

$$\lambda_i(\mathbf{x}) = \sum_{q=1}^{N_{\Gamma_i}} \Lambda_i^q \eta_i^q(\mathbf{x}). \tag{6}$$

Here, N_{Γ_i} represents the dimension of the discrete trace space generated by the basis functions $\eta_i^q(\mathbf{x})$ on the interface Γ_i , while $\boldsymbol{\Lambda}_i = (\Lambda_i^1, \dots, \Lambda_i^{N_{\Gamma_i}}) \in \mathbb{R}^{N_{\Gamma_i}}$ are the coefficients of the linear combination. Following [4], we can choose $\eta_i^q(\mathbf{x})$ to be the non-null restriction on Γ_i of the standard piecewise Lagrange polynomial basis functions, say $\varphi_i^q(\mathbf{x})$, that discretize (5) in Ω_i , that is, $\eta_i^q(\mathbf{x}) = \varphi_i^q(\mathbf{x})|_{\Gamma_i}$. Therefore, the basis functions $\eta_i^q(\mathbf{x})$ are linearly independent, they provide a partition of unity on Γ_i , and they satisfy $\eta_i^q(\mathbf{x}_i^m) = \delta_{qm}$ for all nodes \mathbf{x}_i^m on the interface Γ_i , δ_{qm} being the Kronecker delta. (Although this choice of the basis functions on Γ_i is very convenient for implementation purposes, other choices are possible and they can be accommodated in the overall structure of the algorithm.)

The coefficients Λ_i^q could be considered as additional parameters of the local problem (5), and the latter be solved by PGD. However, in general we expect $N_{\Gamma_i} \gg 1$, and it is well-known that a limitation of MOR methods is that they cannot efficiently handle a large number of parameters. To overcome this difficulty, we exploit the linearity of the operator that governs the boundary value problem and we introduce the following $N_{\Gamma_i} + 1$ problems:

- for all $\boldsymbol{\mu} \in \mathcal{P}$, find $u_{i,0}(\boldsymbol{\mu})$ such that

$$\begin{aligned}
-\nabla \cdot (\nu(\boldsymbol{\mu}) \nabla u_{i,0}(\boldsymbol{\mu})) &= s_i(\boldsymbol{\mu}) && \text{in } \Omega_i, \\
u_{i,0}(\boldsymbol{\mu}) &= 0 && \text{on } \partial\Omega_i \cap \partial\Omega, \\
u_{i,0}(\boldsymbol{\mu}) &= 0 && \text{on } \Gamma_i;
\end{aligned} \tag{7a}$$

- for $q = 1, \dots, N_{\Gamma_i}$ and for all $\boldsymbol{\mu} \in \mathcal{P}$, find $u_{i,q}(\boldsymbol{\mu})$ such that

$$\begin{aligned}
-\nabla \cdot (\nu(\boldsymbol{\mu}) \nabla u_{i,q}(\boldsymbol{\mu})) &= 0 && \text{in } \Omega_i, \\
u_{i,q}(\boldsymbol{\mu}) &= 0 && \text{on } \partial\Omega_i \cap \partial\Omega, \\
u_{i,q}(\boldsymbol{\mu}) &= \eta_i^q && \text{on } \Gamma_i.
\end{aligned} \tag{7b}$$

A schematic representation of this idea is presented in Figure 1. Problem (7a) and all problems (7b) can be solved independently, and, being associated with the linearly independent trace functions η_i^q , the solutions $\{u_{i,q}(\boldsymbol{\mu})\}_{q=1, \dots, N_{\Gamma_i}}$ of (7b) can be considered as a *PGD basis* that can be used to construct the local surrogate model

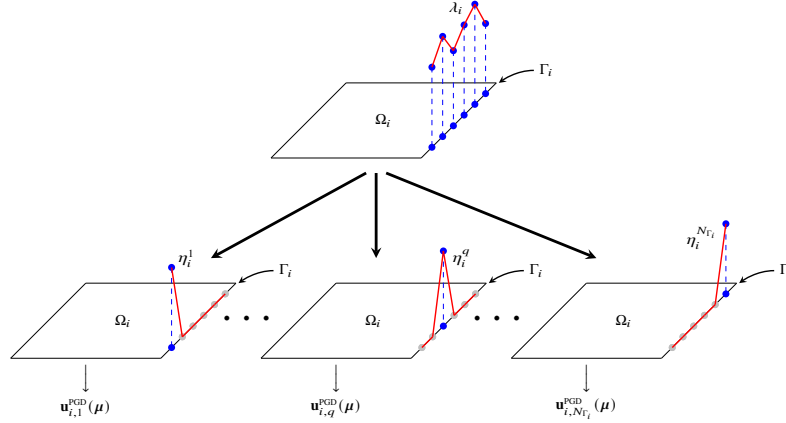


Fig. 1 Partition of the interface nodes as a collection of single independent boundary parameters.

in Ω_i . More precisely, for all $\boldsymbol{\mu} \in \mathcal{P}$ and for a given trace function λ_i on Γ_i , upon considering the expansion (6), the solution of (5) can be obtained as

$$u_i(\boldsymbol{\mu}, \boldsymbol{\Lambda}_i) = u_{i,0}(\boldsymbol{\mu}) + \sum_{q=1}^{N_{\Gamma_i}} \Lambda_i^q u_{i,q}(\boldsymbol{\mu}). \quad (8)$$

This approach is different from the one proposed in [4] and it presents several computational advantages, both in the offline and in the online phase. We focus here on the offline phase and we will discuss the online phase in Sect. 4. In [4], to overcome the difficulty of handling too many interface parameters Λ_i^q , these were clustered in N_i sufficiently small disjoint sets \mathcal{N}_i^j ($j = 1, \dots, N_i$) of *active boundary parameters* with $\bigcup_{j=1, \dots, N_i} \mathcal{N}_i^j = \{1, \dots, N_{\Gamma_i}\}$ and with $\text{card}(\mathcal{N}_i^j) \ll N_{\Gamma_i}$. Then, (6) was replaced by

$$\lambda_i = \sum_{q \in \mathcal{N}_i^1} \Lambda_i^q \eta_i^q + \sum_{q \in \mathcal{N}_i^2} \Lambda_i^q \eta_i^q + \dots + \sum_{q \in \mathcal{N}_i^{N_i}} \Lambda_i^q \eta_i^q, \quad (9)$$

and, instead of (7b), one would solve the N_i independent problems: for $j = 1, \dots, N_i$, for all $\boldsymbol{\mu} \in \mathcal{P}$ and for all $\boldsymbol{\Lambda}_i^j = (\Lambda_i^q)_{q \in \mathcal{N}_i^j} \in \mathcal{Q}_i^j$, find

$$\begin{aligned} -\nabla \cdot (\nu(\boldsymbol{\mu}) \nabla u_{i,j}(\boldsymbol{\mu}, \boldsymbol{\Lambda}_i^j)) &= 0 && \text{in } \Omega_i, \\ u_{i,j}(\boldsymbol{\mu}, \boldsymbol{\Lambda}_i^j) &= 0 && \text{on } \partial\Omega_i \cap \partial\Omega, \\ u_{i,j}(\boldsymbol{\mu}, \boldsymbol{\Lambda}_i^j) &= \sum_{q \in \mathcal{N}_i^j} \Lambda_i^q \eta_i^q && \text{on } \Gamma_i. \end{aligned} \quad (10)$$

Here, $\mathcal{Q}_i^j = \times_{q \in \mathcal{N}_i^j} \mathcal{J}_i^q$ is the space of boundary parameters with each $\mathcal{J}_i^q \subset \mathbb{R}$ ($q = 1, \dots, N_{\Gamma_i}$) a compact set. Finally, the solution of (5) is obtained as

$$u_i(\boldsymbol{\mu}, \Lambda_i) = u_{i,0}(\boldsymbol{\mu}) + \sum_{j=1}^{N_i} u_{i,j}(\boldsymbol{\mu}, \Lambda_i^j). \quad (11)$$

In the case of clustered interface nodes, one must solve a smaller number $N_i \ll N_{\Gamma_i}$ of local problems (10), each depending both on $\boldsymbol{\mu}$ and on the interface parameters Λ_i^j , while (7b) only involves the problem parameter $\boldsymbol{\mu}$. Hence, the proposed approach leads to local problems with significantly reduced dimensionality, at the expense of computing N_{Γ_i} local surrogate models instead of N_i . Nonetheless, it is worth recalling that the local problems (7b) are independent from one another and can be easily solved in parallel. Moreover, before solving (10), a suitable interface parametric space \mathcal{Q}_i^j must be identified to represent an arbitrary trace that is non-zero at the nodes corresponding to the indices in \mathcal{N}_i^j . In principle, each intervals \mathcal{J}_i^q depends on $\boldsymbol{\mu}$ and it should be defined considering physical information about the solution of problem (1) (e.g., the maximum and minimum values that $u(\boldsymbol{\mu})$ can attain within Ω). In the approach proposed in this work, this is not needed, as the auxiliary problems (7b) are independent of Λ_i^j . In particular, it should be noted that not all possible combinations of the parameters Λ_i^q considered in (10) may actually be significant for the problem at hand. For instance, considering very different values of Λ_i^q at adjacent nodes on Γ_i can represent a highly oscillatory trace function, that it is unlikely to be relevant for the solution of an elliptic problem in a smooth domain with sufficiently regular data. Moreover, solving boundary value problems with such localised features can be challenging for PGD due to the possibly large number of modes needed to correctly represent the behaviour of the solution. Therefore, the offline computational cost associated with the clustering procedure can become unnecessarily high, as numerical simulations will show in Sect. 5. Finally, no clustering procedure is needed when working with (7a) and (7b), thus making the implementation simpler.

3.1 Construction of the local surrogate models using PGD

In this section, we outline the procedure to compute the solutions of the local problems (7a) and (7b) using PGD, and we refer to [4] for more details.

Following the standard approach in PGD [2], we assume that all data are given in separated form as the sum of products of functions that depend either on the spatial coordinate \boldsymbol{x} or on the parameters $\boldsymbol{\mu}$. The contribution of the Dirichlet boundary condition in (7b) is handled by introducing ad-hoc, sufficiently smooth modes as usually done in the PGD context. Then, the solution of (7a) and the solution, say $v_{i,q}(\boldsymbol{\mu})$, of (7b), after accounting for the Dirichlet boundary condition on Γ_i , are written in the PGD expansion

$$u_{i,0}(\boldsymbol{\mu}) \approx u_{i,0}^{\text{PGD}}(\boldsymbol{\mu}) = \sum_{m=1}^{M_0} V_{i,0}^m(\boldsymbol{x}) \phi_{i,0}^m(\boldsymbol{\mu}), \quad v_{i,q}(\boldsymbol{\mu}) \approx v_{i,q}^{\text{PGD}}(\boldsymbol{\mu}) = \sum_{m=1}^{M_q} V_{i,q}^m(\boldsymbol{x}) \phi_{i,q}^m(\boldsymbol{\mu}),$$

with $V_{i,0}^m$ and $V_{i,j}^m$ being the m -th spatial modes, and $\phi_{i,0}^m(\boldsymbol{\mu})$ and $\phi_{i,q}^m(\boldsymbol{\mu})$ the parametric modes. The spatial modes are discretized using continuous Galerkin finite

elements, while pointwise collocation is used to approximate the parametric modes. The number of modes M_0 and M_q is automatically determined by a greedy procedure. Since all problems (7a) and (7b) are independent, we compute their PGD approximations separately using the encapsulated PGD library [3]. At the end of this procedure, we obtain the discrete PGD approximations $\mathbf{u}_{i,0}^{\text{PGD}}(\boldsymbol{\mu})$ and $\mathbf{u}_{i,q}^{\text{PGD}}(\boldsymbol{\mu})$ of the solutions of problems (7a) and (7b), respectively.

4 Online phase: PGD-accelerated Schwarz iterations

In this section, we present an efficient algorithm to construct the global solution of (1) for fixed values $\bar{\boldsymbol{\mu}} \in \mathcal{P}$ of the physical parameter.

First of all, considering a suitable conforming finite element approximations of the local problems (3) and (4), we recall (see, e.g., [8]) that one step of the overlapping Schwarz method outlined in Sect. 2.1 corresponds to one Gauss-Seidel iteration to solve the linear system

$$\begin{pmatrix} A_{\Omega_1} & A_{\Gamma_1} & 0 & 0 \\ 0 & I_{\Gamma_1} & -\mathcal{I}_{\Omega_2 \rightarrow \Gamma_1} & 0 \\ 0 & 0 & A_{\Omega_2} & A_{\Gamma_2} \\ -\mathcal{I}_{\Omega_1 \rightarrow \Gamma_2} & 0 & 0 & I_{\Gamma_2} \end{pmatrix} \begin{pmatrix} \mathbf{u}_{\Omega_1}(\bar{\boldsymbol{\mu}}) \\ \mathbf{u}_{\Gamma_1}(\bar{\boldsymbol{\mu}}) \\ \mathbf{u}_{\Omega_2}(\bar{\boldsymbol{\mu}}) \\ \mathbf{u}_{\Gamma_2}(\bar{\boldsymbol{\mu}}) \end{pmatrix} = \begin{pmatrix} \mathbf{f}_{\Omega_1}(\bar{\boldsymbol{\mu}}) \\ \mathbf{0} \\ \mathbf{f}_{\Omega_2}(\bar{\boldsymbol{\mu}}) \\ \mathbf{0} \end{pmatrix}. \quad (12)$$

Here, $\mathbf{u}_{\Omega_i}(\bar{\boldsymbol{\mu}})$ and $\mathbf{u}_{\Gamma_i}(\bar{\boldsymbol{\mu}})$ denote the vectors of nodal values of the unknown solutions inside the domain Ω_i and on Γ_i , respectively, whereas I_{Γ_i} represents the identity matrix on Γ_i and $\mathcal{I}_{\Omega_j \rightarrow \Gamma_i}$ is the interpolation operator between the nodes in Ω_j and those on Γ_i ($i, j = 1, 2, i \neq j$). System (12) can be equivalently reformulated only in terms of the interface unknowns $\mathbf{u}_{\Gamma_1}(\bar{\boldsymbol{\mu}})$ and $\mathbf{u}_{\Gamma_2}(\bar{\boldsymbol{\mu}})$ as the interface system

$$\begin{pmatrix} I_{\Gamma_1} & \mathcal{I}_{\Omega_2 \rightarrow \Gamma_1} A_{\Omega_2}^{-1} A_{\Gamma_2} \\ \mathcal{I}_{\Omega_1 \rightarrow \Gamma_2} A_{\Omega_1}^{-1} A_{\Gamma_1} & I_{\Gamma_2} \end{pmatrix} \begin{pmatrix} \mathbf{u}_{\Gamma_1}(\bar{\boldsymbol{\mu}}) \\ \mathbf{u}_{\Gamma_2}(\bar{\boldsymbol{\mu}}) \end{pmatrix} = \begin{pmatrix} \mathcal{I}_{\Omega_2 \rightarrow \Gamma_1} A_{\Omega_2}^{-1} \mathbf{f}_{\Omega_2}(\bar{\boldsymbol{\mu}}) \\ \mathcal{I}_{\Omega_1 \rightarrow \Gamma_2} A_{\Omega_1}^{-1} \mathbf{f}_{\Omega_1}(\bar{\boldsymbol{\mu}}) \end{pmatrix}, \quad (13)$$

which can be solved using a suitable matrix-free Krylov method. The expensive part of this iterative algorithm is the solution of the local problems in Ω_i (corresponding to $A_{\Omega_i}^{-1}$). This can become especially demanding when a new set of parameters $\boldsymbol{\mu}$ must be considered because, in general, the matrices A_{Ω_i} and A_{Γ_i} can depend on $\boldsymbol{\mu}$, so that the entire procedure should be re-executed from scratch.

To reduce the computational cost of the coupling procedure (13), we exploit the PGD local surrogate models computed in the offline phase. Indeed, we notice that the vector $\mathbf{u}_{\Gamma_i}(\bar{\boldsymbol{\mu}})$ corresponds to the vector of parameters $\boldsymbol{\Lambda}_i$ in (6), so that the i -th equation of the linear system (13) can be rewritten as

$$I_{\Gamma_i} \boldsymbol{\Lambda}_i = \mathcal{I}_{\Omega_j \rightarrow \Gamma_i} \left(A_{\Omega_j}^{-1} \mathbf{f}_{\Omega_j}(\bar{\boldsymbol{\mu}}) + A_{\Omega_j}^{-1} (-A_{\Gamma_j} \boldsymbol{\Lambda}_j) \right). \quad (14)$$

Here, $A_{\Omega_j}^{-1} \mathbf{f}_{\Omega_j}(\bar{\boldsymbol{\mu}})$ corresponds to the vector of the nodal values of the PGD solution of problem (7a), while $A_{\Omega_j}^{-1} (-A_{\Gamma_j} \boldsymbol{\Lambda}_j)$ corresponds to the linear combination, with

coefficient Λ_j , of the vectors of the nodal values of the PGD solutions of problem (7b). All vectors are evaluated for the target parametric values $\bar{\boldsymbol{\mu}} \in \mathcal{P}$. Therefore, using the notation introduced in Sect. 3.1, we can equivalently write

$$I_{\Gamma_i} \boldsymbol{\Lambda}_i = \mathcal{I}_{\Omega_j \rightarrow \Gamma_i} \left(\mathbf{u}_{j,0}^{\text{PGD}}(\bar{\boldsymbol{\mu}}) + \sum_{q=1}^{N_{\Gamma_j}} \Lambda_j^q \mathbf{u}_{j,q}^{\text{PGD}}(\bar{\boldsymbol{\mu}}) \right). \quad (15)$$

For $j = 1, 2$, we can define the local PGD operator

$$\mathcal{A}_j^{\text{PGD}} : \boldsymbol{\Lambda}_j \rightarrow \sum_{q=1}^{N_{\Gamma_j}} \Lambda_j^q \mathbf{u}_{j,q}^{\text{PGD}}(\bar{\boldsymbol{\mu}}) \quad (16)$$

so that, equation (15) and the interface system (13) can respectively be rewritten in the compact forms

$$I_{\Gamma_i} \boldsymbol{\Lambda}_i = \mathcal{I}_{\Omega_j \rightarrow \Gamma_i} \left(\mathbf{u}_{j,0}^{\text{PGD}}(\bar{\boldsymbol{\mu}}) + \mathcal{A}_j^{\text{PGD}} \boldsymbol{\Lambda}_j \right), \quad (17)$$

and

$$\begin{pmatrix} I_{\Gamma_1} & -\mathcal{I}_{\Omega_2 \rightarrow \Gamma_1} \mathcal{A}_2^{\text{PGD}} \\ -\mathcal{I}_{\Omega_1 \rightarrow \Gamma_2} \mathcal{A}_1^{\text{PGD}} & I_{\Gamma_2} \end{pmatrix} \begin{pmatrix} \boldsymbol{\Lambda}_1 \\ \boldsymbol{\Lambda}_2 \end{pmatrix} = \begin{pmatrix} \mathcal{I}_{\Omega_2 \rightarrow \Gamma_1} \mathbf{u}_{2,0}^{\text{PGD}}(\bar{\boldsymbol{\mu}}) \\ \mathcal{I}_{\Omega_1 \rightarrow \Gamma_2} \mathbf{u}_{1,0}^{\text{PGD}}(\bar{\boldsymbol{\mu}}) \end{pmatrix}. \quad (18)$$

Therefore, the online phase of the PGD-enhanced Schwarz method consists in solving (18) by an iterative matrix-free method, e.g., GMRES. The computational cost of each GMRES iteration is mainly due to performing the linear combination (16) ($j = 1, 2$). At convergence, say, at iteration $k = k^*$, the approximation of the solution of the global problem (1) for $\bar{\boldsymbol{\mu}} \in \mathcal{P}$ is given by

$$\mathbf{u}^{\text{PGD}}(\bar{\boldsymbol{\mu}}) = \begin{cases} \mathbf{u}_{1,0}^{\text{PGD}}(\bar{\boldsymbol{\mu}}) + \sum_{q=1}^{N_{\Gamma_1}} (\Lambda_1^q)^{(k^*)} \mathbf{u}_{1,q}^{\text{PGD}}(\bar{\boldsymbol{\mu}}) & \text{in } \Omega_1, \\ \mathbf{u}_{2,0}^{\text{PGD}}(\bar{\boldsymbol{\mu}}) + \sum_{q=1}^{N_{\Gamma_2}} (\Lambda_2^q)^{(k^*)} \mathbf{u}_{2,q}^{\text{PGD}}(\bar{\boldsymbol{\mu}}) & \text{in } \Omega_2 \setminus \Omega_{12}. \end{cases} \quad (19)$$

The approach presented here for the online phase is more efficient than the one based on the clustering of the nodes in the offline phase, see Sect. 3 and [4]. Indeed, in the latter case, the term $A_{\Omega_j}^{-1}(-A_{\Gamma_j} \boldsymbol{\Lambda}_j)$ corresponds to the vector $\mathbf{u}_{j,\Lambda}^{\text{PGD}}(\boldsymbol{\mu}, \boldsymbol{\Lambda}_j)$ of the nodal values of the PGD solution of problem (10). Therefore, (15) becomes

$$I_{\Gamma_i} \boldsymbol{\Lambda}_i = \mathcal{I}_{\Omega_j \rightarrow \Gamma_i} \left(\mathbf{u}_{j,0}^{\text{PGD}}(\bar{\boldsymbol{\mu}}) + \mathbf{u}_{j,\Lambda}^{\text{PGD}}(\boldsymbol{\mu}, \boldsymbol{\Lambda}_j) \right), \quad (20)$$

and the operator $\mathcal{A}_j^{\text{PGD}}$ must be replaced by $A_{j,\Lambda_j}^{\text{PGD}} : \boldsymbol{\Lambda}_j \rightarrow \mathbf{u}_{j,\Lambda}^{\text{PGD}}(\bar{\boldsymbol{\mu}}, \boldsymbol{\Lambda}_j)$, which returns the nodal values of the PGD model that satisfies problem (10) for the boundary parameters $\boldsymbol{\Lambda}_j$ and for $\bar{\boldsymbol{\mu}}$. Since the values $\Lambda_j^{(k)}$ computed at the k -th GMRES iteration do not generally coincide with those obtained when discretizing the parametric space \mathcal{Q}_j , applying the local operator $A_{j,\Lambda_j}^{\text{PGD}}$ requires linearly interpolating the parametric modes that depend on $\boldsymbol{\Lambda}_j$ and that are associated with the available values closest to

$\Lambda_j^{(k)}$. Although this operation is computationally inexpensive, its repeated execution involves a higher computational cost than the linear combination (16).

Moreover, for the approach based on clustering the interface nodes, the parametric domain \mathcal{Q}_j is fixed during the offline phase and, in principle, it could happen that some of the values $\Lambda_j^{(k)}$ computed by GMRES fall outside of the parametric space \mathcal{Q}_j , thus jeopardizing the convergence of the iterative scheme. This cannot occur using the approach described in Sect. 3. Indeed, it is always possible to compute the linear combination (16) for any given coefficients $\Lambda_j^{(k)}$ once the *PGD basis functions* $\mathbf{u}_{j,q}^{\text{PGD}}(\bar{\boldsymbol{\mu}})$ are available. The resulting online coupling algorithm is thus computationally less expensive and more robust.

5 Numerical results

In this section, we reproduce the tests studied in Sects. 5.1 and 5.3 of [4] using the approach explained in this paper to assess its performance versus the technique based on clustering the boundary nodes. Due to the restrictions on the number of pages, we outline the definition of the problems and refer to [4] for a detailed description. For all tests, in the offline phase, the local parametric problems (7a) and (7b) are solved using the encapsulated PGD toolbox [3] with tolerance 10^{-4} to stop the PGD enrichment process, and a compression algorithm [7] with tolerance 10^{-3} is applied to eliminate redundant modes. In the online phase, the interface system (18) is solved by GMRES with stopping tolerance 10^{-6} on the relative residual.

5.1 Bidomain diffusion problem with analytical solution

Consider problem (1) with $v(\mu) = 1 + \mu x$, $\mathbf{x} = (x, y) \in \Omega$ and $\mu \in \mathcal{P} := [1, 50]$ being a scalar parameter. The domain is $\Omega = (0, 2) \times (0, 1)$, split into the two overlapping subdomains $\Omega_1 = (0, 1.05) \times (0, 1)$ and $\Omega_2 = (0.95, 2) \times (0, 1)$, and the source term $s(\mu)$ is chosen such that the exact solution is $u(\mu) = \sin(2\pi x) \sin(2\pi y) + \frac{\mu}{2} xy(y-1)(x-2)$. The parametric interval \mathcal{P} is discretized using step $h_\mu = 10^{-3}$, while Lagrangian \mathbb{Q}_1 elements are used for the space discretization with mesh size $h = 0.05$. The local meshes coincide in the overlap that has width $2h$. Considering this space discretization, there are 19 nodes on each interface so that, following the procedure in Sect. 3, one must solve 19 problems like (7b) and one problem (7a) in each subdomain. The total number of PGD modes computed in Ω_1 is 68 (106 before compression) and 56 (62 before compression) in Ω_2 , with total CPU time of the offline phase of 6.56 s (averaged over 10 runs). In the online phase for the case $\mu = 3$, GMRES converges in 9 iterations with computational time of about 2×10^{-2} s. To assess the accuracy of the DD-PGD approach, we compute the relative error in L^2 norm between the exact solution and the DD-PGD solution for selected values of the parameter μ and we compare them to the errors for the full-order finite element solution u_Ω^h . The results reported in Table 1 show that the accuracy provided by the two coupling approaches is identical. Comparing these results with those in Sect. 5.1 of [4], we remark that the offline computational cost is now reduced by approximately 17 times without loss of ac-

μ	$\ u^{\text{PGD}} - u\ _{L^2(\Omega)} / \ u\ _{L^2(\Omega)}$	$\ u_{\Omega}^h - u\ _{L^2(\Omega)} / \ u\ _{L^2(\Omega)}$
3	9.08×10^{-3}	9.07×10^{-3}
30	3.27×10^{-3}	3.27×10^{-3}

Table 1 Relative error in L^2 norm between the exact solution u , the DD-PGD solution u^{PGD} and the full-order finite element solution u_{Ω}^h .

curacy in the online phase. This is due to the fact that many unneeded parametric combinations occurring in (10) are avoided with the present approach. The number of modes is also lower, thus reducing memory storage. Meanwhile, the online phase is twice as fast and requires fewer iterations with both advantages being attributable to the absence of an interpolation procedure for the boundary parameters.

5.2 Multi-domain diffusion problem

We consider the diffusion equation (1)₁ on the domain Ω shown in Fig. 2 (left) with boundary conditions $u(\boldsymbol{\mu}) = 0$ on Γ_{out} , $\nu(\boldsymbol{\mu})\nabla u(\boldsymbol{\mu}) \cdot \mathbf{n} = 1$ on Γ_{in} and $\nabla u(\boldsymbol{\mu}) \cdot \mathbf{n} = 0$ on $\partial\Omega \setminus (\Gamma_{\text{in}} \cup \Gamma_{\text{out}})$. In this test case, proposed in [6], each subdomain Ω_i^b is characterized by a parametric diffusion coefficient $\nu(\boldsymbol{\mu}) = \mu_i$ ($i = 1, \dots, 9$), while $\nu(\boldsymbol{\mu}) = 1$ in the rest of the domain, so that there are 9 scalar parameters $\mu_i \in \mathcal{P} := [5 \times 10^{-2}, 10]$. As we remarked in [4], the computational domain can be reconstructed by composing four reference subdomains $\hat{\Omega}_i$ ($i = 1, \dots, 4$) upon suitable rotation and/or translation. Therefore, in the offline phase, the local surrogate models are computed only in those reference domains considering discretization step $h_{\mu} = 10^{-3}$ for \mathcal{P} , and \mathbb{Q}_1 Lagrangian elements for the finite element discretization with mesh size $h = 0.0125$ along the external boundary of each subdomain and $h = 0.05$ within the bulk regions Ω_i^b . This space discretization leads to 42 boundary problems (7b) in $\hat{\Omega}_1$ and in $\hat{\Omega}_4$, 63 problems in $\hat{\Omega}_2$, and 84 in $\hat{\Omega}_3$. In $\hat{\Omega}_4$, an additional problem (7a) is considered to account for the Neumann boundary condition on Γ_{in} . The number of PGD modes computed in the offline phase in $\hat{\Omega}_i$ ($i = 1, \dots, 4$) is, respectively, 190 (312 before compression), 272 (469), 336 (604) and 195 (321), with a total computational cost of approximately 35 s. These are significantly fewer modes than those obtained by the clustering procedure that resulted, respectively, in 236 (370 before compression), 386 (672), 509 (930) and 244 (419) modes with computational cost approximately 25 times higher. The number of GMRES iterations and the CPU time obtained for two sets of parameters $\boldsymbol{\mu}$ in the online phase are reported in Table 2, together with the relative errors in l^{∞} norm versus a full-order finite element solution $u_{\Omega}^h(\boldsymbol{\mu})$. The error distribution for the first set of parameters is also shown in Fig. 2 (right).

	μ_1	μ_2	μ_3	μ_4	μ_5	μ_6	μ_7	μ_8	μ_9	# iter.	CPU time	$\frac{\ u^{\text{PGD}}(\boldsymbol{\mu}) - u^h(\boldsymbol{\mu})\ _{l^{\infty}(\Omega)}}{\ u^h(\boldsymbol{\mu})\ _{l^{\infty}(\Omega)}}$
Test 1:	0.1	0.2	0.4	0.8	1.6	3.2	6.4	0.1	0.2	93	1.26 s	1.8×10^{-3}
Test 2:	4.9	4.7	4.8	5.2	5.0	4.9	5.5	5.3	5.1	57	0.79 s	3.4×10^{-4}

Table 2 Number of GMRES iterations, CPU time and relative errors obtained in the online phase for two sets of parametric values.

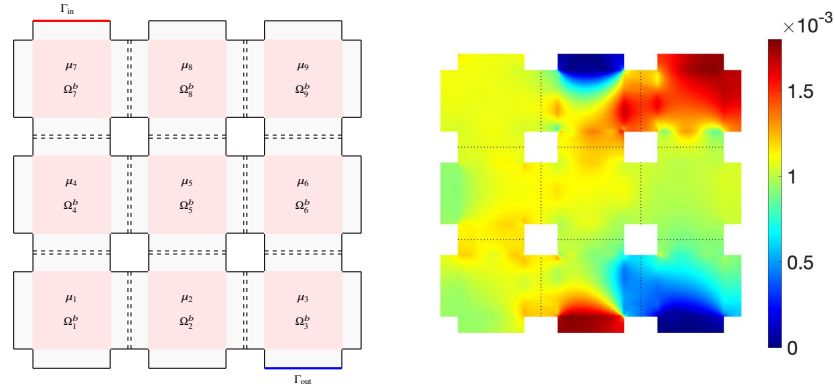


Fig. 2 Computational domain (left) and scaled error $|u^{PGD}(\mu) - u^h(\mu)| / \max_{\Omega} |u^h(\mu)|$ for the first set of parameters reported in Table 2.

We remark that the number of GMRES iterations is significantly lower than in the case of clustering of the nodes where 302 and 126 iterations were needed to converge in Test 1 and 2, respectively. The CPU time is also reduced by approximately 15 and 10 times. It is also worth noticing that the number of performed GMRES iterations (93 and 57, respectively) is now comparable to the 95 and 56 iterations required by the overlapping Schwarz algorithm coupled with the high-fidelity finite element solver, while the proposed strategy offers a wall-clock speed-up of approximately 130 times. The fewer iterations are likely due to the avoidance of the interpolation procedure that was needed by the previous clustering approach.

6 Conclusions

In this work, we presented an improved algorithm that combines overlapping DD and physics-based PGD MOR to solve parametric linear elliptic problems. The proposed approach exploits the basis functions that span the space of the traces of the finite element solution on the domain interfaces to construct a *PGD basis* for the local surrogate models, with a significant reduction of the dimensionality of the parametric subproblems to be solved. These are then efficiently added through parametric linear combinations to obtain the local surrogate models needed at each iteration of the Schwarz algorithm in the online phase. The resulting DD-PGD algorithm outperforms the approach in [4] in terms of simplicity of implementation and of computational cost both in the online and in the offline phase.

Acknowledgements MD acknowledges EPSRC grant EP/V027603/1, BJE the EPSRC Doctoral Training Partnership grant EP/W523987/1, and MG the Spanish Ministry of Science, Innovation and Universities and State Research Agency MICIU/AEI/10.13039/501100011033 (Grants PID2020-113463RB-C33, CEX2018-000797-S) and Generalitat de Catalunya (Grant 2021-SGR-01049). MG is Fellow of the Serra Hünter Programme of the Generalitat de Catalunya.

Competing Interests The authors have no conflicts of interest to declare that are relevant to the content of this chapter.

References

1. Buhr, A., Iapichino, L., Ohlberger, M., Rave, S., Schindler, F., Smetana, K.: Localized model reduction for parameterized problems. In: Benner, P., et al. (eds.) *Snapshot-Based Methods and Algorithms (Volume 2)*, pp. 245–306, De Gruyter (2021).
2. Chinesta, F., Keunings, R., Leygue, A.: *The Proper Generalized Decomposition for Advanced Numerical Simulations*. Springer (2014).
3. Díez, P., Zlotnik, S., García-González, A., Huerta, A.: Encapsulated PGD algebraic toolbox operating with high-dimensional data. *Arch. Comput. Methods Eng.* **27**, 1321–1336 (2020).
4. Discacciati, M., Evans, B.J., Giacomini, M.: An overlapping domain decomposition method for the solution of parametric elliptic problems via proper generalized decomposition. *Comput. Methods Appl. Mech. Engrg.* **418**, 116484 (2024).
5. Discacciati, M., Evans, B.J., Giacomini, M.: An overlapping domain decomposition method for the Stokes and Stokes-Darcy problems via PGD. [arXiv:2507.06861](https://arxiv.org/abs/2507.06861) (2025).
6. Erftang, J.L., Patera, A.T.: Port reduction in parametrized component static condensation and approximation and a posteriori error estimation. *Internat. J. Numer. Methods Engrg.* **96**(5), 269–302 (2013).
7. Modesto, D., Zlotnik, S., Huerta, A.: Proper generalized decomposition for parametrized Helmholtz problems in heterogeneous and unbounded domains: Application to harbor agitation. *Comput. Methods Appl. Mech. Engrg.* **295**, 127–149 (2015).
8. Smith, B., Bjørstad, P., Gropp, W.: *Domain Decomposition: Parallel Multilevel Methods for Elliptic Partial Differential Equations*. Cambridge University Press (1996).