Learning Adaptive FETI-DP Constraints for Irregular Domain Decompositions

Axel Klawonn, Martin Lanser, and Janine Weber

1 Introduction

Adaptive, that is, problem-dependent coarse spaces provide a robust condition number estimate and thus a robust convergence behavior for FETI-DP (Finite Element Tearing and Interconnecting - Dual Primal) and BDDC (Balancing Domain Decomposition by Constraints) methods for highly heterogeneous model problems; see, e.g., [7, 10] for a condition number indicator and a related proof for a specific adaptive coarse space in two spatial dimensions. In general, the setup of an adaptive coarse space usually requires the solution of local eigenvalue problems on edges, faces, or local parts of the domain decomposition interface. Even though the setup and the solution of these eigenvalue problems can be parallelized in a parallel implementation, it can take up the largest part of the overall time to solution, especially for three-dimensional problems. Thus, in [2], we have proposed to train a supervised classification model in form of a dense feedforward neural network to make an a priori decision, which of the eigenvalue problems are actually necessary for a robust FETI-DP coarse space. By testing our approach for different realistic heterogeneous model problems as, e.g., arising from a dual-phase steel in solid mechanics, we have shown that it is possible to drastically reduce the number of necessary eigenvalue problems while still maintaining the robustness of the iterative solver.

In [6], we have extended these results by directly learning the adaptive edge constraints themselves. Hence, we have trained different regression neural network models to compute an a priori approximation of the first $k \in \mathbb{N}$ adaptive edge constraints, which are then used to enhance the classic FETI-DP method. In particular, this approach does not require the setup or the solution of any eigenvalue problems at all. In [6], we have trained the regression neural network models exclusively with

Axel Klawonn, Martin Lanser, and Janine Weber
Department of Mathematics and Computer Science, University of Cologne, Weyertal 86-90, 50931 Köln, Germany, e-mail: axel.klawonn@uni-koeln.de, martin.lanser@uni-koeln.de, janine.weber@uni-koeln.de, url: http://www.numerik.uni-koeln.de and Center for Data and Simulation Science, University of Cologne, url: http://www.cds.uni-koeln.de

training data obtained from straight edges and consequently evaluated the trained network for test problems based on a regular domain decomposition only. Note that the approach of learning the adaptive constraints in an offline phase is in general of interest if a number of problems of the same class has to be solved, for example, diffusion or elasticity problems with different material coefficient distributions.

In this paper, we extend our idea from [6] by training regression neural networks which can be applied to both, regular domain decompositions as well as irregular decompositions as obtained by METIS [4]. To generalize our approach to arbitrary edge structures, we also train the network models with training data obtained from irregular edges and, additionally, with a set of randomized coefficient distributions. We provide numerical results for different heterogeneous stationary diffusion problems in two spatial dimensions for both, regular and irregular domain decompositions, and the adaptive coarse space from [10, 11].

2 Test problem and adaptive FETI-DP

As a test problem, we consider a stationary diffusion problem in two spatial dimensions

$$\begin{aligned} -\operatorname{div}(\rho \nabla u) &= 1 \quad \text{in } \Omega \\ u &= 0 \quad \text{on } \partial\Omega, \end{aligned} \tag{1}$$

where $\rho\colon \Omega := [0,1] \times [0,1] \to \mathbb{R}$ denotes a heterogeneous coefficient function. Its weak formulation is discretized with piecewise linear conforming finite elements.

In this paper, we consider a hybrid, adaptive FETI-DP method which uses supervised machine learning to setup a robust and efficient coarse space. Thus, we decompose our domain $\Omega \subset \mathbb{R}^2$ into a number of nonoverlapping subdomains Ω_i, $i = 1, \ldots, N$. Due to space limitations, we refrain from explaining the classic, that is, the non-adaptive FETI-DP method in detail. For a detailed description of the classic FETI-DP method, we refer to, e.g., [9]. Let us note that in our implementation, we always choose the vertices of the subdomains as primal variables. Additionally, we implement adaptive, that is, problem-dependent edge constraints to enhance the robustness of our methods; see the following discussion. For the remainder of the paper, we denote by $\mathcal{E}_{ij}$ the edge shared by the two neighboring subdomains Ω_i and Ω_j.

The classic FETI-DP condition number bound using exclusively primal vertex constraints is only robust under fairly restrictive assumptions on the coefficient function ρ; see, for example, [8]. Thus, we enhance the FETI-DP method with a very specific adaptive coarse space which was originally introduced in [10, 11].

Here, the main idea is to add selected eigenvectors to the coarse space, which are obtained from the solution of the following generalized local eigenvalue problem for each edge $\mathcal{E}_{ij}$: find $w_{ij} \in \left(\ker S_{ij}\right)^{\perp}$ such that

$$\langle P_{D_{ij}} v_{ij},\ S_{ij} P_{D_{ij}} w_{ij} \rangle = \mu_{ij} \langle v_{ij},\ S_{ij} w_{ij} \rangle \quad \forall\, v_{ij} \in \left(\ker S_{ij}\right)^{\perp}. \tag{2}$$

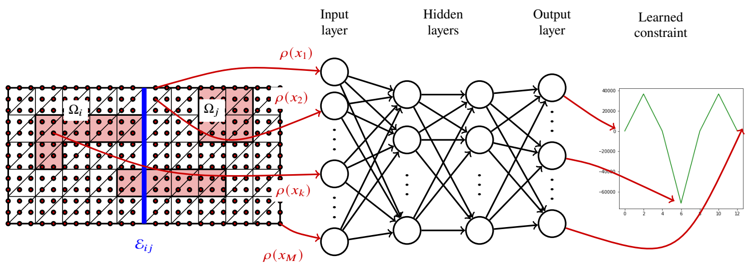

Fig. 1 Visualization of our network models N_l and $\widetilde{N}_l$, $l \leq 3$. As input data for the neural network, we use samples of the coefficent function for the two neighboring subdomains of an edge **(left)**. Here, dark red corresponds to a high coefficient and white corresponds to a low coefficient. The output of the network is a discretized egde constraint **(right)**. Figure taken from [6, Fig. 1].

Here, $S_{ij} = \text{diag}(S^{(i)}, S^{(j)})$ denotes a local Schur complement matrix with $S^{(i)}$ and $S^{(j)}$ being the Schur complements of $K^{(i)}$ and $K^{(j)}$, respectively, and $P_{D_{ij}} = B^T_{D,\mathcal{E}_{ij}} B_{\mathcal{E}_{ij}}$ is a local jump operator, with $B_{D,\mathcal{E}_{ij}} = \left(B^{(i)}_{D,\mathcal{E}_{ij}},\ B^{(j)}_{D,\mathcal{E}_{ij}}\right)$ being a local submatrix of $\left(B^{(i)}_D,\ B^{(j)}_D\right)$ obtained by exclusively taking the rows corresponding to the edge $\mathcal{E}_{ij}$; see [10] for more details. The matrix $B_{\mathcal{E}_{ij}}$ is obtained by taking the same rows from $\left(B^{(i)},\ B^{(j)}\right)$. We assume that R eigenvectors w^r_{ij}, $r = 1, \ldots, R$, belong to eigenvalues which are larger than a user-defined tolerance TOL and then enhance the FETI-DP coarse space with the edge constraint vectors

$$(c^r_{ij})^T := B_{D,\mathcal{E}_{ij}} S_{ij} P_{D_{ij}} w^r_{ij},\ r = 1, \ldots, R \tag{3}$$

using projector preconditioning; see [7, Sections 3,5] for more details. In the following we refer to the constraint vectors as constraints. When enhancing the FETI-DP coarse space with these adaptive constraints one can prove a robust condition number bound, which exclusively depends on the user-defined tolerance TOL and some geometrical constants; see, e.g., [7, Theorem 5.1]. On the one hand, this ensures a robust convergence behavior of the resulting FETI-DP algorithm, but, as a drawback, one has to setup and solve the eigenvalue problems in Eq. (2) for all edges belonging to the interface of our domain decomposition. Hence, in [6], we have proposed a hybrid FETI-DP method which uses a supervised regression model to directly learn approximations of the adaptive edge constraints resulting from Eq. (3) such that the solution of any eigenvalue problems is not necessary.

3 Learning coarse constraints in adaptive FETI-DP

The aim of our work is to compute discrete approximations of the first k adaptive edge constraints resulting from the local eigenvalue problem in Eq. (2) and to use the learned constraints to enhance the classic FETI-DP coarse space; see [6]. In particular, for each of the first k adaptive edge constraints, we train a separate regression neural network model that we denote by $N_l,\ l \leq k$. In the following, we always consider $k = 3$ and thus, train 3 different network models $N_l, l \leq 3$, to obtain 3 discrete approximations of the constraints resulting from the first 3 eigenmodes; see also [6]. As explained in more detail in [6], we additionally train separate neural network models for edges which have direct contact to the Dirichlet boundary $\partial\Omega_D$ of the domain and for edges without any contact to $\partial\Omega_D$ since both cases result in different edge constraints due to the influence of the Dirichlet boundary condition on the local Schur complement matrices S_{ij} in Eq. (2). To distinguish between these different network models, we denote the respective regression networks for edges with direct contact to $\partial\Omega_D$ by $\widetilde{N}_l, l \leq 3$; see also [6].

As input data for all neural network models $N_l,\ l \leq 3$, we use a mesh-independent image representation of the underlying coefficient function ρ within the two subdomains Ω_i and Ω_j adjacent to the edge $\mathcal{E}_{ij}$. The concrete details of the computation of this image representation are described in [2] such that, in the following, we only briefly sketch the main idea. First, we compute an auxiliary grid of points which we denote by *sampling grid* and which is independent of the finite element grid. Then, we evaluate the coefficient function ρ for each of these *sampling points* within the sampling grid and use the corresponding ρ values as input data for the neural networks. In order to make sure that the input data always have the same length and a consistent structure, we define a concrete order within our sampling grid and encode sampling points with the dummy value -1 if they fall outside the two neighboring subdomains for a given edge $\mathcal{E}_{ij}$. Let us note that this is especially relevant for irregular decompositions of the domain as obtained by METIS [4]. In particular, all trained network models N_l and $\widetilde{N}_l,\ l \leq 3$, share the same input data and only differ by their output data in order to define the concrete regression tasks. As specific output data for the different network models, we use discrete values of the adaptive edge constraints resulting from the local edge eigenvalue problems in Eq. (2). For the training of the l-th network N_l, we hence use a discretized version of the respective edge constraint resulting from the eigenvector w_{ij}^l belonging to the eigenvalue μ_{ij}^l. All in all, we use 3200 sampling points as input data for the neural networks and 19 output nodes, that is, 19 discrete values to approximate the adaptive edge constraints. In principle, the output space of our networks corresponds to an edge length defined by $H/h = 20$. However, in order to be able to evaluate the trained network for different finite element discretizations, we use an interpolation technique to generalize our approach to different mesh sizes and thus only use the number of 19 degrees of freedom for each edge as the basis for the interpolation. In case we want to apply the approximated constraints for finer or coarser finite element meshes, we linearly interpolate the obtained regression values by using the finite element mesh points as the interpolation points and the finite element basis functions as interpolation basis. An exemplary visualization of our network models N_l and $\widetilde{N}_l$ is given in Fig. 1.

Other than in [6], where we have trained and tested the different network models exclusively for regular edges, in this paper, we generalize these results also to irregular decompositions obtained by METIS [4]. Therefore, we train the different

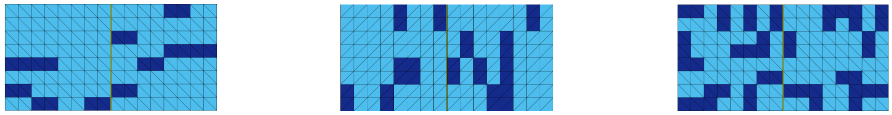

Fig. 2 Examples of three different randomly distributed coefficient functions obtained by using the same randomly generated coefficient for a horizontal (**left**) or vertical (**middle**) stripe of a maximum length of four finite element pixels, as well as by pairwise superimposing (**right**).

networks N_l and $\widetilde{N}_l$, $l \leq 3$, with both, regular and irregular edges. Additionally, in contrast to [6], we do not train the networks with our manually constructed set of coefficient distributions that we have denoted by *smart training data* in [6], but use a set of randomized coefficient distributions. In [3], we have shown that it is possible to achieve comparable accuracy results for the classification model as defined in [3] when using randomized training data with a slight structure compared to the smart training data. Considering these results and with regard to better expected generalization properties in three spatial dimensions, here, we have decided to also train our regression neural networks with randomized coefficient distributions. Three exemplary randomized coefficient distributions where we have additionally controlled the ratio of high versus low coefficient values are shown in Fig. 2. To obtain the entire set of training and validation data, we have generated various randomized coefficient distributions and combined them with pairs of subdomains adjacent to both, straight edges and edges resulting from the respective decompositions obtained by METIS. In particular, to generate the input and output data for the networks, we have used a regular decomposition of the unit square into 4×4 subdomains and a mesh size defined by $H/h \in \{10, 20, 40\}$ as well as the corresponding irregular decompositions obtained by METIS. All in all, this results in 4800 training and validation data configurations. In all coefficient configurations, we always set the high coefficient to $\rho_1 = 1e6$ and the low coefficient to $\rho_2 = 1$. For the selection of the necessary adaptive constraints, we always choose the tolerance $TOL = 100$.

Finally, for each of the network models N_l and $\widetilde{N}_l$, $l \leq 3$, we train a separate dense feedforward regression neural network [1] with 4 hidden layers and 50 neurons per layer. For each layer, we use the ReLU activation function and 20% dropout for each layer. For the optimization process, we have chosen the stochastic gradient descent (SGD) method using the Adam optimizer [5] with its default parameters, the initial learning rate of 0.001, and a batch size of 32. As loss function, we use the MSE (mean squared error) between the true adaptive and the predicted constraint vectors at the output grid points. For the final model, we obtain a MSE of $9.77e$-03 for the training data and a MSE of $4.62e$-02 for the validation data.

4 Numerical results

In this section, we provide numerical results for our proposed hybrid FETI-DP method using the approximated edge constraints as learned by the neural networks in direct comparison with the adaptive coarse space from [10].

To test our approach, we consider both, a regular decomposition and an irregular METIS decomposition of basically the same test problem. In both cases, the underlaying problem is a heterogeneous stationary diffusion problem, which we have already used in [6, Sect. 3]; see also Fig. 3 for a visualization. Only the underlying finite element discretization differs in both cases. Let us remark that this test configuration was of course not included in the training or validation data used for the training of the networks. For the test case with regular subdomains, we decompose our domain $\Omega = [0,1]^2$ into 4×4 square subdomains and use a regular finite element mesh defined by $H/h = 10$. We choose all vertices as primal variables and consider a coefficient contrast of $\rho_1/\rho_2 = 1e6$. In particular, we compare the robustness of the resulting coarse space when implementing our trained edge constraints to the adaptive coarse space from [10] and the condition and iteration numbers from [6, Sect. 3] where we have trained the regression networks exclusively with training data from straight edges. Note again that in this paper, we train the networks with both, training data from straight edges and from irregular edges resulting from a decomposition by METIS. In Fig. 4 (top), we show the two adaptive edge constraints resulting from the local eigenvalue problem in Eq. (2) for the tolerance $TOL = 100$, that is, the ground truth as well as the learned approximations from our regression neural networks. As we can see from Fig. 4 (top), for an exemplary straight edge $\mathcal{E}_{ij}$ between two floating subdomains, both approximations using either just straight edges for the training or using both, straight and irregular edges, result in quantitatively similar approximations of the two adaptive edge constraints. Using the approximated edge constraints in our hybrid FETI-DP method leads to an iteration number of 14 and a condition number estimate of 35.5 when training the network with straight edges only while training the network with both straight and irregular edges results in an iteration number of 17 and a condition number estimate of 57.9; see also Table 1. In particular, both approximate coarse spaces result in robust condition number estimates independent of the coefficient contrast and using both, straight and irregular edges for the training of the network models provides qualitatively similar results as we have obtained in [6].

To test the performance of our approach with a METIS decomposition, we consider a decomposition of the unit square into 4×4 irregular subdomains computed by METIS [4] and we choose 3200 finite elements for each subdomain; see also Fig. 3 (right). Again, we consider a coefficient contrast of $\rho_1/\rho_2 = 1e6$. We evaluate our regression neural networks N_l and $\widetilde{N}_l$, $l \leq 3$, trained with straight and irregular edges for all 34 irregular edges resulting from the domain decomposition obtained by METIS in Fig. 3 (right), and integrate the learned edge constraints into the FETI-DP coarse space. The resulting iteration number and condition number estimate are given in Table 1, where we also show the corresponding values for the adaptive coarse space from [10] and the tolerance $TOL = 100$. As we can observe from Table 1,

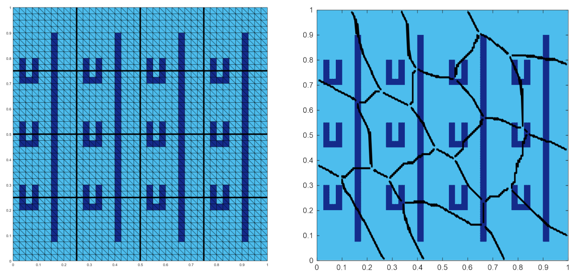

Fig. 3 Heterogeneous test problem: Stationary diffusion problem, coefficient contrast $1e6$, $\Omega = [0, 1]^2$ decomposed into 4×4 subdomains. **Left:** Regular decomposition, mesh size defined by $H/h = 10$. **Right:** Irregular decomposition computed by METIS with 3200 FEs per subdomain.

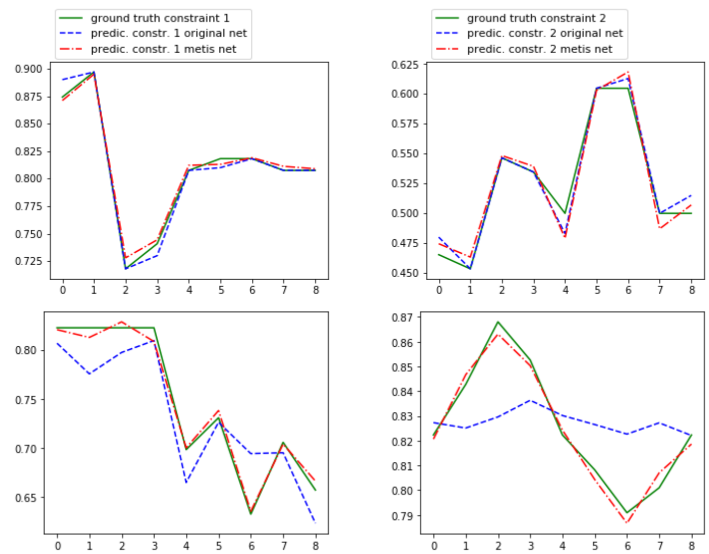

Fig. 4 Results for a straight edge of the regular decomposition (**top row**) and for an exemplary edge of the irregular decomposition (**bottom row**) of the test problem; see Fig. 3. Green, solid line: ground truth for the tolerance TOL = 100. Blue, dashed line: prediction as obtained by the neural networks in [6]. Red, dashed-dotted line: prediction as obtained by the neural networks trained with both, straight and irregular edges. See Table 1 for the resulting condition and iteration numbers.

using the learned constraints leads to a condition number estimate of 67.64 that is clearly independent of the coefficient contrast and in a quantitatively similar order of magnitude as the respective condition number for the adaptive FETI-DP coarse space. Thus, the learned coarse space seems to serve as a good approximation of the respective adaptive FETI-DP coarse space. Furthermore, in Fig. 4 (bottom), we show the learned constraints as well as the ground truth for an exemplary edge within the irregular decomposition. We can see that the learned constraints when training the networks with both, straight and irregular edges, are quantitatively similar to the ground truth. However, evaluating our networks from [6], which were only trained

Table 1 Condition number estimates (cond) and iteration numbers (iter) for the adaptive FETI-DP coarse space and the hybrid coarse spaces as learned by the regression neural networks for the coefficient distributions in Fig. 3. We denote by *METIS nets* the neural networks that are trained with both, straight and irregular edges.

	Regular decomposition		**METIS decomposition**	
	iter	**cond**	**iter**	**cond**
Classic FETI-DP	55	32443	79	375020
Adaptive FETI-DP	10	2.81	19	3.32
Learned constraints from [6]	14	35.56	41	7055.95
Learned constraints from METIS nets	17	57.97	26	67.64

with straight edges, for an irregular edge, provides a relatively poor approximation of the constraints. Note again that the setup of the learned coarse space does not require the solution of any eigenvalue problems at all and the training of the different network models can be executed in parallel and in an apriori offline phase. In particular, in this work, we have shown that it is possible to generalize our results from [6] also to non-straight edges as, e.g., resulting from METIS [4].

References

1. Goodfellow, I., Bengio, Y., and Courville, A. *Deep learning*. MIT press Cambridge (2016).
2. Heinlein, A., Klawonn, A., Lanser, M., and Weber, J. Machine Learning in Adaptive Domain Decomposition Methods - Predicting the Geometric Location of Constraints. *SIAM J. Sci. Comput.* **41**(6), A3887–A3912 (2019).
3. Heinlein, A., Klawonn, A., Lanser, M., and Weber, J. Machine Learning in Adaptive FETI-DP - A Comparison of Smart and Random Training Data. In: *Domain decomposition methods in science and engineering XXV*, *Lect. Notes Comput. Sci. Eng.*, vol. 138, 218–226. Springer (2020).
4. Karypis, G. and Kumar, V. METIS: Unstructured Graph Partitioning and Sparse Matrix Ordering System, Version 4.0. http://www.cs.umn.edu/ metis (2009).
5. Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. In: *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings* (2015). Http://arxiv.org/abs/1412.6980.
6. Klawonn, A., Lanser, M., and Weber, J. Learning Adaptive Coarse Basis Functions of FETI-DP (2022). TR series, Center for Data and Simulation Science, University of Cologne, Germany, Vol. 2022-2. http://kups.ub.uni-koeln.de/id/eprint/62001. Submitted for publication in 06/2022.
7. Klawonn, A., Radtke, P., and Rheinbach, O. A comparison of adaptive coarse spaces for iterative substructuring in two dimensions. *Electron. Trans. Numer. Anal.* **45**, 75–106 (2016).
8. Klawonn, A., Rheinbach, O., and Widlund, O. B. An analysis of a FETI-DP algorithm on irregular subdomains in the plane. *SIAM J. Numer. Anal.* **46**(5), 2484–2504 (2008).
9. Klawonn, A. and Widlund, O. B. Dual-primal FETI methods for linear elasticity. *Comm. Pure Appl. Math.* **59**(11), 1523–1572 (2006).
10. Mandel, J. and Sousedík, B. Adaptive selection of face coarse degrees of freedom in the BDDC and the FETI-DP iterative substructuring methods. *Comput. Methods Appl. Mech. Engrg.* **196**(8), 1389–1399 (2007).
11. Mandel, J., Sousedík, B., and Šístek, J. Adaptive BDDC in three dimensions. *Math. Comput. Simulation* **82**(10), 1812–1831 (2012).