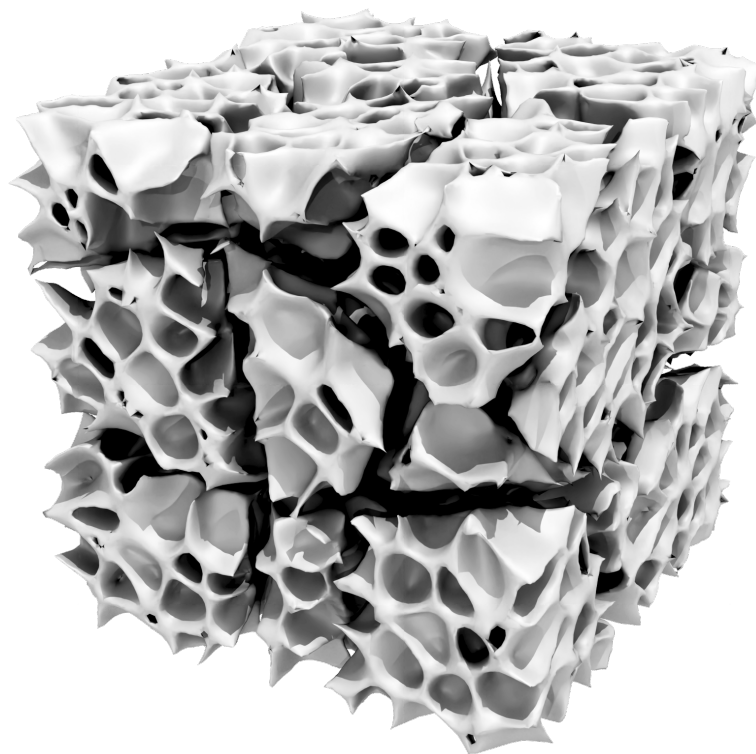# Adaptive Coarse Spaces
for the Overlapping Schwarz Method
and Multiscale Elliptic Problems

Jascha Knepper

Adaptive Coarse Spaces
for the Overlapping Schwarz Method
and Multiscale Elliptic Problems

Inaugural-Dissertation

zur

Erlangung des Doktorgrades

Dr. rer. nat.

der Mathematisch-Naturwissenschaftlichen Fakultät

der Universität zu Köln

vorgelegt von

Jascha Knepper

aus Pforzheim

Köln | 2022

Berichterstatter	Prof. Dr. Axel Klawonn
	(Universität zu Köln)
	Prof. Marcus Sarkis, PhD
	(Worcester Polytechnic Institute)
	Clark R. Dohrmann, PhD
	(Sandia National Laboratories)
Datum der Disputation	23. März 2022

Abstract

In science and engineering, many problems exhibit multiscale properties, making the development of efficient algorithms to compute accurate solutions often challenging. An example of a multiscale problem is given by the deformation of dual-phase steel, a high-strength steel that contains hard inclusions of martensite in a soft matrix material of ferrite.

We consider finite element discretizations of linear, second-order, elliptic partial differential equations with highly heterogeneous coefficient functions. The associated linear systems of equations for fine meshes are large and ill conditioned; in particular, the condition number depends on the heterogeneity of the coefficient function. For the solution of the systems, we employ the conjugate gradient method and additive overlapping Schwarz domain decomposition preconditioners. To obtain a scalable method, it is generally required to solve a coarse problem in each iteration to quickly transport information across the domain.

In most cases, standard coarse spaces only perform satisfactorily for problems with small variations in the coefficient function. Our goal is to construct coarse spaces such that the condition number of the preconditioned system depends only on a user-prescribed tolerance and on a constant that is independent of the coefficient function and of the fine and coarse mesh resolution. We denote a coarse space that satisfies these properties as adaptive and robust.

To achieve robustness, we construct energy-minimizing coarse spaces that take the variations of the coefficient function into account. First, we partition the domain decomposition interface into small components. Adaptivity is then achieved via coarse functions that are based on solving local generalized eigenvalue problems on the interface components. A user-prescribed tolerance is used to select the most effective eigenfunctions, which are subsequently extended energy-minimally to the interior of the subdomains to construct coarse functions.

With a growing number of subdomains, the coarse problem increases in size as well. It is thus important to keep its size as small as possible. To this end, we present techniques

for reducing the dimension of coarse spaces. The first technique—the incorporation of an energy-minimizing extension into the generalized eigenvalue problems—is inherent to all adaptive coarse spaces presented in this thesis. Second, we propose specific interface partitions with a reduced number of interface components that help to incorporate more information about the heterogeneity of the problem into the coarse functions. Third, we enforce additional Dirichlet boundary conditions in the energy-minimizing extensions of the generalized eigenvalue problems to further reduce the coarse space dimension. Each technique increases the complexity to describe and implement the algorithm. Therefore, the optimal combination of techniques depends not only on the specific heterogeneous problem but also on practical restrictions.

We complement the dimension-reduction techniques by presenting modifications that can reduce the computational cost and facilitate a parallel implementation. For all adaptive coarse spaces and variants, we prove condition number bounds that only depend on a user-prescribed tolerance and on a constant that is independent of the typical mesh parameters and of the coefficient function. We provide supporting numerical results for diffusion and linear elasticity problems.

Kurzfassung

In der Wissenschaft und den Ingenieurwissenschaften gibt es vielfältige Multiskalenprobleme, was die Entwicklung effizienter Algorithmen zur Bestimmung präziser Lösungen erschwert. Ein Beispiel dafür ist durch die Verformung von Dualphasenstahl gegeben, einem hochfesten Stahl, welcher harte Martensiteinschlüsse in einer weichen, ferritischen Matrix enthält.

Wir untersuchen in dieser Arbeit Finite-Elemente-Diskretisierungen von linearen, elliptischen partiellen Differentialgleichungen zweiter Ordnung mit stark heterogenen Koeffizientenfunktionen. Die zugehörigen linearen Gleichungssysteme von feinen Gittern sind groß und schlecht konditioniert; insbesondere hängt die Konditionszahl von der Heterogenität der Koeffizientenfunktion ab. Zur Lösung der linearen Gleichungssysteme verwenden wir das konjugierte Gradientenverfahren und additive überlappende Schwarz-Gebietszerlegungsverfahren als Vorkonditionierer. Um eine skalierbare Methode zu erhalten, ist es im Allgemeinen notwendig, in jedem Schritt der Iteration ein Grobgitterproblem zu lösen, welches für den schnellen Informationstransport über das Gebiet hinweg sorgt.

In der Regel sind Standard-Grobgitterräume nur für Probleme mit geringer Variation in der Koeffizientenfunktion gut geeignet. Unser Ziel ist die Konstruktion von Grobgitterräumen, sodass die Konditionszahl des vorkonditionierten Systems nur von einer benutzerdefinierten Toleranz abhängt sowie von einer Konstanten, welche unabhängig von der Koeffizientenfunktion und von der Auflösung des feinen und groben Gitters ist. Wir nennen einen Grobgitterraum, der diese Bedingungen erfüllt, adaptiv und robust.

Um Robustheit zu erreichen, konstruieren wir energieminimierende Grobgitterräume, welche die Variation der Koeffizientenfunktion berücksichtigen. Zunächst partitionieren wir das Interface der Gebietszerlegung in kleine Komponenten. Adaptivität bezüglich der Koeffizientenfunktion wird über Grobgitterfunktionen erreicht, welche auf der Lösung lokaler verallgemeinerter Eigenwertprobleme auf den Interfacekomponenten basieren. Eine benutzerdefinierte Toleranz dient zur Auswahl der nützlichsten Eigenfunktionen, welche anschließend energieminimierend in das Innere der Teilgebiete zur Konstruktion

von Grobgitterfunktionen fortgesetzt werden.

Mit einer wachsenden Anzahl an Teilgebieten wächst auch die Dimension des Grobgitterproblems, weshalb es wichtig ist, diese Dimension möglichst klein zu halten. Zu diesem Zweck stellen wir in dieser Arbeit Techniken zur Reduktion der Dimension von adaptiven Grobgitterräumen vor. Der erste Ansatz – die Integrierung einer energieminimierenden Fortsetzung in das verallgemeinerte Eigenwertproblem – liegt allen adaptiven Grobgitterräumen zugrunde, die in der vorliegenden Arbeit vorgestellt werden. Als Zweites schlagen wir die Nutzung spezieller Interfacepartitionierungen mit einer reduzierten Anzahl an Interfacekomponenten vor, welche es ermöglichen, mehr Informationen über die Heterogenität des Problems in die Grobgitterfunktionen zu integrieren. Zuletzt zeigen wir die Möglichkeit auf, zusätzliche Dirichletrandbedingungen bei der Berechnung energieminimierender Fortsetzungen der verallgemeinerten Eigenwertprobleme zu erzwingen, um eine weitere Reduktion der Grobgitterraumdimension zu erreichen. Jeder Ansatz ist mit einer wachsenden Komplexität der Beschreibung und Implementierung des Algorithmus verbunden. Die Auswahl einer optimalen Kombination der verschiedenen Ansätze ist daher nicht nur vom spezifischen heterogenen Problem abhängig, sondern auch von praktischen Einschränkungen.

Wir komplementieren die Ansätze zur Dimensionsreduktion mit einigen Varianten, welche unter anderem den Rechenaufwand verringern oder eine parallele Implementierung erleichtern. Wir beweisen für alle adaptiven Grobgitterräume und deren Varianten Schranken für die Konditionszahl des vorkonditionierten Systems. Die Schranken hängen lediglich von einer benutzerdefinierten Toleranz ab sowie von einer Konstanten, welche unabhängig von den typischen Gitterparametern und der Koeffizientenfunktion ist. Schlussendlich zeigen wir zu einigen Modellproblemen der Diffusion und linearen Elastizität numerische Ergebnisse.

Acknowledgements

A scientific work—in this case, my doctoral thesis—is always influenced by many people. Studies suggest that we tend to overestimate our own influence on successful endeavors and underestimate the influence and contributions by others and that of pure chance [Tho99; Fra16]. Bearing this in mind, I cannot thank everyone who had a direct or indirect positive influence on the making of this thesis. But I do want to express my gratitude toward those that have had a direct and sizeable influence: my family, colleagues—former, current, and those newly joined and immediately condemned to proofreading—and coauthors, who have accompanied me during my studies and have helped this work come to fruition.

Toward the end of my studies for a master's degree, I was asked by my advisor Professor Dr. Axel Klawonn whether I was interested in pursuing a doctoral degree. At that time, my head was still wrapped around finishing my master's degree. This early initiation allowed a smooth transition from my master's studies to studies toward a doctoral degree. For this, for giving me the opportunity to work in his group, supervising my works the following years, and giving me a substantial amount of freedom in my endeavors, I would like to especially thank him.

For taking the time to review my extensive thesis in a rather short time, I am grateful to my referees Axel Klawonn, Marcus Sarkis, and Clark R. Dohrmann. Thanks are also due to Alexander Heinlein, Axel Klawonn, Oliver Rheinbach, and Olof Widlund for their collaboration and coauthoring the works on which this thesis is based. I owe special thanks to Alexander Heinlein—our collaboration already began with my master's thesis—for countless discussions on adaptive coarse spaces and other topics not included in this thesis.

Furthermore, I am grateful to my current and former colleagues for their support and time spent together, both professionally and personally: Stephanie Friedhoff, Viktor Grimm, Alexander Heinlein, Rebekka-Sarah Hennig, Christian Hochmuth, Martin Kühn, Martin Lanser, Sabine Musielack-Erle, Patrick Radtke, Lea Saßmannshausen, Matthias Uran, Adam Wasiak, and Janine Weber, in nonjudgemental, alphabetic order. Most

Contents

Contents

List of Figures

List of Tables

Notation and Coarse Space Acronyms

Basic Notation

Ω/Ω^h	considered domain $\Omega \subset \mathbb{R}^d$ / finite element nodes of $\overline{\Omega}$
d	space dimension of Ω
$\hat{d}$	scalar diffusion problem: $\hat{d} = 1$; linear elasticity problem: $\hat{d} = d$
$\tau_h(\Omega)$	triangulation: triangles/rectangles/tetrahedra/cuboids
x^h	finite element node
E	coefficient function; Young's modulus in the case of linear elasticity
$\partial\Omega_D/\partial\Omega_N$	subset of $\partial\Omega$ associated with a Dirichlet/Neumann condition
$V^h(\Omega)$	finite element space on Ω (piecewise linear, bilinear, or trilinear)
$I^h(\cdot)$	pointwise interpolation operator of $V^h(\Omega)$
$V^h_{0,\partial\Omega_D}(\Omega)$	space of functions in $V^h(\Omega)$ that vanish on $\partial\Omega_D$
h_T	finite element diameter: $h_T := \mathrm{diam}(T)$
$\|T\|$	$\|T\| := \mathrm{measure}(T) := \int_T 1 \, \mathrm{d}x$
N	number of subdomains
$\{\Omega_i\}_{i=1}^N$	nonoverlapping subdomains
$\{\Omega_i'\}_{i=1}^N$	overlapping subdomains of the overlapping Schwarz method
$\{\tilde{\Omega}_i\}_{i=1}^N$	overlapping subdomains; overlap of one layer of finite elements; used during the proof of a condition number bound

Norms and Seminorms

Let $u \in \left(H^1(\Omega)\right)^{\hat{d}}$.

$A : B$ $\qquad$ $A : B := \operatorname{tr}(A^T B) = \operatorname{tr}(A B^T) = \sum_{i=1}^{\hat{d}} \sum_{j=1}^{d} A_{ij} B_{ij}, \quad A, B \in \mathbb{R}^{\hat{d} \times d}$

$\|A\|_{L^2(\Omega)}$ $\qquad$ $\|A\|_{L^2(\Omega)} := \|A : A\|_{L^2(\Omega)}, \quad A \colon \mathbb{R}^d \to \mathbb{R}^{\hat{d} \times d}$

$\|u\|_{L^p(\Omega)}$ $\qquad$ $\|u\|_{L^p(\Omega)}^p := \int_\Omega \|u(x)\|_{l^p}^p \, \mathrm{d}x, \quad 1 \le p < \infty$

$\|u\|_{L^\infty(\Omega)}$ $\qquad$ $\|u\|_{L^\infty(\Omega)} := \inf\{\, s \ge 0 : \|u(x)\|_{l^\infty} \le s \text{ a.e.} \,\}$

$|u|_{H^1(\Omega)}$ $\qquad$ $|u|_{H^1(\Omega)}^2 := \|\nabla u\|_{L^2(\Omega)}^2, \quad (\nabla u)_{i,j} := \frac{\partial u_i}{\partial x_j}, \nabla u \in \mathbb{R}^{\hat{d} \times d}$

$\|u\|_{H^1(\Omega)}$ $\qquad$ $\|u\|_{H^1(\Omega)}^2 := |u|_{H^1(\Omega)}^2 + \|u\|_{L^2(\Omega)}^2$

$|u|_\alpha$ $\qquad$ $|u|_\alpha := \sqrt{\alpha(u, u)}$, α: symmetric, positive semidefinite bilinear form

$\|u\|_\beta$ $\qquad$ $\|u\|_\beta := \sqrt{\beta(u, u)}$, β: symmetric, positive definite bilinear form

$|u|_{a(B)}$ $\qquad$ $|u|_{a(B)} := \sqrt{a_B(u, u)};$ $\qquad\qquad$ see (6.30)

$|u|_{a, q \to \Omega_\xi}$ $\qquad$ $|u|_{a, q \to \Omega_\xi} := |\mathcal{H}_{q \to \Omega_\xi}(u)|_{a(\Omega_\xi)}, \ q \subset \bar{\xi}, \ \xi \in \mathcal{P}, \ u \in X^h(q);$ $\qquad$ see (7.18)

Interface and its Decomposition

$n(\omega)$ $\qquad$ set of indices of subdomains Ω_k such that $\omega \cap \overline{\Omega}_k \ne \emptyset$

Γ $\qquad$ domain decomposition interface; $\qquad\qquad$ see (1.11)

Γ^h $\qquad$ finite element nodes of Γ; $\Gamma^h = \{\, x^h \in \overline{\Omega} \setminus \partial\Omega_D : |n(x^h)| > 1 \,\}$

$\mathcal{W}$ $\qquad$ wire basket; $\mathcal{W} := \{\, x^h \in \Gamma^h : |n(x^h)| \ge 3 \,\}$

$\mathcal{P}$ $\qquad$ partition of the domain decomposition interface

ξ $\qquad$ interface/coarse component (set of finite element nodes), $\xi \in \mathcal{P}$

$\mathcal{P}(\Omega_i)$ $\qquad$ $\mathcal{P}(\Omega_i) := \{\, \xi \in \mathcal{P} : \xi \cap \overline{\Omega}_i \ne \emptyset \,\};$ $\qquad\qquad$ see (6.34)

NEC $\qquad$ nodal equivalence class; $\qquad\qquad$ see section 5.1

$\mathcal{N}_\xi = \{\xi_i\}_{i=1}^{n_\xi}$ $\qquad$ subcomponents/NECs of a $\xi \in \mathcal{P}$; $\qquad\qquad$ see (5.2) and (5.3)

$\mathcal{N}_{\Gamma^h}$ $\qquad$ NECs of the interface

$\mathcal{N}_{ec,\mathcal{P}}$ $\qquad$ $\mathcal{N}_{ec,\mathcal{P}} := \bigcup_{\xi \in \mathcal{P}} \mathcal{N}_\xi;$ $\qquad\qquad$ see (5.4)

$\mathcal{V}/\mathcal{E}/\mathcal{F}$ $\qquad$ set of coarse nodes/edges/faces; $\qquad\qquad$ see section 5.3

$\mathcal{S}_\Gamma$ $\qquad$ set of (coarse) interface stars; $\qquad\qquad$ see section 5.4

$\mathcal{S}_\mathcal{W}$ $\qquad$ set of (coarse) wire basket stars; $\qquad\qquad$ see section 5.5

Extension Operators

$X^h(\omega)$	space of functions that map from $\omega \subset \mathbb{R}^d$ to $\mathbb{R}^{\hat{d}}$;	see (6.5)
$\overline{\Omega}_\xi$	union of the closure of subdomains adjacent to $\xi \in \mathcal{P}$;	see (6.4)
$\mathcal{H}_{\xi \to \Omega_\xi}(\cdot)$	energy-minimizing extension from ξ to Ω_ξ;	see (6.6)
$z_\xi(\cdot)$	extension-by-zero from ξ to $V^h(\Omega)$;	see (6.9)

For ACMS-type coarse spaces, the energy-minimizing extension $\mathcal{H}_{\overline{\xi} \to \Omega_\xi}(\cdot)$ is used. The following sets are related to $\mathcal{H}_{\overline{\xi} \to \Omega_\xi}(\cdot)$:

$\overline{\xi}/\partial\xi$	$\overline{\xi} := \xi \cup \partial\xi$; $\partial\xi := \bigcup_{\varphi \in \mathcal{B}_\mathrm{p}(\xi)} \varphi \cap \overline{\Omega}_\xi$; see (7.29), (7.32), and section 7.2.1	
$\mathcal{B}_\mathrm{c}(\xi)/\mathcal{B}_\mathrm{p}(\xi)$	"children/parents" of ξ; see (7.30), (7.31), and sections 7.2.3 and 7.5	
$\mathcal{B}_\mathrm{D}(\xi)/\mathcal{B}_\mathrm{A}(\xi)$	"descendants/ancestors" of ξ; see (7.33) and (7.34)	

Constants

$\hat{N}_c$	$\hat{N}_c := \max_{x^h \in \overline{\Omega}} \left| \{ i \in \{1,\ldots,N\} : x^h \in \overline{\Omega}'_i \} \right|$;	see section 1.3
C_τ	maximum number of vertices of any element $T \in \tau_h(\Omega)$;	see (6.33)
C_inv	constant stemming from inverse inequalities;	see (6.26)
$N^\xi/N^e/N^f$	maximum number of interface components / coarse edges / coarse faces in any subdomain;	see (6.34), (7.21), and (7.22)
N^{ξ,tol_ξ}	$N^{\xi,tol_\xi} := \max_{1 \le i \le N} \sum_{\xi \in \mathcal{P}(\Omega_i)} \frac{1}{tol_\xi}$;	see (6.36)
$\mathcal{C}$	$\mathcal{C} := \max_{1 \le i \le N} \sum_{j=1}^N \left| \{ \xi \in \mathcal{P} : i,j \in n(\xi) \} \right|$;	see (6.35)
$\mathcal{C}^{tol_\xi}$	$\mathcal{C}^{tol_\xi} := \max_{1 \le i \le N} \sum_{\xi \in \mathcal{P}(\Omega_i)} \frac{|n(\xi)|}{tol_\xi}$;	see (6.37)
$\hat{\mathcal{C}}^{tol_\xi}$	$\hat{\mathcal{C}}^{tol_\xi} := \max_{1 \le l \le N} \sum_{\xi \in \mathcal{P}(\Omega_l)} \frac{\max_{i=1,\ldots,n_\xi} |n(\xi_i)|}{tol_\xi}$;	see (6.48)
$N^{\partial f \to e}$	$N^{\partial f \to e} := \max_{1 \le i \le N} \sum_{f \in \mathcal{F}(\Omega_i)} |\mathcal{E} \cap \mathcal{B}_\mathrm{p}(f)|$;	see (7.23)
$N^{e,\Sigma}$	$N^{e,\Sigma} := \max_{1 \le i \le N} \sum_{e \in \mathcal{E}(\Omega_i)} |n(e)|$;	see (7.24)
$N^{f,\Sigma}$	$N^{f,\Sigma} := \max_{1 \le i \le N} \sum_{f \in \mathcal{F}(\Omega_i)} |n(f)|$;	see (7.25)
$N^{\partial f \to e,\Sigma}$	$N^{\partial f \to e,\Sigma} := \max_{1 \le i \le N} |n(f)| \sum_{f \in \mathcal{F}(\Omega_i)} |\mathcal{E} \cap \mathcal{B}_\mathrm{p}(f)|$;	see (7.26)

Generalized Eigenvalue Problems

tol_ξ tolerance for the selection of eigenvectors/eigenfunctions

$tol_\mathcal{P}$ $tol_\mathcal{P} := \min_{\xi \in \mathcal{P}} tol_\xi$

$\Pi_\xi u$ spectral projection; see (6.32)

$\Pi_\xi u := \sum_{\lambda_{k,\xi} \leq tol_\xi} \beta_\xi(u, v_{k,\xi}) v_{k,\xi}$, $v_{k,\xi}$: coarse function

$\beta_\xi^K(\cdot, \cdot)$ $\beta_\xi^K(u, v) := \sum_{i=1}^{n_\xi} a_{\Omega_{\xi_i}}(z_{\xi_i}(u), z_{\xi_i}(v))$; see (6.10)

$\hat{h}_T$ h_T or radius of largest insphere or other related measure

$\beta_\xi^M(\cdot, \cdot)$ $\beta_\xi^M(u, v) := \sum_{T \in \tau_h(\Omega_\xi)} \frac{E(T)}{\hat{h}_T^2} \int_T z_\xi(u(x)) \cdot z_\xi(v(x)) \, dx$; see (6.22)

GDSW-type Coarse Space

$\alpha_\xi^K(\cdot, \cdot)$ $\alpha_\xi^K(u, v) := a_{\Omega_\xi}(\mathcal{H}_{\xi \to \Omega_\xi}(u), \mathcal{H}_{\xi \to \Omega_\xi}(v))$; see (6.8)

$\alpha_\xi^{K,l}(\cdot, \cdot)$ $\alpha_\xi^{K,l}(u, v) := a_{\Omega_\xi^l}(\mathcal{H}_{\xi \to \Omega_\xi^l}(u), \mathcal{H}_{\xi \to \Omega_\xi^l}(v))$; see (6.15)

$\alpha_\xi^S(\cdot, \cdot)$ $\alpha_\xi^S(u, v) := \sum_{k \in n(\xi)} a_{\Omega_k}(\mathcal{H}_{\xi \to \Omega_k}(u), \mathcal{H}_{\xi \to \Omega_k}(v))$; see (6.18)

$\alpha_\xi^{S,l}(\cdot, \cdot)$ $\alpha_\xi^{S,l}(u, v) := \sum_{k \in n(\xi)} a_{\Omega_k^l}(\mathcal{H}_{\xi \to \Omega_k^l}(u), \mathcal{H}_{\xi \to \Omega_k^l}(v))$; see (6.20)

ACMS-type Coarse Space

$\overline{\alpha}_\xi^K(\cdot, \cdot)$ $\overline{\alpha}_\xi^K(u, v) := \alpha_{\overline{\xi}}^K(z_\xi(u), z_\xi(v))$; see (7.3)

$\alpha_{\overline{\xi}}^K(u, v) := a_{\Omega_\xi}(\mathcal{H}_{\overline{\xi} \to \Omega_\xi}(u), \mathcal{H}_{\overline{\xi} \to \Omega_\xi}(v))$

$\overline{\alpha}_\xi^{K,l}(\cdot, \cdot)$ $\overline{\alpha}_\xi^{K,l}(u, v) := \alpha_{\overline{\xi}}^{K,l}(z_\xi(u), z_\xi(v))$; see (7.11)

$\alpha_{\overline{\xi}}^{K,l}(u, v) := a_{\Omega_\xi^l}(\mathcal{H}_{\overline{\xi} \to \Omega_\xi^l}(u), \mathcal{H}_{\overline{\xi} \to \Omega_\xi^l}(v))$

$\overline{\alpha}_\xi^S(\cdot, \cdot)$ $\overline{\alpha}_\xi^S(u, v) := \alpha_{\overline{\xi}}^S(z_\xi(u), z_\xi(v))$; see (7.12)

$\alpha_{\overline{\xi}}^S(u, v) := \sum_{k \in n(\xi)} a_{\Omega_k}(\mathcal{H}_{\overline{\xi} \to \Omega_k}(u), \mathcal{H}_{\overline{\xi} \to \Omega_k}(v))$

$\overline{\alpha}_\xi^{S,l}(\cdot, \cdot)$ $\overline{\alpha}_\xi^{S,l}(u, v) := \alpha_{\overline{\xi}}^{S,l}(z_\xi(u), z_\xi(v))$; see (7.13)

$\alpha_{\overline{\xi}}^{S,l}(u, v) := \sum_{k \in n(\xi)} a_{\Omega_k^l}(\mathcal{H}_{\overline{\xi} \to \Omega_k^l}(u), \mathcal{H}_{\overline{\xi} \to \Omega_k^l}(v))$

Acronyms of Methods

Nonadaptive Methods

CG	conjugate gradient method without a preconditioner	
OSL1	one-level additive overlapping Schwarz method;	see section 1.3
GDSW	generalized Dryja–Smith–Widlund;	see section 1.4
RGDSW	reduced-dimension GDSW;	see section 4.1

Adaptive Coarse Spaces

AGDSW	adaptive GDSW;	see chapter 3
	interface partition: coarse nodes, edges, faces (section 5.3)	
RAGDSW	reduced-dimension AGDSW;	see chapter 4
	interface partition: interface stars (section 5.4)	
R–WB–AGDSW	reduced-dimension AGDSW;	see chapter 4
	interface partition: wire basket stars (section 5.5), coarse faces	
OS–ACMS	ACMS-type coarse space;	see section 7.2
	approximate component mode synthesis (ACMS)	
	interface partition: coarse nodes, edges, faces	
	MsFEM-type functions associated with coarse nodes	
R–WB–OS–ACMS	reduced-dimension OS–ACMS;	see section 7.3
	interface partition: wire basket stars (section 5.5), coarse faces	
SHEM	spectral harmonically enriched multiscale coarse space;	see [GLR15]
	interface partition (2D): coarse nodes, edges	
EMR–VB	vertex-based coarse space;	see [EMR19]
	interface partition: coarse nodes, edges, faces	
EMR–WB	wire basket coarse space;	see [EMR19]
	interface partition: wire basket, coarse faces	
GenEO	generalized eigenproblems in the overlaps;	see [SDH+14a]

Coarse Space Variants

The following table lists modifiers used to describe variants. We abbreviate generalized eigenvalue problem with GEVP, left-hand side with LHS, and right-hand side with RHS.

modifier	example	comment
S	RAGDSW–S	sections 3.3.3 and 7.2.6 sum of local Schur complements on LHS of GEVP
(l)	AGDSW–S(3)	section 3.3.2 slab of l layers of finite elements on LHS of GEVP
K	OS–ACMS–K	chapter 3 and section 4.2 stiffness matrix on RHS of GEVP
M	AGDSW–M	section 3.3.4 mass matrix on RHS of GEVP
$\ell(\cdot)$	AGDSW–ℓ(M) AGDSW–ℓ(K)	sections 3.3.1 and 3.3.4 lumped mass or stiffness matrix on RHS of GEVP

Owing to historical reasons, the adaptive GDSW-type coarse spaces use a stiffness matrix on the right-hand side of the generalized eigenvalue problem if not indicated otherwise. For example, AGDSW refers to AGDSW–K and RAGDSW–S to RAGDSW–S–K. Similarly, ACMS-type coarse spaces use a mass matrix on the right-hand side if not indicated otherwise.

1. Introduction

In science and engineering, many problems exhibit multiscale properties for which it is often challenging to compute accurate solutions efficiently. Examples of this are given by the deformation of dual-phase steels—high-strength steels that contain hard inclusions of martensite in a soft matrix material of ferrite [TDY+15]—or by plate tectonics—where the convection of tectonic plates is subjected to large variations in the viscosity at plate boundaries [SGB+10]—or by pneumatic tires, which are composites of a fiber-reinforced rubber that is further reinforced with a steel cord for improved durability and stability [Nak19].

Partial differential equations are ubiquitous in science and engineering and can be used to model a great number of processes. The problems dealt with in this work are linear, second order, and elliptic, for example, the diffusion equation or the equations of linear elasticity. In general, these types of equations cannot be solved analytically (cf. [Eva10]), and we resort to the finite element method for the computation of an approximate solution [Cia02; BS08; LB13].

Often, problems stemming from real-world applications require a mesh to be highly resolved, which results in large and sparse linear systems. An alternative to direct solvers for large and sparse systems are iterative solvers, such as Krylov subspace methods with suitable preconditioners; see, e.g., [Saa03]. The problems considered in this work give rise to symmetric, positive definite system matrices, which allows us to use the preconditioned conjugate gradient method as an iterative solver; cf. [Saa03, sect. 6.7]. A highly resolved finite element mesh results in ill-conditioned system matrices and, thus, the conjugate gradient method is likely to converge slowly. A remedy is, for example,

provided by standard domain decomposition and multigrid methods; see, e.g., [Hac85; TOS00; TW05; XZ17]. For this work, we focus on using additive overlapping Schwarz domain decomposition methods; cf. [SBG96; TW05] and section 1.3 of this thesis.

A key concept is numerical scalability: for a fixed size but an increasing number of subdomains, the number of iterations required for convergence should stay asymptotically constant. Numerical scalability is required to obtain parallel scalability, that is, not only the number of iterations should stay asymptotically constant but also the time to solution. To achieve this, the number of processing units has to scale with the number of subdomains.

Domain decomposition methods, such as the overlapping Schwarz method, generally require the solution of a coarse problem (also called the second level) to obtain scalability; see, e.g., [SBG96; TW05]. With an increasing number of subdomains, the dimension of the associated coarse space increases as well, which in turn increases the time to solve the coarse problem. It is thus of interest to keep the size of the coarse problem as small as possible.

We consider finite element discretizations of problems with highly heterogeneous coefficient functions, which generally lead to large and ill-conditioned linear systems. The condition number of a system does not only depend on the resolution of the discretization but also on the heterogeneity of the underlying problem. In this work, we present algorithms that can be used to efficiently solve a class of highly heterogeneous problems.

A variety of approaches tailored to specific problems exist that fundamentally differ from the ones presented in this work. For example, large drops in a continuous coefficient function, originating from the discretization with a finite element mesh, may be resolved by an adaptive mesh refinement; see, e.g., [SGB+10]. For the simulation of dual-phase steel, homogenization methods can be used; see, e.g., [BO83; HW97; HWC99; KBB01; EE03; BBO04; EEL+07; EH09; HL10; Sch14; EF18; KLU+20; Ura20]. For heterogeneous elliptic problems, standard domain decomposition methods are efficient solvers if subdomain boundaries are aligned with the boundaries of patches on which the coefficient function is constant; cf. [DSW96; SBG96; TW05; GLS07; DKW08a; DKW08b].

The aforementioned examples make substantial assumptions that are generally not satisfied in practice. Contrary, the algorithms discussed in this work require only minor assumptions on the coefficient function. This versatility, however, comes at an additional computational cost in the setup phase and also—because of a larger coarse space—during the iteration.

In this thesis, we present coarse spaces that take the underlying heterogeneities into account to obtain a preconditioned system whose condition number can be controlled by the user; these types of coarse spaces are denoted adaptive and robust. Specifically, we will prove condition number bounds of the type

$$\kappa_2(M^{-1}K) \leq C\left(1 + \frac{1}{tol}\right),$$

where K is the stiffness matrix, M^{-1} the preconditioner, tol a user-prescribed tolerance, and C a constant that is independent of the contrast of the coefficient function and of the typical mesh parameters H (coarse mesh resolution) and h (fine mesh resolution). Furthermore, we will present approaches to reduce the coarse space dimension, the setup cost of a coarse space, and approaches that improve parallel performance.

A substantial number of works have addressed the construction of adaptive coarse spaces for the overlapping Schwarz method. Thereof, many rely on solutions of local generalized eigenvalues problems [GE10b; GE10a; SVZ11; EGLW12; DNSS12; SDH+14a; GLR15; GL17; HKKR18b; HKKR18a; EMR19; HKKR19; HKK+22]. While the core concept is the same for all these coarse spaces, the approaches differ in important aspects, such as the class of admissible partial differential equations, the resulting coarse space dimension, the computational efficiency—which includes parallelizability and the cost for the setup and solution of the local generalized eigenvalue problems—and also the complexity of a parallel implementation.

Nonoverlapping domain decomposition methods are another branch of successful methods for which extensive work has been invested to construct adaptive coarse spaces; see, e.g., [BKK01; MS07; MSŠ12; SR13; PD13; KKR16; KRR16; CW16; Zam16; PD17; KCW17; BPS+17; Küh18; KKR18b; MR18; HKLW20; YDS21].

Nonadaptive coarse spaces that do not incorporate the heterogeneity of the underlying

problem may result in slow convergence. A natural choice for coarse basis functions, which are based on the solution of local problems, are multiscale finite element (MsFEM) functions [HW97; HWC99; EH09]; see, for example, [AH02; GLS07; GS07; Buc13; BIA13; BIA14; GLR15; GL17; BIA18; HKKR18b; EMR19; HL20b]. However, a coarse space based only on MsFEM functions is not robust for arbitrary coefficient functions. To achieve robustness, we must enrich the coarse space with additional functions whose number is a priori unknown and depends on the coefficient function.

Many coarse spaces for heterogeneous problems—for both nonoverlapping and overlapping domain decomposition methods—utilize local generalized eigenvalue problems to adaptively construct coarse functions. In [GE10b; GE10a], Galvis and Efendiev define generalized eigenvalue problems on unions of subdomains that have a subdomain vertex in common, which requires the solution of large eigenvalue problems. To remove coarse functions associated with isolated inclusions of large coefficients in the interior of subdomains, the authors propose a heuristic approach. In [EGLW12], the work of Galvis and Efendiev was generalized to abstract, symmetric positive definite operators; see also [SVZ11].

In [DNSS12; NXDS11; NXD10], Dolean et al. introduce the Dirichlet-to-Neumann coarse space for diffusion problems. It uses smaller generalized eigenvalue problems than in [GE10b; GE10a]; specifically, the domain decomposition interface of each subdomain is associated with a generalized eigenvalue problem. To prove robustness, the authors restrict the class of admissible coefficient functions by assuming a quasi-monotonicity condition on the overlap of the overlapping domain decomposition. However, as the authors mention, this restriction does not appear to be required in practice.

In [SDH+14a; SDH+11], Spillane et al. present the GenEO (generalized eigenproblems in the overlaps) coarse space that defines generalized eigenvalue problems on the overlap of the overlapping domain decomposition; see also [DJN15; BDR+20; JRZ21]. The robustness of the coarse space is proven for arbitrary variations of the coefficient function. In contrast to the Dirichlet-to-Neumann coarse space, however, the eigenvalue problems are larger. Furthermore, and similarly to the Dirichlet-to-Neumann coarse space, finite

element nodes can be associated with multiple eigenvalue problems (cf. table 8.9), which can result in larger-than-necessary coarse space dimensions.

For the spectral, harmonically enriched, multiscale (SHEM) coarse space [GLR15; GL17], Gander, Loneland, and Rahman define generalized eigenvalue problems on a partition of the interface. As a result, each finite element node of the interface is associated with only one eigenvalue problem. The authors consider two-dimensional diffusion problems and partition the interface into subdomain vertices and edges, where the vertices are associated with MsFEM-type functions. On subdomain edges, generalized eigenvalue problems—which are much smaller than those of the Dirichlet-to-Neumann or GenEO coarse spaces—are used to construct coarse functions. A three-dimensional analogue of SHEM was presented by Eikeland, Marcinkowski, and Rahman in [EMR19], where additional eigenvalue problems on subdomain faces are solved. Let us remark that using large eigenvalue problems—as used by the Dirichlet-to-Neumann and GenEO coarse spaces—can also be advantageous, as it may reduce the coarse space dimension.

An approach similar to SHEM—that is based on an interface partition into subdomain vertices and edges (in two dimensions), and which was inspired by the ACMS (approximate component mode synthesis) discretization method [HL10]—was first presented in [HKKR18b] and later used in [HKKR18a; HKKR19; HKK+22; HKKR] as well. In this approach, energy-minimizing extensions are incorporated into the generalized eigenvalue problems to reduce the coarse space dimension.

Based on the insight that the use of larger eigenvalue problems can lead to a reduction in the coarse space dimension, the coarse space AGDSW (adaptive generalized Dryja–Smith–Widlund) from [HKKR19] was modified in [HKK+22] to allow for more general partitions of the interface. All adaptive coarse spaces presented in this work make use of local Neumann stiffness matrices and are, thus, denoted as nonalgebraic; algebraic methods rely only on the fully assembled stiffness matrix.

Let us remark that we do not attempt to give an in-depth comparison of different adaptive coarse spaces, neither of the coarse spaces constructed in this work nor of the ones developed by other authors. Many aspects that are relevant for a thorough comparison

have not been scrutinized and are out of the scope of this thesis. Nevertheless, we include results for a selection of coarse spaces from other authors, namely the coarse spaces GenEO [SDH+14a; SDH+14b], SHEM [GLR15], and the wire basket (here denoted by EMR–WB) and vertex-based (here denoted by EMR–VB) coarse spaces from [EMR19].

Outline: In the following sections of this chapter, we first give a motivational example that justifies the development of adaptive coarse spaces. Subsequently, we introduce the additive overlapping Schwarz method and the (nonadaptive) classical coarse space GDSW (generalized Dryja–Smith–Widlund). To gain an understanding of some important characteristics of coefficient functions and their influence on the convergence speed, we examine numerical results for a selection of coefficient functions in two dimensions.

In chapter 2, we introduce four model problems that will be repeatedly used throughout this work to examine, compare, and confirm the robustness of the constructed coarse spaces.

In chapter 3, a matrix formulation for the construction of the AGDSW coarse space and of four variants is presented. Furthermore, a sample MATLAB code is given for a simple two-dimensional diffusion problem. The chapter is concluded by showing numerical results for three-dimensional diffusion problems.

In chapter 4, a modification of the AGDSW coarse space that allows the use of general types of interface decompositions is given in a matrix formulation; two examples of interface decompositions are defined, resulting in the coarse spaces denoted as RAGDSW and R–WB–AGDSW. We conclude the chapter by giving remarks for an implementation and by providing numerical results.

In the previous chapters, we have relied on a simplified description and understanding of interface decompositions. However, for unstructured domain decompositions, interface decompositions can be highly complex. In chapter 5, we give a technical definition of several types of interface decompositions. We conclude the chapter by showing various statistics for the interface decompositions of the problems in chapter 2.

In chapter 6, we state variational formulations associated with the matrix formulations

in chapters 3 and 4, and we prove a condition number bound of the type

$$\kappa_2(M^{-1}K) \leq C\left(1 + \frac{1}{tol}\right),$$

where K is the stiffness matrix, M^{-1} the preconditioner for the AGDSW or RAGDSW coarse space, tol a user-prescribed tolerance, and C a constant that is independent of the contrast of the coefficient function and of the typical mesh parameters H and h. The chapter is concluded by giving practical considerations related to the condition number: We examine the constants of the condition number bound for various coarse spaces and the domain decompositions of chapter 2. Furthermore, we show how the condition number depends on the fine mesh resolution and on the size of interface components.

In chapter 7, the two coarse spaces OS–ACMS and R–WB–OS–ACMS are introduced, which enforce additional Dirichlet boundary conditions in the generalized eigenvalue problems to achieve a reduction in the coarse space dimension. Numerical results are provided for three-dimensional diffusion problems. Furthermore, we prove a condition number bound for OS–ACMS; the proof for R–WB–OS–ACMS is analogous and provided in appendix A.3. Subsequently, a generalization that encompasses all adaptive coarse spaces in this thesis is given.

In chapter 8, we show numerical results for linear elasticity problems and the meshes, domain decompositions, and coefficient functions defined in chapter 2. Let us note that all numerical results in this thesis are based on a serial implementation using MATLAB 2019a. For comparison, we include some results for the coarse space GenEO from [SDH+14a] and discuss some major differences between GenEO and our coarse spaces in section 8.3.

In chapter 9, we discuss several ideas for reducing the computational cost, which includes the heuristic construction of coarse functions. We conclude this thesis with a summary and ideas for future work in chapter 10.

We remark that, for the convenience of the reader, a notation chapter, which contains an overview of the coarse spaces in this thesis including their acronyms, can be found before the introduction.
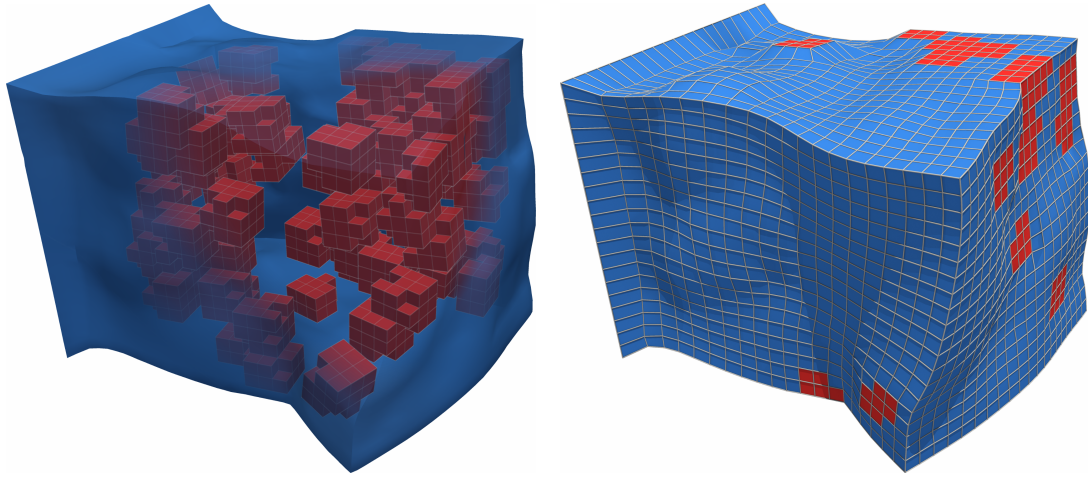
Figure 1.1.: **(Left)** Coefficient function with hard inclusions of $E = 10\,000$ (in red) embedded in a soft material of $E = 1$ (in blue), displayed on the deformed domain for a linear elasticity problem. **(Right)** Surface visualization of the left figure.

1.1. Motivation

We consider a linear elasticity problem on the unit cube with the body clamped on the left, the body force $f = (\,1.5\,,\,0\,,\,0\,)$, and no traction on the remaining boundary. Young's modulus is given by the function in fig. 1.1 (left) and Poisson's ratio ν by 0.4. The domain is discretized with 20^3 trilinear finite elements and subdivided into 4^3 subdomains. The corresponding solution is displayed in fig. 1.1.

We employ different algorithms, which will be discussed in this work, to precondition the conjugate gradient method. During the iteration, a condition number estimate can be obtained with the Lanczos method [Saa03, sect. 6.7.3]. We begin by only using the first level of the additive overlapping Schwarz preconditioner (cf. section 1.3), which is generally a nonscalable domain decomposition method: for an increasing number of subdomains, the number of iterations generally increases as well. An overlap of one layer of finite elements is used.

Even though the model problem is very small, the algorithm requires $3\,504$ iterations

to reduce the relative, unpreconditioned residual by a factor of 10^8; the condition number estimate for the preconditioned operator is given by $1\,854\,519.1$.

Next, we add a second level, i.e., a coarse level to the one-level preconditioner. Here, we use the nonadaptive but scalable GDSW (cf. section 1.4) coarse space. We obtain a condition number estimate of $12\,163.9$ and require 803 iterations to satisfy the convergence criterion. Although this is already a significant improvement over the first level of the preconditioner, it shows the requirement for adaptive coarse spaces.

As we will learn in chapter 3, the GDSW coarse space is in fact the corner stone of the adaptive GDSW method. For this reason, its capabilities in preconditioning heterogeneous problems are fairly good, given that it is a nonadaptive coarse space; see also [DKW08b, table 5.3; Hei16, chapt. 5].

Finally, we use the AGDSW–S coarse space (cf. section 3.3.3) for the coarse level. The tolerance for the selection of eigenfunctions is set to 0.05. We obtain a condition number estimate of 117.0 and require only 104 iterations to meet the convergence criterion. Let us note that the dimensions of the GDSW and AGDSW–S coarse spaces are almost the same with $1\,485$ for the former and $1\,515$ for the latter.

The considered problem is still very small and, as a result, fairly easy to solve. We will introduce much larger and harder problems in chapter 2 for which all tested nonadaptive coarse spaces require more than $2\,000$ iterations to converge.

1.2. Preliminaries

Let $\Omega \subset \mathbb{R}^d$, $d = 2, 3$, be a bounded domain with a Lipschitz boundary, which is decomposed into two disjoint subsets $\partial\Omega_D$ and $\partial\Omega_N$, which are associated with a Dirichlet and a Neumann boundary condition, respectively:

$$\partial\Omega_D \cup \partial\Omega_N = \partial\Omega, \quad \partial\Omega_D \cap \partial\Omega_N = \emptyset.$$

To guarantee the existence of unique solutions to the variational problems given below, we assume that $\partial\Omega_D$ has a positive surface measure. Let a coefficient function $E \in \mathcal{C}^1(\overline{\Omega})$

be given that satisfies

$$0 < E_{\min} \leq E(x) \leq E_{\max}.$$

In the following, we consider two elliptic partial differential equations. First, we consider the scalar stationary diffusion problem: Let $f \in \mathcal{C}^0(\overline{\Omega})$. Then find $u \in \mathcal{C}^2(\overline{\Omega})$ such that

$$
\begin{aligned}
- \operatorname{div}\left(E(x)\nabla u(x)\right) &= f(x) && \forall x \in \Omega, \\
u(x) &= 0 && \forall x \in \partial\Omega_D, \\
\frac{\partial u}{\partial n}(x) &= 0 && \forall x \in \partial\Omega_N,
\end{aligned}
\tag{1.1}
$$

where $n(x)$ is the outer unit normal of $\partial\Omega$.

Second, we consider the displacement $u \in \left(\mathcal{C}^2(\overline{\Omega})\right)^d$ of an elastic, compressible body, deformed by a body force $f \in \left(\mathcal{C}^0(\overline{\Omega})\right)^d$, constrained by a displacement and traction boundary condition. The governing equations of isotropic, linearized elasticity with the respective boundary conditions read

$$
\begin{aligned}
- \operatorname{div}\left(\sigma(u(x))\right) &= f(x) && \forall x \in \Omega, \\
u(x) &= 0 && \forall x \in \partial\Omega_D, \\
\sigma(u(x)) \cdot n(x) &= 0 && \forall x \in \partial\Omega_N,
\end{aligned}
\tag{1.2}
$$

where the linear approximation of the stress tensor is given by Hooke's law as ([Bra07, eq. (1.29)], [Cia88, p. 286])

$$\sigma(u) = 2\mu\varepsilon(u) + \lambda\operatorname{tr}(\varepsilon(u))I;$$

the symmetric, linearized strain tensor $\varepsilon(u)$ is defined as

$$\varepsilon(u) := \tfrac{1}{2}\left(\nabla u + (\nabla u)^T\right).$$

The functions $0 < \lambda(x), \mu(x) \in \mathcal{C}^1(\overline{\Omega})$ are called Lamé parameters.

For the weak formulation of (1.1), we relax the regularity requirements and assume $E, f \in L^2(\Omega)$. We obtain: find $u \in H^1_{0,\partial\Omega_D}(\Omega) := \{\, w \in H^1(\Omega) : w|_{\partial\Omega_D} = 0 \,\}$ such that

$$a_\Omega(u,v) = L(v) \quad \forall v \in H^1_{0,\partial\Omega_D}(\Omega), \tag{1.3}$$

where

$$a_\Omega(u, v) := \int_\Omega E(x)\, \nabla u(x) \cdot \nabla v(x) \,\mathrm{d}x, \quad L(v) := \int_\Omega f(x) v(x) \,\mathrm{d}x. \tag{1.4}$$

By the lemma of Lax–Milgram (cf. [Cia02, theorem 1.1.3], lemma A.1), there exists a unique solution.

For the weak formulation of (1.2), we assume $E \in L^2(\Omega)$ and $f \in (L^2(\Omega))^d$. We define the bilinear form and linear functional

$$a_\Omega(u, v) := \int_\Omega 2\mu(x) \Big(\varepsilon(u(x)) : \varepsilon(v(x)) \Big) \,\mathrm{d}x + \int_\Omega \lambda(x) \Big(\operatorname{div}(u(x)) \operatorname{div}(v(x)) \Big) \,\mathrm{d}x, \tag{1.5}$$

$$L(v) := \int_\Omega f(x) \cdot v(x) \,\mathrm{d}x,$$

and obtain the weak formulation (cf. [Cia88, theorem 6.3-1]): find $u \in \big(H^1_{0,\partial\Omega_D}(\Omega)\big)^d$ such that

$$a_\Omega(u, v) = L(v) \quad \forall v \in \big(H^1_{0,\partial\Omega_D}(\Omega)\big)^d. \tag{1.6}$$

The well-posedness can once again be proven by verifying the assumptions of the lemma of Lax–Milgram; see, e.g., [Cia88, theorem 6.3-5]. Instead of $\lambda(x)$ and $\mu(x)$, we will use the Poisson ratio $0 < \nu < \frac{1}{2}$ and the Young modulus $E(x)$. The following relations hold (cf. [Cia88, p. 128], [Bra07, eq. (1.31)]):

$$\lambda(x) = \frac{E(x)\nu}{(1+\nu)(1-2\nu)}, \qquad \mu(x) = \frac{E(x)}{2(1+\nu)}.$$

We assume that ν is positive and significantly smaller than $1/2$ to model a compressible, elastic body.

For the remainder of this thesis, let $\hat{d}$ be the dimension of the codomain of the solution u; that is, $\hat{d} = d$ for linear elasticity and $\hat{d} = 1$ for the scalar diffusion problem. The assumptions on Ω are changed for the discrete problem: We assume that $\overline{\Omega}$ is—in two dimensions—the closure of the union of disjoint triangles or rectangles and—in three dimensions—of disjoint tetrahedra or cuboids. Furthermore, we assume that Ω is a bounded domain, i.e., a bounded, open, and connected set. The triangulation is denoted by $\tau_h(\Omega)$; the same notation is used for other sets that are unions of finite elements, e.g., $\tau_h(\Omega_i)$ for the triangulation of a subdomain Ω_i. We set $h_T := \operatorname{diam}(T)$ and $h := \min_{T \in \tau_h(\Omega)} h_T$. We assume that $\tau_h(\Omega)$ is shape-regular; i.e., the diameter of an

element over the largest inscribed ball is bounded from above by a constant (for $h_T \to 0$); cf. [TW05, definition B.2; QV08, definition 3.4.1]. For the sake of simplicity, we assume that $h_T \leq 1$, which can be achieved by using a suitable coordinate system.

Above, we have assumed that a zero Dirichlet condition is prescribed on $\partial\Omega_D \subseteq \partial\Omega$ and that $\partial\Omega_D$ has a positive surface measure. For the discretized problem, we further assume that there are enough finite element nodes on $\partial\Omega_D$ such that the discretized problem has a unique solution. In the case of two-dimensional linear elasticity, for example, if only a single node lies on $\partial\Omega_D$, the null space is given by the span of the rotation mode around the node.

Let $V^h(\Omega) \subset (H^1(\Omega))^{\hat{d}}$ denote the conforming finite element space with continuous, piecewise linear, bilinear, or trilinear vector-valued functions. The space that satisfies the zero Dirichlet boundary condition at the finite element nodes of $\partial\Omega_D$ is denoted by $V^h_{0,\partial\Omega_D}(\Omega)$.

In this work, we consider a highly heterogeneous coefficient function E. For the discrete problem, we relax the assumptions on E and, for simplicity, assume that it is piecewise constant and—as mentioned above—positive.

Let the discretization of $a_\Omega(\cdot, \cdot)$ be given by the stiffness matrix K and the discretization of $L(\cdot)$ by the load vector b. For the following, we assume that the zero Dirichlet boundary condition has been incorporated into K by setting those rows and columns of K to unit vectors that correspond to a Dirichlet boundary node. The resulting linear system

$$Ku = b$$

with the symmetric, positive definite stiffness matrix K is solved using the conjugate gradient method, preconditioned with a two-level additive overlapping Schwarz method.

Remark 1.1. *We did not assume Ω to have a Lipschitz boundary in the discrete case. As the bilinear forms defined by (1.4) and (1.5) are symmetric and positive semidefinite on the corresponding finite element space $V^h(\Omega)$, they define a seminorm on $V^h(\Omega)$. Furthermore, the null space is known to consist of constant functions for (1.3) and of (linearized) rigid body modes for (1.6); cf. [TW05, sect. 8.1]. As we enforce sufficient*

Dirichlet boundary conditions, only the zero function is in the null space of the bilinear forms on $V_{0,\partial\Omega_D}^h(\Omega)$. Thus, the stiffness matrices corresponding to the discretizations of the bilinear forms on $V_{0,\partial\Omega_D}^h(\Omega)$ are invertible.

1.3. The Two-Level Additive Overlapping Schwarz Preconditioner

Overlapping Schwarz methods are ubiquitous and often used to accelerate the convergence of a Krylov subspace method; see, e.g., [SBG96; TW05; GLS07; DW09; WC14; SHC15; BDF+15; KC16; HKR16a; Hei16; FQD17; LCYC19; HHK20; HL20b; Hoc20; HKRR20b; HPR22]. We give a description mostly following [TW05, sect. 2.2]; see also [SBG96]. Let us note, however, that the description in [TW05] is not restricted to additive overlapping Schwarz preconditioners.

Let the domain Ω be decomposed into N nonoverlapping subdomains Ω_i that satisfy the same conditions as Ω; that is, $\overline{\Omega}_i$ is the union of elements $T \in \tau_h(\Omega)$ and a connected subset of Ω. Let $H := \max_{i=1,\ldots,N} H_i$, where $H_i := \operatorname{diam}(\Omega_i)$ denotes the subdomain diameter.

We construct a set of overlapping subdomains $\{\Omega_i'\}_{i=1}^N$ from $\{\Omega_i\}_{i=1}^N$ by extending each subdomain Ω_i by k layers of finite elements; cf. fig. 1.2 (left). With each overlapping subdomain Ω_i', we associate a finite element space $V_i \subset V^h(\Omega_i')$ of functions whose support is a subset of Ω_i':

$$V_i := \operatorname{span}\left(\left\{\,\varphi_j|_{\Omega_i'} : 1 \le j \le \dim\left(V_{0,\partial\Omega_D}^h(\Omega)\right) \wedge \operatorname{supp}(\varphi_j) \subseteq \Omega_i'\,\right\}\right),$$

where φ_j are the nodal basis functions of $V_{0,\partial\Omega_D}^h(\Omega)$. Furthermore, we associate a restriction operator R_i with Ω_i', which restricts the degrees of freedom of $V_{0,\partial\Omega_D}^h(\Omega)$ to V_i, and a prolongation operator R_i^T, which extends the degrees of freedom of V_i to $V_{0,\partial\Omega_D}^h(\Omega)$ by zero:

$$R_i \colon V_{0,\partial\Omega_D}^h(\Omega) \to V_i, \quad R_i^T \colon V_i \to V_{0,\partial\Omega_D}^h(\Omega).$$

In a matrix formulation, each row of R_i is associated with a degree of freedom of $V_{0,\partial\Omega_D}^h(\Omega)$ and is set to the corresponding unit vector if the degree of freedom is part of V_i. Then
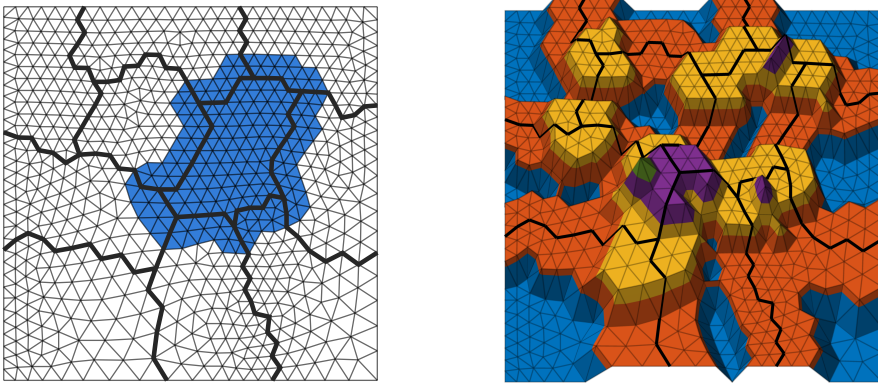
Figure 1.2.: Triangular finite element mesh with 10 subdomains. The domain decomposition interface Γ is represented by thick black lines. **(Left)** An overlapping subdomain Ω'_i with an overlap of two layers of finite elements is highlighted in blue. **(Right)** Visualization for the number of overlapping subdomains a finite element node belongs to. The numbers range from one (blue) to five (green); it is $\hat{N}_c = 5$ (cf. theorem 6.1).

R_i^T is the transpose of R_i. On V_i, we define the bilinear form

$$\tilde{a}_i(u_i, v_i) := a_\Omega(R_i^T u_i, R_i^T v_i) \quad \forall u_i, v_i \in V_i; \tag{1.7}$$

this definition of $\tilde{a}_i(\cdot, \cdot)$ is denoted as using exact local solvers. The corresponding matrix formulation is given by

$$K'_i \colon V_i \to V_i, \quad K'_i = R_i K R_i^T, \quad i = 1, \ldots, N.$$

The one-level additive overlapping Schwarz preconditioner is then defined as

$$M_{\mathrm{OSL1}}^{-1} := \sum_{i=1}^N R_i^T K_i'^{-1} R_i. \tag{1.8}$$

Remark 1.2. *Let us note that—because of the Dirichlet boundary condition on $\partial\Omega'_i$— the degrees of freedom associated with K'_i lie in the interior of Ω'_i. This technical detail is relevant to distinguish the overlap used in the theory and the one used in an*

	$\delta = 1h$	$\delta = 2h$	$\delta = 3h$	$\delta = 4h$	$\delta = 5h$	nodes in $\overline{\Omega}_i$
mean	1 361.8	2 066.4	2 913.9	3 915.9	5 081.8	1 361.8
max	1 419	2 434	3 680	5 165	6 999	1 419

Table 1.1.: Sizes of local overlapping subdomain matrices K_i' for the diffusion equation, the mesh and domain decomposition **(1)** in section 2.1, and different sizes of overlap; cf. tables 2.2 and 3.5. Overlap of k layers of finite elements ($\delta = kh$).

implementation. This is relevant to estimate the cost of solving a linear system with K_i': especially in three dimensions, the size of K_i' can grow quickly if the size of the overlap is increased; cf. table 1.1.

A second, coarse level is generally required to obtain a scalable method; see, e.g., definition 1.3 in [TW05] and the follow-up discussion, [SBG96, sect. 2.1], and [SBG96, sect. 1.4]. However, there are exceptions to this rule; for example, if each subdomain touches the Dirichlet boundary, we can obtain a scalable method without a coarse level. An example is given in table 1.2; see also [CG17; CG18] and [HPR22, sect. 6.2].

In the following, we add a coarse level to the one-level preconditioner. The key ingredient is a coarse space

$$V_0 \subseteq V_{0,\partial\Omega_D}^h(\Omega)$$

that, unlike V_i, is defined globally, i.e., on Ω. We associate a linear interpolation operator

$$R_0^T \colon V_0 \to V_{0,\partial\Omega_D}^h(\Omega)$$

with V_0; in a matrix formulation, each column of R_0^T is given by a coarse basis function in $V_0 \subseteq V_{0,\partial\Omega_D}^h(\Omega)$, which is defined on the fine mesh $\tau_h(\Omega)$. Similarly to before, we define the exact coarse solver

$$\tilde{a}_0(u_0, v_0) \coloneqq a_\Omega(R_0^T u_0, R_0^T v_0) \quad \forall u_0, v_0 \in V_0;$$

the associated coarse matrix

$$K_0 = R_0 K R_0^T$$

N	2	4	8	16	32	64	128	256	512	1 024
(P_D) OSL1	4	6	7	8	7	7	7	7	7	7
GDSW	5	7	10	10	9	9	9	9	9	9
(P_N) OSL1	4	8	16	28	48	88	164	316	617	1 214
GDSW	4	8	13	13	13	13	13	13	12	12

Table 1.2.: Number of iterations required by the one-level additive overlapping Schwarz preconditioner (OSL1) and the GDSW method (section 1.4) for the two-dimensional diffusion problems (P_D) and (P_N) and different numbers of subdomains N: right-hand side of (1.1) is given by $f \equiv 1$; the domain is $\Omega = [0, 1] \times [0, 1/N]$; Ω is decomposed lengthwise into N square subdomains, each containing 64 bilinear finite elements; overlap of three layers of finite elements; (P_D) enforces a zero Dirichlet condition on $\partial\Omega$; (P_N) enforces a zero Dirichlet condition on $\{0\} \times [0, 1/N]$ and a zero Neumann condition on the remaining boundary. Convergence criterion: reduction of the unpreconditioned, relative residual by 10^{-8}.

is the same matrix one would obtain if the stiffness matrix were assembled using the coarse basis functions. The two-level additive overlapping Schwarz preconditioner is defined as

$$M_{\text{OSL2}}^{-1} := R_0^T K_0^{-1} R_0 + M_{\text{OSL1}}^{-1}. \tag{1.9}$$

The specific choice of the coarse space is fundamental in obtaining a robust and scalable method and is one of the main goals of this work. Generally, the condition number of $M_{\text{OSL1}}^{-1} K$ depends on the contrast of the coefficient function—see section 1.5—however, in some cases, the condition can in fact be independent of a large coefficient contrast; see section 1.5.4.

In each iteration of the preconditioned conjugate gradient method, using M_{OSL1}^{-1} defined in (1.8), we need to solve a local problem. By adding a coarse level in (1.9), an additional coarse problem must be solved. Thus—and this is another goal of this work—it is of

interest to use a coarse space that has a small dimension.

Using Lagrangian coarse basis functions on a shape-regular coarse grid, the bound

$$\kappa_2(M_{\mathrm{OSL2}}^{-1}K) \leq C \max_{i=1,\dots,N} \left(\sup_{x,y \in \omega_i} \frac{E(x)}{E(y)} \right) \left(1 + \frac{H}{\delta} \right) \tag{1.10}$$

holds for the diffusion equation (cf. [GLS07]), where $\omega_i := \bigcup_{\overline{\Omega}_k \cap \overline{\Omega}_i \neq \emptyset} \Omega_k$, and δ is the width of the overlap. The bound suggests that the condition number improves with a larger overlap. Indeed, for a small overlap, this is usually the case. However, for a large overlap, we can often observe an increase in the condition number; see section 3.4.2. An explanation may be the fact that the constant $\hat{N}_C$—which is the maximum number of overlapping subdomains $\{\Omega_i'\}_{i=1}^N$ that any finite element node $x^h \in \overline{\Omega}$ can belong to; see fig. 1.2 (right)—influences the condition number bound and increases with δ; see theorem 6.1.

Remark 1.3. *The prolongation R_i^T of the additive overlapping Schwarz preconditioner can hinder its performance; cf. [EG03]. The restricted additive Schwarz method may be used for an improved convergence; cf. [CS99; TW05; SBG96]. However, the preconditioned operator is not symmetric anymore with respect to the a-inner product. As a result, we cannot use the conjugate gradient method but need to resort to, for example, GMRES [Saa03]. Moreover, our convergence analyses in chapter 6 and sections 7.4 and 7.5 are based on the symmetry of the preconditioned operator and are, thus, not valid for the restricted additive Schwarz method.*

1.4. The Generalized Dryja–Smith–Widlund Coarse Space

In this section, we give a description of the generalized Dryja–Smith–Widlund (GDSW) coarse space from [DKW08b; DKW08a; DW09] for the two-level overlapping Schwarz method. It can be regarded as an extension of prior work by Dryja, Smith, and Widlund in [DSW94]. The first results were published in a conference paper [DKW08c] for the 17th International Conference on Domain Decomposition Methods in 2006.

We consider a diffusion problem or compressible linear elasticity, where the coefficients are constant on the subdomains. In [DKW08b], the authors prove the condition number

bound

$$\kappa(M_{\mathrm{GDSW}}^{-1}K) \le C \left(1 + \tfrac{H}{\delta}\right) \left(1 + \log\left(\tfrac{H}{h}\right)\right)^2$$

for the general case of a two-dimensional domain decomposition by John domains and a shape-regular triangulation. The constant C is independent of H, h, and coefficient jumps between subdomains. Here, H/h is defined as

$$H/h := \max_{i=1,\dots,N} \frac{\mathrm{diam}(\Omega_i)}{\min_{T\in\tau_h(\Omega_i)}\mathrm{diam}(T)}.$$

Similarly, H/δ is defined with respect to the size of the overlap δ. In the case of subdomains with Lipschitz boundaries, we can neglect one of the log-factors. A similar result for such general domain decompositions in a three-dimensional setting is—as of yet—not available.

Let us note that for coefficient functions with jumps along as well as across the subdomain boundaries, as encountered in this work, the constant C generally depends on the coefficient contrast; see, e.g., section 3.4 and chapter 8.

In the following, we construct coarse basis functions that define the columns of the matrix R_0^T. To be consistent with the literature, we use the notation $\Phi := R_0^T$. The GDSW preconditioner is then given by

$$M_{\mathrm{GDSW}}^{-1} = \Phi(\Phi^T K \Phi)^{-1}\Phi^T + \sum_{i=1}^{N} R_i^T K_i'^{-1} R_i.$$

1.4.1. Energy-Minimizing Coarse Spaces

The coarse spaces constructed in this work are spanned by energy-minimizing functions; see (1.12) and cf. [TW05; VSG09, sect. 4.4]. More specifically, the coarse functions are energy-minimizing extensions of functions defined on the domain decomposition interface Γ, where

$$\Gamma := \bigcup_{i\ne j}(\partial\Omega_i \cap \partial\Omega_j) \setminus \partial\Omega_D. \tag{1.11}$$

Let Φ and K be partitioned by the interface nodes $\Gamma^h := \{\, x^h : x^h \in \Gamma \,\}$ and the remaining ones R:

$$\Phi = \begin{pmatrix} \Phi_R \\ \Phi_{\Gamma^h} \end{pmatrix}, \quad K = \begin{pmatrix} K_{RR} & K_{R\Gamma^h} \\ K_{\Gamma^h R} & K_{\Gamma^h\Gamma^h} \end{pmatrix}.$$

We note that the set R includes finite element nodes that lie on the Neumann boundary $\partial\Omega_N$. The energy-minimizing extension of Φ_{Γ^h} to R is then defined by

$$\Phi = \begin{pmatrix} \Phi_R \\ \Phi_{\Gamma^h} \end{pmatrix} = H_\Gamma \Phi_{\Gamma^h}, \quad H_\Gamma := \begin{pmatrix} -K_{RR}^{-1}K_{R\Gamma^h} \\ I_{\Gamma^h} \end{pmatrix}, \tag{1.12}$$

where I_{Γ^h} denotes the identity matrix on Γ^h. As the degrees of freedom on the Neumann boundary are part of R, the extension satisfies a zero Neumann boundary condition on $\partial\Omega_N$. Furthermore, by definition of K, the extension is zero on $\partial\Omega_D$. Let us note that K_{RR} is a block-diagonal matrix composed of the submatrices $K_{RR}^{(i)}$ corresponding to the subdomains. We have

$$K_{RR} = \begin{pmatrix} K_{RR}^{(1)} & & \\ & \ddots & \\ & & K_{RR}^{(N)} \end{pmatrix}, \quad K_{\Gamma^h R} = \begin{pmatrix} K_{\Gamma^h R}^{(1)} & \dots & K_{\Gamma^h R}^{(N)} \end{pmatrix}.$$

As a result, the application of H_Γ can be implemented efficiently by concurrent applications of $(K_{RR}^{(i)})^{-1}K_{R\Gamma^h}^{(i)}$.

In this work, we refer to sets of finite element nodes by the superscript h and to a single finite element node by x^h. As for H_Γ, we may drop the superscript h if no ambiguity is introduced.

1.4.2. GDSW Coarse Functions

Let K^N be the stiffness matrix obtained by assembling $a_\Omega(\cdot,\cdot)$ with a Neumann condition on $\partial\Omega$. By "the null space of the problem," we refer to the null space of K^N (which corresponds to that of $\varepsilon(\cdot)$ in the case of linear elasticity). In the case of the diffusion problem, the null space is given by constant functions. In the case of linear elasticity, it is given by rigid body modes; cf. remark 1.4. If the coarse space is able to represent the null space, we can obtain a scalable method ([SBG96; TW05]). Note that, we are not required to fulfill this property in each subdomain; this property has to be satisfied only in subdomains that do not touch the Dirichlet boundary $\partial\Omega_D$.

By definition of the energy-minimizing extension, it is sufficient if the restriction of the null space to the interface is represented. The energy-minimizing extension then
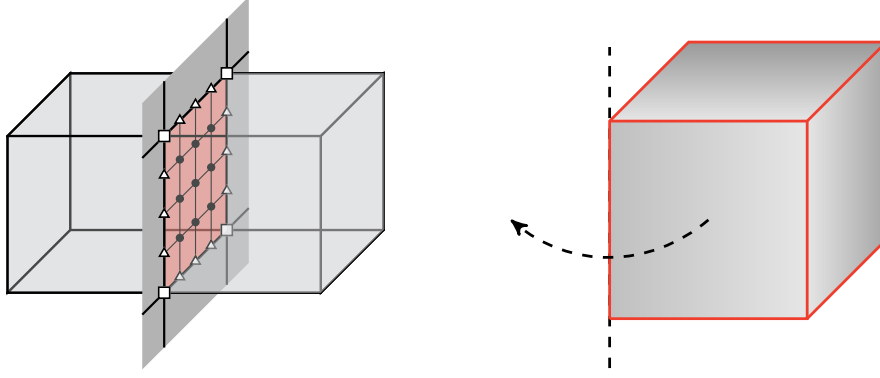
Figure 1.3.: **(Left)** Visualization of two cubic subdomains that share a face (disks), four edges (triangles), and four vertices (squares). **(Right)** Visualization of a rotation of a body around the edge on the left. The edge acts as a hinge and is fixed in place: if the dashed line corresponds to $(0, 0, x_3)$, then r_6 is the zero vector; cf. remark 1.4.

gives a representation of the null space on subdomains that do not touch the Dirichlet boundary $\partial \Omega_D$. However, for large problems, it is not sufficient to use only a small number of coarse functions, in which case scalability is lost; cf. section 6.4.3. Therefore, we decompose the interface into small and disjoint components. The specific type of decomposition is one of the key ingredients for the construction of coarse spaces in this work. Let $\mathcal{P}$ be a partition of Γ^h into disjoint interface components, such that

$$\Gamma^h = \bigcup_{\xi \in \mathcal{P}} \xi.$$

Note that—here and for the rest of this work—ξ denotes an interface component (disjoint from others) that is given by a set of finite element nodes.

The original GDSW method decomposes the interface into subdomain vertices $\mathcal{V}$, edges $\mathcal{E}$, and (in three dimensions) faces $\mathcal{F}$:

$$\Gamma^h = \left(\bigcup_{v \in \mathcal{V}} v \right) \cup \left(\bigcup_{e \in \mathcal{E}} e \right) \cup \left(\bigcup_{f \in \mathcal{F}} f \right).$$

These interface components are degrees of freedoms of Γ^h that are connected and common to the same set of subdomains ([DKW08a, p. 249]).
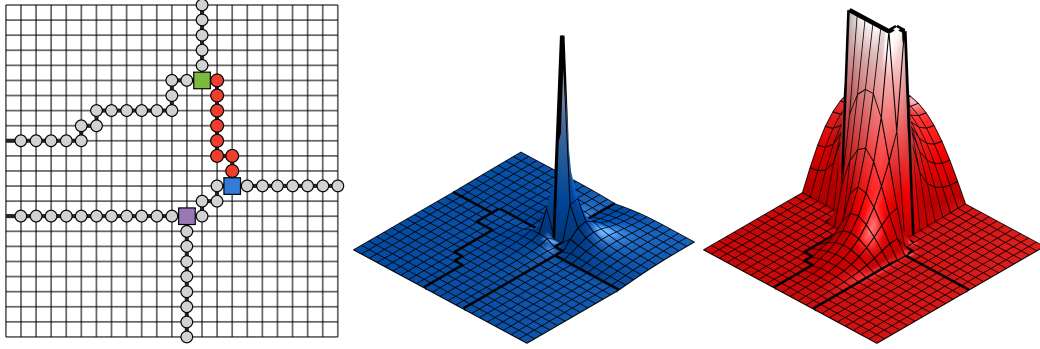
Figure 1.4.: (Compare with [HKK+22, fig. 1]) **(Left)** Decomposition of the interface into 3 vertices (marked with squares) and 7 edges (marked with disks). The Dirichlet boundary $\partial\Omega_D$ is given by the left side of the domain, $\partial\Omega_N$ by the remaining boundary. **(Center/Right)** GDSW basis functions of the diffusion problem with $E \equiv 1$. **(Center)** GDSW vertex function associated with the blue vertex in the left image. **(Right)** GDSW edge function associated with the highlighted red edge in the left image.

In the case of a structured domain decomposition (and for simple unstructured domain decompositions), we can rely on the intuitive understanding of subdomain vertices, edges, and faces; see fig. 1.3 (left). However, for highly unstructured domain decompositions as obtained by, for example, the graph partitioner METIS [KK98], a more rigorous definition is required. Nevertheless, we will postpone this topic until chapter 5 and only note that, for the general case, subdomain vertices, edges, and faces are denoted coarse nodes, edges, and faces, respectively.

Given an interface partition, we restrict the null space to the interface components, extend the trace functions by zero to the remaining interface, and then energy-minimally to the interior. In fig. 1.4, a simple, yet unstructured, domain decomposition of a two-dimensional homogeneous diffusion problem is given. Furthermore, a GDSW vertex function and a GDSW edge function are shown. The Dirichlet boundary $\partial\Omega_D$ is given by the left side of the domain. The effect of the Neumann boundary can be observed in the displayed edge function.

By definition of the GDSW coarse functions, the support of each function is given by the union of subdomains adjacent to the subdomain vertex, edge, or face. As a result—since the stiffness matrix is sparse—the sparsity structure of the coarse matrix $\Phi^T K \Phi$ depends on the domain decomposition.

Let us note that there is no theoretical restriction to using different interface components, apart from the above-mentioned requirement that components should not span large parts of the domain. In a parallel implementation, however, the cost for communication needs to be taken into account. In section 4.1, another type of interface partition will be presented.

The GDSW coarse space has been successfully used in a variety of applications, for example, compressible and almost incompressible linear elasticity [DW09; DW10], incompressible fluid flow [HHK19], fluid-structure interaction [HKR16a; HKR16b], and land-ice simulations [HPR22]. A parallell implementation in the FROSch package (Fast and Robust Overlapping Schwarz) of the Trilinos project [HKRR20a; HKR16a; HKR17; HBH+05] is available. In [HKRR19; HKRR20b; HRR22], the authors have extended the method to a three-level method to be able to solve very large problems for which the coarse problem cannot feasibly be solved using standard solvers.

Finally, let us note that the GDSW coarse space also works well for some problems with heterogeneous coefficient functions; see section 1.5.1. This observation will be motivated in chapter 3.

Remark 1.4. *For linear elasticity, the null space is given by the rigid body modes; cf. [TW05, sect. 8 and A.6.2]. In three dimensions, these are three translations and three linearized rotations. Similarly, in two dimension, there are two translations and one linearized rotation. In three dimensions, a basis for the translations is given by the canonical unit vectors of $\mathbb{R}^3$. The complete set of basis functions $r_1, \ldots, r_6$ is given by*

$$\left\{ \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}, \begin{pmatrix} 0 \\ x_3 \\ -x_2 \end{pmatrix}, \begin{pmatrix} x_3 \\ 0 \\ -x_1 \end{pmatrix}, \begin{pmatrix} x_2 \\ -x_1 \\ 0 \end{pmatrix} \right\},$$

on each finite element node. For the linearized rotations in two dimension, we set $x_3 = 0$,

remove the third coordinate, and obtain the rigid-body-mode basis vectors

$$r_1 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \quad r_2 = \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \quad r_3 = \begin{pmatrix} x_2 \\ -x_1 \end{pmatrix}.$$

Note that in practice, linearized rotations are usually modified by moving the origin of the rotation close to the considered component. Specifically, $x_i - x_i^0$ replaces x_i, where x_i^0 could be, for example, given by the geometric center of the component nodes.

The number of basis functions depends on the interface component that the null space is restricted to. In the case of a single node, we have all the translations but the linearized rotations are all linearly dependent of the translations. Thus, the null space restricted to the node is given by only the translations. In the case of a straight edge in three dimensions, we have three translations but only two linearized rotations, as rotating the edge around itself like a drill yields no change; cf. fig. 1.3 (right).

1.5. Characteristics of Coefficient Functions in Two Dimensions

Before we give a description and analysis of our coarse spaces in the following chapters, we analyze some simple heterogeneous problems to gain a better understanding of the influence of certain characteristics of coefficient functions and of coarse functions that are required to obtain a robust preconditioner. An adaptive coarse space from section 7.2 will serve as a demonstration. However, let us note that the analysis is valid for all our adaptive coarse spaces. Examples showing the differences between the coarse spaces will be given later.

For the numerical results of this section, we consider two-dimensional scalar diffusion problems on the unit square, subdivided into square subdomains and discretized with bilinear finite elements. The right-hand side of (1.3) is given by $f \equiv 1$. On the domain boundary, we impose a zero Dirichlet condition.

The discretized symmetric, positive definite linear system is solved using the conjugate gradient method, preconditioned with no preconditioner (K), with the first level of the additive overlapping Schwarz preconditioner $(M_{\mathrm{OSL1}}^{-1} K)$, or with a two-level additive
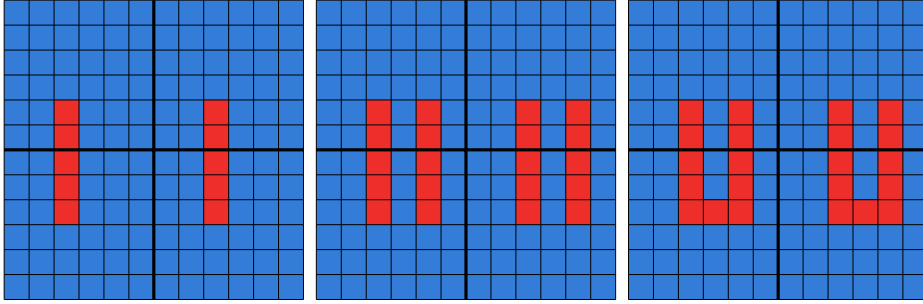
Figure 1.5.: Coefficient functions in two dimensions with a large coefficient of $E = 10^6$ in red and a small coefficient of $E = 1$ in blue on the unit square with 2×2 square subdomains. The interface is indicated by a thick black line, the fine mesh by thin black lines. Each subdomain is discretized with 6×6 bilinear finite elements.

overlapping Schwarz preconditioner for which the coarse space is constructed using a selection of the coarse functions from the OS–ACMS coarse space (here denoted $M^{-1}K$); cf. section 7.2. For the first level, an overlap of one layer of finite elements is used, except in section 1.5.4.

A tolerance of 10^{-8} is chosen for the reduction of the relative, unpreconditioned residual. Unlike in subsequent chapters, the condition number is computed directly and is not estimated by the Lanczos process.

1.5.1. Eigenvalues Associated with Patches of Large Coefficients

The first example is given by the unit square, discretized with 12×12 bilinear finite elements. The unit square is divided into 2×2 square subdomains. The coefficient function consists of two channels of large coefficients that each intersect a horizontal subdomain edge; see fig. 1.5 (left). The preconditioner M^{-1} is constructed using two coarse functions of the OS–ACMS coarse space, one function for each horizontal subdomain edge.

In fig. 1.6, the eigenvalues of the mentioned operators are displayed. To reduce the complexity of fig. 1.6, eigenvalues associated with the Dirichlet boundary (which are equal to one or two) are not displayed. The spectral condition number is given
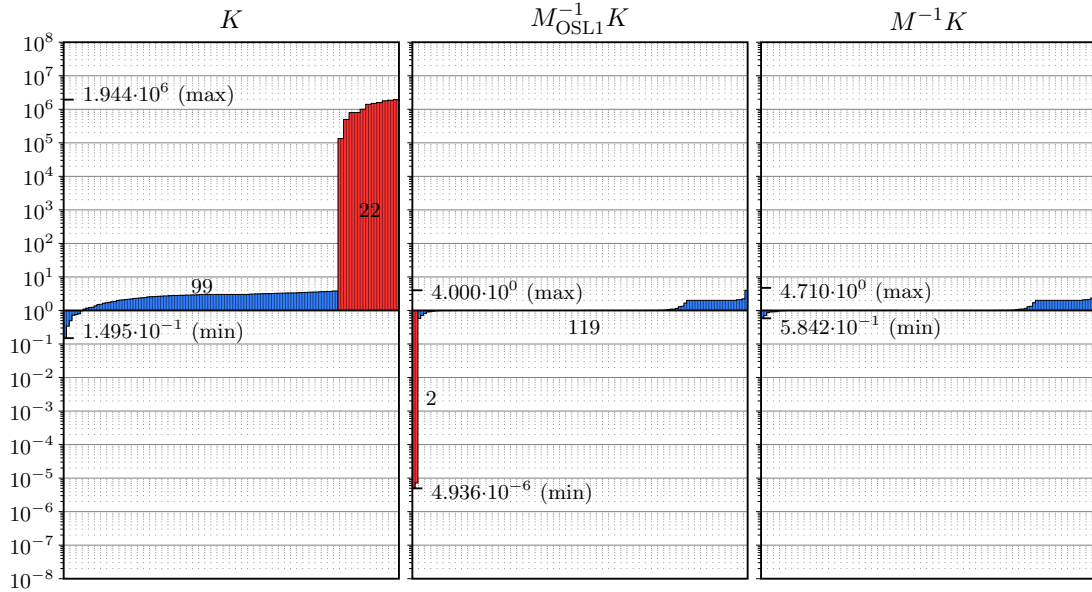
Figure 1.6.: With respect to the 121 interior nodes of the domain: eigenvalues of the stiffness matrix K, the matrix $M_{\text{OSL1}}^{-1}K$, and $M^{-1}K$, where M^{-1} is constructed using two OS–ACMS edge functions. Results correspond to the coefficient function in fig. 1.5 (left). K has 22 eigenvalues larger than and 99 smaller than 10^5. $M_{\text{OSL1}}^{-1}K$ has 2 eigenvalues smaller than and 119 larger than 10^{-5}. $M^{-1}K$ has no "bad" eigenvalues.

by the maximum over the minimum eigenvalue. Thus, according to fig. 1.6, K and $M_{\text{OSL1}}^{-1}K$ are ill conditioned, whereas $M^{-1}K$ is well conditioned. The condition number $\kappa_2(K) = 1.3 \cdot 10^7$ is in line with the bound

$$\kappa_2(K) \leq \frac{C}{h^2} \cdot \frac{E_{\max}}{E_{\min}} \approx C \cdot 10^8, \tag{1.13}$$

where C is independent of h and E; cf. remark 1.5.

Of the 121 eigenvalues of K, 22 are larger than 10^5, while the remaining ones are between 0.1 and 10; cf. [VSM99] and [GE10a, appendix A]. Employing the first level of the additivive overlapping Schwarz preconditioner, the number of extreme eigenvalues is reduced to two; specifically $M_{\text{OSL1}}^{-1}K$ has two eigenvalues smaller than 10^{-5}, while the remaining ones are between 0.1 and 10. The two "bad" eigenvalues of $M_{\text{OSL1}}^{-1}K$

correspond to the two channels of large coefficients intersecting the interface. If there are four channels intersecting the interface as in fig. 1.5 (center), we obtain four bad eigenvalues. If the patches of large coefficients are connected as in fig. 1.5 (right), they act as a single component and we obtain two bad eigenvalues; cf. [GE10b].

The number of bad eigenvalues determines the number of required coarse functions to construct a robust preconditioner. Therefore, for the coefficient functions in fig. 1.5, we require two (left), four (center), and two (right) coarse functions. We note that, to obtain a scalable method, we need to enrich the coarse space with additional functions; cf. sections 1.4 and 6.4.3.

The GDSW coarse space from section 1.4 uses a single coarse function for each edge. According to our analysis above, this should not suffice to obtain a robust preconditioner for the coefficient function in fig. 1.5 (center). Indeed, we obtain a condition number of $336\,413.3$. Contrary, for the coefficient functions in fig. 1.5 (left/right), only a single coarse function per edge is required. For these problems, GDSW is an excellent preconditioner, and we obtain condition numbers of 8.8 (left) and 16.3 (right). This observation will be motivated in chapter 3.

Remark 1.5. *Let K_{hom} denote the stiffness matrix of (1.3) for $E \equiv 1$. Then, by theorem B.32 in [TW05], we have $\kappa_2(K_{\mathrm{hom}}) \leq \frac{C}{h^2}$ for some constant C that is independent of h. Noting that*

$$v^T K v = a_\Omega(v,v) \geq E_{\min}|v|^2_{H^1(\Omega)} = E_{\min}v^T K_{\mathrm{hom}} v,$$
$$v^T K v = a_\Omega(v,v) \leq E_{\max}|v|^2_{H^1(\Omega)} = E_{\max}v^T K_{\mathrm{hom}} v,$$

and

$$\lambda_{\min}(K) = \min_{\|v\|_2=1} v^T K v, \quad \lambda_{\max}(K) = \max_{\|v\|_2=1} v^T K v,$$

for the minimum and maximum eigenvalue of K, the bound of the condition number of K in (1.13) is obtained by

$$\kappa_2(K) = \frac{\lambda_{\max}(K)}{\lambda_{\min}(K)} \leq \frac{E_{\max}}{E_{\min}} \frac{\lambda_{\max}(K_{\mathrm{hom}})}{\lambda_{\min}(K_{\mathrm{hom}})} = \frac{E_{\max}}{E_{\min}}\kappa_2(K_{\mathrm{hom}}) \leq \frac{E_{\max}}{E_{\min}}\frac{C}{h^2}.$$
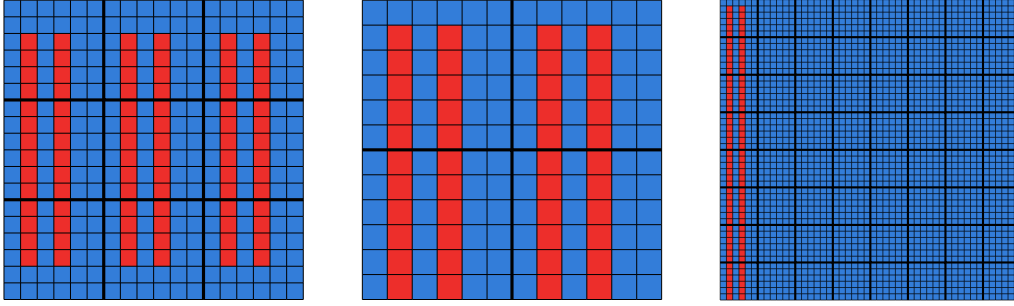
Figure 1.7.: Coefficient functions in two dimensions with a large coefficient of $E = 10^6$ in red and a small coefficient of $E = 1$ in blue on the unit square with 3×3 **(left)**, 2×2 **(center)** and 8×8 **(right)** square subdomains. The interface is indicated by a thick black line, the fine mesh by thin black lines. Each subdomain is discretized with 6×6 bilinear finite elements.

1.5.2. Issues of Scalability

In fig. 1.7 (left), a coefficient function is displayed with 6 channels of large coefficients that each intersect the interface twice. The question is whether this doubles the required amount of coarse functions. The answer is twofold: Disregarding weak scalability, we require 6 coarse functions, one for each channel. This corresponds to the number of bad eigenvalues of $M_{\mathrm{OSL1}}^{-1}K$. However, if we increase the number of subdomains, the condition number deteriorates.

As evidence, we consider the coefficient function in fig. 1.8 (left). The corresponding OS–ACMS coarse function (right) was computed by using the entire interface as the only interface component. We obtain a preconditioned problem with a condition number of 20.7. However, if we increase the number of subdomains to 16×16 and, along with it, the length of the large-coefficient channel as in fig. 1.7 (right), the condition number increases to 544.8, which indicates a loss of scalabilty. By further enriching the coarse space with OS–ACMS vertex functions—which are able to represent the null space—the coarse space dimension increases to 226, but we recover scalability with a condition number of 22.4.

Apart from the scalability issue, it is not feasible to construct our adaptive coarse
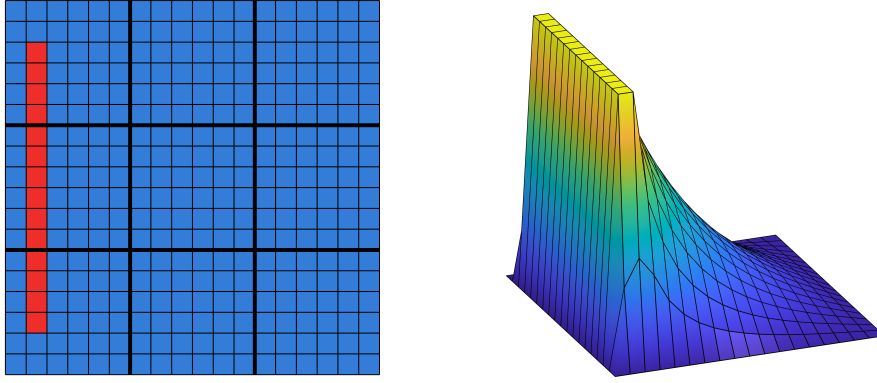
Figure 1.8.: **(Left)** Coefficient function in two dimensions with a large coefficient of $E = 10^6$ in red and a small coefficient of $E = 1$ in blue on the unit square with 3×3 square subdomains. The interface is indicated by a thick black line, the fine mesh by thin black lines. Each subdomain is discretized with 6×6 bilinear finite elements. **(Right)** OS–ACMS coarse function if the entire interface is used as the only interface component.

spaces using large interface components: For the construction, we require the solution of generalized eigenvalue problems, which are as large as the degrees of freedom associated with an interface component. Furthermore, using large components results in a denser coarse matrix. Therefore, in two dimensions, we solve a problem for each subdomain edge or an interface component of comparable size. In fig. 1.9 (left, center), the two relevant OS–ACMS edge functions that we obtain with standard OS–ACMS are shown. As we can see from the sum of the two edge functions (right), it is similar to the function in fig. 1.8 (right) but its support is restricted to the corresponding subdomains. The characteristics of the cofficient function are evident in the computed coarse functions. This observation was used to construct a coarse space heuristically; see section 9.2.

Using smaller interface components comes at the cost of a larger coarse space dimension: for the coefficient function in fig. 1.7 (left), if our adaptive coarse space is based on subdomain edges, we obtain 12 instead of 6 coarse functions (disregarding scalability).
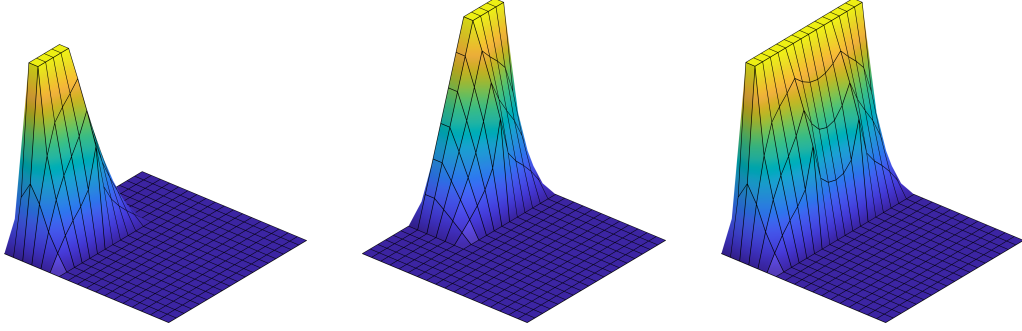
Figure 1.9.: Corresponds to fig. 1.8. **(Left)** OS–ACMS edge function of the horizontal bottom-left coarse edge. **(Center)** OS–ACMS edge function of the horizontal top-left coarse edge. **(Right)** Sum of the two OS–ACMS edge functions.

1.5.3. Patches of Large Coefficients Touching the Dirichlet Boundary

We examine another property of coefficient functions, which was taken into account for the construction of the coefficient functions in chapter 2: a patch of large coefficients touching the Dirichlet boundary can reduce the number of bad eigenvalues. For the coefficient function in fig. 1.7 (center), the largest eigenvalue of the operator $M_{\mathrm{OSL1}}^{-1}K$ is 4 and the smallest one is 0.155. Thus, despite a patch of large coefficients intersecting the interface, using the first level of the additive overlapping Schwarz preconditioner, we obtain a well-conditioned problem (condition number 25.8). This may be explained by the fact that the solution on the patch is essentially known a priori: it is approximately zero.

However, if we increase the number of subdomains, information requires more steps to be passed from the Dirichlet boundary to the interior. For the coefficient function in fig. 1.7 (right), we obtain a condition number of 550.3 for $M_{\mathrm{OSL1}}^{-1}K$. On the other hand, the number of subdomains is substantial, and we know from section 1.3 that the first level of the overlapping Schwarz preconditioner generally does not scale. The results do not improve, however, by using the scalable, nonadaptive coarse space GDSW. For the problem in fig. 1.7 (right), using GDSW, we obtain a condition number of 679.9. Thus, the deterioration of the condition number must be attributed to the heterogeneity of the
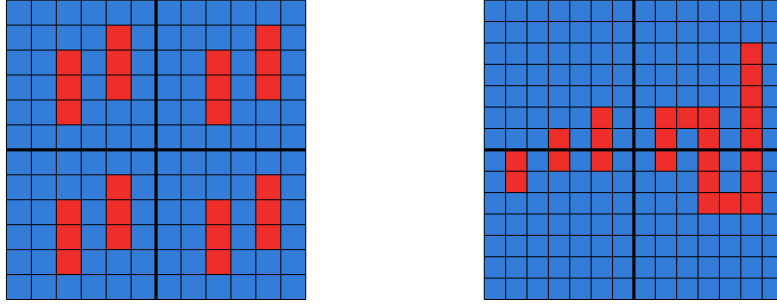
Figure 1.10.: Coefficient functions in two dimensions with a large coefficient of $E = 10^6$ in red and a small coefficient of $E = 1$ in blue on the unit square with 2×2 square subdomains. The interface is indicated by a thick black line, the fine mesh by thin black lines. Each subdomain is discretized with 6×6 **(left)** and 7×7 **(right)** bilinear finite elements, respectively.

problem: by considering a homogeneous coefficient function in fig. 1.7 (right), we obtain a condition number of 141.5 for OSL1 and 9.8 for GDSW.

We conclude that, for our test cases, it is important to decouple patches of large coefficients from the Dirichlet boundary to obtain harder problems, particularly for small problem sizes. However, if the patches of large coefficients span large sections of the domain, scalable nonadaptive coarse spaces are generally not sufficient to obtain robust and scalable preconditioners.

1.5.4. The Influence of the Overlap on Heterogeneous Problems

The size of the overlap has no influence on the construction of the coarse spaces in this work. However, it does have an influence on the performance of the preconditioners for heterogeneous problems. For the following examples, we recall that using an overlap of k layers of finite elements, local overlapping stiffness matrices contain only finite element nodes that extend by $k - 1$ layers; cf. remark 1.2.

As a first example, we consider the coefficient function in fig. 1.10 (left). The patches of large coefficients are completely embedded in the subdomains; we obtain a condition number of 8.6 for OSL1 using an overlap of one layer of finite elements. For the

construction of OSL1, we need to invert local overlapping problems that contain exactly the nodes of the corresponding subdomains. It turns out that, if a patch of large coefficients is embedded in an area that is covered by the local overlapping stiffness matrix, we do not require additional coarse functions.

Using this knowledge, we can infer the size of the overlap that is required for OSL1 to be robust for the coefficient function in fig. 1.10 (right). The leftmost patch of large coefficients is always embedded in the overlapping problem of the bottom-left subdomain; for an overlap of one layer of finite elements (in which case the size of K_i' equals the number of nodes of the nonoverlapping subdomain), three bad eigenvalues remain, one for each of the remaining patches of large coefficients.

The second patch from the left requires the local problem to extend by two layers of finite elements. The third patch is embedded in the overlapping problem of the bottom-left subdomain, using three layers of finite elements, but it is also embedded in the problem of the top-left problem with only two layers of finite elements. Similarly, the minimum number of layers of finite elements for the patch on the right is four. Using an overlap of four layers of finite elements, we have removed all bad eigenvalues and obtain a condition number of 6.5.

2. Coefficient Functions, Meshes, and Domain Decompositions

For comparison and to corroborate the robustness of the discussed coarse spaces, and to further complement the theoretical convergence analysis, we will show numerical results in this work for (1.1) and (1.2), using the following four model problems. Later on—and similarly to chapter 1—we define additional types of problems that are suitable to explain certain mechanics of a coarse space.

The following problems differ in the mesh used, the domain decomposition, and coefficient function E; see [HKKR19; HKK+22], where these identical problems have been used before. An overview of some properties of the model problems in sections 2.1 to 2.4 can be found in table 2.1.

To obtain problems that are numerically difficult to solve and to assess the robustness of the coarse spaces, we rely solely on unstructured domain decompositions, as these are a source of randomness in terms of the coefficient distribution on the interface. For this work, METIS [KK98] was used to obtain unstructured domain decompositions.

We have moreover made sure that—except for model problem **(4)** in section 2.4— patches of large coefficients do not touch the Dirichlet boundary: a problem is usually easier to solve if a patch of large coefficients touches the Dirichlet boundary of the global domain, as this reduces the number of bad eigenvalues; cf. section 1.5.3. The problem in section 2.4 uses a set of 100 randomly generated coefficient distributions. It was constructed to verify the robustness of our coarse spaces and, thus, we have not subjected the coefficient distributions to any restrictions.

We have chosen the overlap as two layers of finite elements for all problems. Note that,

	(1)	**(2)**	**(3)**	**(4)**
boundary conditions	D	N & D	D	D
coefficient values	$\{1, 10^6\}$	$\{1, 10^6, 10^9\}$	$\{1, 10^6\}$	$\{1, 10^6\}$
finite element type	—————	tetrahedron, $\mathcal{P}_1$	—————	
mesh type	struct.	unstruct.	unstruct.	unstruct.
number of finite element nodes	132 651	56 053	588 958	452 522
domain decomposition type	—————	unstruct. (METIS)	—————	
number of subdomains	125	50	100	512
nodes per subdomain (avg.)	1 361.8	1 313.0	6 656.4	1 174.3
overlap (layers of finite elements)	—————	2	—————	
nodes per local overl. problem (avg.)	2 066.0	1 877.4	8 918.7	1 968.8

Table 2.1.: Overview of some properties of model problems **(1)**–**(4)**. Boundary conditions: zero Dirichlet (abbrev. with D), zero Neumann (abbrev. with N), or a combination of the two on different parts of the domain boundary. See remark 1.2 for the definition of the number of nodes of local overlapping problems.

for the solution of the local overlapping problems, this amounts to extracting submatrices from the fully assembled stiffness matrix with an overlap of only one layer of finite elements with respect to the corresponding nodal graph; cf. remark 1.2. On average, the number of nodes of the local overlapping problems is approximately 2 000 for the problems of sections 2.1, 2.2, and 2.4 and almost 9 000 for the problem of section 2.3. In the case of a three-dimensional linear elasticity problem, this results in submatrices three times as large.

Numerical results in section 3.4.2 will show only a moderate improvement in the number of iterations for a growing size of the overlap; see also [HKK+22, table 2]. Furthermore, section 3.4.2 indicates that the sizes of local problems grow quickly with an increase in the size of the overlap; see also table 2.2. For example, for the mesh and domain

		$\delta = 1h$	$\delta = 2h$	$\delta = 3h$	$\delta = 4h$	$\delta = 5h$	$\delta = 6h$
(1)	mean	1 361.8	2 066.0	2 913.9	3 915.9	5 081.8	6 416.6
	max	1 419	2 434	3 680	5 165	6 999	9 164
(2)	mean	1 313.0	1 877.4	2 634.5	3 589.5	4 746.7	6 108.6
	max	1 403	2 098	3 369	4 965	7 155	9 612
(3)	mean	6 656.4	8 918.7	11 843.2	15 237.7	19 092.1	23 458.5
	max	6 963	9 817	14 086	19 131	24 974	31 634
(4)	mean	1 174.3	1 968.8	3 013.0	4 338.5	5 977.2	7 958.0
	max	1 247	2 256	3 643	5 466	7 749	10 546

Table 2.2.: Average and maximum number of nodes per local overlapping problem (cf. section 1.3) for an increasing size of the overlap (in layers of finite elements); cf. table 1.1. **(1)**–**(4)** corresponds to the respective model problem in sections 2.1 to 2.4.

decomposition **(1)** from section 2.1, increasing the size of the overlap from one to six layers of finite elements results in an increase in the average number of nodes of the local overlapping problems from 1 361.8 to 6 416.6. Owing to the complexity of a direct solver, memory constraints, and the moderate improvement for more generous overlaps, we have opted to always use a small overlap of two layers of finite elements.

In fig. 2.1, the average size of subdomains and local overlapping problems is visualized along with the maximum and minimum. From this, we learn that the maximum and minimum deviate little from the average, which is a favorable property for aspects of parallelization.

2.1. Problem (1): Beams of Large Coefficients

The first model problem is given by an unstructured triangulation (with tetrahedra) of the unit cube and 100 beams of large coefficients ($E = 10^6$), which are embedded
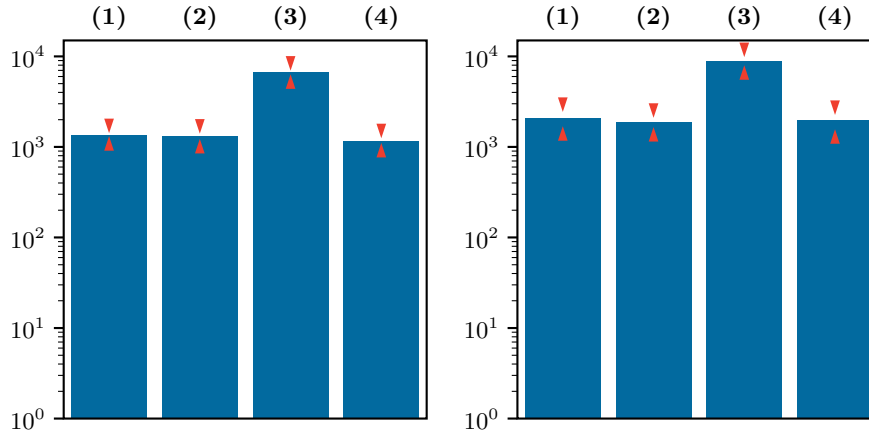
Figure 2.1.: **(Left)** Average number of nodes per subdomain. **(Right)** Average number of nodes per local overlapping problem (cf. section 1.3). **(1)**–**(4)** correspond to sections 2.1 to 2.4. The minimum and maximum numbers (marked by the tips of red triangles) deviate only minimally from the mean.

in a medium with $E = 1$; see fig. 2.2. A zero Dirichlet boundary condition is used on the boundary of the domain. An overview of some properties is given in table 2.1, column **(1)**.

Theoretically—ignoring weak scalability—100 coarse functions are required to remedy the negative influence of the large coefficient contrast, one coarse function per beam. However, as each beam intersects the interface multiple times, the coarse spaces constructed in this work will have a much larger dimension. Furthermore, we may need additional coarse functions to obtain a scalable method.

2.2. Problem (2): Layers of Large Coefficients

The second model problem is an unstructured triangulation with tetrahedra of a complexly shaped domain, four layers of discrete coefficients ($E = 1$ and $E = 10^6$), and a small inclusion of a very large coefficient $E = 10^9$; see fig. 2.3 (top left and right). A zero Dirichlet boundary condition is used on only a small part of the boundary of the domain. On the remainder, a zero Neumann boundary condition is prescribed; see fig. 2.3
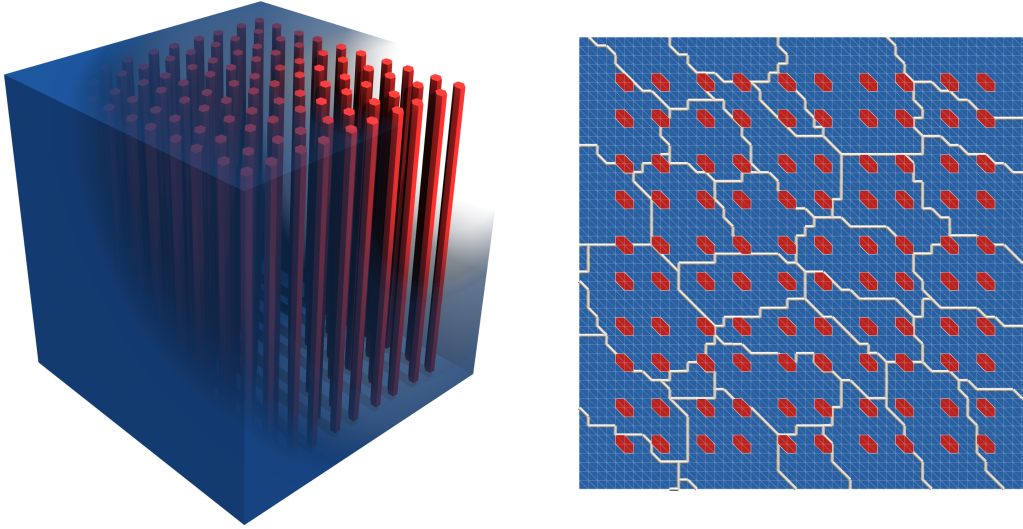
Figure 2.2.: **(Left)** Coefficient function E of problem **(1)** with 100 beams of large coefficients ($E = 10^6$ in red) traversing a medium of $E = 1$ (blue). **(Right)** Cross section of the coefficient function that also shows the fine mesh and subdomain boundaries.

(bottom left). An overview of some properties is given in table 2.1 in column **(2)**.

As there is only a small number of connected patches of large coefficients, we can expect the coarse space dimension to be not much larger than for a homogeneous problem.

2.3. Problem (3): Foamlike Structure of Large Coefficients

The third model problem is an unstructured triangulation with tetrahedra of the unit cube and 27 foamlike structures of large coefficients ($E = 10^6$) embedded in a medium with $E = 1$; see fig. 2.4. The 27 foamlike structures are connected components. If a single and large connected structure were used, the problem would be much easier to solve. In such a case, we would obtain a robust method by using a nonadaptive GDSW-type coarse space. In fig. 2.4, the connected components of the large coefficients are visualized.

A zero Dirichlet boundary condition is prescribed on the boundary of the domain. An overview of some properties is given in table 2.1 in column **(3)**.
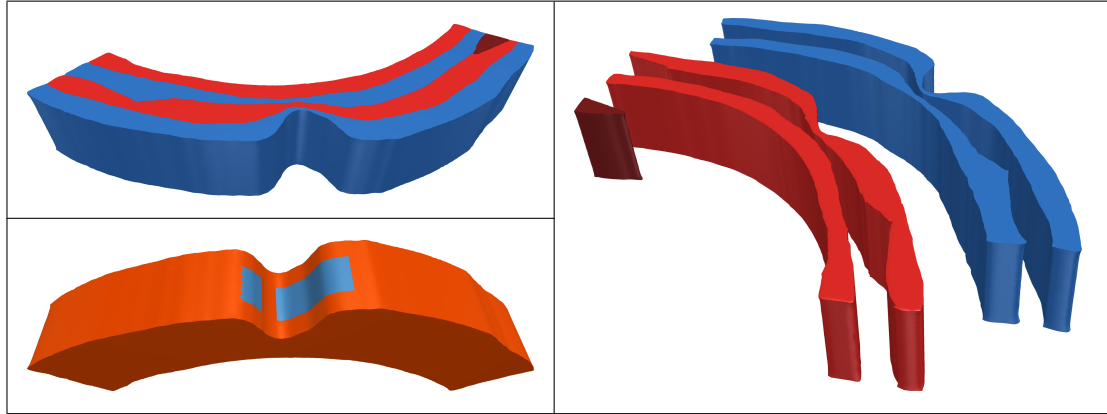
Figure 2.3.: **(Top left)** Coefficient function E of problem **(2)** with multiple layers: $E = 1$ in blue, $E = 10^6$ in light red, $E = 10^9$ in dark red. **(Right)** Visualization of the separated coefficient components. **(Bottom left)** Visualization of the boundary conditions: zero Dirichlet condition on the light blue part and a zero Neumann boundary condition on the remaining part (orange).

2.4. Problem (4): Randomly Distributed Inclusions of Large Coefficients

The fourth model problem is an unstructured triangulation with tetrahedra of the unit cube and randomly distributed inclusions of large coefficients ($E = 10^6$) embedded in a medium with $E = 1$; see fig. 2.5 for a sample. For the numerical results, 100 samples of such coefficient functions are used to obtain averaged results. On average, on 11.08% of finite elements $T \in \tau(\Omega)$, we have $E(T) = 10^6$. This model problem allows us to check the validity of the implementation and the robustness of the methods. A zero Dirichlet boundary condition is used on the boundary of the domain. An overview of some properties is given in table 2.1 in column **(4)**.

Figure 2.4.: **(Top left)** Coefficient function E of problem **(3)** with foamlike structures of large coefficients ($E = 10^6$ in red) inside a medium of $E = 1$ (blue). The coefficient function consists of 27 connected components of large coefficients. **(Top right)** Cross sections of the coefficient function. **(Bottom left)** Connected components of large coefficients shown with different colors. **(Bottom right)** Exploded view of the bottom-left visualization.
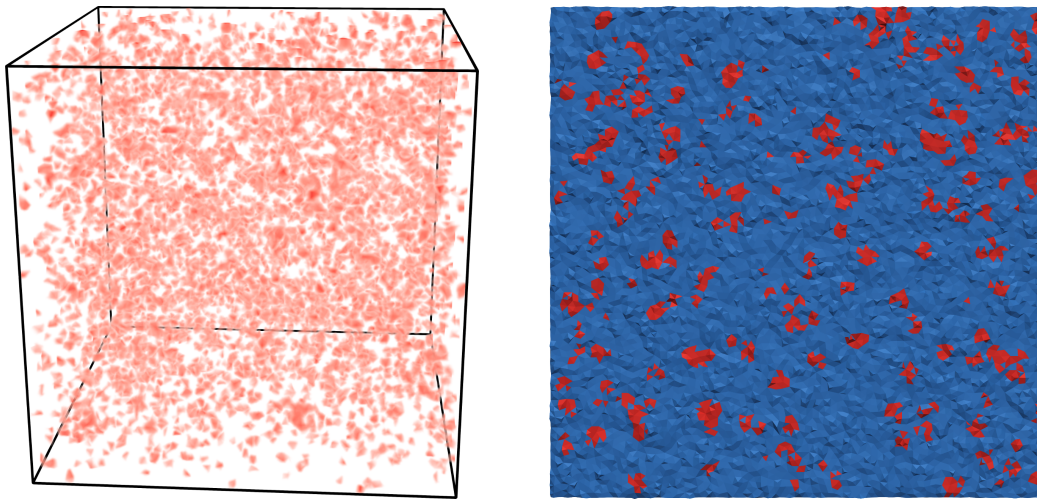
Figure 2.5.: **(Left)** Sample coefficient function E of problem **(4)** with randomly distributed inclusions of large coefficients ($E = 10^6$ in red) inside a medium of $E = 1$. **(Right)** Cross section (one layer of finite elements) of the coefficient function ($E = 1$ in blue).

3. The Adaptive GDSW Coarse Space

In sections 1.1 and 1.5, examples of coefficient functions were given, for which the additive overlapping Schwarz method with the GDSW coarse space is not robust. This motivates the development of coarse spaces that are adaptive with respect to the coefficient function.

In the following, we give a description of the standard adaptive GDSW coarse space and some variants in a matrix formulation. Later on, in section 6.1, the variational analogue is defined and used for the proof of a condition number bound.

The following chapter is based on [HKKR19; HKK+22]. We note that some minor details regarding the coarse space construction were changed for this work. Furthermore, we note that the adaptive GDSW coarse space in [HKKR18a] was improved in [HKKR19]; thus, their construction details differ as well. We refer to all three coarse spaces ([HKKR18a; HKKR19; HKK+22]) as adaptive GDSW-type. The one in [HKK+22] is a reduced-dimension adaptive GDSW coarse space (see chapter 4). The one in [HKKR19] is denoted (standard) adaptive GDSW and is the topic of this chapter; see also [HKL22], where AGDSW is used in the context of nonlinear domain decomposition methods.

3.1. Coarse Space Construction

Similarly to many other adaptive coarse spaces for the overlapping Schwarz method, such as [GE10b; EGLW12; DNSS12; SDH+14a; GLR15; HKKR18b; EMR19], the adaptive GDSW method solves generalized eigenvalue problems of the form

$$S_{\xi\xi}\tau_{*,\xi} = \lambda_{*,\xi}K_{\xi\xi}^{\Omega_\xi}\tau_{*,\xi} \tag{3.1}$$
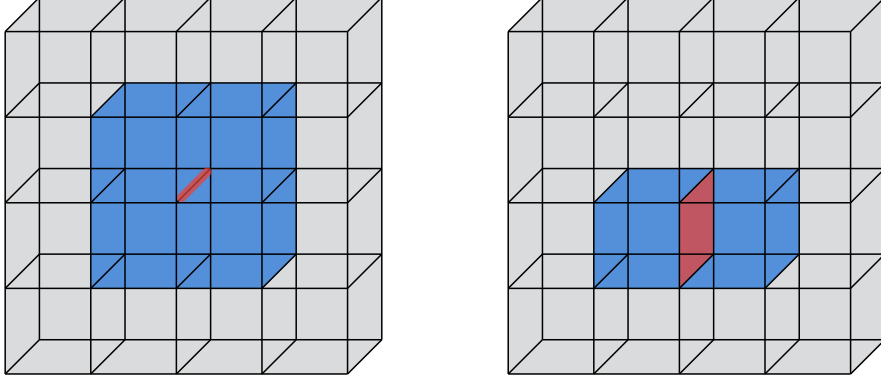
Figure 3.1.: Union of subdomains Ω_ξ adjacent to ξ in blue, where ξ (in red) is a subdomain edge **(left)** and a subdomain face **(right)**, respectively.

on interface components to construct coarse interface functions. Eigenvectors of (3.1) associated with small eigenvalues are selected and extended to the interior of the subdomains to construct coarse basis functions. The matrix $S_{\xi\xi}$ is a Schur complement, and the matrix $K_{\xi\xi}^{\Omega_\xi}$ can essentially be extracted from the fully assembled stiffness matrix K; details are given in the following.

The interface partition $\mathcal{P}$ of the standard adaptive GDSW method is given by the GDSW interface partition, that is, by subdomain vertices, edges, and (in three dimensions) faces. We will give a coarse space description for the three-dimensional case; the two-dimensional case is handled analogously with $\mathcal{F} = \emptyset$. Note that the following description remains valid for interface partitions of unstructured domain decompositions, as given in section 5.3.

Let $\xi \in \mathcal{P}$ be a subdomain edge or face. By Ω_ξ we denote the union of subdomains adjacent to ξ:

$$\Omega_\xi := \overline{\Omega}_\xi \setminus \partial\Omega_\xi, \qquad \overline{\Omega}_\xi := \bigcup_{\substack{i \in \{1,\dots,N\} \\ \partial\Omega_i \cap \xi \neq \emptyset}} \overline{\Omega}_i;$$

cf. fig. 3.1. Note that ξ is a discrete set of finite element nodes and that for, e.g., subdomain edges as in fig. 1.3 (left), the boundary nodes of the edge are not part of ξ.

Let K^{Ω_ξ} be the stiffness matrix that is obtained by assembling $a_{\Omega_\xi}(\cdot,\cdot)$ with a Dirichlet boundary condition on $\partial\Omega_\xi \cap \partial\Omega_D$. We remark that a Dirichlet condition is not enforced

if the construction is carried out according to [HKKR19; HKK+22]. Therein, a Neumann condition is used on the entire boundary of Ω_ξ; see also remark 3.1. K^{Ω_ξ} can be obtained easily if subdomain stiffness matrices are available. We partition K^{Ω_ξ} by the degrees of freedom associated with ξ and those with $\Omega_\xi \setminus \xi$; the latter set is denoted by R. We obtain

$$K^{\Omega_\xi} = \begin{pmatrix} K_{RR}^{\Omega_\xi} & K_{R\xi}^{\Omega_\xi} \\ K_{\xi R}^{\Omega_\xi} & K_{\xi\xi}^{\Omega_\xi} \end{pmatrix},$$

which also defines the matrix $K_{\xi\xi}^{\Omega_\xi}$ on the right-hand side of (3.1). The Schur complement $S_{\xi\xi}$ from (3.1) is defined as

$$S_{\xi\xi} := K_{\xi\xi}^{\Omega_\xi} - K_{\xi R}^{\Omega_\xi}\left(K_{RR}^{\Omega_\xi}\right)^+ K_{R\xi}^{\Omega_\xi}, \tag{3.2}$$

where $\left(K_{RR}^{\Omega_\xi}\right)^+$ is a pseudoinverse of $K_{RR}^{\Omega_\xi}$. The Schur complement originates from the application of an a_{Ω_ξ}-inner product and an energy-minimzing extension; details will be given in section 6.1.

In the case of a diffusion problem, $K_{RR}^{\Omega_\xi}$ is invertible, since it is positive definite. In the case of linear elasticity, however, $K_{RR}^{\Omega_\xi}$ may only be positive semidefinite; see remark 6.5 for details and section 4.4 for remarks on an implementation.

If $\overline{\Omega}_\xi$ does not touch the Dirichlet boundary $\partial\Omega_D$, $S_{\xi\xi}$ and K^{Ω_ξ} are singular; the null space is given by the constant functions or rigid body modes restricted to ξ or Ω_ξ, respectively. In the case of linear elasticity, these matrices can also be singular if $\overline{\Omega}_\xi$ touches $\partial\Omega_D$, in which case linearized rotation modes are part of the null space.

Let the eigenvalues of (3.1) be sorted in nondescending order,

$$0 \le \lambda_{1,\xi} \le \lambda_{2,\xi} \le \cdots \le \lambda_{m,\xi},$$

where m denotes the number of unknowns associated with ξ. We select all eigenvectors $\tau_{*,\xi}$ from (3.1) that correspond to eigenvalues smaller than or equal to a user-prescribed threshold $tol_\xi \ge 0$,

$$\lambda_{*,\xi} \le tol_\xi.$$

The remaining coarse space construction is identical to the GDSW coarse space construction: we extend the selected eigenvectors by zero to the interface nodes Γ^h—let the

extension be denoted by τ_{*,ξ,Γ^h}—and then energy-minimally to the interior to define the coarse functions

$$v_{*,\xi} := H_\Gamma \tau_{*,\xi,\Gamma^h},$$

where H_Γ was defined in (1.12). The columns of the matrix Φ of the Schwarz preconditioner are now given by the selected $v_{*,\xi}$ and by the GDSW vertex functions. Let $V_{\text{GDSW},\mathcal{V}}$ denote the space of GDSW vertex functions. Then we have

$$V_{\text{AGDSW}} = \text{span}\left(\bigcup_{e \in \mathcal{E}} \{\, v_{*,e} : \lambda_{*,e} \leq tol_e \,\} \cup \bigcup_{f \in \mathcal{F}} \{\, v_{*,f} : \lambda_{*,f} \leq tol_f \,\} \cup V_{\text{GDSW},\mathcal{V}}\right).$$

Remark 3.1. *If we assume that for the assembly of K^{Ω_ξ} a Neumann boundary condition is always used on $\partial\Omega_\xi$ (as in [HKKR19; HKK+22]), the Schur complement $S_{\xi\xi}$ is always singular, and its null space is given by the restriction of the null space of K^{Ω_ξ} to ξ. Since $tol_\xi \geq 0$, the null space is always selected for the construction of the coarse space. As a result, if $tol_\xi = 0$ for all subdomain edges and faces, the AGDSW and GDSW coarse spaces are identical. In contrast, we enforce a zero Dirichlet boundary condition on $\partial\Omega_\xi \cap \partial\Omega_D$ for the construction of AGDSW in this work. Therefore, we obtain the GDSW coarse functions only if $\partial\Omega_\xi \cap \partial\Omega_D = \emptyset$.*

Let $tol_\mathcal{P} := tol_\mathcal{E} = tol_\mathcal{F} > 0$ be the smallest tolerance used for the selection of eigenvectors of subdomain edges and faces. In chapter 6, we will prove the condition number bound

$$\kappa_2(M_{\text{AGDSW}}^{-1} K) \leq C\left(1 + \frac{1}{tol_\mathcal{P}}\right), \tag{3.3}$$

where C is independent of H, h, and the contrast of the coefficient function; cf. theorem 6.1. In theorem 6.1, the constant C is expressed explicitly in terms of constants related to the domain decomposition and interface partition. The bound is also valid for the variants of AGDSW that will be introduced in section 3.3; however, it may then contain a dependence on a constant that stems from the application of an inverse inequality.

Note that, in [HKK+22, theorem 11.5], slightly different constants were obtained. Similarly, for a diffusion problem, the constants in [HKKR19, corollary 6.6] differ.

Figure 3.2.: **(Top left)** Mesh with $2 \cdot 20^2$ triangles, two subdomains, a coefficient function of $E = 10^6$ (in red) and $E = 1$ (in blue). Subdomain edge marked with a thick black line. Dirichlet boundary in purple and Neumann boundary in green. **(Top right)** Finite element solution to the corresponding diffusion problem using $\mathcal{P}_1$ basis functions. **(Bottom)** AGDSW coarse functions corresponding to the eigenvalues $1.4 \cdot 10^{-6}$ **(bottom left)** and $2.2 \cdot 10^{-6}$ **(bottom right)**. The next largest eigenvalue is 0.37.

3.2. Sample Code for AGDSW and a Simple Model Problem

In the following, we show a complete but simplified MATLAB code for an implementation of AGDSW and a scalar diffusion problem. Specifically, we consider the unit square divided into two subdomains, a zero Neumann condition on the right side of the domain and a zero Dirichlet condition on the remaining boundary. The coefficient function is given by two channels of large coefficients intersecting the subdomain edge; see fig. 3.2 for more details, the finite element solution, and the two constructed coarse functions. The overlap is chosen to extend by one layer of finite elements.

We begin by defining the only user input: the tolerance for the selection of eigenvectors.

```
1  tol = 0.01; % tolerance for the selection of eigenvectors
```

A triangulation of the unit square is generated and decomposed into the subdomains $[0, 0.5] \times [0, 1]$ and $[0.5, 1] \times [0, 1]$.

```
2  [x1,x2] = meshgrid(linspace(0,1,21)); x1 = x1(:); x2 = x2(:);
3  tri = delaunay(x1,x2); x = [x1,x2];
4  tri_sd{1} = find(all(x1(tri) < 0.5+eps,2)); % Extract triangles
5  tri_sd{2} = find(all(x1(tri) > 0.5-eps,2)); % of subdomains 1&2.
```

The coefficient function is given by two patches of large coefficients: $E = 10^6$ on $(0.2, 0.8) \times (0.2, 0.3)$ and $(0.2, 0.8) \times (0.7, 0.8)$, and $E = 1$ elsewhere.

```
6  E = ones(size(tri,1),1);
7  E(all(x1(tri) > 0.19,2) & all(x1(tri) < 0.81,2) & ...
8        ( (all(x2(tri) > 0.19,2) & all(x2(tri) < 0.31,2)) | ...
9          (all(x2(tri) > 0.69,2) & all(x2(tri) < 0.81,2)) )) = 1e6;
```

Next, we assemble the nonoverlapping local stiffness matrices (with a Neumann boundary) using $\mathcal{P}_1$ basis functions, set up the global stiffness matrix, and incorporate the zero Dirichlet boundary condition on all boundary nodes except for $\{1\} \times (0, 1)$.

```
10  Ki = cell(2,1);                % subdomain stiffness matrices
11  b  = zeros(size(x,1),1);  % global load vector
12  for i = 1:2
13    gradTref = [-1,-1; 1,0; 0,1]; % <- gradient of P1 basisfn on
14    Ki{i} = zeros(size(x,1));     %    reference triangle Tref
15    for j = 1:size(tri_sd{i},1)
16      T = tri(tri_sd{i}(j),:);    % triangle
17      B_T = [x(T(2),:)'-x(T(1),:)' , x(T(3),:)'-x(T(1),:)'];
18      a = abs(det(B_T));
19      gradT = gradTref / B_T;       % gradient of P1 basis on T
20      Ki{i}(T,T) = Ki{i}(T,T) + E(tri_sd{i}(j))*a/2*(gradT*gradT');
21      b(T) = b(T) + a*[1;1;1]/3*0.5; % rhs: f = 1
```

```
22      end
23  end
24  K = Ki{1} + Ki{2};   % global stiffness matrix
25  D = not((x1 > 0) & (x2 > 0) & (x2 < 1)); % Dirichlet bnd nodes
26  K(D,:) = 0; K(:,D) = 0; K(D,D) = eye(nnz(D)); b(D) = 0; % bnd cond
```

By setting up restriction matrices, we can define the first level of the overlapping Schwarz preconditioner. As an overlap of one layer of finite elements is used, the relevant nodes are given by the subdomain nodes.

```
27  nodesOv_i{1} = find(x(:,1) <= 0.5+eps); % nodes in overlapping
28  nodesOv_i{2} = find(x(:,1) >= 0.5-eps); % subdomains
29  KOv = cell(2,1);   % overlapping subdomain matrices
30  R   = cell(2,1);   % restriction matrices
31  for i = 1:2
32      n = nodesOv_i{i}; ln = length(n);
33      KOv{i} = K(n,n);
34      R{i} = sparse((1:ln)',n,ones(ln,1),ln,size(x,1));
35  end
36  OSL1 = @(x) R{1}'*(KOv{1}\(R{1}*x)) + R{2}'*(KOv{2}\(R{2}*x));
```

The interface partition is given by a single component, the subdomain edge $\{0.5\} \times (0, 1)$. Next, we compute the Schur complement with respect to the subdomain edge and its adjacent subdomains.

```
37  gamma = (abs(x(:,1)-0.5) < eps) & not(D);   % interface
38  edge = gamma;   % edge (here: identical to the interface)
39  K12 = K;   % K12 = K^{Omega_edge}. In general: K12 != K.
40  K12_RR = K12(not(edge),not(edge));
41  K12_eR = K12(edge,not(edge));
42  K12_ee = K12(edge,edge);
43  S_ee = K12_ee - K12_eR*(K12_RR\(K12_eR'));   % Schur complement
```

We then set up and solve a generalized eigenvalue problem and extract eigenvectors cor-

responding to an eigenvalue smaller than the user-prescribed tolerance. The eigenvectors are extended energy-minimally to the interior to obtain coarse functions. The coarse matrix is assembled, and the second level of the preconditioner is set up.

```
44  [V,W] = eig(S_ee,K12_ee);    % solve generalized eigenvalue problem
45  V = V(:,abs(diag(W))<tol);   % select eigenvectors
46  K_RR = K(not(gamma),not(gamma));
47  K_RG = K(not(gamma),gamma);
48  phi = zeros(size(x,1),size(V,2));  % coarse basis functions
49  phi(edge,:) = V;
50  phi(not(gamma),:) = -K_RR\(K_RG*phi(gamma,:));  % energy-min ext
51  K0 = phi'*(K*phi);                              % coarse matrix
52  OSL2 = @(x) OSL1(x) + phi*(K0\(phi'*x));        % AGDSW precond
```

Finally, we solve the preconditioned problem, show a plot of the solution, and compute condition numbers for several operators.

```
53  u = pcg(K,b,1e-6,1000,OSL2);
54  trisurf(tri,x(:,1),x(:,2),u); hold on
55  scatter3(x(edge,1),x(edge,2),u(edge),'ro','filled'); hold off
56  fprintf('Coarse space dimension: %d\n',size(K0,1))
57  fprintf('cond(K)      = %g\n',cond(K))
58  fprintf('cond(OSL1(K)) = %g\n',cond(OSL1(K)))
59  fprintf('cond(OSL2(K)) = %g\n',cond(OSL2(K)))
```

We obtain $\kappa(K) = 1.7{\cdot}10^8$, $\kappa(M_{\mathrm{OSL1}}^{-1}K) = 8.0{\cdot}10^6$, and $\kappa(M_{\mathrm{AGDSW}}^{-1}K) = 33.0$. A plot of the solution and the two constructed AGDSW coarse functions are shown in fig. 3.2.

3.3. Variants of AGDSW

3.3.1. Lumped Stiffness Matrix

Generalized eigenvalue problem (3.1) can be transformed into a standard eigenvalue problem if the diagonal of $K_{\xi\xi}^{\Omega_\xi}$ is used on the right-hand side. The modification is
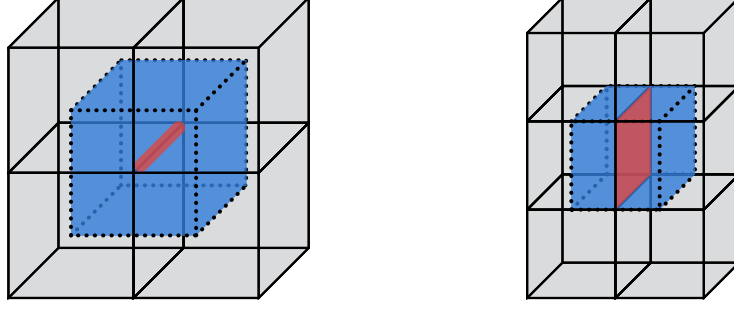
Figure 3.3.: Slab (in blue) surrounding a subdomain edge **(left)** and a subdomain face **(right)**, each in red. Analogue of fig. 3.1.

closely related to a decoupling technique that we need to employ in section 4.2; see also sections 3.3.4 and 6.1.4, and [HKKR18b, sect. 4.3], where the same technique is employed by lumping a scaled mass matrix.

Let $K_{\xi\xi,\text{diag}}^{\Omega_\xi}$ denote the diagonal part of $K_{\xi\xi}^{\Omega_\xi}$, which is identical to the diagonal of the submatrix $K_{\xi\xi}$ of K. Then, we can replace $K_{\xi\xi}^{\Omega_\xi}$ on the right-hand side of (3.1) with $K_{\xi\xi,\text{diag}}^{\Omega_\xi}$. Since $K_{\xi\xi}^{\Omega_\xi}$ is positive definite, we obtain the standard eigenvalue problem

$$\left(K_{\xi\xi,\text{diag}}^{\Omega_\xi}\right)^{-1/2} S_{\xi\xi} \left(K_{\xi\xi,\text{diag}}^{\Omega_\xi}\right)^{-1/2} \hat{\tau}_{*,\xi} = \lambda_{*,\xi} \hat{\tau}_{*,\xi}.$$

The coarse functions are then constructed from the eigenvectors $\tau_{*,\xi} = \left(K_{\xi\xi,\text{diag}}^{\Omega_\xi}\right)^{-1/2} \hat{\tau}_{*,\xi}$ of the associated generalized eigenvalue problem. In some rare situations, the lumped variant can increase the coarse space dimension; cf. section 6.1.1. We indicate that the modification is used by appending $\ell(\text{K})$ to the method's name, for example, AGDSW–$\ell(\text{K})$.

3.3.2. Slabs around Interface Components

In order to reduce the computational cost, we can restrict the domain Ω_ξ of the Schur complement to a slab Ω_ξ^l of l layers of finite elements around the interface component; see fig. 3.3. This approach is denoted economic in [DW16; KRR16; HKKR19].

The only thing that changes for the construction of the coarse space is that

$$S_{\xi\xi}^{\Omega_\xi^l} := K_{\xi\xi}^{\Omega_\xi^l} - K_{\xi R}^{\Omega_\xi^l} \left(K_{RR}^{\Omega_\xi^l}\right)^{+} K_{R\xi}^{\Omega_\xi^l}$$

replaces the Schur complement $S_{\xi\xi}$, where Ω_ξ^l is the union of l layers of finite elements of $\tau_h(\Omega_\xi)$ surrounding ξ. The matrix $K^{\Omega_\xi^l}$ is defined analogously to K^{Ω_ξ} as the assembly of $a_{\Omega_\xi^l}(\cdot, \cdot)$ with a Dirichlet boundary condition on $\partial\Omega_\xi^l \cap \partial\Omega_D$.

We obtain the generalized eigenvalue problem

$$S_{\xi\xi}^{\Omega_\xi^l}\tau_{*,\xi} = \lambda_{*,\xi}K_{\xi\xi}^{\Omega_\xi^l}\tau_{*,\xi}.$$

The rest of the coarse space construction remains the same. We indicate the use of this variant by appending (l) to the method's name, for example, AGDSW(3).

If the corresponding Neumann matrices or element stiffess matrices are available, this variant can reduce the cost of computing the Schur complement. However, a small slab can hinder the detection of patches of large coefficients that are connected and, thus, the coarse space dimension may increase.

We show two examples to demonstrate the effect of the slab size. For more results using the slab variant, we refer to sections 3.4, 4.5, 7.2.7, 7.3.1, appendix B.1, and [HKKR18b; HKKR19].

For comparison, we show results for the coarse space SHEM from [GLR15; GL17]. Similarly to all of our coarse spaces, SHEM solves generalized eigenvalue problems on subdomain edges. However, the left-hand side of an eigenvalue problem is not based on a Schur complement but a stiffness matrix corresponding to a one-dimensional diffusion problem along the respective subdomain edge. A scaled mass matrix is used on the right-hand side of the eigenvalue problem.

We consider a two-dimensional diffusion problem with $f \equiv 1$ for the right-hand side of (1.1) and a mesh and cofficient function that are given by fig. 3.4 (left). On $\partial\Omega$, we prescribe a zero Dirichlet boundary condition. The overlap is given by one layer of finite elements. For the selection of eigenvectors, a tolerance of 0.05 is chosen for all coarse spaces. Let us remark beforehand that the number of iterations and condition numbers are below 12 for all methods.

For the SHEM coarse space, we obtain a dimension of 4, which is identical to the slab variant of AGDSW with $l = 1$. If we increase l to 2, 3, and 4, we obtain coarse space dimensions of 3, 2, and 1. The latter is identical to the standard AGDSW coarse space.
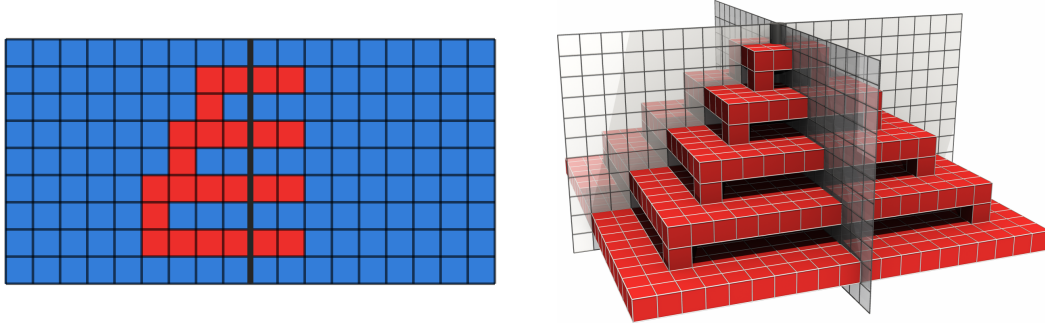
Figure 3.4.: **(Left)** Two subdomains and a coefficient function ($E = 10^6$ in red; $E = 1$ in blue). **(Right)** Connected structure of large coefficients ($E = 10^6$ in red; $E = 1$ elsewhere) given by square plates of decreasing size (bottom to top), connected by pillars. The structure is centered around the common subdomain edge of the four cubic subdomains of the domain. Each subdomain contains 11^3 trilinear finite elements. The coefficient function and visualization are an adaptation of [HKKR19, fig. 7].

Clearly, the coarse space dimension of the slab variant correlates with the structure of the coefficient function that is inside the domain Ω_ξ^l. More details based on the variational description of the method will be given in section 6.1.2.

For the second example, we consider a three-dimensional diffusion problem with a zero Dirichlet boundary condition on $\partial\Omega$. As before, $f \equiv 1$ for the right-hand side of (1.1), and an overlap of one layers of finite elements is used. The coefficient function and further details are given in fig. 3.4 (right) and the corresponding figure caption.

We test the AGDSW coarse space and its slab variant for different sizes l of the slab. To simplify the analysis, we use the full slab for subdomain faces; i.e., $\Omega_\xi^l = \Omega_\xi$ for $\xi \in \mathcal{F}$. The tolerance for the selection of eigenvectors is set to 0.01. We obtain a coarse space dimension of 5 for AGDSW. For the slab variant and $l = 1, 2, 4, 6, 8$, we obtain 9, 8, 7, 6, and 5. Similarly to the two-dimensional case, the coarse space dimension correlates with the number of connected patches of large coefficients that are inside Ω_ξ^l.

We conclude that using the slab variant can reduce the cost to set up a Schur complement, but choosing a small slab can significantly increase the coarse space dimension.
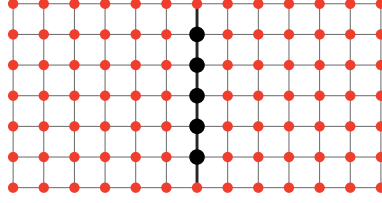
Figure 3.5.: Visualization of $\overline{\Omega}_e \setminus e$ (red disks) and e (black disks) for a subdomain edge in two dimensions.

3.3.3. Sum of Local Schur Complements

In the following, we describe a technique that significantly reduces the computational cost for the setup of the Schur complement in (3.2). The variant is particularly well suited for a parallel setting and also facilitates an implementation considerably; cf. [HKKR19, sect. 7.2; HKK+22, sect. 12]. We remark that the resulting coarse space dimension can be slightly larger than for the standard AGDSW method. However, the numerical results suggest that this increase is fairly small.

Let us consider a subdomain edge e in a two-dimensional setting. By definition (3.2), the Schur complement is defined with respect to the nodes of the open edge; that is, the boundary nodes of the edge are part of $\overline{\Omega}_e \setminus e$. This introduces a weak coupling of the Schur complement between the two subdomains adjacent to the edge; see fig. 3.5. Similarly, in three dimensions, the Schur complement of a subdomain face is only weakly coupled via the boundary nodes. This motivates the replacement of the Schur complement with the sum of Schur complements associated with the individual subdomains. In three dimensions, in the case of subdomain edges, the problem is coupled via subdomain faces. As a result, we can expect the coupling between subdomains to be stronger. However, the numerical results in sections 3.4, 4.5, 7.2.7, 7.3.1, chapter 8, and appendix B show only a moderate increase in the coarse space dimension.

Let ξ be a subdomain edge or face. By $n(\xi)$ we denote the index set of subdomains adjacent to ξ. Then we have

$$\overline{\Omega}_\xi = \bigcup_{k \in n(\xi)} \overline{\Omega}_k.$$

We define Schur complements on $\xi \cap \overline{\Omega}_k$ with respect to Ω_k: Let K^{Ω_k} be the stiffness matrix assembled with $a_{\Omega_k}(\cdot, \cdot)$ using a Dirichlet condition on $\partial\Omega_k \cap \partial\Omega_D$ and a Neumann condition on the remaining boundary of Ω_k. We partition K^{Ω_k} by the degrees of freedom of $\xi \cap \overline{\Omega}_k$ and the remaining ones, R, such that

$$K^{\Omega_k} = \begin{pmatrix} K_{RR}^{\Omega_k} & K_{R\xi}^{\Omega_k} \\ K_{\xi R}^{\Omega_k} & K_{\xi\xi}^{\Omega_k} \end{pmatrix}.$$

The local Schur complements are then defined as

$$S_{\xi\xi}^k := K_{\xi\xi}^{\Omega_k} - K_{\xi R}^{\Omega_k}\left(K_{RR}^{\Omega_k}\right)^+ K_{R\xi}^{\Omega_k}, \quad k \in n(\xi).$$

As before, in the case of a diffusion problem, $K_{RR}^{\Omega_k}$ is positive definite, and $\left(K_{RR}^{\Omega_k}\right)^+$ can be replaced by an inverse. For linear elasticity, $K_{RR}^{\Omega_k}$ may only be positive semidefinite; see remark 6.5 for details and section 4.4 for remarks on an implementation.

To compute the sum of the local Schur complements, we need to map the degrees of freedom of $\xi \cap \overline{\Omega}_k$ to ξ. To this end, let R_{ξ,Ω_k}^T be the required operator. We now define the "sum of local Schur complements"

$$S_{\xi\xi}^S := \sum_{k \in n(\xi)} R_{\xi,\Omega_k}^T S_{\xi\xi}^k R_{\xi,\Omega_k} \tag{3.4}$$

as the replacement of the Schur complement $S_{\xi\xi}$. The modified generalized eigenvalue problem is given by

$$S_{\xi\xi}^S \tau_{*,\xi} = \lambda_{*,\xi} K_{\xi\xi}^{\Omega_\xi} \tau_{*,\xi},$$

where the matrix on the right-hand side is identical to the one in (3.1). This variant can also be applied to the coarse spaces in chapters 4 and 7 and, additionally, in combination with all of the other variants. We will indicate its usage by a trailing S, e.g., AGDSW–S.

The S-variant yields the same condition number bound as in (3.3). For further details and insights, we refer to section 6.1.3, in which a variational description is given.

To demonstrate the effect of the S-variant, we analyze two numerical examples that give different results for AGDSW and AGDSW–S. We consider diffusion problems on a cuboid domain, composed of four cubic subdomains; see fig. 3.6. A zero Dirichlet condition is prescribed on $\partial\Omega$, and the right-hand side of (1.1) is given by $f \equiv 1$.
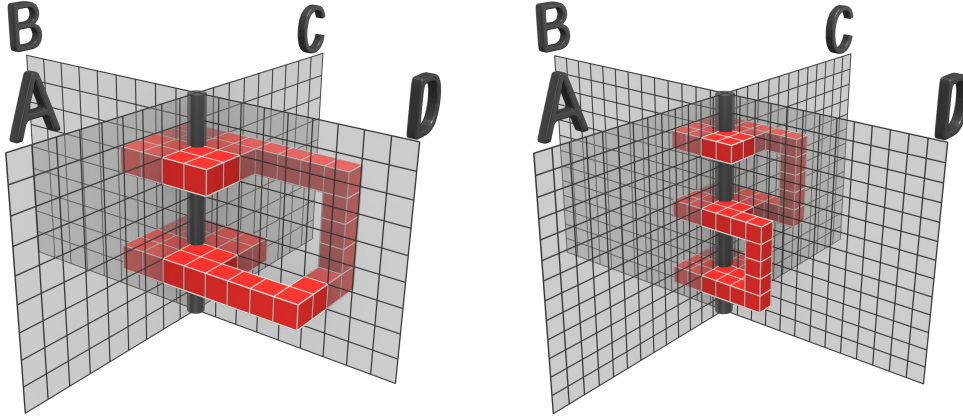
Figure 3.6.: Coefficient functions with $E = 10^6$ (red) and $E = 1$ elsewhere on a domain decomposed into four cubic subdomains. Subdomain faces (labeled A to D) and a subdomain edge are shown.

The structure of large coefficients of the first coefficient function we consider—see fig. 3.6 (left)—is given by two plates that are connected via a corridor. We focus on the edge eigenvalue problem first. Without the corridor, we would obtain two small eigenvalues and, thus, two coarse functions (given a sufficiently large tolerance for the selection of eigenvectors). However, the corridor connects the two plates and—since the entire structure is inside Ω_ξ—we only obtain a single small eigenvalue if AGDSW is used. On the other hand, if AGDSW–S is used as a coarse space, the subdomains are decoupled at the interface, except for the subdomain edge. Thus, we can think of the corridor being cut in half at the subdomain faces. In that case, there is no connection between the two plates anymore. As a result, using AGDSW–S, we obtain two coarse functions.

Let us now focus on the subdomain faces, labeled A to D. As the corridor intersects D, this is the only face whose associated Schur complement holds information about the complete corridor (the corridor is inside the union of subdomains adjacent to D). Therefore, we obtain two coarse functions for each A, B, and C, and only one coarse function for D, using AGDSW. In the case of AGDSW–S, the situation is identical for all faces as the decoupling only takes place on the boundary nodes of the face, which does not influence the present case. To sum up, we obtain a coarse space dimension of 8, using
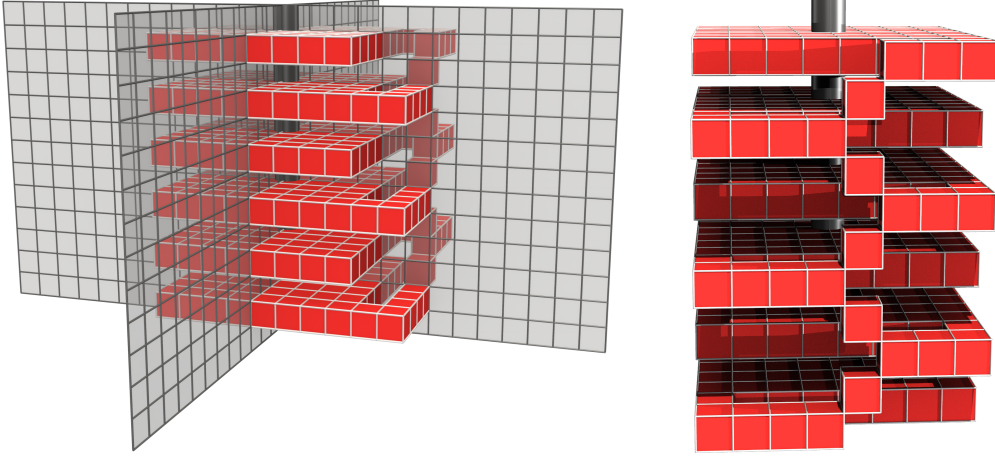
Figure 3.7.: **(Left/Right)** Two visualizations of a coefficient function with $E = 10^6$ (red) and $E = 1$ elsewhere, on a domain decomposed into four cubic subdomains. The subdomain edge is shown as a pole. **(Left)** Subdomain faces are additionally displayed.

AGDSW, and a dimension of 9, using AGDSW–S.

The second coefficient function we consider—see fig. 3.6 (right)—consists of three plates connected via corridors. However, the type of connection is different, and AGDSW and AGDSW–S both detect the presence of only a single connected structure. As before, we consider the edge first. This time, by decoupling the subdomains, the connection between the plates is not lost. The subdomain adjacent to A and D connects the bottom two plates, and the subdomain adjacent to C and D the top two plates. The connection between all three plates is then detected via the subdomain edge. Thus, for this case, AGDSW and AGDSW–S both obtain only one small eigenvalue.

Let us move on to the subdomain faces, for which the situation is once again identical for both coarse spaces. Face A cannot identify the connection between the top and the bottom two plates. Thus, we obtain two coarse functions. Face B cannot identify any connection; we obtain three coarse functions. Face C can detect the connection between the top two plates; we obtain two coarse functions. Face D is aware of all connections; we obtain one coarse function.

We consider a final case to point out that not only the number of connected patches of large coefficients is relevant; see fig. 3.7 for the coefficient function. For the sake of completeness, we remark that—if only eigenvectors are selected that correspond to small eigenvalues—AGDSW constructs a coarse space of dimension 20 and AGDSW–S of dimension 25. However, we would like to focus on the strength of the connectivity of the patch of large coefficients. The entire structure of plates is connected via a "staircase" on the right. Thus, we can expect that only a single coarse function suffices to remove the small eigenvalues. Indeed, if we apply the GDSW approach from section 1.4 but use the entire interface as the only coarse component, we obtain a coarse space dimension of 1 and a condition number of 1 267.8. Of course, this leaves room for improvement. Even for this local structure of large connected coefficients, we seem to require more than one coarse function. The difficulty stems from the fact that the connectivity is fairly weak. It is easy to imagine even weaker connected structures, where each connection is only supported by a single finite element node.

To obtain a condition number below 100, the AGDSW and AGDSW–S coarse spaces require 30 coarse functions (using a tolerance of 0.06 and 0.03, respectively). We conclude that, even though AGDSW–S weakens the detection of connected patches of large coefficients, the knowledge of this connection may sometimes be of little benefit to AGDSW. Furthermore, for complex problems, it may not be sufficient to use only coarse functions that correspond to small eigenvalues (in the order of 10^{-6} for the examples above).

3.3.4. Scaled Mass Matrix

Another variant is defined by using a scaled mass matrix on the right-hand side of generalized eigenvalue problem (3.1). To the best of our knowledge, this does not improve the performance of the preconditioner, as is supported by numerical results in sections 3.4, 7.2.7, 7.3.1, and appendix B. However, we include its description as a variety of other coarse spaces (including the one in chapter 7) use a mass term on the right side of the eigenvalue problem; cf. [DNSS12; GLR15; HKKR18b; EMR19; HKKR].

We define the symmetric, positive definite bilinear form

$$b_\Omega(u,v) := \sum_{T \in \tau_h(\Omega)} \frac{E(T)}{\hat{h}_T^2} \int_\Omega u(x) \cdot v(x) \, \mathrm{d}x, \quad u, v \in V^h(\Omega),$$

where $\hat{h}_T$ may be, for example, the finite element diameter h_T or the radius of the largest insphere of T; cf. [HKKR19, remark 7.4]. For the sake of convenience, and since the triangulation is shape-regular, we use the finite element diameter h_T for the theoretical analysis in chapter 6 but the radius of the largest insphere for numerical results.

From the assembly of $b_\Omega(u,v)$, we obtain the mass matrix M. Let ξ be an edge $e \in \mathcal{E}$ or a face $f \in \mathcal{F}$, and let M be partitioned by the degrees of freedom corresponding to ξ and the remaining ones, R,

$$M = \begin{pmatrix} M_{RR} & M_{R\xi} \\ M_{\xi R} & M_{\xi\xi} \end{pmatrix}.$$

The matrix on the right-hand side of (3.1) is now replaced by $M_{\xi\xi}$. Let us note that this variant can be applied to all our coarse spaces, and it can also be combined with any other variant. We indicate the use of this variant by a trailing M, e.g., AGDSW–M. For details regarding the theory; see section 6.1.4.

As for the original method in section 3.3.1, we can modify the right-hand side by lumping the scaled mass matrix (see also section 6.1.4, [HKKR18b, sect. 4.3], and section 6.1.1): Let $M_{\xi\xi,\mathrm{diag}}$ denote the diagonal part of $M_{\xi\xi}$. Then we can use $M_{\xi\xi,\mathrm{diag}}$ on the right-hand side of (3.1). We indicate that this modification is used by appending $\ell(M)$ to the method's name, for example, AGDSW–$\ell(M)$.

3.4. Numerical Results for Diffusion Problems

In the following, we show numerical results and compare different variants of the standard adaptive GDSW coarse space. We include results for the vertex-based coarse space EMR–VB and the wire basket coarse space EMR–WB from [EMR19]. Furthermore, we show results for the GDSW coarse space, the one-level additive overlapping Schwarz

preconditioner OSL1 without a coarse space, and for the plain conjugate gradient method without any preconditioner, denoted by CG. Note that in the notation chapter, an overview of coarse space acronyms is given.

For the results of this section, we have used the coarse spaces introduced above for more general, unstructured domain decompositions. As mentioned before, the coarse space construction is identical for the unstructured case, but the definition of interface partitions then differs; see chapter 5. The scaling factor $\hat{h}_T$ of the mass matrix variant is set to the radius of the largest insphere of $T \in \tau_h(\Omega)$.

We consider three-dimensional diffusion problems for the meshes and coefficient functions **(2)** and **(3)** in sections 2.2 and 2.3, respectively; results for the other two model problems are given in tables B.1 and B.9. For the right-hand side of (1.1), we use $f \equiv 1$. For all methods, an overlap of two layers of finite elements is chosen.

We use the conjugate gradient method with the convergence criterion

$$\frac{\|r^{(k)}\|_{l^2}}{\|r^{(0)}\|_{l^2}} < 10^{-8},$$

where $r^{(k)}$ is the kth unpreconditioned residual. Let us note that the residual is updated recursively. The initial vector is set to the zero vector and the maximum number of iterations to $2\,000$. A condition number estimate is obtained after the last iteration, using the Lanczos method; cf. [Saa03, sect. 6.7.3].

To compare the coarse spaces, we show results for the condition number $\kappa = \kappa_2(M^{-1}K)$, the number of iterations, the coarse space dimension $\dim V_0$—broken down into the contributions of individual types of interface components $\mathcal{V}$, $\mathcal{E}$, $\mathcal{F}$—and the dimension of the coarse space relative to the size of the stiffness matrix K. For the selection of eigenvectors, we always use the same tolerance for each interface component.

For the first considered model problem from section 2.2, the results in table 3.1 show that the condition numbers of all nonadaptive methods exceed 10^6. The condition number of OSL1 scales with the maximum contrast of the coefficient function (10^6 for problems **(1)** and **(3)**, 10^9 for problem **(2)**); see also table B.1. Nevertheless, the number of iterations of OSL1 and GDSW are comparatively small. This can be explained by the

method	*tol*	it.	κ	dim V_0	($\mathcal{V}$, $\mathcal{E}$, $\mathcal{F}$)	$\frac{\dim V_0}{\text{dof}}$
CG	—	>2 000	$7.0 \cdot 10^9$		—	—
OSL1	—	389	$4.4 \cdot 10^9$		—	—
GDSW	—	125	$3.8 \cdot 10^6$	441	(70, 199, 172)	0.79%
EMR–VB	10^{-6}	66	$1.5 \cdot 10^4$	652	(185, 4, 463)	1.16%
EMR–VB	10^{-4}	49	18.6	746	(185, 4, 557)	1.33%
EMR–WB	10^{-6}	54	$1.2 \cdot 10^4$	960	(97, 862, 1)	1.71%
EMR–WB	10^{-4}	41	14.9	964	(97, 862, 5)	1.72%
AGDSW	10^{-5}	129	$4.2 \cdot 10^5$	483	(70, 200, 213)	0.86%
AGDSW	0.001	49	20.1	500	(70, 215, 215)	0.89%
AGDSW	0.1	49	20.1	500	(70, 215, 215)	0.89%
AGDSW–S	10^{-5}	128	$4.2 \cdot 10^5$	483	(70, 200, 213)	0.86%
AGDSW–S	0.001	49	20.1	500	(70, 215, 215)	0.89%
AGDSW–S	0.1	49	20.1	500	(70, 215, 215)	0.89%
AGDSW–M	10^{-5}	153	$3.9 \cdot 10^5$	485	(70, 201, 214)	0.87%
AGDSW–M	0.001	49	20.1	500	(70, 215, 215)	0.89%
AGDSW–M	0.1	41	13.2	743	(70, 215, 458)	1.33%

Table 3.1.: **(Model problem (2))** Results for the coefficient function in fig. 2.3, the diffusion problem, different methods, and tolerances for the selection of eigenvectors: iteration count, condition number, resulting coarse space dimension, and coarse space dimension over the size of the stiffness matrix. The number of coarse functions associated with subdomain vertices, edges, and faces is given in parentheses.

fact that the problem is fairly small and contains only a small number of disconnected patches of large coefficients. Furthermore, as we will see in chapter 8, the number of iterations will increase considerably if a linear-elastic problem is modeled.

All adaptive coarse spaces attain small condition numbers and numbers of iterations. For each coarse space, we show multiple tolerances for the selection of eigenvectors. For

method	*tol*	it.	κ	$\dim V_0$	($\mathcal{V}$,	$\mathcal{E}$,	$\mathcal{F}$)	$\frac{\dim V_0}{\text{dof}}$
CG	—	>2 000	$5.1 \cdot 10^8$			—		—
OSL1	—	743	$6.5 \cdot 10^6$			—		—
GDSW	—	565	$1.3 \cdot 10^6$	1 593	(328,	750,	515)	0.27%
EMR–VB	10^{-6}	547	$3.6 \cdot 10^4$	2 723	(394,	446, 1 883)		0.46%
EMR–VB	10^{-4}	59	27.1	3 252	(394,	447, 2 411)		0.55%
EMR–WB	10^{-6}	314	$1.1 \cdot 10^4$	5 941	(378, 5 527,		36)	1.01%
EMR–WB	10^{-4}	46	19.1	6 195	(378, 5 527,		290)	1.05%
AGDSW	10^{-4}	549	$5.8 \cdot 10^4$	1 781	(328,	652,	801)	0.30%
AGDSW	0.001	59	30.1	1 814	(328,	678,	808)	0.31%
AGDSW	0.1	51	22.1	1 907	(328,	678,	901)	0.32%
AGDSW–S	10^{-4}	431	$4.9 \cdot 10^4$	1 789	(328,	658,	803)	0.30%
AGDSW–S	0.001	59	29.7	1 815	(328,	679,	808)	0.31%
AGDSW–S	0.1	51	22.1	1 913	(328,	680,	905)	0.32%
AGDSW–M	10^{-4}	168	$2.3 \cdot 10^4$	1 807	(328,	672,	807)	0.31%
AGDSW–M	0.001	59	30.2	1 814	(328,	678,	808)	0.31%
AGDSW–M	0.1	40	12.0	3 890	(328,	684, 2 878)		0.66%

Table 3.2.: **(Model problem (3))** Results for the coefficient function in fig. 2.4, the diffusion problem, different methods and tolerances for the selection of eigenvectors: iteration count, condition number, resulting coarse space dimension, and coarse space dimension over the size of the stiffness matrix. The number of coarse functions associated with subdomain vertices, edges, and faces is given in parentheses.

simple problems, the spectrum of a generalized eigenvalue problem contains a large gap, such that changes of the tolerance may result in only minor differences in the coarse space dimension. For example, table 3.1 shows that AGDSW–M has a coarse space dimension of 500 for a tolerance of 0.001 and an only slightly smaller dimension of 485 for a tolerance of 10^{-5}. And, yet, these 15 coarse functions make the difference as we

obtain a condition number of 386 207.9 without them. If, on the other hand, a tolerance of 0.1 is used, the coarse space dimension increases by almost 50% to 743, yet resulting in only a minor improvement of both the condition number and the number of iterations.

Table 3.1 further shows that the considered variants lead to similar results. However, the spectrum may be shifted in a way that the same tolerance for the selection of eigenvectors can lead to different coarse space dimensions. The results are especially promising for the S-variant as it is suited much better for a parallel implementation.

Before comparing the results with the coarse spaces EMR–VB and EMR–WB, let us note that, generally, a direct comparison of coarse spaces is difficult. A thorough comparison requires not only an efficient, parallel implementation of the coarse spaces considered, but it also depends on the type of problem at hand. Other aspects, such as the complexity of an implementation, can play an important role as well. Hence, a thorough comparison is out of the scope of this work, and we only highlight certain aspects of the coarse spaces.

As we can see from the results in table 3.1, we achieve a significant reduction in the coarse space dimension compared to EMR–VB and EMR–WB. Note that the number of vertex functions of EMR–VB differs from the ones of the other coarse spaces. In [EMR19], the authors have assumed that there exist vertices at the boundaries of subdomain edges. However, here, a Neumann boundary condition is prescribed on most of $\partial\Omega$ for the considered model problem. As a result, many subdomain edges have no or only one incident vertex; as a remedy, we have set dummy vertices. In [HKKR19, table 4], we have given results without this modification, in which case an eigenvalue problem similar to that of AGDSW(1) is obtained for subdomain edges without any incident vertices. For EMR–WB, we note that each node on the wire basket is associated with a coarse function; we have attributed the coarse functions of the wire basket to $\mathcal{V}$ and $\mathcal{E}$ in tables 3.1 and 3.2.

The second considered model problem from section 2.3 supports our previous findings. The problem is larger and has a more complex coefficient function, which is reflected by the larger number of iterations required using the nonadaptive preconditioners. As before,

the differences between AGDSW and AGDSW–S are negligible. If we use a tolerance of 0.1 for AGDSW–M, the coarse space dimension is much larger than for AGDSW and AGDSW–S because of a pronounced shift of the spectrum. We conclude that a slightly smaller tolerance should be used for AGDSW–M to prevent excessively large coarse spaces.

In tables B.1 and B.9, results are given for model problems **(1)** and **(4)**, respectively. In [HKKR19, table 4], results for GDSW and AGDSW–S(3) are given for model problem **(4)** as well. Note, however, that these results are not directly comparable as, here, we use a slightly different interface partition and definition of AGDSW–S.

3.4.1. Slab Variant and Lumped Matrices

Table 3.3 shows results for the slab variant, problems **(2)** and **(3)**, AGDSW, and AGDSW–S; results for problem **(1)** are given in table B.5. For problem **(2)** of section 2.2, the differences between the variants are negligible. However, for problem **(3)** of section 2.3, the coarse space dimension increases by 24.0% for AGDSW and AGDSW–S if a slab extending only one layer of finite elements is used. If the slab extends by three layers of finite elements, the coarse space dimension increases by 4.8% for AGDSW and by 4.9% for AGDSW–S.

Finally, we examine the lumped versions of AGDSW, AGDSW–S, and AGDSW–M in table 3.4. Results for problem **(1)** are given in table B.7. In section 6.1.1, the effect of lumping will be analyzed from a theoretical point of view. This will show that there exist only rare cases for which the coarse space dimension can increase. However, the spectrum may be shifted, which can make direct comparisons difficult. Nevertheless, we expect the differences to be small, which is supported by the results in table 3.4: only the condition number estimate differs minimally in two instances.

3.4.2. Influence of the Size of the Overlap

In chapter 6, we will derive a condition number bound that—unlike the classical estimate (cf. (1.10))—does not include a dependence on the size of the overlap. The results in

E	method	*tol*	it.	κ	$\dim V_0$	($\mathcal{V}$,	$\mathcal{E}$,	$\mathcal{F}$)	$\frac{\dim V_0}{\text{dof}}$
(2)	AGDSW	0.001	49	20.1	500	(70,	215,	215)	0.89%
	AGDSW(3)	0.001	49	20.1	502	(70,	217,	215)	0.90%
	AGDSW(1)	0.001	49	20.0	507	(70,	217,	220)	0.90%
	AGDSW–S	0.001	49	20.1	500	(70,	215,	215)	0.89%
	AGDSW–S(3)	0.001	49	20.1	502	(70,	217,	215)	0.90%
	AGDSW–S(1)	0.001	49	20.0	507	(70,	217,	220)	0.90%
(3)	AGDSW	0.001	59	30.1	1 814	(328,	678,	808)	0.31%
	AGDSW(3)	0.001	58	29.7	1 927	(328,	775,	824)	0.33%
	AGDSW(1)	0.001	57	27.8	2 280	(328,	926,	1 026)	0.39%
	AGDSW–S	0.001	59	29.7	1 815	(328,	679,	808)	0.31%
	AGDSW–S(3)	0.001	58	29.7	1 929	(328,	777,	824)	0.33%
	AGDSW–S(1)	0.001	57	27.8	2 280	(328,	926,	1 026)	0.39%

Table 3.3.: Results for the coefficient functions **(2)** and **(3)** in figs. 2.3 and 2.4, the diffusion problem, different methods, and a tolerance of 0.001 for the selection of eigenvectors: iteration count, condition number, resulting coarse space dimension, and coarse space dimension over the size of the stiffness matrix. If the slab variant is used, the slab width in layers of finite elements is appended in parentheses to the method's name. The number of coarse functions associated with subdomain vertices, edges, and faces is given in parentheses.

table 3.5 show an initial decrease of the number of iterations; see also [HKK+22, table 2]. However, for larger overlaps, the number of iterations can increase; see also [CG18]. For this, we note that the condition number bound (theorem 6.1) depends on $\hat{N}_C$, which is the maximum number of overlapping subdomains a finite element node can belong to.

As the results suggest that—despite the findings in section 1.5.4—the advantage of using a large overlap is at best moderate, and since the sizes of the local overlapping problems grow quickly (see also tables 1.1 and 2.2), we use a moderately sized overlap of two layers of finite elements for all problems defined in chapter 2.

E	method	*tol*	it.	κ	$\dim V_0$	($\mathcal{V}$,	$\mathcal{E}$,	$\mathcal{F}$)	$\frac{\dim V_0}{\text{dof}}$
	AGDSW	0.001	49	20.1	500	(70,	215,	215)	0.89%
	AGDSW–ℓ(K)	0.001	49	20.1	500	(70,	215,	215)	0.89%
(2)	AGDSW–S	0.001	49	20.1	500	(70,	215,	215)	0.89%
	AGDSW–S–ℓ(K)	0.001	49	20.1	500	(70,	215,	215)	0.89%
	AGDSW–M	0.001	49	20.1	500	(70,	215,	215)	0.89%
	AGDSW–ℓ(M)	0.001	49	20.1	500	(70,	215,	215)	0.89%
	AGDSW	0.001	59	30.1	1 814	(328,	678,	808)	0.31%
	AGDSW–ℓ(K)	0.001	59	30.2	1 814	(328,	678,	808)	0.31%
(3)	AGDSW–S	0.001	59	29.7	1 815	(328,	679,	808)	0.31%
	AGDSW–S–ℓ(K)	0.001	59	30.2	1 815	(328,	679,	808)	0.31%
	AGDSW–M	0.001	59	30.2	1 814	(328,	678,	808)	0.31%
	AGDSW–ℓ(M)	0.001	59	30.2	1 814	(328,	678,	808)	0.31%

Table 3.4.: Results for the coefficient functions **(2)** and **(3)** in figs. 2.3 and 2.4, the diffusion problem, different methods, and a tolerance of 0.001 for the selection of eigenvectors: iteration count, condition number, resulting coarse space dimension, and coarse space dimension over the size of the stiffness matrix. If a lumped matrix is used, $\ell(K)$ or $\ell(M)$ is appended to the method's name. The number of coarse functions associated with subdomain vertices, edges, and faces is given in parentheses.

E	method	tol		$\delta = 1h$	$\delta = 2h$	$\delta = 3h$	$\delta = 4h$	$\delta = 5h$
(2)	AGDSW	0.01	it.	59	49	50	51	54
			κ	27.7	20.1	20.9	22.5	25.4
	AGDSW	10^{-5}	it.	180	158	133	151	131
			κ	$4.9 \cdot 10^5$	$4.2 \cdot 10^5$	$4.3 \cdot 10^5$	$4.4 \cdot 10^5$	$4.7 \cdot 10^5$
	size of K_i'		mean	1 313.0	1 877.4	2 634.5	3 589.5	4 746.7
			max	1 403	2 098	3 369	4 965	7 155
(3)	AGDSW	0.01	it.	77	59	52	45	45
			κ	56.6	30.1	21.8	15.8	14.1
	AGDSW	10^{-4}	it.	639	517	497	430	434
			κ	$5.7 \cdot 10^4$	$5.8 \cdot 10^4$	$6.8 \cdot 10^4$	$6.9 \cdot 10^4$	$7.4 \cdot 10^4$
	size of K_i'		mean	6 656.4	8 918.7	11 843.2	15 237.7	19 092.1
			max	6 963	9 817	14 086	19 131	24 974

Table 3.5.: Results for the coefficient functions **(2)** and **(3)** in figs. 2.3 and 2.4, the diffusion problem, AGDSW, two tolerances for the selection of eigenvectors, and different sizes of the overlap $\delta = kh$, where k is the number of layers of finite elements: number of iterations and condition numbers. Average and maximum number of degrees of freedom of the local overlapping stiffness matrices K_i' are given for reference.

4. Reduced-Dimension Adaptive GDSW Coarse Spaces

In the following, we introduce a new type of an adaptive GDSW coarse space. Our goal is to reduce the dimension of the adaptive GDSW coarse space, as in each iteration of the preconditioned Krylov subspace method, a coarse problem needs to be solved. If the dimension of the coarse space grows too large, we either need to reduce its dimension or use parallel solvers that can deal with heterogeneous problems for the coarse solve, e.g., a three-level method where the coarse spaces of the second and third level are constructed adaptively.

The following approach that can reduce the coarse space dimension was initially used in [DW12; DW14; DW17] (see also [DW10] for earlier, related work) and is based on a different partition of the interface. Further works that were inspired by or are based on this approach are, e.g., [HKRW18; HHK20; HKRR20b; Hoc20; HHK21; HKK+22; HPR22; HKRR21; HRR22].

In section 4.1, we define a new type of interface decomposition that can be used to construct a GDSW-type coarse space. Subsequently, in section 4.2, a generalization of the adaptive GDSW coarse space for almost arbitrary interface partitions is presented. The generalization of AGDSW based on the new partition in section 4.1 is denoted by RAGDSW. The description of the coarse space construction uses matrix formulations and closely follows [HKK+22]; the corresponding variational formulations will be given in chapter 6.

The coarse spaces of this chapter essentially yield the same condition number bound as AGDSW in the previous chapter: Let $tol_{\mathcal{P}}$ be the smallest tolerance used for the
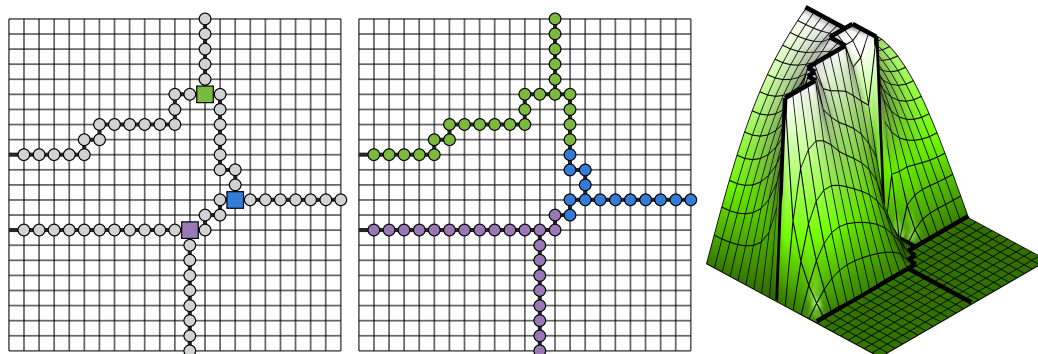
Figure 4.1.: (Compare with [HKK+22, fig. 1]) Analogue of fig. 1.4. **(Left)** Decomposition of Γ^h into 3 subdomain vertices and 7 edges. The Dirichlet boundary $\partial\Omega_D$ is given by the left side of the domain, $\partial\Omega_N$ by the remaining boundary. Vertices are marked with colored squares, edge nodes with gray disks. **(Center)** RGDSW interface decomposition obtained from the decomposition on the left. **(Right)** RGDSW function of diffusion problem with $E \equiv 1$, associated with the interface component at the top (green, center image).

selection of eigenvectors. In chapter 6, we will prove the condition number bound

$$\kappa_2(M_{\mathrm{RAGDSW}}^{-1}K) \leq C\left(1 + \frac{1}{tol_{\mathcal{P}}}\right),$$

where C is independent of H, h, and the contrast of the coefficient function; cf. theorem 6.1.

4.1. Reduced-Dimension GDSW

For a two-dimensional problem, in fig. 4.1 (left), the GDSW interface partition into subdomain vertices and edges is shown. We combine vertices with parts of the adjacent edges to construct a new type of interface partition that results in fewer coarse functions; cf. fig. 4.1 (center). For the example in fig. 4.1, this approach leads to a reduction in the number of interface components from 10 for GDSW to 3 for the new decomposition. In three dimensions, we combine vertices with parts of the adjacent edges and faces;

cf. fig. 4.2. To the new type of interface components, we refer as interface stars. An algorithm for the construction of interface stars, in the case of unstructured as well as structured meshes, is given in section 5.4.

Based on the new interface decomposition, we can define the RGDSW coarse space completely analogously to that of GDSW—see section 1.4—that is, we extend the restriction of the null space by zero from the interface components to the entire interface and then energy-minimally to the interior of the subdomains. An example of an RGDSW coarse function for a homogeneous diffusion problem is given in fig. 4.1 (right); see also fig. 1.4, where the same domain decomposition was used for GDSW.

As the coarse space dimension of GDSW and RGDSW is given by the number of interface components multiplied by the null space dimension of K^N (cf. section 1.4.2), we can obtain a significant reduction in the coarse space dimension, using RGDSW. Let us remark that the precise definition differs from the coarse space in [DW17], but it can be regarded as a variant of the coarse spaces of that paper.

We note that we cannot choose arbitrarily large interface components as this would result in a nonscalable method; see section 6.4.3. Furthermore, components spanning many subdomains are not desirable in a parallel setting as this can significantly increase the communication cost. Therefore, we seek to minimize the number of subdomains adjacent to an interface component ξ: Let $e \in \mathcal{E}$, $f \in \mathcal{F}$, and $\nu \in \mathcal{V}$ be a vertex that is incident to e and f. Then Ω_e and Ω_f are a subset of Ω_ν. Thus, by combining vertices with parts of incident edges and faces, the number of subdomains adjacent to ξ is not increased.

4.2. Adaptive GDSW for a Large Class of Interface Partitions

Our intention is to use the AGDSW generalized eigenvalue problem (3.1) for the new type of interface decomposition introduced in the previous section. On the one hand, this will reduce the number of coarse functions associated with zero eigenvalues (which correspond to RGDSW coarse functions). On the other hand, for larger interface components, it is
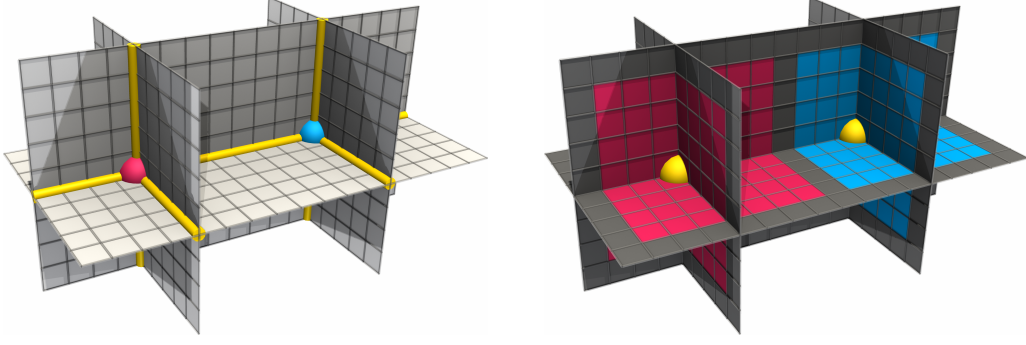
Figure 4.2.: **(Left)** Visualization of the interface of a domain decomposition. Finite element faces in gray, subdomain edges in yellow, and two subdomain vertices in red and blue. **(Right)** Resulting interface stars associated with the subdomain vertices. Finite element faces whose nodes coarse components share are colored in dark gray. See also fig. 5.2.

more likely that connected patches of large coefficients can be detected; cf. section 1.5.1. To obtain a robust coarse space, a modification of the matrix on the right-hand side of (3.1) is required; the reason will be explained in section 6.1.1.

Let us consider the domain decomposition and RGDSW interface partition in fig. 4.3 (left). We decompose each component ξ into subcomponents ξ_i based on subdomain vertices, edges, and (in three dimensions) faces such that

$$\{\xi_i\}_{i=1}^{n_\xi} = \{\, \xi \cap c : c \in \mathcal{V} \cup \mathcal{E} \cup \mathcal{F} \ \wedge \ c \cap \xi \neq \emptyset \,\}, \tag{4.1}$$

where n_ξ is the number of subcomponents of ξ; cf. fig. 4.3 (right). In a two-dimensional setting, we use $\mathcal{F} := \emptyset$ in (4.1). For the example in fig. 4.3, each component ξ is decomposed into five subcomponents ξ_i, where four of them correspond to subdomain edges, and one corresponds to a subdomain vertex. The general idea is the same for unstructured domain decompositions; cf. (5.2).

To define the matrix on the right-hand side of the generalized eigenvalue problem, we use the matrix $K_{\xi\xi}^{\Omega_\xi}$ from (3.1) and decouple it with respect to the subcomponents ξ_i. Let $K_{\xi\xi}^{\Omega_\xi}$ be partitioned by the subcomponents such that

$$K_{\xi\xi}^{\Omega_\xi} = \left(K_{\xi_i\xi_j}^{\Omega_\xi} \right)_{i,j=1}^{n_\xi}.$$
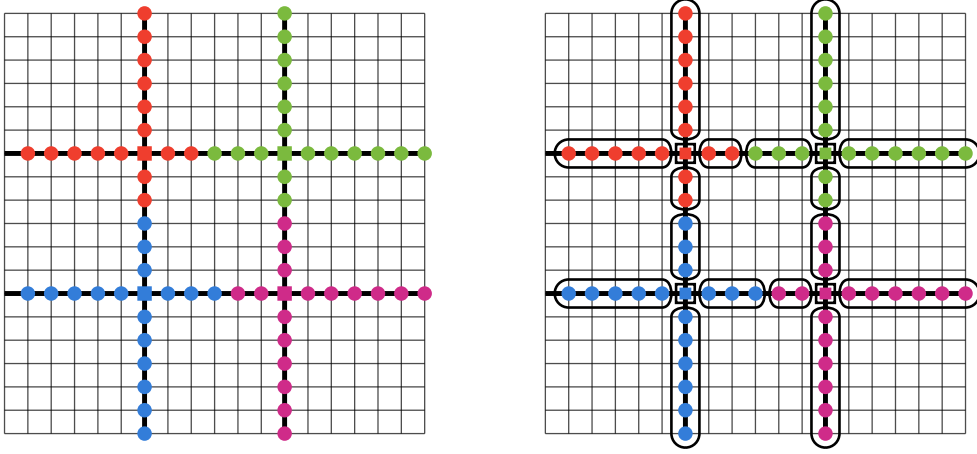
Figure 4.3.: **(Left)** RGDSW interface partition consisting of four components (coarse interface stars); cf. [HKK+22, fig. 1]. Dirichlet boundary condition on the left; Neumann condition on the remaining part of the boundary. **(Right)** Decomposition of the components in the left image into subcomponents; cf. [HKK+22, fig. 2].

Then, we define its replacement for (3.1) by removing all off-diagonal blocks:

$$\widetilde{K}_{\xi\xi} := \operatorname*{blockdiag}_{i=1,\dots,n_\xi}(K_{\xi_i\xi_i}^{\Omega_\xi}).$$

The new generalized eigenvalue problem reads (the Schur complement $S_{\xi\xi}$ from (3.2) remains unaltered)

$$S_{\xi\xi}\tau_{*,\xi} = \lambda_{*,\xi}\widetilde{K}_{\xi\xi}\tau_{*,\xi}. \tag{4.2}$$

The decoupling of $K_{\xi\xi}^{\Omega_\xi}$ is essential to obtain a robust preconditioner: without the block-structure of the matrix, it is possible to construct a case (mesh, domain decomposition, and coefficient function) for which the algorithm fails to obtain a small condition number; see section 6.1.1 for details.

The remainder of the coarse space construction is identical to that in chapter 3: Let the eigenvalues $\lambda_{*,\xi}$ be sorted in nondescending order and the eigenvectors $\tau_{*,\xi}$ accordingly. We select eigenvectors corresponding to an eigenvalue smaller than or equal to a user-prescribed tolerance tol_ξ, extend the selected eigenvectors by zero to Γ^h and

then energy-minimally to the interior of the subdomains. The resulting coarse functions—denoted by $v_{*,\xi}$—define the columns of the matrix Φ of the Schwarz preconditioner. The RAGDSW coarse space is given by

$$V_{\mathrm{RAGDSW}} = \mathrm{span}\left(\bigcup_{\xi \in \mathcal{P}} \{\, v_{*,\xi} : \lambda_{*,\xi} \leq tol_\xi \,\} \right).$$

All variants of adaptive GDSW in sections 3.3.2 to 3.3.4 can be used without modification to construct an RAGDSW coarse space. This includes the variant in section 3.3.4, for which the mass matrix on the right-hand side of the eigenvalue problem does not have to be decoupled with respect to the subcomponents ξ_i of the interface components ξ.

Remark 4.1. *Analogously to remark 3.1, if a Neumann boundary condition is always used on $\partial\Omega_\xi$ for the assembly of K^{Ω_ξ}—as is the case in [HKKR19; HKK+22]—the Schur complement is always singular, and its null space is given by the restriction of the null space of K^{Ω_ξ} to ξ. Since $tol_\xi \geq 0$, the null space is always selected for the construction of the coarse space. As a result, if $tol_\xi = 0$ for all interface components $\xi \in \mathcal{P}$, the RAGDSW and RGDSW coarse spaces are identical. For the construction of RAGDSW in this work, however, we enforce a zero Dirichlet boundary condition on $\partial\Omega_\xi \cap \partial\Omega_D$. Therefore, we obtain the RGDSW coarse functions only if $\partial\Omega_\xi \cap \partial\Omega_D = \emptyset$.*

4.3. An Interface Partition Based on the Wire Basket and Subdomain Faces

RGDSW interface components can be considerably larger than the subdomain faces of GDSW; cf. section 5.7.3. The cost for the setup and solution of the associated generalized eigenvalue problems scales cubically with the size. As a result, the solution may become computationally too expensive in some situations, demanding a domain decomposition with smaller subdomains to obtain smaller interface components or, alternatively, a different interface decomposition. The construction of the RAGDSW coarse space is valid for other types of interface decompositions. Here, we give another example, which can serve as a compromise between AGDSW and RAGDSW.
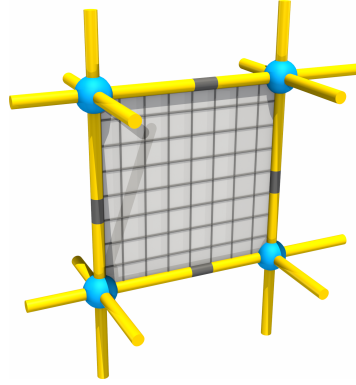
Figure 4.4.: Four wire basket stars (in yellow) and a subdomain face (in gray). Subdomain vertices are marked with blue spheres; finite element edges on the wire basket that belong to multiple stars are colored in dark gray.

For the new type of interface decomposition, we keep the face problems for $f \in \mathcal{F}$ but replace the problems on subdomain edges and vertices with problems on so-called wire basket stars (the wire basket is the union of subdomain vertices and edges). We use the same process to partition the wire basket as we have used for RAGDSW to partition the interface: subdomain vertices are combined with parts of adjacent edges; cf. fig. 4.4. For the construction in the case of unstructured domain decompositions, see section 5.5. By R–WB–AGDSW we denote the resulting adaptive coarse space and by R–WB–GDSW the nonadaptive coarse space. We note that, in two dimensions, the resulting interface decomposition is identical to that of RGDSW.

4.4. Remarks on the Implementation

The classical GDSW coarse space from section 1.4 only requires the fully assembled stiffness matrix K. In contrast, the adaptive coarse spaces of this work—including all variants—require local (Neumann) stiffness matrices that cannot be extracted from K. If the local stiffness matrices are available, however, the corresponding Schur complements can be assembled easily; a sample code was given in section 3.2. Using the S-variant from section 3.3.3, the setup is further simplified since the Schur complements can be

computed by a parallel sum of Schur complements associated with individual subdomains.

The matrix on the right-hand side of the generalized eigenvalue problem can essentially be extracted from the fully assembled stiffness matrix (disregarding the decoupling with respect to the subcomponents), with the only exception being the variant that uses a scaled mass matrix (cf. section 3.3.4).

Let us remark—and mention in advance before the introduction of an ACMS-based coarse space in chapter 7—that the adaptive GDSW-type coarse spaces are easier to implement with respect to arbitrary types of interface decompositions: In the next chapter, interface partitions for unstructured domain decompositions will be defined. These can be highly complex and demanding to construct since many special cases and their interdependencies have to be accounted for in an implementation. For example, coarse edges (subdomain edges in a structured setting) can have coarse nodes (subdomain vertices in a structured setting) at either end but they may also have no incident coarse nodes at all; coarse edges can consist of multiple disconnected components, and each component can even have more than two boundary nodes; cf. fig. 7.4.

The GDSW partition defined in chapter 5 ensures that some favorable properties are satisfied, which can influence the performance and coarse space dimension positively. However, our experience has shown that these restrictions—such as requiring the connectivity of interface components—are not necessary to obtain a robust preconditioner. In contrast, it is more challenging to implement the coarse spaces of chapter 7.

As we have mentioned in chapter 3, the matrix $K_{RR}^{\Omega_\xi}$ (or $K_{RR}^{\Omega_\xi^l}$ in the case of the slab variant, or $K_{RR}^{\Omega_k}$ in the case of the S-variant) can be singular for a linear elasticity problem. If ξ is given by a straight edge or a vertex, in three dimensions, one or three linearized rotation modes are in the null space. In case the S-variant is used, the likelihood of encountering a singular matrix is even larger than for the original method, since $\xi \cap \overline{\Omega}_k$, $k \in n(\xi)$, generally consists of fewer finite element nodes than ξ. However, as we will learn in remark 6.5, the Schur complement always exists and is uniquely defined. Specifically, let $\hat{K} = K^{\Omega_\xi}$, $\hat{K} = K^{\Omega_k}$, $\hat{K} = K^{\Omega_\xi^l}$, or $\hat{K} = K^{\Omega_k^l}$. Then the linear system

$$\hat{K}_{RR}U = \hat{K}_{R\xi} \tag{4.3}$$

always has at least one solution. Furthermore, all solutions lead to the same Schur complement.

The matrix $\hat{K}_{RR}$ is always symmetric and positive semidefinite, which leads to the following possibilities to compute a solution to (4.3): Let us first mention the method we opted to use to obtain the numerical results in this work. By adding a small regularization term $\varepsilon\mathcal{R}$, where $\mathcal{R}$ is symmetric and positive definite, we obtain a symmetric positive definite matrix:

$$\hat{K}_{RR} \leftarrow \hat{K}_{RR} + \varepsilon\mathcal{R}.$$

For the results of this work, we have chosen the diagonal of $\hat{K}_{RR}$ multiplied by $\varepsilon = 10^{-15}$:

$$\hat{K}_{RR} \leftarrow \hat{K}_{RR} + \varepsilon\hat{K}_{RR,\text{diag}}.$$

A disadvantage of this method is the lower accuracy of the solution. Tests have shown that a fairly high accuracy is required, such that ε must be chosen very small.

In the following, we mention a few more possibilites. Theoretically, we could compute a full pseudoinverse (e.g., a Moore–Penrose inverse) of $\hat{K}_{RR}$. Although completely algebraic, this is very expensive in terms of memory requirements and processor time.

For a more efficient and still algebraic method, we can exploit that the eigenvalues of $\hat{K}_{RR}$ are nonnegative. Thus, we may consider using a rank-revealing factorization to compute a solution; cf. [Pan00].

We describe two more approaches, which rely on geometric information. The first approach uses a projection to remove the null space, in which case we require geometric information to construct linearized rotations. If direct solvers are used on the transformed linear systems, care must be taken to avoid dense matrices.

The second approach eliminates a subset of the degrees of freedom of $\hat{K}_{RR}$ and, thereby, effectively enforces an additional Dirichlet condition. At best, no additional computational cost is introduced.

As an example, let us consider three-dimensional linear elasticity and the case of a body that is clamped on a straight pole as in fig. 1.3 (right). The body can rotate around the pole without deforming, and we can prevent it from rotating by fixing an
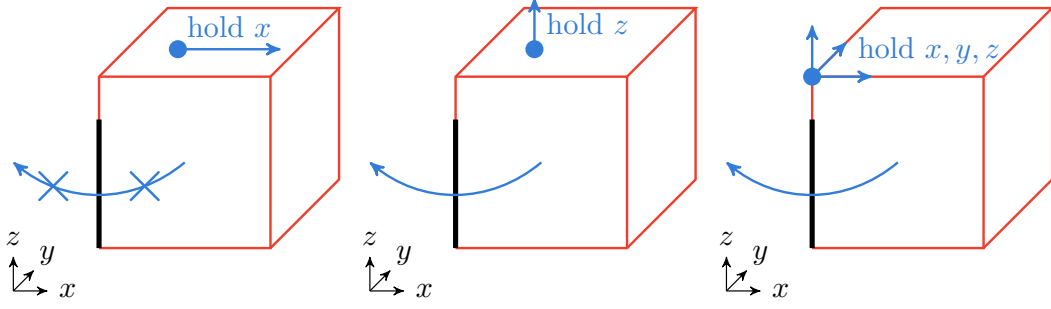
Figure 4.5.: Solid body (in red) clamped on an interface component (thick black line), which is a straight edge; cf. fig. 1.3. Three scenarios for enforcing a displacement at a spot are marked with a blue disk. **(Left)** Hold in x-direction: body cannot rotate around the edge anymore. **(Center)** Hold in z-direction: body can still rotate around the edge. **(Right)** Hold all degrees of freedom of the marked node: body can still rotate around the edge.

additional point in place. The only requirement is that the additional point must not lie on the same straight line as the pole. In terms of an implementation, introducing a Dirichlet condition at a single degree of freedom (that prevents the rotation) suffices. However, the degree of freedom must be selected carefully, which can be accomplished using geometric information. See fig. 4.5 for three examples, where only one selection leads to an invertible problem. In theory, we can use as many additional degrees of freedom as we like. This, however, comes at the cost of an additional, singular Schur complement system that needs to be solved, for example, using a full pseudoinverse.

We assume that, as in (4.3), the linear system

$$\hat{K}_{RR}x = y$$

has a solution. Let $\tilde{D} \subset R$ be a set of degrees of freedom such that $\hat{K}_{\tilde{R},\tilde{R}}$ is invertible, where $\tilde{R}$ denotes the remaining degrees of freedom of R. Note that $\tilde{D}$ must have at least as many elements as the dimension of the null space of $\hat{K}_{RR}$. We partition $\hat{K}_{RR}$, y, and x

by $\tilde{R}$ and $\tilde{D}$ and eliminate $\tilde{R}$ to obtain the Schur complement system

$$
\begin{pmatrix} \hat{K}_{\tilde{R},\tilde{R}} & \hat{K}_{\tilde{R},\tilde{D}} \\ 0 & S_{\tilde{D},\tilde{D}} \end{pmatrix} \cdot \begin{pmatrix} x_{\tilde{R}} \\ x_{\tilde{D}} \end{pmatrix} = \begin{pmatrix} y_{\tilde{R}} \\ y_{\tilde{D}} - \hat{K}_{\tilde{D},\tilde{R}} \hat{K}_{\tilde{R},\tilde{R}}^{-1} y_{\tilde{R}} \end{pmatrix},
$$

where

$$
S_{\tilde{D},\tilde{D}} = \hat{K}_{\tilde{D},\tilde{D}} - \hat{K}_{\tilde{D},\tilde{R}} \hat{K}_{\tilde{R},\tilde{R}}^{-1} \hat{K}_{\tilde{R},\tilde{D}}.
$$

In case $\tilde{D}$ was chosen properly, the null space dimension of $S_{\tilde{D},\tilde{D}}$ is identical to that of $\hat{K}_{RR}$. Since $\tilde{D}$ should be very small, we can use a full pseudoinverse to solve the system above. In case a minimal set $\tilde{D}$ was chosen, this implies that $S_{\tilde{D},\tilde{D}} = 0$ and also $y_{\tilde{D}} - \hat{K}_{\tilde{D},\tilde{R}} \hat{K}_{\tilde{R},\tilde{R}}^{-1} y_{\tilde{R}} = 0$ since the system must have a solution. In that case, solving the transformed system is no more expensive than solving the initial system with $\hat{K}_{RR}$.

4.5. Numerical Results for Diffusion Problems

In tables 4.1 and 4.2, numerical results for model problems **(2)** and **(3)** from sections 2.2 and 2.3 are shown to examine the reduction in the coarse space dimension that is achieved by using the interface decomposition from section 4.1 or section 4.3. Results for the other two model problems are given in tables B.2 and B.9.

As in section 3.4, we consider three-dimensional diffusion problems with $f \equiv 1$ for the right-hand side of (1.1). The convergence criterion is chosen as the reduction of the relative, unpreconditioned residual by 10^{-8}; the initial vector is set to the zero vector, and the iteration is stopped if it does not converge within $2\,000$ iterations. The condition number estimate is obtained using the Lanczos method; cf. [Saa03, sect. 6.7.3].

The results for the nonadaptive preconditioners show that using RGDSW, we can achieve a significant reduction in the coarse space dimension—from 441 to 76 for problem **(2)** and from $1\,593$ to 339 for problem **(3)**—without significantly influencing the number of iterations and condition number.

By using the interface partition of wire basket stars and subdomain faces instead of subdomain vertices, edges, and faces, the coarse space dimension is reduced by 38.4% for problem **(2)** and by 34.7% for problem **(3)**, using R–WB–AGDSW instead of AGDSW;

method	*tol*	it.	κ	$\dim V_0$	($\mathcal{V}$,	$\mathcal{E}$,	$\mathcal{F}$,	$\mathcal{S}$)	$\frac{\dim V_0}{\mathrm{dof}}$
GDSW	—	125	$3.8{\cdot}10^6$	441	(70,	199,	172,	—)	0.79%
RGDSW	—	133	$3.9{\cdot}10^6$	76	(—,	—,	—,	76)	0.14%
AGDSW	10^{-5}	129	$4.2{\cdot}10^5$	483	(70,	200,	213,	—)	0.86%
AGDSW	0.001	49	20.1	500	(70,	215,	215,	—)	0.89%
AGDSW	0.1	49	20.1	500	(70,	215,	215,	—)	0.89%
AGDSW–S	10^{-5}	128	$4.2{\cdot}10^5$	483	(70,	200,	213,	—)	0.86%
AGDSW–S	0.001	49	20.1	500	(70,	215,	215,	—)	0.89%
AGDSW–S	0.1	49	20.1	500	(70,	215,	215,	—)	0.89%
R–WB–AGDSW	10^{-5}	138	$4.3{\cdot}10^6$	299	(—,	—,	213,	86)	0.53%
R–WB–AGDSW	0.001	53	21.5	308	(—,	—,	215,	93)	0.55%
R–WB–AGDSW	0.1	53	21.5	308	(—,	—,	215,	93)	0.55%
R–WB–AGDSW–S	10^{-5}	136	$3.1{\cdot}10^5$	301	(—,	—,	213,	88)	0.54%
R–WB–AGDSW–S	0.001	53	21.5	308	(—,	—,	215,	93)	0.55%
R–WB–AGDSW–S	0.1	53	21.5	308	(—,	—,	215,	93)	0.55%
RAGDSW	10^{-5}	78	$7.8{\cdot}10^4$	109	(—,	—,	—,	109)	0.19%
RAGDSW	0.001	56	24.2	112	(—,	—,	—,	112)	0.20%
RAGDSW	0.1	56	24.2	113	(—,	—,	—,	113)	0.20%
RAGDSW–S	10^{-5}	77	$7.8{\cdot}10^4$	109	(—,	—,	—,	109)	0.19%
RAGDSW–S	0.001	56	24.2	112	(—,	—,	—,	112)	0.20%
RAGDSW–S	0.1	56	23.6	118	(—,	—,	—,	118)	0.21%

Table 4.1.: **(Model problem (2))** Results for the coefficient function in fig. 2.3, the diffusion problem, different methods and tolerances for the selection of eigenvectors: iteration count, condition number, resulting coarse space dimension, and coarse space dimension over the size of the stiffness matrix. The number of coarse functions associated with subdomain vertices, edges, faces, wire basket and interface stars is given in parentheses. $\mathcal{S}$ refers to $\mathcal{S}_{\mathcal{W}}/\mathcal{S}_{\Gamma}$.

method	*tol*	it.	κ	$\dim V_0$	($\mathcal{V}$, $\mathcal{E}$, $\mathcal{F}$, $\mathcal{S}$)	$\frac{\dim V_0}{\text{dof}}$
GDSW	—	565	$1.3{\cdot}10^6$	1 593	(328, 750, 515, —)	0.27%
RGDSW	—	536	$1.1{\cdot}10^6$	339	(—, —, —, 339)	0.06%
AGDSW	10^{-4}	549	$5.8{\cdot}10^4$	1 781	(328, 652, 801, —)	0.30%
AGDSW	0.001	59	30.1	1 814	(328, 678, 808, —)	0.31%
AGDSW	0.1	51	22.1	1 907	(328, 678, 901, —)	0.32%
AGDSW–S	10^{-4}	431	$4.9{\cdot}10^4$	1 789	(328, 658, 803, —)	0.30%
AGDSW–S	0.001	59	29.7	1 815	(328, 679, 808, —)	0.31%
AGDSW–S	0.1	51	22.1	1 913	(328, 680, 905, —)	0.32%
R–WB–AGDSW	10^{-4}	581	$4.0{\cdot}10^4$	1 175	(—, —, 801, 374)	0.20%
R–WB–AGDSW	0.001	57	30.2	1 200	(—, —, 808, 392)	0.20%
R–WB–AGDSW	0.1	52	23.5	1 293	(—, —, 901, 392)	0.22%
R–WB–AGDSW–S	10^{-4}	348	$4.1{\cdot}10^4$	1 185	(—, —, 803, 382)	0.20%
R–WB–AGDSW–S	0.001	58	30.0	1 201	(—, —, 808, 393)	0.20%
R–WB–AGDSW–S	0.1	52	23.5	1 300	(—, —, 905, 395)	0.22%
RAGDSW	10^{-4}	674	$4.4{\cdot}10^4$	627	(—, —, —, 627)	0.11%
RAGDSW	0.001	59	33.7	660	(—, —, —, 660)	0.11%
RAGDSW	0.1	53	24.4	792	(—, —, —, 792)	0.13%
RAGDSW–S	10^{-4}	499	$2.7{\cdot}10^4$	641	(—, —, —, 641)	0.11%
RAGDSW–S	0.001	59	33.7	663	(—, —, —, 663)	0.11%
RAGDSW–S	0.1	50	21.7	899	(—, —, —, 899)	0.15%

Table 4.2.: **(Model problem (3))** Results for the coefficient function in fig. 2.4, the diffusion problem, different methods and tolerances for the selection of eigenvectors: iteration count, condition number, resulting coarse space dimension, and coarse space dimension over the size of the stiffness matrix. The number of coarse functions associated with subdomain vertices, edges, faces, wire basket and interface stars is given in parentheses. $\mathcal{S}$ refers to $\mathcal{S}_{\mathcal{W}}/\mathcal{S}_{\Gamma}$.

a tolerance of 0.001 was selected in both cases. By using RAGDSW, we achieve an even larger reduction of 77.6% for problem **(2)** and 64.1% for problem **(3)** compared to AGDSW. In all cases, the obtained condition number and number of iterations are similar to that of AGDSW. As in section 3.4, the coarse space dimensions are minimal in the following sense: if a tolerance smaller than 0.001 is used to significantly decrease the coarse space dimension, the condition number and number of iterations increase considerably.

Furthermore, as in section 3.4, the S-variant achieves comparable results. In practice, a tolerance between 0.001 and 0.1 would be selected, where 0.1 leads to coarse spaces that are unnecessarily large. However, we will observe later for linear elasticity problems and, especially, for model problem **(4)** that even using a tolerance of 0.01 can lead to much larger condition numbers.

In table 4.3, results for the slab variant are given; see table B.5 for results of problem **(1)**. Using only one layer of finite elements for the slab, we obtain an increase in the coarse space dimension of 14.3% for problem **(2)** and 71.2% for problem **(3)** and RAGDSW. If a slab of three layers of finite elements is used, the results are almost identical. Comparing the results to the ones of AGDSW in table 4.3, the increase in the coarse space dimension is much more pronounced for RAGDSW. This is expected as larger interface components are more likely to detect connected patches of large coefficients, which is hindered by using a small slab. The differences between RAGDSW and RAGDSW–S are negligible.

We conclude that using a small slab may reduce the computational cost, but it may also significantly affect the coarse space dimension. Whether using the slab variant can reduce the computational cost depends on the matrices available during assembly and on the problem type: for simple problems—as problem **(2)**—using a minimal slab may be sufficient.

In table 4.4, results for lumping the stiffness matrix are given. There seems to be no difference in performance between the original and the lumped versions; see also table B.7 for results of problem **(1)**.

E	method	*tol*	it.	κ	$\dim V_0$	$\frac{\dim V_0}{\text{dof}}$
	RAGDSW	0.001	56	24.2	112	0.20%
	RAGDSW(3)	0.001	56	24.1	113	0.20%
(2)	RAGDSW(1)	0.001	56	24.1	128	0.23%
	RAGDSW–S	0.001	56	24.2	112	0.20%
	RAGDSW–S(3)	0.001	56	24.1	113	0.20%
	RAGDSW–S(1)	0.001	56	24.1	128	0.23%
	RAGDSW	0.001	59	33.7	660	0.11%
	RAGDSW(3)	0.001	59	33.7	671	0.11%
(3)	RAGDSW(1)	0.001	58	29.5	1 130	0.19%
	RAGDSW–S	0.001	59	33.7	663	0.11%
	RAGDSW–S(3)	0.001	59	33.7	676	0.11%
	RAGDSW–S(1)	0.001	58	29.5	1 130	0.19%

Table 4.3.: Results for the coefficient functions **(2)** and **(3)** in figs. 2.3 and 2.4, the diffusion problem, different methods, and a tolerance of 0.001 for the selection of eigenvectors: iteration count, condition number, resulting coarse space dimension, and coarse space dimension over the size of the stiffness matrix. If the slab variant is used, the slab width in layers of finite elements is appended in parentheses to the method's name.

E	method	*tol*	it.	κ	$\dim V_0$	$\frac{\dim V_0}{\mathrm{dof}}$
	RAGDSW	0.001	56	24.2	112	0.20%
	RAGDSW–ℓ(K)	0.001	56	24.2	112	0.20%
(2)	RAGDSW–S	0.001	56	24.2	112	0.20%
	RAGDSW–S–ℓ(K)	0.001	56	24.2	112	0.20%
	RAGDSW–M	0.001	56	24.2	112	0.20%
	RAGDSW–ℓ(M)	0.001	56	24.2	112	0.20%
	RAGDSW	0.001	59	33.7	660	0.11%
	RAGDSW–ℓ(K)	0.001	59	33.7	660	0.11%
(3)	RAGDSW–S	0.001	59	33.7	663	0.11%
	RAGDSW–S–ℓ(K)	0.001	59	33.7	663	0.11%
	RAGDSW–M	0.001	59	33.7	660	0.11%
	RAGDSW–ℓ(M)	0.001	59	33.7	660	0.11%

Table 4.4.: Results for the coefficient functions **(2)** and **(3)** in figs. 2.3 and 2.4, the diffusion problem, different methods, and a tolerance of 0.001 for the selection of eigenvectors: iteration count, condition number, resulting coarse space dimension, and coarse space dimension over the size of the stiffness matrix. If a lumped matrix is used, $\ell(K)$ or $\ell(M)$ is appended to the method's name.

5. Interface Partitions

Parts of this section are based on and are similar to [HKK+22, sect. 8].

In the following section, we decompose the interface Γ^h into sets $\mathcal{P}$ of components ξ that satisfy

$$\Gamma^h = \bigcup_{\xi \in \mathcal{P}} \xi \qquad \wedge \qquad \xi \cap \tilde{\xi} = \emptyset \quad \forall \xi, \tilde{\xi} \in \mathcal{P} : \xi \neq \tilde{\xi}.$$

The main focus of this section is the construction of interface partitions for unstructured domain decompositions as required for the meshes in chapter 2. Owing to the complexity of such decompositions—which can be obtained, for example, with a graph partitioner, such as METIS [KK98]—we cannot rely anymore on an intuitive understanding of subdomain vertices, edges, and faces as for structured decompositions, and we require a more robust approach.

5.1. Nodal Equivalence Classes

We construct a partition of the interface by using a generalization of subdomain vertices, edges, and faces. Specifically, we define equivalence classes for finite element nodes based on the set of adjacent subdomains. For example, in two dimensions, the finite element nodes shared by exactly two subdomains define a nodal equivalence class (NEC), which we denote *coarse edge* (except for some special cases; see further below). We also refer to, e.g., [KW06; KR06; DW17], where nodal equivalence classes have been used for this purpose before. Let us note, however, that the precise definitions therein differ from the ones used here.

5. Interface Partitions

The number of adjacent subdomains of an interface component is an important factor in the construction of interface partitions. The cost for communication in a parallel setting and the cost to compute energy-minimizing extensions during the construction of coarse functions increases with the number of nodes in adjacent subdomains. Therefore, we aim to minimize the number of adjacent subdomains. To this end, we will use the set of adjacent subdomains to classify interface components.

Definition 5.1 (Adjacency of Interface Nodes)**.** *Let $x_1^h, x_2^h \in \Gamma^h$ be two finite element nodes on the interface. We say that x_1^h and x_2^h are adjacent if a finite element edge or face $z \subset \Gamma$ exists with $x_1^h, x_2^h \in \overline{z}$.*

We define the neighborhood or adjacency of a finite element node $x^h \in \Omega$ as the set of indices of subdomains that contain x^h:

$$n(x^h) := \{\, i \in \{1, 2, \ldots, N\} : x^h \in \overline{\Omega}_i \,\}.$$

By

$$n(\omega) := \{\, i \in \{1, 2, \ldots, N\} : \exists\, x^h \in \omega \text{ s.t. } x^h \in \overline{\Omega}_i \,\} = \bigcup_{x^h \in \omega} n(x^h) \tag{5.1}$$

we extend the definition to sets of finite element nodes $\omega \subseteq \Omega^h$.

Definition 5.2 (Nodal Equivalence Class (NEC))**.** *Let $\gamma \subset \Gamma^h$ be a set of interface nodes. We define nodal equivalence classes by the relation*

$$x_1^h \sim x_2^h \quad \Longleftrightarrow \quad n(x_1^h) = n(x_2^h),$$

where $x_1^h, x_2^h \in \gamma$. By $\mathcal{N}(x^h)$ we denote the NEC of a node $x^h \in \gamma$; that is, we have $x^h \in \mathcal{N}(x^h)$ and $n(x^h) = n(y^h)$ for all $y^h \in \mathcal{N}(x^h)$.

If $n(x_2^h) \subsetneq n(x_1^h)$, then $\mathcal{N}(x_1^h)$ is said to be an ancestor of $\mathcal{N}(x_2^h)$, which itself is a descendant of $\mathcal{N}(x_1^h)$. If a NEC does not have an ancestor, we call it a root.

In case the entire interface is decomposed into NECs, we call a root a coarse node if it consists of a single node. In three dimensions, NECs with exactly two adjacent subdomains are called coarse faces, and the remaining ones we denote by coarse edges. In two dimensions, every NEC that is not a coarse node is called a coarse edge.

In the case of cuboid subdomains, a coarse node is a subdomain vertex. In general, a root can contain more than a single node (cf. fig. 5.3). For example, the interface components of a beam decomposed lengthwise into cuboid subdomains are given by subdomain faces, which are roots; cf. [DW17, fig. 3].

5.2. Connected Components of Nodal Equivalence Classes

Based on definition 5.1 of adjacent interface nodes, we define connected components of NECs.

Definition 5.3 (Connected Components on the Interface)**.** *Let $\gamma \subset \Gamma^h$ be a set of nodes on the interface. A path on γ is an ordered set $(x_0^h, \ldots, x_s^h)$, $x_i^h \in \gamma$, of adjacent nodes. We call γ a connected component on the interface if there exists a path between any two nodes $x_0^h, x_s^h \in \gamma$.*

It is generally beneficial to use connected interface components as this facilitates the detection of connected patches of large coefficients, which can reduce the coarse space dimension. If a NEC consists of multiple connected components, we divide it into its connected components; see example 5.1. We call an interface component a twin if there exists another interface component that belongs to the same NEC.

Let $\mathcal{N}_{\mathrm{Con},\Gamma^h}$ be the set of connected components of NECs of the interface. The corresponding set of roots is defined as

$$\mathcal{R}_{\mathrm{Con}} \coloneqq \{\, \xi \in \mathcal{N}_{\mathrm{Con},\Gamma^h} : \nexists \tilde{\xi} \in \mathcal{N}_{\mathrm{Con},\Gamma^h} \text{ s.t. } n(\xi) \subsetneq n(\tilde{\xi}) \,\}.$$

Example 5.1 (Connected Components)**.** *In fig. 5.1, an interface partition with four NECs and multiple twins is shown. One NEC is given by two nodes marked with red squares. The two marked nodes are not connected, such that they are treated as individual components, specifically, as coarse nodes. Another NEC, which is marked with green circles, consists of two connected components (coarse edges).*
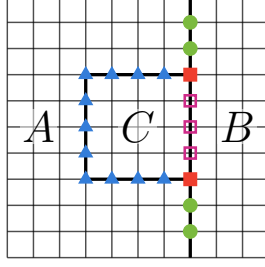
Figure 5.1.: Square domain with the subdomains A, B, and C. Two coarse nodes are marked with red, filled squares, with the adjacent subdomains (A, B, C). The coarse edges are marked with blue triangles (adjacent subdomains (A, C)), magenta square frames (adjacent subdomains (B, C)), and green circles (adjacent subdomains (A, B)); cf. [HKK+22, fig. 2].

5.3. GDSW: Coarse Nodes, Edges, and Faces

To define GDSW interface components, we first decompose the interface into nodal equivalence classes. These are further decomposed into connected components. The resulting interface partition contains coarse nodes $\mathcal{V}$, coarse edges $\mathcal{E}$, and (in three dimensions) coarse faces $\mathcal{F}$. Thus, in three dimensions, the GDSW interface partition is defined by $\mathcal{V} \cup \mathcal{E} \cup \mathcal{F} = \mathcal{N}_{\mathrm{Con},\Gamma^h}$, where

$$\mathcal{F} := \{\, \xi \in \mathcal{N}_{\mathrm{Con},\Gamma^h} : |n(\xi)| = 2 \,\},$$
$$\mathcal{V} := \{\, \xi \in \mathcal{R}_{\mathrm{Con}} : |n(\xi)| \geq 3 \ \wedge \ |\xi| = 1 \,\},$$
$$\mathcal{E} := \mathcal{N}_{\mathrm{Con},\Gamma^h} \setminus (\mathcal{F} \cup \mathcal{V}).$$

In two dimensions, we have $\mathcal{V} \cup \mathcal{E} = \mathcal{N}_{\mathrm{Con},\Gamma^h}$, where

$$\mathcal{V} := \{\, \xi \in \mathcal{R}_{\mathrm{Con}} : |n(\xi)| \geq 3 \ \wedge \ |\xi| = 1 \,\},$$
$$\mathcal{E} := \mathcal{N}_{\mathrm{Con},\Gamma^h} \setminus \mathcal{V}.$$

In case there is an interface component with a small number of nodes—for example, a coarse edge that consists of only two nodes—it may be more efficient to circumvent the setup and solution of a generalized eigenvalue problem and instead treat the two nodes as

coarse nodes; see, for example, fig. 5.2 (top left), in which the leftmost and bottommost edge nodes would be treated as coarse nodes, additionally to the ones highlighted with colors. Note that for unstructured domain decompositions, interface components that consist of only a single finite element node can occur fairly often; see section 5.7.2.

Following the construction of AGDSW in [HKKR19; HKK+22], all generalized eigenvalue problems have zero eigenvalues. If interface components that consist of only a single node are treated as coarse nodes, the constructed coarse functions are identical to the ones associated with zero eigenvalues. In this work, however, we include the Dirichlet boundary $\partial\Omega_D$ into the generalized eigenvalue problems (cf. section 3.1) and usually obtain positive eigenvalues for interface components that have adjacent subdomains that touch $\partial\Omega_D$. As a result, by reclassifying single node coarse edges, the coarse space dimension can increase. For the coarse space that will be introduced in chapter 7, the differences would be even more substantial since zero eigenvalues associated with local generalized eigenvalue problems are even rarer.

In case single node interface components are reclassified as coarse nodes, the modified interface partition is given by $\mathcal{N}_{\mathrm{Con},\Gamma^h} = \hat{\mathcal{V}} \cup \hat{\mathcal{E}} \cup \hat{\mathcal{F}}$, where

$$\hat{\mathcal{F}} := \{\, \xi \in \mathcal{N}_{\mathrm{Con},\Gamma^h} : |n(\xi)| = 2 \ \wedge \ |\xi| \geq 2 \,\},$$
$$\hat{\mathcal{V}} := \{\, \xi \in \mathcal{N}_{\mathrm{Con},\Gamma^h} : |\xi| = 1 \,\},$$
$$\hat{\mathcal{E}} := \mathcal{N}_{\mathrm{Con},\Gamma^h} \setminus (\hat{\mathcal{F}} \cup \hat{\mathcal{V}}).$$

5.4. RGDSW: Interface Stars

The size (number of nodes) of an interface component is an important property for adaptive coarse spaces: a large component can benefit the detection of connected patches of large coefficients and, thus, reduce the coarse space dimension; cf. section 1.5.1. On the other hand, the cost for the setup and solution of a generalized eigenvalue problem increases with a component's size. In the following, we accept an increase in components' sizes compared to GDSW. Furthermore, we will obtain interface components that are more similar in size, which can increase parallel efficiency.

Although GDSW vertex functions are relatively cheap to construct, they unnecessarily inflate the coarse space dimension of AGDSW since their construction does not take the coefficient function into account (they are nonadaptive coarse functions). Similarly—in three dimensions—even if to a lesser degree, AGDSW edge problems are fairly expensive if measured by their contribution. Note that not only the size of a component is relevant but also its shape. A thin and long component (such as a coarse edge) will generally do worse than a compact one that is equally spread out in all directions.

As mentioned in section 4.1, minimizing the number of adjacent subdomains is desirable to improve parallel efficiency. The roots of an interface partition have the largest number of adjacent subdomains compared to their descendants. Thus, with respect to the roots, we cannot reduce the number of adjacent subdomains by using a different interface partition. In the following, we combine roots and some parts of their descendants to construct a partition without unnecessarily inflating the number of adjacent subdomains. We cannot, however, join a root with all its descendants as the components would then overlap. Instead, we use an iterative process to distribute the descendants' nodes between their roots.

We will describe an algorithm to generate an interface partition $\mathcal{S}_\Gamma$ of disjoint, connected components. Note that the algorithm slightly differs from the one in [HKK+22]. As a preprocessing step, we modify $\mathcal{R}_{\text{Con}}$: if a descendant of a root in $\mathcal{R}_{\text{Con}}$ does not have an ancestor in its adjacency, it is reclassified as a root; we call the reclassified component an island and denote the resulting set of roots by $\mathcal{R}_{\text{Con,Adj}}$. By this, we improve the construction of RGDSW interface components to be more similar in size; see examples 5.2 and 5.3.

Each interface component $\xi \in \mathcal{S}_\Gamma$ is associated and initialized with a root in $\mathcal{R}_{\text{Con,Adj}}$. We then select the smallest component and enlarge it by one layer of adjacent interface nodes (as per definition 5.1), where each node must satisfy the following conditions:

- a new node must not have been assigned to another $\xi \in \mathcal{S}_\Gamma$;
- a node's NEC must be a descendant of the component's root; i.e., $n(x^h) \subseteq n(\xi)$.
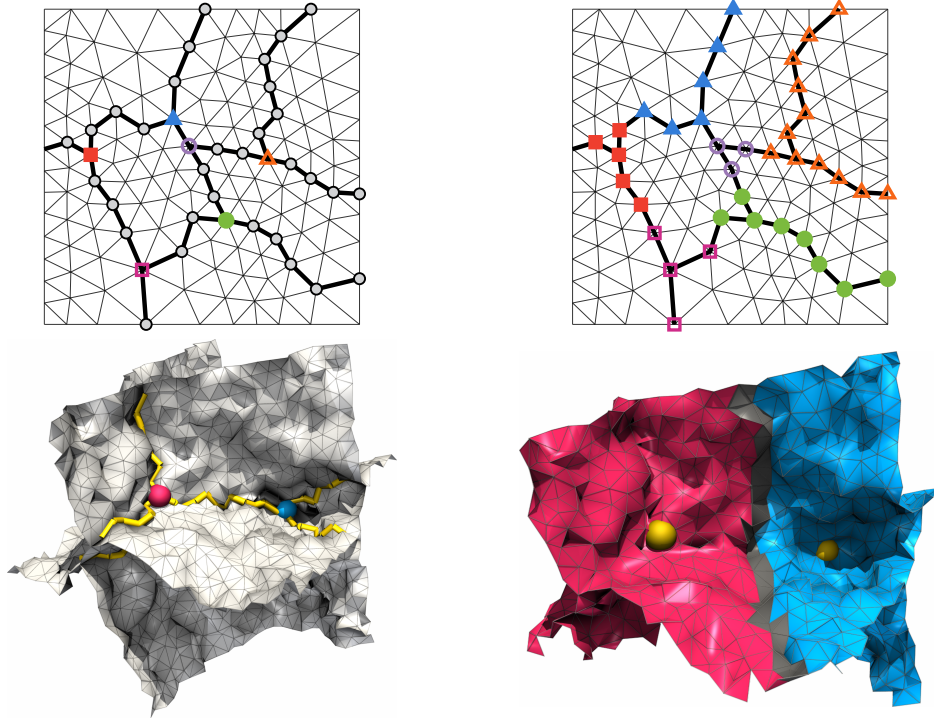
Figure 5.2.: **(Right)** Sample interface partitions for RGDSW in two and three dimensions; cf. [HKK+22, fig. 3] and fig. 4.2 for the structured case. **(Top left)** Interface nodes (gray disks) and coarse nodes (colored markers). Dirichlet boundary condition on the left; Neumann condition on the remaining boundary. **(Top right)** Corresponding RGDSW interface components. **(Bottom left)** Finite element faces in gray, wire basket edges in yellow, and two roots (coarse nodes) in red and blue. **(Bottom right)** Corresponding interface stars; shared finite element faces in dark gray.

We repeat this process until all interface nodes have been assigned to a $\xi \in \mathcal{S}_\Gamma$. The resulting interface components in $\mathcal{S}_\Gamma$ are denoted *(coarse) interface stars.* Samples for interface stars in two and three dimensions are given in fig. 5.2. In section 5.7.3, we compare the size distribution of different interface components.

We note that there is no limitation for the coarse space RAGDSW of chapter 4 to use other types of interface decompositions; see, for example, section 5.5 for another type of
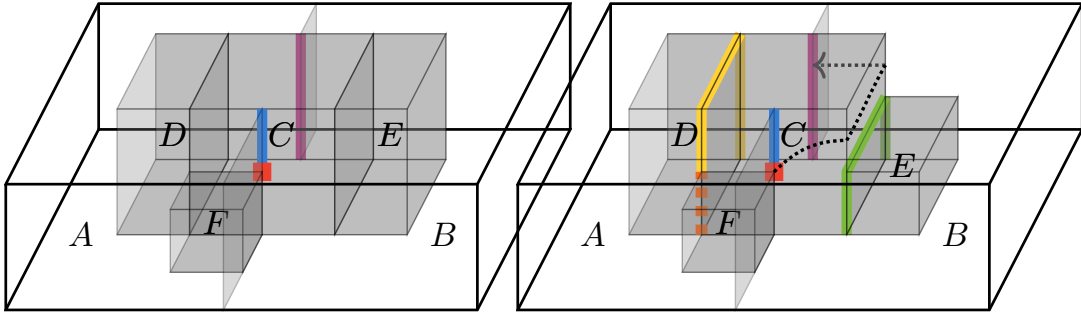
Figure 5.3.: **(Left)** Domain with the subdomains A to F. Coarse node marked with a red square (adjacent subdomains (A, B, C, F)). Two coarse edges marked in blue (solid line in the front) and magenta (solid line in the back) with the adjacent subdomains (A, B, C). **(Right)** Additional markings for three coarse edges: in green (solid line on the right; adjacent subdomains (B, C, E)), in yellow (solid line on the left; adjacent subdomains (A, C, D)), and one in orange (dashed line on the left; adjacent subdomains (A, C, D, F)). The orange and green coarse edge are roots. A dotted black path that leads from the red coarse node to the magenta coarse edge shows a connection via the coarse face (C, B), a descendant of the coarse node.

decomposition. However, as mentioned before, the type of interface decomposition can significantly influence various characteristics of the method.

Example 5.2 (Disconnected Descendant (Island))**.** *We consider the case in fig. 5.3 (left). Therein, we have a domain partitioned into six subdomains, labeled A to F. A coarse node (i.e., root) highlighted with a red square is the ancestor of the coarse edges marked in blue and magenta. The two coarse edges are part of the same NEC, but as they are connected components—by definition of section 5.2—we treat them as separate coarse edges.*

Furthermore, we reclassify the magenta coarse edge as a root: Let $\mathcal{N}$ be the NEC of the coarse node. There does not exist a path on the interface via finite element edges such that all path nodes belong to a NEC that is a descendant of $\mathcal{N}$. In the example at hand,

a path must either touch the subdomain D or E—both of them are not adjacent to the coarse node.

As a result, the magenta coarse edge would never be discovered in the process of constructing interface stars. We could, of course, construct new interface stars from "missing" interface nodes in a post processing step. But, in the considered example, it is beneficial to do this in a preprocessing step to obtain components of similar size.

Example 5.3 (Poorly Connected Descendant (Island))**.** *We consider the case in fig. 5.3 (right). The only difference is that subdomain E has shrunk. Contrary to example 5.2, this has opened up a path from the coarse node via descendants to the magenta coarse edge (a sample path is sketched in the figure).*

Nevertheless, the interface star associated with the red coarse node would either not reach the magenta coarse edge (as the root/coarse edge in green would likely block its path) or it would have an unfavorable, slim shape.

5.5. R–WB–GDSW: Wire Basket Stars and Coarse Faces

The construction of the interface partition for R–WB–GDSW is similar to that of RGDSW. The interface components of GDSW that give the least benefit but exist plentiful are coarse nodes. For RGDSW, we have solved this by using an agglomeration of interface nodes that are close to the respective coarse node. However, this can result in significantly larger components, which increases the computational effort for the setup and solution of generalized eigenvalue problems. As a compromise, we construct an interface partition that uses coarse faces and combines a coarse node with parts of the adjacent edges; cf. fig. 4.4 for an example in the structured case. By doing this, we can ensure that (in general) the largest components are not larger than those of AGDSW. We call the new type of interface component *(coarse) wire basket stars*, and the corresponding set is denoted by $\mathcal{S}_{\mathcal{W}}$. In two dimensions, we define $\mathcal{S}_{\mathcal{W}} \coloneqq \mathcal{S}_{\Gamma}$.

The construction process in three dimensions is identical to that of interface stars, except that the wire basket is used instead of the entire interface.

5.6. Subcomponents of Interface Components

As for the structured description of RAGDSW in chapter 4, we further decompose each interface component $\xi \in \mathcal{P}$ into subcomponents based on nodal equivalence classes; cf. fig. 4.3. Let $\mathcal{N}_{\Gamma^h}$ denote the set of NECs of Γ^h. Then the set of subcomponents of ξ is defined as

$$\mathcal{N}_\xi := \{\, \xi \cap c : c \in \mathcal{N}_{\Gamma^h} \ \wedge \ \xi \cap c \neq \emptyset \,\}, \quad \xi \in \mathcal{P}. \tag{5.2}$$

We define

$$n_\xi := |\mathcal{N}_\xi| \tag{5.3}$$

as the number of subcomponents—or NECs—of ξ. Let ξ_i, $i = 1, \dots, n_\xi$, be the subcomponents of ξ such that

$$\mathcal{N}_\xi = \{\xi_i\}_{i=1}^{n_\xi},$$

and

$$\xi = \bigcup_{i=1}^{n_\xi} \xi_i, \quad \xi_i \cap \xi_j = \emptyset \ (i \neq j),$$

and let

$$\mathcal{N}_{ec,\mathcal{P}} := \bigcup_{\xi \in \mathcal{P}} \mathcal{N}_\xi \tag{5.4}$$

be the set of all subcomponents.

5.7. Interface Partitions of the Model Problems

In the following, for model problems **(1)**–**(4)** in chapter 2, we analyze some properties of the GDSW, R–WB–GDSW, and RGDSW interface partitions.

After deriving a condition number bound in chapter 6, in section 6.4.1, we will analyze the dependence of the condition number bound in theorem 6.1 on the different interface partitions and model problems.

5.7.1. Size of the Neighborhood

The setup of generalized eigenvalue problem (4.2) can be expensive, especially for interface components with many or large adjacent subdomains, as this influences the

	total			single nodes		twins		
	$\mathcal{V}$	$\mathcal{E}$	$\mathcal{F}$	$\mathcal{E}$	$\mathcal{F}$	$\mathcal{V}$	$\mathcal{E}$	$\mathcal{F}$
(1)	419	927	634	130 [14.0%]	20 [3.2%]	6 [1.4%]	16 [1.7%]	2 [0.3%]
(2)	70	199	172	27 [13.6%]	0 [0.0%]	0 [0.0%]	0 [0.0%]	0 [0.0%]
(3)	328	750	515	53 [7.1%]	8 [1.6%]	12 [3.7%]	22 [2.9%]	12 [2.3%]
(4)	2 256	4 688	3 036	697 [14.9%]	101 [3.3%]	16 [0.7%]	56 [1.2%]	8 [0.3%]

Table 5.1.: GDSW interface partition: total number of components in $\mathcal{V}$, $\mathcal{E}$, $\mathcal{F}$, fraction that consists of only a single node, and fraction that is obtained after splitting an interface component because of multiple connected components (twins). **(1)**–**(4)** correspond to sections 2.1 to 2.4.

cost to compute energy-minimizing extensions. For the S-variant of section 3.3.3, the number and size of subdomains adjacent to an interface component is a crucial factor that determines its computational efficiency. For the original method, a crucial factor is the size of the union of subdomains adjacent to an interface component. We have seen before in fig. 2.1 that the subdomain size is fairly consistent and deviates only minimally from the average. The number of nodes in the union of subdomains that are adjacent to an interface component does only deviate moderately from the average as well. The largest deviations are encountered for wire basket stars with a minimum of 73.2% and a maximum of 144.2% with respect to the average; the numbers are almost identical for interface stars. Details are given in table A.1 for all types of interface components.

5.7.2. Single Node and Disconnected Interface Components

In tables 5.1 to 5.3, the number of interface components that consist of only a single node is given for the GDSW, R–WB–GDSW, and RGDSW interface partitions.

For GDSW—except for model problem **(3)**—approximately 14% of coarse edges consist of only a single node. The ratio is reduced to between 0.0% and 3.3% for coarse faces. Table 5.1 further shows that the number of interface components that have been split

	total		single nodes		twins	
	$\mathcal{S_W}$	$\mathcal{F}$	$\mathcal{S_W}$	$\mathcal{F}$	$\mathcal{S_W}$	$\mathcal{F}$
(1)	435	634	0 [0.0%]	20 [3.2%]	6 [1.4%]	2 [0.3%]
(2)	76	172	0 [0.0%]	0 [0.0%]	0 [0.0%]	0 [0.0%]
(3)	339	515	0 [0.0%]	8 [1.6%]	13 [3.8%]	12 [2.3%]
(4)	2 326	3 036	8 [0.3%]	101 [3.3%]	16 [0.7%]	8 [0.3%]

Table 5.2.: R–WB–GDSW interface partition: total number of components in $\mathcal{S_W}$, $\mathcal{F}$, fraction that consists of only a single node, and fraction that is obtained after splitting an interface component because of multiple connected components (twins). **(1)**–**(4)** correspond to sections 2.1 to 2.4.

into their connected components (twins) is small but not insignificant. For example, between 0.0% and 3.7% of coarse nodes in $\mathcal{V}$ are actually coarse edges or faces that have been split apart. The number of twins is similar for coarse nodes, wire basket stars, and interface stars.

With respect to single node interface components, the numbers in table 5.2 are more favorable for wire basket stars: at most 0.3% of wire basket stars consist of only a single node. For the considered model problems, there do not exist any interface stars that consist of only a single node; cf. table 5.3.

In table 5.3, the numbers of roots that consist of a single node are shown and, additionally, the number of islands. Except for model problem **(2)**, the number of roots that are not coarse nodes is between 2.2% and 2.9%. The number of islands is small and varies between 0.0% and 1.4%.

5.7.3. Distribution and Size of Interface Components

In fig. 5.4, the distribution of interface components is displayed for each model problem and the interface partitions GDSW, R–WB–GDSW, and RGDSW. The number of interface components for RGDSW is consistently below 25% of that of GDSW, and

	interface stars $\mathcal{S}_\Gamma$			roots $\mathcal{R}_{\text{Con,Adj}}$	
	total	single nodes	twins	single nodes	islands
(1)	435	0 [0.0%]	6 [1.4%]	425 [97.7%]	6 [1.4%]
(2)	76	0 [0.0%]	0 [0.0%]	70 [92.1%]	0 [0.0%]
(3)	339	0 [0.0%]	13 [3.8%]	329 [97.1%]	1 [0.3%]
(4)	2 325	0 [0.0%]	16 [0.7%]	2275 [97.8%]	19 [0.8%]

Table 5.3.: RGDSW interface partition: total number of components in $\mathcal{S}_\Gamma$, fraction that consists of only a single node, and fraction that is obtained after splitting a root because of multiple connected components (twins). Fraction of roots that consist of a single node and fraction that is not a root but has only descendants in its adjacency (island). The total number of roots is identical to the total number of interface stars; the number of twin roots is identical to the number of twin interface stars. **(1)**–**(4)** correspond to sections 2.1 to 2.4.

R–WB–GDSW is below 60% of that of GDSW. Thus, we can expect a significant reduction in the coarse space dimension. However, this comes at an increase in the size of interface components. For R–WB–GDSW, this is not an issue as wire basket stars are much smaller than coarse faces; see fig. 5.5. For RGDSW, on the other hand, interface stars can be significantly larger than coarse faces. The average interface star is between 60.0% to 73.3% larger than the average coarse face, except for the problem of section 2.2, for which there is an increase of 155.2%. The largest interface star is between 32.4% and 49.3% larger than the largest coarse face, except for the problem of section 2.4, for which there is an increase of 114.2%. For a better parallel performance, it would be beneficial to further optimize the generation process of coarse interface stars to construct interface components that are even more similar in size. However, the histograms for the interface stars show already a significant improvement in the deviation compared to the ones for coarse faces. From this, we conclude that the process used here to generate interface stars is beneficial for parallelization.
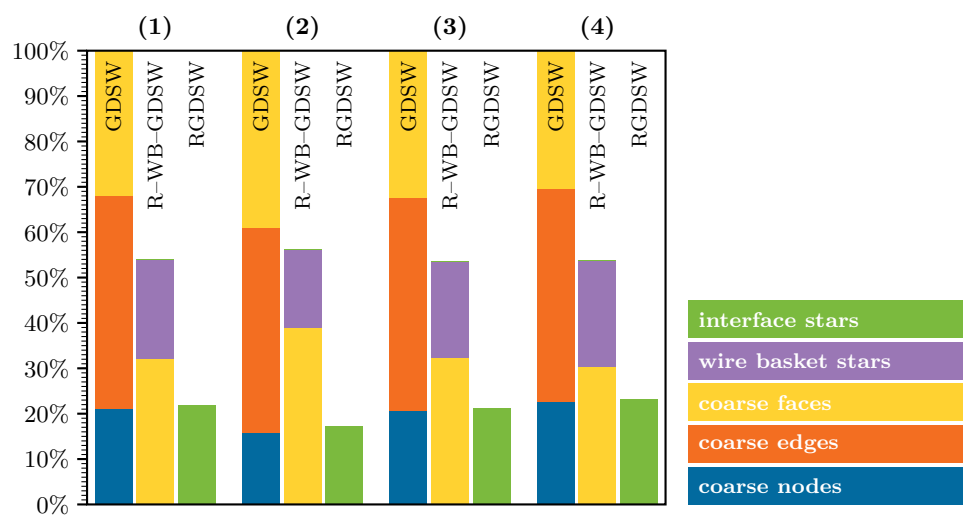
Figure 5.4.: Distribution of interface components for the GDSW, R–WB–GDSW, and RGDSW interface partitions. **(1)**–**(4)** correspond to sections 2.1 to 2.4. For each model problem **(1)**–**(4)**, the fractions are given with respect to the number of GDSW interface components.
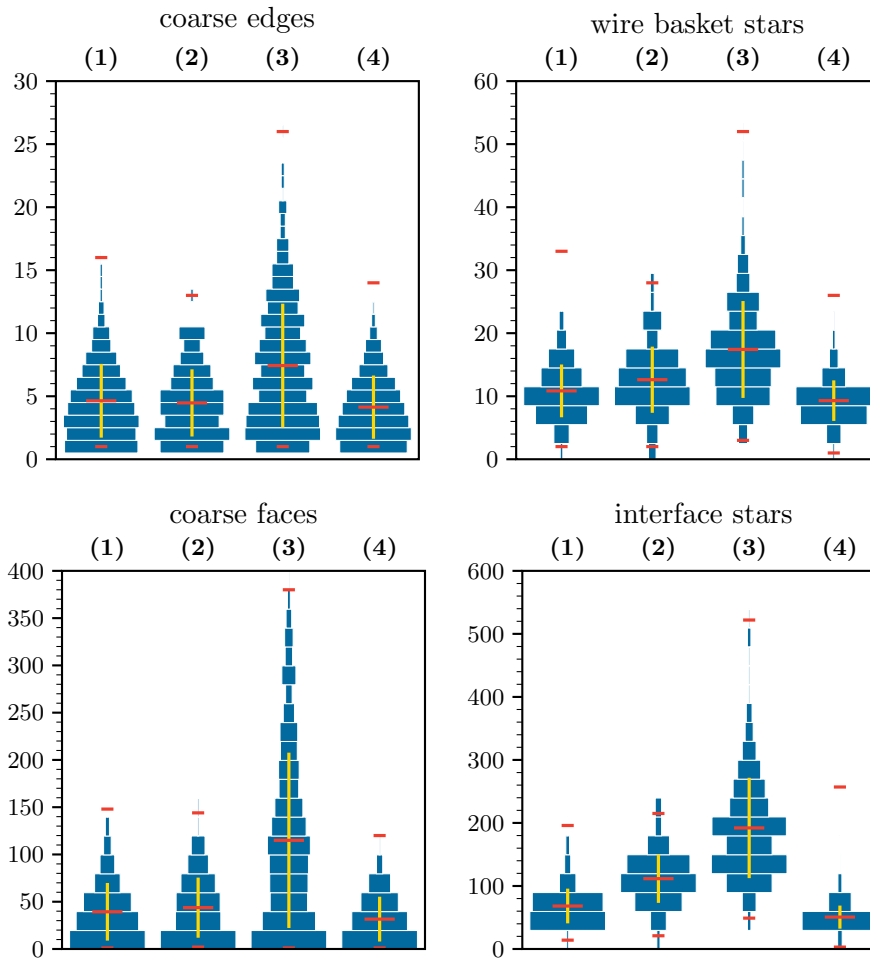
Figure 5.5.: Histograms of the number of nodes per interface component. **(1)**–**(4)** correspond to sections 2.1 to 2.4. The widths of the blue bars indicate the proportion of interface components that consist of the respective amount of nodes. The width of bars is not comparable between different model problems. The minimum, average, and maximum are marked in red. The range of one standard deviation from the average is marked in yellow.

6. Theory of Adaptive GDSW-Type Coarse Spaces

In this chapter, we derive a condition number bound for adaptive GDSW-type coarse spaces; the description is in large parts based on [HKK+22; HKKR19; HKKR18b] (in that order).

As both the preconditioner M_{OSL2}^{-1} and K are symmetric and positive definite, the spectral condition number of the preconditioned operator $P_{\text{ad}} := M_{\text{OSL2}}^{-1} K$ is given by

$$\kappa := \kappa(P_{\text{ad}}) := \kappa_2(P_{\text{ad}}) = \frac{\lambda_{\max}(P_{\text{ad}})}{\lambda_{\min}(P_{\text{ad}})},$$

where $\lambda_{\max}(P_{\text{ad}})$ and $\lambda_{\min}(P_{\text{ad}})$ are the maximum and minimum eigenvalues of P_{ad}:

$$\lambda_{\max}(P_{\text{ad}}) = \sup_{v \in V_{0,\partial\Omega_D}^h(\Omega)\setminus\{0\}} \frac{a_\Omega(P_{\text{ad}}v, v)}{a_\Omega(v, v)}, \qquad \lambda_{\min}(P_{\text{ad}}) = \inf_{v \in V_{0,\partial\Omega_D}^h(\Omega)\setminus\{0\}} \frac{a_\Omega(P_{\text{ad}}v, v)}{a_\Omega(v, v)}.$$

We have the bound

$$a_\Omega(P_{\text{ad}}v, v) \le \omega(\hat{N}_c + 1)a_\Omega(v, v) \quad \forall v \in V_{0,\partial\Omega_D}^h(\Omega), \tag{6.1}$$

for the largest eigenvalue; cf. [TW05, lemma 3.11]. Since we use exact local solvers (see eq. (2.4) and the local stability assumption 2.4 in [TW05]), we have $\omega = 1$. The constant $\hat{N}_c$ is the maximum number of overlapping subdomains $\{\Omega_i'\}_{i=1}^N$ any finite element node $x^h \in \overline{\Omega}$ can belong to; see [TW05, lemma 3.11], the follow-up discussion, and the proof of theorem 4.1 in [DW94]. By lemma 2.5 in [TW05], we have the bound

$$a_\Omega(P_{\text{ad}}v, v) \ge \frac{1}{C_0^2}a_\Omega(v, v) \quad \forall v \in V_{0,\partial\Omega_D}^h(\Omega), \tag{6.2}$$

for the smallest eigenvalue. From (6.1) and (6.2), we obtain the condition number bound

$$\kappa(P_{\mathrm{ad}}) \leq C_0^2(\hat{N}_c + 1). \tag{6.3}$$

The constant C_0^2 stems from the following assumption; finding a lower bound for it will be the goal of this chapter.

Assumption 6.1 (Stable Decomposition). *([TW05, assumption 2.2]) By definition in section 1.3, $V_0 \subset V_{0,\partial\Omega_D}^h(\Omega)$ denotes the coarse space, and V_i, $1 \leq i \leq N$, are local spaces. We assume that there exists a constant $C_0 > 0$ such that for every $u \in V_{0,\partial\Omega_D}^h(\Omega)$ there exists a decomposition*

$$u = \sum_{i=0}^{N} R_i^T u_i, \quad u_i \in V_i, \ 0 \leq i \leq N,$$

that satisfies

$$\sum_{i=0}^{N} \tilde{a}_i(u_i, u_i) \leq C_0^2 a_\Omega(u, u),$$

where $\tilde{a}_i(\cdot, \cdot)$ was defined in (1.7).

6.1. Variational Description of Adaptive GDSW-Type Coarse Spaces

This section follows [HKK+22, sect. 9]. Note however—as was mentioned during the construction of the coarse spaces—that we have slighty changed the definition of the coarse spaces to respect the global Dirichlet boundary condition for the setup of the Schur complement of the generalized eigenvalue problem. This and other minor changes are reflected in the following section and the subsequent proof of a condition number bound.

We will state the variational analogue of the matrix formulations for AGDSW and RAGDSW of chapters 3 and 4. The RAGDSW coarse space can be regarded as a generalization of the AGDSW coarse space as it can be used for arbitrary interface decompositions $\mathcal{P}$. As a result, the following description is valid for both types of coarse spaces.

By definition of (5.1), $n(\xi)$ is the index set that contains all subdomains adjacent to an interface component $\xi \in \mathcal{P}$. The closure of the union of subdomains adjacent to ξ is given by

$$\overline{\Omega}_\xi := \bigcup_{i \in n(\xi)} \overline{\Omega}_i. \tag{6.4}$$

For the construction of coarse functions and during the proof of the condition number bound, we often rely on, and make use of, extension operators. Computing these extensions only requires nodal values as an input. As a result, it is not necessary to use a finite element space for the domain of the extension operator. Instead, we will use functions that map each finite element node to a vector in $\mathbb{R}^{\hat{d}}$. Here, $\hat{d}$ depends on the problem at hand: for a scalar diffusion problem, it is $\hat{d} = 1$; for three-dimensional linear elasticity, it is $\hat{d} = 3$. For an arbitrary set $\omega \subseteq \Omega^h := \{\, x^h : x^h \in \overline{\Omega} \,\}$ of finite element nodes, we define the function space

$$X^h(\omega) := \{v \colon \omega \to \mathbb{R}^{\hat{d}}\} \tag{6.5}$$

of functions mapping from $\omega \subset \mathbb{R}^d$ to $\mathbb{R}^{\hat{d}}$. The space $X^h(\omega)$ may also be identified with a suitable definition of a finite element space $V^h(\omega)$; cf. [HKK+22, sect. 9]. Note that the space contains restrictions of functions that are not finite element functions; see, for example, the proof of lemma 6.5, where this property is used.

The left-hand side of the generalized eigenvalue problems in (3.1) and (4.2) is given by the Schur complement $S_{\xi\xi}$. As we will see below, the Schur complement is closely related to an energy-minimizing extension from ξ to $\overline{\Omega}_\xi$ (cf. [TW05, sect. 4.4; SBG96, sect. 4.6; Cia02, remark 2.1.1; Cia13, theorem 6.1-1]) in the way that the corresponding $S_{\xi\xi}$-inner product of a function is—with respect to the bilinear form $a_{\Omega_\xi}(\cdot, \cdot)$—the energy of the energy-minimizing extension of said function.

Let $\Omega_Q \subset \Omega$ satisfy the same properties as Ω; that is, $\overline{\Omega}_Q$ is the union of finite elements $T \in \tau_h(\Omega)$ and a connected subset of Ω. In the case of RAGDSW, we use $\Omega_Q = \Omega_\xi$, for the S-variants, we use $\Omega_Q = \Omega_k$ for some $k \in \{1, \ldots, N\}$, and for the slab variant, we use $\Omega_Q = \Omega_\xi^l$. Note that the slab variant may be combined with the S-variant, such that $\Omega_Q = \Omega_k^l$.

6. Theory of Adaptive GDSW-Type Coarse Spaces

We define the finite element space $V^h_{0,\xi,\partial\Omega_D}(\Omega_Q)$ of functions on Ω_Q that vanish on ξ and on the respective part of the global Dirichlet boundary $\partial\Omega_D$:

$$V^h_{0,\xi,\partial\Omega_D}(\Omega_Q) := \Big\{ w \in V^h(\Omega_Q) : w(x^h) = 0 \ \forall x^h \in \xi \cup \big(\partial\Omega_D \cap \overline{\Omega}_Q\big) \Big\}.$$

An energy-minimizing extension of a function $\tau_\xi \in X^h(\xi)$ is defined as a solution $v_\xi \in V^h(\Omega_Q)$ to

$$
\begin{aligned}
a_{\Omega_Q}(\, v_\xi \, , \, w \,) &= 0 && \forall w \in V^h_{0,\xi,\partial\Omega_D}(\Omega_Q), \\
v_\xi(x^h) &= \tau_\xi(x^h) && \forall x^h \in \xi \cap \overline{\Omega}_Q, \\
v_\xi(x^h) &= 0 && \forall x^h \in \partial\Omega_D \cap \overline{\Omega}_Q.
\end{aligned}
\tag{6.6}
$$

We denote the corresponding operator by $\mathcal{H}_{\xi\to\Omega_Q}(\cdot)$ such that

$$v_\xi = \mathcal{H}_{\xi\to\Omega_Q}(\tau_\xi).$$

The domain of $\mathcal{H}_{\xi\to\Omega_Q}(\cdot)$ can be extended to any finite element function by using the function's restriction to ξ before the application of $\mathcal{H}_{\xi\to\Omega_Q}(\cdot)$. A solution v_ξ satisfies

$$a_{\Omega_Q}(\, v_\xi \, , \, v_\xi \,) = \min_{\substack{w \in V^h(\Omega_Q) \\ w|_\xi - \tau_\xi = 0 \\ w|_{\partial\Omega_D \cap \overline{\Omega}_Q} = 0}} a_{\Omega_Q}(w, w); \tag{6.7}$$

see lemma A.2.

Let us note that in some cases the bilinear form $a_{\Omega_Q}(\cdot, \cdot)$ is not positive definite on $V^h_{0,\xi,\partial\Omega_D}(\Omega_Q)$. Thus, $\mathcal{H}_{\xi\to\Omega_Q}(\cdot)$ is possibly not uniquely defined; cf. remark 6.5.

We point out that—contrary to [HKKR19; HKK+22]—here, we enforce a zero Dirichlet boundary condition on $\partial\Omega_D$ in (6.6). For problems where large parts of $\partial\Omega$ have a Neumann condition—as, for example, model problem **(2)** in section 2.2—this change has a negligible effect. However, for other types of problems where on large parts of the domain boundary a Dirichlet condition is enforced, a significant reduction in the coarse space dimension can be observed.

Remark 6.1 (Neumann Boundary Condition)**.** *In (6.6), we do not impose a Dirichlet condition on the boundary of Q, except for the part that intersects $\partial\Omega_D$. As a result, a Neumann boundary condition is enforced, which is essential for our type of eigenvalue*

● node of ξ ◯ node of $\overline{\Omega}_\xi \setminus \xi$
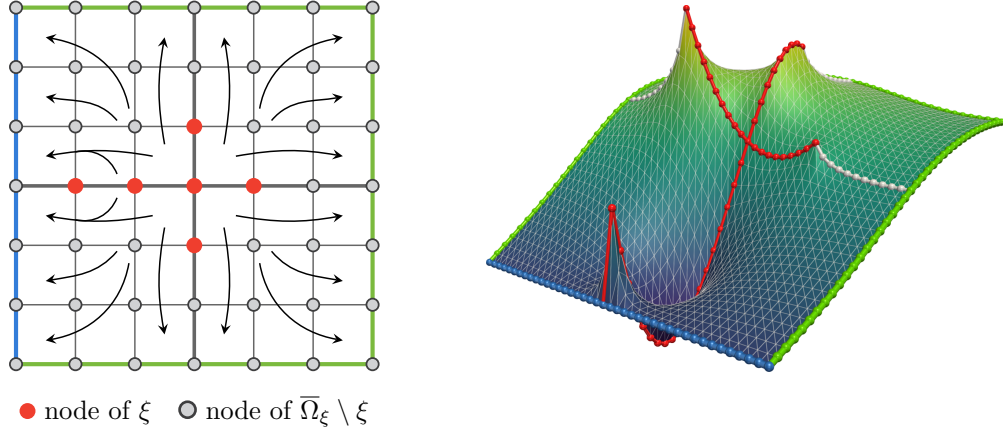
Figure 6.1.: (Compare with [HKK+22, fig. 4].) Dirichlet boundary $\partial\Omega_D$ in blue (on the left of the domain); ξ in red; Neumann boundary of $\mathcal{H}_{\xi\to\Omega_\xi}(\cdot)$ in green. **(Left)** Two-dimensional schematic of the energy-minimizing extension $\mathcal{H}_{\xi\to\Omega_\xi}(\cdot)$ defined by (6.6). **(Right)** Sample extension for the diffusion equation with a homogeneous coefficient function. Nodes of $\Gamma^h \setminus (\xi \cup \partial\Omega_\xi)$ in white.

problem to construct a robust preconditioner. For the proof of the condition number bound, we will use an inequality based on spectral estimates; see lemma 6.2. For its proof, we make use of the energy-minimizing property in (6.7). If we used a zero Dirichlet boundary condition on $\partial\Omega_Q$, it would introduce a forced slope to zero and, thus, a larger energy. In that case, we would not be able to prove lemma 6.2; counterexamples of coefficient functions that show the necessity of the Neumann boundary are given by connected patches of large coefficients that intersect multiple interface components as in fig. 1.7 (left) and fig. 1.8 (left); cf. the discussion for fig. 6.2 below.

Example 6.1. *Figure 6.1 (left) shows the schematic of an energy-minimizing extension from an interface star to its four neighboring subdomains. Two of the subdomains touch the Dirichlet boundary $\partial\Omega_D$ (in blue), which introduces a forced slope to zero, increasing the energy of an extension. For the two-dimensional Laplace equation, a sample energy-minimizing extension (i.e., a discrete harmonic extension) is shown in fig. 6.1 (right).*

6. Theory of Adaptive GDSW-Type Coarse Spaces

Using the definition of the energy-minimizing extension, we can now define a symmetric, positive semidefinite bilinear form that is associated with the left-hand side of generalized eigenvalue problem (4.2):

$$\alpha_\xi^K(u, v) := a_{\Omega_\xi}\big(\mathcal{H}_{\xi \to \Omega_\xi}(u),\, \mathcal{H}_{\xi \to \Omega_\xi}(v)\big) \quad \forall u, v \in X^h(\xi). \tag{6.8}$$

On the right-hand side of the AGDSW generalized eigenvalue problem (3.1), a submatrix of the fully assembled stiffness matrix K corresponding to the degrees of freedom of a coarse edge or face is used. For RAGDSW, in (4.2), the submatrix corresponding to an interface star is used and further decoupled; cf. section 4.2. Extracting the submatrix is equivalent to assembling a stiffness matrix on the respective patch of finite elements adjacent to ξ and enforcing a zero Dirichlet condition on the boundary of the patch. Thus, in a variational setting, we define an operator $z_\xi(\cdot)$ that extends by zero from ξ to $V^h(\Omega)$:

$$z_\xi \colon X^h(\xi) \to V^h(\Omega)$$

$$v \mapsto z_\xi(v) := \begin{cases} v(x^h) & \forall x^h \in \xi, \\ 0 & \forall x^h \in \overline{\Omega} \setminus \xi. \end{cases} \tag{6.9}$$

The domain of $z_\xi(\cdot)$ can be extended to any finite element function by using the function's restriction to ξ before the application of $z_\xi(\cdot)$. For AGDSW, the bilinear form of the right-hand side of (3.1) is given by the symmetric, positive definite bilinear form

$$a_{\Omega_\xi}\big(z_\xi(u),\, z_\xi(v)\big) \quad \forall u, v \in X^h(\xi).$$

For RAGDSW, we need to decouple the bilinear form with respect to the NECs ξ_i of $\xi \in \mathcal{P}$; see section 5.6 for details regarding the subcomponents ξ_i. As in section 4.2, let the set of all NECs of a ξ be given by $\{\xi_i\}_{i=1}^{n_\xi}$. We define the symmetric, positive definite bilinear form

$$\beta_\xi^K(u, v) := \sum_{i=1}^{n_\xi} \beta_{\xi_i}^K(u, v) \quad \forall u, v \in X^h(\xi), \tag{6.10}$$

where

$$\beta_{\xi_i}^K(u, v) := a_{\Omega_{\xi_i}}\big(z_{\xi_i}(u),\, z_{\xi_i}(v)\big) \quad \forall u, v \in X^h(\xi_i).$$

As before, the domain of $\beta_{\xi_i}^K(\cdot, \cdot)$ can be extended to $V^h(\Omega) \times V^h(\Omega)$ by using a function's restriction to ξ_i before the application of $\beta_{\xi_i}^K(\cdot, \cdot)$.

110

Remark 6.2. *Note that the sets in $\mathcal{V}$, $\mathcal{E}$, and $\mathcal{F}$ of the GDSW interface partition are NECs and, thus, we have $\beta_\xi^K(u,v) = a_{\Omega_\xi}(z_\xi(u), z_\xi(v))$ if ξ is associated with a coarse node, edge, or face.*

Remark 6.3. *If the lumped variant of section 3.3.1 is used—that is, the diagonal of the matrix $K_{\xi\xi}^{\Omega_\xi}$ is used on the right-hand side of (4.2)—the corresponding bilinear form is given by*

$$\beta_\xi^K(u,v) := \sum_{x^h \in \xi} a_{\Omega_{\{x^h\}}}(z_{\{x^h\}}(u), z_{\{x^h\}}(v)) \quad \forall u,v \in X^h(\xi), \tag{6.11}$$

where $\{x^h\}$, $x^h \in \xi$, can be regarded as a subcomponent of ξ that contains only a single finite element node. For simplicity, we will always work with the subsets ξ_i of ξ in the following. However, the proof is carried out identically if $\{\xi_i\}_{i=1}^{n_\xi}$ is replaced by $\{x^h\}_{x^h \in \xi}$.

The variational formulation of the RAGDSW generalized eigenvalue problem in (4.2) is given by: find $\tau_{*,\xi} \in X^h(\xi)$ such that

$$\alpha_\xi^K(\tau_{*,\xi}, \theta) = \lambda_{*,\xi}\, \beta_\xi^K(\tau_{*,\xi}, \theta) \quad \forall \theta \in X^h(\xi). \tag{6.12}$$

As $\alpha_\xi^K(\cdot,\cdot)$ is symmetric, positive semidefinite, and $\beta_\xi^K(\cdot,\cdot)$ symmetric, positive definite, there exist eigenpairs $\{(\tau_{k,\xi}, \lambda_{k,\xi})\}_{k=1}^m$ such that $\beta_\xi^K(\tau_{k,\xi}, \tau_{j,\xi}) = \delta_{k,j}$, where $\delta_{k,j}$ is the Kronecker delta, and where $m = \dim(X^h(\xi))$ denotes the number of unknowns of ξ; see lemma A.3.

As before, let the eigenvalues of (6.12) be sorted in nondescending order,

$$0 \le \lambda_{1,\xi} \le \lambda_{2,\xi} \le \cdots \le \lambda_{m,\xi}.$$

We select all eigenfunctions $\tau_{*,\xi}$ from (6.12) that correspond to eigenvalues smaller than or equal to a user-prescribed threshold $tol_\xi \ge 0$ (to obtain a condition number bound that is finite, the tolerance must be positive),

$$\lambda_{*,\xi} \le tol_\xi,$$

and extend them by zero to the interface,

$$\tau_{*,\xi,\Gamma^h} := z_\xi(\tau_{*,\xi})|_{\Gamma^h}.$$

The coarse basis functions associated with ξ are then given by the energy-minimizing extensions

$$v_{*,\xi} := \mathcal{H}_{\Gamma^h \to \Omega}(\tau_{*,\xi,\Gamma^h}) \in V^h_{0,\partial\Omega_D}(\Omega),$$

where $\mathcal{H}_{\Gamma^h \to \Omega}(\cdot)$ is defined by using $\xi = \Gamma^h$ and $\Omega_Q = \Omega$ in (6.6). As we have remarked in section 1.4.1, the extension $\mathcal{H}_{\Gamma^h \to \Omega}(\cdot)$ can be computed to each subdomain individually and, thus, in parallel. We note that, contrary to the local energy-minimizing extensions, $\mathcal{H}_{\Gamma^h \to \Omega}(\cdot)$ is always uniquely defined as a solution vanishes on $\partial\Omega_\xi \cap (\Gamma^h \cup \partial\Omega^h_D)$, which is sufficient to remove any rigid body modes from the problem in the case of linear elasticity.

For an interface partition $\mathcal{P}$, an adaptive GDSW-type coarse space is defined as

$$V_{\mathcal{P}} := \bigoplus_{\xi \in \mathcal{P}} \text{span}\{\, v_{*,\xi} : \lambda_{*,\xi} \leq tol_\xi \,\}. \tag{6.13}$$

Remark 6.4 (GDSW Vertex Functions and Zero Eigenvalues)**.** *The left-hand side of (6.12) is singular in case $\overline{\Omega}_\xi$ does not intersect $\partial\Omega_D$ (and in some special cases if rotation modes are in the null space). The null space is given by the constant functions in the case of the scalar diffusion equation and by rigid body modes in the case of linear elasticity. Hence, the null space dimension equals one for the scalar diffusion problem and, in the case of three-dimensional linear elasticity, three if ξ is a vertex, five if ξ is a straight edge, and six in all other cases.*

For a vertex $\nu \in \mathcal{V}$, the problem has one (in the case of scalar diffusion) and three (in the case of three-dimensional linear elasticity) degrees of freedom. Thus, coarse functions associated with the zero eigenvalues are given by GDSW vertex functions.

Remark 6.5 (Nonuniqueness of Some Local Energy-Minimizing Extensions)**.** *In some cases, an energy-minimizing extension may only be uniquely defined up to some null space element. For the considered model problems, this can only occur for linear elasticity problems and in case the finite element nodes of ξ and $\partial\Omega_D \cap \overline{\Omega}_\xi$ are given by a single node or nodes that lie on a straight edge; see also figs. 1.3 and 4.5. For linear elasticity, null space elements are given by translations and linearized rotations; see remark 1.4. Let us assume that the adjacent subdomains of ξ do not touch $\partial\Omega_D$. If ξ is a vertex, v_ξ a solution to (6.6), and r a linearized rotation around the vertex, then the sum $v_\xi + r$ is a*

solution to (6.6) as well. Equally, a solution to an energy-minimizing extension from a straight edge is only uniquely defined up to the linearized rotation mode around the edge. Thus, in the above cases, the null space of $\mathcal{H}_{\xi \to \Omega_Q}(\cdot)$ is given by one or three linearized rotations.

However, all energy-minimizing solutions v_ξ have the same minimal energy $a_{\Omega_\xi}(v_\xi, v_\xi)$ (see below). As a result, the nonuniqueness of energy-minimizing solutions does not have an influence on the left-hand side of (6.12), and it follows that the Schur complement (3.2) is uniquely defined. Therefore, the particular choice of an operator $\mathcal{H}_{\xi \to \Omega_Q}(\cdot)$ does not change the generalized eigenvalue problem or its solution. We note that this holds true for the Schur complement (3.4) of the S-variant in section 3.3.3 as well; see section 6.1.3 for its variational formulation.

Let $v_\xi = \mathcal{H}_{\xi \to \Omega_Q}(\tau_\xi)$ be a solution to (6.6). For linear elasticity, a rigid body mode r is then given by a solution to (6.6) given that $r|_\xi = 0$. Thus, all solutions to (6.6) are given by $v_\xi + r$. Since $r \in V_{0,\xi,\partial\Omega_D}^h(\Omega_Q)$, r can act as a test function for $\mathcal{H}_{\xi \to \Omega_Q}(\theta)$, $\theta \in X^h(\xi)$. Thus, we have $a_{\Omega_Q}(r, \mathcal{H}_{\xi \to \Omega_Q}(\theta)) = 0$ and obtain the equality

$$a_{\Omega_Q}(v_\xi + r, \mathcal{H}_{\xi \to \Omega_Q}(\theta)) = a_{\Omega_Q}(v_\xi, \mathcal{H}_{\xi \to \Omega_Q}(\theta)) \quad \forall \theta \in X^h(\xi).$$

As a consequence, any operator $\mathcal{H}_{\xi \to \Omega_Q}(\cdot)$ defined by (6.6) yields the same generalized eigenvalue problem (6.12). In section 4.4, we have given options for finding a solution to (6.6) in case it is not uniquely defined.

In section 1.5, we have presented various cases to gain an understanding of some key characteristics of coefficient functions. For example, we have seen that the number of channels (of large coefficients) intersecting a coarse edge correlates one-to-one with the required number of coarse functions. We can now give an intuitive explanation for the choice of the left- and right-hand sides of the generalized eigenvalue problems without having to refer to the proof of the condition number bound. For this, we note that by definition of the energy-minimizing extension—and in case ξ is a NEC—we have

$$\alpha_\xi^K(\tau_\xi, \tau_\xi) \leq \beta_\xi^K(\tau_\xi, \tau_\xi) \quad \forall \tau_\xi \in X^h(\xi), \tag{6.14}$$

or its matrix equivalent

$$\tau_\xi^T S_{\xi\xi} \tau_\xi \leq \tau_\xi^T \widetilde{K}_{\xi\xi} \tau_\xi \quad \forall \tau_\xi \in X^h(\xi).$$

From this follows that all eigenvalues of (6.12) are smaller than or equal to one; this holds true as well for the S-variant and the slab variant but not the variant using a mass matrix (cf. sections 6.1.2 to 6.1.4), nor does it hold for the variants using a lumped stiffness or mass matrix (cf. remarks 6.3 and 6.6). For general interface components ξ and if β_ξ^K is defined by (6.10), an upper bound for the largest eigenvalue is instead given by n_ξ (the number of subcomponents of ξ).

Consider a channel of a large coefficient that intersects a coarse edge $e \in \mathcal{E}$ as in fig. 6.2. Furthermore, let a function τ_e on the coarse edge be given that is constant on the patch of the large coefficient. Then its energy-minimizing extension $\mathcal{H}_{e \to \Omega_e}(\tau_e)$ has a small energy. On the other hand, its extension-by-zero $z_e(\tau_e)$ has a large energy as it quickly descends to zero on a patch of large coefficients. As a result, if the function were an eigenfunction, its eigenvalue would be small. As we will see in section 9.2, we can use heuristically constructed functions similar to τ_e for a coarse space without having to solve an eigenvalue problem.

We remark that the reasoning above also motivates why we have to decouple the right-hand side of the RAGDSW eigenvalue problem; see section 6.1.1.

We consider the coefficient function in fig. 6.2 (left). If a zero Dirichlet condition were prescribed on $\partial \Omega_Q$ in (6.6), the energy of both the energy-minimizing extension and the extension-by-zero would be large; cf. fig. 6.2. As a result, we would not obtain small eigenvalues and, thus, could not detect useful eigenfunctions; cf. remark 6.1. Further examples of coefficient functions similar to that in fig. 6.2 that have structures intersecting not only the corresponding ξ but also other interface components are given in fig. 1.7 (left) and fig. 1.8 (left).

Similarly, the inclusion of the Dirichlet boundary condition on $\partial \Omega_D$ introduces a forced slope to zero toward $\partial \Omega_D$. Thus, patches of large coefficients that touch the Dirichlet boundary do not lead to small eigenvalues if Ω_ξ touches $\partial \Omega_D$. This corresponds to our findings in section 1.5.3.
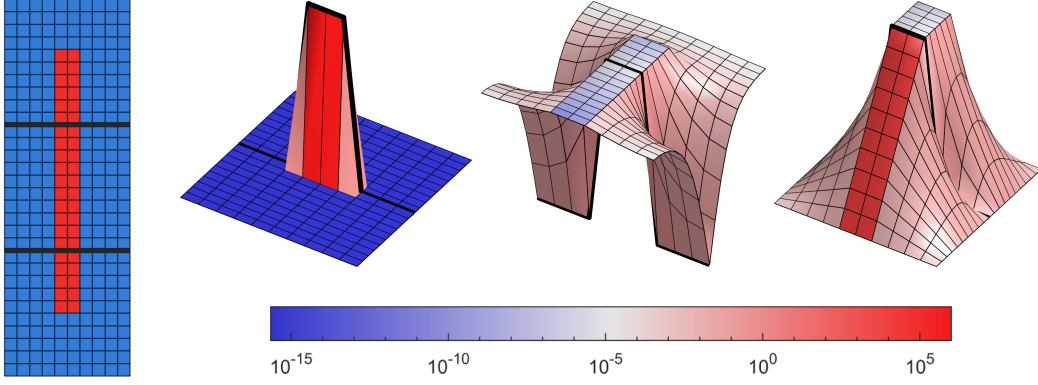
Figure 6.2.: **(Left)** Cofficient function ($E = 10^6$ in red, $E = 1$ in blue) for a diffusion problem on a mesh with two coarse edges (thick, black lines). A zero Dirichlet boundary condition is prescribed at the top of the domain. **(Center left to center right)** Extensions from the bottom coarse edge (to the bottom two subdomains) of a function τ_e that is equal to one on the nodes of e that are associated with a large coefficient and zero elsewhere. Colors (log-scaled) indicate the energy $a_T(v, v)$ of an extension on each finite element T. **(Center left)** Extension-by-zero $z_e(\tau_e)$; we have $a_{\Omega_e}(z_e(\tau_e), z_e(\tau_e)) = 4{\cdot}10^6$. **(Center right)** Energy-minimizing extension $\mathcal{H}_{e\to\Omega_e}(\tau_e)$; we have $a_{\Omega_e}(\mathcal{H}_{e\to\Omega_e}(\tau_e), \mathcal{H}_{e\to\Omega_e}(\tau_e)) = 6.4$. **(Right)** Energy-minimizing extension if a zero Dirichlet condition were prescribed on $\partial\Omega_e$; we have $a_{\Omega_e}(v, v) = 2{\cdot}10^5$ for the extension v.

6.1.1. Why a Decoupling is Required for RAGDSW

The decoupling of $K_{\xi\xi}^{\Omega_\xi}$ is essential to obtain a robust preconditioner, which is also reflected by the proof of lemma 6.5. Therein, the function Ψ is constant on ξ_i and, thus, can be moved outside of the corresponding seminorm in (6.40) without obtaining a contrast dependent estimate. Note that Ψ cannot be chosen to be constant on ξ; cf. the definition of the partition of unity in the proof of lemma 6.6.

In fact, without the decoupling, we can construct a mesh and coefficient function for which the algorithm fails to obtain a small condition number. We consider four two-
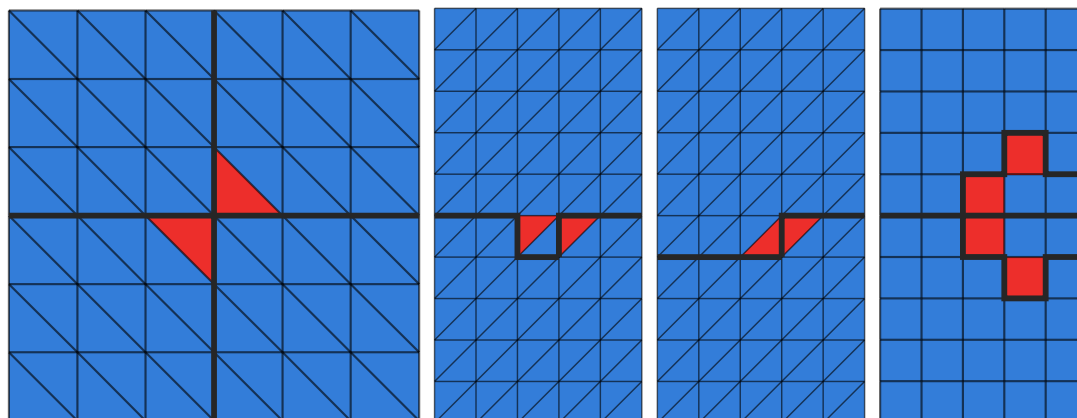
Figure 6.3.: Four domain decompositions (interface marked with a thick, black line) and coefficient functions ($E = 10^6$ in red, $E = 1$ in blue). Zero Dirichlet boundary condition on $\partial\Omega$.

dimensional diffusion problems in fig. 6.3. For all problems, we use only a single interface component. Let a constant function on the interface be given. Then its extension-by-zero is constant on all elements of large coefficients in fig. 6.3. Thus, the energy of the extension is small—it is zero on the elements with large coefficients, since the gradient of the constant function is zero—as is the energy of the energy-minimizing extension. As a consequence, the eigenvalue problem is blind to all coefficient functions in fig. 6.3. By decoupling the right-hand side of the generalized eigenvalue problem with respect to the NECs, we can obtain a large energy on the right-hand side for some of the problems.

For the coefficient function in fig. 6.3 (left), the smallest eigenvalue is given by 0.442. A coarse function would (generally) not be constructed, in which case we obtain a condition number of 384 620.1 if an overlap of one layer of finite elements is used. If we increase the size of the overlap to two layers, the entire structure of large coefficients is taken care of by the first level of the overlapping Schwarz preconditioner (cf. section 1.5.4), and we obtain a condition number of 4.3.

The situation is more favorable in fig. 6.3 (center left and center right) as all nodes that are associated with large coefficients are taken care of by the first level of the Schwarz preconditioner using an overlap of only one layer of finite elements. This is in accordance

with the decoupling of the right-hand side that only separates components into their respective NECs (coarse edges and faces need not be decoupled).

In fig. 6.3 (left), the first level of the Schwarz preconditioner was sufficient to obtain a small condition number if an overlap of two layers of finite elements was used. This raises the question whether this is always the case. However, we can always construct an example that requires an arbitrarily large overlap. An example, for which an overlap of three layers of finite elements is required, is given in fig. 6.3 (right). The condition numbers for an overlap of one, two, and three layers of finite elements are $302\,679.9$, $69\,123.1$, and 2.2.

If we lump the stiffness matrix (cf. remark 6.3)—which amounts to a decoupling with respect to every finite element node of the interface component—for the coefficient functions in fig. 6.3 (center left and center right), we obtain a large energy on the right-hand side and, thus, will unneccessarily construct coarse functions. The situation is similar if a mass matrix or a lumped mass matrix is used.

6.1.2. Variant Using Slabs around Interface Components

One of the key properties of our coarse spaces is the use of the energy-minimizing extension defined by (6.6). The incorporation of the energy-minimizing extension into the generalized eigenvalue problem is not necessary to construct a robust preconditioner; for example, in [GLR15; EMR19], the authors define the left-hand side of a generalized eigenvalue problem associated with a coarse edge based on a diffusion problem along the respective coarse edge. We can achieve a similar effect for our coarse spaces by using a minimal slab of one layer of finite elements. However, as we have seen in sections 3.3.2, 3.4.1, and 4.5, increasing the size of the slab can significantly reduce the coarse space dimension, as this allows the eigenvalue problem to detect connected patches of large coefficients.

Let Ω_ξ^l denote the slab of l layers of finite elements around ξ; cf. fig. 6.4. The generalized eigenvalue problem for the slab variant is then given by (6.12) if we substitute the left-hand
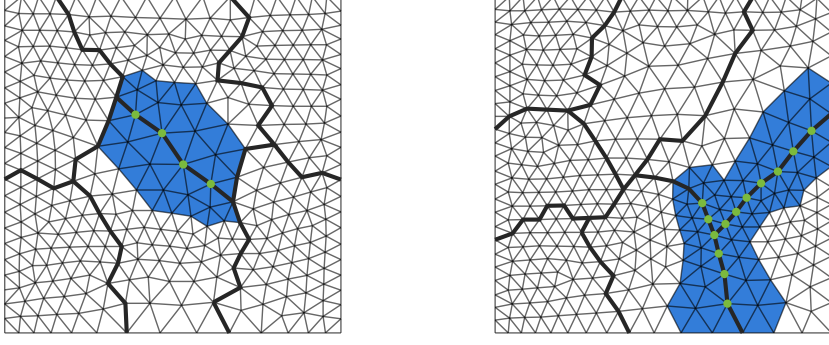
Figure 6.4.: Slab of two layers of finite elements around a coarse edge **(left)** and a coarse star **(right)**.

side with $\alpha_\xi^{K,l}(\cdot, \cdot)$, where

$$\alpha_\xi^{K,l}(u, v) := a_{\Omega_\xi^l}\left(\mathcal{H}_{\xi \to \Omega_\xi^l}(u), \mathcal{H}_{\xi \to \Omega_\xi^l}(v)\right) \quad \forall u, v \in X^h(\xi). \tag{6.15}$$

We then have for $u \in X^h(\xi)$

$$\begin{aligned}
\alpha_\xi^{K,l}(u, u) &\leq a_{\Omega_\xi^l}\left(\mathcal{H}_{\xi \to \Omega_\xi}(u), \mathcal{H}_{\xi \to \Omega_\xi}(u)\right) \\
&\leq a_{\Omega_\xi}\left(\mathcal{H}_{\xi \to \Omega_\xi}(u), \mathcal{H}_{\xi \to \Omega_\xi}(u)\right) = \alpha_\xi^K(u, u).
\end{aligned} \tag{6.16}$$

We will later require this property to work with $\alpha_\xi^K(\cdot, \cdot)$ rather than $\alpha_\xi^{K,l}(\cdot, \cdot)$.

6.1.3. Variant Using a Sum of Local Schur Complements

Let $\xi \in \mathcal{P}$. Then we define the sum of local energy-minimizing extensions of a function $\theta \in X^h(\xi)$ as

$$\mathcal{H}_{\xi \to \Omega_\xi}^S(\theta) := \begin{cases} \mathcal{H}_{\xi \to \Omega_k}(\theta) & \text{in } \Omega_k, k \in n(\xi), \\ 0 & \text{in } \Omega \setminus \Omega_\xi; \end{cases} \tag{6.17}$$

cf. fig. 6.5. We note that $\mathcal{H}_{\xi \to \Omega_\xi}^S(\theta)$ is discontinuous in general. The designation of (6.17) as a sum is more easily motivated by (6.18) and the matrix formulation in (3.4).

Accordingly, the left-hand side of (6.12) is replaced by

$$\alpha_\xi^S(u, v) := \sum_{k \in n(\xi)} a_{\Omega_k}\left(\mathcal{H}_{\xi \to \Omega_k}(u), \mathcal{H}_{\xi \to \Omega_k}(v)\right), \tag{6.18}$$

● node of ξ ○ node of $\overline{\Omega}_\xi \setminus \xi$
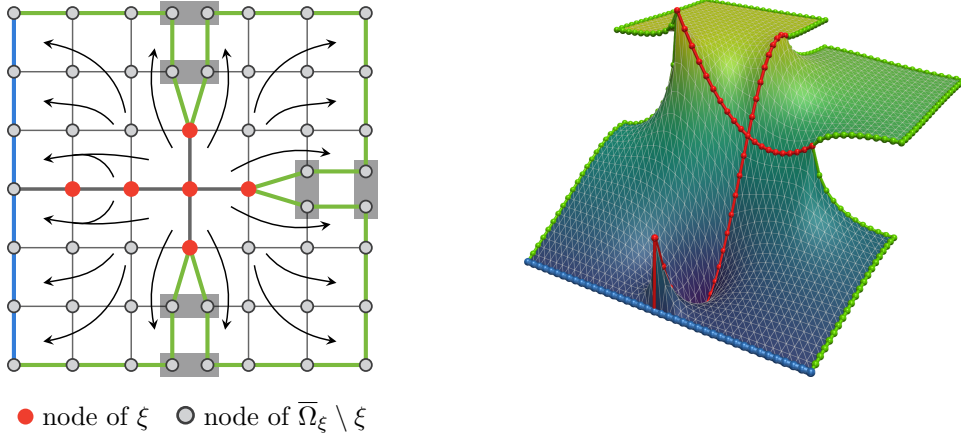
Figure 6.5.: (Compare with [HKK+22, fig. 5].) Analogue of fig. 6.1 for RAGDSW–S. Dirichlet boundary $\partial\Omega_D$ in blue (on the left of the domain); ξ in red; Neumann boundary of $\mathcal{H}^S_{\xi\to\Omega_\xi}(\cdot)$ in green. **(Left)** Schematic of the sum of energy-minimizing extensions $\mathcal{H}^S_{\xi\to\Omega_\xi}(\cdot)$. Gray boxes indicate interface nodes that have been torn apart for visualization purposes. **(Right)** Sample extension for the diffusion equation with a homogeneous coefficient function.

where $u, v \in X^h(\xi)$. The bilinear form $\alpha^S_\xi(\cdot,\cdot)$ satisfies

$$\alpha^S_\xi(u,u) \le \sum_{k\in n(\xi)} a_{\Omega_k}\big(\mathcal{H}_{\xi\to\Omega_\xi}(u), \mathcal{H}_{\xi\to\Omega_\xi}(u)\big) = \alpha^K_\xi(v,v) \quad \forall u \in V^h(\Omega). \qquad (6.19)$$

As for the slab variant, we will later require this property to work with $\alpha^K_\xi(\cdot,\cdot)$ rather than $\alpha^S_\xi(\cdot,\cdot)$. By combining the S-variant and the slab variant, we obtain

$$\alpha^{S,l}_\xi(u,v) \coloneqq \sum_{k\in n(\xi)} a_{\Omega^l_k}\big(\mathcal{H}_{\xi\to\Omega^l_k}(u), \mathcal{H}_{\xi\to\Omega^l_k}(v)\big). \qquad (6.20)$$

By (6.16), we then have for $u \in V^h(\Omega)$

$$\alpha^{S,l}_\xi(u,u) = \sum_{k\in n(\xi)} a_{\Omega^l_k}\big(\mathcal{H}_{\xi\to\Omega^l_k}(u), \mathcal{H}_{\xi\to\Omega^l_k}(u)\big) \le \alpha^{K,l}_\xi(u,u) \le \alpha^K_\xi(u,u). \qquad (6.21)$$

In section 3.3.3, we analyzed the S-variant for a few coefficient functions to demonstrate that using the S-variant can result in a larger coarse space dimension. An increase in the coarse space dimension is explained by the decoupling of the energy-minimizing extension
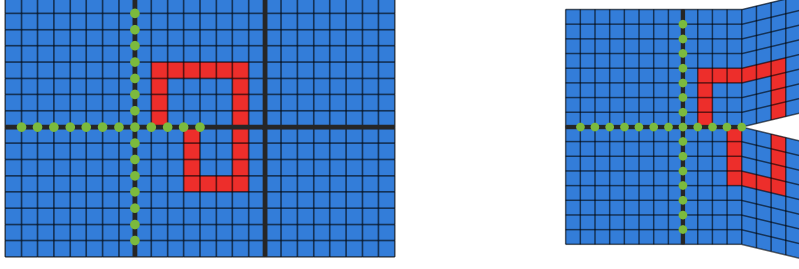
Figure 6.6.: **(Left)** 2×3 subdomains, surrounded by a Dirichlet boundary condition, and two coarse stars, one highlighted with green disks (ξ). Coefficient function $E = 1$ in blue and $E = 10^6$ in red. **(Right)** Schematic for RAGDSW–S of the problem on the left with respect to the highlighted interface component. The corresponding energy-minimizing extension is torn apart at $(\Gamma \setminus \xi) \cap \overline{\Omega}_\xi$.

and $\alpha_\xi^S(\cdot, \cdot)$, respectively; cf. figs. 6.5 and 6.6. In fig. 6.6, the effect of decoupling the energy-minimizing extension is visualized: On the left, the coefficient function consists of one connected patch of large coefficients. On the right—as a consequence of the decoupling—it consists of two connected components, which results in the construction of one additional coarse function.

6.1.4. Variant Using a Scaled Mass Matrix

Let an interface component $\xi \in \mathcal{P}$ be given. We replace the right-hand side of generalized eigenvalue problem (6.12) with the scaled L^2-inner product

$$\beta_\xi^M(u, v) := b_{\Omega_\xi}\big(z_\xi(u), z_\xi(v) \big), \quad u, v \in X^h(\xi), \tag{6.22}$$

where

$$b_{\Omega_\xi}(u, v) := \sum_{T \in \tau_h(\Omega_\xi)} \frac{E(T)}{\hat{h}_T^2} \int_T u(x) \cdot v(x)\, \mathrm{d}x, \quad u, v \in V^h(\Omega).$$

As before, we extend the domain of $\beta_\xi^M(\cdot, \cdot)$ to $V^h(\Omega) \times V^h(\Omega)$. As remarked in section 3.3.4, we use $\hat{h}_T = h_T$ for the theoretical analysis of our coarse spaces but choose $\hat{h}_T$ as the radius of the largest insphere of T to obtain numerical results—we recall that $\tau_h(\Omega)$ is shape-regular.

For the theory in sections 6.2 and 6.3 to be applicable to the mass matrix variant, we need to bound $\beta_\xi^K(u, u)$ from above by $C\beta_\xi^M(u, u)$ for any $u \in X^h(\xi)$ and some constant C.

First, we show the bound for AGDSW, for which the right-hand side of generalized eigenvalue problem (6.12) is given by $\beta_\xi^K(u, v) = a_{\Omega_\xi}(z_\xi(u), z_\xi(u))$. Then, by an inverse inequality (cf. [TW05, lemma B.27; BS08, lemma 4.5.3]), and since E is constant on each element $T \in \tau_h(\Omega)$, we have

$$a_{\Omega_\xi}(z_\xi(u), z_\xi(u)) = \sum_{T \in \tau_h(\Omega_\xi)} E(T)|z_\xi(u)|_{H^1(T)}^2$$

$$\leq C_{\text{inv},1} \sum_{T \in \tau_h(\Omega_\xi)} \frac{E(T)}{h_T^2}\|z_\xi(u)\|_{L^2(T)}^2$$

$$= C_{\text{inv},1}\beta_\xi^M(u, u) \quad \forall u \in X^h(\xi),$$

where the constant $C_{\text{inv},1}$ is independent of the diameter of the finite elements and the coefficient function E.

For the reduced-dimension coarse spaces, the right-hand side of generalized eigenvalue problem (6.12) is given by $\beta_\xi^K(u, v) = \sum_{i=1}^{n_\xi} a_{\Omega_{\xi_i}}(z_{\xi_i}(u), z_{\xi_i}(v))$. Note that it is not necessary to decouple the mass term as was explained in section 6.1.1. Let $u \in X^h(\xi)$. Then

$$\sum_{i=1}^{n_\xi} a_{\Omega_{\xi_i}}(z_{\xi_i}(u), z_{\xi_i}(u)) \leq C_{\text{inv},1} \sum_{i=1}^{n_\xi} b_{\Omega_{\xi_i}}(z_{\xi_i}(u), z_{\xi_i}(u))$$

$$= C_{\text{inv},1} \sum_{T \in \tau_h(\Omega_{\xi_i})} \frac{E(T)}{h_T^2} \sum_{i=1}^{n_\xi} \|z_{\xi_i}(u)\|_{L^2(T)}^2. \tag{6.23}$$

$\sum_{i=1}^{n_\xi} \|z_{\xi_i}(u)\|_{L^2(T)}^2$ can be bounded by an L^∞ term multiplied with the measure $|T|$ of the element T. Then, we can use the fact that

$$\Big\|\sum_{i=1}^{n_\xi} (z_{\xi_i}(u))^2\Big\|_{L^\infty(T)} = \|(z_\xi(u))^2\|_{L^\infty(T)} \tag{6.24}$$

in the scalar case; cf. fig. 6.7. Using an inverse inequality (cf. [BS08, lemma 4.5.3]) and $|T| \leq h_T^d$, we obtain

$$\sum_{i=1}^{n_\xi} \|z_{\xi_i}(u)\|_{L^2(T)}^2 \leq C_{\text{inv},2}\Big\|\sum_{i=1}^{n_\xi} z_{\xi_i}(u)\Big\|_{L^2(T)}^2 = C_{\text{inv},2}\|z_\xi(u)\|_{L^2(T)}^2. \tag{6.25}$$
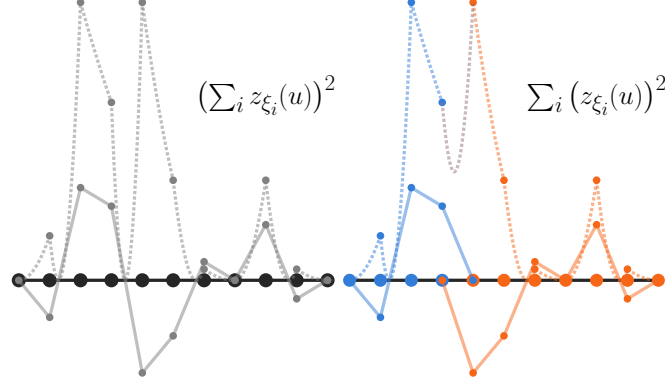
Figure 6.7.: Example of (6.24) in one dimension. **(Left)** Square of the sum of extensions from subcomponents (dotted line) and sum of extensions from subcomponents (solid line). **(Right)** Sum of the square of extensions from subcomponents (dotted line) and extensions from subcomponents (solid lines).

For details and the vector case, see lemma A.4 and corollaries A.1 and A.2. In the following, we assume that $C_{\mathrm{inv},2}$ is the maximum constant of all $T \in \tau_h(\Omega)$, which is independent of h_T. Combining (6.23) and (6.25) gives

$$\beta_\xi^K(u,u) = \sum_{i=1}^{n_\xi} a_{\Omega_{\xi_i}}\left(z_{\xi_i}(u),\, z_{\xi_i}(u)\right) \leq C_{\mathrm{inv}} b_\xi^M(u,u), \tag{6.26}$$

where $C_{\mathrm{inv}} := C_{\mathrm{inv},1} C_{\mathrm{inv},2}$.

Remark 6.6. *If a lumped mass matrix is used as in section 3.3.4, the bilinear form is given by*

$$\beta_\xi^M(u,v) := \sum_{x^h \in \xi} b_{\Omega_{\{x^h\}}}\left(z_{\{x^h\}}(u),\, z_{\{x^h\}}(v)\right) \quad \forall u,v \in X^h(\xi); \tag{6.27}$$

see remark 6.3 for the case of a lumped stiffness matrix. We have

$$\beta_\xi^K(u,u) \leq C_{\mathrm{inv},1} C_\tau \beta_\xi^M(u,u) \quad \forall u \in X^h(\xi),$$

where C_τ denotes the maximum number of vertices of a finite element; cf. (6.39). Note, however, that C_τ can be removed by using

$$\beta_\xi^K(u,v) := \sum_{x^h \in \xi} a_{\Omega_{\{x^h\}}}\left(z_{\{x^h\}}(u),\, z_{\{x^h\}}(v)\right)$$

during the proof of a condition number bound; cf. remark 6.3. To simplify the notation, we henceforth identify ξ_i with $x^h \in \xi$, in case the lumped variant is used.

6.2. Local Spectral Projections

The following lemma is central to the proof of all condition number bounds of the coarse spaces in chapters 3, 4, and 7, in addition to the ones in [GE10b; EGLW12; DNSS12; SDH+14a; GLR15; EMR19]; see [SDH+14a, lemma 2.11] and also [EGLW12, eq. (2.8); DNSS12, theorem 3.1; GLR15, lemma 2.2; HKKR18b, lemma 4.1; EMR19, lemma 3; HKKR19, lemma 5.3; HKK+22, lemma 10.1]; see also [BHMV99] for related work. It allows us to bound—independent of the coefficient contrast—a large-energy term $\beta(\cdot, \cdot)$ by a small-energy term $\alpha(\cdot, \cdot)$; see (6.29) and cf. (6.14). Of course, this involves a penalty, namely the factor tol^{-1}, where tol should be chosen as small as possible to obtain a small coarse space dimension.

Lemma 6.1 (Spectral Projection Estimate)**.** *Let $\alpha(\cdot, \cdot)$ be a symmetric, positive semidefinite bilinear form and $\beta(\cdot, \cdot)$ a symmetric, positive definite bilinear form on the finite-dimensional vector space X. Furthermore, let $tol > 0$ be a tolerance for the selection of eigenfunctions. We consider the generalized eigenvalue problem: find $v \in X$ such that*

$$\alpha(v, w) = \lambda \beta(v, w) \quad \forall w \in X.$$

Then there exist eigenpairs $\{(v_k, \lambda_k)\}_{k=1}^{\dim(X)}$, $\lambda_k \geq 0$, such that the eigenvectors are a β-orthonormal basis of X:

$$\alpha(v_k, w) = \lambda_k \beta(v_k, w) \quad \forall w \in X, \ 1 \leq k \leq \dim(X),$$

$$\beta(v_k, v_l) = \delta_{k,l}, \qquad 1 \leq k, l \leq \dim(X),$$

where $\delta_{k,l}$ is the Kronecker delta. Let $u \in X$. We define the spectral projection

$$\Pi u := \sum_{\lambda_k \leq tol} \beta(u, v_k) v_k$$

and the semi-norm and norm

$$|u|_\alpha := \sqrt{\alpha(u, u)}, \quad \|u\|_\beta := \sqrt{\beta(u, u)}.$$

The operator Π is orthogonal with respect to the bilinear form $\alpha(\cdot,\cdot)$:

$$|u|_\alpha^2 = |\Pi u|_\alpha^2 + |u - \Pi u|_\alpha^2.$$

It follows that

$$|\Pi u|_\alpha^2 \leq |u|_\alpha^2, \quad |u - \Pi u|_\alpha^2 \leq |u|_\alpha^2. \tag{6.28}$$

Furthermore, we have the spectral estimate

$$\|u - \Pi u\|_\beta^2 \leq \tfrac{1}{tol}|u - \Pi u|_\alpha^2. \tag{6.29}$$

Proof. ([SDH+14a, lemma 2.11]) Since $\alpha(\cdot,\cdot)$ is symmetric, positive semidefinite and $\beta(\cdot,\cdot)$ is symmetric, positive definite, a β-orthogonal basis of eigenvectors $\{v_k\}_{k=1}^{\dim(X)}$ of X exists (lemma A.3). Since $\beta(\cdot,\cdot)$ is symmetric and positive definite, $\|\cdot\|_\beta$ is a norm. Therefore, we can assume v_k to be normed, such that $\|v_k\|_\beta = 1$. We can thus assume $\beta(v_k, v_l) = \delta_{k,l}$, and we have for $u \in X$

$$\Pi(\Pi u) = \sum_{\lambda_k \leq tol} \sum_{\lambda_l \leq tol} \beta(u, v_l)\beta(v_l, v_k)v_k = \sum_{\lambda_k \leq tol} \beta(u, v_k)v_k = \Pi u;$$

that is, Π is a projection. We now prove the α-orthogonality property of Π. For any $u \in X$, there exist $c_k \in \mathbb{R}$ such that $u = \sum_{k=1}^{\dim(X)} c_k v_k$. We have

$$u = \sum_{k=1}^{\dim(X)} \beta(u, v_k)v_k,$$

since

$$c_k = \sum_{l=1}^{\dim(X)} c_l\beta(v_l, v_k) = \beta(u, v_k), \quad 1 \leq k \leq \dim(X).$$

Using

$$\alpha(v_k, v_l) = \lambda_k\beta(v_k, v_l) = 0, \quad k \neq l,$$

we obtain the α-orthogonality property of Π:

$$|u|_\alpha^2 = \Big|\sum_{\lambda_k \leq tol} \beta(u, v_k)v_k\Big|_\alpha^2 + \Big|\sum_{\lambda_k > tol} \beta(u, v_k)v_k\Big|_\alpha^2$$

$$= |\Pi u|_\alpha^2 + |u - \Pi u|_\alpha^2.$$

Finally, we show the spectral estimate. From $\beta(v_k, v_l) = \delta_{k,l}$ follows that

$$\|u - \Pi u\|_\beta^2 = \Big\| \sum_{\lambda_k > tol} \beta(u, v_k) v_k \Big\|_\beta^2$$

$$= \sum_{\lambda_k > tol} \sum_{\lambda_l > tol} \beta(u, v_k) \beta(u, v_l) \beta(v_k, v_l)$$

$$= \sum_{\lambda_k > tol} \beta(u, v_k)^2.$$

Using

$$|v_k|_\alpha^2 = \lambda_k \|v_k\|_\beta^2 = \lambda_k,$$

we then have

$$\sum_{\lambda_k > tol} \beta(u, v_k)^2 = \sum_{\lambda_k > tol} \beta(u, v_k)^2 \frac{|v_k|_\alpha^2}{\lambda_k} \leq \frac{1}{tol} \sum_{\lambda_k > tol} \beta(u, v_k)^2 |v_k|_\alpha^2.$$

In a similar way to before, we can now use $\alpha(v_k, v_l) = 0$ for $k \neq l$ and obtain

$$\frac{1}{tol} \sum_{\lambda_k > tol} \beta(u, v_k)^2 |v_k|_\alpha^2 = \frac{1}{tol} \sum_{\lambda_k > tol} \sum_{\lambda_l > tol} \beta(u, v_k) \beta(u, v_l) \alpha(v_k, v_l)$$

$$= \frac{1}{tol} \Big| \sum_{\lambda_k > tol} \beta(u, v_k) v_k \Big|_\alpha^2$$

$$= \frac{1}{tol} |u - \Pi u|_\alpha^2.$$

$\square$

For each $\xi \in \mathcal{P}$, let symmetric bilinear forms $\beta_\xi(\cdot, \cdot)$ and $\alpha_\xi(\cdot, \cdot)$ on $X^h(\xi) \times X^h(\xi)$ be given such that $\beta_\xi(\cdot, \cdot)$ is positive definite and $\alpha_\xi(\cdot, \cdot)$ is positive semidefinite. In the case of standard RAGDSW, we use $\beta_\xi := \beta_\xi^K$ and $\alpha_\xi := \alpha_\xi^K$. For the S-variant, we replace α_ξ with $\alpha_\xi := \alpha_\xi^S$. For the mass variant, we replace β_ξ with $\beta_\xi := \beta_\xi^M$. Similarly, the slab variant uses either $\alpha_\xi := \alpha_\xi^{K,l}$ or $\alpha_\xi := \alpha_\xi^{S,l}$. We define the corresponding norm

$$\|v\|_{\beta_\xi} := \sqrt{\beta_\xi(v,v)} \quad \forall v \in X^h(\Omega^h),$$

and seminorm

$$|v|_{\alpha_\xi} := \sqrt{\alpha_\xi(v,v)} \quad \forall v \in X^h(\Omega^h),$$

and extend their domains to $V^h(\Omega)$ by restricting $v \in V^h(\Omega)$ to Ω^h. Furthermore, let

$$|v|_{a(B)} := \sqrt{a_B(v,v)} \quad \forall v \in V^h(\Omega), \tag{6.30}$$

for any union $B \subset \overline{\Omega}$ of finite elements $T \in \tau_h(\Omega)$.

As a consequence of the energy-minimality property of the operator $\mathcal{H}_{\xi \to \Omega_\xi}(\cdot)$, we have

$$\alpha_\xi^K(v,v) = |\mathcal{H}_{\xi \to \Omega_\xi}(v)|_{a(\Omega_\xi)}^2 \leq |v|_{a(\Omega_\xi)}^2 \quad \forall v \in V^h(\Omega).$$

By (6.16), we have $\alpha_\xi^{K,l}(v,v) \leq \alpha_\xi^K(v,v)$; by (6.19), we have $\alpha_\xi^S(v,v) \leq \alpha_\xi^K(v,v)$; and by (6.21), we have $\alpha_\xi^{S,l}(v,v) \leq \alpha_\xi^K(v,v)$. As result, we obtain

$$\alpha_\xi(v,v) \leq |v|_{a(\Omega_\xi)}^2 \quad \forall v \in V^h(\Omega). \tag{6.31}$$

We define the projection

$$\Pi_\xi \colon V^h(\Omega) \to \mathrm{span}\left(\{\, v_{k,\xi} : \lambda_{k,\xi} \leq tol_\xi \,\}\right) \subset V_{0,\partial\Omega_D}^h(\Omega),$$

$$w \mapsto \sum_{\lambda_{k,\xi} \leq tol_\xi} \beta_\xi(\, w \,,\, v_{k,\xi}\,)v_{k,\xi}, \tag{6.32}$$

where $v_{k,\xi}$ are the coarse functions and $\lambda_{k,\xi}$ the corresponding eigenvalues from (6.12). The next lemma (cf. [HKKR19, lemma 5.4] and [HKK+22, lemma 10.2]) follows directly from lemma 6.1 and can be regarded as a Poincaré-type inequality in case $\beta_\xi(\cdot,\cdot)$ is given by a scaled L^2-inner product.

Lemma 6.2. *([HKK+22, lemma 10.2]) For $\xi \in \mathcal{P}$ and $u \in V^h(\Omega)$, it holds that*

$$\|u - \Pi_\xi u\|_{\beta_\xi}^2 \leq \frac{1}{tol_\xi} \sum_{k \in n(\xi)} |u|_{a(\Omega_k)}^2.$$

Proof. We have

$$\|u - \Pi_\xi u\|_{\beta_\xi}^2 \overset{(6.29)}{\leq} \frac{1}{tol_\xi}|u - \Pi_\xi u|_{\alpha_\xi}^2 \overset{(6.28)}{\leq} \frac{1}{tol_\xi}|u|_{\alpha_\xi}^2 \overset{(6.31)}{\leq} \frac{1}{tol_\xi}|u|_{a(\Omega_\xi)}^2;$$

the proof is completed by noting that $|u|_{a(\Omega_\xi)}^2 = \sum_{k \in n(\xi)} |u|_{a(\Omega_k)}^2$. We remark that the exploitation of (6.29) and (6.31) are two fundamental steps toward the proof of a condition number bound. $\qquad\square$

6.3. Condition Number Bound

In the introduction of this chapter, we found that the constant C_0 has to be determined to fulfill assumption 6.1 of a stable decomposition. We then obtain the condition number bound

$$\kappa(M_{\text{OSL2}}^{-1}K) \leq C_0^2(\hat{N}_C + 1).$$

In the following, we derive the bound

$$C_0^2 \leq 4 + 5\left(1 + \sqrt{C_{\text{inv}}C_\tau \frac{N^\xi}{tol_\mathcal{P}}}\right)^2 + C_{\text{inv}}C_\tau \frac{\mathcal{C}}{tol_\mathcal{P}},$$

where

$$tol_\mathcal{P} := \min_{\xi \in \mathcal{P}} tol_\xi$$

is the smallest tolerance for the selection of eigenfunctions, and C_τ is the maximum number of vertices of any element $T \in \tau_h(\Omega)$;

$$C_\tau := \begin{cases} 3 & \text{if } T \in \tau_h(\Omega) \text{ are triangles,} \\ 4 & \text{if } T \in \tau_h(\Omega) \text{ are rectangles,} \\ 4 & \text{if } T \in \tau_h(\Omega) \text{ are tetrahedra,} \\ 8 & \text{if } T \in \tau_h(\Omega) \text{ are cuboids.} \end{cases} \tag{6.33}$$

Furthermore, N^ξ is the maximum number of interface components $\xi \in \mathcal{P}$ of any subdomain,

$$N^\xi := \max_{1 \leq i \leq N} |\mathcal{P}(\Omega_i)|, \quad \mathcal{P}(\Omega_i) := \{\, \xi \in \mathcal{P} : \xi \cap \overline{\Omega}_i \neq \emptyset \,\}, \tag{6.34}$$

and $\mathcal{C}$ is a measure for the $\mathcal{P}$-connectivity of the domain decomposition: two subdomains i, j are connected if they touch the same interface component $\xi \in \mathcal{P}$, from which follows that $i, j \in n(\xi)$;

$$\mathcal{C} := \mathcal{C}\left(\{\Omega_i\}_{i=1}^N, \mathcal{P}\right) := \max_{1 \leq i \leq N} \sum_{j=1}^N |\{\, \xi \in \mathcal{P} : i, j \in n(\xi) \,\}| = \max_{1 \leq i \leq N} \sum_{\xi \in \mathcal{P}(\Omega_i)} |n(\xi)|. \tag{6.35}$$

6. Theory of Adaptive GDSW-Type Coarse Spaces

To obtain better bounds (see below) for the standard adaptive GDSW coarse space, we define variants of N^ξ and $\mathcal{C}$ that incorporate tol_ξ:

$$N^{\xi,tol_\xi} := \max_{1 \le i \le N} \sum_{\xi \in \mathcal{P}(\Omega_i)} \frac{1}{tol_\xi}, \qquad (6.36)$$

$$\mathcal{C}^{tol_\xi} := \max_{1 \le i \le N} \sum_{\xi \in \mathcal{P}(\Omega_i)} \frac{|n(\xi)|}{tol_\xi}. \qquad (6.37)$$

The inequalities

$$N^{\xi,tol_\xi} \le \frac{N^\xi}{tol_\mathcal{P}},$$

$$\mathcal{C}^{tol_\xi} \le \frac{\mathcal{C}}{tol_\mathcal{P}},$$

hold and become equalities if $tol_\xi = tol_\mathcal{P}$ is satisfied for all $\xi \in \mathcal{P}$.

If we use the standard adaptive GDSW coarse space, the coarse functions associated with coarse nodes are obtained by using $tol_\mathcal{V} = \infty$ for the selection of eigenfunctions—all vertex eigenfunctions are then selected. It is not necessary to set up and solve the respective generalized eigenvalue problems as the solutions to the vertex eigenproblems are known a priori (restrictions of the null space to $\nu \in \mathcal{V}$). Since $tol_\mathcal{V} = \infty$, the constants N^{ξ,tol_ξ} and $\mathcal{C}^{tol_\xi}$ can be significantly smaller than $N^\xi/tol_\mathcal{P}$ and $\mathcal{C}/tol_\mathcal{P}$; cf. section 6.4.1.

We define the projection

$$\Pi_\mathcal{P} \colon V^h(\Omega) \to V_\mathcal{P} \subset V^h_{0,\partial\Omega_D}(\Omega),$$

$$w \mapsto \sum_{\xi \in \mathcal{P}} \Pi_\xi w,$$

onto the coarse space $V_0 := V_\mathcal{P}$ that is obtained by using the respective RAGDSW method; see (6.13) for the definition of $V_\mathcal{P}$ and (6.32) for that of Π_ξ. Let $u \in V^h_{0,\partial\Omega_D}(\Omega)$; we define its coarse component u_0 that is required for the stable decomposition in assumption 6.1 as

$$u_0 := \Pi_\mathcal{P} u.$$

The following proof of a condition number bound uses the (standard) right-hand side

$$\beta^K_\xi(\cdot, \cdot) = \sum_{i=1}^{n_\xi} a_{\Omega_{\xi_i}} \left(z_{\xi_i}(\cdot), z_{\xi_i}(\cdot) \right)$$

of the generalized eigenvalue problem for many parts of the proof. To integrate the mass variant, we will make use of the bound

$$\beta_\xi^K(u,u) \le C_{\mathrm{inv}} b_\xi^M(u,u)$$

from (6.26). We can use the same bound if the lumped mass matrix is used; cf. remark 6.6. If the variant with a lumped stiffness matrix is used, nothing else needs to be taken into account; cf. remark 6.3. In the following, we use

$$\beta_\xi^K(u,u) \le C_{\mathrm{inv}} b_\xi(u,u) \quad \forall u \in V^h(\Omega), \tag{6.38}$$

where $C_{\mathrm{inv}} = 1$ holds for all variants other than the mass variant.

The following lemma extends the result of lemma 6.2 from the local projection $\Pi_\xi u$ to $\Pi_\mathcal{P} u$. For the construction of coarse functions, we have—in a first step—extended eigenfunctions by zero from ξ to the interface. In chapter 7, we will introduce methods that use other types of extensions. However, the extension-by-zero simplifies the following proof considerably.

Lemma 6.3. *(Compare with [HKKR19, lemma 6.2; HKK+22, lemma 11.1].) We have*

$$\|u - u_0\|_{\beta_\xi^K}^2 \le \frac{C_{\mathrm{inv}}}{tol_\xi} \sum_{k \in n(\xi)} |u|_{a(\Omega_k)}^2,$$

for $\xi \in \mathcal{P}$ and $u \in V^h(\Omega)$.

Proof. We exploit the fact that all coarse functions associated with interface components other than ξ are zero on ξ; we obtain

$$\|u - u_0\|_{\beta_\xi^K}^2 = \sum_{i=1}^{n_\xi} |z_{\xi_i}(u - \Pi_\mathcal{P} u)|_{a(\Omega_{\xi_i})}^2 = \sum_{i=1}^{n_\xi} |z_{\xi_i}(u - \Pi_\xi u)|_{a(\Omega_{\xi_i})}^2 = \|u - \Pi_\xi u\|_{\beta_\xi^K}^2.$$

Using (6.38) and lemma 6.2, it follows that

$$\|u - \Pi_\xi u\|_{\beta_\xi^K}^2 \le C_{\mathrm{inv}} \|u - \Pi_\xi u\|_{\beta_\xi}^2 \le \frac{C_{\mathrm{inv}}}{tol_\xi} \sum_{k \in n(\xi)} |u|_{a(\Omega_k)}^2.$$

$\square$

To construct a stable decomposition (cf. assumption 6.1), we need to find bounds for the energy of local components u_i and the coarse component u_0. The following lemma gives a bound for the coarse component, and the subsequent lemma gives a bound that can later be used to derive a bound for the local components.

Lemma 6.4. *(Compare with [HKKR19, lemma 6.3; HKK+22, lemma 11.2].) It holds that*

$$|u_0|_{a(\Omega)} \le |u|_{a(\Omega)} + \sqrt{C_\tau \sum_{\xi \in \mathcal{P}} \|u - u_0\|_{\beta_\xi^K}^2}.$$

Proof. We will first exploit that coarse functions are energy minimizing on each subdomain Ω_i and, subsequently, we will make use of the spectral properties of eigenfunctions.

As u_0 is energy minimizing on each subdomain Ω_i with respect to $|\cdot|_{a(\Omega_i)}$, we have $u_0 = \mathcal{H}_{\Gamma^h \to \Omega}(u_0)$ (cf. [GLR15, lemma 4.1]), and the energy can be bounded from above by using u itself or an extension-by-zero:

$$|u_0|_{a(\Omega)} \le |\mathcal{H}_{\Gamma^h \to \Omega}(u)|_{a(\Omega)} + |\mathcal{H}_{\Gamma^h \to \Omega}(u - u_0)|_{a(\Omega)}$$

$$\le |u|_{a(\Omega)} + |z_{\Gamma^h}(u - u_0)|_{a(\Omega)}.$$

Next, we split the extension-by-zero with respect to the subcomponents of all $\xi \in \mathcal{P}$.

$$|z_{\Gamma^h}(u - u_0)|_{a(\Omega)}^2 = \Big| \sum_{\xi_i \in \mathcal{N}_{ec,\mathcal{P}}} z_{\xi_i}(u - u_0) \Big|_{a(\Omega)}^2$$

$$= \sum_{T \in \tau_h(\Omega)} \Big| \sum_{\xi_i \in \mathcal{N}_{ec,\mathcal{P}}} z_{\xi_i}(u - u_0) \Big|_{a(T)}^2.$$

Here, $\mathcal{N}_{ec,\mathcal{P}}$ is the set of all subcomponents of $\mathcal{P}$ (see (5.4)); we have $\xi_i \cap \xi_j = \emptyset$ for $i \ne j$ and $\bigcup_{\xi_i \in \mathcal{N}_{ec,\mathcal{P}}} \xi_i = \Gamma^h$. In each finite element T, there can be at most C_τ NECs ξ_i and, thus, at most C_τ different functions z_{ξ_i} that are nonzero in T. Hence, using the Cauchy–Schwarz inequality, we have

$$\sum_{T \in \tau_h(\Omega)} \Big| \sum_{\xi_i \in \mathcal{N}_{ec,\mathcal{P}}} z_{\xi_i}(u - u_0) \Big|_{a(T)}^2 \le \sum_{T \in \tau_h(\Omega)} C_\tau \sum_{\xi_i \in \mathcal{N}_{ec,\mathcal{P}}} |z_{\xi_i}(u - u_0)|_{a(T)}^2 \tag{6.39}$$

$$= C_\tau \sum_{\xi_i \in \mathcal{N}_{ec,\mathcal{P}}} |z_{\xi_i}(u - u_0)|_{a(\Omega_{\xi_i})}^2$$

$$= C_\tau \sum_{\xi \in \mathcal{P}} \|u - u_0\|_{\beta_\xi^K}^2.$$

Thus,

$$|u_0|_{a(\Omega)} \leq |u|_{a(\Omega)} + \sqrt{C_\tau \sum_{\xi \in \mathcal{P}} \|u - u_0\|_{\beta_\xi^K}^2}.$$

$\square$

Corollary 6.1. *(Compare with [HKKR19, lemma 6.3; HKK+22, lemma 11.2].) It holds that*

$$|u_0|_{a(\Omega)} \leq \left(1 + \sqrt{C_{\mathrm{inv}} C_\tau N^{\xi, tol_\xi}}\right) |u|_{a(\Omega)}.$$

Proof. By lemmas 6.4 and 6.3, we have

$$|u_0|_{a(\Omega)} \leq |u|_{a(\Omega)} + \sqrt{C_\tau \sum_{\xi \in \mathcal{P}} \|u - u_0\|_{\beta_\xi^K}^2}$$

$$\leq |u|_{a(\Omega)} + \sqrt{C_{\mathrm{inv}} C_\tau \sum_{\xi \in \mathcal{P}} \frac{1}{tol_\xi} \sum_{k \in n(\xi)} |u|_{a(\Omega_k)}^2}.$$

By definition of N^{ξ, tol_ξ} in (6.36), we have

$$\sum_{\xi \in \mathcal{P}} \frac{1}{tol_\xi} \sum_{k \in n(\xi)} |u|_{a(\Omega_k)}^2 \leq N^{\xi, tol_\xi} |u|_{a(\Omega)}^2.$$

Thus,

$$|u_0|_{a(\Omega)} \leq \left(1 + \sqrt{C_{\mathrm{inv}} C_\tau N^{\xi, tol_\xi}}\right) |u|_{a(\Omega)}.$$

$\square$

In lemma 6.6, we will construct a stable decomposition. Therein, an energy bound for the product of $u - u_0$ with a partition of unity function θ_i is derived, where θ_i is associated with Ω_i. The partition of unity is—unlike in the classical theory (cf., e.g., [TW05, lemma 3.4])—not defined with respect to the set of overlapping subdomains $\{\Omega_i'\}_{i=1}^N$ that is used for the first level of the Schwarz preconditioner. For technical reasons, we restrict the support of a partition of unity function θ_i to an overlapping subdomain $\tilde{\Omega}_i \subseteq \Omega_i'$ that is obtained by extending Ω_i by one layer of finite elements.

Let us comment on the technical reasons for the restriction of θ_i to $\tilde{\Omega}_i$ instead of Ω_i'. We will be able to represent θ_i on the overlap $\tilde{\Omega}_i \setminus \Omega_i$, using the extension-by-zero $z_{\Gamma^h}(\theta_i)$. We require the extension-by-zero to make use of the generalized eigenvalue problems

with lemma 6.2. Using the extension-by-zero to represent θ_i would not be possible if the partition of unity were defined on a larger overlap. A different type of eigenvalue problem may be required to obtain a condition number bound that depends on the size of the overlap.

As a consequence, the condition number bound in theorem 6.1 does not reflect that the convergence can benefit from a larger overlap. However, as we have seen in table 3.5, the numerical results suggest that the influence of the overlap is only minor for the considered heterogeneous problems. Let us emphasize that all methods in this work are not restricted to the use of a minimal overlap for the first level of the preconditioner.

The following lemma provides estimates for the energy of the product of a cutoff function—such as θ_i on $\tilde{\Omega}_i \setminus \Omega_i$—and $u - u_0$. The lemma covers two cases: an estimate that is carried out locally on $\tilde{\Omega}_i \setminus \Omega_i$ and one that is carried out globally on Ω.

Lemma 6.5 (Partition of Unity Estimate). *(Compare with [HKK+22, lemma 11.3; HKKR19, lemma 6.4].) Let $l \in \{0, 1, \ldots, N\}$ and*

$$
B := \begin{cases} \tilde{\Omega}_l \setminus \Omega_l & \text{if } l > 0, \\ \Omega & \text{if } l = 0. \end{cases}
$$

Furthermore, we set $\Omega_0 := \Omega$. Let $\Psi \colon \overline{B} \to \mathbb{R}$ be a scalar-valued finite element function such that $\Psi|_{\xi_i}$ is constant on $\xi_i \in \mathcal{N}_{ec,\mathcal{P}}$, $\xi_i \subset \overline{B}$; that is, there exists a constant C_i such that $\Psi(x^h) = C_i$ for all $x^h \in \xi_i$. Moreover, we assume that $0 \leq \Psi \leq 1$ and $\Psi(x^h) = 0$ for $x^h \notin \Gamma^h \cap \overline{\Omega}_l$. Then

$$
\left| I^h(\Psi \cdot (u - u_0)) \right|^2_{a(B)} \leq C_\tau \sum_{\xi \in \mathcal{P}(\Omega_l)} \beta^K_\xi (u - u_0, u - u_0),
$$

where $I^h(\cdot)$ is the pointwise interpolation operator of the finite element space $V^h(\Omega)$.

Proof. We define the set $\mathcal{N}_{ec,\mathcal{P}}(\Omega_l) := \{ \xi_j \in \mathcal{N}_{ec,\mathcal{P}} : \xi_j \subset \overline{\Omega}_l \}$ of subcomponents (NECs of $\xi \in \mathcal{P}$) that are part of $\overline{\Omega}_l$. By definition, we have $\mathcal{P}(\Omega_0) = \mathcal{P}$ and $\mathcal{N}_{ec,\mathcal{P}}(\Omega_0) = \mathcal{N}_{ec,\mathcal{P}}$. Since $\Psi(x^h)$ is only nonzero on the part of the interface that coincides with $\overline{\Omega}_l$, and since $z_{\xi_i}(\cdot)$ acts as an identity operator on ξ_i, it follows that

$$
\left| I^h(\Psi \cdot (u - u_0)) \right|^2_{a(B)} = \left| \sum_{\xi_i \in \mathcal{N}_{ec,\mathcal{P}}(\Omega_l)} z_{\xi_i}(\Psi \cdot (u - u_0)) \right|^2_{a(B)}.
$$

In any finite element $T \in \tau_h(\Omega)$, there can be at most as many NECs ξ_i on which the corresponding extension-by-zero $z_{\xi_i}(\cdot)$ is nonzero as T has vertices—thus, at most C_τ. Consequently, using the Cauchy–Schwarz inequality, we have

$$
\Big| \sum_{\xi_i \in \mathcal{N}_{ec,\mathcal{P}}(\Omega_l)} z_{\xi_i}\big(\Psi \cdot (u - u_0)\big) \Big|^2_{a(B)} = \sum_{T \in \tau_h(B)} \Big| \sum_{\xi_i \in \mathcal{N}_{ec,\mathcal{P}}(\Omega_l)} z_{\xi_i}\big(\Psi \cdot (u - u_0)\big) \Big|^2_{a(T)}
$$

$$
\leq C_\tau \sum_{T \in \tau_h(B)} \sum_{\xi_i \in \mathcal{N}_{ec,\mathcal{P}}(\Omega_l)} \big| z_{\xi_i}\big(\Psi \cdot (u - u_0)\big) \big|^2_{a(T)}
$$

$$
\leq C_\tau \sum_{\xi_i \in \mathcal{N}_{ec,\mathcal{P}}(\Omega_l)} \big| z_{\xi_i}\big(\Psi \cdot (u - u_0)\big) \big|^2_{a(\Omega_{\xi_i})}.
$$

As Ψ is constant on each $\xi_i \in \mathcal{N}_{ec,\mathcal{P}}(\Omega_l)$, and since $0 \leq \Psi \leq 1$, we have

$$
\sum_{\xi_i \in \mathcal{N}_{ec,\mathcal{P}}(\Omega_l)} \big| z_{\xi_i}\big(\Psi \cdot (u - u_0)\big) \big|^2_{a(\Omega_{\xi_i})} = \sum_{\xi_i \in \mathcal{N}_{ec,\mathcal{P}}(\Omega_l)} \big(\Psi|_{\xi_i}\big)^2 |z_{\xi_i}(u - u_0)|^2_{a(\Omega_{\xi_i})} \tag{6.40}
$$

$$
\leq \sum_{\xi_i \in \mathcal{N}_{ec,\mathcal{P}}(\Omega_l)} |z_{\xi_i}(u - u_0)|^2_{a(\Omega_{\xi_i})} \tag{6.41}
$$

$$
\leq \sum_{\xi \in \mathcal{P}(\Omega_l)} \sum_{i=1}^{n_\xi} |z_{\xi_i}(u - u_0)|^2_{a(\Omega_{\xi_i})}
$$

$$
= \sum_{\xi \in \mathcal{P}(\Omega_l)} \beta_\xi^K\big(u - u_0, u - u_0\big).
$$

We remark that in (6.40), the decoupling (with respect to the subcomponents $\xi_i \in \mathcal{N}_{ec,\mathcal{P}}$) of the right-hand side of the RAGDSW generalized eigenvalue problem was exploited. Without the decoupling, the equality

$$
\big| z_{\xi_i}\big(\Psi \cdot (u - u_0)\big) \big|_{a(\Omega_{\xi_i})} = \Psi|_{\xi_i} \cdot |z_{\xi_i}(u - u_0)|_{a(\Omega_{\xi_i})}
$$

would not hold in general. Notably, there does not exist an upper bound for the left-hand side that is independent of the coefficient contrast; cf. section 6.1.1. □

Corollary 6.2. *(Compare with [HKK+22, lemma 11.3; HKKR19, lemma 6.4].) Let the assumptions of lemma 6.5 be satisfied. Then*

$$
|I^h(\Psi \cdot (u - u_0))|^2_{a(B)} \leq C_{\mathrm{inv}} C_\tau \sum_{\xi \in \mathcal{P}(\Omega_l)} \frac{1}{tol_\xi} \sum_{k \in n(\xi)} |u|^2_{a(\Omega_k)}.
$$

6. Theory of Adaptive GDSW-Type Coarse Spaces

If $l = 0$, such that $B = \Omega$, we obtain

$$|I^h(\Psi \cdot (u - u_0))|^2_{a(\Omega)} \leq C_{\text{inv}} C_\tau N^{\xi, tol_\xi} |u|^2_{a(\Omega)}.$$

If the assumptions of lemma 6.5 are satisfied for all $1 \leq l \leq N$, $\Psi_l \colon \overline{B}_l \to \mathbb{R}$, with $B_l \coloneqq \tilde{\Omega}_l \setminus \Omega_l$, we obtain

$$\sum_{l=1}^{N} |I^h(\Psi_l \cdot (u - u_0))|^2_{a(\tilde{\Omega}_l \setminus \Omega_l)} \leq C_{\text{inv}} C_\tau \mathcal{C}^{tol_\xi} |u|^2_{a(\Omega)},$$

where $\mathcal{C}^{tol_\xi}$ is a tolerance-weighted measure for the $\mathcal{P}$-connectivity of the domain decomposition; see (6.37) for its definition.

Proof. Using lemmas 6.5 and 6.3, we have

$$|I^h(\Psi \cdot (u - u_0))|^2_{a(B)} \leq C_\tau \sum_{\xi \in \mathcal{P}(\Omega_l)} \beta^K_\xi (\, u - u_0 \,,\, u - u_0 \,)$$

$$\leq C_{\text{inv}} C_\tau \sum_{\xi \in \mathcal{P}(\Omega_l)} \frac{1}{tol_\xi} \sum_{k \in n(\xi)} |u|^2_{a(\Omega_k)}.$$

For $B = \Omega$, it follows that

$$|I^h(\Psi \cdot (u - u_0))|^2_{a(\Omega)} \leq C_{\text{inv}} C_\tau \sum_{\xi \in \mathcal{P}} \frac{1}{tol_\xi} \sum_{k \in n(\xi)} |u|^2_{a(\Omega_k)} \leq C_{\text{inv}} C_\tau N^{\xi, tol_\xi} |u|^2_{a(\Omega)}.$$

For $B_l = \tilde{\Omega}_l \setminus \Omega_l$ and Ψ_l, we obtain

$$\sum_{l=1}^{N} |I^h(\Psi_l \cdot (u - u_0))|^2_{a(\tilde{\Omega}_l \setminus \Omega_l)} \leq C_{\text{inv}} C_\tau \sum_{l=1}^{N} \sum_{\xi \in \mathcal{P}(\Omega_l)} \frac{1}{tol_\xi} \sum_{k \in n(\xi)} |u|^2_{a(\Omega_k)}$$

$$\leq C_{\text{inv}} C_\tau \mathcal{C}^{tol_\xi} |u|^2_{a(\Omega)}.$$

$\square$

We now prove the existence of a stable decomposition.

Lemma 6.6 (Stable Decomposition)**.** *(Compare with [HKK+22, lemma 11.4; HKKR19, theorem 6.5; HKKR18b, lemma 6.4].) For each $u \in V^h_{0, \partial \Omega_D}(\Omega)$, there exists a decomposition $u = \sum_{i=0}^{N} R_i^T u_i$, $u_i \in V_i$, $0 \leq i \leq N$, such that*

$$\sum_{i=0}^{N} |u_i|^2_{a(\Omega'_i)} \leq C_0^2 |u|^2_{a(\Omega)},$$

where

$$C_0^2 = 4 + 5\left(1 + \sqrt{C_{\mathrm{inv}}C_\tau N^{\xi, tol_\xi}}\right)^2 + C_{\mathrm{inv}}C_\tau \mathcal{C}^{tol_\xi}$$

$$\leq 4 + 5\left(1 + \sqrt{C_{\mathrm{inv}}C_\tau \frac{N^\xi}{tol_\mathcal{P}}}\right)^2 + C_{\mathrm{inv}}C_\tau \frac{\mathcal{C}}{tol_\mathcal{P}},$$

where $\mathcal{C}$ is a measure for the $\mathcal{P}$-connectivity of the domain decomposition; see (6.35) for its definition.

Proof. As introduced above, $\{\tilde{\Omega}_i\}_{i=1}^N$ denotes an overlapping decomposition, where $\tilde{\Omega}_i$ is obtained by extending Ω_i by one layer of finite elements. Let $u \in V_{0,\partial\Omega_D}^h(\Omega)$. Then we define the local components $u_i \in V_i$, $1 \leq i \leq N$, as

$$u_i := I^h(\theta_i \cdot (u - u_0))\big|_{\Omega_i'},$$

where $\{\theta_i\}_{i=1}^N$ is a partition of unity of scalar-valued finite element functions. The functions θ_i are defined by their nodal values

$$\theta_i(x^h) := \begin{cases} \frac{1}{|n(x^h)|} & \text{if } x^h \in \overline{\Omega}_i, \\ 0 & \text{if } x^h \in \overline{\Omega} \setminus \overline{\Omega}_i, \end{cases}$$

where x^h is a finite element node, and $|n(x^h)|$ is the number of subdomains for which $x^h \in \overline{\Omega}_i$ holds. Note that we have $u_i \in V_i$, since $\tilde{\Omega}_i \subset \Omega_i'$. We obtain

$$|u_i|_{a(\Omega_i')}^2 = |u_i|_{a(\tilde{\Omega}_i)}^2 = |I^h(\theta_i(u - u_0))|_{a(\tilde{\Omega}_i)}^2$$

$$= |I^h(\theta_i(u - u_0))|_{a(\tilde{\Omega}_i \setminus \Omega_i)}^2 + |I^h(\theta_i(u - u_0))|_{a(\Omega_i)}^2. \qquad (6.42)$$

As θ_i satisfies the assumptions of corollary 6.2, we obtain for the first term of (6.42)

$$\sum_{i=1}^N |I^h(\theta_i(u - u_0))|_{a(\tilde{\Omega}_i \setminus \Omega_i)}^2 \leq C_{\mathrm{inv}}C_\tau \mathcal{C}^{tol_\xi}|u|_{a(\Omega)}^2. \qquad (6.43)$$

Using the Cauchy–Schwarz inequality, it follows for the second term of (6.42) that

$$|I^h(\theta_i(u - u_0))|_{a(\Omega_i)}^2 \leq 2|I^h((1 - \theta_i)(u - u_0))|_{a(\Omega_i)}^2 + 2|u - u_0|_{a(\Omega_i)}^2. \qquad (6.44)$$

By summing over all subdomains, we obtain a bound for the last term of (6.44), using corollary 6.1:

$$\sum_{i=1}^{N} 2|u - u_0|^2_{a(\Omega_i)} = 2|u - u_0|^2_{a(\Omega)}$$
$$\leq 2 \left(|u|_{a(\Omega)} + |u_0|_{a(\Omega)} \right)^2$$
$$\leq 2 \left(2 + \sqrt{C_{\text{inv}} C_\tau N^{\xi, tol_\xi}} \right)^2 |u|^2_{a(\Omega)}. \qquad (6.45)$$

To bound the first term on the right-hand side of (6.44), we define a cutoff function $\theta \colon \overline{\Omega} \to [0, 1]$ with respect to the interface Γ^h. The function θ is a scalar-valued finite element function and defined by

$$\theta(x^h) := 1 - \frac{1}{|n(x^h)|}, \quad x^h \in \overline{\Omega}.$$

As θ satisfies the assumptions of corollary 6.2, it follows that

$$\sum_{i=1}^{N} 2|I^h(\theta(u - u_0))|^2_{a(\Omega_i)} = 2|I^h(\theta(u - u_0))|^2_{a(\Omega)} \leq 2C_{\text{inv}} C_\tau N^{\xi, tol_\xi} |u|^2_{a(\Omega)}. \qquad (6.46)$$

Let $D := C_{\text{inv}} C_\tau N^{\xi, tol_\xi}$. Using corollary 6.1, (6.43), (6.45), and (6.46), we then obtain

$$\sum_{i=0}^{N} |u_i|^2_{a(\Omega'_i)} = |u_0|^2_{a(\Omega)} + \sum_{i=1}^{N} |u_i|^2_{a(\tilde{\Omega}_i)}$$
$$\leq \left((1 + \sqrt{D})^2 + C_{\text{inv}} C_\tau \mathcal{C}^{tol_\xi} + 2 \left(2 + \sqrt{D} \right)^2 + 2D \right) |u|^2_{a(\Omega)}$$
$$= C_0^2 |u|^2_{a(\Omega)},$$

where

$$C_0^2 = 4 + 5(1 + \sqrt{D})^2 + C_{\text{inv}} C_\tau \mathcal{C}^{tol_\xi}.$$

By using $N^{\xi, tol_\xi} \leq N^\xi / tol_\mathcal{P}$ and $\mathcal{C}^{tol_\xi} \leq \mathcal{C} / tol_\mathcal{P}$, it follows that

$$C_0^2 \leq 4 + 5 \left(1 + \sqrt{C_{\text{inv}} C_\tau \frac{N^\xi}{tol_\mathcal{P}}} \right)^2 + C_{\text{inv}} C_\tau \frac{\mathcal{C}}{tol_\mathcal{P}}.$$

$\square$

With a slight change in the proof of lemmas 6.5 and 6.6, we can improve the constant $\mathcal{C}$ for some interface partitions; see (6.47) in section 6.4.3. However, for the interface partitions constructed in chapter 5, the new constant will be identical to $\mathcal{C}$.

From (6.3) and lemma 6.6, we obtain a bound for the condition number of $M_{\text{RAGDSW}}^{-1}K$. Let us note that slightly different constants are given in [HKK+22, theorem 11.5].

Theorem 6.1. *The condition number of the RAGDSW two-level Schwarz operator is bounded by*

$$\kappa\left(M_{\text{RAGDSW}}^{-1}K\right) \leq \left(4 + 5\left(1 + \sqrt{C_{\text{inv}}C_\tau N^{\xi,tol_\xi}}\right)^2 + C_{\text{inv}}C_\tau \mathcal{C}^{tol_\xi}\right)\left(\hat{N}_c + 1\right)$$

$$\leq \left(4 + 5\left(1 + \sqrt{C_{\text{inv}}C_\tau \frac{N^\xi}{tol_\mathcal{P}}}\right)^2 + C_{\text{inv}}C_\tau \frac{\mathcal{C}}{tol_\mathcal{P}}\right)\left(\hat{N}_c + 1\right),$$

where $\hat{N}_c$ is the maximum number of overlapping subdomains $\{\Omega_i'\}_{i=1}^N$ any finite element node $x^h \in \overline{\Omega}$ can belong to. All constants are independent of H, h, and the contrast of the coefficient function E.

6.4. Practical Aspects of the Condition Number Bound

In the following, for the model problems in chapter 2 and the coarse spaces AGDSW, R–WB–AGDSW, and RAGDSW, we state the constants encountered in the condition number bound in theorem 6.1. Furthermore, we investigate how the eigenvalues of the local generalized eigenvalue problems depend on the fine mesh resolution h. Finally, we show the effect of interface components that span large sections of Ω on the condition number.

6.4.1. Constants of the Condition Number Bound

The constants relevant for the condition number bound in theorem 6.1 for the problems of sections 2.1 to 2.4 are given in table 6.1. The bounds are based on the tolerance for the selection of eigenfunctions $tol_\mathcal{P} = 0.05$. As we use the standard variants of the

	C_τ	$\hat{N}_C$	AGDSW			R–WB–AGDSW			RAGDSW		
			N^{ξ,tol_ξ}	$\mathcal{C}^{tol_\xi}$	κ_{bound}	N^ξ	$\mathcal{C}$	κ_{bound}	N^ξ	$\mathcal{C}$	κ_{bound}
(1)	4	8	1 200	3 340	$3.4 \cdot 10^5$	51	170	$3.1 \cdot 10^5$	34	136	$2.3 \cdot 10^5$
(2)	4	8	620	1 660	$1.8 \cdot 10^5$	22	68	$1.3 \cdot 10^5$	12	48	$8.1 \cdot 10^4$
(3)	4	8	1 340	3 660	$3.8 \cdot 10^5$	53	176	$3.2 \cdot 10^5$	34	138	$2.3 \cdot 10^5$
(4)	4	10	1 440	4 000	$5.0 \cdot 10^5$	55	183	$4.1 \cdot 10^5$	36	145	$2.9 \cdot 10^5$

Table 6.1.: Constants and condition number bounds κ_{bound} from theorem 6.1 for AGDSW, R–WB–AGDSW, RAGDSW, and problems **(1)**–**(4)** from sections 2.1 to 2.4; $\kappa_{\text{bound}} := \left(4 + 5\left(1 + \sqrt{C_\tau N^{\xi,tol_\xi}}\right)^2 + C_\tau \mathcal{C}^{tol_\xi}\right)\left(\hat{N}_c + 1\right)$. For R–WB–AGDSW and RAGDSW, we have $N^{\xi,tol_\xi} = {}^{N^\xi}\!/_{tol_\mathcal{P}}$ and $\mathcal{C}^{tol_\xi} = {}^{\mathcal{C}}\!/_{tol_\mathcal{P}}$; $tol_\mathcal{P} := 0.05$.

coarse spaces, we have $C_{\text{inv}} = 1$. The constants N^{ξ,tol_ξ} and $\mathcal{C}^{tol_\xi}$ are identical to ${}^{N^\xi}\!/_{tol_\mathcal{P}}$ and ${}^{\mathcal{C}}\!/_{tol_\mathcal{P}}$, respectively, in the case of R–WB–AGDSW and RAGDSW; only AGDSW benefits from using N^{ξ,tol_ξ} and $\mathcal{C}^{tol_\xi}$.

From table 6.1, we learn that N^{ξ,tol_ξ} and $\mathcal{C}^{tol_\xi}$ are of considerable magnitude, which results in large condition number bounds. Although, theoretically, it is unknown whether in certain situations we may indeed encounter large condition numbers, the numerical results of this work suggest otherwise—that the magnitude of the bounds are an artifact of the proof. Let us note that this is similar for, e.g., adaptive FETI-DP; see, e.g., [Küh18, theorem 5.7] and the follow-up discussion.

Let us recapitulate the definition of the constants and focus on the bound for the reduced variants (using $tol_\xi = tol_\mathcal{P}$):

$$\kappa(M^{-1}K) \leq \left(4 + 5\left(1 + \sqrt{C_\tau \frac{N^\xi}{tol_\mathcal{P}}}\right)^2 + C_\tau \frac{\mathcal{C}}{tol_\mathcal{P}}\right)\left(\hat{N}_c + 1\right).$$

C_τ is the number of vertices of the finite element used; since we use tetrahedra, we have $C_\tau = 4$.

$\hat{N}_C$ is the maximum number of overlapping subdomains any finite element point can

belong to; we have $\hat{N}_C = 10$ for problem **(4)** and $\hat{N}_C = 8$ for problems **(1)**–**(3)**. We note that $\hat{N}_C$ is equal to 8 for a cube that is partitioned into smaller cubes, in case a sufficiently small overlap is used.

N^ξ is the maximum number of interface components of any subdomain. It is equal to 94, 42, 101, and 106 for AGDSW and the respective problems **(1)**–**(4)**. As expected, N^ξ is significantly smaller for the reduced variants; cf. table 6.1. Histograms for the distribution of the number of interface components per subdomain are shown in fig. A.1. More detailed statistics for each type of interface component are given in fig. A.2.

Finally, the constant $\mathcal{C}$ enters the condition number bound. Let $\mathcal{C}_{ij}$ be the number of interface components that touch both Ω_i and Ω_j. Note that $\mathcal{C}_{ii}$ is the number of interface components of Ω_i and, thus, $N^\xi = \max_{i=1,\dots,N} \mathcal{C}_{ii}$. By definition of $\mathcal{C}$ in (6.35), we have

$$\mathcal{C} = \max_{i=1,\dots,N} \sum_{j=1}^{N} \mathcal{C}_{ij}.$$

It follows that

$$N^\xi \leq \mathcal{C} \leq N^\xi \max_{\xi \in \mathcal{P}} |n(\xi)|.$$

In section 6.4.3, we will show that $\mathcal{C}$ can be replaced by another constant that is smaller for some interface decompositions. Specifically, it is smaller if interface components exist that span across many subdomains. Regardless, it is usually desirable to minimize $\mathcal{C}$: interface components that span across many subdomains increase the communication cost in a parallel setting and can also increase the computational effort for the setup and solution of the respective generalized eigenvalue problem.

6.4.2. Dependence of the Eigenvalues on the Fine Mesh Resolution

The condition numbers of domain decomposition methods often depend on H/h—unlike the bound in theorem 6.1. There, the influence of H/h is hidden by the eigenvalue problem: many eigenvalues decrease with an increase in H/h.

In fig. 6.8, we consider an example of the Poisson equation on the unit square with a zero Dirichlet boundary condition on the left of the domain and a zero Neumann condition on the remaining boundary. The domain is discretized with bilinear finite elements and
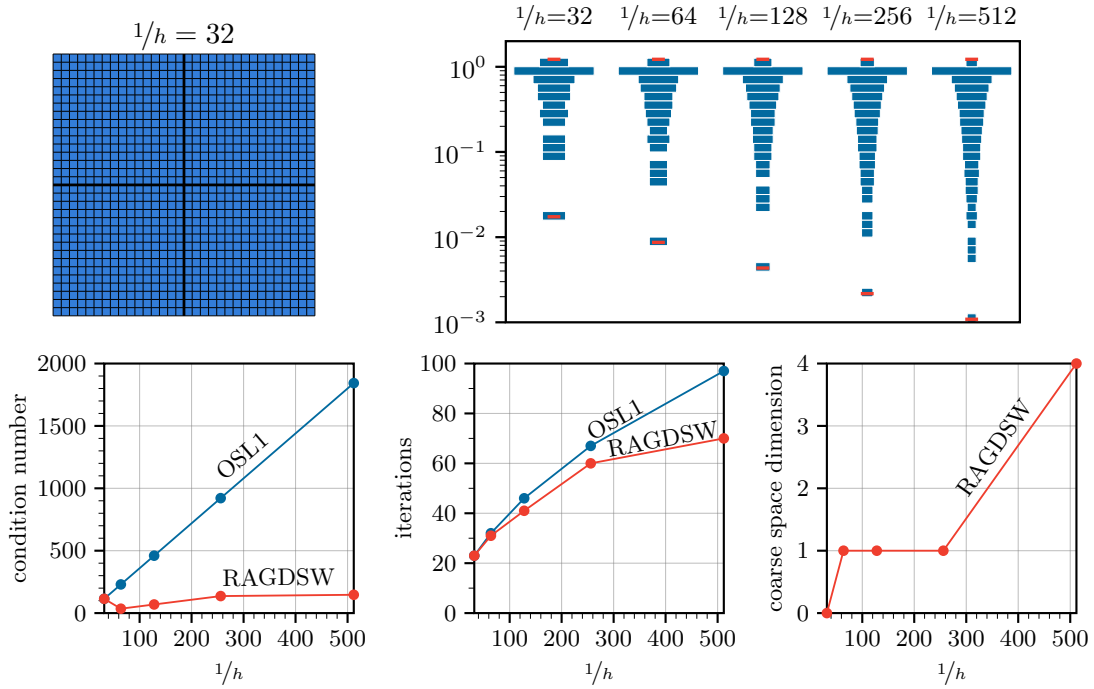
Figure 6.8.: **(Top left)** Sample mesh with 4 subdomains and 32^2 elements. **(Top right)** Histograms of the eigenvalues of the RAGDSW generalized eigenvalue problems for different fine mesh resolutions. Minima and maxima are marked with red bars. The width of bars indicates the number of eigenvalues in the corresponding range; it is scaled with the cube root to improve the visibility of small bars; it is not comparable between different columns. Condition numbers **(bottom left)** and numbers of iterations **(bottom center)** for OSL1 and RAGDSW. Coarse space dimension **(bottom right)** for RAGDSW.

subdivided into 2×2 subdomains. To analyze the effect of the fine mesh resolution h, we vary $1/h$, where we use the length of a side of a square finite element for h (contrary to its definition in section 1.2 based on the diameter). For the preconditioning, we employ OSL1 with an overlap of one layer of finite elements and, based thereon, RAGDSW with a single interface component. For the selection of eigenfunctions, the tolerance 0.01 is used.

In fig. 6.8 (top right), eigenvalues of the generalized eigenvalue problem are shown for
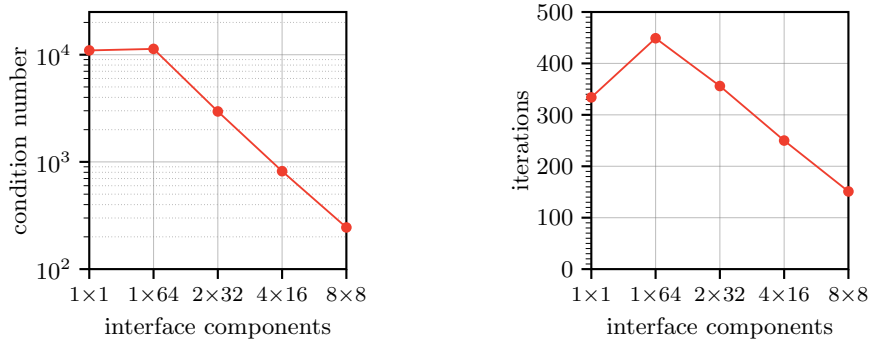
Figure 6.9.: Condition numbers **(left)** and numbers of iterations **(right)** for different types of interface partitions and the GDSW coarse space.

different mesh resolutions; we see that the spectrum is spread out with an increase in the mesh resolution. As a result, the coarse space dimension increases from zero to four (bottom right), and we obtain a robust preconditioner (bottom left and bottom center).

6.4.3. Interface Components Spanning Many Subdomains

Nonadaptive Coarse Space: GDSW

Coarse functions that span large sections of the domain can have a negative impact on the weak scalability of the algorithm. As an example, we consider the Poisson equation on the unit square with a zero Dirichlet boundary condition on the left of the domain and a zero Neumann condition on the remaining boundary. The domain is discretized with bilinear finite elements and subdivided into 64^2 subdomains, each with 16 elements.

To define a partition of the interface, we decompose the domain into $N_x \times N_y$ rectangles and assign all nodes that lie inside a rectangle to an interface component. We choose several partitions such that the number of interface components $N_x \cdot N_y = 64$ is constant. Note that a larger coarse space can outperform a competing one even if the individual coarse functions are of lesser quality, because of the better approximation that may be achieved by sheer quantity. As a result, we keep the number of interface components constant.

Furthermore, we make a comparison with the extreme case in which the entire interface

is used as the only interface component. The interface components of the partition 1×64 are as wide as one subdomain in the y-direction and span the entire domain in the x-direction; the ones of the partition 8×8 span 8 subdomains in both coordinate directions. The other partitions used are 2×32, 4×16, and—as mentioned above—1×1.

In fig. 6.9, we analyze the performance of the GDSW coarse space with regard to the different interface decompositions. An overlap of one layer of finite elements is used. The results show that the smaller the diameter of the interface components is, the better the algorithm performs. Furthermore, if only a single interface component is used, the condition number is similar to the 1×64 case. The number of iterations is significantly lower even though we use only a single interface component instead of 64. We conclude that the diameter of an interface component plays a crucial role for a nonadaptive GDSW-type coarse space, and that we lose scalability by using components that are too large. We remark that the results—for interface components spanning many subdomains—show a dependence of the condition number on $H_{\mathcal{P}}/h$, where $H_{\mathcal{P}}$ is the largest diameter of an interface component.

Adaptive Coarse Space: RAGDSW

The constant $\mathcal{C}$ in the condition number bound in theorem 6.1 is equal to the number of subdomains if only a single interface component is used. For GDSW, we have seen above that the size of the interface component influences the scalability of the algorithm. Thus, supported by the dependence of the condition number bound on $\mathcal{C}$, the question arises whether this holds true for the adaptive variants as well.

Unlike the bound suggests, numerical results do not show this dependence. We consider Poisson's equation on a thin, elongated rectangle, a zero Dirichlet boundary condition on the left of the domain, and a zero Neumann condition on the remaining boundary. The domain is discretized with bilinear finite elements and subdivided lengthwise into $N = 2^k$, $k \in \{3, 4, \ldots, 9\}$, subdomains, each with 16 elements; cf. fig. 6.10 (top left).

The RAGDSW coarse space is used for preconditioning, with a tolerance of 0.05 for the selection of eigenfunctions and an overlap of one layer of finite elements. Figure 6.10
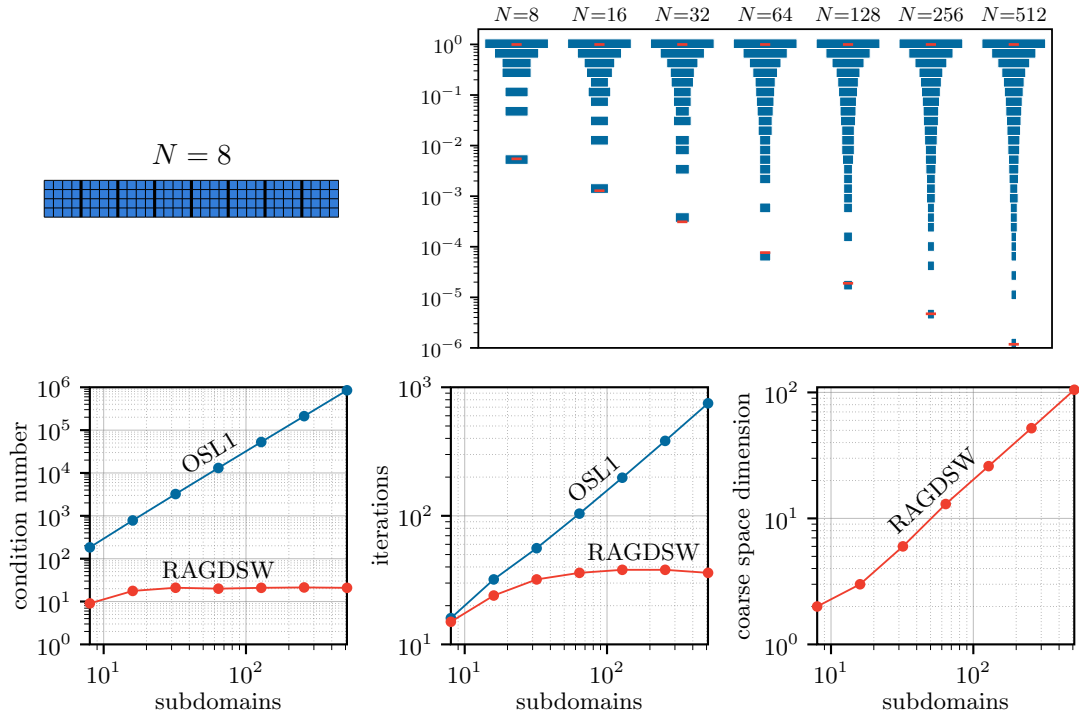
Figure 6.10.: **(Top left)** Sample mesh with $N = 8$ subdomains and 16 elements per subdomain. **(Top right)** Histograms of the eigenvalues of the RAGDSW generalized eigenvalue problem for different numbers of subdomains: The entire interface is used as the only interface component. Minima and maxima are marked in red. The width of bars indicates the number of eigenvalues in the corresponding range; it is not comparable between different columns; it is scaled with the cube root to make narrow bars wider. Condition numbers **(bottom left)** and numbers of iterations **(bottom center)** for OSL1 and RAGDSW. **(Bottom right)** Dimension of the RAGDSW coarse space.

(top right) shows that the eigenvalues of the (single) RAGDSW generalized eigenvalue problem decrease in magnitude with an increasing number of subdomains. On the bottom right, the growth of the coarse space dimension is shown. From the bottom-left and -center graphs, we conclude that RAGDSW is scalable for this type of interface "decomposition."

6. Theory of Adaptive GDSW-Type Coarse Spaces

In fact, we can prove that interface components spanning many subdomains do not negatively influence the condition number bound. From (6.43) and corollary 6.2 follows

$$\sum_{l=1}^{N} \left|I^h\big(\theta_l(u-u_0)\big)\right|^2_{a(\tilde{\Omega}_l\backslash\Omega_l)} \le C_{\mathrm{inv}} C_\tau \sum_{l=1}^{N} \sum_{\xi\in\mathcal{P}(\Omega_l)} \frac{1}{tol_\xi} \sum_{k\in n(\xi)} |u|^2_{a(\Omega_k)} \le C_{\mathrm{inv}} C_\tau \mathcal{C}^{tol_\xi} |u|^2_{a(\Omega)}.$$

Corollary 6.2 followed from lemma 6.5. Therein, we can change the lines of proof beneath (6.41). Let Ψ_l satisfy the assumptions in lemma 6.5 for $l=1,\ldots,N$. We obtain

$$\begin{aligned}
\sum_{l=1}^{N} \left|I^h\big(\Psi_l\cdot(u-u_0)\big)\right|^2_{a(\tilde{\Omega}_l\backslash\Omega_l)} &\le C_\tau \sum_{l=1}^{N} \sum_{\xi_i\in\mathcal{N}_{ec,\mathcal{P}}(\Omega_l)} |z_{\xi_i}(u-u_0)|^2_{a(\Omega_{\xi_i})} \\
&= C_\tau \sum_{\xi_i\in\mathcal{N}_{ec,\mathcal{P}}} |n(\xi_i)|\cdot|z_{\xi_i}(u-u_0)|^2_{a(\Omega_{\xi_i})} \\
&= C_\tau \sum_{\xi\in\mathcal{P}} \sum_{i=1}^{n_\xi} |n(\xi_i)|\cdot|z_{\xi_i}(u-u_0)|^2_{a(\Omega_{\xi_i})} \\
&\le C_\tau \sum_{\xi\in\mathcal{P}} \Big(\max_{i=1,\ldots,n_\xi} |n(\xi_i)|\Big) \beta_\xi^K(u-u_0,u-u_0). \quad (6.47)
\end{aligned}$$

Then, we continue as before and obtain with $\Psi_l=\theta_l$ that

$$\begin{aligned}
\sum_{l=1}^{N} \left|I^h\big(\theta_l(u-u_0)\big)\right|^2_{a(\tilde{\Omega}_l\backslash\Omega_l)} &\le C_{\mathrm{inv}} C_\tau \sum_{\xi\in\mathcal{P}} \frac{1}{tol_\xi} \Big(\max_{i=1,\ldots,n_\xi} |n(\xi_i)|\Big) \sum_{k\in n(\xi)} |u|^2_{a(\Omega_k)} \\
&\le C_{\mathrm{inv}} C_\tau \hat{\mathcal{C}}^{tol_\xi} |u|^2_{a(\Omega)},
\end{aligned}$$

where

$$\hat{\mathcal{C}}^{tol_\xi} := \max_{1\le l\le N} \sum_{\xi\in\mathcal{P}(\Omega_l)} \frac{\max_{i=1,\ldots,n_\xi} |n(\xi_i)|}{tol_\xi}. \quad (6.48)$$

For the interface partitions constructed in chapter 5, the multiplicity $|n(\xi)|$ of ξ is equal to $\max_{i=1,\ldots,n_\xi} |n(\xi_i)|$. Thus, in this case, we have

$$\hat{\mathcal{C}}^{tol_\xi} = \max_{1\le l\le N} \sum_{\xi\in\mathcal{P}(\Omega_l)} \frac{|n(\xi)|}{tol_\xi} = \mathcal{C}^{tol_\xi}.$$

The constant $\hat{\mathcal{C}}^{tol_\xi}$ is smaller than $\mathcal{C}^{tol_\xi}$ for components spanning many subdomains. For example, if the entire interface is used as the only interface component ξ—and using $tol_\xi=1$ for simplicity—we have $N^\xi=1$, $\mathcal{C}^{tol_\xi}=\mathcal{C}=N$, and $\hat{\mathcal{C}}^{tol_\xi}=\max_{\xi_i\in\mathcal{N}_\Gamma} |n(\xi_i)|$, where $\mathcal{N}_{\Gamma^h}$ is the set of NECs of the interface (as per sections 5.1 and 5.6). For problems **(1)**–**(3)**, we have a maximum of five subdomains adjacent to any finite element node; that is, $\max_{\xi_i\in\mathcal{N}_{\Gamma^h}} |n(\xi_i)|=5$; for problem **(4)**, we have $\max_{\xi_i\in\mathcal{N}_{\Gamma^h}} |n(\xi_i)|=6$.

144

7. ACMS-Type Coarse Spaces

This chapter is partly based on [HKKR18b] and [HKKR].

For AGDSW and RAGDSW, we have incorporated energy-minimizing extensions into the generalized eigenvalue problem to obtain a small coarse space dimension; cf. sections 3.3.2, 3.4.1, and 4.5. Subsequently, we have replaced the GDSW with the RGDSW interface partition to obtain a further reduction in the coarse space dimension. In the following, we use the same interface partitions that have already been introduced but change another aspect of the generalized eigenvalue problems. Specifically, we enforce additional zero Dirichlet conditions during the construction of the energy-minimizing extensions $\mathcal{H}_{\xi \to \Omega_\xi}(\cdot)$.

This approach was inspired by the multiscale discretization method ACMS (approximate component mode synthesis) [HL10; HK14; HHKR15; Hei16; HKKR18b; HKKR]. Therein, special basis functions that correspond to our coarse functions are constructed in order to reduce the dimension of the finite element space. The construction process of the special basis functions is identical to that of our coarse basis functions. However, the goal of the ACMS method is to obtain a finite element solution that approximates the exact solution to the variational problem, using fewer degrees of freedom. Our goal, on the other hand, is to build a robust preconditioner. To this end, we need to modify the ACMS space; see [HKKR18b], where this was done for the first time for two-dimensional diffusion problems. In [HKKR], the approach is generalized to three dimensions and linear elasticity problems. Since a few modifications are required, we have added the prefix OS (overlapping Schwarz) to ACMS and named the resulting coarse space OS–ACMS.

7.1. The Discretization Method ACMS

In structural dynamics, it is common to perform modal analyses on individual components of, for example, an aircraft, and then synthesize the local solutions to obtain an approximate global solution. This idea of component mode synthesis (cf. [Hur60; Hur65; CB68; HL10]) can be used to define special finite element functions that approximate a solution more accurately than standard finite element functions, using a comparable number of degrees of freedom.

To construct special basis functions for a two-dimensional diffusion problem—based on the idea of component mode synthesis—in [HL10], the authors first define the CMS (component mode synthesis) special finite element method. For the setup, we solve a generalized eigenvalue problem on each subdomain and one on the entire interface Γ. The solution of an eigenvalue problem on the entire interface is computationally expensive and difficult to parallelize. The authors, thus, propose to approximate the CMS method by replacing the interface problem with edge problems and by using special vertex functions; the resulting method is named ACMS.

The edge eigenvalue problems are given by: find $\tau_e \in X^h(e)$ such that

$$a_{\Omega_e}\big(\mathcal{H}_{\Gamma^h \to \Omega_e}\big(z_e(\tau_{*,e})\big) \,, \, \mathcal{H}_{\Gamma^h \to \Omega_e}\big(z_e(\theta)\big) \big) = \lambda_{*,e} \beta_e^{M,\mathcal{H}}\big(\tau_{*,e} \,, \, \theta \big) \tag{7.1}$$

for all $\theta \in X^h(e)$, where $\beta_e^{M,\mathcal{H}}\big(\tau_{*,e} \,, \, \theta \big) \coloneqq \big(\mathcal{H}_{\Gamma^h \to \Omega_e}\big(z_e(\tau_{*,e})\big) \,, \, \mathcal{H}_{\Gamma^h \to \Omega_e}\big(z_e(\theta)\big) \big)_{L^2(\Omega_e)}$. We note two differences between $\beta_e^{M,\mathcal{H}}$ and the bilinear form β_e^{M}, which we have used before (see (6.22)): the new bilinear form $\beta_e^{M,\mathcal{H}}$ does not use the coefficient function and mesh parameters for scaling, but it contains an energy-minimizing extension.

For the vertices, ACMS uses the following type of MsFEM functions (multiscale finite element method [HW97; HWC99; EH09]): For each edge that is incident to a vertex, a solution to the underlying diffusion problem on the edge is computed using the boundary values 1 (at the selected vertex) and 0. The edge values are then extended by zero to the remaining interface and, subsequently, energy-minimally to the interior. Examples are given by the function in fig. 7.1 (top: center right) and by the function in fig. 7.2 (rightmost).
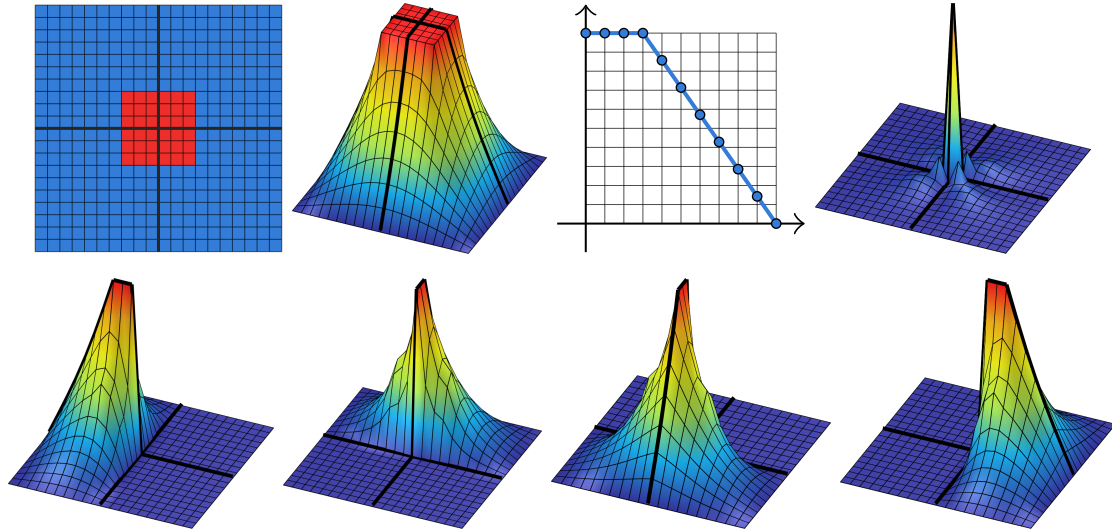
Figure 7.1.: Coefficient function and coarse functions for a two-dimensional diffusion problem on $\Omega = [0,1]^2$ with a zero Dirichlet boundary condition on $\partial\Omega$. **(Top left)** Coefficient function with $E = 10^6$ in red and $E = 1$ in blue on a 2×2 domain; **(top: center left, center right)** sum of AGDSW vertex and edge functions and edge values of the sum on a coarse edge. **(Top right)** AGDSW vertex function. **(Bottom)** AGDSW edge functions corresponding to the smallest eigenvalue on each coarse edge.

7.2. The OS–ACMS Coarse Space

Before we give the technical definition of the OS–ACMS coarse space, let us consider two-dimensional diffusion problems to motivate the necessary changes to the ACMS space to obtain a robust preconditioner.

We consider the problem in fig. 7.1 (top left) with an inclusion of a large coefficient, centered at the coarse node. The respective AGDSW edge functions (bottom) corresponding to the smallest eigenvalues are shown in addition to the AGDSW vertex function (top: rightmost). In section 1.5, we have motivated that only one coarse function is required for the coefficient function in fig. 7.1 to obtain a robust method. If we compute the sum of all AGDSW coarse functions—see fig. 7.1 (top: center left)—we obtain a coarse

function that, by itself, is sufficient to achieve robustness. In fact, this coarse function is the MsFEM-type function used by the ACMS space. By prescribing a value of 1 at the coarse node, 0 at the other end of an edge, and solving the underlying diffusion problem on the edge, we obtain the edge function in fig. 7.1 (top: center right). Repeating this for all edges and extending the computed interface values energy-minimally to the interior, we obtain the displayed MsFEM function (top: center left).

Our goal is to change the AGDSW generalized edge eigenvalue problems to exclude coarse edge functions as shown in fig. 7.1 (bottom), in case these functions are associated with a patch of large coefficients that touches the coarse node. This is achieved by prescribing a zero Dirichlet condition at the boundary nodes of the edge for the computation of the energy-minimizing extension in (6.6). In that case, a slope to zero on a patch of large coefficients is forced, resulting in a large energy and, thus, a relatively large eigenvalue; cf. remark 6.1. The left-hand side of the new generalized edge eigenvalue problem is given by

$$a_{\Omega_e}\big(\,\mathcal{H}_{\bar{e}\to\Omega_e}(z_e(\tau))\,,\,\mathcal{H}_{\bar{e}\to\Omega_e}(z_e(\theta))\,\big),\quad \tau,\theta \in X^h(e),$$

where $\bar{e}$ is the union of the edge e and its boundary nodes. Note that, for the example in fig. 7.1, the boundary nodes ∂e of e consist only of one coarse node. This is sufficient as, by definition of the energy-minimizing extension in (6.6), a zero Dirichlet condition is already prescribed on $\partial\Omega_D$—this differs from the definition in [HKKR18b], where a zero Neumann condition is prescribed on $\partial\Omega_D$. Furthermore, we do not prescribe a zero Dirichlet condition at a boundary node of an edge if the node lies on the Neumann boundary $\partial\Omega_N$.

An MsFEM-type function, which will replace the AGDSW vertex function, is now required to deal with patches of large coefficients that touch a coarse node. For simple problems as in fig. 7.1, it is sufficient to use the MsFEM-type functions of the ACMS space. The same type of MsFEM function is also used for the adaptive coarse spaces in [GLR15; GL17] and, based on the same idea, in [EMR19].

However, for our coarse space, we require a different type of MsFEM function: We consider a model problem with a comb-like coefficient function in fig. 7.2. The sum
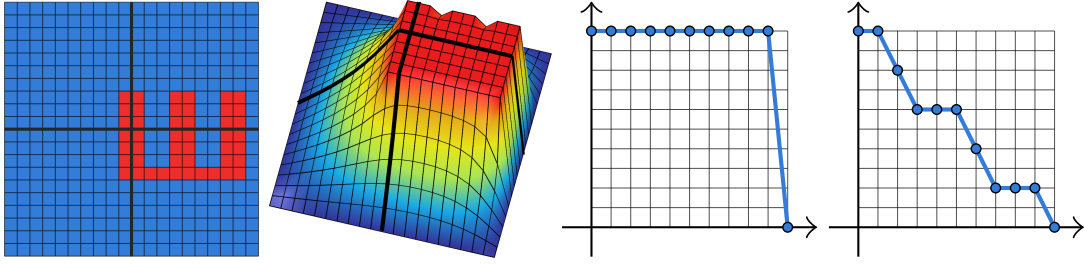
Figure 7.2.: Coefficient function and coarse function for a two-dimensional diffusion problem on $\Omega = [0,1]^2$, with a zero Dirichlet boundary condition on $\partial\Omega$. **(Left)** Coefficient function with $E = 10^6$ in red and $E = 1$ in blue on a 2×2 domain; **(center left)** sum of AGDSW vertex and edge functions and **(center right)** edge values of the sum on the coarse edge that is on the right side of the domain. **(Right)** Edge values (for the coarse edge on the right) of the MsFEM function of the ACMS space.

of the AGDSW coarse functions—using one coarse function per edge as in fig. 7.1—is displayed in the center left. The new type of edge eigenvalue problem, however, will not have small eigenvalues and, thus, coarse edge functions will not be constructed (given a sufficiently large tolerance). For the coarse edge on the right side of the domain, the MsFEM function of the ACMS space is shown in fig. 7.2 (rightmost). If these edge values were extended energy-minimally to the interior, the resulting coarse function would look very different from the one displayed in the center left; moreover its energy would be large, and the resulting coarse space would not be robust.

We need to define edge problems such that we obtain the edge function displayed in the center right of fig. 7.2. This can be achieved as follows: given the Dirichlet conditions in the associated coarse nodes, among all possible edge functions, find the edge function whose energy-minimizing extension has the smallest energy on Ω_e. In other words: Let $X_D^h(\overline{e}) \subset X^h(\overline{e})$ be the space that satisfies the Dirichlet conditions in the boundary nodes of e. Then, find $\tau \in X_D^h(\overline{e})$ such that

$$\left| \mathcal{H}_{\overline{e} \to \Omega_e}(\tau) \right|^2_{a(\Omega_e)} = \min_{\theta \in X_D^h(\overline{e})} \left| \mathcal{H}_{\overline{e} \to \Omega_e}(\theta) \right|^2_{a(\Omega_e)}.$$

This MsFEM-type function was introduced in [HKKR18b] and later used in [HL20b; HL20a; HKL22] as well.

Using the slab variant demands an adjustment of the analysis carried out above. If the slab is sufficiently small, a coarse edge function may be constructed despite the new type of eigenvalue problem. For example, if the slab were given by three layers of finite elements surrounding the right coarse edge of the problem in fig. 7.2, we would not detect that the three channels of large coefficients are connected; two coarse edge functions associated with small eigenvalues would be constructed. In that case, the edge values of a suitable MsFEM-type function would be given by fig. 7.2 (rightmost), which displays the (standard) MsFEM function that is used by the ACMS space. Since the detection of the connected component of large coefficients is rooted in the use of an energy-minimizing extension in the generalized eigenvalue problem, and since the coarse spaces in [GLR15; GL17; EMR19] do not make use of energy-minimizing extensions in such a way, this explains why they can use the standard MsFEM-type functions of the ACMS space.

To construct MsFEM-type functions that are consistent with our coarse edge functions, we must use the same type of energy-minimizing extension as used for the construction of coarse edge functions, for example, $\mathcal{H}_{\overline{e} \to \Omega_e^l}(\cdot)$ in the case of the slab variant.

For coarse faces, we use the same type of eigenvalue problem as for coarse edges. The boundary nodes of coarse faces are given by adjacent coarse nodes as well as adjacent coarse edges.

For the extension of edge eigenfunctions in three dimensions to construct coarse edge functions, we encounter the same difficulty as we have encountered in two dimensions for the computation of edge values of MsFEM functions: we require an extension from the boundary nodes of an adjacent coarse face to the coarse face itself. Fortunately, the extension is computed analogously to the two-dimensional case: We extend the edge values by zero to the boundary of an adjacent coarse face. Using the extended values as Dirichlet boundary conditions, we compute a face function whose energy-minimizing extension has the smallest energy among all possible face functions. The resulting function is then extended by zero to the entire interface and energy-minimally to the interior.

face eigenfunction $\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad$ $f \xrightarrow{z} \Gamma \xrightarrow{\mathcal{H}} \Omega$

edge eigenfunction $\quad\quad\quad\quad\quad$ $e \xrightarrow{z} \partial f \xrightarrow{\mathcal{H}} \overline{f} \xrightarrow{z} \Gamma \xrightarrow{\mathcal{H}} \Omega$

MsFEM-type function $\quad$ $\nu \xrightarrow{z} \partial e \xrightarrow{\mathcal{H}} \overline{e} \xrightarrow{z} \partial f \xrightarrow{\mathcal{H}} \overline{f} \xrightarrow{z} \Gamma \xrightarrow{\mathcal{H}} \Omega$
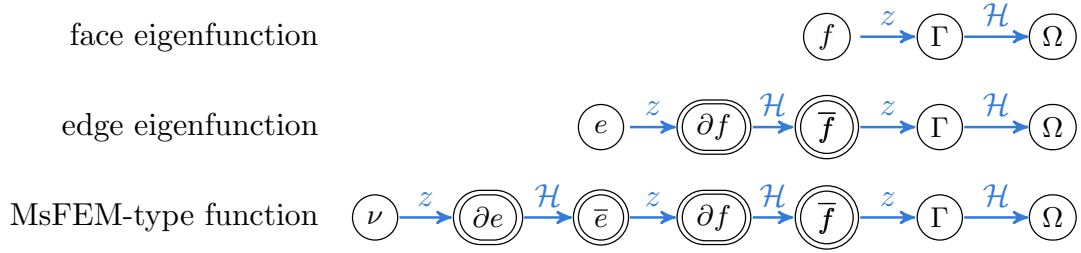
Figure 7.3.: Schematics of the cascaded extensions used to construct MsFEM-type, coarse edge and face functions. ν refers to a coarse node, e to a coarse edge, and f to a coarse face. z is an extension-by-zero and $\mathcal{H}$ an energy-minimizing extension. A highlighted node indicates that multiple components are affected; for example, we extend edge functions to all adjacent faces.

The three-dimensional MsFEM function is constructed by computing the edge values identically to the two-dimensional case and by then extending the values energy-minimally to the faces as above for edge eigenfunctions. The remainder is carried out as in the two-dimensional case. These cascaded extensions are visualized with schematics in fig. 7.3.

The above concept to enforce additional Dirichlet conditions during the computation of the energy-minimizing extension that is associated with the generalized eigenvalue problem will be generalized in section 7.5. In fact, the formulation will even encompass adaptive GDSW-type coarse spaces.

A final modification is required to obtain a robust preconditioner. The right-hand side of the ACMS edge eigenvalue problem in (7.1) contains a specific energy-minimizing extension, such that the right-hand side can assume small values for suitable functions for which it should take on large values. As a result, we will replace it with one of the right-hand sides introduced for AGDSW and RAGDSW. Historically, we use a right-hand side for OS–ACMS that is based on the scaled L^2-inner product; see section 6.1.4.

7.2.1. Technical Preliminaries

The OS–ACMS coarse space uses the same interface decomposition as GDSW and AGDSW, that is, a decomposition into coarse nodes $\mathcal{V}$, coarse edges $\mathcal{E}$, and coarse

faces $\mathcal{F}$ (in three dimensions). In the following, we give a description for the three-dimensional setting. The two-dimensional case is handled analogously: Edge functions in two dimensions are constructed like face functions in three dimensions. In three dimensions, vertex functions are defined via a cascade of extensions from vertices to edges to faces to the interior. In two dimensions, the extension to faces is simply removed, such that we obtain an extension from vertices to edges to the interior.

Generalized eigenvalue problems are used on coarse edges and faces to construct coarse functions. Moreover, we require the use of MsFEM-type functions, which are associated with coarse nodes. In the following, we first define generalized eigenvalue problems on coarse edges and faces. Then, we extend the eigenfunctions, using a new type of extension operator as introduced above.

For an interface component $\xi \in \mathcal{P}$, by $\overline{\xi}$ we denote the union of ξ and its "boundary nodes" $\partial \xi$:

$$\overline{\xi} := \xi \cup \partial \xi.$$

The notion of boundary nodes is—for the most part—intuitive for a structured domain decomposition. It always holds that $\partial \xi \subset \Gamma^h$; that is, nodes on the Dirichlet boundary $\partial \Omega_D$ are never part of $\overline{\xi}$. For an edge $e \in \mathcal{E}$, we define ∂e as the set of coarse nodes incident to e. For a face $f \in \mathcal{F}$, we define ∂f as the set of adjacent coarse nodes and the nodes of adjacent coarse edges. For vertices $\nu \in \mathcal{V}$, we set $\partial \nu := \emptyset$.

We emphasize that the definition of $\overline{\xi}$ above is only valid for the coarse space OS–ACMS of this section. For the generalization of OS–ACMS in section 7.5, the definition depends on the type of boundary conditions used for the energy-minimizing extension. In general, $\partial \xi$ is given by all nodes of $\Gamma^h \cap \overline{\Omega}_\xi$ that a zero Dirichlet condition is prescribed in; in the case of OS–ACMS, these are coarse nodes adjacent to a coarse edge, or coarse nodes and edges adjacent to a coarse face.

7.2.2. Generalized Eigenvalue Problems

Let $\xi \in \mathcal{P}$. The following generalized eigenvalue problem is valid for coarse edges $e \in \mathcal{E}$, coarse faces $f \in \mathcal{F}$, but also for coarse nodes $\nu \in \mathcal{V}$. Similar to AGDSW, however, we do

not actually have to solve an eigenvalue problem to construct vertex functions.

Find $\tau_{*,\xi} \in X^h(\xi)$ such that

$$\overline{\alpha}_\xi^K \left(\tau_{*,\xi} \, , \, \theta \right) = \lambda_{*,\xi} \beta_\xi^M \left(\tau_{*,\xi} \, , \, \theta \right) \quad \forall \theta \in X^h(\xi), \tag{7.2}$$

where $\beta_\xi^M(\cdot, \cdot)$ is the bilinear form from (6.22), using a scaled L^2-inner product, and

$$\overline{\alpha}_\xi^K \left(\tau_{*,\xi} \, , \, \theta \right) \coloneqq \alpha_{\overline{\xi}}^K \left(z_\xi(\tau_{*,\xi}) , \, z_\xi(\theta) \right), \quad \alpha_{\overline{\xi}}^K(u, v) \coloneqq a_{\Omega_\xi} \left(\mathcal{H}_{\overline{\xi} \to \Omega_\xi}(u) \, , \, \mathcal{H}_{\overline{\xi} \to \Omega_\xi}(v) \right). \tag{7.3}$$

Note that $\overline{\xi}$ does not include nodes on the Dirichlet boundary $\partial\Omega_D$, but by definition of (6.6), $\mathcal{H}_{\overline{\xi} \to \Omega_\xi}(\cdot)$ prescribes a zero Dirichlet condition at all $x^h \in \partial\Omega_D$.

In case $\overline{\xi} = \xi$, which can occur in unstructured domain decompositions or at the Neumann boundary $\partial\Omega_N$, eigenvalue problem (7.2) is identical to that of AGDSW–M. Let us note that using the mass variant for OS–ACMS has historical reasons. There is no limitation to using any of the other variants defined for AGDSW.

Let the eigenfunctions and corresponding eigenvalues of (7.2) be given by $\tau_{k,\xi}$, $\lambda_{k,\xi}$, where $1 \le k \le \dim(X^h(\xi))$. We select eigenfunctions that belong to eigenvalues smaller than or equal to a user-prescribed tolerance tol_ξ.

Compared to an adaptive GDSW-type generalized eigenvalue problem, additional zero Dirichlet conditions on $\partial\xi$ are prescribed. As a result, the left-hand side of (7.2) is invertible except for some rare cases, e.g.: if $\partial\xi = \emptyset$ and if all subdomains adjacent to ξ do not touch $\partial\Omega_D$, or—in the case of three-dimensional linear elasticity—if ξ is a straight edge with only one incident coarse node and no adjacent subdomains that touch $\partial\Omega_D$. In the latter case, we obtain two zero eigenvalues (note that the linearized rotation around the edge is not in the null space).

7.2.3. Extensions of Face and Edge Eigenfunctions

Unlike for adaptive GDSW-type coarse spaces, the extension of eigenfunctions differs based on the type of boundary conditions used in the energy-minimizing extension. For faces $f \in \mathcal{F}$, we have enforced zero Dirichlet conditions on all interface nodes adjacent to f. In that case, we can use the same extension as for adaptive GDSW-type coarse

spaces; that is, we extend an eigenfunction $\tau_{*,f}$ by zero to Γ^h and then energy-minimally to Ω: the coarse function associated with $\tau_{*,f}$ is, thus, defined as

$$v_{*,f} := \mathcal{H}_{\Gamma^h \to \Omega}\left(z_f(\tau_{*,f})\right).$$

Contrary, for an edge $e \in \mathcal{E}$, we had to enforce zero Dirichlet conditions on adjacent coarse nodes and not on all interface nodes adjacent to e, that is, not on adjacent coarse faces, as two adjacent interface components cannot both enforce Dirichlet conditions on each other; see section 7.5. As a result, we need to use an energy-minimizing extension to define values on the adjacent faces. Note that, if an edge does not have any adjacent faces, we obtain an adaptive GDSW-type eigenvalue problem, and the extension of an eigenfunction is carried out as for AGDSW.

By $\mathcal{B}_\mathrm{p}(f)$ we denote the parents of f: these are interface components on which we prescribe a zero Dirichlet condition in the energy-minimizing extension in generalized eigenvalue problem (7.2). As a result, the set of parents of f is given by all interface components that need to compute an energy-minimizing extension to f. For OS–ACMS, $\mathcal{B}_\mathrm{p}(f)$ is given by the set of coarse nodes and edges that are adjacent to f. By $\mathcal{B}_\mathrm{c}(e)$ we denote the set of children of an edge e, that is, the adjacent coarse faces.

We define an extension from e to $\overline{f}$ as

$$\tau_{*,e \to \overline{f}} := \mathcal{H}_{\partial f \to \Omega_f}\left(z_e(\tau_{*,e})\right)\Big|_{\overline{f}}. \tag{7.4}$$

Let

$$\tau_{*,e \to \Gamma^h} := \begin{cases} \tau_{*,e \to \overline{f}} & \text{on } f \in \mathcal{B}_\mathrm{c}(e), \\ \tau_{*,e} & \text{on } e, \\ 0 & \text{elsewhere on } \Gamma^h. \end{cases}$$

The coarse function associated with $\tau_{*,e}$ is then defined as

$$v_{*,e} := \mathcal{H}_{\Gamma^h \to \Omega}(\tau_{*,e \to \Gamma^h}).$$

Lemma 7.1. *Let $\xi \in \mathcal{P}$, $\tau_{\partial \xi} \in X^h(\partial \xi)$, and*

$$\tau_{\overline{\xi}} := \mathcal{H}_{\partial \xi \to \Omega_\xi}(\tau_{\partial \xi})\Big|_{\overline{\xi}}.$$

Then it holds that

$$|\mathcal{H}_{\partial\xi\to\Omega_\xi}(\tau_{\partial\xi})|_{a(\Omega_\xi)} = |\mathcal{H}_{\overline{\xi}\to\Omega_\xi}(\tau_{\overline{\xi}})|_{a(\Omega_\xi)} =: |\tau_{\overline{\xi}}|_{\alpha_{\overline{\xi}}^K}.$$

Proof. Let any θ be given such that $\theta = \mathcal{H}_{\partial\xi\to\Omega_\xi}(\theta)$—we recall that the energy-minimizing extension may not be uniquely defined; cf. remark 6.5. If

$$|\mathcal{H}_{\partial\xi\to\Omega_\xi}(\theta)|_{a(\Omega_\xi)} < |\mathcal{H}_{\overline{\xi}\to\Omega_\xi}(\theta)|_{a(\Omega_\xi)},$$

then

$$|\theta|_{a(\Omega_\xi)} < |\mathcal{H}_{\overline{\xi}\to\Omega_\xi}(\theta)|_{a(\Omega_\xi)},$$

which contradicts the energy-minimality of $\mathcal{H}_{\overline{\xi}\to\Omega_\xi}(\cdot)$. Equally, let $w := \mathcal{H}_{\overline{\xi}\to\Omega_\xi}(\theta)$. If

$$|\mathcal{H}_{\overline{\xi}\to\Omega_\xi}(\theta)|_{a(\Omega_\xi)} < |\mathcal{H}_{\partial\xi\to\Omega_\xi}(\theta)|_{a(\Omega_\xi)},$$

then

$$|w|_{a(\Omega_\xi)} < |\mathcal{H}_{\partial\xi\to\Omega_\xi}(w)|_{a(\Omega_\xi)},$$

which contradicts the energy-minimality of $\mathcal{H}_{\partial\xi\to\Omega_\xi}(\cdot)$. $\qquad\square$

From lemma 7.1 follows that $\tau_{\overline{\xi}}$ is a function whose extension $\mathcal{H}_{\overline{\xi}\to\Omega_\xi}(\tau_{\overline{\xi}})$ has the smallest energy $|\cdot|_{a(\Omega_\xi)}$ among all possible functions that are equal to $\tau_{\partial\xi}$ on $\partial\xi$. The corresponding variational formulation is given by: find $\tau_{\overline{\xi}} \in X^h(\overline{\xi})$ with $\tau_{\overline{\xi}}|_{\partial\xi} = \tau_{\partial\xi}$ such that

$$\alpha_{\overline{\xi}}^K\big(\tau_{\overline{\xi}}, z_\xi(\theta)\big) = a_{\Omega_\xi}\big(\mathcal{H}_{\overline{\xi}\to\Omega_\xi}(\tau_{\overline{\xi}}), \mathcal{H}_{\overline{\xi}\to\Omega_\xi}(z_\xi(\theta))\big) = 0 \quad \forall\theta \in X^h(\xi). \qquad (7.5)$$

7.2.4. MsFEM-Type Vertex Function

Let $\nu \in \mathcal{V}$ be a coarse node. Similarly to AGDSW, coarse functions associated with ν can be obtained by suitable extensions of solutions to the generalized eigenvalue problem in (7.2). Since we have $\overline{\nu} = \nu$, no additional Dirichlet conditions are enforced for the energy-minimizing extension. Following the discussion in the introduction of section 7.2, we have to compute an energy-minimizing extension from ν to all children $\mathcal{B}_c(\nu)$, where $\mathcal{B}_c(\nu)$ denotes the set of adjacent coarse edges and faces.

As for AGDSW, let $\tau_{*,\nu}$ be a function of the restriction of the problem's null space to ν. In the case of three-dimensional linear elasticity, these are the translation modes and are given by, for example, the three unit vectors of $\mathbb{R}^3$. Similarly to the extension of the edge eigenfunctions in the last section, we extend $\tau_{*,\nu}$ to all adjacent edges energy-minimally. Let e be an edge that is adjacent to ν. Then the extension from ν to $\bar{e}$ is defined as

$$\tau_{*,\nu\to\bar{e}} := \mathcal{H}_{\partial e\to\Omega_e}\big(z_\nu(\tau_{*,\nu})\big)\big|_{\bar{e}}. \tag{7.6}$$

Let f be a face adjacent to ν. First, we extend all edge functions and $\tau_{*,\nu}$ by zero to the boundary of f:

$$\tau_{*,\nu\to\partial f} := \begin{cases} \tau_{*,\nu\to\bar{e}} & \text{on } e \in \mathcal{E} \cap \mathcal{B}_{\mathrm{c}}(\nu) \cap \mathcal{B}_{\mathrm{p}}(f), \\ \tau_{*,\nu} & \text{on } \nu, \\ 0 & \text{elsewhere on } \partial f. \end{cases}$$

The extension to $\overline{f}$ is defined similarly to the extension of the edge eigenfunctions:

$$\tau_{*,\nu\to\overline{f}} := \mathcal{H}_{\partial f\to\Omega_f}(\tau_{*,\nu\to\partial f})\big|_{\overline{f}}. \tag{7.7}$$

As before, let

$$\tau_{*,\nu\to\Gamma^h} := \begin{cases} \tau_{*,\nu\to\overline{f}} & \text{on } f \in \mathcal{F} \cap \mathcal{B}_{\mathrm{c}}(\nu), \\ \tau_{*,\nu\to\bar{e}} & \text{on } e \in \mathcal{E} \cap \mathcal{B}_{\mathrm{c}}(\nu), \\ \tau_{*,\nu} & \text{on } \nu, \\ 0 & \text{elsewhere on } \Gamma^h. \end{cases}$$

We can now define the coarse function associated with $\tau_{*,\nu}$ as

$$v_{*,\nu} := \mathcal{H}_{\Gamma^h\to\Omega}(\tau_{*,\nu\to\Gamma^h}).$$

The OS–ACMS coarse space is defined as

$$V_{\mathrm{OS-ACMS}} := \mathrm{span}\Big(\{\, v_{k,\nu} : \nu \in \mathcal{V},\ k = 1,\ldots,\hat{d}\,\} \cup \{\, v_{*,\xi} : \xi \in \mathcal{E} \cup \mathcal{F},\ \lambda_{*,\xi} \leq tol_\xi \,\}\Big).$$

7.2.5. Matrix Formulation

Let ξ be an edge $e \in \mathcal{E}$ or a face $f \in \mathcal{F}$. As in chapter 3, let K^{Ω_ξ} be the stiffness matrix that is obtained by assembling $a_{\Omega_\xi}(\cdot,\cdot)$ with a Dirichlet boundary condition on $\partial\Omega_\xi \cap \partial\Omega_D$.

We partition K^{Ω_ξ} by the degrees of freedom associated with $\partial\xi$ and those with $\overline{\Omega}_\xi \setminus \partial\xi$; the latter set is denoted by R. We obtain

$$K^{\Omega_\xi} = \begin{pmatrix} K_{RR}^{\Omega_\xi} & K_{R,\partial\xi}^{\Omega_\xi} \\ K_{\partial\xi,R}^{\Omega_\xi} & K_{\partial\xi,\partial\xi}^{\Omega_\xi} \end{pmatrix}.$$

The energy-minimizing extension of $\tau_{\partial\xi}$ from $\partial\xi$ to R is given by a solution to

$$K_{RR}^{\Omega_\xi}\tau_R = -K_{R,\partial\xi}^{\Omega_\xi}\tau_{\partial\xi}. \tag{7.8}$$

We further partition $K_{RR}^{\Omega_\xi}$ by the degrees of freedom associated with ξ and the remainder $\tilde{R}$. We obtain

$$\begin{pmatrix} K_{\tilde{R}\tilde{R}}^{\Omega_\xi} & K_{\tilde{R}\xi}^{\Omega_\xi} \\ K_{\xi\tilde{R}}^{\Omega_\xi} & K_{\xi\xi}^{\Omega_\xi} \end{pmatrix} \begin{pmatrix} \tau_{\tilde{R}} \\ \tau_\xi \end{pmatrix} = - \begin{pmatrix} K_{\tilde{R},\partial\xi}^{\Omega_\xi} \\ K_{\xi,\partial\xi}^{\Omega_\xi} \end{pmatrix} \tau_{\partial\xi}.$$

As we are only interested in a solution on ξ to (7.8), it is given by a solution τ_ξ to

$$\overline{S}_{\xi\xi}\tau_\xi = -\overline{S}_{\xi,\partial\xi}\,\tau_{\partial\xi}, \tag{7.9}$$

where $\overline{S}_{\xi\xi}$ and $\overline{S}_{\xi,\partial\xi}$ are submatrices of

$$\overline{S}_{\overline{\xi}\,\overline{\xi}} = \begin{pmatrix} \overline{S}_{\xi\xi} & \overline{S}_{\xi,\partial\xi} \\ \overline{S}_{\partial\xi,\xi} & \overline{S}_{\partial\xi,\partial\xi} \end{pmatrix},$$

which is defined as

$$\overline{S}_{\overline{\xi}\,\overline{\xi}} := K_{\overline{\xi}\,\overline{\xi}}^{\Omega_\xi} - K_{\overline{\xi}\tilde{R}}^{\Omega_\xi}\left(K_{\tilde{R}\tilde{R}}^{\Omega_\xi}\right)^+ K_{\tilde{R}\overline{\xi}}^{\Omega_\xi}.$$

As before, $\left(K_{\tilde{R}\tilde{R}}^{\Omega_\xi}\right)^+$ is a pseudoinverse of $K_{\tilde{R}\tilde{R}}^{\Omega_\xi}$. We note that $\overline{S}_{\xi\xi}$ differs from the Schur complement in (3.2), which is used for AGDSW.

The matrix formulation of generalized eigenvalue problem (7.2) is given by

$$\overline{S}_{\xi\xi}\tau_{*,\xi} = \lambda_{*,\xi}M_{\xi\xi}\tau_{*,\xi}, \tag{7.10}$$

where the mass matrix $M_{\xi\xi}$ corresponding to $\beta_\xi^M(\cdot,\cdot)$ was defined in section 3.3.4. Let the eigenvalues of (7.10) be sorted in nondescending order,

$$0 \leq \lambda_{1,\xi} \leq \lambda_{2,\xi} \leq \cdots \leq \lambda_{m,\xi},$$

where m denotes the number of unknowns of ξ. We select all eigenvectors $\tau_{*,\xi}$ from (7.10) that correspond to eigenvalues smaller than or equal to a user-prescribed threshold $tol_\xi \geq 0$:

$$\lambda_{*,\xi} \leq tol_\xi.$$

If ξ is a coarse face, then the remainder is identical to AGDSW: we extend the selected eigenvectors by zero to the interface nodes Γ^h—let the extensions be denoted by τ_{*,ξ,Γ^h}—and then energy-minimally to the interior to define the coarse functions

$$v_{*,f} := H_\Gamma \tau_{*,\xi,\Gamma^h},$$

where H_Γ was defined in (1.12).

If ξ is a coarse edge, we need to extend the edge values to the faces. As we have set up $S_{\overline{ff}}$ for a face eigenvalue problem, we can reuse it. Using (7.9), we obtain $\tau_{*,\overline{f}}$ from $\tau_{*,\partial f}$, where $\tau_{*,\partial f}$ is the extension-by-zero from $\tau_{*,e}$ to ∂f. We repeat this for all faces adjacent to e. The face and edge values are then extended by zero to Γ^h—let the extension be denoted by $\tau_{*,e\to\Gamma^h}$—and, as before, energy-minimally to Ω:

$$v_{*,e} := H_\Gamma \tau_{*,e\to\Gamma^h}.$$

For the MsFEM-type functions, we use the canonical basis vectors of $\mathbb{R}^{\hat{d}}$ for $\tau_{*,\nu}$. Let e be an edge adjacent to ν. We extend $\tau_{*,\nu}$ by zero to ∂e and then energy-minimally to e by using (7.9). We repeat this for all edges adjacent to ν. Note that, we can reuse $S_{\overline{ee}}$, which we have set up for the respective edge eigenvalue problem.

Then as above, using the edge values, we compute the extensions to the faces by using the already available $S_{\overline{ff}}$. We extend the values by zero to the entire interface—let the function be denoted by $\tau_{*,\nu\to\Gamma^h}$—and then energy-minimally to the interior:

$$v_{*,\nu} := H_\Gamma \tau_{*,\nu\to\Gamma^h}.$$

The columns of the matrix Φ of the Schwarz preconditioner are now given by the constructed vectors $v_{*,\xi}$.
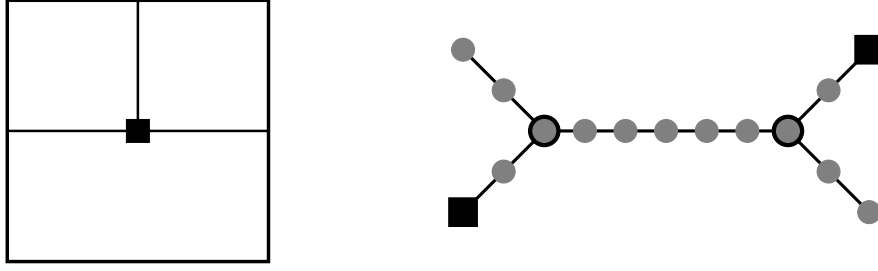
Figure 7.4.: **(Left)** Decomposition of a square into three subdomains with one coarse node (black square) and three coarse edges. **(Right)** Schematic of a coarse edge (gray nodes) with two junctions (marked with black circles) and only two incident coarse nodes (black squares) instead of the maximum of four.

Remark 7.1. *Let $e \in \mathcal{E}$ be a coarse edge. There are multiple cases we need to account for in an implementation. The edge may have only one instead of two incident coarse nodes; cf. fig. 7.4 (left). It may also have none at all; this can occur if both ends touch the Neumann boundary $\partial\Omega_N$, but it can also occur in the interior of Ω. Owing to the complexity of unstructured domain decompositions, there can even exist cases for which a coarse edge has more than two incident coarse nodes: the edge contains a junction. It may also have junctions without incident coarse nodes; cf. fig. 7.4 (right). The description of the OS–ACMS coarse space is valid for all these cases. Depending on the specific implementation, however, we need to be aware of such special cases.*

7.2.6. Variants of the OS–ACMS Coarse Space

We can use the same variants for OS–ACMS as for AGDSW. We can replace the right-hand side of generalized eigenvalue problem (7.2) with the bilinear form $\beta_\xi^K(\cdot, \cdot)$ from (6.10), which is based on a stiffness matrix. We can also use the lumped mass matrix of (6.27) or the lumped stiffness matrix of (6.11).

For the energy-minimizing extension operators—except for $\mathcal{H}_{\Gamma^h \to \Omega}(\cdot)$, which is used in the final step to construct a coarse function from interface values—we can replace the extension $\mathcal{H}_{\bar{\xi} \to \Omega_\xi}(\cdot)$ with a slab variant $\mathcal{H}_{\bar{\xi} \to \Omega_\xi^l}(\cdot)$ as in section 6.1.2, by the S-variant, or a

7. ACMS-Type Coarse Spaces

combination of the two. Accordingly, the left-hand side of (7.2) can be replaced by

$$\overline{\alpha}_\xi^{K,l}(\tau_\xi, \theta) := \alpha_{\overline{\xi}}^{K,l}(z_\xi(\tau_\xi), z_\xi(\theta)), \tag{7.11}$$

$$\overline{\alpha}_\xi^{S}(\tau_\xi, \theta) := \alpha_{\overline{\xi}}^{S}(z_\xi(\tau_\xi), z_\xi(\theta)), \tag{7.12}$$

$$\overline{\alpha}_\xi^{S,l}(\tau_\xi, \theta) := \alpha_{\overline{\xi}}^{S,l}(z_\xi(\tau_\xi), z_\xi(\theta)), \tag{7.13}$$

where

$$\alpha_{\overline{\xi}}^{K,l}(z_\xi(\tau_\xi), z_\xi(\theta)) := a_{\Omega_\xi^l}(\mathcal{H}_{\overline{\xi} \to \Omega_\xi^l}(z_\xi(\tau_\xi)), \mathcal{H}_{\overline{\xi} \to \Omega_\xi^l}(z_\xi(\theta))),$$

$$\alpha_{\overline{\xi}}^{S}(z_\xi(\tau_\xi), z_\xi(\theta)) := \sum_{k \in n(\xi)} a_{\Omega_k}(\mathcal{H}_{\overline{\xi} \to \Omega_k}(z_\xi(\tau_\xi)), \mathcal{H}_{\overline{\xi} \to \Omega_k}(z_\xi(\theta))),$$

$$\alpha_{\overline{\xi}}^{S,l}(z_\xi(\tau_\xi), z_\xi(\theta)) := \sum_{k \in n(\xi)} a_{\Omega_k^l}(\mathcal{H}_{\overline{\xi} \to \Omega_k^l}(z_\xi(\tau_\xi)), \mathcal{H}_{\overline{\xi} \to \Omega_k^l}(z_\xi(\theta))).$$

To obtain a robust algorithm, it is imperative that the same extension operators are used in the generalized eigenvalue problem and for the extensions in (7.4), (7.6), and (7.7). For the S-variant, the correct replacement of the extension operator is not given by $\mathcal{H}_{\partial\xi \to \Omega_\xi}^{S}(\cdot)$ from (6.17) as its restriction to $\overline{\xi}$ is not well defined. Instead, as for (7.5), the extension from $\partial\xi$ to ξ is defined by: find $\tau_{\overline{\xi}} \in X^h(\overline{\xi})$ with $\tau_{\overline{\xi}}|_{\partial\xi} = \tau_{\partial\xi}$ such that

$$\alpha_{\overline{\xi}}^{S}(\tau_{\overline{\xi}}, z_\xi(\theta)) = \sum_{k \in n(\xi)} a_{\Omega_k}(\mathcal{H}_{\overline{\xi} \to \Omega_k}(\tau_{\overline{\xi}}), \mathcal{H}_{\overline{\xi} \to \Omega_k}(z_\xi(\theta))) = 0 \quad \forall \theta \in X^h(\xi). \tag{7.14}$$

Remark 7.2. *For AGDSW, the extension operator $\mathcal{H}_{f \to \Omega_f}(\cdot)$, $f \in \mathcal{F}$, is used for face eigenproblems. During the application of $\mathcal{H}_{f \to \Omega_f}(\cdot)$, the information transfer between the subdomains adjacent to f can only occur via the boundary nodes of f. For OS–ACMS, we use the extension $\mathcal{H}_{\overline{f} \to \Omega_f}(\cdot)$. Since we now include the boundary nodes of f in the extension, the connection between the two subdomains adjacent to f is removed entirely. Consequently, using the S-variant for OS–ACMS face problems, we obtain the same generalized eigenvalue problems as with the original method.*

The following lemma states an energy-minimality property, which we rely on for the proof of a condition number bound.

Lemma 7.2. *Depending on the variant used, let $\alpha_{\bar{\xi}}(\cdot, \cdot)$ be $\alpha_{\bar{\xi}}^K(\cdot, \cdot)$, $\alpha_{\bar{\xi}}^{K,l}(\cdot, \cdot)$, $\alpha_{\bar{\xi}}^S(\cdot, \cdot)$, or $\alpha_{\bar{\xi}}^{S,l}(\cdot, \cdot)$. Similarly to (6.16), (6.19), and (6.21), we then have for $u \in X^h(\bar{\xi})$*

$$|u|_{\alpha_{\bar{\xi}}} \leq |u|_{\alpha_{\bar{\xi}}^K},$$

where

$$|u|_{\alpha_{\bar{\xi}}} := \sqrt{\alpha_{\bar{\xi}}(u, u)}.$$

Proof. We have

$$|u|^2_{\alpha_{\bar{\xi}}^{K,l}} = \alpha_{\bar{\xi}}^{K,l}(u, u) = \quad |\mathcal{H}_{\bar{\xi} \to \Omega_{\xi}^l}(u)|^2_{a(\Omega_{\xi}^l)} \leq \quad |\mathcal{H}_{\bar{\xi} \to \Omega_{\xi}}(u)|^2_{a(\Omega_{\xi}^l)} \leq |\mathcal{H}_{\bar{\xi} \to \Omega_{\xi}}(u)|^2_{a(\Omega_{\xi})},$$

$$|u|^2_{\alpha_{\bar{\xi}}^S} = \quad \alpha_{\bar{\xi}}^S(u, u) = \sum_{k \in n(\xi)} |\mathcal{H}_{\bar{\xi} \to \Omega_k}(u)|^2_{a(\Omega_k)} \leq \sum_{k \in n(\xi)} |\mathcal{H}_{\bar{\xi} \to \Omega_{\xi}}(u)|^2_{a(\Omega_k)} = |\mathcal{H}_{\bar{\xi} \to \Omega_{\xi}}(u)|^2_{a(\Omega_{\xi})},$$

$$|u|^2_{\alpha_{\bar{\xi}}^{S,l}} = \alpha_{\bar{\xi}}^{S,l}(u, u) = \sum_{k \in n(\xi)} |\mathcal{H}_{\bar{\xi} \to \Omega_k^l}(u)|^2_{a(\Omega_k^l)} \leq \sum_{k \in n(\xi)} |\mathcal{H}_{\bar{\xi} \to \Omega_{\xi}}(u)|^2_{a(\Omega_k^l)} \leq |\mathcal{H}_{\bar{\xi} \to \Omega_{\xi}}(u)|^2_{a(\Omega_{\xi})},$$

and, by definition, $|u|_{\alpha_{\bar{\xi}}^K} = |\mathcal{H}_{\bar{\xi} \to \Omega_{\xi}}(u)|_{a(\Omega_{\xi})}$. $\qquad\square$

The following lemma is the analogue of lemma 7.1 for the S-variant.

Lemma 7.3. *Let $\tau_{\bar{\xi}}$ be a solution to (7.14). Then we have*

$$|\tau_{\bar{\xi}}|_{\alpha_{\bar{\xi}}^S} \leq |\mathcal{H}_{\partial \xi \to \Omega_{\xi}}(\tau_{\bar{\xi}})|_{a(\Omega_{\xi})}.$$

Proof. For all $\theta \in X^h(\bar{\xi})$ satisfying $\theta|_{\partial \xi} = \tau_{\bar{\xi}}|_{\partial \xi}$, we have

$$|\tau_{\bar{\xi}}|^2_{\alpha_{\bar{\xi}}^S} = \alpha_{\bar{\xi}}^S(\tau_{\bar{\xi}}, \tau_{\bar{\xi}}) = \sum_{k \in n(\xi)} |\mathcal{H}_{\bar{\xi} \to \Omega_k}(\tau_{\bar{\xi}})|^2_{a(\Omega_k)} \leq \sum_{k \in n(\xi)} |\mathcal{H}_{\bar{\xi} \to \Omega_k}(\theta)|^2_{a(\Omega_k)} \qquad (7.15)$$

since $\tau_{\bar{\xi}}$ is energy-minimal. Let $\theta := \mathcal{H}_{\partial \xi \to \Omega_{\xi}}(\tau_{\bar{\xi}})$. Then, θ fulfills (7.15), and we have

$$\sum_{k \in n(\xi)} |\mathcal{H}_{\bar{\xi} \to \Omega_k}(\tau_{\bar{\xi}})|^2_{a(\Omega_k)} \leq \sum_{k \in n(\xi)} |\mathcal{H}_{\bar{\xi} \to \Omega_k}(\theta)|^2_{a(\Omega_k)}$$

$$\leq \sum_{k \in n(\xi)} |\mathcal{H}_{\bar{\xi} \to \Omega_{\xi}}(\theta)|^2_{a(\Omega_k)}$$

$$= |\mathcal{H}_{\bar{\xi} \to \Omega_{\xi}}(\theta)|^2_{a(\Omega_{\xi})}$$

$$= |\mathcal{H}_{\partial \xi \to \Omega_{\xi}}(\theta)|^2_{a(\Omega_{\xi})}$$

$$= |\mathcal{H}_{\partial \xi \to \Omega_{\xi}}(\tau_{\bar{\xi}})|^2_{a(\Omega_{\xi})},$$

where we have used lemma 7.1 in the second to last step. Note that we did not use $\tau_{\overline{\xi}}$ directly but defined a suitable function θ since we cannot assume that $\tau_{\overline{\xi}} = \mathcal{H}_{\partial\xi\to\Omega_\xi}(\tau_{\overline{\xi}})\big|_{\overline{\xi}}$ holds. $\qquad\square$

The statement and proof are analogous if the slab variant or a combination with the S-variant are used. Thus, if $\tau_{\overline{\xi}}$ is a solution to one of the energy-minimizing extensions from $\partial\xi$ to ξ, we have

$$|\tau_{\overline{\xi}}|_{\alpha_{\overline{\xi}}} \le |\mathcal{H}_{\partial\xi\to\Omega_\xi}(\tau_{\overline{\xi}})|_{a(\Omega_\xi)}, \tag{7.16}$$

where $\alpha_{\overline{\xi}}(\cdot,\cdot)$ is $\alpha_{\overline{\xi}}^K(\cdot,\cdot)$, $\alpha_{\overline{\xi}}^{K,l}(\cdot,\cdot)$, $\alpha_{\overline{\xi}}^S(\cdot,\cdot)$, or $\alpha_{\overline{\xi}}^{S,l}(\cdot,\cdot)$.

As before, we do not introduce a new acronym for the slab variant but indicate its usage by appending the slab size in parentheses. The original OS–ACMS coarse space with a mass matrix on the right-hand side is denoted by OS–ACMS or OS–ACMS–M. If the mass matrix is replaced by a stiffness matrix, we denote the resulting coarse space by OS–ACMS–K. Similarly, if the S-variant is used, the acronyms are given by OS–ACMS–S, OS–ACMS–S–M, and OS–ACMS–S–K.

Matrix Formulation

The matrix formulation of the slab variant is straightforward since we only need to replace Ω_ξ with Ω_ξ^l. In the following, we give the matrix formulation of the S-variant. The slab variant combined with the S-variant is then given if Ω_k, $k \in n(\xi)$, is replaced by Ω_k^l.

As in section 3.3.3, let K^{Ω_k} be the stiffness matrix that is obtained by assembling $a_{\Omega_k}(\cdot,\cdot)$ with a Dirichlet boundary condition on $\partial\Omega_k \cap \partial\Omega_D$. Similarly to section 7.2.5, we partition K^{Ω_k} by the degrees of freedom associated with $\partial\xi \cap \overline{\Omega}_k$, $\xi \cap \overline{\Omega}_k$, and $\overline{\Omega}_k \setminus \overline{\xi}$; the latter set is denoted by $\tilde{R}$. We define the local Schur complement

$$\overline{S}_{\overline{\xi}\overline{\xi}}^k := K_{\overline{\xi}\overline{\xi}}^{\Omega_k} - K_{\overline{\xi}\tilde{R}}^{\Omega_k}\left(K_{\tilde{R}\tilde{R}}^{\Omega_k}\right)^+ K_{\tilde{R}\overline{\xi}}^{\Omega_k}, \quad k \in n(\xi),$$

where $\left(K_{\tilde{R}\tilde{R}}^{\Omega_k}\right)^+$ is a pseudoinverse of $K_{\tilde{R}\tilde{R}}^{\Omega_k}$.

Let R^T_{ξ,Ω_k} denote the prolongation operator that maps the degrees of freedom of $\overline{\xi} \cap \overline{\Omega}_k$ to $\overline{\xi}$ (cf. section 3.3.3). The assembly of $\alpha^S_{\overline{\xi}}(\cdot,\cdot)$ is given by

$$\overline{S}^S_{\overline{\xi}\,\overline{\xi}} := \sum_{k \in n(\xi)} R^T_{\xi,\Omega_k} \overline{S}^k_{\xi\xi} R_{\xi,\Omega_k}.$$

We partition $\overline{S}^S_{\overline{\xi}\,\overline{\xi}}$ by ξ and $\partial\xi$:

$$\overline{S}^S_{\overline{\xi}\,\overline{\xi}} = \begin{pmatrix} \overline{S}^S_{\xi\xi} & \overline{S}^S_{\xi,\partial\xi} \\ \overline{S}^S_{\partial\xi,\xi} & \overline{S}^S_{\partial\xi,\partial\xi} \end{pmatrix}.$$

The generalized eigenvalue problem then reads

$$\overline{S}^S_{\xi\xi} \tau_{*,\xi} = \lambda_{*,\xi} M_{\xi\xi} \tau_{*,\xi}.$$

We note that $\overline{S}^S_{\xi\xi}$ differs from the Schur complement in (3.4), which is used for AGDSW–S.

We further need to replace the operators that extend energy-minimally from the vertices to the edges and those that extend from the edges to the faces. Let $\hat{\xi}$ be a coarse node $\nu \in \mathcal{V}$ or a coarse edge $e \in \mathcal{E}$, and let $\tau_{\hat{\xi}} \in X^h(\hat{\xi})$. If $\hat{\xi}$ is a coarse node, then let $\xi \in \mathcal{E}$ be an adjacent edge. If $\hat{\xi}$ is a coarse edge, then let $\xi \in \mathcal{F}$ be an adjacent face. Let $\tau_{\partial\xi}$ be the extension-by-zero of $\tau_{\hat{\xi}}$ to $\partial\xi$. As we have previously set up $S^S_{\overline{\xi}\,\overline{\xi}}$ for the corresponding edge or face problem, the extension τ_ξ from $\tau_{\partial\xi}$ is given by a solution to

$$\overline{S}^S_{\xi\xi} \tau_\xi = -\overline{S}^S_{\xi,\partial\xi} \tau_{\partial\xi};$$

cf. (7.9).

7.2.7. Numerical Results for Diffusion Problems

In tables 7.1 and 7.2, numerical results for problems **(2)** and **(3)** from sections 2.2 and 2.3 are shown to examine the OS–ACMS coarse space and to compare it with AGDSW and RAGDSW. Results for the other two problems are given in tables B.3, B.10, and B.12. The scaling factor $\hat{h}_T$ of the mass matrix variant is set to the radius of the largest insphere of $T \in \tau_h(\Omega)$.

method	*tol*	it.	κ	dim V_0	($\mathcal{V}$, $\mathcal{E}$, $\mathcal{F}$, $\mathcal{S}_\Gamma$)	$\frac{\dim V_0}{\text{dof}}$
AGDSW	10^{-5}	129	$4.2 \cdot 10^5$	483	(70, 200, 213, —)	0.86%
AGDSW	0.001	49	20.1	500	(70, 215, 215, —)	0.89%
AGDSW	0.1	49	20.1	500	(70, 215, 215, —)	0.89%
RAGDSW	10^{-5}	78	$7.8 \cdot 10^4$	109	(—, —, —, 109)	0.19%
RAGDSW	0.001	56	24.2	112	(—, —, —, 112)	0.20%
RAGDSW	0.1	56	24.2	113	(—, —, —, 113)	0.20%
OS–ACMS–M	10^{-5}	113	$2.1 \cdot 10^5$	110	(70, 39, 1, —)	0.20%
OS–ACMS–M	0.001	45	16.3	115	(70, 44, 1, —)	0.21%
OS–ACMS–M	0.1	36	9.9	448	(70, 103, 275, —)	0.80%
OS–ACMS–K	10^{-5}	122	$3.0 \cdot 10^5$	109	(70, 38, 1, —)	0.19%
OS–ACMS–K	0.001	44	16.3	115	(70, 44, 1, —)	0.21%
OS–ACMS–K	0.1	39	12.5	164	(70, 60, 34, —)	0.29%
OS–ACMS–S–M	10^{-5}	113	$2.1 \cdot 10^5$	110	(70, 39, 1, —)	0.20%
OS–ACMS–S–M	0.001	43	16.3	115	(70, 44, 1, —)	0.21%
OS–ACMS–S–M	0.1	36	9.8	453	(70, 108, 275, —)	0.81%
OS–ACMS–S–K	10^{-5}	122	$3.0 \cdot 10^5$	109	(70, 38, 1, —)	0.19%
OS–ACMS–S–K	0.001	43	16.3	115	(70, 44, 1, —)	0.21%
OS–ACMS–S–K	0.1	39	12.5	165	(70, 61, 34, —)	0.29%

Table 7.1.: **(Model problem (2))** Results for the coefficient function in fig. 2.3, the diffusion problem, different methods and tolerances for the selection of eigenvectors: iteration count, condition number, resulting coarse space dimension, and coarse space dimension over the size of the stiffness matrix. The number of coarse functions associated with subdomain vertices, edges, faces, and interface stars is given in parentheses.

As in section 3.4, we consider three-dimensional diffusion problems with $f \equiv 1$ for the right-hand side of (1.1). The convergence criterion is chosen as the reduction of the relative, unpreconditioned residual by 10^{-8}; the initial vector is set to the zero vector,

method	*tol*	it.	κ	dim V_0	($\mathcal{V}$,	$\mathcal{E}$,	$\mathcal{F}$,	$\mathcal{S}_\Gamma$)	$\frac{\dim V_0}{\text{dof}}$
AGDSW	10^{-4}	549	$5.8 \cdot 10^4$	1 781	(328,	652,	801,	—)	0.30%
AGDSW	0.001	59	30.1	1 814	(328,	678,	808,	—)	0.31%
AGDSW	0.1	51	22.1	1 907	(328,	678,	901,	—)	0.32%
RAGDSW	10^{-4}	674	$4.4 \cdot 10^4$	627	(—,	—,	— ,	627)	0.11%
RAGDSW	0.001	59	33.7	660	(—,	—,	— ,	660)	0.11%
RAGDSW	0.1	53	24.4	792	(—,	—,	— ,	792)	0.13%
OS–ACMS–M	10^{-4}	147	$1.8 \cdot 10^4$	626	(328,	278,	20,	—)	0.11%
OS–ACMS–M	0.001	57	29.7	633	(328,	285,	20,	—)	0.11%
OS–ACMS–M	0.1	37	11.4	3 138	(328,	527,	2 283,	—)	0.53%
OS–ACMS–K	10^{-4}	222	$2.0 \cdot 10^4$	613	(328,	271,	14,	—)	0.10%
OS–ACMS–K	0.001	57	29.6	633	(328,	285,	20,	—)	0.11%
OS–ACMS–K	0.1	48	20.7	993	(328,	383,	282,	—)	0.17%
OS–ACMS–S–M	10^{-4}	110	$1.8 \cdot 10^4$	630	(328,	282,	20,	—)	0.11%
OS–ACMS–S–M	0.001	58	29.6	634	(328,	286,	20,	—)	0.11%
OS–ACMS–S–M	0.1	37	11.4	3 155	(328,	544,	2 283,	—)	0.54%
OS–ACMS–S–K	10^{-4}	201	$2.0 \cdot 10^4$	615	(328,	273,	14,	—)	0.10%
OS–ACMS–S–K	0.001	58	29.6	634	(328,	286,	20,	—)	0.11%
OS–ACMS–S–K	0.1	48	20.6	1 009	(328,	399,	282,	—)	0.17%

Table 7.2.: **(Model problem (3))** Results for the coefficient function in fig. 2.4, the diffusion problem, different methods and tolerances for the selection of eigenvectors: iteration count, condition number, resulting coarse space dimension, and coarse space dimension over the size of the stiffness matrix. The number of coarse functions associated with subdomain vertices, edges, faces, and interface stars is given in parentheses.

and the iteration is stopped if it does not converge within 2 000 iterations. The condition number estimate is obtained using the Lanczos method; cf. [Saa03, sect. 6.7.3].

Tables 7.1 and 7.2 both show that the OS–ACMS coarse spaces achieve similar results

method	*tol*	it.	κ	$\dim V_0$	($\mathcal{V}$, $\mathcal{E}$, $\mathcal{F}$)	$\frac{\dim V_0}{\mathrm{dof}}$
OS–ACMS–K	0.001	57	29.6	633	(328, 285, 20)	0.11%
OS–ACMS–K(3)	0.001	54	29.3	639	(328, 291, 20)	0.11%
OS–ACMS–K(1)	0.001	49	21.4	853	(328, 455, 70)	0.14%
OS–ACMS–S–K	0.001	58	29.6	634	(328, 286, 20)	0.11%
OS–ACMS–S–K(3)	0.001	54	29.5	641	(328, 293, 20)	0.11%
OS–ACMS–S–K(1)	0.001	49	21.4	853	(328, 455, 70)	0.14%

Table 7.3.: **(Model problem (3))** Results for the coefficient function in fig. 2.4, the diffusion problem, different methods, and a tolerance of 0.001 for the selection of eigenvectors: iteration count, condition number, resulting coarse space dimension, and coarse space dimension over the size of the stiffness matrix. If the slab variant is used, the slab width in layers of finite elements is appended in parentheses to the method's name. The number of coarse functions associated with subdomain vertices, edges, and faces is given in parentheses.

to RAGDSW despite the use of the GDSW interface partition. This is promising as it indicates that we can decrease the coarse space dimension considerably by enforcing additional Dirichlet conditions in the energy-minimizing extensions.

The differences between the variants are negligible for the relevant tolerance of 0.001. As a result, in practice, we would choose OS–ACMS–S–K for its superior parallelizability, and since a mass matrix does not have to be assembled.

As for AGDSW and RAGDSW, in table 7.3, an increase in the coarse space dimension is observed in case a minimal slab of only one layer of finite elements is used. However, as was motivated in section 4.5, the increase is less substantial than for RAGDSW, since OS–ACMS uses smaller eigenvalue problems. Results for model problems **(1)** and **(2)** are given in tables B.6 and B.8.

7.3. Reduced-Dimension OS–ACMS

Similarly to the R–WB–AGDSW coarse space, we can replace the edge and vertex functions of OS–ACMS with wire basket functions. To this end, we keep the face problems of OS–ACMS and use the wire basket eigenvalue problems of the R–WB–AGDSW coarse space. As a result, unlike for the edge problems of OS–ACMS, we do not enforce a zero Dirichlet boundary condition on the boundary of a wire basket star. We denote the resulting coarse space by R–WB–OS–ACMS. Note that, in two dimensions, R–WB–OS–ACMS is identical to R–WB–AGDSW.

The extension of face functions is carried out identically to OS–ACMS. However, for the extension of the wire basket functions, we need to proceed as for the edge functions of OS–ACMS. For a star $s \in \mathcal{S}_{\mathcal{W}}$, we set $\partial s := \emptyset$, such that $\overline{s} = s$. Furthermore, we define ∂f as the set of all interface nodes that are adjacent to f. For $f \in \mathcal{F}$, let $\mathcal{B}_{\mathrm{p}}(f) \subset \mathcal{S}_{\mathcal{W}}$ denote the parents of f; that is, the set of wire basket stars that are adjacent to f. By $\mathcal{B}_{\mathrm{c}}(s)$ we denote the children of $s \in \mathcal{S}_{\mathcal{W}}$; that is, the set of faces that are adjacent to s. Let $\tau_{*,s} \in X^h(s)$ be an eigenfunction obtained by the corresponding generalized eigenvalue problem. We define an extension from s to $\overline{f}$, $f \in \mathcal{B}_{\mathrm{c}}(s)$, as

$$\tau_{*,s \to \overline{f}} := \mathcal{H}_{\partial f \to \Omega_f}\big(z_s(\tau_{*,s})\big)\big|_{\overline{f}};$$

cf. (7.4). Let

$$\tau_{*,s \to \Gamma^h} := \begin{cases} \tau_{*,s \to \overline{f}} & \text{on } f \in \mathcal{B}_{\mathrm{c}}(s), \\ \tau_{*,s} & \text{on } s, \\ 0 & \text{elsewhere on } \Gamma^h. \end{cases}$$

We can now define the coarse function associated with $\tau_{*,s}$ as

$$v_{*,s} := \mathcal{H}_{\Gamma^h \to \Omega}(\tau_{*,s \to \Gamma^h}).$$

We denote the new coarse space by R–WB–OS–ACMS or R–WB–OS–ACMS–M. The same variants as for OS–ACMS can be used for R–WB–OS–ACMS.

Remark 7.3. *If we remove the face problems and replace the wire basket problems*

with problems on interface stars, we obtain the RAGDSW coarse space. However, the OS–ACMS framework allows for more general types of coarse spaces that are based on the interface partition $\mathcal{P} = \mathcal{S}_\Gamma$; see section 7.5.

7.3.1. Numerical Results for Diffusion Problems

In tables 7.4 and 7.5, numerical results for problems **(2)** and **(3)** from sections 2.2 and 2.3 are shown to examine the R–WB–OS–ACMS coarse space and to compare it with OS–ACMS and RAGDSW. Results for the other two problems are given in tables B.4, B.11, and B.12. The scaling factor $\hat{h}_T$ of the mass matrix variant is set to the radius of the largest insphere of $T \in \tau_h(\Omega)$.

As in section 3.4, we consider three-dimensional diffusion problems with $f \equiv 1$ for the right-hand side of (1.1). The convergence criterion is chosen as the reduction of the relative, unpreconditioned residual by 10^{-8}; the initial vector is set to the zero vector, and the iteration is stopped if it does not converge within $2\,000$ iterations. The condition number estimate is obtained using the Lanczos method; cf. [Saa03, sect. 6.7.3].

In table 7.4, we can observe a reduction of the coarse space dimension of slightly below 20% for a tolerance of 0.001, using R–WB–OS–ACMS instead of OS–ACMS. The results in table 7.5 show a more pronounced reduction of 35%. This confirms our expectation that replacing the vertex and edge functions by functions obtained from eigenvalue problems on wire basket stars should reduce the coarse space dimension. As in sections 3.4, 4.5, and 7.2.7, using the mass variant does not give an advantage. Furthermore, the differences to the S-variants are negligible.

As R–WB–AGDSW and R–WB–OS–ACMS use the same interface partition, the comparison of their results highlights the relevance of the inclusion of additional Dirichlet conditions in the energy-minimizing extensions used in the generalized eigenvalue problems. With respect to R–WB–AGDSW, we obtain a reduction in the coarse space dimension of almost 70% for problem **(2)** and of 65% for problem **(3)**.

In table 7.6, results for the slab variant are shown, confirming our previous results in sections 3.4, 4.5, and 7.2.7. In table 7.7, we observe almost no differences in the results for

method	*tol*	it.	κ	dim V_0	($\mathcal{V}$, $\mathcal{E}$, $\mathcal{F}$, $\mathcal{S}$)	$\frac{\dim V_0}{\text{dof}}$
R–WB–AGDSW	0.001	53	21.5	308	(—, —, 215, 93)	0.55%
RAGDSW	10^{-5}	78	$7.8 \cdot 10^5$	109	(—, —, —, 109)	0.19%
RAGDSW	0.001	56	24.2	112	(—, —, —, 112)	0.20%
RAGDSW	0.1	56	24.2	113	(—, —, —, 113)	0.20%
OS–ACMS–K	10^{-5}	122	$3.0 \cdot 10^5$	109	(70, 38, 1, —)	0.19%
OS–ACMS–K	0.001	44	16.3	115	(70, 44, 1, —)	0.21%
OS–ACMS–K	0.1	39	12.5	164	(70, 60, 34, —)	0.29%
R–WB–OS–ACMS–M	10^{-5}	114	$2.4 \cdot 10^5$	91	(—, —, 1, 90)	0.16%
R–WB–OS–ACMS–M	0.001	49	15.7	94	(—, —, 1, 93)	0.17%
R–WB–OS–ACMS–M	0.1	42	13.2	368	(—, —, 275, 93)	0.66%
R–WB–OS–ACMS–K	10^{-5}	128	$6.4 \cdot 10^5$	87	(—, —, 1, 86)	0.16%
R–WB–OS–ACMS–K	0.001	49	15.7	94	(—, —, 1, 93)	0.17%
R–WB–OS–ACMS–K	0.1	47	15.5	127	(—, —, 34, 93)	0.23%
R–WB–OS–ACMS–S–M	10^{-5}	104	$2.3 \cdot 10^5$	92	(—, —, 1, 91)	0.16%
R–WB–OS–ACMS–S–M	0.001	49	15.7	94	(—, —, 1, 93)	0.17%
R–WB–OS–ACMS–S–M	0.1	40	12.7	372	(—, —, 275, 97)	0.66%
R–WB–OS–ACMS–S–K	10^{-5}	127	$4.2 \cdot 10^5$	89	(—, —, 1, 88)	0.16%
R–WB–OS–ACMS–S–K	0.001	49	15.7	94	(—, —, 1, 93)	0.17%
R–WB–OS–ACMS–S–K	0.1	47	15.5	127	(—, —, 34, 93)	0.23%

Table 7.4.: **(Model problem (2))** Results for the coefficient function in fig. 2.3, the diffusion problem, different methods and tolerances for the selection of eigenvectors: iteration count, condition number, resulting coarse space dimension, and coarse space dimension over the size of the stiffness matrix. The number of coarse functions associated with subdomain vertices, edges, faces, wire basket and interface stars is given in parentheses. $\mathcal{S}$ refers to $\mathcal{S}_{\mathcal{W}}/\mathcal{S}_\Gamma$.

the combination of the lumped and S-variant of R–WB–OS–ACMS. This combination of variants is appealing to be used in practice because of its simplicity and parallel efficiency.

method	tol	it.	κ	$\dim V_0$	($\mathcal{V}$,	$\mathcal{E}$,	$\mathcal{F}$,	$\mathcal{S}$)	$\frac{\dim V_0}{\mathrm{dof}}$
R–WB–AGDSW	0.001	57	30.2	1 200	(—,	—,	808,	392)	0.20%
RAGDSW	10^{-4}	674	$4.4{\cdot}10^4$	627	(—,	—,	— ,	627)	0.11%
RAGDSW	0.001	59	33.7	660	(—,	—,	— ,	660)	0.11%
RAGDSW	0.1	53	24.4	792	(—,	—,	— ,	792)	0.13%
OS–ACMS–K	10^{-4}	222	$2.0{\cdot}10^4$	613	(328,	271,	14,	—)	0.10%
OS–ACMS–K	0.001	57	29.6	633	(328,	285,	20,	—)	0.11%
OS–ACMS–K	0.1	48	20.7	993	(328,	383,	282,	—)	0.17%
R–WB–OS–ACMS–M	10^{-4}	178	$1.6{\cdot}10^4$	405	(—,	—,	20,	385)	0.07%
R–WB–OS–ACMS–M	0.001	55	25.8	412	(—,	—,	20,	392)	0.07%
R–WB–OS–ACMS–M	0.1	38	12.0	2 694	(—,	—,	2 283,	411)	0.46%
R–WB–OS–ACMS–K	10^{-4}	521	$3.6{\cdot}10^4$	388	(—,	—,	14,	374)	0.07%
R–WB–OS–ACMS–K	0.001	55	25.7	412	(—,	—,	20,	392)	0.07%
R–WB–OS–ACMS–K	0.1	48	20.4	674	(—,	—,	282,	392)	0.11%
R–WB–OS–ACMS–S–M	10^{-4}	119	$1.6{\cdot}10^4$	409	(—,	—,	20,	389)	0.07%
R–WB–OS–ACMS–S–M	0.001	55	25.8	413	(—,	—,	20,	393)	0.07%
R–WB–OS–ACMS–S–M	0.1	36	10.3	2 752	(—,	—,	2 283,	469)	0.47%
R–WB–OS–ACMS–S–K	10^{-4}	370	$3.1{\cdot}10^4$	396	(—,	—,	14,	382)	0.07%
R–WB–OS–ACMS–S–K	0.001	55	25.7	413	(—,	—,	20,	393)	0.07%
R–WB–OS–ACMS–S–K	0.1	49	20.4	677	(—,	—,	282,	395)	0.11%

Table 7.5.: **(Model problem (3))** Results for the coefficient function in fig. 2.4, the diffusion problem, different methods and tolerances for the selection of eigenvectors: iteration count, condition number, resulting coarse space dimension, and coarse space dimension over the size of the stiffness matrix. The number of coarse functions associated with subdomain vertices, edges, faces, wire basket and interface stars is given in parentheses. $\mathcal{S}$ refers to $\mathcal{S}_{\mathcal{W}}/\mathcal{S}_{\Gamma}$.

We remark, however, that for other types of problems, a shift of the spectrum may be observed if lumped variants are used; cf. tables B.12, B.13, B.15, and B.17.

method	*tol*	it.	κ	$\dim V_0$	$(\mathcal{F}, \mathcal{S_W})$	$\frac{\dim V_0}{\text{dof}}$
R–WB–OS–ACMS–K	0.001	55	25.7	412	(20, 392)	0.07%
R–WB–OS–ACMS–K(3)	0.001	51	25.3	424	(20, 404)	0.07%
R–WB–OS–ACMS–K(1)	0.001	47	19.5	767	(70, 697)	0.13%
R–WB–OS–ACMS–S–K	0.001	55	25.7	413	(20, 393)	0.07%
R–WB–OS–ACMS–S–K(3)	0.001	51	24.4	425	(20, 405)	0.07%
R–WB–OS–ACMS–S–K(1)	0.001	47	19.5	767	(70, 697)	0.13%

Table 7.6.: **(Model problem (3))** Results for the coefficient function in fig. 2.4, the diffusion problem, different methods, and a tolerance of 0.001 for the selection of eigenvectors: iteration count, condition number, resulting coarse space dimension, and coarse space dimension over the size of the stiffness matrix. If the slab variant is used, the slab width in layers of finite elements is appended in parentheses to the method's name. The number of coarse functions associated with subdomain faces and wire basket stars is given in parentheses.

7.4. Condition Number Bound for OS–ACMS

In the following, we prove a condition number bound for OS–ACMS. Although we will restrict ourselves to three dimensions, the two-dimensional case is proved analogously; along with a condition number bound for the three-dimensional case, a bound for the two-dimensional case will be stated in theorem 7.1. The following proof will serve as a blueprint for a much more general type of proof in section 7.5 that encompasses fairly general types of coarse spaces, even GDSW-type coarse spaces.

For each $\xi \in \mathcal{P}$, let the symmetric, bilinear forms $\beta_\xi(\cdot, \cdot)$ and $\overline{\alpha}_\xi(\cdot, \cdot)$ on $X^h(\xi) \times X^h(\xi)$ be given, where $\beta_\xi(\cdot, \cdot)$ is positive definite, and $\overline{\alpha}_\xi(\cdot, \cdot)$ is positive semidefinite. In the case of standard OS–ACMS, we set $\beta_\xi := \beta_\xi^M$ and $\overline{\alpha}_\xi := \overline{\alpha}_\xi^K$. For the S-variant, we replace $\overline{\alpha}_\xi$ with $\overline{\alpha}_\xi := \overline{\alpha}_\xi^S$. If a stiffness matrix is used on the right-hand side of the generalized eigenvalue problem, we replace β_ξ with $\beta_\xi := \beta_\xi^K$. Similarly, the slab variant uses either $\overline{\alpha}_\xi := \overline{\alpha}_\xi^{K,l}$ or $\overline{\alpha}_\xi := \overline{\alpha}_\xi^{S,l}$. Accordingly, $\alpha_{\overline{\xi}}$ is given by $\alpha_{\overline{\xi}}^K$, $\alpha_{\overline{\xi}}^S$, $\alpha_{\overline{\xi}}^{K,l}$, or $\alpha_{\overline{\xi}}^{S,l}$. For the

E	method	*tol*	it.	κ	dim V_0	($\mathcal{F}$, $\mathcal{S}_{\mathcal{W}}$)	$\frac{\dim V_0}{\text{dof}}$
(1)	R–WB–OS–ACMS–S–K	0.001	55	27.1	769	(197, 572)	0.58%
	R–WB–OS–ACMS–S–ℓ(K)	0.001	54	27.1	769	(197, 572)	0.58%
(2)	R–WB–OS–ACMS–S–K	0.001	49	15.7	94	(1, 93)	0.17%
	R–WB–OS–ACMS–S–ℓ(K)	0.001	49	15.7	94	(1, 93)	0.17%
(3)	R–WB–OS–ACMS–S–K	0.001	55	25.7	413	(20, 393)	0.07%
	R–WB–OS–ACMS–S–ℓ(K)	0.001	55	25.7	413	(20, 393)	0.07%

Table 7.7.: Results for the coefficient functions **(1)**–**(3)** in figs. 2.2 to 2.4, the diffusion problem, different methods, and a tolerance of 0.001 for the selection of eigenvectors: iteration count, condition number, resulting coarse space dimension, and coarse space dimension over the size of the stiffness matrix. If a lumped matrix is used, $\ell(K)$ or $\ell(M)$ is appended to the method's name. The number of coarse functions associated with subdomain faces and wire basket stars is given in parentheses.

lumped variants, we refer to remarks 6.3 and 6.6.

As in lemma 6.1, we define the seminorms

$$|u|_{\alpha_{\overline{\xi}}} := \sqrt{\alpha_{\overline{\xi}}(u, u)}, \quad |v|_{\overline{\alpha}_{\xi}} := \sqrt{\overline{\alpha}_{\xi}(v, v)}, \quad u \in X^h(\overline{\xi}),\ v \in X^h(\xi),$$

and the norm

$$\|u\|_{\beta_{\xi}} := \sqrt{\beta_{\xi}(u, u)}, \quad u \in X^h(\xi).$$

By lemma 7.2, we have

$$|u|^2_{\alpha_{\overline{\xi}}} = \alpha_{\overline{\xi}}(u, u) \le \alpha^K_{\overline{\xi}}(u, u) = |u|^2_{\alpha^K_{\overline{\xi}}} = |\mathcal{H}_{\overline{\xi} \to \Omega_{\xi}}(u)|^2_{a(\Omega_{\xi})} \quad \forall u \in V^h(\Omega). \tag{7.17}$$

For $q \subset \overline{\xi}$, we define the seminorm

$$|u|_{a, q \to \Omega_{\xi}} := |\mathcal{H}_{q \to \Omega_{\xi}}(u)|_{a(\Omega_{\xi})} \quad \forall u \in V^h(\Omega), \tag{7.18}$$

for which follows that

$$|u|_{a, q \to \Omega_{\xi}} \le |u|_{a(\Omega_{\xi})}. \tag{7.19}$$

As per lemma 6.1, we choose the eigenfunctions such that

$$\beta_\xi(\,v_{k,\xi}\,,\,v_{l,\xi}\,) = \delta_{k,l},$$

where $\delta_{k,l}$ is the Kronecker delta, and $v_{k,\xi}$ are the coarse functions. Let $u \in V^h(\Omega)$. We define the spectral projections

$$\Pi_{\mathcal{V}} u := \sum_{\nu \in \mathcal{V}} \Pi_\nu u, \qquad \Pi_\nu u := \sum_{\lambda_{k,\nu} \leq tol_\nu} \beta_\nu(\,u\,,\,v_{k,\nu}\,) v_{k,\nu},$$

$$\Pi_{\mathcal{E}} u := \sum_{e \in \mathcal{E}} \Pi_e u, \qquad \Pi_e u := \sum_{\lambda_{k,e} \leq tol_e} \beta_e(\,u\,,\,v_{k,e}\,) v_{k,e},$$

$$\Pi_{\mathcal{F}} u := \sum_{f \in \mathcal{F}} \Pi_f u, \qquad \Pi_f u := \sum_{\lambda_{k,f} \leq tol_f} \beta_f(\,u\,,\,v_{k,f}\,) v_{k,f}.$$

The tolerances $tol_e > 0$ and $tol_f > 0$ are user-prescribed. Since all MsFEM-type vertex functions are included in the OS–ACMS coarse space, we set $tol_\nu = \infty$.

We now define the coarse component

$$u_0 := \Pi_{\mathcal{V}} u + \Pi_{\mathcal{E}} \bar{u}_{\mathcal{V}} + \Pi_{\mathcal{F}} \bar{u}_{\mathcal{E}},$$

where

$$\bar{u}_{\mathcal{V}} := u - \Pi_{\mathcal{V}} u, \qquad\qquad \bar{u}_{\mathcal{E}} := \bar{u}_{\mathcal{V}} - \Pi_{\mathcal{E}} \bar{u}_{\mathcal{V}}.$$

The definition reflects the cascaded energy-minimizing extensions; cf. fig. 7.3. For the remainder $u - u_0$, we obtain

$$u - u_0 = u - \Pi_{\mathcal{V}} u - \Pi_{\mathcal{E}} \bar{u}_{\mathcal{V}} - \Pi_{\mathcal{F}} \bar{u}_{\mathcal{E}}$$

$$= \bar{u}_{\mathcal{V}} - \Pi_{\mathcal{E}} \bar{u}_{\mathcal{V}} - \Pi_{\mathcal{F}} \bar{u}_{\mathcal{E}}$$

$$= \bar{u}_{\mathcal{E}} - \Pi_{\mathcal{F}} \bar{u}_{\mathcal{E}}.$$

We can reuse much of the proof of the RAGDSW condition number bound but need to replace lemmas 6.2 and 6.3.

Lemma 7.4 (Vertex Contribution)**.** *Let $\nu \in \mathcal{V}$. Then we have*

$$\|u - u_0\|_{\beta_\nu^K}^2 = 0$$

for $u \in V^h(\Omega)$.

We will henceforth refer to this property as the exact interpolation of u in the vertices, since it follows from lemma 7.4 that $u(\nu) = u_0(\nu)$.

Proof. We show that $u - u_0$ is zero in ν. Then, by definition of $\| \cdot \|_{\beta_\nu^K}$, the proposition follows.

By assumption, we have $b_\nu(v_{i,\nu}, v_{j,\nu}) = \delta_{i,j}$. There exist constants c_j such that $u(\nu) = \sum_{j=1}^{\hat{d}} c_j v_{j,\nu}(\nu)$. Since $\Pi_\mathcal{E} u$ and $\Pi_\mathcal{F} u$ are zero at the vertices and $\Pi_{\tilde\nu} u$ is zero at $\nu \neq \tilde\nu \in \mathcal{V}$, we have $u_0|_\nu = \Pi_\nu u|_\nu$. We obtain

$$u_0(\nu) = \sum_{k=1}^{\hat{d}} v_{k,\nu}(\nu) b_\nu(u, v_{k,\nu}) = \sum_{k=1}^{\hat{d}} v_{k,\nu}(\nu) \sum_{j=1}^{\hat{d}} c_j b_\nu(v_{j,\nu}, v_{k,\nu}) = \sum_{k=1}^{\hat{d}} v_{k,\nu}(\nu) c_k = u(\nu).$$

Thus, we have $u(\nu) - u_0(\nu) = 0$. $\qquad\square$

Lemma 7.5 (Edge Contribution). *Let $e \in \mathcal{E}$. Then we have*

$$\|u - u_0\|_{\beta_e^K}^2 = |z_e(\overline{u}_\mathcal{E})|_{a(\Omega_e)}^2 \leq \frac{4 C_{\mathrm{inv}}}{tol_e} |u|_{a(\Omega_e)}^2$$

for $u \in V^h(\Omega)$.

Proof. The following proof is also valid for the special case $\overline{e} = e$, in which e does not have any adjacent coarse nodes. In that case, however, the proof could be simplified significantly, and the bound could be improved.

Unlike for the adaptive GDSW-type coarse spaces, we cannot exploit the fact anymore that all coarse functions associated with interface components other than e are zero on $\overline{e}$. In a first step, however, we can use that the face functions are zero on edges, and that edge functions of other edges are zero on $\overline{e}$ as well.

$$\begin{aligned}
\|u - u_0\|_{\beta_e^K}^2 &= |z_e(\overline{u}_\mathcal{E} - \Pi_\mathcal{F}\overline{u}_\mathcal{E})|_{a(\Omega_e)}^2 \\
&= |z_e(\overline{u}_\mathcal{E})|_{a(\Omega_e)}^2 \\
&= |z_e(\overline{u}_\mathcal{V} - \Pi_\mathcal{E}\overline{u}_\mathcal{V})|_{a(\Omega_e)}^2 \\
&= |z_e(\overline{u}_\mathcal{V} - \Pi_e\overline{u}_\mathcal{V})|_{a(\Omega_e)}^2 \\
&= \|\overline{u}_\mathcal{V} - \Pi_e\overline{u}_\mathcal{V}\|_{\beta_e^K}^2.
\end{aligned}$$

Using (6.38) and lemma 6.1, it follows that

$$
\begin{aligned}
\|\overline{u}_{\mathcal{V}} - \Pi_e \overline{u}_{\mathcal{V}}\|_{\beta_e^K}^2 &\leq C_{\mathrm{inv}} \|\overline{u}_{\mathcal{V}} - \Pi_e \overline{u}_{\mathcal{V}}\|_{\beta_e}^2 \\
&\leq \frac{C_{\mathrm{inv}}}{tol_e} |\overline{u}_{\mathcal{V}} - \Pi_e \overline{u}_{\mathcal{V}}|_{\overline{\alpha}_e}^2 \\
&\leq \frac{C_{\mathrm{inv}}}{tol_e} |\overline{u}_{\mathcal{V}}|_{\overline{\alpha}_e}^2 \\
&= \frac{C_{\mathrm{inv}}}{tol_e} |z_e(\overline{u}_{\mathcal{V}})|_{\alpha_{\overline{e}}}^2 \\
&= \frac{C_{\mathrm{inv}}}{tol_e} |z_e(u - \Pi_{\mathcal{V}} u)|_{\alpha_{\overline{e}}}^2.
\end{aligned}
$$

Since $\Pi_{\mathcal{V}}$ interpolates u exactly in the vertices (cf. lemma 7.4), we obtain

$$
\begin{aligned}
\frac{C_{\mathrm{inv}}}{tol_e} |z_e(u - \Pi_{\mathcal{V}} u)|_{\alpha_{\overline{e}}}^2 &= \frac{C_{\mathrm{inv}}}{tol_e} |u - \Pi_{\mathcal{V}} u|_{\alpha_{\overline{e}}}^2 \\
&\leq \frac{C_{\mathrm{inv}}}{tol_e} \left(|u|_{\alpha_{\overline{e}}} + |\Pi_{\mathcal{V}} u|_{\alpha_{\overline{e}}} \right)^2.
\end{aligned}
$$

From (7.17) and (7.19) follows that

$$
|u|_{\alpha_{\overline{e}}} \leq |u|_{a, \overline{e} \to \Omega_e} \leq |u|_{a(\Omega_e)}.
$$

By definition of the vertex functions, the edge values are energy minimizing. Thus, it follows from (7.16) (cf. lemma 7.1) that

$$
|\Pi_{\mathcal{V}} u|_{\alpha_{\overline{e}}} \leq |\Pi_{\mathcal{V}} u|_{a, \partial e \to \Omega_e}.
$$

Since $\Pi_{\mathcal{V}}$ interpolates u exactly in the vertices (cf. lemma 7.4), we have

$$
|\Pi_{\mathcal{V}} u|_{a, \partial e \to \Omega_e} = |u|_{a, \partial e \to \Omega_e} \leq |u|_{a(\Omega_e)}.
$$

In total, we have

$$
\|u - u_0\|_{\beta_e^K}^2 \leq \frac{4 C_{\mathrm{inv}}}{tol_e} |u|_{a(\Omega_e)}^2.
$$

$\square$

For a bound on the contribution of face problems, we need to invest more work because of the cascaded extension from vertices to edges to faces of the MsFEM-type functions.

Lemma 7.6 (Face Contribution)**.** *Let $f \in \mathcal{F}$. Then we have*

$$\|u - u_0\|_{\beta_f^K}^2 \leq \frac{8 C_{\mathrm{inv}}}{tol_f} \left(|u|_{a(\Omega_f)}^2 + \sum_{e \in \mathcal{E} \cap \mathcal{B}_{\mathrm{p}}(f)} \frac{4 C_{\mathrm{inv}} C_\tau}{tol_e} |u|_{a(\Omega_e)}^2 \right)$$

for $u \in V^h(\Omega)$, where $\mathcal{E} \cap \mathcal{B}_{\mathrm{p}}(f)$ is the set of edges that are adjacent to f.

Proof. The following proof is also valid if f does not have adjacent coarse nodes or edges. In that case, however, the proof could be simplified significantly and the bound improved.

In a first step, we can use the fact that face functions of faces other than f are zero on $\overline{f}$.

$$\|u - u_0\|_{\beta_f^K}^2 = |z_f(\overline{u}_\mathcal{E} - \Pi_\mathcal{F} \overline{u}_\mathcal{E})|_{a(\Omega_f)}^2$$

$$= |z_f(\overline{u}_\mathcal{E} - \Pi_f \overline{u}_\mathcal{E})|_{a(\Omega_f)}^2$$

$$= \|\overline{u}_\mathcal{E} - \Pi_f \overline{u}_\mathcal{E}\|_{\beta_f^K}^2.$$

Using (6.38) and lemma 6.1, it follows that

$$\|\overline{u}_\mathcal{E} - \Pi_f \overline{u}_\mathcal{E}\|_{\beta_f^K}^2 \leq C_{\mathrm{inv}} \|\overline{u}_\mathcal{E} - \Pi_f \overline{u}_\mathcal{E}\|_{\beta_f}^2$$

$$\leq \frac{C_{\mathrm{inv}}}{tol_f} |\overline{u}_\mathcal{E} - \Pi_f \overline{u}_\mathcal{E}|_{\alpha_f}^2$$

$$\leq \frac{C_{\mathrm{inv}}}{tol_f} |\overline{u}_\mathcal{E}|_{\alpha_f}^2$$

$$= \frac{C_{\mathrm{inv}}}{tol_f} |z_f(\overline{u}_\mathcal{E})|_{\alpha_{\overline{f}}}^2.$$

We have

$$z_f(\overline{u}_\mathcal{E}) = z_{\overline{f}}(\overline{u}_\mathcal{E}) - z_{\partial f}(\overline{u}_\mathcal{E}) = \overline{u}_\mathcal{E} - z_{\partial f}(\overline{u}_\mathcal{E}) \quad \text{on } \overline{f}.$$

Since $\overline{u}_\mathcal{E} = \overline{u}_\mathcal{V} - \Pi_\mathcal{E} \overline{u}_\mathcal{V} = u - \Pi_\mathcal{V} u - \Pi_\mathcal{E} \overline{u}_\mathcal{V}$ is zero in the vertices, we infer

$$z_f(\overline{u}_\mathcal{E}) = \overline{u}_\mathcal{E} - \sum_{e \in \mathcal{E} \cap \mathcal{B}_{\mathrm{p}}(f)} z_e(\overline{u}_\mathcal{E}) \quad \text{on } \overline{f},$$

where $\mathcal{E} \cap \mathcal{B}_{\mathrm{p}}(f)$ are all coarse edges adjacent to f. We obtain

$$|z_f(\overline{u}_\mathcal{E})|_{\alpha_{\overline{f}}} \leq |\overline{u}_\mathcal{E}|_{\alpha_{\overline{f}}} + \left| \sum_{e \in \mathcal{E} \cap \mathcal{B}_{\mathrm{p}}(f)} z_e(\overline{u}_\mathcal{E}) \right|_{\alpha_{\overline{f}}}. \tag{7.20}$$

For the first term on the right-hand side of (7.20), we have

$$|\overline{u}_{\mathcal{E}}|_{\alpha_{\overline{f}}} = |u - \Pi_{\mathcal{V}}u - \Pi_{\mathcal{E}}(u - \Pi_{\mathcal{V}}u)|_{\alpha_{\overline{f}}}$$

$$\leq |u|_{\alpha_{\overline{f}}} + |\Pi_{\mathcal{V}}u + \Pi_{\mathcal{E}}(u - \Pi_{\mathcal{V}}u)|_{\alpha_{\overline{f}}}$$

$$\leq |u|_{a(\Omega_f)} + |\Pi_{\mathcal{V}}u + \Pi_{\mathcal{E}}(u - \Pi_{\mathcal{V}}u)|_{\alpha_{\overline{f}}},$$

where in the last step we have used (7.17) and (7.19). Let

$$w := \Pi_{\mathcal{V}}u + \Pi_{\mathcal{E}}(u - \Pi_{\mathcal{V}}u).$$

Using (7.16) (cf. lemma 7.1), it follows that

$$|w|_{\alpha_{\overline{f}}} \leq |w|_{a,\partial f \to \Omega_f},$$

and

$$|w|_{a,\partial f \to \Omega_f} \leq |u|_{a,\partial f \to \Omega_f} + |u - w|_{a,\partial f \to \Omega_f}$$

$$\leq |u|_{a(\Omega_f)} + |\overline{u}_{\mathcal{E}}|_{a,\partial f \to \Omega_f}.$$

Since $\overline{u}_{\mathcal{E}} = 0$ in the vertices, it follows that

$$|\overline{u}_{\mathcal{E}}|_{a,\partial f \to \Omega_f} = \Big| \sum_{e \in \mathcal{E} \cap \mathcal{B}_{\mathrm{p}}(f)} z_e(\overline{u}_{\mathcal{E}}) \Big|_{a,\partial f \to \Omega_f} \leq \Big| \sum_{e \in \mathcal{E} \cap \mathcal{B}_{\mathrm{p}}(f)} z_e(\overline{u}_{\mathcal{E}}) \Big|_{a(\Omega_f)}.$$

As before, we can use (7.17) and (7.19) to obtain

$$\Big| \sum_{e \in \mathcal{E} \cap \mathcal{B}_{\mathrm{p}}(f)} z_e(\overline{u}_{\mathcal{E}}) \Big|_{\alpha_{\overline{f}}} \leq \Big| \sum_{e \in \mathcal{E} \cap \mathcal{B}_{\mathrm{p}}(f)} z_e(\overline{u}_{\mathcal{E}}) \Big|_{a,\overline{f} \to \Omega_f} \leq \Big| \sum_{e \in \mathcal{E} \cap \mathcal{B}_{\mathrm{p}}(f)} z_e(\overline{u}_{\mathcal{E}}) \Big|_{a(\Omega_f)}$$

for the right term on the right-hand side of (7.20). Similarly to the line of proof in lemma 6.4 and by using a Cauchy–Schwarz inequality, it follows that

$$\Big| \sum_{e \in \mathcal{E} \cap \mathcal{B}_{\mathrm{p}}(f)} z_e(\overline{u}_{\mathcal{E}}) \Big|_{a(\Omega_f)}^2 = \sum_{T \in \tau_h(\Omega_f)} \Big| \sum_{e \in \mathcal{E} \cap \mathcal{B}_{\mathrm{p}}(f)} z_e(\overline{u}_{\mathcal{E}}) \Big|_{a(T)}^2$$

$$\leq C_\tau \sum_{T \in \tau_h(\Omega_f)} \sum_{e \in \mathcal{E} \cap \mathcal{B}_{\mathrm{p}}(f)} |z_e(\overline{u}_{\mathcal{E}})|_{a(T)}^2$$

$$= C_\tau \sum_{e \in \mathcal{E} \cap \mathcal{B}_{\mathrm{p}}(f)} |z_e(\overline{u}_{\mathcal{E}})|_{a(\Omega_f)}^2$$

$$\leq C_\tau \sum_{e \in \mathcal{E} \cap \mathcal{B}_{\mathrm{p}}(f)} |z_e(\overline{u}_{\mathcal{E}})|_{a(\Omega_e)}^2.$$

Using lemma 7.5, we then obtain for (7.20)

$$|z_f(\overline{u}_\mathcal{E})|^2_{\alpha_{\overline{f}}} \leq \left(2|u|_{a(\Omega_f)} + 2\Big| \sum_{e \in \mathcal{E} \cap \mathcal{B}_{\mathrm{p}}(f)} z_e(\overline{u}_\mathcal{E}) \Big|_{a(\Omega_f)} \right)^2$$

$$\leq 8|u|^2_{a(\Omega_f)} + 8C_\tau \sum_{e \in \mathcal{E} \cap \mathcal{B}_{\mathrm{p}}(f)} |z_e(\overline{u}_\mathcal{E})|^2_{a(\Omega_e)}$$

$$\leq 8|u|^2_{a(\Omega_f)} + 8C_\tau \sum_{e \in \mathcal{E} \cap \mathcal{B}_{\mathrm{p}}(f)} \frac{4C_{\mathrm{inv}}}{tol_e} |u|^2_{a(\Omega_e)}.$$

In total, we have

$$\|u - u_0\|^2_{\beta_f^K} \leq \frac{C_{\mathrm{inv}}}{tol_f} |z_f(\overline{u}_\mathcal{E})|^2_{\alpha_{\overline{f}}} \leq \frac{8C_{\mathrm{inv}}}{tol_f} \left(|u|^2_{a(\Omega_f)} + \sum_{e \in \mathcal{E} \cap \mathcal{B}_{\mathrm{p}}(f)} \frac{4C_{\mathrm{inv}}C_\tau}{tol_e} |u|^2_{a(\Omega_e)} \right).$$

$\square$

We will formulate analogues of corollaries 6.1 and 6.2 for OS–ACMS. For brevity, we will not give a bound that depends on the individual tolerances tol_e and tol_f but that uses $tol_\mathcal{E} = \min_{e \in \mathcal{E}} tol_e$ and $tol_\mathcal{F} = \min_{f \in \mathcal{F}} tol_f$. Similarly to (6.34), let

$$N^e := \max_{1 \leq i \leq N} |\mathcal{E}(\Omega_i)|, \quad \mathcal{E}(\Omega_i) := \{ e \in \mathcal{E} : e \cap \overline{\Omega}_i \neq \emptyset \}, \tag{7.21}$$

$$N^f := \max_{1 \leq i \leq N} |\mathcal{F}(\Omega_i)|, \quad \mathcal{F}(\Omega_i) := \{ f \in \mathcal{F} : f \cap \overline{\Omega}_i \neq \emptyset \}, \tag{7.22}$$

be the maximum number of coarse edges and faces of a subdomain, respectively. Furthermore, we define

$$N^{\partial f \to e} := \max_{1 \leq i \leq N} \sum_{f \in \mathcal{F}(\Omega_i)} |\mathcal{E} \cap \mathcal{B}_{\mathrm{p}}(f)| \leq \max_{1 \leq i \leq N} |\mathcal{F}(\Omega_i)| \cdot |\mathcal{E}(\Omega_i)| \leq N^f N^e, \tag{7.23}$$

where $\mathcal{E} \cap \mathcal{B}_{\mathrm{p}}(f)$ is the set of coarse edges adjacent to a coarse face $f \in \mathcal{F}$.

Corollary 7.1. *(Analogue of corollary 6.1) It holds that*

$$|u_0|_{a(\Omega)} \leq \left(1 + \sqrt{4C_{\mathrm{inv}}C_\tau \Big(\frac{N^e}{tol_\mathcal{E}} + \frac{2N^f}{tol_\mathcal{F}} + \frac{8C_{\mathrm{inv}}C_\tau N^{\partial f \to e}}{tol_\mathcal{F} tol_\mathcal{E}} \Big)} \right) |u|_{a(\Omega)};$$

see (6.33) and (6.38) for the definitions of C_τ and C_{inv}.

Proof. Using lemmas 7.4 to 7.6, we have

$$
\begin{aligned}
C_\tau \sum_{\xi \in \mathcal{P}} \|u - u_0\|_{\beta_\xi^K}^2 &= C_\tau \left(\sum_{\nu \in \mathcal{V}} \|u - u_0\|_{\beta_\nu^K}^2 + \sum_{e \in \mathcal{E}} \|u - u_0\|_{\beta_e^K}^2 + \sum_{f \in \mathcal{F}} \|u - u_0\|_{\beta_f^K}^2 \right) \\
&\leq C_\tau \left(\sum_{e \in \mathcal{E}} \frac{4 C_{\mathrm{inv}}}{tol_e} |u|_{a(\Omega_e)}^2 \right. \\
&\qquad + \left. \sum_{f \in \mathcal{F}} \frac{8 C_{\mathrm{inv}}}{tol_f} \left(|u|_{a(\Omega_f)}^2 + \sum_{e \in \mathcal{E} \cap \mathcal{B}_{\mathrm{p}}(f)} \frac{4 C_{\mathrm{inv}} C_\tau}{tol_e} |u|_{a(\Omega_e)}^2 \right) \right) \\
&\leq 4 C_{\mathrm{inv}} C_\tau \left(\frac{1}{tol_{\mathcal{E}}} \sum_{e \in \mathcal{E}} |u|_{a(\Omega_e)}^2 \right. \\
&\qquad + \left. \frac{2}{tol_{\mathcal{F}}} \sum_{f \in \mathcal{F}} \left(|u|_{a(\Omega_f)}^2 + \frac{4 C_{\mathrm{inv}} C_\tau}{tol_{\mathcal{E}}} \sum_{e \in \mathcal{E} \cap \mathcal{B}_{\mathrm{p}}(f)} |u|_{a(\Omega_e)}^2 \right) \right) \\
&\leq 4 C_{\mathrm{inv}} C_\tau \left(\frac{N^e}{tol_{\mathcal{E}}} + \frac{2 N^f}{tol_{\mathcal{F}}} + \frac{8 C_{\mathrm{inv}} C_\tau N^{\partial f \to e}}{tol_{\mathcal{F}} tol_{\mathcal{E}}} \right) |u|_{a(\Omega)}^2.
\end{aligned}
$$

By lemma 6.4, we then obtain

$$
|u_0|_{a(\Omega)} \leq \left(1 + \sqrt{4 C_{\mathrm{inv}} C_\tau \left(\frac{N^e}{tol_{\mathcal{E}}} + \frac{2 N^f}{tol_{\mathcal{F}}} + \frac{8 C_{\mathrm{inv}} C_\tau N^{\partial f \to e}}{tol_{\mathcal{F}} tol_{\mathcal{E}}} \right)} \right) |u|_{a(\Omega)}.
$$

$\square$

For the following corollary, we need to expand the definitions of N^e, N^f, and $N^{\partial f \to e}$ in (7.21) to (7.23), by including the number of subdomains adjacent to an edge or face as scaling factors. Let

$$
N^{e,\Sigma} := \max_{1 \leq i \leq N} \sum_{e \in \mathcal{E}(\Omega_i)} |n(e)|, \tag{7.24}
$$

$$
N^{f,\Sigma} := \max_{1 \leq i \leq N} \sum_{f \in \mathcal{F}(\Omega_i)} |n(f)| = \max_{1 \leq i \leq N} \sum_{f \in \mathcal{F}(\Omega_i)} 2 = 2 N^f, \tag{7.25}
$$

and

$$
N^{\partial f \to e, \Sigma} := \max_{1 \leq i \leq N} |n(f)| \sum_{f \in \mathcal{F}(\Omega_i)} |\mathcal{E} \cap \mathcal{B}_{\mathrm{p}}(f)| \tag{7.26}
$$

$$
= 2 \max_{1 \leq i \leq N} \sum_{f \in \mathcal{F}(\Omega_i)} |\mathcal{E} \cap \mathcal{B}_{\mathrm{p}}(f)| = 2 N^{\partial f \to e}.
$$

We formulate an analogue of corollary 6.2 for OS–ACMS.

Corollary 7.2. *(Analogue of corollary 6.2) Let the assumptions of lemma 6.5 be satisfied. Then*

$$|I^h(\Psi \cdot (u - u_0))|^2_{a(B)} \le \frac{4C_{\text{inv}}C_\tau}{tol_{\mathcal{E}}} \sum_{e \in \mathcal{E}(\Omega_l)} |u|^2_{a(\Omega_e)}$$

$$+ \frac{8C_{\text{inv}}C_\tau}{tol_{\mathcal{F}}} \sum_{f \in \mathcal{F}(\Omega_l)} \left(|u|^2_{a(\Omega_f)} + \frac{4C_{\text{inv}}C_\tau}{tol_{\mathcal{E}}} \sum_{e \in \mathcal{E} \cap \mathcal{B}_{\text{p}}(f)} |u|^2_{a(\Omega_e)} \right).$$

For $l = 0$, such that $B = \Omega$, we obtain

$$|I^h(\Psi \cdot (u - u_0))|^2_{a(\Omega)} \le 4C_{\text{inv}}C_\tau \left(\frac{N^e}{tol_{\mathcal{E}}} + \frac{2N^f}{tol_{\mathcal{F}}} + \frac{8C_{\text{inv}}C_\tau N^{\partial f \to e}}{tol_{\mathcal{F}} tol_{\mathcal{E}}} \right) |u|^2_{a(\Omega)}.$$

If the assumptions of lemma 6.5 are satisfied for all $1 \le l \le N$, $\Psi_l: \overline{B}_l \to \mathbb{R}$, with $B_l := \tilde{\Omega}_l \setminus \Omega_l$, we obtain

$$\sum_{l=1}^{N} |I^h(\Psi_l \cdot (u - u_0))|^2_{a(\tilde{\Omega}_l \setminus \Omega_l)} \le 4C_{\text{inv}}C_\tau \left(\frac{N^{e,\Sigma}}{tol_{\mathcal{E}}} + \frac{4N^f}{tol_{\mathcal{F}}} + \frac{16C_{\text{inv}}C_\tau N^{\partial f \to e}}{tol_{\mathcal{F}} tol_{\mathcal{E}}} \right) |u|^2_{a(\Omega)}.$$

Proof. Using lemmas 6.5 and 7.4 to 7.6, we have

$$|I^h(\Psi \cdot (u - u_0))|^2_{a(B)} \le C_\tau \sum_{\xi \in \mathcal{P}(\Omega_l)} \beta^K_\xi \left(u - u_0, u - u_0 \right)$$

$$\le C_\tau \sum_{e \in \mathcal{E}(\Omega_l)} \frac{4C_{\text{inv}}}{tol_e} |u|^2_{a(\Omega_e)}$$

$$+ C_\tau \sum_{f \in \mathcal{F}(\Omega_l)} \frac{8C_{\text{inv}}}{tol_f} \left(|u|^2_{a(\Omega_f)} + \sum_{e \in \mathcal{E} \cap \mathcal{B}_{\text{p}}(f)} \frac{4C_{\text{inv}}C_\tau}{tol_e} |u|^2_{a(\Omega_e)} \right)$$

$$\le \frac{4C_{\text{inv}}C_\tau}{tol_{\mathcal{E}}} \sum_{e \in \mathcal{E}(\Omega_l)} |u|^2_{a(\Omega_e)}$$

$$+ \frac{8C_{\text{inv}}C_\tau}{tol_{\mathcal{F}}} \sum_{f \in \mathcal{F}(\Omega_l)} \left(|u|^2_{a(\Omega_f)} + \frac{4C_{\text{inv}}C_\tau}{tol_{\mathcal{E}}} \sum_{e \in \mathcal{E} \cap \mathcal{B}_{\text{p}}(f)} |u|^2_{a(\Omega_e)} \right).$$

For $B = \Omega$, it follows that

$$|I^h(\Psi \cdot (u - u_0))|^2_{a(\Omega)} \le 4C_{\text{inv}}C_\tau \left(\frac{N^e}{tol_{\mathcal{E}}} + \frac{2N^f}{tol_{\mathcal{F}}} + \frac{8C_{\text{inv}}C_\tau N^{\partial f \to e}}{tol_{\mathcal{F}} tol_{\mathcal{E}}} \right) |u|^2_{a(\Omega)}.$$

For the sum over $l = 1, \ldots, N$, using (7.24) to (7.26), we obtain

$$\sum_{l=l}^{N} |I^h(\Psi_l \cdot (u - u_0))|^2_{a(\tilde{\Omega}_l \setminus \Omega_l)} \le 4C_{\text{inv}}C_\tau \left(\frac{N^{e,\Sigma}}{tol_{\mathcal{E}}} + \frac{2N^{f,\Sigma}}{tol_{\mathcal{F}}} + \frac{8C_{\text{inv}}C_\tau N^{\partial f \to e,\Sigma}}{tol_{\mathcal{F}} tol_{\mathcal{E}}} \right) |u|^2_{a(\Omega)}$$

$$= 4C_{\text{inv}}C_\tau \left(\frac{N^{e,\Sigma}}{tol_{\mathcal{E}}} + \frac{4N^f}{tol_{\mathcal{F}}} + \frac{16C_{\text{inv}}C_\tau N^{\partial f \to e}}{tol_{\mathcal{F}} tol_{\mathcal{E}}} \right) |u|^2_{a(\Omega)}.$$

$\square$

Lemma 7.7 (Stable Decomposition). *(Compare with [HKK+22, lemma 11.4; HKKR19, theorem 6.5; HKKR18b, lemma 6.4] and lemma 6.6.) For each $u \in V_{0,\partial\Omega_D}^h(\Omega)$, there exists a decomposition $u = \sum_{i=0}^N R_i^T u_i$, $u_i \in V_i$, $0 \le i \le N$, such that*

$$\sum_{i=0}^N |u_i|_{a(\Omega_i')}^2 \le C_0^2 |u|_{a(\Omega)}^2,$$

where

$$C_0^2 := 4 + 5(1 + \sqrt{D_1})^2 + D_0,$$

and

$$D_0 := 4C_{\text{inv}}C_\tau \left(\frac{N^{e,\Sigma}}{tol_{\mathcal{E}}} + \frac{4N^f}{tol_{\mathcal{F}}} + \frac{16C_{\text{inv}}C_\tau N^{\partial f \to e}}{tol_{\mathcal{F}} tol_{\mathcal{E}}} \right), \tag{7.27}$$

$$D_1 := 4C_{\text{inv}}C_\tau \left(\frac{N^e}{tol_{\mathcal{E}}} + \frac{2N^f}{tol_{\mathcal{F}}} + \frac{8C_{\text{inv}}C_\tau N^{\partial f \to e}}{tol_{\mathcal{F}} tol_{\mathcal{E}}} \right). \tag{7.28}$$

Proof. In the proof of lemma 6.6, we replace (6.43) by using corollary 7.2:

$$\sum_{i=1}^N |I^h(\theta_i(u - u_0))|_{a(\tilde{\Omega}_i \setminus \Omega_i)}^2 \le D_0 |u|_{a(\Omega)}^2.$$

We then replace (6.45) by using corollary 7.1:

$$\sum_{i=1}^N 2|u - u_0|_{a(\Omega_i)}^2 = 2|u - u_0|_{a(\Omega)}^2$$

$$\le 2 \left(|u|_{a(\Omega)} + |u_0|_{a(\Omega)} \right)^2$$

$$\le 2 \left(2 + \sqrt{D_1} \right)^2 |u|_{a(\Omega)}^2.$$

Finally, we replace (6.46) by using corollary 7.2:

$$\sum_{i=1}^N 2|I^h(\theta(u - u_0))|_{a(\Omega_i)}^2 = 2|I^h(\theta(u - u_0))|_{a(\Omega)}^2 \le 2D_1 |u|_{a(\Omega)}^2.$$

Analogously to the proof of lemma 6.6, and by using corollary 7.1, we obtain

$$\sum_{i=0}^N |u_i|_{a(\Omega_i')}^2 = |u_0|_{a(\Omega)}^2 + \sum_{i=1}^N |u_i|_{a(\tilde{\Omega}_i)}^2$$

$$\le \left((1 + \sqrt{D_1})^2 + D_0 + 2 \left(2 + \sqrt{D_1} \right)^2 + 2D_1 \right) |u|_{a(\Omega)}^2$$

$$= C_0^2 |u|_{a(\Omega)}^2,$$

where

$$C_0^2 = 4 + 5(1 + \sqrt{D_1})^2 + D_0.$$

$\square$

From (6.3) and lemma 7.7, we obtain a bound for the condition number of $M_{\text{OS–ACMS}}^{-1}K$, analogously to theorem 6.1.

Theorem 7.1. *The condition number of the OS–ACMS two-level Schwarz operator in three dimensions is bounded by*

$$\kappa\left(M_{\text{OS–ACMS}}^{-1}K\right) \le \left(4 + 5(1 + \sqrt{D_1})^2 + D_0\right)\left(\hat{N}_c + 1\right),$$

where D_0 and D_1 are defined in (7.27) and (7.28), and where $\hat{N}_c$ is the maximum number of overlapping subdomains $\{\Omega_i'\}_{i=1}^N$ any finite element node $x^h \in \overline{\Omega}$ can belong to. All constants are independent of H, h, and the contrast of the coefficient function E.

In two dimensions, we have

$$\kappa\left(M_{\text{OS–ACMS}}^{-1}K\right) \le \left(4 + 5\left(1 + \sqrt{D_1^{(2)}}\right)^2 + 2D_1^{(2)}\right)\left(\hat{N}_c + 1\right)$$

$$\le \left(14 + 12D_1^{(2)}\right)\left(\hat{N}_c + 1\right),$$

where

$$D_1^{(2)} := \frac{4C_{\text{inv}}C_\tau N^e}{tol_{\mathcal{E}}}.$$

A condition number bound for R–WB–OS–ACMS is given in theorem A.1. The proof is essentially identical; see appendix A.3.

7.5. Generalization of OS–ACMS and RAGDSW

Let $\mathcal{P}$ be an interface partition. We will generalize the concept of enforcing additional Dirichlet conditions in the energy-minimizing extension that is incorporated into the generalized eigenvalue problem. For the description, we restrict ourselves to the standard OS–ACMS generalized eigenvalue problem in section 7.2.2; the description for variants is analogous.
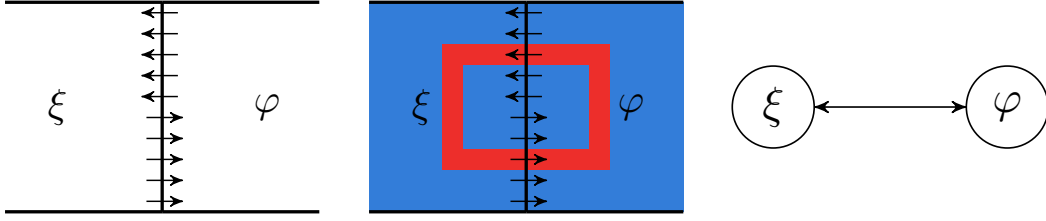
Figure 7.5.: **(Left)** Two adjacent interface components $\xi, \varphi \in \mathcal{P}$, where ξ has a Dirichlet condition on φ on the top part of the common boundary, and φ has a Dirichlet condition on ξ on the bottom part of the common boundary. The direction of an energy-minimizing extension is indicated by arrows; cf. fig. 7.6. The extension is not well defined and the case inadmissible. **(Center)** Coefficient function E on ξ and φ; $E = 10^6$ (red); $E = 1$ (blue). **(Right)** Corresponding graph $\mathcal{G}$ that contains an edge from ξ to φ and from φ to ξ.

In the following, we say that $\xi \in \mathcal{P}$ has a Dirichlet condition on $\varphi \in \mathcal{P}$ if the energy-minimizing extension in the generalized eigenvalue problem associated with ξ enforces a zero Dirichlet condition on $\varphi \cap \overline{\Omega}_\xi$. In that case, an eigenfunction of φ must be extended energy-minimally to ξ to construct a coarse function. In the case of OS–ACMS, coarse faces have a Dirichlet condition on adjacent coarse nodes and edges. Thus, edge eigenfunctions and vertex functions must be extended energy-minimally to adjacent coarse faces.

Two remarks are necessary: First, we may relax the above condition and only prescribe a Dirichlet condition on a subset of $\varphi \cap \overline{\Omega}_\xi$. Second, if ξ has Dirichlet condition on φ, φ must not have a Dirichlet condition on ξ. An example for which the corresponding coarse space would fail to be robust, is given in fig. 7.5. Therein, on the top part of the common boundary of ξ and φ, the component ξ has a Dirichlet condition on φ. Thus, the associated eigenvalue problem cannot detect the patch of large coefficients; cf. fig. 6.2. As a result, the eigenvalue problem associated with φ must detect it. However, as φ has a Dirichlet condition on ξ on the bottom part of the common boundary, it, too, cannot detect the patch of large coefficients. In the following, we assume that Dirichlet conditions are always prescribed on the entirety of $\varphi \cap \overline{\Omega}_\xi$.
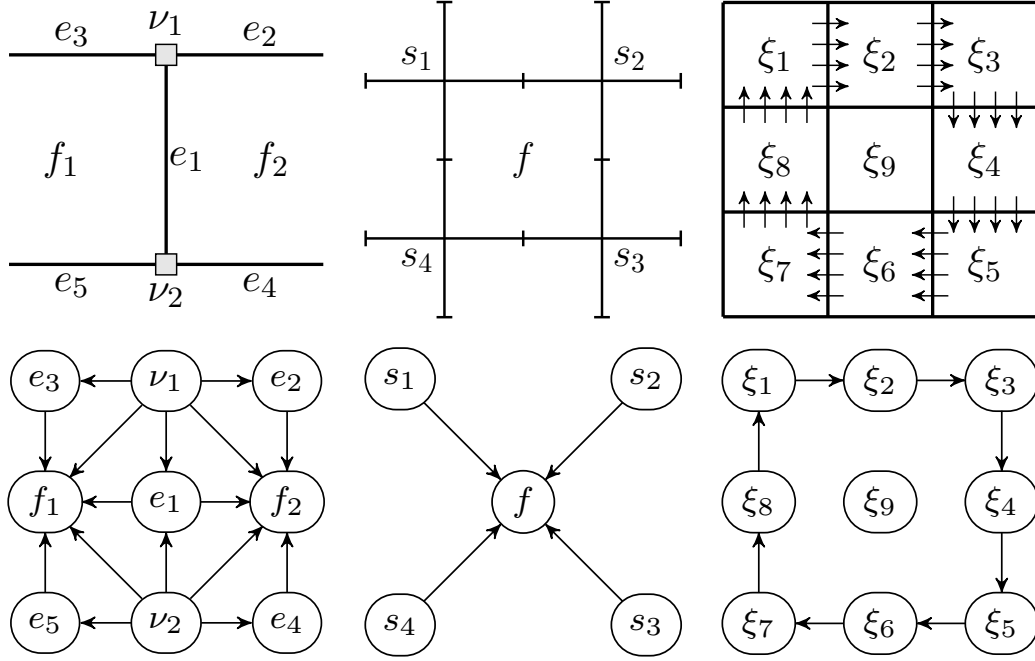
Figure 7.6.: Schematics of interface partitions (top row) and associated graphs $\mathcal{G}$ (bottom row). **(Left)** OS–ACMS interface partition with coarse nodes ν_1, ν_2, coarse edges $e_1, \ldots, e_5$, and coarse faces f_1, f_2. **(Center)** R–WB–OS–ACMS interface partition with interface stars $s_1, \ldots, s_4$, and the coarse face f. **(Right)** Interface partition with the components ξ_i, $i = 1, \ldots, 9$. The direction of energy-minimizing extensions is indicated by arrows: ξ_i has a Dirichlet condition on ξ_{i-1}, $i = 1, \ldots, 8$, where $\xi_0 := \xi_8$.

The Dirichlet conditions prescribed on interface components induce a directed graph

$$\mathcal{G}(\mathcal{P}, D_{\mathcal{P}}),$$

where $D_{\mathcal{P}}$ contains a directed edge (φ, ξ) from φ to ξ if ξ has a Dirichlet condition on φ. An edge of the graph indicates that parts of φ lie in $\overline{\Omega}_\xi$ and that an energy-minimizing extension from φ to ξ is required. To obtain a robust preconditioner, we make the following assumption.

Assumption 7.1. *$\mathcal{G}(\mathcal{P}, D_{\mathcal{P}})$ is a directed, acyclic graph; that is, there does not exist a directed cycle.*

Examples of graphs that do not fulfill this assumption are given in fig. 7.5 (right) and fig. 7.6 (right). The graph corresponding to an adaptive GDSW-type coarse space is given by $\mathcal{G}(\mathcal{P}, D_{\mathcal{P}})$ where $D_{\mathcal{P}} = \emptyset$. For OS–ACMS and R–WB–OS–ACMS, schematics of $\mathcal{G}$ are given in fig. 7.6 (left and center).

Based on $\mathcal{G}$, we can define the boundary of an interface component $\xi \in \mathcal{P}$:

$$\overline{\xi} := \xi \cup \partial \xi, \quad \partial \xi := \{\, \varphi \cap \overline{\Omega}_\xi : \varphi \in \mathcal{P} \ \wedge \ (\varphi, \xi) \in D_{\mathcal{P}} \,\}. \tag{7.29}$$

Thus, by definition, the boundary of ξ contains the finite element nodes of $\varphi \in \mathcal{P}$ that ξ has a Dirichlet condition on and that lie inside $\overline{\Omega}_\xi$. The generalized eigenvalue problem associated with ξ is now given by the one in section 7.2.2.

For the second part of the coarse space construction, we need to define energy-minimizing extensions of eigenfunctions; an extension from one interface component to another is defined analogously to the one in section 7.2.3. Let us consider $\mathcal{G}$ for the OS–ACMS interface partition in fig. 7.6 (left). Although ν_1 extends to f_1, we cannot immediately compute the extension since ν_1 also extends to e_1 and e_3, which in turn both extend to f_1. Thus, we first have to compute an extension from ν_1 to e_1 and from ν_1 to e_3. Given the values on ν_1, e_1, and e_3, we extend by zero to the remaining boundary nodes of f_1 and then energy-minimally to f_1.

Let $\xi \in \mathcal{P}$. As in section 7.2.3, let $\mathcal{B}_{\mathrm{c}}(\xi)$ (children) denote the set of interface components that have a Dirichlet condition on ξ, and let $\mathcal{B}_{\mathrm{p}}(\xi)$ (parents) denote the set of interface components on which ξ has a Dirichlet condition:

$$\mathcal{B}_{\mathrm{c}}(\xi) := \{\, \varphi \in \mathcal{P} : (\xi, \varphi) \in D_{\mathcal{P}} \,\}, \tag{7.30}$$

$$\mathcal{B}_{\mathrm{p}}(\xi) := \{\, \varphi \in \mathcal{P} : (\varphi, \xi) \in D_{\mathcal{P}} \,\}. \tag{7.31}$$

We have

$$\partial \xi = \bigcup_{\varphi \in \mathcal{B}_{\mathrm{p}}(\xi)} \varphi \cap \overline{\Omega}_\xi. \tag{7.32}$$

By $\mathcal{B}_{\mathrm{A}}(\xi)$ (ancestors) we denote the set of interface components from which energy-minimizing extensions to ξ need to be computed. Note that ancestors need not be in the same domain $\overline{\Omega}_\xi$ as ξ, in which case cascaded extensions involving other components

---

Algorithm 1 Cascaded Energy-Minimizing Extension of Eigenfunctions

---

1: Let $\xi \in \mathcal{P}$, $v|_\xi \in X^h(\xi)$. $\triangleright$ Goal: define $v \in X^h(\Omega^h)$.

2: label(ξ) := finished.

3: **for all** $\varphi \in \mathcal{P}$ that satisfy $\varphi \notin \mathcal{B}_{\mathrm{D}}(\xi)$ **do**

4:      $v|_\varphi := 0$.

5:      label(φ) := finished.

6: **repeat**

7:      **for all** $\varphi \in \mathcal{B}_{\mathrm{D}}(\xi)$ that satisfy "label($\tilde{\varphi}$) is finished for all $\tilde{\varphi} \in \mathcal{B}_{\mathrm{p}}(\varphi)$" **do**

8:         Energy-minimally extend from $\partial\varphi$ to φ: $v|_\varphi := \mathcal{H}_{\partial\varphi \to \Omega_\varphi}(v|_{\partial\varphi})\big|_\varphi$.

9:         label(φ) := finished.

10: **until** all components are labeled as finished.

11: Energy-minimally extend $v|_{\Gamma^h}$ to Ω: $v \leftarrow \mathcal{H}_{\Gamma^h \to \Omega}(v|_{\Gamma^h})$.

---

have to be computed; cf. fig. 7.6 (right). By $\mathcal{B}_{\mathrm{D}}(\xi)$ (descendants) we denote the set of interface components to which (directly or indirectly) energy-minimizing extensions from ξ need to be computed.

$$\mathcal{B}_{\mathrm{A}}(\xi) := \begin{cases} \emptyset & \text{if } \mathcal{B}_{\mathrm{p}}(\xi) = \emptyset, \\ \bigcup_{\varphi \in \mathcal{B}_{\mathrm{p}}(\xi)} \left(\{\varphi\} \cup \mathcal{B}_{\mathrm{A}}(\varphi)\right) & \text{else.} \end{cases} \tag{7.33}$$

$$\mathcal{B}_{\mathrm{D}}(\xi) := \begin{cases} \emptyset & \text{if } \mathcal{B}_{\mathrm{c}}(\xi) = \emptyset, \\ \bigcup_{\varphi \in \mathcal{B}_{\mathrm{c}}(\xi)} \left(\{\varphi\} \cup \mathcal{B}_{\mathrm{D}}(\varphi)\right) & \text{else.} \end{cases} \tag{7.34}$$

Note that, since $\mathcal{G}$ is acyclic, these recursive definitions are well defined. An example for which the recursive definition would fail is given by the graph in fig. 7.6 (right); the graph is inadmissable by assumption 7.1: As the graph contains a cycle, the set of ancestors or descendants of, e.g., ξ_1 is not well defined.

In algorithm 1, a procedure for the cascades of energy-minimizing extensions of eigenfunctions is given, which concludes the construction of the new coarse space.

Remark 7.4. *Let Ω be given by the unit cube, partitioned into smaller cubes. To construct a coarse space, we select the RGDSW interface partition and an admissible $\mathcal{G}(\mathcal{P}, D_{\mathcal{P}})$ for*

which $D_{\mathcal{P}} \neq \emptyset$. Let an interface star $\xi \in \mathcal{P}$ be given, and let Z denote the set of interface stars adjacent to ξ. Let us assume that $\tilde{\xi} \in Z$ have a Dirichlet condition on ξ. Then, we need to compute energy-minimizing extensions from ξ to $\tilde{\xi} \in Z$ to $\Omega_{\tilde{\xi}}$. As a result, the coarse function associated with ξ extends by two layers of subdomains surrounding the vertex that ξ is associated with. The construction of such coarse functions that span large areas negatively influences parallelizability. Furthermore, numerical scalability is affected negatively as well; cf. theorem 7.2 and section 6.4.3.

To that effect, the OS–ACMS and R–WB–OS–ACMS coarse spaces are optimal as they only involve extensions to the subdomains directly adjacent to the respective interface components.

In the following, we prove a condition number bound for the constructed coarse space. Let $u \in V^h(\Omega)$ and $tol_\xi > 0$ for $\xi \in \mathcal{P}$. We define the spectral projections

$$\Pi_\xi u := \sum_{\lambda_{k,\xi} \leq tol_\xi} \beta_\xi(u, v_{k,\xi}) v_{k,\xi}.$$

The coarse component is defined as

$$u_0 := \sum_{\varphi \in \mathcal{P}} \Pi_\varphi G(u, \varphi),$$

where

$$G(u, \varphi) := u - \sum_{\tilde{\varphi} \in \mathcal{B}_{\mathrm{A}}(\varphi)} \Pi_{\tilde{\varphi}} G(u, \tilde{\varphi}).$$

The recursion halts after a finite number of steps as the graph $\mathcal{G}$ is acyclic by assumption.

Similarly to OS–ACMS, we can reuse much of the proof of the RAGDSW condition number bound. We replace lemmas 6.2 and 6.3 with the following lemma; the corresponding lemmas in the case of OS–ACMS are lemmas 7.4 to 7.6.

Lemma 7.8. *Let $\xi \in \mathcal{P}$ and $u \in V^h(\Omega)$. Then we have*

$$\|u - u_0\|^2_{\beta_\xi^K} \leq g(u, \xi),$$

where

$$g(u, \xi) := \begin{cases} \frac{C_{\mathrm{inv}}}{tol_\xi} |u|^2_{a(\Omega_\xi)} & \text{if } \mathcal{B}_{\mathrm{p}}(\xi) = \emptyset, \\ \frac{8 C_{\mathrm{inv}}}{tol_\xi} \left(|u|^2_{a(\Omega_\xi)} + C_\tau \sum_{\varphi \in \mathcal{B}_{\mathrm{A}}(\xi)} g(u, \varphi) \right) & \text{else.} \end{cases}$$

Proof. By definition, we have

$$\|u - u_0\|^2_{\beta^K_\xi} = \sum_{\xi_i \in \mathcal{N}_\xi} |z_{\xi_i}(u - u_0)|^2_{a(\Omega_{\xi_i})}.$$

As there are only contributions on ξ of coarse functions that are associated with $\mathcal{B}_A(\xi)$—and of ξ itself—we have

$$
\begin{aligned}
z_{\xi_i}(u - u_0) &= z_{\xi_i}\Big(u - \sum_{\varphi \in \mathcal{P}} \Pi_\varphi G(u, \varphi)\Big) \\
&= z_{\xi_i}\Big(u - \sum_{\varphi \in \mathcal{B}_A(\xi)} \Pi_\varphi G(u, \varphi) - \Pi_\xi G(u, \xi)\Big) \\
&= z_{\xi_i}\Big(G(u, \xi) - \Pi_\xi G(u, \xi)\Big).
\end{aligned}
$$

We obtain

$$
\begin{aligned}
\sum_{\xi_i \in \mathcal{N}_\xi} |z_{\xi_i}(u - u_0)|^2_{a(\Omega_{\xi_i})} &= \sum_{\xi_i \in \mathcal{N}_\xi} \Big|z_{\xi_i}\Big(G(u, \xi) - \Pi_\xi G(u, \xi)\Big)\Big|^2_{a(\Omega_{\xi_i})} \\
&= \|G(u, \xi) - \Pi_\xi G(u, \xi)\|^2_{\beta^K_\xi}.
\end{aligned}
$$

Using (6.38) and lemma 6.1, it follows that

$$\|G(u, \xi) - \Pi_\xi G(u, \xi)\|^2_{\beta^K_\xi} \le \frac{C_{\text{inv}}}{tol_\xi} |G(u, \xi)|^2_{\bar{\alpha}_\xi}. \tag{7.35}$$

If ξ does not have any parents (i.e., $\mathcal{B}_p(\xi) = \emptyset = \mathcal{B}_A(\xi)$), we have $\bar{\xi} = \xi$ and obtain with (7.17) and (7.19) that

$$|G(u, \xi)|^2_{\bar{\alpha}_\xi} = |u|^2_{\bar{\alpha}_\xi} = |u|^2_{\alpha_{\bar{\xi}}} \le |u|^2_{a(\Omega_\xi)}. \tag{7.36}$$

Thus, we have

$$\|u - u_0\|^2_{\beta^K_\xi} \le \frac{C_{\text{inv}}}{tol_\xi} |u|^2_{a(\Omega_\xi)}.$$

Let us now assume that $\mathcal{B}_p(\xi)$ is not empty. We use the equality

$$z_\xi(G(u, \xi)) = z_{\bar{\xi}}(G(u, \xi)) - z_{\partial\xi}(G(u, \xi)) \quad \text{on } \bar{\xi}.$$

Then we have

$$
\begin{aligned}
|G(u,\xi)|_{\overline{\alpha}_\xi} &= \left| z_\xi(G(u,\xi)) \right|_{\alpha_{\overline{\xi}}} \\
&\leq \left| z_{\overline{\xi}}(G(u,\xi)) \right|_{\alpha_{\overline{\xi}}} + \left| z_{\partial\xi}(G(u,\xi)) \right|_{\alpha_{\overline{\xi}}} \\
&= |G(u,\xi)|_{\alpha_{\overline{\xi}}} + \left| z_{\partial\xi}(G(u,\xi)) \right|_{\alpha_{\overline{\xi}}}.
\end{aligned}
\tag{7.37}
$$

For the first term in (7.37), we obtain

$$
\begin{aligned}
|G(u,\xi)|_{\alpha_{\overline{\xi}}} &= \left| u - \sum_{\varphi\in\mathcal{B}_{\mathrm{A}}(\xi)} \Pi_\varphi G(u,\varphi) \right|_{\alpha_{\overline{\xi}}} \\
&\leq |u|_{\alpha_{\overline{\xi}}} + \left| \sum_{\varphi\in\mathcal{B}_{\mathrm{A}}(\xi)} \Pi_\varphi G(u,\varphi) \right|_{\alpha_{\overline{\xi}}} \\
&\leq |u|_{a(\Omega_\xi)} + \left| \sum_{\varphi\in\mathcal{B}_{\mathrm{A}}(\xi)} \Pi_\varphi G(u,\varphi) \right|_{\alpha_{\overline{\xi}}},
\end{aligned}
$$

where we have used (7.17) and (7.19) in the last step.

As a consequence of the energy-minimality of the contributions $\sum_{\varphi\in\mathcal{B}_{\mathrm{A}}(\xi)} \Pi_\varphi G(u,\varphi)$ on $\overline{\xi}$, using (7.16), we have

$$
\left| \sum_{\varphi\in\mathcal{B}_{\mathrm{A}}(\xi)} \Pi_\varphi G(u,\varphi) \right|_{\alpha_{\overline{\xi}}} \leq \left| \sum_{\varphi\in\mathcal{B}_{\mathrm{A}}(\xi)} \Pi_\varphi G(u,\varphi) \right|_{a,\partial\xi\to\Omega_\xi}.
$$

Then

$$
\begin{aligned}
\left| \sum_{\varphi\in\mathcal{B}_{\mathrm{A}}(\xi)} \Pi_\varphi G(u,\varphi) \right|_{a,\partial\xi\to\Omega_\xi} &\leq |u|_{a,\partial\xi\to\Omega_\xi} + \left| u - \sum_{\varphi\in\mathcal{B}_{\mathrm{A}}(\xi)} \Pi_\varphi G(u,\varphi) \right|_{a,\partial\xi\to\Omega_\xi} \\
&\leq |u|_{a(\Omega_\xi)} + |G(u,\xi)|_{a,\partial\xi\to\Omega_\xi},
\end{aligned}
$$

where we have used (7.19) in the last step. For $w\in X^h(\Omega^h)$, using (7.32), we have

$$
w = \sum_{\varphi\in\mathcal{B}_{\mathrm{p}}(\xi)} z_\varphi(w) = \sum_{\varphi\in\mathcal{B}_{\mathrm{A}}(\xi)} z_\varphi(w) \quad \text{on } \partial\xi.
$$

Thus, using (7.19), we obtain

$$
|G(u,\xi)|_{a,\partial\xi\to\Omega_\xi} = \left| \sum_{\varphi\in\mathcal{B}_{\mathrm{A}}(\xi)} z_\varphi(G(u,\xi)) \right|_{a,\partial\xi\to\Omega_\xi} \leq \left| \sum_{\varphi\in\mathcal{B}_{\mathrm{A}}(\xi)} z_\varphi(G(u,\xi)) \right|_{a(\Omega_\xi)}.
$$

Similarly, for the second term of (7.37), we have

$$\left| z_{\partial \xi}\big(G(u, \xi)\big) \right|_{\alpha_{\overline{\xi}}} = \Big| \sum_{\varphi \in \mathcal{B}_{\mathrm{A}}(\xi)} z_{\varphi}\big(G(u, \xi)\big) \Big|_{\alpha_{\overline{\xi}}} \leq \Big| \sum_{\varphi \in \mathcal{B}_{\mathrm{A}}(\xi)} z_{\varphi}\big(G(u, \xi)\big) \Big|_{a(\Omega_{\xi})}.$$

Thus, for (7.37), we obtain

$$|G(u, \xi)|^2_{\overline{\alpha}_{\xi}} \leq 8|u|^2_{a(\Omega_{\xi})} + 8\Big| \sum_{\varphi \in \mathcal{B}_{\mathrm{A}}(\xi)} z_{\varphi}\big(G(u, \xi)\big) \Big|^2_{a(\Omega_{\xi})}. \tag{7.38}$$

In the case of OS–ACMS, if ξ is a coarse edge, the right term on the right-hand side of (7.38) would be zero because of the exact interpolation of the vertex functions in the vertices. As a result, we would obtain the more favorable bound $|G(u, e)|^2_{\overline{\alpha}_e} \leq 4|u|^2_{a(\Omega_e)}$. We will later apply our theory to OS–ACMS and obtain bounds for the constants D_0 and D_1 that are slightly larger than the ones in (7.27) and (7.28), respectively.

Let $\varphi \in \mathcal{B}_{\mathrm{A}}(\xi)$. Then only coarse functions associated with ancestors of φ and φ itself have contributions on φ. Therefore, on φ, the equality

$$G(u, \xi) = u - \sum_{\tilde{\varphi} \in \mathcal{B}_{\mathrm{A}}(\xi)} \Pi_{\tilde{\varphi}} G(u, \tilde{\varphi})$$

$$= u - \sum_{\tilde{\varphi} \in \mathcal{B}_{\mathrm{A}}(\varphi)} \Pi_{\tilde{\varphi}} G(u, \tilde{\varphi}) - \Pi_{\varphi} G(u, \varphi)$$

$$= G(u, \varphi) - \Pi_{\varphi} G(u, \varphi)$$

holds, where we have also used that $\mathcal{B}_{\mathrm{A}}(\varphi) \cup \{\varphi\} \subseteq \mathcal{B}_{\mathrm{A}}(\xi)$. Let $w_{\varphi} \coloneqq G(u, \varphi) - \Pi_{\varphi} G(u, \varphi)$. Similarly to the line of proof in lemma 6.4, and by using a Cauchy–Schwarz inequality, we obtain

$$\Big| \sum_{\varphi \in \mathcal{B}_{\mathrm{A}}(\xi)} z_{\varphi}\big(G(u, \xi)\big) \Big|^2_{a(\Omega_{\xi})} = \Big| \sum_{\varphi \in \mathcal{B}_{\mathrm{A}}(\xi)} z_{\varphi}(w_{\varphi}) \Big|^2_{a(\Omega_{\xi})}$$

$$= \sum_{T \in \tau_h(\Omega_{\xi})} \Big| \sum_{\varphi \in \mathcal{B}_{\mathrm{A}}(\xi)} \sum_{\varphi_i \in \mathcal{N}_{\varphi}} z_{\varphi_i}(w_{\varphi}) \Big|^2_{a(T)}$$

$$\leq C_{\tau} \sum_{T \in \tau_h(\Omega_{\xi})} \sum_{\varphi \in \mathcal{B}_{\mathrm{A}}(\xi)} \sum_{\varphi_i \in \mathcal{N}_{\varphi}} \big| z_{\varphi_i}(w_{\varphi}) \big|^2_{a(T)}$$

$$\leq C_{\tau} \sum_{\varphi \in \mathcal{B}_{\mathrm{A}}(\xi)} \sum_{\varphi_i \in \mathcal{N}_{\varphi}} \big| z_{\varphi_i}(w_{\varphi}) \big|^2_{a(\Omega_{\varphi_i})}$$

$$= C_{\tau} \sum_{\varphi \in \mathcal{B}_{\mathrm{A}}(\xi)} \big\| G(u, \varphi) - \Pi_{\varphi} G(u, \varphi) \big\|^2_{\beta_{\varphi}^K}.$$

190

For (7.35), with (7.38), it follows that

$$\|G(u,\xi) - \Pi_\xi G(u,\xi)\|^2_{\beta^K_\xi} \leq \frac{C_{\text{inv}}}{tol_\xi}\Big(8|u|^2_{a(\Omega_\xi)} + 8C_\tau \sum_{\varphi \in \mathcal{B}_A(\xi)} \|G(u,\varphi) - \Pi_\varphi G(u,\varphi)\|^2_{\beta^K_\varphi}\Big).$$

Thus, we obtain with (7.36) that

$$\|u - u_0\|^2_{\beta^K_\xi} = \|G(u,\xi) - \Pi_\xi G(u,\xi)\|^2_{\beta^K_\xi} \leq g(u,\xi),$$

where

$$g(u,\xi) = \begin{cases} \frac{C_{\text{inv}}}{tol_\xi}|u|^2_{a(\Omega_\xi)} & \text{if } \mathcal{B}_p(\xi) = \emptyset, \\ \frac{8C_{\text{inv}}}{tol_\xi}\Big(|u|^2_{a(\Omega_\xi)} + C_\tau \sum_{\varphi \in \mathcal{B}_A(\xi)} g(u,\varphi)\Big) & \text{else.} \end{cases}$$

$\square$

To derive an explicit condition number bound, we will eliminate the recursion in the definition of $g(u,\xi)$. We define a path to describe the recursion, which stems from the cascaded energy-minimizing extensions. A path must begin with ξ and end with an ancestor $\varphi \in \mathcal{B}_A(\xi)$:

$$p = (\xi = p_1, \, p_2, \, \ldots, \, p_{|p|-1}, \, p_{|p|} = \varphi),$$

where $|p|$ is the length of the path (number of elements); a path can have any length (it must be finite, however, since $\mathcal{G}$ is acyclic). $p = (\xi)$ is an admissible path as well. Furthermore, p_{i+1} must be an ancestor of p_i; that is, $p_{i+1} \in \mathcal{B}_A(p_i)$. We note that p_{i+1} does not have to be a parent of p_i. By path(ξ) we denote the set of all admissible paths for $\xi \in \mathcal{P}$. The set of paths that begin at any interface component and end at ξ is given by

$$\text{path}(\mathcal{P},\xi) := \bigcup_{\varphi \in \mathcal{P}} \bigcup_{\substack{p \in \text{path}(\varphi) \\ p_{|p|} = \xi}} \{p\}.$$

We can now eliminate the recursion from $g(u,\xi)$ and obtain

$$g(u,\xi) = \sum_{\substack{p \in \text{path}(\xi) \\ \varphi := p_{|p|}}} (8^{|p|-1}g_\varphi)C^{|p|}_{\text{inv}}C^{|p|-1}_\tau \left(\prod_{j=1}^{|p|} \frac{1}{tol_{p_j}}\right)|u|^2_{a(\Omega_\varphi)}, \qquad (7.39)$$

191

where

$$g_\varphi := \begin{cases} 1 & \text{if } \mathcal{B}_\mathrm{A}(\varphi) = \emptyset, \\ 8 & \text{else.} \end{cases}$$

We define the related constant

$$N^\mathcal{G} := \max_{1 \le i \le N} \sum_{\xi \in \mathcal{P}(\Omega_i)} \sum_{p \in \mathrm{path}(\mathcal{P}, \xi)} (8^{|p|-1} g_\xi) C_\mathrm{inv}^{|p|} C_\tau^{|p|-1} \left(\prod_{j=1}^{|p|} \frac{1}{tol_{p_j}} \right), \qquad (7.40)$$

where $\mathcal{P}(\Omega_i) = \{ \xi \in \mathcal{P} : \xi \cap \overline{\Omega}_i \ne \emptyset \}$. The constant $N^\mathcal{G}$ is a weighted measure for the $\mathcal{G}$-connectivity of $\mathcal{P}$. In simple terms, $\mathcal{G}$-connectivity can be interpreted as: across how many interface components do cascaded energy-minimizing extensions stretch?

Corollary 7.3. *(Analogue of corollary 7.1) It holds that*

$$|u_0|_{a(\Omega)} \le \left(1 + \sqrt{C_\tau N^\mathcal{G}} \right) |u|_{a(\Omega)}.$$

Proof. Using lemma 7.8 and (7.39), we have

$$C_\tau \sum_{\xi \in \mathcal{P}} \|u - u_0\|_{\beta_\xi^K}^2 \le C_\tau \sum_{\xi \in \mathcal{P}} g(u, \xi) \le C_\tau N^\mathcal{G} |u|_{a(\Omega)}^2.$$

We motivate the second inequality but, for simplicity, ignore the weights encountered in $N^\mathcal{G}$. The question is: how many times does $|u|_{a(\Omega_i)}^2$ appear in $\sum_{\xi \in \mathcal{P}} g(u, \xi)$? In order for $|u|_{a(\Omega_i)}^2$ to appear, it must be part of $|u|_{a(\Omega_\xi)}^2$, where ξ is an interface component adjacent to Ω_i. Thus, we sum over all $\xi \in \mathcal{P}(\Omega_i)$. The new question is: how many times is ξ encountered? It is encountered as many times as there are paths from any $\varphi \in \mathcal{P}$ that end at ξ. Therefore, we sum over all $\varphi \in \mathrm{path}(\mathcal{P}, \xi)$.

Using lemma 6.4, we obtain

$$|u_0|_{a(\Omega)} \le \left(1 + \sqrt{C_\tau N^\mathcal{G}} \right) |u|_{a(\Omega)}.$$

$\square$

We extend the definition of $N^\mathcal{G}$ to include the factor $n^\varphi := \max_{\varphi_i \in \mathcal{N}_\varphi} |n(\varphi_i)|$:

$$N^{\mathcal{G}, \Sigma} := \max_{1 \le i \le N} \sum_{\xi \in \mathcal{P}(\Omega_i)} \sum_{\substack{p \in \mathrm{path}(\mathcal{P}, \xi) \\ \varphi := p_1}} n^\varphi (8^{|p|-1} g_\xi) C_\mathrm{inv}^{|p|} C_\tau^{|p|-1} \left(\prod_{j=1}^{|p|} \frac{1}{tol_{p_j}} \right). \qquad (7.41)$$

Remark 7.5. *In comparison with the proof in chapter 6, $N^{\mathcal{G}}$ is the analogue of $C_{\mathrm{inv}}N^{\xi,tol_\xi}$ (cf. (6.36)) and $N^{\mathcal{G},\Sigma}$ that of $C_{\mathrm{inv}}\hat{\mathcal{C}}^{tol_\xi}$ (cf. (6.48)). In fact, if we use an adaptive GDSW-type coarse space, we have $\mathcal{B}_{\mathrm{A}}(\xi) = \mathcal{B}_{\mathrm{D}}(\xi) = \emptyset$. As a result, the only admissible path for $\xi \in \mathcal{P}$ is given by $p = (\xi)$, and we obtain*

$$N^{\mathcal{G}} = C_{\mathrm{inv}} \max_{1 \leq i \leq N} \sum_{\xi \in \mathcal{P}(\Omega_i)} \frac{1}{tol_\xi} = C_{\mathrm{inv}}N^{\xi,tol_\xi}.$$

Similarly, we have

$$N^{\mathcal{G},\Sigma} = C_{\mathrm{inv}} \max_{1 \leq i \leq N} \sum_{\xi \in \mathcal{P}(\Omega_i)} \Big(\max_{\xi_i \in \mathcal{N}_\xi} |n(\xi_i)| \Big) \frac{1}{tol_\xi} = C_{\mathrm{inv}}\hat{\mathcal{C}}^{tol_\xi}.$$

Corollary 7.4. *(Analogue of corollary 7.2) Let the assumptions of lemma 6.5 be satisfied. Then*

$$|I^h(\Psi \cdot (u - u_0))|_{a(B)}^2 \leq C_\tau \sum_{\xi \in \mathcal{P}(\Omega_l)} g(u,\xi).$$

For $l = 0$, such that $B = \Omega$, we obtain

$$|I^h(\Psi \cdot (u - u_0))|_{a(\Omega)}^2 \leq D_1|u|_{a(\Omega)}^2.$$

If the assumptions of lemma 6.5 are satisfied for all $1 \leq l \leq N$, $\Psi_l \colon \overline{B}_l \to \mathbb{R}$, with $B_l := \tilde{\Omega}_l \setminus \Omega_l$, we obtain

$$\sum_{l=1}^N |I^h(\Psi_l \cdot (u - u_0))|_{a(\tilde{\Omega}_l \setminus \Omega_l)}^2 \leq C_\tau \sum_{\xi \in \mathcal{P}} \Big(\max_{\xi_i \in \mathcal{N}_\xi} |n(\xi_i)| \Big) g(u,\xi) \leq D_0|u|_{a(\Omega)}^2.$$

The constants D_0 and D_1 are defined as

$$D_0 := C_\tau N^{\mathcal{G},\Sigma}, \tag{7.42}$$

$$D_1 := C_\tau N^{\mathcal{G}}. \tag{7.43}$$

Proof. Using lemmas 6.5 and 7.8, we have

$$|I^h(\Psi \cdot (u - u_0))|_{a(B)}^2 \leq C_\tau \sum_{\xi \in \mathcal{P}(\Omega_l)} \beta_\xi^K\big(u - u_0\,,\,u - u_0\big) \leq C_\tau \sum_{\xi \in \mathcal{P}(\Omega_l)} g(u,\xi).$$

For $B = \Omega$, it follows that

$$|I^h(\Psi \cdot (u - u_0))|_{a(\Omega)}^2 \leq C_\tau N^{\mathcal{G}}|u|_{a(\Omega)}^2;$$

cf. the proof of corollary 7.3.

For the sum over $l = 1, \ldots, N$, we note (6.47) in section 6.4.3 and obtain, using (7.39) and (7.41),

$$\sum_{l=l}^{N} |I^h(\Psi_l \cdot (u - u_0))|^2_{a(\tilde{\Omega}_l \setminus \Omega_l)} \leq C_\tau \sum_{\xi \in \mathcal{P}} \left(\max_{\xi_i \in \mathcal{N}_\xi} |n(\xi_i)| \right) g(u, \xi) \leq C_\tau N^{\mathcal{G}, \Sigma} |u|^2_{a(\Omega)}.$$

$\square$

The proof of the existence of a stable decomposition is identical to that of lemma 7.7 for OS–ACMS if we substitute the constants D_0 and D_1 with (7.42) and (7.43). Thus, similarly to theorem 7.1, we obtain the following condition number bound.

Theorem 7.2. *The condition number of an ACMS-type two-level Schwarz operator is bounded by*

$$\kappa \left(M_{\text{ACMS}}^{-1} K \right) \leq \left(4 + 5(1 + \sqrt{D_1})^2 + D_0 \right) \left(\hat{N}_c + 1 \right),$$

where D_0 and D_1 are defined in (7.42) and (7.43), and where $\hat{N}_c$ is the maximum number of overlapping subdomains $\{\Omega_i'\}_{i=1}^{N}$ any finite element node $x^h \in \overline{\Omega}$ can belong to. All constants are independent of H, h, and the contrast of the coefficient function E.

In the following, we apply theorem 7.2 to GDSW-type coarse spaces and the coarse spaces R–WB–OS–ACMS and OS–ACMS.

GDSW-type Coarse Space: According to remark 7.5, the equalities $N^{\mathcal{G}} = C_{\text{inv}} N^{\xi, tol_\xi}$ and $N^{\mathcal{G}, \Sigma} = C_{\text{inv}} \hat{\mathcal{C}}^{tol_\xi}$ are satisfied in the case of a GDSW-type coarse space. Therefore, we obtain

$$D_0 = C_\tau N^{\mathcal{G}, \Sigma} = C_{\text{inv}} C_\tau \hat{\mathcal{C}}^{tol_\xi},$$
$$D_1 = C_\tau N^{\mathcal{G}} = C_{\text{inv}} C_\tau N^{\xi, tol_\xi}.$$

We have recovered the condition number bound in theorem 6.1, taking the improvement in (6.48) into account.

R–WB–OS–ACMS Coarse Space: Let $tol_{\mathcal{S}_{\mathcal{W}}} = \min_{s \in \mathcal{S}_{\mathcal{W}}} tol_s$, and let N^s be the maximum number of wire basket stars of a subdomain:

$$N^s := \max_{1 \leq i \leq N} |\mathcal{S}_{\mathcal{W}}(\Omega_i)|, \quad \mathcal{S}_{\mathcal{W}}(\Omega_i) := \{\, s \in \mathcal{S}_{\mathcal{W}} : s \cap \overline{\Omega}_i \neq \emptyset \,\}.$$

Furthermore, we define

$$N^{\partial f \to s} := \max_{1 \leq i \leq N} \sum_{s \in \mathcal{S}_{\mathcal{W}}(\Omega_i)} |\mathcal{F} \cap \mathcal{B}_{\mathrm{c}}(s)|,$$

where $\mathcal{F} \cap \mathcal{B}_{\mathrm{c}}(s) = \mathcal{B}_{\mathrm{c}}(s)$ is the set of faces adjacent to an $s \in \mathcal{S}_{\mathcal{W}}$. Let

$$N^{s,\Sigma} := \max_{1 \leq i \leq N} \sum_{s \in \mathcal{S}_{\mathcal{W}}(\Omega_i)} |n(s)|.$$

By definition (7.41), we have

$$N^{\mathcal{G},\Sigma} = \max_{1 \leq i \leq N} \sum_{\xi \in \mathcal{P}(\Omega_i)} \sum_{\substack{p \in \mathrm{path}(\mathcal{P},\xi) \\ \varphi := p_1}} n^\varphi (8^{|p|-1} g_\xi) C_{\mathrm{inv}}^{|p|} C_\tau^{|p|-1} \left(\prod_{j=1}^{|p|} \frac{1}{tol_{p_j}} \right),$$

where $n^\varphi = \max_{\varphi_i \in \mathcal{N}_\varphi} |n(\varphi_i)|$, and

$$g_\xi := \begin{cases} 1 & \text{if } \mathcal{B}_{\mathrm{A}}(\xi) = \emptyset, \\ 8 & \text{else.} \end{cases}$$

Since coarse faces do not have descendants, we have $\mathrm{path}(\mathcal{P}, f) = \{(f)\}$ for $f \in \mathcal{S}_{\mathcal{W}}$. Since the ancestors of coarse faces are given by adjacent wire basket stars, the paths in $\mathrm{path}(\mathcal{P}, s)$ are given by (s) and (f_i, s), where $f_i \in \mathcal{B}_{\mathrm{c}}(s)$. A face $f \in \mathcal{F}$ has two adjacent subdomains; thus, $n^f = 2$. Furthermore, we can use fact that $n^s \leq |n(s)|$ for $s \in \mathcal{S}_{\mathcal{W}}$. By $\mathcal{F}(\Omega_i)$ we denote the coarse faces of Ω_i and by $\mathcal{S}_{\mathcal{W}}(\Omega_i)$ wire basket stars that touch $\overline{\Omega}_i$. We obtain

$$N^{\mathcal{G},\Sigma} \leq \max_{1 \leq i \leq N} \left(\sum_{f \in \mathcal{F}(\Omega_i)} 16 \frac{C_{\mathrm{inv}}}{tol_f} + \sum_{s \in \mathcal{S}_{\mathcal{W}}(\Omega_i)} \left(|n(s)| \frac{C_{\mathrm{inv}}}{tol_s} + \sum_{f \in \mathcal{B}_{\mathrm{c}}(s)} \frac{16 C_{\mathrm{inv}}^2 C_\tau}{tol_f tol_s} \right) \right)$$

$$\leq \frac{16 C_{\mathrm{inv}} N^f}{tol_{\mathcal{F}}} + \frac{C_{\mathrm{inv}} N^{s,\Sigma}}{tol_{\mathcal{S}_{\mathcal{W}}}} + \frac{16 C_{\mathrm{inv}}^2 C_\tau N^{\partial f \to s}}{tol_{\mathcal{F}} tol_{\mathcal{S}_{\mathcal{W}}}}.$$

Similarly, we have for $N^{\mathcal{G}}$ in (7.40)

$$
\begin{aligned}
N^{\mathcal{G}} &= \max_{1 \le i \le N} \sum_{\xi \in \mathcal{P}(\Omega_i)} \sum_{p \in \text{path}(\mathcal{P}, \xi)} (8^{|p|-1} g_\xi) C_{\text{inv}}^{|p|} C_\tau^{|p|-1} \left(\prod_{j=1}^{|p|} \frac{1}{tol_{p_j}} \right) \\
&\le \frac{8 C_{\text{inv}} N^f}{tol_{\mathcal{F}}} + \frac{C_{\text{inv}} N^s}{tol_{\mathcal{S}_\mathcal{W}}} + \frac{8 C_{\text{inv}}^2 C_\tau N^{\partial f \to s}}{tol_{\mathcal{F}} tol_{\mathcal{S}_\mathcal{W}}}.
\end{aligned}
$$

Therefore, we obtain for D_0 and D_1 in (7.42) and (7.43)

$$
D_0 = C_\tau N^{\mathcal{G}, \Sigma} \le C_{\text{inv}} C_\tau \left(\frac{16 N^f}{tol_{\mathcal{F}}} + \frac{N^{s, \Sigma}}{tol_{\mathcal{S}_\mathcal{W}}} + \frac{16 C_{\text{inv}} C_\tau N^{\partial f \to s}}{tol_{\mathcal{F}} tol_{\mathcal{S}_\mathcal{W}}} \right),
$$

$$
D_1 = C_\tau N^{\mathcal{G}} \le C_{\text{inv}} C_\tau \left(\frac{8 N^f}{tol_{\mathcal{F}}} + \frac{N^s}{tol_{\mathcal{S}_\mathcal{W}}} + \frac{8 C_{\text{inv}} C_\tau N^{\partial f \to s}}{tol_{\mathcal{F}} tol_{\mathcal{S}_\mathcal{W}}} \right),
$$

which is identical to the result in theorem A.1; see (A.6) and (A.7).

OS–ACMS Coarse Space: The derivation of D_0 and D_1 is similar to the one for R–WB–OS–ACMS. We treat coarse edges similarly to wire basket stars but note that g^e is usually equal to 8 for $e \in \mathcal{E}$ (unlike $g^s = 1$ for $s \in \mathcal{S}_\mathcal{W}$), except if the coarse edge does not have any incident coarse nodes. The inequality $g^e \le 8$ always holds. Furthermore, we note that $tol_\nu = \infty$ for $\nu \in \mathcal{V}$. By $\mathcal{E}(\Omega_i)$ we denote the coarse edges of Ω_i and obtain

$$
\begin{aligned}
N^{\mathcal{G}, \Sigma} &\le \max_{1 \le i \le N} \left(\sum_{f \in \mathcal{F}(\Omega_i)} \frac{16 C_{\text{inv}}}{tol_f} + \sum_{e \in \mathcal{E}(\Omega_i)} \left(8|n(e)| \frac{C_{\text{inv}}}{tol_e} + \sum_{f \in \mathcal{B}_c(e)} \frac{8 \cdot 16 C_{\text{inv}}^2 C_\tau}{tol_e tol_f} \right) \right) \\
&\le 4 C_{\text{inv}} \left(\frac{4 N^f}{tol_{\mathcal{F}}} + \frac{2 N^{e, \Sigma}}{tol_{\mathcal{E}}} + \frac{32 C_{\text{inv}} C_\tau N^{\partial f \to e}}{tol_{\mathcal{F}} tol_{\mathcal{E}}} \right),
\end{aligned}
$$

and

$$
N^{\mathcal{G}} \le 4 C_{\text{inv}} \left(\frac{2 N^f}{tol_{\mathcal{F}}} + \frac{2 N^e}{tol_{\mathcal{E}}} + \frac{16 C_{\text{inv}} C_\tau N^{\partial f \to e}}{tol_{\mathcal{F}} tol_{\mathcal{E}}} \right).
$$

The constants $D_0 = C_\tau N^{\mathcal{G}, \Sigma}$ and $D_1 = C_\tau N^{\mathcal{G}}$ are then slightly larger than their counterparts in (7.27) and (7.28). The reason was given during the proof of lemma 7.8, below (7.38).

8. Numerical Results for Linear Elasticity

In the following, we show numerical results for various coarse spaces, the equations of linear elasticity, and model problems **(1)**–**(4)** from sections 2.1 to 2.4. Additional results are given in appendix B; see also the references below. We include results for the GenEO coarse space from [SDH+14a]; see also section 8.3 for a brief comparison of some major differences between the GenEO coarse space and our coarse spaces. We note that an overview of coarse space acronyms is given in the notation chapter.

The Poisson ratio is chosen as $\nu = 0.4$ and the body force as $f \equiv (\,1\,,1\,,1\,)$. For all methods, an overlap of two layers of finite elements is chosen. The regularization term for linear elasticity problems to compute an energy-minimizing extension is set to $10^{-15} K_{\text{diag}}$, where K_{diag} is the diagonal of the right-hand side used for the generalized eigenvalue problem; cf. section 4.4. The scaling factor $\hat{h}_T$ of the mass matrix variant is set to the radius of the largest insphere of $T \in \tau_h(\Omega)$.

We use the preconditioned conjugate gradient method with the convergence criterion

$$\frac{\|r^{(k)}\|_{l^2}}{\|r^{(0)}\|_{l^2}} < 10^{-8},$$

where $r^{(k)}$ is the kth unpreconditioned residual. The initial vector is set to the zero vector and the maximum number of iterations to $2\,000$. A condition number estimate is obtained after the last iteration, using the Lanczos method; cf. [Saa03, sect. 6.7.3]. We further note that the residual is updated recursively.

To compare the coarse spaces, we show results for the condition number $\kappa = \kappa_2(M^{-1}K)$, the number of iterations, the coarse space dimension $\dim V_0$, and the dimension of the coarse space relative to the size of the stiffness matrix K. For all model problems but **(4)**,

we furthermore show the number of coarse functions associated with coarse nodes ($\mathcal{V}$), edges ($\mathcal{E}$), faces ($\mathcal{F}$), and wire basket ($\mathcal{S}_{\mathcal{W}}$) and interface stars ($\mathcal{S}_{\Gamma}$), where the latter two are abbreviated by $\mathcal{S}$; that is,

$$\mathcal{S} \text{ refers to } \mathcal{S}_{\mathcal{W}} \text{ or } \mathcal{S}_{\Gamma}.$$

For the selection of eigenvectors, we always use the same tolerance for each interface component. For the 100 randomly generated coefficient functions from section 2.4, we state averages and maxima.

In all tables and for each method, we have marked the first row for which the number of iterations is below 100; for the randomly generated coefficient functions, the respective maximum must be smaller than 100.

8.1. GDSW-Type Coarse Spaces

In table 8.1, results for problem **(1)** of section 2.1 are shown. GDSW and RGDSW both do not converge within 2 000 iterations. Considering that the problem size is still fairly small with 132 651 finite element nodes and 125 subdomains, this clearly shows the need for adaptive coarse spaces. All adaptive coarse spaces achieve small numbers of iterations and condition numbers. For this problem, we especially note that the coarse space dimensions of the adaptive coares spaces are at most only slightly larger than that of GDSW, which shows that not many additional coarse functions are required to achieve fast convergence. This is supported by the fact that R–WB–AGDSW requires only a slightly smaller number of coarse functions than AGDSW, which differs from the results in section 4.5 for problems **(2)** and **(3)**, where a much more substantial decrease of the coarse space dimension was achieved using R–WB–AGDSW and RAGDSW.

The results in tables 8.2 and 8.3 for the model problems in sections 2.2 and 2.3 are, qualitatively, very similar to those in section 4.5; however, we do note that, using R–WB–AGDSW and RAGDSW, the reduction in the coarse space dimension is slightly diminished. Here, for model problem **(2)**, we obtain a decrease of 29.8% comparing the highlighted rows of AGDSW and R–WB–AGDSW, and 74.6% for RAGDSW. For the

method	*tol*	it.	κ	$\dim V_0$	($\mathcal{V}$, $\mathcal{E}$, $\mathcal{F}$, $\mathcal{S}$)	$\frac{\dim V_0}{\mathrm{dof}}$
GDSW	—	>2 000	$3.1 \cdot 10^5$	9 996	(1 257, 5 008, 3 731, —)	2.51%
RGDSW	—	>2 000	$3.6 \cdot 10^5$	2 610	(— , — , — , 2 610)	0.66%
AGDSW	0.005	113	277.3	11 316	(1 257, 3 271, 6 788, —)	2.84%
AGDSW	0.01	78	47.9	11 345	(1 257, 3 289, 6 799, —)	2.85%
AGDSW	0.1	67	34.2	11 859	(1 257, 3 448, 7 154, —)	2.98%
AGDSW–S	0.005	94	126.7	11 321	(1 257, 3 276, 6 788, —)	2.84%
AGDSW–S	0.01	78	48.3	11 352	(1 257, 3 296, 6 799, —)	2.85%
AGDSW–S	0.1	65	32.3	12 097	(1 257, 3 630, 7 210, —)	3.04%
R–WB–AGDSW	0.005	113	262.8	9 569	(— , — , 6 788, 2 781)	2.40%
R–WB–AGDSW	0.01	80	53.2	9 586	(— , — , 6 799, 2 787)	2.41%
R–WB–AGDSW	0.1	67	31.8	9 993	(— , — , 7 154, 2 839)	2.51%
R–WB–AGDSW–S	0.005	94	123.1	9 584	(— , — , 6 788, 2 796)	2.41%
R–WB–AGDSW–S	0.01	83	53.3	9 602	(— , — , 6 799, 2 803)	2.41%
R–WB–AGDSW–S	0.1	65	30.5	10 222	(— , — , 7 210, 3 012)	2.57%
RAGDSW	0.005	136	298.9	7 028	(— , — , — , 7 028)	1.77%
RAGDSW	0.01	81	62.5	7 036	(— , — , — , 7 036)	1.77%
RAGDSW	0.1	61	24.7	7 516	(— , — , — , 7 516)	1.89%
RAGDSW–S	0.005	93	207.6	7 055	(— , — , — , 7 055)	1.77%
RAGDSW–S	0.01	77	45.3	7 059	(— , — , — , 7 059)	1.77%
RAGDSW–S	0.1	59	23.6	7 718	(— , — , — , 7 718)	1.94%

Table 8.1.: **(Model problem (1))** Results for the coefficient function in fig. 2.2, the equations of linear elasticity, different methods and tolerances for the selection of eigenvectors: iteration count, condition number, resulting coarse space dimension, and coarse space dimension over the size of the stiffness matrix. The number of coarse functions associated with subdomain vertices, edges, faces, wire basket and interface stars is given in parentheses.

method	*tol*	it.	κ	dim V_0	($\mathcal{V}$,	$\mathcal{E}$,	$\mathcal{F}$,	$\mathcal{S}$)	$\frac{\dim V_0}{\text{dof}}$
GDSW	—	1 424	$1.5 \cdot 10^7$	2 319	(210,	1 081,	1 028,	—)	1.38%
RGDSW	—	1 606	$1.4 \cdot 10^7$	456	(—,	—,	—,	456)	0.27%
AGDSW	0.001	320	4 790.3	2 578	(210,	1 099,	1 269,	—)	1.53%
AGDSW	0.005	88	208.7	2 599	(210,	1 113,	1 276,	—)	1.55%
AGDSW	0.1	65	31.8	2 702	(210,	1 116,	1 376,	—)	1.61%
AGDSW–S	0.001	321	4 757.6	2 579	(210,	1 100,	1 269,	—)	1.53%
AGDSW–S	0.005	88	206.5	2 599	(210,	1 113,	1 276,	—)	1.55%
AGDSW–S	0.1	56	19.0	2 715	(210,	1 121,	1 384,	—)	1.61%
R–WB–AGDSW	0.001	222	1 502.6	1 813	(—,	—,	1 269,	544)	1.08%
R–WB–AGDSW	0.005	87	188.5	1 825	(—,	—,	1 276,	549)	1.09%
R–WB–AGDSW	0.1	60	23.2	1 925	(—,	—,	1 376,	549)	1.14%
R–WB–AGDSW–S	0.001	188	1 119.2	1 815	(—,	—,	1 269,	546)	1.08%
R–WB–AGDSW–S	0.005	88	187.1	1 825	(—,	—,	1 276,	549)	1.09%
R–WB–AGDSW–S	0.1	59	21.3	1 941	(—,	—,	1 384,	557)	1.15%
RAGDSW	0.001	126	517.7	658	(—,	—,	—,	658)	0.39%
RAGDSW	0.005	89	52.0	661	(—,	—,	—,	661)	0.39%
RAGDSW	0.1	65	24.7	849	(—,	—,	—,	849)	0.50%
RAGDSW–S	0.001	107	171.8	661	(—,	—,	—,	661)	0.39%
RAGDSW–S	0.005	85	52.0	662	(—,	—,	—,	662)	0.39%
RAGDSW–S	0.1	59	22.7	959	(—,	—,	—,	959)	0.57%

Table 8.2.: **(Model problem (2))** Results for the coefficient function in fig. 2.3, the equations of linear elasticity, different methods and tolerances for the selection of eigenvectors: iteration count, condition number, resulting coarse space dimension, and coarse space dimension over the size of the stiffness matrix. The number of coarse functions associated with subdomain vertices, edges, faces, wire basket and interface stars is given in parentheses.

diffusion problem, the numbers were 38.4% and 77.6%, respectively.

The reduction in the coarse space dimension in table 8.2 can mostly be attributed to that of RGDSW with respect to GDSW: the dimensions of AGDSW and RAGDSW are fairly close to those of GDSW and RGDSW, from which follows that only a small number of additional, adaptively computed coarse functions are required to obtain a robust preconditioner.

At last, we consider model problem **(4)** from section 2.4. In [HKK+22, table 6], results for GDSW and RGDSW are given: both methods never converge within 2 000 iterations. Note that, the RGDSW interface partition used in this work differs from the one in [HKK+22]; thus, we would obtain slightly different results. The results for the adaptive coarse spaces in table 8.4 indicate that this problem type is much more difficult. Larger tolerances are required to always stay below 100 iterations.

For problems **(1)** and **(4)**, the coarse space dimensions are significantly larger than for the other two problems. For the highlighted rows of RAGDSW, the coarse space dimension over the dimension of the finite element space is 1.77% for problem **(1)**, 2.4% for problem **(4)**, and 0.39% and 0.23% for problems **(2)** and **(3)**.

Let us note that the density of the coefficient functions of problem **(4)**—that is, the ratio of the number of elements with a large coefficient to the total number of elements—is 11.08% and was chosen to obtain a difficult problem. If the density were much larger, many connected structures of large coefficients would result in a smaller coarse space dimension. Similarly, if the density were much smaller, the coarse space dimension would decrease as well.

Similarly to problem **(1)**, for problem **(4)**, using a reduced-dimension coarse space, we do not obtain a decrease of the coarse space dimension that is as large as for the other two problems. This supports the claim that the inclusions of large coefficients do not form many large connected structures.

By comparing the results for the diffusion problems in section 4.5, we observe a more pronounced difference between the original and the S-variant. However, the differences are still small. A direct comparison is difficult as the spectrum is different for both

variants. Most of the time, the larger coarse space dimension of the S-variant results in a lower number of iterations.

Results for a variety of coarse spaces using lumped matrices are given in tables B.13, B.15, and B.17. Results for AGDSW–M are given in tables B.14, B.16, and B.18.

method	*tol*	it.	κ	dim V_0	($\mathcal{V}$,	$\mathcal{E}$,	$\mathcal{F}$,	$\mathcal{S}$)	$\frac{\dim V_0}{\text{dof}}$
GDSW	—	1 856	$1.1{\cdot}10^6$	8 311	(984,	4 267,	3 060,	—)	0.47%
RGDSW	—	1 634	$8.9{\cdot}10^5$	2 034	(—,	—,	—,	2 034)	0.12%
AGDSW	0.01	268	448.5	9 130	(984,	3 436,	4 710,	—)	0.52%
AGDSW	0.05	82	43.3	9 942	(984,	3 612,	5 346,	—)	0.56%
AGDSW	0.1	57	26.5	11 372	(984,	3 641,	6 747,	—)	0.64%
AGDSW–S	0.01	255	432.2	9 184	(984,	3 486,	4 714,	—)	0.52%
AGDSW–S	0.05	72	34.5	10 034	(984,	3 650,	5 400,	—)	0.57%
AGDSW–S	0.1	52	18.3	11 601	(984,	3 731,	6 886,	—)	0.66%
R–WB–AGDSW	0.01	203	360.8	6 847	(—,	—,	4 710,	2 137)	0.39%
R–WB–AGDSW	0.03	112	108.1	7 125	(—,	—,	4 950,	2 175)	0.40%
R–WB–AGDSW	0.05	77	46.0	7 540	(—,	—,	5 346,	2 194)	0.43%
R–WB–AGDSW	0.1	54	20.8	8 970	(—,	—,	6 747,	2 223)	0.51%
R–WB–AGDSW–S	0.01	176	300.2	6 877	(—,	—,	4 714,	2 163)	0.39%
R–WB–AGDSW–S	0.03	92	81.1	7 185	(—,	—,	4 969,	2 216)	0.41%
R–WB–AGDSW–S	0.05	73	39.9	7 648	(—,	—,	5 400,	2 248)	0.43%
R–WB–AGDSW–S	0.1	52	18.2	9 331	(—,	—,	6 886,	2 445)	0.53%
RAGDSW	0.01	160	183.7	3 745	(—,	—,	—,	3 745)	0.21%
RAGDSW	0.03	96	60.6	4 031	(—,	—,	—,	4 031)	0.23%
RAGDSW	0.1	55	23.9	6 430	(—,	—,	—,	6 430)	0.36%
RAGDSW–S	0.01	135	138.8	3 854	(—,	—,	—,	3 854)	0.22%
RAGDSW–S	0.03	93	56.1	4 429	(—,	—,	—,	4 429)	0.25%
RAGDSW–S	0.1	51	16.9	7 470	(—,	—,	—,	7 470)	0.42%

Table 8.3.: **(Model problem (3))** Results for the coefficient function in fig. 2.4, the equations of linear elasticity, different methods and tolerances for the selection of eigenvectors: iteration count, condition number, resulting coarse space dimension, and coarse space dimension over the size of the stiffness matrix. The number of coarse functions associated with subdomain vertices, edges, faces, wire basket and interface stars is given in parentheses.

method	*tol*	it.	κ	dim V_0	$\frac{\dim V_0}{\text{dof}}$
AGDSW	0.01	244.5 (278)	532.0 (1 094.8)	49 156.4 (50 199)	3.6% (3.7%)
AGDSW	0.03	148.0 (165)	183.7 (312.7)	51 094.1 (52 278)	3.8% (3.9%)
AGDSW	0.05	117.3 (132)	114.3 (262.5)	52 582.4 (53 754)	3.9% (4.0%)
AGDSW	0.1	86.9 (96)	59.0 (84.3)	55 879.1 (57 178)	4.1% (4.2%)
AGDSW–S	0.01	222.3 (258)	452.8 (860.5)	50 163.3 (51 421)	3.7% (3.8%)
AGDSW–S	0.03	134.4 (150)	161.4 (310.6)	52 329.3 (53 676)	3.9% (4.0%)
AGDSW–S	0.05	106.3 (121)	94.5 (163.6)	54 151.8 (55 526)	4.0% (4.1%)
AGDSW–S	0.1	78.1 (87)	47.7 (84.5)	58 471.6 (59 950)	4.3% (4.4%)
R–WB–AGDSW	0.01	244.1 (274)	521.0 (798.1)	37 670.6 (38 600)	2.8% (2.8%)
R–WB–AGDSW	0.03	147.4 (168)	177.7 (308.8)	39 601.3 (40 577)	2.9% (3.0%)
R–WB–AGDSW	0.05	117.2 (131)	111.1 (195.1)	41 071.7 (42 057)	3.0% (3.1%)
R–WB–AGDSW	0.1	90.1 (99)	62.2 (90.9)	44 348.1 (45 477)	3.3% (3.3%)
R–WB–AGDSW–S	0.01	216.7 (246)	427.7 (786.3)	39 011.0 (40 069)	2.9% (3.0%)
R–WB–AGDSW–S	0.03	131.9 (145)	149.9 (273.2)	41 172.6 (42 298)	3.0% (3.1%)
R–WB–AGDSW–S	0.05	105.9 (120)	89.3 (155.5)	42 986.2 (44 169)	3.2% (3.3%)
R–WB–AGDSW–S	0.1	78.8 (85)	47.9 (78.4)	47 451.3 (48 793)	3.5% (3.6%)
RAGDSW	0.01	217.8 (247)	422.8 (1 015.2)	25 605.4 (26 304)	1.9% (1.9%)
RAGDSW	0.03	135.9 (153)	150.5 (299.7)	27 551.8 (28 219)	2.0% (2.1%)
RAGDSW	0.05	111.9 (127)	97.0 (139.7)	29 055.2 (29 766)	2.1% (2.2%)
RAGDSW	0.1	85.0 (96)	54.3 (82.1)	32 546.7 (33 253)	2.4% (2.4%)
RAGDSW–S	0.01	196.2 (231)	359.5 (808.1)	27 455.6 (28 099)	2.0% (2.1%)
RAGDSW–S	0.03	122.0 (145)	128.8 (288.4)	29 642.5 (30 293)	2.2% (2.2%)
RAGDSW–S	0.05	99.9 (121)	79.9 (134.2)	31 492.0 (32 177)	2.3% (2.4%)
RAGDSW–S	0.1	75.1 (84)	42.7 (74.7)	36 193.4 (36 852)	2.7% (2.7%)

Table 8.4.: **(Model problem (4))** Average results (maximum in parentheses) for 100 randomly generated coefficient functions (cf. section 2.4), the equations of linear elasticity, different methods and tolerances for the selection of eigenvectors: iteration count, condition number, resulting coarse space dimension, and coarse space dimension over the size of the stiffness matrix.

8.2. ACMS-Type Coarse Spaces

In this section, we examine the coarse spaces OS–ACMS–K, R–WB–OS–ACMS–K, and their S-variants, and focus on the coarse space reduction. For comparison, we have included results of RAGDSW and GenEO. Results for the variants of OS–ACMS and R–WB–OS–ACMS using a mass matrix can be found in tables B.14, B.16, and B.18 to B.20. Results for a variety of coarse spaces using lumped matrices are given in tables B.13, B.15, and B.17.

Using the R–WB–OS–ACMS coarse space, we can achieve a further reduction in the coarse space dimension with respect to RAGDSW; see tables 8.5 to 8.8. For problem **(1)**, we obtain a reduction of 36.7% if $tol = 0.05$ is used for RAGDSW and $tol = 0.01$ for R–WB–OS–ACMS (in which case the numbers of iterations are identical, and the condition numbers are similar). For problems **(2)**–**(4)**, we compare the highlighted rows and obtain reductions of 11.0% (while also converging faster), 16.6%, and 23.9%, respectively. For OS–ACMS, the results vary depending on the problem considered. Apart from the coarse space dimension, the results are similar to those of GDSW-type coarse spaces in the previous section.

We have included results for the GenEO coarse space in tables 8.5 to 8.7. In all cases, the coarse space dimensions of R–WB–OS–ACMS are smaller than that of GenEO if convergence is to be achieved in less than or close to 100 iterations. Let us reiterate our previous remark, however, that a comparison by sole means of the coarse space dimension does not allow us to draw conclusions about the general performance.

The results of this chapter indicate that choosing an optimal tolerance for the selection of eigenvectors that is equally suitable for all model problems appears to be difficult, especially if it is chosen to be identical for all types of interface components. It may be possible to choose the tolerance automatically, for example, based on the ratio of the diameter of Ω_ξ to the finite element diameter, which would essentially incorporate the well known dependence of the condition number on H/h.

method	*tol*	it.	κ	$\dim V_0$	($\mathcal{V}$,	$\mathcal{E}$,	$\mathcal{F}$,	$\mathcal{S}$)	$\frac{\dim V_0}{\text{dof}}$
RAGDSW	0.001	438	2 513.2	6 920	(— ,	— ,	— ,	6 920)	1.74%
RAGDSW	0.01	81	62.5	7 036	(— ,	— ,	— ,	7 036)	1.77%
RAGDSW	0.05	67	34.6	7 156	(— ,	— ,	— ,	7 156)	1.80%
OS–ACMS–K	0.001	581	5 986.8	5 220	(1 257,	2 427,	1 536,	—)	1.31%
OS–ACMS–K	0.005	91	1 283.4	5 389	(1 257,	2 503,	1 629,	—)	1.35%
OS–ACMS–K	0.01	64	34.1	5 504	(1 257,	2 505,	1 742,	—)	1.38%
OS–ACMS–K	0.05	54	20.3	6 123	(1 257,	2 516,	2 350,	—)	1.54%
OS–ACMS–S–K	0.001	424	2 455.5	5 258	(1 257,	2 465,	1 536,	—)	1.32%
OS–ACMS–S–K	0.005	65	34.1	5 393	(1 257,	2 507,	1 629,	—)	1.36%
OS–ACMS–S–K	0.05	54	19.9	6 142	(1 257,	2 535,	2 350,	—)	1.54%
R–WB–OS–ACMS–K	0.001	621	4 753.0	4 247	(— ,	— ,	1 536,	2 711)	1.07%
R–WB–OS–ACMS–K	0.005	94	1 177.1	4 410	(— ,	— ,	1 629,	2 781)	1.11%
R–WB–OS–ACMS–K	0.01	67	32.5	4 529	(— ,	— ,	1 742,	2 787)	1.14%
R–WB–OS–ACMS–K	0.05	61	25.5	5 153	(— ,	— ,	2 350,	2 803)	1.29%
R–WB–OS–ACMS–S–K	0.001	467	5 197.4	4 292	(— ,	— ,	1 536,	2 756)	1.08%
R–WB–OS–ACMS–S–K	0.005	69	32.7	4 425	(— ,	— ,	1 629,	2 796)	1.11%
R–WB–OS–ACMS–S–K	0.05	59	23.7	5 235	(— ,	— ,	2 350,	2 885)	1.32%
GenEO	0.01	176	260.7	6 819		—			1.71%
GenEO	0.1	122	120.1	7 386		—			1.86%
GenEO	0.2	96	76.4	8 280		—			2.08%
GenEO	0.3	75	45.8	9 205		—			2.31%

Table 8.5.: **(Model problem (1))** Results for the coefficient function in fig. 2.2, the equations of linear elasticity, different methods and tolerances for the selection of eigenvectors: iteration count, condition number, resulting coarse space dimension, and coarse space dimension over the size of the stiffness matrix. The number of coarse functions associated with subdomain vertices, edges, faces, wire basket and interface stars is given in parentheses.

method	*tol*	it.	κ	$\dim V_0$	($\mathcal{V}$,	$\mathcal{E}$,	$\mathcal{F}$,	$\mathcal{S}$)	$\frac{\dim V_0}{\text{dof}}$
RAGDSW	0.001	126	517.7	658	(— ,	— ,	— ,	658)	0.39%
RAGDSW	0.005	89	52.0	661	(— ,	— ,	— ,	661)	0.39%
RAGDSW	0.05	79	37.8	673	(— ,	— ,	— ,	673)	0.40%
OS–ACMS–K	0.001	299	4 523.6	815	(210,	581,	24,	—)	0.48%
OS–ACMS–K	0.005	77	100.3	850	(210,	601,	39,	—)	0.51%
OS–ACMS–K	0.05	61	26.8	989	(210,	607,	172,	—)	0.59%
OS–ACMS–S–K	0.001	275	4 597.7	818	(210,	584,	24,	—)	0.49%
OS–ACMS–S–K	0.005	61	26.8	851	(210,	602,	39,	—)	0.51%
OS–ACMS–S–K	0.05	61	26.8	994	(210,	612,	172,	—)	0.59%
R–WB–OS–ACMS–K	0.001	181	4 319.9	568	(— ,	— ,	24,	544)	0.34%
R–WB–OS–ACMS–K	0.005	62	26.7	588	(— ,	— ,	39,	549)	0.35%
R–WB–OS–ACMS–K	0.05	61	26.7	721	(— ,	— ,	172,	549)	0.43%
R–WB–OS–ACMS–S–K	0.001	127	2 211.0	570	(— ,	— ,	24,	546)	0.34%
R–WB–OS–ACMS–S–K	0.005	61	26.7	588	(— ,	— ,	39,	549)	0.35%
R–WB–OS–ACMS–S–K	0.05	61	26.7	721	(— ,	— ,	172,	549)	0.43%
GenEO	0.01	194	243.5	471		—			0.28%
GenEO	0.1	122	80.5	622		—			0.37%
GenEO	0.2	86	40.3	931		—			0.55%

Table 8.6.: **(Model problem (2))** Results for the coefficient function in fig. 2.3, the equations of linear elasticity, different methods and tolerances for the selection of eigenvectors: iteration count, condition number, resulting coarse space dimension, and coarse space dimension over the size of the stiffness matrix. The number of coarse functions associated with subdomain vertices, edges, faces, wire basket and interface stars is given in parentheses.

8.3. Practical Aspects of the GenEO Coarse Space

As we have mentioned before, a thorough comparison of coarse spaces is out of the scope of this work. Nevertheless, we will mention some differences between the GenEO coarse

method	tol	it.	κ	$\dim V_0$	($\mathcal{V}$,	$\mathcal{E}$,	$\mathcal{F}$,	$\mathcal{S}$)	$\frac{\dim V_0}{\text{dof}}$
RAGDSW	0.01	160	183.7	3 745	(— ,	— ,	— ,	3 745)	0.21%
RAGDSW	0.03	96	60.6	4 031	(— ,	— ,	— ,	4 031)	0.23%
RAGDSW	0.1	55	23.9	6 430	(— ,	— ,	— ,	6 430)	0.36%
OS–ACMS–K	0.01	282	570.6	3 850	(984,	2 321,	545,	—)	0.22%
OS–ACMS–K	0.03	113	117.4	4 626	(984,	2 456,	1 186,	—)	0.26%
OS–ACMS–K	0.05	73	35.7	5 335	(984,	2 525,	1 826,	—)	0.30%
OS–ACMS–S–K	0.01	250	445.1	3 899	(984,	2 370,	545,	—)	0.22%
OS–ACMS–S–K	0.03	93	71.1	4 660	(984,	2 490,	1 186,	—)	0.26%
OS–ACMS–S–K	0.05	68	32.2	5 405	(984,	2 594,	1 827,	—)	0.31%
R–WB–OS–ACMS–K	0.01	182	317.0	2 682	(— ,	— ,	545,	2 137)	0.15%
R–WB–OS–ACMS–K	0.03	93	66.9	3 361	(— ,	— ,	1 186,	2 175)	0.19%
R–WB–OS–ACMS–K	0.05	71	39.2	4 020	(— ,	— ,	1 826,	2 194)	0.23%
R–WB–OS–ACMS–S–K	0.01	163	351.2	2 708	(— ,	— ,	545,	2 163)	0.15%
R–WB–OS–ACMS–S–K	0.03	75	38.6	3 402	(— ,	— ,	1 186,	2 216)	0.19%
R–WB–OS–ACMS–S–K	0.05	64	32.1	4 075	(— ,	— ,	1 827,	2 248)	0.23%
GenEO	10^{-4}	184	428.6	2 570		—			0.15%
GenEO	0.03	101	101.4	3 576		—			0.20%
GenEO	0.05	86	64.9	4 219		—			0.24%
GenEO	0.1	69	41.2	5 672		—			0.32%

Table 8.7.: **(Model problem (3))** Results for the coefficient function in fig. 2.4, the equations of linear elasticity, different methods and tolerances for the selection of eigenvectors: iteration count, condition number, resulting coarse space dimension, and coarse space dimension over the size of the stiffness matrix. The number of coarse functions associated with subdomain vertices, edges, faces, wire basket and interface stars is given in parentheses.

space [SDH+14a] and ours, which can further the understanding of the numerical results of the previous section.

method	*tol*	it.	κ	$\dim V_0$	$\frac{\dim V_0}{\text{dof}}$
RAGDSW	0.01	217.8 (247)	422.8 (1 015.2)	25 605.4 (26 304)	1.9% (1.9%)
RAGDSW	0.03	135.9 (153)	150.5 (299.7)	27 551.8 (28 219)	2.0% (2.1%)
RAGDSW	0.05	111.9 (127)	97.0 (139.7)	29 055.2 (29 766)	2.1% (2.2%)
RAGDSW	0.1	85.0 (96)	54.3 (82.1)	32 546.7 (33 253)	2.4% (2.4%)
OS–ACMS–K	0.01	227.3 (283)	625.9 (2 224.3)	21 569.9 (22 093)	1.6% (1.6%)
OS–ACMS–K	0.03	133.7 (155)	173.0 (356.3)	23 678.2 (24 165)	1.7% (1.8%)
OS–ACMS–K	0.05	104.5 (121)	97.3 (296.4)	25 407.8 (25 936)	1.9% (1.9%)
OS–ACMS–K	0.1	76.0 (88)	48.4 (82.3)	29 311.4 (29 925)	2.2% (2.2%)
OS–ACMS–S–K	0.01	195.8 (235)	506.2 (1 795.2)	22 287.0 (22 806)	1.6% (1.7%)
OS–ACMS–S–K	0.03	118.8 (139)	146.1 (368.5)	24 460.2 (24 977)	1.8% (1.8%)
OS–ACMS–S–K	0.05	93.0 (111)	79.1 (144.9)	26 298.2 (26 856)	1.9% (2.0%)
OS–ACMS–S–K	0.1	68.3 (81)	39.6 (83.0)	30 615.3 (31 256)	2.3% (2.3%)
R–WB–OS–ACMS–K	0.01	235.4 (283)	694.4 (2 652.2)	17 298.0 (17 767)	1.3% (1.3%)
R–WB–OS–ACMS–K	0.03	138.8 (159)	177.8 (335.7)	19 419.7 (19 905)	1.4% (1.5%)
R–WB–OS–ACMS–K	0.05	108.5 (123)	102.2 (200.8)	21 096.5 (21 591)	1.6% (1.6%)
R–WB–OS–ACMS–K	0.1	79.8 (91)	51.6 (85.5)	24 770.4 (25 361)	1.8% (1.9%)
R–WB–OS–ACMS–S–K	0.01	193.0 (235)	509.2 (3 801.9)	18 233.9 (18 726)	1.3% (1.4%)
R–WB–OS–ACMS–S–K	0.03	117.7 (139)	139.6 (328.6)	20 511.3 (21 057)	1.5% (1.6%)
R–WB–OS–ACMS–S–K	0.05	93.5 (108)	79.0 (197.6)	22 440.6 (23 007)	1.7% (1.7%)
R–WB–OS–ACMS–S–K	0.1	69.4 (81)	40.9 (73.5)	26 993.8 (27 625)	2.0% (2.0%)

Table 8.8.: **(Model problem (4))** Average results (maximum in parentheses) for 100 randomly generated coefficient functions (cf. section 2.4), the equations of linear elasticity, different methods and tolerances for the selection of eigenvectors: iteration count, condition number, resulting coarse space dimension, and coarse space dimension over the size of the stiffness matrix.

With each subdomain, the GenEO coarse space associates a single generalized eigenvalue problem, which is advantageous for parallelization. The eigenvalue problems are defined on the subdomain overlap and, as a result, are significantly larger than those associated

problem		$\mathcal{S}_\Gamma$	$\mathcal{F}$	$\mathcal{S}_\mathcal{W}$	$\mathcal{E}$	$\mathcal{V}$	GenEO
	# evp	76	172	76	199	70	50
(2)	mean dof(evp)	334.7	131.2	37.9	13.4	3	2 177.5
	max dof(evp)	645	432	84.0	39.0	3	3 078
	sum(dof(evp))	25 437	22 560	2 877	2 667	210	108 873
	# evp	339	515	339	750	328	100
(3)	mean dof(evp)	576.2	344.9	52.3	22.3	3	8 697.1
	max dof(evp)	1 566	1 140	156	78.0	3	12 111
	sum(dof(evp))	195 339	177 624	17 715	16 731	984	869 712

Table 8.9.: Number of eigenvalue problems, mean and maximum number of degrees of freedom per eigenvalue problem, and sum of degrees of freedom per eigenvalue problem for problems **(2)** and **(3)**: interface stars, coarse faces, wire basket stars, coarse edges, coarse nodes, GenEO eigenvalue problem. The sum of degrees of freedom of $\mathcal{S}_\Gamma$ equals $3n$, where n is the number of interface nodes.

with, for example, interface stars. In table 8.9, for linear elasticity and problems **(2)** and **(3)**, the mean and maximum number of degrees of freedom associated with interface components and those for the GenEO eigenvalue problems are given. Especially for problem **(3)**, the differences are substantial: the mean number of degrees of freedom of an interface star is 15 times smaller; the maximum is smaller by factor of 7.7. Because of the cubic complexity of direct eigensolvers, we can expect the computational cost to be significantly larger. As a result, smaller subdomains have to be used, which can increase the coarse space dimension. However, iterative eigensolvers, such as LOBPCG [Kny01; DSYG18], may be able to support the use of larger eigenvalue problems. In that case, suitable preconditioners are usually required to obtain fast convergence; cf. [Sou10].

On the other hand, using eigenvalue problems on large and geometrically compact domains can facilitate the detection of connected patches of large coefficients and, as a result, reduce the coarse space dimension.

Table 8.9 additionally shows that the sum of degrees of freedom of eigenvalue problems equals the number of degrees of freedem of Γ^h for GDSW- and ACMS-type coarse spaces. For the GenEO coarse space, the sum is four to five times as large. As a result, we can expect it to construct coarse functions that are not necessary for robustness. Indeed, consider a channel of a large coefficient intersecting a coarse edge as in fig. 1.5 (left). Then both subdomains adjacent to the coarse edge will construct a coarse function, although only one function is required.

9. Approaches for Reducing the Computational Cost

Adaptive coarse spaces that rely on the solution of local generalized eigenvalue problems can be considered as a last resort if all standard approaches fail to obtain a solution in a reasonable time frame. In particular, robust two-level methods using adaptive coarse spaces are a parallelizable alternative to direct solvers. A major drawback of the presented adaptive coarse spaces is the substantial cost for the setup and solution of the local generalized eigenvalue problems. Therefore, it is of interest to reduce the computational cost of the coarse space setup to make an application more feasible in practice.

The eigenvalue problems themselves are of moderate size, and preliminary results suggest that the cost can mostly be attributed to the setup of the associated Schur complements. Note that it is sufficient to solve standard eigenvalue problems if the variants using lumped matrices are used (cf. sections 3.3.1 and 3.3.4); however, this does not avoid the setup of Schur complements.

9.1. Avoiding the Setup of Schur Complements

Although the eigenvalue problems are of moderate size, in theory, an iterative eigensolver can be employed to avoid the setup of a Schur complement. In [HKKR18b], LOBPCG [Kny01; DSYG18] was used to solve the generalized eigenvalue problems; see also [Sou10; KKR16]. However, it appears that a good preconditioner for the left-hand side is required; cf. [Sou10]. For example, in [HKKR18b], a cholesky decomposition of the Schur complement was used.

Many realistic, heterogeneous problems exhibit a fairly simple structure of the coefficient function. For a significant amount of generalized eigenvalue problems, only the null space functions are then selected; for example, using AGDSW with a tolerance of 10^{-3} for problem **(2)** in section 2.2, only 18 of 199 coarse edges and 43 of 172 coarse faces are associated with more than one coarse function in the case of a diffusion problem. Thus, if the null space functions are available, it is not necessary to set up and compute solutions to many eigenvalue problems. Moreover, each subdomain is usually associated with a considerable amount of eigenvalue problems (see fig. A.1), which is a hindrance for parallel efficiency. Therefore, if the setup and solution of a large amount of eigenvalue problems can be avoided, the overall computational cost of the adaptive coarse space may be reduced considerably. Two simple approaches have been mentioned previously for adaptive FETI-DP and BDDC: the FETI-DP residual after one step [KRR15] or jumps in the coefficient function [KKR17; KKR18a] were used to decide which eigenvalue problems can be omitted. In [KKR20], heuristic analyses of the diagonal entries of the stiffness matrix avoid the solution of redundant generalized eigenvalue problems in a parallel implementation of adaptive FETI-DP; cf. section 9.2 and [Kne16; Hei16].

In [HKLW21b; HKLW21c], the authors use deep neural networks to predict with great accuracy whether an eigenvalue problem has to be solved; see also [HKLW19; HKLW21a]. For example, they consider a steel microsection as the coefficient function, and they can reduce the number of eigenvalue problems required by AGDSW to be solved from 112 to 27, while obtaining essentially identical results with respect to the number of iterations and the condition number. To achieve this, the authors formulate a classification problem and carefully craft a training dataset that, although small, contains the difficult features that coefficient functions exhibit. The coefficient function is then sampled on a slab surrounding a coarse edge to determine whether an eigenvalue problem should be solved.

In the following section, we consider an approach to heuristically construct coarse functions and avoid the setup and solution of generalized eigenvalue problems. We restrict ourselves to diffusion problems. Related, albeit requiring preliminary fixing a coarse space dimension, is the nonadaptive coarse space NSHEM from [GLR15].
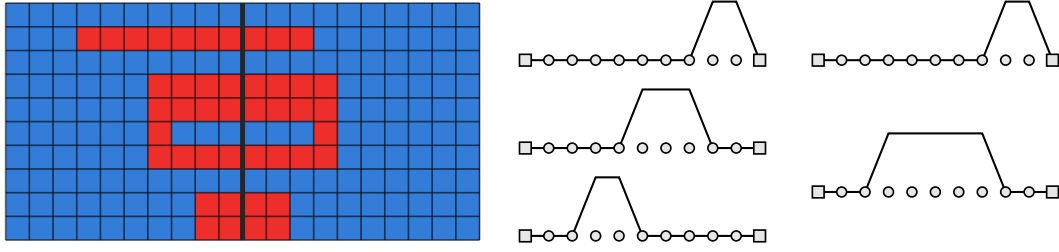
Figure 9.1.: **(Left)** Coefficient function with a coefficient of $E = 10^6$ in red and $E = 1$ in blue. Finite element nodes of the coarse edge and incident coarse nodes are marked. **(Center)** Constructed edge functions using the heuristic approach from [Kne16; Hei16; HKKR18b]. **(Right)** Constructed edge functions using an improved variant of the heuristic approach.

9.2. Heuristic Coarse Space Construction for Diffusion Problems

In [Kne16; Hei16; HKKR18b]—for two-dimensional diffusion problems on structured grids and domain decompositions—the authors presented an approach for the heuristic construction of coarse functions. By the observation of eigenfunctions of edge eigenproblems—cf. figs. 1.8, 1.9, 3.2, 7.1, and 7.2—the similarities between the coefficient functions and eigenfunctions corresponding to small eigenvalues become evident. In [Kne16; Hei16; HKKR18b], edge eigenfunctions are approximated heuristically by partitioning the coefficient function on the edge with respect to small and large coefficients (given a tolerance). For each individual part, a coarse function is constructed by setting the edge function to 1 on the section associated with a large coefficient and to 0 elsewhere; see fig. 9.1 (left and center) and [HKKR18b, fig. 4.2]. Note that, if a patch of large coefficients touches a coarse node or the Dirichlet boundary $\partial\Omega_D$, a coarse edge function will not be constructed; cf. section 7.2. Instead, MsFEM-type functions from the ACMS space are associated with coarse nodes. In [HKLW20], a similar approach was used to construct a nonadaptive FETI-DP and BDDC coarse space that approximates the most relevant eigenvector of each generalized eigenvalue problem; see also [KLW22].

To reduce the coarse space dimension, this approach can be improved by first considering connected components of patches of large coefficients on Ω_e. For the same reason, we
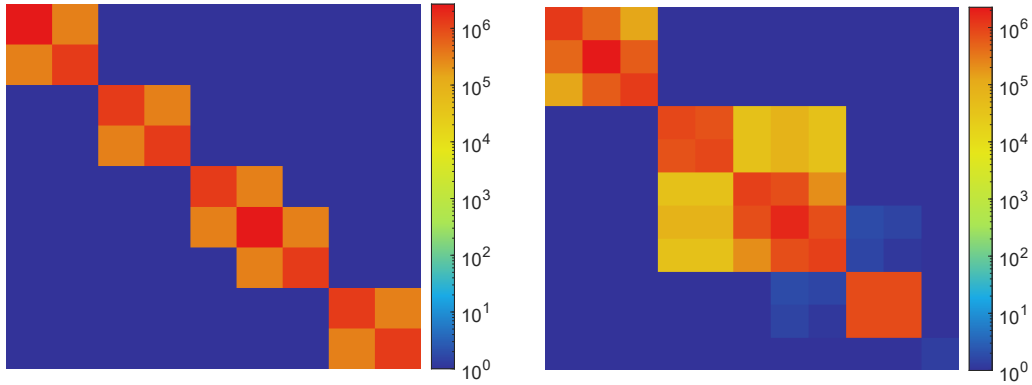
Figure 9.2.: Corresponding to fig. 9.1 (left): submatrix of the fully assembled stiffness matrix K corresponding to the coarse edge including its boundary nodes **(left)** and Schur complement corresponding to $\bar{e}$ and Ω_e **(right)**. Absolute values are displayed; absolute values below 1 are set to 1.

have included the energy-minimizing extension in the setup of the generalized eigenvalue problems of our standard adaptive coarse spaces. The constructed edge functions corresponding to the coefficient function in fig. 9.1 (left) are shown in the same figure on the right. We note that this also requires the new type of MsFEM function from section 7.2 to be used.

For the heuristic approach, we have assumed access to the coefficient function. If a structured mesh and domain decomposition are used, we can easily use the entries of the stiffness matrix instead; cf. fig. 9.2 (left). If φ_i and φ_j are the bilinear basis functions of two adjacent nodes on the coarse edge, then $a_\Omega(\varphi_i, \varphi_j)$ indicates whether the coefficient function has a large value between φ_i and φ_j. For unstructured meshes and domain decompositions, the situation is more complex. Furthermore, elements are of different size and shape, such that $a_\Omega(\varphi_i, \varphi_j)$ can vary significantly.

One of the main goals of this work was to present algorithms that lead to small coarse space dimensions. Using the original heuristic approach described above is similar to using AGDSW with a slab of only one layer of finite elements around the coarse edge. We have seen in section 3.4.1 that this can result in a significant increase in the coarse space dimension. Therefore, we would like to use the (improved) heuristic variant of AGDSW

for larger slabs. If the coefficient function is not available for the construction of edge functions, in order to detect connected patches of large coefficients, we can construct edge functions by partitioning the Schur complement associated with $\overline{e}$; cf. fig. 9.2 (right). However, as we expect the setup of the Schur complement to be the dominating cost of the generalized eigenvalue problems, using a heuristic approach to reduce the computational cost seems futile if we rely on the setup of the Schur complement.

We now give a heuristic approach that neither requires the setup of Schur complements nor access to local Neumann stiffness matrices. We restrict ourselves to the description of heuristic adaptive GDSW-type coarse spaces of the S-variant. Note, however, that the idea is similar for standard adaptive GDSW-type and ACMS-type coarse spaces. In the latter case, the construction is more involved as we not only need to construct interface functions but also suitable heuristic, cascaded energy-minimizing extensions.

Let $\xi \in \mathcal{P}$, $k \in n(\xi)$, and let $\tilde{K}^{\Omega_k} = (\tilde{K}^{\Omega_k}_{ij})_{ij}$ be a local stiffness matrix associated with one of the subdomains adjacent to ξ. Specifically, $\tilde{K}^{\Omega_k}$ is the submatrix of the fully assembled stiffness matrix K corresponding to the finite element nodes of $\overline{\Omega}_k$. We define

$$P := (P_{ij})_{ij}, \quad P_{ij} := \begin{cases} 1 & \text{if } |\tilde{K}^{\Omega_k}_{ij}| > tol_\xi, \\ 0 & \text{else,} \end{cases}$$

where tol_ξ is the heuristic analogue of the tolerance for the selection of eigenfunctions. Let E_ξ be the set of nodes of $\overline{\xi}$ that belong to a finite element T with a large coefficient (in the case of GDSW-type coarse spaces, we had defined $\partial \xi := \emptyset$, such that $\overline{\xi} = \xi$). We can gather this information by extracting the respective diagonal entries of the fully assembled stiffness matrix K. For each node in E_ξ, we define a vector v_i to be the unit vector that is equal to 1 at the selected node and zero on all other nodes of $\overline{\Omega}_k$. By iterating

$$v_i \leftarrow P v_i$$

l times (or until no change in the pattern is observed), the single nonzero value will spread to other finite element nodes that are connected via large coefficients and that are at most at a distance of l layers of finite elements. We call l the heuristic slab width.

After the last iteration, we set all nonzero values to 1. After l steps, additional nodes on ξ may be nonzero, in which case we have detected a connected patch of large coefficients that touches or intersects ξ multiple times. We repeat the process of detecting connected components for all subdomains adjacent to ξ and then merge all detected components. Note that this essentially approximates the S-variant of an adaptive GDSW-type coarse space, since connections are only detected inside subdomains or via ξ.

To obtain a scalable method, we need to make sure that the null space of the underlying problem—that is, the constant functions—can be represented by the coarse space. Since we have constructed binary vectors, this can be determined by computing the sum of all constructed vectors associated with ξ. In case the answer is negative, we can simply replace any of the vectors with the vector corresponding to a constant, nonzero function or—in case there are no coarse functions associated with ξ at all—we add the constant function. The remainder is identical to the construction of our standard adaptive GDSW-type coarse spaces: we extend the functions on ξ by zero to Γ^h and then energy-minimally to the interior of the subdomains.

A difficulty is presented by the requirement to select a tolerance tol_ξ since the entries of the stiffness matrix depend on the size and shape of the elements. In general, tol_ξ must be adapted to each element, depending on its shape and size. The finite elements of the meshes in chapter 2 are sufficiently similar and, furthermore, we use a large coefficient contrast of 10^6, which results in a clear separation of the entries of the stiffness matrix. Choosing a tolerance for these problems is, therefore, not difficult.

In table 9.1, numerical results for problems **(1)**–**(3)**, RAGDSW–S, and its heuristic variant are given. We observe almost identical coarse space dimensions between RAGDSW–S and its heuristic variant if a large slab of 100 is used. Note that $l = 100$ for the slab can be regarded as using the maximum slab; the algorithm can terminate well before reaching 100 iterations if no change is observed.

Furthermore, the results show that the heuristic variant can require considerably more iterations to converge; the largest increase of 59% is obtained for problem **(1)**. This should be expected as the constructed interface functions only approximate eigenfunctions.

E	method	*tol*	it.	κ	$\dim V_0$	$\frac{\dim V_0}{\text{dof}}$
	RAGDSW–S	0.01	59	32.0	1 289	0.97%
	RAGDSW–S[H(100)]	100	94	65.7	1 288	0.97%
(1)	RAGDSW–S[H(10)]	100	94	65.7	1 288	0.97%
	RAGDSW–S[H(3)]	100	95	66.1	1 323	1.00%
	RAGDSW–S[H(1)]	100	93	68.6	2 246	1.69%
	RAGDSW–S	0.01	56	24.2	112	0.20%
	RAGDSW–S[H(100)]	1 000	61	29.5	112	0.20%
(2)	RAGDSW–S[H(10)]	1 000	61	29.5	113	0.20%
	RAGDSW–S[H(3)]	1 000	61	29.7	123	0.22%
	RAGDSW–S[H(1)]	1 000	64	28.7	133	0.24%
	RAGDSW–S	0.01	59	33.7	663	0.11%
	RAGDSW–S[H(100)]	100	75	41.5	661	0.11%
(3)	RAGDSW–S[H(10)]	100	75	41.6	678	0.12%
	RAGDSW–S[H(3)]	100	74	39.3	977	0.17%
	RAGDSW–S[H(1)]	100	73	37.5	1 399	0.24%

Table 9.1.: Results for the coefficient functions **(1)**–**(3)** in figs. 2.2 to 2.4, the diffusion problem, different methods, and a tolerance of 0.01 for the selection of eigenvectors: iteration count, condition number, resulting coarse space dimension, and coarse space dimension over the size of the stiffness matrix. If the modified heuristic variant is used, the slab width in layers of finite elements is appended as [H(l)] to the method's name.

Further improvements may be achieved by constructing smoother interface functions. The results for problem **(4)** in table 9.2 corroborate this disadvantage. Nonetheless, the results are promising considering the rough approximation of eigenfunctions and the fact that we have addressed two of the major drawbacks of adaptive coarse spaces (Schur complements need not be set up, and local Neumann stiffness matrices are not required). Let us remark, however, that no theoretical evidence of the robustness of the heuristic

method	*tol*	it.	κ	$\dim V_0$	$\frac{\dim V_0}{\text{dof}}$
RAGDSW–S	0.01	155.3 (173)	237.4 (606.8)	4 402.1 (4 556)	1.0% (1.0%)
RAGDSW–S	0.1	76.7 (86)	50.1 (85.8)	5 616.4 (5 779)	1.2% (1.3%)
RAGDSW–S[H(10)]	100	153.6 (176)	226.5 (381.6)	5 185.6 (5 388)	1.1% (1.2%)

Table 9.2.: **(Model problem (4))** Average results (maximum in parentheses) for 100 randomly generated coefficient functions (cf. section 2.4), the diffusion problem, different methods and tolerances for the selection of eigenvectors: iteration count, condition number, resulting coarse space dimension, and coarse space dimension over the size of the stiffness matrix. The modified heuristic variant RAGDSW–S[H(10)] uses a slab of 10 layers of finite elements.

coarse space is available.

In the following, we discuss further issues to consider. Unlike the generalized eigenvalue problems of this work, the heuristically constructed coarse space does not take into account that certain coarse functions can be discarded if a subdomain adjacent to the respected interface component touches the Dirichlet boundary $\partial \Omega_D$; cf. section 1.5.3. If the global Neumann stiffness matrix is available, this can be easily incorporated into the construction process: if, during the construction, the vector v_i is nonzero at a Dirichlet boundary node, it is marked for deletion. We can furthermore skip the construction of the null space function if no interface functions have been constructed and if a subdomain adjacent to ξ touches $\partial \Omega_D$.

Let us recall the numerical example in fig. 3.7, for which we observed that using at most one coarse function per connected patch of large coefficients may not be sufficient to obtain a robust preconditioner. In contrast to an eigenvalue problem, the heuristic approach is blind to such problems. Equally, if the fine mesh resolution is increased—unlike for our adaptive coarse spaces (cf. section 6.4.2)—the nonadaptive coarse space will not be enriched, and the condition number will depend on H/h.

10. Conclusion and Future Work

We have presented different coarse spaces and variants and have provided supporting numerical evidence in addition to proofs of condition number bounds. At the core of all presented adaptive coarse spaces are solutions of local generalized eigenvalue problems. The eigenvalue problems are defined on the components of a partition of the domain decomposition interface. In contrast to the GenEO and Dirichlet-to-Neumann coarse spaces, each degree of freedom on the interface is associated with only exactly one eigenvalue problem, which reduces the computational cost and can reduce the coarse space dimension; see also table 8.9.

The coarse spaces have been classified as either GDSW-type or ACMS-type, where the latter is a generalization of the former. The key difference is that ACMS-type coarse spaces may enforce additional Dirichlet conditions in the energy-minimizing extensions that are incorporated into the generalized eigenvalue problems; GDSW-type coarse spaces enforce a Dirichlet condition only on the respective interface component.

We have leveraged three concepts to achieve a small coarse space dimension. First, we have incorporated the energy-minimizing extension into the generalized eigenvalue problem, enabling the algorithm to detect connected patches of large coefficients. Second, we have changed the interface partition for RAGDSW to use larger components, which can further facilitate the detection of connected patches of large coefficients. Third, for the ACMS-type coarse spaces, we have enforced additional Dirichlet conditions, a technique that can remove bad eigenmodes if a patch of large coefficients intersects more than one interface component. Numerical results have confirmed that all three concepts can reduce the coarse space dimension significantly.

The coarse spaces each have their advantages and disadvantages. The standard adaptive GDSW coarse space is comparatively easy to implement but, generally, has the largest coarse space dimension among the coarse spaces discussed. The reduced-dimension AGDSW coarse space can significantly decrease the coarse space dimension but is more difficult to implement. Furthermore, the computational cost for solving local generalized eigenvalue problems is increased. Both coarse spaces—and in fact all adaptive GDSW-type coarse spaces—have in common that they are robust to special cases of interface decompositions; cf. section 4.4 and remark 7.1. In contrast, one must be particularly careful when implementing the OS–ACMS coarse space to cover all special cases. On the other hand, OS–ACMS uses the same interface partition as AGDSW and, thus, its local generalized eigenvalue problems are equal in size. Yet, we can achieve much smaller coarse space dimensions than with AGDSW. R–WB–OS–ACMS can further reduce the coarse space dimension with respect to OS–ACMS. However, its implementation is more demanding than that of AGDSW and, arguably, than that of RAGDSW as well: it requires the computation of energy-minimizing extensions from wire basket stars to coarse faces. We conclude that the choice of a particular coarse space relies on practical considerations, the complexity of the domain decomposition, and the considered coefficient functions.

All numerical results in this work were obtained using a serial MATLAB implementation. For this reason, we have not given a comprehensive comparison with other coarse spaces and limited ourselves to the analysis of a few aspects. A comparison based on efficient parallel implementations that contrast different aspects has to be considered in the future.

We have seen in section 5.7.3 that the interface stars defined in section 5.4 deviate less in size from the mean than components of other types of interface partitions. Nevertheless, there is room for improvement. Additional aspects that are relevant for the construction of interface components are the minimization of the number of energy-minimizing extensions per subdomain, an efficient construction in a parallel environment, and geometrically compact interface components (to reduce the coarse space dimension). Let us note that the setup of the Schur complement essentially involves the computation of many energy-

minimizing extensions. This computational effort could be reduced with an iterative eigensolver; cf. section 9.1. Extensive tests of different interface partitions need to be carried out to gain a more thorough understanding of how different interface components affect the coarse space dimension and parallel efficiency. Furthermore, the synergy of different interface components and the enforcement of additional Dirichlet conditions in the ACMS framework (cf. section 7.5) could be examined.

Numerical results have shown that coarse functions are redundant if the associated patch of large coefficients is embedded in an overlapping subdomain; cf. section 1.5.4. A heuristic or machine learning–based approach—as in chapter 9—or a new or additional type of eigenvalue problem may be required to exclude these redundant coarse functions.

In case all techniques to reduce the coarse space dimension fail to obtain a sufficiently small coarse space dimension, and if iterative solvers are required at the coarse level, we can resort to adaptive multilevel methods; cf., e.g., [SŠM13; Wil14; Wil13; KC16; ZT17; KLRW19; HKRR20b; JRZ21; AGJT21; BSSS22; AJ22]. As the coarse problem can be ill conditioned, we may face the same challenges as for the original problem.

All presented (nonheuristic) adaptive coarse spaces have two major drawbacks: they require subdomain Neumann stiffness matrices, which cannot be extracted from the fully assembled stiffness matrix K, and the setup cost is substantial, making their application only feasible for challenging problems. In section 9.2, we have heuristically constructed an approximation of a standard adaptive coarse space that alleviates these disadvantages to some extent but brings about its own challenges; moreover, the construction has only been formulated for a diffusion problem with a scalar coefficient function.

Recently, there has been work toward adaptive coarse spaces that are constructed algebraically [AG19; Spi21; GS21], that is, by only using the fully assembled stiffness matrix K. However, the additional cost for gaining independence of local Neumann matrices is significant. Therefore, the new methods are not a replacement for adaptive coarse spaces that are constructed nonalgebraically as long as the required matrices are available. See [HS22] for work that is more closely related to the one in this thesis. Let us also remark that, if subdomains are based on an algebraic partition of indices rather

than a geometric partition of the domain, the connectivity of patches of large coefficients may not be exploited anymore, possibly resulting in larger coarse space dimensions. Nevertheless, approaches that do not require local Neumann matrices are a welcome addition and enrich practitioners' choices.

For the proof of condition number bounds in this work, we have relied on an overlapping domain decomposition with minimal overlap of one layer of finite elements. This decomposition is only used during the proof and does not place any restrictions on the size of the overlap used in practice nor does it invalidate the theory for larger overlaps. A new type of eigenvalue problem may be required to remedy this shortcoming. Additionally, and possibly related—in the way that a new type of eigenvalue problem may be required—is the support for higher-order polynomial basis functions.

Furthermore, we have presented numerical results exclusively for diffusion and compressible, linear elasticity problems with scalar coefficient functions. Although additional types of symmetric, positive definite problems are covered by our theoretical analysis (as long as mass variants are not selected), new challenges can emerge as in the case of linear elasticity, where pseudoinverses have to be used; to obtain numerical results and avoid the computation of pseudoinverses, we were able to use a simple regularization that proved to be robust for the problems considered. Future work may involve broadening the class of considered problems, including, for example, indefinite problems; see, for example, [NT21; BDG+22; BDJT21] for recent work based on the GenEO and Dirichlet-to-Neumann coarse spaces.

Finally, the numerical results in this thesis have shown that choosing a suitable tolerance for the selection of eigenfunctions is not straightforward. An a priori choice that is identical for all types of eigenvalue problems can lead to slow convergence or a coarse space dimension that is too large. We mention the work [HKK20], in which the authors analyze the local spectra of a few coarse spaces (including AGDSW, SHEM, and OS–ACMS) for a selection of model problems.

A. Theory

A.1. Technical Tools

Lemma A.1 (Lemma of Lax–Milgram for a Defective Problem). *Let $a(\cdot, \cdot)$ be a symmetric bilinear form on a Hilbert space W with the induced norm $\| \cdot \|_W$. We assume that a Hilbert space $W_0 \subset W$ and a splitting $W_0 = W_{0,\text{kernel}} \oplus W_{0,R}$ of Hilbert spaces $W_{0,\text{kernel}}$ and $W_{0,R}$ exist such that:*

1. $a(\cdot, \cdot)$ is $W_{0,R}$-elliptic; i.e., there exists a constant $\alpha > 0$, which is independent of $u \in W_{0,R}$, such that

$$a(u, u) \geq \alpha \|u\|_W^2 \quad \forall u \in W_{0,R}.$$

2. $W_{0,\text{kernel}}$ is the null space of $a(\cdot, \cdot)$; i.e., for all $v \in W_{0,\text{kernel}}$, we have

$$a(v, w) = 0 \quad \forall w \in W.$$

Furthermore, we assume that $a(\cdot, \cdot)$ is continuous; i.e., there exists a constant C, which is independent of $u, v \in W$, such that

$$|a(u, v)| \leq C \|u\|_W \|v\|_W \quad \forall u, v \in W.$$

For $u_D \in W$, the solution to the following problem is given by $\hat{u} + z \in W_0$, where $\hat{u} \in W_{0,R}$ is uniquely defined and $z \in W_{0,\text{kernel}}$: find $u \in W_0$ such that

$$a(u, v) = -a(u_D, v) \quad \forall v \in W_0. \tag{A.1}$$

Proof. Since $a(\cdot, \cdot)$ is a $W_{0,R}$-elliptic, continuous bilinear form, by the lemma of Lax–Milgram ([Cia02, theorem 1.1.3]), there exists a unique solution to the variational problem:

A. Theory

find $\hat{u} \in W_{0,R}$ such that

$$a(\hat{u}, v) = -a(u_D, v) \quad \forall v \in W_{0,R}.$$

Since—for all $v \in W_{0,\text{kernel}}$—we have $a(v, w) = a(w, v) = 0$ for all $w \in W$, we obtain

$$a(\hat{u}, v) = -a(u_D, v) \quad \forall v \in W_0.$$

For $z \in W_{0,\text{kernel}}$, it follows that

$$a(\hat{u} + z, v) = -a(u_D, v) \quad \forall v \in W_0,$$

where $\hat{u} + z \in W_0$. $\qquad\square$

Lemma A.2 (Energy-Minimality of Functions)**.** *(See also [TW05, lemma 4.9], [Cia02, remark 2.1.1], and [Cia13, theorem 6.1-1].) Under the assumptions of lemma A.1, we have*

$$a(\tilde{u}, \tilde{u}) = \min_{\substack{w \in W \\ w - u_D \in W_0}} a(w, w),$$

where $u = \tilde{u} - u_D \in W_0$ is a solution to (A.1) that is unique up to an element of $W_{0,\text{kernel}}$.

Proof. Let $u \in W_0$ be a solution to (A.1) such that

$$a(u, v) = -a(u_D, v) \quad \forall v \in W_0.$$

With $\tilde{u} := u + u_D \in W$, we have

$$a(\tilde{u}, v) = 0 \quad \forall v \in W_0.$$

Let $w \in W$ such that $w - u_D \in W_0$. Then

$$a(\tilde{u}, w - \tilde{u}) = a(\tilde{u}, (w - u_D) - u) = 0,$$

since $(w - u_D) - u \in W_0$. Therefore, since $a(\cdot, \cdot)$ is symmetric and positive semidefinite, we obtain

$$a(w, w) = a(w - \tilde{u}, w - \tilde{u}) + 2a(\tilde{u}, w) - a(\tilde{u}, \tilde{u})$$
$$= a(w - \tilde{u}, w - \tilde{u}) + 2a(\tilde{u}, w - \tilde{u}) + a(\tilde{u}, \tilde{u}) \geq a(\tilde{u}, \tilde{u}).$$

$\qquad\square$

Lemma A.3. *Let $\alpha(\cdot,\cdot)$ be a symmetric, positive semidefinite bilinear form and $\beta(\cdot,\cdot)$ a symmetric, positive definite bilinear form on the finite-dimensional vector space X over $\mathbb{R}$. We consider the generalized eigenvalue problem: find $v \in X$ such that*

$$\alpha(v, w) = \lambda\beta(v, w) \quad \forall w \in X.$$

Then there exist eigenpairs $\{(v_k, \lambda_k)\}_{k=1}^{\dim(X)}$ such that the eigenvectors are a β-orthonormal basis of X.

Proof. Let $n := \dim(X)$. As X is finite-dimensional, we can associate $\alpha(\cdot,\cdot)$ and $\beta(\cdot,\cdot)$ with matrices A and B and obtain the equivalent problem: find $v \in \mathbb{R}^n$ such that

$$w^T A v = \lambda w^T B v \quad \forall w \in \mathbb{R}^n.$$

Since the equation holds for all $w \in \mathbb{R}^n$, we obtain the problem: find $v \in \mathbb{R}^n$ and $\lambda \in \mathbb{R}$ such that

$$Av = \lambda Bv.$$

Since $\beta(\cdot,\cdot)$ is positive definite and symmetric, we can formulate the equivalent problem

$$\tilde{A}\tilde{v} = \lambda\tilde{v},$$

where $\tilde{A} := B^{-1/2} A B^{-1/2}$ and $\tilde{v} := B^{1/2} v$. Since A is symmetric, positive semidefinite and $B^{-1/2}$ symmetric, positive definite, $\tilde{A}$ is symmetric and positive semidefinite. In case A is even positive definite, $\tilde{A}$ is positive definite as well.

Thus, all eigenvalues λ are positive if A is positive definite, and they are nonnegative if A is positive semidefinite. Furthermore, there exists an orthonormal basis $\{\tilde{v}_k\}_{k=1}^{\dim(X)}$ of eigenvectors. Let $v_k := B^{-1/2}\tilde{v}_k$. Then we have

$$v_k^T B v_j = \tilde{v}_k^T B^{-1/2} B B^{-1/2} \tilde{v}_j = \delta_{k,j},$$

where $\delta_{k,j}$ is the Kronecker delta. $\qquad\square$

Lemma A.4. *Let $T \in \tau_h(\Omega)$, that is, a triangle, a tetrahedron, a rectangle, or a cuboid, and let n be the number of vertices of the element. By φ_i, $i = 1, \ldots, n$, we denote the*

corresponding (scalar) linear, bilinear, or trilinear nodal basis functions. Let $c_i \in \mathbb{R}$, $i = 1, \ldots, n$, be arbitrary coefficients. Then

$$v_1 := \left\| \sum_{i=1}^{n} c_i^2 \varphi_i^2 \right\|_{L^\infty(T)} = \left\| \left(\sum_{i=1}^{n} c_i \varphi_i \right)^2 \right\|_{L^\infty(T)} =: v_2.$$

Proof. We assume that $|c_1| = \max_{i=1,\ldots,n} |c_i|$. It follows that $v_1 \geq c_1^2$ and $v_2 \geq c_1^2$. In the following, we first show that $v_1 \leq c_1^2$ and then $v_2 \leq c_1^2$. First, we have to show that

$$\sum_{i=1}^{n} c_i^2 \varphi_i^2 \leq c_1^2 \quad \text{on } T.$$

We have

$$\sum_{i=1}^{n} c_i^2 \varphi_i^2 \leq \sum_{i=1}^{n} c_1^2 \varphi_i^2 = c_1^2 \sum_{i=1}^{n} \varphi_i^2 \quad \text{on } T.$$

Using $\sum_{i=1}^{n} \varphi_i = 1$, we need to show

$$\sum_{i=1}^{n} \varphi_i^2 \leq 1 = \left(\sum_{i=1}^{n} \varphi_i \right)^2 = \left(\sum_{i=1}^{n} \varphi_i^2 \right) + \left(\sum_{i \neq j} \varphi_i \varphi_j \right) \quad \text{on } T.$$

This is equivalent to

$$0 \leq \sum_{i \neq j} \varphi_i \varphi_j \quad \text{on } T.$$

Since $\mathcal{P}_1$ and $\mathcal{Q}_1$ nodal basis functions are nonnegative, the inequality is satisfied. We have proved that

$$\sum_{i=1}^{n} c_i^2 \varphi_i^2 \leq \sum_{i=1}^{n} c_1^2 \varphi_i^2 \leq c_1^2 \quad \text{on } T.$$

To show $v_2 \leq c_1^2$, we note that

$$\left(\sum_{i=1}^{n} c_i \varphi_i \right)^2 \leq \left(\sum_{i=1}^{n} |c_i| \varphi_i \right)^2 \leq \left(\sum_{i=1}^{n} |c_1| \varphi_i \right)^2 = c_1^2 \quad \text{on } T.$$

$$\square$$

Corollary A.1. *Let $T \in \tau_h(\Omega)$, that is, a triangle, a tetrahedron, a rectangle, or a cuboid, and let n be the number of vertices of the element. We partition the set of vertices into m disjoint subsets, each containing k_i $(i = 1, \ldots, m)$ vertices. As before, we associate the*

corresponding (scalar) linear, bilinear, or trilinear nodal basis functions $\varphi_{i,l}$, $i = 1, \ldots, m$, $l = 1, \ldots, k_i$, with the vertices. Let $c_{i,l} \in \mathbb{R}$ be arbitrary coefficients. Then we define

$$\tilde{\varphi}_i := \sum_{l=1}^{k_i} c_{i,l} \varphi_{i,l}.$$

It follows that

$$v_1 := \Big\| \sum_{i=1}^{m} \tilde{\varphi}_i^2 \Big\|_{L^\infty(T)} = \Big\| \Big(\sum_{i=1}^{m} \tilde{\varphi}_i \Big)^2 \Big\|_{L^\infty(T)} =: v_2.$$

Proof. As the proof is essentially identical to that of lemma A.4, we only show a sketch. We assume that $c_{1,1}$ is the largest coefficient in magnitude. Then we have

$$\sum_{i=1}^{m} \tilde{\varphi}_i^2 \leq c_{1,1}^2 \sum_{i=1}^{m} \Big(\sum_{l=1}^{k_i} \varphi_{i,l} \Big)^2 = c_{1,1}^2 \left(1 - \Big(\sum_{i \neq j} \Big(\sum_{l=1}^{k_i} \varphi_{i,l} \Big) \Big(\sum_{l=1}^{k_j} \varphi_{j,l} \Big) \Big) \right) \leq c_{1,1}^2 \quad \text{on } T.$$

To show $v_2 \leq c_{1,1}^2$, we note that

$$\Big(\sum_{i=1}^{m} \sum_{l=1}^{k_i} c_{i,l} \varphi_{i,l} \Big)^2 \leq c_{1,1}^2 \Big(\sum_{i=1}^{m} \sum_{l=1}^{k_i} \varphi_{i,l} \Big)^2 = c_{1,1}^2 \quad \text{on } T.$$

$\square$

Corollary A.2. *Let $T \in \tau_h(\Omega)$, and let n be the number of the vertices of T. Let $v^{(i)} \in V^h(T)$, $i = 1, \ldots, k$, $k \leq n$, where $V^h(T) \subset (H^1(T))^{\hat{d}}$, $\hat{d} \in \mathbb{N}_{>0}$, and $v^{(i)} = (v_1^{(i)}, \ldots, v_{\hat{d}}^{(i)})$, $i = 1, \ldots, k$. We assume that the $v_j^{(i)}$ fulfill*

$$\Big\| \sum_{i=1}^{k} (v_j^{(i)})^2 \Big\|_{L^\infty(T)} = \Big\| \Big(\sum_{i=1}^{k} v_j^{(i)} \Big)^2 \Big\|_{L^\infty(T)} \quad \forall j = 1, \ldots, \hat{d}. \tag{A.2}$$

Then there exists a constant C_{inv}, which is independent of h_T and $v^{(i)}$, such that

$$\sum_{i=1}^{k} \|v^{(i)}\|_{L^2(T)}^2 \leq C_{\mathrm{inv}} \Big\| \sum_{i=1}^{k} v^{(i)} \Big\|_{L^2(T)}^2.$$

Proof. Using (A.2), an inverse inequality to bound the L^∞ norm by the L^1 norm (cf. [BS08,

lemma 4.5.3]), and $|T| \leq \mathrm{diam}(T)^d = h_T^d$, we obtain

$$
\begin{aligned}
\sum_{i=1}^{k} \|v^{(i)}\|_{L^2(T)}^2 &= \sum_{i=1}^{k} \int_T \sum_{j=1}^{\hat{d}} \left(v_j^{(i)}(x)\right)^2 \mathrm{d}x \\
&= \sum_{j=1}^{\hat{d}} \left\| \sum_{i=1}^{k} (v_j^{(i)})^2 \right\|_{L^1(T)} \\
&\leq \sum_{j=1}^{\hat{d}} \left\| \sum_{i=1}^{k} (v_j^{(i)})^2 \right\|_{L^\infty(T)} |T| \\
&= \sum_{j=1}^{\hat{d}} \left\| \left(\sum_{i=1}^{k} v_j^{(i)} \right)^2 \right\|_{L^\infty(T)} |T| \\
&\leq \frac{C_{\mathrm{inv}}}{h_T^d} \sum_{j=1}^{\hat{d}} \left\| \left(\sum_{i=1}^{k} v_j^{(i)} \right)^2 \right\|_{L^1(T)} |T| \\
&\leq C_{\mathrm{inv}} \sum_{j=1}^{\hat{d}} \left\| \sum_{i=1}^{k} v_j^{(i)} \right\|_{L^2(T)}^2 \\
&= C_{\mathrm{inv}} \left\| \sum_{i=1}^{k} v^{(i)} \right\|_{L^2(T)}^2 .
\end{aligned}
$$

$\square$

A.2. Interface Partition

	nodes in adj. subd. $(\mathcal{E})$				nodes in adj. subd. $(\mathcal{F})$			
	avg.	$\frac{\text{min.}}{\text{avg.}}$	$\frac{\text{max.}}{\text{avg.}}$	$\frac{\text{std.}}{\text{avg.}}$	avg.	$\frac{\text{min.}}{\text{avg.}}$	$\frac{\text{max.}}{\text{avg.}}$	$\frac{\text{std.}}{\text{avg.}}$
(1)	3 903.1	93.1%	133.0%	4.5%	2 658.7	94.0%	105.2%	1.9%
(2)	3 763.1	93.1%	131.8%	5.0%	2 558.6	93.5%	105.9%	3.0%
(3)	19 449.4	95.4%	133.0%	4.5%	13 102.5	95.4%	104.5%	1.7%
(4)	3 359.4	94.2%	133.9%	4.4%	2 290.6	93.7%	106.0%	1.8%

	nodes in adj. subd. $(\mathcal{S}_\Gamma)$				nodes in adj. subd. $(\mathcal{S}_\mathcal{W})$			
	avg.	$\frac{\text{min.}}{\text{avg.}}$	$\frac{\text{max.}}{\text{avg.}}$	$\frac{\text{std.}}{\text{avg.}}$	avg.	$\frac{\text{min.}}{\text{avg.}}$	$\frac{\text{max.}}{\text{avg.}}$	$\frac{\text{std.}}{\text{avg.}}$
(1)	5 085.2	73.8%	127.1%	4.5%	5 085.2	73.8%	127.1%	4.5%
(2)	4 895.5	74.0%	122.3%	6.5%	4 895.5	74.0%	122.3%	6.5%
(3)	25 585.1	96.6%	125.3%	3.7%	25 585.1	96.6%	125.3%	3.7%
(4)	4 391.5	73.2%	144.1%	4.9%	4 391.0	73.2%	144.2%	4.9%

	nodes in adj. subd. $(\mathcal{V})$			
	avg.	$\frac{\text{min.}}{\text{avg.}}$	$\frac{\text{max.}}{\text{avg.}}$	$\frac{\text{std.}}{\text{avg.}}$
(1)	5 095.8	95.1%	126.8%	4.0%
(2)	4 938.8	94.3%	121.2%	4.9%
(3)	25 591.7	96.6%	125.2%	3.7%
(4)	4 395.0	94.7%	144.0%	4.8%

Table A.1.: Number of nodes in the union of subdomains adjacent to interface components (coarse edges, faces, interface stars, wire basket stars, vertices): average and minimum/maximum/standard deviation over the average. **(1)**–**(4)** correspond to sections 2.1 to 2.4.
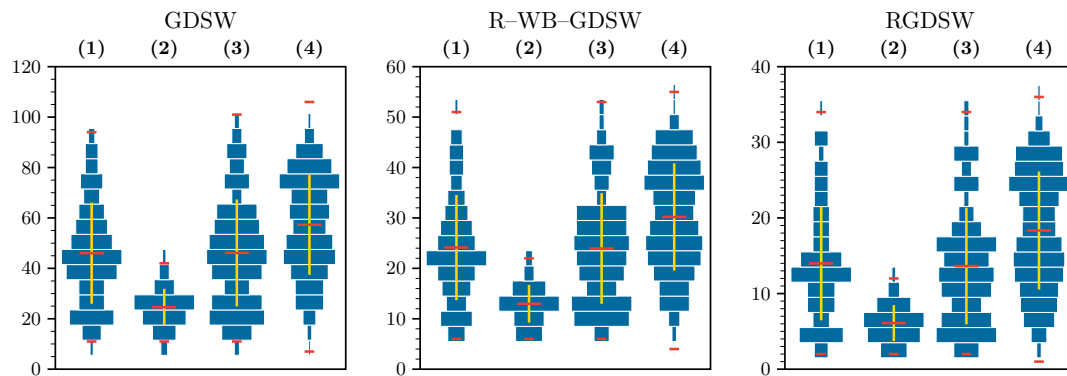
Figure A.1.: Histograms of the number of interface components per subdomain for GDSW, R–WB–GDSW, and RGDSW. **(1)**–**(4)** correspond to sections 2.1 to 2.4. The widths of the blue bars indicate the proportion of subdomains that contain the respective amount of interface components. The width of bars is not comparable between different model problems. The minimum, average, and maximum are marked in red. The range of one standard deviation from the average is marked in yellow.
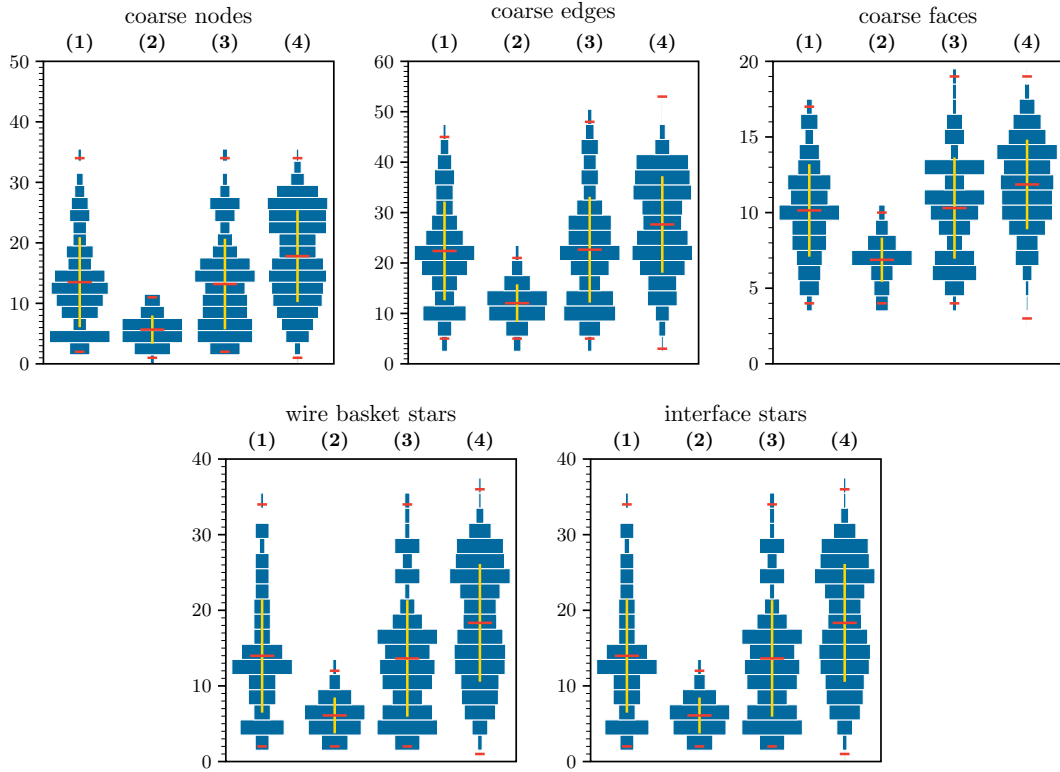
Figure A.2.: Histograms of the number of interface components per subdomain. **(1)**–**(4)** correspond to sections 2.1 to 2.4. The widths of the blue bars indicate the proportion of subdomains that contain the respective amount of interface components. The width of bars is not comparable between different model problems. The minimum, average, and maximum are marked in red. The range of one standard deviation from the average is marked in yellow.

A.3. Condition Number Bound for R–WB–OS–ACMS

Let $u \in V^h(\Omega)$. We define the spectral projections

$$\Pi_{\mathcal{S}_{\mathcal{W}}} u := \sum_{s \in \mathcal{S}_{\mathcal{W}}} \Pi_s u, \qquad \Pi_s u := \sum_{\lambda_{k,e} \leq tol_e} \beta_e\left(u, v_{k,e}\right) v_{k,e},$$

$$\Pi_{\mathcal{F}} u := \sum_{f \in \mathcal{F}} \Pi_f u, \qquad \Pi_f u := \sum_{\lambda_{k,f} \leq tol_f} \beta_f\left(u, v_{k,f}\right) v_{k,f},$$

where the tolerances $tol_s > 0$ and $tol_f > 0$ are user-prescribed. We define the coarse component as

$$u_0 := \Pi_{\mathcal{S}_{\mathcal{W}}} u + \Pi_{\mathcal{F}} \overline{u}_{\mathcal{S}_{\mathcal{W}}}, \quad \overline{u}_{\mathcal{S}_{\mathcal{W}}} := u - \Pi_{\mathcal{S}_{\mathcal{W}}} u.$$

For the remainder $u - u_0$, we obtain

$$u - u_0 = u - \Pi_{\mathcal{S}_{\mathcal{W}}} u - \Pi_{\mathcal{F}} \overline{u}_{\mathcal{S}_{\mathcal{W}}} = \overline{u}_{\mathcal{S}_{\mathcal{W}}} - \Pi_{\mathcal{F}} \overline{u}_{\mathcal{S}_{\mathcal{W}}}.$$

Lemma A.5 (Wire Basket Contribution)**.** *Let $s \in \mathcal{S}_{\mathcal{W}}$. Then we have*

$$\|u - u_0\|^2_{\beta_s^K} = \sum_{s_i \in \mathcal{N}_s} |z_{s_i}(\overline{u}_{\mathcal{S}_{\mathcal{W}}})|^2_{a(\Omega_{s_i})} \leq \frac{C_{\text{inv}}}{tol_s} |u|^2_{a(\Omega_s)}$$

for $u \in V^h(\Omega)$.

Proof. As we use adaptive GDSW-type generalized eigenvalue problems for wire basket stars, there are no contributions of other coarse functions on s. Consequently, we have

$$\|u - u_0\|^2_{\beta_s^K} = \sum_{s_i \in \mathcal{N}_s} |z_{s_i}(\overline{u}_{\mathcal{S}_{\mathcal{W}}} - \Pi_{\mathcal{F}} \overline{u}_{\mathcal{S}_{\mathcal{W}}})|^2_{a(\Omega_{s_i})}$$

$$= \sum_{s_i \in \mathcal{N}_s} |z_{s_i}(\overline{u}_{\mathcal{S}_{\mathcal{W}}})|^2_{a(\Omega_{s_i})}$$

$$= \sum_{s_i \in \mathcal{N}_s} |z_{s_i}(u - \Pi_{\mathcal{S}_{\mathcal{W}}} u)|^2_{a(\Omega_{s_i})}$$

$$= \sum_{s_i \in \mathcal{N}_s} |z_{s_i}(u - \Pi_s u)|^2_{a(\Omega_{s_i})}$$

$$= \|u - \Pi_s u\|^2_{\beta_s^K}.$$

Using (6.38) and lemma 6.2, it follows that

$$\|u - \Pi_s u\|^2_{\beta_s^K} \leq C_{\text{inv}} \|u - \Pi_s u\|^2_{\beta_s} \leq \frac{C_{\text{inv}}}{tol_s} |u|^2_{a(\Omega_s)}.$$

$\square$

Lemma A.6 (Face Contribution)**.** *Let $f \in \mathcal{F}$. Then we have*

$$\|u - u_0\|^2_{\beta^K_f} \leq \frac{8C_{\text{inv}}}{tol_f} \left(|u|^2_{a(\Omega_f)} + \sum_{s \in \mathcal{S}_\mathcal{W} \cap \mathcal{B}_{\text{p}}(f)} \frac{C_{\text{inv}} C_\tau}{tol_s} |u|^2_{a(\Omega_s)} \right)$$

for $u \in V^h(\Omega)$, where $\mathcal{S}_\mathcal{W} \cap \mathcal{B}_{\text{p}}(f)$ is the set of wire basket stars that are adjacent to f.

Proof. The following proof is also valid if f does not have adjacent wire basket stars. In that case, however, the proof could be simplified significantly and the bound improved.

In a first step, we can use the fact that face functions of faces other than f are zero on $\overline{f}$.

$$\|u - u_0\|^2_{\beta^K_f} = |z_f(\overline{u}_{\mathcal{S}_\mathcal{W}} - \Pi_\mathcal{F} \overline{u}_{\mathcal{S}_\mathcal{W}})|^2_{a(\Omega_f)}$$

$$= |z_f(\overline{u}_{\mathcal{S}_\mathcal{W}} - \Pi_f \overline{u}_{\mathcal{S}_\mathcal{W}})|^2_{a(\Omega_f)}$$

$$= \|\overline{u}_{\mathcal{S}_\mathcal{W}} - \Pi_f \overline{u}_{\mathcal{S}_\mathcal{W}}\|^2_{\beta^K_f}.$$

Using (6.38) and lemma 6.1, it follows that

$$\|\overline{u}_{\mathcal{S}_\mathcal{W}} - \Pi_f \overline{u}_{\mathcal{S}_\mathcal{W}}\|^2_{\beta^K_f} \leq C_{\text{inv}} \|\overline{u}_{\mathcal{S}_\mathcal{W}} - \Pi_f \overline{u}_{\mathcal{S}_\mathcal{W}}\|^2_{\beta_f}$$

$$\leq \frac{C_{\text{inv}}}{tol_f} |\overline{u}_{\mathcal{S}_\mathcal{W}} - \Pi_f \overline{u}_{\mathcal{S}_\mathcal{W}}|^2_{\alpha_f}$$

$$\leq \frac{C_{\text{inv}}}{tol_f} |\overline{u}_{\mathcal{S}_\mathcal{W}}|^2_{\alpha_f}$$

$$= \frac{C_{\text{inv}}}{tol_f} |z_f(\overline{u}_{\mathcal{S}_\mathcal{W}})|^2_{\alpha_{\overline{f}}}.$$

We have

$$z_f(\overline{u}_{\mathcal{S}_\mathcal{W}}) = z_{\overline{f}}(\overline{u}_{\mathcal{S}_\mathcal{W}}) - z_{\partial f}(\overline{u}_{\mathcal{S}_\mathcal{W}}) = \overline{u}_{\mathcal{S}_\mathcal{W}} - z_{\partial f}(\overline{u}_{\mathcal{S}_\mathcal{W}}) \quad \text{on } \overline{f}.$$

Using the contributions of wire basket problems, we obtain

$$z_f(\overline{u}_{\mathcal{S}_\mathcal{W}}) = \overline{u}_{\mathcal{S}_\mathcal{W}} - \sum_{s \in \mathcal{S}_\mathcal{W} \cap \mathcal{B}_{\text{p}}(f)} z_s(\overline{u}_{\mathcal{S}_\mathcal{W}}) \quad \text{on } \overline{f},$$

where $\mathcal{S}_\mathcal{W} \cap \mathcal{B}_{\text{p}}(f)$ are all wire basket stars adjacent to f. We obtain

$$|z_f(\overline{u}_{\mathcal{S}_\mathcal{W}})|_{\alpha_{\overline{f}}} \leq |\overline{u}_{\mathcal{S}_\mathcal{W}}|_{\alpha_{\overline{f}}} + \left| \sum_{s \in \mathcal{S}_\mathcal{W} \cap \mathcal{B}_{\text{p}}(f)} z_s(\overline{u}_{\mathcal{S}_\mathcal{W}}) \right|_{\alpha_{\overline{f}}}. \tag{A.3}$$

A. Theory

For the first term on the right-hand side of (A.3), we have

$$|\overline{u}_{\mathcal{S}_\mathcal{W}}|_{\alpha_{\overline{f}}} = |u - \Pi_{\mathcal{S}_\mathcal{W}}u|_{\alpha_{\overline{f}}}$$

$$\leq |u|_{\alpha_{\overline{f}}} + |\Pi_{\mathcal{S}_\mathcal{W}}u|_{\alpha_{\overline{f}}}$$

$$\leq |u|_{a(\Omega_f)} + |\Pi_{\mathcal{S}_\mathcal{W}}u|_{\alpha_{\overline{f}}},$$

where we have used (7.17) and (7.19) in the last step. Using (7.16) (cf. lemma 7.1), we have

$$|\Pi_{\mathcal{S}_\mathcal{W}}u|_{\alpha_{\overline{f}}} \leq |\Pi_{\mathcal{S}_\mathcal{W}}u|_{a,\partial f \to \Omega_f}.$$

Then

$$|\Pi_{\mathcal{S}_\mathcal{W}}u|_{a,\partial f \to \Omega_f} \leq |u|_{a,\partial f \to \Omega_f} + |u - \Pi_{\mathcal{S}_\mathcal{W}}u|_{a,\partial f \to \Omega_f}$$

$$\leq |u|_{a(\Omega_f)} + |\overline{u}_{\mathcal{S}_\mathcal{W}}|_{a,\partial f \to \Omega_f}.$$

We have

$$|\overline{u}_{\mathcal{S}_\mathcal{W}}|_{a,\partial f \to \Omega_f} = \Big| \sum_{s \in \mathcal{S}_\mathcal{W} \cap \mathcal{B}_\mathrm{p}(f)} z_s(\overline{u}_{\mathcal{S}_\mathcal{W}}) \Big|_{a,\partial f \to \Omega_f} \leq \Big| \sum_{s \in \mathcal{S}_\mathcal{W} \cap \mathcal{B}_\mathrm{p}(f)} z_s(\overline{u}_{\mathcal{S}_\mathcal{W}}) \Big|_{a(\Omega_f)}.$$

As before, we can use (7.17) and (7.19) to obtain

$$\Big| \sum_{s \in \mathcal{S}_\mathcal{W} \cap \mathcal{B}_\mathrm{p}(f)} z_s(\overline{u}_{\mathcal{S}_\mathcal{W}}) \Big|_{\alpha_{\overline{f}}} \leq \Big| \sum_{s \in \mathcal{S}_\mathcal{W} \cap \mathcal{B}_\mathrm{p}(f)} z_s(\overline{u}_{\mathcal{S}_\mathcal{W}}) \Big|_{a,\overline{f} \to \Omega_f} \leq \Big| \sum_{s \in \mathcal{S}_\mathcal{W} \cap \mathcal{B}_\mathrm{p}(f)} z_s(\overline{u}_{\mathcal{S}_\mathcal{W}}) \Big|_{a(\Omega_f)}$$

for the right term on the right-hand side of (A.3). Similar to the proofs of lemmas 6.4 and 7.6, we have

$$\Big| \sum_{s \in \mathcal{S}_\mathcal{W} \cap \mathcal{B}_\mathrm{p}(f)} z_s(\overline{u}_{\mathcal{S}_\mathcal{W}}) \Big|^2_{a(\Omega_f)} \leq C_\tau \sum_{s \in \mathcal{S}_\mathcal{W} \cap \mathcal{B}_\mathrm{p}(f)} \sum_{s_i \in \mathcal{N}_s} |z_{s_i}(\overline{u}_{\mathcal{S}_\mathcal{W}})|^2_{a(\Omega_{s_i})}.$$

Using lemma A.5, we then obtain for (A.3)

$$|z_f(\overline{u}_{\mathcal{S}_\mathcal{W}})|^2_{\alpha_{\overline{f}}} \leq \left(2|u|_{a(\Omega_f)} + 2 \Big| \sum_{s \in \mathcal{S}_\mathcal{W} \cap \mathcal{B}_\mathrm{p}(f)} z_s(\overline{u}_{\mathcal{S}_\mathcal{W}}) \Big|_{a(\Omega_f)} \right)^2$$

$$\leq 8|u|^2_{a(\Omega_f)} + 8C_\tau \sum_{s \in \mathcal{S}_\mathcal{W} \cap \mathcal{B}_\mathrm{p}(f)} \sum_{s_i \in \mathcal{N}_s} |z_{s_i}(\overline{u}_{\mathcal{S}_\mathcal{W}})|^2_{a(\Omega_{s_i})}$$

$$\leq 8|u|^2_{a(\Omega_f)} + 8C_\tau \sum_{s \in \mathcal{S}_\mathcal{W} \cap \mathcal{B}_\mathrm{p}(f)} \frac{C_\mathrm{inv}}{tol_s} |u|^2_{a(\Omega_s)}.$$

In total, we have

$$\|u - u_0\|_{\beta_f^K}^2 \leq \frac{C_{\text{inv}}}{tol_f} |z_f(\overline{u}_{\mathcal{S}_\mathcal{W}})|_{\alpha_{\overline{f}}}^2 \leq \frac{8 C_{\text{inv}}}{tol_f} \left(|u|_{a(\Omega_f)}^2 + \sum_{s \in \mathcal{S}_\mathcal{W} \cap \mathcal{B}_{\text{p}}(f)} \frac{C_{\text{inv}} C_\tau}{tol_s} |u|_{a(\Omega_s)}^2 \right).$$

$\square$

Let $tol_{\mathcal{S}_\mathcal{W}} = \min_{s \in \mathcal{S}_\mathcal{W}} tol_s$, and

$$N^s := \max_{1 \leq i \leq N} |\mathcal{S}_\mathcal{W}(\Omega_i)|, \quad \mathcal{S}_\mathcal{W}(\Omega_i) := \{ s \in \mathcal{S}_\mathcal{W} : s \cap \overline{\Omega}_i \neq \emptyset \},$$

be the maximum number of wire basket stars of a subdomain. Furthermore, we define

$$N^{\partial f \to s} := \max_{1 \leq i \leq N} \sum_{s \in \mathcal{S}_\mathcal{W}(\Omega_i)} |\mathcal{F} \cap \mathcal{B}_{\text{c}}(s)|,$$

where $\mathcal{F} \cap \mathcal{B}_{\text{c}}(s)$ is the set of coarse faces adjacent to an $s \in \mathcal{S}_\mathcal{W}$.

Corollary A.3. *(Analogue of corollary 7.1) It holds that*

$$|u_0|_{a(\Omega)} \leq \left(1 + \sqrt{C_{\text{inv}} C_\tau \left(\frac{N^s}{tol_{\mathcal{S}_\mathcal{W}}} + \frac{8 N^f}{tol_{\mathcal{F}}} + \frac{8 C_{\text{inv}} C_\tau N^{\partial f \to s}}{tol_{\mathcal{F}} tol_{\mathcal{S}_\mathcal{W}}} \right)} \right) |u|_{a(\Omega)};$$

see (6.33) and (6.38) for the definitions of C_τ and C_{inv}.

Proof. Using lemmas A.5 and A.6, we have

$$C_\tau \sum_{\xi \in \mathcal{P}} \|u - u_0\|_{\beta_\xi^K}^2 = C_\tau \left(\sum_{s \in \mathcal{S}_\mathcal{W}} \|u - u_0\|_{\beta_s^K}^2 + \sum_{f \in \mathcal{F}} \|u - u_0\|_{\beta_f^K}^2 \right)$$

$$\leq C_\tau \left(\sum_{s \in \mathcal{S}_\mathcal{W}} \frac{C_{\text{inv}}}{tol_s} |u|_{a(\Omega_s)}^2 \right.$$

$$\left. + \sum_{f \in \mathcal{F}} \frac{8 C_{\text{inv}}}{tol_f} \left(|u|_{a(\Omega_f)}^2 + \sum_{s \in \mathcal{S}_\mathcal{W} \cap \mathcal{B}_{\text{p}}(f)} \frac{C_{\text{inv}} C_\tau}{tol_s} |u|_{a(\Omega_s)}^2 \right) \right)$$

$$\leq C_{\text{inv}} C_\tau \left(\frac{1}{tol_{\mathcal{S}_\mathcal{W}}} \sum_{s \in \mathcal{S}_\mathcal{W}} |u|_{a(\Omega_s)}^2 \right.$$

$$\left. + \frac{8}{tol_{\mathcal{F}}} \sum_{f \in \mathcal{F}} \left(|u|_{a(\Omega_f)}^2 + \frac{C_{\text{inv}} C_\tau}{tol_{\mathcal{S}_\mathcal{W}}} \sum_{s \in \mathcal{S}_\mathcal{W} \cap \mathcal{B}_{\text{p}}(f)} |u|_{a(\Omega_s)}^2 \right) \right)$$

$$\leq C_{\text{inv}} C_\tau \left(\frac{N^s}{tol_{\mathcal{S}_\mathcal{W}}} + \frac{8 N^f}{tol_{\mathcal{F}}} + \frac{8 C_{\text{inv}} C_\tau N^{\partial f \to s}}{tol_{\mathcal{F}} tol_{\mathcal{S}_\mathcal{W}}} \right) |u|_{a(\Omega)}^2.$$

A. Theory

By lemma 6.4, we then obtain

$$|u_0|_{a(\Omega)} \leq \left(1 + \sqrt{C_{\mathrm{inv}}C_\tau \left(\frac{N^s}{tol_{\mathcal{S_W}}} + \frac{8N^f}{tol_{\mathcal{F}}} + \frac{8C_{\mathrm{inv}}C_\tau N^{\partial f \to s}}{tol_{\mathcal{F}}tol_{\mathcal{S_W}}}\right)}\right) |u|_{a(\Omega)}.$$

$\square$

For the following proof, we include the number of adjacent subdomains of a wire basket star as a scaling factor in the definition of N^s:

$$N^{s,\Sigma} := \max_{1 \leq i \leq N} \sum_{s \in \mathcal{S_W}(\Omega_i)} |n(s)|. \tag{A.4}$$

This constant will deteriorate if a wire basket star spans many subdomains. Similarly to the discussion in section 6.4.3, we could derive an improved constant that depends on $\max_{i=1,\dots,n_s} |n(s_i)|$ instead of $|n(s)|$. For the sake of simplicity, and since we have $\max_{i=1,\dots,n_s} |n(s_i)| = |n(s)|$ for the wire basket stars constructed in section 5.5, we will not incorporate the improvement into the proof.

We expand the definition of $N^{\partial f \to s}$ to include $|n(f)|$ as a scaling factor:

$$N^{\partial f \to s,\Sigma} := \max_{1 \leq i \leq N} \sum_{s \in \mathcal{S_W}(\Omega_i)} \sum_{f \in \mathcal{F} \cap \mathcal{B}_c(s)} |n(f)| \tag{A.5}$$

$$= 2 \max_{1 \leq i \leq N} \sum_{s \in \mathcal{S_W}(\Omega_i)} |\mathcal{F} \cap \mathcal{B}_c(s)| = 2N^{\partial f \to s}.$$

We formulate an analogue of corollary 7.2 for R–WB–OS–ACMS.

Corollary A.4. *Let the assumptions of lemma 6.5 be satisfied. Then*

$$|I^h(\Psi \cdot (u - u_0))|^2_{a(B)} \leq \frac{C_{\mathrm{inv}}C_\tau}{tol_{\mathcal{S_W}}} \sum_{s \in \mathcal{S_W}(\Omega_l)} |u|^2_{a(\Omega_s)}$$

$$+ \frac{8C_{\mathrm{inv}}C_\tau}{tol_{\mathcal{F}}} \sum_{f \in \mathcal{F}(\Omega_l)} \left(|u|^2_{a(\Omega_f)} + \frac{C_{\mathrm{inv}}C_\tau}{tol_{\mathcal{S_W}}} \sum_{s \in \mathcal{S_W} \cap \mathcal{B}_P(f)} |u|^2_{a(\Omega_s)}\right).$$

If $l = 0$, that is, $B = \Omega$, we obtain

$$|I^h(\Psi \cdot (u - u_0))|^2_{a(\Omega)} \leq D_1 |u|^2_{a(\Omega)}.$$

If the assumptions of lemma 6.5 are satisfied for all $1 \leq l \leq N$, $\Psi_l \colon \overline{B}_l \to \mathbb{R}$, with $B_l := \tilde{\Omega}_l \setminus \Omega_l$, we obtain

$$\sum_{l=1}^{N} |I^h(\Psi_l \cdot (u - u_0))|^2_{a(\tilde{\Omega}_l \setminus \Omega_l)} \leq D_0 |u|^2_{a(\Omega)}.$$

The constants D_0 and D_1 are defined as

$$D_0 := C_{\text{inv}} C_\tau \left(\frac{N^{s,\Sigma}}{tol_{\mathcal{S}_\mathcal{W}}} + \frac{16 N^f}{tol_{\mathcal{F}}} + \frac{16 C_{\text{inv}} C_\tau N^{\partial f \to s}}{tol_{\mathcal{F}} tol_{\mathcal{S}_\mathcal{W}}} \right), \tag{A.6}$$

$$D_1 := C_{\text{inv}} C_\tau \left(\frac{N^s}{tol_{\mathcal{S}_\mathcal{W}}} + \frac{8 N^f}{tol_{\mathcal{F}}} + \frac{8 C_{\text{inv}} C_\tau N^{\partial f \to s}}{tol_{\mathcal{F}} tol_{\mathcal{S}_\mathcal{W}}} \right). \tag{A.7}$$

Proof. Using lemmas 6.5, A.5, and A.6, we have

$$|I^h(\Psi \cdot (u - u_0))|^2_{a(B)} \leq C_\tau \sum_{\xi \in \mathcal{P}(\Omega_l)} \beta^K_\xi \left(u - u_0 \,,\, u - u_0 \right)$$

$$\leq C_\tau \sum_{s \in \mathcal{S}_\mathcal{W}(\Omega_l)} \frac{C_{\text{inv}}}{tol_s} |u|^2_{a(\Omega_s)}$$

$$+ C_\tau \sum_{f \in \mathcal{F}(\Omega_l)} \frac{8 C_{\text{inv}}}{tol_f} \left(|u|^2_{a(\Omega_f)} + \sum_{s \in \mathcal{S}_\mathcal{W} \cap \mathcal{B}_p(f)} \frac{C_{\text{inv}} C_\tau}{tol_s} |u|^2_{a(\Omega_s)} \right)$$

$$\leq \frac{C_{\text{inv}} C_\tau}{tol_{\mathcal{S}_\mathcal{W}}} \sum_{s \in \mathcal{S}_\mathcal{W}(\Omega_l)} |u|^2_{a(\Omega_s)}$$

$$+ \frac{8 C_{\text{inv}} C_\tau}{tol_{\mathcal{F}}} \sum_{f \in \mathcal{F}(\Omega_l)} \left(|u|^2_{a(\Omega_f)} + \frac{C_{\text{inv}} C_\tau}{tol_{\mathcal{S}_\mathcal{W}}} \sum_{s \in \mathcal{S}_\mathcal{W} \cap \mathcal{B}_p(f)} |u|^2_{a(\Omega_s)} \right).$$

For $l = 0$, that is, $B = \Omega$, we obtain

$$|I^h(\Psi \cdot (u - u_0))|^2_{a(\Omega)} \leq C_{\text{inv}} C_\tau \left(\frac{N^s}{tol_{\mathcal{S}_\mathcal{W}}} + \frac{8 N^f}{tol_{\mathcal{F}}} + \frac{8 C_{\text{inv}} C_\tau N^{\partial f \to s}}{tol_{\mathcal{F}} tol_{\mathcal{S}_\mathcal{W}}} \right) |u|^2_{a(\Omega)}.$$

For the sum over $l = 1, \ldots, N$, we obtain using (7.25), (A.4), and (A.5)

$$\sum_{l=l}^{N} |I^h(\Psi_l \cdot (u - u_0))|^2_{a(\tilde{\Omega}_l \setminus \Omega_l)} \leq C_{\text{inv}} C_\tau \left(\frac{N^{s,\Sigma}}{tol_{\mathcal{S}_\mathcal{W}}} + \frac{8 N^{f,\Sigma}}{tol_{\mathcal{F}}} + \frac{8 C_{\text{inv}} C_\tau N^{\partial f \to s, \Sigma}}{tol_{\mathcal{F}} tol_{\mathcal{S}_\mathcal{W}}} \right) |u|^2_{a(\Omega)}$$

$$= C_{\text{inv}} C_\tau \left(\frac{N^{s,\Sigma}}{tol_{\mathcal{S}_\mathcal{W}}} + \frac{16 N^f}{tol_{\mathcal{F}}} + \frac{16 C_{\text{inv}} C_\tau N^{\partial f \to s}}{tol_{\mathcal{F}} tol_{\mathcal{S}_\mathcal{W}}} \right) |u|^2_{a(\Omega)}.$$

$\square$

Using the constants (A.6) and (A.7), the proof of the existence of a stable decomposition is identical to that of lemma 7.7. Thus, as in theorem 7.1, we obtain the following condition number bound.

Theorem A.1. *The condition number of the R–WB–OS–ACMS two-level Schwarz operator (in three dimensions) is bounded by*

$$\kappa \left(M^{-1}_{\text{R–WB–OS–ACMS}} K \right) \leq \left(4 + 5(1 + \sqrt{D_1})^2 + D_0 \right) \left(\hat{N}_c + 1 \right),$$

where D_0 and D_1 are defined in (A.6) and (A.7), and where $\hat{N}_c$ is the maximum number of overlapping subdomains $\{\Omega_i'\}_{i=1}^N$ any finite element node $x^h \in \overline{\Omega}$ can belong to. All constants are independent of H, h, and the contrast of the coefficient function E.

In two dimensions, R–WB–OS–ACMS is identical to R–WB–AGDSW.

B. Numerical Results

B.1. Diffusion Problems

method	*tol*	it.	κ	dim V_0	($\mathcal{V}$,	$\mathcal{E}$,	$\mathcal{F}$)	$\frac{\dim V_0}{\mathrm{dof}}$
CG	—	>2 000	$2.2{\cdot}10^8$		—			—
OSL1	—	1 421	$1.4{\cdot}10^6$		—			—
GDSW	—	1 059	$4.7{\cdot}10^5$	1 980	(419,	927,	634)	1.49%
EMR–VB	10^{-6}	1 132	$8.1{\cdot}10^4$	2 612	(475,	325,	1 812)	1.97%
EMR–VB	10^{-4}	60	26.3	3 243	(475,	344,	2 424)	2.44%
EMR–WB	10^{-6}	438	$1.6{\cdot}10^4$	4 862	(539,	4 181,	142)	3.67%
EMR–WB	10^{-4}	50	26.4	5 189	(539,	4 181,	469)	3.91%
AGDSW	10^{-5}	1 093	$3.4{\cdot}10^5$	1 935	(419,	476,	1 040)	1.46%
AGDSW	0.001	57	25.3	2 375	(419,	698,	1 258)	1.79%
AGDSW	0.1	53	20.5	2 395	(419,	703,	1 273)	1.81%
AGDSW–S	10^{-5}	1 039	$2.4{\cdot}10^5$	2 021	(419,	550,	1 052)	1.52%
AGDSW–S	0.001	57	25.3	2 375	(419,	698,	1 258)	1.79%
AGDSW–S	0.1	53	20.2	2 414	(419,	718,	1 277)	1.82%
AGDSW–M	10^{-5}	1 446	$1.7{\cdot}10^5$	2 174	(419,	591,	1 164)	1.64%
AGDSW–M	0.001	57	25.3	2 375	(419,	698,	1 258)	1.79%
AGDSW–M	0.1	49	18.6	2 696	(419,	725,	1 552)	2.03%

Table B.1.: **(Model problem (1))** Results for the coefficient function in fig. 2.2, the diffusion problem, different methods and tolerances for the selection of eigenvectors: iteration count, condition number, resulting coarse space dimension, and coarse space dimension over the size of the stiffness matrix. The number of coarse functions associated with subdomain vertices, edges, and faces is given in parentheses.

method	*tol*	it.	κ	$\dim V_0$	($\mathcal{V}$,	$\mathcal{E}$,	$\mathcal{F}$,	$\mathcal{S}$)	$\frac{\dim V_0}{\text{dof}}$
GDSW	—	1 059	$4.7 \cdot 10^5$	1 980	(419,	927,	634,	—)	1.49%
RGDSW	—	983	$4.7 \cdot 10^5$	435	(—,	—,	— ,	435)	0.33%
AGDSW	10^{-5}	1 093	$3.4 \cdot 10^5$	1 935	(419,	476,	1 040,	—)	1.46%
AGDSW	0.001	57	25.3	2 375	(419,	698,	1 258,	—)	1.79%
AGDSW	0.1	53	20.5	2 395	(419,	703,	1 273,	—)	1.81%
AGDSW–S	10^{-5}	1 039	$2.4 \cdot 10^5$	2 021	(419,	550,	1 052,	—)	1.52%
AGDSW–S	0.001	57	25.3	2 375	(419,	698,	1 258,	—)	1.79%
AGDSW–S	0.1	53	20.2	2 414	(419,	718,	1 277,	—)	1.82%
R–WB–AGDSW	10^{-5}	1 020	$2.9 \cdot 10^5$	1 396	(—,	—,	1 040,	356)	1.05%
R–WB–AGDSW	0.001	56	26.3	1 830	(—,	—,	1 258,	572)	1.38%
R–WB–AGDSW	0.1	55	21.6	1 849	(—,	—,	1 273,	576)	1.39%
R–WB–AGDSW–S	10^{-5}	947	$2.1 \cdot 10^5$	1 479	(—,	—,	1 052,	427)	1.11%
R–WB–AGDSW–S	0.001	56	26.3	1 830	(—,	—,	1 258,	572)	1.38%
R–WB–AGDSW–S	0.1	54	21.7	1 856	(—,	—,	1 277,	579)	1.40%
RAGDSW	10^{-5}	911	$2.1 \cdot 10^5$	926	(—,	—,	— ,	926)	0.70%
RAGDSW	0.001	59	32.0	1 287	(—,	—,	— ,	1 287)	0.97%
RAGDSW	0.1	56	32.0	1 316	(—,	—,	— ,	1 316)	0.99%
RAGDSW–S	10^{-5}	1 102	$1.7 \cdot 10^5$	1 037	(—,	—,	— ,	1 037)	0.78%
RAGDSW–S	0.001	59	32.0	1 289	(—,	—,	— ,	1 289)	0.97%
RAGDSW–S	0.1	57	32.0	1 320	(—,	—,	— ,	1 320)	1.00%

Table B.2.: **(Model problem (1))** Results for the coefficient function in fig. 2.2, the diffusion problem, different methods and tolerances for the selection of eigenvectors: iteration count, condition number, resulting coarse space dimension, and coarse space dimension over the size of the stiffness matrix. The number of coarse functions associated with subdomain vertices, edges, faces, wire basket and interface stars is given in parentheses. $\mathcal{S}$ refers to $\mathcal{S}_\mathcal{W}/\mathcal{S}_\Gamma$.

method	*tol*	it.	κ	$\dim V_0$	($\mathcal{V}$,	$\mathcal{E}$,	$\mathcal{F}$,	$\mathcal{S}_\Gamma$)	$\frac{\dim V_0}{\mathrm{dof}}$
AGDSW	10^{-5}	1 093	$3.4{\cdot}10^5$	1 935	(419,	476,	1 040,	—)	1.46%
AGDSW	0.001	57	25.3	2 375	(419,	698,	1 258,	—)	1.79%
AGDSW	0.1	53	20.5	2 395	(419,	703,	1 273,	—)	1.81%
RAGDSW	10^{-5}	911	$2.1{\cdot}10^5$	926	(—,	—,	— ,	926)	0.70%
RAGDSW	0.001	59	32.0	1 287	(—,	—,	— ,	1 287)	0.97%
RAGDSW	0.1	56	32.0	1 316	(—,	—,	— ,	1 316)	0.99%
OS–ACMS–M	10^{-5}	1 045	$9.6{\cdot}10^4$	909	(419,	301,	189,	—)	0.69%
OS–ACMS–M	0.001	54	27.8	968	(419,	352,	197,	—)	0.73%
OS–ACMS–M	0.1	41	12.5	1 597	(419,	443,	735,	—)	1.20%
OS–ACMS–K	10^{-5}	1 835	$1.2{\cdot}10^5$	798	(419,	204,	175,	—)	0.60%
OS–ACMS–K	0.001	54	27.8	968	(419,	352,	197,	—)	0.73%
OS–ACMS–K	0.1	47	19.2	1 093	(419,	368,	306,	—)	0.82%
OS–ACMS–S–M	10^{-5}	599	$6.0{\cdot}10^4$	935	(419,	327,	189,	—)	0.70%
OS–ACMS–S–M	0.001	54	27.8	968	(419,	352,	197,	—)	0.73%
OS–ACMS–S–M	0.1	41	12.2	1 609	(419,	455,	735,	—)	1.21%
OS–ACMS–S–K	10^{-5}	1 366	$1.0{\cdot}10^5$	864	(419,	270,	175,	—)	0.65%
OS–ACMS–S–K	0.001	54	27.8	968	(419,	352,	197,	—)	0.73%
OS–ACMS–S–K	0.1	44	16.4	1 095	(419,	370,	306,	—)	0.83%

Table B.3.: **(Model problem (1))** Results for the coefficient function in fig. 2.2, the diffusion problem, different methods and tolerances for the selection of eigenvectors: iteration count, condition number, resulting coarse space dimension, and coarse space dimension over the size of the stiffness matrix. The number of coarse functions associated with subdomain vertices, edges, faces, and interface stars is given in parentheses.

method	tol	it.	κ	$\dim V_0$	($\mathcal{V}$,	$\mathcal{E}$,	$\mathcal{F}$,	$\mathcal{S}$)	$\frac{\dim V_0}{\text{dof}}$
R–WB–AGDSW	0.001	56	26.3	1 830	(— ,	— ,	1 258,	572)	1.38%
RAGDSW	10^{-5}	911	$2.1 \cdot 10^5$	926	(— ,	— ,	— ,	926)	0.70%
RAGDSW	0.001	59	32.0	1 287	(— ,	— ,	— ,	1 287)	0.97%
RAGDSW	0.1	56	32.0	1 316	(— ,	— ,	— ,	1 316)	0.99%
OS–ACMS–K	10^{-5}	1 835	$1.2 \cdot 10^5$	798	(419,	204,	175,	—)	0.60%
OS–ACMS–K	0.001	54	27.8	968	(419,	352,	197,	—)	0.73%
OS–ACMS–K	0.1	47	19.2	1 093	(419,	368,	306,	—)	0.82%
R–WB–OS–ACMS–M	10^{-5}	1 316	$1.8 \cdot 10^5$	623	(— ,	— ,	189,	434)	0.47%
R–WB–OS–ACMS–M	0.001	54	27.2	769	(— ,	— ,	197,	572)	0.58%
R–WB–OS–ACMS–M	0.1	44	14.3	1 316	(— ,	— ,	735,	581)	0.99%
R–WB–OS–ACMS–K	10^{-5}	1 280	$3.1 \cdot 10^5$	531	(— ,	— ,	175,	356)	0.40%
R–WB–OS–ACMS–K	0.001	54	27.2	769	(— ,	— ,	197,	572)	0.58%
R–WB–OS–ACMS–K	0.1	46	15.1	882	(— ,	— ,	306,	576)	0.66%
R–WB–OS–ACMS–S–M	10^{-5}	1 104	$1.2 \cdot 10^5$	691	(— ,	— ,	189,	502)	0.52%
R–WB–OS–ACMS–S–M	0.001	54	27.1	769	(— ,	— ,	197,	572)	0.58%
R–WB–OS–ACMS–S–M	0.1	44	13.2	1 338	(— ,	— ,	735,	603)	1.01%
R–WB–OS–ACMS–S–K	10^{-5}	1 513	$2.1 \cdot 10^5$	602	(— ,	— ,	175,	427)	0.45%
R–WB–OS–ACMS–S–K	0.001	55	27.1	769	(— ,	— ,	197,	572)	0.58%
R–WB–OS–ACMS–S–K	0.1	46	15.1	885	(— ,	— ,	306,	579)	0.67%

Table B.4.: **(Model problem (1))** Results for the coefficient function in fig. 2.2, the diffusion problem, different methods and tolerances for the selection of eigenvectors: iteration count, condition number, resulting coarse space dimension, and coarse space dimension over the size of the stiffness matrix. The number of coarse functions associated with subdomain vertices, edges, faces, wire basket and interface stars is given in parentheses. $\mathcal{S}$ refers to $\mathcal{S}_\mathcal{W}/\mathcal{S}_\Gamma$.

method	*tol*	it.	κ	$\dim V_0$	($\mathcal{V}$,	$\mathcal{E}$,	$\mathcal{F}$,	$\mathcal{S}_\Gamma$)	$\frac{\dim V_0}{\mathrm{dof}}$
AGDSW	0.001	57	25.3	2 375	(419,	698,	1 258,	—)	1.79%
AGDSW(3)	0.001	56	27.0	2 647	(419,	936,	1 292,	—)	2.00%
AGDSW(1)	0.001	57	26.8	2 670	(419,	945,	1 306,	—)	2.01%
AGDSW–S	0.001	57	25.3	2 375	(419,	698,	1 258,	—)	1.79%
AGDSW–S(3)	0.001	57	27.1	2 647	(419,	936,	1 292,	—)	2.00%
AGDSW–S(1)	0.001	57	26.8	2 670	(419,	945,	1 306,	—)	2.01%
RAGDSW	0.001	59	32.0	1 287	(—,	—,	—,	1 287)	0.97%
RAGDSW(3)	0.001	59	31.9	1 288	(—,	—,	—,	1 288)	0.97%
RAGDSW(1)	0.001	59	31.4	1 339	(—,	—,	—,	1 339)	1.01%
RAGDSW–S	0.001	59	32.0	1 289	(—,	—,	—,	1 289)	0.97%
RAGDSW–S(3)	0.001	59	31.9	1 290	(—,	—,	—,	1 290)	0.97%
RAGDSW–S(1)	0.001	59	31.4	1 340	(—,	—,	—,	1 340)	1.01%

Table B.5.: **(Model problem (1))** Results for the coefficient function in fig. 2.2, the diffusion problem, different methods, and a tolerance of 0.001 for the selection of eigenvectors: iteration count, condition number, resulting coarse space dimension, and coarse space dimension over the size of the stiffness matrix. If the slab variant is used, the slab width in layers of finite elements is appended in parentheses to the method's name. The number of coarse functions associated with subdomain vertices, edges, faces, and interface stars is given in parentheses.

method	*tol*	it.	κ	dim V_0	($\mathcal{V}$, $\mathcal{E}$, $\mathcal{F}$, $\mathcal{S_W}$)	$\frac{\dim V_0}{\text{dof}}$
OS–ACMS–K	0.001	54	27.8	968	(419, 352, 197, —)	0.73%
OS–ACMS–K(3)	0.001	54	29.0	978	(419, 362, 197, —)	0.74%
OS–ACMS–K(1)	0.001	54	29.6	988	(419, 365, 204, —)	0.74%
OS–ACMS–S–K	0.001	54	27.8	968	(419, 352, 197, —)	0.73%
OS–ACMS–S–K(3)	0.001	54	28.9	978	(419, 362, 197, —)	0.74%
OS–ACMS–S–K(1)	0.001	54	29.6	988	(419, 365, 204, —)	0.74%
R–WB–OS–ACMS–K	0.001	54	27.2	769	(— , — , 197, 572)	0.58%
R–WB–OS–ACMS–K(3)	0.001	54	27.2	801	(— , — , 197, 604)	0.60%
R–WB–OS–ACMS–K(1)	0.001	54	28.1	817	(— , — , 204, 613)	0.62%
R–WB–OS–ACMS–S–K	0.001	55	27.1	769	(— , — , 197, 572)	0.58%
R–WB–OS–ACMS–S–K(3)	0.001	54	27.2	801	(— , — , 197, 604)	0.60%
R–WB–OS–ACMS–S–K(1)	0.001	54	28.1	817	(— , — , 204, 613)	0.62%

Table B.6.: **(Model problem (1))** Results for the coefficient function in fig. 2.2, the diffusion problem, different methods, and a tolerance of 0.001 for the selection of eigenvectors: iteration count, condition number, resulting coarse space dimension, and coarse space dimension over the size of the stiffness matrix. If the slab variant is used, the slab width in layers of finite elements is appended in parentheses to the method's name. The number of coarse functions associated with subdomain vertices, edges, faces, and wire basket stars is given in parentheses.

method	*tol*	it.	κ	$\dim V_0$	($\mathcal{V}$,	$\mathcal{E}$,	$\mathcal{F}$,	$\mathcal{S}_\Gamma$)	$\frac{\dim V_0}{\mathrm{dof}}$
AGDSW	0.001	57	25.3	2 375	(419,	698,	1 258,	—)	1.79%
AGDSW–ℓ(K)	0.001	57	25.3	2 375	(419,	698,	1 258,	—)	1.79%
AGDSW–S	0.001	57	25.3	2 375	(419,	698,	1 258,	—)	1.79%
AGDSW–S–ℓ(K)	0.001	57	25.3	2 375	(419,	698,	1 258,	—)	1.79%
AGDSW–M	0.001	57	25.3	2 375	(419,	698,	1 258,	—)	1.79%
AGDSW–ℓ(M)	0.001	57	25.3	2 375	(419,	698,	1 258,	—)	1.79%
RAGDSW	0.001	59	32.0	1 287	(—,	—,	— ,	1 287)	0.97%
RAGDSW–ℓ(K)	0.001	59	32.0	1 287	(—,	—,	— ,	1 287)	0.97%
RAGDSW–S	0.001	59	32.0	1 289	(—,	—,	— ,	1 289)	0.97%
RAGDSW–S–ℓ(K)	0.001	59	32.0	1 289	(—,	—,	— ,	1 289)	0.97%
RAGDSW–M	0.001	59	32.0	1 287	(—,	—,	— ,	1 287)	0.97%
RAGDSW–ℓ(M)	0.001	59	32.0	1 287	(—,	—,	— ,	1 287)	0.97%

Table B.7.: **(Model problem (1))** Results for the coefficient function in fig. 2.2, the diffusion problem, different methods, and a tolerance of 0.001 for the selection of eigenvectors: iteration count, condition number, resulting coarse space dimension, and coarse space dimension over the size of the stiffness matrix. If a lumped matrix is used, $\ell(K)$ or $\ell(M)$ is appended to the method's name. The number of coarse functions associated with subdomain vertices, edges, faces, and interface stars is given in parentheses.

method	*tol*	it.	κ	$\dim V_0$	$(\mathcal{V},\ \mathcal{E},\mathcal{F},\mathcal{S}_\mathcal{W})$	$\frac{\dim V_0}{\text{dof}}$
OS–ACMS–K	0.001	44	16.3	115	(70, 44, 1, —)	0.21%
OS–ACMS–K(3)	0.001	45	16.3	115	(70, 44, 1, —)	0.21%
OS–ACMS–K(1)	0.001	44	16.4	116	(70, 45, 1, —)	0.21%
OS–ACMS–S–K	0.001	43	16.3	115	(70, 44, 1, —)	0.21%
OS–ACMS–S–K(3)	0.001	45	16.3	115	(70, 44, 1, —)	0.21%
OS–ACMS–S–K(1)	0.001	45	16.4	116	(70, 45, 1, —)	0.21%
R–WB–OS–ACMS–K	0.001	49	15.7	94	(—, —, 1, 93)	0.17%
R–WB–OS–ACMS–K(3)	0.001	49	15.7	94	(—, —, 1, 93)	0.17%
R–WB–OS–ACMS–K(1)	0.001	49	15.8	95	(—, —, 1, 94)	0.17%
R–WB–OS–ACMS–S–K	0.001	49	15.7	94	(—, —, 1, 93)	0.17%
R–WB–OS–ACMS–S–K(3)	0.001	49	15.7	94	(—, —, 1, 93)	0.17%
R–WB–OS–ACMS–S–K(1)	0.001	49	15.8	95	(—, —, 1, 94)	0.17%

Table B.8.: **(Model problem (2))** Results for the coefficient function in fig. 2.3, the diffusion problem, different methods, and a tolerance of 0.001 for the selection of eigenvectors: iteration count, condition number, resulting coarse space dimension, and coarse space dimension over the size of the stiffness matrix. If the slab variant is used, the slab width in layers of finite elements is appended in parentheses to the method's name. The number of coarse functions associated with subdomain vertices, edges, faces, and wire basket stars is given in parentheses.

method	*tol*	it.	κ	dim V_0	$\frac{\dim V_0}{\text{dof}}$
AGDSW	0.01	192.1 (226)	388.6 (992.8)	9 601.3 (9 891)	2.1% (2.2%)
AGDSW	0.03	148.1 (171)	206.4 (444.2)	9 942.3 (10 193)	2.2% (2.3%)
AGDSW	0.1	92.2 (103)	75.0 (123.0)	10 788.2 (11 009)	2.4% (2.4%)
AGDSW–S	0.01	174.8 (206)	323.9 (890.3)	9 853.0 (10 125)	2.2% (2.2%)
AGDSW–S	0.03	132.1 (149)	165.1 (350.0)	10 195.7 (10 430)	2.3% (2.3%)
AGDSW–S	0.1	82.8 (93)	60.3 (98.5)	11 113.0 (11 338)	2.5% (2.5%)
AGDSW–M	0.01	160.7 (194)	263.6 (500.6)	10 007.0 (10 240)	2.2% (2.3%)
AGDSW–M	0.03	115.7 (136)	122.9 (267.6)	10 728.1 (10 928)	2.4% (2.4%)
AGDSW–M	0.1	68.1 (74)	39.4 (57.3)	13 580.5 (13 787)	3.0% (3.0%)
R–WB–AGDSW	0.01	195.1 (227)	377.1 (842.8)	6 262.1 (6 505)	1.4% (1.4%)
R–WB–AGDSW	0.03	145.9 (165)	197.1 (443.9)	6 617.7 (6 825)	1.5% (1.5%)
R–WB–AGDSW	0.1	89.8 (100)	66.0 (112.1)	7 445.6 (7 621)	1.6% (1.7%)
R–WB–AGDSW–S	0.01	174.0 (208)	295.9 (755.2)	6 548.3 (6 769)	1.4% (1.5%)
R–WB–AGDSW–S	0.03	128.1 (148)	150.9 (281.5)	6 894.3 (7 094)	1.5% (1.6%)
R–WB–AGDSW–S	0.1	81.6 (90)	55.2 (89.9)	7 798.7 (7 966)	1.7% (1.8%)
RAGDSW	0.01	177.8 (199)	302.6 (845.8)	4 040.3 (4 221)	0.9% (0.9%)
RAGDSW	0.03	131.8 (147)	154.5 (258.1)	4 366.6 (4 526)	1.0% (1.0%)
RAGDSW	0.1	83.9 (95)	59.4 (95.6)	5 195.3 (5 338)	1.1% (1.2%)
RAGDSW–S	0.01	155.3 (173)	237.4 (606.8)	4 402.1 (4 556)	1.0% (1.0%)
RAGDSW–S	0.03	118.0 (137)	125.6 (234.9)	4 714.1 (4 859)	1.0% (1.1%)
RAGDSW–S	0.1	76.7 (86)	50.1 (85.8)	5 616.4 (5 779)	1.2% (1.3%)

Table B.9.: **(Model problem (4))** Average results (maximum in parentheses) for 100 randomly generated coefficient functions (cf. section 2.4), the diffusion problem, different methods and tolerances for the selection of eigenvectors: iteration count, condition number, resulting coarse space dimension, and coarse space dimension over the size of the stiffness matrix.

method	*tol*	it.	κ	$\dim V_0$	$\frac{\dim V_0}{\mathrm{dof}}$
OS–ACMS–M	0.01	122.3 (143)	165.7 (346.0)	3 895.6 (3 996)	0.9% (0.9%)
OS–ACMS–M	0.03	91.0 (106)	83.2 (143.5)	4 706.5 (4 842)	1.0% (1.1%)
OS–ACMS–M	0.05	77.2 (96)	58.3 (106.8)	5 476.7 (5 610)	1.2% (1.2%)
OS–ACMS–M	0.1	58.8 (68)	31.1 (59.0)	7 856.1 (7 967)	1.7% (1.8%)
OS–ACMS–S–M	0.01	108.9 (130)	134.5 (331.2)	4 029.9 (4 124)	0.9% (0.9%)
OS–ACMS–S–M	0.03	81.4 (101)	68.6 (156.8)	4 857.6 (5 012)	1.1% (1.1%)
OS–ACMS–S–M	0.05	69.5 (83)	49.0 (156.6)	5 639.6 (5 761)	1.2% (1.3%)
OS–ACMS–S–M	0.1	54.2 (61)	27.3 (50.7)	8 089.3 (8 205)	1.8% (1.8%)
OS–ACMS–K	0.01	144.5 (170)	245.4 (592.9)	3 619.7 (3 715)	0.8% (0.8%)
OS–ACMS–K	0.03	118.6 (142)	152.6 (335.1)	3 771.6 (3 877)	0.8% (0.9%)
OS–ACMS–K	0.05	102.5 (126)	108.5 (251.6)	3 974.3 (4 092)	0.9% (0.9%)
OS–ACMS–K	0.1	79.3 (93)	62.6 (139.2)	4 544.1 (4 644)	1.0% (1.0%)
OS–ACMS–S–K	0.01	128.8 (162)	195.4 (587.5)	3 747.4 (3 846)	0.8% (0.8%)
OS–ACMS–S–K	0.03	104.6 (127)	116.9 (244.1)	3 910.5 (4 023)	0.9% (0.9%)
OS–ACMS–S–K	0.05	91.6 (113)	87.2 (242.7)	4 121.4 (4 245)	0.9% (0.9%)
OS–ACMS–S–K	0.1	71.0 (83)	51.5 (129.0)	4 708.3 (4 825)	1.0% (1.1%)

Table B.10.: **(Model problem (4))** Average results (maximum in parentheses) for 100 randomly generated coefficient functions (cf. section 2.4), the diffusion problem, different methods and tolerances for the selection of eigenvectors: iteration count, condition number, resulting coarse space dimension, and coarse space dimension over the size of the stiffness matrix.

method	*tol*	it.	κ	$\dim V_0$	$\frac{\dim V_0}{\text{dof}}$
R–WB–OS–ACMS–M	0.01	144.8 (167)	221.2 (431.1)	2 831.2 (2 920)	0.6% (0.6%)
R–WB–OS–ACMS–M	0.03	101.9 (117)	103.6 (211.7)	3 578.4 (3 672)	0.8% (0.8%)
R–WB–OS–ACMS–M	0.05	82.8 (93)	66.3 (156.5)	4 229.9 (4 332)	0.9% (1.0%)
R–WB–OS–ACMS–M	0.1	62.1 (70)	35.0 (60.4)	6 286.0 (6 399)	1.4% (1.4%)
R–WB–OS–ACMS–S–M	0.01	121.7 (141)	164.7 (468.1)	3 035.6 (3 116)	0.7% (0.7%)
R–WB–OS–ACMS–S–M	0.03	86.8 (104)	78.5 (156.8)	3 805.9 (3 897)	0.8% (0.9%)
R–WB–OS–ACMS–S–M	0.05	71.3 (83)	50.1 (156.5)	4 495.0 (4 603)	1.0% (1.0%)
R–WB–OS–ACMS–S–M	0.1	55.3 (62)	27.9 (51.1)	6 771.4 (6 875)	1.5% (1.5%)
R–WB–OS–ACMS–K	0.01	178.3 (204)	341.6 (765.3)	2 525.7 (2 632)	0.6% (0.6%)
R–WB–OS–ACMS–K	0.03	133.3 (154)	181.8 (347.3)	2 720.9 (2 804)	0.6% (0.6%)
R–WB–OS–ACMS–K	0.05	110.5 (126)	124.4 (232.4)	2 906.2 (2 993)	0.6% (0.7%)
R–WB–OS–ACMS–K	0.1	82.4 (96)	64.2 (170.8)	3 357.6 (3 438)	0.7% (0.8%)
R–WB–OS–ACMS–S–K	0.01	150.6 (183)	248.3 (594.7)	2 739.7 (2 841)	0.6% (0.6%)
R–WB–OS–ACMS–S–K	0.03	110.9 (135)	129.2 (256.3)	2 923.0 (3 013)	0.6% (0.7%)
R–WB–OS–ACMS–S–K	0.05	94.9 (113)	95.3 (243.2)	3 116.8 (3 202)	0.7% (0.7%)
R–WB–OS–ACMS–S–K	0.1	71.5 (83)	49.9 (128.9)	3 615.3 (3 684)	0.8% (0.8%)

Table B.11.: **(Model problem (4))** Average results (maximum in parentheses) for 100 randomly generated coefficient functions (cf. section 2.4), the diffusion problem, different methods and tolerances for the selection of eigenvectors: iteration count, condition number, resulting coarse space dimension, and coarse space dimension over the size of the stiffness matrix.

method	*tol*	it.	κ	$\dim V_0$	$\frac{\dim V_0}{\text{dof}}$
RAGDSW	0.01	177.8 (199)	302.6 (845.8)	4 040.3 (4 221)	0.9% (0.9%)
RAGDSW	0.03	131.8 (147)	154.5 (258.1)	4 366.6 (4 526)	1.0% (1.0%)
RAGDSW	0.1	83.9 (95)	59.4 (95.6)	5 195.3 (5 338)	1.1% (1.2%)
OS–ACMS–K	0.01	144.5 (170)	245.4 (592.9)	3 619.7 (3 715)	0.8% (0.8%)
OS–ACMS–K	0.03	118.6 (142)	152.6 (335.1)	3 771.6 (3 877)	0.8% (0.9%)
OS–ACMS–K	0.1	79.3 (93)	62.6 (139.2)	4 544.1 (4 644)	1.0% (1.0%)
OS–ACMS–S–K	0.01	128.8 (162)	195.4 (587.5)	3 747.4 (3 846)	0.8% (0.8%)
OS–ACMS–S–K	0.03	104.6 (127)	116.9 (244.1)	3 910.5 (4 023)	0.9% (0.9%)
OS–ACMS–S–K	0.1	71.0 (83)	51.5 (129.0)	4 708.3 (4 825)	1.0% (1.1%)
R–WB–OS–ACMS–K	0.01	178.3 (204)	341.6 (765.3)	2 525.7 (2 632)	0.6% (0.6%)
R–WB–OS–ACMS–K	0.03	133.3 (154)	181.8 (347.3)	2 720.9 (2 804)	0.6% (0.6%)
R–WB–OS–ACMS–K	0.1	82.4 (96)	64.2 (170.8)	3 357.6 (3 438)	0.7% (0.8%)
R–WB–OS–ACMS–S–K	0.01	150.6 (183)	248.3 (594.7)	2 739.7 (2 841)	0.6% (0.6%)
R–WB–OS–ACMS–S–K	0.03	110.9 (135)	129.2 (256.3)	2 923.0 (3 013)	0.6% (0.7%)
R–WB–OS–ACMS–S–K	0.1	71.5 (83)	49.9 (128.9)	3 615.3 (3 684)	0.8% (0.8%)
R–WB–OS–ACMS–S–ℓ(K)	0.01	145.7 (178)	229.5 (592.2)	2 792.4 (2 894)	0.6% (0.6%)
R–WB–OS–ACMS–S–ℓ(K)	0.03	105.5 (123)	115.0 (213.8)	3 127.9 (3 211)	0.7% (0.7%)
R–WB–OS–ACMS–S–ℓ(K)	0.1	68.3 (80)	45.2 (128.4)	4 248.6 (4 349)	0.9% (1.0%)

Table B.12.: **(Model problem (4))** Average results (maximum in parentheses) for 100 randomly generated coefficient functions (cf. section 2.4), the diffusion problem, different methods and tolerances for the selection of eigenvectors: iteration count, condition number, resulting coarse space dimension, and coarse space dimension over the size of the stiffness matrix. If a lumped stiffness matrix is used, $\ell(K)$ is appended to the method's name.

B.2. Linear Elasticity Problems

method	tol	it.	κ	$\dim V_0$	($\mathcal{F}$,	$\mathcal{S}$)	$\frac{\dim V_0}{\text{dof}}$
RAGDSW	0.005	136	298.9	7 028	(— ,	7 028)	1.77%
RAGDSW	0.05	67	34.6	7 156	(— ,	7 156)	1.80%
RAGDSW–ℓ(K)	0.005	155	300.4	7 026	(— ,	7 026)	1.77%
RAGDSW–ℓ(K)	0.03	77	63.4	7 093	(— ,	7 093)	1.78%
RAGDSW–ℓ(K)	0.05	64	28.6	7 221	(— ,	7 221)	1.81%
RAGDSW–M	0.001	380	1 265.2	6 965	(— ,	6 965)	1.75%
RAGDSW–M	0.01	77	65.1	7 067	(— ,	7 067)	1.78%
RAGDSW–ℓ(M)	0.001	322	925.0	6 979	(— ,	6 979)	1.75%
RAGDSW–ℓ(M)	0.01	81	65.2	7 047	(— ,	7 047)	1.77%
RAGDSW–S	0.005	93	207.6	7 055	(— ,	7 055)	1.77%
RAGDSW–S	0.05	68	34.4	7 219	(— ,	7 219)	1.81%
RAGDSW–S–ℓ(K)	0.005	102	303.8	7 054	(— ,	7 054)	1.77%
RAGDSW–S–ℓ(K)	0.03	71	41.6	7 135	(— ,	7 135)	1.79%
R–WB–OS–ACMS–S–K	0.001	467	5 197.4	4 292	(1 536,	2 756)	1.08%
R–WB–OS–ACMS–S–K	0.01	67	32.5	4 545	(1 742,	2 803)	1.14%
R–WB–OS–ACMS–S–ℓ(K)	0.001	485	5 044.8	4 303	(1 548,	2 755)	1.08%
R–WB–OS–ACMS–S–ℓ(K)	0.005	69	33.6	4 464	(1 668,	2 796)	1.12%
R–WB–OS–ACMS–S–ℓ(K)	0.01	66	30.8	4 622	(1 820,	2 802)	1.16%

Table B.13.: **(Model problem (1))** Results for the coefficient function in fig. 2.2, the equations of linear elasticity, different methods and tolerances for the selection of eigenvectors: iteration count, condition number, resulting coarse space dimension, and coarse space dimension over the size of the stiffness matrix. If a lumped matrix is used, $\ell(K)$ or $\ell(M)$ is appended to the method's name. The number of coarse functions associated with subdomain faces and wire basket and interface stars is given in parentheses.

method	*tol*	it.	κ	$\dim V_0$	($\mathcal{V}$,	$\mathcal{E}$,	$\mathcal{F}$,	$\mathcal{S}$)	$\frac{\dim V_0}{\text{dof}}$
AGDSW–M	0.005	112	275.5	11 344	(1 257,	3 285,	6 802,	—)	2.85%
AGDSW–M	0.01	81	47.7	11 401	(1 257,	3 307,	6 837,	—)	2.86%
AGDSW–M	0.1	58	24.2	15 895	(1 257,	3 636,	11 002,	—)	3.99%
OS–ACMS–M	0.001	460	5 356.4	5 314	(1 257,	2 458,	1 599,	—)	1.34%
OS–ACMS–M	0.005	83	1 260.7	5 642	(1 257,	2 503,	1 882,	—)	1.42%
OS–ACMS–M	0.01	58	26.0	5 945	(1 257,	2 507,	2 181,	—)	1.49%
OS–ACMS–M	0.05	53	20.3	7 628	(1 257,	2 606,	3 765,	—)	1.92%
OS–ACMS–S–M	0.001	332	2 418.6	5 339	(1 257,	2 483,	1 599,	—)	1.34%
OS–ACMS–S–M	0.005	59	26.2	5 646	(1 257,	2 507,	1 882,	—)	1.42%
OS–ACMS–S–M	0.01	58	26.0	5 950	(1 257,	2 512,	2 181,	—)	1.50%
OS–ACMS–S–M	0.05	52	19.9	7 702	(1 257,	2 680,	3 765,	—)	1.94%
R–WB–OS–ACMS–M	0.001	520	4 060.8	4 330	(— ,	— ,	1 599,	2 731)	1.09%
R–WB–OS–ACMS–M	0.005	90	1 157.0	4 666	(— ,	— ,	1 882,	2 784)	1.17%
R–WB–OS–ACMS–M	0.01	62	26.1	4 970	(— ,	— ,	2 181,	2 789)	1.25%
R–WB–OS–ACMS–M	0.05	59	21.9	6 602	(— ,	— ,	3 765,	2 837)	1.66%
R–WB–OS–ACMS–S–M	0.001	320	2 796.0	4 369	(— ,	— ,	1 599,	2 770)	1.10%
R–WB–OS–ACMS–S–M	0.005	66	30.6	4 684	(— ,	— ,	1 882,	2 802)	1.18%
R–WB–OS–ACMS–S–M	0.01	64	25.9	4 996	(— ,	— ,	2 181,	2 815)	1.26%
R–WB–OS–ACMS–S–M	0.05	56	20.7	6 776	(— ,	— ,	3 765,	3 011)	1.70%

Table B.14.: **(Model problem (1))** Results for the coefficient function in fig. 2.2, the equations of linear elasticity, different methods and tolerances for the selection of eigenvectors: iteration count, condition number, resulting coarse space dimension, and coarse space dimension over the size of the stiffness matrix. The number of coarse functions associated with subdomain vertices, edges, faces, wire basket and interface stars is given in parentheses.

method	*tol*	it.	κ	$\dim V_0$	($\mathcal{F}$, $\mathcal{S}$)	$\frac{\dim V_0}{\text{dof}}$
RAGDSW	0.005	89	52.0	661	(—, 661)	0.39%
RAGDSW	0.05	79	37.8	673	(—, 673)	0.40%
RAGDSW–ℓ(K)	0.005	89	52.0	661	(—, 661)	0.39%
RAGDSW–ℓ(K)	0.03	83	41.8	668	(—, 668)	0.40%
RAGDSW–M	0.001	132	518.2	658	(—, 658)	0.39%
RAGDSW–M	0.01	86	41.8	665	(—, 665)	0.40%
RAGDSW–ℓ(M)	0.001	114	322.5	659	(—, 659)	0.39%
RAGDSW–ℓ(M)	0.01	82	41.8	662	(—, 662)	0.39%
RAGDSW–S	0.005	85	52.0	662	(—, 662)	0.39%
RAGDSW–S	0.05	76	36.9	704	(—, 704)	0.42%
RAGDSW–S–ℓ(K)	0.005	85	52.0	662	(—, 662)	0.39%
RAGDSW–S–ℓ(K)	0.03	82	41.8	690	(—, 690)	0.41%
R–WB–OS–ACMS–S–K	0.001	127	2 211.0	570	(24, 546)	0.34%
R–WB–OS–ACMS–S–K	0.01	62	26.7	611	(62, 549)	0.36%
R–WB–OS–ACMS–S–ℓ(K)	0.001	124	2 321.6	575	(29, 546)	0.34%
R–WB–OS–ACMS–S–ℓ(K)	0.01	61	26.7	625	(76, 549)	0.37%

Table B.15.: **(Model problem (2))** Results for the coefficient function in fig. 2.3, the equations of linear elasticity, different methods and tolerances for the selection of eigenvectors: iteration count, condition number, resulting coarse space dimension, and coarse space dimension over the size of the stiffness matrix. If a lumped matrix is used, $\ell(K)$ or $\ell(M)$ is appended to the method's name. The number of coarse functions associated with subdomain faces and wire basket and interface stars is given in parentheses.

method	*tol*	it.	κ	$\dim V_0$	($\mathcal{V}$,	$\mathcal{E}$,	$\mathcal{F}$,	$\mathcal{S}$)	$\frac{\dim V_0}{\text{dof}}$
AGDSW–M	0.001	310	4 328.7	2 581	(210,	1 101,	1 270,	—)	1.53%
AGDSW–M	0.005	87	208.4	2 600	(210,	1 114,	1 276,	—)	1.55%
AGDSW–M	0.05	59	28.0	3 151	(210,	1 118,	1 823,	—)	1.87%
OS–ACMS–M	0.001	178	1 215.4	850	(210,	590,	50,	—)	0.51%
OS–ACMS–M	0.005	62	26.8	901	(210,	603,	88,	—)	0.54%
OS–ACMS–M	0.05	46	13.9	1 782	(210,	668,	904,	—)	1.06%
OS–ACMS–S–M	0.001	184	1 140.3	852	(210,	592,	50,	—)	0.51%
OS–ACMS–S–M	0.005	61	26.8	901	(210,	603,	88,	—)	0.54%
OS–ACMS–S–M	0.05	46	13.9	1 806	(210,	692,	904,	—)	1.07%
R–WB–OS–ACMS–M	0.001	107	1 383.4	597	(—,	—,	50,	547)	0.36%
R–WB–OS–ACMS–M	0.005	61	26.7	637	(—,	—,	88,	549)	0.38%
R–WB–OS–ACMS–M	0.05	49	15.8	1 453	(—,	—,	904,	549)	0.86%
R–WB–OS–ACMS–S–M	0.001	110	2 293.1	597	(—,	—,	50,	547)	0.36%
R–WB–OS–ACMS–S–M	0.005	61	26.7	637	(—,	—,	88,	549)	0.38%
R–WB–OS–ACMS–S–M	0.05	48	15.4	1 469	(—,	—,	904,	565)	0.87%

Table B.16.: **(Model problem (2))** Results for the coefficient function in fig. 2.3, the equations of linear elasticity, different methods and tolerances for the selection of eigenvectors: iteration count, condition number, resulting coarse space dimension, and coarse space dimension over the size of the stiffness matrix. The number of coarse functions associated with subdomain vertices, edges, faces, wire basket and interface stars is given in parentheses.

method	*tol*	it.	κ	$\dim V_0$	($\mathcal{F}$,	$\mathcal{S}$)	$\frac{\dim V_0}{\text{dof}}$
RAGDSW	0.005	243	433.6	3 680	(— ,	3 680)	0.21%
RAGDSW	0.05	84	52.6	4 574	(— ,	4 574)	0.26%
RAGDSW–ℓ(K)	0.005	220	428.8	3 709	(— ,	3 709)	0.21%
RAGDSW–ℓ(K)	0.03	96	60.9	4 566	(— ,	4 566)	0.26%
RAGDSW–M	0.005	169	216.2	3 855	(— ,	3 855)	0.22%
RAGDSW–M	0.03	82	44.2	7 285	(— ,	7 285)	0.41%
RAGDSW–ℓ(M)	0.005	166	214.3	3 778	(— ,	3 778)	0.21%
RAGDSW–ℓ(M)	0.03	73	35.5	5 789	(— ,	5 789)	0.33%
RAGDSW–S	0.01	135	138.8	3 854	(— ,	3 854)	0.22%
RAGDSW–S	0.05	70	40.9	5 203	(— ,	5 203)	0.29%
RAGDSW–S–ℓ(K)	0.01	129	115.6	3 960	(— ,	3 960)	0.22%
RAGDSW–S–ℓ(K)	0.03	87	51.9	5 094	(— ,	5 094)	0.29%
R–WB–OS–ACMS–S–K	0.005	301	2 386.0	2 502	(370,	2 132)	0.14%
R–WB–OS–ACMS–S–K	0.01	163	351.2	2 708	(545,	2 163)	0.15%
R–WB–OS–ACMS–S–K	0.03	75	38.6	3 402	(1 186,	2 216)	0.19%
R–WB–OS–ACMS–S–K	0.05	64	32.1	4 075	(1 827,	2 248)	0.23%
R–WB–OS–ACMS–S–K	0.1	48	15.7	5 832	(3 387,	2 445)	0.33%
R–WB–OS–ACMS–S–ℓ(K)	0.005	283	1 569.5	2 646	(513,	2 133)	0.15%
R–WB–OS–ACMS–S–ℓ(K)	0.01	146	231.9	2 986	(820,	2 166)	0.17%
R–WB–OS–ACMS–S–ℓ(K)	0.03	78	39.2	4 295	(2 076,	2 219)	0.24%
R–WB–OS–ACMS–S–ℓ(K)	0.05	57	21.9	5 454	(3 192,	2 262)	0.31%

Table B.17.: **(Model problem (3))** Results for the coefficient function in fig. 2.4, the equations of linear elasticity, different methods and tolerances for the selection of eigenvectors: iteration count, condition number, resulting coarse space dimension, and coarse space dimension over the size of the stiffness matrix. If a lumped matrix is used, $\ell(K)$ or $\ell(M)$ is appended to the method's name. The number of coarse functions associated with subdomain faces and wire basket and interface stars is given in parentheses.

method	*tol*	it.	κ	$\dim V_0$	($\mathcal{V}$,	$\mathcal{E}$,	$\mathcal{F}$,	$\mathcal{S}$)	$\frac{\dim V_0}{\mathrm{dof}}$
AGDSW–M	0.01	214	289.3	9 842	(984,	3 509,	5 349,	—)	0.56%
AGDSW–M	0.03	95	58.8	12 512	(984,	3 613,	7 915,	—)	0.71%
AGDSW–M	0.05	70	32.8	15 276	(984,	3 658,	10 634,	—)	0.86%
OS–ACMS–M	0.01	177	239.9	5 311	(984,	2 407,	1 920,	—)	0.30%
OS–ACMS–M	0.03	79	40.9	8 497	(984,	2 646,	4 867,	—)	0.48%
OS–ACMS–M	0.05	62	26.1	11 479	(984,	2 802,	7 693,	—)	0.65%
OS–ACMS–S–M	0.01	172	259.0	5 348	(984,	2 444,	1 920,	—)	0.30%
OS–ACMS–S–M	0.03	72	42.3	8 582	(984,	2 731,	4 867,	—)	0.49%
OS–ACMS–S–M	0.05	51	24.0	11 631	(984,	2 954,	7 693,	—)	0.66%
R–WB–OS–ACMS–M	0.01	123	123.1	4 079	(—,	— ,	1 920,	2 159)	0.23%
R–WB–OS–ACMS–M	0.03	71	38.7	7 068	(—,	— ,	4 867,	2 201)	0.40%
R–WB–OS–ACMS–M	0.05	54	23.1	9 964	(—,	— ,	7 693,	2 271)	0.56%
R–WB–OS–ACMS–S–M	0.01	110	93.2	4 114	(—,	— ,	1 920,	2 194)	0.23%
R–WB–OS–ACMS–S–M	0.03	62	34.8	7 180	(—,	— ,	4 867,	2 313)	0.41%
R–WB–OS–ACMS–S–M	0.05	50	19.8	10 260	(—,	— ,	7 693,	2 567)	0.58%

Table B.18.: **(Model problem (3))** Results for the coefficient function in fig. 2.4, the equations of linear elasticity, different methods and tolerances for the selection of eigenvectors: iteration count, condition number, resulting coarse space dimension, and coarse space dimension over the size of the stiffness matrix. The number of coarse functions associated with subdomain vertices, edges, faces, wire basket and interface stars is given in parentheses.

method	*tol*	it.	κ	dim V_0	$\frac{\dim V_0}{\mathrm{dof}}$
OS–ACMS–K	0.01	227.3 (283)	625.9 (2 224.3)	21 569.9 (22 093)	1.6% (1.6%)
OS–ACMS–K	0.03	133.7 (155)	173.0 (356.3)	23 678.2 (24 165)	1.7% (1.8%)
OS–ACMS–K	0.05	104.5 (121)	97.3 (296.4)	25 407.8 (25 936)	1.9% (1.9%)
OS–ACMS–K	0.1	76.0 (88)	48.4 (82.3)	29 311.4 (29 925)	2.2% (2.2%)
OS–ACMS–S–K	0.01	195.8 (235)	506.2 (1 795.2)	22 287.0 (22 806)	1.6% (1.7%)
OS–ACMS–S–K	0.03	118.8 (139)	146.1 (368.5)	24 460.2 (24 977)	1.8% (1.8%)
OS–ACMS–S–K	0.05	93.0 (111)	79.1 (144.9)	26 298.2 (26 856)	1.9% (2.0%)
OS–ACMS–S–K	0.1	68.3 (81)	39.6 (83.0)	30 615.3 (31 256)	2.3% (2.3%)
OS–ACMS–M	0.005	226.6 (290)	745.4 (2 450.3)	22 656.6 (23 138)	1.7% (1.7%)
OS–ACMS–M	0.01	159.4 (198)	283.2 (1 218.9)	24 467.5 (25 034)	1.8% (1.8%)
OS–ACMS–M	0.03	95.5 (108)	81.6 (163.1)	29 946.5 (30 580)	2.2% (2.3%)
OS–ACMS–M	0.05	77.8 (90)	52.9 (95.0)	35 074.1 (35 711)	2.6% (2.6%)
OS–ACMS–S–M	0.005	189.1 (237)	463.5 (1 876.5)	23 377.2 (23 863)	1.7% (1.8%)
OS–ACMS–S–M	0.01	135.9 (166)	214.3 (954.1)	25 211.0 (25 816)	1.9% (1.9%)
OS–ACMS–S–M	0.03	84.3 (98)	63.6 (114.7)	30 916.3 (31 565)	2.3% (2.3%)
OS–ACMS–S–M	0.05	69.2 (83)	44.0 (79.0)	36 421.1 (37 083)	2.7% (2.7%)

Table B.19.: **(Model problem (4))** Average results (maximum in parentheses) for 100 randomly generated coefficient functions (cf. section 2.4), the equations of linear elasticity, different methods and tolerances for the selection of eigenvectors: iteration count, condition number, resulting coarse space dimension, and coarse space dimension over the size of the stiffness matrix.

method	*tol*	it.	κ	$\dim V_0$	$\frac{\dim V_0}{\text{dof}}$
R–WB–OS–ACMS–K	0.01	235.4 (283)	694.4 (2 652.2)	17 298.0 (17 767)	1.3% (1.3%)
R–WB–OS–ACMS–K	0.03	138.8 (159)	177.8 (335.7)	19 419.7 (19 905)	1.4% (1.5%)
R–WB–OS–ACMS–K	0.05	108.5 (123)	102.2 (200.8)	21 096.5 (21 591)	1.6% (1.6%)
R–WB–OS–ACMS–K	0.1	79.8 (91)	51.6 (85.5)	24 770.4 (25 361)	1.8% (1.9%)
R–WB–OS–ACMS–S–K	0.01	193.0 (235)	509.2 (3 801.9)	18 233.9 (18 726)	1.3% (1.4%)
R–WB–OS–ACMS–S–K	0.03	117.7 (139)	139.6 (328.6)	20 511.3 (21 057)	1.5% (1.6%)
R–WB–OS–ACMS–S–K	0.05	93.5 (108)	79.0 (197.6)	22 440.6 (23 007)	1.7% (1.7%)
R–WB–OS–ACMS–S–K	0.1	69.4 (81)	40.9 (73.5)	26 993.8 (27 625)	2.0% (2.0%)
R–WB–OS–ACMS–M	0.005	235.2 (290)	822.2 (5 815.7)	18 375.2 (18 864)	1.4% (1.4%)
R–WB–OS–ACMS–M	0.01	163.6 (208)	293.3 (2 185.2)	20 187.5 (20 705)	1.5% (1.5%)
R–WB–OS–ACMS–M	0.03	99.2 (116)	87.4 (171.0)	25 486.9 (26 110)	1.9% (1.9%)
R–WB–OS–ACMS–M	0.05	80.6 (90)	55.6 (96.5)	30 341.0 (30 943)	2.2% (2.3%)
R–WB–OS–ACMS–S–M	0.005	189.0 (249)	466.2 (3 556.4)	19 315.2 (19 836)	1.4% (1.5%)
R–WB–OS–ACMS–S–M	0.01	137.9 (164)	208.6 (1 002.0)	21 194.2 (21 816)	1.6% (1.6%)
R–WB–OS–ACMS–S–M	0.03	85.4 (104)	66.5 (169.1)	27 044.5 (27 694)	2.0% (2.0%)
R–WB–OS–ACMS–S–M	0.05	69.5 (81)	40.8 (77.1)	32 658.5 (33 274)	2.4% (2.5%)

Table B.20.: **(Model problem (4))** Average results (maximum in parentheses) for 100 randomly generated coefficient functions (cf. section 2.4), the equations of linear elasticity, different methods and tolerances for the selection of eigenvectors: iteration count, condition number, resulting coarse space dimension, and coarse space dimension over the size of the stiffness matrix.

Bibliography

[AH02]   AARNES, Jørg and HOU, Thomas Y., "Multiscale Domain Decomposition
            Methods for Elliptic Problems with High Aspect Ratios," *Acta Math.
            Appl. Sin. Engl. Ser.*, vol. 18, no. 1, pp. 63–76, 2002. DOI: 10.1007/
            s102550200004 (cit. on p. 10).

[AGL05]   AHRENS, James, GEVECI, Berk, and LAW, Charles, "ParaView: An End-
            User Tool for Large-Data Visualization," in *The Visualization Handbook*,
            Elsevier Butterworth-Heinemann, 2005, ch. 36, pp. 717–731, ISBN: 978-0-
            12-387582-2. DOI: 10.1016/B978-012387582-2/50038-1 (cit. on p. vi).

[AG19]   AL DAAS, Hussam and GRIGORI, Laura, "A Class of Efficient Locally
            Constructed Preconditioners Based on Coarse Spaces," *SIAM J. Matrix
            Anal. Appl.*, vol. 40, no. 1, pp. 66–91, 2019. DOI: 10.1137/18M1194365
            (cit. on p. 223).

[AGJT21]   AL DAAS, Hussam, GRIGORI, Laura, JOLIVET, Pierre, and TOURNIER,
            Pierre-Henri, "A Multilevel Schwarz Preconditioner Based on a Hierarchy
            of Robust Coarse Spaces," *SIAM J. Sci. Comput.*, vol. 43, no. 3, A1907–
            A1928, 2021. DOI: 10.1137/19M1266964 (cit. on p. 223).

[AJ22]   AL DAAS, Hussam and JOLIVET, Pierre, "A Robust Algebraic Multilevel
            Domain Decomposition Preconditioner for Sparse Symmetric Positive
            Definite Matrices," *SIAM J. Sci. Comput.*, vol. 44, no. 4, A2582–A2598,
            2022. DOI: 10.1137/21M1446320 (cit. on p. 223).

[Aya15]   AYACHIT, Utkarsh, *The ParaView Guide: A Parallel Visualization Appli-
            cation.* Kitware, Inc., 2015, ISBN: 978-1930934306 (cit. on p. vi).

Bibliography

[BBO04] Babuška, Ivo, Banerjee, Uday, and Osborn, John E., "Generalized Finite Element Methods — Main Ideas, Results and Perspective," *Int. J. Comput. Methods.*, vol. 1, no. 1, pp. 67–103, 2004. DOI: 10.1142/S0219876204000083 (cit. on p. 8).

[BO83] Babuška, Ivo and Osborn, John E., "Generalized Finite Element Methods: Their Performance and Their Relation to Mixed Methods," *SIAM J. Numer. Anal.*, vol. 20, no. 3, pp. 510–536, 1983. DOI: 10.1137/0720034 (cit. on p. 8).

[BDF+15] Balzani, Daniel, Deparis, Simone, Fausten, Simon, Forti, Davide, Heinlein, Alexander, Klawonn, Axel, Quarteroni, Alfio, Rheinbach, Oliver, and Schröder, Jörg, "Numerical Modeling of Fluid-Structure Interaction in Arteries with Anisotropic Polyconvex Hyperelastic and Anisotropic Viscoelastic Material Models at Finite Strains," *Int. J. Numer. Meth. Biomed. Engng.*, vol. 32, no. 10, e02756, 2015. DOI: 10.1002/cnm.2756 (cit. on p. 19).

[BSSS22] Bastian, Peter, Scheichl, Robert, Seelinger, Linus, and Strehlow, Arne, "Multilevel Spectral Domain Decomposition," *SIAM J. Sci. Comput.*, S1–S26, 2022, Special Section: 2021 Copper Mountain Conference. DOI: 10.1137/21M1427231 (cit. on p. 223).

[BPS+17] Beirão da Veiga, Lourenço, Pavarino, Luca F., Scacchi, Simone, Widlund, Olof B., and Zampini, Stefano, "Adaptive Selection of Primal Constraints for Isogeometric BDDC Deluxe Preconditioners," *SIAM J. Sci. Comput.*, vol. 39, no. 1, A281–A302, 2017. DOI: 10.1137/15M1054675 (cit. on p. 9).

[BKK01] Bjørstad, Petter, Koster, Jacko, and Krzyżanowski, Piotr, "Domain Decomposition Solvers for Large Scale Industrial Finite Element Problems," in *Applied Parallel Computing: New Paradigms for HPC in Industry and Academia*, ser. Lect. Notes Comput. Sci. 5th International Workshop,

PARA 2000, Bergen, Norway, June 18-21, 2000, vol. 1947, Springer, 2001, pp. 373–383, ISBN: 3-540-41729-X. DOI: `10.1007/3-540-70734-4_44` (cit. on p. 9).

[BDG+22] BOOTLAND, Niall, DOLEAN, Victorita, GRAHAM, Ivan G., MA, Chupeng, and SCHEICHL, Robert, "Overlapping Schwarz Methods with GenEO Coarse Spaces for Indefinite and Non-self-adjoint Problems," *IMA J. Numer. Anal.*, pp. 1–38, 2022. DOI: `10.1093/imanum/drac036` (cit. on p. 224).

[BDJT21] BOOTLAND, Niall, DOLEAN, Victorita, JOLIVET, Pierre, and TOURNIER, Pierre-Henri, "A Comparison of Coarse Spaces for Helmholtz Problems in the High Frequency Regime," *Comput. Math. Appl.*, vol. 98, pp. 239–253, 2021. DOI: `10.1016/j.camwa.2021.07.011` (cit. on p. 224).

[Bra07] BRAESS, Dietrich, *Finite Elements: Theory, Fast Solvers, and Applications in Solid Mechanics*, 3rd ed. Cambridge University Press, 2007, ISBN: 978-0-511-27910-2. DOI: `10.1017/CBO9780511618635` (cit. on pp. 16, 17).

[BS08] BRENNER, Susanne C. and SCOTT, L. Ridgway, *The Mathematical Theory of Finite Element Methods*, 3rd ed., ser. Texts in Applied Mathematics. Springer, 2008, vol. 15, ISBN: 978-0-387-75933-3. DOI: `10.1007/978-0-387-75934-0` (cit. on pp. 7, 121, 229).

[BHMV99] BREZINA, Marian, HEBERTON, Caroline, MANDEL, Jan, and VANĚK, Petr, "An Iterative Method with Convergence Rate Chosen a Priori," University of Colorado Denver, Denver, Colorado, Tech. Rep., 1999 (cit. on p. 123).

[Buc13] BUCK, Marco, "Overlapping Domain Decomposition Preconditioners for Multi-Phase Elastic Composites," dissertation, Technische Universität Kaiserslautern, Kaiserslautern, Germany, 2013. URN: `urn:nbn:de:hbz:386-kluedo-35651` (cit. on p. 10).

[BIA13] BUCK, Marco, ILIEV, Oleg, and ANDRÄ, Heiko, "Multiscale Finite Element Coarse Spaces for the Application to Linear Elasticity," *Open Mathematics*, vol. 11, no. 4, pp. 680–701, 2013. DOI: `10.2478/s11533-012-0166-8`

(cit. on p. 10).

[BIA14]   Buck, Marco, Iliev, Oleg, and Andrä, Heiko, "Multiscale Finite Elements for Linear Elasticity: Oscillatory Boundary Conditions," in *Domain Decomposition Methods in Science and Engineering XXI*, ser. Lect. Notes Comput. Sci. Eng. DD 2012, vol. 98, Springer, 2014, pp. 237–245, isbn: 978-3-319-05789-7. doi: `10.1007/978-3-319-05789-7_20` (cit. on p. 10).

[BIA18]   Buck, Marco, Iliev, Oleg, and Andrä, Heiko, "Domain Decomposition Preconditioners for Multiscale Problems in Linear Elasticity," *Numer. Linear Algebra Appl.*, vol. 25, no. 5, e2171, 2018. doi: `10.1002/nla.2171` (cit. on p. 10).

[BDR+20]   Butler, Richard, Dodwell, Tim J., Reinarz, Anne, Sandhu, Anhad, Scheichl, Robert, and Seelinger, Linus, "High-Performance DUNE Modules for Solving Large-Scale, Strongly Anisotropic Elliptic Problems with Applications to Aerospace Composites," *Comput. Phys. Commun.*, vol. 249, p. 106 997, 2020. doi: `10.1016/j.cpc.2019.106997` (cit. on p. 10).

[CS99]   Cai, Xiao-Chuan and Sarkis, Marcus, "A Restricted Additive Schwarz Preconditioner for General Sparse Linear Systems," *SIAM J. Sci. Comput.*, vol. 21, no. 2, pp. 792–797, 1999. doi: `10.1137/S106482759732678X` (cit. on p. 23).

[CW16]   Calvo, Juan G. and Widlund, Olof B., "An Adaptive Choice of Primal Constraints for BDDC Domain Decomposition Algorithms," *Electron. Trans. Numer. Anal.*, vol. 45, pp. 524–544, 2016 (cit. on p. 9).

[CG17]   Ciaramella, Gabriele and Gander, Martin J., "Analysis of the Parallel Schwarz Method for Growing Chains of Fixed-Sized Subdomains: Part I," *SIAM J. Numer. Anal.*, vol. 55, no. 3, pp. 1330–1356, 2017. doi: `10.1137/16M1065215` (cit. on p. 21).

[CG18]   Ciaramella, Gabriele and Gander, Martin J., "Analysis of the Parallel Schwarz Method for Growing Chains of Fixed-Sized Subdomains: Part II," *SIAM J. Numer. Anal.*, vol. 56, no. 3, pp. 1498–1524, 2018. doi: `10.1137/17M1115885` (cit. on pp. 21, 69).

[Cia88]   Ciarlet, Philippe G., *Mathematical Elasticity: Vol. I: Three-Dimensional Elasticity*, ser. Studies in Mathematics and its Applications. North Holland, 1988, vol. 20, isbn: 0-444-70259-8 (cit. on pp. 16, 17).

[Cia02]   Ciarlet, Philippe G., *The Finite Element Method for Elliptic Problems.* SIAM, 2002, isbn: 978-0-898715-14-9. doi: `10.1137/1.9780898719208` (cit. on pp. 7, 17, 107, 225, 226).

[Cia13]   Ciarlet, Philippe G., *Linear and Nonlinear Functional Analysis with Applications.* SIAM, 2013, isbn: 978-1-611972-58-0 (cit. on pp. 107, 226).

[CB68]   Craig Jr., Roy R. and Bampton, Mervyn C. C., "Coupling of Substructures for Dynamic Analyses," *AIAA Journal*, vol. 6, no. 7, pp. 1313–1319, 1968. doi: `10.2514/3.4741` (cit. on p. 146).

[DKW08a] Dohrmann, Clark R., Klawonn, Axel, and Widlund, Olof B., "A Family of Energy Minimizing Coarse Spaces for Overlapping Schwarz Preconditioners," in *Domain Decomposition Methods in Science and Engineering XVII*, ser. Lect. Notes Comput. Sci. Eng. DD 2006, vol. 60, Springer, 2008, pp. 247–254, isbn: 978-3-540-75199-1. doi: `10.1007/978-3-540-75199-1_28` (cit. on pp. 8, 23, 26).

[DKW08b] Dohrmann, Clark R., Klawonn, Axel, and Widlund, Olof B., "Domain Decomposition for Less Regular Subdomains: Overlapping Schwarz in Two Dimensions," *SIAM J. Numer. Anal.*, vol. 46, no. 4, pp. 2153–2168, 2008. doi: `10.1137/070685841` (cit. on pp. 8, 15, 23).

[DKW08c] Dohrmann, Clark R., Klawonn, Axel, and Widlund, Olof B., "Extending Theory for Domain Decomposition Algorithms to Irregular Subdomains," in *Domain Decomposition Methods in Science and Engineering*

XVII, ser. Lect. Notes Comput. Sci. Eng. DD 2006, vol. 60, Springer, 2008, pp. 255–261, ISBN: 978-3-540-75199-1. DOI: `10.1007/978-3-540-75199-1_29` (cit. on p. 23).

[DW09]   DOHRMANN, Clark R. and WIDLUND, Olof B., "An Overlapping Schwarz Algorithm for Almost Incompressible Elasticity," *SIAM J. Numer. Anal.*, vol. 47, no. 4, pp. 2897–2923, 2009. DOI: `10.1137/080724320` (cit. on pp. 19, 23, 28).

[DW10]   DOHRMANN, Clark R. and WIDLUND, Olof B., "Hybrid Domain Decomposition Algorithms for Compressible and Almost Incompressible Elasticity," *Int. J. Numer. Methods Eng.*, vol. 82, no. 2, pp. 157–183, 2010. DOI: `10.1002/nme.2761` (cit. on pp. 28, 73).

[DW12]   DOHRMANN, Clark R. and WIDLUND, Olof B., "An Alternative Coarse Space for Irregular Subdomains and an Overlapping Schwarz Algorithm for Scalar Elliptic Problems in the Plane," *SIAM J. Numer. Anal.*, vol. 50, no. 5, pp. 2522–2537, 2012. DOI: `10.1137/110853959` (cit. on p. 73).

[DW14]   DOHRMANN, Clark R. and WIDLUND, Olof B., "Lower Dimensional Coarse Spaces for Domain Decomposition," in *Domain Decomposition Methods in Science and Engineering XXI*, ser. Lect. Notes Comput. Sci. Eng. DD 2012, vol. 98, Springer, 2014, pp. 527–535, ISBN: 978-3-319-05789-7. DOI: `10.1007/978-3-319-05789-7_50` (cit. on p. 73).

[DW16]   DOHRMANN, Clark R. and WIDLUND, Olof B., "A BDDC Algorithm with Deluxe Scaling for Three-Dimensional H(curl) Problems," *Commun. Pure Appl. Math.*, vol. 69, no. 4, pp. 745–770, 2016. DOI: `10.1002/cpa.21574` (cit. on p. 55).

[DW17]   DOHRMANN, Clark R. and WIDLUND, Olof B., "On the Design of Small Coarse Spaces for Domain Decomposition Algorithms," *SIAM J. Sci. Comput.*, vol. 39, no. 4, A1466–A1488, 2017. DOI: `10.1137/17M1114272` (cit. on pp. 73, 75, 89, 91).

[DJN15]   DOLEAN, Victorita, JOLIVET, Pierre, and NATAF, Frédéric, *An Introduc-tion to Domain Decomposition Methods: Algorithms, Theory, and Parallel Implementation.* SIAM, 2015, ISBN: 978-1-61197-406-5. DOI: 10.1137/1. 9781611974065 (cit. on p. 10).

[DNSS12]   DOLEAN, Victorita, NATAF, Frédéric, SCHEICHL, Robert, and SPILLANE, Nicole, "Analysis of a Two-Level Schwarz Method with Coarse Spaces Based on Local Dirichlet-to-Neumann Maps," *Comput. Methods Appl. Math.*, vol. 12, no. 4, pp. 391–414, 2012. DOI: 10.2478/cmam-2012-0027 (cit. on pp. 9, 10, 47, 62, 123).

[DSW96]   DRYJA, Maksymilian, SARKIS, Marcus V., and WIDLUND, Olof B., "Multi-level Schwarz Methods for Elliptic Problems with Discontinuous Coefficients in Three Dimensions," *Numer. Math.*, vol. 72, no. 3, pp. 313–348, 1996. DOI: 10.1007/s002110050172 (cit. on p. 8).

[DSW94]   DRYJA, Maksymilian, SMITH, Barry F., and WIDLUND, Olof B., "Schwarz Analysis of Iterative Substructuring Algorithms for Elliptic Problems in Three Dimensions," *SIAM J. Numer. Anal.*, vol. 31, no. 6, pp. 1662–1694, 1994. DOI: 10.1137/0731086 (cit. on p. 23).

[DW94]   DRYJA, Maksymilian and WIDLUND, Olof B., "Domain Decomposition Algorithms with Small Overlap," *SIAM J. Sci. Comput.*, vol. 15, no. 3, pp. 604–620, 1994 (cit. on p. 105).

[DSYG18]   DUERSCH, Jed A., SHAO, Meiyue, YANG, Chao, and GU, Ming, "A Robust and Efficient Implementation of LOBPCG," *SIAM J. Sci. Comput.*, vol. 40, no. 5, pp. C655–C676, 2018. DOI: 10.1137/17M1129830 (cit. on pp. 210, 213).

[EE03]   E, Weinan and ENGQUIST, Bjorn, "The Heterogeneous Multiscale Meth-ods," *Comm. Math. Sci.*, vol. 1, no. 1, pp. 87–132, 2003 (cit. on p. 8).

[EEL+07]   E, Weinan, ENGQUIST, Bjorn, LI, Xiantao, REN, Weiqing, and VANDEN-EIJNDEN, Eric, "The Heterogeneous Multiscale Method: A Review," *Com-

mun. Comput. Phys., vol. 2, no. 3, pp. 367–450, 2007 (cit. on p. 8).

[EGLW12] EFENDIEV, Yalchin, GALVIS, Juan, LAZAROV, Raytcho, and WILLEMS, Jörg, "Robust Domain Decomposition Preconditioners for Abstract Symmetric Positive Definite Bilinear Forms," *ESAIM: Math. Model. Numer. Anal.*, vol. 46, no. 5, pp. 1175–1199, 2012. DOI: `10.1051/m2an/2011073` (cit. on pp. 9, 10, 47, 123).

[EH09] EFENDIEV, Yalchin and HOU, Thomas Y., *Multiscale Finite Element Methods: Theory and Applications*, ser. Surveys and Tutorials in the Applied Mathematical Sciences. Springer, 2009, vol. 4, ISBN: 978-0-387-09496-0. DOI: `10.1007/978-0-387-09496-0` (cit. on pp. 8, 10, 146).

[EG03] EFSTATHIOU, Evridiki and GANDER, Martin J., "Why Restricted Additive Schwarz Converges Faster than Additive Schwarz," *BIT Numerical Mathematics*, vol. 43, no. 5, pp. 945–959, 2003. DOI: `10.1023/B:BITN.0000014563.33622.1d` (cit. on p. 23).

[EF18] EIDEL, Bernhard and FISCHER, Andreas, "The Heterogeneous Multiscale Finite Element Method for the Homogenization of Linear Elastic Solids and a Comparison with the FE2 Method," *Comput. Methods Appl. Mech. Engrg.*, vol. 329, pp. 332–368, 2018. DOI: `10.1016/j.cma.2017.10.001` (cit. on p. 8).

[EMR19] EIKELAND, Erik, MARCINKOWSKI, Leszek, and RAHMAN, Talal, "Overlapping Schwarz Methods with Adaptive Coarse Spaces for Multiscale Problems in 3D," *Numer. Math.*, vol. 142, no. 1, pp. 103–128, 2019. DOI: `10.1007/s00211-018-1008-9` (cit. on pp. 5, 9–12, 47, 62, 63, 67, 117, 123, 148, 150).

[Eva10] EVANS, Lawrence C., *Partial Differential Equations*, 2nd ed., ser. Graduate Studies in Mathematics. AMS, 2010, vol. 19, ISBN: 978-0-8218-4974-3 (cit. on p. 7).

[FQD17]   FORTI, Davide, QUARTERONI, Alfio, and DEPARIS, Simone, "A Parallel Algorithm for the Solution of Large-Scale Nonconforming Fluid-Structure Interaction Problems in Hemodynamics," *J. Comp. Math.*, vol. 35, no. 3, pp. 363–380, 2017. DOI: `10.4208/jcm.1702-m2016-0630` (cit. on p. 19).

[Fra16]   FRANK, Robert H., *Success and Luck: Good Fortune and the Myth of Meritocracy.* Princeton University Press, 2016, ISBN: 9781400880270. DOI: `10.1515/9781400880270` (cit. on p. v).

[GE10a]   GALVIS, Juan and EFENDIEV, Yalchin, "Domain Decomposition Preconditioners for Multiscale Flows in High Contrast Media: Reduced Dimension Coarse Spaces," *Multiscale Model. Simul.*, vol. 8, no. 5, pp. 1621–1644, 2010. DOI: `10.1137/100790112` (cit. on pp. 9, 10, 31).

[GE10b]   GALVIS, Juan and EFENDIEV, Yalchin, "Domain Decomposition Preconditioners for Multiscale Flows in High-Contrast Media," *Multiscale Model. Simul.*, vol. 8, no. 4, pp. 1461–1483, 2010. DOI: `10.1137/090751190` (cit. on pp. 9, 10, 32, 47, 123).

[GL17]   GANDER, Martin J. and LONELAND, Atle, "SHEM: An Optimal Coarse Space for RAS and Its Multiscale Approximation," in *Domain Decomposition Methods in Science and Engineering XXIII*, ser. Lect. Notes Comput. Sci. Eng. DD 2015, vol. 116, Springer, 2017, pp. 313–321, ISBN: 978-3-319-52389-7. DOI: `10.1007/978-3-319-52389-7_32` (cit. on pp. 9–11, 56, 148, 150).

[GLR15]   GANDER, Martin J., LONELAND, Atle, and RAHMAN, Talal, *Analysis of a New Harmonically Enriched Multiscale Coarse Space for Domain Decomposition Methods*, Dec. 2015. arXiv: `1512.05285v1` `[math.NA]` (cit. on pp. 5, 9–12, 47, 56, 62, 117, 123, 130, 148, 150, 214).

[GR09]   GEUZAINE, Christophe and REMACLE, Jean-François, "Gmsh: A 3-D Finite Element Mesh Generator with Built-In Pre- and Post-processing Facilities," *Int. J. Numer. Methods Eng.*, vol. 79, no. 11, pp. 1309–1331, 2009. DOI:

10.1002/nme.2579 (cit. on p. vi).

[GHK+19a] GIESE, Daniel, HEINLEIN, Alexander, KLAWONN, Axel, KNEPPER, Jascha, and SONNABEND, Kristina, "Comparison of MRI Measurements and CFD Simulations of Hemodynamics in Intracranial Aneurysms Using a 3D Printed Model – A Benchmark Problem," *PAMM*, vol. 19, no. 1, e201900398, 2019. DOI: 10.1002/pamm.201900398 (cit. on p. vi).

[GHK+19b] GIESE, Daniel, HEINLEIN, Alexander, KLAWONN, Axel, KNEPPER, Jascha, and SONNABEND, Kristina, "Comparison of MRI Measurements and CFD Simulations of Hemodynamics in Intracranial Aneurysms Using a 3D Printed Model – Influence of Noisy MRI Measurements," *PAMM*, vol. 19, no. 1, e201900401, 2019. DOI: 10.1002/pamm.201900401 (cit. on p. vi).

[GS21] GOUARIN, Loïc and SPILLANE, Nicole, *Fully Algebraic Domain Decomposition Preconditioners with Adaptive Spectral Bounds*, Jun. 2021. arXiv: 2106.10913v1 [math.NA] (cit. on p. 223).

[GLS07] GRAHAM, Ivan G., LECHNER, Patrick O., and SCHEICHL, Robert, "Domain Decomposition for Multiscale PDEs," *Numer. Math.*, vol. 106, no. 4, pp. 589–626, 2007. DOI: 10.1007/s00211-007-0074-1 (cit. on pp. 8, 10, 19, 23).

[GS07] GRAHAM, Ivan G. and SCHEICHL, Robert, "Robust Domain Decomposition Algorithms for Multiscale PDEs," *Numer. Methods Partial Differ. Equ.*, vol. 23, no. 4, pp. 859–878, 2007. DOI: 10.1002/num.20254 (cit. on p. 10).

[Hac85] HACKBUSCH, Wolfgang, *Multi-Grid Methods and Applications*, ser. Springer Ser. Comput. Math. Springer, 1985, vol. 4, ISBN: 978-3-662-02427-0. DOI: 10.1007/978-3-662-02427-0 (cit. on p. 8).

[Hei16] HEINLEIN, Alexander, "Parallel Overlapping Schwarz Preconditioners and Multiscale Discretizations with Applications to Fluid-Structure Interaction and Highly Heterogeneous Problems," dissertation, University of Cologne, Cologne, Germany, Jun. 2016. URN: urn:nbn:de:hbz:38-68419 (cit. on

pp. 15, 19, 145, 214, 215).

[HHKR15]   HEINLEIN, Alexander, HETMANIUK, Ulrich L., KLAWONN, Axel, and RHEIN-
BACH, Oliver, "The Approximate Component Mode Synthesis Special Finite
Element Method in Two Dimensions: Parallel Implementation and Nu-
merical Results," *J. Comput. Appl. Math.*, vol. 289, pp. 116–133, 2015,
Sixth International Conference on Advanced Computational Methods in
Engineering (ACOMEN 2014). DOI: `10.1016/j.cam.2015.02.053` (cit. on
p. 145).

[HHK19]   HEINLEIN, Alexander, HOCHMUTH, Christian, and KLAWONN, Axel, "Mono-
lithic Overlapping Schwarz Domain Decomposition Methods with GDSW
Coarse Spaces for Incompressible Fluid Flow Problems," *SIAM J. Sci.
Comput.*, vol. 41, no. 4, pp. C291–C316, 2019. DOI: `10.1137/18M1184047`
(cit. on p. 28).

[HHK20]   HEINLEIN, Alexander, HOCHMUTH, Christian, and KLAWONN, Axel, "Re-
duced Dimension GDSW Coarse Spaces for Monolithic Schwarz Domain
Decomposition Methods for Incompressible Fluid Flow Problems," *Int. J.
Numer. Meth. Eng.*, vol. 121, no. 6, pp. 1101–1119, 2020. DOI: `10.1002/
nme.6258` (cit. on pp. 19, 73).

[HHK21]   HEINLEIN, Alexander, HOCHMUTH, Christian, and KLAWONN, Axel, "Fully
Algebraic Two-Level Overlapping Schwarz Preconditioners for Elasticity
Problems," in *Numerical Mathematics and Advanced Applications ENU-
MATH 2019*, ser. Lect. Notes Comput. Sci. Eng. Vol. 139, Springer, 2021,
pp. 531–539, ISBN: 978-3-030-55874-1. DOI: `10.1007/978-3-030-55874-
1_52` (cit. on p. 73).

[HKKR18a] HEINLEIN, Alexander, KLAWONN, Axel, KNEPPER, Jascha, and RHEIN-
BACH, Oliver, "An Adaptive GDSW Coarse Space for Two-Level Overlap-
ping Schwarz Methods in Two Dimensions," in *Domain Decomposition
Methods in Science and Engineering XXIV*, ser. Lect. Notes Comput. Sci.

Eng. DD 2017, vol. 125, Springer, 2018, pp. 373–382, ISBN: 978-3-319-93873-8. DOI: `10.1007/978-3-319-93873-8_35` (cit. on pp. 9, 11, 47).

[HKKR18b] Heinlein, Alexander, Klawonn, Axel, Knepper, Jascha, and Rheinbach, Oliver, "Multiscale Coarse Spaces for Overlapping Schwarz Methods Based on the ACMS Space in 2D," *Electron. Trans. Numer. Anal.*, vol. 48, pp. 156–182, 2018. DOI: `10.1553/etna_vol48s156` (cit. on pp. 9–11, 47, 55, 56, 62, 63, 105, 123, 134, 145, 148, 150, 181, 213, 215).

[HKKR19] Heinlein, Alexander, Klawonn, Axel, Knepper, Jascha, and Rheinbach, Oliver, "Adaptive GDSW Coarse Spaces for Overlapping Schwarz Methods in Three Dimensions," *SIAM J. Sci. Comput.*, vol. 41, no. 5, A3045–A3072, 2019. DOI: `10.1137/18M1220613` (cit. on pp. 9, 11, 39, 47, 49, 50, 55–58, 63, 67, 68, 78, 93, 105, 108, 123, 126, 129–134, 181).

[HKKR] Heinlein, Alexander, Klawonn, Axel, Knepper, Jascha, and Rheinbach, Oliver, *Multiscale Coarse Spaces for Overlapping Schwarz Methods Based on the ACMS Space in Three Dimensions*, In preparation (cit. on pp. 11, 62, 145).

[HKK+22] Heinlein, Alexander, Klawonn, Axel, Knepper, Jascha, Rheinbach, Oliver, and Widlund, Olof B., "Adaptive GDSW Coarse Spaces of Reduced Dimension for Overlapping Schwarz Methods," *SIAM J. Sci. Comput.*, vol. 44, no. 3, A1176–A1204, 2022. DOI: `10.1137/20M1364540` (cit. on pp. 9, 11, 27, 39, 40, 47, 49, 50, 58, 69, 73, 74, 77, 78, 89, 92–95, 105–109, 119, 123, 126, 129–134, 137, 181, 201).

[HKK20] Heinlein, Alexander, Klawonn, Axel, and Kühn, Martin J., "Local Spectra of Adaptive Domain Decomposition Methods," in *Domain Decomposition Methods in Science and Engineering XXV*, ser. Lect. Notes Comput. Sci. Eng. DD 2018, vol. 138, Springer, 2020, pp. 167–175, ISBN: 978-3-030-56750-7. DOI: `10.1007/978-3-030-56750-7_18` (cit. on p. 224).

[HKL22]   Heinlein, Alexander, Klawonn, Axel, and Lanser, Martin, "Adaptive Nonlinear Domain Decomposition Methods with an Application to the *p*-Laplacian," *SIAM J. Sci. Comput.*, S152–S172, 2022, Special Section: 2021 Copper Mountain Conference. DOI: `10.1137/21M1433605` (cit. on pp. 47, 150).

[HKLW19]   Heinlein, Alexander, Klawonn, Axel, Lanser, Martin, and Weber, Janine, "Machine Learning in Adaptive Domain Decomposition Methods—Predicting the Geometric Location of Constraints," *SIAM J. Sci. Comput.*, vol. 41, no. 6, A3887–A3912, 2019. DOI: `10.1137/18M1205364` (cit. on p. 214).

[HKLW20]   Heinlein, Alexander, Klawonn, Axel, Lanser, Martin, and Weber, Janine, "A Frugal FETI-DP and BDDC Coarse Space for Heterogeneous Problems," *Electron. Trans. Numer. Anal.*, vol. 53, pp. 562–591, 2020. DOI: `10.1553/etna_vol53s562` (cit. on pp. 9, 215).

[HKLW21a] Heinlein, Alexander, Klawonn, Axel, Lanser, Martin, and Weber, Janine, "Combining Machine Learning and Adaptive Coarse Spaces—A Hybrid Approach for Robust FETI-DP Methods in Three Dimensions," *SIAM J. Sci. Comput.*, vol. 43, no. 5, S816–S838, 2021, Special Section: 2020 Copper Mountain Conference. DOI: `10.1137/20M1344913` (cit. on p. 214).

[HKLW21b] Heinlein, Alexander, Klawonn, Axel, Lanser, Martin, and Weber, Janine, "Combining Machine Learning and Domain Decomposition Methods for the Solution of Partial Differential Equations—A Review," *GAMM-Mitteilungen*, vol. 44, no. 1, e202100001, 2021. DOI: `10.1002/gamm.202100001` (cit. on p. 214).

[HKLW21c] Heinlein, Alexander, Klawonn, Axel, Lanser, Martin, and Weber, Janine, "Predicting the Geometric Location of Critical Edges in Adaptive GDSW Overlapping Domain Decomposition Methods Using Deep Learn-

ing," University of Cologne, Cologne, Germany, Tech. Rep., Mar. 2021. URN: `urn:nbn:de:hbz:38-362571`, accepted for publication in revised form in Domain Decomposition Methods in Science and Engineering XXVI, 2022 (cit. on p. 214).

[HKRR20a]   HEINLEIN, Alexander, KLAWONN, Axel, RAJAMANICKAM, Sivasankaran, and RHEINBACH, Oliver, "FROSch: A Fast And Robust Overlapping Schwarz Domain Decomposition Preconditioner Based on Xpetra in Trilinos," in *Domain Decomposition Methods in Science and Engineering XXV*, ser. Lect. Notes Comput. Sci. Eng. DD 2018, vol. 138, Springer, 2020, pp. 176–184, ISBN: 978-3-030-56750-7. DOI: `10.1007/978-3-030-56750-7_19` (cit. on p. 28).

[HKR16a]   HEINLEIN, Alexander, KLAWONN, Axel, and RHEINBACH, Oliver, "A Parallel Implementation of a Two-Level Overlapping Schwarz Method with Energy-Minimizing Coarse Space Based on Trilinos," *SIAM J. Sci. Comput.*, vol. 38, no. 6, pp. C713–C747, 2016. DOI: `10.1137/16M1062843` (cit. on pp. 19, 28).

[HKR16b]   HEINLEIN, Alexander, KLAWONN, Axel, and RHEINBACH, Oliver, "Parallel Two-Level Overlapping Schwarz Methods in Fluid-Structure Interaction," in *Numerical Mathematics and Advanced Applications ENUMATH 2015*, ser. Lect. Notes Comput. Sci. Eng. Vol. 112, Springer, 2016, pp. 521–530, ISBN: 978-3-319-39929-4. DOI: `10.1007/978-3-319-39929-4_50` (cit. on p. 28).

[HKR17]   HEINLEIN, Alexander, KLAWONN, Axel, and RHEINBACH, Oliver, "Parallel Overlapping Schwarz with an Energy-Minimizing Coarse Space," in *Domain Decomposition Methods in Science and Engineering XXIII*, ser. Lect. Notes Comput. Sci. Eng. DD 2015, vol. 116, Springer, 2017, pp. 353–360, ISBN: 978-3-319-52389-7. DOI: `10.1007/978-3-319-52389-7_36` (cit. on p. 28).

[HKRR19]   Heinlein, Alexander, Klawonn, Axel, Rheinbach, Oliver, and Röver, Friederike, "A Three-Level Extension of the GDSW Overlapping Schwarz Preconditioner in Two Dimensions," in *Advanced Finite Element Methods with Applications: Selected Papers from the 30th Chemnitz Finite Element Symposium 2017*, ser. Lect. Notes Comput. Sci. Eng. FEM 2017, vol. 128, Springer, 2019, pp. 187–204, isbn: 978-3-030-14244-5. doi: `10.1007/978-3-030-14244-5_10` (cit. on p. 28).

[HKRR20b]   Heinlein, Alexander, Klawonn, Axel, Rheinbach, Oliver, and Röver, Friederike, "A Three-level Extension of the GDSW Overlapping Schwarz Preconditioner in Three Dimensions," in *Domain Decomposition Methods in Science and Engineering XXV*, ser. Lect. Notes Comput. Sci. Eng. DD 2018, vol. 138, Springer, 2020, pp. 185–192, isbn: 978-3-030-56750-7. doi: `10.1007/978-3-030-56750-7_20` (cit. on pp. 19, 28, 73, 223).

[HKRR21]   Heinlein, Alexander, Klawonn, Axel, Rheinbach, Oliver, and Röver, Friederike, "A Three-Level Extension for Fast and Robust Overlapping Schwarz (FROSch) Preconditioners with Reduced Dimensional Coarse Space," Technische Universität Bergakademie Freiberg, Freiberg, Germany, Tech. Rep., Jul. 2021. Available at: `https://tu-freiberg.de/fakult1/forschung/preprints`, accepted for publication in revised form in Domain Decomposition Methods in Science and Engineering XXVI, 2022 (cit. on p. 73).

[HKRW18]   Heinlein, Alexander, Klawonn, Axel, Rheinbach, Oliver, and Widlund, Olof B., "Improving the Parallel Performance of Overlapping Schwarz Methods by Using a Smaller Energy Minimizing Coarse Space," in *Domain Decomposition Methods in Science and Engineering XXIV*, ser. Lect. Notes Comput. Sci. Eng. DD 2017, vol. 125, Springer, 2018, pp. 383–392, isbn: 978-3-319-93873-8 (cit. on p. 73).

Bibliography

[HL20a]   HEINLEIN, Alexander and LANSER, Martin, "Additive and Hybrid Nonlinear Two-Level Schwarz Methods and Energy Minimizing Coarse Spaces for Unstructured Grids," *SIAM J. Sci. Comput.*, vol. 42, no. 4, A2461–A2488, 2020. DOI: `10.1137/19M1276972` (cit. on p. 150).

[HL20b]   HEINLEIN, Alexander and LANSER, Martin, "Coarse Spaces for Nonlinear Schwarz Methods on Unstructured Grids," in *Domain Decomposition Methods in Science and Engineering XXV*, ser. Lect. Notes Comput. Sci. Eng. DD 2018, vol. 138, Springer, 2020, pp. 268–276, ISBN: 978-3-030-56750-7. DOI: `10.1007/978-3-030-56750-7_30` (cit. on pp. 10, 19, 150).

[HPR22]   HEINLEIN, Alexander, PEREGO, Mauro, and RAJAMANICKAM, Sivasankaran, "FROSch Preconditioners for Land Ice Simulations of Greenland and Antarctica," *SIAM J. Sci. Comput.*, vol. 44, no. 2, B339–B367, 2022. DOI: `10.1137/21M1395260` (cit. on pp. 19, 21, 28, 73).

[HRR22]   HEINLEIN, Alexander, RHEINBACH, Oliver, and RÖVER, Friederike, "Parallel Scalability of Three-Level FROSch Preconditioners to 220000 Cores Using the Theta Supercomputer," *SIAM J. Sci. Comput.*, S173–S198, 2022, Special Section: 2021 Copper Mountain Conference. DOI: `10.1137/21M1431205` (cit. on pp. 28, 73).

[HS22]   HEINLEIN, Alexander and SMETANA, Kathrin, *A Fully Algebraic and Robust Two-Level Schwarz Method Based on Optimal Local Approximation Spaces*, Jul. 2022. arXiv: `2207.05559v1 [math.NA]` (cit. on p. 223).

[HBH+05]   HEROUX, Michael A., BARTLETT, Roscoe A., HOWLE, Vicki E., HOEKSTRA, Robert J., HU, Jonathan J., KOLDA, Tamara G., LEHOUCQ, Richard B., LONG, Kevin R., PAWLOWSKI, Roger P., PHIPPS, Eric T., SALINGER, Andrew G., THORNQUIST, Heidi K., TUMINARO, Ray S., WILLENBRING, James M., WILLIAMS, Alan, and STANLEY, Kendall S., "An Overview of the Trilinos Project," *ACM Trans. Math. Softw.*, vol. 31, no. 3, pp. 397–423,

2005 (cit. on p. 28).

[HK14]   HETMANIUK, Ulrich L. and KLAWONN, Axel, "Error Estimates for a Two-Dimensional Special Finite Element Method Based on Component Mode Synthesis," *Electron. Trans. Numer. Anal.*, vol. 41, pp. 109–132, 2014 (cit. on p. 145).

[HL10]   HETMANIUK, Ulrich L. and LEHOUCQ, Richard B., "A Special Finite Element Method Based on Component Mode Synthesis," *ESAIM. M2AN*, vol. 44, no. 3, pp. 401–420, 2010. DOI: `10.1051/m2an/2010007` (cit. on pp. 8, 11, 145, 146).

[Hoc20]   HOCHMUTH, Christian, "Parallel Overlapping Schwarz Preconditioners for Incompressible Fluid Flow and Fluid-Structure Interaction Problems," dissertation, University of Cologne, Cologne, Germany, Jun. 2020. URN: `urn:nbn:de:hbz:38-113450` (cit. on pp. 19, 73).

[HW97]   HOU, Thomas Y. and WU, Xiao-Hui, "A Multiscale Finite Element Method for Elliptic Problems in Composite Materials and Porous Media," *J. Comput. Phys.*, vol. 134, no. 1, pp. 169–189, 1997. DOI: `10.1006/jcph.1997.5682` (cit. on pp. 8, 10, 146).

[HWC99]   HOU, Thomas Y., WU, Xiao-Hui, and CAI, Zhiqiang, "Convergence of a Multiscale Finite Element Method for Elliptic Problems with Rapidly Oscillating Coefficients," *Math. Comp.*, vol. 68, no. 227, pp. 913–943, 1999. DOI: `10.1090/S0025-5718-99-01077-7` (cit. on pp. 8, 10, 146).

[Hur60]   HURTY, Walter C., "Vibrations of Structural Systems by Component Mode Synthesis," *Journal of Engineering Mechanics Division*, vol. 86, no. 4, pp. 51–69, 1960. DOI: `10.1061/JMCEA3.0000162` (cit. on p. 146).

[Hur65]   HURTY, Walter C., "Dynamic Analysis of Structural Systems Using Component Modes," *AIAA Journal*, vol. 3, no. 4, pp. 678–685, 1965. DOI: `10.2514/3.2947` (cit. on p. 146).

[JRZ21]   JOLIVET, Pierre, ROMAN, Jose E., and ZAMPINI, Stefano, "KSPHPDDM and PCHPDDM: Extending PETSc with Advanced Krylov Methods and Robust Multilevel Overlapping Schwarz Preconditioners," *Comput. Math. Appl.*, vol. 84, pp. 277–295, 2021. DOI: `10.1016/j.camwa.2021.01.003` (cit. on pp. 10, 223).

[KK98]   KARYPIS, George and KUMAR, Vipin, "A Fast and High Quality Multilevel Scheme for Partitioning Irregular Graphs," *SIAM J. Sci. Comput.*, vol. 20, no. 1, pp. 359–392, 1998. DOI: `10.1137/S1064827595287997` (cit. on pp. 27, 39, 89).

[KCW17]   KIM, Hyea Hyun, CHUNG, Eric, and WANG, Junxian, "BDDC and FETI-DP Preconditioners with Adaptive Coarse Spaces for Three-Dimensional Elliptic Problems with Oscillatory and High Contrast Coefficients," *J. Comput. Phys.*, vol. 349, pp. 191–214, 2017. DOI: `10.1016/j.jcp.2017.08.003` (cit. on p. 9).

[KKR16]   KLAWONN, Axel, KÜHN, Martin J., and RHEINBACH, Oliver, "Adaptive Coarse Spaces for FETI-DP in Three Dimensions," *SIAM J. Sci. Comput.*, vol. 38, no. 5, A2880–A2911, 2016. DOI: `10.1137/15M1049610` (cit. on pp. 9, 213).

[KKR17]   KLAWONN, Axel, KÜHN, Martin J., and RHEINBACH, Oliver, "Adaptive Coarse Spaces for FETI-DP in Three Dimensions with Applications to Heterogeneous Diffusion Problems," in *Domain Decomposition Methods in Science and Engineering XXIII*, ser. Lect. Notes Comput. Sci. Eng. DD 2015, vol. 116, Springer, 2017, pp. 187–196, ISBN: 978-3-319-52389-7. DOI: `10.1007/978-3-319-52389-7_18` (cit. on p. 214).

[KKR18a] KLAWONN, Axel, KÜHN, Martin J., and RHEINBACH, Oliver, "Adaptive FETI-DP and BDDC Methods with a Generalized Transformation of Basis for Heterogeneous Problems," *Electron. Trans. Numer. Anal.*, pp. 1–27, 2018. DOI: `10.1553/etna_vol49s1` (cit. on p. 214).

[KKR18b]   Klawonn, Axel, Kühn, Martin J., and Rheinbach, Oliver, "Adaptive FETI-DP and BDDC Methods with a Generalized Transformation of Basis for Heterogeneous Problems," *Electron. Trans. Numer. Anal.*, vol. 49, pp. 1–27, 2018. doi: `10.1553/etna_vol49s1` (cit. on p. 9).

[KKR20]   Klawonn, Axel, Kühn, Martin J., and Rheinbach, Oliver, "Parallel Adaptive FETI-DP Using Lightweight Asynchronous Dynamic Load Balancing," *Int. J. Numer. Methods Eng.*, vol. 121, no. 4, pp. 621–643, 2020. doi: `10.1002/nme.6237` (cit. on p. 214).

[KLRW19]   Klawonn, Axel, Lanser, Martin, Rheinbach, Oliver, and Weber, Janine, "Preconditioning the Coarse Problem of BDDC methods—Three-Level, Algebraic Multigrid, and Vertex-Based Preconditioners," *Electron. Trans. Numer. Anal.*, vol. 51, pp. 432–450, 2019. doi: `10.1553/etna_vol51s432` (cit. on p. 223).

[KLU+20]   Klawonn, Axel, Lanser, Martin, Uran, Matthias, Rheinbach, Oliver, Köhler, Stephan, Schröder, Jörg, Scheunemann, Lisa, Brands, Dominik, Balzani, Daniel, Gandhi, Ashutosh, Wellein, Gerhard, Wittmann, Markus, Schenk, Olaf, and Janalík, Radim, "EXASTEEL: Towards a Virtual Laboratory for the Multiscale Simulation of Dual-Phase Steel Using High-Performance Computing," in *Software for Exascale Computing: SPPEXA 2016–2019*, ser. Lect. Notes Comput. Sci. Eng. Vol. 136, Springer, 2020, pp. 351–404, isbn: 978-3-030-47956-5. doi: `10.1007/978-3-030-47956-5_13` (cit. on p. 8).

[KLW22]   Klawonn, Axel, Lanser, Martin, and Wasiak, Adam, "Adaptive and Frugal FETI-DP for Virtual Elements," *Vietnam J. Math.*, 2022. doi: `10.1007/s10013-022-00580-5` (cit. on p. 215).

[KRR15]   Klawonn, Axel, Radtke, Patrick, and Rheinbach, Oliver, "FETI-DP Methods with an Adaptive Coarse Space," *SIAM J. Numer. Anal.*, vol. 53, no. 1, pp. 297–320, 2015. doi: `10.1137/130939675` (cit. on p. 214).

[KRR16]   KLAWONN, Axel, RADTKE, Patrick, and RHEINBACH, Oliver, "A Comparison of Adaptive Coarse Spaces for Iterative Substructuring in Two Dimensions," *Electron. Trans. Numer. Anal.*, vol. 45, pp. 75–106, 2016 (cit. on pp. 9, 55).

[KR06]   KLAWONN, Axel and RHEINBACH, Oliver, "A Parallel Implementation of Dual-Primal FETI Methods for Three-Dimensional Linear Elasticity Using a Transformation of Basis," *SIAM J. Sci. Comput.*, vol. 28, no. 5, pp. 1886–1906, 2006. DOI: `10.1137/050624364` (cit. on p. 89).

[KW06]   KLAWONN, Axel and WIDLUND, Olof B., "Dual-Primal FETI Methods for Linear Elasticity," *Commun. Pure Appl. Math.*, vol. 59, no. 11, pp. 1523–1572, 2006. DOI: `10.1002/cpa.20156` (cit. on p. 89).

[Kne16]   KNEPPER, Jascha, "Multiskalen-Grobitterräume für überlappende Schwarz-Gebietszerlegungsverfahren," [Multiscale Coarse Spaces for Overlapping Schwarz Domain Decomposition Methods], German, M.S. thesis, University of Cologne, Cologne, Germany, Jul. 2016 (cit. on pp. 214, 215).

[Kny01]   KNYAZEV, Andrew V., "Toward the Optimal Preconditioned Eigensolver: Locally Optimal Block Preconditioned Conjugate Gradient Method," *SIAM J. Sci. Comput.*, vol. 23, no. 2, pp. 517–541, 2001. DOI: `10.1137/S1064827500366124` (cit. on pp. 210, 213).

[KC16]   KONG, Fande and CAI, Xiao-Chuan, "A Highly Scalable Multilevel Schwarz Method with Boundary Geometry Preserving Coarse Spaces for 3D Elasticity Problems on Domains with Complex Geometry," *SIAM J. Sci. Comput.*, vol. 38, no. 2, pp. C73–C95, 2016. DOI: `10.1137/15M1010567` (cit. on pp. 19, 223).

[KBB01]   KOUZNETSOVA, Varvara, BREKELMANS, W. A. M., and BAAIJENS, Frank P. T., "An Approach to Micro-Macro Modeling of Heterogeneous Materials," *Computational Mechanics*, vol. 27, no. 1, pp. 37–48, 2001. DOI: `10.1007/s004660000212` (cit. on p. 8).

[Küh18]   Kühn, Martin J., "Adaptive FETI-DP and BDDC Methods for Highly Heterogeneous Elliptic Finite Element Problems in Three Dimensions," dissertation, University of Cologne, Cologne, Germany, Feb. 2018. URN: `urn:nbn:de:hbz:38-82344` (cit. on pp. 9, 138).

[LB13]   Larson, Mats G. and Bengzon, Fredrik, *The Finite Element Method: Theory, Implementation, and Applications*, ser. Texts in Computational Science and Engineering. Springer, 2013, vol. 10, isbn: 978-3-642-33286-9. doi: `10.1007/978-3-642-33287-6` (cit. on p. 7).

[LCYC19]   Liao, Zi-Ju, Chen, Rongliang, Yan, Zhengzheng, and Cai, Xiao-Chuan, "A Parallel Implicit Domain Decomposition Algorithm for the Large Eddy Simulation of Incompressible Turbulent Flows on 3D Unstructured Meshes," *Int. J. Numer. Methods Fluids*, vol. 89, no. 9, pp. 343–361, 2019. doi: `10.1002/fld.4695` (cit. on p. 19).

[MS07]   Mandel, Jan and Sousedík, Bedřich, "Adaptive Selection of Face Coarse Degrees of Freedom in the BDDC and the FETI-DP Iterative Substructuring Methods," *Comput. Methods Appl. Mech. Engrg.*, vol. 196, no. 8, pp. 1389–1399, 2007. doi: `10.1016/j.cma.2006.03.010` (cit. on p. 9).

[MSŠ12]   Mandel, Jan, Sousedík, Bedřich, and Šístek, Jakub, "Adaptive BDDC in Three Dimensions," *Math. Comput. Simul.*, vol. 82, no. 10, pp. 1812–1831, 2012. doi: `10.1016/j.matcom.2011.03.014` (cit. on p. 9).

[MR18]   Marcinkowski, Leszek and Rahman, Talal, "Additive Average Schwarz with Adaptive Coarse Spaces: Scalable Algorithms for Multiscale Problems," *Electron. Trans. Numer. Anal.*, vol. 49, pp. 28–40, 2018. doi: `10.1553/etna_vol49s28` (cit. on p. 9).

[Nak19]   Nakajima, Yukio, *Advanced Tire Mechanics.* Springer, 2019, isbn: 978-981-13-5798-5. doi: `10.1007/978-981-13-5799-2` (cit. on p. 7).

[NT21]   Nataf, Frédéric and Tournier, Pierre-Henri, *A GenEO Domain Decomposition Method for Saddle Point Problems*, Dec. 2021. arXiv: `1911.01858v5`

`[cs.DC]` (cit. on p. 224).

[NXD10]   NATAF, Frédéric, XIANG, Hua, and DOLEAN, Victorita, "A Two Level Domain Decomposition Preconditioner Based on Local Dirichlet-to-Neumann Maps," *Compt. R. Math.*, vol. 348, no. 21, pp. 1163–1167, 2010. DOI: `10.1016/j.crma.2010.10.007` (cit. on p. 10).

[NXDS11]   NATAF, Frédéric, XIANG, Hua, DOLEAN, Victorita, and SPILLANE, Nicole, "A Coarse Space Construction Based on Local Dirichlet-to-Neumann Maps," *SIAM J. Sci. Comput.*, vol. 33, no. 4, pp. 1623–1642, 2011. DOI: `10.1137/100796376` (cit. on p. 10).

[Pan00]   PAN, C.-T., "On the Existence and Computation of Rank-Revealing LU Factorizations," *Linear Algebra Appl.*, vol. 316, no. 1, pp. 199–222, 2000. DOI: `10.1016/S0024-3795(00)00120-8` (cit. on p. 81).

[PD13]   PECHSTEIN, Clemens and DOHRMANN, Clark R., *Modern Domain Decomposition Solvers: BDDC, Deluxe Scaling, and an Algebraic Approach*, Slides to a talk by C. Pechstein at NuMa Seminar, JKU Linz, Dec. 2013. Available at: `https://people.ricam.oeaw.ac.at/c.pechstein/pechstein-bddc2013.pdf` (cit. on p. 9).

[PD17]   PECHSTEIN, Clemens and DOHRMANN, Clark R., "A Unified Framework for Adaptive BDDC," *Electron. Trans. Numer. Anal.*, vol. 46, pp. 273–336, 2017 (cit. on p. 9).

[QV08]   QUARTERONI, Alfio and VALLI, Alberto, *Numerical Approximation of Partial Differential Equations*, ser. Springer Ser. Comput. Math. 23. Springer, 2008, reprint, ISBN: 978-3-540-85267-4. DOI: `10.1007/978-3-540-85268-1` (cit. on p. 18).

[Saa03]   SAAD, Yousef, *Iterative Methods for Sparse Linear Systems*, 2nd ed. SIAM, 2003, ISBN: 978-0-898715-34-7. DOI: `10.1137/1.9780898718003` (cit. on pp. 7, 14, 23, 64, 83, 165, 168, 197).

[SVZ11] SCHEICHL, Robert, VASSILEVSKI, Panayot S., and ZIKATANOV, Ludmil T., "Weak Approximation Properties of Elliptic Projections with Functional Constraints," *Multiscale Model. Simul.*, vol. 9, no. 4, pp. 1677–1699, 2011. DOI: `10.1137/110821639` (cit. on pp. 9, 10).

[Sch14] SCHRÖDER, Jörg, "A Numerical Two-Scale Homogenization Scheme: The FE^2-Method," in *Plasticity and Beyond: Microstructures, Crystal-Plasticity and Phase Transitions*, ser. CISM. Springer, 2014, vol. 550, pp. 1–64, ISBN: 978-3-7091-1625-8. DOI: `10.1007/978-3-7091-1625-8_1` (cit. on p. 8).

[SHC15] SHIU, Wen-Shin, HWANG, Feng-Nan, and CAI, Xiao-Chuan, "Parallel Domain Decomposition Method for Finite Element Approximation of 3D Steady State Non-Newtonian Fluids," *Int. J. Numer. Methods Fluids*, vol. 78, no. 8, pp. 502–520, 2015. DOI: `10.1002/fld.4027` (cit. on p. 19).

[SBG96] SMITH, Barry F., BJØRSTAD, Petter E., and GROPP, William D., *Domain Decomposition: Parallel Multilevel Methods for Elliptic Partial Differential Equations*. Cambridge University Press, 1996, ISBN: 0-521-49589-X (cit. on pp. 8, 19, 21, 23, 25, 107).

[Sou10] SOUSEDÍK, Bedřich, "Adaptive-Multilevel BDDC," dissertation, University of Colorado Denver, Denver, Colorado, 2010 (cit. on pp. 210, 213).

[SŠM13] SOUSEDÍK, Bedřich, ŠÍSTEK, Jakub, and MANDEL, Jan, "Adaptive-Multilevel BDDC and Its Parallel Implementation," *Computing*, vol. 95, no. 12, pp. 1087–1119, 2013. DOI: `10.1007/s00607-013-0293-5` (cit. on p. 223).

[Spi21] SPILLANE, Nicole, *Toward a New Fully Algebraic Preconditioner for Symmetric Positive Definite Problems*, Jun. 2021. arXiv: `2106.11574v1` `[math.NA]` (cit. on p. 223).

[SDH+11] SPILLANE, Nicole, DOLEAN, Victorita, HAURET, Patrice, NATAF, Frédéric, PECHSTEIN, Clemens, and SCHEICHL, Robert, "A Robust Two-Level Do-

main Decomposition Preconditioner for Systems of PDEs," *Compt. R. Math.*, vol. 349, no. 23, pp. 1255–1259, 2011. DOI: `10.1016/j.crma.2011.10.021` (cit. on p. 10).

[SDH+14a] SPILLANE, Nicole, DOLEAN, Victorita, HAURET, Patrice, NATAF, Frédéric, PECHSTEIN, Clemens, and SCHEICHL, Robert, "Abstract Robust Coarse Spaces for Systems of PDEs via Generalized Eigenproblems in the Overlaps," *Numer. Math.*, vol. 126, no. 4, pp. 741–770, 2014. DOI: `10.1007/s00211-013-0576-y` (cit. on pp. 5, 9, 10, 12, 13, 47, 123, 124, 197, 208).

[SDH+14b] SPILLANE, Nicole, DOLEAN, Victorita, HAURET, Patrice, NATAF, Frédéric, PECHSTEIN, Clemens, and SCHEICHL, Robert, "Achieving Robustness through Coarse Space Enrichment in the Two Level Schwarz Framework," in *Domain Decomposition Methods in Science and Engineering XXI*, ser. Lect. Notes Comput. Sci. Eng. DD 2012, vol. 98, Springer, 2014, pp. 447–455, ISBN: 978-3-319-05789-7. DOI: `10.1007/978-3-319-05789-7_42` (cit. on p. 12).

[SR13] SPILLANE, Nicole and RIXEN, Daniel J., "Automatic Spectral Coarse Spaces for Robust Finite Element Tearing and Interconnecting and Balanced Domain Decomposition Algorithms," *Int. J. Numer. Methods Eng.*, vol. 95, pp. 953–990, 2013. DOI: `10.1002/nme.4534` (cit. on p. 9).

[SGB+10] STADLER, Georg, GURNIS, Michael, BURSTEDDE, Carsten, WILCOX, Lucas C., ALISIC, Laura, and GHATTAS, Omar, "The Dynamics of Plate Tectonics and Mantle Flow: From Local to Global Scales," *Science*, vol. 329, no. 5995, pp. 1033–1038, 2010. DOI: `10.1126/science.1191223` (cit. on pp. 7, 8).

[TDY+15] TASAN, Cemal Cem, DIEHL, Martin, YAN, Dingshun, BECHTOLD, Marion, ROTERS, Franz, SCHEMMANN, Lars, ZHENG, Chengwu, PERANIO, Nicola, PONGE, Dirk, KOYAMA, Motomichi, TSUZAKI, Kaneaki, and RAABE, Dierk, "An Overview of Dual-Phase Steels: Advances in Microstructure-Oriented Processing and Micromechanically Guided Design," *Annu. Rev. Mater.*

Res., vol. 45, pp. 391–431, 2015. DOI: `10.1146/annurev-matsci-070214-021103` (cit. on p. 7).

[Tho99]   THOMPSON, Suzanne C., "Illusions of Control: How We Overestimate Our Personal Influence," *Current Directions in Psychological Science*, vol. 8, no. 6, pp. 187–190, 1999. DOI: `10.1111/1467-8721.00044` (cit. on p. v).

[TW05]   TOSELLI, Andrea and WIDLUND, Olof B., *Domain Decomposition Methods — Algorithms and Theory*, ser. Springer Ser. Comput. Math. Springer, 2005, vol. 34, ISBN: 978-3-540-20696-5. DOI: `10.1007/b137868` (cit. on pp. 8, 18, 19, 21, 23–25, 28, 32, 105–107, 121, 131, 226).

[TOS00]   TROTTENBERG, Ulrich, OOSTERLEE, Cornelis W., and SCHÜLLER, Anton, *Multigrid.* Academic Press, 2000, ISBN: 0-12-701070-X (cit. on p. 8).

[Ura20]   URAN, Matthias, "High-Performance Computing Two-Scale Finite Element Simulations of a Contact Problem Using Computational Homogenization: Virtual Forming Limit Curves for Dual-Phase Steel," dissertation, University of Cologne, Cologne, Germany, 2020. URN: `urn:nbn:de:hbz:38-114016` (cit. on p. 8).

[VSG09]   VAN LENT, Jan, SCHEICHL, Robert, and GRAHAM, Ivan G., "Energy-Minimizing Coarse Spaces for Two-Level Schwarz Methods for Multiscale PDEs," *Numer. Linear Algebra Appl.*, vol. 16, no. 10, pp. 775–799, 2009. DOI: `10.1002/nla.641` (cit. on p. 24).

[VSM99]   VUIK, C., SEGAL, A., and MEIJERINK, J. A., "An Efficient Preconditioned CG Method for the Solution of a Class of Layered Problems with Extreme Contrasts in the Coefficients," *J. Comput. Phys.*, vol. 152, no. 1, pp. 385–403, 1999. DOI: `doi.org/10.1006/jcph.1999.6255` (cit. on p. 31).

[Wil13]   WILLEMS, Jörg, "Robust Multilevel Solvers for High-Contrast Anisotropic Multiscale Problems," *J. Comput. Appl. Math.*, vol. 251, pp. 47–60, 2013. DOI: `10.1016/j.cam.2013.03.030` (cit. on p. 223).

Bibliography

[Wil14]   Willems, Jörg, "Robust Multilevel Methods for General Symmetric Posi-
           tive Definite Operators," *SIAM J. Numer. Anal.*, vol. 52, no. 1, pp. 103–124,
           2014. DOI: `10.1137/120865872` (cit. on p. 223).

[WC14]   Wu, Yuqi and Cai, Xiao-Chuan, "A Fully Implicit Domain Decomposition
           Based ALE Framework for Three-Dimensional Fluid-Structure Interaction
           with Application in Blood Flow Computation," *J. Comput. Phys.*, vol. 258,
           pp. 524–537, 2014. DOI: `10.1016/j.jcp.2013.10.046` (cit. on p. 19).

[XZ17]   Xu, Jinchao and Zikatanov, Ludmil, "Algebraic Multigrid Methods," *Acta
           Numerica*, vol. 26, pp. 591–721, 2017. DOI: `10.1017/S0962492917000083`
           (cit. on p. 8).

[YDS21]   Yu, Yi, Dryja, Maksymilian, and Sarkis, Marcus, "From Additive Av-
           erage Schwarz Methods to Nonoverlapping Spectral Additive Schwarz
           Methods," *SIAM J. Numer. Anal.*, vol. 59, no. 5, pp. 2608–2638, 2021. DOI:
           `10.1137/20M1386554` (cit. on p. 9).

[Zam16]   Zampini, Stefano, "PCBDDC: A Class of Robust Dual-Primal Methods
           in PETSc," *SIAM J. Sci. Comput.*, vol. 38, no. 5, S282–S306, 2016. DOI:
           `10.1137/15M1025785` (cit. on p. 9).

[ZT17]   Zampini, Stefano and Tu, Xuemin, "Multilevel Balancing Domain Decom-
           position by Constraints Deluxe Algorithms with Adaptive Coarse Spaces
           for Flow in Porous Media," *SIAM J. Sci. Comput.*, vol. 39, no. 4, A1389–
           A1415, 2017. DOI: `10.1137/16M1080653` (cit. on p. 223).

Erklärung zur Dissertation

Ich versichere, dass ich die von mir vorgelegte Dissertation selbständig angefertigt, die benutzten Quellen und Hilfsmittel vollständig angegeben und die Stellen der Arbeit – einschließlich Tabellen, Karten und Abbildungen –, die anderen Werken im Wortlaut oder dem Sinn nach entnommen sind, in jedem Einzelfall als Entlehnung kenntlich gemacht habe; dass diese Dissertation noch keiner anderen Fakultät oder Universität zur Prüfung vorgelegen hat; dass sie – abgesehen von unten angegebenen Teilpublikationen – noch nicht veröffentlicht worden ist, sowie, dass ich eine solche Veröffentlichung vor Abschluss des Promotionsverfahrens nicht vornehmen werde. Die Bestimmungen der Promotionsordnung sind mir bekannt. Die von mir vorgelegte Dissertation ist von Herrn Prof. Dr. Axel Klawonn betreut worden.

Teilpublikationen

- Alexander Heinlein, Axel Klawonn, Jascha Knepper, and Oliver Rheinbach, *Multiscale Coarse Spaces for Overlapping Schwarz Methods Based on the ACMS Space in 2D*, Electron. Trans. Numer. Anal., vol. 48, pp. 156–182, 2018. doi: `10.1553/etna_vol48s156`

- Alexander Heinlein, Axel Klawonn, Jascha Knepper, and Oliver Rheinbach, *An Adaptive GDSW Coarse Space for Two-Level Overlapping Schwarz Methods in Two Dimensions*, in Domain Decomposition Methods in Science and Engineering XXIV, ser. Lect. Notes Comput. Sci. Eng. DD 2017, vol. 125, Springer, 2018, pp. 373–382, isbn: 978-3-319-93873-8, doi: `10.1007/978-3-319-93873-8_35`

- Alexander Heinlein, Axel Klawonn, Jascha Knepper, and Oliver Rheinbach, *Adaptive GDSW Coarse Spaces for Overlapping Schwarz Methods in Three Dimensions*, SIAM J. Sci. Comput., vol. 41, no. 5, A3045–A3072, 2019, doi: `10.1137/18M1220613`

- Alexander Heinlein, Axel Klawonn, Jascha Knepper, Oliver Rheinbach, and Olof

B. Widlund, *Adaptive GDSW Coarse Spaces of Reduced Dimension for Overlapping Schwarz Methods*, SIAM J. Sci. Comput., vol. 44, no. 3, A1176–A1204, 2022. doi: `10.1137/20M1364540`

Köln, 31. Oktober 2022   Jascha Knepper