A Three-level Extension of the GDSW Overlapping Schwarz Preconditioner in Three Dimensions

Alexander Heinlein, Axel Klawonn, Oliver Rheinbach, and Friederike Röver

1 The Standard GDSW Preconditioner

The GDSW (Generalized Dryja–Smith–Widlund) preconditioner is a two-level overlapping Schwarz domain decomposition preconditioner [23] with exact local solvers [5, 4]. The GDSW preconditioner can be written in the form

$$M_{\mathrm{GDSW}}^{-1} = \underbrace{\Phi K_0^{-1} \Phi^T}_{\text{Coarse Level}} + \underbrace{\sum_{i=1}^{N} R_i^T K_i^{-1} R_i}_{\text{First Level}}, \tag{1}$$

where $K_0 = \Phi^T K \Phi$ is the coarse matrix and the $K_i = R_i K R_i^T$, $i = 1, ..., N$, correspond to the local overlapping subdomain problems. By $V_1, \ldots, V_N$, we denote the local subspaces corresponding to the overlapping subdomains, and V_0 denotes the corresponding coarse space. The restriction operators on the subdomain level are defined as $R_i : V^h(\Omega) \to V_i := V^h(\Omega_i')$ for $i = 1, \ldots, N$. The columns of the matrix Φ correspond to the coarse basis function which are chosen to to be discrete harmonic extensions from the interface of the nonoverlapping decomposition to the interior degrees of freedom. The interface values are restrictions of the elements of the null space of the operator to the edges, vertices, and faces. For linear elliptic problems, the condition number of the Schwarz operator is bounded by

Alexander Heinlein and Axel Klawonn
Department of Mathematics and Computer Science, University of Cologne, Weyertal 86-90, 50931 Köln, Germany. E-mail: {alexander.heinlein,axel.klawonn}@uni-koeln.de.
Center for Data and Simulation Science, University of Cologne, Germany, url: http://www.cds.uni-koeln.de

Oliver Rheinbach, Friederike Röver
Institut für Numerische Mathematik und Optimierung, Technische Universität Bergakademie Freiberg, Akademiestr. 6, 09599 Freiberg, Germany. E-mail: {oliver.rheinbach, friederike.roever}@math.tu-freiberg.de.
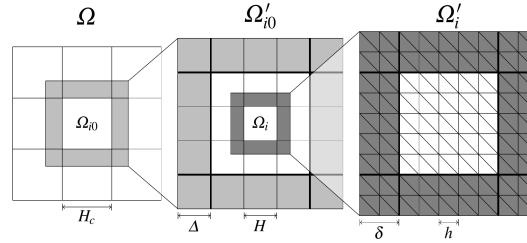
Fig. 1: Structured decomposition of an exemplary two-dimensional computational domain Ω into nonoverlapping subregions Ω_{i0}(left), a zoom into one overlapping subregion Ω'_{i0} consisting of subdomains Ω_i (middle), and a zoom into one overlapping subdomain Ω'_i (right). Each level of zoom corresponds to one level of the preconditioner; image taken from [13].

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

where h is the size of a finite element, H the size of a nonoverlapping subdomain, and δ the width of the overlap; see [4, 5, 6]. Even better condition number estimates are available. For example, the power of the logarithmic term can be reduced to 1 using Option 1 in [7]. An important advantage of the GDSW preconditioner is that it can be constructed in an algebraic fashion from the fully assembled matrix K and without the need of an additional coarse triangulation. This will also facilitate the construction of the three-level GDSW preconditioner presented in the following section.

2 The Three-Level GDSW Preconditioner

If a direct solver is used for the solution of the coarse problem in (1), this can become a bottleneck for a large number of subdomains; cf. [11, 9]. As a remedy, in this paper, we apply the GDSW preconditioner recursively to the coarse problem, resulting in a three-level extension of the GDSW preconditioner; see [13] for the corresponding algorithm in two dimensions. Our three-level GDSW method is related to the three-level BDDC method [24]. A further recursive application of the preconditioner, resulting in a multilevel extension similar to multi-level BDDC methods [18, 2, 16], multilevel Schwarz methods [17, 21], or multigrid methods [8], is algorithmically straightforward but out of the scope of this paper. The scalability of the two-level method can also be improved by reducing the size of the GDSW coarse space; cf. [14, 7]. There, instead of using coarse basis functions corresponding to subdomain edges, vertices, and, faces, new basis functions are constructed, e.g., corresponding only to the vertices. In this paper, we will construct three-level GDSW methods using standard as well as reduced dimension coarse spaces.

To define the three-level GDSW preconditioner, we decompose the domain Ω into nonoverlapping subregions Ω_{i0} of diameter H_c; see [24] and Figure 1 for a

graphical representation of the decomposition Ω in two dimensions. Each subregion is decomposed into nonoverlapping subdomains of diameter about H. Extending each subregion Ω_{i0} to Ω'_{i0} by recursively adding layers of subdomains, an overlapping decomposition into subregions is obtained. The overlap on subregion level is denoted by Δ; the overlap on the subdomain level is denoted by δ, consistent with the notation of the two-level method; see Figure 1.

The three-level GDSW preconditioner then is defined as

$$
M^{-1}_{3GDSW} = \Phi\bigg(\overbrace{\Phi_0 K^{-1}_{00} \Phi^T_0}^{\text{Third Level}} + \overbrace{\sum_{i=1}^{N_0} R^T_{i0} K^{-1}_{i0} R_{i0}}^{\text{Second Level}}\bigg)\Phi^T + \underbrace{\sum_{j=1}^{N} R^T_j K^{-1}_j R_j}_{\text{First Level}}, \qquad (3)
$$

$$\underbrace{\phantom{\Phi\bigg(\Phi_0 K^{-1}_{00} \Phi^T_0 + \sum_{i=1}^{N_0} R^T_{i0} K^{-1}_{i0} R_{i0}\bigg)\Phi^T}}_{\text{Coarse Levels}}$$

where $K_{00} = \Phi^T_0 K_0 \Phi_0$ and $K_{i0} = R_{i0} K_0 R^T_{i0}$. On the subregion level, we define the restriction operators to the overlapping subregions Ω'_{i0} as $R_{i0} : V^0 \to V^0_i := V^0(\Omega'_{i0})$ for $i = 1, ..., N_0$. The respective coarse space is denoted as V_{00} and spanned by the coarse basis functions Φ_0.

3 Implementation and Software Libraries

The parallel three-level GDSW implementation discussed in this paper is based on [9, 11, 12] and uses the Trilinos Epetra linear algebra package. A recent Xpetra version (FROSch - Fast and Robust Overlapping Schwarz framework [10]) is now part of the Trilinos [15] package ShyLU [19].

To test our three-level GDSW implementation, we consider the Poisson problem on the unit cube $[0, 1]^3$ with homogenous Dirichlet boundary conditions on $\partial\Omega$. We use structured domain decompositions into subregions and subdomains; see Figure 1 for a representation of the two-dimensional case. Our model problem is discretized using piecewise linear finite elements. As a default Krylov method, we apply the GMRES method provided by the Trilinos package Belos [3]. Trilinos version 12.11 (Dev) is used; cf. [15].

All numerical experiments were carried out on the JUQUEEN supercomputer at JSC Julich. We use the IBM XL C/C++ compiler for Blue Gene V.12.1, and Trilinos is linked to the ESSL.

To solve the overlapping subdomain and subregion problems and the coarse problem, we always use MUMPS 4.10.0 [1] in symmetric, sequential mode, and interfaced through the Trilinos package Amesos [20]. For our experiments, we always have a one-to-one correspondence of subdomains and processor cores. We use the relative stopping criterion $\|r^k\|_2/\|r^0\|_2 \le 10^{-6}$. Moreover, we assume that we have a fast and scalable method to identify interface degrees of freedom. That cost is therefore neglected in this paper.

3.1 Weak Parallel Scalability of the Three-Level GDSW Preconditioner

In this section, we focus on the weak scalability of our preconditioners. For the numerical scalability of the three-level GDSW preconditioner in two dimensions, more detailed numerical results can be found in [13]. We also compare results for the two and three-level methods using the standard coarse space (denoted by GDSW) and the reduced dimension coarse space (denoted by RGDSW). In particular, we use *Option 1*, which is the completely algebraic variant of the RGDSW coarse space; cf. [7] or [14], respectively.

The number of Krylov iterations is presented in Figure 2 and Table 1. Note that the standard two-level GDSW method fails for more than 13 824 cores since the coarse problem could not be factored any more due to memory limits. All other methods show numerical scalability for up to 64 000 cores. This includes the two-level RGDSW method, which is a remarkable result since RGDSW coarse space is smaller than the standard GDSW coarse space but the coarse basis function have, on average, a larger support (see also Table 2); cf. also [14].

Our results show, that the numerical scalability of both two-level methods is slightly better; cf. Figure 2 and Table 1. Moreover, the number of iterations is higher by almost a factor of two for both three-level methods; this is, however, not surprising since the direct coarse solver is replaced by a (two-level) preconditioner.
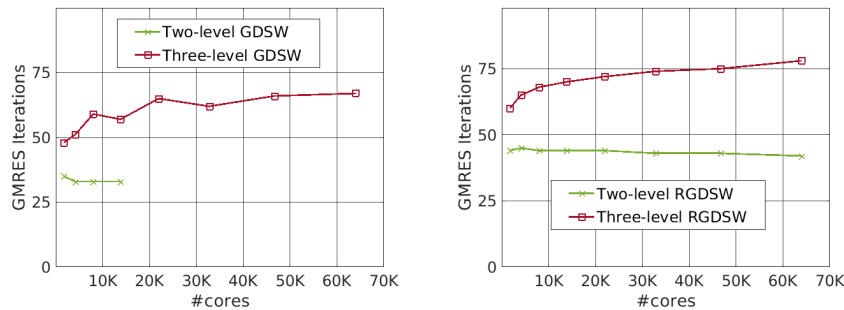

Fig. 2: Weak numerical scalability of the two- and three-level GDSW (left) and the RGDSW (right) preconditioner. All methods are numerically scalable; see Table 1 for the corresponding data.

Let us now consider the computing times, which are more favorable for the three-level methods; see Figure 3 and Table 1. By *Solver Time*, we denote the time to solution, which is the sum of the time for the setup of the preconditioner, denoted *Setup Time*, and the time for the Krylov iteration, which we denote *Krylov Time*. The *Setup Time* includes the factorizations of the matrices on the different levels using the MUMPS sparse direct solver. For RGDSW coarse space, the three-level method is faster than the two-level methods for 4 096 cores and more; see Figure 3 and Table 1. The two-level RGDSW method is consistently the fastest method from 1 728 to to 32 768 cores. However, for 46 656 and 64 000 cores, the three-level method is faster.
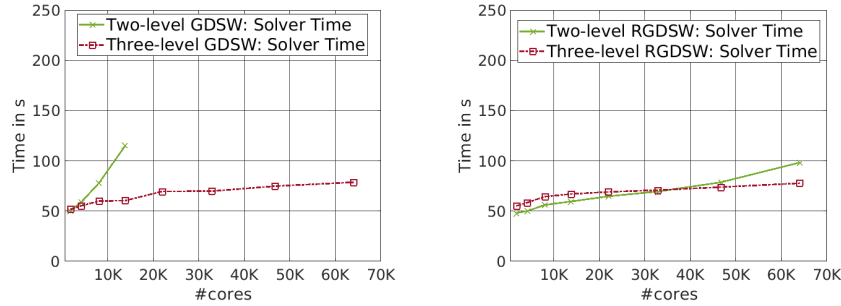
Fig. 3: Weak parallel scalability of the two- and three-level methods using the standard (left) and the reduced coarse space (right); see Table 1 for the data.
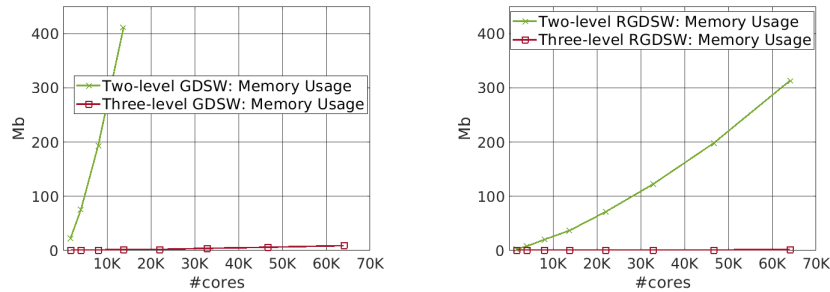
Fig. 4: Memory usage of the MUMPS direct solver for the factorization of the coarse matrix K_0 and K_{00} for the two-level and three-level GDSW method using the standard (left) and reduced coarse space (right); see Table 2 for the corresponding data.

For the largest problem with 1.72 billion degrees of freedom, the *Solver Time* for three-level RGDSW precondtioner (77.7s *Solver Time*) more than 20% faster than two-level RGDSW preconditioner (98.3s *Solver Time*) and also slightly faster than the three-level GDSW preconditioner (78.7s *Solver Time*). However, considering the size of K_0, we expect the two-level RGDSW to fail beyond 100 000 cores while both three-level methods will continue to scale; also cf. the memory usage for the factorazation of K_{00} in Figure 4 and Table 2.

| #Sub-domains = #cores	Two-level GDSW				Three-level GDSW				Two-level RGDSW				Three-level RGDSW			
	Iter	Solver Time	Setup Time	Krylov Time	Iter	Solver Time	Setup Time	Krylov Time	Iter	Solver Time	Setup Time	Krylov Time	Iter	Solver Time	Setup Time	Krylov Time
1 728	35	50.2 s	30.9 s	19.4 s	48	51.8 s	28.3 s	23.4 s	44	**47.9 s**	26.9 s	21.1 s	60	55.2 s	26.2 s	28.9 s
4 096	33	58.7 s	35.5 s	23.2 s	51	55.1 s	30.1 s	25.0 s	45	**50.0 s**	27.6 s	22.4 s	65	58.3 s	26.7 s	31.6 s
8 000	33	77.7 s	46.3 s	31.4 s	59	60.0 s	30.2 s	29.8 s	44	**56.1 s**	32.3 s	23.8 s	68	64.4 s	30.8 s	33.7 s
13 824	33	115.2 s	69.1 s	46.0 s	57	60.4 s	31.3 s	29.1 s	44	**59.6 s**	33.3 s	26.3 s	70	67.0 s	31.9 s	35.1 s
21 952	—	—	—	—	65	69.5 s	35.0 s	34.6 s	44	**64.7 s**	34.6s	30.1 s	72	69.0 s	32.1 s	36.9 s
32 768	—	—	—	—	62	69.8 s	36.2 s	33.6 s	43	**69.4 s**	35.2 s	34.2 s	74	70.8 s	32.6 s	38.2 s
46 656	—	—	—	—	66	74.8 s	37.1 s	37.6 s	43	78.6 s	37.2 s	41.4 s	75	**73.8 s**	33.7 s	40.2 s
64 000	—	—	—	—	67	78.7 s	38.5 s	40.2 s	42	98.3 s	50.2 s	48.1 s	78	**77.7 s**	34.8 s	42.9 s

Table 1: By *Iter*, we denote number of Krylov iterations. The *Solver Time* is the sum of the *Setup Time* and *Krylov Time*. We have $H/h = 30$, $H/\delta = 15$, $H_c/H = 4$, and $H_c/\Delta = 4$. Also see Figure 2 and Figure 3. The fastest *Solver Time* is printed in bold.

References

1. Amestoy, P.R., Duff, I.S., L'Excellent, J.Y., Koster, J.: A fully asynchronous multifrontal solver using distributed dynamic scheduling. SIAM J. Matrix Anal. Appl. **23**(1), 15–41 (2001)
2. Badia, S., Martín, A.F., Principe, J.: Multilevel balancing domain decomposition at extreme scales. SIAM J. Sci. Comput. **38**(1), C22–C52 (2016). DOI:10.1137/15M1013511
3. Bavier, E., Hoemmen, M., Rajamanickam, S., Thornquist, H.: Amesos2 and Belos: Direct and iterative solvers for large sparse linear systems. Scientific Programming **20**(3), 241–255 (2012)
4. Dohrmann, C.R., Klawonn, A., Widlund, O.B.: Domain decomposition for less regular sub-domains: overlapping Schwarz in two dimensions. SIAM J. Numer. Anal. **46**(4), 2153–2168 (2008)
5. Dohrmann, C.R., Klawonn, A., Widlund, O.B.: A family of energy minimizing coarse spaces for overlapping Schwarz preconditioners. In: Domain decomposition methods in science and engineering XVII, *Lect. Notes Comput. Sci. Eng.*, vol. 60, pp. 247–254. Springer, Berlin (2008)
6. Dohrmann, C.R., Widlund, O.B.: Hybrid domain decomposition algorithms for compressible and almost incompressible elasticity. Internat. J. Numer. Methods Engrg. **82**(2), 157–183 (2010)

#Subdomains = #Cores	Size of K_0	Factori- zation Time	Forward- Backward	Memory Usage	Size of K_{00}	Factori- zation Time	Forward- Backward	Memory Usage
		Two-level GDSW				Three-level GDSW		
1 728	10 439	1.28 s	2.08 s	23 Mb	98	<0.01 s	0.03 s	1 Mb
4 096	25 695	4.43 s	5.17 s	76 Mb	279	0.01 s	0.09 s	1 Mb
8 000	51 319	11.25 s	11.31 s	193 Mb	604	0.02 s	0.21 s	1 Mb
13 824	89 999	29.58 s	20.46 s	412 Mb	1 115	0.04 s	0.35 s	2 Mb
21 952		—	—	—	1 854	0.09 s	0.68 s	2 Mb
32 768		—	—	—	2 863	0.15 s	0.99 s	4 Mb
46 656		—	—	—	4 184	0.25 s	1.55 s	6 Mb
64 000		—	—	—	5 589	0.40 s	2.28 s	9 Mb
		Two-level RGDSW				Three-level RGDSW		
1 728	1 331	0.06 s	0.3 s	3 Mb	8	<0.01 s	0.01 s	1 Mb
4 096	3 375	0.25 s	0.87 s	8 Mb	27	<0.01 s	0.02 s	1 Mb
8 000	6 859	0.74 s	1.73 s	20 Mb	64	<0.01 s	0.03 s	1 Mb
13 824	12 167	1.81 s	3.02 s	37 Mb	125	<0.01 s	0.05 s	1 Mb
21 952	19 683	3.66 s	5.31 s	71 Mb	216	<0.01 s	0.08 s	1 Mb
32 768	29 791	6.15 s	8.25 s	122 Mb	343	0.01 s	0.13 s	1 Mb
46 656	42 875	10.39 s	12.53 s	198 Mb	512	0.02 s	0.19 s	1 Mb
64 000	59 319	16.80 s	16.96 s	313 Mb	729	0.03 s	0.27 s	2 Mb

Table 2: Costs for solving the problem on the coarsest level, i.e., using K_0 in the standard two-level GDSW and RGDSW preconditioner and using K_{00} in the three-level GDSW and RGDSW preconditioner. Here, *Factorization Time* is the time Amesos reports for the MUMPS sparse direct solver for the sum of symbolic and numerical factorization of K_0 and K_{00}, respectively; *Forward-Backward* is the sum of all times spent in forward-backward substitutions during the Krylov iteration; *Memory Usage* is the estimated amount of memory allocated by MUMPS during the factorization. See Table 1 for the corresponding *Solver Time*, *Setup Time* and *Krylov Time*. Also see Figures 5, 4.
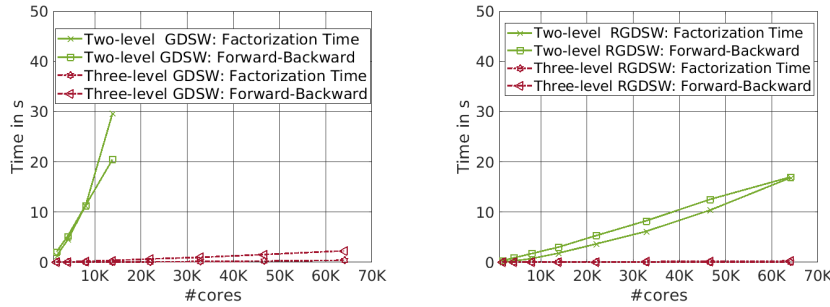
Fig. 5: Computing time for solving the problem on the coarsest level, i.e., using K_0 in the standard two-level method preconditioner ans using K_{00} for the three-level GDSW preconditioner and using the standard coarse space (left) and respectively the reduced coarse space (right). See Table 2 for the corresponding data.

7. Dohrmann, C.R., Widlund, O.B.: On the design of small coarse spaces for domain decomposition algorithms. SIAM J. Sci. Comput. **39**(4), A1466–A1488 (2017)
8. Hackbusch, W.: Multigrid methods and applications, *Springer Series in Computational Mathematics*, vol. 4. Springer-Verlag, Berlin (1985)
9. Heinlein, A.: Parallel overlapping Schwarz preconditioners and multiscale discretizations with applications to fluid-structure interaction and highly heterogeneous problems. Ph.D. thesis, Universität zu Köln (2016)
10. Heinlein, A., Klawonn, A., Rajamanickam, S., Rheinbach, O.: FROSch – a parallel implementation of the GDSW domain decomposition preconditioner in Trilinos (2020). In preparation

11. Heinlein, A., Klawonn, A., Rheinbach, O.: A parallel implementation of a two-level overlapping Schwarz method with energy-minimizing coarse space based on Trilinos. SIAM J. Sci. Comput. **38**(6), C713–C747 (2016)

12. Heinlein, A., Klawonn, A., Rheinbach, O.: Parallel Two-Level Overlapping Schwarz Methods in Fluid-Structure Interaction, pp. 521–530. LNCSE vol 112. Springer International Publishing, Cham (2016)

13. Heinlein, A., Klawonn, A., Rheinbach, O., Röver, F.: A Three-Level Extension of the GDSW Overlapping Schwarz Preconditioner in Two Dimensions, pp. 187–204. Springer International Publishing, Cham (2019). `DOI:10.1007/978-3-030-14244-5_10`. URL `https://doi.org/10.1007/978-3-030-14244-5_10`

14. Heinlein, A., Klawonn, A., Rheinbach, O., Widlund, O.B.: Improving the parallel performance of overlapping Schwarz methods by using a smaller energy minimizing coarse space. In: P.E. Bjørstad, S.C. Brenner, L. Halpern, H.H. Kim, R. Kornhuber, T. Rahman, O.B. Widlund (eds.) Domain Decomposition Methods in Science and Engineering XXIV, pp. 383–392. Springer International Publishing, Cham (2018)

15. Heroux, M.A., Bartlett, R.A., Howle, V.E., Hoekstra, R.J., Hu, J.J., Kolda, T.G., Lehoucq, R.B., Long, K.R., Pawlowski, R.P., Phipps, E.T., Salinger, A.G., Thornquist, H.K., Tuminaro, R.S., Willenbring, J.M., Williams, A., Stanley, K.S.: An overview of the Trilinos Project. ACM Trans. Math. Software **31**(3), 397–423 (2005)

16. J. Sístek B. Sousedík, J.M., Burda, P.: Parallel implementation of multilevel BDDC. In: Numerical mathematics and advanced applications 2011 (2011). 9th European conference on numerical mathematics and advanced applications, Leicester, UK, September 5–9, 2011

17. Kong, F., Cai, X.C.: 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. **38**(2), C73–C95 (2016). `DOI:10.1137/15M1010567`

18. Mandel, J., Sousedík, B., Dohrmann, C.R.: Multispace and multilevel BDDC. Computing **83**(2-3), 55–85 (2008)

19. Rajamanickam, S., Boman, E.G., Heroux, M.A.: Shylu: A hybrid-hybrid solver for multicore platforms. In: 2012 IEEE 26th International Parallel and Distributed Processing Symposium, pp. 631–643 (2012). `DOI:10.1109/IPDPS.2012.64`

20. Sala, M., Stanley, K., Heroux, M.: On the design of interfaces to sparse direct solvers. ACM Trans. Math. Software (TOMS) **34**(2) (2008)

21. Scacchi, S.: A hybrid multilevel Schwarz method for the bidomain model. Comput. Methods Appl. Mech. Engrg. **197**(45-48), 4051–4061 (2008). `DOI:10.1016/j.cma.2008.04.008`

22. Stephan, M., Docter, J.: JUQUEEN: IBM Blue Gene/Q® Supercomputer System at the Julich Supercomputing Centre. Journal of large-scale research facilities **1**, A1 (2015)

23. Toselli, A., Widlund, O.: Domain decomposition methods—algorithms and theory, *Springer Series in Computational Mathematics*, vol. 34. Springer-Verlag, Berlin (2005)

24. Tu, X.: Three-level BDDC in two dimensions. Internat. J. Numer. Meth. Engrg **69**(1), 33–59 (2007). `DOI:10.1002/nme.1753`