A Primal-Dual Box-Constrained QP Pressure Poisson Solver with Topology-Aware Geometry-Inspired Aggregation AMG supplementary material

1 Multigrid Scheme Evaluation

We evaluate various combinations of cycle, coarsening, and smoothing schemes within AMGCG to justify our MG design on two scenarios: cubic domain and maze.

1.1 Cubic Domain Test

We use a 3D cubic domain, where a source term is placed on the domain center while the outermost boundaries are fixed with a zero pressure Dirichlet boundary condition (similar to the example in AMGCL [1]). We compare the following schemes:

1. V+UA+WJ: V-cycle + unsmoothed aggregation + weighted Jacobi;
2. W+UA+WJ: W-cycle + UA + WJ;
3. V+UA+SPAI-0;
4. W+UA+SPAI-0;
5. V+SA+WJ: V + smoothed aggregation + WJ;
6. W+SA+WJ;
7. V+SA+SPAI-0;
8. W+SA+SPAI-0;
9. V+GIA+WJ;
10. W+GIA+WJ;
11. V+GIA+SPAI-0;

Table 1: Evaluation of cycle, coarsening, and smoothing schemes within AMGCG, at a resolution of 128^3 and 256^3. Numbers in the "CG" column are formatted as: preparation time + CG solve time (CG iteration count). Numbers in the "NNZ" column are formatted as: sum of non-zeros over the MG hierarchy in $\mathbf{R}$ and $\mathbf{P}$ + $\mathbf{A}$. Timing is shown in seconds. The best results in each column are highlighted in blue.

Scheme \ Res	128^3		256^3	
	CG	NNZ	CG	NNZ
V+UA+WJ	0.20+3.29 (69)	4.6+18.2	2.03+43.20 (100)	37.4+149.7
W+UA+WJ	0.21+1.80 (20)	4.6+18.2	1.80+17.95 (22)	37.4+149.7
V+UA+SPAI-0	0.22+3.15 (67)	4.6+18.2	1.80+41.65 (99)	37.4+149.7
W+UA+SPAI-0	0.22+1.78 (20)	4.6+18.2	1.70+17.05 (21)	37.4+149.7
V+SA+WJ	0.47+1.21 (21)	17.6+24.1	4.29+13.04 (25)	145.6+199.2
W+SA+WJ	0.46+1.50 (12)	17.6+24.1	3.86+14.12 (13)	145.6+199.2
V+SA+SPAI-0	0.46+1.16 (20)	17.6+24.1	3.90+12.30 (24)	145.6+199.2
W+SA+SPAI-0	0.46+1.47 (12)	17.6+24.1	3.85+13.02 (12)	145.6+199.2
V+GIA+WJ	0.18+1.94 (43)	4.6+16.0	1.55+21.43 (54)	37.6+131.0
W+GIA+WJ	0.17+1.38 (17)	4.6+16.0	1.39+12.17 (17)	37.6+131.0
V+GIA+SPAI-0	0.17+1.90 (42)	4.6+16.0	1.37+21.15 (53)	37.6+131.0
W+GIA+SPAI-0	0.17+1.39 (17)	4.6+16.0	1.36+12.12 (17)	37.6+131.0
V+GIA+RBSGS	0.18+1.27 (29)	4.6+16.0	1.37+12.85 (34)	37.6+131.0
W+GIA+RBSGS	0.17+0.71 (9)	4.6+16.0	1.38+6.13 (9)	37.6+131.0
V+GIA+RBSSOR	0.17+1.00 (23)	4.6+16.0	1.36+10.15 (27)	37.6+131.0
W+GIA+RBSSOR	0.17+0.70 (9)	4.6+16.0	1.38+6.12 (9)	37.6+131.0
V+RS+SPAI-0	1.96+1.90 (15)	21.8+40.3	26.60+20.24 (16)	179.6+336.8

12. W+GIA+SPAI-0;
13. V+GIA+RBSGS;
14. W+GIA+RBSGS;
15. V+GIA+RBSSOR;
16. W+GIA+RBSSOR (ours);
17. V+RS+SPAI-0: V + Ruge-Stüben coarsening + SPAI-0.

As both topology-oblivious and topology-aware GIA generate exactly the same MG hierarchy with this simple domain, we report the performance of topology-oblivious GIA. As UA, SA, and RS coarsening schemes break the regular grid structures for red-black coloring (and the superiority of WJ vs. SPAI-0 depends on contexts [2, 3, 4]), we experiment with WJ (with the optimal parameter $\omega = 6/7$ [5]) and/or SPAI-0 (with the optimal parameter $\omega = 1$ [2]). For W-cycle with WJ/SPAI-0, we use just one resmoothing as this was more efficient than performing two. In this example, we use a termination relative residual of 10^{-10}. While the Ruge-Stüben approach is too expensive in terms of computation and memory cost, we include it for reference. Table 1 summarizes the evaluation settings and results, and Figure 1 gives the log-scale plots of convergence over CG iterations for a selected set of the schemes.

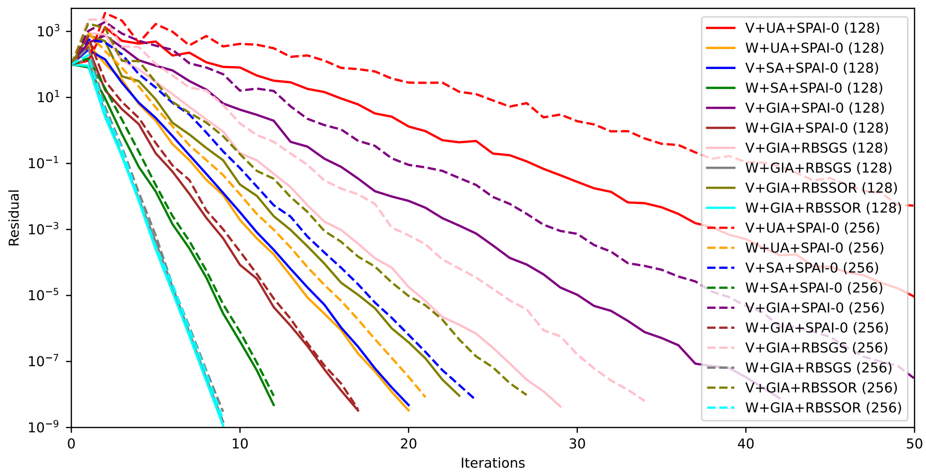

Figure 1: Log-scale profiles of convergence over CG iterations. For each scheme, (128) and (256) represent the used resolutions 128^3 and 256^3, and drawn with solid and dashed lines, respectively. Schemes with W-cycles give nearly overlapping solid and dashed lines, indicating its scalability.

Cycle: Using W-cycles significantly reduces the number of necessary CG iterations compared to V-cycles and scales (almost) linearly because W-cycles more accurately solve the preconditioning system by focusing more on the coarser levels. In particular, W-cycles improve efficiency with the unsmoothed aggregation schemes since the reduced iteration count outweighs their relatively small additional cost compared to V-cycles. With smoothed aggregation, while the reduction of the iteration count from W-cycles is not enough to justify their higher cost for smaller systems given the sufficiently effective V-cycles, W-cycles would become gradually beneficial for larger systems due to their scalability.

Coarsening: The hierarchy construction was relatively quick for both GIA and UA (while GIA was slightly faster due to the given geometric information) whereas SA is much more expensive due to the additional, non-negligible cost of smoothing the prolongation operator. In particular, if we use a looser termination residual or early termination, the setup cost can be more critical. Furthermore, GIA and UA generate much sparser prolongation/restriction operators and coarser-level systems, spending only half the memory compared to SA. These sparse operators and systems also contribute to efficient preconditioning. In addition, GIA generates a better hierarchy than UA, resulting in faster convergence even with the same WJ or SPAI-0 smoothers.

Smoothing: GS-type smoothers (enabled by GIA) significantly reduce the necessary CG iterations and improve efficiency, outperforming the WJ and SPAI-0. While using over-relaxation improves the convergence for V-cycles, it has only negligible effects for W-cycles (see W+GIA+RBSGS and W+GIA+RBSSOR in Table 1), although the termination relative residual (after 9 iterations for both approaches using 256^3) was slightly lower for W+GIA+RBSSOR than W+GIA+RBSGS (1.69×10^{-11} vs. 3.00×10^{-11}). We conjecture that the effectiveness of over-relaxation becomes smaller with a W-cycle because preconditioning effectiveness is almost saturated with the W-cycle by itself.

In summary, we find that our W+GIA+RBSSOR is most efficient in both computational cost and memory usage.

1.2 Maze Test

Next, we experiment with the topology-critical maze example without free surfaces (see the main paper) to evaluate the efficacy of our topology-aware coarsening. We use a grid resolution of 128^3 and 192^3 and first five frames for evaluation. We compare the following schemes:

1. V+SA+SPAI-0;
2. W+SA+SPAI-0;
3. V+GIA+SPAI-0;
4. W+GIA+SPAI-0;
5. V+GIA+RBSGS;
6. W+GIA+RBSGS;
7. V+GIA+RBSSOR;
8. W+GIA+RBSSOR;
9. V+TAGIA+WJ: V + topology-aware GIA + WJ;
10. W+TAGIA+WJ;
11. V+TAGIA+WJ(1): V + TAGIA + WJ with $\omega = 1$
12. W+TAGIA+WJ(1);
13. V+TAGIA+SPAI-0;
14. W+TAGIA+SPAI-0;
15. V+TAGIA+RBSGS;
16. W+TAGIA+RBSGS;
17. V+TAGIA+RBSSOR;
18. W+TAGIA+RBSSOR (ours);
19. V+TAGIA+RBSSOR+SPAI-0: V + TAGIA with RBSSOR at the finest and SPAI-0 at the coarser levels;

Table 2: Evaluation of cycle, coarsening, and smoothing schemes within AMGCG, at a resolution of 128^3 and 192^3. Numbers are formatted as: preparation time + CG solve time (CG iteration count). Timing is shown in seconds. The best results in each column are highlighted in blue.

Scheme \ Res	128^3	192^3
V+SA+SPAI-0	0.64+0.71 (19.4)	2.06+2.69 (21.8)
W+SA+SPAI-0	0.64+0.85 (11.0)	2.04+2.75 (11.0)
V+GIA+SPAI-0	0.40+4.27 (152.2)	1.34+17.00 (174.8)
W+GIA+SPAI-0	0.40+2.22 (43.6)	1.30+8.47 (47.4)
V+GIA+RBSGS	0.40+2.89 (107.4)	1.32+11.67 (125.2)
W+GIA+RBSGS	0.39+1.31 (27.0)	1.30+5.22 (30.6)
V+GIA+RBSSOR	0.40+2.60 (96.2)	1.31+10.95 (117.6)
W+GIA+RBSSOR	0.39+1.25 (25.6)	1.28+5.08 (30.2)
V+TAGIA+WJ	0.43+1.64 (58.4)	1.37+6.25 (63.8)
W+TAGIA+WJ	0.42+0.71 (13.8)	1.36+2.33 (13.0)
V+TAGIA+WJ(1)	0.42+17.20 (595.8)	1.37+66.88 (686.2)
W+TAGIA+WJ(1)	0.42+0.77 (15.2)	1.39+2.65 (14.8)
V+TAGIA+SPAI-0	0.42+1.80 (63.4)	1.37+6.85 (70.4)
W+TAGIA+SPAI-0	0.42+0.74 (14.4)	1.37+2.44 (13.8)
V+TAGIA+RBSGS	0.45+1.19 (37.6)	1.37+3.81 (40.8)
W+TAGIA+RBSGS	0.42+0.37 (7.2)	1.37+1.21 (7.0)
V+TAGIA+RBSSOR	0.41+0.89 (31.0)	1.37+3.24 (34.2)
W+TAGIA+RBSSOR	0.42+0.34 (6.8)	1.40+1.14 (6.6)
V+TAGIA+RBSSOR+SPAI-0	0.42+1.27 (46.6)	1.37+4.94 (52.4)
W+TAGIA+RBSSOR+SPAI-0	0.42+0.61 (12.2)	1.40+1.99 (11.2)

20. W+TAGIA+RBSSOR+SPAI-0;

In this example, we set the termination relative residual to 10^{-8}, maximum CG iteration count $1,000$, and optimal parameters for WJ and SPAI-0, unless otherwise mentioned. Table 2 summarizes the evaluation settings and results (averaged over five frames).

Cycle: Similar to the previous experiment, W-cycle is quite effective and improves the performance as both GIA and TAGIA are unsmoothed aggregation. In particular, while WJ(1) with the suboptimal parameter sometimes fails to converge with V-cycle, it quickly converges with W-cycle, reducing the iteration count from 686.2 to 14.8, which further demonstrates the effectiveness of W-cycles.

Coarsening: As our topology-aware approaches generate more effective coarser-level systems (similar to SA) with a very small extra cost, TAGIA improves the performance due to the reduced number of CG iterations, compared to topology-oblivious GIA. In particular, W+GIA+RBSSOR spends 5.08s with 30.2 iterations while our W+TAGIA+RBSSOR does 1.14s with 6.6 iterations only, achieving the performance gain of 4.5× for the CG solve. In addition, topology-aware approaches with W-cycle (W+SA and W+TAGIA) are scalable whereas W+GIA is not, as the required CG iterations increase by around 10-20% from the resolution 128^3 to 192^3.

Table 3: Performance evaluation with single-precision floating-point. M denotes the number of MPRGP iterations, C number of inner CG iterations on average, and N number of Newton iterations, and T total time in seconds.

Scheme	M	C	N	T
SAAMG-MPRGP	74.0			4.96
IPM-PD (SAAMG-CG)		6.7	6.0	4.84
IPM-PD (GIAAMG-CG)		3.8	4.0	1.83

Smoother: While the hybrid smoother, RBSSOR+SPAI-0, has an advantage that it can be applied to the system with red-black coloring available only at the finest level, our dedicated coarsening scheme enables red-black coloring from the finest to coarsest levels, allowing us to use RBSSOR entirely. Due to the GS-type smoothing, our RBSSOR is more efficient than RBSSOR+SPAI-0, proving that using GS-type smoothers not only at the finest but also the coarser levels can further accelerate convergence.

In summary, we find that our W+TAGIA+RBSSOR is most efficient and scalable in this topology-critical maze scenario.

2 IPM-PD with Single-Precision Floating-Point

To confirm the compatibility of our IPM-PD with single-precision floating-point, we perform a simple experiment using our method, along with SAAMG-MPRGP. We use a 3D cubic domain, where a source term is placed on the domain center while the outermost boundaries are fixed with a zero pressure Dirichlet boundary condition and an upper bound imposed on the solution. We use a grid resolution of 128^3, and compare the following schemes:

1. SAAMG-MPRGP: SAAMG-MPRGP with V-cycle and SPAI-0 [4];
2. IPM-PD (SAAMG-CG): IPM-PD with the inner linear solver, SAAMG-CG using V-cycle and SPAI-0;
3. IPM-PD (GIAAMG-CG): GIAAMG-CG using W-cycle and RBSSOR.

Due to the limited accuracy with single-precision floating-point, it is necessary to choose appropriate parameters for IPM-PD; we use barrier parameter $\mu = 10^{-8}$ and offset value $\epsilon = 10^{-6}$. In addition, as convergence of CG can be delayed with single-precision floating-point [6], it is necessary to use slightly larger numbers of inner CG iterations; we use 8 and 5 for SAAMG-CG and GIAAMG-CG, respectively. Table 3 summarizes the performance results. IPM-PD (GIAAMG-CG) is around 2.7× faster than SAAMG-MPRGP.

References

[1] D. Demidov. Amgcl: An efficient, flexible, and extensible algebraic multigrid implementation. *Lobachevskii Journal of Mathematics*, 40(5):535–546, May 2019.

[2] Oliver Bröker, Marcus J. Grote, Carsten Mayer, and Arnold Reusken. Robust parallel smoothing for multigrid via sparse approximate inverses. *SIAM Journal on Scientific Computing*, 23(4):1396–1417, 2001.

[3] Han Shao, Libo Huang, and Dominik L. Michels. A fast unsmoothed aggregation algebraic multigrid framework for the large-scale simulation of incompressible flow. *ACM Trans. Graph.*, 41(4), jul 2022.

[4] Tetsuya Takahashi and Christopher Batty. A multilevel active-set preconditioner for box-constrained pressure poisson solvers. *Proc. ACM Comput. Graph. Interact. Tech.*, 6(3), aug 2023.

[5] W Briggs, V Henson, and S Mccormick. *A Multigrid Tutorial, Second Edition.* Society for Industrial and Applied Mathematics, 2000.

[6] Robert Bridson. *Fluid Simulation for Computer Graphics.* A K Peters/CRC Press, 2015.