Optimized Schwarz-based Nonlinear Preconditioning for Elliptic PDEs

Yaguang Gu and Felix Kwok

1 Introduction

In this paper, we consider the following nonlinear elliptic equation,

$$\begin{cases} \eta u - \nabla \cdot (a(x,u,\nabla u)\nabla u) = f \text{ in } \Omega, \\ \mathcal{B}u = h \text{ on } \partial\Omega, \end{cases} \tag{1}$$

where $\eta \geq 0$, $a(x,u,\nabla u)$ is a positive scalar function uniformly bounded away from zero, and $\mathcal{B}u$ represents boundary conditions (e.g. Dirichlet or Neumann) such that the problem is well posed. This type of equation often arises from the implicit discretization of a time-dependent problem or from a steady state calculation, for example the Forchheimer equation [5] in porous media flow.

Once the problem (1) is discretized, there are many ways to solve the large nonlinear algebraic problem by domain decomposition methods. A classical approach is to use the Newton-Krylov-Schwarz method [1]: the problem is first attacked by Newton's method, and within each Newton iteration, the linearized problem is solved using a Krylov method with a Schwarz domain decomposition preconditioner. Alternative approaches consist of applying these components in a different order. One such possibility, known as the Nested Iteration approach, was formulated in [7, 8] for nonlinear parabolic PDEs: the solution (in space and time) is first rewritten as the fixed point of a parallel Schwarz waveform relaxation iteration. Next, using the interface values as primary unknowns, one derives the fixed point equation, which is then solved using Newton's method. Within each Newton iteraiton, the Jacobian

Yaguang Gu
Department of Mathematics, Hong Kong Baptist University, Kowloon Tong, Hong Kong, e-mail: 1648298@life.hkbu.edu.hk

Felix Kwok
Department of Mathematics, Hong Kong Baptist University, Kowloon Tong, Hong Kong e-mail: felix_kwok@hkbu.edu.hk

systems are solved by a Krylov method, where each matrix-vector multiplication corresponds to the solution of a linear parabolic problem.

For elliptic problems, the authors of [3] introduced the Restricted Additive Schwarz Preconditioned Exact Newton (RASPEN) method, which can be regarded as the Newton-accelerated version of the Restricted Additive Schwarz (RAS) method with classical (Dirichlet) transmission conditions. Similar to Nested Iteration, the RAS method is first written in fixed point form, and the resulting fixed point equation is solved by Newton's method, using a Krylov method as linear solver for calculating the Newton step. Unlike the ASPIN method [2], which uses approximate Jacobians, RASPEN uses exact Jacobians: it was shown in [3] that the product of the exact Jacobian matrix with an arbitrary vector can be obtained using components already computed during the subdomain solves, so the Newton corrections can be calculated cheaply. Thus, RASPEN is a true Newton method and converges quadratically close to the solution. Nonetheless, the Krylov solver within each Newton iteration converges relatively slowly, which is typical of classical RAS methods with Dirichlet transmission conditions. In this paper, we propose an optimized RASPEN (ORASPEN) method, where a zeroth order optimized (i.e. Robin) transmission condition is used to communicate information across subdomain interfaces. This allows us to take advantage of the extra Robin parameter to obtain faster convergence in the Krylov solver, just like in optimized Schwarz methods for linear problems.

2 The ORASPEN method

In this section, we derive the ORASPEN method and explain how the matrix-vector multiplication by the Jacobian can be performed by reusing components from the subdomain solves. We first recall the RASPEN method with classical transmisison conditions, as defined in [3]. Assume that the physical domain Ω is decomposed into overlapping subdomains $\Omega = \bigcup_{i=1}^{K} \Omega_i$. Then given the n-th iterate u^n, the restricted Additive Schwarz (RAS) method first calculates $u_i^{n+1} = G_i(u^n), \quad i = 1, 2, \ldots, K$, where G_i is the local solution operator which produces solutions to local subdomain problems by freezing degrees of freedom outside Ω_i. More concretely, suppose we use a finite element discretization of (1) to obtain for the ℓ-th degree of freedom

$$F_\ell(u) = \int_\Omega (\eta u \phi_\ell + a(x, u, \nabla u)\nabla u \cdot \nabla \phi_\ell)\, dx - \int_\Omega f \phi_\ell \, dx = 0, \tag{2}$$

where $\phi_\ell \in H_0^1(\Omega)$ denotes the ℓ-th finite element basis function. Let $F(u) = (F_1(u), F_2(u), \ldots)^T$ be the set of all such equations, so that the global nonlinear problem has the form $F(u) = 0$. If u^n is a finite element function whose trace on $\partial\Omega_i$ is used as Dirichlet values for the subdomain solve on Ω_i, then the subdomain solution $u_i^{n+1} = G_i(u^n) \in V_i$ can be obtained by solving the equation

$$R_i F(P_i G_i(u^n) + (I - P_i R_i)u^n) = 0 \tag{3}$$

for the unknown $G_i(u^n)$, where R_i is the restriction operator from the finite element space $V \subset H_0^1(\Omega)$ to the subspace $V_i = V \cap H_0^1(\Omega_i)$, and $P_i = R_i^T$ is the prolongation operator. Note that the subdomain solution u_i^{n+1} is none other than the solution to the following problem when we apply the parallel classical Schwarz method for (1):

$$\begin{cases} \eta u_i^{n+1} - \nabla \cdot (a(x, u_i^{n+1}, \nabla u_i^{n+1}) \nabla u_i^{n+1}) = f & \text{in } \Omega_i, \\ \mathcal{B} u_i^{n+1} = h & \text{on } \partial\Omega_i \cap \partial\Omega, \\ u_i^{n+1} = u_j^n & \text{on } \partial\Omega_i \cap \bar{\Omega}_j,\ j \in \mathcal{I}_i, \end{cases} \tag{4}$$

where $\mathcal{I}_i$ contains the indices of all the subdomains that have overlap with Ω_i.

Once the $G_i(u^n)$ are calculated for each i, the new global iterate is formed using the relation

$$u^{n+1} = \sum_{i=1}^{K} \tilde{P}_i G_i(u^n). \tag{5}$$

Here, $\tilde{P}_i$ is the *restricted* prolongation operator, formed from the P_i above and a partition of unity, so that the relation $\sum_{i=1}^{K} \tilde{P}_i R_i = I$ holds; see detailed definitions in [3]. When the iteration (5) converges, it does so linearly in general. The RASPEN idea consists of forming the fixed point equation

$$\tilde{\mathcal{F}}(u) = \sum_{i=1}^{K} \tilde{P}_i G_i(u) - u = 0 \tag{6}$$

and applying Newton's method to solve (6). This requires calculating the Jacobian $\tilde{\mathcal{F}}'(u)$, which in turn requires the derivative $G_i'(u)$. The latter can be obtained by differentiating (3).

We now derive the ORASPEN algorithm by showing how to incorporate optimized transmission conditions. In the ORASPEN algorithm, we still solve (6) by Newton, except that the underlying fixed point iteration (5) is replaced by the optimized RAS method of [9], so the local solution operator $G_i(u)$ is now based on Robin transmission conditions rather than Dirichlet.

Let $u_i^* = R_i u^*$ be the restriction of u^* to Ω_i, u^* being the solution to (1). Given a set of initial guesses $(u_i^0)_{i=1}^K$, the parallel optimized Schwarz method generates a sequence $(u_i^n)_{i=1}^K$, $n = 0, 1, \ldots$, that approximate $(u_i)_{i=1}^K$ by

$$\begin{cases} \eta u_i^{n+1} - \nabla \cdot (a(x, u_i^{n+1}, \nabla u_i^{n+1}) \nabla u_i^{n+1}) = f & \text{in } \Omega_i, \\ \mathcal{B} u_i^{n+1} = h & \text{on } \partial\Omega_i \cap \partial\Omega, \\ a(x, u_i^{n+1}, \nabla u_i^{n+1}) \frac{\partial u_i^{n+1}}{\partial \mathbf{n}_i} + p u_i^{n+1} = a(x, u_j^n, \nabla u_j^n) \frac{\partial u_j^n}{\partial \mathbf{n}_i} + p u_j^n & \text{on } \partial\Omega_i \cap \bar{\Omega}_j,\ j \in \mathcal{I}_i, \end{cases} \tag{7}$$

where p is the Robin parameter and $\mathbf{n}_i$ is the unit outward-pointing normal vector. If finite elements are used to discretize (7), then for each basis function ϕ_ℓ^i with support in Ω_i, the corresponding residual function becomes

$$F_\ell^i(u_i^{n+1}) = A_\ell^i(u_i^{n+1}) - \int_{\Omega_i} f \phi_\ell^i \, dx - \int_{\Gamma_i} g \phi_\ell^i \, ds, \tag{8}$$

where $g = \left(a(u_j^n, \nabla u_j^n)\frac{\partial}{\partial \mathbf{n}_i} + p\right)(u_j^n)$, $\Gamma_i = \partial\Omega_i \setminus \partial\Omega$, and

$$A_\ell^i(u_i) = \int_{\Omega_i} (\eta u_i \phi_\ell^i + a(x, u_i, \nabla u_i)\nabla u_i \cdot \nabla \phi_\ell^i)\, dx + \int_{\Gamma_i} p u_i \phi_\ell^i \, ds.$$

The evaluation of g, which involves Robin traces and must be taken in the weak sense, is non-trivial. Therefore, we mimic the approach in [4] for the linear case and exploit the equivalence between optimized parallel Schwarz and optimized RAS: we update the local solution via the full approximation scheme

$$A^i(u_i^{n+1}) - A^i(R_i u^n) = -R_i F(u^n), \tag{9}$$

where $A^i(u_i) = (A_1^i(u_i), A_2^i(u_i), \ldots)^T$, and $F(u^n) = (F_1(u^n), F_2(u^n), \ldots)^T$ is the global residual as defined in (2). Under the usual coercivity assumptions, (9) defines a mapping $G_i : u^n \mapsto u_i^{n+1}$. The fixed point iteration is completed by the update formula $u^{n+1} = \sum_{i=1}^K \tilde{P}_i G_i(u^n)$, as in (5). It is clear from (9) that if $u^n = u^*$ is the exact solution of $F(u) = 0$, then $u_i^{n+1} = R_i u^n$, so the exact solution is a fixed point of the iteration. Thus, the ORASPEN approach consists of solving (6), but with the G_i now defined by (9) instead of (3).

To calculate the Newton steps necessary for the solution of (6), one must solve linear systems involving the Jacobian matrix $\tilde{\mathcal{F}}'(u)$. Since (O)RASPEN uses Krylov methods for solving such linear systems, we need to know how to multiply $\tilde{\mathcal{F}}'(u)$ by an arbitrary vector v. Differentiating (6) with respect to u and multiplying the result by v gives

$$\tilde{\mathcal{F}}'(u^n)v = \sum_{i=1}^K \tilde{P}_i G_i'(u^n)v - v. \tag{10}$$

To evaluate $G_i'(u^n)v$, we let $u_i^{n+1} = G_i(u^n)$ in (9) and differentiate implicitly to obtain

$$\frac{\partial A^i}{\partial u}(u_i^{n+1})G_i'(u^n) - \frac{\partial A^i}{\partial u}(R_i u^n)R_i = -R_i F'(u^n).$$

Isolating $G_i'(u^n)$ in the above and substituting into (10) yields

$$\tilde{\mathcal{F}}'(u^n)v = \sum_{i=1}^K \tilde{P}_i \left(\frac{\partial A^i}{\partial u}(u_i^{n+1})\right)^{-1} \left(\frac{\partial A^i}{\partial u}(R_i u^n)R_i v - R_i F'(u^n)v\right) - v. \tag{11}$$

Note that $\frac{\partial A^i}{\partial u}(u_i^{n+1})$ is none other than the Jacobian matrix for the subdomain problem (9). If Newton's method was used to solve these subdomain problems, this Jacobian would have already been formed and factored during the calculation of u_i^{n+1}, so the multiplication by $\left(\frac{\partial A^i}{\partial u}(u_i^{n+1})\right)^{-1}$ in (11) requires only a forward-backward substitution involving the precomputed LU factors. Thus, the Krylov iterations have relatively low computational cost.

To understand the convergence of the Krylov method, it is instructive to consider the linear case, when $a(x, u, \nabla u) \equiv a(x)$ is independent of u, and $\frac{\partial A^i}{\partial u} =: J_i$ is

independent of u_i^{n+1}. In that case, (11) simplifies to

$$\tilde{\mathcal{F}}'(u^n)v = -\sum_{i=1}^{K} \tilde{P}_i J_i^{-1} R_i F'(u^n)v,$$

which is identical to the preconditioned matrix for optimized RAS [9]. Therefore, for well-chosen Robin parameters, we expect ORASPEN to exhibit much faster convergence than classical RASPEN in terms of inner Krylov iterations, even when the number of outer Newton iterations remains similar. This will be verified experimentally in the next section.

3 Numerical Results

In this section, we illustrate the behaviour of ORASPEN by comparing it with classical one-level RASPEN, as defined in [3], for two model problems. All tests in this section are discretized using the P1 (conforming piecewise linear) finite element method. In the first test, we show results for the nonlinear diffusion problem

$$\begin{cases} -\nabla \cdot ((1+u^2)\nabla u) = x\sin(y) & \text{in } \Omega = [0,1]\times[0,1], \\ u = 1 & \text{on } x = 1, \\ \frac{\partial u}{\partial n} = 0 & \text{elsewhere,} \end{cases} \tag{12}$$

with the initial guess $u^0 = 1$.

We compare the linear and nonlinear iteration counts needed by ORASPEN with those needed by RASPEN for the 4×4 subdomain test case, using different Robin parameters p and mesh ratios H/h. In Table 1, we report the following numbers:

- **Nits**, the number of outer Newton iterations required for convergence to within a tolerance of 10^{-8};
- **Lits**, the number of linearized subdomain problems that must be solved. This number includes (i) all linear solves within the subdomain problems, and (ii) all multiplications by the matrix $(\partial A^i/\partial u)^{-1}$ within GMRES due to Equation (11);
- **Avg Lits**, the average number of linear iterations per Neweton step; and
- p, the Robin parameter that leads to the lowest iteration counts for each mesh ratio H/h.

We also include the number of unpreconditioned classical Newton iterations required for convergence. Although one cannot use these numbers to directly compare classical Newton with (O)RASPEN (we must also consider which preconditioner to use, and how many preconditioned GMRES iterations are required by the Jacobian solves within each Newton step), such numbers are useful for determining the difficulty of the unpreconditioned problem. Here, we observe that ORASPEN always requires the fewest nonlinear iterations, compared to classical Newton and RASPEN.

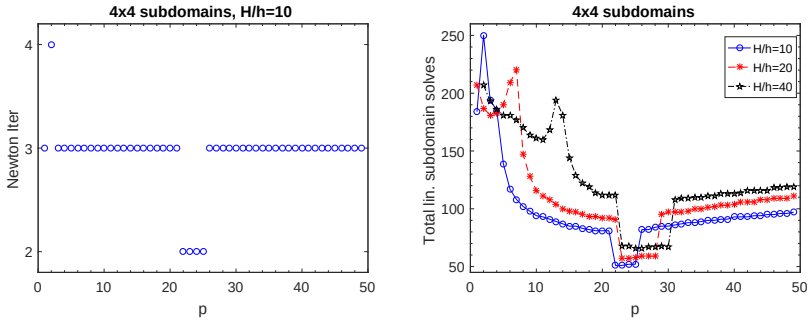

Fig. 1: Numerical results for the nonlinear diffusion problem. Left: Newton iteration counts for 4×4 subdomains with $H/h = 10$, H being the diameter of the subdomain. Right: Total linear iteration counts for the 4×4 subdomain case with different Robin parameters and mesh sizes.

We show in Figure 1 the linear and nonlinear iteration counts as a function of the Robin parameter p for different ratios H/h. We can see that the Robin parameter p has a large impact on the linear and nonlinear iteration counts. Observe that when ORASPEN is used with the optimal Robin parameter, the average number of linear iterations per Newton becomes much lower for ORASPEN than for RASPEN, and this number does not grow as quickly as for RASPEN when we refine the mesh. Moreover, with the optimal Robin parameter p, ORASPEN converges in two Newton iterations, which is slightly better than the three iterations required by RASPEN. Note that these extra savings are tolerance dependent: if we change TOL in the stopping criterion from 10^{-8} to 10^{-10}, then it will take at least three Newton iterations for ORASPEN to converge. Nevertheless, since ORASPEN needs fewer linear iterations per Newton step than RASPEN, ORASPEN will still outperform RASPEN, even when both methods take three Newton iterations to converge.

Next, we fix the ratio H/h and vary the number of subdomains; see the results in Table 2. We observe that ORASPEN again requires fewer linear iterations to converge than RASPEN, which is consistent with the linear case [6]. However, the iteration counts for both methods grow with the number of subdomains, as the inner subdomains move farther and farther away from the physical boundary.

For the second set of tests, we consider the same Forchheimer problem as in [3]:

$$\begin{cases} -\nabla \cdot \mathbf{q} = 0 & \text{in } \Omega = [0,1] \times [0,1], \\ \mathbf{q} \cdot \mathbf{n} = 0 & \text{on } \partial\Omega \setminus (\Gamma_{d0} \cup \Gamma_{d1}), \\ u = 0 \text{ on } \Gamma_{d0}, & u = 1 \quad \text{on } \Gamma_{d1}, \end{cases} \tag{13}$$

where

$$\mathbf{q} = \frac{2\Lambda(x,y)\nabla u}{1 + \sqrt{1 + 4\beta|\Lambda(x,y)\nabla u|}},$$

$\Gamma_{d0} = \{(x,y) \in \partial\Omega; x + y < 0.2\}$ and $\Gamma_{d1} = \{(x,y) \in \partial\Omega; x + y > 1.8\}$. The permeability $\Lambda(x,y)$ is equal to 1000 except in the two inclusions $[0, 0.5] \times [0.2, 0.4]$ and $[0.5, 1] \times [0.6, 0.8]$, where it is equal to 1. The nonlinearity of the Forchheimer

Table 1: Linear and nonlinear iteration counts for 4×4 subdomains for the nonlinear diffusion problem with different mesh sizes. An overlap of 4-cell widths and a stopping criterion of 10^{-8} are used for all tests.

H/h	Classical Newton	RASPEN			ORASPEN			
	Nits	Nits	Lits	Avg Lits	p	Nits	Lits	Avg Lits
10	4	3	113	37.67	22	2	51	25.50
20	4	3	152	50.67	23	2	57	28.50
40	4	3	208	69.33	25	2	66	33.00

Table 2: Linear and nonlinear iteration counts for different numbers of subdomains for the nonlinear diffusion problem with the fixed ratio $H/h = 10$. An overlap of 4-cell widths and a stopping criterion of 10^{-8} are used for all tests.

$N \times N$	Classical Newton	RASPEN			ORASPEN			
	Nits	Nits	Lits	Avg Lits	p	Nits	Lits	Avg Lits
2×2	4	3	59	19.67	12	2	34	17.00
4×4	4	3	113	37.67	22	2	51	25.50
8×8	4	3	211	70.33	28	3	133	44.33
16×16	4	3	418	139.33	31	3	247	82.33

equation is much stronger than in the first test problem, due to the appearance of ∇u in the denominator of $\mathbf{q}$ and the large contrast in $\Lambda(x, y)$. Therefore, we adopt the continuation approach, where we solve (13) first for $\beta = 0$ (which is a linear problem), then for $\beta = 0.1$ and $\beta = 1$, using the solution for the previous β as the initial guess for the next one. (Without continuation, classical Newton takes 15–20 iterations to converge, whereas (O)RASPEN takes only 4–8 in our examples.) For the *fixed* fine mesh shown in Figure 2, we vary the number of subdomains and show the iteration counts for ORASPEN and RASPEN in Table 3. We again observe significantly lower linear iteration counts in ORASPEN than in classical RASPEN. Finally, we remark that the performance of ORASPEN is sensitive to the Robin parameter p, as can be seen from Figure 3. A poor choice of the Robin parameter may lead to a higher number of nonlinear iterations compared to classical RASPEN, negating the benefits of faster linear convergence. A good parameter choice for this problem, and more generally for ORASPEN, is therefore the subject of ongoing work.

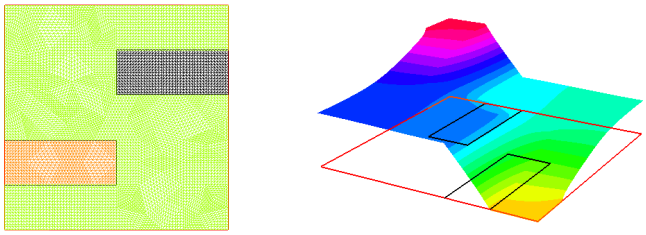

Fig. 2: Fixed grid for the 2×2, 4×4 and 8×8 subdomain test cases, and the solution profile for the Forchheimer problem.

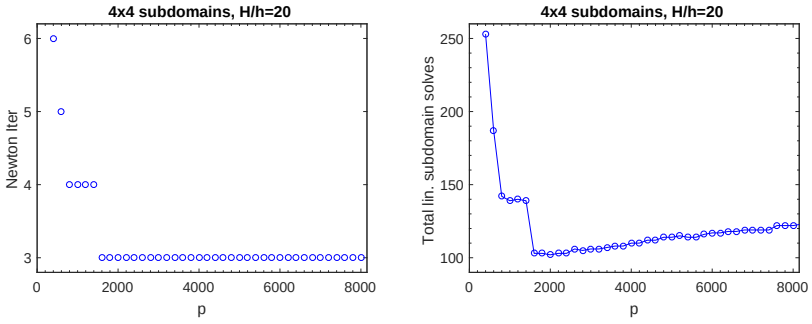

Fig. 3: Newton and linear iteration counts for the Forchheimer problem as a function of p.

Table 3: Linear and nonlinear iteration counts for the Forchheimer problem, with a continuation sequence of $\beta = 0, 0.1, 1$. An overlap of 4-cell widths and a stopping criterion of 10^{-8} are used for all tests.

β	$N \times N$	Classical Newton	RASPEN			ORASPEN			
		Nits	Nits	Lits	Avg Lits	p	Nits	Lits	Avg Lits
0.1	2×2	5	3	112	37.33	1800	3	70	23.33
	4×4	5	3	158	52.67	2000	3	103	34.33
	8×8	5	3	236	78.67	2200	4	218	54.50
1.0	2×2	4	3	109	36.33	200	2	43	21.50
	4×4	4	2	101	50.50	950	2	69	34.50
	8×8	4	3	232	77.33	1050	3	161	53.67

Acknowledgements This work is partially supported by the Hong Kong Research Grants Council (ECS-22300115) and the National Natural Science Foundation of China (RFYS-11501483).

References

1. Cai, X.C., Gropp, W.D., Keyes, D.E., Tidriri, M.D.: Newton-Krylov-Schwarz methods in CFD. In: Numerical methods for the Navier-Stokes equations, pp. 17–30. Springer (1994)
2. Cai, X.C., Keyes, D.E.: Nonlinearly preconditioned inexact Newton algorithms. SIAM J. Sci. Comput. **24**(1), 183–200 (2002)
3. Dolean, V., Gander, M.J., Kheriji, W., Kwok, F., Masson, R.: Nonlinear preconditioning: How to use a nonlinear Schwarz method to precondition Newton's method. SIAM J. Sci. Comput. **38**(6), A3357–A3380 (2016)
4. Dolean, V., Jolivet, P., Nataf, F.: An Introduction to Domain Decomposition Methods: Algorithms, Theory, and Parallel Implementation, vol. 144. SIAM (2015)
5. Forchheimer, P.: Wasserbewegung durch Boden. Z. Ver. Deutsch, Ing. **45**, 1782–1788 (1901)
6. Gander, M.J.: Optimized Schwarz methods. SIAM J. Numer. Anal. **44**(2), 699–731 (2006)
7. Haeberlein, F.: Time Space Domain Decomposition Methods for Reactive Transport — Application to CO2 Geological Storage. Ph.D. thesis, Université Paris-Nord - Paris XIII (2011). URL `https://tel.archives-ouvertes.fr/tel-00634507`
8. Haeberlein, F., Halpern, L., Michel, A.: Newton-Schwarz optimised waveform relaxation Krylov accelerators for nonlinear reactive transport. In: Domain decomposition methods in science and engineering XX, pp. 387–394. Springer (2013)
9. St-Cyr, A., Gander, M.J., Thomas, S.J.: Optimized multiplicative, additive, and restricted additive Schwarz preconditioning. SIAM J. Sci. Comput. **29**(6), 2402–2425 (2007)