A Multiscale Domain Decomposition Method for Flow and Transport Problems

Victor Ginting[1] and Bradley McCaskill[2]

1 Background

It has been widely recognized that one of the major challenges in the simulation of flow and transport problems is finding the numerical solution of the pressure equation [3]. Typically we seek to find the pressure solution, p, such that

$$\begin{cases} -\nabla \cdot (k \nabla p) = f & \text{in } \Omega \\ p = p_{\mathrm{D}} & \text{on } \Gamma_{\mathrm{D}} \\ -k \nabla p \cdot \boldsymbol{n} = g_{\mathrm{N}} & \text{on } \Gamma_{\mathrm{N}}, \end{cases} \tag{1}$$

where k represents the positive elliptic coefficient, and f represents a forcing function. The associated Dirichlet, and Neumann boundary conditions are given by p_{D} and g_{N} respectively. The corresponding variational formulation is to find p, with $(p - p_{\mathrm{D}}) \in V$, that satisfies

$$a(p, v) = \ell(v) \quad \forall v \in V, \tag{2}$$

where $V = \{v \in H^1(\Omega) : v = 0 \text{ on } \Gamma_{\mathrm{D}}\}$, and

$$a(p, v) = \int_{\Omega} k \nabla p \cdot \nabla v \, \mathrm{d}\boldsymbol{x}, \text{ and } \ell(v) = \int_{\Omega} f v \, \mathrm{d}\boldsymbol{x} - \int_{\Gamma_{\mathrm{N}}} g_{\mathrm{N}} v \, \mathrm{d}\boldsymbol{l}. \tag{3}$$

Assuming sufficient regularity of the data, the Lax-Milgram Theorem guarantees a unique solution of (3). The chief difficulty in approximating p stems from the heterogeneity of k, which can occur in multiple scales. This heterogeneity directly dictates the degree of the mesh resolution on which the approximate solution is found. In turn this results in a very high dimensional algebraic system which must be solved.

Department of Mathematics, University of Wyoming, WY 82071, USA vginting@uwyo.edu · Department of Mathematics, University of Wyoming, WY 82071, USA bmccaski@uwyo.edu

With the advances of parallel computing, domain decomposition as a general framework has gained a stronger role in efficiently finding accurate solutions to problems of this type. In this paper, we propose an iterative procedure for solving (3) that relies on a one-time preprocessing step where a set of independent subdomain problems are computed. This preprocessing step yields a set of so called multiscale basis functions with which the global solution is represented. Continuity of the solution at the interface is established by imposing Robin Transmission conditions on each subdomain interface. This imposition is accomplished in an iterative manner. In the following section we describe an iterative domain decomposition technique that serves as the backbone for our proposed procedure.

2 A Domain Decomposition with Robin Transmission Conditions

We decompose the domain Ω into a set of non-overlapping subdomains $\{\Omega_j\}_{j=1}^{N_{\rm sd}}$, and construct a local problem on each subdomain. For ease of notation we define $\mathcal{N}_m$ to be the set of indices for subdomains that share an edge with Ω_m. For example, $\mathcal{N}_m = \{l, r, b, t\}$ is associated with the subdomain presented in Figure 1. Each local problem is supplied with a boundary condition that allows for the continuity of the solution and its flux at each subdomain interface to be maintained. In particular, for each $n \in \mathcal{N}_m$, we impose

$$-k\nabla p_m \cdot \boldsymbol{e}_m - \gamma_{mn} p_m = g_{mn} \quad \text{on } \Gamma_{mn}, \tag{4}$$

where γ_{mn} is a positive constant, $\boldsymbol{e}_m$ represents the exterior unit normal respective to subdomain Ω_m, $\Gamma_{mn} = \partial\Omega_m \cap \partial\Omega_n$, and the value of g_{mn} comes from the neighbouring subdomain Ω_n, expressed as

$$g_{mn} = k\nabla p_n \cdot \boldsymbol{e}_n - \gamma_{nm} p_n \quad \text{on } \Gamma_{mn}. \tag{5}$$

To establish the iterative procedure, it is assumed that g_{mn} is known, namely, from the previous iteration level. The resulting local variational formulation is to find $p_m \in H^1(\Omega_m)$ such that

$$a_m(p_m, w) + \sum_{n\in\mathcal{N}_m} b_{mn}(p_m, w) = \ell_m(w) + \sum_{n\in\mathcal{N}_m} r_{mn}(w) \ \ \forall w \in H^1(\Omega_m), \tag{6}$$

where

$$\begin{aligned} a_m(v, w) &= \int_{\Omega_m} k\nabla v \cdot \nabla w \ \mathrm{d}\boldsymbol{x}, \quad b_{mn}(v, w) = \int_{\Gamma_{mn}} \gamma_{mn} v w \ \mathrm{d}\boldsymbol{l}, \\ \ell_m(w) &= \int_{\Omega_m} f w \ \mathrm{d}\boldsymbol{x}, \quad r_{mn}(w) = -\int_{\Gamma_{mn}} g_{mn} w \ \mathrm{d}\boldsymbol{l}. \end{aligned} \tag{7}$$

We use (6) to develop an iterative technique for approximating (2) whose algorithm is presented in Algorithm 1. At the practical level, this iteration does converge to the true solution [2, 4, 5], but it requires that we calculate a new local solution on every subdomain for each step of the iteration. Depending on the initial guess, and the number of subdomains, this can greatly exceed the computational time required to solve the problem with traditional methods.

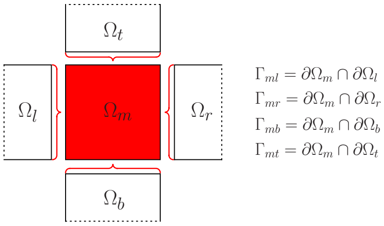
Ω_t
Ω_l
Ω_m
Ω_r
Ω_b
$\Gamma_{ml} = \partial\Omega_m \cap \partial\Omega_l$
$\Gamma_{mr} = \partial\Omega_m \cap \partial\Omega_r$
$\Gamma_{mb} = \partial\Omega_m \cap \partial\Omega_b$
$\Gamma_{mt} = \partial\Omega_m \cap \partial\Omega_t$

Fig. 1 An internal rectangular subdomain $\varOmega_m$, and its neighbouring subdomains $\{\varOmega_n\}$

Algorithm 1

```
Set initial guess for $\{p_m^{(0)}\}_{m=0}^{N_{\mathrm{sd}}}$
for $it = 1$ until convergence do
    Construct $g_{mn}^{(it-1)}$, for all $n \in \mathcal{N}_m$, $m = 1, \dots, N_{\mathrm{sd}}$
    Solve (6) to get $p_m^{(it)}$ for $m = 1, \dots, N_{\mathrm{sd}}$
end for
```

3 Incorporation of Multiscale Basis Functions

To alleviate the aforementioned burden of calculation, our strategy is to form a preprocessing step aimed at collecting the finescale heterogeneity information on each subdomain. This information is stored in the so called subdomain multiscale basis functions. Here our motivation is to find an approximate solution to (6) that is expressed as a linear combination of these multiscale basis functions.

For each $n \in \mathcal{N}_m$ we decompose $\varGamma_{mn}$ into a union of nonoverlapping segments $\{I_{mn}^i\}_{i=1}^{k_{mn}}$, and denote by $\{z_{mn}^i\}_{i=0}^{k_{mn}}$ the associated vertices. For simplicity we assume uniformity of these segments as they relate to neighbouring subdomains.

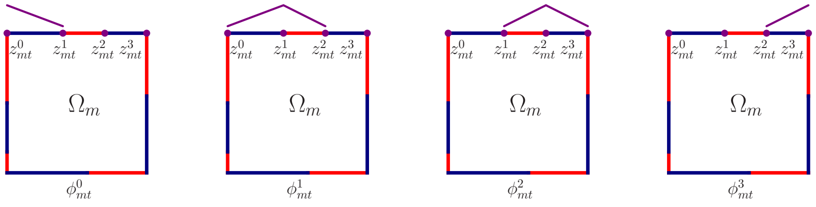

Fig. 2 Example of "hat" functions associated with an edge $\varGamma_{mt}$. On edges $\varGamma_{mb}, \varGamma_{mr}, \varGamma_{ml}$ the value of these functions is zero.

We set

$$\widetilde{g}_{mn} = \sum_{i=0}^{k_{mn}} g_{mn}(z^i_{mn})\phi^i_{mn}, \tag{8}$$

where $\{\phi^i_{mn}\}_{i=0}^{k_{mn}}$ is the usual "hat" nodal basis function corresponding to $\{z^i_{mn}\}_{i=0}^{k_{mn}}$ expressed in a parametric form associated with $\mathit{\Gamma}_{mn}$. Examples of these "hat" functions are presented in Figure 2. For our approximate solution we construct a new variational formulation. Find $\widetilde{p}_m \in H^1(\mathit{\Omega}_m)$, satisfying

$$a_m(\widetilde{p}_m, w) + \sum_{n\in\mathcal{N}_m} b_{mn}(\widetilde{p}_m, w) = \ell_m(w) + \sum_{n\in\mathcal{N}_m} \widetilde{r}_{mn}(w) \;\; \forall w \in H^1(\mathit{\Omega}_m), \tag{9}$$

where

$$\widetilde{r}_{mn}(w) = -\sum_{i=0}^{k_{mn}} g_{mn}(z^i_{mn}) \int_{\mathit{\Gamma}_{mn}} \phi^i_{mn} w \;\mathrm{d}\boldsymbol{l}. \tag{10}$$

With this formulation, the same iteration as in Algorithm 1 could have been done. It is worth noting that there are two sources of error that are committed when conducting the iteration based on (9). The first error is shared by the iteration using (6), namely resulting from the fact that in practice only a finite number of iterations are used. The second error stems from the replacement of g_{mn} by $\widetilde{g}_{mn}$, i.e., an approximation error. There is a nonlinear interaction between these two error components. We expect, however, that at the asymptotic level of systematic refinement ($k_{mn} \to \infty$ and convergence is reached), $\widetilde{p} = \sum_{m=1}^{N_{\mathrm{sd}}} \widetilde{p}_m 1_{\mathit{\Omega}_m}$ should converge to p. What is more important is that the formulation (9) provides a building block for the construction of subdomain multiscale basis functions as part of the preprocessing step. The approximate solution on each subdomain is then represented using these basis functions.

To each $\mathit{\Gamma}_{mn}$ we associate a set of multiscale basis functions $\{\psi^i_{mn}\}_{i=0}^{k_{mn}}$, where $\psi^i_{mn} \in H^1(\mathit{\Omega}_m)$ is the solution to the variational formulation

$$a_m(\psi^i_{mn}, w) + \sum_{n\in\mathcal{N}_m} b_{mn}(\psi^i_{mn}, w) = r^i_{mn}(w) \; \forall w \in H^1(\mathit{\Omega}_m). \tag{11}$$

The linear functional in (11) is given by

$$r^i_{mn}(w) = -\int_{\mathit{\Gamma}_{mn}} \phi^i_{mn} w \;\mathrm{d}\boldsymbol{l}. \tag{12}$$

When $f \neq 0$, we compute an extra multiscale basis function $\hat{\psi}_m \in H^1(\mathit{\Omega}_m)$ that satisfies

$$a_m(\hat{\psi}_m, w) + \sum_{n\in\mathcal{N}_m} b_{mn}(\hat{\psi}_m, w) = \ell_m(w) \; \forall w \in H^1(\mathit{\Omega}_m). \tag{13}$$

On each subdomain we set $V_m = \mathrm{span}\big\{\psi^i_{mn}, i = 1, \cdots, k_{mn}, n \in \mathcal{N}_m, \hat{\psi}_m\big\}$ and seek $\widetilde{p}_m \in V_m$, i.e.,

$$\widetilde{p}_m = \hat{\psi}_m + \sum_{n\in\mathcal{N}_m} \sum_{i=0}^{k_{mn}} \alpha^i_{mn} \psi^i_{mn} \approx p_m. \tag{14}$$

An approximation of the global solution is now recaptured by determining the values of each $\boldsymbol{\alpha}_{mn} = [\alpha^0_{mn}, \ldots, \alpha^{k_{mn}}_{mn}]$ that induce the continuity condition outlined in (4), and imposed in (6). Thus, for each ϕ_j associated with an interface edge Γ_{mn} we require

$$\sum_{j=0}^{k_{mn}} \alpha^j_{mn} \int_{\Gamma_{mn}} \phi^j_{mn} \phi^i_{mn} \; \mathrm{d}\boldsymbol{l} = \int_{\Gamma_{mn}} \widetilde{g}_{mn} \phi^i_{mn} \; \mathrm{d}\boldsymbol{l}, \quad \forall \; i = 0, \cdots, k_{mn} + 1. \tag{15}$$

Here we note that this continuity condition yields a linear system governing $\boldsymbol{\alpha}_{mn}$. The associated matrix is tridiagonal and of dimension $k_{mn} + 1$. At a practical level the calculation of $\widetilde{g}_{mn}$ can be performed using $\widetilde{p}_n$, the multiscale representation of p_n. The iterative procedure presented in Algorithm 1, is now modified to be an iteration governing each $\boldsymbol{\alpha}_{mn}$. The modified iteration is presented in Algorithm 2.

Algorithm 2

```
Calculate $\{\psi^i_{mn}\}_{i=1}^{k_{mn}}$, for all $n \in \mathcal{N}_m$, $m = 1, \ldots, N_{\mathrm{sd}}$
Set initial guess $\boldsymbol{\alpha}^{(0)}_{mn}$, for all $n \in \mathcal{N}_m$, $m = 1, \ldots, N_{\mathrm{sd}}$
for $it = 1$ until convergence do
    Calculate $\widetilde{g}^{(it-1)}_{mn}$, for all $n \in \mathcal{N}_m$, $m = 1, \ldots, N_{\mathrm{sd}}$
    Solve for $\boldsymbol{\alpha}^{(it)}_{mn}$, satisfying (15) for all $n \in \mathcal{N}_m$, $m = 1, \ldots, N_{\mathrm{sd}}$
    Set $\widetilde{p}^{(it)}_m$, $m = 1, \ldots, N_{\mathrm{sd}}$
end for
```

4 Numerical Examples

In this section we present two studies. First, we present a convergence study of our method when applied to a problem with a known solution. We then apply our method to a single phase flow model, and compare the results with traditional methods. To calculate the multiscale basis functions, we use the traditional continuous Galerkin FEM to solve (11) and (13).

4.1 Convergence Study

We first explore the behaviour of the approximate solution in terms of the discretization parameters. In particular, it is interesting to study the interaction between the subdomain and the segment configuration. The former determines how many local problems are created while the latter determines the number of multiscale basis functions to represent a particular local problem. The subdomain size is denoted by H and the segment size is denoted by $\widetilde{h}$. The interplay between the two parameters reflects a choice of balancing the accuracy and efficiency of the approximate solution.

For this purpose, we choose a problem with a known solution. The problem is posed in $(0,1)^2$ with a zero Neumann condition on $x_2 = 0, 1$ and a Dirichlet condition on $x_1 = 0, 1$. We assume that $f = 0$ and $k(\boldsymbol{x}) = a_1(x_1)a_2(x_2)$, where a_1 and a_2 are

$$\begin{aligned} a_1(x_1) &= [0.25 - 0.999(x_1 - {x_1}^2)\sin(11.2\pi x_1)]^{-1} \\ a_2(x_2) &= [0.25 - 0.999(x_2 - {x_2}^2)\cos(5.2\pi x_2)]^{-1}, \end{aligned}$$

yielding $k_{\max}/k_{\min} \approx 2 \times 10^4$. Comparison of the effect that various segment and subdomain configurations have on the accuracy of the resulting approximate solution are presented in Table 1. In this example, the finescale solution is found on a grid of 256×256 rectangles (i.e., $h = 1/256$) and this finescale mesh is the base for the configuration of $\widetilde{h}$ after H is determined.

	L_2		H_1	
$\widetilde{h}$	$H = 0.250$	$H = 0.125$	$H = 0.250$	$H = 0.125$
h	0.000217	0.000217	0.02070	0.02070
$2h$	0.000217	0.000218	0.02074	0.02079
$4h$	0.000223	0.000241	0.02137	0.02201
$8h$	0.000252	0.000485	0.03068	0.03559
$16h$	0.001159	0.002538	0.08002	0.10412

Table 1 Comparison of the L_2-norm, H_1-norm of the approximate solution found using various segment lengths, and subdomain sizes

We note that when $\widetilde{h} = h$ the resulting solution has exactly the same error estimates as solutions found with the traditional Galerkin FEM on the fine mesh. For a fixed H, the errors of the proposed method stay relatively unchanged as $\widetilde{h}$ is increased. This can be taken as a potential advantage of the proposed method; lower dimensional V_m can still produce a relatively accurate numerical solution. This of course reduces the number of multiscale basis functions which must be calculated. Furthermore, results in Table 1 indicate that the errors seem to be less sensitive to H. Traditionally, it has been established (see for example [4, 5]) that an increase in subdomain interfaces (i.e., the finer H is) can potentially increase the number of iterations needed for convergence to a desired tolerance. Thus, this indication suggests that only fewer subdomains (i.e., less interfaces) are required to extract accurate solutions, which results in fewer iterations for convergence. On the other hand, this can potentially mean that the multiscale basis functions are governed by a higher dimensional problem, which correlates to a higher computational load in the preprocessing step. In the end, a problem dependent choice of H and $\widetilde{h}$ leads to an optimized scenario of calculation.

4.2 Applications to Single Phase Flow

The mathematical model is

$$\partial_t S + \boldsymbol{u} \cdot \nabla\lambda(S) = 0, \quad \text{with specified I.C. + B.C. and } \boldsymbol{u} = -k(\boldsymbol{x})\nabla p,$$

where S represents the saturation and $\nabla \cdot \boldsymbol{u} = 0$, i.e., elliptic PDE governing the pressure p. The boundary condition for p is the same as the one in the previous subsection. The model is a typical one way coupling equation where the pressure is first solved and the velocity $\boldsymbol{u}$ is constructed from it, which in turn is used as an input in solving the transport equation. We applied the postprocessing technique [1] to recover a locally conservative flux $\boldsymbol{u} \cdot \boldsymbol{n}$ on the finescale grid. Then a first order up-winding scheme is used to determine the time evolved saturation value. The elliptic coefficient that is used for this model is shown in Figure 3. This elliptic coefficient is posed on 240×240 grid and has a ratio $k_{\text{max}}/k_{\text{min}} \approx 6.4 \times 10^4$. In Figure 4 we show a visual comparison of the saturation solution at various time steps, for our method and traditional methods. In Figure 5 a plot of the relative difference between the solution found with the proposed method and the solution found with traditional methods is presented.

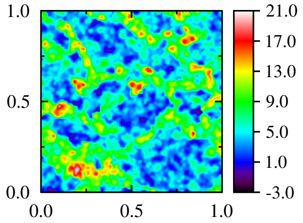

Fig. 3 A logarithmic plot of $k(\boldsymbol{x})$ used in single phase flow simulation

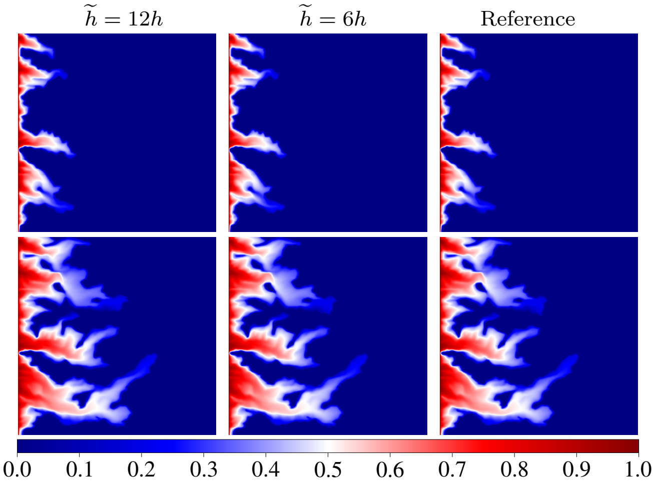

Fig. 4 Comparison of saturation at $t = 0.003$ (top), $t = 0.009$ (bottom), all results use $H = 0.25$.

5 Conclusion

We have proposed an iterative multiscale domain decomposition method with certain favourable properties. By incorporating the multiscale basis functions

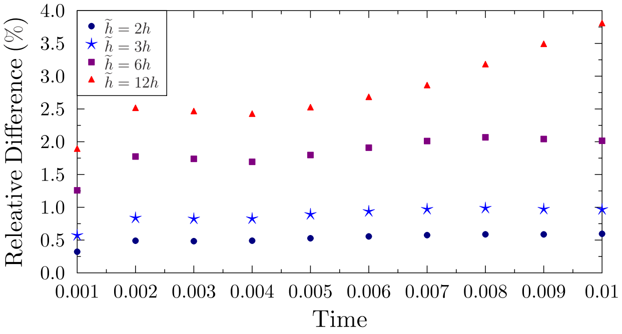

Fig. 5 Comparison of the L^2-error of the saturation difference between our method and traditional Galerkin FEM, for various choices of $\widetilde{h}$. In all cases, $H = 0.25$.

into an iterative domain decomposition procedure we have reduced its computational demand. The numerical examples suggest that our method is capable of recapturing accurate solutions that are comparable to those found with traditional methods. In the future we will extend the capability of the method to multiphase flow models. We are also interested in conducting a rigorous convergence analysis of the proposed method.

References

[1] L. Bush and V. Ginting. On the Application of the Continuous Galerkin Finite Element Method for Conservation Problems. *SIAM J. Sci. Comput.*, 35(6):A2953–A2975, 2013.

[2] Wenbin Chen, Max Gunzburger, Fei Hua, and Xiaoming Wang. A parallel Robin-Robin domain decomposition method for the Stokes-Darcy system. *SIAM J. Numer. Anal.*, 49(3):1064–1084, 2011.

[3] Zhangxin Chen, Guanren Huan, and Yuanle Ma. *Computational methods for multiphase flows in porous media*. Computational Science & Engineering. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2006.

[4] J. Douglas, Jr., P. J. Paes-Leme, J. E. Roberts, and Jun Ping Wang. A parallel iterative procedure applicable to the approximate solution of second order partial differential equations by mixed finite element methods. *Numer. Math.*, 65(1):95–108, 1993.

[5] Alfio Quarteroni and Alberto Valli. *Domain decomposition methods for partial differential equations*. Numerical Mathematics and Scientific Computation. The Clarendon Press Oxford University Press, New York, 1999. Oxford Science Publications.