A Vanka-based parameter-robust multigrid relaxation for the Stokes–Darcy Brinkman problems

Yunhui He

Department of Computer Science, The University of British Columbia, Vancouver, British Columbia, Canada

Correspondence
Yunhui He, Department of Computer Science, The University of British Columbia, Vancouver, BC, Canada. Email: yunhui.he@ubc.ca

Funding information
The University of British Columbia

Abstract
We consider a block-structured multigrid method based on Braess–Sarazin relaxation for solving the Stokes–Darcy Brinkman equations discretized by the marker and cell scheme. In the relaxation scheme, an element-based additive Vanka operator is used to approximate the inverse of the corresponding shifted Laplacian operator involved in the discrete Stokes–Darcy Brinkman system. Using local Fourier analysis, we present the stencil for the additive Vanka smoother and derive an optimal smoothing factor for Vanka-based Braess–Sarazin relaxation for the Stokes–Darcy Brinkman equations. Although the optimal damping parameter is dependent on meshsize and physical parameter, it is very close to one. In practice, we find that using three sweeps of Jacobi relaxation on the Schur complement system is sufficient. Numerical results of two-grid and V(1,1)-cycle are presented, which show high efficiency of the proposed relaxation scheme and its robustness to physical parameters and the meshsize. Using a damping parameter equal to one gives almost the same convergence results as these for the optimal damping parameter.

KEYWORDS
Braess–Sarazin relaxation, local Fourier analysis, multigrid, Stokes–Darcy Brinkman equations, Vanka smoother

1 | INTRODUCTION

The numerical solution of fluid flow problems is an important topic in computational science and engineering, which has received much attention in the last few decades. Stokes–Darcy Brinkman problem is one of them used to model fluid motion in porous media with fractures. The discretization of the fluid flow problems often leads to a saddle-point system, which is ill-conditioned. Designing fast numerical solution of these problems is often challenging due to the small magnitude of physical parameters of the model.

We consider the multigrid numerical solution of the Stokes–Darcy Brinkman equations

\[ -\varepsilon^2 \Delta u + u + \nabla p = f \quad \text{in } \Omega, \]
\[ \nabla \cdot u = g \quad \text{in } \Omega, \]
\[ u = 0 \quad \text{on } \partial \Omega, \]
where \( \epsilon \in (0, 1] \). The source term \( g \) is assumed to satisfy the solvability condition

\[
\int_{\Omega} g \, d\Omega = 0.
\]

Then, Equations (1a)–(1c) have a unique solution.

A variety of discretization schemes are available for Equations (1a)–(1c), including finite element methods,\(^7\)–\(^{13}\) finite difference techniques,\(^14\),\(^15\) and divergence-conforming B-spline methods.\(^16\) When \( \epsilon = 0 \), the model problem is reduced to the Darcy problem.\(^17\),\(^18\) For \( \epsilon \in (0, 1] \), designing a robust discretization and numerical solver is challenging. The convergence rate deteriorates as the Stokes–Darcy Brinkman becomes Darcy-dominating when certain stable Stokes elements are used,\(^19\) for example, Taylor–Hood element. While, as the Stokes–Darcy Brinkman problem becomes Stokes-dominating when Darcy stable elements such as the lowest order Raviart–Thomas elements are used, degradation on convergence is observed.\(^20\)

Upon discretization, large-scale indefinite linear systems typically need to be solved, at times repeatedly. For saddle-point systems, within the context of multigrid, there are several effective block-structured relaxation schemes for solving such linear systems, such as Braess–Sarazin smoother,\(^21\)–\(^{23}\) distributive smoother,\(^22\),\(^24\) Schwarz-type smoothers,\(^25\) Vanka smoother,\(^26\)–\(^{30}\) and Uzawa-type relaxation.\(^27\),\(^31\)–\(^{33}\)

We note also that a number of effective preconditioning methods are available for the Stokes–Darcy Brinkman problems, for example the scalable block diagonal preconditioner,\(^34\) and Uzawa-type preconditioning.\(^35\),\(^36\) Multigrid methods are studied in depth.\(^37\)–\(^{41}\) Braess–Sarazin, Uzawa, and Vanka smoothers within multigrid with finite element discretization have been discussed.\(^40\) However, the convergence rate is highly dependent on physical parameters. A Gauss–Seidel smoother based on a Uzawa-type iteration is studied,\(^27\) where the authors provide an upper bound on the smoothing factor. Moreover, the performance of Uzawa with a Gauss–Seidel type coupled Vanka smoother\(^5\) has been investigated,\(^27\) in which the pressure and the velocities in a grid cell, are updated simultaneously, showing that the actual convergence of the W-cycle of Uzawa is approximately the same as that obtained by the Vanka smoother.

Our interest is in the marker and cell scheme (MAC), a finite difference method on a staggered mesh. On a uniform mesh discretization, the method is second-order accurate for both velocity and pressure.\(^15\) We propose a Vanka-type Braess–Sarazin relaxation (V-BSR) scheme for the Stokes–Darcy Brinkman equations discretized by the MAC scheme on staggered meshes. In contrast to the Vanka smoother,\(^27\) our work builds an algorithm that decouples velocity and pressure, which is often preferred considering the cost efficiency. Specifically, in our relaxation scheme, the shifted Laplacian operator, \(-\epsilon^2 \Delta u + u\), is solved by an additive Vanka-type smoother. Instead of solving many subproblems involved in Vanka setting, we derive the stencil of the Vanka smoother, which means that we can form the global matrix of the Vanka smoother. As a result, in our multigrid method we only have matrix-vector products. This represents significant savings compared to traditional methods that require computationally expensive exact solves; in V-BSR, we solve the Schur complement system by only two or three iterations of the Jacobi method, which achieves the same performance as that of exact solve. We apply local Fourier analysis (LFA) to select the multigrid damping parameter and predict the actual multigrid performance. From this analysis, we derive an optimal damping parameter and optimal smoothing factors. Those parameters are dependent on physical parameters and the meshsize, which means that we can propose adaptive damping parameter in each multigrid level. The optimal parameter turns out to be close to one and relatively insensitive to physical parameters and meshsize. This allows for an easy choice of an approximately optimal damping parameter. We quantitatively compare the results with optimal parameter and the value of one from LFA and present numerical results of two-grid and V-cycle multigrid to validate the high efficiency of our methods.

The rest of the work is organized as follows. In Section 2, we review the MAC scheme for our model problem and propose the afore-mentioned Vanka-based Braess–Sarazin relaxation. We apply LFA to study the smoothing process in Section 3, where optimal LFA smoothing factor is derived. In Section 4, we present our LFA predictions for the two-grid method and actual multigrid performance. Finally, we draw conclusions in Section 5.

### 2 Discretization and Relaxation

As mentioned in Section 1, we use throughout the well-known MAC scheme to solve (1). For the discretization of (1), a staggered mesh is needed to guarantee numerical stability. The discrete unknowns \( u, v, p \) are placed in different locations; see Figure 1. The stability and convergence of the MAC scheme for this problem has been studied.\(^15\)
The stencil representation of MAC for the Stokes–Darcy Brinkmann equations is

\[ K_h = \begin{pmatrix} -\varepsilon^2 \Delta_h + I & 0 & (\partial_x)_{h/2} \\ 0 & -\varepsilon^2 \Delta_h + I & (\partial_y)_{h/2} \\ -(\partial_x)_{h/2} & -(\partial_y)_{h/2} & 0 \end{pmatrix}, \]

(2)

where

\[ -\Delta_h = \frac{1}{h^2} \begin{bmatrix} -1 & -1 & 4 \\ -1 & 4 & -1 \end{bmatrix}, \quad (\partial_x)_{h/2} = \frac{1}{h} \begin{bmatrix} 1 & 0 \\ -1 \end{bmatrix}, \quad (\partial_y)_{h/2} = \frac{1}{h} \begin{bmatrix} 0 & 1 \\ -1 \end{bmatrix}. \]

After discretization, the corresponding linear system is

\[ K_h x_h = \begin{pmatrix} A & B^T \\ B & 0 \end{pmatrix} \begin{pmatrix} u_h \\ p_h \end{pmatrix} = \begin{pmatrix} f_h \\ g_h \end{pmatrix} = b_h, \]

(3)

where \( A \) is the matrix corresponding to the discretization of \(-\varepsilon^2 \Delta u + u\), and \( B^T \) is the discrete gradient.

In order to solve (3) efficiently by multigrid we use BSR, within the smoother

\[ M_h = \begin{pmatrix} C & B^T \\ B & 0 \end{pmatrix}, \]

(4)

where \( C \) is an approximation to \( A \). In the context of preconditioning, such an approach is known as constraint preconditioning\(^{42-44}\) and it has received quite a bit of attention due to its attractive property of computing interim approximate solutions that satisfy the constraints.

A number of studies\(^{22,45}\) have shown that the efficiency of solving the Laplacian will determine the convergence of BSR. To construct an efficient approximation \( C \), we first investigate the discrete operator \(-\varepsilon^2 \Delta u + u\) denoted by

\[ L = A + I, \]

(5)

where \( A \) corresponds to the five-point discretization of operator \(-\varepsilon^2 \Delta u\). The stencil notation for the discrete operator \(-\varepsilon^2 \Delta u + u\) is

\[ L = \frac{\varepsilon^2}{h^2} \begin{bmatrix} -1 & 4 + \frac{h^2}{\varepsilon^2} & -1 \\ -1 & 4 + r & -1 \\ -1 & -1 & -1 \end{bmatrix} = \frac{\varepsilon^2}{h^2} \begin{bmatrix} -1 & 4 + r & -1 \\ -1 & 4 + r & -1 \\ -1 & -1 & -1 \end{bmatrix}. \]

(6)

where \( r = \frac{h^2}{\varepsilon^2} \). When \( r = 0 \), (6) reduces to the discretization of \(-\varepsilon^2 \Delta u\).
Recently, we proposed an additive element-wise Vanka smoother\textsuperscript{46} for $\Delta u$. Our current goal is to extend our approach to (5). An immediate challenge here contrary to Reference\textsuperscript{46} is the difference in scale between the discretized scaled Laplacian and the identity operator.

Denote the element-wise smoother as $M_e$, which has the form

$$M_e = \sum_{j=1}^{N} V_j^T D_j L_j^{-1} V_j,$$

(7)

where $D_j = \frac{1}{4} I$ with $I$ be the $4 \times 4$-identity matrix, $L_j$ is the coefficient matrix of $j$th subproblem defined for one element, and $V_j$ is a restriction operator mapping the global vector to the $j$th subproblem. We consider

$$C^{-1} = \begin{pmatrix} M_e & 0 \\ 0 & M_e \end{pmatrix}.$$

The relaxation scheme for (3) is

$$x_h^{k+1} = x_h^k + \omega M_h^{-1} (b_h - K_h x_h^k).$$

(8)

We refer to the relaxation (8) as Vanka-based Braess–Sarazin relaxation (V-BSR).

Let $b_h - K_h x_h^k = (r_u, r_p)$. In (8), we need to solve for $(\delta u, \delta p) = M_h^{-1} (r_u, r_p)$ by

$$(BC^{-1} B^T) \delta p = BC^{-1} r_u - r_p,$$

$$\delta u = C^{-1} (r_u - B^T \delta p).$$

(9)

Solving (9) exactly is prohibitive and impractical in the current context, and it has been shown in a few studies\textsuperscript{22,23} that an inexact solve can be applied and perform well. In the sequel, we will present a smoothing analysis for the exact solve, but in practice, for assessing the performance of the multigrid scheme we apply a few iterations of weighted Jacobi to (9).

The relaxation error operator for (8) is given by

$$S_h = I - \omega M_h^{-1} K_h,$$

(10)

where $\omega$ is a damping parameter to be determined.

For a two-grid method, the error propagation operator is

$$E_h = S_h^{\nu_1} (I - P_h (K_{2h})^{-1} R_h K_h) S_h^{\nu_2},$$

(11)

where $K_{2h}$ is rediscritization for the coarse-grid operator and the integers $\nu_1$ and $\nu_2$ are the numbers of pre- and post-smoothing steps. For simplicity, we denote the overall number of those steps by $\nu = \nu_1 + \nu_2$. We consider simple restriction operators using six points for the $u$ and $v$ components of the velocity, that is,

$$R_{h,u} = \frac{1}{8} \begin{bmatrix} 1 & 1 \\ 2 & 2 \\ 1 & 1 \end{bmatrix}, \quad R_{h,v} = \frac{1}{8} \begin{bmatrix} 1 & 1 \\ 2 & \star \\ 1 & 2 \end{bmatrix},$$

where $\star$ denotes the position (on the coarse grid) at which the discrete operator is applied. For interpolation, we take $P_{h,u} = 4R_{h,u}^T$ and $P_{h,v} = 4R_{h,v}^T$. For the restriction for the pressure, we use

$$R_{h,p} = \frac{1}{4} \begin{bmatrix} 1 & 1 \\ \star & \star \\ 1 & 1 \end{bmatrix}.$$
and $P_{h,p} = 4R_{h,p}^T$. Consequently,

$$R_h = \begin{pmatrix} R_{h,u} & 0 & 0 \\ 0 & R_{h,v} & 0 \\ 0 & 0 & R_{h,p} \end{pmatrix}, \quad P_h = 4R_h^T.$$ 

### 3 LOCAL FOURIER ANALYSIS

To identify a proper parameter $\omega$ in (8) to construct fast multigrid methods, we apply LFA\textsuperscript{47,48} to examine the multigrid relaxation scheme. The LFA smoothing factor, see Definition 2, often offers a sharp prediction of actual multigrid performance.

**Definition 1.** Let $L_h = [s_k] h$ be a scalar stencil operator acting on grid $G_h$ as

$$L_h w_h(x) = \sum_{k \in V} s_k w_h(x + kh),$$

where $s_k \in \mathbb{R}$ is constant, $w_h(x) \in l^2(G_h)$, and $V$ is a finite index set. Then, the symbol of $L_h$ is defined as:

$$\tilde{L}_h(\theta) = \sum_{k \in V} s_k e^{i \theta k}, \quad i^2 = -1. \quad \text{(12)}$$

We consider standard coarsening. The low and high frequencies are given by

$$\theta \in T_L = \left[ -\frac{\pi}{2}, \frac{\pi}{2} \right] \cup \mathbb{Z}; \quad \theta \in T_H = \left[ -\frac{\pi}{2}, \frac{3\pi}{2} \right] \setminus T_L.$$

**Definition 2.** We define the LFA smoothing factor for relaxation error operator $S_h$ as

$$\mu_{\text{loc}}(S_h) = \max_{\theta \in T_H} \{ \rho(\tilde{S}_h(\theta)) \},$$

where $\rho(\tilde{S}_h(\theta))$ stands for the spectral radius of $\tilde{S}_h(\theta)$.

The symbol of $S_h$ defined in (10) is a $3 \times 3$ matrix since $K_h$ is a $3 \times 3$ block system; see (2). The same holds for $\mathcal{M}_h$, see (4), and the symbol of each block is a scalar. For more details on how to compute the symbol of $S_h$, refer to other studies.\textsuperscript{2,29} Since $\mu_{\text{loc}}(S_h)$ is a function of the parameter $\omega$, we are interested in minimizing $\mu_{\text{loc}}(S_h)$ over $\omega$ to obtain a fast convergence speed. We define the optimal smoothing factor as

$$\mu_{\text{opt}} = \min_{\omega} \mu_{\text{loc}}(S_h).$$

For the two-grid error operator $E_h$ defined in (11), the two-grid LFA convergence factor is

$$\rho_h(\nu) = \max_{\theta \in T_H} \{ \rho(\tilde{E}_h(\omega, \theta)) \},$$

where $\tilde{E}_h$ is the two-grid error operator symbol, $\rho(\tilde{E}_h)$ stands for the spectral radius of matrix $\tilde{E}_h$, and the integer $\nu$ is the sum of the number of pre- and post-smoothing steps. Since $E_h$ contains the coarse and fine grid operators, its symbol is a $12 \times 12$ matrix, including four harmonic frequencies.

From this point onward, let us drop the subscript $h$, unless it is necessary.

The element-wise Vanka-type smoother has been successfully applied to complex-shifted Laplacian systems arising from optimal control problem.\textsuperscript{50} Here, we consider an element-wise additive Vanka smoother applied to (6). The subproblem coefficient matrix $L_j$ in (7) has a symmetric structure

$$L_j = \frac{e^{2}}{h^2} \begin{pmatrix} 4 + r & -1 & -1 & 0 \\ -1 & 4 + r & 0 & -1 \\ -1 & 0 & 4 + r & -1 \\ 0 & -1 & -1 & 4 + r \end{pmatrix},$$

where $r = \frac{h^2}{e^2}$. 
It follows that

\[
L_j^{-1} = \frac{h^2}{c^2} \begin{pmatrix}
  a & b & b & c \\
  b & a & c & b \\
  b & c & a & b \\
  c & b & b & a
\end{pmatrix},
\]

(14)

where

\[
a = \frac{r^2 + 8r + 14}{(2 + r)(4 + r)(6 + d)},
\]

(15a)

\[
b = \frac{1}{(2 + r)(6 + r)},
\]

(15b)

\[
c = \frac{2}{(2 + r)(4 + r)(6 + r)}.
\]

(15c)

It is easy to show that \( a > b > c \), which is useful for our analysis.

Based on (7) and (14), the stencil of the element-wise Vanka smoother \( M_e \) is given by

\[
M_e = \frac{h^2}{4c^2} \begin{pmatrix}
  c & 2b & c \\
  2b & 4a & 2b \\
  c & 2b & c
\end{pmatrix}.
\]

Using (12), we have

\[
\tilde{L} = \frac{c^2}{h^2} (4 + r - 2 \cos \theta_1 - 2 \cos \theta_2),
\]

\[
\tilde{M}_e = \frac{h^2}{c^2} (a + b \cos \theta_1 + b \cos \theta_2 + c \cos \theta_1 \cos \theta_2).
\]

Let \( t = c^2(4 + r - 2 \cos \theta_1 - 2 \cos \theta_2) \) and \( \hat{t} = \frac{c^2}{a + b \cos \theta_1 + b \cos \theta_2 + c \cos \theta_1 \cos \theta_2} \). Then,

\[
\tilde{K} = \frac{1}{h^2} \begin{pmatrix}
  t & 0 & i2h \sin(\theta_1/2) \\
  0 & t & i2h \sin(\theta_2/2) \\
- i2h \sin(\theta_1/2) & -i2h \sin(\theta_2/2) & 0
\end{pmatrix},
\]

and

\[
\tilde{M} = \frac{1}{h^2} \begin{pmatrix}
  \hat{t} & 0 & i2h \sin(\theta_1/2) \\
  0 & \hat{t} & i2h \sin(\theta_2/2) \\
- i2h \sin(\theta_1/2) & -i2h \sin(\theta_2/2) & 0
\end{pmatrix}.
\]

To identify the eigenvalues of \( \tilde{M}^{-1} \tilde{K} \), we first compute the determinant of \( \tilde{K} - \lambda \tilde{M} \):

\[
| \tilde{K} - \lambda \tilde{M} | = \frac{1}{h^2} \begin{vmatrix}
  t - \hat{t} & 0 & (1 - \lambda)i2h \sin(\theta_1/2) \\
  0 & t - \hat{t} & (1 - \lambda)i2h \sin(\theta_2/2) \\
- (1 - \lambda)i2h \sin(\theta_1/2) & -(1 - \lambda)i2h \sin(\theta_2/2) & 0
\end{vmatrix}
\]

\[
= \frac{1}{h^2} (t - \hat{t})(1 - \lambda)^2 \left( (i2h \sin(\theta_1/2))^2 + (i2h \sin(\theta_2/2))^2 \right)
\]

\[
= 4\hat{t} \left( (\sin(\theta_1/2))^2 + (\sin(\theta_2/2))^2 \right) (1 - \lambda)^2(\lambda - t/\hat{t}).
\]
The three eigenvalues of $\tilde{\mathcal{M}}^{-1}\tilde{\mathcal{K}}$ are $1, 1$ and $\frac{1}{4}$, where

$$
\lambda^*(r; \cos \theta_1, \cos \theta_2) = (a + b \cos \theta_1 + b \cos \theta_2 + c \cos \theta_1 \cos \theta_2)(4 + r - 2 \cos \theta_1 - 2 \cos \theta_2).
$$

(16)

For $\theta \in T^2$, it is easy to show that

$$(\cos \theta_1, \cos \theta_2) \in D = [-1, 1] \times [-1, 0] \bigcup [-1, 0] \times [0, 1].
$$

(17)

Next, we explore the range of $\lambda^*$ over $\theta$ for high frequencies.

**Theorem 1.** For $\theta \in T^2$,

$$
\max_\theta \lambda^*(r; \cos \theta_1, \cos \theta_2) = \lambda^*(r; -1, -1) = (a - 2b + c)(8 + r) =: d_1(r).
$$

$$
\min_\theta \lambda^*(r; \cos \theta_1, \cos \theta_2) = \lambda^*(r; 1, 0) = (a + b)(2 + r) =: d_2(r).
$$

Proof. For simplicity, let $\eta_1 = \cos \theta_1$ and $\eta_2 = \cos \theta_2$. Then, we rewrite (16) as

$$
\lambda^* = \psi(\eta_1, \eta_2) = (a + b\eta_1 + b\eta_2 + c\eta_1\eta_2)(4 + r - 2\eta_1 - 2\eta_2).
$$

We first consider the critical point of $\psi(\eta_1, \eta_2)$ in $D$, see (17), by computing the partial derivatives of $\psi(\eta_1, \eta_2)$, which are given by

$$
\psi_\eta'(\eta_1, \eta_2) = rb + 4b - 2a - 4b\eta_1 + (4c + cr - 4b)\eta_2 - 2c\eta_2^2 - 4c\eta_1\eta_2 = 0,
$$

(18)

$$
\psi_\eta''(\eta_1, \eta_2) = rb + 4b - 2a - 4b\eta_1 + (4c + cr - 4b)\eta_1 - 2c\eta_1^2 - 4c\eta_1\eta_2 = 0.
$$

(19)

Subtracting (19) from (18) gives

$$
(\eta_1 - \eta_2)(2(\eta_1 + \eta_2) - 4 - r) = 0.
$$

(20)

It follows that $\eta_1 = \eta_2$ or $2(\eta_1 + \eta_2) - 4 - r = 0$. However, $\eta_1 + \eta_2 < 2$, so the latter does not have a real solution. For $\eta_1 = \eta_2$, we replace $\eta_2$ by $\eta_1$ in (18), leading to

$$
6c\eta_1^2 - (4c + cr - 8b)\eta_1 - (rb + 4b - 2a) = 0.
$$

(21)

We claim that there is no real solution for (21) for $r > 0$. We will show that the discriminant is not positive. We first simplify $rb + 4b - 2a$. Using (15a) and (15b) gives

$$
rb + 4b - 2a = \frac{4 + r}{(2 + r)(6 + r)} - \frac{2(r^2 + 8r + 14)}{(2 + r)(4 + r)(6 + r)}
$$

$$
= \frac{(4 + r)^2 - 2(r^2 + 8r + 14)}{(2 + r)(4 + r)(6 + r)}
$$

$$
= -\frac{1}{4 + r}.
$$

(22)

Using (15b) and (15c), the discriminant of (21) is

$$
\Phi = (4c + cr - 8b)^2 + 4 \cdot 6c(rb + 4b - 2a)
$$

$$
= \left(\frac{8 + 2r}{(2 + r)(4 + r)(6 + r)} - \frac{8(4 + r)}{(2 + r)(4 + r)(6 + r)}\right)^2 - \frac{48}{(2 + r)(4 + r)(6 + r)} \frac{1}{4 + r}
$$

$$
= \left(\frac{-6}{(2 + r)(6 + r)}\right)^2 - \frac{48}{(2 + r)(4 + r)^2(6 + r)}
$$

$$
= \frac{12}{(2 + r)(6 + r)} \left(\frac{3}{(2 + r)(6 + r)} - \frac{4}{(4 + r)^2}\right)
$$

$$
= -\frac{12r(r + 8)}{(2 + r)^2(4 + r)^2(6 + r)^2} \leq 0.
$$

(23)
The case \( r = 0 \) has been discussed,\(^6\) where \( \psi'_{\eta_1}(\eta_1, \eta_2) = \psi'_{\eta_2}(\eta_1, \eta_2) = 0 \) gives \((\eta_1, \eta_2) = (-1, -1)\) the boundary point of \( D \), and \( \lambda_{\max}^* = \frac{4}{0^2} \). When \( r > 0 \), (20) has no real solution and \( \psi(\eta_1, \eta_2) \) cannot have extreme values at interior of \( D \). This means that we only need to find the extreme values of \( \psi(\eta_1, \eta_2) \) at the boundary of \( D \), see (17). To do this, we split the boundary of \( D \) as follows:

\[
\begin{align*}
\partial D_1 &= \{-1\} \times [-1, 1], \\
\partial D_2 &= [-1, 1] \times \{ -1 \}, \\
\partial D_3 &= \{1\} \times [-1, 0], \\
\partial D_4 &= [0, 1] \times \{ 0 \}, \\
\partial D_5 &= \{ 0 \} \times [0, 1], \\
\partial D_6 &= [-1, 0] \times \{ 1 \}.
\end{align*}
\]

Due to the symmetry of \( \psi(\eta_1, \eta_2) \), that is \( \psi(\eta_1, \eta_2) = \psi(\eta_2, \eta_1) \), we only need to find the extreme values of \( \psi(\eta_1, \eta_2) \) at \( \partial D_1, \partial D_3, \) and \( \partial D_4 \). We present below the results.

1. For \((\eta_1, \eta_2) \in \partial D_1\),

\[
\psi(\eta_1, \eta_2) = \psi(-1, \eta_2) = (a - b + b\eta_2 - c\eta_2)(6 + r - 2\eta_2).
\]  

(22)

Note that the two roots of the quadratic form (22) are \(\frac{6 + r}{2} \) and \(\frac{a - b}{c - b} \). Using (15), we have

\[
\frac{a - b}{c - b} = - (5 + r).
\]

Thus, the axis of symmetry is \(\eta_2 = \frac{(6 + r)/2 - (5 + r)}{2} = -1 - \frac{r}{4} \leq -1 \). Using the fact that \( a > b > c \), see (15a)–(15c), the quadratic function opens downward. Therefore, the maximum and minimum of \( \psi(-1, \eta_2) \) for \( \eta_2 \in [-1, 1] \) are

\[
\begin{align*}
\psi(-1, \eta_2)_{\text{max}} &= \psi(-1, -1) = (a - 2b + c)(8 + r), \\
\psi(-1, \eta_2)_{\text{min}} &= \psi(-1, 1) = (a - c)(4 + r).
\end{align*}
\]  

(23)

2. For \((\eta_1, \eta_2) \in \partial D_3\),

\[
\psi(\eta_1, \eta_2) = \psi(1, \eta_2) = (a + b + b\eta_2 + c\eta_2)(2 + r - 2\eta_2).
\]  

(24)

The two roots of quadratic form (24) are \(\frac{2 + r}{2} \) and \(\frac{-a + b}{b + c} \). Using (15), we have

\[
\frac{-a + b}{b + c} = -(3 + r).
\]

Thus, the axis of symmetry is \(\eta_2 = \frac{(2 + r)/2 - (3 + r)}{2} = -1 - \frac{r}{4} \leq -1 \). Using the fact that \( a > b > c \), the quadratic function opens downward. It follows that for \( \eta_2 \in [-1, 0] \), the maximum and minimum of \( \psi(1, \eta_2) \) are given by

\[
\begin{align*}
\psi(1, \eta_2)_{\text{max}} &= \psi(1, -1) = (a - c)(4 + r), \\
\psi(1, \eta_2)_{\text{min}} &= \psi(1, 0) = (a + b)(2 + r).
\end{align*}
\]

3. For \((\eta_1, \eta_2) \in \partial D_4\),

\[
\psi(\eta_1, \eta_2) = \psi(\eta_1, 0) = (a + b\eta_1)(4 + r - 2\eta_1).
\]

Note that the two roots of quadratic form (24) are \(\frac{4 + r}{2} \) and \(\frac{-a}{b} \). Using (15a) and (15b), we have

\[
\frac{-a}{b} = -(4 + r) + \frac{2}{4 + r}.
\]
Thus, the axis of symmetry is \( \eta_1 = \frac{(4r + 2 - (4r) + \frac{1}{r})}{2} = -1 - \frac{r}{4} + \frac{1}{4r} < 0 \). Thus, the maximum and minimum of \( \psi(\eta_1, 0) \) for \( \eta_1 \in [0, 1] \) are

\[
\psi(\eta_1, 0)_{\text{max}} = \psi(0, 0) = a(4 + r), \\
\psi(\eta_1, 0)_{\text{min}} = \psi(0, 1) = (a + b)(2 + r).
\]  

(25)

Based on the above discussions, the minimum of \( \psi(\eta_1, \eta_2) \) over \( \partial D \) is

\[
\psi(\eta_1, \eta_2)_{\text{min}} = \psi(1, 0) = \psi(0, 1) = (a + b)(2 + r).
\]

Next, we compare \( \psi(-1, -1) \) (see (23)) and \( \psi(0, 0) \) (see (25)) to determine the maximum. Using (15), we have

\[
\psi(-1, -1) - \psi(0, 0) = (a - 2b + c)(8 + r) - a(4 + r)
= 4a + (c - 2b)(8 + r)
= \frac{4(r^2 + 8r + 14)}{(2 + r)(4 + r)(6 + r)} - \frac{2 - 2(4 + r)}{(2 + r)(4 + r)(6 + r)}(8 + r)
= \frac{2r^2 + 10r + 8}{(2 + r)(4 + r)(6 + r)} > 0.
\]

It follows that the maximum of \( \psi(\eta_1, \eta_2) \) is given by

\[
\psi(\eta_1, \eta_2)_{\text{max}} = \psi(-1, -1) = (a - 2b + c)(8 + r).
\]

Thus, for \( (\eta_1, \eta_2) \in D \), the maximum and minimum of \( \psi(\eta_1, \eta_2) \) are \( \psi(-1, -1) \) and \( \psi(1, 0) = \psi(0, 1) \), respectively.

Based on the results in Theorem 1, we can further estimate the range of extreme values of \( \lambda^* \), which plays an important role in determining the optimal smoothing factor for V-BSR.

**Theorem 2.** Suppose \( r \in [0, \infty) \). Then,

\[
d_1(r) = \frac{8 + r}{6 + r}, \\
d_2(r) = \frac{3 + r}{4 + r}.
\]

Furthermore,

\[
1 < d_1(r) \leq \frac{4}{3}, \\
\frac{3}{4} \leq d_2(r) < 1.
\]

**Proof.** Using (15), we simplify \( d_1(r) \) as follows:

\[
d_1(r) = (a - 2b + c)(8 + r)
= \frac{r^2 + 8r + 14 - 2(4 + r) + 2}{(2 + r)(4 + r)(6 + r)}(8 + r)
= \frac{8 + r}{6 + r}.
\]

Since \( d_1(r) \) is a decreasing function of \( r \), \( \max d_1(r) = d_1(0) = \frac{4}{5} \).

Using (15a) and (15b), we have

\[
d_2(r) = (a + b)(2 + r)
= \frac{r^2 + 8r + 14 + 4 + r}{(2 + r)(4 + r)(6 + r)}(2 + r)
= \frac{r + 3}{r + 4}.
\]

Since \( d_2(r) \) is an increasing function of \( r \), \( \min d_2(r) = d_2(0) = \frac{3}{4} \).
Now, we are able to derive the optimal smoothing factor for V-BSR for the Stokes–Darcy Brinkman problems.

**Theorem 3.** For the V-BSR relaxation scheme (8) the optimal smoothing factor is given by

\[
\mu_{\text{opt}}(r) = \min_{\omega} \max_{\theta \in \mathbb{T}^2\Gamma} [1 - \omega, 1 - \omega \lambda^*] = \frac{3r + 14}{2r^2 + 21r + 50}.
\] (27)

provided that

\[
\omega = \omega_{\text{opt}} = \frac{2r^2 + 20r + 48}{2r^2 + 21r + 50}.
\] (28)

Moreover,

\[
\mu_{\text{opt}}(r) \leq \frac{7}{25} = 0.28.
\]

**Proof.** From Theorem 1 and (26), we know that

\[
\max_{\theta \in \mathbb{T}^2\Gamma} [1 - \omega, 1 - \omega \lambda^*] = \max_{\theta \in \mathbb{T}^2\Gamma} [1 - \omega d_2(r), 1 - \omega d_1(r)].
\] (29)

To minimize \(\max_{\theta \in \mathbb{T}^2\Gamma} [1 - \omega \lambda^*]\), we require

\[
|1 - \omega d_2(r)| = |1 - \omega d_1(r)|,
\]

which gives \(\omega_{\text{opt}}(r) = \frac{2}{d_1(r) + d_2(r)}\). Using Theorem 2, we obtain

\[
\omega_{\text{opt}}(r) = \frac{2}{d_1(r) + d_2(r)} = \frac{2r^2 + 20r + 48}{2r^2 + 21r + 50}
\]

and

\[
\mu_{\text{opt}}(r) = \frac{d_1(r) - d_2(r)}{d_1(r) + d_2(r)} = \frac{3r + 14}{2r^2 + 21r + 50} \leq \frac{7}{25}.
\]

**Remark 1.** When \(r = 0\), Theorem 3 is consistent with the existing results, which amount to applying an element-wise Vanka smoother to the Poisson equation.

**Proposition 1.** \(\mu_{\text{opt}}(r)\) given by (27) is a decreasing function of \(r\).

**Proof.** The derivative of \(\mu_{\text{opt}}\) is given by

\[
\mu'_{\text{opt}} = \frac{-2(r^2 + 28r + 72)}{(2r^2 + 21r + 50)^2} < 0.
\]

This suggests that when \(r\) increases, the optimal smoothing factor decreases.

Let us look at the optimal parameter, (28). It can be shown that

\[
\omega'_{\text{opt}}(r) = \frac{2(r^2 + 4r - 4)}{(2r^2 + 21r + 50)^2}.
\]

When \(r \in [0, 2\sqrt{2} - 2]\), \(\omega_{\text{opt}}(r)\) is decreasing and when \(r \in [2\sqrt{2} - 2, \infty)\), \(\omega_{\text{opt}}(r)\) is increasing. It follows that

\[
(\omega_{\text{opt}}(r))_{\min} = \omega_{\text{opt}}(2\sqrt{2} - 2) = \left(\sqrt{2} - 1\right)\left(4\sqrt{2} + 16\right) + 24 \approx 0.959 \leq \omega_{\text{opt}}(r) < 1.
\] (30)
Thus, for simplicity, if we take \( \omega = 1 \), then (29) gives

\[
\mu(\omega = 1) = \max_{\omega=1} \{ |1 - \omega d_2(r)|, |1 - \omega d_1(r)| \} = \frac{2}{6 + r} \leq \frac{1}{3}. \tag{31}
\]

In practice, we can consider \( \omega = 1 \). In multigrid, for fixed \( \epsilon \), in each level, the relaxation schemes has a different convergence speed in each level, which can be computed from (31). However, note that (31) is a decreasing function of \( r = \frac{h^2}{\epsilon} \) or \( h \). This means that at the coarse level, the relaxation scheme has a smaller convergence speed compared with that of the fine level.

4 | NUMERICAL EXPERIMENTS

In this section, we first compute LFA two-grid convergence factors using two choices of the damping parameter, that is, \( \omega = 1 \) and \( \omega = \omega_{\text{opt}} \) defined in (28), then we report V-cycle multigrid results for different values of the physical parameter \( \epsilon \).

4.1 | LFA prediction

We compute the two-grid LFA convergence factor (13), using \( h = 1/64 \). \( \omega = 1 \) and \( \omega = \omega_{\text{opt}} \), see (28), derived from optimizing LFA smoothing factors for different \( \epsilon \). We are more interested in small values of \( \epsilon \). Here, we follow the work of Sun et al.\textsuperscript{15} to take \( \epsilon = 1, 2^{-2}, 2^{-4}, 2^{-6}, 2^{-8} \). From Table 1 we see a strong agreement between two-grid convergence factors \( \rho_h(1) \) and the LFA smoothing factors. Moreover, the convergence factors for optimal \( \omega \) are slightly better than those for \( \omega = 1 \), which is reasonable since the optimal \( \omega \), see (30), is very close to 1. From our smoothing analysis, we know that even though the smoothing factor is dependent on \( h \) and \( \alpha \), the upper bound on the smoothing factor is \( \frac{1}{3} \). This is also confirmed by our two-grid LFA convergence factor \( \rho_h(1) \) in Table 1.

To illustrate how the smoothing factor changes as a function of \( r = \frac{h^2}{\epsilon} \), we plot \( \mu_{\text{opt}} \) defined in (27) and \( \mu(\omega = 1) \) in (31) as functions of \( r \) in Figure 2. It is evident that when \( r \) increases, the smoothing factor decreases and approaches zero, and \( \mu(\omega = 1) \) tends towards \( \mu_{\text{opt}} \).

4.2 | Multigrid performance

Consider the model problems (1) on the unit square domain \([0, 1] \times [0, 1]\) with an exact solution,\textsuperscript{15} and given by

\[
\begin{align*}
\delta u(x, y) &= \pi \sin^2(\pi x) \sin(2\pi y), \\
\psi(x, y) &= -\pi \sin(2\pi x) \sin^2(\pi y), \\
\rho(x, y) &= \sin(\pi y) - \frac{2}{\pi},
\end{align*}
\]

| \( \epsilon, \omega = 1 \) | \( \mu_{\text{opt}} \) | \( \rho_h(1) \) | \( \rho_h(2) \) | \( \rho_h(3) \) | \( \rho_h(4) \) |
|---|---|---|---|---|---|
| 1 | 0.333 | 0.333 | 0.119 | 0.054 | 0.043 |
| 2^{-2} | 0.333 | 0.333 | 0.119 | 0.054 | 0.042 |
| 2^{-4} | 0.330 | 0.330 | 0.115 | 0.052 | 0.040 |
| 2^{-6} | 0.286 | 0.286 | 0.082 | 0.023 | 0.012 |
| 2^{-8} | 0.091 | 0.091 | 0.008 | 0.001 | 0.000 |
| 1, \omega_{\text{opt}} | 0.280 | 0.280 | 0.096 | 0.056 | 0.044 |
| 2^{-2} | 0.280 | 0.280 | 0.096 | 0.056 | 0.044 |
| 2^{-4} | 0.276 | 0.276 | 0.093 | 0.055 | 0.042 |
| 2^{-6} | 0.233 | 0.233 | 0.057 | 0.026 | 0.014 |
| 2^{-8} | 0.069 | 0.069 | 0.005 | 0.000 | 0.000 |
FIGURE 2  Smoothing factors with optimal $\omega$ and $\omega = 1$ as functions of $r = \frac{h^2}{\epsilon^2}$.

with $g = 0$. The source term is computed via $f = (f_1, f_2) = -\epsilon^2 \Delta u + \nabla p$, and it is

$$
\begin{align*}
f_1 &= (4\pi^3 \epsilon^2 + \pi) \sin^2(\pi x) \sin(2\pi y) - 2\pi^3 \epsilon^2 \cos(2\pi x) \sin(2\pi y), \\
f_2 &= -(4\pi^3 \epsilon^3 + \pi) \sin(2\pi x) \sin^2(\pi y) + 2\pi^3 \epsilon^2 \sin(2\pi x) \cos(2\pi y) + \pi \cos(\pi y).
\end{align*}
$$

To validate our theoretical LFA predictions, we compute the actual multigrid convergence factors by

$$
\hat{\rho}_h^{(k)} = \left( \frac{||r_k||}{||r_0||} \right)^{1/k},
$$

where $r_k = b_h - K_h z_k$ is the residual and $z_k$ is the $k$th multigrid iteration. The initial guess is chosen randomly. In our test, we report $\hat{\rho}_h^{(k)} = \hat{\rho}_h$ with the smallest $k$ such that $||r_k||/||r_0|| \leq 10^{-10}$.

As mentioned before, computing the exact solution of the Schur complement system (9) is expensive. For our multigrid tests, we apply a few weighted ($\omega J$) Jacobi iterations to the Schur complement system. We choose $\omega J = 0.8$ which seems more robust to $\epsilon$. The number of Jacobi iterations is set as three.

4.2.1 Two-grid results

We first report actual two-grid convergence factor using exact solve (Matlab’s backslash) for the Schur complement with different parameter $\omega$ and different total number of smoothing steps $\nu$, see Table 2. We see the actual two-grid convergence factor is slightly better than the LFA predictions reported in Table 1. This tiny gap can be ignored.

Next, we are more interested in exploring the actual two-grid convergence factor using inexact solve for the Schur complement with different parameter $\omega$ and different total number of smoothing steps $\nu$, and comparing the results with those in Table 2 where the Schur complement system is solved exactly. Table 3 shows the two-grid performance using three Jacobi iterations for the Schur complement system. We see that for $\epsilon = 1, 2^{-2}, 2^{-4}$, the inexact solve can achieve almost the same performance as that of exact solve. While for $\epsilon = 2^{-6}, 2^{-8}$, the exact solve for the Schur complement system has a better performance than that of inexact solve. This might suggest that in practice more iterations are needed for the inexact solve for the Schur complement system. However, due to the satisfactory convergence factor of the actual performance, we do not further explore this. Table 3 also indicates that for small $\epsilon$ or large $\nu$, using $\omega = 1$ and $\omega = \omega_{opt}$ leads to similar performance.
TABLE 2 Two-grid measured convergence factor, $\hat{\omega}(\nu)$, for $\omega = 1$ and $\omega = \omega_{\text{opt}}$ (see (28)), using exact solve for the Schur complement and $h = 1/64$.  

| $\epsilon$ | $\omega = 1$ | $\omega = \omega_{\text{opt}}$ | $\hat{\omega}(1)$ | $\hat{\omega}(2)$ | $\hat{\omega}(3)$ | $\hat{\omega}(4)$ | $\hat{\omega}(1)$ | $\hat{\omega}(2)$ | $\hat{\omega}(3)$ | $\hat{\omega}(4)$ |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 0.319 | 0.111 | 0.034 | 0.020 | 0.265 | 0.086 | 0.028 | 0.020 |
| $2^{-2}$ | 0.316 | 0.109 | 0.034 | 0.020 | 0.266 | 0.084 | 0.028 | 0.020 |
| $2^{-4}$ | 0.300 | 0.095 | 0.028 | 0.017 | 0.247 | 0.069 | 0.025 | 0.017 |
| $2^{-6}$ | 0.209 | 0.045 | 0.009 | 0.003 | 0.162 | 0.028 | 0.007 | 0.003 |
| $2^{-8}$ | 0.035 | 0.002 | 0.000 | 0.000 | 0.027 | 0.000 | 0.000 | 0.000 |

TABLE 3 Two-grid measured convergence factor, $\hat{\omega}(\nu)$, for $\omega = 1$ and $\omega = \omega_{\text{opt}}$ (see (28)), using three Jacobi iterations for solving the Schur complement and $h = 1/64$.  

| $\epsilon$ | $\omega = 1$ | $\omega = \omega_{\text{opt}}$ | $\hat{\omega}(1)$ | $\hat{\omega}(2)$ | $\hat{\omega}(3)$ | $\hat{\omega}(4)$ | $\hat{\omega}(1)$ | $\hat{\omega}(2)$ | $\hat{\omega}(3)$ | $\hat{\omega}(4)$ |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 0.319 | 0.111 | 0.033 | 0.023 | 0.266 | 0.082 | 0.030 | 0.023 |
| $2^{-2}$ | 0.317 | 0.109 | 0.033 | 0.023 | 0.264 | 0.080 | 0.030 | 0.024 |
| $2^{-4}$ | 0.300 | 0.094 | 0.029 | 0.021 | 0.248 | 0.068 | 0.029 | 0.023 |
| $2^{-6}$ | 0.209 | 0.047 | 0.023 | 0.015 | 0.163 | 0.044 | 0.024 | 0.016 |
| $2^{-8}$ | 0.145 | 0.035 | 0.020 | 0.015 | 0.165 | 0.040 | 0.019 | 0.015 |

4.2.2 V(1,1)-cycle results

A two-grid method is computationally costly since we have to solve the coarse problem directly and if the initial mesh is fine, then the next coarser mesh may give rise to a large problem as well. In practice, deeply-nested W-cycle and V-cycle are preferred. We now explore the V(1,1)-cycle (meaning V-cycle with one pre- and one post-smoothing) multigrid methods with two choices of $\omega$ ($\omega = 1$ and $\omega = \omega_{\text{opt}}$ defined in (28)), and vary values of the physical parameter $\epsilon$ (viscosity, see (1a)). In order to study the sensitivity of solving the Schur complement system, we consider one, two, and three Jacobi iterations for Schur complement system. We consider different $n \times n$ finest meshgrids, where $n = 32, 64, 128, 256$, and the coarsest grid has four cells in each direction.

**Exact solve for Schur complement system:** To have a clear idea what the V-cycle performance is with exact solve for the Schur complement system, we report the iteration accounts in Table 4, where Matlab's backslash is used for solving the Schur complement system. We see that for small $\epsilon$, it takes less iterations to achieve the stopping criterion. Using the optimal $\omega$ only saves one iteration compared with the case using $\omega = 1$. This is consistent with our analysis, see Figure 2 and Table 1.

**One iteration for Schur complement system:** We first report the iteration counts for V(1,1)-cycle multigrid methods using one iteration of Jacobi relaxation for the Schur complement system in Table 5 to achieve the tolerance $\|r_h\|/\|r_0\| \leq 10^{-10}$. We see that using $\omega = 1$ and optimal $\omega$ give similar performance, which is similar to these results shown in Table 4 where exact solver is used for the Schur complement system. When $\epsilon = 2^{-6}, 2^{-8}$, the iteration count increase dramatically. To mitigate the effect of this degradation, we will consider two or three Jacobi iterations for the Schur complement system.

**Two iterations for Schur complement system:** We report the convergence history of the relative residual norm $\|r_h\|/\|r_0\|$ as a function of the number of V(1,1)-cycles using two Jacobi iterations for Schur complement system. Figure 3 reports the results for $\epsilon = 1$. We see that using optimal $\omega$ takes 12 V(1,1)-cycle iterations to achieve the stopping tolerance and it takes 13 iterations for $\omega = 1$. The convergence behavior is independent of meshsize $h$. A similar performance is seen for $\epsilon = 2^{-2}, 2^{-4}, 2^{-6}, 2^{-8}$, see Figures A1–A4 in the Appendix. For smaller values of $\epsilon$, the iteration count does not increase, which is around 12 iterations. We omit these results here. Using optimal $\omega$ has one iteration number fewer than that of $\omega = 1$. Thus, it is simple and reasonable to use $\omega = 1$ in practice.
**TABLE 4** Iteration accounts for V(1,1)-cycle multigrid with exact solve for the Schur complement system.

| $\epsilon, \omega = 1$ | $n = 32$ | $n = 64$ | $n = 128$ | $n = 256$ |
|-----------------------|---------|---------|---------|---------|
| 1                     | 12      | 12      | 13      | 13      |
| $2^{-2}$              | 12      | 12      | 12      | 13      |
| $2^{-4}$              | 10      | 10      | 11      | 12      |
| $2^{-6}$              | 6       | 8       | 9       | 10      |
| $2^{-8}$              | 3       | 4       | 6       | 7       |

| $\epsilon, \omega_{\text{opt}}$ | $n = 32$ | $n = 64$ | $n = 128$ | $n = 256$ |
|-------------------------------|---------|---------|---------|---------|
| 1                             | 11      | 11      | 11      | 12      |
| $2^{-2}$                      | 11      | 11      | 11      | 12      |
| $2^{-4}$                      | 9       | 10      | 10      | 11      |
| $2^{-6}$                      | 5       | 7       | 8       | 9       |
| $2^{-8}$                      | 3       | 4       | 5       | 6       |

**TABLE 5** Iteration accounts for V(1,1)-cycle multigrid with one Jacobi iteration for solving the Schur complement system.

| $\epsilon, \omega = 1$ | $n = 32$ | $n = 64$ | $n = 128$ | $n = 256$ |
|-----------------------|---------|---------|---------|---------|
| 1                     | 13      | 13      | 13      | 15      |
| $2^{-2}$              | 12      | 13      | 13      | 14      |
| $2^{-4}$              | 11      | 11      | 12      | 12      |
| $2^{-6}$              | 23      | 18      | 13      | 11      |
| $2^{-8}$              | 50      | 50      | 47      | 34      |

| $\epsilon, \omega_{\text{opt}}$ | $n = 32$ | $n = 64$ | $n = 128$ | $n = 256$ |
|-------------------------------|---------|---------|---------|---------|
| 1                             | 12      | 12      | 12      | 15      |
| $2^{-2}$                      | 12      | 12      | 12      | 14      |
| $2^{-4}$                      | 11      | 11      | 11      | 11      |
| $2^{-6}$                      | 26      | 19      | 14      | 12      |
| $2^{-8}$                      | 50      | 50      | 50      | 38      |

**Three iterations for Schur complement system:** We explore V(1,1)-cycle iterations with two choices of $\omega$ and a varying physical parameter $\epsilon$, using three Jacobi iterations for Schur complement system. We report the history of relative residual $||r_k||/||r_0||$ as a function of the V(1,1)-cycle iteration counts for $n \times n$ meshgrid ($n = 32, 64, 128, 256$). Figure 4 shows the results for $\epsilon = 1$. We see that using optimal $\omega$ takes 12 iterations of V(1,1)-cycle to achieve the stopping tolerance and it takes 13 iterations for $\omega = 1$. We see that the convergence behavior is independent of meshsize $h$. For $\epsilon = 2^{-2}, 2^{-4}, 2^{-6}, 2^{-8}$, we observe similar performance shown in Figures A5–A8 in the Appendix. Compared with two Jacobi iterations for solving the Schur complement system, three Jacobi iterations give a slightly better results for small $\epsilon = 2^{-6}, 2^{-8}$, which needs 7 and 8 iterations, respectively. Again using optimal $\omega$ has one iteration number less than that of $\omega = 1$. Thus, it is simple and reasonable to use $\omega = 1$ in practice.

We summarize that for $\epsilon = 1, 2^{-2}, 2^{-4}$, V(1,1)-cycle multigrid using two or three Jacobi iterations for the Schur complement system takes almost the same number of multigrid iterations to achieve the same performance as that of a direct solver for the Schur complement system shown in Table 4. For $\epsilon = 2^{-6}, 2^{-8}$, V(1,1)-cycle multigrid with two or three Jacobi iterations for the Schur complement system needs around 10 iterations, which is slightly more than these shown in Table 4. However, two or three Jacobi iterations are enough to achieve robustness V(1,1)-cycle multigrid with respect to meshgrid and physical parameter.
5 | CONCLUSIONS

We consider a parameter-robust multigrid method for solving the discrete system of Stokes–Darcy problems, with the marker and cell scheme used for the discretization. The resulting linear system is a saddle-point system. In contrast to existing Vanka smoothers, where the velocities and pressure unknowns in a grid cell are updated simultaneously, we propose Vanka-based Braess–Sarazin relaxation scheme, where the Laplace-like term in the saddle-point system is solved by an additive Vanka algorithm. This approach decouples the velocities and pressure unknowns. Moreover, only matrix-vector products are needed in our proposed multigrid method. LFA is used to analyze the smoothing process and help choose the optimal parameter that minimizing LFA smoothing factor. From LFA, we derive the stencil of additive Vanka for the Laplace-like operator, which can help form the global iteration matrix, avoiding solving many subproblems in the classical additive Vanka setting.
Our main contribution is that we derive the optimal algorithmic parameter and optimal LFA smoothing factor for Vanka-based Braess–Sarazin relaxation scheme, and show that this scheme is highly efficient with respect to physical parameter. Our theoretical results reveal that although the optimal damping parameter is related to physical parameter and meshsize, it is very close to one. We also present the theoretical LFA smoothing factor with damping parameter one. In Vanka-based Braess–Sarazin relaxation, we have to solve a Schur complement system. Direct solver is often expensive. We propose an inexact version of Vanka-based Braess–Sarazin relaxation, where we apply only two or three iterations of Jacobi to the Schur complement system to achieve the same performance as that of an exact solve. We show that using damping parameter one can achieve almost the same performance as that of optimal result, and the results are close to exact version. Thus, using damping parameter one is recommended. Our V-cycle multigrid illustrates high efficiency of our relaxation scheme and robustness to physical parameter.

We comment that the proposed Vanka-based Braess–Sarazin multigrid method can be used as a preconditioner for Krylov subspace methods. We have limited ourselves for the MAC scheme on uniform grids. It will be interesting in exploring its extension to non-uniform grids in future.

ACKNOWLEDGMENTS
The author is grateful to the anonymous referee for the helpful comments. This study was funded by The University of British Columbia.

CONFLICT OF INTEREST STATEMENT
This study does not have any conflicts to disclose.

DATA AVAILABILITY STATEMENT
Data sharing is not applicable to this article as no new data were created or analyzed in this study.

REFERENCES
1. Bathe KJ, Ledeza GA. Benchmark problems for incompressible fluid flows with structural interactions. Comput Struct. 2007;85(11-14):628–44.
2. Brandt A, Dinar N. Multigrid solutions to elliptic flow problems. In: Parter SV, editor. Numerical methods for partial differential equations. Amsterdam: Elsevier; 1979. p. 53–147.
3. Brandt A, Livne OE. Multigrid techniques: 1984 guide with applications to fluid dynamics. Revised ed. Philadelphia, PA: SIAM; 2011.
4. Connor JJ, Brebbia CA. Finite element techniques for fluid flow. Boston, MA: Newnes; 2013.
5. Vanka SP. Block-implicit multigrid solution of Navier-Stokes equations in primitive variables. J Comput Phys. 1986;65(1):138–58.
6. Wesseling P, Oosterlee CW. Geometric multigrid with applications to computational fluid dynamics. J Comput Appl Math. 2001;128(1-2):311–34.
7. Gulbransen AF, Hauge VL, Lie KA. A multiscale mixed finite-element method for vuggy and naturally fractured reservoirs. SPE J. 2010;15(02):395–403.
8. Vassilevski PS, Villa U. A mixed formulation for the Brinkman problem. SIAM J Numer Anal. 2014;52(1):258–81.
9. Xie X, Xu J, Xue G. Uniformly-stable finite element methods for Darcy-Stokes-Brinkman models. J Comput Math. 2008;26:437–55.
10. Xu X, Zhang S. A new divergence-free interpolation operator with applications to the Darcy–Stokes–Brinkman equations. SIAM J Sci Comput. 2010;32(2):855–74.
11. Zhai Q, Zhang R, Mu L. A new weak Galerkin finite element scheme for the Brinkman model. Commun Comput Phys. 2016;19(5):1409–34.
12. Zhang S, Xie X, Chen Y. Low order nonconforming rectangular finite element methods for Darcy-Stokes problems. J Comput Math. 2009;27:400–24.
13. Zhao L, Chung E, Lam MF. A new staggered DG method for the Brinkman problem robust in the Darcy and Stokes limits. Comput Methods Appl Mech Eng. 2020;364:112986.
14. He J. Finite difference simulation of the Stokes-Brinkman equation for transient flow in naturally fractured carbonate karst reservoirs [PhD thesis]; 2017.
15. Sun Y, Rui H. Stability and convergence of the mark and cell finite difference scheme for Darcy-Stokes-Brinkman equations on non-uniform grids. Numer Methods Partial Differ Equ. 2019;35(2):509–27.
16. Evans JA, Hughes TJ. Isogeometric divergence-conforming B-splines for the Darcy–Stokes–Brinkman equations. Math Models Methods Appl Sci. 2013;23(04):671–741.
17. Arrarás A, Gaspar FJ, Portero L, Rodrigo C. Multigrid solvers for multipoint flux approximations of the Darcy problem on rough quadrilateral grids. Comput Geosci. 2021;25(2):715–30.
18. Harder C, Paredes D, Valentin F. A family of multiscale hybrid-mixed finite element methods for the Darcy equation with rough coefficients. J Comput Phys. 2013;245:107–30.
19. Hannukainen A, Juntunen M, Stenberg R. Computations with finite element methods for the Brinkman problem. Comput Geosci. 2011;15(1):155–66.
20. Mardal KA, Tai XC, Winther R. A robust finite element method for Darcy–Stokes flow. SIAM J Numer Anal. 2002;40(5):1605–31.
21. Braess D, Sarazin R. An efficient smoother for the Stokes problem. Appl Numer Math. 1997;23(1):3–19.
22. He Y, MacLachlan SP. Local Fourier analysis of block-structured multigrid relaxation schemes for the Stokes equations. Numer Linear Algebra Appl. 2018;25(3):e2147.
23. Zulehner W. A class of smoothers for saddle point problems. Computing. 2000;65(3):227–46.
24. Chen L, Hu X, Wang M, Xu J. A multigrid solver based on distributive smoother and residual overweighing for Oseen problems. Numer Math. 2015;129(2):237–52.
25. Schöberl J, Zulehner W. On Schwarz-type smoothers for saddle point problems. Numer Math. 2003;95(2):377–99.
26. Claus L, Bolten M. Nonoverlapping block smoothers for the Stokes equations. Numer Linear Algebra Appl. 2021;28(6):e2389.
27. Gaspar FJ, Notay Y, Oosterlee CW, Rodrigo C. A simple and efficient segregated smoother for the discrete Stokes equations. SIAM J Sci Comput. 2014;36(3):A1187–206.
28. Manservisi S. Numerical analysis of Vanka-type solvers for steady Stokes and Navier–Stokes flows. SIAM J Numer Anal. 2006;44(5):2025–56.
29. Molenaar J. A two-grid analysis of the combination of mixed finite elements and Vanka-type relaxation. In: Hackbusch W, Trottenberg U, editors. Multigrid Methods III. Berlin: Springer; 1991. p. 313–23.
30. Wobker H, Turek S. Numerical studies of Vanka-type smoothers in computational solid mechanics. Adv Appl Math Mech. 2009;1(1):29–55.
31. Bramble JH, Pasciak JE, Vassilev AT. Analysis of the inexact Uzawa algorithm for saddle point problems. SIAM J Numer Anal. 1997;34(3):1072–92.
32. Elman HC, Golub GH. Inexact and preconditioned Uzawa algorithms for saddle point problems. SIAM J Numer Anal. 1994;31(6):1645–61.
33. Luo P, Rodrigo C, Gaspar F, Oosterlee C. On an Uzawa smoother in multigrid for poroelasticity equations. Numer Linear Algebra Appl. 2017;24(1):e2074.
34. Vassilevski PS, Villa U. A block-diagonal algebraic multigrid preconditioner for the Brinkman problem. SIAM J Sci Comput. 2013;35(5):S3–S17.
35. Kobelkov GM, Olshanskii MA. Effective preconditioning of Uzawa type schemes for a generalized Stokes problem. Numer Math. 2000;86(3):443–70.
36. Sarin V, Sameh A. An efficient iterative method for the generalized Stokes problem. SIAM J Sci Comput. 1998;19(1):206–26.
37. But MM. On multigrid solver for generalized Stokes equations. Punjab Univ J Math. 2018;50:53–66.
38. Coley C, Benzaken J, Evans JA. A geometric multigrid method for isogeometric compatible discretizations of the generalized Stokes and Oseen problems. Numer Linear Algebra Appl. 2018;25(3):e2145.
39. Kanschat G, Lazarov R, Mao Y. Geometric multigrid for Darcy and Brinkman models of flows in highly heterogeneous porous media: a numerical study. J Comput Appl Math. 2017;310:174–85.
40. Lin M, Reusken A. A comparative study of efficient iterative solvers for generalized Stokes equations. Numer Linear Algebra Appl. 2008;15(1):13–34.
41. Olshanskii M. Multigrid analysis for the time dependent Stokes problem. Math Comput. 2012;81(277):57–79.
42. Keller C, Gould NI, Wathen AJ. Constraint preconditioning for indefinite linear systems. SIAM J Matrix Anal Appl. 2000;21(4):1300–17.
43. Chidyagwai P, Ladeheim S, Syld DB. Constraint preconditioning for the coupled Stokes–Darcy system. SIAM J Sci Comput. 2016;38(2):A668–90.
44. Rees T, Greif C. A preconditioner for linear systems arising from interior point optimization methods. SIAM J Sci Comput. 2007;29(5):1992–2007.
45. He Y. Novel mass-based multigrid relaxation schemes for the Stokes equations. Appl Math Comput. 2023;440:127665.
46. Greif C, He Y. A closed-form multigrid smoothing factor for an additive Vanka-type smoother applied to the Poisson equation. Numer Linear Algebra Appl. 2023;e2500.
47. Wienands R, Joppich W. Practical Fourier analysis for multigrid methods. Boca Raton, FL: CRC Press; 2004.
48. Trottenberg U, Oosterlee CW, Schuller A. Multigrid. Cambridge, MA: Academic Press; 2000.
49. Farrell PE, He Y, MacLachlan SP. A local Fourier analysis of additive Vanka relaxation for the Stokes equations. Numer Linear Algebra Appl. 2021;28(3):e2306.
50. He Y, Liu J. A Vanka-type multigrid solver for complex-shifted Laplacian systems from diagonalization-based parallel-in-time algorithms. Appl Math Lett. 2022;132:108–25.

How to cite this article: He Y. A Vanka-based parameter-robust multigrid relaxation for the Stokes–Darcy Brinkman problems. Numer Linear Algebra Appl. 2023;30(6):e2514. https://doi.org/10.1002/nla.2514
Here, we report the convergence history of V(1,1)-cycle multigrid methods with two or three Jacobi iterations for solving the Schur complement system.

**FIGURE A1** Convergence history: Number of iterations versus relative residual of V(1,1)-cycle with \( \epsilon = 2^{-2} \) and two Jacobi iterations for Schur complement system (left \( \omega = 1 \) and right optimal \( \omega \)).

**FIGURE A2** Convergence history: Number of iterations versus relative residual of V(1,1)-cycle with \( \epsilon = 2^{-4} \) and two Jacobi iterations for Schur complement system (left \( \omega = 1 \) and right optimal \( \omega \)).
FIGURE A3  Convergence history: Number of iterations versus relative residual of V(1,1)-cycle with $\epsilon = 2^{-6}$ and two Jacobi iterations for Schur complement system (left $\omega = 1$ and right optimal $\omega$).

FIGURE A4  Convergence history: Number of iterations versus relative residual of V(1,1)-cycle with $\epsilon = 2^{-8}$ and two Jacobi iterations for Schur complement system (left $\omega = 1$ and right optimal $\omega$).
FIGURE A5  Convergence history: Number of iterations versus relative residual of V(1,1)-cycle with $\epsilon = 2^{-2}$ and three Jacobi iterations for Schur complement system (left $\omega = 1$ and right optimal $\omega$).

FIGURE A6  Convergence history: Number of iterations versus relative residual of V(1,1)-cycle with $\epsilon = 2^{-4}$ and three Jacobi iterations for Schur complement system (left $\omega = 1$ and right optimal $\omega$).
FIGURE A7  Convergence history: Number of iterations versus relative residual of $V(1,1)$-cycle with $\epsilon = 2^{-6}$ and three Jacobi iterations for Schur complement system (left $\omega = 1$ and right optimal $\omega$).

FIGURE A8  Convergence history: Number of iterations versus relative residual of $V(1,1)$-cycle with $\epsilon = 2^{-8}$ and three Jacobi iterations for Schur complement system (left $\omega = 1$ and right optimal $\omega$).