Brabazon, Keeran J. and Hubbard, Matthew E. and Jimack, Peter K. (2014) Nonlinear multigrid methods for second order differential operators with nonlinear diffusion coefficient. Computers and Mathematics with Applications, 68 (12A). pp. 1619-1634. ISSN 0898-1221

Nonlinear Multigrid Methods for Second Order Differential Operators with Nonlinear Diffusion Coefficient

K.J. Brabazon[a,*], M.E. Hubbard[a,b], P.K. Jimack[a]

[a]*School of Computing, University of Leeds, Leeds, LS2 9JT, UK*
[b]*School of Mathematical Sciences, The University of Nottingham, University Park, Nottingham, NG7 2RD*

Abstract

Nonlinear multigrid methods such as the Full Approximation Scheme (FAS) and Newton-multigrid (Newton-MG) are well established as fast solvers for nonlinear PDEs of elliptic and parabolic type. In this paper we consider Newton-MG and FAS iterations applied to second order differential operators with nonlinear diffusion coefficient. Under mild assumptions arising in practical applications, an approximation (shown to be sharp) of the execution time of the algorithms is derived, which demonstrates that Newton-MG can be expected to be a faster iteration than a standard FAS iteration for a finite element discretisation. Results are provided for elliptic and parabolic problems, demonstrating a faster execution time as well as greater stability of the Newton-MG iteration. Results are explained using current theory for the convergence of multigrid methods, giving a qualitative insight into how the nonlinear multigrid methods can be expected to perform in practice.

Keywords: Nonlinear Multigrid, Newton's Method, Nonlinear Diffusion
2010 MSC: 35J60, 35J92, 35K55

1. Introduction

Nonlinear multigrid iterations such as the Full Approximation Scheme (FAS) [9] and Newton-Multigrid (Newton-MG) [12, 48] methods have been widely used to solve elliptic and parabolic nonlinear problems on large scales (cf. [4, 17, 18, 21, 27, 42] amongst others). There exists very little convergence theory for the nonlinear methods (e.g. [2, 22, 25, 28]) and only limited comparison as to which method should be preferred in practice [23, 26, 33, 34, 47],

*Corresponding Author
Email addresses: scs5kjb@leeds.ac.uk (K.J. Brabazon), Matthew.Hubbard@nottingham.ac.uk (M.E. Hubbard), p.k.jimack@leeds.ac.uk (P.K. Jimack)

where the comparisons are limited to specific applications. In this paper we present a more general framework for comparing the relative efficiency of the Newton-MG and FAS methods for a broad class of nonlinear problems. This requires a detailed discussion of the efficient implementation of these schemes, followed by a theoretical assessment of their running times in a finite element setting for a general second order nonlinear operator. The comparison is based upon a detailed analysis of their costs per cycle, followed by a theoretical discussion of their convergence properties and how this theory may be used when comparing the techniques. As there exists no algebraic variant of FAS multigrid, the geometric algorithms are compared.

The remainder of the paper is structured as follows. In section 2 we briefly present the linear and nonlinear multigrid iterations, followed by a detailed discussion of the theoretical running time of Newton-MG and FAS in section 3. Section 4 describes and applies the relevant convergence theory for linear and nonlinear multigrid iterations. In section 5 model problems are introduced, which are used to produce results in section 6, to demonstrate the applicability of the theory from sections 3 and 4. Conclusions are given in section 7, which summarise the reasons why a Newton-MG iteration should be preferred over an FAS iteration when using a finite element discretisation.

2. Background

In this section we introduce the basic concepts and notation required for the definition of both linear and nonlinear geometric multigrid algorithms. A more detailed introduction can be found in [12, 22, 48]. The problem to be solved is presented as an operator equation. Once an operator is discretised, and an appropriate basis for a discrete subspace has been chosen, the discrete operator equation may be considered an algebraic system of equations. In the following we move between considering operator equations and the corresponding algebraic systems of equations, as appropriate.

2.1. Linear Multigrid Algorithms

We wish to solve the linear operator equation given by

$$\mathcal{A}u(x) = f(x), \quad x \in \Omega \tag{2.1}$$

where the domain $\Omega \in \mathbb{R}^d$ has boundary $\partial\Omega$, and $\mathcal{A} : \mathcal{V} \to \mathcal{V}$ for some vector space $\mathcal{V}$. From this point on we omit the explicit dependence on $x \in \mathbb{R}^d$. It is assumed that there is a unique $u^* \in \mathcal{V}$ satisfying equation (2.1). We are interested in the approximate solution of (2.1) based upon discretisations using a sequence of finite-dimensional grids

$$\Omega_1 \subset \Omega_2 \subset \ldots \subset \Omega_J, \tag{2.2}$$

which are sets of connected points in Ω. We also consider the sequence of finite-dimensional function spaces

$$V_1 \subset V_2 \subset \ldots \subset V_J \subset \mathcal{V}, \tag{2.3}$$

where each V_l, $l = 1, \ldots, J$ is defined on grid Ω_l. Given (2.3), we consider the discretised system of equations

$$A_l u_l = f_l \tag{2.4}$$

where $A_l : V_l \to V_l$ is the projection of the continuous operator $\mathcal{A}$ onto the finite-dimensional space V_l. We assume there are unique $u_l^* \in V_l$, $l = 1, \ldots, J$ that satisfy (2.4). For the purposes of this paper V_l, $l = 1, \ldots, J$ is the standard piecewise linear finite element function space defined on grid Ω_l.

2.1.1. Linear Multigrid as a Solver

In this section we describe the linear geometric multigrid method and introduce some notation. A discussion of necessary conditions for convergence of geometric multigrid is presented in section 4.

We introduce operators

$$\begin{aligned} R_l &: V_l \to V_{l-1}, \quad l = 2, \ldots, J \\ P_l &: V_{l-1} \to V_l, \quad l = 2, \ldots, J, \end{aligned} \tag{2.5}$$

which are restriction and prolongation operators, respectively, that allow the transfer of functions between different subspaces. Since the exact solution to (2.4) is $u_l^* \in V_l$, the error e_l and defect r_l in approximation u_l, defined by

$$e_l = u_l^* - u_l, \quad r_l = f_l - A_l u_l,$$

satisfy the operator equation

$$A_l e_l = r_l. \tag{2.6}$$

We assume there exist operators $S_l : V_l \to V_l$, $l = 2, \ldots, J$, called smoothing operators, that have the property that they are effective at removing high frequency components from the error [12, 48]. A correction term is calculated on a coarser grid in the *coarse grid correction* step. We fix a number ν of smooths to perform before (pre-smoothing) and after (post-smoothing) a coarse grid correction step. In general the number of pre- and post-smooths may differ, and one of the smoothing steps may be left out entirely [12, 22, 48].

Consider that we wish to solve (2.4) on grid Ω_l, $l \neq 1$, for the exact discrete solution $u_l^* \in V_l$. A single step of the geometric multigrid algorithm is then outlined in algorithm 2.1, where $u_l^{(j)} \in V_l$ represents the approximation to the solution u_l^* after j iterations of linear multigrid have been performed. This iteration can be performed until some appropriate convergence / failure criteria are met. On the coarsest grid level the exact inverse is very inexpensive to compute, provided that $|\Omega_1|$ is small, and the running time of the algorithm is $O(n)$ for particular choices of A (cf. [12, 48]). There is a closed form representation of the linear multigrid V-cycle operator, which for the rest of this paper is denoted $M_l : V_l \to V_l$. The exact representation can be found in [48]. A multigrid operator should possess the smoothing and coarse grid correction properties [22]. That is, the smoother should remove high frequency components from the error, and the coarse grid correction should give a good approximation to the fine grid error after the high frequency components have been removed.

Algorithm 2.1 Multigrid V-Cycle

Require: A_l, A_{l-1}, S_l, P_l, R_l

function LINEAR-MG$(l, u_l^{(j)}, f_l, \nu)$
  Set $u_l' = S_l^\nu u_l^{(j)}$
  if $l = 2$ **then** Calculate $\tilde{e}_{l-1} = A_{l-1}^{-1} R_l (f_l - A_l u_l')$
  else Set $\tilde{e}_{l-1} =$ LINEAR-MG$(l-1, 0, R_l(f_l - A_l u_l'), \nu)$
  return $u_l^{(j+1)} = S_l^\nu (u_l^{(j)} + P_l \tilde{e}_{l-1})$

2.1.2. Multigrid Preconditioned Linear Iterations

Multigrid iterations are frequently used as a preconditioner for a different iterative method. For a more thorough discussion of such methods see [43]. The discrete problem (2.4) can be solved iteratively using a right-preconditioned Krylov subspace method, such as conjugate gradients (CG) or GMRES [43]. As the preconditioner we use a single multigrid V-cycle such that we solve

$$A_l M_l v_l = f_l, \quad M_l^{-1} u_l = v_l. \tag{2.7}$$

Multigrid V-cycles used as a preconditioner for a Krylov subspace solver have been shown to be optimal in many situations (e.g. [13, 16, 49]).

2.2. Nonlinear Multigrid Algorithms

In this section we introduce nonlinear multigrid methods, using concepts introduced for linear multigrid methods.

2.2.1. Newton-Multigrid (Newton-MG)

An inexact Newton method is presented, which is given in detail in [14]. Consider the discrete nonlinear system

$$F_J\left(u_J(x)\right) = 0, \tag{2.8}$$

for nonlinear operator $F_J : V_J \to V_J$. Ω_J and V_J are the finest grid, and associated subspace, in the hierarchies (2.2) and (2.3), respectively. Assume there exists a solution u_J^* to (2.8) and that this is unique in

$$B_{u_J^*} \equiv \left\{u_J \in V_J : ||u_J - u_J^*||_{V_J} < \delta\right\},$$

for some $\delta > 0$. Let F_J be Frèchet differentiable [38] in $B_{u_J^*}$ and assume that this derivative is invertible. A single step of an inexact Newton method is then given by algorithm 2.2.

Algorithm 2.2. *Inexact Newton*

$$u_J^{(j+1)} = u_J^{(j)} - A_{u_J} F_J(u_J^{(j)})$$

In algorithm 2.2 $u_J^{(j)}$ is the approximation after j inexact Newton iterations, and A_{u_J} is some approximation to the inverse of the Fréchet derivative of F_J (i.e. $A_{u_J} \approx F'_J(u_J^{(j)})^{-1}$). The definition of A_{u_J} is required, which may be any linear operator that approximates the inverse of the derivative of F_J at $u_J^{(j)}$. Letting A_{u_J} be a number of iterations of the linear multigrid operator M_J (i.e. $A_{u_J} = M_J^p$ for some $(p \in \mathbb{N}) > 0$) applied to the equation $F'_J(u_J^{(j)})e_J = F_J(u_J^{(j)})$ gives a Newton-Multigrid (Newton-MG) iteration. We may use other choices for A_{u_J}, including iterative Krylov subspace methods, and their preconditioned variants, as described in section 2.1.

2.2.2. Full Approximation Scheme (FAS)

Brandt [9] was one of the first to introduce nonlinear multigrid, which seeks to use concepts from the linear multigrid iteration and apply them directly in the nonlinear setting. The iteration devised in [9] is called the Full Approximation Scheme (FAS) and it is this version of nonlinear multigrid that is considered in this paper. The Nonlinear Multilevel Method (NMLM) of Hackbusch [22] is a generalisation of FAS, but adds extra parameters into the algorithm. For brevity of discussion we consider only the FAS algorithm here, but note that the NMLM is a globally convergent iteration [25, 40]. Consider the discrete nonlinear problem

$$A_l(u_l) = f_l, \quad l = 1, \ldots, J, \tag{2.9}$$

with exact solution u_l^* unique in some $B_{u_l^*}$. This is equivalent to (2.8) with $F_l(u_l) = A_l(u_l) - f_l$. The defect equation reads

$$A_l(u_l^*) = r_l + A_l(u_l), \tag{2.10}$$

where the defect $r_l = A_l(u_l^*) - A_l(u_l)$. As in the linear case, FAS is a combination of smoothing and coarse grid correction. The smoothing operators S_l are nonlinear, but should possess the smoothing property, as in the linear case [22]. There are nonlinear variants of standard Jacobi and Gauss-Seidel smoothers, including variants to perform block smoothing (cf. [9, 12, 22, 48]). Using the prolongation and restriction operators given in (2.5), the nonlinear V-Cycle is defined in algorithm 2.3.

Algorithm 2.3 FAS Multigrid

Require: A_l, A_{l-1}, S_l, P_l, R_l

1: **function** FAS($l, u_l^{(j)}, f_l, \nu$)
2: Set $u'_l = S_l^\nu(u_l^{(j)})$
3: Set $u'_{l-1} = R_l u'_l$; $f_{l-1} = R_l(f_l - A_l(u'_l)) + A_{l-1}(u'_{l-1})$
4: **if** $l = 2$ **then** Calculate $\tilde{u}_{l-1} = A_{l-1}^{-1}(f_{l-1})$
5: **else** Set $\tilde{u}_{l-1} = \text{FAS}(l-1, u'_{l-1}, f_{l-1}, \nu)$
return $u_l^{(j+1)} = S_l^\nu(u'_l + P_l(\tilde{u}_{l-1} - u'_{l-1}))$

Algorithm 2.3 is very similar to algorithm 2.1, the main difference being that the value to solve for on the coarse grids in FAS is the exact solution instead

of the error in the approximation. If the operators are linear then algorithm 2.3 is algebraically equivalent to algorithm 2.1 and hence is a generalisation of geometric multigrid to the nonlinear setting.

3. Computational Cost of FAS versus Newton-MG

There are several papers in which FAS multigrid is compared against Newton-MG for particular model problems [23, 33–35, 47]. Most of these compare the running times as well as the convergence factors, and come to the same conclusion - that the execution time for Newton-MG methods is less than for FAS multigrid [33–35, 47]. For a finite difference discretisation the running times of the algorithms are shown to be much closer [26]. For finite differences the calculation of the nonlinear residual is cheaper than for finite elements, as integrals over elements do not have to be calculated. The framework developed here can be used to show that this means that the execution time of an FAS iteration, relative to a Newton-MG iteration, will be less than for finite elements. For brevity this is not discussed in this paper.

In this section we consider the Newton-MG and FAS iterations applied to a general nonlinear problem. A theoretical bound on the running time is developed that allows a direct comparison between the two methods and makes it clear why Newton-MG methods perform faster than FAS multigrid methods. In section 6 the sharpness of these bounds is demonstrated for model problems introduced in section 5. A note to make is that [23] finds that the Nonlinear Multilevel Method (NMLM) [22] performs better than a standard Newton-MG iteration. This is to be expected, though, as NMLM is a global algorithm. The same advantages of using Newton-MG over FAS can be found when comparing a global version of Newton's method to the NMLM iteration, although this comparison is not presented in this paper.

We characterise the amount of computational effort required to perform each of the algorithms in terms of a *work unit*. One work unit (denoted W_l) is the amount of time required to calculate the nonlinear residual on grid Ω_l. The amount of work required depends on the number of dimensions in which we are working, as well as the discretisation and choice of smoothers. In this section we present a general formula for calculating the amount of computational time required for a single linear or nonlinear V-cycle when using a finite element discretisation with piecewise linear basis functions on simplexes in arbitrary dimension. We then take an illustrative example to demonstrate the sharpness of the estimates.

We consider that we are working with the grid hierarchy described in (2.2), where $\Omega_l \subset \mathbb{R}^d$, $l = 1, \ldots, J$. We consider a linear finite element discretisation with nodal basis functions such that $\text{span}\{\varphi_i^l\}_{i=1}^{N_l} = \mathcal{V}_l$, $l = 1, \ldots, J$ on a quasi-regular simplicial partitioning $\mathcal{T}_l$ (i.e. a triangulation in two dimensions, and a partitioning into tetrahedra in three dimensions) on grid Ω_l. The superscript l for the basis functions is dropped when it is apparent from the context which grid is referred to. N_l is the number of unknowns on grid Ω_l. A fine grid simplicial partitioning is gained from a bisection, quadrisection or octasection of the coarse

simplicial partitioning in one, two or three dimensions, respectively. In two dimensions and example of a quadrisection method is gained by connecting the mid-points of the edges of the elements. Hence, if N_l represents the number of unknowns on grid Ω_l, the number of unknowns on the next finest grid Ω_{l+1} is given by

$$N_{l+1} \approx 2^d N_l. \tag{3.1}$$

We consider that a problem has been given in weak form as

$$\mathcal{F}(u, \varphi) = 0, \quad \forall \varphi \in \mathcal{V} \tag{3.2}$$

and solve the discrete weak form given by

$$F_l(u_l, \varphi_i) = 0, \quad i = 1, \ldots, N_l, \tag{3.3}$$

for $l = 1, \ldots, J$. The application of F_l can be calculated as a sum over the elements $\mathcal{K} \in \mathcal{T}_l$ as follows:

$$F_l(u_l, \varphi_i) = \sum_{\mathcal{K} \in \mathrm{supp}(\varphi_i)} F_l^{(\mathcal{K})}(u_l, \varphi_i), \tag{3.4}$$

where $F_l^{(\mathcal{K})}$ is the discrete operator restricted to element $\mathcal{K} \in \mathcal{T}_l$. Let the Frèchet derivative of $\mathcal{F}(u, \varphi)$ at u be denoted by

$$\mathcal{F}_u(\psi, \varphi) = D\left[\mathcal{F}(u, \varphi)\right](\psi) \tag{3.5}$$

and assume that this exists and is invertible for all $u \in \mathcal{V}$. Then the Jacobian matrix

$$\Big[F_{u,l}(\varphi_j, \varphi_i)\Big]_{i,j=1}^{N_l} \tag{3.6}$$

exists and is invertible, where $F_{u,l}(\,\cdot\,,\,\cdot\,)$ is the discretisation of the Frèchet derivative on grid Ω_l. For practical purposes we assume that the entries in the Jacobian matrix are calculated using numerical differentiation, as for complex problems this requires less computational time to calculate than using exact formulas for the derivative. The entry in row i, column j of the Jacobian matrix is constructed as follows:

$$F_{u,l}(\varphi_j, \varphi_i) \approx \sum_{\mathcal{K} \in \mathrm{supp}(\varphi_i) \cap \mathrm{supp}(\varphi_j)} \frac{F_l^{(\mathcal{K})}(u_l + \epsilon\varphi_j, \varphi_i) - F_l^{(\mathcal{K})}(u_l, \varphi_i)}{\epsilon}. \tag{3.7}$$

The value $F_l^{(\mathcal{K})}(u_l, \varphi_i)$ is used in the calculation of the nonlinear residual, and can be re-used when calculating the entries in the Jacobian matrix. Hence, per simplex, when calculating the residual we can perform an extra local function evaluation at each of the nodes to obtain the element-wise contribution to the Jacobian matrix. This means that in d dimensions we require one function evaluation to calculate the residual and an extra $d+1$ function evaluations (per simplex) to calculate the contribution to the Jacobian matrix. Therefore the

cost of calculating the Jacobian matrix and the residual is $(d+2)W_l$, where W_l is the cost of calculating the residual on grid Ω_l. Using similar reasoning we see that calculating the diagonal entries of the Jacobian matrix requires an extra local function evaluation per node on each element. Hence the cost to calculate the nonlinear residual and the diagonals of the Jacobian matrix is $2W_l$. The diagonals of the Jacobian matrix are used in the nonlinear smoothing operator.

We finally note that using (3.1) we can characterise the cost of calculating the nonlinear residual on grid Ω_{l-1} as

$$W_{l-1} \approx \frac{1}{2^d} W_l. \tag{3.8}$$

Let $C^{(l)}_{\mathrm{NMG}}$ be the cost of performing a single Newton iteration on grid Ω_l, and $C^{(l)}_{\mathrm{FAS}}$ the cost of performing a single FAS V-cycle. From algorithms 2.2 and 2.3 we see that

$$C^{(l)}_{\mathrm{NMG}} \approx C^{(l)}_{\mathrm{RJ}} + \sum_{i=1}^{l-1} C^{(i)}_{\mathrm{J}} + pC^{(l)}_{\mathrm{LMG}} \tag{3.9}$$

$$C^{(l)}_{\mathrm{FAS}} \approx C^{(l)}_{\mathrm{S}}(\nu_1 + \nu_2) + C^{(l)}_{\mathrm{RHS}} + C^{(l-1)}_{\mathrm{FAS}} \tag{3.10}$$

for $C^{(l)}_{\mathrm{RJ}}$ the cost of calculating the nonlinear residual and Jacobian; $C^{(l)}_{\mathrm{J}}$ the cost of calculating the Jacobian; p the number of linear V-cycles to perform per Newton iteration; $C^{(l)}_{\mathrm{LMG}}$ the cost of performing a linear V-cycle; $C^{(l)}_{\mathrm{S}}$ the cost of a nonlinear smoothing operation; ν_1 and ν_2 the number of pre- and post-smoothing iterations, respectively; and $C^{(l)}_{\mathrm{RHS}}$ the cost of calculating the perturbed right-hand side for FAS. In the characterisations (3.9) and (3.10) we have made the assumption that the nonlinear operations dominate the execution time of the two iterations, except for the inclusion of the linear multigrid cycle. As will be seen this is a fair assumption for the simple model problems presented, and becomes more accurate the more complicated the nonlinear operator is.

In (3.9) we see that the cost of the Newton iteration is given as the cost of calculating the residual and Jacobian matrix on the current grid level plus the cost of performing the linear multigrid iterations. We also include the calculation of the Jacobian matrix on the coarser grid levels here. This is because the results presented are for when the linearisation of the nonlinear operator is re-discretised on each grid level. We may instead use a Galerkin coarse grid operator (cf. [12, 48]), in which case the cost of interpolating the fine grid operator onto coarser grids should replace the cost of re-discretising the Jacobian on each grid level.

In (3.10) we see that the cost of the FAS iteration is given as the cost of smoothing on the current grid, calculating the perturbed right-hand side for the next coarsest grid, and performing an FAS V-cycle on the next coarsest grid. From previous discussion we have that

$$C^{(l)}_{\mathrm{RJ}} = C^{(l)}_{\mathrm{J}} = (d+2)W_l. \tag{3.11}$$

To calculate the cost of the nonlinear smoother we need to know which nonlinear smoother we are using. Assuming that we are performing a pointwise nonlinear smooth, such as nonlinear Jacobi (cf. [38]) we have to calculate the nonlinear residual in each iteration, as well as the diagonals of the current Jacobian matrix. Using previous discussion the cost of this is

$$C_{\mathrm{S}}^{(l)} = 2W_l. \tag{3.12}$$

This cost is accurate if we are performing a pointwise nonlinear Jacobi iteration. A pointwise Gauss-Seidel iteration requires the recalculation of the operator over the support of a basis function every time that a pointwise value is updated [47]. Hence a full pointwise nonlinear Gauss-Seidel iteration is considerably more expensive than the Jacobi iteration. If some block smooth is performed we need to calculate at least some off-diagonal entries in the Jacobian matrix. This means that the cost of a block smooth will also be considerably higher than the cost of the pointwise Jacobi smoother. In this investigation we consider the case of a Jacobi smoother, as this gives the smallest cost per iteration. Note that it is possible that a novel nonlinear smoother may be developed which is more efficient, or has much improved convergence properties, compared to a pointwise Jacobi iteration. Then FAS may become more competitive. However, it is highly unlikely that a novel smoother will give a large increase in performance, as when dealing with a nonlinear operator the nonlinear residual must be re-calculated in each smoothing step.

Finally we need to find the cost of calculating the perturbed right-hand side for the next coarsest grid level. From algorithm 2.3 we see that to calculate the perturbed right-hand side we calculate the residual on the current grid level, and apply the nonlinear operator to the restricted approximation on the coarser grid. The cost of applying the nonlinear operator is approximately the cost of calculating the residual on a given grid. Hence we have that

$$C_{RHS}^{(l)} = W_l + W_{l-1} = (1 + 2^{-d})W_l. \tag{3.13}$$

We can now characterise the cost of both of the nonlinear iterations. We first consider Newton-MG, leaving the characterisation of the running time of the linear multigrid iteration until later. We find that

$$\begin{aligned} C_{\mathrm{NMG}}^{(l)} &= (d+2)W_l + \sum_{i=0}^{l-1}(d+2)W_i + pC_{\mathrm{LMG}}^{(l)} \\ &= (d+2)\sum_{i=1}^{l}\frac{1}{2^{id}}W_l + pC_{\mathrm{LMG}}^{(l)} \\ &\leq (d+2)\frac{2^d}{2^d-1}W_l + pC_{\mathrm{LMG}}^{(l)}. \end{aligned} \tag{3.14}$$

We are interested in knowing the running time per linear V-cycle, so we introduce a scaled variable

$$\tilde{C}_{\mathrm{NMG}}^{(l)} \equiv \frac{C_{\mathrm{NMG}}^{(l)}}{p} \tag{3.15}$$

to be the cost per linear V-cycle of the Newton-MG algorithm.

Now consider the FAS multigrid iteration. We can approximate the cost of a single nonlinear V-cycle using the previous discussions as

$$\begin{aligned} C_{\text{FAS}}^{(l)} &= 2W_l(\nu_1+\nu_2) + (1+2^{-d})W_l + C_{\text{FAS}}^{(l-1)} \\ &= \left(2(\nu_1+\nu_2)+1+2^{-d}\right)W_l + C_{\text{FAS}}^{(l-1)} \\ &= \left(2(\nu_1+\nu_2)+1+2^{-d}\right)\left(W_l + \frac{W_l}{2^d}\right) + C_{\text{FAS}}^{(l-2)} \\ &= \cdots \\ &= \left(2(\nu_1+\nu_2)+1+2^{-d}\right)\left(\sum_{i=1}^{l-1}\frac{W_l}{2^{d(i-1)}}\right) + C_{\text{FAS}}^{(1)} \\ &\leq \left(2(\nu_1+\nu_2)+1+2^{-d}\right)\frac{2^d}{2^d-1}W_l, \end{aligned} \tag{3.16}$$

where in the last step we have assumed that the time taken to solve on the coarsest grid level $C_{\text{FAS}}^{(1)}$ is negligible compared to W_l. This is usually the case, but requires that the solution of the nonlinear equation is well approximated on the coarsest grid. If this is not the case then higher dimensional coarse spaces are required and the cost of the coarsest grid solve starts to have an adverse effect on the execution time.

We now compare the cost of a single nonlinear V-cycle to the cost of a linear V-cycle for a more specific problem. We restrict ourselves to $d = 2$ and estimate the running time of the linear multigrid algorithm in terms of a work unit. From empirical experiment we have found that an upper limit for a single V-cycle is given by

$$C_{\text{LMG}}^{(l)} \leq \frac{3}{2}W_l, \tag{3.17}$$

where three linear pre- and post-smooths are performed. This ratio is a very good estimate for a simple model problem (cf. discussion of the p-Laplacian in section 5), and becomes more and more pessimistic as the complexity of the nonlinear operator increases. As the complexity of the nonlinear operator increases the cost of a work unit W_l increases. However, the cost of the a linear multigrid iteration remains constant as we calculate the linear operators required prior to starting the linear iteration. Hence, the more complicated the problem the less time a linear V-cycle takes compared to the cost of calculating the nonlinear residual. Therefore, the results presented here are for a worst-case scenario.

We consider the case that we perform equal numbers of pre- and post-smooths by setting $\nu_1 = \nu_2 = \nu$, and consider the amount of time spent per FAS iteration to get

$$C_{\text{FAS}}^{(l)} \approx \left(4\nu + \frac{5}{4}\right)\frac{4}{3}W_l = \begin{cases} 7W_l, & \nu = 1 \\ \frac{37}{3}W_l, & \nu = 2 \\ \frac{53}{3}W_l, & \nu = 3. \end{cases} \tag{3.18}$$

The cheapest iteration therefore costs approximately $7W_l$. We compare this to the cost of the Newton-MG iteration per linear V-Cycle

$$\tilde{C}^{(i)}_{\text{NMG}} \approx \left(\frac{16}{3p} + \frac{3}{2}\right) W_l = \begin{cases} \frac{41}{6} W_l, & p = 1 \\ \frac{25}{6} W_l, & p = 2 \\ \frac{59}{18} W_l, & p = 3. \end{cases} \tag{3.19}$$

The most expensive cost per iteration for Newton-MG is less than the cheapest cost per iteration for FAS, and see that the more accurately we solve the Newton step (i.e. the more linear V-cycles we perform) the less the cost per linear V-cycle is. Hence a good approach for the Newton-MG method would be to minimise the number of Newton iterations, whilst maximising the convergence factor per linear V-cycle. There are discussions on how to do this (for example [48]). For the discussion in this paper we concentrate on using a fixed number of linear iterations per Newton step, which is often enough to get an efficient optimal order iteration. In practice it is found that using three linear V-cycles per Newton step often gives the fastest execution time for the iteration.

Increasing the number of smoothing iterations for the FAS iteration increases the cost per V-cycle, but does improve the convergence factor. Our empirical experiments suggest that three smoothing steps per FAS V-cycle are often optimal. Using three pre- and post-smoothing steps the cost per FAS V-cycle is approximated by

$$C^{(l)}_{\text{FAS}} = \frac{53}{3} W_l \tag{3.20}$$

and the cost per linear V-cycle for Newton-MG is approximated by

$$\tilde{C}^{(l)}_{\text{NMG}} = \frac{59}{18} W_l. \tag{3.21}$$

The ratio between the costs per iteration is then given by

$$\frac{C^{(l)}_{\text{FAS}}}{\tilde{C}^{(l)}_{\text{NMG}}} = \frac{318}{59} \approx 5.4. \tag{3.22}$$

This shows that, so long as we don't perform more than 5.4 times more linear V-cycles in the Newton-MG iteration than nonlinear FAS V-cycles to get convergence, the Newton-MG will be the quicker iteration. Results supporting this, as well as demonstrating that similar numbers of V-cycles are required for convergence for Newton-MG and FAS, are given in section 6, indicating that Newton-MG is faster than FAS in terms of running time. The results given in section 6 also highlight the sharpness of the estimates presented above.

4. Convergence Theory

Having carefully analysed the cost per V-cycle in the previous section we now present an overview of the existing theory for the convergence of nonlinear

multigrid methods, including some of the most relevant gaps, for which the theory is still lacking. We begin with a discussion of the convergence theory for linear multigrid and apply this in the case of Newton-MG. We then compare and contrast what is known about the convergence of nonlinear multigrid methods.

4.1. Linear Multigrid

The linear multigrid theory is well established for the case of symmetric positive definite operators. There are three main ways in which a multigrid iteration can be analysed: the theory suggested by Hackbusch [22]; local Fourier analysis [9, 10, 32, 37, 50]; and subspace correction theory [7, 8, 52]. In this overview we concentrate on subspace correction theory, as this gives the strongest results, although in practice it is often useful to use the local Fourier analysis, as this is capable of giving quantitative estimates on the convergence factor of a multigrid iteration [12, 48, 50].

For the problem

$$\begin{aligned} \mathcal{A}u &= f, \quad x \in \Omega, \\ u &= 0, \quad x \in \partial\Omega, \end{aligned} \tag{4.1}$$

in the case where $\mathcal{A}$ is a symmetric positive definite linear operator the subspace correction theory [52] tells us that when $\mathcal{A}$ contains a coefficient function that varies mildly over the spatial domain then the multigrid convergence factor is independent of mesh parameters and the coefficient function. There are no regularity assumptions made on the solution u or the right-hand side f, as there were prior to [7]. In the case where coefficient functions in $\mathcal{A}$ are highly varying, and possibly discontinuous, over the domain, the theory by Scheichl et al [44] tells us in which situations the convergence of a multigrid method is independent of the size of the jumps in coefficient.

As well as the convergence factor being independent of mesh parameters, it can be shown [12, 48] that the time complexity of geometric multigrid is $O(n)$ (for n the number of unknown nodes on a grid) when a reduction in the norm of a residual is required below a fixed tolerance. This means that multigrid is referred to as an 'optimal order' algorithm.

Convergence proofs for symmetric positive definite operators representing linear second order differential operators use the fact that the energy norm can be used to define a norm on a Hilbert space, which is equivalent to the $H_0^1(\Omega)$ norm [5, 52]. The extensive theory from Hilbert spaces can then be used to demonstrate the convergence of the iteration in the $H_0^1(\Omega)$ norm under mild assumptions (cf. [44] and [52]).

In the case where $\mathcal{A}$ is a non-symmetric or indefinite operator the standard geometric multigrid algorithm with a Jacobi (Gauss-Seidel) smoother may be a sub-optimal iteration. Convergence has not been proved to be independent of mesh parameters beyond cases where the operator is a 'compact perturbation' [6] from the symmetric positive definite case. There has been little work to extend the theory beyond this limited statement as it is known that, in practice, the standard geometric multigrid algorithm is not suited, without modification, to solve problems that are indefinite or highly non-symmetric. Also, the subspace

correction theory is not applicable in the non-symmetric or indefinite cases, as the operator can no longer be used to define an inner product on a Hilbert space. Hence the theory by Hackbusch [22] or local Fourier analysis needs to be applied.

4.2. Newton-Multigrid (Newton-MG)

Newton methods are well established and there are many resources regarding the implementation and analysis of the technique. The monograph by Deuflhard [14] is referred to often here, although there is a wealth of other resources available. The convergence theory of Newton methods may be classified as local or global convergence proofs. Local proofs show that a standard Newton method applied to the discrete problem

$$F(u) = 0 \tag{4.2}$$

will converge to a local solution u^* given some initial approximation $u^{(0)} \in B_{u^*} \equiv \{u : \|u - u^*\| < \delta\}$ for some $\delta > 0$. B_{u^*} is termed a ball of guaranteed convergence. Local convergence results for Newton's method often show quadratic convergence of the method as the approximation approaches the exact solution.

Global Newton methods use some damping parameter such that the iteration reads

$$u^{(j+1)} = u^{(j)} - \gamma F'(u^{(j)})^{-1} F(u^{(j)}), \quad \gamma > 0. \tag{4.3}$$

The parameter γ can be chosen to depend on $u^{(j)}$, $F(u^{(j)})$, and/or $F'(u^{(j)})$ to extend the radius of the ball of guaranteed convergence [14], and hence give a more global convergence. Here we have used the notation $F'(u)$ to denote the Jacobian matrix of F evaluated at u.

As with the convergence results of all nonlinear iterative methods the choice of the initial approximation is very important, as this must be 'close enough' to the exact solution to guarantee convergence. For a more thorough discussion of Newton methods and the convergence thereof we refer to [14]. In the scope of this paper we are interested in what the linear multigrid theory can tell us about the convergence of a Newton-MG iteration. Due to the construction of an inexact Newton method (cf. algorithm 2.2) the analysis may be split into the analysis of the nonlinear outer iteration and the linear inner iteration. The only assumption that needs to be made is that the linear inner iteration must be solved 'accurately enough' [14, 28] to allow for an appropriate correction to be calculated in the nonlinear outer iteration. As this is a matter of choosing appropriate stopping criteria for the inner iteration it is possible to consider the convergence of the linear and nonlinear iterations separately. The requirement that the linear inner iteration is convergent is a necessary but not sufficient condition for the convergence of the nonlinear outer iteration. Therefore the linear multigrid theory may be used to indicate when the outer iteration will not converge (i.e. when the inner iteration is not convergent).

We now introduce the general second order PDE with homogeneous Dirichlet boundary conditions given by

$$\begin{aligned} -\nabla \cdot \{a(u, \nabla u, x)\nabla u\} + b(x)\nabla u + c(x)u = 0, \quad x \in \Omega \\ u = 0, \quad x \in \partial\Omega. \end{aligned} \tag{4.4}$$

In this paper we allow a to depend on either u or ∇u, but, for simplicity, not both. Let $c \geq 0$ for all $x \in \Omega$ be a bounded function. We assume that there exists a weak solution $u^* \in \mathcal{V}$ unique in $B_{u^*} \equiv \{u : \|u - u^*\|_{\mathcal{V}} < \delta\}$ for some $\delta > 0$. The weak formulation of (4.4) is given by

$$\mathcal{F}(u, v) \equiv \int_\Omega a(u, \nabla u, x)\nabla u \nabla v + b(x)\nabla u v + c(x)uv \, \mathrm{d}x = 0, \quad \forall v \in \mathcal{V} \tag{4.5}$$

and the Frèchet derivative

$$\begin{aligned} &\mathcal{F}_u(w, v) \equiv D[\mathcal{F}(u, v)](w) = \\ &\int_\Omega D\left[a(u, \nabla u, x)\right](w) \cdot \nabla u \nabla v + a(u, \nabla u, x)\nabla w \nabla v + b(x)\nabla w v + c(x)wv \, \mathrm{d}x. \end{aligned} \tag{4.6}$$

The notation $\mathcal{F}_u(w, v)$ is chosen to reflect that the Frèchet derivative is linear in both w and v. It is assumed that $\mathcal{F}_u$ exists and is invertible in B_{u^*}.

For this paper we consider a particular structure for (4.5) such that the linear multigrid theory can be applied to the linear inner iteration. We set $b(x) = 0$, because, depending on the choice of the nonlinear function $a(u, \nabla u, x)$, a non-symmetric term similar to that multiplied by b may arise in the Frèchet derivative. It should be clear from later discussions what effect the addition of the extra function $b(x)$ has on the convergence of an iteration. We split the Frèchet derivative into a symmetric and non-symmetric part

$$\mathcal{F}_u(w, v) = \mathcal{F}_u^s(w, v) + \mathcal{F}_u^{ns}(w, v) \tag{4.7}$$

with symmetric $\mathcal{F}_u^s$ and non-symmetric $\mathcal{F}_u^{ns}$. The model problems in section 5 will consider the case of $\mathcal{F}_u^{ns} = 0$ and $\|\mathcal{F}_u^s\| \gg \|\mathcal{F}_u^{ns}\|$ for $\|\cdot\|$ some appropriate norm. This leaves the following linear equation to solve for $w^* \in H_0^1(\Omega)$ at each Newton iteration

$$\begin{aligned} \mathcal{F}_u(w, v) = -\mathcal{F}(u, v), \quad \forall v \in \mathcal{V}, \quad x \in \Omega \\ w = 0, \quad x \in \partial\Omega. \end{aligned} \tag{4.8}$$

In (4.8) we note that the homogeneous Dirichlet boundary condition is maintained from (4.4). In fact, any boundary conditions specified for the nonlinear problem become homogeneous for the arising linearisation. This is an advantage of the Newton method over FAS, as the treatment of difficult boundary conditions is made a lot simpler in the homogeneous case.

We now consider the space in which a solution to (4.5) exists, if at all. Since the nonlinearity is in the highest order derivative, if a weak solution exists

it must exist in the Sobolev space $W_0^{1,p}(\Omega)$ for some $p > 2$ [28] (see [11, pg. 235-6] for a detailed discussion). As is usual $W_0^{1,p}(\Omega)$ is the set of functions in $W^{1,p}$ with compact support in Ω. To the best of our knowledge there are no fixed-step techniques that can guarantee convergence in the $W_0^{1,p}$ norm. In [28], the convergence of the Newton iteration is proved in this case, but a special iterative operator needs to be constructed in order to make the energy norm equivalent to the norm in some Hilbert space [28, §5]. In section 6 we demonstrate that when using a standard geometric multigrid algorithm we can observe mesh independent convergence of a Newton-MG method, such that we can avoid the construction of these linear operators. To the authors' knowledge there is no theory that proves convergence in this case, which is the most simple to implement, often giving optimal results.

4.3. Nonlinear Multigrid (FAS)

There are very few rigorous results regarding the convergence of nonlinear multigrid methods, in particular the FAS scheme. An almost complete bibliography on the matter is given by [22, 24, 25, 40, 41, 46, 51], although most of these deal with the case of the Nonlinear Multilevel Method (NMLM) proposed by Hackbusch [22], which is a generalisation of the FAS scheme. The most complete results in the field were obtained by Reusken and Hackbusch [25, 40, 41], in which convergence was proven for a class of nonlinear elliptic problems where the nonlinearity occurs in the zero order term. To the best of our knowledge there exists no valid theory for the convergence of FAS for the case where the nonlinearity is in the highest order term. This is due, in part, to the fact that it is not known how to show that the method is convergent in the natural $W^{1,p}$ norm for $p > 2$. The theory by Xie [51] and Hackbusch [22] are set in Banach spaces, but the assumptions required to be satisfied are not easy to show in the case where the nonlinearity is in the highest order derivative.

Whilst there does not exist any theory for FAS in the case where the nonlinearity is in the highest order derivative, there does exist some convergence theory for the case of nonlinear iterations (such as nonlinear Gauss-Seidel and Jacobi iterations) [38] which could give a qualitative insight into the convergence behaviour of an FAS iteration. In [38, §10] it is shown that the convergence of nonlinear Jacobi (Gauss-Seidel) iterations is bounded by the convergence of an inexact Newton iteration using linear Jacobi (Gauss-Seidel) iterations. A condition for the convergence, therefore, of the nonlinear iterations is that a Newton method should be convergent. Since FAS is a combination of nonlinear smooths it seems sensible to assume that a similar result could hold for FAS. This conjecture is supported by results given in section 6.

5. Model Problems

The problems for which empirical results are presented in section 6 are introduced in this section, with a short discussion and justification as to why they are considered. All of these model problems are defined on some $\Omega \subset \mathbb{R}^2$.

5.1. 4-Laplacian

The p-Laplacian operator is given by

$$\begin{aligned} -\nabla \cdot \left(|\nabla u|^{p-2}\nabla u\right) = f(x), \quad x \in \Omega, \\ u = 0, \quad x \in \partial\Omega. \end{aligned} \tag{5.1}$$

In this paper we concentrate on the case $p = 4$ as this example allows us to highlight differences between Newton-MG and FAS methods. Qualitatively similar results are observed in numerical experiments for $p > 4$, so the results are not presented here. A ball of guaranteed convergence is found to shrink as p is increased. A more careful treatment is required in the discretisation of the problem for $p \in (1, 2)$ than will be presented here, cf. [15]. Existence and uniqueness of solutions for the p-Laplacian are known for $p \in (1, \infty)$ [15]. The weak formulation of the problem is given by

$$\begin{aligned} \mathcal{F}(u, v) \equiv \int_\Omega |\nabla u|^2 \nabla u \nabla v \, \mathrm{d}x - \int_\Omega fv \, \mathrm{d}x = 0, \quad x \in \Omega, \\ u = 0, \quad x \in \partial\Omega \end{aligned} \tag{5.2}$$

and the Frèchet derivative by

$$\mathcal{F}_u(w, v) = \int_\Omega |\nabla u|^2 \nabla w \nabla v \, \mathrm{d}x + \int_\Omega 2(\nabla u \nabla w)(\nabla u \nabla v) \, \mathrm{d}x. \tag{5.3}$$

By inspection of (5.3) we see that $\mathcal{F}_u(w, v)$ gives a symmetric positive definite bilinear form. Using the linear multigrid theory we can therefore say that under mild assumptions a standard geometric linear multigrid method will be convergent for the linear inner iteration, as is supported by empirical results in section 6.

Three perturbations of the 4-Laplacian are presented in this paper, given by

$$-\nabla \cdot \left(\alpha |\nabla u|^2 \nabla u\right) = f \tag{5.4a}$$

$$-\nabla \cdot \left(\left\{1 + \alpha |\nabla u|^2\right\} \nabla u\right) = f \tag{5.4b}$$

$$-\nabla \cdot \left(\alpha \left\{1 + |\nabla u|^2\right\} \nabla u\right) = f \tag{5.4c}$$

for piecewise constant function $\alpha(x) > 0$, $x \in \bar{\Omega}$, which may be highly varying. The domain considered is the unit square $\Omega \equiv (0, 1)^2$. The model problems in (5.4b) and (5.4c) include a linear Laplacian term (and hence the Jacobian is still symmetric positive definite), and are used to show the convergence behaviour of a Newton-MG iteration as the size of the nonlinearity grows compared to the linear term.

5.2. Porous Medium Equation

The porous medium equation is given by [36]

$$\begin{aligned} u_t = \nabla \cdot \left(u^m \nabla u\right), \quad x \in \Omega \equiv (-1, 1)^2, \quad t > t_0, \\ u = 0, \quad x \in \partial\Omega, \\ u(t_0) = u_0. \end{aligned} \tag{5.5}$$

For a discussion on the finite element formulation see [1]. In this paper we consider the case of $m = 2$.

Equation (5.5) may be discretised in time using a single-step implicit method such as backward Euler or Crank-Nicolson to give (at each time-step) an equation of the form

$$u^{(n+1)} - \frac{\Delta t}{K}\left(\nabla \cdot ((u^{(n+1)})^2 \nabla u^{(n+1)})\right) = f(u^{(n)}), \tag{5.6}$$

for $u^{(n)} = u(\,\cdot\,, t_n)$, $t_n = t_0 + n\Delta t$ for fixed Δt. The right-hand side $f(u^{(n)})$ and the constant K depend on the time discretisation scheme that is used. Treating the function $u^{(n+1)}$ as the unknown variable and considering $f(u^{(n)})$ as some given right-hand side we can consider equation (5.6) to be of the form

$$u - C\nabla \cdot \left(u^2 \nabla u\right) = f, \tag{5.7}$$

for unknown u, known f, and constant $C > 0$ having the same effect as the time-step in (5.6). This is the form of equation to which a nonlinear multigrid method is applied in an implicit time discretisation. Hence the evaluation of how the nonlinear multigrid methods will behave for the solution of a porous medium equation with an implicit time discretisation requires only to evaluate how each nonlinear method operates for the problem (5.7) with homogeneous Dirichlet boundary conditions.

We consider the weak form of (5.7), given by

$$\mathcal{F}(u, v) \equiv \int_\Omega uv \ \mathrm{d}x + C \int_\Omega \alpha u^2 \nabla u \nabla v \ \mathrm{d}x - \int_\Omega fv \ \mathrm{d}x \tag{5.8}$$

and the Frèchet derivative

$$\mathcal{F}_u(w, v) = \int_\Omega wv \ \mathrm{d}x + C \int_\Omega 2uw\nabla u \nabla v \ \mathrm{d}x + C \int_\Omega u^2 \nabla w \nabla v \ \mathrm{d}x. \tag{5.9}$$

There is a symmetric positive definite and a non-symmetric part of the Frèchet derivative given by

$$\mathcal{F}_u^{\mathrm{s}}(w, v) = \int_\Omega wv \ \mathrm{d}x + C \int_\Omega u^2 \nabla w \nabla v \ \mathrm{d}x, \tag{5.10}$$

$$\mathcal{F}_u^{\mathrm{ns}}(w, v) = C \int_\Omega 2uw\nabla u \nabla v \ \mathrm{d}x. \tag{5.11}$$

From the discussion of the convergence of standard geometric multigrid algorithms in section 4, we should find that, so long as $\|\mathcal{F}_u^{\mathrm{s}}\| \gg \|\mathcal{F}_u^{\mathrm{ns}}\|$, a standard linear multigrid algorithm will be convergent for the inner iteration. We note that this condition can be controlled by ensuring that C is small enough, and hence we can always obtain a convergent iteration. Results provided in section 6 support these statements. We note that the form of the non-symmetric part is similar to the inclusion of the function $b(x)$ in (4.4), which contributes to the non-symmetric part of the linearisation. This does not change the condition $\|\mathcal{F}_u^{\mathrm{s}}\| \gg \|\mathcal{F}_u^{\mathrm{ns}}\|$ from the above discussion, but may require a smaller time-step to be taken in order to obtain a convergent nonlinear iteration.

6. Empirical Results

In this section we present results demonstrating the superiority of the Newton-MG iteration over the FAS iteration. For the results presented the coarsest grid used for all experiments was a uniform 4×4 grid, with 9 interior nodes and 32 triangular elements.

6.1. Running time of Newton-MG vs FAS

In tables 1 and 2 we contrast the predicted with the actual running time to perform one hundred nonlinear iterations (either Newton iterations or FAS V-cycles). We show the predictions for varying numbers of linear V-cycles per Newton iteration (Table 1) and varying number of pre- and post-smooths for the FAS V-cycle (Table 2). The timing results show the cost of performing the nonlinear iterations and disregard the amount of time taken to initialise the algorithms (e.g. calculation of the right-hand sides for the nonlinear equation, grid set-up, memory (de-)assignment, etc.). A single work unit W_l was taken as the average time to perform one thousand residual calculations (note that the choice of coefficient $\alpha(x)$ has no effect on the running time). The grid for which the results are presented is a regular 512×512 triangular grid. The work unit for the 4-Laplacian was found to be 0.075s, and for the PME 0.079s, on this grid. These values were used in equations (3.19) and (3.18) to calculate the approximate execution time (denoted $\tilde{C}_{\mathrm{NMG}}$ for Newton-MG and C_{FAS} for FAS multigrid).

Linear MG-Cycles	Cost					
	4-Laplacian				PME	
	$\tilde{C}_{\mathrm{NMG}}$	(5.4a)	(5.4b)	(5.4c)	$\tilde{C}_{\mathrm{NMG}}$	(5.7)
1	51.25	49.61	49.93	49.45	53.98	44.11
2	62.50	59.62	60.30	60.53	65.83	56.01
3	73.75	70.84	72.53	72.06	77.68	69.43
4	85.00	82.06	82.48	84.32	89.53	79.57

Table 1: Predicted and actual running times (s) for the execution of 100 Newton iterations with varying numbers of linear V-cycles applied to the model problems from section 5

As can be seen in table 1, the theoretical value is remarkably close for the Newton-MG iteration applied to the 4-Laplacian equation. The bound is more pessimistic for the porous medium equation (PME), which can be explained by the implementation. For the PME we made more use of storing values to be re-used in the calculation of the Jacobian matrix than for the 4-Laplacian problem. Hence the calculation of the Jacobian matrix does not take the expected factor four times longer to be calculated. This effect becomes more pronounced if larger savings can be made from the re-use of calculations. More complex operators require more floating point operations to be performed per residual calculation and so it is more likely that more effective use can be made of reusing previous calculations. Therefore the more complicated the nonlinear operator the more pessimistic the approximation will be.

# of Smooths	Cost					
	4-Laplacian				PME	
	C_{FAS}	(5.4a)	(5.4b)	(5.4c)	C_{FAS}	(5.7)
1	52.5	48.43	47.48	46.78	55.30	52.41
2	92.5	82.18	84.24	81.43	97.43	90.94
3	132.5	120.21	120.63	118.50	139.57	135.40
4	172.5	154.23	151.07	150.05	181.70	172.23

Table 2: Predicted and actual running times (s) for the execution of 100 FAS V-cycles with varying numbers of smooths applied to the model problems from section 5

The results in table 2 show that the theoretical bound is not as accurate for the FAS iteration for the 4-Laplacian as it is for Newton-MG, but the estimate is better for the PME. The estimates for the 4-Laplacian are pessimistic because the cost of calculating the Jacobian diagonals is less than double the cost of calculating the residual for the 4-Laplacian, whereas it is much closer to double for the PME.

In the calculation of the linearisation we note that we perform more calculations per element in Newton-MG than in FAS and so reuse calculated values more often. Therefore there is a greater opportunity for computational savings naturally available for inexact Newton iterations which is not the case for FAS. The theoretical bound is therefore likely to be approximated more sharply in the case of FAS the more complicated an operator becomes. This highlights that the Newton-MG is likely to become faster (per iteration) compared to FAS for more complex nonlinear operators.

In section 3 the prediction was made that the ratio between the per V-cycle execution time for FAS compared to Newton-MG, when performing three linear V-cycles per Newton iteration and three pre- and post-smoothing steps per FAS V-cycle, would be approximately 5.4. Let $C_{\text{NMG}}^{(4\text{L})}$ ($C_{\text{NMG}}^{(\text{PME})}$) represent the actual cost for the 4-Laplacian problem (5.4a) (PME) per linear V-cycle using three linear iterations per Newton step, and $C_{\text{FAS}}^{(4\text{L})}$ ($C_{\text{FAS}}^{(\text{PME})}$) the actual cost for the 4-Laplacian problem (5.4a) (PME) using three pre- and post-smoothing iterations in the FAS iteration. Then

$$\frac{C_{\text{FAS}}^{(4\text{L})}}{C_{\text{NMG}}^{(4\text{L})}} \approx 5.09 \quad \text{and} \quad \frac{C_{\text{FAS}}^{(\text{PME})}}{C_{\text{NMG}}^{(\text{PME})}} \approx 5.85.$$

The running times of a single iteration of each of the algorithms does not tell us about the convergence behaviour of the algorithms. In the next subsections we present results for the FAS and Newton-MG iterations that demonstrate the stability and efficiency of the algorithms.

6.2. 4-Laplacian

We consider the solution of the 4-Laplacian problems (5.4a), (5.4b) and (5.4c) on the domain $\Omega \equiv (0,1)^2$. The coefficient function α is piecewise constant, and the distribution of the coefficient on the domain is shown in figure 1. The right-hand side is chosen such that the exact solution is $u^* = \sin(\pi x)\sin(\pi y)$

when $\alpha(x) \equiv 1.0$. The results in this section are presented where the initial approximation $u^{(0)} = 0.7u^*$. A note is made at the end what effect using a nested iteration (cf. [12, 48]) to obtain an initial approximation on the finest grid has on the convergence of the two methods. The coarsest uniform grid on which the discontinuities in the coefficient function are fully captured (i.e. all discontinuities coincide with element edges) is a 64×64 grid. In the bulk of the domain the function α takes some constant value C, and in the shaded regions we fix $\alpha = 1.0$. From the theory in [44] we know that convergence of a linear multigrid iteration will not be independent of the size of the jump in the coefficient when a local *quasi-monotonicity* property [44] fails to hold on coarse grid levels. In the example given this occurs for the case where $C < 1$. In the case where $C > 1$ the convergence of a linear multigrid iteration will be independent of the jumps in the coefficient. We should therefore find that convergence of the Newton-MG iteration deteriorates for decreasing $C < 1$, and may find that the convergence is independent of the size of the jumps in the coefficient for the case $C > 1$. There exists no theory to suggest how FAS should perform.

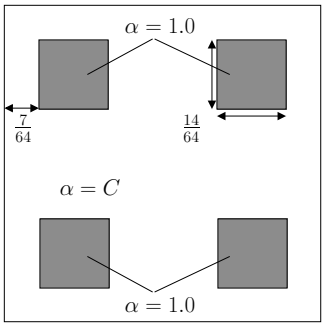
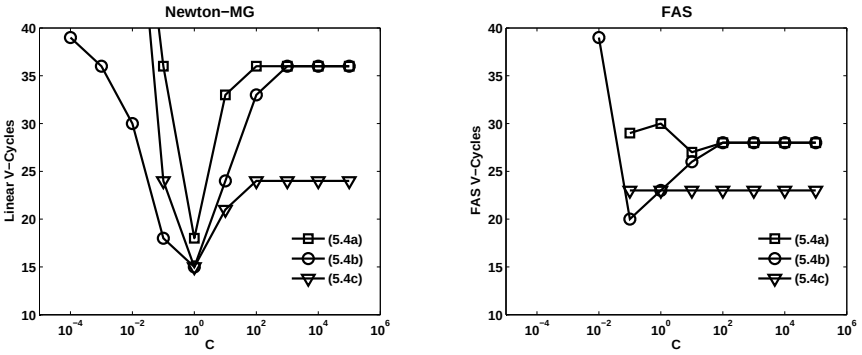


Figure 1: Sample distribution of α on the domain $\Omega \equiv (0,1)^2$

Figures 2a and 2b show the number of V-cycles required to reduce the initial residual in the approximation by a factor of 10^{-7}. Absence of any point from the graphs indicates non-convergence. In figure 2a convergence is not obtained for problems (5.4a) and (5.4c) for values of $C < 10^{-2}$. As predicted by the linear theory the Newton-MG algorithm breaks down for decreasing $C < 1$. For values of $C > 1$ we find that the convergence of the nonlinear iteration is independent of the size of the jump in the coefficient. We see similar behaviour for the FAS multigrid algorithm, except that the algorithm is significantly less stable for small values of C. For larger values of C the number of V-cycles required for convergence is comparable in the two cases.

A representative set of running times on different grids for problem (5.4b), using the distribution of α given in figure 1 with $C = 10^5$, is shown in figure 3a. The algorithms are optimal in terms of running time, scaling linearly with respect to the number of unknowns in the system. From the graphs in figure 2a and 2b we see that for this problem we require 28 FAS iterations and 36 linear multigrid iterations for convergence. Using the approximate running times of the iterations we predict a 4.2 times speed-up using the Newton-MG over the

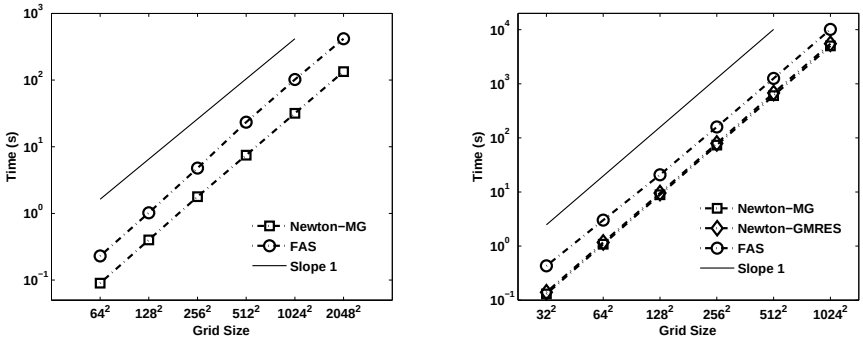


(a) Number of linear V-cycles for convergence of 4-Laplacian type problems with α as in figure 1

(b) Number of FAS V-cycles for convergence of 4-Laplacian type problems with α as in figure 1

Figure 2: Convergence of Newton-MG and FAS multigrid cycles for 4-Laplacian type problems on a 512×512 grid

FAS. In practice we find that (due to the fact that the implementation requires time to set up various approximations; (de-)allocate memory; perform I/O; and other operations) the actual speed up for the Newton-MG iteration is 3.12. For problems in which more Newton / FAS iterations need to be performed we find that the execution time of the nonlinear iterations will begin to dominate the overall running time and we will see a move towards the predicted improvement in running times.



(a) Running time of 4-Laplacian for Newton-MG and FAS multigrid for the case $C = 10^5$ in figure 1

(b) Execution time for FAS and Newton-MG for the PME with time-step $\Delta t = h$ for grid spacing h

Figure 3: Comparison of execution times for model problems

We note that, using the theory from [44], we can define situations in which Newton-MG methods will not converge. Using different configurations of the piecewise constant function α on the domain, including situations in which the geometry is more complex, the results remain qualitatively the same as presented here, so long as the initial approximation to the problem is close enough to the exact solution and a local quasi-monotonicity property holds on all grids. The

convergence behaviour of the FAS multigrid iteration seems to be very similar to that of the Newton-MG iteration in the case of using pointwise (non)linear smoothers.

We make a note on how the two methods are applied in a *nested iteration* (cf. [12, 48]), as this is often useful in practice to obtain an initial approximation on a fine grid. In the nested iteration, approximations are interpolated from coarse grids to finer grids, where a number of iterations of some iterative solution method are performed on each grid in order to obtain a 'good' initial approximation to the solution on the next finest grid. It is well known that in practice the convergence of a Newton iteration may not be monotonic, unless an approximation is in a ball of guaranteed convergence. On the other hand the authors have not observed a situation in which the FAS iteration does not converge monotonically (assuming it converges). FAS can be used effectively within a nested iteration, with often only a single FAS iteration required to give a 'good' approximation on the next finest grid. However, it may be the case that several Newton iterations are required to obtain the same reduction in the residual gained by the first FAS iteration. Hence the relative cost of performing a nested iteration with the FAS algorithm may be less than performing a nested iteration with Newton's method. However, it is still expected that the running time of a Newton iteration is superior to an FAS iteration if a nested approximation is used.

The problem of finding a 'good' initial approximation may be of some concern for the case of an elliptic type problem, such as the p-Laplacian, but is not important in the time dependent setting, where a good initial approximation may be gained using information from the solution at the previous time-step. Results are presented in the next section demonstrating the superiority of a Newton-MG iteration over the FAS iteration for a time dependent problem.

6.3. Porous Medium Equation (PME)

In this section we present results for Newton-MG, FAS and a Newton method where the linear system is approximated by a multigrid preconditioned GMRES iteration (Newton-GMRES) applied to the PME for the case where the solution is a self-similar solution of Barenblatt [3]

$$\begin{aligned} u(x,t) &= \frac{1}{\mu^2}\left[\max\left\{1-\left(\frac{r}{r_0\mu}\right)^2, 0\right\}\right]^{\frac{1}{m}}, \\ r = \sqrt{x_1^2+x_2^2}, \quad \mu &= \left(\frac{t}{t_0}\right)^{\frac{1}{2(1+m)}}, \quad t_0 = \frac{r_0^2 m}{4(1+m)}, \end{aligned} \tag{6.1}$$

setting $m = 2$. We solve this on the domain $\Omega \equiv (-1,1)^2$, and impose homogeneous Dirichlet boundary conditions. The solution given is non-negative and has an infinite gradient at the boundary of the support of the function, which means that the gradient of the discrete approximations will tend to infinity with decreasing grid spacing. We cannot expect the convergence of a nonlinear multigrid iteration to be independent of the grid spacing in this case, but in [16, 45]

it is shown that, for a Lipschitz continuous solution u to the porous medium equation, the convergence rate deteriorates proportional to the grid spacing. To prevent the error in the time derivative from dominating the time step should be taken proportional to the grid spacing, and hence the nonlinear multigrid methods remain of optimal order. We note that (6.1) is not Lipschitz continuous, but we still observe the same optimal behaviour (cf. table 3 and figure 3b). The restriction on the time-step also holds in the case of FAS, and the maximum allowable time-step (i.e. the largest time-step for which each nonlinear method gives a convergent iteration) for the FAS is bounded by the maximum allowable time-step for a corresponding Newton iteration. This supports the conjecture put forward in section 4.3 that convergence of the FAS iteration is dependent on the convergence of a Newton iteration.

Grid Size	Backward Euler			Crank-Nicolson		
	MG	FAS	GMRES	MG	FAS	GMRES
32×32	0.255	0.101	>10	0.348	0.111	>10
64×64	0.0930	0.0389	>10	0.0828	0.0560	6.74
128×128	0.0189	0.0147	4.01	0.0215	0.0158	0.544
256×256	0.00732	0.00599	0.242	0.00968	0.00702	0.138
512×512	0.00340	0.00282	0.0353	0.00452	0.00311	0.0423
1024×1024	0.00140	0.00120	0.0114	0.00193	0.00134	0.0145
2048×2048	0.000669	0.000627	0.00417	0.000924	0.000702	0.00568

Table 3: Maximum allowable time-step (s) for FAS and inexact Newton iterations for the PME with exact solution given by (6.1) with $r_0 = 0.3$

Results are also presented for a multigrid preconditioned Newton-GMRES iteration. The preconditioner is a single V-cycle applied to the symmetric part of the Jacobian. From table 3 we see that the GMRES iteration gives a more stable algorithm. From figure 3b we see that there is very little extra cost in performing the GMRES iteration for a large increase in the stability of the method. The results for the multigrid-preconditioned GMRES iteration are included to demonstrate that it is relatively simple to increase the stability of an inexact Newton method by utilising a more suitable linear inner iteration. This flexibility is not present for the FAS multigrid. Even changing the smoother from the pointwise Jacobi to a pointwise Gauss-Seidel gives a large increase in the execution time of the algorithm.

In order to model materials with different diffusivities on the domain we may introduce a piecewise constant coefficient function, similar to that used in the 4-Laplacian (cf. (5.4a)). The problem to solve is then given by

$$u_t = \nabla \cdot (\alpha u^m \nabla u) \,.$$

For increasing α the linearisation becomes dominated by the non-symmetric part, and convergence of the inner iteration, and hence the outer iteration, is no longer independent of the size of α. The convergence is independent of the size of the jumps in α, though, and we can always obtain a convergent iteration by using a small enough time-step to ensure that the symmetric time derivative

term dominates. The distribution of the coefficient α on the domain again has the same effect as described for the 4-Laplacian (results not shown), and as predicted in [44].

The execution time of the algorithms is optimal with regard to the grid spacing (cf. figure 3b) with the execution time of the Newton-MG being quicker than that of the FAS iteration. On a 512×512 grid and 1024×1024 grid the execution time for the Newton-MG iteration has stabilised to be a little more than twice as fast as the FAS iteration.

7. Conclusions

The results presented here suggest that a Newton-MG method should be preferred over an FAS iteration for reasons of:

i *Efficiency:* The Newton-MG iteration may be much more efficient than an FAS iteration. In particular the theory in section 3 shows that as a nonlinear operator becomes more complicated, the cheaper the Newton iteration becomes with respect to a work unit W_l on a grid Ω_l. This does not hold true for the FAS iteration. We also observe a faster execution time for Newton-MG in practice.

ii *Stability:* The stability of the Newton method appears slightly better than that of an FAS iteration and numerical experiment supports a conjecture made in section 4 that the convergence of an FAS iteration requires the convergence of a Newton iteration.

iii *Flexibility:* There is far greater flexibility to choose components in the algorithm for an inexact Newton method. In particular, FAS is a solver for elliptic and parabolic problems. On the other hand Newton's method may be used in conjunction with a suitable linear iteration that is preconditioned by multigrid. This may increase robustness and can provide a basis for solving a wider range of PDEs.

   In any multigrid algorithm the operator that is most important for convergence of the algorithm is the smoothing operator. Changing the smoother in the linear case does not require frequent updates of the linear operator or residual, and hence one incurs less heavy cost penalties when choosing more sophisticated smoothers. In the nonlinear setting, even choosing a nonlinear pointwise Gauss-Seidel iteration as a smoother requires a considerable amount of work to update the nonlinear residual after each pointwise update. Hence, for the FAS, a faster execution time, relative to a work unit, is observed for simple problems in which a pointwise Jacobi smoother is appropriate.

iv *Theoretical Grounding:* For the FAS iteration there is very little theory available to inform the choice of components in forming a 'good' iteration for a specific problem. Conversely, there exists a wealth of literature for Newton iterations and linear multigrid iterations separately. As demonstrated, this

theory can be used to infer how a Newton-MG iteration will perform for a nonlinear problem. Sophisticated algorithms may be developed by taking advantage of the knowledge of these two methods individually, such as the truncated non-smooth Newton multigrid [20]. In addition, there are no spectral variants of the FAS such as algebraic multigrid (cf. [48]), and no methods such as monotone and truncated multigrid methods [30, 31] which can be used to take advantage of certain properties of the underlying operator, or to overcome difficulties presented by the underlying operator.

There are some situations, however, in which the FAS iteration may be of more use than a Newton iteration. The memory requirements are less for the FAS iteration than for a Newton iteration, as the (large) sparse Jacobian matrices do not need to be stored in memory [33, 34, 47]. In situations in which memory requirements are an issue Jacobian-free Newton-Krylov (JFNK) methods [29] provide a less memory-intensive way of implementing an approximate Newton-Krylov method. Implementations include multigrid preconditioned GMRES variants, e.g. [19, 39]. However, JFNK methods do not remove the need to store at least a preconditioning matrix (if preconditioning is to be performed). For very large problem sizes it could be the case that there is not enough memory available to perform an approximate Newton iteration, but there is for an FAS iteration. In this situation an FAS iteration would be preferred.

The conclusions drawn here have been shown to hold in the case of simple systems of equations (cf. [33–35]), but should hold for more complex systems of equations as well. The estimates for the relative execution times can be used for adaptively refined grids and hierarchies in which the grids are non-overlapping, so long as the number of points on each grid is estimated accurately enough.

This paper gives a framework for assessing the performance of nonlinear multigrid methods in the case of a scalar nonlinear equation implemented sequentially. It would be interesting to observe in more detail how the algorithms perform in the case of systems of equations and implementations in parallel, which can be the basis for future investigations.

8. Acknowledgements

We would like to thank Robert Scheichl and Eike Müller for valuable discussions, in particular regarding the theory of multigrid methods. The first author acknowledges the EPSRC for their financial support.

[1] M. Baines, M. Hubbard, P. Jimack, A. Jones, Scale-invariant moving finite elements for nonlinear partial differential equations in two dimensions, Applied Numerical Mathematics 56 (2006) 230–252.

[2] R. Bank, D. Rose, Analysis of a multilevel iterative method for nonlinear finite element equations, Mathematics of Computation 39 (1982) 453–465.

[3] G. Barenblatt, Scaling, self-similarity, and intermediate asymptotics: dimensional analysis and intermediate asymptotics, volume 14 of *Cambridge Texts in Applied Mathematics*, Cambridge University Press, 1996.

[4] P. Bastian, O. Ippisch, F. Rezanezhad, H. Vogel, K. Roth, Numerical simulation and experimental studies of unsaturated water flow in heterogeneous systems, in: W. Jäger, R. Rannacher, J. Warnatz (Eds.), Reactive Flows, Diffusion and Transport, Springer Berlin / Heidelberg, 2005.

[5] D. Braess, Finite Elements: Theory, fast solvers, and applications in solid mechanics, second ed., Cambridge University Press, 2001.

[6] J. Bramble, D. Kwak, J. Pasciak, Uniform convergence of multigrid V-cycle iterations for indefinite and nonsymmetric problems, SIAM Journal of Numerical Analysis 31 (1994) 1746–1763.

[7] J. Bramble, J. Pasciak, J. Wang, J. Xu, Convergence estimates for multigrid algorithms without regularity assumptions, Mathematics of Computation 57 (1991) 23–45.

[8] J. Bramble, J. Pasciak, J. Wang, J. Xu, Convergence estimates for product iterative methods with applications to domain decomposition, Mathematics of Computation 57 (1991) 1–21.

[9] A. Brandt, Multilevel adaptive solutions to boundary value problems, Mathematics of Computation 31 (1977) 333–390.

[10] A. Brandt, Rigorous local mode analysis of multigrid, in: Preliminary Proc. 4th Copper Mountain Conf. on Multigrid Methods, Copper Mountain, Colorado.

[11] S. Brenner, L. Scott, The Mathematical Theory of Finite Element Methods, number 15 in Texts in Applied Mathematics, 3rd ed., Springer, 2008.

[12] W. Briggs, V. Henson, S. McCormick, A Multigrid Tutorial, 2nd ed., Society for Industrial and Applied Mathematics, 2000.

[13] J. Brown, B. Smith, A. Ahmadia, Achieving textbook multigrid efficiency for hydrostatic ice sheet flow, SIAM Journal on Scientific Computing 35 (2013) B359–B375.

[14] P. Deuflhard, Newton Methods for Nonlinear Problems: Affine Invariance and Adaptive Algorithms, number 35 in Springer Series in Computational Mathematics, Springer Berlin / Heidelberg, 2004.

[15] L. Diening, C. Kreuzer, Linear convergence of an adaptive finite element method for the p-laplacian equation, SIAM Journal on Numerical Analysis 46 (2008) 614–638.

[16] M. Donatelli, M. Semplice, S. Serra-Capizzano, Analysis of multigrid preconditioning for implicit PDE solvers for degenerate parabolic equations, SIAM Journal on Matrix Analysis and Applications 32 (2011) 1125–1148.

[17] M. Donatelli, M. Semplice, S. Serra-Capizzano, AMG preconditioning for nonlinear degenerate parabolic equations on nonuniform grids with application to monument degradation, Applied Numerical Mathematics 68 (2013) 1–18.

[18] P. Gaskell, P. Jimack, M. Sellier, H. Thompson, Efficient and accurate time-adaptive multigrid simulation of droplet spreading, International Journal for Numerical Methods in Fluids 45 (2004) 1161–1186.

[19] L. Ge, F. Sotiropoulos, A numerical method for solving the 3D unsteady incompressible navier-stokes equations in curvilinear domains with complex immersed boundaries, Journal of Computational Physics 225 (2007) 1782–1809.

[20] C. Gräser, R. Kornhuber, Multigrid methods for obstacle problems, Journal of Computational Mathematics 27 (2009) 1–44.

[21] J. Green, P. Jimack, A. Mullis, J. Rosam, An adaptive, multilevel scheme for the implicit solution of three-dimensional phase-field equations, Numerical Methods for Partial Differential Equations 27 (2011) 106–120.

[22] W. Hackbusch, Multi-Grid Methods and Applications, volume 4 of *Springer Series in Computational Mathematics*, Springer-Verlag, 1985.

[23] W. Hackbusch, Comparison of different multi-grid variants for nonlinear equations, Zeitschrift für Angewandte Mathematik und Mechanic 72 (1992) 148–151.

[24] W. Hackbusch, A. Reusken, On global multigrid convergence for nonlinear problems, in: W. Hackbusch (Ed.), Robust Multi-Grid Methods, Notes on Numerical Fluid Dynamics, Vieweg, Braunschweig, 1988, pp. 105–113.

[25] W. Hackbusch, A. Reusken, Analysis of a damped nonlinear multilevel method, Numerische Mathematik 55 (1989) 225–246.

[26] V. Henson, Multigrid methods for nonlinear problems: An overview, in: Proceedings of the Society of Photo-Optical Instrumentation Engineers (SPIE), pp. 36–48.

[27] G. Jouvet, C. Gräser, An adaptive Newton multigrid method for a model of marine ice sheets, Journal of Computational Physics 252 (2013) 419–437.

[28] T. Kim, J. Pasciak, P. Vassilevski, Mesh-independent convergence of the modified inexact Newton method for a second order non-linear problem, Numerical Linear Algebra with Applications 13 (2006) 23–47.

[29] D. Knoll, D. Keyes, Jacobian-free Newton-Krylov methods: a survey of approaches and applications, Journal of Computational Physics 193 (2004) 357–397.

[30] R. Kornhuber, Monotone multigrid methods for elliptic variational inequalities I, Numerische Mathematik 96 (1994) 167–184.

[31] R. Kornhuber, Monotone multigrid methods for elliptic variational inequalities II, Numerische Mathematik 72 (1996) 481–499.

[32] S. MacLachlan, C. Oosterlee, Local Fourier analysis for multigrid with overlapping smoothers applied to systems of PDEs, Numerical Linear Algebra with Applications 18 (2011) 751–774.

[33] D. Mavriplis, Multigrid approaches to non-linear diffusion problems on unstructured meshes, Numerical Linear Algebra with Applications 8 (2001) 499–512.

[34] D. Mavriplis, An assessment of linear versus non-linear multigrid methods for unstructured mesh solvers, Journal of Computational Physics 175 (2002) 302–325.

[35] J. Molenaar, Multigrid Methods for Fully Implicit Oil Reservoir Simulation, Technical Report, TWI, Delft University of Technology, 1995.

[36] J. Murray, Mathematical Biology: An Introduction, 3rd ed., Springer, 2002.

[37] A. Napov, Y. Notay, Smoothing factor, order of prolongation and actual multigrid convergence, Numerische Mathematik 118 (2011) 457–483.

[38] J. Ortega, W. Rheinboldt, Iterative Solution of Nonlinear Equations in Several Variables, Computer Science and Applied Mathematics, Academic Press, 1970.

[39] H. Park, D. Knoll, R. Rauenzahn, C. Newman, J. Densmore, A. Wollaber, An efficient and time accurate, moment-based scale-bridging algorithm for thermal radiative transfer problems, SIAM Journal on Scientific Computing 35 (2013) S18–S41.

[40] A. Reusken, Convergence of the multigrid full approximation scheme for a class of elliptic mildly nonlinear boundary value problems, Numerische Mathematik 52 (1988) 251–277.

[41] A. Reusken, Convergence of the multilevel full approximation scheme including the V-cycle, Numerische Mathematik 53 (1988) 663–686.

[42] J. Rosam, P. Jimack, A. Mullis, An adaptive, fully implicit, multigrid phase-field model for the quantitative simulation of non-isothermal binary alloy solidification, Acta Mathematica 56 (2008) 4559–4569.

[43] Y. Saad, Iterative Methods for Sparse Linear Systems, 2nd ed., Society for Industrial and Applied Mathematics, 2003.

[44] R. Scheichl, P. Vassilevski, L. Zikatanov, Multilevel methods for elliptic problems with highly varying coefficients on non-aligned coarse grids, SIAM Journal on Numerical Analysis 50 (2012) 1675–1694.

[45] M. Semplice, Preconditioned implicit solvers for nonlinear PDEs in monument conservation, SIAM Journal on Scientific Computing 32 (2010) 3071–3091.

[46] V. Shaidurov, Multigrid Methods for Finite Elements, 2nd ed., Kluwer Academic Publishers, 1995.

[47] L. Stals, Comparison of non-linear solvers for the solution of radiation transport equations, Electronic Transactions on Numerical Analysis 15 (2003) 78–93.

[48] U. Trottenberg, C. Oosterlee, A. Schüller, Multigrid, Academic Press, 2001.

[49] E. Vecharynski, A. Knyazev, Absolute value preconditioning for symmetric indefinite linear systems, SIAM Journal on Scientific Computing 35 (2013) A696–A718.

[50] R. Wienands, W. Joppich, Practical Fourier Analysis for Multigrid Methods, Chapman and Hall/CRC, 2005.

[51] D. Xie, New nonlinear multigrid analysis, in: Seventh Copper Mountain Conference on Multigrid Methods, pp. 793–808.

[52] J. Xu, Iterative methods by space decomposition and subspace correction, SIAM Review 34 (1992) 581–613.