3

Multiscale finite element methods for nonlinear equations

3.1 MsFEM for nonlinear problems. Introduction

The objective of this chapter is to present a generalization of the MsFEM to nonlinear problems ([110, 112, 113, 104]) which was first presented in [110]. This generalization, as the MsFEM for linear problems, has two main ingredients: a global formulation and multiscale localized "basis" functions. We discuss numerical implementation issues and applications.

Let $\mathcal{T}_h$ be a coarse-scale partition of Ω, as before. We denote by W_h a usual finite-dimensional space, which possesses approximation properties, for example, piecewise linear functions over triangular elements, as defined before. In further presentation, K is a coarse element that belongs to $\mathcal{T}_h$. To formulate MsFEMs for general nonlinear problems, we need (1) a multiscale mapping that gives us the desired approximation containing the small-scale information and (2) a multiscale numerical formulation of the equation.

We consider the formulation and analysis of MsFEMs for general nonlinear elliptic equations

$$-\operatorname{div} k(x,p,\nabla p) + k_0(x,p,\nabla p) = f \ \text{ in } \Omega, \qquad p = 0 \ \text{ on } \partial\Omega, \tag{3.1}$$

where $k(x,\eta,\xi)$ and $k_0(x,\eta,\xi)$, $\eta \in \mathbb{R}$, $\xi \in \mathbb{R}^d$ satisfy the general assumptions (6.42)–(6.46), which are formulated later. Note that here k and k_0 are nonlinear functions of p as well as ∇p. Moreover, both k and k_0 are heterogeneous spatial fields. Later, we extend the MsFEM to nonlinear parabolic equations where k and k_0 are also heterogeneous functions with respect to the time variable.

Multiscale mapping. Unlike MsFEMs for linear problems, "basis" functions for nonlinear problems need to be defined via nonlinear maps that map coarse-scale functions into fine-scale functions. We introduce the mapping $E^{MsFEM} : W_h \to \mathcal{P}_h$ in the following way. For each coarse-scale element $v_h \in W_h$, we denote by $v_{r,h}$ the corresponding fine-scale element (r stands for resolved), $v_{r,h} = E^{MsFEM} v_h$. Note that for linear problems in Chapter

Y. Efendiev, T.Y. Hou, *Multiscale Finite Element Methods: Theory and Applications,* 47
Surveys and Tutorials in the Applied Mathematical Sciences 4,
DOI 10.1007/978-0-387-09496-0_3, © Springer Science+Business Media LLC 2009

2 (also in Chapter 4), we have used the subscript h (e.g., p_h) to denote the approximation of the fine-scale solution, whereas for nonlinear problems p_h stands for the approximation of the homogenized solution and $p_{r,h}$ is the approximation of the fine-scale solution. The spatial field $v_{r,h}$ is defined via the solution of the local problem

$$-\operatorname{div} k(x, \eta^{v_h}, \nabla v_{r,h}) = 0 \text{ in } K, \tag{3.2}$$

where $v_{r,h} = v_h$ on ∂K and $\eta^{v_h} = (1/|K|)\int_K v_h dx$ for each K (coarse element). The equation (3.2) is solved in each K for given $v_h \in W_h$. Note that the choice of η^{v_h} guarantees that (3.2) has a unique solution. In nonlinear problems, $\mathcal{P}_h$ is no longer a linear space (although we keep the same notation). We would like to point out that different boundary conditions can be chosen as in the case of linear problems to obtain more accurate solutions and this is discussed later. For linear problems, E^{MsFEM} is a linear operator, where for each $v_h \in W_h$, $v_{r,h}$ is the solution of the linear problem. Consequently, $\mathcal{P}_h$ is a linear space that can be obtained by mapping a basis of W_h. This is precisely the construction presented in [143] for linear elliptic equations (see Section 3.3).

An illustrating example. To illustrate the multiscale mapping concept, we consider the equation

$$\operatorname{div}(k(x,p)\nabla p) = f. \tag{3.3}$$

In this case, the multiscale map is defined in the following way. For each $v_h \in W_h$, $v_{r,h}$ is the solution of

$$div(k(x, \eta^{v_h})\nabla v_{r,h}) = 0 \text{ in } K \tag{3.4}$$

with the boundary condition $v_{r,h} = v_h$ on ∂K. For example, if K is a triangular element and v_h are piecewise linear functions, then the nodal values of v_h will determine $v_{r,h}$. Equation (3.4) is solved on the fine grid, in general. In the one-dimensional case, one can obtain an explicit expression for E^{MsFEM} (see (3.12)). The map E^{MsFEM} is nonlinear; however, for a fixed v_h, this map is linear. In fact, one can represent $v_{r,h}$ using multiscale basis functions as $v_{r,h} = \sum_i \alpha_i \phi_i^{v_h}$, where $\alpha_i = v_h(x_i)$ (x_i being nodal points) and $\phi_i^{v_h}$ are multiscale basis functions defined by

$$\operatorname{div}(k(x, \eta^{v_h})\nabla \phi_i^{v_h}) = 0 \text{ in } K, \quad \phi_i^{v_h} = \phi_i^0 \text{ on } \partial K.$$

Consequently, linear multiscale basis functions can be used to represent $v_{r,h}$. We can further assume that the basis functions can be interpolated via a simple linear interpolation

$$\phi_i^{\eta_0} \approx \beta_1 \phi_i^{\eta_1} + \beta_2 \phi_i^{\eta_2}, \tag{3.5}$$

where β_1, β_2 are interpolation constants that depend on η_0, η_1, and η_2. For example, if $\eta_1 < \eta_0 < \eta_2$, then $\beta_1 = (\eta_0 - \eta_1)/(\eta_2 - \eta_1)$ and $\beta_2 = (\eta_2 - \eta_0)/(\eta_2 -$

η_1). In this case, one can compute the basis functions for some pre-defined values of ηs and interpolate for any other η. We can also use the combined set of basis functions $\{\phi_i^{\eta_1}, \phi_i^{\eta_2}\}$ for representing the solution for a set of values of η.

Multiscale numerical formulation. As discussed earlier, one can use various global formulations for MsFEM. Our goal is to find $p_h \in W_h$ (consequently, $p_{r,h}(= E^{MsFEM} p_h) \in \mathcal{P}_h$) such that $p_{r,h}$ "approximately" satisfies the fine-scale equations. When substituting $p_{r,h}$ into the fine-scale system, the resulting equations need to be projected onto coarse-dimensional space because $p_{r,h}$ is defined via p_h. This projection is done by multiplying the fine-scale equation with coarse-scale test functions. First, we present a Petrov–Galerkin formulation of MsFEM for nonlinear problems. The multiscale finite element formulation of the problem is the following. Find $p_h \in W_h$ (consequently, $p_{r,h}(= E^{MsFEM} p_h) \in \mathcal{P}_h$) such that

$$\langle \kappa_{r,h} p_h, v_h \rangle = \int_\Omega f v_h dx, \quad \forall v_h \in W_h, \tag{3.6}$$

where

$$\langle \kappa_{r,h} p_h, v_h \rangle = \sum_{K \in \mathcal{T}_h} \int_K (k(x, \eta^{p_h}, \nabla p_{r,h}) \cdot \nabla v_h + k_0(x, \eta^{p_h}, \nabla p_{r,h}) v_h) dx. \tag{3.7}$$

As we notice that the fine-scale equation is multiplied by coarse-scale test functions from W_h. Note that the above formulation of MsFEM is a generalization of the Petrov–Galerkin MsFEM introduced earlier for linear problems.

We note that the method presented above can be extended to systems of nonlinear equations.

Pseudo-code. In the computations, we seek $p_h = \sum_i p_i \phi_i^0 \in W_h$ which satisfies (3.6). This equation can be written as a nonlinear system of equations for p_i,

$$A(p_1, ..., p_i, ...) = b, \tag{3.8}$$

where A is given by (3.7). Here, p_i can be thought as nodal values of p_h on the coarse grid. To find the form of A, we take $v_h = \phi_i^0$ in (3.7). This yields the ith equation of the system (3.8) denoted by $A_i(p_1, ..., p_i, ...) = b_i$, where $b_i = \int_\Omega f \phi_i^0 dx$. Denote by K_i triangles with the common vertex x_i. Then,

$$A_i(p_1, ..., p_i, ...) = \sum_{K_i} \int_{K_i} (k(x, \eta^{p_h}, \nabla p_{r,h}) \cdot \nabla \phi_i^0 + k_0(x, \eta^{p_h}, \nabla p_{r,h}) \phi_i^0) dx.$$

In each K_i, $\eta^{p_h} = \sum_j p_j \int_{K_i} \phi_j^0 dx$, where j are the nodes of the triangles with common vertex i. Later, we present a one-dimensional example, where an explicit expression for A_i is presented. It is clear that A_i will depend only on the nodal values of p_j which are defined at the nodes of the triangles with common vertex i. This system is usually solved by an iterative method on a

Algorithm 3.1.1

Construct a coarse grid.
Until convergence, do
- For each coarse element, compute the multiscale map $E : W_h \to \mathcal{P}_h$ according to (3.2).
- Solve the coarse variational formulation using (3.6) and (3.7).

coarse grid and the local solutions can be re-used and treated independently in each coarse-grid block. In Section 3.7, we discuss some of the iterative methods.

One dimensional example. We consider a simple one-dimensional case

$$-(k(x,p)p')' = f,$$

$p(0) = p(1) = 0$, where $'$ refers to the spatial derivative. We assume that the interval $[0,1]$ is divided into N segments

$$0 = x_0 < x_1 < x_2 < \cdots < x_i < x_{i+1} < \cdots < x_N = 1.$$

For a given $p_h \in W_h$, $p_{r,h}$ is the solution of

$$(k(x,\eta^{p_h})p'_{r,h})' = 0, \tag{3.9}$$

where $p_{r,h}(x_i) = p_h(x_i)$ for every interior node x_i. In the interval $[x_{i-1}, x_i]$, (3.9) can be solved. To compute (3.7), we only need to evaluate $k(x,\eta^{p_h})p'_{r,h}$. Noting that this quantity is constant, $k(x,\eta^{p_h})p'_{r,h} = c(x_{i-1},x_i)$ (directly follows from (3.9)), we can easily find that

$$p'_{r,h} = c(x_{i-1},x_i)/k(x,\eta^{p_h}), \tag{3.10}$$

where $\eta^{p_h} = \frac{1}{2}(p_h(x_{i-1}) + p_h(x_i))$. Taking the integral of (3.10) over $[x_{i-1}, x_i]$, we have

$$p_h(x_i) - p_h(x_{i-1}) = c(x_{i-1},x_i)\int_{x_{i-1}}^{x_i} \frac{1}{k(x,\eta^{p_h})}dx.$$

Consequently,

$$c(x_{i-1},x_i) = k(x,\eta^{p_h})p'_{r,h} = \frac{p_h(x_i) - p_h(x_{i-1})}{\int_{x_{i-1}}^{x_i} \frac{1}{k(x,\eta^{p_h})}dx}.$$

To evaluate (3.7) (note that $k_0 = 0$) with $v_h = \phi_i^0$, we have

$$A_i(p_h) = \int_{x_{i-1}}^{x_i} c(x_{i-1},x_i)(\phi_i^0)'dx + \int_{x_i}^{x_{i+1}} c(x_i,x_{i+1})(\phi_i^0)'dx$$
$$= \frac{p_h(x_i) - p_h(x_{i-1})}{\int_{x_{i-1}}^{x_i} \frac{1}{k(x,\eta^{p_h})}dx}\int_{x_{i-1}}^{x_i} (\phi_i^0)'dx + \frac{p_h(x_{i+1}) - p_h(x_i)}{\int_{x_i}^{x_{i+1}} \frac{1}{k(x,\eta^{p_h})}dx}\int_{x_i}^{x_{i+1}} (\phi_i^0)'dx.$$

Denoting $p_i = p_h(x_i)$ and taking into account that $\int_{x_{i-1}}^{x_i} (\phi_i^0)' dx = 1$, $\int_{x_i}^{x_{i+1}} (\phi_i^0)' dx = -1$, we have

$$A_i(p_{i-1}, p_i, p_{i+1}) = \frac{p_i - p_{i-1}}{\int_{x_{i-1}}^{x_i} \frac{1}{k(x, \frac{1}{2}(p_{i-1}+p_i))} dx} - \frac{p_{i+1} - p_i}{\int_{x_i}^{x_{i+1}} \frac{1}{k(x, \frac{1}{2}(p_i+p_{i+1}))} dx}. \quad (3.11)$$

Using the above calculations, one can easily write down an explicit expression for the multiscale map, $E^{MsFEM} : p_h \to p_{r,h}$. In particular, from (3.10), it can be shown that $p_{r,h}$ in $[x_{i-1}, x_i]$ is given by

$$p_{r,h}(x) = p_h(x_{i-1}) + \frac{p_h(x_i) - p_h(x_{i-1})}{\int_{x_{i-1}}^{x_i} \frac{1}{k(x, \eta^{p_h})} dx} \int_{x_{i-1}}^{x} \frac{dx}{k(x, \eta^{p_h})}. \quad (3.12)$$

One can use explicit solutions (see page 117, [220]) in a general case

$$-(k(x, p, p')') = f$$

to write down the variational formulation of (3.7) via the nodal values p_i. In particular, denote $\xi = \xi(x, \eta, c)$ to be the solution of

$$k(x, \eta, \xi) = c.$$

Then, from (3.2), we obtain $p'_{r,h} = \xi(x, \eta^{p_h}, c(x_{i-1}, x_i))$. Taking the integral of this equation over $[x_{i-1}, x_i]$, we obtain

$$p_h(x_i) - p_h(x_{i-1}) = \int_{x_{i-1}}^{x_i} \xi(x, \eta^{p_h}, c(x_{i-1}, x_i)) dx.$$

Because $\eta^{p_h} = \frac{1}{2}(p_{i-1} + p_i)$, we have the following implicit equation for $c(x_{i-1}, x_i)$

$$p_i - p_{i-1} = \int_{x_{i-1}}^{x_i} \xi(x, \frac{1}{2}(p_{i-1} + p_i), c(x_{i-1}, x_i)) dx.$$

With this implicit expression for $c(x_{i-1}, x_i)$, we have

$$\begin{aligned} A_i(p_h) &= \int_{x_{i-1}}^{x_i} c(x_{i-1}, x_i)(\phi_i^0)' dx + \int_{x_i}^{x_{i+1}} c(x_i, x_{i+1})(\phi_i^0)' dx \\ &= c(x_{i-1}, x_i) - c(x_i, x_{i+1}). \end{aligned}$$

This expression shows that $A_i(p_h)$ nonlinearly depends on p_{i-1}, p_i, and p_{i+1} and provides an explicit expression for the system of nonlinear equations that result from (3.7).

3.2 Multiscale finite volume element method (MsFVEM)

Next, we present a different formulation that provides a mass conservative method. By its construction, the finite volume method has local conservative properties [118] and it is derived from a local relation, namely the balance equation/conservation expression on a number of subdomains which are called control volumes. The finite volume element method can be considered as a Petrov–Galerkin finite element method, where the test functions are constants defined in a dual grid. For simplicity, we consider a triangular coarse grid. Consider a triangle K, and let z_K be its barycenter. The triangle K is divided into three quadrilaterals of equal area by connecting z_K to the midpoints of its three edges. We denote these quadrilaterals by K_z, where $z \in Z_h(K)$ are the vertices of K. Also we denote $Z_h = \bigcup_K Z_h(K)$, and Z_h^0 are all vertices that do not lie on $\partial\Omega_D$, where $\partial\Omega_D$ are Dirichlet boundaries. In this case, the control volume V_z is defined as the union of the quadrilaterals K_z sharing the vertex z (see Figure 3.1). The MsFVEM is to find $p_h \in W_h$ (consequently,

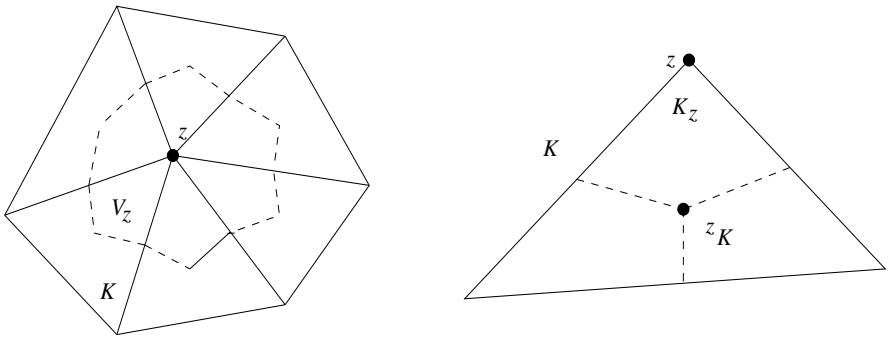

Fig. 3.1. *Left*: Portion of triangulation sharing a common vertex z and its control volume. *Right*: Partition of a triangle K into three quadrilaterals.

$p_{r,h} = E^{MsFEM} p_h$) such that

$$-\int_{\partial V_z} k\left(x, \eta^{p_h}, \nabla p_{r,h}\right) \cdot n \, ds + \int_{V_z} k_0\left(x, \eta^{p_h}, \nabla p_{r,h}\right) \, dx = \int_{V_z} f \, dx \quad \forall z \in Z_h^0, \tag{3.13}$$

where n is the unit normal vector pointing outward on ∂V_z. Note that the number of control volumes which satisfies (3.13) is the same as the dimension of W_h. The equation (3.13) gives rise to the finite-dimensional system of equations that provide the solution at the coarse nodes.

3.3 Examples of $\mathcal{P}_h$

Linear case. For linear operators, $\mathcal{P}_h$ can be obtained by mapping a basis of W_h because E^{MsFEM} is a linear operator. Define a basis of W_h, $W_h = \text{span}(\phi_i^0)$, where ϕ_i^0 are standard linear basis functions (assuming K is a triangular or tetrahedral element). Denote by ϕ_i the map of each basis function ϕ_i^0 (i.e., $\phi_i = E^{MsFEM}\phi_i^0$). From the definition of E^{MsFEM} it follows that ϕ_i satisfies

$$-\operatorname{div}(k(x)\nabla\phi_i) = 0 \text{ in } K \in \mathcal{T}_h \tag{3.14}$$

and $\phi_i = \phi_i^0$ on ∂K. These are the basis functions defined for MsFEM in Chapter 2.

Special nonlinear case. For the special case, $k(x, p, \nabla p) = k(x)b(p)\nabla p$, $\mathcal{P}_h$ can be related to the linear case. Indeed, for this case, the local problems associated with the multiscale mapping E^{MsFEM} (see (3.2)) have the form

$$-\operatorname{div}(k(x)b(\eta^{v_h})\nabla v_{r,h}) = 0 \text{ in } K.$$

Because η^{v_h} are constants over K, the local problems satisfy the linear equations,

$$-\operatorname{div}(k(x)\nabla\phi_i) = 0 \text{ in } K,$$

and $\mathcal{P}_h$ can be obtained by mapping a basis of $\mathcal{T}_h$ as it is done in the linear case. Thus, for this case, the basis functions are the same as those for the linear problem.

$\mathcal{P}_h$ *using subdomain problems.* One can use the solutions of smaller subdomain (smaller than $K \in \mathcal{T}_h$), RVE, problems to approximate the solutions of the local problems (3.2). This can be done if the small region can be used to represent the heterogeneities within the coarse-grid block, for example, periodic heterogeneities when the size of the period is much smaller than the coarse-grid block size. As in the linear case, we would like to use the solution in smaller regions to approximate the integrals on the right-hand side of (3.7). In these cases, we can solve (3.2) in a subdomain RVE with boundary conditions v_h restricted onto the subdomain boundaries as done in Section 2.6. More precisely, instead of (3.2), the following local problem is solved,

$$-\operatorname{div} k(x, \eta^{v_h}, \nabla\tilde{v}_{r,h}) = 0 \text{ in } K_{\text{loc}}, \tag{3.15}$$

where $\tilde{v}_{r,h} = v_h$ on ∂K_{loc} and $\eta^{v_h} = \frac{1}{|K|}\int_K v_h dx$ for each K (coarse element). The integrals in (3.7) can be computed using K_{loc},

$$\langle \kappa_{r,h} p_h, v_h\rangle \approx \sum_{K\in\mathcal{T}_h} \frac{|K|}{|K_{\text{loc}}|}\int_{K_{\text{loc}}} (k(x, \eta^{p_h}, \nabla\tilde{p}_{r,h})\cdot\nabla v_h + k_0(x, \eta^{p_h}, \nabla\tilde{p}_{r,h})v_h)dx, \tag{3.16}$$

where $\tilde{p}_{r,h}$ are only computed in K_{loc} using (3.15). The equations (3.15) and (3.16) provide the formulation of MsFEM when using regions smaller than the coarse-grid block.

As in the case of linear problems, it was shown that ([220])

$$\lim_{\epsilon\to 0}\frac{1}{|K|}\int_K k(x,\eta^{v_h},\nabla v_{r,h})dx=\frac{1}{|K|}\int_K k^*(x,\eta^{v_h},\nabla v^0_{r,h})dx, \tag{3.17}$$

where ϵ is the characteristic length scale and $v^0_{r,h}$ is the homogenized part of $v_{r,h}$ defined in a G-convergence setting (e.g., [220]). In particular, $v^0_{r,h}$ satisfies $\operatorname{div} k^*(x,\eta^{v_h},v^0_{r,h})=0$ in K, $v^0_{r,h}=v_h$ on ∂K; a similar result holds in $K_{\rm loc}$ (cf. (2.21)). As in the linear case, it is easy to show that if $k^*(x,\eta,\xi)$ is smooth spatial function, then $v^0_{r,h}$ is approximately equal to v_h for small h. From here, one can show that $(1/|K_{\rm loc}|)\int_{K_{\rm loc}} k(x,\eta^{v_h},\nabla\tilde{v}_{r,h})dx$ approximates $(1/|K|)\int_K k(x,\eta^{v_h},\nabla v_{r,h})dx$ in the limit $\lim_{h\to 0}\lim_{\epsilon\to 0}$. Based on (3.17), one can evaluate the integrals on the right-hand side of (3.7). To find the fine-scale approximation, the local solutions defined by (3.15) can be extended to the whole domain. This is based on the homogenization concept and $\nabla\tilde{v}_{r,h}$ is extended periodically in each coarse-grid block.

One can also use periodic homogenization and first-order correctors to approximate the solution of the local problem if $k(x,x/\epsilon,\eta,\xi)$ and $k_0(x,x/\epsilon,\eta,\xi)$ are locally periodic with respect to $y=x/\epsilon$. In this case, for each coarse grid block and $v_h\in W_h$, the following cell problem is solved,

$$\operatorname{div}_y(k(x,y,\eta^{v_h},\nabla_x v_h+\nabla_y N_{v_h}))=0 \text{ in } Y, \tag{3.18}$$

where Y is the period and N_{v_h} is the periodic function with zero average (assume v_h is piecewise linear; i.e., ∇v_h is constant within K). Then, it can be shown that

$$\frac{1}{|K|}\int_K k(x,x/\epsilon,\eta^{v_h},\nabla v_{r,h})dx\approx\frac{1}{|Y|}\int_Y k(x,y,\eta^{v_h},\nabla v_h+\nabla N_{v_h})dx$$

in the limit as $\epsilon/h\to 0$. Consequently, the local periodic solution (3.18) can be used to approximate the right-hand side of (3.7). This provides CPU savings when there is strong scale separation (cf. Section 2.6).

3.4 Relation to upscaling methods

One can draw a parallel between multiscale methods and upscaling/homogenization techniques. First, we briefly describe an upscaling technique for (3.1) which is derived from homogenization methods (e.g., [220]). The main idea of upscaling techniques is to form a coarse-scale equation and pre-compute the effective coefficients. In the case of nonlinear elliptic equations, the coarse-scale equation has the same form as the fine-scale equation except that the fluxes $k(\cdot,\cdot,\cdot)$ and $k_0(\cdot,\cdot,\cdot)$ are replaced by effective homogenized fluxes. The effective coefficients in upscaling methods are computed using the solution of the local problem in a representative volume. For each $\eta\in\mathbb{R}$ and $e\in\mathbb{R}^d$, the following local problem is solved

$$\operatorname{div} k(x,\eta,\nabla\phi_e)=0 \text{ in } K \tag{3.19}$$

with $\phi_e(x)=x\cdot e$ on ∂K. The effective coefficients are computed in each K as

$$\tilde{k^*}(\eta,e)=\frac{1}{|K|}\int_K k(x,\eta,\nabla\phi_e)dx,\ \ \tilde{k_0^*}(\eta,e)=\frac{1}{|K|}\int_K k_0(x,\eta,\nabla\phi_e)dx. \tag{3.20}$$

We note that $\tilde{k^*}$ and $\tilde{k_0^*}$ are not the same as the homogenized coefficients and (3.20) can be computed for any point in the domain by placing the point at the center of K. If k and k_0 are periodic with respect to spatial variables, one can solve the local problems (3.19) over the period (with periodic boundary conditions) and perform averaging (3.20) over the period. One can also use various boundary conditions, including oversampling methods, when solving (3.19). Once the effective coefficients are calculated, the coarse-scale equation

$$\operatorname{div} k^*(x,p^*,\nabla p^*)+k_0^*(x,p^*,\nabla p^*)=f$$

with $k^*=\tilde{k}^*$ and $k_0^*=\tilde{k}_0^*$ is solved. In practice, one can pre-compute k^* and k_0^* for different values of $\eta\in\mathbb{R}$ and $e\in\mathbb{R}^d$, and use interpolation for evaluating k^* and k_0^* for other values of $\eta\in\mathbb{R}$ and $e\in\mathbb{R}^d$. Note that for linear problems, it is sufficient to solve (3.19) for d linearly independent vectors $e_1,...,e_d$ in $\mathbb{R}^d$ because $\phi_e=\sum_i\beta_i\phi_{e_i}$ if $e=\sum_i\beta_i e_i$. This is not the case for nonlinear problems and one needs to consider all possible $\eta\in\mathbb{R}$ and $e\in\mathbb{R}^d$.

MsFEMs do not compute effective parameters explicitly. One can show that, as in the case of linear problems, MsFEM for nonlinear problems is similar to upscaling methods. However, in MsFEMs, "the effective parameters" (in the form of local solutions) are computed on-the-fly. Note that one can compute the effective parameters based on these local solutions. The computation of the local solutions on-the-fly is more efficient when one deals with a limited range of values of $\eta=(1/|K|)\int_K v_h dx$ and $e=\nabla v_h$. Indeed, many simulations in practice do not require a lookup table of k^* and k_0^* for all possible values of $\eta\in\mathbb{R}$ and $e\in\mathbb{R}^d$, and the computation on-the-fly can save CPU time. Moreover, one can still use pre-computed local solutions to compute the effective coefficients, and then store them. These effective coefficients can be used in the simulation to approximate k^* and k_0^* for those values of $\eta\in\mathbb{R}$ and $e\in\mathbb{R}^d$ that are not computed. Moreover, in MsFEMs, one can use a larger set of multiscale basis functions for more accurate approximation. For example, for the simple nonlinear elliptic equation (3.3), one can use multiscale basis functions corresponding to several values of η and avoid the interpolation step (cf. (3.5)).

3.5 Multiscale finite element methods for nonlinear parabolic equations

In this section, we present an extension of MsFEM to nonlinear parabolic equations. We consider

$$\frac{\partial}{\partial t}p - \operatorname{div} k(x,t,p,\nabla p) + k_0(x,t,p,\nabla p) = f. \tag{3.21}$$

For the nonlinear parabolic equations, the space–time operator E^{MsFEM} is constructed in the following way. For each $v_h \in W_h$ there is a corresponding element $v_{r,h} = E^{MsFEM} v_h$ that is defined by

$$\frac{\partial}{\partial t}v_{r,h} - \operatorname{div} k(x,t,\eta^{v_h},\nabla v_{r,h}) = 0 \text{ in } K \times [t_n, t_{n+1}], \tag{3.22}$$

with boundary condition $v_{r,h} = v_h$ on ∂K, and $v_{r,h}(t = t_n) = v_h$. Here $\eta^{v_h} = (1/|K|)\int_K v_h dx$.

Next, we present a global formulation of MsFEM. Our goal is to find $p_h \in W_h$ ($p_{r,h} = E^{MsFEM} p_h$) at time $t = t_{n+1}$ such that

$$\int_{t_n}^{t_{n+1}} \int_\Omega \left(\frac{\partial}{\partial t}p_h\right) v_h dx dt + \kappa(p_h, v_h) = \int_{t_n}^{t_{n+1}} \int_\Omega f v_h dx dt, \quad \forall v_h \in W_h, \tag{3.23}$$

where

$$\kappa(p_h, w_h) = \sum_K \int_{t_n}^{t_{n+1}} \int_K (k(x,t,\eta^{p_h},\nabla p_{r,h})\cdot\nabla w_h + k_0(x,t,\eta^{p_h},\nabla p_{r,h}) w_h) dx dt.$$

The expression (3.23) can be further simplified to

$$\int_\Omega p_h(x,t_{n+1}) v_h dx - \int_\Omega p_h(x,t_n) v_h dx + \kappa(p_h, v_h) = \int_{t_n}^{t_{n+1}} \int_\Omega f v_h dx dt, \forall v_h \in W_h.$$

Here $p_{r,h}$ is the solution of the local problem (3.22) for a given p_h, $\eta^{p_h} = (1/|K|)\int_K p_h dx$, and p_h is known at $t = t_n$. If p_h at time $t = t_{n+1}$ is used in $\kappa(p_h, v_h)$, then the resulting method is implicit; that is

$$\int_\Omega p_h(x,t_{n+1}) v_h dx - \int_\Omega p_h(x,t_n) v_h dx + \kappa(p_h(x,t_{n+1}), v_h) = \int_{t_n}^{t_{n+1}} \int_\Omega f v_h dx dt, \quad \forall v_h \in W_h.$$

If p_h at time $t = t_n$ is used in $\kappa(p_h, v_h)$, then the resulting method is explicit. The Petrov–Galerkin formulation of the MsFEM can be replaced by the finite volume formulation as is done for nonlinear elliptic equations.

We would like to note that the operator E^{MsFEM} can be constructed using larger domains as is done in MsFEMs with oversampling [145]. This way one reduces the effects of the boundary conditions and initial conditions. In particular, for temporal oversampling it is sufficient to start the computations before t_n and end them at t_{n+1}. Consequently, the oversampling domain for $K \times [t_n, t_{n+1}]$ consists of $[\tilde{t}_n, t_{n+1}] \times K_E$, where $\tilde{t}_n < t_n$ and $K \subset K_E$. We would like to note that oscillatory initial conditions can be imposed (without

using oversampling techniques) based on the solution of the elliptic part of the local problems (3.22). These initial conditions at $t = t_n$ are the solutions of

$$-\operatorname{div}(k(x,t,\eta,\nabla p_{r,h})) = 0 \text{ in } K, \tag{3.24}$$

or

$$-\operatorname{div}(\overline{k}(x,\eta,\nabla p_{r,h})) = 0 \text{ in } K, \tag{3.25}$$

where $\overline{k}(x,\eta,\xi) = (1/(t_{n+1}-t_n))\int_{t_n}^{t_{n+1}} k(x,\tau,\eta,\xi)d\tau$ and $p_{r,h} = p_h$ on ∂K. The latter can become efficient depending on the interplay between the temporal and spatial scales.

Note that in the case of periodic media the local problems can be solved in a single period in order to construct $\kappa(p_h, w_h)$. In general, one can solve the local problems in a domain different from K (an element) to calculate $\kappa(p_h, w_h)$. Note that the numerical advantages of our approach over the fine scale simulation are similar to those of MsFEMs. In particular, for each Newton's iteration a linear system of equations on a coarse grid is solved. Moreover, the local solutions can be re-used and treated independently in each coarse grid block.

For some special cases the operator E^{MsFEM} introduced in the previous section can be simplified (see [112]). In general one can avoid solving the local parabolic problems if the ratio between temporal and spatial scales is known, and solve instead a simplified equation. For example, let the spatial scale be ϵ^β and the temporal scale be ϵ^α; that is, $k(x,t,\eta,\xi) = k(x/\epsilon^\beta, t/\epsilon^\alpha, \eta, \xi)$. If $\alpha < 2\beta$ one can solve instead of (3.22) the local problem $-\operatorname{div}(k(x,t,\eta^{p_h},\nabla p_{r,h})) = 0$, if $\alpha > 2\beta$ one can solve instead of (3.22) the local problem $-\operatorname{div}(\overline{k}(x,\eta^{p_h},\nabla p_{r,h})) = 0$, where $\overline{k}(x,\eta,\xi)$ is an average over time of $k(x,t,\eta,\xi)$, and if $\alpha = 2\beta$ we need to solve the parabolic equation in $K \times [t_n, t_{n+1}]$, (3.22).

We would like to note that, in general, one can use (3.24) or (3.25) as oscillatory initial conditions and these initial conditions can be efficient for some cases. For example, for $\alpha > 2\beta$ with initial conditions given by (3.25) the solutions of the local problems (3.22) can be computed easily because they are approximated by (3.25). Moreover, one can expect better accuracy with (3.25) for the case $\alpha > 2\beta$ because this initial condition is more compatible with the local heterogeneities compared to the artificial linear initial conditions (cf. (3.22)).

One-dimensional example. We consider a simple one-dimensional case

$$\frac{\partial p}{\partial t} - (k(x,t,p)p')' = f,$$

$p(0) = p(1) = 0$, $p(t=0) = p_0(x)$. As before, we assume that the interval $[0,1]$ is divided into N segments $0 = x_0 < x_1 < x_2 < \cdots < x_i < x_{i+1} < \cdots < x_N = 1$ and the time interval $[0,T]$ is divided into M segments $0 = t_0 < t_1 < t_2 < \cdots < t_i < t_{i+1} < \cdots < t_M = T$. We present a discrete formulation

for the fully implicit method where the local problem (3.22) is elliptic (see discussion above); that is $(k(x,t,\eta^{v_h},v'_{r,h}))'=0$.

Denote by $p_i^n = p(x=x_i,t=t_n)$. Taking $v_h=\phi_i^0$ in (3.23) and using (3.11), one can easily get

$$(p_j^{n+1}-p_j^n)\int_\Omega \phi_j^0\phi_i^0 dx + A(p_{i-1}^{n+1},p_i^{n+1})(p_i^{n+1}-p_{i-1}^{n+1}) \\ -A(p_i^{n+1},p_{i+1}^{n+1})(p_{i+1}^{n+1}-p_i^{n+1}) = \int_\Omega f\phi_i^0 dx,$$

where

$$A(p_{i-1}^{n+1},p_i^{n+1}) = \frac{1}{\int_{x_{i-1}}^{x_i}\frac{1}{k(x,t_{n+1},\frac{1}{2}(p_{i-1}^{n+1}+p_i^{n+1}))}dx},$$
$$A(p_i^{n+1},p_{i+1}^{n+1}) = \frac{1}{\int_{x_i}^{x_{i+1}}\frac{1}{k(x,t_{n+1},\frac{1}{2}(p_i^{n+1}+p_{i+1}^{n+1}))}dx}.$$

3.6 Summary of convergence of MsFEM for nonlinear partial differential equations

The convergence of MsFEM for nonlinear problems has been studied for problems with scale separation (not necessarily periodic). These convergence results use homogenization or G-convergence results for nonlinear partial differential equations (see, e.g., [220] and Appendix B). To discuss these results, we assume that the fine scale is ϵ. It can be shown that the solution p converges (up to a subsequence) to p_0 in an appropriate norm, where $p_0\in W_0^{1,\gamma}(\Omega)$ is a solution of a homogenized equation

$$-\operatorname{div} k^*(x,p_0,\nabla p_0) + k_0^*(x,p_0,\nabla p_0) = f, \tag{3.26}$$

where k^* and k_0^* are homogenized coefficients.

In [112] it was shown using G-convergence theory that

$$\lim_{h\to 0}\lim_{\epsilon\to 0}\|p_h-p_0\|_{W_0^{1,\gamma}(\Omega)}=0, \tag{3.27}$$

(up to a subsequence) where p_0 is a solution of (3.26) and p_h is a MsFEM solution given by (3.6). Here γ is a parameter related to the monotonicity (see (6.42)–(6.46)). This result can be obtained without any assumption on the nature of the heterogeneities and cannot be improved because there could be infinitely many scales $\alpha(\epsilon)$ present such that $\alpha(\epsilon)\to 0$ as $\epsilon\to 0$.

For the periodic case, it can be shown that MsFEM converges in the limit as $\epsilon/h\to 0$. To show the convergence for $\epsilon/h\to 0$, we consider $h=h(\epsilon)$, such that $h(\epsilon)\gg\epsilon$ and $h(\epsilon)\to 0$ as $\epsilon\to 0$. We would like to note that this limit as well as the proof of the periodic case is different from (3.27), where

the double-limit is taken. In contrast to the proof of (3.27), the proof of the periodic case requires the correctors for the solutions of the local problems.

We present the convergence results for MsFEM solutions. For general nonlinear elliptic equations under the assumptions (stated later) (6.42)–(6.46) the strong convergence of MsFEM solutions can be shown. In the proof of this fact we show the form of the truncation error (in a weak sense) in terms of the resonance errors between the mesh size and small scale ϵ and explicitly derive the resonance errors. Under the general conditions, such as (6.42)–(6.46), one can prove strong convergence of MsFEM solutions without an explicit convergence rate (cf. [245]). To convert the obtained convergence rates for the truncation errors into the convergence rate of MsFEM solutions, additional assumptions, such as monotonicity, are needed.

Next, we formulate convergence theorems (see Section 6.2 and [104] for details).

Theorem 3.1. *Assume $k(x,\eta,\xi)$ and $k_0(x,\eta,\xi)$ are ϵ periodic functions with respect to x and let p_0 be a solution of* (3.26) *and p_h is a MsFEM solution given by* (3.6)*. Moreover, we assume that ∇p_h is uniformly bounded in $L^{\gamma+\alpha}(\Omega)$ for some $\alpha > 0$. Then*

$$\lim_{\epsilon\to 0} \|p_h - p_0\|_{W_0^{1,\gamma}(\Omega)} = 0, \tag{3.28}$$

where $h = h(\epsilon) \gg \epsilon$ and $h \to 0$ as $\epsilon \to 0$ (up to a subsequence).

Theorem 3.2. *Let p_0 and p_h be the solutions of the homogenized problem* (3.26) *and MsFEM* (3.6)*, respectively, with the coefficient $k(x,\eta,\xi) = k(x/\epsilon,\xi)$ and $k_0 = 0$. Then*

$$\|p_h - p_0\|_{W_0^{1,\gamma}(\Omega)} \le C\Big(\Big(\frac{\epsilon}{h}\Big)^{\beta} + h^{\delta}\Big), \tag{3.29}$$

where γ and δ depend on operator constants defined in (6.42)–(6.46)*.*

We note that in Theorem 3.2, we assume that the equation is monotone, whereas in Theorem 3.1, we assume that the equation is pseudo-monotone. As discussed in Section 6.2, the monotonicity allows us to obtain explicit convergence rates. For parabolic equations, one can prove similar results (see [112]). One can also prove the convergence of $p_{r,h}$ to the fine-scale solution p in $W^{1,\gamma}$ (e.g., [104, 112, 113]). Finally, we refer to [136] for convergence results of oversampling methods for nonlinear problems and applications to material science.

3.7 Numerical results

In this section we present some numerical results for MsFEMs for nonlinear elliptic equations. More numerical examples relevant to subsurface applications

can be found in [104]. We present numerical results for both the MsFEM and MsFVEM. We use an inexact-Newton algorithm as an iterative technique to tackle the nonlinearity. For the numerical examples below, we use $k(x,p,\nabla p) = k(x,p)\nabla p$. Let $\{\phi_i^0\}_{i=1}^{N_{dof}}$ be the standard piecewise linear basis functions of W_h. Then MsFEM solution may be written as

$$p_h = \sum_{i=1}^{N_{dof}} \alpha_i \, \phi_i^0 \tag{3.30}$$

for some $\alpha = (\alpha_1, \alpha_2, ..., \alpha_{N_{dof}})^T$. Recall that $p_h \in W_h$ is an approximation for a homogenized solution, and $p_{r,h}$ is an approximation for a fine-scale solution. We need to find α such that

$$F(\alpha) = 0, \tag{3.31}$$

where $F : \mathbb{R}^{N_{dof}} \to \mathbb{R}^{N_{dof}}$ is a nonlinear operator such that

$$F_i(\alpha) = \sum_{K \in K^h} \int_K k(x, \eta^{p_h}) \nabla p_{r,h} \cdot \nabla \phi_i^0 \, dx - \int_\Omega f \, \phi_i^0 \, dx. \tag{3.32}$$

We note that in (3.32) α is implicitly buried in η^{p_h} and $p_{r,h}$. An inexact-Newton algorithm is a variation of Newton's iteration for a nonlinear system of equations, where the Jacobian system is only approximately solved. To be specific, given an initial iterate α^0, for $k = 0, 1, 2, \cdots$ until convergence do the following

- Solve $F'(\alpha^k)\delta^k = -F(\alpha^k)$ by some iterative technique until $\|F(\alpha^k) + F'(\alpha^k)\delta^k\| \le \beta_k \|F(\alpha^k)\|$.
- Update $\alpha^{k+1} = \alpha^k + \delta^k$.

In this algorithm $F'(\alpha^k)$ is the Jacobian matrix evaluated at iteration k. We note that when $\beta_k = 0$ we have recovered the classical Newton iteration. Here we have used

$$\beta_k = 0.001 \left(\frac{\|F(\alpha^k)\|}{\|F(\alpha^{k-1})\|} \right)^2, \tag{3.33}$$

with $\beta_0 = 0.001$. Choosing β_k this way, we avoid oversolving the Jacobian system when α^k is still considerably far from the exact solution.

Next we present the entries of the Jacobian matrix. For this purpose, we use the following notations. Let $K_i^h = \{K \in \mathcal{T}_h : z_i \text{ is a vertex of } K\}$, $I^i = \{j : z_j \text{ is a vertex of } K \in K_i^h\}$, and $K_{ij}^h = \{K \in K_i^h : K \text{ shares } \overline{z_i z_j}\}$. We note that we may write $F_i(\alpha)$ as follows

$$F_i(\alpha) = \sum_{K \in K_i^h} \left(\int_K k(x, \eta^{p_h}) \nabla p_{r,h} \cdot \nabla \phi_i^0 \, dx - \int_K f \, \phi_i^0 \, dx \right), \tag{3.34}$$

with

$$-\operatorname{div}(k(x,\eta^{p_h})\,\nabla p_{r,h}) = 0 \text{ in } K \text{ and } p_{r,h} = \sum_{z_m \in Z_K} \alpha_m\,\phi_m^0 \text{ on } \partial K, \quad (3.35)$$

where Z_K is all the vertices of element K. It is apparent that $F_i(\alpha)$ is not fully dependent on all $\alpha_1, \alpha_2, ..., \alpha_d$. Consequently, $\partial F_i(\alpha)/\partial\alpha_j = 0$ for $j \notin I^i$. To this end, we denote $\psi_j = \partial p_{r,h}/\partial\alpha_j$. By applying the chain rule of differentiation to (3.35) we have the following local problem for ψ_j

$$-\operatorname{div}(k(x,\eta^{p_h})\,\nabla\psi_j) = \frac{1}{3}\operatorname{div}(\frac{\partial k(x,\eta^{p_h})}{\partial p}\,\nabla p_{r,h}) \text{ in } K \text{ and } \psi_j = \phi_j \text{ on } \partial K. \quad (3.36)$$

The fraction 1/3 comes from taking the derivative in the chain rule of differentiation. In the formulation of the local problem, we have replaced the nonlinearity in the coefficient by η^{p_h}, where for each triangle K $\eta^{p_h} = 1/3\sum_{i=1}^{3}\alpha_i^K$, which gives $\partial\eta^{p_h}/\partial\alpha_i = 1/3$. Moreover, for a rectangular element the fraction 1/3 should be replaced by 1/4.

Thus, provided that $v_{r,h}$ has been computed, then we may compute ψ_j using (3.36). Using the above descriptions we have the expressions for the entries of the Jacobian matrix:

$$\frac{\partial F_i}{\partial\alpha_i} = \sum_{K\in K_i^h}\left(\frac{1}{3}\int_K \frac{\partial k(x,\eta^{p_h})}{\partial p}\nabla p_{r,h}\cdot\nabla\phi_i^0\,dx + \int_K k(x,\eta^{p_h})\nabla\psi_i\cdot\nabla\phi_i^0\,dx\right) \quad (3.37)$$

$$\frac{\partial F_i}{\partial\alpha_j} = \sum_{K\in K_{ij}^h}\left(\frac{1}{3}\int_K \frac{\partial k(x,\eta^{p_h})}{\partial p}\nabla p_{r,h}\cdot\nabla\phi_i\,dx + \int_K k(x,\eta^{p_h})\nabla\psi_j\cdot\nabla\phi_i^0\,dx\right) \quad (3.38)$$

for $j \neq i$, $j \in I^i$.

The implementation of the oversampling technique is similar to the procedure presented earlier, except the local problems in larger domains are used. As in the nonoversampling case, we denote $\psi_j = \partial v_{r,h}/\partial\alpha_j$, such that after applying the chain rule of differentiation to the local problem we have:

$$\begin{aligned} -\operatorname{div}(k(x,\eta^{p_h})\,\nabla\psi_j) = \frac{1}{3}\operatorname{div}(\frac{\partial k(x,\eta^{p_h})}{\partial p}\,\nabla v_{r,h}) \text{ in } K_E \\ \psi_j = \phi_j^0 \text{ on } \partial K_E, \end{aligned} \quad (3.39)$$

where η^{p_h} is computed over the corresponding element K and ϕ_j^0 is understood as the nodal basis functions on oversampled domain K_E. Then all the rest of the inexact-Newton algorithms are the same as in the nonoversampling case. Specifically, we also use (3.37) and (3.38) to construct the Jacobian matrix of the system. We note that we only use ψ_j from (3.39) pertaining to the element K.

From the derivation (both for oversampling and nonoversampling) it is obvious that the Jacobian matrix is not symmetric but sparse. Computation of this Jacobian matrix is similar to computing the stiffness matrix resulting

from standard finite elements, where each entry is formed by accumulation of element-by-element contributions. Once we have the matrix stored in memory, then its action to a vector is straightforward. Because it is a sparse matrix, devoting some amount of memory for entry storage is inexpensive. The resulting linear system is solved using a preconditioned biconjugate gradient stabilized method.

An an example to illustrate the convergence of the nonlinear MsFEM, we consider the following problem

$$\begin{aligned} -\mathrm{div}(k(x/\epsilon, p)\nabla p) &= -1 \quad \text{in } \Omega, \\ p &= 0 \quad \text{on } \partial\Omega, \end{aligned} \tag{3.40}$$

where $\Omega = [0,1] \times [0,1]$, $k(x/\epsilon, p) = k(x/\epsilon)/\left(1+p\right)^{l(x/\epsilon)}$, with

$$k(x/\epsilon) = \frac{2 + 1.8\sin(2\pi x_1/\epsilon)}{2 + 1.8\cos(2\pi x_2/\epsilon)} + \frac{2 + \sin(2\pi x_2/\epsilon)}{2 + 1.8\cos(2\pi x_1/\epsilon)} \tag{3.41}$$

and $l(x/\epsilon)$ is generated from $k(x/\epsilon)$ such that the average of $l(x/\epsilon) = Ck(x/\epsilon)$ over Ω is 2 with an appropriate choice of C. Here we use $\epsilon = 0.01$. Because the exact solution for this problem is not available, we use a well-resolved numerical solution using the standard finite element method as a reference solution. The resulting nonlinear system is solved using the inexact-Newton algorithm. The reference solution is solved on a 512×512 mesh. Tables 3.1 and 3.3 present the relative errors of the solution with and without oversampling, respectively. Here N is the number of coarse blocks in each direction. In Tables 3.2 and 3.4, the relative errors for the multiscale finite volume element method are presented. The relative errors are computed as the corresponding error divided by the norm of the solution. In each table, the second, third, and fourth columns list the relative error in the L^2, H^1, and L^∞ norm, respectively. As we can see from these two tables, the oversampling significantly improves the accuracy of the multiscale method.

In our next example, we consider the problem with nonperiodic coefficients, where $k(x,\eta) = k(x)/(1+\eta)^{\alpha(x)}$. The coefficient $k(x) = \exp(\beta(x))$ is chosen such that $\beta(x)$ is a realization of a random field with the spherical variogram [85], the correlation lengths $l_1 = 0.2$, $l_2 = 0.02$, and with the variance $\sigma = 1$. The function $\alpha(x)$ is chosen such that $\alpha(x) = k(x) + \text{const}$ with the spatial average of 2. As for the boundary conditions we use "left-to-right flow" in the $\Omega = [0,5] \times [0,1]$ domain, $p = 1$ at the inlet ($x_1 = 0$), $p = 0$ at the outlet ($x_1 = 5$), and no flow boundary conditions on the lateral sides $x_2 = 0$ and $x_2 = 1$. In Table 3.5 we present the relative error for a multiscale method with oversampling. Similarly, in Table 3.6 we present the relative error for a multiscale finite volume method with oversampling. Clearly, the oversampling method captures the effects induced by the large correlation features. Both H^1 and horizontal flux errors are under five percent. Similar results have been observed for various kinds of nonperiodic heterogeneities.

In the next set of numerical examples, we test the MsFEM for problems with fluxes $k(x, \eta)$ that are discontinuous in space. The discontinuity in the fluxes is introduced by multiplying the underlying permeability function $k(x)$ by a constant in certain regions, while leaving it unchanged in the rest of the domain. As an underlying permeability field, $k(x)$, we choose the random field used for the results in Table 3.5. In the numerical example, the discontinuities are introduced along the boundaries of the coarse elements. In particular, $k(x)$ on the left half of the domain is multiplied by a constant J, where $J = \exp(4)$. The results in Table 3.7 show that the MsFEM converges and the error falls below five percent for relatively large coarsening. For the second numerical example (Table 3.8), the discontinuities are not aligned with the boundaries of the coarse elements. In particular, the discontinuity boundary is given by $x_2 = x_1\sqrt{2}+0.5$; that is the discontinuity line intersects the coarse-grid blocks. Similar to the aligned case, $\exp(4)$ jump magnitude is considered. The results presented above demonstrate the robustness and accuracy of our approach for anisotropic fields, where h and ϵ are nearly the same, and the fluxes that are discontinuous spatial functions.

As for CPU comparisons, we have observed more than 92% CPU savings when using MsFEMs without oversampling. With the oversampling approach, the CPU savings depend on the size of the oversampled domain. For example, if the oversampled domain size is two times larger than the target coarse block (half coarse block extension on each side) we have observed 70% CPU savings for a 64×64 and 80% CPU savings for a 128×128 coarse grid. In general, the computational cost will decrease if the oversampled domain size is close to the target coarse block size, and this cost will be close to the cost of the MsFEM without oversampling. Conversely, the error decreases if the size of the oversampled domains increases. In the numerical examples, we have observed the same errors for the oversampling methods using either one coarse block extension or half coarse block extensions. The latter indicates that the leading resonance error is eliminated for the problems under consideration by using a smaller oversampled domain. Oversampled domains with one coarse block extension are previously used in simulations of flow through heterogeneous porous media. As indicated in [145], one can use large oversampled domains for simultaneous computations of the several local solutions. Moreover, parallel computations will improve the speed of the method because the MsFEM is well suited for parallel computation [145]. For the problems where $k(x, \eta, \xi) = k(x)b(\eta)\xi$ (see Section 3.3 and Section 5.3 for applications) our multiscale computations are very fast because the basis functions are built in the beginning of the computations. In this case, we have observed more than 95% CPU savings. We again would like to remark that the local solutions can be re-used in our multiscale simulations. This is similar to homogenization where the homogenized fluxes are computed once.

Table 3.1. Relative MsFEM Errors Without Oversampling

N	L^2-Norm		H^1-Norm		L^∞-Norm	
	Error	Rate	Error	Rate	Error	Rate
32	0.029		0.115		0.03	
64	0.053	-0.85	0.156	-0.44	0.0534	-0.94
128	0.10	-0.94	0.234	-0.59	0.10	-0.94

Table 3.2. Relative MsFVEM Errors Without Oversampling

N	L^2-Norm		H^1-Norm		L^∞-Norm	
	Error	Rate	Error	Rate	Error	Rate
32	0.03		0.13		0.04	
64	0.05	-0.65	0.19	-0.60	0.05	-0.24
128	0.058	-0.19	0.25	-0.35	0.057	-0.19

Table 3.3. Relative MsFEM Errors with Oversampling

N	L^2-Norm		H^1-Norm		L^∞-Norm	
	Error	Rate	Error	Rate	Error	Rate
32	0.0016		0.036		0.0029	
64	0.0012	0.38	0.019	0.93	0.0016	0.92
128	0.0024	-0.96	0.0087	1.14	0.0026	-0.71

Table 3.4. Relative MsFVEM Errors with Oversampling

N	L^2-Norm		H^1-Norm		L^∞-Norm	
	Error	Rate	Error	Rate	Error	Rate
32	0.002		0.038		0.005	
64	0.003	-0.43	0.021	0.87	0.003	0.72
128	0.001	1.10	0.009	1.09	0.001	1.08

Table 3.5. Relative MsFEM Errors for Random Heterogeneities, Spherical Variogram, $l_1 = 0.20$, $l_2 = 0.02$, $\sigma = 1.0$

N	L^2-Norm		H^1-Norm		L^∞-Norm		hor. flux	
	Error	Rate	Error	Rate	Error	Rate	Error	Rate
32	0.0006		0.0505		0.0025		0.025	
64	0.0002	1.58	0.029	0.8	0.001	1.32	0.017	0.57
128	0.0001	1	0.016	0.85	0.0005	1	0.011	0.62

Table 3.6. Relative MsFVEM Errors for Random Heterogeneities, Spherical Variogram, $l_1 = 0.20$, $l_2 = 0.02$, $\sigma = 1.0$

N	L^2-Norm		H^1-Norm		L^∞-Norm		hor. flux	
	Error	Rate	Error	Rate	Error	Rate	Error	Rate
32	0.0006		0.0515		0.0025		0.027	
64	0.0002	1.58	0.029	0.81	0.0013	0.94	0.018	0.58
128	0.0001	1	0.016	0.85	0.0005	1.38	0.012	0.58

Table 3.7. Relative MsFEM Errors for Random Heterogeneities, Spherical Variogram, $l_1 = 0.20$, $l_2 = 0.02$, $\sigma = 1.0$, Aligned Discontinuity, Jump = exp(4)

N	L^2-Norm		H^1-Norm		L^∞-Norm		hor. flux	
	Error	Rate	Error	Rate	Error	Rate	Error	Rate
32	0.0011		0.1010		0.0068		0.195	
64	0.0006	0.87	0.0638	0.66	0.0045	0.59	0.109	0.84
128	0.0003	1.00	0.0349	0.87	0.0024	0.91	0.063	0.79

Table 3.8. Relative MsFEM Errors for Random Heterogeneities, Spherical Variogram, $l_1 = 0.20$, $l_2 = 0.02$, $\sigma = 1.0$, Nonaligned Discontinuity, Jump = exp(4)

N	L^2-Norm		H^1-Norm		L^∞-Norm		hor. flux	
	Error	Rate	Error	Rate	Error	Rate	Error	Rate
32	0.0067		0.1775		0.1000		0.164	
64	0.0016	2.07	0.0758	1.23	0.0288	1.80	0.077	1.09
128	0.0009	0.83	0.0687	0.14	0.0423	-0.55	0.039	0.98

3.8 Discussions

An alternative approach for solving nonlinear problems using MsFEM is to linearize them. For example, the nonlinear elliptic or parabolic equations considered in this chapter can be linearized, for example, as

$$\frac{\partial p^n}{\partial t} - \operatorname{div}(b(x,t,p^{n-1},\nabla p^{n-1})\nabla p^n) + k_0(x,t,p^{n-1},\nabla p^{n-1}) = f, \qquad (3.42)$$

where $b(x,t,\eta,\xi)\cdot\xi = k(x,t,\eta,\xi)$. At this point, we assume that the solutions of this linearized equation converge to a solution of the original nonlinear equation. Then, at every iteration, one can apply the MsFEM where the basis functions are constructed based on heterogeneous coefficients b that change at every iteration. Assuming that we can approximate the solution accurately with the MsFEM at every iteration and the error of approximation does not propagate, one can show that this procedure converges under some conditions.

One can also perform upscaling based on a linearized equation (3.42). In this case, the approximation of the upscaled solution of the limiting equation

(in the sense of linearized iterations) can be obtained. More precisely, if one denotes $b_{n-1}(x,t) = b(x,t,p^{n-1},\nabla p^{n-1})$ and $g_{n-1}(x,t) = k_0(x,t,p^{n-1},\nabla p^{n-1})$, then the upscaled equation corresponding to (3.42) will have the form

$$\frac{\partial p^{*,n}}{\partial t} - \operatorname{div}(b^*_{n-1}(x,t)\nabla p^{*,n}) + g^*_{n-1}(x,t) = f.$$

Assuming that the solution of this iterative procedure converges, the final upscaled equation will have the form

$$\frac{\partial p^*}{\partial t} - \operatorname{div}(b^*(x,t)\nabla p^*) + g^*(x,t) = f.$$

Note that the obtained effective parameters (e.g., b^*) implicitly contain the information about the upscaled solution p^*. Because the upscaled solution contains the information about the global boundary conditions and source terms in a nonlinear fashion, one cannot re-use the upscaled coefficients (e.g., b^*) if the source or boundary conditions are changed.

The approximation of the local problems in the presence of scale separation or periodicity is discussed in this chapter. In particular, the local solutions and the evaluation of the variational formulation (see (3.7)) can be carried out in smaller regions. These issues are discussed in greater detail in [136]. One can attempt to use approximate solutions in the variational formulation (3.7) to compute the resulting system of nonlinear equations (e.g., fewer Newton iterations). To our best knowledge, these approximations are not considered in the literature.

In this section, we considered nonlinear elliptic and parabolic equations; however, the proposed methods can be applied in more general situations (see Section 2.4). In Chapter 5, we discuss the application of nonlinear MsFEMs to Richards' equations and to fluid flows in deformable porous media. The latter involves coupled nonlinear equations involving the interface dynamics between the fluid and solid components of the media. The methods discussed in this chapter can also be applied in material sciences. For example, in [136], the author applies numerical homogenization methods similar to those discussed in this chapter to nonlinear elasticity. In particular, the paper [136] explores oversampling techniques in nonlinear heterogeneous equations both numerically and analytically. The author proves the convergence of the method with oversampling, for convex and quasi-convex energies, in the context of general heterogeneities. This analysis provides an interesting variational interpretation of the Petrov–Galerkin formulation of the nonconforming multiscale finite element method for periodic problems.