Monografías de la Real Academia de Ciencias de Zaragoza **33**, 21–38, (2010).

Efficient implementation of box-relaxation multigrid methods for the poroelasticity problem on semi-structured grids

Francisco José Gaspar, Francisco Javier Lisbona, and Carmen Rodrigo

Universidad de Zaragoza, Pedro Cerbuna 12, 50009, Zaragoza, Spain.

Abstract

We consider the numerical solution of a poroelasticity problem using a stabilized finite element method (FEM), based on the perturbation of the flow equation. Semi-structured triangular grids and stencil-based implementation of the linear FEM for displacements and pressure are used. An efficient procedure to construct the stencils associated with the basic diferential operators involved in the poroelasticity equations, using some reference stencils computed on a canonical hexagon, is presented. To solve the algebraic system of saddle point type, geometric multigrid methods based on box-relaxation are proposed which result to have a good performance for the considered problem. Numerical results are presented to show the behavior of the method.

1 Introduction

The theory of poroelasticity addresses the time dependent coupling between the deformation of a porous material and the fluid flow inside. Although this problem was first studied by Terzaghi in [18], its general statement was given by Biot in some papers, see [4, 5, 6]. Biot's consolidation models are used to study problems in a wide range of scientific disciplines, as geomechanics, hydrogeology, petrol engineering and biomechanics, for example. Here, we consider the quasi-static Biot model that sometimes is referred as the incompressible case model. The state of a poroelastic medium is characterized by the elastic displacements $\mathbf{u}$, and fluid pressure p at each point. It is well known that discretization by linear finite elements for both unknown fields results in an unstable method giving non-physical oscillations in the approximation of the pressure field. To overcome this trouble, we shall use a stabilized finite element scheme presented in [1], which permits us to use linear finite element spaces for both displacements and pressure, providing solutions without oscillations, independently of the chosen discretization parameters.

Finite element methods are usually considered to work with unstructured grids, due to its flexibility. These grids offer advantages with regard to their better fitting to complex geometries. However, an important issue in the finite element solution of PDE problems concerns the construction and storage of the large sparse system matrix. This is usually done by the process so-called "assembly", and due to the sparse character of the resulting matrix it is very important the way in which it is stored. Data structures commonly used to this purpose work with a system of indirect indexing to access only the non-zero entries of the matrix which sometimes leads to some performance difficulties. On the other hand, the data structures are much more efficient when working with structured grids. A good alternative which combines the advantages of both types of meshes is to work with semi-structured grids, that is, to consider an unstructured mesh as coarsest grid in order to fit well the domain geometry, and to apply regular refinements to its elements.

One of the most important aspects in the numerical solution of partial differential equations is the efficient solution of the corresponding large systems of equations arising from their discretization. Multigrid methods [7, 12, 19] are among the most powerful techniques for solving such type of algebraic systems, and they have become very popular among the scientific community. Geometric multigrid methods are characterized by employing a hierarchy of grids. We are interested in the use of semi-structured triangular grids, where a nested hierarchy of grids is obtained by dividing each triangle into four congruent ones, connecting the midpoints of their edges. These grids provide a suitable framework for the implementation of a geometric multigrid algorithm, permitting the use of stencil-based data structures, see [3], being necessary only a few stencils to represent the discrete operator. Besides, an efficient procedure to compute such stencils using canonical stencils associated with a reference hexagon is proposed, providing expressions which give the stencils corresponding to an arbitrary triangle.

The choice of a suitable smoother is an important feature for the design of an efficient geometric multigrid method, and even it requires special attention when one works with systems of PDEs because the smoother should smooth the error for all unknowns. Moreover, for saddle point problems, see [2], numerical experiments show that smoothing factors of standard collective point-wise relaxations are not satisfactory [19]. The poroelasticity problem is an example of such type of systems, and its resolution by multigrid on semi-structured grids is the aim of this paper. An overview of multigrid methods for discretizations on rectangular grids of saddle point problems is presented in [19], where box-relaxation appears as one of the most suitable smoothers for this kind of problems. It consists of decomposing the mesh into small subdomains and treating them separately, that is, all (or a part of) the equations corresponding to the points in each subdomain are solved simultaneously as a system. This class of smoothers was introduced by Vanka in [21], to solve the finite difference discretization on rectangular grids of the Navier-Stokes

equations. Since then, much literature can be found about the application of this type of smoothers, mainly in the field of Computational Fluid Dynamics (CFD) [13, 20]. There are less papers concerning to the performance of this relaxation in the context of Computational Solid Mechanics (CSM), see for example [22]. However, for discretizations of the poroelasticity problem on rectangular grids, it has been proved to obtain very good results with these smoothers. For instance, in [10] a box-relaxation is performed for a discretization of the problem on staggered rectangular grids. Hence, it seems a good idea to extend box-relaxation to regular triangular grids.

The outline of this paper is the following. In Section 2, the formulation of the poroelasticity problem, as well as its stabilized finite element discretization, will be introduced. In Section 3, an efficient stencil-based implementation on semi-structured grids of the stabilized FEM scheme is developed. Finally, Section 4 is devoted to introduce the proposed geometric multigrid algorithm, based on Vanka-type relaxation on triangular grids, and to present some numerical experiments illustrating the behavior of such smoothers for the poroelasticity problem.

2 Poroelasticity problem

We consider the quasi-static Biot's model for soil consolidation. The porous medium is assumed to be linearly elastic, homogeneous and isotropic, and to be saturated by an incompressible fluid. According to Biot's theory [4], the mathematical model of a consolidation process is given by the following system of equations:

$$\text{equilibrium equation:} \quad \operatorname{div} \boldsymbol{\sigma}' - \alpha \nabla p = \mathbf{g}, \quad \text{in } \Omega, \tag{1}$$

$$\text{constitutive equation:} \quad \boldsymbol{\sigma}' = \lambda \operatorname{tr}(\boldsymbol{\epsilon}) \mathbf{I} + 2\mu \boldsymbol{\epsilon}, \quad \text{in } \Omega, \tag{2}$$

$$\text{compatibility condition:} \quad \boldsymbol{\epsilon}(\mathbf{u}) = \frac{1}{2}(\nabla \mathbf{u} + \nabla \mathbf{u}^t), \quad \text{in } \Omega, \tag{3}$$

$$\text{Darcy's law:} \quad \mathbf{q} = -\frac{\kappa}{\eta} \nabla p, \quad \text{in } \Omega, \tag{4}$$

$$\text{continuity equation:} \quad \nabla \cdot \mathbf{q} + \alpha \frac{\partial}{\partial t}(\nabla \cdot \mathbf{u}) = f, \quad \text{in } \Omega, \tag{5}$$

where Ω is an open bounded region of $\mathbb{R}^n$, $n \leq 3$, with regular boundary Γ, λ and μ are the so-called Lamé coefficients, α is the Biot-Willis constant which we will assume to be equal to one, κ is the permeability of the porous medium, and η is the viscosity of the fluid. $\boldsymbol{I}$ represents the identity tensor, $\mathbf{u}$ is the displacement vector, p is the pore pressure, $\boldsymbol{\sigma}'$ and $\boldsymbol{\epsilon}$ are the effective stress and strain tensors for the porous medium, and $\mathbf{q}$ is the percolation velocity of the fluid relative to the soil, where we ignore gravity effects. The source terms in the right-hand side $\mathbf{g}$ and f represent a density of applied body forces and a forced fluid extraction or injection process, respectively.

To complete the formulation of the problem, appropriate boundary conditions must be included. For instance, we can consider

$$p = 0, \quad \boldsymbol{\sigma}' \, \mathbf{n} = \mathbf{t}, \quad \text{on } \Gamma_t,$$
$$\mathbf{u} = \mathbf{0}, \quad \frac{\kappa}{\eta}(\nabla p) \cdot \mathbf{n} = 0, \quad \text{on } \Gamma_c, \tag{6}$$

where $\mathbf{n}$ is the unit outward normal to the boundary and $\Gamma_t \cup \Gamma_c = \Gamma$, with Γ_t and Γ_c disjoint subsets of Γ. At the initial time, the following incompressibility condition

$$\nabla \cdot \mathbf{u}\,(\mathbf{x}, 0) = 0, \quad \mathbf{x} \in \Omega, \tag{7}$$

is fulfilled. To establish the variational formulation of the problem, the following function spaces $\mathcal{Q} = \{q \in H^1(\Omega) \,|\, q = 0 \text{ on } \Gamma_t\}$, and $\mathcal{U} = \{\mathbf{u} \in (H^1(\Omega))^n \,|\, \mathbf{u} = \mathbf{0} \text{ on } \Gamma_c\}$, are considered, where $H^1(\Omega)$ is the well-known subspace of square integrable scalar-valued functions with also square integrable first derivatives. Denoting by $(\,\cdot\,,\,\cdot\,)$ the usual inner product between square integrable functions, and by introducing the bilinear forms

$$a(\mathbf{u}, \mathbf{v}) = 2\mu \sum_{i,j=1}^{n} (\epsilon_{ij}(\mathbf{u}), \epsilon_{ij}(\mathbf{v})) + \lambda(\nabla \cdot \mathbf{u}, \nabla \cdot \mathbf{v}), \quad b(p, q) = \frac{\kappa}{\eta} \sum_{i=1}^{n} \left(\frac{\partial p}{\partial x_i}, \frac{\partial q}{\partial x_i} \right),$$

where $\epsilon_{ij}(\mathbf{u})$ are the entries of the effective strain tensor $\boldsymbol{\epsilon}(\mathbf{u}) = \dfrac{1}{2}(\nabla \mathbf{u} + \nabla \mathbf{u}^t)$, the variational formulation of problem (1)–(5) with boundary and initial conditions given in (6) and (7) reads:

For each $t \in (0, T]$, find $(\mathbf{u}(t), p(t)) \in \mathcal{U} \times \mathcal{Q}$ such that

$$a(\mathbf{u}(t), \mathbf{v}) + (\nabla p(t), \mathbf{v}) = (\mathbf{g}, \mathbf{v}) + (\mathbf{t}, \mathbf{v})_{\Gamma_t}, \quad \forall\, \mathbf{v} \in \mathcal{U}, \tag{8}$$

$$(\frac{\partial}{\partial t}(\nabla \cdot \mathbf{u}(t)), q) + b(p(t), q) = (f, q), \quad \forall q \in \mathcal{Q}, \tag{9}$$

with the initial condition $(\nabla \cdot \mathbf{u}(0), q) = 0, \forall\, q \in L^2(\Omega)$, and where

$$(\mathbf{t}, \mathbf{v})_{\Gamma_t} = \int_{\Gamma_t} \mathbf{t} \cdot \mathbf{v} \, \mathrm{d}\Gamma.$$

The most common way to solve poroelasticity problems is to use finite element methods, see for example [15]. However, standard finite element discretizations give satisfactory solutions only when the solution is smooth, since when sharp pressure gradients appear, these methods turn out to be unstable in the sense that strong non-physical oscillations appear in the approximation of the pressure. This oscillatory behavior is minimized with the use of FEM methods satisfying the LBB stability condition [9], although these oscillations are not completely eliminated, and therefore other stabilization techniques are

necessary, see [14, 16, 17] for example. In [1], using continuous piecewise linear approximation spaces for displacements and pressure, a new stabilization based on the perturbation of the flow equation is given, providing solutions without oscillations independently of the chosen discretization parameters. Here, this stabilized finite element scheme will be used.

Let us consider a triangulation $\mathcal{T}_h$ of Ω, which is assumed to satisfy the usual admissibility assumption. Let $S_h^1 \subset H^1(\Omega)$ be the C^0 piecewise linear polynomial finite element space. Let be $\mathcal{U}_h = \mathcal{U} \bigcap (S_h^1 \times S_h^1)$ and $\mathcal{Q}_h = \mathcal{Q} \bigcap S_h^1$. By considering these finite dimensional spaces, of dimensions n_u and n_p, respectively, and using an implicit time discretization, the following discrete formulation of the considered problem is obtained:

For a time-step $k \geq 1$, find $(\mathbf{u}_h^k, p_h^k) \in \mathcal{U}_h \times \mathcal{Q}_h$ such that

$$a(\mathbf{u}_h^k, \mathbf{v}_h) + (\nabla p_h^k, \mathbf{v}_h) = (\mathbf{g}^k, \mathbf{v}_h) + (\mathbf{t}, \mathbf{v}_h)_{\Gamma_t}, \quad \forall \mathbf{v}_h \in \mathcal{U}_h, \tag{10}$$

$$(\nabla \cdot \mathbf{u}_h^k, q_h) + \tau\, b(p_h^k, q_h) = (\nabla \cdot \mathbf{u}_h^{k-1}, q_h) + \tau\, (f^k, q_h) \quad \forall q_h \in \mathcal{Q}_h, \tag{11}$$

where τ is the time discretization parameter.

Let $\widetilde{\boldsymbol{\varphi}_i}$ be a vector nodal basis of $\mathcal{U}_h$, with all its components equal to φ_i, and φ_j a nodal basis of $\mathcal{Q}_h$. As consequence, the discrete approximations at time-step k can be written as

$$\mathbf{u}_h{}^k = \sum_{i=1}^{n_u} \mathbf{U}_i^k \varphi_i, \quad p_h^k = \sum_{j=1}^{n_p} P_j^k \varphi_j,$$

and the discrete formulation (10)-(11), can be expressed as a system of linear algebraic equations as follows

$$\begin{bmatrix} A & G \\ D & \tau\, B \end{bmatrix} \begin{bmatrix} \mathbf{U}^k \\ \mathbf{P}^k \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ D & 0 \end{bmatrix} \begin{bmatrix} \mathbf{U}^{k-1} \\ \mathbf{P}^{k-1} \end{bmatrix} + \begin{bmatrix} \mathbf{G}^k \\ \tau\, \mathbf{F}^k \end{bmatrix}, \tag{12}$$

where $\mathbf{U}^k$ and $\mathbf{P}^k$ represent the unknown vectors in the $k^{\,th}$ time step: $(\mathbf{U}_1^k, \mathbf{U}_2^k, \ldots, \mathbf{U}_{n_u}^k)$ and $(P_1^k, P_2^k, \ldots, P_{n_p}^k)$, A is the elasticity matrix, B is the diffusive matrix multiplied by a coefficient κ/η, and G and D are the gradient and divergence matrices, respectively. $\mathbf{G}^k$ and $\mathbf{F}^k$ are the right hand side vectors in the $k^{\,th}$ time step, with components $\mathbf{G}_i^k = (\mathbf{g}^k, \widetilde{\boldsymbol{\varphi}_i}) + (\mathbf{t}, \widetilde{\boldsymbol{\varphi}_i})_{\Gamma_t}, i = 1, \ldots, n_u$ and $F_i^k = (f^k, \varphi_i), i = 1, \ldots, n_p$, respectively, and $D\mathbf{U}^0 = \mathbf{0}$.

As mentioned before,we consider the stabilized scheme presented in [1]. In such scheme, a term which arises from the discretization of the time derivative of the Laplacian of the pressure multiplied by a coefficient $\beta = h^2/4(\lambda + 2\mu)$, where h is the space discretization parameter, is added in the flow equation, and thus, the corresponding discrete

variational problem results in:

For $k \geq 1$, find $(\mathbf{u}_h^k, p_h^k) \in \mathcal{U}_h \times \mathcal{Q}_h$ such that

$$a(\mathbf{u}_h^k, \mathbf{v}_h) + (\nabla p_h^k, \mathbf{v}_h) = (\mathbf{g}^k, \mathbf{v}_h) + (\mathbf{t}, \mathbf{v}_h)_{\Gamma_t}, \quad \forall\, \mathbf{v}_h \in \mathcal{U}_h,$$
$$(\nabla \cdot \mathbf{u}_h^k, q_h) + (\tau + \beta')b(p_h^k, q_h) = \tau(f^k, q_h) + (\nabla \cdot \mathbf{u}_h^{k-1}, q_h) + \beta' b(p_h^{k-1}, q_h),\ \forall q_h \in \mathcal{Q}_h,$$

which in matrix-form reads as

$$\begin{bmatrix} A & G \\ D & (\tau+\beta')B \end{bmatrix} \begin{bmatrix} \mathbf{U}^k \\ \mathbf{P}^k \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ D & \beta' B \end{bmatrix} \begin{bmatrix} \mathbf{U}^{k-1} \\ \mathbf{P}^{k-1} \end{bmatrix} + \begin{bmatrix} \mathbf{G}^k \\ \tau\, \mathbf{F}^k \end{bmatrix}, \tag{13}$$

where $\beta' = \beta \dfrac{\eta}{\kappa}$.

3 Implementation on semi-structured triangular grids

The stabilized finite element scheme introduced in Section 2 for problem (1)-(5) is considered on a particular triangulation of the domain related to a semi-structured grid obtained by local regular refinement of an input unstructured triangulation. The semi-structured character of the grid allows the use of low-cost memory storage of the discrete operator based on stencil formulation. An efficient procedure to construct these stencils by means of a reference hexagon is presented further on.
Let us denote $\mathcal{T}_0$ a coarse triangulation of Ω, which is assumed to be fine enough in order to fit the geometry of the domain. Once this coarse triangulation is given, each one of its triangles is divided into four congruent triangles connecting the midpoints of their edges, and this is repeated until a mesh $\mathcal{T}_f$ is obtained with the desired fine scale. This strategy generates a hierarchy of meshes, $\mathcal{T}_0 \subset \mathcal{T}_1 \subset \cdots \subset \mathcal{T}_f$.

For the implementation of the finite element method, we wish to store the coefficient matrix using a stencil–wise procedure, since a few types of stencils are enough to describe the discrete operator. For this, we distinguish three different types of points in the grid: interior nodes of a triangle of the coarsest grid, vertices of $\mathcal{T}_0$ and nodes lying on the edges of $\mathcal{T}_0$, see Figure 1. Depending on the location of a node in the grid, the way in which the discrete operator is described is different. A stencil form for interior points to each coarse triangle and different stencil forms for nodes lying on the edges of $\mathcal{T}_0$, are enough, since both types of points have a regular structure. However, in order to describe the discrete operator in the nodes of $\mathcal{T}_0$, which is unstructured, it is necessary to construct the stiffness matrix on the coarsest grid, by the usual assembly process, which will be scaled depending on the refinement level we are working with. In fact, two different data structures must be used, one of them totally unstructured, whereas the other, corresponding to the most

of the nodes, is a hierarchical structure, see Figure 1. This methodology, see [3], resembles the way of working with finite difference methods on block–structured grids.

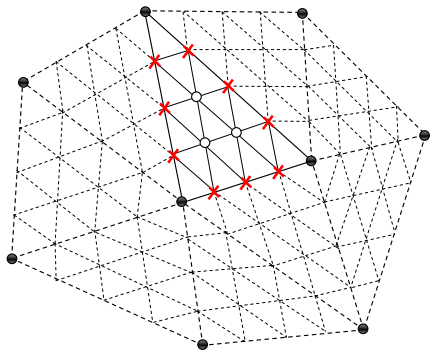

Figure 1.— Different kinds of nodes on a triangle of the coarsest grid: interior points (white circles), nodes lying on the edges (crosses), and vertexes of the unstructured grid (black circles).

Next, we concentrate on constructing the stencil associated with an interior point of a triangle $\mathcal{T}$ of the coarsest grid. To this end, we are going to define the regular grid arising inside this triangle. By considering a unitary basis of $\mathbb{R}^2$, $\{\mathbf{e'}_1, \mathbf{e'}_2\}$, fitting the geometry of the triangle, as we can see in Figure 2, we can define the grid for a refinement level $1 \leq \ell \leq f$ in the following way:

$$G_{\mathcal{T},\ell} = \{\mathbf{x} = (x,y)|_{\{\mathbf{e}'_i\}} \mid x = n\,h_1,\ y = m\,h_2,\ \ n = 0,\ldots,2^\ell,\ m = 0,\ldots,n\},$$

where $\mathbf{h} = (h_1, h_2)$ is the grid spacing in triangle $\mathcal{T}$, associated with the refinement level ℓ. Hence, a local numeration with double index can be fixed in each one of the triangles

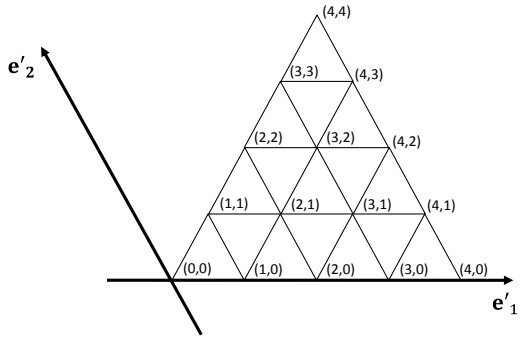

Figure 2.— New basis in $\mathbb{R}^2$ fitting the geometry of a triangle of the coarsest grid, and local numeration.

of the coarsest grid. For a refinement level ℓ, nodes are referred by (n,m), $n = 0,\ldots,2^\ell$, $m = 0,\ldots,n$, in such a way that the indexes of the vertices of the triangle are $(0,0)$,

$(2^\ell, 0)$, $(2^\ell, 2^\ell)$, as we can also observe in Figure 2 for $\ell = 2$. This way of numbering nodes is very convenient for identifying the neighboring nodes, which will be crucial when performing the geometric multigrid method.

For simplicity of presentation of the numerical scheme, we will consider only homogeneous boundary conditions. Spaces $\mathcal{U}$ and $\mathcal{Q}$ are the corresponding subspaces of functions in $(H^1(\Omega))^2$ and $H^1(\Omega)$, respectively, vanishing in the Dirichlet boundary, and the associated finite element spaces are built as in Section 2. Let us consider an interior node $\mathbf{x}_i$ to the grid $G_{\mathcal{T},\ell}$, associated with unknowns $\mathbf{u}_i^k$ and p_i^k. Since there is a biyective correspondence between the global and the local numeration, node $\mathbf{x}_i$ corresponds to an index (n, m), and therefore $\mathbf{u}_i^k$ and p_i^k can be denoted as $\mathbf{u}_\ell^k(\mathbf{x}_{n,m})$ and $p_\ell^k(\mathbf{x}_{n,m})$. Thus, the equations of system (13) corresponding to such interior point can be written in terms of discrete operators as follows:

$$A_\ell \mathbf{u}_\ell^k(\mathbf{x}_{n,m}) + G_\ell p_\ell^k(\mathbf{x}_{n,m}) = \mathbf{g}^k(\mathbf{x}_{n,m}), \tag{14}$$

$$D_\ell \mathbf{u}_\ell^k(\mathbf{x}_{n,m}) + (\tau + \beta) B_\ell p_\ell^k(\mathbf{x}_{n,m}) = D_\ell \mathbf{u}_\ell^{k-1}(\mathbf{x}_{n,m}) + \beta' B_\ell p_\ell^{k-1}(\mathbf{x}_{n,m}) + \tau f^k(\mathbf{x}_{n,m}),$$

where $\mathbf{u}_\ell^{k-1}(\mathbf{x}_{n,m})$ and $p_\ell^{k-1}(\mathbf{x}_{n,m})$ are known values, since represent the solution at previous time step, and A_ℓ, B_ℓ, G_ℓ, and D_ℓ denote the local discrete operators corresponding to an interior point of the considered triangle. As $\mathbf{u}_\ell^k(\mathbf{x}_{n,m}) = (u_\ell^k(\mathbf{x}_{n,m}), v_\ell^k(\mathbf{x}_{n,m}))^t$, operators A_ℓ and G_ℓ are vector discrete operators, which in stencil notation are given by

$$A_\ell = \begin{bmatrix} A_\ell^{11} & A_\ell^{12} \\ A_\ell^{21} & A_\ell^{22} \end{bmatrix}, \qquad G_\ell = \begin{bmatrix} G_\ell^x \\ G_\ell^y \end{bmatrix},$$

where

$$A_\ell^{ij} = \begin{bmatrix} 0 & a_{01}^{ij} & a_{11}^{ij} \\ a_{-10}^{ij} & a_{00}^{ij} & a_{10}^{ij} \\ a_{-1-1}^{ij} & a_{0-1}^{ij} & 0 \end{bmatrix}, \quad G_\ell^x = \begin{bmatrix} 0 & g_{01}^x & g_{11}^x \\ g_{-10}^x & g_{00}^x & g_{10}^x \\ g_{-1-1}^x & g_{0-1}^x & 0 \end{bmatrix}, \quad G_\ell^y = \begin{bmatrix} 0 & g_{01}^y & g_{11}^y \\ g_{-10}^y & g_{00}^y & g_{10}^y \\ g_{-1-1}^y & g_{0-1}^y & 0 \end{bmatrix}.$$

Analogously, D_ℓ is given by $\begin{bmatrix} D_\ell^x & D_\ell^y \end{bmatrix}$, where $D_\ell^x = G_\ell^x$ and $D_\ell^y = G_\ell^y$, whereas B_ℓ is a simple scalar discrete operator

$$B_\ell = \begin{bmatrix} 0 & b_{01} & b_{11} \\ b_{-10} & b_{00} & b_{10} \\ b_{-1-1} & b_{0-1} & 0 \end{bmatrix}.$$

Notice that each interior node is the center of a hexagon H, consisting of six congruent triangles, so the only unknowns appearing in the equations of node (n, m) are the corresponding to indexes $(n+1, m)$, $(n-1, m)$, $(n, m+1)$, $(n, m-1)$, $(n+1, m+1)$, $(n-1, m-1)$, see Figure 3. This allows us to write previous equations (14) in the following

way

$$\sum_{(\kappa_1,\kappa_2)\in\mathcal{I}} a^{11}_{\kappa_1,\kappa_2} u^k_\ell(\mathbf{x}_{n+\kappa_1,m+\kappa_2}) + \sum_{(\kappa_1,\kappa_2)\in\mathcal{I}} a^{12}_{\kappa_1,\kappa_2} v^k_\ell(\mathbf{x}_{n+\kappa_1,m+\kappa_2}) + \sum_{(\kappa_1,\kappa_2)\in\mathcal{I}} g^x_{\kappa_1,\kappa_2} p^k_\ell(\mathbf{x}_{n+\kappa_1,m+\kappa_2}) = \int_H g^k_1 \varphi_{n,m} \mathrm{d}\mathbf{x},$$

$$\sum_{(\kappa_1,\kappa_2)\in\mathcal{I}} a^{21}_{\kappa_1,\kappa_2} u^k_\ell(\mathbf{x}_{n+\kappa_1,m+\kappa_2}) + \sum_{(\kappa_1,\kappa_2)\in\mathcal{I}} a^{22}_{\kappa_1,\kappa_2} v^k_\ell(\mathbf{x}_{n+\kappa_1,m+\kappa_2}) + \sum_{(\kappa_1,\kappa_2)\in\mathcal{I}} g^y_{\kappa_1,\kappa_2} p^k_\ell(\mathbf{x}_{n+\kappa_1,m+\kappa_2}) = \int_H g^k_2 \varphi_{n,m} \mathrm{d}\mathbf{x},$$

$$\sum_{(\kappa_1,\kappa_2)\in\mathcal{I}} d^x_{\kappa_1,\kappa_2} u^k_\ell(\mathbf{x}_{n+\kappa_1,m+\kappa_2}) + \sum_{(\kappa_1,\kappa_2)\in\mathcal{I}} d^y_{\kappa_1,\kappa_2} v^k_\ell(\mathbf{x}_{n+\kappa_1,m+\kappa_2}) + (\tau+\beta) \sum_{(\kappa_1,\kappa_2)\in\mathcal{I}} b_{\kappa_1,\kappa_2} p^k_\ell(\mathbf{x}_{n+\kappa_1,m+\kappa_2}) =$$

$$\sum_{(\kappa_1,\kappa_2)\in\mathcal{I}} d^x_{\kappa_1,\kappa_2} u^{k-1}_\ell(\mathbf{x}_{n+\kappa_1,m+\kappa_2}) + \sum_{(\kappa_1,\kappa_2)\in\mathcal{I}} d^y_{\kappa_1,\kappa_2} v^{k-1}_\ell(\mathbf{x}_{n+\kappa_1,m+\kappa_2}) + \beta' \sum_{(\kappa_1,\kappa_2)\in\mathcal{I}} b_{\kappa_1,\kappa_2} p^{k-1}_\ell(\mathbf{x}_{n+\kappa_1,m+\kappa_2}) +$$

$$\tau \int_H f^k \varphi_{n,m} \mathrm{d}\mathbf{x}, \tag{15}$$

where $\mathcal{I}$ is the set $\mathcal{I} = \{(\kappa_1,\kappa_2) \,|\, \kappa_1,\kappa_2 = -1,0,1\} \subset \mathbb{Z}^2$.

In order to efficiently compute the previous stencils associated with discrete operators, we use a strategy similar to that used in the standard finite element assembly, taking in this case a reference hexagon. In order to illustrate such construction, we begin considering operator B_ℓ. The stencil form for operator B_ℓ reads

$$B_\ell = \frac{\kappa}{\eta} \cdot \begin{bmatrix} 0 & \displaystyle\int_{T_2\cup T_3} \nabla\varphi_{n,m+1}\cdot\nabla\varphi_{n,m}\,\mathrm{d}\mathbf{x} & \displaystyle\int_{T_1\cup T_2} \nabla\varphi_{n+1,m+1}\cdot\nabla\varphi_{n,m}\,\mathrm{d}\mathbf{x} \\ \displaystyle\int_{T_3\cup T_4} \nabla\varphi_{n-1,m}\cdot\nabla\varphi_{n,m}\,\mathrm{d}\mathbf{x} & \displaystyle\int_{\bigcup_{i=1}^6 T_i} \nabla\varphi_{n,m}\cdot\nabla\varphi_{n,m}\,\mathrm{d}\mathbf{x} & \displaystyle\int_{T_1\cup T_6} \nabla\varphi_{n+1,m}\cdot\nabla\varphi_{n,m}\,\mathrm{d}\mathbf{x} \\ \displaystyle\int_{T_4\cup T_5} \nabla\varphi_{n-1,m-1}\cdot\nabla\varphi_{n,m}\,\mathrm{d}\mathbf{x} & \displaystyle\int_{T_5\cup T_6} \nabla\varphi_{n,m-1}\cdot\nabla\varphi_{n,m}\,\mathrm{d}\mathbf{x} & 0 \end{bmatrix}, \tag{16}$$

where $T_i,\ i = 1,\ldots,6$ are the triangles composing the hexagon H around node (n,m), and the nodal basis functions $\varphi_{k,l}$ are referred to the local numeration, see Figure 3. In order to have an effective computation of this stencil we will use a reference hexagon $\hat{H}$ with center $\hat{\mathbf{x}}_{0,0} = (0,0)$ and vertices $\hat{\mathbf{x}}_{1,0} = (1,0)$, $\hat{\mathbf{x}}_{1,1} = (1,1)$, $\hat{\mathbf{x}}_{0,1} = (0,1)$, $\hat{\mathbf{x}}_{-1,0} = (-1,0)$, $\hat{\mathbf{x}}_{-1,-1} = (-1,-1)$, $\hat{\mathbf{x}}_{0,-1} = (0,-1)$, and an affine transformation F_H mapping hexagon $\hat{H}$ onto H, $\mathbf{x} = F_H(\hat{\mathbf{x}}) = \mathcal{B}_H\,\hat{\mathbf{x}} + b_H$ with

$$\mathcal{B}_H = \begin{pmatrix} x_{n+1,m} - x_{n,m} & x_{n+1,m+1} - x_{n+1,m} \\ y_{n+1,m} - y_{n,m} & y_{n+1,m+1} - y_{n+1,m} \end{pmatrix}, \quad b_H = \begin{pmatrix} x_{n,m} \\ y_{n,m} \end{pmatrix},$$

where $(x_{k,l}, y_{k,l})$ are the coordinates of the nodes $\mathbf{x}_{k,l}$. Note that matrix $\mathcal{B}_H$ is proportional with factor $2^{-\ell}$, where ℓ is the refinement level, to the matrix associated with the affine transformation between $\hat{T}_1$ (see Figure 3) and the current triangle of the input coarsest grid. With these definitions, we can translate the degrees of freedom and basis functions (denoted here by $\hat{\varphi}$) on the reference hexagon to degrees of freedom and basis functions

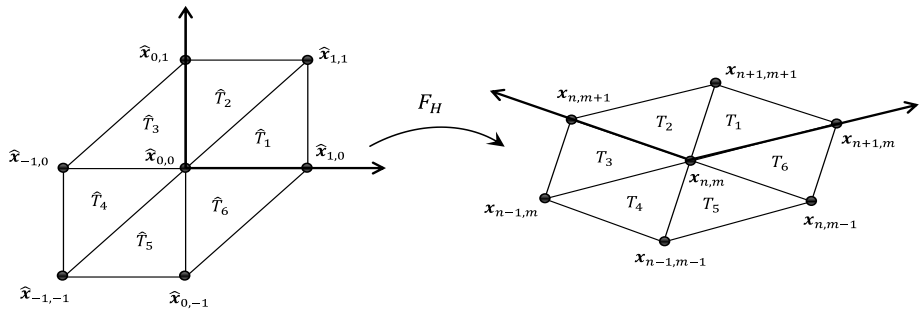

Figure 3.— Reference hexagon and corresponding affine transformation F_H.

on the hexagon H. In particular, we have

$$\hat{\varphi}_{k,l} = \varphi_{k,l} \circ F_H, \quad \nabla\hat{\varphi}_{k,l} = \mathcal{B}_H^t \nabla\varphi_{k,l} \circ F_H.$$

By applying the change of variable associated with the affine mapping, the entries of the stencil (16) yield the following expressions

$$b_{0,1} = |\det \mathcal{B}_H| \frac{\kappa}{\eta} \Big(\int_{\hat{T}_2} (\mathcal{B}_H^{-1})^t \nabla\hat{\varphi}_{0,1} \cdot (\mathcal{B}_H^{-1})^t \nabla\hat{\varphi}_{0,0} \mathrm{d}\hat{\mathbf{x}} + \int_{\hat{T}_3} (\mathcal{B}_H^{-1})^t \nabla\hat{\varphi}_{0,1} \cdot (\mathcal{B}_H^{-1})^t \nabla\hat{\varphi}_{0,0} \mathrm{d}\hat{\mathbf{x}} \Big),$$

$$b_{1,1} = |\det \mathcal{B}_H| \frac{\kappa}{\eta} \Big(\int_{\hat{T}_1} (\mathcal{B}_H^{-1})^t \nabla\hat{\varphi}_{1,1} \cdot (\mathcal{B}_H^{-1})^t \nabla\hat{\varphi}_{0,0} \mathrm{d}\hat{\mathbf{x}} + \int_{\hat{T}_2} (\mathcal{B}_H^{-1})^t \nabla\hat{\varphi}_{1,1} \cdot (\mathcal{B}_H^{-1})^t \nabla\hat{\varphi}_{0,0} \mathrm{d}\hat{\mathbf{x}} \Big),$$

$$b_{-1,0} = |\det \mathcal{B}_H| \frac{\kappa}{\eta} \Big(\int_{\hat{T}_3} (\mathcal{B}_H^{-1})^t \nabla\hat{\varphi}_{-1,0} \cdot (\mathcal{B}_H^{-1})^t \nabla\hat{\varphi}_{0,0} \mathrm{d}\hat{\mathbf{x}} + \int_{\hat{T}_4} (\mathcal{B}_H^{-1})^t \nabla\hat{\varphi}_{-1,0} \cdot (\mathcal{B}_H^{-1})^t \nabla\hat{\varphi}_{0,0} \mathrm{d}\hat{\mathbf{x}} \Big),$$

$$b_{0,0} = |\det \mathcal{B}_H| \frac{\kappa}{\eta} \Big(\sum_{i=1}^{6} \int_{\hat{T}_i} (\mathcal{B}_H^{-1})^t \nabla\hat{\varphi}_{0,0} \cdot (\mathcal{B}_H^{-1})^t \nabla\hat{\varphi}_{0,0} \mathrm{d}\hat{\mathbf{x}} \Big),$$

$$b_{1,0} = |\det \mathcal{B}_H| \frac{\kappa}{\eta} \Big(\int_{\hat{T}_1} (\mathcal{B}_H^{-1})^t \nabla\hat{\varphi}_{1,0} \cdot (\mathcal{B}_H^{-1})^t \nabla\hat{\varphi}_{0,0} \mathrm{d}\hat{\mathbf{x}} + \int_{\hat{T}_6} (\mathcal{B}_H^{-1})^t \nabla\hat{\varphi}_{1,0} \cdot (\mathcal{B}_H^{-1})^t \nabla\hat{\varphi}_{0,0} \mathrm{d}\hat{\mathbf{x}} \Big),$$

$$b_{-1,-1} = |\det \mathcal{B}_H| \frac{\kappa}{\eta} \Big(\int_{\hat{T}_4} (\mathcal{B}_H^{-1})^t \nabla\hat{\varphi}_{-1,-1} \cdot (\mathcal{B}_H^{-1})^t \nabla\hat{\varphi}_{0,0} \mathrm{d}\hat{\mathbf{x}} + \int_{\hat{T}_5} (\mathcal{B}_H^{-1})^t \nabla\hat{\varphi}_{-1,-1} \cdot (\mathcal{B}_H^{-1})^t \nabla\hat{\varphi}_{0,0} \mathrm{d}\hat{\mathbf{x}} \Big),$$

$$b_{0,-1} = |\det \mathcal{B}_H| \frac{\kappa}{\eta} \Big(\int_{\hat{T}_5} (\mathcal{B}_H^{-1})^t \nabla\hat{\varphi}_{0,-1} \cdot (\mathcal{B}_H^{-1})^t \nabla\hat{\varphi}_{0,0} \mathrm{d}\hat{\mathbf{x}} + \int_{\hat{T}_6} (\mathcal{B}_H^{-1})^t \nabla\hat{\varphi}_{0,-1} \cdot (\mathcal{B}_H^{-1})^t \nabla\hat{\varphi}_{0,0} \mathrm{d}\hat{\mathbf{x}} \Big).$$

Now, defining the 2×2 matrix $C_H = \mathcal{B}_H^{-1}(\mathcal{B}_H^{-1})^t$,

$$C_H = \begin{pmatrix} c_{11}^H & c_{12}^H \\ c_{12}^H & c_{22}^H \end{pmatrix},$$

the stencil (16) has the expression

$$B_\ell = |\det \mathcal{B}_H| \frac{\kappa}{\eta} \left(c_{11}^H \hat{S}_{xx} + 2\, c_{12}^H \hat{S}_{xy} + c_{22}^H \hat{S}_{yy} \right),$$

where

$$\hat{S}_{xx} = \begin{bmatrix} 0 & 0 & 0 \\ -1 & 2 & -1 \\ 0 & 0 & 0 \end{bmatrix}, \quad \hat{S}_{xy} = \hat{S}_{yx} = \frac{1}{2} \begin{bmatrix} 0 & 1 & -1 \\ 1 & -2 & 1 \\ -1 & 1 & 0 \end{bmatrix}, \quad \hat{S}_{yy} = \begin{bmatrix} 0 & -1 & 0 \\ 0 & 2 & 0 \\ 0 & -1 & 0 \end{bmatrix},$$

are the stencils associated with operators $-\partial_{xx}$, $-\partial_{xy}$, $-\partial_{yx}$, and $-\partial_{yy}$, in the reference hexagon.

Analogously, we can obtain similar expressions in function of these reference stencils, for discrete operators $A_\ell^{ij},\ i,j=1,2$, $G_\ell^x = D_\ell^x$, and $G_\ell^y = D_\ell^y$.
For the stencils corresponding to the elasticity operator A_ℓ, some of their coefficients are given by

$$\begin{aligned}
a_{01}^{11} &= \int_{T_2\cup T_3}\Big[(\lambda+2\mu)\frac{\partial\varphi_{n,m+1}}{\partial x}\frac{\partial\varphi_{n,m}}{\partial x}+\mu\frac{\partial\varphi_{n,m+1}}{\partial y}\frac{\partial\varphi_{n,m}}{\partial y}\Big]\,\mathrm{d}\mathbf{x},\\
a_{01}^{12} &= \int_{T_2\cup T_3}\Big[\lambda\frac{\partial\varphi_{n,m+1}}{\partial y}\frac{\partial\varphi_{n,m}}{\partial x}+\mu\frac{\partial\varphi_{n,m+1}}{\partial x}\frac{\partial\varphi_{n,m}}{\partial y}\Big]\,\mathrm{d}\mathbf{x},\\
a_{01}^{21} &= \int_{T_2\cup T_3}\Big[\lambda\frac{\partial\varphi_{n,m+1}}{\partial x}\frac{\partial\varphi_{n,m}}{\partial y}+\mu\frac{\partial\varphi_{n,m+1}}{\partial y}\frac{\partial\varphi_{n,m}}{\partial x}\Big]\,\mathrm{d}\mathbf{x},\\
a_{01}^{22} &= \int_{T_2\cup T_3}\Big[\mu\frac{\partial\varphi_{n,m+1}}{\partial x}\frac{\partial\varphi_{n,m}}{\partial x}+(\lambda+2\mu)\frac{\partial\varphi_{n,m+1}}{\partial y}\frac{\partial\varphi_{n,m}}{\partial y}\Big]\,\mathrm{d}\mathbf{x},
\end{aligned}$$

and the rest of them have analogous definitions. Using the change of variable previously introduced, and defining the inverse of the matrix of the transformation as:

$$\mathcal{B}_H^{-1} = \begin{pmatrix} b_{11}^H & b_{12}^H \\ b_{21}^H & b_{22}^H \end{pmatrix},$$

the four scalar stencils corresponding to A_ℓ can be written in terms of the reference stencils in the following way:

$$\begin{aligned}
A_\ell^{11} &= |\det\mathcal{B}_H|(((\lambda+2\mu)(b_{11}^H)^2+\mu(b_{12}^H)^2)\,\hat{S}_{xx}+(\mu(b_{22}^H)^2+(\lambda+2\mu)(b_{21}^H)^2)\,\hat{S}_{yy}\\
&\quad+((\lambda+2\mu)b_{11}^Hb_{21}^H+\mu b_{22}^Hb_{12}^H)\,(\hat{S}_{xy}+\hat{S}_{yx})),\\
A_\ell^{12} &= |\det\mathcal{B}_H|((\lambda+\mu)b_{11}^Hb_{12}^H\,\hat{S}_{xx}+(\lambda+\mu)b_{22}^Hb_{21}^H\,\hat{S}_{yy}+(\lambda b_{12}^Hb_{21}^H+\mu b_{22}^Hb_{11}^H)\,\hat{S}_{xy}\\
&\quad+(\lambda b_{22}^Hb_{11}^H+\mu b_{21}^Hb_{12}^H)\,\hat{S}_{yx}),\\
A_\ell^{21} &= |\det\mathcal{B}_H|((\lambda+\mu)b_{11}^Hb_{12}^H\,\hat{S}_{xx}+(\lambda+\mu)b_{22}^Hb_{21}^H\,\hat{S}_{yy}+(\lambda b_{22}^Hb_{11}^H+\mu b_{21}^Hb_{12}^H)\,\hat{S}_{xy}\\
&\quad+(\lambda b_{12}^Hb_{21}^H+\mu b_{22}^Hb_{11}^H)\,\hat{S}_{yx}),\\
A_\ell^{22} &= |\det\mathcal{B}_H|(((\lambda+2\mu)(b_{12}^H)^2+\mu(b_{11}^H)^2)\,\hat{S}_{xx}+(\mu(b_{21}^H)^2+(\lambda+2\mu)(b_{22}^H)^2)\,\hat{S}_{yy}\\
&\quad+((\lambda+2\mu)b_{22}^Hb_{12}^H+\mu b_{11}^Hb_{21}^H)\,(\hat{S}_{xy}+\hat{S}_{yx})),
\end{aligned}$$

Now, it is straightforward to see that the stencils corresponding to $G_\ell^x = D_\ell^x$ and $G_\ell^y = D_\ell^y$,

are given by

$$G_\ell^x = \begin{bmatrix} 0 & \displaystyle\int_{T_2\cup T_3} \frac{\partial \varphi_{n,m+1}}{\partial x}\,\varphi_{n,m}\,\mathrm{d}\mathbf{x} & \displaystyle\int_{T_1\cup T_2} \frac{\partial \varphi_{n+1,m+1}}{\partial x}\,\varphi_{n,m}\,\mathrm{d}\mathbf{x} \\ \displaystyle\int_{T_3\cup T_4} \frac{\partial \varphi_{n-1,m}}{\partial x}\,\varphi_{n,m}\,\mathrm{d}\mathbf{x} & \displaystyle\int_{\bigcup_{i=1}^6 T_i} \frac{\partial \varphi_{n,m}}{\partial x}\,\varphi_{n,m}\,\mathrm{d}\mathbf{x} & \displaystyle\int_{T_1\cup T_6} \frac{\partial \varphi_{n+1,m}}{\partial x}\,\varphi_{n,m}\,\mathrm{d}\mathbf{x} \\ \displaystyle\int_{T_4\cup T_5} \frac{\partial \varphi_{n-1,m-1}}{\partial x}\,\varphi_{n,m}\,\mathrm{d}\mathbf{x} & \displaystyle\int_{T_5\cup T_6} \frac{\partial \varphi_{n,m-1}}{\partial x}\,\varphi_{n,m}\,\mathrm{d}\mathbf{x} & 0 \end{bmatrix},$$

and

$$G_\ell^y = \begin{bmatrix} 0 & \displaystyle\int_{T_2\cup T_3} \frac{\partial \varphi_{n,m+1}}{\partial y}\,\varphi_{n,m}\,\mathrm{d}\mathbf{x} & \displaystyle\int_{T_1\cup T_2} \frac{\partial \varphi_{n+1,m+1}}{\partial y}\,\varphi_{n,m}\,\mathrm{d}\mathbf{x} \\ \displaystyle\int_{T_3\cup T_4} \frac{\partial \varphi_{n-1,m}}{\partial y}\,\varphi_{n,m}\,\mathrm{d}\mathbf{x} & \displaystyle\int_{\bigcup_{i=1}^6 T_i} \frac{\partial \varphi_{n,m}}{\partial y}\,\varphi_{n,m}\,\mathrm{d}\mathbf{x} & \displaystyle\int_{T_1\cup T_6} \frac{\partial \varphi_{n+1,m}}{\partial y}\,\varphi_{n,m}\,\mathrm{d}\mathbf{x} \\ \displaystyle\int_{T_4\cup T_5} \frac{\partial \varphi_{n-1,m-1}}{\partial y}\,\varphi_{n,m}\,\mathrm{d}\mathbf{x} & \displaystyle\int_{T_5\cup T_6} \frac{\partial \varphi_{n,m-1}}{\partial y}\,\varphi_{n,m}\,\mathrm{d}\mathbf{x} & 0 \end{bmatrix},$$

And following calculations similar to those for the previous operators, they can be written in terms of reference stencils in the following way

$$G_\ell^x = |\det \mathcal{B}_H| \left(b_{11}^H\, \hat{S}_x + b_{21}^H\, \hat{S}_y \right),$$
$$G_\ell^y = |\det \mathcal{B}_H| \left(b_{12}^H\, \hat{S}_x + b_{22}^H\, \hat{S}_y \right),$$

where

$$\hat{S}_x = \frac{1}{6} \begin{bmatrix} 0 & -1 & 1 \\ -2 & 0 & 2 \\ -1 & 1 & 0 \end{bmatrix}, \quad \hat{S}_y = \frac{1}{6} \begin{bmatrix} 0 & 2 & 1 \\ 1 & 0 & -1 \\ -1 & -2 & 0 \end{bmatrix},$$

are the stencils corresponding to operators ∂_x and ∂_y computed in the reference hexagon.

Finally, we normalize equations in (15) with the factor $|\det \mathcal{B}_H|$, and then the right–hand sides are approximations of $\mathbf{g}^k(\mathbf{x}_{n,m})$ and $f^k(\mathbf{x}_{n,m})$. Notice that these stencil forms of the discrete operators in function of the reference stencils give the stencil corresponding to an interior point of an arbitrary triangle with any geometry and any position in the plane.

With obvious modifications of the above process, it is possible to construct the stencil associated with the nodes located at the edges in $\mathcal{T}_0$ in terms of the basic stencils and the appropriate affine transfonmations.

4 Multigrid based on Vanka-type smoothers

The design of an efficient geometric multigrid method for a concrete problem depends strongly on the choice of adequate components of the algorithm. These components have to be chosen so that they efficiently interplay with each other in order to obtain a good

connection between the relaxation and the coarse-grid correction. In this paper, linear interpolation is chosen as the prolongation, and its adjoint as the restriction operator. The discrete operator on each mesh in the hierarchy results from the direct discretization of the system of partial differential equations on the corresponding grid. Due to the semi-structured character of the grid, we will use a block-wise multigrid algorithm, in which each triangle of the coarsest triangulation is treated as a different block with regard to the smoothing process. However, there are several points in the algorithm, where information from neighboring triangles must be transferred, and to facilitate this communication each triangle of the coarsest grid is augmented by an overlap-layer of so-called ghost nodes that surround it.

As commented in the introduction, standard smoothers, as simple point-wise gauss-seidel smoothers (with any ordering), are not appropriate for saddle point type problems. We shall see that box-relaxation can be taken as a suitable smoother to deal with poroelasticity problems. It consists of decomposing the grid into small overlapped subdomains and looping over all of them solving the system arising from the equations corresponding to the points in the subdomain. There are many variants of box-type smoothers, they can differ in the choice of the subdomains which are solved simultaneously, and in the way in which the local systems are solved. Also, the different subdomains can be visited in differente orderings, for example red-black or three-color ordering (see [11]), yielding to a wide variety of box-relaxations. Here, we only deal with a pair of them. Firstly, we consider a point-wise box Gauss-Seidel iterative algorithm, which consists of simultaneously updating all unknowns corresponding to the nodes located at the vertexes of an hexagon centered on a grid point, together with the unknowns at this point. This means that 21 unknowns corresponding to displacement and pressure unknowns (see left Figure 4) are relaxed simultaneously and therefore, a 21×21 system has to be solved for each box. In the variant considered here, the subdomains are visited in lexicographic order. The need of solving such systems makes these smoothers expensive. A simplified variant of them can be considered by only coupling the unknowns associated with a cell in the grid, that is, unknowns located at the three vertexes of each triangle, see right Figure 4. This implies to solve a 9×9 system on each triangular cell. This smoother is known as cell-wise box relaxation. In this paper, a variant of this cell-wise box smoother is considered, in particular a red-black cell-wise box-relaxation is used. It consists of looping the triangular cells of the grid in a checkerboard manner, that is, first the up-oriented triangles are updated, and the down-oriented ones are relaxed in the second partial relaxation step.

In order to see the suitability of box-smoothers for the considered problem, we solve the so-called poroelasticity footing problem on the computational rectangular domain depicted in left part of Figure 5, with dimensions $\Omega = [0, 1] \times [0, \sqrt{3}/2]$.

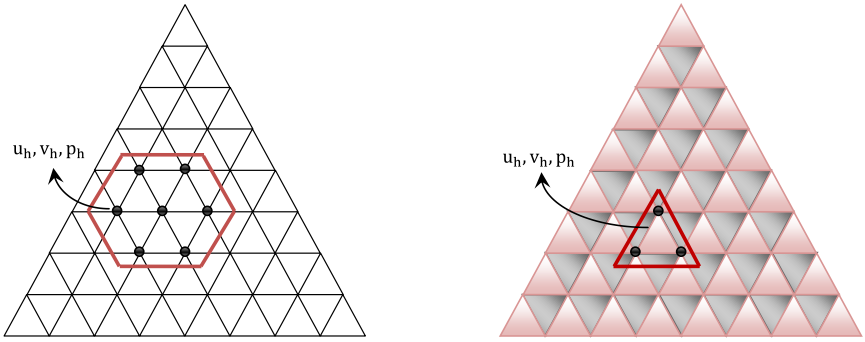

Figure 4.— Unknowns simultaneously updated in point-wise and cell-wise box Gauss-Seidel.

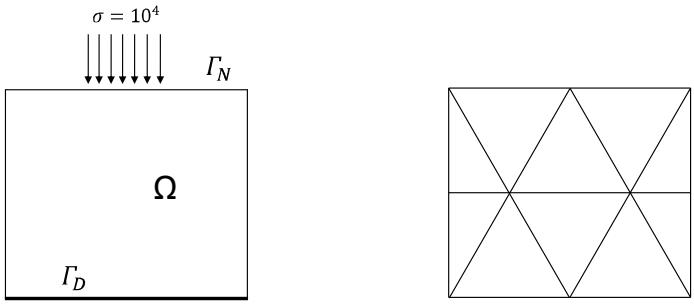

Figure 5.— Rectangular computational domain, and considered coarsest triangulation.

The body is assumed rigid at the bottom and a uniform load is applied in a strip of length 0.4, in the central part of the upper boundary. Besides, the whole contour is assumed free to drain. More concretely, the considered boundary conditions are the following

$$p = 0, \text{ on } \Gamma = \partial\Omega,$$
$$\boldsymbol{\sigma}' \mathbf{n} = \mathbf{t}, \text{ on } \Gamma_{t,1},$$
$$\boldsymbol{\sigma}' \mathbf{n} = \mathbf{0}, \text{ on } \Gamma_{t,2},$$
$$\mathbf{u} = \mathbf{0}, \text{ on } \Gamma_c = \Gamma \backslash \{\Gamma_{t,1} \cup \Gamma_{t,2}\},$$

where $\mathbf{t} = (0, -10^4)^t$, and

$$\Gamma_{t,1} = \{(x, y) \in \Gamma \,|\, y = \sqrt{3}/2,\ 0.3 \leq x \leq 0.7\},$$

$$\Gamma_{t,2} = \{(x, y) \in \Gamma \,|\, y = \sqrt{3}/2,\ 0 \leq x \leq 0.3 \text{ or } 0.7 \leq x \leq 1\},$$

and the material properties of the porous medium are given in Table 1. The considered time-step is $\tau = 10^{-2}$.

The coarsest triangulation, composed of 10 triangles, is also depicted in right part of Figure 5. From this grid, a regular refinement process is applied to each element of the

Property	Value	Unit
Young's modulus	3×10^4	N/m^2
Poisson's ratio	0.2	-
Permeability	10^{-10}	m^2
Fluid viscosity	10^{-3}	Pas

Table 1.— Material parameters for the considered poroelastic problem.

triangulation until to achieve a target grid with the desired fine scale to approximate the solution of the problem. For all the numerical experiments performed next, the stopping criterion per time step is that the absolute residual should be less than 10^{-6}.

First of all, a standard smoother is considered in order to see the convergence troubles that appear when trying to solve the system of poroelasticity with the corresponding multigrid method. In particular, a three-color Gauss-Seidel is considered, since in [11] this smoother resulted to have a better performance than other standard smoothers for the Poisson problem. Three color smoother consists of splitting grid nodes into three disjoint sets, with each set having a different color, and simultaneously updating all nodes of the same color. The good convergence properties displayed by this smoother for the Poisson problem are lost when it is used to solve a poroelasticity problem. This can be seen in Figure 6, where the history of the convergence of a multigrid algorithm based on three-color smoother, applied to the considered poroelasticity problem, is shown. An $F(2,1)$ cycle is used, and for different numbers of refinement levels it is shown that this smoother is not robust with respect to the space discretization parameter, since the number of iterations, necessary to reach the desired value for the residual, grows up as the number of refinement levels increase, yielding a significative deterioration of the convergence. Besides, even divergence can be seen for some refinement levels.

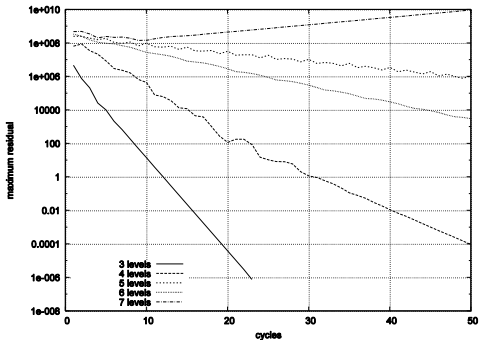

Figure 6.— History of the convergence of multigrid based on three-color smoother for poroelasticity.

This unsatisfactory convergence of the multigrid method, is improved by considering box-relaxation. Next, results for both box-smoothers previously introduced, are presented. In Figure 7, the obtained results for the convergence of the multigrid method based on cell-wise (right-hand panel) and point-wise box-smoothers (left-hand panel) are displayed. It is observed a very good behavior of these methods for the poroelasticity problem, and it can be seen that even the convergence improves when a finer grid is considered. Besides, a similar behavior of both smoothers is reported, being cheaper the cell-wise box-relaxation.

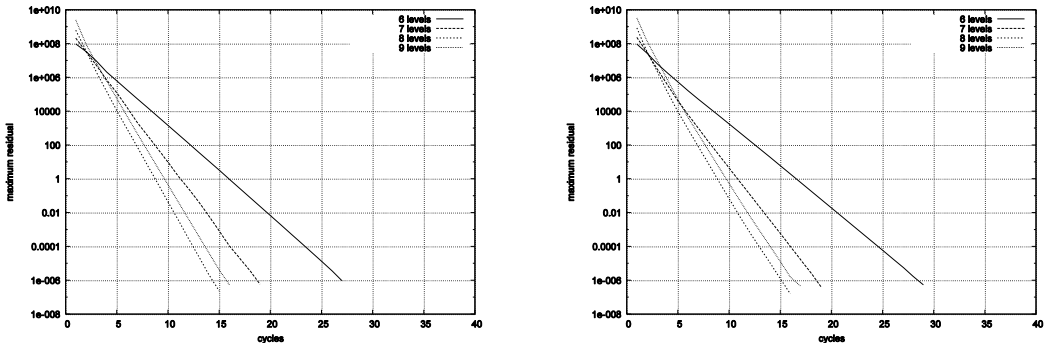

Figure 7.— Convergence of point-wise and cell-wise box-smoothers, respectively, for different numbers of refinement levels.

Finally, the behavior of $V-$ and $F-$cycles for cell-wise and point-wise box-smoothers is analyzed. In Figure 8, the convergence obtained with $V(2,1)$ and $F(2,1)-$ cycles is displayed for both smoothers. It is observed that $F-$ cycles provide good convergence, whereas $V-$cycles yield to divergence of the method. This is due to the poor coarse-grid correction which appears when stabilization terms are added to the equations. $F-$cycles, which invest more work on coarser grids, can overcome these troubles, but on the contrary $V-$cycles do not manage it. Moreover, an increase on the number of pre- and post-smoothing steps does not improve significantly the results. Some techniques to beat these problems, as residual overweighting and defect correction strategies, see [8], will be investigated in the future.

References

[1] Aguilar, G., Gaspar, F., Lisbona, F., Rodrigo, C.: 2008, "Numerical stabilization of Biot's consolidation model by a perturbation on the flow equation", *Int. J. Numer. Meth. Engng.* **75**, 1282–1300.

[2] Benzi, M., Golub, G.H., Liesen, J.: 2005, "Numerical solution of saddle point problems", *Acta Numerica* **14**, 1–137. Cambridge University Press, United Kingdom.

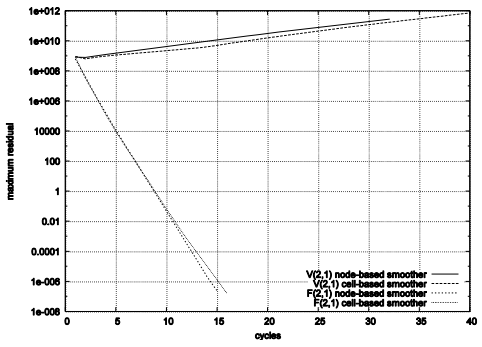

Figure 8.— Convergence of point-wise and cell-wise box-smoothers, respectively, for $F(2,1)$ and $V(2,1)$ cycles.

[3] Bergen, B., Gradl, T., Hülsemann, F., Rüde, U.: 2006, "A massively parallel multigrid method for finite elements". *Comput. Sci. Eng.* **8**, 56–62.

[4] Biot, M.: 1941, "General theory of three dimensional consolidation". *J. Appl. Phys.* **12**, 155–169.

[5] Biot M.: 1955, "Theory of elasticity and consolidation for a porous anisotropic solid". *J. Appl. Phys.* **33**:182–185.

[6] Biot M.: 1956 "General solutions of the equation of elasticity and consolidation for a porous material". *J. Appl. Mech.* **78**:91–96.

[7] Brandt, A.: 1977, "Multi-level adaptive solutions to boundary-value problems". *Math. Comput.* **31**, 333–390.

[8] Brandt, A., Yavneh, I.: 1993, "Accelerated multigrid convergence and high-Reynolds recirculating flows". *SIAM J. Sci. Comput.* **14**, 607–626.

[9] Brezzi, F.: 1974, "On the existence, uniqueness and approximation of saddle-point problems arising from Lagrange multipliers". *RAIRO Modelisation Mathematique et Analyse Numerique* **8**, 129–151.

[10] Gaspar, F.J., Lisbona, F.J., Oosterlee, C.W., Wienands, R.: 2004, "A systematic comparison of coupled and distributive smoothing in multigrid for the poroelasticity system". *Numer. Linear Algebra Appl.* **11**, 93–113.

[11] Gaspar, F.J., Gracia, J.L., Lisbona, F.J.: 2009, "Fourier analysis for multigrid methods on triangular grids". *SIAM J. Sci. Comput.* **31**, 2081–2102.

[12] Hackbusch, W.: 1985, *Multi-grid methods and applications*. Springer, Berlin.

[13] John, V., Tobiska, L.: 2000, "Numerical performance of smoothers in coupled multigrid methods for the parallel solution of the incompressible Navier-Stokes equations". *Int. J. Numer. Meth. Fluids* **33**, 453–473.

[14] Korsawe, J., Starke, G., Wang, W., Kolditz, O.: 2006, "Finite element analysis of poro-elastic consolidation in porous media: standard and mixed approaches.". *Computer Methods in Applied Mechanics and Engineering* **195**, 1096–1115.

[15] Lewis, R.W., Schrefler B.A.: 1998, *The Finite Element Method in the Static and Dynamic Deformation and Consolidation of Porous Media*. Wiley, New York.

[16] Mira, P., Pastor M., Li, T., Liu, X.: 2003, "A new stabilized enhanced strain element with equal order of interpolation for soil consolidation problems". *Computer Methods in Applied Mechanics and Engineering* **192**, 4257–4277.

[17] Pastor M., Li, T., Liu, X., Zienkiewicz, O.C.: 1999, "Stabilized low-order finite elements for failure and localization problems in undrained soils and foundations". *Computer Methods in Applied Mechanics and Engineering* **174**, 219–234.

[18] Terzaghi, K.: 1943, *Theoretical Soil Mechanics*. John Wiley, New York.

[19] Trottenberg, U., Oosterlee, C.W., Schüller, A.: 2001, *Multigrid*. Academic Press, New York.

[20] Turek, S.: 1999, *Efficient solvers for incompressible flow problems: an algorithmic and computational approach*. Springer, Berlin.

[21] Vanka, S.P.: 1986, "Block-implicit multigrid solution of Navier-Stokes equations in primitive variables". *J. Comput. Phys.* **65**, 138–158.

[22] Wobker, H., Turek, S.: 2009, "Numerical studies of Vanka-type smoothers in computational solid mechanics". *Adv. Appl. Math. Mech.* **1**, 29–55.