Numerical modelling of void inclusions in porous media

Antonio Rodríguez-Ferran*, Josep Sarrate and Antonio Huerta

Departament de Matemàtica Aplicada III, E.T.S. de Ingenieros de Caminos

Edifici C2, Campus Nord, Universitat Politècnica de Catalunya

E-08034 Barcelona, Spain. web page: www-lacan.upc.es

SUMMARY

Numerical modelling of porous flow in a low-permeability matrix with high-permeability inclusions is a challenging task, because the large ratio of permeabilities ill-conditions the finite element system of equations. We propose a coupled model where Darcy flow is used for the porous matrix and potential flow for the inclusions. We discuss appropriate interface conditions in detail and show that the head drop in the inclusions can be prescribed in a very simple way. Algorithmic aspects are treated in full detail. Numerical examples show that this coupled approach precludes ill-conditioning and is more efficient than heterogeneous Darcy flow.

KEY WORDS: porous media; Darcy theory; potential flow; large permeability; coupled model

---

*Correspondence to: Antonio Rodríguez-Ferran, Departament de Matemàtica Aplicada III, E.T.S. de Ingenieros de Caminos. Edifici C2, Campus Nord, Universitat Politècnica de Catalunya. E-08034 Barcelona, Spain.

1. INTRODUCTION

Modelling flow in porous media is needed in many fields of civil and mechanical engineering, such as geotechnics, subsurface hydrology, pollution emission control and water treatment. A basic ingredient is Darcy theory, which establishes the proportionality, through a permeability factor, between flow and gradient of piezometric head [1, 2].

In some applications, the porous domain contains voids (i.e. zones without the porous skeleton) or "quasi-voids" (i.e. porous zones with a much higher permeability, several orders of magnitude larger). This is the case, for instance, of binary media consisting of a low-permeability matrix with high-permeability inclusions [3]. In saturated flow conditions, both the voids and the pores in the porous skeleton are filled with the fluid under consideration.

One possible approach to model these inclusions is to use a very high permeability (the real permeability for high-permeability inclusions or a fictitious value for voids). However, it is a well-known fact that the large variations in permeability lead to ill-conditioning of the resulting linear system of equations arising from finite element discretization [4]. The large condition number affects negatively the convergence properties of iterative solvers.

Because of this, a different approach is proposed here: to model inclusions with potential flow instead of porous media theory. This approach has three clear advantages: (1) no large permeability is assumed for the inclusions, so there is no ill-conditioning of the matrix and the performance of the iterative solvers greatly improves; (2) the flow in the porous medium and in the inclusions is computed separately so the computational efficiency increases (two smaller systems of equations are solved instead of a single larger system); (3) localized head drops in the inclusions can be prescribed in a very simple way.

The proposed approach only requires one additional physical hypothesis: head is uniform

in the interface of the inclusion with the porous medium. This poses no restriction on the applicability of the proposed approach, because it is equivalent to considering the inclusion as an infinitely permeable porous medium. The two flow problems (i.e. in the porous medium and in the inclusions)are connected simply by imposing the continuity of flow in the interface.

The outline of paper follows. The flow in the porous medium in the presence of inclusions is covered in section 2. First, section 2.1 treats inclusions with no head drop; then section 2.2 describes how to prescribe the head drop in the inclusions. Section 3 deals with the flow in the inclusions. The computational aspects are discussed in section 4. The proposed approach is illustrated in section 5 by means of three numerical examples. The concluding remarks of section 6 close the paper.

2. FLOW IN THE POROUS MEDIUM

Flow in porous media is modelled by means of Darcy equation [1, 2]

$$\mathbf{v}_{\text{por}} = -k\nabla h \tag{1}$$

where h is the piezometric head, k is the permeability, $\mathbf{v}_{\text{por}}$ is the Darcy velocity and ∇ is the usual nabla operator. An isotropic permeability, represented by the scalar k, is assumed for presentation purposes. The approach proposed here, however, can also be applied to the general case of anisotropic permeability, represented by a symmetric positive-definite permeability tensor.

Assuming there are no sinks or sources, replacing equation (1) into the continuity equation results in

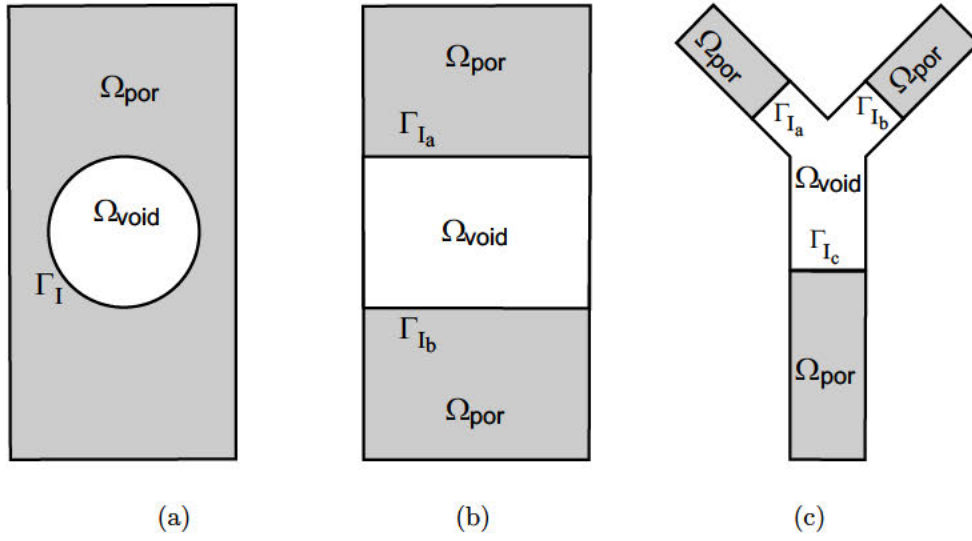$$\nabla \cdot (k\nabla h) = 0 \tag{2}$$

Figure 1. Inclusions with (a) one, (b) two and (c) three interfaces with the porous medium

For simplicity, only two types of boundary conditions for equation (2) are considered: prescribed head (non-homogeneous Dirichlet) and zero flux (homogeneous Neumann)

$$h = h^{\text{pres}} \qquad \text{in } \Gamma_D \qquad (3)$$

$$\mathbf{v}_{\text{por}} \cdot \mathbf{n} = 0 \qquad \text{in } \Gamma_N \qquad (4)$$

where h^{pres} is a given prescribed head and $\mathbf{n}$ is the outward unit normal. Other boundary conditions, such as prescribed fluxes or mixed conditions, can be incorporated in a straightforward manner.

2.1. Treatment of inclusions with no head drop

Let us consider a porous medium with an inclusion, see figure 1.

One possible approach to solve the problem is to treat the domain in figure 1 as an heterogeneous porous medium, with a very large permeability in the inclusion (corresponding

to the real permeability in the case of quasi-voids and to a fictitious porous material in the case of voids). However, the large difference in permeabilities between Ω_{por} and Ω_{void} leads to ill-conditioning in the finite element equations.

A different approach is proposed here. The porous and void domains are treated separately. First, the flow in the porous medium Ω_{por} is computed. To do so, only one additional hypothesis is required: the piezometric head h is assumed to be uniform in the interface Γ_{I}:

$$h = h^{\text{inter}} \qquad \text{in } \Gamma_{\text{I}} \tag{5}$$

Note that h^{inter} is unknown and that Γ_{I} can be disjoint, see figures 1(b) and 1(c).

Hypothesis (5) is physically sound; it amounts to regarding the inclusion as a porous material with an infinite permeability, as discussed in the appendix I and illustrated in section 5 with some numerical examples.

Finite element discretization of equation (2) yields

$$\mathbf{K}_{\text{por}}\mathbf{h} = \mathbf{0} \tag{6}$$

where $\mathbf{h}$ is the vector of nodal head values and $\mathbf{K}_{\text{por}}$ is the permeability matrix. This matrix is singular because the constraints on $\mathbf{h}$ are still not accounted for. These constraints are linear and can be written in matrix format like

$$\mathbf{C}_{\text{por}}\mathbf{h} = \mathbf{b}_{\text{por}} \tag{7}$$

where $\mathbf{C}_{\text{por}}$ is a rectangular matrix with `numcon` (number of constraints) rows and `numnp` (number of nodal points) columns, and $\mathbf{b}_{\text{por}}$ is a vector of `numcon` prescribed values.

There are two types of linear constraints in our problem:

1. Dirichlet boundary conditions. The discrete version of equation (3) is

$$h_I = h_I^{\text{pres}} \qquad \text{for all } I \in \mathcal{D} \tag{8}$$

where $\mathcal{D}$ indexes the nodes in Γ_{D}. The entries in $\mathbf{C}_{\mathrm{por}}$ and $\mathbf{b}_{\mathrm{por}}$ are $c_{II} = 1$ and $b_I = h_I^{\mathrm{pres}}$ for $I \in \mathcal{D}$.

2. Uniform head at the interface. The discrete version of equation (5) is

$$h_I = h^{\mathrm{inter}} \qquad \text{for all } I \in \mathcal{I} \tag{9}$$

where $\mathcal{I}$ indexes the nodes in Γ_{I}. Since h^{inter} is an unknown, this equation cannot be directly used to define the entries in $\mathbf{C}_{\mathrm{por}}$ and $\mathbf{b}_{\mathrm{por}}$. First, it must be rewritten into

$$h_I - h_R = 0 \qquad \text{for an arbitrary } R \in \mathcal{I} \text{ and for all } I \in \mathcal{I} - \{R\} \tag{10}$$

Equation (10) simply states that the value of h in all the nodes in Γ_{I} must be the same than for an arbitrary, reference node R. The entries in $\mathbf{C}_{\mathrm{por}}$ and $\mathbf{b}_{\mathrm{por}}$ are $c_{II} = 1$, $c_{IR} = -1$, $b_I = 0$.

Imposing the linear constraints (7) into the singular system (6) renders a regular system. Two strategies may be used: the Lagrange-multiplier technique or an ad-hoc transformation method. These techniques are discussed in detail in section 4.

Remark 1. *For simplicity, the case of a single inclusion has been assumed in this section, but the extension to multiple inclusions is straightforward. Consider for instance the case of two inclusions, with interfaces Γ_{I_1} and Γ_{I_2}. The linear constraints are*

$$h_I - h_{R_1} = 0 \qquad \text{for an arbitrary } R_1 \in \mathcal{I}_1 \text{ and for all } I \in \mathcal{I}_1 - \{R_1\}$$

$$h_I - h_{R_2} = 0 \qquad \text{for an arbitrary } R_2 \in \mathcal{I}_2 \text{ and for all } I \in \mathcal{I}_2 - \{R_2\}$$

where $\mathcal{I}_1$ and $\mathcal{I}_2$ index the nodes in Γ_{I_1} and Γ_{I_2} respectively.

2.2. Prescribing the head drop in the inclusions

Up to now, we have assumed there is no head drop in the inclusions, see equation (5). With the proposed approach, however, it is very simple to prescribe a non-zero head drop in each inclusion. This is needed in some applications to model the localized head drop in the inclusion (related, for instance, to the shape and tortuosity of the inclusion, or to the use of fabrics of low permeability to materialize the interface between the porous medium and the inclusion in industrial devices).

Consider, for instance, the inclusion of figure 1(b), with a disjoint interface $\Gamma_\mathrm{I} = \Gamma_{\mathrm{I_a}} \cup \Gamma_{\mathrm{I_b}}$. To prescribe the head drop in the inclusion, the constraints

$$\left.\begin{aligned} h &= h^{\mathrm{inter_a}} && \text{in } \Gamma_{\mathrm{I_a}} \\ h &= h^{\mathrm{inter_b}} && \text{in } \Gamma_{\mathrm{I_b}} \\ \Delta h &= h^{\mathrm{inter_b}} - h^{\mathrm{inter_a}} \end{aligned}\right\} \tag{11}$$

are needed, where $h^{\mathrm{inter_a}}$ and $h^{\mathrm{inter_b}}$ are the unknown uniform piezometric heads at $\Gamma_{\mathrm{I_a}}$ and $\Gamma_{\mathrm{I_b}}$ respectively and Δh is the prescribed head drop in the inclusion.

Following the same arguments of section 2.1, the constraints (11) can be recast as

$$\left.\begin{aligned} h_I - h_{\widehat{R}} &= 0 && \text{for an arbitrary } \widehat{R} \in \mathcal{I}_\mathrm{a} \text{ and for all } I \in \mathcal{I}_\mathrm{a} - \{\widehat{R}\} \\ h_I - h_{\widehat{R}} &= \Delta h && \text{for all } I \in \mathcal{I}_\mathrm{b} \end{aligned}\right\} \tag{12}$$

where $\mathcal{I}_\mathrm{a}$ and $\mathcal{I}_\mathrm{b}$ index the nodes in $\Gamma_{\mathrm{I_a}}$ and $\Gamma_{\mathrm{I_b}}$ respectively. The non-zero entries in matrix $\mathbf{C}_{\mathrm{por}}$ and vector $\mathbf{b}_{\mathrm{por}}$, see equation (7), are $c_{II} = 1$ and $c_{I\widehat{R}} = -1$ for an arbitrary $\widehat{R} \in \mathcal{I}_\mathrm{a}$ and for all $I \in \mathcal{I}_\mathrm{a} - \{\widehat{R}\} \cup \mathcal{I}_\mathrm{b}$ and $b_I = \Delta h$ for all $I \in \mathcal{I}_\mathrm{b}$.

3. FLOW IN THE VOID INCLUSIONS

Once the piezometric head h is obtained, the velocity field $\mathbf{v}_{\mathrm{por}}$ in the porous medium can be computed from equation (1). In many applications, this information is enough. This is the case, for instance, if the goal of the analysis is determining the input and output flows in the Dirichlet boundaries.

For other applications, on the contrary, the velocity field in the inclusions is also needed. When dealing with transport phenomena, for instance, the velocity in the whole domain, not only in the porous medium, is required to represent convective transport. This section describes how to determine the velocity field in the inclusions.

Assuming a potential flow, the velocity in the void inclusion $\mathbf{v}_{\mathrm{void}}$ is expressed as

$$\mathbf{v}_{\mathrm{void}} = -\nabla\phi \tag{13}$$

where ϕ is the potential. Replacement of equation (13) in the continuity equation results in the Laplacian equation

$$\nabla^2\phi = 0 \tag{14}$$

The boundary condition for this equation is derived by prescribing the continuity of flux in the interface Γ_{I}:

$$\mathbf{v}_{\mathrm{por}} \cdot \mathbf{n}_{\mathrm{por}} + \mathbf{v}_{\mathrm{void}} \cdot \mathbf{n}_{\mathrm{void}} = 0 \tag{15}$$

where $\mathbf{n}_{\mathrm{por}}$ and $\mathbf{n}_{\mathrm{void}}$ are the unit outward normals in the interface associated to the porous medium and the void inclusion ($\mathbf{n}_{\mathrm{void}} = -\mathbf{n}_{\mathrm{por}}$), see figure 2. This same equation is used in [5] to couple Navier-Stokes and Darcy models.

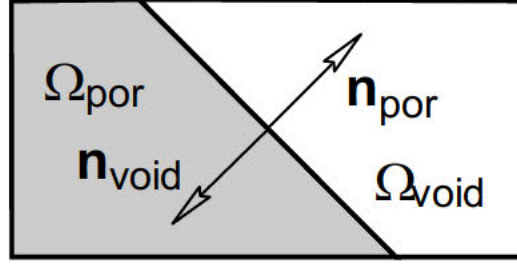By using the gradient relations (1) and (13), equation (15) can be written as a non-

Figure 2. Unit outward normals at the interface

homogeneous Neumann boundary condition

$$-\nabla\phi \cdot \mathbf{n}_{\text{void}} = k\nabla h \cdot \mathbf{n}_{\text{por}} \qquad \text{in } \Gamma_{\text{I}} \tag{16}$$

where the RHS is a known quantity because the flow in the void inclusions is computed, if desired, *after* obtaining h in the porous medium.

An homogeneous Neumann condition is needed in the exterior boundary of the inclusion

$$\mathbf{v}_{\text{void}} \cdot \mathbf{n} = 0 \qquad \text{in } \Gamma_{\text{N}} \tag{17}$$

Note that the solution of the Laplacian equation (14) is not unique if only Neumann boundary conditions are prescribed. To avoid this indetermination, the potential ϕ must be set to zero in an arbitrary point of the domain

$$\phi = 0 \qquad \text{in an arbitrary } p \in \Omega_{\text{void}} \tag{18}$$

Finite element discretization of the partial differential equation (14) and its boundary conditions (16), (17) and (18) yields

$$\left.\begin{aligned} \mathbf{K}_{\text{void}}\phi &= \mathbf{f}_{\text{void}} \\ \mathbf{C}_{\text{void}}\phi &= \mathbf{0} \end{aligned}\right\} \tag{19}$$

where $\mathbf{K}_{\text{void}}$ is the Laplacian matrix, ϕ is the vector of nodal potential values, $\mathbf{f}_{\text{void}}$ is the vector of nodal fluxes associated to the RHS of equation (16) and $\mathbf{C}_{\text{void}}$ is the constraint matrix. The

discrete counterpart of equation (18) is simply $\phi_R = 0$ for an arbitrary node R, so $\mathbf{C}_{\text{void}}$ is a one-row matrix with $c_{1R} = 1$ and $c_{1I} = 0$ for $I \neq R$.

The only open question is how to compute the vector $\mathbf{f}_{\text{void}}$ of nodal fluxes. The way to proceed depends on whether one chooses the Lagrange-multiplier technique or the ad-hoc transformation method for the porous domain, see section 4. In both cases, however, it is useful to rewrite equation (16) as

$$\mathbf{f}_{\text{void}}\big|_{\Gamma_{\text{I}}} = -\mathbf{f}_{\text{por}}\big|_{\Gamma_{\text{I}}} \tag{20}$$

to emphasize the continuity of flux at the interface.

4. COMPUTATIONAL ASPECTS

4.1. The Lagrange-multiplier technique

In the Lagrange-multiplier technique [6, 7, 8], the constraints are added to the original system of equations. For the flow in the porous medium, for instance, the unconstrained system (6) of order `numnp` is enlarged by adding the `numcon` constraints of equation (7) and `numcon` unknowns (the Lagrange multipliers λ, one per constraint):

$$\begin{bmatrix} \mathbf{K}_{\text{por}} & \mathbf{C}_{\text{por}}^T \\ \mathbf{C}_{\text{por}} & \mathbf{0} \end{bmatrix} \begin{Bmatrix} \mathbf{h} \\ \boldsymbol{\lambda} \end{Bmatrix} = \begin{Bmatrix} \mathbf{0} \\ \mathbf{b}_{\text{por}} \end{Bmatrix} \tag{21}$$

Note that the first block-equation in (21) reads $\mathbf{K}_{\text{por}}\mathbf{h} = -\mathbf{C}_{\text{por}}^T\boldsymbol{\lambda}$. In fact, the Lagrange multipliers represent the "reactions" associated to the linear constraints. Thus, the nodal fluxes in the porous domain are simply

$$\mathbf{f}_{\text{por}} = -\mathbf{C}_{\text{por}}^T\boldsymbol{\lambda} \tag{22}$$

The flow in the inclusions is treated in a similar way. It follows directly from equations (20) and (22) that the nodal fluxes at the interface Γ_{I} for the void problem are simply

$$\mathbf{f}_{\mathrm{void}}|_{\Gamma_{\mathrm{I}}} = \mathbf{C}_{\mathrm{por}}^{T}\boldsymbol{\lambda}|_{\Gamma_{\mathrm{I}}} \tag{23}$$

The rest of the components of $\mathbf{f}_{\mathrm{void}}$ (i.e. for nodes not in Γ_{I}) are zero.

The Lagrange-multiplier technique has two widely known drawbacks: (1) the dimension of the original problem is increased and (2) the enlarged matrix is not positive definite. It has, on the other hand, two clear advantages: (1) general linear constraints can be handled in a straightforward manner and (2) it is a technique naturally adapted to object-oriented programming [8, 9].

The Lagrange-multiplier approach is summarized in figure 3.

4.2. An ad-hoc transformation method

The basic idea of transformation methods is to employ the constraints associated to the boundary conditions to transform the singular, unconstrained matrix into a regular matrix. The order of the problem is either maintained [10, 11, 12] or reduced [13, 14].

This approach has two clear advantages: (1) the dimension of the original problem is not increased, (2) the regular matrix is symmetric positive definite.

The main disadvantage of most transformation methods is that they are rather cumbersome to implement in presence of general, multi-point constraints. In a recent contribution [15], Ainsworth presents a transformation method of the first type (order is maintained) which handles, in a systematic way, general linear constraints.

It must be noted, however, that our linear constraints are rather simple. This has motivated an ad-hoc, computationally efficient transformation method, illustrated here for the case of

Flow in the porous medium

- Assemble permeability matrix $\mathbf{K}_{\text{por}}$ from elementary matrices $\mathbf{K}_{\text{por}}^{\text{e}}$

- Build constraint matrix $\mathbf{C}_{\text{por}}$ and constraint vector $\mathbf{b}_{\text{por}}$ associated to:

    - Dirichlet boundary conditions [equation (8)]

    - Inclusions with no head drop [equation (10)]

    - Inclusions with prescribed head drop [equation (12)]

- Solve enlarged linear system (21):

$$
\begin{bmatrix} \mathbf{K}_{\text{por}} & \mathbf{C}_{\text{por}}^{T} \\ \mathbf{C}_{\text{por}} & \mathbf{0} \end{bmatrix} \begin{Bmatrix} \mathbf{h} \\ \boldsymbol{\lambda} \end{Bmatrix} = \begin{Bmatrix} \mathbf{0} \\ \mathbf{b}_{\text{por}} \end{Bmatrix}
$$

- Compute Darcy velocity $\mathbf{v}_{\text{por}}$ [equation (1)]

Flow in the inclusions

- Assemble Laplacian matrix $\mathbf{K}_{\text{void}}$ from elementary matrices $\mathbf{K}_{\text{void}}^{\text{e}}$

- Build constraint matrix $\mathbf{C}_{\text{void}}$ associated to one node per inclusion prescribed to zero [equation (18)]

- Build flux vector $\mathbf{f}_{\text{void}}$ [equation (23)]

- Solve enlarged linear system:

$$
\begin{bmatrix} \mathbf{K}_{\text{void}} & \mathbf{C}_{\text{void}}^{T} \\ \mathbf{C}_{\text{void}} & \mathbf{0} \end{bmatrix} \begin{Bmatrix} \boldsymbol{\phi} \\ \boldsymbol{\lambda} \end{Bmatrix} = \begin{Bmatrix} \mathbf{f}_{\text{void}} \\ \mathbf{0} \end{Bmatrix}
$$

- Compute velocity $\mathbf{v}_{\text{void}}$ [equation (13)]

Figure 3. The Lagrange-multiplier technique

Flow in the porous medium

- Assemble reduced matrix $\mathbf{K}'_{\text{por}}$ from elementary matrices $\mathbf{K}^{\text{e}}_{\text{por}}$

- Assemble reduced flux vector $\mathbf{f}'_{\text{por}}$ associated to:

    − Non-homogeneous Dirichlet boundary conditions

    − Inclusions with prescribed head drop

- Solve reduced linear system $\mathbf{K}'_{\text{por}}\mathbf{h}' = \mathbf{f}'_{\text{por}}$

- Build full vector of nodal head values $\mathbf{h}$

- Compute Darcy velocity $\mathbf{v}_{\text{por}}$

Flow in the inclusions

- Compute full vector of nodal fluxes $\mathbf{f}_{\text{por}}$

- Assemble reduced matrix $\mathbf{K}'_{\text{void}}$ from elementary matrices $\mathbf{K}^{\text{e}}_{\text{void}}$

- Build reduced vector of nodal fluxes $\mathbf{f}'_{\text{void}}$ from $\mathbf{f}_{\text{por}}$

- Solve reduced linear system $\mathbf{K}'_{\text{void}}\boldsymbol{\phi}' = \mathbf{f}'_{\text{void}}$

- Build full vector of nodal potential values $\boldsymbol{\phi}$

- Compute velocity $\mathbf{v}_{\text{void}}$

Figure 4. The ad-hoc transformation method

two inclusions (with and without head drop). The basic steps are summarized in figure 4. A more detailed algorithm in pseudo-code can be found in appendix II.

Dirichlet boundary conditions   The one-point Dirichlet boundary conditions of equations (8) and (18) are treated via the usual row-and-column adjustment. In the porous domain (non-homogeneous conditions), the known term $-k_{IJ}h_J^{\text{pres}}$ is added to the RHS of equation (6) for all $J \in \mathcal{D}$ (i.e. for all the nodes in the Dirichlet boundary Γ_{D}), and row and column J are removed from the system. In the inclusions (homogeneous conditions), we simply remove rows and columns R and $\widehat{R}$.

Uniform head at the interface (porous domain)   Only one unknown h_R is required to describe the head at the interface of an inclusion, see equation (10). This means that, for all $J \in \mathcal{I}$, the unknowns h_J can be condensated into a single unknown h_R simply by adding up all rows J into row R and all columns J into column R.

This also holds for inclusions with a prescribed head drop, see equation (12): the head in the interface $\mathcal{I}_{\text{a}} \cup \mathcal{I}_{\text{b}}$ is represented by the unknown $h_{\widehat{R}}$. The known terms $-k_{IJ}\Delta h$ associated to the head drop are assembled in the RHS for all the nodes of $\Gamma_{\text{I}_{\text{a}}}$.

Reduced system (porous domain)   The resulting regular system is

$$\mathbf{K}'_{\text{por}}\mathbf{h}' = \mathbf{f}'_{\text{por}} \tag{24}$$

where $\mathbf{h}'$ is the reduced vector of nodal head values (with known Dirichlet values excluded, only one value per interface), $\mathbf{K}'_{\text{por}}$ is the reduced permeability matrix and $\mathbf{f}'_{\text{por}}$ is the reduced vector of nodal fluxes which contains the terms $-k_{IJ}h_J^{\text{pres}}$ and $-k_{IJ}\Delta h$.

From a practical viewpoint, it is very important to remark that there is no need to assemble the singular system (6) and then start suppressing and condensating rows and columns as indicated; the reduced system (24) can be directly assembled with an adequate assignment of equation numbers to nodes [16] (no equation assigned to Dirichlet nodes; same equation

assigned to all nodes in the interface of each inclusion).

Nodal fluxes in the interface   With the transformation method, the fluxes in the interface nodes are not obtained together with the vector of piezometric heads $\mathbf{h}$. However, they can be computed a posteriori by a simple matrix-vector product

$$\mathbf{f}_{\mathrm{por}} = \mathbf{K}_{\mathrm{por}}\mathbf{h} \tag{25}$$

and $\mathbf{f}_{\mathrm{void}}$ can be constructed from here as commented in section 4.1.

Remark 2. *Note that, in reality, only the components of $\mathbf{f}_{\mathrm{por}}$ in the interface are really needed. The vector $\mathbf{f}_{\mathrm{por}}|_{\Gamma_{\mathrm{I}}}$ can be assembled, without the global matrix-vector product of equation (25), by using only the relevant elementary matrices (i.e. associated to elements with nodes in Γ_{I}).*

Reduced system (inclusions)   The resulting regular system is

$$\mathbf{K}'_{\mathrm{void}}\phi' = \mathbf{f}'_{\mathrm{void}} \tag{26}$$

where rows and columns R and $\widehat{R}$ have been removed.

5. NUMERICAL EXAMPLES

The proposed approach for the treatment of void inclusions in a porous medium is illustrated here by means of three numerical examples. The ad-hoc transformation method of section 4.2 is used in all the computations to handle the boundary conditions. The resulting linear systems are solved with the diagonally preconditioned conjugate gradient (DPCG) method [17] with a convergence tolerance for the relative error of 0.5×10^{-7} in "displacements" and 0.5×10^{-6} in "forces". For simplicity, dimensionless variables are used, although this is by no means necessary.

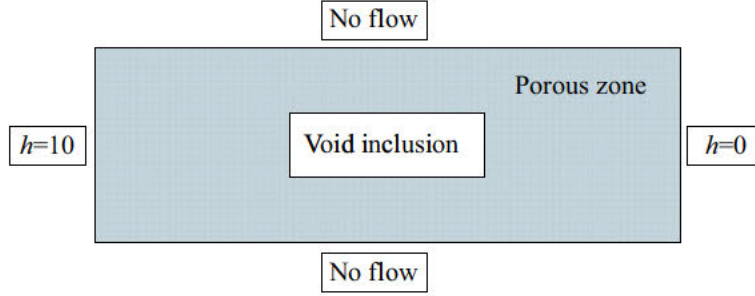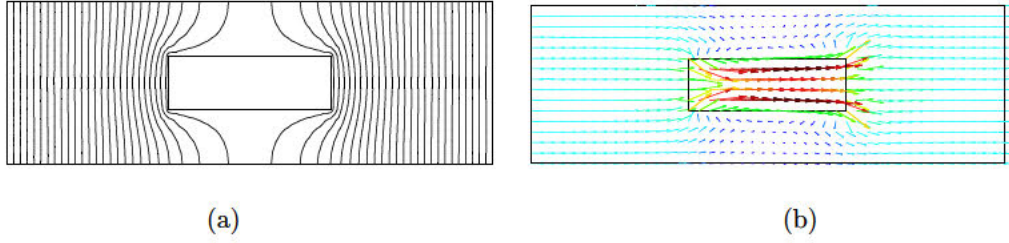Figure 5. 2D example with interior inclusion. Problem statement

(a)   (b)

Figure 6. 2D example with interior inclusion. Fields of: (a) isolines of piezometric head; (b) velocity

5.1. *Two-dimensional domain with one interior void inclusion*

As a first example, consider the two-dimensional domain of figure 5. The rectangular porous domain has an interior rectangular void inclusion. Piezometric head is prescribed in the two edges of the rectangular domain. Zero flux is prescribed in the two lateral walls of the domain. A structured mesh of 15×45 square finite elements is used in the analysis. A reference permeability $k_{\mathrm{por}} = 1$ is taken for the porous domain.

The proposed approach based on potential flow is used to treat the void inclusion. The piezometric head is assumed to be uniform in the interface with the porous medium. Since there is a single interface, no head drop can be prescribed.

The results are depicted in figure 6. The effect of the void inclusion is clear, both in the
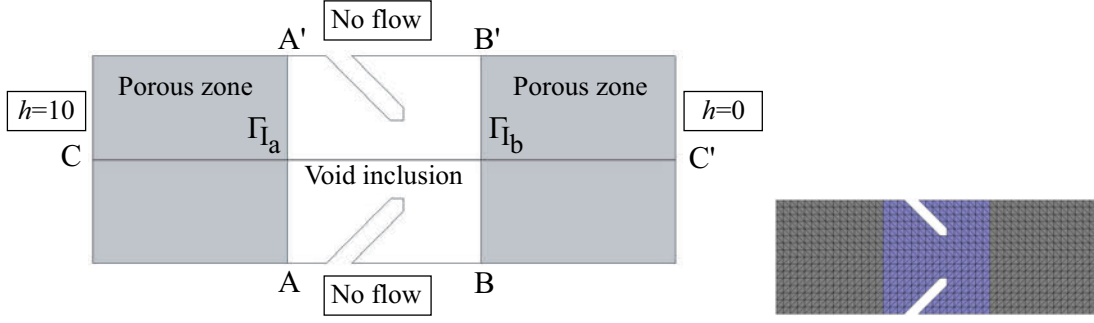
Figure 7. 2D example with one exterior void inclusion. The void inclusion with two fins separates the two porous zones. The finite element mesh has 701 elements and 409 nodes.

piezometric head and velocity fields. Since the piezometric head is uniform in the interface, the isolines "wrap" the void inclusion, see figure 6(a). Note also that the flow across the inclusion is larger than across the upper and bottom porous zones, see figure 6(b).

5.2. *Two-dimensional domain with one exterior void inclusion*

In the second example, two blocks of porous media are separated by a void inclusion with two fins, see figure 7. As in the previous example, Dirichlet boundary conditions and homogeneous Neumann boundary conditions are prescribed respectively in the two edges and the two lateral walls of the domain. Figure 7 also shows the structured mesh of 701 triangular elements and 409 nodes (due to symmetry, the computational domain is half of the physical domain).

Again, a reference permeability $k_{\mathrm{por}} = 1$ is taken for the porous domain. In the first analysis, no head drop is prescribed in the void inclusion. This inclusion is treated with the two approaches discussed in this paper: (i) as a fictitious porous medium with permeability $k_{\mathrm{void}} = 10^n k_{\mathrm{por}}$, with increasing values of n (classical approach); (ii) with potential flow (approach proposed here).

The effect of power n on the simulation results is depicted in figure 8. Figure 8(a) shows the

velocity field for $n = 0$, $n = 5$ and the proposed approach. Taking $n = 0$ (i.e. $k_{\text{void}} = k_{\text{por}}$), the overall permeability of the domain, and hence the velocity, are underestimated. With $n = 5$ and the proposed approach, on the contrary, the same results are obtained. Figure 8(b) shows the head profile along interfaces Γ_{I_a} and Γ_{I_b} for $n = 0, \ldots, 5$ and the proposed approach. Again, increasing n leads to the same solution than using potential flow for the inclusion. Note also that the head is uniform in each interface for large values of n. This result corroborates the validity of hypothesis (5), which is justified in Appendix I on physical terms. Since no head drop is prescribed in the inclusion, the uniform head is the same for the two interfaces, $h^{\text{inter}_a} = h^{\text{inter}_b}$.

Table I summarizes the convergence results. Increasing the fictitious permeability k_{void} clearly affects the conditioning of the permeability matrix: indeed, a tenfold increase in k_{void} results in a tenfold increase in the condition number, see table I(a). This ill-conditioning has a moderate impact in the required number of iterations for this simple 2D example. This is not the case in larger, 3D problems, as illustrated in section 5.3.

With the proposed approach there is no ill-conditioning, because the void inclusion is not modelled with a large fictitious permeability, see table I(b). Note also that the aggregate number of iterations is similar to that of the fictitious-permeability approach. However, since two smaller linear systems (one for the porous domain and one for the void inclusion) instead of a single, larger linear system (for the whole domain) are solved, a similar number of iterations translates into a significantly smaller computational cost.

The effect of n on convergence can also be seen in figure 9, where the relative residual error $||\mathbf{r}^k||_2 / ||\mathbf{b}||_2$ (where $\mathbf{r}$ is the residual vector, $\mathbf{b}$ is the RHS vector and $|| \cdot ||_2$ denotes the Euclidean or 2-norm) and the approximate relative error in the solution, $||\mathbf{x}^k - \mathbf{x}^{k+1}||_2 / ||\mathbf{x}^{k+1}||_2$
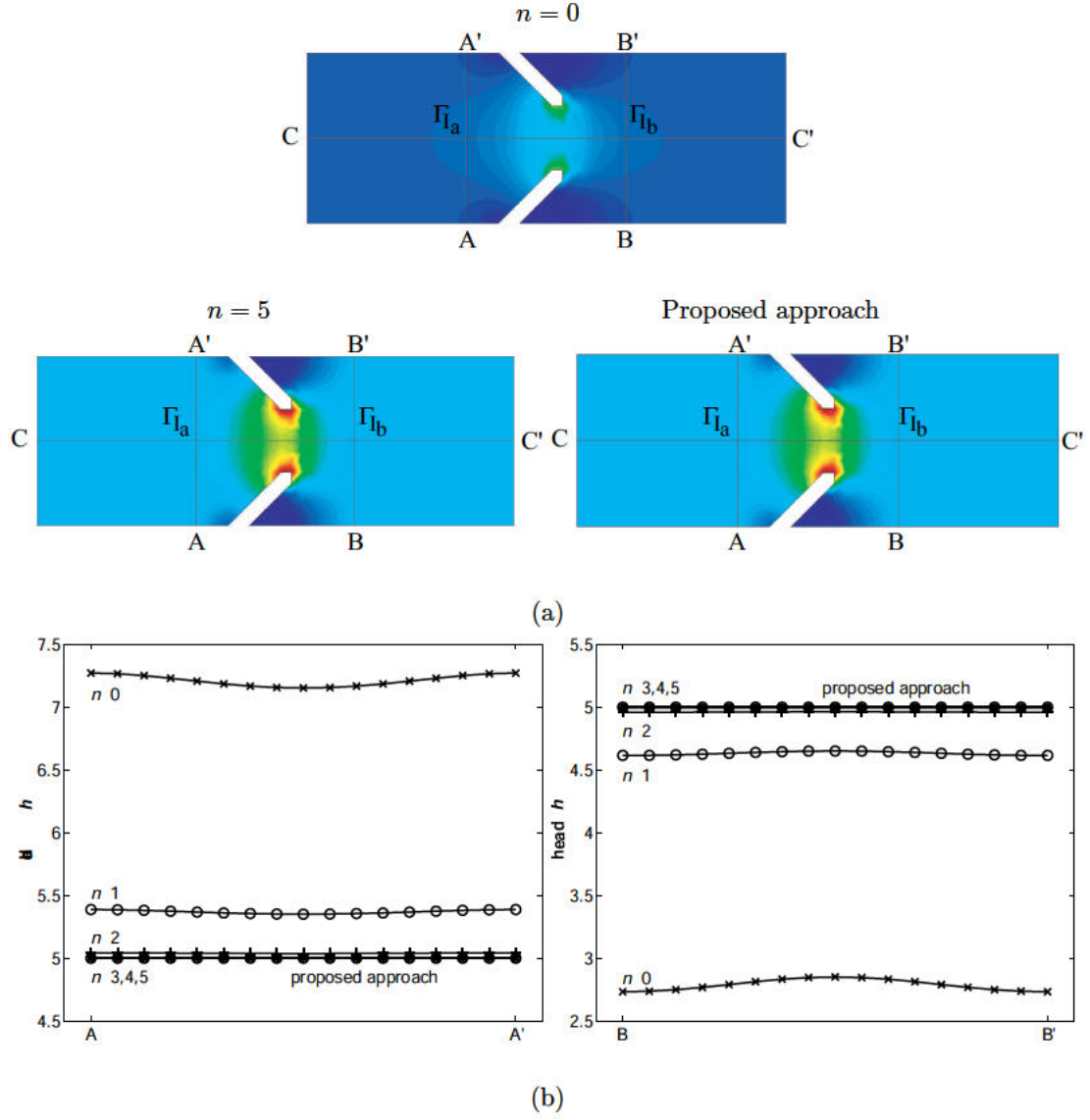
(a)

(b)

Figure 8. 2D example with exterior void inclusion: (a) velocity field for two different values of the fictitious permeability and for the proposed approach; (b) head profile along the interfaces Γ_{I_a} and Γ_{I_b}.

Table I. 2D example with exterior void inclusion. Condition number of linear systems and number of iterations: (a) if the inclusion is treated as a porous medium, large values of permeability lead to ill-conditioning; (b) with the proposed approach, there is no ill-conditioning.

Permeability of inclusion: $k_{\text{void}} = 10^n k_{\text{por}}$					
n	cond($\mathbf{K}'$)	iter	n	cond($\mathbf{K}'$)	iter
0	2.38×10^3	96	3	1.99×10^6	106
1	2.03×10^4	100	4	1.99×10^7	110
2	1.99×10^5	103	5	1.99×10^8	112

(a)

Proposed approach		
Material	cond($\mathbf{K}'$)	iter
Porous	1.18×10^3	37
Inclusion	3.87×10^3	65

(b)
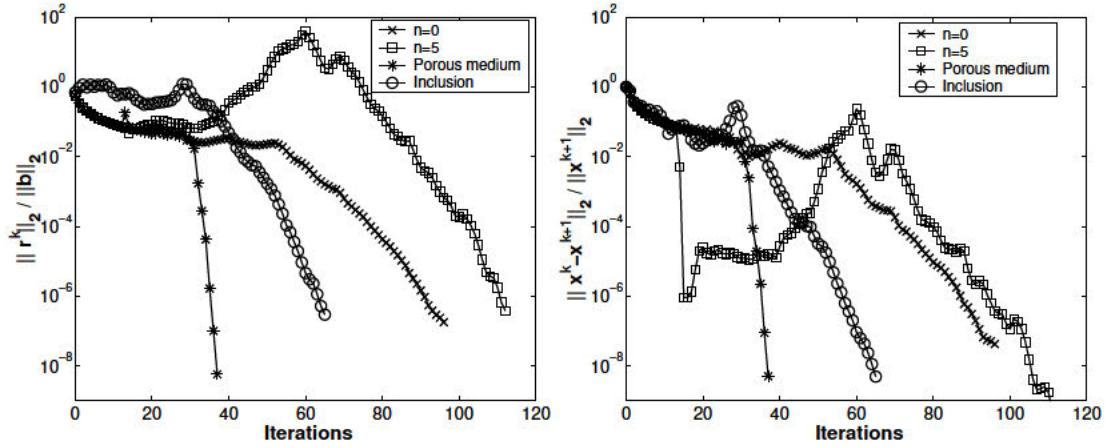
Figure 9. 2D example with exterior void inclusion. Convergence of the DPCG solver for two different values of the inclusion permeability and for the proposed approach: (a) relative residual; (b) approximate relative error in the solution

are plotted versus the iteration counter k. Note that, for $n = 5$, and due to ill-conditioning, both error measures consistently increase up to iteration 60, and then start decreasing.

Remark 3. *The increase in the relative error in the solution in some of the curves of figure*
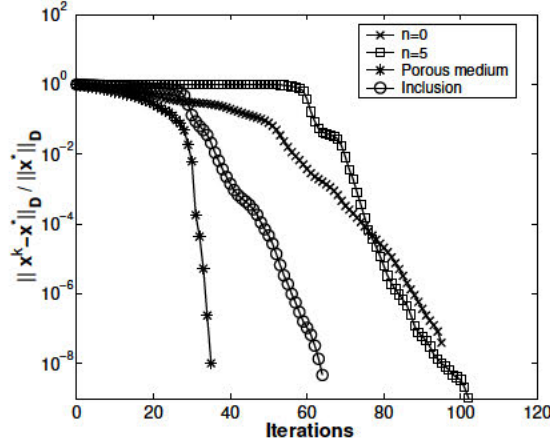
Figure 10. 2D example with exterior void inclusion. Monotonic convergence is obtained for the energy norm associated to preconditioner $\mathbf{D}$ of the relative error with respect to the solution $\star$.

9(b) is not in contradiction with the convergence properties of the conjugate gradient method. For the non-preconditioned CG method, monotonic convergence (i.e. error reduction at each iteration) in 2-norm is guaranteed [18]. Note, however, that (1) this theoretical result involves the relative error of $\mathbf{x}^k$ with respect to the exact solution $\mathbf{x}^\star$, not the approximate relative error with respect to $\mathbf{x}^{k+1}$, and (2), for the DPCG method, monotonic convergence holds if the energy norm associated to the diagonal preconditioner $\mathbf{D}$ is used. Indeed, $\|\mathbf{x}^k-\mathbf{x}^\star\|_{\mathbf{D}}/\|\mathbf{x}^\star\|_{\mathbf{D}} := \sqrt{(\mathbf{x}^k - \mathbf{x}^\star)^T \mathbf{D}(\mathbf{x}^k - \mathbf{x}^\star)}/\sqrt{(\mathbf{x}^\star)^T \mathbf{D}\mathbf{x}^\star}$ decreases at each iteration (but very slowly in the ill-conditioned case $n = 5$), see figure 10.

In a more detailed analysis, it could be necessary to account for the head loss associated to the two fins in the void inclusion. As discussed in section 2.2, with the proposed approach it is very simple to prescribe the head drop between the two interfaces.

For instance, a head drop of $\Delta h = 1$ can be added to the boundary conditions of figure 7. If the flow is reversed (i.e. $h = 0$ in the left edge, $h = 10$ in the right edge), on the other hand,

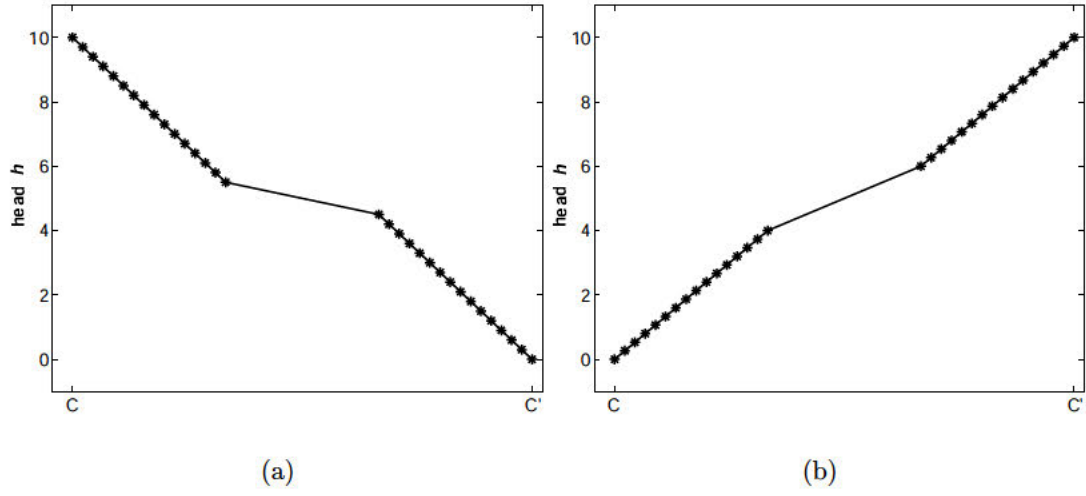(a)   (b)

Figure 11. 2D example with exterior void inclusion. Head profile along the axis of symmetry CC' for two different values of the head drop in the inclusion: (a) $\Delta h = 1$, flow to the right; (b) $\Delta h = 2$, flow to the left.

one would like to prescribe a larger head drop (of, say, $\Delta h = 2$), because now the flow in the inclusion is "against" the fins.

Figure 11 shows, for these two cases, the head profile along the axis of symmetry of the domain. The prescribed head drop is clearly visible in these plots. As expected, the convergence results are the same previously obtained for $\Delta h = 0$ (that is, table I(b) and figure 9), because Δh is assembled in the RHS vector and does not affect the condition number of the matrix.

5.3. Three-dimensional example with multiple inclusions

The three-dimensional domain of figure 12 consists of nine porous blocks connected by eight void inclusions (four in the top layer and four in the bottom layer). Boundary conditions and the structured finite element mesh of 6576 elements and 9261 nodes are also shown.

Like in section 5.2, the problem is solved with the fictitious-permeability approach (with $n = 0, \ldots, 5$) and with the proposed approach. Table II summarizes the convergence results.
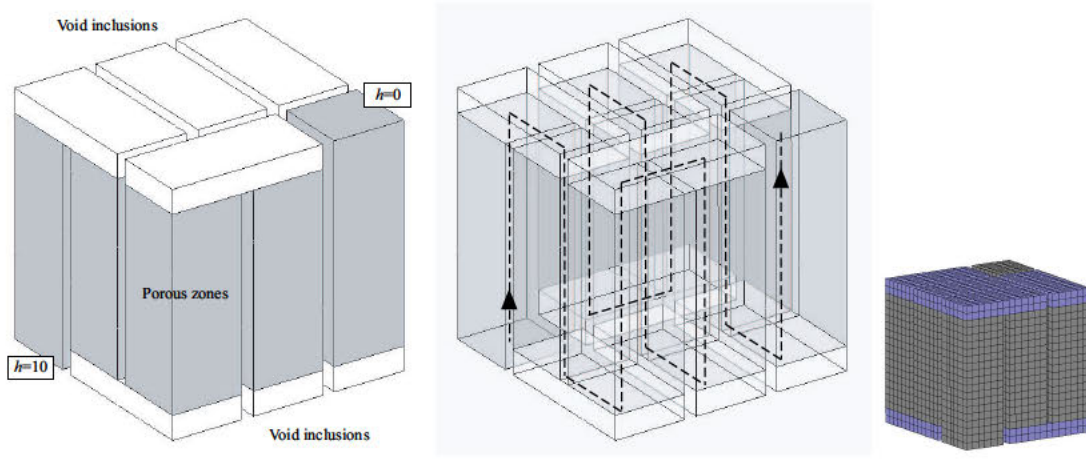
Figure 12. 3D example. The cubic domain is divided into nine porous zones and eight intermediate void inclusions. The finite element mesh has 6576 elements and 9261 nodes

The evolution of the condition number with n is similar to that of the two-dimensional case, see table I. In this case, however, ill-conditioning significantly affects the required number of iterations and thus the CPU time.

This can also be seen in figure 13, where the convergence history of three analyses ($n = 0$, $n = 5$ and proposed approach) is plotted. Note the poor convergence behaviour for $n = 5$, with eight peaks in the curve of relative error vs. iterations (one per void inclusion).

The results obtained with the proposed approach are depicted in figure 14. Figure 14(a) shows the velocity field and figure 14(b) the head field in the porous blocks. Note the uniform head at the interfaces and also the uniform velocity in the nine porous blocks (of equal cross-section).
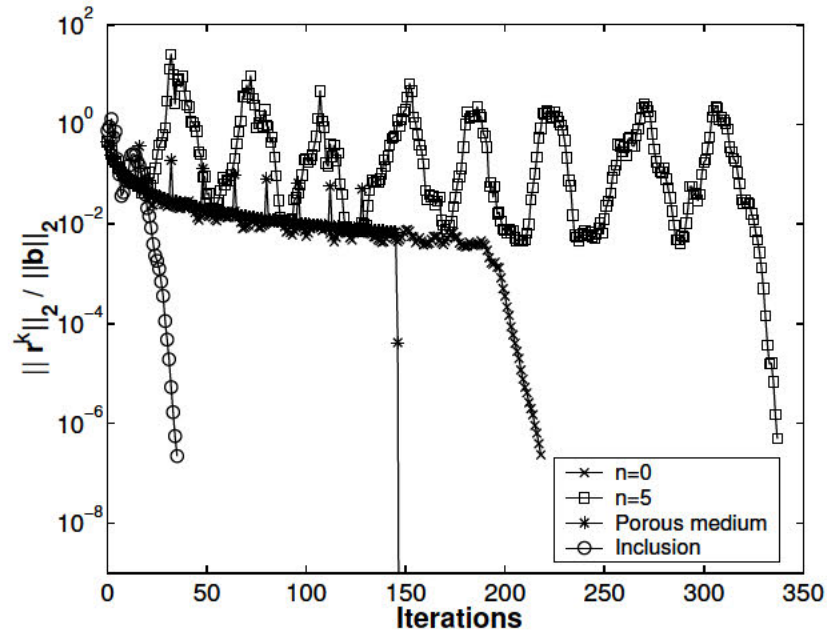
Figure 13. 3D example. Convergence of the DPCG solver for two different values of the inclusion permeability and for the proposed approach.

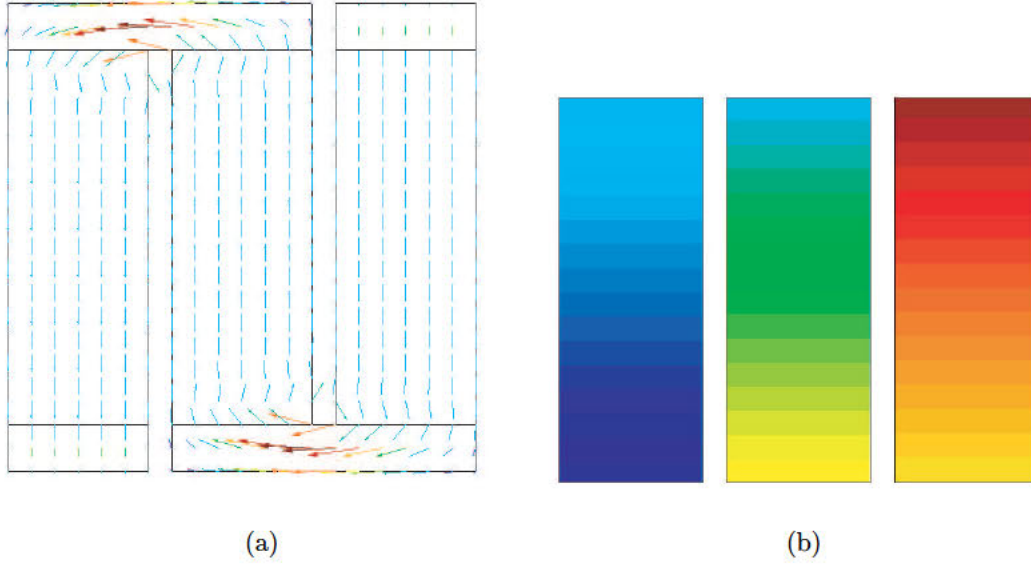(a)   (b)

Figure 14. 3D example. (a) velocity field and (b) head field in porous blocks obtained with the proposed approach. Vertical section across the mid-plane.

Table II. 3D example. Condition number of linear systems and number of iterations for (a) different values of the inclusion permeability and (b) the proposed approach. Ill-conditioning due to large permeability has a significant impact in convergence behaviour

Permeability of inclusion: $k_{\text{void}} = 10^n k_{\text{por}}$

n	cond($\mathbf{K}'$)	iter	CPU time (s)	n	cond($\mathbf{K}'$)	iter	CPU time (s)
0	4.00×10^4	218	6.5	3	2.60×10^7	286	7.1
1	2.78×10^5	241	6.6	4	2.60×10^8	319	8
2	2.62×10^6	266	7.4	5	2.60×10^9	337	8.5

(a)

Proposed approach

Material	cond($\mathbf{K}'$)	iter	CPU time (s)
Porous	2.43×10^3	148	—
Inclusion	5.91×10^3	35	—
Total	—	183	4.6

(b)

6. CONCLUDING REMARKS

The use of potential flow leads to an efficient treatment of void inclusions in a porous medium. There is no need to use a large fictitious permeability to model the inclusion, so ill-conditioning in the linear system is precluded. This clearly improves the convergence of iterative solvers, especially in large 3D analyses.

The proposed approach can be regarded as a domain decomposition technique, in the sense that two sub-problems (one for the porous domain and one for the inclusions) are solved.

Continuity of flux is invoked to link the two sub-problems. The resulting boundary conditions can be handled via either the Lagrange-multiplier technique or an ad-hoc transformation method. With this transformation method, discussed in full algorithmic detail, it is very straightforward to implement the proposed approach in a standard finite element code for porous flow analysis.

Besides computational efficiency, another advantage of the decomposition approach suggested here is that the head drop in the inclusions can be prescribed in a very simple way, as an additional input. There is no need to translate the desired head drop into an equivalent fictitious permeability for the inclusion.
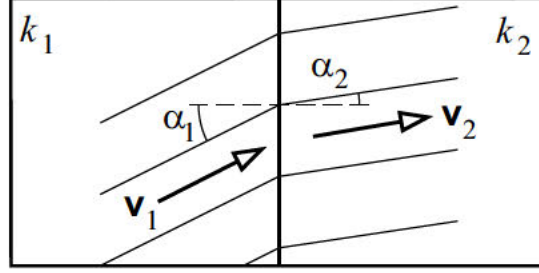
Figure 15. Interface between two porous materials of different permeability

APPENDIX I: PHYSICAL MEANING OF UNIFORM HEAD AT THE INTERFACE

Figure 15 shows the interface between two porous materials of permeabilities k_1 and k_2. It is a well-known result in porous media theory that the change in slope of the streamlines is given by [2]

$$\frac{\tan \alpha_1}{\tan \alpha_2} = \frac{k_1}{k_2} \qquad (27)$$

where α_1 and α_2 are the angles of the streamlines with the normal to the interface, as indicated in figure 15. On the other hand, continuity of flux, equation (15), can be recast as

$$\frac{\cos \alpha_1}{\cos \alpha_2} = \frac{||\mathbf{v_1}||}{||\mathbf{v_2}||} \qquad (28)$$

Assume now that permeability k_1 is finite but k_2 is (i.e. tends to) infinity. According to equation (27), this implies that $\tan \alpha_1 / \tan \alpha_2$ is zero, so either *i)* $\alpha_1 = 0$ or *ii)* $\alpha_2 = \pi/2$. This second option can be discarded because, according to equation (28), it would imply a physically unrealistic infinite ratio of velocities. This means that $\alpha_1 = 0$: the streamlines are orthogonal to the interface in the porous medium with finite permeability k_1. Since the equipotential lines (i.e. lines with equal piezometric head) and the streamlines are orthogonal, we conclude that the interface is indeed an equipotential line.

APPENDIX II: PSEUDO-CODE FOR THE AD-HOC TRANSFORMATION METHOD

Figure 16 contains a detailed pseudo-code version of the proposed ad-hoc transformation method. Following the notation introduced by Hughes [6], we use letters A and B to denote global node numbers, a and b for local node numbers, P and Q for equation numbers and e for element numbers. The relation between these four numberings is given by arrays IEN, ID and LM:

IEN: elements node array; $\mathrm{IEN}(a,e) = A$

ID: destination array; $\mathrm{ID}(A) = P$

LM: location matrix; $\mathrm{LM}(a,e) = P$

Since two linear systems are solved (one for the porous domain and one for the inclusions), two different destination arrays ($\mathrm{ID}_{\mathrm{por}}$ and $\mathrm{ID}_{\mathrm{void}}$) and location matrices ($\mathrm{LM}_{\mathrm{por}}$ and $\mathrm{LM}_{\mathrm{void}}$) are needed.

The key issue in this algorithm is the proper assignment of equation numbers to nodes, steps 1 and 7. The rest is very standard in finite element codes. As a consequence, implementing this algorithm into an existing finite element code is a straightforward task.

```
% FLOW IN THE POROUS MEDIUM
% 1.  Assign equation numbers P to nodes I
      P = 0
      forall nodes I in Ω_por
          I ∈ D                    ⟹ ID_por(I) = 0
          I = R                    ⟹ P = P + 1; ID_por(I) = P
          I ∈ I − {R}              ⟹ ID_por(I) = ID_por(R)
          I = R̂                    ⟹ P = P + 1; ID_por(I) = P
          I ∈ I_a − {R̂} ∪ I_b      ⟹ ID_por(I) = ID_por(R̂)
          otherwise                ⟹ P = P + 1; ID_por(I) = P
      end % forall I
```

%
% **2. Compute reduced matrix $\mathbf{K}'_{\mathrm{por}}$ and reduced vector $\mathbf{f}'_{\mathrm{por}}$**
    forall elements e in Ω_{por}
        forall nodes a in element
            $A = \mathrm{LM}_{\mathrm{por}}(a, e)$
            $A = 0 \Longrightarrow$ next a
            forall nodes b in element
                $B = \mathrm{LM}_{\mathrm{por}}(b, e)$
                $I = \mathrm{IEN}(b, e)$
                Compute $\mathbf{K}^e_{\mathrm{por}}(a, b)$
                $I \notin \mathcal{D} \Longrightarrow \mathbf{K}'_{\mathrm{por}}(A, B) = \mathbf{K}'_{\mathrm{por}}(A, B) + \mathbf{K}^e_{\mathrm{por}}(a, b)$
                $I \in \mathcal{D} \cup \mathcal{I}_{\mathrm{b}} \Longrightarrow \mathbf{f}'_{\mathrm{por}}(A) = \mathbf{f}'_{\mathrm{por}}(A) - \mathbf{K}^e_{\mathrm{por}}(a, b)\mathbf{b}_{\mathrm{por}}(I)$
            end % forall b
        end % forall a
    end % forall e
%
% **3. Solve the linear system $\mathbf{K}'_{\mathrm{por}}\mathbf{h}' = \mathbf{f}'_{\mathrm{por}}$**
%
% **4. Build vector of nodal head values $\mathbf{h}$ from $\mathbf{h}'$ and $\mathbf{b}_{\mathrm{por}}$**
    forall nodes I in Ω_{por}
        $P = \mathrm{ID}_{\mathrm{por}}(I)$
        $I \in \mathcal{D} \qquad \Longrightarrow \mathbf{h}(I) = \mathbf{b}_{\mathrm{por}}(I)$
        $I \in \mathcal{I}_{\mathrm{b}} \qquad \Longrightarrow \mathbf{h}(I) = \mathbf{h}'(P) + \mathbf{b}_{\mathrm{por}}(I)$
        otherwise $\Longrightarrow \mathbf{h}(I) = \mathbf{h}'(P)$
    end % forall I
%
% **5. Compute the Darcy velocity $\mathbf{v}_{\mathrm{por}}$**
%
% **FLOW IN THE INCLUSIONS**
% **6. Compute vector of nodal fluxes $\mathbf{f}_{\mathrm{por}}$**
    forall elements e in Ω_{por}
        forall nodes a in element
            $I = \mathrm{IEN}(a, e)$
            $I \notin \mathcal{I} \cup \mathcal{I}_{\mathrm{a}} \cup \mathcal{I}_{\mathrm{b}} \Longrightarrow$ next a
            forall nodes b in element
                $J = \mathrm{IEN}(b, e)$
                Compute $\mathbf{K}^e_{\mathrm{por}}(a, b)$
                $\mathbf{f}_{\mathrm{por}}(I) = \mathbf{f}_{\mathrm{por}}(I) + \mathbf{K}^e_{\mathrm{por}}(a, b)\mathbf{h}(J)$
            end % forall b
        end % forall a
    end % forall e

```
%
% 7.  Assign equation numbers P to nodes I
        P = 0
        forall nodes I in Ω_void
            I ∈ {R, R̂} ⟹ ID_void(I) = 0
            otherwise   ⟹ P = P + 1; ID_void(I) = P
        end % forall I
%
% 8.  Compute reduced matrix K'_void and reduced vector f'_void
        forall elements e in Ω_void
            forall nodes a in element
                A = LM_void(a, e)
                A = 0 ⟹ next a
                forall nodes b in element
                    B = LM_void(b, e)
                    I = IEN(b, e)
                    Compute K^e_void(a, b)
                    I ∉ {R, R̂} ⟹ K'_void(A, B) = K'_void(A, B) + K^e_void(a, b)
                    I ∈ (ℐ − {R}) ∪ (ℐ_a − {R̂}) ∪ ℐ_b ⟹ f'_void(B) = −f_por(I)
                end % forall b
            end % forall a
        end % forall e
%
% 9.  Solve the linear system K'_void φ' = f'_void
%
% 10. Build vector of nodal potential values φ from φ'
        forall nodes I in Ω_void
            P = ID_void(I)
            I ∈ {R, R̂} ⟹ φ(I) = 0
            otherwise   ⟹ φ(I) = φ'(P)
        end % forall I
%
% 11. Compute the velocity v_void
```

Figure 16. Pseudo-code of the ad-hoc transformation method

ACKNOWLEDGEMENTS

REFERENCES

1. T.W. Lambe and R.W. Whitman. *Soil Mechanics*. Wiley, 1969.

2. R. Lancellotta. *Geotechnical Engineering*. Balkema, 1995.

3. C. Knudby, J. Carrera, and G.E. Fogg. An empirical method for the evaluation of the equivalent conductivity of low-permeability matrices with high-permeability inclusions. In *Proceedings of the International Groundwater Symposium 2002, Berkeley, California, USA*, 2002.

4. B.B. Dykaar and P.K. Kitanidis. Determination of the effective hydraulic conductivity for heterogeneous porous media using a numerical spectral approach. 1. Method. *Water Resources Research*, 28(4):1155–1166, 1992.

5. M. Discacciati, E. Miglio, and A. Quarteroni. Mathematical and numerical models for coupling surface and groundwater flows. *Applied Numerical Mathematics*, 43(1–2):57–74, 2002.

6. T.J.R. Hughes. *The Finite Element Method – Linear Static and Dynamic Finite Element Analysis*. Prentice-Hall, 1987.

7. O.C. Zienkiewicz and R.L. Taylor. *The Finite Element Method: Basic Formulation and Linear Problems*. McGraw-Hill, 1989.

8. A. Rodríguez-Ferran and A. Huerta. Adapting Broyden method to handle linear constraints imposed via Lagrange multipliers. *International Journal for Numerical Methods in Engineering*, 46(12):2011–2026, 1999.

9. Y. Dubois-Pèlerin and P. Pegon. Linear constraints in object-oriented finite element programming. *Computer Methods in Applied Mechanics and Engineering*, 154(1–2):31–39, 1998.

10. J.F. Abel and M.S. Shephard. An algorithm for multipoint constraints in finite element analysis. *International Journal for Numerical Methods in Engineering*, 14(3):464–467, 1979.

11. H.L. Schreyer and D.A. Parsons. Direct application of constraints to symmetric algebraic systems. *Communications in Numerical Methods in Engineering*, 11(7):563–573, 1995.

12. J.P. Webb. Imposing linear constraints in finite-element analysis. *Communications in Numerical Methods in Engineering*, 6(6):471–475, 1990.

13. J. Barlow. Constraint relationships in linear and nonlinear finite element analyses. *International Journal for Numerical Methods in Engineering*, 18(4):521–533, 1982.

14. M.S. Shephard. Linear multipoint constraints applied via transformation as part of a direct stiffness assembly process. *International Journal for Numerical Methods in Engineering*, 20(11):2107–2112, 1984.

15. M. Ainsworth. Essential boundary conditions and multi-point constraints in finite element analysis. *Computer Methods in Applied Mechanics and Engineering*, 190(48):6323–6339, 2001.

16. J. Sarrate, A. Huerta, and J. Donea. Arbitrary Lagrangian-Eulerian formulation for fluid-rigid body interaction. *Computer Methods in Applied Mechanics and Engineering*, 190(24–25):3171–3188, 2001.

17. C.T. Kelley. *Iterative Methods for Linear and Nonlinear Equations*. SIAM, 1995.

18. G. W. Stewart. *Afternotes Goes to Graduate School*. SIAM, 1998.