A control volume finite element method for adaptive mesh simulation of flow in heap leaching

Peyman Mostaghimi · Brendan S. Tollit ·
Stephen J. Neethling · Gerard J. Gorman ·
Christopher C. Pain

Received: / Accepted:

P. Mostaghimi (✉) · S.J. Neethling
Rio Tinto Centre for Advanced Mineral Recovery, Department of Earth Science and Engineering, Imperial College London
e-mail: peyman@imperial.ac.uk

P. Mostaghimi · B.S. Tollit · G.J. Gorman · C.C. Pain
Applied Modelling and Computational Group, Department of Earth Science and Engineering, Imperial College London
Abstract Accurate determination of fluid flow within heap leaching is crucial for understanding and improving performance of the process. Numerical methods have the potential to assist by modeling the process and studying the transport phenomena within the porous medium. This paper presents an adaptive numerical scheme to solve for unsaturated incompressible flow in porous media with applications to heap leaching. The governing equations are Darcy's law and the conservation of mass. An implicit pressure explicit saturation method (IMPES) is used to decouple the pressure and saturation equations. Pressure is discretized using a control volume finite element method (CVFEM), while for saturation a node centered control volume method is employed. The scheme is equipped with dynamic anisotropic mesh adaptivity to update the mesh resolution as the leaching solution infiltrates through the heap. This allows for high fidelity modelling of multiscale features within the flow. The method is verified against the Buckley-Leverett problem where a quasi-analytical solution is available. The method is applied for two-phase flow of air and leaching solution in a simplified two-dimensional heap geometry. We compare accuracy and CPU efficiency of adaptive mesh against static mesh and demonstrate the potential to achieve high spatial accuracy at low computational cost through the use of anisotropic mesh adaptivity.

Keywords: multiphase flow · porous media · heap leaching · control volume finite element method · adaptive mesh
1 Introduction

Numerical modelling of multiphase flow in porous media is of great importance in many fields of engineering and science [1, Chap. 1]. Often these models are solved using finite difference method [2, Chap. 3]. However, finite difference methods typically lack the flexibility required to represent complex geometries and capture the evolving features associated with the fluid flow. Unstructured finite volume/element methods are capable of capturing complex three dimensional geometries and can be combined with dynamic adaptive mesh techniques to achieve high fidelity models. While the formulation and methods described in this paper can be generalized to different areas of transport in porous media, here we focus on the application of an adaptive control volume finite element scheme to simulate and analyse the transport of multiphase fluids in heap leaching.

Heap leaching is a mineral processing method to extract metals such as copper, zinc, and gold from ores. In this process, crushed ores are heaped on a permeable layer. Then, the constructed heap is irrigated with a leaching solution from the top. The solution percolates through the heap and dissolves the metal. The pregnant solution (i.e. the leaching solution containing the dissolved metal) is then collected from the bottom of the heap. The pregnant solution is subsequently processed and the target minerals are recovered [3, Chap. 2]. Heap leaching was once a niche method for extracting metal from some difficult ores. Because the process can be applied for metal recovery from low grades ores at relatively low operating and capital costs, it is now being applied more widely. The main disadvantage of heap leaching is the comparatively low recoveries achieved compared to other methods such as the froth flotation followed by smelting (30-60% in heap leaching compared to 85-95% in froth flotation [4, Chap. 16, 5, Chap. 2]). Key aspects of this inefficiency are problems in fluid flow and mass transport. Numerical methods have the potential to assist by modelling transport phenomena within a heap. Numerical simulation of heap leaching is a cost effective method to optimize the recovery parameters and design the irrigation strategies.

In this paper, we focus on accurate numerical simulation of flow within a heap, especially the initial wetting of the heap. The governing equations for multiphase flow in porous media consist of Darcy’s law and the conservation of mass. Different numerical approaches have been applied to solve for flow in soils and heap leaching process. Hills et al. [6] developed a water content based algorithm to solve for one-dimensional flow through unsaturated soils. They compared the computational efficiency and mass balance error of the water content based algorithm with finite difference head based methods. The water content algorithm gave less mass balance error and was faster than the head based algorithms for flow in very dry soils. Later, Kirkland et al. [7] extended the water based formulation for simulation of flow in saturated soils. Pan et al. [8]
proposed two mass distributed finite element methods for flow in variably saturated soils. They showed that their methods are free of oscillation and produce less numerical dispersion at the wetting front than the traditional mass lumped scheme. Munoz et al. [9] proposed a two-dimensional mathematical model based on the finite difference discretization of Darcy's law and validated their results by comparing against a radial flow experiment. McBride et al. [10] applied a finite volume method to solve for flow and modelled a gold oxide heap successfully. Cariaga et al. [11] employed a mixed hybrid finite element approach to solve for liquid and air flow in a two-dimensional pilot-scale heap. Bennet et al. [12] presented a comprehensive model for simulation of copper sulphide heap leaching. Their model was implemented in the framework of PHYSICA CFD package [13] that employs a conservative finite volume formulation for simulation of flow in porous media. Further examples of the use of numerical methods for the heap leaching simulation can be found in [14, 15, 16]. McBride et al. [17] gave a review on some common numerical techniques to solve for the flow in a heap leaching process.

One of the main challenges of simulation of heap leaching is capturing the sharp wetting front and predicting the breakthrough time accurately. In addition, the numerical method must be efficient in CPU time due to long-term duration of this process. Previous studies have applied a large collection of numerical methods to flow simulations on static meshes where a fine resolution is required throughout the domain to capture the development of the steep wetting front. This is computationally expensive and sometimes infeasible at the large scales. To overcome these limitations, we employ anisotropic mesh adaptation. This allows mesh resolution to change dynamically in response to local solution error estimates. It involves topological operations on elements including node insertion, node deletion, edge movement, and edge swap [18, 19]. We adapt the mesh based on the curvature of saturation of the liquid phase. Therefore, as the liquid phase infiltrates through the heap, the sharp boundaries between the unsaturated and saturated regions are captured by a fine mesh while a coarse mesh is formed in other regions of the heap.

The developed numerical scheme uses an implicit pressure explicit saturation (IMPES) method to split the governing equations to pressure and saturation equations. The saturation equation is spatially discretized using node centered control volume formed around the unstructured finite element mesh. The face values are determined through an upwind scheme. The pressure equation is spatially discretized using a continuous control volume finite element method (CVFEM) to achieve consistency with the discrete saturation equation. The numerical scheme is implemented within Fluidity. Fluidity is an open source CFD package capable of solving a number of different governing equations for fluid flow on arbitrary unstructured meshes [20]. It contains
advanced numerical features and has been applied to many geophysical and industrial applications [21, 19, 22].

The remainder of this paper is structured as follows. In the next section we review the mathematical formulation of multiphase flow in porous media. Then, the numerical discretization employed to solve for the flow is described. The method is verified against the Buckley-Leverett problem where a quasi-analytical solution is available [23]. We compare the accuracy and CPU efficiency of adaptive mesh against static mesh with the same number of nodes. Finally, the application of the presented numerical approach for modeling a heap leaching process on a simplified two-dimensional geometry is illustrated.

2 Multiphase flow in porous media

2.1 Governing equations

Darcy’s law along with the conservation of mass equation form the basis of multiphase incompressible flow in porous media [1, Chap. 5]. Assuming no sources or sinks in the domain, the mass conservation for each phase can be written as,

\[ \frac{\partial (\phi \rho_l S_l)}{\partial t} + \nabla \cdot (\rho_l u_l) = 0, \tag{1} \]

where \( \phi \) is the porosity of the porous medium, \( \rho_l \) is the density, \( S_l \) is the saturation and \( u_l \) is the superficial velocity of phase \( l \).

Darcy’s law for phase \( l \) is given by,

\[ u_l = -\frac{K_l}{\mu_l} (\nabla p_l - \rho_l g), \tag{2} \]

where \( p_l \) is the pressure, \( K_l \) is the permeability, \( \mu_l \) is the viscosity of phase \( l \) and \( g \) is acceleration due to gravity.

In addition to the conservation of mass and Darcy’s law, the following relationships must be satisfied to close the system of equations,

\[ \sum_l S_l = 1, \tag{3} \]

\[ p_c (S_w) = p_n - p_w, \tag{4} \]

where \( p_c \) is the capillary pressure, \( p_n \) and \( p_w \) are pressure of non-wetting phase and wetting phase respectively. Hereafter, subscripts \( w \) and \( n \) are used to denote the wetting and non wetting phases respectively.
2.2 Permeability

Permeability ($K_t$) is a measure of ease of fluid flow through a porous medium [1, Chap. 5]. It is decomposed into absolute permeability ($k_a$) and relative permeability ($k_r$). Absolute permeability is a characteristic property of the porous medium which is not affected by the presence of another fluid. However, when more than one fluid exists in the porous medium, the ratio of the effective permeability of each phase to the absolute permeability determines the relative permeability such that

$$K_t = k_a k_r.$$  \hspace{1cm} (5)

For particle packings, the absolute permeability can be calculated with a reasonable error using other properties of the porous medium that can be measured more conveniently [24]. The classical Kozeny-Carman equation is a common way of estimating absolute permeability of monosized packs of spherical particles [25]. However, Garcia et al. [26] showed that the Kozeny-Carman relation overpredicts the permeability of a porous medium consisting of irregular particles. They proposed a correlation for absolute permeability as a function of porosity and the harmonic mean diameter of the particles ($d_p$),

$$k_a = 0.11d_p^2 \phi^{5.6}. \hspace{1cm} (6)$$

Average diameter of particles in a heap leaching process is typically $0.5cm$ and porosity of a heap is around 0.35. Using equation 6, leads to an absolute permeability of $7.7 \times 10^{-9} m^2$.

In contrast to the absolute permeability, relative permeability needs to be updated as the liquid phase infiltrates through the heap. For the heap model, we apply the Brooks-Corey correlations [27],

$$k_{rw} = \left( \frac{S_w - S_{wr}}{1 - S_{wr}} \right)^4,$$  \hspace{1cm} (7)

$$k_{rn} = \left( 1 - \frac{S_w - S_{wr}}{1 - S_{wr}} \right)^2 \left( 1 - \left( \frac{S_w - S_{wr}}{1 - S_{wr}} \right)^2 \right),$$  \hspace{1cm} (8)

where $S_{wr}$ is the residual saturation of the wetting phase.

2.3 Capillary pressure

The pressure of two immiscible fluids has a jump discontinuity across their interfaces. This pressure difference is the capillary pressure and defined as $p_c = p_n - p_w$. In the
heap leaching process, air is considered as the non-wetting phase and the leaching solution is the wetting phase. Brooks and Corey [27] presented a correlation for the capillary pressure such that

$$p_c(S_w) = p_e \left( \frac{S_w - S_{wr}}{1 - S_{wr}} \right)^{-\frac{1}{\alpha}},$$  \hspace{1cm} (9)

where $p_e$ is the entry pressure and $\alpha$ is the pore size distribution index. Entry pressure can be calculated by dividing the surface tension of the non-wetting phase in contact with the wetting phase to the characteristic length of the porous medium. In the heap leaching application, it is approximated as the surface tension of water in contact with air ($0.07 \frac{N}{m}$) divided by the average particles' diameter (0.5cm). This gives us an entry pressure of $p_e = 14Pa$ and we assume $\alpha = 2$ for the pore size distribution index.

3 Numerical method

With some algebraic manipulations, the governing equations can be rewritten as pressure and saturation equations. Assuming incompressible fluids and substituting $u_l$ from equation 2 into equation 6, the saturation equation is expressed as

$$\frac{\partial (\phi S_l)}{\partial t} = \nabla \cdot \left[ \frac{K_l}{\mu_l} \nabla p_l - \frac{K_l \rho_l}{\mu_l} \mathbf{q} \right] = 0.$$  \hspace{1cm} (10)

The pressure equation is then formed by summing equation 10 for all phases and using the constraint from equation 3,

$$\nabla \cdot \left[ \sum_l \left( \frac{K_l}{\mu_l} \nabla p_l - \frac{K_l \rho_l}{\mu_l} \mathbf{q} \right) \right] = 0.$$

The primary variables to be solved are $p_n$ and $S_w$ from which $p_w$ and $S_n$ can be obtained diagnostically using equations 3 and 4.

3.1 Temporal discretization

For temporal discretization, we employ the IMPES algorithm. In this algorithm, the pressure equation at time $t^{n+1}$ is solved implicitly while the saturation is at the previous time level, $t^n$,

$$\nabla \cdot \left[ \sum_l \left( \frac{K_l^n}{\mu_l} \nabla p_l^{n+1} - \frac{K_l^n \rho_l}{\mu_l} \mathbf{q} \right) \right] = 0.$$  \hspace{1cm} (12)

Solving for the pressure implicitly, the saturation equation is solved explicitly. Assuming that the porosity is invariant in time in equation 10, we obtain
Fig. 1 The node centered control volume mesh (shown by solid lines) is constructed around continuous finite element mesh (shown by dashed lines) by connecting the centroids of the neighboring finite elements to the edge midpoints [20].

\[
\frac{\phi(S_i^{n+1} - S_i^n)}{\Delta t} = \nabla \cdot \left[ \frac{K_i^n}{\mu_i} \nabla p_i^{n+1} - \frac{K_i^n \rho_i}{\mu_i} q \right].
\]  

(13)

3.2 Spatial discretization

Since the pressure is discretized by a finite element basis and saturation is solved on node centered control volume basis, they are written as,

\[
p^n = \sum_{j=1}^{M} \tilde{p}_j^n N_a,
\]

(14)

\[
S^n = \sum_{j=1}^{M} \tilde{S}_j^n M_b,
\]

(15)

where \( N_a \) is standard Lagrangian finite element basis, \( M_b \) is 1 inside the control volume \( b \) constructed around the node \( j \) (see Figure 1) and zero everywhere else. The \( \tilde{p}_j \) and \( \tilde{S}_j \) are the pressure and saturation values at node \( j \).

To generate a linear system of equations for pressure, a Petrov-Galerkin weighted residual method is used. This involves multiplying equation 12 by a control volume based weight function and integrating over \( \Omega \),

\[
- \int_{\Omega} M_b \nabla \cdot \left( \sum_i \left( \frac{K_i^n}{\mu_i} \nabla p_i^{n+1} - \frac{K_i^n \rho_i}{\mu_i} q \right) \right) dV = 0.
\]

(16)

Since \( M_b \) is 1 inside control volume \( b \) and zero everywhere else, we obtain
\[ - \int_{\Omega_{CV_b}} \nabla \cdot \left( \sum_l \left( \frac{K^n_l}{\mu_l} \nabla p^{n+1}_l - \frac{K^n_l \rho_l}{\mu_l} \frac{q}{2} \right) \right) dV = 0. \]  

(17)

Applying the divergence theorem, we obtain

\[ - \int_{\Gamma_{CV_b}} \mathbf{n} \cdot \left( \sum_l \left( \frac{K^n_l}{\mu_l} \nabla p^{n+1}_l - \frac{K^n_l \rho_l}{\mu_l} \frac{q}{2} \right) \right) d\Gamma = 0, \]  

(18)

where \( \Gamma_{CV_b} \) is the control volume surface bounding \( CV_b \) and \( \mathbf{n} \) is the unit outward pointing normal. The assembly of the linear system for pressure is now reduced to integration over the control volume surfaces. The absolute permeability and viscosity are element-wise constant. The relative permeability is represented with the same control volume basis as the saturation. An upwind scheme is applied for its face value and thus the saturation of the control volume located upstream of the face is employed for the calculation of the relative permeability.

To generate a linear system for saturation, a standard node centered control volume weight and basis is used to obtain

\[ \int_{\Omega_{CV_b}} \frac{\phi S^{n+1}}{\Delta t} dV = \int_{\Gamma_{CV_b}} \mathbf{n} \cdot \left( \frac{K^n_l}{\mu_l} \nabla p^{n+1}_l - \frac{K^n_l \rho_l}{\mu_l} \frac{q}{2} \right) d\Gamma + \int_{\Omega_{CV_b}} \frac{\phi S^n}{\Delta t} dV. \]  

(19)

For consistency, the relative permeability face values used in the pressure matrix are also used for the saturation advection matrix. This ensures that the discrete saturation equations summed over all phases produce the discrete pressure equation.

4 Results and discussions

4.1 The Buckley-Leverett problem

We compare our numerical results against the quasi-analytical solution for the Buckley-Leverett problem to show the numerical accuracy of the scheme and check its order of convergence. The porous medium is initially saturated with phase 1 \( (S_1 = 1) \). Phase 2 displaces the phase 1 by a constant flux. We assume a uniform absolute permeability of \( 10^{-8} m^2 \), a uniform porosity of 0.5 and a displacing velocity of \( u = 10^{-3} m.s^{-1} \) on the inlet. The viscosity of both phases is \( 10^{-5} Pa.s \). The relative permeability functions are

\[ k_{r1} = S_1^2, \]  

(20)
\[ k_{r2} = (1 - S_1)^2. \] (21)

Figure 2 shows the comparison of the saturation and pressure profiles obtained from numerical simulation using 10, 20, 40, and 80 equidistant one-dimensional elements with the quasi-analytical solution after 0.4 PVI (pore-volume injected). The numerical simulation shows a very good agreement with the quasi-analytical solution. The simulation results improve for finer meshes for both pressure and saturation and the analytical front saturation is matched very closely for 80 elements as it can be seen in Figure 2 (a). As Figure 2 (b) indicates, the pressure ahead of the front matches the quasi-analytical solution for all cases since the flow in that region is single phase.

We test the order of convergence of the scheme based on the error in predicting the saturation and pressure profiles for each mesh. We obtain the \( L_1 \) norm of error for saturation from

\[ E_S = \|S - S_A\|_1 = \int_0^1 \|S - S_A\| \, dx, \] (22)

where \( E_S \) is the \( L_1 \) norm of error for saturation and \( S_A \) is the quasi-analytical saturation. The same error can be defined for the pressure profile \( (E_P) \).

Figure 3 (a) shows the \( L_1 \) norm of error for saturation for different meshes, along with a line with slope of 0.9, on a double logarithmic plot. Error in predicting pressure is also shown on Figure 3 (b) along with a line with slope of 0.9 . The results imply that the order of convergence for both saturation and pressure is 0.9. This is consistent with the use of the first order control volume scheme.

4.2 A two-dimensional heap model

Assuming that the transport within a heap occurs in a vertical two-dimensional plane, we simulate the two-phase flow of air and leaching solution on a two-dimensional geometry as shown in Figure 4.

The width and height of the heap are assumed to be 100 and 15 meters respectively. The porosity and permeability are assumed to be uniform and invariant in time - \( \phi = 0.35 \) and \( k_a = 7.7 \times 10^{-9} m^2 \). Density and viscosity of air are 1.2\( kg \cdot m^{-3} \) and 1.8 \( \times 10^{-5} Pa \cdot s \) respectively. For the leaching solution, these two parameters are assumed to be 1010\( kg \cdot m^{-3} \) and 8.9 \( \times 10^{-4} Pa \cdot s \).

As the initial conditions, the pressure of air is assumed to be zero within the heap. The heap is initially saturated with air \( (S_n = 1) \). The leaching solution is applied to the heap from the top at a constant flux of \( u_w = 10 mm \cdot hour^{-1} \). On the sides, we assume no-flow boundary conditions. The residual saturation of the leaching solution in the heap is assumed to be 0.15.
Fig. 2 Saturation (a) and pressure (b) profile for 10, 20, 40, and 80 elements in comparison with the quasi-analytical solution.
Fig. 3 L1 norm of error for saturation (a) and pressure (b) versus element edge length for the analysis of the order of convergence. Dashed lines have a slope of 0.9.
The time step size is 10 seconds. Dynamic mesh adaptivity is applied based on the curvature of saturation. After each 100 time steps, the curvature of saturation in the horizontal and vertical directions are calculated. Then, a new mesh is generated to capture the sharp boundaries between the saturated and unsaturated regions with a minimum number of elements. The minimum edge length is $0.3$ m and $5$ m in the vertical and horizontal directions respectively. The maximum allowed edge length in the vertical and horizontal directions is $3$ m and $50$ m respectively.

Figure 5 shows the saturation distribution and the generated anisotropic mesh within the heap after $t = 30$ hours. It can be seen that fine elements are formed in the region of heap where the saturation rapidly varies, as expected. Looking at the saturation profile, we can see the sharp boundary between the saturated and unsaturated regions is captured accurately.

To highlight the advantage of the dynamic mesh adaptivity for simulation of heap leaching, we compare our adaptive results against a static mesh model of heap. The number of nodes of the static mesh is equal to the average number of nodes of adaptive mesh simulation of liquid infiltration (before reaching steady state). The static mesh contains 255 nodes. Figure 6 shows the discretized heap using static mesh and the saturation distribution within the heap after $t = 30$ hours. Since the mesh is not updated based on the flow, larger discretization error is observed in comparison with dynamic adaptive mesh with the same number of nodes (see Figures 6 b and 5 b).

Figure 7 shows the saturation profile over a vertical line at $x = 50$ m using static and adaptive mesh. The adaptive mesh produces a sharp saturation front. Due to numerical dispersion, the boundary between the saturated and unsaturated regions is smeared for static mesh. In addition, the predicted breakthrough time using static mesh...
and adaptive mesh are 105 and 112 hours respectively. Due to numerical dispersion in saturation calculation, static mesh predicts an early breakthrough time for the leaching solution.

Figure 8 shows the number of nodes during the numerical simulation. For adaptive mesh simulation, as the saturation front moves out of the heap, the number of nodes gradually reduces to 133. This is an important feature of our adaptive scheme for simulation of long-term processes such as heap leaching. Using mesh adaptivity, long-term heap leaching can be simulated more efficiently with less number of elements to discretize the geometry after the steady state.

Figure 9 shows the CPU time for 1000 successive time steps during the simulation. The simulations were performed on a PC with a 3.00GHz CPU. Before the steady state, applying adaptive mesh requires on average 5% more CPU time in comparison with the static mesh with the same number of nodes. In other words, the additional computations for adapting the mesh increases the overall CPU time by only a small fraction. Note that the adaptive mesh reduces the numerical dispersion error significantly while requires 5% excessive CPU time. Moreover, as the system reaches the steady state, the adaptive method reduces CPU time by approximately 31% relative to the static mesh.
This is because of the reduction in the number of nodes as the wetting front leaves the heap (8). In addition, we have performed another simulation using static mesh with the same numerical dispersion error as the adaptive mesh. This high resolution static mesh has 21245 nodes. Even though, the accuracy of the adaptive mesh and the high resolution static mesh is similar, the adaptive mesh is approximately 46 times faster than the high-resolution static mesh.

Figure 10 shows the average saturation of the leaching solution within the heap versus time. The average saturation increases approximately linearly within the heap until the breakthrough time. Following the breakthrough of the leaching solution, the average saturation does not vary considerably as the process can be considered at a steady state. This numerical observation is in a good agreement with the physical behaviour measured by Ilankoon and Neethling [28] in packed beds of particles with similar liquid addition rate to heap leaching. Their measurements indicate the same trend for variation of saturation where the liquid holdup increases linearly until the breakthrough time. After that, the liquid holdup remains constant.
Fig. 7 Vertical saturation profile at $x = 50m$ within the heap using static and adaptive mesh. The sharp saturation front is smeared using static discretization.

Fig. 8 The number of discrete nodes for the heap leaching simulation versus time using adaptive and static mesh.
**Fig. 9** CPU time for 1000 successive time steps during the heap leaching simulation using adaptive and static mesh.

**Fig. 10** Average saturation of the leaching solution within the heap versus time.
5 Conclusion

This paper presented a control volume finite element method for numerical simulation of multiphase flow in porous media. The governing equations are conservation of mass and Darcy's law. Pressure is discretized spatially using a control volume finite element method. For saturation, a node centered control volume method is employed. The numerical scheme is equipped with dynamic mesh adaptivity that can achieve high spatial accuracy for multiscale problems at low computational cost. The applied mesh adaptivity algorithm is general and can be used to adapt the mesh based on any field of interest. The accuracy of the scheme to solve for pressure and saturation is verified by comparing the numerical results against the quasi-analytical solution for the Buckley-Leverett problem. In addition, the order of convergence of the method for pressure and saturation is calculated to be 0.9. We showed the application of the method for the effective simulation of flow in heap leaching. We examined the use of mesh adaptivity to capture the sharp boundaries between the unsaturated and saturated regions and compared our results against a static mesh with the same number of nodes. The adaptivity algorithm improves the numerical accuracy significantly while it increases CPU time only by approximately 5% before the steady state. When the system reaches the steady state, the adaptive mesh is faster than the static mesh by approximately 31 %. The developed flow simulator is a first step for a comprehensive and efficient large-scale simulation of reactive flow in porous media and heap leaching.

References

3. van Zyl DJA, Hutchison IPG, Kiel JE (1988) Introduction to evaluation, design, and operation of precious metal heap leaching projects. Society of Mining Engineers, Colorado


27. Brooks RH, Corey AT (1964) Hydraulic properties of porous media. Hydrology Papers, Colorado State University, Colorado