A parallel scientific software for heterogeneous hydrogeoloy

J. Erhel [a] *, J.-R. de Dreuzy [b], A. Beaudoin [c], E. Bresciani [a], and D. Tromeur-Dervout [d]

[a]INRIA,35042 Rennes Cedex, France

[b]Geosciences Rennes, UMR CNRS 6118, France

[c]LMPG, University of Le Havre, France

[d]CDCSP, University of Lyon 1, France

1. Introduction

Numerical modelling is an important key for the management and remediation of groundwater resources [28]. As opposed to surface water and to highly karstic geologies, groundwater does not flow in well-identified open streams but is like water flowing in the voids of a sponge. Groundwater is highly dependent on the percentage of pores (porosity), size and connectivity of the pores that controls the permeability of the medium. Its quality depends on the biochemical reactivity of the crossed geological media and on the kinetics of the biochemical reactions. These parameters (porosity, permeability, reactivity) are highly variable. Several field experiments show that the natural geological formations are highly heterogeneous, leading to preferential flow paths and stagnant regions. The contaminant migration is strongly affected by these irregular water velocity distributions. Transport of contaminant by advection and dispersion induce a large spreading of the particles generally called plume. The characterization of the plume remains a much debated topic [7,16,25]. Fractures and heterogeneous sedimentary units cannot be identified whatever the remote data, whether geophysical, geological or hydraulic. Because data give a rather scarce description of the medium hydraulic properties, predictions rely heavily on numerical modelling. Numerical modelling should integrate the multi-scale geological heterogeneity, simulate the hydraulic flow and transport phenomena and quantify uncertainty coming from the lack of data. Analytical techniques like homogenization or perturbations are in general not relevant. Modelling must thus be performed in a probabilistic framework that transfers the lack of data on prediction variability [1]. Practically, random studies require running a large number of simulations, for two reasons. First, non intrusive Uncertainty Quantification methods rely on sampling of data. Second, the questions addressed must consider a large panel of parameters (Peclet number, variance of probabilistic models, etc). The hydraulic simulations must be performed on domains of a large size, at the scale of management of the groundwater resource or at the scale of the homogeneous medium type in terms of geology. This domain must be discretized at a fine resolution to take into account the scale of geological heterogeneities. Also, large time ranges must be considered in order to determine an asymptotic behaviour. High performance computing is thus necessary to carry out these large scale simulations.

Therefore we have developed the parallel scientific platform HYDROLAB, which integrates in a modular structure physical models and numerical methods for simulating flow and transport

*this work was partly funded by the Grid'5000 grant from the French government

DOI: 10.1007/978-3-540-92744-0_5,

in heterogeneous porous and fractured media. It includes stochastic models and Uncertainty Quantification methods for dealing with random input data. It relies as much as possible on existing free numerical libraries such as parallel sparse linear solvers. This policy increases the global reliability and efficiency of HYDROLAB. Our software is also fully parallel and follows a powerful object-oriented approach. Most existing codes for groundwater flow and transport contaminant do not gather all these original features. To our knowledge, there is no public parallel code performing groundwater simulations in 3D fracture networks. Popular free software for reservoir modeling, environmental studies or nuclear waste management, such as MODFLOW [19], MT3DMS [29] and TOUGH2 [24] are not easy to use for highly heterogeneous porous media; moreover, MODFLOW and MT3DMS are not parallel. Other software, such as IPARS [26] or ALLIANCES [23], are private or commercial.

Our paper is organized as follows: we first describe the physical models processed by HYDROLAB; then we describe the numerical methods and algorithms implemented in the software with details about parallel features. After a short synopsis on HYDROLAB, we discuss performance results on parallel clusters of various numerical experiments. The paper ends up with concluding remarks about future work.

2. Physical models

2.1. Geometry and data

We consider two types of geological domains, either porous media or fractured media. Permeability values span more than ten orders of magnitude from 10^{-12} m/s in non fractured crystalline rocks or clays to 10^{-1} m/s for gravels [14]. Variability can be huge not only at the transition between geological units, like between sand and clay, but also within units. Since data is lacking, the physical model of porous media is governed by a random permeability field, following a given distribution law. In order to cover a broad-range of structures, we consider finitely correlated, fractal or multifractal distribution laws. The geometry is kept very simple and is a deterministic 2D or 3D regular polyhedron. In rocks, the fractures can represent only 0.01 % of the volume although they are the sole water carrying structures. Fractured media are by nature very heterogeneous and multi-scale, so that homogenisation approaches are not relevant. In an alternative approach, called the discrete approach, flow equations are solved on the fracture network, explicitly taking into account the geometry of the fractured medium, and thus the structure of the connected network. According to recent observations of fractured rocks, fractures are characterized by a variety of shapes and a broad range of lengths and apertures. The physical model is thus based on a complex and random geometry, where the geological medium (the surrounding matrix) is a cube; fractures are ellipses with random distributions of eccentricity, length, position and orientation. The main feature of our model is the fracture length distribution, which is given by a power law. We first consider an impervious matrix, because almost no water flows in it [9,6]. For some applications, it is also important to consider the coupling between the matrix and the fractures.

2.2. Governing equations

Well tests are the first source of information on the hydraulic properties of underground media. Numerical modelling of these tests involves computation of transient flow in large domains. We assume here that the medium is saturated and that the water density is constant. Flow is governed by the classical Darcy law and the mass conservation law:

$$\begin{cases} \epsilon v = -K\nabla h, \\ s\partial h/\partial t + \nabla.(\epsilon v) = q \end{cases} \tag{1}$$

where the Darcy velocity v and the hydraulic head h are unknown, K is a given permeability field, ϵ is a given porosity, s is a given storativity parameter and q is a given source term. An initial condition and boundary conditions close the system. For steady-state flows, the time derivative disappears. In order to characterize plumes of contaminant, numerical simulations consist in computing the velocity field over large spatial domains and simulating solute transport over large temporal scales [16]. We consider here one inerte solute and a steady-state flow. Governing equations are:

$$\frac{\partial(\epsilon c)}{\partial t} + \nabla.(\epsilon c v) - \nabla.(\epsilon D \nabla c) = 0 \tag{2}$$

where c is the unknown solute concentration, v is the previous Darcy velocity and D is the given dynamic dispersion tensor. Boundary conditions and an initial condition complete the equation.

3. Numerical methods

3.1. Uncertainty Quantification methods

The physical model is not complex in itself since equations are linear but both the broad geological heterogeneity and the lack of measures require dealing with uncertain physical and geometrical properties. The permeability field is a random variable and the geometry of fracture networks is random. Hydrogeological studies aim at describing the random velocity field and the random dispersion of solute. They must rely on Uncertainty Quantification methods. Currently, we use a classical Monte-Carlo approach [8]. It is non intrusive in the sense that each sample is a deterministic simulation, using classical spatial and temporal discretizations. It is easy to implement and can be applied to any random field, either data or geometry. However, only the mean of random output can be attained with few samples; other statistical results require much more samples. Therefore, more sophisticated UQ methods can be considered, such as non intrusive spectral methods or intrusive Galerkin methods [27,1].

3.2. Spatial and temporal discretizations

For each sample, we have to discretize the set of Partial Differential Equations. In flow computations, we use a method of lines in the case of transient equations and get a system of ODEs after spatial discretization. For porous media, the geometry is simple and the mesh is a regular grid. In both transient and steady-state cases, we choose a finite volume method, which computes accurately the velocity and ensures mass conservation. For fracture networks, the geometry is complex, so that it is quite difficult to generate a mesh. We have designed a two-fold mesh generator, with a preprocessing step before meshing each fracture separately [3]. We first discretize the intersections, so that they are uniquely defined in the mesh. Then, by considering these discretized intersections as boundaries, we mesh each fracture with a 2D mesh generator. We consider steady-state flows and use a hybrid mixed finite element method, based on the unstructured mesh. Again, the method computes accurately the velocity and ensures mass conservation. The unknowns in the hybrid method are the hydraulic head in each cell and on each face, so that the continuity equations are easy to implement at the intersections.

At each time step, we get a large sparse symmetric positive definite linear system. Direct solvers are efficient for medium size systems but suffer from too big memory requirements for large size systems. Therefore, we prefer to use iterative solvers. Algebraic multigrid methods appear to be very efficient, either as such or as preconditioners of Conjugate Gradient method [4].

In transport computations, we choose a random walker method, because it does not introduce numerical dispersion in advection-dominated problems. We generate a bunch of particles for

the initial condition and solve a stochastic differential equation to compute their trajectories. In the case of pure advection, this is simply a deterministic particle tracker method, whereas dispersion is modeled by a random motion. We use an explicit first-order Euler scheme to solve the equation.

3.3. Parallel algorithms

In each simulation, the spatial and temporal dimensions are very large, in order to study multi-scale effects and asymptotic behaviour. Numerical modelling must overcome two main difficulties, memory size and runtime, in order to solve very large linear systems and to simulate over a large number of time steps. High performance computing is thus necessary to carry out these large scale simulations.

Because memory requirements are very high, it is mandatory to distribute data from the beginning to the end of the simulation. Therefore, everything is done in parallel, including data generation, mesh generation, flow computation, transport computation. Parallel algorithms are based on a subdomain decomposition. In porous media, we subdivide the domain into regular blocks, which can be slices or rectangles in 2D. Each processor owns a subdomain and ghost cells surrounding the domain. This classical tool reduces communication costs between processors. In fracture networks, we subdivide the network into groups of fractures, assigned to different processors. Computations are done in parallel and communications occur for synchronizing data at the intersections.

Once the data and the mesh are generated, it is possible to launch flow computations and transport computations. In flow computations, we use a parallel sparse linear solver. We have developed interfaces to several libraries, such as SUPERLU [22], PSPASES [18], HYPRE [12]. The distributed matrix and right-hand side are input data to the solver. Parallel computations inside the solver may redistribute the matrix but the output solution is distributed as the right-hand side. The velocity field is computed in each processor from the hydraulic head returned by the solver. Thanks to this structure, we can compare different sparse solvers. We require a linear complexity with the matrix size N, a robust solver for heterogeneous permeability fields, parallel scalability. We observe that direct solvers are robust and scalable but have a complexity in $O(N^{1.5})$ [4]. Preconditioned Conjugate Gradient methods have also a poor complexity [10]. Domain decomposition methods seem very promising [17], thus we are developing such methods [11], extending the Aitken-Schwarz methodology [15]. Algebraic multigrid methods have a linear complexity and are robust [10]. Therefore, in this paper, we compare geometric and algebraic multigrid methods, using HYPRE.

In transport computations, we have designed a simple but efficient parallel algorithm which allows to consider large temporal domains [2]. Our parallel algorithm takes advantage of a subdomain decomposition and of the non interaction between the particles. This hybrid parallel strategy is quite original, since most of previous work uses either one or the other [20],[5],[21]. This strategy allows to run simulations on very large computational domains. Actually, we use the domain decomposition and data distribution arising from permeability generation and flow computation [3]. Communications with processors owning neighboring subdomains must occur when particles exit or enter the subdomain. We add a global synchronisation point after the exchanges of particles, in order to define a global state where each processor knows the total number of particles still in the domain. By the way, it guarantees that all RECV operations have been successful and that the algorithm may execute new SEND operations. Thus neither deadlock nor starvation may occur and the algorithm terminates safely. We also take advantage of the independence of the particles and launch them by bunches. The idea is to initiate a pipeline of particles. At each global synchronisation point, the master processors inject a new

bunch of particles, until all particles have been injected into the domain. All processors track the particles in their subdomain, then exchange particles on their boundaries, finally synchronize globally to define the total number of particles in activity.

4. Parallel software HYDROLAB

We have developed a fully parallel object-oriented software platform, called HYDROLAB, which provides a generic platform to run Monte-Carlo numerical simulations of flow and transport in highly heterogeneous porous media and in fracture networks. The software is organized in three main parts, respectively dedicated to physical models, numerical methods and various utilitaries. Physical models include heterogeneous porous media and discrete fracture networks. We plan to develop a coupled model for porous fractured media. The physical model is composed of several tasks: the generation of the random geometry and the random permeability field, the mesh generation, the spatial discretization of the flow operator. The numerical methods include PDE solvers, ODE solvers, linear solvers, multilevel methods, a random walker, UQ methods. Utilitaries are generic modules including Input/Output, results management, parallel tools, grid tools, geomotry shapes, visualization, etc. This modularity and genericity allows a great flexibility and portability, enhancing the development of new applications. The software is written in C++ and is implemented on machines with Unix or Windows systems. Graphical functions are written with OpenGL and parallel programming relies on the MPI library. The software integrates open-source libraries such as sparse linear solvers for flow computation.

Our commitment is to design free packages available for downloading on the website of the platform (http://hydrolab.univ-rennes1.fr/).

5. Numerical results for 2D porous media

We have done multiparametric Monte-Carlo simulations for solving steady-state flow and transport problems with 2D heterogeneous porous media [8]. We have done many experiments on a parallel cluster of the Grid'5000 computing resource installed at INRIA in Rennes and we got good performances with parallel sparse linear solvers and our parallel particle tracker. Some results can be found in [3], [4] and [2].

5.1. Numerical experiments

In this paper, we study the case of a 2D porous medium where the permeability field K follows a log-normal correlated law, where $Y = ln(K)$ is defined by a mean m and a covariance function C given by $C(r) = \sigma^2 \exp(-\frac{|r|}{\lambda})$, where σ^2 is the variance of the log hydraulic conductivity, $|r|$ represents the separation distance between two points and λ denotes the correlation length scale. The length λ is typically in the range $[0.1m, 100m]$ and the variance σ^2 is in the interval $[0, 7]$. These two ranges encompass most of the generally studied values.

We consider a steady-state flow and solve first the flow equations, then the transport equations of an inerte solute with the computed velocity. In flow equations (1) and transport equations (2), we assume a constant porosity $\epsilon = 1$. In flow equations, boundary conditions are no-flow on upper and lower sides and specified-head $h = 0$ on left side, $h = 1$ on right side. In transport equations, we consider only molecular diffusion, assumed homogeneous and isotropic. The ratio of diffusion to advection is measured by the Peclet number defined by $Pe = \frac{\lambda U}{D_m}$, where D_m is the molecular diffusion coefficient, with a typical range of $[100, \infty]$, where ∞ means pure advection. This range covers advection-dominated transport models, whereas diffusion-dominated models require smaller values. In transport equations, the initial condition at $t = 0$ is the injection of the solute. In order to overcome the border effects, the inert solute is injected at a given distance

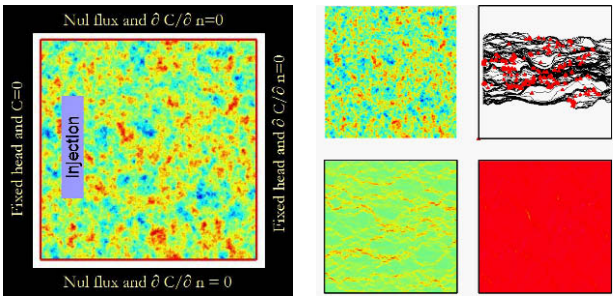

Figure 1. Numerical conditions (left) and simulation result for heterogeneous porous media (right): permeability field (top left), solute transport (top right), horizontal velocity field (bottom left), vertical velocity field (bottom right).

Parameter	**Values**
σ	$1, 2, 2.5, 3$
λ	10
Pe	$100, 1000, \infty$
Δx	1
n	$128, 256, 512, 1024, 2048, 4096$
tol	10^{-11}
N_p	2000
solver	SMG,AMG, TRACKER

Table 1
Physical and numerical parameters used for heterogeneous 2D porous medium with log-normal correlated permeability

of the left side. Boundary conditions are no-mass-flux on upper and lower sides, specified zero concentration on left side, free mass outflow on right side. An example of simulation output is depicted in Figure 1.

5.2. Complexity analysis

Many physical parameters have a strong impact on the computations, such as the variance, the correlation length or the Peclet number. Also the computational requirements are governed by numerical parameters such as linear solver, problem size and convergence thresholds. Here, we consider a square domain with as many cells in each direction. The problem size is defined by the mesh step Δx, the number of cells n in each direction of the mesh and by the number of particles N_p in the random walker. In iterative sparse linear solvers, another numerical parameter is the threshold tol for convergence. Physical and numerical parameters are summarized in Table 1. The number of particles is large enough to ensure convergence of the random walker and the mesh step is small enough to get an accurate flow field. The correlation length is also fixed. Here, we consider multigrid sparse linear solvers: SMG and Boomer-AMG from the HYPRE library [13], [12].

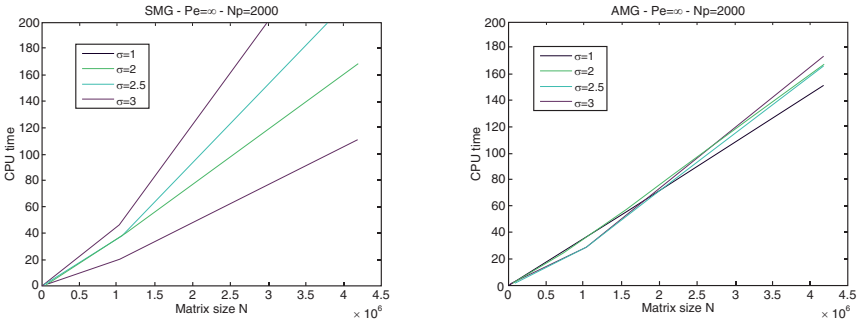

Figure 2. CPU time of flow computations using SMG geometric multigrid (left) and AMG algebraic multigrid (right). The matrix size is $N = n^2$.

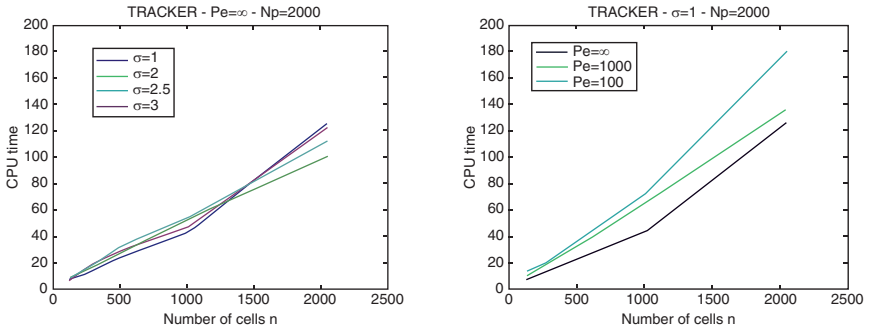

Figure 3. CPU time of transport computations using a particle tracker. Impact of the variance (left) and of the Peclet number (right). The number of cells is each direction is n.

One of the objectives of the simulations is to analyze the effect of heterogeneity and Peclet number on solute dispersion. Therefore, it is very important to develop a software with computational times as much independent of the variance σ and the Peclet number Pe as possible. Another objective is to estimate the asymptotic dispersion; therefore, we have to run the transport simulation with very large times; because all particles of the random walker must stay in the domain, this implies to run flow and transport simulations in very large computational domains. We analyze the effect of the size of the computational domain.

In Figure 2, we plot the CPU time of AMG and SMG multigrid solvers versus the matrix size $N = n^2$, for different values of σ. We observe in both cases a linear complexity, as expected from the theory. However, the behaviors are different when σ varies. The geometric multigrid solver SMG is more efficient for $\sigma = 1$, the geometric solver SMG and the algebraic solver AMG are equivalent for $\sigma = 2$ and AMG becomes more efficient for $\sigma = 2.5$ and $\sigma = 3$. Therefore, the best choice depends on the value of σ.

In Figure 3, we plot the CPU time of the particle tracker versus the number of cells n in each

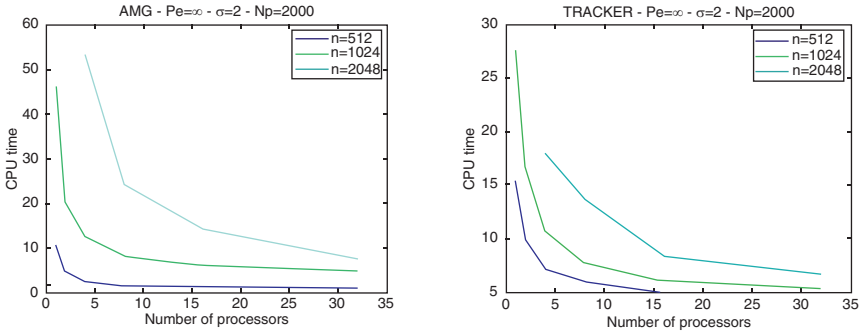

Figure 4. Parallel CPU time of flow (left) and transport (right) computations using respectively algebraic multigrid (AMG) a particle tracker. Impact of the matrix size $N = n^2$. The heterogeneity is driven by $\sigma = 2$ and transport is pure advection with $Pe = \infty$.

direction. The tracker is nearly independent of the variance σ. We observe a complexity which is roughly linear with n (thus with $\sqrt{N}$). For small values of n, the transport computation is slower than the flow computation, but the CPU times become equivalent for $n = 2048$ and the transport will become much faster than the flow for larger n because of the different complexities.

Whereas flow computations are obviously independent of the Peclet number, we observe that the computational time of transport computations increases with the Peclet number. However, for these large values of Pe, the transport computations remains cheaper than the flow computations with a large number of cells.

5.3. Scalability analysis

From the sequential results, it is clear that the algebraic multigrid solver is very efficient, has a linear complexity with the number of cells in the mesh and is not sensitive to the heterogeneity. Therefore, we choose this solver for parallel flux computations. We run experiments on a cluster composed of two nodes of 32 computers each. Each computer is a 2.2 Ghz AMD Opteron bi-processor with 2 Go of RAM. Inside each node, computers are interconnected by a Gigabit Ethernet Network Interface, and the two nodes are interconnected by a Gigabit Ethernet switch. In Figure 4, we plot the CPU time of flux (using AMG) and transport (using our particle tracker) computations with a varying number of processors P. Here, the heterogeneity is $\sigma = 2$ and we consider a pure advection case ($Pe = \infty$). We observe good speed-ups for the different sizes of the computational domain and a quite good scalability for both parallel modules. We did more experiments with various σ and Pe and got very similar results.

6. Current and future work

We have designed a parallel modular software, called HYDROLAB, for solving flow and transport problems in heterogeneous and fractured geological media. Our performance results show that our software is very efficient on parallel clusters for dealing with random 2D heterogeneous porous media. We are extending this work to 3D domains and have designed multilevel methods based on subdomain decompositions. As far as fracture networks are concerned, we are now able to generate a mesh for any network and to solve steady-state flow problems [3]. We are doing more experiments for improving parallel performances.

Future work concern transient flow problems in porous and fractured media and transport problems in fracture networks. Also, we plan to investigate non intrusive Uncertainty Quantification methods, in order to improve efficiency and to get more statistical information than with Monte-Carlo simulations.

REFERENCES

1. I. Babuska, R. Tempone, and G. Zouraris. Solving elliptic boundary value problems with uncertain coeffcients by the finite element method: the stochastic formulation. *Computer methods in applied mechanics and engineering*, 194:1251–1294, 2005.
2. A. Beaudoin, J-R. de Dreuzy, and J. Erhel. An efficient parallel particle tracker for advection-diffusion simulations in heterogeneous porous media. In A.-M. Kermarrec, L. Boug, and T. Priol, editors, *Euro-Par 2007, LNCS 4641*, pages 705–714. Springer-Verlag, Berlin, Heidelberg, 2007.
3. A. Beaudoin, J-R. de Dreuzy, J. Erhel, and H. Mustapha. Parallel simulations of underground flow in porous and fractured media. In G.R. Joubert, W.E. Nagel, F.J. Peters, O. Plata, P. Tirado, and E. Zapata, editors, *Parallel Computing: Current and Future Issues of High-End Computing*, volume 33 of *NIC Series*, pages 391–398. NIC, 2006.
4. A. Beaudoin, J. Erhel, and J.-R. de Dreuzy. A comparison between a direct and a multigrid sparse linear solvers for highly heterogeneous flux computations. In *Eccomas CFD 2006*, volume CD, 2006.
5. J.-R. Cheng and P. Plassmann. The accuracy and performance of parallel in-element particle tracking methods. In *Proceedings of the Tenth SIAM Conference on Parallel Processing for Scientific Computing*, pages 252 – 261. Portsmouth, VA, 2001.
6. C. Clauser. Permeability of crystalline rock. *Eos Trans. AGU*, 73:237–238, 1992.
7. G. Dagan, A. Fiori, and I. Jankovic. Flow and transport in highly heterogeneous formations: 1. conceptual framework and validity of first-order approximations. *Water Resources Research*, 9, 2003.
8. J.-R. de Dreuzy, A. Beaudoin, and J. Erhel. Asymptotic dispersion in 2D heterogeneous porous media determined by parallel numerical simulations. *Water Resource Research*, to appear.
9. J-R. de Dreuzy, P. Davy, and O. Bour. Hydraulic properties of two-dimensional random fracture networks following power law distributions of length and aperture. *Water Resources Research*, 38(12), 2002.
10. R. Detwiler, S. Mehl, H. Rajaram, and W. Cheung. Comparison of an algebraic multigrid algorithm to two iterative solvers used for modeling ground water flow and transport. *groundwater*, 40(3):267–272, 2002.
11. J. Erhel, A. Frullone, D. Tromeur-Dervout, and J.-R. de Dreuzy. Aitken-Schwarz DDM to solve Darcy flow in heterogeneous underground media. In *Parallel Matrix Algorithms and Applications (PMAA06)*, 2006.
12. R. D. Falgout, J. E. Jones, and U. Meier Yang. Pursuing scalability for Hypre's conceptual interfaces. *ACM Trans. Math. Softw.*, 31(3):326–350, 2005.
13. R.D. Falgout, J.E. Jones, and U.M. Yang. *Numerical Solution of Partial Differential Equations on Parallel Computers*, chapter The Design and Implementation of Hypre, a Library of Parallel High Performance Preconditioners, pages 267–294. Springer-Verlag, 2006.
14. Alan R. Freeze and John A. Cherry. *Groundwater*. Prentice Hall, 1979.
15. M. Garbey and D. Tromeur-Dervout. On some Aitken-like acceleration of the Schwarz method. *Internat. J. Numer. Methods Fluids*, 40(12):1493–1513, 2002.

16. L. Gelhar. *Stochastic Subsurface Hydrology*. Engelwood Cliffs, New Jersey, 1993.
17. I. G. Graham, P. O. Lechner, and R. Scheichl. Domain decomposition for multiscale PDEs. *Numerische Mathematik*, 106(4):589–626, 2007.
18. A. Gupta, F. Gustavson, M. Joshi, G. Karypis, and V. Kumar. Pspases: An efficient and scalable parallel sparse direct solver. In Tianruo Yang, editor, *Kluwer International Series in Engineering and Computer Science*, volume 515, 1999.
19. A.W. Harbaugh, E.R. Banta, M.C. Hill, and M.G. McDonald. MODFLOW-2000, the U.S. geological survey modular ground-water model – user guide to modularization concepts and the ground-water flow process. Open-File Report 00-92, U.S. Geological Survey, 2000.
20. E. Huber, D. Spivakovskaya, H. Lin, and A. Heemink. The parallel implementation of forward-reverse estimator. In P. Wesseling, E. Onate, and J. Periaux, editors, *ECCOMAS CFD*. TU Delft, The Netherlands, 2006.
21. B. Kaludercic. Parallelisation of the lagrangian model in a mixed eulerian-lagrangian cfd algorithm. *Journal of Parallel and Distributed Computing*, 64:277–284, 2004.
22. X. S. Li and J. W. Demmel. SuperLU-DIST: A scalable distributed-memory sparse direct solver for unsymmetric linear systems. *ACM Transactions on Mathematical Software (TOMS)*, 29(2):110–140, 2003.
23. P. Montarnal, A. Dimier, E. Deville, E. Adam, J. Gaombalet, A. Bengaouer, L. Loth, and C. Chavant. Coupling methodology within the software platform Alliances. In E. Oate M. Papadrakakis and B. Schrefler, editors, *Int. Conf. on Computational Methods for Coupled Problems in Science and Engineering COUPLED PROBLEMS 2005*, Barcelona, 2005. CIMNE.
24. K. Pruess, C. Oldenburg, and G. Moridis. Tough2 user's guide, version 2.0. report LBNL-43134, Lawrence Berkeley National Laboratory, 1999.
25. P. Salandin and V. Fiorotto. Solute transport in highly heterogeneous aquifers. *Water Resources Research*, 34:949–961, 1998.
26. J. Wheeler. Ipars user's manual. Technical report, University of Texas at Austin, 2000.
27. D. Zhang and Z. Lu. An efficient high-order perturbation approach for flow in random porous media via Karhunen-Love and polynomial expansions. *Journal of Computational Physics*, 194:773–794, 2004.
28. C. Zheng and G. D. Bennett. *Applied Contaminant Transport Modeling; second edition*. John Wiley & Sons, New-York, 2002.
29. C. Zheng and P. Wang. MT3DMS: a modular three-dimensional multi-species model for simulation of advection, dispersion and chemical reactions of contaminants in groundwater systems: documentation and user's guide. Serdp-99-1, U.S. Army Engineer Research and Development Center, 1999.