3D-Parallel Simulation of Contaminant Diffusion in Waste Disposals

Alexandre Francisco[1], Bruno Pereira[1], and José Adilson de Castro[1]

[1] Universidade Federal Fluminense,
Avenida dos Trabalhadores, 420,
272525-125 Volta Redonda, Brazil
{afrancisco, brunoap, adilson}@metal.eeimvr.uff.br

Abstract. In this work we simulate the contaminant transport problem in three dimensions that takes place throughout soil of waste disposals. Such problem is modeled by a diffusion-dominated equation. The solution is addressed by using hybridized mixed finite elements for the spatial discretization of the equation. The resulting linear algebraic system is handled by an iterative domain decomposition procedure. This procedure is naturally parallelizable, and permits to implement an algorithm in distributed memory machines in order to save on CPU time. The numerical results from serial and parallel codes are in a good agreement with experimental results, and performance measures indicate that the parallelizable procedure is able to perform an efficient simulation.

Keywords: Parallelizable procedure; Mixed finite elements.

1 Introduction

Parallel simulation can be an efficient way to investigate contaminant transport throughout soil of waste disposals. In this paper we apply a parallelizable numerical procedure to solving the diffusive transport problem in such porous media. Hybridized mixed finite elements are used for the spatial discretization of the governing equation. The resultant linear algebraic problems from this discretization are accomplished by an iterative domain decomposition procedure [1].

This paper is an extension of the work devised in [2], in which the parallelizable procedure was applied to approximating the solution of 2D-diffusive problems. Now, simulation results using the parallel code developed for three-dimensional problems are presented. The procedure permits to implement a parallel processing in order to obtain efficiency. It is naturally parallelizable in machines with distributed memory and allocates small memory space. Once it does not require the resolution of large linear equation systems, it is rapid. And it is of simple implementation concerning its code development.

Numerical results from serial and parallel codes are compared with experimental data. Both the serial and the parallel codes are able to reproduce the transport processes, but the parallel is the most efficient in a computational point of view.

2 The Governing Equations

The contaminant transport is described by the partial differential equation that considers the mechanisms of advection, dispersion and chemical reaction. The chemical reaction is modeled as a sorption process. Sorption means the mass transfer between the contaminants dissolved in water and the contaminants sorbed in the porous medium. In this model, the hydrodynamic dispersion is set close to molecular diffusion, since mechanical dispersion is introduced through contaminant mass transfer between the aqueous and the solid domains [3].

Assuming that the advection process can be neglected, contaminant transport can be described by the following equation:

$$\frac{\partial}{\partial t}(\rho_W \varepsilon c_W) + \nabla \cdot (-\rho_W \varepsilon D_W \nabla c_W) = S\,, \tag{1}$$

where ρ_w is the water density, ε the porosity, c_w be the contaminant concentration in the water phase, D_w is the molecular diffusion coefficient in the water phase, S is the source/sink term.

The source/sink term is assigned to the mass transfer kinetic of the contaminant solute as a function of time. The mass transfer kinetic of the contaminant solute can be expressed by the following relation [4,5]:

$$S = -k\beta_e A_{SW} \rho_W \varepsilon \left(c_W - c_{eq}\right)^{\alpha} c_W\,, \tag{2}$$

where k is the kinetic constant, β_e is the mass transfer coefficient, A_{sw} is the contact area between the water and the porous medium, c_{eq} is the equilibrium concentration of the contaminant solute in the leakage within the granular soil, and α is an exponent of the equation.

3 Numerical Procedure

For approximating the parabolic problem given by Eq. (1), we employ an implicit scheme for the discretization in time along with hybridized mixed finite elements for the spatial discretization [6,7]. This technique is appropriate to obtain accurate diffusion flux computations. A domain decomposition procedure is applied towards the solution of the algebraic system resulting from the spatial discretization.

3.1 Time and Spatial Discretization

Let Δt denote the time step. Define $t^n = n\Delta t$, and we shall denote that the approximate solution for variables at times t^n by $\beta^n \cong \beta(t^n)$.

With an implicit scheme for the time discretization, we rewrite the Eq. (1) using a mixed formulation:

$$\rho_w \varepsilon \frac{c_w^{n+1} - c_w^n}{\Delta t} + \nabla \cdot \left(\vec{d}^{n+1} \right) = S^{n+1}, \tag{3}$$

$$\vec{d}^{n+1} = -\rho_w \varepsilon D_w \nabla c_w^{n+1}, \tag{4}$$

where $\vec{d}$ is the diffusive flux.

An approximate solution to the system (3) and (4) is achieved through a spatial discretization as follows. Let $\Omega = [0, Lx] \times [0, Ly] \times [0, Lz]$ be a domain with a boundary $\partial\Omega$, on which $\Gamma = \partial\Omega$, $\Gamma_j = \Gamma \cap \partial\Omega_j$, $\Gamma_{jk} = \Gamma_{kj} = \partial\Omega_j \cap \partial\Omega_k$. Set $H = \{hx, hy, hz\}$, where $hx = Lx/Nx$, $hy = Ly/Ny$ and $hz = Lz/Nz$. Then, let $\{\Omega_j, j = 1,...,N\}$ be a partition of Ω, where $N = Nx \times Ny \times Nz$.

For simplicity in presentation, we shall treat the case in which $\{\Omega_j\}$ is a partition of Ω into individual elements used to discretize the system (3) and (4). In addition to requiring that $\{c_w^{n+1}, \vec{d}^{n+1}\}_j$ be a solution in the element Ω_j, it is necessary to impose the consistency conditions

$$c_{w_j}^{n+1} = c_{w_k}^{n+1}, \quad on \quad \Gamma_{jk} \tag{5}$$

and

$$\vec{d}_j^{n+1} \cdot \vec{n}_j + \vec{d}_k^{n+1} \cdot \vec{n}_k = 0, \quad on \quad \Gamma_{jk} \tag{6}$$

where $\vec{n}_j$ is the unit outer vector normal to Ω_j.

3.2 Mixed Finite Elements and Iterative Procedure

Let $V_j = H(div, \Omega_j)$ and $W_j = L^2(\Omega_j)$ for $j = 1,...,N$. The weak formulation of system (3) and (4) with the domain decomposed according to the discussion above is given

by seeking $\{c_w^{n+1}, \vec{d}^{\,n+1}\}_j \in W_j \times V_j$, $j = 1,...,N$, such that (after an integration by parts step),

$$(\rho\varepsilon \frac{c_w^{n+1} - c_w^n}{\Delta t}, w)_{\Omega_j} + (\nabla \cdot \vec{d}^{\,n+1}, w)_{\Omega_j} = (S^{n+1}, w)_{\Omega_j} \tag{7}$$

and

$$(\frac{1}{\rho\varepsilon D_w} \vec{d}^{\,n+1}, \vec{v})_{\Omega_j} + (c_w^{n+1}, \nabla \cdot \vec{v})_{\Omega_j} - \sum_k \left\langle c_w^{n+1}, \vec{v} \cdot \vec{n} \right\rangle_{\Gamma_{jk}} = 0 \tag{8}$$

where $\vec{v} \in V_j$ and $w \in W_j$.

Let us introduce Lagrange multipliers as contaminant concentrations on the edges $\{\Gamma_{jk}\}$ and replace Eq. (5) and (6) by equivalent Robin transmission boundary conditions. Thus, consistency conditions for the contaminant concentration will be given by

$$-\chi \vec{d}_j^{\,n+1} \cdot \vec{n}_j + l_{jk}^{n+1} = \chi \vec{d}_k^{\,n+1} \cdot \vec{n}_k + l_{kj}^{n+1}, \quad on \quad \Gamma_{jk} \tag{9}$$

where χ is a positive function on $\cup\Gamma_{jk}$, and l_{jk} is the Lagrange multiplier defined on Γ_{jk}, as seen from Ω_j.

We shall consider lowest index Raviart-Thomas spaces for the spatial discretization of the system (3) and (4). The element Ω_j will be taken to be a cube of side length $hx = hy = hz = h$. For simplicity in the presentation, we suppress the subscript j. Then, the system of equations in discretized form can be written as

$$\frac{\rho_w \varepsilon}{\Delta t} c_w^{n+1} + \frac{1}{h} \sum_\beta d_\beta^{n+1} = S^{n+1} + \frac{\rho_w \varepsilon}{\Delta t} c_w^n \tag{10}$$

and

$$d_\beta^{n+1} = -2 \frac{\rho_w \varepsilon D_w}{h} \left(l_\beta^{n+1} - c_w^{n+1} \right) \tag{11}$$

with the consistency conditions written as

$$-\chi d_{\beta}^{n+1} + l_{\beta}^{n+1} = \chi \tilde{d}_{\beta'}^{n+1} + \tilde{l}_{\beta'}^{n+1}, \quad on \quad \Gamma_{\beta} \tag{12}$$

where β denotes the six edges of a cubic element, with $\beta = L, R, U, D, F, B$. We introduce the superscript "~" to denote variables of adjacent elements and β' denote the corresponding edge of the adjacent element under consideration.

The resulting linear algebraic system (10), (11) and (12) consists of 13 variables $\{c_w^{n+1}, d_{\beta}^{n+1}, l_{\beta}^{n+1}\}$ and 13 equations for each element Ω_j.We use an iterative domain decomposition procedure in order to localize the calculations to problems over smaller domains than Ω. Here, it is feasible to localize to each Ω_j. The Eq. (12) is used in the above system to express all Lagrange multipliers at the new iteration level in terms of the Lagrange multipliers and diffusive fluxes of the adjacent elements at the previous iteration level. Then the Eq. (11) takes the form

$$d_{\beta}^{n+1} = \frac{\xi}{1+\chi\xi} c_w^{n+1} - \frac{\xi}{1+\chi\xi}\left(\chi \tilde{d}_{\beta'}^{n} + \tilde{l}_{\beta'}^{n}\right) \tag{13}$$

where $\xi = 2\dfrac{\rho_w \varepsilon D_w}{h}$. We refer to [8] for details about this iterative procedure.

Substituting Eq. (13) into Eq. (10), we have

$$\frac{\rho_w \varepsilon}{\Delta t} c_w^{n+1} + \frac{1}{h}\sum_{\beta} \frac{\xi}{1+\chi\xi} c_w^{n+1} = S^{n+1} + \frac{\rho_w \varepsilon}{\Delta t} c_w^{n}$$

$$+\frac{1}{h}\sum_{\beta} \frac{\xi}{1+\chi\xi}\left(\chi \tilde{d}_{\beta'}^{n} + \tilde{l}_{\beta'}^{n}\right) \tag{14}$$

Considering a linearization procedure for computing the mass transfer rate, the source/sink term is evaluated by the following scheme:

$$S^{n+1} = -k\beta_e A_{sw} \rho_w \varepsilon \left(E c_w^{n+1} - c_{eq}\right)^{\alpha} c_w^{n+1} \tag{15}$$

where an extrapolation operator is defined as follows:

$$E c_w^{n+1} = c_w^{n}, \quad n \geq 0 \tag{16}$$

4 Parallel Implementation

The parallel technique applied in this work, previously proposed by Vatsa and co-workers [9], establishes the distribution of computational load equally among all nodes belonging to the computational grid. Particular care was taken during the parallelization process concerning the domain division algorithm implementation and the parallel routines to keep the original code structure.

In order to achieve a good load balance distribution, an algorithm was implemented by decomposing the physical domain into subdomains (which contain several elements) which are assigned to distinct processors. Each processor allocates memory for the elements contained in its subdomain and for a buffer zone consisting of one layer of elements outside the subdomain. Data on the boundary of a subdomain is sent (received) to (from) neighboring subdomain using the buffer zones to perform the data transfer. Thus, local problems in each subdomain are solved at new levels, while quantities related to neighboring subdomains are evaluated at old levels.

This technique assures that each computing node solves parts of physical domain individually. Besides the gain in computational speed, the size of the problem or the number of maximum control volumes can be increased as far as the number of processors in the computational grid is linearly increased. The generic physical sub-domain division is presented in Figure 1, illustrating the data transmission pattern.

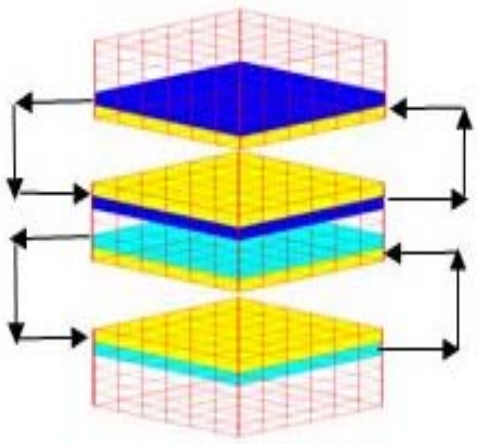

Fig. 1. Physical subdomain division and data transmission pattern.

The computer code is written in the FORTRAN language and the library adopted is the public domain version of MPI (Message Passing Interface). The communication technique adopted among computers was the blocking and the standard mode.

5 Numerical Simulation

Our numerical experiments were performed in a physical domain with impermeable boundary conditions. A cubic domain has 0.1 m × 0.1 m × 0.1 m discretized by $50 \times 50 \times 50$ computational grid. We ran the code using one, 2, 4, 6, 8, 10, 12, 14, and 16 processors on CPU Pentium IV workstation system. We considered the time step equal to $\Delta t = 15$ s. The following data were held fixed in our experiments: porosity $\varepsilon = 0.76$, molecular diffusion coefficient in the water $D_w = 6.342 \times 10^{-10}$ m^2/s, kinetic constant $k = 2.5 \times 10^{-2}$, equilibrium concentration $c_{eq} = 7.544 \times 10^{-1}$ kg/m^3, exponent number $\alpha = 1.75$.

5.1 One-Dimensional Problem

The initial condition was specified as follows. NH_4^+ concentrations of 9.2×10^{-2} kg/m^3 and 1.815 kg/m^3 were assigned on [0.0 m, 0.05 m] and [0.05 m, 0.1 m], respectively along the depth direction. Numerical results using the serial code and the parallel code using two processors are presented in Figure 2, at 72 hours of simulation. A concentration profile can be observed as a consequence of the NH_4^+ contaminant transport that takes place in the cell. We can note that numerical results are in good agreement with the experimental data.

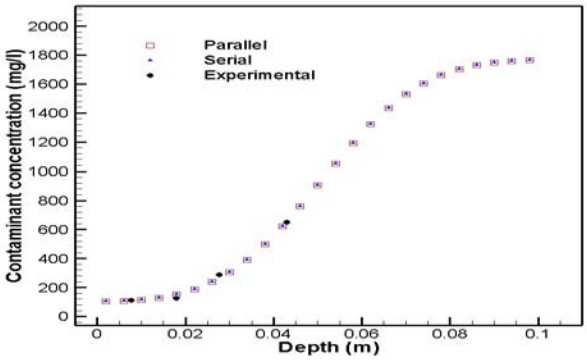

Fig. 2. Concentration profiles by using the serial and parallel codes.

In Figure 3 we present the CPU time needed to run the serial and parallel codes at 24, 48 and 72 hours of simulation. We can see that the decrease in time is not so much as expected, because some time is wasted by the communication among processors in parallel simulations.

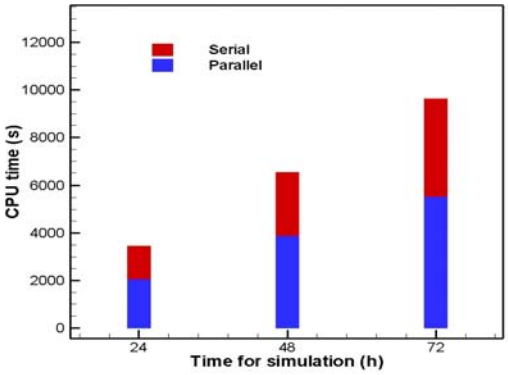

Fig. 3. CPU time for the serial and parallel codes.

5.2 Three-Dimensional Problem

We illustrate the diffusive transport in three space dimensions by using only the parallel code. The initial condition is depicted in Figure 4: NH_4^+ concentration of 1.815 kg/m^3 is assigned in the red box and 9.2×10^{-2} kg/m^3 elsewhere. Concentration surfaces at 72 hours are also presented in this picture. The numerical result indicates that the parallel code is able to simulate properly the diffusive problem.

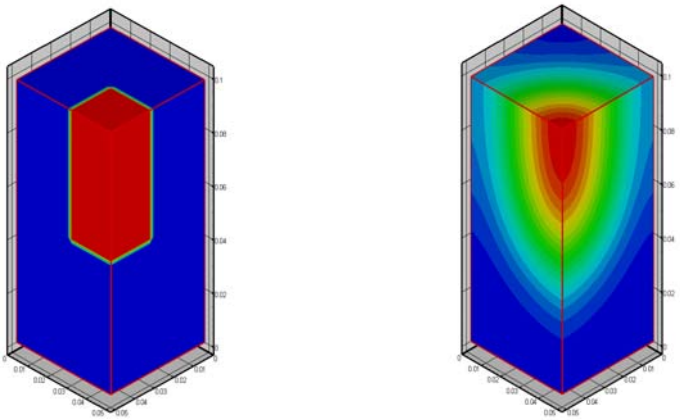

Fig. 4. Concentration surfaces at initial condition (left) and at 72 hours (right).

As a result, the total time needed to perform the simulation is almost 800 seconds for a single workstation, while 100 seconds is required if 16 processors are used. The corresponding measured speedup (the ratio of the time spent by a single workstation to perform a task to the time spent by N processors of the system to perform the same

task is plotted against N) and the linear speedup appear in Figure 5. Note that the speedup for the parallel code does not increase from 12 processors on.

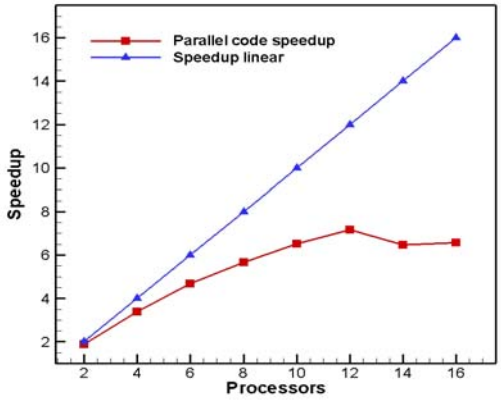

Fig. 5. Speedup curves.

5 Conclusions

A 3-D parallel code has been developed to solve accurate and efficiently the contaminant transport problem in experimental cells, considering diffusion and sorption processes as the environmental driven forces.

The iterative procedure permits naturally to implement an algorithm in distributed memory machines in order to save on execution time. The parallel code implemented for solving the diffusion problem has presented advantages in relation to the serial code. Thus, the parallel code has proved to be an efficient way to simulate the contaminant transport in experimental cells.

References

1. Douglas, J., Paes Leme, P.J., Roberts, J.E., and Wang, J.: A parallel iterative procedure applicable to the approximate solution of second order partial differential equations by mixed finite element methods. Numerical Mathematics 65, pp. 95, (1993)
2. Francisco, A. S., Pereira, Bruno, and De Castro, J. A.: A parallelizable procedure for contaminant diffusion in waste disposal. Transactions in Engineering Sciences 56, 163-172, (2007)
3. Zheng, C., and Wang, P. P.: MT3DMS: A Modular Three-Dimensional Multispecies Transport Model fo Simulation of Advection, Dispersion, and Chemical Reactions of Contaminants in Groundwater Systems; Documentation and User's Guide. Technical report SERDP-99-1, University of Alabama (1999)
4. Poirier, D.R., and Geiger, G.H.: Transport Phenomena in Materials. Ed. TMS (1994)

5. Kawasaki, N., Kinoshita, H., Oue, T., Nakamura, T., and Tanada, S.: Study on Adsorption Kinetic of Aromatic Hydrocarbons onto Activated Carbon in Gaseous Flow. Journal of Colloid and Interface Science 275, 40-43 (2004)
6. Douglas, J., Pereira, F., and Yeh, L.M.: A parallelizable characteristic scheme for two phase flow I: Single porosity models. Computational and Applied Mathematics 14, 73-96 (1995)
7. Douglas, J., Furtado, F. and Pereira, F.: Parallel methods for immiscible displacement in porous media. Wuhan University Journal of Natural Sciences 1, 502-507 (1996)
8. Douglas, J., Furtado, F., and Pereira, F.: On the numerical simulation of waterflooding of heterogeneous petroleum reservoirs. Computational Geosciences 1(2), 155-190 (1997)
9. Vatsa, V. N., and Hammond, D., P.: Viscow flow computations for complex geometries on parallel computers, Advances in Engineering Software, 29, 337-343 (1998)