



Parallel Processing of Three-Dimensional Field-Scale Reactive Transport Applications

Mohammed Adil Sbai

► To cite this version:

Mohammed Adil Sbai. Parallel Processing of Three-Dimensional Field-Scale Reactive Transport Applications. XVI International Conference on Computational Methods in Water Resources (CMWR XVI), Jun 2006, Copenhagen, Denmark. <http://proceedings.cmwr-xvi.org/contributionDisplay.py?contribId=296&sessionId=12&confId=a051>. hal-00589865

HAL Id: hal-00589865

<https://brgm.hal.science/hal-00589865>

Submitted on 2 May 2011

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

PARALLEL PROCESSING OF THREE-DIMENSIONAL FIELD-SCALE REACTIVE TRANSPORT APPLICATIONS

M. A. SBAI¹

¹BRGM (French Geological Survey), Water Department, Groundwater and Geochemistry Modeling

3, Avenue C. Guillemin, BP 6009, 45060 Orléans Cedex 2, France.

e-mail: a.sbai@brgm.fr

ABSTRACT

In order to achieve field-scale reactive transport simulations in a reasonable CPU time, a parallel computational model is developed for RTAFF a fully three dimensional reactive transport simulator. The MPICH portable implementation of the message passing interface standard is used for communication among nodes in a Beowulf cluster of PC's. Parallel computations are carried out in different subdomains mapped to each processor in the Local Area Network during the advection and dispersion steps of the time-split procedure. The domain decomposition method is implemented using row-wise and column-wise overlapping subdomains simultaneously for the mixed hybrid finite element kernel. Computations of the reaction step are performed using a newly developed load balancing algorithm. This algorithm maps cells into processors based on global convergence history analysis from the previous time step. The efficiency of this approach is applied on a demonstrative field scale example showing that the load balancing algorithm efficiently localizes and redistributes reaction fronts cells among the processors. Furthermore, this application shows the scalability of the overall parallel algorithm and the achievement of several-orders-of-magnitude speedup among a range of processors. Modeling results provide new insights into detailed subsurface reactive-transport processes which could be taken simultaneously by scientists and engineers in either research or field-scale applications.

1. INTRODUCTION

The shape and scope of multi-component reactive transport models changed dramatically over the past decade. Advances in several branches of subsurface hydrology, aqueous and surface geochemistry has enabled an increased sophistication in modeling capabilities. Therefore, modern computer codes should offer prospective possibilities for modeling field-scale multi-component flow and transport coupled with complex and heterogeneous geochemical reactions. However, simply feeding many chemical reactions from a large geochemical database into large scale models is beyond the capabilities of currently available single processor workstations or PC's. Available parallel computing methodologies and ressources are therefore the only relevant means to achieve such simulations in a

Date: March 13, 2006.

reasonable timeframe. When it seems that the path to a parallel computational model is a natural and straightforward approach, designing and efficiently implementing a scalable parallel reactive transport model is a non trivial task. Indeed, in the framework of an operator time-split technique (SNIA or SIA approaches [1]) a rough distribution of the geochemistry workload over the entire available processors will not yield an impressive parallel speedup. Pioneering work by [2] showed examples of dynamic computational roughness typical to heterogeneous geochemical transport. Therefore, innovative load balancing algorithms have to be designed for the distributed geochemical calculations task. At the other hand, partitioning algorithms and preconditioning of high performance linear solvers for unstructured meshes are not obvious tasks. A number of available scientific toolkits for scalable parallel programming [3] provide a management framework for the inter-communication and data partitioning details. However, such approach was not suited for the work presented in this paper, because our primary focus was on speeding up the computational geochemistry part. Indeed, previous experience on some selected practical problems showed that more than 90% simulation time was exclusively devoted to mixed equilibrium kinetics calculations. Parallel domain decomposition of the flow and transport solvers, although functional at the time of this writing, is implemented only on *ijk* structured grids.

Adoption of parallel computing technology brings advantages and constraints on its own right. The most limiting being hardware availability for a user's community. Luckily, modern computer technology allows to setup clusters of PC's from cheap commodity hardware delivering massive computing power. The most popular distributed architecture falling in this category are Beowulf clusters. These are based on open-source software packaged with currently available distributions of Linux operating systems. Thus, by targetting such architectures the computer package will run unmodified for systems accessible to many modelers, as well as in highly priced shared memory workstations available in a number of sites at the national level.

The primary objective of this work is to provide parallel numerical technology allowing us to perform reactive transport applications with several tens thousands cells ($\approx 50,000$), more than a dozen of master chemical species, and many thousands of time steps ($\approx 10,000$). This corresponds hopefully to a total CPU time reduction from a couple of days computing time to few hours. Next sections will describe the steps taken in our initial efforts on model development, evaluation, and testing on the basis of a three-dimensional model for deep acid gas (a mixture of H_2S and CO_2) injection in a heterogeneous carbonated reservoir.

2. PARTAFF MODEL DEVELOPMENT

The computer program PARTAFF (**P**arallel **R**eactive **T**ransport and **F**luid **F**low) simulates multi-component, reactive solute transport in three-dimensional saturated aquifers and single phase reservoirs. It can be viewed as a multi-processor extension of RTAFF simulator [4]. It is based on a modular structure including a simulation coordinator module, a fully implicit mixed hybrid finite element module for computing the flow in quadrilateral and hexahedral structured meshes, and an explicit *high-resolution* finite volume transport module. Meanwhile, geochemical calculations are based on a mixed formulation which

couples a Gibbs free energy minimizer for constrained thermodynamic equilibrium equations with several integration methods for kinetically controlled reactions.

Numerical solutions for pressure heads, *Darcy's* velocities, concentrations or activities of total dissolved components, mineral masses fractions, activities of user-selected secondary species, and rates of kinetically controlled reactions are obtained using a sequential approach. A four-stage advective-dispersive-reactive (ADR) operator-splitting algorithm [4] is used at each time step to decouple reaction and transport parts. Recently, a two-step time-split algorithm based on SIA[1] approach has been also incorporated in the simulator to minimize time-splitting errors. More details on the program formulation and validations should be found elsewhere [5].

2.1. Hardware and software requirements. PARTAFF code use the portable implementation of the message passing interface standard provided by the MPICH project [6]. We used only 29 routines from the 217 forming the whole library. It is a single program multiple data model (SPMD) that creates a fixed number of parallel tasks at the beginning of the program execution. Thus, a copy of the simulator is executed in each processor, while the data is being distributed over all available processors. This methodology does not allow for a smooth parallelization effort as provided by other standards (such as OpenMP [7]) which may not pay-off for porting many existing models. This is not the case for PARTAFF which has been designed from the assets as a parallel and multiple platform code.

Model evaluation and testing presented in this paper are performed on a Beowulf cluster of 12 dual-processor nodes with Intel Xeon processors running at 3.06 GHz and 2 GB local memory. All nodes use the Red Hat Linux operating system. The code was compiled with Portland Group high-performance suite of Fortran 90 and C/C++ compilers.

2.2. Parallel algorithms.

2.2.1. Domain decomposition (DD) of flow and transport computations. The simulation is carried out for 3D overlapping domains. The use of DD with overlapping [8] layers have been recognized as being able to accelerate the convergence of the parallel computation. Hence, in this study the DD domain is decomposed into a number of overlapping subtasks, the computation of each subtask is carried out in one processor. In this study the domain is decomposed into a number of horizontal, transversal and vertical overlapping slabs using task decomposition as shown in Figure 1. The vertical decomposition is only allowed if no production or injection wells exist in the computational domain.

The governing equations are solved for each subtask with appropriate boundary conditions. The solution process continues until the parallel preconditioned conjugate gradient iteration converge. Primary variables have to communicate between the processors at ghost points using blocking communication and the new values of primary variables are used in each subtask.

2.2.2. Load balancing of geochemical computations (The Heavy Light Hierarchically Randomized Splitting Algorithm). A successful load balancing algorithm in this context should capture dynamic physics of the underlying problem. It is expected that the geochemistry workload will be highly dependent upon positions in space and time. As the advective and dispersive waves travel over the computational domain, zones of different behavior

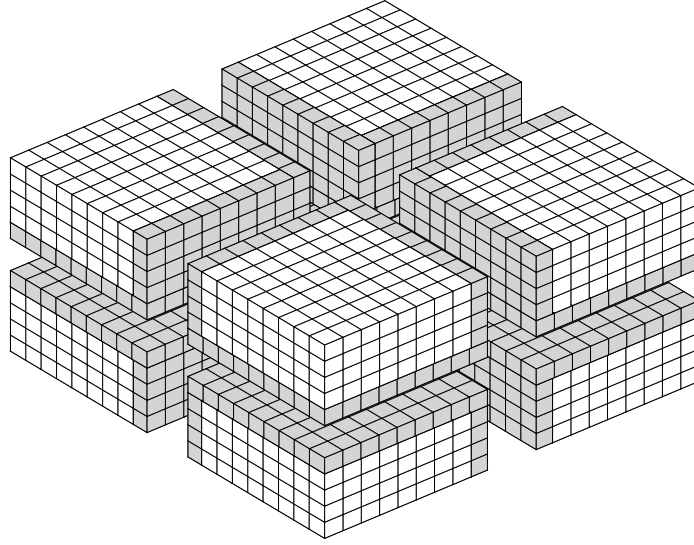


FIGURE 1. Example of a 3D mesh partitioned into eight (2x2x2) logical sub-domain processors. Ghost cells of each sub-domain (in grey shaded color) enable communications during preconditioned linear solver iterations.

will develop throughout. Cells located in zones of nearly constant primary dependent variables [9] (total component concentrations) will need much less time to converge since the expected changes are minor. However, cells crossed by the traveling waves will exhibit the most significant workload. The steps involved in the developed algorithm for this task are given as follows:

- (1) Estimate the required work for each grid-block assigned to a Beowulf node at the current time step. We assume that it equals the global geochemistry work calculated in the previous time step. A list of inactive cells undergoing minor total concentrations, C_k , change such that:

$$\max_{k=1}^{N_c} |\Delta C_k^{m-1}| < \frac{\varepsilon}{2} \quad (1)$$

is updated in this step. All cells in this list do not contribute to the estimated work for processors, and are consequently excluded from the following treatments. In Equation 1, m is the time stepping level, k is the chemical component index, N_c is total number of components, ΔC_k^{m-1} is the change of concentration of the k^{th} component at the previous time level, and ε is the effective zero concentration (i.e. $\varepsilon = 10^{-30}$ M).

- (2) Build a randomized spectral workload in Log_{10} scale that would equilibrate the global work among the processors. This is done by constructing several subgroups of cells ranging from the most lighter to the most heavier, and assigning an average workload for each subgroup. Then, groups of cells are sent (through non blocking point to point MPICH calls) from heavy to light (listening) nodes according to the new average workload distribution.

- (3) Perform parallel batch geochemical calculations for all nodes. First, local cells to each processor will be processed. Next, non local cells buffers moved in the previous step are processed. Finally, lighter nodes return newly computed buffers to sending processors to ensure proper updating with regard to the succeeding local transport step of the global time-split procedure.

This algorithm does not only try to equilibrate the workload between the available processors. It also eliminates very light cells needing fewer iterations counts and satisfying equation 1 from the computational process. For typical applications, those *noisy* cells increase the parallel overhead, although not dramatically, and could be safely ignored letting the algorithm concentrating on space regions with noticeable concentrations changes. This improves the coarsening of the parallel algorithm, minimizes communication calls, and speeds up the cells classification process.

3. MODEL TESTING AND PERFORMANCE RESULTS

3.1. Description of the test case problem. The geochemistry behind this test case problem is taken from the study by [10] guided in the French Geological Survey. Its main purpose was to assess the chemical disequilibrium and sensitivities associated with heterogeneous kinetic rate laws typical for acid gas injection operations in a depleted carbonated reservoir. Batch and one-dimensional simulations were performed at the laboratory core scale ($\approx 0.1\text{m}$ length). Herein a larger scale model is constructed to study the plume displacement mechanisms and the impact on rock mineralogy at the injector vicinity and farther away as well. The three dimensional model domain is a $1000\text{m} \times 1000\text{m} \times 100\text{m}$ box ($101 \times 101 \times 5$ uniform grid cells in the x, y and z directions respectively) where the mean vertical level is 1500m deep. A log-normally stochastic hydraulic conductivity field is constructed using the turning bands algorithm by [11], with an average of $2 \times 10^{-2} \text{ m/day}$, dimensionless variance of 1.0, and correlation lengths equal to 5 times the grid cells size in the three principle dimensions. The injection flow rate is assumed to be continuous for the first 20 years of the total 50 years simulation period. The well is centered in the study area and is assumed to have a total penetration along the reservoir thickness.

3.2. Evaluation of model performance. A methodological assessment of any parallel model performance involve a number of metrics. Common metrics are program relative speedup and relative efficiency defined respectively as [12]:

$$S_n = \frac{T_1}{T_n} \quad (2)$$

$$E_n(\%) = 100 * \frac{S_n}{n} \quad (3)$$

where T_1 and T_n are run times with one and n processors, respectively. S_n , the relative speedup with n processors, is a practical measure of the reduction in time achieved when increasing the number of processors. It is also as the number of apparent processors to the parallel computational engine. It is only limited by the number of uniprocessor chunks in the program and by the communication workload. E_n , the relative efficiency with n

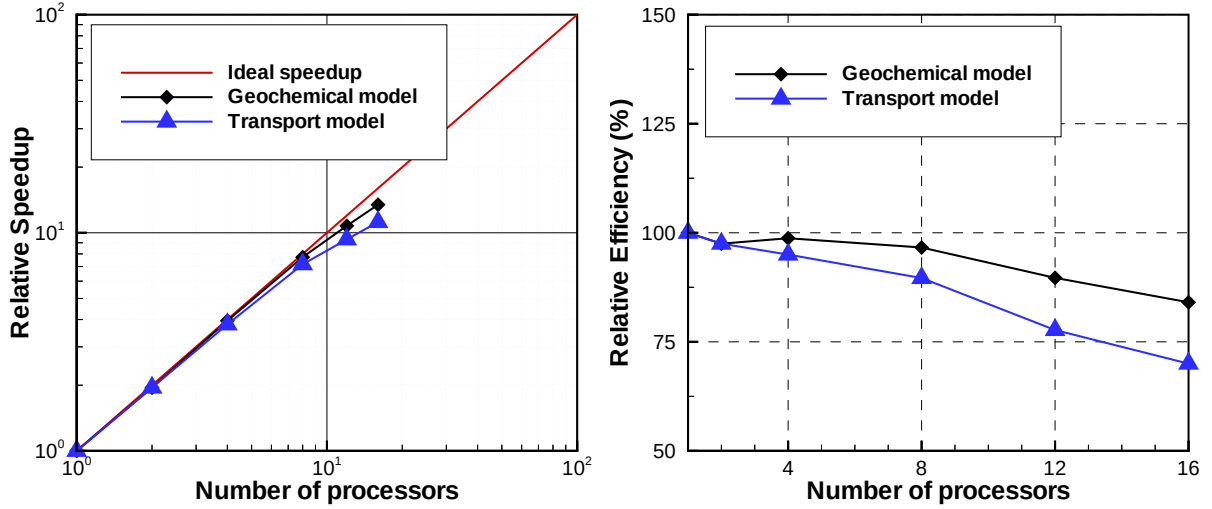


FIGURE 2. Relative speedup and relative efficiency of parallel transport and geochemical calculations for the test case problem.

processors, measures to which extent the parallel algorithm is effective during program execution.

Total execution time of the program is splitted into (i) CPU time in flow solver, (ii) CPU time in transport solver, (iii) CPU time in geochemical calculations, (iv) idle time, (v) communication time, and (vi) load balancing and parallel overhead time. Next, we will be interested only on relative speedup and relative efficiency associated to the transport and geochemical reaction tasks of the whole simulation. The load balancing algorithm takes only 0.3% of the total simulation time indicating that the geochemical load balancing algorithm is a coarse grain parallel task with a large workload to communication times ratio.

Figure 2 illustrates the relative speedup and relative efficiency of the batch geochemical engine and solute transport models respectively. Strong testing of the problem scalability is performed as a function of increasing powers of 2 number of processors up to 16 processors limit.

Clearly, it is shown that both parts of the simulation scale very well on the considered range of processors. The speedup is near to optimal up to eight processors, but maintains an important speedup of 13.45 for the 16 processor simulation. However, the transport speedup decreases more rapidly than the geochemical speedup. This is due to the increased communication overhead but also to the good scalability of the geochemistry workload making the transport speedup less apparent. However, this problem becomes of minor impact when much less CPU time is spent in the transport engine. For the range of processors taken into account the geochemical model efficiency remains always higher than 81%. The efficiency of the transport decreases much rapidly, even if it exceeds 70%. It is expected to be much less optimal for a high number of processors, deserving that much research must be guided for solving higher resolution problems on massively parallel computers in the future.

4. SUMMARY

This work presents a parallel simulation technology for multi-dimensional reactive transport applications. The technique involves a unified and portable set of parallel algorithm tools for saturated groundwater flow, solute transport, and geochemical calculations using MPICH message passing interface standard. This provides a simulation framework for systems ranging from multi-processor workstations, to cluster of PC's or workstations, and modern mainframe supercomputers. To demonstrate the relevance of this method, a test example for acid gas injection in a depleted carbonated reservoir is provided, and yields the following conclusions:

- The parallel algorithms are effective in reducing the total CPU time for field-scale reactive transport applications to an acceptable timeframe.
- The relative importance of efficiently rebalancing the geochemistry workload is demonstrated, and shows to be critical to the scalability of the overall parallel computational model.
- Presented parallel computing techniques can particularly enhance modeling capabilities, achieving several orders-of-magnitude speedup for field-scale and high-resolution applications.
- Much research must be conducted to evaluate the performance of the current algorithms in scenarios with increased geochemical complexity.

ACKNOWLEDGMENTS

The author would like to thank Dr Mohamed Azaroual for many enlightening discussions on modeling geochemical transport processes.

REFERENCES

1. C.I. Steefel and K.T.B. MacQuarrie, Approaches to modeling of reactive transport in porous media. In P.C. Lichtner, C.I. Steefel, and E.H. Oelkers, eds., *Reactive Transport in Porous Media, Reviews in Mineralogy*, 34. Mineralogical Society of America, Washington, D.C., p.83-130, 1996.
2. M.F. Wheeler, T. Arbogast, S. Bryant, and J. Eaton, Efficient parallel computation of spatially heterogeneous geochemical reactive transport, in Computational Methods in Water Resources XII, Vol. 1: *Computational Methods in Contamination and Remediation of Water Resources*, V. N. Burganos et al., eds., Computational Mechanics Publications, Southampton, U.K., 1998, pp. 453-460.
3. S. Balay, W.D. Gropp, L.C. McInnes and B.F. Smith, Efficient Management of Parallelism in Object Oriented Numerical Software Libraries, *Modern Software Tools in Scientific Computing*, E. Arge, A.M. Bruaset and H.P. Langtangen, Eds., Birkhauser Press, Boston, MA, 163-202, 1997.
4. A. Sbai and M. Azaroual, A numerical model for miscible displacement of multicomponent reactive species, in *Computational Methods in Water Resources XV*, Vol. 1: C.T. Miller et al., eds., Elsevier Sciences, Chapell-Hill, USA, 2004, pp. 850-860.
5. A. Sbai, M. Azaroual and A. Menjoz, Numerical solution of mixed thermodynamic equilibrium and kinetically controlled equations in RTAFF code. I- evaluation of coupled geochemical transport (in French). BRGM/RP-53001, 131pp., 2004.
6. W. Gropp, E. Lusk, N. Doss, and A. Skjellum, A high-performance, portable implementation of the MPI message passing interface standard. *Parallel Computing J.*, 22(6):789-828, 1996.
7. R. Chandra, R. Menon, L. Dagum, D. Kohr, D. Maydan, J. McDonald, *Parallel Programming in OpenMP*. Morgan Kaufmann, ISBN: 1-55860-671-8, 2001.

8. W.D. Gropp and D.E. Keyes, Domain decomposition methods in computational fluid dynamics. *Int. J. Numer. Methods in Fluids*, 14, 147-155, 1992.
9. G.T. Yeh and V.S. Tripathi, A critical evaluation of recent developments in hydrogeochemical transport models of reactive multichemical components. *Water Resources Research*, 25(1), 93-108, 1989.
10. M. Azaroual, J-C. Lachapagne, E. Caroli and I. Czernichowski-Lauriol, Thermo-kinetic modelling of mass-exchange mechanisms between water-rock-gas induced by acid gas injection in carbonate reservoirs. AAPG Annual Convention, Abstract Number: ID97039, June 19-22, Calgary, Canada, 2005.
11. A.F.B. Tompson, R. Ababou and L.W. Gelhar, Implementation of the Three-Dimensional Turning Bands Random Field Generator, *Water Resources Research*, 25(10), 2227-2243, 1989.
12. I.T. Foster, *Designing and Building Parallel Programs*. Addison-Wesley, Reading, MA, 1995.