Proceedings of the VIII Conference on
Logic and Computer Science LIRA '97
Novi Sad, Yugoslavia, September 1-4, 1997, pp. 3-8

General Parallel Algorithm to the Solution of the Geophysical Inversion Problem Applied to the Transputer System

Vladimir Filipović, Dušan Tošić,
Faculty of Mathematic, University of Belgrade, Studentski trg 16, 11000 Beograd
Dragan Urošević,
Institute of Mathematics, Knez Mihaila 35, 11000 Beograd
Jozef Kratica
Faculty of Machine Eng. University of Belgrade, 27. Marta 80, 11000 Beograd

Abstract. The Geophysical inversion problem is described and methods to the solution of this problem are mentioned. Some ways for parallelization of these methods are considered and a parallel algorithm is implemented on transputer system. To apply this parallel algorithm a special configuration of transputer system into hypercube parallel computer is made. The whole program (for the cube-manager and nodes of the hypercube) is written in programming language C for the transputers. Some test-examples for the measurement of the performances of applied parallel algorithm are presented.

Keywords: Parallel algorithm, tomography, geophysical inversion problem, transputer, hypercube.

1. Introduction

The problem of determining the characteristics of an underground region by using given measurements data is known as Geophysical Inversion Problem. The tomographic methods are often used in the solution of this problem. Those methods are based on the estimation of the P-wave velocity within some region of space through which the rays have passed. The distribution of velocity is used for characterizing the subsurface. We deal with the problem of velocity estimation from first-arrival traveltimes in a two-dimensional cross-hole geometry. The transmitters are located in one borehole, while receivers are in another borehole and (possibly) on the surface. The rectangular plane between two boreholes is the examined cross-hole region.

Standard tomographic procedures are based on discretization of the cross-hole region (see [1],[2],[3]). The rectangular between two boreholes is divided into a grid of cells and v(x,y) is assumed to be constant over the area covered by any one cell.

Traveltime is related to the velocity structure by the equation

$$t_i = \int_{R_i} \frac{ds}{v(x,y)} \tag{1}$$

where t_i is the traveltime of the ith ray, ds is the differential raypath length of the ith ray, $v(x,y)$ is the two-dimensional velocity function and R_i is the raypath trajectory of the ith ray. (Reciprocal velocity value $1/v(x,y)$ is often named as slowness and denoted by $u(x,y)$. Instead of the velocity distribution, we will look for the slowness distribution in the cross-hole area.) If the slowness area is discretized into N cells, then equation (1) can be approximated as:

$$t_i = \sum_{j=1}^{N} d_{ij} u_j \tag{2}$$

where d_{ij} is the length of the ith ray through the jth cell and u_j the unknown slowness in the jth cell. It is assumed here that d_{ij} equals zero in the cells not traversed by the ith ray. If we have M rays traveling through N cells, the traveltime equations can be written in matrix form as

$$\mathbf{t} = \mathbf{D}\mathbf{u} \tag{3}$$

where **t** is an $Mx1$ column vector of traveltimes, **u** is an $Nx1$ column vector of slowness and **D** is an MxN matrix of segment lengths through the cells (see [2]).

2. Outline of the models and techniques

Two main models are used in the application of tomographic methods. These are: the linear and curve-line models. In the linear model the straight raypaths between two boreholes are assumed, while in the curve-line model, the curve-line raypaths are assumed. The combinations of these models and their modifications are possible.

In both of those cases the following problems arise:

1. The data T cannot be measured exactly and the proposed model is only an approximation. This causes (3) to be an inconsistent system. ([2], [3])
2. The solution of (3) is not unique in general. (It is possible that the number of equations in (3) is: less, equal or greater than the number of unknowns.)
3. The matrix D in (3) is sparse and may often be too large.

With regard to 1, 2 and 3, the direct matrix-inversion or pseudo-inversion is not convenient for system (3). Because of that special iterative techniques are developed (see: [2] and [5]). The general schemata in applying those techniques are as fellows. An initial estimation of slowness u_{ini} is made and the vector of traveltimes t_{calc} is calculated by using (3). The difference between the observed and calculated times t_{obs} - t_{calc} is redistributed back along each raypath and the corrections of slowness are made by using some minimization criterion ([2],[3], [5]). Rays are traced again through the cross-hole region and the process is repeated. Iterative process can be described by:

$$u^{(q+1)} = P_N P_{N-1} \ldots P_1(u^{(q)})$$

or

$$u^{(q+1)} = \frac{1}{N}\sum_{i=1}^{N} P_i(u^{(q)}) \quad , \quad q=0,1,\ldots$$

where

$$P_i(u) = u + w\frac{t(i) - d^T(i)u}{d^T(i)d(i)} d(i)$$

($0 < \omega < 2$ - the relaxation parameter) and **d**(i) the *i*th row of the matrix **D**.

There are many variations of these general techniques (see [2],[3]).

3. Parallelisation of used techniques

The process of Image reconstruction is often related to considerable computation. The applying of parallel computers may be useful in this field. In [7] a parallel tomographic algorithm based on the linear model is proposed. It is recognized later that the proposed algorithm is independent of the used model. In this work we propose a modification of this algorithm as a general parallel scheme for the iterative techniques applied to the solution of large linear systems of equations and adopted for the transputer system.

Initially the algorithm in [7] was designed for Intel iPSC/1 hypercube parallel computers. The hypercube computers consist of a host processor, i.e., the cube manager and large number of identical processors (nodes) that work concurrently and each is provided with its own memory. If an hypercube has 2^n processors and each of them is interconnected with n neighbors, it is an n-dimensional hypercube. The communication between the concurrent processes is realized by message passing. Each node in an hypercube has an operating system kernel that provides: running processes within that node, sending and receiving messages and routing the messages that flow through the node ([4], [6]) .

Moreover, the transputer system is without a strong architecture, but with a very flexible one. It could be configured in different parallel architectures. To make a general hypercube configuration (with a changeable number of nodes) the additional programming work is necessary. By using additional functions we configured transputers into hypercube parallel computer with maximum 8 nodes. The following configuration file for transputer system is used:

```
1,tomo02_1,R0,0,2,3,5;
2,tomo02_2,R1, ,1,4,6;
3,tomo02_2,S4, ,4,1,7;
4,tomo02_2,S2, ,3,2,8;
5,tomo02_2,S1, ,6,7,1;
6,tomo02_2,R8, ,5,8,2;
7,tomo02_2,S3, ,8,5,3;
8,tomo02_2,S7, ,7,6,4;
```

In each line a transputer is defined by the following data: number of transputer, file name, reset signal and the numbers of transputers linked with the defined transputer. The first transputer has the role of cube-manager.

Unfortunately, the structure of T-800 transputers (in fact, the adapter for the software configuration of transputers) does not enable more nodes in a hypercube topology (see: [8] and [9]). The functions related to the configuration of the system are incorporated into our program. Our program starts with the reading of the configuration and other data. After that the computation related to the solution of linear system equations is begun.

4. Pseudocode of program

Denote with p the maximum number of available nodes in a hypercube. According to the concept of hypercube parallel computers we will describe the procedures (in fact, the pseudocodes of procedures) for both the nodes and the cube manager. The same procedure is used for all nodes, but a separate one for the cube manager.

Denote with T the number of transmitters and with R the number of receivers, The following solution is for the case $M = np = T*R$ ($n \in N$), where M is the total number of rays. This case is acceptable for practical application.

Let us introduce the additional notation:

eps - the accuracy of slowness and

max - maximal number of iterations.

A global design of our program is done by the following pseudocodes:

Node pseudocode

BEGIN
RECEIVE all data from the cube manager
including the configuration data
UPDATE $u^{(0)}$
SEND $u^{(0)}$ to cube manager.
END

Cube-manager pseudocode

BEGIN
READ Configuration of transputer system
WRITE Configuration of transputer system
READ
parameters of discretization
T, R, p, eps, max and u_{ini}.
SET k:=0; n:=T*R **DIV** p; $u^{(0)}:=u_{ini}$
REPEAT

```
    SET u(1):=0; u(2) := u(0)
    FOR i:= 1 TO n DO
       FOR j:=1 TO p DO
          SEND (to jth node)
             parameters of discretization
             data related to ((i-1)p+j)th ray,
             u(0)
       FOR j:=1 TO p DO
          RECEIVE (from jth node)
       COMPUTE u(1):=(u(1)+u(0))/p
       SET u(0) := u(1); k:=k+1
UNTIL ||u(1)-u(2)|| < eps OR k>max
IF   ||u(1)-u(2)|| < eps   THEN print u(1)
                           ELSE print message.
END
```

5. Implementation and examples

The main step in this pseudocode is **UPDATE** $u^{(0)}$. In the parallelizing of the Geophysical Inversion Problem this is the most complicated step and it depends on the method used for the solution of linear system equations. During the testing of this program, the step **UPDATE** $u^{(0)}$ was very simplified: we increased all components of $u^{(0)}$ by one. (For the organization of the parallel program the computation itself is not important because different norms could be used.) With this simplification the program has been implemented on transputer hypercube by using programming languages C.

Table 1

Dimension	Weight	Times (in sec) for 1 transputer	Times (in sec) for 8 transputers
35	1	0.00819	0.04448
	12	0.05606	0.09689
	100	0.74016	0.14835
	1000	8.00038	1.09945
400	1	0.09017	0.45958
	40	3.61645	0.92089
	100	8.28723	1.63002
	1000	90.22355	12.27577
2500	1	0.56550	2.84141
	100	56.53792	10.20633
	1000	564.96403	77.10771
4900	1	1.12512	5.59065
	100	103.63398	20.21971
	140	157.20038	26.15200
	1000	------------	153.36377

The performances of this implementation are measured for different dimensions of the problem (the number of equations) and for the different kinds of computations in nodes (weight). The weight is the number of repetitions of increasing all components of $\mathbf{u}^{(0)}$ by one. (For example, if the weight is 100, then in each node of hypercube the component of $u^{(0)}$ is increased by one 100 times.) In all the test-examples T=6 and R=6, i.e. the number of transmitters and receivers is constant. The characteristic results obtained for the 1 and 8 transputers are presented in the Table 1.

6. Conclusion

By analyzing the obtained results we conclude that the computation on 8 transputers may take more time then computation on 1 transputer if the computation in node is not complex (the first line in each row of the table). Moreover, if the computation in node is complex, the obtained results are very good. The best results are obtained if the number of equations in (3) is large and the computation in node is complex. In this case (which is only interesting for practices) the parallelizing of tomographic algorithms for the solution of Geophysical Inversion Problem s might be very successful.

References

[1] Bois P, La Porte M, Lavergne M, Thomas G, Essai de determination automatique des vitesses sismiques par mesures entre puits, *Geophysical prospecting*, **XIX** (1) (1971) 42-83.

[2] Dines K.A. and Lytle R.J, Computerized Geophysical Tomography, *Proc of IEEE* 67 (7) (1979) 1065-1073.

[3] Ivanson S, Seismic Borehole Tomography-Theory and Computational Methods, *Proc of IEEE* 74 (2) (1986) 328 - 338.

[4] Johnsson S.L. Communication Efficient Basic Linear Algebra Computations on Hypercube Architectures, *J. of Par. And Dist. Comp.* 4 (1987) 133-172.

[5] Peterson J.E, Paulsson B.N.P, McEvilly T.V, Applications of algebraic reconstruction techniques to crosshole seismic data, *Geophysics* **50** (10) (1985) 1566-1580.

[6] Saad Y. and Schultz M.H, Topological Properties of Hypercubes, *IEEE Tran. on Comp.* 37(7) (1988), 867-872.

[7] Tošić D, Geophysical Inversion problem and tomographic parallel algorithms, Proc. *International Conference on Technical Information*, Timisoara, Romania, **4** (1994) 140-148.

[8] Transputer development system, Inmos, Prentice Hall, New York,..., Tokyo, 1988.

[9] Transputer technical notes, Inmos, Prentice Hall, New York,..., Tokyo, 1989.