Transactions on Information and Communications Technologies vol 3, © 1993 WIT Press, www.witpress.com, ISSN 1743-3517

Symmetry and BEM on transputers

J.Lobry

Department of Electrical Engineering, Faculté Polytechnique de Mons, Bd. Dolez, 31, 7000 Mons, Belgium

ABSTRACT

Geometrical and material symmetries are found in many linear field problems. They can be taken into account thanks to a rationale called the Group Representation Theory. This method consists in reducing an original problem into a family of smaller ones, the overall solution is the sum of the sub-problems. In this paper, we propose the implementation of the Boundary Element Method on distributed memory computers for two-dimensional problems in this context. Different parallel strategies are analysed and compared. Experimental results are obtained on a Meiko Computing Surface with 16 T800 transputers.

INTRODUCTION

After recent experiences of Bryant et al. [1], Daoudi and Lobry [2,3] in the parallelization of the Boundary Element Method (BEM) on distributed memory computers, we investigate in problems that exhibit geometrical and material symmetries. Such problems are often met in engineering applications. When general excitations and complex symmetry are considered, the Group Representation Theory [4,5] is the only tool that allows to take geometrical symmetry into account. It consists in reducing an original problem to a set of subproblems, defined on a cell of symmetry of the domain under study. The global solution is obtained from superposition of the partial ones. Some papers have shown that large computational cost savings and reduction in memory volume are achievable with this too hardly used method; e.g. Bossavit [6], Bonnet [7], Allgower et al. [8], Lobry and Broche [9].

In this paper, we propose the exploitation of symmetry with the BEM for the 2-D Laplace's problem on transputers. The abelian case is considered for sake of clarity but the conclusions should not be much altered for the general

non-abelian case. The ring architecture is chosen because of its efficiency and we study three parallel algorithms for the problem. The first is the reference, it consists in solving on a single ring without any symmetry consideration. The last two take advantage of symmetry on one or several rings interconnected. We analyse and compare the execution times for the assembling and solution of the related linear system.

The algorithms have been implemented on a Meiko Computing Surface with 16 T800 transputers. Experimental timing results are presented and compared.

THE BEM FOR THE 2-D LAPLACE'S PROBLEM

Consider the Laplace's problem defined on a 2-D domain Ω for a scalar potential ϕ (Fig. 1a) :

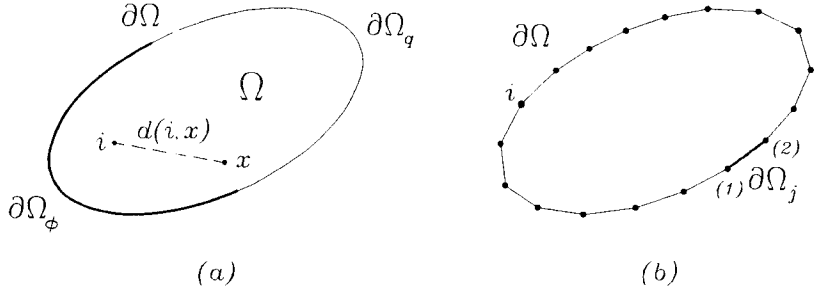

Figure 1. (a) 2-D Laplace's problem and (b) BEM

$$\nabla^2 \phi = 0 \quad in\ \Omega \tag{1}$$

with the boundary conditions on $\partial\Omega$:

$$\begin{aligned} \phi &= \phi_0 \quad on\ \partial\Omega_\phi \\ \frac{\partial\phi}{\partial n} &= 0 \quad on\ \partial\Omega_q \end{aligned} \tag{2}$$

The procedure that leads to a boundary integral formulation has been described by Brebbia [10], we briefly recall it for convenience.

A weighting function ("kernel") w_i is first defined for any point i of Ω so

that :

$$\nabla^2 w_i = -\delta_i \qquad hence : w_i(x) = \frac{1}{2\pi} \ln \frac{1}{d(i,x)}$$

where d(i,x) denotes the distance between the points i and x (fig. 1a).

From equation (1), the following integral statement may be derived :

$$\int_\Omega w_i \nabla^2 \phi \, d\Omega = 0$$

Applying a Green identity, we finally obtain the boundary integral formulation of the 2-D Laplace's equation :

$$c_i \phi_i = \oint_{\partial\Omega} \left(w_i \frac{\partial \phi}{\partial n} - \phi \frac{\partial w_i}{\partial n} \right) d\Gamma , \qquad \begin{aligned} c_i &= 1 \ if \ i \in \Omega - \partial\Omega \\ &= \frac{\alpha}{2\pi} \ if \ i \in \partial\Omega \end{aligned} \tag{3}$$

for all i in Ω (α is the internal angle in i).

In order to solve (3), one divide the boundary $\partial\Omega$ into n straight line elements $\partial\Omega_j$ (fig. 1b) on which the potential ϕ and its normal derivative (denoted q from now on) are defined by a linear interpolation. Hence the discretized expression :

$$c_i \phi_i + \sum_{j=0}^{n-1} \int_{\partial\Omega_j} \phi \frac{\partial w}{\partial n} \, d\Gamma = \sum_{j=0}^{n-1} \int_{\partial\Omega_j} w \frac{\partial \phi}{\partial n} \, d\Gamma \tag{4}$$

The integrals of (4) can be numerically or analytically calculated so that we obtain the form :

$$c_i \phi_i + \sum_{j=0}^{n-1} \left(h_{i,j}^{(1)} \phi_{j,1} + h_{i,j}^{(2)} \phi_{j,2} \right) = \sum_{j=0}^{n-1} \left(g_{i,j}^{(1)} q_{j,1} + g_{i,j}^{(2)} q_{j,2} \right) \tag{5}$$

where $\phi_{j,k}$ and $q_{j,k}$ (k=1,2) are the values of ϕ and q on the nodes 1 and 2 of the element $\partial\Omega_j$. $h_{i,j}^{(k)}$ and $g_{i,j}^{(k)}$ are coefficients coming from integration.

Writing (5) for all i of $\partial\Omega$ and substituting the boundary conditions of the problem, a $n \times n$ linear system is deduced :

$$S\,x = f \tag{6}$$

where the matrix S is dense and x is the vector of the unknown ϕ and q.

SYMMETRY IN FIELD PROBLEMS

In many field problems, geometrical and material symmetries are often met. Taking it into account is easy in linear regions when the source fields share some part of the symmetry in an obvious way. But the intuitive approach fails with complex symmetries and general source fields. Nevertheless, it is possible to take full advantage of symmetry by studying the problem on a cell of symmetry. The background is the Group Representation Theory that we briefly describe below; see e.g. Hamermesh [4] or Serre [5] for more details.

The Group Representation Theory - some elements

Let Ω be a spatial domain that presents some symmetry we describe by the isometries g (rotations and reflections) that leave Ω globally invariant. Those are transformations acting on points. They form a finite group G = {e, f, g,...} called *symmetry group* where a composition law is defined. If the law is commutative, the group G is said *abelian*. We shall restrict our presentation to this case. The number of elements in the group is the *order of the group* n_G.

Figure 2 illustrates this by the abelian cyclic group C_3.

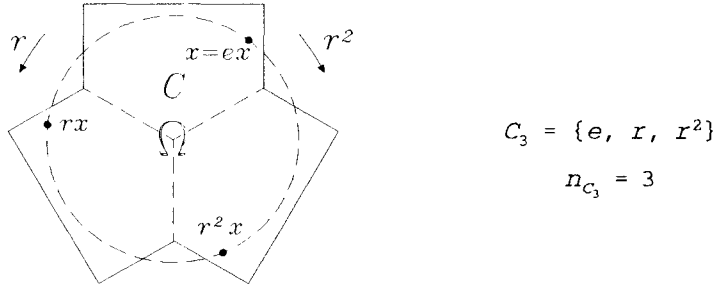

Figure 2. Example : the cyclic group C_3.

It is clear the subdomain C (*cell of symmetry*) regenerates Ω from the

symmetry operations :

$$\Omega = \bigcup_{g \in G} gC$$

In a similar way as the g's act on points, linear operators O_g acting on complex functions ϕ of a vector space V are defined :

$$(O_g \phi)(x) = \phi(g^{-1}x)$$

they are such that "the function ϕ is shifted by O_g as x is shifted by g".

These operators form a group G' isomorphic to G and the mapping O between the groups G and G' is called a *linear representation of the group G in the representation space V.* If we consider the group of matrices D(g) related to the operators O_g and a basis of V, we get a *matrix representation of G in V.*

A full theory about finite groups shows that any linear representation of a finite group can be decomposed in some sub-representations called *irreducible representations.* Their number and degree are well defined for a given group and finding all of them is an important point in group theory. In particular, for the abelian groups, it can be shown that the number of distinct irreducible representations is equal to the order of the group. Moreover, they are all of degree one (complex numbers).

For the C_3 group presented above, there are three irreducible representations (numbered ν), they are given by :

$$D^{(\nu)}(r^k) = e^{j2\nu k\pi/3} \quad \nu = 0,1,2 \text{ and } k = 0,1,2$$

We now present the *fundamental decomposition theorem.*

Let us define the following operators, called *projectors*, for each irreducible representation ν of a group G of order n_G (= 1 to n_G) :

$$P^{(\nu)} = \frac{1}{n_G} \sum_{g \in G} D^{(\nu)*}(g) O_g$$

where * denotes the complex conjugation.

The $P^{(\nu)}$'s transform any function ϕ of the representation space V into the functions $\phi^{(\nu)}$ so that :

$$\phi = \sum_{\nu=1}^{n_G} \phi^{(\nu)} \quad where \quad \phi^{(\nu)} = P^{(\nu)} \phi \tag{7}$$

and the $\phi^{(\nu)}$'s verify the *ν-symmetric* property :

$$O_g \, \phi^{(\nu)} = D^{(\nu)}(g) \; \phi^{(\nu)} \tag{8}$$

So, given any function ϕ of V, the operators $P^{(\nu)}$ decompose it into n_G functions $\phi^{(\nu)}$ that exhibit, from (8), some symmetry properties so that they can be regenerated by their only restriction on a symmetry cell of the domain.

How to apply those concepts to a field problem is easy to guess. Starting from an original problem described by a set of equations defined on a domain Ω with abelian symmetry, we have to determine a symmetry cell and to solve on it for each of the source fields derived from the irreducible representations. The overall solution is finally obtained from superposition.

BEM AND SYMMETRY

Consider now the domain Ω shown in figure 3. Its symmetry group is G of order n_G. The boundary $\partial\Omega$ presents the symmetry and we may choose a symmetry cell Γ so that :

$$\partial\Omega = \bigcup_{g \in G} g\Gamma$$

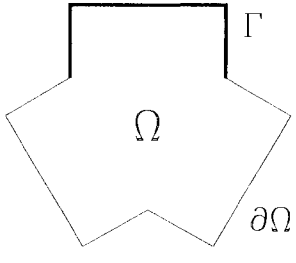

Figure 3.

Exploiting this property of the cell and denoting $a_{\partial\Omega}(\phi, w_i)$ the right-hand side

of relation (4), we may write the relationship :

$$\begin{aligned} a_{\partial\Omega}(\phi, w_i) &= \sum_{g\in G} a_{g^{-1}\Gamma}(\phi, w_i) \\ &= \sum_{g\in G} a_{\Gamma}(O_g\phi, O_g w_i) \end{aligned} \tag{9}$$

Applying the theory described in the previous section, we decompose the unknown function ϕ as in the definition (7). Then, we get the components for which the formulation (3) should be solved with the corresponding boundary conditions on Γ. Those conditions are calculated from (7) applied on conditions (2) and from the property (8).

Introducing each component in the formulation (3), substituting the integral identity (9) and using property (8), we obtain :

$$\begin{aligned} c_i\ \phi^{(\nu)}(i) &= \sum_{g\in G} a_{\Gamma}(O_g\phi^{(\nu)}, O_g w_i) \\ &= \sum_{g\in G} a_{\Gamma}(D^{(\nu)}(g)\,\phi^{(\nu)}, O_g w_i) \\ &= a_{\Gamma}(\phi^{(\nu)}, \sum_{g\in G} D^{(\nu)}(g)\, O_g w_i) \\ &= n_G\, a_{\Gamma}(\phi^{(\nu)}, w_i^{(\nu)^*}) \end{aligned}$$

Finally we have to solve the following n_G problems associated to the n_G irreducible representations $\nu = 1$ to n_G ("symmetrized kernels") :

$$c_i\ \phi^{(\nu)}(i) = n_G \int_{\Gamma} \left(w_i^{(\nu)^*} \frac{\partial\phi^{(\nu)}}{\partial n} - \phi^{(\nu)} \frac{\partial w_i^{(\nu)^*}}{\partial n} \right) d\Gamma \tag{10}$$

where :

$$w_i^{(\nu)}(x) = P^{(\nu)} w_i(x) = \frac{1}{n_G} \sum_{g\in G} D^{(\nu)^*}(g)\ \frac{1}{2\pi} \ln \frac{1}{d(i, g^{-1}x)} \qquad (i\in\Gamma).$$

We see that the original problem defined over the entire boundary $\partial\Omega$ has been replaced by *an equivalent set of subproblems* defined over the reduced open boundary Γ.

The discretization of (10) leads now to a set of linear systems of equations similar to (6). Once the subsystems are solved, the partial solutions have to be added as shown in equation (7).

Taking advantage of geometrical symmetry allows large economies in computational efforts with finite or boundary elements, since the computational cost grows much faster than the size of the problem considered. See e.g. [6,7,8,9]. But it is possible to go further in the reduction of the execution time by using *parallel processing*. This is the very purpose of our paper.

PARALLEL IMPLEMENTATION

Starting from our experience [2,3], we choose the ring architecture with memory distributed computers as the primitive topology for our problem. The solution method is the Gaussian elimination. The row wrapped interleaved storage analyzed in [3] is considered for the assembling and solving steps because of its efficiency and easiness of implementation.

Theoretical analysis of the computation and communication times are studied with the model of Saa [11]. The communication time of n consecutive data items between two neighbouring processors is assumed of the form $n\tau+\beta$ where β is the start-up time and τ is the time to transfer one data item. The elementary computation time is ω.

We first briefly recall the row-oriented algorithm without any symmetry consideration.

BEM 2-D on the ring architecture

We consider a ring composed of p identical processors numbered from P_0 to P_{p-1}. The linear system (6) is subdivised into p blocks each of n/p rows and we assign the block composed of the rows $i+kp$ for $0 \leq k \leq (n/p)-1$ to processor P_i. The theory clearly states that parallel system assembly is achieved without any communication, in a natural way. The execution time $T_{ass}(p)$ is then equal to the computation time :

$$T_{ass}(p) = O\left(\frac{\alpha n^2}{p}\omega\right) \tag{11}$$

The parallelization of the Gaussian elimination algorithm has to manage the row wrapped storage. In order to obtain a numerical stability, a partial pivoting strategy is needed. Hence the search of the maximum pivot element requires to interchange rows so that many communications are involved. The related algorithm has been described in [3] and the execution time is roughly :

$$T_{gs}(p) = O\left(\frac{2n^3}{3p}\omega + p(n\beta + n^2\frac{\tau}{2})\right) \qquad (12)$$

where the computation time term is $\propto n^3$ and the communication time is $\propto p$.

If we plot the global time $T_{tot}=T_{ass}+T_{gs}$ versus the number p of processors for a given size n, we find a curve presenting a minimum due to the combination of the 1/p and p terms. Moreover, the p for minimum increases as n increases.

Exploitation of symmetry

In order to exploit symmetry, two extreme types of implementation may be considered with the ring topology.

The first consists in adapting the ring algorithm described above to symmetry considerations : for an abelian group of order n_G, the n_G sub-problems are solved successively on the ring.

The second idea exploits the independance of the subproblems by solving each of them on separated rings, the global solution is then formed in a single extra processor. From this point of vue, the decomposition of an original problem into several subproblems may also be regarded as a possible parallel strategy for managing symmetry.

Those two opposite situations may be generalized by the following. Let p be the number of processors, they can be arranged by partitioning them in k rings composed of p/k processors on which n_G/k subproblems are to be solved. In this context, k should be a divisor of p and n_G for an optimum load balancing (fig. 4).

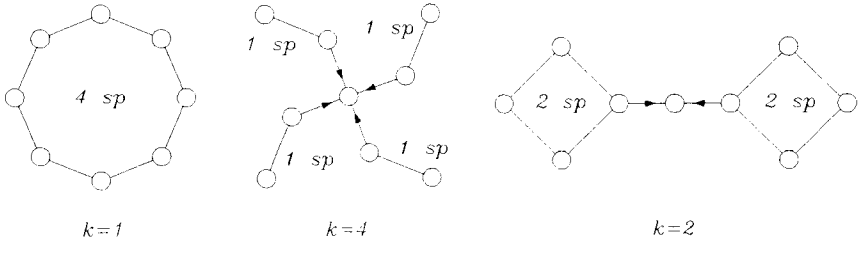

Figure 4. k-ring architectures, example with $n_G=4$, p=8 (sp : subproblem).

In order to minimize the time cost required for the subsystems assembly, a part of our algorithm is common to the computation of the symmetrized kernels before treating each the irreducible representations. The time required for the assembling step is then given by :

$$T_{ass_k}(p) = O\left(\left(\frac{k}{n_G} \frac{\alpha n^2}{p} + \frac{\alpha' n^2}{p} \right) \omega \right) \tag{13}$$

where $\alpha' << \alpha$.

This time should be compared to (11).

The relationship (12) gives the execution time for solving a $n \times n$ linear system, the time required to solve the n_G/k subproblems of size n/n_G is then :

$$T_{gs_k}(p) = O\left(\frac{1}{n_G^2} \frac{2n^3}{3p} \omega + \frac{p}{k^2} \left(n\beta + \frac{1}{n_G} n^2 \frac{\tau}{2} \right) \right) \tag{14}$$

When $k=1$, we have the situation of a single ring of processors and we find respectively :

$$T_{ass_k}(p) \approx k\, T_{ass_1}(p)$$

$$T_{gs_k}(p) \approx \frac{1}{k} T_{gs_1}\left(\frac{p}{k} \right)$$

The global tendancy is not obvious since a k-rings architecture decreases the efficiency of the forming step but speeds-up the solving step. So, there is a trade-off that can only be determined by conducting numerical experiments as presented in the next section.

Remark : The last superposition step is of minor importance since the communication and computation times are of order n.

EXPERIMENTAL RESULTS

The experimental results are obtained from a Meiko Computing Surface with 16 T800 transputers. We use C-language and CS-Tools environment for the implementation. The time parameters are the following :

$\beta \approx 145\ \mu s,\quad \tau \approx 6.5\ \mu s,\quad \omega \approx 1.5\ \mu s$ (+) *to* 45 μs (log)

In this section the condition stating p/k as an integer is sometimes violated and the extra processor which collects and superposes the partial solutions is one of the p processors used. In fact, this emphasizes the practical situation where the user has got up to p processors and has to build at best the possible configurations related to its problem. Nevertheless, the timing results are in accordance with theoretical expressions.

First of all, the cases k=1 and n_G are compared. The example proposed is the same as shown in figure 3, the domain Ω exhibits the abelian group C_3 presented above and $n_G=3$.

The boundary conditions are three potentials of any values on the top of each arm of Ω and we have to find ϕ and its normal derivative on the boundary $\partial\Omega$. The group representation theory allows the decomposition of the problem into 3 subproblems to be solved on Γ.

Several combinations are considered with the number p of processors and the size n of the problem. Figure 5 gives a plot of the ratio $\gamma = T_{tot,ns}/T_{tot,ws}$ of the global execution times T_{tot} with no and with symmetry consideration on a single ring as a function of p for different values of n = 48, 96, 192, 384. As expected, we see that the exploitation of symmetry is generally more interesting than no taking advantage of it ($\gamma > 1$). The computational gain increases as the size n increases and the number of processors p decreases. This tendancy is corroborated by the theoretical times (11) to (14).

The comparison between k=1 and k=3 in figures 6 a, b, c, d points out that the relative efficiency depends on the number of processors for a given n. When few processors are used, the 1-ring solution is the best. But as p increases, the amount of communications due to the Gaussian elimination ($\propto p$) is such that the 3-rings option begins better. The transition number of processors increases with n.

In order to generalize as explained above, we consider the case of the cyclic group C_6 where n_G equals to 6. The experimental results obtained from an appropriate example are given for n=96 and 384. If we now plot the global time as a function of k for fixed n and various values of p (figs. 7 a, b), the advantage of taking k>1 is obtained as p exceeds a critical value p^*. This threshold increases with n and the related k^* seems increasing with p. Those conclusions are in agreement with the theoretical times and constitute a generalization of the results obtained with $n_G=3$.

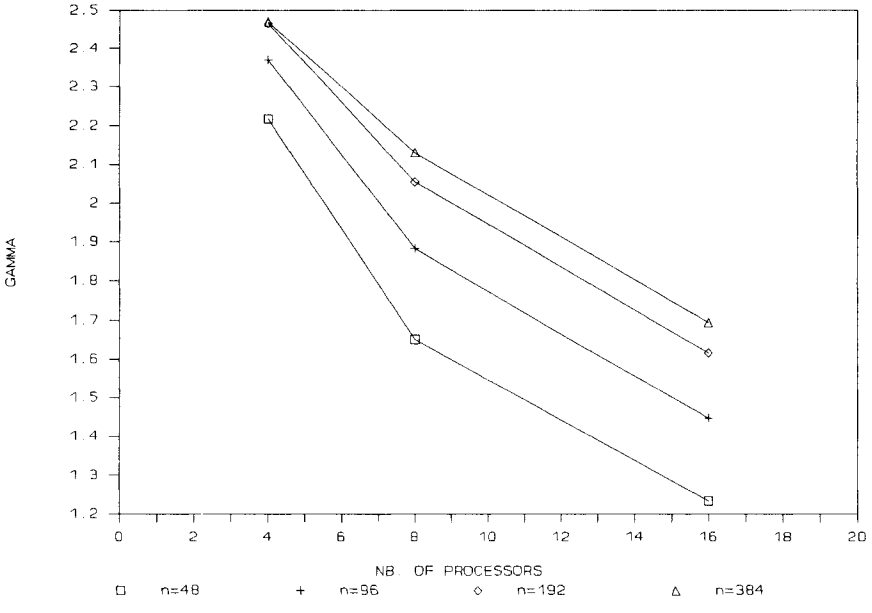

Figure 5. Ratio γ versus p.

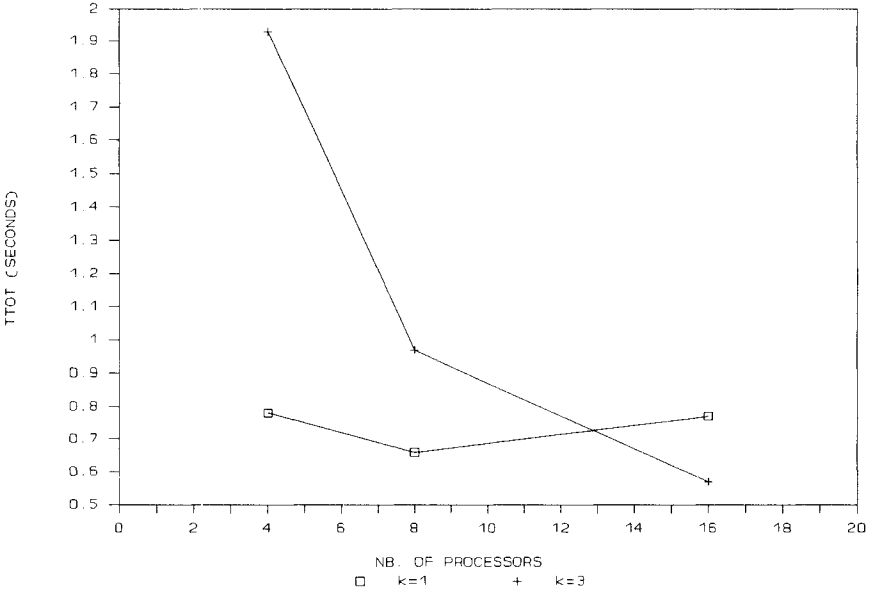

Figure 6a. $n_G=3$, T_{tot} versus p, n=48.

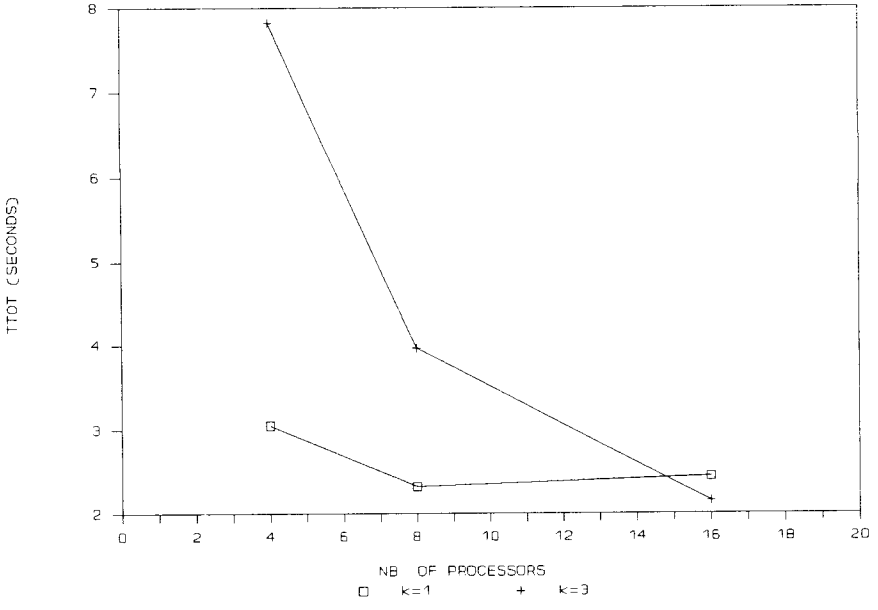

Figure 6b. $n_G=3$, T_{tot} versus p, n=96.

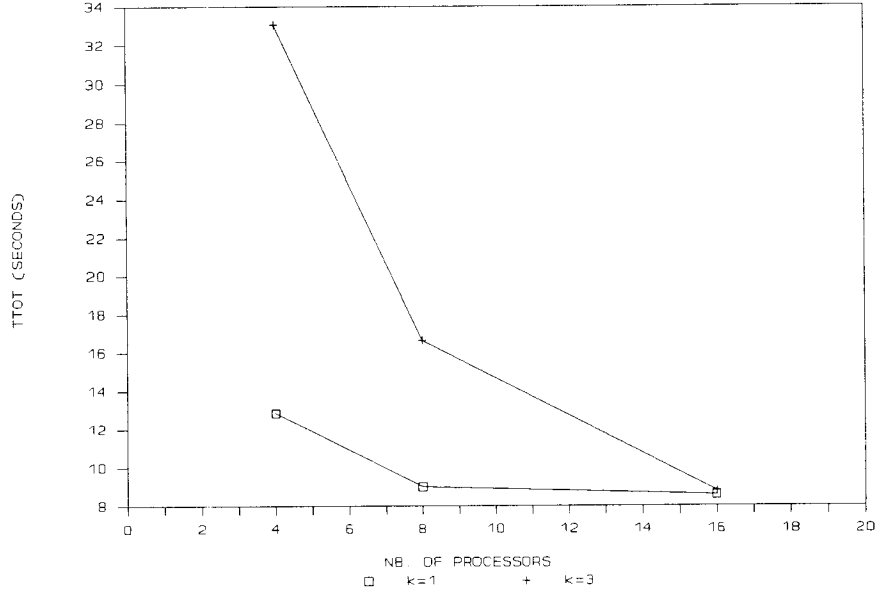

Figure 6c. $n_G=3$, T_{tot} versus p, n=192.

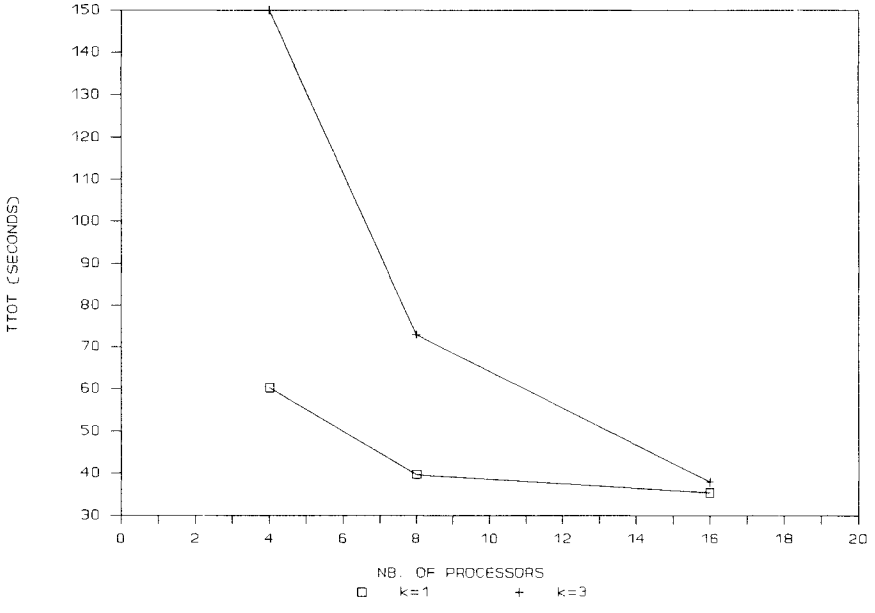

Figure 6d. $n_G=3$, T_{tot} versus p, n=384.

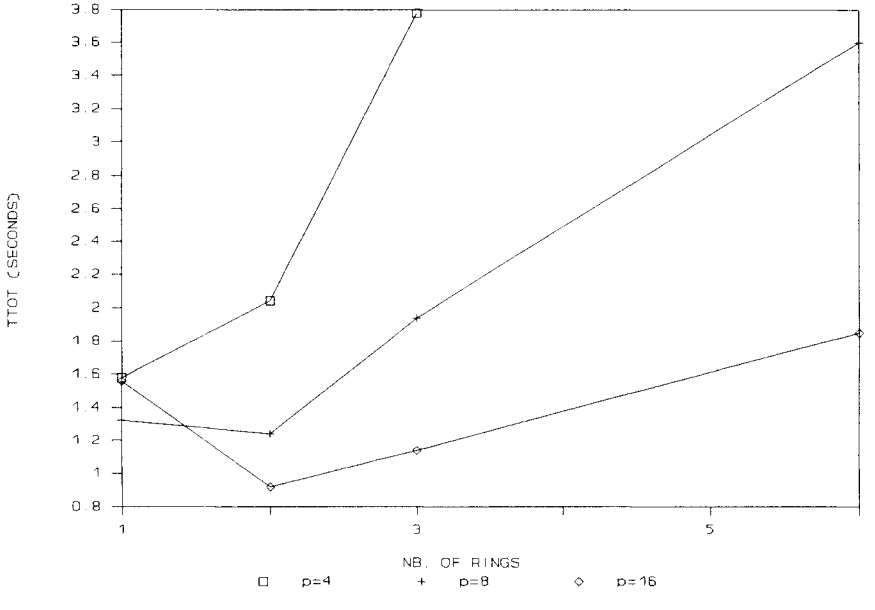

Figure 7a. $n_G=6$, T_{tot} versus p, n=96.

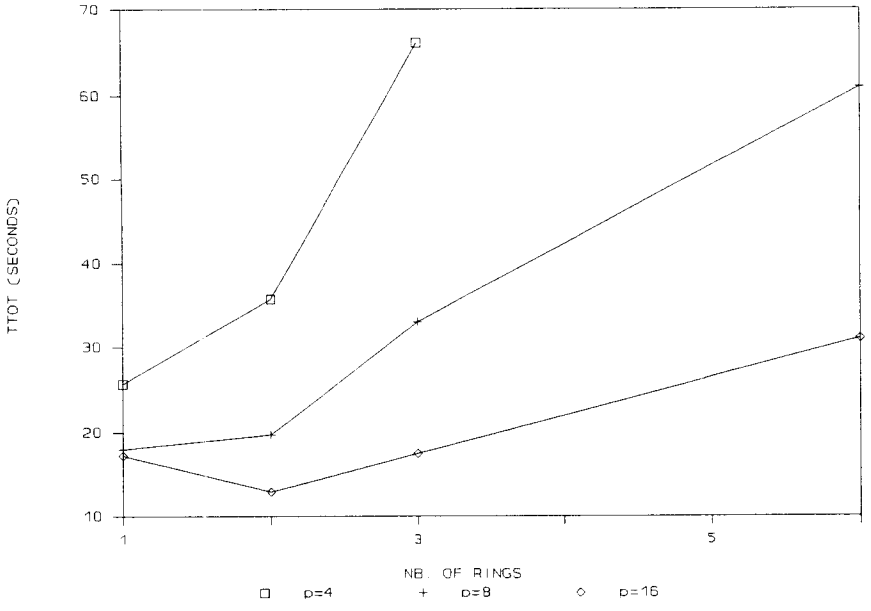

Figure 7b. $n_G=6$, T_{tot} versus p, n=384.

From the timing results, we deduce some interesting conclusions.

If we dispose of a given set of p processors, the topology (k) is strongly function of the size n of the problem. Generally, a single ring (k=1) should be used if p is small. But as p increases, the best solution tends to be a multi-ring (k>1) topology, this is all the more true as n is small. However, since the execution times asymptotically increase with p, the optimum number of processors may be lower than p if a large quantity of processors is available.

CONCLUSIONS

In this paper, a Boundary Element Method for solving the 2-D Laplace's problem with symmetrical domains has been implemented on a multi-processors (16 T800 transputers) architecture. Several configurations based on the ring topology were considered. A compromise exists between single ring and multi-rings topologies. It depends on the number of processors and the size of the problem. The abelian case has been considered but the general non-abelian case should present about the same conclusions, yet the implementation is more complicated because of some coupled problems.

REFERENCES

1. Bryant, C.F., Roberts M.H. and Trowbridge C.W. 'Implementing a Boundary Element Method on a Transputer System' *IEEE Trans. Magn.*, Vol. MAG-26, no. 2, pp.819 - 822, 1990.
2. Daoudi, E.M. and Lobry, J. 'Implementation of a Boundary Element Method on Distributed Memory Computers' *Parallel Computing*, Vol. 18, pp. 1317 - 1324, 1992.
3. Lobry, J. and Daoudi, E.M. 'Implémentation de la Méthode des Eléments des Frontières pour les problèmes de magnétostatique 3D sur architecture parallèle à mémoire distribuée' *J. Phys. III France*, pp. 2055 - 2068, 1992.
4. Hamermesh, M. *Group theory*, Addison-Wesley Publishing Company, Massachusetts, Palo Alto and London, 1964.
5. Serre, J.-P. *Représentations linéaires des groupes finis*, Hermann, Paris, 1971.
6. Bossavit, A. 'The exploitation of geometrical symmetry in 3-D eddy-currents computation' *IEEE Trans. Magn.*, vol. MAG-21, no. 6, pp.2307-2309, 1985.
7. Allgower, E.L., Böhmer, K., Georg, K. and Miranda, R. 'Exploiting Symmetry in Boundary Elements' *SIAM J. Numer. Anal.*, Vol. 29, pp. 534-52, 1992.
8. Bonnet, M. On the use of geometrical symmetry in the boundary element methods for 3D elasticity," *BETECH 91* (C.A. Brebbia, ed.), CMP (Southampton), 1991.
9. Lobry, J. and Broche, C. 'Exploitation of the Geometrical Symmetry in the Boundary Element Method with the Group Representation Theory' to appear in *IEEE Trans. Magn.*.
10. Brebbia, C.A. *The Boundary Element Method for engineers*, Pentech Press, Plymouth, 1980.
11. Saa, Y. and Schultz, M.H. 'Data communication in hypercubes' RR428, Computer Science Dpt., Yale University, 1985.