A computational approach for three-dimensional probabilistic discrete cracking in concrete

C. N. M. Paz, L. F. Martha
Department of Civil Engineering and Technology Group on Computer Graphics (Tecgraf)
Pontifical Catholic University of Rio de Janeiro (PUC-Rio) – Brazil Rua Marquês de São Vicente 225, RJ / Brazil Z.C. 22453-900 Telephone: 55-21-3114-1193 Fax:55-21-3114-1848
e-mail: {mena,lfm}@tecgraf.puc-rio.br

J. L.D. Alves, N. F.F. Ebecken, E. M.R. Fairbairn , and A. L.G.A. Coutinho
COPPE/UFRJ - Program of Civil Engineering Federal University of Rio de Janeiro - Brazil
C.P. 68506, 21945-970 Telephone:55-21-2562-7390Fax:55-21-2562-8464
e-mail: jalves@lamce.ufrj.br, nelson@ntt.ufrj.br, , eduardo@labest.coc.ufrj.br, alvaro@nacad.ufrj.br

ABSTRACT: This work presents a probabilistic crack approach based on the Monte Carlo method, implemented in a 3D fully parallelized finite element code (Paz, 2000). The cracking scheme used is the discrete crack approach. In this approach the heterogeneity of the material is taken into account by considering the properties to vary spatially following a normal distribution determined by the mean and the standard deviation of the considered material properties.
Fracturing is modeled by 3D interface elements generated in a previously defined region within the mesh (Paz 2000). The interface elements are triangular base prisms connecting adjacent faces of neighboring tetrahedra. These elements simulate crack opening through relative displacements between the triangular faces.

1 INTRODUCTION:

1.1 Probabilistic model

Concrete cracking depends on several relevant factors such as water/cement ratio, casting and curing, loading conditions, etc. Due to the random distribution of constituents and initial defects, the heterogeneity governs the overall cracking behavior and related size effects on concrete fracture. The probabilistic crack approach based on the direct Monte Carlo method developed by Rossi and coworkers 1994a,b 1996,1997, takes this stochastic process into account by assigning in finite elements analysis, randomly distributed material properties (tensile strength, Young's modulus) to both the solid elements and the interface elements (figure 1). The stochastic process is introduced at the local scale of the material, by considering that cracks are created within the concrete with different energy dissipation depending on the spatial distribution of constituents and initial defects.

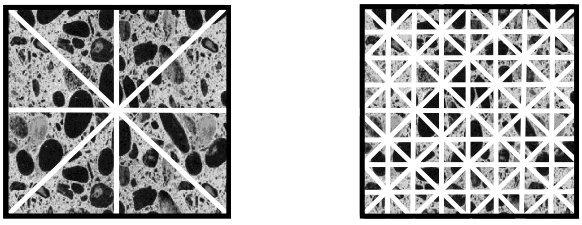


Figure 1: This stochastic process into account by assigning in finite elements analysis, randomly distributed material properties.

The local material behavior in concrete is assumed to obey a perfect elastic brittle behavior, so that the random distribution of local cracking energies can be replaced by a random distribution of local strengths. Therefore, solid elements are elastic, while interface elements are considered elastic-brittle.

Thus, the present probabilistic model involves a number of mechanic properties of the material to be determined, which constitutes the modeling data. From a large number of direct tensile tests it was found that a normal law describes rather well the experimental distribution (Rossi *et al.* 1994b). These characteristics are: $f_{ct,\mu}$ and E_μ, the means of the tensile strength and of the Young's modulus respectively; $f_{ct,\sigma}$ and E_σ, the standard deviations of the tensile strength and of the Young's modulus respectively. The following analytical expressions were proposed:

$$f_{ct,\mu} = 6.5\,(V_t/V_g)^{-a}; \quad f_{ct,\sigma}/f_{ct,\alpha\mu} = 0.35\,(V_t/V_g)^{-b} \qquad (1)$$

$$E_\mu = E \qquad E_\sigma / E = 0.15\,(V_t/V_g)^{-c} \qquad (2)$$

where V_t is the volume of the two finite elements contiguous to an individual contact element of the mesh; V_g is the volume of the coarsest aggregate; E is the average Young's modulus that does not exhibit significant volume effects. For cylinder specimens, whose dimensions are 160 mm in diameter

and 320 mm high constants *a, b* and *c* are related to the compressive strength f_c given by the relations,

$$a = 0.25 - 3{,}6\times10^{-3}(f_c) + 1.3\times10^{-5}(f_c)^2$$

$$b = 4.5\times10^{-2} + 4.5\times10^{-3}(f_c) - 1.8\times10^{-5}(f_c)^2$$

$$c = 0.116 + 2.7\times10^{-3}(f_c) - 3.4\times10^{-6}(f_c)^2 \qquad (3)$$

In these expressions the compressive strength f_c represents the quality of the concrete matrix, while the volume of the coarsest aggregate V_g, refers to the elementary material heterogeneity.

Equations (1) to (3) show that the smaller the scale of observation, the larger the fluctuation of the local mechanical properties, and thus the (modeled) heterogeneity of the matter. In other words, the finer the mesh, the greater the modeled heterogeneity in terms of Young's modulus and tensile strength.

Let m_v the number of tetrahedra and m_i the number of interface elements in a given mesh. Also let the set of all Young's modulus of the solid elements be denoted by $\mathbf{E}$ and $\mathbf{f}_{ct}$ the at of all tensile strength of interface elements

According to Rossi *et al*. 1994b these individual local tensile strengths and Young's modulus are represented by normal distributions having the densities:

$$g_f(f_{ct}) = \frac{1}{f_{ct,\sigma}\sqrt{2\pi}}\exp\left[-\frac{1}{2}\left(\frac{f_{ct,\mu}}{f_{ct,\sigma}}\right)^2\right] \qquad (4)$$

$$g_E(E) = \frac{1}{E_\sigma\sqrt{2\pi}}\exp\left[-\frac{1}{2}\left(\frac{E-E_\mu}{E_\sigma}\right)^2\right] \qquad (5)$$

where $g_f(f_{ct})$ and $g_E(E)$ are density functions for the tensile strength f_{ct} and the Young's modulus E, respectively, and x_μ and x_σ denote the mean and standard deviation of the distribution of quantity x. For the problem at hand, it is possible to find a sample of m_i values $f_{ct,ii}$, each value corresponding to an interface element, and m_v values E_{iv}, each value corresponding to a volume element, by using a standard routine for generation of random numbers for a given normal distribution (Press *et al*., 1992).

The problem with this approach is that these statistical moments are not known a priori for the characteristic volume of the finite elements used in the analysis. However, some methods have been proposed to determine these parameters by means of inverse analysis using neural networks Paz 2000, Farbairn 1999, 2000a, b.

The solution for this probabilistic approach is obtained by means of a Monte Carlo simulation. As depicted in figure 2, in a Monte Carlo simulation a number of *n* samples are generated for a given normal distribution and some characteristic responses of the structure. A characteristic response may be for example a stress crack-width $\sigma - w$ curve. Let the *j*th samples correspond to the *j*th $\sigma - w$ curve. This *j*th $\sigma - w$ curve is composed of discrete values, σ_k^j and w_k^j, where the superscript *j* indicates the sample and the subscript *k* the discrete value of the $\sigma - w$ curve. The discrete w values are assumed known and the response is defined exclusively by the values of σ_k^j. The mean curve composed by pairs σ_k^{mean}, w_k then simply reads

$$\sigma_k^{mean,j} = \frac{1}{j}\sum_{l=1}^{j}\sigma_k^l \qquad (6)$$

The Monte Carlo simulation is stopped when

$$\left|\sigma_k^{mean,j} - \sigma_k^{mean,j-1}\right| \le tol \qquad (7)$$

where *tol* is the prescribed tolerance to check the convergence of the procedure. Where convergence is reached the number of samples is set to $n = j$. This total number of samples *n*, corresponding to a Monte Carlo converged simulation, clearly depends on *tol*, which is a measure of the accuracy required by the analysis. It also depends on the heterogeneity of the material represented by the standard deviation. The more heterogeneous is the material the greater is the number of samples necessary to obtain a converged solution. Our experience in this field indicates that 15 to 30 samples are sufficient to obtain a converged $\sigma - w$ curve.

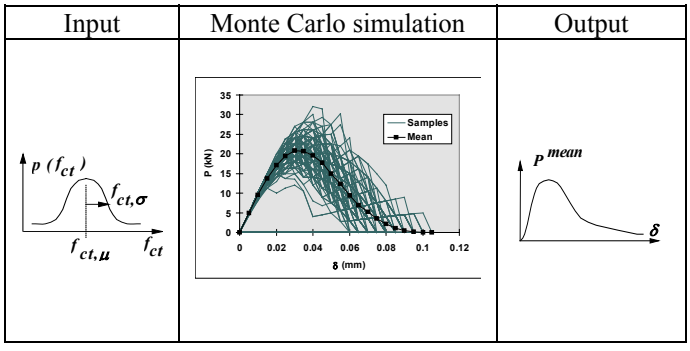


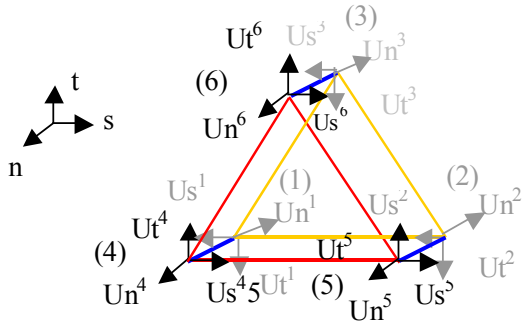
Figure 2: Monte Carlo simulation.

2 DISCRETE CRACKING: 3D INTERFACE ELEMENTS

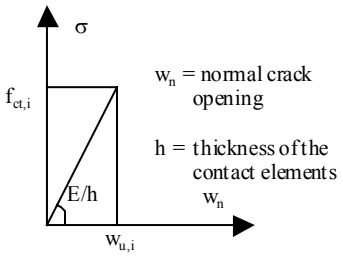
The finite element cracking model is a discrete model for which volume elements are always elastic and cracking occurs in elastic-brittle (almost rigid

brittle) contact elements placed between two neighbor surfaces of the volume elements. The 3D interface elements (Paz 2000) depicted in Figure 3 (a) can be thought as triangular base prisms connecting adjacent faces of neighboring tetrahedra.

These elements are formulated to represent relative displacements between the triangular faces to simulate crack opening.



(a)



(b)

Figure 3: (a) An interface element and its degrees of freedom in a local system; (b) Elastic-brittle contact law.

The constitute law of the 3D interface element is defined by equation (8) for a non cracked elastic state, characterized by $\sigma_n < f_{ct,i}$. When the tensile strength is exceeded, the elements reach a cracked stage and module E_c and G_c are set to zero as shown in figure 3b.

$$\Delta\boldsymbol{\sigma} = \mathbf{D}_{cr}\,\Delta\mathbf{w} = \begin{Bmatrix}\Delta\sigma_n\\ \Delta\sigma_s\\ \Delta\sigma_t\end{Bmatrix} = \begin{bmatrix}E_{c/h} & 0 & 0\\ 0 & G_c/h & 0\\ 0 & 0 & G_c/h\end{bmatrix}\begin{Bmatrix}\Delta w_n\\ \Delta w_s\\ \Delta w_t\end{Bmatrix} \quad (8)$$

In equation (8) the subscript *n* normal stands for, while stand for *s* and *t* tangential indicating the direction respective to the crack plane, *w* are the relative displacements between the two faces of the interface element, *h* is the width of interface element, E_c and G_c are respectively the normal (Young's) and the shear modulus along crack plane.

Equations (8) and figure 3 define the elastic-britle constitute behavior. The thickness *h* plays the role of a penalization parameters and should be conveniently chosen not to affect the solution.

The kinematic relation for the interface element is given by:

$$\Delta\mathbf{w} = \mathbf{B}\,\Delta\mathbf{a}_l^e \quad (9)$$

$\Delta\mathbf{w}$ is the crack opening incremental and $\Delta\mathbf{a}_l^e$ is the vector incremental nodal displacements for the interface element

Applying a standard displacement based F. E. formulation, the resulting tangent stiffness matrix for the interface element is given is:

$$\mathbf{K}^e_{\text{Intf}} = \int_\Omega \mathbf{B}^T\,\mathbf{D}_{cr}\,\mathbf{B}\,d\Omega \quad (10)$$

The interface elements are generated contiguous to the faces of selected tetrahedra elements. This selection is performed by the user, defining a 3D box inside the mesh that contains the target elements.

Remark Our experience indicates that to increase robustness of nonlinear solution process we have to limit only one interface element to "crack" at each nonlinear iteration.

3 3D INTERFACE ELEMENTS MESH GENERATION

Fracturing is modeled by 3D interface elements generated in a previously defined cracked region. This selection is performed by the user defining a 3D box inside the mesh that contains the target elements. The interface elements are generated (Paz 2000) contiguous to the faces of the selected tetrahedra.

The procedure initially establishes the neighborhoods of the faces, then it maps how many elements share each node. Later it creates the nodes necessary to the interface elements, all nodes having the same coordinates of the neighbor node. After this all nodes with the same coordinates are visited, and the first element of the loop takes the existing node of the initial mesh of tetrahedra (see figure 4a) and, for the other elements that share this node, it is introduced a new node numbering (see figure 4b) Later connectivities are created, introducing the interface elements according to an ordering previously established.

Interface elements with collapsed nodes provide continuity to the elements outside the cracked region (see figure 4c). These elements are implemented using an artifice that allows the use of elements with six nodes. This artifice consists of multiple references for the same collapsed nodes duplicating the node numbering for the elements outside the 3D box.

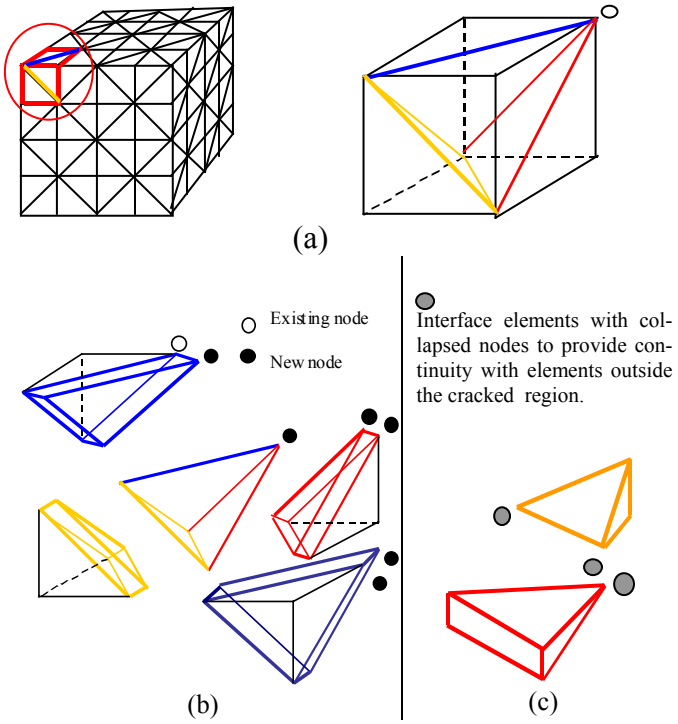


Figure 4: (b,c) 3D Interface elements mesh generation.

4 COMPUTATIONAL STRATEGIES; PARALLEL VECTOR PROCESSOR (PVP)

4.1 Solution of equilibrium equations and the inexact Newton Method

Traditional finite element technology for non linear problems involves the repeated solution of systems of sparse linear equations by a direct solution method, that is, some variant of Gauss elimination. The updating and factorization of the sparse global stiffness matrix can result in extremely large storage requirements and a very large number of floating point operations.

In this paper we employ an Inexact Newton method (Kelley 1995), to solve large-scale three-dimensional incremental elastic-brittle problems. In the Inexact Newton Method, at each non linear iteration, a linear system of finite element equations is approximately solved by the preconditioned conjugate gradient method (PCG).

The implementation of global matrix-vector products are easily parallelized in different computer architectures, performing element level products followed by global assembly. This type of implementation is often referred to element-by-element (EBE) schemes. Matrix-vector products computed by EBE schemes are memory intensive, requiring more operations than the product with the assembled matrix, because element matrices have many overlapping non-zero entries. However, particularly for large scale non linear problems EBE methods have been very successful, because they handle large sparse matrices in a simple and straightforward manner. Besides, efficient preconditioners may be derived keeping the same data structure. For a recent review of such topics see Hughes 1987.

When solving iteratively the finite element system of linear equations, it is straightforward to employ inexact versions of the standard Newton-like methods. In this case, tolerances for the inner iterative driver may be adaptively selected to minimize computational effort towards the solution, giving rise to the following algorithm:

Given u_{tol}, r_{tol}, η_i relative and residual tolerance.

Compute stiffness tetrahedra matrix $\mathbf{K}_{Tetra}$

do k=1,2….., number of load increments do

Compute external forces vector

$$\mathbf{F}_{ext}^{k} = \mathbf{F}_{nodal}^{K} + \mathbf{F}_{volume} + \mathbf{F}_{\bar{\sigma}} - \left(\mathbf{K}_{tetra}\, \bar{\mathbf{U}}_k + \mathbf{K}_{Intef}\, \bar{\mathbf{U}}_k \right)$$

do i=1,2 ..., while convergence

Compute internal forces vector,

$$\mathbf{F}_{\text{int}}^{i} = \left(\mathbf{F}_{\text{int}}^{i} \right)_{Tetra} + \left(\mathbf{F}_{\text{int}}^{i} \right)_{Intf}$$

Compute residual vector,

$$\boldsymbol{\psi}^{i} = \mathbf{F}_{\text{int}}^{i} - \mathbf{F}_{ext}^{i}$$

Update stiffness interface matrix $\mathbf{K}_{\text{Intf}}^{\text{i}}$

$$\mathbf{A}^{i} = \mathbf{K}_{Tetra} + \mathbf{K}_{Intf}^{i}$$

Compute tolerance for iterative driver, η_i

Solver: $\mathbf{A}^{i}\ \Delta\mathbf{u} = \boldsymbol{\psi}^{i}$ for tolerance η_i

Update solution,

$$\mathbf{U} = \mathbf{U} + \Delta\mathbf{u}$$

if $\dfrac{\| \Delta\mathbf{u} \|}{\| \mathbf{U} \|} \le utol$ and $\dfrac{\| \Delta\boldsymbol{\psi}^{i} \|}{\| \mathbf{F}_{ext}^{k} \|} \le rtol$ then convergence

end while i.

end do k.

Note that in $\mathbf{F}_{ext}^{k}$ we account for nodal forces, body forces $\bar{\ }$ and prescribed displacements and stresses $\mathbf{U}$, $\boldsymbol{\sigma}$. The total internal forces vector $\mathbf{F}_{\text{int}}^{i}$ is the sum of the solid elements internal forces $\left(\mathbf{F}_{\text{int}}^{i} \right)_{Tetra}$ plus the interface element internal forces vector $\left(\mathbf{F}_{\text{int}}^{i} \right)_{Intf}$. The total stiffness matrix is the sum of the continuum matrix $\mathbf{K}_{Tetra}$ plus the interface matrix $\mathbf{K}_{Intf}^{i}$ updated at each non linear iteration.

We adopted a simple nodal block-diagonal preconditioner. Therefore, the most expensive computational kernel in the linear solver is the matrix-vector product. According to the above algorithm, an approximate solution is obtained when the Inexact Newton termination criterion is satisfied, that is, when,

$$\left\| \mathbf{A}^{i}\ \Delta\mathbf{u} - \Psi^{i} \right\| \le \eta_i \left\| \Psi^{i} \right\| \qquad (11)$$

We selected η_i as suggested by Kelley [11], based on a measure of how far the nonlinear iteration is from the solution, that is,

$$\eta_i^A = \min\left(\eta_{\max}, \gamma \frac{\left\| \boldsymbol{\psi}^i \right\|^2}{\left\| \boldsymbol{\psi}^o \right\|^2}\right), \qquad 0 < \gamma < 1 \quad (12)$$

If η_i^A is uniformly limited away from 1, and taking $\eta_i = \max\left(\eta_{\min}, \eta_i^A\right)$ Kelley [11] has shown general convergence properties when Equation (12) is used. To avoid that η_i^A be too small when the nonlinear iteration is away from the solution, Kelley also suggests the following modification,

$$\eta_i^B = \begin{cases} \min\left(\eta_{\max}, \eta_i^A\right) & , if\ \gamma\, \eta_{i-1}^2 < 0.1 \\ \min\left(\eta_{\max}, \max\left(\eta_i^A, \gamma\, \eta_{i-1}^2\right)\right) & , if\ \gamma\, \eta_{i-1}^2 \geq 0.1 \end{cases} \quad (13)$$

In some cases $\|\boldsymbol{\psi}_i\|$ can be very small, well beyond the required accuracy, resulting in undesired work. To remedy this over solving Kelley 1995 proposes to compute η_i^C using;

$$\eta_i^C \left(\eta_{\max}, \max\left(\eta_i^B, 05\, rtol \frac{\left\| \mathbf{F}_{ext}^k \right\|}{\left\| \boldsymbol{\psi}^i \right\|} \right) \right) \quad (14)$$

and finally taking $\eta_i = \max\left(\eta_{\min}, \eta_i^C\right)$.
Our experience indicates that selecting $\eta_{\max}, = 0.1\ and\ 10^{-3} \leq \eta_{\min} \leq 10^{-6}$ for *utol* and *rtol* in the usual range, that is, 10^{-3} to 10^{-2}, is enough for practical engineering computations. Typical values for γ and *g* are 0.5 and 0.1, respectively

4.2 Matrix-vector products element-by-element, EBE

In the present implementation matrix vector products in EBE, PCG are computed as:

$$\mathbf{A}\,\mathbf{p} = \sum_{i=1}^{N_{tetra}} \left(\mathbf{K}_{\text{Tetra}}\ \ \mathbf{p}_i \right) + \sum_{j=1}^{N_{Intf}} \left(\mathbf{K}_{\text{Intf}}\ \ \mathbf{p}_j \right) \quad (15)$$

where, N_{tetra} is the number of tetrahedra, N_{intf} is the number of interface elements, $\mathbf{K}_{\text{Tetra}}$ and $\mathbf{K}_{\text{Intf}}$ are respectively element matrices for the tetrahedra and interface; $\mathbf{p}_i$ and $\mathbf{p}_j$ are the components of $\mathbf{p}$ restricted to the degrees of freedom of two element type.

Stiffness matrices for tetrahedra are computed and stored at the beginning of the analysis since they are elastic.

Stiffness matrix for interface elements are updated at every nonlinear iteration.

The arrays of element stiffness matrices are stored taking into account their symmetry; in the case of the element tetrahedra 78 coefficients are stored and for the interface element only 18 coefficients are stored, exploring the particular structure of the discrete gradient operator.

The mesh coloring algorithm of Hughes 1987 was extended to block both solid and interface elements into disjoint groups thus enabling full vectorization an parallelization of the operations involved in equation (15).

5 NUMERICAL EXAMPLE

The analized example is direct tension tests (figure 5). This numerical example was analyzed by Rossi et al 1997 in a bidimensional program .

The numerical experiment were controlled by a field of uniform displacements applied at the top of the test specimen in 30 incremental steps $\Delta u = 4.0 x 10^{-3}$. The boundary conditions restrain the degrees of freedom in the vertical at the bottom. Concrete with maximum aggregate diameter of 10.00 mm was used. Its average tensile strength and Young's modulus at the age of 28 days were: $f_{ct} = 3.0 \quad MPa$ and $E_c = 30000 \quad MPa$.

The final mesh figure 6, has 4,756 elements, where 2,000 are tetrahedra and 2,726 are interface elements, and node numbers 4,181. The final mesh figure7, has 3,292 elements, where 2,000 are tetrahedra and 1,292 are interface elements and node numbers 2,131.

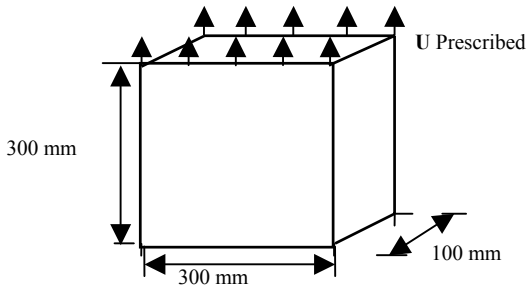


Figure 5: Uniaxial tension specimen geometry, dimensions, load, boundary conditions

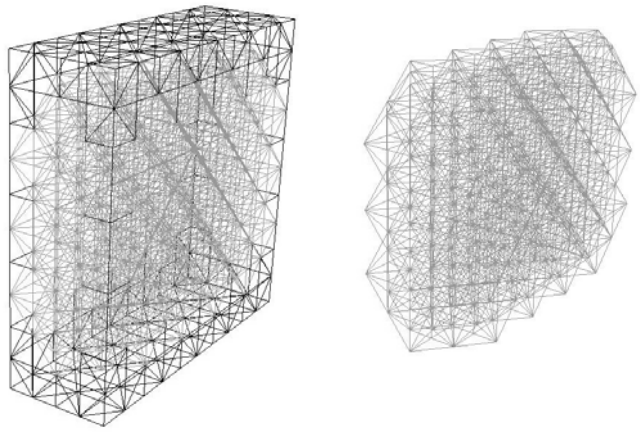


Figure 6: Representations of the computational mesh for the simulation of a direct tension test. It is also shown aside the resulting mesh for the interface elements considering. (mesh a)

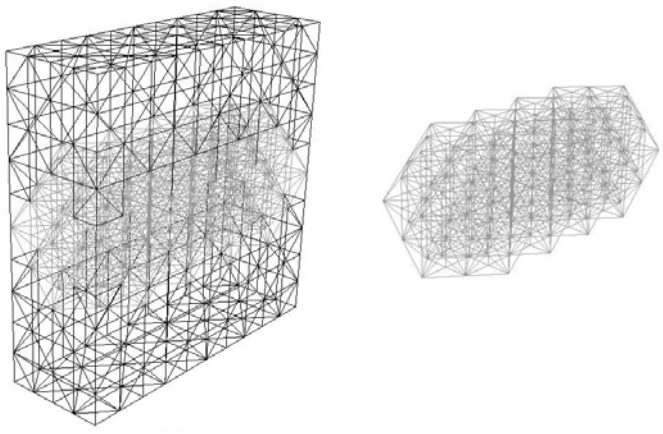


Figure 7: Representations of the computational mesh for the simulation of a direct tension test. It is also shown aside the resulting mesh for the interface elements considering. (mesh b)

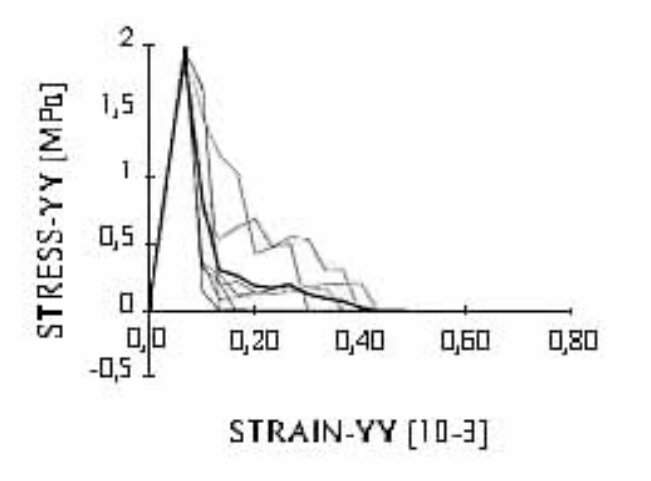


Figure 8 : Curve stress – strain Rossi et al. 1997

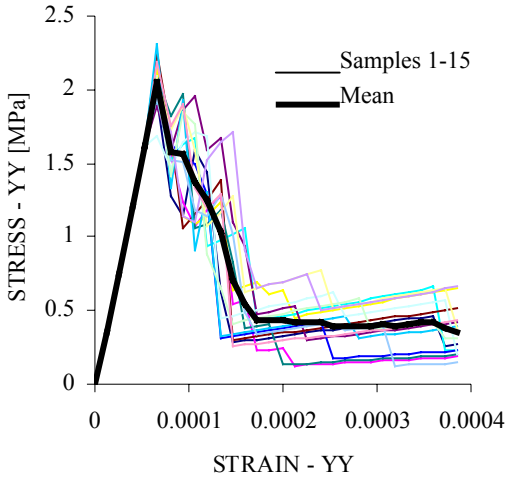


Figure 9: Results for the complete Monte Carlo simulation (mesh a).

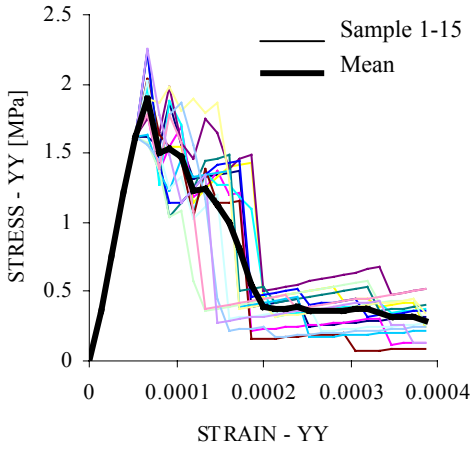


Figure 10: Results for the complete Monte Carlo simulation (mesh b).

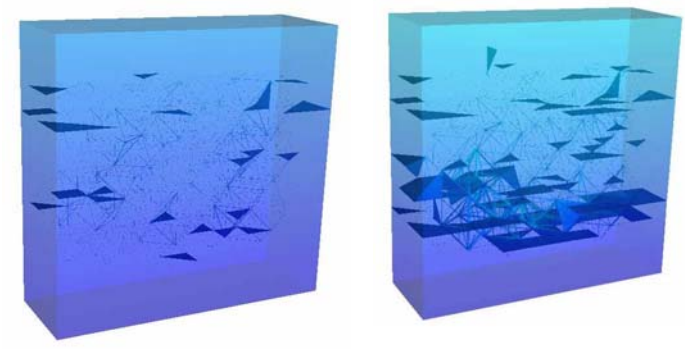


Figure 11: Crack evolution for numerical simulation (mesh a)

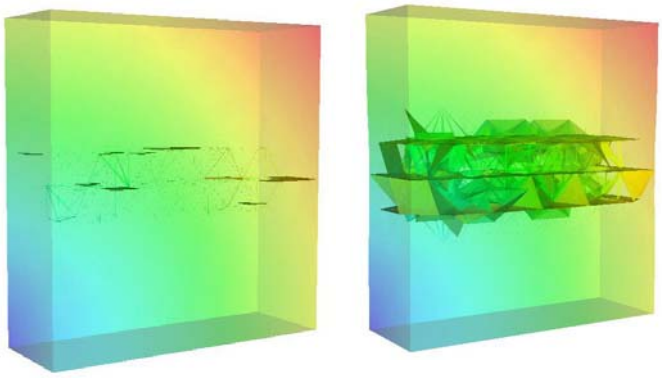


Figure 12: Crack evolution for numerical simulation (mesh b)

6 COMPUTATIONAL PERFORMANCE

A detailed vector performance analysis is obtained by the summary of the PERFVIEW's Report presented in figures 13 and 14. The CPU time of the vectorized single processor run for CRAY T90 are 1.48 hours and 1.02 hours (mesh (a) and (b) respectively). This table list for a single CPU run, the Mflop/s rates for the three top routines. The routines **Smatv-fint** and **Smatv-tetra** are respectively responsible for the matrix-vector products for the in-

terface and tetrahedra elements, routines are the computations kernels the routine **PCG-block,** implements the iterative solver the nodal block diagonal Preconditioned Conjugate Gradient (PCG).

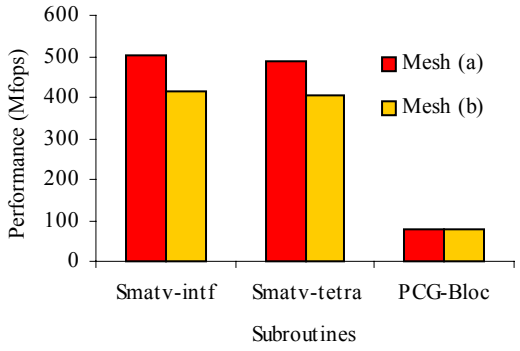


Figure 13: Performance Analysis – The top three subroutines – The 2 Meshes

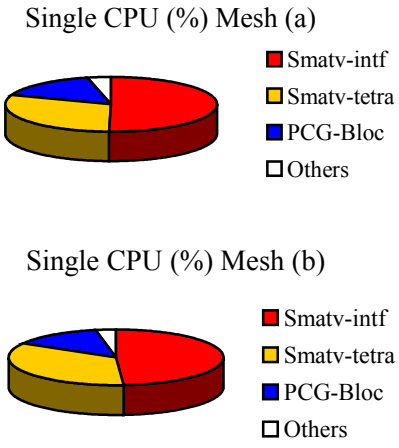


Figure 14: Performance Analysis- Single CPU (%) – The top the subroutines

The code achieved good vectorization on the CRAY T90 for a mesh (a). The top three subroutines are consume the major CPU utilization in the whole analysis.

The parallel performance is shown figures 15 (a,b) as obtained from a summary ATEXPERT, report autotasking performance tool. The top five subroutines assigned for parallel analysis are presented for the direct tension test. The routines **Fint-tetra** and **Fint-intf** evaluate respectively the internal force vector of the interface and tetrahedra elements. The routine **Kintf** computer update interface elements stiffness.

According to ATEXPERT tool this program appears to be 99.2 percent parallel and 0.8 percent serial. Amdahl's Law predicts the program could expect to achieve a 3.9 times speedup on 4 CPU's. A 3.8 speeup is predicted with 4 CPU's on dedicated system.

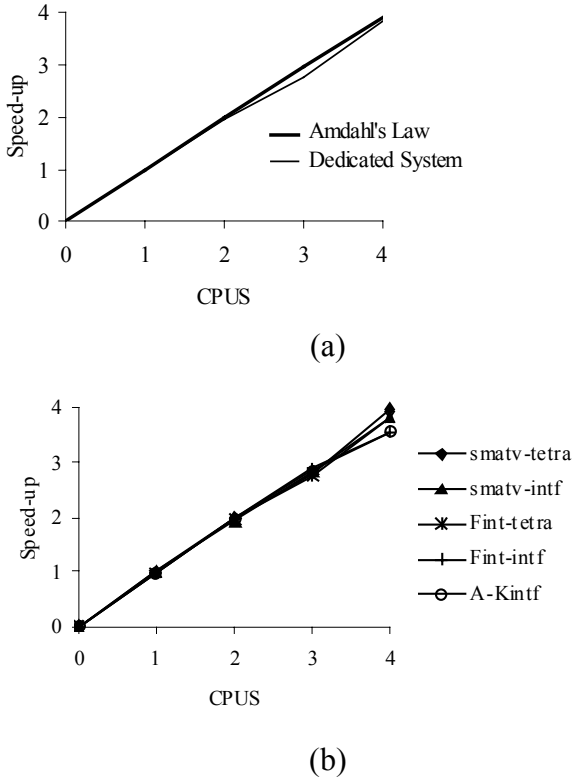


Figure 15: (a) Program Summary of the ATEXPERT's Report
(b) ATEXPERT's Report -The top 5 subroutines on a Dedicated Speedup

7 CONCLUDING REMARKS

This paper presented the optimized implementation of Rossi's a probabilistic model for the simulation of cracking in concrete structures. This model in based on the assumption that some particularities of the cracking behavior of concrete, such as strain softening, cracking evolution and size-effects are derived from the heterogeneous characteristics of the material.

The probabilistic methodology presented in this paper corresponds to the 3D analysis of a strongly nonlinear material that develops cracks. In adition, the finite elements analysis must be called several times within a Monte Carlo simulation. Therefore, the code needs to be optimized in such a way that the simulation time does not exceed a practical limit.

The code achieved a very good level for both parallel performance and vetorization. The most demanding routines, which implement the matrix-vector-multiply computational kernel for the interface and tetrahedral elements, are "fully" parallelized (~ 99%) and responsible for over 80% of CPU time. The results emphasizes the suitability of the

implemented code on the parallel-vector machine, CRAY T90 for 2 CPU's, which presented a flop rate of 614 Mflop/s and a parallel speed-up of 3.8 for 4 CPU's.

Extensive use of element-by-element techniques within the computational kernels comprised in the iterative solution drivers provided a natural way for achieving high Flop rates and good parallel speed-up's. Furthermore, element-by-element techniques, avoid completely the formation and handling of large sparse matrices. Therefore, the computational strategies presented herein provide a natural way to deal with more complex scenarios, particularly those involving three-dimensional problems.

Acknowledgments

The authors are indebted to the Computer Graphics Technology Group TECGRAF/PUC-Rio, High Performance Computing Center NACAD/COPPE/UFRJ, and the Laboratory of Computational Methods in Engineering of the Program of Civil Engineering LAMCE /COPPE/UFRJ. CESUP/UFRGS is gratefully acknowledged for the computer time provided in the CRAY T90. This work was partially supported by CNPq grants N^{o} 150039/01-8(NV).

REFERENCES

Coutinho A.L.G.A., Martins M.A.D., Alves J.L.D., Landau L., and Moraes A. 2001. Edge-based finite element tecniques for nonlinear solid mechanics problems, *Int. J. for Numerical Methods in Engineering*, 50 (9), 2050-2068.

Fairbairn E.M.R., Ebecken N.F.F., Goulart E. and Paz C.N.M. 1998. Probabilistic modelling of Concrete cracking using Monte Carlo and Neural networks to solve the inverse problem, *Computational Structures Technology 1998 & Engineering Computational Technology 1998 (Edinburgh), Editor: B.H.V. Topping, Advances in Engineering Computational Technology – Neural computing for engineering computations*, ISBN 0-948749-55-5, 215-220.

Fairbairn E. M. R., Paz C. N. M., Ebecken N. F. F., and Ulm F-J., 1999. Use of neural network for fitting of probabilistic scaling model parameter, *Int. J. Fracture*, 95, 315-324.

Fairbairn E. M. R., Ebecken N. F. F., Paz C. N. M. and Ulm F-J., 2000a. Determination of probabilistic parameters of concrete: solving the inverse problem by using artificial neural networks, Computers and Structures, 78, 497-503.

Fairbairn E. M. R., Debeux V.J.C., Paz C. N. M. and Ebecken N. F. F. 2000b. Applications of probabilistic Aproach to the Analisysis of gravite Dam Centrifuge, *Test, 8th ASCE Specialty Conference on probabilistic Mechanics and Structural Reability. 24-26 July, USA, CD ROMM Proceedings* PMC 2000-216.

Hughes T. J. R. 1987. Algorithms for parabolic problems, Element-by-element (EBE) implicit methods. In: The Finite element method analysis, Chaper 8, p 483-489, *New Jersey*: Prentice – Hall.

Kelley C. T. 1995. Iterative Methods for Linear and Nonlinear Equations, Frontiers in applied mathematics, SIAM ??Society for Industrial and Applied Mathematics, Philadelphia.

Paz C. N. M. 2000. Development and Implementation Probabilistic Model for 2D and 3D Discrete Cracking Concrete in Parallel Computing, D.Sc. Thesis, Dept, of Civil Engineering, COPPE/UFRJ, Federal University of Rio de Janeiro, Brazil [in Portuguese] .

Paz C. N. M., Martha L.F., Fairbairn E. M. R., Alves J.L.D., Ebecken N. F. F. and Coutinho A.L.G.A. 2002a. Parallel implementation and Development of a probabilistic Model for 3D Discrete Concrete Cracking", *WCCM Fifth Congress on Computational Mechanics July 7-12, Vienna, Austria Eds: H.A. Mang, F.G. rammerstorfer, J. Eberhardsteiner.* ISBN 3-9501554-0-6. http://wccm.tuwiwn.ac.at.

Paz C. N. M., Martha L.F., Fairbairn E. M. R., Alves J.L.D., Ebecken N. F. F. and Coutinho A.L.G.A. 2002b. Probabilistic model 3D discrete cracking concrete in parallel computing, *VECPAR 2002, 5th International meeting, high performance computing for computational science, University of Porto FEUP, Porto, Portugal June 26-28.*

Press W. H., Teukolski S., Vetterling W.T. and Flannery B. 1992. Numerical Recipes, *Cambridge University Press*,.

RossP. i, X. Wu, Maou F. le, and Belloc A. 1994a. Scale effect on concrete in tension, *Materials and Structures*, 27 (172), 437-444.

Rossi P., Wu X, Maou F. le, and Belloc 1994b. Scale effect on concrete in tension , *Materials and Structures,* 27 (172), 437-444.

Rossi P., Ulm F.-J., and Hachi F. 1996. Compressive behavior of concrete: physical mechanisms and modeling, *Journal of Engineering Mechanics ASCE*, 122 (11), 1038-1043.

Rossi P. and Ulm F.-J. 1997. Size effects in the biaxial behavior of concrete: physical mechanisms and modeling, *Materials and Structures*, 30 (198), 210-216.