Application of parallel computation to material models with a large number of internal variables

Frédéric Feyel * — Georges Cailletaud ** François-Xavier Roux ***

** ONERA DMSE/LCME*
29, avenue de la Division Leclerc, F-92320 Châtillon cedex

*** Centre des Matériaux de l'Ecole des Mines, URA CNRS 860*
BP 87, F-91003 Evry cedex

**** ONERA CHP, 29, avenue de la Division Leclerc, F-92320 Châtillon cedex*

ABSTRACT. *The finite element code ZéBuLoN has been parallelized using the FETI subdomain decomposition method for the use of non-linear mechanical behavior models which require a high number of internal variables. This paper describes the chosen parallelization technique, the polycrystalline model – a non-linear model requiring a large number of internal variables developed at the Centre des Matériaux de l'Ecole des Mines de Paris – and shows some tests which bear out the efficiency of the promoted methods for the computation of tridimensional complex structures involving a strong non-linear behavior.*

RÉSUMÉ. *Le code de calcul par éléments finis ZéBuLoN a été parallélisé à l'aide de la méthode de décomposition de domaines FETI en vue de l'utilisation de modèles de comportement non linéaires à grand nombre de variables internes. Cet article expose la technique de parallélisation choisie, le modèle polycristallin – un modèle non linéaire à grand nombre de variables internes développé au Centre des Matériaux de l'Ecole des Mines de Paris – et montre quelques exemples de calcul qui mettent en évidence l'efficacité des méthodes de calcul parallèle couplées à des modèles de comportement sophistiqués pour le calcul de structures tridimensionnelles induisant des comportement fortement non linéaires.*

KEY WORDS: *finite elements, parallel computation, viscoplasticity, polycrystalline model, FETI method.*
MOTS-CLÉS : *éléments finis, calcul parallèle, viscoplasticité, modèle polycristallin, méthode FETI.*

Revue européenne des éléments finis. Volume 7 - n° 1-2-3/1998, pages 55 à 72

1. Introduction

The continuous increase of the power of the computers allows the engineer to enrich the geometrical description of the structures and to use more sophisticated behavior models, increasing the number of internal variables.

Classic finite element codes contain two major stages: integration of the constitutive equations (local stage) and computation of the global tangent response of the structure. The cost of the first stage depends on the number of elements and on the material behavior while the second depends on the number of degrees of freedom in the structure and on the topology (maximal band or front width).

The parallelization of finite element codes is necessary as soon as the time needed for the solution of the «local problem» becomes significant. Let us note that parallelizing the local stage is straightforward on any type of machine, as long as, in the classical approach, the constitutive equations in each Gauss point only depend on the loading history and on some internal variables in the same point. Nevertheless, the global stage has to be parallelized too, as a sequential treatment of this problem may become predominant, after having distributed the local stage on a large number of processors.

The model class aimed in this study includes all the models using state variables to describe the hardening of the material. In the case of macroscopic models, the number of variables may be typically between one and twenty. As a matter of fact, we focus on models «with a large number of internal variables», designed to represent the behavior of polycrystalline materials. This kind of model introduces 100 to 1000 grains, each of them being represented by many variables (for instance 24 for FCC crystals). It is then shown that, combining strong integration methods at the local stage and a sub-domain method for the global stage, realistic computations can be carried out, with more than ten thousand degrees of freedom, and also some thousands of internal variables for each integration point.

The first part will describe the parallelization of the global stage (parallel linear system solving). The polycrystalline model will be showed in a second part. The last part will show, through simulation examples, the efficiency of the coupling of parallel computations and a high number internal variables models.

2. The Sub-Domain Solving Method

Let Ω be a finite element discretization of the structure, and Ω_i a partition in S sub-domains of Ω:

$$\Omega = \bigcup_{i=1}^{S} \Omega_i$$

The parallelization of the global stage is complex because it needs a global stiffness matrix, which is only defined on the whole structure.

Let $\underset{\sim}{B}_i$ be the trace operators which give the restriction on Ω of a boundary defined field on Ω_i. With such notations, the problem to be solved is the following: «find the solution of the linear tangent system $\underset{\sim}{K}\underline{q} = \underline{F}$ on Ω knowing *only* the stiffness matrices $\underset{\sim}{K}_i$ on Ω_i».

2.1. The FETI Method

The **FETI** method ([3]) is a dual method, which consists in giving new unknowns: the forces $\underline{\lambda}_i$ to be enforced on domain boundaries to ensure displacement continuity across sub-domains (primal methods would favor displacement on boundaries). These forces $\underline{\lambda}_i$ are estimated with an iterative conjugate gradient method. We have indeed the following equivalence:

$$\underset{\sim}{K}\underline{q} = \underline{F} \Leftrightarrow \begin{cases} \underset{\sim}{K}_i\underline{q}_i = \underline{F}_i + \underset{\sim}{B}_i^t\underline{\lambda}_i, \text{ local equilibrium of each sub-domain } \Omega_i \\ \sum_{i=1}^{S} \underset{\sim}{B}_i\underline{q}_i = \underline{0}, \text{ displacement continuity} \end{cases} \tag{1}$$

A specific problem is related to the method, as it produces local problems with Neumann boundary conditions (force imposed conditions). If a sub-domain does not include any boundary where the Dirichlet (displacement imposed) condition is applied, or if this condition does not lock the solid motion, the local problem may be ill-conditoned. In other words, because of the chosen splitting, it can arise that some of the following local systems are not invertible:

$$\underset{\sim}{K}_i\underline{q}_i = \underline{F}_i + \underset{\sim}{B}_i^t\underline{\lambda}_i \tag{2}$$

2.1.1. Local Neumann Problem Solving

Let $\underline{\underline{A}}$ be a non invertible matrix factorized as follows:

$$\underline{\underline{A}} = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}, \text{ with } \dim A_{11} = \operatorname{rank} A = k \tag{3}$$

In a new frame associated with the Gauss factorization of $\underline{\underline{A}}$, we can write:

$$\begin{bmatrix} I & 0 \\ -A_{21}A_{11}^{-1} & I \end{bmatrix} \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \begin{bmatrix} I & -A_{11}^{-1}A_{12} \\ 0 & I \end{bmatrix} = \begin{bmatrix} A_{11} & 0 \\ 0 & A_{22} - A_{21}A_{11}^{-1}A_{12} \end{bmatrix} \tag{4}$$

Such a transformation does not change the rank of $\underline{\underline{A}}$, thus, the rank of the right hand side matrix is equal to the rank of $\underline{\underline{A}}$, and then $A_{22} - A_{21}A_{11}^{-1}A_{12} = 0$.

This demonstrates that the columns of the following matrix $\underline{\underline{N}}$ make a base of the kernel of $\underline{\underline{A}}$:

$$\underline{\underline{N}} = \begin{bmatrix} -A_{11}^{-1} A_{12} \\ I \end{bmatrix} \tag{5}$$

With these two features, the linear system

$$\begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} \tag{6}$$

can be «inverted», using a Gauss pivoting associated with A_{11} as following ($\underline{\underline{N}}\underline{\alpha}$ is a rigid body motion):

$$\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} A_{11}^{-1} & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} + \underline{\underline{N}}\underline{\alpha} \tag{7}$$

That means that a solution can be obtained with the sum of a kernel element and the pseudo-invert:

$$\underline{\underline{A}}^{+} = \begin{bmatrix} A_{11}^{-1} & 0 \\ 0 & 0 \end{bmatrix} \tag{8}$$

In practice, the decomposition (3) is obtained with a full symmetric pivoting of the matrix during its factorization. The detection of null pivots allows the building of A_{11} and to compute $\underline{\underline{N}}$.

2.1.2. FETI Method Application

Using the previous relations, the system (1) can be rewritten, using the continuity condition of $\underline{q}_i$:

$$\begin{cases} \underline{q}_i = \underset{\sim}{K}_i^{+} \left(\underline{F}_i + \underset{\sim}{B}_i^t \underline{\lambda} \right) + [N]_i \, \underline{\alpha}_i \\ [N]_i^t \left(\underline{F}_i + \underset{\sim}{B}_i^t \underline{\lambda} \right) = \underline{0} \end{cases} \tag{9}$$

$$\sum_{i=1}^{N} \underset{\sim}{B}_i \underset{\sim}{K}_i^{+} \underset{\sim}{B}_i^t \underline{\lambda} + \sum_{i=1}^{N} \underset{\sim}{B}_i \, [N]_i \, \underline{\alpha}_i = - \sum_{i=1}^{N} \underset{\sim}{B}_i \underset{\sim}{K}_i^{+} \underline{F}_i \tag{10}$$

The last two equations can be seen as an hybrid system on $\underline{\lambda}$ and $\underline{\alpha}_i$:

$$\begin{bmatrix} \underline{\underline{D}} & \underline{\underline{E}} \\ {}^t\underline{\underline{E}} & \underline{\underline{0}} \end{bmatrix} \begin{bmatrix} \underline{\lambda} \\ \underline{\alpha} \end{bmatrix} = \begin{bmatrix} \underline{c} \\ \underline{d} \end{bmatrix}, \text{ with: } \begin{cases} \underline{\underline{D}} \equiv \sum_{i=1}^{N} \underset{\sim}{B}_i \underset{\sim}{K}_i^{+} \underset{\sim}{B}_i^t \text{ dual Schur complement} \\ \underline{\underline{E}}\,\underline{\alpha} = \sum_{i=1}^{N} \underset{\sim}{B}_i [N]_i \underline{\alpha}_i \text{ jump of rigid body motions} \end{cases} \tag{11}$$

2.2. *Hybrid System Solving*

This hybrid problem is solved using a projected conjugate gradient method. This is the same as the conjugate gradient method with the extra projection $\underset{\sim}{P}\underline{g}$ of gradient $\underline{g}$ on the kernel of ${}^t\underline{\underline{E}}$ to force the constraint ${}^t\underline{\underline{E}}\,\underline{\lambda} = \underline{d}$.

The application of this projection ensures that the search direction is in the kernel of ${}^t\underline{\underline{E}}$, which automatically satisfies the constraint ${}^t\underline{\underline{E}}\,\underline{\lambda} = \underline{d}$ if $\underline{\lambda}$ is well initialized.

When convergence is achieved, $\underset{\sim}{P}\underline{g} = \underline{0}$, the gradient $\underline{g}$ is in the image of ${}^t\underline{\underline{E}}$, and a vector $\underline{\alpha}$ has been found, such as $\underline{g} = \underline{\underline{D}}\,\underline{\lambda} - \underline{c} = -\underline{\underline{E}}\,\underline{\alpha}$ with the constraint ${}^t\underline{\underline{E}}\,\underline{\lambda} = \underline{d}$: the $\underline{\lambda}$ and $\underline{\alpha}$ found by this method are then the solution of system (11).

The only difficulty is the computation of the projection operator $\underset{\sim}{P}$. It consists in finding a vector $\underline{\alpha}$ such as:

$$P\underline{g} = \underline{g} + \underline{\underline{E}}\,\underline{\alpha} \tag{12}$$

with

$${}^t\underline{\underline{E}}\,\underline{\underline{P}}\,\underline{g} = \underline{0}, \tag{13}$$

and then

$$\underset{\sim}{P} = \underset{\sim}{1} - \underline{\underline{E}}\left({}^t\underline{\underline{E}}\,\underline{\underline{E}}\right)^{-1} {}^t\underline{\underline{E}} \tag{14}$$

This computation requires a product by $\underline{\underline{E}}$ and ${}^t\underline{\underline{E}}$ and the solution of the sparse linear system ${}^t\underline{\underline{E}}\,\underline{\underline{E}}\,\alpha = -{}^t\underline{\underline{E}}\,\underline{g}$. Its dimension is the number of sub-domains. The cost of this computation is then insignificant.

2.3. *FETI Method and Machine Architecture*

As the FETI method involves a domain decomposition, it seems natural to use it on multi processor machines: all processors will process one or some sub-domains. Two type of machine can be addressed:

- *shared memory* : all processors can address the whole random access memory and a refereeing system is used to handle RAM access conflicts. The processors can «chat» together through the RAM. The developer can, in fact, ignore the multi processor structure of the machine and can rely on compilers to manage the work distribution between all processors. But as the number of processors increases, access conflicts become more frequent and the memory management may be very complex, so that the global performances of the computer are drammatically damaged.

- *distributed memory* : each processor has its own private memory and can communicate with the others through a high performance network. This last approach assumes that communication latencies are small regarding computation time (useful time). This is known as «large grain parallelism». Memory access constraint does not exist any more, but the developer has to manage himself message exchange through local network using special libraries (for instance, PVM or MPI). A local area network of workstations is a good example of such a machine.

However it remains possible to use a shared memory computer as a distributed memory one; each processor is restricted to a given memory area. The programmer choice is then no longer an architecture choice, but the choice between two programming paradigms: *vectorization* against *message passing*. FETI method is in line with message passing programming.

Only the computation of the jump of the approximated solution across domain boundaries and the computation of projection operator really needs a global cooperative work on the whole structure. Furthermore this two operations involve a computation cost which is negligible regarding the solving of local equilibrium if all sub-domain contain enough elements.

FETI method is therefore, a 100 % parallel method using message passing, which can run on all types of computer architectures.

3. Description of the Constitutive Equations

The constitutive equations used in this study are based on the local state method. This method assumes that the local thermodynamic state in a given point and a given time is *totally* defined by the knowledge of the values at this time of specific variables – the internal variables – the values of which only depend on the loading history at the chosen point. According on the number of variables, the deformation mechanisms can be described macroscopically or on a microscale.

In the following, we will be concerned with viscoplasticity in small strains, which require the introduction of the strain partition into an elastic and a viscoplastic part:

$$\underset{\sim}{\varepsilon} = \underset{\sim}{\varepsilon}^e + \underset{\sim}{\varepsilon}^p$$

The mechanical state at time t can be computed knowing the value of all internal variables at t. State laws are written in the following general form (Y represents a vector holding all internal variables):

$$\begin{aligned} \underset{\sim}{\sigma} &= f(\varepsilon, Y) \\ \dot{Y} &= g(\varepsilon, Y) \end{aligned} \tag{15}$$

It is a differential system of first order, which can be solved using a Runge-Kutta technique or Theta method when the number of variables is less than 100, and by the Runge-Kutta method only for the case of a large number of variables (several thousand). The polycrystalline model presented in the next section, respects the previous formalism.

3.1. The Polycrystalline Model

3.1.1. Constitutive Equations For One Grain

It is assumed that slip is the predominant deformation mechanism, and that Schmid's law is valid. The resolved shear stress can then be used as a critical variable to evaluate the inelastic flow. A viscoplastic framework is chosen, in order to avoid the problems related with the determination of the active slip systems in plastic models. A threshold is introduced both in positive and negative directions on each slip system: twelve octahedral slip systems will be used for FCC materials (and not twenty four). On the other hand, such a formalism leads to define two variables for each slip system, *id est* $r^s$and $x^s$for slip system s, corresponding respectively to isotropic hardening (expansion of the elastic domain), and kinematic hardening (translation of the elastic domain). A system will be active provided its resolved shear stress $\tau^s$is greater than $x^s + r^s$ or less than $x^s - r^s$ and the slip rate will be known as long as stress and the hardening variables are known. The number of variables for the single crystal is then $2N$ (with N the number of systems in the grain). The state variables used to define the evolution of $r^s$and $x^s$are the accumulated slip $v^s$for isotropic hardening and the variable $\alpha^s$for kinematic hardening.

Knowing the stress tensor applied to the grain g, $\sigma^g$the resolved shear stress for system s can be classically written according to Eqn.16, $\underline{n}^s$ and $\underline{l}^s$ being respectively, for the system s, the normal to the slip plane and the slip direction in this plane. The hardening variables $x^s$and r^scan then be expressed as a function of $\alpha^s$and v^sfollowing Eqn.17, their actual values allowing then to compute the viscoplastic slip rate $\dot{\gamma}^s$, the viscoplastic strain rate tensor $\underset{\sim}{\dot{\varepsilon}}^g$ (Eqn.18), and the hardening rules (Eqn.19). The present formulation gives a saturation of the hardening in both monotonic and cyclic loading, and takes into account the interactions between the slip systems, through h_{rs} matrix, as in [24, 11]. Nine material dependent coefficients are involved in the model $(K, n, c, d, R_0, Q_1, b_1, Q_2, b_2)$.

$$\tau^s = \underset{\sim}{\sigma}^g : \underset{\sim}{m}^s = 1/2\,\underset{\sim}{\sigma}^g : (\underline{n}^s \otimes \underline{l}^s + \underline{l}^s \otimes \underline{n}^s) \tag{16}$$

$$x^s = c\alpha^s \qquad r^s = R_0 + Q_1 \left(\sum_r h_{rs} \left\{ 1 - e^{-b_1 v^r} \right\} \right) + Q_2 \left\{ 1 - e^{-b_2 v^s} \right\} \tag{17}$$

$$\dot{\gamma}^s = \dot{v}^s sign\,(\tau^s - x^s) \qquad \underset{\sim}{\dot{\varepsilon}}^g = \sum_s \underset{\sim}{m}^s \dot{\gamma}^s \tag{18}$$

$$\dot{v}^s = \frac{\langle |\tau^s - x^s| - r^s \rangle^n}{K} \qquad \text{with } \langle x \rangle = Max(x, 0), \qquad \dot{\alpha}^s = \dot{\gamma}^s - d\alpha^s \dot{v}^s \tag{19}$$

Such a formulation is an extension of the classical crystallographic approach for single crystal modeling in plasticity or in viscoplasticity (see for instance [23, 16, 1]). Due to the saturation of the hardening and the presence of kinematic hardening, it is valid for the simulation of cyclic loadings. It has been extensively used for single crystal modeling, including Finite Element simulations, for copper [18] or nickel base super alloy [17, 21, 20].

3.1.2. Stress Concentration Rule

In a polycrystalline aggregate, one grain may be characterized by its shape, size, crystallographic orientation, location with respect to the surface of the material, *etc* ... In the present modeling, we just retain the crystallographic orientation, and put in the same *crystallographic phase* all the grains having the same orientation. The alloy is then considered as a n–phase material, each phase being defined by a set of Euler angles, and the model will then be used to describe the mean behavior of all of them. Such an approach is valid provided the grain size remains small with respect to the volume element considered in the future F.E. calculations, that is each elementary volume around a Gauss point must contain a sufficient number of grains. In these conditions, the concentration rule will define the way for computing the local stress tensor $\underset{\sim}{\sigma}^g$, which will be uniform in the phase, starting from the macroscopic stress tensor $\underset{\sim}{\sigma}$.

The self–consistent scheme is a good solution to schematically represent the phase interaction. The first papers on the subject were devoted to the definition of the elastic accommodation [15] and the plastic accommodation [12]. A pure viscoplastic formulation was also used in the past (see for instance [13, 19]), but the elastoviscoplastic approximation is still in construction. Anyway, it can be observed that, for the simple case where elasticity is uniform in the phases, the relation (20) summarize the results given by several models, according to the definition of α, with a specific mention to $\alpha = 1$ [15], and to α defined as

Variable	# of reals
Cumulated viscoplastic strain	1
$\underset{\sim}{E}_v$	6
$\underset{\sim}{\beta}^g$	$6G$
$\underset{\sim}{\varepsilon}^g$	$6G$
x^s	$12G$
r^s	$12G$
Total	$36G+7$

Table 1. *Summary of the variables of the polycrystalline model.*

a decreasing function of the macroscopic elastoplastic secant modulus, starting from 1 at the onset of plastic flow, as proposed in [2]. This last formulation is equivalent to Hill's model for radial loading. It demonstrates that the corrective term in the accommodation rule must be non-linear.

$$\underset{\sim}{\sigma}^g = \underset{\sim}{\sigma} + \alpha\,\mu\left(\underset{\sim}{\varepsilon}^p - \underset{\sim}{\varepsilon}^g\right) \tag{20}$$

The model used in the present study is an evolution of a previous approach [4]. It is based on a phenomenological rule which reproduces nonlinear accommodation by means of a new variable β^gin each grain [22], which can be calibrated by means of a specific identification procedure to numerically check the self–consistent assumption. Many shapes can be chosen for purpose, the rule (21) has been successfully applied to FCC materials in the past. As a matter of fact, it superimposes a sort of non-linear kinematic hardening [7] and a linear kinematic hardening (provided the value of δ is non zero) at the intergranular level. The D and δ values must not be considered as coefficients. They are determined after the local behavior is fixed. Writing this relation imposes to keep in memory one tensor in each grain, and to consider also (Eqn.22) the mean value of all these tensors in the aggregate $\underset{\sim}{\beta}$, computed using the volume fraction f^g of each grain.

$$\begin{cases} \underset{\sim}{\sigma}^g &= \underset{\sim}{\sigma} + \mu\left(\underset{\sim}{\beta} - \underset{\sim}{\beta}^g\right) \\ \dot{\underset{\sim}{\beta}}^g &= \dot{\underset{\sim}{\varepsilon}}^g - D\left\{\underset{\sim}{\beta}^g - \delta\underset{\sim}{\varepsilon}^g\right\}\sum_g f^g \sum_s \dot{v}^s \end{cases} \tag{21}$$

$$\underset{\sim}{\varepsilon}_p = \sum_g f_g \underset{\sim}{\varepsilon}^g \qquad\qquad \underset{\sim}{\beta} = \sum_g f^g \underset{\sim}{\beta}^g \tag{22}$$

3.1.3. Summary of the Model

Table 1 summarizes all the internal variables of the model. If G is the grain number of each element volume, this model requires $36G + 7$ real variables

(note that, if δ is equal to zero, it is not necessary to keep $\underset{\sim}{\varepsilon}^g$ in memory, so that this number is only $30G + 7$). The corresponding differential system of $36G + 7$ (or $30G + 7$) variables has then to be solved for the evaluation of the constitutive equations. In order to have a good precision in the representation of the microstructure, the number of grains G has to be chosen large enough. A typical value in the computations is $G = 40$, the model requires then 1447 (or 1207) internal variables, that is about 11 Ko memory for each Gauss point in a finite element computation.

4. Implementation

This model was implemented in ZéBuLoN, a finite element code written in C++ and using object oriented programming. This code was parallelized using the FETI method.

The advantages of OOP were shown by [10]. We want just to emphasize two critical points:

- *Message passing library independence* : OOP provides a total independence of all architecture and message passing library. The *same* executable is used for performing both sequential and parallel computations; all message passing specific calls are grouped in a single class, the default behavior of which, is to issue an error and stop the code. In addition, different message passing libraries can be implemented in the same executable.
- *Code reuse* : as shown in section 2.1., FETI method uses a projected conjugate gradient algorithm. The computation of the projection requires the inversion of a small system the assembling of which is complex (see equation 14). This system is then solved using another conjugate gradient method (level 2) which uses the same code as the main conjugate gradient (level 1).

  Furthermore, the improvements made on solving level 2 (for instance, the projection on former computed search directions) are automatically beneficial at level 1, and *vice-versa*.

5. Validation and Examples

Before checking the scalability of the chosen FETI method, we size up, in the first section, the costs of local and global stages in sequential use.

5.1. Local Stage vs *Global Stage*

In order to study the influence of the two stages on the total costs, a simple structure with an easy rule for having an increasing number of degrees of freedom has to be chosen. We consider for that purpose a regular cubic structure split in $N = n \times n \times n$ quadratic elements with 20 nodes and 27 Gauss

n	5	6	7	8	9	10	11
time(s)	5.46	16.55	45.85	134.45	286.93	607.45	1116.07

Table 2. *Solver timing for different $n \times n \times n$ cube size.*

points. Boundary conditions are imposed, so that strain and stress fields are homogeneous.

The computations are performed on a network of SparcUltra2 workstations running at 290 MHz.

5.1.1. Sequential Case

- *Global stage*: this topology leads to the following results for the number of degrees of freedom and for the optimal front:

$$\begin{array}{ll} \text{Degrees of freedom} & D = 3\left((1+n)(2n+1)^2 + n(n+1)^2\right) \approx n^3 \\ \text{Front} & F = 3\left((1+n)(2n+1) + n(n+1) + \ldots\right) \approx n^2 \end{array} \tag{23}$$

The resolution time using the frontal method [14] is proportional to the product $D \times F^2$, that is, in this case:

$$\begin{aligned} T_{glob} \approx K\{&1215n^7 + 5670n^6 + 10908n^5 + 11151n^4 + \\ &6507n^3 + 2160n^2 + 378n + 27\} \end{aligned} \tag{24}$$

fitting (see figure 1) equation 24 with the computed values of table 2 let the proportionality constant be $K \approx 3.19 \times 10^{-}8s$.

- *Local stage*: the cube is assumed to be made of a polycrystalline material, described in the previous model. As the evaluation of material behavior is local and as the fields $\underset{\sim}{\sigma}$ and $\underset{\sim}{\varepsilon}$ remain homogeneous, this cost is equal to $27c \times n^3$, where c is the cost of the computation of the volume element. The cost of the polycrystalline model, with a Runge-Kutta integration method, is a linear function of the number of internal variables ($36G + 7$, G is the number of grains), see figure 2, as soon as this number is large enough (in fact 200 internal variables). The cost of the local stage is then:

$$T_{loc} \approx 27n^3(\alpha G + \beta) \tag{25}$$

α and β are constants, the value of which are fitted on curve 2.

$$\begin{aligned} \alpha &= 4.713 \times 10^{-3}s \\ \beta &= -1.62s \end{aligned}$$

Figure 3 shows the contour of $\rho = T_{loc}/T_{glob}$.

5.1.2. Parallel Case

Let us suppose that we have a p processor computer and an automatic splitting program which produces balanced sub-domains. We want to estimate the «user cost» of the computation, this means the computation time of the slower processor. With such conditions, the cost of the local stage is simply divided by p:

$$T^p_{loc} = \frac{T_{loc}}{p} \tag{26}$$

The cost of the global parallelized stage is not so easy to estimate: it depends on the mesh topology and the associated splitting. We can, nevertheless, build an optimistic approximation. Suppose that the cube is split into p slices of $n/p \times n \times n$ elements (this is usually a «bad» topological splitting). The FETI method requires a factorization of the stiffness matrix on each sub-domain, whose costs are $T^p_{glob} \approx DF^2 \approx n^7/p^3 \approx N^{7/3}/p$. It also requires some other operations, the cost of which are neglected regarding factorization cost, if all local problems are large enough.

Figure 4 shows the location of the line $T^p_{loc}/T^p_{glob} = 1$, for a various number of processors, p, in a plane figuring the number of variables in each Gauss point and the number of degrees of freedom.

The comparison of these two curves demonstrates the interest of parallel computation to speed up the local stage, but also the need to parallelize the global stage using the FETI method to ensure high performances while computing high structure with many degrees of freedom on massively parallel computers. For instance, if a model requiring 500 internal variables per Gauss point is used, figure 3 shows that, for a mesh with 10^5 degrees of freedom, local stage cost is equal to global stage cost in a sequential resolution. After having divided the local cost by a factor p, with a parallel computation, it is also required to treat the global cost, otherwise the computation will remain jammed by the global cost. On the other hand, if both local and global stages are parallelized, figure 4 shows that, with $p = 4$ processors, the balance between local and global stages is reached only with a mesh with more than 10^6 degrees of freedom.

5.2. The Triaxial Specimen

The computations made for the purpose of validation (section 5.3.) and the examples (section 5.4.) were performed, using a triaxial specimen designed at *Laboratoire de Mécanique et Technologie de l'Ecole Normale Supérieure de Cachan* (figure 5.1.1.). For symmetry reasons, only one eighth of the structure is meshed. This mesh has 8288 elements, 3358 nodes and 69090 Gauss points.

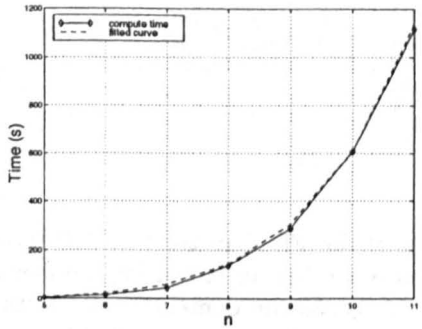

Figure 1. *Solver timing for different $n \times n \times n$ cube size.*

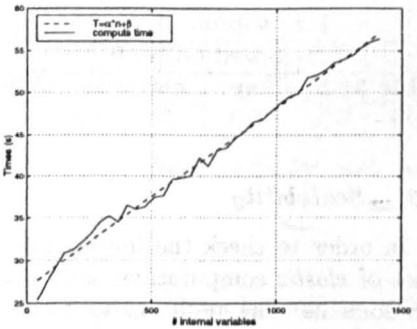

Figure 2. *Computation time of the volume element using different number of internal variables.*

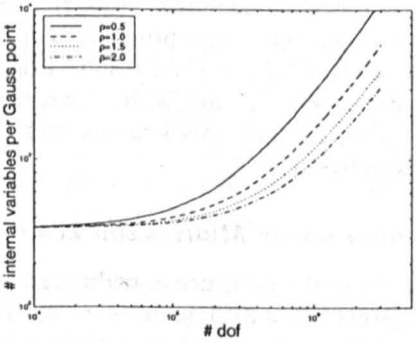

Figure 3. *Curves $T_{loc}/T_{glob} = \rho$.*

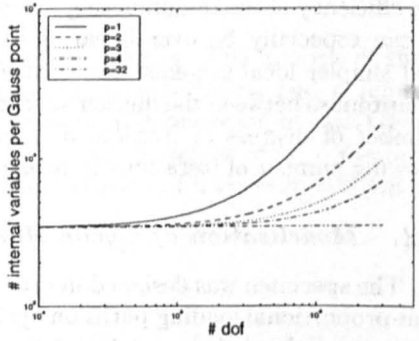

Figure 4. *Curves $T^p_{loc}/T^p_{glob} = 1$.*

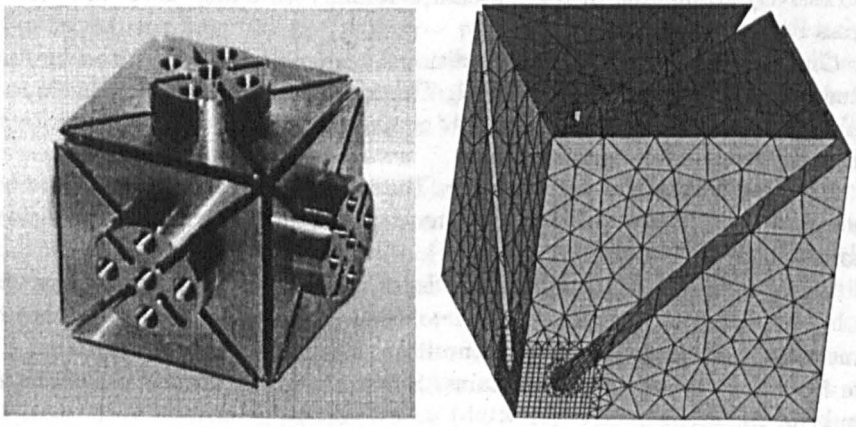

Figure 5. *Photo and mesh of the triaxial specimen.*

# proc.	7	14	16	32	64	80
Speed-up	7.0	13.3	10.9	15.2	25.0	30.4

Table 3. *Speed-up* vs *number of processors*

5.3. Scalability

In order to check the implementation of the FETI method in ZéBuLoN, a set of *elastic* computations were carried out using an increasing number of sub-domains. The mesh was split into 7, 14, 16, 32 sub-domains with splitting program ONERA-Splitmesh. These tests ran on a Paragon at ONERA.

The results in table 3 and figure 5.5. show that the FETI method in ZéBuLoN ensures the scalability: the speed-up grows linearly for a large number of processors.

This speed-up depends on the mesh topology, on the splitting quality and on the efficiency of the re-numbering scheme (modified Sloan scheme in ZéBuLoN). It can especially be over linear, if the splitting operation produces simpler and simpler local systems: the efficiency of the FETI method results from a compromise between the quicker solution of the local systems (with a decreasing number of degrees of freedom and front size) in each sub-domains and the growing number of iterations to achieve convergence.

5.4. Modelisation of Cyclic Hardening under Multi-Axial Load

The specimen was designed in order to study the influence of tridimensional non-proportional loading paths on cyclic hardening of 316L stainless steel. The specimen is loaded on a triaxial tension-compression machine, allowing the application of three independent stresses on three orthogonal axes.

A cyclic load is applied to the specimen, the three axes x, y and z being successively stimulated in tension-compression, with a step backwards to null stress between each cycle.

One can observe a significant additional hardening ([6]) which can be simulated using the polycrystalline model. The aggregate is made up with 40 grains which involves a total of $69090 \times 1447 \simeq 100$ millions internal variables for the whole mesh.

As some symmetries are present in the mesh, a natural decomposition can be found using seven sub-domains, as reported in table 5.4. (this is also a geometrically balanced decomposition).

Using such a decomposition, the local front widths are small (regarding global mesh front), and the parallel solution of the global problem takes ten times less time than the same computation using the sequential code. Figure 5.5. shows the seven sub-domains. Sub-domain number 1 is the one at the center of the mesh.

Thanks to the global parallelism (linear system solving), and due to the heavy computations needed for the integration of the polycrystalline model,

	Whole mesh	1	2	3	4	5	6	7
# of nodes	3358	271	603	625	603	625	603	625
# of elements	8288	245	1326	1355	1326	1355	1326	1355
front	1104	132	177	189	177	183	177	183

Table 4. *Number of elements, nodes and maximal front width for each sub-domain.*

99 % of CPU time is used to perform the local integration. It could be 100% parallel if the load was balanced.

The computation were performed at *Ecole des Mines de Paris* on a SP2 using seven 39H-66 MHz processors. It lasted about ten CPU days for three cycles.

5.5. Result Analysis

One can find in [9] a detailed analysis of the results. The interest of such computations (which combines a fine geometrical description and a refined behavior law) is to give an access to microstructural information in each Gauss point of the mesh, in addition to classical results (isovalues of $\underset{\sim}{\sigma}$, $\underset{\sim}{\varepsilon}$ figure 9, Mises). For instance, histogram 8 shows the evolution of active slip systems in the center of the specimen.

6. Conclusion

Some simple structure computations, using polycrystalline model, were previously tested ([5]). They remain limited, as the classical machine architecture is not powerful enough in terms of available memory and speed.

Using parallel computers in non-linear structural computations gives an access to realistic simulations. It is now possible to associate a fine description of geometry (*id est* a fine F.E. discretization) and sophisticated non-linear behavior models, allowing a fine modeling of the deformation process. A particular model is presented in this paper, but the propose technique for parallel computation is model-independent.

There are still some problems left. The most important is the load balance: as the evaluation of the constitutive equations depends on the local viscoplastic state, the time needed for each group of element will change according to the loading history. In the present implementation, a group of element is statically attached to a processor, so that all the processors have to wait for the slower one, leading to load imbalance when viscoplastic redistributions are present. Dynamic load balancing is needed to solve this problem. A simple way to do this is to move internal variables before (data migration) and after (back to owner) behavior evaluation at the local stage. Since these variables are stored in a separated, well defined, class, this is quite easy.

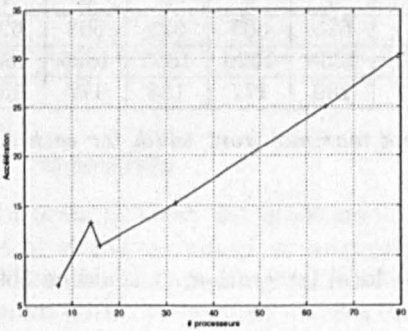

Figure 6. *Speed-up* vs *number of processors*

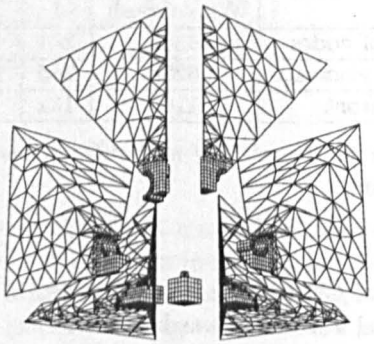

Figure 7. *7 sub-domains decomposition.*

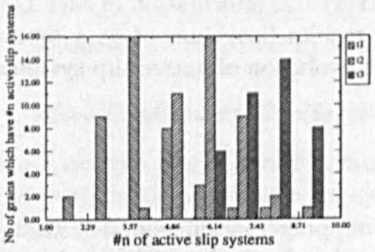

Figure 8. *Number of active slip systems, at third cycle, maximum of branch x.*

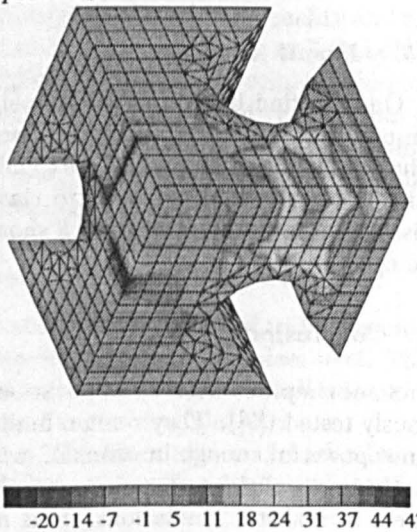

Figure 9. *Isovalues of stress σ_{33}, at third cycle, maximum of branch x.*

On the other hand, different assumptions can be used for describing the local microstructure. The present model was fitted for polycrystalline materials, that is for random micro-structure. Fully ordered micro-structures (composite materials) can also be investigated. In this case, a new mesh is introduced in each Gauss point, to describe the microstructure, using the so-called FE^2 method (two scales FE method) will respect both the local behavior of each phase and the local load balance, allowing the use to apply delamination and damage processes [8].

References

[1] R. J. Asaro. Crystal plasticity. *J. Appl. Mech.*, 50:921–934, 1983.

[2] M. Berveiller and A. Zaoui. An extension of the self-consistent scheme to plastically flowing polycrystal. *JMPS*, 6:325–344, 1979.

[3] F.-X. R. C. Fahrat. Implicit parallel processing in structural mechanics. *CMA*, 2(1), 1994.

[4] G. Cailletaud. A micromechanical approach to inelastic behaviour of metals. *IJP*, 8:55–73, 1992.

[5] G. Cailletaud and P. Pilvin. Utilisation de modèles polycristallins pour le calcul par éléments finis. *REEF*, 3(4):515–541, 1994.

[6] S. Calloch and D. Marquis. Additional hardening due to tension–torsion nonproportional loadings: Influence of the loading path shape. STP 1280, pages 113–130. ASTM, 1997.

[7] J.-L. Chaboche. Constitutive equations for cyclic plasticity and cyclic viscoplasticity. *IJP*, 5:247–302, 1989.

[8] F. Feyel. *Parallélisme et approches multi-échelles en mécanique des matériaux.* PhD thesis, École Normale Supérieure de Mines de Paris, 1999. *In progress.*

[9] F. Feyel, S. Calloch, D. Marquis, and G. Cailletaud. F.e. computation of a triaxial specimen using a polycrystalline model. *CMS*, accepted, 1997.

[10] R. Foerch. *Un environnement orienté objet pour la modélisation en mécanique des matériaux.* PhD thesis, École Nationale Supérieure des Mines de Paris, 1996.

[11] P. Franciosi. The concepts of latent hardening and strain hardening in metallic single crystals. *ACTAMET*, 33:1601–1612, 1985.

[12] R. Hill. Continuum micro–mechanisms of elastoplastic polycrystals. *JMPS*, 13:89–101, 1965.

[13] J. HUTCHINSON. Elastic–plastic behaviour of polycrystalline metals and composites. *PRS*, A319:247–272, 1966.

[14] M. IRONS AND D. OWEN. *An introduction to finite element computations.* Pineridge Press, 1979.

[15] E. KRÖNER. Zur plastischen verformung des vielkristalls. *ACTAMET*, 9:155–161, 1961.

[16] J. MANDEL. Une généralisation de la théorie de la plasticité de w.t. koiter. *Int. J. Solids Struct.*, 1:273–295, 1965.

[17] L. MÉRIC AND G. CAILLETAUD. Single crystal modeling for structural calculations. part 2: Finite element implementation. *JEMT*, 113:171–182, 1991.

[18] L. MÉRIC, G. CAILLETAUD, AND M. GASPÉRINI. F.e. calculations of copper bicrystal specimens submitted to tension–compression tests. *ACTAMET*, 42(3):921–935, 1985.

[19] A. MOLINARI, G. R. CANOVA, AND S. AHZI. A self–consistent approach to the large deformation polycrystal viscoplasticity. *ACTAMET*, 35:2983–2994, 1987.

[20] D. NOUAILHAS AND G. CAILLETAUD. *IJP*, 11(4):451–470, 1985.

[21] D. NOUAILHAS, J.-P. CULIÉ, G. CAILLETAUD, AND L. MÉRIC. F.e. analysis of the stress–strain behaviour of single–crystal tubes. *EJMS*, 14(1):137–154, 1985.

[22] P. PILVIN. The contribution of micromechanical approaches to the modelling of inelastic behaviour. In A. Pineau, G. Cailletaud, and T. Lindley, editors, *Fourth Int. Conf. on Biaxial/multiaxial Fatigue*, volume 1, pages 31–46. ESIS, may 31–june 3 1994. Saint–Germain, France.

[23] G. TAYLOR. Plastic strain in metals. *J Inst. Metals*, 62:307–324, 1938.

[24] T. B. U.F. KOCKS. Latent hardening in aluminium. *ACTAMET*, 14:87–98, 1966.