

VIPLAS: FLUIDOS VISCOPLASTICOS POR ELEMENTOS FINITOS

Margarita M. Ruda
Sergio Felicelli
Daniela Arnica
Gustavo Buscaglia

Sección Mecánica Computacional - DIN
Centro Atómico Bariloche C.N.E.A
8400 Bariloche (R.N.)

RESUMEN

Se discute la implementación del programa VIPLAS que utiliza el método de elementos finitos para resolver, por un algoritmo de lagrangeano aumentado, las ecuaciones de flujo en estado estacionario de un fluido viscoplastico incompresible. El potencial de disipación viscoplastica interna del material debe ser una función convexa, pero no necesariamente diferenciable, del tensor velocidad de deformación. Hay una versión para casos bidimensionales y otra para simetría de revolución.

ABSTRACT

Code VIPLAS uses finite elements and augmented lagrangean techniques to solve numerically the quasistatic flow equations of incompressible viscoplastic fluids. The internal viscoplastic dissipation potential is a convex, but not necessarily differentiable, function of the material deformation rate. Plane and axisymmetric versions are available.

INTRODUCCION

La determinación por métodos numéricos de campos de velocidad, presión y temperatura de fluidos no-newtonianos, sometidos a fuerzas externas, es de considerable interés práctico en varias industrias: cerámicos, metales, polímeros, pinturas, alimentos, etc. Los resultados se pueden aplicar al diseño de equipos, o a la detección de las condiciones óptimas de operación.

En general, la caracterización del comportamiento no-newtoniano por medio de una ecuación constitutiva lleva a sistemas de ecuaciones diferenciales suficientemente no lineales como para que su resolución numérica no sea trivial. En particular, el método de elementos finitos presenta como ventajas una gran facilidad para adaptarse a geometrías complejas y un manejo simple de las condiciones de contorno, de modo que el mismo programa puede usarse en distintos tipos de problemas con modificaciones mínimas.

En este trabajo nos concretaremos al caso de fluidos viscoplásticos, sin efectos elásticos, incompresibles, cuya ecuación constitutiva implique la existencia de un potencial de disipación interna que sea un funcional convexo, continuo y coercitivo del tensor velocidad de deformación. Despreciando términos de inercia y usando el método de elementos finitos, el problema se reduce a la minimización con restricciones de un funcional convexo no necesariamente diferenciable. La resolución numérica del problema se ha hecho según un algoritmo propuesto por P. Le Tallec (1986).

El trabajo está organizado de la siguiente manera: primero definiremos el modelo matemático, tanto el continuo como su discretización por elementos finitos mixtos de orden 1. A continuación reformularemos el problema discreto con técnicas de lagrangeano aumentado y describiremos luego la implementación del algoritmo antes mencionado dentro del sistema SDP (P. de Gouvea, 1986). Por último mostraremos ejemplos de prueba realizados con el programa.

MODELO MATEMATICO: FORMULACION VARIACIONAL

En el caso quasi-estático, el campo de velocidades v y de un material viscoplástico que fluye en un dominio Ω de $R(N)$ ($N=2$ o 3) está determinado por el principio de trabajos virtuales y la ley constitutiva del material, junto con el requerimiento de incompresibilidad. Si la ley constitutiva establece que el campo

del tensor de tensiones de Cauchy, sumado al campo de la presión hidrostática, pertenece al subgradiente de un potencial de disipación interna \mathcal{D} , donde \mathcal{D} es función convexa del gradiente simétrico del campo de velocidades, es posible probar que el conjunto de ecuaciones antes nombradas equivale a la siguiente formulación variacional (extremal):

$$\begin{aligned} \text{Min } J(v) = & \int_{\Omega} \mathcal{D}(Dv) \, dx - \int_{\Omega} f \cdot v \, dx - \int_{\Gamma^2} g \cdot v \, da \\ & \text{sobre } K = \{ v \in V^{1,p}(\Omega) : \text{div } v = 0, v \cdot \nu = 0 \text{ en } \Gamma^1 \} \end{aligned} \quad (1)$$

donde :

- Ω = dominio en \mathbb{R}^N ($N = 2$ o 3)
- Γ^1 = región del borde con condiciones de Dirichlet
- Γ^2 = región del borde con condiciones de Neumann
- v = campo vectorial de velocidades
- f, g = fuerzas externas de volumen y de superficie, respectivamente.
- $Dv = 1/2 [\nabla v + (\nabla v)^T]$ en $\mathbb{R}^{N \times N}$
- $V^{1,p}(\Omega)$ = espacio de Sobolev de campos vectoriales v definidos en Ω tales que $|v|^p$ y $|\nabla v|^p$ sean integrables sobre Ω

Algunos ejemplos del potencial de disipación \mathcal{D} :

Material de Norton:

$$\mathcal{D}(Dv, x) = 1/p [k(x)/2]^p |Dv|^p$$

(para fluido de Newton $p=2$, $k = \nu$)

Material de Bingham:

$$\mathcal{D}(Dv, x) = \sqrt{2} s(x) |Dv| + \nu(x) |Dv|^2$$

(para fluido de Newton $s=0$, $\nu = \nu$)

La existencia y unicidad de la solución está probada por Le Tallec (1986)

PROBLEMA DISCRETO:

El dominio se discretiza con elementos finitos mixtos de orden uno, como el mostrado en la figura 1. para el caso plano bidimensional. Dicho elemento cumple con la condición de Brezzi, y es muy sencillo de utilizar. El espacio de presiones aproximadas es el espacio de funciones escalares continuas cuya restricción a cada triángulo es un polinomio de grado uno. Cada elemento triangular de presión está dividido en cuatro subtriángulos de igual área, en cuyos vértices se determina el campo de velocidades. El gradiente simétrico del campo de velocidades termina siendo constante en cada subtriángulo.

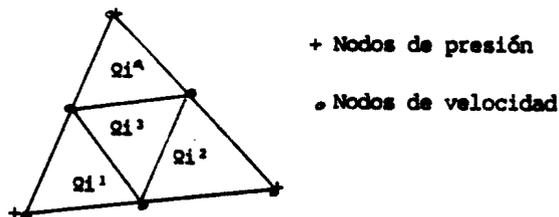


Fig. 1: Descomposición del elemento Ω_i en cuatro subtriángulos

El problema (1) discretizado se transforma en un problema de minimización de una función J_h convexa, con restricciones lineales, en un número finito de incógnitas. Ellas son: el campo de velocidades discretizadas y el campo de presiones discretizadas. J_h es convexa, pero no es cuadrática (salvo en el caso de un fluido newtoniano), y puede no ser diferenciable (como en el caso de un material de Tresca).

Lo tradicional ha sido resolver el problema (1) en forma particular para cada tipo de potencial que se conoce. Así, en el caso de un material de Norton (o ley de la potencia), se suele trabajar con el concepto de viscosidad generalizada; el sistema de ecuaciones resultante puede resolverse por el método de Newton-Raphson, ya que \mathcal{S} es doblemente diferenciable [2], o por el algoritmo iterativo propuesto por Tanner [3]. En este último caso se fija la viscosidad, se resuelve el campo de velocidades

incompresible, se calcula el gradiente simétrico del campo de velocidades, se ajusta la viscosidad y se itera hasta convergencia en velocidades y presiones.

En este trabajo, en cambio, se ataca directamente el problema (1) discretizado. Se crea una nueva variable \underline{H} , que es una matriz simétrica de $N \times N$ constante en cada elemento de velocidades. Dicha variable agrega una nueva restricción al problema: en la solución, \underline{H} será igual al gradiente simétrico del campo de velocidades. Esta restricción se trata con técnicas de lagrangeano aumentado; el sistema a resolver es ahora el siguiente:

Hallar el punto de ensilladura $((v, H), \Lambda)$ de:

$$S((v, H), \Lambda) = F(H) + G(v) + 1/2 R ||Dv - H||^2 - \langle \Lambda, Dv - H \rangle \quad (2)$$

sobre $K_h \times Y_h \times Y_h$

donde: $K_h = \{v : v \in V_h, \int_{\Omega} p \operatorname{div} v \, dx = 0 \text{ para todo } p \in P_h\}$

$P_h = \{p \in C^0(\Omega) : p|_{\Omega_i} \text{ polinomio de grado 1}\}$

$V_h = \{v \in C^0(\Omega) : v = v^0 \text{ en } \Gamma^1, v|_{\Omega_{ij}} \text{ polinomio de grado 1}\}$

$Y_h = \{D : D : \Omega \rightarrow R, D' = D, D|_{\Omega_{ij}} \text{ polinomio de grado 0}\}$

En el punto de ensilladura, \underline{v} solución es el campo de velocidades de divergencia nula, \underline{H} es el gradiente simétrico de dicho campo y $\underline{\Lambda}$ es el desviador de tensiones de Cauchy con signo opuesto; S tiende a J_h [1].

Lo interesante de este enfoque es que se dividieron las dificultades:

- la restricción de la divergencia nula sólo afecta a \underline{v}
- la no-linealidad de la ecuación constitutiva sólo afecta a \underline{H} , y éste es constante en cada elemento de velocidad. Esto implica por un lado, que podría resolverse esta parte del problema en paralelo; por el otro, que cualquier variación en la expresión del potencial puede tratarse fácilmente dentro de cada elemento.

Lo poco interesante es que ahora hay más variables: a los campos de velocidades y de presiones discretizados, hay que añadir el gradiente simétrico del campo de velocidades H , y el multiplicador de Lagrange Λ .

ALGORITMO DE RESOLUCION:

El algoritmo utilizado es de tipo Uzawa, para poder aprovechar la descomposición del problema ya mencionada. La resolución del primal se hace por relajación en bloques. Pruebas de convergencia pueden leerse en Le Tallec (1986).

El algoritmo es el siguiente:

```
 $\Lambda, H$  dados.  $H = H$   
0 j-1 0,k j-1  
  
for j = 0,n do:  
  for k = 1,m calcular  $v = v, H = H$  ; do:  
    j m j m  
  
     $\mathcal{E}(v, \dots) < \mathcal{E}(w, \dots)$  para todo  $w \in K_h$  (i)  
    j,k  
  
     $\mathcal{E}(., H, .) < \mathcal{E}(., G, .)$  para todo  $G \in Y_h$  (ii)  
    j,k  
  
  end do  
   $\Lambda = \Lambda - R(Dv - H)$  ,  $\Lambda \in Y_h$  (iii)  
  j+1 j j j  
  
end do
```

El punto (i) es la minimización de un funcional cuadrático con restricciones lineales, numéricamente similar a un problema de Stokes. El punto (ii) es una minimización elemento a elemento.

La convergencia se mide con el siguiente criterio:

$$\|Dv - H\| / \|H\| < \epsilon$$

A continuación mostraremos cómo se resolvieron los subproblemas (i) e (ii) en el programa VIPLAS.

1. Resolvedor rápido de Stokes.

Como "problema de Stokes" nos referimos a la siguiente formulación variacional [2], [3]:

$$\begin{aligned} \int_{\Omega} Du:Dw \, dx - \int_{\Omega} p \operatorname{div} w \, dx &= \int_{\Omega} f \cdot w + \int_{\Gamma^2} g \cdot w \, da \\ \int_{\Omega} q \operatorname{div} u \, dx &= 0 \end{aligned} \quad (3)$$

Desarrollando el lagrangeano aumentado de la ecuación (2) con \underline{H} y $\underline{\Lambda}$ constantes, el problema (i) es un problema de Stokes de la siguiente forma:

$$\begin{aligned} R/2 \int_{\Omega} Du:Dw \, dx - \int_{\Omega} p \operatorname{div} w \, dx &= \\ \int_{\Omega} f \cdot w + \int_{\Omega} (R \underline{H} + \underline{\Lambda}):Dw \, dx + \int_{\Gamma^2} g \cdot w \, da \\ \int_{\Omega} q \operatorname{div} u \, dx &= 0 \end{aligned} \quad (4)$$

Las ecuaciones discretizadas por elementos finitos de este problema, en los casos plano y axisimétrico, forman el siguiente sistema de ecuaciones lineales por bloques:

$$\begin{pmatrix} A1 & B1' \\ & A2 & B2' \\ B1 & B2 \end{pmatrix} x = \begin{pmatrix} v1 \\ v2 \\ p \end{pmatrix} = \begin{pmatrix} b1 \\ b2 \\ c \end{pmatrix} \quad (5)$$

donde $v1$ y $v2$ se refieren a las velocidades según x e y (r y z respectivamente) y p a la presión. Para resolverlo usamos un algoritmo de gradientes conjugados sobre la presión (Glowinski, 1984), que resultó necesitar mucho menos memoria y ser más rápido en un factor de 10 que la subrutina MA18 de Harwell. El algoritmo

aprovecha el hecho de que A_1 y A_2 son matrices simétricas, definidas positivas, para triangularizar por bloques el sistema descrito más arriba, con lo que se llega al siguiente sistema lineal en presiones:

$$\begin{aligned}(B_1 A_1^{-1} B_1')p &= B_1 A_1^{-1} b_1 - c \\(B_2 A_2^{-1} B_2')p &= B_2 A_2^{-1} b_2 - c\end{aligned}\tag{6}$$

Dicho sistema se resuelve por gradientes conjugados, como ya se ha dicho, y las velocidades se recuperan así:

$$\begin{aligned}v_1 &= A_1^{-1} (b_1 - B_1'p) \\v_2 &= A_2^{-1} (b_2 - B_2'p)\end{aligned}\tag{7}$$

A_1 y A_2 se triangularizan de una vez al comienzo del programa, por lo que en lugar de calcular sus inversas se hace retrosustitución con sus factores triangulares.

2. Optimización elemento a elemento

Desarrollando el lagrangeano aumentado de la ecuación (2) con \underline{v} y $\underline{\Lambda}$ constantes (el caso (ii)), y dado que \underline{H} , \underline{Dv} y $\underline{\Lambda}$ son constantes en cada subelemento de velocidad, tenemos que resolver en cada elemento el siguiente problema:

$$\text{Min}_{Q_i} \int (\vartheta(H) + R/2 |H-Dv| + \Lambda:H) dx \text{ sobre } K_h \tag{8}$$

cuya solución es la siguiente:

$$H = Q H_i Q' \tag{9}$$

donde Q es una matriz ortogonal de $R(N \times N)$ que diagonaliza la siguiente matriz simétrica \underline{A}_i :

$$A_i = R D_v - A \quad (10)$$

Las columnas de Q se forman con los autovectores normalizados de A_i . H_i es una matriz diagonal cuyos valores se obtienen a partir de la siguiente minimización en $R(N)$:

$$\text{Min } j(H_i) = \varnothing(H_i) + R/2 |H_i|^2 - \sum_k A_{ik} H_{ik} \quad (11)$$

Para resolver la minimización, se acota el valor de H_i y se hace una primera búsqueda con un algoritmo de sección áurea (adaptado de la subrutina ZXGSN de IMSL). Si falla, y el potencial es diferenciable, se usa el método de Newton-Raphson. Si también falla, o si el potencial no es diferenciable, hay una opción de interpolación cuadrática (adaptado de la subrutina ZXLSF, de IMSL).

IMPLEMENTACION EN EL SISTEMA SDP

Durante el año 1987 entró en vigencia el convenio Argentino Brasileño en Mecánica Computacional (CNEA-CONICET-LNCC-CNPq). Por ello decidimos utilizar el sistema SDP de desarrollo de programas de elementos finitos para implementar el algoritmo descrito más arriba. Dado que no había en el sistema brasileño ningún programa de fluidos, ni tampoco se utilizaba el sistema de almacenamiento de matrices ralas, sólo se usó del SDP original toda la parte referida a administración de memoria y manejo de vectores, con la idea de crear programas de aplicación ("aplicativos") en fluidos.

El sistema SDP es un sistema modular, con acceso a un banco de datos. Cualquier programa de elementos finitos se divide, para el sistema, en tres o más programas, cuyos resultados se intercambian por medio del banco de datos. Eso no incluye, hasta ahora, la parte de generación de mallas y de visualización de resultados, que hacemos con los programas ENREDO y PITUCO desarrollados en nuestro grupo por M. Venere [8,9].

Todas las matrices, salvo las matrices elementales, se almacenan como ralas segun la técnica desarrollada por S. Pissanetzky [7] en el Centro Atómico Bariloche. Para la triangulación de matrices simétricas, y el ensamblaje, también se usaron algunas subrutinas desarrolladas por Pissanetzky.

El esquema de cálculo es el siguiente:

- PRE : Lee todos los datos necesarios: coordenadas y conectividad de red de velocidad, de presión y de presión-velocidad; condiciones de Dirichlet y de Neumann, tipo de simetría. Este programa sirve para otros programas de fluidos.
- PRO : Calcula (y almacena) por elemento, áreas, derivadas parciales, matrices elementales según cada componente de la velocidad, y las matrices elementales presión-velocidad. Sirve para otros programas de fluidos.
- PRE-SOL: Ensambla matrices elementales en bloques (eq. 5) y segundos miembros. Triangulariza los bloques de velocidad. Sirve para otros programas de fluidos
- SOLVR : Aplica el algoritmo descripto. Sirve sólo para el programa de aplicación VIPLAS.

EJEMPLOS DE PRUEBA

El programa se corrió con varios ejemplos de prueba. Nos detendremos en la longitud de entrada. La figura 2. muestra el campo de velocidades y el de presiones obtenidos para un fluido de Norton con $k = 1/\sqrt{2}$, $p = 1.5$, con una red de 231 nodos de velocidad y 66 de presión. La figura 3. muestra la variación de la convergencia del algoritmo para distintos exponentes p , con $k = 1/\sqrt{2}$ y el factor $R = 1$. Por último, la figura 4. muestra que el algoritmo es relativamente poco sensible a la variación del factor R .

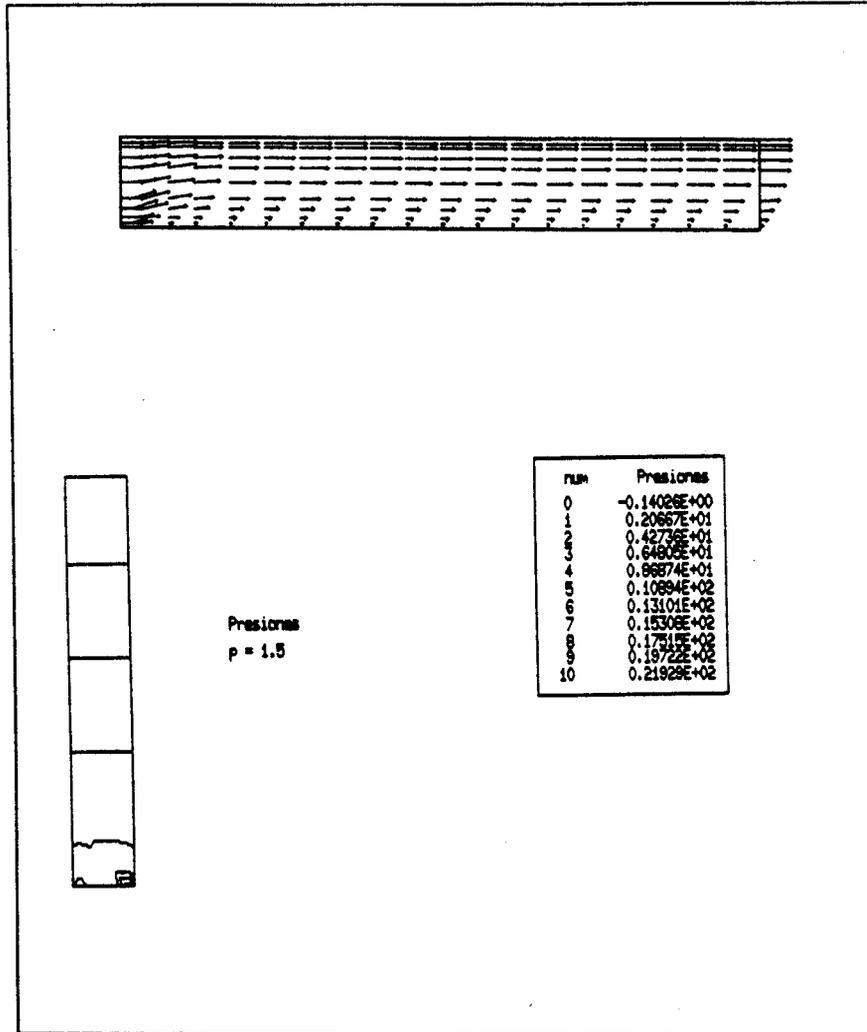


Fig. 2.: Campo de velocidades y de presiones de un fluido de Norton, $p = 1.5$, $k = 1/2$, $R = 1$. Red de 231 nodos de velocidad y 66 de presión.

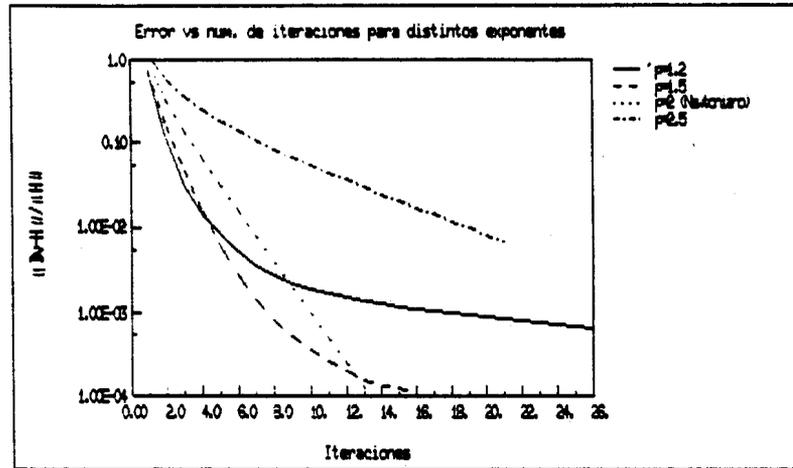


Fig. 3.: Convergencia del algoritmo para un material de Norton, distintos exponentes, $k = 1/\sqrt{2}$, $R = 1$.

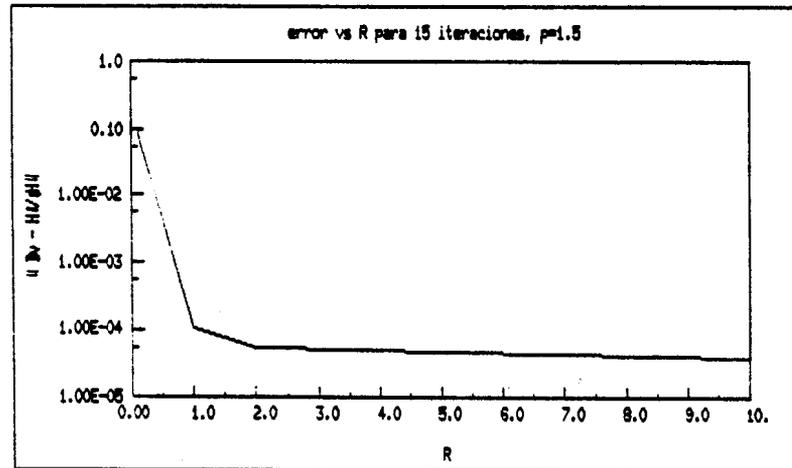


Fig. 4.: Variación de la convergencia del algoritmo con el parámetro R. Fluido de Norton, $p=1.5, k = 1/\sqrt{2}$.

Sin embargo, tenemos todavía dudas sobre lo robusto que pueda ser el código. Hay muchas variables que pueden ser ajustadas para que funcione en forma óptima, por ejemplo: la tolerancia con que se resuelve el problema de Stokes, las tolerancias de cada minimización, el número máximo de iteraciones en cada minimización, el número óptimo de ciclos internos en el algoritmo general, etc.

CONCLUSIONES

Se implementó el código VIPLAS que utiliza el método de elementos finitos para resolver numéricamente las ecuaciones de flujo en estado estacionario de un fluido viscoplástico incompresible, cuyo comportamiento siga la ley de la potencia o la de Bingham. Hay una versión para casos bidimensionales y otra para simetría de revolución.

El algoritmo utiliza técnicas de lagrangeano aumentado con lo que el problema original quedó reducido a la resolución de tres subproblemas sencillos:

- a. un programa cuadrático, para lo cual se usó un resolvidor rápido de Stokes,
- b. la minimización sin restricciones, elemento a elemento, de una función convexa en R^2 con tres opciones implementadas: sección áurea, Newton-Raphson e interpolación cuadrática,
- c. una actualización algebraica muy simple.

El código fue implementado dentro del sistema SDP (Sistema de Desarrollo de Programas), mediante el convenio CONICET-CNPq, y utiliza técnicas de matrices ralas desarrolladas en el Centro Atómico Bariloche.

Se presentaron ejemplos de prueba. El algoritmo es capaz de buenos resultados numéricos, pero al tener muchos parámetros ajustables por el usuario, no es demasiado robusto.

REFERENCIAS

- [1] Le Tallec, P. "Numerical Solution of Viscoplastic Flow Problems by Augmented Lagrangians", IMA J. of Numerical Analysis, vol. 6, 1986, págs. 185-219.
- [2] Glowinski, R. "Numerical Methods for Nonlinear Variational Problems", Springer Series in Comp. Physics, Springer-Verlag, 1984.

- [3] Crochet, M.J., Davies, A.R., Walters, K., "Numerical Simulation of Non-newtonian flows", Rheology Series, 1, Elsevier Sci. Pub., 1985.
- [4] Pereira de Gouvea, J., "SDP: un Sistema para Desenvolvimento de Programas Baseados no Metodo dos Elementos Finitos", Tesis de Doctor en Ciencias de Ing. Mécanica, PUC, Rio de Janeiro, 1986.
- [5] IMSL S-R071-E09.0, VAX, 1982.
- [6] Tanner, R.I., Nickell, R.E., Bilger, R.W. "Finite Element Methods for the solution of some incompressible non-Newtonian fluid mechanics problems with free surfaces", Comp. Meth. Appl. Mech. Eng. vol. 6, 1975, págs. 155-179.
- [7] Pissanetzky, S. "Sparse Matrix Technology", Assoc. Press, 1984.
- [8] Venere, M. "Manual del programa ENREDO", pub. interna, CAB-CNEA, 1987.
- [9] Venere, M. "Manual del Programa PITUCO", pub. interna, CAB-CNEA, 1987.