

IMPLEMENTACION DE LOS METODOS DE RIKS-WEMPNER, BATOZ-DHATT, POWELL-SIMONS Y OTROS EN EL PROGRAMA DE ELEMENTOS FINITOS F.E.A.P.

JOSE M. SANCHO AZNAL

y

JESUS ORTIZ HERRERA

*Departamento de Estructuras de Edificación,
E.T.S. de Arquitectura,
Univ. Polit. de Madrid, Madrid.*

RESUMEN

Los fundamentos teóricos del método de Riks y su utilidad para el paso de puntos límite en el análisis no lineal de estructuras, están actualmente bien establecidos, pero resta aún una importante labor de normalización y divulgación de dicho método en un formato de macroinstrucciones apropiadas, flexible para su aplicación en múltiples variantes, y que permite incluso su fácil transformación en otras técnicas también útiles para el paso de puntos límite.

SUMMARY

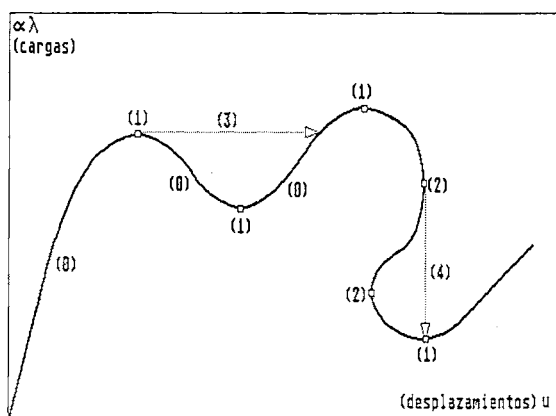
The theoretical fundaments and aplications of the Riks method are today well stablished. However a certain task is left in normalizing and divulgating the method. This paper tries to contribute to this task implementing this method in form of a set of "macrostatements" allowing the flexible use of several variants and extensions to other techniques of overcoming limit points.

INTRODUCCION

En el análisis de estructuras, tanto determinados comportamientos del material como los efectos "geométricamente" no lineales, pueden generar la aparición de "puntos límite" de diversa naturaleza en el camino del equilibrio o curva carga-desplazamiento (Figura 1), produciéndose singularidades numéricas bien conocidas, que afectan seriamente a las técnicas numéricas de resolución más usuales.

En este sentido, es conocido que en las obras clásicas de Zienkiewicz¹, Oden², etc., no se incluyen procedimientos apropiados para el paso de puntos límite, o bien hacen referencia a técnicas muy rudimentarias al respecto. Sin embargo, en los últimos años

Recibido: Febrero 1990



- (0) Camino de equilibrio estático (incluye intervalos estables e inestables)
- (1) Puntos límite controlados por carga
- (2) Puntos límite controlados por desplazamiento
- (3) Pandeo bajo control de cargas
- (4) Pandeo bajo control de desplazamiento

Figura 1. Puntos límite en la respuesta de estructuras de comportamiento no lineal.

tales procedimientos han sido desarrollados y su aplicación al análisis no lineal estático de estructuras ha quedado bien establecida.

Probablemente la más celebrada de estas técnicas sea el método de Riks^{3,4} y Wempner⁵ quienes añaden a las ecuaciones del método de Newton-Raphson una ecuación de restricción, que fija la longitud del vector incremental que une los puntos representativos de los estados inicial y final de cada incremento de carga en el espacio de cargas y desplazamientos; en contrapartida de esta nueva ecuación, el incremento del factor de carga pasa a ser considerado como una incógnita adicional; posteriormente se describen aspectos y variaciones importantes de este método que salvan algunas de las dificultades que aparecen en su formulación, y consiguen no sólo el objetivo de superar los puntos límite, sino además mejorar incluso las características de convergencia del método usual de Newton-Raphson; y todo ello sin alterar la estructura básica de este método, lo cual se traduce en modificaciones verdaderamente simples de los programas convencionales de análisis no lineal por el método de los elementos finitos.

No es la técnica de Risk-Wempner, por supuesto, la única disponible para el paso de puntos límite, pero sí es, con certeza, la más atractiva de las actualmente disponibles. Así, puede citarse la técnica de supresión de iteraciones en el entorno del punto límite, debida a Bergan⁶, que requiere incrementos de carga muy reducidos en dicho entorno, y además conlleva un cierto error o desviación del camino de equilibrio. El "método de los muelles artificiales" de Wright y Gaylord⁷ implica una elección artificial y sujeta a prueba y error, de los "muelles" apropiados. El "método de control de desplazamiento" de Argyris⁸ requiere también una elección externa del algoritmo, en este caso de una de las variables de desplazamiento como parámetro de control; por otra parte, en general los métodos de control por desplazamiento fallan en el entorno de puntos límite del tipo 2 señalado en la Figura 1, abortando el recorrido del camino de equilibrio y hurtando,

en tales casos, información relativa a la naturaleza del colapso.

A pesar de ello, las técnicas de control de desplazamiento han sido mejoradas por diversos autores y su incorporación a los programas usuales de análisis se considera conveniente (la implementación del método de Riks propuesta en este trabajo admite, como variante del usuario, el control de desplazamiento, de dos pasos, que permite mantener la matriz de rigidez original del sistema, sin reducción de su dimensión, lo cual simplifica su aplicación y elude posibles alteraciones del punto límite derivadas de la manipulación de dicha matriz.

Esta técnica de dos pasos es utilizada también en el método de Riks-Wempner como un procedimiento de eludir la gravísima complicación que supone la incorporación de una ecuación de restricción que destruye la simetría y el ancho de banda de la matriz del sistema. Otras modificaciones del método, que se comentan en apartados posteriores, consisten en la fijación de la "longitud del arco", en el espacio cargas-desplazamientos, sólo en la primera iteración de cada incremento de carga, tras lo cual el proceso iterativo continúa en un plano normal a la tangente de la curva carga-desplazamiento; opcionalmente, este plano normal puede actualizarse en cada iteración, lo cual aproxima más el proceso a la idea original de iteración "en la esfera".

Otros métodos de interés para el paso de puntos límite han sido presentados por Powell y Simons¹⁰, de los cuales el método de iteración "de trabajo constante" resulta especialmente atractivo por corresponder a un concepto físico más que a un mero artificio numérico (esta técnica puede también fácilmente incorporarse como una variante de la implementación del método de Riks aquí formulada).

A pesar de su carácter relativamente reciente, el método de Riks comienza a ser de frecuente uso en aplicaciones diversas, pudiendo mencionarse a modo de ejemplos los trabajos de Irls¹², Rothert et al.¹³, Sancho y Ortiz¹⁴, etc. Los ejemplos de prueba de la versión del método de Riks aquí propuesta han sido contrastados con ejemplos tomados de algunos de los autores aquí citados, así como de Powell y Simons¹⁰.

Los autores del siguiente trabajo no pretenden arrogarse aportaciones originales al método de Riks, puesto que consideran que sus fundamentos teóricos y posibles modificaciones y variantes están suficientemente establecidos, entre otros, en los trabajos de Crisfield^{16,17} y Ramm¹⁸. Lo que sí aporta es un intento de normalización de la implementación de este método, de manera flexible para su utilización en diversas variantes, permitiendo incluso, a nivel de usuario, su transformación en otros métodos; y de manera fácil a programas de análisis por macroinstrucciones, habiéndose elegido el programa FEAP¹ como particularmente representativo y divulgado. Los autores de este trabajo aportan incluso el listado de las reformas a efectuar en dicho programa, con el deseo expreso de que constituya una herramienta útil para la divulgación del método de Riks y de que contribuya a la normalización de su uso.

DESCRIPCION DEL METODO DE RIKS-WEMPNER

Gran parte de los procedimientos de solución de problemas lineales en el análisis de estructuras por el método de los elementos finitos se basan, como se sabe, en la aplicación incremental de las cargas actuantes, utilizándose en cada incremento de carga

métodos iterativos para aproximar el movimiento de la estructura a una situación en la que exista equilibrio. Esta condición de equilibrio se suele expresar mediante el sistema de ecuaciones:

$$\mathbf{r} = \lambda \mathbf{q} - \mathbf{f}(\mathbf{u}) = 0 \quad (1)$$

siendo $\mathbf{u}$ el vector de desplazamientos nodales

$\mathbf{f}$ el vector de fuerzas internas dependientes de $\mathbf{u}$

$\mathbf{q}$ el vector de fuerzas nodales externas

λ un factor multiplicador del vector $\mathbf{q}$

$\mathbf{r}$ el vector de fuerzas residuales o desequilibradas

La aplicación incremental de las cargas se suele llevar a cabo mediante la fijación de niveles de carga:

$$\lambda_1 \mathbf{q}, \lambda_2 \mathbf{q}, \lambda_3 \mathbf{q} \dots \lambda_i \mathbf{q}$$

en saltos:

$$\Delta \lambda_i \mathbf{q} = \lambda_i \mathbf{q} - \lambda_{i-1} \mathbf{q}$$

correspondiendo a cada nivel de carga $\lambda_i \mathbf{q}$ un estado de desplazamientos $\mathbf{u}_i$ que se puede obtener como integración de sucesivos incrementos:

$$\Delta \mathbf{u}_i = \mathbf{u}_i - \mathbf{u}_{i-1}$$

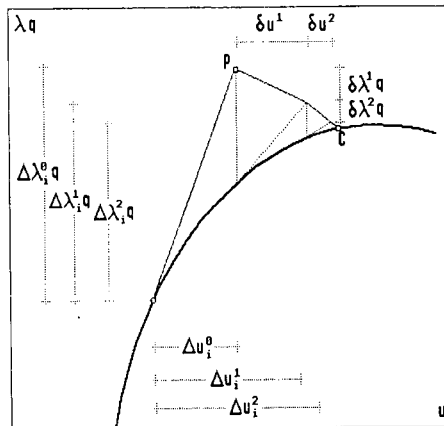


Figura 2.

En cada incremento de carga se puede obtener una primera aproximación (predicción) al estado de equilibrio (Figura 2 punto P)

$$(\Delta \mathbf{u}_i^0, \Delta \lambda_i^0)$$

mediante:

$$\Delta u_i^0 = K_{i-1}^{-1} \Delta \lambda_i^0 q \quad (2)$$

Siendo K_{i-1} la matriz de rigidez tangente en una situación anterior $i - 1$.

Si esta aproximación se introduce en las ecuaciones de equilibrio (1) se obtendrá en general un vector de fuerzas desequilibradas r distinto de cero:

$$r_i = (\lambda_{i-1} + \Delta \lambda_i^0) q + f(u_{i-1} + \Delta u_{i-1}^0) \neq 0$$

necesitándose un proceso de aproximación (corrección) a una situación de equilibrio (Figura 2 punto C) mediante un algoritmo iterativo que elimine gradualmente r . Para ello será necesario calcular nuevos valores:

$$(\Delta u_i^k, \Delta \lambda_i^k)$$

a partir de correcciones:

$$\delta u_i^k = \Delta u_i^k - \Delta u_i^{k-1}$$

$$\delta \lambda_i^k = \Delta \lambda_i^k - \Delta \lambda_i^{k-1}$$

indicándose con el superíndice k pasos en el proceso de corrección y con el subíndice i pasos en el proceso incremental.

Si, como suele hacerse en los procedimientos de iteración clásicos, el factor λ se mantiene constante, estas correcciones pueden calcularse mediante:

$$\delta u_i^k = K^{-1} r_i^{k-1} \quad \delta \lambda_i^k = 0 \quad (3)$$

siendo K la matriz de rigidez tangente que se mantiene constante en cada salto de carga (Método de Newton-Raphson modificado) o se calcula a partir de la situación anterior (Método de Newton-Raphson). A este proceso de solución en el que se especifican saltos de carga dentro de los cuales λ es constante se suele denominar "Método de control de carga".

En estructuras que presentan puntos críticos en su respuesta es posible que el sistema de ecuaciones de equilibrio (1) no tenga solución si el factor λ se mantiene constante en el proceso de corrección. En estos casos resulta imposible conocer el valor de las cargas λq correspondientes a dichos puntos críticos así como el comportamiento de la estructura en estados postcríticos.

Una manera de evitar este inconveniente consistió en tratar el factor λ como una variable más del problema, de forma que el sistema (1) tenga solución. Hará falta para ello imponer una condición adicional a las n ecuaciones de equilibrio ya que existe una nueva incógnita λ a añadir a los n grados de libertad del problema.

Si se considera un espacio (u, λ) de dimensión $n+1$, esta nueva condición puede ser entendida como una superficie que constituye el lugar geométrico de los puntos (u_i, λ_i) obtenidos en el proceso iterativo de corrección.

Por otra parte, puede definirse una curva de carga o trayectoria de equilibrio como el lugar geométrico de los puntos de equilibrio en dicho espacio (u, λ) . La condición usual del método de control de carga:

$$\lambda = \text{cte.}$$

puede entenderse en este espacio como un (hiper) plano ortogonal al eje $O-\lambda$ que puede cortar o no a la curva de carga.

Una manera de garantizar la intersección de la curva de carga con la superficie que representa la condición adicional y por tanto una manera de garantizar la convergencia del proceso iterativo de corrección, consiste en imponer como condición adicional superficies cerradas alrededor del punto de equilibrio anterior. La más sencilla de estas superficies es una (hiper) esfera de radio Δl cuya ecuación es:

$$(\Delta \mathbf{u}_i^k)^T \Delta \mathbf{u}_i^k + (\alpha \Delta \lambda_i^k)^2 = \Delta l^2 \quad (4)$$

en la que α es un coeficiente numérico introducido con el fin de homogeneizar las dimensiones y magnitud numérica de los términos de la ecuación. Esta condición fue propuesta independientemente por Riks y Wempner y ha sido el origen de distintos métodos de solución de problemas no lineales.

Es posible, de manera más sencilla, obtener convergencia en la mayoría de los casos utilizando en lugar de superficies cerradas, un (hiper) plano normal a la tangente en la curva de equilibrio en el punto de equilibrio anterior (Figura 3c) expresado por la condición de ortogonalidad:

$$(\Delta \mathbf{u}_i^0)^T \delta \mathbf{u}_i^k + \alpha^2 \Delta \lambda_i^0 \delta \lambda_i^k = 0 \quad (5)$$

Puede asegurarse la convergencia si se utiliza en cada corrección un plano distinto, normal no a la tangente inicial del salto sino a una secante actualizada sin más que utilizar la condición (Figura 3d):

$$(\Delta \mathbf{u}_i^{k-1})^T \delta \mathbf{u}_i^k + \alpha^2 \Delta \lambda_i^{k-1} \delta \lambda_i^k = 0 \quad (6)$$

Desafortunadamente, el hecho de tratar λ como una nueva incógnita en el problema, implica que el sistema de ecuaciones de equilibrio aumentado en una ecuación más del tipo (4) (5) o (6) no es simétrico ya en su conjunto, ni tampoco está configurado en banda como es usual en las formulaciones clásicas del Método de los Elementos Finitos. Ello supone, en principio, una cierta dificultad a la hora de incorporar estos métodos de corrección en los programas usuales de análisis.

Es posible superar este inconveniente si se utiliza una técnica semejante a la propuesta por Batoz y Dhatt⁹ para la resolución de problemas no lineales mediante control de desplazamiento. Mediante este procedimiento las correcciones $\delta \mathbf{u}_i^k$ en el proceso iterativo pueden ser calculadas en dos pasos mediante la descomposición:

$$\delta \mathbf{u}_i^k = **\delta \mathbf{u}_i^k + *\delta \mathbf{u}_i^k \quad (7)$$

siendo $**\delta \mathbf{u}_i^k$ la variación del vector de desplazamientos debida al proceso iterativo de Newton Raphson modificado manteniendo λ constante:

$$**\delta \mathbf{u}_i^k = \mathbf{K}^{-1} \mathbf{r}_i^{k-1}$$

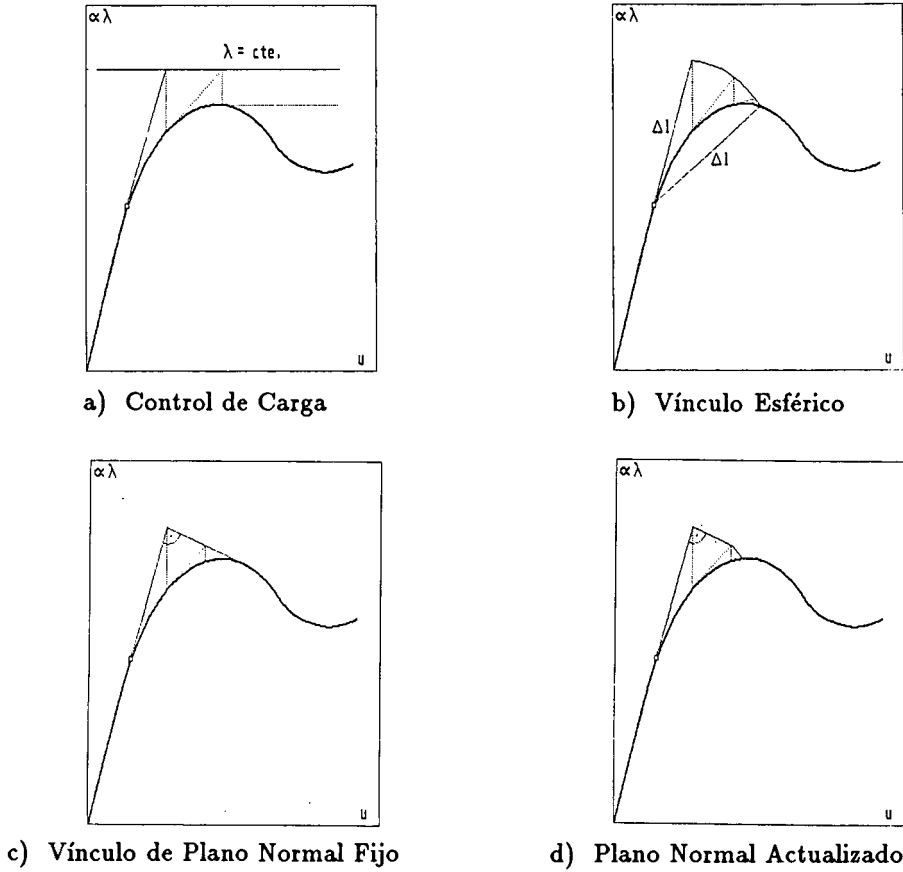


Figura 3.

y ${}^*\delta u_i^k$ la variación del vector de desplazamientos debida a la variación $\delta \lambda_i^k$ del factor multiplicador de las cargas externas aplicadas:

$${}^*\delta u_i^k = \mathbf{K}^{-1} \delta \lambda_i^k \mathbf{q}$$

Nótese que si $\mathbf{K}^{-1}$ es constante durante el proceso de corrección puede hacerse:

$${}^*\delta u_i^k = \delta \lambda_i^k {}^*u_i \quad (8)$$

siendo:

$${}^*u_i = \mathbf{K}_i^{-1} \mathbf{q}$$

La variación $\delta \lambda_i^k$ del factor de carga se puede obtener a partir de las ecuaciones (5) o (6) que reescritas teniendo en cuenta (7) y (8) toman la forma:

ecuación (5) ("plano normal constante")

$$(\Delta u_i^0)^T ({}^*\delta u_i^k) + \delta \lambda_i^k ((\Delta u_i^0)^T {}^*u_i + \alpha^2 \Delta \lambda_i^0) = 0$$

y

$$\delta\lambda_i^k = -\frac{(\Delta\mathbf{u}_i^0)^T (**\delta\mathbf{u}_i^k)}{(\Delta\mathbf{u}_i^0)^T * \mathbf{u}_i + \alpha^2 \Delta\lambda_i^0} \quad (9)$$

ecuación (6) ("plano normal actualizado")

$$(\Delta\mathbf{u}_i^{k-1})^T (**\delta\mathbf{u}_i^k) + \delta\lambda_i^k ((\Delta\mathbf{u}_i^{k-1})^T * \mathbf{u}_i + \alpha^2 \Delta\lambda_i^{k-1}) = 0$$

$$\delta\lambda_i^k = -\frac{(\Delta\mathbf{u}_i^{k-1})^T (**\delta\mathbf{u}_i^k)}{(\Delta\mathbf{u}_i^{k-1})^T * \mathbf{u}_i + \alpha^2 \Delta\lambda_i^{k-1}} \quad (10)$$

Mediante este procedimiento de dos etapas el sistema aumentado de $n+1$ incógnitas puede ser resuelto utilizando la estructura de datos y algoritmos usuales en los programas de elementos finitos.

Elección del tamaño del salto de carga

La idea de tratar el factor multiplicador de las cargas λ como una incógnita adicional en el proceso de corrección de la solución, puede además ser utilizada para decidir la estrategia de avance en el proceso incremental. De esta forma, en lugar de utilizar saltos de carga $\Delta\lambda_i^q$ determinados de antemano, puede llevarse a cabo un control del proceso de incrementación mediante otro criterio tal como fijar la longitud del vector tangente $(\Delta\lambda_i^0, \Delta\mathbf{u}_i^0)$ correspondiente a la primera aproximación dentro de cada salto mediante la ecuación (4) particularizada:

$$(\Delta\mathbf{u}_i^0)^T \Delta\mathbf{u}_i^0 + (\alpha\Delta\lambda_i^0)^2 = \Delta l^2 = \text{cte.}$$

Ya se comprende que en la práctica resulta difícil estimar un valor razonable de Δl por lo que se suele fijar un valor inicial de $\Delta\lambda_1^0$ para el primer incremento a partir del cual se calcula Δl mediante:

$$\Delta l = \sqrt{(\Delta\mathbf{u}_1^0)^T \Delta\mathbf{u}_1^0 + (\alpha\Delta\lambda_1^0)^2} \quad (11)$$

con

$$\Delta\mathbf{u}_1^0 = \Delta\lambda_1^0 * \mathbf{u}_1$$

siendo $*\mathbf{u}_1$ el vector de desplazamientos definido anteriormente:

$$*\mathbf{u}_1 = \mathbf{K}_0^{-1} \mathbf{q}$$

en los saltos posteriores se obtendrá a partir de:

$$\Delta\lambda_i^0 = \pm \sqrt{\frac{\Delta l^2}{(*\mathbf{u}_i)^T * \mathbf{u}_i + \alpha^2}} \quad (12)$$

tomándose signo positivo según que el determinante de la matriz de rigidez K_i utilizada al principio del salto para obtener u_i ; sea de signo positivo y negativo en caso contrario. De esta forma es posible seguir las ramas de la trayectoria de equilibrio correspondientes a factores de carga decreciente.

Si se utilizan algoritmos de resolución del sistema de ecuaciones basados en la factorización de la matriz de rigidez en producto de una matriz triangular inferior con términos unidad en la diagonal principal por otra triangular superior, el signo del determinante se calcula de forma muy eficiente. Basta para ello calcular el signo del producto de los términos de la diagonal de la matriz triangular superior:

$$K = L U$$

$$\text{signo} (\det K) = \text{signo} (\det L \cdot \det U) = \text{sign} \frac{\prod_{i=1}^n U_{ii}}{\prod_{i=1}^n |U_{ii}|} = \prod_{i=1}^n \text{sign} \frac{U_{ii}}{|U_{ii}|}$$

Ajuste automático de la longitud del salto de carga

En el apartado anterior se suponía que la longitud Δl del vector tangente $(\Delta \lambda_i^0, \Delta u_i^0)$ se mantenía constante en el proceso de carga. Se comprende fácilmente que si esto es así el proceso de corrección requerirá mayor número de iteraciones para conseguir una cierta aproximación en zonas de la trayectoria de equilibrio con gran curvatura que en otras zonas de curvatura menor. Para evitar esto puede variarse Δl con vistas a obtener un número "cuasi-constante" de iteraciones en cada incremento utilizando una expresión propuesta por Ramm¹⁸.

$$\Delta l_i = \Delta l_{i-1} \sqrt{\frac{I_d}{I_{i-1}}} \quad (13)$$

en la que I_{i-1} es el número de iteraciones necesarias para conseguir una cierta aproximación en el paso anterior e I_d el número de iteraciones que se desea mantener constante.

Este sencillo procedimiento produce automáticamente pequeños saltos de carga en zonas de comportamiento altamente no lineal y saltos de carga mayores en zonas de comportamiento casi lineal.

ORGANIZACION DEL PROCESO DE ANALISIS

Las técnicas de solución anteriormente descritas pueden ser fácilmente introducidas en un programa de elementos finitos clásico. En particular el programa FEAP desarrollado por R.L. Taylor¹ resulta especialmente adecuado para ello, dada su estructura modular y su utilización de la idea de macroprograma para configurar algoritmos variables de solución de problemas.

En lo que sigue se supone que el lector está familiarizado con la estructura del programa FEAP.

A continuación se exponen las modificaciones necesarias para hacer posible la utilización las técnicas descritas anteriormente en el programa FEAP. Para ello es

```

SUBROUTINE PMACR (UL,XL,TL,LD,P,S,IE,D,ID,X,IX,F,T,JDIAG,B,DR,
1 CTF,NDF,NDM,NEN1,NST,NEND)
LOGICAL AFR,BFR,CFR,AFL,BFL,CFL,DFL,EFL,FFL,GFL,PCOMP
CHARACTER*4 O,HEAD,PDIS,Z,BC,DI,CD,TE,FD,WD,ENDM,CT,CTL
C
C
REAL*4 LAMDA,LAMDAO,LAMDAN,LAMDAA,DLAMDA
REAL*4 VECLN,UIINORM,UIDOT,ALFA,FNUM,FDEN,CORREC
INTEGER MOLD,MDES
LOGICAL RKFL,DPFL,UIFL,UNFL,UIIFL,UAFL
C
C
COMMON M(1)
.....

FFL = .FALSE.
GFL = .TRUE.
C
RKFL = .FALSE.
DPFL = .TRUE.
UIFL = .TRUE.
UIIFL = .TRUE.
UNFL = .TRUE.
UAFL = .TRUE.
LAMDA = 0.0
ALFA = 1.0
C
NE = NEND
.....

GO TO 330
6 IF (NPLD.GT.0) PROPLD(TIME,0)
C
IF(RKFL) PROP = LAMDA
C
CALL PLOAD(ID,F,DR,NNEQ,PROP)
.....

```

Listado 1. Modificaciones en la subrutina PMACR.

necesario introducir ciertos cambios en las subrutina PMACR y ACTCOL (UACTCL), cambios que en ningún caso afectan a la estructura general del programa.

Las modificaciones necesarias en la subrutina PMACR se reducen al desarrollo de seis nuevas macroinstrucciones, al establecimiento de algunas variables lógicas, y

```

C
      SUBROUTINE ACTCOL (A,B,JDIAG,NEQ,AFAC,BACK,DPFL)
      LOGICAL AFAC,BACK,DPFL
C
      COMMON /ENGYS/ AENGY
      DIMENSION A(1) ,B(1) ,JDIAG(1)
      AENGY = 0.0
C
      DETU=1.0
C
      JR = 0
      .....
      D = A(I)
      A(I) = A(I)/A(ID)
      A(JD) = A(JD) - D*A(I)
400 CONTINUE
C
      DETU=ABS(DETU*A(JD))/(DETU*A(JD))
C
500 IF (BACK) B(J) = DOT (A(JR+1),B(IS - 1),JH-1)
600 JR = JD
C
C
      IF (DETU .LT. 0.0) DPFL = .FALSE.
      WRITE (6,2001) DETU
2001   FORMAT ( DETERMINANTE = ,E15.5)
C
C
      IF (.NOT.BACK) RETURN
      .....

```

Listado 2. Modificaciones en la subrutina ACTCOL.

a la utilización de áreas de memoria para el almacenamiento de dos nuevos vectores de desplazamiento. En la subrutina de solución del sistema de ecuaciones ACTCOL (o UACTCL) se requiere únicamente una pequeña modificación con el fin de conocer el signo del determinante de la matriz de rigidez. Se utilizan dos nuevas subrutinas ADDVEC y STOVEC que facilitan las operaciones entre los distintos vectores de desplazamientos utilizados.

En los listados parciales de las rutinas PMACR y ACTCOL (listados 1 y 2) se recuadran las líneas de programa a introducir. Las modificaciones generales de la rutina PMACR necesarias para la creación de macroinstrucciones está documentada en [1] por lo que no se indican aquí remitiendo al lector a dicha referencia.

```

      SUBROUTINE ADDVEC(A,B,FMULT,N)
C
C suma al vector A el producto del escalar FMULT por el vector B
C
      DIMENSION A(1),B(1)
      DO 10 I= 1,N
10    A(I)=A(I) + FMULT*B(I)
      RETURN
      END

      SUBROUTINE STOVEC(A,B,N)
C almacena el vector B en la dirección de A

      DIMENSION A(1),B(1)
      DO 10 I= 1,N
10    A(I)=B(I)
      RETURN
      END

```

Listado 3. Nuevas subrutinas ADDVEC Y STOVEC.

```

C -----
C      *DTAN* : define la dirección de la tangente
C
C      M(NUI) : puntero al vector de desplazamientos *u1
C      F      : vector de cargas externas
C      DPFL   : indicador lógico del signo del determinante
C              de la matriz de rigidez
C -----
24    IF (UIFL) CALL PSETM(NUI,NE,NEQ*IPR,UIFL)
      PROP=1.0
      CALL PLOAD (ID,F,M,(NUI),NNEQ,PROP)
      BFR= .TRUE.
      IF (CFR) GO TO 241
      CALL ACTCOL (M(NA),M(NUI),JDIAG,NEQ,AFR,BFR,DPFL)
      GO TO 242
241   CALL UACTCL(M(NA),M(NC),M(NUI),JDIAG,NEQ,AFR,BFR,DPFL)
242   AFR =.FALSE.
      IF (.NOT. BFR) GO TO 330
      BFR = .FALSE.
      GO TO 330

```

Listado 4. Macroinstrucción DTAN.

MACROINSTRUCCION DTAN

Esta macroinstrucción define la dirección de la tangente inicial en cada salto de carga. Basta para ello determinar el vector de desplazamiento $*u_i$ (ecuación (8)) correspondiente al vector de cargas externas q y almacenarlo en un nuevo espacio de memoria. Este vector permanecerá inalterado hasta que se vuelva a llamar a DTAN. La subrutina de solución ACTCOL o UACTCL devuelve una nueva variable lógica DPFL de valor .TRUE. si el determinante de la matriz de rigidez utilizada para obtener $*u_i$ es positivo o .FALSE. en caso contrario.

MACROINSTRUCCION LENG (MDES)

Esta macroinstrucción determina la longitud Δl_1 del vector tangente (λ_i^0 , δu_i^0) correspondiente a la primera aproximación dentro de cada salto. Si se trata del primer incremento del proceso de carga se utiliza la ecuación (11) y en el caso general la ecuación (12).

Se puede ajustar el valor de Δl_i con objeto de obtener un número "cuasi-constante" de iteraciones si se introduce como primera variable de la macroinstrucción el número deseado de iteraciones "MDES".

```

C - - - - -
C  *LENG* : determina la longitud del vector tangente
C
C      M(NUA) : puntero al vector de desplazamientos en el
C              inicio del incremento
C      LAMDA  : factor multiplicador de las cargas
C      LAMDAO : incremento de LAMDA en la primera
C              aproximación en cada salto de carga
C      LAMDAA : valor de LAMDA en el inicio del
C              incremento
C      VECLEN : longitud del vector tangente
C      MDES   : número deseado de iteraciones
C      MOLD   : número de iteraciones en el salto anterior
C      ALFA   : coeficiente numérico de homogeneización
C      RKFL   : indicador lógico:
C              .FALSE. = primer salto de carga
C              .TRUE.  = saltos sucesivos
C      UINORM,UIDOT,CORREC: variables auxiliares
C
C - - - - -
25  MDES=CTR(1,L)
    UIDOT=DOT(M(NUI),M(NUI),NEQ)
    UINORM=SQRT(ALFA)*ALFA+UIDOT)
    IF (.NOT. RKFL) GO TO 251

```

```

C
C incremento general de carga
C
      LAMDAO=VECLEN/UITORM
      CORREC =FLOAT(MDES)/FLOAT(MOLD)
      IF (MDES .EQ. 0) CORREC =1.0
      LAMDAO=LAMDAO*SQRT(CORREC)
      GO TO 252

C
C primer incremento de carga
C
251  LAMDAO = DT
      VECLN = LAMDAO * UITORM
      RKLF = .TRUE.

C
252  IF (.NOT. DPFL) LAMDAO= -LAMDAO
      LAMDA=LAMDA
      IF (UAFL) CALL PSETM(NUA,NE,NEQ*IPR,UAFL)
      CALL STOVEC(M(NUA),B,NEQ)
      LAMDA=LAMDA+LAMDA
      CALL ADDVEC(B,M(NUI),LAMDAO,N)
      MOLD=0
      GO TO 330

```

Listado 5. Macroinstrucción LENG.

MACROINSTRUCCION DNOR

Esta instrucción define o actualiza el plano normal según se utilice la condición de "plano normal constante" o "plano normal normal actualizado".

Simplemente calcula:

$$\begin{aligned} u_i^{k-1} - u_{i-1} &= \Delta u_N \\ \lambda_i^{k-1} - \lambda_{i-1} &= \Delta \lambda_N \end{aligned}$$

si esta instrucción se ejecuta una sola vez en cada salto se tiene:

$$\begin{aligned} \Delta u_N &= u_i^0 - u_{i-1} = \Delta u_i^0 \\ \Delta \lambda_N &= \lambda_i^0 - \lambda_{i-1} = \Delta \lambda_i^0 \end{aligned}$$

si dentro del proceso iterativo de corrección se repite la llamada a DNOR entonces:

$$\begin{aligned} \Delta u_N &= u_i^{k-1} - u_{i-1} = \Delta u_i^{k-1} \\ \Delta \lambda_N &= \lambda_i^{k-1} - \lambda_{i-1} = \Delta \lambda_i^{k-1} \end{aligned}$$

```

C -----
C      *DNOR* : define o actualiza el plano normal
C
C      M(NUN) : puntero al vector de desplazamientos
C                incrementales desde el inicio del salto
C      LAMDAN  incremento de LAMDA desde el inicio del salto
C
C -----
26      IF (UNFL) CALL PSETM(NUN,NE,NEQ*IPR,UNFL)
        LAMDAN = LAMDA-LAMDA*ALFA
        CALL STOVEC (M(NUN),B,NEQ)
        CALL ADDVEC (M(NUN),M,(NUA),-1.0,NEQ)
        GO TO 330

```

Listado 6. Macroinstrucción DNOR.

MACROINSTRUCCION RIKS

Esta instrucción calcula los valores actualizados de los desplazamientos totales u_i^k y el factor de carga actualizado λ_i^k mediante:

$$\begin{aligned}
 u_i^k &= u_i^{k-1} + \delta \lambda_i^k * u_i \\
 \lambda_i^k &= \lambda_i^{k-1} + \delta \lambda_i^k
 \end{aligned}$$

```

C -----
C      *RIKS* : calcula el nuevo factor de carga y desplazamientos
C                que cumplen la condición definida por DNOR
C
C      M(NUII) : puntero al vector variación de desplazamientos
C                asociados al residuo anterior ** $\delta u_1$ 
C      DLAMDA  : variación de LAMDA
C      FNUM,FDEN : variables auxiliares
C
C -----
27      IF (UIIFL) CALL PSETM(NUII,NE,NEQ*IPR,UIIFL)
        CALL STOVEC(M(NUII),DR,NEQ)
        FNUM = DOT(M(NUN),M(NUII),NEQ)
        FDEN = DOT(M(NUN),M(NUI),NEQ) + ALFA*ALFA*LAMDAN
        DLAMDA = -FNUM/FDEN
        CALL ADDVEC (B,M(NUI),DLAMDA,NEQ)
        LAMDA = LAMDA + DLAMDA
        MOLD = MOLD + 1
        GO TO 330

```

Listado 7. Macroinstrucción RIKS.

utilizando las ecuaciones (9) y (10) correspondientes a las condiciones de "plano normal constante" y "plano normal actualizado" respectivamente. Nótese que estas dos ecuaciones pueden unificarse en una sola utilizando los valores ΔuN y $\Delta \lambda N$ calculados en la macroinstrucción DNOR haciendo:

$$\delta \lambda_i^k = - \frac{(\Delta uN)^T ** \delta u_i^k}{(\Delta uN)^T * u_i + \alpha^2 \Delta \lambda N}$$

Esta instrucción debe ir antecedida de la instrucción SOLV en la que se actualiza el vector u_i^{k-1} mediante:

$$u_i^{k-1} = u_i^{k-1} + ** \delta u_i^k$$

MACROINSTRUCCION SCAL (α)

Simplemente se define el valor del coeficiente de homogeneización α . Esta instrucción es opcional y si no se incluye, el programa tomará $\alpha = 1$.

```
C -----
C      *SCAL* : define el coeficiente de homogeneización
C
C -----
28      ALFA = CTR(1,L)
        GO TO 330
C -----
```

Listado 8. Macroinstrucción SCAL.

MACROPROGRAMAS DE SOLUCION TIPICOS

A continuación se muestran algunos ejemplos de macroprogramas que a partir de las nuevas instrucciones introducidas permiten implementar los procedimientos de solución descritos en la primera parte del artículo.

El más sencillo corresponde a un algoritmo de incremento de cargas en n saltos con corrección en un plano normal constante e iteraciones tipo Newton-Raphson modificado (Listado 9a). Con objeto de que el lector aprecie con facilidad las diferencias con el proceso usual de control de carga (Listado 9b) se muestra también en columna separada el macroprograma correspondiente a este último.

El método de corrección basado en la actualización del plano normal se implementa de forma sencilla a partir del macroprograma anterior introduciendo la macroinstrucción DNOR en el bucle interior.


```

DT
PROP 1
SCAL  $\alpha$  (opcional)
LOOP N
TANG
DTAN
LENG (MDES)
DNOR
LOOP M
FORM
SOLV
RIKS
NEXT
NEXT
END

```

Listado 9a. Plano Normal Constante.

```

DT
PROP 1

LOOP N
TIME
TANG

LOOP M
FORM
SOLV

NEXT
NEXT
END

```

Listado 9b. Control de Carga.

```

DT
PROP 1
SCAL  $\alpha$ 
LOOP N
TANG
DTAN
LENG (MDES)
LOOP N
DNOR
FORM
SOLV
RIKS
NEXT
NEXT
END

```

Listado 10. Plano Normal Actualizado.

Por último se presenta el macroprograma correspondiente a un proceso de corrección con actualización del plano normal e iteraciones tipo Newton-Raphson (no modificado). (Nótese que en este caso es necesario, además de incluir la instrucción **TANG** en el bucle interior, incluir la instrucción **DTAN** inmediatamente antes de la instrucción **RIKS**, además de la que ya aparece antes de **LENG**).

```

DT
PROP 1
SCAL  $\alpha$ 
LOOP N
TANG
DTAN
LENG (MDES)
LOOP N
DNOR
FORM
TANG
SOLV
DTAN
RIKS
NEXT
NEXT
END

```

Listado 11. Plano normal actualizado e iteraciones $N - R$ no modificado.

EJEMPLOS

Los anteriores algoritmos han sido utilizados para resolver con éxito distintos tipos de problemas estructurales no lineales. A continuación se presentan algunos ejemplos de estructuras de comportamiento altamente no lineal que confirman satisfactoriamente la validez de las implementaciones propuestas. En el Apéndice figuran los listados de datos de cada uno de los ejemplos presentados.

Ejemplo 1: Estructura articulada simple con “durchslag”

En la Figura 4 se muestra un arco triarticulado muy rebajado utilizado frecuentemente en la literatura técnica como ejemplo de validación ya que su formulación analítica¹² es sencilla y permite una comprobación fácil. Se utiliza para su análisis el método de plano normal constante apreciándose incluso con saltos de carga grandes un buen trazado de la curva de equilibrio.

Ejemplo 2: Estructura articulada con “durchslag” y retroceso

La Figura 5 muestra una estructura compuesta a partir de dos estructuras del tipo de la del Ejemplo 1 que permite, mediante una elección apropiada de las constantes de las barras, obtener una respuesta extraordinariamente complicada para una estructura con sólo dos grados de libertad. Un modelo similar aparece descrito por J. Molina y E. Alarcón en la referencia [19] y constituye un excelente ejemplo del fenómeno del retroceso de desplazamiento con carga creciente.

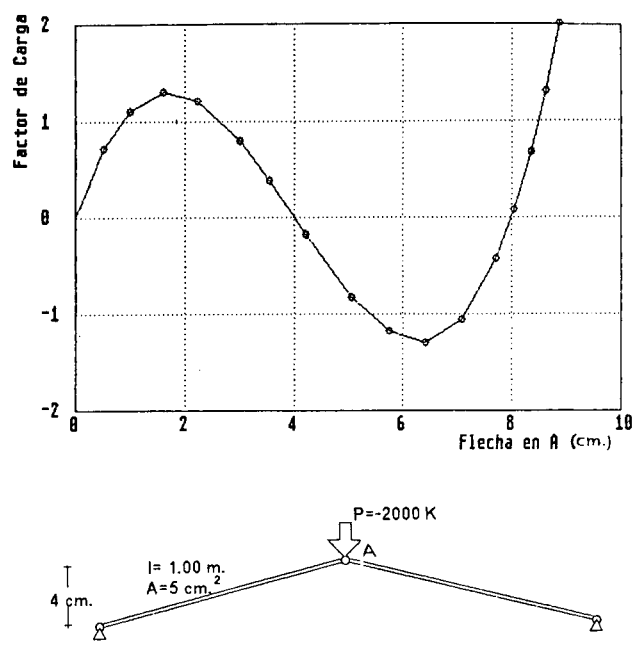


Figura 4.

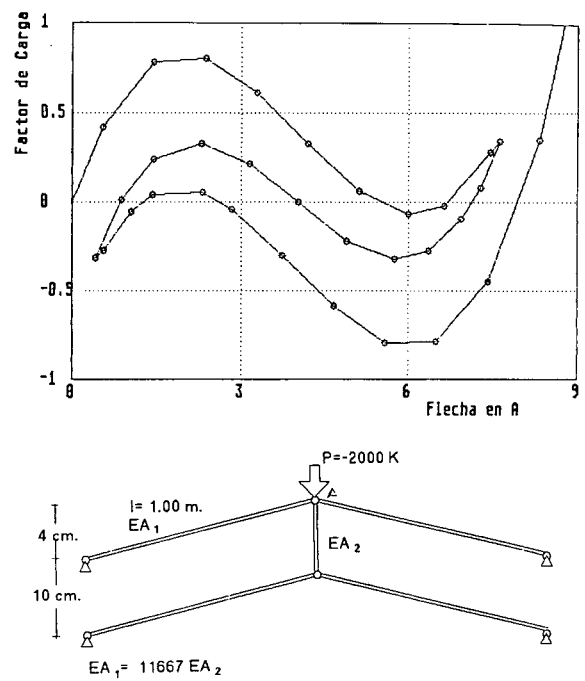


Figura 5.

Ejemplo 3: Celosía de clave rebajada

En la Figura 6 se muestra por último una estructura analizada anteriormente por G. Powell y J. Simmons en la referencia [10]. Esta estructura presenta una pérdida localizada de resistencia con un retroceso súbito en el desplazamiento de algunos grados de libertad. Es interesante hacer notar que en la referencia citada se reconocía la no convergencia de algunas formulaciones allí propuestas, propugnándose como única solución un procedimiento de control de desplazamiento y recomendando la investigación sobre procedimientos de elección automática del grado de libertad a controlar.

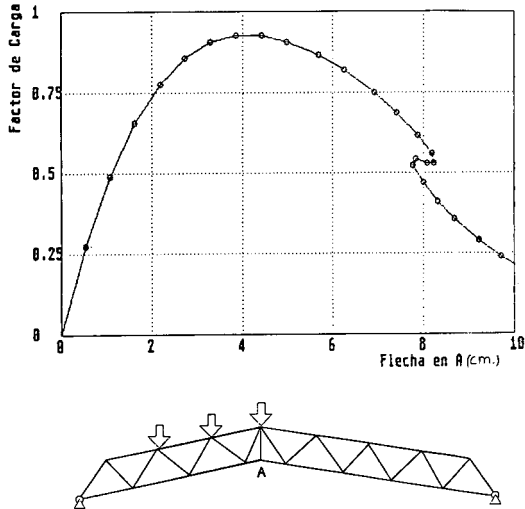


Figura 6.

APENDICE. LISTADOS DE DATOS DE LOS EJEMPLOS

FEAP	** EJEMPLO 1 **				
3	2	1	2	2	2
COORD					
1					
2	99.84		4.		
3	199.68		0.		
ELEM					
1	1	1	2		
5	1	2	3		
MATE					
1	2				
2100000.	25.0				
BOUND					
1		1	1		
3		1	1		

```

FORC
  2          -1000.
END
MACR
DT      0.75
TOL     .0001
SCAL    1.
PROP
LOOP    20
TANG
DTAN
LENG    3
DNOR
LOOP    6
FORM
SOLV
RIKS
NEXT
NEXT
END
  0.0      100.          1.
STOP

```

```

FEAP      ** EJEMPLO 2 **
  6      5      2      2      2      2
COOR
  1          10
  2      99.84      14.
  3      199.68      10.
  4          -10.
  5      99.84      -6.
  6      199.68      -10.
ELEM
  1      1      1      2
  2      1      2      3
  3      1      4      5
  4      1      5      6
  5      2      2      5
MATE
  1      2
2100000.  25.0
  2      2
180.      25.
BOUN
  1          1      1
  3          1      1
  4          1      1
  6          1      1
FORC
  2          -2000.
END
MACR

```

```

DT          .5
TOL         .01
SCAL
PROP
LOOP        35
TANG
DTAN
LENG        3
LOOP        6
DNOR
FORM
SOLV
RIKS
NEXT
NEXT
END
0.0      100.      1.
STOP

```

```

FEAP      ** EJEMPLO 3 **
18      33      1      2      2      2
COOR
1      2
7      30.      6.
2      2      5.0      5.
8      35.      11.
9      35.      7.
10     2      40.      6.222
18     80.      0.0
11     2      45.      9.444
17     75.      4.778
ELEM
1      1      1      3      2
5      1      9      10
6      1      10     12     2
10     1      2      4      2
13     1      8      11
14     1      11     13     2
17     1      1      2      1
25     1      8      10
26     1      10     11     1
MATE
1      2
2100000. 4.286
BOUN
1      1      1
18     1      1
FORC
4      -40000.
6      -40000.
8      -40000.
END

```

```

MACR
DT      .30
TOL     .01
PROP
LOOP    30
TANG
DTAN
LENG     3
DNOR
LOOP     5
FORM
SOLV
RIKS
NEXT
NEXT
END
      0.0    100.          1.
STOP

```

REFERENCIAS

1. O.C. Zienkiewicz, "El método de los elementos finitos", Reverté, Barcelona, (1980).
2. J.T. Oden, "Finite Elements of Nonlinear Continua", McGraw-Hill, N.Y., (1972).
3. E. Riks, "The Application of Newton's Method to the Problem of Elastic Stability", *J. Appl. Mech.*, Vol. 39, pp. 1060-1066, (1972).
4. E. Riks, "An incremental approach to the solution of snapping and buckling problems", *Int. J. Solids Structures*, Vol. 15, pp. 524-551, (1979).
5. G.A. Wempner, "Discrete Aproximations related to Non-Linear Theories of solids", *Int. J. Solids Struct.*, Vol. 7, pp. 1581-1599, (1971).
6. P.G. Bergan, "Solution Algorithmes for Nonlinear structural Problems", *Int. Conf. on "Eng. Appl. of the F.E. Method"*, Hovik, Norway, published by A.S. Computas, (1979).
7. E.W. Wright y E.H. Gaylord, "Analaysis of Umbraced Multy-Story Steel Rigid Frames", *J. Structural Division, ASCE*, 94, pp. 1143-1163, (1968).
8. J.H. Argyris, "Continua and Discontinua", *Proc. 1st. Conf. "Matrix Meth. Struct. Mech."*, Wright-Patterson A.F.B., pp. 11-189, Ohio, (1965).
9. J.L. Batoz y G. Dhatt, "Incremental Displacement Algorithms for Nonlinear Problems", *Int. J. Num. Meth. Engng.*, Vol. 14, pp. 1262-1267.
10. G. Powell, J. Simons, "Improved Iteration Strategy for Nonlinear Structures", *Int. J. for Numerical Methods en Engineering*, Vol. 17, pp. 1455-1467, (1981).
11. G. Dahlquist, A. Björck, "Numerical Methods", Prentice-Hall, N.Y., (1974).
12. R. Irlés, "Un modelo numérico para el análisis del colapso en entramados metálicos", *Tesis Doctoral, ETS de ICCP*, Valencia, (1985).
13. H. Rothert, T. Dickel y D. Renner, "Snap-Trough Buckling of Reticulated Space Trusses", *J. of the Structural Division, ASCE*, (1987).
14. J.M. Sancho y J. Ortiz, "Vertical Shear Strength of Encased Composite Beams", Comunicación enviada a la "Engineering Foundation Conference on Composite Construction de A.S.C.E.", (1987).
16. M.A. Crisfield, "Solution Procedures for Non-Linear Structural Problems", *Recent Advances In Non-Linear Computational Mechanics*, Pinerridge Press, (1982).

17. M.A. Crisfield, "A Fast Incremental-Iterative Solution Procedure that Handles Snap-Trough", *Computers and Structures*, Vol. 13, pp. 55-62, Pergamon Press, G.B., (1981).
18. E. Ramm, "Strategies for Tracing the Nonlinear Response Near Limit Points", *Nonlinear Finite Element Analysis in Structural Mechanics*, Springer-Verlag, Berlín, (1981).
19. J. Molinas y E. Alarcón, "Un nuevo método de longitud de arco para estructuras", *Revista Internacional de Métodos Numéricos para Cálculo y Diseño en Ingeniería*, Vol. 4, no. 3, pp. 369-384, (1988).