

## FLUX2D y FLUX3D SOFTWARES DE CAD PARA LA MODELIZACION DE CAMPOS ELECTROMAGNETICOS

Orietta SANTANA, Gérard MEUNIER,  
Jean Louis COULOMB, Jean Claude SABONNADIÈRE

Laboratoire d'Electrotechnique de Grenoble (UA CNRS 355)  
ENSIEG, BP.45  
38402 St. Martin d'Hères  
FRANCIA

### RESUMEN

*La utilización de métodos numéricos para modelar (modelling) fenómenos electromagnéticos, nos ha permitido la realización de los softwares FLUX2D y FLUX3D en el Laboratoire d'Electrotechnique de Grenoble.*

*FLUX2D (2 dimensiones y simetría de revolución) y FLUX3D (3 dimensiones) son softwares para el cálculo de campos electromagnéticos que utilizan el método de los Elementos Finitos para resolver ecuaciones de derivadas parciales. Estos softwares comprenden 3 partes :*

*El Pre-procesador que recibe las informaciones que describen la geometría y las propiedades físicas del modelo en forma interactiva. Los elementos de la malla son generadas automáticamente (elementos triangulares en 2D o tetraédricos en 3D) a partir de los nodos que se apoyan en la geometría (fronteras del objeto).*

*El Procesador de resolución, que calcula los valores de las incógnitas en cada nodo de la malla o mallaje (mesh). Los elementos finitos son rectilíneos (primer orden) o curvilíneos (segundo orden). Las no-linealidades a causa de materiales no-saturables son tratadas con el método de Newton-Raphson. El sistema de ecuaciones lineales es resuelto con el método ICCG (Incompleto Conjugate Cholesky Gradient).*

*El Post-procesador que es concebido para analizar, evaluar y presentar gráficamente los resultados del procesador de cálculo. Así, podemos visualizar líneas isovalores (2D) o superficies isovalores (3D) para magnitudes escalares; una red de flechas para magnitudes vectoriales y todo sobre planos de corte (cross section) o sobre las fronteras del objeto. También es posible de evaluar magnitudes globales como fuerzas y momentos que se ejercen en una región o en todo el objeto.*

Las primeras aplicaciones de CAD (Diseño Asistido por Computador) vieron la luz en el campo de la electrónica donde la necesidad de concebir circuitos impresos y luego integrados a muy grande escala ha hecho indispensable el uso de la informática. Su empleo ha sido rápidamente repandido en la aviónica (CAD de la Industria de la Aviación) donde la complejidad de la concepción de aviones o helicópteros necesita desde el punto de vista mecánico y de estructuras el empleo de softwares bastante elaborados. Poco a poco las técnicas de CAD han ganado campos diversos como la industria del automóvil, la arquitectura, la concepción de cableajes electrónicos. Hoy en día, nuevas aplicaciones se desarrollan en todos los campos de la industria.

La industria eléctrica no se ha escapado a este fenómeno. El decenio próximo verá sin lugar a dudas la generalización de tales sistemas de los cuales las primeras implantaciones han demostrado por parte de los usuarios un gran interés.

En la concepción de estructuras electromagnéticas, el conocimiento del campo magnético es necesario para predecir:

- los fenómenos locales (densidad de flujo magnético, densidad de corriente, densidad de fuerza, ...) que caracterizan el funcionamiento interno de las máquinas;
- los parámetros globales (reluctancias, momentos, ...) que conciernen el comportamiento de las máquinas en una cadena de conversión electromecánica.

Antiguamente, la distribución del campo en las máquinas eléctricas era establecida con métodos analíticos lineales, transformaciones conformes y métodos de circuitos magnéticos lineales. Los resultados obtenidos con estos métodos eran mejorados utilizando factores correctores empíricos que tomaban en cuenta las no-linealidades. Sin embargo, la complejidad geométrica creciente así como la carrera hacia la optimización, nos probaron la necesidad de métodos de análisis más rigurosos.

Los computadores han favorecido el desarrollo de métodos numéricos. El método de las diferencias finitas fué el primero en ser empleado. Bajo este método, las ecuaciones de derivadas parciales representando el problema del campo son remplazadas por un conjunto de ecuaciones en diferencias finitas expresadas en los nodos de la red de puntos repartidos en el objeto modelado. Este procedimiento genera un conjunto de ecuaciones lineales, que una vez resueltas, dan una distribución aproximada del campo. Este método ha sido bastante utilizado a pesar de la dificultad de implantación en un software CAD a causa de los problemas de discretización inherentes al método.

El método de los elementos finitos se basa en una formulación variacional de las ecuaciones de derivadas parciales representando el problema del campo. Ini-

cialmente utilizado en análisis de estructuras mecánicas y en transferencia térmica, permite la resolución de problemas bidimensionales de electrostática, de magnetostática y de magnetodinámica, lineales o no, en sistemas cartesianos o de simetría de revolución. Este método ha sido implantado con éxito en los softwares de CAD, donde FLUX2D desarrollado en el *Laboratoire d'Electrotechnique de Grenoble*, es una ilustración, gracias a la relativa facilidad de discretización de todo el dominio bidimensional en elementos finitos y gracias a la capacidad de resolver los problemas no-lineales.

La extrapolación del método de los elementos finitos al caso tridimensional, encierra ciertos problemas como:

- la multiplicidad de formulaciones posibles
- la dificultad para discretizar objetos tridimensionales
- la dificultad para visualizar los resultados obtenidos.

Sin embargo, el método de los elementos finitos es el único método que actualmente permite resolver problemas tridimensionales no lineales en una óptica general de CAD. FLUX3D es una prueba de esta realización.

## II. PRE-PROCESADOR

El papel que desempeña este módulo es la adquisición de datos y la preparación de las informaciones para el tratamiento del problema con el método de los elementos finitos. El pre-procesador de los softwares FLUX2D y FLUX3D realiza las siguientes funciones:

- parametrización
- descripción geométrica del objeto
- mallaje
- preparación del problema.

A continuación presentaremos las funciones del pre-procesador de FLUX3D ya que el de FLUX2D es más sencillo y está comprendido en el primero.

### 1. Parametrización

La experiencia adquirida en la utilización de softwares elementos finitos nos ha mostrado la importancia de la descripción parametrizada de la geometría. Generalmente un usuario de un programa como FLUX3D no estará satisfecho de su primer cálculo y deseará modificar una parte de su modelo geométrico (el espesor del entrehierro de un contactor, por ejemplo) para luego poder calcular un nuevo campo magnético y analizar los nuevos resultados físicos.

La parametrización debe permitir al usuario poder visualizar en forma sencilla con respecto a una característica importante de las especificaciones, el efecto de la variación de uno o varios parámetros constructivos del objeto. Por ejemplo, dada una pieza eléctrica y una magnitud fija que se desea obtener, se debe poder jugar

con las cantidades de material (fierro, cobre, ...) o las medidas de la pieza. En general se trata de optimizar el volumen y la cantidad de material.

## 2. Descripción geométrica del objeto

Un programa interactivo nos permite crear objetos a partir de la característica más elemental: las coordenadas de los puntos que constituyen las fronteras del objeto (figura 1). A esta fase la llamamos *geometría*. Enseguida viene la fase de *topología* que consiste en crear también de forma interactiva las líneas que pueden ser :

- segmentos de recta (entre 2 puntos)
- arcos de círculo (entre 3 puntos).

Luego son creadas las **facetas** del objeto a partir de una lista de líneas consecutivas, estas facetas pueden ser:

- planas, poligonales (a partir de segmentos, mínimo 3)
- planas, no poligonales (a partir de segmentos y/o arcos)
- cilíndricas (mínimo 2 segmentos y 2 arcos).

Los **volúmenes** son finalmente creados con la unión de varias facetas.

Las estructuras de datos empleadas en el caso de FLUX3D para esta etapa son las siguientes :

*tipo\_entidad* PARAMETRO :

NUMERO            entero [1] ;  
NOMBRE            caracteres [16] ;  
VALOR             real [1] ;

*fin\_entidad* ;

*tipo\_entidad* PUNTO :

NUMERO            entero [1] ;  
COORDENADAS    real [3] ;  
FORMULA           caracteres [80] ;

*fin\_entidad* ;

*tipo\_entidad* LINEA :

NUMERO            entero[1] ;  
selección entre:  
SEGMENTO        PUNTO [2] ;  
ARCO              PUNTO [3] ;

*fin\_elección* ;

*fin\_entidad* ;

*tipo\_entidad* FACETA:

NUMERO            entero [1] ;  
LINEAS            LINEA [2:99] ;  
ORIENTACION    entero [2:99] ;  
COLOR             caracteres [16] ;

*fin\_entidad* ;

*tipo\_entidad* VOLUMEN:

    NUMERO                    entero [1] ;

    FACETTAS                FACETTA [4:99] ;

*fin\_entidad* ;

donde *tipo\_entidad* corresponde a la estructura **record** del Pascal. Cada *tipo\_entidad* se identifica con un nombre y una serie de atributos que pueden ser enteros, reales y/o asociaciones hacia otras entidades (**link**). Por ejemplo, LINEA.SEGMENTO es una asociación hacia 2 entidades PUNTO. Más detalles en cuanto a la estructuración de datos del pre-procesador se encuentran en [2].

### 3. Mallaje

La generación de la malla consiste en determinar una cantidad de nodos y de elementos finitos que formen una discretización aceptable del objeto. Esta discretización se hace generalmente en 5 etapas:

- 1) Discretización automática de las líneas.
- 2) Triangulación de cada faceta del objeto (algoritmo de Delaunay 2D).
- 3) Discretización de las facetas del objeto (algoritmo de Delaunay).
- 4) Tetraedrisación (algoritmo de Delaunay 3D).
- 5) Creación de nodos internos al volumen (algoritmo de Delaunay).
- 6) Regularización automática de nodos (algoritmo de Laplace).

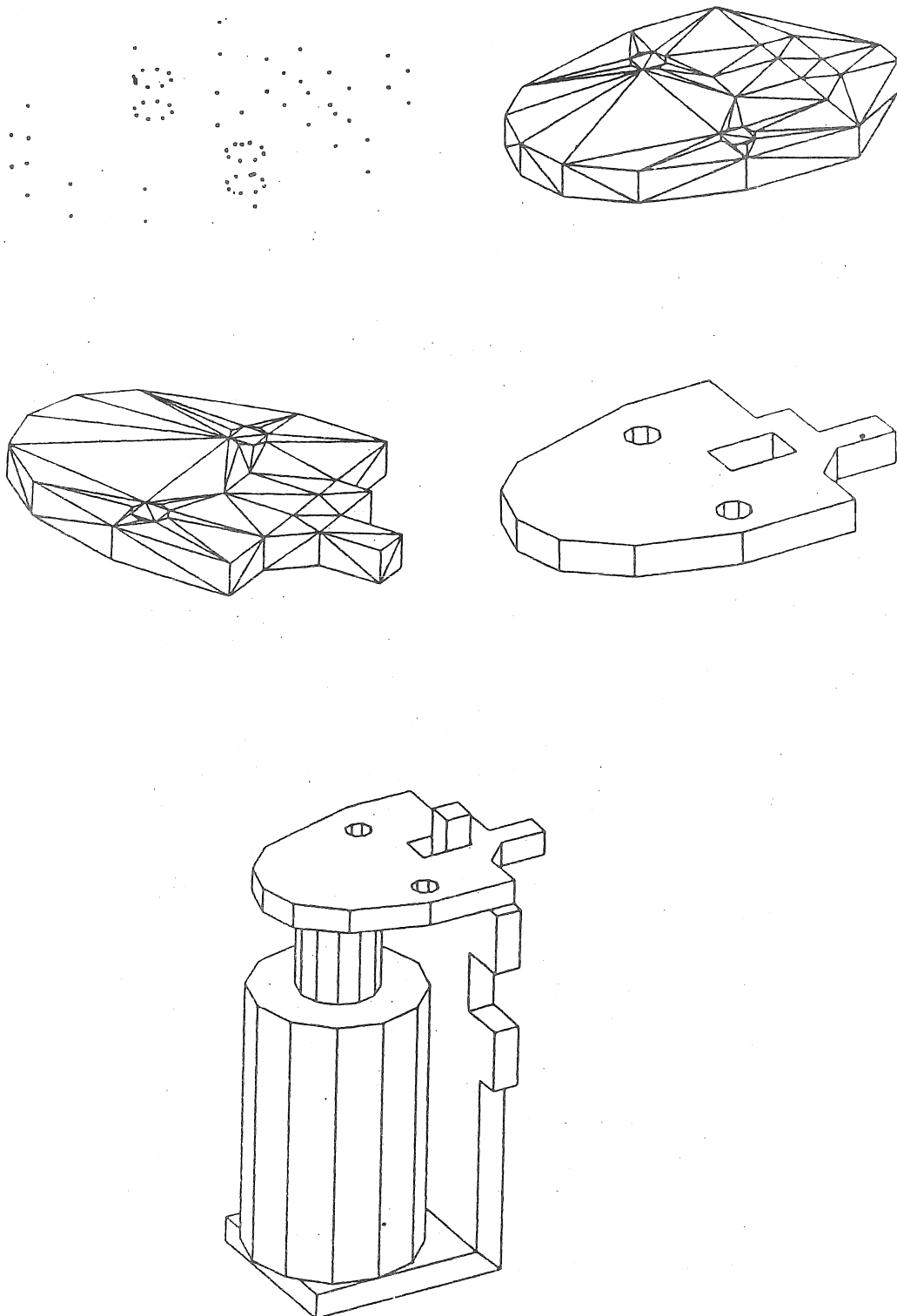
La discretización de líneas se realiza de forma automática. Los nodos son repartidos de forma uniforme (la distancia entre nodos es la misma) o concentrada a una extremidad (exponencial).

El algoritmo de Delaunay genera a partir de un grupo de nodos iniciales repartidos de forma cualquiera en el espacio, un conjunto de elementos triangulares (2D) o tetraédricos (3D). Estos elementos tienen la característica de ser los *más equilaterales posibles* y además la esfera circunscrita a un elemento dado (a los 4 nodos de un tetraedro por ejemplo) no contiene ningún otro nodo. Más detalles sobre este algoritmo en [1].

Así en la etapa 2 se aplica el algoritmo de Delaunay a cada faceta del objeto a partir de los nodos creados en las líneas. En la etapa 4 los nodos iniciales son los nodos de las facetas del objeto.

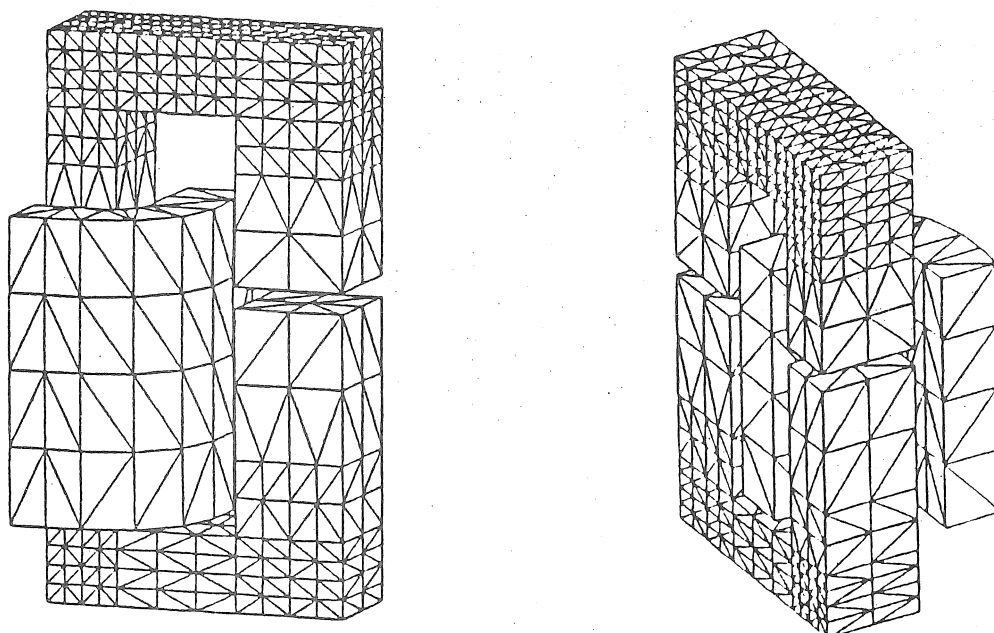
Las etapas 3 y 5 consisten en insertar nuevos nodos correctamente posicionados en los elementos *muy grandes*, es decir en los que el volumen excede a un cierto volumen deseado, o en los elementos que no respetan las especificaciones de calidad (ángulos muy agudos, tetraedros planos). Cada nuevo nodo se inserta en el centro de gravedad del elemento, un mallaje local es entonces construido alrededor del nodo.

Para la última etapa se aplica el algoritmo de Laplace [5]. Cada nodo que no forma parte de una frontera es desplazado hacia el centro de gravedad de los nodos que le son conectados. Este procedimiento es simple y converge rápidamente (2 o 3 iteraciones).

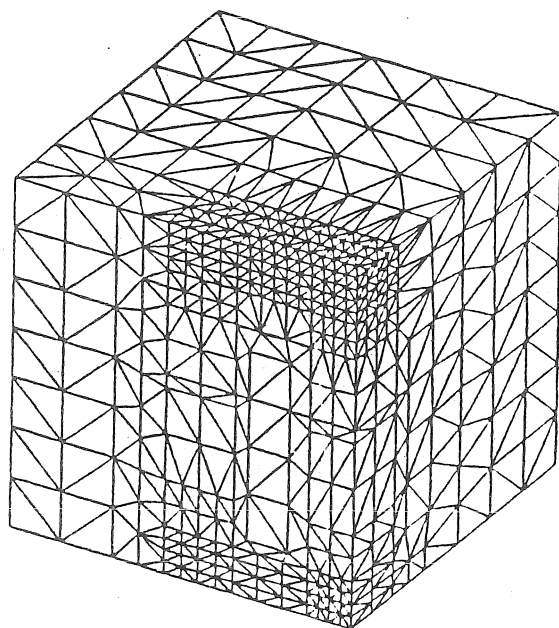


*Figura 1.*

Proceso de construcción de un objeto.



**MAILLAJE EN ELEMENTOS FINITOS  
DE UN CUARTO DE CONTACTOR ELECTROMAGNETICO**



**MAILLAJE DEL CONTACTOR Y DEL AIRE QUE LO ENGLOBA**

*Figura 2.*

#### **4. Preparación del problema**

La preparación del problema consiste en la operación de definición de regiones y fronteras:

- descripción de las propiedades del material (por ej. conductividad térmica, permitividad, ..)
- descripción de las fuentes (por ej. fuente de corriente)
- descripción de las condiciones en los límites
- descripción de las condiciones iniciales para los problemas dependientes del tiempo.

### **III. PROCESADOR**

El procesador de cálculo permite resolver problemas de:

- electrostática
- magnetostática escalar
- magnetostática vectorial
- magnetostática mixta
- conductor
- electrocinética

La estructura del procesador es tal que una nueva formulación puede ser fácilmente implantada. Acepta los elementos finitos de primer o segundo orden, rectilíneos o curvilíneos, tetraédricos, prismáticos o hexaédricos. Las no-linealidades dadas por la presencia de materiales saturables son resueltas con el método de Newton-Raphson.

Los sistemas de ecuaciones algebraicas de matrices dispersas simétricas definidas positivas son resueltos con el método de los gradientes conjugados con pre-condicionamiento por descomposición de Cholesky incompleto (ICCG). Los otros sistemas son resueltos con el método de los bigradientes conjugados con pre-condicionamiento.

### **IV. POST-PROCESADOR**

#### **1. Evaluación de una cantidad local utilizando formulas**

Un post-procesador que permite la evaluación de cantidades locales a partir de resultados brutos del cálculo con elementos finitos puede ser construido siguiendo:

- una técnica por procedimiento: el programa permite el cálculo de un conjunto de cantidades predefinidas y nada más que ellas. El principal inconveniente de esta técnica es la necesidad de aumentar nuevas instrucciones al interior del programa para implantar la evaluación de una cantidad no prevista inicialmente.



- una técnica por fórmula: el usuario expresa las cantidades deseadas bajo la forma de una expresión aritmética (por ej.  $F = J \times B$ ) en función de cantidades de base tales como las coordenadas, las variables de estado y las características físicas. Este método permite efectuar todos los cálculos deseados sin modificar el programa que puede ser idéntico sea cual fuera la formulación y la dimension. Este método es el que ha sido implantado.

### Administración de parámetros

El post-procesador permite el manejo interactivo de parámetros (estos parámetros son diferentes a los que fueron descritos en el pre-procesador). Un parámetro se caracteriza por:

- el número de componentes (1 para un escalar, 3 para un vector)
- su tipo (real o complejo)
- su origen que puede ser:
  - variables de estado ( $A, \Phi, \dots$ )
  - características físicas ( $J, V, \dots$ )
  - derivada de variables de estado (gradiente, rot, div)
  - coordenadas de espacio ( $X, Y, Z$ )
  - fórmula (combinación algebraica de parámetros existentes)
  - cantidad resultante de una integración global.

### Biblioteca de sub-programas para la evaluación de fórmulas

Esta biblioteca es una extensión en el caso vectorial o complejo de una biblioteca existente utilizada para la manipulación de fórmulas escalares por los pre-procesadores. Ella permite la evaluación de fórmulas construidas a partir de las funciones de base (seno, coseno, raíz cuadrada, ...) y los operadores usuales (+, -, \*, /, \*\*). Cada expresión contiene uno o varios parámetros (por ej.  $\text{SQRT}((NU*B-HC)**2)$  o  $\text{MOD}(NU*B-HC)$  ).

### Utilización de fórmulas en el post-procesador

La base de datos de los parámetros ha sido inicializada dinámicamente y automáticamente con las variables de estado y las características físicas correspondiente a la formulación en curso. Por ejemplo, en el caso de un problema tridimensional de magnetostática vectorial, tenemos los parámetros iniciales siguientes:

- A : potencial vector, variable de estado, 3 componentes reales
- J : densidad de corriente, característica física, 3 componentes reales
- NU : reluctividad magnética, característica física, 1 componenete real (isotrópico)
- HC : campo coercitivo, característica física, 3 componentes reales
- X,Y,Z : coordenadas de espacio

El usuario puede construir cualquier fórmula utilizando los parámetros precedentes. Esta fórmula podría convertirse en un nuevo parámetro para ser utilizado en nuevas fórmulas. Por ejemplo:

- creación de un nuevo parámetro  $B = \text{rot}(A)$
- creación de un nuevo parámetro  $H = NU * B - HC$
- evaluación de  $\text{MOD}(H)$

En un punto dado, la evaluación de una fórmula necesita las siguientes etapas:

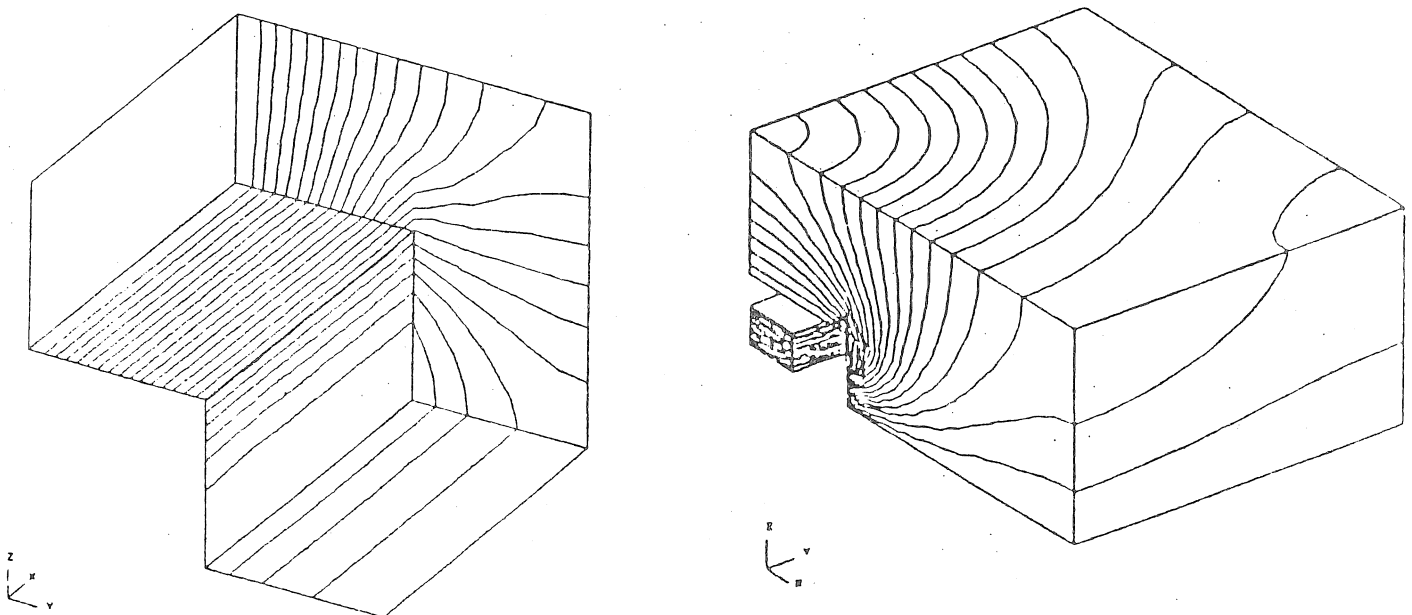
- entrada interactiva y análisis sintáxico y semántico
- determinación de la lista jerárquica de parámetros utilizados (un parámetro puede depender de otro parámetro)
- cálculo de parámetros utilizados en función de su origen
- evaluación gracias a los útiles de la biblioteca.

## 2. Visualización de variables de espacio utilizando fórmulas

Las variables de espacio (representadas con fórmulas) pueden ser visualizadas en dos tipos diferentes de soporte (definido de forma interactiva): *secciones planas o fronteras de un objeto*. En los 2 casos, la superficie se representa de manera interna por un conjunto de triángulos (intersección del plano con los elementos finitos, o descomposición de fronteras) que nos permiten eliminar las líneas escondidas (*hidden lines*). El usuario puede pedir:

- una representación de curvas de nivel de una magnitud escalar (figura 3).
- una representación con flechas de una magnitud vectorial (figura 4).

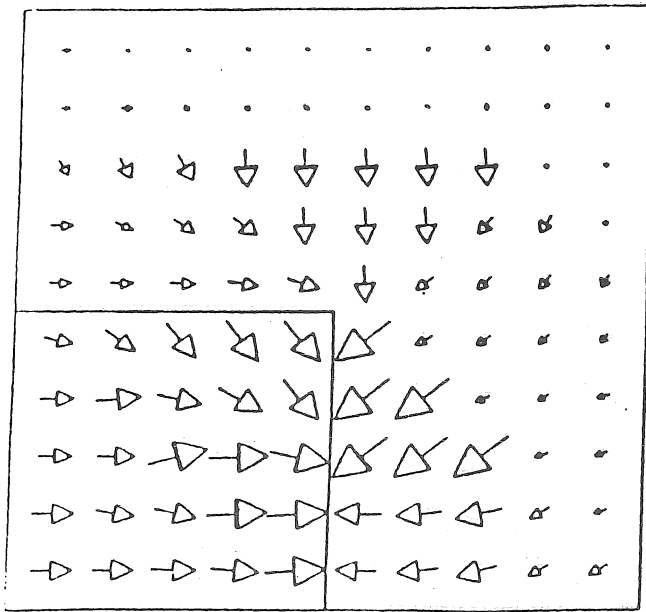
Además el usuario puede pedir la representación bajo la forma de grafo (X,Y) una magnitud a lo largo de una línea arbitraria (figura 5).



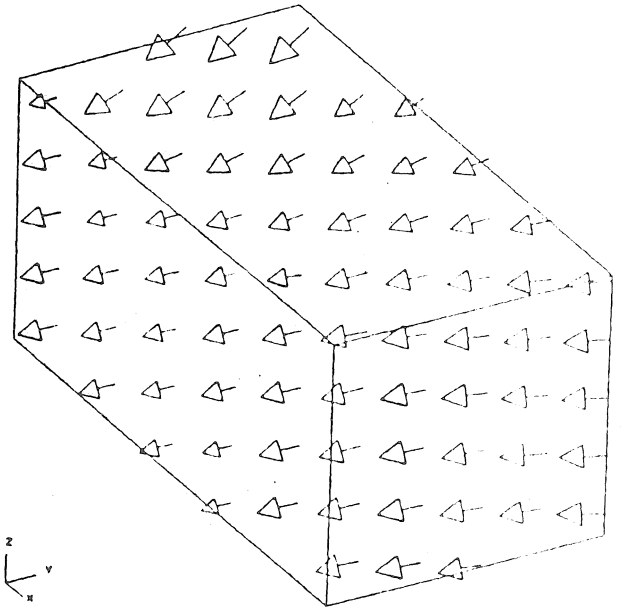
EQUIPOTENCIALES

*Figura 3.*

FLUX2D

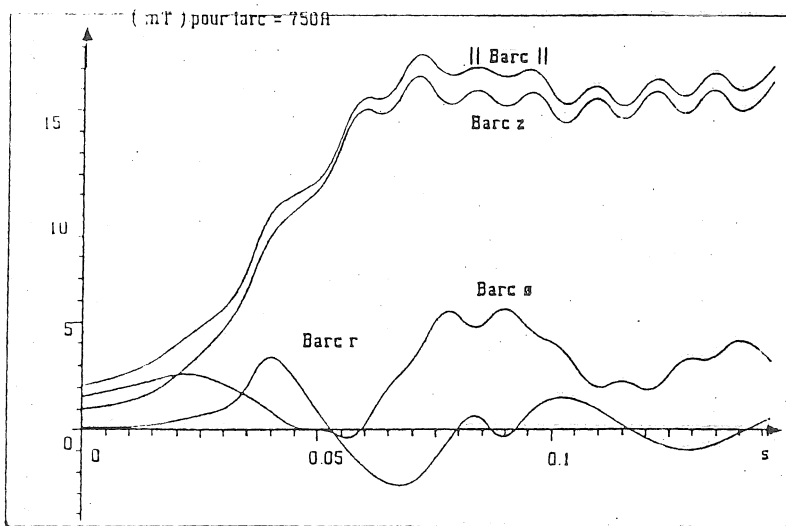


FLUX3D - 511 -

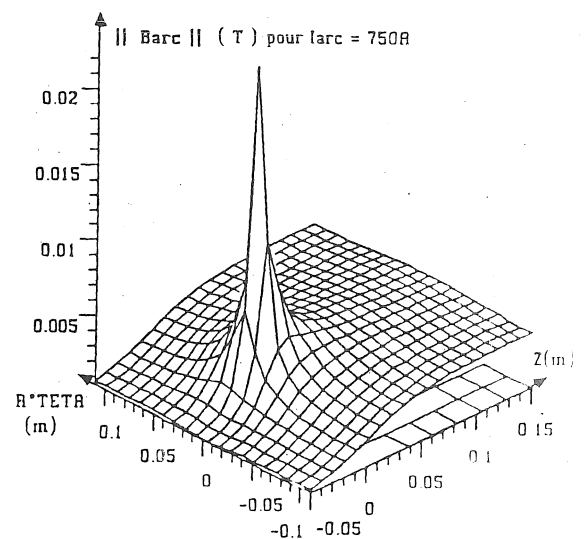


Red de flechas.

Figura 4.

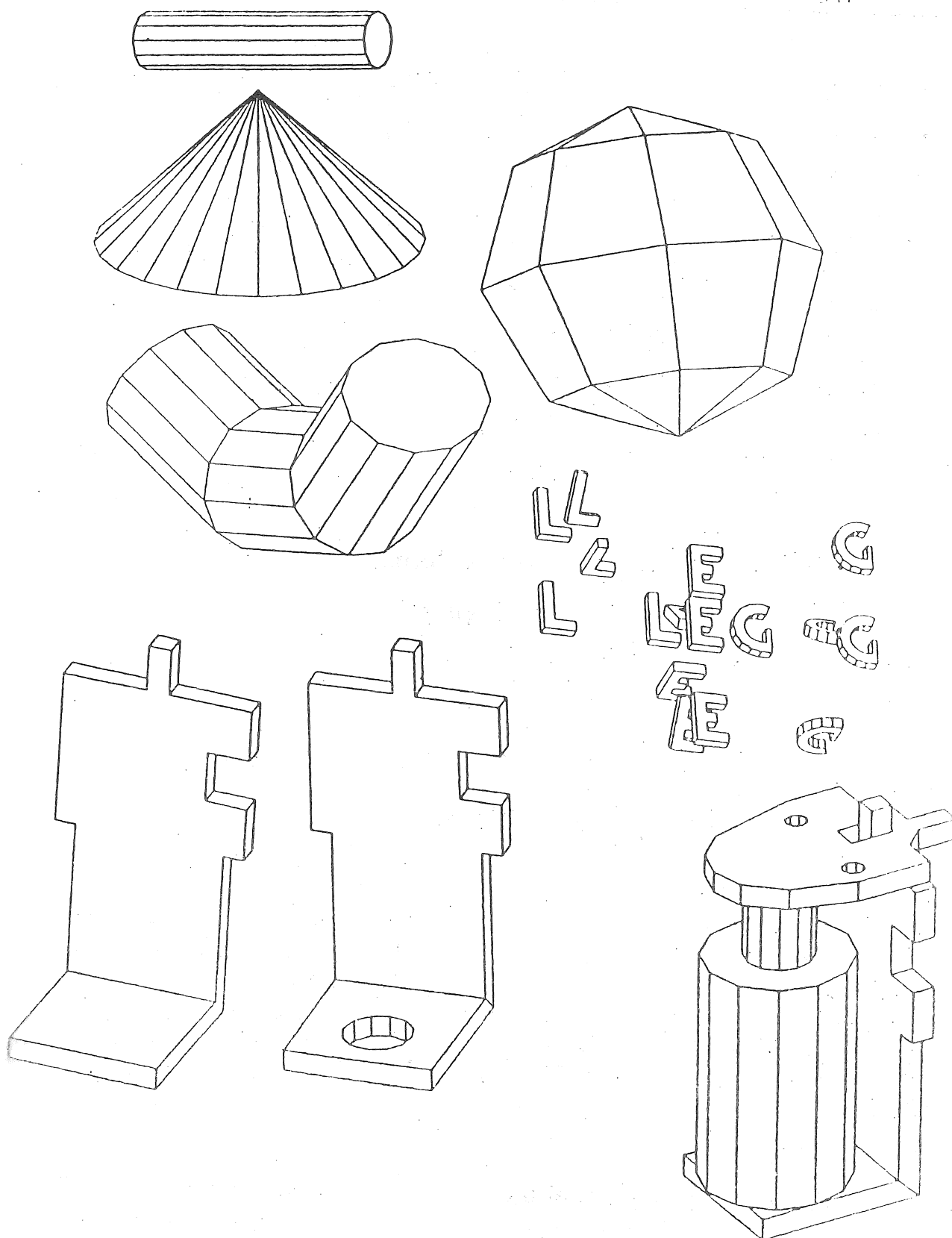


Variación de una magnitud  
a lo largo de un segmento.



Variación de una magnitud  
a lo largo de un rectángulo.

Figura 5.



Visualización de objetos eliminando líneas escondidas  
(hidden lines)

*Figura 6.*

### 3. Cálculo de cantidades globales utilizando las fórmulas

El post-procesador brinda la posibilidad de calcular interactivamente las integrales de una fórmula. La cuadratura concierne el objeto entero o una región en particular. Sobre cada elemento finito se utiliza una integración numérica con el método de Gauss. Por ejemplo:

- la integral de 1 da el volumen;
- la integral de  $H \times B/2$  da la energía.

El resultado de una integración puede ser almacenado opcionalmente como parámetro para una utilización futura.

### BIBLIOGRAFIA

- [1] Y. Du Terrail, O. Santana, G. Meunier, J.L. Coulomb, GET3D: un mailleur en elements finis tetraedriques pour la modelisation des champs electro-magnetiques. Artículo presentado a STRUCAD'86, Paris, Octubre 1986.
- [2] O. Santana, JPh. Iafrate, J.L. Coulomb, A data structure for a finite element package. Artículo presentado a COMPUMAG'87, Graz-Austria, Agosto 1987.
- [3] J.C. Sabonnadière, J.L. Coulomb, Eléments finis et CAO. Paris, Ediciones Hermes, 1986.
- [4] G. Meunier, J.L. Coulomb, D. Savalle, Use of formulæ in finite elements post-processing. Artículo presentado a COMPUMAG'87, Graz-Austria, Agosto 1987.
- [5] JPh. Iafrate, O. Santana, J.L. Coulomb, DB-LIB: a tool box of data description and management. Artículo presentado en ICIAM'87, París, Junio 1987.
- [6] Jean-Louis Coulomb, Analyse tridimensionnelle des champs électriques et magnétiques par la méthode des éléments finis. Tesis de Docteur es-ciencias physiques, INP de Grenoble, Junio 1981.
- [7] Yves Du Terrail, Modélisation géométrique et topologique en 3 dimensions pour l'application de la méthode des éléments finis en électromagnétisme. Tesis de Doctor del INP de Grenoble, Octubre 1985.
- [8] Didier Savalle, Contribution à la conception et à la réalisation d'un processeur d'exploitation et de visualisation des résultats d'un calcul de champs électromagnétiques tridimensionnels. Tesis de Doctor del INP de Grenoble, Mayo 1987.