



**ESTUDIO DEL MÉTODO DE NUBE DE PUNTOS 2-D APLICADO A LA ECUACIÓN DE
LAPLACE (FLUJO POTENCIAL) SOBRE EL PERFIL NACA 0012**

JAVIER MAGGIOLI - SERGIO IDELSOHN

Grupo de tecnología Mecánica - INTEC - CONICET. U. N. del Litoral. Güemes 3450.
e-mail: jmaggiol@venus.unl.edu.ar - Santa Fe - ARGENTINA

RESUMEN:

En este trabajo presentamos un método de los denominados *libres de malla*, para resolver numéricamente un sistema discretizado de ecuaciones diferenciales. Esto significa que la discretización de la función desconocida y sus derivadas son obtenidas solo a partir de las coordenadas de los puntos dentro del dominio en estudio. La discretización del dominio en este método requerirá de una grilla de puntos, pero este requerimiento es sin duda mucho más sencillo y rápido de satisfacer, sobre todo en 3D, con respecto a una topología de elementos geométricos requerida en los métodos clásicos FEM y FVM. La aplicación del método se hizo a un perfil NACA 0012 dentro de un flujo potencial (ecuación de Laplace). Los resultados presentados muestran la buena aproximación del método, aún cuando tenemos pocos puntos sobre el perfil.

ABSTRACT:

This research deals with one of the 'free mesh' methods used to resolve numerically a discretized differential equations system. This means that the unknown function and its derivatives discretization are obtained just starting from the studied domain points coordinates. The domain discretization will need an array of points. Certainly, this is easier and faster as compared to the geometrical elements topology required by the classic FEM and FVM methods, specially in 3D applications. The method was applied to a NACA 0012 profile in a potential flow (Laplace equation). The results show the good approximation reached even when the profile has few points.

1. INTRODUCCIÓN:

El método de elementos finitos y el método de volúmenes finitos son técnicas muy usadas y eficientes en la resolución numérica de distintos problemas en ingeniería. Una de las principales ventajas de estos métodos es la obtención de buenas aproximaciones locales sobre dominios complicados.

Los dos métodos anteriores dividen al dominio total en un número finito de subdominios, necesarios para la interpolación y posterior integración numérica.

Es por esto que los subdominios son limitados a ciertas formas geométricas simples. Estos métodos no presentan mayores dificultades en problemas 2-D, pero traen algunos inconvenientes en el tratamiento de problemas 3-D sobre todo en lo concerniente a la generación de la malla de elementos (subdominios 3-D). La generación de mallas 3-D se plantea en la actualidad como uno de los grandes desafíos del FEM y del FVM y a pesar de permanentes avances en este campo sigue siendo el cuello de botella en los cálculos a escala industrial para fluidos, absorbiendo en muchos casos mayor costo y tiempo que la solución numérica del problema.

En las últimas décadas se han buscado soluciones alternativas, intentando el desarrollo de métodos que no necesiten de una malla de elementos, estos métodos se han denominado *libres de malla*.

En este trabajo se presenta una de estas alternativa denominada "*método de nube de puntos*".

Aquí la discretización de la función desconocida y sus derivadas se definen solo a partir de la posición (coordenadas) de los puntos dentro del dominio analizado. Esto lo hacemos mediante un polinomio de interpolación basado en los datos de las coordenadas de los puntos de pequeños dominios llamados nubes.

No es necesario que la cantidad de puntos dentro de estos dominios coincida con el grado del polinomio de aproximación como en el caso de elementos finitos, (generalmente el número de puntos en estos

dominios es mayor que el grado del polinomio) por lo que la función no pasa exactamente por los puntos del polinomio, sino que los aproxima. Para suavizar esta aproximación utilizamos la conocida técnica de mínimos cuadrados; esta interpolación será substancialmente mejorada en el dominio elegido (nube), mediante una función de peso (campana de Gauss). De esta forma nos queda una Aproximación por Mínimos Cuadrados Difusa, en nuestro caso aplicada a la ecuación de Laplace (flujo potencial) para problemas de fluidos. Este procedimiento lo repetimos para cada nodo del dominio total, armando en cada caso un dominio local alrededor del nodo en cuestión, este nuevo dominio se llama "nube del nodo i ".

Para evitar completamente la necesidad de una malla de elementos geoméricamente definidos, el sistema de ecuaciones es obtenido mediante la técnica estándar de colocación puntual.

Un criterio importante tenido en cuenta en este trabajo es el de mantener mínimo el ancho de banda para resolver el sistema de ecuaciones en forma directa y con la menor cantidad de operaciones posible. El almacenamiento de las matrices también se busco óptimo, para que sea posible resolver y almacenar en una PC un caso considerable de nodos.

El ejemplo utilizado para la aplicación del método es el clásico perfil alar simétrico NACA 0012 sin ángulo de ataque y dentro de un dominio rectangular Ω con flujo estacionario y condiciones de contorno Γ (método de líneas de corriente), como se ve en la Fig. 1.

Se han obtenido muy buenos resultados de C_p , que comparados con los resultados analíticos hablan de una muy buena aproximación sobre el perfil para los ejemplos corridos.

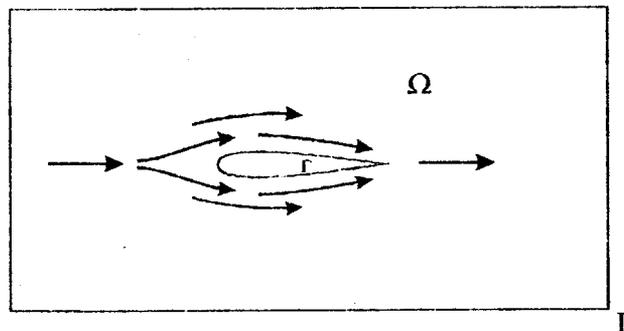


Figura 1. Perfil NACA 0012 en un dominio rectangular con flujo estacionario.

2. TEORÍA Y FUNDAMENTOS DEL MÉTODO

Si tenemos el siguiente sistema de ecuaciones diferenciales:

$$A(u) = b \quad \text{en } \Omega \quad (1)$$

con condiciones de contorno:

$$B(u) = t \quad \text{en } \Gamma_t$$

$$y \quad u - u_p = 0 \quad \text{en } \Gamma_u \quad (2)$$

Que se cumplen sobre un dominio Ω y sobre un contorno $\Gamma = \Gamma_t + \Gamma_u$.

Donde A y B son operadores diferenciales, u es el vector de incógnitas, b y t representan las fuerzas externas o fuentes actuando sobre el dominio Ω y a lo largo del contorno Γ_t respectivamente. Finalmente, u_p es el valor prescripto de u sobre el contorno Γ_u .

En la [Ref 1, Cap. 10.2] se presenta la formulación de Laplace.

Un procedimiento general para resolver numéricamente el sistema diferencial de ecuaciones es el método de los residuos ponderados [Ref. 1, Cap 9], en donde una función desconocida u es aproximada mediante algún proceso al valor \hat{u} , a partir de allí y reordenando, las ecuación (1) y (2) quedan como:

$$\int_{\Omega} W_i (A(\hat{u}) - b) d\Omega + \int_{\Gamma_t} \bar{W}_i (B(\hat{u}) - t) d\Gamma_t + \int_{\Gamma_u} \bar{W}_i (\hat{u} - u_p) d\Gamma_u = 0 \quad (3)$$

En donde las funciones de peso W_i , \bar{W}_i y \bar{W}_i pueden ser definidas de distintas maneras.

Los métodos FEM, FVM y DF pueden ser considerados casos particulares de la ecuación (3).

Para darle características locales al problema, la función u puede ser aproximada por una combinación de funciones definidas localmente como

$$u(x) \equiv \hat{u}(x) = \sum_{i=1}^{n_p} N_i(x) u_i^h = N^T(x) u^h \quad (4)$$

En donde n_p es el número total de puntos en el dominio Ω y donde

$$N_i(x) \neq 0 \quad \text{si} \quad x \in \Omega_i \quad (5)$$

$$N_i(x) = 0 \quad \text{si} \quad x \notin \Omega_i \quad (6)$$

Aquí, Ω_i es un subdominio de Ω , este tendrá n puntos siendo siempre $n \ll n_p$, recordemos que Ω tiene n_p puntos.

En la ecuación (4), u_i^h es el valor de u en el punto i .

En el FEM este valor coincide con el valor exacto de la función en el punto, $\therefore u(x_i) = u_i^h$.

Para el Método de Nube de Puntos, este valor se aproxima al valor exacto de la función en el punto. Esta aproximación se hace con una función suave que evita los saltos si los valores de los puntos vecinos son muy diferentes. El estudio y selección de esta función es uno de los puntos mas importantes en la aplicación de este método. \therefore en el Método de Nube de Puntos $u(x_i) \approx u_i^h$.

Por otro lado, en el FEM y en el FVM el subdominio Ω_i también es el dominio de interpolación, siendo en estos casos el dominio total Ω dividido en elementos, por lo que la función N_i puede tener algunas discontinuidades (la función misma o su derivada) en los bordes de los elementos.

Una característica común del FEM y del FVM es que ambos necesitan una malla para interpolar y para la posterior integración numérica de la ecuación (3) [Ref.1 Cap. 8.9 y 8.10].

Es la intención de este trabajo independizarnos totalmente de la malla, proponiendo un método que se encuadre entre los denominados métodos libres de malla. En la Fig. 2 vemos el perfil NACA 0012 dentro de un dominio cubierto por una grilla de puntos de los que usaremos sus coordenadas.

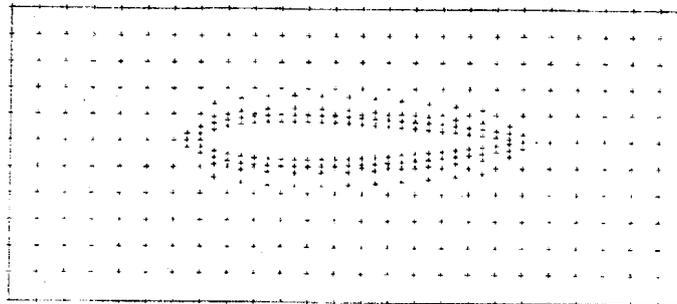


Figura 2. Perfil NACA 0012 dentro de un dominio cubierto por una grilla de puntos.

2.1. GENERALIDADES DE LOS MÉTODOS LIBRES DE MALLAS

Básicamente un método libre de malla debe satisfacer las siguientes condiciones:

- 1) La discretización de la función desconocida y de sus derivadas deben ser definidas solo por la posición de los puntos dentro del dominio analizado.
- 2) a) No es necesario un volumen o superficie de integración, o
b) Cualquier volumen o superficie de integración debe ser independiente del procedimiento de interpolación tomado.
- 3) La función de peso y su derivada debe ser definida solo por la posición de los puntos dentro del dominio analizado.

A la condición 2) a) la satisfacen los procedimientos de diferencias finitas y de colocación puntual.

La condición 2) b) será satisfecha por aquellos métodos con dominios de colocación, o aquellos otros que usen grillas de fondo para la integración.

La condición 1) es cumplida por el método de Rayleigh-Ritz [Ref. 2], en donde la función de forma N_i es definida sobre todo el dominio Ω , evitando así las características locales de la aproximación.

El FEM define las funciones de forma N_i superponiendo regiones (ensamblando elementos), que unidos constituyen el dominio Ω . Es por esto que diferentes interpolaciones serán posibles para un número de puntos dados, simplemente cambiando la orientación o la forma de estas regiones, de esta forma violamos la condición 1).

Por otro lado, aunque el FVM no define explícitamente una interpolación de la forma de la ecuación (4), es bien sabido que esto es equivalente a usar una función de forma lineal sobre Ω_i , definido de alguna manera como en el FEM.

En la próxima sección veremos una de las formas más conocidas de aproximación para realizar una interpolación numérica, basada en un número finito de puntos, la "aproximación por mínimos cuadrados".

2.2. INTERPOLACIÓN BASADA EN LOS DATOS DEL PUNTO

Dado un dominio de interpolación Ω_i de una función $u(x)$ y dado s_i donde $i=1,2,3,\dots,n$, siendo n la cantidad de puntos con coordenadas $x_i \in \Omega_i$.

La función desconocida u puede ser aproximada en Ω_i por un polinomio

$$u(x) \equiv \hat{u}(x) = \sum_{i=1}^m p_i(x) \alpha_i = p(x)^T \alpha \quad (7)$$

Donde α_i = coeficientes a determinar.

p_i = funciones de las variables. (En nuestro caso la base del polinomio).

En este caso, el vector $p(x)$ contiene los monomios del espacio de coordenadas, de forma tal que la base esté completa. Así, para un problema 2-D

$$p^T = [1, x, y] \quad m=3 \quad (8 a)$$

$$p^T = [1, x, y, xy, x^2, y^2] \quad m=6 \quad (8 b)$$

Estos vectores definen una base lineal y una cuadrática respectivamente. En este trabajo optamos por la base cuadrática como veremos más adelante.

Los coeficientes $\alpha = [\alpha_1, \alpha_2, \dots, \alpha_m]^T$ en (7) puede también ser función de x .

La función $u(x)$ puede entonces ser aproximada en los n puntos correspondientes a Ω_i .

$$u^h = \begin{Bmatrix} u_1^h \\ u_2^h \\ \vdots \\ u_n^h \end{Bmatrix} \equiv \begin{Bmatrix} \hat{u}_1 \\ \hat{u}_2 \\ \vdots \\ \hat{u}_n \end{Bmatrix} = \begin{Bmatrix} p_1^T \\ p_2^T \\ \vdots \\ p_n^T \end{Bmatrix} \alpha = C \alpha \quad (9)$$

En donde $u_i^h = u(x_i)$ son los valores desconocidos de la función u en el punto i , $\hat{u}_i = \hat{u}(x_i)$ son los valores aproximados y $p_i^T = p^T(x_i)$ los monomios que aproximan. En el FEM, el grado del polinomio que aproxima y la cantidad de puntos en el subdominio de aproximación coincide, $\therefore m = n$. En ese caso C es una matriz cuadrada y los coeficientes α se obtiene luego de igualar u^h con $C\alpha$ en (9). (Recordemos que en el FEM, la aproximación toma el valor de exacto de la función en el punto. $u_i^h = \hat{u}_i$).

$$\alpha = C^{-1}u^h \quad (10)$$

$$u \cong \hat{u} = p^T C^{-1}u^h = N^T u^h \quad (11)$$

Donde

$$N^T [N_1, \dots, N_n] = p^T C^{-1} \quad \text{y} \quad N_i = \sum_{j=1}^m p_j(x) C_j^{-1} \quad (12)$$

La función de forma $N_i(x)$ en el FEM, debe satisfacer la condición estándar

$$N_i(x_j) = 1 \quad i = j \quad (13)$$

$$N_i(x_j) = 0 \quad i \neq j \quad i, j = 1, 2, \dots, m$$

Si la nube Ω_i tiene 6 puntos y como habíamos optado por la base cuadrática, el polinomio pasará exactamente por estos puntos, por lo que también aquí será $u_i^h = \hat{u}_i$, pero si tiene mas puntos, 9 por ejemplo, el polinomio aproximará el valor de ellos. (En este trabajo se vió que 9 es la mejor cantidad de puntos en la nube para una buena aproximación del polinomio, $\therefore u_i^h \approx \hat{u}_i$).

2.3. APROXIMACIÓN POR MÍNIMOS CUADRADOS (LSQ)

Como $n > m$, C no es una matriz cuadrada y la aproximación no dará valores exactos de la función en los puntos, $u_i^h \neq \hat{u}_i$. Este problema puede resolverse simplemente si aproximamos los valores de u en los puntos; esto lo haremos minimizando la diferencia al cuadrado respecto de los parámetros α_i .

$$J = \sum_{j=1}^n (u_j^h - \hat{u}(x_j))^2 = \sum_{j=1}^n (u_j^h - p_j^T \alpha)^2 \quad (14)$$

Recordemos que n es la cantidad de puntos en la vecindad del punto y dentro de un subdominio Ω_i . (esto se denomina *nube de i*).

La ecuación (14) es una aproximación por mínimos cuadrados estándar (LSQ).

La minimización estándar dada es :

$$\alpha = \bar{C}^{-1} u^h \quad \text{con} \quad \bar{C}^{-1} = A^{-1} B \quad (15)$$

Donde

$$A = \sum_{j=0}^n p(x_j) p^T(x_j) \quad (16 a)$$

$$B = [p(x_1), p(x_2), \dots, p(x_n)] \quad (16 b)$$

La aproximación final será la dada por la ecuación (11), sustituyendo C por \bar{C} dada arriba. Las nuevas funciones de forma quedarán entonces

$$N_i = \sum_{j=1}^m p_j(x) \bar{C}_j^{-1} \quad (17)$$

Aquí, la función de forma N_i es armada a partir de las distancias, en coordenadas locales, de los nodos en cada nube respecto del nodo central (i) de esa nube. El nodo i se convierte en centro de coordenadas locales en cada nube. Por esto cada nube tendrá asociada una matriz $\bar{C}_{j,i}$.

$\bar{C}_{j,6}^i = (\text{número de puntos de la nube, base del polinomio elegido})^{\text{cantidad total de puntos en el dominio}}$
con

$$\bar{C} (j,1)^i = 1.$$

$$\bar{C} (j,2)^i = x \quad \text{Distancia en } x \text{ de } i \text{ a } j.$$

$$\bar{C} (j,3)^i = y \quad \text{Distancia en } y \text{ de } i \text{ a } j.$$

$$\bar{C} (j,4)^i = x^2 \quad \text{Distancia en } x \text{ al cuadrado de } i \text{ a } j.$$

$$\bar{C} (j,5)^i = x \cdot y \quad \text{Distancia en } x \text{ multiplicada por la distancia en } y.$$

$$\bar{C} (j,6)^i = y^2 \quad \text{Distancia en } y \text{ al cuadrado de } i \text{ a } j.$$

La aproximación LSQ es bastante popular para la interpolación de derivadas numéricas cuando tenemos datos de los puntos (diferencias finitas) sobre todo en mecánica de los sólidos y en mecánica de los fluidos. La principal desventaja de la aproximación LSQ es el rápido deterioro de la aproximación si el número de puntos n (nube) excede en mucho al número de m (términos del polinomio en p).

Esta dificultad puede ser superada si usamos una ponderación sobre las interpolaciones por mínimos cuadrados, esto se conoce como interpolación por mínimos cuadrados difusa, y es lo que veremos en la próxima sección.

2.4. APROXIMACIÓN POR MÍNIMOS CUADRADOS DIFUSA (DLS)

Como dijimos en la sección anterior, la aproximación LSQ puede ser mejorada sobre una región Ω_i , en donde las derivadas desconocidas son evaluadas a partir de las distancias entre sus coordenadas al cuadrado, si ponderamos, pesamos a la interpolación con una función $\varphi(x)$.

$$J = \sum_{j=1}^n \varphi(x_j) (u_j^h - \hat{u}(x_j))^2 = \sum_{j=1}^n \varphi(x_j) (u_j^h - p_j^T \alpha)^2 \quad (18)$$

La función $\varphi(x)$ es usualmente construida. Un camino es tomar un valor unitario en el nodo i , en donde la función (o su derivada) es computada, y que se valla desvaneciendo en una región Ω_φ circundante al nodo i .

Esta región Ω_φ define el número de puntos dentro de la región a interpolar (nube); se ha comprobado que produce buenas aproximaciones cuando tomamos 9 o 10 nodos bien distribuidos alrededor del nodo i .

Una típica elección para $\varphi(x)$ es la campana de Gauss.

A este tipo de aproximación le llamamos *aproximación difusa*. La minimización estándar de la ecuación (18) con respecto a los α , dados para la función de forma de (17) quedará ahora para A y B de la siguiente manera.

$$A = \sum_{j=1}^n \varphi(x_j) p(x_j) p^T(x_j) \quad (19)$$

$$B = [\varphi(x_1) p(x_1), \varphi(x_2) p(x_2), \dots, \varphi(x_n) p(x_n)] \quad (20)$$

$\varphi_j^i = (\text{número de puntos de la nube})^{\text{cantidad de puntos del dominio}}$

∴ definimos una campana de Gaus para cada nube Ω_i . En este punto es necesario hacer una aclaración: si durante el proceso de armado de la función de forma en cada nodo j , el programa detecta alguna singularidad en la nube toma un nodo mas. Por lo tanto j no es siempre el número de puntos definido como cantidad de nodos de la nube, sino que en algunos casos puede haber nubes con un valor de j mayor que el predefinido.

De esta manera en cada nube Ω_i tendré un valor de la función de forma $N_{i,j}^i$ para cada nodo j . Este valor lo podríamos almacenar en un vector, pues solo es función de j para cada nube, pero en la matriz N^i almaceno otros parámetros que utilizaré mas adelante.

$$\therefore N_{6,j}^i = N^i(6, \text{cantidad de nodos de la nube})$$

Aquí, cada fila l de la matriz guardará distintas operaciones de las funciones de forma de la nube Ω_i y en cada columna tendré la cantidad de puntos j que utilizó esa nube.

- $N^i(1, j)$ = guarda la función de forma del nodo j de la nube i .
- $N^i(2, j)$ = guarda la derivada primera en x de la función de forma del nodo j de la nube i .
- $N^i(3, j)$ = guarda la derivada primera en y de la función de forma del nodo j de la nube i .
- $N^i(4, j)$ = guarda la derivada segunda en x de la función de forma del nodo j de la nube i .
- $N^i(5, j)$ = guarda las derivadas primeras cruzadas en x y y de la función de forma del nodo j ...
- $N^i(6, j)$ = guarda la derivada segunda en y de la función de forma del nodo j de la nube i .

El producto de las derivadas cruzadas no lo usaremos posteriormente y por ello no lo guardamos, por lo tanto $l = 5$ y $N_{5,j}^i = N^i(5, \text{cantidad de nodos de la nube})$.

Una vez armada la matriz con datos de la función de forma para cada punto j perteneciente a cada nube Ω_i , calculo la matriz de rigidez \mathbf{K} y el vector de términos independientes \mathbf{b} . Esto lo veremos en la Sección 2.7.

2.5. DERIVADA DE LAS ECUACIONES DISCRETIZADAS

La elección de diferentes funciones de peso en la aplicación del método de los residuos ponderados en la ecuación (3), dará distintos grupos de ecuaciones discretizadas. Si nuestra premisa es mantenernos independientes del uso de la malla, la función de peso debe ser definida independiente de la necesidad de esta. La forma más simple para resolver este problema es la conocida colocación puntual, en donde $\bar{W}_i = \overline{\bar{W}}_i = \overline{\overline{\bar{W}}}_i = \delta_i$, δ_i es el operador delta de Dirac.

\therefore tendremos un grupo de ecuaciones para la ecuación (3) de la siguiente forma:

$$[A(\hat{u})]_i - b_i = 0 \quad \text{en} \quad \Omega \quad (21)$$

$$[B(\hat{u})]_i - t_i = 0 \quad \text{en} \quad \Gamma_t \quad (22)$$

$$u_i^h - u_p = 0 \quad \text{en} \quad \Gamma_u \quad (23)$$

En donde $\hat{u}_i = u_i^h$ en (23) solo en el contorno, para simplificar. Esto se verá con más detalle en la próxima sección. Esta opción también es utilizada en DF y nos lleva a un sistema de ecuaciones en donde podemos escribir la forma estándar:

$$\mathbf{K}\mathbf{u}^h = \mathbf{b} \quad (24)$$

Aquí, no es necesario preservar la simetría de la matriz de rigidez, el armado de dicha matriz está en la sección 2.7.; el vector \mathbf{u}^h es el de las incógnitas; \mathbf{b} es el vector que contiene las contribuciones de los términos de las fuerzas b , t y el vector \mathbf{u}_p de los valores prescriptos.

En la próxima sección de este capítulo discutiremos sobre las condiciones de contorno.

2.6. CONDICIONES DE CONTORNO

Una diferencia importante entre el FEM y el método de Nube de Puntos, es la introducción de las condiciones de contorno en nuestro modelo de línea de corriente

$$u - u_p = 0 \quad \text{en} \quad \Gamma_u \quad (25)$$

En el FEM esto se hace fácilmente forzando la solución de la ecuación para que

$$u^h(x_i) = u_p(x_i) \quad \text{para} \quad x_i \in \Gamma_u \quad (26)$$

En el método de Nube de Puntos este procedimiento no es tan simple

$$u^h(x_j) \neq \tilde{u}(x_j) \quad (27)$$

Distintas alternativas para imponer esta condición de contorno en el método de Nubes de Puntos son descritas en [Ref. 4, Cap. 5].

Aquí utilizamos la aproximación por mínimos cuadrados para imponer la condición de contorno.

Esta simple aproximación consiste en hacer cumplir sobre el punto la condición de contorno (como en el FEM). Los valores prescritos los impongo igualando el nodo correspondiente de la diagonal, nodo i de la nube, a un valor muy grande y el término independiente de esa fila igualado al mismo valor multiplicado por el valor prescrito. De esta manera al pasar al otro lado de la igualdad el valor de la diagonal se simplificará con el valor del término independiente y solo quedará el valor prescrito igualado a la incógnita en esa fila.

$$u^h(x_j) = \tilde{u}_p(x_j) \quad \text{para } x_j \in \Gamma_u \quad (28)$$

Note que en este caso es esencial que el mínimo cuadrado de la condición de contorno sea satisfecho en el sentido de que el Método de Nube de Puntos minimiza la distancia al cuadrado.

$$\varphi(x_j)[u^h(x_j) - \tilde{u}_p(x_j)]^2 \quad (29)$$

Por esto el sistema total de ecuaciones a resolver es de $n_p - n_u$ puntos, con n_p igual al total de número de puntos y n_u igual al número de puntos del contorno en estudio Γ_u .

En los puntos en donde tenemos condiciones de borde (normales prescritas), la rigidez tiene el valor de la fila de la matriz de forma de la derivada primera en x multiplicada por el valor de la velocidad prescrita en esa dirección mas la fila de la derivada primera en y multiplicada por el valor de la velocidad prescrita en esa dirección.

$$K_j^i = (N^i(2, j) * \text{valor prescrito en } x + N^i(3, j) * \text{valor prescrito en } y) \quad (30)$$

∴ en esa fila la rigidez tomará este valor.

2.7. RESOLUCIÓN DEL SISTEMA DE ECUACIONES

La matriz \mathbf{K} se arma usando los valores de la matriz N_{ij}^l donde $l = 5$ en cada nube.

$$K_j^i = (N^i(4, j) + N^i(5, j)) * 2 \quad (31)$$

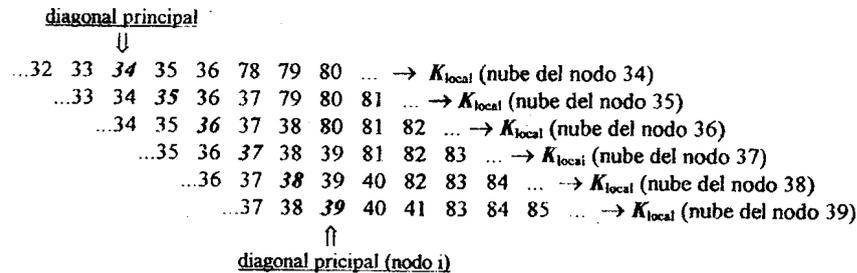
Como el elemento de N_{ij}^l que contenía las derivadas cruzadas no fue almacenado, en su lugar tendremos el valor de las derivadas segundas en y de la función de forma del nodo j en la nube i .

∴ cada una de estas filas tendrá tantas columnas como puntos tomó esa nube. Estos son aportes locales de cada nube Ω_i a la rigidez total del sistema.

En los puntos en donde tenemos condiciones de borde (normales prescritas), la rigidez toma otro valor, esto lo vimos en la sección 2.6., ecuación (30).

Un punto muy interesante en este método, a diferencia del FEM, es que aquí la matriz de rigidez global ($\mathbf{K}_{\text{global}} = \mathbf{K}$) no es ensamblada, es armada directamente por filas a partir de los aportes de rigidez locales ($\mathbf{K}_{\text{local}} = \mathbf{K}^i$).

$$\mathbf{K}_{i,j} = \mathbf{K}_{i,j} \quad (32)$$



Esta es la matriz de rigidez global. Los puntos de la matriz de rigidez global que están fuera de la nube en cada fila valen cero.

Una vez hecho esto, leo el ancho de banda hacia adelante y hacia atrás para cada nodo y obtengo el mayor ancho de banda total que utilizaré para resolver el sistema de ecuaciones. El ancho de banda queda en función de la nube con mayor diferencia entre el nodo j mas pequeño y el mas grande.

Una vez armada la matriz de rigidez K_{local} y el vector de términos independientes b , pasamos a resolver el sistema.

El algoritmo utilizado para resolver el sistema en forma directa es el conocido método de Gauss, pero en este caso optimizado para el ancho de banda leído.

Una vez resuelto el sistema y recordando la ecuación (4),

$$\hat{u}_i = \sum_{j=1}^m N_{i,j} * u_j^h \quad (33)$$

Donde u_j^h es el vector que obtuvimos al resolver el sistema de ecuaciones, m es la cantidad de nodos de la nube i , $N_{i,j}$ es la primera fila de la matriz de forma según lo vimos en la sección 2.4. y \hat{u}_i es el valor de la función potencial que queremos aproximar.

Finalmente se calculan las velocidades con las que se obtiene el CP sobre el perfil:

$$v_x = \sum_{j=1}^m N_{2,j} * u_j^h \quad (34)$$

$$v_y = \sum_{j=1}^m N_{3,j} * u_j^h \quad (35)$$

Sumando vectorialmente estas velocidades obtengo la velocidad en cada punto del dominio v_i . Como se ve hemos utilizado las cinco filas de la matriz N_{ij} ($i = 5$) almacenadas.

3. RESULTADOS

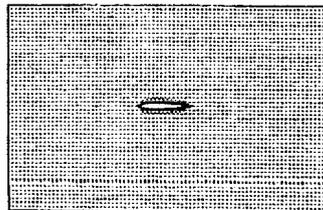


Figura 3. Dominio con el perfil NACA 0012. Sobre este dominio se calculó el CP.

.102	.117	.132	.147	.162	.177	.192
.101	.116	.131	<u>.146</u>	.161	.176	.191
.100	.115	<u>.130</u>	<u>.145</u>	<u>.160</u>	.175	.190
.99	<u>.114</u>	<u>.129</u>	<u>.144</u>	<u>.159</u>	<u>.174</u>	.189
.98	.113	<u>.128</u>	<u>.143</u>	<u>.158</u>	.173	.188
.97	.112	<u>.127</u>	<u>.142</u>	.157	<u>.172</u>	.187
.96	.111	.126	.141	.156	.171	.186

Figura 4. Detalle de un nube dentro del dominio. Los números subrayados representan la nube.

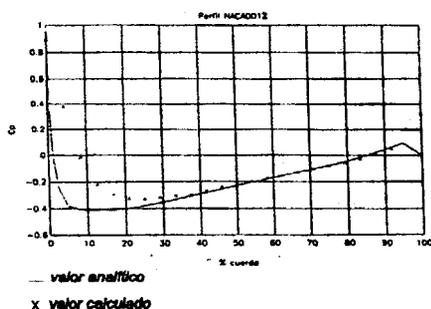


Figura 5. Resultados obtenidos para nubes de 8 puntos

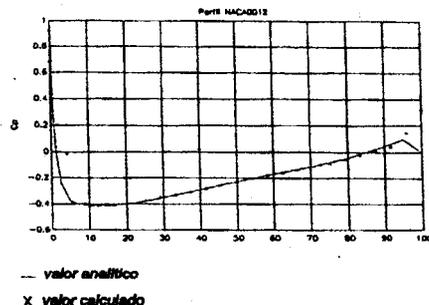


Figura 6. Resultados obtenidos para nubes de 9 puntos.

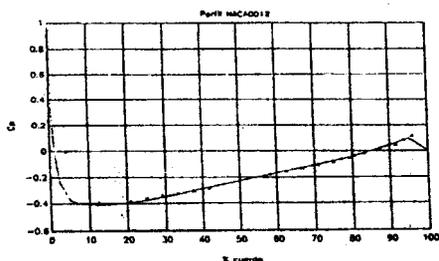


Figura 7. Resultados obtenidos para nubes de 10 puntos.

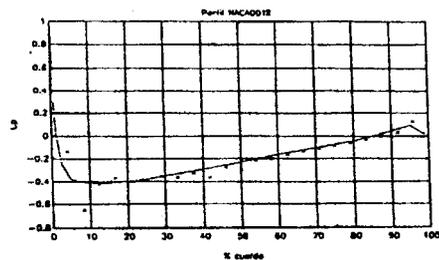


Figura 8. Resultados obtenidos para nubes de 12 puntos.

Los valores analíticos se obtuvieron de la [Ref. 3].

4. CONCLUSIONES

Los resultados obtenidos por este método presentan muy buenas aproximaciones a los valores analíticos como se ve en el capítulo anterior, aún cuando los puntos sobre el perfil son pocos. Es de esperarse iguales resultados para grillas no estructuradas e inclusive en 3D que es el objetivo último del estudio de este método. En este trabajo también se concluyó que las mejores aproximaciones se obtienen para nubes de 9 o 10 puntos.

Finalmente y en lo referente al armado de las nubes, es importante que estas sean lo mas centradas posible, es decir que tomen puntos en todas direcciones y no solo en un sentido, (excepto en los bordes del dominio) pues se observó que las nubes deformadas producen malos resultados.

5. AGRADECIMIENTOS

Agradecemos la colaboración del Ing. Carlos Sacco y del Grupo de Tecnología Mecánica del INTEC.

6. REFERENCIAS

1. O. C. Zienkiewicz, R. L. Taylor - El Método de los Elementos Finitos. Cuarta Edición Vol 1. 1994 Barcelona - España.
2. E. Oñate - Cálculo de estructuras por el método de Elementos Finitos. Barcelona - España.
3. Y. Abbott, A. Von Doenhoff - Theory of wing sections.
4. E. Oñate, S. Idelsohn, O. C. Zienkiewicz - Finite point methods in computational mechanics.