■ Apuntes Detallados de Métodos Numéricos en Ingeniería Aeroespacial (MNIA)

1. Fundamentos: Aproximación Polinomial y Derivación

A. La Necesidad del Polinomio
Los métodos numéricos utilizan polinomios porque son fáciles de manejar por el ordenador, ya que están definidos por un número finito de coeficientes. El objetivo es encontrar una solución o una aproximación de derivadas que satisfaga una relación con un error controlable.

Serie de Taylor: Es la base para aproximar una función f(x) alrededor de un punto x0 y es la herramienta principal para derivar las fórmulas de Diferencias Finitas.

Error de Truncamiento (R_n(x)): La diferencia entre la función exacta y su aproximación polinomial. Se expresa como R_n(x) = O((Δx)^p), donde p es el orden de precisión del método.

B. Derivación Numérica
La derivada de una función se reemplaza por una combinación lineal de los valores de la función en puntos discretos (el stencil).

Función PesiDer.m (Método de Taylor):
La matriz M se construye con expansiones de Taylor, donde cada fila es una ecuación del tipo:
$f(x_j) \approx \Sigma_{k=0}^{N-1} w_k * f^{(k)}(x_c)$
Se resuelve el sistema matricial M W = I para obtener la matriz de pesos W. El vector de pesos para la p-ésima derivada es la fila (p+1) de W.

Función DerivPoly.m (Método del Polinomio Interpolante):
El método se basa en que la derivada del polinomio interpolante P(x) es una aproximación de la derivada de la función original f(x).
La clave está en que la traslación c = x - xc simplifica la derivada: el valor de la p-ésima derivada del polinomio P(x) en x=xc es simplemente p! veces el coeficiente a_{p+1} de la forma canónica:
$P(x) = \Sigma_{k=0}^{n} a_k (x-xc)^k$.

---

2. Interpolación Avanzada

A. Interpolación Monomial y Nodos de Chebyshev
Interpolación de Lagrange (Monomial): Consiste en encontrar un único polinomio de grado N-1 que pase por N puntos.

El método usa la matriz de Vandermonde (V a = f) para encontrar los coeficientes a.

Función de Runge: f(x) = 1/(1 + x^2). Ilustra el fenómeno de Runge: al aumentar el grado del polinomio con nodos equiespaciados, la interpolación diverge en los bordes.

Nodos de Chebyshev: Son la solución al fenómeno de Runge. Están concentrados cerca de los bordes para distribuir el error uniformemente.

B. Interpolación a Tramos (Spline)
Spline Cúbica: Polinomio cúbico por subintervalo, continua hasta la segunda derivada.

Grados de Libertad:

Spline Natural: S''(x0)=0, S''(xn)=0.
Spline Completa: S'(x0)=f'(x0), S'(xn)=f'(xn).

---

3. Ejemplos de Matlab Primordiales y Diferentes

Ejemplo 1: Difusión Estacionaria 1D con BCs Mixtas
Ecuación: -k u'' = p(x)
Dominio: x ∈ [0, L]
BCs: u(0)=T0 (Dirichlet), u'(L)=gL (Neumann)

Código — Tarea — Explicación:
A = -k/(Dx^2) * gallery('tridiag', N-2, 1, -2, 1);
→ Discretiza u'' con esquema central (1, -2, 1).

q(1) = q(1) + k/(Dx^2)*T0;
→ BC Dirichlet: se pasa al vector q.

A(N-2, N-2) = A(N-2, N-2) - k/(Dx^2);
→ BC Neumann: modifica el último coeficiente diagonal.

q(N-2) = q(N-2) + k/(Dx^2)*(g_L * Dx);
→ Término fuente por Neumann.

T = [T0; T; TL];
→ Orlar la solución añadiendo nodos frontera.

Ejemplo 2: Poisson 2D con Dominio Irregular
Ecuación: $\nabla^2 \phi = q$

G = numgrid('L', N);
→ Mapea el dominio. 0 = contorno, >0 = nodos internos.

L = -delsq(G);
→ Construcción de matriz Laplaciana.

N_incognitas = length(find(G));
→ Tamaño del sistema.

x = L \ q;
→ Resolver sistema.

X = G; X(G>0) = x;
→ Reconstrucción 2D.

Ejemplo 3: Difusión Transitoria 2D (Calor)
Ecuación: $\partial u/\partial t = \alpha \nabla^2 u$

Esquema FTCS:
$u_{i,j}^{m+1} = u_{i,j}^{m} + \Delta t * [\alpha \nabla^2 u]_{i,j}^{m}$

beta = alpha * dt / (h^2);
→ Estabilidad requiere beta ≤ 0.25.

L_op = gallery('poisson', N_int);
→ Operador Laplaciano.

I = speye(N_int^2);
→ Matriz identidad.

M_FTCS = I + beta * L_op;
→ Matriz de marcha temporal.

for m = 1:M_pasos
u_nueva = M_FTCS * u_actual;
→ Marcha temporal.

FIN — Apuntes completos sin simplificación.