Risposta in Frequenza

$$\mathbf{T}\mathbf{x} = w$$

$$\mathbf{T} = \mathbf{G} + s\mathbf{C}$$

$$\mathbf{T} = \mathbf{G} + j\omega\mathbf{C}$$

Metodi di soluzione di sistemi lineari
fattorizzazione LU (*scomposizione triangolare*)

$$\mathbf{Tx} = \mathbf{b}$$

$$\mathbf{T} = \mathbf{LU}$$

$$\mathbf{L} = \begin{bmatrix} l_{11} & & & & \\ l_{21} & l_{22} & & \mathbf{0} & \\ l_{31} & l_{32} & l_{33} & & \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ l_{n1} & l_{n2} & l_{n3} & \cdots & l_{nn} \end{bmatrix}$$

$$\mathbf{U} = \begin{bmatrix} 1 & u_{12} & u_{13} & \cdots & u_{1n} \\ & 1 & u_{23} & \cdots & u_{2n} \\ & & 1 & \cdots & u_{3n} \\ & \mathbf{0} & & & \vdots \\ & & & & 1 \end{bmatrix}$$

Soluzione

$$\mathbf{LUx} = \mathbf{b} \quad \rightarrow \quad \begin{array}{l} \mathbf{Ux} = \mathbf{z} \\ \mathbf{Lz} = \mathbf{b} \end{array}$$

si ricava **z** (forward substitution):

$$\begin{array}{ll} l_{11} z_1 & = b_1 \\ l_{21} z_1 + l_{22} z_2 & = b_2 \\ l_{31} z_1 + l_{32} z_2 + l_{33} z_3 & = b_3 \\ & \vdots \\ l_{n1} z_1 + l_{n2} z_2 + \cdots + l_{nn} z_n & = b_n. \end{array}$$

$$z_1 = b_1 / l_{11}$$

$$z_2 = (b_2 - l_{21} z_1) / l_{22}$$

$$z_3 = (b_3 - l_{31} z_1 - l_{32} z_2) / l_{33},$$

in generale:

$$z_1 = b_1 / l_{11}$$

$$z_i = \left(b_i - \sum_{j=1}^{i-1} l_{ij} z_j \right) \Big/ l_{ii}; \qquad i = 2, 3, \ldots, n.$$

fattorizzazione LU

si ricava **x** (back substitution)

$$
\begin{aligned}
x_1 + u_{12}x_2 + u_{13}x_3 + \cdots + u_{1n}x_n &= z_1 \\
x_2 + u_{23}x_3 + \cdots + u_{2n}x_n &= z_2 \\
&\vdots \\
x_{n-1} + u_{n-1,n}x_n &= z_{n-1} \\
x_n &= z_n.
\end{aligned}
$$

in generale:

$$x_n = z_n$$

$$x_i = z_i - \sum_{j=i+1}^{n} u_{ij}x_j; \qquad i = n-1, n-2, \ldots, 1.$$

fattorizzazione LU - vantaggi

- Calcolo del determinante: $$\det \mathbf{T} = \prod_{i=1}^{n} l_{ii}$$
- Se cambia solo il vettore **w** (termini noti) non occorre rieffettuare la scomposizione ma solo rieseguire le sostituzioni *forward e back*
- Sistemi trasposti, della forma $\mathbf{T}^t \mathbf{x} = \mathbf{c}$, possono essere risolti usando la stessa scomposizione LU del sistema di partenza (utile nel calcolo delle sensibilità)

Scomposizione: algoritmo di *Crout*

Onere computazionale equivalente ad eliminazione Gaussiana: $\propto \dfrac{n^3}{3}$

Funzioni di Rete

$$F(s) = \frac{Y_o(s)}{Y_i(s)} = \frac{N(s)}{D(s)} = \frac{\sum_{i=0}^{n} a_i s^i}{\sum_{i=0}^{m} b_i s^i}$$

$$\mathbf{TX} = \mathbf{W}$$

$$F = \mathbf{d}^t \mathbf{X}$$

$$F = \mathbf{d}^t \mathbf{T}^{-1} \mathbf{W} = \mathbf{d}^t \frac{\text{adj } \mathbf{T}}{\det \mathbf{T}} \mathbf{W} \longrightarrow$$ tutte le f.d.r. hanno lo stesso denominatore uguale a det **T**

$$\mathbf{T} = \mathbf{G} + s\mathbf{C} \longrightarrow$$ det **T** e adj **T** sono polinomi in s

Determinazione della Funzione di rete

poniamo:

$$s = s_i$$

applichiamo la fattorizzazione LU:

$$\mathbf{LUX} = \mathbf{W}$$

calcoliamo il denominatore:

$$D(s_i) = \det \mathbf{T}(s_i) = \det \mathbf{L}(s_i) = \prod_{j=1}^{\text{size of } L} l_{jj}$$

risolviamo il sistema (forward e back substitution):

$$F(s_i) = \frac{N(s_i)}{D(s_i)}$$

calcoliamo il numeratore:

$$N(s_i) = D(s_i)F(s_i)$$

possiamo ripetere il calcolo per vari valori di s
e poi determinare i due polinomi N(s) e D(s)
mediante una tecnica di interpolazione polinomiale

Interpolazione sul cerchio di raggio unitario

Supponiamo di conoscere *n*+1 punti:

$$s_i, \qquad y_i = f(s_i) \qquad i = 0,1,\ldots,n$$

e di voler determinare i coefficienti del polinomio che passa per questi punti:

$$P_n(s) = \sum_{j=0}^{n} a_j s^j$$

Si scelgono i punti s_i uniformemente spaziati sul cerchio di raggio unitario:

$$s_o = 1$$

$$s_k = \exp\left(\frac{2k\pi j}{n+1}\right) = w^k \qquad w = \exp\left(\frac{2\pi j}{n+1}\right)$$

si ottiene:

$$a_j = \frac{1}{n+1}\sum_{k=0}^{n} y_k w^{-jk}$$

Trasformata di Fourier Discreta (DFT)

$$y_k = \sum_{j=0}^{n} a_j w^{jk}$$

Stima dell’ordine del polinomio: n < = numero componenti reattivi.

Determinazione diretta di poli e zeri della Funzione di rete

- Algoritmo QZ

$$\det\left[s\mathbf{C}+\mathbf{G}\right]=\prod\left(\alpha_i+\beta_i s\right)$$

$$\alpha_i \;\; complessi$$

$$\beta_i \;\; reali$$

C.B. Moler, G.W. Stewart, *An algorithm for generalized matrix eigenvalue problems.* SIAM Journ. On Numerical Analysis, V. 10, N. 2, April 1973.