Metodo di Aberth

Esistono metodi iterativi che approssimano simultaneamente tutti gli zeri di un polinomio, uno di essi è il metodo di Aberth:

$$x_i^{(k+1)} = x_i^{(k)} - \frac{1}{\frac{p'(x_i^{(k)})}{p(x_i^{(k)}))} - \sum_{j=1, j\neq i}^{n} \frac{1}{x_i^{(k)} - x_j^{(k)}}}$$

per un dato valore di $\boldsymbol{x}^{(0)}$.

- Si può scegliere $x_k^{(0)} = \mathrm{e}^{(\alpha+2\pi k)\boldsymbol{i}/n}$
- Non si conoscono teoremi di convergenza globale del metodo

Metodo di Aberth: implementazione

Un polinomio è un vettore di $n+1$ numeri complessi da dichiarare

```
    COMPLEX(8),DIMENSION(n+1) ::  p
```

dove $p(1)$ è il termine noto e $p(n+1)$ è il coefficiente principale.

Per calcolare il valore di $p(x)$ e $p'(x)$ si può usare il metodo di Horner

```
y=p(n+1)
DO i=n,1,-1
   y=y*x+p(i)
END DO
```

```
z=p(n+1)*n
DO i=n,2,-1
   z=z*x+p(i)*(i-1)
END DO
```

Metodo di Aberth: implementazione

Per calcolare

$$\sum_{j=1, j\neq i}^{n} \frac{1}{x_i^{(k)} - x_j^{(k)}}$$

se xk è il vettore $\boldsymbol{x}^{(k)}$

```
sum=0
DO i=1,n
   IF (j/=i) sum=sum+1/(xk(i)-xk(j))
END DO
```

Metodo di Aberth: implementazione

Come criterio di arresto si può scegliere

$$\|\boldsymbol{x}^{(k)} - \boldsymbol{x}^{(k-1)}\| < \varepsilon$$

che in FORTRAN diventa

```
IF (MAXVAL(ABS(xnew-xold))<1.0d-12) EXIT
```

all'interno di un ciclo `DO`

Metodo di Strassen

Si possono moltiplicare due matrici 2×2

$$\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{bmatrix}$$

con sole 7 moltiplicazioni, con il metodo di Strassen.

Metodo di Strassen

$$
\begin{cases}
p_1 = (a_{11} + a_{22})(b_{11} + b_{22}) \\
p_2 = (a_{21} + a_{22})b_{11} \\
p_3 = a_{11}(b_{12} - b_{22}) \\
p_4 = a_{22}(b_{21} - b_{11}) \\
p_5 = (a_{11} + a_{12})b_{22} \\
p_6 = (a_{21} - a_{11})(b_{11} + b_{12}) \\
p_7 = (a_{12} - a_{22})(b_{21} + b_{22})
\end{cases}
$$

$$
\begin{cases}
c_{11} = p_1 + p_4 - p_5 + p_7 \\
c_{21} = p_2 + p_4 \\
c_{12} = p_3 + p_5 \\
c_{22} = p_1 + p_3 - p_2 + p_6
\end{cases}
$$

Metodo di Strassen

Il caso 2×2 è relativamente semplice

Se le matrici sono $2^k \times 2^k$ è possibile decomporle in blocchi ed applicare lo stesso algoritmo ricorsivamente ai blocchi.

- L'algortimo così ottenuto ha costo $O(n^{\log_2 7})$ anzichè il *solito* $O(n^3)$

L'implementazione consiste nel riscrivere le formule viste sopra ed usare la ricorsione

Per selezionare i sottoblocchi si usa la sintassi `a11=a(1:n/2,1:n/2)`