1. Fondamenti Matematici

1.1 Interpolazione Polinomiale

L'interpolazione polinomiale è un metodo matematico che cerca di costruire una funzione polinomiale che passi esattamente attraverso un insieme di punti dati. Data una serie di punti (x_0,y_0), ..., (x_n,y_n), dove gli x_i sono distinti, vogliamo trovare un polinomio p(x) tale che $p(x_i) = y_i$ per ogni i.

Il teorema fondamentale dell'interpolazione polinomiale ci assicura che esiste un unico polinomio di grado ≤ n che soddisfa queste condizioni. Questo polinomio è chiamato "polinomio interpolante".

1.2 La Base Polinomiale

Un polinomio di grado n può essere scritto in diversi modi:

1. Base Monomiale (standard):
   $p(x) = a_0 + a_1x + a_2x^2 + ... + a_nx^n$
2. Base di Chebyshev:
   $p(x) = c_0T_0(x) + c_1T_1(x) + ... + c_nT_n(x)$
   dove $T_i(x)$ sono i polinomi di Chebyshev

I polinomi di Chebyshev sono definiti dalla relazione di ricorrenza:

- $T_0(x) = 1$
- $T_1(x) = x$
- $T_{n+1}(x) = 2xT_n(x) - T_{n-1}(x)$

1.3 Matrice di Vandermonde

La matrice di Vandermonde è lo strumento matematico che ci permette di passare dai valori nei punti ai coefficienti del polinomio interpolante.

In base monomiale, la matrice ha la forma:

```
V = [1      x₀      x₀²     ...  x₀ⁿ    ]
    [1      x₁      x₁²     ...  x₁ⁿ    ]
    [1      x₂      x₂²     ...  x₂ⁿ    ]
    [...    ...     ...     ...  ...    ]
    [1      xₙ      xₙ²     ...  xₙⁿ    ]
```

Nella base di Chebyshev, invece, la matrice diventa:

```
V = [T₀(x₀)  T₁(x₀)  T₂(x₀)  ...  Tₙ(x₀)]
    [T₀(x₁)  T₁(x₁)  T₂(x₁)  ...  Tₙ(x₁)]
    [T₀(x₂)  T₁(x₂)  T₂(x₂)  ...  Tₙ(x₂)]
    [...     ...     ...     ...  ...   ]
    [T₀(xₙ)  T₁(xₙ)  T₂(xₙ)  ...  Tₙ(xₙ)]
```

1.4 Il Fenomeno di Runge

Il fenomeno di Runge è un problema fondamentale nell'interpolazione polinomiale che si manifesta quando si utilizzano punti equispaziati con polinomi di grado elevato.

Consideriamo la funzione di Runge classica:
$f(x) = 1/(1 + 25x^2)$ nell'intervallo [-1,1]

Quando tentiamo di interpolare questa funzione con punti equispaziati, si verificano oscillazioni indesiderate ai bordi dell'intervallo. Queste oscillazioni:

- Aumentano all'aumentare del grado del polinomio
- Sono più pronunciate vicino ai bordi dell'intervallo
- Non diminuiscono aggiungendo più punti

1.5 Punti di Leja

I punti di Leja sono una sequenza speciale di punti costruita per ottimizzare l'interpolazione polinomiale. La loro costruzione si basa sul seguente algoritmo:

1. Si sceglie un punto iniziale ξ_0
2. Per ogni nuovo punto ξ_k, si sceglie quello che massimizza:
   $|\det(VDM(\xi_0, ..., \xi_{k-1}, \xi))|$

Questa scelta ha diverse proprietà vantaggiose:

- Massimizza la stabilità numerica
- Riduce il fenomeno di Runge
- Produce una distribuzione di punti ottimale per l'interpolazione

1.6 Costante di Lebesgue

La costante di Lebesgue è un indicatore matematico della qualità dell'interpolazione. È definita come:

$\Lambda_n = \max\{\sum|l_i(x)|\}$, dove $l_i(x)$ sono i polinomi di Lagrange

Questa costante:

- Fornisce un limite superiore all'errore di interpolazione

- Cresce con il grado del polinomio
- È minima per certe distribuzioni di punti (come i punti di Chebyshev)

2. Implementazione in MATLAB

2.1 Funzioni Base di MATLAB

Nel nostro progetto, utilizziamo diverse funzioni MATLAB fondamentali:

```
linspace(a,b,n)    % Crea n punti equispaziati tra a e b
% Esempio: linspace(-1,1,5) produce [-1, -0.5, 0, 0.5, 1]

tic/toc            % Misura tempo di esecuzione
% Esempio:
tic;
% codice da misurare
tempo = toc;

clear, clc         % Pulizia workspace e console
```

2.2 Costruzione della Matrice di Vandermonde

La funzione `build_chebyshev_vandermonde` implementa la matrice di Vandermonde usando i polinomi di Chebyshev:

```
function V = build_chebyshev_vandermonde(x, d)
    n = length(x);
    V = zeros(n, d+1);
    for i = 1:n
        for j = 1:d+1
            V(i,j) = cos((j-1) * acos(x(i)));
        end
    end
end
```

Questa implementazione sfrutta la relazione:
$T_n(\cos(\theta)) = \cos(n\theta)$

2.3 Calcolo dei Punti di Leja

Il calcolo dei punti di Leja viene implementato in due modi:

1. Metodo diretto (DLP.m):

```
function dlp = DLP(x, d)
    n = length(x);
```

```
    dlp = zeros(d+1, 1);
    dlp(1) = x(1);

    for k = 1:d
        prod = ones(n, 1);
        for j = 1:k
            prod = prod .* abs(x - dlp(j));
        end
        [~, max_idx] = max(prod);
        dlp(k+1) = x(max_idx);
    end
end
```

2. Metodo con fattorizzazione LU (DLP2.m), più efficiente per gradi elevati.

2.4 Calcolo della Costante di Lebesgue

La funzione `leb_con` calcola la costante di Lebesgue:

```
function L = leb_con(z, x)
    n = length(z);
    L_vals = zeros(length(x), 1);

    for i = 1:length(x)
        sum = 0;
        for j = 1:n
            num = 1; den = 1;
            for k = 1:n
                if k ~= j
                    num = num * (x(i) - z(k));
                    den = den * (z(j) - z(k));
                end
            end
            sum = sum + abs(num/den);
        end
        L_vals(i) = sum;
    end

    L = max(L_vals);
end
```

3. Note sulla Stabilità Numerica

L'uso della base di Chebyshev invece della base monomiale standard porta diversi vantaggi:

1. Miglior Condizionamento:

- La matrice di Vandermonde in base di Chebyshev ha un numero di condizionamento molto inferiore
- Questo significa minore propagazione degli errori di arrotondamento

2. Stabilità dell'Interpolazione:
   - I coefficienti in base di Chebyshev hanno magnitudini più uniformi
   - L'errore di valutazione del polinomio è meglio controllato
3. Combinazione con Punti di Leja:
   - La distribuzione dei punti di Leja si adatta naturalmente alla base di Chebyshev
   - Insieme forniscono un'interpolazione robusta anche per gradi elevati

4. Problemi Comuni e Soluzioni

4.1 Instabilità Numerica

Problema: Perdita di precisione nei calcoli
Soluzione:

- Uso della base di Chebyshev
- Implementazione accurata dei punti di Leja
- Evitare l'inversione esplicita di matrici

4.2 Fenomeno di Runge

Problema: Oscillazioni ai bordi dell'intervallo
Soluzione:

- Uso di punti non equispaziati (Leja)
- Monitoraggio della costante di Lebesgue
- Limitazione del grado del polinomio quando necessario

4.3 Efficienza Computazionale

Problema: Tempi di calcolo elevati per gradi alti
Soluzione:

- Uso dell'algoritmo DLP2 basato su LU
- Implementazione efficiente della matrice di Vandermonde
- Sfruttamento delle proprietà ricorsive dei polinomi di Chebyshev