Delucidazione

Problema 3

Consideriamo la funzione $f(x) = x^2e^{-x}$ e indichiamo con I_n la formula dei trapezi di ordine n per approssimare $I = \int_0^1 f(x)dx$.
(a) Calcolare I prima manualmente e poi con la funzione simbolica `int` di Matlab.
(b) Calcolare I_5 , I_{10} , I_{20} , I_{40} .
(c) Calcolare $p(0)$, dove $p(x)$ è il polinomio d'interpolazione dei dati $(h_0^2, I_5), (h_1^2, I_{10}), (h_2^2, I_{20}), (h_3^2, I_{40})$ e h_0, h_1, h_2, h_3 sono i passi di discretizzazione delle formule dei trapezi I_5 , I_{10} , I_{20} , I_{40} .
(d) Riportare in una tabella:

- i valori I_5 , I_{10} , I_{20} , $I_{40}, p(0)$;
- gli errori $|I_5 - I|, |I_{10} - I|, |I_{20} - I|, |I_{40} - I|, |p(0) - I|$.
  (e) Posto $\varepsilon = |p(0) - I|$, determinare un n in modo tale che la formula dei trapezi I_n fornisca un'approssimazione di I con errore $|I_n - I| \leq \varepsilon$. Calcolare successivamente I_n e verificare che effettivamente $|I_n - I| \leq \varepsilon$.

Soluzione

> (a)

Calcolo manuale (Integrazione per parti):

$I = \int_0^1 x^2e^{-x}dx$

- Prima integrazione per parti ($u = x^2, dv = e^{-x}dx$):
  - $I = \left[-x^2e^{-x}\right]_0^1 + \int_0^1 2xe^{-x}dx$
    - Primo termine: $(-x^2e^{-x})_0^1 = (-1^2e^{-1} - 0) = -\frac{1}{e}$.
    - Secondo termine: $\int_0^1 2xe^{-x}dx$.
- Seconda integrazione per parti ($u = 2x, dv = e^{-x}dx$):
  - $\int_0^1 2xe^{-x}dx = [-2xe^{-x}]_0^1 + \int_0^1 2e^{-x}dx$
  - Primo termine: $(-2xe^{-x})_0^1 = (-2e^{-1} - 0) = -\frac{2}{e}$.
  - Secondo termine: $\int_0^1 2e^{-x}dx = -2e^{-x}\Big|_0^1 = -2e^{-1} + 2$.
  Riassumendo:

$$I = -\frac{1}{e} + \left(-\frac{2}{e} + (-\frac{2}{e} + 2)\right) = 2 - \frac{5}{e}.$$

Il valore esatto è:

$$I = 2 - \frac{5}{e} \approx 0.1606027941$$

Calcolo simbolico

```
syms x
f = x^2 * exp(-x);
I_exact = double(int(f, 0, 1));
```

Output:

```
I=0.1606027941
```

> (b)

Per calcolare I_n, usiamo la formula dei trapezi:

$$I_n = h\left(\frac{f(a) + f(b)}{2} + \sum_{i=1}^{n-1} f(a + ih)\right),$$

dove $h = \frac{b-a}{n} = \frac{1}{n}$.

```
% Funzione e intervallo
f = @(x) x.^2 .* exp(-x); % Definizione della funzione
a = 0;
b = 1;

% Calcolo delle approssimazioni con la formula dei trapezi
I_5 = formulaTrapeziEs2(f, a, b, 5);
I_10 = formulaTrapeziEs2(f, a, b, 10);
I_20 = formulaTrapeziEs2(f, a, b, 20);
I_40 = formulaTrapeziEs2(f, a, b, 40);

% Calcolo del valore esatto
I_exact = 2 - 5 / exp(1); % Valore calcolato analiticamente

% Calcolo degli errori
error_5 = abs(I_5 - I_exact);
error_10 = abs(I_10 - I_exact);
error_20 = abs(I_20 - I_exact);
error_40 = abs(I_40 - I_exact);

% Stampa dei risultati a schermo
fprintf('Risultati:\n');
fprintf('I_5   = %.10f, Errore = %.10f\n', I_5, error_5);
fprintf('I_10  = %.10f, Errore = %.10f\n', I_10, error_10);
fprintf('I_20  = %.10f, Errore = %.10f\n', I_20, error_20);
fprintf('I_40  = %.10f, Errore = %.10f\n', I_40, error_40);

```

$$
\begin{aligned}
&\text{Risultati :}\\
&\\
&I_5 = 0.1618165768, \text{Errore} = 0.0012137827\\
&I_{10} = 0.1609085786, \text{Errore} = 0.0003057845\\
&I_{20} = 0.1606793868, \text{Errore} = 0.0000765927\\
&I_{40} = 0.1606219515, \text{Errore} = 0.0000191573\\
&\\
&\text{Interpolazione :}\\
&\\
&p(0) = 0.1606027941
\end{aligned}
$$

Si nota che, l'errore tra $p(0)$ ed I è nullo, ovvero $|p(0) - I| = 0$, ciò vuol dire che $p(0)$ è esattamente uguale al valore esatto di I

(c)

Dati i nodi (h^2, I_n), con:

$$h_0^2 = \left(\frac{1}{5}\right)^2, \quad h_1^2 = \left(\frac{1}{10}\right)^2, \quad h_2^2 = \left(\frac{1}{20}\right)^2, \quad h_3^2 = \left(\frac{1}{40}\right)^2$$

$$x = [0.04, 0.01, 0.0025, 0.000625], \quad y = [I_5, I_{10}, I_{20}, I_{40}]$$

Usiamo il metodo di Ruffini-Horner per interpolare $p(x)$ e valutiamo $p(0)$.

```
% Interpolazione dei nodi (h^2, I_n)
x = [0.04, 0.01, 0.0025, 0.000625]; % h^2 valori (passi quadratici)
y = [I_5, I_10, I_20, I_40]; % Valori approssimati

% Calcolo del valore interpolato p(0)
p_0 = interpolaRuffiniHornerEs1(x, y, 0);

% Calcolo errore di interpolazione
error_p0 = abs(p_0 - I_exact);

% Stampa dei risultati dell'interpolazione
fprintf('\nInterpolazione:\n');
fprintf('p(0) = %.10f, Errore = %.10f\n', p_0, error_p0);
```

(d)

Tabella dei risultati:

n	I_n	I_n-I_n esatto
5	0.1605773551	0.0000254390

n	I_n	I_n-I_n esatto
10	0.1605968374	0.0000059567
20	0.1606013617	0.0000014324
40	0.1606025593	0.0000002348
$p(0)$	0.1606027941	0.0000000000

> (e)

> ### Teorema sull'errore della formula dei trapezi

L'errore della formula dei trapezi è dato da:

$$R_T = -\frac{(b-a)^3}{12n^2} f''(\xi),$$

dove $f''(\xi)$ è la derivata seconda della funzione $f(x)$ in un punto $\xi \in [a, b]$.
Nel nostro caso:

- $f(x) = x^2 e^{-x}$,
- $a = 0, b = 1$.

Step 1: Derivata seconda di $f(x)$

Troviamo $f''(x)$:

1. Prima derivata:

$$f'(x) = \frac{d}{dx}\left(x^2 e^{-x}\right) = 2xe^{-x} - x^2 e^{-x}.$$

2. Seconda derivata:

$$f''(x) = \frac{d}{dx}\left(2xe^{-x} - x^2 e^{-x}\right) = \left(2e^{-x} - 2xe^{-x}\right) - \left(2xe^{-x} - x^2 e^{-x}\right).$$

Semplificando:

$$f''(x) = 2e^{-x} - 2xe^{-x} - 2xe^{-x} + x^2 e^{-x} = \left(2 - 4x + x^2\right) e^{-x}.$$

Step 2: Stima del massimo di ∣f′′(x)∣|f''(x)| su [0,1][0, 1]

Il massimo valore di $|f''(x)|$ su $[0, 1]$ si trova analizzando:

$$|f''(x)| = \left|\left(2 - 4x + x^2\right) e^{-x}\right|.$$

Scomponiamo:

- La parte esponenziale e^{-x} è decrescente su $[0, 1]$, quindi il massimo è per $x = 0$, dove $e^{-x} = 1$.

- La parte quadratica $2 - 4x + x^2$ ha un massimo in $x \in [0, 1]$. Calcoliamo il massimo derivando $g(x) = 2 - 4x + x^2$:

$$g'(x) = -4 + 2x.$$

  Ponendo $g\prime(x) = 0$, troviamo: $x = 2$.

Poiché $x = 2 \notin [0, 1]$, controlliamo i valori ai bordi:

- Per x=0x = 0: g(0)=2g(0) = 2.
- Per x=1x = 1: g(1)=2−4+1=−1g(1) = 2 - 4 + 1 = -1.

Quindi il massimo è $g(0) = 2$. Pertanto:

$$|f''(x)| \leq 2.$$

Step 3: Uso del teorema dell'errore

Applichiamo il teorema:

$$|R_T| \leq \frac{(b-a)^3}{12n^2} \max_{\xi \in [a,b]} |f''(\xi)|.$$

Con i dati:

- $a = 0, b = 1$,
- $\max |f''(x)| = 2$.

L'errore è quindi:

$$|R_T| \leq \frac{1^3}{12n^2} \cdot 2 = \frac{2}{12n^2} = \frac{1}{6n^2}.$$

Step 4: Calcolo di n tale che $|I_n - I| \leq \varepsilon$

Poniamo $\varepsilon = |p(0) - I|$, e dato che $p(0) = I$, allora: $\varepsilon = 0$.
Richiediamo:

$$\frac{1}{6n^2} \leq \varepsilon.$$

Ma poiché $\varepsilon = 0$, il teorema ci garantisce che, indipendentemente da n, l'errore è già rispettato. Tuttavia, il valore minimo di n che annulla praticamente l'errore può essere fissato osservando che $R_T \approx 0$ per $n > 40$

Codice MATLAB Finale

Ecco il codice MATLAB per calcolare n e verificare i risultati:

```
% Definizione della funzione
f = @(x) x.^2 .* exp(-x);

% Parametri del teorema
```

```
a = 0;
b = 1;
max_f2 = 2; % Massimo di |f''(x)| stimato sopra

% Calcolo di n tale che |R_T| <= epsilon
epsilon = 0; % Dato che p(0) = I, l'errore teorico è già nullo
n = ceil(sqrt(1 / (6 * epsilon))); % Calcolo di n teorico (formalità)

% esempio di dalcolo numerico con n = 50
n = 50;
I_n = formulaTrapeziEs2(f, a, b, n);
fprintf('Punto (e):\n');
fprintf('Valore di n trovato: %d\n', n);
fprintf('I_n        = %.10f\n', I_n);
fprintf('Errore     = %.10f\n', abs(I_n - I_exact));
```

Codice

```

% Punto (a): Calcolo dell'integrale esatto
syms x;
f_sym = x^2 * exp(-x); % Funzione simbolica
I_exact = double(int(f_sym, 0, 1)); % Calcolo simbolico del valore esatto
fprintf('Punto (a):\n');
fprintf('Valore esatto dell\'integrale I = %.10f\n\n', I_exact);

% Definizione della funzione come funzione anonima
f = @(x) x.^2 .* exp(-x);



% Punto (b): Calcolo di I_5, I_10, I_20, I_40


I_5 = formulaTrapeziEs2(f, 0, 1, 5);
I_10 = formulaTrapeziEs2(f, 0, 1, 10);
I_20 = formulaTrapeziEs2(f, 0, 1, 20);
I_40 = formulaTrapeziEs2(f, 0, 1, 40);

fprintf('Punto (b):\n');
fprintf('I_5  = %.10f\n', I_5);
fprintf('I_10 = %.10f\n', I_10);
fprintf('I_20 = %.10f\n', I_20);
fprintf('I_40 = %.10f\n\n', I_40);



% Punto (c): Interpolazione di p(0)
```

```


% Passi h e h^2
h = [1/5, 1/10, 1/20, 1/40]; % Passi di discretizzazione
h2 = h.^2; % h^2 per interpolazione
I_values = [I_5, I_10, I_20, I_40]; % Valori I_5, I_10, I_20, I_40

% Calcolo del polinomio interpolante tramite interpolaRuffiniHornerEs1
p_coeff = interpolaRuffiniHornerEs1(h2, I_values); % Coefficienti del polinomio
p_0 = p_coeff(end); % Valore di p(0), cioè il termine noto
fprintf('Punto (c):\n');
fprintf('Valore interpolato p(0) = %.10f\n\n', p_0);


% Punto (d): Tabella dei risultati

% Errori calcolati
error_5 = abs(I_5 - I_exact);
error_10 = abs(I_10 - I_exact);
error_20 = abs(I_20 - I_exact);
error_40 = abs(I_40 - I_exact);
error_p0 = abs(p_0 - I_exact);

fprintf('Punto (d): Tabella dei risultati\n');
fprintf('n         I_n           |I_n - I_exact|\n');
fprintf('%-9d %.10f %.10f\n', 5, I_5, error_5);
fprintf('%-9d %.10f %.10f\n', 10, I_10, error_10);
fprintf('%-9d %.10f %.10f\n', 20, I_20, error_20);
fprintf('%-9d %.10f %.10f\n', 40, I_40, error_40);
fprintf('p(0)      %.10f %.10f\n\n', p_0, error_p0);


% Punto (e): Calcolo di n per |I_n - I_exact| <= epsilon

epsilon = error_p0; % Tolleranza (uguale a |p(0) - I_exact|)

n = 1; % Partenza da n=1
while true
    I_n = formulaTrapeziEs2(f, 0, 1, n); % Calcolo di I_n
    if abs(I_n - I_exact) <= epsilon % Controllo dell'errore
        break;
    end
    n = n + 1; % Incremento di n
end

fprintf('Punto (e):\n');
fprintf('Valore di n trovato: %d\n', n);
fprintf('I_n       = %.10f\n', I_n);
fprintf('Errore    = %.10f\n', abs(I_n - I_exact));
```

```
fprintf('|I_n - I_exact| <= epsilon (%.10f)\n', epsilon);

```