Lucrarea de laborator nr. 12

I. Scopul lucrării

Integrarea numerică

II. Conținutul lucrării

1. Formula generală de cuadratură numerică. Formula de cuadratură Newton - Côtes .
2. Formula dreptunghiurilor.
3. Formula trapezelor.
4. Formula lui Simson.
5. Formule pentru calculul aproximativ al unei integrale duble

III. Prezentarea lucrării

III.1. Formula generală de cuadratură numerică. Formula de cuadratură Newton-Cotes.

Fie f : [a, b] → **R** o funcție continuă. Ne punem problema să calculăm valoarea aproximativă a integralei $\int_a^b f(x)\rho(x)dx$, unde ρ : [a, b]→**R** este o funcție continuă strict pozitivă numită pondere. Considerăm $x_0, x_1, \ldots, x_n$ n+1 puncte distincte din intervalul [a, b], şi notăm $y_i = f(x_i)$ pentru orice i = 0,1,…n. Fie L_n polinomul Lagrange asociat lui f şi punctelor considerate:

$$L_n(x) = \sum_{i=0}^{n} y_i \frac{(x-x_0)(x-x_1)\ldots(x-x_{i-1})(x-x_{i+1})\ldots(x-x_n)}{(x_i-x_0)(x_i-x_1)\ldots(x_i-x_{i-1})(x_i-x_{i+1})\ldots(x_i-x_n)}$$

Înlocuind f prin L_n, obținem formula de aproximare

$$\int_a^b f(x)\rho(x)dx \approx \int_a^b L_n(x)\rho(x)dx$$

Reprezentăm punctele sub forma

$x_i = \frac{a+b}{2} + \frac{b-a}{2} t_i$, i ∈[-1, 1].

$$\int_a^b L_n(x)\rho(x)dx =$$

$$= \int_a^b \rho(x)\sum_{i=0}^n f(x_i)\frac{(x-x_0)(x-x_1)....(x-x_{i-1})(x-x_{i+1})...(x-x_n)}{(x_i-x_0)(x_i-x_1)....(x_i-x_{i-1})(x_i-x_{i+1})...(x_i-x_n)}dx$$

$$= \sum_{i=0}^n f(x_i)\int_a^b \rho(x)\frac{(x-x_0)(x-x_1)....(x-x_{i-1})(x-x_{i+1})...(x-x_n)}{(x_i-x_0)(x_i-x_1)....(x_i-x_{i-1})(x_i-x_{i+1})...(x_i-x_n)}dx$$

Facem schimbarea de variabilă

$$x = \frac{a+b}{2} + \frac{b-a}{2}t$$

şi obținem

$$\int_a^b L_n(x)\rho(x)dx =$$

$$\frac{b-a}{2}\sum_{i=0}^n f(x_i)\int_{-1}^1 \rho(\frac{a+b}{2} + \frac{b-a}{2}t)\frac{(t-t_0)\cdots(t-t_{i-1})(t-t_{i+1})\cdots(t-t_n)}{(t_i-t_0)\cdots(t_i-t_{i-1})(t_i-t_{i+1})\cdots(t_i-t_n)}dt.$$

Formula

$$\int_a^b f(x)\rho(x)dx \approx$$

$$\frac{b-a}{2}\sum_{i=0}^n f(x_i)\int_{-1}^1 \rho(\frac{a+b}{2} + \frac{b-a}{2}t)\frac{(t-t_0)\cdots(t-t_{i-1})(t-t_{i+1})\cdots(t-t_n)}{(t_i-t_0)\cdots(t_i-t_{i-1})(t_i-t_{i+1})\cdots(t_i-t_n)}dt.$$

se numeşte ***formula generală de cuadratură***.

Presupunem că f este de clasă C^{n+1}. Pentru a evalua restul formulei ținem seama că eroarea cu care polinomul de interpolare aproximează funcția este dată de

$$|f(x) - L_n(x)| \le \frac{M}{(n+1)!}\left|(x-x_0)(x-x_1)\cdots(x-x_n)\right|,$$

unde $M = \sup_{x\in[a,b]} |f^{(n+1)}(x)|$

$$R_n(f) = \int_a^b f(x)\rho(x)dx - \int_a^b \rho(x)L_n(x)dx = \int_a^b (f(x) - L_n(x))\rho(x)dx$$

$$|R_n(f)| = \left|\int_a^b (f(x) - L_n(x))\rho(x)dx\right| \le \int_a^b \left| f(x) - L_n(x) \right|\rho(x)dx$$

Procedând ca mai înainte obținem ***restul formulei generale de cuadratură***:

$$|R_n(f)| \le \frac{M}{(n+1)!}\left(\frac{b-a}{2}\right)^{n+2}\int_{-1}^{1}\rho\left(\frac{a+b}{2}+\frac{b-a}{2}t\right)\left|(t-t_0)(t-t_1)\cdots(t-t_n)\right|dt,$$

unde $M = \sup\limits_{x\in[a,b]} |f^{(n+1)}(x)|$.

Presupunem că $\rho \equiv 1$, punctele sunt echidistante şi că $x_0 = a$, $x_n = b$. Deci

$$x_i = a + \frac{b-a}{n} i,\ i = 0, 1, 2, \dots, n.$$

Aproximând sub integrală funcția cu polinomul de interpolare Lagrange şi făcând schimbarea de variabilă

$$x = a + \frac{b-a}{n} t,$$

obținem

$$\int_a^b f(x)\rho(x)dx \approx (b-a)\sum_{i=0}^{n} H_i^{(n)} f(x_i) = (b-a)\sum_{i=0}^{n} H_i^{(n)} f\left(a + i\frac{b-a}{n}\right)$$

unde

$$H_i^{(n)} = \frac{1}{n}\frac{(-1)^{n-i}}{(n-i)!\,i!}\int_0^n t(t-1)\dots(t-i+1)(t-i-1)\dots(t-n)dt$$

$$= \frac{1}{n}\frac{(-1)^{n-i}}{(n-i)!\,i!}\int_0^n \frac{t(t-1)\dots(t-n)}{t-i}dt$$

Numerele $H_i^{(n)}$ se numesc ***coeficienții Newton - Côtes*** iar formula se aproximare se numeşte ***formula Newton - Côtes***. Au loc relațiile

$$\sum_{i=0}^{n} H_i^{(n)} = 1 \text{ şi } H_i^{(n)} = H_{n-i}^{(n)},\ i = 0,1,\dots,n.$$

Dacă f este o aplicație de clasă C^{n+1} are loc următoarea formulă de evaluare a restului:

$$\left|\int_a^b f(x)\rho(x)dx - (b-a)\sum_{i=0}^{n} H_i^{(n)} f\left(a + i\frac{b-a}{n}\right)\right| \leq$$

$$\frac{1}{(n+1)!}\left(\frac{b-a}{n}\right)^{n+2}\int_0^n |t(t-1)(t-2)\cdots(t-n)|dt \sup_{x\in[a,b]} |f^{(n+1)}(x)|.$$

De reținut că există funcții continue pentru care şirul $\left((b-a)\sum_{i=0}^{n} H_i^{(n)} f(x_i)\right)_n$ nu converge către $\int_a^b f(x)dx$.

III.2. Formula dreptunghiurilor

Fie f : [a, b] → **R** o funcție de clasă C^1. Aplicăm formula generală de cuadratură pentru ρ≡1, n=0, $x_0 = \frac{a+b}{2}$ (deci $t_0 = 0$). Obținem

$$\int_a^b f(x)dx \approx (b-a)\, f\left(\frac{a+b}{2}\right)$$

cu o eroare $\leq \frac{(b-a)^2}{4} \sup_{x\in[a,b]} |f'(x)|$.

Considerăm o diviziune $(x_0, x_1, \ldots, x_n)$ a intervalului [a, b] cu puncte echidistante ($x_i = a + \frac{b-a}{n} i$, i = 0, 1, 2, …, n.) şi aplicăm pe fiecare subinterval $[x_i, x_{i+1}]$ formula de aproximare

$$\int_{x_i}^{x_{i+1}} f(x)dx \approx (x_{i+1}-x_i)\, f\left(\frac{x_i + x_{i+1}}{2}\right)$$

Ținând cont că $\int_a^b f(x)dx = \sum_{i=0}^{n-1}\int_{x_i}^{x_{i+1}} f(x)dx$, obținem următoarea formulă de cuadratură numită ***formula dreptunghiurilor***:

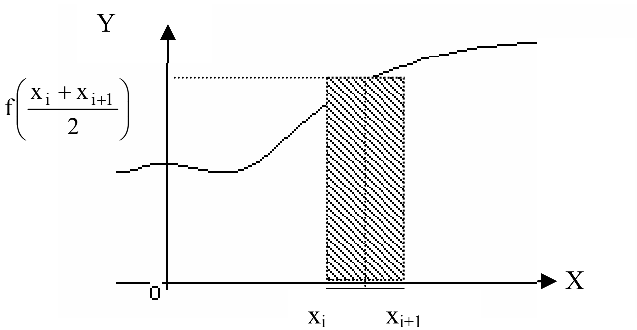
$$\int_a^b f(x)dx \approx \frac{b-a}{n}\sum_{i=0}^{n-1} f\left(\frac{x_i + x_{i+1}}{2}\right).$$

Restul (eroarea) este dat de:

$$\left|\int_a^b f(x)dx - \frac{b-a}{n}\sum_{i=0}^{n-1} f\left(\frac{x_i + x_{i+1}}{2}\right)\right| \leq \frac{(b-a)^2}{4n} \sup_{x\in[a,b]} |f'(x)|.$$

Interpretarea geometrică a formulei dreptunghiurilor

Fie D_i dreptunghiul cu o dimensiune dată intervalul $[x_i, x_{i+}]$ şi cu cealaltă dimensiune dată $f\left(\frac{x_i + x_{i+1}}{2}\right)$.



Atunci aria dreptunghiului D_i este

$$(x_{i+1}-x_i)\, f\left(\frac{x_i + x_{i+1}}{2}\right) = \frac{b-a}{n}\, f\left(\frac{x_i + x_{i+1}}{2}\right),$$

şi deci formula dreptunghiurilor presupune aproximarea $\int_a^b f(x)dx$ prin suma ariilor dreptunghiurilor D_i, i = 0, 1, …n-1.

Proceduri MAPLE pentru calculul valorii aproximative a unei integrale definite folosind formula dreptunghiurilor

Procedura `dreptunghiuri1` are drept parametri funcția care se integrează, limitele de integrare, şi numărul de subintervale din diviziune. Procedura întoarce valoarea aproximativă a integralei obținută aplicând formula dreptunghiurilor. Procedura `dreptunghiuri2` este similară, cu singura deosebire că în locul numărului de subintervale se introduce un număr pozitiv eps ce reprezintă eroarea maximă.

```
>dreptunghiuri1 := proc(f, a, b, n)
local i, iab, h, h0;
    iab := 0;
    h := (b - a)/n;
```

```
    h0 := a + 1/2*h;
    for i from 0 to n - 1 do
      iab := iab + evalf(f(h0 + i*h))
    od;
    iab := iab*h;
    RETURN(evalf(iab))
end;
```

```
>dreptunghiuri2 := proc(f, a, b, eps)
local i, iab, h, h0, n;
    n := floor(1/4*(b - a)^2/eps) + 1;
    print(`Numar de pasi`, n);
    iab := 0;
    h := (b - a)/n;
    h0 := a + 1/2*h;
    for i from 0 to n - 1 do
      iab := iab + evalf(f(h0 + i*h))
    od;
    iab := iab*h;
    RETURN(evalf(iab))
end;
```

Exemple

```
>f:=(x->x^7*ln(x)+x*cos(x));
```

$$f := x \to x^7 \ln(x) + x \cos(x)$$

```
> evalf(int(f(x),x=2..3));
                        778.3339881
> dreptunghiuri1(f,2,3,5);
                        768.4434052
> dreptunghiuri1(f,2,3,50);
                        778.2346342
> dreptunghiuri1(f,2,3,500);
                        778.3329938
> dreptunghiuri2(f,2,3,0.01);
                     Numar de pasi, 26
                                778.9666004
> dreptunghiuri2(f,2,3,0.001);
                     Numar de pasi, 251
                        778.3300450
```

```
> dreptunghiuri2(f,2,3,0.0001);
                              Numar de pasi, 2501
                                 778.3339477
> g:=(x->exp(-x^2));

                      g := x -> exp(-x^2)

> evalf(int(g(x),x=0..1));
                           .7468241330
> dreptunghiuri1(g,0,1,5);
                           .7480532524
> dreptunghiuri1(g,0,1,50);
                           .7468363956
> dreptunghiuri1(g,0,1,500);
                           .7468242530
> dreptunghiuri2(g,0,1,0.01);
                       Numar de pasi, 26
                           .7468694861
> dreptunghiuri2(g,0,1,0.001);
                      Numar de pasi, 251
                           .7468246211
> dreptunghiuri2(g,0,1,0.0001);
                      Numar de pasi, 2501
                           .7468241316
```

III.3. Formula trapezelor

Fie f : [a, b] → **R** o funcție de clasă C^2. Aplicăm formula generală de cuadratură pentru $\rho\equiv 1$, n=1, $x_0 = a$, $x_1 = b$ (deci $t_0 = -1$ şi $t_1 = 1$). Obținem

$$\int_a^b f(x)dx \approx \frac{b-a}{2}(f(a)+f(b))$$

cu o eroare $\leq \frac{(b-a)^3}{12}\sup_{x\in[a,b]}|f''(x)|$.

Considerăm o diviziune $(x_0, x_1, \ldots, x_n)$ a intervalului [a, b] cu puncte echidistante ($x_i = a + \frac{b-a}{n} i$, i = 0, 1, 2, …, n.) şi aplicăm pe fiecare subinterval $[x_i, x_{i+1}]$ formula de aproximare

$$\int_{x_i}^{x_{i+1}} f(x)dx \approx \frac{x_{i+1}-x_i}{2}(f(x_i)+f(x_{i+1}))$$

Ținând cont că $\int_a^b f(x)dx = \sum_{i=0}^{n-1}\int_{x_i}^{x_{i+1}} f(x)dx$, obținem următoarea formulă de cuadratură numită ***formula trapezelor***:

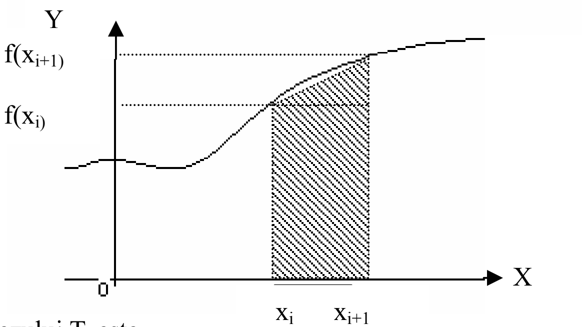
$$\int_a^b f(x)dx \approx \frac{b-a}{n}\left(\frac{f(a)+f(b)}{2}+\sum_{i=1}^{n-1} f(x_i)\right).$$

Restul (eroarea) este dat de:

$$\left|\int_a^b f(x)dx - \frac{b-a}{n}\left(\frac{f(a)+f(b)}{2}+\sum_{i=1}^{n-1} f(x_i)\right)\right| \le \frac{(b-a)^3}{12n^2}\sup_{x\in[a,b]}|f''(x)|.$$

Interpretarea geometrică a formulei trapezelor

Fie T_i trapezul dreptunghic cu înălțimea egală cu lungimea intervalului $[x_i, x_{i+]}]$ şi cu bazele $f(x_i)$ şi $f(x_{i+1})$.



Atunci aria trapezului T_i este

$$\frac{x_{i+1}-x_i}{2}\,(f(x_i) + f(x_{i+1})) = \frac{b-a}{2n}\,(f(x_i) + f(x_{i+1})),$$

şi deci formula trapezelor arată că $\int_a^b f(x)dx$ se poate aproxima prin suma ariilor trapezelor T_i, $i = 0, 1, \ldots n-1$.

Proceduri MAPLE pentru calculul valorii aproximative a unei integrale definite folosind formula trapezelor

Procedura `trapeze1` are drept parametri funcția care se integrează, limitele de integrare, şi numărul de subintervale din diviziune. Procedura întoarce valoarea aproximativă a integralei obținută aplicând formula trapezelor. Procedura `trapeze2` este similară, cu singura deosebire că în locul numărului de subintervale se introduce un număr pozitiv eps ce reprezintă eroarea maximă.

```
>trapeze1 := proc(f, a, b, n)
local i, iab, h;
    iab := 0;
    h := (b - a)/n;
    for i to n - 1 do
      iab := iab + evalf(f(a + i*h))
    od;
    iab := iab + 1/2*f(a) + 1/2*f(b);
    iab := iab*h;
    RETURN(evalf(iab))
end;
```

```
>trapeze2 := proc(f, a, b, eps)
local i, iab, h, n;
    n := floor(abs(1/12*(b - a)^3/eps)^(1/2)) + 1;
    print(`Numar de pasi`, n);
    h := (b - a)/n;
    iab := 0;
    for i to n - 1 do
      iab := iab + evalf(f(a + i*h))
    od;
    iab := iab + 1/2*f(a) + 1/2*f(b);
    iab := iab*h;
    RETURN(evalf(iab))
end;
```

Exemple

```
>f:=(x->x^7*ln(x)+x*cos(x));
```

$$f := x \to x^7 \ln(x) + x \cos(x)$$

```
> evalf(int(f(x),x=2..3));
                     778.3339881
> trapeze1(f,2,3,5);
                     798.1539466
> trapeze1(f,2,3,50);
                     778.5326999
```

```
> trapeze1(f,2,3,500);
                        778.3359754
> trapeze2(f,2,3,0.01);
                     Numar de pasi, 3
                        833.1348364
> trapeze2(f,2,3,0.001);
                     Numar de pasi, 10
                        783.2986759
> trapeze2(f,2,3,0.0001);
                     Numar de pasi, 29
                        778.9246586
> g:=(x->exp(-x^2));

                   g := x -> exp(-x^2)

> evalf(int(g(x),x=0..1));
                        .7468241330
> trapeze1(g,0,1,5);
                        .7443683397
> trapeze1(g,0,1,50);
                        .7467996064
> trapeze1(g,0,1,500);
                        .7468238866
> trapeze2(g,0,1,0.01);
                     Numar de pasi, 3
                        .7399864751
> trapeze2(g,0,1,0.001);
                     Numar de pasi, 10
                        .7462107961
> trapeze2(g,0,1,0.0001);
                     Numar de pasi, 29
                        .7467512252
> trapeze2(g,0,1,10^(-8));
                    Numar de pasi, 2887
                        .7468241295
```

III.4. Formula lui Simpson

Fie $f : [a, b] \to \mathbf{R}$ o funcţie de clasă C^4. Aplicăm formula generală de cuadratură pentru $\rho\equiv1$, $n=2$, $x_0 = a$, $x_1 = \frac{a+b}{2}$, $x_2 =b$ (deci $t_0 = -1$ şi $t_1 = 0$, $t_2 =1$). Obţinem

$$\int_a^b f(x)dx \approx \frac{b-a}{6}\left(f(a) + 4f\left(\tfrac{a+b}{2}\right) + f(b)\right)$$

Considerăm o diviziune (x_0, x_1, …., x_{2n}) a intervalului [a, b] cu puncte echidistante ($x_i = a + \frac{b-a}{2n} i$, i = 0, 1, 2, …, 2n.) şi aplicăm pe fiecare subinterval $[x_i, x_{i+2}]$ cu i par formula de aproximare

$$\int_{x_i}^{x_{i+2}} f(x)dx \approx \frac{x_{i+2}-x_i}{6}\left(f(x_i) + 4f(x_{i+1}) + f(x_{i+2})\right)$$

Ținând cont că $\int_a^b f(x)dx = \sum_{\substack{i=0 \\ i\ \text{par}}}^{2n-2} \int_{x_i}^{x_{i+2}} f(x)dx$, obținem următoarea formulă de cuadratură numită ***formula lui Simpson***:

$$\int_a^b f(x)dx \approx \frac{b-a}{6n}\left(f(a) + f(b) + 2\sum_{i=1}^{n-1} f(x_{2i}) + 4\sum_{i=1}^{n-1} f(x_{2i-1})\right).$$

Restul (eroarea) este dat de:

$$\left|\int_a^b f(x)dx - \frac{b-a}{6n}\left(f(a) + f(b) + 2\sum_{i=1}^{n-1} f(x_{2i}) + 4\sum_{i=1}^{n-1} f(x_{2i-1})\right)\right| \le$$

$$\frac{(b-a)^5}{2880n^4} \sup_{x\in[a,b]} \left|f^{(4)}(x)\right|.$$

Proceduri MAPLE pentru calculul valorii aproximative a unei integrale definite folosind formula lui Simson

Procedura `simpson1` are drept parametri funcția care se integrează, limitele de integrare, şi numărul de subintervale din diviziune. Procedura întoarce valoarea aproximativă a integralei obținută aplicând formula lui Simpson. Procedura `simpson2` este similară, cu singura deosebire că în locul numărului de subintervale se introduce un număr pozitiv eps ce reprezintă eroarea maximă.

```
>simpson1 := proc(f, a, b, n)
local i, iab, iab1, iab2, h;
    iab := evalf(f(a) + f(b));
    h := 1/2*(b - a)/n;
    iab1 := 0;
```

```
    iab2 := 0;
    for i to n - 1 do
     iab1 := iab1 + evalf(f(a + 2*i*h))
    od;
    for i to n do
      iab2 := iab2 + evalf(f(a + (2*i - 1)*h))
    od;
    iab := iab + 2*iab1 + 4*iab2;
    iab := 1/3*iab*h;
    RETURN(evalf(iab))
end;


>simpson2 := proc(f, a, b, eps)
local i, iab, iab1, iab2, h, n;
    n := floor(abs(1/2880*(b - a)^5/eps)^(1/4)) + 1;
    print(`Numar de pasi`, n);
    iab := evalf(f(a) + f(b));
    h := 1/2*(b - a)/n;
    iab1 := 0;
    iab2 := 0;
    for i to n - 1 do
      iab1 := iab1 + evalf(f(a + 2*i*h))
    od;
    for i to n do
     iab2 := iab2 + evalf(f(a + (2*i - 1)*h))
    od;
    iab := iab + 2*iab1 + 4*iab2;
    iab := 1/3*iab*h;
    RETURN(evalf(iab))
end;
```

Exemple

```
>f:=(x->x^7*ln(x)+x*cos(x));
```

$$f := x \to x^7 \ln(x) + x \cos(x)$$

```
> evalf(int(f(x),x=2..3));
                    778.3339881
> simpson1(f,2,3,5);
                    778.3469189
> simpson1(f,2,3,50);
```

```
                    778.3339893
> simpson1(f,2,3,500);
                    778.3339876
> simpson2(f,2,3,0.01);
                 Numar de pasi, 1
                    786.1020213
> simpson2(f,2,3,0.001);
                 Numar de pasi, 1
                    786.1020213
> simpson2(f,2,3,0.0001);
                 Numar de pasi, 2
                    778.8348129
> g:=(x->exp(-x^2));

               g := x -> exp(-x^2 )

> evalf(int(g(x),x=0..1));
                    .7468241330
> simpson1(g,0,1,5);
                    .7468249483
> simpson1(g,0,1,50);
                    .7468241323
> simpson1(g,0,1,500);
                    .7468241309
> simpson2(g,0,1,0.01);
                 Numar de pasi, 1
                    .7471804289
> simpson2(g,0,1,0.001);
                 Numar de pasi, 1
                    .7471804289
> simpson2(g,0,1,0.0001);
                 Numar de pasi, 2
                    .7468553799
> simpson2(g,0,1,10^(-8));
                 Numar de pasi, 14
                    .7468241459
```

III.5 Formule pentru calculul aproximativ al unei integrale duble

Fie φ_1, φ_2 : [a, b] $\to$ **R** două funcții de clsă C^1 şi fie

$$D = \{(x,y) \in R^2 \mid a \le x \le b,\ \varphi_1(x) \le y \le \varphi_2(x)\ \}.$$

Atunci D este un domeniu simplu. Considerăm o funcție continuă $f : D \to \mathbf{R}$. Ne punem problema aproximării integralei $\iint_D f(x,y)dxdy$. D fiind un domeniu simplu putem trece la integrale iterate:

$$\iint_D f(x,y)dxdy = \int_a^b \left(\int_{\varphi_1(x)}^{\varphi_2(x)} f(x,y)dy \right) dx = \int_a^b F(x)dx$$

unde F(x) = $\int_{\varphi_1(x)}^{\varphi_2(x)} f(x,y)dy$. Considerăm o rețea de noduri (x_i, y_j), i = 0, 1, …, n şi j = 0, 1, …, m. Aplicând integralei $\int_a^b F(x)dx$ o formulă de cuadratură se obține:

$$\iint_D f(x,y)dxdy \approx \sum_{i=0}^{n} A_i F(x_i)$$

unde

$$F(x_i) = \int_{\varphi_1(x_i)}^{\varphi_2(x_i)} f(x_i, y)dy .$$

Aplicând şi acestei ultime integrale o formulă de cuadratură:

$$F(x_i) \approx \sum_{j=0}^{m_i} B_{ij} f(x_i, y_j)$$

rezultă formula de aproximare

$$\iint_D f(x,y)dxdy \approx \sum_{i=0}^{n} \sum_{j=0}^{m_i} A_i B_{ij} f(x_i, y_j).$$

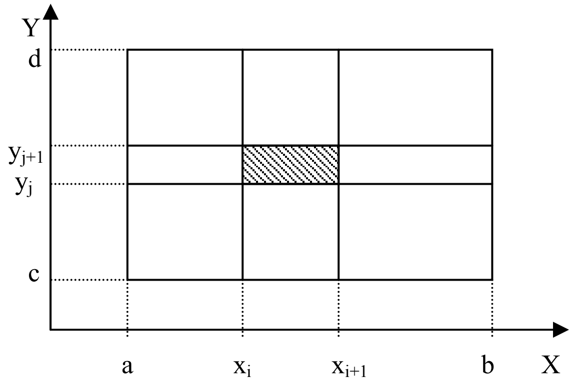
Să considerăm cazul în care D este un dreptunghi:

$$D = \{(x,y) \in \mathbf{R}^2 \mid a \le x \le b,\ c \le y \le d\ \}.$$

şi rețeaua este formată din punctele

$$x_i = a + ih,\ i = 0, 1, \dots, n,\ h = \frac{b-a}{n}$$

$$y_i = c + jk,\ j = 0, 1, \dots, m,\ k = \frac{d-c}{m}$$



Vom obține valoarea aproximativă a integralei funcției f pe dreptunghiul de vârfuri (x_i, y_i), (x_i, y_{i+1}) , (x_{i+1}, y_i) şi (x_{i+1}, y_{i+1}) prin aplicarea repetată a formulei trapezelor:

$$I_{ij} = \int_{x_i}^{x_{i+1}} \left(\int_{y_j}^{y_{j+1}} f(x, y) dy \right) dx \approx \int_{x_i}^{x_{i+1}} \left(\frac{k}{2} \left(f(x, y_j) \right) + f(x, y_{j+1}) \right) dx$$

$$= \frac{k}{2} \left(\int_{x_i}^{x_{i+1}} f(x, y_j) dx + \int_{x_i}^{x_{i+1}} f(x, y_{j+1}) dx \right)$$

$$\approx \frac{k}{2} \frac{h}{2} \left(f(x,_i\ y_j) + f(x,_{i+1}\ y_j) + f(x_i, y_{j+1}) + f(x_{i+1}, y_{j+1}) \right)$$

Astfel valoarea aproximativă a integralei

$$\iint_D f(x, y) dxdy =$$

$$\sum_{i=0}^{n-1} \sum_{j=0}^{m-1} I_{ij} \approx \sum_{i=0}^{n-1} \sum_{j=0}^{m-1} \frac{kh}{4} \left(f(x,_i\ y_j) + f(x,_{i+1}\ y_j) + f(x_i, y_{j+1}) + f(x_{i+1}, y_{j+1}) \right).$$

Presupumem că

$$x_i = a + ih,\ i = 0, 1, \ldots, 2n,\ h = \frac{b-a}{2n}$$

$$y_i = c + jk,\ j = 0, 1, \ldots, 2m,\ k = \frac{d-c}{2m}$$

Dacă pentru calculul integralei I_{ij} cu i şi j pare se aplică formula Simpson repetată se obține

$$I_{ij} = \int_{x_i}^{x_{i+2}} \left(\int_{y_j}^{y_{j+2}} f(x, y) dy \right) dx$$

$$\approx \int_{x_i}^{x_{i+2}} \left(\frac{k}{6} \left(f(x, y_j) + 4f(x, y_{j+1}) + f(x, y_{j+2}) \right) \right) dx$$

$$= \frac{k}{3} \left(\int_{x_i}^{x_{i+2}} f(x, y_j) dx + 4\int_{x_i}^{x_{i+2}} f(x, y_{j+1}) dx + \int_{x_i}^{x_{i+2}} f(x, y_{j+2}) dx \right)$$

$$\approx \frac{hk}{9} (f(x_i, y_j) + f(x_{i+2}, y_j) + f(x_i, y_{j+2}) + f(x_{i+2}, y_{j+2}) +$$
$$+ 4(f(x_{i+1}, y_j) + f(x_{i+1} y_{j+1}) + f(x_{i+2}, y_{j+1}) + f(x_{i+1}, y_{j+2})) +$$
$$+ 16f(x_{i+1}, y_{j+1}))$$

Notăm

$$F_{ij} = \frac{hk}{9} (f(x_i, y_j) + f(x_{i+2}, y_j) + f(x_i, y_{j+2}) + f(x_{i+2}, y_{j+2}) +$$
$$+ 4(f(x_{i+1}, y_j) + f(x_{i+1} y_{j+1}) + f(x_{i+2}, y_{j+1}) + f(x_{i+1}, y_{j+2})) +$$
$$+ 16f(x_{i+1}, y_{j+1}))$$

Atunci

$$\iint_D f(x, y) dxdy = \sum_{i=0}^{n-1} \sum_{j=0}^{m-1} I_{2i,2j} \approx \sum_{i=0}^{n-1} \sum_{j=0}^{m-1} F_{2i,2j}$$

Proceduri MAPLE pentru calculul valorii aproximative a unei integrale pe un dreptunghi

Procedura `dtrapeze` are drept parametri funcția care se integrează, limitele de integrare, şi numerele n şi m care determină punctele din rețea. Procedura întoarce valoarea aproximativă a integralei duble obținută prin aplicarea repetată a formulei trapezelor. Procedura `dsimpson` este similară, cu singura deosebire că se aplică repetat formula Simpson.

```
>dtrapeze := proc(f, a, b, c, d, n, m)
local i, j, iabcd, h, k;
    iabcd := 0;
    h := (b - a)/n;
    k := (d - c)/m;
    for i from 0 to n - 1 do for j from 0 to m - 1
do iabcd :=
          iabcd + evalf(f(a + i*h, c + j*k)
```

```
              + f(a + i*h, c + (j + 1)*k)
              + f(a + (i + 1)*h, c + j*k)
              + f(a + (i + 1)*h, c + (j + 1)*k))
         od
     od;
     iabcd := 1/4*iabcd*h*k;
     RETURN(evalf(iabcd))
end;



>dsimpson := proc(f, a, b, c, d, n, m)
local i, j, iabcd, h, k;
     iabcd := 0;
     h := 1/2*(b - a)/n;
     k := 1/2*(d - c)/m;
     for i from 0 to n - 1 do
       for j from 0 to m - 1 do
          iabcd :=
          iabcd + evalf(f(a + 2*i*h, c + 2*j*k)
          + f(a + (2*i + 2)*h, c + 2*j*k)
          + f(a + 2*i*h, c + (2*j + 2)*k)
          + f(a + (2*i + 2)*h, c + (2*j + 2)*k)
          + 4*f(a + (2*i + 1)*h, c + (2*j + 2)*k)
          + 4*f(a + (2*i + 1)*h, c + 2*j*k)
          + 4*f(a + 2*i*h, c + (2*j + 1)*k)
          + 4*f(a + (2*i + 2)*h, c + (2*j + 1)*k)
          + 16*f(a + (2*i + 1)*h, c + (2*j + 1)*k))
         od
     od;
     iabcd := 1/9*iabcd*h*k;
     RETURN(evalf(iabcd))
end;
```

Exemple

```
> f:=((x,y)->x^7*ln(x+y)+x*y*cos(x));
```

$$f := (x, y) \rightarrow x^7 \ln(x + y) + x\, y \cos(x)$$

```
> evalf(int(int(f(x,y),y=1..3),x=2..3));
                     2418.125737
> dtrapeze(f,2,3,1,3,5,10);
                     2470.913760
```

```
> dsimpson(f,2,3,1,3,5,10);
                        2418.156015
> dtrapeze(f,2,3,1,3,50,100);
                        2418.654858
> dsimpson(f,2,3,1,3,25,50);
                        2418.125786
> g:=((x,y)->exp(-x^2-y^2));
```

$$g := (x, y) \to \exp(-x^2 - y^2)$$

```
> evalf(int(int(g(x,y),y=0..1),x=0..1));
                        .5577462855
> dtrapeze(g,0,1,0,1,5,5);
                        .5540842250
> dsimpson(g,0,1,0,1,5,5);
                        .5577475033
> dtrapeze(g,0,1,0,1,50,50);
                        .5577096510
> dsimpson(g,0,1,0,1,50,50);
                        .5577462899
```

Probleme propuse

Studiați comenzile: leftsum, middlesum, trapezoid, simpson, leftbox, rightbox din pachetul student din MAPLE.