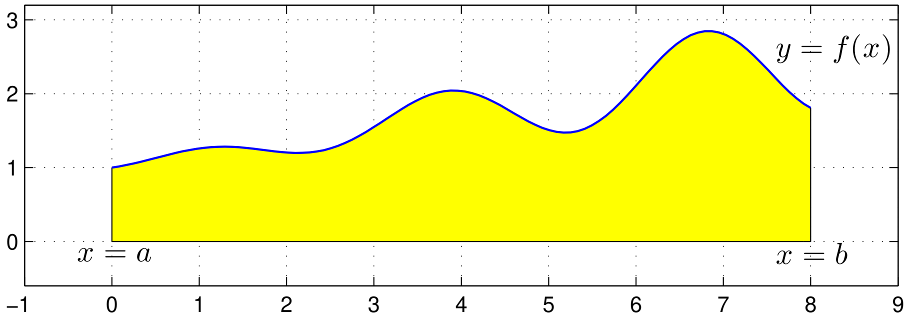
Integraler

1 Inledning

Ibland kan man inte bestämma integraler exakt utan man får nöja sig med att beräkna approximationer. T.ex. $\int_0^1 \mathrm{e}^{x^2}\,dx$ kan inte beräknas exakt, eftersom det inte finns någon användbar primitiv funktion. Det kan också vara så att integranden bara är känd i vissa punkter, t.ex. vi har en serie med mätdata.

2 Beräkningsmetoder

Den geometriska tolkningen av integralen $\int_a^b f(x)\,dx$ är arean av ytan mellan grafen av integranden $y = f(x)$ och x-axeln, dvs. $y = 0$, mellan $x = a$ och $x = b$.



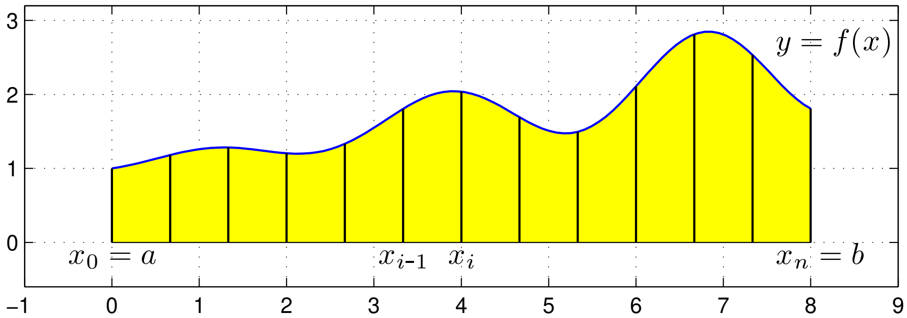
Vi gör en likformig indelning av intervallet $a \le x \le b$

$$a = x_0 < x_1 < x_2 < \cdots < x_{n-1} < x_n = b$$

så att vi får n lika långa delintervall $x_{i-1} \le x \le x_i$ med bredden $h = \frac{b-a}{n}$.

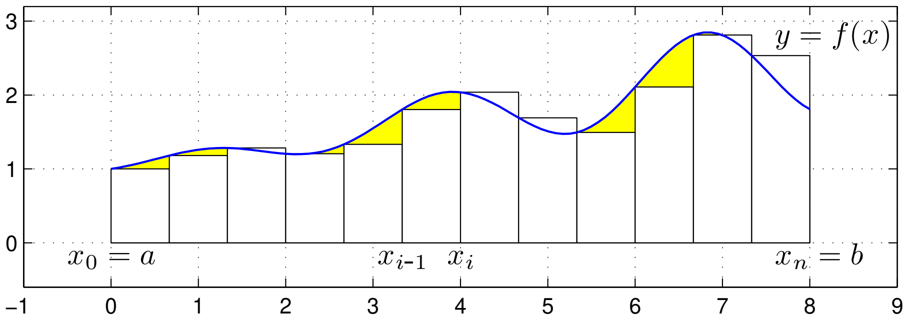
Sedan delar vi upp integralen i en summa av delintegraler över varje delintervall

$$\int_a^b f(x)\,dx = \sum_{i=1}^{n} \int_{x_{i-1}}^{x_i} f(x)\,dx$$



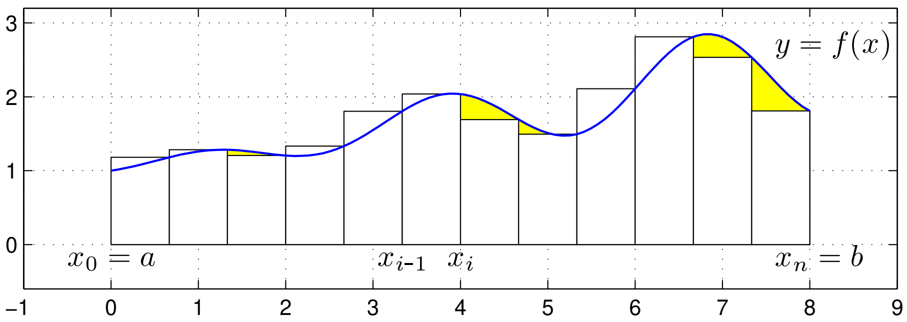
Om vi approximerar $f(x)$ med $f(x_{i-1})$ i intervallen $x_{i-1} \leq x \leq x_i$ får vi **vänster rektangelregel**

$$\int_a^b f(x)\,dx \approx \sum_{i=1}^{n} h\,f(x_{i-1})$$



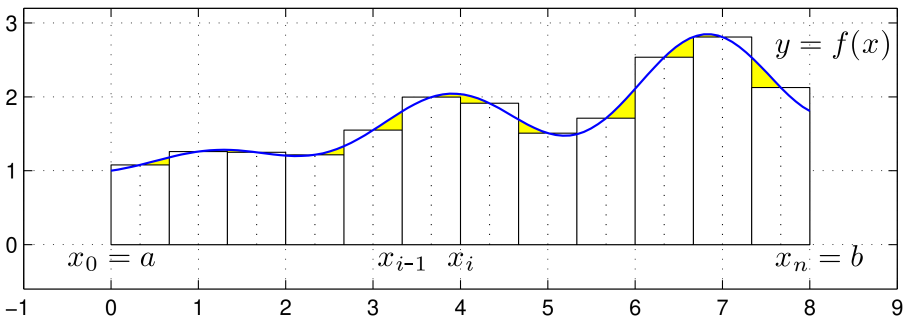
Om vi approximerar $f(x)$ med $f(x_i)$ i intervallen $x_{i-1} \leq x \leq x_i$ får vi **höger rektangelregel**

$$\int_a^b f(x)\,dx \approx \sum_{i=1}^{n} h\,f(x_i)$$



Om vi approximerar $f(x)$ med $f(m_i)$ i intervallen $x_{i-1} \leq x \leq x_i$, där m_i är mittpukterna i intervallen, får vi **mittpunktsmetoden**

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



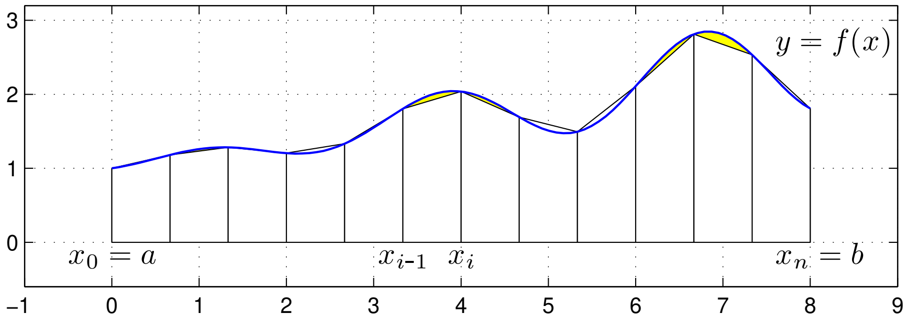
I Adams kapitel 5 definieras (konstrueras) integralen $\int_a^b f(x)\,dx$ med hjälp av Riemannsumman

$$\sum_{i=1}^{n} f(c_i)h_i$$

Metoderna ovan är olika varianter av Riemannsummor, med $c_i = x_{i-1}$, $c_i = x_i$ respektive $c_i = m_i$, och $h_i = h$.

Vi kan också approximera integralen med medelvärdet av vänster och höger rektangelregel och får då **trapetsmetoden**

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



Antag att vi vill beräkna $\int_0^1 x\sin(x)\,dx$ med vänster rektangelregel med $n = 100$. Vi skulle kunna göra så här

```
>> n=100;
>> a=0; b=1;
>> f=@(x)x.*sin(x);
>> h=(b-a)/n
>> q=0;
>> for i=0:n-1
        x=a+i*h;
        q=q+h*f(x);
   end
>> q
```

Att använda en `for`-sats är oftast inte effektivt i MATLAB. Vi genererar hellre en vektor av alla funktionsvärdena $f(x_i)$ och sedan summerar dessa enligt

```
>> n=100;
>> a=0; b=1;
>> f=@(x)x.*sin(x);
>> x=linspace(a,b,n+1);
>> h=(b-a)/n;
>> q=sum(h*f(x(1:n)))
```

Detta sätt att organisera en beräkning kallas att vektorisera den, dvs. man genererar först en eller flera vektorer och utför sedan den önskade beräkningen på dem. De komponentvisa operationerna `.*` `./` `.^` är exempel på vektoriserade operationer. Vi använde funktionen `sum` som snabbt summerar en vektor.

Uppgift 1. Beräkna en approximation av integralen $\int_0^1 x\,\sin(x)\,dx$ med vänster och höger rektangelregel samt mittpunkts- och trapetsmetoderna. Använd `sum`.

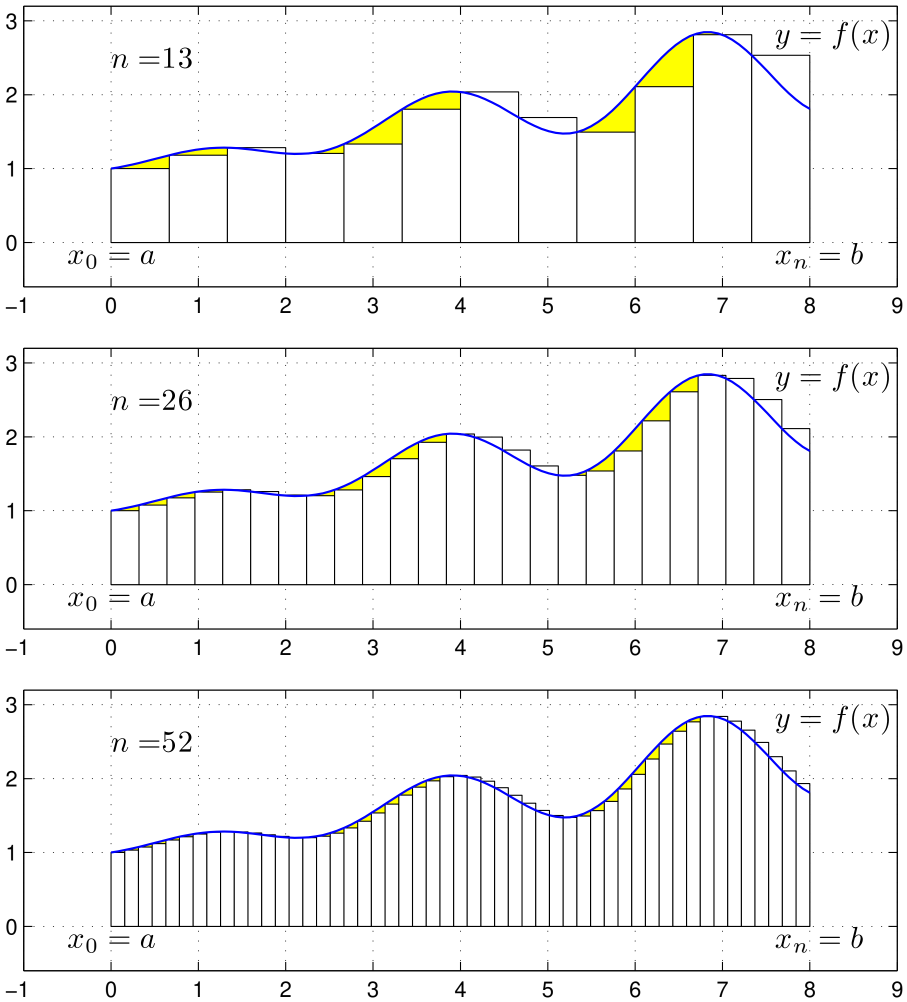
Uppgift 2. Skriv en funktion med namnet `min_integral` och anropet `q=min_integral(f,I,n,k)` som beräknar integralen approximativt. Du skall använda programskalet `min_integral.m` på MATLAB-hemsidan.

Uppgift 3. Testa ditt program på följande integraler. Variera metodval och antal delintervall n.

(a). $\int_0^1 \mathrm{e}^{-x^2}\, dx$ **(b).** $\int_{-1}^1 \frac{1}{1+x^2}\, dx$ **(c).** $\int_0^1 \tan(\sqrt{x})\, dx$

3 Konvergens

För metoderna ovan gäller att samtliga är konvergenta, dvs. låter vi antal delintervall n gå mot oändligheten så går approximationerna mot integralens värde. Vi ser på några bilder för vänster rektangelregel där n blir allt större



Vi ser att vi allt bättre täcker upp ytan under grafen med allt fler och smalare staplar.

Nu räcker det i praktiken inte med konvergens. Vi måste få en bra approximation på en kort tid, dvs. inte behöva ta n alltför stort.

För vänster och höger rektangelregel gäller att om vi fördubblar antal delintervall så halveras felet i approximationen av integralen. För mittpunkts- och trapetsmetoderna gäller vid samma fördubbling att felet delas med fyra, dvs. mycket bättre utdelning.

Uppgift 4. Vi ser på integralen $\int_0^1 x \sin(x)\, dx$ igen. Beräkna integralen exakt (för hand). Jämför exakt värde med de approximationer vi får med metoderna ovan för olika antal delintervall n. Hur stort blir felet? Tag t.ex först $n = 50$ och sedan $n = 100$, beräkna felen i approximationerna och se efter hur felen förändras.

4 Färdiga program i Matlab

Det finns färdiga funktioner i Matlab för att integrera. En sådan funktion är `integral`.

Vill vi beräkna integralen av $f(x) = x\sin(x)$ över intervallet $0 \leq x \leq 1$ med `integral` skulle det kunna göras så här

```
>> f=@(x)x.*sin(x);
>> a=0; b=1;
>> q=integral(f,a,b)
```

För att slippa problem, ta för vana att beskriva integranden som om du skulle rita dess funktionsgraf, dvs. tänk på `x` som en vektor och använd komponentvisa operationer.

Uppgift 5. Gör övning 17 i Adams 5.7. Rita en bild av området. Använd `integral`. Ledning: Se först på exempel 3 i samma kapitel.

Uppgift 6. Beräkna arean av det slutna området mellan graferna till funktionerna

$$g(x) = \mathrm{e}^{-\frac{x^2}{2}} \quad \text{och} \quad h(x) = x^2 - 3x + 2\,.$$

Rita en bild av området. Använd `fzero`, `fill` och `integral`.

1 Målsättning

Avsikten med denna laboration är att se lite på metoder för att approximera integraler $\int_a^b f(x)\,dx$. Många integrander, speciellt i samband med tekniska beräkningar, saknar användbara formler för primitiv funktion. Då återstår endast att approximera integralen. Vi skriver ett litet eget program för integralberäkning `min_integral`, där vi prövar några olika metoder. Avslutningsvis bekantar vi oss med `integral`, ett färdigt program för integralberäkning i Matlab.

2 Kommentarer och förklaringar

Vänster och höger rektangelregel kallas för *första ordningens* metoder. Fördubblar vi antalet delintervall så halveras felet, dvs. approximationen blir dubbelt så noggrann.

Mittpunkts- och trapetsmetoderna kallas för *andra ordningens* metoder. Fördubblar vi antalet delintervall så delas felet med fyra, dvs. approximationen blir fyra gånger så noggrann.

De senare metoderna är mycket mer effektiva. Detta är viktigt i tekniska beräkningar eftersom man sällan beräknar en enda integral utan i allmännhet en stor mängd integraler (och då som en del i en större beräkning).

Det färdiga programmet `integral` bygger på metoder som är mer avancerade är de vi tittat på och som vi inte har möjlighet att presentera här. Programmet är dessutom adaptivt, det innebär att integrationspunkterna inte placeras likformigt över intervallet utan där det lönar sig bäst med tanke på noggrannhet och effektivitet.

I uppgift 5 kan vi lätt se exakt var graferna skär varandra. Däremot i uppgift 6 måste vi bestämma skärningspunkterna med en beräkningsmetod för ekvationslösning, t.ex. `fzero`. Det går helt enkelt inte att skriva upp en användbar formel för skärningspunkterna.

3 Lärandemål

Efter denna laboration skall du kunna

- redogöra för den grundläggande idén bakom de olika metoderna för integralberäkning
- beräkna integraler $\int_a^b f(x)\,dx$, genom att beskriva f som en function i Matlab och beräkna en approximation med `min_integral` eller `integral`