

Méthodes informatiques pour physiciens

introduction à C++ et résolution de problèmes de physique par ordinateur

Leçon # 5 : Introduction à l'intégration numérique

Alessandro Bravar

Alessandro.Bravar@unige.ch

tél.: 96210

bureau: EP 206

assistants

Johanna Gramling

Johanna.Gramling@unige.ch

tél.: 96368

bureau: EP 202A

Mark Rayner

Mark.Rayner@unige.ch

tél.: 96263

bureau: EP 219

<http://dpnc.unige.ch/~bravar/C++2015/L5>

pour les notes du cours, les exemples, les corrigés, ...

Plan du jour #5

Récapitulatif et corrigé de la leçon #4

Intégration numérique

Formule du rectangle

Formule du trapèze

Formule de Simpson

exemples d'intégration numérique

* Différentiation numérique

différence finie

différence finie à droite et à gauche

différence finie centrale

Textes conseilles pour l'analyse numérique

M.-Y. Bachmann et al. (CRM)

Méthodes numériques

référence

W.H. Press et al.

Numerical Recipes

Récapitulatif de la leçon #4

Les Tableaux

unidimensionnel

```
double a[5] : a[0] . . . a[4]
```

multidimensionnel

```
double b[5][3]
```

opérations avec les tableaux (élément par élément !)

passage des tableaux aux fonctions

Les structures des données

Entrée / Sortie depuis un fichier

bibliothèque `fstream`

`ifstream`

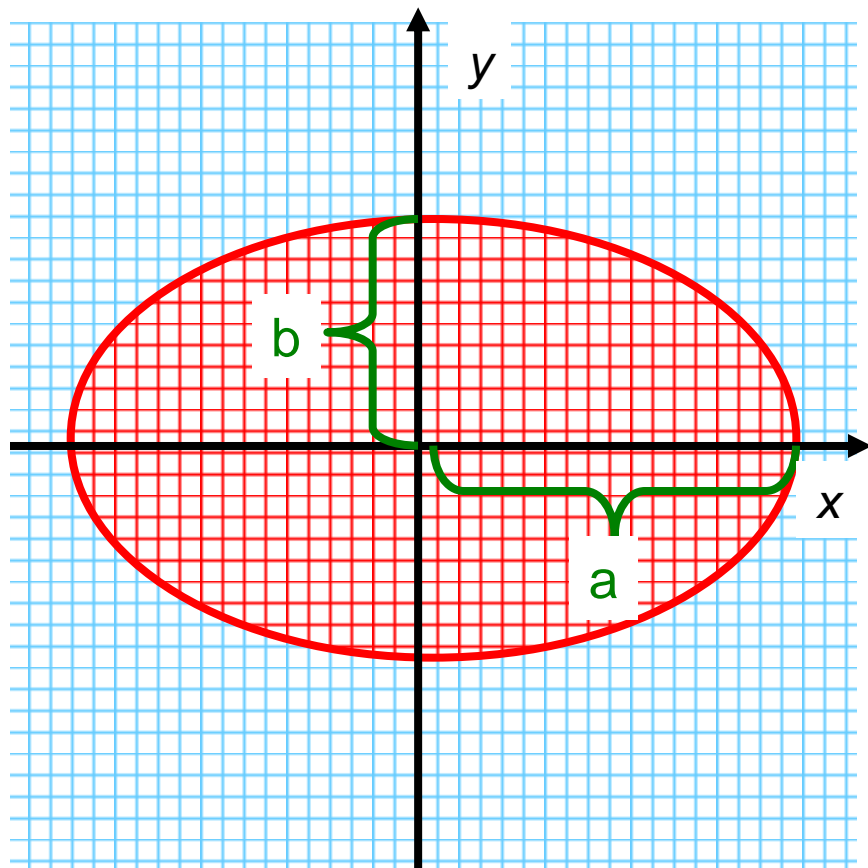
`ofstream`

Intégration numérique - introduction

Exemple: surface d'une ellipse $S = a \times b \times \pi$

a – axe majeur de l'ellipse

b – axe mineur de l'ellipse



Equation
de l'ellipse :

$$\left(\frac{x}{a}\right)^2 + \left(\frac{y}{b}\right)^2 = 1$$

Souvenez-vous du calcul de π !

Surface =

nombre de carrés inscrits dans l'ellipse \times
surface du carré

\Rightarrow surface sous-estimée $S(1)$

Pour compter les carrés inscrits dans
l'ellipse, il faut p. ex. écrire une boucle

voir [Ellipse.cpp](#)

Comment améliorer le résultat ?

- utiliser des carrés plus petits
- considérer tous les carrés dont le centre est dans l'ellipse
- considérer tous les carrés touchés par l'ellipse

\Rightarrow surface surestimée $S(2)$

Surface ? p.ex. $[S(1) + S(2)] / 2$

Erreur ? p.ex. $[S(1) - S(2)] / 2$

```

int main() {
    double a;    cin >> a;
    double b;    cin >> b;
    int na;     cin >> na;

    int somme = 0;
    double x, y, r;
    double dx = a / double(na);
    double dy = dx;
    int nb = int(b/dy) + 1;
    double x0 = dx/2.;
    double y0 = dy/2.;
    for (int ix=0; ix<=na; ix++) {
        for (int iy=0; iy<=nb; iy++) {
            x = ix*dx + x0;
            y = iy*dy + y0;
            r = (x/a * x/a) + (y/b * y/b);
            if (r<1.) somme++;
        }
    }

    double s = 4. * somme * (dx*dy);
    cout << "La surface calculee est " << s << endl;
    cout << "et la surface exacte est " << M_PI*a*b << endl;

    return 0;
}

```

saisie des paramètres
de l'ellipse

dimension du segment de l'axe a
on assume la même taille pour l'axe b

centre du premier carré

boucle sur les deux axes de l'ellipse
on « compte » le nombre de carrés
dans un seul quadrant et
on multiplie le résultat par 4

centre du carré i, j

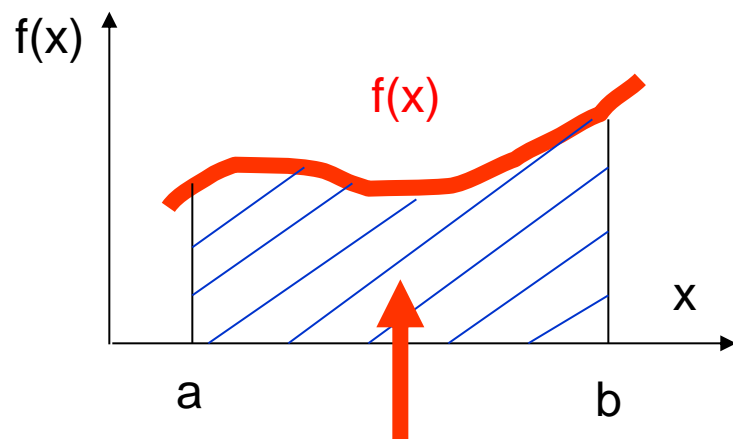
équation de l'ellipse
est-ce que le centre du carré
est dans l'ellipse ?

Intégration numérique

En général, il n'est pas toujours possible de trouver la primitive d'une fonction arbitraire ou d'utiliser celle-ci. Dans certains cas, la fonction à intégrer n'est connue que par les valeurs qu'elle prend sur un ensemble fini de points.

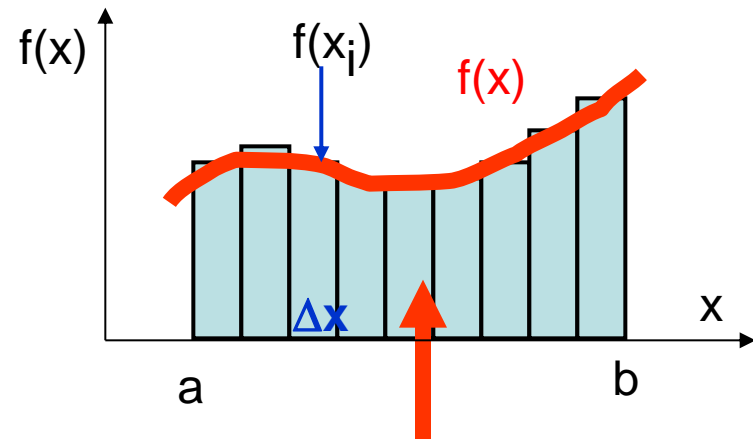
Dans ces cas, il faut considérer des méthodes numériques pour intégrer la fonction.

Du point de vue de l'ordinateur, l'intégration numérique est une opération plutôt simple et naturelle: c'est un peu comme dessiner une fonction sur le papier millimétrique et compter les carrés sous la courbe.



$$\int_a^b f(x) dx = F(b) - F(a)$$

\Rightarrow



$$\lim_{\Delta x \rightarrow 0} \sum_{i=1}^{i=n} f(x_i) \Delta x \quad \Delta x = \frac{b-a}{n}$$

Discrétisation du problème: on divise l'intervalle (a, b) en n sous-intervalles Δx

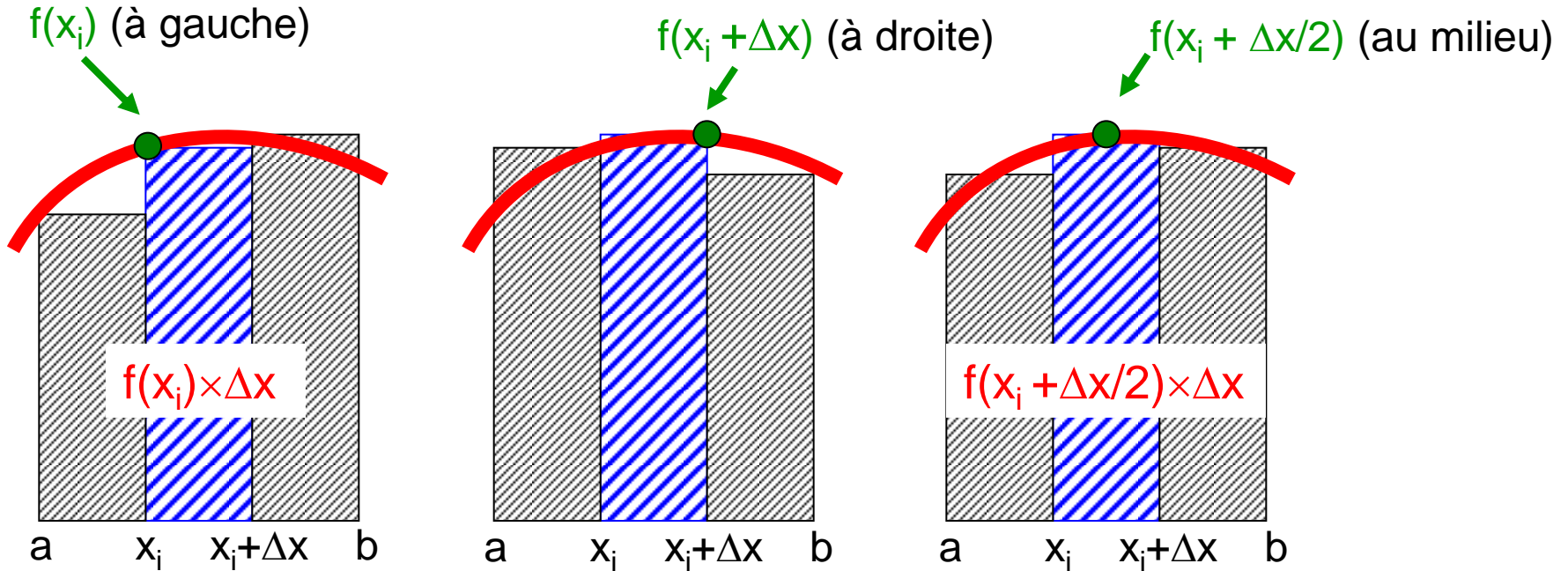
Q1 comment approximer $f(x)$?

Q2 comment choisir x_i et Δx ?

plus Δx est petit (grand nombre de segments), plus l'approximation sera précise

Intégration numérique

comment approximer $f(x)$ et où calculer $f(x_i)$?



$$I_G = \sum_0^{n-1} f(x_i) \cdot \Delta x$$

$$I_D = \sum_1^n f(x_i) \cdot \Delta x$$

$$I_M = \sum_0^{n-1} f(x_i + \Delta x/2) \cdot \Delta x$$

N : nombre de partitions de l'intervalle $[a, b]$

Les trois cas sont équivalents :

l'approximation est du même ordre !

Cette méthode est connue comme **Formule du Rectangle**.

Elle fonctionne si la fonction ne change pas trop vite.

Δx doit être adapté au taux de changement de la fonction.

$$\Delta x = (b - a) / N$$

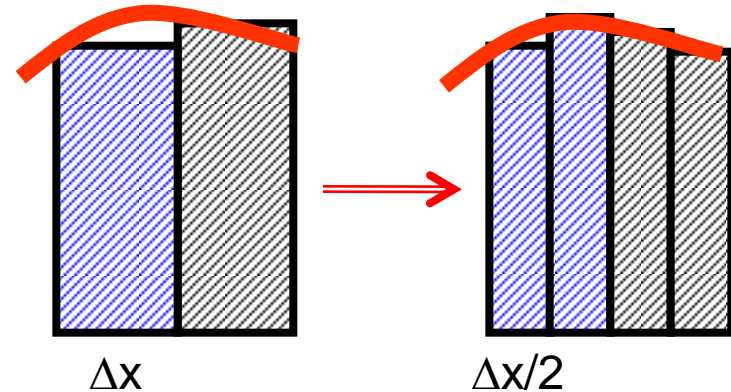
$$x_i = a + i \cdot \Delta x$$

Précision de la méthode

Pour améliorer la précision de l'approximation, on peut augmenter le nombre de segments :

$$n \rightarrow 2n \quad \text{et} \quad \Delta x \rightarrow \Delta x / 2$$

mais le nombre des calculs augmente !



Si $f'(x) > 0$ I_G sous-estime et I_D surestime l'intégrale : $I_G < I < I_D$

Si $f'(x) < 0$ I_G surestime et I_D sous-estime l'intégrale : $I_G > I > I_D$

Et dans tout les deux cas : $\min(I_G, I_D) < I < \max(I_G, I_D)$!

Donc, une approche simple est de calculer I_G et I_D et d'en prendre la moyenne (formule du Trapèze, voir plus tard), i.e. :

$$I = \frac{1}{2}(I_G + I_D)$$

Pour estimer l'erreur sur I , on peut considérer la différence entre I_G et I_D , i.e. :

$$\Delta I = \frac{1}{2}|I_G - I_D|$$

En général, on est intéressé à l'erreur relative (ou précision), i.e. :

$$\frac{\Delta I}{I} = \frac{|I_G - I_D|}{(I_G + I_D)}$$

p.ex. calculez I avec une précision de 3%

$$\Rightarrow \Delta I / I < 0.03$$

Implémentation

Pour intégrer la fonction $f(x)$ sur l'intervalle $[a,b]$:

1) on divise l'intervalle $[a,b]$ en N segments de longueur $\Delta x = |b-a| / N$

2) on écrit une boucle pour

calculer l'aire du "rectangle" $f(x_i) \times \Delta x$ dans les N points $f(a)$, $f(a+\Delta x)$, $f(a+2\Delta x)$, ...

et on additionne les N "rectangles" : $I = f(a) \Delta x + f(a+\Delta x) \Delta x + \dots + f(a+(N-1)\Delta x) \Delta x$

Formule du Rectangle à gauche

```
double a, b;
int N = 1000; //divisions de l'intervalle a-b
double deltax = fabs(b-a)/N; //longueur de l'intervalle

//calcul de l'integrale
double integrale = 0.;
for (int i=0; i<N; i++) //rectangle a gauche
    integrale += fonc(a+deltax*i) * deltax; voir Integrale1.cpp
```

on a introduit une fonction C++ pour décrire la fonction à intégrer

Même procédure avec la Formule du Rectangle calculée à droit :

on considère les points les N points $f(a+\Delta x)$, $f(a+2\Delta x)$, ... $f(a+N\Delta x)$

Pour accélérer l'exécution on peut multiplier par Δx à la fin. voir Integrale2.cpp

Formule du Trapèze

Une méthode très répandue est la **Formule du Trapèze**.

Elle consiste à prendre la moyenne des résultats (intégrales) obtenus avec la Formule du Rectangle calculée à gauche (I_G) et à droite (I_D) :

$$I = \frac{1}{2}(I_G + I_D) = \frac{1}{2} \sum_{i=0}^{n-1} [f(x_i) + f(x_i + \Delta x)] \cdot \Delta x$$

Tout ça est équivalente à calculer l'aire du trapèze formé par le 4 points

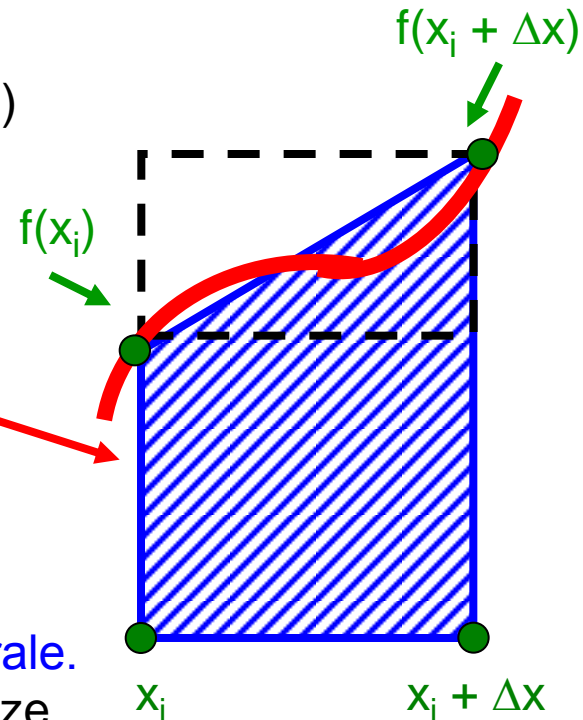
$$x_i, \quad x_i + \Delta x, \quad f(x_i), \quad f(x_i + \Delta x)$$

Cette formule donne une meilleure approximation de l'intégrale. Comme pour la Formule du Rectangle, la Formule du Trapèze fonctionne bien si la fonction ne change pas trop vite.

Pour ne pas faire deux fois le même calcul (optimisation) on peut réécrire la Formule :

$$I = f(a) \cdot \Delta x / 2 + \sum_{i=1}^{n-1} f(x_i) \cdot \Delta x + f(b) \cdot \Delta x / 2$$

Dans des méthodes adaptatives (plus avancée), la largeur Δx du sous-intervalle est déterminée par le taux de changement de la fonction : où la fonction change vite on utilise un Δx plus petit, où la fonction change lentement on utilise un Δx plus grand. P. ex. pour intégrer $\sin x / x$ il faut faire très attention pour $x \rightarrow 0$.



Exemples

soit $f(x)$ une fonction arbitraire à intégrer entre a et b ;

l'intervalle $[a, b]$ est divisé en N sous intervalles de largeur $h = (b-a) / N$

Formule du Rectangle à gauche

```
double h = fabs(b-a)/div;

double integrale = 0.;
for (int i=0; i<div; i++)
    integrale += f(a+i*h)*h;
```

Formule du Rectangle à droite

```
double h = fabs(b-a)/div;

double integrale = 0.;
for (int i=1; i<=div; i++)
    integrale += f(a+i*h);
integrale *= h;
```

voir [Integrale2.cpp](#)

Formule du Trapèze

```
double h = fabs(b-a)/div;

double integrale = 0.;
for (int i=0; i<div; i++)
    integrale +=
        f(a+i*h)+f(a+(i+1)*h);
integrale *= h/2.;
```

```
double h = fabs(b-a)/div;

double integrale = 1./2.*f(a);
for (int i=1; i<div; i++)
    integrale += f(a+i*h);
integrale += 1./2.*f(b);
integrale *= h;
```

voir [Integrale3.cpp](#)

Formule de Simpson

Pour améliorer ultérieurement la précision, on peut approximer la fonction $f(x)$ sur l'intervalle $[x - \Delta x/2, x + \Delta x/2]$ avec une parabole passant par $f(x - \Delta x/2)$, $f(x)$ et $f(x + \Delta x/2)$

→ **Formule de Simpson** (très populaire)

On prend l'expansion de Taylor de la fonction f autour de x_i

$$f(x_i + \Delta x) = f(x_i) + f'(x_i) \cdot \Delta x + \frac{1}{2!} f''(x_i) \cdot \Delta x^2 + \frac{1}{3!} f'''(x_i) \cdot \Delta x^3 + \dots$$

et on intègre la série entre $x_i - \Delta x/2$ et $x_i + \Delta x/2$:

$$\int_{x_i - \Delta x/2}^{x_i + \Delta x/2} f(x) dx = f(x_i) \int_{x_i - \Delta x/2}^{x_i + \Delta x/2} dx + f'(x_i) \int_{x_i - \Delta x/2}^{x_i + \Delta x/2} (x - x_i) dx + \frac{1}{2!} f''(x_i) \int_{x_i - \Delta x/2}^{x_i + \Delta x/2} (x - x_i)^2 dx + \frac{1}{3!} f'''(x_i) \int_{x_i - \Delta x/2}^{x_i + \Delta x/2} (x - x_i)^3 dx + \dots$$

Seuls les termes avec une dérivée paire sont différents de zéro :

$$\int_{x_i - \Delta x/2}^{x_i + \Delta x/2} f(x) dx = f(x_i) \Delta x + \frac{1}{3} f''(x_i) \frac{\Delta x^3}{8} + O(\Delta x^5)$$

Pour continuer nous avons besoin de $f''(x_i)$, i.e. la 2^{ème} dérivée de $f(x)$ calculée en x_i (voir dérivation numérique)

$$f''(x) = \frac{[f(x_i + \Delta x/2) - f(x_i)] / (\Delta x/2) - [f(x_i) - f(x_i - \Delta x/2)] / (\Delta x/2)}{(\Delta x/2)}$$

$$= \frac{f(x_i + \Delta x/2) - 2f(x_i) + f(x_i - \Delta x/2)}{(\Delta x/2)^2} + O(\Delta x^2)$$

et l'intégrale de $f(x)$ sur l'intervalle $[x_i - \Delta x/2, x_i + \Delta x/2]$ devient :

$$I = \frac{1}{3} [f(x_i - \Delta x/2) + 4f(x_i) + f(x_i + \Delta x/2)] \cdot \Delta x/2$$

à noter les coefficients de f :
le point au milieu pèse 4 fois plus
que le points aux extrémités

Et l'intégrale sur l'intervalle $[a,b]$ est donnée par $(\frac{1}{3} \cdot \Delta x/2 = \frac{1}{6} \cdot \Delta x)$

$$I = \frac{1}{6} \sum_{i=1}^N [f(x_i - \Delta x/2) + 4f(x_i) + f(x_i + \Delta x/2)] \cdot \Delta x$$

En résumé, nous avons approximé la fonction avec une **parabole** (3 points).

```
double a, b;
int div;
double h = fabs(b-a)/div;

double integrale = 0.;
for (int i=0; i<div; i++)
    integrale +=
        f(a+i*h) + 4.*f(a+h/2.+i*h) + f(a+h+i*h);
integrale *= h/6.;
```

voir [Integrale4.cpp](#)

Intégrale multidimensionnelle

Les méthodes pour l'intégration numérique en deux ou plusieurs dimensions sont moins développées que en une dimension.

On généralise le méthode du rectangle (force brute !).

Imaginez une distribution des charges électriques ponctuelles q_i , situées en r_i .

Le champ électrique \mathbf{E} dans un point quelconque \mathbf{P} est donné par (principe de superposition) :

$$\vec{E}(\vec{P}) = \frac{1}{4\pi\epsilon_0} \sum_{i=1}^{i=n} q_i \frac{(\vec{P} - \vec{r}_i)}{|\vec{P} - \vec{r}_i|^3}$$

Et maintenant imaginez une distribution de charge arbitraire $\rho(\mathbf{r})$.
Le champ électrique \mathbf{E} dans le point \mathbf{P} est donné par l'intégrale :

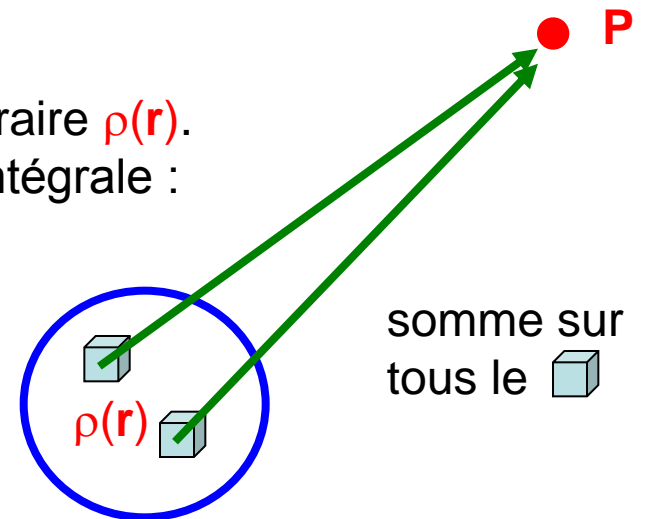
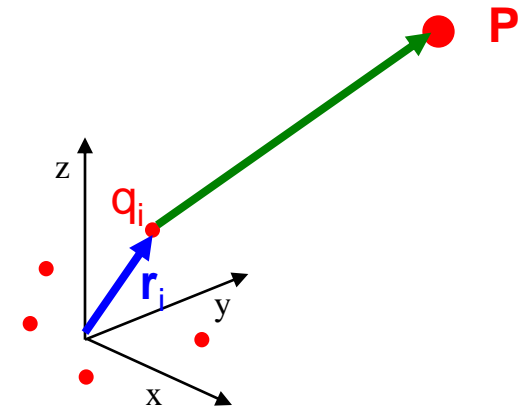
$$\vec{E}(\vec{P}) = \frac{1}{4\pi\epsilon_0} \int_V \rho(\vec{r}) \frac{(\vec{P} - \vec{r})}{|\vec{P} - \vec{r}|^3} d^3\vec{r}$$

Pour intégrer cette expression,

on divise le volume V en petit cubes (sous volumes)

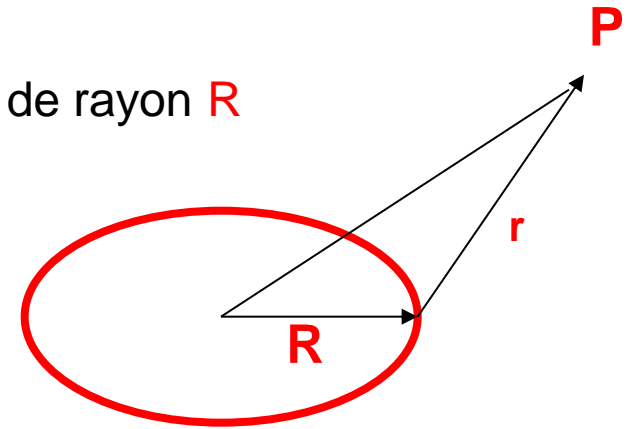
→ discrétisation du problème

et on additionne les contributions de chaque cube au champ total (voir exercices 7 et 8).



Exemple : champ électrique

Calculez le champ électrique généré par un anneau de rayon R et de charge Q dans un point quelconque P .
(On peut résoudre le problème analytiquement, si le point X est situé sur l'axe de symétrie z .)



D'abord il faut analyser le problème !
(chaque problème est différent !)

Pour calculer le champ électrique en P on peut procéder de la manière suivante :

1) on introduit un système des coordonnées cartésiennes avec l'origine dans le centre de l'anneau

2) on divise l'anneau en n (petits) segments de longueur

$$l = R \times \Delta\phi \quad (\Delta\phi = 2\pi / n)$$

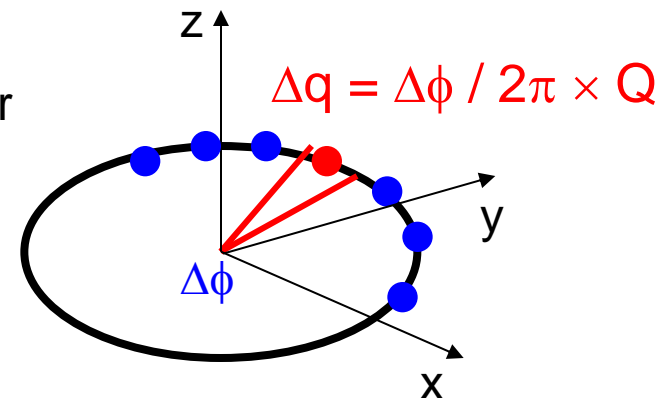
→ discrétisation du problème

3) la charge Δq de chaque segment est donnée par

$$\Delta q = Q / n = Q \times \Delta\phi / 2\pi$$

et on approxime le segment avec un point
(comme si toute la charge est concentrée dans ce point)

discrétisation



4) le champ électrique $|\mathbf{E}|$ en \mathbf{P} généré par la charge Δq est donné par

$$1/4\pi\epsilon_0 \Delta q / r^2$$

et il est dirigé le long de la ligne joignant le segment au point \mathbf{X} (r est la distance) :

$$\mathbf{r} = (P_x - R \cos \phi , P_y - R \sin \phi , P_z)$$

5) on calcule le champ électrique (E_x, E_y, E_z) généré par ce segment (point) dans \mathbf{P} .
(nous avons «paramétré» \mathbf{E} en fonction de ϕ .)

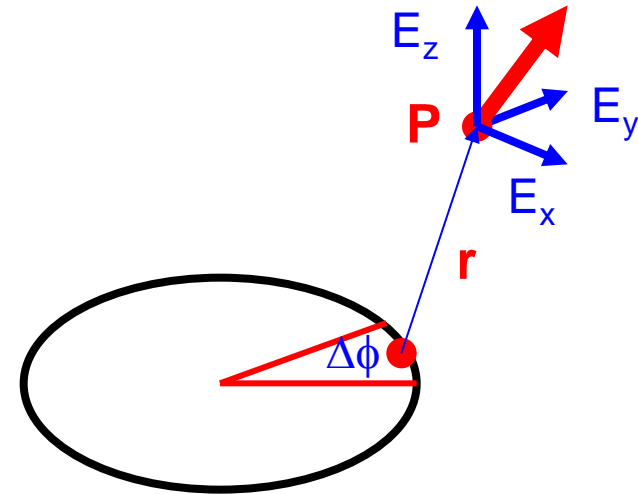
6) on additionne la contribution au champ électrique de chaque segment au point \mathbf{P} avec une boucle sur ϕ :

$$\begin{aligned} E_x &= \sum_{i=0}^{i=n-1} E_x (i \times \Delta\phi + \Delta\phi / 2) \\ E_y &= \dots & E_z &= \dots \end{aligned}$$

en résumé :

on a paramétré l'anneau avec ϕ et on a parcouru l'anneau avec un boucle sur ϕ

Pour tester le programme, calculez E sur l'axe z et vérifiez la réponse du programme !



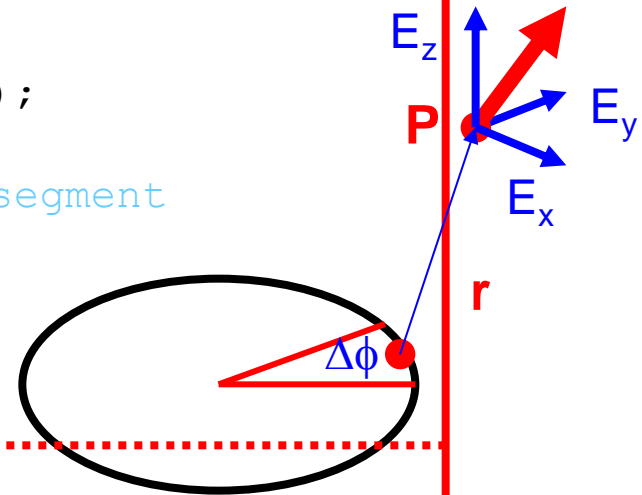

```

for (int iseg=0; iseg<div; iseg++) {
    phi = deltaPhi/2. + iseg*deltaPhi;
    xseg = R*cos(phi); yseg = R*sin(phi); zseg = 0.;

    //calcul des composantes du vecteur E
    fieldComp(xseg,yseg,zseg,xp,yp,zp,q,ex,ey,ez);

    //et on addition le champ genere par chaque segment
    fieldX += ex;
    fieldY += ey;
    fieldZ += ez;
}

```



```

void fieldComp(double x, double y, double z,
               double xp, double yp, double zp,
               double q, double &ex, double &ey, double &ez) {

    //distance entre le centre du segment et le point P
    double dist = distanceToPoint(x,y,z,xp,yp,zp);

    //champ electrique |E| en X
    double e = 1./(4.*M_PI*eps0) * q / (dist*dist);
    //composantes de E
    ex = e * (xp-x) / dist;
    ey = e * (yp-y) / dist;
    ez = e * (zp-z) / dist;
}

```

voir ChampElect.cpp

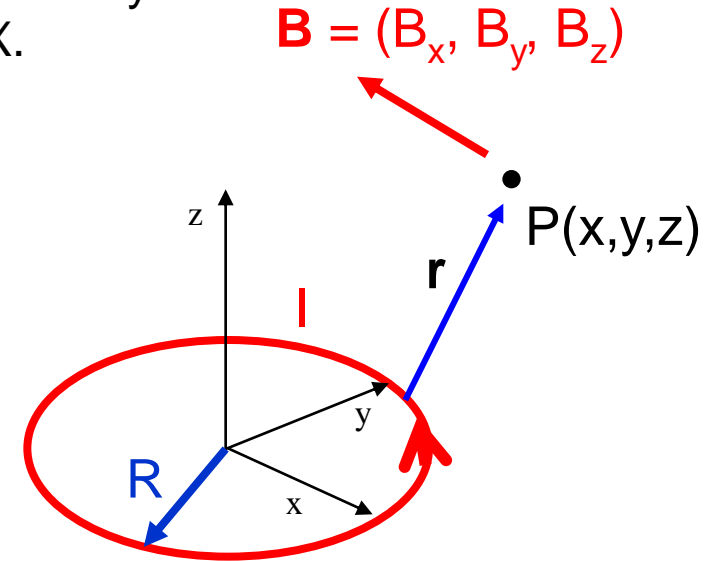
Exemple : champ magnétique

Calculez le champ magnétique produit par un anneau de rayon R parcouru par un courant I dans un point quelconque X .
(On peut résoudre le problème analytiquement, si le point X est situé sur l'axe de symétrie z .)

D'abord il faut analyser le problème !
(similaire au précédent)

Pour calculer le champ magnétique \mathbf{B} en X on peut procéder de la manière suivante :

- 1) on introduit un système des coordonnées cartésiennes avec l'origine dans le centre de l'anneau
- 2) on divise l'anneau en n (petits) segments de longueur $l = R \times \Delta\phi$ ($\Delta\phi = 2\pi / n$)
→ discrétisation du problème
- 3) le champ magnétique $|d\mathbf{B}|$ en X généré par le segment est donné par la lois de Biot-Savart.
La difficulté se trouve dans le calcul du produit vectorielle $d\mathbf{l} \times \mathbf{r}$ (algèbre) et non pas dans la physique du problème.
- 4) on additionne la contribution au champ magnétique de chaque segment au point X avec une boucle sur ϕ .



$$d\mathbf{B} = \frac{\mu_0}{4\pi} I \frac{d\mathbf{l} \times \mathbf{r}}{r^3}$$
$$\mathbf{B} = \frac{\mu_0}{4\pi} I \oint_l \frac{d\mathbf{l} \times \mathbf{r}}{r^3}$$
$$\Rightarrow \approx \frac{\mu_0}{4\pi} I \sum_i \frac{d\mathbf{l}_i \times \mathbf{r}_i}{r_i^3}$$

* Différentiation numérique

Comme pour l'intégration, dans certains cas, il est difficile de calculer la dérivée d'une fonction analytiquement ou bien la fonction à différentier n'est connue que par ses valeurs sur un ensemble fini de points (p.ex., des mesures expérimentales). Dans tous ces cas, il faut considérer des méthodes numériques afin d'approcher la quantité recherchée. L'intérêt principal de la différentiation numérique sera l'étude et la solution des équations différentielles (p.ex. calcul de la trajectoire).

définition de la dérivée

$$\frac{df(x)}{dx} = \lim_{h \rightarrow 0} \frac{f(x+h) - f(x)}{h} \cong$$

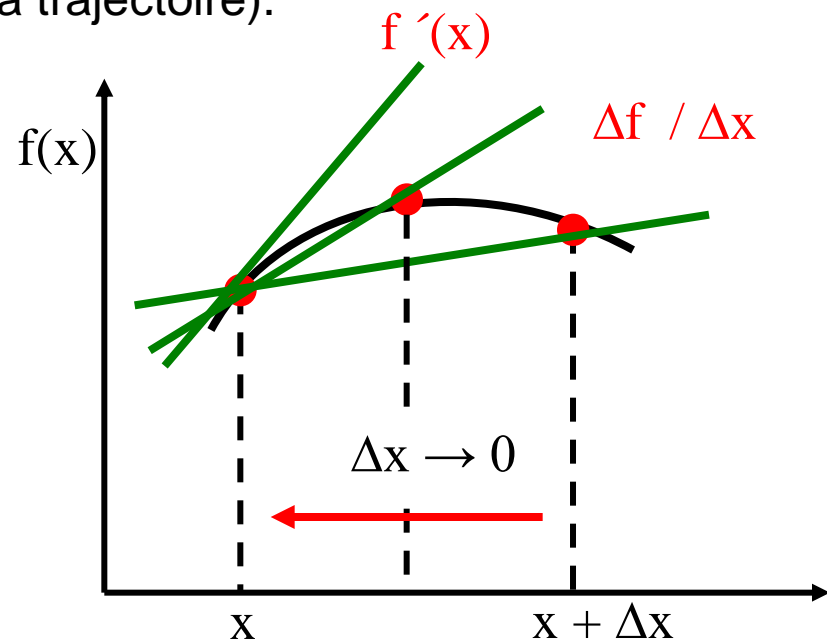
différence finie $\frac{f(x + \Delta x) - f(x)}{\Delta x} = \frac{\Delta f}{\Delta x}$

vitesse

$$v(t) = \frac{dx}{dt} = \lim_{\Delta t \rightarrow 0} \frac{x(t + \Delta t) - x(t)}{\Delta t}$$

vitesse moyenne

$$\Delta v = \frac{x(t + \Delta t) - x(t)}{\Delta t}$$



Q1: comment faire la limite avec l'ordinateur ?

Q2: comment choisir Δx (ou h) ?

étant donné la précision finie de l'ordinateur !

* Problèmes avec la différentiation

Comment choisir h ?

Les nombres à virgule flottante (approximation des nombres réels) ont une précision ϵ finie : $\sim 10^{-7}$ pour les `float` et $\sim 10^{-16}$ pour les `double`.

si $\Delta f = f(x+h) - f(x) < \epsilon \rightarrow f'(x) = 0$

si $h < \epsilon \rightarrow h = 0$ et division par zéro ! \rightarrow inf

si $|f(x+h) - f(x)| < \epsilon$ et $h < \epsilon \rightarrow$ division 0 sur 0 \rightarrow NaN (Not a Number)

Et on se retrouve dans l'impossibilité de faire la limite au sens mathématique.

Théorème de Taylor

$$f'(x) = \frac{f(x + \Delta x) - f(x)}{\Delta x} - \frac{1}{2} f''(\zeta) \Delta x$$

avec $x < \zeta < x + \Delta x$;

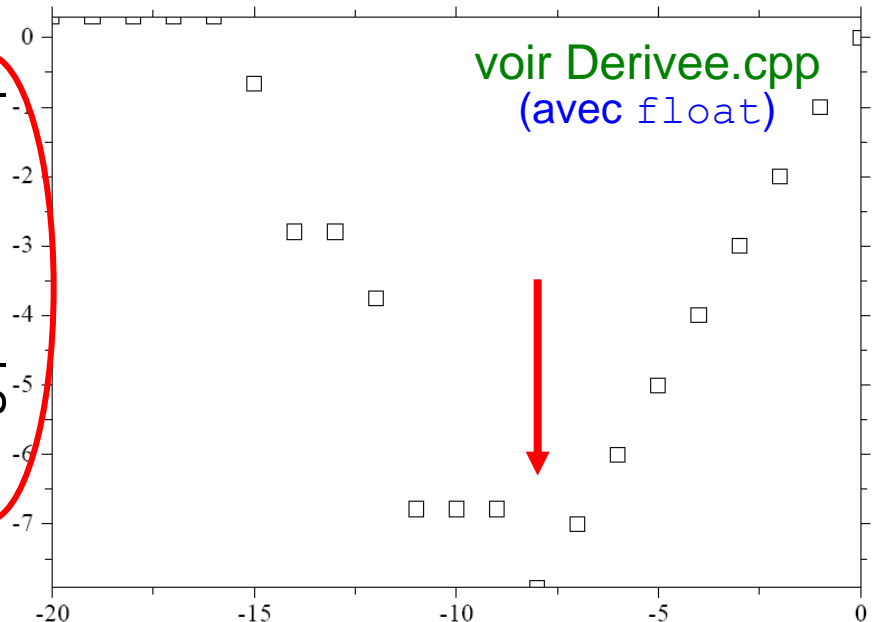
si nous ignorons le 2^{ème} terme
nous faisons un erreur de troncation

p.ex.: $f(x) = x^2$ dans $x = 1$

$f'(x) = 2x$

$\Delta f / \Delta x = [f(x + \Delta x) - f(x)] / \Delta x (= 2x + \Delta x)$

$\log |\Delta f / \Delta x - f'|$



* Différence finie centrale

Pour améliorer la précision, on considère deux intervalles symétriques à droite et à gauche du point x (i.e. $x - h$ et $x + h$) où l'on veut approximer la dérivée :

Développement en série de Taylor

$$f(x+h) = f(x) + hf'(x) + \frac{1}{2}h^2 f''(x) + \dots + \dots$$

$$f(x-h) = f(x) - hf'(x) + \frac{1}{2}h^2 f''(x) - \dots + \dots$$

différence finie à droite

différence finie à gauche

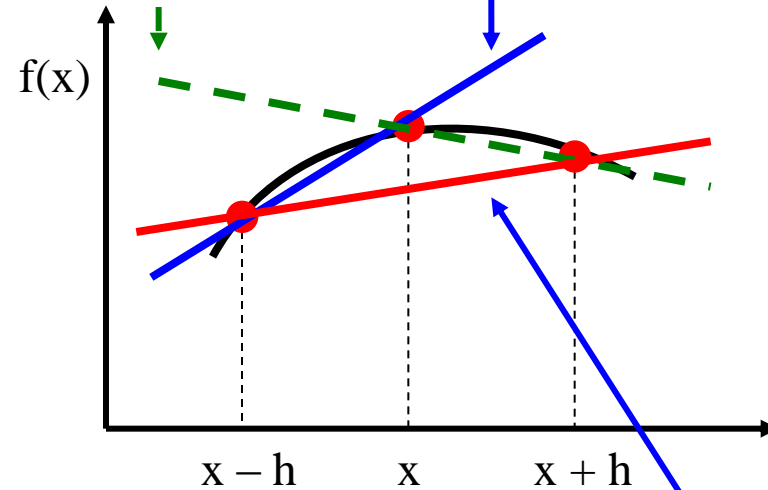
si l'on prend la différence (f calculée en $x-h$ et $x+h$)

$$f(x+h) - f(x-h) = 2hf'(x) + 2\frac{h^3}{3!} f'''(x) + \dots$$

l'on obtient une meilleure approximation de f'
différence finie centrale

$$f'(x) = \frac{f(x+h) - f(x-h)}{2h} + O(h^2)$$

pente de la corde



Si nous considérons un ensemble p.ex. de 5 points ($x - 2h, x - h, x, x + h, x + 2h$) (formule d'ordre supérieur), nous obtiendrons des approximations encore plus précises.
A essayer !

si l'on prend la somme de la
la différence finie à droite et
la différence finie à gauche
l'on obtient la 2^{ème} dérivée f''
(nous avons utilisé cette expression
pour dériver la Formule de Simpson)

$$f''(x) = \frac{f(x+h) - 2f(x) + f(x-h)}{h^2} + O(h^2)$$

implémentation

```
int main() {  
    const int STEPMAX = 20;  
  
    float h, dfNum, dfCalc;  
    float x = 1.;  
    float delta;  
    for (int step=0; step<STEPMAX; step++) {  
        h = pow(10., -step);  
        dfNum = pow((x+h), 2) - pow(x, 2) / h;  
        dfCalc = 2.*x;  
        delta = log10(fabs(dfCalc - dfNum));  
  
        cout << setprecision(24) << fixed << h << " " << delta << endl;  
    }  
  
    return 0;  
}
```

$$f(x) = x^2$$

$$f'(x) = 2x$$

$$\Delta f(x) / \Delta x = [f(x+\Delta x) - f(x)] / \Delta x$$

calculée pour x = 1

dérivée numérique

dérivée analytique

différence

Exercices – série 5

Exercices et problèmes d'intégration

1. Développez les programmes de la page 11.

Ecrivez un programme pour calculer l'intégrale I d'une fonction arbitraire. Utilisez une fonction C++ pour définir $f(x)$. Essayez avec une fonction simple, p.ex. $a_0 + a_1 x + a_2 x^2$.

2. Intégrez $\sin(x)$ et $\cos(x)$ entre 0 et π

avec différentes partitions pour l'intervalle $0 - \pi$ ($N = 4, 8, 16, 256, \dots$)

et comparez les résultats avec la formule du trapèze et la formule de Simpson.

3. Dérivez la formule du trapèze et la formule de Simpson en détail.

Quelle est la précision de chaque méthode ?

4. Calculez le volume et la surface du solide engendré par la rotation de la courbe

$f(x) = \sin(x) + x^2 + 2$ autour de l'axe x entre 0 et 1 ($0 < x < 1$) par intégration numérique avec la formule du rectangle et la formule du trapèze.

5. La vitesse d'une particule est donnée par :
$$v = \frac{t^3}{\sqrt{1+3t^2}} \text{ m/s}$$

Quelle est la distance d parcourue par la particule après 5 secondes ?

La vitesse initiale est 0. Calculez d avec une erreur de 1%.

6. Une particule se déplace avec une vitesse initiale v_0 . La force de frottement de l'air est $F = -Cv$. Écrire l'équation du mouvement. Après combien du temps la vitesse v sera $(v_0/\sqrt{2})$? (problème unidimensionnel)

7. La Terre n'est pas une sphère parfaite. P. ex. on doit en tenir compte pour mettre les satellites en orbite terrestre. Calculez le champ gravitationnel généré par un ellipsoïde (rayons a , $b=a$, $c=2a$) en fonction de la distance et l'angle azimutal (quelques points) et comparez-le avec celui d'une sphère (voir page 14).

8. Moment d'inertie:

Supposez que la densité d'une sphère varie comme $\rho(r) = \rho_0 \exp(-r / r_0)$, où r est la distance du centre de la sphère. Le rayon de la sphère mesure R et soit $r_0 = R$. Calculez le moment d'inertie I si l'axe de rotation passe par le centre de la sphère (voir page 14).

$$I = \int_V a^2 \cdot dm = \int_V a^2 \cdot \rho \cdot dV \quad a \text{ est la distance à l'axe de rotation}$$

Puis calculez I si l'axe de rotation passe à $2r_0$ du centre de la sphère.

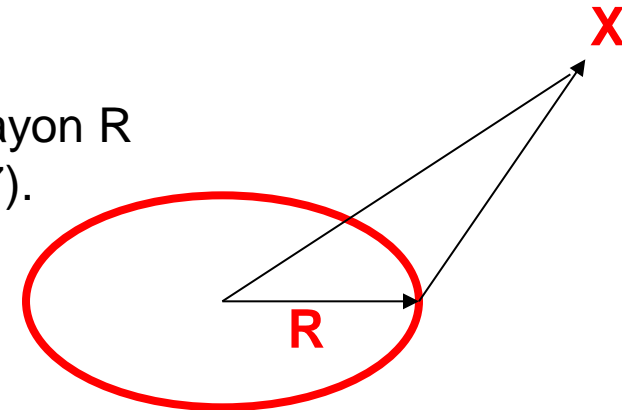
Indications:

- 1) diviser la sphère en petits cubes de volume ΔV
- 2) la masse de chaque cube est donnée par $dm = \rho(r) \cdot \Delta V$
- 3) additionner chaque cube à I

9. Champ électrique:

Calculez le champ électrique \mathbf{E} produit par un anneau de rayon R et charge Q pour un point quelconque X (voir pages 15 –17).

Décrivez les vecteurs position et champ électrique avec des tableaux.



10. Champ magnétique:

Cette fois, l'anneau ci-dessus est parcouru par un courant I . Calculez \mathbf{B} (voir page 18).²⁵

Exercices et problèmes de Différentiation

1. Ecrivez un programme qui calcule

- $\sin(x)$ et $\cos(x)$
- la différence finie à droite et à gauche
- la différence finie centrale

et comparez les résultats avec la solution analytique.

Variez Δh .

2. Ecrivez un programme qui calcule la première et la seconde dérivée d'une fonction arbitraire.

3. La vie moyenne $t_{1/2}$ du cobalt ^{60}Co est 5,271 ans. Ecrivez un programme qui calcule l'activité de la source en fonction du temps et de la masse de la source.

4. Estimez l'erreur d'arrondi lors du calcul d'une dérivée avec la différence finie central. Ecrivez un programme qui calcule l'erreur absolue $\Delta(h) = |f'(x) - f'(x)_{\text{approx}}|$.

Utilisez la fonction $f(x) = x^2$ pour $x = 1$ (voir `derivee.cpp`).

Essayez aussi avec $f(x) = x^3$ pour $x = 1$.