Dérivées numériques d'ordre 1 – Erreur d'arrondis

Introduction à l'analyse numérique

Prof. Marco Picasso

Search MOOC

Video

Dérivée d'ordre 1 ; erreur d'arrondis

$$u(x) = 0 + \varepsilon \sin k\pi x \qquad \dot{u}(t) = f(t) \qquad u(t) = u(0) + \int_0^t f(s)$$

$$u'(x) = \varepsilon k\pi \cos k\pi x$$

$$\int u(x) = -\frac{\varepsilon}{k\pi} \cos k\pi x$$

Voilà venu le temps de parler d'erreur d'arrondis. Donc, en règle générale, dériver un signal bruité augmente le bruit et intégrer un signal bruité diminue le bruit. Donc, pourquoi ? Prenons, par exemple, une fonction *u* de *x*. Imaginons que la vraie fonction soit zéro et pour une raison x ou y, il y ait un bruit petit, *ε sinus k π x*, où *k* est grand. Donc si vous calculez-- Donc ici l'amplitude du bruit, c'est *ε* sur la fonction. Si vous calculez maintenant la dérivée *u'(x)*, eh bien, c'est *ε* fois *k π*, fois cosinus *k π x*. Donc le bruit sur la dérivée, c'est *ε* fois *k π*. Donc le bruit est devenu grand parce que *k*, a priori, est grand. Par contre, si vous intégrez, la primitive de *u* de *x* c'est *ε* sur *k π*, fois le cosinus *k π x*, avec un signe moins. Et donc cette fois-ci le bruit, *ε*, a été divisé par *k π*, *k* étant grand, le bruit a été diminué. Donc en général, quand on résout une équation différentielle, donc *u* point de *t* égal *f(t)*, en fait on intègre, on ne dérive pas, on intègre. *u(t)* c'est *u(0)* plus l'intégrale de zéro à *t* de *f(s) ds*.

Notes

Summary

0m 04s

Dérivée d'ordre 1 ; erreur d'arrondis

$u(x) = 0 + \varepsilon \sin k\pi x$ $\qquad \dot{u}(t) = f(t) \qquad u(t) = u(0) + \int_0^t f(s)\,ds$

$u'(x) = \varepsilon k\pi \cos k\pi x$

$\int u(x) = -\frac{\varepsilon}{k\pi} \cos k\pi x$

double $c = 1.0/3.0$; $\qquad c = 1/3 \qquad \bar{c} = 0,\underbrace{333 \ldots 333}_{16 \text{ chiffres}} \qquad |c - \bar{c}| = \frac{1}{3} 10^{-16} = |c|\, 10^{-N}$ N chiffres sign.

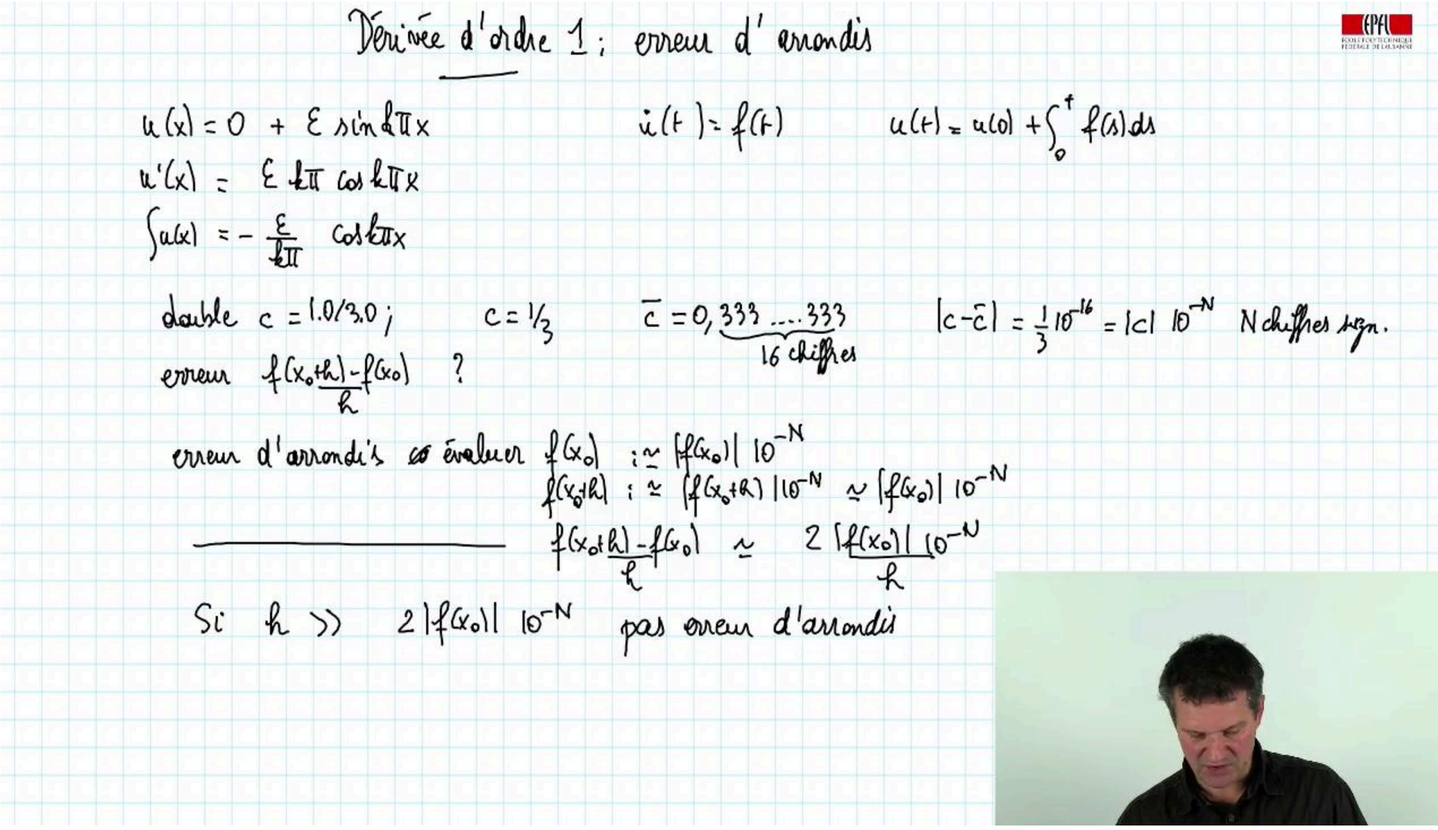
erreur

Par chance, lorsqu'on résout une équation différentielle ou une équation dérivée partielle, on intègre quelque chose, le second membre, et donc on diminue le bruit. Donc on n'a pas ce problème-là mais il faut savoir qu'en règle générale, lorsqu'on dérive quelque chose, un signal numérique, on augmente le bruit. Alors, comment pouvons-nous formaliser un peu ces choses-là ? Donc *double c* égal un tiers, 1.0 divisé par 3.0, dans un programme écrit en *C*, par exemple. Donc c égal un tiers est le chiffre exact et puis son approximation sur un ordinateur à N chiffres significatifs sera c barre égal-- par exemple si je prends 16 chiffres significatifs, ce qui est le cas lorsque j'utilise des *double*, eh bien, j'obtiens 0,333, et ainsi de suite, *...33*. Il y a ici 16 3 sur un ordinateur à 16 chiffres significatifs. Donc l'erreur *c* moins *c* barre, valeur absolue, ce sera, approximativement, un tiers 10^ -16, c'est-à-dire *c*, la valeur exacte, un tiers, fois *10^ -N*, étant donné que j'ai un ordinateur à N chiffres significatifs. Voilà, maintenant quelle est l'erreur que je fais, quelle est l'erreur d'arrondis lorsque j'utilise une formule de différence finie ?

Notes

Summary

1m 34s



Par exemple, *f(x0 + h)- f(x0)*, divisé par h, pour approcher la dérivée *f'(x0)*. Donc l'erreur d'arrondis pour évaluer *f(x0)*, c'est approximativement *f(x0)*, en valeur absolue, fois *10^ -N*, N est le nombre de chiffres significatifs, 16 en pratique. Donc si je veux évaluer maintenant *f(x0 + h)*, on va mettre à peu près égal à *f(x0 + h) 10^ -N*, mais comme *h* est censé être petit, c'est à peu près *f(x0) 10^ -N*. Donc l'erreur d'arrondis pour évaluer notre formule de différence finie, *f'(x0) - f(x0)*, sur *h*, c'est à peu près *2 f(x0) 10^ -N*, sur *h*. Alors on observe cet effet des erreurs d'arrondis au moment où *h* devient de l'ordre du numérateur, ici, de *f(x0) 10^ -N*. Premièrement, si h est beaucoup plus grand que *2 f(x0) 10^ -N*, alors on n'observe pas l'effet des erreurs d'arrondis, ce qui est le cas en général puisque je vous rappelle que *N* vaut 16 donc *2 f(x0) 10^ -16*, en général, est très petit. Par contre, si *h* devient de l'ordre de cette quantité-là, on va observer l'effet des erreurs d'arrondis.

Notes

Summary

3m 12s