Matière : ANALYSE NUMERIQUE1

2ème Année Licence Mathématiq

ue

Année universitaire 2021/2022

Les Cours de la matière ‘Analyse numérique1’ sont destinés aux étudiants du 1er cycle (licence) et même au 2ème cycle (Master) universitaire en mathématiques, Physique, chimie , statistiques et les filières de technologie .

Ils expliquent les différentes approches numériques permettant de résoudre numériquement les problèmes du monde réels modélisés ou formalisés sous forme mathématiques notamment en fonctions ou ensemble d’équations.

Dans le premier chapitre1 nous commençons tout d’abord à présenter les concepts de base de l’analyse numérique en montrant les types d’erreurs qu’on peut rencontrer dans le calcul numérique. Dans le deuxième nous abordons en brefs quelques méthodes numériques itératives qui, chacune à ses hypothèses, permettrons d’approximer les racines de fonctions ou équations de la forme F(x)=0 sur un domaine D. Le choix de méthode dépend de la classe à laquelle appartient F(x) (F est transcendante, F est polynomiale, …), Le troisième chapitre est consacré à l’étude de l’approximation des fonctions connues en points en faisant de l’interpolation dans le sens le plus large de terme. Enfin dans le chapitre 4 on présentera la dérivation et l’intégration numérique des fonctions empiriques.

………

Professeur DJEFFAL Lakhdar

Chapitre1

**CONCEPTS DE BASE**

**DE L’ANALYSE NUMERIQUE**

1. **Introduction**

Dans ce chapitre on va voir le contraste entre les méthodes numériques et les méthodes exactes. Parfois, ce contraste entre les mathématiques appliquées et les mathématiques pures.

Plusieurs problèmes issus de plusieurs domaines scientifiques ou non (en Biologie, en Physique, en Chimie, Environnement, ...) formalisés sous forme de fonctions ou d’équations dont la résolution analytique (détermination de solution explicite) est difficile, compliquée, voir impossible à avoir. Face à ce genre de problématique qu’on fait recours à un autre outil mathématique puissant de résolution approchées des problèmes appelé ’Analyse Numérique’ et qui se sert en générale d’une machine digitale appelé ordinateur ou micro-ordinateur pour aide et la faciliter de résolution.

Qu’est ce que L’analyse Numérique?

L’ analyse numérique est la conception et l’étude des algorithme finis (méthodes numériques implémentées sur machine) du point de vue convergence et stabilité pour obtenir des solutions approchées certainement avec erreurs mais de degré très minime.

Généralement on blâme la machine (micro-ordinateur ou calculateur) à cause des résultats erronés, malgré que celle-ci peu parfois être la source des erreurs, souvent la nature du processus et le système de numération utilisé introduisent beaucoup d’erreurs sur la réponse voulue. Les erreurs apparaissent quand le nombre des données sont changés au système de numération de la machine et vice versa.

De plus les erreurs sont faites lorsqu’on travail dans un système de numération qui contient un ensemble d’éléments fini. Ceci est le cas de machine digitale.

De ce faite, lorsque l’on met une méthode numérique et son implémentation sur machine pour résoudre un problème, il est très recommandé de faire une étude critique des résultats obtenus et d’avoir une idée des erreurs que produit la méthode et la technologie utilisée. On parle alors de problématique de l’évaluation d’erreurs.

1. **Boite Noir**

L’informatique implique le développement et l’utilisation d’organes de traitement de l’information. L’information est présentée dans une certaine forme à la boite noire et celle-ci donne en retour une information sous une autre forme considérée comme plus utiles.

* 1. Données(Entrée)

Les données proviennent en générale d’un certain nombre de mesure et dépendent aussi du modèle mathématique choisi d’un certain processus (expérience) qu’on veut analyser. Donc, les données sont approximés et contiennent des erreurs.

* 1. Calcul

La machine (calculateur par exemple) peut aussi introduire des erreurs liées à elle même due à la capacité limitée des bits correspondant à la représentation d’une donnée et aussi des erreurs due aux méthodes numériques utilisées.

2.3 Résultats (sortie)

La sortie dépend des calculs et des entrées. Parfois on ne considère que la relation entre les données et la sortie. C'est-à-dire on ne s’intéresse pas à la machine.

Exemple

Supposons qu’on veut déterminer l’entrée sachant que la sortie soit égale à la valeur 5 par exemple sans aucune autre information sur le problème. Quelle sera la première raisonnable entrée à essayer.

Le premier choix possible : on commence par zéro ou 1. Si on a l’information suivante pour x0=5 et cette entrée nous donne y0=5.2 alors quelle sera le prochain test (choix).

La deuxième possibilité :

1. Fonction décroissante localement x1<x0
2. Fonction croissante localement, on choisi x1 / x1>x0 pour s’approcher de mieux

Supposons que la fonction est décroissante, on essaye de prendre x1/x1<x0 (=5), donc on prend par exemple x1=4 et on suppose que la machine nous donne 4.25 et quelle serait le prochain test ?

Supposons que la fonction est croissante, on essaye de prendre x2/x0<x1<x2 , alors on est certain que le résultat y2 est tel que y0<y1<y2 et ainsi de suite jusqu’à ce qu’on tombe sur le résultat donné yi=5.

Alors comme on le voit, ce procédé peut durée beaucoup, donc on doit agir en faisant des hypothèses supplémentaires sur l’équation ou la fonction de sortie. Pour cela on linéarise la fonction y=a x+b ? nous avons y0=a x0+b et y1=a x1+b et de ces deux équations linéaires on calcul la valeur d’entrée qu’il faut pour la sortie y=5. Ce qui donne x=91/19 = 4.7894736……

Remarque

Les 19 chiffres après la virgule se répète constamment (infiniment), il est possible que le calculateur accepte le chiffre sous forme de quotient (x=91/19) cependant, à son tour doit le changer sous la forme décimale avec un nombre limités de place en mémoire (de bits) et ignore les autres. Ceci entraîne la possibilité d’avoir des erreurs appelées ‘erreurs d’arrondis’.

1. **Vérification et Normalisation**

Pour bien expliquer la vérification et la normalisation pour savoir exactement jusqu’à quel ordre on est proche de la solution exacte d’un problème donné prenons l’exemple de la fonction quadratique suivant



Comparons  par rapport à la valeur *0,* on peut dire que la racine  est très proche à la racine exacte de la fonction et même si on arrondi la valeur *3.99999999* à la valeur 4 alors = . Sous cette forme, on voit que les vraies racines de la fonction sont

On peut remarquer que sous cette forme la racine approchée , la valeur diffère de la valeur *0* par contre la racine  est exacte. C'est-à-dire :

* La valeur de l’équation quadratique diffère de *0* par (pourapprochée)
* Mais la racine  diffère de la racine exacte par 
* Il est clair que l’erreur sur la racine est 10000 fois plus grande que l’erreur sur l’équation quadratique.

Prenons l’exemple, multiplions cette équation par , on aura

 substituons  dans l’équation et calculons l’expression, on aura  et donc  n’est plus une bonne solution.

Multiplions cette fois ci l’expression par la valeur, on aura

 substituons  dans l’équation et calculons l’expression, on aura  et donc la racine n’est plus vraiment une meilleure solution de . En remarque bien que dans ce dernier cas l’erreur sur la racine n’a pas changée par contre la valeur de l’équation quadratique à complètement changée.

En résumé, une substitution de racines possibles dans l’équation quadratique peut entraîner de graves erreurs telle que :

- de bonnes réponses vont nous sembler approximatives

- de fausses réponses vont nous sembler très bonnes solutions

- de proches solutions vont nous sembler mauvaises solutions

Pour éviter ce type d’erreurs, il faut essayer de normaliser l’équation ou de reconstruire (rétablir) l’équation quadratique à partir de ces racines et ayant le même coefficient de terme de.

**Méthode de normalisation**

On va redéfinir la variable indépendante et multiplier l’équation par une constante de telle sorte que :

* Le coefficient du premier et dernier terme de l’équation est égal à 1 en valeur absolue
* Le coefficient du terme intermédiaire est nul (ou 0).

En effet, soit l’équation quadratique, on pose , on substitue dans l’équation quadratique on aura, après calcule le terme intermédiaire ce qui donne Alors le cas *A=0* à exclure, donc on prend le second cas

Substituons la valeur de *B* dans l’équation quadratique  on aura 

Maintenant si  ceci d’une part et d’autre part nous voulons que le premier terme  (critères de normalisation).

Alors deux cas peuvent se présenter :

1ercas : Si () >0 

En remplaçant dans, on aura. On remplace *A* et *t* par leurs expressions calculées dans l’expression, on aura 

2ème cas Si () < 0 

En remplaçant dans, on aura. On remplace *A* et *t* par leurs expressions calculées dans l’expression, on aura.

La normalisation à pour but de simplifier les équations.

Tous les concepts et les résultats qualitatifs rencontrés dans cette section s’applique aussi à tous les polynômes de degré supérieur à deux.

1. **Erreurs**

Dans cette section, on va étudier les différents types d’erreurs et comment les estimer.

Soit *x* un nombre approché du nombre exacte *X*.

Si  valeur approchée par défaut.

Si  valeur approchée par excès.

* Erreur absolue: Appelée aussi erreur mesurée. Elle donne l’imprécision sur une mesure effectuée et elle est donnée par .
* Erreur relative : l’erreur relative  est le rapport de l’erreur absolue du nombre et du module du nombre exacte. Parfois il est préférable d’écrire. Cette erreur est appelée aussi le pourcentage d’erreur car elle s’exprime en %
* Source principal d’erreurs :

1. erreurs due à la position même du problème (la modélisation ou simulation d’un phénomène physique quelconque n’est en générale pas parfaite, ceci fait ainsi apparaître des erreurs qu’on appelle erreurs de méthodes)
2. erreurs associées en analyse mathématiques, aux processus infinis (les fonctions qui figurent dans les formules mathématiques sont souvent données sous la forme de suite infinies ou de séries [par développement limités . En mathématiques on est contraint de y mettre fin à, certain termes de la suite, en les considérant comme une approximation de la solution cherchée. Le processus ainsi arrêté donne lieu à une erreur dite erreurs de troncature].
3. erreurs associées aux systèmes de numération appelées erreurs d’arrondi
4. erreurs dues aux constantes physiques et paramètres numériques, dont les valeurs ne peuvent être déterminées qu’approximativement à partir d’un certain nombre d’expérience. Ces erreurs sont dites erreurs initiaux.
5. erreurs résultant d’une opération due aux erreurs des termes initiaux. Les erreurs de données (entrées) se propagent sur le résultat des calculs (sortie). Ces erreurs sont appelées erreurs propagées.
6. **Existence et Unicité de la solution**

L’existence et l’unicité de la solution repose généralement su le théorème des valeurs intermédiaires.

Théorème

Soit f une fonction non linéaire, si *f* est continue et prend le signe différent aux bornes de l’intervalle, alors il existe une solution  tel que 

1. **Multiplicité**

Soit *f* une fonction non linéaire définie de, continue et dérivable,  la racine de la fonction *f,* Si  alors, on dit que la racineest une racine de la fonction *f* de multiplicité*.*

Par convention, si la racine  est dite racine simple de la fonction**.**

1. **Conditionnement et stabilité**

Le conditionnement et la stabilité d’une fonction non linéaire *f* définie desont les concepts fondamentaux de l’analyse numérique. Ils mesurent la dépendance de la racine de la fonction *f* par rapport à ses données, ceci afin de contrôler la validité et la sensibilité d'un algorithme numérique pour le calcul de sa solution (appelé souvent Racines ou Zéros de fonction). Ces deux concepts sont relatives à la propagation des erreurs surtout ceux de type arrondis dans les calculs.

On dit qu’une fonction (équation) est **bien conditionnée** si elle possède un conditionnement bas. Elle est dite  **mal conditionnée** si elle possède un conditionnement élevé.

**Définition**

On appelle conditionnement d'une fonction numérique *f* de classe *C*1 en un point *x*, le nombre.

Et le conditionnement du calcul de racines de la fonction *f* est donné par. Pour remarque, les racines multiples sont toujours mal conditionnées.

**TD- N°1**

Exercice I.I (devoir à la maison)

Dans le cours, on vous a présenter comment analyser mathématiquement une équation de second degré. Refaire la même analyse pour une équation polynomiale de 3ème degré en répondant aux questions suivantes:

* Chercher les formules donnant les racines de polynôme de degré 3.
* Déterminer les cas où les erreurs signifiantes peuvent arrivées.
* Donner les formules pour les racines pouvant éviter ces erreurs.
* Donner un exemple et le résoudre en utilisant 4 chiffres.
* Normaliser le polynôme 
* Trouver les racines assez précise pour l’équation quadratique 
* Exercice I.2
* 1/ quel est l’objectif d’étude de l’analyse numérique? L’analyse numérique a t-il une relation directe avec l’informatique? Expliquer.
* 2/Expliquer en donnant un exemple les questions B, C,et D suivantes:
* B/ qu’est ce que Erreur d’arrondi?
* C/ qu’est ce que Erreur d’approximation?
* D/qu’est ce que Erreur de troncature?

Exercice I.3

Soit la fonction quadratique

* évaluer la valeur de tel que  une racine approchée de 
* quelle est l’erreur absolue et relative sur ?
* comparer l’erreur absolue de  à l’erreur sur la variable 

Exercice I.4

Soit 

Supposons que les constantes *a, b, c, d* et *e* sont obtenus d’une expérience aux 100 èm prés.

 avec  

a/ Déterminer 

b/Déterminer et  avec 

Exercice I.5

Considérer la fonction 

* calculer avec trois chiffres significatifs
* calculer l’erreur absolue, d’arrondi et totale

Chapitre 2

**RACINES D’UNE EQUATION (FONCTION)**

***F(X)=0***

1. **Introduction**

Étant donnée un problème mathématiques formulés sous forme d’une fonction *F* définie de l’ensemble IR dans IR ou dans C (Nombres complexes) de degrés élevé dont le domaine de définition est une partie D de IR. Le problème est chercher et déterminer les solutions x appartenant au domaine D qui vérifient *F(x)=0*. Ceci est connu en analyse numérique par les racines ou bien zéros de l’équation ou de la fonction *F*. Les fonctions qui s’expriment sous forme d’équation simples sont facile à résoudre et qui donne même des solutions exactes, néanmoins pour des degrés élevé d’après la théorie d’Évariste Gallois [2] , il n’existe pas de formules générales directes de résolution des fonctions (équations) ayant la forme polynomiale et encore moins si les fonctions sont sous forme exponentielle, logarithmique etc. D’où le recours aux approches numériques est nécessaire pour pouvoir les résoudre et donner des solutions approximatives meilleures.

**Exemple motivant [3].** On veut déterminer le volume *V* occupé par un gaz de température *T* et de pression *p*. L’équation d’état du gaz est formulée comme suit :

Où sont deux coefficients dépendants de la nature du gaz, *N* est le nombre de molécules contenues dans le volume *V* et *k* la constante de Boltzman. Il faut donc résoudre une équation non linéaire d’inconnu *V*. Ceci revient à trouvez les zéros de la fonction

L’existence et l’unicité de la solution exacte est plus compliquée à montrer dans ce cas, parce que l’équation est algébrique (non linéaire) et de degré élevé.

Autres exemples:

Nous pouvons aussi rencontrerquelques autres formes mathématiques de fonctions et équations non linéaires et dont la recherche de solution est vraiment très compliquée.



Face à ce genre de problématique sans solution (racines ou zéros))explicite ou compliquée et difficile à trouver, nous abordons en brefs quelques méthodes numériques itératives dans ce chapitre, qui, chacune à ses hypothèses, permettrons d’approximer les racines de fonctions ou équations de la forme F(x)=0 sur un domaine D. Le choix de méthode dépend de la classe à laquelle appartient F(x) et surtout la stabilité et le temps de réponse.

1. **Méthode de bissection (dichotomie, balayage)**

La méthode de bissection est une méthode numérique qui localise les solutions approximatives de la fonction *F(x)=0* en balayant son intervalle d’étude par la technique de subdivision en parties égales. Cette méthode se base sur le théorème des valeurs intermédiaires [4 (Voir cours de Analyse 1èr année Licence)] pour assurer la convergence malgré la lenteur qu’il peut y avoir et elle fonctionne avec des racines multiples.

**Rappel**

On considère une fonction F continue sur un intervalle contenu du domaine de définition D. Si alors il existe entre les points *a* et *b* au moins une racine (une solution) et si de plus *F(x)* est monotone la solution est unique dans.

**Principe de la méthode**

Soit *F(x)* une fonction continue sur , construire une suite finie de valeurs de *x* réparties surde la manière suivante :

1. 
2. ****
3. 

1. *On ré-itère le même processus à partir de l’étape 2 jusqu’à ce qu’on localise la solution*

*Remarque*

*Comme l’algorithme est itératif, le* ***test d’arrêt*** *du processus de calcul peut - être :*

* *Soit* 
* *Soit la détermination d’une valeur approchée prés de*  *et qui vérifie l’inégalité* 
* *Soit atteindre le nombre d’itération calculés N à l’avance est qui est donné par l’inégalité* 
* *Soit les résultats numériques calculés se stabilisent d’une itération à une autre*

**Théorème** Soit *F(x)* une fonction continue sur vérifiant. Si l’algorithme de la méthode dichotomie construit une suite de valeurs  et arrive jusqu’à l’itération n alors la convergence est assurée vers et on à l’estimation d’erreur.

Démonstration

La solution approchée par la méthode de dichotomie dite aussi de bissection est déterminée par une construction itérative de suite  dont la taille d’étude est  parce qu’à chaque itération on subdivise l’intervalle d’étude en 2. La limite de cette suite est finie (lim quand , ceci d’une part. D’autre part  avec  et comme la convergence est assurée d’où l’inégalité de l’estimation.

L’inconvénient de cette méthode est que la convergence est lente et encadre la solution.

Exercice I-1



Solution I-1

, l’intervalle ou se situe la racine de est, l’estimation de l’erreur ne doit pas dépasser. Appliquons l’algorithme étape par étape jusqu’à localisation de la meilleure racine de. Le principe est de subdiviser l’intervalle  en plusieurs sous intervalles égaux tout en appliquant à chaque subdivision le théorème des valeurs intermédiaires. Comme l’erreur d’estimation est donnée, on peut calculer la distance entre les solutions consécutives déterminées et vérifier si la racine est bonne ou non (i.e. utiliser le test d’arrêt)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| k | Intervalles d’études | Th. V. Intermediaires | Racines calculées |  |
| 0 |  |  |  |  |
| 1 |  |  |  | k>O, Non |
| 2 |  |  |  | Non |
| 3 |  |  |  | Non |
| 4 |  |  | 11/16 | Oui, c’est la racine cherchée,  Même, on remarque que la racine se répète, donc se stabilise, on arrête alors le processus de calcul |

1. **Méthode de Newton-Raphson**

La méthode de Newton est une autre méthode numérique dite méthode des tangentes qui en plus des hypothèses de dichotomie, elle recommande que la fonction *F(x)* soit de classe C2  et *F’(x)* gardeun signe constant non nul.

En se basant sur la formule de Taylor, cette méthode permet de construire une suite de valeurs qui converge plus rapide que la méthode de dichotomie et n’encadre pas la solution.

En effet, la formule de Taylor est donnée au voisinage *xi* par



Si *x* est proche de *xi* alors les termes deviennent très <<<<<0 et par conséquent on peut les négliger. Donc

. Et comme alors on aura 

L’estimation meilleure peut être, d’où la formule de Newton qui construit la suite de valeurs itérative est, avec valeur initiale choisis arbitrairement.

**Le test d’arrêt** : on arrête le processus de calcule de valeurs si la distance est vérifié ou bien les racines déterminées se stabilise d’une itération à une autre.

**Convergence de la méthode**

La formule de Newton ressemble bien à un problème de point fixe qui s’écrit sous la forme



Nous remarquons que la détermination de la solution séparée de est équivalente à la détermination du point fixe qu’on note par exemple par tel que.

Nous utilisons le théorème du point fixe local. ,

, si est le point fixe de

Calculons 

Cette fonction g est contractante, elle est continue, par conséquent la suite est de Cauchy donc convergente.

**Important**

Si nous voulons s’en passer de cette approche de point fixe et que la méthode de Newton Raphson est recommandée pour son utilisation alors on peut se contenter de la vérification des conditions appelés les conditions de Newton Raphson suivantes:

Soit *F(x)* une fonction de classe *C2* sur ***:***

1. *Si* *(Solution surement existe)*
2. *Si (la fonction est monotone), ceci implique que*

*, d’ou la convergence*

1. *, i.e elle garde un signe constant sur * alors le choix de la solution de départdoit satisfaire*.* Ce choix est facultatif (non obligatoire), il est utilisé seulement pour accélérer la convergence, sinon prendre une solution de départ arbitraire.
2. Une fois les conditions sont réunis et satisfaites, appliquons l’algorithme itératif donné de Newton-Raphson et générer ou construire la suite de solution
3. L’estimation d’erreur est donnée par tels que pour tout .

**Exemple**

Soit. Calculer la solution de cette fonction.

Calculer la solution de la fonction donnée revient à déterminer sa racine  telle que. Nous avons la fonction donc  dérivable dans,

L’existence de solution peut-être vérifié par le théorème des valeurs intermédiaires, nous avons 

Alors cherchons cette racine (ou solution).

Par application de la méthode Newton Raphson, nous allons construire une suite itérative de valeurs(en commençant par à choisir arbitrairement et selon la condition ) qui converge vers la solution par.

Les étapes de la mise en œuvre :

Soit l’indice *k* représente les itérations de construction de la suite.

Etape1- Pour, on choisit la solution initiale. Pour accélérer la convergence, on choisi directement et prendre le milieu de l’intervalle, donc . En plus la fonction dérivéeau point 

Etape2- calculer la solution suivante par

La solution Pour cela on vérifie le test d’arrêt suivant . On ne cannait pas l’estimation, regardons alors si la solution nouvel est stable par rapport à l’ancienne ou bien la distance entre cette nouvel solution et la solution exacte si elle existe est nul.

Si oui STOP arrêté le processus de calcul, la meilleurs solution ou racine approximative estde la fonction donnée.

Sinon passer à l’itération suivante et aller à l’étape2 et continué à chercher et localiser la solution.

Le tableau de valeurs de la suite construite est :

|  |  |  |
| --- | --- | --- |
| Itération *k* | Suite | Observation |
| 0 | 0.7853981635 |  |
| 1 | 0.73995361937 |  |
| 2 | 0.73990851781 |  |
| 3 | 0.73990851332 |  |
| 4 | 0.73990851332 | la racine se répète |
| 5 | 0.73990851332 | la racine se répète, elle est donc stable, on arrête |

**Remarques** On a présenté seulement les méthodes numériques de calcul de solutions approximatives d’une fonction (équation ) à coefficients réels et qui ont des solutions réelles. Supposons par exemple dans la méthode de Newton Raphson, le calcul de dérivée de la fonction en question donnée est difficile, compliquée ou voir même impossible.

Et/ou encore supposons que nous avons une équation sous forme de polynôme de degrés élevé et à coefficient réel et/ou complexe à résoudre mais qui n’a que des solutions Complexe  *(*exemple) . Comment fait-on pour déterminer la solution ?

Alors, pour la 1ère situation, il existe des méthodes numériques qui se basent sur l’approximation de la dérivée de la fonction par le taux d’accroissements. Elles recommandent deux pas ou deux itérés initiaux, c’est le cas de la méthode de la sécante par exemple.

Pour la 2ème situation, Les méthodes numériques qui permettent de calculer les racines sont : Si les coefficients sont complexes, il est conseillé d’appliquer la méthode de Laguerre. Si le polynôme est à coefficients réels, on utilise de préférence la méthode de Bairstow qui donne des solutions réelles et aussi des solutions complexes.

Pour plus de détail, nous renvoyons le lecteur aux: [www.bib.math.net](http://www.bib.math.net), [www.unige.ch/-haire/poly/chap2.pdf](http://www.unige.ch/-haire/poly/chap2.pdf), [www.lamfa.u-picardie.fr/chehab](http://www.lamfa.u-picardie.fr/chehab)

et quelques ouvrages des auteurs A.GOUDIN, BOUMAHRAT, FANK JEDZEJEWSKI, SIBONY, BARANGER et la plus part des ouvrages ayant comme titre: ‘Analyse numérique’

**TD - N°2**

Exercice2.1

Soit *F(x)* une fonction de classe *C*2 et monotone sur et qui vérifie le théorème des valeurs intermédiaires sur, partons d’une solution initiale arbitraire  et qui satisfait. Montrer que :

1. le processus de Newton - Raphson converge vers l’Unique solution 
2. l’estimation Avec M majorant de *F’’ (x)* et m minorant des *F’(x)* **Est** équivalente à 

Exercice 2.2

Soit *F(x)* une fonction continue et monotone sur,   Donner une interprétation géométrique du procédé de Newton –Raphson

Exercice 2.3

Démontrer que, pour appliquer la méthode de bissection en vue d’approximer la racine d’une fonction continue et monotone sur, en partant d’une solution initiale avec une précision donnée, le nombre d’itération *n* est fini et donné par:  et la convergence est assuré.

Exercice2.4

Soitune fonction continue sur

1. Montrer que la fonction *F(x)* admet une racine  séparée sur 
2. Étudier les conditions de Newton-Raphson pour bien choisir la solution initiale et calculer le Majorant et le minorant de la fonction
3. Calculer la solution approchée  de la fonction *F(x)*  à une précision 

Exercice 2.5 (TP)

Soit une fonction  continue et monotone sur l’intervalle 

Écrire un programme (en langage de programmation C ou bien matlab) qui permet de

1. Estimer le nombre d’itération pour atteindre la solution approchée à une précision 
2. Calculer et imprimer la solution approximative

Chapitre 3

**INTERPOLATION POLYNOMIALE**

**1. Introduction**

Dans la majorité du temps, les problèmes mathématiques issue de la médecine, de la physique, la finance, du marketing, etc. sont donnés sous forme de fonctions empiriques, c’est à dire connues en un certain nombre fini de points relevé des mesures expérimentales. Leurs résolutions est difficile voir impossible si on ne passe pas par une stratégie incontournable, ceci concerne les techniques de modélisation et d’approximation, précisément l’interpolation ou l’extrapolation, c'est à dire la problématique d’approximation des fonctions d’une manière générale.

1. **Contexte générale de l’interpolation polynomiale**

Alors de quoi s’agit-il ?

Considérons un cadre extrêmement simple et pour autant la problématique est délicate dés le départ. Soit le plan <XOY>, on suppose connue une fonction seulement en certains nœuds ou nombre de points réels distincts obtenue à la suite d’une expérience et qui prend respectivement des valeurs réelles en ces points.

Autrement on dit, on se donne  points de mesures d’observations, à chacun de ces points, correspond une grandeur de mesure. Supposons qu’on à besoin de bâtir toute une théorie et/ou d’effectuer des opérations mathématiques (dérivation, intégration, estimer des grandeurs de mesure en d’autres points d’observations situés entre ceux donnés au départ :  ) sur une telle fonction qui est inconnue explicitement.

Pour répondre à ce questionnement, il faut chercher un moyen de reconstruire et reformuler la fonction connue en points par une forme analytique, le plus souvent par un polynôme afin de pouvoir l’évaluer mathématiquement et avoir présent à l’esprit l’aspect d’erreurs d’approximation, ceci concerne l’interpolation (extrapolation) polynomiale.

Pourquoi Polynôme ?

* Premièrement, un polynôme est simple à évaluer (voir Schéma de HORNER)
* Deuxièmement, le théorème dit de WEIERSTRASS prouve et justifie l’utilisation d’approximation de fonction par des polynômes. (regardons la formule de Taylor dans le développement limité par exemple).

**3. Définition de l’Interpolation polynomiale**

L’interpolation est une opération mathématique permettant de remplacer une fonction analytique (explicite) ou une courbe de fonction connue en certain nombre de points réels  par une autre fonction (ou courbe) plus simple, mais qui coïncide avec la première en un nombre fini de points (ou de valeurs) donnés au départ.

Suivant le nombre de ces points, où la fonction (courbe) est connue, l’interpolation est dite linéaire si le nombre de points (donnés) est deux. Elle consiste à joindre les points  donnés par des segments de droite, autrement dit il suffit de choisir l’unique polynôme (fonction affine) de degré 1 dont la syntaxe est. On cherche les coefficients *a* et *b* tel que .

Si le nombre de points donnés est trois, et qui sont deux à deux distincts, il suffit de choisir l’unique polynôme de degré 2 (équation parabole), la solution est à peine délicate à trouver et qui vérifie  (sauf si les trois points donnés sont alignés, dans ce cas on se Ramène au cas linéaire).

Dans le cas où le nombre de points donnés est assez important, la reconstruction et la reformulation de la fonction connue en ces points donnés devient de plus en plus difficile à déterminer. D’où l’utilisation de la procédure d’approximation de fonction.

Il existe en gros trois types d’approximations:

1. Approximation par Interpolation polynomiale
2. Approximation par Interpolation par une fonction spline cubique
3. Approximation au sens des moindres carrées

Dans ce chapitre on s’intéresse au type d’approximation de fonctions par interpolation polynomiale qui est connue par ces méthodes de Lagrange et de Newton. Ce type est utilisé lorsque le nombre de points point est peu, deux à deux distincts et que la distance est faible et aussi la fonction f est régulière (f et ses dérivées sont continues sur l’intervalle .

**4. Interpolation polynomiale de Lagrange**

’’La première forme générale d’interpolation polynomiale est bien celle de Lagrange.’’

Lagrange dit la chose suivante : Est-ce que l’on peut construire une approximation pour remplacer une fonction  connue seulement en plusieurs points deux à deux distincts de telle sorte que =

Donc, chercher un polynôme d’interpolation , avec l’ensemble des polynômes de degré . Rappelons qu’un polynôme dans l’espace s’écrit, la donnée des constantes réellesdétermine entièrement le polynôme .

L’idée c’est de dire, il y a conditions ou bien points pour déterminer les  paramètresqui déterminent de manière unique le polynôme .

**4.1 Existence et Unicité**

Théorème

Soit valeurs ou points connues d’une fonction définie de, alors il existe un et un seul polynôme de degré *k* tels que.

Preuve

Le théorème consiste à dire qu’on va prendre un polynôme  et écrire les conditions 

Alors



Finalement on a un système d’équations linéaires non homogène à  variables inconnues .

Écrivons ce système linéaire sous forme matricielle :



On remarque que, la matrice associée a une forme exceptionnelle connue sous le nom de matrice VANDERMONDE. Nous avons une combinaison linéaire dans chaque équation d’inconnues.

Cette matrice par définition a un déterminant non nul, car tous les points connus au départ de la fonction *f* sont deux à deux distincts. D’où l’existence et l’unicité du polynôme d’interpolation qui remplace la fonction *f* connue enpoints et qui passe par tous ses points, c’est à dire qui interpole les valeurs données de *f.*

Remarque

Nous pouvons aussi démontrer l’existence et l’unicité du polynôme d’interpolation de Lagrange en passant par la description de système homogène.

**4.2 Formule explicite du polynôme d’interpolation**

Dans ce paragraphe, on va construire la formule que, Lagrange a mise en évidence.

Le polynôme d’interpolation de Lagrange peut être explicité en partant des propriétés suivantes:

1- soit  une fonction connues en , deux à deux distincts et qui prend des valeurs  ;

2- soitun polynôme qui vérifie  ;

* 3- 

La 2ème propriété veut dire que le polynôme d’interpolationqu’on va construire appartient à l’espace vectoriel des polynômes de degré au plus *k*. Il s’écrirait en termes de polynômes dans cet espace sous forme  où pour et les constituent une base de l’espace vectoriel des polynômes de degré  (c’est l’analyse vectoriel). Cette formule est une égalité fonctionnelle (on à une combinaison linéaire des polynômes), alors on pourrait se dire d’ailleurs les une base de.

Donc, de l’écriture vectorielle ou fonctionnelle de décomposition dans une base qu’on cherche à construire, ça veut dire numériquement.

Valorisons maintenant l’écriture en prenant pour chaque, on obtient alors et d’après la 3ème propriété. Il faudrait que la combinaison se dégénère lorsqu’on choisit correctement les polynômes  et les coefficientspuissent être associés aux valeurs connues  que suggère l’interpolation. Donc, faire dégénéreret de n’avoir qu’un seul coefficient  égale à, ça veut dire  (symbole de Kronecker).

Maintenant si  alors. Autrement dit,, et par conséquent, la formule de polynôme d’interpolation est , ceci d’une part. D’autre part, on va aller plus profondément sur le caractère explicite des polynômes de base de Lagrange.

Nous avons  et, alors pour, Ensuite ce polynôme de base de Lagrange a comme racines.

Calculons  : Deux cas se présentent, le cas ou les en tout points tel que  se traduit par : pour, l’expression qui permet de valider ce cas est  et ce dernier à pour racines. Le cas ou les, pour calculer et valider ce 2ème cas, il faut établir des corrections avec le numérateur déjà trouvé. Ceci nous amène à établir l’opération de division dont le numérateur est le résultat du 1er cas et le dénominateur est le produit des termes. D’où le polynôme explicite de base de Lagrange, et la formule générale de polynôme d’interpolation par la méthode de Lagrange est donné comme suit: 

**4.3 Algorithme basé sur la forme de Lagrange**

*Début*

Entrées :

Étant donné un nombre entier *k>0*

– le vecteur indicé  des *(k + 1)* points d’interpolation , pour i = 0, 1, ..., *k,*

– le vecteur indicé f des *(k + 1)* valeurs d’interpolation , pour i = 0, 1, ..., *k,*

– le point *X*

*P = 0* ;

Pour *i=0 à k* faire

{

*L = 1* ;

Pour *j = 0 à (i – 1)* faire { *L := L × (X −)/( −)*; fin pour }

Pour *j = (i + 1) à k* faire { L := L × (X −)/( −) ; fin pour }

*P := P + fi× L*

}

Sorties : Écrire *P (*Le résultat est dans la variable *P*).

*Stop et Fin*

Cet algorithme est fini et nécessite 2(2d + 1) (d + 1) ∼ 4d2 opérations

**4.4 Estimation de l’erreur de Lagrange**

Le polynôme d’interpolation de Lagrange  est explicité par les propriétés suivantes:

1- soit  une fonction connues en et distinct et qui prend des valeurs  ;

2- soitun polynôme () et qui interpole la fonction 

3- 

A partir du polynôme trouvé, il est très évident d’interpoler la fonction en d’autres points différent complètement des valeurs données au début. Graphiquement, il est naturel qu’on constate qu’il peut y avoir une petite distance entre la courbe initiale correspondant à la même fonction passant seulement par les points  et celle qui passe par les points  et les points d’où l’introduction et l’existence de l’erreur.

Pour mesurer cette distance, on estimer l’erreur qu’on note par  telle que .

Tout d’abord énonçons les résultats suivant

**4.4.1 Lemme**

Soit une fonction définie de  et dérivable alors si possède au moins (n+2) racines distincts la fonction dérivéepossède (n+1) racines distincts sur.

Preuve

Appliquer le théorème de Rolle entre deux racines consécutives de la fonction

**4.4.2 Corollaire**

Soit une fonction définie de  et, si possède au moins (n+2) racines distincts la fonction possède au moins une racine sur.

Preuve

Par récurrence, appliquons le résultat du lemme énoncé ci-dessus

**4.4.3Théorème**

Soit une fonction définie de  et) ; supposant que les donnéesvérifient la relation . Soit  le polynôme d’interpolation de Lagrange de la fonctionaux pointsalors   .

Preuve

N’oublions pas que l’erreur est due lorsqu’on a choisis des points différents de ceux donnés initialement.

La fonction  s’annule en points distincts. D’après le théorème de Rolle la fonction dérivée s’annule en au moins points distincts de. A nouveau la fonction dérivée s’annule en au moins points distincts deetc. et ainsi de suite jusqu’à ce qu’on obtient la dérivée nème de la fonctionqui s’annule en un point ou qui admet une racine qu’on appelle par exemple de. Donc il existe bien cette racine qui vérifie le théorème.

Le point  étant inconnu, on cherche une majoration et on a le corollaire suivant

**4.4 .4Corollaire**

Soit une fonction définie de  et, supposant que les donnéesvérifient la relation

Alors par majoration on à le résultat



Et en termes de norme, on a



L'idée fut donc de minimiser L = max x ∈ [ a , b ] | ( x − x 0 ) … ( x − x n ) | {\displaystyle L=\max \_{x\in [a,b]}\left|(x-x\_{0})\dots (x-x\_{n})\right|} le pour *n* points donnés. Ceci revient à dire qu’il faut avoir un bon choix des points initiaux ou la fonction est connue. On verra par la suite, comment se fait ce bon choix, voir paragraphe 6.

**5. Interpolation Newtonienne**

Une autre manière de calculer et d’obtenir rapidement une approximation polynomiale d’une fonction empirique (expérimentale) surtout, si les points *xi* sont répartir de façon équidistante, est d’utiliser le calcul des différences divisées. Par conséquent, la forme de Newton pour le polynôme d'interpolation est privilégiée par rapport à celle de Lagrange, d’où l'interpolation newtonienne.

**5.1 Différence divisée**

Les différences divisées sont des quantités qui correspondent à une discrétisation des dérivées successives d'une fonction. Ces quantités sont définies et calculées grâce à une procédure qui s’appelle lui-même, c'est-à-dire de manière récursive, en généralisant la formule du taux d’accroissement. Elles sont utilisées en particulier en interpolation newtonienne.

**5.2 Calcul des différences divisées**

Étant donnés*n + 1 {\displaystyle n+1}*, *(k+1)* abscisses distincts et les valeurs . Les quantités des différences divisées sont calculées de la manière suivante

Notation : D’abord f [ x 0 , … , x n ] {\displaystyle f[x\_{0},\dots ,x\_{n}]} les différences divisées sont notés par[ y 0 , … , y n ] {\displaystyle [y\_{0},\dots ,y\_{n}]} le symbole crochetet par convention, on écrit et nous avons 



Et donc la quantité des différences divisées peut être donnée par 

Pour expliciter le processus récursif de calcul de ces différences divisées, on utilise le schéma sous forme de tableau. Pour  couples de points donnés, on crée un tableau de lignes et colonnes. Les deux premières colonnes contiennent les couples et les autres contiennent les résultats instantanés de calcul des différences divisées et qui sont insérées à partir de la 2ème ligne du tableau.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  | Etape1 | Etape2 | Etape3 |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
| … |  |  |  |  | … |

**5.3 Polynôme d’interpolation de Newton**

Étant donnés*n + 1 {\displaystyle n+1}*, *(n+1)* abscisses distincts et les valeurs. Nous avons vu, qu’il existe un polynôme de degré, qui interpole cette fonction et qui vérifie l’égalité  alors, on peut dire aussi 



Et donc 

Or, (forme générale d’un polynôme)

, le coefficientreprésente ici les différences divisées d’ordre  de la fonction et s’écrit :

Il existe deux types de formes pour déterminer le polynôme de Newton :

Forme progressive : le polynôme d’interpolation est donné par



Forme régressive : le polynôme d’interpolation est donné par



Notons bien, que les polynômes d’interpolation de Newton des deux formes sont égaux et identique.

Exercice. Soit une fonction connue en points*n + 1 {\displaystyle n+1}*, trois points deux à deux distincts et qui prend ses valeurs. Déterminer le polynôme de Newton qui interpole la fonction donnée.

Solution. Choisissons par exemple la forme progressive :

On dresse le schéma sous forme de tableau comme indiqué auparavant et on commence à le remplir par les quantités des différences divisée, puis on détermine le polynôme en question.

Rappel

Le polynôme d’interpolation de Newton (forme progressive) est donné par



Nous avons les donnéeset les valeurs.

=0, On remarque, qu’il faut tout d’abord calculer les quantités des différences divisées pour. Donc on dresse un tableau de 3 lignes et 4 colonnes (parce que on à trois points)

|  |  |  |  |
| --- | --- | --- | --- |
|  |  | Etape1 | Etape2 |
| =1 | =0 |  |  |
| =2 | =3 |  |  |
| =3 | =2 |  |  |

Une fois le calcul des différences divisées est terminé, on passe à la détermination du polynôme de Newton 

Nous avons 3 points, donc on aura un polynôme de degré  qui passe par ces points. Donc et 

Et donc 

Donc le polynôme d’interpolation de Newton qui interpole la fonction donnée en points 

# 6 . Polynôme de Tchebychev

# Il est très important d’approximer au mieux une fonction connue en points et d’améliorer la convergence de l’interpolation particulièrement de Lagrange.

# L’idée de base est de partir avec un bon choix des points d’interpolation afin de minimiser le grand écart entre la fonction à interpoler et son approximation polynomiale appelé souvent ’’Phénomène de Runge’’. De nombreux types de polynômes orthogonaux comme ceux de Legendre, Tchebychev permettent de répondre à ce problème. Les polynômes de Tchebychev constitue un outil important en Analyse numérique pour l’interpolation polynomiale de fonction et capable de répondre aux critères de bon choix des points ou la fonction est connue.

# 6.1 Définition

Un polynôme de Tchebychev est un terme d’une suite de polynômes orthogonaux particuliers reliés à la formule de MOIVRE. Il existe deux suites de polynômes de Tchebychev, l'une nommée polynômes de Tchebychev de première espèce et notée  et l'autre nommée polynômes de Tchebychev de seconde espèce et notée *Un* de degré n.

Ces deux suites sont définies par récurrence : ,

Les deux premiers termes pour les suites *et Un* sont initialement donnés par



# Lesexpressions *et Un* sont des suites de polynômes orthogonaux par rapport à un produit scalaire de fonctions, associé à la fonction dite fonction poids.

# Comme ces polynômes sont reliés à la formule trigonométrique de Moivre, alors, Tchebychev considère et défini une autre alternative des polynômes

# Sur

C'est-à-dire**:** soit****. On appelle polynôme de Tchebychev de degré *n,* l’application définit comme suit : ****

# Rappel

Le produit scalaire de fonctions le plus simple est l'intégrale du produit de ces fonctions, sur un intervalle borné

Considérons deux fonction *f* et *g* sur  alors : 

⟨ f , g ⟩ = ∫ a b f ( x ) g ( x )   d x {\displaystyle \langle f,g\rangle =\int \_{a}^{b}f(x)g(x)~\mathrm {d} x} Plus généralement, on peut introduire une « fonction poids » *W*(*x*) dans l'intégrale (sur l'intervalle d’intégration*] a, b [,* *W* doit être à valeurs finies et strictement positives, et l'intégrale du produit de la fonction poids par un polynôme doit être finie ; les bornes *a*, *b* peuvent être infinies) : 

⟨ f , g ⟩ = ∫ a b f ( x ) g ( x ) W ( x )   d x {\displaystyle \langle f,g\rangle =\int \_{a}^{b}f(x)g(x)W(x)~\mathrm {d} x} Avec cette définition du produit scalaire, deux fonctions sont orthogonales entre elles si leur produit scalaire est égal à zéro (de la même manière que deux vecteurs sont orthogonaux (perpendiculaires) si leur produit scalaire égale zéro). L'intervalle d'intégration est appelé intervalle d'orthogonalité.

**2. Racines de polynôme de Tchebychev**

**Proposition.** Soit****. On appelle polynôme de Tchebychev de degré *n,* l’application définit comme suit****. Admet exactement *n* racines simples définies par : ****

Le choix de détermination des points d’interpolation ****est fait de telle manière que 

Autrement dit,  pour tout polynôme *Q* normalisé de degré inférieur à (n+1).

Preuve

1/ On justifie le choix de racine

Nous avons par définition ****

 Substituant par son expression ****

On aura 

2/ On démontre cette expression

* **TD- N°3**

**Exercice3.1**

Soit C:\Users\pc\AppData\Local\Temp\ksohtml\wps8BF9.tmp.pngune fonction connue en *n + 1 {\displaystyle n+1}*C:\Users\pc\AppData\Local\Temp\ksohtml\wps8C0A.tmp.png pointsC:\Users\pc\AppData\Local\Temp\ksohtml\wps8C0B.tmp.png, c’est à dire la fonction C:\Users\pc\AppData\Local\Temp\ksohtml\wps8C0C.tmp.png prend ses valeursC:\Users\pc\AppData\Local\Temp\ksohtml\wps8C1C.tmp.png. Montrer qu’il existe un unique polynôme C:\Users\pc\AppData\Local\Temp\ksohtml\wps8C1D.tmp.pngde degré C:\Users\pc\AppData\Local\Temp\ksohtml\wps8C1E.tmp.png interpolant la fonction C:\Users\pc\AppData\Local\Temp\ksohtml\wps8C1F.tmp.pngsi et seulement si les points C:\Users\pc\AppData\Local\Temp\ksohtml\wps8C20.tmp.png donnés sont deux a deux distincts.

**Exercice3.2**

Soit la fonction C:\Users\pc\AppData\Local\Temp\ksohtml\wps8C21.tmp.pngdonnée sous la forme suivante

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *I* | C:\Users\pc\AppData\Local\Temp\ksohtml\wps8C22.tmp.png | 1 | 2 | 3 |
| C:\Users\pc\AppData\Local\Temp\ksohtml\wps8C23.tmp.png | -1 | 2 | 4 | 5 |
| C:\Users\pc\AppData\Local\Temp\ksohtml\wps8C34.tmp.png | -2 | 43 | 213 | 376 |

* Déterminer le polynôme d’interpolation de Lagrange de la fonction C:\Users\pc\AppData\Local\Temp\ksohtml\wps8C36.tmp.pngpassant par les points donnés C:\Users\pc\AppData\Local\Temp\ksohtml\wps8C37.tmp.png
* Déterminer le polynôme d’interpolation de Newton de la fonction C:\Users\pc\AppData\Local\Temp\ksohtml\wps8C48.tmp.pngpassant par les points donnés C:\Users\pc\AppData\Local\Temp\ksohtml\wps8C49.tmp.png
* Quelle est le degré du polynôme d’interpolation déterminé ? Justifier votre réponse.
* Donner la valeur approximative d’extrapolation de la fonction *f* au point *x=5.4*

**Exercice3.3**

Soit C:\Users\pc\AppData\Local\Temp\ksohtml\wps8C4A.tmp.pngune fonction connue en *n + 1 {\displaystyle n+1}*C:\Users\pc\AppData\Local\Temp\ksohtml\wps8C4B.tmp.png valeursC:\Users\pc\AppData\Local\Temp\ksohtml\wps8C4C.tmp.pngaux points C:\Users\pc\AppData\Local\Temp\ksohtml\wps8C5D.tmp.pngdistincts. Le polynôme de Lagrange de degré C:\Users\pc\AppData\Local\Temp\ksohtml\wps8C5E.tmp.pnginterpolant cette fonction C:\Users\pc\AppData\Local\Temp\ksohtml\wps8C5F.tmp.pngest donné par l’expressionC:\Users\pc\AppData\Local\Temp\ksohtml\wps8C60.tmp.png,.

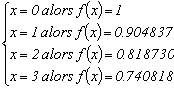
Maintenant, si les points donnés C:\Users\pc\AppData\Local\Temp\ksohtml\wps8C61.tmp.png sont équidistants et répartis dans un intervalle fini avec un pas fixe est égal àC:\Users\pc\AppData\Local\Temp\ksohtml\wps8C62.tmp.png, c'est-à-direC:\Users\pc\AppData\Local\Temp\ksohtml\wps8C63.tmp.png.

* Montrer que le polynôme de Lagrange de degré C:\Users\pc\AppData\Local\Temp\ksohtml\wps8C64.tmp.pnginterpolant cette fonction C:\Users\pc\AppData\Local\Temp\ksohtml\wps8C75.tmp.pngest donné par l’expressionC:\Users\pc\AppData\Local\Temp\ksohtml\wps8C76.tmp.png, avec C:\Users\pc\AppData\Local\Temp\ksohtml\wps8C77.tmp.png est tel que C:\Users\pc\AppData\Local\Temp\ksohtml\wps8C78.tmp.png
* soit la fonctionC:\Users\pc\AppData\Local\Temp\ksohtml\wps8C79.tmp.png donnée comme suit

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *I* | C:\Users\pc\AppData\Local\Temp\ksohtml\wps8C89.tmp.png | 1 | 2 | 3 |
| C:\Users\pc\AppData\Local\Temp\ksohtml\wps8C8A.tmp.png | 1 | 1.001 | 1.002 | 1.003 |
| C:\Users\pc\AppData\Local\Temp\ksohtml\wps8C8B.tmp.png | 1 | 1.00033 | 1.00066 | 1.00099 |

Calculer C:\Users\pc\AppData\Local\Temp\ksohtml\wps8C8C.tmp.png

**Exercice3.4**

On considère la fonction C:\Users\pc\AppData\Local\Temp\ksohtml\wps8C8D.tmp.png qui, pour .

* Calculer la valeur de la fonction C:\Users\pc\AppData\Local\Temp\ksohtml\wps8C9F.tmp.png
* Calculer l’erreur d’interpolation de Lagrange

**Exercice3.5**

D’après le cours, le polynôme de Tchebychev de degré *n* est défini sur l’intervalleC:\Users\pc\AppData\Local\Temp\ksohtml\wps8CA0.tmp.pngpar la relation mathématiqueC:\Users\pc\AppData\Local\Temp\ksohtml\wps8CA1.tmp.png.

* En posantC:\Users\pc\AppData\Local\Temp\ksohtml\wps8CA2.tmp.png, montrer que les polynômes C:\Users\pc\AppData\Local\Temp\ksohtml\wps8CB2.tmp.png peuvent être définie par la relation de récurrence suivante : C:\Users\pc\AppData\Local\Temp\ksohtml\wps8CB3.tmp.png
* Montrer que deux polynôme de Tchebychev distinctsC:\Users\pc\AppData\Local\Temp\ksohtml\wps8CB4.tmp.png et C:\Users\pc\AppData\Local\Temp\ksohtml\wps8CB5.tmp.png constitue un système orthogonale surC:\Users\pc\AppData\Local\Temp\ksohtml\wps8CC6.tmp.png relativement avec la fonction poids C:\Users\pc\AppData\Local\Temp\ksohtml\wps8CC7.tmp.png
* Soit la fonction C:\Users\pc\AppData\Local\Temp\ksohtml\wps8CC8.tmp.pngdéfinie sur l’intervalleC:\Users\pc\AppData\Local\Temp\ksohtml\wps8CC9.tmp.png par le tableau de valeur :

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| *i* | C:\Users\pc\AppData\Local\Temp\ksohtml\wps8CCA.tmp.png | 1 | 2 | 3 | 4 |
| C:\Users\pc\AppData\Local\Temp\ksohtml\wps8CCB.tmp.png | 1.5 | 1.6 | 1.7 | 1.8 | 1.9 |
| C:\Users\pc\AppData\Local\Temp\ksohtml\wps8CCC.tmp.png | 0.9975 | 0.9996 | 0.9917 | 0.9738 | 0.9463 |

* Calculer approximativement la valeur de C:\Users\pc\AppData\Local\Temp\ksohtml\wps8CCD.tmp.pngau point C:\Users\pc\AppData\Local\Temp\ksohtml\wps8CCE.tmp.png
* A partir du développement limité de la fonction C:\Users\pc\AppData\Local\Temp\ksohtml\wps8CDF.tmp.png au voisinage de zéro à l’ordre 5, déduire le développement limité de la fonctionC:\Users\pc\AppData\Local\Temp\ksohtml\wps8CE0.tmp.png en fonction des polynômes de Tchebychev de degré 1, degré 3 et degré 5

**Exercice TP3**

Écrire un programme basé sur la forme de Lagrange qui calcule le polynôme d’interpolation P d’une fonction *f* connue aux points et qui prend ses valeurs. Pour un point donné z calculer la valeur de *f(z)*

Donc

En Entrées, on a :

– le vecteur indicé  des *(k + 1)* points d’interpolation  *pour i = 0, 1, ..., k,*

– le vecteur indicé f des *(k + 1)* valeurs d’interpolation  *pour i = 0, 1, ..., k,*

– le point z donné, pour estimer P(z)

En Sorties: Le résultat est dans la variable P.