ANALYSE NUMÉRIQUE

T.P. N° 1

Alexandre Peltier
Louis Pendu

N° Groupe : B6

Observations :

Notes :	ANALYSE :		**Total :**
	RÉSULTATS :		
	PROGRAMMATION :		
	RAPPORT :		

EISTI, 25 mars 2011

Rapport du TP1

25 mars 2011

Table des matières

1 Introduction

Nous sommes ici confrontés à un problème typiquement informatique. En effet il s'agit d'évaluer les erreurs de représentation d'une machine selon le nombre de bits de mantisse de celle-ci. Afin de visualiser ce phénomène nous avons travaillé sur un polynôme de degré trois en le calculant sur un intervalle spécifique et de deux façons différentes, par la méthode classique et la méthode de Horner et en simulant des machines de différentes taille de mantisse.L'intervalle en question est [1-δ,1+δ] (le δ pouvant prendre la valeur 0,1 ou 0,5 selon le choix de l'utilisateur au début du programme)
Voici un exemple de polynôme sous une forme classique et sous la forme de Horner :

$$P(x) = a_0 + a_1 * x + ... + a_n * x^n \tag{1}$$

$$H(x) = a_0 + x * (a_1 + x(...x(a_{n-2} + x(a_{n-1} + x * a_n))...)) \tag{2}$$

Les nombres à virgule flottante sont les nombres les plus souvent utilisés dans un ordinateur pour représenter des valeurs non entières. Ce sont des approximations de nombres réels.Les nombres à virgule flottante possèdent un signe s (dans -1, 1), une mantisse M et un exposant E.Ils s'écrivent sous la forme $x = (-1)^s * M^E$. La mantisse est donc le nombre de chiffres significatifs d'un nombre.
Tous les graphes utilisés et les valeurs calculées dans ce rapport ont été réalisés avec une mantisse de 16 bits et sur l'intervalle [0.9,1.1] supposé contenir 500 points.
A l'aide de ce rapport nous allons montrer que la manière d'effectuer des calculs influe fortement sur les erreurs que peut commettre une machine.

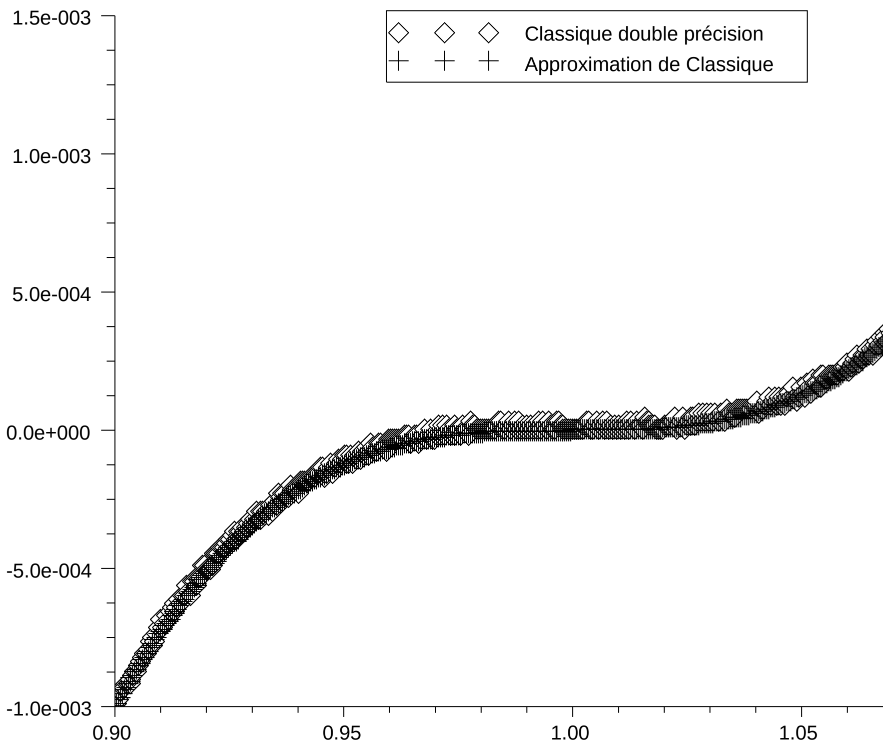
2 Analyse

2.1 Première partie : Calcul classique

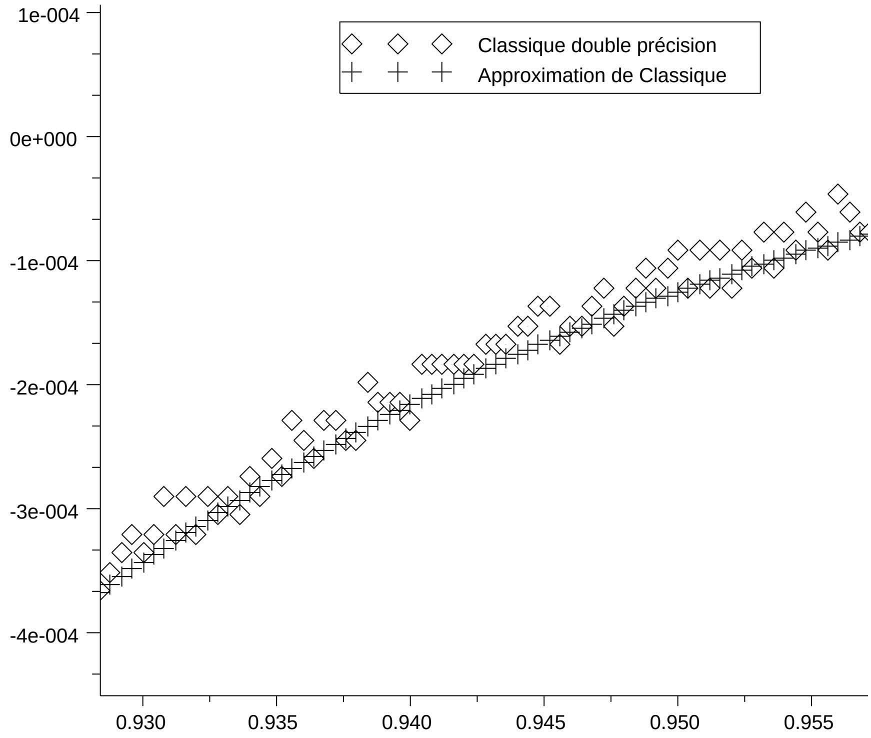
Comme premier travail nous avons codé en scilab le polynôme suivant :

$$P(x) = -1 + 3 * x - 3 * x^2 + x^3 \tag{1}$$

Nous avons donc calculé la valeur de ce polynôme avec la double précision de la machine utilisée afin d'obtenir des valeurs de référence lors de la comparaison avec les valeurs calculées avec une précision moindre.

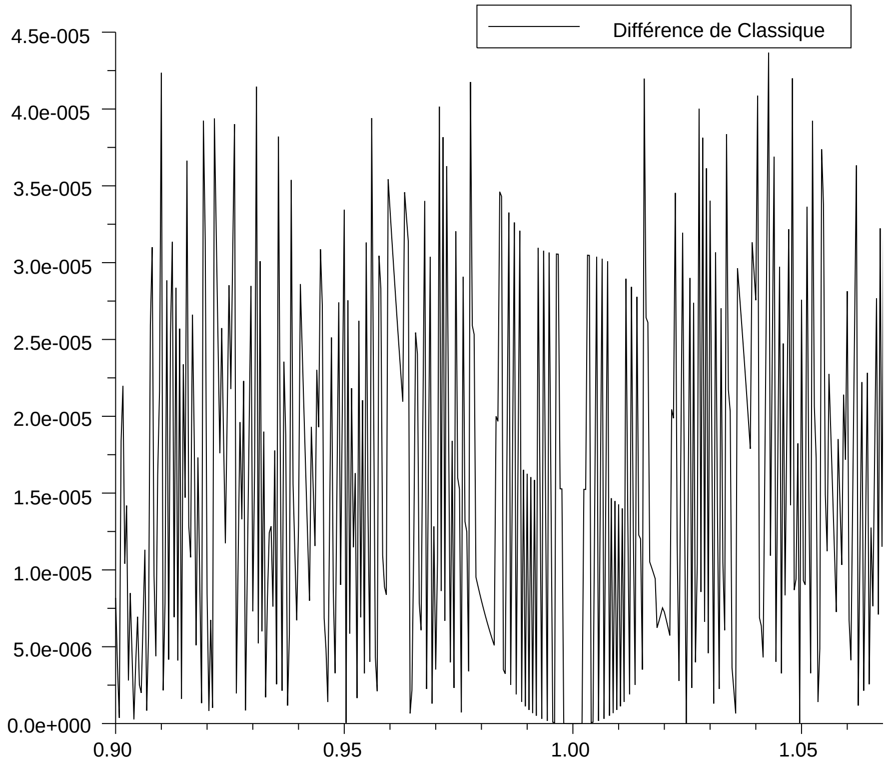


Le graphe obtenu ci-dessus est difficilement interprétable et lisible car nos 2 courbes sont sensiblement égales. C'est pourquoi nous joignons ci-dessous un zoom de ce graphe afin de pouvoir expliquer le résultat obtenu.



On voit bien ici après zoom que nos deux courbes sont très proches mais qu'il y a une erreur entre les deux.

Maintenant on s'intéresse à cette erreur. Le graphe ci-dessous représente donc la différence de chaque point entre le calcul classique et le calcul approché



Sur ce graphe on se rend compte que l'erreur absolue est comprise entre 0 et $4.5 * 10^{-5}$.

Avec Scilab nous avons aussi calculé l'erreur moyenne absolue qui correspond à une moyenne de toutes les valeurs affichées sur le graphe précédent. On a obtenu une erreur absolue :

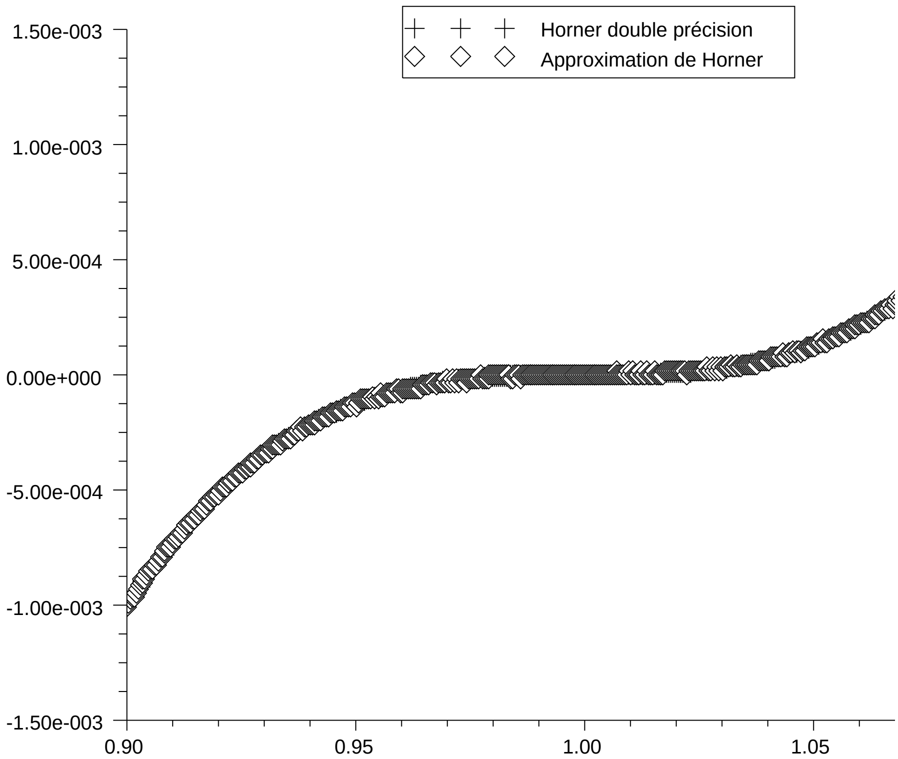
$$E(classique) = 1.5822 * 10^{-5} \tag{2}$$

Malgré tout, afin de minimiser cette erreur on essaye de faire une meilleure approximation en utilisant la méthode de Horner.

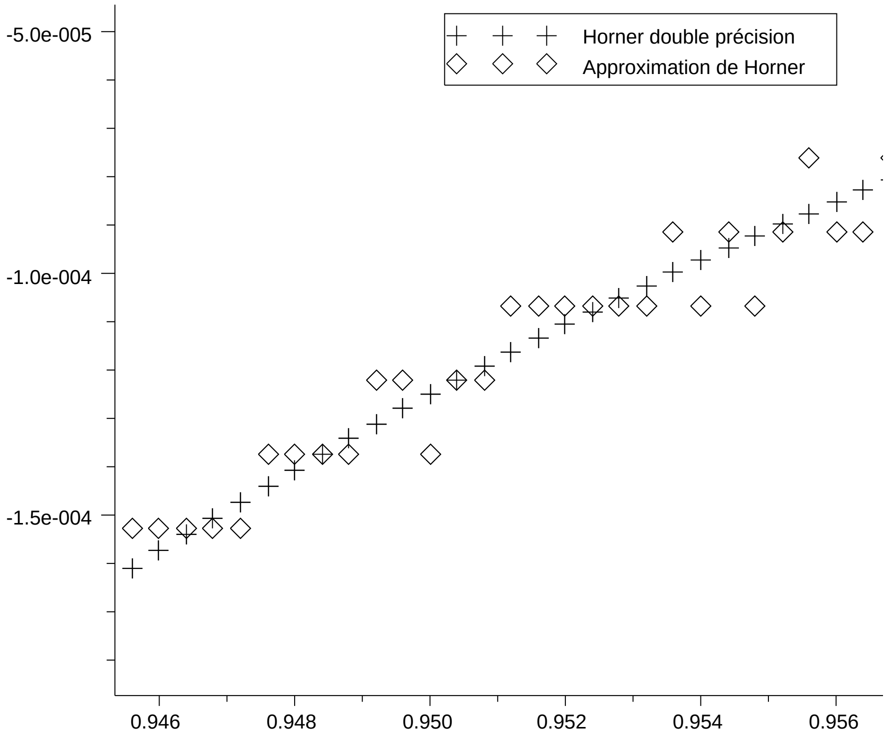
2.2 Deuxième partie : Calcul de Horner

Afin de mener à bien notre objectif qui est de démontrer l'influence de l'algorithme de calcul sur la précision du résultat, nous avons aussi codé ce même polynôme mais sous la forme de Horner :

$$P(x) = -1 + x(3 + x(-3 + x)) \tag{3}$$

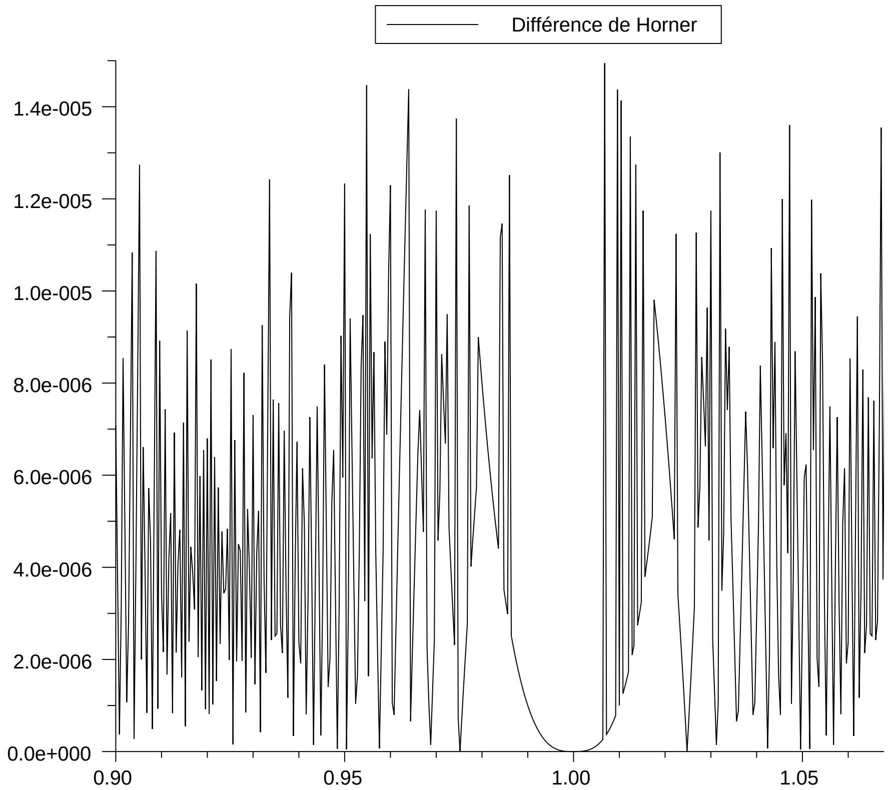


Le graphe obtenu ci-dessus est assez similaire au premier graphe obtenu au début du TP. Il est tout aussi difficile de le lire c'est pourquoi par un souci de clarté et pour une étude plus fine nous avons préféré zoomer sur une partie de ce graphe.



Lors du zoom on voit que nos courbes sont assez similaires mais il existe une erreur que nous allons étudier, déterminer puis comparer par la suite avec les résultats par la méthode classique.

A présent on s'intéresse donc à cette erreur. Le graphe ci-dessous représente donc la différence de chaque point entre le calcul de Horner et celui induit par son approximation.



Sur ce graphe on se rend compte que l'erreur absolue est comprise entre 0 et $1.5 * 10^{-5}$. Mais ce il ne permet pas de la comparer avec celle obtenue par la méthode classique car les valeurs sont bien trop dispersées.
Avec Scilab nous avons tout comme pour l'autre méthode calculé l'erreur moyenne absolue afin de pouvoir la comparer avec l'erreur précédente. Cette erreur correspond à une moyenne de toutes les valeurs affichées sur le graphe précédent. On a obtenu une erreur absolue :

$$E(horner) = 4.62718 * 10^{-6} \tag{4}$$

2.3 Troisième Partie : Comparaison

Dans cette partie nous nous intéressons à la comparaison des deux erreurs afin de définir laquelle des deux méthodes doit être préconisée pour calculer notre polynôme. Pour rappel voici les moyennes des deux erreurs absolues que nous avons calculé :

$$E(classique) = 1.5822 * 10^{-5} \tag{5}$$

$$E(horner) = 4.62718 * 10^{-6} \tag{6}$$

On constate donc que l'erreur par la méthode classique est trois fois plus grande que celle de Horner. Par conséquent il est préférable, afin d'avoir une erreur moindre, d'utiliser la méthode de Horner pour calculer notre polynôme.

2.4 Quatrième Partie : Importance de la mantisse

La précision de la machine simulée est évidement un facteur essentiel dans l'approximation d'une valeur. C'est pourquoi notre programme Scilab propose à l'utilisateur de choisir le nombre de bits de la mantisse (entre 16 et 23) qu'il utilisera pour calculer les valeurs approchées.
Pour rappel, les valeurs calculées précédemment et les graphes nous avons utilisé une mantisse de 16 bits.Voila les erreurs que nous obtenons pour une mantisse de 23 bits :

$$E(classique) = 1.30713 * 10^{-7} \tag{7}$$

$$E(horner) = 4.02634 * 10^{-8} \tag{8}$$

On constate d'après ces résultats que la taille de la mantisse influe grandement sur les erreurs puisque en passant d'une mantisse de 16 à 23 bits les erreurs ont pratiquement été divisées par 100.

3 Conclusion

Grâce ce travail d'analyse numérique nous constatons que :

-La méthode de calcul joue un rôle prédominant dans l'importance de l'erreur. En effet le polynôme calculé sous la forme de horner nous fournit une erreur moindre que lorsqu'il est calculé sous la forme classique.

-La taille de la mantisse est inversement proportionnelle à l'erreur puisque, si l'on augmente celle-ci on diminue l'erreur. Ce qui est logique au vue de la définition de la mantisse.

Pour conclure nous pouvons affirmer que la précision des résultats d'un programme informatique ne dépend pas uniquement de l'efficacité de la machine qui le fait fonctionner, mais aussi de la façon dont ce programme a été pensé et conçu.