Modèles de calcul :
Multiplication de Polynômes
MASTER 1 CCA

BLIBEK RANIA 21215298
IDRES AMINE 21322043

7 janvier 2024

Chapitre 1

Présentation du projet

La multiplication de polynômes est une opération fondamentale en mathématiques appliquées, trouvant des applications dans divers domaines tels que le traitement du signal, l'informatique, et l'algèbre linéaire.
Cependant, lorsque les polynômes impliqués sont de degrés élevés, la multiplication conventionnelle peut devenir coûteuse en termes de temps d'exécution.

Le projet que nous entreprenons s'inscrit dans le cadre d'une exploration approfondie des concepts mathématiques et algorithmiques fondamentaux, visant à enrichir notre compréhension des opérations arithmétiques sur les nombres complexes, à maîtriser l'algorithme FFT (Fast Fourier Transform), à développer des méthodes de multiplication de polynômes, et enfin, à comparer l'efficacité de différentes approches.
Cette entreprise intellectuelle et technique représente une étape significative dans notre parcours académique en informatique et en mathématiques, nous offrant une opportunité unique d'appliquer des concepts théoriques complexes à des problèmes pratiques concrets.

Contexte et Importance des Nombres Complexes :

Les nombres complexes, bien que souvent perçus comme abstraits, jouent un rôle crucial dans divers domaines scientifiques, notamment en ingénierie, en physique et en informatique.
En décomposant un nombre en une partie réelle et une partie imaginaire, les nombres complexes fournissent un outil puissant pour représenter et manipuler des grandeurs vectorielles. Dans le cadre de notre projet, nous allons approfondir notre compréhension de l'arithmétique des nombres complexes en mettant en œuvre des opérations telles que l'addition, la soustraction, la multiplication et la division, avec une attention particulière portée à la gestion des cas particuliers et à la préservation de la précision numérique.

Objectifs Algorithmiques et FFT :

Une composante essentielle de notre projet est la mise en œuvre de l'algorithme FFT, une méthode efficace pour calculer les transformations de Fourier rapides. Cette approche algorithmique est particulièrement pertinente dans le contexte de la multiplication de polynômes.
L'algorithme FFT nous offre la possibilité de transformer le problème de la multiplication de polynômes dans le domaine fréquentiel, réduisant ainsi la complexité temporelle de l'opération de manière significative. Notre objectif est de maîtriser cette technique et de comprendre comment elle peut être appliquée de manière optimale .

Multiplication de Polynômes et Comparaison d'Efficacité :

Une partie substantielle de notre projet consiste en la mise en œuvre de deux approches distinctes pour la multiplication de polynômes.
L'algorithme naïf, basé sur une simple multiplication terme à terme, sera comparé à une version optimisée utilisant l'algorithme FFT.
Nous cherchons à évaluer l'efficacité de ces méthodes en fonction de la taille des polynômes et à analyser les gains obtenus en termes de complexité temporelle.
Cette étape cruciale nous permettra de tirer des conclusions sur les scénarios dans lesquels l'utilisation de l'algorithme FFT offre un avantage significatif.

En conclusion, notre projet se propose d'explorer en profondeur les fondements mathématiques et algorithmiques qui sous-tendent les opérations sur les nombres complexes, l'algorithme FFT et la multiplication de polynômes.

Nous aspirons à transcender la simple implémentation technique en comprenant les motivations théoriques derrière chaque choix algorithmique, ouvrant ainsi la voie à une maîtrise plus profonde des concepts mathématiques complexes appliqués à des problèmes informatiques concrets.

Chapitre 2

Implémentation :

2.1 Implémentation des Opérations Arithmétiques sur des Nombres Complexes :

Description des Structures et Fonctions Utilisées :

Dans notre projet, nous avons défini une structure Complex pour représenter un nombre complexe, composé de parties réelle et imaginaire de type double.
Cette structure nous permet de manipuler les nombres complexes de manière modulaire, en regroupant leurs composants au sein d'une seule entité.

Nous avons également mis en place un ensemble de fonctions dédiées à la manipulation de nombres complexes, comprenant des opérations telles que l'addition, la soustraction, la multiplication, la division et le calcul de la magnitude.

Explication de la Logique derrière Chaque Opération :

Addition et Soustraction :
L'addition de deux nombres complexes a et b se fait en ajoutant leurs parties réelles et imaginaires respectives.
La soustraction est similaire, avec la soustraction des parties réelles et imaginaires de b de celles de a.

Multiplication et Division :
La multiplication de deux nombres complexes a et b suit la règle distributive, avec un ajustement pour les termes imaginaires.
La division est réalisée en multipliant le numérateur et le dénominateur par le conjugué du dénominateur.

Cette section du projet vise à fournir une base solide pour comprendre et manipuler les nombres complexes, permettant ainsi une transition fluide vers des opérations plus avancées telles que la FFT et la multiplication de polynômes.

2.2 Implémentation de l'Algorithme FFT :

Description de la Fonction FFT et IFFT :

Dans notre projet, nous avons implémenté l'algorithme de transformée de Fourier rapide (FFT) pour effectuer une transformation rapide entre le domaine des coefficients d'un polynôme et le domaine des fréquences.
La fonction fft prend en entrée un tableau de nombres complexes L, sa taille n, et un paramètre sign indiquant si la transformation est directe (sign = 1) ou inverse (sign = -1).

La fonction ifft représente l'algorithme de transformée de Fourier inverse (IFFT), qui est utilisé pour revenir du domaine des fréquences au domaine des coefficients.
Elle appelle la fonction fft avec le signe opposé et normalisé ensuite les résultats en divisant par la taille du vecteur.

Gestion de la Taille des Vecteurs pour Qu'ils Soient des Puissances de 2 :

L'algorithme FFT est le plus efficace lorsqu'il est appliqué à des vecteurs dont la taille est une puissance de 2.
Pour gérer cela, notre implémentation utilise une approche récursive de division du problème.
Si la taille du vecteur n n'est pas une puissance de 2, l'algorithme divise le vecteur en parties pairs et impairs, puis récursivement applique la FFT à ces sous-vecteurs de taille réduite.

Cette approche garantit que chaque appel récursif travaille sur des vecteurs de taille puissance de 2, optimisant ainsi les performances de l'algorithme.

Cette section de notre projet représente une étape cruciale pour la multiplication de polynômes à l'aide de l'algorithme FFT.
En transformant les polynômes dans le domaine des fréquences, nous optimisons la multiplication, préparant ainsi le terrain pour une comparaison d'efficacité entre l'algorithme naïf et celui basé sur la FFT.

2.3 Implémentation de la Multiplication Naïve de Polynômes :

Description de l'Algorithme Naïf :I

L'algorithme naïf pour la multiplication de polynômes que nous avons implémenté suit une approche simple et directe.
La fonction multiplicationnaive prend en entrée deux tableaux d'entiers représentant les coefficients des polynômes, ainsi que leurs degrés respectifs (n et m).
Elle retourne le tableau résultant de la multiplication des deux polynômes.

L'algorithme consiste à parcourir tous les termes des deux polynômes et à accumuler les produits dans le tableau résultant.
La complexité de cet algorithme est quadratique, ce qui peut le rendre inefficace pour des polynômes de degrés élevés.

Cette approche simple, bien que facile à comprendre, peut devenir inefficace pour des polynômes de degrés élevés en raison de sa complexité quadratique.

2.4 Implémentation de la Multiplication de Polynômes avec l'Algorithme FFT :

Description de l'Algorithme FFT :
Notre implémentation de la multiplication de polynômes avec l'algorithme FFT est basée sur la transformation des polynômes du domaine des coefficients au domaine des fréquences à l'aide de la FFT.
Les polynômes sont représentés sous forme de nombres complexes, où la partie réelle correspond aux coefficients du polynôme, et la partie imaginaire est initialisée à zéro.

La multiplication dans le domaine des fréquences est effectuée en multipliant les coefficients correspondants des polynômes.
Ensuite, la transformée de Fourier inverse (IFFT) est appliquée pour revenir au domaine des coefficients.

Conversion entre le Domaine des Coefficients et le Domaine des Fréquences :

Les polynômes sont transformés dans le domaine des fréquences en appliquant la FFT à leurs représentations complexes.
Les coefficients réels des polynômes sont utilisés pour initialiser les parties réelles des nombres complexes, tandis que les parties imaginaires sont initialisées à zéro.

Une fois la multiplication dans le domaine des fréquences effectuée, l'IFFT est appliquée pour obtenir le résultat dans le domaine des coefficients.

Cette approche utilise efficacement la FFT pour optimiser la multiplication de polynômes, surtout lorsque les degrés des polynômes deviennent importants.
Elle offre une meilleure complexité asymptotique par rapport à l'algorithme naïf.

2.5 Présentation détaillée des Résultats :

L'évaluation de l'efficacité des deux méthodes de multiplication de polynômes, à savoir la méthode naïve et l'algorithme FFT, a nécessité l'exécution de tests de performance approfondis.
Ces tests ont été élaborés dans le but de mesurer avec précision les temps d'exécution pour différentes tailles de polynômes, fournissant ainsi un aperçu détaillé de la performance relative de chaque approche.

Collecte des Données :

Pour chaque méthode, des jeux de données variés, représentant des polynômes de différentes tailles, ont été générés.
Ces polynômes ont été soigneusement choisis pour couvrir une plage significative de scénarios, allant de petites tailles à des polynômes de grande envergure.
Les temps d'exécution ont été enregistrés avec précision pour chaque taille de polynôme.

Fichiers de Résultats :

Les résultats sont soigneusement consignés dans deux fichiers distincts : "timesnaive.txt" pour la méthode naïve et "timesfft.txt" pour l'algorithme FFT.
Chaque ligne de ces fichiers correspond à une taille spécifique de polynôme, et la valeur enregistrée représente le temps d'exécution associé.

Utilisation de Scripts Gnuplot pour la Visualisation :

Afin de faciliter la compréhension et l'interprétation des résultats, des scripts Gnuplot ont été créés.
Chaque script Gnuplot est conçu de manière à présenter de manière claire et visuelle les performances respectives de la méthode naïve et de l'algorithme FFT.
Ces graphiques facilitent la comparaison et la compréhension des résultats obtenus, offrant ainsi un aperçu intuitif des avantages de chaque approche en fonction de la taille des polynômes.

Chapitre 3

Résultats obtenus :

La phase de comparaison d'efficacité a été réalisée à travers une série de tests approfondis, visant à évaluer les performances des deux méthodes de multiplication de polynômes que nous avons implémentées.
Ces tests ont été conçus pour mesurer avec précision les temps d'exécution en fonction de différentes tailles de polynômes.

Cette approche visuelle offre une perspective claire et concise sur la manière dont les performances évoluent en fonction de la taille des polynômes.
Les graphiques obtenus à partir des scripts Gnuplot permettent une interprétation rapide et facilitent la comparaison des deux algorithmes.
Cette méthode d'analyse graphique ajoute une dimension intuitive à notre évaluation, offrant ainsi une compréhension plus approfondie des performances relatives des méthodes naïve et FFT dans le contexte de la multiplication de polynômes.

La différence entre la multiplication naïve de polynômes et la multiplication utilisant l'algorithme FFT (Transformée de Fourier Rapide) réside essentiellement dans leur approche algorithmique, leur complexité temporelle, et leur performance dans des contextes spécifiques.
Examions ces deux méthodes de multiplication de polynômes de manière détaillée.
Multiplication Naïve de Polynômes :
La multiplication naïve de polynômes implique une approche élémentaire, où chaque terme du premier polynôme est multiplié par chaque terme du second polynôme, et les résultats sont ensuite additionnés pour former le polynôme final.
C'est un processus simple à comprendre et à mettre en œuvre, mais sa complexité temporelle est quadratique, ce qui signifie qu'elle peut devenir inefficace pour des polynômes de grandes tailles.
Multiplication de Polynômes avec FFT :
L'algorithme FFT, en revanche, adopte une approche plus sophistiquée en exploitant la transformée de Fourier rapide.
Cette méthode divise les polynômes d'entrée en sous-polynômes de taille plus petite, applique la transformée de Fourier sur ces sous-polynômes, effectue la multiplication termes à termes dans le domaine de la fréquence, puis applique l'inverse de la transformée de Fourier pour obtenir le résultat final. L'utilisation de FFT réduit la complexité temporelle à O(nlog[U+2061]n)O(nlogn), ce qui est significativement plus efficace pour des polynômes de grande taille.
Comparaison et Meilleur Cas d'Utilisation :

La multiplication naïve excelle pour des polynômes de taille relativement petite en raison de sa simplicité et de sa facilité d'implémentation.
Cependant, à mesure que la taille des polynômes augmente, la multiplication avec FFT devient rapidement plus performante en raison de sa complexité temporelle inférieure.

Meilleur Cas d'Utilisation :

Multiplication Naïve : Convient aux situations où la taille des polynômes est petite, et la simplicité de l'algorithme est un avantage.
Multiplication avec FFT : Idéale pour des polynômes de grande taille, où la réduction significative du temps d'exécution devient cruciale.

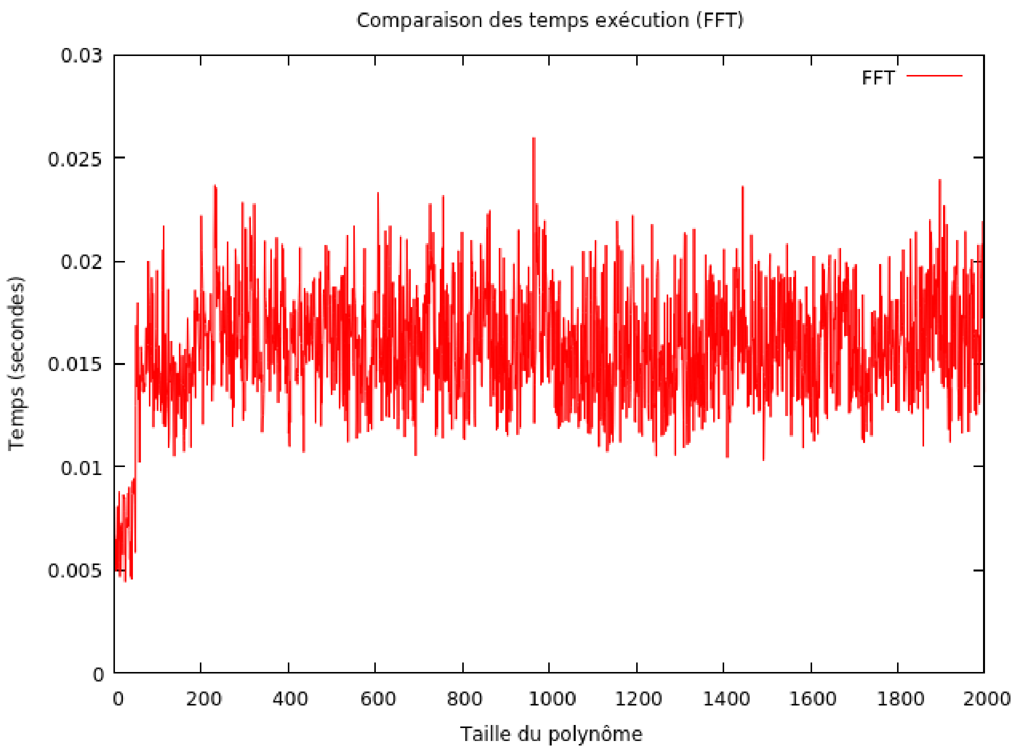

FIGURE 3.1

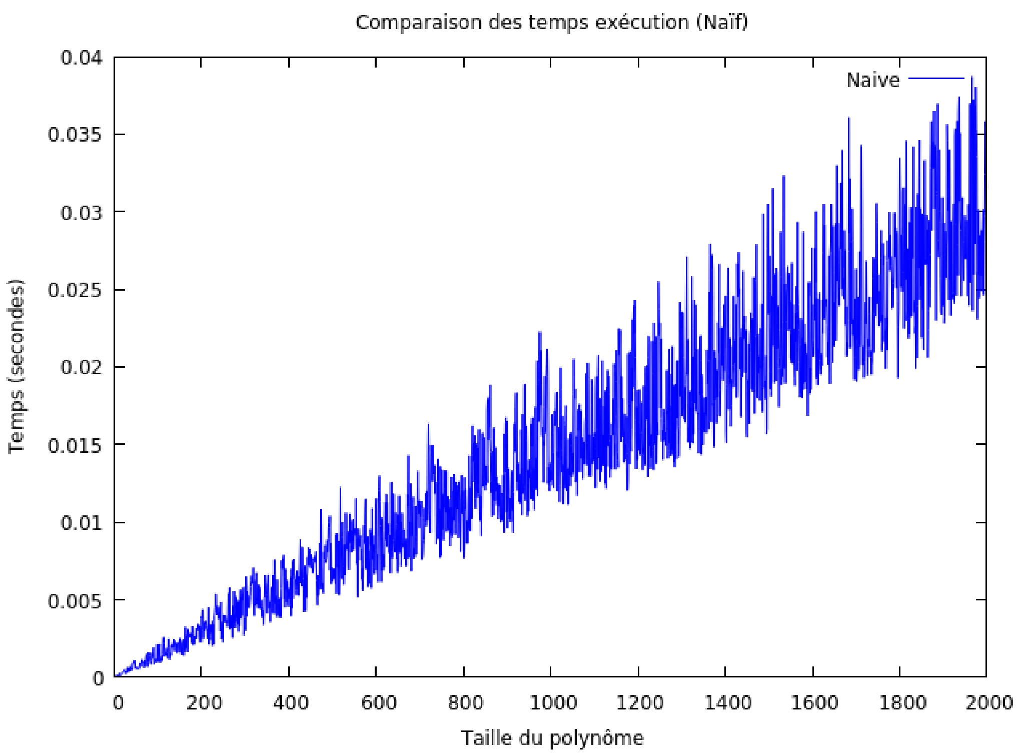

FIGURE 3.2

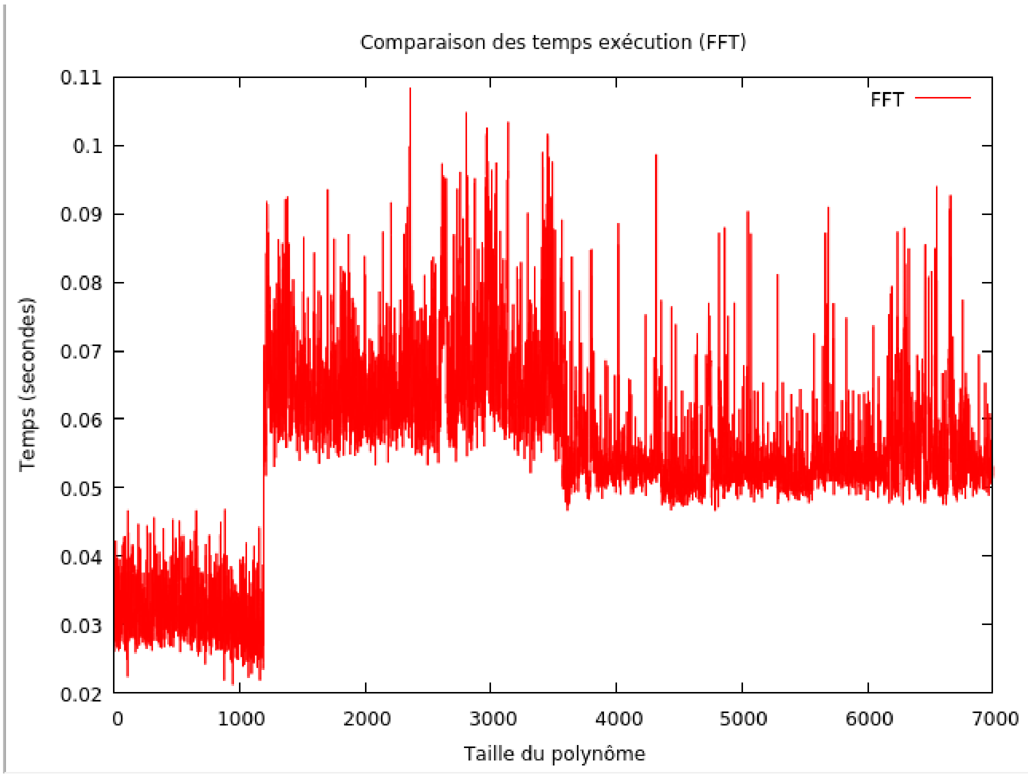

Figure 3.3

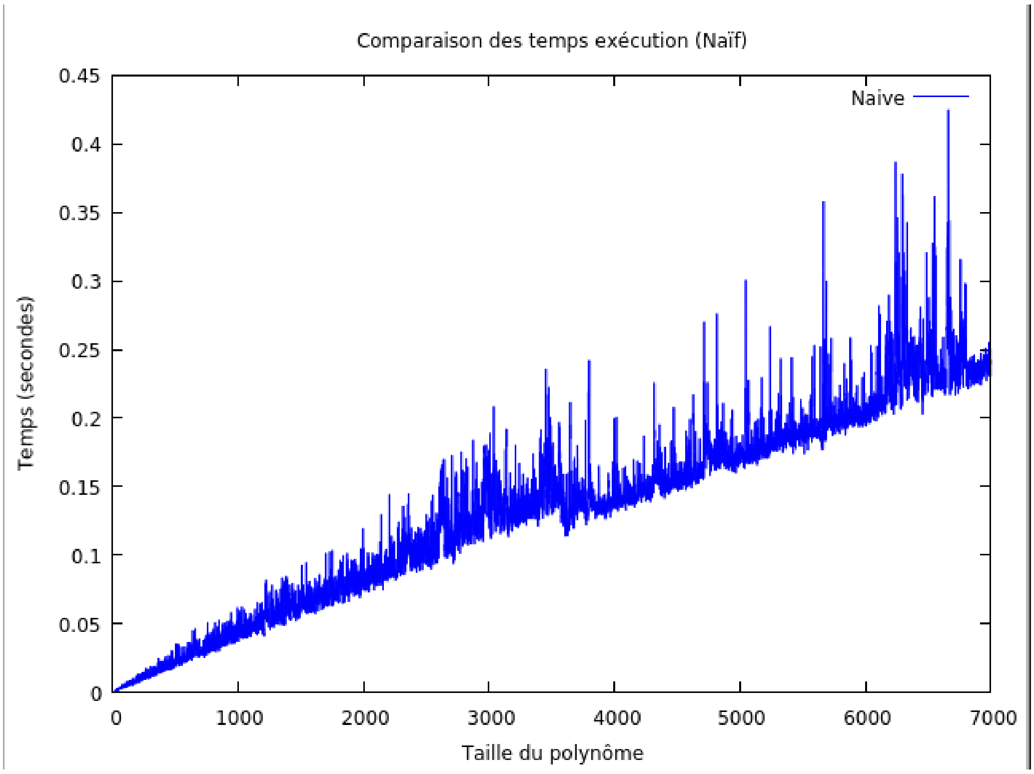

Figure 3.4

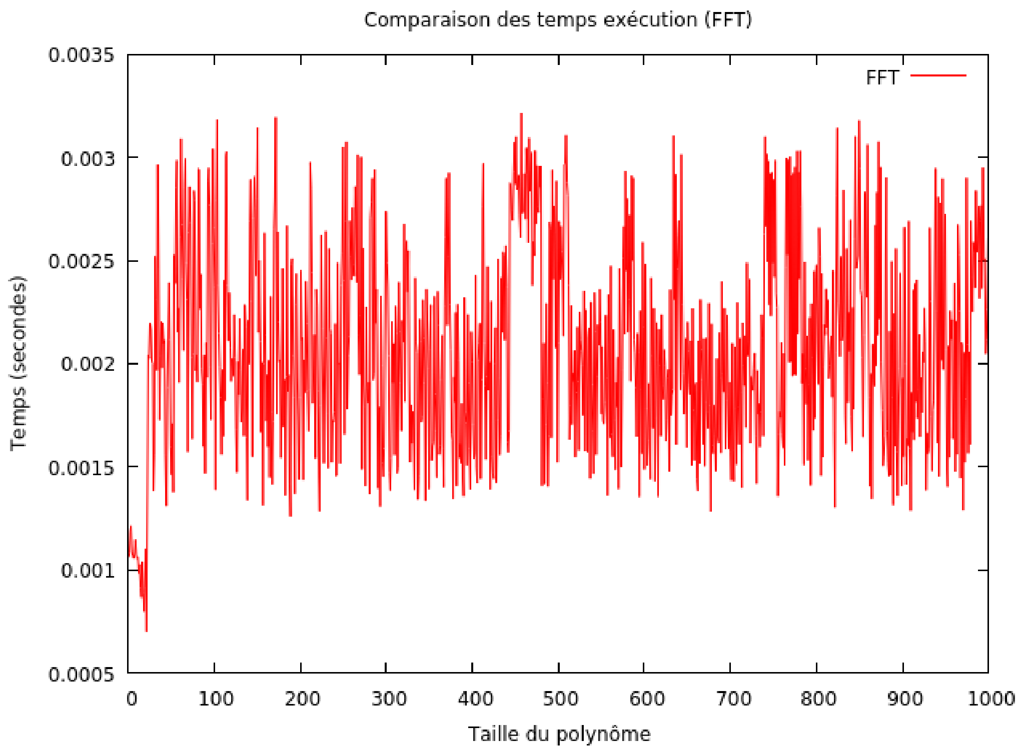

FIGURE 3.5

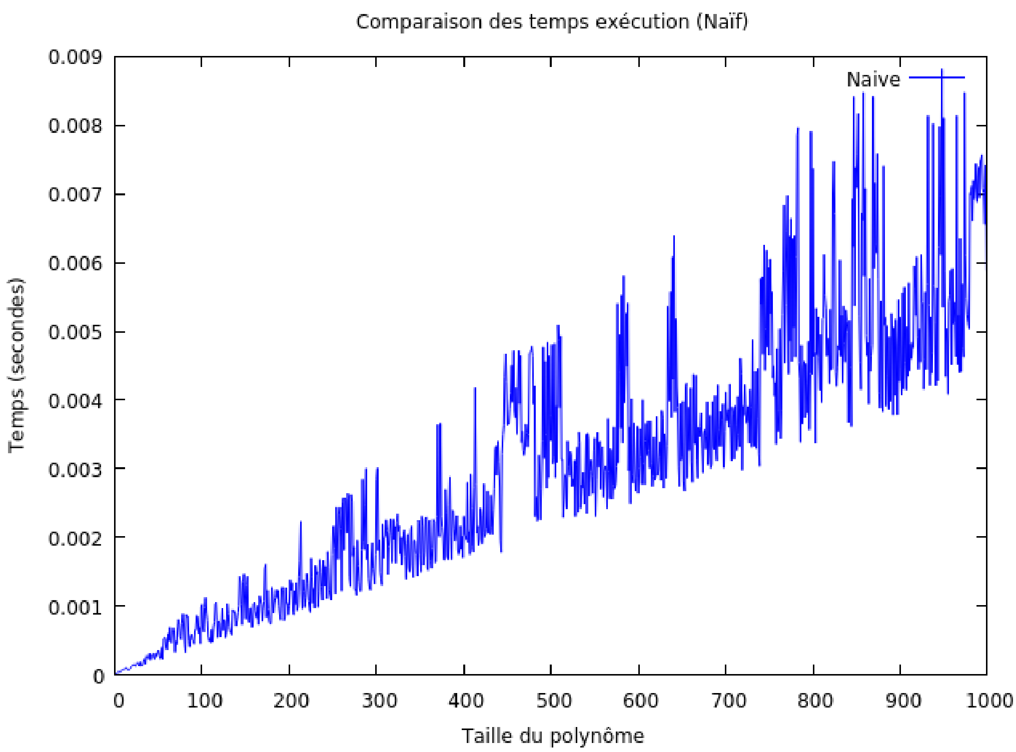

FIGURE 3.6

Chapitre 4

Conclusion :

Le projet dans son ensemble a été une immersion fascinante dans le monde des algorithmes numériques, de la manipulation de polynômes, et de la performance des opérations mathématiques.
À travers la mise en œuvre d'opérations arithmétiques sur des nombres complexes, de l'algorithme FFT (Transformée de Fourier Rapide), et de la multiplication de polynômes, nous avons atteint une compréhension plus profonde des subtilités algorithmiques et des implications pratiques de ces concepts.
Le parcours de ce projet n'a pas été exempt de défis stimulants, chacun servant de tremplin pour un apprentissage plus approfondi et une collaboration renforcée au sein de l'équipe.
Ces défis ont forgé notre compréhension collective des subtilités algorithmiques et ont mis en lumière l'importance cruciale de l'esprit d'équipe.

L'un des défis majeurs a été de s'assurer que les vecteurs utilisés dans l'algorithme FFT avaient une taille optimale, c'est-à-dire une puissance de 2.
La gestion de la mémoire et la récursivité de l'algorithme ont demandé une réflexion approfondie. L'équipe a travaillé en étroite collaboration pour élaborer des solutions élégantes et assurer une allocation de mémoire efficace.

L'esprit d'équipe a été le pilier de notre progression. Les discussions ouvertes, le partage d'idées et la résolution collective des problèmes ont renforcé notre compréhension mutuelle et ont contribué à des solutions plus robustes.
Chacun a apporté sa perspective unique, créant un environnement où les défis étaient perçus comme des opportunités d'apprentissage.

Cette expérience a transcendé la simple réalisation d'un projet académique, elle a été une exploration collective, un voyage où la somme des compétences individuelles a donné naissance à quelque chose de plus grand.
C'est dans la résolution commune des difficultés que l'esprit d'équipe s'est révélé être l'élément catalyseur de notre succès.
Ce projet ne se limite pas à des lignes de code et à des résultats, il incarne l'esprit collaboratif qui est essentiel pour relever les défis complexes du monde de l'informatique et des mathématiques computationnelles.

En conclusion, ce projet nous a fourni une expérience pratique et approfondie dans le domaine de l'algèbre numérique et des algorithmes avancés.
Il a renforcé notre capacité à traduire des concepts théoriques en code fonctionnel et à évaluer les performances des algorithmes dans des contextes spécifiques.

Pour l'avenir, des améliorations pourraient être envisagées, telles que l'optimisation des opérations sur les nombres complexes, l'exploration d'autres techniques de multiplication de polynômes, ou même l'extension du projet vers d'autres domaines liés aux mathématiques computationnelles.

En somme, ce projet n'a pas seulement élargi notre compréhension des algorithmes mathématiques, mais il a également stimulé notre créativité dans la résolution de problèmes complexes, posant ainsi les bases d'une expertise croissante dans le domaine des calculs numériques avancés.