Programmation sous Matlab

Durée 1^{H30}.

1 Restitution du travail

Pour restituer votre travail, vous devez placer tous les fichiers de vos programme dans un dossier **dont le titre est votre nom** sur le bureau de votre espace de travail – par exemple **DurandBruno**. Ce dossier doit contenir **au minimum** un fichier dont le titre est **examen104.m** contenant votre programme principal. En fin d'épreuve, vous sortez de la salle en laissant votre session ouverte. L'enseignant passera récupérer votre travail sur une clé USB et fermera votre session.

Dans votre intérêt, ce dossier ne doit contenir que les fichiers utiles.

Tous les documents sont autorisés, par contre les échanges quels qu'ils soient sont interdits. Vous pouvez utiliser des documents que vous auriez sur une clé USB, par contre vous ne pouvez charger ces documents que pendant la première demi-heure de l'épreuve. **Si une clé USB persiste sur votre ordinateur au bout d'une demi-heure, cela sera considéré comme une tentative de fraude**.

Attention ! ! ! une heure limite de restitution de votre travail vous sera précisé en début de séance. Après cette heure, vous devez quitter la salle. Deux minutes de plus comptent pour un point de moins sur la note finale.

N'oubliez pas de commenter votre programme. La qualité de la présentation du programme comptera naturellement pour la note finale. N'oubliez pas que les majuscules et le minuscules sont différentes sous Matlab. Respectez les consignes. Bon travail.

2 But du travail

Le but de ce travail est de résoudre le problème qui a longtemps porté le nom de"quadrature du cercle" c'est à dire le calcul approximatif du nombre π par une méthode de Monte Carlo (appelée ainsi car elle utilise un tirage aléatoire).

Pour que ce travail soit intéressant, il fera l'objet d'une illustration.

Le principe du calcul du nombre π est le suivant :
- on tire au hasard des points dans un espace $3D$ dans le cube de coté $[-1, 1]^3$,
- on sélectionne uniquement ceux inclus dans la sphère de rayon 1,
- on on calcule la proportion des points qui sont dans la sphère.

On montre que la proportion des points du cube qui sont aussi dans la sphère converge vers le rapport du volume de la sphère et ce celle du cube.
Volume du cube : $V_C = L^3$ (où L est la largeur du cube).
Volume de la sphère : $V_S = \frac{4}{3}\pi R^3$ (ou R est le rayon de la sphère).

Ici on a $L = 2R = 2$, donc $\frac{V_S}{V_C} = \frac{\frac{4}{3}\pi}{2^3} = \frac{\pi}{6}$. Donc $6\frac{V_S}{V_C} = \pi$.

La convergence de cet algorithme est très rapide pour les premières décimales mais cela se gâte par la suite. Ce qu'on vous propose ici c'est de faire un petit programme structuré montrant cette convergence.

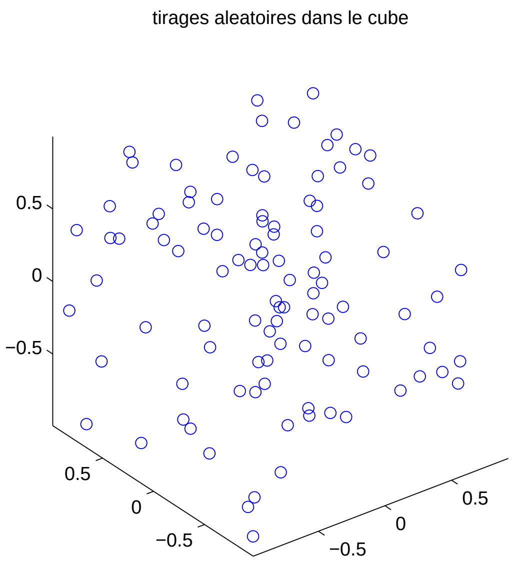

FIGURE 1 – Exemple de figure à obtenir.

3 Développement du logiciel

Remarque : toutes vos figures doivent avoir un titre et les axes doivent être commentés.

Question 1. Faites une fonction dont le prototype d'appel est :
$[C] = TirageAleatoire(N)$,
donnant un vecteur de N points $3D$ dans un cube de côté $[-1, 1]$ (c'est à dire des points dont les coordonnées sont comprises entre -1 et $+1$).

Question 2. Appelez cette fonction dans votre programme principal avec 100 points dans un premier temps. Faites afficher les points $3D$ sous forme de petits cercles bleus (voir Figure 1) en prenant bien garde que les trois axes aient la même longueur à l'affichage. Intitulez cette figure "tentative de calcul du nombre pi".

Question 3. Faites une fonction dont le prototype d'appel est :
$[S] = SelectionSphere(C)$,
donnant le vecteur des points de C qui sont inclus dans la sphère unité (c'est à dire dont les coordonnées (x, y, z) sont telles que $x^2 + y^2 + z^2 \leq 1$).

Question 4. Appelez cette fonction dans votre programme principal à la suite de la fonction précédente. Superposez les points ainsi calculés sur la figure précédente en affichant les points de S sous forme de petites étoiles rouges. Ajoutez une grille sur cette figure.

Question 5. Faites une fonction dont le prototype d'appel est :
$[estime_pi] = EstimationDePi(C, S)$,
donnant une estimation du nombre π sur la base du nombre de points de C et du nombre de points de S. Vérifiez la différence entre la vraie valeur de π et son estimation avec un tirage de 100 points.

On va maintenant utiliser vos fonctions pour regarder la convergence de l'estimation de π vers π. On utilisera pour valeur de référence, celle donnée par Matlab ®.

Question 6. Dans une boucle faites varier N de 100 à 10000 par pas de 100. Pour chaque valeur de N, affichez la superpositions des points du cube et de ceux de la sphère (attention l'affichage doit avoir lieu pendant le calcul) et calculez une estimation $\hat{\pi}$ de la valeur de π grâce aux fonctions que vous avez programmé. Affichez sur un graphique l'évolution de la différence $|\hat{\pi} - \pi|$ en fonction de N sur une figure intitulée “convergence vers pi”.

Question 7. Faites la même expérience (sans afficher à nouveau les points) en faisant afficher sur une autre figure $20log(\hat{\pi}/\pi)$ en fonction de $log(N)$. Répétez cette expérience 8 fois en superposant les courbes avec des couleurs différentes (vous pouvez demander a Matlab de vous aider pour faire ça).