

Calcul Haute Performance - Travaux Pratiques 1
--

Partie 1 : Programmation de la méthode du pivot de Gauss en C++

Les fichiers .h et .cpp associés à des classes *Vecteur*, *Matrice* et *Solveur* vous sont fournis ainsi qu'un programme principal *Main.cpp*.

Le travail consiste ici à compléter ces différents fichiers-sources, en particulier la méthode *SolveGauss*, pour permettre la résolution d'un système linéaire par la méthode du pivot de Gauss.

1. Compléter la phase de triangularisation de la matrice A et la vérifier pour les systèmes SYS1 (dimensions 3 et 4), SYS2 et SYS3.
2. Compléter la résolution du système triangulaire et appliquer l'algorithme aux systèmes SYS1 (dimensions 3, 4 et 100), SYS2 et SYS3.
3. Vérifier la validité des solutions en comparant les produits $A*x$ aux vecteurs b (vous pourrez utiliser à cette fin la surcharge de l'opérateur * entre des objets de type *Matrice* et *Vecteur* ainsi qu'une méthode calculant la norme d'un *Vecteur*).

Partie 2 : Programmation de la méthode de la plus forte pente en C++

1. Compléter les surcharges d'opérateur nécessaires à l'expression de l'algorithme de la plus forte pente en vous inspirant de la surcharge donnée (fichier *Solveur.cpp*).
2. Compléter la classe *Solveur* avec une méthode *SolveSD* qui permettra la résolution d'un système linéaire par la méthode de la plus forte pente. La méthode prendra en paramètres un nombre d'itérations maximal et une valeur maximale sur la norme du vecteur résidu. Donner les solutions numériques obtenues pour les systèmes SYS1 (dimension 4) et SYS2 après 100 itérations.
3. Stocker les normes successives du vecteur résidu dans un objet de type *Vecteur*. Exporter son contenu dans un fichier texte puis tracer l'évolution de cette quantité au fil des itérations pour le système SYS1 (dimension=100, résidu maximal de 0.001).

Remarques :

- S'appuyer sur du code source commenté et des captures d'écran pour justifier les réponses.
- Commenter pertinemment les résultats.
- Donner vos fichiers sources .cpp et .h **dans un état compilable**.