

Introduction

Le calcul scientifique est une discipline qui consiste à développer, analyser et appliquer des méthodes relevant de domaines mathématiques aussi variés que l'analyse, l'algèbre linéaire, la géométrie, la théorie de l'approximation, les équations fonctionnelles, l'optimisation ou le calcul différentiel. Le développement de l'analyse numérique, et des mathématiques en général, a été et sera toujours nécessaire pour la résolution des problèmes de plus en plus

complexes posés par la physique, les sciences biologiques, les sciences de l'ingénieur, l'économie et la finance et des autres sciences.

Ce polycopie s'adresse aux étudiants en domaine de science de la matière et sciences de la technologies des filières physique fondamental de deuxième et troisième années et aussi des autres spécialités comme Génie civil, Génie des procédés,..., et rejoint le programme officiel du cours d'analyse numérique de ces filière. Il présente les différentes méthodes avec les fondements théoriques nécessaires, et les exemples d'utilisation de ces méthodes facilitent la conception de leurs diagrammes avec les différents langages.

J'écrirai l'algorithme de chaque méthode pour faciliter de faire le programme avec les différents langages.

Les connaissances nécessaires pour l'assimilation du contenu de cet ouvrage sont les fonctions non linéaires $f(x) = 0$, interpolation, intégration numérique, la résolution du système linéaire et la résolution des équations différentielles.

Toutes les méthodes numériques sont programmées par le biais du "langage" Matlab. Ce dernier est commercialisé par la société MathWorks (<http://www.mathworks.com/>).

Matlab est un langage interprété, son fonctionnement est différent des langages classiques (Fortran, Pascal, ...), dits langages compilés. Un algorithme écrit en langage interprété nécessite pour fonctionner un interprète. Ce dernier est un programme traduisant directement les instructions, en langage machine, au fur et à mesure de leurs exécutions. L'interprète analyse séquentiellement la syntaxe de l'algorithme avant de le dérouler dynamiquement. En revanche, dans un langage compilé, le code source est lu dans un premier temps puis compilé par un compilateur qui le convertit en langage machine directement compréhensible par l'ordinateur. Il en résulte ainsi, qu'un langage interprété sera plus lent qu'un langage compilé à cause de la conversion dynamique de l'algorithme, alors que cette opération est réalisée préalablement pour un langage compilé. Néanmoins, l'un des avantages majeur d'un langage interprété, tient à la facilité de détection d'éventuelles erreurs de programmation.