COURS DE CALCUL FORMEL EN M1 : TP 3

CALCUL DE SÉRIES PAR ITÉRATION DE NEWTON

L'objectif de ce TP est l'utilisation de l'itération de Newton pour la résolution en série de plusieurs types d'équations ou de systèmes. Une motivation importante est fournie par les séries génératrices d'énumération en combinatoire : il s'agit de séries formelles de la forme

$$\sum_{n=0}^{\infty} a_n z^n \qquad \text{ou} \qquad \sum_{n=0}^{\infty} a_n \frac{z^n}{n!},$$

où a_n désigne le nombre d'objets de taille n dans une certaine famille. Dans le premier cas, la série génératrice est dite *ordinaire* ; elle est *exponentielle* dans le second.

1. Introduction : arbres binaires

Un arbre binaire à n sommets est soit vide ($n = 0$), soit composé d'une racine et de deux sous-arbres binaires de taille k et $n-k-1$. Les nombres C_n d'arbres binaires de taille n s'appellent les nombres de Catalan.

(1) Écrire une procédure prenant une taille n en argument et renvoyant C_n à partir d'une récurrence non-linéaire simple.

(2) À partir de $C_0, \dots, C_{10}$, deviner un polynôme dont la série génératrice $\sum C_n z^n$ est solution.

(3) Résoudre ce polynôme et vérifier que sa solution donne bien les C_n jusqu'à $n = 100$.

(4) Deviner une récurrence linéaire satisfaite par les C_n, et la résoudre.

2. Arbres d'arité 5 et équations algébriques

Ces arbres sont définis comme les arbres binaires, chaque sommet interne ayant cinq sous-arbres et non plus 2.

(5) Calculer le nombre B_n d'arbres d'arité 5 à n sommets, pour $n = 0, \dots, 20$.

(6) Deviner un polynôme dont la série génératrice $\sum B_n z^n$ est solution (il faudra changer des réglages par défaut de `gfun`, voir `?gfun,Parameters`).

(7) À l'aide de ce polynôme et d'une itération de Newton, calculer les nombres B_n, $n = 0, \dots, 63$.

(8) (À la fin, s'il reste du temps). Utiliser les 50 premiers de ces nombres pour deviner une récurrence sur les B_n, la résoudre, et confirmer la solution en comparant la valeur pour $n = 60$.

3. Des arbres bicolorés et un système

Il est possible de considérer des règles plus ou moins complexes de construction. Par exemple, des arbres dont les sommets peuvent être bleus ou verts, ont une arité finie mais illimitée, avec la contrainte qu'une racine bleue a au moins un fils vert, alors qu'une racine verte a au moins deux fils bleus. Ces arbres ont des séries génératrices d'énumération qui sont solution de

$$B(z) = z + \frac{zV(z)}{(1-V(z))(1-B(z))}, \qquad V(z) = z + \frac{zB(z)^2}{(1-V(z))(1-B(z))}.$$

(9) Construire la matrice jacobienne du système ;

(10) l'utiliser pour former une itération de Newton ;

(11) calculer le nombre d'arbres bicolorés dont la racine est bleue pour les tailles $n = 0, \dots, 31$.

4. Arbres ordonnés et équations différentielles

Un arbre ordonné de taille n est un arbre dont chaque nœud porte une étiquette distincte entre 1 et n, de telle sorte que l'étiquette d'une racine soit toujours plus petite que les étiquettes de ses fils. Pour les arbres ternaires (d'arité 3) ordonnés, l'équation satisfaite par la série génératrice exponentielle est

$$Y'(z) = 1 + Y(z)^3.$$

(12) Déterminer l'équation différentielle linéaire à utiliser à chaque itération de Newton, et écrire une procédure permettant de la résoudre en série ;

(13) calculer ensuite par itération de Newton les 32 premiers coefficients.

5. Arbres 2-3 et équation fonctionnelle

Les arbres 2-3 sont une structure de données utilisée en bases de données. Ils permettent la recherche, l'insertion et la suppression en temps amorti logarithmique. Les nœuds internes ont deux ou trois fils, et les nœuds externes sont tous au même niveau. Leur série génératrice d'énumération vérifie l'équation fonctionnelle

$$T(z) = z + T(z^2 + z^3).$$

(14) Déterminer l'équation linéarisée à résoudre pour l'itération de Newton ;

(15) écrire une procédure permettant de résoudre cette équation par itération ;

(16) écrire l'itération de Newton et calculer les 64 premiers coefficients de la série.