

Rekursionen - Wie, was, warum?

Unterlagen für Lehrende

Univ.-Prof. Dr. C. Fuchs
22.04.2025

Grundsätzlich: Eine lineare Rekursionsgleichung hat die Form (A_n) mit $A_{n+k} + a_1 A_{n+k-1} + \dots + a_k A_n = f(n)$ für $n \geq 0$. Wir beschäftigen uns mit ihrer Lösung (A_n) sowie der Begründung und Herleitung von expliziten Darstellungen für das n -te Glied A_n der Lösungsfolge.

Technologieeinsatz mit Sage:

Link zu SageCell: <https://sagecell.sagemath.org/>

Ein nützliches Sage-Paket `rec_sequences`

Einige nützliche Sage-Befehle: `LinearSolveSequence`, `FunkctionalEquation`

Programm in Stichworten:

Existenz und Eindeutigkeit von Lösungen einer Rekursionsgleichung mit Anfangswerten, Binet-Darstellung (Begründung und Herleitung mit Hilfe des charakteristischen Polynoms und mit erzeugenden Funktionen).

Zur Motivation und weitere Beispiele:

- $F_0 = F_1 = 1, F_{n+2} - F_{n+1} - F_n = 0$ erzeugt die Fibonacci-Zahlenfolge,
- $A_0 = 1, A_{n+1} - (n+1)A_n = 0$ erzeugt $A_n = n!$,
- $T_0 = 0, T_{n+1} - 2T_n = 1$ gibt die Komplexität bei den Türmen von Hanoi,
- $V_0 = 2, V_{n+1} - V_n = 2n$ zählt die Anzahl der Teile, die durch n Venn-Diagramme erzeugt werden,
- $C_0 = 1, C_1 = 3, C_{n+2} - 4C_{n+1} + 4C_n = 1$ (Cullen-Zahlen),
- $A_0 = 0, A_1 = 1, A_{n+2} - 5A_{n+1} + 6A_n = 0$,
- $F_{0,k} = 0, F_{1,k} = 1, F_{n+2,k} - kF_{n+1,k} - F_{n,k} = 0$ (k -Fibonacci-Zahlen),
- $T_0 = T_1 = 0, T_2 = 1, T_{n+3} - T_{n+2} - T_{n+1} - T_n = 0$ (Tribonacci-Zahlen).

Eine Gleichung der Form $A_{n+k} + a_1 A_{n+k-1} + \dots + a_k A_n = f(n)$ heißt eine lineare Rekursionsgleichung der Ordnung k (oder k -ter Ordnung). Eine Lösung der Gleichung ist eine Folge (A_n) , welche die Gleichung für alle natürlichen Zahlen n erfüllt. Sind Werte $A_0, \dots, A_{k-1}$ gegeben und ist $f = 0$, ist die Folge offenbar eindeutig bestimmt. Es ist häufig von (vor allem theoretischen) Interesse, eine explizite Formel für A_n zu haben. Was das ist, wie man sie erhält und begründet, davon handelt diese Einheit.

Weitere Sprechweisen: Sind alle a_i konstant, spricht man von einer linearen Rekursionsgleichung mit konstanten Koeffizienten. Ist $f = 0$ so heißt die Gleichung homogen, andernfalls inhomogen. k nennt man die Ordnung der Rekursionsgleichung.

Wir starten mit den Fibonacci-Zahlen. Sie beschreiben z.B. die Anzahl der Kaninchenpaare nach n Schritten; sie treten aber auch als Worst-case-Fall beim euklidischen Algorithmus auf. Bekanntlich gilt (die sogenannte Binet-Formel)

$$F_n = \frac{1}{\sqrt{5}} \left[\left(\frac{1 + \sqrt{5}}{2} \right)^n - \left(\frac{1 - \sqrt{5}}{2} \right)^n \right]$$

für alle $n \geq 0$. Die Zahl $\phi = (1 + \sqrt{5})/2$ heißt der Goldene Schnitt; er erfüllt $(1 + \phi) : \phi = \phi : 1$ bzw.

$\phi^2 - \phi - 1 = 0$. Mit dieser Abkürzung kann die Formel auch so geschrieben werden:

$$F_n = \frac{\phi^n - (1 - \phi)^n}{\sqrt{5}}.$$

Führe die [Arbeitsaufträge 1 bis 4](#) aus.

Wie lässt sich diese Formel herleiten? Wir stellen zwei Methoden vor:

1. Wir interpretieren die Folge als Funktion $f(n) = F_n$ und versuchen es mit dem Ansatz $f(n) = \alpha^n$. Es folgt $\alpha = \phi$ bzw. $\alpha = 1 - \phi$. Offenbar ist auch $f(n) = a\phi^n + b(1 - \phi)^n$ eine Lösung. Mit Hilfe der Anfangswerte folgt $a = -b = 1/\sqrt{5}$ und somit die obige Formel.

2. Wir packen die gesamte Folge in eine Funktion (genannt erzeugende Funktion) und definieren

$$f(x) = \sum_{n=0}^{\infty} F_n x^n.$$

Aus der Rekursionsgleichung folgt durch Multiplikation mit x^n und durch Aufsummieren über alle n , unter Zuhilfenahme der Anfangswerte $F_0 = F_1 = 1$, dass

$$f(x) = \frac{x}{1 - x - x^2}.$$

Diese Funktion lässt sich mit Hilfe der Partialbruchzerlegung wie folgt darstellen

$$f(x) = x \left(\frac{a}{1 - \phi x} + \frac{b}{1 - (1 - \phi)x} \right),$$

wobei $a = \phi/\sqrt{5}, b = -(1 - \phi)/\sqrt{5}$ sind (warum? Koeffizientenvergleich!). Mit Hilfe der geometrischen Reihe $1/(1 - x) = 1 + x + x^2 + \dots$ folgt wieder die gewünscht Darstellung.

Mit Hilfe derselben Methode (wirklich?, was tut man mit f ?), lassen sich auch in den anderen Beispielen explizite Formeln angeben. Bearbeite die [Arbeitsaufträge 5 bis 10](#).

Die explizite Formel (als polynomiell-exponentieller Ausdruck, z.B. für $f = 0$) ist zudem eindeutig; die Begründung führt auf die Frage, ob/wie oft/wann $A_n = 0$ gilt. Die Antwort liefert der berühmte Satz von Skolem-Mahler-Lech (alle Lösungen n liegen in einer endlichen Menge und endlich vielen arithmetischen Folgen $an + b$); die Begründung verwendet p -adische Analysis und ist nicht effektiv (berechenbar). Am sogenannten Skolem-Problem (gib einen Algorithmus, der für eine gegebene Folge (A_n) entscheidet, ob ein solches n existiert) wird zur Zeit sehr intensiv geforscht. Bearbeite die [Arbeitsaufträge 11 und 12](#).

Bearbeite die [Arbeitsaufträge 13 und 14](#).

Literatur:

1. A. Markovic: Lineare Rekursionen, Bachelorarbeit, Universität Salzburg, 2023
2. C. Fuchs: Kombinatorik, Vorlesungsskript, TU Graz, 2002

Rekursionen - Wie, was, warum?

Handout

Univ.-Prof. Dr. C. Fuchs
22.04.2025

Grundsätzlich: Eine lineare Rekursionsgleichung hat die Form $A_{n+k} + a_1 A_{n+k-1} + \dots + a_k A_n = f(n)$ für $n \geq 0$. Wir beschäftigen uns mit ihrer Lösung (A_n) sowie der Begründung und Herleitung von expliziten Darstellungen für das n -te Glied A_n der Lösungsfolge.

Technologieeinsatz mit Sage:

Ein nützliches Sage-Paket `rec_sequences`

Einige nützliche Sage-Befehle: `LinearSolveSequence`, `FunkctionalEquation`

Programm in Stichworten:

Existenz und Eindeutigkeit von Lösungen einer Rekursionsgleichung mit Anfangswerten, Binet-Darstellung (Begründung und Herleitung mit Hilfe des charakteristischen Polynoms und mit erzeugenden Funktionen).

Arbeitsaufträge:

1. Berechne F_{100} mit Hilfe der Rekursion sowie mit Hilfe der Binet-Darstellung.
2. Experimentiere mit dem Fehler bei Näherung von F_n durch $\phi^n / \sqrt{5}$.
3. Gib basierend auf der Binet-Darstellung eine Näherungsformel für F_n an.
4. Zeige, dass F_{n+1}/F_n gegen ϕ konvergiert.
5. Durch sogenannte Venn-Diagramme werden Mengen graphisch dargestellt. Jede Menge wird durch ein Oval ersetzt. Dabei ergibt sich folgendes kombinatorische Problem: Gegeben seien n Ovale. In wie viele Teile zerlegen sie die Ebene maximal? Bestimme eine Rekursionsformel für die Anzahl V_n der Teile.
6. Bestimme eine explizite Formel für V_n in der vorigen Aufgabe mit der Ansatzmethode.
7. Bestimme eine explizite Formel für V_n in der vorigen Aufgabe mit erzeugenden Funktionen.
8. Bestimme eine explizite Formel für die Laufzeitanalyse der Türme von Hanoi.
9. Bestimme eine explizite Formel für die Cullen-Zahlen.
10. Bestimme eine explizite Formel für $A_0 = 0, A_1 = 1, A_{n+2} = 5A_{n+1} - 6A_n$. Berechne A_{100} .
11. Gibt es Folgenglieder 0 in den obigen Beispielen? Welche? Warum nicht (noch mehr)?
12. Gib ein Beispiel für eine Rekursion, die unendlich viele 0-Einträge hat.
13. Bestimme eine explizite Formel für die k -Fibonacci-Zahlen.
14. Bestimme eine explizite Formel für die Tribonacci-Zahlen.



Was ist eine lineare Rekursionsgleichung (z.B. Fibonacci-Zahlen)?

Wie lautet die explizite Formel und wie wird sie hergeleitet (z.B. Binet-Formel für Fibonacci-Zahlen)?

Das hat mir gefallen. Das hat mir nicht gefallen.