Datenkompression (Rote), SS 2003, 3. Übungsblatt

Abgabe: bis Freitag, 9.5.2003, in der Vorlesung; Programmieraufgabe bis 23.5.2003.

12. (0 Punkte) Die Länge eines Kodeworts für ein Symbol mit Wahrscheinlichkeit p_i bein Shannon-Kode ist $\lceil -\log_2 p_i \rceil$. Kann es sein, dass dieses Symbol bei einem optimalen Kode eine Länge größer als $\lceil -\log_2 p_i \rceil$ hat?

13. Der Golomb-Kode (10 Punkte)

    Bestimmen Sie einen optimalen Kode für die *geometrische Verteilung*

    $$p_i = p \cdot (1-p)^i, \text{ für } i = 0, 1, 2, \ldots$$

    für den Parameter (a) $p = 0{,}6$ (b) $p = 0{,}5$ (c) $p = 0{,}125$. (p_i ist die Wahrscheinlichkeit, dass man bei der Wiederholung eines Experiments, das Erfolgswahrscheinlichkeit p hat, i Misserfolge hat, bevor das Experiment das erste Mal gelingt.)

    (Achtung: Da es unendlich viele Blätter gibt, kann man den Huffman-Algorithmus nicht ohne Weiteres anwenden.)

    Zusatzfrage zu Aufgabe (c): Wenn man eine zufällige 0-1-Folge in Dreierblöcken liest, dann ist p_i die Wahrscheinlichkeit, dass man i Blöcke lesen muss, bevor man einen Block 000 trifft. Man kann dementsprechend einen Kode mit folgender Dekodierungsregel verwenden:

    > Lies die kodierte Nachricht in Dreierblöcken bis zu einem 000-Block. Das dekodierte Symbol i ist die Anzahl der gelesenen Blöcke minus 1.

    Warum ist der Kode, der durch diese Regel dekodiert wird, kein (guter) Kode für die Aufgabe?

14. (0 Punkte) Unter welchen Bedingungen an die Wahrscheinlichkeiten $p_1, \ldots, p_n$ haben alle Kodewörter im Huffman-Kode die gleiche Länge?

15. (5 Zusatzpunkte) Beim Huffman-Algorithmus kann es vorkommen, dass mehrere Knoten das gleiche Gewicht haben und die Auswahl des nächsten Schrittes nicht eindeutig ist. Wie kann man in diesem Fall die Auswahl der beiden nächsten Knoten zum Vereinigen steuern, sodass man möglichst keine langen Kodewörter erhält? (Genau gesagt: damit das längste Kodewort möglichst kurz ist.)

16. (0 Punkte) Versuchen Sie eine Verteilung zu finden, bei der das längste Kodewort in einem optimalen Kode mindestens Länge L hat und die kleinste Wahrscheinlichkeit möglichst groß ist, für verschiedene Werte von L.

17. (25 Punkte, Abgabe bis Freitag 23. Mai, in der Vorlesung) Schreiben Sie ein Programm, das eine Datei einliest und als Folge von Bytes mit dem Huffman-Algorithmus komprimiert. Wählen Sie eine der beiden folgenden Möglichkeiten:

    (a) Die Häufigkeiten werden in einem ersten Durchlauf bestimmt; der Huffman-Kode wird am Anfang der komprimierten Datei gespeichert. (Sie können dabei auf Aufgabe 11 zurückgreifen.)

    (b) Verwenden Sie einen adaptiven Huffman-Code.