Randomisierte Algorithmen SS 2000

4. Übungsblatt

22. Sei $n = 2m$ eine gerade natürliche Zahl. Betrachten Sie das Permutation-Routing-Problem (PRP) für den n-dimensionalen Hyperwürfel. Für jeden Startprozessor i mit binärer Darstellung $(\underbrace{i_1, i_2, \ldots, i_m}_{a} \underbrace{i_{m+1}, i_{m+2}, \ldots, i_{2m}}_{b})$ sei der Zielprozessor $d(i)$ durch die binäre Darstellung (b, a) gegeben. Zeigen Sie, daß der Bit-Fixierungsalgorithmus $\Omega(\sqrt{N/n})$ Schritte benötigt, um diese Instanz des PRP zu lösen, wobei $N := 2^n$.

23. Betrachten Sie den in der Vorlesung beschriebenen 2-Phasen Algorithmus für das PRP im n-dimensionalen Hyperwürfel. Zeigen Sie, daß die erwartete Anzahl der von diesem Algorithmus benötigten Schritte weniger als $15n$ ist. (Im synchronen Modell entspricht ein Schritt dem Befördern eines Pakets entlang einer Kante.)

24. Betrachten Sie einen unfairen Münzenwurf bei dem Kopf mit Wahrscheinlichkeit $p \geq a$ fällt, wobei a eine positive Konstante bzw. p ein unbestimmter Parameter ist. Entwerfen Sie eine Prozedur zur Approximierung des p-Wertes durch $\tilde{p}$, sodaß für jede Realisierung der Konstanten $0 < a, \epsilon, \delta < 1$ die Ungleichung $Pr[|p - \tilde{p}| > \epsilon p] < \delta$ gilt. Sei N die Anzahl der Münzwürfe, die für die Bestimmung von $\tilde{p}$ benötigt werden. Geben Sie eine nicht-triviale untere Schranke für N.

25. Seien X_1,X_2,..., X_n unabhängige geometrisch verteilte Zufallsvariablen mit Erwartungswert jeweils 2. (X_i sei die Anzahl der Münzwürfe bis zum erstmaligen Auftreten von Kopf bei einer Folge von unabhängigen Münzwürfen, in denen Kopf mit jeweils einer fixen Wahrscheinlichkeit p fällt. Siehe auch Übungsbeispiel 3.) Sei $X = \sum_{i=1}^{n} X_i$ und sei δ eine positive reelle Konstante. Verwenden Sie moment-generierende Funktionen und die Chernoff-Technik um eine nicht-triviale obere Schranke für $Pr[X > (1 + \delta)2n]$ zu ermitteln.

26. Betrachten Sie einen BPP Algorithmus A (BPP=Bounded Probability Polynomial, siehe Vorlesung) mit Fehlerwahrscheinlichkeit von $1/2 - 1/p(n)$, wobei $p(n)$ ein von der Inputgröße n abhängiges Polynom ist. Verwenden Sie die Chernoff-Schranke für die Binomialverteilung, um zu zeigen, daß eine polynomiale Anzahl von Wiederholungen von A ausreicht, um die Fehlerwahrscheinlichkeit auf $1/2^n$ zu reduzieren.

27. Sei G ein Graph mit n Knoten und $nd/2$ Kanten, wobei d eine positive natürliche Konstante ist. Betrachten Sie das folgende probabilistische Experiment zur Bestimmung einer unabhängigen Menge in G. (Eine unabhängige Menge in G ist eine Menge von Knoten $I \subseteq V$ sodaß es keine Kanten zwischen den Knoten von I gibt.) Unabhängig von einander lösche jeden Knoten $v \in V$ (und alle mit v inzidierenden Kanten) mit Wahrscheinlichkeit $1 - 1/d$.

    (a) Berechnen Sie die erwartete Anzahl von Knoten und die erwartete Anzahl von Kanten, die am Ende des Experiments in G übrig sind.

    (b) Verwenden Sie das Ergebnis von (a), um zu zeigen, daß es in jedem Graphen mit n Knoten und $nd/2$ Kanten eine unabhängige Menge mit mindestens $n/2d$ Knoten gibt.

    (c) Sei G ein 3-regulärer Graph (ein Graph, in dem alle Knoten Grad 3 haben). Aus dem obigen probabilistischen Experiment bilden wir den folgenden randomisierten Algorithmus: Jeder der Knoten wird mit Wahrscheinlichkeit $2/3$ (unabhängig von den anderen Knoten) gelöscht. Für jede übriggebliebene Kante löschen wir einen ihrer Endknoten. Die am Ende dieses Prozesses übriggebliebenen Knoten bilden eine unabhängige Menge in G. Ermitteln Sie eine nicht-triviale obere Schranke für die Wahrscheinlichkeit, daß dieser Algorithmus eine unabhängige Menge mit weniger als $n(1 - \epsilon)/6$ Knoten erzeugt, wobei ϵ eine Konstante mit $0 < \epsilon < 1$ ist.

28. **Das Mengen-Überdeckungsproblem (the set cover problem).**
Gegeben sind n Teilmengen $S_1, S_2, \ldots, S_n$ einer Grundmenge U. Gesucht wird eine Menge $T \subseteq U$ mit minimaler Kardinalität für die $S_i \cap T \neq \emptyset$ für alle $1 \leq i \leq n$ gilt. Eine alternative Formulierung des Problems ist die folgende: Für eine gegebene 0-1 Matrix $M = (m_{ij})$, finde einen 0-1 Vektor x, sodaß die Ungleichung $\sum_j m_{ij} x_j \geq 1$ für alle Zeilen i der Matrix M gilt und $\sum_j x_j$ minimiert wird. Ein Vektor x, der die Ungleichung $\sum_j m_{ij} x_j \geq 1$ für alle Zeilen i der Matrix M erfüllt, heißt eine Überdeckung von M. Die Summe $\sum_j x_j$ wird als Größe der Überdeckung bezeichnet. Die Anzahl n von Mengen entspricht der Anzahl der Zeilen in M und die i-te Zeile in M entspricht dem Inzidenzvektor von S_i.

Sei $c(M)$ die Größe der kleinsten Überdeckung von M und n die Anzahl der Zeilen in M. Verwenden Sie zuerst lineares Programmieren und dann randomisiertes Runden um einen $O(\log(n))$-randomisierten Monte Carlo Algorithmus für das Mengen-Überdeckungsproblem in M zu entwickeln.