Karlsruher Institut für Technologie

Stefan Walzer
ITI Prof. Sanders
Wintersemester 2023/2024

Übungsblatt 4 – Probability Amplification

Randomisierte Algorithmik

Aufgabe 1 – Probability Amplification mit zweiseitigem Fehler

Angenommen ein Monte-Carlo Algorithmus A beantwortet ein Entscheidungsproblem mit Wahrscheinlichkeit $1/2 + \varepsilon$ korrekt und sonst falsch (für ein $\varepsilon > 0$). Sei A' der Algorithmus der t mal unabhängig A ausführt und sich für die häufigere Antwort entscheidet. Zeige, dass die Fehlerwahrscheinlichkeit von A' höchstens $e^{-2t\varepsilon^2}$ ist.

Hinweise: Für die Rechnung könnte nützlich sein: $\sum_{i=0}^{t} \binom{t}{i} = 2^t$.

Aufgabe 2 – Pfadfinder

Gegeben sei ein ungerichteter Graph G mit n Knoten und m Kanten. Gesucht ist ein Pfad der Länge k, der keinen Knoten mehrfach besucht. Der naive Brute-Force Ansatz hat eine Laufzeit von $O(n^k)$. Wir betrachten einen einfachen, randomisierten Ansatz, der wie folgt funktioniert. Im ersten Schritt wird jedem Knoten v ein Label $L(v)$ zufällig gleichverteilt aus $\{1, \ldots, k\}$ zugewiesen. Im zweiten Schritt wird von jedem Knoten v mit $L(v) = 1$ eine modifizierte Breitensuche gestartet, bei der ein Knoten w nur dann von einem Knoten u entdeckt werden kann, falls $L(u) = L(w) + 1$ gilt. Wird ein Knoten mit Label k entdeckt wird ein Pfad der Länge k konstruiert und ausgegeben.

(a) Zeige, dass der Algorithmus mit Wahrscheinlichkeit $1/k^k$ einen Pfad der Länge k findet, falls einer existiert.

(b) Verbessere die Erfolgswahrscheinlichkeit auf $1 - 1/n$ mit Probability Amplification. Welche Gesamtlaufzeit ergibt sich?

Bemerkung: Die Idee heißt “Color Coding”. Es gibt noch raffiniertere Varianten.

Aufgabe 3 – Bonus: Random-Walk Löser für 3-SAT

Sei $\varphi(x_1, \ldots, x_n)$ eine 3-SAT Formel mit n Variablen und m Klauseln. Wir suchen eine Lösung mit folgendem Algorithmus:

Algorithm randomWalkSolver(φ):

```
sample x_1, ..., x_n ~ Ber(1/2)
for k = 1 to n/2 do
    if φ(x_1, ..., x_n) = 1) then
        break
    sei C eine beliebige unerfüllte Klausel von φ
    sample j ~ U({1, 2, 3})
    sei x_i die jte Variable in C
    x_i ← 1 − x_i
if φ(x_1, ..., x_n) = 1) then
    return (x_1, ..., x_n)
return ⊥
```

Falls φ unerfüllbar ist, gilt offenbar randomWalkSolver$(\varphi) = \perp$. Andernfalls sei x^* eine Lösung. Zeige:

(a) Mit Wahrscheinlichkeit $\geq 1/2$ stimmt die initial gesampelte Belegung $(x_1, \ldots, x_n)$ an mindestens $n/2$ Stellen mit x^* überein.

(b) Falls $\varphi(x_1, \ldots, x_n) = 0$ so führt eine Iteration der Schleife mit Wahrscheinlichkeit $1/3$ dazu, dass eine weitere Variable von $(x_1, \ldots, x_n)$ so wie in x^* gesetzt wird.

(c) Verwende Probability Amplification um einen Algorithmus mit Erfolgswahrscheinlichkeit $1 - 1/n$ zu erhalten. Was ist die Gesamtlaufzeit?