Randomisierte Algorithmen

7. Random Walks

Thomas Worsch

Fakultät für Informatik
Karlsruher Institut für Technologie

Wintersemester 2019/2020

Überblick

Ein randomisierter Algorithmus für 2-SAT

Random Walks

Widerstandsnetzwerke

Randomisierte Algorithmen für Zusammenhangstests

Überblick

Problem 2SAT

„viel einfacher“ als 3SAT

- aussagenlogische Formeln
  - n Variablen (*nicht:* Vorkommen von Literalen)
  - in konjunktiver Normalform, wobei
  - jede Klausel zwei Literale enthält
- Probleminstanz: eine solche Formel F
- Frage: Ist F erfüllbar?
- Problem ist in $\mathbf{P}$
  - $a \vee b$ ist äquivalent zu $(\bar{a} \Longrightarrow b) \wedge (\bar{b} \Longrightarrow a)$
  - reflexiv transitive Hülle
  - F nicht erfüllbar gdw. $(\bar{x} \Longrightarrow^* x) \wedge (x \Longrightarrow^* \bar{x})$ für eine Variable x
  - mit anderem Algorithmus sogar in Linearzeit lösbar
- Das Problem ist $\mathbf{NL}$-vollständig.

7.1 Randomisierter Algorithmus (für 2SAT)

ein guter Einstieg in die folgenden Kapitel

Sei zunächst F eine erfüllbare 2SAT-Formel

7.1 Randomisierter Algorithmus (für 2SAT)

ein guter Einstieg in die folgenden Kapitel

Sei zunächst F eine erfüllbare 2SAT-Formel

$B \leftarrow \langle \textit{zufällige Belegung aller } x_1, \ldots, x_n \textit{ mit Werten} \rangle$

7.1 Randomisierter Algorithmus (für 2SAT)

ein guter Einstieg in die folgenden Kapitel

Sei zunächst F eine erfüllbare 2SAT-Formel

$B \leftarrow \langle$*zufällige Belegung aller* $x_1, \ldots, x_n$ *mit Werten*$\rangle$
while $F(B) = \mathit{false}$ **do**

od

7.1 Randomisierter Algorithmus (für 2SAT)

ein guter Einstieg in die folgenden Kapitel

Sei zunächst F eine erfüllbare 2SAT-Formel

$B \leftarrow \langle \textit{zufällige Belegung aller } x_1, \ldots, x_n \textit{ mit Werten} \rangle$
while $F(B) = \textit{false}$ **do**

$\quad B(x_i) \leftarrow$ **not** $B(x_i)$
od

7.1 Randomisierter Algorithmus (für 2SAT)

ein guter Einstieg in die folgenden Kapitel

Sei zunächst F eine erfüllbare 2SAT-Formel

$B \leftarrow \langle$*zufällige Belegung aller* $x_1, \ldots, x_n$ *mit Werten*$\rangle$
while $F(B) = \mathit{false}$ **do**
  $k \leftarrow \langle$*von* B *nicht erfüllte Klausel in* $F\rangle$
  $x_i \leftarrow \langle$*zufällig gewählte Variable in* $k\rangle$
  $B(x_i) \leftarrow$ **not** $B(x_i)$
od

7.2 Analyse des Algorithmus

- Es sei A eine Variablenbelegung, die F erfüllt.
  Es bezeichne j die Anzahl Variablen x mit $B(x) = A(x)$.
- Betrachte linearen Pfad P_{n+1} mit möglichen Werten $0, 1, \ldots, n$ für j als Knoten.
  $\Longrightarrow$ Belegung B entspricht einem Knoten.
- Bei jedem Schleifendurchlauf:
  - j wird um 1 erhöht oder erniedrigt.
  - Veränderung von B entspricht dem Schritt zu einem der beiden Nachbarknoten.
- Algorithmus beginnt „schlimmstenfalls“ bei $j = 0$ und endet spätestens dann, wenn zum ersten Mal $j = n$ erreicht wird.

7.3 Fortsetzung der Analyse

- Erhöhung der Knotennummer
  mit Wahrscheinlichkeit $\geq 1/2$
- Laufzeit des Algorithmus: Anzahl Schritte
  - vom Startknoten bis zum Finden einer erfüllenden Belegung
- Abschätzung nach oben:
  Anzahl Schritte von Knoten 0 zu Knoten n.
- Typische Fragestellung bei **Random Walks**: Was ist der Erwartungswert für die Anzahl Schritte, um von einem bestimmten Startknoten zu einem bestimmten Zielknoten zu gelangen?

7.4 Modifizierter Algorithmus

Wir werden zeigen:

- Der Erwartungswert für Anzahl Schritte von Knoten 0 nach Knoten n entlang eines einzelnen Pfades ist $\leq n^2$.
- Markov-Ungleichung: Wahrscheinlichkeit, dass Random Walk der Länge $2n^2$ *nicht* zum Ziel führt, ist $\leq 1/2$.

Deswegen …

7.4 Modifizierter Algorithmus (2)

$B \leftarrow \langle \textit{zufällige Belegung aller } x_1, \ldots, x_n \textit{ mit Werten} \rangle$
$m \leftarrow 0 \ \langle \textit{Zähler für die Anzahl Versuche} \rangle$
while $F(B) = \mathit{false}$ **and** $m < 2n^2$ **do**
    $k \leftarrow \langle \textit{von } B \textit{ nicht erfüllte Klausel in } F \rangle$
    $x_i \leftarrow \langle \textit{zufällig gewählte Variable in } k \rangle$
    $B(x_i) \leftarrow$ **not** $B(x_i)$
    $m \leftarrow m + 1$
od
if $F(B) = \mathit{true}$ **then**
    return $\langle F \textit{ erfüllbar durch } B \rangle$
else
    return $\langle F \textit{ nicht erfüllbar} \rangle$
fi

Überblick

7.5 Einfache Random Walks

- $G = (V, E)$ endlicher, ungerichteter, zusammenhängender Graph mit $|V| = n \geq 2$ Knoten und $|E| = m$ Kanten
- Für $u \in V$ bezeichne $\Gamma(u) = \{v \mid (u, v) \in E\}$ die Menge der *Nachbarn*, so dass $d(u) = |\Gamma(u)|$ der Grad von u ist.
- Vorstellung: ein Teilchen, Objekt, … läuft auf dem Graphen herum:
  - zu jedem Zeitpunkt an einem Knoten des Graphen
  - ein Schritt: Bewegung über zufällig gewählte Kante zu einem Nachbarknoten
- *einfacher* Random Walk:
  jede Kante mit gleicher Wahrscheinlichkeit $1/d(u)$ gewählt

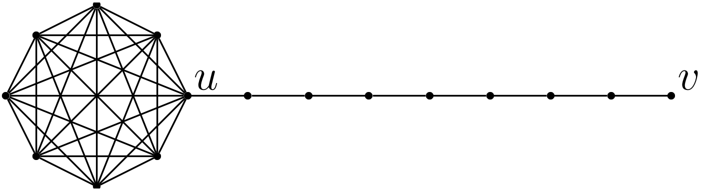
7.6 Definition

- *m_{uv}:* Erwartungswert für die Anzahl Schritte, um bei einem in u startenden Random Walk erstmals zu Knoten v zu gelangen.
- *Wechselzeit C_{uv}* $= m_{uv} + m_{vu}$

7.7 Beispiel

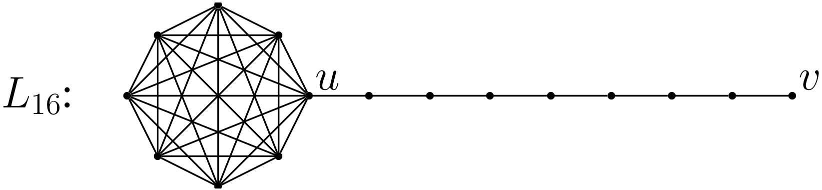
„Lollipop-Graph“ $L_n = (\{1, \ldots, n\}, E)$:
Clique der Größe $n/2$ mit „angeklebtem Pfad“ der Länge $n/2$

L_{16}:



7.7 Beispiel

„Lollipop-Graph“ $L_n = (\{1, \ldots, n\}, E)$:
Clique der Größe $n/2$ mit „angeklebtem Pfad“ der Länge $n/2$



Wir werden sehen, dass für diese Beispielgraphen gilt:

$$m_{uv} \in \Theta(n^3) \qquad \text{aber} \qquad m_{vu} \in \Theta(n^2)$$

Überblick

7.8 Widerstandsnetzwerke

- ungerichteter zshg. Graph G ohne Schlingen
  $\longrightarrow$ Widerstands-Netzwerk $N(G)$:
  ersetze jede Kante von G durch einen Widerstand von 1 Ω
- Zwischen Knoten $u \neq v$ ergibt sich ein *effektiver Widerstand* R_{uv}.
- R_{uv} ist der Quotient U_{uv}/I_{uv} aus einer zwischen u und v angelegten Spannung und dem dann fließenden Strom.
- einfache Fälle:
  - Reihenschaltung von Widerständen R_k:
    Gesamtwiderstand $R = \sum_k R_k$
  - Parallelschaltung von Widerständen R_k:
    Gesamtwiderstand $R = 1/(\sum_k 1/R_k)$

7.9 Satz

Es sei G ein ungerichteter Graph mit m Kanten.
Dann gilt für alle Knoten u und v in G:

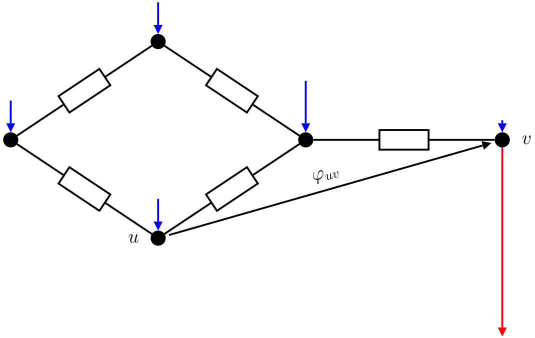
$$C_{uv} = 2m \cdot R_{uv}$$

Beweis: über „elektrische Charakterisierung“ von m_{uv}

7.10 Lemma

Sei φ_{uv} die Spannung bei u relativ zu v, wenn

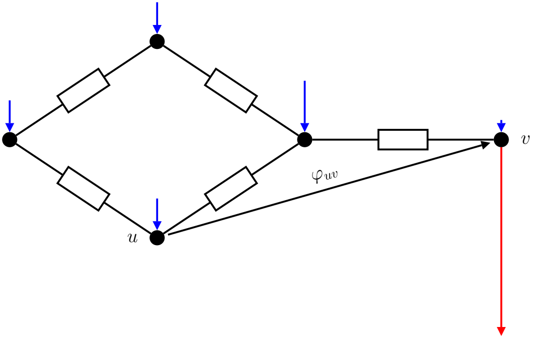
- jedem Knoten x Strom $d(x)$ Ampere injiziert wird und
- der Gesamtstrom von $2m$ Ampere bei v abgeführt wird.



7.10 Lemma

Sei φ_{uv} die Spannung bei u relativ zu v, wenn

- jedem Knoten x Strom $d(x)$ Ampere injiziert wird und
- der Gesamtstrom von $2m$ Ampere bei v abgeführt wird.



Dann ist $m_{uv} = \varphi_{uv}$.

7.11 Beweis

- von $u \neq v$ zu $w \in \Gamma(u)$ fließender Strom ist $\varphi_{uv} - \varphi_{wv}$
- Kirchhoffsche Regel: an jedem Knoten u ist zufließender Strom = abfließender Strom
- $$d(u) = \sum_{w \in \Gamma(u)} (\varphi_{uv} - \varphi_{wv})$$

  bzw. $$d(u) + \sum_{w \in \Gamma(u)} \varphi_{wv} = d(u)\varphi_{uv} \, .$$

7.11 Beweis

- von $u \neq v$ zu $w \in \Gamma(u)$ fließender Strom ist $\varphi_{uv} - \varphi_{wv}$
- Kirchhoffsche Regel: an jedem Knoten u ist zufließender Strom = abfließender Strom
- $$d(u) = \sum_{w \in \Gamma(u)} (\varphi_{uv} - \varphi_{wv})$$

  bzw. $$d(u) + \sum_{w \in \Gamma(u)} \varphi_{wv} = d(u)\varphi_{uv}\ .$$
- Linearität der Erwartungswerte liefert für $u \in V \setminus \{v\}$:

  $$m_{uv} = \frac{1}{d(u)} \sum_{w \in \Gamma(u)} (1 + m_{wv})$$

  bzw. $$d(u) + \sum_{w \in \Gamma(u)} m_{wv} = d(u)m_{uv}\ .$$

7.11 Beweis

- von $u \neq v$ zu $w \in \Gamma(u)$ fließender Strom ist $\varphi_{uv} - \varphi_{wv}$
- Kirchhoffsche Regel: an jedem Knoten u ist zufließender Strom = abfließender Strom
- $$d(u) = \sum_{w \in \Gamma(u)} (\varphi_{uv} - \varphi_{wv})$$

  bzw. $$d(u) + \sum_{w \in \Gamma(u)} \varphi_{wv} = d(u)\varphi_{uv} \,.$$
- Linearität der Erwartungswerte liefert für $u \in V \setminus \{v\}$:

  $$m_{uv} = \frac{1}{d(u)} \sum_{w \in \Gamma(u)} (1 + m_{wv})$$

  bzw. $$d(u) + \sum_{w \in \Gamma(u)} m_{wv} = d(u)m_{uv} \,.$$

7.11 Beweis (2)

- Zweimal das gleiche lineare Gleichungssystem,
- das offensichtlich lösbar ist
- gleich: Lösung eindeutig
- Also ist $\varphi_{uv} = m_{uv}$.

7.11 Beweis (3)

Eindeutigkeit der Lösung

- Es seien φ_{uv} und ψ_{uv} zwei Lösungen, also

$$\forall v\ \forall u \neq v : d(u)\varphi_{uv} = d(u) + \sum_{w\in\Gamma(u)} \varphi_{wv} \text{ und}$$

$$\forall v\ \forall u \neq v : d(u)\psi_{uv} = d(u) + \sum_{w\in\Gamma(u)} \psi_{wv}$$

- also für jedes Paar (u, v)

$$\varphi_{uv} - \psi_{uv} = \frac{1}{|\Gamma(u)|} \sum_{w\in\Gamma(u)} (\varphi_{wv} - \psi_{wv})$$

7.11 Beweis (4)

Eindeutigkeit der Lösung, Fortsetzung

$$\varphi_{uv} - \psi_{uv} = \frac{1}{|\Gamma(u)|} \sum_{w \in \Gamma(u)} (\varphi_{wv} - \psi_{wv})$$

- fixiere beliebiges v und
  wähle u so, dass $\varphi_{uv} - \psi_{uv}$ minimal wird
- also gilt für alle $w \in \Gamma(u)$:

$$\varphi_{uv} - \psi_{uv} = \varphi_{wv} - \psi_{wv}$$

- die Minimalität von $\varphi_{xv} - \psi_{xv}$ „vererbt“ sich
  von jedem Knoten x zu seinen Nachbarn
- also ist für jedes x die Differenz $\varphi_{xv} - \psi_{xv}$ gleich (minimal)

7.11 Beweis (5)

Eindeutigkeit der Lösung, Fortsetzung:

- für alle $w \in \Gamma(u)$:

$$\varphi_{uv} - \psi_{uv} = \varphi_{wv} - \psi_{wv}$$

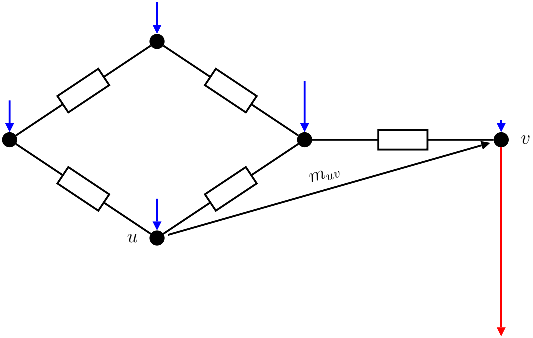
- betrachte beliebige Kante uw:

$$\begin{aligned}
\varphi_{uw} - \psi_{uw} &= (\varphi_{uv} - \varphi_{wv}) - (\psi_{uv} - \psi_{wv}) \\
&= (\varphi_{uv} - \psi_{uv}) - (\varphi_{wv} - \psi_{wv}) \\
&= 0
\end{aligned}$$

- also stimmen φ und ψ für alle Kanten überein
- also stimmen sie sogar für alle Knotenpaare überein (betrachte Pfade)

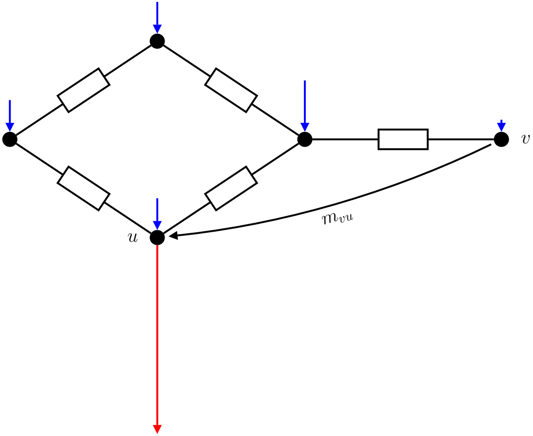
7.12 Beweis von Satz 7.9: $C_{uv} = 2m \cdot R_{uv}$

Wir wissen schon: $m_{uv} = \varphi_{uv}$.



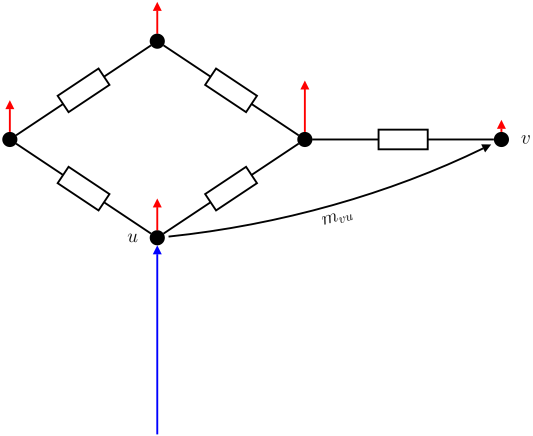
7.12 Beweis (2)

Analog: $m_{vu} = \varphi_{vu}$.



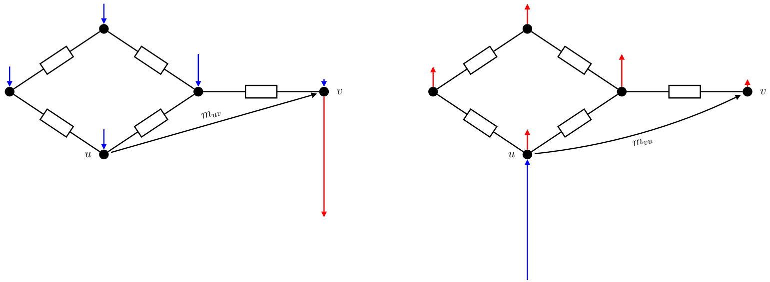
7.12 Beweis (3)

Vorzeichen umdrehen bei $m_{vu} = \varphi_{vu}$:



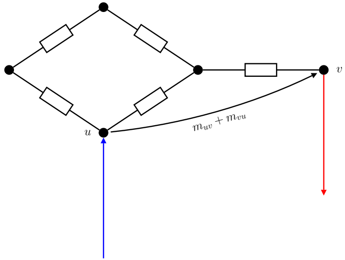
7.12 Beweis (4)

addieren (Widerstandsnetzwerke sind linear) von



7.12 Beweis (5)

Ergebnis:

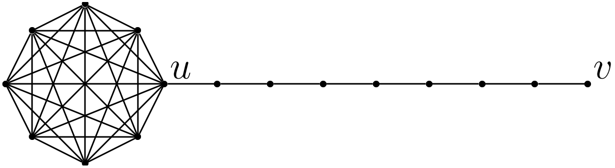


Nach dem Ohmschen Gesetz ist aber $\varphi_{uv} + \varphi_{vu}$ gerade $2mR_{uv}$.

7.13 Beispiel

Lollipop-Graph L_n:

L_{16}:

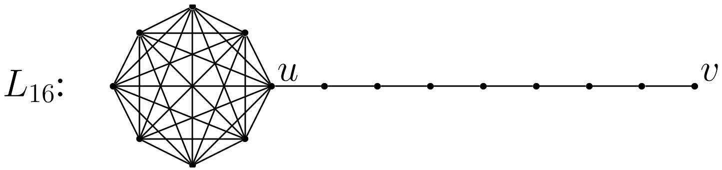


Bestimmung von m_{uv}:

- An den $n/2$ Clique-Knoten werden jeweils $\Theta(n/2)$ Ampere injiziert
- Also: Gesamtstrom $\Theta(n^2)$ Ampere
- Er fließt über u nach v und verursacht an allen $n/2$ Widerständen unterwegs jeweils einen Spannungsabfall von $\Theta(n^2)$ Volt
- Also Gesamtspannungsabfall $\Theta(n^3)$

7.13 Beispiel (2)

Lollipop-Graph L_n:



Bestimmung von m_{vu}:

- An den Knoten $1 + n/2, 2 + n/2, \ldots, n$ werden jeweils 2 Ampere (bzw. 1 bei n) injiziert, die zu u fließen.
- Also fließen durch Widerstand zwischen Knoten $n - i$ und $n - i - 1$ gerade $2(i + 1)$ Ampere (für $0 \leq i \leq n/2 + 1$).
- Die Spannungsabfälle summieren sich zu $\Theta(n^2)$.

7.14 Korollar

Für jeden Graphen mit n Knoten und beliebigen Knoten u und v gilt:

$$C_{uv} < n^3 \ .$$

7.15 Beweis

- $|V| = n \implies |E| = m \leq n(n-1)/2$
- Maximaler effektiver Widerstand R_{uv} nach oben beschränkt durch die Länge kürzester Wege von u nach v.
- Reihenschaltung ist der schlimmste Fall, d. h. $R_{uv} \leq n - 1$.
- Also ist $C_{uv} = 2mR_{uv} < n^3$.

Überblick

7.16 Problem

- USTCON: undirected s–t connectivity
- Gegeben: ungerichteter Graph und zwei Knoten s und t
- Frage: Sind s und t in der gleichen Zusammenhangskomponente, also durch einen Pfad miteinander verbunden?

Anmerkungen zu USTCON

- zunächst klar: USTCON liegt in $\mathbf{NL} = \mathrm{NSPACE}(\log n)$
- Savitch (1970) $\rightsquigarrow$ USTCON in $\mathrm{DSPACE}((\log n)^2)$
- Aleliunas (1979): USTCON randomisiert in poly. Zeit und $\log n$ Platz (kommt gleich)
- Lewis et al. (1982): USTCON ist $\mathbf{SL}$-vollständig ($\mathbf{L} \subseteq \mathbf{SL} \subseteq \mathbf{NL}$)
- Nisan et al. (1992): USTCON in $\mathrm{DSPACE}((\log n)^{1.5})$
- Reingold (2004): USTCON in $\mathrm{DSPACE}(\log n)$, also
  $$\mathbf{SL} = \mathbf{L} = \mathrm{DSPACE}(\log n)$$

7.17 Algorithmus

- simuliere einen Random Walk, der bei s startet

7.17 Algorithmus

- simuliere einen Random Walk, der bei s startet
- mache maximal $2n^3$ Schritte
  - falls man dabei auf Knoten t trifft: Antwort YES
  - falls man nie auf Knoten t trifft: Antwort NO

7.18 Lemma

Die Wahrscheinlichkeit, dass Algorithmus 7.17 fälschlicherweise NO ausgibt, ist höchstens $1/2$.

7.19 Beweis

- Falsche Antwort: s und t in der gleichen Zusammenhangskomponente, aber der Random Walk findet t nicht.
- Korollar 7.14: Erwartungswert $m_{st} \leq n^3$.
- Markov-Ungleichung: Wahrscheinlichkeit, dass ein Random Walk mehr als doppelt solange benötigt, um von s nach t zu gelangen, ist höchstens $1/2$.

7.20 Problem

- STCON: s–t connectivity
- Gegeben: gerichteter Graph und zwei Knoten s und t
- Frage: Sind s und t in der gleichen Zusammenhangskomponente, also durch einen Pfad miteinander verbunden?

- STCON ist vollständig für **NL**

7.20 Problem

- STCON: s–t connectivity
- Gegeben: gerichteter Graph und zwei Knoten s und t
- Frage: Sind s und t in der gleichen Zusammenhangskomponente, also durch einen Pfad miteinander verbunden?

- STCON ist vollständig für **NL**

NB: vorangegangener Algorithmus untauglich wegen möglicher „Einbahnstraßensackgassen"

7.21 Algorithmus

Abwechselnd die beiden folgenden Phasen:

1. Simuliere ausgehend von s einen Random Walk der Länge maximal $n-1$.
   Wird dabei t erreicht: Ausgabe YES.
2.

7.21 Algorithmus

Abwechselnd die beiden folgenden Phasen:

1. Simuliere ausgehend von s einen Random Walk der Länge maximal $n-1$.
   Wird dabei t erreicht: Ausgabe YES.
2. Es werden $\log n^n = n \log n$ Zufallsbits „gewürfelt“.
   Wenn sie alle 0 sind: Ausgabe NO.

7.22 Lemma

a) Algorithmus 7.21 kann so implementiert werden, dass der Platzbedarf kleiner gleich $O(\log n)$ ist.

b) Wenn kein Pfad von s nach t existiert: Ausgabe nie YES, also stets NO, sofern Algorithmus hält.

c) Wenn ein Pfad von s nach t existiert: Ausgabe YES mit Wahrscheinlichkeit größer gleich $1/2$.

7.23 Beweis

a) Platzbedarf:
   - Phase 1: bisherige Länge des Random Walk und aktuelle Knotennummer können in $O(\log n)$ Bits gespeichert werden.
   - Phase 2: zähle, wieviele Bits schon gewürfelt wurden und ob alle gleich 0 waren. Es genügen $\log(n \log n) \in O(\log n)$ Bits.

b) Wenn kein Pfad existiert, wird kein Random Walk von s nach t führen, also wird auch nie YES ausgegeben.

7.23 Beweis (2)

c) Falls Pfad von s nach t existiert:

- insgesamt höchstens n^n Pfade der Länge $n-1$
- also Wahrscheinlichkeit, dass in Phase 1 ein Pfad $s \rightsquigarrow t$ gefunden wird, mindestens n^{-n}.
- Die Wahrscheinlichkeit, dass in Phase 2 die falsche Antwort gegeben wird, ist $\leq (1-n^{-n})n^{-n} \leq n^{-n}$.
- X: Zufallsvariable, die angibt, in welchem Durchlauf die richtige Antwort gegeben wird
- p: Wahrscheinlichkeit, dass überhaupt die richtige Antwort gegeben wird. Dann ist

$$\begin{aligned} p = \mathbf{P}(X \geq 1) &= \mathbf{P}(X = 1) + \mathbf{P}(X \geq 2) \\ &\geq n^{-n} + (1-n^{-n})^2 p \geq n^{-n} + (1-2n^{-n})p \ . \end{aligned}$$

- Auflösen nach p ergibt: $p \geq 1/2$.

Zusammenfassung

- Es gibt Zusammenhänge zwischen Random Walks und elektrischen Widerstandsnetzwerken.
- Random Walks kann man benutzen, um in Graphen zwei Knoten auf Verbundenheit zu testen.