(iii) DNF, KNF sind <u>nicht</u> eindeutig.

Schwieriges Problem („Logikentwurf"):
(two-level logic minimization)
Finde effiziente / kurze DNF / KNF.

<u>Def.</u>: (a) Eine Disjunktion von Literalen heißt Klausel.

(b) Eine Formel ist in k-KNF, falls sie in KNF ist, und falls jede ihrer Klauseln höchstens k Literale enthält.

<u>Satz</u> Zu jeder Formel F gibt es eine Formel F' in 3-KNF, für die gilt: F erfüllbar $\Leftrightarrow$ F' erfüllbar.

<u>Bew.</u>: Sei F oBdA in KNF. Sei

(*) $L_{i,1} \vee \ldots \vee L_{i,m_i}$ eine Klausel von F.

Ersetze diese durch

(**) $(L_{i,1} \vee \ldots \vee L_{i,m_i-2} \vee A)$
$\wedge (\neg A \vee L_{i,m_i-1} \vee L_{i,m_i})$.

Hierdurch reduziert sich die Anzahl der Literale um 1. Außerdem ist (*) erfüllbar $\Longleftrightarrow$ (**) erfüllbar, weil:

<u>(*) erfüllbar $\Rightarrow$ (**) erfüllbar:</u>

Betrachte Modell für (*). Falls $L_{i,1} \vee \ldots \vee L_{i,m_i-2}$ den Wert 1 annimmt, setze $A = 0$. Falls nicht, dann muß $L_{i,m_i-1} \vee L_{i,m_i}$ den Wert 1 annehmen. Dann setze $A = 1$.

<u>(**) erfüllbar $\Rightarrow$ (*) erfüllbar:</u>

Betrachte Modell für (**). Falls A mit 0 belegt wird, dann muß $L_{i,1} \vee \ldots \vee L_{i,m_i-2}$ den Wert 1 bekommen, d.h. (*) ist erfüllbar. Falls A mit 1 belegt wird, dann muß $L_{i,m_i-1} \vee L_{i,m_i}$ den Wert 1 bekommen, d.h. (*) ist erfüllbar. $\boxtimes$

<u>Bem.:</u> Die obige Transformation produziert 3-KNF, die nicht viel länger als die ursprüngliche KNF ist. Man erhält $O(n \cdot \max\{m_i : i = 1, \ldots, n\})$ viele Klauseln (anstatt n).

§4 „Schnelle", randomisierte Algorithmen für das Erfüllbarkeitsproblem von k-KNF-Formeln

<u>Algorithmus</u> (Schöning, 1999)

<u>Eingabe</u> k-KNF-Formel F mit n atomaren Teilformeln, $M \in \mathbb{N}$.

<u>Ausgabe</u> Modell $\mathcal{A}$ für F oder „kein Modell gefunden".

1. Wähle zufällige, gleichverteilte Startbelegung $\mathcal{A}$
2. Wiederhole M-mal:

   if $\mathcal{A}$ ist Modell von F:

   stop

   else:

   wähle Klausel C von F, für die $\mathcal{A}$ kein Modell ist

   wähle zufällig, gleichverteilt eine der ($\leq k$) Variablen in C

   negiere den Wert dieser Variable in $\mathcal{A}$.
3. „kein Modell gefunden".

Satz Falls $k = 2$ und F erfüllbar ist, dann findet der Algo. ein Modell nach einer erwarteten Anzahl von $M \leq n^2$ Schleifendurchläufen.

Bew.: Sei $\mathcal{S}$ ein Modell von F und seien $A_0, A_1, A_2, \ldots$ die Belegungen, die der Algo. erzeugt. Definiere

f_ℓ = Anzahl Variablen, die in $\mathcal{S}$ und A_ℓ unterschiedlich sind.

Klar: (i) $0 \leq f_\ell \leq n$

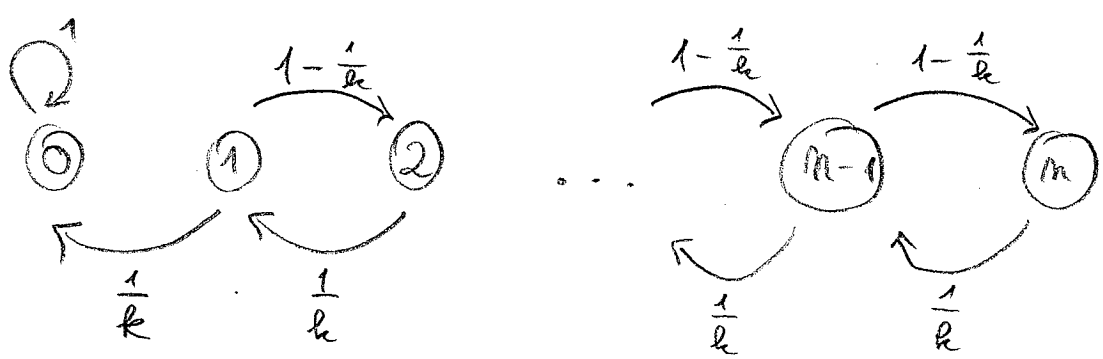
(ii) wenn $f_\ell = 0$, dann stoppt der Algo. (evtl. auch schon früher)

(iii) $f_{\ell+1} = f_\ell \pm 1$

(iv) $\Pr[f_{\ell+1} = f_\ell - 1] \geq \frac{1}{k}$

(v) $\Pr[f_0 = i] = \frac{1}{2^n}\binom{n}{i}$

Betrachte die Irrfahrt („random walk")



Definiere

d_i = erwartete Anzahl von Schleifen durchläufen, wenn mit $f_0 = i$ gestartet wurde.

Dann gilt

$$d_0 = 0$$

$$d_i = \frac{1}{2}(d_{i-1}+1) + \frac{1}{2}(d_{i+1}+1) \quad \text{für } i \in [n-1]$$

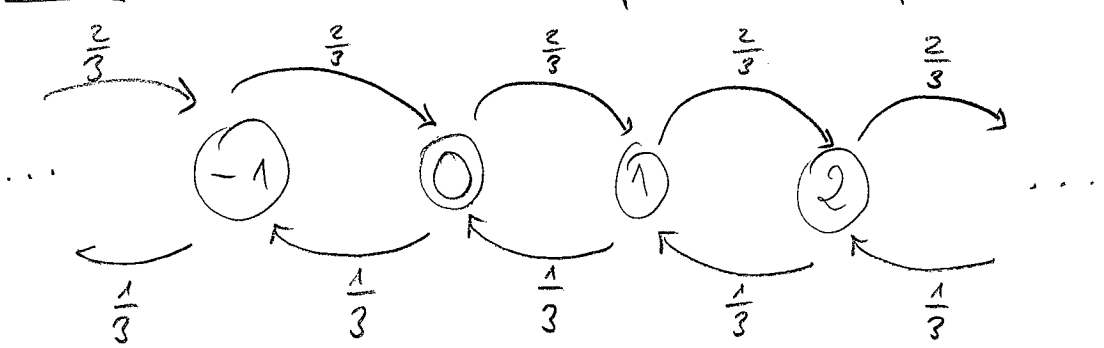
$$d_n = d_{n-1} + 1$$

Auflösen ($\rightarrow$ Aufgabe) liefert

$$d_i = 2ni - i^2 \leq n^2, \quad i = 0, \dots, n. \qquad \boxtimes$$

<u>Satz</u> Falls $k = 3$ und F erfüllbar ist, dann findet der Algo. nach höchstens $3n$ Schleifendurchläufen mit Wahrscheinlichkeit $\left(\frac{3}{4}\right)^n \Theta\left(\frac{1}{\sqrt{n}}\right)$ ein Modell.

<u>Bew.</u>: Betrachte vereinfachte Irrfahrt:



Wenn der Algo. lange läuft, hat er die Tendenz nach rechts zu laufen

Ang. $f_0 = i$. Definiere einen

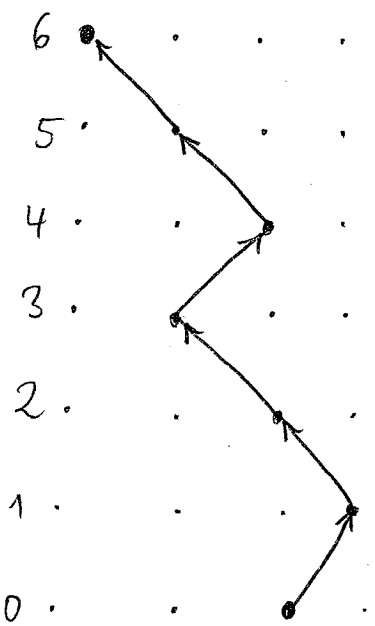
<u>i-Weg</u>: Weg von i nach 0, der i Schritte nach rechts, $2i$ Schritte nach links macht und die 0 nach $3i$ Schritten zum ersten Mal erreicht.

Es gilt: $\Pr[i\text{-Weg}] = \left(\frac{1}{3}\right)^{2i}\left(\frac{2}{3}\right)^{i}$.

$W(i) = \#\, i\text{-Wege}$ (Anzahl der i-Wege)

<u>Beh.</u>: $W(i) = \binom{3i-1}{i} - \binom{3i-1}{i-1}$.

<u>Bew.</u>: Darstellung von i-Wegen im Koordinatensystem:
z.B. stelle 2-Weg R, L, L, R, L, L in $\mathbb{Z} \times [0, 3i]$ wie folgt dar:



Müssen Wege zählen, die von $(i,0)$ nach $(1, 3i-1)$ führen.
[Der letzte Schritt ist immer von $(1, 3i-1)$ nach $(0, 3i)$.].

Es gibt $\binom{3i-1}{i}$ solcher Wege, wobei jedoch auch Wege gezählt werden, die evtl. 0 nicht nach $3i$ Schritten zum ersten Mal erreichen.