

Möglichkeiten zur Reduktion der Fehlerrate

(1) Volumen-konstante Algorithmen

Viele DNA-Algorithmen sind von folgender Gestalt:

(1) Erzeuge große Menge an DNA-Sequenzen (= potentielle Lösungen)

(2) Wende eine Folge von m Filteroperationen auf die DNA-Lösung an.

Jede Filteroperation reduziert das DNA-Volumen

Ziel: Genau die ersten Lösungen des Problems sollen alle m Filteroperationen überleben.

↳ Volumen-reduzierender Algorithmus

• Beispiel hierfür: Adleman's Lösung des Hamilton-Pfad Problems.

• Wir betrachten im folgenden eine Abstraktion eines Volumen-reduzierenden DNA-Algorithmus:

(1) Erzeuge 2^m viele DNA-Stränge ($m = \text{Inputgröße}$) \rightarrow DNA-Lösung L_0

Nur ein Strang ist gut (d.h. repräsentiert eine erste Lösung des zu lösenden Problems, alle anderen Stränge sind schlecht.)

(2) Folge von m Filteroperationen:
for $i := 1$ to m do

$L_i := \text{filter}_i(L_{i-1})$
endfor

- Annahme: Jede Operation filter_i reduziert das DNA-Volumen um den gleichen Faktor:

$$\frac{|L_i|}{|L_{i-1}|} = \frac{1}{v} \quad (i \in \{1, \dots, m\}) \text{ für eine Zahl } v \in \mathbb{R}, v > 1.$$

Außerdem soll nach m Schritten nur noch ein DNA-Strang überleben:

$$|L_m| = 1, \text{ d.h. } |L_0| = 2^m = v^m$$

$$\rightarrow v = 2^{m/m}$$

- Beachte: Nach $\frac{m}{m}$ Schritten hat sich das DNA-Volumen halbiert:

$$v^{\frac{m}{m}} = 2^{\frac{m}{m}} = 2$$

- Unsere Hoffnung ist, dass L_m nur den guten Strang enthält.

- $p_i :=$ Wahrscheinlichkeit, dass in der i -ten Filteroperation filter_i ein DNA-Strang nicht nach L_i kommt, obwohl er nach L_i hätte kommen sollen.

Wir nehmen an: $p_1 = p_2 = \dots = p_m =: p$

- $q_i :=$ Wahrscheinlichkeit, dass in der i -ten Filteroperation filter_i ein DNA-Strang nach L_i kommt, obwohl er ~~bleibt~~ aus der DNA-Lösung hätte entfernt werden sollen.

Wir nehmen an: $q_1 = q_2 = \dots = q_m =: q$

- Typische Werte: $p = 0,05$, $q = 10^{-6}$
- Problem: Bei $m = 1000$ Schritten gilt:

$$\text{Prob}[\text{guter Strang} \in L_m] = (1-p)^m \approx 5,3 \cdot 10^{-23} !$$

Wir modifizieren unseren abstrakten DNA-Algorithmus wie folgt:

(1) Erzeuge 2^m viele DNA-Stränge
(1 guter Strang, $2^m - 1$ schlechte Stränge)

(2) for $i := 1$ to m do
 $L_i := \text{filter}_i(L_{i-1})$
 if " i ist Vielfaches von $\frac{m}{2}$ " then
 dupliziere jeden Strang in L_i
 mittels Polymerase-Ketten-Reaktion (PCR)

endfor feste Konstante

(3) for $j := 1$ to t do
 Wähle einen beliebigen Strang
 $s \in L_m$ aus.
 if " s ist Kopie des guten Strang" then
 output(YES)
 endif

~~then~~
 output(YES)

endfor
 output(NO)

Beachte: Auf Grund unserer Annahmen halbiert sich alle $\frac{m}{2}$ Filteroperationen das DNA-Volumen.

Durch das Duplizieren mittels Polymerase-Ketten-Reaktion wird also wieder das ursprüngliche DNA-Volumen hergestellt.

↳ Volumen-konstanter Algorithmus.

Wir analysieren zunächst die Wahrscheinlichkeit

$$p_s := \text{Prob}[\underbrace{\text{guter Strang}}_{g_i} \in L_m]$$

(s steht für survival)

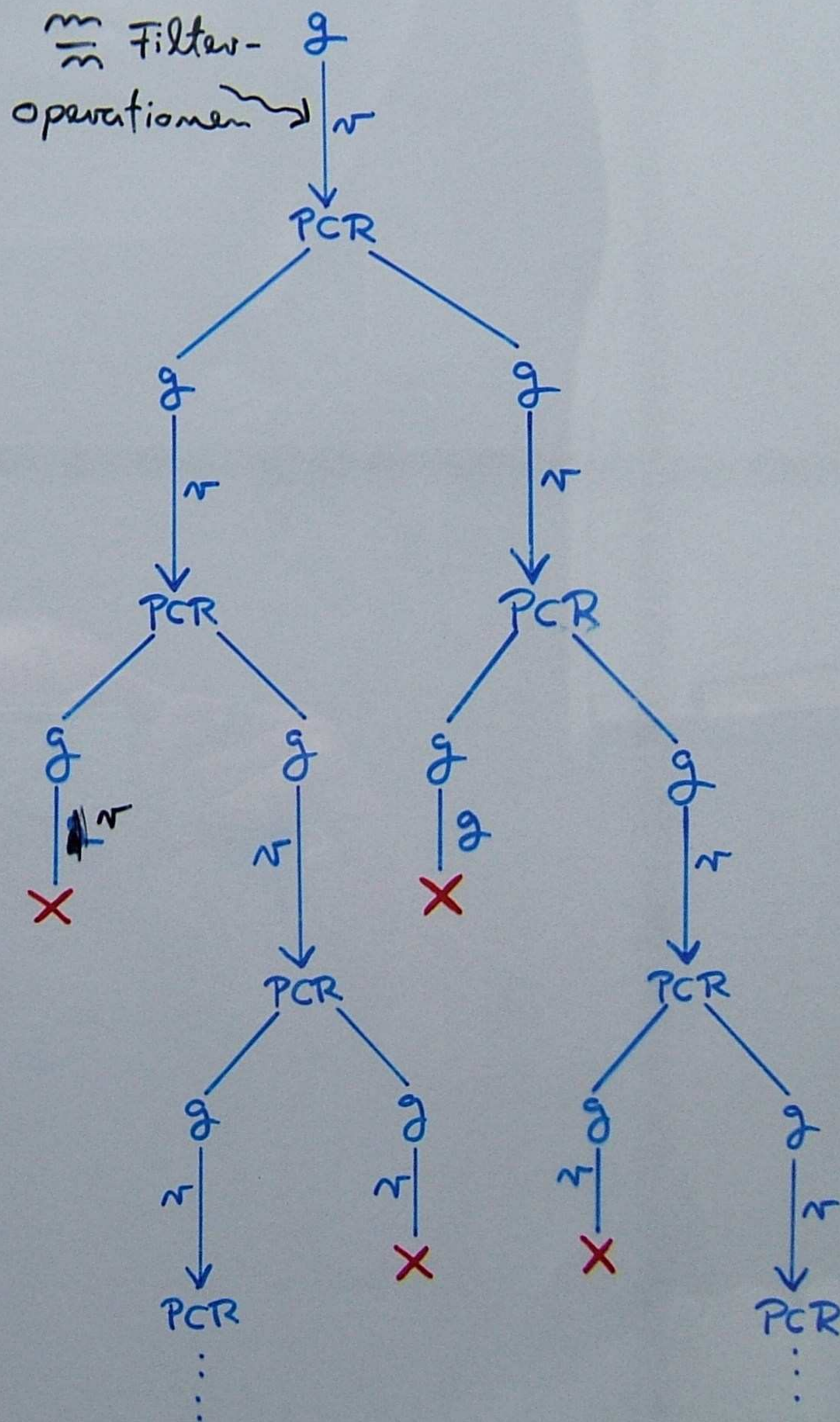
Die Wahrscheinlichkeit, dass g die ersten $\frac{m}{n}$ Filteroperationen überlebt, ist

$$v := (1-p)^{\frac{m}{n}}$$

Hat g die ersten $\frac{m}{n}$ Filteroperationen überlebt, so wird g in der PCR dupliziert.

Jeder der Duplikate durchläuft dann wieder $\frac{m}{n}$ Filteroperationen bis zur nächsten PCR.

↳ Verzweigungsprozess



Sei $x_i := \text{Prob}[\text{nach den ersten } i \cdot \binom{m}{n} \text{ Filteroperationen existiert keine Kopie von } g \text{ mehr}]$

Wir interessieren uns für x_m ,
genauer für $1 - x_m = p_s$

Es gilt $x_0 = 0$

$$x_{i+1} = (1-r) + r \cdot x_i^2$$

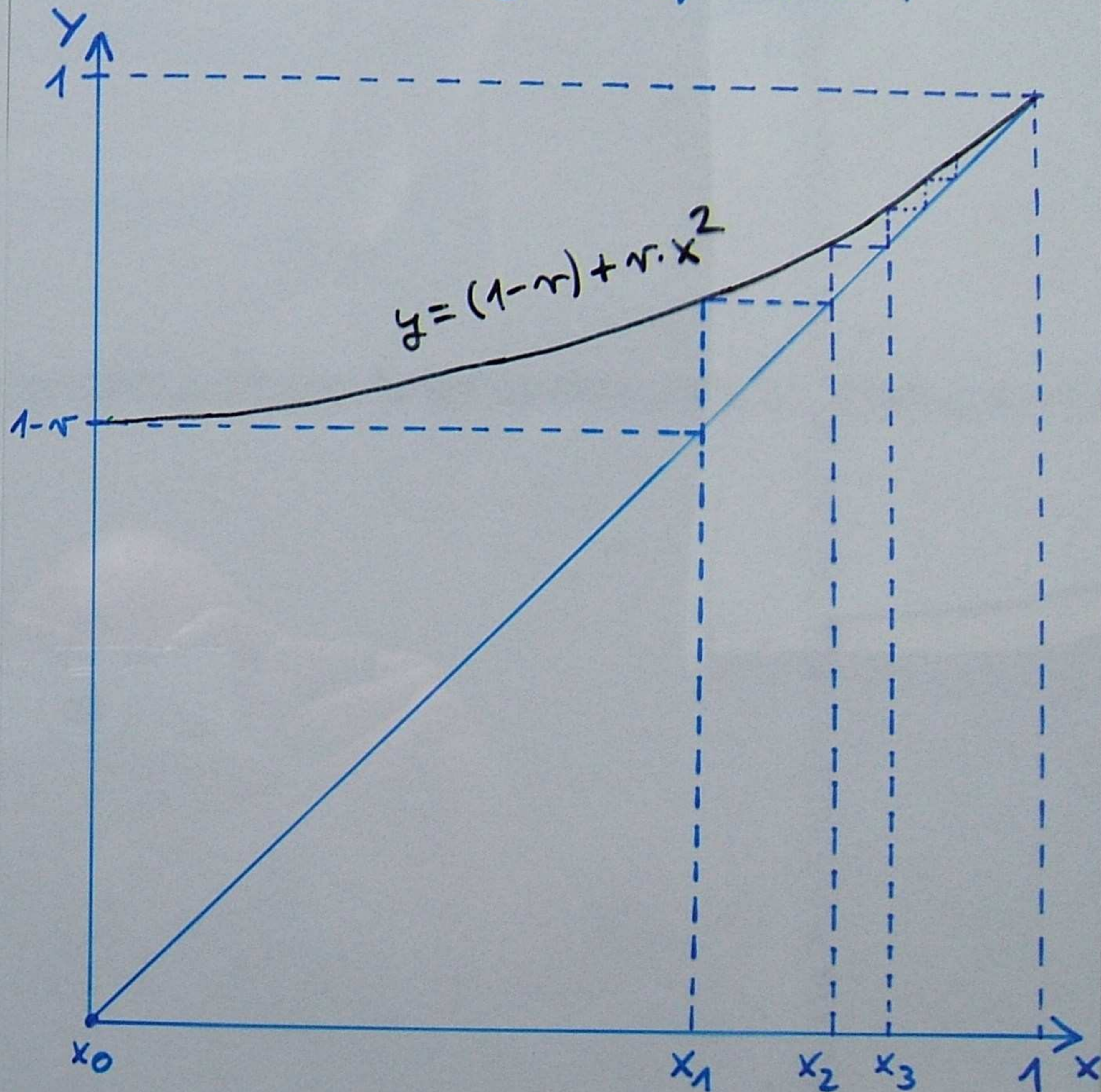
Betrachte die Gleichung

$$y = (1-r) + r \cdot y^2$$

Diese hat zwei Lösungen:
 $y_0 = 1$ und $y_1 = \frac{1}{r} - 1$

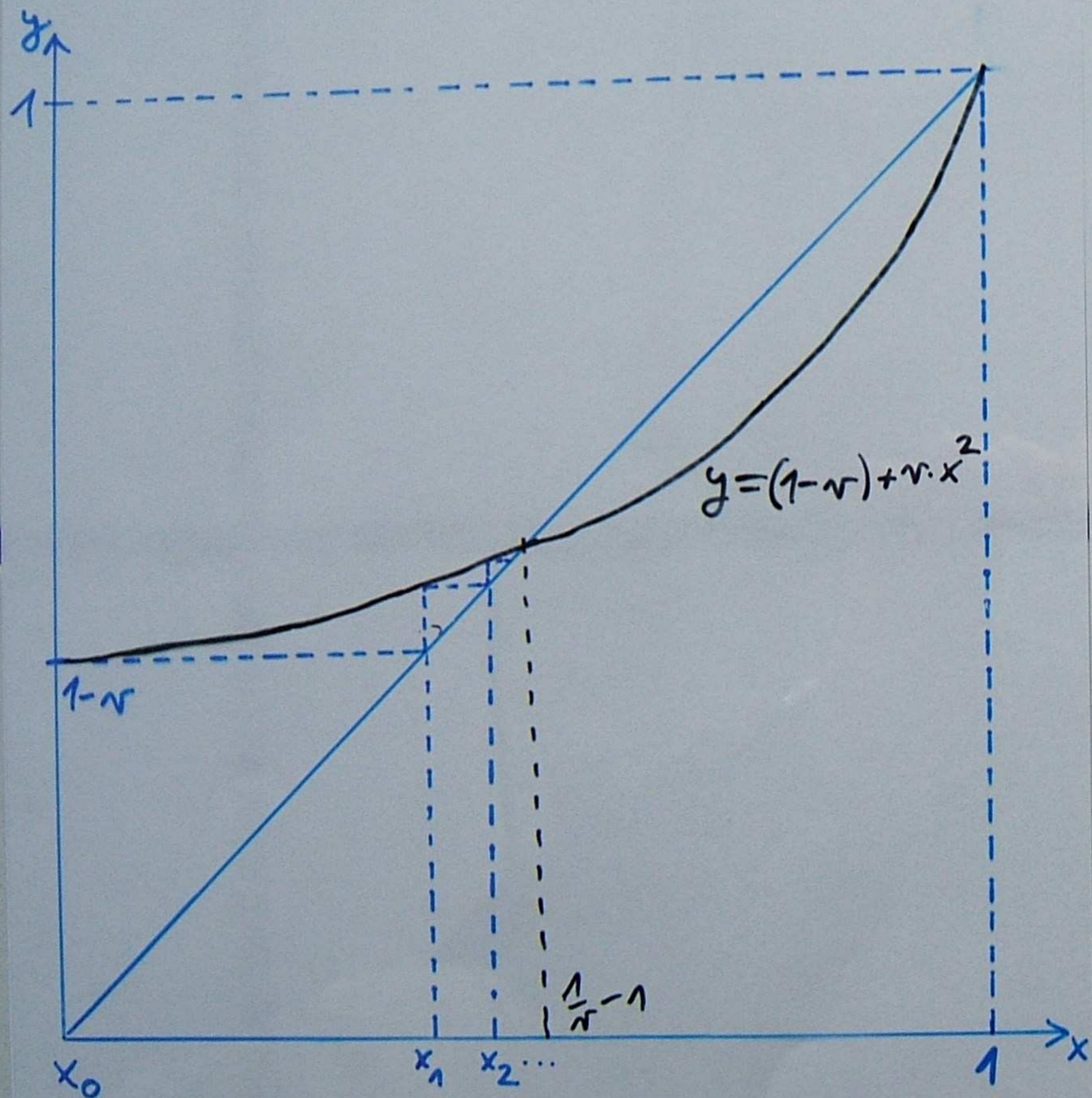
(eine dieser Lösungen ist die Wahrscheinlichkeit, dass die "Population der guten Stränge, g " irgendwann ausstirbt. Welche?)

1. Fall: $r \leq \frac{1}{2} \rightarrow \frac{1}{r} - 1 \geq 1$



Es gilt $x_i \rightarrow 1$ für $i \rightarrow \infty$

2. Fall $r > \frac{1}{2}$



Es gilt $x_i \rightarrow \frac{1}{r} - 1$ für $i \rightarrow \infty$

Für $m \geq 50$ (und $r = (1-p)^{\frac{m}{n}} > \frac{1}{2}$)
gilt somit

$$P_S = 1 - x_m \approx 2 - r^{-1} = \underline{\underline{2 - (1-p)^{-\frac{m}{n}}}}$$

Einige Werte für $m = 900$ (Anzahl d. Filteroperationen), $n = 60$ (Inputlänge)

$$\hookrightarrow \frac{m}{n} = 15$$

p	P_S	$(1-p)^{900}$
0,03	0,42	$\approx 10^{-12}$
0,02	0,65	$\approx 10^{-8}$
0,01	0,85	$\approx 10^{-4}$

$$(1-p)^{\frac{m}{n}} = r \approx 0,63$$

Wahrscheinlichkeit,
dass gute String y
alle 900 Filter-
operationen
überlebt, falls
PCR mit alle
15 Schritte an-
gewendet wird.

WVF

Wie groß ist das Verhältnis

$$\frac{\# \text{ gute Stränge}}{\# \text{ schlechte Stränge}} \quad (*)$$

in der Lösung L_m ? (Sollte nicht zu klein sein)

Beachte: PCR ändert nicht den Quotienten (*). Wir ignorieren deshalb im folgenden die PCR-Schritte.

Für einen Strang x sei $M(x)$ die Anzahl der Filteroperationen, bei denen x herausgefiltert (d.h. entfernt) werden soll.

↳ Für den guten Strang g gilt $M(g) = 0$, für jeden schlechten Strang x gilt $M(x) > 0$.

$$\begin{aligned} \text{Prob}[x \in L_m] &= q^{M(x)} \cdot (1-p)^{m-M(x)} \\ &= (1-p)^m \cdot \left(\frac{q}{1-p}\right)^{M(x)} \end{aligned}$$

Wir berechnen nun

$$E_g = \frac{E[\text{Anzahl gute Stränge in } L_m]}{E[\text{Anzahl von Strängen in } L_m \text{ insgesamt}]}$$

$$\begin{aligned} E_g &= (1-p)^m \\ E &= \sum_{x \in \{0,1\}^m} (1-p)^m \cdot \left(\frac{q}{1-p}\right)^{M(x)} \end{aligned}$$

Sei $M_i := |\{x \in \{0,1\}^m \mid M(x) = i\}|$
($M_0 := 1$)
denn es gibt genau einen guten Strang

$$\rightarrow E = (1-p)^m \cdot \sum_{i \geq 0} \left(\frac{q}{1-p}\right)^i \cdot M_i$$

Für $q = 10^{-6}$, $p = 0.05$, $m \leq 70$

gilt bereits

$$\left(\frac{q}{1-p}\right)^4 \cdot M_4 \leq \left(\frac{q}{1-p}\right)^4 \cdot 2^{70}$$

$$\approx 0,0014$$

$$\rightarrow \frac{E_q}{E} = \frac{1}{\sum_{i \geq 0} M_i \cdot \left(\frac{q}{1-p}\right)^i}$$

$$\approx \frac{1}{1 + M_1 \left(\frac{q}{1-p}\right) + M_2 \left(\frac{q}{1-p}\right)^2 + M_3 \left(\frac{q}{1-p}\right)^3}$$

Zahlenbeispiel:

$$q = 10^{-6}, p = 0.05, M_1 = 10^8, \\ M_2 = 10^{14}, M_3 = 10^{20}$$

$$\rightarrow \frac{E_q}{E} \approx \frac{1}{334}$$

Dies wäre akzeptabel:

Beachte: In unserem modifizierten (Volumen-konstanten) Algorithmus wählen wir am Ende (Schritt (3)) t viele Stränge aus L_m zufällig aus, und akzeptieren, falls mind. einer der ausgewählten Stränge gut ist.

Im obigen Zahlenbeispiel sollten wir $t \approx 334$ wählen.

Wir können $\frac{E_g}{E}$ auch dadurch
größer machen, indem wir jede
Filteroperation zweimal durchführen.
filter:

→ Für jeden Strang $x \in \{0,1\}^m$
verdoppelt sich $M(x)$

→ M_1, M_2, M_3 werden kleiner.

(M_1, M_3 werden 0, der neue
 M_2 -Wert ist der alte M_1 -Wert)