ARITHMETISCHE CODES

Gegeven: Binaire bron $\{X_n\}_{n=1,2,\cdots,N}$
met $Pr\{X_1=x_1, X_2=x_2, \cdots, X_N=x_N\}$.

Huffman-Constructie hiervoor:

- Optimale FV code (kleinste rate)
- constructietijd: minstens evenredig met 2^N
- geheugenbeslag: minstens evenredig met 2^N
- encodeer/decodeertijd: minimaal
  ⇒ RAADPLEGEN TABEL of BOOM

→ Gevolg: Huffman code alleen bruikbaar voor $N < 15$ à 25

Voor N groot: Arithmetische Codes

- Bijna optimaal
- constructietijd: geen
- geheugenbeslag: lineair in N (zelfs constant kan)
- encodeer/decodeertijd kwadratisch in N (zelfs lineair kan)

⇒ BEREKENEN VAN CODEWOORD uit BRONWOORD en VICEVERSA

(gebaseerd op Elias algoritme)

ELIAS: Associeer rijtjes met intervallen

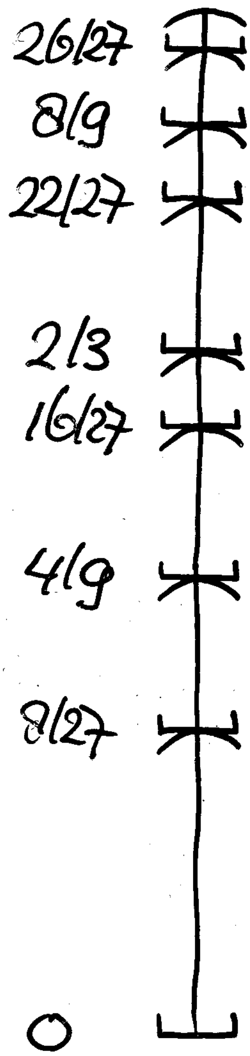
— orden de rijtjes in $\{0,1\}^N$ lexicografisch

— $I(x^N) \triangleq [Q(x^N), Q(x^N)+P(x^N))$

$(x^N \triangleq x_1 x_2 \cdots x_N)$ met $Q(x^N) \triangleq \Pr\{X^N < x^N\}$

en $P(x^N) \triangleq \Pr\{X^N = x^N\}$

voorbeeld: $\Pr\{X_n = 1 \mid X_1 = x_1, \cdots, X_{n-1} = x_{n-1}\} = 1/3$,
$N = 3$.



1/27 $I(111) = [26/27, 1)$

2/27 $I(110) =$

2/27 $I(101) =$

4/27 $I(100) = \ldots$

2/27 $I(011) =$

4/27 $I(010) = [4/9, 16/27)$

4/27 $I(001) = [8/27, 4/9)$

8/27 $I(000) = [0, 8/27)$

Elias Algorithme

Eigenschappen:

A: lengte $I(x^N) = \Pr\{X^N = x^N\}$

B: $\tilde{x}^N \neq x^N \Rightarrow I(\tilde{x}^N) \cap I(x^N) = \phi$

immers stel $\tilde{x}^N < x^N$ dan

$$Q(\tilde{x}^N) + P(\tilde{x}^N) = \Pr\{X^N \leq \tilde{x}^N\}$$
$$\leq \Pr\{X^N < x^N\} = Q(x^N)$$

C: $$\bigcup_{x^N \in \{0,1\}^N} I(x^N) = [0,1)$$

want: $Q(0^N) = 0$

$$Q(x^N) + P(x^N) = Q(x^N + 1) \quad , x^N \neq 1^N$$

$$Q(1^N) + P(1^N) = 1$$

Dus de rijtjes $x^N \in \{0,1\}^N$ delen het eenheidsinterval $[0,1)$ op in disjuncte deelintervallen met lengte $\Pr\{X^N = x^N\}$.

<u>Compressie</u>: Door het blokrijtje x^N door een punt $C(x^N) \in I(x^N)$ te representeren. ($C(x^N)$ = binaire fractie.)

<u>Meer precies</u>:

$$C(x^N) \triangleq \left\lceil Q(x^N) \cdot 2^{k(x^N)} \right\rceil \cdot 2^{-k(x^N)}$$

met $k(x^N)$ minimaal (≥ 0)
maar zodanig dat $C(x^N) \in I(x^N)$

<u>Nu geldt</u>: $k(x^N) \leq \left\lceil -{}^{2}\log P(x^N) \right\rceil \quad \circledast$

want

$$\begin{aligned} Q(x^N) \leq C(x^N) &< Q(x^N) + 2^{-\lceil -\log P(x^N) \rceil} \\ &\leq Q(x^N) + 2^{\log P(x^N)} \\ &= Q(x^N) + P(x^N) \end{aligned}$$

<u>Voorbeeld</u>: $I(100) = \left[2/3, \; 2/3 + 4/27 \right)$

$-\log P(100) \cong 2.75$ dus $k(100) \leq 3$

$k = 3 \quad C(100) = \lceil 2/3 \cdot 8 \rceil \cdot 1/8 = 6/8 < 22/27$

$\underline{k = 2} \quad C(100) = \lceil 2/3 \cdot 4 \rceil \cdot 1/4 = 3/4 < 22/27$

$k = 1 \quad C(100) = \lceil 2/3 \cdot 2 \rceil \cdot 1/2 = 1 \not< 22/27$

Omdat $C(x^N) \in [0,1)$ hoeft $C(x^N)$ uit niet meer dan $k(x^N)$ binaire digits te bestaan.

voorbeeld: $I(100) = [2/3, 2/3 + 4/27)$ dan $k=2$ en $C(100) = 3/4 = .11$

Gemiddelde codewoordlengte:

$$E[k(x^N)] = \sum_{x^N \in \{0,1\}^N} \Pr\{X^N = x^N\} k(x^N)$$

$$\leq \sum_{x^N \in \{0,1\}^N} \Pr\{X^N = x^N\} \cdot \lceil -\log P(x^N) \rceil$$

$$\leq \sum_{x^N \in \{0,1\}^N} \Pr\{X^N = x^N\} \cdot \left(\log \frac{1}{\Pr\{X^N = x^N\}} + 1 \right)$$

$$= H(X^N) + 1.$$

Dit is vergelijkbaar met Huffman codering, maar ...

voorbeeld: $Pr\{X_n = 1 | -\} = 1/3$, $N = 3$.

x^3	$P(x^3)$	$Q(x^3)$	$I(x^3)$	$C(x^3)$
000	8/27	0	$[0, 8/27)$	$0 = .$
001	4/27	8/27	$[8/27, 4/9)$	$3/8 = .011$
010	4/27	4/9	$[4/9, 16/27)$	$1/2 = .1$
011	2/27	16/27	$[16/27, 2/3)$	$5/8 = .101$
100	4/27	2/3	$[2/3, 22/27)$	$3/4 = .11$
101	2/27	22/27	$[22/27, 8/9)$	$7/8 = .111$
110	2/27	8/9	$[8/9, 26/27)$	$15/16 = .1111$
111	1/27	26/27	$[26/27, 1)$	$31/32 = .11111$

We zien dat $E\,l(x^3) = 1.81$ en dit is zelfs minder dan $H(x^3) = 2.75$ bit... (?)
Code is niet prefix-vrij, ofwel
Codewoordlengte-informatie ontbreekt!

Daarom plaatsen we een prefix voor het codewoord $C(x^N)$ die aangeeft hoe lang $C(x^N)$ is.

Codering van de natuurlijke getallen.

(ook Elias)

$g \in \{1, 2, 3, \cdots\}$, niet begrensd!
$b(g)$ is de binaire representatie van g,
$l(g)$ is de lengte van $b(g)$.

Voorbeeld: $g = 13$, $b(13) = 1101$, $l(13) = 4$.

We kunnen g nu prefix-vrij encoderen als volgt:

$$c(g) = \underbrace{00 \cdots 0}_{\# = l(g)-1} \underbrace{1 \cdots}_{b(g)}$$

$l(g)-1$ nullen gevolgd door $b(g)$.

Voorbeeld: $g = 13$, $c(13) = 000\ 1101$

Decoderen is nu heel eenvoudig: we tellen eerst het aantal nullen (dit kan omdat $b(g)$ altijd begint met een 1) en weten nu hoe lang $b(g)$ is, dus kunnen we $b(g)$ lezen.

Voorbeeld: $c = 000\ 1101$,
dus $l(g) = 3 + 1 = 4$, daarom $g = 13$.

Omdat bij de arithmetische code $k(x^N) \in \{0, 1, 2, \cdots\}$ plaatsen we de prefix $c(k(x^N)+1)$ voor het codewoord $C(x^N)$.

voorbeeld:

x^3	$C(x^3)$	$c(k(x^3)+1), C(x^3)$
000	.	1,
001	.011	00100,011
010	.1	010,1
011	.101	00100,101
100	.11	011,11
101	.111	00100,111
110	.1111	00101,1111
111	.11111	00110,11111

want
$c(1) = 1$
$c(2) = 010$
$c(3) = 011$
$c(4) = 00100$
$c(5) = 00101$
$c(6) = 00110$

De totale code is nu prefix-vrij. Gemiddelde codewoordlengte $= 5.04$

Bij voldoend grote N is de lengte van de prefix verwaarloosbaar klein.

Efficiente bepaling van $I(x^n)$:

Encoder vormt $I(\phi)=[0,1)$ om tot achtereenvolgens $I(x_1), I(x_1x_2), \ldots, I(x_1x_2\cdots x_n)$

Laat $I(x^n) \triangleq [Q(x^n), Q(x^n)+P(x^n))$

($x^n \triangleq x_1x_2\cdots x_n$) met $Q(x^n)=\Pr\{X^n<x^n\}$ en $P(x^n)=\Pr\{X^n=x^n\}$ dan

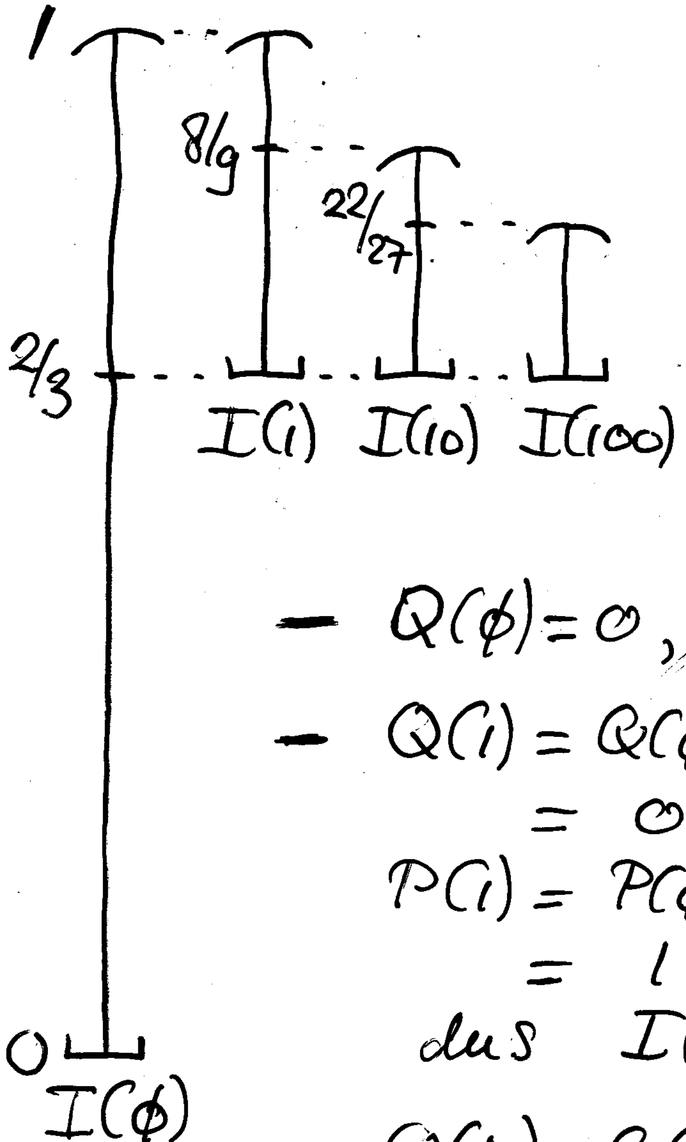
$$\begin{aligned}Q(x^n) &= \Pr\{X^n<x^n\}\\ &=\Pr\{X^{n-1}<x^{n-1}\}+\Pr\{X^{n-1}=x^{n-1}, X_n<x_n\}\\ &=\Pr\{X^{n-1}<x^{n-1}\}+\Pr\{X^{n-1}=x^{n-1}\}\Pr\{X_n<x_n|X^{n-1}=x^{n-1}\}\\ &=Q(x^{n-1})+P(x^{n-1})\Pr\{X_n<x_n|X^{n-1}=x^{n-1}\}\end{aligned}$$

en

$$\begin{aligned}P(x^n) &= \Pr\{X^n=x^n\}.\\ &=\Pr\{X^{n-1}=x^{n-1}\}\Pr\{X_n=x_n|X^{n-1}=x^{n-1}\}\\ &=P(x^{n-1})\Pr\{X_n=x_n|X^{n-1}=x^{n-1}\}\end{aligned}$$

dus $I(x^n)$ kan makkelijk bepaald worden uit $Q(x^{n-1})$ en $P(x^{n-1})$ dus uit $I(x^{n-1})$.

We gaan uit van $Q(\phi)=0$ en $P(\phi)=1$.



<u>Voorbeeld:</u>

$Pr\{X_n = 1 \mid -\} = 1/3$,
$N = 3$.

- $Q(\phi) = 0,\ P(\phi) = 1,\ I(\phi) = [0, 1)$
- $Q(1) = Q(\phi) + P(\phi) \cdot Pr\{X_1 < 1\}$
  $= 0 + 1 \cdot 2/3 = 2/3$
  $P(1) = P(\phi) \cdot Pr\{X_1 = 1\}$
  $= 1 \cdot 1/3 = 1/3$
  dus $I(1) = [2/3, 1)$
- $Q(10) = Q(1) + P(1) \cdot Pr\{X_2 < 0\}$
  $= 2/3 + 1/3 \cdot 0 = 2/3$
  $P(10) = P(1) \cdot Pr\{X_2 = 0\}$
  $= 1/3 \cdot 2/3 = 2/9$
  dus $I(10) = [2/3, 8/9)$
- $Q(100) = Q(10) + P(10) Pr\{X_3 < 0\}$
  $= 2/3 + 2/9 \cdot 0 = 2/3$
  $P(100) = P(10) \cdot Pr\{X_3 = 0\}$
  $= 2/9 \cdot 2/3 = 4/27$
  dus $I(100) = [2/3, 22/27)$

De encoder bepaalt $I(x^N)$ recursief en daarna $C(x^N) \in I(x^N)$.

Hoe werkt de decoder?

merk op dat omdat $C(x^N) \in I(x^N)$

$$C(x^N) < Q(x^N) + P(x^N) = \Pr\{X^N \leq x^N\}$$
$$\leq \Pr\{X^n \leq x^n\} = \Pr\{X^n \leq x^{n-1}, 0\}$$
$$= \Pr\{X^n < x^{n-1}, 1\} = Q(x^{n-1}, 1) \text{ als } x_n = 0$$

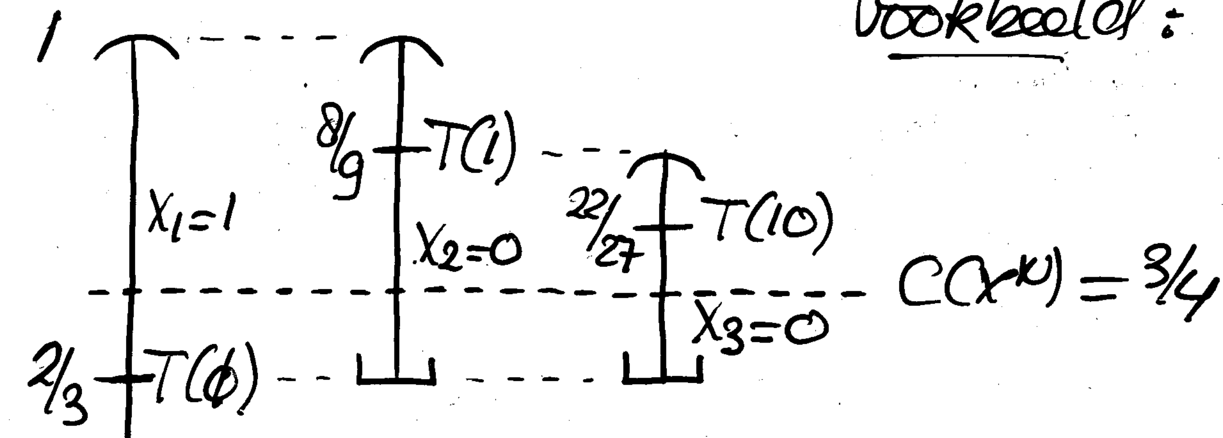
en

$$C(x^N) \geq Q(x^N) = \Pr\{X^N < x^N\}$$
$$\geq \Pr\{X^n < x^n\}$$
$$= \Pr\{X^n < x^{n-1}, 1\} = Q(x^{n-1}, 1) \text{ als } x_n = 1$$

Dus kunnen we de drempel $T(x^{n-1}) \triangleq Q(x^{n-1}, 1)$ gebruiken om te bepalen of $x_n = 0$ of $x_n = 1$ is nadat we x^{n-1} bepaald hebben.

Merk op dat $T(x^{n-1})$ uit $Q(x^{n-1})$ en $P(x^{n-1})$ kan worden bepaald. De decoder moet dus ook $Q(x^n)$ en $P(x^n)$ updaten.

voorbeeld:

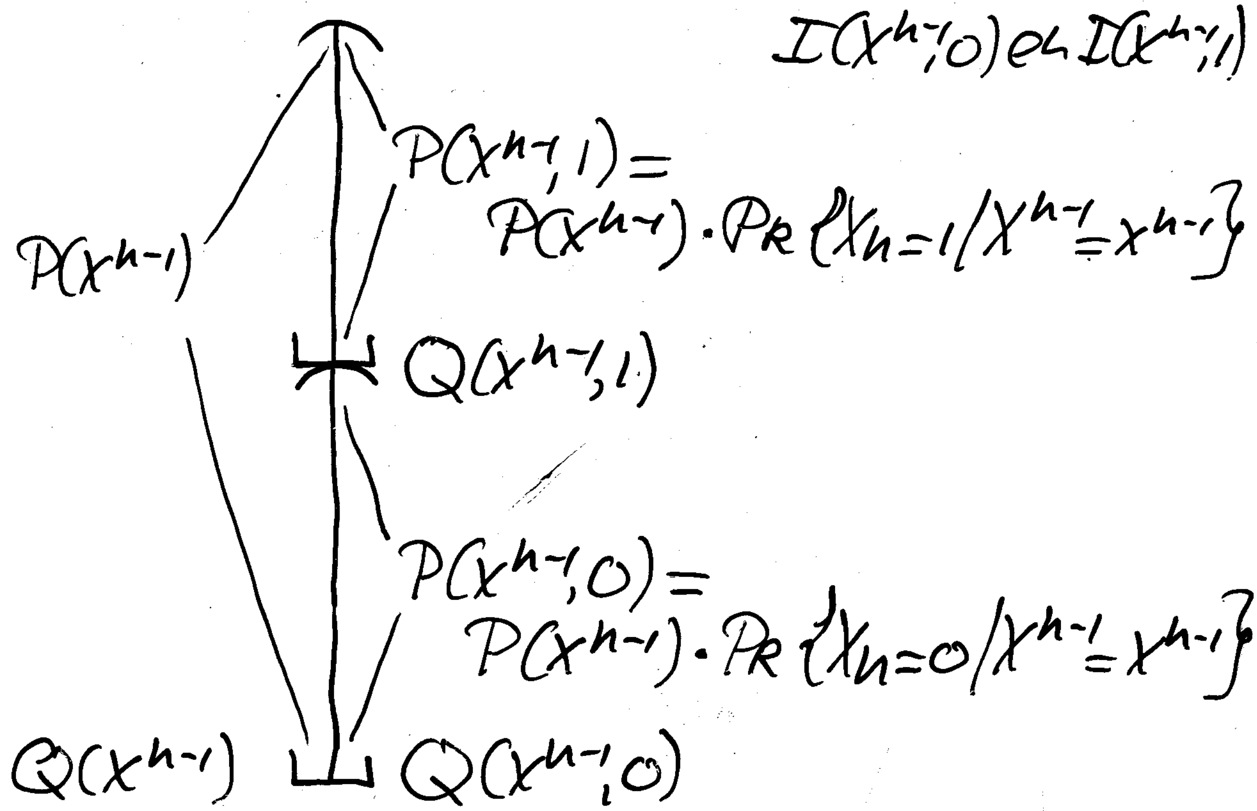


$Q(\phi)=0$, $P(\phi)=1$ dus

- $T(\phi) = Q(1) = Q(\phi)+P(\phi)\cdot 2/3 = 2/3$
  aangezien $3/4 \geq 2/3$ is $X_1 = 1$
  nu $Q(1) = 2/3$, $P(1) = 1/3$
- $T(1) = Q(11) = Q(1)+P(1)\cdot 2/3 = 8/9$
  aangezien $3/4 < 8/9$ is $X_2=0$
  nu $Q(10) = 2/3$, $P(10) = 2/9$
- $T(10) = Q(101) = Q(10)+P(10)\cdot 2/3 = 22/27$
  aangezien $3/4 < 22/27$ is $X_3 = 0$

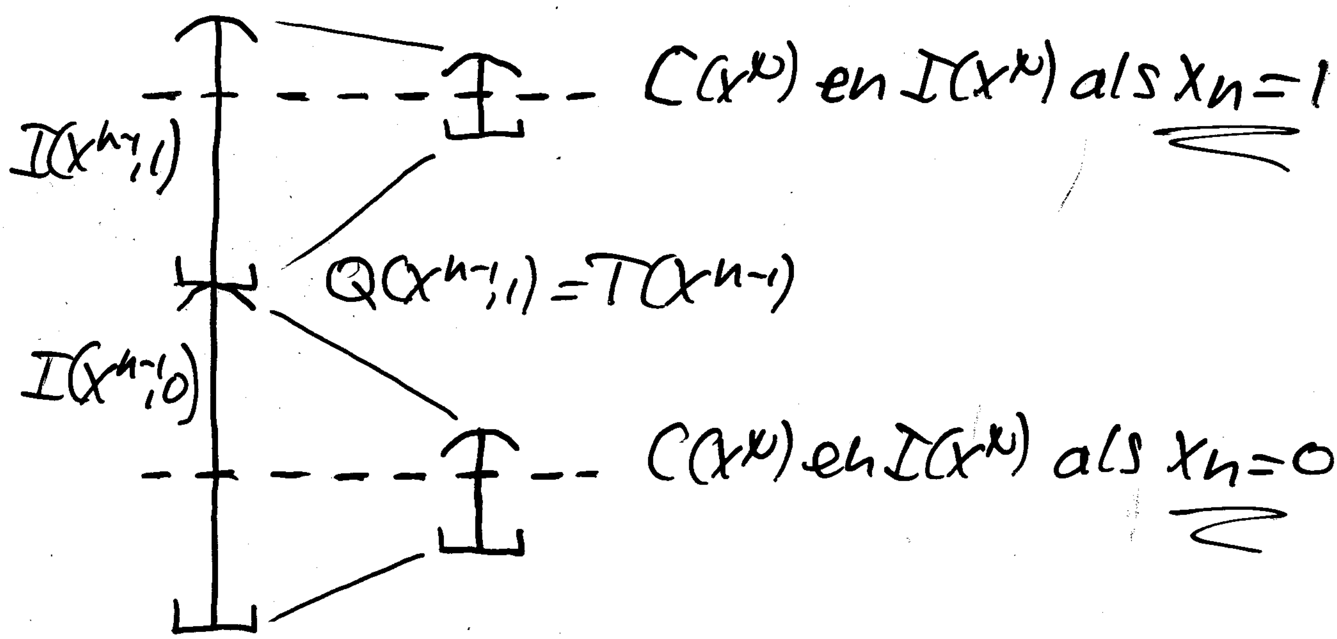
0

<u>Twee figuren:</u> $I(x^{n-1})$ wordt opgedeeld in twee deelintervallen: $I(x^{n-1};0)$ en $I(x^{n-1};1)$



Nu is duidelijk dat

$I(\phi) \supset I(x_1) \supset I(x_1 x_2) \supset \dots \supset I(x^N)$



Complexiteit: In verband met eenduidigheid zijn $Pr\{X_n=0 | X^{n-1}=x^{n-1}\}$ en $Pr\{X_n=1 | X^{n-1}=x^{n-1}\}$ BREUKEN.

Neem aan dat de noemer van deze breuken begrensd is (bijv. ≤ 255).

Dan neemt de opslagruimte voor $P(x^N)$ lineair toe met N.
voorbeeld: $Pr\{X_n=1|-\}=1/3$, $N=3$
$P(\phi)=\frac{1}{1}$, $P(1)=\frac{1}{3}$, $P(10)=\frac{2}{9}$, $P(100)=\frac{4}{27}$
Hetzelfde geldt voor $Q(x^N)$ overigens.
voorbeeld: $Pr\{X_n=1|-\}=1/3$, $N=3$
$Q(\phi)=\frac{0}{1}$, $Q(1)=\frac{2}{3}$, $Q(10)=\frac{6}{9}$, $Q(100)=\frac{18}{27}$

De rekentijd bij het bepalen van $P(x^N)$ en $Q(x^N)$ neemt kwadratisch toe met N.

Het bepalen van het codewoord $C(x^N)$ en ook de decodeertijd blijken kwadratisch met N toe te nemen.

Decodeervertraging: Eerst moet het hele bronrijtje verwerkt zijn alvorens $C(x^N)$ kan worden bepaald. Vertraging bedraagt dus N bronsymbolen. De decoder moet in principe ook het hele codewoord kennen om aan de slag te kunnen gaan.

Vaste registerlengte implementatie

(Rubin)

Encoderen en decoderen bestaan voor een groot deel uit het steeds updaten van $Q(x^n)$ en $P(x^n)$

Representatie: we representeren nu $Q(x^n)$ en $P(x^n)$ als
- binaire fracties
- die de onderstaande vorm hebben:

	$l(x^n)$	B	
$Q(x^n) =$	$.q_1 q_2 \cdots q_{l(x^n)}$	$-q(x^n)-$	$\cdots$
$P(x^n) =$	$.0\,0 \cdots 0$	$-p(x^n)-$	$\cdots$
	verandert niet meer	wordt nog aan gewerkt	nog niet aan begonnen

Omdat $Q(x^n) \approx Pr\{X^n < x^n\}$ en $P(x^n) \approx Pr\{X^n = x^n\}$ door deze representatie, zal de gemiddelde codewoordlengte wat groter worden.

Bij B (registerlengte) $= 16$ is dit al niet meer waar te nemen.

Eisen aan registerinhouden $q(x^n)$ en $p(x^n)$:

Aangezien de eindwaarde

$$Q(x^N) \in I(x^N) \subset I(x^n)$$

geldt dat

$$Q(x^n) \leq Q(x^N) < Q(x^n) + P(x^n)$$

Als nu $q(x^n) + p(x^n) \leq 2^B$ zijn zal dus $q_1 q_2 \cdots q_{L(x^n)}$ niet meer kunnen veranderen. ⊛

voorbeeld:

$$\begin{array}{rll}
 & & B=4 \\
Q(x^n) = & .011010 & \boxed{0011} = q(x^n) \\
P(x^n) = & .000000 & \boxed{1011} = p(x^n) \\
 & & \hphantom{\boxed{1011}} + \\
\hline
Q(x^n)+P(x^n) = & .011010 & 1110
\end{array}$$

Verder moet $I(x^n)$ deelbaar blijven dus moet $P(x^n) > 0$ zijn ⊛⊛

Combineren van ⊛ en ⊛⊛ levert

$$0 \leq q(x^n) < q(x^n) + p(x^n) \leq 2^B \quad (C_1)$$

Merk op dat $p(x^n) = 2^B$ kan zijn en dan niet meer in het register past. We rekenen daarom met $p(x^n) - 1$. Hier vergeten we dit even.

Schalen: Het definitief worden van een nieuwe binaire digit in $Q(x^n)$.

A: als $q(x^n)+p(x^n) \leq 2^{B-1}$ dan moet $q_{L(x^n)+1} = 0$ zijn!

Nu: $L(x^n) := L(x^n)+1$; $q(x^n) := 2 * q(x^n)$
$p(x^n) := 2 * p(x^n)$

voorbeeld: $Q(x^n) = .011010\,\boxed{0011}$
$P(x^n) = .000000\,\boxed{0100}$ +

$Q(x^n)+P(x^n) = .011010\;0111$

wordt: $Q(x^n) = .011010\underline{0}\,\boxed{0110}$
$P(x^n) = .0000000\,\boxed{1000}$

B: als $q(x^n) \geq 2^{B-1}$ dan moet $q_{L(x^n)+1} = 1$ zijn!

Nu: $L(x^n) := L(x^n)+1$; $q(x^n) := 2*(q(x^n)-2^{B-1})$
$p(x^n) := 2 * p(x^n)$

voorbeeld: $Q(x^n) = .011010\,\boxed{1001}$
$P(x^n) = .000000\,\boxed{0101}$ +

$Q(x^n)+P(x^n) = .011010\;1110$

wordt: $Q(x^n) = .011010\underline{1}\,\boxed{0010}$
$P(x^n) = .0000000\,\boxed{1010}$

Merk op dat na schalen (C_1) weer geldt.

Na het schalen geldt ook dat

$$g(x^n) < 2^{B-1} < g(x^n) + p(x^n) . \qquad (C_2)$$

Dit impliceert dat $p(x^n) \geq 2$ is, dus dat $I(x^n)$ deelbaar is.

Opdelen: Bepalen van het nieuwe interval.

Ook nu geldt:

$$g(x^n) = g(x^{n-1}) + p(x^{n-1}) \cdot Pr\{X_n < x_n \mid X^{n-1} = x^{n-1}\}$$

$$p(x^n) = p(x^{n-1}) \cdot Pr\{X_n = x_n \mid X^{n-1} = x^{n-1}\}$$

Aangezien $p(x^{n-1}) \cdot Pr\{X_n = \cdots\}$ meestal geen geheel getal is passen we $Pr\{X_n = \cdots\}$ aan.

Om de gemiddelde codewoordlengte niet te veel te laten toenemen moet deze aanpassing gering zijn en zodanig dat de beide deelintervallen lengte ≥ 1 krijgen.
Omdat na het schalen (C_2) geldt is $p(x^{n-1}) \geq 2$ dus kan dit.

voorbeeld: $Q(x^{n-1}) = .011010\,\boxed{0011}$
$P(x^{n-1}) = .000000\,\boxed{1011}$

$$Pr\{X_n = 1 / X^{n-1} = x^{n-1}\} = 1/3$$

Omdat $p(x^{n-1}) = 11$ rekenen we verder met $Pr\{X_n = 1 / X^{n-1} = x^{n-1}\} = 4/11$. Dan

$x_n = 0$: $q(x^{n-1}, 0) = q(x^{n-1}) = 3$
$p(x^{n-1}, 0) = p(x^{n-1}) \cdot 7/11 = 7$

dus $Q(x^{n-1}, 0) = .011010\,\boxed{0011}$
$P(x^{n-1}, 0) = .000000\,\boxed{0111}$

$x_n = 1$: $q(x^{n-1}, 1) = q(x^{n-1}) + p(x^{n-1}) \cdot 7/11 = 10$
$p(x^{n-1}, 1) = p(x^{n-1}) \cdot 4/11 = 4$

dus $Q(x^{n-1}, 1) = .011010\,\boxed{1010}$
$P(x^{n-1}, 1) = .000000\,\boxed{0100}$

Na het opdelen geldt (C_1) weer.

Het encoderen:

- initialiseren: $q(\phi) = 0$ $\quad l(\phi) = 0$
  $p(\phi) = 2^B$
- voor $n = 1, \cdots, N$: schaal indien mogelijk deel op en ga verder met x_n - deelinterval
- beeindiging: $c(x^N) = Q(x^N)$

(immers $Q(x^N) \in I(x^N)$,
hiermee wordt de codewoordlengte
$k(x^N) = l(x^N) + B$,
verder betekent dit dat $q_1, q_2, \cdots$
direct na bepaling al verstuurd of opgeslagen hadden mogen worden → lagere decodeer-vertraging)

Het decoderen:

De decoder bepaalt net als de encoder $q(x^n)$ en $p(x^n)$.

Nadat $x_1, x_2, \cdots, x_{n-1}$ bepaald zijn, schaalt de decoder indien mogelijk en vormt dan de drempel $T(x^{n-1}) = Q(x^{n-1}, 1)$.

voorbeeld: $Q(x^{n-1}) = .011010\ \boxed{0011}$
$P(x^{n-1}) = .000000\ \boxed{1011}$
$Pr\{X_n = 1 \mid X^{n-1} = x^{n-1}\} = 1/3$

$T(x^{n-1}) = Q(x^{n-1}, 1) = .011010\ \boxed{1010} = t(x^{n-1})$

Laat $C(x^N) = Q(x^N) = .011010\ \boxed{1100}\ xxxx \ldots$

$\boxed{1100} = c(x^{n-1})$

Nu: zijn de eerste $l(x^{n-1})$ binaire digits van $T(x^{n-1}) = Q(x^{n-1}, 1)$ en $C(x^N) = Q(x^N)$ aan elkaar gelijk

Dus: $C(x^N) \geq T(x^{n-1}) \Leftrightarrow c(x^{n-1}) \geq t(x^{n-1})$

daarom $x_n = 0$ als $c(x^{n-1}) < t(x^{n-1})$
$x_n = 1$ als $c(x^{n-1}) \geq t(x^{n-1})$

voorbeeld $c(x^{n-1}) = 12$, $t(x^{n-1}) = 10$
dus $x_n = 1$.

Merk op dat de decoder bij het schalen ook $c(x^{n-1})$ moet aanpassen:

als $c(x^{n-1}) \geq 2^{B-1}$

dan $c(x^{n-1}) := 2 * (c(x^{n-1}) - 2^{B-1}) + c_{next}$

anders $c(x^{n-1}) := 2 * c(x^{n-1}) + c_{next}$

Voorbeeld:

c_{next}

$c(x^N) = .011010 \boxed{1100} 1xxxx \cdots$

$c(x^N) = .0110101 \boxed{1001} xxxx \cdots$

Concluderend: decoderen is

— initialiseren: $a(\phi) = 0$ $\qquad l(\phi) = 0$

$b(\phi) = 2^B$

$c(\phi) = \sum_{b=1,B} c_b 2^{B-b}$

— voor $n = 1, N$: schaal indien mogelijk
bepaal $t(x^{n-1})$ en vergelijk
met $c(x^{n-1}) \rightarrow x_n$
deel daarna op en
ga verder met x_n-
deelinterval.

Het blijkt dat de decoder met een stuk van het codewoord al aan de gang kan!

Complexiteit van de vaste registerlengte implementatie:

behodigde opslagruimte:

enkele registers van B bin. digits.

behodigde rekentijd:

lineair met N (meer precies is echter vaste tijd per bronsymbool plus vaste tijd per codesymbool)

Decodeervertraging: B codesymbolen.

VRAAG: Laat $\Pr\{X_n = 1 \mid X^{n-1} = x^{n-1}\} = 1/3$, $n = 1, 2, 3, 4$. Encodeer het bronrijtje $x^4 = 1010$ met het Elias algorithme en ook met het Rubin algorithme ($B=4$).
Decodeer x^4 met het Eliasalgorithme als $C(x^4) = .0001001$
Decodeer ook x^4 met het Rubinalgorithme als $C(x^4) = .10101010$ ($B=4$)

Wanneer is een arithmetische code bruikbaar?

— Als we de kans $Pr\{X_n=1 / X^{n-1}=x^{n-1}\}$ kennen voordat x_n geëncodeerd (gedecodeerd) moet worden.

— Als we een goeie benadering van de kans $Pr\{X_n=1 / X^{n-1}=x^{n-1}\}$ hebben voordat x_n geëncodeerd (gedecodeerd) moet worden.

— Als we rekencapaciteit hebben.
(optellen, vermenigvuldigen)

Literatuur:

F. Jelinek, "Probabilistic Information Theory," McGraw-Hill, 1968, pp. 476-489.
(Elias algorithme)

P. Elias, "Universal codeword sets and representations of the integers," IEEE Trans. Inform. Theory, vol. IT-21, pp. 194-203, 1975.

F. Rubin, "Arithmetic Stream Coding using fixed precision registers," IEEE Trans. Inform. Theory, vol. IT-25, pp. 672-675, 1979.