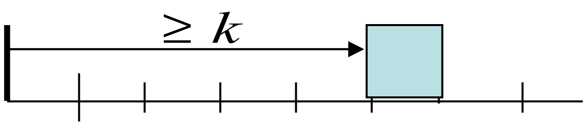
Dr. Wilhelm, Claus: Symmetrischer Binärkanal mit Gedächtnis
A-Model, insbesondere für M2M-Communication; Beispiele für Einsteiger und Studenten, Testbeispiele mit Excel;
*Grundlagen siehe **downloads**!*

Beispiel 1
Gegeben sei das **"wireless short-packet (WSP) protocol"** ISO/IEC 14543-3-10 für M2M (Machine to Machine) Communication, 125 kbit/s data rate, 868,3 MHz.
Jeder Frame wird dreimal in einer Richtung übertragen, ohne feedback.
Der Header started mit 8 PRE-bits zur Synchronisation, gefolgt von 4 SOF-bits als Startsynchronisation: 101010101001.

Wenn innerhalb dieser 12 bits ein oder mehr Fehler auftreten, kann der frame nicht starten.
Gesucht ist die Wahrscheinlichkeit u(k) für den nächsten Bitfehler, der nach k = 12 oder mehr fehlerfreien bits folgt;
Denn die Wahrscheinlichkeit, dass der Frame nicht started ist: $1 - u(k = 12)$



$u(0) = 1; u(1) = \alpha \cdot e^{-\beta}$; wobei $e^{-\beta} = (1 - p_e^{1/\alpha})$;
mit der Bitfehlerwahrscheinlichkeit $p_e < 0.01$
und in der Praxis mit einem Bündelungsfaktor $(1 - \alpha)$; $0.5 < \alpha < 0.95$
(für den BSC-Kanal, ohne Gedächtnis ist $(1 - \alpha) = 0$;).
Mit Excel , kann recursiv wie folgt gerechnet werden:

$$u(k+1) = u(k)\frac{k+\alpha}{k+1}e^{-\beta}$$

Gegeben sei:
a) pe = 0.001; α = 0.9
b) pe = 0.01; α = 0.7 (starke Bündelfehler)

Ergebnis::

$$1 - u(k = 12; pe = 0.001; \alpha = 0.9) = 0.27685541$$

$$1 - u(k = 12; pe = 0.01; \alpha = 0.7) = 0.66283848$$

$$0.66283848^3 = 0.2912213$$

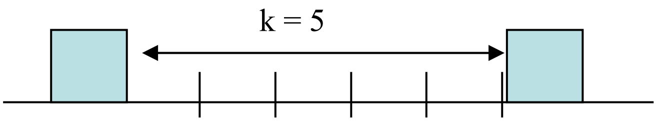
$$P(kein - start; 3mal) = 0.66283848^3 = 29\%$$

Trotz Dreifachübertragung startet der Frame in 29% aller Fälle nicht oder falsch..

Beispiel 2

Gesucht ist die Wahrscheinlichkeit v(k) daß nach dem letzten Bitfehler eine Lücke der Länge k folgt:

a) k = 0 bits (Doppelfehler);
b) k = 5 bits
c) k = 48 bits ?
mit pe = 0.001; α = 0.9



wobei

$$v(k) = u(k) - u(k+1); u(0) = 1;$$

Entsprechend Beispiel 1 kann das Ergebnis recursiv berechnet werden.
Ergebnisse::

v(0) = *0.10041774*
v(5) = *0.01349111*
v(48) = *0.00155455*

Beispiel 3

Gesucht ist die Wahrscheinlichkeit $p_b(n)$, daß ein Block der Länge n fehlerhaft ist.

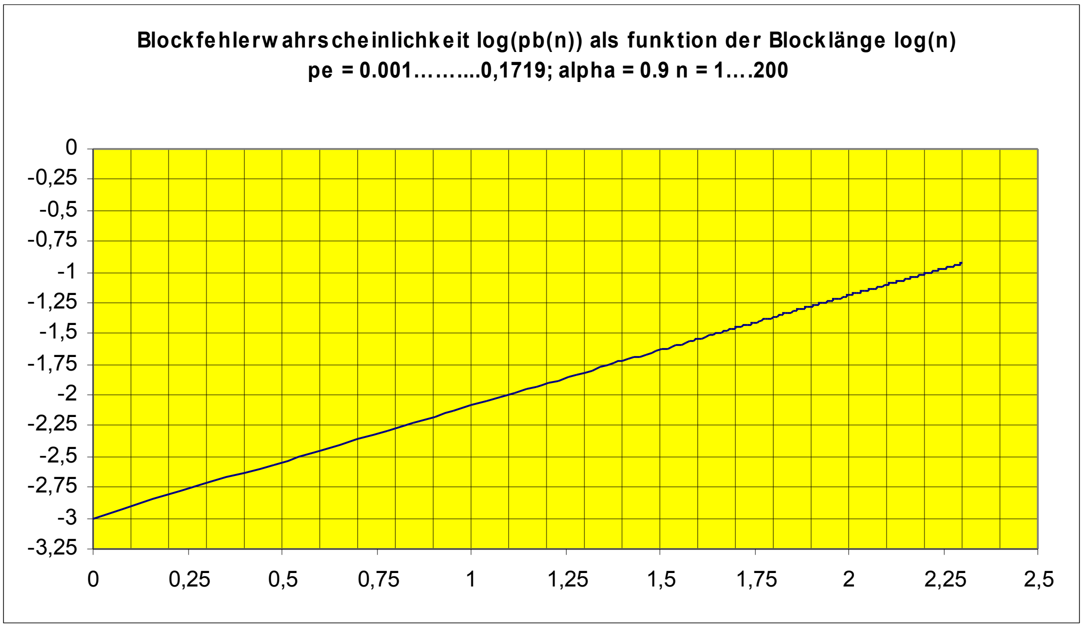
mit pe = 0.001; α = 0.9 ; $e^{-\beta} = 0.99953584$

$$p_b(n) = p_e \sum_{k=0}^{n-1} u(k); \quad u(k+1) = u(k)\frac{k+\alpha}{k+1}e^{-\beta}; u(0) = 1; u(1) = \alpha \cdot e^{-\beta};$$

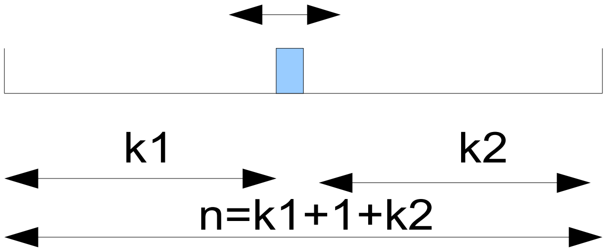
$$p_b(n+1) = p_b(n) + p_e \cdot u(n);$$

$$p_b(1) = p_e$$

Die Ergebnisse: *pb(1) = 0.001; pb(5) = 0.00438348; pb(48) = 0.03350905*



Beispiel 4:
Wir suchen die Wahrscheinlichkeit für alle Einzelfehler im Block der Länge *n*,
also die Wahrscheinlichkeitssumme für ***n*** verschiedene Einzelfehlermuster:



$$p(n+1;g=1) = p(n;g=1)\cdot\frac{(2\alpha+n-1)}{n}\cdot e^{-\beta}$$

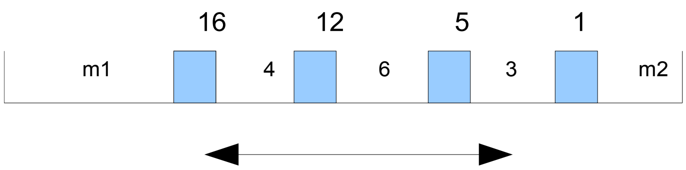
mit $p(n=1;g=1) = p_e = 0.001;\ \alpha = 0.9$

die Ergebnisse:
p(5;g =1) = 0.00382320
p(31; g =1) = 0.01647384

Beispiel 5
Linearer Blockcode CCITT CRC 16
Gegeben sei ein ***(48 ; 16)*** Blockcode mit der Blocklänge ***n,*** und ***k = 16*** Kontroll-Bits sowie ***32*** Informations-Bit.
Das Generatorpolynom ist:

$$x^{16} + x^{12} + x^5 + 1$$

Ein fehlerhafter Block bleibt unerkannt falsch, wenn die Fehlerstruktur mit dem obigen Polynom identisch ist. Für diese kürzeste unerkannte Struktur gibt es 31 mögliche Positionen.
Insgesamt existieren $2^{16} = 65536$ solcher unerkannten Strukturen.



$$P(strukturCRC16; n=48) = p(31;g=1)\cdot v(4)\cdot v(6)\cdot v(3)$$

Es gibt 31 socher Strukturen im Block der Länge $n = 48$;
mit pe = 0.001; alpha $\alpha = 0{,}9$;
Das Ergebnis ist das Produkt folgender vier Faktoren:

P(31;g=1) = 0.01345740
v(3) = 0.02100726
v(4) = 0.01645272
v(6) = 0.01141715

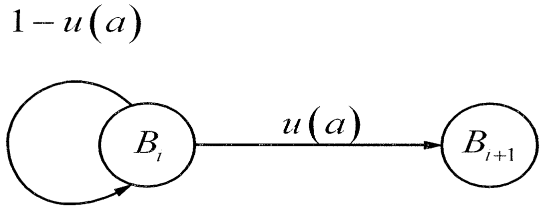
also: **5.3103867E-8;**
und die Wahrscheinlichkeit für einen fehlerhaften Block ist:

pb(48) = **3.350905E-2** (siehe Beispiel 3);

Beispiel 6

Verteilung der Fehleranzahl in Bursts :

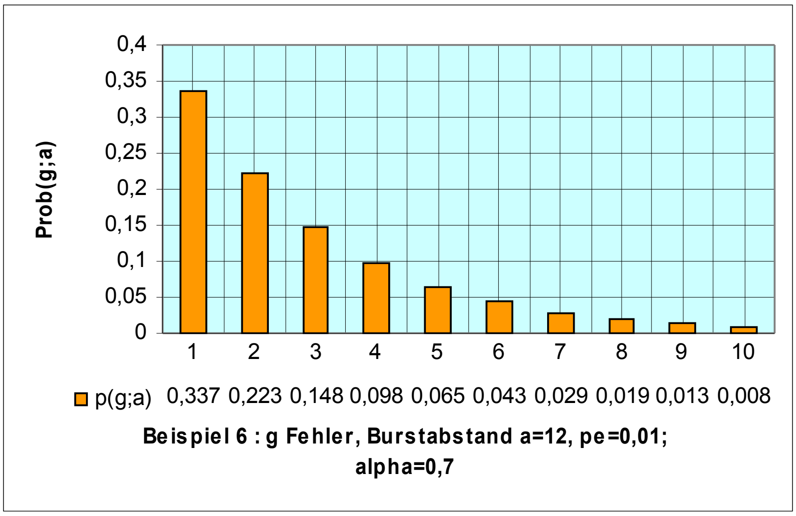
Ein Burst beginnt mit einem Bitfehler und endet, wenn a fehlerfreie Bits folgen
Die Anzahl der fehlerhaften Bit im Burst ist das Burstgewicht g.
Wenn der Burst-Parameter $a = 1$ ist, dann ist das Burstgewicht g gleich der Burstlänge l
Man berechne die Wahrscheinlichkeiten für **1, 2, 3, 4, 5, and 20** Fehler im Burst.
Gegeben: $p_e = 0.01; \alpha = 0.7; a = 12$



$$P(g;a) = u(a) \cdot \left[1 - u(a)\right]^{g-1}$$

$u(a = 12)$ wird rekursiv berechnet wie im **Beispiel 1**

$u(12) = 1 - 0.66283848 = 0.33716152$



Ergebnisse:

$P(g = 1;12) = 0.33716152; P(g = 2;12) = 0.22348363;$

$P(g = 3;12) = 0.14813355; P(g = 4;12) = 0.09818862$

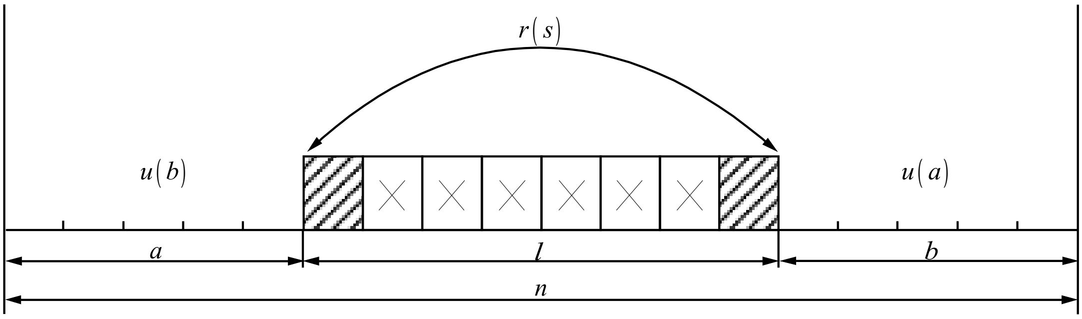
$P(g = 5;12) = 0.06508319; P(g = 20;12) = 0.0001632834$

Beispiel 7

Verteilung $p(l;n)$ für Bursts der Länge l in Blöcken der Länge *n = 5* bits

Für alle l = 1,2, 3,….***n;*** Bitfehlerwahrscheinlichkeit $p_e = 0.0008488$;

Bündelungsfaktor $(1-\alpha) = 0.16; \alpha = 0.84$; and $e^{-\beta} = (1-p_e^{1/\alpha}) = 0.9997792931\cdot$



Die Formel dafür ist abgeleitet in ***downloads „Kanalmodell Teil 1"*** ***(73)***

$$p(l;n) = p_\mathrm{e} \frac{2\alpha\,(2\alpha+1)\cdots(2\alpha+n-l-1)}{(n-l)!} \mathrm{e}^{-\beta(n-l)}$$

$$\times \left(1 - \frac{\alpha}{1!}\mathrm{e}^{-\beta} - \frac{\alpha\,(1-\alpha)}{2!}\mathrm{e}^{-2\beta} - \cdots - \frac{\alpha\,(1-\alpha)\cdots(l-2-\alpha)}{(l-1)!}\mathrm{e}^{-(l-1)\beta}\right).$$

Für die Berechnung gemäß (73), ist es besser, folgende Umformung zu benutzen:

$$p(l,n) = p_u(n-l+1)\cdot r_s(l-1)$$

und jeden dieser zwei Faktoren einzeln rekursiv zu berechnen,
wobei $p_u(n-l+1)$ die Wahrscheinlichkeit ist für einen Einzelfehler im Block
der Länge $n = a+b+l$, (siehe Beispiel 4).
Der Faktor $r_s(l-1)$ ist die Wahrscheinlichkeit für einen Burst der Länge $\boldsymbol{l}$.

Vor der Programmierung in einer höheren Sprache, wie z.B ***Matlab or C++****, bietet sich die folgende Testlösung an, die in Excel programmierbar ist, was hilfreich ist, um die zur Lösung angewendete Methode zu verstehen.*

Zunächst berechnen wir recursiv:

$$p_u(l=5;n=5) = p_e$$

$$p_u(l=4;n=5) = p_u(5;5)\cdot(2\alpha/1)\cdot e^{-\beta}$$

$$p_u(l=3;n=5) = p_u(4;5)\cdot((2\alpha+1)/2)\cdot e^{-\beta}$$

$$p_u(l=2;n=5) = p_u(3;5)\cdot((2\alpha+2)/3)\cdot e^{-\beta}$$

$$p_u(l=1;n=5) = p_u(2;5)\cdot((2\alpha+3)/4)\cdot e^{-\beta}$$

Und ebenso:

$$r_s(l=1)=1$$

$$r_s(l=2)=r_s(l=1)-(\alpha\ /1!)\cdot e^{-\beta}$$

$$r_s(l=3)=r_s(l=2)-((\alpha\cdot(1-\alpha)/2!)\cdot e^{-2\beta}$$

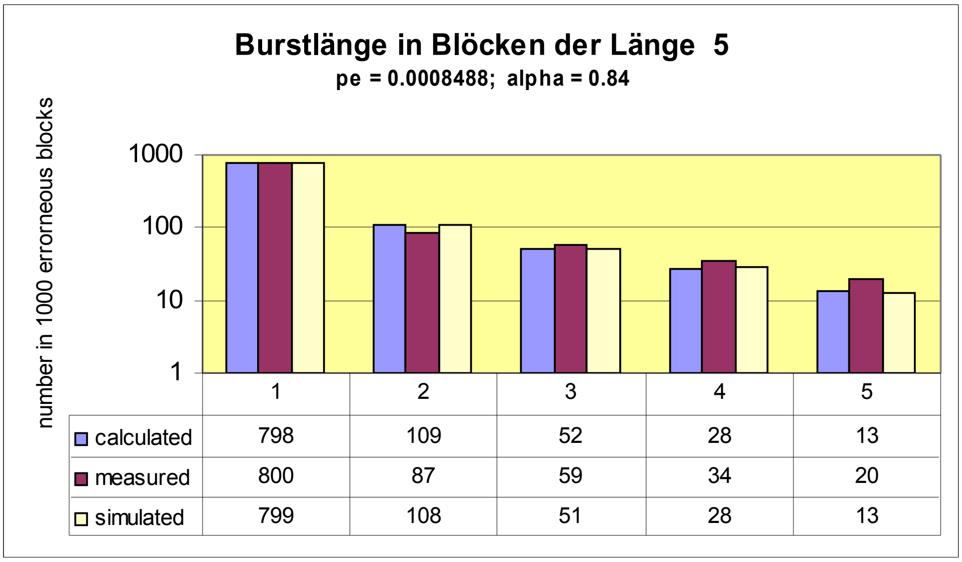
$$r_s(l=4)=r_s(l=3)-((\alpha\cdot(1-\alpha)\cdot(2-\alpha)/3!)\cdot e^{-3\beta}$$

$$r_s(l=5)=r_s(l=4)-((\alpha\cdot(1-\alpha)\cdot(2-\alpha)\cdot(3-\alpha)/4!)\cdot e^{-4\beta}$$

Nachfolgend die mit Excel berechneten Ergebnisse:

pe=0.0008488	alpha=0.84	n=5		pb(5)=		
length l	pu	rs	p(l;n=5) = pu*ps	**3,4334588506E-03**	measured	simulated
				10^3*p(l;n=5)/pb(5)		
1	0,0027399865	1,0000000000	2,7399865000E-03	798	800	799
2	0,0023423858	0,1601853938	3,7521599180E-04	109	87	108
3	0,0019099752	0,0930150535	1,7765644541E-04	52	59	51
4	0,0014256693	0,0670482543	9,5588637774E-05	28	34	28
5	0,0008488000	0,0530292775	4,5011250742E-05	13	20	13
		Summe=pb(5)	3,4334588257E-03	1000	1000	999

Man kann sehen, dass größere Längen häufiger auftreten, weil in diesem Modell das Gedächtnis nicht über den letzten Bitfehler hinaus reicht. Anderseits zeigt der χ^2-Test keine signifikante Abweichung.
*Jedoch zwecks Vergleich von Codes oder Prozeduren ist das A-Modell ausreichend und in den meisten Fällen einfach zu benutzen. Der Vorteil ist, **eine erste einfache Methode** zur Berechnung der Summen von Ereignissen zu haben.*



Beispiel 8

Verteilung $p(l;n)$ der Burstlänge l in Blöcken der Länge *n = 15* bits

mit l = 1,2, 3,….***n;*** Bitfehlerwahrscheinlichkeit $p_e = 0.0008; \alpha = 0.8; e^{-\beta} = 0.9998548645$

Man kann dies in derselben Weise rekursiv entwickeln wie im Beispiel 7, ausgehend von der Basisformel (73)

Nachfolgend die mit Excel berechneten Ergebnisse:

pe=0.00085		pb(15)=		5806 err.blocks
length l	p(l;n=15)	**7,9153586826E-03**	calculated	simulated
		p(l;n=15)/pb(15)	5806*p(l;n=15)/pb(15)	
1	0,0047821493	0,6041607831	3508	3469
2	0,0009177902	0,1159505509	673	750
3	0,0005267615	0,0665492899	386	355
4	0,0003681637	0,0465125732	270	293
5	0,0002794990	0,0353109709	205	195
6	0,0002216502	0,0280025466	163	168
7	0,0001802212	0,0227685450	132	128
8	0,0001485954	0,0187730469	109	111
9	0,0001232668	0,0155731161	90	267
10	0,0001021736	0,0129082716	75	(sum of 9+10+11+12)
11	8,3988972658E-05	0,0106108865	62	
12	6,7771249004E-05	0,0085619934	50	
13	5,2749022590E-05	0,0066641355	39	70
14	3,8106656360E-05	0,0048142678	28	(sum of 13+14+15)
15	2,2471904462E-05	0,0028390254	16	
Sum total	7,9153587051E-03	1,0000000028E+00	5806	5806