Technische Universität München
Lehrstuhl Informatik VIII
Prof. Dr.-Ing. Georg Carle
Prof. Dr.-Ing. Wolfgang Utschick
Stephan M. Günther
Maximilian Riemensberger

Tutorials for Network Coding (IN3300)
Tutorial 1 – 2014/10/21

Problem 1 FEC with ARQ

Consider a simple wireless network consisting of two nodes s and t. Node s transmits packets of $l = 15\,808$ bit each. The channel has a bit error rate of $\epsilon = 10^{-4}$.

If a transmission of s is successfully received by t, an acknowledgement is triggered and sent back to s. We assume orthogonal scheduling, i. e, there are no additional losses due to collisions. Further we assume that acknowledgements do not get lost lost.

a)* Let X be a random variable that counts the number of bit errors in a given packet. Determine the probability for a successful transmission, i. e., $\Pr[X = 0]$.

$X \sim \mathrm{Bin}(l, \epsilon)$ and therefore

$$\Pr[X = i] = \binom{l}{i}\epsilon^i(1-\epsilon)^{l-i} \text{ and}$$
$$\Pr[X = 0] = (1-\epsilon)^{1976\cdot 8} = 20{,}58\,\%.$$

b) Let T denote a random variable that counts the number of transmissions until a packet is acknowledged. Determine $\Pr[T = i]$ and $\Pr[T \le i]$ in general and for $i = 7$.

$T \sim \mathrm{Geo}(p)$ with $p = \Pr[X = 0]$ and therefore

$$\Pr[T = i] = (1-p)^{i-1}p,$$
$$\Pr[T \le i] = \sum_{m=1}^{i} \Pr[T = m] = 1 - (1-p)^i, \text{ and}$$
$$\Pr[T \le 7] = 80{,}07\,\%$$

c) Determine the expectation $\mathrm{E}[T]$, i. e., the average number of transmissions that are needed until successful reception.

$T \sim \mathrm{Geo}(p)$ with $p = \Pr[X = 0]$ and therefore

$$\mathrm{E}[T] = \frac{1}{p} = 4.86.$$

To secure transmissions node s now employs a FEC code which maps source symbols of $k = 247\,\text{bit}$ to coded symbols of $n = 255\,\text{bit}$. The code is able to detect and correct a single bit-error in each coded symbol.

d) Determine the probability that a single symbol can be recovered at the receiver.

$$\begin{aligned}\Pr[X \le 1] &= \sum_{i=0}^{1} \binom{n}{i} \epsilon^i (1-\epsilon)^{n-i} \\ &= (1-\epsilon)^n + n\epsilon(1-\epsilon)^{n-1} \\ &= 99{,}97\,\%\end{aligned}$$

e) Let Z count the number of incorrect transmitted symbols. Determine the probability for a successful transmission during the first attempt for the whole packet if FEC is used.

A packet is split into $m = \frac{l \cdot 8}{k} = 64$ symbols. The probability that an individual symbol can be recovered is $q = 99{,}97\,\%$ and the error probability is therefore $1-q$. Then we have $Z \sim \text{Bin}(m, 1-q)$ and therefore

$$\Pr[Z = i] = \binom{m}{i}(1-q)^i q^{m-i} \text{ and}$$
$$\Pr[Z = 0] = q^m \approx 98{,}10\,\%.$$

Problem 2 Linear dependency of random vectors

Let $\boldsymbol{c} \in F_q^n[x]$ denote coding vectors which are drawn independently and uniformly distributed. Coding vectors are assembled to a coding matrix $\boldsymbol{C} = [\boldsymbol{c}_1 \ \ldots \ \boldsymbol{c}_m] \in F_q^{n\times m}[x]$ at the receiver. The receiver is able to decode if $\operatorname{rank} \boldsymbol{C} = n$. Let ρ_{mn}^k denote the probability that $\operatorname{rank} \boldsymbol{C} = k \le n$ after receiving $m \ge k$ coding vectors.

a)* Determine the probability ρ_{1n}^1, i. e., the probability to draw a random vector $\boldsymbol{c} \ne \mathbf{o}$.

There is a total of q^n different vectors in $F_q^n[x]$. The probability to draw one specific vector is thus q^{-n}. Consequently we have

$$\begin{aligned}\rho_{1n}^1 &= 1 - \rho_{1n}^0 \\ &= 1 - q^{-m}.\end{aligned}$$

b) Determine the probability ρ_{2n}^2, i. e., two random vectors are linear independent.

When we draw $\boldsymbol{c}_1$, there are $q^n - 1$ possible choices. Given a specific $\boldsymbol{c}_1 \ne \mathbf{o}$ the number of possible linear combinations that can be formed by $\boldsymbol{c}_1$ only is obviously q. Therefore we must not draw one of those q vectors for $\boldsymbol{c}_2$, which leaves $q^n - q$ valid choices. Therefore we have

$$\begin{aligned}\rho_{2n}^2 &= \frac{q^n-1}{q^n}\frac{q^n-q}{q^n} \\ &= \frac{(q^n-1)(q^n-q)}{q^{2n}} \\ &= \prod_{k=0}^{1}\left(1-q^{-n+k}\right).\end{aligned}$$

c) Determine the probability ρ_{mn}^m for $m \leq n$.

Observing that we can form a total of q^2 linear combinations from two linear independent vectors $\boldsymbol{c}_1$ and $\boldsymbol{c}_2$, we can conclude that there are q^k linear combinations from k linear independent vectors. This leaves $q^n - q^k$ linear independent vectors for $0 \leq k \leq n$. Therefore we have

$$\begin{aligned}\rho_{mn}^m &= \frac{q^n-1}{q^n}\frac{q^n-q}{q^n}\cdot\ldots\cdot\frac{q^n-q^{m-1}}{q^n} \\ &= \frac{(q^n-1)(q^n-q)\cdot\ldots\cdot(q^n-q^{m-1})}{q^{mn}} \\ &= \prod_{k=0}^{m-1}\frac{q^n-q^k}{q^n} = \prod_{k=0}^{m-1}\left(1-q^{-n+k}\right).\end{aligned}$$

d) Determine the probability ρ_{mn}^n for $m \geq n$.

Since $\operatorname{rank}\boldsymbol{C} = \operatorname{rank}\boldsymbol{C}^T$, we can consider $\boldsymbol{C}^T \in F_q^{m\times n}$. With $m \geq n$ we have the same situation as in c) except that m and n are swapped. This immediately gives

$$\rho_{mn}^m = \prod_{k=0}^{n-1}\left(1-q^{-m+k}\right).$$

1 Maß beer for the first one who comes up with an argument similar to a)–c).

Let X denote a random variable counting the number of random vectors $\boldsymbol{c}_k \in F_q^n[x]$ drawn until the matrix $\boldsymbol{C} = [\boldsymbol{c}_1 \ldots \boldsymbol{c}_m] \in F_q^{n\times m}[x]$ has rank n. The probability for $X < n$ is obviously 0. For $m > n$ the probability is given by $\rho_{m-1,n}^n$ and thus

$$\Pr[X < m] = \begin{cases} 0 & m \leq n, \\ \rho_{m-1,n}^n & m > n. \end{cases}$$

e)* Derive $\mathrm{E}[X]$ for $n = 32$ and $q \in \{2, 4, 16, 256\}$. As far as we know $\mathrm{E}[X]$ has no closed form. Simplify the expression as much as possible and then use Matlab to determine numerical results.

Hint: $\mathrm{E}[X] = \sum_{m=1}^{\infty}\Pr[X \geq m]$.

$$\begin{aligned}\mathrm{E}[X] &= \sum_{m=1}^{\infty} \Pr[X \geq m] = \sum_{m=1}^{\infty} (1 - \Pr[X < m]) \\ &= n + \sum_{m=n+1}^{\infty} \left(1 - \rho_{m-1,n}^{n}\right) = n + \sum_{m=n}^{\infty} \left(1 - \rho_{m,n}^{n}\right)\end{aligned}$$

Numerical results:

q	$\mathrm{E}[X]$	# linear dependent packets
2	17.60	1.60
4	16.42	0.42
16	16.10	0.07
256	16.00	0.00

- The chance to draw linear dependent vectors reduces significantly in q.
- For $q = 256$, the more exact result is 0.0039 excess packets per generation of $n = 16$.
- These values are widely independent of n and only change for very small n, i. e., $n < 8$.

You should try the Matlab scripts provided in the Git repository. Plot the probabilities for different n and q.