

1. gyakorlat: Jelek leírása

1.1. Az autokorrelációs függvény haszna

Tekintsük az ún. előrejelzési feladatot: megfigyeljük a ξ jel értékét a t_1 időpillanatban, és a megfigyelt érték, ξ_{t_1} alapján becsülni kívánjuk a jel értékét a t_2 időpillanatban. Ha a legegyszerűbb felépítésű (lineáris) becslésre szorítkozunk, és a becslés hibáját az ún. négyzetes középhibával értékeljük, akkor a legjobb becslési szabályt éppen a jel autokorrelációs függvényének ismeretében alkothatjuk meg.

E példában ui. a lineáris becslés a

$$\mu = a \cdot \xi_{t_1}$$

szabály alkalmazását jelenti, valamilyen a érték mellett. A becslés négyzetes középhibája alatt az

$$E = M(\xi_{t_2} - \mu)^2$$

nemnegatív számot értjük, amely a függvénye:

$$E = M(\xi_{t_2})^2 - 2a \cdot M(\xi_{t_1} \xi_{t_2}) + a^2 \cdot M(\xi_{t_1})^2.$$

A (négyzetes középben) legjobb lineáris becslés paramétere az az a érték, amelyre E minimális, ez az érték

$$a_0 = \frac{M(\xi_{t_1} \xi_{t_2})}{M(\xi_{t_1}^2)}.$$

Megállapítható, hogy a legjobb becslés paraméterét a jel autokorrelációs függvénye csakugyan meghatározza, csakúgy, mint a legjobb becslés négyzetes középhibáját. Ha

$$L(t_1, t_2) = M(\xi_{t_1} \xi_{t_2}),$$

akkor

$$a_0 = \frac{L(t_1, t_2)}{L(t_1, t_1)} \text{ és } E_0 = L(t_2, t_2) \left[1 - \frac{L^2(t_1, t_2)}{L(t_1, t_1) \cdot L(t_2, t_2)} \right].$$

Fontos tanulságok:

1/ Az előrejelzési feladat igényes megoldásához *szükséges* ismeret a folyamat autokorrelációs függvénye. Ez az ismeret ugyanakkor elegendő is *lehet* az efféle feladatok megoldásához.

2/ Tekintve, hogy a négyzetes középhiba semmiképpen sem lehet negatív, E_0 sem lehet negatív. Ezért bizonyosan igaz, hogy

$$L^2(t_1, t_2) \leq L(t_1, t_1) \cdot L(t_2, t_2),$$

ami persze nem más, mint a Cauchy, Schwartz, Bunyakovszki féle egyenlőtlenség körülményeinkre szabott változata.

3/ Ha a folyamat stacionárius, akkor $L(t_1, t_2) = R(t_1 - t_2)$, (továbbá $M(\xi_t) = M(\xi_0) = m$, állandó, de ez most nem érdekes), és így

$$a_0 = \frac{R(t_2 - t_1)}{R(0)} \cdot \text{és} E_0 = R(0) \cdot \left[1 - \frac{R^2(t_2 - t_1)}{R^2(0)} \right],$$

valamint

$$R^2(t_2 - t_1) \leq R^2(0).$$

4/ Ha a becslés jóságát nem a négyzetes középhibával mérnők, a legjobb becslés megalkotásához nem lenne elegendő a jel autokorrelációs függvényének ismerete. Sőt, ilyen kevés ismeret birtokában még a hiba mértékének kiszámítására sem igen nyílik lehetőség. Gyakran hasznos, értelmes hibakritérium lehetne pl. az ún. abszolút középhiba,

$$D = M \left| \xi_{t_2} - \mu \right|,$$

amelynek meghatározása általában az autokorrelációs függvénynél mélyebb ismereteket igényel.

1.2. Az autokorrelációs függvény és a spektrális sűrűségfüggvény

A stacionárius folyamatok autokorrelációs függvényének Fourier transzformáltját - ha létezik - a folyamat spektrális sűrűségfüggvényének nevezzük. Ez valós folyamatoknál nemnegatív, páros függvény. Ha a ξ jelet, amelynek spektrális sűrűségfüggvénye $s_\xi(\cdot)$, szűrjük egy $H(\cdot)$ átviteli függvényű szűrővel akkor a szűrő η kimenő jelének $s_\eta(\cdot)$ spektrális sűrűségfüggvénye

$$s_\eta(f) = |H(f)|^2 \cdot s_\xi(f).$$

Igaz továbbá, hogy a spektrális sűrűségfüggvény integrálja a jel átlagteljesítményét szolgáltatja:

$$M(\eta_t^2) = R(0) = \int_{-\infty}^{\infty} s_\eta(f) df.$$

A spektrális sűrűségfüggvény e tulajdonságai abból következnek, hogy őt az autokorrelációs függvény Fourier transzformáltjaként definiáltuk. Ugyanakkor e tulajdonságok lehetővé teszik, hogy a spektrális sűrűségfüggvénynek szemléletes fizikai jelentést tulajdonítsunk, s ez a kép gyakran segíti egyes jelenségek, összefüggések megértését.

Tételezzük fel ugyanis, hogy a ξ jelet egy keskeny, Δ sáv szélességű, amúgy "ideális" sáv szűrővel szűrjük. Pontosabban, a szűrő átviteli függvénye:

$$H(f) = \begin{cases} 1, & \text{ha } f \in [f_0, f_0 + \Delta] \\ 1, & \text{ha } f \in [-f_0, -(f_0 + \Delta)] \\ 0, & \text{egyébként} \end{cases}$$

Érdekes kérdés, mit mondhatunk ekkor a szűrő kimenő jelének átlagteljesítményéről. A szűrőn nyilván csak a jel azon szinuszos - koszinuszos összetevői jutnak át, amelyek frekvenciája a szűrő áteresztősávjába (f_0 és $f_0+\Delta$ közé) esik. A kérdés tehát egyrészt ezen jelösszetevők összteljesítménye. Másrészt viszont az η kimenő jel teljesítménye:

$$M(\eta_t^2) = R_\eta(0) = \int_{-\infty}^{\infty} s_\eta(f) df = \int_{-\infty}^{\infty} s_\xi \cdot |H(f)|^2 df$$

Kihasználhatjuk, hogy az integrandus mindkét tényezője páros függvény, továbbá, hogy egyikük az integrálás tartományának jelentős részhalmazain zérus. Így:

$$M(\eta_t^2) = 2 \cdot \int_{f_0}^{f_0+\Delta} s_\xi(f) \cdot |H(f)|^2 df = 2 \cdot \int_{f_0}^{f_0+\Delta} s_\xi(f) df .$$

Ha Δ elegendően kicsiny, és $s_\xi(\cdot)$ folytonos, akkor az integrálás középértéktételei szerint

$$M(\eta_t^2) \approx 2 \cdot s_\xi(f_0) \cdot \Delta .$$

Tehát: megalapozott az a szemléletes kép, miszerint a spektrális sűrűségfüggvény értéke az f_0 frekvencián az f_0 frekvencia egységnyi széles környezetében lévő szinuszos (és koszinuszos) jelösszetevők átlagteljesítményét adja meg.

1.3. Példa: a sávhatárolt fehér zaj

Egy jel (legyen pl. ξ) spektrális sűrűségfüggvénye

$$s_\xi(f) = \begin{cases} s_0, & \text{ha } |f| \leq B \\ 0, & \text{egyébként} \end{cases} .$$

Az ilyen jel *sávhatárolt* (hiszen B-nél nagyobb frekvenciájú összetevői nincsenek), és *fehér* (a nemzérus intenzitású jelösszetevők azonos erősségűek, s a fehér fény összetevői is ilyenek). Határozzuk meg e jel autokorrelációs függvényét!

Megoldás:

Az autokorrelációs függvény a spektrális sűrűségfüggvény inverz Fourier transzformáltja:

$$R_\xi(\tau) = \int_{-\infty}^{\infty} s_\xi(f) \cdot e^{j2\pi f\tau} df .$$

Most

$$R_\xi(\tau) = \int_{-B}^B s_0 \cdot e^{j2\pi f\tau} df = s_0 \cdot \left. \frac{e^{j2\pi f\tau}}{j2\pi\tau} \right|_{-B}^B = s_0 \cdot \frac{e^{j2\pi B\tau} - e^{-j2\pi B\tau}}{j2\pi\tau}$$

Felismerve a szinuszfüggvény Euler-féle alakját, írhatjuk, hogy

$$R_{\xi}(\tau) = 2Bs_0 \cdot \frac{\sin(2\pi B\tau)}{2\pi B\tau}.$$

Ez a függvény számos híradástechnikai modellben felbukkan. Lényeges tulajdonságai:

1/ Argumentumának növekedő (abszolút) értékeire annak reciprokával majorálhatóan tűnik el (tart a zérushoz). Ez azt jelenti, hogy a jel távoli mintái között is van számottevő kapcsolat.

2/ Zérus értéket vesz fel, valahányszor $\tau = k/(2B)$, ahol k egész, de nem zérus.

3/ A függvény globális maximuma $\tau = 0$ -ban van.

4/ Szépen látható, hogy páros függvény, de nem mindenütt pozitív függvény (viszont pozitív definit függvény, de ez más kérdés).

1.4. Példa: Exponenciálisan lecsengő autokorrelációs függvényű folyamat

Egy jel (legyen pl. ξ) autokorrelációs függvénye

$$R_{\xi}(\tau) = R_0 \cdot e^{-|\tau|/T}.$$

Határozzuk meg a jel spektrális sűrűségfüggvényét!

Megoldás:

A megadott autokorrelációs függvény Fourier transzformáltját kell képezni:

$$s_{\xi}(f) = \int_{-\infty}^{\infty} R_{\xi}(\tau) \cdot e^{-j2\pi f\tau} d\tau.$$

Most az autokorrelációs függvény szakaszonként eltérő képlettel írható le, ezért

$$s_{\xi}(f) = \int_{-\infty}^0 R_0 \cdot e^{\tau/T - j2\pi f\tau} d\tau + \int_0^{\infty} R_0 \cdot e^{-\tau/T - j2\pi f\tau} d\tau.$$

Az exponensben szereplő független változót kiemelve, az integrálásokat elvégezve:

$$s_{\xi}(f) = R_0 \frac{e^{(1/T - j2\pi f)\tau}}{1/T - j2\pi f} \Big|_{-\infty}^0 + R_0 \frac{e^{(-1/T - j2\pi f)\tau}}{-1/T - j2\pi f} \Big|_0^{\infty}.$$

A behelyettesítés után:

$$s_{\xi}(f) = R_0 \frac{1}{1/T - j2\pi f} + R_0 \frac{1}{1/T + j2\pi f}.$$

Elvégezve a kijelölt műveletet:

$$s_{\xi}(f) = R_0 \frac{2/T}{(1/T)^2 + (2\pi f)^2} = \frac{2TR_0}{1 + (2\pi fT)^2}.$$

Figyelemreméltó, hogy ez a spektrális sűrűségfüggvénye az elsőfokú aluláteresztővel megsűrűt szélessávú fehér zajnak, ha a szűrő határfrekvenciája

$$f_0 = \frac{1}{2\pi T}.$$

1.5. Példa: Visszhang hatása stacionárius jelre

Az η folyamat a zérus várható értékű, $R_\xi(\tau) = R_0 \cdot e^{-\alpha|\tau|}$ autokorrelációs függvényű, ξ stacionárius folyamat lineáris transzformáltja: $\eta_t = \xi_t - \xi_{t-T}$.

Határozzuk meg az η folyamat autokorrelációs függvényét, továbbá a ξ és az η folyamatok spektrális sűrűségfüggvényeinek hányadosát a frekvencia függvényében!

Megoldás:

Az autokorrelációs függvény:

$$L_\eta(t_1, t_2) = M(\xi_{t_1} - \xi_{t_1-T})(\xi_{t_2} - \xi_{t_2-T}) = \\ R_\xi(t_2 - t_1) - R_\xi(t_2 - t_1 + T) - R_\xi(t_2 - t_1 - T) + R_\xi(t_2 - t_1) = \\ 2R_\xi(t_2 - t_1) - R_\xi(t_2 - t_1 + T) - R_\xi(t_2 - t_1 - T)$$

Az autokorrelációs függvény csak az időkoordináták különbségétől függ, így a kimenő folyamat stacionárius. A kimenő jel autokorrelációs függvényének Fourier transzformáltja, azaz spektrális sűrűségfüggvénye:

$$s_\eta(f) = 2s_\xi(f) - s_\xi(f) \cdot e^{j2\pi fT} - s_\xi(f) \cdot e^{-j2\pi fT} =$$

$$s_\xi(f)(2 - 2\cos(2\pi fT)) = 4 \cdot s_\xi(f) \cdot \sin^2(\pi fT)$$

A keresett hányados tehát:

$$4 \cdot s_\xi(f) \cdot \sin^2(\pi fT).$$

Ki tudta volna ezt előre megmondani (és milyen megfontolással)?

1.6. Példa: Mekkora nyereség várható egy Wiener szűrőtől?

A jel spektrális sűrűségfüggvénye

$$s_\xi(f) = \begin{cases} A^2 s_0 (1 - |f|/B), & \text{ha } |f| < B \\ 0, & \text{egyébként} \end{cases}$$

a zaj pedig szélessávban állandó, s_0 értékű. Határozzuk meg a szűrt jel négyzetes középhibáját, ha szűrésre B sávszélességű ideális aluláteresztőt, illetve, ha Wiener szűrőt használunk!

Megoldás:

Az ideális aluláteresztőn átszivárgó zaj teljesítménye $2Bs_0$. A Wiener szűrő átviteli függvénye (a teljesség kedvéért érdemes felírni és felrajzolni is):

$$G(f) = \frac{A^2 s_0 (1 - |f|/B)}{s_0 + A^2 s_0 (1 - |f|/B)}.$$

A Wiener szűrés maradék négyzetes középhibája:

$$E = \int_{-\infty}^{\infty} \left(s_{\xi}(f) |1 - G(f)|^2 + s_{\nu}(f) |G(f)|^2 \right) df = \dots = \int_{-\infty}^{\infty} \frac{s_{\xi}(f) s_{\nu}(f)}{s_{\xi}(f) + s_{\nu}(f)} df.$$

Behelyettesítve:

$$E = 2 \int_0^B \frac{s_0 \cdot A^2 s_0 (1 - f/B)}{s_0 + A^2 s_0 (1 - f/B)} df = \dots = 2 B s_0 \left(1 - \frac{1}{A^2} \ln(1 + A^2) \right).$$

Az összevethető jelteljesítmény mindkét esetben a bemeneti jel hasznos összetevője, ez nyilván $A^2 s_0 B$. A teljesítményarányok összevetése mindemellett nem túl hasznos, hiszen elfedné azt a tényt, hogy Wiener szűrésnél a kimenő jel hasznos összetevője torz (lineáris torzítás van benne).

1.7. Példa: Az előkiemelés nyeresége

A jel és a zaj spektrális sűrűségfüggvénye legyen ugyanolyan, mint az előző feladatban! Határozzuk meg a legjobb előkiemelő átviteli függvényét, s számítsuk ki, mekkora jel-zaj viszony nyereségre számíthatunk alkalmazásával!

Megoldás:

Az előkiemelő amplitúdókarakterisztikájának négyzete:

$$|P(f)|^2 = c \cdot \frac{s_{\nu}(f)}{s_{\xi}(f)} = \frac{c}{\sqrt{A^2(1 - |f|/B)}}.$$

A jel-zaj viszony javulását meghatározó arány:

$$\frac{\int_{-\infty}^{\infty} s_{\xi}(f) df \cdot \int_{-\infty}^{\infty} s_{\xi}(f) df}{\left(\int_{-\infty}^{\infty} \sqrt{s_{\xi}(f) s_{\xi}(f)} df \right)^2} = \dots = \frac{A^2 s_0 B \cdot 2 s_0 B}{4 \cdot \left(\int_0^B A s_0 \sqrt{1 - |f|/B} \cdot df \right)^2}$$

Kiszámítva:

$$\frac{A^2 s_0 B \cdot 2 s_0 B}{4 \cdot \left(\int_0^B A s_0 \sqrt{1 - |f|/B} \cdot df \right)^2} = \frac{2 A^2 s_0^2 B^2}{4 \cdot (A s_0)^2 \left[B(-2/3)(1 - |f|/B)^{3/2} \Big|_0^B \right]^2} = \frac{9}{8}.$$

Megjegyzés: e a példáink azt mutatják, hogy ezek az eljárások nem kecsegtetnek átütő sikerrel. Ez azért van így, mert a feltételezett sűrűségfüggvények elég laposak, ezeket nem igen lehet tovább lapítani. Csúcsosabb sűrűségfüggvényeknél lényegesen nagyobb lehet a nyereség, elérheti a 10 dB nagyságrendet is.

2. gyakorlat: Mintavételezés, kvantálás, tömörítés

2.1. Jelek visszaállítása egyenközű mintáinak sorozatából

A feladat az $x(\cdot)$ jel visszaállítása, azaz $x(t)$ értékeinek képzése minden t időpillanatra, pusztán az $x_k = x(kT)$, $k = 0, \pm 1, \pm 2, \dots$ értékek, az ún. minták felhasználásával. A tanult szabály $x(t)$ értékét a

$$\hat{x}(t) = T \cdot \sum_k x_k \cdot h(t - kT)$$

összeggel becsüli. Ez a becslés műszakilag kényelmesen valósítható meg D/A átalakító és egy $h(\cdot)$ súlyfüggvényű (ún. visszaállító vagy interpoláló) szűrő segítségével. A visszaállított jel általában eltér az eredetitől, a becslésnek vannak hibái. A bekövetkező eltérések egy részéről szemléletes képet ad, ha megvizsgáljuk a visszaállított és az eredeti jel spektrumának kapcsolatát. E kapcsolatra igaz, hogy:

$$\hat{X}(f) = H(f) \cdot \sum_k X(f - kf_s) = H(f) \cdot X_p(f),$$

ahol f_s az ún. mintavételi frekvencia, T reciproka. Az $X_p(\cdot)$ függvényt a mintasorozat spektrumának nevezzük. Ha $X_p(\cdot)$ olyan, hogy minden f -re létezik k , amelyre

$$X_p(f) = X(f - kf_s)$$

igaz, akkor a mintasorozat spektrumát átlapolásmentesnek nevezzük (ilyenkor az X_p -t definiáló összegzésnek csakis legfeljebb egyetlen tagja vehet fel zérustól eltérő értéket az f frekvencián).

Ha a mintasorozat spektruma átlapolásmentes, akkor a

$$H(f) = \begin{cases} 1 & , ha X(f) \neq 0 \\ 0, & ha X_p(f) \neq 0 \text{ és } X(f) = 0 \\ * & \text{egyébként tetszőleges} \end{cases}$$

átviteli függvényű szűrő tökéletesen rekonstruálja az eredeti jelet.

2.2. A visszaállítás jellegzetes hibái

A minták véges pontosságú ábrázolása a forrása az ún. kvantálási zajnak (lásd az előadás, könyv anyagát).

A mintasorozat spektrumának átlapolásmentességét biztosítandó a mintavételezésnek kitett jelet eleve szűrjük. Ha ennek az ún. antialiasing (bemenő) szűrőnek nem elég nagy a zárósávi csillapítása, akkor a visszaállítás során - a szorosan vett átviteli sávon kívüli esetleges jelösszetevők következtében - ún. *alias* (ál) jelek keletkezhetnek.

A visszaállító szűrő zárósávi viselkedése lehet a felelős a rekonstruált jelben megjelenő, az eredeti jel sávján kívüli, *szivárgó* jelösszetevők megjelenéséért.

A bemenő és a kimenő szűrő áteresztősávja egyaránt lineáris torzítást okoz a visszaállított jelben.

2.3. Példa: szivárgás és aliasing

Egy 8 kHz mintavételi frekvenciával dolgozó mintavételező rendszer bemenő és kimenő szűrője azonos:

$$H_I(f) = H_O(f) = \begin{cases} 1, & \text{ha } |f| \leq 3\text{ kHz} \\ 2.5 - |f/2|, & \text{ha } 3\text{ kHz} < |f| \leq 5\text{ kHz} \\ 0.01, & \text{ha } 5\text{ kHz} < |f| \leq 10\text{ kHz} \\ 0, & \text{egyébként} \end{cases}$$

A rendszer a 2 V amplitúdójú, 1 kHz frekvenciájú szinuszos jelösszetevőt - szivárgó komponensektől eltekintve - amplitúdóhelyesen viszi át. Határozzuk meg a szivárgó jelösszetevők amplitúdóját és frekvenciáját! Milyen jel keletkezik, ha a bemenő jel 2 V amplitúdójú, 4.5 kHz frekvenciájú szinuszos jel?

Megoldás:

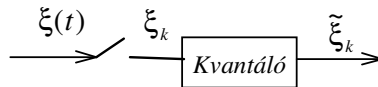
Az eredeti jel spektruma - még ha olyan elfajuló is, mint esetünkben - a mintavételi frekvencia harmonikusaival eltolódva hozza létre a mintasorozat spektrumát. Így a mintasorozat spektruma az $\pm n \cdot 8 \pm 1\text{ kHz}$ frekvenciájú harmonikus összetevőket tartalmazza. E komponensek amplitúdója a kimenő szűrő előtti ponton azonos. A szűrő megritkítja ezeket a jelösszetevőket: eliminálja mindazokat, amelyeknek a frekvenciája 10 kHz -nél nagyobb (illetve -10 kHz -nél kisebb). Így csak a $\pm 1\text{ kHz}$ és a $\pm 8 \pm 1\text{ kHz}$ frekvenciájú harmonikus összetevők maradnak meg, az előbbiek csillapítatlanul, az utóbbiak $H(\pm 8 \pm 1)$ értékének megfelelően csillapítva. Végül is tehát elmondható, hogy a visszaállított jelben 1 kHz , 7 kHz és 9 kHz frekvenciájú szinuszos összetevők lesznek jelen, ha az első amplitúdója 2 V , akkor a többieké 0.02 V .

A második kérdés megválaszolásához figyelembe kell vennünk, hogy ezt a jelet (e jel mindkét harmonikus komponensét) a bemenő szűrő csillapítja (egy 0.25 értékű faktorral). A mintasorozat spektruma most $\pm n \cdot 8 \pm 4.5\text{ kHz}$ frekvenciájú összetevőket tartalmaz, amelyek közül a kimenő szűrő véges csillapítással csak a $\pm 4.5\text{ kHz}$ és a $\pm(8-4.5) = \pm 3.5\text{ kHz}$ frekvenciájúakat engedi át. Az előbbi összetevőkön az erősítés 0.25 , az utóbbiakon 0.75 -szeres. Tehát az eredő jelnek két szinuszos

komponense lesz, a 3.5 kHz frekvenciájú alias jel amplitúdója 0.375 V, míg a valódi bemeneti jelnek megfelelő összetevő amplitúdója csak 0.125 V.

2.4. Példa: mintavételezés és kvantálás

Egy zérus várható értékű, stacionárius és ergodikus jelet mintavételezünk, majd a mintákat a (-1V,1V) intervallumban egyenletes lépésközzel kvantáljuk egy 4 bites kvantálóval.



A folyamat spektrális sűrűségfüggvénye a következő

$$s_{\xi}(f) = \begin{cases} 10^{-5} [V^2/Hz], & \text{ha } |f| \leq 10kHz \\ 0, & \text{egyébként} \end{cases}$$

- Mekkora a korreláció az egymás után kettővel következő minták között, ha a mintavételi idő $T = 10^{-5}$ sec?
- Hogyan kell megválasztani a mintavételi időt, hogy esetleg független mintákat kapjunk?
- Milyen ismeretekre volna még szüksége ahhoz, hogy megítélhesse: milyen valószínűséggel lesz a kvantáló szolgáltatása kódszó éppen 1101 értékű?

Megoldás:

a) A korrelációs fv. a spektrális sűrűségfüggvényből

$$R_{\xi}(\tau) = \int_{-\infty}^{\infty} s_{\xi}(f) e^{j2\pi f\tau} df = 2s_0 B \frac{\sin(2\pi B\tau)}{2\pi B\tau}$$

Ha $T=10^{-5}$ sec, és kettővel egymás utáni minták korrelációját kell számolni, akkor

$$E(\xi(t)\xi(t+2T)) = R_{\xi}(2T) = 0.2 \frac{\sin(0.4\pi)}{0.4\pi}$$

b) A minták függetlenségének szükséges kelléke, hogy korrelálatlanok legyenek, azaz $R_{\xi}(kT_0) = 0 \Rightarrow \sin(2\pi BkT_0) = 0 \Rightarrow T_0 = 50 \cdot 10^{-6} s$

c) Ismernünk kellene a kódolási szabályt - az intervallumok és a kódszavak összerendelését -, és a minták (egydimenziós) valószínűségeloszlását

2.5. Példa: kvantálási zaj, jel-zaj viszony

Becsüljük meg a ξ jel teljesítményét és a minták ábrázolásához felhasznált kódszavak méretét, ha tudjuk, hogy a mintánként kvantálásból származó kvantálási zaj teljesítménye kb. N_0 , a jel-zaj viszony pedig kb. ρ !

Megoldás:

A jel és a zaj teljesítményének hányadosa:

$$\frac{M(\xi^2)}{N_0} \cong \rho \approx 2^{2n},$$

tehát körülbelül $n = \frac{1}{2} \cdot \log_2 \rho$ méretű kódszavakat használnak (plusz 1..4 bitet, a nagyobb csúcstényező miatt). A jel teljesítménye $\rho \cdot N_0$ nagyságrendjébe esik. Ennél nagyobb teljesítményű jel túlvezérlést okoz, kisebb teljesítményű esetén pedig a kvantálási zaj teljesítménye változatlan lévén, a jel-zaj viszony romlik.

2.6. Példa: rekurzív kódoló egy lépéses predikcióval

Legyen a ξ jel egyenközű mintáinak sorozata olyan, hogy

$$\xi_k = a \cdot \xi_{k-1} + \varepsilon_k, \quad (2.1)$$

ahol $0 < a < 1$, ε pedig független, zérus várható értékű, σ^2 teljesítményű minták sorozata. Becsüljük meg a kvantálási zaj teljesítményét, ha mintánként kódolunk, és a megkívánt jel-zaj viszony ρ ! Hány bites mintákat kell használnunk? Mit változtat a helyzeten, ha egy olyan rekurzív kódolót használhatunk, amelyben a predikciót a dekódolt jelből a

$$\hat{\xi}_k = a \cdot \tilde{\xi}_{k-1},$$

a dekódolt mintát pedig a

$$\tilde{\xi}_k = \hat{\xi}_k + Q(\xi_k - \hat{\xi}_k)$$

szabállyal képezzük ($\hat{\xi}_k$ a predikált, $\tilde{\xi}_k$ a dekódolt mintát, Q az egyeneses kvantálást jelenti).

Megoldás:

(2.1) két tagja független (hiszen első tagját az ε sorozat korábbi tagjai határozzák meg), ezért

$$M(\xi_k^2) = a^2 \cdot M(\xi_{k-1}^2) + M(\varepsilon_k^2).$$

ξ stacionárius, teljesítménye pozíciófüggetlen, tehát:

$$M(\xi_k^2) = \frac{\sigma^2}{1 - a^2}.$$

Ha a kvantálót ehhez a jelhez illesztettük, és ρ jel-zaj viszonyt produkál, akkor a kvantálási zaj teljesítménye

$$N_0 = \frac{1}{\rho} \cdot \frac{\sigma^2}{1 - a^2}.$$

A mintákat leíró kódszavak hosszát a jel-zaj viszony szabja meg:

$$n \cong \frac{1}{2} \cdot \log_2 \rho.$$

Nézzük, mi a helyzet a rekurzív kódolásnál (rajzoljuk fel a blokksémát is, bejelölve a pontos bemeneti, a rekonstruált kimeneti, és a kimenetből predikált mintákat)! A rekonstruált jelminták hibája:

$$\xi_k - \tilde{\xi}_k = \xi_k - \hat{\xi}_k - Q(\xi_k - \hat{\xi}_k) = \xi_k - a \cdot \tilde{\xi}_{k-1} - Q(\xi_k - a \cdot \tilde{\xi}_{k-1}).$$

A felismerés: ez a hiba ugyanaz (éppen az), mint $\xi_k - a \cdot \tilde{\xi}_{k-1}$ kvantálási hibája! Ha e hibajel teljesítménye N_0 (a dekódolt minták jel-zaj viszonya ρ , ugyanaz, mint előbb), akkor viszont nincs szükség n bites kvantálásra, elég a p bites:

$$2^{2p} \cong \frac{M((\xi_k - \hat{\xi}_k)^2)}{N_0} = \frac{M((\xi_k - a \cdot \tilde{\xi}_{k-1})^2)}{N_0}.$$

A predikció (a predikált érték) hibája legalább ϵ_k (s ha a dekódolt minta nem túl rossz, akkor a predikált érték hibája döntő mértékben ez), ezért:

$$2^{2p} \cong \frac{\rho^2}{N_0} \Rightarrow p = \frac{1}{2} \log_2(\rho(1-a^2)).$$

Kis átalakítással:

$$p = \frac{1}{2} \log_2 \rho + \frac{1}{2} \log_2(1-a^2) = n + \frac{1}{2} \log_2(1-a^2).$$

Az összeg második tagja mindig negatív, ugyanazt a jelminőséget tehát a mintánkénti kódolásnál ("PCM kódolásnál") kevesebb bittel is biztosítani lehet.

2.7. Példa: kódolás a Karhunen-Loève transzformációval

Az előző példában vizsgált jel alapvető sajátossága, hogy szomszédos mintái nagyon hasonlóak. A KL transzformációban járatlan "józan agrárszakembert" ez motiválhatja arra, hogy a kódolandó mintasorozatot mintapárok sorozatára bontsa, majd az egyes mintapárok összegét, illetve különbségét kvantálja, alkalmasint azonos lépcsőméretű kvantálóval. Tehát a kvantált minták:

$$\begin{aligned} \eta_+ &= Q(\xi_1 + \xi_2) \\ \eta_- &= Q(\xi_1 - \xi_2) \end{aligned}$$

amelyekből dekódolt jelminták a

$$\begin{aligned} \tilde{\xi}_1 &= \frac{1}{2}(\eta_+ + \eta_-) \\ \tilde{\xi}_2 &= \frac{1}{2}(\eta_+ - \eta_-) \end{aligned}$$

transzformációval képezhető. A dekódolt jelminták hibájának szórásnégyzete akkor lesz N_0 , ha kvantálási hibák teljesítménye (η -k hibájának teljesítménye) $2N_0$ (Ehhez szükséges, hogy a "közös módusú" és a "differenciálmódusú" mintán elkövetett kvantálási hiba elég független legyen). Ebből viszont megítélhető, milyen méretű kódszavak kellenek az η -k (tipikus értékeinek) ábrázolásához:

$$n_+ \cong \frac{1}{2} \log_2 \left(\frac{\sigma^2 + (1+a)^2 \frac{\sigma^2}{1-a^2}}{2N_0} \right) = \frac{1}{2} \log_2 \left(\frac{\sigma^2}{N_0(1-a^2)} (1+a) \right) = \frac{1}{2} \log_2 \square + \frac{1}{2} \log_2(1+a)$$

másrészt:

$$n_- \cong \frac{1}{2} \log_2 \left(\frac{\sigma^2 + (1-a)^2 \frac{\sigma^2}{1-a^2}}{2N_0} \right) = \frac{1}{2} \log_2 \left(\frac{\sigma^2}{N_0(1-a^2)} (1-a) \right) = \frac{1}{2} \log_2 \square + \frac{1}{2} \log_2(1-a)$$

A mintapár leírására szolgáló bitek száma tehát:

$$n_+ + n_- \cong \log_2 \rho + \frac{1}{2} \log_2(1+a) + \frac{1}{2} \log_2(1-a) = \log_2 \rho + \frac{1}{2} \log_2(1-a^2),$$

az előző eredményekkel összevethető, egyetlen mintára vonatkoztatott érték:

$$n_{KL} = \frac{1}{2} (n_+ + n_-) \cong \frac{1}{2} \log_2 \rho + \frac{1}{4} \log_2(1-a^2).$$

Egy ravasz kérdés: miért rosszabb ez, mint a rekurzív kódolásnál?

Előzmények: fontos feltételezés, hogy a kvadratúramoduláció a D1 téma során terítékre került.

Jelek felbontása kvadratúrakomponensekre:

Legyen az $\eta_t, t \in (-\infty, \infty)$ jel sávkorlátozott az (F_1, F_2) sávra, ahol ráadásul kössük ki, hogy $F_2 < 2F_1$ (azaz a jel relatív sáv szélessége legyen kicsi)! Demoduláljuk ezt a jelet egy $F \in [F_1, F_2]$ vivőfrekvenciájú, $F_2 - F_1$ határfrekvenciájú szűrővel működő kvadratúrademodulátorral! Ezt a két demodulált jelet nevezzük az $\eta_t, t \in (-\infty, \infty)$ jel (F vivőhöz tartozó) kvadratúrakomponensének, legyenek ezek rendre η^c és η^s !

A kvadratúrakomponensek néhány tulajdonsága:

1. ezek konstrukciójukból fakadóan lassan változó jelek;
2. a kvadratúrakomponensek kvadratúramodulációval **előállítják** a jelet:
$$\eta_t^c \cos(2\pi Ft) - \eta_t^s \sin(2\pi Ft) = \eta_t, t \in (-\infty, \infty),$$
3. ha η stacionárius folyamat, akkor kvadratúrakomponensei is stacionáriusak;
4. korrelálatlanok, azaz

$$M(\eta_t^c \eta_t^s) = 0, t \in (-\infty, \infty),$$

5. és a spektrális sűrűségfüggvényük azonos, közelebbről:

$$s_c(f) = s_s(f) = s_\eta(f + F) + s_\eta(f - F), \text{ ha } |f| \leq F_2 - F_1,$$

és itt $s_\eta(\cdot)$ a jel spektrális sűrűségfüggvényét jelöli.

A 2. állítást tkp. elegendő η egyetlen szinuszos komponensére belátni. Az állítás az $F+f$ frekvenciájú (pl. koszinuszos) jelkomponensre igaz, hiszen ennek kvadratúrakomponensei $\cos(2\pi ft)$ és $\sin(2\pi ft)$, márpedig valóban igaz, hogy

$$\cos(2\pi ft) \cos(2\pi Ft) - \sin(2\pi ft) \sin(2\pi Ft) = \cos(2\pi(f + F)t).$$

Az 5. állítás valójában semmivel sem mond többet, mint amit a kétoldalsávú AM jel spektrumáról tanultunk a D1 téma keretei között, ugyanez áll a 3. állításra is. A 4. állítás nem triviális, nem is bizonyítjuk, annyit elmondhatunk, hogy azért igaz, mert egy stacionárius folyamatban a szinuszos és a koszinuszos összetevők minden frekvencián egyensúlyban kell legyenek (különben lesznek kitüntetett időpontok). A koszinuszok és a szinuszosok szorzatának viszont zérus az átlaga.

Megjegyzések:

1. A kvadratúrafelbontás tétele sokkal általánosabb keretek között is igaz. A tett megszorításokra azért van szükség, mert így a "bizonyítások" igen egyszerűek, s a kvadratúrakomponensek is "szemléletesek". További vizsgálataink hatókörét e megszorítások ugyanakkor érdemben nem befolyásolják.

2. Különösképpen érdekes helyzet, ha a kvadratúrafelbontás vivője a felbontandó jel sávjának szélére (vagy kicsit azon is túl) esik, azaz, ha

$$F \leq F_1, \text{ vagy } F \geq F_2.$$

Ekkor a kvadratúrakomponensek minden szinuszos összetevője azonos nagyságú, fázisaik viszont éppen $\pi/2$ -vel térnek el egymástól, másként szólva: ők egymás Hilbert transzformáltjai. Ez különösen SSB jelek leírásánál érdekes

Jelek modulációs felbontása:

Közismert, hogy azonos frekvenciájú szinuszos és koszinuszos jelek összege változatlan frekvenciájú szinuszos jel, megváltozott amplitúdóval és fázissal. Az összeg képzésére memorizálására több szemléletes módszert is tanultunk különféle tárgyokban, pl. komplex számos, vagy forgó vektoros ábrázolások.

Ezek az összegzési szabályok szikrát sem változnak, ha a szereplő szinuszos - koszinuszos tagok amplitúdói függvényei az időnek. Így tehát a kvadratúrafelbontás képletét átírva:

$$\eta_t = \sqrt{(\eta_t^c)^2 + (\eta_t^s)^2} \cos\left(2\pi Ft + \arctan \frac{\eta_t^s}{\eta_t^c}\right).$$

A formálisan nyilvánvalóan helyes kifejezés mondanivalója, hogy minden (legalább minden, viszonylag keskenysávú) jel felfogható amplitúdóban és fázisban egyidejűleg (és nem is független tartalommal) modulált jelnek. Az átalakítás jelentősége, hogy a jel ezen, modulációs alakjából gyakran közvetlenül leolvasható amúgy bonyolult rendszertechnikai beavatkozások hatása. Pl. azonnal felírható, mi lesz egy ideális burkoló-leválasztás eredménye, ha a blokk bemenetére η kerül.

Valós értékű jel komplex alakban

Korábban is láttunk rá példát, hogy fogalmi egyszerűsítést eredményezhet, ha egy valós értékű függvényt egy komplex értékű függvény valós (vagy éppen imaginárius) részeként kezeljük. (szinuszos és koszinuszos helyett szívesen használunk komplex exponenciális függvényt). A valós jel modulációs előállítását szinte kiabál egy effajta felírásért. Nyilvánvalóan igaz, hogy

$$\eta_t = \operatorname{Re} \left[\sqrt{(\eta_t^c)^2 + (\eta_t^s)^2} \exp\left(j2\pi Ft + j \arctan \frac{\eta_t^s}{\eta_t^c}\right) \right].$$

A szögletes zárójelen belüli komplex kifejezést nevezzük a jel komplex alakjának, s ez felismerhetően

$$\tilde{\eta}_t = (\eta_t^c + j\eta_t^s) \cdot e^{j2\pi Ft}.$$

A jel komplex alakja egy sajátos, komplex vivőjű, komplex modulációs tartalommal felszerelt AM jel. Mivel a modulációs tartalom a harmonikus vivőnek kizárólag az "amplitúdóját" befolyásolja, mi sem természetesebb, mint az, hogy a modulációs tartalmat a jel komplex burkolójának nevezzük, ez tehát

$$b_\eta(t) = \eta_t^c + j\eta_t^s.$$

A komplex burkoló függ attól, milyen F választással éltünk (modulált jelek esetén persze elég természetes F -nek a modulált jel vivőfrekvenciáját választani), így talán pontosabb volna a jelnek az F frekvenciára vonatkoztatott komplex burkolójáról beszélni. Érdekes viszont, hogy a komplex burkoló abszolút értéke független F -től.

Belátható, hogy a komplex jel csak pozitív frekvenciájú harmonikus komponensekkel rendelkezik (de a bizonyításról most eltekintünk), e komponensek viszont éppen kétakkorák, mint a valós jel összetevői. Elmondhatjuk, e komplex reprezentáció különösképpen modulált jeleket érő lineáris torzítások hatásának vizsgálatánál előnyös.

A zaj hatása a demodulált jelre, különféle modulációs rendszerekben

Jelölések:

S : a modulált jel teljesítménye

B : a modulált jel sáv szélessége

s_0 : a szélessávú additív vonali zaj spektrális sűrűsége

v, v^c, v^s : a vevőszűrőn átjutott zaj, ill. az ő kvadratúrakomponensei

ξ : a moduláló jel

1. AM-DSB, szorzó demodulátorral

A modulált jel: $\xi_i \cos(2\pi Ft)$, teljesítménye: $S = M(\xi_i^2)/2$

A vevőszűrőn átjutott jel és zaj együtt:

$$\xi_i \cos(2\pi Ft) + v_i^c \cos(2\pi Ft) + v_i^s \sin(2\pi Ft)$$

(a zaj kvadratúrafelbontásának vonatkoztatási frekvenciáját azonosnak választjuk a modulált jel vivőfrekvenciájával)

A demodulátor a koszinuszos vivő együtthatóját hiszi modulációs tartalomnak, a demodulált zajos jel tehát $\xi_i + v_i^c$.

A zaj kvadratúrakomponensének spektrális sűrűsége most $2s_0$, ezért a zaj teljesítménye: $2s_0 \cdot 2B$.

A demodulált jelben mutatkozó jel-zaj viszony így: $S/(2s_0B)$

2. AM-SSB, szorzó demodulátorral

A modulált jel kvadratúramodulált alakban: $\xi_i \cos(2\pi Ft) - \hat{\xi}_i \sin(2\pi Ft)$,

A kvadratúrakomponensek korrelálatlanok, teljesítményük azonos, így a jel teljesítménye: $S = 2 \cdot M(\xi_i^2)/2 = M(\xi_i^2)$

A vevőszűrőn átjutott jel és zaj együtt:

$$\xi_i \cos(2\pi Ft) + \hat{\xi}_i \sin(2\pi Ft) + v_i^c \cos(2\pi Ft) + v_i^s \sin(2\pi Ft)$$

(a zaj kvadratúrafelbontásának vonatkoztatási frekvenciáját azonosnak választjuk a modulált jel vivőfrekvenciájával)

A demodulátor a koszinuszos vivő együtthatóját hiszi modulációs tartalomnak, a demodulált zajos jel tehát $\xi_t + v_t^c$.

A zaj kvadratúrakomponensének spektrális sűrűsége most csak s_0 , ezért a zaj teljesítménye: $s_0 \cdot 2B$.

A demodulált jelben mutatkozó jel-zaj viszony így: $S/(2s_0B)$, ugyanaz, mint előbb.

3. AM-DSB, burkoló demodulátorral

A modulált jel: $(V + \xi_t) \cdot \cos(2\pi Ft)$, teljesítménye: $S = V^2/2 + M(\xi_t^2)/2$

A vevőszűrőn átjutott jel és zaj együtt:

$$(V + \xi_t) \cos(2\pi Ft) + v_t^c \cos(2\pi Ft) + v_t^s \sin(2\pi Ft)$$

(a zaj kvadratúrafelbontásának vonatkoztatási frekvenciáját azonosnak választjuk a modulált jel vivőfrekvenciájával). E jel modulációs felbontása:

$$\sqrt{(V + \xi_t + v_t^c)^2 + (v_t^s)^2} \cos\left(2\pi Ft + \arctan \frac{v_t^s}{V + \xi_t + v_t^c}\right).$$

A demodulátor (feltéve persze, hogy valóban képes rá) az ingadozó fázisú vivő burkolóját választja le. Ha a zaj gyenge, akkor

$$\sqrt{(V + \xi_t + v_t^c)^2 + (v_t^s)^2} \cong |V + \xi_t + v_t^c|.$$

Mindaddig, amíg a moduláló jel nem túl nagy, a jobb oldalon az abszolút érték függvény argumentuma előjeltartó, s így lényegében a demodulált jel: $V + \xi_t + v_t^c$.

A zaj kvadratúrakomponensének spektrális sűrűsége most $2s_0$, ezért a zaj teljesítménye: $2s_0 \cdot 2B$.

Tételezzük fel továbbá, hogy a moduláló jel csúcsstényezője c , s azt is, hogy V éppen a jel csúcsértékével egyenlő! Ezért:

$$V^2/2 = c^2 \cdot M(\xi_t^2)/2.$$

A demodulált jelben mutatkozó jel-zaj viszony így: $S/(2s_0B)$

$$\frac{2S}{(1+c^2) \cdot 4s_0B} = \frac{1}{1+c^2} \cdot \frac{S}{2s_0B}.$$

4. Fázismoduláció

A modulált jel most: $V \cos(2\pi Ft + \mu_t)$, teljesítménye: $S = V^2/2$

A demodulátor kiszippantja a koszinusz fázisát, az maga a demodulált jel, teljesítménye: $M(\mu_t^2)$, ami éppen az effektív fázislököt négyzete, azaz Φ_{eff}^2 .

A demodulált zajt a moduláló jel szüneteiben vizsgáljuk, ekkor vevőszűrőn átjutott jel és zaj együtt:

$$V \cos(2\pi Ft) + v_t^c \cos(2\pi Ft) + v_t^s \sin(2\pi Ft)$$

(a zaj kvadratúrafelbontásának vonatkoztatási frekvenciáját azonosnak választjuk a modulált jel vivőfrekvenciájával)

Az összegjel modulációs felbontása:

$$\sqrt{(V + v_t^c)^2 + (v_t^s)^2} \cos\left(2\pi Ft + \arctan \frac{v_t^s}{V + v_t^c}\right)$$

A demodulátor számára ebből a koszinusz fázisa fontos, s ha a zaj gyenge, akkor ez

$$\arctan \frac{v_t^s}{V + v_t^c} \cong \frac{v_t^s}{V}.$$

Ez a fáziszaj, amely $2s_0/V^2$ spektrális sűrűségű, a teljesítménye:

$$N = 2 \cdot \frac{2s_0}{V^2} \cdot B.$$

A demodulált jelben mutatkozó jel-zaj viszony így:

$$\Phi_{eff}^2 / \left(2s_0 B \frac{2}{V^2}\right) = \frac{S}{2s_0 B} \cdot \Phi^2$$

A jel-zaj viszony lényegesen jobb lehet, mint a különféle AM esetében. Az igaz, hogy a nagyobb fázislököt nagyobb sáv szélességű jellel jár. Lényeges lehet, hogy a jel sáv szélessége viszont nem növelhető korlátlanul, túl nagy sáv szélességnél a demodulátorba jutó zaj túlnövi a gyenge zajú közelítés korlátait.

Maradandó tanulság, hogy jobban járunk (jel-zaj viszony tekintetében), ha rendelkezésre álló jel teljesítményt széles sávban szétkenjük

5. Frekvenciamoduláció

A modulált jel most: $V \cos(2\pi Ft + \mu_t)$, teljesítménye: $S = V^2/2$

A demodulátor kiszippantja a koszinusz fázisát, és deriválja azt, tehát a demodulált jel teljesítménye: $M(\mu_t^2)$, ami éppen az effektív (kör)frekvencialököt négyzete, azaz $(2\pi)^2 \cdot F_{eff}^2$.

A demodulált zajt a moduláló jel szüneteiben vizsgáljuk, ekkor vevőszűrőn átjutott jel és zaj együtt:

$$V \cos(2\pi Ft) + v_t^c \cos(2\pi Ft) + v_t^s \sin(2\pi Ft)$$

(a zaj kvadratúrafelbontásának vonatkoztatási frekvenciáját azonosnak választjuk a modulált jel vivőfrekvenciájával)

Az összegjel modulációs felbontása:

$$\sqrt{(V + v_t^c)^2 + (v_t^s)^2} \cos\left(2\pi Ft + \arctan \frac{v_t^s}{V + v_t^c}\right)$$

A demodulátor számára ebből a koszinusz fázisa fontos, s ha a zaj gyenge, akkor ez

$$\arctan \frac{v_t^s}{V + v_t^c} \cong \frac{v_t^s}{V}.$$

Ez a fáziszaj, amely így $2s_0/V^2$ spektrális sűrűségű, még nem a végeredmény, hiszen a frekvenciademodulátor kiszippant és derivál. A deriválás átviteli függvénye még befolyásolja a demodulált zajt, amelynek a teljesítménye így:

$$N = \int_{-B}^B \frac{2s_0}{V^2} \cdot |j2\pi f|^2 df = 2 \cdot \frac{2s_0}{V^2} \cdot (2\pi)^2 \cdot \frac{B^3}{3}.$$

A demodulált jelben mutatkozó jel-zaj viszony így:

$$F_{eff}^2 / \left(2s_0 B \frac{2}{V^2} \frac{B^2}{3} \right) = \frac{S}{2s_0 B} \cdot \frac{1}{3} \cdot \left(\frac{F_{eff}}{B} \right)^2$$

Ez az eredmény is azt mutatja, hogy korlátos adóteljesítmény és adott (csatorna)zaj mellett van lehetőség a jel-zaj viszony javítására, ez azonban áldozattal jár, a csatornát szélesebb sávban kell használni.

4. Forráskódolás és hibavédő kódolás

4.1. Forráskódolás

Alapfeladatunk szimbólumsorozatok megjelenítése bináris elemek sorozatának segítségével. Az egyértelmű megfejthetőségen túl követelmény az ábrázolás tömörsége. A kézenfekvő, minden szimbólumot ugyanolyan (L) hosszú bitsorozattal leíró kódolás lehet ugyan egyértelműen megfejthető (ha a szimbólumok N száma nem nagyobb, mint az L elemű bináris sorozatok száma, 2^L), de általában nem túl tömör, hiszen nem számol azzal a körülménnyel, hogy egyes szimbólumok ritkák, mások viszont gyakoriak. Ésszerű lehet viszont az olyan kódolás, amely a gyakori szimbólumokat kevés elemmel jeleníti meg. A különböző hosszúságú kódszavak valamennyien nem lehetnek igen rövidek, sőt, túl sok rövid kódszó sem lehet (pl. nem lehet három egybites kódszó, ez ellentmondana az egyértelmű megfejthetőség követelményének). Az egyértelmű megfejthetőségnél szorosabb követelmény lehet a folyamatos olvashatóság (a megfejtéshez ne kelljen megvárni a teljes kódolt sorozat végét). Bizonyítható, hogy a legtömörebb kódolás L átlagos kódszóhosszára igaz a

$$H \leq L < 1 + H$$

egyenlőtlenség, ahol

$$H = \sum_{i=1}^N p_i \cdot \log_2 \left(\frac{1}{p_i} \right)$$

a forrás entrópiája (p_i az egyes forrásszimbólumok valószínűségét jelenti).

Ennél több is igaz: ha nem szimbólumokat, hanem szimbólumcsoportokat kódolunk, akkor az egyetlen szimbólum megjelenítésére fordított bitek várható száma tetszőlegesen megközelítheti az entrópia értékét. K méretű szimbólumcsoportokat kódolva:

$$H \leq \frac{1}{K} L^{(K)} < \frac{1}{K} + H.$$

4.2. Shannon kód szerkesztése

A forrásszimbólumokat valószínűségeik szerint csökkenő sorba rendezzük, azaz $p_1 \geq p_2 \geq \dots \geq p_n$. Shannon szerint mármint az i -edik szimbólumot megjelenítő kódszó az $F_i \in [0, 1]$ szám $l_i = \lceil \log_2 1/p_i \rceil$ bite kerekített értéke, ahol

$$F_i = \sum_{k=1}^{i-1} p_k.$$

Az alábbi táblázat egy 6 szimbólumos forrás kódjának elkészítését mutatja.

szimb./ i	x1/1	x2/2	x3/3	x4/4	x5/5	x6/6
p_i	0.4	0.25	0.12	0.1	0.08	0.05
l_i	2	2	4	4	4	5
F_i	0.0	0.4	0.65	0.77	0.87	0.95
bináris F_i	0.00000	0.0110	0.10100	0.11000	0.11011	0.11100
kódszó	00	01	1010	1100	1101	11100

A Shannon kód gyakran rövidíthető. Példánkban látható, hogy az utolsó szimbólumhoz tartozó kódszó utolsó eleme elhagyható, a kód ettől még prefix (mentes) marad. Ha sort tudunk rá keríteni - elég időnk van - célszerű kiszámítani a kód átlagos szóhosszát, s összevetni az entrópiával.

4.3. Huffman kód

Huffman Coding

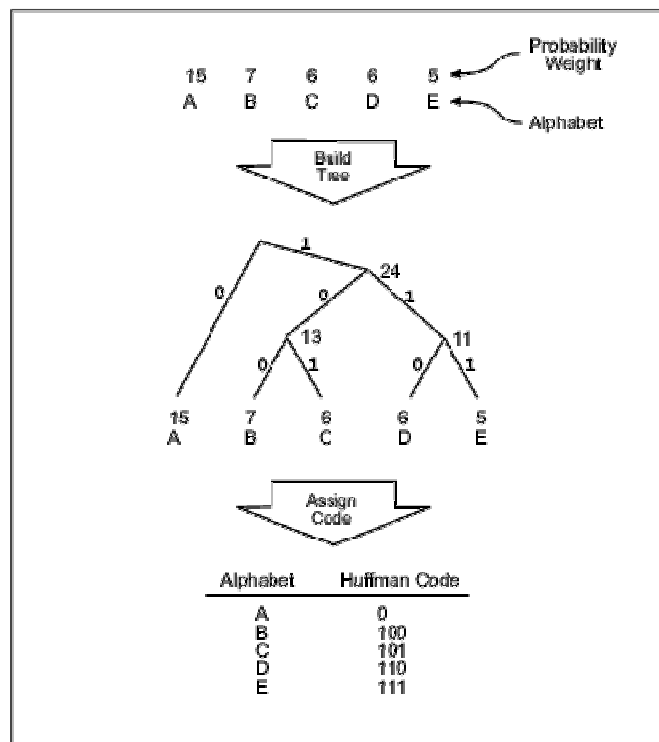


Figure 2.4: The Huffman coding algorithm example. This example has 5 source alphabets with their corresponding frequencies above the symbols.

Huffman coding [#!lossless01!#] is a classical data compression technique. It has been used in various compression applications, including image compression. It is a simple but yet elegant compression technique that can supplement other compression algorithms. It utilizes the statistical property of alphabets in source stream, and then produces respective codes for these alphabets. These codes are of variable code length using integral number of bits. The codes for alphabets having higher probability of occurrence are shorter than those codes for alphabets having lower probability. This simple idea causes a reduction in the average code length, and thus the overall size of compressed data is smaller than the original. For the details of the compression process, it is based on building a binary tree that holds all alphabets in the source at its leaf nodes, and with their corresponding alphabets' probabilities at the side (Fig 2.4). The tree is built by going through the following steps: -

1. Each of the alphabets is laid out as leaf node which is going to be connected. The alphabets are ranked according to their weights, which represents the probabilities of their occurrences in the source.
2. Two nodes with the lowest weights are combined to form a new node, which is a parent node of these two nodes. This parent node is then considered as a representative of the two nodes with a weight equal to the sum of the weights of two nodes. Moreover, one of the child is assigned a "0" and another is assigned a "1".
3. Nodes are then successively combined as above until a binary tree containing all of nodes is created.
4. The code representing a given alphabet can be determined by going from the root of the tree to the leaf node representing the alphabet. The accumulation of symbol "0" and "1" is the code of that alphabet.

By using this procedure, the alphabets are naturally assigned codes that reflect probability distribution. Highly probable alphabets will be given short codes, and improbable alphabets will have long codes. Therefore, the average code length will be reduced. If the statistic of alphabets is very biased to some particular alphabets, the reduction will be very significant. A compression technique related to Huffman coding is the Shannon-Fano coding, which works basically as Huffman coding. The major different is that Shannon-Fano coding is using a top-down design of tree instead of the bottom-up approach in Huffman coding, but they show similar performance to each other.

4.3. Tömörítés a gyakorlatban

Ha volna rá idő, érdemes lenne ismertetni a Lempel-Ziv kódolás alap gondolatát. Segítségképpen szolgáljon Marosi Gyuszi gyűjtése, íme:

LZW and GIF explained----Steve Blackstock

I hope this little document will help enlighten those of you out there who want to know more about the Lempel-Ziv Welch compression algorithm, and, specifically, the implementation that GIF uses.

Before we start, here's a little terminology, for the purposes of this document:

"character": a fundamental data element. In normal text files, this is just a single byte. In raster images, which is what we're interested in, it's an index that specifies the color of a given pixel. I'll refer to an arbitrary character as "K".

"charstream": a stream of characters, as in a data file.

"string": a number of continuous characters, anywhere from one to very many characters in length. I can specify an arbitrary string as "[...]K".

"prefix": almost the same as a string, but with the implication that a prefix immediately precedes a character, and a prefix can have a length of zero. So, a prefix and a character make up a string. I will refer to an arbitrary prefix as "[...]".

"root": a single-character string. For most purposes, this is a character, but we may occasionally make a distinction. It is [...]K, where [...] is empty.

"code": a number, specified by a known number of bits, which maps to a string.

"codestream": the output stream of codes, as in the "raster data"

"entry": a code and its string.

"string table": a list of entries; usually, but not necessarily, unique.

That should be enough of that.

LZW is a way of compressing data that takes advantage of repetition of strings in the data. Since raster data usually contains a lot of this repetition, LZW is a good way of compressing and decompressing it.

For the moment, let's consider normal LZW encoding and decoding. GIF's variation on the concept is just an extension from there.

LZW manipulates three objects in both compression and decompression: the charstream, the codestream, and the string table. In compression, the charstream is the input and the codestream is the output. In decompression, the codestream is the

input and the charstream is the output. The string table is a product of both compression and decompression, but is never passed from one to the other.

The first thing we do in LZW compression is initialize our string table. To do this, we need to choose a code size (how many bits) and know how many values our characters can possibly take. Let's say our code size is 12 bits, meaning we can store 0->FFF, or 4096 entries in our string table. Let's also say that we have 32 possible different characters. (This corresponds to, say, a picture in which there are 32 different colors possible for each pixel.) To initialize the table, we set code#0 to character#0, code #1 to character#1, and so on, until code#31 to character#31. Actually, we are specifying that each code from 0 to 31 maps to a root. There will be no more entries in the table that have this property.

Now we start compressing data. Let's first define something called the "current prefix". It's just a prefix that we'll store things in and compare things to now and then. I will refer to it as "[.c.]". Initially, the current prefix has nothing in it. Let's also define a "current string", which will be the current prefix plus the next character in the charstream. I will refer to the current string as "[.c.]K", where K is some character. OK, look at the first character in the charstream. Call it P. Make [.c.]P the current string. (At this point, of course, it's just the root P.) Now search through the string table to see if [.c.]P appears in it. Of course, it does now, because our string table is initialized to have all roots. So we don't do anything. Now make [.c.]P the current prefix. Look at the next character in the charstream. Call it Q. Add it to the current prefix to form [.c.]Q, the current string. Now search through the string table to see if [.c.]Q appears in it. In this case, of course, it doesn't. Aha! Now we get to do something. Add [.c.]Q (which is PQ in this case) to the string table for code#32, and output the code for [.c.] to the codestream. Now start over again with the current prefix being just the root P. Keep adding characters to [.c.] to form [.c.]K, until you can't find [.c.]K in the string table. Then output the code for [.c.] and add [.c.]K to the string table. In pseudo-code, the algorithm goes something like this:

```
[1] Initialize string table;
[2] [.c.] <- empty;
[3] K <- next character in charstream;
[4] Is [.c.]K in string table?
(yes: [.c.] <- [.c.]K;
    go to [3];
)
(no: add [.c.]K to the string table;
    output the code for [.c.] to the codestream;
    [.c.] <- K;
```



```
    go to [3];  
  )
```

It's as simple as that! Of course, when you get to step [3] and there aren't any more characters left, you just output the code for [.c.] and throw the table away. You're done.

Wanna do an example? Let's pretend we have a four-character alphabet: A,B,C,D. The charstream looks like ABACABA. Let's compress it. First, we initialize our string table to: #0=A, #1=B, #2=C, #3=D. The first character is A, which is in the string table, so [.c.] becomes A. Next we get AB, which is not in the table, so we output code #0 (for [.c.]),

and add AB to the string table as code #4. [.c.] becomes B. Next we get [.c.]A = BA, which is not in the string table, so output code #1, and add BA to the string table as code #5. [.c.] becomes A. Next we get AC, which is not in the string table. Output code #0, and add AC to the string table as code #6. Now [.c.] becomes C. Next we get [.c.]A = CA, which is not in the table. Output #2 for C, and add CA to table as code#7. Now [.c.] becomes A. Next we get AB, which IS in the string table, so [.c.] gets AB, and we look at ABA, which is not in the string table, so output the code for AB, which is #4, and add ABA to the string table as code #8. [.c.] becomes A. We can't get any more characters, so we just output #0 for the code for A, and we're done. So, the codestream is #0#1#0#2#4#0.

A few words (four) should be said here about efficiency: use a hashing strategy. The search through the string table can be computationally intensive, and some hashing is well worth the effort. Also, note that "straight LZW" compression runs the risk of overflowing the string table - getting to a code which can't be represented in the number of bits you've set aside for codes. There are several ways of dealing with this problem, and GIF implements a very clever one, but we'll get to that.

An important thing to notice is that, at any point during the compression, if [...]K is in the string table, [...] is there also. This fact suggests an efficient method for storing strings in the table. Rather than store the entire string of K's in the table, realize that any string can be expressed as a prefix plus a character: [...]K. If we're about to store [...]K in the table, we know that [...] is already there, so we can just store the code for [...] plus the final character K.

Ok, that takes care of compression. Decompression is perhaps more difficult conceptually, but it is really easier to program.

Here's how it goes: We again have to start with an initialized string table. This table comes from what knowledge we have about the charstream that we will

eventually get, like what possible values the characters can take. In GIF files, this information is in the header as the number of possible pixel values. The beauty of LZW, though, is that this is all we need to know. We will build the rest of the string table as we decompress the codestream. The compression is done in such a way that we will never encounter a code in the codestream that we can't translate into a string.

We need to define something called a "current code", which I will refer to as "<code>", and an "old-code", which I will refer to as "<old>". To start things off, look at the first code. This is now <code>. This code will be in the initialized string table as the code for a root. Output the root to the charstream. Make this code the old-code <old>. *Now look at the next code, and make it <code>. It is possible that this code will not be in the string table, but let's assume for now that it is. Output the string corresponding to <code> to the codestream. Now find the first character in the string you just translated. Call this K. Add this to the prefix [...] generated by <old> to form a new string [...]K. Add this string [...]K to the string table, and set the old-code <old> to the current code <code>. Repeat from where I typed the asterisk, and you're all set. Read this paragraph again if you just skimmed it!!! Now let's consider the possibility that <code> is not in the string table. Think back to compression, and try to understand what happens when you have a string like P[...]P[...]PQ appear in the charstream. Suppose P[...] is already in the string table, but P[...]P is not. The compressor will parse out P[...], and find that P[...]P is not in the string table. It will output the code for P[...], and add P[...]P to the string table. Then it will get up to P[...]P for the next string, and find that P[...]P is in the table, as the code just added. So it will output the code for P[...]P if it finds that P[...]PQ is not in the table. The decompressor is always "one step behind" the compressor. When the decompressor sees the code for P[...]P, it will not have added that code to its string table yet because it needed the beginning character of P[...]P to add to the string for the last code, P[...], to form the code for P[...]P. However, when a decompressor finds a code that it doesn't know yet, it will always be the very next one to be added to the string table. So it can guess at what the string for the code should be, and, in fact, it will always be correct. If I am a decompressor, and I see code#124, and yet my string table has entries only up to code#123, I can figure out what code#124 must be, add it to my string table, and output the string. If code#123 generated the string, which I will refer to here as a prefix, [...], then code#124, in this special case, will be [...] plus the first character of [...]. So just add the first character of [...] to the end of itself. Not too bad. As an example (and a very common one) of this special case, let's assume we have a raster image in which the first three pixels have the same color value. That is, my charstream looks like: QQQ.... For the sake of argument, let's say we have 32 colors, and Q is the color#12. The compressor will

generate the code sequence 12,32,... (if you don't know why, take a minute to understand it.) Remember that #32 is not in the initial table, which goes from #0 to #31. The decompressor will see #12 and translate it just fine as color Q. Then it will see #32 and not yet know what that means. But if it thinks about it long enough, it can figure out that QQ should be entry#32 in the table and QQ should be the next string output. So the decompression pseudo-code goes something like:

```
[1] Initialize string table;
[2] get first code: <code>;
[3] output the string for <code> to the charstream;
[4] <old> = <code>;
[5] <code> <- next code in codestream;
[6] does <code> exist in the string table?
    (yes: output the string for <code> to the charstream;
        [...] <- translation for <old>;
        K <- first character of translation for <code>;
        add [...]K to the string table;    <old> <- <code>; )
    (no: [...] <- translation for <old>;
        K <- first character of [...];
        output [...]K to charstream and add it to string table;
        <old> <- <code>
    )
[7] go to [5];
```

Again, when you get to step [5] and there are no more codes, you're finished. Outputting of strings, and finding of initial characters in strings are efficiency problems all to themselves, but I'm not going to suggest ways to do them here. Half the fun of programming is figuring these things out!

Now for the GIF variations on the theme. In part of the header of a GIF file, there is a field, in the Raster Data stream, called "code size". This is a very misleading name for the field, but we have to live with it. What it is really is the "root size". The actual size, in bits, of the compression codes actually changes during compression/decompression, and I will refer to that size here as the "compression size". The initial table is just the codes for all the roots, as usual, but two special codes are added on top of those. Suppose you have a "code size", which is usually the number of bits per pixel in the image, of N. If the number of bits/pixel is one, then N must be 2: the roots take up slots #0 and #1 in the initial table, and the two special

codes will take up slots #4 and #5. In any other case, N is the number of bits per pixel, and the roots take up slots #0 through $\#(2^{**}N-1)$, and the special codes are $(2^{**}N)$ and $(2^{**}N + 1)$. The initial compression size will be $N+1$ bits per code. If you're encoding, you output the codes $(N+1)$ bits at a time to start with, and if you're decoding, you grab $(N+1)$ bits from the codestream at a time. As for the special codes: `<CC>` or the clear code, is $(2^{**}N)$, and `<EOI>`, or end-of-information, is $(2^{**}N + 1)$. `<CC>` tells the compressor to re-initialize the string table, and to reset the compression size to $(N+1)$. `<EOI>` means there's no more in the codestream. If you're encoding or decoding, you should start adding things to the string table at `<CC> + 2`. If you're encoding, you should output `<CC>` as the very first code, and then whenever after that you reach code #4095 (hex FFF), because GIF does not allow compression sizes to be greater than 12 bits. If you're decoding, you should reinitialize your string table when you observe `<CC>`. The variable compression sizes are really no big deal. If you're encoding, you start with a compression size of $(N+1)$ bits, and, whenever you output the code $(2^{**}(\text{compression size})-1)$, you bump the compression size up one bit. So the next code you output will be one bit longer. Remember that the largest compression size is 12 bits, corresponding to a code of 4095. If you get that far, you must output `<CC>` as the next code, and start over. If you're decoding, you must increase your compression size AS SOON AS YOU write entry $\#(2^{**}(\text{compression size}) - 1)$ to the string table. The next code you READ will be one bit longer. Don't make the mistake of waiting until you need to add the code $(2^{**}\text{compression size})$ to the table. You'll have already missed a bit from the last code. The packaging of codes into a bitsream for the raster data is also a potential stumbling block for the novice encoder or decoder. The lowest order bit in the code should coincide with the lowest available bit in the first available byte in the codestream. For example, if you're starting with 5-bit compression codes, and your first three codes are, say, `<abcde>`, `<fghij>`, `<klmno>`, where e, j, and o are bit#0, then your codestream will start off like:

```
byte#0: hijabcde  
byte#1: .klmnofg
```

So the differences between straight LZW and GIF LZW are: two additional special codes and variable compression sizes. If you understand LZW, and you understand those variations, you understand it all!

Just as sort of a P.S., you may have noticed that a compressor has a little bit of flexibility at compression time. I specified a "greedy" approach to the compression, grabbing as many characters as possible before outputting codes. This is, in fact, the

standard LZW way of doing things, and it will yield the best compression ratio. But there's no rule saying you can't stop anywhere along the line and just output the code for the current prefix, whether it's already in the table or not, and add that string plus the next character to the string table. There are various reasons for wanting to do this, especially if the strings get extremely long and make hashing difficult. If you need to, do it.

Hope this helps out.----steve blackstock

4.4. Digitális csatorna minősítése (jellemzése)

A digitális csatorna akkor "tökéletes", ha a kimeneti szimbólumokból teljes bizonyossággal lehet következtetni a bemenetiekre. Ha e következtetés olykor hibás, (de e hibák függetlenek, valószínűségük stabil, stb.) akkor zajos csatornáról beszélünk. A zajos csatorna nagyvonalú jellemzésére szolgál a hibavalószínűség.

Kicsit más természetű, a csatorna átviteli képességét alaposabban jellemző adat az ún. csatornakapacitás, amely a forrásüzenetek hosszának és a forrásüzeneteket hiánytalanul célba juttató csatornaszimbólumok számának hányadosaként értelmezhető. A csatornakapacitást általában bináris, maximális entrópiájú forrásra vonatkoztatjuk, ekkor értéke arról tájékoztat, hogy a csatornát a lehető legügyesebben használva, egyetlen működtetése hány bináris elem *tökéletes* átvitelét teszi lehetővé. Egyes használat alatt a hibásan átvitt elemek javításának - pótlásának - módszereit értjük.

4.5. Példa: az eltörlődéses csatorna kapacitása

Egyszerű, szemléletes példája a digitális csatornáknak az ún. eltörlődéses csatorna. Bemeneti és kimeneti szimbólumai azonosak, kivéve azt a kimeneti szimbólumot, amely az eltörlődött, olvashatatlan pozíció(ka)t jelzi. Az eltörlődéstől eltekintve ez a csatorna nem hibázik, azaz, ha egy pozíció olvasható, akkor pontosan tudjuk, mi is volt a hozzá tartozó bemeneti szimbólum. Legyen egy bináris eltörlődéses csatornán az eltörlődés valószínűsége q ! Határozzuk meg a csatorna kapacitását!

Megoldás:

Azt kell vizsgálnunk, n bites üzenet átvitele érdekében (esetleg idealizált körülmények között) hányszor kell működtetnünk a csatornát. Első próbálkozásképpen átvisszük az n hosszú üzenetet. A túloldal ennek jó részét hibátlanul megkapja, de várhatóan nq pozíció olvashatatlan lesz, tehát nq számú bitet

pótolni kell. Ha az adó tudja, üzenetének melyek voltak az eltörlődött elemei (erre szolgálhat e "gondolatkísérletben" egy ideális visszajelzőcsatorna), akkor a javítóüzenete kizárólag ezeket az elemeket tartalmazza, ez összesen nq bit. Persze a javítóüzenet egyes elemei is elmosódhatnak, az eltörlődött elemek száma várhatóan nq^2 . Ezeket is pótolni kell, újabb javítóüzenetek segítségével. Összesen tehát

$$m = n + nq + nq^2 + \dots = \frac{n}{1-q}$$

számú csatornaműködtetésre került sor. A csatorna egyetlen működtetése átlagban

$$C = \frac{n}{m} = 1 - q$$

bit átvitelét valósítja meg. Azt mondjuk, az eltörlődéses csatorna kapacitása $1-q$.

4.6. Hibajavítás lineáris kóddal

Vizsgáljuk azt a bináris (8,4) kódot, amelynek generátormátrixa

$$\mathbf{G} = \begin{bmatrix} 1 & 0 & 0 & 00 & 0 & 1 & 1 \\ 0 & 1 & 0 & 00 & 1 & 1 & 0 \\ 0 & 0 & 1 & 10 & 1 & 0 & 1 \\ 0 & 0 & 0 & 11 & 0 & 0 & 1 \end{bmatrix}.$$

Állapítsuk meg, szisztematikus-e ez a kód, és állítsuk elő a paritásellenőrző mátrixát! Próbáljuk meg kitalálni, hány hiba jelzésére/javítására lehet alkalmas! Képezzük azt a generátormátrixot, amely szisztematikusán állítja elő ugyanezt a kódot! Képezzük az $u = [0 \ 1 \ 1 \ 0]$ üzenethez tartozó kódszót, s tételizzük fel, hogy e kódszó első bitje meghibásodik! Demonstráljuk a (táblázatos) hibajavítás folyamatát!

Megoldás:

A mátrix baloldalán elhelyezkedő almátrixa nem egységmátrix, tehát a kód nem lehet szisztematikus.

A mátrix sorai mind kódszavak, s legtöbbjük csak három 1-es elemet tartalmaz (a súlyuk 3). E kódszavak távolsága a csupa 0-t tartalmazó kódszótól (amely minden lineáris kódnak eleme) mindössze 3. Következésképpen e kód minimális távolsága sem lehet háromnál nagyobb. Hogy nem is kisebb, ez csak akkor látszik, ha valamennyi kódszót előállítjuk, s kikeressük közülük a minimális súlyút. Az a kód, amelynek a kódtávolsága 3, 2 hibát jelezni, 1 hibát javítani tud.

A mátrix két utolsó sorát összeadva a baloldali egységmátrix létrehozható:

$$\mathbf{G}_{sz} = \begin{bmatrix} 1 & 0 & 0 & 00 & 0 & 1 & 1 \\ 0 & 1 & 0 & 00 & 1 & 1 & 0 \\ 0 & 0 & 1 & 01 & 1 & 0 & 0 \\ 0 & 0 & 0 & 11 & 0 & 0 & 1 \end{bmatrix}$$

A paritásellenőrző mátrix:

$$\mathbf{H} = \begin{bmatrix} 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Az \mathbf{u} üzenethez tartozó kódszó:

$$\mathbf{c}(\mathbf{u}) = \mathbf{u} \cdot \mathbf{G}_{sz} = [0 \quad 1 \quad 1 \quad 0 \quad 1 \quad 0 \quad 1 \quad 0].$$

A meghibásodott kódszó:

$$\hat{\mathbf{c}}(\mathbf{u}, \mathbf{e}) = [1 \quad 1 \quad 1 \quad 0 \quad 1 \quad 0 \quad 1 \quad 0].$$

Ezt szorozzuk a paritásellenőrző mátrix transzponáltjával, hogy előállítsuk a szindrómát:

$$\mathbf{s} = \hat{\mathbf{c}} \cdot \mathbf{H}^T = [0 \quad 0 \quad 1 \quad 1].$$

Már csak az lehet kérdés, melyik egyhibás hibavektornak ez a szindrómája. Természetesen annak, amelynek az első eleme 1. Érdemes megfigyelni, hogy az a kéthibás hibavektor, amelynek az *utolsó* két eleme 1, ugyanezt a szindrómát szolgáltatja.

4.7. Hibajelzés polinomkóddal

Tekintsük a (7,3) bináris polinomkódot, amelynek generátormátrixa

$$g(x) = x^4 + x^3 + x^2 + 1 !$$

Állapítsuk meg, ciklikus-e az a kód, amelyet ez a generátormátrix állít elő! Kövessük végig a szisztematikus kód kódszavának előállítását, feltéve, hogy az üzenetvektor $\mathbf{u} = [0 \quad 1 \quad 0]$! Vizsgáljuk meg a hibajelzés folyamatát is, a kódszó első bitjének meghibásodását feltételezve!

Megoldás:

A kód ciklikussága tekintetében az a meghatározó, hogy a generátorpolinom osztója-e az $x^7 + 1$ polinomnak, vagy sem. Az osztást elvégezve zérus a maradék, tehát a kód ciklikus.

Az üzenetvektornak megfelelő polinom

$$u(x) = x .$$

A szisztematikus kódszavakat (kódpolinomokat) úgy képezzük, hogy a megfelelő mértékben "fel"tolt (x^{n-k} -val megszorzott) üzenetpolinomot olyan függelékkel egészítjük ki, amely biztosítja, hogy az eredmény osztható legyen a generátorpolinommal. E függelék természetesen

$$p(x) = \text{rem} \frac{x^{n-k} \cdot u(x)}{g(x)}.$$

A számolást elvégezve $p(x) = 0 \cdot x^3 + 1 \cdot x^2 + 1 \cdot x + 1$ adódik. Így a keresett kódszó:

$$c = [0 \quad 1 \quad 0 \quad 0 \quad 1 \quad 1 \quad 1].$$

Az első bitjében hibás kódszónak megfelelő polinom:

$$\hat{c}(x) = x^6 + x^5 + x^2 + x + 1.$$

Képeznünk kell osztási maradékát a generátorpolinommal, s ez most

$$s(x) = x^3 + x^2 + x \neq 0.$$

A szindróma nem zéruspolinom, tehát történt hiba.

5. Modulációs eljárások

5.1. A lineáris modulációs lánc alapsávi ekvivalense

Bizonyítsa be, hogy az 5.1.ábra szerinti, $H_p(\cdot)$, $H_D(\cdot)$, $H_{Ch}(\cdot)$ átviteli függvényekkel és az F frekvenciával, mint vivőfrekvenciával jellemzett lineáris (szorzó demodulátoros) rendszer tulajdonképpen a

$$H_e(f) = H_p(f) \frac{H_{Ch}(f+F) + H_{Ch}(f-F)}{4} H_D(f)$$

átviteli függvényű lineáris transzformációt valósítja meg!

Megoldás

Kísérjük végig, mi történik a bemenőjel egyetlen, f frekvenciájú harmonikus (nem szinuszos, hanem komplex exponenciális) összetevőjével! Az 5.1.ábra egyes pontjain ezeket a jeleket tüntettük fel. Az áttekintés fontos tanulsága, hogy a bizonyítandó tétel csak korlátozással érvényes: nem ad számot a keletkező (demodulált) jel $f-2F$, illetve $f+2F$ frekvenciájú összetevőjéről. Egyúttal persze az is megállapítható, hogy ezek az összetevők *értelmes* paraméterértékek esetén elhanyagolhatóak.

Megjegyezhető, hogy a TV képjel átvitelére éppen ilyen modulációs eljárást használnak.

További megjegyzés: a vizsgált modell $H_{Ch}(\cdot)$ átviteli függvényű eleme azt sugallja, hogy a csatorna diszperzióját írja le. Ez így is van, de jól jegyezzük meg: a két *szorzó funkció közötti* csatornáról van szó, s így beleértendő az adó kimenő szűrője és a vevő bemenő szűrője is! Valójában tehát az *adószűrő*, az adószűrővel kialakított modulált jelet átvivő *átviteli csatorna* és a modulált jelet a sávon kívüli zajoktól (és más felhasználóktól származó jelektől) megtisztító *vevőszűrő* együtteséről van szó

5.1. ábra:

$$(i) - \left[\begin{array}{c} \text{előszűrő} \\ H_p(\cdot) \end{array} \right] - (2) - \left(\begin{array}{c} \text{szorzás} \\ \cos(2\pi Ft) \end{array} \right) - (3) - \left[\begin{array}{c} \text{adó csatorna} \\ \text{szűrő } H_{Ch}(\cdot) \\ \text{szűrő} \end{array} \right] - (4) - \left(\begin{array}{c} \text{szorzás} \\ \cos(2\pi Ft) \end{array} \right) - (5) - \left[\begin{array}{c} \text{utószűrő} \\ H_D(\cdot) \end{array} \right] - (6)$$

$$\begin{aligned} 1: & e^{j2\pi ft} \\ 2: & H_p(f) \cdot e^{j2\pi ft} \\ 3: & \frac{1}{2} H_p(f) \cdot e^{j2\pi(f+F)t} + \frac{1}{2} H_p(f) \cdot e^{j2\pi(f-F)t} \\ 4: & \frac{1}{2} H_p(f) \cdot H_{Ch}(f+F) \cdot e^{j2\pi(f+F)t} + \frac{1}{2} H_p(f) \cdot H_{Ch}(f-F) \cdot e^{j2\pi(f-F)t} \end{aligned}$$

5:

1
4

6: ezt tessék kitalálni!

5.2. Szögmodulált jel sávszélessége

Becsülje meg az FM jel a sávszélességét, ha a vivőfrekvencia 10 MHz, a moduláló jel $f_m = 20 \text{ kHz}$ frekvenciájú szinuszos jel, a (frekvencia)modulációs tényező (fázislököt) pedig $m_f = 3.4$!

Megoldás:

A tanult arany szabály szerint a modulált jelben ilyenkor (szinuszos moduláló jel esetén) azok az összetevők számottevőek, amelyek a vivőfrekvencia

$$\frac{B_s}{2} = \alpha(m_f) \cdot f_m$$

sugarú környezetében vannak. α értéke becsülhető a

$$\alpha = 1 + \sqrt{m_f} + m_f$$

képlettel. A behelyettesítést elvégezve esetünkben

$$B_s = 2 \cdot 20 \cdot (1 + \sqrt{3.4} + 3.4) \cong 250 \text{ kHz}$$

adódik.

Megjegyzés: ha kedvünk van, mondhatjuk a következő indoklást. A szögmodulált jel

$$s_{s\omega M}(t) = \cos(2\pi Ft + \mu_t) = \cos \mu_t \cdot \cos(2\pi Ft) - \sin \mu_t \cdot \sin(2\pi Ft)$$

alakjában μ jelenti a modulációs tartalmat, a vivőt az ő nemlineáris függvényei modulálják. Ha azonban μ véges (esetünkben az, hiszen a maximális értéke, m_f adott), akkor e véges szélességű tartomány felett ezek a nemlineáris függvények közelíthetők egy-egy polinommal. Az α paraméter éppen a kielégítő közelítést szolgáltató polinom fokszámával azonosítható. Ha tekintetbe vesszük, hogy egy közelítő polinomnak legalább a közelítendő függvény zérushelyeit helyesen kell szolgáltatnia, s hogy a szinusz- és a koszinuszfüggvény zérusai között π a távolság, α -ra durva becslés is adható. A közelítő polinom gyöktényezőinek száma legalább

$$\alpha(m_f) > \left\lceil 2 \cdot \frac{m_f}{\alpha} \right\rceil.$$

Ez természetesen csak egy alsó becslés, kísérletezni kellene, mekkora biztonsági faktorról számolva lehet a szinusz és koszinuszfüggvényeket egy véges szélességű tartomány felett polinommal jól közelíteni.

5.3. A kislökötű fázismodulátor nemlineáris torzítása

Tanultuk, hogy kislökötű szögmodulált (fázismodulált) jelek előállításának kézenfekvő módszere a

$$\eta_r = \cos(2\pi Ft) - \mu_r \cdot \sin(2\pi Ft)$$

szabály megvalósítása (AM-DSB/SC és 90 fokkal eltolt vivő). Ez a kifejezés nyilván a modulált jel kvadratúrafelbontása. Vizsgáljuk meg közelebbről, milyen is ez a modulált jel!

Megoldás:

Írjuk fel a modulált jel ún. modulációs felbontását! Ez valójában nem jelent mást, mint az eredő jel amplitúdójának és fázisának meghatározását. Tkp. középiskolai trigonometriai ismeretekből következik, hogy

$$\eta_r = \sqrt{1 + \mu_r^2} \cdot \cos(2\pi Ft + \arctan \mu_r).$$

Eredményünk tanulságai:

- járulékos AM jött létre, ez nem öröm, de nem is nagy baj, a frekvenciasokszorozón múlik, érzékeny-e a bemenő jel ezen fogyatékoságára.

- a modulációs tartalom van némi nemlineáris torzulás, hiszen nem μ_r , hanem $\arctan \mu_r$ tényleges tartalom. Az árkusz tangens függvény sorfejtésében az elsőfokú tag után harmadfokú következik, ha tehát a fázislököt 0.1 körüli, akkor a hiba mindössze néhány (kb. három) tízezred, relatív értékben néhány ezrelék. 0.5 (radiános) fázislökötnél a hiba már kb. 5 század, azaz 10 százalék.

- fordított értelmű tanulság is levonható. Ha előállítunk egy tisztességes fázismodulált jelet, majd sávját korlátozzuk, akkor a kislökötű jelhez hasonló szerkezetű jelet állítunk elő, csak a lökete nem kicsi. Következmény: nemlineáris torzítás.

5.4. A szimbólumközi áthallás elkerülése

Gondoljuk át, milyen jelzési sebességek esetén nincs szimbólumközi áthallás, ha az elemi jel spektruma

$$M(f) = \begin{cases} \frac{1}{2B} \cdot (1 - |f/B|), & \text{ha } |f| \leq B \\ 0, & \text{egyébként} \end{cases} !$$

Gondoljuk át azt is, mi a következménye annak, ha a jelzési sebesség kicsi az elemi jel sávszélességéhez képest!

Megoldás:

A megadott spektrumhoz tartozó időfüggvény is kitalálható, hiszen ez a háromszög egy felezélességű "téglalap" önmagával alkotott konvolúciója. Az időfüggvény tehát:

$$m(t) = \left(\frac{\sin(\pi Bt)}{\pi Bt} \right)^2.$$

Tekintve, hogy ennek a függvénynek nullhelyei $1/B$ egész sokszorosainál vannak, a jelzési idő (az időrések szélessége) is $1/B$ sokszorosa lehet, s ekkor nincs szimbólumközi áthallás.

A frekvenciatartományi kép alapján is erre a következtetésre jutunk. Ha ugyanis a jelzési sebesség éppen B , az egymásra lapolódó eltolt háromszögek éppen konstansra egészítik ki egymást. Ha a jelzési sebesség B -nek tört része, akkor még e konstansokat is egymásra kell lapolni, persze az összeg ekkor is állandó marad.

Ez a kép ad módot a második kérdés tisztázására. Ha a jelzési sebesség kicsi, akkor sok eltolt spektrum lapolódik egymásra, s bármilyen is a szélessávú spektrum alakja, sok egymásra lapolódó szeletének összege *közel ugyanaz* a konstans lesz. Minél több szelet játszik szerepet, ezek annál keskenyebbek, annál inkább konstans az eredő. Ha pedig kicsi az eredmény-függvény ingadozása, akkor az azt jelenti, hogy kicsi a megmaradt szimbólumközi áthallás is.

5.5. A szimbólumközi áthallás romboló hatása

Tételezzük fel, hogy az $m(\cdot)$ elemi jel valamely mintavételi fázisban ugyan kielégíti Nyquist feltételét (azaz T közötti mintái rendre $m_0 = 1, m_{\pm 1} = 0, m_{\pm 2} = 0, \dots$), de a mintavételi fázis időzítési hiba következtében elcsúszik, és a meghatározó jelminták $m_0 = 0.99, m_{+1} = 0.1, m_{-1} = -0.1, m_{\pm 2} = 0, \dots$ értékűek lesznek. Becsüljük meg a hibavalószínűséget meghatározó jel-zaj viszony leromlását 2, 4 és 8 szintű rendszerben!

Megoldás:

Feltételezzük, hogy a 2, 4 és 8 szintű rendszerben a jelamplitúdók rendre $\pm 1, \pm 1$ és ± 3 valamint $\pm 1, \pm 3, \pm 5$ és ± 7 értékűek. Mivel a jelamplitúdók közötti távolság azonos, azonos zajban e rendszerek hibavalószínűsége is összevethető (bár enyhén különböző, miért is?). Akkor, amikor nincs szimbólumközi áthallás, a hibavalószínűséget meghatározó jel-zaj arány $1/\sigma$, ahol σ a zajminták szórása. Az

időzítési hiba esetén az adatjel mintái elmosódnak, az aktuális időrés amplitúdója mellett ± 0.1 súlytényezővel a szomszédos időrések amplitúdói is "látszanak". Legrosszabb esetben a szomszédos amplitúdók abszolút értéke maximális, előjelük pedig olyan, hogy hatásuk ugyanabba az irányba tolja el a vett minta értékét. Az eltérő tartalmat hordozó jelminták távolsága így lecsökken, a fél-távolság $0.99 - d_{\max}(0.1+0.1)$ lesz. A zaj szórását a dolog nem érinti, ezért a jel-zaj viszony csökkenése rendre 0.79, 0.39-szeres. 8 szintű rendszerben nagy baj van: $7 \cdot 0.2 > 0.99$. Ez azt jelenti, hogy még zaj nélkül is hibás olykor a döntés.

Tanulság: a szimbólumközi áthallás mindig gondot jelent, de ez a gond többszintű rendszerben hatványozottan jelentkezik.

5.6. Példa a szimbólumközi áthallás kiegyenlítésére

Próbáljuk meg a 6.3. példában szereplő adatjelet egy háromegyütthatós FIR szűrővel kiegyenlíteni! Becsüljük meg a hibavalószínűséget meghatározó jel-zaj viszonyt!

Megoldás:

Tudományos terminológiával élve a "zero forcing" kiegyenlítési stratégiát választjuk. Tekintsük az adatjel három egymás utáni mintáját!

$$\begin{aligned} s_{k-1} &= -0.1 \cdot d_{k-2} + 0.99 \cdot d_{k-1} + 0.1 \cdot d_{k-0} + v_{k-1} \\ s_{k-0} &= -0.1 \cdot d_{k-1} + 0.99 \cdot d_{k-0} + 0.1 \cdot d_{k+1} + v_{k-0} \\ s_{k+1} &= -0.1 \cdot d_{k-0} + 0.99 \cdot d_{k+1} + 0.1 \cdot d_{k+2} + v_{k+1} \end{aligned}$$

Szorozzuk meg az első sort 0.1/0.99-el, a harmadikat pedig -0.1/0.99-el, majd adjuk hozzá őket a második sorhoz!

$$\begin{aligned} \hat{s}_k &= 0.101 \cdot s_{k-1} + s_{k-0} - 0.101 \cdot s_{k+1} = \\ &= -0.0101 \cdot d_{k-2} + 0.0 \cdot d_{k-1} + (0.99 + 0.0101 + 0.0101) \cdot d_{k-0} + 0.0 \cdot d_{k+1} - 0.0101 \cdot d_{k+2} + \\ &+ 0.0101 \cdot v_{k-1} + v_{k-0} - 0.0101 \cdot v_{k+1} \end{aligned}$$

Végeredményben:

$$\begin{aligned} \hat{s}_k &= -0.0101 \cdot d_{k-2} + 1.0102 \cdot d_{k-0} - 0.0101 \cdot d_{k+2} + \\ &+ 0.0101 \cdot v_{k-1} + v_{k-0} - 0.0101 \cdot v_{k+1} \end{aligned}$$

Jól látható, hogy beavatkozásunk nyomán a szimbólumközi áthallás jelentősen (kb. a tizedére) csökkent. Megfigyelhető, hogy d_{k-0} együtthatója is megváltozott (most éppen nőtt, de akár csökkenhetett volna is), s módosult a mintát zavaró zaj teljesítménye (szórása) is. Mindent összevetve a javulás óriási, a kiegyenlített rendszer 32 szintű átvitelre is alkalmas, hiszen $31 \cdot 0.0202 \cong 0.63 < 1.0102$.

Felmerülhet a kérdés, lehet-e tovább javítani (hiszen 64 szintű átvitelre még ez a rendszer sem alkalmas) a helyzetet. A válasz igenlő, több jelminta (pl. öt a három helyett) lineáris kombinációját képezve a második szomszédok hatása is eliminálható, persze ekkor a döglött ló átkerül a harmadik szomszédba. De mindenesetre kevesebb marad belőle.

5.7. Mely tényezők rontják a döntés biztonságát ...?

... szinkron PAM és QAM rendszerekben. Kiindulhatunk abból, hogy egyszerű szinkron PAM rendszerben a döntés időreseként történik, a vett (esetleg kiegyenlített) jelmintákat rögzített döntési küszöbökkel hasonlítjuk össze. A hibázások alapvető oka a jelmintákat terhelő zaj, de mint láttuk, a zaj iránti érzékenységet gyökeresen leronthatja a szimbólumközi áthallás. Az is igaz, hogy a szimbólumközi áthallás egyik forrása (általában azért nem a legjelentősebb forrása) a rossz mintavételi pozíció. Másféle hibaforrás, de említést érdemel az elemi jel fő mintájának eltérése a várt értéktől, s itt az is zavaró lehet, ha nagyobb, mint amire a döntőkészülék számít. Ez a hiba fölfogható egy alaperősítésbeli bizonytalanságnak, amely, ha elég nagy, önmagában is okozhat kiértékelési hibát. Pl. 8 szintű rendszerben 20%-os erősítéstöbblet az 5 amplitúdójú "tartalomból" kb. 6 egységnyi jelmintát termel, s ekkora mintákat a döntő már joggal vélhet 7 amplitúdójú tartalomból eredőnek. Már csak ezért is fontos hangsúlyozni: egy valamire való szinkron PAM vevőben azonos fontosságú az időzítéskinyerési, az automatikus erősítésszabályozási és a kiegyenlítési funkció. Megjegyezhető még, hogy ezek az egységek vica-versa képesek egymás hibáit tompítani.

A QAM rendszereket kényelmes két, egyidejűleg működő szinkron PAM rendszernek tekinteni. E két PAM rendszerben a jelamplitúdók (az időrések tartalma) lehetnek függetlenek (pl. igazi 4*4-es QAM), de lehetnek koordináltak is (pl. 8 fázisú PSK). E rendszerekben minőségileg új jelenség a két PAM jel közötti csatolás, a kvadratúracsatornák áthallása. A kiegyenlítőnek nem elég a szimbólumközi áthallás eltüntetésével foglalkozni, a másik csatornából áthallott törmelék is kompenzálnia kell. Másik jellegzetes kellemetlenség a demoduláló vivő fázis-, esetleg frekvenciahibája. Szerencsére van mód (hiszen az időrések tartalma diszkrét) az efféle hiba észlelésére és mérésére, s ez megteremti a védekezés lehetőségét is.

5. Modulációs eljárások

5.1. A lineáris modulációs lánc alapsávi ekvivalense

Bizonyítsa be, hogy az 5.1.ábra szerinti, $H_p(\cdot)$, $H_D(\cdot)$, $H_{Ch}(\cdot)$ átviteli függvényekkel és az F frekvenciával, mint vivőfrekvenciával jellemzett lineáris (szorzó demodulátoros) rendszer tulajdonképpen a

$$H_e(f) = H_p(f) \frac{H_{Ch}(f+F) + H_{Ch}(f-F)}{4} H_D(f)$$

átviteli függvényű lineáris transzformációt valósítja meg!

Megoldás

Kísérjük végig, mi történik a bemenőjel egyetlen, f frekvenciájú harmonikus (nem szinuszos, hanem komplex exponenciális) összetevőjével! Az 5.1.ábra egyes pontjain ezeket a jeleket tüntettük fel. Az áttekintés fontos tanulsága, hogy a bizonyítandó tétel csak korlátozással érvényes: nem ad számot a keletkező (demodulált) jel $f-2F$, illetve $f+2F$ frekvenciájú összetevőjéről. Egyúttal persze az is megállapítható, hogy ezek az összetevők *értelmes* paraméterértékek esetén elhanyagolhatóak.

Megjegyezhető, hogy a TV képjel átvitelére éppen ilyen modulációs eljárást használnak.

További megjegyzés: a vizsgált modell $H_{Ch}(\cdot)$ átviteli függvényű eleme azt sugallja, hogy a csatorna diszperzióját írja le. Ez így is van, de jól jegyezzük meg: a két *szorzó funkció közötti* csatornáról van szó, s így beleértendő az adó kimenő szűrője és a vevő bemenő szűrője is! Valójában tehát az *adószűrő*, az adószűrővel kialakított modulált jelet átvivő *átviteli csatorna* és a modulált jelet a sávon kívüli zajoktól (és más felhasználóktól származó jelektől) megtisztító *vevőszűrő* együtteséről van szó

5.1. ábra:

$$(i) - \left[\begin{array}{c} \text{előszűrő} \\ H_p(\cdot) \end{array} \right] - (2) - \left(\begin{array}{c} \text{szorzás} \\ \cos(2\pi Ft) \end{array} \right) - (3) - \left[\begin{array}{c} \text{adó csatorna vevő} \\ \text{szűrő } H_{Ch}(\cdot) \text{ szűrő} \end{array} \right] - (4) - \left(\begin{array}{c} \text{szorzás} \\ \cos(2\pi Ft) \end{array} \right) - (5) - \left[\begin{array}{c} \text{utószűrő} \\ H_D(\cdot) \end{array} \right] - (6)$$

$$1: e^{j2\pi ft}$$

$$2: H_p(f) \cdot e^{j2\pi ft}$$

$$3: \frac{1}{2} H_p(f) \cdot e^{j2\pi(f+F)t} + \frac{1}{2} H_p(f) \cdot e^{j2\pi(f-F)t}$$

$$4: \frac{1}{2} H_p(f) \cdot H_{Ch}(f+F) \cdot e^{j2\pi(f+F)t} + \frac{1}{2} H_p(f) \cdot H_{Ch}(f-F) \cdot e^{j2\pi(f-F)t}$$

5:

1
4

6: ezt tessék kitalálni!

5.2. Szögmodulált jel sávszélessége

Becsülje meg az FM jel a sávszélességét, ha a vivőfrekvencia 10 MHz, a moduláló jel $f_m = 20 \text{ kHz}$ frekvenciájú szinuszos jel, a (frekvencia)modulációs tényező (fázislöklet) pedig $m_f = 3.4$!

Megoldás:

A tanult arany szabály szerint a modulált jelben ilyenkor (szinuszos moduláló jel esetén) azok az összetevők számottevőek, amelyek a vivőfrekvencia

$$\frac{B_s}{2} = \alpha(m_f) \cdot f_m$$

sugarú környezetében vannak. α értéke becsülhető a

$$\alpha = 1 + \sqrt{m_f} + m_f$$

képlettel. A behelyettesítést elvégezve esetünkben

$$B_s = 2 \cdot 20 \cdot (1 + \sqrt{3.4} + 3.4) \cong 250 \text{ kHz}$$

adódik.

Megjegyzés: ha kedvünk van, mondhatjuk a következő indoklást. A szögmodulált jel

$$s_{s\omega M}(t) = \cos(2\pi Ft + \mu_t) = \cos \mu_t \cdot \cos(2\pi Ft) - \sin \mu_t \cdot \sin(2\pi Ft)$$

alakjában μ jelenti a modulációs tartalmat, a vivőt az ω nemlineáris függvényei modulálják. Ha azonban μ véges (esetünkben az, hiszen a maximális értéke, m_f adott), akkor e véges szélességű tartomány felett ezek a nemlineáris függvények közelíthetők egy-egy polinommal. Az α paraméter éppen a kielégítő közelítést szolgáltató polinom fokszámával azonosítható. Ha tekintetbe vesszük, hogy egy közelítő polinomnak legalább a közelítendő függvény zérushelyeit helyesen kell szolgáltatnia, s hogy a szinusz- és a koszinuszfüggvény zérusai között π a távolság, α -ra durva becslés is adható. A közelítő polinom gyöktényezőinek száma legalább

$$\square(m_f) > \left\lceil 2 \cdot \frac{m_f}{\square} \right\rceil.$$

Ez természetesen csak egy alsó becslés, kísérletezni kellene, mekkora biztonsági faktorial számolva lehet a szinusz és koszinuszfüggvényeket egy véges szélességű tartomány felett polinommal jól közelíteni.

5.3. A kislöketű fázismodulátor nemlineáris torzítása

Tanultuk, hogy kislöketű szögmodulált (fázismodulált) jelek előállításának kézenfekvő módszere a

$$\eta_t = \cos(2\pi Ft) - \mu_t \cdot \sin(2\pi Ft)$$

szabály megvalósítása (AM-DSB/SC és 90 fokkal eltolt vivő). Ez a kifejezés nyilván a modulált jel kvadratúrafelbontása. Vizsgáljuk meg közelebbről, milyen is ez a modulált jel!

Megoldás:

Írjuk fel a modulált jel ún. modulációs felbontását! Ez valójában nem jelent mást, mint az eredő jel amplitúdójának és fázisának meghatározását. Tkp. középiskolai trigonometriai ismeretekből következik, hogy

$$\eta_t = \sqrt{1 + \mu_t^2} \cdot \cos(2\pi Ft + \arctan \mu_t).$$

Eredményünk tanulságai:

- járulékos AM jött létre, ez nem öröm, de nem is nagy baj, a frekvenciasokszorozón múlik, érzékeny-e a bemenő jel ezen fogyatékoságára.

- a modulációs tartalomban van némi nemlineáris torzulás, hiszen nem μ_t , hanem $\arctan \mu_t$ tényleges tartalom. Az árkusz tangens függvény sorfejtésében az elsőfokú tag után harmadfokú következik, ha tehát a fázislöket 0.1 körüli, akkor a hiba mindössze néhány (kb. három) tízezred, relatív értékben néhány ezrelék. 0.5 (radiános) fázislöketnél a hiba már kb. 5 század, azaz 10 százalék.

- fordított értelmű tanulság is levonható. Ha előállítunk egy tisztességes fázismodulált jelet, majd sávját korlátozzuk, akkor a kislöketű jelhez hasonló szerkezetű jelet állítunk elő, csak a lökete nem kicsi. Következmény: nemlineáris torzítás.

5.4. A szimbólumközi áthallás elkerülése

Gondoljuk át, milyen jelzési sebességek esetén nincs szimbólumközi áthallás, ha az elemi jel spektruma

$$M(f) = \begin{cases} \frac{1}{2B} \cdot (1 - |f/B|), & \text{ha } |f| \leq B \\ 0, & \text{egyébként} \end{cases} !$$

Gondoljuk át azt is, mi a következménye annak, ha a jelzési sebesség kicsi az elemi jel sávszélességéhez képest!

Megoldás:

A megadott spektrumhoz tartozó időfüggvény is kitalálható, hiszen ez a háromszög egy felezélességű "téglalap" önmagával alkotott konvolúciója. Az időfüggvény tehát:

$$m(t) = \left(\frac{\sin(\square Bt)}{\square Bt} \right)^2.$$

Tekintve, hogy ennek a függvénynek nullhelyei $1/B$ egész sokszorosainál vannak, a jelzési idő (az időrések szélessége) is $1/B$ sokszorosa lehet, s ekkor nincs szimbólumközi áthallás.

A frekvenciatartományi kép alapján is erre a következtetésre jutunk. Ha ugyanis a jelzési sebesség éppen B , az egymásra lapolódó eltolt háromszögek éppen konstansra egészítik ki egymást. Ha a jelzési sebesség B -nek tört része, akkor még e konstansokat is egymásra kell lapolni, persze az összeg ekkor is állandó marad.

Ez a kép ad módot a második kérdés tisztázására. Ha a jelzési sebesség kicsi, akkor sok eltolt spektrum lapolódik egymásra, s bármilyen is a szélessávú spektrum alakja, sok egymásra lapolódó szeletének összege *közel ugyanaz* a konstans lesz. Minél több szelet játszik szerepet, ezek annál keskenyebbek, annál inkább konstans az eredő. Ha pedig kicsi az eredmény-függvény ingadozása, akkor az azt jelenti, hogy kicsi a megmaradt szimbólumközi áthallás is.

5.5. A szimbólumközi áthallás romboló hatása

Tételezzük fel, hogy az $m(\cdot)$ elemi jel valamely mintavételi fázisban ugyan kielégíti Nyquist feltételét (azaz T közü mintái rendre $m_0 = 1, m_{\pm 1} = 0, m_{\pm 2} = 0, \dots$), de a mintavételi fázis időzítési hiba következtében elcsúszik, és a meghatározó jelminták $m_0 = 0.99, m_{+1} = 0.1, m_{-1} = -0.1, m_{\pm 2} = 0, \dots$ értékűek lesznek. Becsüljük meg a hibavalószínűséget meghatározó jel-zaj viszony leromlását 2, 4 és 8 szintű rendszerben!

Megoldás:

Feltételezzük, hogy a 2, 4 és 8 szintű rendszerben a jelamplitúdók rendre $\pm 1, \pm 1$ és ± 3 valamint $\pm 1, \pm 3, \pm 5$ és ± 7 értékűek. Mivel a jelamplitúdók közötti távolság azonos, azonos zajban e rendszerek hibavalószínűsége is összevethető (bár enyhén különböző, miért is?). Akkor, amikor nincs szimbólumközi áthallás, a hibavalószínűséget meghatározó jel-zaj arány $1/\sigma$, ahol σ a zajminták szórása. Az

időzítési hiba esetén az adatjel mintái elmosódnak, az aktuális időrés amplitúdója mellett ± 0.1 súlytényezővel a szomszédos időrések amplitúdói is "látszanak". Legrosszabb esetben a szomszédos amplitúdók abszolút értéke maximális, előjelük pedig olyan, hogy hatásuk ugyanabba az irányba tolja el a vett minta értékét. Az eltérő tartalmat hordozó jelminták távolsága így lecsökken, a fél-távolság $0.99 - d_{\max} (0.1 + 0.1)$ lesz. A zaj szórását a dolog nem érinti, ezért a jel-zaj viszony csökkenése rendre 0.79, 0.39-szeres. 8 szintű rendszerben nagy baj van: $7 \cdot 0.2 > 0.99$. Ez azt jelenti, hogy még zaj nélkül is hibás olykor a döntés.

Tanulság: a szimbólumközi áthallás mindig gondot jelent, de ez a gond többszintű rendszerben hatványozottan jelentkezik.

5.6. Példa a szimbólumközi áthallás kiegyenlítésére

Próbáljuk meg a 6.3. példában szereplő adatjelet egy háromegetűthetős FIR szűrővel kiegyenlíteni! Becsüljük meg a hibavalószínűséget meghatározó jel-zaj viszonyt!

Megoldás:

Tudományos terminológiával élve a "zero forcing" kiegyenlítési stratégiát választjuk. Tekintsük az adatjel három egymás utáni mintáját!

$$\begin{aligned} s_{k-1} &= -0.1 \cdot d_{k-2} + 0.99 \cdot d_{k-1} + 0.1 \cdot d_{k-0} + v_{k-1} \\ s_{k-0} &= -0.1 \cdot d_{k-1} + 0.99 \cdot d_{k-0} + 0.1 \cdot d_{k+1} + v_{k-0} \\ s_{k+1} &= -0.1 \cdot d_{k-0} + 0.99 \cdot d_{k+1} + 0.1 \cdot d_{k+2} + v_{k+1} \end{aligned}$$

Szorozzuk meg az első sort 0.1/0.99-el, a harmadikat pedig -0.1/0.99-el, majd adjuk hozzá őket a második sorhoz!

$$\begin{aligned} \hat{s}_k &= 0.101 \cdot s_{k-1} + s_{k-0} - 0.101 \cdot s_{k+1} = \\ &= -0.0101 \cdot d_{k-2} + 0.0 \cdot d_{k-1} + (0.99 + 0.0101 + 0.0101) \cdot d_{k-0} + 0.0 \cdot d_{k+1} - 0.0101 \cdot d_{k+2} + \\ &+ 0.0101 \cdot v_{k-1} + v_{k-0} - 0.0101 \cdot v_{k+1} \end{aligned}$$

Végeredményben:

$$\begin{aligned} \hat{s}_k &= -0.0101 \cdot d_{k-2} + 1.0102 \cdot d_{k-0} - 0.0101 \cdot d_{k+2} + \\ &+ 0.0101 \cdot v_{k-1} + v_{k-0} - 0.0101 \cdot v_{k+1} \end{aligned}$$

Jól látható, hogy beavatkozásunk nyomán a szimbólumközi áthallás jelentősen (kb. a tizedére) csökkent. Megfigyelhető, hogy d_{k-0} együtthatója is megváltozott (most éppen nőtt, de akár csökkenhetett volna is), s módosult a mintát zavaró zaj teljesítménye (szórása) is. Mindent összevetve a javulás óriási, a kiegyenlített rendszer 32 szintű átvitelre is alkalmas, hiszen $31 \cdot 0.0202 \cong 0.63 < 1.0102$.

Felmerülhet a kérdés, lehet-e tovább javítani (hiszen 64 szintű átvitelre még ez a rendszer sem alkalmas) a helyzetet. A válasz igenlő, több jelminta (pl. öt a három helyett) lineáris kombinációját képezve a második szomszédok hatása is eliminálható, persze ekkor a döglött ló átkerül a harmadik szomszédba. De mindenesetre kevesebb marad belőle.

5.7. Mely tényezők rontják a döntés biztonságát ...?

... szinkron PAM és QAM rendszerekben. Kiindulhatunk abból, hogy egyszerű szinkron PAM rendszerben a döntés időreseként történik, a vett (esetleg kiegyenlített) jelmintákat rögzített döntési küszöbökkel hasonlítjuk össze. A hibázások alapvető oka a jelmintákat terhelő zaj, de mint láttuk, a zaj iránti érzékenységet gyökeresen leronthatja a szimbólumközi áthallás. Az is igaz, hogy a szimbólumközi áthallás egyik forrása (általában azért nem a legjelentősebb forrása) a rossz mintavételi pozíció. Másféle hibaforrás, de említést érdemel az elemi jel fő mintájának eltérése a várt értéktől, s itt az is zavaró lehet, ha nagyobb, mint amire a döntőkészülék számít. Ez a hiba fölfogható egy alaperősítésbeli bizonytalanságnak, amely, ha elég nagy, önmagában is okozhat kiértékelési hibát. Pl. 8 szintű rendszerben 20%-os erősítéstöbblet az 5 amplitúdójú "tartalomtól" kb. 6 egységnyi jelmintát termel, s ekkora mintákat a döntő már joggal vélhet 7 amplitúdójú tartalomtól eredőnek. Már csak ezért is fontos hangsúlyozni: egy valamire való szinkron PAM vevőben azonos fontosságú az időzítéskinyerési, az automatikus erősítésszabályozási és a kiegyenlítési funkció. Megjegyezhető még, hogy ezek az egységek vica-versa képesek egymás hibáit tompítani.

A QAM rendszereket kényelmes két, egyidejűleg működő szinkron PAM rendszernek tekinteni. E két PAM rendszerben a jelamplitúdók (az időresek tartalma) lehetnek függetlenek (pl. igazi 4*4-es QAM), de lehetnek koordináltak is (pl. 8 fázisú PSK). E rendszerekben minőségileg új jelenség a két PAM jel közötti csatolás, a kvadratúracsatornák áthallása. A kiegyenlítőnek nem elég a szimbólumközi áthallás eltüntetésével foglalkozni, a másik csatornából áthallott törmelék is kompenzálnia kell. Másik jellegzetes kellemetlenség a demoduláló vivő fázis-, esetleg frekvenciahibája. Szerencsére van mód (hiszen az időresek tartalma diszkrét) az efféle hiba észlelésére és mérésére, s ez megteremti a védekezés lehetőségét is.

1. Feladat

Szubjektív hangminőség tesztek alapján a következő Jel/Interferencia értékeket kell betartani a keskenysávú FM analóg rendszereknél.

A rendszer: $S/I \geq 17$ dB, Csatorna sávszélesség $B_c=30$ kHz.

B rendszer: $S/I \geq 18$ dB, Csatorna sávszélesség $B_c=25$ kHz.

C rendszer: $S/I \geq 23$ dB, Csatorna sávszélesség $B_c=12.5$ kHz.

A GSM rendszernél, mely digitális GMSK modulációt használ, ezek az értékek:

GSM: $S/I \geq 9$ dB, Csatorna sávszélesség $B=200$ kHz. (Egy vivőn 8 TDMA hozzáférés történik)

A rendelkezésünkre álló frekvenciasáv $B=25$ MHz.

Feladat:

Határozza meg az egyes rendszerek szükséges clusterméretét.

Vesse össze az egyes rendszereket a minimális clustermérettel megvalósított lefedettség mellett kapacitás szempontjából.

Megoldás:

Körsugárzó bázisállomás antennákat használva az előadás anyag alapján a legközelebbi 6 azonos csatornás interferenciát keltő cella hatására létrejövő S/I

$$\frac{S}{I} = \frac{R^{-n}}{\sum_{i=1}^6 (D_i)^{-n}} = \frac{(D/R)^n}{6} = \frac{(\sqrt{3N})^n}{6} = \frac{Q^4}{6}$$

$$N=i^2+ij+j^2$$

Ezen kifejezéseket felhasználva:

A rendszer:

$$\frac{S}{I} = 50.1 \leq \frac{(D/R)^4}{6} = \frac{(\sqrt{3N})^4}{6} = \frac{Q^4}{6}$$

$$N \geq \frac{(\sqrt[4]{S/I \cdot 6})^2}{3} = \frac{(\sqrt{S/I \cdot 6})}{3} = \frac{\sqrt{50.1 \cdot 6}}{3} = 5.78$$

$N=7$ cellából álló rendszert választunk, az egy cellában alkalmazható vivőfrekvenciák száma (ha a vezérlő csatornák alkalmazásától eltekintünk)

$C=W/N=(B/B_c)/N=(25/0.03)/7=119$ csatornát használhatunk minden cellában egyidejűleg.

Végezzük el a számításokat a másik két analóg rendszerre:

B rendszer:

$$N \geq 6.48$$

Válasszunk $N=7$ rendszert.

Ekkor a kapacitás $C=(25/0.025)/7=142$

C rendszer:

$$N \geq 11.5$$

Válasszunk $N=12$ rendszert. Mutassuk meg, hogy ez az $N=9$ után a következő lehetséges clusterméret.

Ekkor a kapacitás $C=(25/0.0125)/12=166$.

Az előző S/I feltételvizsgálatokat az előadás következő táblázata alapján is elvégezhetjük volna.

	N	Q	S/I	C=W/N
i=1, j=1	3	3	11.3	W/3
i=2, j=0	4	3.46	13.78	W/4
i=2, j=1	7	4.58	18.65	W/7
i=3, j=0	9	5.2	20.86	W/9
i=2, j=2	12	6	23.34	W/12
i=3, j=1	13	6.24	24.03	W/13

Az analóg rendszerek közül tehát a C rendszer nyújtja a legjobb kapacitást. (Feltéve persze, ha 12.5 kHz sávszélességen hasonló minőséget nyújt, mint a másik két rendszer.)

A GSM rendszernél először végezzük el a hasonló vizsgálatot.

A táblázatból is megállapítható, hogy az előírt $S/I=9$ dB értéket az $N=3$ cluster teljesíti.

Ekkor a kapacitás

$$C=(25/0.2)/3=41.6=41$$

Tudjuk azonban, hogy egy vivőn 8 TDMA hozzáférés történik, tehát a kapacitás

$$C'=41 \cdot 8=328$$

Ezen vizsgálat alapján a GSM rendszer sokkal jobb, mint az előző három analóg rendszer.

Viszont.

Itt terjedelem miatt nem részletezendő okok miatt (vezérlő csatornák száma, cellák flexibilisebb kezelése) szektorizálást alkalmaznak.

Végezzük el az előző számításokat 120 ill. 60°-os szektorokra, $N=3$ clusterre.

	N	Q	S/I	W/N
i=1, j=1	3	3	14.3 (120°)	W/3/3
i=1, j=1	3	3	16.07 (60°)	W/3/6

120°-os szektorizálás

Az S/I viszony ekkor a táblázatból 14.3 dB, a kapacitás pedig $C=328/3=109$

60°-os szektorizálás

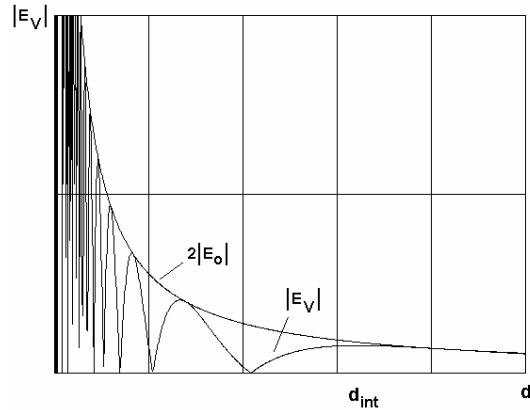
Az S/I viszony ekkor a táblázatból 16.07 dB, a kapacitás pedig $C=328/6=54$

Ezen vizsgálat alapján megállapíthatjuk, hogy a rendszer kapacitása (természetesen nagyon kiváló interferencia védettség mellett) rosszabb lett, mint az előző rendszereké.

A rádiócsatorna

1. Mozgó rádióösszeköttetés térerőssége

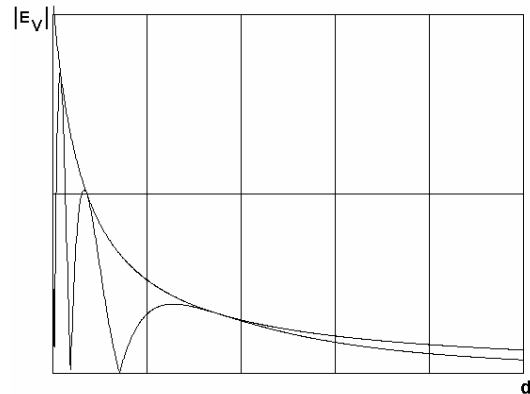
Az $|E_V|$ térerősséget ábrázoljuk a d szakasztávolság függvényében.



2.5. ábra Kétutas rádióösszeköttetés térerőssége

A rádiószakasznak az állandóhelyű antenna és d_{int} távolság közötti részét interferencia zónának nevezzük, ahol mint az a 2.5. ábrán jól látható, a térerősség minimum és maximumhelyei váltva követik egymást. Az interferencia zónán kívül a térerősség $1/d^2$ -tel arányos, szemben a szabadtéri rádióösszeköttetés $1/d$ -vel arányos térerősségével.

Ennek láthatóvá tételére nagyítsuk ki a 2.5. ábra jobb oldali tartományát. (2.6. ábra)



2.6. ábra Kétutas rádióösszeköttetés térerőssége

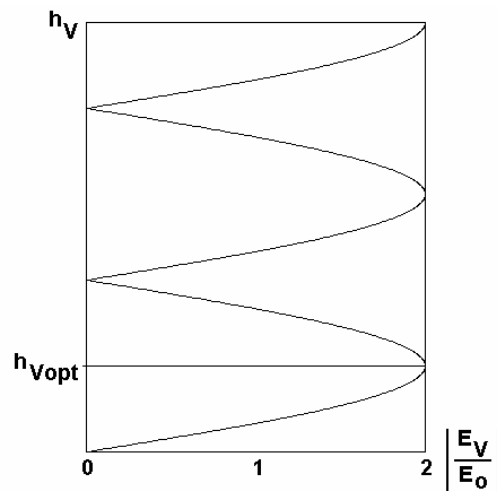
Az interferencia zóna határának kiszámításához vizsgáljuk meg a (2.20) kifejezés szinusz függvényének argumentumát. Az interferencia zóna határát az adja, ahol az argumentum $\pi/2$ -vel egyenlő.

$$\frac{2\pi}{\lambda} \frac{h_A h_V}{d_{int}} = \frac{\pi}{2} \quad (2.21)$$

$$d_{int} = \frac{4h_A h_V}{\lambda} \quad (2.22)$$

2. Állandóhelyű rádióösszeköttetés

Állandóhelyű rádióösszeköttetéseknel a cél az optimális vevőantenna magasság meghatározása.



2.7. ábra Állandóhelyű rádióösszeköttetés térerőssége

Az optimális vevőantenna magasságot ugyancsak a (2.22) összefüggésből kapjuk, innen

$$h_{vopt} = \frac{\lambda d}{4h_A} \quad (2.23)$$

3. Kétutas terjedés - szelektív fading

A kétutas terjedés hatása és a frekvenciaszelektív fading hatásának bemutatása:

A rádiócsatorna feszültség átviteli függvénye az előzőekben bemutatott mechanizmus következtében a következő lesz:

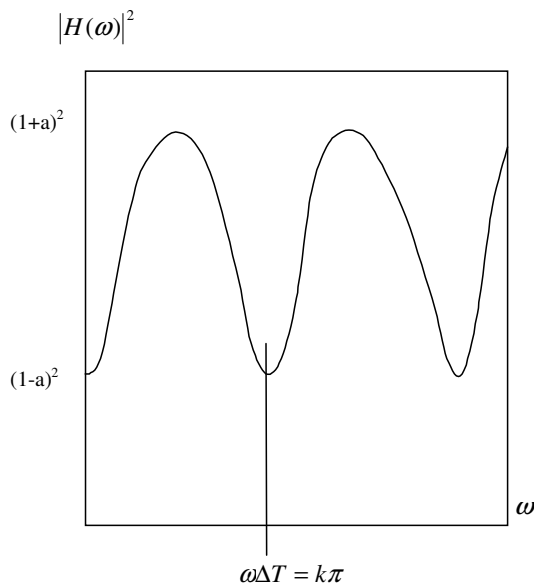
$$H(\omega) = 1 + a \cdot e^{j\omega\Delta T}$$

ahol

a a reflektált hullám relatív amplitudója a közvetlen hullámhoz képest

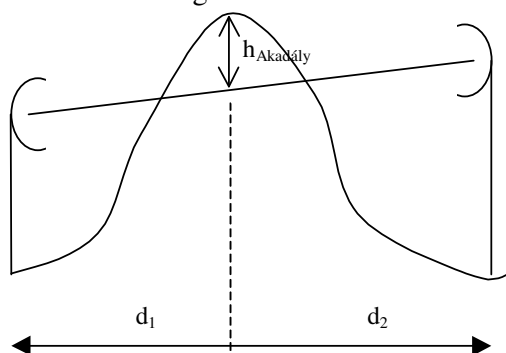
ΔT a reflektált hullám beérkezési késleltetése

$$|H(\omega)|^2 = 1 + a^2 + 2a \cdot \cos \omega\Delta T$$



4. Diffrakció

Határozzuk meg a vevőantennánál mérhető térerősséget a következő geometriára:



A feladatot a késél diffrakciós modellel oldjuk meg, a késél relatív benyúlása

$$\nu = h_{\text{Akadaly}} \sqrt{\frac{2(d_1 + d_2)}{\lambda d_1 d_2}}$$

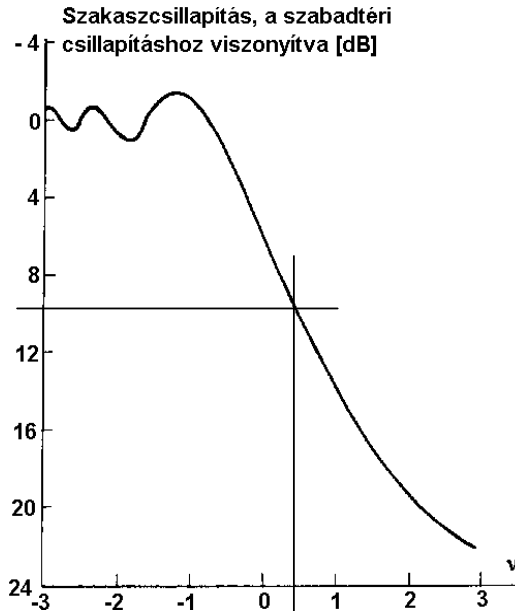
A feladat adatai:

$$h_{\text{Akadaly}} = 20 \text{ m}$$

$$d_1 = d_2 = 10 \text{ km}$$

$$\text{üzemi frekvencia} = 300 \text{ MHz}$$

$$\nu = 20 \sqrt{\frac{2 \cdot 20 \cdot 10^3}{1 \cdot 10 \cdot 10^3 \cdot 10 \cdot 10^3}} = 0.4$$



A szabadtéri terjedés télerősségéhez viszonyítva 10 dB többletsillapítást okoz az akadály.

Így a télerősség a vevőantennánál:

$$E = E_0 / L = \frac{\sqrt{60P_A G_A}}{(d_1 + d_2)} \frac{1}{L} = \frac{\sqrt{60 \cdot 1 \cdot 1}}{20 \cdot 10^3} \frac{1}{3.16} = 122.6 \mu V / m$$

5. Rádióösszeköttetés jel/zaj viszonya

1. Feladat

Határozzuk meg a levezető kábelből és egy előerősítőből álló rendszer zajtényezőjét mindkét sorrendű összekapcsolás esetén. Adatok: kábel hossza 15 m, fajlagos csillapítása 1 dB/m, hőmérséklete 290 K, erősítő zajtényezője 3 dB, erősítése 20 dB.

Megoldás:

A 15 m hosszúságú kábel

$$L = 15 \text{ m} \cdot 1 \text{ dB/m} = 15 \text{ dB}$$

csillapítást okoz. Az erősítő zajtényezője viszonzszámban:

$$F_a = 3 \text{ dB} = 10^{0.3} = 1.995$$

A kábel zajtényezője megegyezik annak csillapításával (mivel hőmérséklete 290 K), továbbá dB-ben kifejezett erősítése megegyezik a dB-ben megadott csillapításának -1 szeresével.

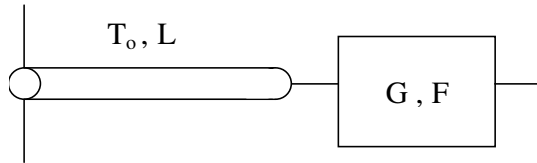
$$F_c = L = 15 \text{ dB} = 31.62$$

$$G_c = \frac{1}{L} = \frac{1}{31.62}$$

A láncba kapcsolt erősítők eredő zajtényezője:

$$F = F_1 + \frac{F_2 - 1}{G_1}$$

Most vizsgáljuk meg először a kábel + erősítő együttest



Behelyettesítve a kábel paramétereit (zajtényező= L , erősítés= $1/L$)

$$F_{c \rightarrow a} = F_c + \frac{F_a - 1}{G_c} = LF_a$$

összefüggés adódik. Az eredő zajtényező:

$$15 + 3 = 18 \text{ dB}$$

(Általános szabály: ha egy csillapító kapcsolódik egy átviteli elem elé, akkor az eredő zajtényező a csillapítással növekszik (a csillapító hőmérséklete 290 K)).

A másik sorrendű összekapcsolásnál:

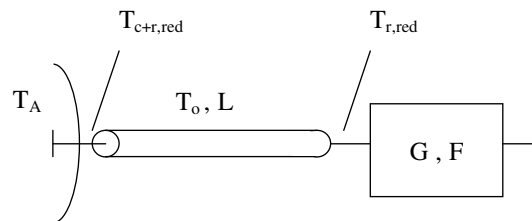
$$F_{a \rightarrow c} = F_a + \frac{F_c - 1}{G_a} = 1.995 + \frac{31.62 - 1}{100} = 2.301 = 3.6 \text{ dB}$$

Következtetés: a kábel csillapítása által okozott zajtényező növekedést már nem lehet kiszárni erősítővel sem kompenzálni.

2. Feladat

Mekkora romlást okoz az eredő zajhőmérsékletben az antennát és az előerősítőt összekötő 1 dB csillapítású, 290 K hőmérsékletű kábel, ha az antenna saját zajhőmérséklete 20 K? Az előerősítő bemenetétől mérve a vevő zajtényezője 0.5 dB. Adjuk meg a romlást dB-ben.

Megoldás:



A vevő zajtényezője

$$F_r = 10^{0.05} = 1.122$$

Az összekötő kábel nélküli eset:

$$T_{r,red} = T_0(F_r - 1) = 290(1.122 - 1) = 35.39 \text{ K}$$

A vevő bemenetére redukált zajhőmérséklete összeadódik az antenna zajhőmérsékletével:

$$T_1 = T_a + T_{r,red} = 20 + 35.39 = 55.39 \text{ K}$$

A kábel beiktatásakor:

Az előző példa alapján a rendszer zajtényezője a csillapítással romlik, 1.5 dB-lel növekszik.

$$F_{c+r} = 10^{0.15} = 1.413$$

Az előzőekhez hasonlóan:

$$T_{c+r,red} = T_0(F_{c+r} - 1) = 290(1.413 - 1) = 119.8 \text{ K}$$

$$T_2 = T_a + T_{c+r,red} = 20 + 119.8 = 139.3 \text{ K}$$

A kábel beiktatása miatti romlás így:

$$10 \log \frac{T_2}{T_1} = 10 \log \frac{139.3}{55.39} = 4 \text{ dB}$$

3. Feladat

Egy rádióösszeköttetés szükséges adóteljesítményét kell meghatározni.

A jel/zaj viszony 30 dB, a terjedésből adódó fading tartalékra 40 dB-t tervezünk.

A további adatok: üzemi frekvencia 10 GHz, szakasztávolság 30 km, az adó- és vevőantenna nyeresége egyaránt 40 dB, a vevőantenna zajhőmérséklete 290 K, a sáv szélesség 20 MHz, a vevő zajtényezője 3 dB.

Megoldás:

A szabadtéri csillapítás

$$a_0 = 20 \log \left(\frac{4\pi d}{\lambda} \right) - (G_a + G_v)$$

Ezt a fading tartalékkal kell növelnünk

$$a_{sz} = a_0 + 40 \text{ dB} = 61.98 + 40 = 102 \text{ dB}$$

A rendszer zajhőmérséklete:

$$T_e = T_A + (F_v - 1)T_0 = F_v T_0$$

A zajteljesítmény a vevő bemenetén:

$$P_z = kBT_e = -204 + 10 \log B + 10 \log (T_e/T_0) = -204 + 73 + 3 = -128 \text{ dBW}$$

A szükséges adóteljesítmény így:

$$P_A = (S/N) + a_{sz} + P_z = 30 + 102 + (-128) = 4 \text{ dBW} = 2.5 \text{ W}$$