En modellering av enkel utjämnare i MATLAB

Ioannis Tilaveridis
930901-9215

Kirill Narmack
931119-0715

18 December 2013

Introduktion

Projektuppgiften är att konstruera en enkel utjämnare (eller equalizer), som eliminerar brus vid signalöverföring. Ett exempel på sådant brus som programmet ska eliminera är att en person kan höra sin röst i ett telefonsamtal.

Teori

Inom telekommunikation uppstår brus pga att radiovågen studsar mot något föremål istället för att direkt gå från sändare till mottagare. Detta resulterar i att mottagaren får tidsförskjutna och amplitudsskadade versioner av samma utsända signal.

Problembeskrivning

Man vill överföra en tidskontinuerlig meddelandesignal x(t) som bara kan anta värdena 0, 1, 2, 3. Dessutom vet man att den mottagna datasignalen y(t) ges av:

$$y(t) = A_0 x(t) + A_1 x(t-\beta) + e(t) \qquad \textbf{(1)}$$

där e(t) är brus som adderas under överföringen.
En känd för mottagaren testsignal skickas ut innan meddelandesignalen x(t):
$x_{test}(t) = \sin(2\pi f_0 t)$ under ett nT tidsintervall, där T är signalens periodtid.

Motsvarande mottagna sampelsignal:

$$y_{test}(t) = A_0 x_{test}(t) + A_1 x_{test}(t-\beta) + e(t) \qquad \textbf{(2)}$$

Problemlösning

Man kan återskapa x(t) genom att Fourier-Tansformera sambandet mellan signalerna till frekvensplanet. Fouriertransformen av **(1)** ger oss:
$Y(f) = A_0 X(f) + A_1 X(f) e^{-j2\pi f\beta}$ där bruset e(t) försummas.

Då fås: $Y(f) = X(f)(A_0 + A_1 e^{-j2\pi f\beta})$ eller $Y(f) = X(f)H(f)$ **(3)**

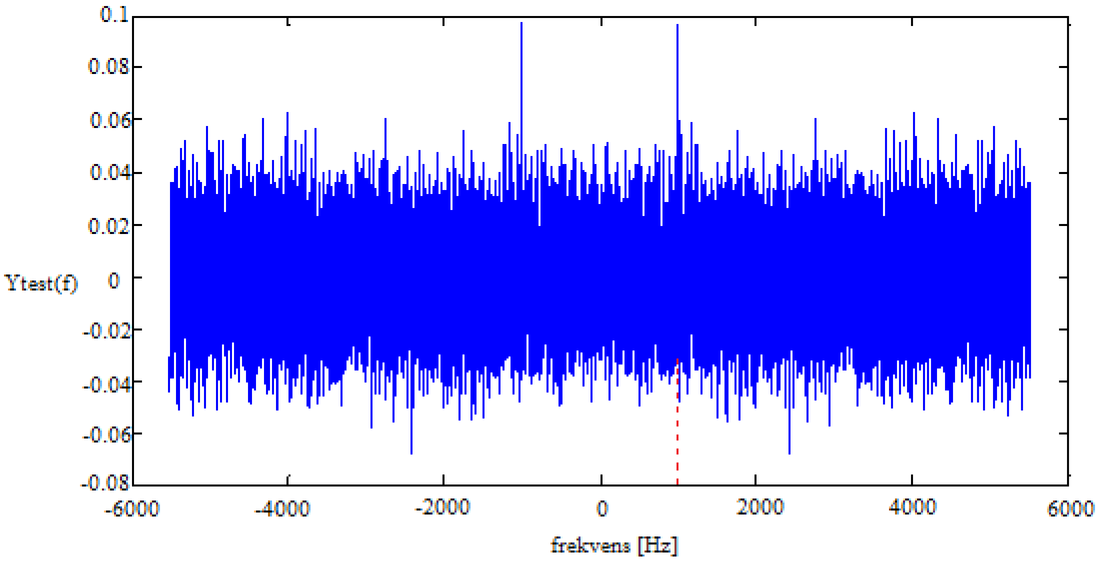
där $H(f) = (A_0 + A_1 e^{-j2\pi f\beta})$ är ett funktionsuttryck av A_0 , A_1 , β.

Ur **(3)** fås: $(f) = \frac{Y_{test}(f)}{X_{test}(f)}$ **(5)** där $X_{test}(f) \neq 0$

$$x_{test}(t) = \begin{cases} \sin(2\pi f_0 t), & 0 \le t \le P \\ 0, & för\ övrigt \end{cases} \quad \overset{FT}{\rightarrow} \quad X_{test}(f) = \frac{1}{2j}(\delta(f - f_0) - \delta(f + f_0))$$

De δ-pulser är 0 för alla f förutom för $f = \pm f_0$. Just då får man ett nollskilt värde på $X_{test}(f)$.

För att bestämma värdet på f_0 kan man utgå från figuren för Ytest(f) :



Eftersom MATLAB börjar söka från $-\infty$ till $+\infty$ går det att finna det maximala $Y_{test}(f)$ värdet, vilket motsvarar $-f_0$ (första värdet) och $+f_0$ (andra värdet). Då kan man antigen finna $-f_0$ och ta den negativa versionen av det, eller sätta alla negativa frekvenser till noll och börja söka från 0 till $+\infty$. Enligt figuren kan man ungefär se vart f_0 kommer hamna i (röda sträckan).

Det kommer finnas en relation mellan FT av $x_{test}(t)$ och FS av en period av $x_{test}(t)$:

$x_{test}(t) = \sin(2\pi f_0 t)$ där $t \in [0, P]$

Fourier-Transformen av $x_{test}(t)$: $X_{test}(f) = \int_{-\infty}^{\infty} x_{test}(t) \cdot e^{-j2\pi f t} dt$

Fourier-Serien av $x_{test}(t)$: $x_{test}(t) = \sum_{n=-\infty}^{\infty} c_n \cdot e^{j2\pi f n t}$

där $c_n = \frac{1}{P} \int_{1\ Period} x_{test}(t) \cdot e^{-j2\pi f n t} dt$

Då blir: $c_n = \frac{1}{T} \cdot X_{test}(nf)$ där $n \in [0, N]$ ty $x_{test}(t) = 0$ utanför en periods intervall $[-\frac{T}{2}, \frac{T}{2}]$.

Alltså FS-koefficienterna c_n är värdena av FT samplade på ett rutnät av bredd $\frac{1}{T}$ (eller multiplicerad med ett rutnät av bredd $\frac{1}{T}$) .

Man kan få fram A_0 och A_1 med hjälp av $H(f_0)$ på följande sätt :

$H(f_0) = (A_0 + A_1 e^{-j2\pi f_0 \beta})$ där man vet att $H(f_0)$ är komplex.
Det blir: $Re\{H(f_0)\} + j\, Im\{H(f_0)\} = A_0 + A_1 \cos(2\pi\beta f_0) - j(A_1 sin(2\pi\beta f_0))$

Då fås: $\begin{cases} Re\{H(f_0)\} = A_0 + A_1 \cos(2\pi\beta f_0) \\ Im\{H(f_0)\} = -A_1 sin(2\pi\beta f_0) \end{cases}$ eller: $\begin{cases} A_0 = A_1 \cos(2\pi\beta f_0) - Re\{H(f_0)\} \\ A_1 = -\frac{Im\{H(f_0)\}}{sin(2\pi\beta f_0)} \end{cases}$

De trösklade signalerna x(t) och y(t) kan plottas i MATLAB :