

# Adaptív kiegyenlítési eljárás rádiócsatornán történő adatátvitelre

PHAM THI HUONG

BME Mikrohullámú Híradástechnikai Tanszék

## ÖSSZEFOGLALÁS

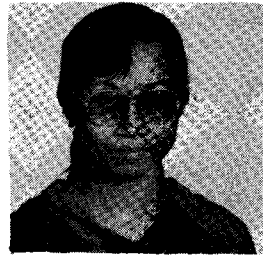
A rövidhullámú rádiócsatornákon 1200 bit/s vagy ennél nagyobb sebességű adatátvitel ki van téve káros hatásoknak mint például különböző lineáris torzítás, gyors csatornaváltozás és különböző fading. Ebben a cikkben a döntésvisszacsatolt kiegyenlítővel foglalkozunk, amellyel kiegészítve a vevő alkalmas lesz megbízható vételre. A cikk a legkisebb átlag négyzetes (Least Mean Square) kiegyenlítő működését tárgyalja többutas, fadinges csatornákon, levezetve egy zárt formulát a stabil működés feltételére. Továbbá a cikk összehasonlítja a legkisebb átlag négyzetes (LMS) és rekurzív legkisebb négyzetes (Recursive Least Square, RLS) kiegyenlítő működését a csatornákon és új kiegyenlítési eljárást javasol, mely kihasználja az RLS kiegyenlítő gyors konvergálás és az LMS kiegyenlítő jó stabilitási tulajdonságát, az eredményeket egy többutas, számítógépszimulált csatornán való futással támasztja alá.

## I. Bevezetés

A rádiócsatorna rövid- és ultrarövidhullámú sávja újabban az érdeklődés középpontjába került, mint nagy távolságú digitális adatátviteli lehetőség, miután a VLSI technológia fejlődése lehetővé teszi a megbízható, nem túl bonyolult és nem túl drága modemek tervezését. A rövidhullámú csatornán azonban több torzításfajta lép fel, elsősorban a többutas hullámterjedés miatt. A többutas terjedés úgy fogható fel, mint egy átvitel több, különböző relatív amplitudóval és késleltetési idővel rendelkező csatornák egy csoportja. Ez az átviteli közeg okozza többek között a szelektív fadinget, a Doppler fadinget, a Doppler eltolódást és szimbólumközi interferenciát. Kézenfekvő, hogy a rövidhullámú sávon (továbbiakban RH) modemekben valamilyen kiegyenlítési eljárást kell alkalmazni, mely ki tudja küszöbölni -vagy legalább csökkenteni a fenti csatornajelenségeket.

A csatorna frekvenciaszelektivitása miatt valamilyen nemlineáris eljárást érdemes alkalmaznunk mint döntésvisszacsatolt kiegyenlítést vagy maximum likelihood becslést (MLSE, vagy más néven Viterbi algoritmus). A hatékonysága ellenére a MLSE vevő bonyolultsága korlátozza az alkalmazását, ráadásul az összehasonlító tanulmányok a tényleges RH rádiócsatornán nem mutatták ki az MLSE vevő lényeges felsőbbségét a döntésvisszacsatolt kiegyenlítőhöz képest [1]. Az utóbbiaknak mérsékelt bonyolultsága és jó csatorna-követési képessége indokolja alkalmazását az idődiszperzív többutas csatornán, mint például az RH csatorna.

Ismert, hogy a legkisebb négyzetes hibafüggvényen alapuló adaptív jelfeldolgozó algoritmusok (angolul



PHAM THI HUONG

1961-ben született. 1985-ben szerezte meg a villamosmérnöki oklevelet a Műszaki Egyetem Mikrohullámú Híradástechnikai Tanszékén 1987-től ösztöndíjas aspiráns. Érdeklődési köre a digitális adatátvitel és jelfeldolgozás, hírközlési, hullámterjedési elmélet.

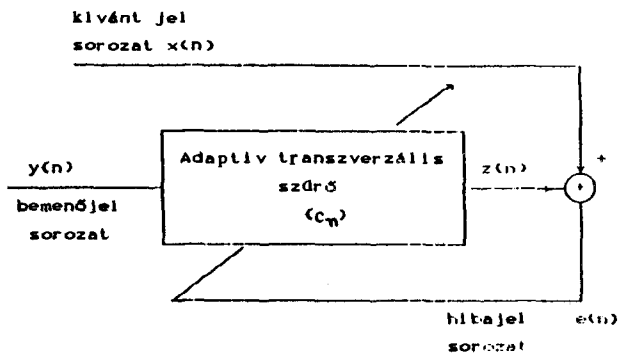
Least Squared) nagyon gyorsan konvergálnak és kitűnő a követési képességük egy ismeretlen, időben változó vektorokhoz. Ezen algoritmusok stacionárius állapotbeli viselkedésének részletes analízise található Eleftheriou és Falconer munkáiban [1, 2, 6]. A legkisebb átlagnégyzetes hibájú algoritmusok (angolul Least Mean Squared) konvergálási sebessége pedig lassú, különösen ha nagy a bemeneti kovarianciamátrix sajátértékeinek szórása [3, 4], azonban jó a stabilitási tulajdonságuk. Az alkalmazásuk azokon a csatornákon, ahol a csatornaváltozás lassú a jelsebességhez képest, a kielégítő határfokuk és egyszerű kivitelezhetőségük miatt indokolt. Gitlin [6] levezette a stabil működésük feltételét időinvariáns csatornákra.

Ez a cikk megpróbálja az LMS kiegyenlítő viselkedésének idődiszperzív csatornákon, a tranziens működés analízisének alapos megértését adni, megadja a stabil működési feltételt ezekre a csatornákra. Továbbá egyszerű és hatékony eljárást tartalmaz adaptív kiegyenlítésre viszonylag lassan változó csatornákon történő rövid üzenetek átvitelére.

A cikk felépítése a következő. A II. fejezet röviden leírja az LMS kiegyenlítő alapját, a viselkedését az időinvariáns csatornán. A III. fejezet megtárgyalja a tranziens viselkedését időben változó csatornán, levezetve stabil működés feltételét erre a csatornára. A IV. fejezetben kerül sor az RLS kiegyenlítő előnyeinek és hátrányainak tárgyalására, majd ezek után ismertet egy új eljárást, mely kihasználja az RLS algoritmus gyors konvergáltságát és az LMS jó stabilitási tulajdonságát. Az V. fejezet a számítógépes szimulációval kapott kísérleti eredményeket tartalmazza.

## II. Az LMS kiegyenlítő és becslési hibája

Ebben a fejezetben röviden leírom az LMS adaptív probléma alapját. Az 1. ábra mutatja az általános adaptív szűrő struktúráját. Itt a szűrő transzverzális elrendezésű, a késleltető elemek száma  $N$ , így  $N$  dimenziójú vektor lesz  $y$ , a késleltető vonalban tárolt minták



1. ábra. Adaptív szűrő rendszer

H577-1

vektora, és  $c_k$  a kiegyenlítő megcsapolásainak vektora  $x_n$ , az ún. kívánt impulzusválasz. A kiegyenlítő kimenete a  $k$ -adik időpontban  $z_k = y_k^T \cdot c_k$  ( $T$  jelöli a vektor transzponálását), amely eltérhet az ideális kimenettől,  $x_n$ -től. A hibajel:

$$e_k = y_k^T \cdot c_k - x_k$$

Az LMS adaptáló algoritmus  $c$  változtatásával minimalizálja az átlag-négyzetes hibát:

$$e_k = M |y_k^T \cdot c_k - x_k|^2 \quad (1)$$

ahol  $M$  jelöli az átlagolást az  $x_n$  sorozaton keresztül.

Az adaptálási algoritmus a legnagyobb meredekség módszerével számítva:

$$c_{k+1} = c_k - \Delta \cdot \frac{\partial e_k}{\partial c_k}$$

ahol  $\Delta$  az algoritmus léptéke. Az ún. legkisebb átlag-négyzetes gradiens algoritmust kapjuk, ha helyettesítjük a determinisztikus gradienst annak torzítatlan, de zajos becslésével,  $y_k(y_k^T \cdot c_k - x_k)$ -vel. Így a  $c_k$  vektort szabályozzuk a következő szerint:

$$c_{k+1} = c_k - \Delta \theta_k (y_k^T \cdot c_k - x_k) = c_k - \Delta y_k e_k \quad (2)$$

Az LMS algoritmus átlag négyzeteshibája (1. kifejezés) két fő forrásból jön létre. Az első hibaforrás a gradiens becslésének hibájára vonatkozik. Ezt a hibát "becslési hibának" hívjuk és abból adódik, hogy a kiszámított megcsapolási vektor eltér az optimális értékétől. A második hibaforrás a nemstacionárius jelek szűrésére vonatkozik. Ezt a hibát "késési hibának" hívjuk és az okozza, hogy az adaptív rendszer igyekszik követni a bemenő jel változását. A harmadik hibaforrás, mely a véges pontosságú számításokból adódik a digitális szűrőknél, - ellentétben a rekurzív algoritmusokkal - az LMS szűrőnél nem számottevő, így itt nem kell figyelembe venni [2].

Ha kivonjuk (2) kifejezést mindkét oldalából  $c_{opt}$ -t, azt kapjuk:

$$(c_{k+1} - c_{opt}) = (I - \Delta y_k y_k^T)(c_k - c_{opt}) - \Delta y_k e_{kopt} \quad (3)$$

ahol  $I$  az egységmátrix,

$$e_{kopt} = y_k^T c_{opt} - x_k$$

Ebből kiindulva Gitlin [5] független  $y_k$  feltételezésével a következő rekurzív összefüggést kapta az átlag-négyzetes hibára:

$$e_{\Delta k+1} = [1 - 2\Delta\lambda + \Delta^2 N \rho \lambda^{-2}] e_{\Delta k} + \Delta^2 \epsilon_{min} N \rho \lambda^{-2} \quad (4)$$

ahol  $\lambda$  a kovarianciamátrix átlag sajátértéke, ami megegyezik az átlagjeltejesítménnyel a kiegyenlítő bemenetén és  $\zeta = \lambda_{max}/\lambda$ . Ebből a kifejezésből azt látjuk, hogy az LMS algoritmus konvergál, ha első tag szorzó tényezőjének abszolút értéke kisebb egynél, vagyis:

$$0 < \Delta < \frac{2}{N\rho\lambda}$$

Ez az LMS algoritmus stabil működésének feltétele az idővariáns csatornán, amivel Qureshinél [3] is találkoztunk.

### III. Az LMS kiegyenlítő csatornakövetési képessége

Ezek után megvizsgáltam az LMS algoritmus viselkedését abban az esetben, amikor a csatorna paramétereiben időben változnak, így  $c_{opt} = A^{-1}v$  is. Az előző fejezetben a 3. kifejezést csak akkor kapjuk, ha a bal oldalon szereplő  $k+1$ -edik időponthoz tartozó  $c_{opt}$  egyenlő a jobb oldalon szereplő  $k$ -adik időhöz tartozó  $c_{opt}$ -tal. Így (5) kifejezésbeli konvergálási feltétel is csak az idővariáns csatornára érvényes. Az időben változó csatornán ez a feltétel drasztikusan megváltozik, a felső határ csökken, az alsó határ pedig nő.

Az algoritmus csatornakövetési képességét a késési hiba határozza meg:

$$e_{késés} = M [(c_k - c_{opt}(k))^T A^{-1} (c_k - c_{opt}(k))]^2$$

Ha ismerjük  $c_{opt}(k)$  pontos időfüggvényét, ki tudjuk számítani a késési hibát is. Itt azonban általánosan csak a stabil feltételt vezessük le az időben változó csatornára.

A  $c_{opt}$  változásától származó négyzetes maradékhibát egy iterációra a következőképpen számíthatjuk:

$$e_k = [c_{opt}(k+1) - c_{opt}(k)]^T A^{-1} [c_{opt}(k+1) - c_{opt}(k)]$$

és nevezzük  $e_{max}$ -nak a legnagyobb  $e_k$ -t, vagyis a csatorna leggyorsabb változása által okozott hibát. (4) egyenlet szerint egy iteráció alatt a maradék négyzetes hiba  $e_{\Delta k} - e_{\Delta k+1} = [2\Delta\lambda - \Delta^2 N \zeta \lambda^{-2}] e_{\Delta k} - \Delta^2 \epsilon_{min} N \zeta \lambda^{-2}$ -t változik. Az algoritmus konvergálási feltétele így:

$$e_{\Delta k} - e_{\Delta k+1} > c_{max}$$

Ha elhanyagoljuk az optimális beállítás esetén nyert négyzetes hibát  $\epsilon_{\min}$ -ot és legyen  $\alpha$  egyenlő  $\epsilon_{\max} / \epsilon_{\Delta k}$ -val ( $\alpha$  pozitív), akkor:

$$2\Delta\lambda \mid \Delta^2 N \zeta \lambda^{-2} > \alpha$$

Ennek megoldása pedig:

$$\frac{1 - \sqrt{1 - \alpha N \rho}}{N \rho \lambda} < \Delta < \frac{1 + \sqrt{1 - \alpha N \rho}}{N \rho \lambda} \quad (6)$$

Tehát az idődiszperzív csatorna esetén a (5) konvergálási feltétel a fenti kifejezésre módosul, ahol  $\alpha$  egy csatornafüggő paraméter.

#### IV. Az RLS kiegyenlítő konvergálási sebessége

Most nézzük a rekurzív legkisebb négyzetes hibájú algoritmus viselkedését az idődiszperzív csatornán.

A legkisebb négyzetes hibájú algoritmus minimalizálja a következő függvényt:

$$z_n = \sum_{k=0}^n \omega^{n-k} |x_k - y_k^T \cdot c_n|^2 \quad (7)$$

ahol  $\omega$  pozitív és kisebb egynél, amelyet az algoritmus csatornakövetési képessége miatt vezetünk be és exponenciális tényezőnek hívjuk.

Mint látjuk [6]-ban, az RLS algoritmusok konvergálása nagyon gyors. Független bemenő jel feltételezésével  $N$  iteráció után ( $N$  a transzverzális szűrő hossza) a  $c$  megcsapolási vektor eléri az optimális értékét. A csatorna változásainak követési képessége is kitűnő. Az átlag négyzetes hiba egy geometriai sor szerint konvergál, melynek időállandója nem függ a kovarianciamátrix sajátértékeitől és egyenlő az algoritmus "memóriájával",  $1/(1-\omega)$  értékeivel [2].

Az RLS algoritmus hátránya a számítási bonyolultság. Itt  $N \times N$  mátrix elemeit kell számítani és tárolni, így az iterációként szükséges számítások száma arányos  $N^2$ -tel. Azonkívül a digitális alkalmazásnál a véges pontosságú számítások miatt gyakran instabil, különösen amikor az exponenciális tényező  $\omega$  kisebb 1-nél (pl.  $\omega = 0,9$ ) [2, 3]. Ilyen szempontból a legkisebb átlag négyzetes algoritmus (LMS) nagyon stabil, viszont ez az algoritmus nagyon lassan konvergál, különösen, ha  $\zeta$  sokkal nagyobb 1-nél, így a csatorna változásainak követési képessége korlátozott [1]. Az LMS algoritmus az optimális  $\Delta$  léptékkel sem konvergál olyan gyorsan, mint legkisebb négyzetes algoritmus. Az előbb tárgyalt indokok miatt a kísérletekben együtt kerül sor az LMS és az RLS algoritmus alkalmazására úgy, hogy kihasználhatóbbá válik az RLS algoritmus gyorsan konvergáló tulajdonsága és az LMS algoritmus jó stabilitása.

A beállításnál az RLS algoritmussal számítjuk ki az optimális megcsapolási vektort  $c_{\text{opt}}$ , majd  $N$  iteráció után ( $N$  a kiegyenlítő hossza), amikor  $c = c_{\text{opt}}$  [6], akkor áttérünk az LMS algoritmusra. Az RLS algoritmus

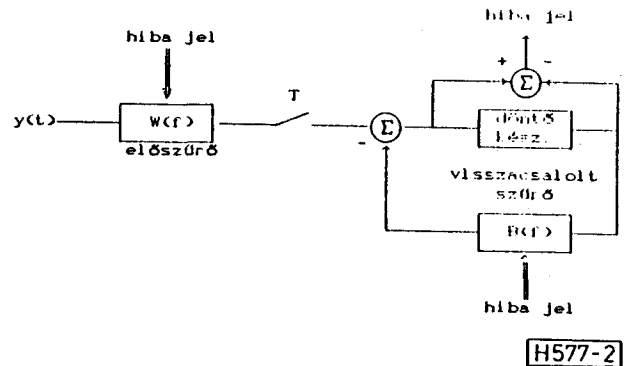
alatt  $\omega = 1$ -t használjuk, hiszen a beállításnál a csatornakövetési szempont nem fontos. Az átkapcsolás után az LMS algoritmusnál a  $\Delta$  lépték hatása a konvergálási sebességre már nem olyan kritikus mint a hagyományos LMS-nél, hiszen akkorra a beállítás már megtörtént.

Ilyen módon egy olyan algoritmust kapunk, amely gyorsan konvergál és stabil. A számítási igény az RLS algoritmushoz képest szintén lecsökken.

A következő fejezetben azt fogjuk látni a kísérletekben, hogy az átkapcsolás zökkenőmentesen megy végbe, nem okoz semmilyen instabilitást.

#### V. Szimulációs eredmények

A fenti fejezetekben leírt elmélet számítógépes szimulációval lett tesztelve. A vizsgált adaptív szűrő egy döntésvisszacsatolt transzverzális szűrő, melynek struktúráját a 2. ábrán láthatjuk. Ha az előszűrő megcsapolásai éppen szimbólumidejű távolságra vannak egymástól, akkor szinkron előszűrőről beszélünk, ha ennél kisebb a távolság (pl.  $T/2$ ) akkor felosztott előszűrőről beszélünk. A kísérletben mind szinkron előszűrőjű, mind  $T/2$  felosztott előszűrőjű változatnál a mintavételezési frekvencia 1200 Hz, a  $T/2$  szűrőnél



2. ábra. Döntésvisszacsatolt szűrő

pedig 2400 Hz. A szűrő hossza  $N=8$  (az előszűrő és visszacsatolt szűrő hossza egyaránt 4).

Az RH csatorna alapsávú ekvivalense egy késleltető vonallal modellezhető, melynek megcsapolási tényezői időben változnak. A kísérletben kétutas modellt használtam, az egyik út állandó amplitúdójú, a másik szinusz-négyzet függvényrel változik (használt fading-frekvencia 0,5 Hz). A késleltetési idő a két út között valamint a fadingmélység változik. A fadinges csatorna időfüggvénye a következő:

$$f(t) = b_1 + b_2 [1 - \text{csin}^2(2\pi \cdot f_{\text{fad}} \cdot t - \tau)]$$

ahol  $b_1$  és  $b_2$  a két út amplitúdója,

$c$  és  $\tau$  a második út fading mélysége és késleltetési ideje,

és a fading frekvenciája  $f_{\text{fad}} = 0,5$  Hz.

A szűrő bemenő jelét úgy kaptam, hogy egy tetszőleges karaktorsorozat ASCII kódjának QAM modulált jele a művonalon lett továbbítva. A kapott jelhez továbbá nulla várható értékű Gauss zajt adunk (a hasz-

nált jel-zaj viszony 10 dB), az így kapott jel a szűrő bemenetét szolgálja.

Az értékeléshez az egyes iteráció négyzetes hibáját rögzítettem és rajzoltattam ki.

#### A. Az LMS kiegyenlítő csatorna-követési képessége

Az LMS adaptálási algoritmust használva különböző algoritmus léptékekkel azt tapasztaltam, hogy a csatorna-követési képesség szorosan függ az algoritmus léptéktől, a kiegyenlítő nem stabil nagyon kis algoritmus léptéknél. Kis algoritmus léptékekkel a kiegyenlítő csatorna-követési képessége csökken, a megcsapolási tényező vektora mintegy "lemarad" az optimális értékektől. Ezt bizonyítja a (6) kifejezés helyességét a (5)-sel szemben. A (6) kifejezésben szereplő /alfa/ mennyiséget sajnos nem tudtam meghatározni, így a  $\Delta_{max}$  és  $\Delta_{min}$  határértékeket csak kísérletezéssel tudtam meghatározni.

1. táblázat

LMS kiegyenlítő léptékének határai

Csatorna	szinkron		T/2	
	$\delta_{min}$	$\delta_{max}$	$\delta_{min}$	$\delta_{max}$
c=0.3, /tau/=0.125T	0.02	0.1	0.01	0.55
c=0.3, /tau/=0.25T	0.02	0.1	0.01	0.06
c=0.3, /tau/=0.5 T	0.02	0.1	0.01	0.06
c=0.3, /tau/=1.0 T	0.02	0.1	0.01	0.055
c=0.6, /tau/=1.0 T	0.02	0.09	0.02	0.05
c=0.99, /tau/=1.0T	0.025	0.08	0.025	0.045

Az 1. táblázatban feltüntettem néhány jellegzetes kísérleti csatorna minimális és maximális megengedett léptékeit mind szinkron, mind T/2 felosztott előszűrő-jű kiegyenlítőre. A táblázatban szereplő  $\tau$  a késleltetési idő T szimbólumidőben kifejezve, c a fadingmélység.

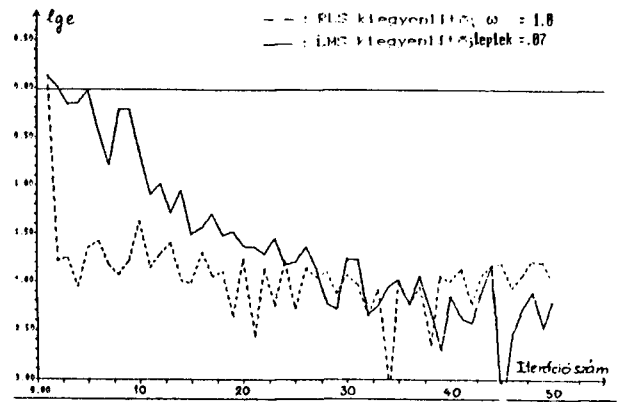
A táblázatból a következőket figyelhetjük meg: az algoritmus lépték határértékei nem változnak (vagy kevésbé változnak) a késleltetési idővel, viszont változik jelentősen a fadingmélységgel, vagyis a csatorna változási sebességgel (azonos fadingfrekvenciánál minél nagyobb a fadingmélység, annál nagyobb a csatornaváltozás bizonyos idő alatt). Ez szintén (6) kifejezés helyességére enged következtetni, hiszen /alfa/ arányos a csatorna változásával.

A kísérleti futtatásnál azt figyeltük még, hogy a legyorsabb kezdeti beállást eredményező algoritmus

lépték kb.  $\Delta_{közép} = \frac{\Delta_{max} + \Delta_{min}}{2}$  -nél található, ami a (6)

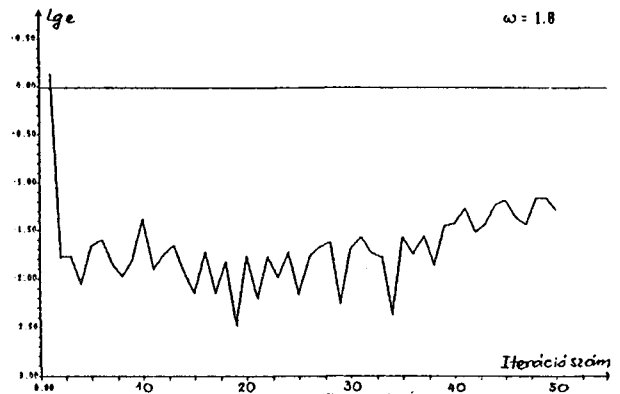
kifejezés szerint egyenlő  $1/N \cdot \zeta \lambda$  értékével.

A programot a nem fadinges csatornán is lefuttattam. Érdekes megfigyelni, hogy az LMS kiegyenlítő a fadinges közegben jobban működik, mint a nemfadinges közegben.



H577-3

3. ábra. Az LMS és RLS kiegyenlítő összehasonlítása konvergálási szempontból



H577-4

4. ábra. Az RLS kiegyenlítő divergálása (/tau/=0.5, c=0.99)

2. táblázat

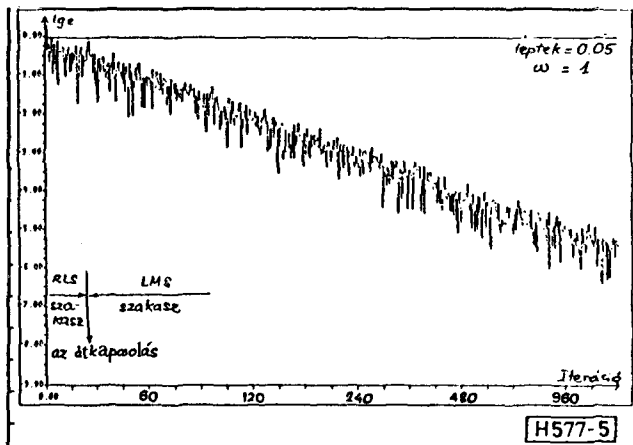
RLS-LMS kiegyenlítő léptékhatárai és átlag négyzetes hibái

Csatorna	$\delta_{min}$	$\delta_{max}$ ( $\delta_{min}$ )	$\epsilon_{\hat{a}}$ ( $\delta_{max}$ )	$\epsilon_{\hat{a}}$ ( $\delta_{közép}$ )	$\epsilon_{\hat{a}}$
c=0.3, $\tau=0.125T$	0.005	0.1	0.58	0.63	0.56
c=0.3, $\tau=0.25T$	0.005	0.08	0.89	0.55	0.51
c=0.3, $\tau=0.5T$	0.005	0.04	0.56	0.55	0.55
c=0.3, $\tau=1.0T$	0.01	0.04	0.55	0.54	0.54
c=0.6, $\tau=1.0T$	0.01	0.04	0.55	0.54	0.54
c=0.99, $\tau=1.0T$	0.01	0.04	0.55	0.55	0.54

#### B. Az RLS algoritmussal induló LMS kiegyenlítési eljárás

Itt leírt kísérletekben megvizsgáltam a 4. fejezetben leírt eljárás felsőbbiségét az LMS és RLS algoritmus-hoz képest. A két algoritmust együtt alkalmazó eljárást a továbbiakban RLS-LMS eljárásnak fogom hívni.

A 3. ábrán összehasonlítottam a leggyorsabban beálló LMS és RLS kiegyenlítő maradék négyzetes hiba görbéjének a konvergálását ugyan azon a csatornán ( $\tau=0.125T$ , c=0.3), a vízszintes tengely az iterációk



5. ábra. Az RLS-LMS kiegyenlítő viselkedése fadinges csatornán ( $\tau = 12.5\%T$ ,  $c = 0.3$ ,  $\# = 0.05$ )

számát mutatja. Az ábrán azt látjuk, hogy míg az RLSDS kiegyenlítő 4-5. iteráció után beáll (az előszűrő hossza 4), addig az LMS kiegyenlítő kb. 30. iterációnál áll be.

Viszont a rekurzív kiegyenlítőnek nagyon rossz a stabilitása. Egy idő múlva a négyzetes hiba görbéje kezd divergálni, különösen, ha nagy a futási időkülönbség a két út között. A 4. ábra mutat egy ilyen görbét az RLS kiegyenlítőnél ( $\tau = 0.5$ ,  $c = 0.99$ ). Ez a divergálás a véges pontosságú számításokból adódhat [1] (a programokban 16 bites lebegő pontosságú számábrázolást használtam).

Tehát a rekurzív kiegyenlítők nagyon gyorsan konvergálnak (a konvergálási idő körülbelül a kiegyenlítő előszűrőjének a feltöltési ideje) de nagyon rossz a stabilitása. Az LMS kiegyenlítőnek pedig jó a stabilitása de lassan konvergál.

Az RLS-LMS eljárás megvalósításánál  $n = 8$  iterációnál történik az átkapcsolás. Az eljárás stabilitásának

megállapítása végett megvizsgáltam az első 16 lépés négyzetes hibáját valamint kiszámítottam ezeknek az átlagát. Az eljárás szintén különböző  $\Delta$  léptékekkel történik. A 2. táblázatban feltüntettem a szinkron kiegyenlítővel kapcsolatos eredményeket: különböző csatornára kapott  $\Delta_{\max}$ ,  $D_{\min}$ ,  $D_{\text{közep}}$  értékeket, ezekhez tartozó  $\epsilon_{\text{átlag}}$ -ot az első 16 lépés után.

A táblázatból a következőt figyelhetjük meg: a  $\Delta$  lépték változtatásával az LMS szakaszban nem kapunk javulást a beállítás szempontjából. Ez érthető is, mivel az RLS szakaszban a kiegyenlítő lényegében már beáll. Ennél az eljárásnál a csatorna követési szempont a fontosabb a  $\Delta$  beállításnál. Az 5. ábra egy teljes futtatás hibagörbéjét tartalmazza ( $\tau = 5T$ ,  $c = 0.99$ ). Az ábrán látszik, hogy nincs instabilitás az átkapcsolásnál, a beállítás pedig már az RLS szakaszban megtörtént.

#### IRODALOM

- [1] E. Eleftheriou-D. Falconer: Adaptive Equalization Techniques for HF Channels, IEEE Jour. on Sel. Areas in Commun., vol. SAC-5, Feb. 1987, pp. 238-246.
- [2] E. Eleftheriou-D. Falconer: Tracking Property and Steady State Performance of RLS Adaptive Filter Algorithms, IEEE Trans on Acoust., Speech and Sig. Proc., vol. ASSP-34, Oct. 1986, pp. 1097-1110.
- [3] Sahid U. H. Qureshi: Adaptive Equalization, Proceeding of IEEE, vol. 73, Sept. 1985. pp. 1349-1387.
- [4] G. Ungerboeck: Theory on The Speed of Communication, IBM Journal Res. Devel., vol. 16, Nov. 1972. pp. 546-555.
- [5] R. D. Gitlin-S. B. Weinstein: On the Required Tap Weight Precision for Digitally-Implemented Adaptive Equalizer, Bell Syst. Tech J., vol. 58, Feb. 1979, pp. 301-321.
- [6] M. S. Mueller: On the Rapid Initial Convergence of Least Square Equalizer Adjustment Algorithms, Bell Syst. Tech. J., vol. 60, Dec 1981, pp. 2345-2358.