Adaptív digitális szűrés II.

DR. VARGA IMRE
BME Híradástechnikai Elektronika Intézet

ÖSSZEFOGLALÁS

A cikk második és befejező része a FIR szűrőkre a legkisebb négyzetek módszerén alapuló rekurzív algoritmusokat mutatja be részletesen, külön figyelmet fordítva az O(N) számításigényű gyors változatokra. Végül az adaptív IIR szűrők problémáit tekinti át

5. Legkisebb négyzetek módszere: az RLS (recnrsive least-squares) algoritmus

A Wiener- és Kalman-szűrövei szemben az optimális lineáris becslő problémáját nem csak stochasztikus, hanem determisztikus koncepció alapján is meg lehet közelíteni. A regressziós feladatot már Gauss megfogalmazta; ez az *1. ábra* adaptív FIR szűrőjénél azt jelenti, hogy az n számú $d_1, d_2, ..., d_n$ értéket kívánjuk legkisebb négyzetes hibával előállítani n számú adott bemeneti érték lineáris kombinációjaként, melyek a FIR szűrőben mint $x_1, x_2, ..., x_n$ jelvektorok szerepelnek [6]:

$$e_k = d_k - y_k = d_k - h^T_n x_k, \quad k = 1, 2, ..., n \tag{5-1}$$

$$J(h_n) = \sum_{k=1}^{n} \lambda^{n-k} e^2_k \to \min \tag{5-2}$$

ahol λ exponenciális súlyozó tényező, $0 < \lambda \leq 1$. Az n hosszú ablakhoz tartozó h_n szűrővektor az (5-2) feltételnek megfelelően, a

$$\Phi_n h_n = \Theta_n \tag{5-3}$$

determinisztikus normálegyenlet megoldása, ahol

$$\Phi_n = \sum_{k=1}^{n} \lambda^{n-k} x_k x_k^t \tag{5-4}$$

a determinisztikus korrelációs mátrix és

$$\Theta_n = \sum_{k=1}^{n} \lambda^{n-k} x_k d_k \tag{5-5}$$

a determinisztikus keresztkorrelációs vektor a d és x bemenet között. Ez a két mennyiség könnyen számolható rekurzívan, ha (5-4)-ben és (5-5)-ben az összeg n-edik tagját leválasztjuk:

DR. VARGA IMRE

BME Villamosmérnöki Karán végzett 1982-ben. 1982–1984 között a BME Elméleti Villamosságtan Tanszéken volt tudományos továbbképzési ösztöndíjas, a hálózatelmélet témakörén belül lineáris hálózatok érzékenység- és toleranciaproblémáival foglalkozott. 1984-ben műszer- és irányítástechnikai szakmérnöki és egyetemi doktori oklevelet szerzett. 1984 óta a BME Híradástechnikai Elektronika Intézetben dolgozik. Jelenlegi fő érdeklődési területe a digitális jelfeldolgozás, adaptív szűrés.

$$\Phi_n = \lambda \Phi_{n-1} + x_n x_n^T \tag{5-6}$$
$$\Theta_n = \lambda \Theta_{n-1} + x_n d_n \tag{5-7}$$

Az optimális h_n szűrővektor rekurzív számításához (5-3) megoldásra van szükség, vagyis nem elegendő a rekurziót Φ_n-re felírni, hanem Φ_n^{-1}-re is szükséges. Ezt az

$$A = B^{-1} + CD^{-1}C^T \tag{5-8}$$
$$A^{-1} = B - BC(D + C^TBC)^{-1}C^TB$$

mátrixinverziós (Sherman-Morrison) lemma segítségével tehetjük meg, ahol

$$\begin{aligned} A &= \Phi_n \\ B^{-1} &= \lambda \Phi_{n-1} \\ C &= x_n \\ D &= 1. \end{aligned} \tag{5-9}$$

Így az adódik, hogy

$$\Phi^{-1}_n = \lambda^{-1} \Phi^{-1}_{n-1} - k^{-1} k_n x^T_n \Phi^{-1}_{n-1}, \tag{5-10}$$

ahol

$$k_n = \Phi^{-1}_n x_n = \frac{\lambda^{-1} \Phi^{-1}_{n-1} x_n}{1 + \lambda^{-1} x^T_n \Phi^{-1}_{n-1} x_n} \tag{5-11}$$

az ún. erősítésvektor.

Ha az (5-3) egyenletből kifejezett h_n-bc behelyettesítjük az (5-7) egyenletet, majd felhasználjuk az (5-10) és (5-11) rekurziót, akkor a szűrővektor rekurzív becslésére a

$$h_n = h_{n-1} + k_n \alpha_n$$
$$\alpha_n = d_n - h^T_{n-1} x_n \tag{5-12}$$

Beérkezett: 1988. X. 6. (*)

egyenlet adódik, α_n itt az a priori (adaptálás előtti, előző h_{n-1}-ből számolt) hibát jelenti. Az exponenciálisan súlyozott RLS algoritmust az (5–10), (5-11) (5–12) egyenletek összessége alkotja. Inicializálásként $h_o=0$ és $\Phi_o^{-1} = \delta^{-1}1$ szokás választani, ahol δ kis pozitív szám (10^{-3} ... 10^{-6}), I az egységmátrix.

Az RLS algoritmus lényeges tulajdonsága, hogy $\lambda=1$ esetén teljesül az ortogonalitás elve. Beszorozva ugyanis (5–1) mindkét oldalát x_k-val és összegezve k=1, ..., n-re

$$\sum_{k=1}^{n} x_k x^T_k h_n = \sum_{k=1}^{n} x_k d_k - \sum_{k=1}^{n} x_k c_k,$$

felhasználva továbbá az (5–3), (5–4) és (5–5) összefügéseket,

$$\sum_{k=1}^{n} x_k e_k = 0$$

adódik, ami azt jelenti, hogy az optimumhoz tartozó hibák ortogonálisak a bemeneti jelvektorokra. Beszorozva mindkét oldalt h^T_n-vél,

$$\sum_{k=1}^{n} h^T_n x_k e_k = \sum_{k=1}^{n} y_k e_k = 0$$

adódik, tehát a szűrő kimenete, vagyis a d_k mennyiségek becsült értékei ortogonálisak az e_k hibákra. Azért nevezik az (5–3) egyenletet normálegyenletnek, mert a négyzetes hibaösszeg akkor a legkisebb, ha a hiba merőleges a kimeneti becslőre.

Az RLS algoritmus n adott számhoz – származzék az determinisztikus függvényből vagy stochasztikus folyamatból – legkisebb négyzetes hiba értelemben rendel egy optimális h_n vektort. Digitális jelfeldogozás esetén a mintavételezés révén egy időben soros bemeneti adatfolyam áll rendelkezésre s ezért egy rekurzív algoritmus előnyös, mely ha n adatra ismert az optimális h_n becslő, (n+1) adatra ennek felhasználásával határozza meg h_{n+1}-et.

Az RLS algoritmus főbb tulajdonságai az alábbiak:

1. Annak ellenére, hogy az RLS-probléma megfogalmazásában nem szerepel várható érték, felvethető a kérdés, a véges számú adatra optimalizáló algoritmus milyen becslőt állít elő, ha $n\to\infty$. Belátható, hogy $\lambda=1$ esetén (végtelen memória) az RLS algoritmus a Wiener-optimumhoz tart. Nincs maradékhiba s ennek az az oka, hogy az RLS algoritmus nem alkalmaz közelítést.

2. Az algoritmus $\lambda=1$ esetén a hiba sok szűrőre átlagolt négyzetes átlagában tipikusan egy nagyságrenddel gyorsabban konvergál, mint az LMS algoritmus (*l. a 2. ábrát*).

3. A konvergencia sebessége nem függ a korrelációs mátrix kondicionáltságától, a legnagyobb és legkisebb sajátérték arányától, mert maga a mátrix szerepel a rekurzióban.

4. Az algoritmus $\lambda=1$ esetén minden addigi információt felhasználva állítja elő a becslőt, végtelen memóriájú: ez a választás stacionárius jeleknél célszerű, $\lambda<1$ esetén véges memóriájú, exponenciálisan felejtő: ez a választás nemstacionárius esetben célszerű.

5. Az LMS algoritmusénál lényegesen kedvezőbb konvergenciatulajdonságok ára a lényegesen nagyobb, $O(N^2)$ számításigény a mátrix-vektor szorzások miatt.

6. Numerikus stabilitás. Az LMS algoritmus tárgyalásánál vázolt problémákon kívül itt az is előfordulhat, hogy az akkumulálódó hibák hatására a korrelációs mátrix szingulárissá válik, az algoritmus leáll, vagy erősen rosszul kondicionálttá válik, ami lerontja a pontosságot [3].

7. Az inicializálási fázisban a $\delta 1$ mátrix torzítja a h_n becslőt, belátható viszont, hogy $n\to\infty$ esetén a torzítás nullához tart. Ezek szerint az együttható hiba átlagban konvergens.

Miután az RLS és a Kalman-algoritmust leíró bizonyos egyenletek azonosak, felvethető a kérdés, mi a kapcsolat közöttük. Az RLS feladat megfogalmazása determinisztikus: véges adatra optimalizál; a Kalman szűrőé stochasztikus: már kiinduláskor stochasztikus jeleket kezel. A Kalman-algoritmus speciális esete – FIR szűrő statikus közegben – megegyezik az RLS algoritmussal, ha $\lambda=1$, így a (4–25)–(4–28) egyenleteknek az (5–10)–(5–12) egyenletek felelnek meg. A nemstacionárius esetben Q_v leírt közelítésével kapott Kalman-algoritmus pedig az exponenciálisan felejtő RLS algoritmussal azonos ($\lambda=(1+q)^{-1}$), a (4–35)–(4–38) egyenleteknek az (5–10)–(5–12) egyenletek felelnek meg.

6. Gyors és O(N) számításigényű algoritmusok

Az adaptív FIR szűrők második generációját azok az algoritmusok jelentik, melyek a Kalman ill. RLS algoritmusokkal matematikailag azonosak és így azoknak megfelelő kedvező konvergenciatulajdonságokkal, pontos beállással és az LMS algoritmuséhoz közelálló számításigénnyel rendelkeznek. Az első ilyen algoritmus, az ún. fast Kalman vagy fast RLS (FRLS) algoritmus [6] a bemeneti soros adatfolyam időbeli „eltolhatóságát” kihasználva, a Kalman-erősítést nem a mátrix-vektor szorzásokon keresztül adaptálja, hanem közvetlenül. Ennek a módszernek a fő hátránya a numerikus instabilitás véges szóhosszúságú megvalósítás esetén, melyet csak dupla pontosság alkalmazásával lehetett kiküszöbölni. Ez azonban gyakorlati feladatoknál nem járható út. Számításigénye I0N (szorzások, osztások száma).

A második algoritmus az FAEST (Fast Aposteriori Error Sequential Technique) nevet kapta [2], mely a számításigényt tovább csökkenti 7N-re. Ezt részletesebben ismertetjük.

Az RLS algoritmus levezetésénél láttuk, hogy a szűrő együtthatóvektorának adaptációját leíró egyenlet korrekciós tagja a Kalman-erősítés és az a priori hiba szorzata. Hasonlóképpen értelmezhető egy olyan w erősítésvektor, mely az a posteriori hibával

$$\epsilon_N(n) = d(n) - h^T{}_N(n)\,x_N(n) \qquad (6-1)$$

szorozva adja az adaptációs rekurziót:

$$h_N(n+1) = h_N(n) + w_N(n+1)\,\epsilon_N(n+1). \qquad (6-2)$$

(A vektorok indexe a szűrő hossza, zárójelben az időpont áll.) Az RLS algoritmushoz hasonlóan az erősítésvektor kielégíti a

$$\Phi_N(n)\,w_N(n+1) = x_N(n+1) \qquad (6-3)$$

determinisztikus normálegyenletet, ahol

$$\Phi_N(n) = \sum_{i=N}^{n} x_N(n)\,x_N{}^T(n) \qquad (6-4)$$

a determinisztikus korrelációs mátrix. Az FAEST algoritmus a bemeneti adatfolyam eltolhatóságát használja ki és a

$$\Phi_N(n-1)\,w_N(n) = x_N(n)$$

$$\Phi_N(n)\,w_N(n+1) = x_N(n+1) \qquad (6-5)$$

$$\Phi_{N+1}(n)\,w_{N+1}(n+1) = x_{N+1}(n+1)$$

egyenletekből $w_N(n)$ ismeretében $w_N(n+1)$-et határozza meg rekurzíve. Miután

$$\Phi_{N+1}(n) = \begin{bmatrix} r^f{}_{ON}(n) & r^{fT}{}_N(n-1) \\ r^f{}_N(n) & R_N(n-1) \end{bmatrix} = \begin{bmatrix} R_N(n-1) & r^b{}_N(n) \\ r^{bT}{}_N(n) & r_{on}{}^b(n) \end{bmatrix} \qquad (6-6)$$

alakban particionálható, inverzét a particionált mátrixok invertálási szabálya szerint ki lehet fejezni. Ezt behelyettesítve a

$$w_{N+1}(n+1) = \Phi_{N+1}{}^{-1}(n+1)\,x_{N+1}(n+1) \qquad (6-7)$$

egyenletbe, rekurziót kapunk az együtthatóvektorra:

$$w_{N+1}(n+1) = \begin{bmatrix} w_N(n+1) \\ 0 \end{bmatrix} - \frac{e^b{}_N(n+1)}{\alpha^b{}_N(n)} \begin{bmatrix} b_N(n) \\ 1 \end{bmatrix} \qquad (6-8)$$

illetve

$$w_{N+1}(n+1) = \begin{bmatrix} 0 \\ w_N(n) \end{bmatrix} - \frac{e_N{}^f(n+)}{\alpha^f{}_N(n)} \begin{bmatrix} 1 \\ a_N(n) \end{bmatrix} \qquad (6-9)$$

ahol

$$\alpha_N{}^f(n) = r_{ON}{}^f(n) + a_N{}^T(n)\,r_N{}^f(n)\,,\ r_{ON}{}^f(n) = \sum_{i=N}^{n} x^2(i) \qquad (6-10)$$

$$\alpha_N{}^b(n) = r_{ON}{}^b(n) + b_N{}^T(n)\,r_N{}^b(n)\,,\ r_{ON}{}^b(n) = \sum_{i=N}^{n} x^2(i-N)$$

$$e^f{}_N(n+1) = x(n+1) - a_N{}^T(n)x_N(n) \qquad (6-11)$$

$$e_N{}^b(n+1) = x(n+1-N) - b_N{}^T(n)\,x_N(n+1).$$

Az a és b vektor interpretálható, mint egy forward ül. egy backward prediktor együtthatóvektorai, melyek normálegyenlete

$$\Phi_N(n-1)\,a_N(n) = r_N{}^f(n)\,,\ r_N{}^f(n) = \sum_{i=N}^{n} x_N(i-1)\,x(i) \qquad (6-12)$$

$$\Phi_N(n)\,b_N(n) = r_N{}^b(n)\,,\ r_N{}^b(n) = \sum_{i=N}^{n} x_N(i)\,x(i-N)$$

Így tehát $w_N(n)$ ismeretében először $w_{N+1}(n+1)$-et tudjuk kiszámítani, majd ebből $w_N(n+1)$-et. A h együtthatók adaptálásához még az $\epsilon_N(n+1)$ a posteriori hibára kell rekurziót találnunk, továbbá ez az n-edik időpontban ismeretlen. Először az

$$e_N(n+1) = d(n+1) - h_N{}^T(n)\,x_N(n+1) \qquad (6-13)$$

a priori és az $\epsilon_N(n+1)$ a posteriori hiba kapcsolatát írjuk fel. Az adaptációs egyenletek a w^* Kalman-erősítéssel és az e a priori hibával

$$\Phi_N(n)\,w_N{}^*(n) = x_N(n) \qquad (6-14)$$

$$h_N(n+1) = h_N(n) + w_N{}^*(n+1)\,e_N(n+1) \qquad (6-15)$$

alakúak. Írjuk fel a mátrixinverziós lemmával

$$\Phi_N{}^{-1}(n+1) = \Phi_N{}^{-1}(n) - w_N(n+1)\,w_N{}^T(n+1) / \alpha_N(n+1) \qquad (6-16)$$

$$\alpha_N(n+1) = 1 - x_N{}^T(u+1)\,w_N(n+1), \quad (6$$

majd figyelembe véve (6-3)-at és (6

$$w_N{}^*(n+1) = \Phi_N{}^{-1}(n+1)\,x_N(n$$

adódik. (6-1) és (6-

$$\ldots - e_N(n+1) / \alpha_N(o+1). \qquad (6-19)$$

Az a posteriori hiba rekurzióját most α_N rekurziójára vezettük vissza. Ezt azonban könnyen meg tudjuk adni, hiszen (6-9), (6-11) és (6-17) felhasználásával

$$\alpha_{N+1}(n+1) = \alpha_N(n) + \frac{e_N{}^f(n+1)}{\alpha_N{}^f(n)}\,e_N{}^f(n+1) \qquad (6-20)$$

adódik, de (6-8) és (6-11) miatt

$$\alpha_N(n+1) = \alpha_{N+1}(n+1) + \frac{e_N^b(n+1)}{\alpha_N^b(n)} e_N^b(n+1) \qquad (6-21)$$

Ez azt jelenti, hogy $\alpha_N(n)$ ismeretében először $\alpha_{N+1}(n+1)$ számolható (6–20); majd $\alpha_N(n+1)$ (6–21). A forward és backward prediktor rekurziói hasonló gondolatmenettel adódnak, a (6–11) egyenleteken kívül

$$a_N(n+1) = a_N(n) + w_N(n)\, e_N^f(n+1) \,/\, \alpha_N(n)$$

$$\alpha_N^f(n+1) = \alpha_N^f(n) + (e_N^f(n+1))^2 \,/\, \alpha_N(n) \qquad (6-22)$$

$$b_N(n+1) = b_N(n) + w_N(n+1)\, e_N^b(n+1) \,/\, \alpha_N(n+1)$$

$$\alpha_N^b(n+1) = a_N^b(n) + (e_N^b(n+1))^2 \,/\, \alpha_N(n+1)$$

Az FAEST algoritmust a (6–2), (6–8), (6–9), (6–11), (6–13), (6–19), (6–20), (6–21) és (6–22) egyenletek összessége alkotja. Az algoritmussal kapcsolatban az a tapasztalat, hogy hosszú idejű numerikus stabilitása sokszor nem kedvező, divergenssé válik a $t=N$ időpontban illetve több ezer vagy tízezer iteráció után [2], [3]. Ezen periodikus újrainicializálással lehet segíteni (l. alább).

A gyors RLS algoritmusok harmadik képviselője az FTF (Fast Transversal Filters) algoritmus [4]. Ennek legegyszerűbb, normalizálatlan változata lényegében azonos az FAEST-vel, számításigényük is azonos. A kvázinormalizált ill,. normalizált változatok a stabilitás növelését célozzák, de számításigényük is nagyobb. Az FTF algoritmusok a legkisebb négyzetes hibájú becslő előállításához – az RLS algoritmushoz hasonlóan – nem alkalmaznak közelítést. Az FTF algoritmus legfeljebb négy FIR szűrőt tartalmaz, melyek bemenete közösen az x jelet fogadja. Az első szűrő forward prediktor, a második backward prediktor, a harmadik a Kalman-erősítést számítja, végül a negyedik szűrő a „tényleges" szűrő, mely a kívánt kimenetet állítja elő. Mind a négy szűrő két üzemmódban működik: szűrés és adaptáció.

Az inicializálási periódusra, mely az $1 \le n \le N+1$ intervallumot foglalja magában (N a szűrő hossza), külön algoritmus vonatkozik. A tapasztalat azt mutatja, hogy az FTF algoritmusok hosszú idejű numerikus stabilitása nem kielégítő az esetek nagy hányadában [3]. A problémát jelentősen enyhíti a periodikus újrainicializálás technikája [4]. A periodikusan végrehajtott inicializálási szakaszok alatt az adaptációról egy LMS algoritmus gondoskodik, mely tapasztalat szerint nem rontja el az adaptációt az inicializálási szakasz rövidsége miatt.

A gyors RLS algoritmusok hosszúidejű numerikus problémái további tökéletesítés szükségességét bizonyítják, hogy mire a jelfeldolgozó eszközök sebessége eléri azt a szintet, mely ezen bonyolultabb algoritmusok implementálásához szükséges, már e szempontból is kielégítő adaptív FIR szűrő algoritmus álljon rendelkezésre. Ez ma még nyitott kérdés; pl. az FTF algoritmus stabilitását redundanciával, 9N számításigénnyel jelentősen javítani lehet.

7. Adaptív IIR szűrők

Az adaptív FIR szűrés problémája – mint az előző pontokban leírtakból kitűnik – matematikailag jól kezelhető és áttekinthető algoritmusokra vezet. Ha azonban sok együtthatóra van szükség, akkor a számításigény már igen jelentőssé válik. Ez tipikusan az adaptív szűrő rendszer-identifikáció és inverz modellezés alkalmazásában fordul elő, amikor az identifikálandó rendszer végtelen ill. véges impulzusválaszú. Ilyenkor az adaptív IIR szűrők lényegesen előnyösebbek lehetnek, mert sokkal kisebb fokszámmal és ezáltal számításigénnyel pontosabb modellezést tesznek lehetővé, főként nagy pólusjóságú esetben.

Az IIR szűrők rekurzív jellege ugyanakkor néhány olyan nehézséget vet fel, melyek a FIR szűrőknél nem merültek fel. Ezek az alábbiak. Az IIR szűrők a FIR struktúrával ellentétben nem strukturálisan stabilak, vagyis nem stabilak az együtthatók bármely értéke mellett, ezért vagy gondoskodni kell a stabilitás folyamatos (literációkénti) ellenőrzéséről, vagy az algoritmusnak biztosítania kell a stabilitást. A második probléma abban áll, hogy az IIR szűrő számláló és nevező együtthatóira felírt átlagos négyzetes hibafelület multimodiális lehet, ezért gondoskodni kell arról, hogy az algoritmus a globális optimumhoz konvergáljon. További problémát jelent az a tény, hogy a gradiens kifejezése lényegesen bonyolultabb, továbbá jelentősen megnehezíti a konvergenciaanalízist az, hogy a nagy jóságú pólusok nagy időállandókat jelentenek.

Az adaptív IIR szűrők egyik csoportja a gradiensmódszeren alapszik [5], [9]. Az IIR szűrőt leíró differenciaegyenlet

$$y_n = \sum_{i=1}^{N} a_i\, y_{n-1} + \sum_{i=0}^{M} b_i\, x_{n-1}\,, \qquad (7-1)$$

így az átlagos négyzetes hibát

$$J_n = \frac{1}{2} E\,\{e_n^2\} = \frac{1}{2} E\,\{\,(d_n - y_n)^2\} \qquad (7-2)$$

minimalizáló együtthatókra a gradiensmódszer

$$\hat{a}_{n+1} = \hat{a}_n - \mu_n \nabla_a J_n$$
$$\hat{b}_{n+1} = \hat{b}_n - \mu_n \nabla_b J_n \qquad (7-3)$$

alakú, ahol

$$\nabla_c J_n = -E\,\{e_n \nabla_c y_n\}\,, \qquad (7-4)$$

a ∇_c a c szerinti gradienst jelöli (c itt a vagy b). Ha az a és b együtthatókat összefogjuk egy

$$h = [a\ b]^T \qquad (7-5)$$

vektorba, akkor (7−1) szerint rekurzíve megadható

$$\nabla_h y_n = x_n + \sum_{i=1}^{N} a_i \nabla_h y_{n-1} \qquad (7-6)$$

ahol

$$x_n = [\, y_{n-1} \ldots y_{n-N}\, x_n \ldots x_{n-M}]^T. \qquad (7-7)$$

A gradienst az LMS algoritmushoz hasonlóan közelíthetjük, ha a jelek statisztikái a priori ismeretlenek:

$$\nabla_h J_n \approx -\, e_n \nabla_h y_a \qquad (7-8)$$

így tehát

$$h_{n+1} = h_n + \mu_n e_n \nabla_h y_n \qquad (7-9)$$

$$y_n = h_n^T x_n \qquad (7-10)$$

és mivel (7−6)-ban a helyett csak a_n ismert, újabb közelítés szükséges:

$$\nabla_h y_n = x_n + \sum_{i=1}^{N} a_{ni} y_{n-1} \qquad (7-11)$$

A gradiens típusú algoritmust IIR szűrőkre a (7−9) − (7−11) egyenletek adják meg. A két közelítés ellensúlyozásaképp μ_n-et igen kicsire kell választani, hogy h_n elég lassan változzon és így elég közel kerülhessen a megoldáshoz. Ez a módszer nem garantálja a globális optimumhoz történő konvergenciát. Vizsgálatok szerint az unimodalitást úgy lehet valószínűsíteni, hogy a szűrő „elégséges fokszámú", vagyis fokszáma nagyobb vagy egyenlő a modellezendő rendszer (ismeretlen) fokszámánál [5], [9]. Mivel azonban a túlparaméterezés is rejt veszélyeket magában, összességében azt állíthatjuk, hogy a fokszám megválasztása igen kritikus.

Az algoritmus $O(N^2)$ bonyolultságát a gradiens további közelítésével lehet O(N)-re csökkenteni [5]:

$$\nabla_h y_n \approx x_n \qquad (7-12)$$

Az ún. Feintuch-algoritmus a (7−9), (7−10), (7−12) egyenletek összessége, mely teljesen hasonló az LMS algoritmushoz. Ez a módszer további egyszerű, megbízható algoritmusok kidolgozására inspirálta a kutatókat.

Az adaptív IIR szűrők másik csoportja a nemlineáris stabilitáselméletben gyökerezik. A HARF és SHARF (Simple Hyperstable Recursive Filter) algoritmusoknak [5], [9] a Feintuch-algoritmus speciális esete. Adaptív IIR szűrő kidolgozására más út is kínálkozik, mint pl. az állapotváltozós módszer, a racionális törtfüggvény együtthatóit megelőzően az impulzusválasz együtthatóinak adaptálására stb. Kísérletek irányultak a direkt struktúrától eltérő (párhuzamos, kaszkád, rács) struktúra esetleges előnyeinek kihasználására. Ezek a vizsgálatok még nem tekinthetők lezártnak s remélhető, hogy minden szempontból megfelelő IIR algoritmushoz vezetnek.

Összegezés

Az adaptív szűrők a jelfeldolgozás, a szabályozástechnika egyre fontosabb eszközeivé válnak. Ezt alapvetően a jelfeldolgozás technológiájának döntő fejlődése tette lehetővé.

A stochasztikus approximáció és az LMS-algoritmus legegyszerűbb módszereit sikeresen alkalmazták különböző gyakorlati jelfeldolgozási feladatok megoldására. A gyors RLS és Kalman algoritmusok − melyek lényegesen kedvezőbb konvergenciatulajdonságokkal rendelkeznek szintén O(N) számításigény mellett − a második generációt képviselik, implementálásukhoz a mai élenjáró technológiára van szükség (lebegőpontos, nagysebességű VLSI jelfeldolgozó processzor). Az adaptív IIR szűrők speciális feladatokra jól kidolgozottak, de a stabilitást és a globális optimumhoz történő konvergenciát általános esetben biztosító algoritmus még kutatás tárgya. Érdekes perspektívát nyújthatnak a nemlineáris adaptív szűrők, az eddig kidolgozottak számításigénye túl nagy.

IRODALOM

[1] *Caraiscos, C. − Liu, B.:* A roundoff error analysis of the LMS adaptive algorithm. IEEE Trans. ASSP, vol. ASSP−32, No. 1, Feb. 1984, pp. 34−41.

[2] *Carayannis, G. − Manolakis, D. G. − Kalouptsidis, N.:* A fast sequential algorithm for adaptive filtering. IEEE Trans. ASSP, vol. ASSP−31, No. 6, Dec. 1983, pp. 1394−1402.

[3] *Cioffi, J. M.:* Limited-precision effects in adaptive filtering. IEEE Trans. CAS, vol. CAS−34, No. 7, July 1987, pp. 821−833.

[4] *Cioffi, J. M. − Kailath, T.:* Fast, recursive-least-squares transversal filters for adaptive filtering. IEEE Trans. ASSP, vol. ASSP−32, No. 2, Apr. 1984, pp. 304−337.

[5] *Cowan, C. F. N. − Grant, P. M.:* Adaptive filters. Prentice−Hall, Engl.Cliffs, N.J., 1985.

[6] *Hayldn, S.:* Adaptive filter theory. Prentice-Hall, Engl.Cliffs,N.J.,1986.

[7] *Narayan, S. S. − Peterson, A. M. − Narashimba, M. J.:* Transform domain LMS algorithm. IEEE Trans. ASSP, vol. ASSP−31, No. 3, June 1983, pp. 609−614.

[8] *Robbins, H. − Monro, S.:* A stochastic approximation method. Ann. Math. Statis., vol. 22., 1951, pp. 400−407.

[9] *Treichler, J. R. − Johnson, C. R. − Larimore, M. G.:* Theory and design of adaptive filters. J. Wiley, N. Y., 1987.

[10] *Widrow,B. − Stearns,S.:* Adaptive signal processing. Prentice−Hall, Eng.Cliffs,N.J.,1985.