

Πανεπιστήμιο Πατρών

Τμήμα Ηλεκτρολόγων Μηχανικών και Τεχνολογίας Υπολογιστών

Τομέας Συστημάτων και Αυτομάτου Ελέγχου

ΠΡΟΣΑΡΜΟΣΤΙΚΟΣ ΕΛΕΓΧΟΣ

Διάλεξη 5

Πάτρα 2008

Χρονικά μεταβαλλόμενες παράμετροι

Στο πρόβλημα των ελαχίστων τετραγώνων οι παράμετροι θ_i θεωρούνται πως είναι σταθερές. Παρά ταύτα σε πολλά προβλήματα προσαρμοστικού ελέγχου, έχουμε παραμέτρους που μεταβάλλονται αργά με τον χρόνο. Σε αυτήν την περίπτωση αντικαθιστούμε το κριτήριο των ελαχίστων τετραγώνων με το:

$$V(\theta, t) = \frac{1}{2} \sum_{i=1}^t \lambda^{t-i} (y(i) - \phi^T(i)\theta)^2, \quad 0 \leq \lambda \leq 1$$

Η παράμετρος λ καλείται συντελεστής απώλειας μνήμης του συστήματος (forgetting factor).

Θεωρούμε ότι ο πίνακας $\Phi(t)$ είναι πλήρης τάξης για $t \geq t_0$. Η παράμετρος θ που ελαχιστοποιεί το προηγούμενο κριτήριο δίδεται αναδρομικά:

$$\hat{\theta}(t) = \hat{\theta}(t-1) + K(t)(y(t) - \phi^T(t)\hat{\theta}(t-1))$$

$$K(t) = P(t-1)\phi(t)\left(\lambda I + \phi^T(t)P(t-1)\phi(t)\right)^{-1}$$

$$P(t) = \left(I - K(t)\phi^T(t)\right)P(t-1) / \lambda$$

Exponentially Weighted RLS (Recursive Least Squares)

$$\bar{V}(n) = e^T(n)\Lambda(n)e(n)$$

$$\Lambda(n) = \text{diag} \{ \lambda^n, \lambda^{n-1}, \dots, \lambda, 1 \}$$

Premultiplying “Prewindowed” με $\Lambda^{1/2}(n)$ και λύνοντας για $\hat{H}(n)$ παρνούμε:

$$\hat{H}(n) = \left[U^T(n)\Lambda(n)U(n) \right]^{-1} U^T(n)\Lambda(n)d(n)$$

Ορίζουμε: $P(n) = \left[U^T(n)\Lambda(n)U(n) \right]^{-1}$

Έχουμε:

$$P^{-1}(n+1) = \lambda U^T(n)\Lambda(n)U(n) + \phi(n+1)\phi^T(n+1)$$

$$= \lambda P^{-1}(n) + \phi(n+1)\phi^T(n+1)$$

$$P(n+1) = \frac{P(n) - P(n)\phi(n+1)\left[\lambda + \phi^T(n+1)P(n)\phi(n+1)\right]^{-1}\phi^T(n+1)P(n)}{\lambda}$$

$$\hat{H}(n+1) = P(n+1)U^T(n+1)\Lambda(n+1)d(n+1)$$

$$= P(n+1)\left[\lambda U^T(n)\Lambda(n)d(n) + \phi(n+1)d(n+1)\right]$$

$$K(n+1) = P(n+1)\phi(n+1) = P(n)\phi(n+1)\left[\lambda + \phi^T(n+1)P(n)\phi(n+1)\right]^{-1}$$

Αρχικοποίηση:

$$\hat{H}(0) = 0$$

$$P(0) = \sigma I$$

$$\phi(0) = 0$$

Σε κάθε βήμα ισχύει:

$$e(n) = d(n) - \phi^T(n)\hat{H}(n-1), \quad (= e_{\hat{H}(n-1)}(n))$$

$$P(n) = \frac{P(n-1) - P(n-1)\phi(n)\left[\lambda + \phi^T(n)P(n-1)\phi(n)\right]^{-1}\phi^T(n)P(n-1)}{\lambda}$$

$$K(n) = P(n-1)\phi(n)\left[\lambda + \phi^T(n)P(n-1)\phi(n)\right]^{-1} \quad (\text{Ανανέωση κέρδους})$$

Επίσης σε κάθε βήμα ισχύει:

$$\hat{H}(n) = \hat{H}(n-1) + K(n)e(n)$$

(Ανανέωση των συντελεστών φίλτρου)

$$\phi(n+1) = Z\phi(n) + [y(n+1), 0, \dots, 0]^T$$

(Ανανέωση των καταστάσεων φίλτρου)

όπου $Z = \begin{bmatrix} 0 & & & & \\ 1 & 0 & & & \\ & \ddots & \ddots & & \\ & & & 1 & 0 \end{bmatrix}$

Απλοποιημένοι αλγόριθμοι

Ο RLS αλγόριθμος έχει όπως είδαμε δύο μεταβλητές καταστάσεως, $\hat{\theta}$ και P , που πρέπει να ενημερώνονται σε κάθε βήμα. Η ενημέρωση του διανύσματος $\hat{\theta}$ απαιτεί $2n$ προσθέσεις και $2n$ πολλαπλασιασμούς ανά βήμα. Αντίθετα η ενημέρωση του πίνακα P απαιτεί $1.5n(n+1)$ προσθέσεις, $1.5n(n+1)$ πολλαπλασιασμούς και $0.5n(n+1)$ διαιρέσεις ανά βήμα.

Υπάρχουν αρκετοί απλοποιημένοι αλγόριθμοι που αποφεύγουν την ενημέρωση του πίνακα P , με κόστος βέβαια βραδύτερη σύγκλιση.

Ένας τέτοιος αλγόριθμος είναι και ο αλγόριθμος προβολής Kaczmarz που ακολουθεί.

Έχουμε:

$$y(t) = \phi^T(t)\theta$$

Η μέτρηση $y(t)$ περιέχει πληροφορία μόνο στην κατεύθυνση $\phi(t)$ στον χώρο των παραμέτρων, επομένως:

$$\hat{\theta}(t) = \hat{\theta}(t-1) + a\phi(t)$$

Όπου η παράμετρος a επιλέγεται έτσι ώστε:

$$y(t) = \phi^T(t)\hat{\theta}(t) = \phi^T(t)\hat{\theta}(t-1) + a\phi^T(t)\phi(t)$$

Οπότε:

$$a = \frac{1}{\phi^T(t)\phi(t)} \left(y(t) - \phi^T(t)\hat{\theta}(t-1) \right)$$

Και προκύπτει ο αλγόριθμος Kaczmarz:

$$\hat{\theta}(t) = \hat{\theta}(t-1) + \frac{\phi(t)}{\phi^T(t)\phi(t)} \left(y(t) - \phi^T(t)\hat{\theta}(t-1) \right)$$

Υποθέτοντας ότι τα δεδομένα έχουν παραχθεί από την εξίσωση:

$$y(t) = \varphi^T(t)\theta$$

με παράμετρο $\theta = \theta^0$. Τότε προκύπτει από τον προηγούμενο αλγόριθμο ότι το παραμετρικό σφάλμα:

$$\tilde{\theta} = \theta^0 - \hat{\theta}$$

ικανοποιεί την σχέση:

$$\tilde{\theta}(t) = A(t)\tilde{\theta}(t-1)$$

όπου

$$A(t) = I - \frac{\phi(t)\phi^T(t)}{\phi^T(t)\phi(t)}$$

(A(t) : Πίνακας Προβολής)

Για την αποφυγή του προβλήματος που δημιουργείται όταν $\phi(t) = 0$, ο αλγόριθμος αυτός στην πράξη τροποποιείται ως εξής:

$$\hat{\theta}(t) = \hat{\theta}(t-1) + \frac{\gamma\phi(t)}{a + \phi^T(t)\phi(t)} \left(y(t) - \phi^T(t)\hat{\theta}(t-1) \right)$$

όπου $a \geq 0$, $0 < \gamma < 2$

Adaptive FIR Filtering

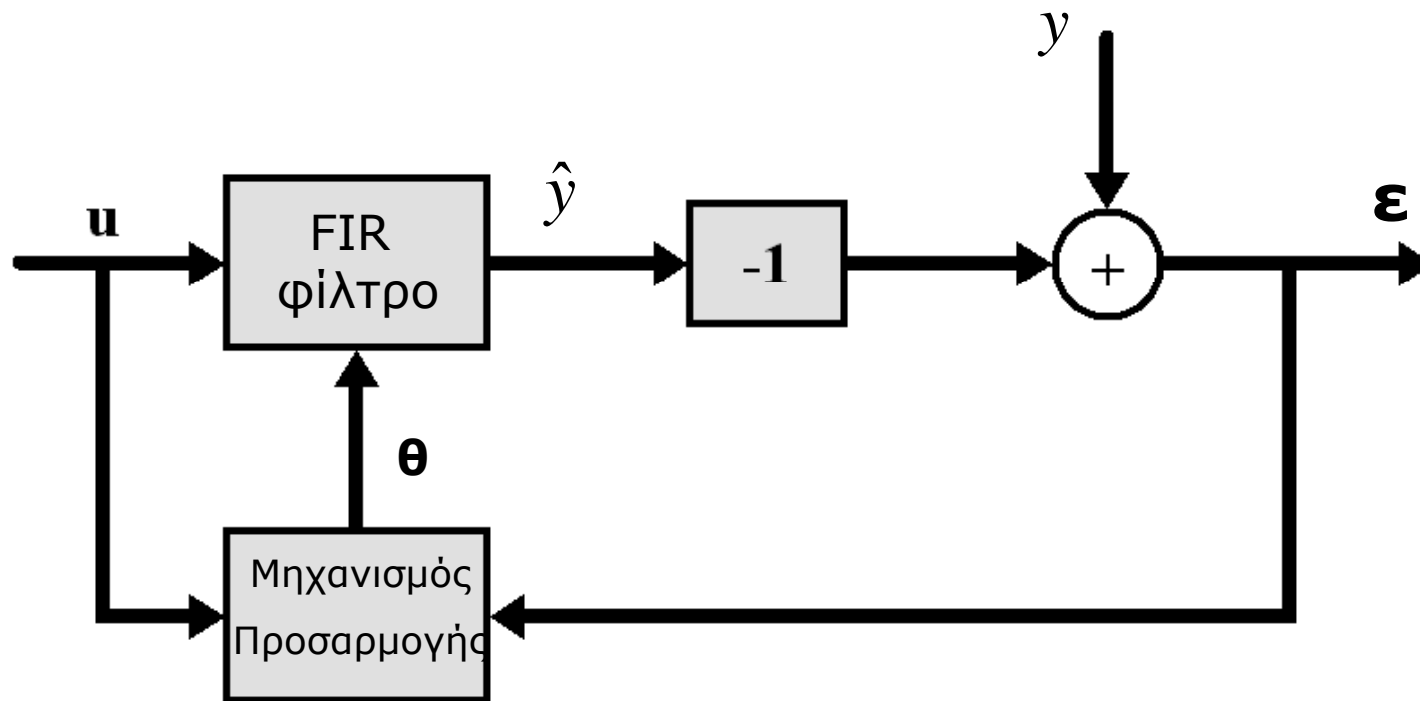
Θεωρούμε ένα μοντέλο (Σ) που περιγράφεται από την σχέση:

$$y(n) = +h_1 u(n-1) + \dots + h_N y(n-N)$$

με $H^T = [h_1 \quad \dots \quad h_N]$

Επομένως:

$$y(n) = H^T \begin{bmatrix} u(n-1) \\ \vdots \\ u(n-N) \end{bmatrix}$$



Block διάγραμμα αναδρομικού παραμετρικού εκτιμητή για FIR μοντέλο

Έχουμε :

$$V(n) = \sum_{s=0}^n e^2(s)$$

$$e(n) = d(n) - y(n)$$

$$V(n) = e^T(n)e(n)$$

$$\hat{H}(n) = [U^T(n)U(n)]^{-1} U^T(n)d(n)$$

Ισχύει ότι:

$$P(n) = [U^T(n)U(n)]^{-1}$$

$$P(n+1) = \begin{bmatrix} U^T(n) & \Phi(n+1) \end{bmatrix} \begin{bmatrix} U(n) \\ \Phi^T(n+1) \end{bmatrix}^{-1} = [U^T(n)U(n) + \Phi(n+1)\Phi^T(n+1)]^{-1}$$

$$\Phi^T(n+1) = [u(n) \quad \dots \quad u(n-N+1)]$$

$$P(n+1) = P(n) - P(n)\Phi(n+1)[1 + \Phi^T(n+1)P(n)\Phi(n+1)]^{-1}\Phi^T(n+1)P(n) \quad (1)$$

(λόγω της ιδιότητας: $[A + BCD]^{-1} = A^{-1} - A^{-1}B(C + DA^{-1}B)^{-1}DA^{-1}$)

$$\hat{H}(n+1) = P(n+1)U^T(n+1)d(n+1) = P(n+1)[U^T(n)d(n) + \Phi(n+1)d(n+1)] \quad (2)$$

Επομένως από τις σχέσεις (1), (2) προκύπτει:

$$\hat{H}(n+1) = \hat{H}(n) - P(n)\Phi(n+1)\left[1 + \Phi^T(n+1)P(n)\Phi(n+1)\right]^{-1} \Phi^T(n+1)\hat{H}(n) \\ + P(n+1)\Phi(n+1)d(n+1)$$

Ορίζουμε:

$$K(n+1) = P(n+1)\Phi(n+1) = \\ = P(n)\Phi(n+1)\left[1 + \Phi^T(n+1)P(n)\Phi(n+1)\right]^{-1} \quad (3)$$

$$\hat{H}(n+1) = \hat{H}(n) + K(n+1)\left[d(n+1) - \Phi^T(n+1)\hat{H}(n)\right] \quad (4)$$

Από εξισώσεις (1),(3) και (4) RLS

$$P(0) = \begin{bmatrix} U^T(0) & U(0) \end{bmatrix} \text{ ή } P(0) = \sigma I, \sigma\text{-large, } \hat{H}(0) = -P(0)U^T(0)d(0) \text{ ή } \hat{H}(0) = 0$$

Sliding window μορφή του RLS

Εκτίμηση ως συνάρτηση των δεδομένων σε ένα τετράγωνο παράθυρο μήκους L .

$$\{u(n-L+1), K, u(n)\}$$

- Πρόσθεση νέου σημείου $[y(n)]$

$$P^1(n) = P(n-1) - P(n-1)\Phi(n) \left[1 + \Phi^T(n)P(n-1)\Phi(n) \right]^{-1} \Phi^T(n)P(n-1)$$

$$\hat{H}^1(n) = \hat{H}(n-1) + K(n) \left[y(n) - \Phi^T(n)P(n-1) \right]$$

$$K(n) = P(n-1)\Phi(n) \left[1 + \Phi^T(n)P(n-1)\Phi(n) \right]^{-1}$$

- Απόρριψη σημείου δεδομένων $[y(n-L)]$

$$P(n) = P^1(n) + P^1(n)\Phi(n-L)\left[1 + \Phi^T(n-L)P^1(n)\Phi(n-L)\right]^{-1} \Phi^T(n-L)P^1(n)$$

$$\hat{H}(n) = \hat{H}^1(n) - K^1(n)\left[y(n-L) - \Phi^T(n-L)\hat{H}^1(n)\right]$$

$$K^1(n) = P^1(n)\Phi(n-L)\left[1 + \Phi^T(n-L)P^1(n)\Phi(n-L)\right]^{-1}$$

Πλεονεκτήματα:

- Χρησιμοποιείται για παρακολούθηση απότομων αλλαγών στην δυναμική του συστήματος.

Μειονεκτήματα:

- (L+N) σημεία δεδομένων πρέπει να αποθηκευτούν
- Η υπολογιστική πολυπλοκότητα διπλασιάζεται