

Εργασία Μέρος Β - Θεωρία Εκτίμησης και Ανίχνευσης

Ομάδα 29

Αριστείδης Δασκαλόπουλος

arisdask@ece.auth.gr

AEM: 10640

Γεώργιος Ρουσομάνης

rousman@ece.auth.gr

AEM: 10703

Ιούνιος 2025

Εισαγωγή

Στόχος: Αποθορυβοποίηση πολυκαναλικών HEF σημάτων, $\mathbf{y}[k] \in \mathbb{R}^N$, που περιέχουν παρεμβολές λόγω ανοιγοκλεισίματος ματιών (blinks).

→ Το σήμα $y[k]$ μοντελοποιείται ως:

$$\mathbf{y}[k] = \mathbf{v}[k] + \mathbf{d}[k], \quad k \in \{1, \dots, K\} \text{ με } K \text{ το πλήθος των μετρήσεων,}$$

όπου:

- $\mathbf{v}[k]$ είναι το καθαρό δυναμικό/σήμα από την δραστηριότητα του εγκεφάλου,
- $\mathbf{d}[k]$ ο θόρυβος (artifact).

→ Για την αποθορυβοποίηση των δεδομένων με Wiener φίλτρο θα εξετάσουμε δύο μεθόδους:

1. Στην πρώτη—offline—μέθοδο (*Sections 1.**), θα προσπαθήσουμε να εκτιμήσουμε τα καθαρά δυναμικά

$$\theta = [\mathbf{v}[0], \mathbf{v}[1], \dots, \mathbf{v}[K-1]],$$

με βάση τις τρέχουσες, παρελθοντικές και μελλοντικές τιμές $[\mathbf{y}[0], \dots, \mathbf{y}[K-1]]$ —*Smoothing*.

2. Στη δεύτερη—online—μέθοδο (*Section 2*), θα εκτιμήσουμε καθένα από τα

$$\theta = \mathbf{v}[k],$$

με βάση μόνο τις τρέχουσες και παρελθοντικές τιμές των μετρήσεων, $[\mathbf{y}[0], \dots, \mathbf{y}[k]]$ —*Filtering*.

→ Για την αρχική περίπτωση, που εφαρμόζουμε την Smoothing μέθοδο, ακολουθούμε δύο προσεγγίσεις. Στην πρώτη—μονοκαναλική—προσέγγιση (*Section 1.1*) κάθε κανάλι θα αντιμετωπίζεται ανεξάρτητα από τα υπόλοιπα, ενώ στην δεύτερη—πολυκαναλική—(*Section 1.2*) θα λαμβάνεται υπόψη η συσχέτιση μεταξύ των καναλιών. Υπενθυμίζουμε ότι ο θόρυβος που οφείλεται στο ανοιγοκλείσιμο των ματιών είναι εντονότερος στα μετωπιαία ηλεκτρόδια.

Σχόλιο

Παρουσιάζουμε και τις δύο προσεγγίσεις αν και περιμένουμε καλύτερα αποτελέσματα στην πολυκαναλική—μιας και σε αυτήν συνυπολογίζουμε την συσχέτιση μεταξύ των καναλιών. Αυτό που έχει αξία να συγκρίνουμε μεταξύ των δύο είναι το πόσο μεγάλη θα είναι η διαφορά των σφαλμάτων που εισάγει η αποθορυβοποίηση στην πράξη. Από αυτήν την διαφορά θα εξάγουμε συμπεράσματα για το κατά πόσο επηρεάζει ο συνυπολογισμός της συσχέτιση μεταξύ των καναλιών την ακρίβεια των αποτελεσμάτων μας.

1.1 Smoothing: *Μονοκαναλικό* Wiener

Για το i -οστό κανάλι συμβολίζουμε ως:

\mathbf{y}_i : τις μετρήσεις, \mathbf{v}_i : το καθαρό δυναμικό, \mathbf{d}_i : τον θόρυβο,
για τις χρονικές στιγμές $k = 0, \dots, K - 1$.

Η εκτίμηση του φίλτρου Wiener είναι:

$$\hat{\mathbf{v}}_i = \mathbf{C}_{vy}^{(i)} (\mathbf{C}_{yy}^{(i)})^{-1} \mathbf{y}_i, \quad i = 1, \dots, N,$$

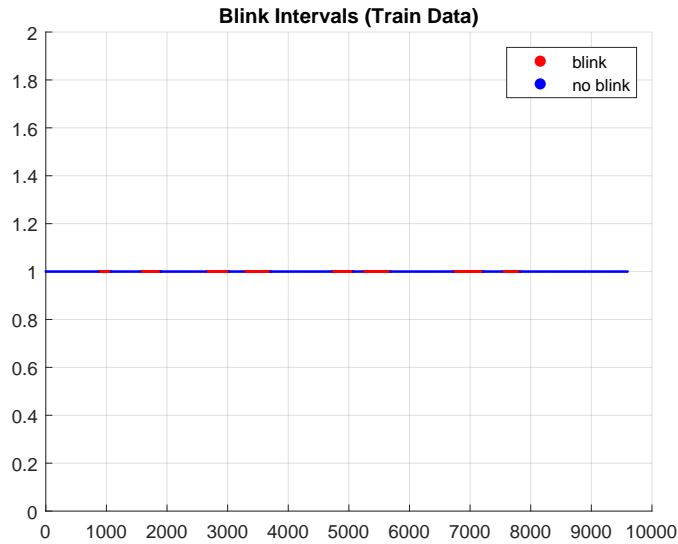
με $\mathbf{C}_{vy}^{(i)}$ τον πίνακα συνδιασποράς των \mathbf{v}_i & \mathbf{y}_i , και $\mathbf{C}_{yy}^{(i)}$ τον πίνακα αυτοδιασποράς του \mathbf{y}_i .

Αν θεωρήσουμε ότι τα \mathbf{y}_i , \mathbf{v}_i , και \mathbf{d}_i είναι **σήματα μηδενικής μέσης τιμής** (οι χρονοσειρές των \mathbf{y}_i δεν απορρίπτουν αυτήν την υπόθεση), και ότι τα \mathbf{v}_i και \mathbf{d}_i είναι μεταξύ τους ανεξάρτητα, έχουμε:

$$\mathbf{C}_{yy}^{(i)} = \mathbb{E}[\mathbf{y}_i \mathbf{y}_i^T] = \mathbf{R}_{yy}^{(i)}$$

$$\mathbf{C}_{vy}^{(i)} = \mathbb{E}[\mathbf{v}_i \mathbf{y}_i^T] = \mathbb{E}[\mathbf{v}_i (\mathbf{d}_i + \mathbf{v}_i)^T] = \mathbf{R}_{vv}^{(i)},$$

με $\mathbf{R}_{yy}^{(i)}$, $\mathbf{R}_{vv}^{(i)}$ τους πίνακες αυτοδιακύμανσης των χρονοσειρών \mathbf{y}_i , \mathbf{v}_i αντίστοιχα.



Σχήμα 1: Διαστήματα με και χωρίς θόρυβο στα δεδομένα εκπαίδευσης

Στα δεδομένα εκπαίδευσης γνωρίζουμε εκ των προτέρων τις χρονικές στιγμές κατά τις οποίες οι μετρήσεις περιέχουν θόρυβο (με τις στιγμές αυτές να είναι προφανώς κοινές για όλα τα κανάλια). Επίσης, από το Σχήμα 1 παρατηρούμε ότι οι χρονικές αυτές στιγμές διαμορφώνουν διαστήματα. Ορίζουμε επομένως:

$$Q = \{k = 0, \dots, K - 1 : \mathbf{y}[k] = \mathbf{v}[k]\} = \bigcup_j Q_j$$

με Q_j το j -οστό διάστημα όπου $\mathbf{y}[k] = \mathbf{v}[k]$ και

$$P = \{k = 0, \dots, K - 1 : \mathbf{y}[k] = \mathbf{d}[k] + \mathbf{v}[k]\} = \bigcup_j P_j$$

με P_j το j -οστό διάστημα όπου $\mathbf{y}[k] = \mathbf{d}[k] + \mathbf{v}[k]$.

Θα εκτιμήσουμε τον πίνακα $\mathbf{R}_{vv}^{(i)}$ από τα δεδομένα απουσίας θορύβου, δηλαδή από το σύνολο Q . Ενώ αντίστοιχα, ο πίνακας $\mathbf{R}_{yy}^{(i)}$ θα προκύψει από τα δεδομένα παρουσίας θορύβου, δηλαδή από το σύνολο P .

Θεωρώντας ότι η χρονοσειρά \mathbf{y}_i είναι *στάσιμη* εντός κάθε διαστήματος Q_j , ο πίνακας αυτοσυσχέτισης $\mathbf{R}_{vv}^{(i,j)}$ του i -οστού καναλιού στο j -οστό διάστημα, Q_j , όπου απουσιάζει ο θόρυβος είναι:

$$\mathbf{R}_{vv}^{(i,j)} = \begin{bmatrix} r_v^{(i,j)}[0] & r_v^{(i,j)}[1] & \cdots & r_v^{(i,j)}[L-1] \\ r_v^{(i,j)}[1] & r_v^{(i,j)}[0] & \cdots & r_v^{(i,j)}[L-2] \\ \vdots & \vdots & \ddots & \vdots \\ r_v^{(i,j)}[L-1] & r_v^{(i,j)}[L-2] & \cdots & r_v^{(i,j)}[0] \end{bmatrix}$$

όπου $r_v^{(i,j)}[\tau] = \mathbb{E}[v_i[n]v_i[n+\tau]]$, με $n, \dots, n+\tau \in Q_j$, είναι η συνάρτηση αυτοσυσχέτισης του \mathbf{v}_i στο διάστημα Q_j , και L η μέγιστη υστέρηση (lag). Προφανώς, η μέγιστη υστέρηση που θα θέσουμε θα πρέπει να είναι μικρότερη από το μικρότερο μήκος διαστήματος με ή χωρίς θόρυβο, δηλαδή:

$$L \leq \min \left\{ \min_j |Q_j|, \min_j |P_j| \right\},$$

όπου ο συμβολισμός $|S|$ εκφράζει το πλήθος των στοιχείων του συνόλου S .

Από την παραπάνω ανάλυση έχουμε ότι ο πίνακας αυτοσυσχέτισης $\mathbf{R}_{vv}^{(i)}$ μπορεί να προκύψει από τον σταθμισμένο μέσο όρο όλων των $\mathbf{R}_{vv}^{(i,j)}$ ως εξής:

$$\mathbf{R}_{vv}^{(i)} = \frac{1}{|Q|} \sum_j |Q_j| \mathbf{R}_{vv}^{(i,j)}, \quad i = 1, \dots, N$$

Εργαζόμενοι κατά ανάλογο τρόπο για τα δεδομένα παρουσία θορύβου, έχουμε:

$$\mathbf{R}_{yy}^{(i)} = \frac{1}{|P|} \sum_j |P_j| \mathbf{R}_{yy}^{(i,j)}, \quad i = 1, \dots, N$$

Έχοντας υπολογίσει τους δύο παραπάνω πίνακες αυτοσυσχέτισης, μπορούμε να βρούμε τον πίνακα Wiener για το i -οστό κανάλι:

$$\mathbf{W}^{(i)} = \mathbf{R}_{vv}^{(i)} (\mathbf{R}_{yy}^{(i)})^{-1} \in \mathbb{R}^{L \times L}$$

Το φίλτρο εφαρμόζεται διαδοχικά στις μετρήσεις του εκάστοτε καναλιού, σε παράθυρα μήκους L , σύμφωνα με τη σχέση:

$$\hat{\mathbf{v}}_i[k : k + L - 1] = \mathbf{W}^{(i)} \mathbf{y}_i[k : k + L - 1], \quad i = 1, \dots, N.$$

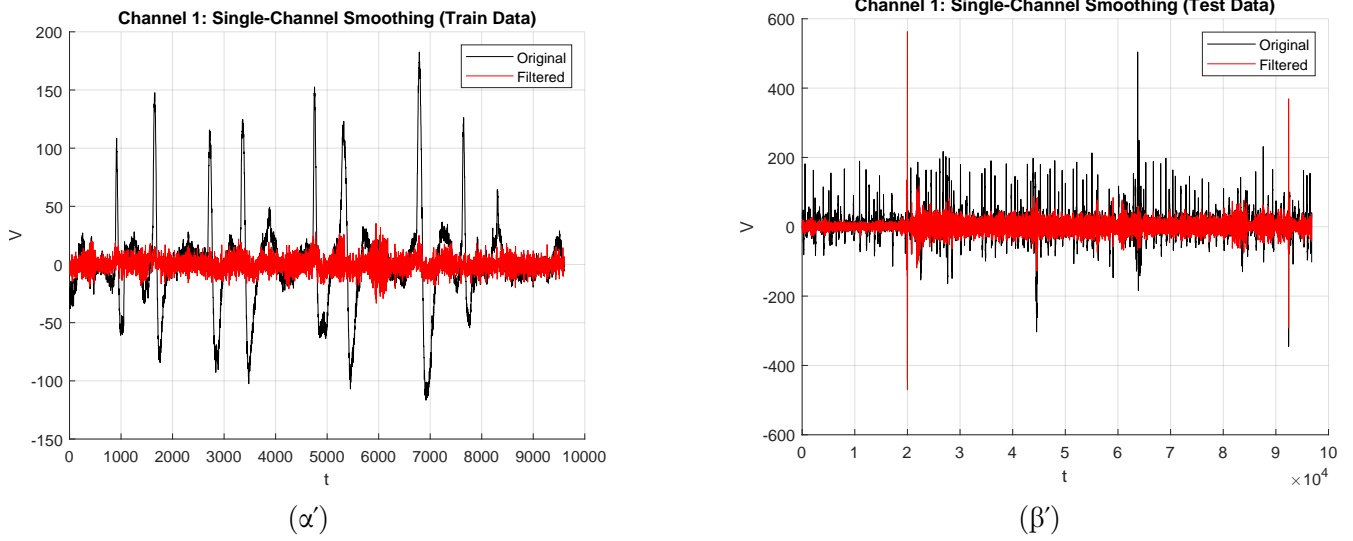
Από την πάνω σχέση είναι προφανές ότι για την εκτίμηση του \mathbf{v}_i χρησιμοποιούνται κάθε φορά οι τελευταίες L μετρήσεις του \mathbf{y}_i .

Σχόλιο

Είναι σημαντικό να τονίσουμε σε αυτό το σημείο ότι η συνάρτηση αυτοσυσχέτισης, και συνακόλουθα ο πίνακας αυτοσυσχέτισης, λαμβάνει υπόψιν μόνο γραμμικές συσχετίσεις. Συνεπώς, το φίλτρο Wiener βασίζεται στην υπόθεση της γραμμικής αυτοσυσχέτισης και δεν λαμβάνει υπόψιν μη γραμμικές συσχετίσεις μεταξύ των δεδομένων της χρονοσειράς.

Στο Σχήμα 2 φαίνεται η εφαρμογή του φίλτρου στα δεδομένα εκπαίδευσης (αριστερά) και στα δεδομένα ελέγχου (δεξιά) για το κανάλι 1. Παρατηρούμε ότι στα δεδομένα εκπαίδευσης η εξομάλυνση των μετρήσεων όπου υπάρχει θόρυβος (διαστήματα με μεγάλες διακυμάνσεις) είναι επιτυχής με κόστος ωστόσο την αλλοίωση της πληροφορίας στα διαστήματα όπου δεν υπάρχει θόρυβος. Επίσης, βλέπουμε ότι στα δεδομένα ελέγχου το φίλτρο αδυνατεί να επιτύχει την αποθορυβοποίηση του σήματος όταν παρουσιάζονται μεγάλα spikes του δυναμικού $> 200V$. Αυτό εξηγείται από το

γεγονός ότι το Wiener φίλτρο βασίζεται σε στατιστικά χαρακτηριστικά που υπολογίζονται από τα δεδομένα εκπαίδευσης, όπου τέτοιες ακραίες τιμές δεν εμφανίζονται. Τα spikes είναι μη-γραμμικά και υψηλής ενέργειας γεγονότα που παραβιάζουν τις υποθέσεις στάσιμης και γραμμικής συσχέτισης πάνω στις οποίες βασίζεται το φίλτρο, με αποτέλεσμα την αδυναμία του να τα εξομαλύνει επαρκώς.



Σχήμα 2: Εφαρμογή μονοκαναλικού Smoothing Wiener Filter στα α) δεδομένα εκπαίδευσης β) δεδομένα ελέγχου

1.2 Smoothing: Πολυκαναλικό Wiener

Μέχρι στιγμής, στην ανάλυσή μας εξετάζαμε κάθε κανάλι μεμονωμένα, αγνοώντας τη συσχέτιση που υπάρχει μεταξύ των μετρήσεών τους. Αυτό, ωστόσο, δεν είναι απολύτως ορθό, καθώς ο θόρυβος είναι εντονότερος στα μετωπιαία ηλεκτρόδια· επομένως, υπάρχει συσχέτιση μεταξύ των μετρήσεων διαφορετικών καναλιών. Στην παρούσα ενότητα θα συμπεριλάβουμε αυτές τις συσχετίσεις μεταξύ των καναλιών.

Έστω $\mathbf{Y}_t \in \mathbb{R}^{N \cdot L}$ το διάνυσμα υστέρησης, το οποίο περιέχει τις τιμές όλων των καναλιών σε ένα παράθυρο μήκους L :

$$\mathbf{Y}_t = \begin{bmatrix} \mathbf{y}_1[t : t + L - 1] \\ \mathbf{y}_2[t : t + L - 1] \\ \vdots \\ \mathbf{y}_N[t : t + L - 1] \end{bmatrix}$$

όπου $\mathbf{y}_i[t : t + L - 1] \in \mathbb{R}^L$ είναι οι μετρήσεις του i -οστού καναλιού. Επίσης, γράφουμε $\mathbf{Y}_t \equiv \mathbf{Y}_t^{(j)}$, με $t, \dots, t + L - 1 \in P_j$. Χωρίς βλάβη της γενικότητας, θεωρούμε ότι το διάστημα P_j ξεκινά από τη χρονική στιγμή $t = 1$.

Σχηματίζουμε τον πίνακα:

$$\mathbf{Y}^{(j)} = \begin{bmatrix} \mathbf{Y}_1^{(j)} & \mathbf{Y}_2^{(j)} & \dots & \mathbf{Y}_T^{(j)} \end{bmatrix} \in \mathbb{R}^{N \cdot L \times T}, \quad T = |P_j| - L + 1,$$

ή

$$\mathbf{Y}^{(j)} = \begin{bmatrix} \mathbf{y}_1[1:L] & \mathbf{y}_1[2:L+1] & \cdots & \mathbf{y}_1[T-L+1:T] \\ \mathbf{y}_2[1:L] & \mathbf{y}_2[2:L+1] & \cdots & \mathbf{y}_2[T-L+1:T] \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{y}_N[1:L] & \mathbf{y}_N[2:L+1] & \cdots & \mathbf{y}_N[T-L+1:T] \end{bmatrix}$$

Αν υποθέσουμε ότι $\mathbb{E}[\mathbf{Y}_t^{(j)}] = 0$, για κάθε $t = 1, \dots, T$, τότε ο πίνακας συσχέτισης $\mathbf{R}_{yy}^{(j)}$ για το διάστημα P_j δίνεται από:

$$\mathbf{R}_{yy}^{(j)} = \frac{1}{T} \sum_{i=1}^T \mathbf{Y}_i^{(j)} (\mathbf{Y}_i^{(j)})^\top = \frac{1}{T} \mathbf{Y}^{(j)} \cdot (\mathbf{Y}^{(j)})^\top$$

Αν, επιπλέον, οι χρονοσειρές δύο καναλιών p και q είναι από κοινού στάσιμες, δηλαδή ισχύει:

$$\mathbb{E}[\mathbf{y}_p[k] \cdot \mathbf{y}_q[l]] = r_{pq}[k-l] \in \mathbb{R},$$

δηλαδή η εταιροσυσχέτισή τους εξαρτάται μόνο από την υστέρηση $\tau = k-l$ και όχι από τον χρόνο, τότε:

$$\mathbf{R}_{yy}^{(j)} = \begin{bmatrix} \mathbf{R}_{11} & \mathbf{R}_{12} & \cdots & \mathbf{R}_{1N} \\ \mathbf{R}_{21} & \mathbf{R}_{22} & \cdots & \mathbf{R}_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{R}_{N1} & \mathbf{R}_{N2} & \cdots & \mathbf{R}_{NN} \end{bmatrix}$$

όπου $\mathbf{R}_{pq} \in \mathbb{R}^{L \times L}$ είναι ο πίνακας συνδιασποράς μεταξύ των καναλιών p και q για υστερήσεις $\tau = 0, \dots, L-1$. Το στοιχείο $[\mathbf{R}_{pq}]_{k,l} = r_{pq}[k-l]$ δείχνει πόσο συσχετίζεται η τιμή του καναλιού p για υστέρηση $k-1$ με την τιμή του καναλιού q για υστέρηση $l-1$.

Όπως και στη μονοκαναλική προσέγγιση, ο πίνακας \mathbf{R}_{yy} δίνεται από τον σταθμισμένο μέσο όρο των πινάκων διασποράς για όλα τα διαστήματα όπου υπάρχει θόρυβος:

$$\mathbf{R}_{yy} = \frac{1}{|P|} \sum_j |P_j| \mathbf{R}_{yy}^{(j)}$$

Εργαζόμενοι κατ' αντίστοιχο τρόπο στα διαστήματα Q_j όπου απουσιάζει ο θόρυβος, προκύπτει ο πίνακας \mathbf{R}_{vv} .

Ο πίνακας Wiener (Wiener Smoothing Matrix) θα είναι:

$$\mathbf{W} = \mathbf{R}_{vv} \mathbf{R}_{yy}^{-1} \in \mathbb{R}^{N \cdot L \times N \cdot L}$$

Και εδώ το φίλτρο εφαρμόζεται διαδοχικά σε παράθυρα μήκους L σύμφωνα με την σχέση:

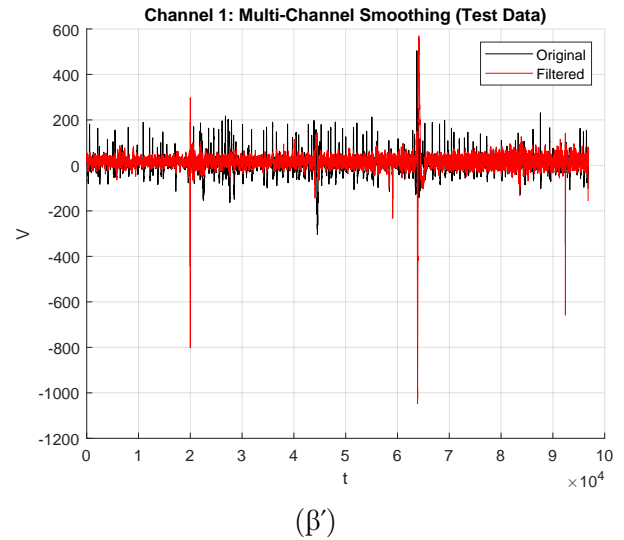
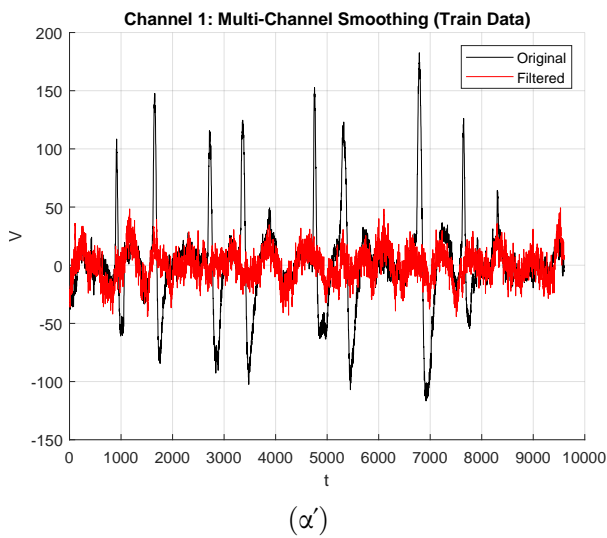
$$\hat{\mathbf{V}}_k = \mathbf{W} \cdot \mathbf{Y}_k$$

όπου $\hat{\mathbf{V}}_k \in \mathbb{R}^{N \cdot L}$ το διάνυσμα (πίνακας στήλης) με τις εκτιμήσεις του καθαρού σήματος όλων των καναλιών στο χρονικό παράθυρο $k, \dots, k+L-1$.

Στο Σχήμα 3 φαίνεται η εφαρμογή του φίλτρου στα δεδομένα εκπαίδευσης (αριστερά) και στα δεδομένα ελέγχου (δεξιά) για το κανάλι 1. Είναι εμφανές ότι, σε αυτή την περίπτωση, η απώλεια πληροφορίας στα καθαρά δεδομένα εκπαίδευσης είναι μικρότερη σε σχέση με τη μονοκαναλική προσέγγιση. Αυτό οφείλεται στο γεγονός ότι στην πολυκαναλική προσέγγιση λαμβάνουμε υπόψη τη

συσχέτιση μεταξύ των καναλιών, οδηγώντας σε μικρότερες τιμές μέσου τετραγωνικού σφάλματος (άρα και RMSE).

Ωστόσο, παρατηρούμε και σε αυτή την περίπτωση ότι το φίλτρο αδυνατεί να εξομαλύνει spikes του δυναμικού $> 200V$, καθώς, όπως έχουμε αναφέρει, οφείλονται σε μη γραμμικές συσχετίσεις που δεν λαμβάνει υπόψη το φίλτρο Wiener.



Σχήμα 3: Εφαρμογή πολυκαναλικού Smoothing Wiener Filter στα α) δεδομένα εκπαίδευσης β) δεδομένα ελέγχου

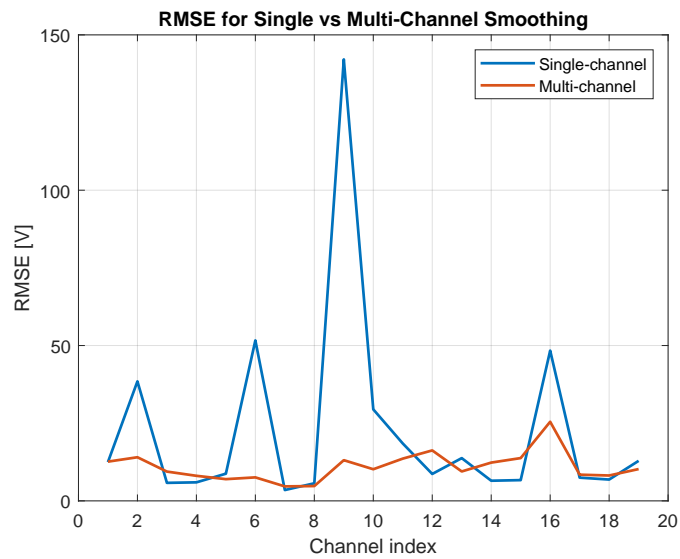
Υλοποίηση και Σύγκριση Φίλτρων 1.1 & 1.2

Στο παρακάτω τμήμα κώδικα παρουσιάζεται η κατασκευή των πινάκων αυτοσυσχέτισης R_{ss} και R_{xx} , για το μονοκαναλικό Wiener Smoothing. Τα δεδομένα έχουν προηγουμένως χωριστεί σε `καθαρά` και `θορυβώδη` χρονικά διαστήματα, με βάση τις θέσεις των blinks. Για κάθε τέτοιο διάστημα, υπολογίζεται ο αντίστοιχος πίνακας αυτοσυσχέτισης, ο οποίος στη συνέχεια προστίθεται στον συνολικό πίνακα με βάρος (συντελεστή α) ανάλογο του σχετικού μήκους του διαστήματος σε σχέση με το πλήθος των συνολικών καθαρών ή θορυβωδών δειγμάτων. Σημειώνεται ότι, για κάθε διάστημα, από τα δεδομένα αφαιρείται η μέση τιμή όλων των καθαρών ή θορυβωδών δειγμάτων, και όχι μόνο αυτών του εκάστοτε διαστήματος.

```
1 noisy_mean = mean(x_train(blinks));
2 clean_mean = mean(x_train(not_blinks));
3
4 for i = 1:length(cleanIntervals)
5     idx = cleanIntervals{i}(1):cleanIntervals{i}(2);
6     data = x_train(idx) - clean_mean;
7     r_full = xcorr(data, M - 1, 'unbiased');
8     alpha = length(idx) / n_clean;
9     R_ss = R_ss + alpha * toeplitz(r_full(M:end));
10 end
11
12 for i = 1:length(noisyIntervals)
13     idx = noisyIntervals{i}(1):noisyIntervals{i}(2);
14     data = x_train(idx) - noisy_mean;
15     r_full = xcorr(data, M - 1, 'unbiased');
16     alpha = length(idx) / n_noisy;
17     R_xx = R_xx + alpha * toeplitz(r_full(M:end));
18 end
```

Στην μονοκαναλική περίπτωση, ο πίνακας αυτοσυσχέτισης R_{ss} υπολογίζεται με τη συνάρτηση `xcorr` και μετατρέπεται σε πίνακα Toeplitz. Αντίθετα, στην πολυκαναλική περίπτωση, χρησιμοποιείται ένας υστέρησης X που ενσωματώνει τόσο χρονικές όσο και διακαναλικές εξαρτήσεις, και ο R_{ss} προκύπτει ως ο σταθμισμένος μέσος όρος των XX^T . Παρακάτω φαίνεται ο τρόπος υπολογισμού του πίνακα αυτοσυσχέτισης R_{ss} καθώς και η συνάρτηση που κατασκευάζει τον πίνακα υστερήσεων.

```
1 for i = 1:length(cleanIntervals)
2     idx = cleanIntervals{i}(1):cleanIntervals{i}(2);
3     data = x_train(:, idx) - clean_mean;
4     X = build_lagged_matrix(data, L); % [C*M x T]
5     alpha = length(idx) / n_clean;
6     R_ss = R_ss + alpha * (X * X') / size(X, 2);
7 end
8
9 function X = build_lagged_matrix(data, M)
10     % data: [C x T]
11     [C, T] = size(data);
12     T_windows = T - M + 1;
13     X = zeros(C * M, T_windows);
14
15     for t = 1:T_windows
16         block = data(:, t:t+M-1); % [C x M]
17         X(:, t) = block(:); % [C*M x 1]
18     end
19 end
```



Σχήμα 4: RMSE μονοκαναλικού και πολυκαναλικού Smoothing για όλα τα κανάλια

Από το Σχήμα 4 παρατηρούμε ότι το πολυκαναλικό *Smoothing* επιτυγχάνει μικρότερες τιμές RMSE σε σύγκριση με το μονοκαναλικό στην πλειονότητα των καναλιών. Αυτό οφείλεται στο γεγονός ότι η πολυκαναλική προσέγγιση αξιοποιεί τις διακαναλικές συσχετίσεις κατά την αποθρομβοποίηση, μειώνοντας την απώλεια πληροφορίας. Συγκριτικά, η μέση τιμή του RMSE για το μονοκαναλικό *Smoothing* είναι 22.814, ενώ για το πολυκαναλικό 11.0073.

2 Filtering: Πολυκαναλικό Wiener

Motivation

Με την παραπάνω ανάλυση που αφορούσε το Smoothing (και ιδιαίτερα την πολυκαναλική προσέγγιση) έχουμε ήδη καλύψει το βασικό πρόβλημα της υλοποίησης ενός Wiener φίλτρου για την αποθορυβοποίηση των δεδομένων μας από τα artifacts των blinks. Από εδώ και πέρα υπάρχουν πολλές εναλλακτικές κατευθύνσεις τις οποίες θα μπορούσαμε να εξερευνήσουμε, χτίζοντας πάνω στην βασική ανάλυση που παραθέσαμε.

Αυτήν που στην συνέχεια θα δείξουμε είναι η περίπτωση που, αντί για Smoothing, επιλέγουμε να κάνουμε Filtering. Ο λόγος για τον οποίο θεωρούμε πως έχει σημαντικό ενδιαφέρον είναι επειδή πρόκειται για μια online μέθοδο. Άρα, αν θέλαμε να εφαρμόσουμε Wiener φίλτρο που να προσαρμόζει τον πίνακα W σε πραγματικό χρόνο (όταν έρχεται καινούρια πληροφορία με την οποία μπορεί να κάνει training), τότε η μέθοδος που δείξαμε παραπάνω δεν θα μας ήταν χρήσιμη—μιας και όπως είδαμε σε αυτήν το signal sample εκτιμάται και από "μελλοντικά" data.

"Στο Filtering πρόβλημα καλούμαστε να εκτιμήσουμε κάθε φορά το σήμα $\mathbf{v}[k]$ από το dataset των $[\mathbf{y}[0], \dots, \mathbf{y}[k]]$. Το πρόβλημα αυτό είναι επαναλαμβανόμενο, για κάθε τιμή του k , έως ότου ολοκληρω το σήμα $\mathbf{v}[k]$, για $k = 0, 1, \dots, K - 1$ να έχει εκτιμηθεί."

(from Fundamentals of Statistical Signal Processing, Volume-i Estimation Theory)

Από το παραπάνω έχουμε ότι στόχος μας σε αυτό το πρόβλημα, για κάθε $k = 0, 1, \dots, K - 1$, είναι ο υπολογισμός του $\hat{\mathbf{v}}[k] = \mathbf{W} \cdot \mathbf{Y}_k \in \mathbb{R}^N$, όπου το $\mathbf{Y}_k \in \mathbb{R}^{N \cdot L}$ είναι όπως ορίστηκε στην προηγούμενη ανάλυση με την μόνη διαφορά ότι αναφέρεται στο dataset των $[\mathbf{y}[0], \dots, \mathbf{y}[k]]$ (και με κατάλληλα μικρό L για μικρές τιμές του k).

Ο δε πίνακας W , Wiener Filtering Matrix, αυτήν την φορά δίνεται ως:

$$\mathbf{W} = \mathbf{r}_{vv}'^T \mathbf{R}_{yy}^{-1} \in \mathbb{R}^{N \times N \cdot L}$$

Έχουμε ήδη δείξει στο προηγούμενο Section πως προκύπτουν τα \mathbf{Y}_k και \mathbf{R}_{yy} από ένα dataset $[\mathbf{y}[0], \dots, \mathbf{y}[k]]$. Το μόνο που μένει να αναλυθεί εκτενέστερα είναι το $\mathbf{r}_{vv}'^T$ για το οποίο έχουμε:

$$\mathbf{r}_{vv}'^T = [\mathbf{r}_{vv}[L - 1] \quad \mathbf{r}_{vv}[L - 2] \quad \dots \quad \mathbf{r}_{vv}[0]] \in \mathbb{R}^{N \times N \cdot L}.$$

Παρατηρώντας τον παραπάνω τύπο σε σχέση με αυτόν του \mathbf{R}_{vv} (που προκύπτει αντίστοιχα με τον \mathbf{R}_{yy}), στο Wiener Smoothing Matrix, μπορεί κανείς να δει ότι υπάρχει μια αντιστοιχία στο περιεχόμενο των πινάκων.

Στην πράξη αυτό που αλλάζει είναι ότι τώρα δεν θέλουμε να εκτιμήσουμε όλες τις τιμές $[\hat{\mathbf{v}}[k - L + 1], \dots, \hat{\mathbf{v}}[k]]$ (όπως κάνουμε με το Smoothing για παράθυρα υστέρησης L)—παρά μόνο την τελική $\hat{\mathbf{v}}[k]$. Άρα, από αυτό προκύπτει ότι δεν χρειάζεται να πάρουμε τον πίνακα αυτοσυσχετίσης με όλες τις υστερήσεις $\mathbf{R}_{vv} \in \mathbb{R}^{N \cdot L \times N \cdot L}$, αλλά απλώς χρειαζόμαστε τις αυτοσυσχετίσεις που υπάρχουν στον πίνακα $\mathbf{r}_{vv}'^T \in \mathbb{R}^{N \times N \cdot L}$ (που αφορούν μόνο την εκτίμηση της $\hat{\mathbf{v}}[k]$). Οι αυτοσυσχετίσεις αυτές περιέχονται και στον πίνακα \mathbf{R}_{vv} , συνεπώς και ο τρόπος/υλοποίηση με τον οποίο χρειάζεται να τις υπολογίσουμε είναι ισοδύναμος της ανάλυσης που ήδη παραθέσαμε.

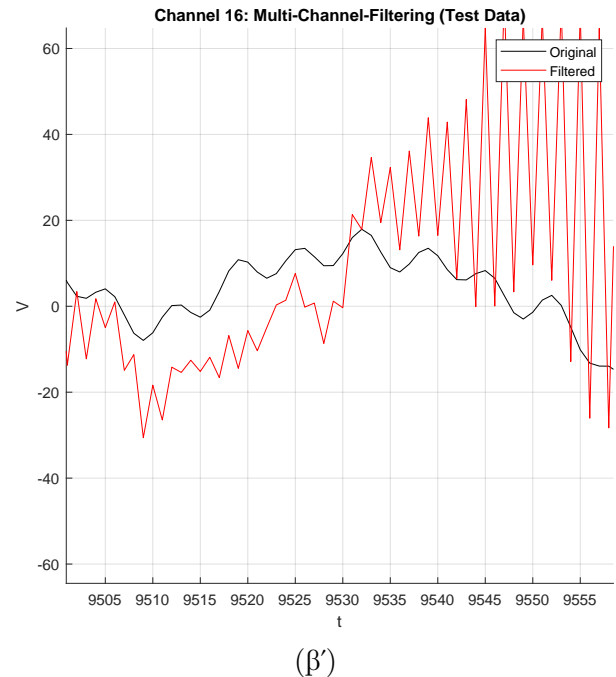
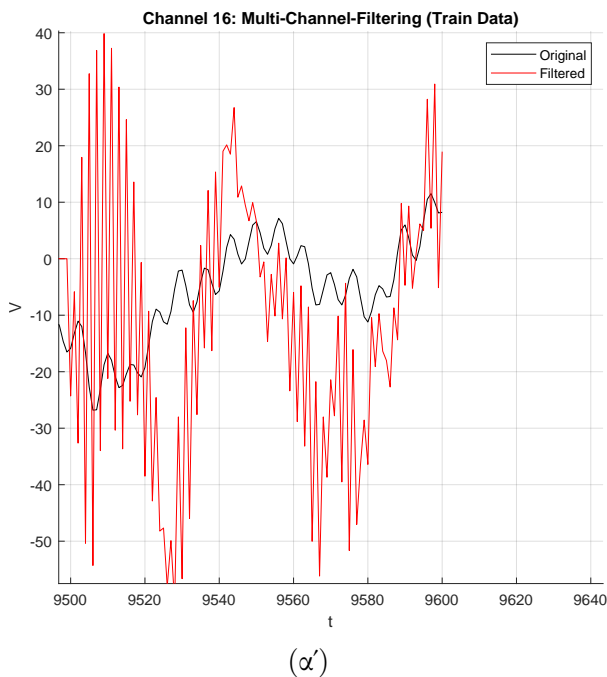
Έχοντας ολοκληρώσει την ανάλυση που αφορά το πρόβλημα του Filtering, μιας και πλέον για τις εκτιμήσεις των $\hat{\mathbf{v}}[k]$ όλοι οι προς υπολογισμό πίνακες έχουν εξηγηθεί, μπορούμε να περάσουμε στο επόμενο Section στο οποίο παραθέτουμε λεπτομέρειες της υλοποίησης του κώδικα και σχετικές γραφικές παραστάσεις.

Υλοποίηση και Αποτελέσματα Wiener Filtering

Για την επίλυση του Filtering προβλήματος στο MATLAB υλοποιήσαμε την συνάρτηση: `function [v_train, v_test, rmse] = wiener_filtering_multichannel(x_train, x_test, blinks, M)`. Αυτήν αποτελεί μία παραλλαγή της αντίστοιχης συνάρτησης για πολυκαναλικό Smoothing φίλτρο.

Αυτήν την φορά καλούμαστε, σύμφωνα με την θεωρητική ανάλυση που παραθέσαμε, να εκτιμήσουμε όλα τα $\hat{v}[k]$, ένα την φορά—μίας και λύνουμε επαναληπτικά το πρόβλημα για όλα τα υποσύνολα $[y[0], \dots, y[k]]$. Αυτό έχει ως αποτέλεσμα να έχουμε μία πιο αργή επίλυση. Γι' αυτό παραθέτουμε την εφαρμογή του φίλτρου σε ένα μικρό portion του αρχικού dataset (τόσο για τα training όσο και για τα test data).

Τρέχοντας το τελευταίο section του εκτελέσιμου MATLAB κώδικα (που μπορεί να βρεθεί σε αυτό το repo, μαζί με τα util functions και τα datasets) παίρνουμε τις παρακάτω γραφικές παραστάσεις. Στο Σχήμα 5 φαίνεται η εφαρμογή του φίλτρου στα δεδομένα εκπαίδευσης (αριστερά) και στα δεδομένα ελέγχου (δεξιά) για το κανάλι 16. Η επιλογή του καναλιού που θέλουμε να κάνουμε plot μπορεί να αλλάξει μέσω της μεταβλητής `plot_channel` (στο αρχείο `main.m` του πηγαίου κώδικα).



Σχήμα 5: Εφαρμογή πολυκαναλικού Filtering Wiener Filter στα α) δεδομένα εκπαίδευσης β) δεδομένα ελέγχου

Είναι εμφανές ότι, σε αυτή την περίπτωση, η απώλεια πληροφορίας στα καθαρά δεδομένα εκπαίδευσης είναι επίσης σχετικά μικρή (όπως και στην smoothing πολυκαναλική προσέγγιση). Αυτό οφείλεται στο γεγονός ότι στην πολυκαναλική προσέγγιση λαμβάνουμε υπόψη τη συσχέτιση μεταξύ των καναλιών, οδηγώντας και πάλι σε μικρές τιμές μέσου τετραγωνικού σφάλματος (άρα και RMSE).

```
1 for p = 9500:size(x_train, 2)
2   x_train_ = x_train(:, 1:p);
3   x_test_  = x_test(:, 1:p);
4   blinks_  = blinks(blinks <= p);
```

```

5
6 [C, N] = size(x_train_);
7 [cleanIntervals, noisyIntervals, max_window_size] = find_intervals(blinks_, N);
8
9 % Same as smoothing for the x_train_ and x_test_ datasets
10
11 % Estimate R_ss
12 % (...)
13 % Estimate R_xx
14 % (...)
15
16 % Wiener filter
17 W = R_ss / R_xx;
18
19 % Apply Wiener filter on training and testing data
20 s_train = apply_wiener_filtering_multichannel(x_train_, W, M);
21 s_test = apply_wiener_filtering_multichannel(x_test_, W, M);
22
23 v_train(:, p) = s_train(:, size(s_train, 2));
24 v_test(:, p) = s_test(:, size(s_test, 2));
25 end
26 % Compute RMSE across channels (on clean intervals only)
27 % ...

```

Listing 1: Computation of Autocorrelation Matrices for Wiener Filter

Η μέθοδος αυτή όμως, σε σχέση με την Smoothing, δεν είναι το ίδιο αποδοτική μιας και εδώ ο πίνακας W με το "πέρασ του χρόνου" (δηλαδή με περισσότερες μετρήσεις από το train dataset) κάνει όλο και καλύτερες προσαρμογές (ενώ η Smoothing κάνει όλες τις προσαρμογές εξ' αρχής, έχοντας πρόσβαση σε όλα τα δεδομένα). Οι προσαρμογές στον χρόνο για τον πίνακα W γίνονται όσο διαβάζουμε το train dataset, ενώ για λόγους σύγκρισης χρησιμοποιούμε (κάθε χρονική στιγμή) τον ίδιο πίνακα W και για το test dataset.

Σχόλιο:

Για τον υπολογισμό του Wiener Matrix (στον κώδικα που παρατίθεται), για χάρη συνοχής και ευκολίας στην συγγραφή του κώδικα, ακολουθούμε την ίδια μέθοδο με αυτήν του Smoothing για το subset του αρχικού dataset που μελετάμε. Όπως εξηγήσαμε και στην θεωρητική ανάλυση, η διαφορά έγκειται στο ότι μας ενδιαφέρει *μόνο* το $\hat{\mathbf{v}}[k]$ και όχι τα προηγούμενα lags/υστερήσεις (μιας και τις υστερήσεις τις έχουμε ήδη υπολογίσει στα προηγούμενα iterations της for loop). Θα μπορούσαμε για υπολογιστική απλότητα να αποφύγουμε τις έξτρα πράξεις (όπως δείχνουμε και στην θεωρητική ανάλυση), αλλά ούτως ή άλλως εστιάσαμε την προσοχή μας σε ένα μόνο σημείο του dataset.

Γενική Παρατήρηση

Στην ανάλυση που έχουμε δείξει ως τώρα, για όλες τις διαφορετικές προσεγγίσεις εφαρμογής φίλτρων Wiener, εφαρμόζαμε το φίλτρο σε όλο το dataset—είτε υπάρχουν blinks είτε όχι. Αυτό έχει ως αποτέλεσμα στις περιπτώσεις που υπάρχουν blinks (ή τουλάχιστον που ξέρουμε ότι υπάρχουν/train dataset) να βλέπουμε όντως μια προσπάθεια αποθρομβοποίησης από το φίλτρο. Όμως στις περιπτώσεις όπου δεν υπάρχουν blinks, με το να περνάμε τα σήματα από το φίλτρο οδηγούμαστε στην αλλοίωσή τους (το φίλτρο ελαφρώς παραμορφώνει το σήμα πληροφορίας).

Ένας τρόπος να το αποφύγουμε αυτό είναι να εκτιμήσουμε πρώτα σε ποιά χρονικά διαστήματα του dataset υπάρχουν blinks/θόρυβος. Γνωρίζουμε από την εισαγωγή της εργασίας ότι ο θόρυβος αυτός "εμφανίζεται συχνά με πολύ μεγαλύτερο πλάτος από την κοινή εγκεφαλική δραστηριότητα που καταγράφεται από το ΗΕΓ". Επίσης ξέρουμε ότι επηρεάζει ειδικά τα μετωπιαία ηλεκτρόδια. Αξιοποιώντας αυτές τις πληροφορίες μπορούμε να εφαρμόσουμε τεχνικές ανίχνευσης των διαστημάτων όπου έχουμε blinks, μελετώντας αμιγώς τα κανάλια που αφορούν τα μετωπιαία ηλεκτρόδια.

Στην συνέχεια, έχοντας βρει τα διαστήματα αυτά, μπορούμε να εφαρμόσουμε Wiener φίλτρο μόνο στα σημεία όπου υπάρχουν blinks. Βέβαια είναι σημαντικό να αναφέρουμε πως η τεχνική ανίχνευσης των blinks στο dataset πρέπει να επιλεγεί με ιδιαίτερη προσοχή. Συγκεκριμένα, πρέπει η μέθοδος ανίχνευσης να μπορεί να βρίσκει με όσο το δυνατόν μεγαλύτερη πιθανότητα όλα τα δείγματα στα οποία έχουμε blinks. Με άλλα λόγια η πιθανότητα να μην ανιχνεύσουμε ένα blink πρέπει να είναι ιδιαίτερα μικρή. Αντίθετα, μπορούμε να συμβιβαστούμε (trade-off) στο να έχουμε μεγάλη πιθανότητα σε false positives, δηλαδή στο να ανιχνεύει blinks ενώ στην πραγματικότητα δεν υπάρχουν. Ο λόγος που δεν μας επηρεάζουν τόσο τα false positives είναι επειδή η εναλλακτική μας θα ήταν να περνούσαμε όλο το dataset από το φίλτρο (η παραμόρφωση του σήματος δεν είναι ιδιαίτερα μεγάλη).