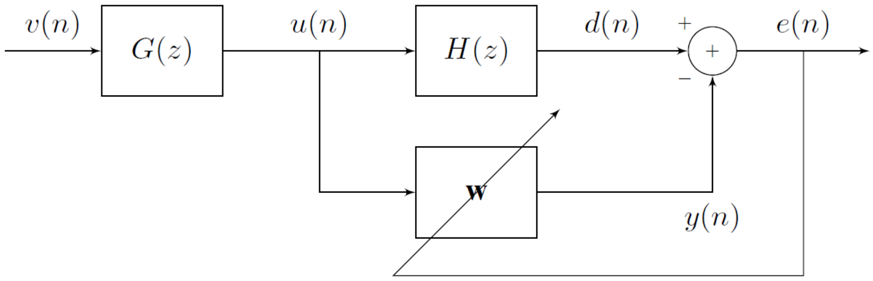
ΨΗΦΙΑΚΑ ΦΙΛΤΡΑ

2η Εργασία: Προσαρμογή στο πεδίο της συχνότητας

Βαγενάς Θεόδωρος- Παναγιώτης

Α.Ε.Μ: 8112

Θεωρητική ανάλυση



$$u(n) = -0.34\,u(n-1) + v(n)$$

α)Μετατροπή του DFT σε FFT $(\boldsymbol{n} = \mathbf{2}^{\boldsymbol{q}})$

$$\boldsymbol{x_k} = \sum_{j=0}^{2^q-1} x_j e^{-2\pi i \frac{jk}{2^q}} = \sum_{j=0}^{2^{q-1}-1} x_{(2j)} e^{-2\pi i \frac{2jk}{2^q}} + \sum_{j=0}^{2^{q-1}-1} x_{(2j+1)} e^{-2\pi i \frac{(2j+1)k}{2^q}} =$$

$$= \sum_{j=0}^{2^{q-1}-1} x_{(2j)} e^{-2\pi i \frac{jk}{2^{q-1}}} + e^{-2\pi i \frac{k}{2^q}} \sum_{j=0}^{2^{q-1}-1} x_{(2j+1)} e^{-2\pi i \frac{jk}{2^{q-1}}} \quad , k = 0,1,\dots,\frac{n}{2} - 1$$

$$W_N = e^{\frac{2\pi i}{N}} \quad , \quad W_{\frac{N}{2}} = {W_N}^2 \quad , W_{N/2}^{-jk} = e^{-2\pi i \frac{jk}{2^{q-1}}} \quad , \quad W_N^{-k} = e^{-2\pi i \frac{k}{2^q}}$$

$$X[k] = X_0[k] + W_N^{-k} * X_1[k]$$

$$\boldsymbol{x} = \begin{bmatrix} \boldsymbol{x}^{(1)} + Wn\,\boldsymbol{x}^{(2)} \\ \boldsymbol{x}^{(1)} - Wn\,\boldsymbol{x}^{(2)} \end{bmatrix} \qquad , Wn = diag([w_n^0\, w_n^1 \dots w_n^{k-1}])$$

Στο αρχείο matlab fftproof.m έχει συμπεριληφθεί η υλοποίηση του fft βασιζόμενη στον DFT όπως παρουσιάστηκε θεωρητικά παραπάνω. Το σφάλμα όπως φαίνεται είναι ίδιο με αυτό του fft της τάξεως του 10^{-15} οπότε η απόδειξη είναι σωστή.

β) FFT recursive

Στο αρχείο fft_recursive.m υπάρχει μία αναδρομική συνάρτηση για τον υπολογισμό του FFT για είσοδο της οποίας το μήκος είναι δύναμη του 2.

Όσων αφορά την πολυπλοκότητα ο DFT απαιτεί:

$(N-1)^2$ *μιγαδικούς πολλαπλασιασμούς ΚΑΙ* $N(N-1)$ *μιγαδικές προσθέσεις* το οποίο ισοδυναμεί με $FLOPS = 6(N-1)^2 + 2N(N-1)$

$$DFT = O(N^2)$$

Όσων αφορά την πολυπλοκότητα ο FFT απαιτεί:

Αριθμός μιγαδικών προσθέσεων: $A(N) = 2A\left(\frac{N}{2}\right) + N$

Αριθμός μιγαδικών πολλαπλασιασμών:$M(N) = 2M\left(\frac{N}{2}\right) + (\frac{N}{2} - 1)$

Άρα:

$\frac{N}{2}log_2N - N + 1 \;\; μιγαδικούς\ πολλαπλασιασμούς$

$KAI\ Nlog_2N\ μιγαδικές\ προσθέσεις$

το οποίο ισοδυναμεί με $FLOPS = 6\ (\frac{N}{2}log_2N - N + 1) + 2Nlog_2N$

$$FFT = O(Nlog_2N)$$

Η πολυπλοκότητα αυτή αποδεικνύεται και μέσω του MASTER THEOREM καθώς έχουμε:$T(n) = 2T\left(\frac{n}{2}\right) + cn \;\; και\ cn = \Theta(n)\ άρα\ T(n) = \Theta(nlogn)$

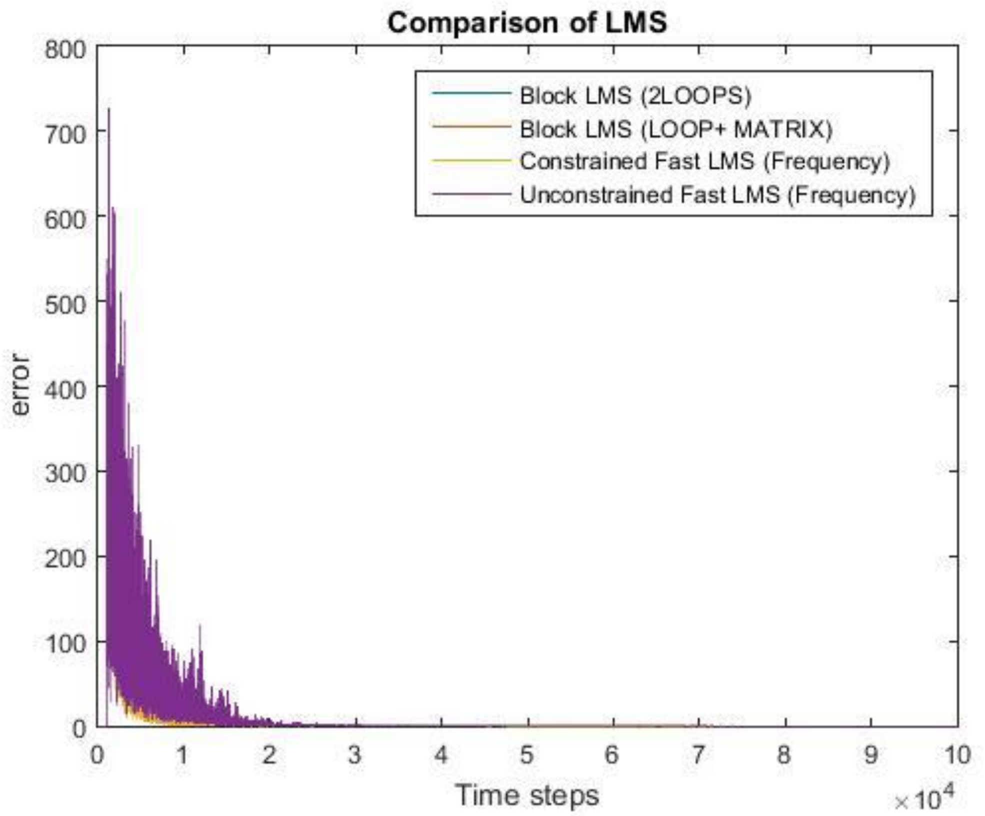
Παρατηρώντας τον αναδρομικό κώδικα για FFT μπορούμε να διαπιστώσουμε ότι η πολυπλοκότητα του είναι η ίδια με αυτή που παρουσιάστηκε παραπάνω.

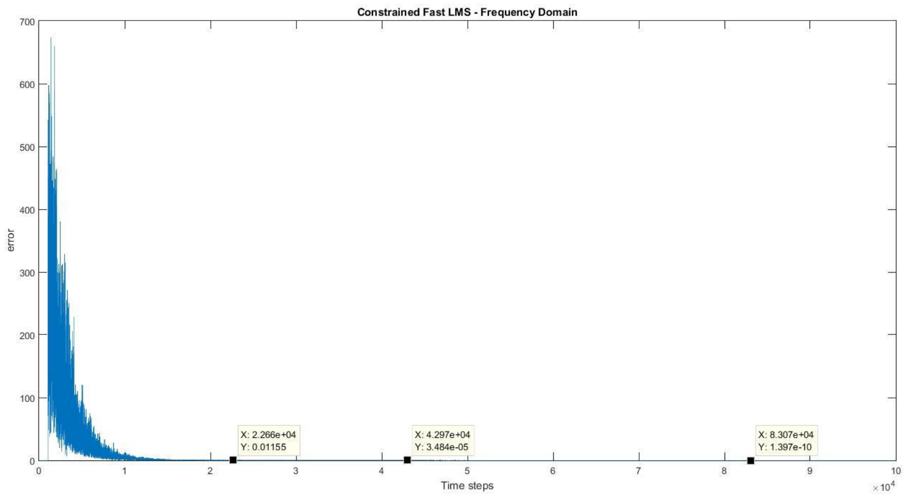
γ) Στο αρχείο c_erg2_conv.m βρίσκονται οι υπολογισμοί της συνέλιξης με τους τρόπους που ζητούνται.

δ) Στο αρχείο lms_tests.m βρίσκονται οι υπολογισμοί του BLOCK LMS με τους τρόπους που ζητούνται.

Συμπεράσματα- Διαγράμματα για τη σύγκλιση των αλγορίθμων Block LMS

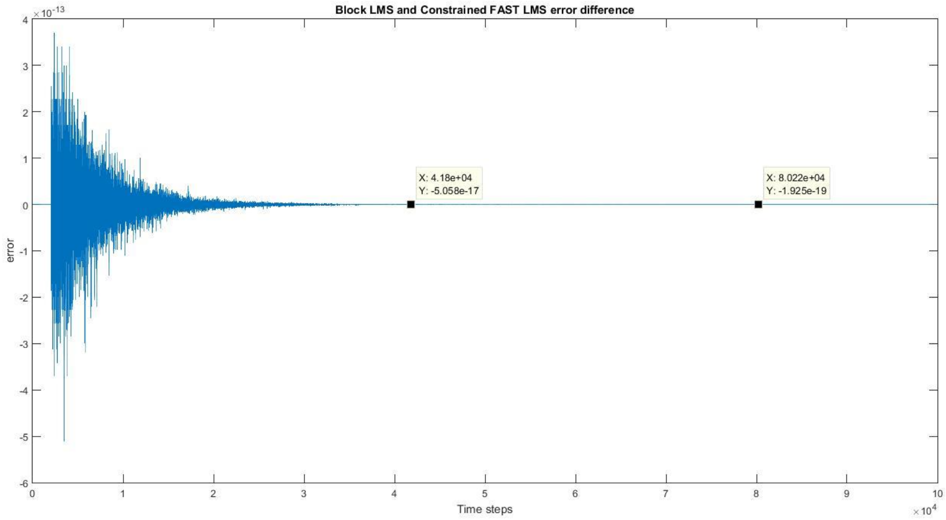
Παρακάτω παρουσιάζονται οι Learning Curves για τις υλοποιήσεις του Block LMS. Στις 3 πρώτες φαίνεται ότι το μέσο τετραγωνικό σφάλμα συγκλίνει στο μηδέν με απόκλιση της τάξεως 10^{-2} για 21000 βήματα και τείνει σε σφάλμα της τάξεως 10^{-12} για μεγαλύτερο αριθμό βημάτων περίπου 90000 βημάτων. Για τον unconstrained η σύγκλιση είναι πιο αργή καθώς για να επιτευχθεί χρειάστηκε να μειωθεί η παράμετρος μ (step size) με αποτέλεσμα να χρειάζεται μεγαλύτερος αριθμός βημάτων για να προσεγγίσει το μηδέν.





Παρακάτω φαίνεται το διάγραμμα της διαφοράς μεταξύ του μέσου τετραγωνικού σφάλματος του BLOCK LMS(2 nested loops) και του Constrained FAST LMS (Frequency Domain). O Constrained Fast LMS έχει σφάλμα μόνιμης κατάστασης της τάξεως του 10^{-20} το οποίο δείχνει στην ουσία μηδενικό σφάλμα πολύ μεγάλης ακρίβειας ενώ για τον ίδιο αριθμό βημάτων ο Unconstrained Fast LMS φτάνει σε σφάλμα της τάξεως 10^{-8} και σε σφάλμα μικρότερης τάξης αντίστοιχης του constrained για 1,5 φορά μεγαλύτερο αριθμό βημάτων(unconstrained->πιο αργή σύγκλιση) . Επίσης το σφάλμα του Constrained FAST LMS σε σχέση με του Block

LMS είναι ελάχιστο και για μικρό αριθμό βημάτων όπως φαίνεται ήδη από τα 23000 βήματα έχουμε ακρίβεια της τάξεως του 10^{-16} .



Όσων αφορά την ταχύτητα των υπολογισμών στη Matlab μετά από δοκιμές προέκυψε ότι οι χρόνοι των υπολογισμών για n=100000 με τους 4 διαφορετικούς αλγορίθμους υπολογισμού του Block LMS είναι:

2 nested loops : 0.98 seconds

Loop + matrix : 1.5 seconds

Constrained Fast LMS Frequency Domain : 0.013 seconds

Unconstrained Fast LMS Frequency Domain : 0.009 seconds

Επομένως ο πιο γρήγορος τρόπος υπολογισμού του Block LMS στη Matlab είναι **φαινομενικά** ο Unconstrained Fast LMS στο πεδίο τη συχνότητας γιατί στην πραγματικότητα πετυχαίνει μικρότερη ακρίβεια από τον constrained για τον ίδιο αριθμό βημάτων. Επομένως ο Constrained Fast LMS συγκλίνει πιο γρήγορα.