



**ΤΕΧΝΟΛΟΓΙΚΟ ΕΚΠΑΙΔΕΥΤΙΚΟ ΙΔΡΥΜΑ ΔΥΤΙΚΗΣ ΕΛΛΑΔΑΣ
ΣΧΟΛΗ ΔΙΟΙΚΗΣΗΣ ΚΑΙ ΟΙΚΟΝΟΜΙΑΣ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΜΜΕ**

**ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ
ΔΗΜΙΟΥΡΓΙΑ ΕΚΠΑΙΔΕΥΤΙΚΟΥ ΥΛΙΚΟΥ
ΓΙΑ ΨΗΦΙΑΚΗ ΕΠΕΞΕΡΓΑΣΙΑ ΣΗΜΑΤΟΣ**

**ΓΕΩΡΓΑΚΟΠΟΥΛΟΥ ΑΙΚΑΤΕΡΙΝΗ
ΠΑΠΑΧΡΗΣΤΟΥ ΠΑΝΑΓΙΩΤΑ**

ΕΠΟΠΤΕΥΩΝ ΚΑΘΗΓΗΤΗΣ: ΚΟΥΤΡΑΣ ΑΘΑΝΑΣΙΟΣ

ΠΥΡΓΟΣ, 2017

ΠΙΣΤΟΠΟΙΗΣΗ

Πιστοποιείται ότι η διπλωματική εργασία με θέμα:

«ΔΗΜΙΟΥΡΓΙΑ ΕΚΠΑΙΔΕΥΤΙΚΟΥ ΥΛΙΚΟΥ ΓΙΑ ΨΗΦΙΑΚΗ ΕΠΕΞΕΡΓΑΣΙΑ ΣΗΜΑΤΟΣ»

του φοιτητή του Τμήματος ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΜΜΕ

ΓΕΩΡΓΑΚΟΠΟΥΛΟΥ ΑΙΚΑΤΕΡΙΝΗ

ΠΑΠΑΧΡΗΣΤΟΥ ΠΑΝΑΓΙΩΤΑ

παρουσιάστηκε δημόσια και εξετάσθηκε στο Τμήμα ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΜΜΕ στις

_____ / _____ / _____

Ο ΕΠΙΒΛΕΠΩΝ

Κούτρας Αθανάσιος

Ο ΠΡΟΕΔΡΟΣ ΤΟΥ ΤΜΗΜΑΤΟΣ

ΟΝΟΜΑΤΕΠΩΝΥΜΟ ΕΠΙΒΛΕΠΟΝΤΑ

Δρ. ΑΘΑΝΑΣΙΟΣ ΚΟΥΤΡΑΣ

ΕΠΙΚ. ΚΑΘΗΓΗΤΗΣ

ΥΠΕΥΘΥΝΗ ΔΗΛΩΣΗ ΠΕΡΙ ΜΗ ΛΟΓΟΚΛΟΠΗΣ

Βεβαιώνω ότι είμαι συγγραφέας αυτής της εργασίας και ότι κάθε βοήθεια την οποία είχα για την προετοιμασία της, είναι πλήρως αναγνωρισμένη και αναφέρεται στην εργασία. Επίσης, έχω αναφέρει τις όποιες πηγές από τις οποίες έκανα χρήση δεδομένων, ιδεών ή λέξεων, είτε αυτές αναφέρονται ακριβώς είτε παραφρασμένες. Ακόμα δηλώνω ότι αυτή η γραπτή εργασία προετοιμάστηκε από εμένα προσωπικά και αποκλειστικά και ειδικά για την συγκεκριμένη πτυχιακή εργασία και ότι θα αναλάβω πλήρως τις συνέπειες εάν η εργασία αυτή αποδειχθεί ότι δεν μου ανήκει.

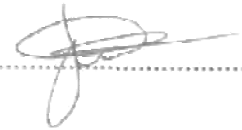
ΟΝΟΜΑΤΕΠΩΝΥΜΟ ΣΠΟΥΔΑΣΤΗ 1

ΑΜ

ΥΠΟΓΡΑΦΗ

ΠΑΠΑΧΡΗΣΤΟΥ ΠΑΝΑΓΟΤΑ.....

943.....



ΟΝΟΜΑΤΕΠΩΝΥΜΟ ΣΠΟΥΔΑΣΤΗ 2

ΑΜ

ΥΠΟΓΡΑΦΗ

(σε περίπτωση που είναι απαραίτητο)

ΓΕΩΡΓΙΑΚΟΠΟΥΛΟΥ ΑΙΚΑΤΕΡΙΝΗ.....

883.....



ΟΝΟΜΑΤΕΠΩΝΥΜΟ ΣΠΟΥΔΑΣΤΗ 3

ΑΜ

ΥΠΟΓΡΑΦΗ

(σε περίπτωση που είναι απαραίτητο)

.....

.....

.....

Περίληψη

Η ανάπτυξη του επιστημονικού κλάδου της ψηφιακής επεξεργασίας σήματος οφείλεται κυρίως στη ραγδαία εξέλιξη των υπολογιστών. Η ψηφιακή επεξεργασία σήματος βρίσκει εφαρμογές σε πολλούς σύγχρονους επιστημονικούς κλάδους, συνεπώς είναι απαραίτητο οι επιστήμονες των κλάδων αυτών να εκπαιδευτούν για να αποκτήσουν εμπειρία στις τεχνικές ψηφιακής επεξεργασίας σήματος.

Στο πλαίσιο της παρούσας εργασίας χρησιμοποιήθηκαν συγκεκριμένες προδιαγραφές ποιότητας για την ανάπτυξη υποστηρικτικού υλικού, που αφορά τη διδασκαλία του εργαστηριακού μαθήματος «Ψηφιακή Επεξεργασία Σήματος», καλύπτοντας 13 διδακτικές ενότητες. Για κάθε ενότητα αναπτύχθηκε μία εργαστηριακή άσκηση ως έντυπο φύλλο εργασίας με ορισμένους μαθησιακούς στόχους. Επίσης, για κάθε ενότητα αναπτύχθηκε μία διαδραστική διεπαφή σε Matlab (ψηφιακό υλικό), που μπορεί να χρησιμοποιηθεί για την εκπόνηση της αντίστοιχης εργαστηριακής άσκησης χρησιμοποιώντας το σχετικό φύλλο εργασίας.

Ο βασικός σκοπός ανάπτυξης του υποστηρικτικού υλικού είναι η πιο ευχάριστη και ξεκούραστη παρουσίαση της ύλης, επιτρέποντας την ευκολότερη κατανόηση και την εμβάθυνση στις βασικές έννοιες και τεχνικές της ψηφιακής επεξεργασίας σήματος. Οι δομημένες εργαστηριακές ασκήσεις, καθώς και το αντίστοιχο διαδραστικό λογισμικό ασκήσεων σε Matlab, προάγουν την ενεργητική και ανακαλυπτική μάθηση, αφού επιτρέπουν στον χρήστη να πειραματιστεί με το λογισμικό εισάγοντας διαφορετικά δεδομένα κάθε φορά και μελετώντας τα αποτελέσματα της εκτέλεσης, τα οποία εμφανίζονται με χρήση πινάκων και γραφικών παραστάσεων.

Βασική καινοτομία της παρούσας εργασίας είναι η υποστήριξη της κάθε διδακτικής ενότητας από μία ξεχωριστή διεπαφή σε Matlab, παρέχοντας στον χρήστη τη δυνατότητα να πειραματιστεί είτε με τα έτοιμα πρότυπα φύλλα εργασίας είτε με δικά του δεδομένα. Ωστόσο, τα φύλλα εργασίας και οι διεπαφές Matlab μπορούν να χρησιμοποιηθούν και ανεξάρτητα μεταξύ τους.

Περιεχόμενα

| | |
|--|-----------|
| Περίληψη..... | V |
| Περιεχόμενα | i |
| Κατάλογος Εικόνων | v |
| Κατάλογος Σχημάτων..... | vii |
| Κατάλογος Πινάκων..... | ix |
| Εισαγωγή | xi |
| 1 Ψηφιακή Επεξεργασία Σήματος..... | 1 |
| 1.1 Τι πραγματεύεται | 1 |
| 1.2 Εφαρμογές..... | 2 |
| 1.3 Διδασκαλία της Ψηφιακής Επεξεργασίας Σήματος..... | 4 |
| 2 Εκπαιδευτικό Υλικό | 7 |
| 2.1 Μορφές Εκπαιδευτικού Υλικού | 7 |
| 2.2 Προδιαγραφές Εκπαιδευτικού Υλικού | 8 |
| 2.3 Έντυπο Εκπαιδευτικό Υλικό | 10 |
| 2.3.1 Προδιαγραφές Έντυπου Εκπαιδευτικού Υλικού | 10 |
| 2.3.2 Προδιαγραφές Φύλλων Εργασίας | 13 |
| 2.4 Ψηφιακό Εκπαιδευτικό Υλικό | 16 |
| 2.4.1 Προδιαγραφές Ψηφιακού Εκπαιδευτικού Υλικού..... | 19 |
| 2.4.1.1 Ψηφιακό Περιεχόμενο | 19 |
| 2.4.1.2 Τεχνολογικό Μέσο | 22 |
| 2.4.1.3 Παιδαγωγική/Διδακτική Εφαρμογή..... | 22 |
| 2.4.2 Προδιαγραφές Διαδραστικού Λογισμικού Ασκήσεων..... | 23 |
| 2.5 Προσδοκώμενα Αποτελέσματα και Μαθησιακοί Στόχοι..... | 25 |
| 3 MATLAB..... | 30 |
| 3.1 Βασικά Χαρακτηριστικά..... | 30 |

| | | |
|-----|---|-----|
| 3.2 | Το Περιβάλλον <i>GUIDE</i> | 31 |
| 4 | Σχεδίαση και Ανάπτυξη του Υποστηρικτικού Υλικού | 41 |
| 4.1 | Περιγραμμά Υλης..... | 41 |
| 4.2 | Έντυπο Υλικό – Φύλλα Εργασίας Εργαστηριακών Ασκήσεων | 43 |
| | Εργαστηριακή Άσκηση 1: Εισαγωγή στο Matlab | 45 |
| | Εργαστηριακή Άσκηση 2: Δημιουργία Βασικών Σημάτων Διακριτού Χρόνου – Βασικές Πράξεις | 88 |
| | Εργαστηριακή Άσκηση 3: Μετασχηματισμοί της Ανεξάρτητης Μεταβλητής – Χαρακτηριστικά Μεγέθη ΣΔΧ | 105 |
| | Εργαστηριακή Άσκηση 4: Ιδιότητες Συστημάτων | 121 |
| | Εργαστηριακή Άσκηση 5: Γραμμική Συνέλιξη | 137 |
| | Εργαστηριακή Άσκηση 6: Εξισώσεις Διαφορών | 174 |
| | Εργαστηριακή Άσκηση 7: Μετασχηματισμός Fourier Διακριτού Χρόνου - DTFT | 189 |
| | Εργαστηριακή Άσκηση 8: Απόκριση Συχνότητας..... | 213 |
| | Εργαστηριακή Άσκηση 9: Διακριτός Μετασχηματισμός Fourier - DFT | 225 |
| | Εργαστηριακή Άσκηση 10: Μετασχηματισμός Z | 243 |
| | Εργαστηριακή Άσκηση 11: Συνάρτηση Μεταφοράς Συστήματος..... | 256 |
| | Εργαστηριακή Άσκηση 12: Φίλτρα | 267 |
| | Εργαστηριακή Άσκηση 13: Δειγματοληψία | 282 |
| 4.3 | Ψηφιακό Υλικό – Λογισμικό Ασκήσεων | 291 |
| | Εργαστηριακή Άσκηση 1: Εισαγωγή στο Matlab | 293 |
| | Εργαστηριακή Άσκηση 2: Δημιουργία Βασικών Σημάτων Διακριτού Χρόνου – Βασικές Πράξεις | 294 |
| | Εργαστηριακή Άσκηση 3: Μετασχηματισμοί της Ανεξάρτητης Μεταβλητής – Χαρακτηριστικά Μεγέθη ΣΔΧ | 294 |
| | Εργαστηριακή Άσκηση 4: Ιδιότητες Συστημάτων | 295 |
| | Εργαστηριακή Άσκηση 5: Γραμμική Συνέλιξη | 296 |
| | Εργαστηριακή Άσκηση 6: Εξισώσεις Διαφορών | 297 |
| | Εργαστηριακή Άσκηση 7: Μετασχηματισμός Fourier Διακριτού Χρόνου – DTFT..... | 298 |

| | |
|--|------------|
| Εργαστηριακή Άσκηση 8: Απόκριση Συχνότητας | 299 |
| Εργαστηριακή Άσκηση 9: Διακριτός Μετασχηματισμός Fourier – DFT | 300 |
| Εργαστηριακή Άσκηση 10: Μετασχηματισμός Z | 301 |
| Εργαστηριακή Άσκηση 11: Συνάρτηση Μεταφοράς | 302 |
| Εργαστηριακή Άσκηση 12: Φίλτρα | 303 |
| Εργαστηριακή Άσκηση 13: Δειγματοληψία | 304 |
| 5 Συμπεράσματα | 305 |
| 6 Προοπτικές – Μελλοντικές Βελτιώσεις | 307 |
| Βιβλιογραφία | 308 |

Κατάλογος Εικόνων

| | |
|--|-----|
| Εικόνα 1: Επίπεδα μαθησιακών στόχων γνωστικού τομέα σύμφωνα με τον Bloom (1956)..... | 29 |
| Εικόνα 2: Οδηγός δημιουργίας παραθύρου | 32 |
| Εικόνα 3: Περιβάλλον σχεδίασης νέου παραθύρου του GUIDE | 33 |
| Εικόνα 4: Παράθυρο αποθήκευσης εφαρμογής GUIDE | 34 |
| Εικόνα 5: Γραμμή εργαλείων διαχείρισης του GUIDE | 36 |
| Εικόνα 6: Παράθυρο Align Objects του GUIDE | 38 |
| Εικόνα 7: Παράθυρο Menu Editor του GUIDE | 38 |
| Εικόνα 8: Παράθυρο Tab Order Editor του GUIDE | 39 |
| Εικόνα 9: Παράθυρο Toolbar Editor του GUIDE | 39 |
| Εικόνα 10: Παράθυρο Object Browser του GUIDE | 39 |
| Εικόνα 11: Παράθυρο Editor του GUIDE | 40 |
| Εικόνα 12: Παράθυρο Property Inspector του GUIDE | 40 |
| Εικόνα 13: Κεντρική οθόνη λογισμικού ασκήσεων | 293 |
| Εικόνα 14: Μήνυμα 1 ^{ης} εργαστηριακής άσκησης | 293 |
| Εικόνα 15: Διεπαφή 2 ^{ης} εργαστηριακής άσκησης | 294 |
| Εικόνα 16: Διεπαφή 3 ^{ης} εργαστηριακής άσκησης | 295 |
| Εικόνα 17: Διεπαφή 4 ^{ης} εργαστηριακής άσκησης | 296 |
| Εικόνα 18: Διεπαφή 5 ^{ης} εργαστηριακής άσκησης | 297 |
| Εικόνα 19: Διεπαφή 6 ^{ης} εργαστηριακής άσκησης | 298 |
| Εικόνα 20: Διεπαφή 7 ^{ης} εργαστηριακής άσκησης | 299 |
| Εικόνα 21: Διεπαφή 8 ^{ης} εργαστηριακής άσκησης | 300 |
| Εικόνα 22: Διεπαφή 9 ^{ης} εργαστηριακής άσκησης | 301 |
| Εικόνα 23: Διεπαφή 10 ^{ης} εργαστηριακής άσκησης | 302 |
| Εικόνα 24: Διεπαφή 11 ^{ης} εργαστηριακής άσκησης | 303 |
| Εικόνα 25: Διεπαφή 12 ^{ης} εργαστηριακής άσκησης | 303 |
| Εικόνα 26: Διεπαφή 13 ^{ης} εργαστηριακής άσκησης | 304 |

Κατάλογος Σχημάτων

| | |
|---|-----|
| Σχήμα 1: Αναπαράσταση συστήματος..... | 140 |
| Σχήμα 2: (Αριστερά) Διασύνδεση δύο συστημάτων σε σειρά και (δεξιά) η ισοδύναμη συνολική διάταξη..... | 156 |
| Σχήμα 3: (Αριστερά) Παράλληλη διασύνδεση δύο συστημάτων και (δεξιά) η ισοδύναμη συνολική διάταξη..... | 162 |

Κατάλογος Πινάκων

| | |
|--|-----|
| Πίνακας 1: Αντιστοιχία ρημάτων σε επίπεδα μαθησιακών στόχων Bloom (1956) | 27 |
| Πίνακας 2: Στοιχεία ελέγχου της εργαλειοθήκης του GUIDE | 34 |
| Πίνακας 3: Στοιχεία γραμμής εργαλείων διαχείρισης του GUIDE..... | 36 |
| Πίνακας 4: Βασικότερα ζεύγη μετασχηματισμού Z | 246 |

Εισαγωγή

Η ανάπτυξη του επιστημονικού κλάδου της ψηφιακής επεξεργασίας σήματος οφείλεται κυρίως στη ραγδαία εξέλιξη των υπολογιστών. Η ψηφιακή επεξεργασία σήματος βρίσκει εφαρμογές σε πολλούς σύγχρονους επιστημονικούς κλάδους, όπως της ιατρικής, της πληροφορικής, της φυσικής κ.ο.κ, συνεπώς θεωρείται απαραίτητο οι επιστήμονες των κλάδων αυτών να εκπαιδευτούν στο αντικείμενο της ψηφιακής επεξεργασίας σήματος για να αποκτήσουν καλή θεωρητική, αλλά και πρακτική γνώση των τεχνικών της και των τρόπων αξιοποίησής τους. Για το σκοπό αυτόν, έχει αναπτυχθεί πληθώρα εργαλείων και εκπαιδευτικών λογισμικών – εφαρμογών, που στοχεύουν στην εκπαίδευση μέσω υπολογιστή των βασικών εννοιών, τεχνικών και εφαρμογών της ψηφιακής επεξεργασίας σήματος. Αρωγός στη διδασκαλία της ψηφιακής επεξεργασίας σήματος, εκτός από το βασικό εγχειρίδιο διδασκαλίας που παρέχεται στους φοιτητές, είναι και το καλά σχεδιασμένο υποστηρικτικό εκπαιδευτικό υλικό, το οποίο συναντάται σε έντυπη και ψηφιακή μορφή, συνήθως ως διαδραστικό λογισμικό ασκήσεων που συνοδεύεται από κατάλληλα φύλλα εργασίας με συγκεκριμένους μαθησιακούς στόχους.

Το Matlab αποτελεί ένα από τα δημοφιλέστερα προγραμματιστικά περιβάλλοντα που χρησιμοποιούνται για την ανάπτυξη αλληλεπιδραστικών εκπαιδευτικών εφαρμογών στην εκμάθηση εννοιών και τεχνικών της ψηφιακής επεξεργασίας σήματος. Οι εφαρμογές αυτές είναι εύκολα επεκτάσιμες, λόγω της ανοικτής τους αρχιτεκτονικής, διαθέτουν εύχρηστες και φιλικές προς τον εκπαιδευόμενο γραφικές διεπαφές, παρέχουν άμεση οπτική ανατροφοδότηση και γενικά ενδείκνυνται για μάθηση μέσω προσομοιώσεων, πειραματισμού και παρατήρησης.

Στη συνέχεια ακολουθεί μία σύντομη περιγραφή της δομής της εργασίας.

Στην πρώτη και στη δεύτερη ενότητα παρουσιάζεται το θεωρητικό πλαίσιο της εργασίας, ενώ στις επόμενες ενότητες ακολουθεί το πρακτικό μέρος της εργασίας.

Πιο συγκεκριμένα, στην πρώτη ενότητα γίνεται σύντομη αναφορά στον επιστημονικό κλάδο της ψηφιακής επεξεργασίας σήματος, των ζητημάτων που

πραγματεύεται και των πεδίων εφαρμογής της, καθώς και στη διδασκαλία του μαθήματος της ψηφιακής επεξεργασίας σήματος στην τριτοβάθμια εκπαίδευση.

Στη δεύτερη ενότητα, αρχικά παρουσιάζονται οι μορφές και οι προδιαγραφές που πρέπει να διαθέτει το εκπαιδευτικό υλικό ώστε να θεωρείται ποιοτικό και αποτελεσματικό. Ακολούθως, παρουσιάζονται οι προδιαγραφές του έντυπου εκπαιδευτικού υλικού με εστίαση στις προδιαγραφές των φύλλων εργασίας, καθώς και οι προδιαγραφές του ψηφιακού εκπαιδευτικού υλικού με εστίαση στις προδιαγραφές του διαδραστικού λογισμικού ασκήσεων. Επίσης, γίνεται αναφορά στους μαθησιακούς στόχους και στα προσδοκώμενα αποτελέσματα, δεδομένου ότι αποτελούν το βασικό δομικό στοιχείο όλων των μαθησιακών δραστηριοτήτων.

Στην τρίτη ενότητα γίνεται αναφορά στο Matlab και παρουσιάζεται συνοπτικά το περιβάλλον δημιουργίας γραφικών διεπαφών GUIDE του Matlab. Τα εργαλεία αυτά χρησιμοποιήθηκαν για την ανάπτυξη εκπαιδευτικού διαδραστικού λογισμικού ασκήσεων, το οποίο σε συνδυασμό με κατάλληλα σχεδιασμένα φύλλα εργασίας στοχεύουν στην υποστήριξη της εργαστηριακής διδασκαλίας του μαθήματος της ψηφιακής επεξεργασίας σήματος. Η ανάπτυξη του υποστηρικτικού υλικού διδασκαλίας (διαδραστικό λογισμικό ασκήσεων και φύλλα εργασίας) παρουσιάζεται αναλυτικά στην ενότητα 4.

Στην τέταρτη ενότητα παρουσιάζεται αναλυτικά η δημιουργία του υποστηρικτικού υλικού για την ψηφιακή επεξεργασία σήματος. Πιο συγκεκριμένα, παρουσιάζεται το περίγραμμα της διδακτέας ύλης του εργαστηρίου της ψηφιακής επεξεργασίας σήματος, το οποίο χρησιμοποιήθηκε ως βάση για τη δημιουργία μίας σειράς πρότυπων φύλλων εργασιών με κατάλληλες μαθησιακές δραστηριότητες, καθώς και ενός διαδραστικού λογισμικού ασκήσεων (εκπαιδευτικής εφαρμογής) σε Matlab, για την εκπόνηση – εκτέλεση των δραστηριοτήτων των φύλλων εργασίας.

Στην πέμπτη ενότητα παρουσιάζονται τα συμπεράσματα και στην έκτη ενότητα οι προοπτικές – μελλοντικές βελτιώσεις του υλικού της παρούσας εργασίας.

1 Ψηφιακή Επεξεργασία Σήματος

1.1 Τι πραγματεύεται

Η ψηφιακή επεξεργασία σήματος (ΨΕΣ) ασχολείται με την αναπαράσταση σημάτων διακριτού χρόνου ως ακολουθιών αριθμών ή συμβόλων, καθώς και με την επεξεργασία των σημάτων αυτών. Η ψηφιακή και η αναλογική επεξεργασία σήματος συναποτελούν το διεπιστημονικό γνωστικό πεδίο των εφαρμοσμένων μαθηματικών, γνωστό ως επεξεργασία σήματος.

Ο σκοπός της ψηφιακής επεξεργασίας σήματος συνήθως είναι η μέτρηση, το φιλτράρισμα ή η συμπίεση συνεχόμενων αναλογικών φυσικών σημάτων. Το πρώτο βήμα συνήθως είναι η μετατροπή του σήματος από αναλογικό σε ψηφιακό, μέσω της δειγματοληψίας και της κβάντισης σήματος με τη βοήθεια ενός μετατροπέα αναλογικού σήματος σε ψηφιακό (ADC), ο οποίος μετασχηματίζει το αναλογικό σήμα σε μια ακολουθία από αριθμούς. Συχνά όμως το ζητούμενο σήμα εξόδου είναι επίσης αναλογικό, παρόλο που η επεξεργασία έχει ψηφιακό χαρακτήρα, επομένως χρειάζεται και ένας μετατροπέας ψηφιακού σήματος σε αναλογικό (DAC). Αν και αυτή η διαδικασία μπορεί να είναι πιο πολύπλοκη σε σχέση με την αναλογική επεξεργασία και έχει διακριτό πεδίο τιμών, η χρήση υπολογιστικής ισχύος στην ψηφιακή επεξεργασία σήματος έχει πολλά πλεονεκτήματα σε σχέση με την αναλογική επεξεργασία σήματος σε πολλές εφαρμογές, όπως ο εντοπισμός και η διόρθωση λαθών στις επικοινωνίες και η συμπίεση δεδομένων (Broesch, Stranneby, & Walker, 2008).

Οι αλγόριθμοι ΨΕΣ εκτελούνται από παλιά σε υπολογιστές, σε εξειδικευμένους επεξεργαστές που ονομάζονται ψηφιακοί επεξεργαστές σήματος σε υλικό που κατασκευάζεται ειδικά για αυτόν τον σκοπό, όπως τα ολοκληρωμένα κυκλώματα εφαρμογών. Σήμερα χρησιμοποιούνται και άλλες τεχνολογίες για την ψηφιακή επεξεργασία σήματος, όπως οι ισχυροί μικροεπεξεργαστές γενικής χρήσης, οι συστοιχίες επιτόπια προγραμματιζόμενων πυλών (FPGA), οι ψηφιακοί ελεγκτές

σήματος (κυρίως σε εφαρμογές στη βιομηχανία, όπως ο έλεγχος κινητήρων) και οι επεξεργαστές ροής (stream processors) (Stranneby & Walker, 2004).

1.2 Εφαρμογές

Στις βασικές εφαρμογές της ψηφιακής επεξεργασίας σήματος συγκαταλέγονται:

- η επεξεργασία ηχητικών σημάτων,
- η συμπίεση ήχου,
- η ψηφιακή επεξεργασία εικόνας,
- η συμπίεση βίντεο,
- η επεξεργασία φωνής,
- η αναγνώριση φωνής,
- οι ψηφιακές τηλεπικοινωνίες,
- τα ραντάρ,
- τα σόναρ,
- η σεισμολογία και
- η βιοϊατρική (Βικιπαίδεια)

Μερικά πιο συγκεκριμένα παραδείγματα είναι:

- η συμπίεση φωνής και η μετάδοση σε ψηφιακά κινητά τηλέφωνα,
- η ψηφιακή διόρθωση δωματίου του ήχου στα ηχοσυστήματα και σε εφαρμογές ενίσχυσης ήχου,
- η πρόγνωση του καιρού,
- οι οικονομικές προβλέψεις,
- η επεξεργασία δεδομένων από σεισμούς,
- η ανάλυση και ο έλεγχος βιομηχανικών διεργασιών,
- η ιατρική απεικόνιση, όπως οι αξονικές και οι μαγνητικές τομογραφίες,
- η συμπίεση MP3,
- τα γραφικά υπολογιστών,
- η επεξεργασία εικόνας,

- τα φίλτρα crossover και οι ισοσταθμιστές (equalizers) των ηχείων hi-fi, και τα ηχητικά εφέ που χρησιμοποιούνται μαζί με τους ενισχυτές ηλεκτρικής κιθάρας (Βικιπαίδεια)

Ανάλογα με τις απαιτήσεις της εφαρμογής, οι λειτουργίες της ψηφιακής επεξεργασίας σήματος μπορούν να υλοποιηθούν σε υπολογιστές γενικής χρήσης ή σε υπολογιστές με ενσωματωμένους εξειδικευμένους μικροεπεξεργαστές, που ονομάζονται ψηφιακοί επεξεργαστές σήματος (digital signal processors).

Όταν η επεξεργασία δεν χρειάζεται να γίνεται σε πραγματικό χρόνο, μπορεί να γίνει για λόγους οικονομίας με κάποιον υπολογιστή γενικής χρήσης και με τα σήματα εισόδου/εξόδου να βρίσκονται σε αρχεία, κάτι που δε διαφέρει πολύ από άλλες εφαρμογές επεξεργασίας δεδομένων. Χρησιμοποιούνται μαθηματικές τεχνικές ψηφιακής επεξεργασίας σήματος (όπως ο γρήγορος μετασχηματισμός Φουριέ – FFT) και τα δείγματα-δεδομένα θεωρείται ότι είναι κατανεμημένα ομοιόμορφα στον χρόνο και στον χώρο. Ένα τέτοιο παράδειγμα είναι η επεξεργασία ψηφιακών φωτογραφιών με λογισμικό, όπως το Photoshop.

Όταν όμως η εφαρμογή απαιτεί απόκριση σε πραγματικό χρόνο, η ψηφιακή επεξεργασία σήματος συνήθως υλοποιείται χρησιμοποιώντας εξειδικευμένους επεξεργαστές, όπως είναι οι DSP56000, TMS320 ή SHARC. Αυτοί συνήθως επεξεργάζονται τα δεδομένα χρησιμοποιώντας αριθμητική σταθερής υποδιαστολής (αν και κάποιες εκδόσεις τους μπορούν να χρησιμοποιήσουν αριθμητική κινητής υποδιαστολής) και είναι πιο ισχυροί. Για πιο γρήγορες εφαρμογές μπορούν να χρησιμοποιηθούν FPGA (JpFix). Από το 2007 έχουν αρχίσει να εμφανίζονται πολυπύρηνες υλοποιήσεις ψηφιακών επεξεργαστών σήματος από εταιρείες, όπως η Freescale και η Stream Processors. Για ακόμα πιο γρήγορες εφαρμογές με πάρα πολλούς χρήστες μπορούν να σχεδιαστούν ολοκληρωμένα κυκλώματα εφαρμογών κατά περίπτωση, ενώ για πιο αργές εφαρμογές μπορεί να αρκεί ένας πιο αργός επεξεργαστής, όπως ένας μικροελεγκτής. Πολλές εφαρμογές ψηφιακής επεξεργασίας σήματος υλοποιούνται σε ενσωματωμένα συστήματα

χρησιμοποιώντας ισχυρούς προσωπικούς υπολογιστές με πολυπύρηνους επεξεργαστές (Βικιπαίδεια).

1.3 Διδασκαλία της Ψηφιακής Επεξεργασίας Σήματος

Το μάθημα της Ψηφιακής Επεξεργασίας Σήματος συνήθως αποτελεί υποχρεωτικό μάθημα των προπτυχιακών προγραμμάτων σπουδών σε τμήματα πανεπιστημίων και τεχνολογικών ιδρυμάτων, τα οποία σχετίζονται με το αντικείμενο της Πληροφορικής.

Αποτελείται από θεωρητικό και εργαστηριακό μέρος, με το εργαστηριακό μέρος να δίνει έμφαση στην χρήση ειδικών λογισμικών (λχ Matlab), που προάγουν την καλύτερη κατανόηση και την εμβάθυνση στα ζητήματα της ψηφιακής επεξεργασίας σήματος.

Με την ολοκλήρωση του μαθήματος από τους φοιτητές/τριες, στα συνηθέστερα αναμενόμενα μαθησιακά αποτελέσματα συγκαταλέγονται:

- Η εμπειριστατωμένη γνώση και κριτική κατανόηση της θεωρίας ψηφιακού σήματος και της σχέσης του με το αναλογικό σήμα
- Γνώση και δεξιότητες στα μαθηματικά εργαλεία που βοηθούν στην αναλυτική μελέτη και επεξεργασία ψηφιακών σημάτων αλλά και στην κατανόηση των περιορισμών ή προβλημάτων που ανακύπτουν κατά την επεξεργασία σημάτων στο ψηφιακό πεδίο
- Γνώση και ικανότητες σχεδιασμού, σύνθεσης, προγραμματισμού, εκσφαλμάτωσης και επιδιόρθωσης λειτουργίας ψηφιακού συστήματος με χρήση υπολογιστή

Ακολουθεί ένα ενδεικτικό περίγραμμα των συνηθέστερων ζητημάτων που καλύπτονται στο πλαίσιο της διδασκαλίας της Ψηφιακής Επεξεργασίας Σήματος:

- Αναλογικά σήματα, Μετασχηματισμός Fourier (FT), σειρά Fourier.
- Συστήματα LTI συνεχούς χρόνου.

- Σήματα διακριτού χρόνου άπειρης χρονικής έκτασης, Μετασχηματισμός Fourier Διακριτού Χρόνου (DTFT)
- Συστήματα LTI διακριτού χρόνου, γραμμική συνέλιξη,
- Δειγματοληψία, Θεώρημα Nyquist και επικάλυψη, μετατροπείς A/D, σχέση με το αναλογικό πεδίο
- Ψηφιακά σήματα πεπερασμένης έκτασης, Διακριτός Μετασχηματισμός Fourier (DFT), σχέση με FT και DTFT, διακριτότητα DFT και επιλογή παραθύρου, κυκλική συνέλιξη και σχέση με τη γραμμική
- Εξισώσεις διαφορών, Μετασχηματισμός Z, εισαγωγή στα FIR και IIR φίλτρα
- Ανάλυση ψηφιακών σημάτων με χρήση ηλεκτρονικού υπολογιστή
- Σχεδίαση και υλοποίηση συστήματος ψηφιακής επεξεργασίας σήματος τη χρήση ηλεκτρονικού υπολογιστή

Για το θεωρητικό μέρος του μαθήματος, η εκπαιδευτική διαδικασία πραγματοποιείται συνήθως με δια ζώσης διαλέξεις, ενώ για το εργαστηριακό μέρος πραγματοποιείται με την εκπόνηση εργαστηριακών ασκήσεων με χρήση ειδικού λογισμικού, όπως το Matlab.

Ως κύριο εκπαιδευτικό υλικό χρησιμοποιείται το βιβλίο – σύγγραμμα που παρέχει το ίδιο το ίδρυμα στους φοιτητές/τριες του, αλλά συνηθίζεται ο εκπαιδευτής να παρέχει υποστηρικτικό υλικό κατά τη διδασκαλία του μαθήματος, το οποίο μπορεί να συμπεριλαμβάνει:

- Χρήση διαφανειών με πολυμεσικό υλικό κατά την διδασκαλία στην τάξη,
- Εξειδικευμένο λογισμικό προσομοίωσης ψηφιακών διεργασιών στην εργαστηριακή εκπαίδευση,
- Εξειδικευμένο λογισμικό προγραμματισμού και εκτέλεσης ψηφιακών διεργασιών σε hardware ειδικού σκοπού στην εργαστηριακή εκπαίδευση,
- βοηθητικό υλικό μελέτης, ασκήσεις, λυμένα θέματα εξετάσεων, σημειώσεις και φυλλάδια εργαστηρίου

Επίσης, συνηθίζεται πλέον η μαθησιακή διαδικασία να υποστηρίζεται από ηλεκτρονικές πλατφόρμες και συστήματα διαχείρισης μάθησης, όπως είναι το Moodle, το eClass κοκ. Εκεί αναρτώνται ανακοινώσεις, καθώς και το εκπαιδευτικό υλικό του μαθήματος, όπως οι παρουσιάσεις των διαλέξεων, τα φύλλα εργασίας με τις ασκήσεις, θέματα εξετάσεων κοκ, τα οποία συνήθως είναι οργανωμένα σε εβδομαδιαίες ενότητες. Επίσης, οι πλατφόρμες αυτές παρέχουν εργαλεία σύγχρονης και ασύγχρονης επικοινωνίας μεταξύ εκπαιδευομένων και εκπαιδευομένων – εκπαιδευτή, όπως χώρους συζητήσεων forums, ανταλλαγής μηνυμάτων κοκ, προκειμένου να ενισχύσουν την συνεργατική και πιο ενεργητική μάθηση.

2 Εκπαιδευτικό Υλικό

Η ποιότητα της παρεχόμενης εκπαίδευσης σε όλες τις βαθμίδες εκπαίδευσης βασίζεται σε μεγάλο βαθμό στην ποιότητα του εκπαιδευτικού υλικού, καθώς και στην ποιότητα της επικοινωνίας (με την ευρεία έννοια) μεταξύ του εκπαιδευόμενου από την μια μεριά και του εκπαιδευτή και του εκπαιδευτικού οργανισμού από την άλλη. Ειδικότερα, για την τριτοβάθμια και την άτυπη εκπαίδευση, το εκπαιδευτικό υλικό σχεδιάζεται με γνώμονα τη βασική αρχή ότι η μάθηση αποτελεί αλληλεπίδραση του εκπαιδευόμενου με το μαθησιακό υλικό.

2.1 Μορφές Εκπαιδευτικού Υλικού

Συχνά χρησιμοποιείται ο όρος εκπαιδευτικό «πακέτο» για να παραστήσει το σύνολο των μορφών του εκπαιδευτικού υλικού (Ματραλής, 1998, 1999). Έτσι, εκτός του έντυπου υλικού, μπορούν να χρησιμοποιηθεί οπτικοακουστικό υλικό και λογισμικό (εκπαιδευτικά προγράμματα).

Το έντυπο εκπαιδευτικό υλικό συναντάται στις εξής μορφές:

- Βιβλία και εγχειρίδια
- Οδηγοί μελέτης
- Φύλλα εργασίας
- Χάρτες
- Άρθρα από εφημερίδες και περιοδικά

Το οπτικοακουστικό/λογισμικό εκπαιδευτικό υλικό συναντάται στις εξής μορφές:

- Ηχογραφημένες κασέτες ήχου, δίσκοι, CDs
- Ραδιοφωνικές εκπομπές
- Slides ή φωτογραφικά φιλμ
- Φιλμ ταινιών ή αποσπάσματά τους
- Video Tapes
- Τηλεοπτικές εκπομπές
- Εκπαιδευτικά προγράμματα με ηλεκτρονικό υπολογιστή

- Εικονοδιάσκεψη
- Εφαρμογές σε υπολογιστή, αλληλεπιδραστικά προγράμματα, εφαρμογές πολυμέσων, εφαρμογές μέσω Διαδικτύου
- Wikis

Η εξέλιξη των επικοινωνιακών τεχνολογιών, η μείωση του κόστους των υπολογιστών και των τηλεπικοινωνιών, η εξοικείωση ολοένα και μεγαλύτερου μέρους του πληθυσμού με τους υπολογιστές και το Διαδίκτυο συμβάλλουν καθοριστικά σε αυτό. Οι δικτυακές τεχνολογίες και ειδικότερα το Διαδίκτυο μπορούν να χρησιμοποιηθούν για να στηρίξουν τη διδασκαλία και τη μάθηση, σύμφωνα με τις σύγχρονες κοινωνικές και εποικοδομιστικές θεωρίες για την κατάκτηση της γνώσης (Κορδάκη & Λάσκαρης, 2003).

Επίσης, πρέπει πάντα να λαμβάνεται υπόψη, κατά τη δημιουργία του υλικού, ότι οποιαδήποτε κι αν είναι η μορφή του, αυτό πρέπει να δίνει την ευκαιρία στους εκπαιδευόμενους να μάθουν μόνοι τους (Race, 1999), (Γκιάσος & Κουτσούμπα, 2003).

Τέλος, οποιοδήποτε τεχνολογικό ή επικοινωνιακό μέσο κι αν χρησιμοποιηθεί πρέπει να εντάσσεται σε μια παιδαγωγική διαδικασία και να ακολουθεί συγκεκριμένες παιδαγωγικές αρχές, προκειμένου να βελτιώσει την αποτελεσματικότητα της μαθησιακής διαδικασίας (Παναγιωτακόπουλος, Πιερρακέας, & Πιντέλας, 2003).

2.2 Προδιαγραφές Εκπαιδευτικού Υλικού

Υπάρχουν ποικίλα στοιχεία που θα μπορούσαν να καταταχτούν ως χαρακτηριστικά του εκπαιδευτικού υλικού (Καμέας, 2012). Συγκεκριμένα, εντοπίζουμε επτά βασικά χαρακτηριστικά – προδιαγραφές, που καθορίζουν την ποιότητα του εκπαιδευτικού υλικού (Καμέας, 2012):

- 1) **Δομή:** Ο τρόπος δόμησης του εκπαιδευτικού υλικού θα πρέπει κάθε φορά να λαμβάνει υπόψη του τις ιδιαιτερότητες που προκύπτουν από το γνωστικό αντικείμενο που πραγματεύεται και να διευκολύνει την απόκτηση νέων

γνώσεων, αλλά και την σύνδεση των νέων γνώσεων με το γνωσιακό υπόβαθρο και επίπεδο των εκπαιδευόμενων. Με μια καλά σχεδιασμένη δομή εξασφαλίζουμε την καλύτερη οργάνωση και τη λογική αλληλουχία του περιεχομένου. Τα υπο-χαρακτηριστικά που σχετίζονται με τη Δομή περιλαμβάνουν διακριτά τμήματα που βρίσκονται στην αρχή (π.χ. εισαγωγή), στο μέσο (π.χ. παραδείγματα) ή στο τέλος (π.χ. σύνοψη) του κειμένου (Race, 2001).

- 2) **Μορφή:** Η μορφή του περιεχομένου έχει μεγάλη σημασία για τον εκπαιδευόμενο, γιατί δημιουργεί την πρώτη εντύπωση και τον προδιαθέτει ευνοϊκά απέναντι στο θέμα. Η προσοχή του εκπαιδευόμενου προσελκύεται, μέσα από τη διαφορετικότητα των γραμματοσειρών και του μεγέθους τους, των διαφορετικών χρωμάτων και του κενού, έτσι ώστε ο εκπαιδευόμενος να μπορεί να διακρίνει τις σημαντικές πληροφορίες (Μικρόπουλος, 2000). Συστήνεται η χρήση καταλόγων, αριθμημένων ή μη, οι οποίοι προσελκύουν την προσοχή των εκπαιδευόμενων. Η τελική μορφή του περιεχομένου είναι ευθύνη του καλλιτεχνικού επιμελητή (Race, 2001).
- 3) **Οργάνωση:** Με την καλή οργάνωση του περιεχομένου εξασφαλίζουμε τη σαφή διάκριση των τμημάτων και υποτμημάτων του περιεχομένου, τη λογική αλληλουχία των νοημάτων, τη σωστή διάταξη των ιδεών και τη σταδιακή προσέγγιση των γνωστικών πεδίων. Το εκπαιδευτικό περιεχόμενο σε κάθε περίπτωση θα πρέπει να έχει λογική σύνδεση και συνοχή. Αυτό επιτυγχάνεται με την οργάνωση των περιεχομένων με τρόπο ώστε ο εκπαιδευόμενος να μπορεί να ενσωματώνει τις παρεχόμενες γνώσεις στις ήδη υπάρχουσες και να μπορεί να τις χρησιμοποιεί σε νέες καταστάσεις.
- 4) **Προσπέλαση:** Ο σωστός σχεδιασμός της προσπέλασης εξασφαλίζει την πρόσβαση σε διαφορετικά σημεία του περιεχομένου και την ομαλή μετάβαση από μια ενότητα στην επόμενη ή από ένα κεφάλαιο στο επόμενο (Μακράκης, 2000).
- 5) **Ανατροφοδότηση/Ανάδραση:** Οι ανατροφοδοτήσεις που περιέχονται στο περιεχόμενο λειτουργούν «θεραπευτικά» από μαθησιακής άποψης. Δίνουν

την ευκαιρία στους εκπαιδευόμενους να μαθαίνουν μέσω της πράξης. Είναι υποχρεωτική η χρήση ανατροφοδοτήσεων στο κείμενο (Race, 1999).

- 6) **Κίνητρα μάθησης:** Τα κίνητρα μάθησης δημιουργούν στους εκπαιδευόμενους τη θέληση για μάθηση. Διεγείρουν την προσοχή των εκπαιδευόμενων και τους δίνουν τη δυνατότητα να συμμετέχουν ενεργά και να μαθαίνουν μέσα από τις εμπειρίες που αποκομίζουν (Race, 2001).
- 7) **Μετα-περιγραφή:** Η μετα - περιγραφή περιγράφει το πλαίσιο μέσα στο οποίο χρησιμοποιείται το εκπαιδευτικό υλικό. Επίσης, μπορεί να χρησιμοποιηθεί για αναζήτηση του υλικού (π.χ. με λέξεις-κλειδιά).

2.3 Έντυπο Εκπαιδευτικό Υλικό

Το έντυπο εκπαιδευτικό υλικό συναντάται στις εξής μορφές:

- Βιβλία και εγχειρίδια
- Οδηγοί μελέτης
- Φύλλα εργασίας
- Χάρτες
- Άρθρα από εφημερίδες και περιοδικά

2.3.1 Προδιαγραφές Έντυπου Εκπαιδευτικού Υλικού

Κατά τη διάρκεια της εξέλιξής του το έντυπο εκπαιδευτικό υλικό απέκτησε τα εξής χαρακτηριστικά (Ματραλής, 1998, 1999):

- Αναφορά σαφώς διατυπωμένων στόχων και προσδοκώμενων αποτελεσμάτων σε κάθε ενότητα ή τμήμα του υλικού.
- Σαφές, επεξηγηματικό και φιλικό κείμενο.
- Παραδείγματα και μελέτες περίπτωσης.
- Ερωτήσεις και ασκήσεις αυτοαξιολόγησης.
- Δραστηριότητες και ασκήσεις για πρακτική άσκηση των σπουδαστών.
- Κατατετμημένη παρουσίαση της ύλης.
- Επεξηγηματικοί τίτλοι και υπότιτλοι των ενότητων.

- Σαφώς διατυπωμένη επίγνωση των δυσκολιών που πιθανώς θα συναντήσει ο σπουδαστής.
- Βιβλιογραφικές αναφορές, οδηγό περαιτέρω μελέτης και οδηγίες για ανεύρεση συμπληρωματικών πηγών.
- Οδηγό μελέτης και οδηγίες για τη χρήση όλων των μορφών του υλικού.
- Εμβάθυνση και γενικότερα αξιοποίηση της πολυτροπικότητας του κειμένου (Χοντολίδου, 1999).

Είναι αναγκαία, ακόμη, η επιλογή του κατάλληλου τόνου και ύφους στο εκπαιδευτικό υλικό. Η καθημερινή, προσιτή και μη τυπική γλώσσα προτιμάται όλο και περισσότερο στα σύγχρονα εγχειρίδια, ενώ ο τόνος είναι κατά πολύ λιγότερο τυπικός απ' όσο σε μερικά παλαιότερα διδακτικά βιβλία ή άρθρα περιοδικών. Το φιλικό ύφος γραφής, η απλή διατύπωσή του, οι επεξηγήσεις σημαντικών εννοιών, οι συνόψεις σημαντικών σημείων, οι περιγραφές διαδικασιών, οι απεικονίσεις όπου μπορούν να αντικαταστήσουν ένα εκτενές τμήμα κειμένου, καθιστούν ένα κείμενο περισσότερο ελκυστικό, ενώ αντίθετα η ύπαρξη ασαφειών δυσκολεύει πολύ τον εκπαιδευόμενο.

Ένα καλό εγχειρίδιο παρουσιάζει περισσότερα οπτικά ερεθίσματα (πίνακες δεδομένων, διαγράμματα, γραφήματα, φωτογραφίες, κ.ά.) μιας και οι οπτικές εικόνες απομνημονεύονται ευκολότερα απ' ότι εκείνες που περιγράφονται με λέξεις.

Η εισαγωγή στην αρχή κάθε κεφαλαίου βοηθά τον εκπαιδευόμενο να κρίνει ποια κεφάλαια τον ενδιαφέρουν περισσότερο τη συγκεκριμένη στιγμή. Η εισαγωγή δίνει, επίσης, στο συγγραφέα μια μοναδική ευκαιρία να κερδίσει την πρώτη καλή εντύπωση του αναγνώστη. Σε περίπτωση που η εισαγωγή είναι ανιαρή, οι εκπαιδευόμενοι μπορεί να κλείσουν το βιβλίο δίχως ν' αρχίσουν καν το διάβασμα. Εάν όμως η εισαγωγή τους υποκινήσει, είναι πιθανότερο οι εκπαιδευόμενοι να ενεργοποιηθούν και να εμπλακούν.

Ο σχεδιασμός και η διατύπωση του σκοπού και των προσδοκώμενων αποτελεσμάτων είναι, επίσης, αρκετά σημαντική. Ο σκοπός θα πρέπει να αποτελεί μια μικρή σε έκταση προέκταση του τίτλου του κεφαλαίου, που έχει ως στόχο να κατατοπίσει καλύτερα τον εκπαιδευόμενο αναφορικά με το τι, σε γενικές γραμμές, μπορεί να μάθει στο κεφάλαιο αυτό. Στη συνέχεια τα προσδοκώμενα αποτελέσματα αποτελούν περιγραφή αυτών που θα είναι ικανός να κάνει (ή ενδεχομένως να κάνει καλύτερα) ο εκπαιδευόμενος, όταν θα έχει μελετήσει το κεφάλαιο που ακολουθεί. Για να είναι χρήσιμα για τους εκπαιδευόμενους, ώστε να μην τα αγνοήσουν, θα πρέπει να συνοδεύονται πολύ καλά με τις ασκήσεις αυτοαξιολόγησης και τις δραστηριότητες κάθε κεφαλαίου.

Σημαντικά στοιχεία του έντυπου υλικού αποτελούν οι ασκήσεις αυτοαξιολόγησης και οι δραστηριότητες, συνοδευόμενες από τις κατάλληλες απαντήσεις τους στο τέλος του κεφαλαίου ή του βιβλίου. Τα δυο αυτά στοιχεία καθορίζουν κυρίως το βαθμό αλληλεπίδρασης του έντυπου υλικού με τον εκπαιδευόμενο και τον εμπλέκουν κατά τον καλύτερο δυνατό τρόπο στη μαθησιακή διεργασία. Οι απαντήσεις των ασκήσεων αυτοαξιολόγησης και των δραστηριοτήτων που περιγράφουν τις σωστές ενέργειες που πρέπει να κάνει ο εκπαιδευόμενος κατά την εκτέλεσή τους, αλλά και τα πιθανά λάθη του, τον βοηθούν να αξιολογεί ο ίδιος την επίδοσή του. Η ανατροφοδότηση που παρέχουν τέτοιου τύπου απαντήσεις βοηθά τους εκπαιδευόμενους να ανακαλύψουν μόνοι τους ακριβώς για ποιο λόγο, παραδείγματος χάριν, κάνουν τα συγκεκριμένα λάθη.

Επίσης, είναι χρήσιμο στο έντυπο εκπαιδευτικό υλικό να παρέχονται συμβουλές για το πώς πρέπει να μελετηθεί. Οι συμβουλές αυτές μπορούν να δίνονται στα διάφορα σχόλια μελέτης τα οποία ενσωματώνονται στο κείμενο και παρέχουν συγκεκριμένες οδηγίες και συμβουλές που σχετίζονται άμεσα με τα σημεία στα οποία βρίσκονται.

Απαραίτητη στο τέλος κάθε γνωστικής ενότητας είναι και η σύνοψη, η οποία αποτελεί ένα πρόσφορο σημείο στάσης και αυτοελέγχου του φοιτητή ως προς την πορεία της μελέτης του και ταυτόχρονα ένα σημείο κατάλληλο για να απευθυνθεί ο

συγγραφέας ξανά στο φοιτητή και να τον κατευθύνει εξατομικευμένα στη μελέτη του. Είναι φανερό ότι είναι σημαντικό η σύνοψη να έχει στενή σχέση με τα προσδοκώμενα μαθησιακά αποτελέσματα και να μην επαναλαμβάνει απλώς τα κύρια σημεία της γνωστικής ενότητας, αλλά να χρησιμοποιείται ως προαναγγελία για τα επόμενα, δηλαδή αν μια ενότητα αποτελεί «σκαλοπάτι» για την επόμενη, είναι χρήσιμο να εξηγείται με συντομία αυτό που ακολουθεί και σχετίζεται άμεσα με τα παρόντα.

Επίσης, ένας οδηγός για περαιτέρω μελέτη βοηθά τους σπουδαστές να αξιοποιούν ορισμένα βιβλία, άρθρα ή ακόμη και οπτικοακουστικά ή ηλεκτρονικά εκπαιδευτικά μέσα ως συνέχεια του έντυπου εκπαιδευτικού υλικού που μελετούν. Απαραίτητη προϋπόθεση, βέβαια, για την επιλογή τους είναι η βαθιά γνώση εκ μέρους του συγγραφέα των βιβλίων και λοιπών εκπαιδευτικών μέσων που προτείνει, καθώς επίσης και η δυνατότητα πρόσβασης ή αγοράς τους. Πρέπει, ακόμη, αυτές οι πηγές μάθησης να συνδέονται με τα προσδοκώμενα αποτελέσματα και να αναγκάζουν τους σπουδαστές να συγκρίνουν και να αντιδιαστέλλουν διαφορετικές πηγές όταν ένα συγκεκριμένο θέμα αντιμετωπίζεται με διαφορετικούς τρόπους. Σκόπιμο, επίσης, είναι στον οδηγό για περαιτέρω μελέτη να εστιάζονται συγκεκριμένες σελίδες ή τμήματα, ώστε να μειώνεται η διαδικασία της ανάγνωσης του λιγότερο σχετικού ή λιγότερο σημαντικού υλικού.

2.3.2 Προδιαγραφές Φύλλων Εργασίας

Ακολουθούν προδιαγραφές που αφορούν τη σχεδίαση ενός φύλλου εργασίας, τόσο από τη μεριά του διδάσκοντα όσο και από αυτή του σπουδαστή (Κορδάκη, 2010):

Ø Γενικές Προδιαγραφές:

- Να έχει ξεκάθαρο στόχο τη διδασκαλία
- Να είναι δομημένο με τη μορφή μαθημάτων για αυτοεκπαίδευση
- Να προσφέρει δυνατότητες αξιολόγησης της πορείας του σπουδαστή
- Να αξιοποιεί τα πολυμέσα (multimedia)
- Να παρέχει ευκολίες για εκτύπωση

Ø Προδιαγραφές Φύλλων Εργασίας για το Διδάσκοντα:

A. Εισαγωγή – Ψυχολογική και Γνωστική Προετοιμασία:

- Διαμόρφωση θερμού συναισθηματικού κλίματος ασφάλειας για το σπουδαστή (Διατυπώστε ένα εισαγωγικό σχόλιο που θα χρησιμοποιήσετε)
- Διαμόρφωση κινήτρου για το μάθημα (Διατυπώστε ένα εισαγωγικό σχόλιο ώστε να προκαλέσετε το ενδιαφέρον του μαθητή για το μάθημα)
- Ενημέρωση των σπουδαστών για το τι θα επακολουθήσει (Με πολύ λίγες λέξεις)
- Ενημέρωση των σπουδαστών για τους στόχους του μαθήματος (Μετά το τέλος του μαθήματος θα πρέπει να είσατε ικανοί να....)
- Διερεύνηση πρότερης γνώσης (Τι έχετε ακούσει ή διαβάσει για... Τι νομίζετε για...)
- Διερεύνηση προαπαιτούμενης γνώσης

B. Εισαγωγική δραστηριότητα:

- στόχος,
- τι αναμένουμε,
- περιγραφή της δραστηριότητας,
- τι ερωτήσεις θα κάνουμε προφορικά και τι γραπτά,
- παράθεση απαραίτητων στοιχείων,
- δημιουργία και παράθεση σχετικών υλικών στους σπουδαστές,
- δημιουργία ομάδων,
- τι θα κάνει κάθε μέλος της ομάδας

C. Δραστηριότητες εμπέδωσης:

- στόχος,
- τι αναμένουμε,
- περιγραφή της δραστηριότητας,
- τι ερωτήσεις θα κάνουμε προφορικά και τι γραπτά,

- παράθεση απαραίτητων στοιχείων,
- δημιουργία και παράθεση σχετικών υλικών στους σπουδαστές,
- δημιουργία ομάδων,
- τι θα κάνει κάθε μέλος της ομάδας

D. Αξιολόγηση:

- τι αναμένουμε,
- περιγραφή,
- τι ερωτήσεις θα κάνουμε προφορικά και τι γραπτά (Η αξιολόγηση πρέπει να ελέγχει αν επιτεύχθηκαν οι στόχοι που αρχικά βάλαμε στο μάθημα, οι οποίοι δεν πρέπει να είναι πολλοί)

E. Μεταγνωστική Φάση:

- Συνόψιση (Να φανεί ένα διάγραμμα συνόψισης)
- Δουλειά για το σπίτι (Να περιγραφτεί σαφώς)
- Μεταγνωστική αξιολόγηση (Τι κερδίσατε από αυτό το μάθημα;)

Ø Προδιαγραφές Φύλλων Εργασίας για το Σπουδαστή:

A. Εισαγωγή – Ψυχολογική και Γνωστική Προετοιμασία:

- Ενημέρωση των σπουδαστών για το αντικείμενο που θα μάθουν (Επιγραμματικά ο τίτλος)
- Ενημέρωση των σπουδαστών για τους στόχους του μαθήματος (Μετά το τέλος του μαθήματος θα πρέπει να είμαστε ικανοί να... α)... β)... γ)...

B. Εισαγωγική δραστηριότητα:

- περιγραφή,
- παράθεση απαραίτητων στοιχείων,
- παράθεση σχετικών υλικών στους σπουδαστές,
- ερωτήσεις που θα απαντήσουν,
- τι θα κάνει κάθε μέλος της ομάδας,
- αφήνουμε χώρο στο χαρτί για να διεκπεραιωθεί η δραστηριότητα

C. Δραστηριότητες εμπέδωσης:

- περιγραφή,

- παράθεση απαραίτητων στοιχείων,
- παράθεση σχετικών υλικών στους σπουδαστές,
- ερωτήσεις που θα απαντήσουν,
- τι θα κάνει κάθε μέλος της ομάδας,
- αφήνουμε χώρο στο χαρτί για να διεκπεραιωθεί η δραστηριότητα

D. Αξιολόγηση:

- περιγραφή της δραστηριότητας αξιολόγησης και των ερωτήσεων αξιολόγησης

E. Μεταγνωστική Φάση:

- Συνοψίστε...
- Δουλειά για το σπίτι (Να περιγραφτεί)
- Μεταγνωστική αξιολόγηση (Τι κερδίσατε από αυτό το μάθημα;)

2.4 Ψηφιακό Εκπαιδευτικό Υλικό

Ως ψηφιακό υλικό θεωρούμε το κατασκεύασμα το οποίο συνδυάζει ψηφιακό περιεχόμενο (digital content), κάποιο μέσο διάθεσης του περιεχομένου (media) και έχει ένα συγκεκριμένο σκοπό ή εφαρμογή (application). Εάν ο σκοπός είναι εκπαιδευτικός, τότε αναφερόμαστε σε Ψηφιακό Εκπαιδευτικό Υλικό (ΨΕΥ). Για παράδειγμα σε ένα ΨΕΥ το περιεχόμενο είναι το κείμενο, το τεχνολογικό μέσο είναι μια εφαρμογή υπερκειμένου, ενώ η παιδαγωγική/διδασκτική εφαρμογή είναι το πλαίσιο μέσα στο οποίο χρησιμοποιείται η εφαρμογή (πχ για συμπληρωματική εκπαίδευση σε ένα εργαστηριακό μάθημα). Συνεπώς, το ΨΕΥ αποτελείται από τις παρακάτω τρεις συνιστώσες: α) Ψηφιακό περιεχόμενο, β) Τεχνολογικό μέσο, γ) Παιδαγωγική/Διδασκτική εφαρμογή.

Το ψηφιακό περιεχόμενο εμφανίζεται σε πολλούς τύπους καθένας από τους οποίους αποθηκεύεται σε ένα σύνολο διαφορετικών μορφοτύπων (formats). Οι βασικοί τύποι ψηφιακού περιεχομένου, όπως αυτοί ορίζονται από τη διεθνή κοινότητα, είναι πέντε:

- 1) Κείμενο (Text): Ως κείμενο θεωρείται ένα συνεκτικό σύνολο χαρακτήρων, λέξεων ή παραγράφων, το οποίο μπορεί να εμπεριέχει και στατικό οπτικό υλικό.
- 2) Ακουστικό υλικό (Audio): Αναφέρεται κυρίως σε ηχογραφήσεις που είναι διαθέσιμες στον τελικό χρήστη για αναπαραγωγή.
- 3) Στατικό οπτικό υλικό (Graphic): Πρόκειται για στατικό ψηφιακό υλικό που βασίζεται στην οπτική (και όχι λεκτική ή ηχητική) αναπαράσταση. Περιλαμβάνει φωτογραφίες, εικόνες, χάρτες, διαγράμματα, κ.λπ.
- 4) Οπτικοακουστικό υλικό (Video): Στην κατηγορία αυτή αναφέρεται υλικό το οποίο έχει παραχθεί με τη βοήθεια οπτικοακουστικών μέσων εγγραφής.
- 5) Κινούμενο οπτικό υλικό (Animation): Στην κατηγορία αυτή αναφέρεται το υλικό σε μορφή κινούμενης εικόνας, το οποίο δε θεωρείται ως οπτικοακουστικό υλικό. Κύριος αντιπρόσωπος είναι τα κινούμενα σχέδια τα οποία μπορούν να αναπαραστήσουν διαδικασίες ή πειράματα που είναι δύσκολο να βιντεοσκοπηθούν σε πραγματικό περιβάλλον.

Το ΨΕΥ επηρεάζεται άμεσα από την ανάπτυξη και εξέλιξη των νέων τεχνολογιών και ειδικότερα της Τεχνολογίας Πληροφοριών και Επικοινωνίας (ΤΠΕ), καθώς και των υπολογιστικών συστημάτων. Για να μελετήσουμε το ΨΕΥ χρειάζεται να δούμε το περιεχόμενο και το βαθμό στον οποίο αυτό εξυπηρετεί τους εκπαιδευτικούς σκοπούς για τους οποίους προορίζεται, καθώς και τη δομή του ψηφιακού περιεχομένου και τα τεχνολογικά μέσα που χρησιμοποιούνται προκειμένου να μεταδοθεί και να συνεισφέρει στη δημιουργία εμπειριών μάθησης.

Ο ρόλος που το ψηφιακό εκπαιδευτικό υλικό διαδραματίζει στη μαθησιακή διαδικασία πρέπει να σχετίζεται άμεσα με τη θεωρία μάθησης ή την επιστημολογία γνώσης που διέπει το σχεδιασμό τόσο της πληροφορίας στο ψηφιακό υλικό, όσο και του τρόπου χρήσης της πληροφορίας αυτής διαμέσου διδακτικών παρεμβάσεων. Είναι, πλέον, αρκετά σαφές ότι οι αρχές του συμπεριφορισμού, του κονστρουκτιβισμού ή της κοινωνικο-πολιτισμικής ψυχολογίας μας ωθούν να εστιάσουμε σε διαφορετικά σημεία της διαδικασίας μάθησης και διδασκαλίας και

μας παραπέμπουν να δούμε διαφορετικούς ρόλους στη σχέση υλικού – εκπαιδευτή – εκπαιδευομένου. Για παράδειγμα, στο πλαίσιο του συμπεριφορισμού έμφαση δίδεται στη διαδικασία μάθησης ως ερέθισμα-απόκριση, ενώ ο κονστрукτιβισμός εστιάζει στη σημασία του γνωστικού εμποδίου και της ενεργούς ενασχόλησης του ατόμου για την επίλυση αυτών των εμποδίων με στόχο την εννοιολογική αλλαγή.

Αντίθετα, στο πλαίσιο της κοινωνικο-ιστορικο-πολιτισμικής θεώρησης, έχει υποστηριχθεί η σημασία της διαμεσολάβησης για την ανάπτυξη μαθησιακών δεξιοτήτων. Έτσι, δίνεται έμφαση στην οργάνωση της διδακτικής παρέμβασης και στο ρόλο των πολιτισμικών εργαλείων, που υποστηρίζουν δυναμικά τη διδακτική διαδικασία. Η μάθηση δεν αποτελεί προϊόν μιας μοναχικής και ατομικής ενασχόλησης με κάποιο έργο, αλλά είναι το αποτέλεσμα συλλογικής και ιστορικής δράσης σε συστηματικά οργανωμένα πλαίσια δραστηριοτήτων (Vygotsky, 1978).

Οι Hedegaard (1999) και Davydov (1999) εξηγούν ότι τα κύρια δομικά στοιχεία στην οργάνωση των μαθησιακών δραστηριοτήτων είναι οι διδακτικοί στόχοι οι οποίοι προσδιορίζουν που στοχεύει η διδακτική παρέμβαση, οργανώνουν τις δραστηριότητες σε αυτόνομες ενότητες δράσεις, συνδέουν τις διάφορες μορφές γνώσεις και οι μαθησιακές δράσεις που διευκρινίζουν το είδος της δράσης των εκπαιδευομένων, παρέχουν πλαίσια έκφρασης προσωπικών ιδεών, εξάσκησης, διερεύνησης αλλά και πρόκληση ενδιαφέροντος για νέα γνώση.

Στις παραπάνω θεωρητικές προσεγγίσεις για τη μάθηση, το ΨΕΥ μπορεί να ανταποκριθεί ικανοποιητικά, λόγω των πλεονεκτημάτων του. Τα τελευταία χρόνια έχει δοθεί έμφαση στον προβληματισμό για τη διδακτική αξιοποίηση του ΨΕΥ και πιο συγκεκριμένα στο γεγονός ότι το ΨΕΥ, σε συνάρτηση με τις νέες τεχνολογίες πληροφοριών και επικοινωνίας, δεν αποτελούν από μόνα τους μέσα μάθησης, αλλά αντίθετα πρέπει να ειπωθούν ως αναπόσπαστα μέρη της μαθησιακής διαδικασίας. Το ΨΕΥ (πχ μια ιστοσελίδα με υλικό, ένα βίντεο, ένα λογισμικό, κλπ) δεν αποτελεί από μόνο του περιβάλλον μάθησης, καθώς μπορεί να καθιστά τα άτομα παθητικούς δέκτες της ψηφιακής πληροφορίας. Σε αυτήν τη διαδικασία βασική αφετηρία αποτελεί η επιλογή της κατάλληλης διδακτικής στρατηγικής έτσι ώστε να

τεκμηριώνεται η προστιθέμενη αξία ψηφιακού εκπαιδευτικού υλικού στη διαδικασία μάθησης και διδασκαλίας του γνωστικού αντικειμένου. Η επιλογή του κατάλληλου ΨΕΥ πρέπει να γίνει με βάση τα ποιοτικά του χαρακτηριστικά από τα οποία θα προκύπτει και η καταλληλότητα του.

2.4.1 Προδιαγραφές Ψηφιακού Εκπαιδευτικού Υλικού

Η ποιότητα του ΨΕΥ είναι μια αφηρημένη έννοια και για να μετρηθεί είναι απαραίτητο να υπάρχει ένα μοντέλο ποιότητας. Επειδή μόλις τα τελευταία χρόνια έχει αναπτυχθεί μεγάλος όγκος ΨΕΥ, ενώ η συνεχής εξέλιξη της τεχνολογίας είχε ως αποτέλεσμα να μεταβάλλεται συνεχώς η μορφή του, δεν υπάρχει στη βιβλιογραφία σαφής ορισμός της ποιότητας του ΨΕΥ. Όπως είναι λογικό, η έννοια της ποιότητας του ΨΕΥ συνδέεται άμεσα με την έννοια της ποιότητας στην εκπαίδευση. Ένας τύπος του ψηφιακού εκπαιδευτικού υλικού είναι το λογισμικό. Τόσο για την εκπαίδευση όσο και για το λογισμικό έχουν γίνει προσεγγίσεις της έννοιας της ποιότητάς τους. Συνεπώς, αυτές οι συγκεκριμένες προσεγγίσεις μπορούν να συνδυαστούν και να χρησιμοποιηθούν και για την έννοια της ποιότητας του ΨΕΥ, αφού αυτή είναι συνυφασμένη τόσο με την ποιότητα του μέσου (λογισμικού), όσο και με την ποιότητα του σκοπού (εκπαιδευτής). Βέβαια, το ΨΕΥ δεν εξαρτάται μόνο από το μέσο και τον σκοπό, αλλά και από πολλά άλλα ιδιαίτερα χαρακτηριστικά τα οποία καθορίζουν και αυτά την ποιότητα του.

2.4.1.1 Ψηφιακό Περιεχόμενο

Το ψηφιακό εκπαιδευτικό περιεχόμενο είναι ουσιαστικό στοιχείο της εκπαιδευτικής διαδικασίας και κατ' επέκταση του ψηφιακού εκπαιδευτικού υλικού. Το εκπαιδευτικό περιεχόμενο θα πρέπει να είναι εναρμονισμένο με το αντίστοιχο πρόγραμμα σπουδών στα πλαίσια του οποίου θα χρησιμοποιηθεί, να συσχετίζει γνώσεις από διαφορετικές επιστημονικές περιοχές υλοποιώντας μια περισσότερο διαθεματική προσέγγιση του μαθησιακού στόχου. Επιπλέον, το εκπαιδευτικό περιεχόμενο πρέπει να παραμένει συνεπές ως προς τους διατυπωμένους μαθησιακούς στόχους και οι στόχοι αυτοί να εναρμονίζονται με τη συνολική λογική της διδακτικής ενότητας.

Όταν αναφερόμαστε στο εκπαιδευτικό περιεχόμενο αναφερόμαστε κατά κύριο λόγο στην επιστημονική γνώση που το ΨΕΥ μεταφέρει και η οποία θα πρέπει να είναι επιστημονικά τεκμηριωμένη και αξιόπιστη, χωρίς διαστρεβλώσεις ή παραλείψεις που είναι δυνατόν να την αλλοιώσουν. Σημαντικό είναι να αποφεύγονται μεροληπτικές ή μη τεκμηριωμένες απόψεις, μονομερής, επιλεκτική ή άνιση παρουσίαση πληροφοριών καθώς και επιπόλαιη χρήση εκφράσεων της καθομιλουμένης. Στην περίπτωση που παρουσιάζει ενδεχόμενες αντικρουόμενες επιστημονικές απόψεις, αυτές θα πρέπει να δίνονται με αντικειμενικότητα. Οι νέες γνώσεις θα πρέπει να παρουσιάζονται με σωστή τεκμηρίωση και όπου είναι απαραίτητο να υπάρχουν παραπομπές. Απαραίτητη τέλος, είναι η κατάλληλη σύνδεση του εκπαιδευτικού περιεχομένου με το ηλικιακό και γνωστικό υπόβαθρο των εκπαιδευομένων (Παναγιωτακόπουλος, Πιερρακέας, & Πιντέλας, 2003).

Το εκπαιδευτικό περιεχόμενο αποτελεί συχνά το άλφα και το ωμέγα της μάθησης και πρέπει να επικεντρώνεται στην αλληλεπίδραση διότι αυτό που έχει πραγματικά σημασία είναι αυτό που κάνουν οι εκπαιδευόμενοι. Η μάθηση προκύπτει μέσα από την προσπάθεια και στη συνέχεια μέσα από την ανατροφοδότηση. Συνεπώς, το εκπαιδευτικό περιεχόμενο πρέπει να διαθέτει κάποια παιδαγωγικά και τεχνικά χαρακτηριστικά ώστε να υποκινεί στους εκπαιδευόμενους τη «θέληση της μάθησης», να καλλιεργεί και να διαμορφώνει την αυτοεκτίμηση των εκπαιδευομένων και να ανταποκρίνεται πλήρως στις ανάγκες μιας ομάδας – στόχου εκπαιδευομένων (Race, 1999).

Τα χαρακτηριστικά ποιότητας του ψηφιακού εκπαιδευτικού περιεχομένου είναι:

- 1) **Δομή:** Με μια καλή σχεδιασμένη δομή εξασφαλίζουμε την καλύτερη οργάνωση και τη λογική αλληλουχία του περιεχομένου. Επίσης, η δομή επηρεάζει την αφομοίωση του περιεχομένου από τους εκπαιδευόμενους. Ο τρόπος δόμησης του εκπαιδευτικού υλικού θα πρέπει κάθε φορά να λαμβάνει υπόψη του τις ιδιαιτερότητες που προκύπτουν από το γνωστικό αντικείμενο που πραγματεύεται και να διευκολύνει την απόκτηση νέων

γνώσεων, αλλά και τη σύνδεση των νέων γνώσεων με το γνωστικό υπόβαθρο και επίπεδο των εκπαιδευομένων (Race, 2001).

- 2) **Μορφή:** Η μορφή του περιεχομένου έχει μεγάλη σημασία για τον εκπαιδευόμενο, γιατί δημιουργεί την πρώτη εντύπωση και τον προδιαθέτει ευνοϊκά απέναντι στο θέμα. Η προσοχή του εκπαιδευόμενου προσελκύεται, μέσα από τη διαφορετικότητα των γραμματοσειρών και του μεγέθους τους, των διαφορετικών χρωμάτων, έτσι ώστε ο εκπαιδευόμενος να μπορεί να διακρίνει τις σημαντικές πληροφορίες (Μικρόπουλος, 2000). Επιπλέον η χρήση σχημάτων, εικόνων, πινάκων και γραφημάτων προσελκύουν την προσοχή των εκπαιδευομένων. Σημαντικό είναι, επίσης, ότι η μορφή προσδίδει καλαισθησία στο περιεχόμενο (Race, 2001).
- 3) **Οργάνωση:** Με την καλή οργάνωση του περιεχομένου εξασφαλίζουμε τη σαφή διάκριση των τμημάτων και υπο-τμημάτων του περιεχομένου, τη λογική αλληλουχία των νοημάτων, τη σωστή διάταξη των ιδεών και τη σταδιακή προσέγγιση των γνωστικών πεδίων.
- 4) **Προσπέλαση:** Με την προσπέλαση εξασφαλίζουμε την πρόσβαση σε διαφορετικά σημεία του περιεχομένου και την ομαλή μετάβαση από μια ενότητα στην επόμενη ή από ένα κεφάλαιο στο επόμενο, την άμεση επιστροφή στην αρχική οθόνη, κ.λπ. Υπάρχουν πέντε τρόποι προσπέλασης του περιεχομένου: η γραμμική ή σειριακή, η ιεραρχική, η μικτή-ιεραρχική, η ομόκεντρη και η υπερμεσική. Το εκπαιδευτικό περιεχόμενο μπορεί να ακολουθεί έναν από αυτούς τους τρόπους προσπέλασης ή να συνδυάζει περισσότερους, σε κάθε περίπτωση όμως θα πρέπει να έχει συγκεκριμένη δομή και οργάνωση περιεχομένων με λογική σύνδεση και συνοχή (Μακράκης, 2000).
- 5) **Ανατροφοδότηση/Ανάδραση:** Οι ανατροφοδοτήσεις που περιέχονται στο περιεχόμενο του ψηφιακού εκπαιδευτικού υλικού δίνουν την ευκαιρία στους εκπαιδευόμενους να μαθαίνουν μέσω της πράξης (Race, 1999).
- 6) **Κίνητρα Μάθησης:** Τα κίνητρα μάθησης που περιέχονται στο ψηφιακό περιεχόμενο δημιουργούν στους εκπαιδευόμενους τη θέληση για μάθηση.

Διεγείρουν την προσοχή των εκπαιδευόμενων και τους δίνουν τη δυνατότητα να συμμετέχουν ενεργά και να μαθαίνουν μέσα από τις εμπειρίες που αποκομίζουν (Race, 2001).

2.4.1.2 Τεχνολογικό Μέσο

Η τεχνική αρτιότητα του ψηφιακού εκπαιδευτικού υλικού αφορά το τεχνολογικό μέσο και περιλαμβάνει ένα σύνολο καθαρά τεχνικών θεμάτων που αφορούν διαδικασίες εγκατάστασης, δυνατότητα λειτουργίας σε δίκτυο, ασφάλεια στη μετάδοση δεδομένων, τεχνικές προδιαγραφές για τη χρήση γραφικών, ήχου, πολυμέσων, κ.ά. Το προτεινόμενο μοντέλο υιοθετεί τα χαρακτηριστικά και υπο-χαρακτηριστικά ποιότητας του πρότυπου ISO/IEC 9126 (Βικιπαίδεια - ISO IEC 9126, 2016) για την αξιολόγηση της τεχνικής αρτιότητας. Στο πρότυπο ISO/IEC 9126 κάθε χαρακτηριστικό και κάθε υπο-χαρακτηριστικό περιγράφεται περαιτέρω από κατάλληλα εξωτερικά και εσωτερικά γνωρίσματα, τα οποία αξιολογούνται από μετρικές. Οι μετρικές προσδιορίζονται στα πλαίσια του τρόπου εφαρμογής τους, της μέτρησης, της διατύπωσης, του σκοπού τους, της ερμηνείας της ποιότητας, των δεδομένων εισόδου και των τελικών αποδεκτών.

2.4.1.3 Παιδαγωγική/Διδακτική Εφαρμογή

Η παιδαγωγική και διδακτική καταλληλότητα αναφέρεται στις διδακτικές προσεγγίσεις και τις διαδικασίες μάθησης που υιοθετεί το εκπαιδευτικό υλικό, καθώς και στη συμβατότητα αυτών με το μαθησιακό προφίλ των εκπαιδευόμενων στους οποίους απευθύνεται το εκπαιδευτικό υλικό. Σύμφυτο στοιχείο της παιδαγωγικής καταλληλότητας είναι και τα κίνητρα μάθησης.

Για την παιδαγωγική/διδακτική εφαρμογή τα χαρακτηριστικά ποιότητας είναι:

- **Μετα-Περιγραφή:** Η μετα-περιγραφή περιγράφει το πλαίσιο μέσα στο οποίο χρησιμοποιείται το ψηφιακό εκπαιδευτικό υλικό.
- **Προσαρμογή:** Η προσαρμογή περιγράφει το πλαίσιο μέσα στο οποίο τροποποιείται ή μεταβάλλεται το ψηφιακό εκπαιδευτικό υλικό.

2.4.2 Προδιαγραφές Διαδραστικού Λογισμικού Ασκήσεων

Το διαδραστικό λογισμικό είναι λογισμικό με το οποίο ο χρήστης μπορεί να αλληλεπιδράσει. Με τον τρόπο αυτό εφαρμόζει κατευθείαν τις γνώσεις που του προσφέρονται και βλέπει τα αποτελέσματα και αξιολογεί τα συμπεράσματα. Το διαδραστικό λογισμικό περιλαμβάνει πολλά και διαφορετικά μαθησιακά αντικείμενα, όπως δοκίμιο, οπτικοακουστικά μέσα, υπερκείμενο, ασκήσεις, πειράματα, κ.λπ., τα οποία παρουσιάζει μέσα από κοινή διεπαφή, η οποία υποστηρίζει συνεκτικό διάλογο με το χρήστη. Ακόμη, μπορεί να εφαρμόζει διαφορετικές εκπαιδευτικές στρατηγικές, να περιλαμβάνει μοντέλο ή στερεότυπα εκπαιδευόμενου, να προσαρμόζει την πρόσβαση στο εκπαιδευτικό περιεχόμενο, να αξιολογεί τον εκπαιδευόμενο και να παρέχει ανατροφοδότηση.

Ακολουθούν οι γενικές, τεχνικές και εκπαιδευτικές προδιαγραφές που πρέπει να ακολουθούνται στην περίπτωση διαδραστικού λογισμικού ασκήσεων σε ψηφιακή μορφή (ΤΕΙ Θεσσαλίας, 2015).

Ø Γενικές Προδιαγραφές:

- Να αξιοποιεί όλα τα επικοινωνιακά μέσα (κείμενο, εικόνα, ήχο, μουσική, video, σχέδια και κινούμενα σχέδια)
- Όλα τα στοιχεία να «δένουν» μεταξύ τους
- Να είναι αλληλεπιδραστική

Ø Τεχνικές Προδιαγραφές:

- Να είναι γραμμένη για πολλά λειτουργικά συστήματα
- Να έχει εύκολη εγκατάσταση
- Να είναι εύκολος και εργονομικός ο χειρισμός
- Να είναι εύκολη η μεταφορά αρχείων και η εκτύπωση
- Να είναι αισθητικά μελετημένο
- Να είναι διαχρονική
- Να παρέχει ευκολίες για εκτύπωση

Ø Εκπαιδευτικές Προδιαγραφές:

- Να είναι ξεκάθαροι οι εκπαιδευτικοί στόχοι

- Να έχει σχεδιαστεί από εκπαιδευτικούς
- Να δίνει τη δυνατότητα να χρησιμοποιηθεί η εφαρμογή πολλαπλώς
- Να τονίζει τα βασικά σημεία του μαθήματος
- Να προσφέρει επιπλέον πληροφορίες (κυρίως με πολυμεσική μορφή)
- Να αξιοποιεί την αλληλεπιδραστικότητα
- Να δημιουργεί ανακεφαλαιωτικές σημειώσεις
- Να δημιουργεί ερωτήσεις και ασκήσεις
- Να αξιοποιεί τα πολυμεσικά στοιχεία

Σημαντικό επίσης είναι να περιέχονται πληροφορίες προς τους σπουδαστές και να γίνεται αναλυτική περιγραφή του τρόπου με τον οποίο το συγκεκριμένο λογισμικό μπορεί να ενταχθεί στην εκπαιδευτική διαδικασία. Για το σκοπό αυτό, θα πρέπει να καταγραφούν πληροφορίες για τα εξής (Καμέας, 2012):

- Ø Περιγραφή της εκπαιδευτικής χρήσης του διαδραστικού λογισμικού
- Ø Περιγραφή των εκπαιδευτικών και μαθησιακών στόχων του διαδραστικού λογισμικού και πιο συγκεκριμένα:
 - τι θα αποκομίσει ο σπουδαστής μελετώντας το
 - τι συμπληρώνει από το υπάρχον εκπαιδευτικό υλικό
 - τι βελτιώνει στο υπάρχον εκπαιδευτικό υλικό
 - τι συνοδεύει ή αναλύει σε μεγαλύτερο βάθος, από το υπάρχον εκπαιδευτικό υλικό
 - γιατί ο σπουδαστής θα πρέπει να το μελετήσει
- Ø Σύνδεση του διαδραστικού λογισμικού με το υπάρχον (έντυπο και ψηφιακό – εφόσον υπάρχει) εκπαιδευτικό υλικό. Πιο συγκεκριμένα, θα πρέπει να καταγραφούν όλες οι πληροφορίες που αφορούν:
 - τις προαπαιτούμενες γνώσεις, δηλαδή τι θα πρέπει να έχει μελετήσει ο σπουδαστής πριν ασχοληθεί με το συγκεκριμένο λογισμικό και ποιες έννοιες θα πρέπει να έχει κατακτήσει

- αν τυχόν το συγκεκριμένο λογισμικό αντικαθιστά ή συμπληρώνει κάποιο κεφάλαιο ή ενότητα
- οποιαδήποτε άλλη πληροφορία συνδέει το συγκεκριμένο λογισμικό με το έντυπο και το ψηφιακό εκπαιδευτικό υλικό (εφόσον υπάρχει) και την εκπαιδευτική διαδικασία
- Ø Περιγραφή όλων των διαφορετικών στερεοτύπων σπουδαστών, που υποστηρίζει το διαδραστικό λογισμικό και θα πρέπει να ληφθούν υπόψη κατά τον σχεδιασμό
- Ø Εκτίμηση του χρόνου που απαιτείται για τη μελέτη του συγκεκριμένου διαδραστικού λογισμικού ή για τις επιμέρους ενότητές του
- Ø Τρόποι μελέτης – χρήσης του λογισμικού και σενάρια εφαρμογής του. Θα πρέπει να δίνονται οδηγίες προς τους σπουδαστές, που θα τους καθοδηγούν σχετικά με τους τρόπους που μπορούν να μελετήσουν το συγκεκριμένο λογισμικό, ανάλογα με το διαθέσιμο χρόνο τους και το σημείο που θα βρίσκονται στην ολοκλήρωση της ύλης όταν το μελετούν. Για το σκοπό αυτό, προτείνεται να καταγραφεί τουλάχιστον ένα σενάριο μελέτης, το οποίο θα παρουσιάζει σε μορφή διαγράμματος ροής τον προτεινόμενο τρόπο μελέτης.

2.5 Προσδοκώμενα Αποτελέσματα και Μαθησιακοί Στόχοι

Είναι μια μορφή δήλωσης του «αποτελέσματος της μάθησης» (Race, 1999), καθώς περιγράφουν την επιθυμητή τελική κατάσταση του εκπαιδευόμενου, όσον αφορά τις αλλαγές που συμβαίνουν κατά τη διάρκεια της μαθησιακής διαδικασίας και πρέπει να είναι διατυπωμένοι με μετρήσιμα ρήματα (Gagné, 1977). Αποτελούν ευδιάκριτη και ακριβή έκφραση πρόθεσης ώστε να βοηθήσουν τους εκπαιδευόμενους να καταλάβουν ακριβώς τι είδους ικανότητες πρόκειται να αποκτήσουν και τους παρέχουν τα μέσα για να ελέγχουν την πρόοδό τους. Απαρτίζουν το σημασιολογικό χάρτη του υλικού και η παράθεσή τους αποτελεί έναν «χαρακτηριστικό τρόπο να παρακινηθεί ο εκπαιδευόμενος» αφού ξέρει τι θα έχει επιτύχει στο τέλος της μελέτης του (Lionarakis, 1999).

Ως προσδοκώμενο αποτέλεσμα ορίζεται η συμπεριφορά, την οποία αναμένουμε να εμφανίσει το άτομο μετά το πέρας μιας εκπαιδευτικής διαδικασίας (Mager, 1984). Η αναμενόμενη συμπεριφορά θα πρέπει να είναι παρατηρήσιμη και επαληθεύσιμη κατά αντικειμενικό τρόπο. Για να προσδιορίζεται σαφώς ένα αντικειμενικό προσδοκώμενο αποτέλεσμα θα πρέπει να χαρακτηρίζεται ως SMART:

- Εξειδικευμένο (Specific), χρησιμοποιώντας ένα ενεργητικό ρήμα που περιγράφει την παρατηρούμενη συμπεριφορά με ακρίβεια
- Μετρήσιμο (Measurable), χρησιμοποιώντας ένα ρήμα που περιγράφει μετρήσιμη συμπεριφορά
- Εφικτό (Attainable/Achievable), δηλαδή ρεαλιστικό, λαμβάνοντας υπόψη υφιστάμενες συνθήκες και τυχόν βοηθητικά στοιχεία
- Σχετικό (Relevant), υπονοώντας ότι πρέπει να διατηρεί συνέπεια με τις εκπαιδευτικές υποχρεώσεις και τις εργασίες του εκπαιδευόμενου στα πλαίσια της διδασκαλίας
- Χρονικά καθορισμένο (Timely/Time-Bound), προσδιορίζοντας ένα ρεαλιστικό χρονικό πλαίσιο για να επιτευχθεί η απόδοση, όπου απαιτείται

Ο Mager (1984) προτείνει επιπλέον μια ευρέως αποδεκτή στρατηγική για τη δημιουργία μαθησιακών στόχων – προσδοκώμενων αποτελεσμάτων. Σύμφωνα με αυτή τη στρατηγική, ένα καλά δομημένο προσδοκώμενο αποτέλεσμα θα πρέπει να αποτελείται από τα εξής τέσσερα στοιχεία:

- 1) **Ακροατήριο:** περιλαμβάνει τους εκπαιδευόμενους που μελετούν το υλικό. Κατά τη συγγραφή προσδοκώμενων αποτελεσμάτων αναφερόμαστε συνήθως σε αυτούς σε δεύτερο πρόσωπο (π.χ. θα είστε σε θέση να...)
- 2) **Απόδοση:** εκφράζεται από ένα ενεργητικό ρήμα – ικανό να περιγράφει μετρήσιμες ενέργειες και συγκεκριμένα επίπεδα γνώσης – και αναφέρεται στην παρατηρούμενη συμπεριφορά. Κάθε προσδοκώμενο αποτέλεσμα σχετίζεται με μία συμπεριφορά, άρα χρησιμοποιεί ένα ενεργητικό ρήμα. Η παρουσία περισσότερων του ενός ρήματος στην ίδια πρόταση συνήθως

υποδηλώνει την ανάγκη ανάλυσης του προσδοκώμενου αποτελέσματος σε επιμέρους αποτελέσματα

- 3) **Συνθήκη:** περιγράφει τις προϋποθέσεις κάτω από τις οποίες εκδηλώνεται η παρατηρούμενη συμπεριφορά. Επίσης, προσδιορίζει τα μέσα, τις διαδικασίες και τις πιθανές διευκολύνσεις που παρέχονται για την πραγματοποίηση του προσδοκώμενου αποτελέσματος ή γενικότερα το περιβάλλον που επηρεάζει την επιτέλεσή του
- 4) **Κριτήριο:** εκφράζει τον βαθμό στον οποίο πρέπει να εκδηλωθεί η παρατηρούμενη συμπεριφορά ώστε να έχουμε αποδεκτά επίπεδα απόδοσης. Συνεπώς, αποτελεί το μετρήσιμο στοιχείο ενός καλά δομημένου προσδοκώμενου αποτελέσματος. Στη λεκτική διατύπωση ενός προσδοκώμενου αποτελέσματος μπορούν να συμπεριληφθούν παραπάνω από ένα κριτήρια

Αναφορικά με τα ενεργητικά ρήματα που εκφράζουν την απόδοση, προτείνεται να αντιστοιχίζονται σε κάποιο από τα επίπεδα της ταξινομίας γνωστικών ικανοτήτων του Bloom (1956). Στον ακόλουθο πίνακα (Πίνακας 1) φαίνεται η προτεινόμενη αντιστοίχιση των ενεργητικών ρημάτων σε καθένα επίπεδο της ταξινομίας μαθησιακών στόχων του Bloom.

Πίνακας 1: Αντιστοιχία ρημάτων σε επίπεδα μαθησιακών στόχων Bloom (1956)

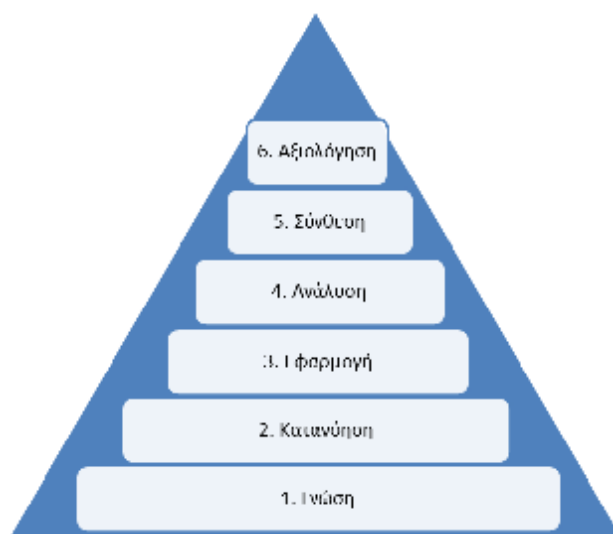
| Επίπεδο Bloom | Σύντομη Περιγραφή | Ρήματα |
|---------------|--|--|
| Γνώση | Ενέργειες που οδηγούν στην ικανότητα ανάκλησης πρόσφατα αποκτημένων γνώσεων | ανακαλώ, αναφέρω, απαγγέλω, απαριθμώ, δηλώνω, καταγράφω, ονομάζω, ορίζω, περιγράφω, ταυτίζω |
| Κατανόηση | Ενέργειες που οδηγούν στην ικανότητα σύλληψης νοημάτων | αναγνωρίζω, γενικεύω, διακρίνω, εξηγώ, επεκτείνω, επισημαίνω, μετατρέπω, παραφράζω, συνοψίζω |
| Εφαρμογή | Ενέργειες που οδηγούν στην ικανότητα χρήσης της αποκτημένης γνώσης σε νέες καταστάσεις | αλλάζω, γράφω, επιλύω, κατασκευάζω, μεταφέρω, μετρώ, προετοιμάζω, υπολογίζω, χρησιμοποιώ |
| Ανάλυση | Ενέργειες που οδηγούν στην ικανότητα διαχωρισμού των ιδεών στα επιμέρους συστατικά και ανάδειξης των μεταξύ τους σχέσεων | αναλύω, διαχωρίζω, επεξηγώ, κάνω διάγραμμα, σκιαγραφώ, συμπεραίνω, συσχετίζω |

| | | |
|------------|---|--|
| Σύνθεση | Ενέργειες που οδηγούν στην ικανότητα σύνθεσης πολλών στοιχείων σε ένα σύνολο | αναπτύσσω, δημιουργώ, επινοώ, κατηγοριοποιώ, οργανώνω, παράγω, συνδυάζω, συνθέτω, τροποποιώ |
| Αξιολόγηση | Ενέργειες που οδηγούν στην ικανότητα κρίσης της αξίας ή της ποιότητας των ενεργειών με βάση συγκεκριμένα κριτήρια | αντιπαραθέτω, αξιολογώ, δικαιολογώ, εκτιμώ, επιλέγω, κάνω κριτική, κρίνω, συγκρίνω, υποστηρίζω |

Πιο συγκεκριμένα, ο Bloom (1956) ταξινόμησε τους μαθησιακούς στόχους σε τρία βασικά επίπεδα – τομείς:

- **Γνωστικό:** που αφορά στόχους που σχετίζονται με τις διεργασίες της γνώσης και την ανάπτυξη διανοητικών δεξιοτήτων.
- **Συναισθηματικό:** που αφορά στόχους που σχετίζονται με τα συναισθήματα και τις στάσεις.
- **Ψυχοκινητικό:** που αφορά στόχους που σχετίζονται με την ανάπτυξη σωματικών δεξιοτήτων.

Επίσης, πρότεινε ταξινομήσεις των μαθησιακών στόχων για το γνωστικό και το συναισθηματικό τομέα, ιεραρχώντας τους στόχους ανάλογα με τη δυσκολία και την πολυπλοκότητά τους. Αναφορικά με το γνωστικό τομέα, διέκρινε έξι (6) κατηγορίες – επίπεδα δεξιοτήτων, τις οποίες ο εκπαιδευόμενος κατακτά διαδοχικά, ξεκινώντας από την πρώτη (απλούστερη) και συνεχίζοντας προς τις επόμενες (δυσκολότερες). Στην παρακάτω εικόνα (Εικόνα 1) φαίνονται οι κατηγορίες - επίπεδα ταξινόμησης των μαθησιακών στόχων του γνωστικού τομέα, σύμφωνα με τον Bloom.



Εικόνα 1: Επίπεδα μαθησιακών στόχων γνωστικού τομέα σύμφωνα με τον Bloom (1956)

Ακολουθεί μία σύντομη περιγραφή καθενός επιπέδου:

- **Γνώση:** το χαμηλότερο επίπεδο, που σχετίζεται με την ανάκληση των πρότερων γνώσεων. Ουσιαστικά ελέγχεται η απομνημόνευση και η δυνατότητα άρτιας παρουσίασης.
- **Κατανόηση:** σχετίζεται με την κατανόηση κάποιας έννοιας, την ερμηνεία προβλημάτων και οδηγιών και τη δήλωση ενός προβλήματος με διαφορετικές λέξεις. Ουσιαστικά ελέγχεται η δυνατότητα ερμηνείας και κατηγοριοποίησης.
- **Εφαρμογή:** σχετίζεται με την εφαρμογή – χρήση της γνώσης σε νέες καταστάσεις για την επίλυση προβλημάτων. Εξετάζεται κατά πόσο η νέα γνώση έχει κατανοηθεί σε βάθος, ώστε να αποτελεί εργαλείο για την επίλυση νέων ζητημάτων.
- **Ανάλυση:** σχετίζεται με την ανάλυση στοιχείων και σχέσεων και την εύρεση της οργανωτικής τους δομής.
- **Σύνθεση:** σχετίζεται με την κατανόηση των σχέσεων που συνδέουν κάποια συστατικά και τη σύνθεση μίας νέας δομής ή ενός νέου ενιαίου συνόλου.
- **Αξιολόγηση:** το υψηλότερο επίπεδο, που σχετίζεται με την αποτίμηση της αξίας της νέας γνώσης και τη διατύπωση αξιολογικών κρίσεων.

3 MATLAB

Στην ενότητα αυτή γίνεται αναφορά στο εργαλείο Matlab και παρουσιάζεται συνοπτικά το περιβάλλον δημιουργίας γραφικών διεπαφών GUIDE του Matlab. Τα εργαλεία αυτά χρησιμοποιήθηκαν για την ανάπτυξη εκπαιδευτικού διαδραστικού λογισμικού ασκήσεων, το οποίο σε συνδυασμό με κατάλληλα σχεδιασμένα φύλλα εργασίας στοχεύουν στην υποστήριξη της εργαστηριακής διδασκαλίας του μαθήματος της ψηφιακής επεξεργασίας σήματος. Η ανάπτυξη του υποστηρικτικού υλικού διδασκαλίας (διαδραστικό λογισμικό ασκήσεων και φύλλα εργασίας) παρουσιάζεται αναλυτικά στην ενότητα 4.

3.1 Βασικά Χαρακτηριστικά

Το MATLAB είναι μία γλώσσα υψηλού επιπέδου, που χρησιμοποιείται για τεχνικούς υπολογισμούς. Το όνομα Matlab προέρχεται από τα αρχικά των λέξεων Matrix Laboratory και η λειτουργία του βασίζεται κυρίως στην χρήση πινάκων, τα στοιχεία των οποίων μπορούν να είναι πραγματικοί ή μιγαδικοί αριθμοί. Σχεδιάστηκε με αρχικό σκοπό την υπολογιστική επίλυση απλών και σύνθετων μαθηματικών προβλημάτων, ωστόσο σήμερα χρησιμοποιείται σχεδόν σε όλους τους επιστημονικούς κλάδους. Τα προγράμματα που είναι γραμμένα σε Matlab είναι γενικά σαφή, ευανάγνωστα και εύκολα στην κατανόηση. Η γλώσσα είναι αυστηρά δομημένη στη σύνταξη των εντολών και διαθέτει όλα τα χαρακτηριστικά μίας σύγχρονης γλώσσας προγραμματισμού, η οποία υποστηρίζεται από ένα εξελιγμένο προγραμματιστικό περιβάλλον με πληθώρα βοηθητικών εργαλείων. Τα προγράμματα σε Matlab είναι ανεξάρτητα της πλατφόρμας στην οποία εκτελούνται, ενώ ένα από τα βασικά χαρακτηριστικά τους είναι το γεγονός ότι ο προγραμματιστής δεν απαιτείται να ορίζει τον τύπο των μεταβλητών. Επιπλέον, η δυνατότητα του Matlab να κάνει συμβολικές πράξεις και ο μεγάλος αριθμός έτοιμων βιβλιοθηκών που διαθέτει, το κατατάσσει στα καλύτερα λογισμικά της κατηγορίας του (Marchand & Holland, 2003).

3.2 Το Περιβάλλον GUIDE

Το MATLAB διαθέτει ένα ενσωματωμένο περιβάλλον – εργαλειοθήκη για τη σχεδίαση φιλικών στον χρήστη γραφικών διεπαφών. Το περιβάλλον αυτό, που καλείται και ως GUIDE (Graphical User Interface Design Environment), διαθέτει μία πληθώρα έτοιμων στοιχείων ελέγχου, όπως πλήκτρα, πλαίσια, μενού, άξονες, πίνακες κοκ.

Μία γραφική διεπαφή αποτελείται από διάφορα παράθυρα, που μπορούν να διαθέτουν ποικίλα στοιχεία ελέγχου, όπως πεδία κειμένου, πλήκτρα, μπάρες κύλισης κοκ. Τα παράθυρα μπορούν να καλούν το ένα το άλλο, να δέχονται είσοδο από τον χρήστη, να επικοινωνούν μεταβιβάζοντας δεδομένα μεταξύ τους και γενικότερα να επιτελούν διάφορες λειτουργίες (Lent, 2013). Προκειμένου το GUIDE να το πετύχει αυτό, για κάθε ένα παράθυρο δημιουργεί δύο αρχεία, ένα Fig - file και ένα M - file. Πιο συγκεκριμένα:

- Το Fig – file είναι το παράθυρο (figure), όπου το MATLAB αποθηκεύει τα στοιχεία ελέγχου και την ακριβή τους θέση. Ουσιαστικά, στο αρχείο αυτό ο προγραμματιστής σχεδιάζει την εμφάνιση του παραθύρου.
- Το M – file είναι το αρχείο στο οποίο ο προγραμματιστής συγγράφει τον κώδικα που αφορά τη συμπεριφορά των στοιχείων ελέγχου του παραθύρου, προκειμένου το παράθυρο να επιτελεί τις επιθυμητές λειτουργίες.

Επισημαίνεται ότι κάθε fig - file πρέπει να συνοδεύεται από ένα ομώνυμο αρχείο m – file. Αν χαθεί ή καταστραφεί το M – file, τότε το παράθυρο δε θα είναι λειτουργικό. Όταν ο χρήστης δημιουργεί ένα νέο παράθυρο, το GUIDE δημιουργεί αυτομάτως και τους δύο προαναφερθέντες τύπους αρχείων (Βαρσάμης, 2006).

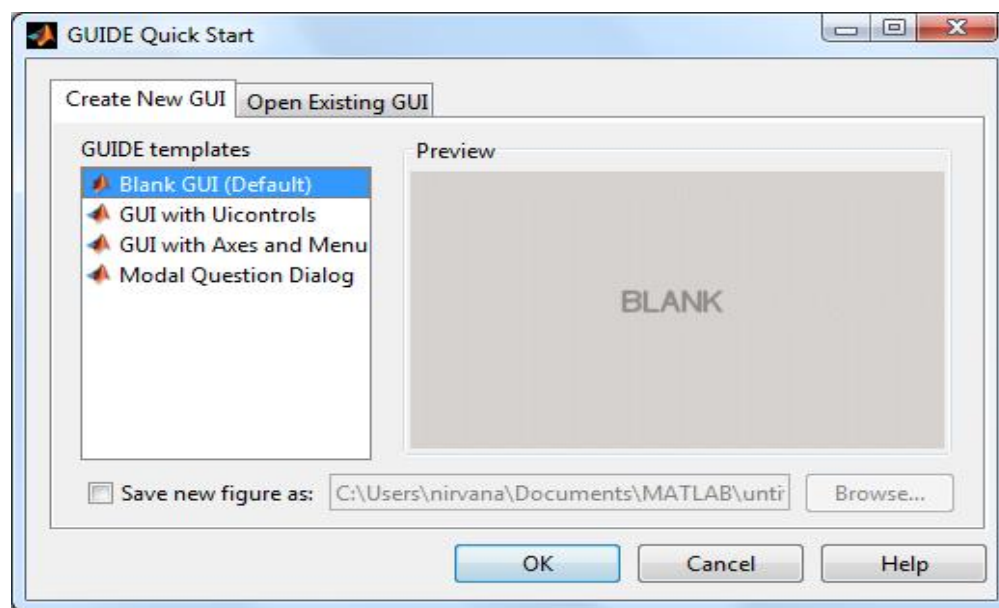
Η εκκίνηση του GUIDE γίνεται με δύο τρόπους:

1. Γράφοντας την εντολή «guide» στη γραμμή εντολών του Matlab
2. Από το μενού του Matlab «File» à «New» à «GUI»

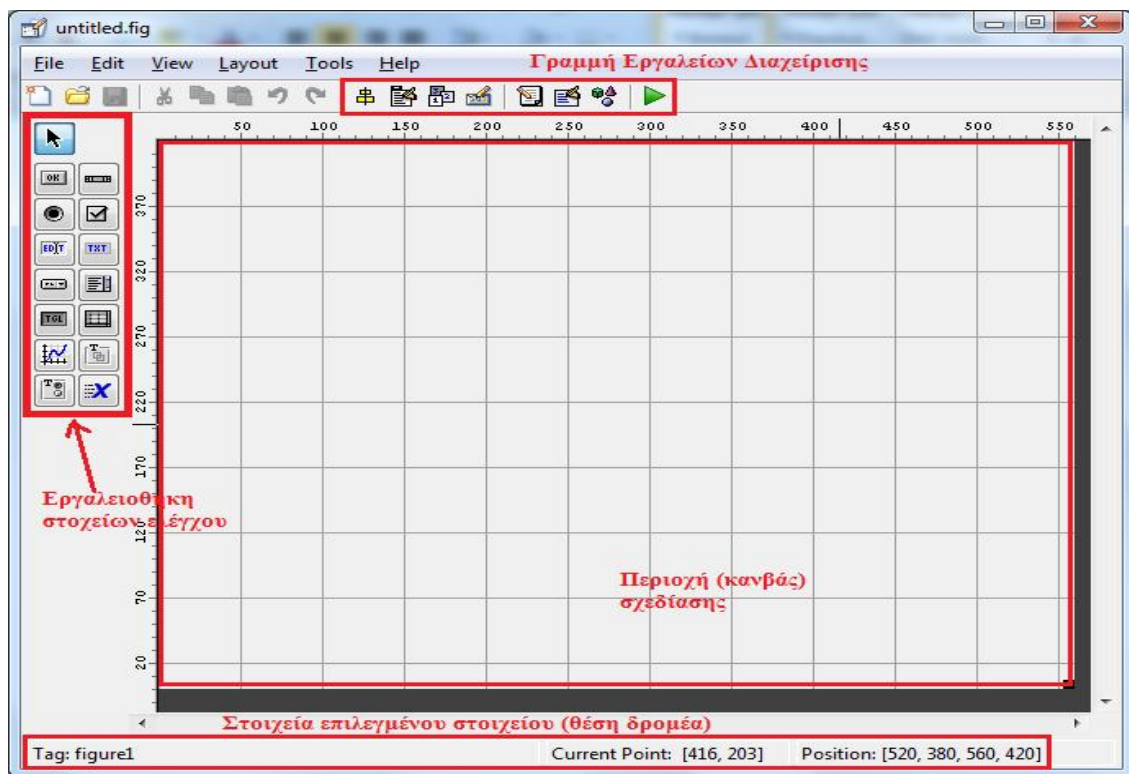
Αμέσως θα ξεκινήσει ένας οδηγός που θα καθοδηγήσει τον χρήστη στη

δημιουργία ενός παραθύρου. Αρχικά ο χρήστης θα πρέπει να επιλέξει αν θέλει να δημιουργήσει ένα κενό παράθυρο (Blank GUI) ή ένα παράθυρο βασισμένο σε κάποια πρότυπα (πχ με άξονες και μενού) ή αν θέλει να ανοίξει ένα έτοιμο παράθυρο. Μόλις ο χρήστης επιλέξει το είδος του παραθύρου που επιθυμεί, τότε θα εμφανιστεί το νέο παράθυρο στο περιβάλλον σχεδίασης του GUIDE.

Στις ακόλουθες εικόνες (Εικόνα 2 και Εικόνα 3) φαίνεται ο οδηγός δημιουργίας ενός παραθύρου του GUIDE και το περιβάλλον σχεδίασης του νέου παραθύρου. Τα βασικά τμήματα του περιβάλλοντος σχεδίασης του GUIDE έχουν κόκκινη σήμανση.

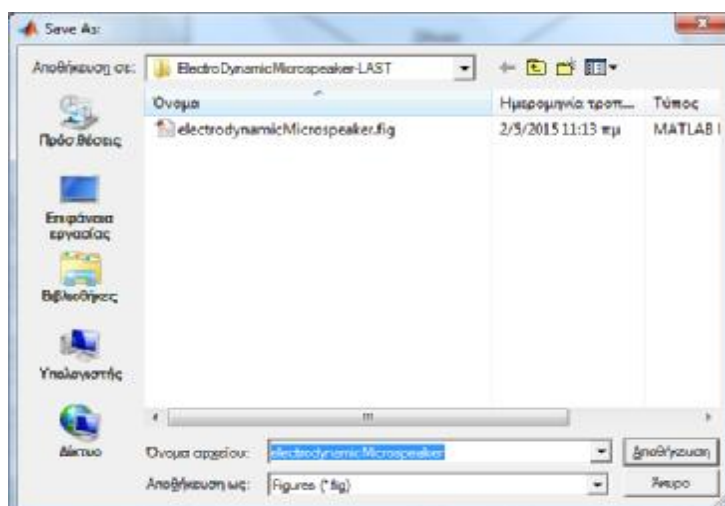


Εικόνα 2: Οδηγός δημιουργίας παραθύρου



Εικόνα 3: Περιβάλλον σχεδίασης νέου παραθύρου του GUIDE

Για την αποθήκευση του παραθύρου, ο χρήστης πρέπει να επιλέξει «File» à «Save as...», να επιλέξει τη θέση και το όνομα που θέλει να δώσει στο παράθυρο και να πατήσει το πλήκτρο «Αποθήκευση». Το Matlab θα δημιουργήσει το fig – file και το M – file και θα εμφανίσει σε ένα νέο παράθυρο τον κώδικα του M-file που αφορά το παράθυρο. Στην Εικόνα 4 φαίνεται το παράθυρο αποθήκευσης εφαρμογής GUIDE.












Εικόνα 4: Παράθυρο αποθήκευσης εφαρμογής GUIDE







Στο παράθυρο σχεδίασης του GUIDE ο προγραμματιστής μπορεί να σχεδιάσει τη γραφική διεπαφή που θα εξυπηρετεί την εφαρμογή του. Για το σκοπό αυτόν, το GUIDE διαθέτει μία εργαλειοθήκη στοιχείων ελέγχου, όπου ο προγραμματιστής μπορεί να επιλέξει και να σύρει πάνω στον κανβά σχεδίασης το επιθυμητό στοιχείο.

Στον πίνακα που ακολουθεί (Πίνακας 2) περιγράφεται η λειτουργία καθενός στοιχείου ελέγχου της εργαλειοθήκης του GUIDE. Επισημαίνεται ότι ο προγραμματιστής μπορεί να γράψει κώδικα για οποιοδήποτε στοιχείο ελέγχου, προκειμένου να καθορίσει τη συμπεριφορά του κατά την εκτέλεση της εφαρμογής.

Πίνακας 2: Στοιχεία ελέγχου της εργαλειοθήκης του GUIDE

| Εικονίδιο | Λεκτικό | Περιγραφή |
|---|-------------|---|
|  | Select | Εργαλείο επιλογής και μετακίνησης αντικειμένων. |
|  | Push Button | Δημιουργεί ένα πλήκτρο, το οποίο εκτελεί μια διαδικασία (Callback) όταν πατηθεί από τον χρήστη κατά την εκτέλεση της εφαρμογής. |
|  | Slider | Δημιουργεί μία μπάρα κύλισης, η οποία δέχεται μία αριθμητική είσοδο εντός ενός πεδίου τιμών |

| | | |
|---|--------------|---|
| | | που ορίζει ο προγραμματιστής. |
|  | Radio Button | Δημιουργεί ένα πλήκτρο τύπου radio. Ο προγραμματιστής πρέπει να φροντίζει ώστε όταν κατά την εκτέλεση της εφαρμογής ο χρήστης επιλέξει κάποιο radio button, τότε να απενεργοποιούνται τα υπόλοιπα radio buttons. Χρησιμοποιείται όταν θέλουμε να δώσουμε τη δυνατότητα στον χρήστη να επιλέξει μόνο μία επιλογή από ένα πλήθος διαθέσιμων επιλογών. |
|  | Check Box | Δημιουργεί ένα πλήκτρο τύπου check. Μοιάζει με ένα radio button, όμως όταν επιλέγεται (τσεκάρεται) από τον χρήστη κατά την εκτέλεση της εφαρμογής, δεν χρειάζεται να απενεργοποιούνται τα υπόλοιπα. |
|  | Edit Text | Δημιουργεί ένα πλαίσιο μέσα στο οποίο ο χρήστης μπορεί να εισάγει δεδομένα μέσω του πληκτρολογίου κατά την εκτέλεση της εφαρμογής. Στη συνέχεια, τα δεδομένα αυτά μπορεί να τα αξιοποιήσει η εφαρμογή. |
|  | Static Text | Επιτρέπει τη δημιουργία στατικών κειμένων, για την παροχή πληροφοριών προς τον χρήστη κατά την εκτέλεση της εφαρμογής. |
|  | Pop-Up Menu | Δημιουργεί μία αναπτυσσόμενη λίστα επιλογών. |
|  | List Box | Δημιουργεί μία απλή λίστα επιλογών. |

| | | |
|---|-----------------|--|
|  | Toggle Button | Δημιουργεί ένα πλήκτρο τύπου «toggle». Μοιάζει με ένα απλό πλήκτρο (push button) με τη διαφορά ότι μπορεί να έχει δύο τιμές, τις «On» και «Off». |
|  | Table | Χρησιμοποιείται για την οργάνωση και την παρουσίαση δεδομένων σε μορφή πίνακα. |
|  | Axes | Επιτρέπει την τοποθέτηση γραφικών στοιχείων, όπως εικόνων και γραφημάτων. |
|  | Panel | Χρησιμοποιείται για την ομαδοποίηση πολλών εργαλείων. |
|  | Button Group | Χρησιμοποιείται για την ομαδοποίηση πολλών radio buttons ή toggle buttons. |
|  | ActiveX Control | Επιτρέπει την προσθήκη και εμφάνιση ActiveX χειριστηρίων. Χρησιμοποιείται μόνο στα Windows. |

Επιπλέον, το περιβάλλον του GUIDE διαθέτει μία γραμμή εργαλείων διαχείρισης, η οποία παρέχει άμεση πρόσβαση σε χρήσιμα εργαλεία. Η γραμμή των εργαλείων διαχείρισης φαίνεται στην Εικόνα 5.











Εικόνα 5: Γραμμή εργαλείων διαχείρισης του GUIDE

Στον πίνακα που ακολουθεί (Πίνακας 3) περιγράφεται η λειτουργία καθενός στοιχείου της γραμμής εργαλείων διαχείρισης του GUIDE.

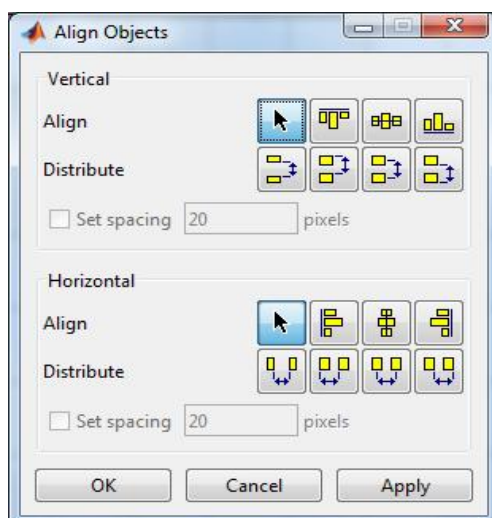
Πίνακας 3: Στοιχεία γραμμής εργαλείων διαχείρισης του GUIDE

| Εικονίδιο | Λεκτικό | Περιγραφή |
|-----------|---------|-----------|
|-----------|---------|-----------|

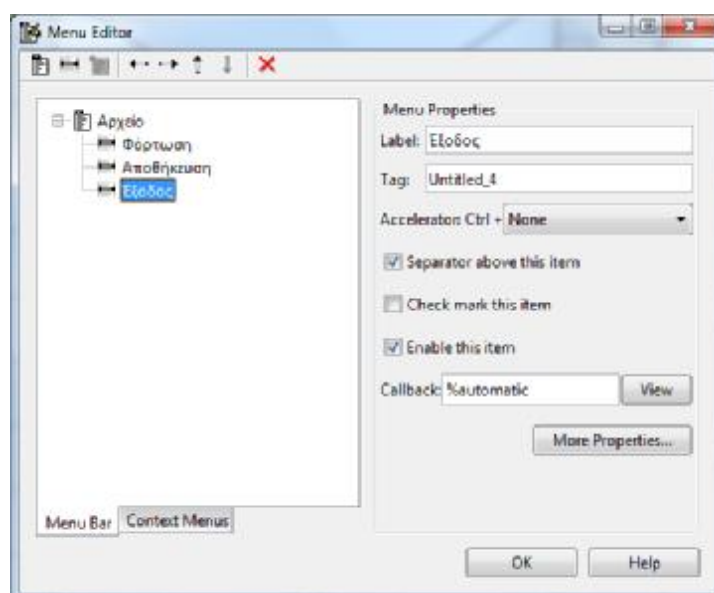
| | | |
|---|--------------------|--|
|  | Align Objects | Εργαλείο για τη στοίχιση των αντικειμένων – στοιχείων ελέγχου στο παράθυρο της εφαρμογής. |
|  | Menu Editor | Επιτρέπει τη δημιουργία και διαχείριση ενός βασικού μενού για το παράθυρο της εφαρμογής. Επιπλέον, επιτρέπει και τη δημιουργία αναπτυσσόμενων μενού, που ενεργοποιούνται όταν ο χρήστης κάνει δεξί κλικ κατά την εκτέλεση της εφαρμογής. |
|  | Tab Order Editor | Χρησιμεύει για τον ορισμό της σειράς με την οποία τα στοιχεία ελέγχου του παραθύρου αποκτούν την εστίαση, όταν ο χρήστης πατάει το πλήκτρο «Tab» κατά την εκτέλεση της εφαρμογής. |
|  | Toolbar Editor | Επιτρέπει τη δημιουργία γραμμής εργαλείων στο παράθυρο της εφαρμογής. |
|  | Editor | Ανοίγει το παράθυρο επεξεργασίας του κώδικα του παραθύρου, ο οποίος βρίσκεται στο ομώνυμο M – file. |
|  | Property Inspector | Πολύ χρήσιμο εργαλείο για τον προγραμματιστή, αφού παρέχει άμεση πρόσβαση στις ιδιότητες του επιλεγμένου στοιχείου ελέγχου. Οι ιδιότητες αυτές εμφανίζονται σε ένα παράθυρο. |
|  | Object Browser | Επίσης πολύ χρήσιμο εργαλείο για τον προγραμματιστή, αφού εμφανίζει την ιεραρχία των στοιχείων ελέγχου που έχει τοποθετήσει στο παράθυρο της εφαρμογής. |

| | | |
|---|------------|------------------------------------|
|  | Run Figure | Εκτελεί το παράθυρο της εφαρμογής. |
|---|------------|------------------------------------|

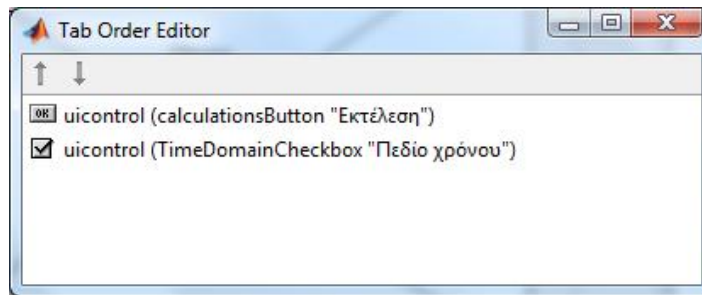
Επίσης, στις παρακάτω εικόνες (Εικόνα 6 ως Εικόνα 12) παρουσιάζονται τα παράθυρα που αντιστοιχούν στα στοιχεία της γραμμής εργαλείων διαχείρισης του GUIDE.



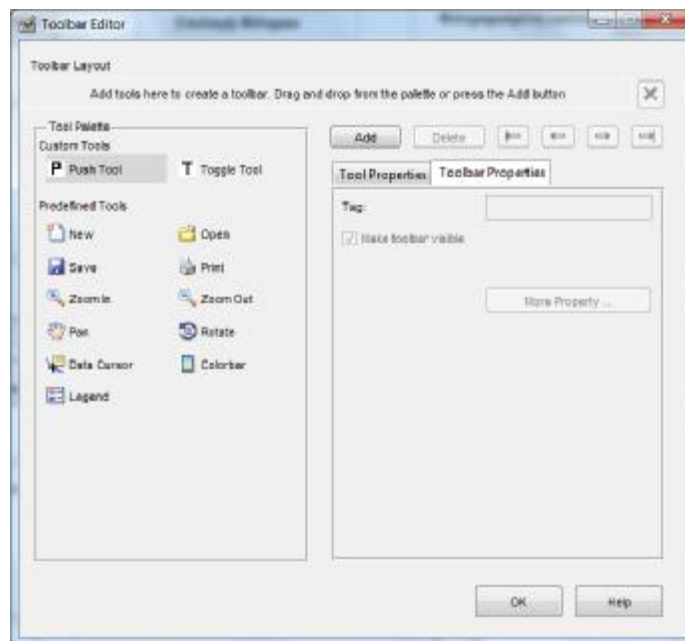
Εικόνα 6: Παράθυρο Align Objects του GUIDE



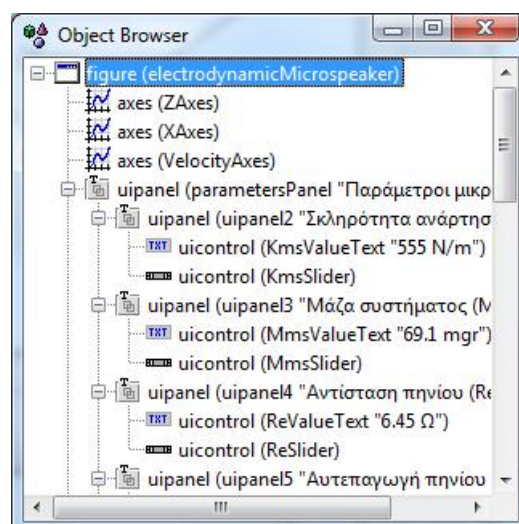
Εικόνα 7: Παράθυρο Menu Editor του GUIDE



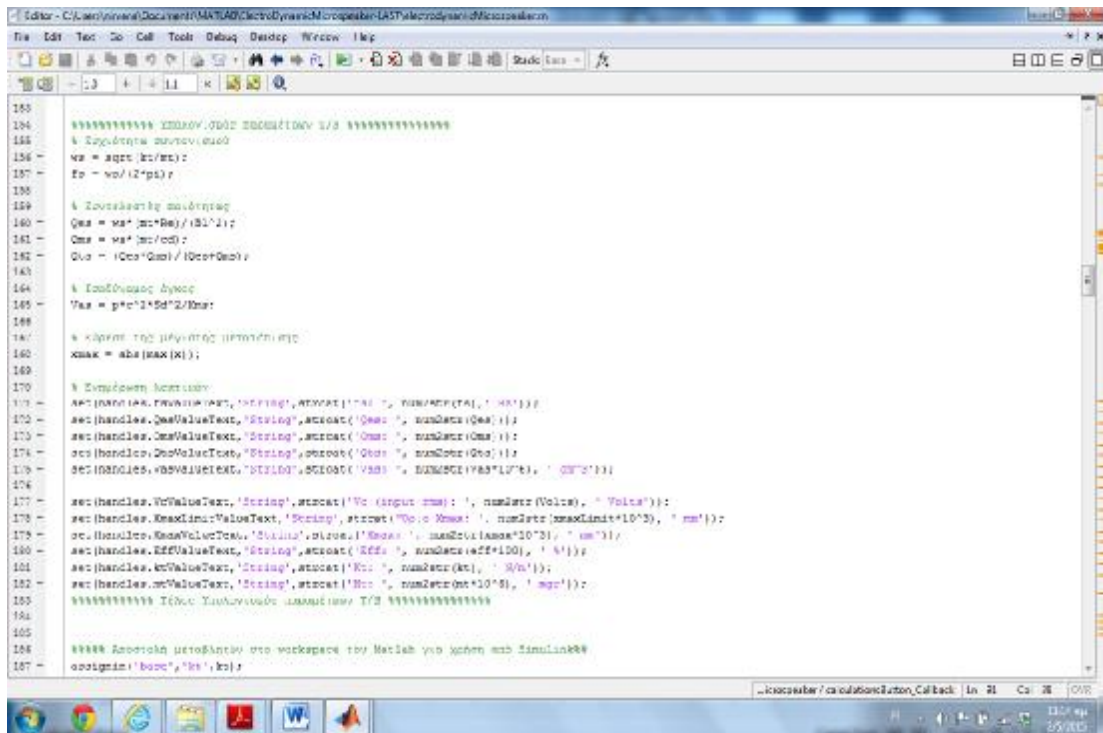
Εικόνα 8: Παράθυρο Tab Order Editor του GUIDE



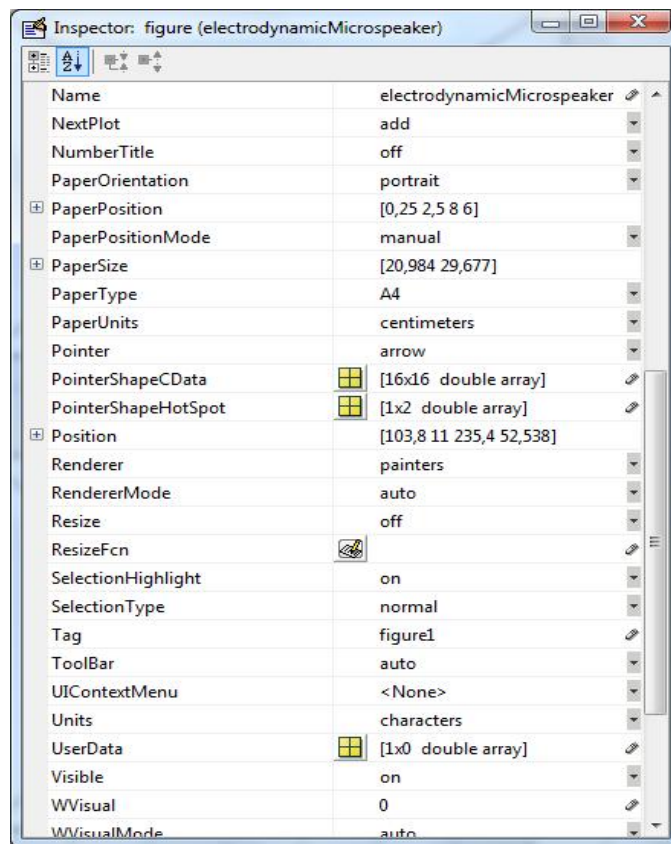
Εικόνα 9: Παράθυρο Toolbar Editor του GUIDE



Εικόνα 10: Παράθυρο Object Browser του GUIDE



Εικόνα 11: Παράθυρο Editor του GUIDE



Εικόνα 12: Παράθυρο Property Inspector του GUIDE

4 Σχεδίαση και Ανάπτυξη του Υποστηρικτικού Υλικού

Το υποστηρικτικό υλικό που αναπτύχθηκε στο πλαίσιο της παρούσας εργασίας αφορά τη διδασκαλία του εργαστηριακού μέρους του μαθήματος «Ψηφιακή Επεξεργασία Σήματος», το οποίο καλύπτει δεκατρείς (13) ενότητες διδασκαλίας. Για καθεμία από τις ενότητες αυτές αναπτύχθηκε μία εργαστηριακή άσκηση σε μορφή έντυπου φύλλου εργασίας, με καλά ορισμένους μαθησιακούς στόχους. Επίσης, για καθεμία ενότητα αναπτύχθηκε μία διαδραστική διεπαφή σε Matlab (ψηφιακό υλικό), η οποία μπορεί να χρησιμοποιηθεί για την εκπόνηση της αντίστοιχης εργαστηριακής άσκησης χρησιμοποιώντας το φύλλο εργασίας της άσκησης. Το σύνολο των διεπαφών είναι προσβάσιμο μέσω μίας κεντρικής διεπαφής – οθόνης, από όπου ο χρήστης μπορεί να επιλέξει την εργαστηριακή άσκηση που θέλει να εκτελέσει.

Ο βασικός σκοπός ανάπτυξης του υποστηρικτικού υλικού είναι η πιο ευχάριστη και ξεκούραστη παρουσίαση της ύλης, επιτρέποντας την ευκολότερη κατανόηση και την εμπάθυνση στις βασικές έννοιες και τεχνικές της ψηφιακής επεξεργασίας σήματος. Οι δομημένες εργαστηριακές ασκήσεις, καθώς και το αντίστοιχο διαδραστικό λογισμικό ασκήσεων σε Matlab, προάγουν την ενεργητική και ανακαλυπτική μάθηση, αφού επιτρέπουν στον χρήστη να πειραματιστεί με το λογισμικό εισάγοντας διαφορετικά δεδομένα κάθε φορά και μελετώντας τα αποτελέσματα της εκτέλεσης, τα οποία εμφανίζονται με χρήση πινάκων και γραφικών παραστάσεων.

4.1 Περίγραμμα Ύλης

Ακολουθεί το περίγραμμα της ύλης, την οποία καλύπτει το υλικό που αναπτύξαμε για την υποστήριξη της διδασκαλίας δεκατριών (13) θεματικών ενοτήτων του εργαστηρίου του μαθήματος της Ψηφιακής Επεξεργασίας Σήματος:

- 1) **Εισαγωγή στο MATLAB**
- 2) **Σήματα διακριτού χρόνου (ΣΔΧ) – Παραγωγή και πράξεις μεταξύ ΣΔΧ**
 - Παραγωγή θεμελιωδών ΣΔΧ
 - κρουστικός παλμός

- μοναδιαίο βήμα
 - μοναδιαία κλίση
 - πραγματικό εκθετικό σήμα
 - μιγαδικό εκθετικό σήμα
 - ημιτονοειδές σήμα – η έννοια της συχνότητας
 - Πράξεις μεταξύ σημάτων
 - πρόσθεση σημάτων
 - πολλαπλασιασμός σημάτων
- 3) Σήματα Διακριτού Χρόνου - Μετασχηματισμοί της ανεξάρτητης μεταβλητής – Χαρακτηριστικά μεγέθη ΣΔΧ**
- Μετασχηματισμοί της ανεξάρτητης μεταβλητής
 - χρονική μετατόπιση
 - αντιστροφή
 - κλιμάκωση στο χρόνο (πολλαπλασιασμός και διαίρεση συχνότητας)
 - Χαρακτηριστικά μεγέθη ΣΔΧ
 - ενέργεια
 - στιγμιαία και μέση ισχύς
- 4) Συστήματα Διακριτού Χρόνου – Ταξινόμηση συστημάτων**
- Ταξινόμηση συστημάτων
 - αιτιατά
 - στατικά
 - χρονικά αμετάβλητα
 - ομογενή
 - ευσταθή
 - αντιστρέψιμα
 - Χαρακτηριστικά και ιδιότητες συστημάτων
- 5) Συστήματα Διακριτού Χρόνου - Συνέλιξη**
- Ορισμός και υπολογισμός συνέλιξης
 - Ιδιότητες συνέλιξης
- 6) Συστήματα Διακριτού Χρόνου – Εξισώσεις διαφορών**

- Επίλυση Γραμμικών Διαφορικών Εξισώσεων με Σταθερούς Συντελεστές
- 7) **Μετασχηματισμός Fourier διακριτού χρόνου (DTFT)**
- Υπολογισμός ευθύ και αντίστροφου μετασχηματισμού
 - Υπολογισμός κυκλικής συνέλιξης
- 8) **Απόκριση Συχνότητας**
- Υπολογισμός απόκρισης συχνότητας με χρήση DTFT
 - Ψηφιακά Φίλτρα
 - Διασύνδεση συστημάτων
- 9) **Διακριτός μετασχηματισμός Fourier (DFT)**
- DFT και FFT
 - Υπολογισμός κυκλικής συνέλιξης
 - Υπολογισμός γραμμικής συνέλιξης με DFT
- 10) **Μετασχηματισμός Z**
- Ορισμός και ιδιότητες μετασχηματισμού Z
 - Περιοχή σύγκλισης
 - Ανάπτυξη συστήματος σε μερικά κλάσματα
- 11) **Μετασχηματισμός Z**
- Συνάρτηση μεταφοράς
 - Μελέτη ευστάθειας με διάγραμμα πόλων - μηδενικών
- 12) **Ψηφιακά Φίλτρα**
- Σχεδίαση ψηφιακών φίλτρων
 - Φίλτρα FIR
 - Φίλτρα IIR
- 13) **Μετατροπή σήματος από Αναλογική σε Ψηφιακή Μορφή**
- Περιοδική δειγματοληψία
 - Κβαντισμός και κωδικοποίηση

4.2 Έντυπο Υλικό – Φύλλα Εργασίας Εργαστηριακών Ασκήσεων

Καθεμία από τις δεκατρείς (13) ενότητες διδασκαλίας καλύπτεται από μία εργαστηριακή άσκηση διάρκειας περίπου 2 ωρών. Για την καθεμία εργαστηριακή

άσκηση σχεδιάστηκε ένα φύλλο εργασίας με κατάλληλες δραστηριότητες, ακολουθώντας τις προδιαγραφές που αναφέρθηκαν στην ενότητα 2.

Κάθε φύλλο εργασίας έχει την εξής δομή:

- Εισαγωγικά στοιχεία
 - Σκοπός
 - Αναμενόμενα μαθησιακά αποτελέσματα (μαθησιακοί στόχοι)
 - Εκτιμώμενη διάρκεια
 - Προαπαιτούμενα
 - Εισαγωγή
 - Κατάλογος χρησικών συναρτήσεων Matlab
- Δραστηριότητες – Πρότυπες ασκήσεις
 - Σύντομη θεωρία
 - Ασκήσεις – Παραδείγματα για λύση στο Matlab
 - Λύσεις των ασκήσεων με τον κώδικα Matlab και την έξοδο που θα εμφανιστεί όταν θα εκτελεστεί ο κώδικας
- Άλυτες ασκήσεις
- Επίλογος – Σύνδεση με τα επόμενα
- Βιβλιογραφία

Ακολουθούν τα φύλλα εργασίας που αναπτύχθηκαν για τις εργαστηριακές ασκήσεις του μαθήματος.

Εργαστηριακή Άσκηση 1: Εισαγωγή στο Matlab

1 Εισαγωγικά Στοιχεία

1.1 Σκοπός

Σκοπός της παρούσας εργαστηριακής άσκησης είναι η εξοικείωση των φοιτητών – σπουδαστών με το περιβάλλον του Matlab και τις βασικές αρχές που το διέπουν. Για το σκοπό αυτό γίνεται μία εκτενής αναφορά στις βασικές έννοιες και εντολές – λειτουργίες του, στους τρόπους διαχείρισης και απεικόνισης των δεδομένων, καθώς και στους τρόπους διαχείρισης της ροής εκτέλεσης των εντολών.

1.2 Αναμενόμενα Μαθησιακά Αποτελέσματα

Με την ολοκλήρωση της παρούσας εργαστηριακής άσκησης, ο φοιτητής – σπουδαστής θα είναι σε θέση να:

- Αναγνωρίζει τα βασικά μέρη του περιβάλλοντος εργασίας του Matlab
- Γράφει εντολές στο παράθυρο εντολών
- Χρησιμοποιεί το περιβάλλον βοήθειας του Matlab
- Εκτελεί αριθμητικές πράξεις και να γράφει σύνθετες αριθμητικές εκφράσεις
- Χρησιμοποιεί μεταβλητές για την καταχώρηση των αποτελεσμάτων και να τους δίνει αντιπροσωπευτικά ονόματα
- Παρατηρεί και διαχειρίζεται το workspace του Matlab
- Κατασκευάζει και διαχειρίζεται τους μιγαδικούς αριθμούς
- Χειρίζεται τις βασικές μαθηματικές συναρτήσεις
- Κατασκευάζει και να διαχειρίζεται σχηματισμούς αριθμών, όπως διανύσματα και πίνακες αριθμών
- Δημιουργεί τα δικά του script – files (M – files) και να κατανοεί τα πλεονεκτήματα χρήσης τους
- Εκτελεί λογικές και σχεσιακές πράξεις και να γράφει σύνθετες εκφράσεις
- Ελέγχει τη ροή εκτέλεσης των εντολών, χρησιμοποιώντας βρόχους επανάληψης με τις εντολές for και while

- Ελέγχει τη ροή εκτέλεσης των εντολών, χρησιμοποιώντας δομές ελέγχου με τις εντολές `if – else` και `switch – case`
- Δημιουργεί απλές γραφικές παραστάσεις για συναρτήσεις συνεχών και διακριτών μεταβλητών

1.3 Εκτιμώμενη Διάρκεια

Ο εκτιμώμενος χρόνος που απαιτείται από τον φοιτητή – σπουδαστή για την ολοκλήρωση της παρούσας εργαστηριακής άσκησης είναι 2 – 4 διδακτικές ώρες.

1.4 Προαπαιτούμενα

Εγκατάσταση του περιβάλλοντος Matlab έκδοσης τουλάχιστον R2012a.

1.5 Εισαγωγή

Στην άσκηση αυτή θα πραγματοποιήσουμε μία εκτενή αναφορά στις βασικές έννοιες και αρχές του Matlab. Πιο συγκεκριμένα θα καλυφθούν τα εξής θέματα:

- Περιβάλλον εργασίας του Matlab
- Αριθμητικές πράξεις, μεταβλητές και το `workspace`
- Μιγαδικοί αριθμοί
- Μαθηματικές συναρτήσεις
- Σχηματισμοί αριθμών: διανύσματα γραμμής και στήλης, πίνακες
- `M – files`
- Λογικές και σχεσιακές πράξεις
- Έλεγχος ροής με τις εντολές επανάληψης `for`, `while` και δομές ελέγχου `if – else` και `switch – case`
- Γραφικές παραστάσεις

Η παρούσα εργαστηριακή άσκηση θα αποτελέσει χρήσιμο βοήθημα στη διάρκεια εκπόνησης όλων των εργαστηριακών ασκήσεων. Η εκτέλεση των παραδειγμάτων που παρουσιάζονται αποτελεί προαπαιτούμενο, ούτως ώστε οι φοιτητές – σπουδαστές να αποκτήσουν την απαιτούμενη γνώση και εξοικείωση για την επιτυχή

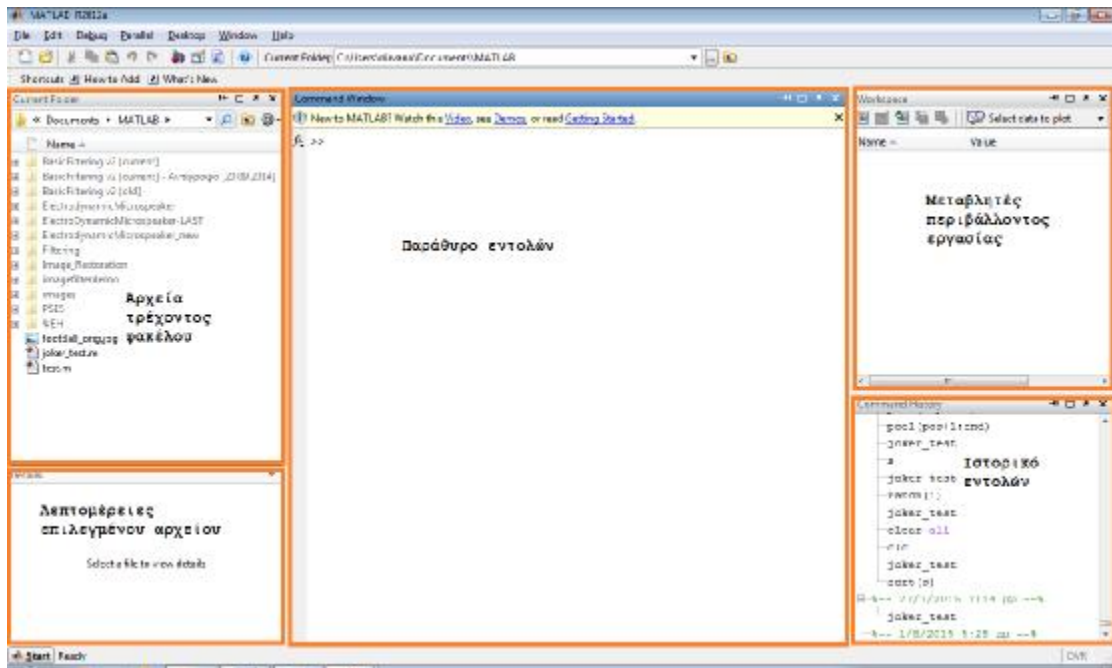
επίτευξη των αναμενόμενων μαθησιακών στόχων του εργαστηριακού μαθήματος της Ψηφιακής Επεξεργασίας Σήματος.

2 Περιβάλλον Εργασίας

Με την εκκίνηση του Matlab ανοίγει το παράθυρο εντολών (Command Window) που είναι το κύριο μέσο επικοινωνίας με το πρόγραμμα. Στην οθόνη εμφανίζεται το προτροπικό του Matlab (>>), αναμένοντας την πληκτρολόγηση εντολών από τον χρήστη.

Στην εικόνα που ακολουθεί φαίνεται το περιβάλλον εργασίας του Matlab, που χωρίζεται στις εξής περιοχές - παράθυρα:

1. Παράθυρο εντολών: Στο παράθυρο αυτό ο χρήστης πληκτρολογεί τις εντολές προς εκτέλεση
2. Χώρος εργασίας: Εμφανίζει τις μεταβλητές που είναι αποθηκευμένες στον χώρο εργασίας και είναι άμεσα προσβάσιμες
3. Ιστορικό εντολών: Εμφανίζει ιστορικό με όλες τις εντολές που έχει εισάγει ο χρήστης
4. Τρέχων φάκελος: Εμφανίζει τα αρχεία του φακέλου στον οποίο βρίσκεται ο χρήστης
5. Λεπτομέρειες: Εμφανίζει τις λεπτομέρειες σχετικά με το αρχείο που έχει επιλεγμένο ο χρήστης.



Μερικές από τις πιο κοινές εντολές φαίνονται στον ακόλουθο πίνακα.

| Εντολή | Λειτουργία – Περιγραφή |
|-------------------------|---|
| >> exit ή >> quit | Έξοδος από το περιβάλλον του Matlab |
| >> help | Εμφάνιση καταλόγου με όλα τα θέματα για τα οποία παρέχεται βοήθεια |
| >> help <i>topic</i> | Εμφάνιση βοήθειας για το θέμα με όνομα <i>topic</i> |
| >> help <i>function</i> | Εμφάνιση βοήθειας για τη συνάρτηση με όνομα <i>function</i> |
| >> demo | Έναρξη του ενσωματωμένου περιβάλλοντος επίδειξης των χαρακτηριστικών – δυνατοτήτων του Matlab |

Η ανάκληση των πρόσφατων εντολών γίνεται με τα πλήκτρα ↑ και ↓.

3 Απλές Αριθμητικές Πράξεις – Μεταβλητές – Workspace

Το Matlab προκειμένου να υποστηρίξει τις μαθηματικές πράξεις διαθέτει τα σύμβολα (τελεστές) που φαίνονται στον πίνακα που ακολουθεί:

| Σύμβολο (τελεστής) | Πράξη |
|--------------------|-----------------|
| ^ | Ύψωση σε δύναμη |
| * | Πολλαπλασιασμός |
| / ή \ | Διαίρεση |
| + | Πρόσθεση |
| - | Αφαίρεση |

Συνδυάζοντας τα προηγούμενα σύμβολα μπορούμε κατασκευάσουμε όσο πολύπλοκες εκφράσεις επιθυμούμε. Ωστόσο, η σειρά με την οποία εκτελούνται οι πράξεις υπόκειται στους τυπικούς κανόνες προτεραιότητας. Πιο συγκεκριμένα:

- Οι πράξεις εκτελούνται από αριστερά προς τα δεξιά.
- Την μεγαλύτερη προτεραιότητα έχει η πράξη της ύψωσης σε δύναμη, ενώ ακολουθούν ο πολλαπλασιασμός και η διαίρεση με ίδια προτεραιότητα μεταξύ τους και τελευταίες εκτελούνται οι πράξεις πρόσθεσης και αφαίρεσης με ίδια προτεραιότητα μεταξύ τους.
- Αν θέλουμε να καθορίσουμε εμείς τη σειρά των πράξεων, μπορούμε να χρησιμοποιήσουμε παρενθέσεις. Σε περίπτωση πολλαπλών παρενθέσεων οι πράξεις γίνονται από μέσα προς τα έξω.

Ακολουθούν μερικά παραδείγματα αριθμητικών εκφράσεων και τα αποτελέσματα που επιστρέφει το Matlab.

| Κώδικας Matlab | Αποτέλεσμα |
|----------------|-------------------|
| >> 53 +78 | ans = 131 |
| >> 127*9721 | ans = 1234567 |
| >> 34^56 | ans = 5.7919e+085 |
| >> 2/4+24/144 | ans = 0.6667 |
| >> 635-789 | ans = -154 |

Παρατηρούμε ότι οποιαδήποτε αριθμητική έκφραση και να γράψουμε, μόλις πατήσουμε το πλήκτρο ENTER, το Matlab επιστρέφει μία απάντηση η οποία καταχωρείται σε μία μεταβλητή με το όνομα ans. Τη μεταβλητή αυτή, που ανανεώνεται σε κάθε πράξη, μπορούμε να την χρησιμοποιήσουμε στην επόμενη εντολή είτε για να δούμε απλά το περιεχόμενό της είτε για να την χρησιμοποιήσουμε σε νέες πράξεις.

| Κώδικας Matlab | Αποτέλεσμα |
|---------------------|----------------|
| >> 3*2 +8*52-23/345 | ans = 421.9333 |
| >> ans | ans = 421.9333 |
| >> ans*2 | ans = 843.8667 |

Η ans είναι ένα παράδειγμα χρήσης μεταβλητών στο Matlab. Μπορούμε δηλαδή να αποθηκεύσουμε πληροφορίες σε κάποιο όνομα και να τις ανακαλούμε όταν χρειάζεται, απλά γράφοντας το όνομα αυτό.

Στον κώδικα που ακολουθεί, αρχικά δημιουργούμε τις μεταβλητές apples, oranges, peaches και στη συνέχεια ορίζουμε τις μεταβλητές no_of_fruits, cost και average_cost χρησιμοποιώντας τις 3 πρώτες μεταβλητές.

| Κώδικας Matlab | Αποτέλεσμα |
|--|------------------------|
| >> apples=13 | apples = 13 |
| >> oranges=22 | oranges = 22 |
| >> peaches=8 | peaches = 8 |
| >> no_of_fruits=apples+oranges+peaches | no_of_fruits = 43 |
| >> cost=27*apples+32*oranges+25*peaches | cost = 1255 |
| >> average_cost=cost/no_of_fruits | average_cost = 29.1860 |

Για τα ονόματα που μπορούμε να δώσουμε στις μεταβλητές υπάρχουν οι εξής κανόνες:

- Στα ονόματα των μεταβλητών γίνεται διάκριση των πεζών και κεφαλαίων γραμμάτων. Τα ονόματα `cost`, `Cost`, `Cost`, `COST` αναφέρονται σε διαφορετικές μεταβλητές.
- Οι χαρακτήρες στο όνομα μίας μεταβλητής μπορούν να είναι ως 31. Οι παραπάνω χαρακτήρες αγνοούνται.
- Τα ονόματα των μεταβλητών πρέπει να αρχίζουν από λατινικό γράμμα, ενώ οι επόμενοι χαρακτήρες μπορούν να είναι οποιοσδήποτε συνδυασμός λατινικών γραμμάτων, ψηφίων και συμβόλων υπογράμμισης (`_`).
- Τα ειδικά σύμβολα, σημεία στίξης και τα κενά απαγορεύονται στα ονόματα των μεταβλητών.

Το Matlab διαθέτει ορισμένες **ειδικές μεταβλητές**. Αυτές είναι: `ans`, `pi`, `eps`, `flops`, `inf`, `nan`, `i`, `j`, `nargin`, `nargout`, `realmin`, `realmax`. Από αυτές αναφέρουμε προς το παρόν, την `ans` η οποία αποθηκεύει το αποτέλεσμα της κάθε πράξης που εκτελούμε, την `pi` = π = 3.1416..., την `inf` που αναπαριστά το άπειρο, την `nan` που σημαίνει not-a-number και χρησιμοποιείται για τις απροσδιόριστες μορφές (πχ 0/0) και τις $i = j = \sqrt{-1}$.

Οι μεταβλητές διατηρούνται στο λεγόμενο Matlab Workspace και μπορούν να ανακληθούν όποτε το θελήσουμε. Αν θέλουμε να δούμε το περιεχόμενο μίας μεταβλητής αρκεί να γράψουμε το όνομά της.

| Κώδικας Matlab | Αποτέλεσμα |
|------------------------------|--------------------------|
| <code>>> apples</code> | <code>apples = 13</code> |

Αν δεν θυμόμαστε το όνομα μίας μεταβλητής, η εντολή `who` εμφανίζει τα ονόματα όλων των μεταβλητών.

| Κώδικας Matlab | Αποτέλεσμα |
|---------------------------|--|
| <code>>> who</code> | Your variables are: <code>ans</code> <code>average_cost</code> <code>no_of_fruits</code> <code>peaches</code> <code>apples</code> <code>cost</code> <code>oranges</code> |

Η εντολή `clear` διαγράφει όλες τις μεταβλητές, η εντολή `clear apples` διαγράφει μόνο τη μεταβλητή `apples`, ενώ η `clear a*` διαγράφει όλες τις μεταβλητές που αρχίζουν από `a`.

Ακολουθούν μερικά σύμβολα που έχουν ειδική λειτουργία όταν εμφανίζονται μέσα σε εντολές.

Το σύμβολο `%` δηλώνει ότι το κείμενο που ακολουθεί είναι σχόλιο και αγνοείται από το Matlab.

| Κώδικας Matlab | Αποτέλεσμα |
|---|--------------------------|
| <code>>> apples=13 % Αριθμός μήλων</code> | <code>apples = 13</code> |

Σε μία γραμμή μπορούμε να γράψουμε πολλές εντολές, αρκεί να τις χωρίσουμε με κόμμα (,) ή ελληνικό ερωτηματικό (;). Επίσης, το κόμμα (,) επιτρέπει την εμφάνιση των αποτελεσμάτων, ενώ το ελληνικό ερωτηματικό (;) την εμποδίζει.

| Κώδικας Matlab | Αποτέλεσμα |
|---|---|
| <code>>> apples=18,oranges=33;peaches=62</code> | <code>apples = 18</code> <code>peaches = 62</code> |

Επίσης, η 3 διαδοχικές τελείες (...) δηλώνουν ότι μία έκφραση συνεχίζεται στην επόμενη γραμμή. Ωστόσο, δεν επιτρέπεται να χωρίσουμε το όνομα μεταβλητής ή τα σχόλια σε δύο γραμμές.

| Κώδικας Matlab | Αποτέλεσμα |
|--|-------------------------------------|
| <code>>> average_cost=cost/... no_of_fruist</code> | <code>average_cost = 27.3628</code> |

4 Μιγαδικοί Αριθμοί

Το Matlab αναγνωρίζει αυτόματα τους μιγαδικούς αριθμούς όποτε αυτοί προκύπτουν. Παρακάτω παραθέτονται μερικά παραδείγματα μιγαδικών αριθμών.

| Κώδικας Matlab | Αποτέλεσμα |
|---|-------------------------|
| >> z=1-2i % το i είναι το φανταστικό μέρος | z = 1.0000 - 2.000i |
| >> w=1-2j % το j είναι το φανταστικό μέρος | w = 2.0000 - 1.000i |
| >> z1=3*(2-sqrt(-1)*3) % Είναι sqrt(-1) = i = j | z1 = 6.0000 - 9.000i |
| >> z2=sqrt(-2) | z2 = 0 + 1.4142i |
| >> z3=6+sin(0.5)*i | z3 = 6.0000 + 0.4794i |
| >> z4=(z3+w)/z1 | z4 = 0.4759 + 0.4605i |
| >> i^2 | ans = -1.0000 + 0.0000i |
| >> mag_z4=abs(z4) | mag_z4 = 0.6622 |
| >> angle_z4=angle(z4) | angle_z4 = 0.7689 |
| >> deg_z4=angle_z4*180/pi % μετατροπή σε μοίρες | deg_z4 = 44.0543 |
| >> real_z4=real(z4) | real_z4 = 0.4759 |
| >> imag_z4=imag(z4) | imag_z4 = 0.4605 |

Στα προηγούμενα παραδείγματα εκτός από τις απλές πράξεις χρησιμοποιήσαμε και μερικές συναρτήσεις, όπως την `sqrt` (τετραγωνική ρίζα) και `sin` (ημίτονο). Η συνάρτηση `abs` υπολογίζει το μέτρο – πλάτος ενός μιγαδικού αριθμού ή απλά την απόλυτη τιμή όταν ο αριθμός είναι πραγματικός. Η `angle` υπολογίζει τη γωνία του μιγαδικού σε ακτίνια, ενώ οι `real` και `imag` υπολογίζουν αντίστοιχα το πραγματικό και φανταστικό μέρος ενός μιγαδικού αριθμού.

5 Μαθηματικές Συναρτήσεις

Το Matlab εκτός από τις απλές πράξεις έχει τη δυνατότητα να υπολογίζει τιμές για έναν μεγάλο αριθμό βασικών μαθηματικών συναρτήσεων. Με την εντολή `help elfun`

εμφανίζεται κατάλογος με όλες τις συναρτήσεις αυτές, οι κυριότερες από τις οποίες φαίνονται στον ακόλουθο πίνακα.

| Συνάρτηση | Περιγραφή | Κώδικας Matlab | Αποτέλεσμα |
|-----------|---|---------------------|---------------|
| sin(x) | Ημίτονο (x σε ακτίνια) | >> sin(pi/4) | ans = 0.7071 |
| asin(x) | Αντίστροφο ημίτονο (x σε ακτίνια) | >> asin(0.7071) | ans = 0.7854 |
| sind(x) | Ημίτονο (x σε μοίρες) | >> sind(45) | ans = 0.7071 |
| asind(x) | Αντίστροφο ημίτονο (x σε μοίρες) | >> asind(0.7071) | ans = 44.9995 |
| cos(x) | Συνημίτονο (x σε ακτίνια) | >> cos(pi/4) | ans = 0.7071 |
| acos(x) | Αντίστροφο συνημίτονο (x σε ακτίνια) | >> acos(0.7071) | ans = 0.7854 |
| cosd(x) | Συνημίτονο (x σε μοίρες) | >> cosd(45) | ans = 0.7071 |
| acosd(x) | Αντίστροφο συνημίτονο (x σε μοίρες) | >> acosd(0.7071) | ans = 44.9995 |
| tan(x) | Εφαπτομένη (x σε ακτίνια) | >> tan(pi/4) | ans = 1.0000 |
| atan(x) | Αντίστροφη εφαπτομένη (x σε ακτίνια) | >> atan(0.7071) | ans = 0.6155 |
| tand(x) | Εφαπτομένη (x σε μοίρες) | >> tand(45) | ans = 1.0000 |
| atand(x) | Αντίστροφη εφαπτομένη (x σε μοίρες) | >> atand(0.7071) | ans = 35.2641 |
| exp(x) | Εκθετική συνάρτηση (e^x) | >> exp(2) | ans = 7.3891 |
| pow2(x) | Εκθετική με βάση το 2 (2^x) | >> pow2(3) | ans = 8.0000 |
| sqrt(x) | Τετραγωνική ρίζα (\sqrt{x}) | >> sqrt(9) | ans = 3.0000 |
| log10(x) | Λογάριθμος με βάση το 10 ($\log_{10} x$) | >> log10(1000) | ans = 3.000 |
| log2(x) | Λογάριθμος με βάση το 2 ($\log_2 x$) | >> log2(32) | ans = 5.000 |

| | | | |
|----------|--|---------------------------------|--------------------------|
| abs(x) | Απόλυτη τιμή ή μέτρο μιγαδικού x | >> abs(-32) >> abs(2-3i) | ans = 32 ans = 3.6056 |
| real(x) | Πραγματικό μέρος μιγαδικού | >> real(2-3i) | ans = 2.0000 |
| imag(x) | Φανταστικό μέρος μιγαδικού | >> imag(2-3i) | ans = -3.0000 |
| angle(x) | Γωνία φάσης του μιγαδικού | >> angle(2-3i) | ans = -0.9828 |
| conj(x) | Συζυγής μιγαδικού (z*) | >> conj(2-3i) | ans = 2.0000+3.000i |
| fix(x) | Αφαίρεση δεκαδικών ψηφίων | >> fix(2.6) >> fix(-2.6) | ans = 2 ans = -2 |
| floor(x) | Ο μεγαλύτερος ακέραιος που δεν υπερβαίνει το x | >> floor(2.6) >> floor(-2.6) | ans = 2 ans = -3 |
| ceil(x) | Ο μικρότερος ακέραιος που υπερβαίνει το x | >> ceil(2.6) >> ceil(-2.6) | ans = 3 ans = -2 |
| round(x) | Στρογγυλοποίηση στον πλησιέστερο ακέραιο | >> round(2.6) >> round(-2.6) | ans = 3 ans = -3 |
| rem(x,y) | Υπόλοιπο της διαίρεσης x/y | >> rem(13,5) | ans = 3 |

6 Σχηματισμοί Αριθμών

Οι πράξεις και οι μαθηματικοί υπολογισμοί στους οποίους αναφερθήκαμε ως τώρα αφορούσαν απλούς αριθμούς (ακέραιους, πραγματικούς ή μιγαδικούς). Συχνά απαιτείται να επαναλάβουμε κάποιους υπολογισμούς για ένα σύνολο αριθμών. Ωστόσο, δεδομένου ότι οι επαναλαμβανόμενες πράξεις απαιτούν χρόνο και υπολογιστικούς πόρους, το Matlab διαθέτει τη δυνατότητα να διαχειρίζεται ολόκληρους σχηματισμούς αριθμών. Ένας σχηματισμός αριθμών είναι ένα σύνολο από αριθμούς που είναι οργανωμένοι σε συγκεκριμένες θέσεις – διευθύνσεις. Τέτοιοι σχηματισμοί είναι τα **διανύσματα** και οι **πίνακες**.

6.1 Διανύσματα Γραμμής

Ας υποθέσουμε ότι θέλουμε να δημιουργήσουμε μία λίστα με τους εξής αριθμούς:

$$0, 0.1\pi, 0.2\pi, 0.3\pi, 0.4\pi, 0.5\pi, 0.6\pi, 0.7\pi, 0.8\pi, 0.9\pi, \pi$$

Για να δημιουργήσουμε αυτόν το σχηματισμό αριθμών μπορούμε να δώσουμε την εξής εντολή:

| Κώδικας Matlab |
|--|
| >> x=[0, 0.1*π, 0.2*π, 0.3*π, 0.4*π, 0.5*π, 0.6*π, 0.7*π, 0.8*π, 0.9*π, π] |
| Αποτέλεσμα |
| x = Columns 1 through 7 0 0.3142 0.6283 0.9425 1.2566 1.5708 1.8850 Columns 8 through 11 2.1991 2.5133 2.8274 3.1416 |

Παρατηρούμε ότι αρκεί να γράψουμε όλους τους αριθμούς της λίστας μέσα σε αγκύλες, χρησιμοποιώντας ως διαχωριστικό το κόμμα ή το κενό διάστημα. Ολόκληρος ο σχηματισμός αυτός θα αποθηκευτεί στη μεταβλητή x. Αν τώρα θελήσουμε για παράδειγμα να υπολογίσουμε το ημίτονο καθενός αριθμού του σχηματισμού, δηλαδή της λίστας των αριθμών, αρκεί να δώσουμε την εντολή:

| Κώδικας Matlab |
|--|
| >> y=sin(x) |
| Αποτέλεσμα |
| y = Columns 1 through 7 0 0.3090 0.5878 0.8090 0.9511 1.0000 0.9511 Columns 8 through 11 0.8090 0.5878 0.3090 0.0000 |

Έτσι θα δημιουργηθεί ένας νέος σχηματισμός αριθμών, ο y . Και οι δύο σχηματισμοί που κατασκευάσαμε, δηλαδή ο x και ο y , είναι διανύσματα γραμμής, δηλαδή τα στοιχεία τους είναι παρατεταγμένα σε μία γραμμή με 11 στήλες.

Αν θελήσουμε να αναφερθούμε σε ένα συγκεκριμένο στοιχείο του διανύσματος, αρκεί να δώσουμε τη θέση του (ή αλλιώς τη διεύθυνσή του) μέσα στο διάνυσμα. Για παράδειγμα, για να λάβουμε το 3^ο στοιχείο του x και το 5^ο του y θα δώσουμε:

| Κώδικας Matlab | Αποτέλεσμα |
|----------------|--------------|
| >> x(3) | ans = 0.6283 |
| >> y(5) | ans = 0.9511 |

Για να αναφερθούμε σε ένα ολόκληρο τμήμα του διανύσματος πρέπει να χρησιμοποιήσουμε το ειδικό σύμβολο της άνω-κάτω τελείας (:). Για παράδειγμα, αν θέλουμε να πάρουμε τα 5 πρώτα στοιχεία του διανύσματος x θα γράψουμε:

| Κώδικας Matlab | Αποτέλεσμα |
|----------------|-------------------------------------|
| >> x(1:5) | ans = 0 0.3142 0.6283 0.9425 1.2566 |

Το σύμβολο ":" μας δίνει τη δυνατότητα επιλογής στοιχείων ενός διανύσματος με πολλούς τρόπους, όπως φαίνεται στα ακόλουθα παραδείγματα:

| Κώδικας Matlab | Αποτέλεσμα |
|--|---|
| % Από το 7 ^ο ως το τελευταίο στοιχείο του x >> x(7:end) | ans = 1.8850 2.1991 2.5133 2.8274 3.1416 |
| % Από το 3 ^ο μέχρι το 1 ^ο στοιχείο του y σε ανάποδη σειρά >> y(3:-1:1) | ans = 0.5878 0.3090 0 |
| % Από το 2 ^ο μέχρι το 7 ^ο στοιχείο του x με βήμα 2 (δηλαδή ανά δύο στοιχεία). >> x(2:2:7) | ans = 0.3142 0.9425 1.5708 |
| % Το 8 ^ο , το 2 ^ο , το 9 ^ο και το 1 ^ο στοιχείο του | ans = 0.8090 0.3090 0.5878 0 |

| | |
|--|--|
| <p>y με αυτή τη σειρά</p> <pre>>> y([8 2 9 1])</pre> | |
|--|--|

Θα εξετάσουμε μερικούς εναλλακτικούς τρόπους κατασκευής διανυσμάτων, αφού ο τρόπος που αναφέραμε αρχικά (παράθεση όλων των στοιχείων του διανύσματος) είναι πρακτικά ανεφάρμοστος όταν τα στοιχεία είναι πολλά.

Για να κατασκευάσουμε το διάνυσμα x του οποίου τα στοιχεία αρχίζουν από το 0 και τελειώνουν στο 1, με βήμα αύξησης 0.1 και επιπλέον είναι όλα πολλαπλασιασμένα με το π, θα χρησιμοποιήσουμε το σύμβολο ":" ως εξής:

| Κώδικας Matlab | Αποτέλεσμα |
|------------------------------------|---|
| <pre>>> x=[0:0.1:1]*pi</pre> | <pre>x = Columns 1 through 7 0 0.3142 0.6283 0.9425 1.2566 1.5708 1.8850 Columns 8 through 11 2.1991 2.5133 2.8274 3.1416</pre> |

Μία άλλη μέθοδος κατασκευής διανύσματος δίνεται από τη συνάρτηση `linspace`, που συντάσσεται ως εξής:

`linspace(αρχική τιμή, τελική τιμή, αριθμός στοιχείων)`

Στο τελευταίο παράδειγμα χρησιμοποιώντας τη `linspace` έχουμε:

| Κώδικας Matlab | Αποτέλεσμα |
|---|---|
| <pre>>> x=linspace(0, pi, 11)</pre> | <pre>x = Columns 1 through 7 0 0.3142 0.6283 0.9425 1.2566 1.5708 1.8850 Columns 8 through 11 2.1991 2.5133 2.8274 3.1416</pre> |

Σε πιο σύνθετες περιπτώσεις μπορούμε να συνθέσουμε δύο ή περισσότερα διανύσματα. Για παράδειγμα, ας θεωρήσουμε αρχικά τα διανύσματα *a* και *b*, με τα οποία θα κατασκευάσουμε το διάνυσμα *c* από τη σύνθεση των *a* και *b*.

| Κώδικας Matlab | Αποτέλεσμα |
|---|---|
| <pre>>> a=1:5, b=1:2:9 >> c=[b a]</pre> | <pre>a = 1 2 3 4 5 b = 1 3 5 7 9 c = 1 3 5 7 9 1 2 3 4 5</pre> |

Αν θέλουμε να κατασκευάσουμε ένα διάνυσμα *d* από το 1^ο, 3^ο και 5^ο στοιχείο του *b* και τους αριθμούς 1, 0, 1 αρκεί να γράψουμε:

| Κώδικας Matlab | Αποτέλεσμα |
|--|---------------------------------|
| <pre>>> d=[b(1:2:5) 1 0 1]</pre> | <pre>d = 1 5 9 1 0 1</pre> |

6.2 Διανύσματα Στήλης

Τα διανύσματα στήλης έχουν οργανωμένα τα στοιχεία τους σε μία στήλη και πολλαπλές γραμμές. Ο πρώτος άμεσος τρόπος κατασκευής ενός διανύσματος στήλης είναι η παράθεση των στοιχείων μέσα σε αγκύλες χρησιμοποιώντας σαν διαχωριστικό σύμβολο το ελληνικό ερωτηματικό (;). Για παράδειγμα:

| Κώδικας Matlab | Αποτέλεσμα |
|-----------------------------------|--------------------------|
| <pre>>> c=[1;2;3;4;5]</pre> | <pre>c = 1 2 3 4 5</pre> |

Άλλος τρόπος είναι η κατασκευή ενός διανύσματος γραμμής και η αναστροφή του με το σύμβολο αναστροφής, που είναι ο τόνος (^). Για παράδειγμα:

| Κώδικας Matlab | Αποτέλεσμα |
|---------------------------|---------------------------|
| <pre>>> a=1:4</pre> | <pre>a = 1 2 3 4</pre> |

| | |
|---------|----------------------|
| >> b=a' | b = 1 2 3 4 |
|---------|----------------------|

Το ίδιο σύμβολο μπορεί να χρησιμοποιηθεί για την αντιστροφή ενός διανύσματος στήλης σε διάνυσμα γραμμής. Θεωρώντας το διάνυσμα b του τελευταίου παραδείγματος έχουμε:

| Κώδικας Matlab | Αποτέλεσμα |
|----------------|-------------|
| >> c=b' | b = 1 2 3 4 |

Ιδιαίτερη προσοχή απαιτείται στην περίπτωση αναστροφής μιγαδικών αριθμών. Η χρήση του συμβόλου αναστροφής (') προκαλεί τον υπολογισμό του ανάστροφου διανύσματος με τους συζυγείς των αρχικών στοιχείων. Προκειμένου να γίνει αναστροφή των μιγαδικών χωρίς χρήση των συζυγών τους, το Matlab διαθέτει το συνδυασμό των συμβόλων της τελείας και του τόνου (.'). Για τους πραγματικούς αριθμούς τα σύμβολα αυτά δεν έχουν διαφορά. Για παράδειγμα:

| Κώδικας Matlab | Αποτέλεσμα |
|-----------------------------------|---|
| >> a=1:4 >> d=a+a*i >> e=d' | a = 1 2 3 4 d = 1.0000+1.0000i 2.0000+2.0000i 3.0000+3.0000i 4.0000+4.0000i e = 1.0000 - 1.0000i 2.0000 - 2.0000i 3.0000 - 3.0000i 4.0000 - 4.0000i |
| >> ee=d.' | ee = 1.0000 + 1.0000i 2.0000 + 2.0000i 3.0000 + 3.0000i 4.0000 + 4.0000i |

6.3 Πίνακες

Επέκταση των διανυσμάτων γραμμής και στήλης αποτελούν οι σχηματισμοί που αποτελούνται από πολλαπλές γραμμές και στήλες και λέγονται πίνακες. Η δημιουργία τους ακολουθεί τους ίδιους κανόνες με αυτούς των διανυσμάτων. Οι χαρακτήρες του κενού και του κόμμα (,) χρησιμοποιούνται για να διαχωρίζουν στοιχεία της ίδιας γραμμής, ενώ το ελληνικό ερωτηματικό (;) για να διαχωρίζει γραμμές. Για παράδειγμα, για να κατασκευάσουμε έναν πίνακα 2 γραμμών και 4 στηλών μπορούμε να γράψουμε τις εξής εντολές:

| Κώδικας Matlab | Αποτέλεσμα |
|---|--|
| <code>>> g=[1 2 3 4;5 6 7 8]</code> | <code>g = 1 2 3 4</code> <code>5 6 7 8</code> |

6.4 Πράξεις Αριθμών με Σχηματισμούς

Όλες οι γνωστές πράξεις μπορούν να γίνουν μεταξύ ενός απλού αριθμού και ενός σχηματισμού αριθμών (δηλαδή διανύσματος ή πίνακα). Το αποτέλεσμα είναι ένας νέος σχηματισμός που περιέχει τα αποτελέσματα των πράξεων του καθενός στοιχείου του σχηματισμού με τον απλό αριθμό. Για παράδειγμα, αν θέλουμε να αφαιρέσουμε τον αριθμό 2 από κάθε στοιχείο του πίνακα `g` του τελευταίου παραδείγματος:

| Κώδικας Matlab | Αποτέλεσμα |
|---------------------------|---|
| <code>>> g-2</code> | <code>ans = -1 0 1 2</code> <code>3 4 5 6</code> |

Προφανώς, μπορούμε να κάνουμε και συνδυασμό πράξεων όπως:

| Κώδικας Matlab | Αποτέλεσμα |
|-----------------------------|---|
| <code>>> g*2-1</code> | <code>ans = 1 3 5 7</code> <code>9 11 13 15</code> |

6.5 Πράξεις Ανάμεσα σε Σχηματισμούς

Οι πράξεις ανάμεσα σε σχηματισμούς είναι αρκετά σύνθετες, ειδικά όταν οι διαστάσεις τους διαφέρουν. Θα εξετάσουμε την περίπτωση σχηματισμών με ίδιες διαστάσεις (δηλαδή πλήθος γραμμών και στηλών). Στην περίπτωση αυτή οι πράξεις της πρόσθεσης, αφαίρεσης, πολλαπλασιασμού και διαίρεσης γίνονται στοιχείο με στοιχείο. Ακολουθούν μερικά χαρακτηριστικά παραδείγματα:

| Κώδικας Matlab | Αποτέλεσμα |
|---|---|
| <code>>> g=[1:4;5:8;9:12]</code> | <code>g =</code> 1 2 3 4 5 6 7 8 9 10 11 12 |
| <code>>> h=[1 1 1 1;2 2 2 2;3 3 3 3]</code> | <code>h =</code> 1 1 1 1 2 2 2 2 3 3 3 3 |
| <code>>> g+h</code> | <code>ans =</code> 2 3 4 5 7 8 9 10 12 13 14 15 |
| <code>>> g-h</code> | <code>ans =</code> 0 1 2 3 3 4 5 6 6 7 8 9 |

Παρατηρούμε ότι η πρόσθεση και η αφαίρεση γίνονται κανονικά, στοιχείο με στοιχείο. Ωστόσο, για τον πολλαπλασιασμό θα πρέπει να χρησιμοποιηθεί ο συνδυασμός συμβόλων της τελείας και του αστερίσκου (.*), ώστε η πράξη να εφαρμοστεί σε επίπεδο στοιχείου με στοιχείο. Διαφορετικά εκτελείται η πράξη του πολλαπλασιασμού πινάκων της όπως ορίζεται στη γραμμική άλγεβρα. Ο αντίστοιχος συμβολισμός χρησιμοποιείται και για τη διαίρεση στοιχείου με στοιχείο, όπως επίσης και για την ύψωση σε δύναμη. Επεκτείνοντας το τελευταίο παράδειγμα έχουμε:

| Κώδικας Matlab | Αποτέλεσμα |
|----------------|---|
| >> g.*h | ans = 1 2 3 4 10 12 14 16 27 30 33 36 |
| >> g./h | ans = 1.0000 2.0000 3.0000 4.0000 2.5000 3.0000 3.5000 4.0000 3.0000 3.3333 3.6667 4.0000 |
| >> g.^h | ans = 1 2 3 4 25 36 49 64 729 1000 1331 1728 |

6.6 Σχηματισμοί με Στοιχεία 0 ή 1

Ιδιαίτερα χρήσιμη για πολλές εφαρμογές είναι η δυνατότητα να κατασκευάσουμε σχηματισμούς με όλα τα στοιχεία τους να είναι 0 ή 1. Για το σκοπό αυτόν χρησιμοποιούμε τις συναρτήσεις:

zeros(γραμμές, στήλες)

ones(γραμμές, στήλες)

Ακολουθούν μερικά παραδείγματα:

| Κώδικας Matlab | Αποτέλεσμα |
|--|--|
| % Δημιουργία πίνακα 3x3 με 1 >> ones(3) | ans = 1 1 1 1 1 1 1 1 1 |
| % Δημιουργία πίνακα 2x5 με 0 >> zeros(2,5) | ans = 0 0 0 0 0 0 0 0 0 0 |
| % Δημιουργία πίνακα με 1, που έχει τις ίδιες διαστάσεις με αυτές του g >> size(g) >> ones(size(g)) | ans = 3 4 ans = 1 1 1 1 1 1 1 1 1 1 1 1 |

Η εντολή `size(g)` βρίσκει τις διαστάσεις (γραμμές, στήλες) του πίνακα `g`.

6.7 Διαχείριση Σχηματισμών

Ο όρος «διαχείριση» αναφέρεται στις δυνατότητες που παρέχει το Matlab για την εισαγωγή, εξαγωγή και αναδιοργάνωση υποσυνόλων στοιχείων. Ακολουθούν μερικά χαρακτηριστικά παραδείγματα:

| Κώδικας Matlab | Αποτέλεσμα |
|--|--|
| % Δημιουργία του πίνακα A >> A=[1 2 3;4 5 6;7 8 9] | A = 1 2 3 4 5 6 7 8 9 |
| % Αλλαγή του στοιχείου της 3 ^{ης} γραμμής και 3 ^{ης} στήλης με το 0 >> A(3,3)=0 | A = 1 2 3 4 5 6 7 8 0 |
| % Αύξηση του πίνακα A σε 6 στήλες και ταυτόχρονη τοποθέτηση του 1 στο στοιχείο που είναι στη 2 ^η γραμμή και 6 ^η στήλη >> A(2,6)=1 | A = 1 2 3 0 0 0 4 5 6 0 0 1 7 8 9 0 0 0 |
| % Αλλαγή όλων των στοιχείων της 4 ^{ης} στήλης σε 4 >> A(:,4)=4 | A = 1 2 3 4 0 0 4 5 6 4 0 1 7 8 9 4 0 0 |
| % Κατασκευή νέου πίνακα B με τις γραμμές του αρχικού πίνακα A σε αντίστροφη σειρά >> A=[1 2 3;4 5 6;7 8 9] >> B=A(3:-1:1, 1:3) | A = 1 2 3 4 5 6 7 8 9 B = 7 8 9 4 5 6 1 2 3 |
| % Το ίδιο με το προηγούμενο μπορεί να γίνει και με την χρήση του συμβόλου ":" για να δηλώσουμε όλες τις στήλες >> B=A(3:-1:1, :) | B = 7 8 9 4 5 6 1 2 3 |
| % Για να πάρουμε την 1 ^η και την 3 ^η στήλη του B και να | C = 1 2 3 7 9 |

| | |
|--|--|
| <p>τις τοποθετήσουμε πίσω (δεξιά) από τις στήλες του A, δημιουργώντας έτσι έναν νέο πίνακα C</p> <p>>> C=[A B(:, [1 3])]</p> | <pre> 4 5 6 4 6 7 8 9 1 3 </pre> |
| <p>% Για να φτιάξουμε έναν πίνακα B από τα στοιχεία του A που βρίσκονται στις 2 πρώτες γραμμές και στις 2 τελευταίες στήλες</p> <p>>> B=A(1:2,2:3)</p> | <pre> B = 2 3 5 6 </pre> |
| <p>% Για να φτιάξουμε έναν πίνακα B από τα στοιχεία του A που βρίσκονται στην 1^η και 3^η γραμμή και στην 1^η και 3^η στήλη</p> <p>>> B=A([1 3],[1 3])</p> | <pre> B = 1 3 7 9 </pre> |
| <p>% Για να μετατρέψουμε τον πίνακα A σε διάστημα στήλης</p> <p>>> B=A(:)</p> | <pre> B = 1 4 7 2 5 8 3 6 9 </pre> |
| <p>% Για να διαγράψουμε τη 2^η στήλη του πίνακα A</p> <p>>> A(:, 2)=[]</p> | <pre> A = 1 3 4 6 7 9 </pre> |
| <p>% Για τον ανάστροφο του πίνακα A</p> <p>>> A=A.'</p> | <pre> A = 1 4 7 3 6 9 </pre> |
| <p>% Για να αντικαταστήσουμε τη 2^η γραμμή του A με το διάνυσμα (1, 4, 7)</p> <p>>> A(2,:)= [1 4 7]</p> | <pre> A = 1 4 7 1 4 7 </pre> |
| <p>% Για να φτιάξουμε νέον πίνακα B με 4 επαναλήψεις της 2^{ης} στήλης του A</p> <p>>> B=A(:, [2 2 2 2])</p> | <pre> B = 4 4 4 4 4 4 4 4 </pre> |

Οι τρόποι διαχείρισης πινάκων που αναφέραμε ως τώρα χρησιμοποιούν δύο δείκτες για τον εντοπισμό των στοιχείων. Μερικές φορές όμως είναι βολικότερη η χρήση ενός μόνο δείκτη. Στις περιπτώσεις αυτές η αρίθμηση γίνεται κατά στήλες:

| Κώδικας Matlab | Αποτέλεσμα |
|--|---|
| <pre>% Δημιουργία πίνακα D >> D=[1 2 3 4; 5 6 7 8; 9 10 11 12]</pre> | <pre>D = 1 2 3 4 5 6 7 8 9 10 11 12</pre> |
| <pre>% Αναφορά με έναν δείκτη >> D(2) >> D(10)</pre> | <pre>ans = 5 ans = 4</pre> |

Εκτός από τη χρήση δεικτών για τη διαχείριση διανυσμάτων και πινάκων, μπορούμε να χρησιμοποιήσουμε συνθήκες με σχέσεις. Η εργασία αυτή γίνεται με τη συνάρτηση `find` που αναζητάει τα στοιχεία ενός πίνακα που ικανοποιούν κάποια συγκεκριμένη συνθήκη – σχέση. Ακολουθούν μερικά παραδείγματα:

| Κώδικας Matlab | Αποτέλεσμα |
|---|--|
| <pre>>> x= -3:3 % Βρίσκω τις θέσεις του πίνακα που τα στοιχεία του είναι κατά απόλυτη τιμή μεγαλύτερα του 1 >> k=find(abs(x)>1) % Βρίσκω τις θέσεις >> y=x(k) % Τώρα βρίσκω τα στοιχεία αυτά</pre> | <pre>x = -3 -2 -1 0 1 2 3 k = 1 2 6 7 y = -3 -2 2 3</pre> |
| <pre>% Εναλλακτικά, η ίδια εργασία μπορεί να γίνει με μία εντολή ως εξής: y=x(abs(x)>1)</pre> | <pre>y = -3 -2 2 3</pre> |
| <pre>% Παράδειγμα σε πίνακα A=[1:3;4:6;7:9] find(A>5)</pre> | <pre>A = 1 2 3 4 5 6 7 8 9 ans = 3 6 8</pre> |

| | |
|--|---|
| | 9 |
|--|---|

Παρατηρούμε ότι στην περίπτωση πίνακα, εμφανίζονται οι θέσεις των στοιχείων που είναι μεγαλύτερα από 5, αλλά η μέτρηση των θέσεων γίνεται κατά στήλες. Αν θέλουμε να εμφανίσουμε τον αριθμό γραμμής και στήλης κάθε στοιχείου, τότε:

| Κώδικας Matlab | Αποτέλεσμα |
|-----------------------|--|
| >> [i, j] = find(A>5) | i = 3 3 2 3 j = 1 2 3 3 |

Για τη διαχείριση πινάκων και διανυσμάτων είναι σημαντικό να γνωρίζουμε τις διαστάσεις τους. Το Matlab παρέχει τις συναρτήσεις `size` και `length` για τον υπολογισμό των διαστάσεων των σχηματισμών.

- Η `size` επιστρέφει ένα διάνυσμα με πρώτο στοιχείο τον αριθμό των γραμμών ενός πίνακα και δεύτερο στοιχείο τον αριθμό των στηλών του πίνακα.
- Η `length` επιστρέφει τη μεγαλύτερη διάσταση του πίνακα (δηλαδή τον αριθμό των γραμμών ή στηλών, ανάλογα με το ποιος είναι μεγαλύτερος).

Ακολουθούν μερικά παραδείγματα:

| Κώδικας Matlab | Αποτέλεσμα |
|----------------------------------|--|
| >> A=[1:8;2:9] | A = 1 2 3 4 5 6 7 8 2 3 4 5 6 7 8 9 |
| >> size(A) >> [r c] = size(A) | ans = 2 8 r = 2 c = 8 |

| | |
|---|----------------------------------|
| <pre>>> r=size(A,1) >> c=size(A,2) >> length(A)</pre> | <pre>r = 2 c = 8 ans = 8</pre> |
| <pre>>> B=pi:0.01:2*pi; >> size(B) >> length(B)</pre> | <pre>ans = 1 315 ans = 315</pre> |

Συχνά χρειάζεται να υπολογιστεί το ελάχιστο ή μέγιστο στοιχείο ενός σχηματισμού ή ακόμα το άθροισμα ή το γινόμενο των στοιχείων του. Αυτό γίνεται με τις συναρτήσεις `min`, `max`, `sum`, `prod` αντίστοιχα. Ακολουθούν μερικά παραδείγματα:

| Κώδικας Matlab | Αποτέλεσμα |
|---|---|
| <pre>% Δημιουργία διανύσματος A >> A=[1 2 -3 4 5 4 3 2 1]</pre> | <pre>A = 1 2 -3 4 5 4 3 2 1</pre> |
| <pre>>> min(A) >> max(A) >> sum(A) >> prod(A)</pre> | <pre>ans = -3 ans = 5 ans = 19 ans = -2880</pre> |
| <pre>% Δημιουργία πίνακα B, που στην 1^η γραμμή έχει τα στοιχεία του A και στη 2^η τα ίδια στοιχεία του A αυξημένα κατά 3 >> B=[A; A+3]</pre> | <pre>B = 1 2 -3 4 5 4 3 2 1 4 5 0 7 8 7 6 5 4</pre> |
| <pre>>> min(B) >> max(B) >> sum(B) >> prod(B)</pre> | <pre>ans = 1 2 -3 4 5 4 3 2 1 ans = 4 5 0 7 8 7 6 5 4 ans = 5 7 -3 11 13 11 9 7 5 ans = 4 10 0 28 40 28 18 10 4</pre> |
| <pre>>> min(min(B)) >> max(max(B)) >> sum(sum(B)) >> prod(prod(B))</pre> | <pre>ans = -3 ans = 8 ans = 65 ans = 0</pre> |

Παρατηρούμε ότι στους πίνακες οι `min`, `max`, `sum`, `prod` εφαρμόζονται σε κάθε στήλη χωριστά επιστρέφοντας ένα διάνυσμα. Έτσι, για την εύρεση του ζητούμενου αποτελέσματος, απαιτείται η διπλή εφαρμογή της αντίστοιχης συνάρτησης.

7 M – Files


Για απλά προβλήματα η εργασία με μορφή εντολών και απαντήσεων στο παράθυρο εντολών του Matlab είναι γρήγορη και αποτελεσματική. Συνήθως όμως, χρησιμοποιούμε το Matlab για να επιλύσουμε κάποιο πιο σύνθετο πρόβλημα, το οποίο είτε απαιτεί πολλές εντολές είτε οι εντολές πρέπει να επαναξιολογούνται μετά από αλλαγή στις τιμές κάποιων μεταβλητών. Στις περιπτώσεις αυτές, η σύνταξη εντολών στο παράθυρο εντολών και η άμεση εκτέλεσή τους μία προς μία είναι κουραστική και χρονοβόρα. Λύση στο πρόβλημα αυτό δίνει το Matlab επιτρέποντας τη δημιουργία ειδικών αρχείων που περιέχουν εντολές. Τα αρχεία αυτά είναι εκτελέσιμα, δηλαδή μπορούμε να τα καλέσουμε μέσα από το Matlab έτσι ώστε να εκτελεστούν οι εντολές που περιέχουν, σαν να τις είχαμε εκτελέσει στο παράθυρο εντολών. Τα αρχεία αυτά λέγονται `script files` ή **M-files**.

Θα παρουσιάσουμε τη δημιουργία και την χρήση των M-files με ένα απλό παράδειγμα. Με τις γνώσεις που έχουμε ως τώρα για το Matlab, έστω ότι θέλουμε να επιλύσουμε εξισώσεις δεύτερου βαθμού (τριώνυμα) της μορφής: $ax^2+bx+c=0$. Θα ζητήσουμε δηλαδή από το Matlab να υπολογίσει τις 2 ρίζες $\frac{-b \pm \sqrt{b^2-4ac}}{2a}$ για διάφορες τιμές των συντελεστών a , b , c .

Για να δημιουργήσουμε το M-file επιλέγουμε διαδοχικά από το μενού: **File** → **New** → **M-file** (ή **script file**), οπότε ανοίγει ένα νέο παράθυρο για τη σύνταξη των εντολών:

Κώδικας Matlab

```
% Παράδειγμα M-file για την επίλυση της δευτεροβάθμιας εξίσωσης
% a*x^2 + b*x + c = 0
a=2 % Συντελεστής του x^2
b=3 % Συντελεστής του x
c=8 % Σταθερά
% Υπολογισμός των 2 ριζών
r1=(-b+sqrt(b^2-4*a*c))/(2*a)
r2=(-b-sqrt(b^2-4*a*c))/(2*a)
```

Από το μενού επιλέγουμε File  Save για να αποθηκεύσουμε το αρχείο με το όνομα example.m. Τώρα πλέον, για να εκτελεστούν όλες οι εντολές που υπάρχουν στο αρχείο αρκεί να γράψουμε στο παράθυρο εντολών το όνομα του αρχείου (χωρίς την επέκταση .m).

| Κώδικας Matlab | Αποτέλεσμα |
|----------------|---|
| >> example | a = 2 b = 3 c = 8 r1 = -0.7500+ 1.8540i r2 = -0.7500- 1.8540i |

Στην ονομασία των M-files απαιτείται προσοχή, ώστε μην δίνουμε ονόματα μεταβλητών ή συναρτήσεων. Όταν στο παράθυρο εντολών του Matlab γράψουμε το όνομα ενός m-file, τότε το Matlab ακολουθεί την εξής ιεραρχική αναζήτηση:

1. Πρώτα ελέγχει αν το όνομα που γράψαμε είναι όνομα μεταβλητής.
2. Αν δεν εντοπίσει μεταβλητή με το όνομα αυτό, τότε ελέγχει αν είναι όνομα συνάρτησης.
3. Αν δεν βρεθεί ούτε συνάρτηση τότε αναζητά m-file με το όνομα που γράψαμε.

Η εκτέλεση των εντολών του M-file γίνεται ακριβώς σαν να τις είχαμε γράψει στο παράθυρο εντολών. Δηλαδή, όλες οι εντολές έχουν πρόσβαση στις μεταβλητές του

workspace και όλες οι μεταβλητές που δημιουργούνται από την εκτέλεση του m-file καταγράφονται στο workspace.

Αν θέλουμε να εμφανίζεται στο παράθυρο εντολών μόνο η τιμή μίας μεταβλητής και όχι το όνομά της, μπορούμε να χρησιμοποιήσουμε την εντολή `disp`. Γενικά, η εντολή `disp` χρησιμοποιείται για να εμφανίζουμε μηνύματα στο παράθυρο εντολών κατά τη διάρκεια εκτέλεσης του m-file. Ας παρατηρήσουμε με ένα παράδειγμα τις διαφορές στην εμφάνιση της μεταβλητής `a`:

| Κώδικας Matlab | Αποτέλεσμα |
|--|--------------------|
| <code>>> a</code> | <code>a = 2</code> |
| <code>>> disp('Η λύση είναι: ')</code> | Η λύση είναι: |
| <code>>>disp(a)</code> | 2 |

Επίσης, η χρήση M-files παρέχει τη δυνατότητα να χρησιμοποιούμε τις ίδιες εντολές αλλάζοντας μόνο τις τιμές κάποιων αρχικών μεταβλητών. Η εντολή `input` είναι αυτή που μας επιτρέπει να πληκτρολογήσουμε την τιμή κάποιας μεταβλητής κατά τη διάρκεια εκτέλεσης του M-file. Ας τροποποιήσουμε το αρχείο `example.m` του προηγούμενου παραδείγματος ενσωματώνοντας εντολές `input` και `disp` και στη συνέχεια ας το εκτελέσουμε:

| Κώδικας Matlab |
|--|
| <pre>% Example1.m - Βελτιωμένο M-file για την επίλυση της δευτεροβάθμιας εξίσωσης % a*x^2 + b*x + c = 0 a=input('Δώστε το συντελεστή του x^2: ') b=input('Δώστε το συντελεστή του x: ') c=input('Δώστε τη σταθερά: ') disp('Οι ρίζες είναι: ') r1=(-b+sqrt(b^2-4*a*c))/(2*a); r2=(-b-sqrt(b^2-4*a*c))/(2*a); disp(r1),disp(r2)</pre> |

Αν τώρα αποθηκεύσουμε με το όνομα `example1.m` και στη συνέχεια εκτελέσουμε το αρχείο αυτό, τότε θα πάρουμε:

| Κώδικας Matlab | Αποτέλεσμα |
|----------------|---|
| >> example1 | <p>Δώστε το συντελεστή του x^2: 2</p> <p>Δώστε το συντελεστή του x: 3</p> <p>Δώστε τη σταθερά: 8</p> <p>Οι ρίζες είναι:</p> <p>r1 = -0.7500+ 1.8540i</p> <p>r2 = -0.7500- 1.8540i</p> |

Γενικά, με τις εντολές `disp`, `input` και το σύμβολο του ελληνικού ερωτηματικού (;) μπορούμε να πετύχουμε όποια εμφάνιση μηνυμάτων και αποτελεσμάτων θέλουμε.

8 Λογικές και Σχεσιακές Πράξεις

Με τις λογικές και τις σχεσιακές πράξεις μπορούμε να ελέγξουμε τη ροή εκτέλεσης των εντολών. Το αποτέλεσμα των λογικών και των σχεσιακών πράξεων – εκφράσεων είναι 0 όταν η έκφραση είναι ψευδής και 1 όταν είναι αληθής. Στους 2 επόμενους πίνακες καταγράφονται οι σχεσιακοί και οι λογικοί τελεστές.

| Σύμβολο (τελεστής) | Σχεσιακή Πράξη |
|--------------------|----------------------|
| < | Μικρότερο από |
| <= | Μικρότερο ή ίσο από |
| > | Μεγαλύτερο από |
| >= | Μεγαλύτερο ή ίσο από |
| == | Ίσο με |
| ~= | Όχι ίσο (Διάφορο) |

Προσοχή πρέπει να δοθεί στο σύμβολο της ισότητας (==), το οποίο δεν θα πρέπει να συγχέεται με το σύμβολο καταχώρησης (=). Υπενθυμίζεται ότι το σύμβολο της ισότητας (==) συγκρίνει δύο μεταβλητές και επιστρέφει 1 αν αληθεύει η ισότητα, αλλιώς επιστρέφει 0.

| Σύμβολο (τελεστής) | Λογική Πράξη |
|--------------------|--------------|
| & | Και (AND) |
| | ή (OR) |
| ~ | Όχι (NOT) |

Ακολουθούν μερικά παραδείγματα χρήσης λογικών και σχεσιακών τελεστών. Αρχικά θα δημιουργήσουμε έναν τυχαίο πίνακα 5x5 με τη βοήθεια της συνάρτησης `rand`, η οποία παράγει ομοιόμορφα κατανεμημένους τυχαίους αριθμούς στο (0,1).

| Κώδικας Matlab | Αποτέλεσμα |
|---|--|
| <pre>% Δημιουργία 5x5 πίνακα τυχαίων αριθμών >> A=rand(5) % ή A=rand(5,5)</pre> | <pre>A = 0.8687 0.4314 0.1361 0.8530 0.0760 0.0844 0.9106 0.8693 0.6221 0.2399 0.3998 0.1818 0.5797 0.3510 0.1233 0.2599 0.2638 0.5499 0.5132 0.1839 0.8001 0.1455 0.1450 0.4018 0.2400</pre> |
| <pre>% Εντοπισμός θέσεων του A με τιμή > 0.5 >> A>0.5</pre> | <pre>ans = 1 0 0 1 0 0 1 1 1 0 0 0 1 0 0 0 0 1 1 0 1 0 0 0 0</pre> |
| <pre>% Για τις θέσεις του A με τιμή <= 0.2 >> A<=0.2</pre> | <pre>ans = 0 0 1 0 1 1 0 0 0 0 0 1 0 0 1 0 0 0 0 1 0 1 1 0 0</pre> |

| | |
|---|--|
| <pre>% Δημιουργία πίνακα B από το πρώτο δεκαδικό ψηφίο καθενός στοιχείου του A >> B = fix(A*10)</pre> | <pre>B = 8 4 1 8 0 0 9 8 6 2 3 1 5 3 1 2 2 5 5 1 8 1 1 4 2</pre> |
| <pre>% Έλεγχος ποια στοιχεία του B έχουν τιμή ίση με 4 >> B==4</pre> | <pre>ans = 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0</pre> |
| <pre>% Δημιουργία πίνακα C, που έχει 1 στις αντίστοιχες θέσεις του B που έχουν τιμή στο διάστημα [2,6] >> C=(B>=2 & B<=6)</pre> | <pre>C = 0 1 0 0 0 0 0 0 1 1 1 0 1 1 0 1 1 1 1 0 0 0 0 1 1</pre> |

9 Έλεγχος Ροής των Εντολών

Το Matlab διαθέτει τρόπους ελέγχου της ροής των εντολών ενός προγράμματος με βάση αποφάσεις που λαμβάνονται από λογικούς ελέγχους. Οι τρόποι αυτοί είναι οι βρόχοι επανάληψης με τις εντολές `for` και `while` και οι δομές ελέγχου με τις εντολές `if – else` και `switch – case`.

9.1 Βρόχοι με την Εντολή For

Συντάσσονται ως εξής:

```
for n=λίστα_αριθμών
    εντολές_προς_εκτέλεση
end
```

Η for χρησιμοποιείται για την επαναλαμβανόμενη εκτέλεση μίας ομάδας εντολών για έναν καθορισμένο αριθμό φορών. Μάλιστα οι εντολές που βρίσκονται ανάμεσα στις for και end εκτελούνται μία φορά για κάθε στήλη της **λίστας_αριθμών**. Για παράδειγμα:

| Κώδικας Matlab | Αποτέλεσμα |
|--|---|
| <pre>>> for n=1:10 x(n)=sin(n*pi/10); end >> x</pre> | <pre>x = Columns 1 through 8 0.3090 0.5878 0.8090 0.9511 1.0000 0.9511 0.8090 0.5878 Columns 9 through 10 0.3090 0.0000</pre> |

Παρά το γεγονός ότι οι εκφράσεις της μορφής “n=1:10” είναι οι πιο συνηθισμένες στους βρόχους for, μπορούμε ανάλογα την περίπτωση να χρησιμοποιήσουμε οποιαδήποτε άλλη έκφραση πίνακα ή διανύσματος. Για παράδειγμα:

| Κώδικας Matlab | Αποτέλεσμα |
|---|---|
| <pre>>> data=[3 9 45 6;7 16 -1 5] >> for n=data x=n(1)-n(2) end</pre> | <pre>data = 3 9 45 6 7 16 -1 5 x = -4 x = -7 x = 46 x = 1</pre> |

Επίσης, επιτρέπονται και οι εμφωλευμένοι βρόχοι for, για παράδειγμα:

| Κώδικας Matlab | Αποτέλεσμα |
|--|--|
| <pre>>> for n=1:5 for m=5:-1:1 A(n,m)=n^2+m^2; end end</pre> | <pre>A = 2 5 10 17 26 5 8 13 20 29</pre> |

| | |
|------|----------------|
| end | 10 13 18 25 34 |
| end | 17 20 25 32 41 |
| >> A | 26 29 34 41 50 |

Για να έχουμε μεγαλύτερη ταχύτητα στην εκτέλεση των εντολών μέσα σε βρόχους, πρέπει να προσέχουμε τα εξής:

- Να αποφεύγουμε να χρησιμοποιούμε τη for όταν μπορούμε να λύσουμε το πρόβλημα με πιο άμεσο τρόπο. Για παράδειγμα:

| Κώδικας Matlab | Αποτέλεσμα |
|---------------------------------|---|
| >> n=1:10; >> x=sin(n*pi/10) | x = Columns 1 through 6 0.3090 0.5878 0.8090 0.9511 1.0000 0.9511 Columns 7 through 10 0.8090 0.5878 0.3090 0.0000 |

- Αν πρόκειται να δημιουργήσουμε έναν πίνακα με εντολές που βρίσκονται μέσα σε ένα βρόχο for, είναι καλύτερα να ορίσουμε τον πίνακα πριν το βρόχο for, δίνοντας σε όλα τα στοιχεία του πίνακα κάποια αρχική τιμή (συνήθως 0 ή 1). Με τον τρόπο αυτόν αποφεύγουμε την αύξηση των διαστάσεων του πίνακα σε κάθε βήμα επανάληψης του βρόχου, αφού η αύξηση αυτή είναι χρονοβόρα και απαιτεί αρκετό χώρο μνήμης. Για παράδειγμα, θα ήταν καλύτερα να είχαμε γράψει:

| Κώδικας Matlab | Αποτέλεσμα |
|---|---|
| >> A=zeros(5,5); >> for n=1:5 for m=5:-1:1 A(n,m)=n^2+m^2; end end | A = 2 5 10 17 26 5 8 13 20 29 10 13 18 25 34 17 20 25 32 41 26 29 34 41 50 |

| | |
|------|--|
| >> A | |
|------|--|

9.2 Βρόχοι με While

Συντάσσονται ως εξής:

```
while συνθήκη
    εντολές_προς_εκτέλεση
end
```

Η while χρησιμοποιείται για την επαναλαμβανόμενη εκτέλεση μίας ομάδας εντολών όσο ισχύει μία συνθήκη που έχουμε ορίσει. Εδώ ο αριθμός των επαναλήψεων δεν είναι καθορισμένος όπως στη for. Μάλιστα οι εντολές που βρίσκονται ανάμεσα στις while και end εκτελούνται για όσο διάστημα η συνθήκη είναι αληθής.

Για παράδειγμα, έστω ότι θέλουμε να προσεγγίσουμε την τιμή της γεωμετρικής σειράς $\sum_{n=0}^{\infty} \left(\frac{1}{3}\right)^n$, χρησιμοποιώντας ακρίβεια 6 δεκαδικών ψηφίων. Αυτό που θα κάνουμε είναι να υπολογίσουμε την τιμή της $\left(\frac{1}{3}\right)^n$ αρχικά για n=0 και μετά θα αυξάνουμε κάθε φορά το n κατά 1, όσο η τιμή της $\left(\frac{1}{3}\right)^n$ είναι μεγαλύτερη από 0.000001 (6 δεκαδικά ψηφία). Επίσης, οι εντολές θα μπορούσαν να αποθηκευθούν σε ένα m-file. Στον κώδικα που ακολουθεί φαίνονται οι εντολές του σχετικού m-file.

| Κώδικας Matlab | Αποτέλεσμα |
|--|--------------------|
| % Αρχικές τιμές | 1 |
| sum=0; n=0; accur=0.000001; | 1.3333333333333333 |
| % Για να εμφανίζονται ως 14 δεκαδικά ψηφία | 1.4444444444444444 |
| format long; | 1.481481481481481 |
| while (1/3)^n > accur | 1.493827160493827 |
| sum=sum+(1/3)^n; | 1.497942386831276 |
| | 1.499314128943759 |

| | |
|----------------------|-------------------|
| n=n+1; | 1.499771376314586 |
| disp(sum) | 1.499923792104862 |
| end | 1.499974597368287 |
| | 1.499991532456096 |
| disp('Επαναλήψεις=') | 1.499997177485366 |
| disp(n) | 1.499999059161789 |
| disp('Αποτέλεσμα=') | Επαναλήψεις= |
| disp(sum) | 13 |
| | Αποτέλεσμα= |
| | 1.499999059161789 |

Παρατηρώντας τα αποτελέσματα της εκτέλεσης, βλέπουμε ότι η διαδικασία τερματίστηκε μετά από 12 βήματα. Πράγματι, στο 13^ο βήμα (τελευταίο βήμα) έχουμε $(\frac{1}{3})^{13} = 6.272254743863065e-07 < 0.000001$.

9.3 Δομές Ελέγχου με if – else

Συχνά είναι απαραίτητο κάποιες εντολές να εκτελούνται εφόσον ικανοποιείται κάποια λογική συνθήκη. Για το σκοπό αυτόν το Matlab διαθέτει την εντολή if – else. Στην απλούστερη μορφή της συντάσσεται ως εξής:

if συνθήκη

Εντολές1

end

Οι εντολές μεταξύ των if και end εκτελούνται μόνο αν η συνθήκη είναι αληθής.

Αν υπάρχουν δύο εναλλακτικές περιπτώσεις, τότε η εντολή if συντάσσεται ως εξής:

if συνθήκη

Εντολές1

else

Εντολές2

end

Στην περίπτωση αυτή η 1^η ομάδα εντολών (Εντολές1) εκτελείται εφόσον η συνθήκη είναι αληθής, διαφορετικά εκτελείται η 2^η ομάδα εντολών (Εντολές2).

Αν υπάρχουν πολλές εναλλακτικές περιπτώσεις, τότε η if συντάσσεται ως εξής:

if συνθήκη1

Εντολές1

elseif συνθήκη2

Εντολές2

elseif συνθήκη3

Εντολές3

.....

else

Εντολές

end

Στην περίπτωση αυτή εκτελείται η ομάδα εντολών που αντιστοιχεί στην πρώτη αληθή συνθήκη που θα συναντήσει το Matlab, ενώ οι υπόλοιπες συνθήκες αγνοούνται. Η else δεν είναι απαραίτητο να υπάρχει, ωστόσο, ο ρόλος της είναι να εκτελούνται κάποιες εντολές όταν καμμία από τις συνθήκες δεν είναι αληθής.

Ως παράδειγμα, έστω ότι θέλουμε να υπολογίσουμε τους 20 πρώτους όρους της ακολουθίας a_n , η οποία ορίζεται ως εξής:

$$a_n = \begin{cases} 1, & n = 3k \\ \frac{2k-1}{k+1}, & n = 3k+1 \\ \frac{k}{2k+1}, & n = 3k+2 \end{cases}$$

Για το σκοπό του παραδείγματος θα χρησιμοποιήσουμε τη συνάρτηση $\text{rem}(x,y)$ που υπολογίζει το υπόλοιπο της ακέραιης διαίρεσης x/y , ενώ επίσης, οι εντολές θα μπορούσαν να αποθηκευθούν σε ένα m-file. Στον κώδικα που ακολουθεί φαίνονται οι εντολές του σχετικού m-file.

| Κώδικας Matlab | Αποτέλεσμα |
|---|--|
| <pre>A=zeros(1,20); for n=1:20 if rem(n,3)==0 a(n)=1; elseif rem(n,3)==1 a(n)=(2*n-1)/(n+1); else a(n)=n/(2*n+1); end end disp(a)</pre> | <pre>Columns 1 through 6 0.5000 0.4000 1.0000 1.4000 0.4545 1.0000 Columns 7 through 12 1.6250 0.4706 1.0000 1.7273 0.4783 1.0000 Columns 13 through 18 1.7857 0.4828 1.0000 1.8235 0.4857 1.0000 Columns 19 through 20 1.8500 0.4878</pre> |

9.4 Δομές Ελέγχου με switch – case

Χρησιμοποιούνται όταν σύνολα εντολών πρέπει να εκτελεστούν αν ικανοποιείται κάποια λογική συνθήκη η οποία όμως εξαρτάται από έναν επαναλαμβανόμενο έλεγχο ισότητας. Στις περιπτώσεις αυτές, για απλότητα, προτιμούμε την εντολή switch – case αντί της if – else. Η switch – case συντάσσεται ως εξής:

switch **έκφραση**

case **έλεγχος1**

Εντολές1

case **έλεγχος2**

Εντολές2

case **έλεγχος3**

Εντολές3

.....

otherwise

Εντολές

end

Εδώ γίνεται πρώτα έλεγχος αν η έκφραση ισούται με τον έλεγχο1 οπότε εκτελούνται οι εντολές1 και οι υπόλοιπες αγνοούνται. Διαφορετικά ο έλεγχος

προχωρά με τον ίδιο τρόπο στον έλεγχο2. Η διαδικασία συνεχίζεται μέχρι να επαληθευθεί κάποια ισότητα. Αν αποτύχουν όλοι οι έλεγχοι, τότε εκτελούνται οι εντολές μετά την otherwise.

Θα τροποποιήσουμε το προηγούμενο παράδειγμα, το οποίο χρησιμοποιήσαμε και στη δομή ελέγχου if –else, για να δείξουμε με ποιον τρόπο μπορούμε να αντικαταστήσουμε τις εντολές if – else με τις εντολές switch – case, απλοποιώντας έτσι τη μορφή των εκφράσεων.

| Κώδικας Matlab | Αποτέλεσμα |
|--|--|
| <pre>A=zeros(1,20); for n=1:20 switch rem(n,3) case 0 a(n)=1; case 1 a(n)=(2*n-1)/(n+1); otherwise a(n)=n/(2*n+1); end end disp(a)</pre> | <pre>Columns 1 through 6 0.5000 0.4000 1.0000 1.4000 0.4545 1.0000 Columns 7 through 12 1.6250 0.4706 1.0000 1.7273 0.4783 1.0000 Columns 13 through 18 1.7857 0.4828 1.0000 1.8235 0.4857 1.0000 Columns 19 through 20 1.8500 0.4878</pre> |

10 Γραφικές Παραστάσεις

Για τη δημιουργία γραφικών παραστάσεων στο Matlab οι πιο συχνά χρησιμοποιούμενες συναρτήσεις φαίνονται στον παρακάτω πίνακα:

| Συνάρτηση | Επεξήγηση |
|-----------|---|
| figure | Δημιουργεί νέο παράθυρο για την απεικόνιση γραφικών παραστάσεων |

| | |
|---------|--|
| plot | Δημιουργεί μία γραφική παράσταση |
| subplot | Τμηματοποίηση ενός παραθύρου για γραφικές παραστάσεις σε γραμμές και στήλες, προκειμένου στο ίδιο παράθυρο να απεικονιστούν πολλαπλές γραφικές παραστάσεις |
| stem | Γραφική παράσταση διακριτών σημάτων – ακολουθιών |
| title | Τοποθετεί τίτλο σε μια γραφική παράσταση |
| axis | Ορίζει τα όρια των αξόνων και άλλες ιδιότητες των αξόνων σε μια γραφική παράσταση |
| grid | Προσθέτει – αφαιρεί πλέγμα στις γραφικές παραστάσεις, βελτιώνοντας την ανάγνωσή τους |

Ακολουθούν μερικά παραδείγματα που αξιοποιούν όσα έχουν αναφερθεί μέχρι τώρα.

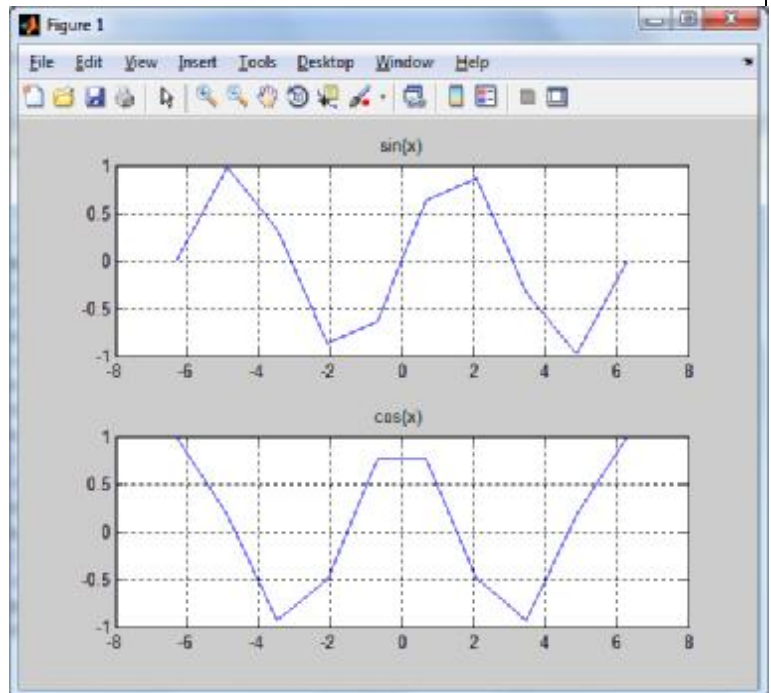
Παράδειγμα

- A. Να γίνουν οι γραφικές παραστάσεις των συναρτήσεων $\sin(x)$ και $\cos(x)$ στο διάστημα $x=[-2\pi, 2\pi]$ χρησιμοποιώντας 10 δείγματα.
- B. Ομοίως, αλλά χρησιμοποιώντας 1000 δείγματα.
- C. Να γίνουν οι γραφικές παραστάσεις των ακόλουθων συνεχών συναρτήσεων στο διάστημα $x=[-10, 10]$, χρησιμοποιώντας 1000 δείγματα.
- $f_1(x)=2x + 1$ και
 - $f_2(x)=x^2 + 2x + 1$
- D. Ομοίως, αλλά οι συναρτήσεις να θεωρηθούν ότι είναι διακριτές στο διάστημα $n=[-10, 10]$.

Κώδικας Matlab

```
%% Υλοποίηση ερωτήματος Α  
  
% Δημιουργία διανύσματος 10 θέσεων  
x=linspace(-2*pi,2*pi,10);  
  
% Υπολογισμός τιμών των  
συναρτήσεων  
y1=sin(x);  
y2=cos(x);  
  
% Κατασκευή γραφικής παράστασης  
figure(1);  
subplot(2,1,1);  
plot(x,y1);  
title('sin(x)');  
grid on  
subplot(2,1,2);  
plot(x,y2);  
title('cos(x)');  
grid on
```

Αποτέλεσμα




```

%% Υλοποίηση ερωτήματος B

%Δημιουργία διανύσματος 1000
θέσεων

x=linspace(-2*pi,2*pi,1000);

% Υπολογισμός τιμών των
συναρτήσεων

y1=sin(x);

y2=cos(x);

% Κατασκευή γραφικής παράστασης

figure(2);

subplot(2,1,1);

plot(x,y1);

title('sin(x)');

grid on

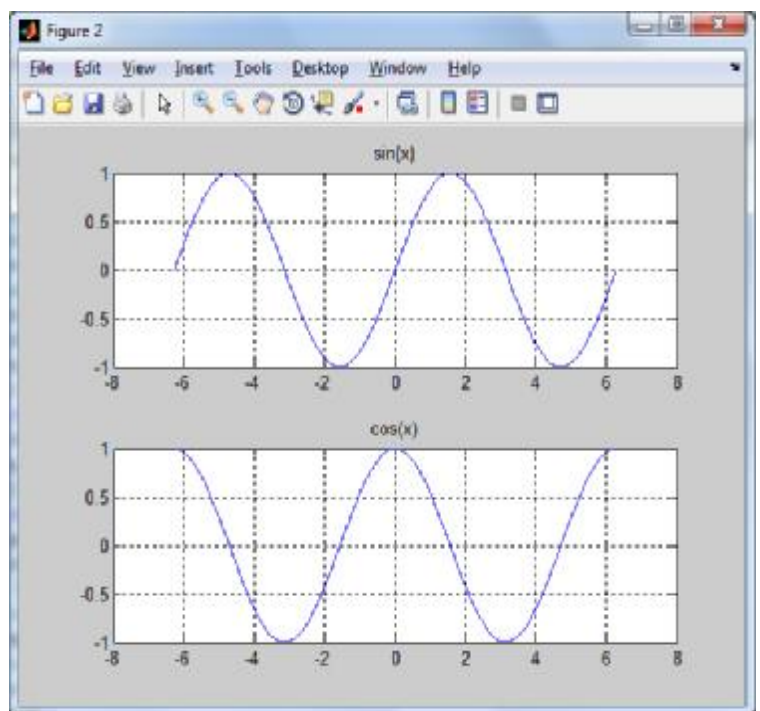
subplot(2,1,2);

plot(x,y2);

title('cos(x)');

grid on

```



Παρατήρηση:

Συγκρίνοντας τις 2 γραφικές παραστάσεις των ερωτημάτων A και B, φαίνεται ότι η γραφική παράσταση του ερωτήματος B είναι πολύ πιο λεπτομερής. Αυτό οφείλεται στο ότι χρησιμοποιήσαμε πολύ μεγαλύτερο πλήθος τιμών (1000) σε σχέση με το ερώτημα A.

```

%% Υλοποίηση ερωτήματος C

%Δημιουργία διανύσματος 1000
θέσεων

x=linspace(-10,10,1000);

% Υπολογισμός τιμών των
συναρτήσεων

f1=2*x+1;

f2=x.^2+2*x+1;

% Κατασκευή γραφικής παράστασης

figure(1);

subplot(2,1,1);

plot(x,f1);

title('f_1(x)=2x+1');

grid on

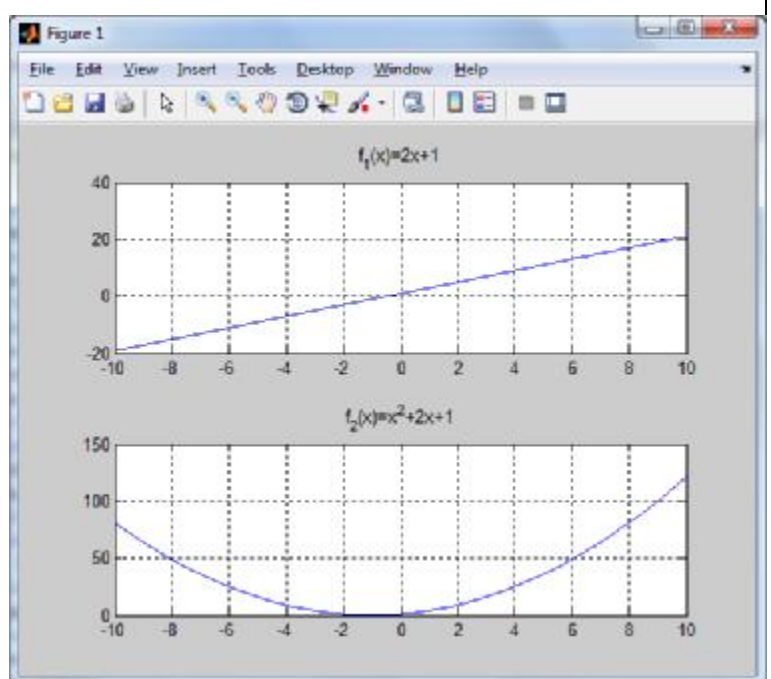
subplot(2,1,2);

plot(x,f2);

title('f_2(x)=x^2+2x+1');

grid on

```



```

%% Υλοποίηση ερωτήματος D

%Δημιουργία διανύσματος διακριτών
θέσεων στο [-10,10]

n=-10:10;

% Υπολογισμός τιμών των
συναρτήσεων

% Η διακριτή μεταβλητή συνήθως
χαρκτηρίζεται με το γράμμα n

f1=2*n+1;

f2=n.^2+2*n+1;

% Κατασκευή γραφικής παράστασης

figure(2);

subplot(2,1,1);

stem(n,f1);

title('f_1(n)=2n+1');

grid on

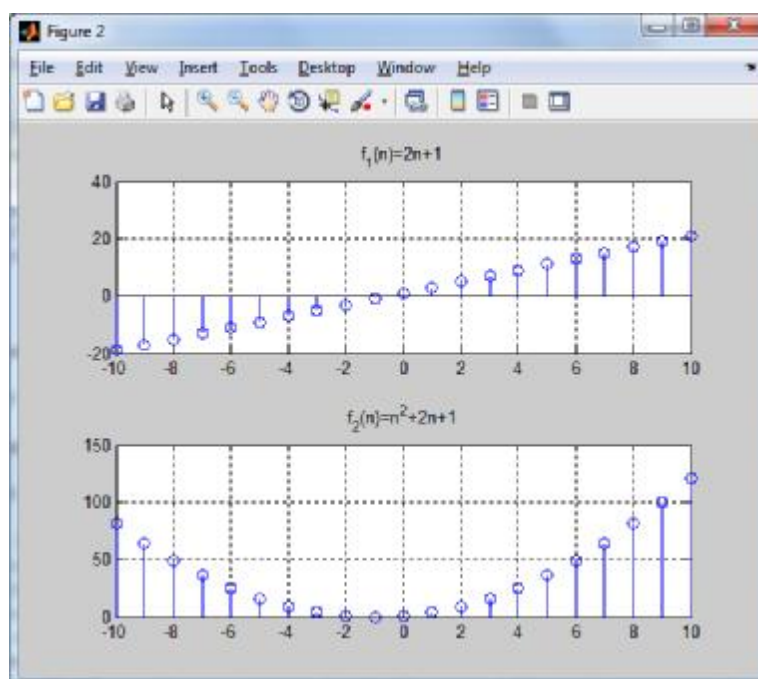
subplot(2,1,2);

stem(n,f2);

title('f_2(n)=n^2+2n+1');

grid on

```



Παρατήρηση:

Συγκρίνοντας τις 2 γραφικές παραστάσεις των ερωτημάτων C και D, γίνεται σαφής η διαφορά μεταξύ των συναρτήσεων συνεχούς και διακριτής μεταβλητής.

Επίσης, τονίζεται το γεγονός ότι για τις διακριτές συναρτήσεις χρησιμοποιούμε ακέραιες τιμές για την ανεξάρτητη μεταβλητή, καθώς και τη συνάρτηση `stem` για την γραφική απεικόνιση.

Αντίθετα, στις συναρτήσεις που η ανεξάρτητη μεταβλητή τους είναι συνεχής, χρησιμοποιούμε πραγματικές τιμές, καθώς και τη συνάρτηση `plot` για την γραφική απεικόνιση.

11 Άλυτες Ασκήσεις

Εκτελέστε όλα τα παραδείγματα και τις λυμένες ασκήσεις της παρούσας εργαστηριακής άσκησης στο Matlab

12 Επίλογος – Σύνδεση με τα Επόμενα

Στην παρούσα εργαστηριακή άσκηση καλύφθηκαν οι βασικοί χρονικοί μετασχηματισμοί των ΣΔΧ, όπως η μετατόπιση, η κλιμάκωση, η αντιστροφή, καθώς και ο συνδυασμός πολλαπλών μετασχηματισμών. Επίσης, μελετήθηκαν χαρακτηριστικά μεγέθη των ΣΔΧ, όπως η ενέργεια και η ισχύς.

Στην επόμενη εργαστηριακή άσκηση θα παρουσιαστούν η δημιουργία Σημάτων Διακριτού Χρόνου (ΣΔΧ), καθώς και η εκτέλεση των βασικών πράξεων μεταξύ τους.

13 Βιβλιογραφία

Hayes , Μ. Η. (2002). *Ψηφιακή Επεξεργασία Σήματος*. (Α. Δημητριάδης, Μεταφρ.) Θεσσαλονίκη: Τζιόλα.

Hsu, Η. Ρ. (2002). *Αναλογικές και Ψηφιακές Επικοινωνίες*. Θεσσαλονίκη: Τζιόλα.

Jayant, Ν. Σ., & Noll, Ρ. (1984). *Digital Coding of Waveforms: Principles and Applications to Speech and Video*. Prentice Hall.

Mano, Μ. Μ., & Ciletti, Μ. D. (2012). *Digital Design (5 εκδ.)*. Pearson Education.

MathWorks - DSP. (2017). Ανάκτηση 2 10, 2017, από Getting Started with Matlab DSP: <https://www.mathworks.com/help/dsp/getting-started-with-dsp-system-toolbox.html>

Proakis, J. G., & Salehi, Μ. (2008). *Digital Communications (5 εκδ.)*. McGraw-Hill Higher Education.

Μουρτζόπουλος, Ι. (2012). *Ψηφιακή Τεχνολογία Ήχου*. Πάτρα: Πανεπιστήμιο Πατρών.

Εργαστηριακή Άσκηση 2: Δημιουργία Βασικών Σημάτων Διακριτού Χρόνου – Βασικές Πράξεις

1 Εισαγωγικά Στοιχεία

1.1 Σκοπός

Σκοπός της παρούσας εργαστηριακής άσκησης είναι η εξοικείωση των φοιτητών – σπουδαστών με τη δημιουργία των βασικών σημάτων διακριτού χρόνου (ΣΔΧ) στο MATLAB και την εκτέλεση βασικών πράξεων μεταξύ τους.

1.2 Αναμενόμενα Μαθησιακά Αποτελέσματα

Με την ολοκλήρωση της παρούσας εργαστηριακής άσκησης, ο φοιτητής – σπουδαστής θα είναι σε θέση να:

- Κατασκευάζει τα θεμελιώδη ΣΔΧ, όπως:
 - Μοναδιαία κρουστική ώση
 - Μοναδιαία βηματική ακολουθία
 - Γραμμική ακολουθία
 - Πραγματικό εκθετικό σήμα
 - Μιγαδικό εκθετικό σήμα
 - Ημιτονοειδές σήμα
 - Συνημιτονοειδές σήμα
- Εκτελεί πράξεις μεταξύ ΣΔΧ, όπως:
 - πρόσθεση σημάτων
 - αφαίρεση σημάτων
 - πολλαπλασιασμός σημάτων

1.3 Εκτιμώμενη Διάρκεια

Ο εκτιμώμενος χρόνος που απαιτείται από τον φοιτητή – σπουδαστή για την ολοκλήρωση της παρούσας εργαστηριακής άσκησης είναι 2 διδακτικές ώρες.

1.4 Προαπαιτούμενα

Διάλεξη 1 της θεωρίας.

1.5 Εισαγωγή

Στην άσκηση αυτή θα ασχοληθούμε με τα σήματα διακριτού χρόνου. Θα γνωρίσουμε τα πιο βασικά σήματα διακριτού χρόνου, καθώς και τις στοιχειώδεις πράξεις που εφαρμόζονται σε τέτοιου είδους σήματα. Όλα αυτά θα αποτελέσουν τα απαραίτητα εργαλεία για την μελέτη των συστημάτων και την ανάλυση των σημάτων που θα μας απασχολήσουν σε όλη την έκταση αυτού του μαθήματος.

1.6 Κατάλογος Χρήσιμων Συναρτήσεων Matlab

Κατά την εκπόνηση των ασκήσεων χρησιμοποιούνται οι ακόλουθες συναρτήσεις – εντολές του Matlab:

| Συνάρτηση | Επεξήγηση |
|-----------|---|
| size | Επιστρέφει το μέγεθος πίνακα |
| length | Επιστρέφει το μήκος ενός πίνακα-διανύσματος |
| zeros | Δημιουργία πίνακα – διανύσματος με μηδενικά |
| sin | Συνάρτηση ημιτόνου |
| cos | Συνάρτηση συνημιτόνου |
| exp | Εκθετική συνάρτηση |
| real | Επιστρέφει το πραγματικό μέρος μιγαδικού αριθμού |
| imag | Επιστρέφει το φανταστικό μέρος μιγαδικού αριθμού |
| figure | Δημιουργεί νέο παράθυρο για την απεικόνιση γραφικών παραστάσεων |
| plot | Δημιουργεί μία γραφική παράσταση |
| subplot | Τμηματοποίηση ενός παραθύρου για γραφικές παραστάσεις σε γραμμές και στήλες, προκειμένου στο ίδιο παράθυρο να |

| | |
|-------|--|
| | απεικονιστούν πολλαπλές γραφικές παραστάσεις |
| stem | Γραφική παράσταση διακριτών σημάτων – ακολουθιών |
| title | Τοποθετεί τίτλο σε μια γραφική παράσταση |
| axis | Ορίζει τα όρια των αξόνων και άλλες ιδιότητες των αξόνων σε μια γραφική παράσταση |
| grid | Προσθέτει – αφαιρεί πλέγμα στις γραφικές παραστάσεις, βελτιώνοντας την ανάγνωσή τους |

2 Βασικά Σήματα Διακριτού Χρόνου

Στη μελέτη Διακριτών Σημάτων υπάρχει ένας αριθμός από βασικά σήματα που εμφανίζονται συχνά. Τα σήματα που περιγράφονται στη συνέχεια θεωρούνται ως τα πιο βασικά (στοιχειώδη) σήματα διακριτού χρόνου.

2.1 Μοναδιαία Κρουστική Ακολουθία (Μοναδιαία Κρουστική Ώση)

Η μοναδιαία κρουστική ακολουθία είναι το βασικότερο και ταυτόχρονα το απλούστερο σήμα, το οποίο ορίζεται ως εξής:

$$d(n) = \begin{cases} 1, & n = 0 \\ 0, & n \neq 0 \end{cases}$$

Άσκηση 1^η

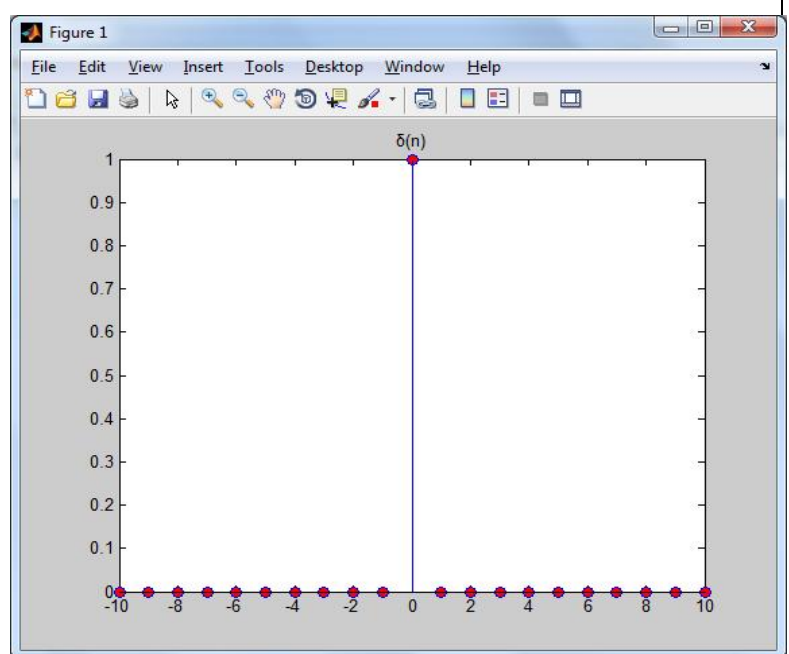
Να σχεδιαστεί το σήμα $x(n)=\delta(n)$ για το διάστημα $n=[-10, 10]$

| Κώδικας Matlab | Αποτέλεσμα |
|----------------|------------|
|----------------|------------|

```

n=[-10:10];
d=zeros(length(n));
d(n==0)=1;
figure;
stem(n,d,'MarkerFaceColor','red'
);
title('δ(n)');

```



2.2 Μοναδιαία Βηματική Ακολουθία

Η μοναδιαία βηματική ορίζεται ως εξής:

$$u(n) = \begin{cases} 1, & n \geq 0 \\ 0, & n < 0 \end{cases}$$

Άσκηση 2^η

Να σχεδιαστεί το σήμα $x(n)=u(n)$ για το διάστημα $n=[-5, 10]$.

Κώδικας Matlab

Αποτέλεσμα


```

n=[-5:10];

u=zeros(length(n));

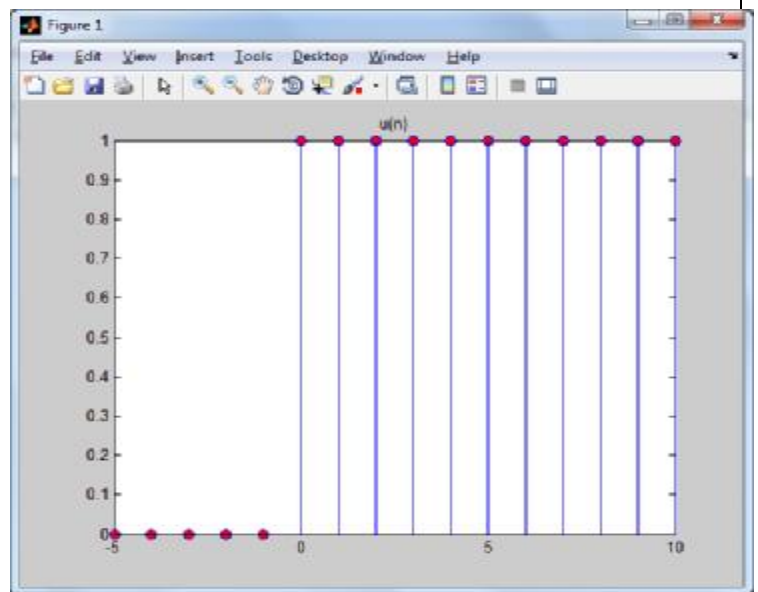
u(n>=0)=1;

figure;

stem(n,u,'MarkerFaceColor','red')
;

title('u(n)');

```



2.3 Γραμμική Ακολουθία

Η γραμμική ακολουθία ορίζεται ως εξής:

$$x(n) = A^n, \text{ όπου } A = \text{πραγματικός αριθμός}$$

Άσκηση 3^η

Να σχεδιαστεί το σήμα $x(n) = A^n$, για $A=1$ και $A=2$, στο διάστημα $n=[-10,10]$.

| Κώδικας Matlab | Αποτέλεσμα |
|----------------|------------|
|----------------|------------|

```

n=[-10:10];

A1=1;

x1=A1*n;

A2=2;

x2=A2*n;

figure;

subplot(2,1,1)

stem(n,x1,'MarkerFaceColor','red'
);

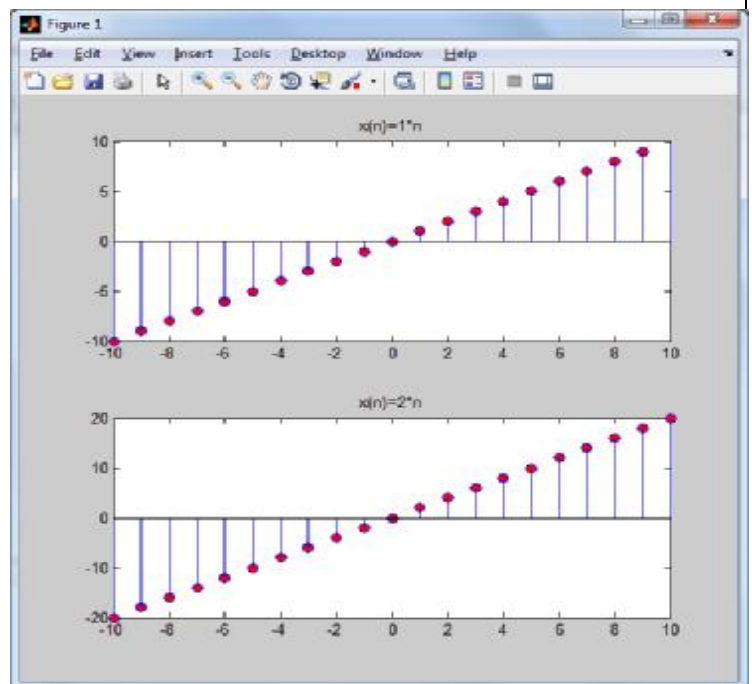
title('x(n)=1*n');

subplot(2,1,2)

stem(n,x2,'MarkerFaceColor','red'
);

title('x(n)=2*n');

```



2.4 Εκθετικά Σήματα

Τα εκθετικά σήματα ορίζονται ως εξής:

$$x(n) = \alpha^n$$

Η τιμή της βάσης (α) καθορίζει τη μορφή του σήματος. Πιο συγκεκριμένα:

- Αν α = πραγματικός αριθμός, τότε:
 - Αν $|\alpha| < 1$, τότε $x(n)$ = φθίνουσα ακολουθία
 - Αν $|\alpha| > 1$, τότε $x(n)$ = αύξουσα ακολουθία
- Αν α = μιγαδικός αριθμός, δηλαδή $\alpha = r \cdot e^{j\omega}$, τότε $x(n) = \alpha^n = (r \cdot e^{j\omega})^n = r^n \cdot e^{j\omega n}$, που αναλύεται σε πραγματικό και φανταστικό μέρος με χρήση του ημιτόνου και συνημιτόνου ως εξής:

$$x(n) = r^n \cdot e^{j\omega n} = r^n \cdot [\cos(\omega n) + j \cdot \sin(\omega n)]$$

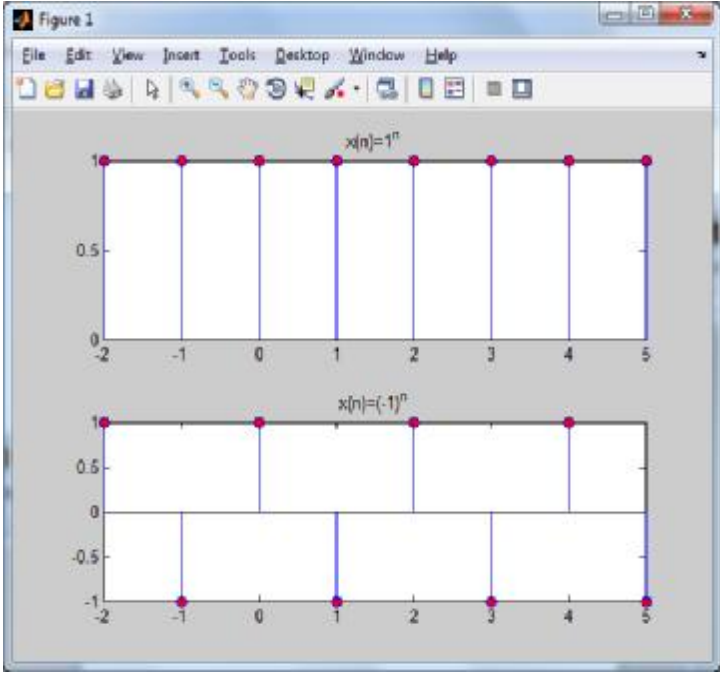
Μπορούμε να διακρίνουμε τις εξής περιπτώσεις για το r :

- Αν $r = 1$, τότε το πραγματικό και το φανταστικό μέρος της $x(n)$ είναι μία συνημιτονοειδής και μία ημιτονοειδής ακολουθία.
- Αν $r < 1$, τότε έχουμε φθίνουσες ημιτονοειδείς ακολουθίες
- Αν $r > 1$, τότε έχουμε αύξουσες ημιτονοειδείς ακολουθίες

Άσκηση 4η

Να σχεδιαστεί το σήμα $x(n) = \alpha^n$, για $n=[-2,5]$ και για τις εξής τιμές του α :

- $\alpha = 1$ και $\alpha = -1$
- $\alpha = 0.5$, $\alpha = -0.5$
- $\alpha = 2$, $\alpha = -2$

| Κώδικας Matlab | Αποτέλεσμα |
|---|--|
| <pre>n=[-2:5]; a=1; x11=a.^n; a=-1; x12=a.^n; figure(1); subplot(2,1,1) stem(n,x11,'MarkerFaceColor','red'); title('x(n)=1^n'); subplot(2,1,2) stem(n,x12,'MarkerFaceColor','red');</pre> |  |

```
title('x(n)=(-1)^n');
```

```
a=0.5;
```

```
x21=a.^n;
```

```
a=-0.5;
```

```
x22=a.^n;
```

```
figure(2);
```

```
subplot(2,1,1)
```

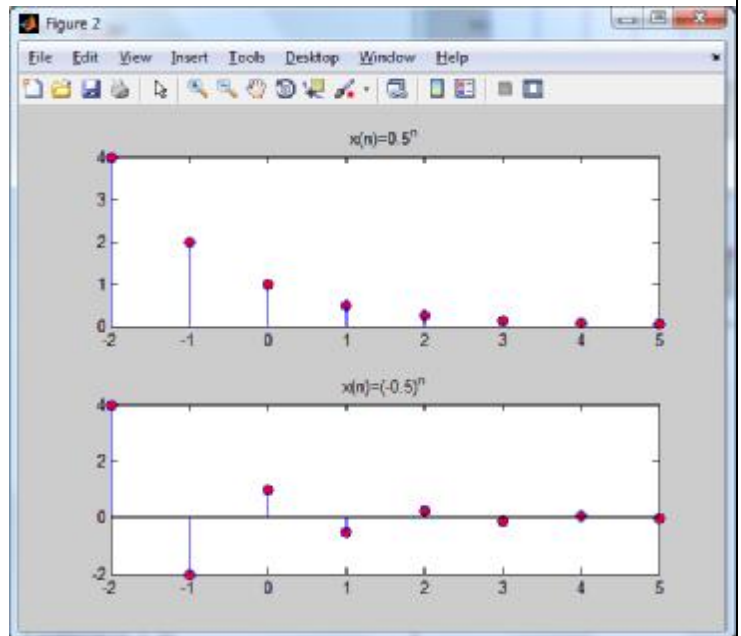
```
stem(n,x21,'MarkerFaceColor','red');
```

```
title('x(n)=0.5^n');
```

```
subplot(2,1,2)
```

```
stem(n,x22,'MarkerFaceColor','red');
```

```
title('x(n)=(-0.5)^n');
```



```
a=2;
```

```
x31=a.^n;
```

```
a=-2;
```

```
x32=a.^n;
```

```
figure(3);
```

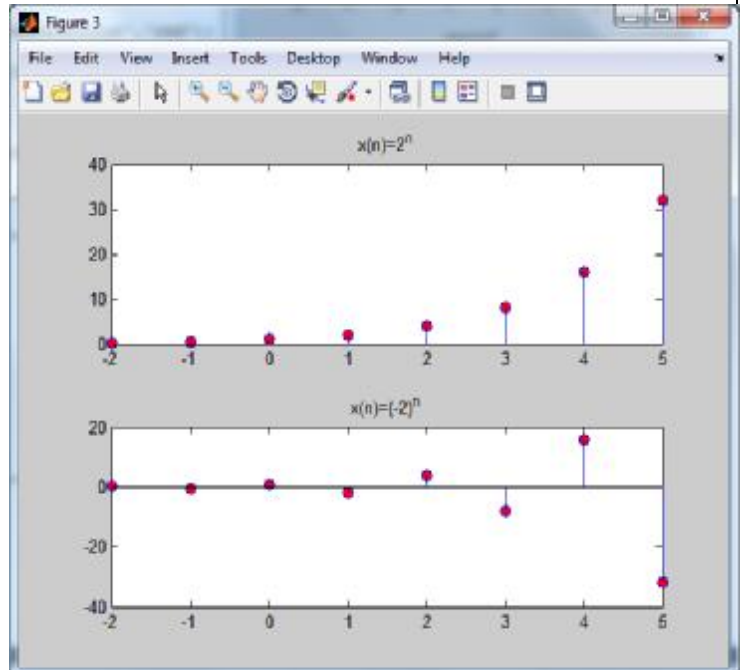
```
subplot(2,1,1)
```

```
stem(n,x31,'MarkerFaceColor','red');
```

```
title('x(n)=2^n');
```

```
subplot(2,1,2)
```

```
stem(n,x32,'MarkerFaceColor','red');
```



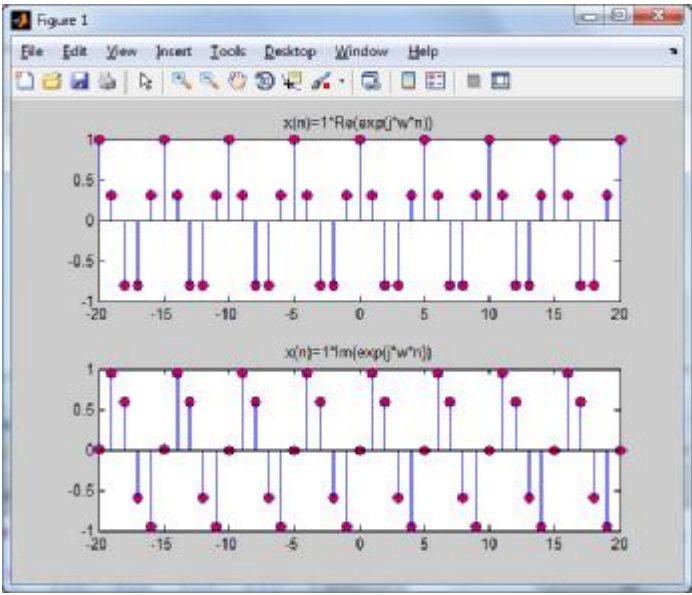
```
);  
  
title('x(n)=(-2)^n');
```

Άσκηση 5η

Να σχεδιαστεί το πραγματικό και το φανταστικό μέρος του σήματος: $x(n) = r^* e^{j\omega n}$, στο διάστημα $n=[-20,20]$ με περίοδο $N=5$, για τις εξής τιμές του r :

- $r = 1$
- $r = 3$
- $r = -3$

Υπόδειξη: $\omega=2\pi/N$

| Κώδικας Matlab | Αποτέλεσμα |
|---|--|
| <pre>n=[-20:20]; N=5; r=1; w=2*pi/N; x1=r*exp(j*w*n); figure(1); subplot(2,1,1); stem(n,real(x1),'MarkerFaceColor','red'); title('x(n)=1*Re(exp(j*w*n))'); subplot(2,1,2); stem(n,imag(x1),'MarkerFaceColor','red'); title('x(n)=1*Im(exp(j*w*n))');</pre> |  |

```

r=3;

x2=r*exp(j*w*n);

figure(2);

subplot(2,1,1);

stem(n,real(x2),'MarkerFaceColor','red'
);

title('x(n)=3*Re(exp(j*w*n))');

subplot(2,1,2);

stem(n,imag(x2),'MarkerFaceColor','red'
);

title('x(n)=3*Im(exp(j*w*n))');

r=-3;

x3=r*exp(j*w*n);

figure(3);

subplot(2,1,1);

stem(n,real(x3),'MarkerFaceColor','red'
);

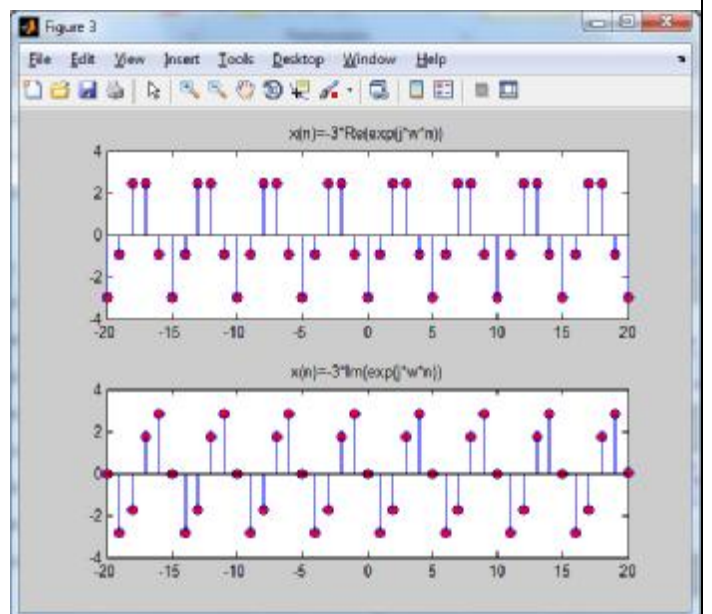
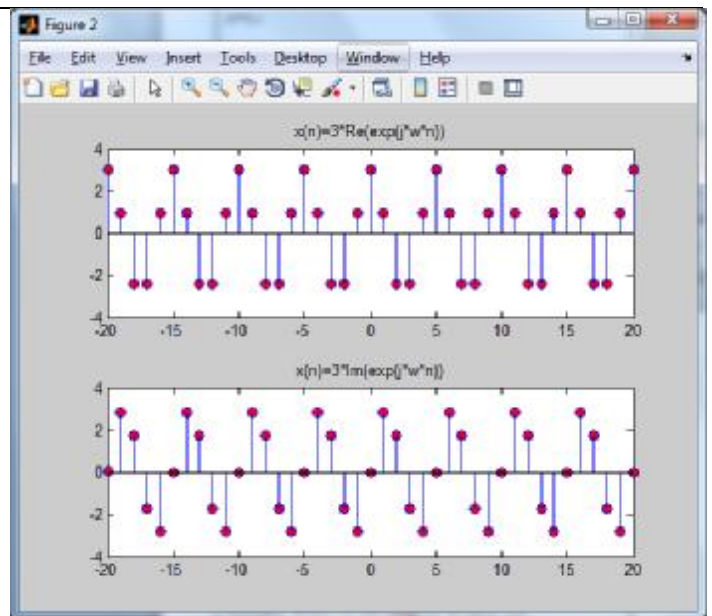
title('x(n)=-3*Re(exp(j*w*n))');

subplot(2,1,2);

stem(n,imag(x3),'MarkerFaceColor','red'
);

title('x(n)=-3*Im(exp(j*w*n))');

```



2.5 Ημιτονοειδή και Συνημιτονοειδή Σήματα

Ένα ημιτονοειδές ΣΔΧ εκφράζεται ως εξής: $x(n) = A \cdot \sin(\omega n + \theta)$, ενώ ένα συνημιτονοειδές ΣΔΧ εκφράζεται ως εξής: $x(n) = A \cdot \cos(\omega n + \theta)$, όπου:

- ∅ A = πλάτος του σήματος
- ∅ $\omega = 2\pi f$, η συχνότητα του σήματος σε ακτίνια ανά δείγμα και
- ∅ f = η συχνότητα σε κύκλους ανά δείγμα (μετράται σε Hz)
- ∅ n = ακέραιος, που αντιπροσωπεύει τον αριθμό (θέση) του δείγματος και
- ∅ θ = η φάση σε ακτίνια

Συνεπώς, χρησιμοποιώντας τη σχέση $\omega = 2\pi f$, μπορούμε να ξαναγράψουμε τις προηγούμενες σχέσεις του ημιτονοειδούς και του συνημιτονοειδούς σήματος στην εξής μορφή:

- Ημιτονοειδές ΣΔΧ: $x(n) = A \cdot \sin(2\pi f n + \theta)$
- Συνημιτονοειδές ΣΔΧ: $x(n) = A \cdot \cos(2\pi f n + \theta)$

Άσκηση 6^η

Να σχεδιαστούν τα σήματα: $x(n) = A \cdot \sin(2\pi f n + \theta)$ και $x(n) = A \cdot \cos(2\pi f n + \theta)$ για $A=2$, $f=1/16$, $\theta = \pi/3$, στο διάστημα $n=[0, 100]$.

| Κώδικας Matlab | Αποτέλεσμα |
|---|------------|
| <pre>n=[0:100]; A=2; f=1/16; th=pi/3; x1=A*sin(2*pi*f*n+th); x2=A*cos(2*pi*f*n+th);</pre> | |

```

figure;

subplot(2,1,1);

stem(n,x1,'MarkerFaceColor','red'
);

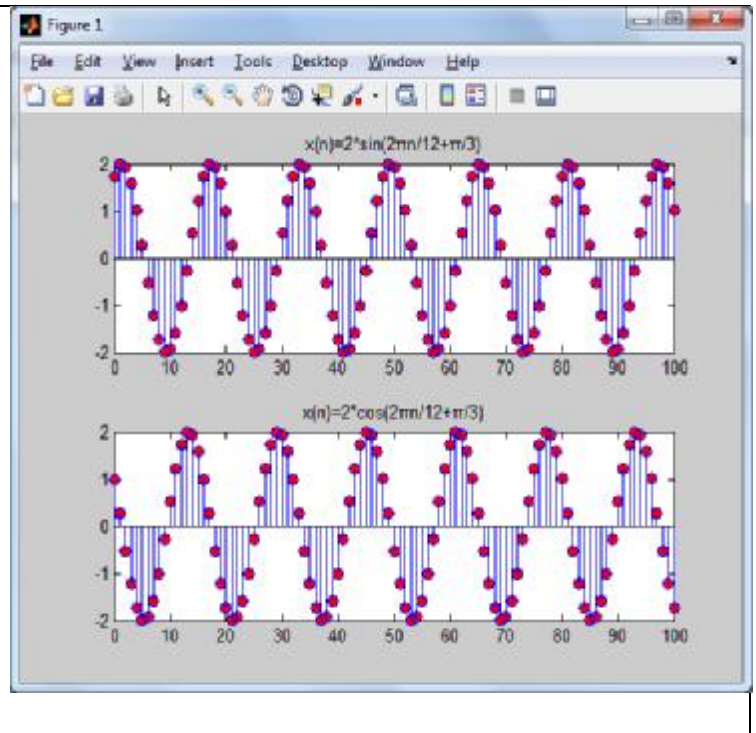
title('x(n)=2*sin(2πn/12+π/3)');

subplot(2,1,2);

stem(n,x2,'MarkerFaceColor','red'
);

title('x(n)=2*cos(2πn/12+π/3)');

```



Άσκηση 7^η

Να σχεδιαστεί το σήμα $x(n) = \cos(2\pi n/4) + \sin(2\pi n/6)$ για το διάστημα $n=[0,100]$.

Κώδικας Matlab

```

n=[0:100];

f1=1/4;

f2=1/6;

x=cos(2*pi*f1*n)+sin(2*pi*f2*n);

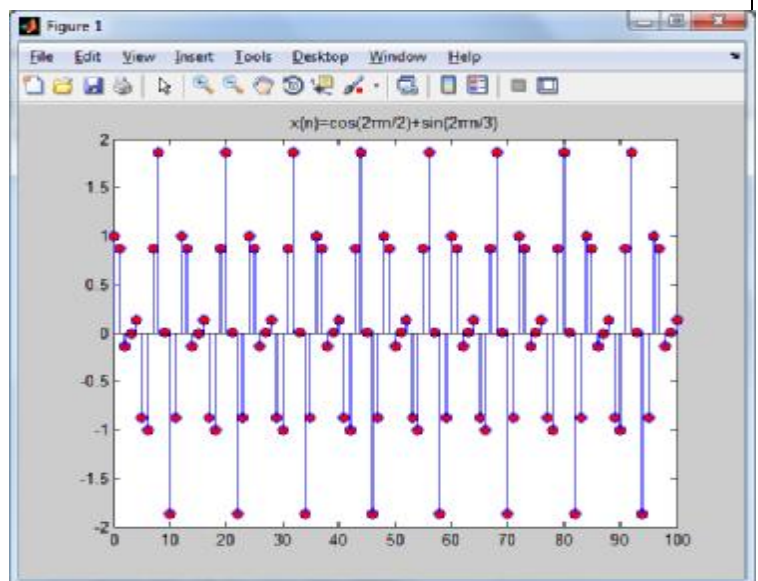
figure;

stem(n,x,'MarkerFaceColor','red');

title('x(n)=cos(2πn/4)+sin(2πn/6)')
;

```

Αποτέλεσμα



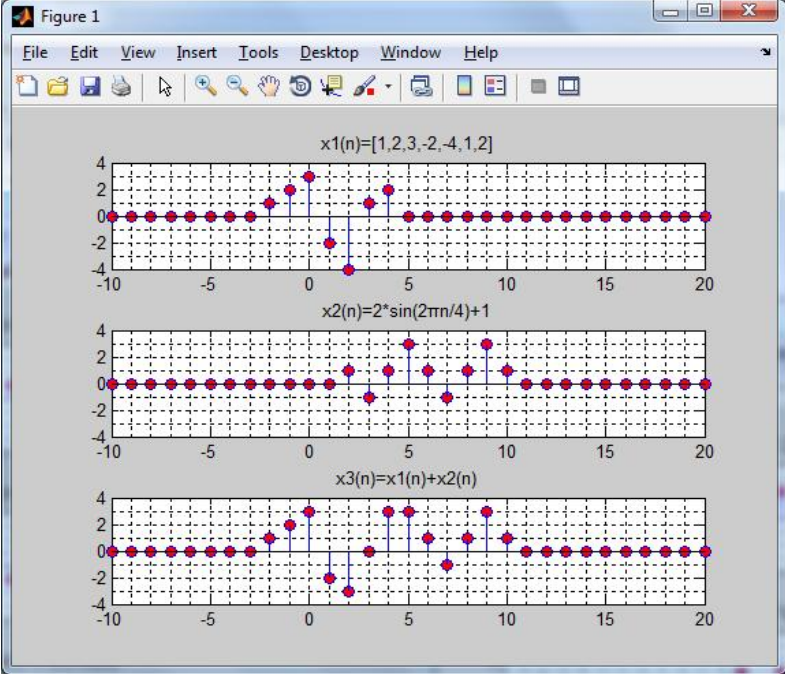
3 Βασικές Πράξεις Μεταξύ ΣΔΧ

Οι βασικές πράξεις (πρόσθεση, αφαίρεση, πολλαπλασιασμός και διαίρεση) μεταξύ των ΣΔΧ πραγματοποιούνται δείγμα – δείγμα σε καθεμία θέση.

Άσκηση 8^η

Δίνονται τα σήματα $x_1(n)=\{1,2,3,-2,-4,1,2\}$ με $n=[-2,4]$, και $x_2(n)=2*\sin(2\pi n/4)+1$ με $n=[2,10]$. Να σχεδιαστούν στο διάστημα $n=[-10,20]$ τα σήματα:

- $x_1(n)+x_2(n)$
- $x_1(n)-x_2(n)$ και
- $x_1(n)*x_2(n)$

| Κώδικας Matlab | Αποτέλεσμα |
|--|--|
| <pre>n=[-10:20]; x1=[1 2 3 -2 -4 1 2]; x1=[zeros(1,8) x1 zeros(1,16)]; x2=2*sin(2*pi*[2:10]/4)+1; x2=[zeros(1,12) x2 zeros(1,10)]; x3=x1+x2; x4=x1-x2; x5=x1.*x2; figure(1);</pre> |  |

```

subplot(3,1,1)

stem(n,x1,'MarkerFaceColor','red'
);

axis([-10,20,-4,4]);

title('x1(n)=[1,2,3,-2,-4,1,2]');

grid minor;

subplot(3,1,2)

stem(n,x2,'MarkerFaceColor','red'
);

axis([-10,20,-4,4]);

title('x2(n)=2*sin(2*pi*n/4)+1');

grid minor;

subplot(3,1,3)

stem(n,x3,'MarkerFaceColor','red'
);

axis([-10,20,-4,4]);

title('x3(n)=x1(n)+x2(n)');

grid minor;

figure(2);

subplot(3,1,1)

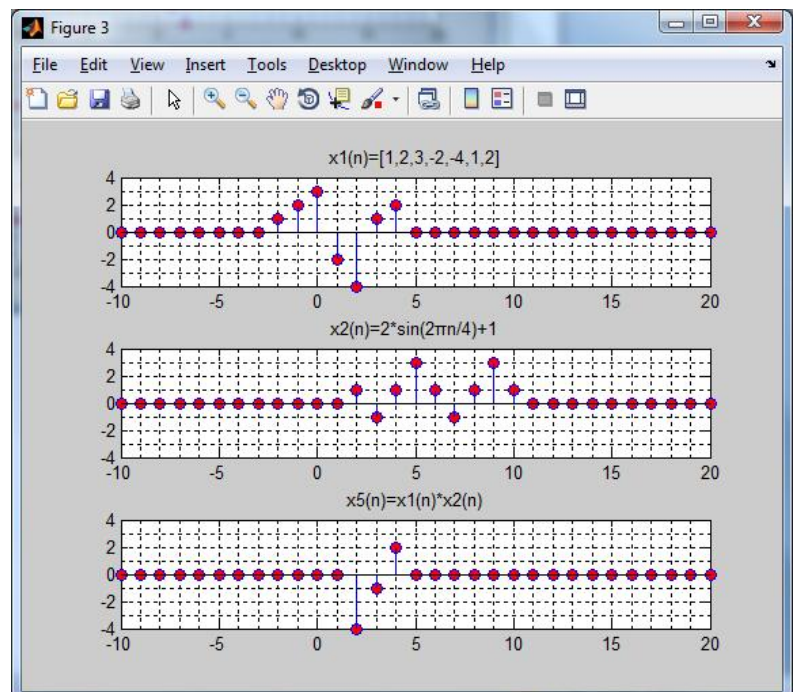
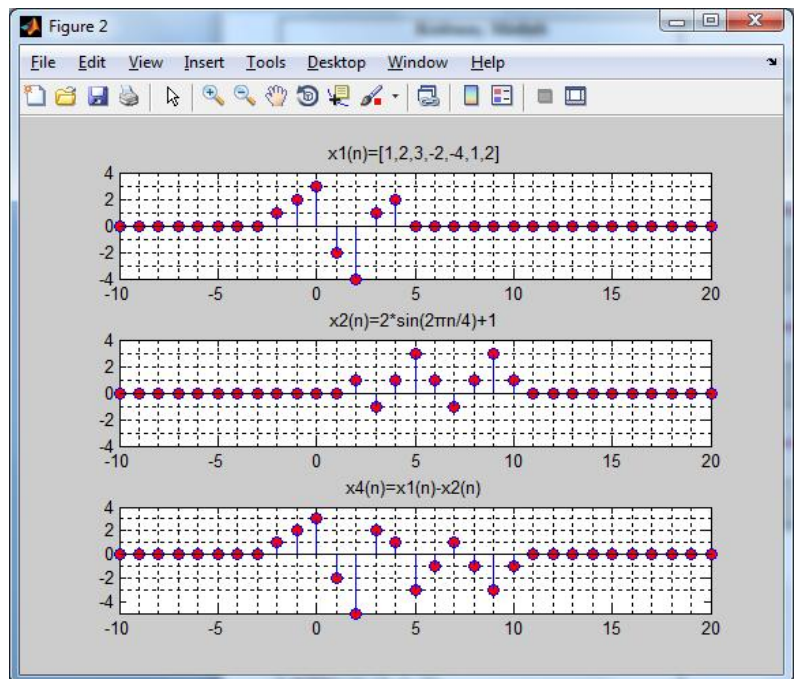
stem(n,x1,'MarkerFaceColor','red'
);

axis([-10,20,-4,4]);

title('x1(n)=[1,2,3,-2,-4,1,2]');

grid minor;

```



```

subplot(3,1,2)

stem(n,x2,'MarkerFaceColor','red'
);

axis([-10,20,-4,4]);

title('x2(n)=2*sin(2πn/4)+1');

grid minor;

subplot(3,1,3)

stem(n,x4,'MarkerFaceColor','red'
);

axis([-10,20,-5,4]);

title('x4(n)=x1(n)-x2(n)');

grid minor;

figure(3);

subplot(3,1,1)

stem(n,x1,'MarkerFaceColor','red'
);

axis([-10,20,-4,4]);

title('x1(n)=[1,2,3,-2,-4,1,2]');

grid minor;

subplot(3,1,2)

stem(n,x2,'MarkerFaceColor','red'
);

axis([-10,20,-4,4]);

title('x2(n)=2*sin(2πn/4)+1');

grid minor;

```

```
subplot(3,1,3)

stem(n,x5,'MarkerFaceColor','red'
);

axis([-10,20,-4,4]);

title('x5(n)=x1(n)*x2(n)');

grid minor;
```

4 Άλυτες Ασκήσεις

1. Να σχεδιάσετε με το Matlab τα παρακάτω σήματα:

- $x_1(n) = 2d(n) + 3d(n-1) + 5d(n-2)$
- $x_2(n) = 3u(n+5)$ για $-10 \leq n \leq 8$
- $x_3(n) = u(n) - u(n-5)$ για $-2 \leq n \leq 12$
- $x_4(n) = u(n)u(n-2)$ για $-3 \leq n \leq 10$

2. Να σχεδιάσετε με το Matlab τα παρακάτω σήματα:

$$x_1(n) = 10 \cos\left(\frac{\rho n}{10}\right) [u(n) - u(n-13)] \quad \text{για } n = [0:13].$$

$$x_2(n) = 7 \cos\left(\frac{7\rho n}{3}\right) [u(n) - u(n-30)] \quad \text{για } n = [0:30].$$

5 Επίλογος – Σύνδεση με τα Επόμενα

Στην παρούσα εργαστηριακή άσκηση καλύφθηκε η κατασκευή των βασικότερων σημάτων διακριτού χρόνου που συναντώνται στην ψηφιακή επεξεργασία σήματος, καθώς και η εκτέλεση των βασικών πράξεων μεταξύ τους.

Στην επόμενη εργαστηριακή άσκηση θα παρουσιαστούν οι μετασχηματισμοί της ανεξάρτητης μεταβλητής, όπως η χρονική μετατόπιση, αντιστροφή και η κλιμάκωση στο χρόνο, καθώς και χαρακτηριστικά μεγέθη των σημάτων, όπως η ενέργεια και η ισχύς.

6 Βιβλιογραφία

Hayes , Μ. Η. (2002). *Ψηφιακή Επεξεργασία Σήματος*. (Α. Δημητριάδης, Μεταφρ.) Θεσσαλονίκη: Τζιόλα.

Hsu, Η. Ρ. (2002). *Αναλογικές και Ψηφιακές Επικοινωνίες*. Θεσσαλονίκη: Τζιόλα.

Jayant, Ν. S., & Noll, Ρ. (1984). *Digital Coding of Waveforms: Principles and Applications to Speech and Video*. Prentice Hall.

Mano, Μ. Μ., & Ciletti, Μ. D. (2012). *Digital Design (5 εκδ.)*. Pearson Education.

MathWorks - DSP. (2017). Ανάκτηση 2 10, 2017, από Getting Started with Matlab DSP: <https://www.mathworks.com/help/dsp/getting-started-with-dsp-system-toolbox.html>

Proakis, J. G., & Salehi, Μ. (2008). *Digital Communications (5 εκδ.)*. McGraw-Hill Higher Education.

Μουρτζόπουλος, Ι. (2012). *Ψηφιακή Τεχνολογία Ήχου*. Πάτρα: Πανεπιστήμιο Πατρών.

Εργαστηριακή Άσκηση 3: Μετασχηματισμοί της Ανεξάρτητης Μεταβλητής – Χαρακτηριστικά Μεγέθη ΣΔΧ

1 Εισαγωγικά Στοιχεία

1.1 Σκοπός

Σκοπός της παρούσας εργαστηριακής άσκησης είναι η εξοικείωση των φοιτητών – σπουδαστών με τη εκτέλεση των βασικών μετασχηματισμών της μεταβλητής «n» των ΣΔΧ στο MATLAB, καθώς και με τον υπολογισμό χαρακτηριστικών μεγεθών τους, όπως η ενέργεια και η ισχύς.

1.2 Αναμενόμενα Μαθησιακά Αποτελέσματα

Με την ολοκλήρωση της παρούσας εργαστηριακής άσκησης, ο φοιτητής – σπουδαστής θα είναι σε θέση να:

- Εκτελεί τους βασικούς μετασχηματισμούς της ανεξάρτητης μεταβλητής, όπως:
 - χρονική μετατόπιση
 - αντιστροφή
 - κλιμάκωση στο χρόνο (πολλαπλασιασμός και διαίρεση συχνότητας)
 - πολλαπλοί – διαδοχικοί μετασχηματισμοί
- Υπολογίζει τα χαρακτηριστικά μεγέθη ενός ΣΔΧ, όπως:
 - ενέργεια
 - στιγμιαία και μέση ισχύς

1.3 Εκτιμώμενη Διάρκεια

Ο εκτιμώμενος χρόνος που απαιτείται από τον φοιτητή – σπουδαστή για την ολοκλήρωση της παρούσας εργαστηριακής άσκησης είναι 2 διδακτικές ώρες.

1.4 Προαπαιτούμενα

Διάλεξη 1 και 2 της θεωρίας.

1.5 Εισαγωγή

Στην άσκηση αυτή θα ασχοληθούμε ξανά με τα σήματα διακριτού χρόνου. Θα γνωρίσουμε τους βασικούς μετασχηματισμούς της ανεξάρτητης μεταβλητής στα σήματα διακριτού χρόνου, όπως οι μετατοπίσεις και η κλιμάκωση, καθώς και τα μεγέθη της ενέργειας και της ισχύος. Αυτά θα αποτελέσουν απαραίτητα εργαλεία για την μελέτη των συστημάτων και την ανάλυση των σημάτων, που θα μας απασχολήσουν σε όλη την έκταση αυτού του μαθήματος.

1.6 Κατάλογος Χρήσιμων Συναρτήσεων Matlab

Κατά την εκπόνηση των ασκήσεων χρησιμοποιούνται οι ακόλουθες συναρτήσεις – εντολές του Matlab:

| Συνάρτηση | Επεξήγηση |
|-----------|---|
| length | Επιστρέφει το μήκος ενός πίνακα-διανύσματος |
| zeros | Δημιουργία πίνακα – διανύσματος με μηδενικά |
| find | Βρίσκει τον δείκτη θέσης ενός διανύσματος / πίνακα, σύμφωνα με κάποια συνθήκη. Πχ, η $\text{find}(n \neq 0)$ βρίσκει τον δείκτη στο διάνυσμα n , όπου το n έχει την τιμή 0. |
| fliplr | Αναποδογυρίζει μίας ακολουθίας, ώστε το πρώτο στοιχείο να γίνει τελευταίο και το τελευταίο πρώτο |
| sin | Συνάρτηση ημιτόνου |
| sum | Άθροισμα όρων |
| abs | Απόλυτη τιμή |
| fprintf | Εμφάνιση μορφοποιημένου κειμένου στην οθόνη |
| figure | Δημιουργεί νέο παράθυρο για την απεικόνιση γραφικών παραστάσεων |
| plot | Δημιουργεί μία γραφική παράσταση |

| | |
|---------|--|
| subplot | Τμηματοποίηση ενός παραθύρου για γραφικές παραστάσεις σε γραμμές και στήλες, προκειμένου στο ίδιο παράθυρο να απεικονιστούν πολλαπλές γραφικές παραστάσεις |
| stem | Γραφική παράσταση διακριτών σημάτων – ακολουθιών |
| title | Τοποθετεί τίτλο σε μια γραφική παράσταση |
| axis | Ορίζει τα όρια των αξόνων και άλλες ιδιότητες των αξόνων σε μια γραφική παράσταση |
| grid | Προσθέτει – αφαιρεί πλέγμα στις γραφικές παραστάσεις, βελτιώνοντας την ανάγνωσή τους |

2 Μετασχηματισμοί της Ανεξάρτητης Μεταβλητής

Στη μελέτη Διακριτών Σημάτων υπάρχει ένας αριθμός από μετασχηματισμούς στον χρόνο, οι οποίοι εμφανίζονται πολύ συχνά. Οι μετασχηματισμοί αυτοί είναι:

1. Μετατόπιση – ολίσθηση
2. Αντιστροφή
3. Κλιμάκωση στον χρόνο

2.1 Μετατόπιση - Ολίσθηση

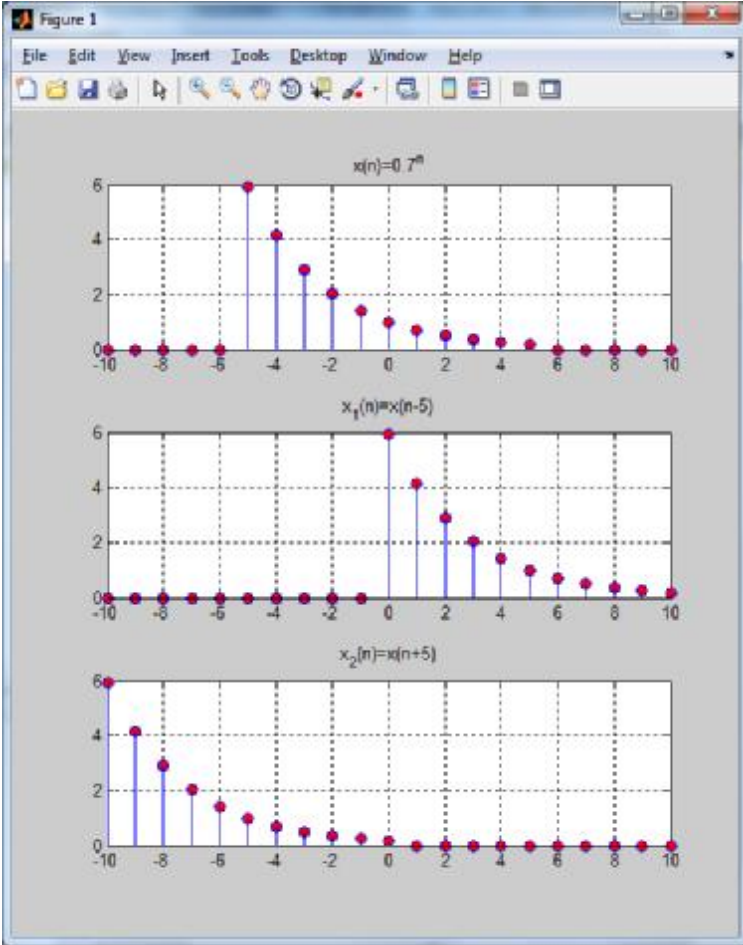
Η χρονική μετατόπιση (ολίσθηση) της ακολουθίας – σήματος $x(n)$ ορίζεται από τη σχέση $x(n - n_0)$ και προκαλεί μετατόπιση του σήματος $x(n)$ προς τα δεξιά ή προς τα αριστερά κατά n_0 θέσεις, ανάλογα με το πρόσημο του n_0 . Πιο συγκεκριμένα:

- Αν $n_0 < 0$, τότε έχουμε μετατόπιση του αρχικού σήματος $x(n)$ προς τα **δεξιά** (καθυστέρηση) κατά n_0 θέσεις. Πχ, για $n_0=3$, το σήμα $x(n-3)$ προκύπτει από το σήμα $x(n)$, μετατοπίζοντας τις τιμές – πλάτη κατά 3 θέσεις δεξιά.
- Αν $n_0 > 0$, τότε έχουμε μετατόπιση του αρχικού σήματος $x(n)$ προς τα **αριστερά** (προπορεία) κατά n_0 θέσεις. Πχ, για $n_0=3$, το σήμα $x(n+3)$ προκύπτει από το σήμα $x(n)$, μετατοπίζοντας τις τιμές – πλάτη κατά 3 θέσεις αριστερά.

Άσκηση 1^η

Έστω το σήμα $x(n)=\alpha^n$, όπου $\alpha=0.7$ και ορίζεται στο διάστημα $n=[-5,5]$. Να σχεδιαστούν στο διάστημα $n=[-10,10]$ τα εξής σήματα:

- $x_1(n)=x(n-5)$ και
- $x_2(n)=x(n+5)$

| Κώδικας Matlab | Αποτέλεσμα |
|---|---|
| <pre>n=[-10:10]; k=find(n==0); x=zeros(1,length(n)); a=0.7; n1=[-5:5]; x(k-5:k+5)=a.^n1; x1=zeros(1,length(n)); %x(n-5) x1(k:end)=x(k-5:k+5); x2=zeros(1,length(n)); %x(n+5) x2(1:k)=x(k-5:k+5); figure(1); subplot(3,1,1); stem(n,x,'MarkerFaceColor','red'); title('x(n)=0.7^n'); grid on subplot(3,1,2); stem(n,x1,'MarkerFaceColor','red');</pre> |  |

| | |
|---|--|
| <pre> title('x_1(n)=x(n-5)'); grid on subplot(3,1,3); stem(n,x2,'MarkerFaceColor','red'); title('x_2(n)=x(n+5)'); grid on % Εναλλακτικός τρόπος για x(n-5) %x1=[zeros(1,10) x(1+5:end-5)]; % Εναλλακτικός τρόπος για x(n+5) %x2=[x(1+5:end-5) zeros(1,10)]; </pre> | |
|---|--|

2.2 Αντιστροφή

Η αντιστροφή στον χρόνο του σήματος $x(n)$ ορίζεται από τη σχέση $x(-n)$ και προκαλεί την αναδίπλωση του σήματος γύρω από τη θέση 0.

Άσκηση 2^η

Έστω το σήμα $x(n)=\alpha^n$, όπου $\alpha=0.7$ και ορίζεται στο διάστημα $n=[-5,5]$. Να σχεδιαστεί στο διάστημα $n=[-10,10]$ το σήμα: $x(-n)$.

| | |
|-----------------------|-------------------|
| Κώδικας Matlab | Αποτέλεσμα |
|-----------------------|-------------------|

```

n=[-10:10];

k=find(n==0);

x=zeros(1,length(n));

a=0.7;

n1=[-5:5];

x(k-5:k+5)=a.^n1;

x1=fliplr(x); %x(-n)

n1=-fliplr(n1);

figure(1);

subplot(2,1,1);

stem(n,x,'MarkerFaceColor','red');

title('x(n)=0.7^n');

grid on

subplot(2,1,2);

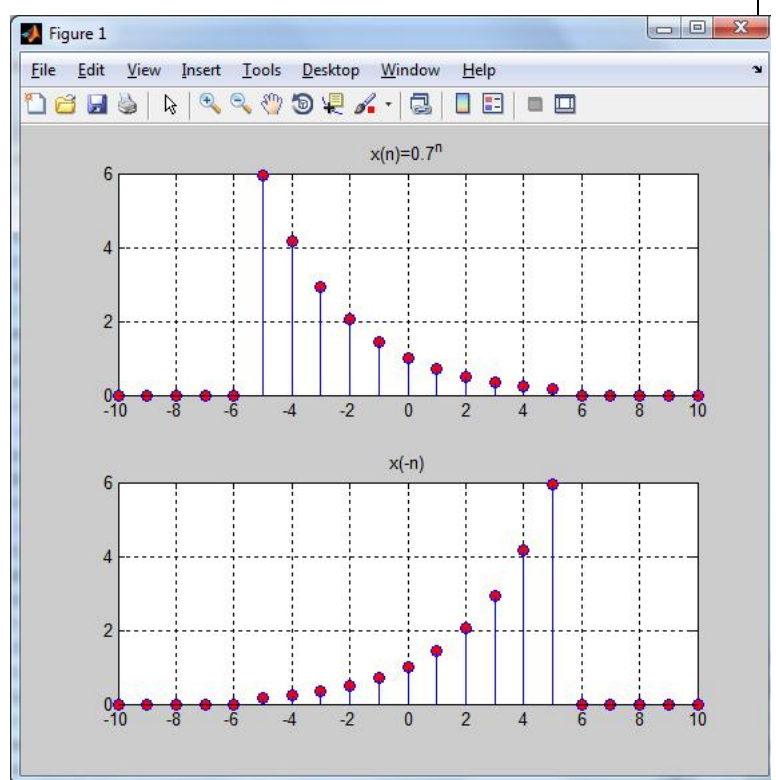
stem(n,x1,'MarkerFaceColor','red');

title('x(-n)');

grid on

% Εναλλακτικός τρόπος για x(-n)
% x1=x(length(n):-1:1);

```



2.3 Κλιμάκωση στον Χρόνο

Η κλιμάκωση στον χρόνο του σήματος $x(n)$ ορίζεται από τις σχέσεις $x(M \cdot n)$ και $x(M/n)$ (όπου $M, N > 0$ και ακέραιοι). Πιο συγκεκριμένα:

- Η σχέση $x(M \cdot n)$ εκφράζει διαίρεση συχνότητας και

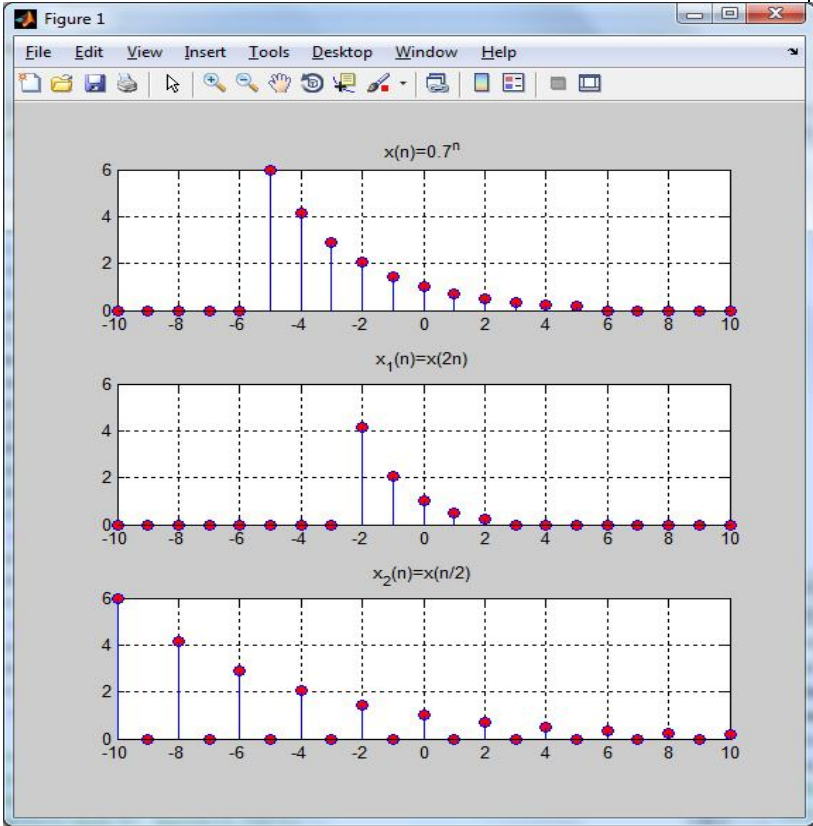
- Η σχέση $x(M/n)$ εκφράζει πολλαπλασιασμό συχνότητας

Υπενθύμιση: Οι αριθμοί M, N είναι ακέραιοι θετικοί

Άσκηση 3^η

Έστω το σήμα $x(n)=a^n$, όπου $a=0.7$ και ορίζεται στο διάστημα $n=[-5,5]$. Να σχεδιαστούν στο διάστημα $n=[-10,10]$ τα εξής σήματα:

- $x_1(n)=x(2n)$ και
- $x_2(n)=x(n/2)$,

| Κώδικας Matlab | Αποτέλεσμα |
|---|--|
| <pre> n=[-10:10]; k=find(n==0); x=zeros(1,length(n)); a=0.7; n1=[-5:5]; x(k-5:k+5)=a.^n1; x1=zeros(1,length(n)); %x(2n) pos=floor(5/2); rm=rem(5,2); x1(k-pos:k+pos)=x(k-5+rm:2:k+5- rm); x2=zeros(1,length(n)); %x(n/2) x2(1:2:end)=x(k-5:k+5); </pre> |  <p>Figure 1 displays three vertically stacked plots illustrating the signal $x(n)=0.7^n$ and its decimated and interpolated versions. The top plot shows $x(n)=0.7^n$ with n ranging from -10 to 10. The middle plot shows $x_1(n)=x(2n)$ with n ranging from -10 to 10. The bottom plot shows $x_2(n)=x(n/2)$ with n ranging from -10 to 10.</p> |

```

figure(1);

subplot(3,1,1);

stem(n,x,'MarkerFaceColor','red')
;

title('x(n)=0.7^n');

grid on

subplot(3,1,2);

stem(n,x1,'MarkerFaceColor','red'
);

title('x_1(n)=x(2n)');

grid on

subplot(3,1,3);

stem(n,x2,'MarkerFaceColor','red'
);

title('x_2(n)=x(n/2)');

grid on

```

2.4 Πολλαπλοί Μετασχηματισμοί

Πολύ συχνά θα χρειαστεί να χρησιμοποιήσουμε σήματα που εμπεριέχουν πολλαπλούς μετασχηματισμούς, λχ μετατόπιση, κλιμάκωση και αντιστροφή στον χρόνο. Στην περίπτωση αυτή και προκειμένου να σχεδιαστεί σωστά το σήμα, οι μετασχηματισμοί απαιτείται να εκτελεστούν αυστηρώς με την ακόλουθη σειρά:

1. Μετατόπιση χρόνου
2. Κλιμάκωση χρόνου
3. Αντιστροφή

Άσκηση 4η

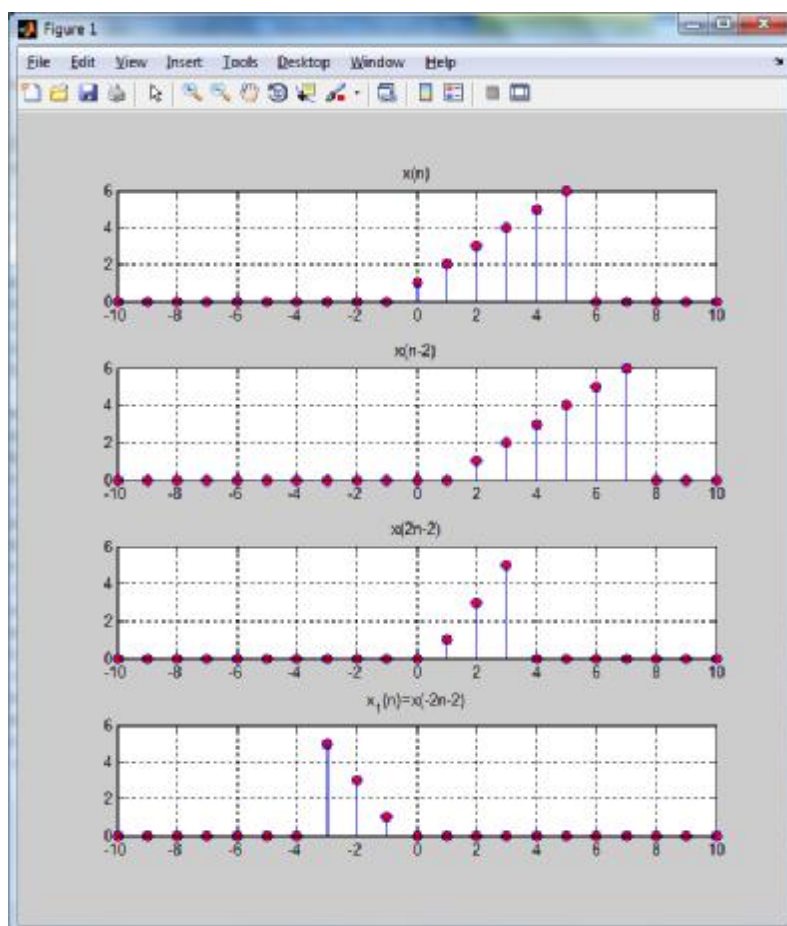
Δίνεται το σήμα $x(n) = \{1,2,3,4,5,6\}$ για $n=[0,5]$. Να σχεδιαστούν τα παρακάτω σήματα στο διάστημα $n=[-10,10]$:

- $x_1(n)=x(-2n-2)$
- $x_2(n)=|n-1| * x(-2n-2)$

Κώδικας Matlab

```
n=[-10:10];  
len=length(n);  
k=find(n==0);  
x=zeros(1,len);  
x(k:k+5)=[1:6];  
  
% x(n-2)  
x1=zeros(1,len);  
x1(k+2:end)=x(k:end-2);  
  
% x(2n-2)  
x2=zeros(1,len);  
x2(k-5:k+5)=x1(1:2:end);  
  
% x(-2n-2)  
x3=fliplr(x2);n3=-fliplr(n);  
  
figure(1);  
subplot(4,1,1);  
stem(n,x,'MarkerFaceColor','red');
```

Αποτέλεσμα



```

title('x(n)');

grid on

subplot(4,1,2);

stem(n,x1,'MarkerFaceColor','red');

title('x(n-2)');

grid on

subplot(4,1,3);

stem(n,x2,'MarkerFaceColor','red');

title('x(2n-2)');

grid on

subplot(4,1,4);

stem(n3,x3,'MarkerFaceColor','red');

title('x_1(n)=x(-2n-2)');

grid on

% Εναλλακτικός τρόπος

%n=[-10:10];

%x=[zeros(1,10) 1 2 3 4 5 6
zeros(1,5)];

%x1=zeros(1,length(n));

%k=find(n==0);

%for i=-10:10

%pos=-2*i-2+k;

%if (pos>= 1 & pos<= 21)

%    x1(k+i)=x(-2*i-2+k);

%end

```

```

%end

%figure(1)

%subplot(2,1,1),stem(n,x)

%subplot(2,1,2),stem(n,x1)

```

```

x4=abs(n-1).*x3;

figure(2)

subplot(3,1,1);

stem(n,x,'MarkerFaceColor','red');

title('x(n)');

grid on

subplot(3,1,2);

stem(n3,x3,'MarkerFaceColor','red');

title('x_1(n)=x(-2n-2)');

grid on

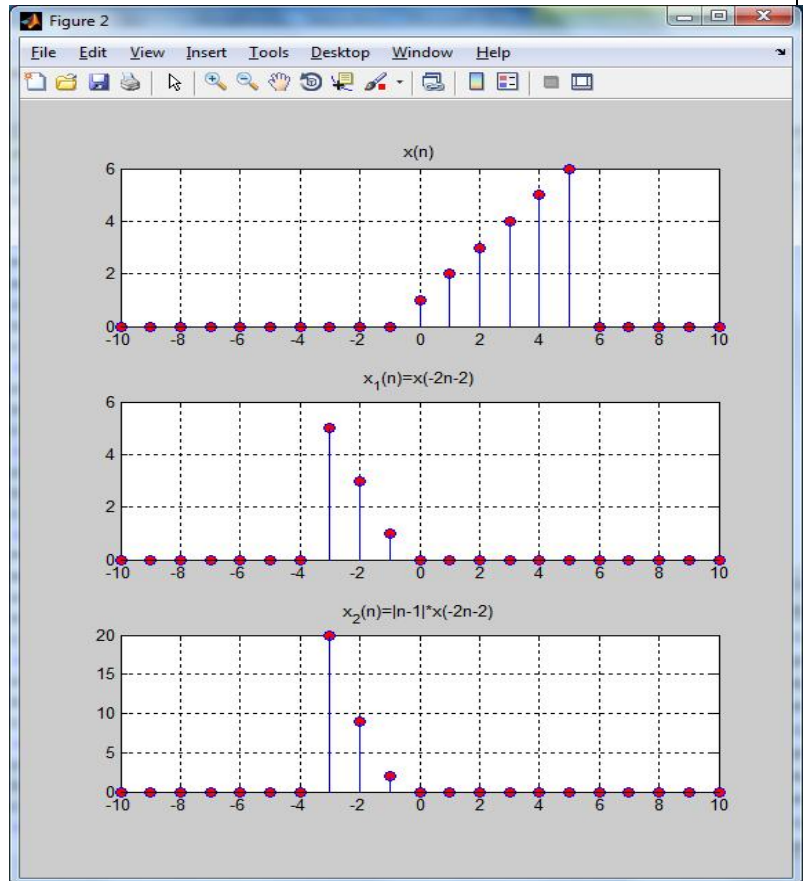
subplot(3,1,3);

stem(n,x4,'MarkerFaceColor','red');

title('x_2(n)=|n-1|*x(-2n-2)');

grid on

```



Άσκηση 5η

Να σχεδιαστούν στο διάστημα $n=[-10,10]$ τα σήματα:

- $x_0(n)=[u(-n+3) - u(-n-4)]$
- $x_1(n) = 2\cos(\pi n/2)*[u(-n+3) - u(-n-4)]$
- $x_2(n) = x(n-2) + \delta(n+1) + \delta(n-1) + \delta(n-3)$

Κώδικας Matlab

```

n=[-10:10];

len=length(n);

k=find(n==0);

u1=zeros(1,len);

u1(k-3:end)=1;

u1=fliplr(u1);

u2=zeros(1,len);

u2(k+4:end)=1;

u2=fliplr(u2);

n1=-fliplr(n);

x0=zeros(1,len);

x0=u1-u2;

x1=zeros(1,len);

x1=2*cos(pi/2*n).*(u1-u2);

%x(n-2)

x2=zeros(1,len);

x2(1+2:end)=x1(1:end-2);

d1=zeros(1,len);

d1(k-1)=1;

d2=zeros(1,len);

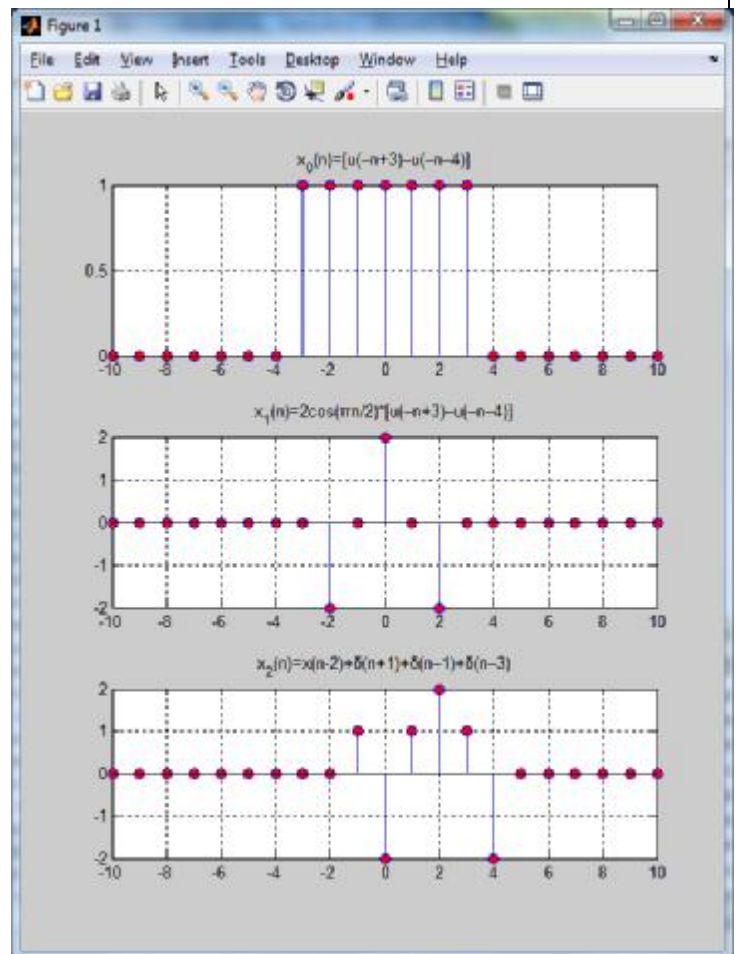
d2(k+1)=1;

d3=zeros(1,len);

d3(k+3)=1;

```

Αποτέλεσμα



```

x2=x2+d1+d2+d3;

figure(1); subplot(3,1,1);

stem(n1,
x0, 'MarkerFaceColor', 'red');

title('x_0(n)=[u(-n+3)-u(-n-4)]');

grid on; subplot(3,1,2);

stem(n1,
x1, 'MarkerFaceColor', 'red');

title('x_1(n)=2cos(mn/2)*[u(-n+3)-
u(-n-4)]');

grid on; subplot(3,1,3);

stem(n1,x2, 'MarkerFaceColor', 'red'
);

title('x_2(n)=x(n-2)+δ(n+1)+δ(n-
1)+δ(n-3)'); grid on;

```

3 Χαρακτηριστικά Μεγέθη ΣΔΧ

Δύο πολύ βασικά χαρακτηριστικά ενός ΣΔΧ είναι η ενέργεια και η ισχύς. Πιο συγκεκριμένα:

- Η **ολική ενέργεια** ενός ΣΔΧ δίνεται από τη σχέση:

$$E_x = \sum_{n=-\infty}^{+\infty} |x[n]|^2$$

- Αν το σήμα είναι πραγματικό τότε ισχύει:

$$E_x = \sum_{n=-\infty}^{+\infty} x^2[n]$$

- Η μέση ισχύς ενός ΣΔΧ δίνεται από τη σχέση:

$$P_x = \lim_{N \rightarrow \infty} \frac{1}{2N + 1} \sum_{n=-N}^N |x[n]|^2$$

- Γενικότερα, η μέση ισχύς ενός ΣΔΧ σε ένα διάστημα $n=[N_1, N_2]$ δίνεται από τη σχέση:

$$P_x = \frac{1}{N_2 - N_1 + 1} \sum_{n=N_1}^{N_2} |x[n]|^2$$

- Αν το σήμα είναι πραγματικό, τότε η μέση ισχύς σε ένα διάστημα $n=[N_1, N_2]$ δίνεται από τη σχέση:

$$P_x = \frac{1}{N_2 - N_1 + 1} \sum_{n=N_1}^{N_2} x^2[n]$$

Άσκηση 6^η

Να υπολογιστεί η ενέργεια και η μέση ισχύς των παρακάτω σημάτων στο διάστημα $n=[-5, 10]$:

- Δίνεται το πραγματικό σήμα $x_1(n) = \{1, 2, 3, -2, -4, 1, 2\}$ με $n=[-2, 4]$.
- Δίνεται το μιγαδικό σήμα $x_2(n) = \{1+j, 2, 3-j, -2, -4-2j, 1, 2-2j\}$ με $n=[-2, 4]$.
- Δίνεται το πραγματικό σήμα $x_3(n) = 2\sin(\pi n/4) * [u(n+2) - u(n-6)]$

| Κώδικας Matlab | Αποτέλεσμα |
|--|---|
| <pre>N1=-5; N2=10; n=[N1:N2]; x1=[zeros(1,3) 1 2 3 -2 -4 1 2 zeros(1,6)]; x2=[zeros(1,3) 1+j 2 3-j -2 -4-2j 1 2-2j</pre> | <pre>x1(n)={1,2,3,-2,-4,1,2}, n=[-2,4] E1=39.000000 P1=2.437500 x2(n)={1+j,2,3-j,-2,-4-2j,1,2-2j}, n=[-2,4]</pre> |

| | |
|---|--|
| <pre> zeros(1,6)]; u1=zeros(1,length(n)); u1(n>=-2)=1; u2=zeros(1,length(n)); u2(n>=6)=1; x3=zeros(1,length(n)); x3=2*sin(pi/4*n).*(u1-u2); E1=sum(x1.^2); E2=sum(abs(x2).^2); E3=sum(x3.^2); P1=1/(N2-N1+1)*sum(x1.^2); P2=1/(N2-N1+1)*sum(abs(x2).^2); P3=1/(N2-N1+1)*sum(x3.^2); str1 = fprintf('x1(n)={1,2,3,-2,-4,1,2}, n=[-2,4]\nE1=%f\nP1=%f\n', E1,P1); str2 = fprintf('x2(n)={1+j,2,3-j,-2,-4-2j,1,2-2j}, n=[-2,4]\nE2=%f\nP2=%f\n', E2,P2); str3 = fprintf('x3(n)=2sin(pi/4)[u(n+2)-u(n-6)], n=[-2,4]\nE3=%f\nP3=%f\n', E3,P3); </pre> | <pre> E2=49.000000 P2=3.062500 x3(n)=2sin(pi/4)[u(n+2)-u(n-6)], n=[-2,4] E3=16.000000 P3=1.000000 </pre> |
|---|--|

4 Άλυτες Ασκήσεις

- Δίνεται το σήμα $x(n) = 5\sin(\pi n/3)[u(n+12)-u(n-20)]$ για $n=[0,5]$. Να σχεδιαστούν τα παρακάτω σήματα στο διάστημα $n=[-30,30]$:
 - $x_1(n)=x(-2n-2)$
 - $x_2(n)=x(n-2)+x(n+2)$
- Για τα προηγούμενα σήματα να υπολογίσετε την ενέργεια και τη μέση ισχύ τους στο διάστημα $n=[-30,30]$

5 Επίλογος – Σύνδεση με τα Επόμενα

Στην παρούσα εργαστηριακή άσκηση καλύφθηκαν οι βασικοί χρονικοί μετασχηματισμοί των ΣΔΧ, όπως η μετατόπιση, η κλιμάκωση, η αντιστροφή, καθώς και ο συνδυασμός πολλαπλών μετασχηματισμών. Επίσης, μελετήθηκαν χαρακτηριστικά μεγέθη των ΣΔΧ, όπως η ενέργεια και η ισχύς.

Στην επόμενη εργαστηριακή άσκηση θα παρουσιαστούν οι βασικές ταξινομίες, τα χαρακτηριστικά και οι ιδιότητες των συστημάτων διακριτού χρόνου.

6 Βιβλιογραφία

Hayes , Μ. Η. (2002). *Ψηφιακή Επεξεργασία Σήματος*. (Α. Δημητριάδης, Μεταφρ.) Θεσσαλονίκη: Τζιόλα.

Hsu, Η. Ρ. (2002). *Αναλογικές και Ψηφιακές Επικοινωνίες*. Θεσσαλονίκη: Τζιόλα.

Jayant, Ν. S., & Noll, Ρ. (1984). *Digital Coding of Waveforms: Principles and Applications to Speech and Video*. Prentice Hall.

Mano, Μ. Μ., & Ciletti, Μ. D. (2012). *Digital Design (5 εκδ.)*. Pearson Education.

MathWorks - DSP. (2017). Ανάκτηση 2 10, 2017, από Getting Started with Matlab DSP: <https://www.mathworks.com/help/dsp/getting-started-with-dsp-system-toolbox.html>

Proakis, J. G., & Salehi, Μ. (2008). *Digital Communications (5 εκδ.)*. McGraw-Hill Higher Education.

Μουρτζόπουλος, Ι. (2012). *Ψηφιακή Τεχνολογία Ήχου*. Πάτρα: Πανεπιστήμιο Πατρών.

Εργαστηριακή Άσκηση 4: Ιδιότητες Συστημάτων

1 Εισαγωγικά Στοιχεία

1.1 Σκοπός

Σκοπός της παρούσας εργαστηριακής άσκησης είναι η εξοικείωση των φοιτητών – σπουδαστών με τις βασικές ιδιότητες των συστημάτων διακριτού χρόνου, η γραμμικότητα, αιτιατότητα, ευστάθεια κοκ.

1.2 Αναμενόμενα Μαθησιακά Αποτελέσματα

Με την ολοκλήρωση της παρούσας εργαστηριακής άσκησης, ο φοιτητής – σπουδαστής θα είναι σε θέση να ελέγχει αν ένα σύστημα διακριτού χρόνου έχει τις εξής ιδιότητες:

- Έχει μνήμη
- Είναι ομογενές
- Έχει την ιδιότητα της υπέρθεσης (επαλληλίας)
- Είναι γραμμικό
- Είναι χρονικά αμετάβλητο
- Είναι αιτιατό
- Είναι ευσταθές

1.3 Εκτιμώμενη Διάρκεια

Ο εκτιμώμενος χρόνος που απαιτείται από τον φοιτητή – σπουδαστή για την ολοκλήρωση της παρούσας εργαστηριακής άσκησης είναι 2 διδακτικές ώρες.

1.4 Προαπαιτούμενα

Διάλεξη 3 της θεωρίας.

1.5 Εισαγωγή

Στην άσκηση αυτή θα ασχοληθούμε με τις ιδιότητες των συστημάτων διακριτού χρόνου. Όπως θα δούμε σε επόμενες ενότητες, προκειμένου να μπορέσουμε να μελετήσουμε ένα σύστημα, συχνά απαιτείται το σύστημα αυτό να ικανοποιεί κάποιες προϋποθέσεις - ιδιότητες, όπως λχ να είναι γραμμικό, χρονικά αμετάβλητο, ευσταθές και αιτιατό. Είναι σημαντικό, λοιπόν, να μπορούμε να ελέγχουμε αν ένα σύστημα διακριτού χρόνου έχει τις ιδιότητες αυτές.

1.6 Κατάλογος Χρήσιμων Συναρτήσεων Matlab

Κατά την εκπόνηση των ασκήσεων χρησιμοποιούνται οι ακόλουθες συναρτήσεις – εντολές του Matlab:

| Συνάρτηση | Επεξήγηση |
|-----------|--|
| length | Επιστρέφει το μήκος ενός πίνακα-διανύσματος |
| zeros | Δημιουργία πίνακα – διανύσματος με μηδενικά |
| figure | Δημιουργεί νέο παράθυρο για την απεικόνιση γραφικών παραστάσεων |
| plot | Δημιουργεί μία γραφική παράσταση |
| subplot | Τμηματοποίηση ενός παραθύρου για γραφικές παραστάσεις σε γραμμές και στήλες, προκειμένου στο ίδιο παράθυρο να απεικονιστούν πολλαπλές γραφικές παραστάσεις |
| stem | Γραφική παράσταση διακριτών σημάτων – ακολουθιών |
| title | Τοποθετεί τίτλο σε μια γραφική παράσταση |
| grid | Προσθέτει – αφαιρεί πλέγμα στις γραφικές παραστάσεις, βελτιώνοντας την ανάγνωσή τους |

2 Ιδιότητες Συστημάτων Διακριτού Χρόνου

Στη μελέτη των συστημάτων διακριτού χρόνου υπάρχει ένας αριθμός ιδιοτήτων οι οποίες χαρακτηρίζουν τα συστήματα αυτά. Οι βασικότερες ιδιότητες για τις οποίες εξετάζουμε ένα σύστημα είναι αν το σύστημα:

1. Είναι δυναμικό ή στατικό (Αν το σύστημα έχει μνήμη ή όχι)
2. Είναι ομογενές
3. Έχει την ιδιότητα της υπέρθεσης (επαλληλίας)
4. Είναι γραμμικό
5. Είναι χρονικά αμετάβλητο
6. Είναι αιτιατό
7. Είναι ευσταθές

2.1 Δυναμικά και Στατικά Συστήματα

Ένα σύστημα είναι στατικό (δηλαδή χωρίς μνήμη) όταν κάθε χρονική στιγμή η έξοδος του εξαρτάται από την είσοδο την ίδια χρονική στιγμή. Το αντίθετο συμβαίνει για ένα δυναμικό σύστημα (δηλαδή με μνήμη). Για παράδειγμα το σύστημα $y(n) = x^2(n)$ είναι στατικό, αφού έξοδος και είσοδος αναφέρονται στον ίδιο χρόνο. Ωστόσο, το σύστημα $y(n) = x(n) + x(n-1)$ είναι δυναμικό, επειδή η έξοδος εξαρτάται και από περασμένες τιμές της εισόδου, δηλαδή την $x(n-1)$.

Άσκηση 1^η

Εξετάστε αν τα παρακάτω συστήματα είναι δυναμικά. Ως είσοδο θεωρήστε το σήμα $x(n) = \{1, 2, 3, 4, 5\}$, για $n = [0, 4]$. Επίσης, η μελέτη να γίνει στο διάστημα $n = [-5, 10]$.

- a) $y(n) = x^2(n)$
- b) $y(n) = x(n+1)$
- c) $y(n) = x(n-1)$

| Κώδικας Matlab | Αποτέλεσμα |
|----------------|------------|
|----------------|------------|


```
% Σήμα εισόδου
```

```
n=-5:10;
```

```
x=[zeros(1,5) 1:5 zeros(1,6)];
```

```
% έξοδος  $y(n)=x^2(n)$ 
```

```
y1=x.*x;
```

```
% έξοδος  $y(n)=x(n+1)$ 
```

```
y2=[x(2:end) 0];
```

```
% έξοδος  $y(n)=x(n-1)$ 
```

```
y3=[0 x(1:end-1)];
```

```
figure(1);
```

```
subplot(2,1,1);
```

```
stem(n,x);grid on;
```

```
title('x(n)');
```

```
subplot(2,1,2);
```

```
stem(n,y1);grid on;
```

```
title('y(n)=x^2(n)');
```

```
figure(2);
```

```
subplot(2,1,1);
```

```
stem(n,x);grid on;
```

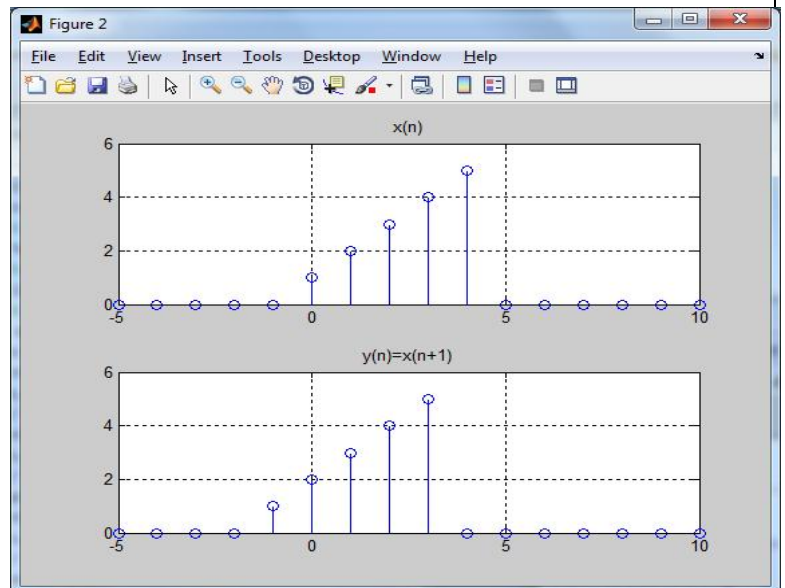
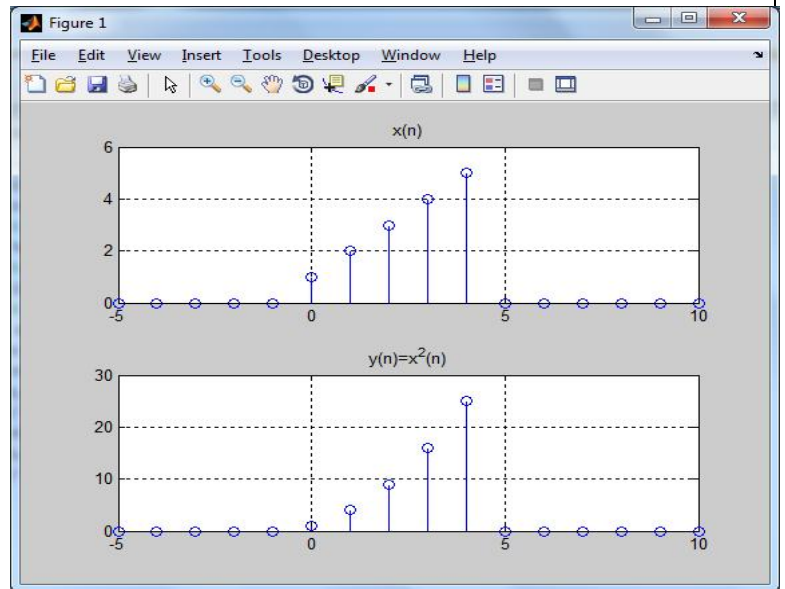
```
title('x(n)');
```

```
subplot(2,1,2);
```

```
stem(n,y2);grid on;
```

```
title('y(n)=x(n+1)');
```

```
figure(3);
```



```

subplot(2,1,1);

stem(n,x);grid on;

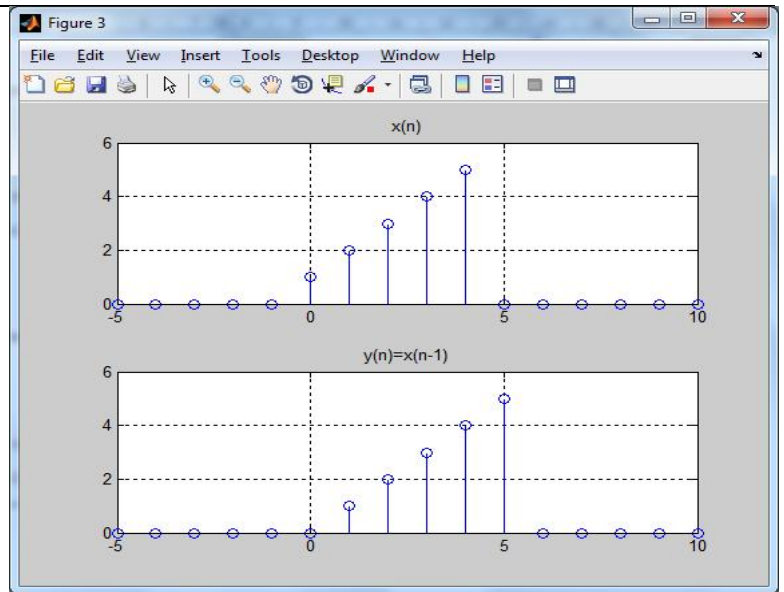
title('x(n)');

subplot(2,1,2);

stem(n,y3);grid on;

title('y(n)=x(n-1)');

```



Παρατήρηση: Το πρώτο σύστημα είναι στατικό, ενώ τα δύο επόμενα είναι δυναμικά.

2.2 Ομογενή Συστήματα

Ένα σύστημα λέγεται ομογενές όταν ο πολλαπλασιασμός της εισόδου $x(n)$ με μία σταθερά c , οδηγεί σε πολλαπλασιασμό της εξόδου με την ίδια σταθερά, δηλαδή θα πρέπει να ισχύει:

$$T[cx(n)] = cT[x(n)], \text{ για οποιαδήποτε σταθερά } c \text{ και είσοδο } x(n)$$

Άσκηση 2^η

Εξετάστε αν τα παρακάτω συστήματα είναι ομογενή. Ως είσοδο θεωρήστε το σήμα $x(n)=\{1 \ 2 \ 3 \ 4 \ 5\}$, για $n=[0,4]$. Επίσης, η μελέτη να γίνει στο διάστημα $n=[-5, 10]$ και ως σταθερά χρησιμοποιήστε την $c=2$.

- $y(n)=x^2(n)$
- $y(n)=x(n+1)$

Κώδικας Matlab

```
% Σήμα εισόδου

n=-5:10;

x=[zeros(1,5) 1:5 zeros(1,6)];

c=2;

% Για  $y(n)=x^2(n)$ 

y1=(c*x).^2;

y1_hom=c*(x.^2);

figure(1);

subplot(2,1,1);

stem(n,y1);grid on;

title('y(n)=cx^2(n)');

subplot(2,1,2);

stem(n,y1_hom);grid on;

title('y(n)=[cx(n)]^2');

% Για  $y(n)=x(n+1)$ 

xb=[x(2:end) 0];

y2=c*xb;

y2_hom=c*xb;

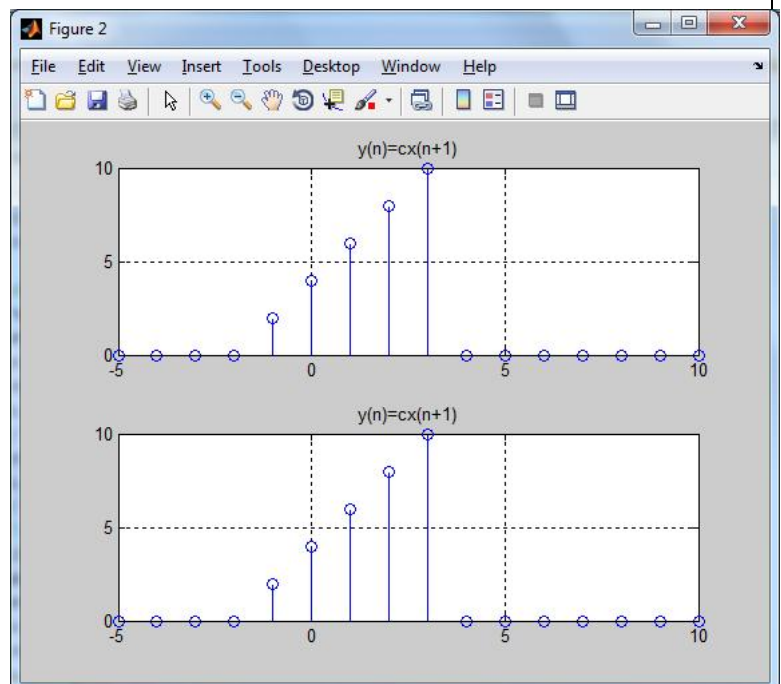
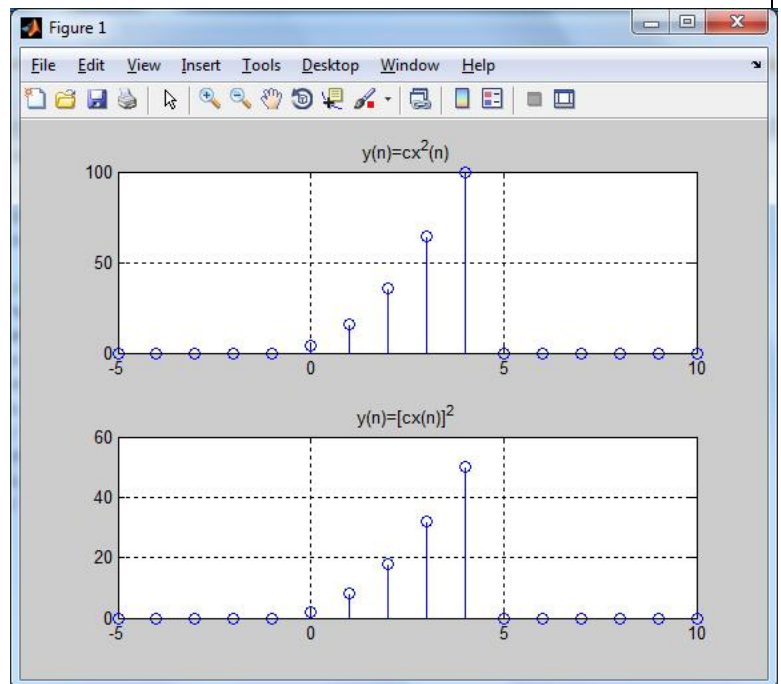
figure(2);

subplot(2,1,1);

stem(n,y2);grid on;

title('y(n)=cx(n+1)');
```

Αποτέλεσμα



Παρατήρηση: Το πρώτο σύστημα δεν είναι ομογενές, ενώ το δεύτερο είναι ομογενές.

| | |
|---|--|
| <pre>subplot(2,1,2); stem(n,y2_hom);grid on; title('y(n)=cx(n+1)');</pre> | |
|---|--|

2.3 Υπέρθθεση (Επαλληλία)

Για ένα σύστημα ισχύει η αρχή της υπέρθεσης, όταν η έξοδος του στο άθροισμα ενός αριθμού εισόδων ισούται με το άθροισμα των επιμέρους εξόδων σε καθεμία από αυτές τις εισόδους, δηλαδή θα πρέπει να ισχύει:

$$T[x_1(n)+x_2(n)] = T[x_1(n)] + T[x_2(n)], \text{ για οποιαδήποτε σήματα εισόδου } x_1(n) \text{ και } x_2(n)$$

Άσκηση 3^η

Εξετάστε αν για τα παρακάτω συστήματα ισχύει η αρχή της υπέρθεσης. Ως εισόδους θεωρήστε τα σήματα $x_1(n)=\{1 \ 2 \ 3 \ 4 \ 5\}$, για $n=[0,4]$ και $x_2(n)=u(n)-u(n-6)$. Επίσης, η μελέτη να γίνει στο διάστημα $n=[-5, 10]$.

- a) $y(n)=x^2(n)$
- b) $y(n)=x(n+1)+x(n)+x(n-1)$

| | |
|-----------------------|-------------------|
| Κώδικας Matlab | Αποτέλεσμα |
|-----------------------|-------------------|

```

n=-5:10;

x1=[zeros(1,5) 1:5 zeros(1,6)];

x2=[zeros(1,5)          ones(1,6)
zeros(1,5)];

% Για y(n)=x^2(n)

ya=(x1+x2).^2;

ya_h=x1.^2+x2.^2;

figure(1);

subplot(2,1,1);

stem(n,ya);grid on;

title('y_{total}(n)');

subplot(2,1,2);

stem(n,ya_h);grid on;

title('y(n)=y1(n)+y2(n)');

% Για y(n)=x(n+1)+x(n)+x(n-1)

x11=[x1(2:end) 0];

x12=[0 x1(1:end-1)];

x21=[x2(2:end) 0];

x22=[0 x2(1:end-1)];

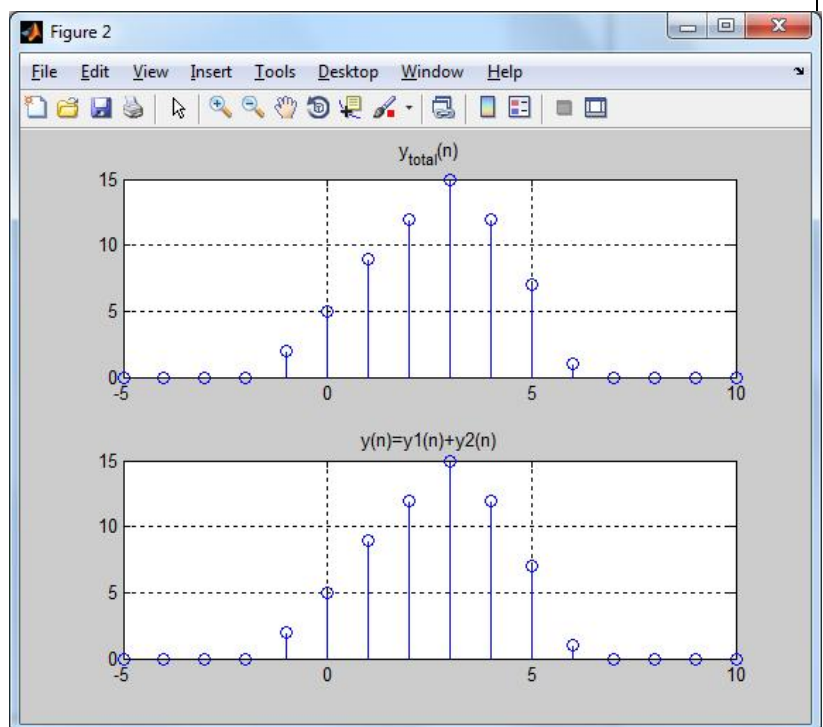
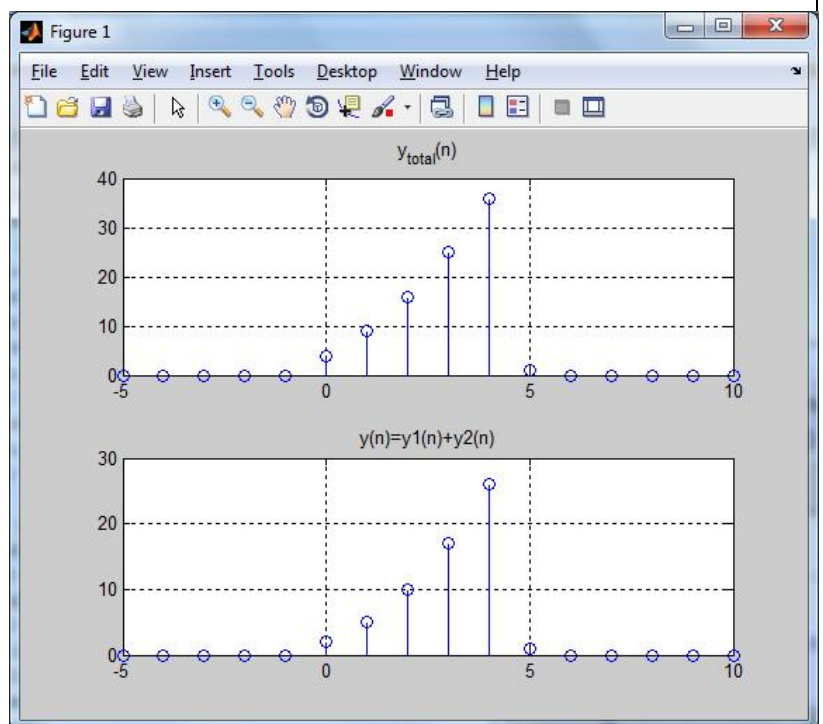
xb1=x11+x1+x12;

xb2=x21+x2+x22;

yb_h=xb1+xb2;

xb=x1+x2;

```



Παρατήρηση: Για το πρώτο σύστημα δεν ισχύει η αρχή της υπέρθεσης, ενώ για το δεύτερο ισχύει.

```

xb_plus1=[xb(2:end) 0];

xb_minus1=[0 xb(1:end-1)];

yb_total=xb_plus1+xb+xb_minus1;

figure(2);

subplot(2,1,1);

stem(n,yb_total);grid on;

title('y_{total}(n)');

subplot(2,1,2);

stem(n,yb_h);grid on;

title('y(n)=y1(n)+y2(n)');

```

2.4 Γραμμικότητα

Ένα σύστημα λέγεται γραμμικό όταν είναι ομογενές και ισχύει για αυτό η αρχή της υπέρθεσης, δηλαδή θα πρέπει να ισχύει:

$$T[ax_1(n)+bx_2(n)] = aT[x_1(n)] + bT[x_2(n)], \text{ για οποιαδήποτε } x_1(n), x_2(n) \text{ και σταθερές } a, b.$$

Άσκηση 4η

Εξετάστε αν τα παρακάτω συστήματα είναι γραμμικά. Ως εισόδους θεωρήστε τα σήματα $x_1(n)=\{1 \ 2 \ 3 \ 4 \ 5\}$, για $n=[0,4]$ και $x_2(n)=u(n)-u(n-6)$. Επίσης, η μελέτη να γίνει στο διάστημα $n=[-5, 10]$ και ως σταθερές να χρησιμοποιηθούν οι $a=2$ και $b=3$.

- $y(n)=x^2(n)$
- $y(n)=x(n)+x(n-1)$

Κώδικας Matlab

Αποτέλεσμα

```

a=2; b=3; n=-5:10;

x1=[zeros(1,5) 1:5 zeros(1,6)];

x2=[zeros(1,5)          ones(1,6)
zeros(1,5)];

% Για  $y(n)=x^2(n)$ 

ya=a*x1.*x1+b*x2.*x2;

xa_total= a*x1+b*x2;

ya_total=xa_total.^2;

figure(1);

subplot(2,1,1);

stem(n,ya);grid on;

title('aT[x1(n)]+bT[x2(n)]');

subplot(2,1,2);

stem(n,ya_total);grid on;

title('T[ax1(n)+bx2(n)]');

% Για  $y(n)=x(n)+x(n-1)$ 

xb1=x1+[0 x1(1:end-1)];

xb2=x2+[0 x2(1:end-1)];

yb=a*xb1+b*xb2;

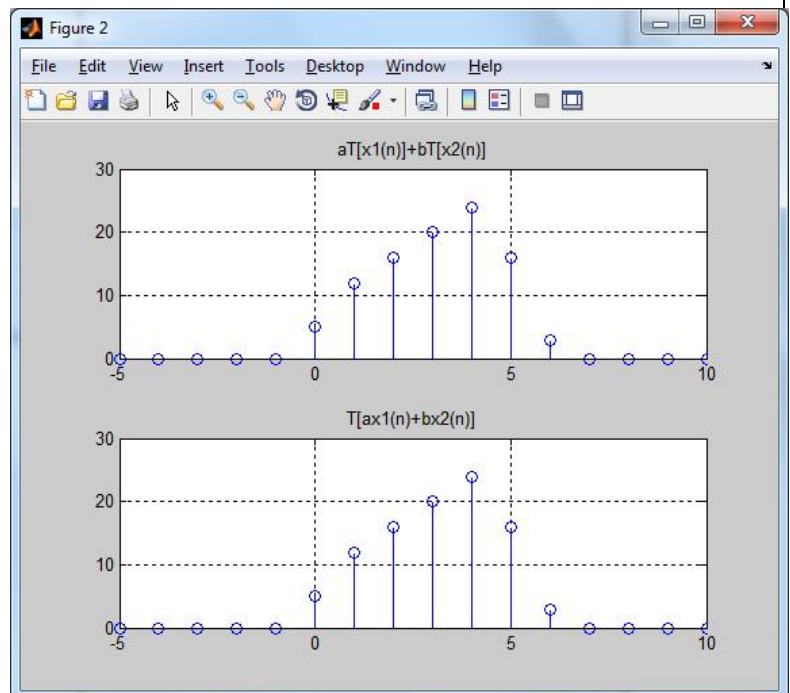
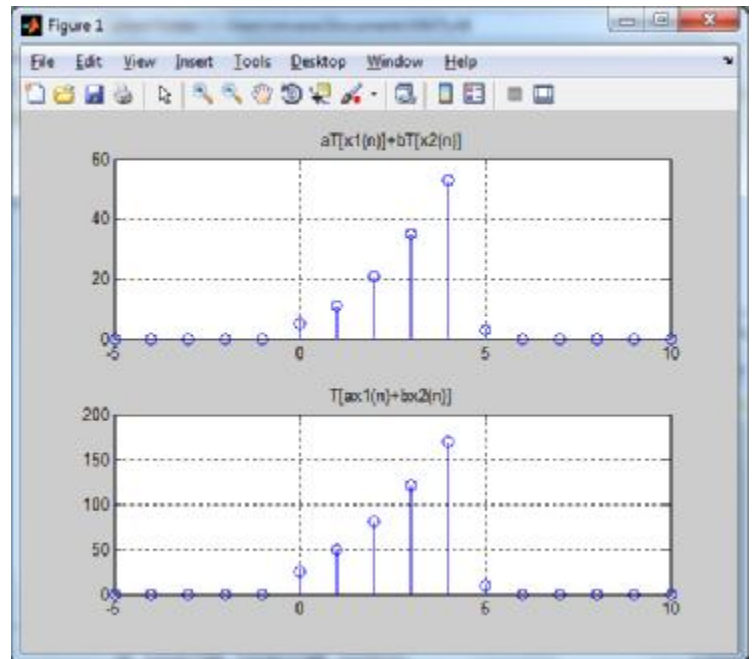
xb_total1=a*xb1+b*xb2;

xb_total2=[0 xb_total1(1:end-1)];

yb_total=xb_total1+xb_total2;

figure(2)

```



Παρατήρηση: Το πρώτο σύστημα δεν είναι γραμμικό, ενώ το δεύτερο είναι γραμμικό.

```

subplot(2,1,1);

stem(n,yb);grid on;

title('aT[x1(n)]+bT[x2(n)]');

subplot(2,1,2);

stem(n,yb_total);grid on;

title('T[ax1(n)+bx2(n)]');

```

2.5 Χρονικά Αμετάβλητα Συστήματα

Ένα σύστημα λέγεται χρονικά αμετάβλητο (ή αμετάβλητο στη μετατόπιση) όταν σε μία μετατόπιση (καθυστέρηση) στην είσοδο κατά n_0 , οδηγεί σε μία ίδια μετατόπιση κατά n_0 στην έξοδο.

Άσκηση 5η

Εξετάστε αν τα παρακάτω συστήματα είναι αμετάβλητα στη μετατόπιση. Ως εισόδους θεωρήστε για το πρώτο σύστημα το σήμα $x_1(n)=\{1\ 2\ 3\ 4\ 5\}$, για $n=[0,4]$, ενώ για το δεύτερο σύστημα το σήμα $x_2(n)=\delta(n)$. Η μελέτη να γίνει στο διάστημα $n=[-5, 10]$ για καθυστέρηση κατά $n_0=1$.

- $y(n)=x^2(n)$
- $y(n)=x(n)+x(-n)$

| Κώδικας Matlab | Αποτέλεσμα |
|----------------|------------|
|----------------|------------|


```

n=-5:10;

x1=[zeros(1,5) 1:5 zeros(1,6)];

x2=zeros(1,length(n));

x2(n==0)=1;

% Για y(n)=x^2(n)

y1=x1.*x1;

x1_delayed=[0 x1(1:end-1)];

y1_delayed=x1_delayed.*x1_delayed;

figure(1)

subplot(2,2,1)

stem(n,x1);grid on;

title('x_1(n)');

subplot(2,2,2)

stem(n,y1);grid on;

title('y_1(n)');

subplot(2,2,3)

stem(n,x1_delayed);grid on;

title('x_1(n-1)');

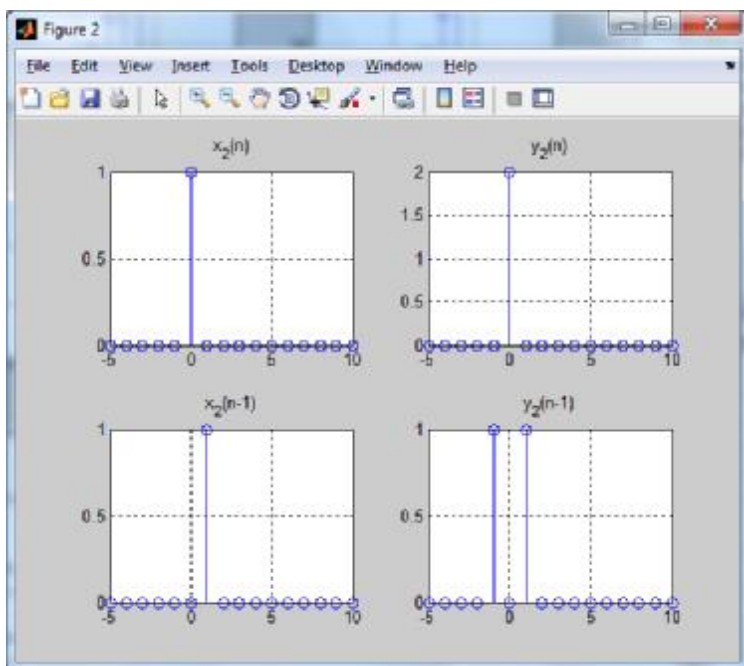
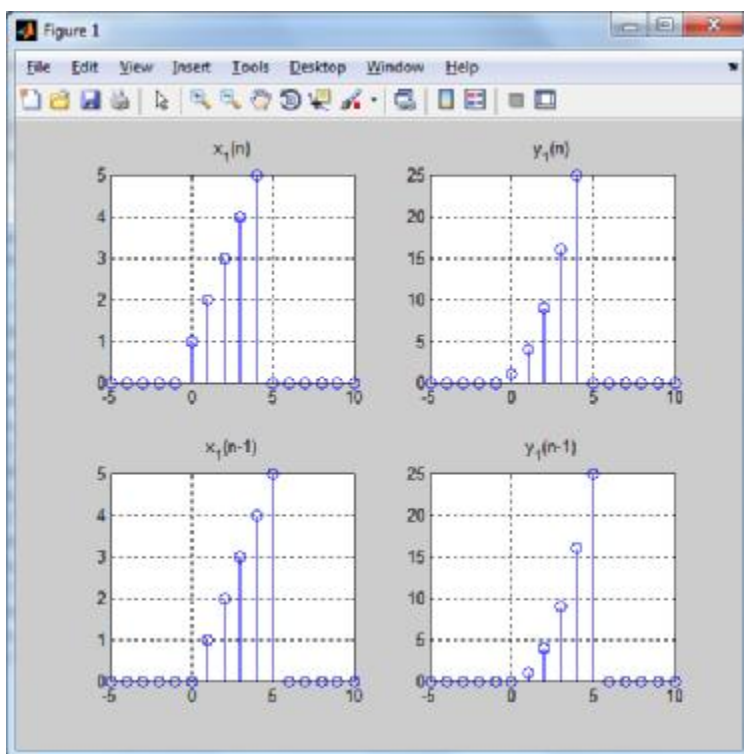
subplot(2,2,4)

stem(n,y1_delayed);grid on;

title('y_1(n-1)');

% Για y(n)=x(n)+x(-n), με x(n)=δ(n)

```



Παρατήρηση: Το πρώτο σύστημα είναι αμετάβλητο στη μετατόπιση, ενώ το δεύτερο όχι.

```

% άρα  $y(n)=\delta(n)+\delta(-n)=2\delta(n)$ 

y2=x2+x2; % επειδή  $\delta(-n)=\delta(n)$ 

% Για  $x(n-1)=\delta(n-1)$ 

%  $y(n)=\delta(n-1)+\delta(-n-1)=\delta(n-1)+\delta(n+1)$ 

x2_delayed=[0 x2(1:end-1)];

y2_delayed=x2_delayed+[x2(2:end) 0];

figure(2)

subplot(2,2,1), stem(n,x2);

grid on; title('x_2(n)');

subplot(2,2,2), stem(n,y2);

grid on; title('y_2(n)');

subplot(2,2,3), stem(n,x2_delayed);

grid on; title('x_2(n-1)');

subplot(2,2,4), stem(n,y2_delayed);

grid on; title('y_2(n-1)');

```

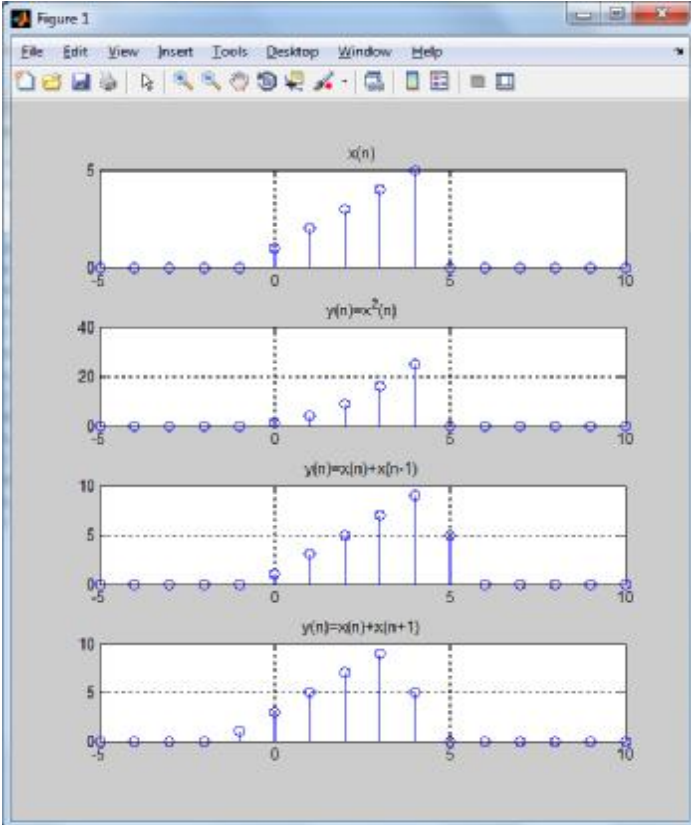
2.6 Αιτιατότητα

Ένα σύστημα λέγεται αιτιατό όταν η έξοδό του κάθε χρονική στιγμή n_0 εξαρτάται μόνο από την είσοδο μέχρι τη χρονική στιγμή n_0 . Για παράδειγμα το σύστημα $y(n)=x(n)+x(n-1)$ είναι αιτιατό, ενώ το σύστημα $y(n)=x(n)+x(n+1)$ δεν είναι αιτιατό.

Άσκηση 6η

Εξετάστε αν τα παρακάτω συστήματα είναι αιτιατά. Ως είσοδο θεωρήστε το σήμα $x(n)=\{1\ 2\ 3\ 4\ 5\}$, για $n=[0,4]$. Η μελέτη να γίνει στο διάστημα $n=[-5, 10]$.

- $y(n)=x^2(n)$
- $y(n)=x(n)+x(n-1)$
- $y(n)=x(n)+x(n+1)$

| Κώδικας Matlab | Αποτέλεσμα |
|--|--|
| <pre> n=-5:10; x=[zeros(1,5) 1:5 zeros(1,6)]; y1=x.*x; y2=x+[0 x(1:end-1)]; y3=x+[x(2:end) 0]; figure(1) subplot(4,1,1); stem(n,x);grid on; title('x(n)'); subplot(4,1,2); stem(n,y1);grid on; title('y(n)=x^2(n)'); subplot(4,1,3); stem(n,y2);grid on; title('y(n)=x(n)+x(n-1)'); subplot(4,1,4); stem(n,y3);grid on; title('y(n)=x(n)+x(n+1)'); </pre> |  <p>Παρατήρηση: Στο παραπάνω παράθυρο, η πρώτη γραφική παράσταση είναι του σήματος εισόδου $x(n)$. Οι τρεις επόμενες γραφικές παραστάσεις αφορούν τις εξόδους των τριών συστημάτων που μελετάμε, όπου τα δύο πρώτα συστήματα φαίνεται ότι είναι αιτιατά.</p> <p>Για το τρίτο σύστημα παρατηρούμε ότι την χρονική στιγμή $n= -1$ η έξοδος εξαρτάται από μελλοντικές τιμές της εισόδου, οπότε δεν είναι αιτιατό.</p> |

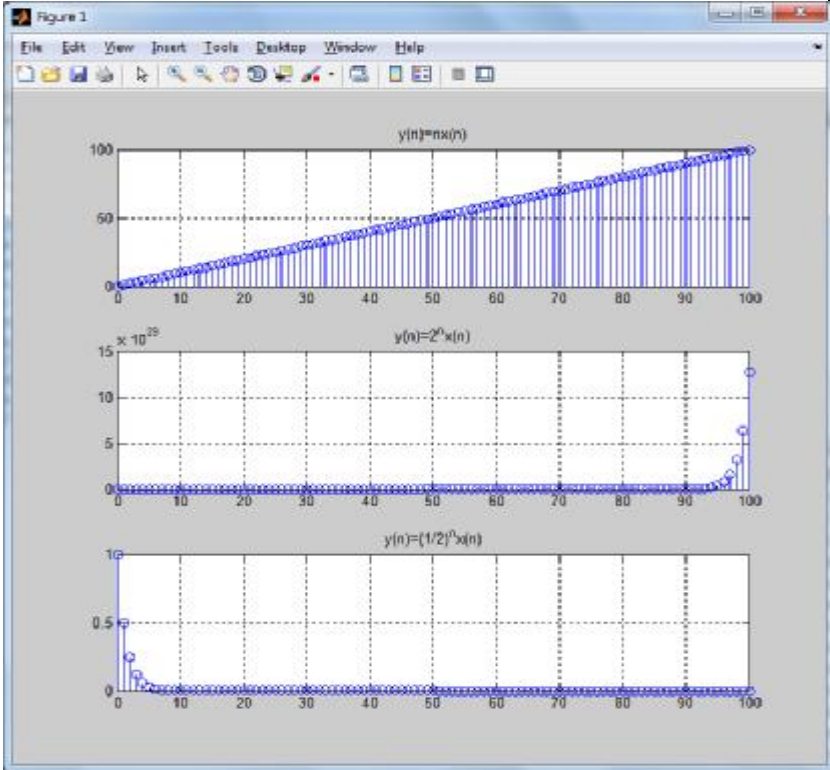
2.7 Ευστάθεια

Ένα σύστημα λέγεται ευσταθές όταν για κάθε πεπερασμένη είσοδο, η έξοδος είναι επίσης πεπερασμένη.

Άσκηση 7η

Εξετάστε αν τα παρακάτω συστήματα είναι ευσταθή. Ως είσοδο θεωρήστε την βηματική ακολουθία $u(n)$, στο διάστημα $n=[0,100]$.

- $y(n)=nx(n)$
- $y(n)=2^n x(n)$
- $y(n)=(1/2)^n x(n)$

| Κώδικας Matlab | Αποτέλεσμα |
|---|--|
| <pre>n=0:100; x=ones(1,101); y1=n.*x; y2=2.^n.*x; y3=(1/2).^n.*x; figure(1) subplot(3,1,1); stem(n,y1);grid on; title('y(n)=nx(n)'); subplot(3,1,2); stem(n,y2);grid on; title('y(n)=2^nx(n)'); subplot(3,1,3); stem(n,y3);grid on; title('y(n)=(1/2)^nx(n)');</pre> |  <p>Παρατήρηση: Τα δύο πρώτα συστήματα δεν είναι ευσταθή αφού η έξοδός τους τείνει να αυξάνει διαρκώς, ενώ το τρίτο είναι ευσταθές.</p> |

3 Άλυτες Ασκήσεις

1. Εξετάστε αν τα παρακάτω συστήματα είναι ευσταθή και αιτιατά. Ως είσοδο θεωρήστε την ακολουθία $x(n)=\delta(n+3)+2\delta(n+2)+3\delta(n+1)+\delta(n)+2\delta(n-2)+2\delta(n-3)$, στο διάστημα $n=[0,100]$.

$$\alpha) y(n)=nx(n) \quad \beta) y(n)=2^n x(n) \quad \gamma) y(n)=(1/2)^n x(n)$$

4 Επίλογος – Σύνδεση με τα Επόμενα

Στην παρούσα εργαστηριακή άσκηση καλύφθηκαν οι ιδιότητες ενός συστήματος διακριτού χρόνου, όπως αν είναι στατικό ή δυναμικό, η ομογένεια, η υπέρθεση, η γραμμικότητα, η χρονική αμεταβλητότητα, η αιτιατότητα και η ευστάθεια.

Στην επόμενη εργαστηριακή άσκηση θα παρουσιαστεί η πράξη της γραμμικής συνέλιξης, η οποία χρησιμοποιείται για τον υπολογισμό της εξόδου των γραμμικών και χρονικά αμετάβλητων συστημάτων διακριτού χρόνου.

5 Βιβλιογραφία

Hayes , Μ. Η. (2002). *Ψηφιακή Επεξεργασία Σήματος*. (Α. Δημητριάδης, Μεταφρ.) Θεσσαλονίκη: Τζιόλα.

Hsu, Η. Ρ. (2002). *Αναλογικές και Ψηφιακές Επικοινωνίες*. Θεσσαλονίκη: Τζιόλα.

Jayant, Ν. S., & Noll, Ρ. (1984). *Digital Coding of Waveforms: Principles and Applications to Speech and Video*. Prentice Hall.

Mano, Μ. Μ., & Ciletti, Μ. D. (2012). *Digital Design (5 εκδ.)*. Pearson Education.

MathWorks - DSP. (2017). Ανάκτηση 2 10, 2017, από Getting Started with Matlab DSP: <https://www.mathworks.com/help/dsp/getting-started-with-dsp-system-toolbox.html>

Proakis, J. G., & Salehi, Μ. (2008). *Digital Communications (5 εκδ.)*. McGraw-Hill Higher Education.

Μουρτζόπουλος, Ι. (2012). *Ψηφιακή Τεχνολογία Ήχου*. Πάτρα: Πανεπιστήμιο Πατρών.

Εργαστηριακή Άσκηση 5: Γραμμική Συνέλιξη

1 Εισαγωγικά Στοιχεία

1.1 Σκοπός

Σκοπός της παρούσας εργαστηριακής άσκησης είναι η εξοικείωση των φοιτητών – σπουδαστών με τον υπολογισμό της γραμμικής συνέλιξης ΣΔΧ και των ιδιοτήτων της, για τον υπολογισμό της απόκρισης ενός συστήματος.

1.2 Αναμενόμενα Μαθησιακά Αποτελέσματα

Με την ολοκλήρωση της παρούσας εργαστηριακής άσκησης, ο φοιτητής – σπουδαστής θα είναι σε θέση να:

- Εκτελεί την πράξη της γραμμικής συνέλιξη μεταξύ σημάτων διακριτού χρόνου
- Γνωρίζει και να αποδεικνύει τις εξής ιδιότητες της συνέλιξης:
 - αντιμεταθετική
 - προσεταιριστική
 - επιμεριστική
- Χρησιμοποιεί τη συνέλιξη για τον υπολογισμό της απόκρισης ενός συστήματος διακριτού χρόνου
- Χρησιμοποιεί τη συνέλιξη για τον υπολογισμό της κρουστικής απόκρισης ενός συστήματος διακριτού χρόνου
- Αναγνωρίζει τις βασικές διασυνδέσεις συστημάτων διακριτού χρόνου, όπως:
 - διασύνδεση σε σειρά
 - παράλληλη διασύνδεση
- Υπολογίζει τη συνολική κρουστική απόκριση των βασικών και σύνθετων διασυνδέσεων συστημάτων διακριτού χρόνου
- Υπολογίζει την απόκριση σύνθετων διασυνδέσεων συστημάτων διακριτού χρόνου

- Κατανοεί τις εφαρμογές της συνέλιξης και των ιδιοτήτων της για τον υπολογισμό της εξόδου ενός συστήματος

1.3 Εκτιμώμενη Διάρκεια

Ο εκτιμώμενος χρόνος που απαιτείται από τον φοιτητή – σπουδαστή για την ολοκλήρωση της παρούσας εργαστηριακής άσκησης είναι 2 – 4 διδακτικές ώρες.

1.4 Προαπαιτούμενα

- Διάλεξη 3 της θεωρίας.
- Επίσης, απαιτείται εξοικείωση με:
 - τους τρόπους αναπαράστασης των σημάτων,
 - τις πράξεις μεταξύ σημάτων και
 - τους μετασχηματισμούς της ανεξάρτητης μεταβλητής n (μετατόπιση, κλιμάκωση, αντιστροφή).

1.5 Εισαγωγή

Στην άσκηση αυτή θα ασχοληθούμε με την έννοια της κρουστικής απόκρισης συστήματος, με τον υπολογισμό της γραμμικής συνέλιξης καθώς και με την αντιμεταθετική, προσεταιριστική και επιμεριστική της ιδιότητα για σήματα διακριτού χρόνου. Επίσης, θα ασχοληθούμε με τους βασικούς τύπους διασύνδεσης συστημάτων διακριτού χρόνου και θα αξιοποιήσουμε την πράξη της συνέλιξης και τις ιδιότητές της, προκειμένου να υπολογίσουμε την απόκριση (έξοδο) σύνθετων συστημάτων. Επισημαίνεται ότι η γραμμική συνέλιξη αποτελεί τη θεμελιώδη πράξη για τον υπολογισμό της απόκρισης συστήματος (δηλαδή της εξόδου) στα συστήματα διακριτού χρόνου.

1.6 Κατάλογος Χρήσιμων Συναρτήσεων Matlab

Κατά την εκπόνηση των ασκήσεων χρησιμοποιούνται οι ακόλουθες συναρτήσεις – εντολές του Matlab:

| Συνάρτηση | Επεξήγηση |
|-----------|--|
| length | Επιστρέφει το μήκος ενός πίνακα-διανύσματος |
| zeros | Δημιουργία πίνακα – διανύσματος με μηδενικά |
| fliplr | Αναποδογυρίζει μία ακολουθία, ώστε το πρώτο στοιχείο να γίνει τελευταίο και το τελευταίο πρώτο |
| conv | Υπολογίζει τη συνέλιξη μεταξύ δύο ακολουθιών – σημάτων |
| sin | Συνάρτηση ημιτόνου |
| abs | Απόλυτη τιμή |
| figure | Δημιουργεί νέο παράθυρο απεικόνισης γραφικών παραστάσεων |
| plot | Δημιουργεί μία γραφική παράσταση |
| subplot | Τμηματοποίηση ενός παραθύρου για γραφικές παραστάσεις σε γραμμές και στήλες, προκειμένου στο ίδιο παράθυρο να απεικονιστούν πολλαπλές γραφικές παραστάσεις |
| stem | Γραφική παράσταση διακριτών σημάτων – ακολουθιών |
| title | Τοποθετεί τίτλο σε μια γραφική παράσταση |
| axis | Ορίζει τα όρια των αξόνων και άλλες ιδιότητες των αξόνων σε μια γραφική παράσταση |
| grid | Προσθέτει – αφαιρεί πλέγμα στις γραφικές παραστάσεις, βελτιώνοντας την ανάγνωσή τους |

2 Γραμμική Συνέλιξη

2.1 Υπολογισμός Απόκρισης Συστήματος

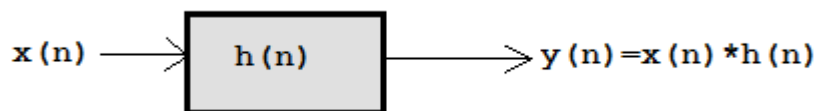
Ένα γραμμικό και χρονικά αμετάβλητο (ΓΧΑ) σύστημα διακριτού χρόνου μπορεί να περιγραφεί από την κρουστική του απόκριση $h(n)$. Η γνώση της $h(n)$ ενός

συστήματος είναι σημαντική, δεδομένου ότι εφόσον την γνωρίζουμε μπορούμε πάντα να υπολογίσουμε την απόκριση (έξοδο) του συστήματος, για οποιαδήποτε είσοδο τροφοδοτήσουμε το σύστημα.

Ως **κρουστική απόκριση** $h(n)$ ενός συστήματος ορίζεται η απόκριση του συστήματος (έξοδος), όταν του δίνουμε ως είσοδο την μοναδιαία κρουστική ακολουθία $\delta(n)$.

Η **απόκριση** $y(n)$ ενός **ΓΧΑ συστήματος (έξοδος)** σε είσοδο $x(n)$ υπολογίζεται με την πράξη της γραμμικής συνέλιξης μεταξύ του σήματος εισόδου $x(n)$ και της κρουστικής απόκρισης $h(n)$ του συστήματος, ενώ μαθηματικά ορίζεται ως εξής:

$$y(n) = x(n) * h(n) = \sum_{k=-\infty}^{k=+\infty} x(k)h(n - k)$$



Σχήμα 1: Αναπαράσταση συστήματος

Με την ίδια πράξη μπορούμε να υπολογίσουμε και την κρουστική απόκριση $h(n)$ ενός συστήματος, αρκεί να δώσουμε για είσοδο την $x(n)=\delta(n)$, οπότε προκύπτει ότι:

$$y(n) = \delta(n) * h(n) = h(n)$$

Ο υπολογισμός της συνέλιξης 2 σημάτων στο Matlab γίνεται με τη συνάρτηση `conv`, η οποία συντάσσεται ως εξής:

$$y=conv(x,h)$$

όπου:

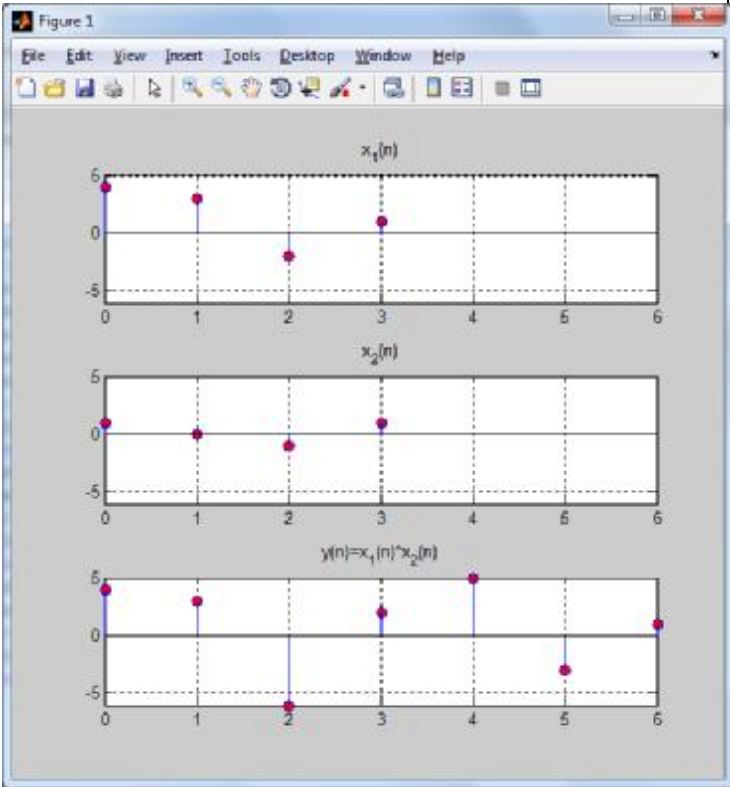
- y =αποτέλεσμα της συνέλιξης (ή έξοδος ενός συστήματος)
- x : σήμα εισόδου
- h : κρουστική απόκριση συστήματος

Προσοχή πρέπει να δοθεί στα εξής σημεία κατά τον υπολογισμό της συνέλιξης:

- Τα σήματα x και h θα πρέπει να είναι ορισμένα στο ίδιο χρονικό διάστημα n . Αν τα δύο σήματα αναφέρονται σε διαφορετικά διαστήματα, τότε θα πρέπει να συμπληρώσουμε τα διαστήματα με μηδενικά, ώστε να οριστούν στο ίδιο διάστημα.
- Αν τα διαστήματα των 2 σημάτων είναι τα $[L1, R1]$ και $[L2, R2]$ αντίστοιχα, τότε τα πλάτη της συνέλιξης θα βρίσκονται στο διάστημα $[L1+L2, R1+R2]$.
- Το διάστημα στο οποίο βρίσκονται τα πλάτη της συνέλιξης είναι μήκους: $\text{length}(x)+\text{length}(h)-1$. Συνεπώς, η απεικόνιση των πλατών της συνέλιξης πρέπει να γίνεται στο διπλάσιο διάστημα από αυτό που έχουμε ορίσει.

Άσκηση 1^η

Έστω το σήμα $x_1(n)=\{4, 3, -2, 1\}$ για $n=[0,3]$ και το σήμα $x_2(n)=\{1, 0, -1, 1\}$ για $n=[0,3]$. Να υπολογιστεί η συνέλιξή τους.

| Κώδικας Matlab | Αποτέλεσμα |
|---|--|
| <pre> n1=[0:3]; x1=[4 3 -2 1]; n2=[0:3]; x2=[1 0 -1 1]; y=conv(x1,x2); nol=[0+0:3+3]; figure(1) subplot(3,1,1) stem(n1,x1,'MarkerFaceColor','red') title('x_1(n)') axis([nol(1) nol(end) min(y) max(y)]) grid on </pre> |  |

```

subplot(3,1,2)

stem(n2,x2,'MarkerFaceColor','red')

title('x_2(n)')

axis([nol(1) nol(end) min(y) max(y)])

grid on

subplot(3,1,3)

stem(nol,y,'MarkerFaceColor','red')

title('y(n)=x_1(n)*x_2(n)')

axis([nol(1) nol(end) min(y) max(y)])

grid on

```

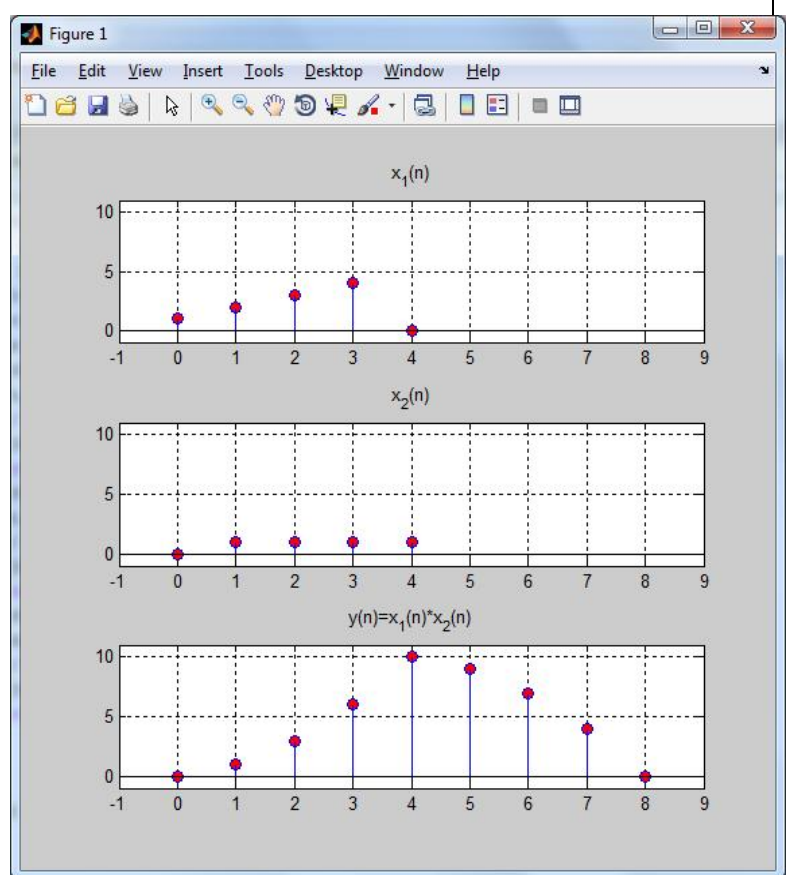
Άσκηση 2^η

Έστω τα σήματα $x_1(n) = \{1, 2, 3, 4\}$ για $n = [0, 3]$ και $x_2(n) = \{1, 1, 1, 1\}$ για $n = [1, 4]$. Να υπολογιστεί η συνέλιξή τους.

| Κώδικας Matlab | Αποτέλεσμα |
|----------------|------------|
|----------------|------------|

% Επειδή τα 2 διαστήματα είναι διαφορετικά επιλέγουμε $n=[0,4]$ ώστε να καλύπτουμε και τα 2 διαστήματα και συμπληρώνουμε κατάλληλα με μηδενικά τα 2 σήματα

```
n=[0:4];
x1=[1 2 3 4 0];
x2=[0 1 1 1 1];
nol=[0:8];
y=conv(x1,x2);
figure(1)
subplot(3,1,1)
stem(n,x1,'MarkerFaceColor','red')
title('x_1(n)')
axis([nol(1)-1 nol(end)+1 min(y)-1 max(y)+1]), grid on
subplot(3,1,2)
stem(n,x2,'MarkerFaceColor','red')
title('x_2(n)')
axis([nol(1)-1 nol(end)+1 min(y)-1 max(y)+1]), grid on
subplot(3,1,3)
stem(nol,y,'MarkerFaceColor','red')
title('y(n)=x_1(n)*x_2(n)')
axis([nol(1)-1 nol(end)+1 min(y)-1 max(y)+1]), grid on
```



Άσκηση 3^η

Δίνεται σύστημα με κρουστική απόκριση $h(n) = \{5, -4, 3, -2, 1, -1\}$ για $n = [-2, 3]$. Να βρεθεί η απόκριση του συστήματος αν το τροφοδοτήσουμε με είσοδο:

- $x_1(n) = \delta(n)$
- $x_2(n) = u(n) - u(n-5)$
- $x_3(n) = 2\delta(n+1) - 3\delta(n) + \delta(n-2)$
- $x_4(n) = x_2(-n-1) + 3x_3(-n+1) - n$, όπου $n = [-10, 10]$

Κώδικας Matlab

% Ερώτημα α)

% Η κρουστική απόκριση είναι ορισμένη στο $n = [-2, 3]$, ενώ το $x_1(n) = \delta(n)$ έχει μόνο μία μονάδα στη θέση $n = 0$. Συνεπώς, θα ορίσω το $x_1(n)$ στο διάστημα $n = [-2, 3]$, συμπληρώνοντας τα πλάτη με 0

```
n = [-2:3];
```

```
h = [5 -4 3 -2 1 -1];
```

```
x1 = [0 0 1 0 0 0];
```

```
nol = [-2+(-2):3+3];
```

```
y1 = conv(x1, h);
```

```
figure(1)
```

```
subplot(3,1,1)
```

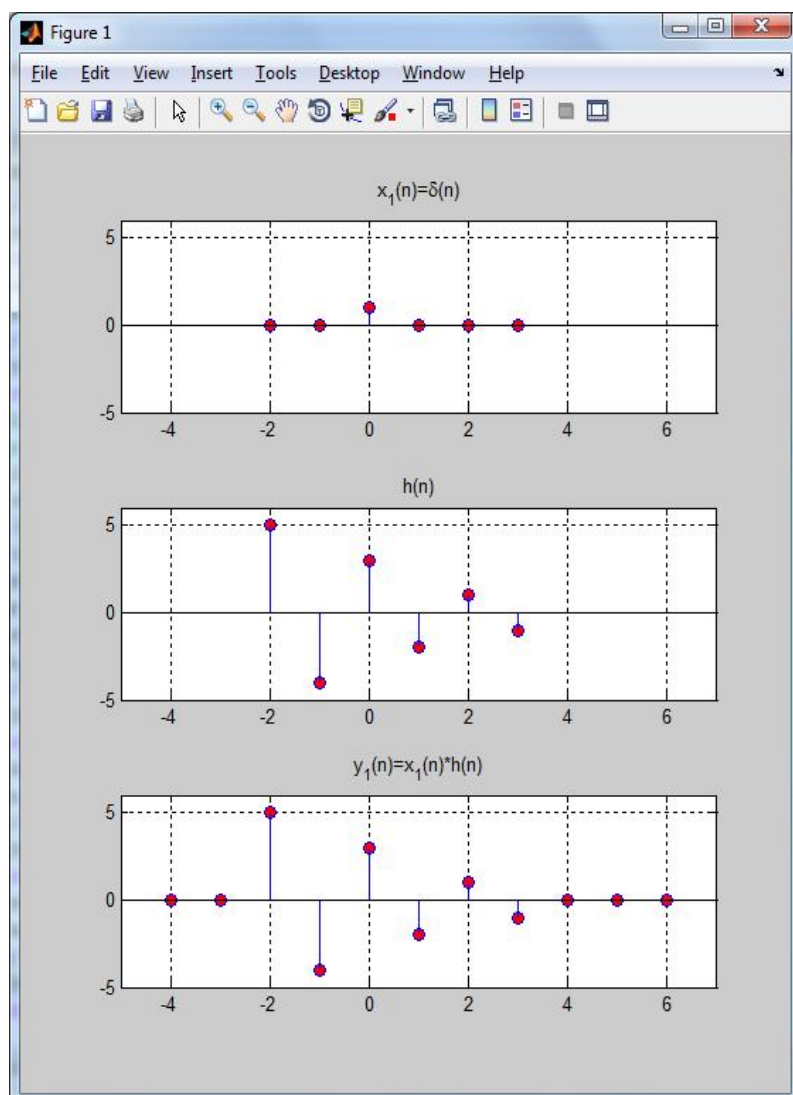
```
stem(n, x1, 'MarkerFaceColor', 'red')
```

```
title('x_1(n) = \delta(n)')
```

```
axis([nol(1)-1 nol(end)+1 min(y1)-1 max(y1)+1])
```

```
grid on
```

Αποτέλεσμα



```

subplot(3,1,2)

stem(n,h,'MarkerFaceColor','red')

title('h(n)')

axis([nol(1)-1 nol(end)+1 min(y1)-1
max(y1)+1]), grid on

subplot(3,1,3)

stem(nol,y1,'MarkerFaceColor','red')

title('y_1(n)=x_1(n)*h(n)')

axis([nol(1)-1 nol(end)+1 min(y1)-1
max(y1)+1]), grid on

```

Παρατηρήσεις:

Όταν δώσουμε ως είσοδο στο σύστημα την $\delta(n)$, τότε στην έξοδο εμφανίζεται η κρουστική απόκριση $h(n)$ του συστήματος, επαληθεύοντας έτσι τον ορισμό της κρουστικής απόκρισης.

Η συνέλιξη, λοιπόν, ενός σήματος με τη $\delta(n)$ δίνει ως αποτέλεσμα το ίδιο το σήμα.

% Ερώτημα b)

% Η κρουστική απόκριση είναι ορισμένη στο $n=[-2,3]$, ενώ το $x_2(n)=u(n)-u(n-5)=1$, για $n=[0,4]$. Συνεπώς, θα ορίσω το $h(n)$ και το $x_2(n)$ στο διάστημα $n=[-2,4]$, συμπληρώνοντας τα πλάτη τους με 0

```

n=[-2:4];

h=[5 -4 3 -2 1 -1 0];

x2=[0 0 1 1 1 1 1];

nol=[-2+(-2):4+4];

y2=conv(x2,h);

figure(2)

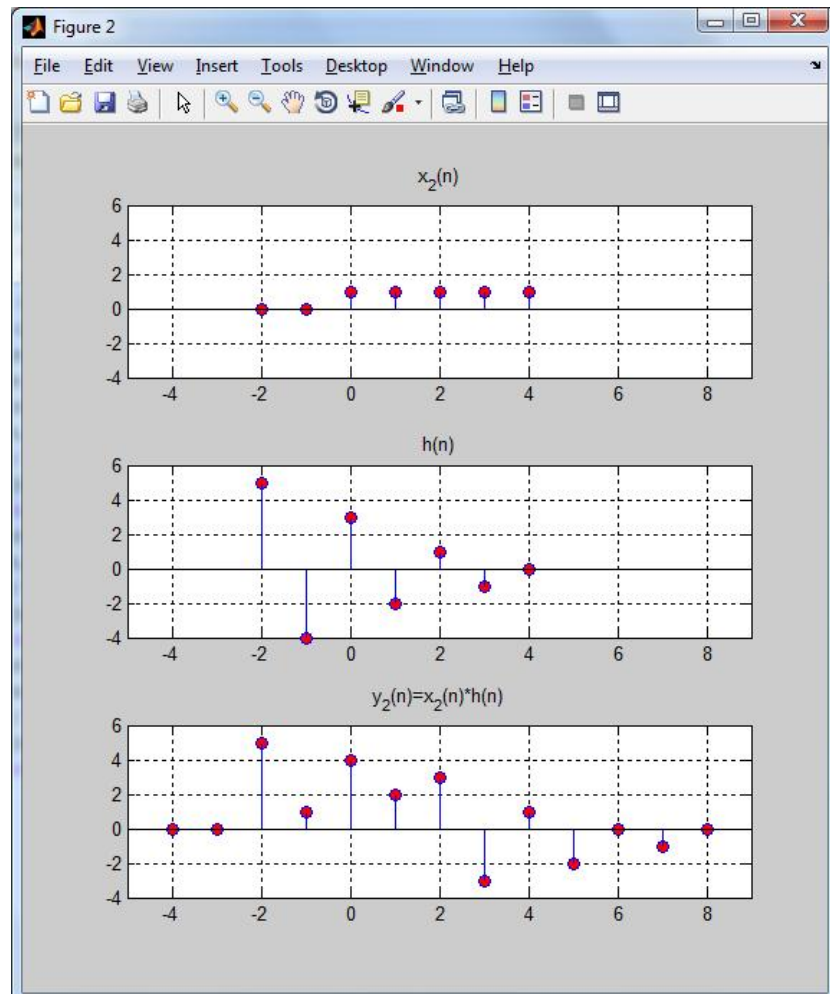
subplot(3,1,1)

stem(n,x2,'MarkerFaceColor','red')

title('x_2(n)')

axis([nol(1)-1 nol(end)+1 min(y2)-1

```



```
max(y2)+1]), grid on

subplot(3,1,2)

stem(n,h,'MarkerFaceColor','red')

title('h(n)')

axis([nol(1)-1 nol(end)+1 min(y2)-1
max(y2)+1]), grid on

subplot(3,1,3)

stem(nol,y2,'MarkerFaceColor','red')

title('y_2(n)=x_2(n)*h(n)')

axis([nol(1)-1 nol(end)+1 min(y2)-1
max(y2)+1]), grid on
```

% Ερώτημα c)

% Το x_3 μπορεί να γραφτεί και ως
εξής: $x_3(n) = \{2, -3, 0, 1\}$ με $n = [-1, 2]$. Η
κρουστική απόκριση είναι ορισμένη στο
 $n = [-2, 3]$, ενώ το $x_3(n)$ στο $n = [-1, 2]$.
Συνεπώς, θα ορίσω το $x_2(n)$ στο
διάστημα $n = [-2, 3]$, συμπληρώνοντας τα
πλάτη του με 0

```
n=[-2:3], h=[5 -4 3 -2 1 -1];
```

```
x3=[0 2 -3 0 1 0], nol=[-2+(-2):3+3];
```

```
y3=conv(x3,h);
```

```
figure(3)
```

```
subplot(3,1,1)
```

```
stem(n,x3,'MarkerFaceColor','red')
```

```
title('x_3(n)')
```

```
axis([nol(1)-1 nol(end)+1 min(y3)-1  
max(y3)+1]), grid on
```

```
subplot(3,1,2)
```

```
stem(n,h,'MarkerFaceColor','red')
```

```
title('h(n)')
```

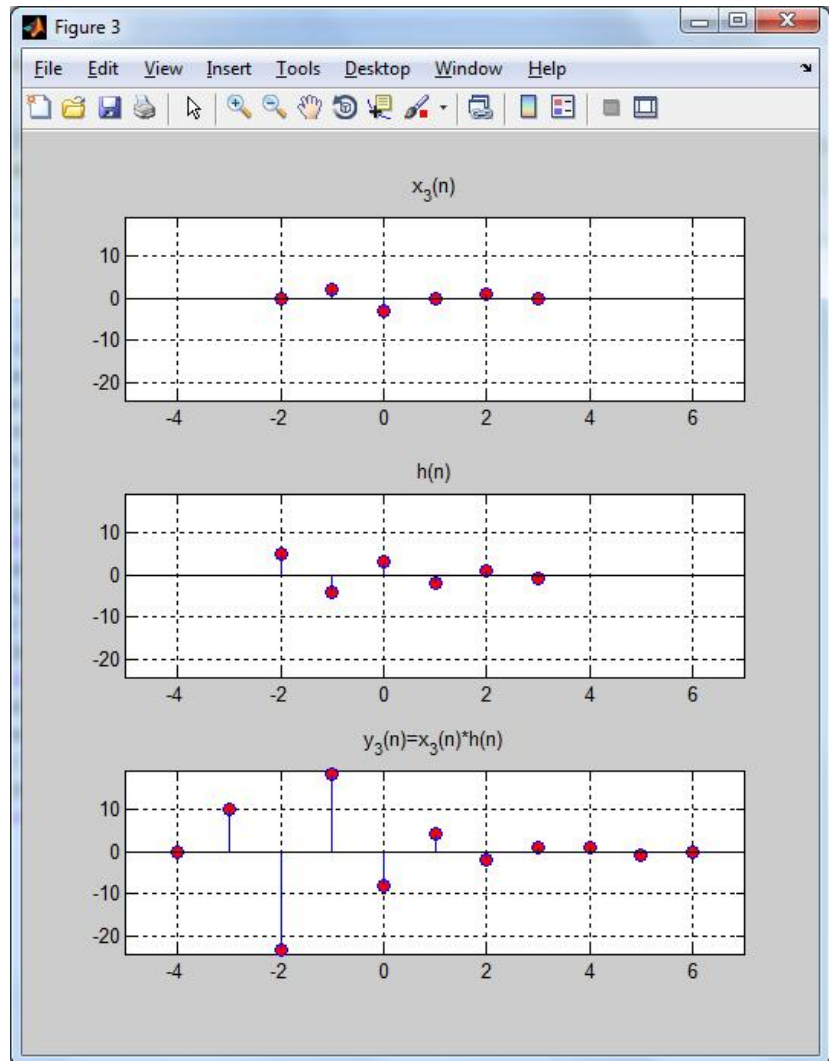
```
axis([nol(1)-1 nol(end)+1 min(y3)-1  
max(y3)+1]), grid on
```

```
subplot(3,1,3)
```

```
stem(nol,y3,'MarkerFaceColor','red')
```

```
title('y_3(n)=x_3(n)*h(n)')
```

```
axis([nol(1)-1 nol(end)+1 min(y3)-1  
max(y3)+1]), grid on
```



% Ερώτημα d)

% Όλα τα σήματα για ευκολία θα οριστούν στο διάστημα $n=[-10,10]$

```
n=[-10:10];
```

```
len=length(n);
```

```
h=[zeros(1,8) 5 -4 3 -2 1 -1  
zeros(1,7)];
```

```
x2=[zeros(1,10) ones(1,5)  
zeros(1,6)];
```

```
% x2(-n-1)
```

```
x2a=[zeros(1,1) x2(1:end-1)];
```

```
x2a=fliplr(x2a);
```

```
x3=zeros(1,len);
```

```
x3(n==-1)=2;
```

```
x3(n==0)=-3;
```

```
x3(n==2)=1;
```

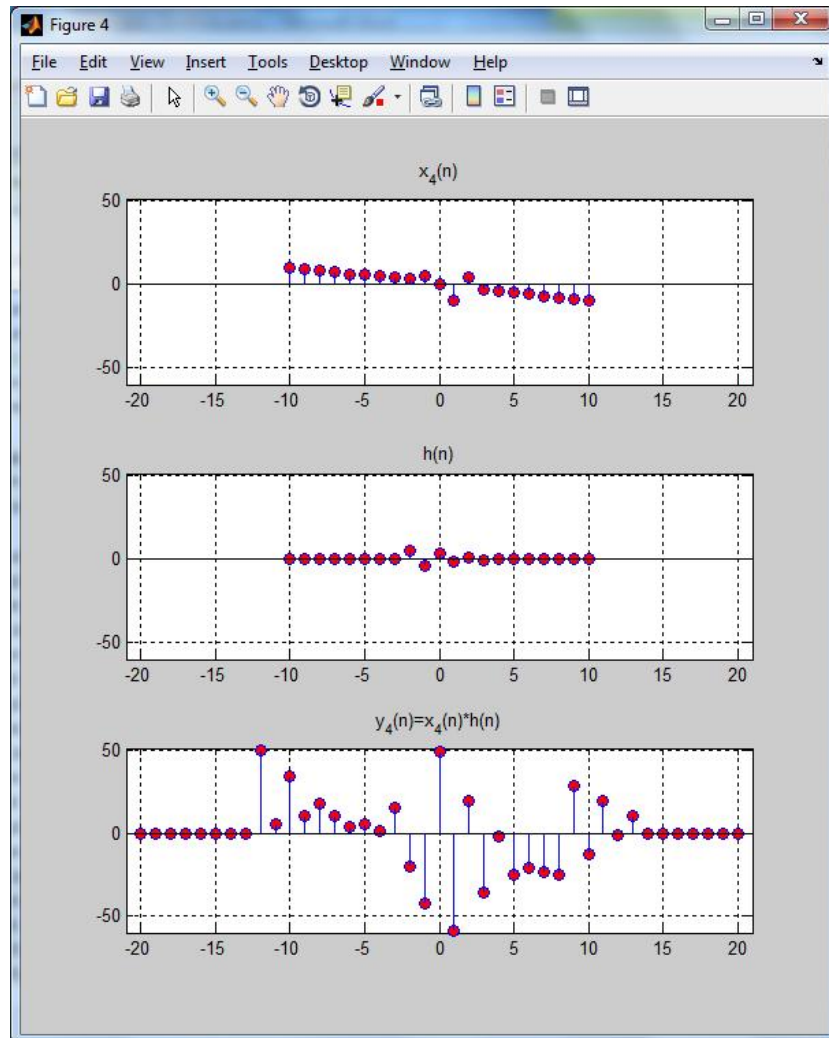
```
% x3(-n+1)
```

```
x3a=[x3(1+1:end) zeros(1,1)];
```

```
x3a=fliplr(x3a);
```

```
x4=x2a+3*x3a-n;
```

```
nol=-10+(-10):10+10;
```



```

y4=conv(x4,h);

figure(4)

subplot(3,1,1)

stem(n,x4,'MarkerFaceColor','red')

title('x_4(n)')

axis([nol(1)-1 nol(end)+1 min(y4)-1
max(y4)+1])

grid on

subplot(3,1,2)

stem(n,h,'MarkerFaceColor','red')

title('h(n)')

axis([nol(1)-1 nol(end)+1 min(y4)-1
max(y4)+1])

grid on

subplot(3,1,3)

stem(nol,y4,'MarkerFaceColor','red')

title('y_4(n)=x_4(n)*h(n)')

axis([nol(1)-1 nol(end)+1 min(y4)-1
max(y4)+1])

grid on

```

Άσκηση 4η

Δίνεται σύστημα με κρουστική απόκριση $h(n)=3\delta(n+2)-\delta(n-1)-2\delta(n-2)$. Να βρεθεί η έξοδος του συστήματος αν το τροφοδοτήσουμε με είσοδο:

a) $x_1(n) = n[u(n+2) - u(n-3)]$

b) $x_2(n) = |n|[u(-n+2) - u(-n-3)]$

c) $x_3(n) = \sin(n\pi/2)[u(n+2) - u(n-3)]$

Θεωρείστε ότι όλα τα σήματα είναι ορισμένα στο διάστημα $n = [-10, 10]$.

Κώδικας Matlab

`% Ερώτημα α)`

`n = [-10:10];`

`len = length(n);`

`h = zeros(1, len);`

`h(n == -2) = 3;`

`h(n == 1) = -1;`

`h(n == 2) = -2;`

`u = [zeros(1, 10) ones(1, 11)];`

`% u(n+2)`

`u2 = [u(1+2:end) ones(1, 2)];`

`% u(n-3)`

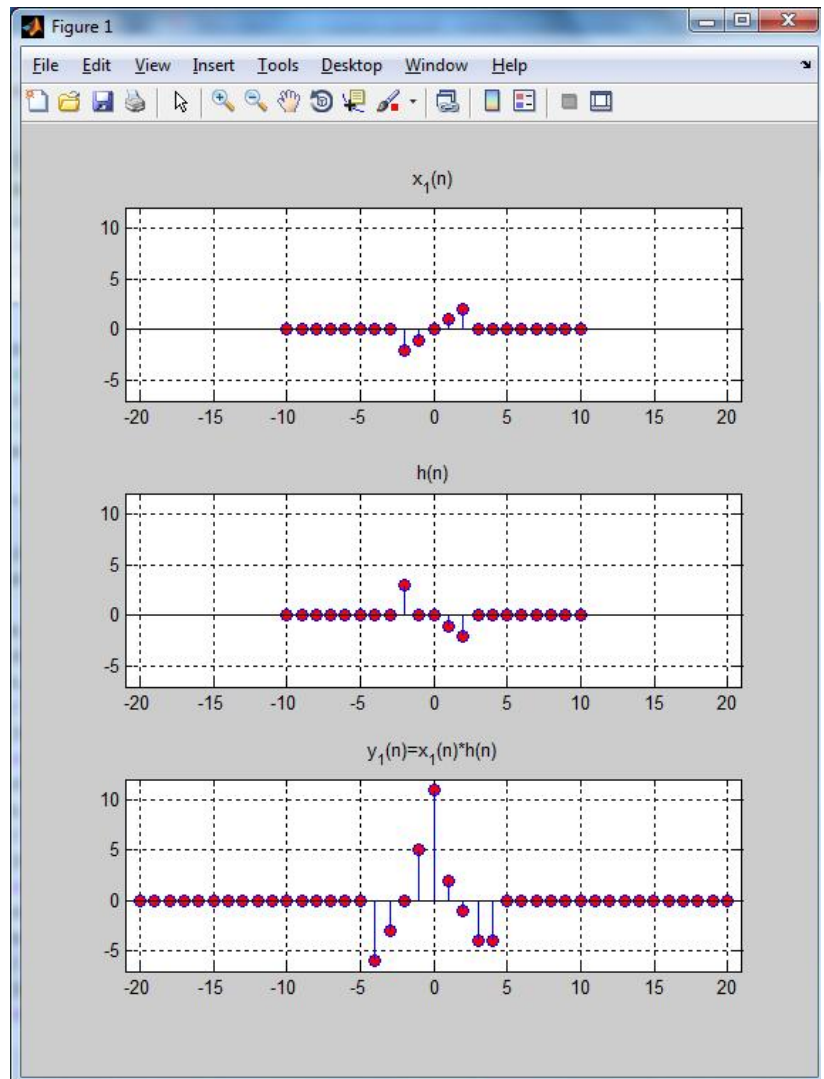
`u3 = [zeros(1, 3) u(1:end-3)];`

`x1 = n .* (u2 - u3);`

`no1 = -10 + (-10):10+10;`

`y1 = conv(x1, h);`

Αποτέλεσμα



```
figure(1)

subplot(3,1,1)

stem(n,x1,'MarkerFaceColor','red')

title('x_1(n)')

axis([nol(1)-1 nol(end)+1 min(y1)-1
max(y1)+1])

grid on

subplot(3,1,2)

stem(n,h,'MarkerFaceColor','red')

title('h(n)')

axis([nol(1)-1 nol(end)+1 min(y1)-1
max(y1)+1])

grid on

subplot(3,1,3)

stem(nol,y1,'MarkerFaceColor','red')

title('y_1(n)=x_1(n)*h(n)')

axis([nol(1)-1 nol(end)+1 min(y1)-1
max(y1)+1])

grid on
```

```
% Ερώτημα b)
```

```
n=[-10:10];
```

```
len=length(n);
```

```
h=zeros(1,len);
```

```
h(n==-2)=3;
```

```
h(n==1)=-1;
```

```
h(n==2)=-2;
```

```
u=[zeros(1,10) ones(1,11)];
```

```
% u(-n+2)
```

```
u2=[u(1+2:end) ones(1,2)];
```

```
u2=fliplr(u2);
```

```
% u(-n-3)
```

```
u3=[zeros(1,3) u(1:end-3)];
```

```
u3=fliplr(u3);
```

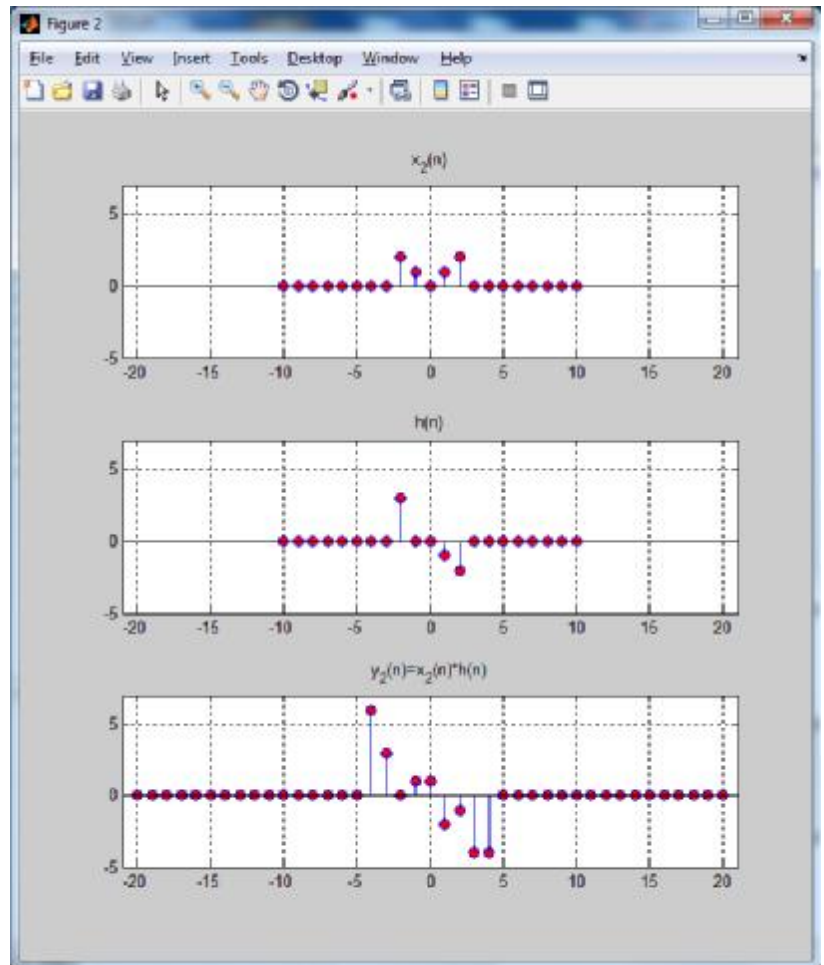
```
x2=abs(n).*(u2-u3);
```

```
nol=-10+(-10):10+10;
```

```
y2=conv(x2,h);
```

```
figure(2)
```

```
subplot(3,1,1)
```



```
stem(n,x2,'MarkerFaceColor','red')

title('x_2(n)')

axis([nol(1)-1 nol(end)+1 min(y2)-1
max(y2)+1])

grid on

subplot(3,1,2)

stem(n,h,'MarkerFaceColor','red')

title('h(n)')

axis([nol(1)-1 nol(end)+1 min(y2)-1
max(y2)+1])

grid on

subplot(3,1,3)

stem(nol,y2,'MarkerFaceColor','red')

title('y_2(n)=x_2(n)*h(n)')

axis([nol(1)-1 nol(end)+1 min(y2)-1
max(y2)+1])

grid on
```

```
% Ερώτημα c)
```

```
n=[-10:10];
```

```
len=length(n);
```

```
h=zeros(1,len);
```

```
h(n==-2)=3;
```

```
h(n==1)=-1;
```

```
h(n==2)=-2;
```

```
u=[zeros(1,10) ones(1,11)];
```

```
% u(n+2)
```

```
u2=[u(1+2:end) ones(1,2)];
```

```
% u(n-3)
```

```
u3=[zeros(1,3) u(1:end-3)];
```

```
x3=sin(n*pi/2).*(u2-u3);
```

```
nol=-10+(-10):10+10;
```

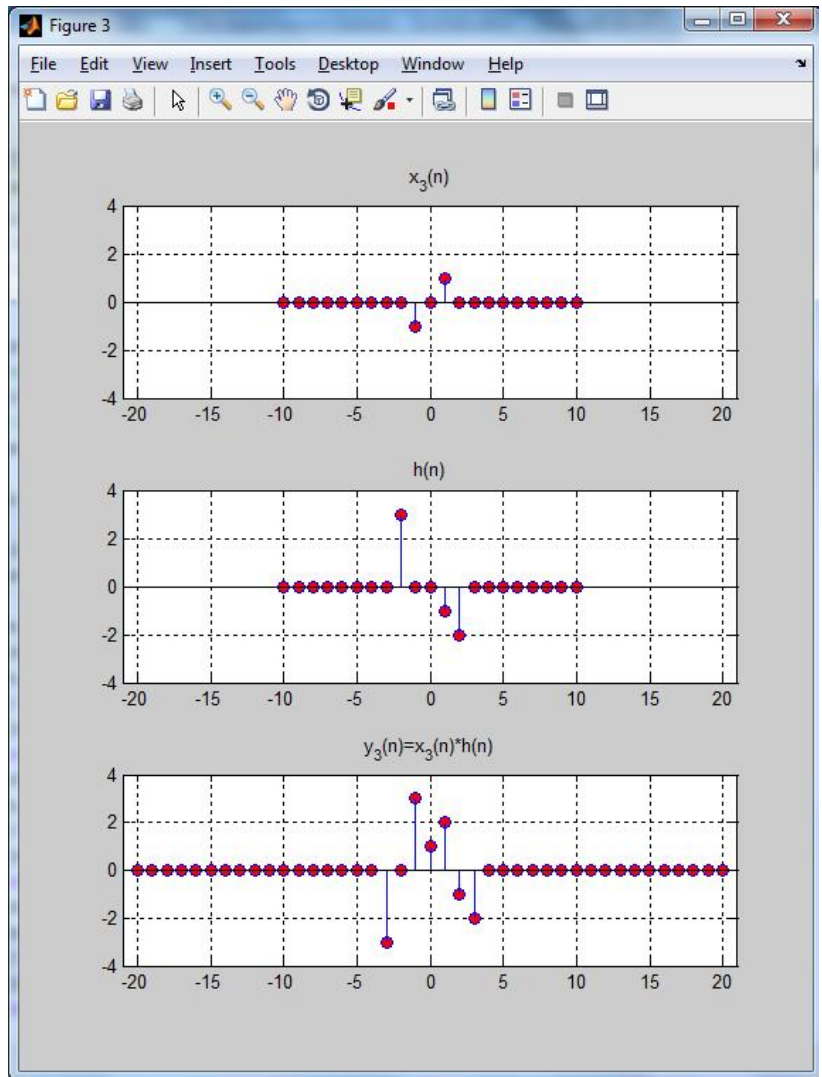
```
y3=conv(x3,h);
```

```
figure(3)
```

```
subplot(3,1,1)
```

```
stem(n,x3,'MarkerFaceColor','red')
```

```
title('x_3(n)')
```



```

axis([nol(1)-1 nol(end)+1 min(y3)-1
max(y3)+1])

grid on

subplot(3,1,2)

stem(n,h,'MarkerFaceColor','red')

title('h(n)')

axis([nol(1)-1 nol(end)+1 min(y3)-1
max(y3)+1])

grid on

subplot(3,1,3)

stem(nol,y3,'MarkerFaceColor','red')

title('y_3(n)=x_3(n)*h(n)')

axis([nol(1)-1 nol(end)+1 min(y3)-1
max(y3)+1])

grid on

```

2.2 Ιδιότητες Συνέλιξης και Εφαρμογές

Η συνέλιξη έχει μερικές ιδιότητες οι οποίες είναι χρήσιμες στον υπολογισμό της απόκρισης σύνθετων συστημάτων. Πιο συγκεκριμένα, ισχύουν οι εξής ιδιότητες:

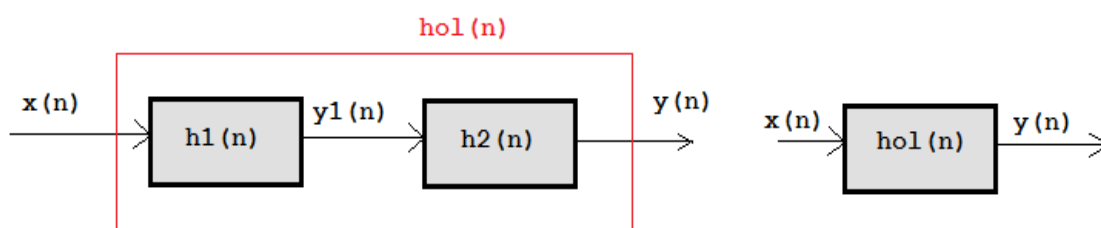
1. **Αντιμεταθετική:** $x_1(n) * x_2(n) = x_2(n) * x_1(n)$, που σημαίνει ότι δεν παίζει ρόλο ποιο σήμα θα θεωρήσουμε ως πρώτο και ποιο ως δεύτερο κατά τη συνέλιξή τους.
2. **Προσεταιριστική:** $x_1(n) * [x_2(n) * x_3(n)] = x_1(n) * [x_2(n) * x_3(n)]$. Η ιδιότητα αυτή βρίσκει εφαρμογή σε διασυνδέσεις συστημάτων που βρίσκονται σε **διάταξη σειράς**.

3. **Επιμεριστική:** $x_1(n) * [x_2(n) + x_3(n)] = x_1(n) * x_2(n) + x_1(n) * x_3(n)$.

Η ιδιότητα αυτή βρίσκει εφαρμογή σε διασυνδέσεις συστημάτων που βρίσκονται σε **παράλληλη διάταξη**.

2.2.1 Διασύνδεση Συστημάτων σε Σειρά

Ας θεωρήσουμε ότι το σήμα $x(n)$ δίνεται ως είσοδος στο 1^ο σύστημα με κρουστική απόκριση $h_1(n)$, παράγοντας έξοδο $y_1(n)$. Στη συνέχεια, η έξοδος $y_1(n)$ τροφοδοτεί το 2^ο σύστημα με κρουστική απόκριση $h_2(n)$, το οποίο παράγει έξοδο $y(n)$.



Σχήμα 2: (Αριστερά) Διασύνδεση δύο συστημάτων σε σειρά και (δεξιά) η ισοδύναμη συνολική διάταξη

Υπολογίζοντας τις εξόδους των 2 επιμέρους συστημάτων έχουμε:

- Για την έξοδο $y_1(n) = x(n) * h_1(n)$
- Για την έξοδο $y(n) = y_1(n) * h_2(n) = x(n) * h_1(n) * h_2(n)$

Αναφορικά με τη συνολική διάταξη, μπορούμε να θεωρήσουμε, λοιπόν, ότι:

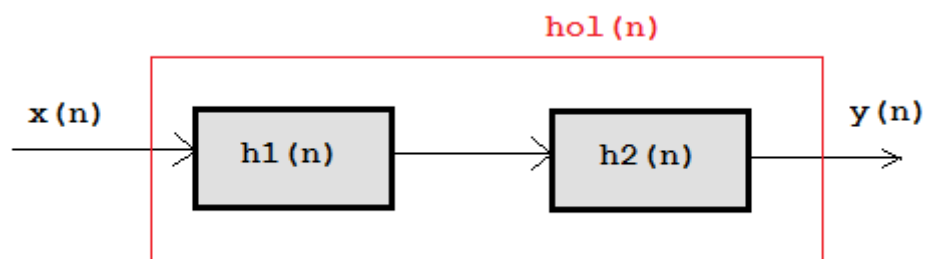
- έχει κρουστική απόκριση $hol(n) = h_1(n) * h_2(n)$ και
- τροφοδοτούμενη με είσοδο $x(n)$ παράγει έξοδο $y(n) = x(n) * hol(n)$

Άσκηση 5^η

Δίνονται τα σήματα $h_1(n) = \{1, 2, 3, 0, 3, 2, 1\}$ με $n = [-2, 4]$ και $h_2(n) = \{-1, 1, -1\}$ με $n = [-1, 1]$.

- a) Να υπολογιστούν οι συνελίξεις $h_1(n) * h_2(n)$ και $h_2(n) * h_1(n)$. Τι παρατηρείτε;

- b) Αν τα σήματα $h_1(n)$ και $h_2(n)$ είναι οι κρουστικές αποκρίσεις των υποσυστημάτων της διάταξης που φαίνεται στο ακόλουθο σχήμα, τότε να υπολογίσετε τη συνολική κρουστική απόκριση $h_{ol}(n)$ της διάταξης.
- c) Αν στο συνολικό σύστημα δοθεί ως είσοδος το σήμα $x(n)=\{1,-1,0,0,2,-2\}$ με $n=[0,5]$, τότε να υπολογιστεί η απόκριση συστήματος $y(n)$.



| Κώδικας Matlab | Αποτέλεσμα |
|----------------|------------|
|----------------|------------|

```
% Ερώτημα α)
```

```
% Για το h1 είναι n=[-2,4] και για το  
h2 είναι n=[-1,1], συνεπώς θα  
χρησιμοποιήσουμε το διάστημα n=[-2,4]
```

```
n=-2:4;
```

```
h1=[1 2 3 0 3 2 1];
```

```
h2=[0 -1 1 -1 0 0 0];
```

```
nol=-2+(-2):4+4;
```

```
ya=conv(h1,h2);
```

```
yb=conv(h2,h1);
```

```
figure(1)
```

```
subplot(3,1,1)
```

```
stem(n,h1,'MarkerFaceColor','red')
```

```
title('h_1(n)')
```

```
axis([nol(1)-1 nol(end)+1 min(h1)  
max(h1)]), grid on
```

```
subplot(3,1,2)
```

```
stem(n,h2,'MarkerFaceColor','red')
```

```
title('h_2(n)')
```

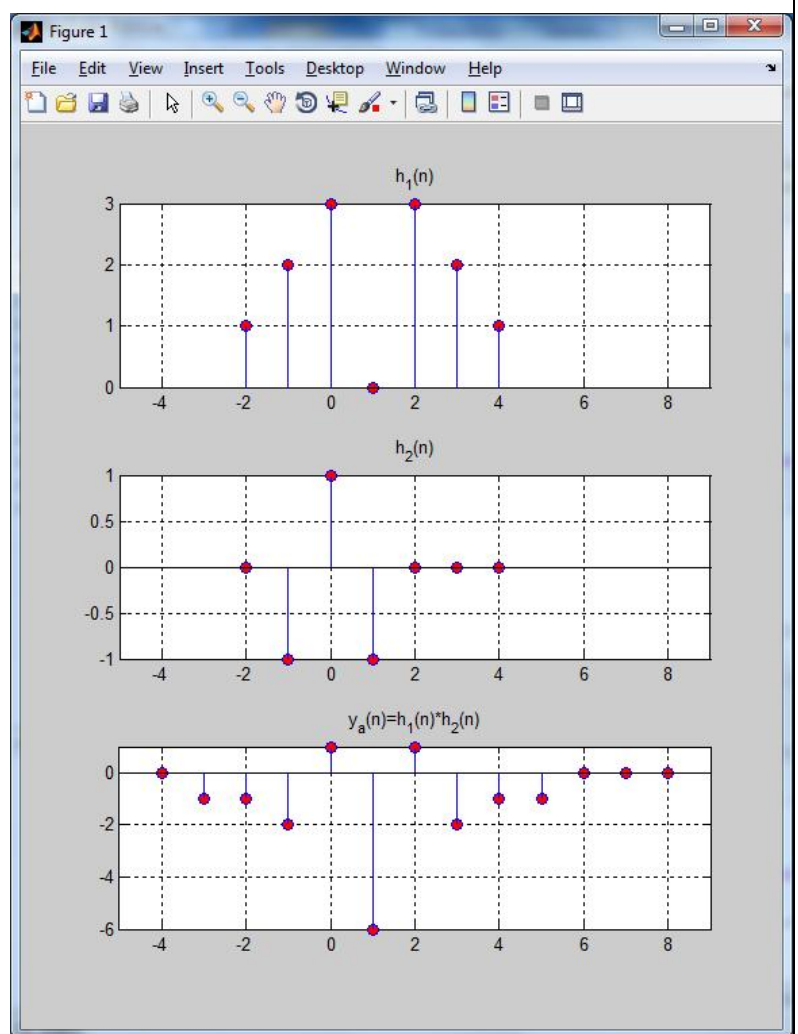
```
axis([nol(1)-1 nol(end)+1 min(h2)  
max(h2)]), grid on
```

```
subplot(3,1,3)
```

```
stem(nol,ya,'MarkerFaceColor','red')
```

```
title('y_a(n)=h_1(n)*h_2(n)')
```

```
axis([nol(1)-1 nol(end)+1 min(ya)  
max(ya)]), grid on
```



```

figure(2)

subplot(3,1,1)

stem(n,h1,'MarkerFaceColor','red')

title('h_1(n)')

axis([nol(1)-1 nol(end)+1 min(h1)
max(h1)])

grid on

subplot(3,1,2)

stem(n,h2,'MarkerFaceColor','red')

title('h_2(n)')

axis([nol(1)-1 nol(end)+1 min(h2)
max(yb)])

grid on

subplot(3,1,3)

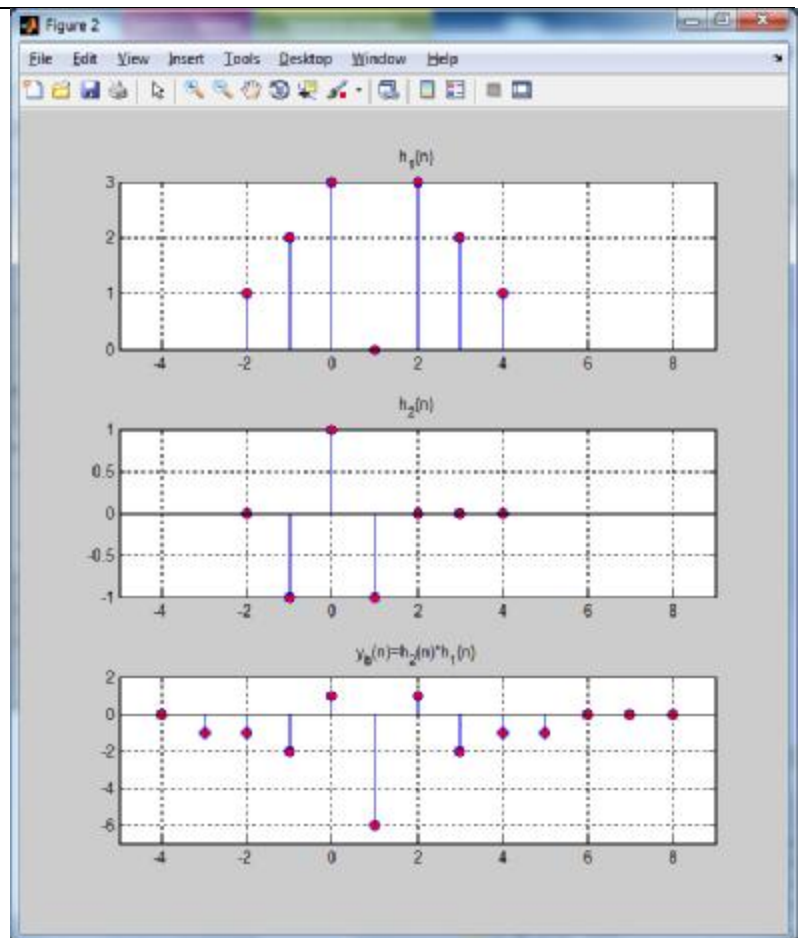
stem(nol,yb,'MarkerFaceColor','red')

title('y_b(n)=h_2(n)*h_1(n)')

axis([nol(1)-1 nol(end)+1 min(yb)-1
max(yb)+1])

grid on

```



Παρατηρήσεις:

Ουσιαστικά επιβεβαιώνεται η αντιμεταθετική ιδιότητα της συνέλιξης.

% Ερώτημα b)

% Για να βρεθεί η συνολική κρουστική απόκριση του συστήματος, αρκεί να δώσουμε ως είσοδο την $\delta(n)$

% Πάλι ως διάστημα έχω το $n=[-2,4]$

$n=-2:4;$

$d=zeros(1,length(n));$

$d(n==0)=1;$

$h1=[1 2 3 0 3 2 1];$

$nol=-2+(-2):4+4;$

$y1=conv(d,h1);$

$h2=zeros(1,length(nol));$

$h2(nol==-1)=-1;$

$h2(nol==0)=1;$

$h2(nol==1)=-1;$

$nol1=nol(1)+nol(1):nol(end)+nol(end);$

$y=conv(y1,h2);$

% Υπολογισμός μέσω προσεταιριστικής ιδιότητας συνέλιξης, δηλαδή $hol(n)=h1(n)*h2(n)$

$n2=-2:4;$

$h1=[1 2 3 0 3 2 1];$

$h2=[0 -1 1 -1 0 0 0];$

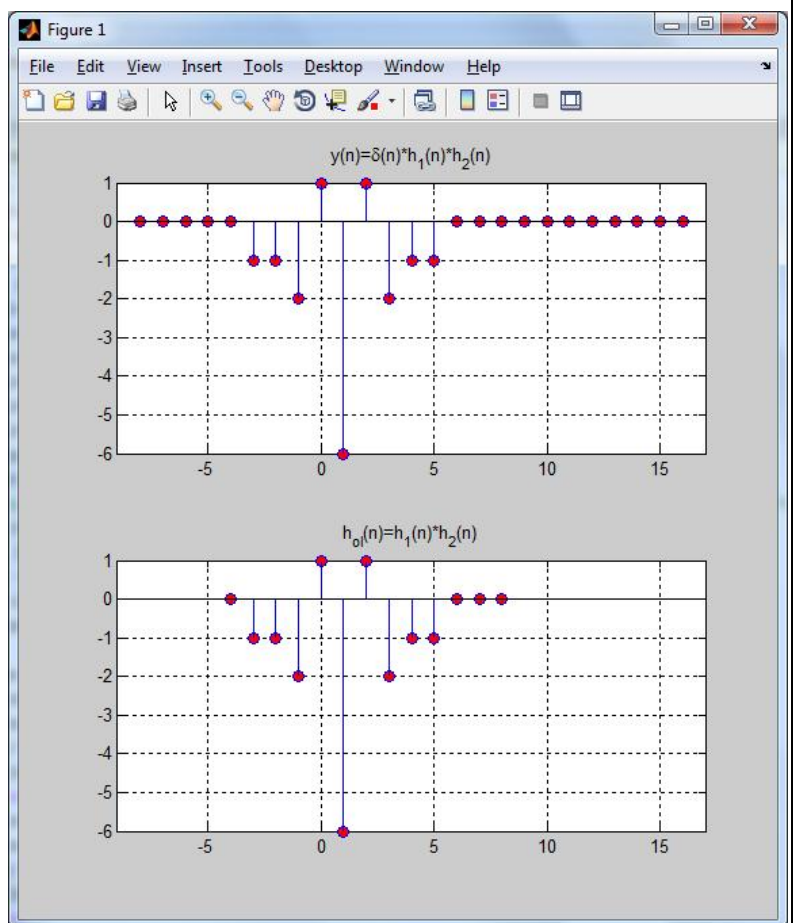
$nol2=-2+(-2):4+4;$

$hol=conv(h1,h2);$

$figure(1)$

$subplot(2,1,1)$

$stem(nol1,y,'MarkerFaceColor','red')$



Παρατηρήσεις:

Ουσιαστικά επιβεβαιώνεται η προσεταιριστική ιδιότητα της συνέλιξης.

Επίσης, υπενθυμίζεται ότι αν σε ένα σύστημα δώσουμε ως είσοδο την $x(n)=\delta(n)$, τότε το σύστημα δίνει στην έξοδό του την κρουστική του απόκριση. Φαίνεται, λοιπόν, ότι

$$y(n) = x(n) * hol(n), \text{ όπου } hol(n) = h1(n) * h2(n) \text{ και}$$

$$y(n) = \delta(n) * hol(n) = hol(n)$$

Συνεπώς, όταν 2 συστήματα με κρουστικές αποκρίσεις $h1(n)$ και $h2(n)$ αντίστοιχα, συνδέονται **σε σειρά**, τότε έχουν συνολική κρουστική απόκριση που υπολογίζεται ως:

$$hol(n) = h1(n) * h2(n)$$

% Ερώτημα c)

% Από το ερώτημα a) γνωρίζουμε ότι η $h_{ol}(n) = \{-1, -1, -2, 1, -6, 1, -2, -1, -1\}$ για $n = [-3, 5]$

% Επειδή $x(n) = \{1, -1, 0, 0, 2, -2\}$ με $n = [0, 5]$, θα ορίσουμε και τα 2 σήματα στο $n = [-3, 5]$ και θα συμπληρώσουμε με μηδενικά το $x(n)$

```
n=-3:5;
```

```
hol=[-1 -1 -2 1 -6 1 -2 -1 -1];
```

```
x=[0 0 0 1 -1 0 0 2 -2];
```

```
nol=-3+(-3):5+5;
```

```
y=conv(x,hol);
```

```
figure(1)
```

```
subplot(3,1,1)
```

```
stem(n,x,'MarkerFaceColor','red')
```

```
title('x(n)')
```

```
axis([nol(1)-1 nol(end)+1 min(x)  
max(x)])
```

```
grid on
```

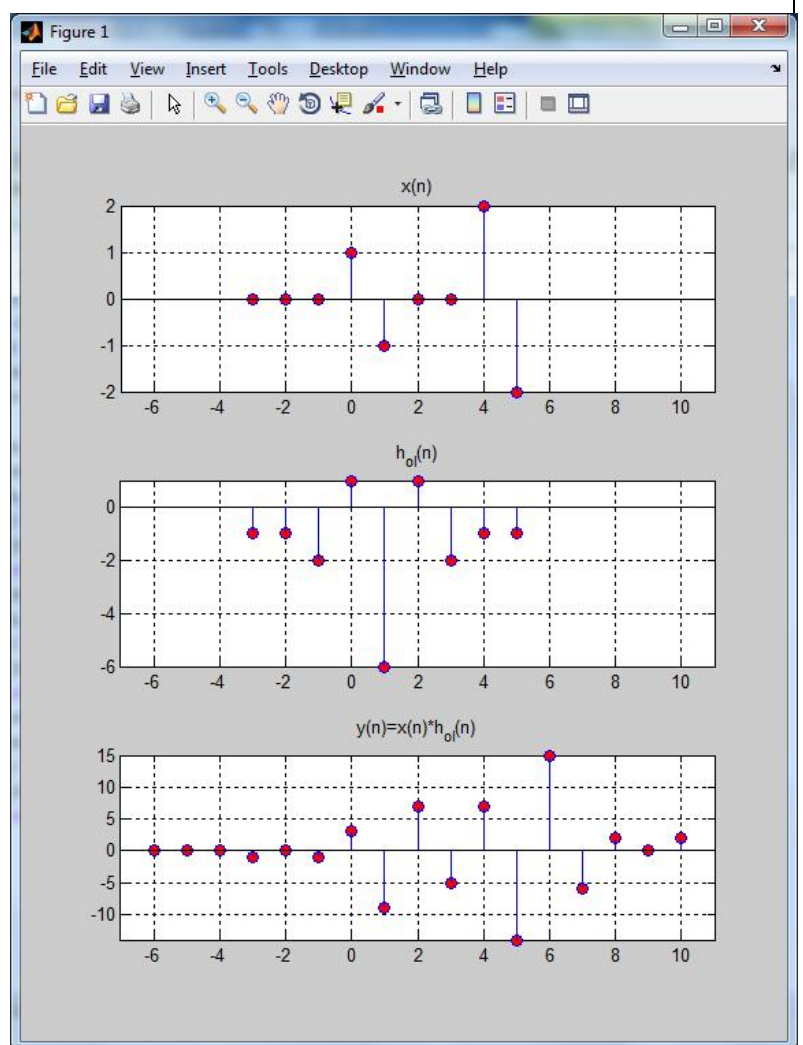
```
subplot(3,1,2)
```

```
stem(n,hol,'MarkerFaceColor','red')
```

```
title('h_{ol}(n)')
```

```
axis([nol(1)-1 nol(end)+1 min(hol)  
max(hol)])
```

```
grid on
```



```

subplot(3,1,3)

stem(nol,y,'MarkerFaceColor','red')

title('y(n)=x(n)*h_{01}(n)')

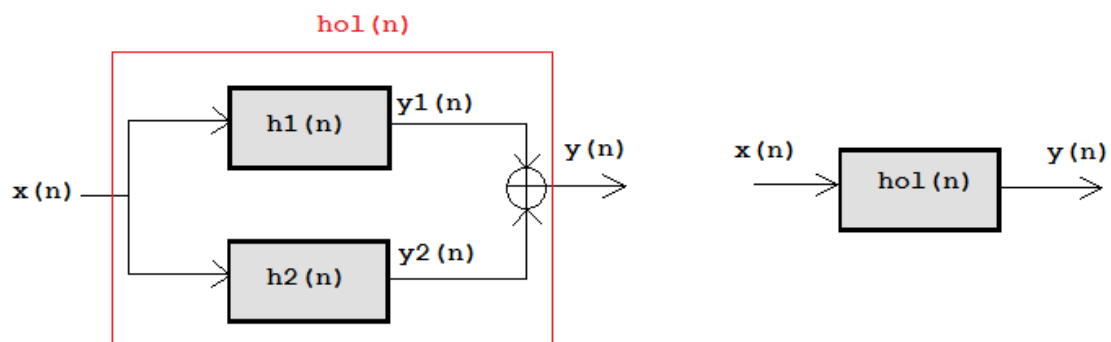
axis([nol(1)-1 nol(end)+1 min(y)
max(y)])

grid on

```

2.2.2 Παράλληλη Διασύνδεση Συστημάτων

Ας θεωρήσουμε ότι το σήμα $x(n)$ δίνεται ως είσοδος στη διάταξη του ακόλουθου σχήματος, οπότε η $x(n)$ θα αποτελέσει είσοδο και στα δύο συστήματα. Το 1^ο σύστημα με κρουστική απόκριση $h_1(n)$ παράγει έξοδο $y_1(n)$, ενώ το 2^ο σύστημα με κρουστική απόκριση $h_2(n)$ παράγει έξοδο $y_2(n)$. Στη συνέχεια, οι έξοδοι $y_1(n)$ και $y_2(n)$ προστίθενται για να δώσουν την έξοδο της συνολικής διάταξης $y(n)$.



Σχήμα 3: (Αριστερά) Παράλληλη διασύνδεση δύο συστημάτων και (δεξιά) η ισοδύναμη συνολική διάταξη

Υπολογίζοντας τις εξόδους των 2 επιμέρους συστημάτων έχουμε:

- Για την έξοδο $y_1(n) = x(n) * h_1(n)$
- Για την έξοδο $y_2(n) = x(n) * h_2(n)$
- Η συνολική έξοδος της διάταξης: $y(n) = y_1(n) + y_2(n) = x(n) * h_1(n) + x(n) * h_2(n) = x(n) * [h_1(n) + h_2(n)]$ (επιμεριστική ιδιότητα συνέλιξης)

Αναφορικά με τη συνολική διάταξη, μπορούμε να θεωρήσουμε, λοιπόν, ότι:

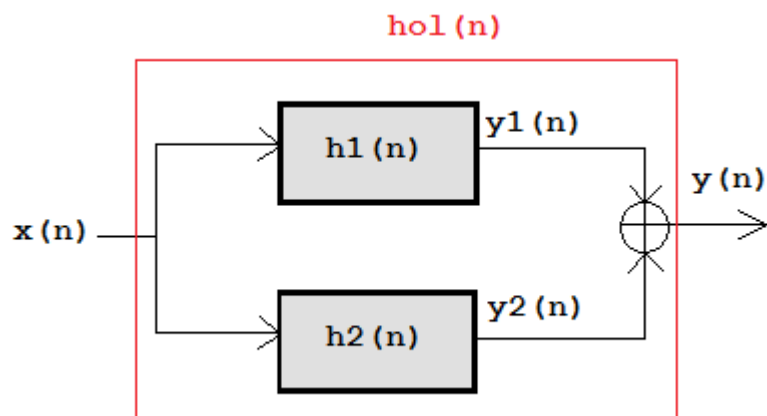
- έχει κρουστική απόκριση $h_{01}(n) = h_1(n) + h_2(n)$ και

- τροφοδοτούμενη με είσοδο $x(n)$ παράγει έξοδο $y(n) = x(n) * h_0(n)$

Άσκηση 6^η

Δίνονται τα σήματα $h_1(n) = \{1, 2, 3, 0, 3, 2, 1\}$ με $n = [-2, 4]$ και $h_2(n) = \{-1, 1, -1\}$ με $n = [-1, 1]$.

- Αν τα σήματα $h_1(n)$ και $h_2(n)$ είναι οι κρουστικές αποκρίσεις των υποσυστημάτων της διάταξης που φαίνεται στο ακόλουθο σχήμα, τότε να υπολογίσετε τη συνολική κρουστική απόκριση $h_0(n)$ της διάταξης.
- Αν στο συνολικό σύστημα δοθεί ως είσοδος το σήμα $x(n) = \{1, -1, 0, 0, 2, -2\}$ με $n = [0, 5]$, τότε να υπολογιστεί η απόκριση συστήματος $y(n)$ της διάταξης.



| | |
|-----------------------|-------------------|
| Κώδικας Matlab | Αποτέλεσμα |
|-----------------------|-------------------|

% Ερώτημα α)

% Για να βρεθεί η συνολική κρουστική απόκριση του συστήματος, αρκεί να δώσουμε ως είσοδο την $\delta(n)$

% θεωρώ το διάστημα $n=[-2,4]$

$n=-2:4;$

$d=zeros(1,length(n));$

$d(n==0)=1;$

$h1=[1 2 3 0 3 2 1];$

$nol1=-2+(-2):4+4;$

$y1=conv(d,h1);$

$h2=[0 -1 1 -1 0 0 0];$

$y2=conv(d,h2);$

$y=y1+y2;$

% Υπολογισμός μέσω επιμεριστικής ιδιότητας συνέλιξης, δηλαδή $hol(n)=h1(n)+h2(n)$

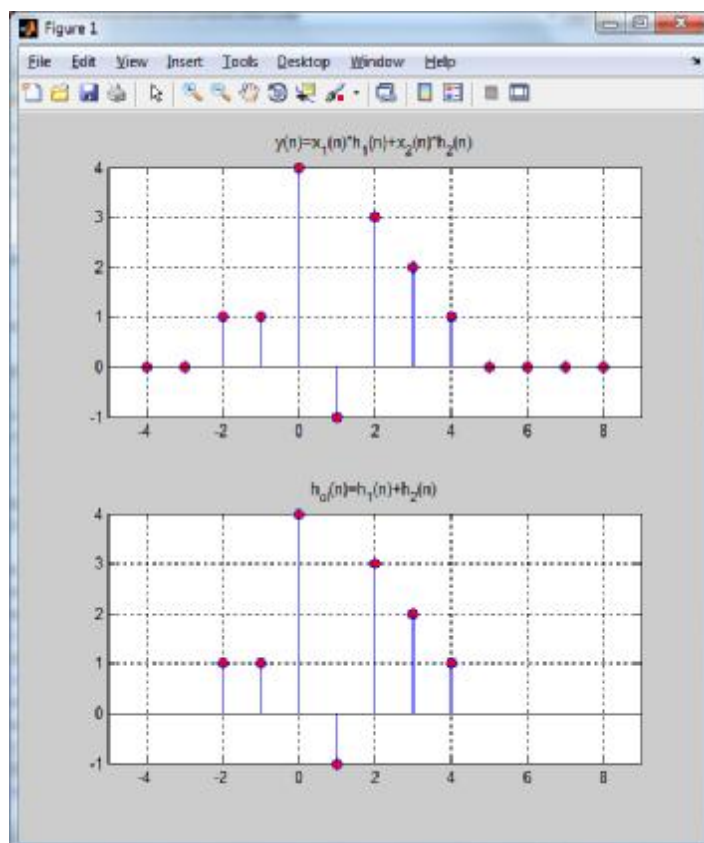
$n2=-2:4;$

$h1=[1 2 3 0 3 2 1];$

$h2=[0 -1 1 -1 0 0 0];$

$nol2=n2;$

$hol=h1+h2;$



Παρατηρήσεις:

Ουσιαστικά επιβεβαιώνεται η επιμεριστική ιδιότητα της συνέλιξης.

Επίσης, υπενθυμίζεται ότι αν σε ένα σύστημα δώσουμε ως είσοδο την $\delta(n)$, τότε το σύστημα δίνει στην έξοδο του την κρουστική του απόκριση. Φαίνεται, λοιπόν, ότι:

$$y(n) = x(n) * hol(n), \text{ με } hol(n) = h1(n) + h2(n)$$

$$\text{και } y(n) = \delta(n) * hol(n) = hol(n)$$

Συνεπώς, όταν 2 συστήματα με κρουστικές αποκρίσεις $h1(n)$ και $h2(n)$ αντίστοιχα, συνδέονται **παράλληλα** έχουν συνολική κρουστική απόκριση που

```

figure(1)

subplot(2,1,1)

stem(nol1,y,'MarkerFaceColor','red')

title('y(n)=x_1(n)*h_1(n)+x_2(n)*h_2(n)')

axis([nol1(1)-1 nol1(end)+1 min(y)
max(y)]), grid on

subplot(2,1,2)

stem(nol2,hol,'MarkerFaceColor','red')

title('h_{ol}(n)=h_1(n)+h_2(n)')

axis([nol1(1)-1 nol1(end)+1 min(hol)
max(hol)]), grid on

```

υπολογίζεται ως εξής:

$$hol(n) = h_1(n) + h_2(n)$$

```

% Ερώτημα b)

% Από το ερώτημα a) γνωρίζουμε ότι η
hol(n)={1,1,4,-1,3,2,1} για n=[-2,4]

% Επειδή x(n)={1,-1,0,0,2,-2} με n=[0,5],
θα ορίσουμε και τα 2 σήματα στο n=[-2,5]
και θα συμπληρώσουμε με μηδενικά τα x(n) και
hol(n)

n=-2:5;

hol=[1 1 4 -1 3 2 1 0];

x=[0 0 1 -1 0 0 2 -2];

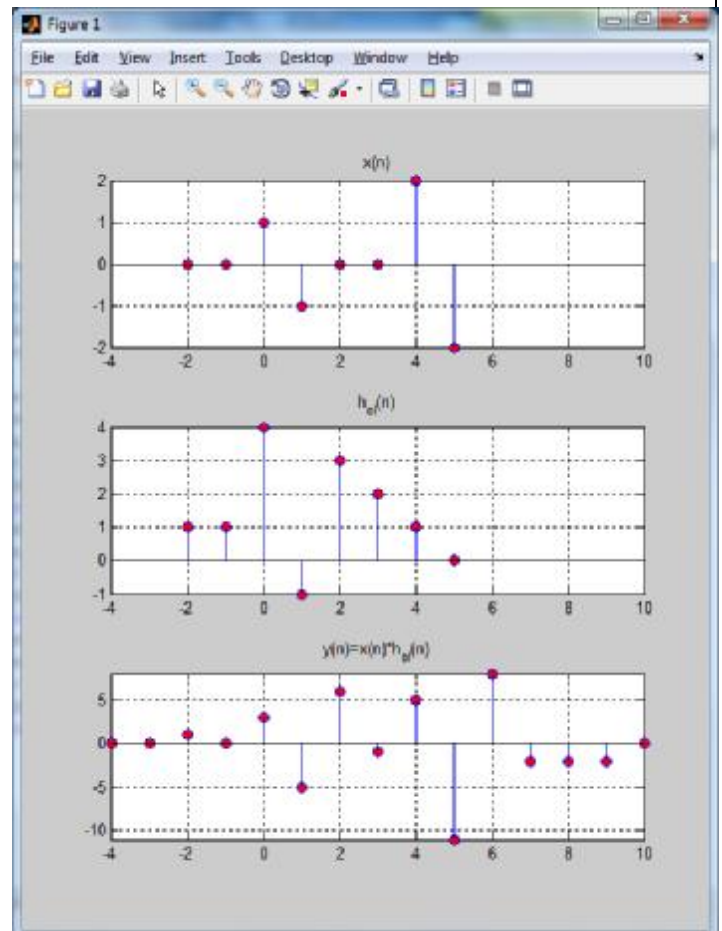
nol=-2+(-2):5+5;

y=conv(x,hol);

figure(1)

subplot(3,1,1)

```



```

stem(n,x,'MarkerFaceColor','red')

title('x(n)')

axis([nol(1) nol(end) min(x) max(x)])

grid on

subplot(3,1,2)

stem(n,hol,'MarkerFaceColor','red')

title('h_{ol}(n)')

axis([nol(1) nol(end) min(hol) max(hol)])

grid on

subplot(3,1,3)

stem(nol,y,'MarkerFaceColor','red')

title('y(n)=x(n)*h_{ol}(n)')

axis([nol(1) nol(end) min(y) max(y)])

grid on

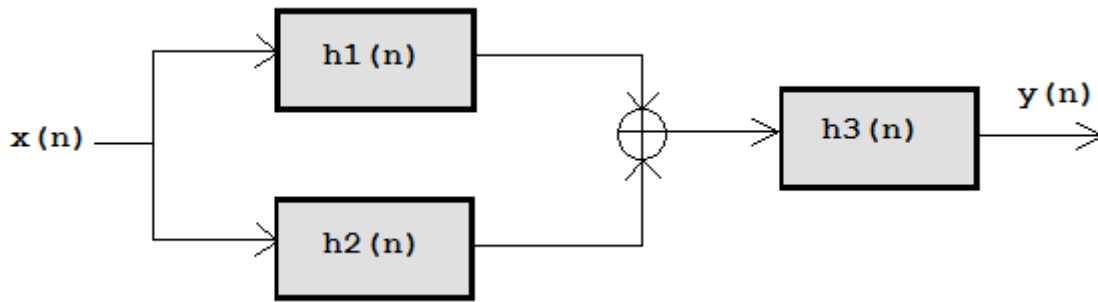
```

2.2.3 Σύνθετες Διασυνδέσεις

Στην περίπτωση σύνθετων διασυνδέσεων (διατάξεων), προσπαθούμε να εντοπίσουμε ποια επιμέρους υποσυστήματα της συνολικής διάταξης βρίσκονται σε σειρά και ποια παράλληλα, ώστε αρχικά να υπολογίσουμε τη συνολική κρουστική απόκριση της διάταξης. Στη συνέχεια, αρκεί να συνελίξουμε την είσοδο που μας δίνεται με τη συνολική κρουστική απόκριση που υπολογίσαμε για την διάταξη, έτσι ώστε να υπολογίσουμε την απόκριση του συνολικού συστήματος (έξοδος της διάταξης).

Άσκηση 7η

Δίνεται η διάταξη συστημάτων του παρακάτω σχήματος.



Αν είναι γνωστές οι κρουστικές αποκρίσεις $h_1(n)=\{1,2,3,4\}$ με $n=[0,3]$, $h_2(n)=3\delta(n+1)+2\delta(n-1)$, $h_3(n)=|n-2|[u(-n+3)-u(-n-2)]$ με $n=[-10,10]$, τότε να υπολογιστεί η έξοδος $y(n)$ της συνολικής διάταξης, όταν η είσοδος είναι:

- a) $x(n)=\sin(n\pi/2)[u(n+3) - u(n-2)]$ με $n=[-10, 10]$
- b) $x(n)=\delta(n)-\delta(n-1)+\delta(n-3)$

Παρατήρηση: Αρχικά θα υπολογίσουμε τη συνολική κρουστική απόκριση της διάταξης και στη συνέχεια θα τη συνελίξουμε με την είσοδο που μας δίνεται κάθε φορά.

Τα συστήματα με $h_1(n)$ και $h_2(n)$ βρίσκονται σε παράλληλη διασύνδεση, συνεπώς μπορούμε να θεωρήσουμε ότι έχουν συνολική κρουστική απόκριση έστω $h_{12}(n)$, όπου $h_{12}(n) = h_1(n) + h_2(n)$.

Το σύστημα με $h_3(n)$ βρίσκεται σε σειρά με το σύνθετο σύστημα $h_{12}(n)$, σχηματίζοντας έτσι το συνολικό σύστημα με συνολική κρουστική απόκριση $h_{ol}(n)$, όπου $h_{ol}(n) = h_{12}(n) * h_3(n) = [h_1(n) + h_2(n)] * h_3(n)$

| Κώδικας Matlab | Αποτέλεσμα |
|----------------|------------|
|----------------|------------|

```

% Η συνολική κρουστική απόκριση της
διάταξης είναι hol(n)=[h1(n)+h2(n)]*h3(n)

% Υπολογισμός h12(n)=h1(n)+h2(n):

% Το h1(n) έχει n=[0,3], ενώ το
h2(n)=3δ(n+1)+2δ(n-1)={3,0,2} με n=[-1,1]
οπότε θεωρούμε το διάστημα n12=[-1,3]

n12=-1:3;

h1=[0 1 2 3 4];

h2=[3 0 2 0 0];

h12=h1+h2;

% Υπολογισμός hol(n)=h12(n)*h3(n)

% Το h12(n) έχει n12=[-1,3] ενώ το h3(n)
είναι ορισμένο στο n=[-10,10], οπότε
θεωρούμε το διάστημα n=[-10,10]

n=-10:10;

% Συμπλήρωση του h12(n) με 0
h12=[zeros(1,9) h12 zeros(1,7)];

% Βρίσκω το h3(n)=|n-2|[u(-n+3)-u(-n-2)]

u=[zeros(1,10) ones(1,10+1)]; % u(n)

u1=[u(3+1:end) ones(1,3)]; % u(n+3)

u1=fliplr(u1); % u(-n+3)

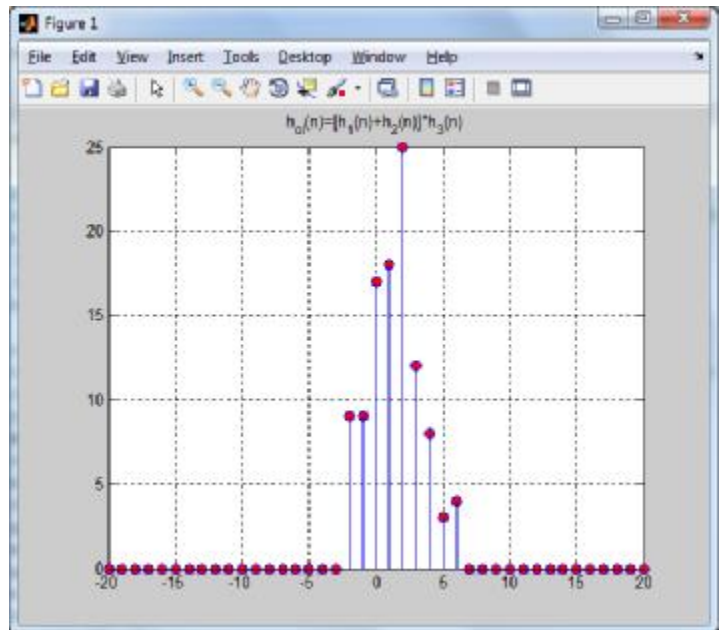
u2=[zeros(1,2) u(1:end-2)]; % u(n-2)

u2=fliplr(u2); % u(-n-2)

h3=abs(n-2).*(u1-u2); % h3(n)

nol=-10+(-10):10+10;

```



Παρατηρούμε ότι η συνολική κρουστική απόκριση $hol(n)$ έχει μη μηδενικές τιμές στο $n=[-2,6]$, με $hol(n)=\{9,9,17,18,25,12,8,3,4\}$

Ωστόσο, έχουμε ορίσει την $hol(n)$ στο $n=[-20,20]$. Αν θέλουμε να κρατήσουμε μόνο τις τιμές της $hol(n)$ στο διάστημα $[-2,6]$ αρκεί να γράψουμε την εντολή:

hol=hol(nol>= -2 & nol<=6), η οποία δίνει:

hol = 9 9 17 18 25 12 8 3 4

```

hol=conv(h12,h3); % hol(n)

% Γραφική παράσταση της hol(n)

figure(1)

stem(nol,hol,'MarkerFaceColor','red')

title('h_{ol}(n)=[h_1(n)+h_2(n)]*h_3(n)')

axis([nol(1) nol(end) min(hol) max(hol)])

grid on

```

% Ερώτημα α)

% Έχουμε βρει την hol(n) και την έχουμε ορίσει στο nol=[-20,20]

% Θα υπολογίσουμε τα πλάτη της x(n) και θα εκτελέσουμε τη συνέλιξη των x(n),h(n)

% Υπολογισμός της x(n)

```
%x(n)=sin(nπ/2)[u(n+3)-u(n-2)],n=[-10,10]
```

```
n=-10:10;
```

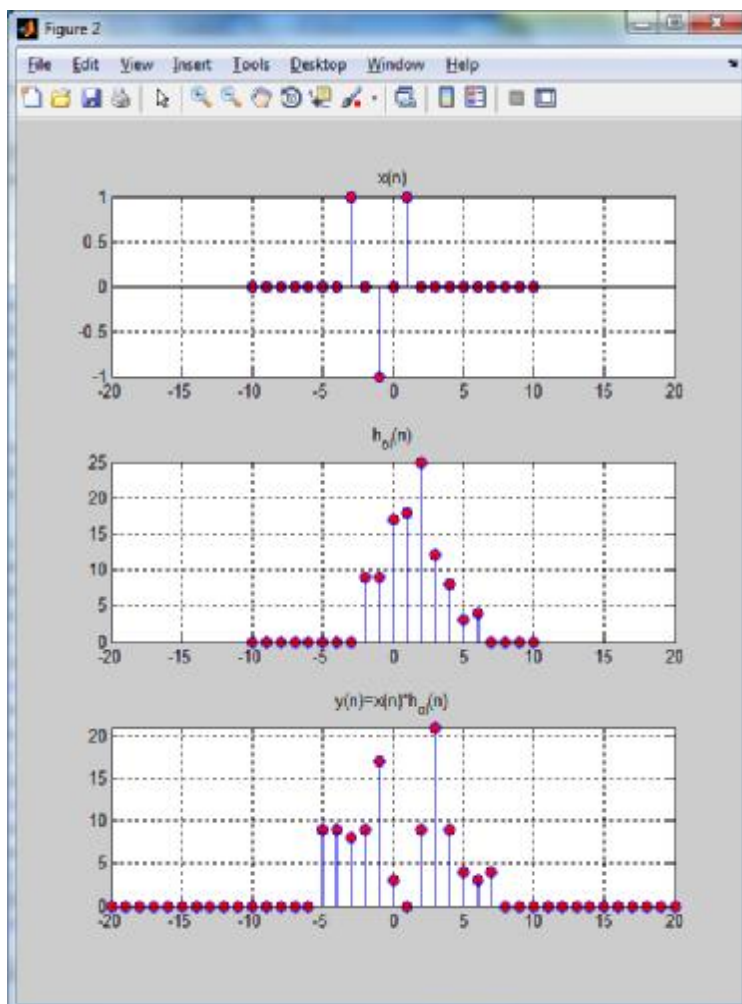
```
u=[zeros(1,10) ones(1,10+1)]; % u(n)
```

```
u1=[u(3+1:end) ones(1,3)]; % u(n+3)
```

```
u2=[zeros(1,2) u(1:end-2)]; % u(n-2)
```

```
x=sin(n*pi/2).*(u1-u2); % x(n)
```

% Για τη συνέλιξη των x(n) και h(n) θα ορίσουμε τα σήματα στο ίδιο διάστημα. Επειδή η x(n) έχει n=[-10,10], ενώ η hol(n) έχει τα μη μηδενικά πλάτη στο διάστημα n=[-2,6], μπορούμε να ορίσουμε ως



```

διάστημα το n=[-10,10] και από την hol(n)
να κρατήσουμε τα πλάτη στο διάστημα n=[-
10,10]

h=hol(nol>=-10 & nol<=10);

nol=-10+(-10):10+10;

y=conv(x,h);

% Γραφική παράσταση αποτελέσματος

figure(2)

subplot(3,1,1)

stem(n,x,'MarkerFaceColor','red')

title('x(n)')

axis([nol(1) nol(end) min(x) max(x)])

grid on

subplot(3,1,2)

stem(n,h,'MarkerFaceColor','red')

title('h_{ol}(n)')

axis([nol(1) nol(end) min(h) max(h)])

grid on

subplot(3,1,3)

stem(nol,y,'MarkerFaceColor','red')

title('y(n)=x(n)*h_{ol}(n)')

axis([nol(1) nol(end) min(y) max(y)])

grid on

```

% Ερώτημα b)

```
% Έχουμε βρει την hol(n) και την έχουμε  
ορίσει στο nol=[-20,20]
```

```
%  $x(n)=\delta(n)-\delta(n-1)+\delta(n-3)=\{1,-1,0,1\}$  με  
n=[0,3]
```

```
% Γνωρίζουμε ότι τα μη μηδενικά πλάτη της  
hol(n) βρίσκονται στο n=[-2,6] και ότι για  
την x(n) είναι n=[0,3]. Άρα, μπορούμε να  
ορίσουμε ως διάστημα το n=[-2,6] και να  
συμπληρώσουμε με 0 το x(n)
```

```
n=-2:6;
```

```
x=[0 0 1 -1 0 1 0 0 0];
```

```
% Για την hol(n) κρατάμε τα πλάτη στο  
διάστημα n=[-2,6]
```

```
h=hol(nol>=-2 & nol<=6);
```

```
nol=-2+(-2):6+6;
```

```
y=conv(x,h);
```

```
% Γραφική παράσταση αποτελέσματος
```

```
figure(3)
```

```
subplot(3,1,1)
```

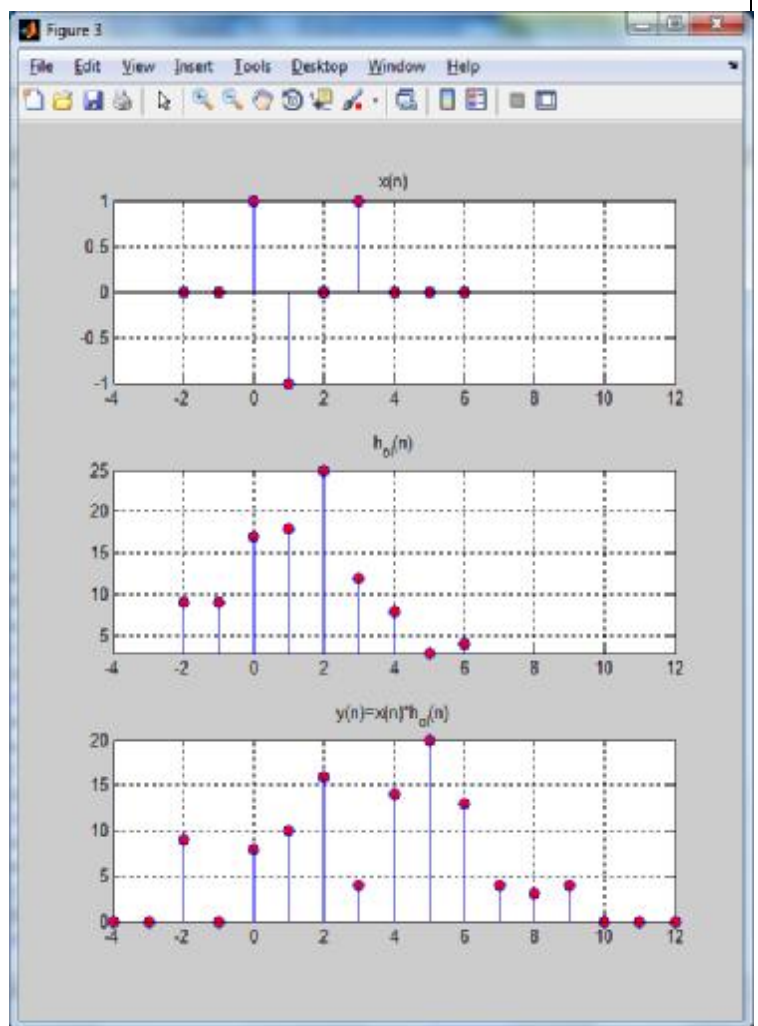
```
stem(n,x,'MarkerFaceColor','red')
```

```
title('x(n)')
```

```
axis([nol(1) nol(end) min(x) max(x)])
```

```
grid on
```

```
subplot(3,1,2)
```




```

stem(n,h,'MarkerFaceColor','red')

title('h_{ol}(n)')

axis([nol(1) nol(end) min(h) max(h)])

grid on

subplot(3,1,3)

stem(nol,y,'MarkerFaceColor','red')

title('y(n)=x(n)*h_{ol}(n)')

axis([nol(1) nol(end) min(y) max(y)])

grid on

```

3 Άλυτες Ασκήσεις

1. Έστω σύστημα με κρουστική απόκριση $h(n) = |n-2| \cdot [u(n) - u(n-6)]$ για $n = [-10, 10]$.

Να βρεθεί η απόκριση του συστήματος αν το τροφοδοτήσουμε με είσοδο:

α) $x_1(n) = \delta(n)$

β) $x_2(n) = u(n) - u(n-5)$

γ) $x_3(n) = 2\delta(n+1) - 3\delta(n) + \delta(n-2)$

δ) $x_4(n) = [-1 \ 1 \ 2 \ -2 \ 0]$, όπου $n = [-1, 3]$

2. Έστω τα υποσυστήματα με κρουστικές αποκρίσεις $h_1(n) = \{1, 2, 3, 4\}$ με $n = [0, 3]$ και $h_2(n) = 3\delta(n+1) + 2\delta(n-1)$ με $n = [-10, 10]$. Να υπολογιστεί η έξοδος $y(n)$ της συνολικής διάταξης αν τα υποσυστήματα διασυνδεθούν σε σειρά και παράλληλα. Ως είσοδο θεωρείστε τα ακόλουθα σήματα:

α) $x(n) = \sin(n\pi/2)[u(n) - u(n-5)]$ με $n = [-10, 10]$

β) $x(n) = \delta(n) - 2\delta(n-1) - 3\delta(n-3)$

4 Επίλογος – Σύνδεση με τα Επόμενα

Στην παρούσα εργαστηριακή άσκηση καλύφθηκε η έννοια της κρουστικής απόκρισης ενός συστήματος διακριτού χρόνου και η πράξη της γραμμικής συνέλιξης

με τις ιδιότητές της (αντιμεταθετική, προσεταιριστική, επιμεριστική). Επίσης, ασχοληθήκαμε με τους βασικούς τύπους διασύνδεσης (σε σειρά και παράλληλα) συστημάτων και με τον υπολογισμό της απόκρισης (εξόδου) σύνθετων διατάξεων συστημάτων διακριτού χρόνου.

Στην επόμενη εργαστηριακή άσκηση θα παρουσιαστούν οι γραμμικές διαφορικές εξισώσεις σταθερών συντελεστών (ή αλλιώς εξισώσεις διαφορών) και οι εφαρμογές τους στα συστήματα διακριτού χρόνου.

5 Βιβλιογραφία

Hayes , Μ. Η. (2002). *Ψηφιακή Επεξεργασία Σήματος*. (Α. Δημητριάδης, Μεταφρ.) Θεσσαλονίκη: Τζιόλα.

Hsu, H. P. (2002). *Αναλογικές και Ψηφιακές Επικοινωνίες*. Θεσσαλονίκη: Τζιόλα.

Jayant, N. S., & Noll, P. (1984). *Digital Coding of Waveforms: Principles and Applications to Speech and Video*. Prentice Hall.

Mano, M. M., & Ciletti, M. D. (2012). *Digital Design (5 εκδ.)*. Pearson Education.

MathWorks - DSP. (2017). Ανάκτηση 2 10, 2017, από Getting Started with Matlab DSP: <https://www.mathworks.com/help/dsp/getting-started-with-dsp-system-toolbox.html>

Proakis, J. G., & Salehi, M. (2008). *Digital Communications (5 εκδ.)*. McGraw-Hill Higher Education.

Μουρτζόπουλος, Ι. (2012). *Ψηφιακή Τεχνολογία Ήχου*. Πάτρα: Πανεπιστήμιο Πατρών.

Εργαστηριακή Άσκηση 6: Εξισώσεις Διαφορών

1 Εισαγωγικά Στοιχεία

1.1 Σκοπός

Σκοπός της παρούσας εργαστηριακής άσκησης είναι η εξοικείωση των φοιτητών – σπουδαστών με τον υπολογισμό της απόκρισης συστήματος (δηλαδή της εξόδου), όταν το σύστημα περιγράφεται από μία γραμμική εξίσωση διαφορών σταθερών Συντελεστών (ΓΕΔΣΣ).

1.2 Αναμενόμενα Μαθησιακά Αποτελέσματα

Με την ολοκλήρωση της παρούσας εργαστηριακής άσκησης, ο φοιτητής – σπουδαστής θα είναι σε θέση να:

- Υπολογίζει την απόκριση (έξοδο) ενός συστήματος, όταν το σύστημα περιγράφεται από μία ΓΕΔΣΣ
- Υπολογίζει την κρουστική απόκριση ενός συστήματος, όταν το σύστημα περιγράφεται από μία ΓΕΔΣΣ.

1.3 Εκτιμώμενη Διάρκεια

Ο εκτιμώμενος χρόνος που απαιτείται από τον φοιτητή – σπουδαστή για την ολοκλήρωση της παρούσας εργαστηριακής άσκησης είναι 2 διδακτικές ώρες.

1.4 Προαπαιτούμενα

Διάλεξη 4 της θεωρίας.

1.5 Εισαγωγή

Στην άσκηση αυτή θα ασχοληθούμε με τον υπολογισμό της εξόδου και της κρουστικής απόκρισης συστημάτων, όταν αυτά περιγράφονται από ΓΕΔΣΣ.

Στην προηγούμενη εργαστηριακή άσκηση, είδαμε ότι η γραμμική συνέλιξη μέσω της εντολής `conv` του Matlab, μας επιτρέπει να υπολογίσουμε την έξοδο ενός

συστήματος, υπό την προϋπόθεση όμως ότι η κρουστική απόκριση $h(n)$ του συστήματος είναι πεπερασμένη (όχι άπειρη). Ωστόσο, για συστήματα με άπειρη κρουστική απόκριση η `conv` δεν δίνει σωστά αποτελέσματα. Η λύση στο πρόβλημα αυτό δίνεται αν περιγράψουμε το σύστημα μέσω μίας εξίσωσης διαφορών.

1.6 Κατάλογος Χρήσιμων Συναρτήσεων Matlab

Κατά την εκπόνηση των ασκήσεων χρησιμοποιούνται οι ακόλουθες συναρτήσεις – εντολές του Matlab:

| Συνάρτηση | Επεξήγηση |
|----------------------|--|
| <code>length</code> | Επιστρέφει το μήκος ενός πίνακα-διανύσματος |
| <code>zeros</code> | Δημιουργία πίνακα – διανύσματος με μηδενικά |
| <code>filter</code> | Υπολογίζει την απόκριση ενός συστήματος που περιγράφεται από μία εξίσωση διαφορών |
| <code>impz</code> | Υπολογίζει την κρουστική απόκριση ενός συστήματος που περιγράφεται από μία εξίσωση διαφορών |
| <code>figure</code> | Δημιουργεί νέο παράθυρο απεικόνισης γραφικών παραστάσεων |
| <code>subplot</code> | Τμηματοποίηση ενός παραθύρου για γραφικές παραστάσεις σε γραμμές και στήλες, προκειμένου στο ίδιο παράθυρο να απεικονιστούν πολλαπλές γραφικές παραστάσεις |
| <code>stem</code> | Γραφική παράσταση διακριτών σημάτων – ακολουθιών |
| <code>title</code> | Τοποθετεί τίτλο σε μια γραφική παράσταση |
| <code>grid</code> | Προσθέτει – αφαιρεί πλέγμα στις γραφικές παραστάσεις |

2 Περιγραφή Συστήματος Μέσω Εξίσωσης Διαφορών

Κάθε γραμμικό και χρονικά αμετάβλητο (ΓΧΑ) σύστημα μπορεί να περιγραφεί από μία εξίσωση διαφορών με σταθερούς συντελεστές (ΓΕΔΣΣ) της μορφής:

$$\sum_{k=0}^N a_k y(n-k) = \sum_{m=0}^M b_m x(n-m)$$

όπου:

- $y(n-k)$ και $x(n-m)$ η έξοδος και η είσοδος κάθε χρονική στιγμή,
- a_k και b_k οι συντελεστές της εξόδου και της εισόδου αντίστοιχα.

Αν θεωρήσουμε ότι $a_0=1$, οπότε η προηγούμενη σχέση γράφεται ως εξής:

$$y(n) = \sum_{m=0}^M b_m x(n-m) - \sum_{k=1}^N a_k y(n-k)$$

2.1 Υπολογισμός Απόκρισης Συστήματος

Ο υπολογισμός της απόκρισης (εξόδου) ενός συστήματος που περιγράφεται από μία ΓΕΔΣΣ γίνεται στο Matlab με τη συνάρτηση `filter`, η οποία συντάσσεται ως εξής:

$$y = \text{filter}(b,a,x)$$

όπου:

- y : έξοδος του συστήματος
- b : συντελεστές των όρων της εισόδου x της ΓΕΔΣΣ
- a : συντελεστές των όρων της εξόδου y της ΓΕΔΣΣ
- x : σήμα εισόδου

Όταν έχουμε μία ΓΕΔΣΣ και θέλουμε να υπολογίσουμε την έξοδο $y(n)$ του συστήματος, εργαζόμαστε ως εξής:

1. Μεταφέρουμε στο αριστερό μέρος της εξίσωσης όλους τους όρους που αναφέρονται στην έξοδο y και τους τοποθετούμε στη σειρά ξεκινώντας από τον όρο $y(n)$ και συνεχίζοντας με τους $y(n-1)$, $y(n-2)$ κοκ.

2. Έτσι, στο δεξιό μέρος θα έχουν απομείνει οι όροι που αναφέρονται στην είσοδο x , τους οποίους επίσης τοποθετούμε στη σειρά ξεκινώντας από τον $x(n)$ και συνεχίζοντας με τους $x(n-1)$, $x(n-2)$ κ.ο.κ.
3. Φτιάχνουμε το διάνυσμα a με τους συντελεστές των όρων της εξόδου y και το διάνυσμα b με τους συντελεστές των όρων της εισόδου x . Για τους όρους που λείπουν θέτουμε τον αντίστοιχο συντελεστή ίσο με 0.
4. Υπολογίζουμε την έξοδο y με την εντολή $y = \text{filter}(b,a,x)$, όπου x το σήμα εισόδου. Η έξοδος y θα έχει τόσα στοιχεία όσα έχει και το σήμα εισόδου x .

Άσκηση 1^η

Έστω σύστημα που περιγράφεται από την ΓΕΔΣΣ: $y(n) = 0.5y(n-1) + x(n) - 0.9x(n-2)$

Αν τροφοδοτήσουμε το σύστημα με το σήμα εισόδου $x(n) = \{1, 2, 3\}$ με $n=[0,2]$, τότε να υπολογιστεί η έξοδος του συστήματος $y(n)$, για χρόνο $n=[-5, 20]$.

Παρατήρηση: Μεταφέροντας αριστερά τους όρους y , η ΓΕΔΣΣ γράφεται ως εξής: $y(n) - 0.5y(n-1) = x(n) - 0.9x(n-2)$, οπότε θα έχουμε $a=[1 \ -0.5]$ και $b=[1 \ 0 \ -0.9]$. Στο δεξί μέλος δεν υπάρχει ο όρος $x(n-1)$, άρα θέτουμε το συντελεστή του ίσο με 0.

| Κώδικας Matlab | Αποτέλεσμα |
|----------------|------------|
|----------------|------------|

```

n=[-5:20]; % Συνολικός χρόνος

% Ορίζω το x(n) στο ζητούμενο
χρόνο

x=zeros(1,length(n));

x(n==0)=1;

x(n==1)=2;

x(n==2)=3;

% Διανύσματα συντελεστών a,b

a=[1 -0.5];

b=[1 0 -0.9];

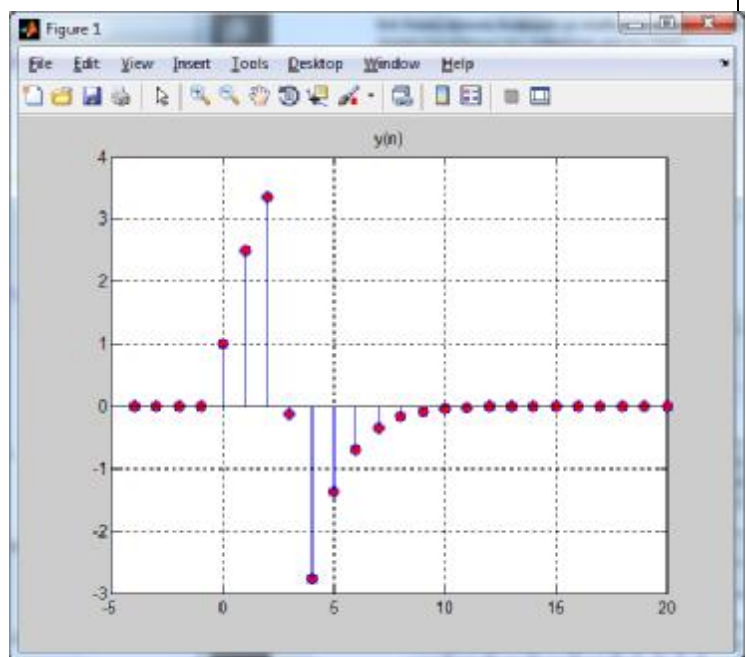
y=filter(b,a,x);

figure(1)

stem(n,y,'MarkerFaceColor','red')

title('y(n)'), grid on

```



Άσκηση 2^η

Σύστημα περιγράφεται από την ΓΕΔΣΣ: $y(n)+0.3y(n-1)-0.3y(n-2)=0.5x(n)-x(n-1)$.
 Αν τροφοδοτήσουμε το σύστημα με το σήμα εισόδου $x(n) = \{3 \ 4 \ 1 \ 1 \ 0 \ 0 \ 1 \ 1 \ 0 \ 0 \ 1\}$ με $n=[0,10]$, τότε να υπολογιστεί η έξοδος του συστήματος $y(n)$ για $n=[-2, 15]$.

Παρατήρηση: Δεν χρειάζεται να μεταφέρουμε και να διατάξουμε τους όρους y , ενώ από την ΓΕΔΣΣ δεν απουσιάζει κάποιος ενδιάμεσος όρος y ή x . Άρα από τη ΓΕΔΣΣ φαίνεται αμέσως ότι $a=[1 \ 0.3 \ -0.3]$ και $b=[0.5 \ -1]$.

Κώδικας Matlab

Αποτέλεσμα

```

n=[-2:15]; % Συνολικός χρόνος

% Ορίζω το x(n) στο ζητούμενο
χρόνο

x=[3 4 1 1 0 0 1 1 0 0 1];

x=[0 0 x 0 0 0 0 0];

% Διανύσματα συντελεστών a,b

a=[1 0.3 -0.3];

b=[0.5 -1];

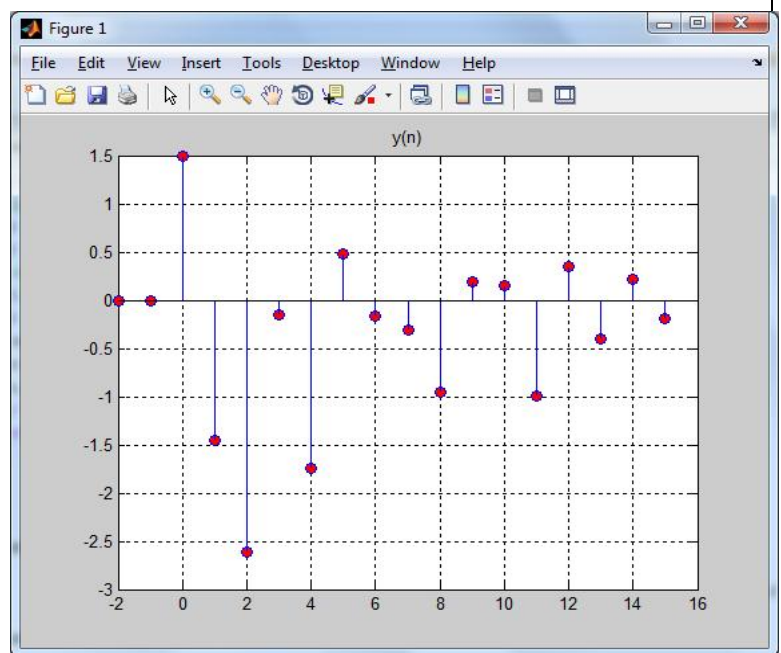
y=filter(b,a,x);

figure(1)

stem(n,y,'MarkerFaceColor','red')

title('y(n)'),grid on

```



Άσκηση 3^η

Σύστημα περιγράφεται από την ΓΕΔΣΣ: $0.5y(n) = 0.3y(n - 2) + 0.5x(n) - x(n - 2)$. Αν τροφοδοτήσουμε το σύστημα με είσοδο $x(n) = \{-1 0 1 2 3 4\}$ με $n = [-2, 3]$, τότε να υπολογιστεί η έξοδος του συστήματος $y(n)$ για $n = [-10, 100]$.

Παρατήρηση: Μεταφέρουμε στο αριστερό μέρος της ΓΕΔΣΣ τους όρους y , οπότε η ΓΕΔΣΣ γράφεται ως εξής: $0.5y(n) - 0.3y(n - 2) = 0.5x(n) - x(n - 2)$. Επίσης, απουσιάζουν οι όροι $y(n - 1)$ και $x(n - 1)$, οπότε θέτοντας τους συντελεστές τους ίσους με 0 θα είναι: $a = [0.5 0 -0.3]$ και $b = [0.5 0 -1]$.

| Κώδικας Matlab | Αποτέλεσμα |
|----------------|------------|
|----------------|------------|


```

n=[-10:100]; % Συνολικός χρόνος

% Ορίζω το x(n) στο ζητούμενο
χρόνο

x=zeros(1,length(n));

x(n==-2)=-1;

x(n==0)=1;

x(n==1)=2;

x(n==2)=3;

x(n==3)=4;

% Διανύσματα συντελεστών a,b

a=[0.5 0 -0.3];

b=[0.5 0 -1];

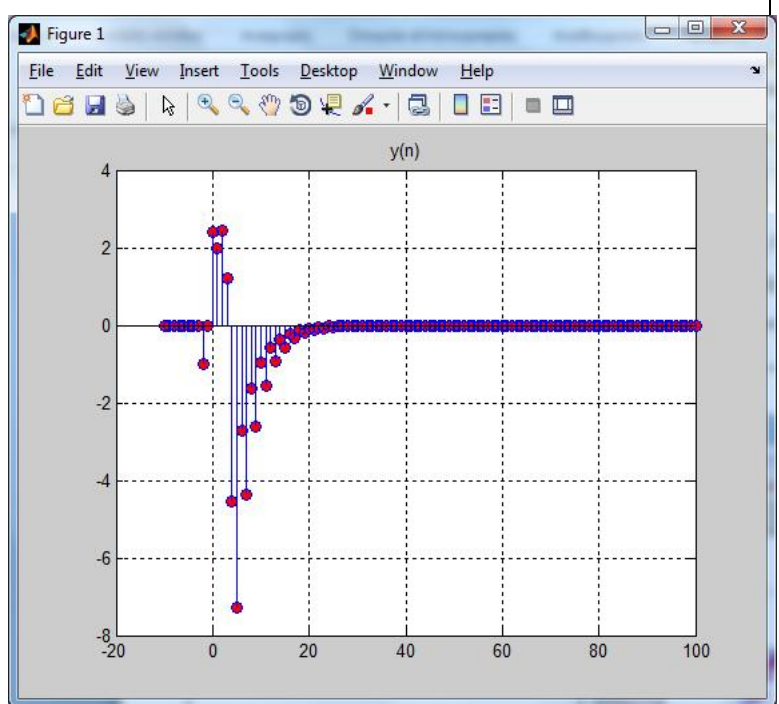
y=filter(b,a,x);

figure(1)

stem(n,y,'MarkerFaceColor','red')

title('y(n)'),grid on

```



2.2 Υπολογισμός Κρουστικής Απόκρισης Συστήματος

Σε ένα σύστημα περιγράφεται μέσω μίας ΓΕΔΣΣ μπορούμε να βρούμε την κρουστική του απόκριση $h(n)$ με δύο τρόπους:

1. Αν δώσουμε στο σύστημα ως είσοδο την κρουστική ώση $\delta(n)$, τότε το σύστημα θα εμφανίσει στην έξοδό του την κρουστική του απόκριση $h(n)$.
2. Αν χρησιμοποιήσουμε την εντολή `impz`. Η εντολή αυτή συντάσσεται ως εξής:

$$h = \text{impz}(b,a,n)$$

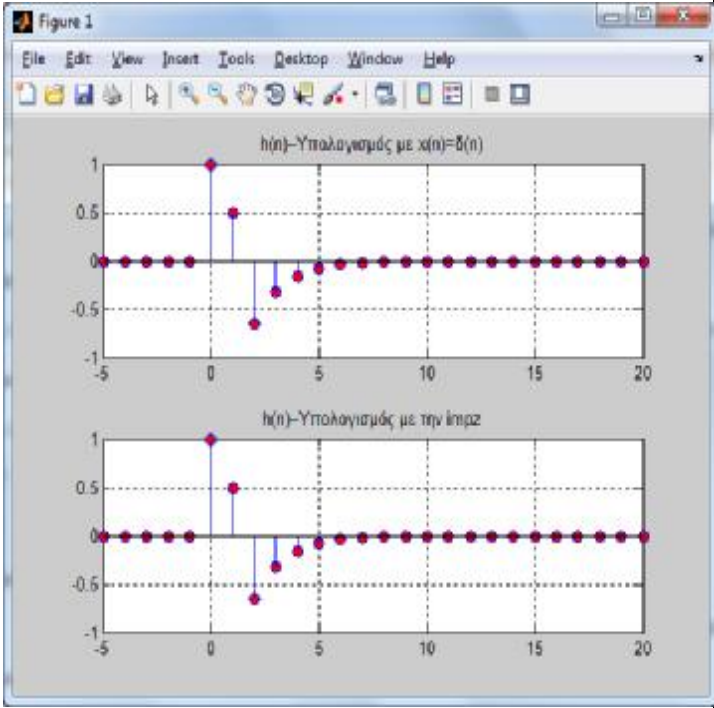
όπου:

- h : η κρουστική απόκριση του συστήματος
- b : συντελεστές των όρων της εισόδου x της ΓΕΔΣΣ
- a : συντελεστές των όρων της εξόδου y της ΓΕΔΣΣ
- n : ο χρόνος που θέλουμε να υπολογιστεί η κρουστική απόκριση

Άσκηση 4^η

Σύστημα περιγράφεται από την ΓΕΔΣΣ: $y(n) = 0.5y(n-1) + x(n) - 0.9x(n-2)$.
Υπολογίστε την κρουστική απόκριση $h(n)$ για $n=[-5, 20]$ και με τους δύο τρόπους.

Παρατήρηση: Η ΓΕΔΣΣ γράφεται ως εξής: $y(n) - 0.5y(n-1) = x(n) - 0.9x(n-2)$,
οπότε θα έχουμε $a=[1 \ -0.5]$ και $b=[1 \ 0 \ -0.9]$.

| Κώδικας Matlab | Αποτέλεσμα |
|--|---|
| <pre>% Α τρόπος - δίνω ως είσοδο x(n)=δ(n) n=[-5:20]; % Συνολικός χρόνος % Ορίζω το x(n) στο ζητούμενο χρόνο x=zeros(1,length(n)); x(n==0)=1; % Διανύσματα συντελεστών a,b a=[1 -0.5]; b=[1 0 -0.9]; h1=filter(b,a,x); % Β τρόπος - χρήση της impz h2=impz(b,a,n); figure(1)</pre> |  <p><u>Παρατηρήσεις:</u></p> |

| | |
|---|--|
| <pre>subplot(2,1,1) stem(n,h1,'MarkerFaceColor','red') title('h(n)-Υπολογισμός με x(n)=δ(n)') grid on subplot(2,1,2) stem(n,h2,'MarkerFaceColor','red') title('h(n)-Υπολογισμός με την impz') grid on</pre> | <p>Επιβεβαιώνεται ότι και οι δύο τρόποι δίνουν τα ίδια αποτελέσματα.</p> |
|---|--|

Άσκηση 5^η

Σύστημα περιγράφεται από την ΓΕΔΣΣ: $0.5y(n) = 0.3y(n - 2) + 0.5x(n) - x(n - 2)$. Να υπολογιστεί η κρουστική του απόκριση $h(n)$ για $n=[-5, 30]$ και με τους 2 τρόπους.

Παρατήρηση: Η ΓΕΔΣΣ γράφεται ως εξής: $0.5y(n) - 0.3y(n - 2) = 0.5x(n) - x(n - 2)$, όπου $a=[0.5 \ 0 \ -0.3]$ και $b=[0.5 \ 0 \ -1]$.

| Κώδικας Matlab | Αποτέλεσμα |
|----------------|------------|
|----------------|------------|

```

% A τρόπος - δίνω είσοδο x(n)=δ(n)

n=[-5:30]; % Συνολικός χρόνος

% Ορίζω το x(n) στο ζητούμενο χρόνο
x=zeros(1,length(n));

x(n==0)=1;

% Διανύσματα συντελεστών a,b
a=[0.5 0 -0.3]; b=[0.5 0 -1];

h1=filter(b,a,x);

% B τρόπος - χρήση της impz

h2=impz(b,a,n);

figure(1)

subplot(2,1,1)

stem(n,h1,'MarkerFaceColor','red')

title('h(n)-Υπολογισμός με x(n)=δ(n)')

grid on

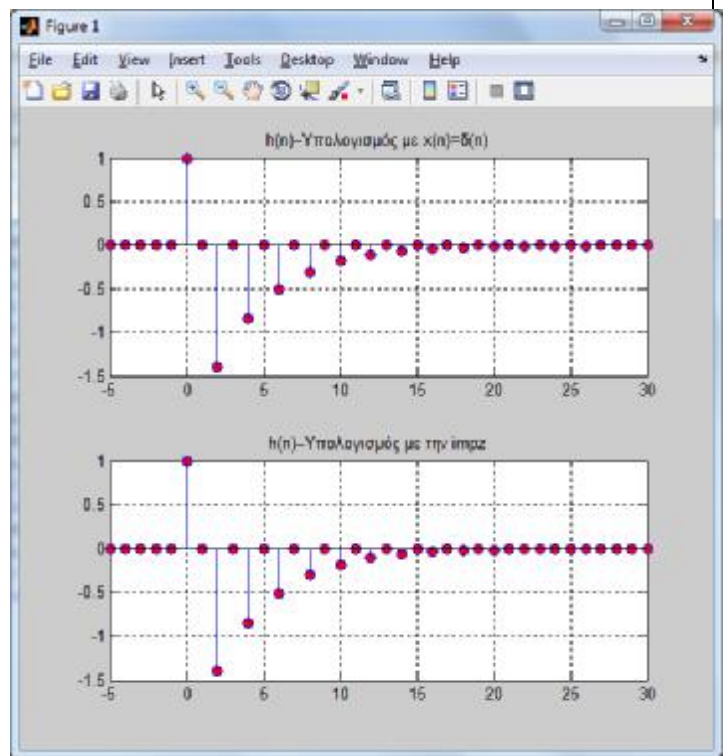
subplot(2,1,2)

stem(n,h2,'MarkerFaceColor','red')

title('h(n)-Υπολογισμός με την impz')

grid on

```



Παρατηρήσεις:

Επιβεβαιώνεται ότι και οι δύο τρόποι δίνουν τα ίδια αποτελέσματα.

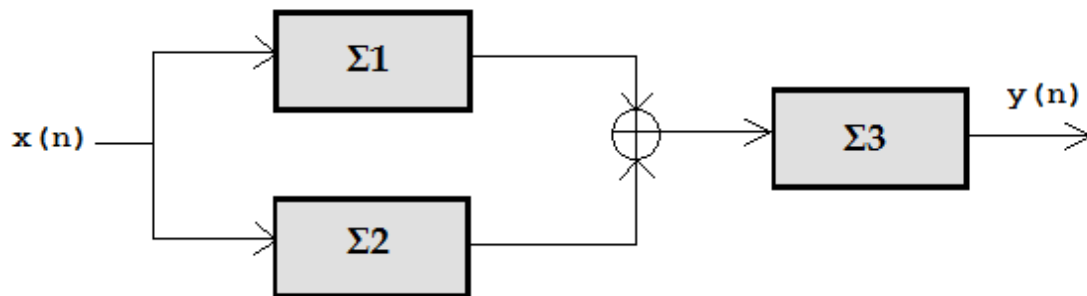
2.3 Διασυνδέσεις Συστημάτων

Όταν έχουμε διασύνδεση συστημάτων εντοπίζουμε ποια από τα επιμέρους υποσυστήματα βρίσκονται σε σειρά και ποια παράλληλα, προκειμένου να υπολογίσουμε τη συνολική κρουστική απόκριση της διάταξης.

Για να βρούμε την απόκριση του συνολικού συστήματος αρκεί να συνελίξουμε την είσοδο που μας δίνεται με τη συνολική κρουστική απόκριση που υπολογίσαμε.

Άσκηση 6η

Δίνεται η διάταξη συστημάτων του παρακάτω σχήματος.



Για τα συστήματα Σ1, Σ2 και Σ3 γνωρίζουμε τα εξής:

- Το Σ1 έχει ΓΕΔΣΣ: $2y(n) = -y(n-1) - 0.5y(n-2) + 0.1x(n-1) + 0.1x(n-3)$
- Το Σ2 έχει ΓΕΔΣΣ: $y(n) - 0.5x(n) = 1.5x(n) + x(n-2)$
- Το Σ3 έχει ΓΕΔΣΣ: $y(n) - 0.5y(n-2) + 0.8y(n-3) = x(n) + 0.4x(n-1) - 1.4x(n-2)$

Να βρεθούν τα εξής:

- a) η συνολική κρουστική απόκριση της παραπάνω διάταξης για $n=[0, 20]$
- b) η απόκριση του συστήματος για $n=[0, 20]$, όταν δώσουμε είσοδο την $x(n)=n0.5^n$ με $n=[0, 10]$

Παρατηρήσεις:

- Για το ερώτημα α) παρατηρούμε ότι τα συστήματα Σ1 και Σ2 βρίσκονται παράλληλα, ενώ το σύστημα Σ3 βρίσκεται σε σειρά με αυτά. Έτσι, η συνολική κρουστική απόκριση είναι $h_{ol}(n) = [h_1(n) + h_2(n)] * h_3(n)$, όπου h_1 , h_2 και h_3 οι κρουστικές αποκρίσεις των Σ1, Σ2 και Σ3 αντίστοιχα. Για τον υπολογισμό τους μπορούμε να χρησιμοποιήσουμε την εντολή `impz`.

Αναδιατάσσοντας τους όρους οι ΓΕΔΣΣ για τα επιμέρους συστήματα γράφονται ως εξής:

- Σ1: $2y(n) + y(n-1) + 0.5y(n-2) = 0.1x(n-1) + 0.1x(n-3)$
 - Σ2: $y(n) = 2x(n) + x(n-2)$
 - Σ3: $y(n) - 0.5y(n-2) + 0.8y(n-3) = x(n) + 0.4x(n-1) - 1.4x(n-2)$
- Για το ερώτημα b) θα συνελίξουμε τη δοσμένη είσοδο $x(n)$ με τη συνολική κρουστική απόκριση $h_{ol}(n)$ που βρήκαμε στο ερώτημα α).

| Κώδικας Matlab | Αποτέλεσμα |
|--|------------|
| <pre> % Ερώτημα a) % Η συνολική κρουστική απόκριση είναι hol(n)=[h1(n)+h2(n)]*h3(n) n=[0:20]; % Βρίσκω την κρουστική απόκριση του Σ1 a1=[2 1 0.5]; b1=[0 0.1 0 0.1]; h1=impz(b1,a1,n); % Βρίσκω την κρουστική απόκριση του Σ2 a2=[1]; b2=[2 0 1]; h2=impz(b2,a2,n); % Βρίσκω την κρουστική απόκριση του Σ3 a3=[1 0 -0.5 0.8]; b3=[1 0.4 -1.4]; h3=impz(b3,a3,n); </pre> | |

```

h12=h1+h2;

nol=[0+0:20+20];

hol=conv(h12,h3);

hol_2=hol(nol>=0 & nol<=20);

% Γραφική παράσταση της hol(n) στο
n=[0,20]

figure('name','Συνολική κρουστική
απόκριση')

stem(n,hol_2,'MarkerFaceColor','red')

title('h_{ol}(n)=[h_1(n)+h_2(n)]*h_3(n)
')

grid on

```

```

% Ερώτημα b)

% Έχουμε βρει την hol(n) και την έχουμε
ορίσει στο nol=[0,40]

% Υπολογισμός της x(n), όπου
x(n)=n0.5^n με n=[0,10]

nx=[0:10];

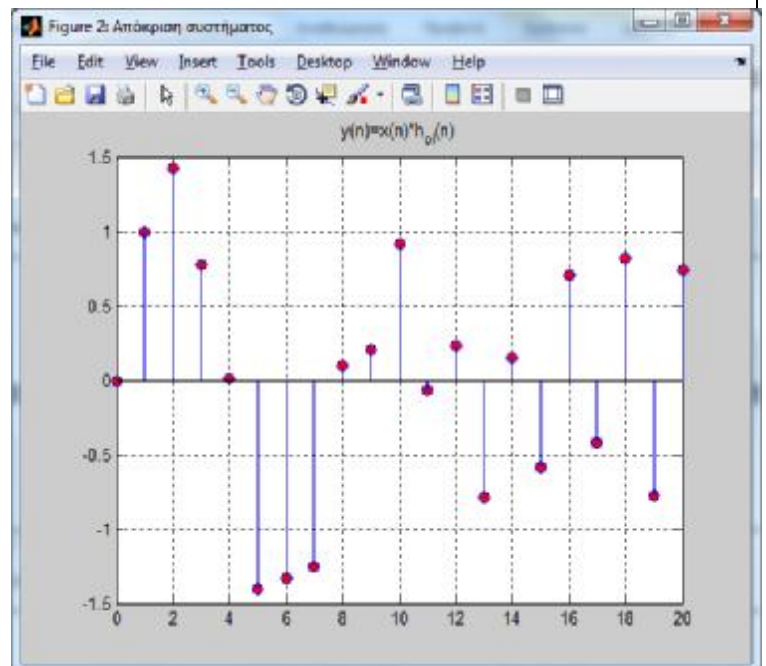
x=nx.*(0.5.^nx); % x(n)

% Για τη συνέλιξη των x(n) και hol(n)
θα ορίσουμε τα σήματα στο ίδιο διάστημα.
Επειδή η x(n) έχει n=[0,10], ενώ η hol(n)
έχει n=[0,40], μπορούμε να ορίσουμε ως
διάστημα το n=[0,40] και να συμπληρώσουμε
με 0 την x(n)

x=[x zeros(1,30)];

nf=0+0:40+40;

```



```

y=conv(x,hol);

y=y(nf>=0 & nf<=20); %Κρατάω για
n=[0,20]

%Γραφική παράσταση απόκρισης στο
n=[0,20]

figure('name','Απόκριση συστήματος')

stem(n,y,'MarkerFaceColor','red')

title('y(n)=x(n)*h_{ol}(n)'), grid on

```

3 Άλυτες Ασκήσεις

1. Να υπολογίσετε και να σχεδιάσετε την κρουστική απόκριση του συστήματος που περιγράφεται από την εξίσωση διαφορών
$$y(n) + 0.1y(n-1) - 0.2y(n-2) + 0.4y(n-4) = 0.5x(n) - 0.4x(n-1)$$
 για $n=[0, 100]$
2. Αν τροφοδοτήσουμε το προηγούμενο σύστημα με είσοδο $x(n) = \{-1 \ 0 \ 1 \ 2 \ 3 \ 4\}$ με $n=[-2, 3]$, να υπολογιστεί η έξοδος του συστήματος $y(n)$ για $n=[0, 100]$

4 Επίλογος – Σύνδεση με τα Επόμενα

Στην παρούσα εργαστηριακή άσκηση παρουσιάστηκαν οι εξισώσεις διαφορών ως εναλλακτικός τρόπος περιγραφής των συστημάτων. Ασχοληθήκαμε με τον υπολογισμό της εξόδου και της κρουστικής απόκρισης συστημάτων, όταν αυτά περιγράφονται μέσω γραμμικών εξισώσεων διαφορών με σταθερούς συντελεστές (ΓΕΔΣΣ).

Στην επόμενη εργαστηριακή άσκηση θα παρουσιαστεί ο μετασχηματισμός Fourier διακριτού χρόνου (DTFT).

5 Βιβλιογραφία

Hayes , Μ. Η. (2002). *Ψηφιακή Επεξεργασία Σήματος*. (Α. Δημητριάδης, Μεταφρ.) Θεσσαλονίκη: Τζιόλα.

Hsu, H. P. (2002). *Αναλογικές και Ψηφιακές Επικοινωνίες*. Θεσσαλονίκη: Τζιόλα.

Jayant, N. S., & Noll, P. (1984). *Digital Coding of Waveforms: Principles and Applications to Speech and Video*. Prentice Hall.

Mano, M. M., & Ciletti, M. D. (2012). *Digital Design (5 εκδ.)*. Pearson Education.

MathWorks - DSP. (2017). Ανάκτηση 2 10, 2017, από Getting Started with Matlab DSP: <https://www.mathworks.com/help/dsp/getting-started-with-dsp-system-toolbox.html>

Proakis, J. G., & Salehi, M. (2008). *Digital Communications (5 εκδ.)*. McGraw-Hill Higher Education.

Μουρτζόπουλος, Ι. (2012). *Ψηφιακή Τεχνολογία Ήχου*. Πάτρα: Πανεπιστήμιο Πατρών.

Εργαστηριακή Άσκηση 7: Μετασχηματισμός Fourier Διακριτού Χρόνου - DTFT

1 Εισαγωγικά Στοιχεία

1.1 Σκοπός

Σκοπός της παρούσας εργαστηριακής άσκησης είναι η εξοικείωση των φοιτητών – σπουδαστών με τον υπολογισμό του μετασχηματισμού Fourier Διακριτού Χρόνου (DTFT) σημάτων διακριτού χρόνου (ΣΔΧ), καθώς και του αντίστροφου μετασχηματισμού (I-DTFT).

1.2 Αναμενόμενα Μαθησιακά Αποτελέσματα

Με την ολοκλήρωση της παρούσας εργαστηριακής άσκησης, ο φοιτητής – σπουδαστής θα είναι σε θέση να:

- Υπολογίζει:
 - τον DTFT ενός ΣΔΧ
 - τον I-DTFT ενός ΣΔΧ
- Κατασκευάζει τη γραφική παράσταση του πλάτους και της φάσης του DTFT ενός ΣΔΧ

1.3 Εκτιμώμενη Διάρκεια

Ο εκτιμώμενος χρόνος που απαιτείται από τον φοιτητή – σπουδαστή για την ολοκλήρωση της παρούσας εργαστηριακής άσκησης είναι 2 διδακτικές ώρες.

1.4 Προαπαιτούμενα

Διάλεξη 5 της θεωρίας.

1.5 Εισαγωγή

Στην άσκηση αυτή θα ασχοληθούμε με τον DTFT και τον I-DTFT για ΣΔΧ. Ο DTFT αποτελεί βασικό εργαλείο της ψηφιακής επεξεργασίας σήματος αφού επιτρέπει την ανάλυση σημάτων στο πεδίο της συχνότητας. Επίσης, ο DTFT επιτρέπει τη μελέτη

της συμπεριφοράς ενός συστήματος στο πεδίο της συχνότητας, μέσω του υπολογισμού της απόκρισης συχνότητας του συστήματος.

1.6 Κατάλογος Χρήσιμων Συναρτήσεων Matlab

Κατά την εκπόνηση των ασκήσεων χρησιμοποιούνται οι ακόλουθες συναρτήσεις – εντολές του Matlab:

| Συνάρτηση | Επεξήγηση |
|-----------|--|
| length | Επιστρέφει το μήκος ενός πίνακα-διανύσματος |
| zeros | Δημιουργία πίνακα – διανύσματος με μηδενικά |
| linspace | Δημιουργεί διάνυσμα ισαπέχοντων τιμών |
| exp | Υπολογίζει τον εκθετικό όρο e^x |
| sum | Άθροισμα όρων |
| abs | Απόλυτη τιμή (μέτρο μιγαδικού) |
| angle | Υπολογίζει τη φάση (γωνία) μιγαδικού |
| strcat | Συνενώνει δύο αλφαριθμητικά |
| num2str | Μετατρέπει αριθμητικά δεδομένα σε αλφαριθμητικά |
| fft | Υπολογίζει τον DFT χρησιμοποιώντας τον αλγόριθμο γρήγορου μετασχηματισμού Fourier |
| fftshift | Μετατοπίζει το φάσμα ώστε οι μηδενικές συχνότητες να βρίσκονται στο κέντρο (χρήσιμο για την απεικόνιση του φάσματος) |
| ifft | Υπολογίζει τον αντίστροφο DFT χρησιμοποιώντας τον αλγόριθμο γρήγορου μετασχηματισμού Fourier |
| disp | Εμφανίζει ένα μήνυμα ή την τιμή μίας μεταβλητής |
| figure | Δημιουργεί νέο παράθυρο για την απεικόνιση γραφικών |

| | |
|---------|--|
| | παραστάσεων |
| plot | Δημιουργεί μία γραφική παράσταση |
| subplot | Τμηματοποίηση ενός παραθύρου για γραφικές παραστάσεις σε γραμμές και στήλες, προκειμένου στο ίδιο παράθυρο να απεικονιστούν πολλαπλές γραφικές παραστάσεις |
| stem | Γραφική παράσταση διακριτών σημάτων – ακολουθιών |
| title | Τοποθετεί τίτλο σε μια γραφική παράσταση |
| axis | Ορίζει τα όρια των αξόνων και άλλες ιδιότητες των αξόνων σε μια γραφική παράσταση |
| grid | Προσθέτει – αφαιρεί πλέγμα στις γραφικές παραστάσεις, βελτιώνοντας την ανάγνωσή τους |
| xlabel | Τοποθετεί τίτλο στον οριζόντιο άξονα μίας γραφικής παράστασης |

2 Μετασχηματισμοί DTFT και I-DTFT

Ο μετασχηματισμός Fourier Διακριτού Χρόνου (DTFT) αποτελεί βασικό εργαλείο για την ανάλυση των μη περιοδικών διακριτών σημάτων στο πεδίο της συχνότητας. Η εφαρμογή του DTFT σε ένα μη περιοδικό διακριτό σήμα στο πεδίο του χρόνου, οδηγεί στη μετατροπή του σε περιοδικό και συνεχές σήμα στο πεδίο της συχνότητας. Επίσης, το βασικό πλεονέκτημα του DTFT είναι η ιδιότητά του να μετατρέπει τις συνελίξεις (πεδίο του χρόνου) σε πολλαπλασιασμούς (στο πεδίο της συχνότητας).

Ο DTFT ενός διακριτού σήματος $x(n)$, συμβολίζεται $X(e^{j\omega})$ και υπολογίζεται από τη σχέση:

$$X(e^{j\omega}) = \sum_{n=-\infty}^{+\infty} x(n)e^{-j\omega n}$$

Για να ορίζεται ο DTFT ενός διακριτού σήματος $x(n)$, θα πρέπει η απόλυτη τιμή του $x(n)$ να συγκλίνει απόλυτα, δηλαδή να ισχύει ότι:

$$\sum_{n=-\infty}^{+\infty} |x(n)| = S < \infty$$

Επειδή ο DTFT $X(e^{j\omega})$ ενός διακριτού σήματος $x(n)$ είναι μιγαδική συνάρτηση, η οποία εξαρτάται από την πραγματική μεταβλητή « ω » (κυκλική συχνότητα), μπορεί να γραφτεί συναρτήσει του πραγματικού και του φανταστικού μέρους του $X(e^{j\omega})$, ως εξής:

$$X(e^{j\omega}) = X_{re}(e^{j\omega}) + jX_{im}(e^{j\omega})$$

όπου: $X_{re}(e^{j\omega})$ και $X_{im}(e^{j\omega}) = \text{πραγματικό και φανταστικό μέρος αντίστοιχα.}$

Εναλλακτικά, μπορεί να γραφτεί και στην εξής μορφή:

$$X(e^{j\omega}) = |X(e^{j\omega})|e^{j\theta(\omega)}$$

$$\text{όπου: } \theta(\omega) = \arg\{X(e^{j\omega})\} = \tan^{-1} \left\{ \frac{X_{im}(e^{j\omega})}{X_{re}(e^{j\omega})} \right\}$$

Η ποσότητα $|X(e^{j\omega})|$ λέγεται συνάρτηση πλάτους ή πλάτος φάσματος, ενώ η ποσότητα $\theta(\omega)$ λέγεται συνάρτηση φάσης ή φάση φάσματος.

Ο DTFT $X(e^{j\omega})$ είναι περιοδική και συνεχής συνάρτηση της κυκλικής συχνότητας « ω » με περίοδο 2π . Για την αναπαράστασή του συνηθίζεται να χρησιμοποιούνται οι συχνότητες στο διάστημα $[-\pi, \pi]$.

Επίσης, όταν γνωρίζουμε τον DTFT $X(e^{j\omega})$ ενός διακριτού σήματος $x(n)$ και επιθυμούμε να ανακατασκευάσουμε το αρχικό σήμα $x(n)$, τότε μπορούμε να υπολογίσουμε τον I-DTFT της $X(e^{j\omega})$, ο οποίος δίνεται από τη σχέση:

$$x(n) = \frac{1}{2\pi} \int_{-\pi}^{\pi} X(e^{j\omega}) e^{j\omega n} d\omega$$

Οι πράξεις του ευθύ DTFT και του αντίστροφου DTFT (I-DTFT) συνηθίζεται για συντομία να συμβολίζονται αντίστοιχα ως εξής:

$$x(n) \xleftrightarrow{DTFT} X(e^{j\omega})$$

$$X(e^{j\omega}) \xleftrightarrow{I-DTFT} x(n)$$

Άσκηση 1^η

Έστω το σήμα $x(n)=\alpha^n$, όπου $\alpha=0.8$, ορισμένο στο διάστημα $n=[0,20]$. Να υπολογίσετε και να σχεδιάσετε τον DTFT της $x(n)$ στο διάστημα συχνοτήτων:

- a) $\omega=[-\pi, \pi]$ και
- b) $\omega=[-5\pi, 5\pi]$.

Και στις δύο περιπτώσεις, για τη συχνότητα « ω » να ορίσετε διάνυσμα $L=512$ ισαπέχουσων συχνοτήτων. Επίσης, για τον υπολογισμό του DTFT να χρησιμοποιήσετε τη σχέση ορισμού, δηλαδή ότι: $X(e^{j\omega}) = \sum_{n=-\infty}^{+\infty} x(n)e^{-j\omega n}$

| Κώδικας Matlab | Αποτέλεσμα |
|----------------|------------|
|----------------|------------|

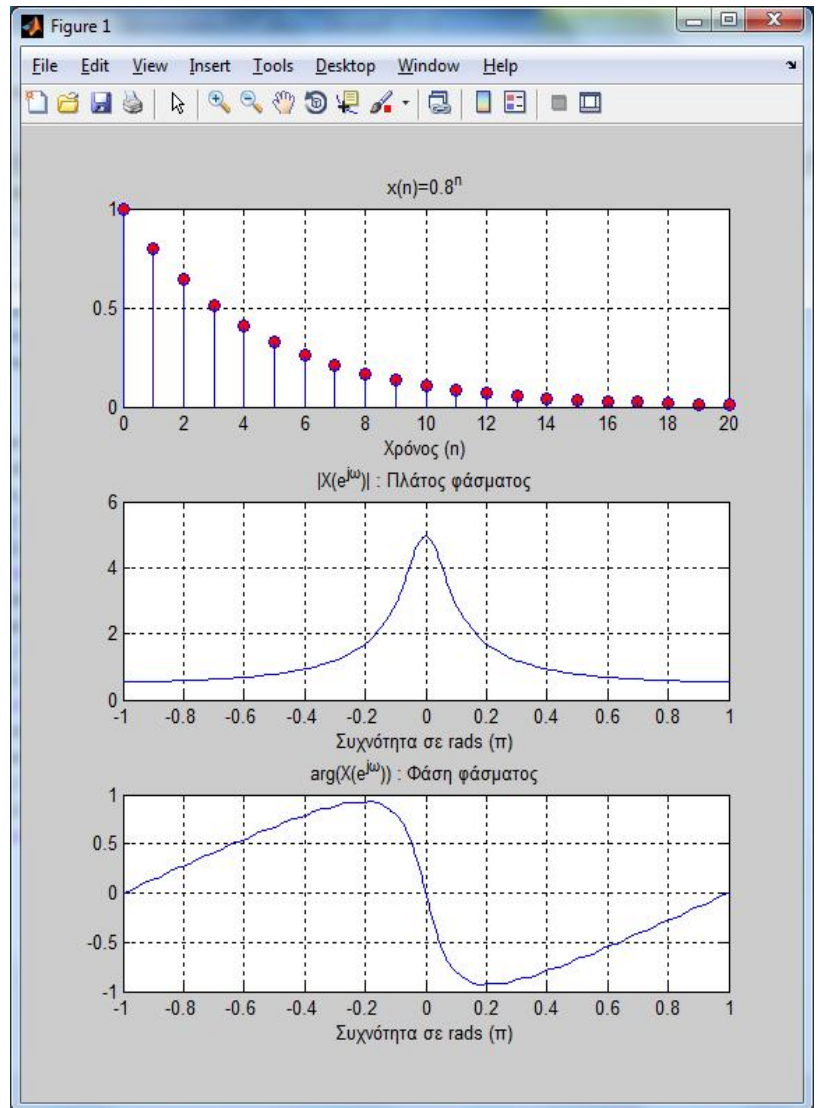
A)

```
n=0:20; x=0.8.^n;
w=linspace(-pi,pi,512);
X=zeros(1,length(w));
for k=1:length(w)
    X(k)=sum(x.*exp(-j*w(k).*n));
end

figure(1)
subplot(3,1,1)
stem(n,x,'MarkerFaceColor','red')
title('x(n)=0.8^n');
xlabel('Χρόνος (n)'), grid on

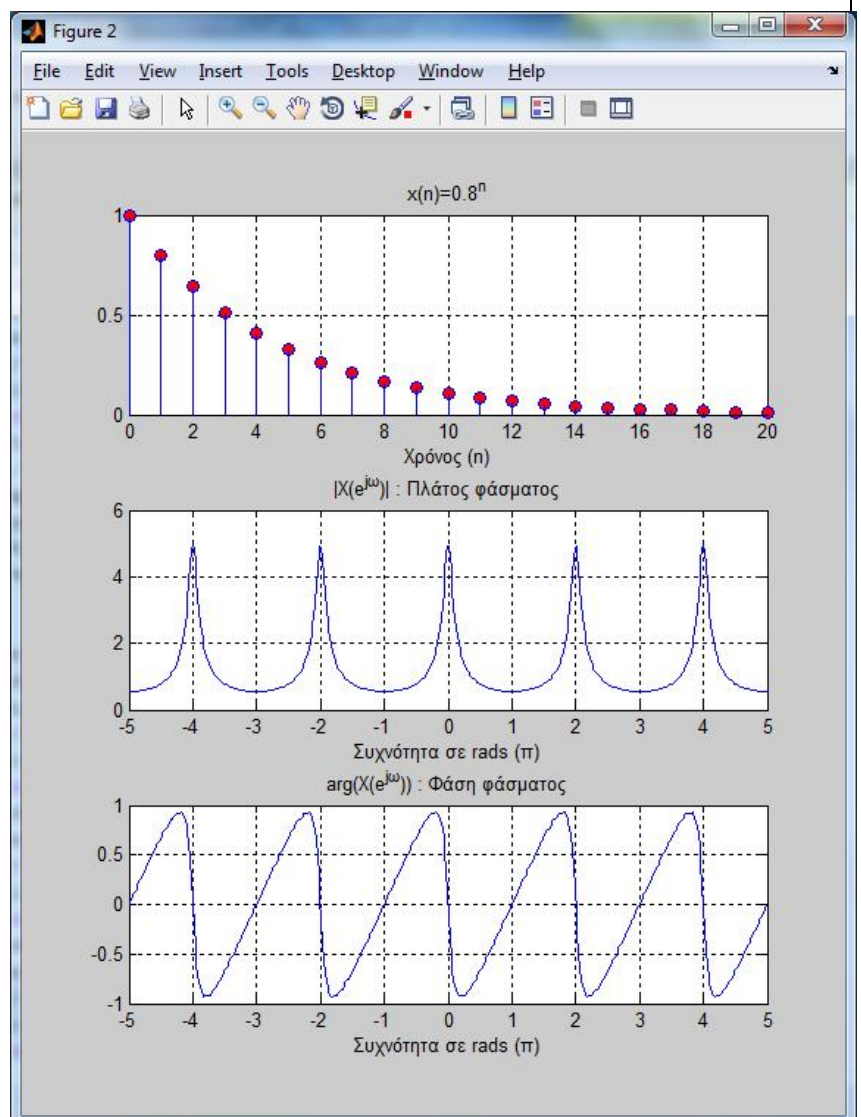
subplot(3,1,2)
plot(w/pi,abs(X))
title('|X(e^{j\omega})| : Πλάτος φάσματος');
xlabel('Συχνότητα σε rads (\pi)')
grid on

subplot(3,1,3)
plot(w/pi,angle(X))
title('arg(X(e^{j\omega})) : Φάση φάσματος');
xlabel('Συχνότητα σε rads (\pi)')
grid on
```



B)

```
n=0:20; x=0.8.^n;  
w=linspace(-5*pi,5*pi,512);  
X=zeros(1,length(w));  
for k=1:length(w)  
    X(k)=sum(x.*exp(-j*w(k).*n));  
end  
  
figure(2)  
subplot(3,1,1)  
stem(n,x,'MarkerFaceColor','red')  
title('x(n)=0.8^n');  
xlabel('Χρόνος (n)'), grid on  
subplot(3,1,2)  
plot(w/pi,abs(X))  
title('|X(e^{j\omega})| : Πλάτος  
φάσματος');  
xlabel('Συχνότητα σε rads (\pi)')  
grid on  
subplot(3,1,3)  
plot(w/pi,angle(X))  
title('arg(X(e^{j\omega})) : Φάση  
φάσματος');  
xlabel('Συχνότητα σε rads (\pi)')  
grid on
```



Παρατήρηση:

Στο τελευταίο σύνθετο γράφημα είναι εμφανές ότι τόσο το πλάτος όσο και η φάση του DTFT είναι περιοδικές συναρτήσεις με περίοδο 2π .

Άσκηση 2^η

Έστω το σήμα $x(n) = \begin{cases} 1, & \text{για } n = [-10, 10] \\ 0, & \text{αλλού} \end{cases}$, ορισμένο στο διάστημα $n = [-50, 50]$.

Να υπολογιστεί και να σχεδιαστεί ο DTFT στο διάστημα συχνοτήτων $\omega = [-\pi, \pi]$, όπου για τη συχνότητα « ω » να ορίσετε διάνυσμα ισαπέχουσων συχνοτήτων μήκους:

- a) $L=50$
- b) $L=101$ (δηλαδή, όσο είναι το μήκος του σήματος στο πεδίο του χρόνου)
- c) $L=256$ και
- d) $L=512$.

Για τον υπολογισμό του DTFT να χρησιμοποιήσετε τη σχέση ορισμού, δηλαδή ότι:

$$X(e^{j\omega}) = \sum_{n=-\infty}^{+\infty} x(n)e^{-j\omega n}.$$

Επίσης, να κατασκευάσετε ξεχωριστά συγκεντρωτικά γραφήματα για τα πλάτη και τις φάσεις του DTFT για τις διαφορετικές τιμές του L . Τι παρατηρείτε;

| Κώδικας Matlab | Αποτέλεσμα |
|----------------|------------|
|----------------|------------|

```

% Δημιουργία σήματος x(n)

n=[-50:50];

x=zeros(1,length(n));

x(n>=-10 & n<=10)=1;

% Χρήση L=50 συχνοτήτων

L1=50;

w1=linspace(-pi,pi,L1);

X1=zeros(1,length(w1));

for k=1:length(w1)

    X1(k)=sum(x.*exp(-j*w1(k).*n));

end

s=strcat('DTFT L=',num2str(L1),'
συχνότητες');

figure('name',s)

subplot(3,1,1)

stem(n,x,'MarkerFaceColor','red')

title('x(n)=1, n=[-10, 10]');

xlabel('Χρόνος (n)')

grid on

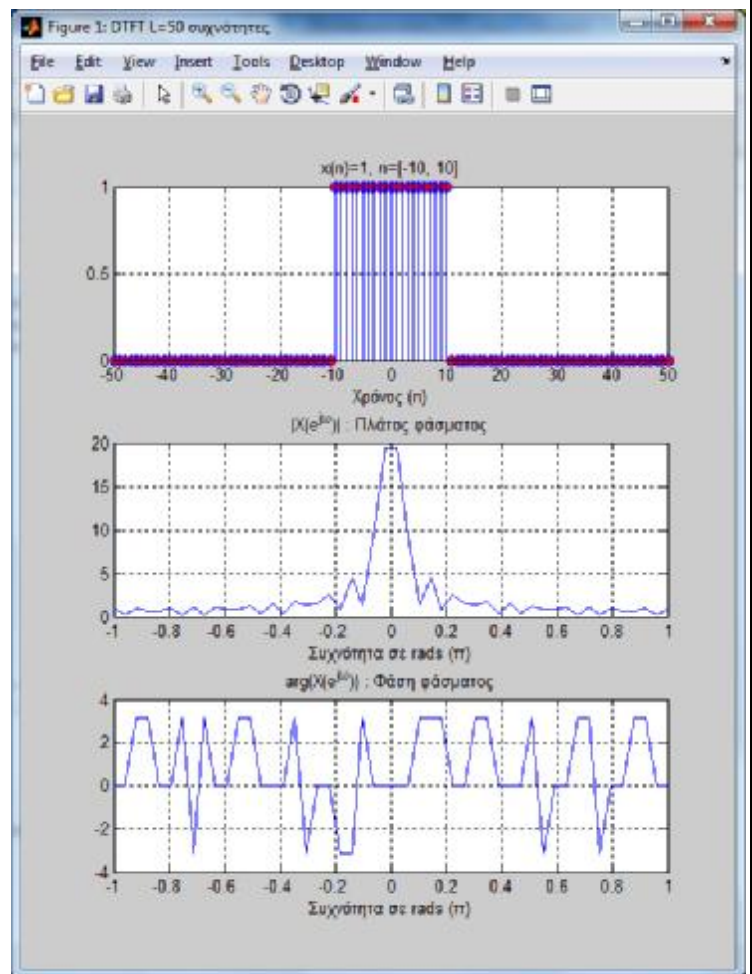
subplot(3,1,2)

plot(w1/pi,abs(X1))

title('|X(e^{j\omega})| : Πλάτος
φάσματος');

xlabel('Συχνότητα σε rads (\pi)')

```



```

grid on

subplot(3,1,3)

plot(w1/pi,angle(X1))

title('arg(X(e^{j\omega})) : Φάση
φάσματος');

xlabel('Συχνότητα σε rads (π)')

grid on

% Χρήση L=101 συχνοτήτων
L2=101;
w2=linspace(-pi,pi,L2);
X2=zeros(1,length(w2));

for k=1:length(w2)

    X2(k)=sum(x.*exp(-j*w2(k).*n));

end

s=strcat('DTFT L=',num2str(L2),'
συχνότητες');

figure('name',s)

subplot(3,1,1)

stem(n,x,'MarkerFaceColor','red')

title('x(n)=1, n=[-10, 10]');

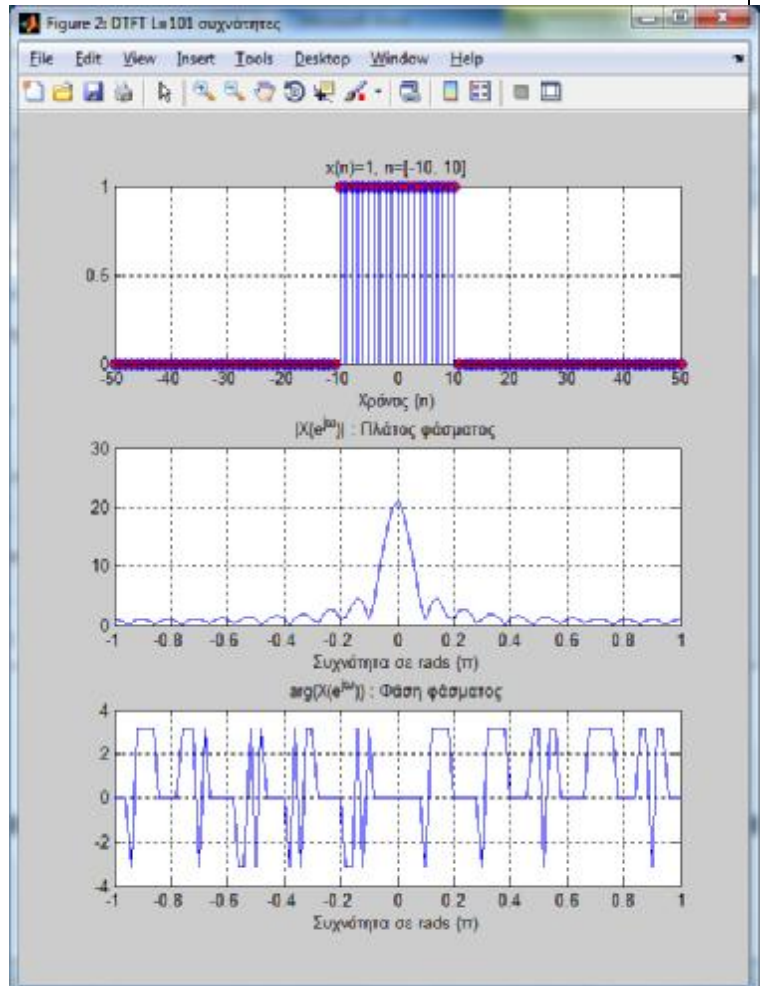
xlabel('Χρόνος (n)')

grid on

subplot(3,1,2)

plot(w2/pi,abs(X2))

```



```

title('|X(e^{j\omega})| : Πλάτος
φάσματος');

xlabel('Συχνότητα σε rads (π)')

grid on

subplot(3,1,3)

plot(w2/pi,angle(X2))

title('arg(X(e^{j\omega})) : Φάση
φάσματος');

xlabel('Συχνότητα σε rads (π)')

grid on

% Χρήση L=256 συχνοτήτων
L3=256;

w3=linspace(-pi,pi,L3);

X3=zeros(1,length(w3));

for k=1:length(w3)

    X3(k)=sum(x.*exp(-j*w3(k).*n));

end

s=strcat('DTFT L=',num2str(L3),'
συχνότητες');

figure('name',s)

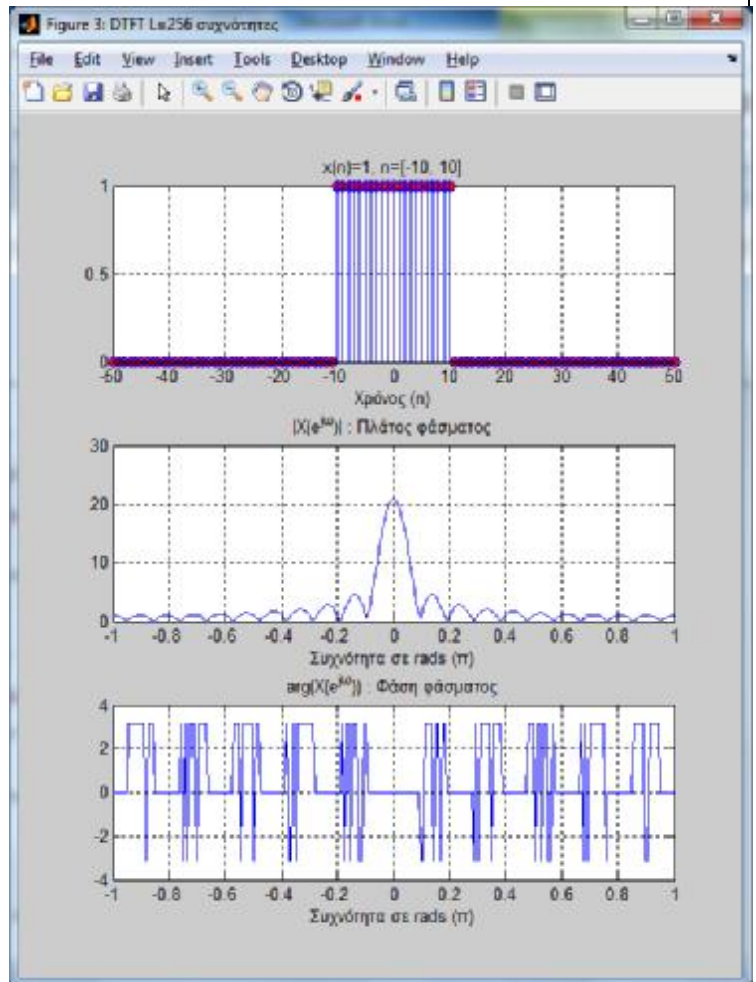
subplot(3,1,1)

stem(n,x,'MarkerFaceColor','red')

title('x(n)=1, n=[-10, 10]');

xlabel('Χρόνος (n)')

```



```

grid on

subplot(3,1,2)

plot(w3/pi,abs(X3))

title('|X(e^{j\omega})| : Πλάτος
φάσματος');

xlabel('Συχνότητα σε rads (π)')

grid on

subplot(3,1,3)

plot(w3/pi,angle(X3))

title('arg(X(e^{j\omega})) : Φάση
φάσματος');

xlabel('Συχνότητα σε rads (π)')

grid on

% Χρήση L=512 συχνοτήτων

L4=512;

w4=linspace(-pi,pi,L4);

X4=zeros(1,length(w4));

for k=1:length(w4)

    X4(k)=sum(x.*exp(-j*w4(k).*n));

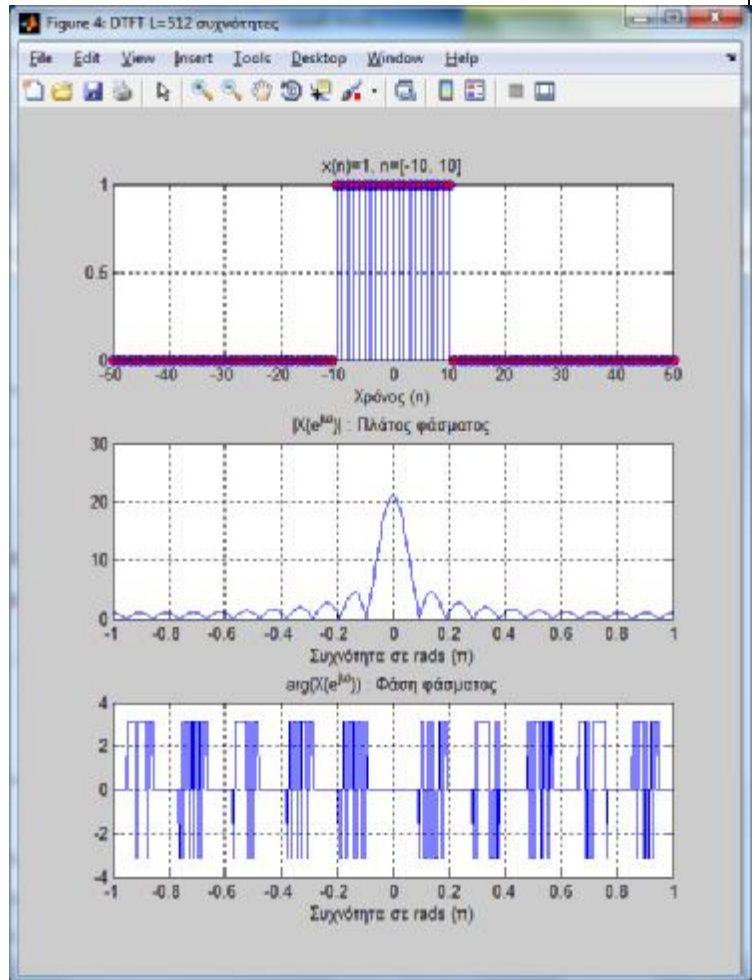
end

s=strcat('DTFT L=',num2str(L4),'
συχνότητες');

figure('name',s)

subplot(3,1,1)

```



```

stem(n,x,'MarkerFaceColor','red')

title('x(n)=1, n=[-10, 10]');

xlabel('Χρόνος (n)')

grid on

subplot(3,1,2)

plot(w4/pi,abs(X4))

title('|X(e^{j\omega})| : Πλάτος
φάσματος');

xlabel('Συχνότητα σε rads (\pi)')

grid on

subplot(3,1,3)

plot(w4/pi,angle(X4))

title('arg(X(e^{j\omega})) : Φάση
φάσματος');

xlabel('Συχνότητα σε rads (\pi)')

grid on

%Κατασκευή συγκεντρωτικών
γραφημάτων

figure('name','Σύγκριση πλατών DTFT
για L=50, 101, 256, 512 συχνοτητες')

subplot(4,1,1)

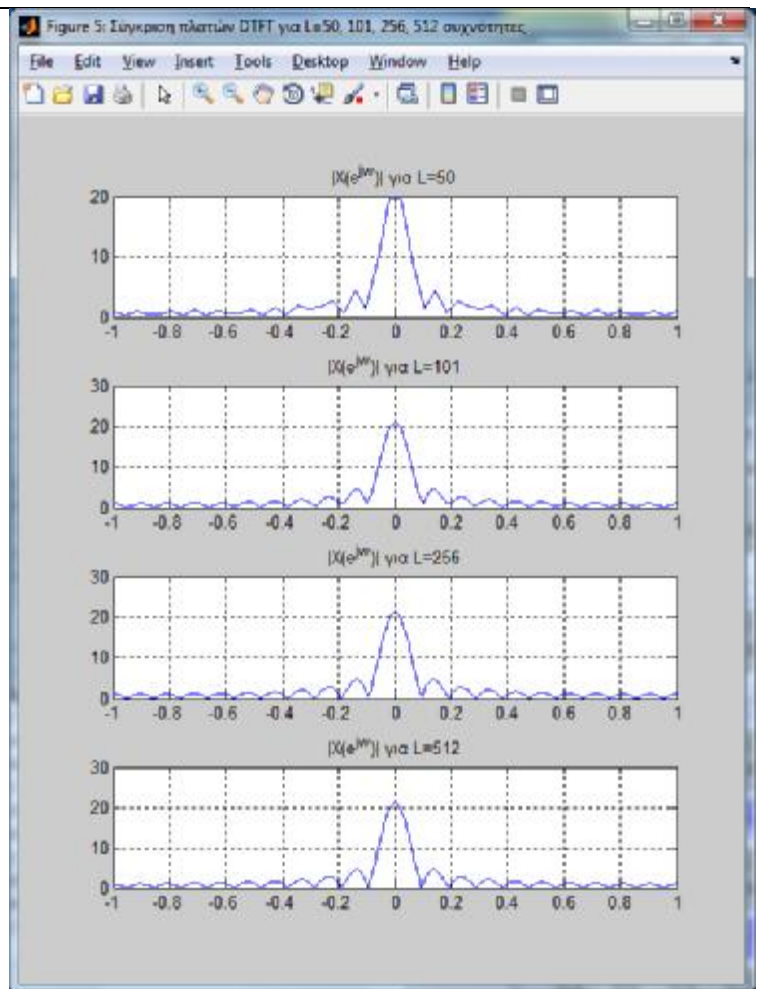
plot(w1/pi,abs(X1)); grid on;

title('|X(e^{j\omega})| για L=50')

subplot(4,1,2)

plot(w2/pi,abs(X2)); grid on;

```

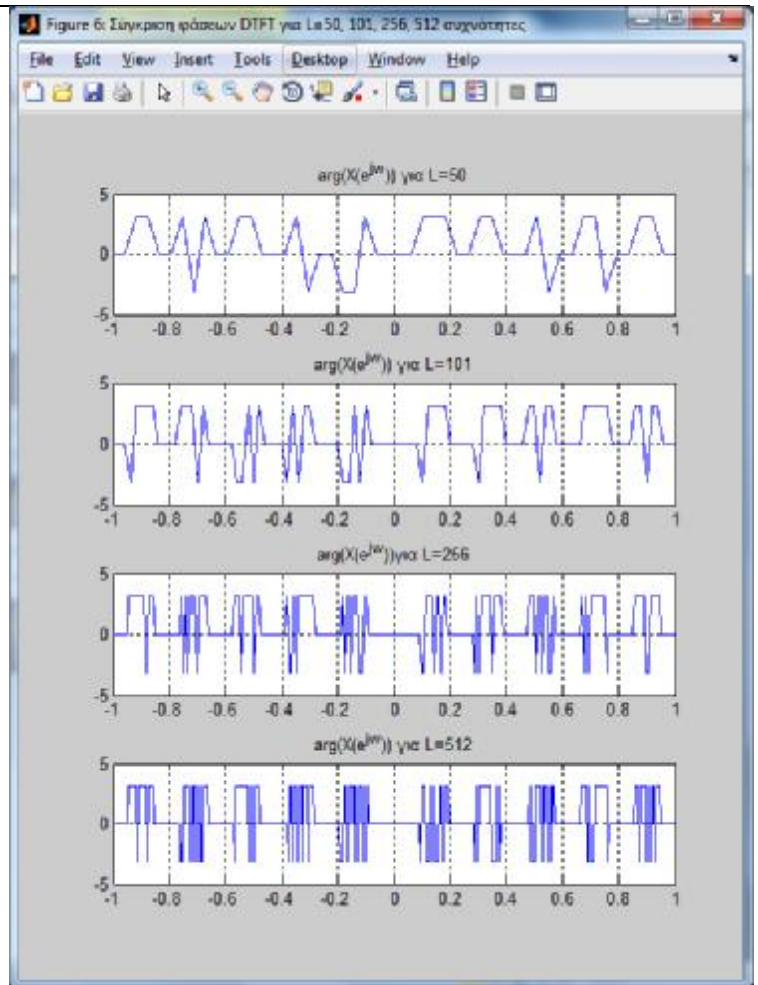


```

title('|X(e^{jw})| για L=101')
subplot(4,1,3)
plot(w3/pi, abs(X3)); grid on;
title('|X(e^{jw})| για L=256')
subplot(4,1,4)
plot(w4/pi, abs(X4)); grid on;
title('|X(e^{jw})| για L=512')

figure('name','Σύγκριση φάσεων DTFT
για L=50, 101, 256, 512 συχνότητες')
subplot(4,1,1)
plot(w1/pi, angle(X1)); grid on;
title('arg(X(e^{jw})) για L=50')
subplot(4,1,2)
plot(w2/pi, angle(X2)); grid on;
title('arg(X(e^{jw})) για L=101')
subplot(4,1,3)
plot(w3/pi, angle(X3)); grid on;
title('arg(X(e^{jw})) για L=256')
subplot(4,1,4)
plot(w4/pi, angle(X4)); grid on;
title('arg(X(e^{jw})) για L=512')

```



Παρατήρηση:

Από τα γραφήματα φαίνεται ότι αυξάνοντας τη συχνοτική ανάλυση, δηλαδή το πλήθος των συχνοτήτων που χρησιμοποιούμε για τον υπολογισμό του DTFT, βελτιώνεται η λεπτομέρεια απεικόνισης του DTFT.

Ο κανόνας είναι ότι το πλήθος L των συχνοτήτων θα πρέπει να είναι τουλάχιστον ίσο με το μήκος του σήματος.

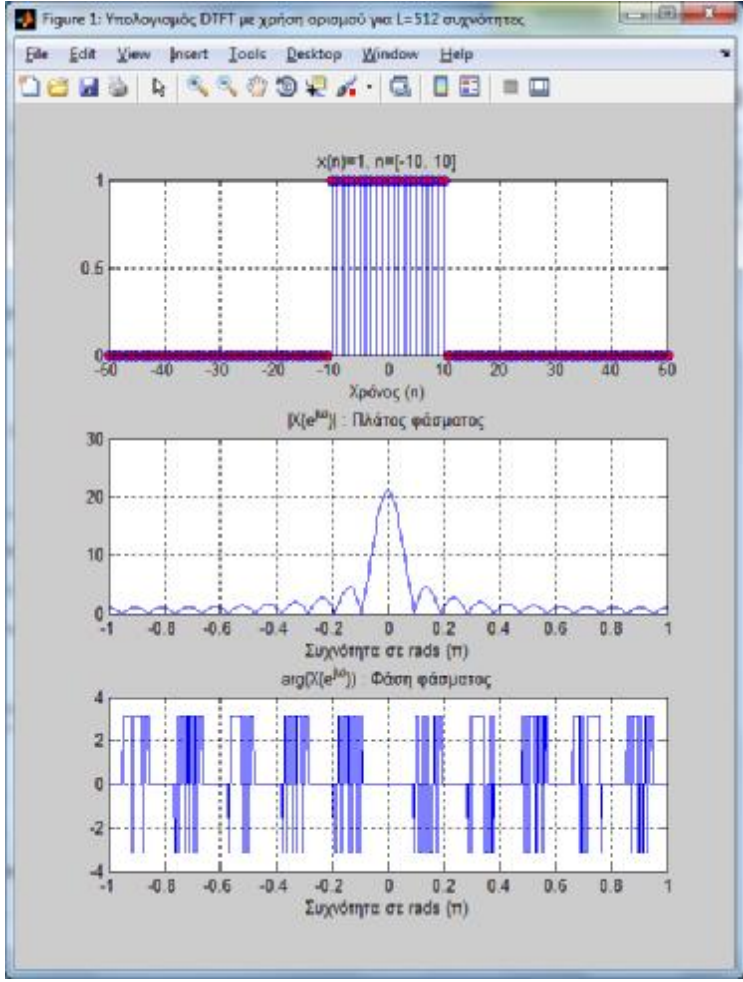
Άσκηση 3^η

Έστω το σήμα $x(n) = \begin{cases} 1, & \text{για } n = [-10, 10] \\ 0, & \text{αλλού} \end{cases}$, ορισμένο στο διάστημα $n = [-50, 50]$.

Να υπολογιστεί και να σχεδιαστεί ο DTFT στο διάστημα συχνοτήτων $\omega = [-\pi, \pi]$, όπου για τη συχνότητα « ω » να ορίσετε δάνυσμα ισαπέχουσων συχνοτήτων μήκους $L=512$. Για τον υπολογισμό του DTFT να χρησιμοποιήσετε

- τη σχέση ορισμού, δηλαδή ότι: $X(e^{j\omega}) = \sum_{n=-\infty}^{+\infty} x(n)e^{-j\omega n}$
- τη συνάρτηση fft του Matlab.

Να συγκρίνετε τα αποτελέσματα εμφανίζοντάς τα σε ένα κοινό γράφημα.

| Κώδικας Matlab | Αποτέλεσμα |
|---|---|
| <pre> % Δημιουργία σήματος x(n) n=[-50:50]; x=zeros(1,length(n)); x(n>=-10 & n<=10)=1; % Υπολογισμός DTFT από σχέση ορισμού L=512; w=linspace(-pi,pi,L); X1=zeros(1,length(w)); for k=1:length(w) X1(k)=sum(x.*exp(-j*w(k).*n)); end s=strcat('Υπολογισμός DTFT με χρήση ορισμού για L=',num2str(L),' συχνότητες'); figure('name', s) </pre> |  |


```

subplot(3,1,1)

stem(n,x,'MarkerFaceColor','red')

title('x(n)=1, n=[-10, 10]');

xlabel('Χρόνος (n)')

grid on

subplot(3,1,2)

plot(w/pi, abs(X1))

title('|X(e^{j\omega})| : Πλάτος φάσματος');

xlabel('Συχνότητα σε rads (\pi)')

grid on

subplot(3,1,3)

plot(w/pi, angle(X1))

title('arg(X(e^{j\omega})) : Φάση φάσματος');

xlabel('Συχνότητα σε rads (\pi)')

grid on

% Υπολογισμός DTFT μέσω fft

X2=fft(x,L);

% Ολίσθηση από το [0,2\pi] στο [-\pi,\pi]

X2_shifted=fftshift(X2);

s=strcat('Υπολογισμός DTFT με χρήση fft για L=',num2str(L),' συχνότητες');

figure('name', s)

```

```

subplot(3,1,1)

stem(n,x,'MarkerFaceColor','red')

title('x(n)=1, n=[-10, 10]');

xlabel('Χρόνος (n)')

grid on

subplot(3,1,2)

plot(w/pi, abs(X2_shifted))

title('|X(e^{j\omega})| : Πλάτος φάσματος');

xlabel('Συχνότητα σε rads (\pi)')

grid on

subplot(3,1,3)

plot(w/pi, angle(X2_shifted))

title('arg(X(e^{j\omega})) : Φάση φάσματος');

xlabel('Συχνότητα σε rads (\pi)')

grid on

%Κατασκευή συγκεντρωτικών
γραφημάτων

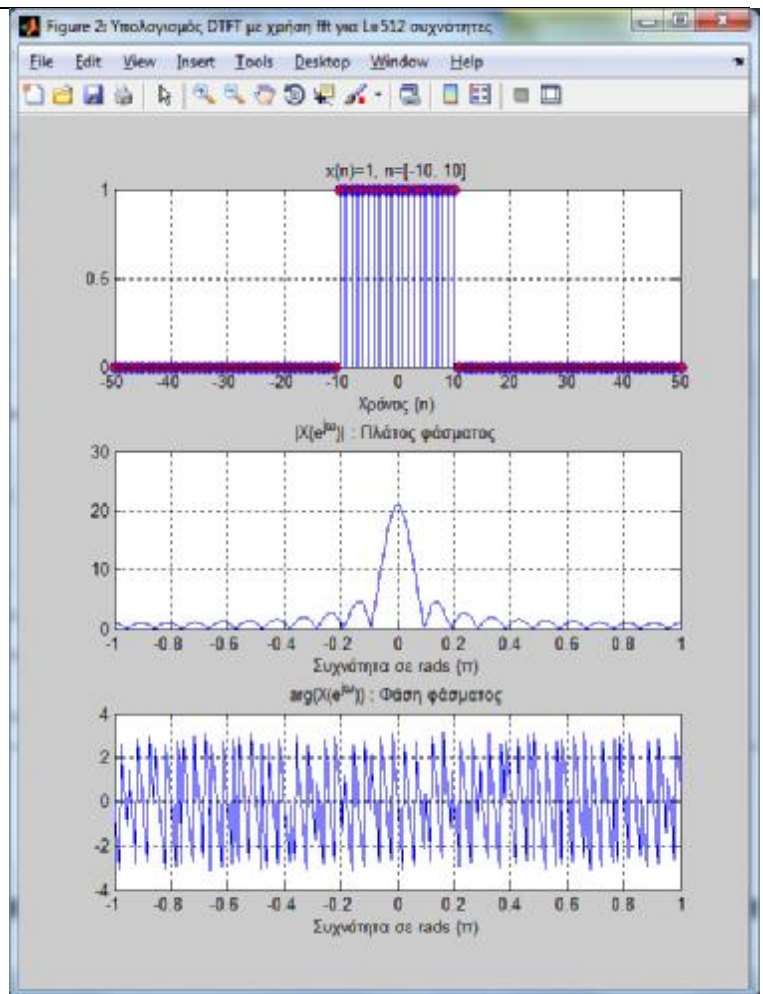
figure('name','Σύγκριση υπολογισμού
DTFT από σχέση ορισμού και fft')

subplot(2,2,1)

plot(w/pi, abs(X1)); grid on;

title('|X(e^{j\omega})| (σχέση
ορισμού)')

```



```

xlabel('Συχνότητα σε rads (π)')

subplot(2,2,2)

plot(w/pi, abs(X2_shifted));grid
on;

title('|X(e^{jw})| (fft)')

xlabel('Συχνότητα σε rads (π)')

subplot(2,2,3)

plot(w/pi, angle(X1)); grid on;

title('arg(X(e^{jw})) (σχέση
ορισμού)')

xlabel('Συχνότητα σε rads (π)')

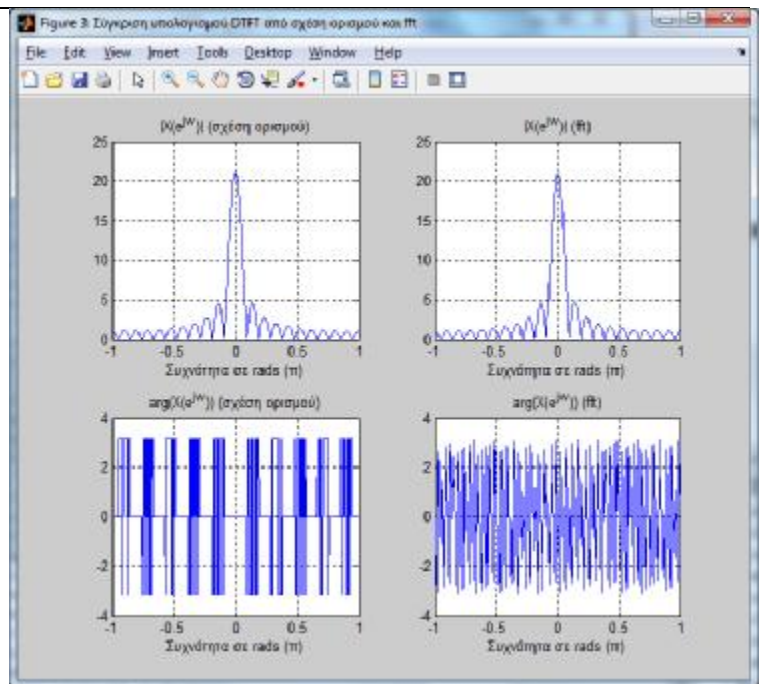
subplot(2,2,4)

plot(w/pi, angle(X2_shifted));
grid on;

title('arg(X(e^{jw})) (fft)')

xlabel('Συχνότητα σε rads (π)')

```



Παρατηρήσεις:

Η συνάρτηση `fft` χρησιμοποιείται για τον γρήγορο υπολογισμό του μετασχηματισμού DFT ενός διακριτού σήματος. Ωστόσο, επειδή ο DFT αποτελεί δειγματοληψία του DTFT, μπορούμε να υπολογίσουμε τον DTFT με αρκετά καλή προσέγγιση μέσω του DFT και της συνάρτησης `fft` του Matlab.

Από τα γραφήματα φαίνεται ότι η συνάρτηση του Matlab `fft` παράγει το ίδιο αποτέλεσμα για το πλάτος του DTFT, με αυτό που προκύπτει όταν υπολογίζουμε τον DTFT με χρήση του ορισμού.

Ωστόσο, επειδή ο `fft` παράγει το αποτέλεσμα στο $[0, 2\pi]$, απαιτείται η μετατόπιση των τιμών στο $[-\pi, \pi]$. Αυτό επιτυγχάνεται με την εντολή `fftshift`.

| | |
|--|---|
| | Ωστόσο, η φάση φαίνεται να διαφέρει. Αυτό οφείλεται στον προσεγγιστικό τρόπο υπολογισμού που χρησιμοποιείται από τον fft. |
|--|---|

Άσκηση 4^η

Να κατασκευάσετε συνάρτηση που θα υπολογίζει και θα σχεδιάζει τον DTFT ενός διακριτού σήματος $x(n)$ χρησιμοποιώντας τον ορισμό του DTFT. Η συνάρτηση θα πρέπει να **δέχεται ως εισόδους**:

- Τα πλάτη (τιμές) του σήματος $x(n)$
- Το πλήθος L των συχνοτήτων, που θα χρησιμοποιηθούν για τον υπολογισμό του DTFT. Θεωρήστε ως διάστημα συχνοτήτων το $[-\pi, \pi]$.

Επιπλέον, η συνάρτηση θα πρέπει να **επιστρέφει ως εξόδους**:

- Τις τιμές του DTFT $X(e^{j\omega})$
- Τις συχνότητες κατάτμησης του διαστήματος $[-\pi, \pi]$.

Τέλος, η συνάρτηση θα εμφανίζει γράφημα με το πλάτος και τη φάση του φάσματος του σήματος.

Κώδικας Matlab

```
function [X w] = DTFT(x,n,L)
% Υπολογισμός DTFT από εξίσωση ορισμού στο [-π,π]
% Είσοδοι:
% x=πλάτη σήματος εισόδου
% n=θέσεις σήματος εισόδου
% L=πλήθος συχνοτήτων
% Εξοδοι:
% X=ο DTFT του σήματος x
% w=συχνότητες που χρησιμοποιήθηκαν
```

```

w=[];
X=[];

if (length(x)~=length(n))
    disp('Δεν ορίσατε σωστά το σήμα x ή τις θέσεις n')
    disp('Πρέπει μήκος σήματος=πλήθος θέσεων')
    s=strcat('Μήκος(x)=',num2str(length(x)),',',
Μήκος(n)=',num2str(length(n)));
    disp(s)
    return
elseif (L<length(x))
    disp('Δεν ορίσατε σωστά το πλήθος συχνοτήτων L')
    disp('Πρέπει L>=μήκος σήματος')
    s=strcat('L=',num2str(L),', Μήκος(x)=',num2str(length(x)));
    disp(s)
    return
else
    disp('DTFT calculations...')
    w=linspace(-pi,pi,L);
    X=zeros(1,length(w));
    for k=1:L
        X(k)=sum(x.*exp(-j*w(k).*n));
    end
    disp('DONE')
end

% Γραφική απεικόνιση φάσματος (DTFT) του σήματος x
s=strcat('Υπολογισμός DTFT με χρήση ορισμού για L=',num2str(L),'
συχνότητες');
figure('name', s)
subplot(3,1,1)
stem(n,x,'MarkerFaceColor','red')
title('x(n)');
xlabel('Χρόνος (n)')
grid on
subplot(3,1,2)
plot(w/pi, abs(X))

```

```

title('|X(e^{j\omega})| : Πλάτος φάσματος');
xlabel('Συχνότητα σε rads (π)')
grid on
subplot(3,1,3)
plot(w/pi, angle(X))
title('arg(X(e^{j\omega})) : Φάση φάσματος');
xlabel('Συχνότητα σε rads (π)')
grid on

end

```

Άσκηση 5^η

Έστω το σήμα $x(n) = \begin{cases} 1, & \text{για } n = [-10, 10] \\ 0, & \text{αλλού} \end{cases}$, ορισμένο στο διάστημα $n = [-50, 50]$,

το οποίο χρησιμοποιήσαμε και στην 3^η άσκηση.

- Να υπολογιστεί ο DTFT του σήματος με χρήση της συνάρτησης `fft` του Matlab και στη συνέχεια να σχεδιαστεί στο διάστημα συχνοτήτων $\omega = [-\pi, \pi]$, χρησιμοποιώντας ως πλήθος συχνοτήτων το μήκος του σήματος $x(n)$.
- Στη συνέχεια χρησιμοποιώντας τη συνάρτηση `ifft` του Matlab, να ανακατασκευάσετε το αρχικό σήμα $x(n)$, εκτελώντας τον αντίστροφο μετασχηματισμό Fourier διακριτού χρόνου (I-DTFT) και να σχεδιάσετε το ανακατασκευασμένο σήμα. Τι παρατηρείτε;

| Κώδικας Matlab | Αποτέλεσμα |
|----------------|------------|
|----------------|------------|

```

% Δημιουργία σήματος x(n)

n=[-50:50];

x=zeros(1,length(n));

x(n>=-10 & n<=10)=1;

L=length(n);

w=linspace(-pi,pi,L);

X=fft(x,L);

figure('name', 'DTFT')

subplot(2,1,1)

stem(n,x,'MarkerFaceColor','red')

title('x(n)');

xlabel('Χρόνος (n)');grid on

subplot(2,1,2)

plot(w/pi, fftshift(abs(X)))

title('|X(e^{j\omega})| : Πλάτος φάσματος'); grid on

xlabel('Συχνότητα σε rads (\pi)')

% Μετασχηματισμός I-DTFT

x_original=ifft(X,L);

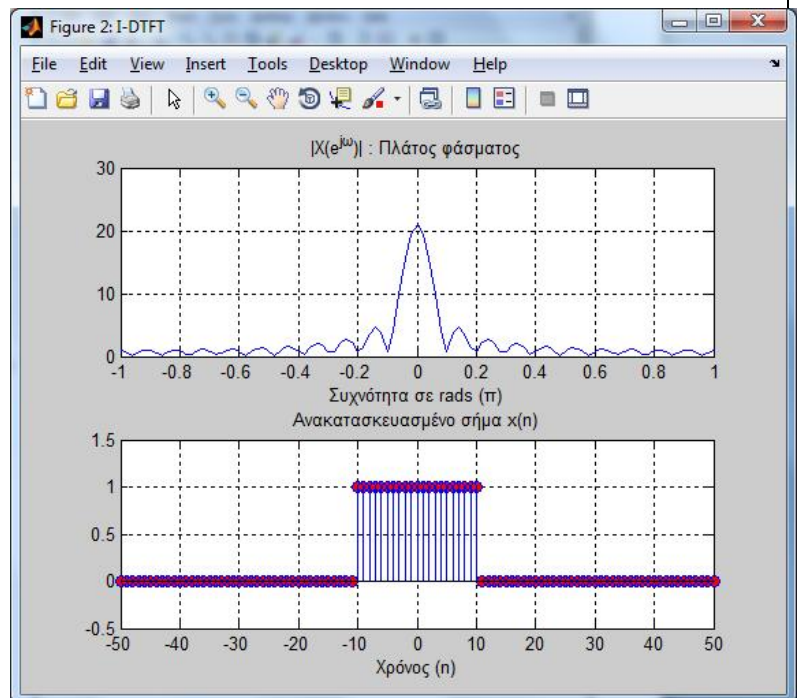
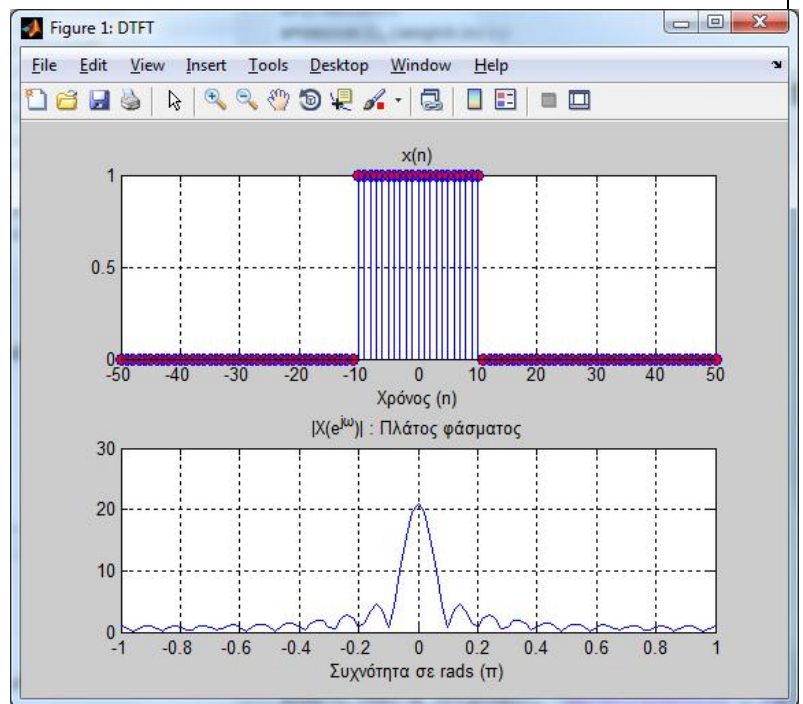
figure('name', 'I-DTFT')

subplot(2,1,1)

plot(w/pi, fftshift(abs(X)))

title('|X(e^{j\omega})| : Πλάτος φάσματος'); grid on

```



Παρατηρήσεις:

Παρατηρούμε ότι μέσω της συνάρτησης `ifft`, η οποία εκτελεί τον αντίστροφο μετασχηματισμό, επιτυγχάνουμε

| | |
|--|--|
| <pre> xlabel('Συχνότητα σε rads (π)') subplot(2,1,2) stem(n,real(x_original), 'MarkerFaceColor','red') title('Ανακατασκευασμένο σήμα x(n)'); xlabel('Χρόνος (n)');grid on </pre> | <p>την πλήρη ανακατασκευή του αρχικού σήματος $x(n)$.</p> |
|--|--|

3 Άλυτες Ασκήσεις

1. Έστω το σήμα $x(n) = u(n) - u(n-10)$, ορισμένο στο διάστημα $n=[-10, 20]$. Να υπολογιστεί και να σχεδιαστεί ο DTFT στο διάστημα συχνοτήτων $\omega=[-\pi, \pi]$, όπου για τη συχνότητα « ω » να ορίσετε διάνυσμα ισαπέχουσων συχνοτήτων μήκους $L=256$. Επίσης, να κατασκευάσετε γραφήματα για το πλάτος και τη φάση του DTFT.
2. Επαναλάβετε τα προηγούμενα για το σήμα $x(n) = 5\sin(\pi n/3)[u(n) - u(n-10)]$, ορισμένο στο διάστημα $n=[-10, 20]$, με $\omega=[-\pi, \pi]$ και $L=256$.

4 Επίλογος – Σύνδεση με τα Επόμενα

Στην παρούσα εργαστηριακή άσκηση καλύφθηκε ο υπολογισμός του ευθύ και του αντίστροφου μετασχηματισμού Fourier διακριτού χρόνου (DTFT και I-DTFT). Για τον υπολογισμό του DTFT χρησιμοποιήθηκαν δύο μέθοδοι: α) η εξίσωση ορισμού και β) η συνάρτηση `fft` του Matlab. Για τον υπολογισμό του I-DTFT, που μας επιτρέπει να ανακατασκευάσουμε το αρχικό διακριτό σήμα $x(n)$, χρησιμοποιήθηκε η συνάρτηση `ifft` του Matlab. Τέλος, παρουσιάστηκε ο τρόπος γραφικής απεικόνισης του φάσματος (DTFT) ενός διακριτού σήματος.

Στην επόμενη εργαστηριακή άσκηση θα καλυφθούν ζητήματα που αφορούν την απόκριση συχνότητας συστημάτων διακριτού χρόνου, διάφορα ψηφιακά φίλτρα, καθώς και οι τρόποι διασύνδεσή τους.

5 Βιβλιογραφία

Hayes , Μ. Η. (2002). *Ψηφιακή Επεξεργασία Σήματος*. (Α. Δημητριάδης, Μεταφρ.) Θεσσαλονίκη: Τζιόλα.

Hsu, H. P. (2002). *Αναλογικές και Ψηφιακές Επικοινωνίες*. Θεσσαλονίκη: Τζιόλα.

Jayant, N. S., & Noll, P. (1984). *Digital Coding of Waveforms: Principles and Applications to Speech and Video*. Prentice Hall.

Mano, M. M., & Ciletti, M. D. (2012). *Digital Design (5 εκδ.)*. Pearson Education.

MathWorks - DSP. (2017). Ανάκτηση 2 10, 2017, από Getting Started with Matlab DSP: <https://www.mathworks.com/help/dsp/getting-started-with-dsp-system-toolbox.html>

Proakis, J. G., & Salehi, M. (2008). *Digital Communications (5 εκδ.)*. McGraw-Hill Higher Education.

Μουρτζόπουλος, Ι. (2012). *Ψηφιακή Τεχνολογία Ήχου*. Πάτρα: Πανεπιστήμιο Πατρών.

Εργαστηριακή Άσκηση 8: Απόκριση Συχνότητας

1 Εισαγωγικά Στοιχεία

1.1 Σκοπός

Σκοπός της παρούσας εργαστηριακής άσκησης είναι η εξοικείωση των φοιτητών – σπουδαστών με τον υπολογισμό της απόκρισης συχνότητας απλών και σύνθετων διατάξεων συστημάτων, καθώς των βασικών ψηφιακών φίλτρων.

1.2 Αναμενόμενα Μαθησιακά Αποτελέσματα

Με την ολοκλήρωση της παρούσας εργαστηριακής άσκησης, ο φοιτητής – σπουδαστής θα είναι σε θέση να:

- Υπολογίζει την απόκριση συχνότητας απλών συστημάτων
- Υπολογίζει την απόκριση συχνότητας μίας σύνθετης διάταξης με συστήματα συνδεδεμένα σε σειρά και/ή παράλληλα
- Υπολογίζει την απόκριση συχνότητας των βασικών ψηφιακών φίλτρων, όπως του:
 - Κατωδιαβατού φίλτρου
 - Ανωδιαβατού φίλτρου
 - Ζωνοδιαβατού φίλτρου
 - Ζωνοφρακτικού φίλτρου
- Κατασκευάζει τη γραφική παράσταση του πλάτους και της φάσης της απόκρισης συχνότητας συστημάτων

1.3 Εκτιμώμενη Διάρκεια

Ο εκτιμώμενος χρόνος που απαιτείται από τον φοιτητή – σπουδαστή για την ολοκλήρωση της παρούσας εργαστηριακής άσκησης είναι 2 διδακτικές ώρες.

1.4 Προαπαιτούμενα

Διάλεξη 6 της θεωρίας, μετασχηματισμός DTFT.

1.5 Εισαγωγή

Στην άσκηση αυτή θα ασχοληθούμε με την χρήση του DTFT για τον υπολογισμό της απόκρισης συχνότητας συστημάτων και φίλτρων.

1.6 Κατάλογος Χρήσιμων Συναρτήσεων Matlab

Κατά την εκπόνηση των ασκήσεων χρησιμοποιούνται οι ακόλουθες συναρτήσεις – εντολές του Matlab:

| Συνάρτηση | Επεξήγηση |
|-----------|--|
| length | Επιστρέφει το μήκος ενός πίνακα-διανύσματος |
| zeros | Δημιουργία πίνακα – διανύσματος με μηδενικά |
| linspace | Δημιουργεί διάνυσμα ισαπέχοντων τιμών |
| abs | Απόλυτη τιμή (μέτρο μιγαδικού) |
| angle | Υπολογίζει τη φάση (γωνία) μιγαδικού |
| fft | Υπολογίζει τον DFT χρησιμοποιώντας τον αλγόριθμο γρήγορου μετασχηματισμού Fourier |
| fftshift | Μετατοπίζει το φάσμα ώστε οι μηδενικές συχνότητες να βρίσκονται στο κέντρο (χρήσιμο για την απεικόνιση του φάσματος) |
| conv | Υπολογίζει τη συνέλιξη δύο σημάτων |
| filter | Υπολογίζει την απόκριση (έξοδο) ενός συστήματος – φίλτρου |
| freqz | Υπολογίζει την απόκριση συχνότητας ενός συστήματος |
| impz | Υπολογίζει την κρουστική απόκριση ενός συστήματος |
| figure | Δημιουργεί νέο παράθυρο για την απεικόνιση γραφικών παραστάσεων |
| plot | Δημιουργεί μία γραφική παράσταση |

| | |
|---------|--|
| subplot | Τμηματοποίηση ενός παραθύρου για γραφικές παραστάσεις σε γραμμές και στήλες, προκειμένου στο ίδιο παράθυρο να απεικονιστούν πολλαπλές γραφικές παραστάσεις |
| stem | Γραφική παράσταση διακριτών σημάτων – ακολουθιών |
| title | Τοποθετεί τίτλο σε μια γραφική παράσταση |
| axis | Ορίζει τα όρια των αξόνων και άλλες ιδιότητες των αξόνων σε μια γραφική παράσταση |
| grid | Προσθέτει – αφαιρεί πλέγμα στις γραφικές παραστάσεις, βελτιώνοντας την ανάγνωσή τους |
| xlabel | Τοποθετεί τίτλο στον οριζόντιο άξονα μίας γραφικής παράστασης |

2 Απόκριση Συχνότητας

Η απόκριση συχνότητας ενός συστήματος συμβολίζεται $H(e^{j\omega})$ και είναι ο μετασχηματισμός DTFT της κρουστικής απόκρισης $h(n)$ του συστήματος. Η απόκριση συχνότητας συστήματος προκύπτει ως εξής:

$$y(n) = x(n) * h(n) \xleftrightarrow{DTFT} Y(e^{j\omega}) = X(e^{j\omega})H(e^{j\omega})$$

$$H(e^{j\omega}) = \frac{Y(e^{j\omega})}{X(e^{j\omega})}$$

Ο υπολογισμός της απόκρισης συχνότητας ενός συστήματος, όταν γνωρίζουμε την κρουστική του απόκριση $h(n)$ υπολογίζεται μέσω του DTFT από τη σχέση:

$$H(e^{j\omega}) = \sum_{n=-\infty}^{+\infty} h(n)e^{-j\omega n}$$

Βέβαια, για να ορίζεται ο DTFT της $h(n)$, θα πρέπει η απόλυτη τιμή του $h(n)$ να συγκλίνει απόλυτα, δηλαδή να ισχύει ότι:

$$\sum_{n=-\infty}^{+\infty} |h(n)| = S < \infty$$

Η συνάρτηση $H(e^{j\omega})$ είναι μιγαδική συνάρτηση, η οποία εξαρτάται από την πραγματική μεταβλητή « ω » (κυκλική συχνότητα), οπότε μπορεί να γραφτεί συναρτήσει του πραγματικού και του φανταστικού μέρους της, ως εξής:

$$H(e^{j\omega}) = H_{re}(e^{j\omega}) + jH_{im}(e^{j\omega})$$

όπου: $H_{re}(e^{j\omega})$ και $H_{im}(e^{j\omega}) =$ πραγματικό και φανταστικό μέρος αντίστοιχα.

Εναλλακτικά, μπορεί να γραφτεί και στην εξής μορφή:

$$H(e^{j\omega}) = |H(e^{j\omega})|e^{j\theta(\omega)}$$

$$\text{όπου: } \theta(\omega) = \arg\{H(e^{j\omega})\} = \tan^{-1}\left\{\frac{H_{im}(e^{j\omega})}{H_{re}(e^{j\omega})}\right\}$$

Η ποσότητα $|H(e^{j\omega})|$ λέγεται συνάρτηση πλάτους ή πλάτος φάσματος, ενώ η ποσότητα $\theta(\omega)$ λέγεται συνάρτηση φάσης ή φάση φάσματος.

Η συνάρτηση $H(e^{j\omega})$ είναι περιοδική και συνεχής συνάρτηση της κυκλικής συχνότητας « ω » με περίοδο 2π . Για την αναπαράστασή του συνηθίζεται να χρησιμοποιούνται οι συχνότητες στο διάστημα $[-\pi, \pi]$.

Επίσης, όταν γνωρίζουμε την απόκριση συχνότητας $H(e^{j\omega})$ ενός συστήματος μπορούμε να ανακατασκευάσουμε την κρουστική του απόκριση $h(n)$, υπολογίζοντας τον αντίστροφο μετασχηματισμό DTFT (I-DTFT) της $H(e^{j\omega})$, ο οποίος δίνεται από τη σχέση:

$$h(n) = \frac{1}{2\pi} \int_{-\pi}^{\pi} H(e^{j\omega}) e^{j\omega n} d\omega$$

Οι πράξεις του DTFT και του αντίστροφου DTFT (I-DTFT) για τον υπολογισμό της απόκρισης συχνότητας και της κρουστικής απόκρισης αντίστοιχα, συνηθίζεται για συντομία να συμβολίζονται ως εξής:

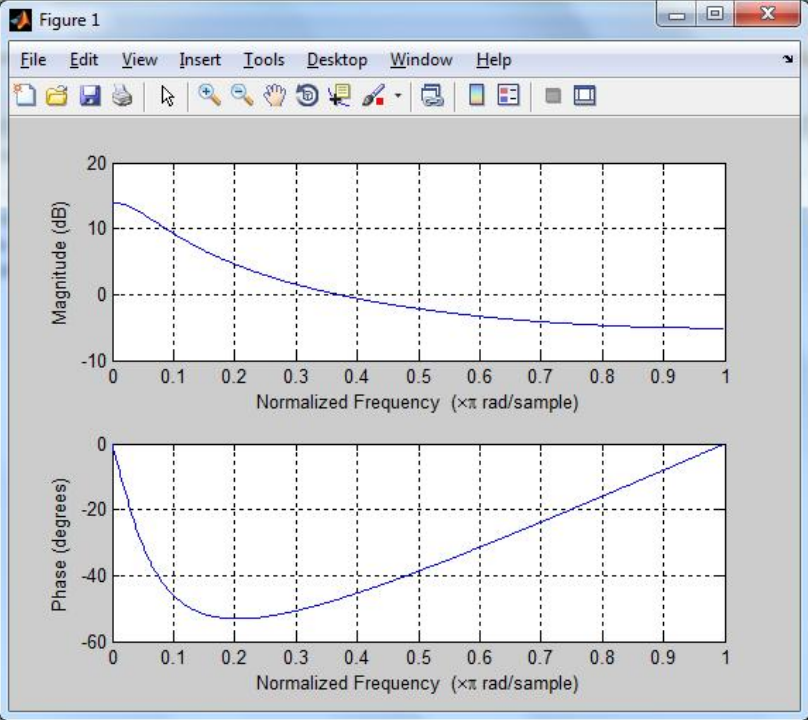
$$h(n) \xleftrightarrow{DTFT} H(e^{j\omega})$$

$$H(e^{j\omega}) \xleftrightarrow{I-DTFT} h(n)$$

Άσκηση 1^η

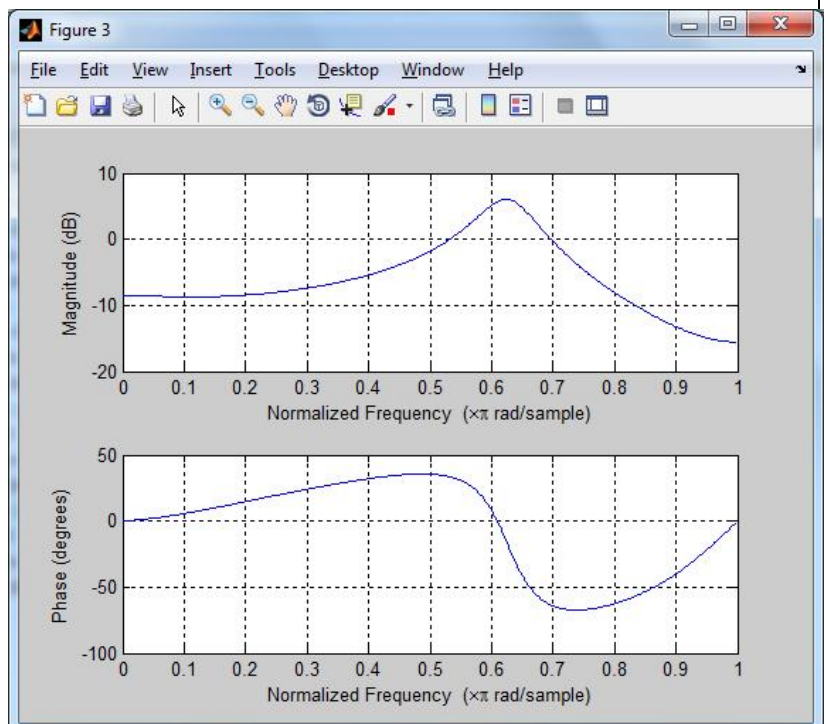
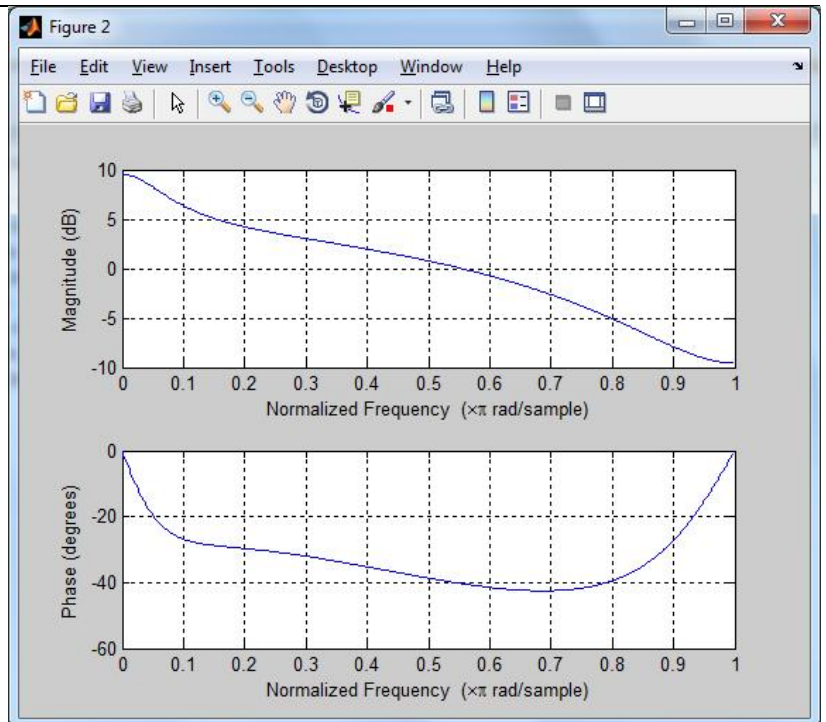
Να υπολογιστεί και να σχεδιαστεί το πλάτος και η φάση της απόκρισης συχνότητας των συστημάτων που περιγράφονται από τις εξισώσεις διαφορών:

- $y(n+1) - 0.8y(n) = x(n+1)$
- $y(n) - 0.8y(n-1) = x(n) - 0.4x(n-2)$
- $2y(n) + 0.6y(n-2) - y(n-3) = x(n) - 0.4x(n-2)$

| Κώδικας Matlab | Αποτέλεσμα |
|---|--|
| <pre> %y(n+1)-0.8y(n)=x(n+1) a=[1 -0.8]; b=[1]; figure(1) freqz(b,a) %b)y(n)-0.8y(n-1)=x(n)-0.4x(n-2) a=[1 -0.8]; b=[1 0 -0.4]; figure(2) freqz(b,a) %c)2y(n)+0.6y(n-2)-y(n-3)=x(n)- 0.4x(n-2) a=[2 0 0.6 -1]; b=[1 0 -0.4]; </pre> |  |

figure(3)

freqz(b,a)



Παρατήρηση:

Η freqz κατασκευάζει για κάθε σύστημα τις γραφικές παραστάσεις του πλάτους και της φάσης της

| | |
|--|--|
| | κανονικοποιημένης απόκρισης συχνότητας, στο διάστημα συχνοτήτων $[0, \pi]$. |
|--|--|

Άσκηση 2^η

Δίνεται ένα γραμμικό και χρονικά αμετάβλητο (ΓΑΚΜ) σύστημα, που περιγράφεται από την εξίσωση διαφορών: $y(n) = 1.1y(n-1) - 0.5y(n-2) - 0.3y(n-4) + 0.5x(n) - 0.2x(n-1)$. Υπολογίστε:

- την απόκριση συχνότητας του συστήματος
- την κρουστική απόκριση του συστήματος για $n=[0,100]$
- την έξοδο του συστήματος σε είσοδο $x(n)=u(n)-u(n-6)$, στο διάστημα $n=[0, 100]$

| Κώδικας Matlab | Αποτέλεσμα |
|----------------|------------|
|----------------|------------|


```

% y(n)-1.1y(n-1)+0.5y(n-2)+0.3y(n-
4) = 0.5x(n)-0.2x(n-1)

a=[1 -1.1 0.5 0 0.3]; b=[0.5 -0.2];

figure(1)

freqz(b,a)

n=0:100;

h=impz(b,a,n);

figure(2);

stem(n,h);title('h(n)');

x=[ones(1,6) zeros(1,95)];

y=filter(b,a,x);

figure(3);

subplot(3,1,1);

stem(n,x);title('x(n)');grid on;

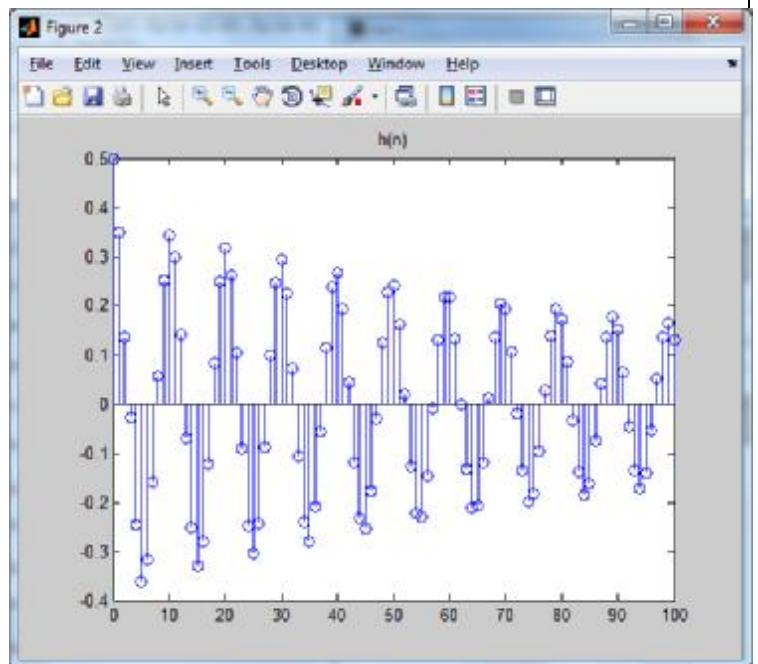
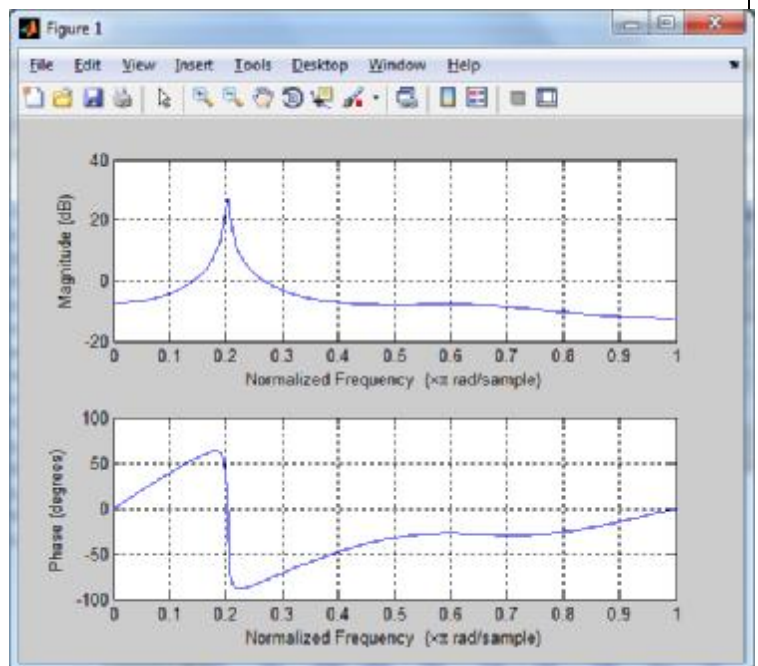
subplot(3,1,2);

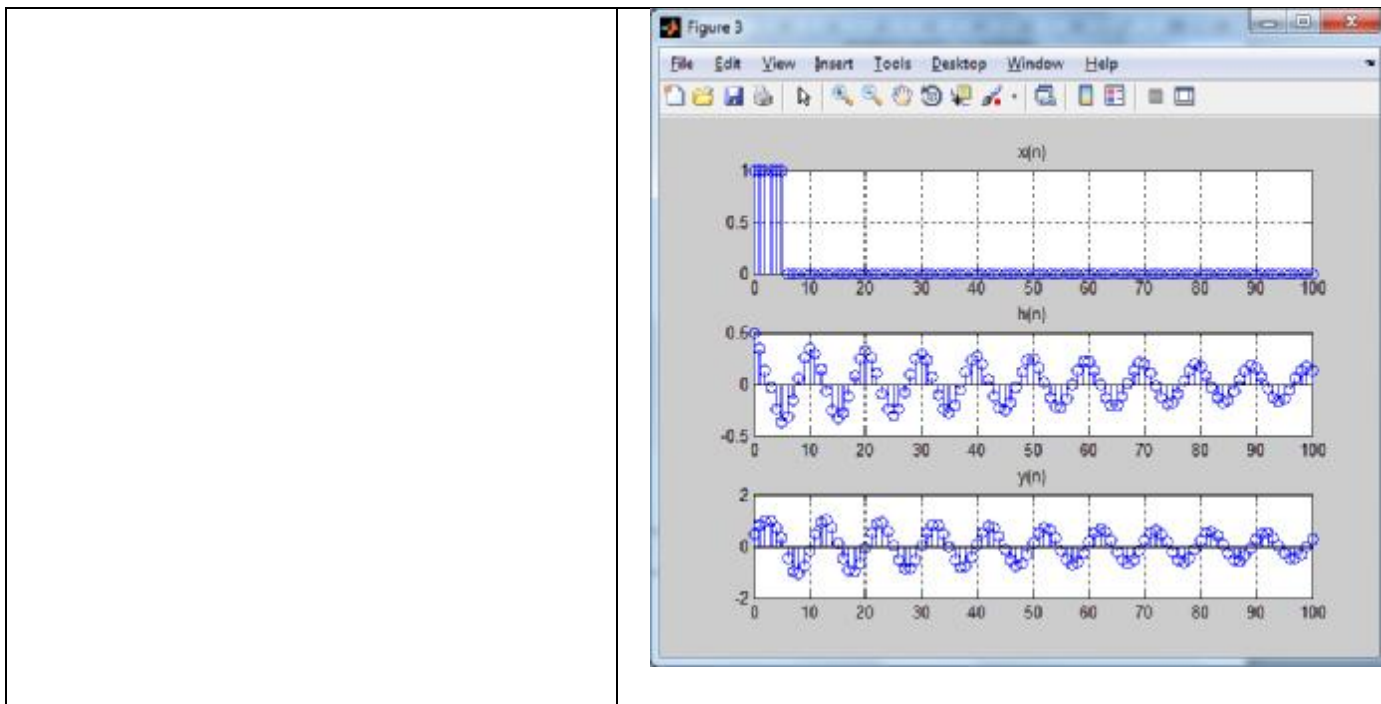
stem(n,h);title('h(n)');grid on;

subplot(3,1,3);

stem(n,y);title('y(n)');grid on;

```





Άσκηση 3^η

Δίνεται σύστημα με κρουστική απόκριση $h(n)=0.9^n u(n)$, για $n=[0,100]$. Υπολογίστε:

- την απόκριση συχνότητας $H(e^{j\omega})$ και σχεδιάστε το πλάτος και τη φάση της, για $L=512$ σημεία και για τις συχνότητες $[-\pi, \pi]$
- την έξοδο του συστήματος σε είσοδο $x(n)=\delta(n-1)$

Κώδικας Matlab

Αποτέλεσμα

```

% h(n)=0.9^n u(n)

n=0:100;

h=0.9.^n;

L=512;

w=linspace(-pi,pi,L);

H=fft(h,L);

figure('name', 'Απόκριση
συχνότητας');

subplot(3,1,1);

stem(n,h,'MarkerFaceColor','red');

title('h(n)');

xlabel('Χρόνος (n)');grid on

subplot(3,1,2);

plot(w/pi, abs(fftshift(H)));

title('|H(e^{j\omega})|');grid on;

xlabel('Συχνότητα σε rads (π)');

subplot(3,1,3);

plot(w/pi, angle(fftshift(H)));

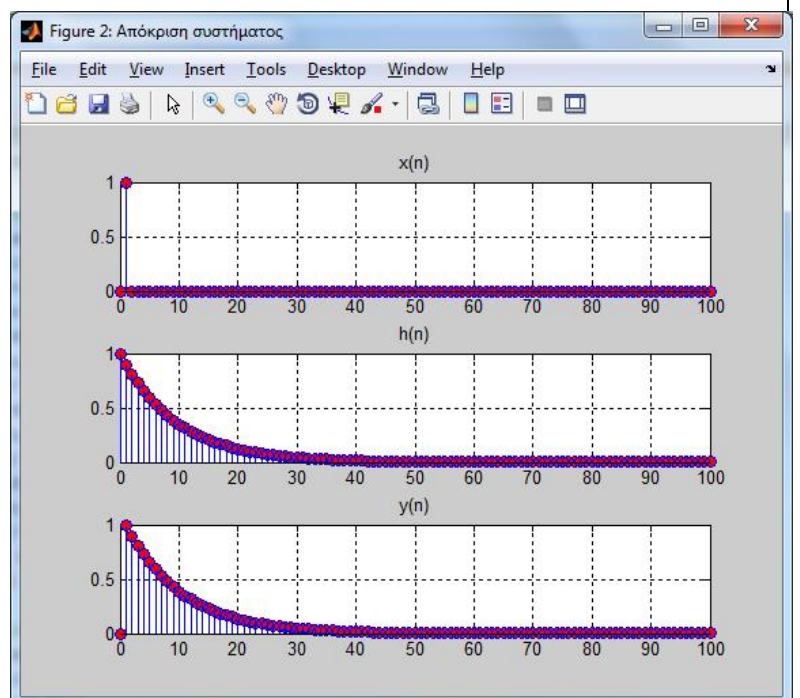
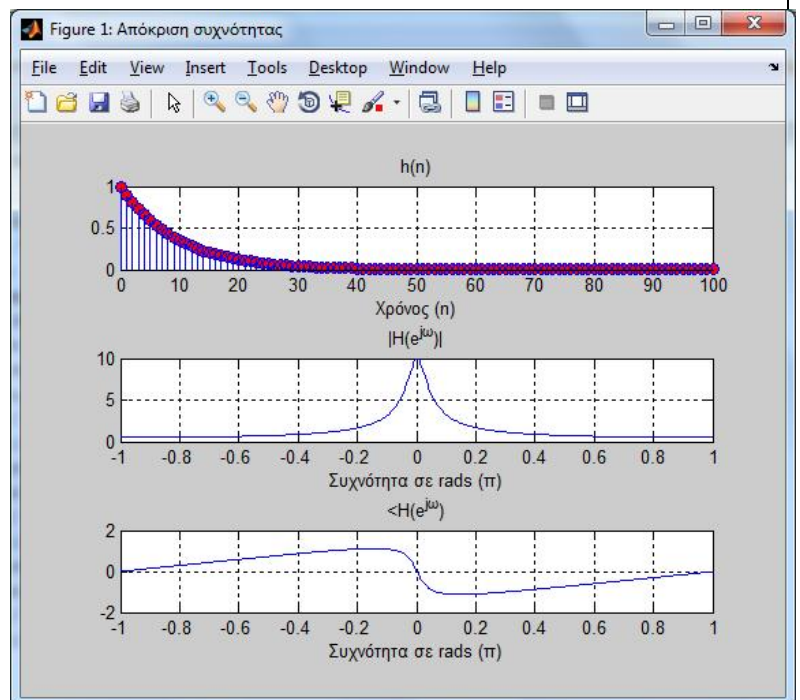
title('<H(e^{j\omega})');grid on;

xlabel('Συχνότητα σε rads (π)');

%x(n)=\delta(n-1)

x=zeros(1,101);

```



| | |
|--|--|
| <pre> x(n==1)=1; y=conv(x,h); figure('name', 'Απόκριση συστήματος'); subplot(3,1,1); stem(n,x,'MarkerFaceColor','red'); title('x(n)');grid on; subplot(3,1,2); stem(n,h,'MarkerFaceColor','red'); title('h(n)');grid on; subplot(3,1,3); stem(n,y(1:101),'MarkerFaceColor', 'red'); title('y(n)');grid on; </pre> | |
|--|--|

3 Άλυτες Ασκήσεις

1. Δίνεται ένα γραμμικό και χρονικά αμετάβλητο (ΓΑΚΜ) σύστημα, που περιγράφεται από την εξίσωση διαφορών: $y(n) = 0.3y(n-1) + 0.7x(n)$. Υπολογίστε και σχεδιάστε:
 - την απόκριση συχνότητας του συστήματος
 - την κρουστική απόκριση του συστήματος για $n=[0,100]$
 - την έξοδο του συστήματος σε είσοδο $x(n)=u(n)-u(n-10)$, στο $n=[0, 100]$

4 Επίλογος – Σύνδεση με τα Επόμενα

Στην παρούσα εργαστηριακή άσκηση καλύφθηκε ο υπολογισμός της απόκρισης συχνότητας συστημάτων διακριτού χρόνου, καθώς επίσης και τα ψηφιακά φίλτρα και η διασύνδεση συστημάτων στο πεδίο της συχνότητας.

Στην επόμενη εργαστηριακή άσκηση θα καλυφθούν ζητήματα που αφορούν τον διακριτό μετασχηματισμό Fourier (DFT).

5 Βιβλιογραφία

Hayes , Μ. Η. (2002). *Ψηφιακή Επεξεργασία Σήματος*. (Α. Δημητριάδης, Μεταφρ.) Θεσσαλονίκη: Τζιόλα.

Hsu, Η. Ρ. (2002). *Αναλογικές και Ψηφιακές Επικοινωνίες*. Θεσσαλονίκη: Τζιόλα.

Jayant, Ν. S., & Noll, Ρ. (1984). *Digital Coding of Waveforms: Principles and Applications to Speech and Video*. Prentice Hall.

Mano, Μ. Μ., & Ciletti, Μ. D. (2012). *Digital Design (5 εκδ.)*. Pearson Education.

MathWorks - DSP. (2017). Ανάκτηση 2 10, 2017, από Getting Started with Matlab DSP: <https://www.mathworks.com/help/dsp/getting-started-with-dsp-system-toolbox.html>

Proakis, J. G., & Salehi, Μ. (2008). *Digital Communications (5 εκδ.)*. McGraw-Hill Higher Education.

Μουρτζόπουλος, Ι. (2012). *Ψηφιακή Τεχνολογία Ήχου*. Πάτρα: Πανεπιστήμιο Πατρών.

Εργαστηριακή Άσκηση 9: Διακριτός Μετασχηματισμός Fourier - DFT

1 Εισαγωγικά Στοιχεία

1.1 Σκοπός

Σκοπός της παρούσας εργαστηριακής άσκησης είναι η εξοικείωση των φοιτητών – σπουδαστών με τον Διακριτό μετασχηματισμό Fourier (DFT) και την χρήση του στον υπολογισμό γραμμικών και κυκλικών συνελίξεων μεταξύ σημάτων διακριτού χρόνου (ΣΔΧ).

1.2 Αναμενόμενα Μαθησιακά Αποτελέσματα

Με την ολοκλήρωση της παρούσας εργαστηριακής άσκησης, ο φοιτητής – σπουδαστής θα είναι σε θέση να:

- Υπολογίζει:
 - τον DFT ενός ΣΔΧ με τη συνάρτηση `fft` του Matlab
 - τον I-DTFT ενός ΣΔΧ με τη συνάρτηση `ifft` του Matlab
- Υπολογίζει την κυκλική συνέλιξη δύο ΣΔΧ μέσω των DFT και I-DFT
- Υπολογίζει τη γραμμική συνέλιξη δύο ΣΔΧ μέσω των DFT και I-DFT

1.3 Εκτιμώμενη Διάρκεια

Ο εκτιμώμενος χρόνος που απαιτείται από τον φοιτητή – σπουδαστή για την ολοκλήρωση της παρούσας εργαστηριακής άσκησης είναι 2 διδακτικές ώρες.

1.4 Προαπαιτούμενα

Διάλεξη 7 της θεωρίας.

1.5 Εισαγωγή

Στην άσκηση αυτή θα ασχοληθούμε με τον DFT και τον I-DFT για σήματα διακριτού χρόνου. Δεδομένης της δυσκολίας υπολογισμού του DTFT, είναι πρακτικά πολύ πιο εύκολο να ψηφιοποιήσουμε τη συχνότητα, λαμβάνοντας δείγματα του DTFT. Αυτός

ο μετασχηματισμός λέγεται Διακριτός μετασχηματισμός Fourier (DFT) και ουσιαστικά αποτελεί δειγματοληψία του DTFT στη συχνότητα.

1.6 Κατάλογος Χρήσιμων Συναρτήσεων Matlab

Κατά την εκπόνηση των ασκήσεων χρησιμοποιούνται οι ακόλουθες συναρτήσεις – εντολές του Matlab:

| Συνάρτηση | Επεξήγηση |
|-----------|--|
| length | Επιστρέφει το μήκος ενός πίνακα-διανύσματος |
| zeros | Δημιουργία πίνακα – διανύσματος με μηδενικά |
| abs | Απόλυτη τιμή (μέτρο μιγαδικού) |
| angle | Υπολογίζει τη φάση (γωνία) μιγαδικού |
| real | Υπολογίζει το πραγματικό μέρος μιγαδικού |
| imag | Υπολογίζει το φανταστικό μέρος μιγαδικού |
| fft | Υπολογίζει τον DFT χρησιμοποιώντας τον αλγόριθμο γρήγορου μετασχηματισμού Fourier |
| ifft | Υπολογίζει τον αντίστροφο DFT χρησιμοποιώντας τον αλγόριθμο γρήγορου μετασχηματισμού Fourier |
| figure | Δημιουργεί νέο παράθυρο για την απεικόνιση γραφικών παραστάσεων |
| plot | Δημιουργεί μία γραφική παράσταση |
| subplot | Τμηματοποίηση ενός παραθύρου για γραφικές παραστάσεις σε γραμμές και στήλες, προκειμένου στο ίδιο παράθυρο να απεικονιστούν πολλαπλές γραφικές παραστάσεις |
| stem | Γραφική παράσταση διακριτών σημάτων – ακολουθιών |
| title | Τοποθετεί τίτλο σε μια γραφική παράσταση |

| | |
|--------|--|
| xlim | Ορίζει τα όρια τιμών του άξονα x σε μία γραφική παράσταση |
| grid | Προσθέτει – αφαιρεί πλέγμα στις γραφικές παραστάσεις, βελτιώνοντας την ανάγνωσή τους |
| xlabel | Τοποθετεί τίτλο στον οριζόντιο άξονα μίας γραφικής παράστασης |

2 Διακριτός Μετασχηματισμός Fourier (DFT)

Το φάσμα (DTFT) ενός διακριτού σήματος είναι γενικά δύσκολο να υπολογιστεί με τη βοήθεια ψηφιακών κυκλωμάτων, εξαιτίας του γεγονότος ότι η συχνότητα «ω» είναι συνεχής συνάρτηση, άρα έχει άπειρες τιμές στο διάστημα που είναι ορισμένη. Για το λόγο αυτό, είναι πρακτικά πολύ πιο εύκολο να θεωρούμε τη συχνότητα ως διακριτή ποσότητα και να υπολογίζουμε δείγματα του φάσματος για πεπερασμένο πλήθος συχνοτήτων. Αυτή η ψηφιοποίηση της συχνότητας του DTFT μας οδηγεί στο Διακριτό μετασχηματισμό Fourier (DFT), που αποτελεί δειγματοληψία της συχνότητας του DTFT.

2.1 Ορισμός των Μετασχηματισμών DFT και I-DFT

Ο DFT N-σημείων ενός διακριτού σήματος $x(n)$ συμβολίζεται ως $X(k)$ και υπολογίζεται από τη σχέση:

$$X(k) = \sum_{n=0}^{N-1} x(n)e^{-j\frac{2\pi k}{N}n}, \text{ όπου } k = 0, 1, \dots, N-1$$

Ο DFT προκύπτει από τον DTFT αν θέσουμε $\omega = \frac{2\pi k}{N}$, όπου N είναι το πλήθος των σημείων δειγματοληψίας (συνήθως ισούται με το μήκος του σήματος) και το k παίρνει διακριτές τιμές στο $[0, N-1]$. Έτσι, λοιπόν, ο DFT ενός σήματος αποτελείται από πεπερασμένο πλήθος τιμών, οι οποίες μάλιστα είναι μιγαδικές.

Αν γνωρίζουμε τον DFT $X(k)$ ενός διακριτού σήματος $x(n)$ μπορούμε να ανακατασκευάσουμε το αρχικό σήμα $x(n)$, υπολογίζοντας τον I-DFT της $X(k)$ από τη σχέση:

$$x(n) = \frac{1}{N} \sum_{k=0}^{N-1} X(k) e^{j\frac{2\pi k}{N}n}, \text{ όπου } n = 0, 1, \dots, N-1$$

Συνήθως στις σχέσεις των DFT και I-DFT θέτουμε $e^{-j\frac{2\pi}{N}} = W_N$, οπότε γράφονται συντομότερα ως εξής:

$$X(k) = \sum_{n=0}^{N-1} x(n) W_N^{nk}, \text{ όπου } k = 0, 1, \dots, N-1$$

$$x(n) = \frac{1}{N} \sum_{k=0}^{N-1} X(k) W_N^{-nk}, \text{ όπου } n = 0, 1, \dots, N-1$$

Οι πράξεις του DFT και του αντίστροφου DFT (I-DFT) συμβολίζονται αντίστοιχα:

$$x(n) \xleftrightarrow{DFT} X(k)$$

$$X(k) \xleftrightarrow{I-DFT} x(n)$$

Στο Matlab οι μετασχηματισμοί DFT και I-DFT υπολογίζονται μέσω των συναρτήσεων `fft` και `ifft` αντίστοιχα. Οι συναρτήσεις αυτές υπολογίζουν τους μετασχηματισμούς χρησιμοποιώντας τον αλγόριθμο γρήγορου μετασχηματισμού Fourier (FFT), ο οποίος μειώνει το υπολογιστικό κόστος. Η συνάρτηση `fft` συντάσσεται ως εξής:

X=fft(x, N), όπου:

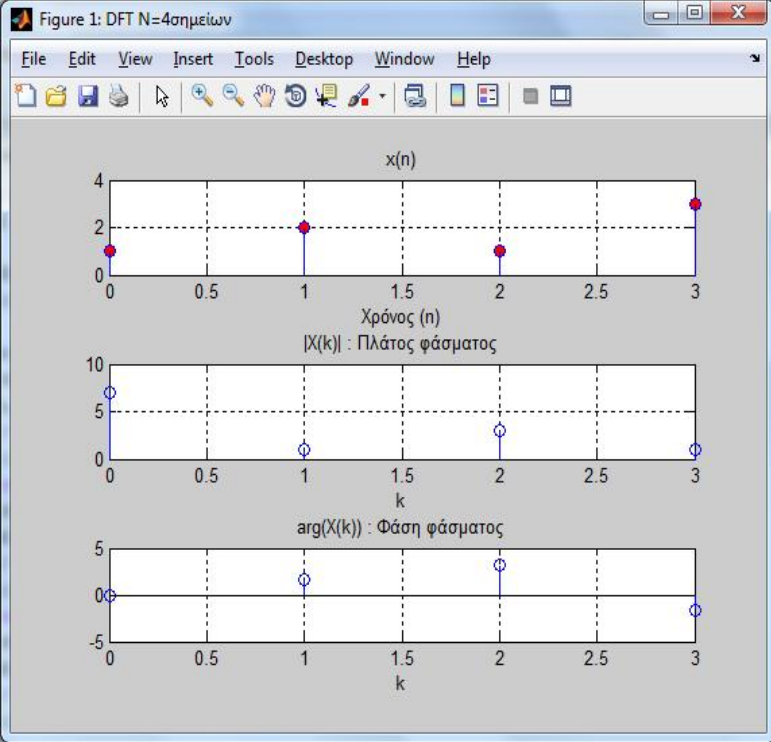
- X = σήμα εισόδου $x(n)$
- N = πλήθος σημείων DFT. Αν δεν δοθεί, τότε ισούται με το μήκος του x
- X = DFT του x

Υπενθυμίζεται ότι οι τιμές του DFT είναι μιγαδικοί αριθμοί, συνεπώς για τη σωστή απεικόνιση του DFT, πρέπει να σχεδιάζουμε τις γραφικές παραστάσεις του πλάτους και της φάσης του DFT.

Άσκηση 1^η

Έστω το σήμα $x(n)=\delta(n)+2\delta(n-1)+\delta(n-2)+3\delta(n-3)$. Να υπολογίσετε με χρήση του fft και να σχεδιάσετε τον DFT της $x(n)$ για:

- c) $N = 4$ σημεία
- d) $N = 8$ σημεία και
- e) $N = 16$ σημεία

| Κώδικας Matlab | Αποτέλεσμα |
|---|--|
| <pre> A) n=0:3; x=[1 2 1 3]; % Σήμα x(n) N1=4; k1=0:N1-1; % k X1=fft(x,4); % DFT 4 σημείων figure('name','DFT N=4σημείων') subplot(3,1,1) stem(n,x,'MarkerFaceColor','red') title('x(n)'); xlabel('Χρόνος (n)'), grid on </pre> |  |

```

subplot(3,1,2)

stem(k1,abs(X1))

title('|X(k)| : Πλάτος
φάσματος');

xlabel('k'), grid on

subplot(3,1,3)

stem(k1,angle(X1))

title('arg(X(k)) : Φάση
φάσματος');

xlabel('k'), grid on

```

B)

```

n=0:7;

% Συμπληρώνω με 0 το x(n)
x=[1 2 1 3 zeros(1,4)];

N2=8;

k2=0:N2-1; % k
X2=fft(x,8); % DFT 8 σημείων

figure('name','DFT N=8σημείων')

subplot(3,1,1)

stem(n,x,'MarkerFaceColor','red')

title('x(n)');

xlabel('Χρόνος (n)'), grid on

subplot(3,1,2)

stem(k2,abs(X2))

title('|X(k)| : Πλάτος φάσματος');

xlabel('k'), grid on

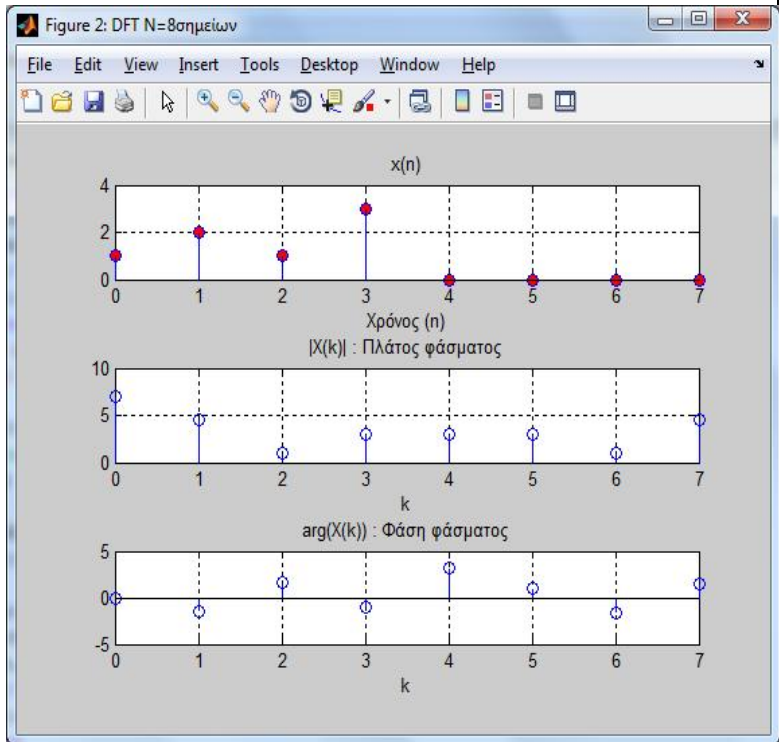
subplot(3,1,3)

stem(k2,angle(X2))

title('arg(X(k)) : Φάση φάσματος');

xlabel('k'), grid on

```



```

title('|X(k)| : Πλάτος
φάσματος');

xlabel('k'), grid on

subplot(3,1,3)

stem(k2,angle(X2))

title('arg(X(k)) : Φάση
φάσματος');

xlabel('k'), grid on

```

C)

```

n=0:15;

% Συμπληρώνω με 0 το x(n)

x=[1 2 1 3 zeros(1,12)];

N2=16;

k2=0:N2-1; % k

X2=fft(x,16); % DFT 16 σημείων

figure('name','DFT N=16σημείων')

subplot(3,1,1)

stem(n,x,'MarkerFaceColor','red')

title('x(n)');

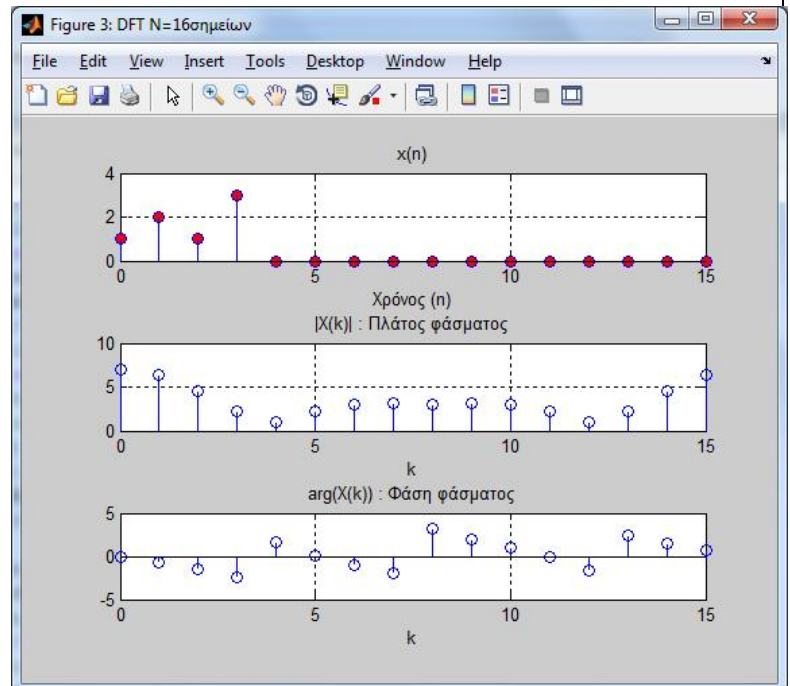
xlabel('Χρόνος (n)'), grid on

subplot(3,1,2)

stem(k2,abs(X2))

title('|X(k)| : Πλάτος
φάσματος');

```



```

xlabel('k'), grid on

subplot(3,1,3)

stem(k2,angle(X2))

title('arg(X(k)) : Φάση
φάσματος');

xlabel('k'), grid on

```

Άσκηση 2^η

Έστω το σήμα $x(n)=\sin(\pi n/6)$, ορισμένο στο διάστημα $n=[0, 255]$.

- a) Να υπολογιστεί με χρήση της συνάρτησης `fft` ο DFT $X(k)$. Επίσης, να δημιουργηθεί νέο παράθυρο στο οποίο να φαίνονται οι γραφικές παραστάσεις για το πλάτος, τη φάση, το πραγματικό μέρος και το φανταστικό μέρος του $X(k)$.
- b) Με χρήση της συνάρτησης `ifft` να ανακατασκευαστεί το αρχικό σήμα $x(n)$. Να δημιουργηθεί νέο παράθυρο στο οποίο να φαίνονται οι γραφικές παραστάσεις του αρχικού και του ανακατασκευασμένου σήματος, προκειμένου να είναι δυνατή η οπτική σύγκρισή τους.

| Κώδικας Matlab | Αποτέλεσμα |
|----------------|------------|
|----------------|------------|

```

n=0:255; % χρόνος - δείγματα

N=length(n); % Πλήθος σημείων

x=sin(pi*n/6); % Σήμα x(n)

X=fft(x); % αλλιώς X=fft(x,N)

k=0:N-1; % k

figure('name','Μετασχηματισμός
DFT')

subplot(2,2,1)

stem(k,abs(X),'fill')

title('|X(k)|')

xlabel('k'), grid on

subplot(2,2,2)

stem(k,angle(X),'fill')

title('arg(X(k))')

xlabel('k'), grid on

subplot(2,2,3)

stem(k,real(X),'fill')

title('Re(X(k))')

xlabel('k'), grid on

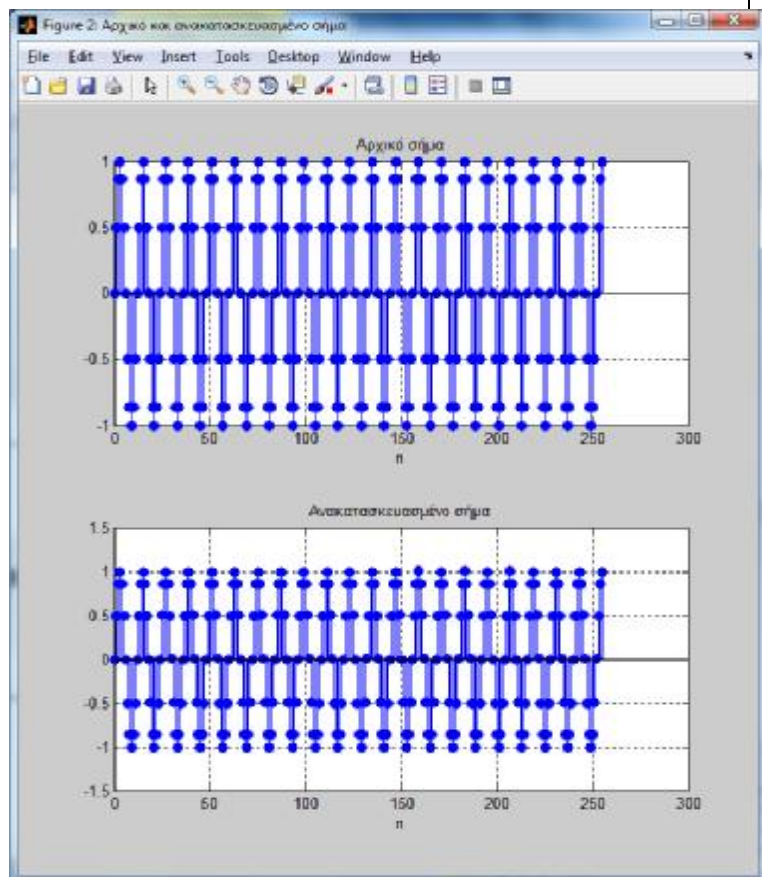
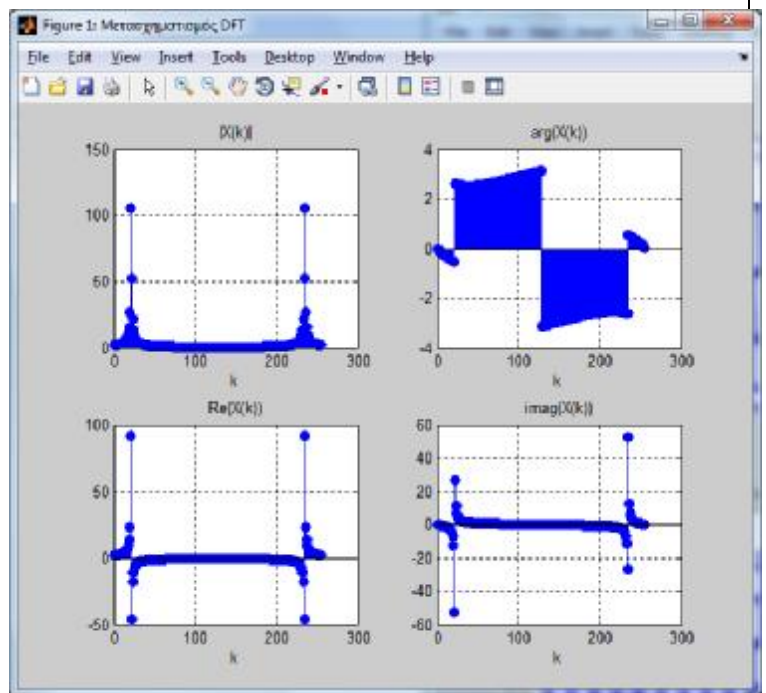
subplot(2,2,4)

stem(k,imag(X),'fill')

title('imag(X(k))')

xlabel('k'), grid on

```



```

% Ανακατασκευή αρχικού σήματος
x_rebuilted=ifft(X); % I-DFT

figure('name','Αρχικό και
ανακατασκευασμένο σήμα')

subplot(2,1,1)

stem(n,x,'fill')

title('Αρχικό σήμα')

xlabel('n'), grid on

subplot(2,1,2)

stem(n,x_rebuilted,'fill')

title('Ανακατασκευασμένο σήμα')

xlabel('n'), grid on

```

2.2 Κυκλική Συνέλιξη

Η κυκλική συνέλιξη N σημείων για δύο σήματα $x_1(n)$ και $x_2(n)$ είναι ένα σήμα $y(n)$ μήκους N , το οποίο υπολογίζεται ως εξής:

$$y(n) = x_1(n) \otimes x_2(n) = \sum_{k=0}^{N-1} x_1(k) x_2((n-k))_N$$

Η $x_2((n-k))_N$ είναι η κυκλικά μετατοπισμένη ακολουθία $x_2(n)$ κατά k θέσεις.

Ο υπολογισμός της κυκλικής συνέλιξης $y(n)$ μήκους N σημείων, για δύο σήματα $x_1(n)$ και $x_2(n)$, γίνεται χρησιμοποιώντας τους μετασχηματισμούς DFT και I-DFT, ακολουθώντας τα εξής βήματα:

1. Υπολογίζουμε τα μήκη N_1, N_2 των σημάτων $x_1(n)$ και $x_2(n)$ αντίστοιχα και βρίσκουμε το μέγιστο μήκος εκ των δύο, δηλαδή $N = \max\{N_1, N_2\}$
2. Υπολογίζουμε τους DFT N σημείων $X_1(k)$ και $X_2(k)$ των $x_1(n)$ και $x_2(n)$ αντίστοιχα
3. Υπολογίζουμε το γινόμενο $Y(k) = X_1(k) * X_2(k)$
4. Εκτελούμε I-DFT της $Y(k)$, οπότε βρίσκουμε τη $y(n)$, που ισούται με τη ζητούμενη κυκλική συνέλιξη, δηλαδή την $y(n) = x_1(n) \otimes x_2(n)$.

Υπενθυμίζεται ότι για τον υπολογισμό των DFT και I-DFT μπορούμε να χρησιμοποιήσουμε τις συναρτήσεις `fft` και `ifft` αντίστοιχα.

Άσκηση 3^η

$$\text{Έστω τα σήματα } \begin{cases} x(n) = \delta(n) - 2\delta(n-2) + 4\delta(n-5) \\ w(n) = \{1, 1, 2, -1\}, n \in [0, 3] \end{cases}$$

Με χρήση των DFT και I-DFT να υπολογιστούν:

- Η κυκλική συνέλιξη $y_1(n) = x(n) \otimes w(n)$.
- Η κυκλική συνέλιξη $y_2(n) = x(n) \otimes w(n)$ για $N=8$ σημεία.

Συγκρίνετε τα αποτελέσματα εμφανίζοντας κατάλληλα γραφήματα.

| Κώδικας Matlab | Αποτέλεσμα |
|----------------|------------|
|----------------|------------|

% Στο πρώτο ερώτημα δεν μας αναφέρεται το πλήθος των σημείων της κυκλικής συνέλιξης, οπότε θα χρησιμοποιήσουμε το μήκος του μεγαλύτερου σήματος

% Δημιουργία σημάτων $x(n)$ και $w(n)$

```
x=[1 0 -2 0 0 4];
```

```
w=[1 1 2 -1];
```

```
N1=length(x);
```

```
N2=length(w);
```

```
Na=max(N1,N2);
```

```
X1=fft(x,Na);
```

```
W1=fft(w,Na);
```

```
Y1=X1.*W1;
```

```
y1=ifft(Y1);
```

% Στο δεύτερο ερώτημα μας ζητείται κυκλική συνέλιξη $N=8$ σημείων, οπότε θα εκτελέσουμε τους μετασχηματισμούς για $N=8$ σημεία

```
Nb=8;
```

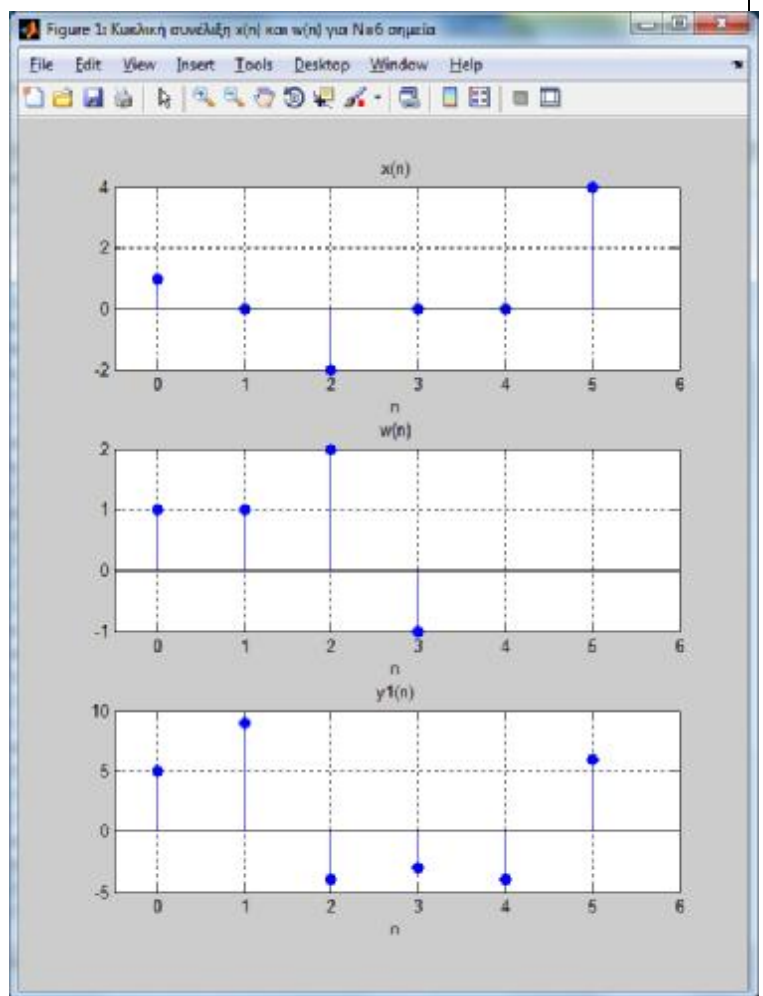
```
X2=fft(x,Nb);
```

```
W2=fft(w,Nb);
```

```
Y2=X2.*W2;
```

```
y2=ifft(Y2);
```

% Γραφικές παραστάσεις



Προκύπτει ότι:

$y1 =$

5 9 -4 -3 -4 6

```

figure('name', 'Κυκλική συνέλιξη
x(n) και w(n) για N=6 σημεία')

subplot(3,1,1)

stem(0:length(x)-1,x,'fill')

xlim([-0.5 Na])

title('x(n)'), xlabel('n'), grid on

subplot(3,1,2)

stem(0:length(w)-1,w,'fill')

xlim([-0.5 Na])

title('w(n)'), xlabel('n'), grid on

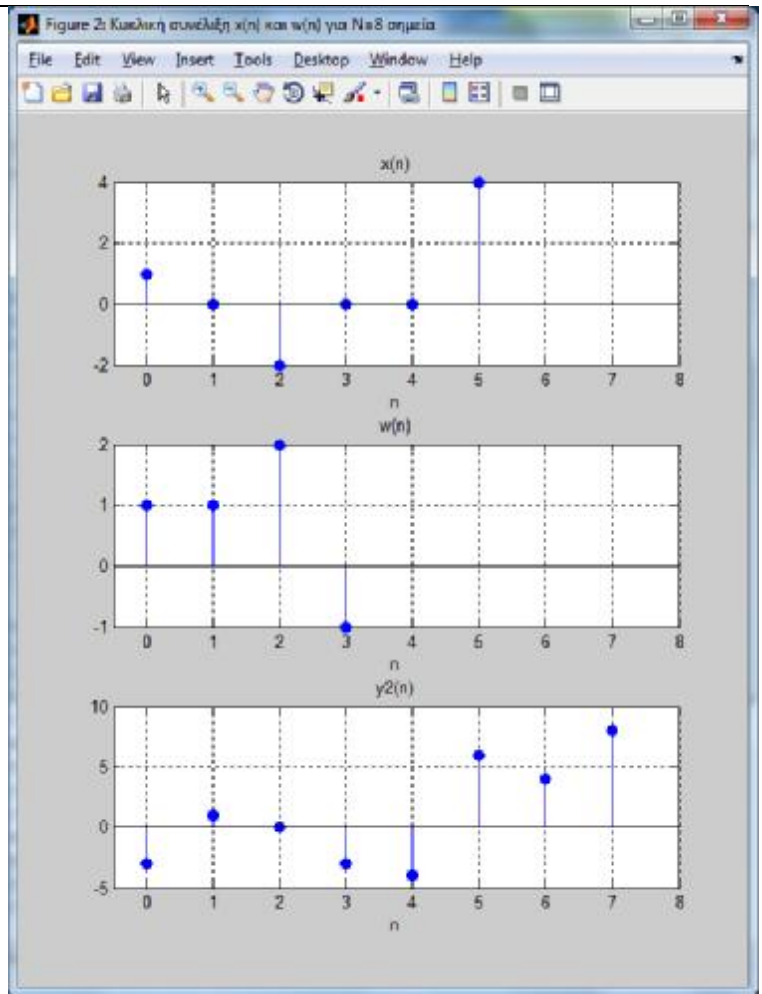
subplot(3,1,3)

stem([0:Na-1],y1,'fill')

xlim([-0.5 Na])

title('y1(n)'), xlabel('n'), grid on

```



```

figure('name', 'Κυκλική συνέλιξη
x(n) και w(n) για N=8 σημεία')

subplot(3,1,1)

stem(0:length(x)-1,x,'fill')

xlim([-0.5 Nb])

title('x(n)'), xlabel('n'), grid on

subplot(3,1,2)

stem(0:length(w)-1,w,'fill')

xlim([-0.5 Nb])

```

Προκύπτει ότι:

y2 =

-3 1 0 -3 -4 6 4 8

```

title('w(n)'),xlabel('n'), grid on

subplot(3,1,3)

stem([0:Nb-1],y2,'fill')

xlim([-0.5 Nb])

title('y2(n)'),xlabel('n'), grid on

```

2.3 Γραμμική Συνέλιξη μέσω DFT

Η γραμμική συνέλιξη δύο σημάτων $x_1(n)$ και $x_2(n)$ με μήκη N_1 και N_2 αντίστοιχα, υπολογίζεται εύκολα με χρήση του DFT, ακολουθώντας τα εξής βήματα:

1. Υπολογίζουμε τα μήκη N_1 , N_2 των σημάτων $x_1(n)$ και $x_2(n)$ αντίστοιχα και βρίσκουμε το μήκος L της γραμμικής συνέλιξης, δηλαδή $L = N_1 + N_2 - 1$
2. Υπολογίζουμε τους DFT L σημείων $X_1(k)$ και $X_2(k)$ των $x_1(n)$ και $x_2(n)$ αντίστοιχα
3. Υπολογίζουμε το γινόμενο $Y(k) = X_1(k) * X_2(k)$
4. Εκτελούμε I-DFT της $Y(k)$, οπότε βρίσκουμε τη $y(n)$, που είναι η κυκλική συνέλιξη L σημείων των $x_1(n)$ και $x_2(n)$, δηλαδή η $y(n) = x_1(n) \otimes x_2(n)$, που όμως ταυτίζεται με τη γραμμική συνέλιξη των $x_1(n)$ και $x_2(n)$.

Άσκηση 4^η

$$\text{Έστω τα σήματα } \begin{cases} x(n) = \delta(n) - 2\delta(n-2) + 4\delta(n-5) \\ w(n) = \{1, 1, 2, -1\}, n \in [0, 3] \end{cases}$$

Με χρήση των DFT και I-DFT να υπολογιστεί η γραμμική συνέλιξη $y(n) = x(n) * w(n)$ και να επαληθευτεί το αποτέλεσμα με τη συνάρτηση conv. Συγκρίνετε τα αποτελέσματα εμφανίζοντας κατάλληλα γραφήματα.

Κώδικας Matlab

```
% Μας ζητείται η γραμμική συνέλιξη  
με χρήση DFT, οπότε αρχικά θα βρούμε  
το μήκος της  $L=N1+N2-1$  και μετά θα  
υπολογίζουμε την κυκλική συνέλιξη  
για  $L$  σημεία
```

```
% Δημιουργία σημάτων  $x(n)$  και  $w(n)$ 
```

```
x=[1 0 -2 0 0 4];
```

```
w=[1 1 2 -1];
```

```
N1=length(x);
```

```
N2=length(w);
```

```
L=N1+N2-1;
```

```
X=fft(x,L);
```

```
W=fft(w,L);
```

```
Y=X.*W;
```

```
y=ifft(Y);
```

```
% Επαλήθευση με την conv
```

```
y_conv=conv(x,w);
```

```
% Γραφικές παραστάσεις
```

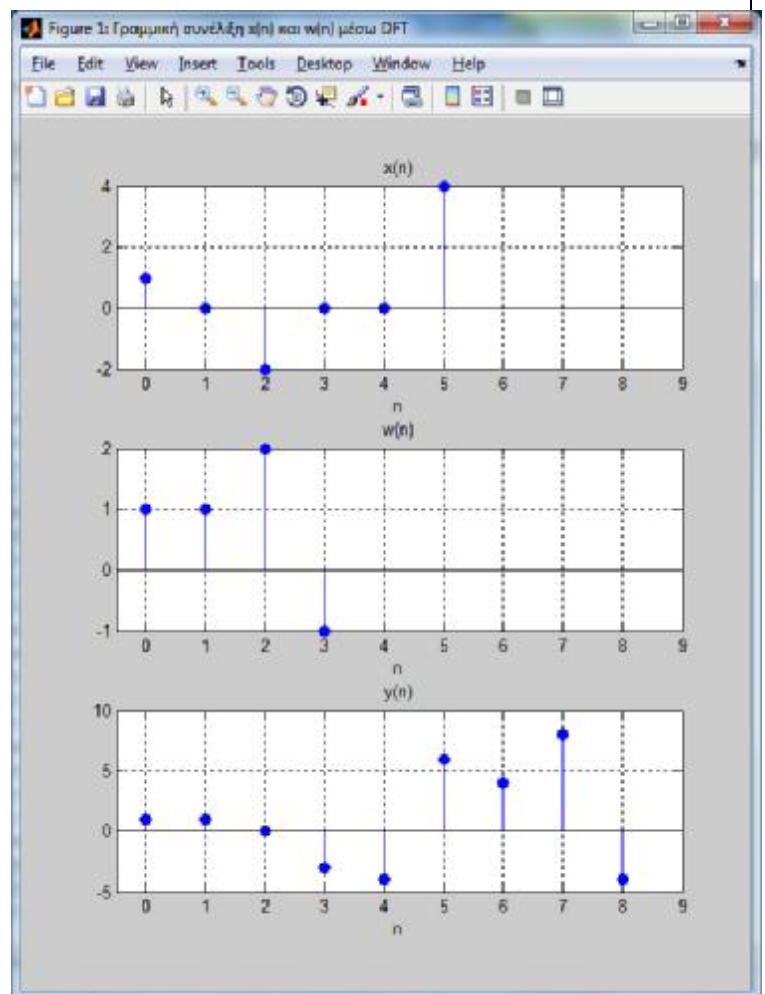
```
figure('name', 'Γραμμική συνέλιξη  
x(n) και w(n) μέσω DFT')
```

```
subplot(3,1,1)
```

```
stem(0:length(x)-1,x,'fill')
```

```
xlim([-0.5 L])
```

Αποτέλεσμα



Προκύπτει ότι:

$Y=$

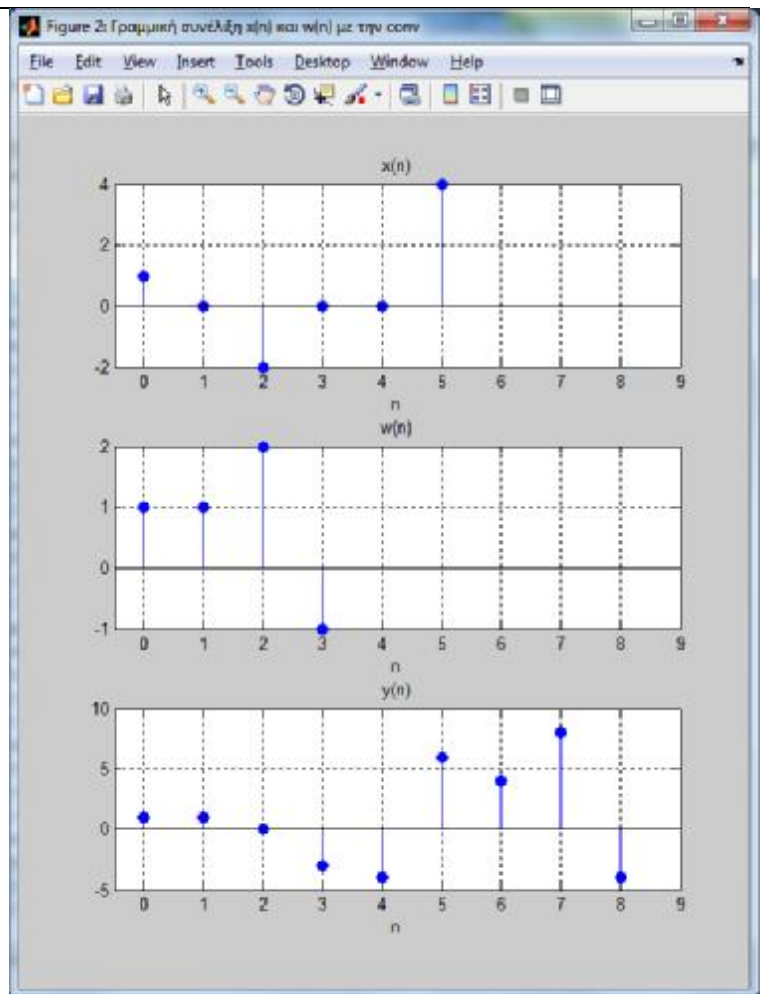
| | | | | |
|--------|--------|--------|---------|---------|
| 1.0000 | 1.0000 | 0.0000 | -3.0000 | -4.0000 |
| 6.0000 | 4.0000 | 8.0000 | -4.0000 | |

```

title('x(n)'), xlabel('n'), grid on
subplot(3,1,2)
stem(0:length(w)-1,w,'fill')
xlim([-0.5 L])
title('w(n)'), xlabel('n'), grid on
subplot(3,1,3)
stem([0:L-1],y,'fill')
xlim([-0.5 L])
title('y(n)'), xlabel('n'), grid on

figure('name', 'Γραμμική συνέλιξη
x(n) και w(n) με την conv')
subplot(3,1,1)
stem(0:length(x)-1,x,'fill')
xlim([-0.5 L])
title('x(n)'), xlabel('n'), grid on
subplot(3,1,2)
stem(0:length(w)-1,w,'fill')
xlim([-0.5 L])
title('w(n)'),
xlabel('n'),
grid on
subplot(3,1,3)
stem([0:L-1],y_conv,'fill')
xlim([-0.5 L])

```



Προκύπτει ότι:

$y_{conv} =$

1 1 0 -3 -4 6 4 8 -4

```
title('y(n)'),  
  
xlabel('n'),  
  
grid on
```

3 Άλυτες Ασκήσεις

1. Έστω τα σήματα $\begin{cases} x(n) = \delta(n) - \delta(n-1) + 2\delta(n-3) \\ w(n) = \{1, 0, -1, 0, 1, 2\}, n \in [0, 5] \end{cases}$

Υπολογίστε:

- τους DFT των $x(n)$ και $w(n)$ για $N=256$ σημεία
- την κυκλική συνέλιξη $y_1(n) = x(n) \otimes w(n)$.
- την κυκλική συνέλιξη $y_2(n) = x(n) \otimes w(n)$ για $N=5$ σημεία
- την γραμμική συνέλιξη των $x(n)$ και $w(n)$ με χρήση της κυκλικής συνέλιξης

4 Επίλογος – Σύνδεση με τα Επόμενα

Στην παρούσα εργαστηριακή άσκηση καλύφθηκε ο υπολογισμός του ευθύ και του αντίστροφου διακριτού μετασχηματισμού Fourier (DFT και I-DFT), μέσω των συναρτήσεων `fft` και `ifft` του Matlab. Επίσης, παρουσιάστηκαν τα βήματα υπολογισμού της κυκλικής συνέλιξης δύο σημάτων, καθώς και τα βήματα υπολογισμού της γραμμικής συνέλιξης μέσω των DFT και I-DFT (ταύτιση κυκλικής και γραμμικής συνέλιξης).

Στην επόμενη εργαστηριακή άσκηση θα καλυφθούν ζητήματα που αφορούν τον μετασχηματισμό Z , όπως ο ορισμός και οι βασικές του ιδιότητες, η περιοχή σύγκλισης (ROC) και η ανάπτυξη του μετασχηματισμού Z σε μερικά κλάσματα.

5 Βιβλιογραφία

- Hayes , Μ. Η. (2002). *Ψηφιακή Επεξεργασία Σήματος*. (Α. Δημητριάδης, Μεταφρ.) Θεσσαλονίκη: Τζιόλα.
- Hsu, Η. Ρ. (2002). *Αναλογικές και Ψηφιακές Επικοινωνίες*. Θεσσαλονίκη: Τζιόλα.
- Jayant, Ν. S., & Noll, Ρ. (1984). *Digital Coding of Waveforms: Principles and Applications to Speech and Video*. Prentice Hall.
- Mano, Μ. Μ., & Ciletti, Μ. D. (2012). *Digital Design (5 εκδ.)*. Pearson Education.
- MathWorks - DSP*. (2017). Ανάκτηση 2 10, 2017, από Getting Started with Matlab DSP: <https://www.mathworks.com/help/dsp/getting-started-with-dsp-system-toolbox.html>
- Proakis, J. G., & Salehi, Μ. (2008). *Digital Communications (5 εκδ.)*. McGraw-Hill Higher Education.
- Μουρτζόπουλος, Ι. (2012). *Ψηφιακή Τεχνολογία Ήχου*. Πάτρα: Πανεπιστήμιο Πατρών.

Εργαστηριακή Άσκηση 10: Μετασχηματισμός Z

1 Εισαγωγικά Στοιχεία

1.1 Σκοπός

Σκοπός της παρούσας εργαστηριακής άσκησης είναι η εξοικείωση των φοιτητών – σπουδαστών με τον μετασχηματισμό Z (ευθύ και αντίστροφο), καθώς και την ανάλυση ενός ρητού μετασχηματισμού Z σε μερικά κλάσματα.

1.2 Αναμενόμενα Μαθησιακά Αποτελέσματα

Με την ολοκλήρωση της παρούσας εργαστηριακής άσκησης, ο φοιτητής – σπουδαστής θα είναι σε θέση να:

- Γνωρίζει το μετασχηματισμό Z των βασικών σημάτων
- Γνωρίζει τις βασικές ιδιότητες του μετασχηματισμού Z
- Υπολογίζει:
 - Τον μετασχηματισμό Z ενός ΣΔΧ με τη συνάρτηση `ztrans` του Matlab
 - Τον αντίστροφο μετασχηματισμό Z με τη συνάρτηση `iztrans` του Matlab
- Αναλύει σε άθροισμα κλασμάτων έναν μετασχηματισμό Z που έχει ρητή μορφή

1.3 Εκτιμώμενη Διάρκεια

Ο εκτιμώμενος χρόνος που απαιτείται από τον φοιτητή – σπουδαστή για την ολοκλήρωση της παρούσας εργαστηριακής άσκησης είναι 2 διδακτικές ώρες.

1.4 Προαπαιτούμενα

Διάλεξη 8 της θεωρίας.

1.5 Εισαγωγή

Στην άσκηση αυτή θα ασχοληθούμε με τον ευθύ και αντίστροφο μετασχηματισμό Z για σήματα διακριτού χρόνου. Ο μετασχηματισμός Z αποτελεί το βασικότερο εργαλείο ανάλυσης των συστημάτων διακριτού χρόνου και αποτελεί επέκταση του μετασχηματισμού Fourier.

1.6 Κατάλογος Χρήσιμων Συναρτήσεων Matlab

Κατά την εκπόνηση των ασκήσεων χρησιμοποιούνται οι ακόλουθες συναρτήσεις – εντολές του Matlab:

| Συνάρτηση | Επεξήγηση |
|------------------|--|
| conv | Υπολογίζει τη συνέλιξη δύο σημάτων |
| syms | Δηλώνει συμβολικές μεταβλητές |
| subs | Αποδίδει τιμές σε μία συμβολική μεταβλητή μίας έκφρασης και στη συνέχεια υπολογίζει την τιμή της έκφρασης |
| ztrans | Υπολογίζει τον μετασχηματισμό Z μίας ακολουθίας |
| iztrans | Υπολογίζει τον αντίστροφο μετασχηματισμό Z |
| residue/residuez | Υπολογισμός πόλων, μηδενικών και υπολοίπου ενός μετασχηματισμού Z κατά την ανάλυσή του σε μερικά κλάσματα |
| pretty | Μετατρέπει μία συνάρτηση σε πιο ευπαρουσίαστη μορφή |
| figure | Δημιουργεί νέο παράθυρο για την απεικόνιση γραφικών παραστάσεων |
| subplot | Τμηματοποίηση ενός παραθύρου για γραφικές παραστάσεις σε γραμμές και στήλες, προκειμένου στο ίδιο παράθυρο να απεικονιστούν πολλαπλές γραφικές παραστάσεις |
| stem | Γραφική παράσταση διακριτών σημάτων – ακολουθιών |
| title | Τοποθετεί τίτλο σε μια γραφική παράσταση |

2 Μετασχηματισμός Z

Ο μετασχηματισμός Z μετασχηματίζει τις εξισώσεις διαφορών που περιγράφουν τα γραμμικά και χρονικά αμετάβλητα συστήματα διακριτού χρόνου σε αλγεβρικές εξισώσεις και έχει καθιερωθεί ως το βασικό εργαλείο μελέτης και σχεδίασης συστημάτων διακριτού χρόνου.

2.1 Ορισμός των Μετασχηματισμών Z και I-Z

Ο μετασχηματισμός Z ενός διακριτού σήματος $x(n)$ συμβολίζεται ως $X(z)$ και υπολογίζεται από τη σχέση:

$$X(z) = \sum_{n=-\infty}^{+\infty} x(n)z^{-n}$$

Αν γνωρίζουμε τον μετασχηματισμό Z $X(z)$ ενός διακριτού σήματος $x(n)$, μπορούμε να ανακατασκευάσουμε το αρχικό σήμα $x(n)$, υπολογίζοντας τον αντίστροφο μετασχηματισμό Z (I-Z) από τη σχέση:

$$x(n) = \frac{1}{2\pi j} \oint X(z)z^{n-1}dz$$

Οι πράξεις του ευθύ και του αντίστροφου μετασχηματισμού Z συμβολίζονται ως:

$$x(n) \xleftrightarrow{Z} X(z)$$

$$X(z) \xleftrightarrow{I-Z} x(n)$$

Στο Matlab οι μετασχηματισμοί Z και I-Z υπολογίζονται μέσω των συναρτήσεων `ztrans` και `iztrans` αντίστοιχα, αφού προηγουμένως έχουμε ορίσει συμβολικές μεταβλητές για τον χρόνο n και τη μιγαδική συχνότητα z .

Η συνάρτηση `ztrans` συντάσσεται ως εξής:

X=ztrans(x, z), όπου:

- x = συνάρτηση της οποίας θέλουμε να υπολογίσουμε το μετασχηματισμό Z

- z = μεταβλητή με βάση την οποία θα εκφράσουμε το μετασχηματισμό Z
- X = ο ζητούμενος μετασχηματισμός Z

Η συνάρτηση `iztrans` συντάσσεται ως εξής:

$x = \text{iztrans}(X, n)$, όπου:

- X = συνάρτηση της οποίας θέλουμε να υπολογίσουμε τον αντίστροφο μετασχηματισμό Z (δηλαδή να την επαναφέρουμε στο χρόνο)
- n = μεταβλητή με βάση την οποία θα εκφράσουμε τον αντίστροφο μετασχηματισμό Z (εδώ n είναι ο χρόνος)
- x = η ζητούμενη ακολουθία $x(n)$ στον χρόνο

Υπενθυμίζεται ότι πριν την χρήση των συναρτήσεων `ztrans()` και `iztrans()` πρέπει να έχουμε δηλώσει τις συμβολικές μεταβλητές n, z με την εντολή «`syms n z`».

Στον πίνακα που ακολουθεί καταγράφονται τα βασικότερα ζεύγη του μετασχηματισμού Z .

Πίνακας 4: Βασικότερα ζεύγη μετασχηματισμού Z

| Ακολουθία | Μετασχηματισμός Z | Περιοχή Σύγκλισης RoC |
|---|--|--------------------------|
| $\delta(n)$ | $\frac{1}{1}$ | $\forall z$ |
| $u(n)$ | $\frac{1}{1 - z^{-1}}$ | $ z > 1$ |
| $-u(-n - 1)$ | $\frac{1}{1 - z^{-1}}$ | $ z < 1$ |
| $a^n \cdot u(n)$ | $\frac{1}{1 - a \cdot z^{-1}}$ | $ z > a $ |
| $-b^n \cdot u(-n - 1)$ | $\frac{1}{1 - b \cdot z^{-1}}$ | $ z < b $ |
| $[a^n \cdot \sin(\omega_0 \cdot n)] \cdot u(n)$ | $\frac{(a \cdot \sin \omega_0) \cdot z^{-1}}{1 - (2 \cdot a \cdot \cos \omega_0) \cdot z^{-1} + a^2 \cdot z^{-2}}$ | $ z > a $ |
| $[a^n \cdot \cos(\omega_0 \cdot n)] \cdot u(n)$ | $\frac{1 - (a \cdot \cos \omega_0) \cdot z^{-1}}{1 - (2 \cdot a \cdot \cos \omega_0) \cdot z^{-1} + a^2 \cdot z^{-2}}$ | $ z > a $ |
| $n \cdot a^n \cdot u(n)$ | $\frac{a \cdot z^{-1}}{(1 - a \cdot z^{-1})^2}$ | $ z > a $ |
| $-n \cdot b^n \cdot u(-n - 1)$ | $\frac{b \cdot z^{-1}}{(1 - b \cdot z^{-1})^2}$ | $ z < b $ |

Άσκηση 1^η

- a) Έστω σήμα $x(n)=(1/2)^n u(n)+(1/3)^n u(n)$. Να βρείτε τον μετασχηματισμό Z του $x(n)$.
- b) Έστω σήμα $x(n)=2^n u(n)$. Να βρείτε τον μετασχηματισμό Z του $x(n)$.
- c) Έστω σήμα $x(n)=\{3, 5, 2, -3, 0, 7\}$ για $n=[-2, 3]$. Βρείτε τον μετασχηματισμό $X(Z)$.

| Κώδικας Matlab | Αποτέλεσμα |
|---|---|
| <p>A)</p> <pre>% ορίζω συμβολικές μεταβλητές z, n syms n z; % Κατασκευάζω το σήμα x(n) x=(1/2)^n+(1/3)^n; % Υπολογίζω το μετασχηματισμό Z X=ztrans(x, n, z) pretty(X)</pre> | $X = \frac{z}{z - 1/2} + \frac{z}{z - 1/3}$ |
| <p>B)</p> <pre>% ορίζω συμβολικές μεταβλητές z, n syms n z; % Κατασκευάζω το σήμα x(n) x=2^n; % Υπολογίζω το μετασχηματισμό Z X=ztrans(x, n, z) pretty(X)</pre> | $X = \frac{z}{z - 2}$ |
| <p>C)</p> <pre>n=-2:3; % Κατασκευάζω το σήμα x(n) x=[3, 5, 2, -3, 0, 7]; % Χρήση ορισμού X=sum(x.*(z.^(-n))) pretty(X)</pre> | $X = 5z^3 - \frac{3}{z} + 3z^2 + \frac{7}{z^3} + 2$ |

Άσκηση 2^η

Να βρείτε τις ακολουθίες που έχουν τους ακόλουθους μετασχηματισμούς Z και να υπολογιστούν οι 5 πρώτοι όροι τους:

c) $X_1(z) = \frac{z}{z-2}$

d) $X_2(z) = \frac{2z^2+7z}{z^2+z-2}$

| Κώδικας Matlab | Αποτέλεσμα |
|---|---|
| <pre>A) % ορίζω συμβολικές μεταβλητές n,z syms n z; % Κατασκευάζω τον X(z) X=z/(z-2); % Βρίσκω το σήμα x(n) x=iztrans(X,n) pretty(x) % Βρίσκω τους 5 πρώτους όρους της x(n) subs(x,[0:4])</pre> | <pre>x = 2^n n 2 ans = 1 2 4 8 16</pre> |
| <pre>B) % ορίζω συμβολικές μεταβλητές n,z syms n z; % Κατασκευάζω τον X(z) X=(2*z^2+7*z)/(z^2+z-2); % Βρίσκω το σήμα x(n)</pre> | <pre>x = 3 - (-2)^n n 3 - (-2) ans =</pre> |

| | |
|---|---------------|
| <pre>x=iztrans(X,n) pretty(x) % Βρίσκω τους 5 πρώτους όρους της x(n) subs(x,[0:4])</pre> | 2 5 -1 11 -13 |
|---|---------------|

2.2 Βασικές Ιδιότητες Μετασχηματισμού Z

Παρακάτω καταγράφονται οι βασικές ιδιότητες του μετασχηματισμού Z:

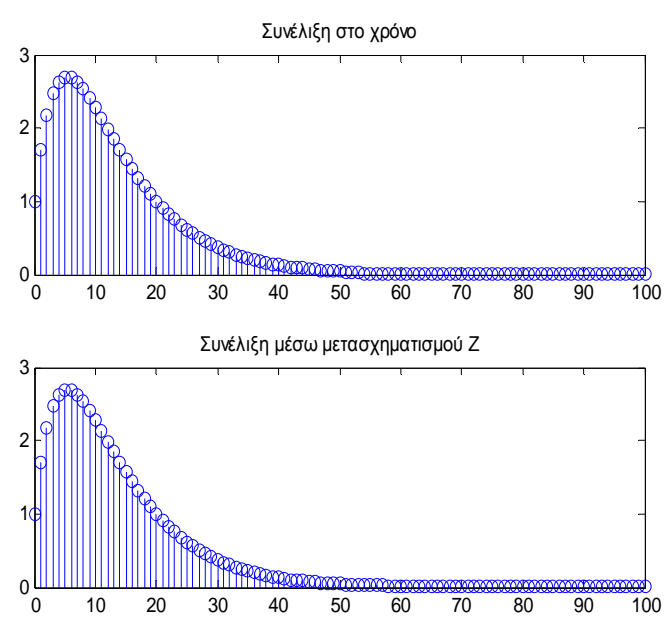
1. Γραμμικότητα: $ax_1(n) + bx_2(n) \xleftrightarrow{Z} aX_1(z) + bX_2(z)$
2. Μετατόπιση στον χρόνο: $x(n - n_0) \xleftrightarrow{Z} X(z)z^{-n_0}$
3. Μετατόπιση στη συχνότητα: $a^n x(n) \xleftrightarrow{Z} X\left(\frac{z}{a}\right)$
4. Αναδίπλωση: $x(-n) \xleftrightarrow{Z} X\left(\frac{1}{z}\right)$
5. Συζυγία: $x^*(n) \xleftrightarrow{Z} X^*(z^*)$
6. Παραγωγή στο πεδίο Z: $nx(n) \xleftrightarrow{Z} -z \frac{dX(z)}{dz}$
7. Συνέλιξη στο πεδίο του χρόνου: $x_1(n) * x_2(n) \xleftrightarrow{Z} X_1(z)X_2(z)$

Άσκηση 3^η

Έστω τα σήματα $x_1(n)=0.9^n$ και $x_2(n)=0.8^n$ για $n=[0, 50]$. Να αποδείξετε την ιδιότητα της συνέλιξης στο πεδίο του χρόνου του μετασχηματισμού Z.

Συγκρίνετε τα αποτελέσματα εμφανίζοντας κατάλληλα γραφήματα.

| | |
|-----------------------|-------------------|
| Κώδικας Matlab | Αποτέλεσμα |
|-----------------------|-------------------|

| | |
|---|--|
| <pre> % Αριστερό σκέλος ιδιότητας n=0:50; x1=0.9.^n; x2=0.8.^n; ya=conv(x1,x2); % Δεξιό σκέλος ιδιότητας syms n z x1=0.9.^n; x2=0.8.^n; X1=ztrans(x1,z); X2=ztrans(x2,z); Y=X1*X2; y=iztrans(Y,n); n=0:100; yb=subs(y,n); figure subplot(2,1,1),stem(0:100, ya) title('Συνέλιξη στο χρόνο') subplot(2,1,2),stem(0:100,yb) title('Συνέλιξη μέσω μετασχηματισμού Z') </pre> |  <p>Παρατήρηση: Οι γραφικές παραστάσεις είναι ίδιες</p> |
|---|--|

2.3 Ανάλυση σε Μερικά Κλάσματα

Πολύ συχνά ο μετασχηματισμός Z έχει τη μορφή ρητής συνάρτησης, δηλαδή κλάσματος πολυωνύμων της μορφής:

$$X(z) = \frac{B(z)}{A(z)} = \frac{b_m z^m + b_{m-1} z^{m-1} + \dots + b_1 z + b_0}{a_n z^n + a_{n-1} z^{n-1} + \dots + a_1 z + a_0}, \text{ όπου } a_i, b_i \in R$$

Προκειμένου να είναι ευκολότερος ο υπολογισμός του αρχικού σήματος $x(n)$ στο πεδίο του χρόνου, μέσω του αντίστροφου μετασχηματισμού Z, συνηθίζεται να αναλύουμε τη ρητή συνάρτηση $X(z)$ σε άθροισμα απλούστερων κλασμάτων (μερικά κλάσματα), όπου το καθένα κλάσμα έχει μία από τις μορφές του πίνακα της υποενότητας 2.1 και γνωρίζουμε την αρχική του ακολουθία. Η ανάλυση σε μερικά

κλάσματα μπορεί να γίνει θεωρώντας είτε θετικές είτε αρνητικές δυνάμεις για τη μιγαδική μεταβλητή z . Πιο συγκεκριμένα:

- Για θετικές δυνάμεις του z :

$$X(z) = \frac{B(z)}{A(z)} = \frac{R(1)}{z-P(1)} + \frac{R(2)}{z-P(2)} + \dots + \frac{R(n)}{z-P(n)} + K(z)$$

- Για αρνητικές δυνάμεις του z :

$$X(z) = \frac{B(z)}{A(z)} = \frac{R(1)}{1-P(1)z^{-1}} + \frac{R(2)}{1-P(2)z^{-1}} + \dots + \frac{R(n)}{1-P(n)z^{-1}} + K_1 + K_2z^{-1} + \dots$$

Όπου $P(i), R(i)$ και $K(i)$ είναι αντίστοιχα οι ρίζες του παρονομαστή, οι ρίζες του αριθμητή και το υπόλοιπο της ρητής συνάρτησης $X(z)$.

Στο Matlab για χρησιμοποιούμε τη συνάρτηση `residue()` για την ανάλυση σε κλάσματα με θετικές δυνάμεις της z και τη συνάρτηση `residuez()` για την ανάλυση σε κλάσματα με αρνητικές δυνάμεις της z . Η σύνταξη των εντολών έχει ως εξής:

- Για θετικές δυνάμεις της z : `[R, P, K] = residue(b,a)`
- Για αρνητικές δυνάμεις της z : `[R, P, K] = residuez(b,a)`

Όπου a και b είναι οι συντελεστές των δυνάμεων του παρονομαστή και αριθμητή αντίστοιχα, ενώ τα R, P, K είναι αντίστοιχα τα διανύσματα των μηδενικών, των πόλων και του υπόλοιπου του μετασχηματισμού.

Στο σημείο αυτό υπενθυμίζουμε ότι **πόλοι** χαρακτηρίζονται οι τιμές του z που μηδενίζουν τον παρονομαστή στο κλάσμα του μετασχηματισμού, ενώ **μηδενικά** οι τιμές του z που μηδενίζουν τον αριθμητή.

Άσκηση 4^η

Να γίνει ανάλυση σε μερικά κλάσματα του μετασχηματισμού $X(z) = \frac{2z^2+7z}{z^2+z-2}$ χρησιμοποιώντας:

- θετικές δυνάμεις της z και
- αρνητικές δυνάμεις της z

| Κώδικας Matlab | Αποτέλεσμα |
|--|--|
| <pre> a=[1 1 -2]; b=[2 7 0]; % με θετικές δυνάμεις [R1,P1,K1]=residue(b,a) % με αρνητικές δυνάμεις [R2,P2,K2]=residuez(b,a) </pre> | <p>R1 = [2 3]</p> <p>P1 = [-2 1]</p> <p>K1 = [2]</p> <p>R2 = [-1 3]</p> <p>P2 = [-2 1]</p> <p>K2 = [0]</p> <p><u>Δηλαδή:</u></p> <p>Θετικές δυνάμεις: $X(z) = \frac{2}{z-(-2)} + \frac{3}{z-1} + 2$</p> <p>Αρνητικές δυνάμεις: $X(z) = \frac{-1}{1-(-2)z^{-1}} + \frac{3}{1-1z^{-1}}$</p> |

Άσκηση 5^η

Να γίνει ανάλυση σε μερικά κλάσματα του μετασχηματισμού $X(z) = \frac{z^2+3z+1}{z^3+5z^2+2z-8}$ χρησιμοποιώντας:

- θετικές δυνάμεις της z και
- αρνητικές δυνάμεις της z

| Κώδικας Matlab | Αποτέλεσμα |
|--|---|
| <pre> a=[1 5 2 -8]; b=[0 1 3 1]; % με θετικές δυνάμεις [R1,P1,K1]=residue(b,a) % με αρνητικές δυνάμεις [R2,P2,K2]=residuez(b,a) </pre> | <pre> R1 = [0.5000 0.1667 0.3333] P1 = [-4.0000 -2.0000 1.0000] K1 = [] R2 = [-0.1250 -0.0833 0.3333] P2 = [-4.0000 -2.0000 1.0000] K2 = [-0.1250] </pre> <p><u>Δηλαδή:</u></p> <p>Θετικές δυνάμεις: $X(z) = \frac{0.5}{z-(-4)} + \frac{0.1667}{z-(-2)} + \frac{0.3333}{z-1}$</p> <p>Αρνητικές δυνάμεις: $X(z) = \frac{-0.125}{1-(-4)z^{-1}} + \frac{-0.0833}{1-(-2)z^{-1}} + \frac{0.3333}{1-z^{-1}} + (-0.125)$</p> |

Άσκηση 6^η

Να γίνει ανάλυση σε μερικά κλάσματα του μετασχηματισμού $X(z) = \frac{3z^3+8z^2+4}{z^2+5z+4}$

χρησιμοποιώντας:

- c) θετικές δυνάμεις της z και
- d) αρνητικές δυνάμεις της z

| Κώδικας Matlab | Αποτέλεσμα |
|-------------------------|----------------------------|
| <pre> a=[1 5 4]; </pre> | <pre> R1 = [20 3] </pre> |

| | |
|--|--|
| <pre> b=[3 8 0 4]; % με θετικές δυνάμεις [R1,P1,K1]=residue(b,a) % με αρνητικές δυνάμεις [R2,P2,K2]=residuez(b,a) </pre> | <pre> P1=[-4 -1] K1=[3 -7] R2=[1.2500 3.0000] P2=[-4 -1] K2=[-1.2500 1.0000] </pre> <p><u>Δηλαδή:</u></p> <p>Θετικές δυνάμεις: $X(Z) = \frac{20}{z-(-4)} + \frac{3}{z-(-1)} + 3z - 7$</p> <p>Αρνητικές δυνάμεις:</p> $X(z) = \frac{1.25}{1-(-4)z^{-1}} + \frac{3}{1-(-1)z^{-1}} + (-1.25) + 1z^{-1}$ |
|--|--|

3 Άλυτες Ασκήσεις

1. Έστω σήμα $x(n) = (1/2)^n u(n) + (2)^n u(n)$. Να βρείτε τον μετασχηματισμό Z του $x(n)$.
2. Έστω ο μετασχηματισμός $X(z) = z/(z-2)$. Να βρείτε την αρχική ακολουθία $x(n)$.
3. Έστω ο μετασχηματισμός $X(z) = 3/(z - 0.25 - 0.125z^{-1})$. Να γίνει ανάλυση σε μερικά κλάσματα και να βρεθούν οι πόλοι και τα μηδενικά του.

4 Επίλογος – Σύνδεση με τα Επόμενα

Στην παρούσα εργαστηριακή άσκηση καλύφθηκε ο υπολογισμός του ευθύ και του αντίστροφου μετασχηματισμού Z, μέσω των συναρτήσεων `ztrans` και `iztrans` του Matlab. Επίσης, παρουσιάστηκαν οι μετασχηματισμοί Z των βασικότερων σημάτων και οι βασικές ιδιότητες του μετασχηματισμού Z. Τέλος, ασχοληθήκαμε με τις συναρτήσεις `residue` και `residuez` για την ανάλυση ενός ρητού μετασχηματισμού Z

σε μερικά κλάσματα με θετικές και αρνητικές δυνάμεις της μιγαδικής μεταβλητής z , αντίστοιχα.

Στην επόμενη εργαστηριακή άσκηση θα καλυφθούν ζητήματα που αφορούν τον υπολογισμό της συνάρτησης μεταφοράς συστημάτων διακριτού χρόνου και τη μελέτη της ευστάθειάς τους μέσω του διαγράμματος πόλων – μηδενικών.

5 Βιβλιογραφία

Hayes , Μ. Η. (2002). *Ψηφιακή Επεξεργασία Σήματος*. (Α. Δημητριάδης, Μεταφρ.) Θεσσαλονίκη: Τζιόλα.

Hsu, Η. Ρ. (2002). *Αναλογικές και Ψηφιακές Επικοινωνίες*. Θεσσαλονίκη: Τζιόλα.

Jayant, Ν. Σ., & Noll, Ρ. (1984). *Digital Coding of Waveforms: Principles and Applications to Speech and Video*. Prentice Hall.

Mano, Μ. Μ., & Ciletti, Μ. D. (2012). *Digital Design (5 εκδ.)*. Pearson Education.

MathWorks - DSP. (2017). Ανάκτηση 2 10, 2017, από Getting Started with Matlab DSP: <https://www.mathworks.com/help/dsp/getting-started-with-dsp-system-toolbox.html>

Proakis, J. G., & Salehi, Μ. (2008). *Digital Communications (5 εκδ.)*. McGraw-Hill Higher Education.

Μουρτζόπουλος, Ι. (2012). *Ψηφιακή Τεχνολογία Ήχου*. Πάτρα: Πανεπιστήμιο Πατρών.

Εργαστηριακή Άσκηση 11: Συνάρτηση Μεταφοράς Συστήματος

1 Εισαγωγικά Στοιχεία

1.1 Σκοπός

Σκοπός της παρούσας εργαστηριακής άσκησης είναι η εξοικείωση των φοιτητών – σπουδαστών με τη μελέτη της συνάρτησης μεταφοράς συστημάτων διακριτού χρόνου, μέσω του διαγράμματος πόλων – μηδενικών.

1.2 Αναμενόμενα Μαθησιακά Αποτελέσματα

Με την ολοκλήρωση της παρούσας εργαστηριακής άσκησης, ο φοιτητής – σπουδαστής θα είναι σε θέση να:

- Υπολογίζει τη συνάρτηση μεταφοράς
- Κατασκευάζει το διάγραμμα πόλων – μηδενικών
- Εξετάζει την ευστάθεια ενός συστήματος

1.3 Εκτιμώμενη Διάρκεια

Ο εκτιμώμενος χρόνος που απαιτείται από τον φοιτητή – σπουδαστή για την ολοκλήρωση της παρούσας εργαστηριακής άσκησης είναι 2 διδακτικές ώρες.

1.4 Προαπαιτούμενα

Διάλεξη 9 της θεωρίας.

1.5 Εισαγωγή

Η συνάρτηση μεταφοράς ενός συστήματος διακριτού χρόνου αποτελεί τη μαθηματική έκφραση της συμπεριφοράς του συστήματος στο πεδίο Z. Συνήθως, η συνάρτηση μεταφοράς έχει ρητή μορφή, συνεπώς είναι απαραίτητη η γνώση της μεθόδου ανάλυσης σε μερικά κλάσματα, που περιγράψαμε στην προηγούμενη ενότητα, καθώς και των ιδιοτήτων του μετασχηματισμού Z.

1.6 Κατάλογος Χρήσιμων Συναρτήσεων Matlab

Κατά την εκπόνηση των ασκήσεων χρησιμοποιούνται οι ακόλουθες συναρτήσεις – εντολές του Matlab:

| Συνάρτηση | Επεξήγηση |
|-----------|---|
| syms | Δηλώνει συμβολικές μεταβλητές |
| roots | Βρίσκει τις ρίζες ενός πολυωνύμου |
| abs | Υπολογίζει το μέτρο (απόλυτη τιμή) |
| pole | Βρίσκει τους πόλους μίας συνάρτησης μεταφοράς |
| zero | Βρίσκει τα μηδενικά μίας συνάρτησης μεταφοράς |
| zplane | Κατασκευάζει το διάγραμμα πόλων – μηδενικών |
| pzmap | Κατασκευάζει το διάγραμμα πόλων – μηδενικών μίας συνάρτησης μεταφοράς |
| ztrans | Υπολογίζει τον μετασχηματισμό Z μίας ακολουθίας |
| tf | Υπολογίζει τη συνάρτηση μεταφοράς με θετικές δυνάμεις της z |
| filt | Υπολογίζει τη συνάρτηση μεταφοράς με αρνητικές δυνάμεις z |
| pretty | Μετατρέπει μία συνάρτηση σε πιο ευπαρουσίαστη μορφή |
| figure | Δημιουργεί νέο παράθυρο για την απεικόνιση γραφικών παραστάσεων |

2 Συνάρτηση Μεταφοράς Συστήματος

Η συνάρτηση μεταφοράς ενός συστήματος διακριτού χρόνου συμβολίζεται με $H(z)$ και είναι ο μετασχηματισμός Z της κρουστικής απόκρισης του συστήματος. Η μελέτη της συνάρτησης μεταφοράς ενός συστήματος μας επιτρέπει να κατανοήσουμε την ακριβή συμπεριφορά του συστήματος, ενώ για το σκοπό αυτόν χρησιμοποιούμε τον μετασχηματισμό Z και της ιδιότητές του, καθώς και τεχνικές όπως η ανάλυση της συνάρτησης μεταφοράς σε μερικά κλάσματα.

2.1 Μελέτη της Συνάρτησης Μεταφοράς ενός Συστήματος

Η συνάρτηση μεταφοράς $H(z)$ ενός συστήματος είναι ο μετασχηματισμός Z της κρουστικής του απόκρισης $h(n)$, δηλαδή είναι:

$$h(n) \stackrel{z}{\leftrightarrow} H(z)$$

Με δεδομένο ότι στα γραμμικά και αμετάβλητα στη μετατόπιση συστήματα διακριτού χρόνου η έξοδος, η είσοδος και η κρουστική απόκριση των συστημάτων συνδέονται μέσω της πράξης της γραμμικής συνέλιξης, μπορούμε μέσω της ιδιότητας της συνέλιξης του μετασχηματισμού Z να λύσουμε ως προς την $H(z)$ και να καταλήξουμε στη σχέση:

$$y(n) = x(n) * h(n) \stackrel{z}{\leftrightarrow} Y(z) = X(z)H(z) \Leftrightarrow H(z) = Y(z)/X(z)$$

Συνεπώς, η συνάρτηση μεταφοράς ενός συστήματος υπολογίζεται ως ο λόγος του μετασχηματισμού Z της εξόδου προς το μετασχηματισμό Z της εισόδου του συστήματος, δηλαδή:

$$H(z) = Y(z)/X(z)$$

Στην τελευταία σχέση, οι τιμές του z για τις οποίες ο παρονομαστής της $H(z)$ μηδενίζει λέγονται πόλοι του συστήματος, ενώ οι τιμές του z για τις οποίες ο αριθμητής της $H(z)$ μηδενίζει λέγονται μηδενικά του συστήματος. Οι πόλοι καθορίζουν τα όρια του πεδίου ορισμού της συνάρτησης μεταφοράς, αφού πρόκειται για τιμές που δεν επιτρέπεται να βρίσκονται στο πεδίο ορισμού ή αλλιώς στην περιοχή σύγκλισης ROC του συστήματος.

Οι τιμές των πόλων είναι εκείνες που χαρακτηρίζουν την ευστάθεια ή την αστάθεια του συστήματος. Πιο συγκεκριμένα, ένα σύστημα είναι ευσταθές όταν όλοι οι πόλοι του βρίσκονται εντός του μοναδιαίου κύκλου ή αλλιώς, ένα σύστημα είναι ευσταθές όταν το μέτρο του καθενός πόλου είναι μικρότερο από 1.

Όταν ένα σύστημα περιγράφεται μέσω μίας εξίσωσης διαφορών, τότε μπορούμε να κατασκευάσουμε πολύ εύκολα τη συνάρτηση μεταφοράς του μέσω της

συνάρτησης `tf` ή της `filt`. Η `tf` υπολογίζει τη συνάρτηση μεταφοράς με χρήση θετικών δυνάμεων της z , ενώ η `filt` με αρνητικές δυνάμεις της z . Η σύνταξή τους είναι:

- `H=tf(b, a, 1)`
- `H=filt(b, a, 1)`

Όπου `a` και `b` είναι τα διανύσματα με τους συντελεστές της εξόδου και της εισόδου της εξίσωσης διαφορών.

Επίσης, εφόσον έχουμε υπολογίσει τη συνάρτηση μεταφοράς με την `tf` ή `filt`, χρησιμοποιώντας τις εντολές `pole(H)`, `zero(H)` και `pzmap(H)` μπορούμε να βρούμε τους πόλους, τα μηδενικά και το διάγραμμα πόλων – μηδενικών της συνάρτησης μεταφοράς του συστήματος.

Άσκηση 1^η

Έστω το σύστημα με συνάρτηση μεταφοράς $H(z) = \frac{2z^2+3z}{z^2+0.4z-0.5}$. Να βρείτε τους πόλους και τα μηδενικά του συστήματος και να καθορίσετε αν το σύστημα είναι ευσταθές. Επίσης, να κατασκευάσετε το διάγραμμα πόλων – μηδενικών.

| Κώδικας Matlab | Αποτέλεσμα |
|----------------|------------|
|----------------|------------|


```

% Συντελεστές αριθμητή
b=[2 3 0];
% Συντελεστές παρονομαστή
a=[1 0.4 1];

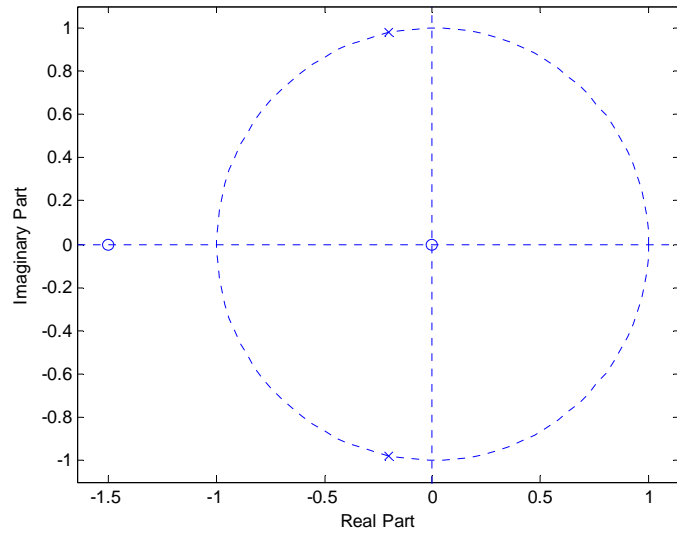
% Μηδενικά
zr=roots(b)

% Πόλοι
pl=roots(a)

% Έλεγχος ευστάθειας
metro_polwn=abs(pl)

% Διάγραμμα πόλων-μηδενικών
zplane(b,a)

```



$zr = [0 \ -1.5000]$

$pl = [-0.2000 + 0.9798i \quad -0.2000 - 0.9798i]$

$metro_polwn = [1 \ 1]$

Παρατήρηση: Οι πόλοι είναι ακριβώς πάνω στον μοναδιαίο κύκλο. Το σύστημα χαρακτηρίζεται ως κρίσιμο ή οριακά ευσταθές

Άσκηση 2^η

Έστω σύστημα με κρουσική απόκριση $h(n)=0.9^n u(n)$. Βρείτε τη συνάρτηση μεταφοράς του συστήματος και ελέγξτε την ευστάθειά του.

| Κώδικας Matlab | Αποτέλεσμα |
|---------------------------------|----------------|
| <pre> syms n z h=0.9.^n; </pre> | $H=z/(z-9/10)$ |

```

H=ztrans(h,z)
pretty(H)

% Συντελεστές αριθμητή
b=[1 0];
% Συντελεστές παρονομαστή
a=[1 -9/10];

% Μηδενικά
zr=roots(b)

% Πόλοι
pl=roots(a)

% Έλεγχος ευστάθειας
metro_polwn=abs(pl)

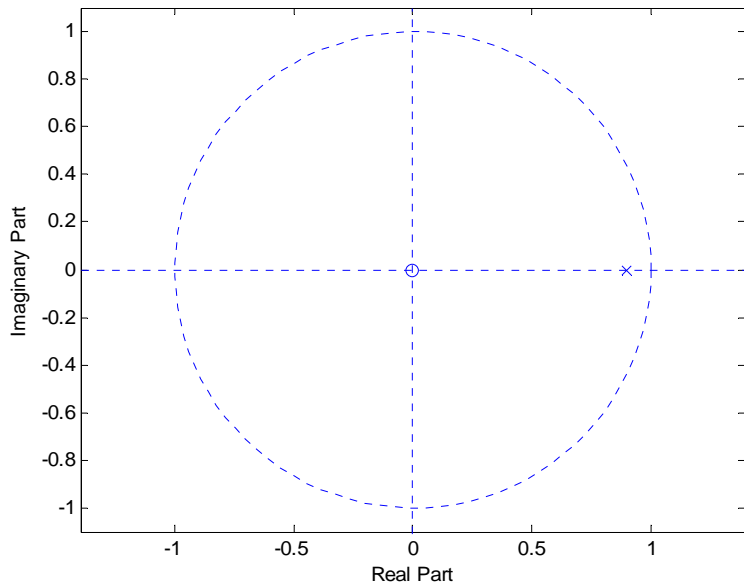
% Διάγραμμα πόλων-μηδενικών
zplane(b,a)

```

zr = 0

pl = 0.9000

metro_polwn = 0.9000



Παρατήρηση: Οι πόλοι είναι μέσα στον μοναδιαίο κύκλο. Το σύστημα χαρακτηρίζεται ως ευσταθές.

Άσκηση 3^η

Βρείτε τη συνάρτηση μεταφοράς του συστήματος και ελέγξτε την ευστάθειά του όταν περιγράφεται από την εξίσωση: $y(n+2) + 0.7y(n+1) + 0.1y(n) = 0.5x(n-1) + x(n)$. Η συνάρτηση μεταφοράς να εκφραστεί τόσο με θετικές όσο με αρνητικές δυνάμεις της z .

Κώδικας Matlab

Αποτέλεσμα

```

% Συντελεστές εξόδου y
a=[1 0.7 0.1 0];
% Συντελεστές εισόδου x
b=[0 0 1 0.5];

% με θετικές δυνάμεις
H1=tf(b,a,1)

p1=pole(H1)
metro_polwn1=abs(p1)
z1=zero(H1)
figure,pzmap(H1)

% με αρνητικές δυνάμεις
H2=filt(b,a,1)
p2=pole(H2)
metro_polwn2=abs(p2)
z2=zero(H2)
figure,pzmap(H2)

```

```

H1 =

      z + 0.5
-----
z^3 + 0.7 z^2 + 0.1 z

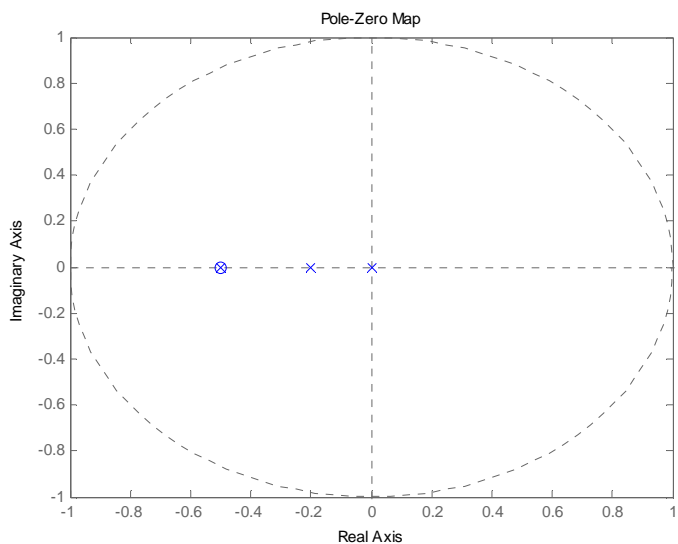
Sample time: 1 seconds
Discrete-time transfer function.

```

```
p1 = [ 0 -0.5000 -0.2000]
```

```
metro_polwn1 = [ 0 0.5000 0.2000]
```

```
z1 = [ -0.5000]
```



```

H2 =

      z^-2 + 0.5 z^-3
-----
1 + 0.7 z^-1 + 0.1 z^-2

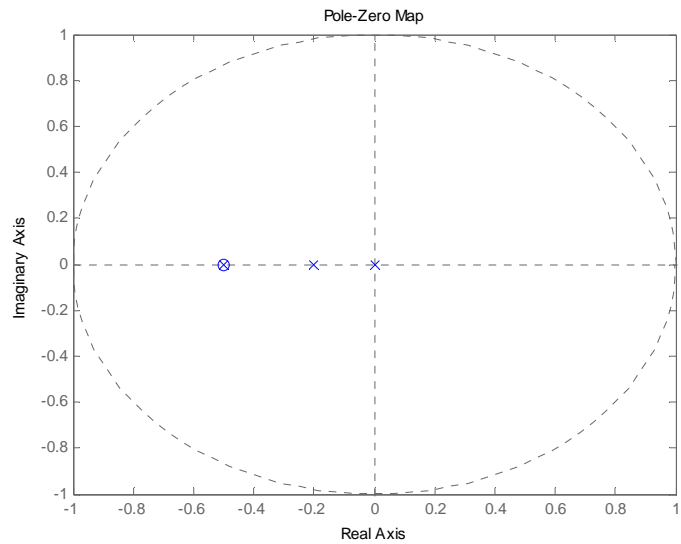
Sample time: 1 seconds
Discrete-time transfer function.

```

```
p2 = [ 0 -0.5000 -0.2000]
```

```
metro_polwn2 = [ 0 0.5000 0.2000]
```

$z2 = [-0.5000]$



Παρατήρηση: Προφανώς η χρήση τόσο των θετικών όσο των αρνητικών δυνάμεων της z μας δίνει ίδια αποτελέσματα για το σύστημά μας. Φαίνεται ότι οι πόλοι βρίσκονται μέσα στον μοναδιαίο κύκλο, συνεπώς το σύστημα είναι ευσταθές.

Άσκηση 4^η

Βρείτε τη συνάρτηση μεταφοράς του συστήματος και ελέγξτε την ευστάθειά του όταν περιγράφεται από την εξίσωση: $y(n) + 1.5y(n-1) + 0.5y(n-2) = x(n) + x(n-1)$. Η συνάρτηση μεταφοράς να εκφραστεί με θετικές και με αρνητικές δυνάμεις της z .

| Κώδικας Matlab | Αποτέλεσμα |
|----------------|------------|
|----------------|------------|

```

% Συντελεστές εξόδου y
a=[1 1.5 0.5];
% Συντελεστές εισόδου x
b=[1 1 0];

% με θετικές δυνάμεις
H1=tf(b,a,1)

p1=pole(H1)
metro_polwn1=abs(p1)
z1=zero(H1)
figure,pzmap(H1)

% με αρνητικές δυνάμεις
H2=filt(b,a,1)
p2=pole(H2)
metro_polwn2=abs(p2)
z2=zero(H2)
figure,pzmap(H2)

```

```

H1 =

      z^2 + z
-----
      z^2 + 1.5 z + 0.5

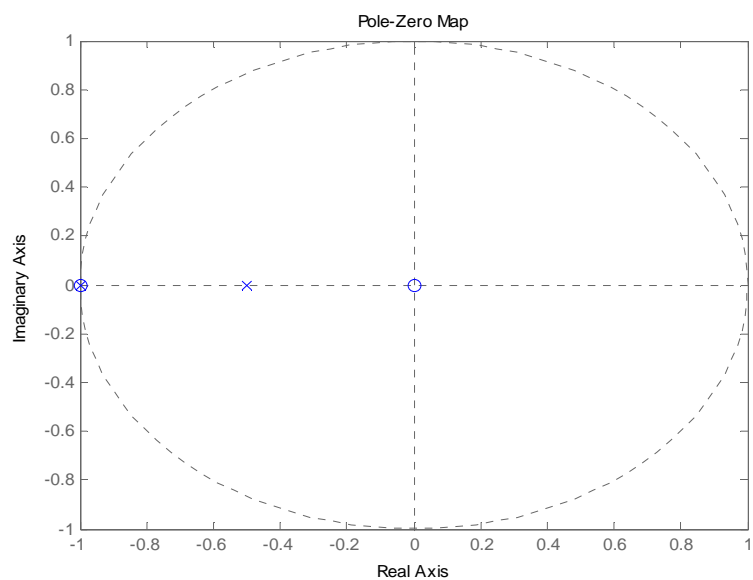
Sample time: 1 seconds
Discrete-time transfer function.

```

```
p1 = [ -1.0000 -0.5000]
```

```
metro_polwn1 = [ 1.0000 0.5000]
```

```
z1 = [ 0 -1]
```



```

H2 =

      1 + z^-1
-----
      1 + 1.5 z^-1 + 0.5 z^-2

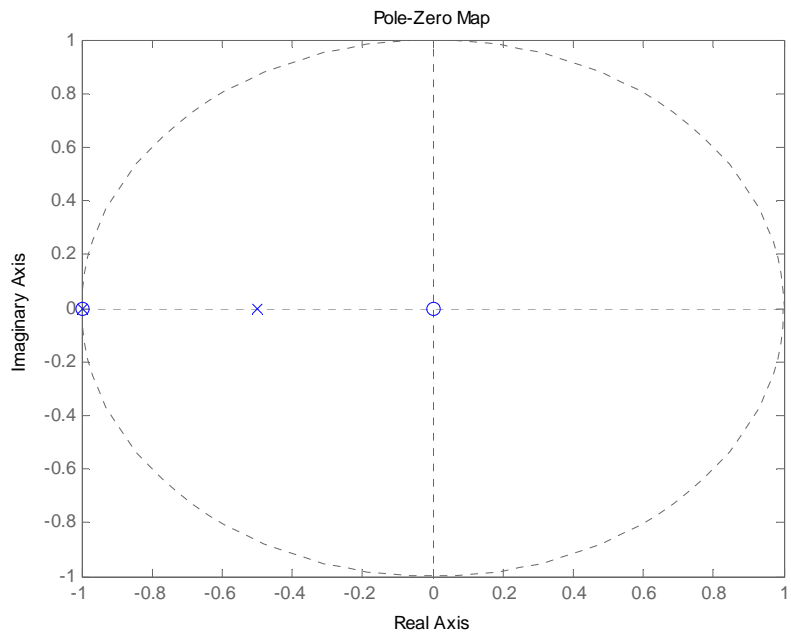
Sample time: 1 seconds
Discrete-time transfer function.

```

```
p2 = [ -1.0000 -0.5000]
```

metro_polwn2 =[1.0000 0.5000]

z2 =[0 -1]



Παρατήρηση: Ένας πόλος είναι ακριβώς πάνω στον μοναδιαίο κύκλο. Το σύστημα χαρακτηρίζεται ως κρίσιμο ή οριακά ευσταθές.

3 Άλυτες Ασκήσεις

1. Βρείτε τη συνάρτηση μεταφοράς του συστήματος, τους πόλους και τα μηδενικά και ελέγξτε την ευστάθειά του όταν περιγράφεται από την εξίσωση:

α) $y(n) + 1.5y(n-1) = 0.5x(n)$

β) $y(n) = 2x(n) - x(n-1)$

4 Επίλογος – Σύνδεση με τα Επόμενα

Στην παρούσα εργαστηριακή άσκηση καλύφθηκε η εύρεση της συνάρτησης μεταφοράς συστημάτων διακριτού χρόνου, ο υπολογισμός των πόλων και των μηδενικών της και η μελέτη της ευστάθειας του συστήματος μέσω του αντίστοιχου διαγράμματος πόλων – μηδενικών.

Στην επόμενη εργαστηριακή άσκηση θα καλυφθούν ζητήματα που αφορούν τη σχεδίαση ψηφιακών FIR και IIR φίλτρων.

5 Βιβλιογραφία

Hayes , Μ. Η. (2002). *Ψηφιακή Επεξεργασία Σήματος*. (Α. Δημητριάδης, Μεταφρ.) Θεσσαλονίκη: Τζιόλα.

Hsu, Η. Ρ. (2002). *Αναλογικές και Ψηφιακές Επικοινωνίες*. Θεσσαλονίκη: Τζιόλα.

Jayant, Ν. S., & Noll, Ρ. (1984). *Digital Coding of Waveforms: Principles and Applications to Speech and Video*. Prentice Hall.

Mano, Μ. Μ., & Ciletti, Μ. D. (2012). *Digital Design (5 εκδ.)*. Pearson Education.

MathWorks - DSP. (2017). Ανάκτηση 2 10, 2017, από Getting Started with Matlab DSP: <https://www.mathworks.com/help/dsp/getting-started-with-dsp-system-toolbox.html>

Proakis, J. G., & Salehi, Μ. (2008). *Digital Communications (5 εκδ.)*. McGraw-Hill Higher Education.

Μουρτζόπουλος, Ι. (2012). *Ψηφιακή Τεχνολογία Ήχου*. Πάτρα: Πανεπιστήμιο Πατρών.

Εργαστηριακή Άσκηση 12: Φίλτρα

1 Εισαγωγικά Στοιχεία

1.1 Σκοπός

Σκοπός της παρούσας εργαστηριακής άσκησης είναι η εξοικείωση των φοιτητών – σπουδαστών με τη μελέτη ψηφιακών FIR και IIR φίλτρων.

1.2 Αναμενόμενα Μαθησιακά Αποτελέσματα

Με την ολοκλήρωση της παρούσας εργαστηριακής άσκησης, ο φοιτητής – σπουδαστής θα είναι σε θέση να:

- Σχεδιάζει ψηφιακά FIR φίλτρα
- Σχεδιάζει ψηφιακά IIR φίλτρα

1.3 Εκτιμώμενη Διάρκεια

Ο εκτιμώμενος χρόνος που απαιτείται από τον φοιτητή – σπουδαστή για την ολοκλήρωση της παρούσας εργαστηριακής άσκησης είναι 2 διδακτικές ώρες.

1.4 Προαπαιτούμενα

Διάλεξη 10 της θεωρίας.

1.5 Εισαγωγή

Στην επεξεργασία σήματος, η λειτουργία ενός φίλτρου είναι να απομακρύνει τα ανεπιθύμητα μέρη ενός σήματος, όπως ένα τυχαίο θόρυβο ή να εξάγει χρήσιμα κομμάτια ενός σήματος, όπως οι συνιστώσες που βρίσκονται σε μια συγκεκριμένη περιοχή συχνοτήτων.

Η εμφάνιση των ψηφιακών φίλτρων άρχισε στα μέσα της δεκαετίας του 60, όταν οι ψηφιακοί υπολογιστές έγιναν προσιτοί για σοβαρή ερευνητική εργασία. Ο μετασχηματισμός όμως Z που είναι το βασικό εργαλείο των ψηφιακών φίλτρων ήταν γνωστός αρκετά χρόνια πριν. Τα ψηφιακά φίλτρα κάνουν την ίδια δουλειά που

κάνουν και τα αναλογικά, με τη διαφορά ότι η αναλογική προσέγγιση, σε μερικές περιπτώσεις είναι δύσκολο ή ακατόρθωτο να πραγματοποιηθεί.

Η χρήση των ψηφιακών φίλτρων έχει πολλά πλεονεκτήματα, όπως τέλεια αναπαραγωγή, εγγυημένη στάθμη λειτουργία, ευκολία στην αλλαγή των χαρακτηριστικών του φίλτρου, καθώς και δυνατότητα μοιρασμού χρόνου των ίδιων ψηφιακών κυκλωμάτων μεταξύ πολλών φίλτρων. Το τελευταίο και το πιο σπουδαίο πλεονέκτημα είναι η δημιουργία ψηφιακών φίλτρων σε ολοκληρωμένο κύκλωμα.

1.6 Κατάλογος Χρήσιμων Συναρτήσεων Matlab

Κατά την εκπόνηση των ασκήσεων χρησιμοποιούνται οι ακόλουθες συναρτήσεις – εντολές του Matlab:

| Συνάρτηση | Επεξήγηση |
|-----------|---|
| input | Εισαγωγή δεδομένων από τον χρήστη |
| fir1 | Κατασκευή γραμμικού FIR φίλτρου |
| fir2 | Κατασκευή μη γραμμικού FIR φίλτρου |
| freqz | Υπολογίζει την απόκριση συχνότητας συστήματος |
| abs | Υπολογίζει την απόλυτη τιμή ή μέτρο μιγαδικού |
| log10 | Υπολογίζει το λογάριθμο με βάση το 10 |
| cheby1 | Κατασκευάζει IIR φίλτρο τύπου Chebyshev-I |
| cheb1ord | Υπολογίζει την τάξη και τις συχνότητες IIR φίλτρου τύπου Chebyshev-I |
| cheb2ord | Υπολογίζει την τάξη και τις συχνότητες IIR φίλτρου τύπου Chebyshev-II |
| butter | Κατασκευάζει IIR φίλτρο τύπου Butterworth |
| plot | Δημιουργεί μία γραφική παράσταση |

| | |
|--------|--|
| legend | Τοποθετεί υπόμνημα σε μία γραφική παράσταση |
| title | Τοποθετεί τίτλο σε μια γραφική παράσταση |
| xlabel | Τοποθετεί τίτλο στον οριζόντιο άξονα γραφικής παράστασης |
| ylabel | Τοποθετεί τίτλο στον κάθετο άξονα μίας γραφικής παράστασης |
| grid | Προσθέτει – αφαιρεί πλέγμα στις γραφικές παραστάσεις, βελτιώνοντας την ανάγνωσή τους |

2 Σχεδίαση Ψηφιακών Φίλτρων

Τα ψηφιακά φίλτρα διακρίνονται σε αυτά που έχουν πεπερασμένη διάρκεια (FIR) και σε αυτά που έχουν άπειρη διάρκεια (IIR). Τα FIR είναι πάντα ευσταθή, ενώ τα IIR έχουν χαμηλότερο κόστος υλοποίησης (χαμηλότερη τάξη) σε σχέση με τα FIR ίδιων προδιαγραφών. Το βασικό εργαλείο για τη σχεδίαση και μελέτη των φίλτρων είναι ο μετασχηματισμός Z , όπου για την περιγραφή ενός συστήματος χρησιμοποιείται η συνάρτησης μεταφοράς του.

Οι βασικοί σχεδιαστικοί στόχοι κατά τη σχεδίαση ψηφιακών φίλτρων είναι η επίτευξη:

- Μικρής κυμάτωσης στη ζώνη διέλευσης
- Μεγάλης απόσβεσης στη ζώνη αποκοπής
- Μικρού εύρους της ζώνης μετάβασης
- Απλής κυκλωματικής υλοποίησης

2.1 Σχεδίαση FIR Φίλτρων

Τα FIR φίλτρα είναι φίλτρα με πεπερασμένη (όχι άπειρη) κρουστική απόκριση και διάρκεια, ενώ είναι πάντα ευσταθή.

Υπάρχουν 4 είδη FIR φίλτρων: α) χαμηλοπερατά (lowpass), β) υψηλερατά (highpass), γ) ζωνοπερατά (bandpass) και δ) ζωνοφρακτικά (bandstop). Ακολουθεί το βασικό τυπολόγιο αυτών των φίλτρων:

Χαμηλοπερατό: $2f_c T \text{sinc}(2\pi f_c T(n - M)), f_c = \frac{f_p + f_s}{2}$

Υψηλερατό: $-2f_c T \text{sinc}(2\pi f_c T(n - M)), f_c = \frac{f_p + f_s}{2}$

Ζωνοπερατό: $2f_{c2} T \text{sinc}(2\pi f_{c2} T(n - M)) - 2f_{c1} T \text{sinc}(2\pi f_{c1} T(n - M)),$

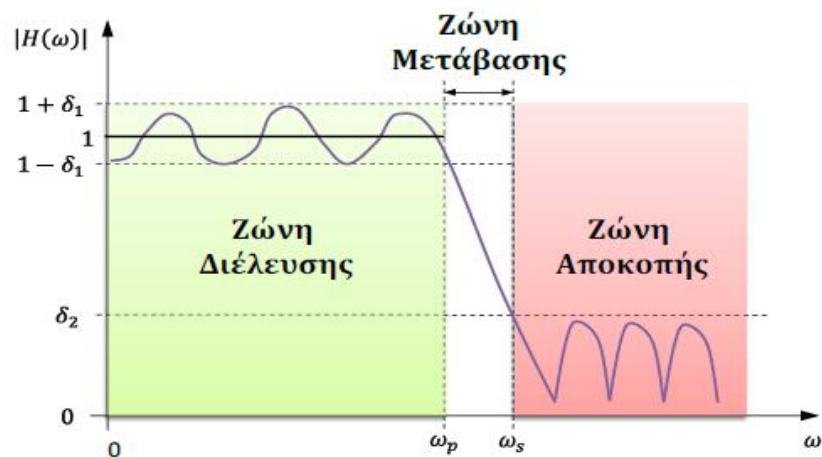
$$f_{c1} = \frac{f_{p1} + f_{s1}}{2} \text{ και } f_{c2} = \frac{f_{p2} + f_{s2}}{2}$$

Ζωνοφρακτικό: $-2f_{c2} T \text{sinc}(2\pi f_{c2} T(n - M)) + 2f_{c1} T \text{sinc}(2\pi f_{c1} T(n - M)),$

$$f_{c1} = \frac{f_{p1} + f_{s1}}{2} \text{ και } f_{c2} = \frac{f_{p2} + f_{s2}}{2}$$

Υπενθυμίζεται ότι $\text{sinc}(x) = \frac{\sin(x)}{x}$

Παρακάτω απεικονίζονται οι σχεδιαστικές παράμετροι ενός χαμηλοπερατού φίλτρου.



2.1.1 Φίλτρο FIR1

Το φίλτρο FIR1 αποτελεί ένα από τα ψηφιακά FIR φίλτρα που βασίζεται στη μέθοδο της γραμμικότητας. Με την εντολή `fir1` στο Matlab μπορούμε να σχεδιάσουμε όλων των ειδών τα φίλτρα. Εξ' ορισμού το φίλτρο είναι κανονικοποιημένο ώστε η απόκριση πλάτους του φίλτρου στην κεντρική συχνότητα της ζώνης διάβασης να είναι 0 dB.

Η εντολή σύνταξης του φίλτρου FIR1 παίρνει μία από τις εξής μορφές:

- `b = fir1(n,Wn)`
- `b = fir1 (n, Wn,'ftype')`
- `b = fir1 (n,Wn>window)`
- `b = fir1 (n,Wn,'ftype',window)`
- `b = fir1 (...,'normalization')`

Η εντολή `b = fir1(n,Wn)` αποτελεί ένα φίλτρο γραμμικής φάσης (βασίζεται στη μέθοδο παραθύρου Hamming) με κανονικοποιημένη συχνότητα αποκοπής Wn . Η Wn είναι ένας αριθμός που λαμβάνει τιμές από 0 έως 1, όπου το 1 παριστά τη συχνότητα Nyquist. Αν η Wn είναι ένα διάνυσμα δύο στοιχείων $Wn = [w1\ w2]$, τότε η `fir1` επιστρέφει ζωνοπερατό φίλτρο με ζώνη διάβασης $w1 \ll w2$. Αν η Wn είναι ένα διάνυσμα πολλών στοιχείων $Wn = [w1\ w2\ \dots\ wn]$, τότε η `fir1` επιστρέφει ένα ζωνοπερατό φίλτρο τάξης n με ζώνες διάβασης $0 < w1, w1 \ll w2, \dots, wn < 1$. Ως n ορίζεται είναι η τάξη του χαμηλοπερατού FIR φίλτρου.

Η εντολή `b = fir1(n,Wn,'ftype')` διευκρινίζει τον τύπο του φίλτρου, όπου `ftype` είναι:

- 'high' για υψηλοπερατό φίλτρο με συχνότητα αποκοπής Wn
- 'stop' για ζωνοδιαβατό φίλτρο, αν $Wn = [w1\ w2]$
- 'DC-1' για να γίνει η πρώτη ζώνη του πολυζωνοπερατού φίλτρου ζώνη διάβασης

- 'DC-0' για να γίνει η πρώτη ζώνη του πολυζωνοπερατού φίλτρου ζώνη αποκοπής.

Η εντολή `b = fir1(n,Wn>window)` χρησιμοποιεί ένα παράθυρο ορισμένο σε ένα διάλυμα με $n+1$ στοιχεία. Όταν δεν ορίζεται παράθυρο, η εντολή `fir1` χρησιμοποιεί ένα παράθυρο Hamming μήκους $n+1$.

Η εντολή `b = fir1(n,Wn,'ftype',window)` δέχεται τις παραμέτρους `ftype` και `window`.

Η εντολή `b = fir1(...,'normalization')` χρησιμοποιείται για να ορίσουμε αν το πλάτος του φίλτρου είναι κανονικοποιημένο. Η παράμετρος `normalization` μπορεί να είναι:

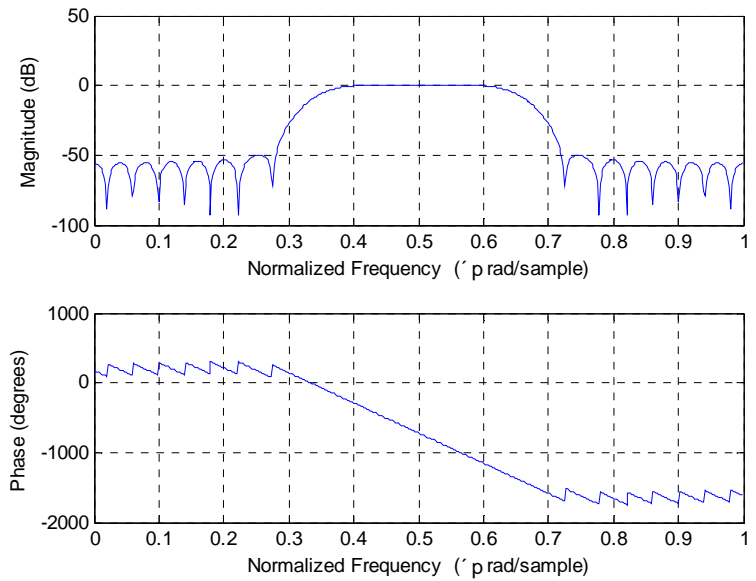
- 'scale' (default): Κανονικοποιεί το φίλτρο ώστε το πλάτος του φίλτρου στην κεντρική συχνότητα της ζώνης διάβασης να είναι 0 dB.
- 'noscale': δεν κανονικοποιεί το φίλτρο. Η εντολή `fir1` εφαρμόζει καθυστέρηση $n/2$.

Άσκηση 1^η

Να σχεδιαστεί ένα ζωνοδιαβατό FIR φίλτρο $N=48$ ης τάξης στις συχνότητες $f_1=0.35$ και $f_2=0.65$.

| Κώδικας Matlab | Αποτέλεσμα |
|----------------|------------|
|----------------|------------|

```
b = fir1(48,[0.35 0.65]);  
freqz(b,1,512)
```

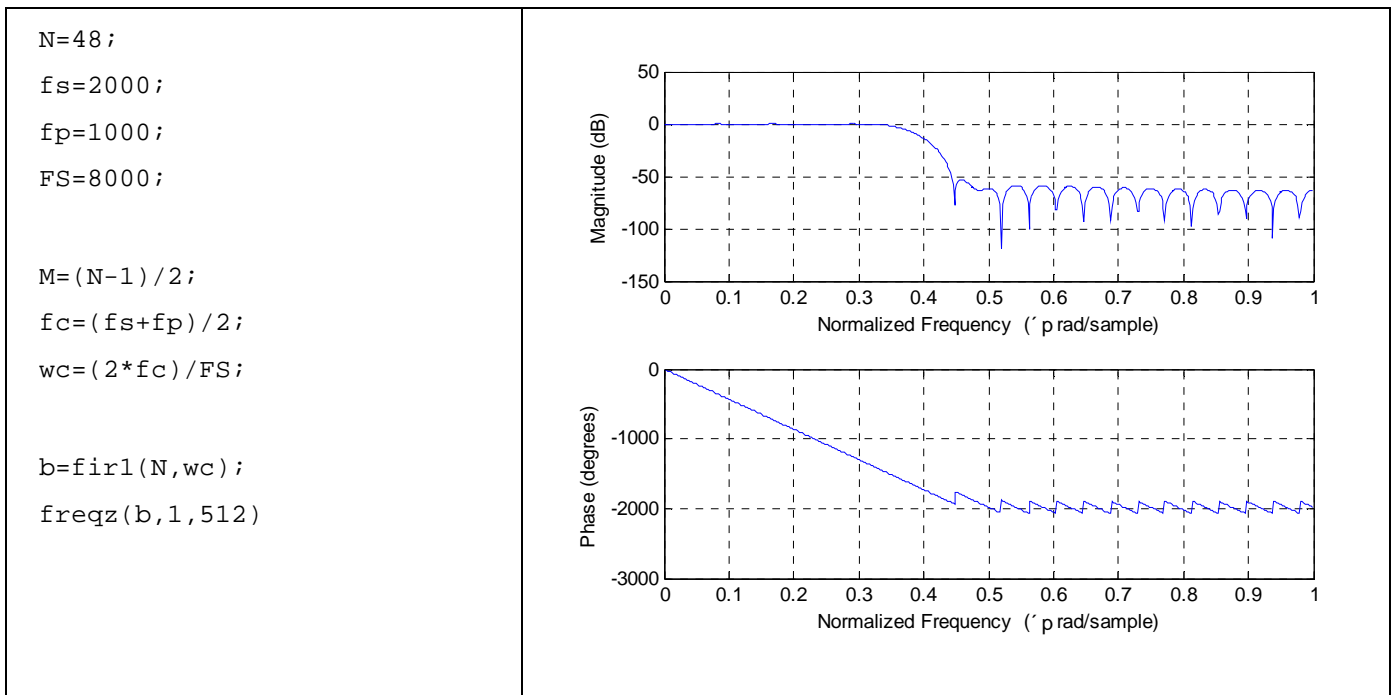


Άσκηση 2^η

Να σχεδιαστεί ένα χαμηλοπερατό FIR φίλτρο $N=48$ ης τάξης με τις εξής προδιαγραφές: συχνότητα αποκοπής $f_s=2000\text{Hz}$, συχνότητα διέλευσης $f_p=1000\text{Hz}$ και συχνότητα δειγματοληψίας $F_S=8000$.

Κώδικας Matlab

Αποτέλεσμα



2.1.2 Φίλτρο FIR2

Με την εντολή `fir2` σχεδιάζουμε ένα ψηφιακό FIR φίλτρο με αυθαίρετα σχηματοποιημένη συχνοτική απόκριση. Η `fir2` χρησιμοποιεί πάντα ένα φίλτρο περιπτής τάξης. Αυτό συμβαίνει γιατί για φίλτρα άρτιας τάξης η συχνοτική απόκριση στη συχνότητα Nyquist είναι απαραίτητως 0. Για τιμή άρτιας τάξης η εντολή `fir2` την αυξάνει κατά 1.

Η εντολή σύνταξης του φίλτρου FIR2 παίρνει μία από τις εξής μορφές:

- `b = fir2(n, f, m)`
- `b = fir2(n, f, m, window)`
- `b = fir2(n, f, m, npt)`
- `b = fir2(n, f, m, npt, window)`
- `b = fir2(n, f, m, npt, lap)`
- `b = fir2(n, f, m, npt, lap, window)`

Η εντολή `b = fir2(n, f, m)` επιστρέφει ένα διάνυσμα σειράς `b` που περιέχει `n+1` συνιστώσες ενός `n` τάξης FIR φίλτρου. Το `f` είναι το διάνυσμα που περιέχει τα στοιχεία της συχνότητας από 0 έως 1 (το 1 ανταποκρίνεται στην συχνότητα Nyquist).

Το πρώτο στοιχείο πρέπει να είναι 0 και το τελευταίο 1. Τα στοιχεία πρέπει να είναι σε αύξουσα σειρά. Το m είναι το διάνυσμα που περιέχει την επιθυμητή απόκριση μεγέθους στα στοιχεία συχνότητας που δηλώνει το διάνυσμα f . Το f και το m πρέπει να είναι του ίδιου μήκους. Χρησιμοποιώντας την `plot(f,m)` μπορούμε να δούμε το σχήμα του φίλτρου.

Η εντολή `b = fir2(n,f,m>window)` χρησιμοποιεί ένα παράθυρο που περιγράφεται από το διάνυσμα `window`. Το διάνυσμα `window` πρέπει να είναι μήκους $n+1$. Αν δεν ορίζεται παράθυρο, τότε η `fir2` χρησιμοποιεί ένα παράθυρο Hamming (`hamming`) μήκους $n+1$.

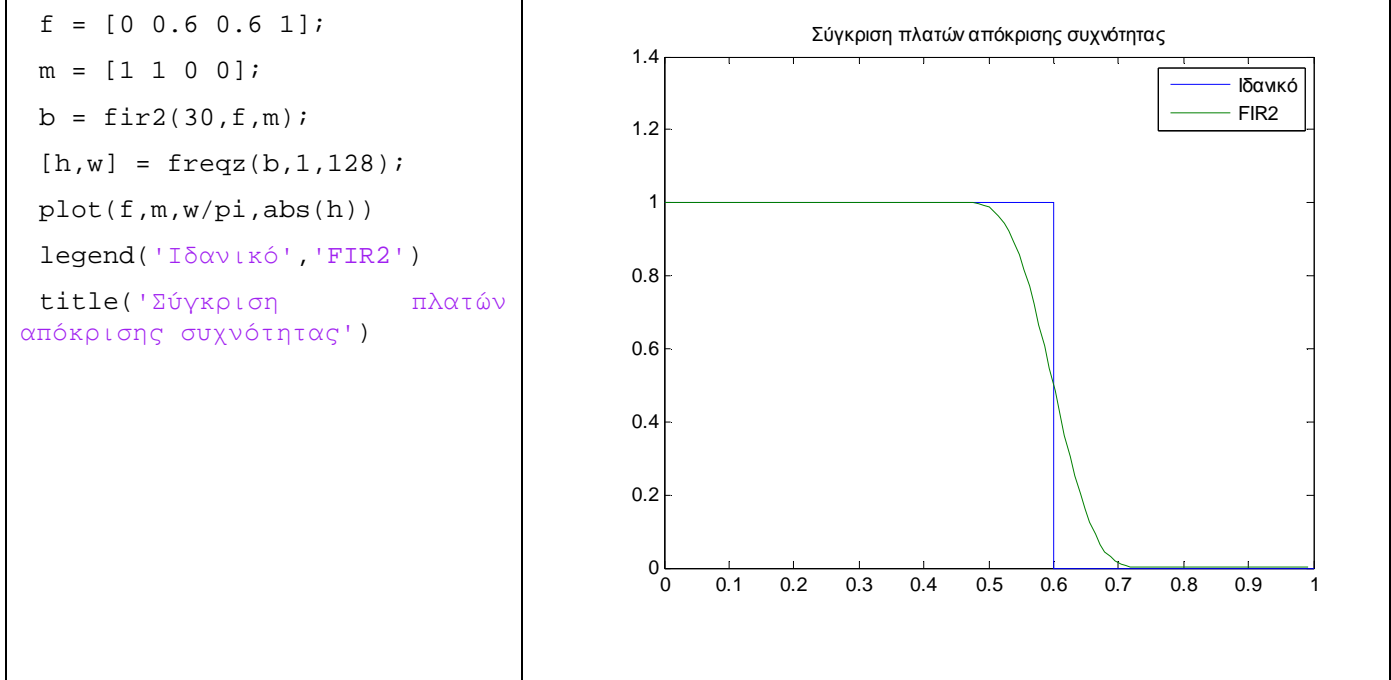
Η εντολή `b = fir2(n,f,m,npt)` ή `b = fir2(n,f,m,npt>window)` καθορίζει τον αριθμό των σημείων `npt` για το πλέγμα (`grid`) πάνω στο οποίο η `fir2` παρεμβάλλει την συχνοτική απόκριση, με ή χωρίς καθορισμό του παραθύρου.

Η εντολή `b = fir2(n,f,m,npt,lap)` και η `b = fir2(n,f,m,npt,lap>window)` προσδιορίζουν το μέγεθος της περιοχής και του `lap` που η `fir2` εισάγει γύρω από τα επαναλαμβανόμενα συχνοτικά σημεία, με ή χωρίς καθορισμό του παραθύρου.

Άσκηση 3^η

Να σχεδιαστεί ένα χαμηλοπερατό FIR φίλτρο $N=30$ ης τάξης και να παρασταθεί σε ένα διάγραμμα η επιθυμητή και η πραγματική συχνοτική απόκριση.

| Κώδικας Matlab | Αποτέλεσμα |
|----------------|------------|
|----------------|------------|



2.2 Σχεδίαση IIR Φίλτρων

Τα IIR φίλτρα είναι φίλτρα άπειρης κρουστικής απόκρισης (απόκριση αόριστης διάρκειας). Τα IIR φίλτρα υπερτερούν έναντι των FIR γιατί απαιτούν χαμηλότερη τάξη για τις ίδιες προδιαγραφές.

Το MATLAB περιέχει την εργαλειοθήκη επεξεργασίας σημάτων (Signal Processing Toolbox) που προσφέρει μεγάλο αριθμό ενσωματωμένων συναρτήσεων για τη σχεδίαση IIR φίλτρων που ικανοποιούν δοσμένες προδιαγραφές της απόκρισης συχνότητας. Για IIR φίλτρα προσφέρονται συναρτήσεις για καθεμιά από τις τεχνικές προσέγγισης του μέτρου της απόκρισης συχνότητας, Butterworth, Chebyshev-I, Chebyshev-II. Πιο συγκεκριμένα είναι:

- $[b, a] = \text{butter}(N, Wn)$
- $[b, a] = \text{cheby1}(N, Rp, Wn)$
- $[b, a] = \text{cheby2}(N, Rs, Wn)$

Οι συναρτήσεις αυτές χρησιμοποιούνται για να προσδιορίσουν απευθείας τη συνάρτηση συστήματος του ψηφιακού χαμηλοπερατού διαβατού φίλτρου τάξης N

με κανονικοποιημένη συχνότητα αποκοπής της ζώνης διάβασης W_n που είναι ένας αριθμός μεταξύ 0 και 1, οπότε η συχνότητα δειγματοληψίας υπονοείται ίση προς 2 Hz. Οι παράμετροι είναι:

- R_p για το φίλτρο Chebyshev-I είναι η κυμάτωση της ζώνης διάβασης σε dB
- R_s για το φίλτρο Chebyshev-II είναι η ελάχιστη εξασθένιση στη ζώνη αποκοπής σε dB
- Τα διανύσματα στήλης b , a που επιστρέφουν αυτές οι συναρτήσεις έχουν διάσταση $N+1$ και περιέχουν τους συντελεστές των πολυωνύμων του αριθμητή και παρονομαστή της συνάρτησης μεταφοράς του συστήματος σε αύξουσες αρνητικές δυνάμεις του z .
- Αφότου υπολογιστούν οι συντελεστές, η απόκριση συχνότητας μπορεί να υπολογιστεί χρησιμοποιώντας τη συνάρτηση $\text{freqz}(b,a,w)$, όπου w είναι ένα σύνολο ψηφιακών συχνοτήτων στην περιοχή $[0, \pi]$. Οι αντίστοιχες αναλογικές συχνότητες ω λαμβάνουν τιμές στο διάστημα $[0, 1]$.

Οι συναρτήσεις που χρησιμοποιούνται για την εύρεση μηδενικών και πόλων των IIR χαμηλοπερατών φίλτρων είναι:

- $[r, p, k] = \text{butter}(N, W_n)$
- $[r, p, k] = \text{cheby1}(N, R_p, W_n)$
- $[r, p, k] = \text{cheby2}(N, R_s, W_n)$

Οι συναρτήσεις αυτές επιστρέφουν τα διανύσματα r , p μήκους N που περιέχουν τα μηδενικά και τους πόλους της συνάρτησης μεταφοράς, ενώ το k περιέχει το συντελεστή κέρδους. Το διάγραμμα πόλων μηδενικών μπορεί να σχεδιαστεί με τη συνάρτηση $\text{zplane}(r,p)$.

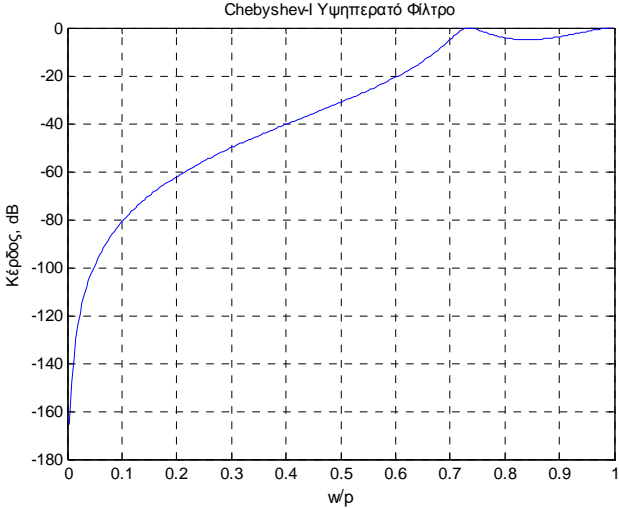
Για την κατασκευή ζωνοπερατού φίλτρου τάξης $2N$, μπορούμε να χρησιμοποιήσουμε τις ίδιες συναρτήσεις, ορίζοντας το W_n να είναι ένα διδιάστατο διάνυσμα όπου $W_n = [W_1 \ W_2]$, με ζώνη διάβασης $W_1 < \Omega < W_2$.

Για τη σχεδίαση ενός υπερπαρατού φίλτρου αρκεί να προστεθεί ως τελευταίο όρισμα της συνάρτησης το 'high' , πχ $[r, p, k] = \text{cheby1}(N, Rp, Wn, 'high')$.

Για τη σχεδίαση ενός ζωνοφρακτικού φίλτρου αρκεί να προστεθεί το όρισμα 'stop', πχ $[r, p, k] = \text{cheby2}(N, Rs, Wn, 'stop')$

Άσκηση 4^η

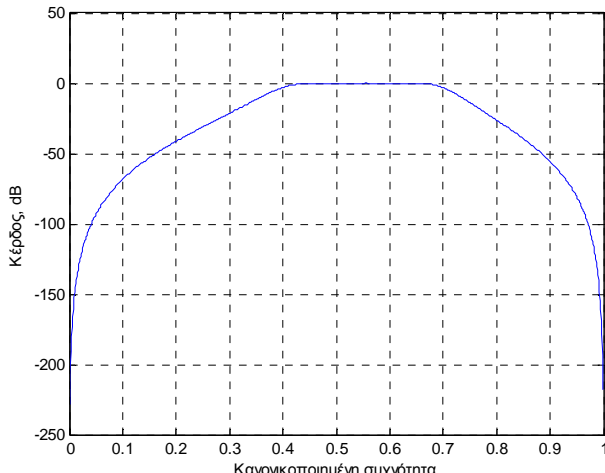
Να σχεδιαστεί ένα υπερπαρατό φίλτρο Chebyshev τύπου I. Οι προδιαγραφές σχεδίασης να προσδιορίζονται από το χρήστη κατά τη φάση εκτέλεσης.

| Κώδικας Matlab | Αποτέλεσμα |
|---|--|
| <pre> Wp = input('Κανονικοποιημένη συχνότητα διέλευσης = '); % 0.7 Ws = input('Κανονικοποιημένη συχνότητα αποκοπής = '); % 0.5 Rp = input('Κυμάτωση ζώνης διέλευσης σε dB = '); % 5 Rs = input('Ελάχιστη εξασθένιση ζώνης αποκοπής σε dB = '); % 20 [N,Wn] = cheblord(Wp,Ws,Rp,Rs); [b,a] = cheby1(N,Rp,Wn,'high'); [h,omega] = freqz(b,a,256); plot(omega/pi,20*log10(abs(h)));grid; xlabel('\omega/\pi'); ylabel('Κέρδος, dB'); title('Chebyshev-I Υψηπερατό Φίλτρο'); </pre> |  |

Άσκηση 5^η

Να γίνει σχεδίαση ενός ζωνοδιαβατού φίλτρου Butterworth όγδοης τάξης με συχνότητες αποκοπής 0.4 και 0.7 αντιστοίχως. Οι προδιαγραφές σχεδίασης προσδιορίζονται από το χρήστη κατά τη φάση εκτέλεσης.

Παρατήρηση: Από τα δεδομένα προκύπτει πως $N=8$, $W1=0.4$, $W2=0.7$

| Κώδικας Matlab | Αποτέλεσμα |
|--|---|
| <pre>% Butterworth Bandpass Filter Design N = input ('Τάξη φίλτρου N='); % 8 M=N/2; W1=input('Συχνότητα αποκοπής w1=');%0.4 W2=input('Συχνότητα αποκοπής w2=');%0.7 Wn=[W1 W2]; [b,a] = butter(M, Wn); w=0:0.01/pi:pi; h=freqz(b,a,w); gain=20*log10(abs(h)); plot(w/pi,gain); grid; xlabel('Κανονικοποιημένη συχνότητα'); ylabel('Κέρδος, dB');</pre> |  |

Άσκηση 6^η

Να υπολογιστεί η τάξη ενός υπερπαρατοχικού Chebyshev-II φίλτρου σε συχνότητα δειγματοληψίας 4KHz με προδιαγραφές: συχνότητα αποκοπής 1000Hz, συχνότητα διάβασης 600Hz, κυμάτωση της ζώνης διάβασης 1 dB και ελάχιστη εξασθένιση στη ζώνη αποκοπής 40 dB.

| Κώδικας Matlab | Αποτέλεσμα |
|--|--|
| <pre> FT = input('Συχνότητα δειγματοληψίας= '); %4000 Fp = input('Συχνότητα διέλευσης= '); %600 Fs = input('Συχνότητα αποκοπής= '); %1000 Rp = input('Κυμάτωση ζώνης διέλευσης= '); %1 Rs = input('Ελάχιστη εξασθένιση ζώνης αποκοπής= '); %40 Wp=Fp*2/FT; Ws=Fs*2/FT; [N, Wn]=cheb2ord(Wp, Ws, Rp, Rs) </pre> | <p>Συχνότητα δειγματοληψίας= 4000</p> <p>Συχνότητα διέλευσης= 600</p> <p>Συχνότητα αποκοπής= 1000</p> <p>Κυμάτωση ζώνης διέλευσης= 1</p> <p>Ελάχιστη εξασθένιση ζώνης αποκοπής= 40</p> <p>N = 5</p> <p>Wn = 0.5000</p> |

3 Άλυτες Ασκήσεις

1. Να υπολογιστεί η τάξη ενός υπερπαρατού Chebyshev-II φίλτρου σε συχνότητα δειγματοληψίας 2KHz με προδιαγραφές: συχνότητα αποκοπής 1000Hz, συχνότητα διάβασης 800Hz, κυμάτωση της ζώνης διάβασης 1 dB και ελάχιστη εξασθένιση στη ζώνη αποκοπής 30 dB.

4 Επίλογος – Σύνδεση με τα Επόμενα

Στην παρούσα εργαστηριακή άσκηση καλύφθηκε η υλοποίηση και μελέτη ψηφιακών FIR και IIR φίλτρων.

Στην επόμενη εργαστηριακή άσκηση θα καλυφθεί η διαδικασία δειγματοληψίας ενός αναλογικού σήματος για την κατασκευή του αντίστοιχου διακριτού σήματος, καθώς και η ανακατασκευή του αρχικού αναλογικού σήματος από το διακριτό σήμα με χρήση του θεωρήματος δειγματοληψίας.

5 Βιβλιογραφία

Hayes , Μ. Η. (2002). *Ψηφιακή Επεξεργασία Σήματος*. (Α. Δημητριάδης, Μεταφρ.) Θεσσαλονίκη: Τζιόλα.

Hsu, Η. Ρ. (2002). *Αναλογικές και Ψηφιακές Επικοινωνίες*. Θεσσαλονίκη: Τζιόλα.

Jayant, Ν. Σ., & Noll, Ρ. (1984). *Digital Coding of Waveforms: Principles and Applications to Speech and Video*. Prentice Hall.

Mano, Μ. Μ., & Ciletti, Μ. D. (2012). *Digital Design (5 εκδ.)*. Pearson Education.

MathWorks - DSP. (2017). Ανάκτηση 2 10, 2017, από Getting Started with Matlab DSP: <https://www.mathworks.com/help/dsp/getting-started-with-dsp-system-toolbox.html>

Proakis, J. G., & Salehi, Μ. (2008). *Digital Communications (5 εκδ.)*. McGraw-Hill Higher Education.

Μουρτζόπουλος, Ι. (2012). *Ψηφιακή Τεχνολογία Ήχου*. Πάτρα: Πανεπιστήμιο Πατρών.

Εργαστηριακή Άσκηση 13: Δειγματοληψία

1 Εισαγωγικά Στοιχεία

1.1 Σκοπός

Σκοπός της παρούσας εργαστηριακής άσκησης είναι η εξοικείωση των φοιτητών – σπουδαστών με τη διαδικασία δειγματοληψίας ενός αναλογικού σήματος σε διακριτό και την ανακατασκευή του αρχικού αναλογικού σήματος με χρήση του θεωρήματος της δειγματοληψίας.

1.2 Αναμενόμενα Μαθησιακά Αποτελέσματα

Με την ολοκλήρωση της παρούσας εργαστηριακής άσκησης, ο φοιτητής – σπουδαστής θα είναι σε θέση να:

- Γνωρίζει το θεώρημα δειγματοληψίας
- Μετατρέπει αναλογικά σήματα σε διακριτά σήματα
- Ανακατασκευάζει το αρχικό αναλογικό σήμα από το διακριτό σήμα
- Εκτιμάει την απαιτούμενη συχνότητα δειγματοληψίας, ώστε να μην εμφανίζονται φαινόμενα αναδίπλωσης.

1.3 Εκτιμώμενη Διάρκεια

Ο εκτιμώμενος χρόνος που απαιτείται από τον φοιτητή – σπουδαστή για την ολοκλήρωση της παρούσας εργαστηριακής άσκησης είναι 1 διδακτική ώρα.

1.4 Προαπαιτούμενα

Διάλεξη 11 της θεωρίας.

1.5 Εισαγωγή

Στην άσκηση αυτή θα ασχοληθούμε με τη δειγματοληψία αναλογικών σημάτων για την κατασκευή διακριτών σημάτων. Η διαδικασία της δειγματοληψίας αποτελεί

το πρώτο στάδιο της μετατροπής ενός σήματος από αναλογική σε ψηφιακή μορφή, έτσι ώστε να είναι εφικτή η περαιτέρω (ψηφιακή) επεξεργασία του.

1.6 Κατάλογος Χρήσιμων Συναρτήσεων Matlab

Κατά την εκπόνηση των ασκήσεων χρησιμοποιούνται οι ακόλουθες συναρτήσεις – εντολές του Matlab:

| Συνάρτηση | Επεξήγηση |
|-----------|--|
| length | Επιστρέφει το μήκος ενός πίνακα-διανύσματος |
| figure | Δημιουργεί νέο παράθυρο για την απεικόνιση γραφικών παραστάσεων |
| plot | Δημιουργεί μία γραφική παράσταση |
| subplot | Τμηματοποίηση ενός παραθύρου για γραφικές παραστάσεις σε γραμμές και στήλες, προκειμένου στο ίδιο παράθυρο να απεικονιστούν πολλαπλές γραφικές παραστάσεις |
| stem | Γραφική παράσταση διακριτών σημάτων – ακολουθιών |
| title | Τοποθετεί τίτλο σε μια γραφική παράσταση |
| axis | Ορίζει τα όρια τιμών του άξονα x και y σε μία γραφική παράσταση |
| grid | Προσθέτει – αφαιρεί πλέγμα στις γραφικές παραστάσεις, βελτιώνοντας την ανάγνωσή τους |
| xlabel | Τοποθετεί τίτλο στον οριζόντιο άξονα μίας γραφικής παράστασης |

2 Δειγματοληψία

Δειγματοληψία (sampling) ονομάζεται η διαδικασία με την οποία παράγουμε ένα διακριτό σήμα από ένα συνεχές σήμα. Το διακριτό σήμα προκύπτει από τη

καταγραφή των τιμών του συνεχούς σήματος σε μια σειρά από διακριτά και ισαπέχοντα σημεία – χρονικές στιγμές.

Θεώρημα δειγματοληψίας Nyquist ή Shannon: Ο ρυθμός δειγματοληψίας $f_s=1/T_s$ πρέπει να είναι τουλάχιστον διπλάσιος από την υψηλότερη ημιτονοειδή συχνοτική συνιστώσα που περιέχεται στο σήμα.

Ο ρυθμός δειγματοληψίας ή ρυθμός Nyquist μετράται σε Hz ή δείγματα ανά δευτερόλεπτο (sps). Επισημαίνεται ότι η δειγματοληψία που είναι χαμηλότερη από το ρυθμό Nyquist καλείται υποδειγματοληψία και μπορεί να παράγει ψευδή σήματα, λόγω του φαινομένου της αναδίπλωσης για τις συχνότητες που περιέχονται στο σήμα και είναι μεγαλύτερες από το μισό της συχνότητας Nyquist.

Η ανακατασκευή του αρχικού σήματος από το διακριτό σήμα επιτυγχάνεται με βάση το θεώρημα δειγματοληψίας μέσω της σχέσης:

$$x_a(t) = \sum_{n=-\infty}^{+\infty} x_a(nT_s) \text{sinc}(f_s(t - nT_s))$$

Όπου $x_a(t)$ είναι το ανακτημένο αναλογικό σήμα, $x_a(nT_s)$ το διακριτό σήμα και $\text{sinc}(x) = \frac{\sin(\pi x)}{\pi x}$

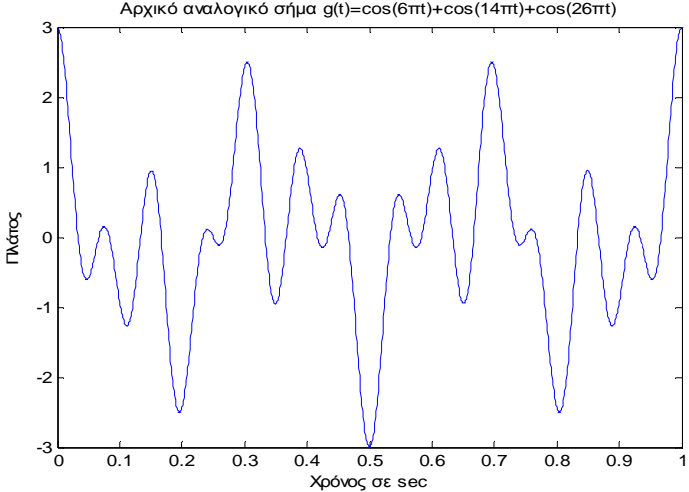
Άσκηση 1^η

Έστω τα αναλογικά σήματα $g_1(t)=\cos(6\pi t)$, $g_2(t)=\cos(14\pi t)$, $g_3(t)=\cos(26\pi t)$ και το άθροισμά τους $g(t)=g_1(t)+g_2(t)+g_3(t)$, όπου t είναι ο αναλογικός χρόνος με $t \in [0, 1]$;

- f) Να σχεδιάσετε το αναλογικό σήμα $g(t)$ χρησιμοποιώντας ως συχνότητα δειγματοληψίας την $f_s=1000\text{Hz}$
- g) Δημιουργήστε διακριτό σήμα με δειγματοληψία του αναλογικού σήματος $g(t)$ χρησιμοποιώντας ως συχνότητα δειγματοληψίας την $f_s=20\text{Hz}$ και στη συνέχεια προσπαθήστε να ανακατασκευάσετε το αρχικό αναλογικό σήμα. Σχεδιάστε σε ένα παράθυρο τις γραφικές παραστάσεις του αρχικού

αναλογικού σήματος $g(t)$, του διακριτού και του ανακατασκευασμένου αναλογικού σήματος. Τι παρατηρείται για το αρχικό αναλογικό σήμα και το ανακατασκευασμένο σήμα;

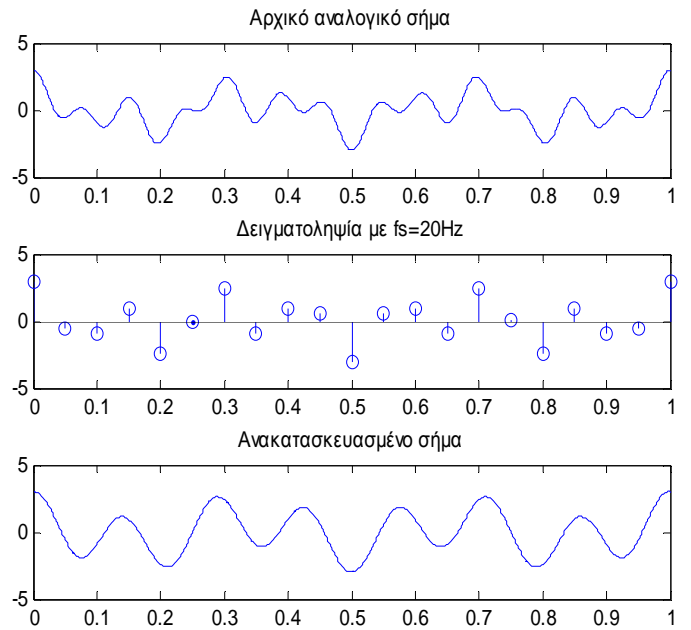
- h) Εκτελέστε την ίδια διαδικασία όπως στο προηγούμενο ερώτημα χρησιμοποιώντας ως συχνότητα δειγματοληψίας την $fs=30\text{Hz}$. Τι παρατηρείται για το αρχικό αναλογικό σήμα και το ανακατασκευασμένο σήμα;

| Κώδικας Matlab | Αποτέλεσμα |
|--|---|
| <pre>A) clear all, clc; % Αναλογικό σήμα fs=1000;Ts=1/fs;t=0:Ts:1; g1=cos(6*pi*t); g2=cos(14*pi*t); g3=cos(26*pi*t); gt=g1+g2+g3; plot(t,gt) xlabel('Χρόνος σε sec') ylabel('Πλάτος') title('Αρχικό αναλογικό σήμα g(t)=cos(6πt)+cos(14πt)+cos(26πt)')</pre> |  |

B)

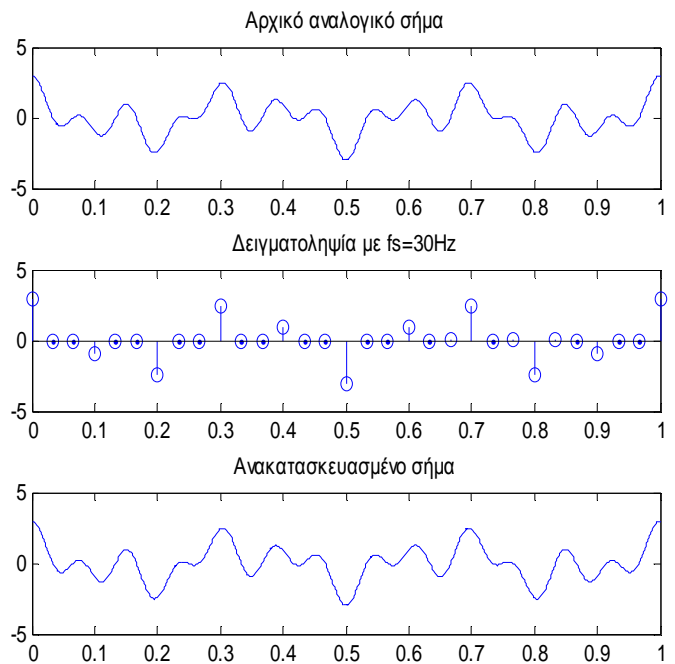
```
% Θυμίζουμε ότι
fnyquist=2*fmax=2*13=26Hz

% Δειγματοληψία με fs=20<fnyquist
fs1=20;Ts1=1/fs1;n=0:Ts1:1;
gn1=cos(6*pi*n);
gn2=cos(14*pi*n);
gn3=cos(26*pi*n);
gn=gn1+gn2+gn3;
% Ανακατασκευή αρχικού σήματος
for i=1:length(gt)
    grec(i)=sum(gn.*sinc(fs1*(t(i)-n)));
end
figure
subplot(3,1,1),plot(t,gt)
title('Αρχικό αναλογικό σήμα')
subplot(3,1,2),stem(n,gn)
str=strcat('Δειγματοληψία με
fs=',num2str(fs1),'Hz'); title(str)
subplot(3,1,3),plot(t,grec)
title('Ανακατασκευασμένο σήμα')
```



C)

```
% Δειγματοληψία με fs=30<fnyquist
fs1=30;Ts1=1/fs1;n=0:Ts1:1;
gn1=cos(6*pi*n);
gn2=cos(14*pi*n);
gn3=cos(26*pi*n);
gn=gn1+gn2+gn3;
% Ανακατασκευή αρχικού σήματος
for i=1:length(gt)
    grec(i)=sum(gn.*sinc(fs1*(t(i)-n)));
end
figure
subplot(3,1,1),plot(t,gt)
title('Αρχικό αναλογικό σήμα')
subplot(3,1,2),stem(n,gn)
str=strcat('Δειγματοληψία με
fs=',num2str(fs1),'Hz'); title(str)
subplot(3,1,3),plot(t,grec)
title('Ανακατασκευασμένο σήμα')
```

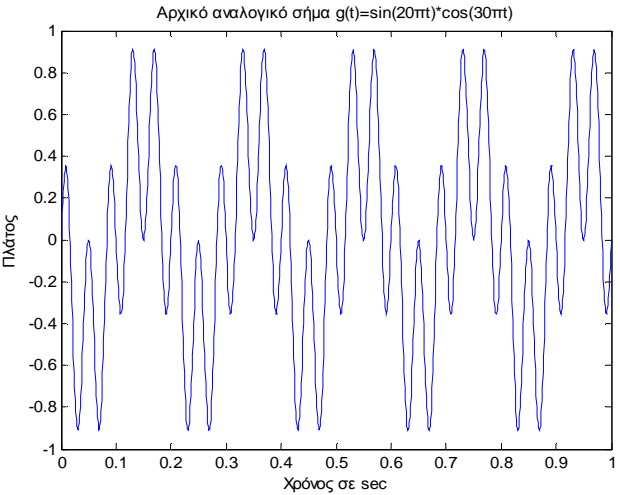
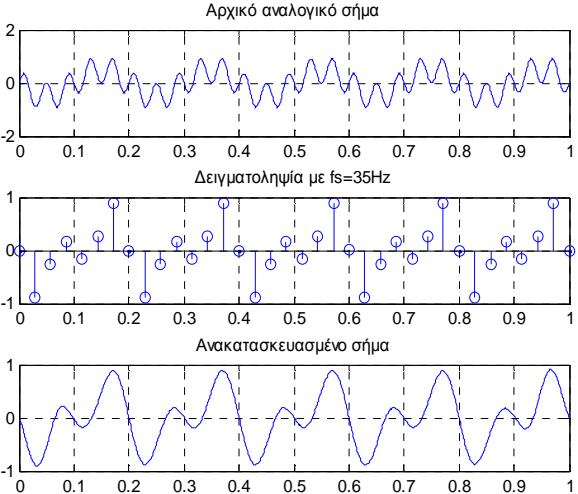


Άσκηση 2^η

Έστω τα αναλογικά σήματα $g_1(t)=\sin(20\pi t)$, $g_2(t)=\cos(30\pi t)$ και το γινόμενό τους $g(t)=g_1(t)*g_2(t)$, όπου t είναι ο αναλογικός χρόνος με $t \in [0, 1]$;

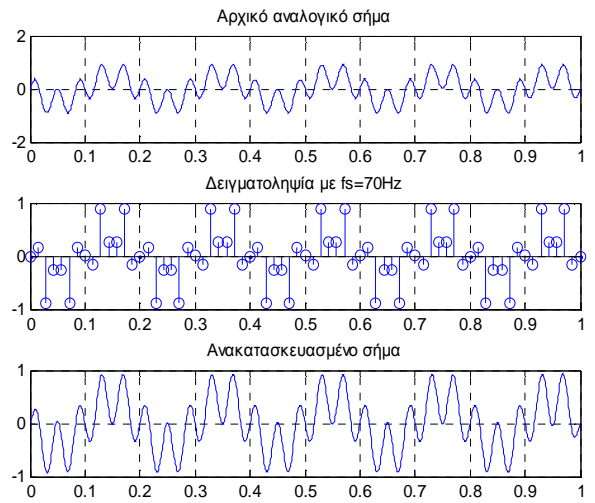
- Να σχεδιάσετε το αναλογικό σήμα $g(t)$ χρησιμοποιώντας ως συχνότητα δειγματοληψίας την $fs=1000\text{Hz}$
- Δημιουργήστε διακριτό σήμα με δειγματοληψία του αναλογικού σήματος $g(t)$ χρησιμοποιώντας ως συχνότητα δειγματοληψίας την $fs=35\text{Hz}$ και στη συνέχεια προσπαθήστε να ανακατασκευάσετε το αρχικό αναλογικό σήμα. Σχεδιάστε σε ένα παράθυρο τις γραφικές παραστάσεις του αρχικού αναλογικού σήματος $g(t)$, του διακριτού και του ανακατασκευασμένου αναλογικού σήματος. Τι παρατηρείται για το αρχικό αναλογικό σήμα και το ανακατασκευασμένο σήμα;

c) Εκτελέστε την ίδια διαδικασία όπως στο προηγούμενο ερώτημα χρησιμοποιώντας ως συχνότητα δειγματοληψίας την $f_s=70\text{Hz}$. Τι παρατηρείται για το αρχικό αναλογικό και το ανακατασκευασμένο σήμα;

| Κώδικας Matlab | Αποτέλεσμα |
|--|--|
| <pre> A) clear all, clc; % Αναλογικό σήμα fs=1000;Ts=1/fs;t=0:Ts:1; gt=sin(20*pi*t).*cos(30*pi*t); plot(t,gt), xlabel('Χρόνος σε sec'), ylabel('Πλάτος') title('Αρχικό αναλογικό σήμα g(t)=sin(20πt)*cos(30πt)') </pre> |  <p>Αρχικό αναλογικό σήμα $g(t)=\sin(20\pi t)\cos(30\pi t)$</p> |
| <pre> B) % Δειγματοληψία με fs=35 fs1=35;Ts1=1/fs1;n=0:Ts1:1; gn=sin(20*pi*n).*cos(30*pi*n); % Ανακατασκευή αρχικού σήματος for i=1:length(gt) grec(i)=sum(gn.*sinc(fs1*(t(i)-n))); end figure subplot(3,1,1),plot(t,gt) title('Αρχικό αναλογικό σήμα') axis([0 1 -2 2]),grid on subplot(3,1,2),stem(n,gn),grid on str=strcat('Δειγματοληψία με fs=',num2str(fs1),'Hz'); title(str) subplot(3,1,3),plot(t,grec), title('Ανακατασκευασμένο σήμα'),grid on </pre> |  <p>Αρχικό αναλογικό σήμα</p> <p>Δειγματοληψία με $f_s=35\text{Hz}$</p> <p>Ανακατασκευασμένο σήμα</p> |

C)

```
% Δειγματοληψία με fs=70
fs1=70;Ts1=1/fs1;n=0:Ts1:1;
gn=sin(20*pi*n).*cos(30*pi*n);
% Ανακατασκευή αρχικού σήματος
for i=1:length(gt)
    grec(i)=sum(gn.*sinc(fs1*(t(i)-n)));
end
figure
subplot(3,1,1),plot(t,gt)
title('Αρχικό αναλογικό σήμα')
axis([0 1 -2 2]),grid on
subplot(3,1,2),stem(n,gn),grid on
str=strcat('Δειγματοληψία με
fs=',num2str(fs1),'Hz'); title(str)
subplot(3,1,3),plot(t,grec),
title('Ανακατασκευασμένο σήμα'),grid on
```



Παρατήρηση:

Αν αξιοποιούσαμε την τριγωνομετρική σχέση $\sin A \cos B = \frac{1}{2}[\sin(A-B) + \sin(A+B)]$, το σήμα $g(t) = \sin(20\pi t) \cos(30\pi t)$ θα μπορούσε να γραφτεί στη μορφή $g(t) = \frac{1}{2}[\sin(-10\pi t) + \sin(50\pi t)]$, από την οποία φαίνεται ότι η μεγαλύτερη συχνότητα είναι η 25Hz και η ελάχιστη συχνότητα δειγματοληψίας (ρυθμός Nyquist) πρέπει να είναι $f_s = 2 \cdot 25 = 50\text{Hz}$

3 Άλυτες Ασκήσεις

1. Έστω τα αναλογικά σήματα $g_1(t) = \sin(20\pi t)$, $g_2(t) = \cos(30\pi t)$ και το άθροισμά τους $g(t) = g_1(t) + g_2(t)$, όπου t είναι ο αναλογικός χρόνος με $t \in [0, 1]$;
 - a) Σχεδιάστε το σήμα $g(t)$ με συχνότητα δειγματοληψίας $f_s = 1000\text{Hz}$
 - b) Δημιουργήστε διακριτό σήμα με δειγματοληψία του αναλογικού σήματος $g(t)$ χρησιμοποιώντας ως συχνότητα δειγματοληψίας την $f_s = 35\text{Hz}$ και στη συνέχεια προσπαθήστε να ανακατασκευάσετε το αρχικό αναλογικό σήμα. Σχεδιάστε σε ένα παράθυρο τις γραφικές παραστάσεις του αρχικού αναλογικού σήματος $g(t)$, του διακριτού και του ανακατασκευασμένου

αναλογικού σήματος. Τι παρατηρείται για το αρχικό αναλογικό σήμα και το ανακατασκευασμένο σήμα;

- c) Εκτελέστε την ίδια διαδικασία όπως στο προηγούμενο ερώτημα χρησιμοποιώντας ως συχνότητα δειγματοληψίας την $f_s=70\text{Hz}$. Τι παρατηρείτε για το αρχικό αναλογικό και το ανακατασκευασμένο σήμα;

4 Επίλογος

Στην παρούσα εργαστηριακή άσκηση καλύφθηκε η διαδικασία δειγματοληψίας ενός αναλογικού σήματος για την κατασκευή του αντίστοιχου διακριτού σήματος, καθώς και η ανακατασκευή του αρχικού αναλογικού σήματος από το διακριτό σήμα με χρήση του θεωρήματος δειγματοληψίας.

5 Βιβλιογραφία

Hayes , Μ. Η. (2002). *Ψηφιακή Επεξεργασία Σήματος*. (Α. Δημητριάδης, Μεταφρ.) Θεσσαλονίκη: Τζιόλα.

Hsu, Η. Ρ. (2002). *Αναλογικές και Ψηφιακές Επικοινωνίες*. Θεσσαλονίκη: Τζιόλα.

Jayant, Ν. Σ., & Noll, Ρ. (1984). *Digital Coding of Waveforms: Principles and Applications to Speech and Video*. Prentice Hall.

Mano, Μ. Μ., & Ciletti, Μ. D. (2012). *Digital Design (5 εκδ.)*. Pearson Education.

MathWorks - DSP. (2017). Ανάκτηση 2 10, 2017, από Getting Started with Matlab DSP: <https://www.mathworks.com/help/dsp/getting-started-with-dsp-system-toolbox.html>

Proakis, J. G., & Salehi, Μ. (2008). *Digital Communications (5 εκδ.)*. McGraw-Hill Higher Education.

Μουρτζόπουλος, Ι. (2012). *Ψηφιακή Τεχνολογία Ήχου*. Πάτρα: Πανεπιστήμιο Πατρών.

4.3 Ψηφιακό Υλικό – Λογισμικό Ασκήσεων

Στην υποενότητα αυτή παρουσιάζεται η σχεδίαση και η υλοποίηση του ψηφιακού υποστηρικτικού υλικού.

Για καθεμία διδακτική ενότητα αναπτύχθηκε μία διαδραστική διεπαφή – εφαρμογή σε Matlab, που μπορεί να χρησιμοποιηθεί για την εκπόνηση της αντίστοιχης εργαστηριακής άσκησης χρησιμοποιώντας το φύλλο εργασίας της άσκησης. Το σύνολο των διεπαφών είναι προσβάσιμο μέσω μίας κεντρικής διεπαφής – οθόνης, από όπου ο χρήστης μπορεί να επιλέξει την εργαστηριακή άσκηση που θέλει να εκτελέσει.

Η σχεδίαση των διεπαφών έγινε σύμφωνα με τις προδιαγραφές που αναφέρθηκαν στην ενότητα 2 της παρούσας εργασίας και ακολουθεί τους κανόνες ευχρηστίας του Nielsen (1994).

Η υλοποίηση των διεπαφών έγινε με το περιβάλλον GUIDE του Matlab έκδοσης 2012a, θέτοντας τον βασικό περιορισμό ότι η εφαρμογή τρέχει αποκλειστικά μέσα από το περιβάλλον GUIDE του Matlab έκδοσης τουλάχιστον 2012a.

Υπενθυμίζεται ότι σκοπός των εφαρμογών Matlab (ψηφιακού υλικού) είναι η παροχή ενός γραφικού εκπαιδευτικού περιβάλλοντος για την υποβοήθηση εκμάθησης βασικών και προχωρημένων εννοιών και τεχνικών της ψηφιακής επεξεργασίας σήματος, στο πλαίσιο διδασκαλίας του εργαστηριακού μαθήματος της ψηφιακής επεξεργασίας σήματος. Το σύνολο του ψηφιακού υλικού θα μπορεί να χρησιμοποιηθεί ως βοήθημα από τους σπουδαστές μαζί με κατάλληλα διαμορφωμένα φύλλα εργασιών (όπως αυτά που αναπτύχθηκαν στο πλαίσιο της τρέχουσας εργασίας) για την εκμάθηση βασικών και προχωρημένων τεχνικών της ψηφιακής επεξεργασίας σήματος, όπως η γραμμική και κυκλική συνέλιξη, οι μετασχηματισμοί DTFT, DFT και Z, η δειγματοληψία κ.ο.κ.

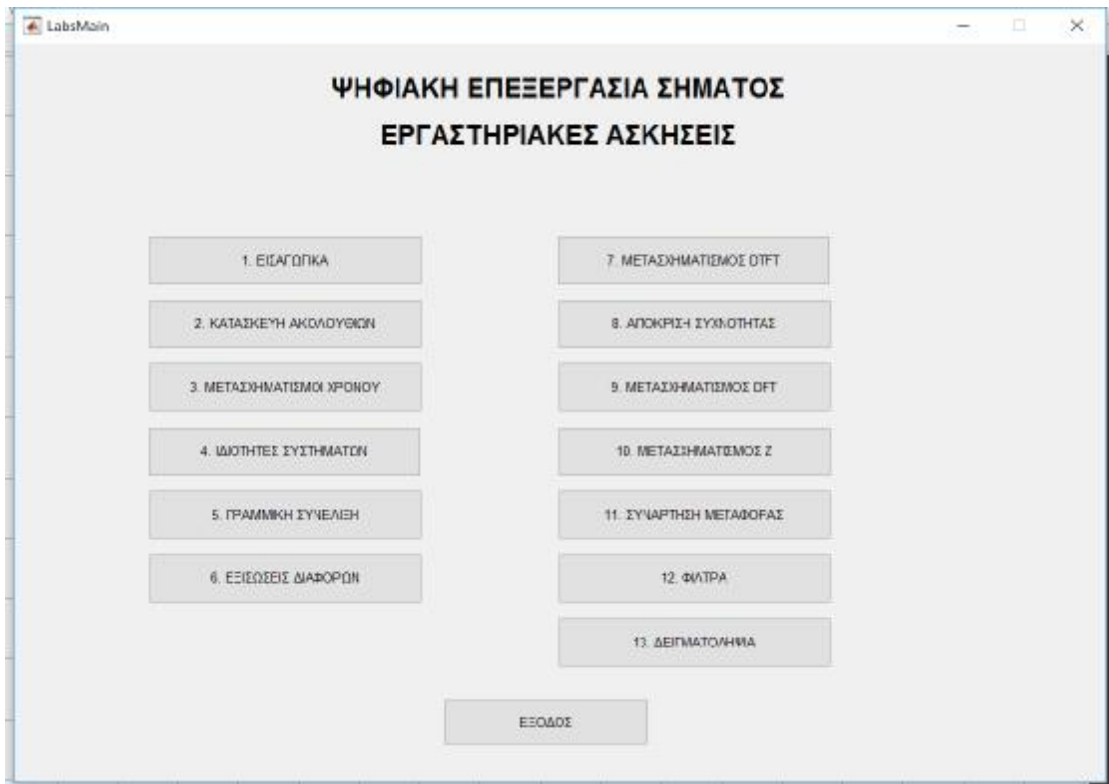
Η επιλογή του Matlab και του ενσωματωμένου περιβάλλοντος GUIDE ως προγραμματιστικό περιβάλλον για την ανάπτυξη του ψηφιακού υλικού έγινε λόγω των πλεονεκτημάτων που παρουσιάζει, όπως αναφέρονται στην ενότητα 3 της

παρούσας εργασίας. Στα βασικότερα πλεονεκτήματα συγκαταλέγονται: α) η ευκολία που παρέχει στην υλοποίηση γραφικών διεπαφών υψηλής διαδραστικότητας, β) η δυνατότητα πειραματισμού που παρέχει στον χρήστη και γ) η παρουσίαση των αποτελεσμάτων με χρήση πολλαπλών τεχνικών απεικόνισης, όπως πίνακες και γραφήματα, επιτρέποντας την καλύτερη μελέτη των αποτελεσμάτων.

Υπενθυμίζεται ότι η βασική καινοτομία της παρούσας εργασίας είναι η υποστήριξη της καθημίας διδακτικής ενότητας από μία ξεχωριστή διαδραστική διεπαφή σε Matlab, παρέχοντας στον χρήστη τη δυνατότητα να πειραματιστεί είτε χρησιμοποιώντας τα έτοιμα πρότυπα φύλλα εργασίας είτε εισάγοντας δικά του δεδομένα. Ωστόσο, τα φύλλα εργασίας και οι διεπαφές Matlab μπορούν να χρησιμοποιηθούν και ανεξάρτητα μεταξύ τους.

Καθεμία από τις δεκατρείς (13) διδακτικές ενότητες υποστηρίζεται από μία εφαρμογή – διεπαφή Matlab και το σύνολό τους είναι προσβάσιμο από μία κεντρική διεπαφή – οθόνη.

Στην εικόνα που ακολουθεί (Εικόνα 13) παρουσιάζεται η κεντρική διεπαφή – οθόνη του λογισμικού ασκήσεων, μέσω της οποίας ο χρήστης μπορεί να επιλέξει την ενότητα – εργαστηριακή άσκηση που τον ενδιαφέρει.

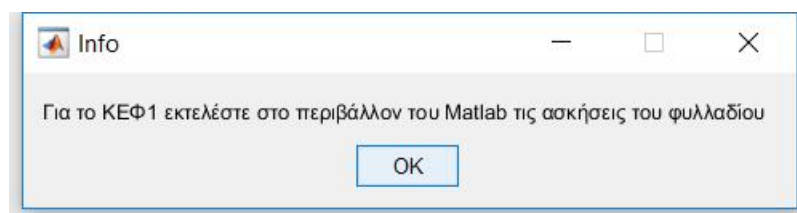


Εικόνα 13: Κεντρική οθόνη λογισμικού ασκήσεων

Παρακάτω ακολουθεί μία σύντομη περιγραφή καθεμίας διεπαφής.

Εργαστηριακή Άσκηση 1: Εισαγωγή στο Matlab

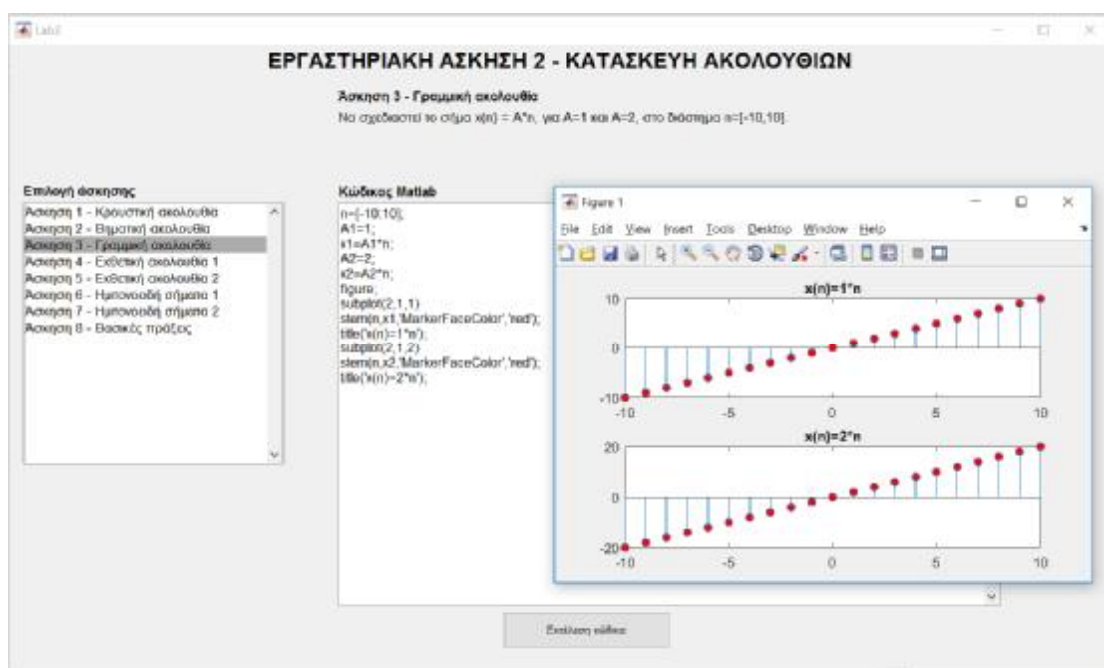
Όταν ο χρήστης πατήσει στην κεντρική διεπαφή το πλήκτρο «1. ΕΙΣΑΓΩΓΙΚΑ», εμφανίζεται μήνυμα που τον προτρέπει να εκτελέσει στο περιβάλλον του Matlab τις λυμένες ασκήσεις της 1ης εργαστηριακής άσκησης, όπως δίνονται στο αντίστοιχο φύλλο εργασίας (Εικόνα 14).



Εικόνα 14: Μήνυμα 1^{ης} εργαστηριακής άσκησης

Εργαστηριακή Άσκηση 2: Δημιουργία Βασικών Σημάτων Διακριτού Χρόνου – Βασικές Πράξεις

Όταν ο χρήστης πατήσει στην κεντρική διεπαφή το πλήκτρο «2. ΚΑΤΑΣΚΕΥΗ ΑΚΟΛΟΥΘΙΩΝ», εμφανίζεται παράθυρο στο οποίο υπάρχουν 8 λυμένες ασκήσεις κατασκευής ακολουθιών με τον αντίστοιχο κώδικα σε Matlab, τις οποίες ο χρήστης μπορεί να εκτελέσει άμεσα και να δει τα αποτελέσματά τους (Εικόνα 15).

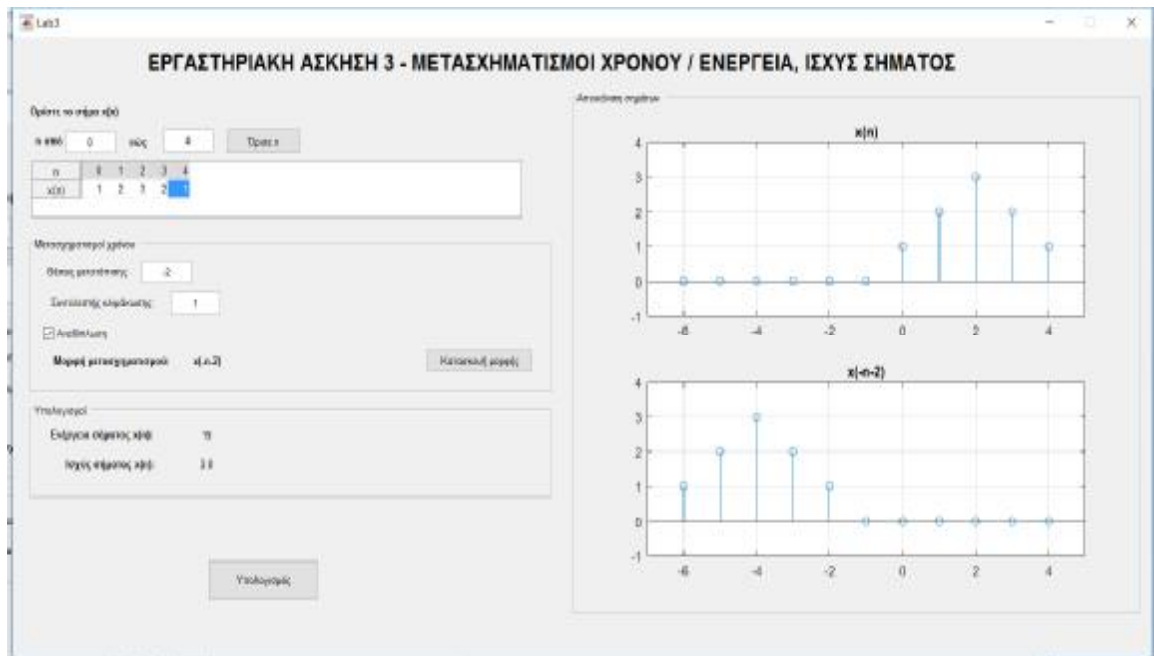


Εικόνα 15: Διεπαφή 2^{ης} εργαστηριακής άσκησης

Εργαστηριακή Άσκηση 3: Μετασχηματισμοί της Ανεξάρτητης Μεταβλητής – Χαρακτηριστικά Μεγέθη ΣΔΧ

Όταν ο χρήστης πατήσει στην κεντρική διεπαφή το πλήκτρο «3. ΜΕΤΑΣΧΗΜΑΤΙΣΜΟΙ ΧΡΟΝΟΥ», εμφανίζεται παράθυρο στο οποίο μπορεί να ορίσει τις θέσεις (πεδία n από, έως) και τα πλάτη ενός σήματος $x(n)$. Ακολούθως μπορεί να κατασκευάσει τον επιθυμητό μετασχηματισμό χρόνου, εισάγοντας τις θέσεις μετατόπισης, το συντελεστή κλιμάκωσης και επιλέγοντας αν υπάρχει αναδίπλωση, οπότε πατώντας το πλήκτρο «Υπολογισμός» υπολογίζονται και εμφανίζονται η ενέργεια και η ισχύς του σήματος $x(n)$, ενώ στο δεξί μέρος της διεπαφής

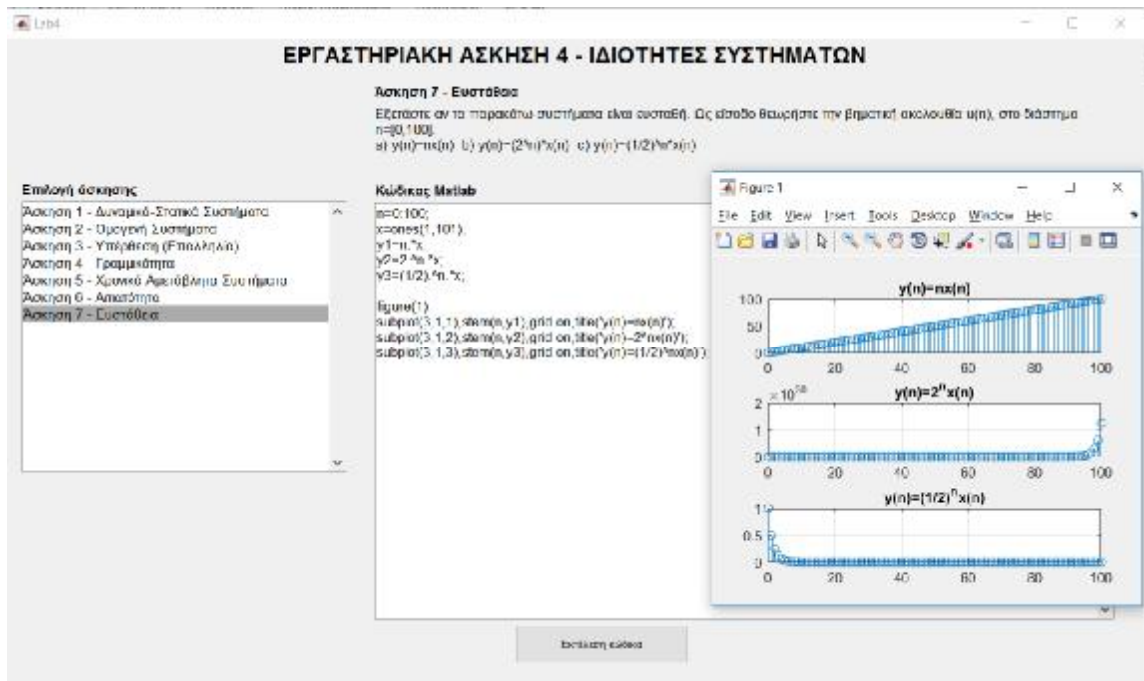
εμφανίζονται οι γραφικές παραστάσεις του αρχικού και του μετασχηματισμένου σήματος (Εικόνα 16).



Εικόνα 16: Διεπαφή 3^{ης} εργαστηριακής άσκησης

Εργαστηριακή Άσκηση 4: Ιδιότητες Συστημάτων

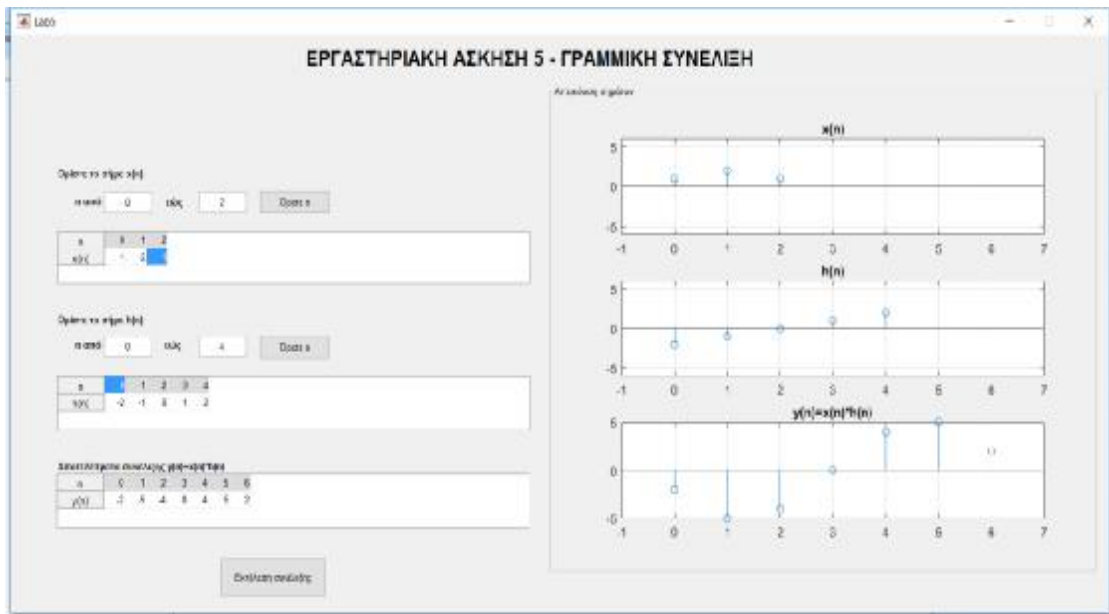
Όταν ο χρήστης πατήσει στην κεντρική διεπαφή το πλήκτρο «4. ΙΔΙΟΤΗΤΕΣ ΣΥΣΤΗΜΑΤΩΝ», εμφανίζεται παράθυρο στο οποίο υπάρχουν 7 λυμένες ασκήσεις που αφορούν τις ιδιότητες των συστημάτων με τον αντίστοιχο κώδικα σε Matlab, τις οποίες ο χρήστης μπορεί να εκτελέσει άμεσα και να δει τα αποτελέσματά τους (Εικόνα 17).



Εικόνα 17: Διεπαφή 4^{ης} εργαστηριακής άσκησης

Εργαστηριακή Άσκηση 5: Γραμμική Συνέλιξη

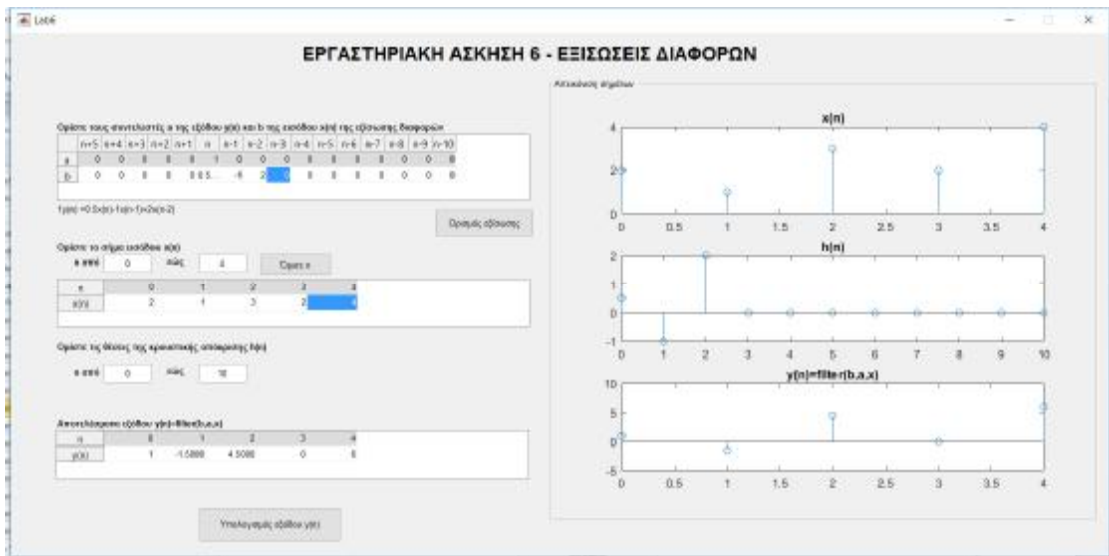
Όταν ο χρήστης πατήσει στην κεντρική διεπαφή το πλήκτρο «5. ΓΡΑΜΜΙΚΗ ΣΥΝΕΛΙΞΗ», εμφανίζεται παράθυρο στο οποίο μπορεί να ορίσει τις θέσεις (πεδία **από, έως**) και τα πλάτη για τα σήματα $x(n)$ και $h(n)$. Πατώντας το πλήκτρο «Εκτέλεση συνέλιξης» υπολογίζεται και εμφανίζεται η συνέλιξη των δύο σημάτων σε μορφή πίνακα (πλάτη – θέσεις), ενώ στο δεξί μέρος της διεπαφής εμφανίζονται οι γραφικές παραστάσεις των σημάτων $x(n)$, $h(n)$ και της συνέλιξης $y(n)$ (Εικόνα 18).



Εικόνα 18: Διεπαφή 5^{ης} εργαστηριακής άσκησης

Εργαστηριακή Άσκηση 6: Εξισώσεις Διαφορών

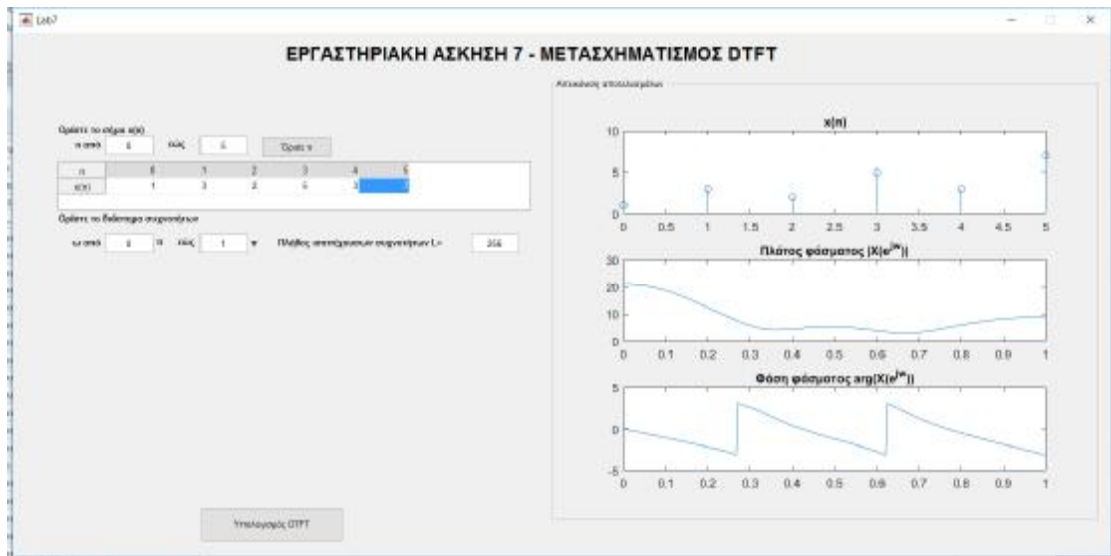
Όταν ο χρήστης πατήσει στην κεντρική διεπαφή το πλήκτρο «6. ΕΞΙΣΩΣΕΙΣ ΔΙΑΦΟΡΩΝ», εμφανίζεται παράθυρο στο οποίο μπορεί να ορίσει την εξίσωση διαφορών ενός συστήματος, συμπληρώνοντας τους συντελεστές **a** και **b** της εξόδου $y(n)$ και της εισόδου $x(n)$ αντίστοιχα. Επίσης, ο χρήστης μπορεί να ορίσει το σήμα εισόδου $x(n)$ εισάγοντας τις θέσεις (πεδία **n από, έως**) και τα πλάτη του σήματος $x(n)$, καθώς και τις θέσεις της κρουστικής απόκρισης $h(n)$ του συστήματος (πεδία **n από, έως**). Πατώντας το πλήκτρο «Υπολογισμός εξόδου $y(n)$ » υπολογίζεται και εμφανίζεται η έξοδος του συστήματος μέσω της συνάρτησης `filter` του Matlab σε μορφή πίνακα (πλάτη – θέσεις), ενώ στο δεξί μέρος της διεπαφής εμφανίζονται οι γραφικές παραστάσεις των σημάτων $x(n)$, $h(n)$ και της εξόδου $y(n)$ (Εικόνα 19).



Εικόνα 19: Διεπαφή 6^{ης} εργαστηριακής άσκησης

Εργαστηριακή Άσκηση 7: Μετασχηματισμός Fourier Διακριτού Χρόνου – DTFT

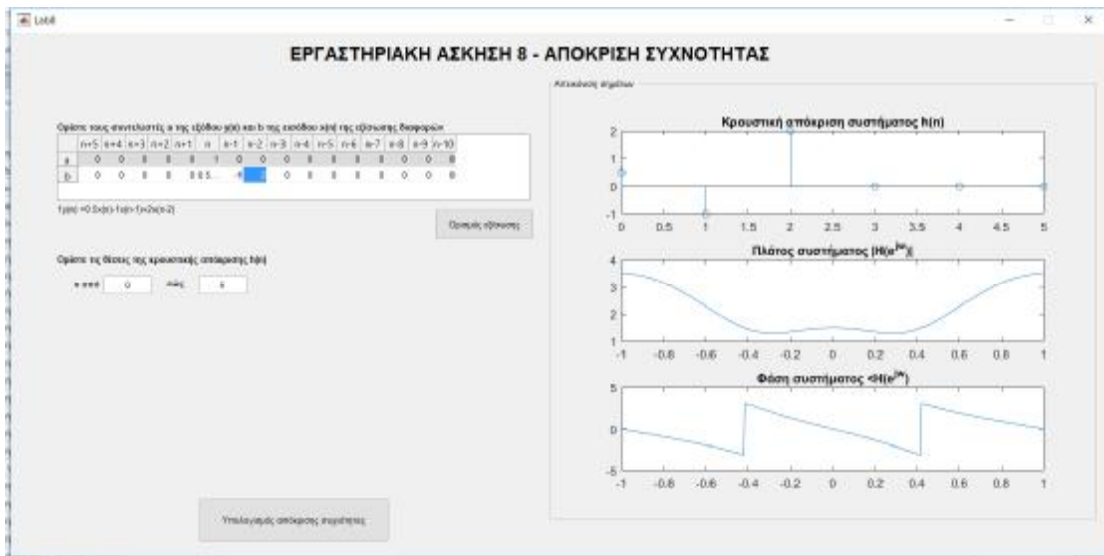
Όταν ο χρήστης πατήσει στην κεντρική διεπαφή το πλήκτρο «7. ΜΕΤΑΣΧΗΜΑΤΙΣΜΟΣ DTFT», εμφανίζεται παράθυρο στο οποίο μπορεί να ορίσει το σήμα εισόδου $x(n)$ εισάγοντας τις θέσεις (πεδία **η από, έως**) και τα πλάτη, καθώς και το διάστημα συχνοτήτων (ω από, έως) και το πλήθος L των ισαπέχουσων συχνοτήτων στις οποίες θα χωριστεί το διάστημα της γωνιακής συχνότητας. Πατώντας το πλήκτρο «Υπολογισμός DTFT» υπολογίζεται ο DTFT και στο δεξί μέρος της διεπαφής εμφανίζονται οι γραφικές παραστάσεις του σήματος $x(n)$ και του πλάτους και της φάσης του φάσματος $X(e^{j\omega})$ (Εικόνα 20).



Εικόνα 20: Διεπαφή 7^{ης} εργαστηριακής άσκησης

Εργαστηριακή Άσκηση 8: Απόκριση Συχνότητας

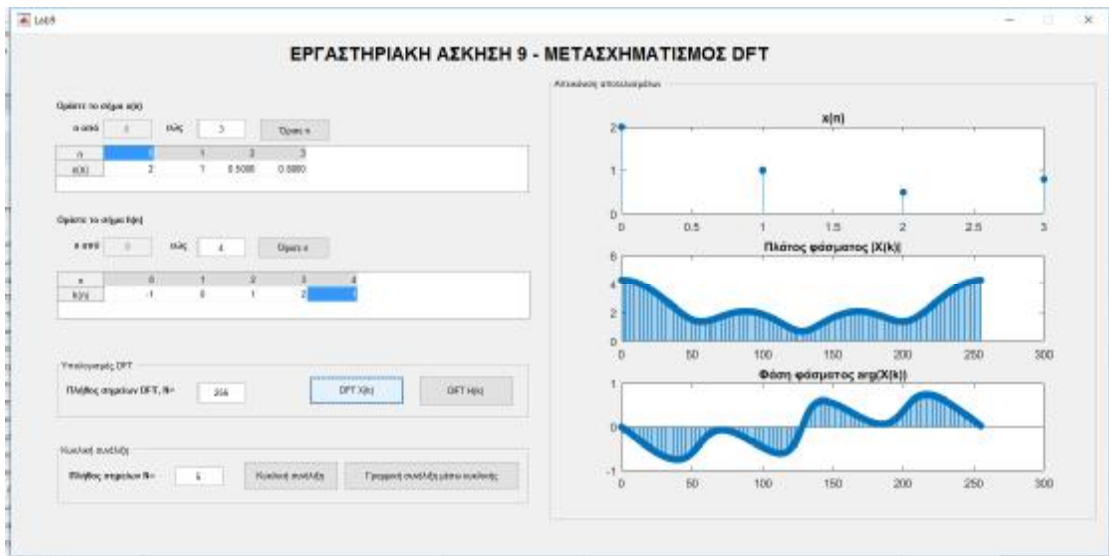
Όταν ο χρήστης πατήσει στην κεντρική διεπαφή το πλήκτρο «8. ΑΠΟΚΡΙΣΗ ΣΥΧΝΟΤΗΤΑΣ», εμφανίζεται παράθυρο στο οποίο μπορεί να ορίσει την εξίσωση διαφορών ενός συστήματος, συμπληρώνοντας τους συντελεστές **a** και **b** της εξόδου $y(n)$ και της εισόδου $x(n)$ αντίστοιχα. Επίσης, ο χρήστης μπορεί να εισάγει τις θέσεις (πεδία **n από, έως**) της κρουστικής απόκρισης $h(n)$. Πατώντας το πλήκτρο «Υπολογισμός απόκρισης συχνότητας» υπολογίζεται η απόκριση συχνότητας του συστήματος και στο δεξί μέρος της διεπαφής εμφανίζονται οι γραφικές παραστάσεις της κρουστικής απόκρισης $h(n)$ και του πλάτους και της φάσης της απόκρισης συχνότητας του συστήματος $H(e^{j\omega})$ (Εικόνα 21).



Εικόνα 21: Διεπαφή 8^{ης} εργαστηριακής άσκησης

Εργαστηριακή Άσκηση 9: Διακριτός Μετασχηματισμός Fourier – DFT

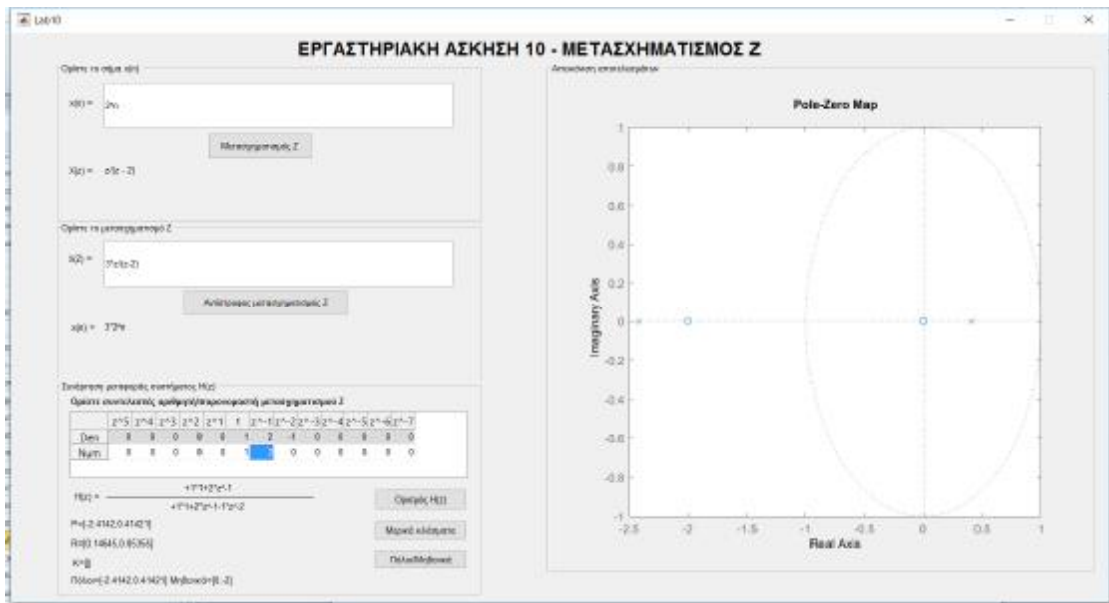
Όταν ο χρήστης πατήσει στην κεντρική διεπαφή το πλήκτρο «9. ΜΕΤΑΣΧΗΜΑΤΙΣΜΟΣ DFT», εμφανίζεται παράθυρο στο οποίο μπορεί να ορίσει τις θέσεις (πεδία η **από, έως**) και τα πλάτη για τα σήματα x(n) και h(n). Επίσης, ο χρήστης μπορεί να εισάγει το πλήθος N των σημείων του DFT, καθώς και το πλήθος των σημείων της κυκλικής συνέλιξης. Πατώντας το πλήκτρο «DFT X(k)» ή «DFT H(k)» υπολογίζεται ο αντίστοιχος DFT και στο δεξί μέρος της διεπαφής εμφανίζονται οι γραφικές παραστάσεις του σήματος x(n) ή h(n) και του πλάτους και της φάσης του αντίστοιχου DFT X(k) ή H(k). Επίσης, πατώντας το πλήκτρο «Κυκλική συνέλιξη» και «Γραμμική συνέλιξη μέσω κυκλικής» υπολογίζονται οι αντίστοιχες πράξεις και στο δεξί μέρος της διεπαφής εμφανίζονται οι αντίστοιχες γραφικές παραστάσεις (Εικόνα 22).



Εικόνα 22: Διεπαφή 9^{ης} εργαστηριακής άσκησης

Εργαστηριακή Άσκηση 10: Μετασχηματισμός Z

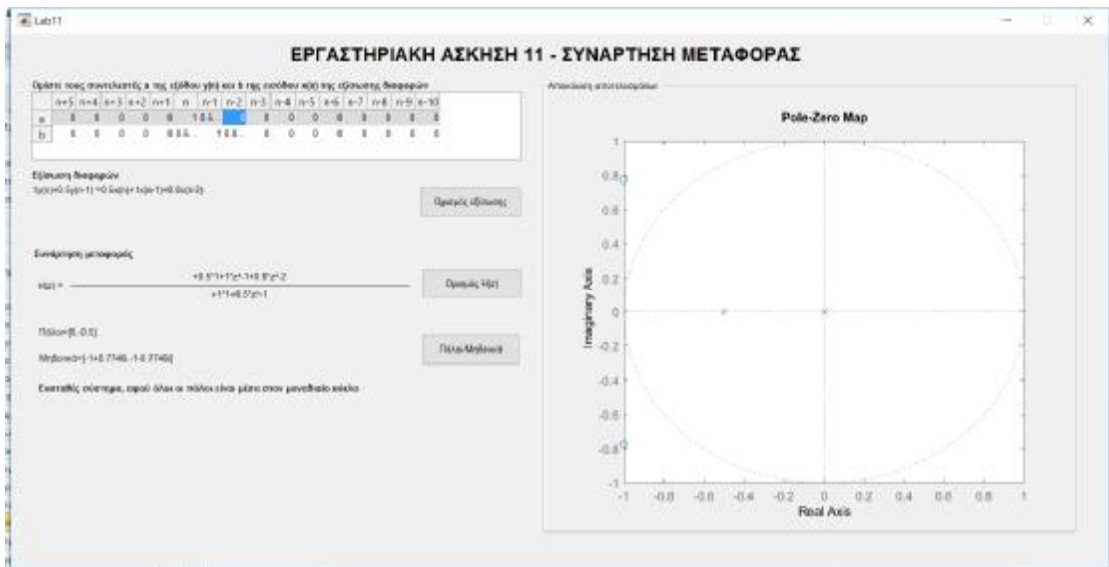
Όταν ο χρήστης πατήσει στην κεντρική διεπαφή το πλήκτρο «10. ΜΕΤΑΣΧΗΜΑΤΙΣΜΟΣ Z», εμφανίζεται παράθυρο στο οποίο μπορεί να ορίσει τον τύπο ενός σήματος $x(n)$ και να υπολογίσει τον μετασχηματισμό $X(z)$. Επίσης, μπορεί να κάνει και το αντίστροφο, δηλαδή να εισάγει τον τύπο ενός μετασχηματισμού Z και να υπολογίσει την αρχική ακολουθία στον χρόνο (εφόσον υπολογίζεται). Τέλος, ο χρήστης μπορεί να ορίσει τη συνάρτησης μεταφοράς του συστήματος εισάγοντας τους συντελεστές του αριθμητή (NUM) και του παρονομαστή (DEN) της $H(z)$ και στη συνέχεια να κάνει ανάλυση σε μερικά κλάσματα και να υπολογίσει τους πόλους και τα μηδενικά της $H(z)$ και να εμφανίσει το αντίστοιχο διάγραμμα πόλων/μηδενικών (Εικόνα 23Εικόνα 22).



Εικόνα 23: Διεπαφή 10^{ης} εργαστηριακής άσκησης

Εργαστηριακή Άσκηση 11: Συνάρτηση Μεταφοράς

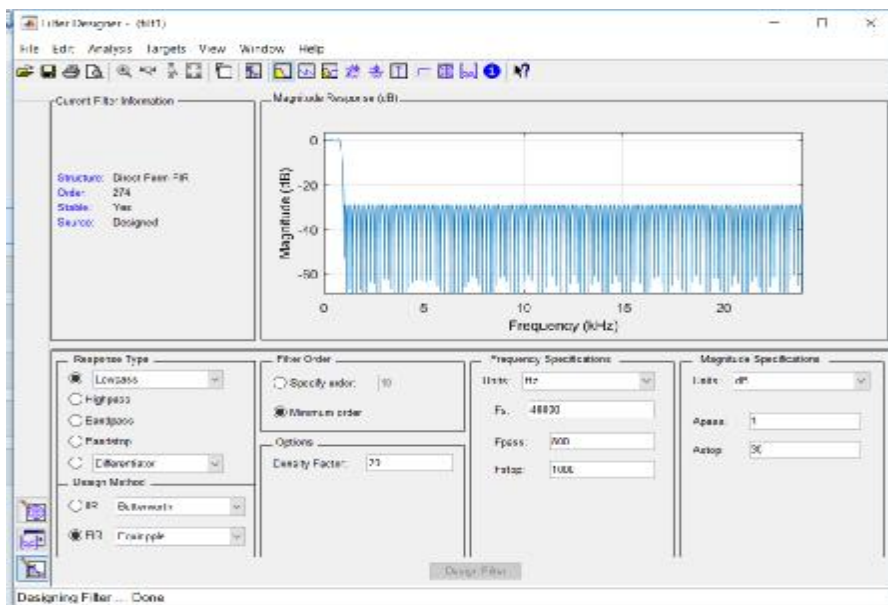
Όταν ο χρήστης πατήσει στην κεντρική διεπαφή το πλήκτρο «11. ΣΥΝΑΡΤΗΣΗ ΜΕΤΑΦΟΡΑΣ», εμφανίζεται παράθυρο στο οποίο μπορεί να ορίσει την εξίσωση διαφορών ενός συστήματος, συμπληρώνοντας τους συντελεστές a και b της εξόδου $y(n)$ και της εισόδου $x(n)$ αντίστοιχα. Επίσης, ο χρήστης πατώντας το πλήκτρο «Ορισμός $H(z)$ » υπολογίζεται και εμφανίζεται η συνάρτηση μεταφοράς του συστήματος, ενώ πατώντας στο πλήκτρο «Πόλοι/Μηδενικά» υπολογίζονται και εμφανίζονται οι πόλοι, τα μηδενικά και η ευστάθεια του συστήματος, ενώ στο δεξί μέρος της διεπαφής εμφανίζεται το διάγραμμα πόλων και μηδενικών της $H(z)$ (Εικόνα 24).



Εικόνα 24: Διεπαφή 11^{ης} εργαστηριακής άσκησης

Εργαστηριακή Άσκηση 12: Φίλτρα

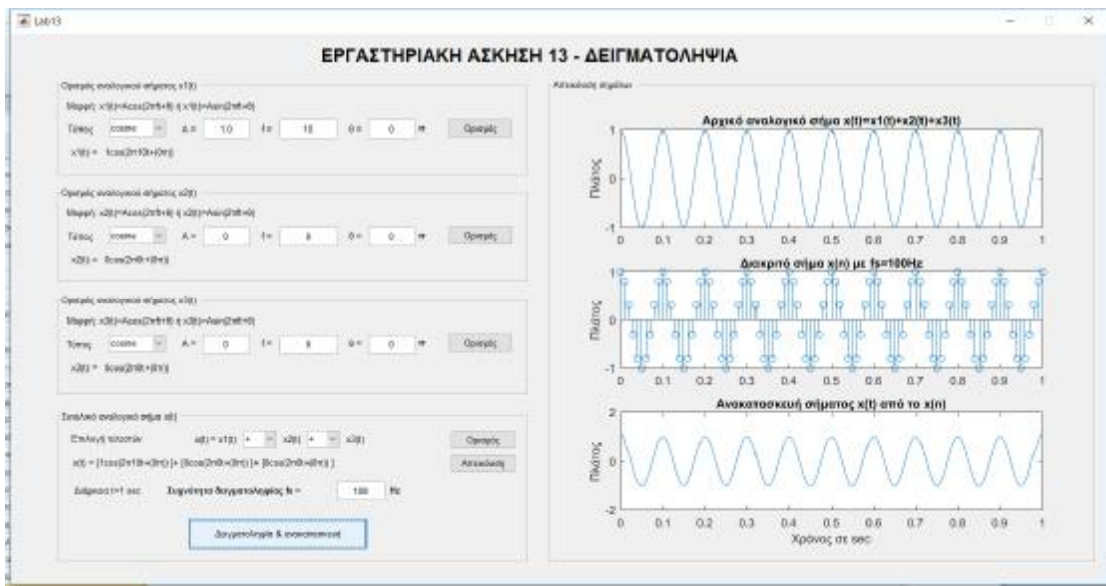
Όταν ο χρήστης πατήσει στην κεντρική διεπαφή το πλήκτρο «12. ΦΙΛΤΡΑ», ανοίγει το περιβάλλον του εργαλείου sptool του Matlab για ορισμό φίλτρων (Εικόνα 25).



Εικόνα 25: Διεπαφή 12^{ης} εργαστηριακής άσκησης

Εργαστηριακή Άσκηση 13: Δειγματοληψία

Όταν ο χρήστης πατήσει στην κεντρική διεπαφή το πλήκτρο «13. ΔΕΙΓΜΑΤΟΛΗΨΙΑ», εμφανίζεται παράθυρο στο οποίο μπορεί να ορίσει τα χαρακτηριστικά μέχρι τριών αναλογικών σημάτων και να χρησιμοποιήσει κάποια πράξη μεταξύ τους για να ορίσει ένα πιο σύνθετο αναλογικό σήμα, εισάγοντας και τη συχνότητα δειγματοληψίας. Πατώντας ο χρήστης το πλήκτρο «Δειγματοληψία και ανακατασκευή», δημιουργείται διακριτό σήμα από τη δειγματοληψία του αναλογικού σήματος και ακολούθως, με βάση το διακριτό σήμα γίνεται ανακατασκευή του αρχικού αναλογικού σήματος. Στο δεξί μέρος της διεπαφής εμφανίζονται οι αντίστοιχες γραφικές παραστάσεις, όπου ο χρήστης μπορεί να κάνει οπτική σύγκριση του αρχικού αναλογικού και του ανακατασκευασμένου σήματος, προκειμένου να συμπεράνει αν η συχνότητα δειγματοληψίας που επέλεξε ήταν ικανοποιητική ή όχι (Εικόνα 24).



Εικόνα 26: Διεπαφή 13^{ης} εργαστηριακής άσκησης

5 Συμπεράσματα

Παρουσιάστηκε το βασικό θεωρητικό πλαίσιο της εργασίας, εστιάζοντας στη διδασκαλία της ψηφιακής επεξεργασίας σήματος, καθώς και στις προδιαγραφές που πρέπει να έχει το εκπαιδευτικό υλικό, τόσο το έντυπο όσο και το ψηφιακό, προκειμένου να θεωρείται ποιοτικό. Όσον αφορά το έντυπο υλικό εστίασαμε στις προδιαγραφές των φύλλων εργασίας, ενώ όσον αφορά το ψηφιακό υλικό εστίασαμε στο διαδραστικό λογισμικό ασκήσεων. Επίσης, έγινε ειδική αναφορά στους μαθησιακούς στόχους – προσδοκώμενα αποτελέσματα, δεδομένης της σημαντικότητάς τους στην κατασκευή δομημένου υποστηρικτικού υλικού, όπως εργαστηριακών ασκήσεων.

Ακολούθως, παρουσιάστηκαν τα βασικά χαρακτηριστικά του Matlab και του περιβάλλοντος GUIDE που διαθέτει για τη δημιουργία γραφικών διεπαφών.

Επίσης, παρουσιάστηκε αναλυτικά η σχεδίαση και η ανάπτυξη του υποστηρικτικού υλικού, έντυπου – φύλλα εργασίας και ψηφιακού – λογισμικό ασκήσεων με το Matlab, καλύπτοντας δεκατρείς (13) διδακτικές ενότητες του εργαστηριακού μαθήματος της «Ψηφιακή Επεξεργασία Σήματος».

Για καθεμία διδακτική ενότητα αναπτύχθηκε μία διαδραστική διεπαφή – εφαρμογή σε Matlab, που μπορεί να χρησιμοποιηθεί για την εκπόνηση της αντίστοιχης εργαστηριακής άσκησης χρησιμοποιώντας το φύλλο εργασίας της άσκησης. Το σύνολο των διεπαφών είναι προσβάσιμο μέσω μίας κεντρικής διεπαφής – οθόνης, από όπου ο χρήστης μπορεί να επιλέξει την εργαστηριακή άσκηση που θέλει να εκτελέσει.

Υπενθυμίζεται ότι σκοπός των φύλλων εργασίας (έντυπο υλικό) και των εφαρμογών Matlab (ψηφιακού υλικού) είναι η υποστήριξη και η υποβοήθηση εκμάθησης βασικών και προχωρημένων εννοιών και τεχνικών της ψηφιακής επεξεργασίας σήματος, στο πλαίσιο διδασκαλίας του εργαστηριακού μαθήματος της ψηφιακής επεξεργασίας σήματος. Το σύνολο του ψηφιακού υλικού μπορεί να χρησιμοποιηθεί ως βοήθημα από τους σπουδαστές μαζί με κατάλληλα

διαμορφωμένα φύλλα εργασιών (όπως αυτά που αναπτύχθηκαν στο πλαίσιο της τρέχουσας εργασίας) για την εκμάθηση βασικών και προχωρημένων τεχνικών της ψηφιακής επεξεργασίας σήματος, όπως η γραμμική και κυκλική συνέλιξη, οι μετασχηματισμοί DTFT, DFT και Z, η δειγματοληψία κ.ο.κ.

Βασική καινοτομία της παρούσας εργασίας είναι η υποστήριξη της καθημίας διδακτικής ενότητας από μία ξεχωριστή δαδραστική διεπαφή σε Matlab, παρέχοντας στον χρήστη τη δυνατότητα να πειραματιστεί είτε χρησιμοποιώντας τα πρότυπα φύλλα εργασίας που υλοποιήθηκαν στο πλαίσιο της τρέχουσας εργασίας είτε εισάγοντας δικά του δεδομένα. Επίσης, τα φύλλα εργασίας και οι εφαρμογές Matlab μπορούν να χρησιμοποιηθούν και ανεξάρτητα μεταξύ τους.

6 Προοπτικές – Μελλοντικές Βελτιώσεις

Το υποστηρικτικό υλικό που αναπτύχθηκε στο πλαίσιο της παρούσας εργασίας καλύπτει τα βασικότερα θέματα που πραγματεύεται ένα τυπικό εισαγωγικό εργαστηριακό μάθημα ψηφιακής επεξεργασίας σήματος, για μονοδιάστατα σήματα (όπως φωνής, μουσικής, τηλεπικοινωνιών).

Συνεπώς, στις μελλοντικές βελτιώσεις μπορούν να ενταχθούν:

- Τεχνικές και ζητήματα επεξεργασίας 2D σημάτων (εικόνες)
- Υλοποίηση περισσότερων πρότυπων φύλλων εργασίας με λυμένα παραδείγματα σε Matlab
- Διάθεση του συνόλου του υποστηρικτικού υλικού μέσω ενός σύγχρονου συστήματος διαχείρισης μάθησης, όπως το Moodle

Βιβλιογραφία

- Bloom, B. (1956). *Taxonomy of Educational Objectives. Handbook I: The Cognitive Domain*. New York: David McKay Co Inc.
- Broesch, J. D., Stranneby, D., & Walker, W. (2008). *Digital Signal Processing: Instant access*. Butterworth-Heinemann.
- Davydov, V. (1999). *What is Real Learning Activity?*
- Gagné, R. M. (1977). *The Conditions of Learning and Theory of Instruction*. Fort Worth: Holt, Rinehart and Winston Inc.
- Hayes, M. H. (2002). *Ψηφιακή Επεξεργασία Σήματος*. (Α. Δημητριάδης, Μεταφρ.) Θεσσαλονίκη: Τζιόλα.
- Hedegaard, M. (1999). The Influence of Societal Knowledge Traditions on Children's Thinking and Conceptual Development. (M. Hedegaard, & J. Lompscher, Επιμ.) *Learning Activity and Development*.
- Hsu, H. P. (2002). *Αναλογικές και Ψηφιακές Επικοινωνίες*. Θεσσαλονίκη: Τζιόλα.
- Jayant, N. S., & Noll, P. (1984). *Digital Coding of Waveforms: Principles and Applications to Speech and Video*. Prentice Hall.
- JpFix. (n.d.). *Digital Photo Retouching & Enhancement Service*. Ανάκτηση 2 10, 2017, από http://www.jpfix.com/About_Us/Articles/FPGA-Based_Image_Processing_Ac/fpga-based_image_processing_ac.html
- Lent, C. S. (2013). *Learning To Program With Matlab: Building GUI Tools*. Wiley & Sons.
- Lionarakis, A. (1999). Distance, but not distant education – The first steps into the 21st century. Στο M. Keegan (Επιμ.), *L' Enseignement a Distance a l' aube du troisieme millenaire* (σσ. 157-174). Centre National d' Enseignement a Distance.

- Mager. (1984). *Preparing instructional objectives*. Belmont: David S. Lake.
- Mano, M. M., & Ciletti, M. D. (2012). *Digital Design* (5 εκδ.). Pearson Education.
- Marchand, P., & Holland, T. O. (2003). *Graphics And GUIs With Matlab* (3rd ed.). Washington DC: Chapman & Hall/CRC.
- MathWorks - DSP*. (2017). Ανάκτηση 2 10, 2017, από Getting Started with Matlab DSP: <https://www.mathworks.com/help/dsp/getting-started-with-dsp-system-toolbox.html>
- Nielsen, J., & Mack, R. L. (1994). *Usability Inspection Methods*. New York, NY: John Wiley & Sons.
- Proakis, J. G., & Salehi, M. (2008). *Digital Communications* (5 εκδ.). McGraw-Hill Higher Education.
- Race, P. (1999). *Το εγχειρίδιο της ανοιχτής εκπαίδευσης*. Αθήνα: Μεταίχμιο.
- Race, P. (2001). *500 Πρακτικές Συμβουλές για την Ανοικτή και Ευέλικτη Εκπαίδευση*. Αθήνα: Μεταίχμιο.
- Stranneby, D., & Walker, W. (2004). *Digital Signal Processing and Applications* (2 ed.). Elsevier.
- Vygotsky, S. (1978). *Mind in Society. The Development of Higher Psychological*.
- Βαρσάμης, Δ. (2006). *MATLAB Graphical User Interfaces: Σημειώσεις GUI για το εργαστήριο του μαθήματος Ασαφή Συστήματα*. Σέρρες: ΤΕΙ Σερρών - Τμήμα Πληροφορικής και Επικοινωνιών.
- Βικιπαίδεια - ISO IEC 9126*. (2016). Ανάκτηση 2 10, 2017, από ISO/IEC 9126: https://en.wikipedia.org/wiki/ISO/IEC_9126
- Βικιπαίδεια*. (n.d.). *Ψηφιακή Επεξεργασία Σήματος*. Ανάκτηση 2 10, 2017, από https://el.wikipedia.org/wiki/Ψηφιακή_επεξεργασία_σήματος

- Γκικόσος, Γ., & Κουτσούμπα, Μ. (2003). Μια θεωρητική προσέγγιση για το σχεδιασμό διδακτικού υλικού στην ανοικτή και εξ αποστάσεως πολυμορφική εκπαίδευση. Στο Α. Λιοναράκης (Επιμ.), *2ο Πανελλήνιο Συνέδριο για την Ανοικτή και εξ Αποστάσεως Εκπαίδευση* (σσ. 654-661). Αθήνα: Προπομπός.
- Καμέας, Α. (2012). *ΕΕΥΕΜ*. Ανάκτηση 2 10, 2017, από ΕΑΠ: <https://goo.gl/7TI5g4>
- Κορδάκη, Μ. (2010). *Προσωπική ιστοσελίδα Μαρίας Κορδάκη*. Ανάκτηση 2 19, 2017, από Φύλλα εργασίας: <https://www.ceid.upatras.gr/webpages/faculty/kordaki/csei/lessonplan.pdf>
- Κορδάκη, Μ., & Λάσκαρης, Α. (2003). Σύγχρονες Θεωρίες Μάθησης και Αξιολόγηση Ολοκληρωμένων Διαδικτυακών Περιβαλλόντων Διδασκαλίας και Μάθησης. Στο Α. Λιοναράκης (Επιμ.), *2ο Πανελλήνιο Συνέδριο για την Ανοικτή και εξ Αποστάσεως Εκπαίδευση*. Αθήνα: Προπομπός.
- Μακράκης, Β. (2000). *Υπερμέσα στην Εκπαίδευση*. Αθήνα: Μεταίχμιο.
- Ματραλής, Χ. (1998, 1999). Το έντυπο υλικό στην εκπαίδευση από απόσταση. Στο Α. Κόκκος, Α. Λιοναράκης, & Χ. Ματραλής (Επιμ.), *Ανοικτή και εξ' αποστάσεως εκπαίδευσης - Το εκπαιδευτικό υλικό και οι νέες τεχνολογίες*. Γ. Πάτρα: ΕΑΠ.
- Μικρόπουλος, Τ. (2000). *Εκπαιδευτικό λογισμικό: Θέματα σχεδίασης και αξιολόγησης λογισμικού υπερμέσων*. Αθήνα: Κλειδάριθμος.
- Μουρτζόπουλος, Ι. (2012). *Ψηφιακή Τεχνολογία Ήχου*. Πάτρα: Πανεπιστήμιο Πατρών.
- Παναγιωτακόπουλος, Χ., Πιερρακέας, Χ., & Πιντέλας, Π. (2003). *Το Εκπαιδευτικό Λογισμικό και η Αξιολόγησή του*. Αθήνα: Μεταίχμιο.
- ΤΕΙ Θεσσαλίας. (2015). Ανάκτηση 2 10, 2017, από ΤΕΙ - Θεσσαλίας: <https://e-class.teilar.gr/>
- Χοντολίδου, Ε. (1999). Εισαγωγή στην έννοια της πολυτροπικότητας, *Γλωσσικός Υπολογιστής*. 1(1), σσ. 115-117.