

Δομές Συστημάτων Διακριτού Χρόνου

Επιμέλεια: Γιώργος Π. Καφεντζης
Δρ. Επιστήμης Η/Υ Πανεπιστημίου Κρήτης
Δρ. Επεξεργασίας Σήματος Πανεπιστημίου Rennes 1

3 Δεκεμβρίου 2015

1 Εισαγωγή

Ως τώρα έχουμε συναντήσει τρεις διαφορετικές περιγραφές ενός ΓΧΑ συστήματος:

- την εξίσωση διαφορών εισόδου-εξόδου (με σταθερούς συντελεστές) του
- την κρουστική απόκρισή του $h[n]$
- τη συνάρτηση μεταφοράς του $H(z)$

Και οι τρεις αυτές περιγραφές είναι θεωρητικά ισοδύναμες. Όταν όμως υλοποιούμε στην πράξη τέτοια συστήματα, δηλ. με αναλογικά ή ψηφιακά κυκλώματα, η εξίσωση διαφορών ή η συνάρτηση μεταφοράς πρέπει να μετατραπεί σε έναν αλγόριθμο ή σε μια δομή που να μπορεί να πραγματοποιηθεί με την υπάρχουσα τεχνολογία. Όπως θα δούμε σε αυτό το κεφάλαιο, βασικοί δομικοί λίθοι που χρησιμοποιούνται σε αυτές τις δομές είναι οι βασικές πράξεις της πρόσθεσης, του πολλαπλασιασμού με μια σταθερά, και της καθυστέρησης (αποθήκευσης στη μνήμη).

Ας δούμε ένα απλό παράδειγμα. Έστω το σύστημα με συνάρτηση μεταφοράς

$$H(z) = \frac{b_0 + b_1 z^{-1}}{1 - a z^{-1}}, \quad |z| > |a| \quad (1)$$

Η κρουστική απόκριση του παραπάνω συστήματος δίνεται εύκολα (έτσι δεν είναι; :-)) ως:

$$h[n] = b_0 a^n u[n] + b_1 a^{n-1} u[n-1] \quad (2)$$

και η πρώτη τάξης εξίσωση διαφορών δίνεται ως

$$y[n] - a y[n-1] = b_0 x[n] + b_1 x[n-1] \quad (3)$$

Αν θέλουμε να υλοποιήσουμε αυτό το σύστημα, η πράξη της συνέλιξης δε θα ήταν βολική, γιατί η κρουστική απόκριση $h[n]$ είναι άπειρη σε διάρκεια, οπότε θα χρειαζόμασταν άπειρες θέσεις μνήμης για να αποθηκεύσουμε την κρουστική απόκριση ή το αποτέλεσμα της συνέλιξης.

Όμως μπορούμε να γράψουμε την εξίσωση διαφορών ως

$$y[n] = a y[n-1] + b_0 x[n] + b_1 x[n-1] \quad (4)$$

η οποία μας δίνει τη βάση ενός αναδρομικού αλγορίθμου για τον υπολογισμό της εξόδου οποιαδήποτε χρονική στιγμή n , συναρτήσει της προηγούμενης εξόδου $y[n-1]$ και των εισόδων $x[n]$ και $x[n-1]$. Όπως γνωρίζετε, αν θεωρήσουμε συνθήκες αρχικής ηρεμίας στο σύστημα (δηλ. αν $x[n] = 0, n < 0$, τότε $y[n] = 0, n < 0$) και χρησιμοποιήσουμε τη Σχέση (4) ως αναδρομική εξίσωση για τον υπολογισμό της εξόδου, το σύστημα θα

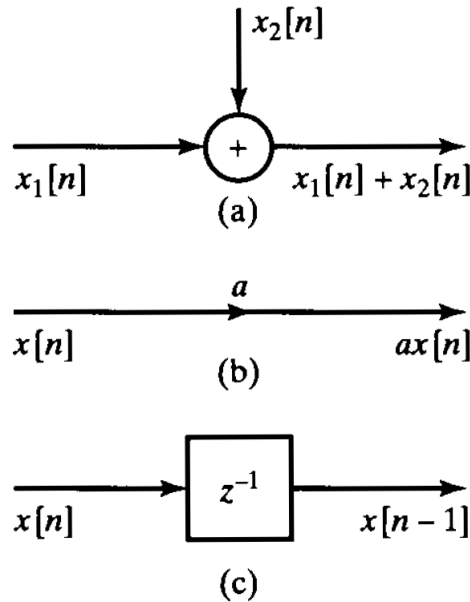
είναι τότε γραμμικό και χρονικά αμετάβλητο. Μπορούμε να γενικεύσουμε το πρόβλημά μας σε συστήματα N τάξης, αλλά είτε το παραπάνω σύστημα είτε η γενίκευσή του σε N τάξη δεν αποτελούν το μοναδικό αλγόριθμο υπολογισμού ενός συστήματος. Θα δούμε ότι υπάρχουν πολλές επιλογές για την υλοποίηση ενός συστήματος που έχει μια δεδομένη εξίσωση διαφορών. Η διαφορά ανάμεσα σε αυτές τις επιλογές συναντώνται

1. στο πλήθος των πράξεων
2. στο πλήθος των θέσεων μνήμης (καθυστέρηση)
3. στην ευρωστία τους στην υλοποίηση από μηχανές που χρησιμοποιούν πεπερασμένο πλήθος bits, δηλ. στη συμπεριφορά τους όταν η αριθμητική ακρίβεια είναι περιορισμένη

2 Αναπαράσταση Εξισώσεων Διαφορών με Διάγραμμα

Η υλοποίηση ενός ΓΧΑ συστήματος με χρήση αναδρομικού αλγόριθμου ο οποίος προέρχεται από την εξίσωση διαφορών απαιτεί προηγούμενες τιμές της εισόδου ή της εξόδου, όπως φαίνεται και στη Σχέση (4). Η καθυστέρηση των τιμών μιας ακολουθίας υπονοεί ότι υπάρχει ανάγκη αποθήκευσής τους σε μια μνήμη. Επίσης, πρέπει να έχουμε μηχανισμούς πολλαπλασιασμού τιμών με σταθερές, όπως και μηχανισμούς πρόσθεσης τιμών.

Με βάση τα παραπάνω, τα βασικά δομικά στοιχεία για την υλοποίηση ενός ΓΧΑ συστήματος είναι οι αθροιστές, οι πολλαπλασιαστές, και οι θέσεις μνήμης. Μια αναπαράστασή τους φαίνεται στο Σχήμα (1). Μια μικρή μόνο παρατήρηση για τη μοναδιαία καθυστέρηση, καθώς τα υπόλοιπα διαγράμματα είναι προφανή.



Σχήμα 1: Διαγράμματα για (α) πρόσθεση ακολουθιών, (β) πολλαπλασιασμό ακολουθίας με σταθερά, (γ) μοναδιαία καθυστέρηση.

Σε ψηφιακές υλοποιήσεις, η καθυστέρηση μπορεί να υλοποιηθεί ως ένας καταχωρητής για κάθε μοναδιαία καθυστέρηση. Σε αναλογικές υλοποιήσεις, η καθυστέρηση μπορεί να υλοποιηθεί ως συσκευή αποθήκευσης φορτίου (όμοια με τους πυκνωτές). Στα διαγράμματα που θα χρησιμοποιήσουμε, θα συμβολίσουμε την καθυστέρηση με τον όρο z^{-1} , όπως στο Σχήμα.

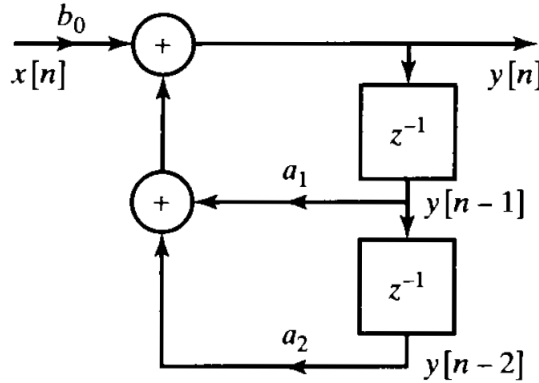
Ας προσπαθήσουμε λοιπόν τώρα να κατασκευάσουμε ένα διάγραμμα για το σύστημα που περιγράφεται ως

$$y[n] - a_1y[n-1] - a_2y[n-2] = b_0x[n] \quad (5)$$

του οποίου η συνάρτηση μεταφοράς του δίνεται ως

$$H(z) = \frac{b_0}{1 - a_1z^{-1} - a_2z^{-2}} \quad (6)$$

Το διάγραμμα που προκύπτει φαίνεται στο Σχήμα (2). Όταν το σύστημα υλοποιείται είτε σε έναν υπολογιστή



Σχήμα 2: Παράδειγμα διαγράμματος για την εξίσωση διαφορών της Σχέσης (5).

είτε σε ένα chip επεξεργασίας σήματος, δομές όπως αυτές του Σχήματος (2) χρησιμοποιούνται σαν βάσεις για το πρόγραμμα που υλοποιεί το σύστημα. Αν το σύστημα υλοποιείται με VLSI τεχνολογία, το διάγραμμα είναι η βάση για τον ορισμό της αρχιτεκτονικής του συστήματος, από πλευράς υλικού. Σε κάθε περίπτωση, τέτοια διαγράμματα δείχνουν ρητά ότι πρέπει να χρησιμοποιηθεί μνήμη για την αποθήκευση μεταβλητών (στην περίπτωσή μας, για τα $y[n-1]$, $y[n-2]$). Επίσης, βλέπουμε από το Σχήμα (2) ότι η έξοδος $y[n]$ υπολογίζεται δημιουργώντας πρώτα το γινόμενο των $a_1y[n-1]$ και $a_2y[n-2]$, μετά προσθετοντάς τα μεταξύ τους, και τέλος, προσθετοντάς το αποτέλεσμα στο $b_0x[n]$. Άρα, το διάγραμμα αναπαριστά με βολικό τρόπο την πολυπλοκότητα που σχετίζεται με τον αλγόριθμο υπολογισμού, τα βήματα του αλγορίθμου, και το ποσό του υλικού που απαιτείται για την πραγματοποίηση του συστήματος!

Ας γενικεύσουμε το παράδειγμά μας σε μεγαλύτερης τάξης εξίσωση διαφορών, ως

$$y[n] - \sum_{k=1}^N a_k y[n-k] = \sum_{k=0}^M b_k x[n-k] \quad (7)$$

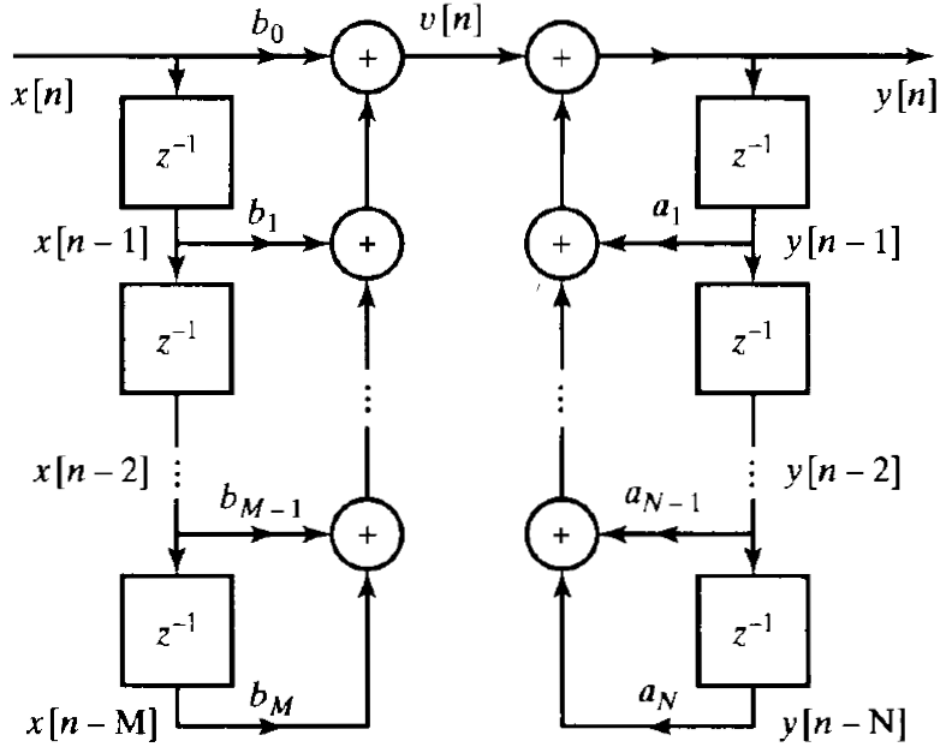
η οποία έχει συνάρτηση μεταφοράς

$$H(z) = \frac{\sum_{k=0}^M b_k z^{-k}}{1 - \sum_{k=1}^N a_k z^{-k}} \quad (8)$$

Ξαναγράφοντας τη Σχέση (7) ως αναδρομικό τύπο υπολογισμού της εξόδου, θα έχουμε

$$y[n] = \sum_{k=1}^N a_k y[n-k] + \sum_{k=0}^M b_k x[n-k] \quad (9)$$

Το διάγραμμα που αναπαριστά αυτόν τον τρόπο υπολογισμού της εξόδου του συστήματος φαίνεται στο Σχήμα (3). Το διάγραμμα του Σχήματος (3) είναι μια ρητή απεικόνιση της Σχέσης (9). Συγκεκριμένα,



Σχήμα 3: Παράδειγμα διαγράμματος για την εξίσωση διαφορών της Σχέσης (7).

αναπαριστά το ζεύγος εξισώσεων διαφορών

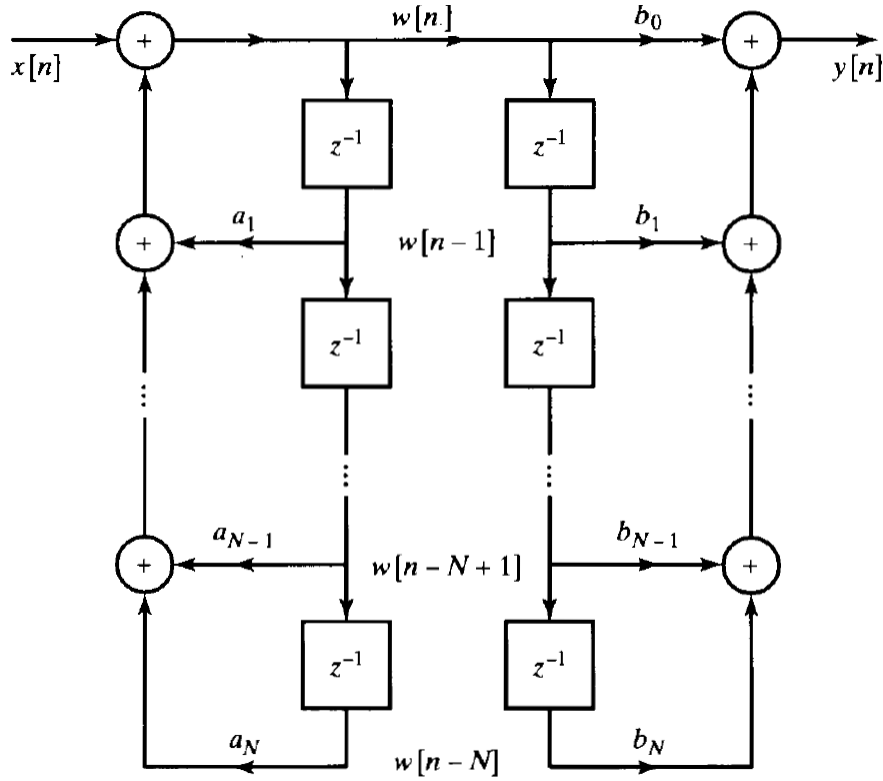
$$v[n] = \sum_{k=0}^M b_k x[n-k] \quad (10)$$

$$y[n] = \sum_{k=1}^N a_k y[n-k] + v[n] \quad (11)$$

Μια σημείωση εδώ για τη σειρά των πράξεων. Ο αθροιστής που χρησιμοποιούμε ως δομικό στοιχείο έχει δυο εισόδους, άρα σύμφωνα με το Σχήμα (3), το γινόμενο $a_N y[n-N]$ και το γινόμενο $a_{N-1} y[n-N+1]$ πρέπει να υπολογιστούν πρώτα, μετά να προστεθούν, και το άθροισμά τους να προστεθεί στο $a_{N-2} y[n-N+2]$, κ.ο.κ. Αφού υπολογιστεί το $y[n]$, οι μεταβλητές καθυστέρησης πρέπει να αναβαθμιστούν μετακινώντας το $y[n-N+1]$ στον καταχωρητή που κρατά το $y[n-N]$, κ.ο.κ.

Ένα διάγραμμα μπορεί να αναδιαταχθεί ή και τροποποιηθεί με πολλούς τρόπους χωρίς να μεταβάλλουμε τη συνάρτηση μεταφοράς που αναπαριστά. Κάθε μετατροπή ή αναδιάταξη αναπαριστά μια διαφορετική υλοποίηση του ίδιου συστήματος, δηλ. έναν διαφορετικό υπολογιστικό αλγόριθμο για το ίδιο σύστημα. Για παράδειγμα, το παράδειγμα του Σχήματος (3) μπορεί να ιδωθεί ως δυο συστήματα σε σειρά, με το πρώτο να υπολογίζει το $v[n]$ από το $x[n]$, και το δεύτερο να υπολογίζει το $y[n]$ από το $v[n]$. Δεδομένου ότι καθένα από αυτά τα δυο συστήματα είναι ΓΧΑ (θεωρώντας συνθήκες αρχικής ηρεμίας για τους καταχωρητές), η σειρά με την οποία βρίσκονται τα δυο συστήματα στο διάγραμμα μπορεί να αλλάξει, χωρίς να επηρεάζει αυτό το συνολικό σύστημα! Αυτή η αναδιάταξη φαίνεται στο Σχήμα (4). Στο Σχήμα (4) έχουμε υποθέσει ότι η τάξη αριθμητή και παρονομαστή είναι ίδια ($M = N$). Αν είναι διαφορετική, τότε απλά κάποιοι συντελεστές a_k ή b_k θα απαλειφθούν από το Σχήμα, και οι αντίστοιχοι κλάδοι θα εξαφανιστούν.

Αν δούμε τα δυο αυτά διαγράμματα με όρους συνάρτησης μεταφοράς, το Σχήμα (3) μπορεί να ιδωθεί



Σχήμα 4: Αναδιάταξη διαγράμματος για την εξίσωση διαφορών της Σχέσης (7).

ως η υλοποίηση του $H(z)$ μέσω της γραφής του ως

$$H(z) = H_2(z)H_1(z) = \left(\frac{1}{1 - \sum_{k=1}^N a_k z^{-k}} \right) \sum_{k=0}^M b_k z^{-k} \quad (12)$$

ή ισοδύναμα, ως το ζεύγος εξισώσεων

$$V(z) = H_1(z)X(z) = \left(\sum_{k=0}^M b_k z^{-k} \right) X(z) \quad (13)$$

$$Y(z) = H_2(z)V(z) = \left(\frac{1}{1 - \sum_{k=1}^N a_k z^{-k}} \right) V(z) \quad (14)$$

Από την άλλη μεριά, το διάγραμμα του Σχήματος (4) μπορεί να ιδωθεί ως η υλοποίηση του $H(z)$ μέσω της γραφής του ως

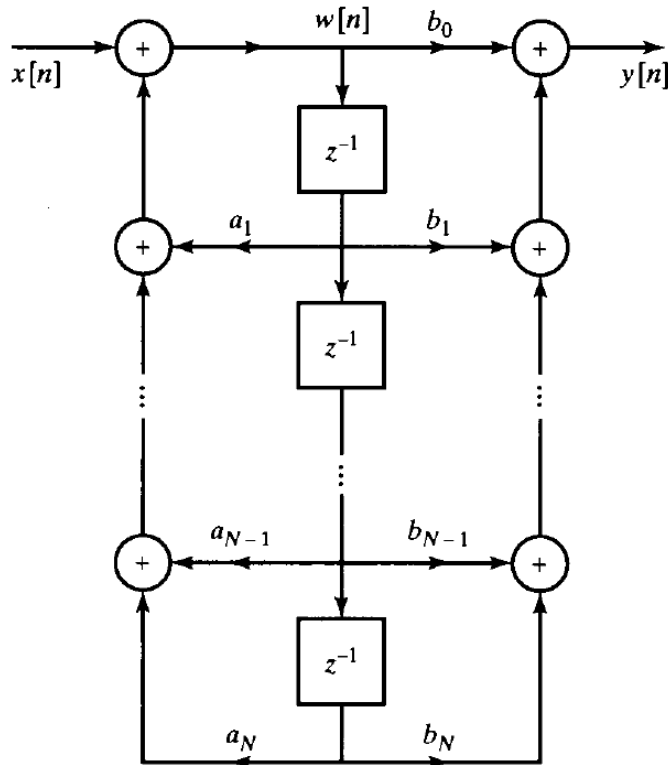
$$H(z) = H_1(z)H_2(z) = \sum_{k=0}^M b_k z^{-k} \left(\frac{1}{1 - \sum_{k=1}^N a_k z^{-k}} \right) \quad (15)$$

ή ισοδύναμα, ως το ζεύγος εξισώσεων

$$W(z) = H_2(z)X(z) = \left(\frac{1}{1 - \sum_{k=1}^N a_k z^{-k}} \right) X(z) \quad (16)$$

$$Y(z) = H_1(z)W(z) = \left(\sum_{k=0}^M b_k z^{-k} \right) W(z) \quad (17)$$

Τα διαγράμματα των Σχημάτων (3, 4) έχουν αρκετές σημαντικές διαφορές, τις οποίες πρέπει να επισημά-
 νουμε. Στο διάγραμμα (3) υλοποιούνται πρώτα τα μηδενικά του $H(z)$, δηλ. το υποσύστημα $H_1(z)$, και μετά
 οι πόλοι, δηλ. το υποσύστημα $H_2(z)$. Στο διάγραμμα (4) υλοποιούνται πρώτα οι πόλοι του $H(z)$, δηλ. το
 υποσύστημα $H_2(z)$, και μετά τα μηδενικά, δηλ. το υποσύστημα $H_1(z)$. Θεωρητικά, η σειρά της υλοποίησης
 δεν επηρεάζει το συνολικό σύστημα. Όμως, όπως θα δούμε, όταν μια εξίσωση διαφορών υλοποιείται με
 αριθμητική πεπερασμένης ακρίβειας, μπορεί να υπάρξουν σημαντικές διαφορές ανάμεσα στα δυο συστήματα
 που είναι θεωρητικά ισοδύναμα. Μια ακόμα σημαντική διαφορά είναι ο αριθμός των καθυστερήσεων στα
 δυο διαγράμματα. Και στα δυο, απαιτούνται συνολικά $N + M$ στοιχεία καθυστέρησης (μνήμη). Όμως, το
 διάγραμμα του Σχήματος (4) μπορεί να επανασχεδιαστεί, αν προσέξουμε ότι το ίδιο ακριβώς σήμα ($w[n]$)
 αποθηκεύεται δυο φορές, από μια φορά σε κάθε αλυσίδα υπολογισμών. Κατά συνέπεια, μπορούμε να πετά-
 ξουμε τη μια αλυσίδα, χρησιμοποιώντας μόνο μια αλυσίδα θέσεων μνήμης, κοινή ανάμεσα στις δυο αλυσίδες
 υπολογισμών. Αυτή η παρατήρηση οδηγεί στο διάγραμμα του Σχήματος (5). Σε αυτό το διάγραμμα, οι



Σχήμα 5: Αναδιάταξη διαγράμματος για την εξίσωση διαφορών της Σχέσης (7) με λιγότερες θέσεις μνήμης.

συνολικές καθυστερήσεις (θέσεις μνήμης) είναι λιγότερες από αυτές των δυο προηγούμενων διαγραμμάτων.
 Για την ακρίβεια, οι θέσεις μνήμης τώρα είναι ο ελάχιστος αριθμός θέσεων μνήμης l που απαιτείται για την
 υλοποίηση ενός συστήματος με συνάρτηση μεταφοράς $H(z)$ όπως στη Σχέση (8), και δίνεται εν γένει από
 τη σχέση

$$l = \max(N, M) \quad (18)$$

Μια υλοποίηση η οποία γίνεται με τον ελάχιστο αριθμό στοιχείων καθυστέρησης αναφέρεται συχνά ως
 υλοποίηση κανονικής μορφής - *canonic form*. Η μη κανονική μορφή υλοποίησης που φαίνεται στο Σχήμα (3)
 αναφέρεται ως υλοποίηση Direct Form I ενός γενικού συστήματος τάξης N , επειδή είναι μια ευθεία (direct)
 υλοποίηση της εξίσωσης διαφορών. Το διάγραμμα του Σχήματος (5) αναφέρεται συχνά ως υλοποίηση Direct
 Form II ή Canonic Direct Form.

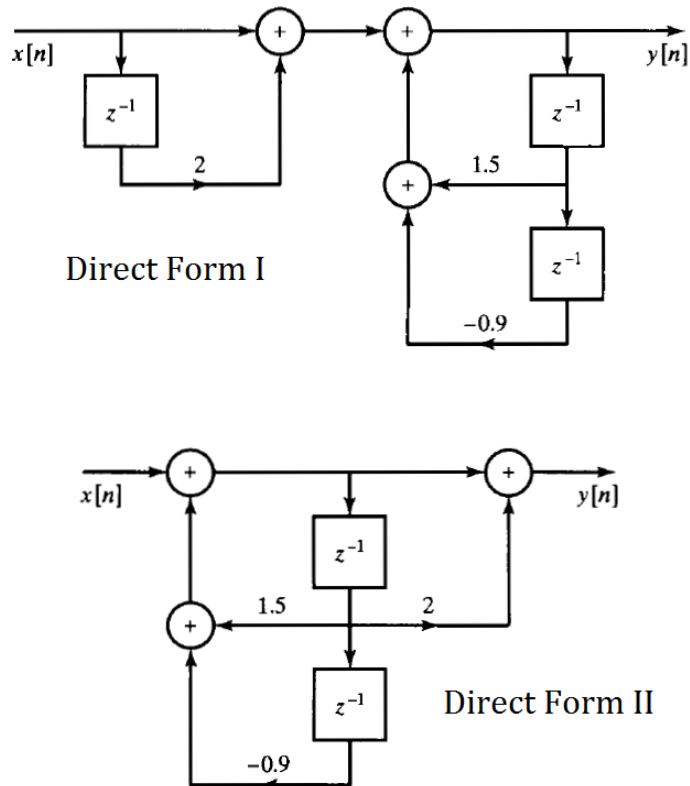
Ας δούμε ένα μικρό παράδειγμα.

Παράδειγμα:

Έστω το ΓΧΑ σύστημα που δίνεται από τη συνάρτηση μεταφοράς

$$H(z) = \frac{1 + 2z^{-1}}{1 - 1.5z^{-1} + 0.9z^{-2}} \quad (19)$$

Συγκρίνοντας την παραπάνω εξίσωση με τη Σχέση (8), παρατηρούμε ότι $b_0 = 1$, $b_1 = 2$, $a_1 = 1.5$, $a_2 = -0.9$. Έτσι, το Direct Form I και Direct Form II διάγραμμα φαίνεται στο Σχήμα (6). Προσέξτε ότι οι συντελεστές του παρονομαστή (πλην της μονάδας του πρώτου συντελεστή) έχουν αντίθετα πρόσημα από αυτά στην συνάρτηση μεταφοράς. Επίσης, προσέξτε ότι η Direct Form II σχεδίαση απαιτεί μόνο δυο καθυστερήσεις, μια λιγότερη από τις τρεις που απαιτεί η υλοποίηση σε Direct Form I.



Σχήμα 6: Direct Form I και II για το σύστημα του παραδείγματος.

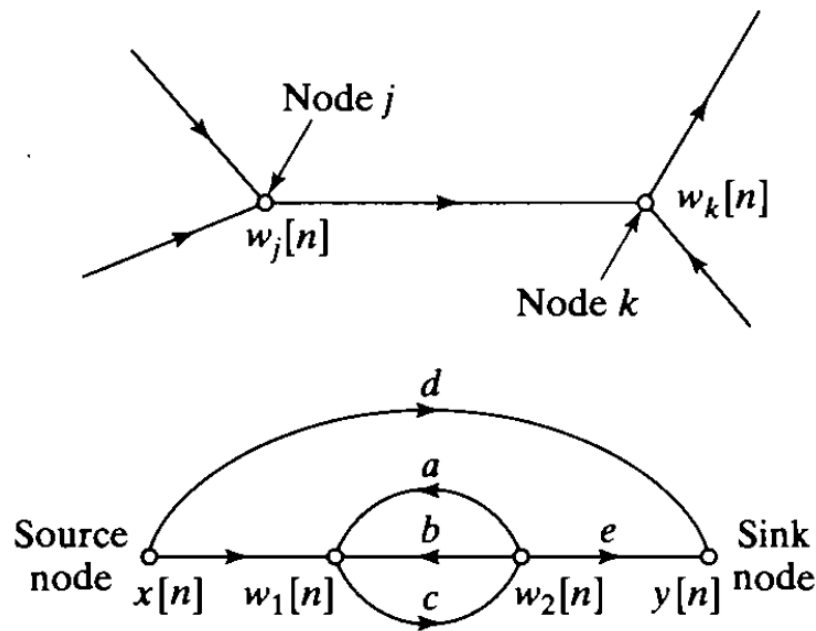
Παρ' όλο που τα διαγράμματα που σχεδιάζουμε ως τώρα είναι όμορφα και ξεκάθαρα, η σχεδίασή τους είναι επίπονη και κουραστική. Θα εισάγουμε μια διαφορετική αναπαράσταση, πιο εύκολη και γρήγορη στο σχεδιασμό της, που είναι μια καλή εναλλακτική στα διαγράμματα που παρουσιάσαμε ως τώρα.

3 Αναπαράσταση Εξισώσεων Διαφορών με Γράφους

Ένας γράφος μιας εξίσωσης διαφορών είναι ουσιαστικά το ίδιο ακριβώς με ένα διάγραμμα, εκτός από μερικές διαφορές στη σχεδίαση. Τυπικά, ένας γράφος είναι ένα δίκτυο από κατευθυνόμενους κλάδους (branches) που συνδέονται σε κόμβους (nodes). Κάθε κόμβος μπορεί να συσχετιστεί με μια μεταβλητή ή σταθερή τιμή. Η τιμή που σχετίζεται με τον k -οστό κόμβο μπορεί να αναπαρασταθεί ως w_k ή, μια και συνήθως θα παρουσιάζουμε ακολουθίες (δηλ. σήματα), η τιμή αυτή μπορεί να γραφεί ως $w_k[n]$. Ο κλάδος (j, k) δηλώνει έναν κλάδο με προέλευση τον κόμβο j και προορισμό τον κόμβο k . Η κατεύθυνση

αυτή δηλώνεται επάνω στον κλάδο με ένα βέλος περίπου στη μέση του. Κάθε κλάδος έχει μια είσοδο και μια έξοδο. Η είσοδος του κλάδου (j, k) μπορεί να ονομαστεί ως $w_j[n]$, ενώ η αντίστοιχη έξοδος $w_k[n]$. Για γραμμικούς γράφους - που μόνο αυτοί θα μας απασχολήσουν - η έξοδος από έναν κλάδο είναι ένας γραμμικός μετασχηματισμός της εισόδου του κλάδου. Το πιο απλό παράδειγμα τέτοιου μετασχηματισμού είναι ο πολλαπλασιασμός με μια σταθερά, δηλ. η έξοδος του κλάδου είναι απλά ένα πολλαπλάσιο της εισόδου του. Η δήλωση του μετασχηματισμού αυτού γίνεται με αναγραφή της σταθεράς του γινομένου επάνω στο βέλος του κλάδου. Όταν δεν υπάρχει τέτοια αναγραφή, τότε ο κλάδος δηλώνει το μοναδιαίο μετασχηματισμό, ή αλλιώς τον πολλαπλασιασμό της εισόδου με τη μονάδα. Εξ' ορισμού, η τιμή σε κάθε κόμβο του γράφου είναι το άθροισμα των τιμών των εξόδων από όλους τους κλάδους που καταλήγουν σε αυτόν τον κόμβο.

Τέλος, οι κόμβοι είναι δυο ειδών: πηγαίοι - *source* κόμβοι και τερματικοί - *sink* κόμβοι. Πηγαίοι λέγονται οι κόμβοι χωρίς είσοδο, ενώ τερματικοί λέγονται αυτοί που έχουν μόνο είσοδο. Τα παραπάνω φαίνονται γραφικά στο Σχήμα (7). Ας δούμε ένα παράδειγμα για το πως μετατρέπεται (ή πώς μοιάζει) ένα



Σχήμα 7: Σχεδίαση γράφου.

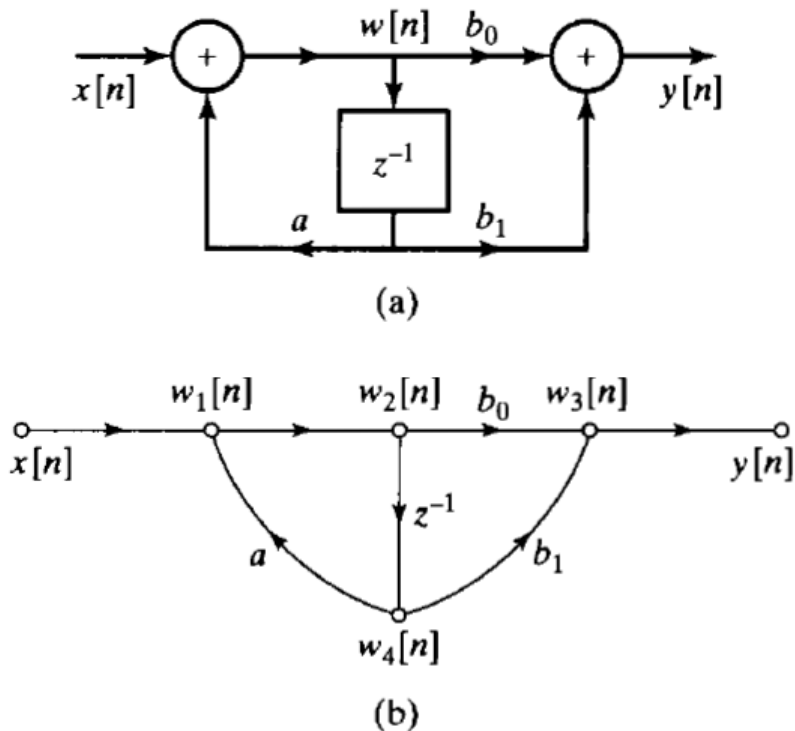
διάγραμμα που έχουμε δει ως τώρα, με τον αντίστοιχο γράφο του. Έστω το σύστημα

$$H(z) = \frac{b_0 + b_1 z^{-1}}{1 - a z^{-1}} \quad (20)$$

του οποίου το Direct Form II διάγραμμα φαίνεται στο Σχήμα (8)(a). Ο αντίστοιχος γράφος φαίνεται στο Σχήμα (8)(b). Η σύγκρισή τους πρέπει να σας κάνει προφανές ότι υπάρχει ευθεία αντιστοίχιση μεταξύ των κλάδων του διαγράμματος και των κλάδων του γράφου. Η σημαντική διαφορά ανάμεσα στα δυο σχέδια είναι ότι οι κόμβοι στους γράφους αναπαριστούν τόσο σημεία διακλάδωσης όσο και άθροισμα. Αντίθετα, στο διάγραμμα χρησιμοποιούσαμε ειδικό σύμβολο για την άθροιση. Αν και θα χρησιμοποιήσουμε γράφους λόγω απλότητας σχεδίασης, θα χρησιμοποιήσουμε επίσης και μερικά θεωρήματα για να καταλάβουμε περισσότερο τη δομή τους και για να εξετάσουμε εναλλακτικές επιλογές στην υλοποίηση γραμμικών συστημάτων.

Προτού προχωρήσουμε στην εξέταση γράφων για διάφορα συστήματα, ας μείνουμε λίγο παραπάνω στο γράφο του Σχήματος (8). Χρησιμοποιώντας ενδιάμεσες μεταβλητές σε κάθε κόμβο, μπορούμε να γράψουμε τις εξισώσεις

$$w_1[n] = a w_2[n] + x[n] \quad (21)$$



Σχήμα 8: Παράδειγμα διαγράμματος και μετατροπή σε γράφο.

$$w_2[n] = w_1[n] \quad (22)$$

$$w_3[n] = b_0 w_2[n] + b_1 w_4[n] \quad (23)$$

$$w_4[n] = w_2[n - 1] \quad (24)$$

$$y[n] = w_3[n] \quad (25)$$

Οι παραπάνω εξισώσεις ορίζουν έναν αλγόριθμο πολλαπλών βημάτων για τον υπολογισμό της εξόδου του ΓΧΑ συστήματος. Αυτές οι εξισώσεις μας δηλώνουν ουσιαστικά την προτεραιότητα των πράξεων που συνήθως συμβαίνουν στην υλοποίηση IIR συστημάτων. Οι παραπάνω εξισώσεις δε γίνεται να εκτελεστούν με τυχαία σειρά. Η 2η και η 5η εξίσωση, για παράδειγμα, είναι απλώς μετονομασία μεταβλητών, ενώ η 4η αναπαριστά την αναβάθμιση της μνήμης του συστήματος, απλώς αντικαθιστώντας τα περιεχόμενα ενός καταχωρητή που αντιπροσωπεύει το $w_4[n]$ από την τιμή του $w_2[n]$. Όμως αυτή η εργασία θα πρέπει να γίνεται με συνέπεια είτε πριν είτε μετά την εκτίμηση όλων των υπόλοιπων εξισώσεων. Σε αυτήν την περίπτωση, χρειαζόμαστε συνθήκες αρχικής ηρεμίας, ορίζοντας $w_2[-1] = 0$ ή $w_4[0] = 0$. Είναι λοιπόν ξεκάθαρο ότι οι παραπάνω εξισώσεις πρέπει να εκτελεστούν με τη δοθείσα σειρά. Φυσικά, οι δυο τελευταίες μπορούν να αλλάξουν θέση, ή η 4η μπορεί πάντα να υλοποιείται πρώτη απ' όλες.

Τέλος, σίγουρα παρατηρήσατε ότι κάποιες μεταβλητές είναι περιττές, καθώς απλά μετονομάζουν κάποιες ήδη υπάρχουσες. Πράγματι, οι παραπάνω εξισώσεις μπορούν να γραφούν πιο συνοπτικά ως

$$w_2[n] = a w_2[n - 1] + x[n] \quad (26)$$

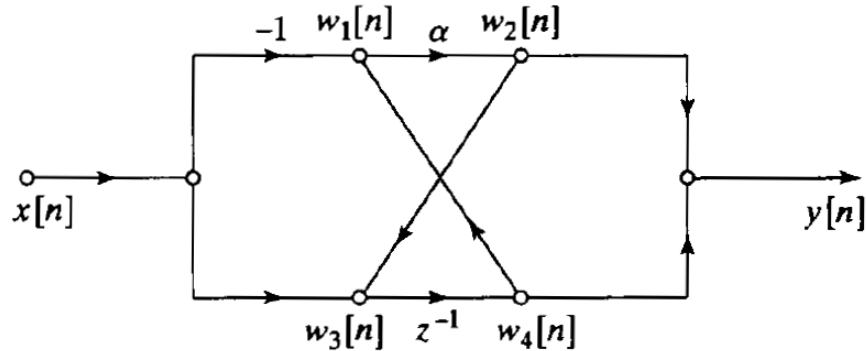
$$y[n] = b_0 w_2[n] + b_1 w_2[n - 1] \quad (27)$$

Ένας καλός κανόνας είναι να βάζουμε ενδιάμεσες μεταβλητές μόνο σε κόμβους όπου αθροίζονται κλάδοι, και όχι σε κόμβους διακλάδωσης. Επίσης, είναι πολύ προτιμότερο να δουλεύουμε στο χώρο του μετασχ. Ζ όταν εξετάζουμε ένα γράφο, παρά στο χώρο του χρόνου. Ως τώρα δεν το κάναμε αυτό, και τώρα είναι η κατάλληλη στιγμή να δείξουμε αυτή τη χρησιμότητα του μετασχ. Ζ στην εύρεση της εξίσωσης διαφορών

που αναπαριστά ένας γράφος.

Παράδειγμα:

Έστω ο γράφος του Σχήματος (9). Ο γράφος αυτός δε μοιάζει με κάποια γνωστή Direct Form σχεδία-



Σχήμα 9: Γράφος παραδείγματος.

ση. Άρα δεν μπορούμε να γράψουμε την εξίσωση διαφορών με απλή παρατήρηση του γράφου. Όμως αν χρησιμοποιήσουμε ενδιάμεσες μεταβλητές (σε όλους τους κόμβους, αν και περιττό), θα έχουμε τις εξής εξισώσεις:

$$w_1[n] = w_4[n] - x[n] \quad (28)$$

$$w_2[n] = aw_1[n] \quad (29)$$

$$w_3[n] = w_2[n] + x[n] \quad (30)$$

$$w_4[n] = w_3[n - 1] \quad (31)$$

$$y[n] = w_2[n] + w_4[n] \quad (32)$$

Αν μετατρέψουμε αυτές τις εξισώσεις στο χώρο του Z, θα έχουμε

$$W_1(z) = W_4(z) - X(z) \quad (33)$$

$$W_2(z) = aW_1(z) \quad (34)$$

$$W_3(z) = W_2(z) + X(z) \quad (35)$$

$$W_4(z) = z^{-1}W_3(z) \quad (36)$$

$$Y(z) = W_2(z) + W_4(z) \quad (37)$$

Μπορούμε εύκολα να απαλείψουμε τα $W_1(z)$, $W_3(z)$ με αντικατάσταση, και να έχουμε

$$W_2(z) = a(W_4(z) - X(z)) \quad (38)$$

$$W_4(z) = z^{-1}(W_2(z) + X(z)) \quad (39)$$

$$Y(z) = W_2(z) + W_4(z) \quad (40)$$

Λύνοντας ως προς $W_2(z)$ και $W_4(z)$ θα έχουμε

$$W_2(z) = \frac{a(z^{-1} - 1)}{1 - az^{-1}} X(z) \quad (41)$$

$$W_4(z) = \frac{z^{-1}(1 - a)}{1 - az^{-1}} X(z) \quad (42)$$

Οπότε τελικά το $Y(z)$ θα είναι

$$Y(z) = \left(\frac{a(z^{-1} - 1) + z^{-1}(1 - a)}{1 - az^{-1}} \right) X(z) = \left(\frac{z^{-1} - a}{1 - az^{-1}} \right) X(z) \quad (43)$$

Οπότε τελικά η συνάρτηση μεταφοράς για το σύστημα που υλοποιεί αυτός ο γράφος είναι η

$$H(z) = \frac{z^{-1} - a}{1 - az^{-1}} \quad (44)$$

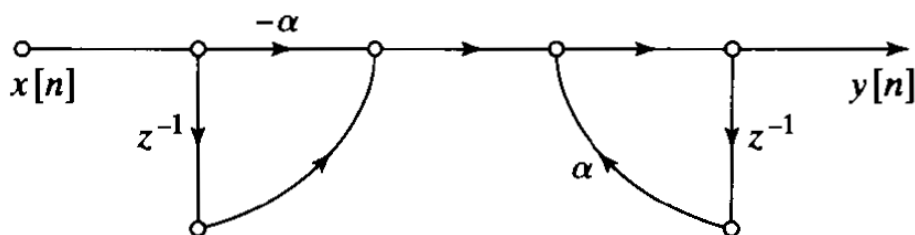
από την οποία προκύπτει ότι η κρουστική απόκριση είναι η

$$h[n] = a^{n-1}u[n-1] - a^{n+1}u[n] \quad (45)$$

και η εξίσωση διαφορών

$$y[n] - ay[n-1] = -ax[n] + x[n-1] \quad (46)$$

Ο γράφος που υλοποιεί σε Direct Form I το παραπάνω σύστημα φαίνεται στο Σχήμα (10).



Σχήμα 10: Γράφος παραδείγματος σε Direct Form I.

Το παράδειγμα αυτό μας έδειξε δυο πράγματα: πρώτον, το πόσο χρήσιμος μπορεί να είναι ο μετασχ. Ζ στην εύρεση της συνάρτησης μεταφοράς (και άρα, της κρουστικής απόκρισης και της εξίσωσης διαφορών) που υλοποιεί ένας γράφος. Δεύτερον, το πώς διαφορετικές αναπαραστάσεις γράφων ορίζουν διαφορετικούς αλγόριθμους υλοποίησης συστημάτων, που ο καθένας έχει διαφορετικές απαιτήσεις υπολογιστικών πόρων. Για παράδειγμα, βλέπετε ότι ο αρχικός γράφος απαιτεί μόλις έναν πολλαπλασιασμό και μια θέση μνήμης, ενώ ο Direct Form I γράφος απαιτεί δυο πολλαπλασιασμούς και δυο θέσεις μνήμης. Ο Direct Form II γράφος απαιτεί μια λιγότερη θέση μνήμης, όμως απαιτεί τον ίδιο αριθμό πολλαπλασιασμών με τον Direct Form I γράφο, δηλ. δυο πολλαπλασιασμούς. Σχεδιάστε τον για να βεβαιωθείτε! :-)

Όπως θα ξέρετε ήδη, τα συστήματα μπορούν να χωριστούν σε δυο κατηγορίες: τα Πεπερασμένης Κρουστικής Απόκρισης (FIR), και τα Άπειρης Κρουστικής Απόκρισης (IIR). Ας δούμε πόσο διαφορετικοί οι γράφοι για κάθε κατηγορία.

4 Βασικές Δομές IIR Συστημάτων

Ας ξεκινήσουμε από τα IIR συστήματα. Θα δούμε εδώ ότι υπάρχουν αρκετοί διαφορετικοί γράφοι που υλοποιούν μια συγκεκριμένη συνάρτηση μεταφοράς. Η κατάλληλη επιλογή γράφου πρέπει να λάβει υπόψη της διάφορα θέματα, όπως η υπολογιστική πολυπλοκότητα. Για παράδειγμα, σε μερικές ψηφιακές υλοποιήσεις, μας ενδιαφέρουν δομές με όσο γίνεται λιγότερους πολλαπλασιασμούς και καθυστερήσεις, κι αυτό γιατί ο πολλαπλασιασμός είναι γενικά χρονοβόρος και κοστίζει σε ψηφιακό υλικό, και γιατί κάθε καθυστέρηση αντιστοιχεί σε έναν καταχωρητή μνήμης. Κατά συνέπεια, η μείωση στον αριθμό των πολλαπλασιασμών συνεπάγεται αύξηση της ταχύτητας, και η μείωση στον αριθμό καθυστερήσεων συνεπάγεται μείωση στις απαιτήσεις μνήμης.

Άλλες, περισσότερο λεπτές, ισορροπίες απαιτούνται σε υλοποιήσεις VLSI, όπου συνήθως το εμβαδό του chip είναι σημαντική παράμετρος, ενώ η απλότητα και η αρθρωτή υλοποίηση είναι επίσης επιθυμητά στοιχεία. Σε πολυεπεξεργαστικό περιβάλλον, η πιο σημαντική παράμετρος είναι ο καταμερισμός του αλγορίθμου και της επικοινωνίας μεταξύ των επεξεργαστών. Επίσης, μεγάλης σημασίας είναι η επίδραση της πεπερασμένης ακρίβειας που αναγκαστικά υπάρχει σε πραγματικά συστήματα, λόγω πεπερασμένου πλήθους bits. Αυτή η επίδραση εξαρτάται από τον τρόπο οργάνωσης των υπολογισμών του γράφου, δηλ. από τη δομή του γράφου. Αυτό έχει ως αποτέλεσμα να προτιμούμε μερικές φορές ένα γράφο που δεν ικανοποιεί τις ελάχιστες απαιτήσεις μνήμης ή πολλαπλασιασμών, αν είναι αρκετά εύρωστος στην επίδραση της πεπερασμένης ακρίβειας. Θα μιλήσουμε για αυτά τα θέματα ακρίβειας λίγο αργότερα.

4.1 Direct Forms

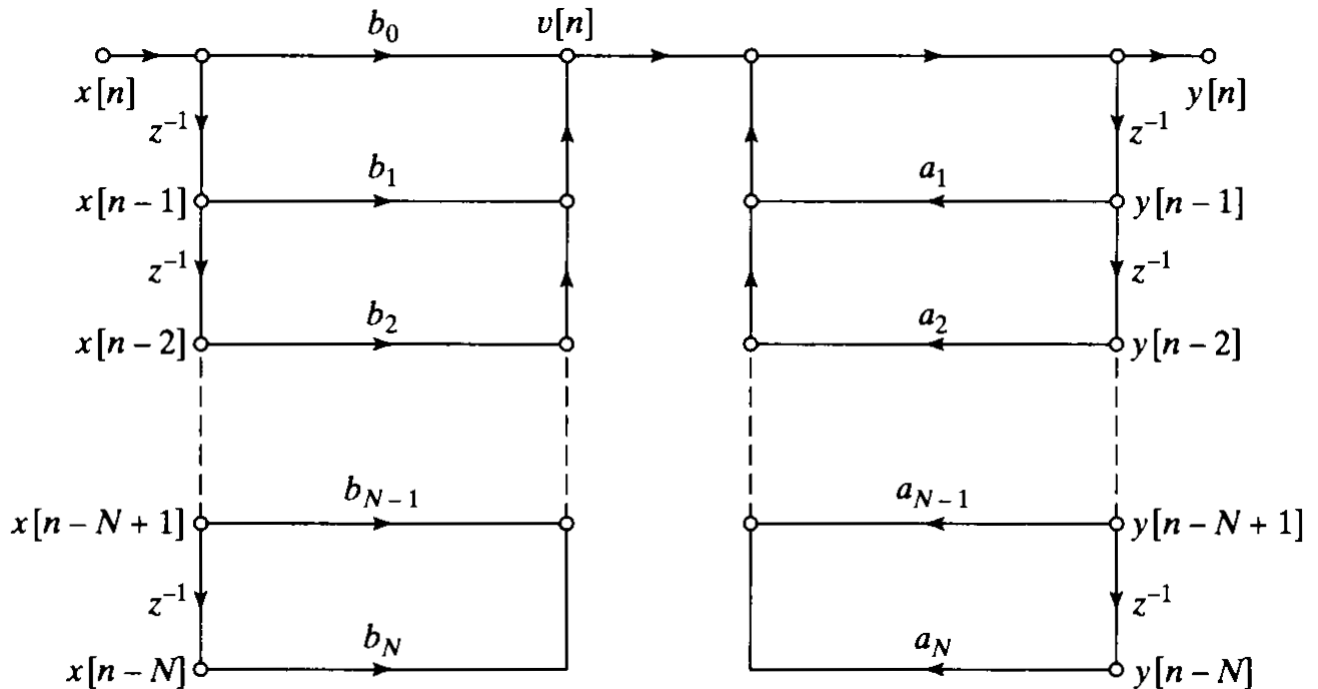
Ήδη έχουμε μια ιδέα για τη μορφή ενός Direct Form γράφου, τόσο του Direct Form I όσο και του II. Έστω η εξίσωση διαφορών

$$y[n] - \sum_{k=1}^N a_k y[n-k] = \sum_{k=0}^M b_k x[n-k] \quad (47)$$

η οποία ανταποκρίνεται στη συνάρτηση μεταφοράς

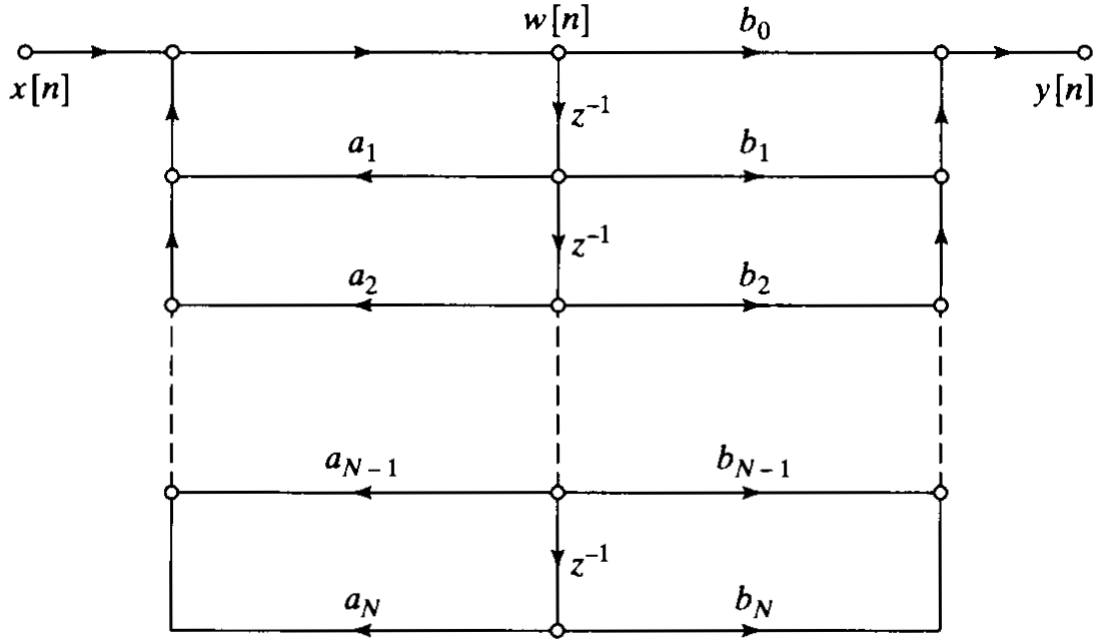
$$H(z) = \frac{\sum_{k=0}^M b_k z^{-k}}{1 - \sum_{k=1}^N a_k z^{-k}} \quad (48)$$

Στα Σχήματα (11) και (12) φαίνεται η γενική μορφή αυτών των δυο Direct Form γράφων, με την υπόθεση ότι η τάξη αριθμητή και παρονομαστή είναι ίδια ($M = N$). Η σχεδίαση των γράφων είναι τέτοια ώστε κάθε



Σχήμα 11: Direct Form I

κόμβος να έχει το πολύ δυο εισόδους. Ένας κόμβος ενός γράφου μπορεί να έχει περισσότερες εισόδους,



Σχήμα 12: Direct Form II

αλλά όπως είπαμε και νωρίτερα, αυτή η “σύμβαση” των δυο εισόδων έχει ως αποτέλεσμα γράφους που είναι πιο κοντά σε προγράμματα και αρχιτεκτονικές υλοποίησης των υπολογισμών που απαιτούνται για μια εξίσωση διαφορών που αναπαριστά ένας γράφος. Ας δούμε κι ένα παράδειγμα.

Παράδειγμα:

Έστω η συνάρτηση μεταφοράς

$$H(z) = \frac{1 + 2z^{-1} + z^{-2}}{1 - 0.75z^{-1} + 0.125z^{-2}} \quad (49)$$

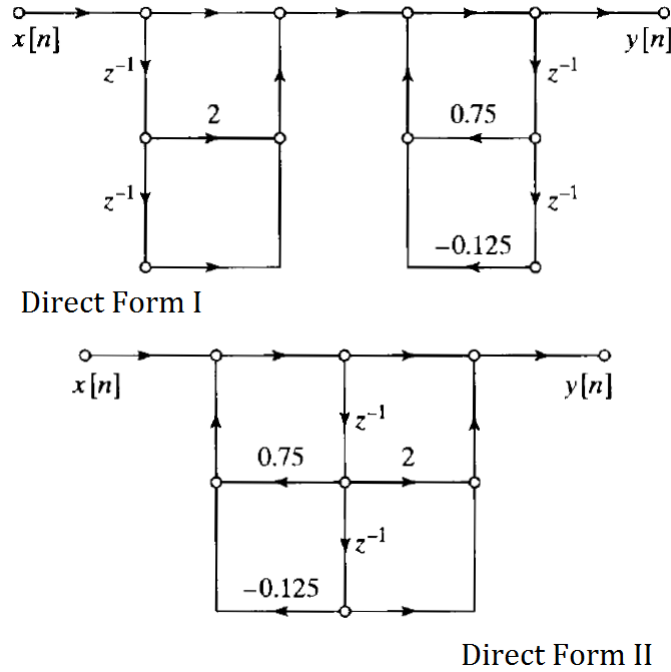
Μπορούμε να σχεδιάσουμε τους γράφους Direct Form I, II απευθείας, καθώς η συνάρτηση μεταφοράς είναι γραμμένη σε βολική μορφή. Έτσι, καταλήγουμε στο Σχήμα (13).

4.2 Μορφή σε Σειρά - Cascade Form

Οι Direct Forms προέρχονται από την απλή υλοποίηση των πολυωνύμων αριθμητή και παρονομαστή, όπως αυτοί παρουσιάζονται στη συνάρτηση μεταφοράς. Αν όμως παραγοντοποιήσουμε αριθμητή και παρονομαστή, τότε θα έχουμε

$$H(z) = A \frac{\prod_{k=1}^{M_1} (1 - f_k z^{-1}) \prod_{k=1}^{M_2} (1 - g_k z^{-1})(1 - g_k^* z^{-1})}{\prod_{k=1}^{N_1} (1 - c_k z^{-1}) \prod_{k=1}^{N_2} (1 - d_k z^{-1})(1 - d_k^* z^{-1})} \quad (50)$$

με $M = M_1 + 2M_2$ και $N = N_1 + 2N_2$. Σε αυτήν την έκφραση, οι όροι πρώτης τάξης αναπαριστούν μηδενικά, f_k , και πόλους c_k , στον πραγματικό άξονα. Οι όροι δεύτερης τάξης αναπαριστούν συζυγή ζεύγη μιγαδικών μηδενικών, g_k, g_k^* , και πόλων d_k, d_k^* . Προφανώς η παραπάνω συνάρτηση μεταφοράς ανταποκρίνεται σε ένα πραγματικό σύστημα. Η γραφή αυτή αναπαριστά μια κατηγορία δομών που περιέχουν σε σειρά όρους πρώτης και δεύτερης τάξης, ως υποσυστήματα. Υπάρχει αρκετή ελευθερία στην επιλογή της σύνθεσης των



Σχήμα 13: Direct Form I, II για ένα παράδειγμα IIR συστήματος.

υποσυστημάτων και στη σειρά με την οποία υλοποιούνται. Στην πράξη βέβαια, είναι συχνά επιθυμητή η υλοποίηση σε σειρά με τον ελάχιστο δυνατό αριθμό υπολογισμών και μνήμης. Μια αφθρική δομή που έχει πολλά πλεονεκτήματα μπορεί να πραγματοποιηθεί συνδυάζοντας ζεύγη πραγματικών όρων και συζυγών μιγαδικών σε όρους δεύτερης τάξης, έτσι ώστε η Σχέση (50) να γράφει ως

$$H(z) = \prod_{k=1}^{N_s} \frac{b_{0k} + b_{1k}z^{-1} + b_{2k}z^{-2}}{1 - a_{1k}z^{-1} - a_{2k}z^{-2}} \quad (51)$$

όπου $N_s = \lfloor (N+1)/2 \rfloor$ είναι ο μεγαλύτερος ακέραιος στο λόγο $(N+1)/2$. Γράφοντας το $H(z)$ στη μορφή αυτή, υποθέσαμε ότι $M \leq N$ και ότι οι πραγματικοί πόλοι και τα πραγματικά μηδενικά συνδυάζονται σε ζεύγη. Αν υπάρχει περιττός αριθμός από πραγματικά μηδενικά, ένας από τους συντελεστές b_{2k} θα είναι μηδέν, ενώ αν υπάρχει περιττός αριθμός από πραγματικούς πόλους, ένας από τους συντελεστές a_{2k} θα είναι μηδέν. Τα επιμέρους δεύτερης τάξης υποσυστήματα μπορούν να υλοποιηθούν είτε με Direct Form I είτε με Direct Form II. Όμως έχουμε δει ότι μπορούμε να φτιάξουμε δομή σε σειρά με τον ελάχιστο δυνατό αριθμό από πολλαπλασιασμούς και τον ελάχιστο δυνατό αριθμό από καθυστερήσεις, αν χρησιμοποιήσουμε την Direct Form II δομή σε κάθε υποσύστημα. Ένα παράδειγμα βης τάξης συστήματος με χρήση δεύτερης τάξης Direct Form II υποσυστημάτων παρουσιάζεται στο Σχήμα (14). Ας δούμε ένα παράδειγμα.

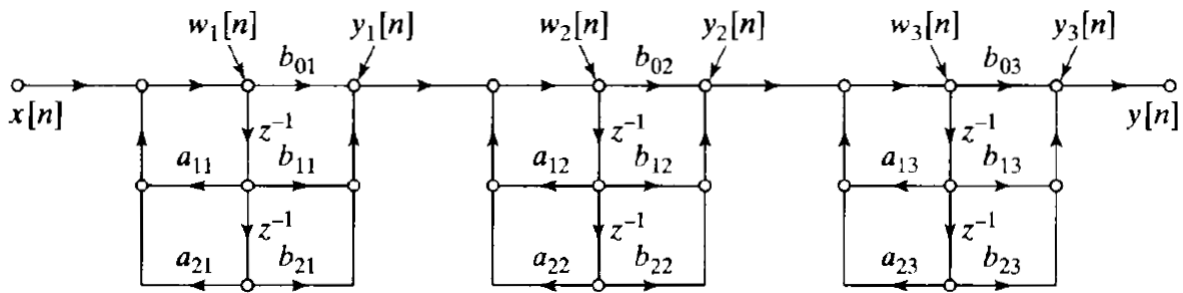
Παράδειγμα:

Έστω το σύστημα

$$H(z) = \frac{1 + 2z^{-1} + z^{-2}}{1 - 0.75z^{-1} + 0.125z^{-2}} = \frac{(1 + z^{-1})(1 + z^{-1})}{(1 - \frac{1}{2}z^{-1})(1 - \frac{1}{4}z^{-1})} \quad (52)$$

Η υλοποίηση σε σειρά με χρήση Direct Form I, II γράφους φαίνεται στο Σχήμα (15).

Ένα τελευταίο σχόλιο σχετικά με τον ορισμό της συνάρτησης μεταφοράς για έναν γράφο σε σειρά... Όπως ορίστηκε το σύστημα της Σχέσης (51), κάθε υποσύστημα δεύτερης τάξης έχει πέντε πολλαπλασιασμούς. Για να δείτε τη σύγκριση, ας υποθέσουμε ότι $M = N$, όπως δίνεται στη Σχέση (48), και επιπλέον,



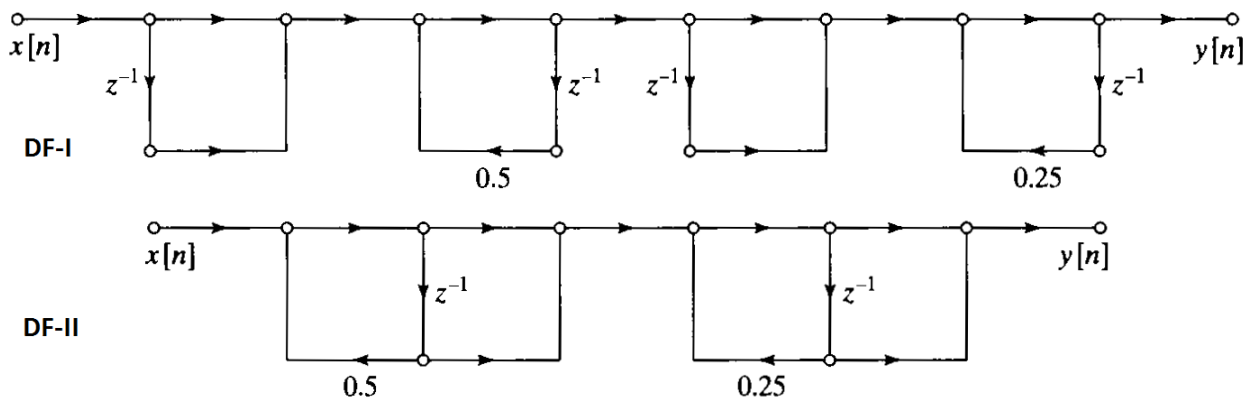
Σχήμα 14: Γράφος 6ης τάξης με δομή σε σειρά, από υποσυστήματα Direct Form II.

ας υποθέσουμε ότι N είναι άρτιος ακέραιος, έτσι ώστε $N_s = N/2$. Οι γράφοι Direct Form I, II θα έχουν $2N + 1$ πολλαπλασιασμούς, ενώ ο γράφος σε σειρά, όπως δίνεται από τη Σχέση (51) έχει $5N/2$ πολλαπλασιασμούς. Για το σύστημα 6ης τάξης του Σχήματος (14), απαιτούνται 15 πολλαπλασιασμοί, ενώ οι αντίστοιχοι Direct Form γράφοι απαιτούν 13 πολλαπλασιασμούς.

Ένας άλλος ορισμός της μορφής σε σειρά είναι

$$H(z) = b_0 \prod_{k=1}^{N_s} \frac{1 + \hat{b}_{1k}z^{-1} + \hat{b}_{2k}z^{-2}}{1 - a_{1k}z^{-1} - a_{2k}z^{-2}} \quad (53)$$

όπου έχουμε διαρέσει κάθε όρο του αριθμητή με το b_{0k} , παίρνοντας έτσι νέους συντελεστές $\hat{b}_{ik} = \frac{b_{ik}}{b_{0k}}$, για $i = 1, 2$ και $k = 1, 2, \dots, N_s$. Αυτή η μορφή του $H(z)$ προτείνει μια σχεδίαση σε σειρά με υποσυστήματα δεύτερης τάξης που απαιτούν τέσσερις πολλαπλασιασμούς, με συνολική σταθερά b_0 για όλο το σύστημα. Αυτή η μορφή σε σειρά έχει τον ίδιο αριθμό πολλαπλασιασμών με τις Direct Form δομές. Όπως θα δούμε πιο κάτω, η έκδοση με τα υποσυστήματα των πέντε πολλαπλασιασμών χρησιμοποιείται συχνά όταν ο γράφος υλοποιείται σε υλικό που χρησιμοποιεί αριθμητική μη κινητής υποδιαστολής (fixed-point), γιατί έτσι μπορεί να διανεμηθεί η σταθερά του συστήματος, και έτσι να ελέγχεται το μέγεθος του σήματος σε διάφορα μέρη του αλγορίθμου που είναι κρίσιμης σημασίας. Όταν χρησιμοποιείται αριθμητική κινητής υποδιαστολής (floating point) και το εύρος του μεγέθους του σήματος δεν αποτελεί θέμα, η έκδοση με τα υποσυστήματα των τεσσάρων πολλαπλασιασμών μπορεί να χρησιμοποιηθεί για να μειώσει την ποσότητα των υπολογισμών. Επιπλέον, αν υπάρχουν μηδενικά επάνω στο μοναδιαίο κύκλο, τότε $\hat{b}_{2k} = 1$ και τότε απαιτούνται μολις τρεις πολλαπλασιασμοί ανά υποσύστημα δεύτερας τάξης.



Σχήμα 15: Γράφος σε σειρά με υποσυστήματα Direct Form I, II.

4.3 Παράλληλη Μορφή - Parallel Form

Στη συζήτησή μας όταν αναλύαμε συστήματα στο χώρο του Z , χρησιμοποιούσαμε συχνά το Ανάπτυγμα σε Μερικά Κλάσματα. Ας δούμε αν αυτό το ανάπτυγμα μας προτείνει κάποια δομή σχεδίασης που είναι χρήσιμη. Ας θυμηθούμε ότι η Σχέση (48) μπορεί να γραφεί ως

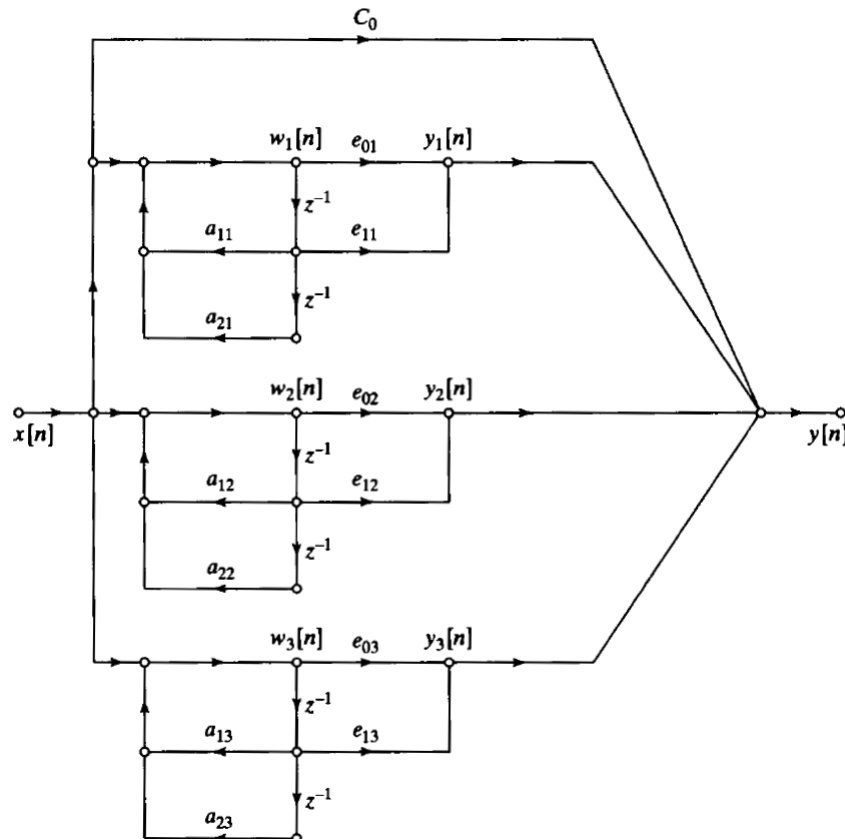
$$H(z) = \sum_{k=0}^{N_p} C_k z^{-k} + \sum_{k=1}^{N_1} \frac{A_k}{1 - c_k z^{-1}} + \sum_{k=1}^{N_2} \frac{B_k (1 - e_k z^{-1})}{(1 - d_k z^{-1})(1 - d_k^* z^{-1})} \quad (54)$$

με $N = N_1 + 2N_2$. Αν $M \geq N$, τότε $N_p = M - N$, και άρα ο πρώτος όρος υπάρχει, ειδάλλως $N_p < 0$ και τότε τον απορρίπτουμε από την παραπάνω σχέση. Αν οι συντελεστές a_k, b_k της Σχέσης (48) είναι πραγματικοί αριθμοί, τότε οι συντελεστές A_k, B_k, C_k, c_k, e_k του αναπτύγματος είναι πραγματικοί αριθμοί. Σε αυτήν την περίπτωση, η συνάρτηση μεταφοράς μπορεί να ερμηνευθεί ως μια αναπαράσταση ενός παράλληλου συνδυασμού ΠΙΡ υποσυστημάτων πρώτης και δεύτερης τάξης, με την πιθανότητα να υπάρχουν επιπλέον N_p κλάδοι καθυστέρησης της εισόδου. Εναλλακτικά, μπορούμε να ομαδοποιήσουμε τους πραγματικούς πόλους σε ζεύγη, έτσι ώστε το $H(z)$ να γραφεί ως

$$H(z) = \sum_{k=0}^{N_p} C_k z^{-k} + \sum_{k=1}^{N_s} \frac{e_{0k} + e_{1k} z^{-1}}{1 - a_{1k} z^{-1} - a_{2k} z^{-2}} \quad (55)$$

όπου $N_s = \lfloor (N + 1)/2 \rfloor$. Ξανά, αν $N_p = M - N < 0$, ο πρώτος όρος δεν περιλαμβάνεται στην παραπάνω σχέση.

Ένα γενικό παράδειγμα σχεδίασης παράλληλης μορφής φαίνεται στο Σχήμα (16), για ένα σύστημα $M = N = 6$ τάξης. Μπορείτε να βρείτε τις εξισώσεις διαφορών που τον αποτελούν; :-)



Σχήμα 16: Παράλληλος γράφος 6ης τάξης.

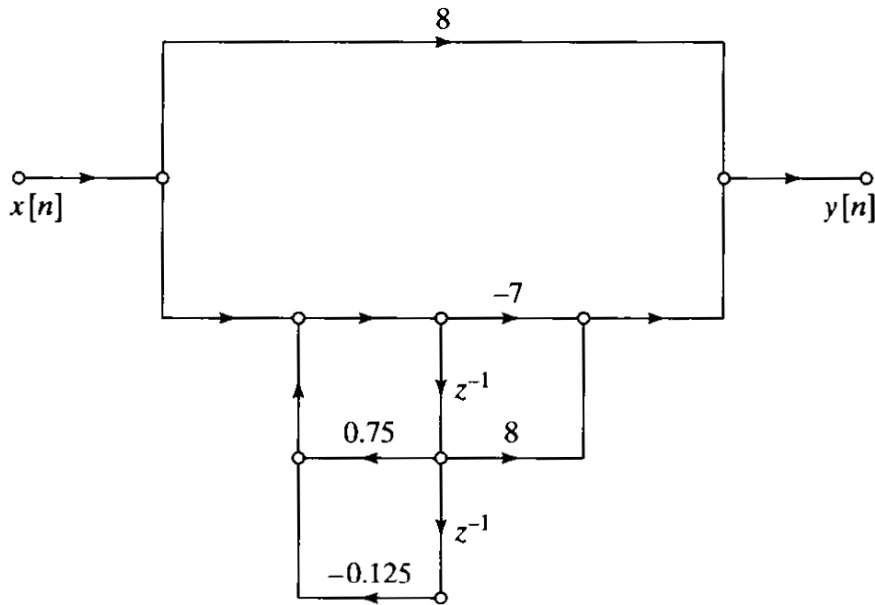
Παράδειγμα:
Έστω το σύστημα

$$H(z) = \frac{1 + 2z^{-1} + z^{-2}}{1 - 0.75z^{-1} + 0.125z^{-2}} \quad (56)$$

Το Ανάπτυγμα σε Μερικά Κλάσματα είναι

$$H(z) = 8 + \frac{-7 + 8z^{-1}}{1 - 0.75z^{-1} + 0.125z^{-2}} \quad (57)$$

και μπορεί να υλοποιηθεί ως Parallel Form γράφος όπως στο Σχήμα 17. Επίσης, το ίδιο σύστημα μπορεί



Σχήμα 17: Parallel Form I παραδείγματος.

να γραφεί ως άθροισμα στοιχείων πρώτης τάξης, δηλ. ως

$$H(z) = 8 + \frac{18}{1 - 0.5z^{-1}} - \frac{25}{1 - 0.25z^{-1}} \quad (58)$$

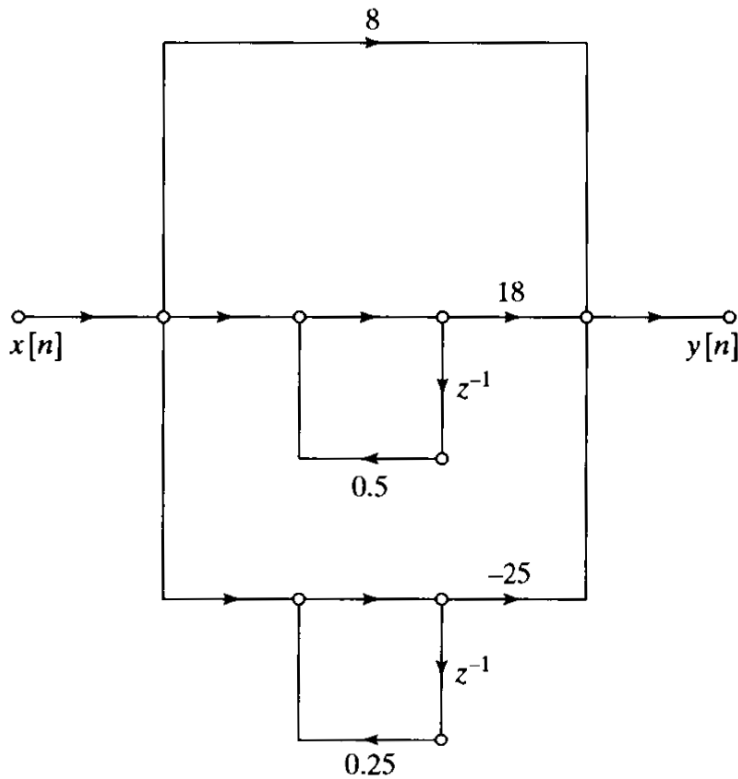
το οποίο μπορεί να υλοποιηθεί ως Parallel Form γράφος όπως φαίνεται στο Σχήμα 18.

4.4 Ανάδραση στα IIR Συστήματα

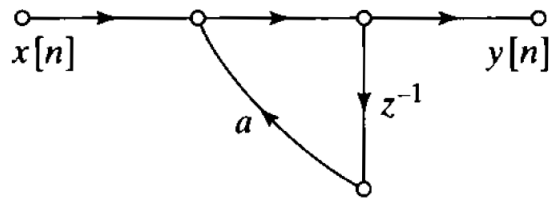
Προτού αφήσουμε τη συζήτησή μας για γράφους IIR συστημάτων, ας επισημάνουμε μερικά πράγματα. Ίσως προσέξατε ότι όλοι οι γράφοι σε αυτήν την παράγραφο είχαν βρόχους ανάδρασης, δηλ. είχαν κλειστά μονοπάτια που ξεκινούσαν από έναν κόμβο και κατέληγαν ξανά στον ίδιο κόμβο, μέσα από άλλους κλάδους. Μια τέτοια δομή στο γράφο υπονοεί ότι μια μεταβλητή κόμβου που εμπλέκεται σε μια ανάδραση εξεργάζεται άμεσα ή έμμεσα από τον εαυτό της. Ας πάρουμε ένα απλό παράδειγμα. Έστω η εξίσωση διαφορών

$$y[n] = ay[n - 1] + x[n] \quad (59)$$

η οποία αναπαρίσταται από το γράφο του Σχήματος (19). Τέτοιοι βρόχοι ανάδρασης είναι αναγκαίοι - αλλά όχι ικανοί - να παράγουν άπειρες σε διάρκεια κρουστικές αποκρίσεις. Αυτό μπορείτε να το καταλάβετε αν σκεφτείτε ένα γράφο χωρίς βρόχους ανάδρασης. Σε μια τέτοια περίπτωση, οποιοδήποτε μονοπάτι από



Σχήμα 18: Parallel Form II παραδείγματος.



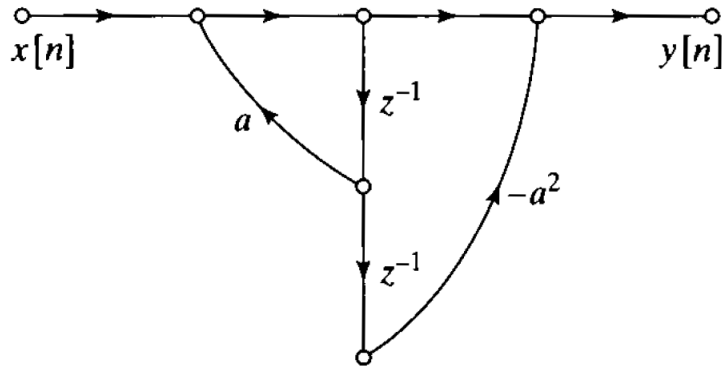
Σχήμα 19: Σύστημα με βρόχο ανάδρασης.

την είσοδο στην έξοδο περνά από κάθε στοιχείο καθυστέρησης μόνο μια φορά. Έτσι, η μεγαλύτερη καθυστέρηση που μπορεί να συμβεί είναι αυτή που η είσοδος περνάει από το μονοπάτι που περιλαμβάνει όλες τις καθυστερήσεις του συστήματος. Οπότε, αν δεν υπάρχει βρόχος ανάδρασης, η κρουστική απόκριση δεν μπορεί να είναι μεγαλύτερη από το συνολικό αριθμό των στοιχείων καθυστέρησης στο γράφο. Από αυτό μπορούμε εύκολα να συμπεράνουμε ότι αν δεν έχουμε βρόχους ανάδρασης, τότε ο γράφος υλοποιεί ένα σύστημα που έχει μόνο μηδενικά (και φυσικά πόλους μόνο στο $z = 0$), και το πλήθος αυτών των μηδενικών δεν μπορεί να είναι μεγαλύτερο από τον αριθμό των καθυστερήσεων στο γράφο.

Ας επανέλθουμε όμως στο παράδειγμά μας. Σκεφτείτε ότι στο γράφο του Σχήματος (19) παρουσιάζεται ως είσοδος το σήμα $x[n] = \delta[n]$. Γνωρίζετε ασφαλώς ότι η έξοδος θα είναι η κρουστική απόκριση $h[n]$ του συστήματος. Ας ακολουθήσουμε όμως το ταξίδι της συνάρτησης $x[n] = \delta[n]$ μέσα στο γράφο. Βλέπουμε ότι το $\delta[n]$ συνεχώς κυκλοφορεί μέσα στο βρόχο ανάδρασης, με αύξον (αν $|a| > 1$) ή φθίνον (αν $|a| < 1$) πλάτος, έτσι ώστε η κρουστική απόκριση να είναι $h[n] = a^n u[n]$. ΜΕ αυτόν τον τρόπο, ένας βρόχος ανάδρασης δημιουργεί μια άπειρης διάρκειας κρουστική απόκριση.

Άρα, αν ένα σύστημα έχει πόλους, ο γράφος του σίγουρα θα έχει βρόχους ανάδρασης. Από την άλλη, αν ένας γράφος έχει βρόχους ανάδρασης, τότε δε σημαίνει απαραίτητα ότι περιγράφει ένα *IIP* σύστημα. Επίσης,

όπως ήδη ξέρετε, ούτε η παρουσία πόλων στη συνάρτηση μεταφοράς σημαίνει απαραίτητα ότι η κρουστική απόκριση που περιγράφει το σύστημα είναι άπειρης διάρκειας. Καταλαβαίνετε γιατί; Η απάντηση βρίσκεται στο γεγονός ότι μπορεί οι πόλοι να ακυρώνονται από τα μηδενικά του συστήματος! Το Σχήμα (20) δείχνει ένα γράφο με βρόχο ανάδρασης. Είναι όμως αυτό το σύστημα άπειρης διάρκειας κρουστικής απόκρισης; Η



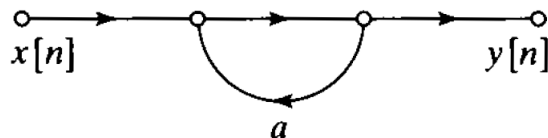
Σχήμα 20: FIR σύστημα με βρόχο ανάδρασης.

συνάρτηση μεταφοράς του δίνεται από

$$H(z) = \frac{1 - a^2 z^{-2}}{1 - az^{-1}} = \frac{(1 - az^{-1})(1 + az^{-1})}{1 - az^{-1}} = 1 + az^{-1} \longleftrightarrow h[n] = \delta[n] + a\delta[n - 1] \quad (60)$$

που είναι πεπερασμένης διάρκειας κρουστική απόκριση! Αυτό το σύστημα είναι ένα απλό παράδειγμα μιας γενικότερης κατηγορίας FIR συστημάτων που λέγονται *συστήματα δειγματοληψίας συχνότητας - frequency-sampling systems*.

Κλείνοντας, αξίζει να πούμε ότι οι βρόχοι ανάδρασης είναι ιδιαίτερα προβληματικοί όταν χρειαστεί να υλοποιηθούν. Όπως έχουμε ήδη δει, πρέπει να μπορούμε να υπολογίζουμε τις ενδιάμεσες μεταβλητές κόμβων με τέτοια σειρά ώστε όλες οι αναγκαίες τιμές να είναι διαθέσιμες όταν χρειαστεί. Σε κάποιες περιπτώσεις, δεν υπάρχει τρόπος διάταξης των βημάτων των υπολογισμών ώστε αυτό να συμβαίνει. Τέτοιοι γράφοι λέγονται *μη-υπολογίσιμοι - non-computable*. Ένας απλός μη υπολογίσιμος γράφος φαίνεται στο Σχήμα (21). Η εξίσωση διαφορών για αυτόν το γράφο είναι



Σχήμα 21: Μη υπολογίσιμο σύστημα.

$$y[n] = ay[n] + x[n] \quad (61)$$

Προφανώς δεν μπορούμε να υπολογίσουμε το $y[n]$ γιατί στο δεξιό μέλος της εξίσωσης υπάρχει ξανά η ποσότητα αυτή. Το ότι ο γράφος είναι μη υπολογίσιμος δε σημαίνει ότι δεν μπορούμε και να λύσουμε την εξίσωση. Όπως καταλαβαίνετε, η λύση της παραπάνω εξίσωσης είναι απλά

$$y[n] = \frac{1}{1 - a} x[n] \quad (62)$$

Άρα το κλειδί για την υπολογισιμότητα των γράφων είναι ότι κάθε βρόχος ανάδρασης πρέπει να έχει ένα τουλάχιστον στοιχείο καθυστέρησης.

5 Ανάστροφη Μορφή - Transposed Form

Η θεωρία Γράφων μας παρέχει πολλούς τρόπους με τους οποίους μπορούμε να αλλάζουμε - μετασχηματίζουμε τους γράφους που έχουμε ήδη συζητήσει, χωρίς να αλλάζουν το σύστημα που υλοποιούν. Μια τέτοια διαδικασία ονομάζεται *αναστροφή γράφου* - *graph reversal*, και μας παρέχει μερικές καλές εναλλακτικές μορφές γράφων, με ενδιαφέρουσες ιδιότητες. Ας τις δούμε.

Οι κανόνες για την εύρεση του ανάστροφου γράφου (οι οποίοι δεν αποδεικνύονται εδώ¹) είναι οι εξής:

1. Αναστρέφουμε τη φορά όλων των κλάδων.
2. Κάθε κόμβος άθροισης μετατρέπεται σε κόμβο διακλάδωσης, και το αντίστροφο. Ελέγχουμε τις ενδιάμεσες μεταβλητές για ενδεχόμενες αλλαγές.
3. Αλλάζουμε θέση μεταξύ εισόδου και εξόδου.
4. Σχεδιάζουμε το γράφο ξανά, αναστρέφοντας τον ώστε να παρουσιάζεται η είσοδος στα αριστερά.

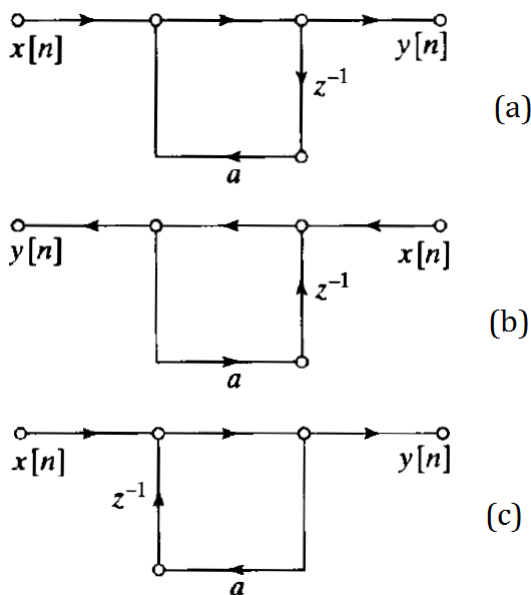
Ας το δούμε στην πράξη σε ένα παράδειγμα.

Παράδειγμα:

Έστω το απλό σύστημα

$$H(z) = \frac{1}{1 - az^{-1}} \quad (63)$$

Για να βρούμε τον ανάστροφο γράφο, ακολουθούμε τους κανόνες, οι οποίοι δίνουν διαδοχικά τους γράφους του Σχήματος (22).



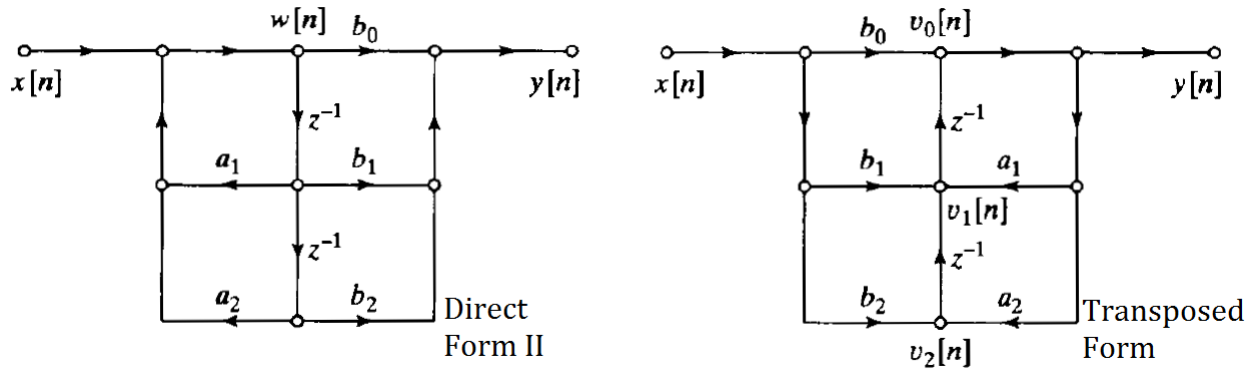
Σχήμα 22: Εύρεση ανάστροφου γράφου: (a): αρχικός γράφος, (b): ανάστροφη μορφή με χρήση κανόνων 1 – 3, (c): επανασχεδίαση με είσοδο στα αριστερά.

Ας το επαναλάβουμε σε ένα πιο σύνθετο παράδειγμα.

Παράδειγμα:

Έστω ο Direct Form II γράφος του Σχήματος (23), που αντιστοιχεί σε ένα σύστημα δευτέρας τάξης με πόλους και μηδενικά. Εφαρμόζοντας ξανά τους κανόνες 1 – 4, καταλήγουμε στον ανάστροφο γράφο του

¹ Δείτε τις εργασίες των Mason, Zimmermann, 1960 και Phillips, Nagle, 1995.



Σχήμα 23: Direct Form II και ανάστροφος γράφος παραδείγματος δευτέρας τάξης.

Σχήματος (23). Μπορείτε να δείξετε ότι οι εξισώσεις διαφορών των δυο γράφων περιγράφουν το ίδιο σύστημα;

Η εφαρμογή των κανόνων για την εύρεση του αναστρόφου μπορεί να γίνει σε οποιονδήποτε από τους γράφους που συναντήσαμε ως τώρα. Ας γενικεύσουμε αυτήν την εφαρμογή, ξεκινώντας από έναν Direct Form II γράφο, και καταλήγοντας σε έναν transposed γράφο. Το Σχήμα (24) δείχνει τη μετατροπή αυτή σε ένα γενικότερο πλαίσιο. Όμοια συμβαίνει και η μετατροπή ενός Direct Form I γράφου σε transposed γράφο. Μπορούμε να κάνουμε μια ενδιαφέρουσα παρατήρηση αν συγκρίνουμε τους δυο γράφους του Σχήματος (24). Ενώ στο Direct Form II γράφο, υλοποιούμε πρώτα τους πόλους και μετά τα μηδενικά, βλέπουμε ότι στον transposed γράφο υλοποιούμε πρώτα τα μηδενικά και μετά τους πόλους, χωρίς την επιπλέον χρήση μνήμης που είχαμε όταν το κάναμε αυτό στη Direct Form I σχεδίαση! Αυτό είναι μερικές φορές σημαντικό όταν υλοποιούμε αυτά τα συστήματα με αναλογικά ή ψηφιακά κυκλώματα.

Τέλος, η αναστροφή γράφου τονίζει ακόμα περισσότερο την άπειρη ποικιλία των δομών υλοποίησης που υπάρχει για ένα δεδομένο σύστημα που περιγράφεται από μια ρητή συνάρτηση μεταφοράς. Η αναστροφή μας παρέχει μια απλή διαδικασία για τη δημιουργία νέων δομών συστημάτων. Τα προβλήματα που προέκυψαν από την υλοποίηση τέτοιων συστημάτων σε μηχανές πεπερασμένης ακρίβειας έδωσαν κίνητρο στους ερευνητές να αναπτύξουν ακόμα περισσότερες κατηγορίες δομών, που δεν είναι βολικό να συζητήσουμε εδώ, καθώς μας αρκεί να γνωρίζουμε τις πιο συχνά χρησιμοποιούμενες δομές συστημάτων.

6 Βασικές Δομές FIR Συστημάτων

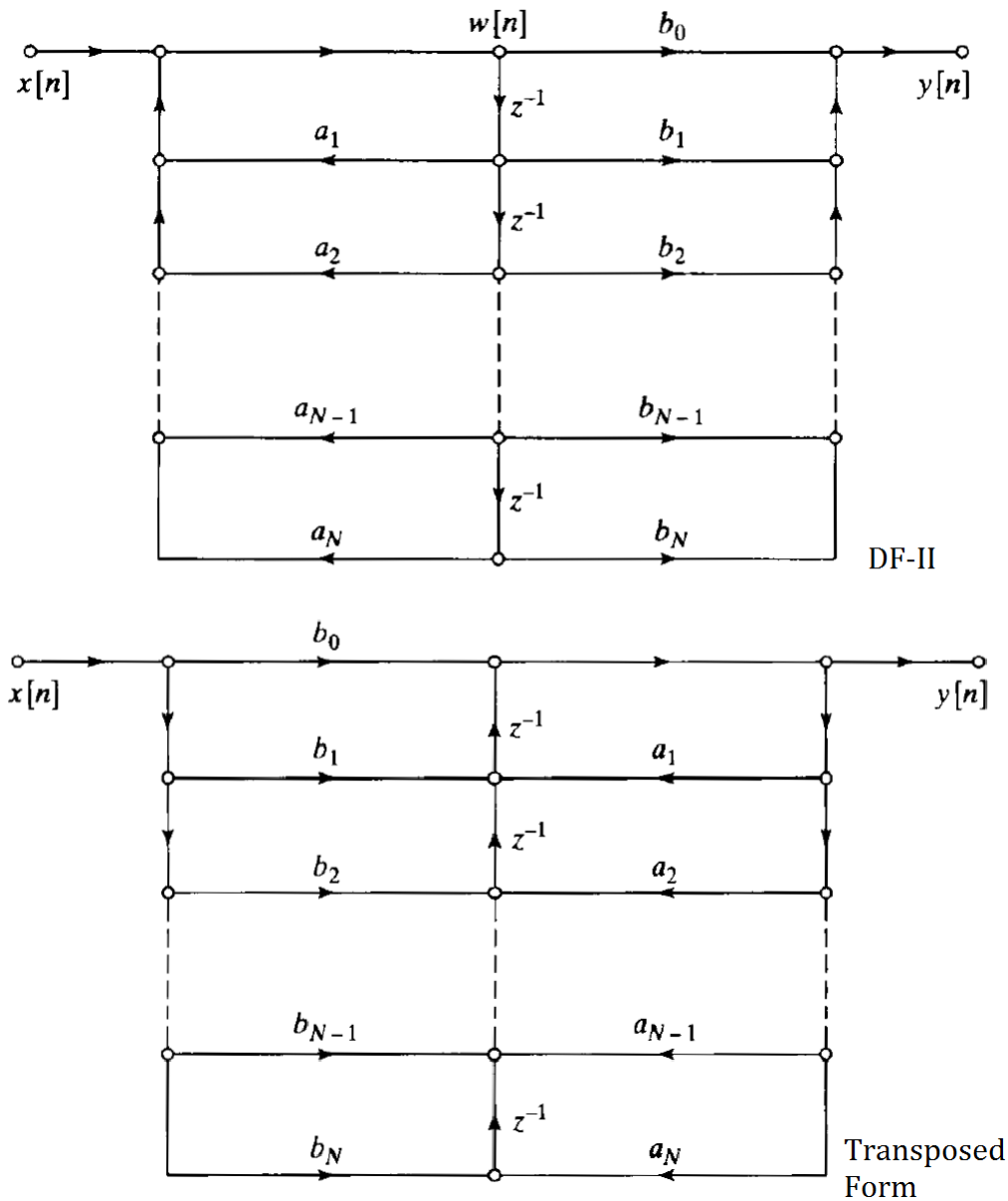
Για τα FIR συστήματα, τα πράγματα είναι σχετικά απλούστερα. Παρ'οτι οι δομές Direct Form, σε σειρά, και παράλληλα που συζητήσαμε προηγουμένως αφορούν IIR συστήματα, τα οποία φυσικά περιέχουν τα FIR ως ειδική περίπτωση, υπάρχουν ανεξάρτητες δομές, ειδικές για FIR συστήματα.

6.1 Direct Forms

Για αιτιατά FIR συστήματα, η συνάρτηση μεταφοράς έχει μόνο μηδενικά, και πόλους μόνο στο $z = 0$. Οπότε η εξίσωση διαφορών γράφεται ως

$$y[n] = \sum_{k=0}^M b_k x[n - k] \quad (64)$$

Σε αυτήν την περίπτωση, οι μορφές Direct Form I, II που έχουμε ήδη δει, μετατρέπονται σε μια μόνο Direct Form, αυτή του Σχήματος (25), η οποία παρουσιάζεται μαζί με την ανάστροφη μορφή της, ακολουθώντας τους κανόνες που περιγράψαμε νωρίτερα. Η Direct Form σχεδίαση λέγεται και Transversal Form.



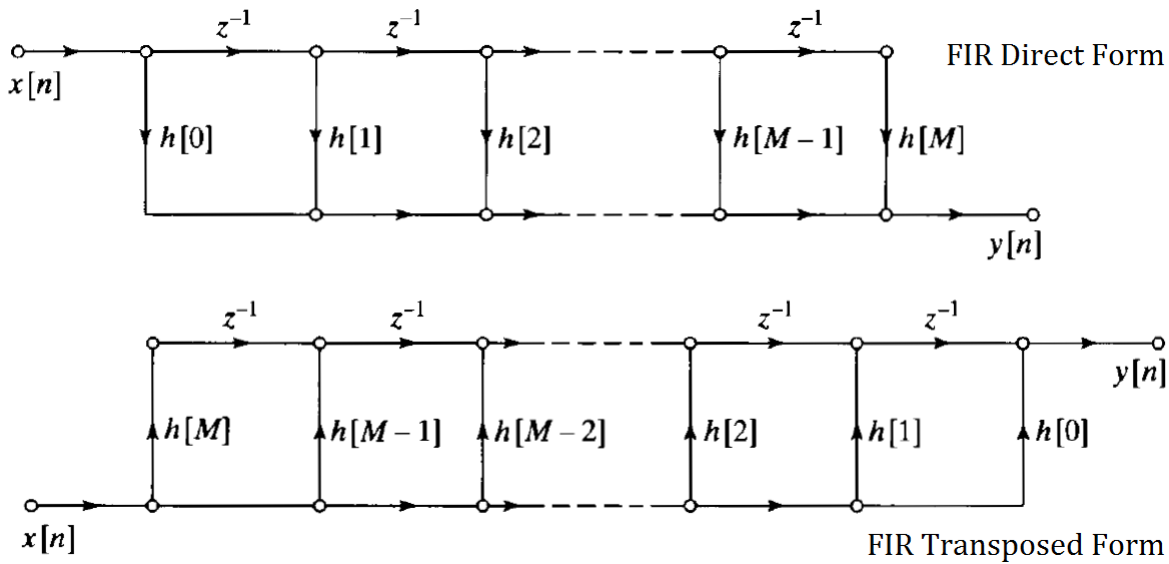
Σχήμα 24: Direct Form II και ανάστροφος γράφος.

6.2 Μορφή σε Σειρά - Cascade Form

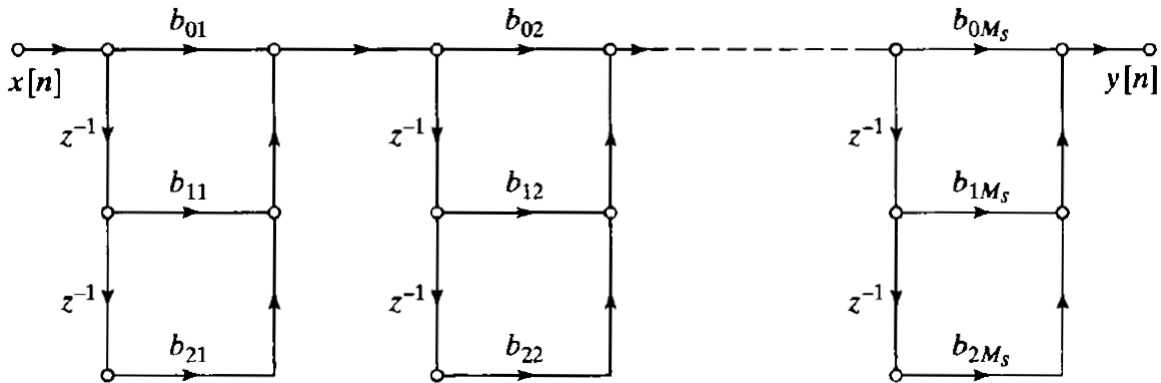
Η μορφή σε σειρά ενός FIR συστήματος προέρχεται από την παραγοντοποίηση του πολυωνύμου της συνάρτησης μεταφοράς, δηλ.

$$H(z) = \sum_0^M h[n]z^{-n} = \prod_{k=1}^{M_s} (b_{0k} + b_{1k}z^{-1} + b_{2k}z^{-2}) \quad (65)$$

όπου $M_s = \lfloor (M+1)/2 \rfloor$. Αν το M είναι περιττό, τότε ένας από τους συντελεστές b_{2k} θα είναι μηδέν. Μια γενική αναπαράσταση της μορφής σε σειρά φαίνεται στο Σχήμα (26).



Σχήμα 25: Direct Form και ανάστροφος γράφος για ένα FIR σύστημα.



Σχήμα 26: Μορφή σε σειρά για ένα FIR σύστημα.

6.3 Δομές για Συστήματα Γραμμικής Φάσης

Μια ειδική κατηγορία FIR συστημάτων είναι αυτά που έχουν γραμμική φάση. Έχουμε δείξει ότι τα συστήματα αυτά ικανοποιούν κάποια συμμετρία, όπως

$$h[M - n] = h[n], \quad n = 0, 1, \dots, M \quad (66)$$

για τα Τύπου I, II, ή

$$h[M - n] = -h[n], \quad n = 0, 1, \dots, M \quad (67)$$

για τα Τύπου III, IV. Οποιαδήποτε κι αν ισχύει από τις δυο συμμετρίες, ο αριθμός των πολλαπλασιασμών που απαιτούνται για την υλοποίησή τους μπορεί να μειωθεί σχεδόν στο μισό! Για να το δούμε αυτό, ας θεωρήσουμε ότι το M είναι άρτιος ακέραιος, και αντιστοιχεί σε ένα Τύπου I ή III σύστημα. Τότε:

$$y[n] = \sum_{k=0}^M h[k]x[n - k] \quad (68)$$

$$= \sum_{k=0}^{M/2-1} h[k]x[n - k] + h[M/2]x[n - M/2] + \sum_{k=M/2+1}^M h[k]x[n - k] \quad (69)$$

$$= \sum_{k=0}^{M/2-1} h[k]x[n-k] + h[M/2]x[n-M/2] + \sum_{k=0}^{M/2-1} h[M-k]x[n-M+k] \quad (70)$$

Για Τύπου I συστήματα, έχουμε

$$y[n] = \sum_{k=0}^{M/2-1} h[k](x[n-k] + x[n-M+k]) + h[M/2]x[n-M/2] \quad (71)$$

ενώ για Τύπου III συστήματα, έχουμε

$$y[n] = \sum_{k=0}^{M/2-1} h[k](x[n-k] + x[n-M+k]) \quad (72)$$

Αν δουλέψουμε αντίστοιχα για τα Τύπου II, IV συστήματα (με M περιττό δηλαδή), θα καταλήξουμε ότι για τα Τύπου II συστήματα, έχουμε

$$y[n] = \sum_{k=0}^{(M-1)/2} h[k](x[n-k] + x[n-M+k]) \quad (73)$$

ενώ για τα Τύπου IV, θα είναι

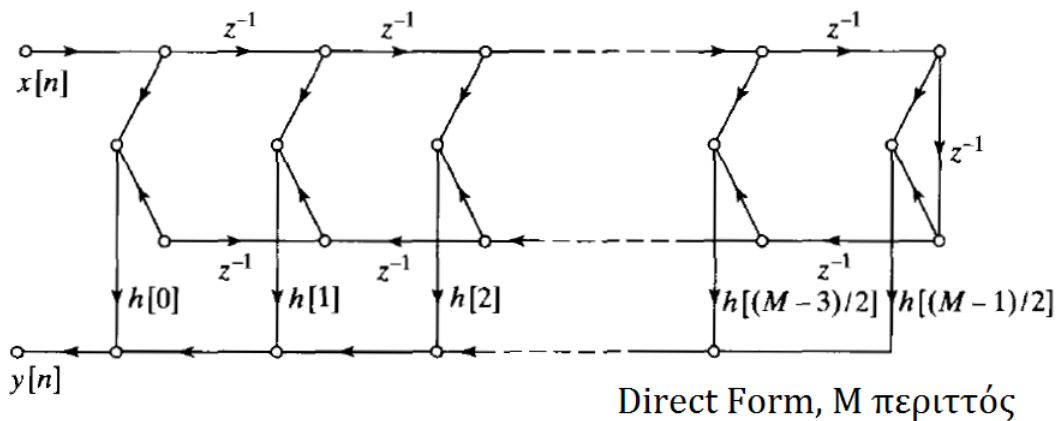
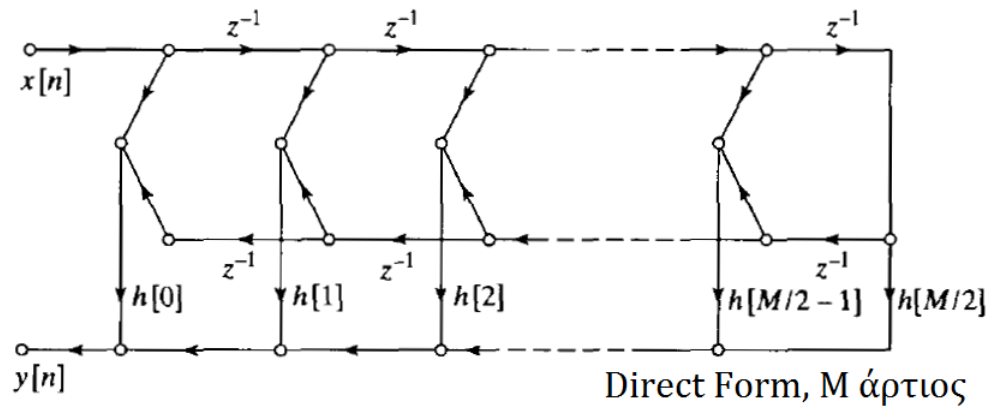
$$y[n] = \sum_{k=0}^{(M-1)/2} h[k](x[n-k] - x[n-M+k]) \quad (74)$$

Προσέξτε ότι τα παραπάνω συστήματα έχουν $M/2 + 1$, $M/2$ ή $(M + 1)/2$ πολλαπλασιασμούς, αντί για M πολλαπλασιασμούς που θα είχαν αν τα υλοποιούσαμε σε γενική Direct Form. Οι γράφοι των παραπάνω συστημάτων φαίνονται στο Σχήμα (27).

Ας θυμηθούμε λίγο ξανά τη συζήτηση που κάναμε για τα FIR συστήματα γραμμικής φάσης. Είχαμε δει τότε ότι η συμμετρία τους αναγκάζει το διάγραμμα πόλων-μηδενικών να είναι μιας συγκεκριμένης μορφής. Θυμηθείτε ότι αν υπήρχε ένα μηδενικό στη θέση z_0 , τότε υπήρχε κι ένα μηδενικό στη θέση $1/z_0$. Αν επιπλέον απαιτούσαμε το $H(z)$ να ανταποκρίνεται σε πραγματικό $h[n]$, τότε κάθε ένα από τα μηδενικά είχε και το συζυγή του. Κατά συνέπεια, τα πραγματικά μηδενικά βρίσκονταν σε αμοιβαίες θέσεις επάνω στον άξονα των πραγματικών. Τα μιγαδικά μηδενικά εκτός μοναδιαίου κύκλου έρχονταν σε ομάδες των τεσσάρων, με συζυγή αμοιβαίες θέσεις για όλα. Τέλος, ένα μηδενικό επάνω στο μοναδιαίο κύκλο, είχε μόνο το συζυγές του (που είναι και το αμοιβαίο του). Άρα, τα μιγαδικά μηδενικά επάνω στο μοναδιαίο κύκλο μπορούμε να τα θεωρήσουμε ως ζεύγη. Αυτός ο διαχωρισμός θα μας είναι χρήσιμος όταν μιλήσουμε αργότερα για χβαντισμό των συντελεστών με σκοπό τη διατήρηση της γραμμικής φάσης.

7 Κβαντισμός και Ακρίβεια

Όλη η συζήτησή μας γύρω από την υλοποίηση ΓΧΑ συστημάτων με διάφορες μορφές είχε ως βάση το γεγονός ότι διαφορετικές δομές υλοποίησης, που θεωρητικά είναι ισοδύναμες, συμπεριφέρονται τελείως διαφορετικά όταν υλοποιούνται με αριθμητική πεπερασμένης ακρίβειας. Εν γένει, οι μορφές σε σειρά και παράλληλα είναι πιο εύρωστες στον κβαντισμό των συντελεστών τους, ενώ οι direct forms είναι πολύ πιο ευαίσθητες. Οι τελευταίες σχεδόν ποτέ δεν υλοποιούν συστήματα μεγαλύτερης τάξης από τη δεύτερη (δευτεροβάθμια πολυώνυμα). Ας κάνουμε μια μικρή εισαγωγή στα προβλήματα αυτά και στην αντιμετώπισή τους.



Σχήμα 27: Direct Forms για ένα FIR σύστημα γραμμικής φάσης.

7.1 Κβαντισμός

Η επιρροή του κβαντισμού ενός συντελεστή ενός αιτιατού κι ευσταθούς ΓΧΑ συστήματος, όπως ο πραγματικός συντελεστής (πόλος) a στη συνάρτηση μεταφοράς

$$H(z) = \frac{1}{1 - az^{-1}} \quad (75)$$

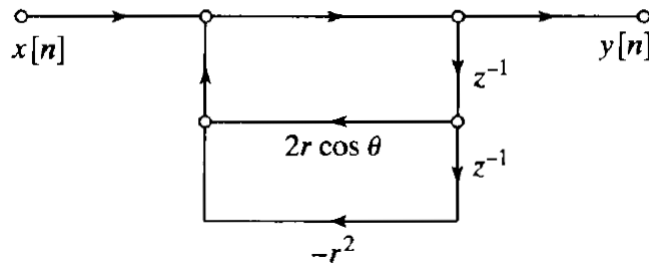
είναι ξεχωριστά διαχειρίσιμη από την υλοποίηση της εξίσωσης διαφορών. Αυτό σημαίνει ότι ο άπειρης ακρίβειας συντελεστής a αντικαθίσταται από την κβαντισμένη τιμή του, και η συνάρτηση μεταφοράς που προκύπτει ελέγχεται αν ικανοποιεί τις αρχικές προδιαγραφές ή αν ο κβαντισμός υποβάθμισε την απόδοση του συστήματος σε επίπεδα μη αποδεκτά. Για παράδειγμα, αν έχουμε στη διάθεσή μας $B + 1$ bits για τον κβαντισμό του συντελεστή a , τότε πρέπει να εξετάσουμε αν η συνάρτηση μεταφοράς

$$\hat{H}(z) = \frac{1}{1 - \hat{a}z^{-1}} \quad (76)$$

που χρησιμοποιεί τον κβαντισμένο συντελεστή \hat{a} , είναι αρκετά κοντά στην επιθυμητή συνάρτηση μεταφοράς. Ξέρουμε ότι υπάρχουν μόνο 2^{B+1} διαφορετικοί $B + 1$ -αδικοί αριθμοί, και άρα ο πόλος a του $H(z)$ μπορεί να πραγματοποιηθεί μόνο σε 2^{B+1} θέσεις του πραγματικού άξονα του μιγαδικού επιπέδου. Σκεφτείτε ότι όσο πιο λίγα bits διαθέτουμε, τόσο ευκολότερα μπορεί να κβαντιστεί ο συντελεστής a σε αριθμό εκτός του μοναδιαίου κύκλου, και το σύστημά μας να γίνει ασταθές!!

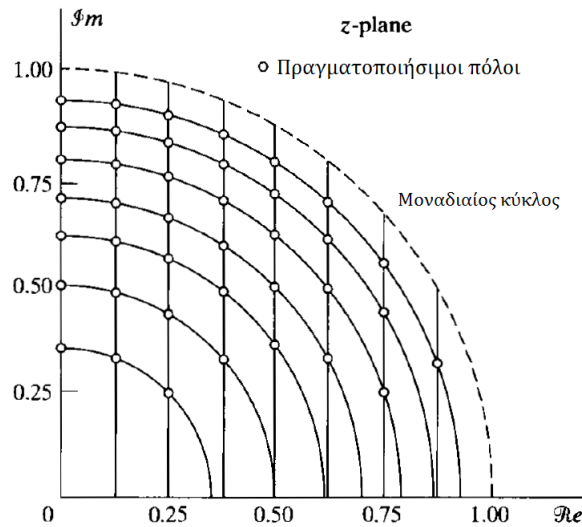
7.2 Κβαντισμός σε Συστήματα Δεύτερης Τάξης

Ας θεωρήσουμε ένα συζυγές ζεύγος πόλων που υλοποιείται ως Direct Form, όπως το Σχήμα (28). Με αριθμητική άπειρης ακρίβειας, ο γράφος αυτός έχει πόλους στο $z = re^{j\theta}$ και $z = re^{-j\theta}$. Όμως, αν



Σχήμα 28: Direct Form για ένα ζεύγος συζυγών πόλων.

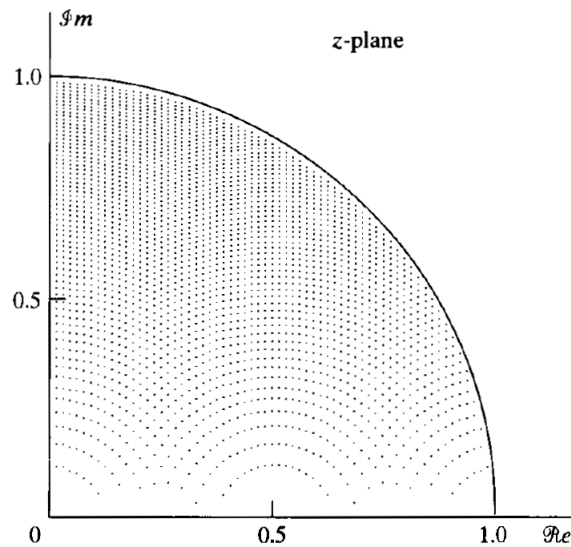
κβαντιστούν οι συντελεστές $2r \cos(\theta)$ και $-r^2$, έχουμε μόνο έναν πεπερασμένο αριθμό από πιθανές θέσεις που μπορούν να πάρουν οι πόλοι στο μιγαδικό επίπεδο. Οι πόλοι πρέπει να βρίσκονται στο πλέγμα του μιγαδικού επιπέδου που ορίζεται από τις τομές μεταξύ των ομόκεντρων κύκλων (που ανταποκρίνονται στον κβαντισμό του r^2) και των κάθετων γραμμών (που αντιστοιχούν στον κβαντισμό του $2r \cos(\theta)$). Ένα τέτοιο



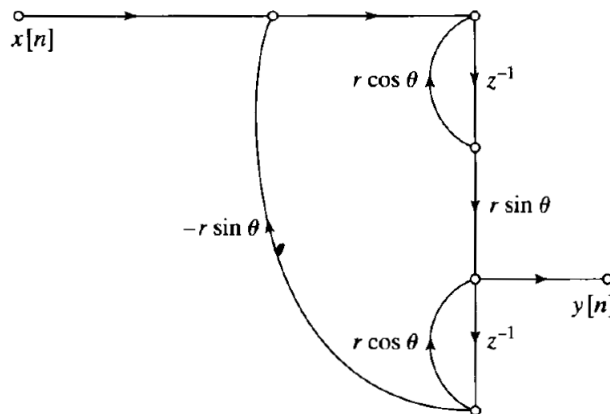
Σχήμα 29: Κβαντισμός με τέσσερα bits.

πλέγμα φαίνεται στο Σχήμα (29) για κβαντισμό 4 bits (3 για την τιμή, και ένα bit για το πρόσημο). Έτσι, το r^2 περιορίζεται σε 8 διαφορετικές τιμές (7 θετικές συν τη μηδενική), ενώ το $2r \cos(\theta)$ σε 7 διαφορετικές θετικές τιμές, 8 αρνητικές, και τη μηδενική: σύνολο 16 τιμές. Το Σχήμα (30) δείχνει μια πιο πυκνή κβάντιση που αντιστοιχεί σε 7 bits (6 bits για την τιμή, και ένα για το πρόσημο). Προσέξτε ότι το πλέγμα είναι ιδιαίτερα αραιό κοντά στον πραγματικό άξονα. Έτσι, πιθανοί πόλοι που βρίσκονται κοντά στο $\theta = 0$ ή $\theta = \pi$ ίσως μετατοπιστούν περισσότερο από κάποιους κοντά στο $\theta = \pi/2$. Γενικότερα λοιπόν, ο κβαντισμός είναι ένα πρόβλημα και μπορεί να θεωρήσει κανείς ότι θα μειώσει την απόδοση ενός συστήματος.

Μια εναλλακτική δομή για την πραγματοποίηση των πόλων $z = re^{\pm j\theta}$ φαίνεται στο Σχήμα (31). Η μορφή αυτή λέγεται coupled form. Για την υλοποίηση αυτού του συστήματος, πρέπει να κβαντίσουμε τις τιμές $r \cos(\theta)$ και $r \sin(\theta)$. Αυτές οι τιμές αντιστοιχούν στις καρτεσιανές συντεταγμένες των πόλων, δηλ. στο πραγματικό και φανταστικό μέρος τους. Οπότε, οι κβαντισμένες τιμές των πόλων πρέπει να βρίσκονται στην τομή των οριζώντιων και κάθετων γραμμών του Σχήματος (32), για μια υλοποίηση με 4 bits. Αντίθετα, για



Σχήμα 30: Κβαντισμός με επτά bits.



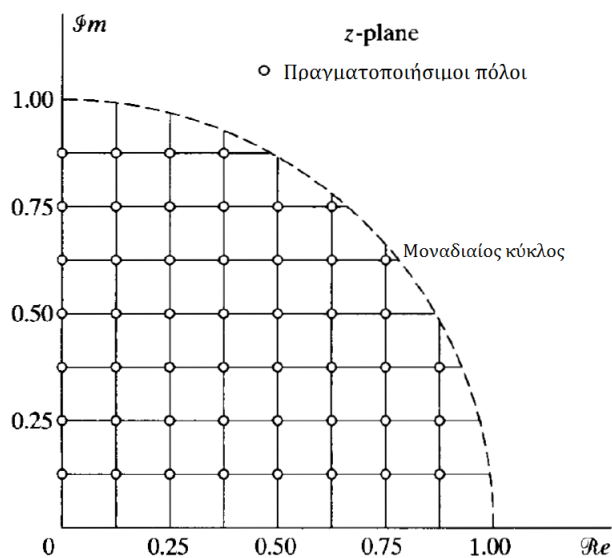
Σχήμα 31: Coupled Form γράφος.

μια υλοποίηση με 7 bits, οι πιθανές θέσεις των πόλων φαίνονται στο Σχήμα (33). Σε αυτήν την περίπτωση, η πυκνότητα των πιθανών θέσεων πόλων είναι ομοιόμορφη σε όλο το εσωτερικό του μοναδιαίου κύκλου. Για την επίτευξη αυτής της πιο ομοιόμορφης πυκνότητας, απαιτούνται διπλάσιοι πολλαπλασιασμοί. Σε κάποιες περιπτώσεις, αυτή η πολυπλοκότητα ίσως δικαιολογείται για να επιτύχει κανείς μεγαλύτερη ακρίβεια στη θέση ενός πόλου.

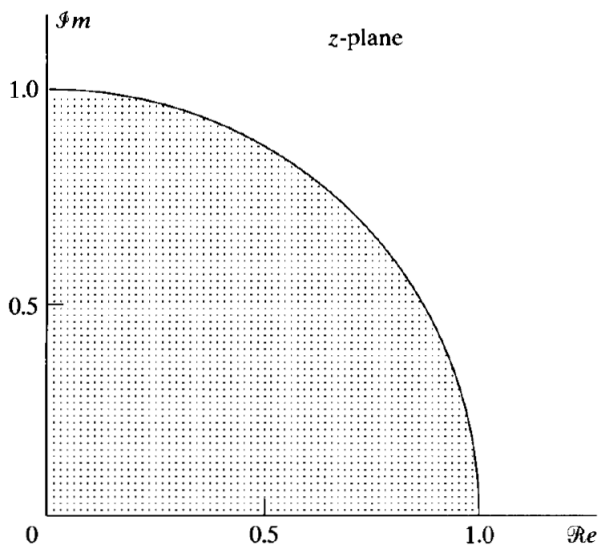
7.3 Διατήρηση Γραμμικής Φάσης

Γνωρίζετε ότι το μεγαλύτερο πλεονέκτημα ενός FIR συστήματος είναι η πιθανότητα να έχει γραμμική φάση. Όταν ένα FIR σύστημα γραμμικής φάσης υλοποιείται σε Direct Form δομή, τότε μπορεί ναδειχθεί ότι η φάση του δεν επηρεάζεται από τον κβαντισμό. Ασφαλώς, οι θέσεις των μηδενικών μπορεί να αλλάξουν, αλλά η συμμετρία τους στο μιγαδικό επίπεδο, ευνοεί τη διατήρηση της γραμμικής φάσης. Με άλλα λόγια, ό,τι σφάλμα κβαντισμού συμβεί σε ένα μηδενικό z_0 , το ίδιο σφάλμα θα συμβεί και στα συζυγή αμοιβαία τους! Παρ'όλο που το μέτρο της φασματικής απόκρισης μπορεί να μεταβάλλεται ανάλογα με το πλήθος των bits, η φάση θα παραμένει ακριβώς γραμμική!

Σχηματικά, μπορείτε να δείτε αυτό το φαινόμενο στο Σχήμα (34). Σε αυτό το σχήμα, το διάγραμμα πόλων-μηδενικών για 8 bits μας βάζει σε σκέψεις. Σε τέτοιες περιπτώσεις, όπου το σφάλμα κβαντισμού

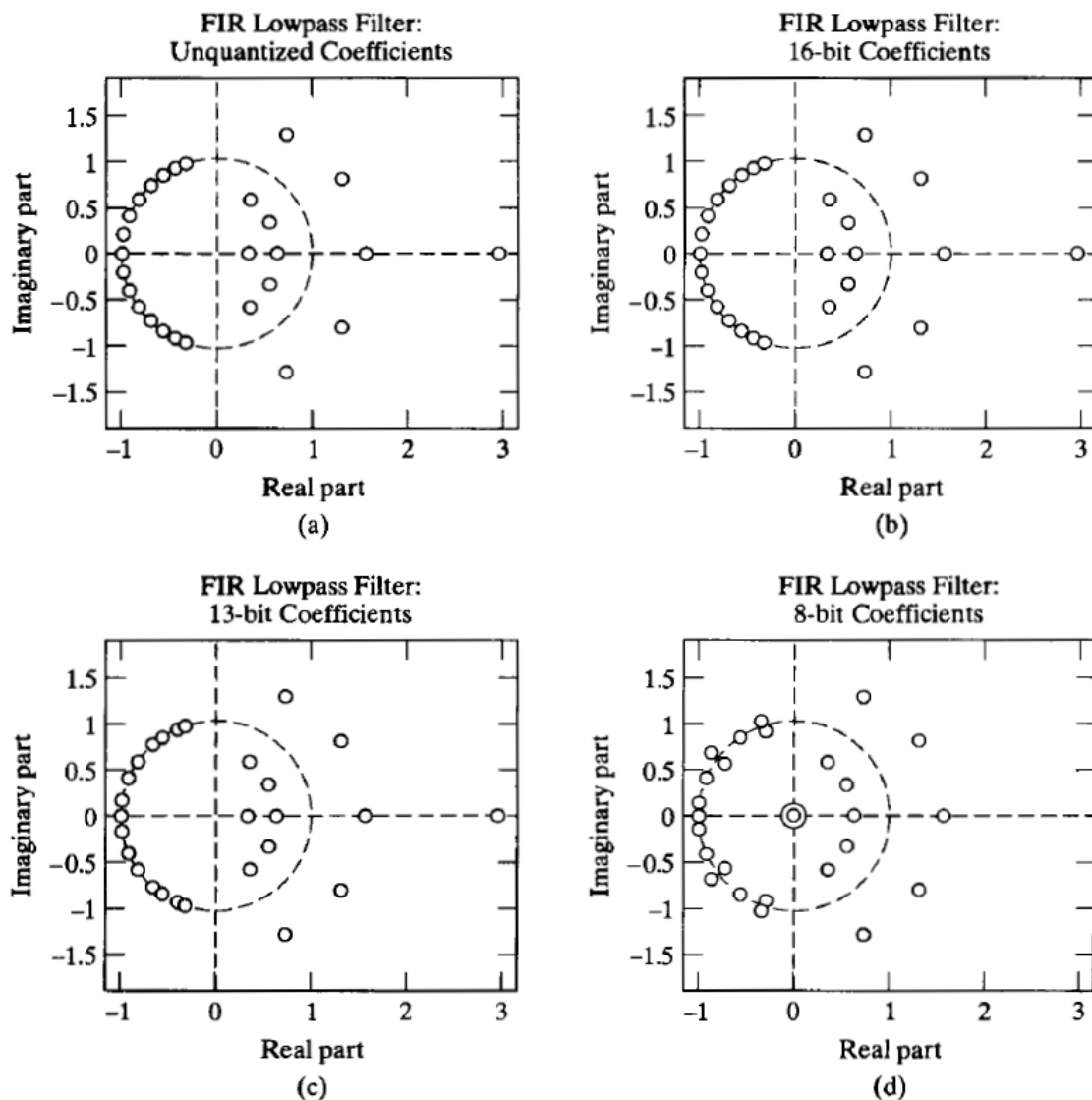


Σχήμα 32: Υλοποίηση Coupled Form γράφου με τέσσερα bits.



Σχήμα 33: Υλοποίηση Coupled Form γράφου με επτά bits.

είναι μεγάλο, ή για συστήματα μεγάλης τάξης που τα μηδενικά είναι πολύ κοντά μεταξύ τους, ίσως αξίζει τον κόπο να πραγματοποιήσουμε μικρότερα σύνολο μηδενικών ανεξάρτητα, με ένα γράφο σε σειρά. Για να διατηρήσουμε όμως τη συνολική φάση γραμμική, κάθε τέτοιο σύνολο πρέπει να έχει γραμμική φάση. Για παράδειγμα, αν χρησιμοποιήσουμε ζεύγη μηδενικών επάνω στο μοναδιαίο κύκλο, δηλ. παράγοντες της μορφής $(1+az^{-1}+z^{-2})$, το μηδενικό μπορεί να κινηθεί μόνο επάνω στο μοναδιαίο κύκλο όταν ο συντελεστής a χβαντιστεί. Αυτός ο διαχωρισμός αποτρέπει τα μηδενικά να κινηθούν μακριά από το μοναδιαίο κύκλο. Όμοια, αν ομαδοποιήσουμε σε ζεύγη τα πραγματικά μηδενικά (ένα μηδενικό με το αμοιβαίο του), τα μηδενικά αυτά θα παραμείνουν πραγματικά μετά τον χβαντισμό. Επίσης, μηδενικά στις θέσεις $z = \pm 1$ μπορούν να υλοποιηθούν ακριβώς με συστήματα πρώτης τάξης. Τέλος, αν θέλουμε να πραγματοποιήσουμε ένα ζεύγος συζυγών μηδενικών εντός του μοναδιαίου κύκλου με ένα σύστημα δεύτερης (αντί τέταρτης) τάξης, πρέπει να διασφαλίσουμε ότι για κάθε μιγαδικό μηδενικό εντός του μοναδιαίου κύκλου, υπάρχει ένα συζυγές αμοιβαίο μηδενικό εκτός του μοναδιαίου κύκλου. Αυτό μπορεί να γίνει αν εκφράσουμε τον παράγοντα τέταρτης τάξης που αναπαριστά τετράδες μηδενικών (όπως είχαμε δει όταν συζητούσαμε για τα συστήματα



Σχήμα 34: Επιρροή κβαντισμού συντελεστών ενός FIR συστήματος γραμμικής φάσης για διάφορα πλήθη bits.

γραμμικής φάσης), σε δυο παράγοντες δευτέρας τάξης. Για παράδειγμα, έστω ότι τα μηδενικά είναι στις θέσεις $z = re^{j\theta}$ και $z = r^{-1}e^{-j\theta}$. Τότε

$$1 + cz^{-1} + dz^{-2} + cz^{-3} + z^{-4} = (1 - 2r \cos(\theta)z^{-1} + r^2z^{-2}) \frac{1}{r^2}(r^2 - 2r \cos(\theta) + z^{-2}) \quad (77)$$

Το σύστημα αυτό χρησιμοποιεί τους ίδιους συντελεστές, $-2r \cos(\theta)$ και r^2 , για να πραγματοποιήσει τόσο τα μηδενικά εντός του μοναδιαίου κύκλου, όσο και τα αμοιβαία συζυγή τους εκτός του κύκλου. Έτσι, η γραμμική φάση διατηρείται υπό συνθήκες κβαντισμό.