



ΑΡΙΣΤΟΤΕΛΕΙΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΟΝΙΚΗΣ
ΤΜΗΜΑ ΜΑΘΗΜΑΤΙΚΩΝ
ΜΕΤΑΠΤΥΧΙΑΚΟ ΠΡΟΓΡΑΜΜΑ ΣΠΟΥΔΩΝ
“ΘΕΩΡΗΤΙΚΗ ΠΛΗΡΟΦΟΡΙΚΗ ΚΑΙ ΘΕΩΡΙΑ ΣΥΣΤΗΜΑΤΩΝ
ΚΑΙ ΕΛΕΓΧΟΥ”

ΑΡΙΘΜΗΤΙΚΗ ΕΠΙΛΥΣΗ ΤΩΝ ΕΞΙΣΩΣΕΩΝ
SYLVESTER - Η ΕΞΙΣΩΣΗ SYLVESTER
ΣΤΗ ΘΕΩΡΙΑ ΕΛΕΓΧΟΥ - ΕΦΑΡΜΟΓΕΣ ΜΕ MATLAB

ΜΕΤΑΠΤΥΧΙΑΚΗ ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

Δημήτριος Χ. Κατωτοκίδης

Επιβλέπουσα: Μαρία Γουσίδου-Κουτίτα
Αναπλ. Καθηγήτρια Α.Π.Θ

Θεσσαλονίκη, Ιούνιος 2007



ΑΡΙΣΤΟΤΕΛΕΙΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΟΝΙΚΗΣ
ΤΜΗΜΑ ΜΑΘΗΜΑΤΙΚΩΝ
ΜΕΤΑΠΤΥΧΙΑΚΟ ΠΡΟΓΡΑΜΜΑ ΣΠΟΥΔΩΝ
"ΘΕΩΡΗΤΙΚΗ ΠΛΗΡΟΦΟΡΙΚΗ ΚΑΙ ΘΕΩΡΙΑ ΣΥΣΤΗΜΑΤΩΝ
ΚΑΙ ΕΛΕΓΧΟΥ"

**ΑΡΙΘΜΗΤΙΚΗ ΕΠΙΛΥΣΗ ΤΩΝ ΕΞΙΣΩΣΕΩΝ
SYLVESTER - Η ΕΞΙΣΩΣΗ SYLVESTER
ΣΤΗ ΘΕΩΡΙΑ ΕΛΕΓΧΟΥ - ΕΦΑΡΜΟΓΕΣ ΜΕ MATLAB**

ΜΕΤΑΠΤΥΧΙΑΚΗ ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

Δημήτριος Χ. Κατωτοικίδης

Επιβλέπουσα: Μαρία Γουσιδου-Κουτίτα
Αναπλ. Καθηγήτρια Α.Π.Θ

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή

.....
Μ. Γουσιδου-Κουτίτα
Αναπλ. Καθηγήτρια Α.Π.Θ

.....
Α. Βαρδουλάκης
Καθηγητής Α.Π.Θ

.....
Ν. Καραμπετάκης
Επικ. Καθηγητής Α.Π.Θ

Θεσσαλονίκη, Ιούνιος 2007

.....
Δημήτριος Χ. Κατωτοκίδης
Πτυχιούχος Μαθηματικός Α.Π.Θ.

Copyright © Δημήτριος Χ. Κατωτοκίδης, 2007.
Με επιφύλαξη παντός δικαιώματος. All rights reserved.

Απαγορεύεται η αντιγραφή, αποθήκευση και διανομή της παρούσας εργασίας, εξ' ολοκλήρου ή τμήματος αυτής, για εμπορικό σκοπό. Επιτρέπεται η ανατύπωση, αποθήκευση και διανομή για σκοπό μη κερδοσκοπικό, εκπαιδευτικής ή ερευνητικής φύσης, υπό την προϋπόθεση να αναφέρεται η πηγή προέλευσης και να διατηρείται το παρόν μήνυμα. Ερωτήματα που αφορούν τη χρήση της εργασίας για κερδοσκοπικό σκοπό πρέπει να απευθύνονται προς τον συγγραφέα.

Οι απόψεις και τα συμπεράσματα που περιέχονται σε αυτό το έγγραφο εκφράζουν τον συγγραφέα και δεν πρέπει να ερμηνευτεί ότι εκφράζουν τις επίσημες θέσεις του Α.Π.Θ.

ΠΕΡΙΛΗΨΗ

Στην παρούσα εργασία γίνεται κυρίως λόγος για την αριθμητική επίλυση της εξίσωσης Sylvester $XA + BX = C$. Η συγκεκριμένη εξίσωση παρουσιάζει αρκετό ενδιαφέρον στον τομέα αριθμητικής ανάλυσης. Ασχολούμαστε επίσης με κάποια αποτελέσματα ανάλυσης διαταραχής για την εξίσωση Sylvester καθώς επίσης και με την αριθμητική επίλυση της γενικευμένης εξίσωσης Sylvester $AXB^T + CXD^T = E$, την αριθμητική επίλυση της διακριτής εξίσωσης Sylvester $BXA - X = C$, και της εξίσωσης Sylvester-παρατηρητή $XA - FX = GC$. Η εξίσωση Sylvester-παρατηρητή παίζει σημαντικό ρόλο στον τομέα της θεωρίας έλεγχου αφού εμφανίζεται σε μερικά προβλήματα αυτής.

Συγκεκριμένα στο πρώτο κεφάλαιο παρουσιάζουμε κάποιες εισαγωγικές έννοιες που θα πρέπει να είναι γνωστές στον αναγνώστη για την καλύτερη κατανόηση της εργασίας.

Στο δεύτερο κεφάλαιο γίνεται λόγος για την αριθμητική επίλυση της εξίσωσης Sylvester κυρίως με την Hessenberg-Schur μέθοδο και τον αλγόριθμο των Bartels-Stewart. Επίσης παρουσιάζουμε και έναν Hessenberg-Schur αλγόριθμο για την διακριτή εξίσωση Sylvester. Τέλος ένα μέρος του κεφαλαίου αυτού είναι αφιερωμένο στην αριθμητική επίλυση της γενικευμένης εξίσωσης Sylvester με έναν Hessenberg-Schur αλγόριθμο και έναν Bartels-Stewart αλγόριθμο που παρουσιάζουμε.

Στο τρίτο κεφάλαιο έχουμε την εξίσωση Sylvester-παρατηρητή που εμφανίζεται άμεσα στο πρόβλημα εκτίμησης κατάστασης ενός state space συστήματος και στο πρόβλημα ανατοποθέτησης ιδιοτιμών. Επίσης παρουσιάζουμε και την λεγόμενη περιορισμένη εξίσωση Sylvester-παρατηρητή που έχει έναν ιδιαίτερο ρόλο στην θεωρία έλεγχου.

Τέλος στο παράρτημα παρουσιάζουμε μερικά προγράμματα, M-files, που έγιναν από τον συγγραφέα της διπλωματικής αυτής στην Matlab, βάσει των αλγορίθμων που παρουσιάζονται σε αυτήν την εργασία.

ΛΕΞΕΙΣ ΚΛΕΙΔΙΑ

Εξίσωση Sylvester, εξίσωση Sylvester-παρατηρητής, γενικευμένη εξίσωση Sylvester, εκτίμηση κατάστασης μέσω εξίσωσης Sylvester, ανατοποθέτηση ιδιοτιμών μέσω εξίσωσης Sylvester.

ABSTRACT

In this current thesis we present numerical methods for the solution of the Sylvester equation $XA + BX = C$. This equation has a great interest in the department of numerical analysis. We discuss some perturbation theories for the Sylvester equation and we also present numerical methods for solving the generalized Sylvester equation $AXB^T + CXD^T = E$, the discrete Sylvester equation $BXA - X = C$, and the Sylvester-observer equation $XA - FX = GC$. The Sylvester-observer equation plays an important role in Control Theory, as this equation appears in some control theory problems.

In the first chapter we present some basic theory, so that the reader has a better understanding of this paper.

In chapter two we present some methods for obtaining the numerical solutions of the Sylvester equation, mainly the Hessenberg-Schur method and the Bartels-Stewart algorithm. We also present a Hessenberg-Schur method for the discrete Sylvester equation. A part of this chapter is dedicated in the numerical solution of the generalized Sylvester equation with a Hessenberg-Schur and a Bartels-Stewart algorithm.

In chapter 3 we have the Sylvester-observer equation, which directly appears in state estimation of a state space model and in the pole placement problem. We also present the constrained Sylvester-observer equation, which has a special role in control theory.

In the end in the appendix we present some programs, M-files, created by the author of this thesis. These M-files are based on the algorithms presented in this thesis, and they are created in Matlab.

KEY WORDS

Sylvester equation, Sylvester-observer equation, generalized Sylvester equation, state estimation via Sylvester equation, pole assignment via Sylvester equation.

ΠΡΟΛΟΓΟΣ

Ευχαριστώ θερμά την επιβλέπουσα αναπληρώτρια καθηγήτρια κα. Γουσίδου - Κουτίτα Μαρία, τόσο για τις υποδείξεις της, όσο και για τη συνολική επίβλεψη της εργασίας, καθώς επίσης και για το διαρκές ενδιαφέρον που έδειξε καθ' όλη την διάρκεια εκπόνησης της εργασίας. Τέλος θα ήθελα να την ευχαριστήσω για την καθοδήγηση της και την αμέριστη συμπαράστασή της.

Θερμές ευχαριστίες οφείλω στα μέλη της τριμελούς επιτροπής, τον καθηγητή κ. Βαρδουλάκη Αντώνιο-Ιωάννη και τον επίκουρο καθηγητή κ. Καραμπετάκη Νικόλαο για τον χρόνο που αφιέρωσαν στη μελέτη καθώς και την αξιολόγηση της εργασίας.

**Αριθμητική επίλυση των εξισώσεων Sylvester – Η εξίσωση
Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab**

ΠΙΝΑΚΑΣ ΠΕΡΙΕΧΟΜΕΝΩΝ

ΠΕΡΙΛΗΨΗ.....	5
ABSTRACT.....	6
ΠΡΟΛΟΓΟΣ.....	7
ΠΕΡΙΕΧΟΜΕΝΑ.....	9

Κεφάλαιο	Σελίδα
1. ΕΙΣΑΓΩΓΙΚΑ.....	13
1.1 Ιδιόμορφοι πίνακες	13
1.2 Νόρμες πινάκων και διανυσμάτων.....	13
1.3 Αποτελεσματικότητα αλγορίθμου, κατάσταση του προβλήματος, ευστάθεια αλγορίθμου και ακρίβεια της λύσης.....	15
1.4 Αριθμητική επίλυση του γραμμικού συστήματος $Ax=b$	16
1.5 Στοιχειώδεις ορθογώνιοι μετασχηματισμοί.....	19
1.6 Η QR παραγοντοποίηση.....	23
1.7 Hessenberg αναγωγή πινάκων και η πραγματική μορφή Schur.....	26
1.8 Αναγωγή σε Hessenberg-τριγωνική μορφή.....	31
1.9 Ελεγχιμότητα και παρατηρησιμότητα.....	34
2. ΑΡΙΘΜΗΤΙΚΗ ΕΠΙΛΥΣΗ ΚΑΙ ΚΑΤΑΣΤΑΣΗ ΤΗΣ ΕΞΙΣΩΣΗΣ SYLVESTER ΚΑΙ ΤΗΣ ΓΕΝΙΚΕΥΜΕΝΗΣ ΕΞΙΣΩΣΗΣ SYLVESTER.....	37
2.1 Εισαγωγή.....	37
2.2 Ύπαρξη και μοναδικότητα των λύσεων της εξίσωσης Sylvester.....	38
2.3 Ανάλυση διαταραχής της εξίσωσης Sylvester.....	39
2.4 Ο αριθμός κατάστασης της εξίσωσης Sylvester.....	43
2.4.1 Ο αριθμός κατάστασης της εξίσωσης Sylvester.....	43
2.4.2 Καθορισμός αρρωστημένης κατάστασης από τις ιδιοτιμές.....	44

**Αριθμητική επίλυση των εξίσωσεων Sylvester – Η εξίσωση
Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab**

2.5 Αριθμητικές μέθοδοι για την επίλυση της εξίσωσης Sylvester.....	47
2.5.1 Αριθμητική αστάθεια της διαγωνιοποίησης, της Jordan κανονικής μορφής, και των τεχνικών σχηματισμού συνοδύων πινάκων.....	49
2.5.2 Λύνοντας την εξίσωση Sylvester με την μέθοδο Schur (Bartels-Stewart).....	51
2.5.3 Η Hessenberg-Schur μέθοδος για την εξίσωση Sylvester.....	56
2.5.4 Η Hessenberg μέθοδος για την εξίσωση Sylvester $AX+XB=C$	62
2.5.5 Η Hessenberg-Schur μέθοδος για την διακριτή εξίσωση Sylvester.....	68
2.6 Σύγκριση διαφορετικών μεθόδων και συμπεράσματα.....	69
2.7 Επίλυση της γενικευμένης εξίσωσης Sylvester $AXB^T + CXD^T = E$	71
2.7.1 Η γενικευμένη εξίσωση Sylvester.....	71
2.7.2 Αριθμός κατάστασης της γενικευμένης εξίσωσης Sylvester.....	71
2.7.3 Προσπαθώντας να επιλύσουμε την γενικευμένη εξίσωση Sylvester.....	72
2.7.4 Η μέθοδος Bartels-Stewart για την επίλυση της γενικευμένης εξίσωσης Sylvester	73
2.7.5 Η Hessenberg-Schur μέθοδος για την επίλυση της γενικευμένης εξίσωσης Sylvester.....	76
2.7.6 Σύγκριση των δυο μεθόδων.....	78
2.8 Εφαρμογές.....	81

**3. Η ΕΞΙΣΩΣΗ SYLVESTER – ΠΑΡΑΤΗΡΗΤΗΣ ΣΑΝ ΕΚΤΙΜΗΤΗΣ
ΚΑΤΑΣΤΑΣΗΣ ΚΑΙ ΤΟ ΠΡΟΒΛΗΜΑ ΑΝΑΘΕΣΗΣ ΤΩΝ**

ΙΔΙΟΤΙΜΩΝ.....	98
3.1 Εισαγωγή.....	98
3.2 Εκτίμηση κατάστασης μέσω των ιδιοτιμών.....	99
3.3 Εκτίμηση κατάστασης μέσω της εξίσωσης Sylvester.....	101
3.4 Εκτίμηση κατάστασης ελαττωμένης τάξης.....	108
3.5 Non-singular λύσεις τις εξίσωσης Sylvester-παρατηρητή.....	114
3.6 Συνδυασμός ανάδρασης κατάστασης και σχεδιασμού παρατηρητή.....	117
3.7 Αριθμητική επίλυση της περιορισμένης Sylvester-παρατηρητή	

εξίσωσης	119
3.8 Η εξίσωση Sylvester-παρατηρητής και το πρόβλημα ανατοποθέτησης των ιδιοτιμών.....	125
ΠΑΡΑΡΤΗΜΑ.....	129
ΒΙΒΛΙΟΓΡΑΦΙΑ.....	171

**Αριθμητική επίλυση των εξισώσεων Sylvester – Η εξίσωση
Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab**

1. ΕΙΣΑΓΩΓΙΚΑ

Εδώ θα παρουσιάσουμε βασικές έννοιες και θα δώσουμε ορισμούς για πίνακες ή διανύσματα και γενικά οτιδήποτε αφορά την διπλωματική εργασία αυτή έτσι ώστε να γίνει καλύτερη η περαιτέρω κατανόηση της.

1.1 Ιδιόμορφοι πίνακες

Ένας πίνακας A θα λέμε ότι είναι μη-αντιστρέψιμος ή μη-ομαλός ή ιδιόμορφος (singular) εάν δεν υπάρχει ο αντίστροφος του δηλαδή δεν υπάρχει ο A^{-1} , ενώ θα λέμε ότι ένας πίνακας είναι αντιστρέψιμος ή ομαλός ή μη ιδιόμορφος (nonsingular) εάν υπάρχει ο A^{-1} .

Εάν ένας πίνακας A λοιπόν είναι singular, και άρα δεν υπάρχει ο αντίστροφος του, τότε θα ισχύει ότι η ορίζουσα του δεν είναι διάφορη του μηδενός δηλαδή ισχύει ότι $\det(A) = 0$ και άρα ο πίνακας χάνει θα λέγαμε τάξη. Το αν ένας πίνακας χάνει ή όχι τάξη μας ενδιαφέρει παρά πολύ ιδιαίτερα εάν αυτός ο πίνακας αποτελεί λύση γραμμικών εξισώσεων τύπου Sylvester αλλά και Lyapunov. Αυτό διότι οι λύσεις των εξισώσεων Sylvester, που θα μελετήσουμε εδώ πέρα, έχουν ιδιαίτερη σημασία στην θεωρία ελέγχου αλλά και σε πολλούς άλλους τομείς των μαθηματικών.

1.2 Νόρμες πινάκων και διανυσμάτων

Η νόρμα ενός πίνακα A συμβολίζεται ως $\|A\|$ και ικανοποιεί κάποιες σχέσεις

$$\|A\| \geq 0, \text{ όπου η ισότητα ισχύει αν-ν } A = 0$$

$$\|cA\| = |c| \|A\|, \text{ όπου } c \text{ σταθερά}$$

$$\|A + B\| \leq \|A\| + \|B\|$$

Αριθμητική επίλυση των εξισώσεων Sylvester – Η εξίσωση Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab

Όσον αφορά τώρα τα διανύσματα έχουμε τις εξής νόρμες για το διάνυσμα a που χρησιμοποιούνται πιο συχνά

$$\|a\|_1 = |a_1| + |a_2| + \dots + |a_n|$$

$$\|a\|_2 = \sqrt{(a^T a)} = \sqrt{(a_1^2 + a_2^2 + \dots + a_n^2)}$$

$$\|a\|_\infty = \max \{|a_i| : i = 1, 2, \dots, n\}$$

Όσον αφορά τώρα τους πίνακες έχουμε τις εξής νόρμες για τον πίνακα $A_{m \times n}$ που χρησιμοποιούνται πιο συχνά

Η frobenius νόρμα:

$$\|A\|_F = \sqrt{\sum_{i=1}^m \sum_{j=1}^n a_{ij}^2}$$

ενώ έχουμε και τις λεγόμενες p – νόρμες ($p = 1, 2, \dots, \infty$)

$$\|A\|_p = \max \left\{ \frac{\|Ax\|_p}{\|x\|_p} : x \neq 0 \right\}.$$

Αυτές οι νόρμες ικανοποιούν την ανισότητα $\|AB\| \leq \|A\| \|B\|$, ενώ μπορεί να δειχτεί ότι

$$\|A\|_1 = \max \left\{ \sum_{i=1}^m |a_{ij}| : j = 1, 2, \dots, n \right\}$$

$\|A\|_2 = \sqrt{\rho(A^T A)}$, όπου $\rho(A^T A)$ η φασματική ακτίνα του $A^T A$ δηλαδή η μεγαλύτερη του ιδιοτιμή.

$$\|A\|_{\infty} = \max \left\{ \sum_{j=1}^n |a_{ij}| : i = 1, 2, \dots, m \right\}$$

Τέλος να πούμε πως αν έχουμε πίνακες ορθογώνιους Q, Z ($Q^T Q = I, Z^T Z = I$) τότε η frobenius και η 2-νόρμα έχουν την ιδιότητα

$$\|QAZ\|_F = \|A\|_F, \quad \|QAZ\|_2 = \|A\|_2$$

για έναν πίνακα A .

1.3 Αποτελεσματικότητα αλγορίθμου, κατάσταση του προβλήματος, ευστάθεια αλγορίθμου και ακρίβεια της λύσης

Οι δυο πιο επιθυμητές ιδιότητες που θα πρέπει να έχει ένας αλγόριθμος είναι να είναι αριθμητικά ευσταθής, και να είναι αποδοτικός ή αλλιώς αποτελεσματικός.

Η αποδοτικότητα ή αλλιώς η αποτελεσματικότητα ενός αλγορίθμου μετριέται από το πόσο υπολογιστικό χρόνο χρειάζεται ο αλγόριθμος αυτός για να εφαρμοστεί.

Ένα θεωρητικό μέτρο για να μετρήσουμε την αποτελεσματικότητα ενός αλγορίθμου είναι τα λεγόμενα flops. Ένα flop (floating point operation) (αριθμητικές πράξεις με αριθμούς κινητού σημείου) είναι η εσωτερική παράσταση του αθροίσματος, διαφοράς, γινομένου ή πηλίκου : + , - , * , ή /.

Επίσης ένας αλγόριθμος θα ονομάζεται $O(n^p)$ αλγόριθμος εάν το κυρίαρχο μέρος (το μεγαλύτερο σε τάξη) του αριθμού που προκύπτει από την μέτρηση των flops του αλγορίθμου είναι πολλαπλάσιο του n^p .

Αριθμητική επίλυση των εξίσωσεων Sylvester – Η εξίσωση Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab

Τέλος όταν συγκρίνουμε την αποτελεσματικότητα δυο αλγορίθμων δεν πρέπει να δίνουμε μεγάλη έμφαση στην ακριβή μέτρηση των flops.

Τώρα θα θέλαμε να πούμε πως ένα πρόβλημα , όσον αφορά ένα δεδομένο σύνολο δεδομένων , είναι σε αρρωστημένη κατάσταση (ill-conditioned) ή σε κακή κατάσταση εάν ένα μικρό σχετικό σφάλμα στα δεδομένα μπορεί να προκαλέσει μεγάλο σχετικό σφάλμα στην υπολογισμένη λύση, άσχετα από την μέθοδο επίλυσης που χρησιμοποιούμε. Ειδιάλλως θα λέμε ότι το πρόβλημα μας είναι σε καλή κατάσταση.

Βέβαια το αν ένα πρόβλημα είναι σε καλή κατάσταση ή όχι αυτό εξαρτάται από το πρόβλημα, τη μέθοδο επίλυσης του και από τα δεδομένα κυρίως. Διότι το πρόβλημα μας για κάποια δεδομένα είναι σε καλή κατάσταση και για κάποια αλλά δεδομένα είναι σε αρρωστημένη κατάσταση.

Ένας αλγόριθμος θα λέγεται ότι είναι αριθμητικά ευσταθής εάν παράγει μια ακριβής λύση σε ένα κοντινό πρόβλημα, δηλαδή ο αλγόριθμος αυτός λύνει ακριβώς ένα πρόβλημα του οποίου τα δεδομένα είναι κοντά στα πραγματικά δεδομένα.

Έτσι όσον αφορά για την ακρίβεια της υπολογισμένης λύσης έχουμε να πούμε πως και η αριθμητική ευστάθεια του αλγορίθμου και η κατάσταση του προβλήματος παίζουν ρόλο. Εάν το πρόβλημα βρίσκεται σε αρρωστημένη κατάσταση, τότε δεν μπορούμε να εγγυηθούμε για την ακρίβεια των αποτελεσμάτων μας όσο ευσταθής και αν είναι ο αλγόριθμος που χρησιμοποιούμε.

Από την άλλη εάν ο αλγόριθμος είναι αριθμητικά ευσταθής και στην συνέχεια εφαρμοστεί σε πρόβλημα που βρίσκεται σε καλή κατάσταση τότε το αποτέλεσμα που θα πάρουμε θα είναι αριθμητικά ακριβές.

1.4 Αριθμητική επίλυση του γραμμικού συστήματος $Ax=b$

Κατά την διάρκεια αριθμητικής επίλυσης των εξισώσεων, αλλά και σε άλλα προβλήματα, θα θέλαμε να επιλύσουμε το σύστημα $Ax=b$, όπου A είναι ένας πίνακας $n \times n$, και b ένα διάνυσμα $n \times 1$.

Θα μπορούσε κανείς να πει πως θα μπορούσαμε εύκολα να βρούμε την λύση του παραπάνω γραμμικού συστήματος εάν παίρναμε τον αντίστροφο του πίνακα A , που πάντοτε θα υπάρχει αν θέλουμε το σύστημα μας να έχει μοναδική λύση, και βρίσκαμε την λύση x από $x=A^{-1}b$. Δυστυχώς όμως μια τέτοια προσέγγιση δεν θα ήταν αριθμητικά πρακτική ιδέα αφού γενικά χρησιμοποιούνται περισσότεροι υπολογισμοί για τον υπολογισμό του A^{-1} και έχουμε αποτελέσματα με λιγότερη ακρίβεια και περισσότερο υπολογιστικό χρόνο.

Ένα ωραίο παράδειγμα είναι η επίλυση του απλού συστήματος (εξίσωσης) $3x=27$. Η ακριβής λύση είναι $x=27/3=9$ και βρέθηκε με ένα flop (με μια διαίρεση). Αν όμως πάμε να λύσουμε το πρόβλημα βρίσκοντας τον αντίστροφο του πίνακα A τότε η λύση x θα δίνεται από $x=\frac{1}{3} \times 27=0.3333 \times 27=8.9991$ που είναι ένα λιγότερο ακριβές αποτέλεσμα. Επίσης χρειάζεται δυο flops, μια διαίρεση και ένα πολλαπλασιασμό.

Έτσι μια μέθοδος επίλυσης του συστήματος $Ax=b$ που προτείνεται είναι η χρήση απαλοιφής Gauss με μερική οδήγηση. Εδώ στην ουσία χρησιμοποιούμε την LU παραγοντοποίηση του πίνακα A που βασίζεται στην απαλοιφή Gauss με μερική οδήγηση. Έτσι έχοντας την παραγοντοποίηση του πίνακα A , δηλαδή $PA=LU$ τότε το σύστημα $Ax=b$ είναι ισοδύναμο με τα δυο τριγωνικά συστήματα

$$\begin{aligned} Ly &= Pb = b' \\ Ux &= y \end{aligned} .$$

Ο πίνακας U θα είναι ένας άνω τριγωνικός πίνακας, ο πίνακας L θα είναι ένας κάτω τριγωνικός πίνακας της μορφής

**Αριθμητική επίλυση των εξισώσεων Sylvester – Η εξίσωση
Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab**

$$L = \begin{pmatrix} 1 & 0 & 0 & \dots & \dots & 0 \\ -m_{21} & 1 & 0 & \dots & \dots & 0 \\ -m_{31} & -m_{32} & 1 & \dots & \dots & 0 \\ \vdots & \vdots & \ddots & \ddots & & \vdots \\ \vdots & \vdots & & \ddots & \ddots & 0 \\ -m_{n1} & -m_{n2} & -m_{n3} & \dots & -m_{n,n-1} & 1 \end{pmatrix}$$

όπου τα m_{ij} είναι τα αντίθετα των πολλαπλασιαστών (multipliers) (που προκύπτουν από την απαλοιφή Gauss με μερική οδήγηση) . Ο πίνακας P θα είναι ένας αντιμεταθετικός πίνακας (permutation matrix) δηλαδή θα έχει ακριβώς ένα μη μηδενικό στοιχείο σε κάθε γραμμή και στήλη και αυτό θα είναι 1 ενώ τα υπόλοιπα στοιχεία θα είναι μηδέν. Ο πίνακας P θα μας δείχνει αν είχαμε κάποια εναλλαγή γραμμών στον πίνακα A κατά την διάρκεια της απαλοιφής Gauss με μερική οδήγηση. Ο πίνακας P θα είναι πάντα ένας ορθογώνιος πίνακας.

Το σύστημα $Ly = Pb = b'$ είναι ένα κάτω τριγωνικό σύστημα και το y μπορεί να βρεθεί από τον τύπο

$$\text{For } i = 1, 2, \dots, n$$

$$y_i = \left(b_i - \sum_{j=1}^{i-1} l_{ij} y_j \right) / l_{ii}$$

που είναι γνωστός ως προς τα εμπρός αντικατάσταση, ενώ το σύστημα $Ux = y$ είναι ένα άνω τριγωνικό σύστημα και το x μπορεί να βρεθεί από τον τύπο

$$\text{For } i = n, n-1, \dots, 1$$

$$x_i = \left(y_i - \sum_{j=i+1}^n u_{ij} x_j \right) / u_{ii}$$

που είναι γνωστός ως πίσω αντικατάσταση.

Η παραπάνω μέθοδος θεωρείται αριθμητικά ευσταθής ,ενώ αν επιθυμούμε να λύσουμε το σύστημα $AX = B$ όπου εδώ B, X είναι πίνακες διάστασης $n \times m$ τότε αυτό μπορεί να λυθεί με την παραπάνω μέθοδο εάν γράψουμε τους πίνακες B, X ως

$$B = (b_1, b_2, \dots, b_m)$$

$$X = (x_1, x_2, \dots, x_m)$$

και αρχίσουμε και λύνουμε τα συστήματα

$$Ax_j = b_j \quad \text{για} \quad j = 1, 2, \dots, m$$

όπως γνωρίζουμε.

Τέλος να πούμε πως αυτήν την μέθοδο χρησιμοποιούμε και εμείς για να λύσουμε τα όποια συστήματα της μορφής $Ax = b$ συναντούμε στην πορεία. Αυτό γίνεται με την συνάρτηση **mygaussel** που έφτιαξε ο συγγραφέας αυτής της διπλωματικής και η συνάρτηση αυτή παρουσιάζεται εκτενώς στο παράρτημα. Η συνάρτηση αυτή χρησιμοποιείται, δηλαδή καλείται, και από διάφορες άλλες συναρτήσεις που θα δούμε παρακάτω, που έφτιαξε επίσης ο συγγραφέας αυτής της διπλωματικής.

1.5 Στοιχειώδεις ορθογώνιοι μετασχηματισμοί

Τα περισσότερα από τα υπολογιστικά προβλήματα στα οποία συμπεριλαμβάνονται πίνακες λύνονται με την αναγωγή των πινάκων αυτών σε τριγωνικές μορφές και γενικότερα θα λέγαμε σε κάποιες ειδικές μορφές. Η αναγωγή των πινάκων σε αυτές τις ειδικές μορφές γίνεται με την χρήση στοιχειωδών ορθογωνίων μετασχηματισμών.

Αυτό γιατί οι ορθογώνιοι μετασχηματισμοί μας εξασφαλίζουν την αριθμητική ευστάθεια της μεθόδου που χρησιμοποιούμε για την επίλυση του προβλήματος μας.

Έτσι αν x δοσμένο διάνυσμα και θέλουμε να μηδενίσουμε το στοιχείο του που βρίσκεται στην θέση j θα πρέπει να πάρουμε ως p, q τα:

$$p = \frac{x_i}{\sqrt{(x_i^2 + x_j^2)}} \quad q = \frac{x_j}{\sqrt{(x_i^2 + x_j^2)}}$$

τότε θα ισχύει $p^2 + q^2 = 1$ και

$$y_i = \sqrt{(x_i^2 + x_j^2)}$$
$$y_j = 0$$

Έτσι μπορούμε να κάνουμε μια σειρά από επίπεδες περιστροφές σε διάνυσμα ή σε πίνακα για να δημιουργήσουμε μηδενικά σε συγκεκριμένες θέσεις που εμείς θέλουμε.

Μια επίπεδη περιστροφή μπορεί να εφαρμοστεί σε ένα πίνακα A για να δημιουργήσει μετασχηματισμούς ομοιότητας RAR^T .

Τέλος να πούμε πως ένας πολλαπλασιασμός από αριστερά του πίνακα A με την παρακάτω επίπεδη περιστροφή

$$R = \begin{pmatrix} 1 & & & & & & \\ & \ddots & & & & & \\ & & 1 & & & & \\ & & & p & q & & \\ & & & -q & p & & \\ & & & & & 1 & \\ & & & & & & \ddots & \\ & & & & & & & 1 \end{pmatrix}$$

θα άλλαζε μονάχα τις i και $i+1$ γραμμές του πίνακα A , διάστασης εδώ $n \times n$, αφού θα ίσχυε πως

Αριθμητική επίλυση των εξισώσεων Sylvester – Η εξίσωση
Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab

for $j = 1, \dots, n$

$$a_{ij} \leftarrow p a_{ij} + q a_{i+1,j}$$

$$a_{i+1,j} \leftarrow p a_{i+1,j} - q a_{ij}$$

Ομοίως μπορούν να γίνουν και πολλαπλασιασμοί από δεξιά.

Τώρα μια στοιχειώδης αντανάκλαση (ή Householder) είναι ένας πίνακας της μορφής

$$U = I - \frac{uu^T}{h}$$

όπου u είναι ένα διάνυσμα και ισχύει

$$h = \frac{1}{2} \|u\|_2^2.$$

Ο πίνακας αυτός U θα είναι συμμετρικός και ορθογώνιος. Πολλαπλασιάζοντας τον πίνακα U με ένα διάνυσμα ή ένα πίνακα μπορούμε να εισαγάγουμε ταυτόχρονα πολλά μηδενικά στο διάνυσμα αυτό ή στον πίνακα .

Σχηματικά για ένα διάνυσμα a θα έχουμε ότι

$$a \xrightarrow{U} Ua = (-g, 0, \dots, 0)^T.$$

Έτσι για ένα διάνυσμα a μπορούμε να βρούμε μια στοιχειώδη αντανάκλαση έτσι ώστε

$$Ua = -ge_1 \quad e_1 = (1 \ 0 \ \dots \ 0)^T.$$

Θα ισχύει πως

$$u = a + ge_1$$

$$g = \text{sign}(a_1) \|a\|_2$$

όπου εδώ το g λαμβάνεται του ίδιου προσήμου με το a_1 έτσι ώστε να αποφευχθεί η αφαίρεση στην παρακάτω σχέση:

$$u = \begin{pmatrix} u_1 \\ u_2 \\ \vdots \\ u_n \end{pmatrix} = \begin{pmatrix} a_1 + g \\ a_2 \\ \vdots \\ a_n \end{pmatrix}$$

Επίσης μπορεί να δειχτεί εύκολα ότι ισχύει

$$h = g^2 + ga_1 = gu_1.$$

Αν τώρα θέλουμε να εφαρμόσουμε μια στοιχειώδη αντανάκλαση U σε ένα πίνακα A διάστασης $m \times n$ μπορούμε να χωρίσουμε τον πίνακα αυτόν σε διανύσματα δηλαδή

$$A = (a_1, a_2, \dots, a_n)$$

και να εφαρμόσουμε την στοιχειώδη αντανάκλαση ως εξής

$$UA = (Ua_1, Ua_2, \dots, Ua_n).$$

1.6 Η QR παραγοντοποίηση

Αν έχουμε έναν πίνακα A διάστασης $m \times n$ τότε υπάρχει ένας ορθογώνιος πίνακας Q διάστασης $m \times m$ και ένας άνω τριγωνικός πίνακας R διάστασης $m \times n$ έτσι ώστε να ισχύει

$$A = QR.$$

Αριθμητική επίλυση των εξίσωσεων Sylvester – Η εξίσωση Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab

Μια τέτοια παραγοντοποίηση λέγεται QR παραγοντοποίηση. Εάν $m \geq n$ και εάν ο πίνακας Q χωρίζεται ως $Q = (Q_1, Q_2)$ όπου Q_1 είναι ένας πίνακας όπου έχει τις n πρώτες στήλες του πίνακα Q και υπάρχει άνω τριγωνικός πίνακας R_1 διάστασης $n \times n$ τέτοιος ώστε

$$R = \begin{pmatrix} R_1 \\ 0 \end{pmatrix}$$

τότε θα έχουμε

$$A = Q_1 R_1.$$

Τέτοιου είδους παραγοντοποίηση του A λέγεται QR παραγοντοποίηση οικονομικού μεγέθους.

Υπάρχουν αρκετοί τρόποι για να υπολογίσει κανείς την QR παραγοντοποίηση ενός πίνακα. Μπορούμε να υπολογίσουμε την QR παραγοντοποίηση χρησιμοποιώντας στοιχειώδεις αντανάκλασεις ή επίπεδες περιστροφές.

Στην QR παραγοντοποίηση ενός πίνακα A χρησιμοποιώντας στοιχειώδεις αντανάκλασεις, θα έχουμε πίνακες στοιχειώδους αντανάκλασης U_1, U_2, \dots, U_n έτσι ώστε να ισχύει

$$U_n \dots U_2 U_1 A = R$$

και ο πίνακας R να γίνει άνω τριγωνικός. Έτσι αν $U_1 U_2 \dots U_n = Q$ τότε ο Q θα είναι ορθογώνιος αφού U_i είναι ορθογώνιος και εν τέλει θα έχουμε $Q^T A = R$ ή $A = QR$ όπου

$$R = \begin{pmatrix} R_1 \\ 0 \end{pmatrix}.$$

Δημήτριος Χ. Κατωτοκίδης

Οι πίνακες U_i δημιουργούνται έτσι ώστε να ισχύει $A^{(i)} = U_i A^{(i-1)}$ με $A^{(0)} = A$ και ο $A^{(i)}$ να έχει κάτω από την διαγώνιο μηδενικά στοιχεία στην i στήλη. Δηλαδή αν

$$A = (a_1, a_2, \dots, a_n)$$

και έχουμε μια $m \times m$ στοιχειώδη αντανάκλαση U_1 έτσι ώστε

$$U_1 a_1 = (r_{11}, 0, \dots, 0)^T$$

τότε θα έχουμε

$$A^{(1)} = U_1 A = \begin{pmatrix} r_{11} & r_{12} & \cdots & r_{1n} \\ \mathbf{0} & & & \\ \vdots & & A^{(1)'} & \\ \mathbf{0} & & & \end{pmatrix}$$

Μια δεύτερη αντανάκλαση U_2' διαλέγεται έτσι ώστε να μετασχηματίσει την πρώτη στήλη a_1' του $A^{(1)'$ ως εξής

$$U_2' a_1' = (r_{22} \quad 0 \quad \dots \quad 0)^T.$$

Έτσι εάν

$$U_2 = \begin{pmatrix} 1 & \mathbf{0} \\ \mathbf{0} & U_2' \end{pmatrix}$$

τότε

**Αριθμητική επίλυση των εξίσωσεων Sylvester – Η εξίσωση
Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab**

$$A^{(2)} = U_2 A^{(1)} = \begin{pmatrix} r_{11} & r_{12} & r_{13} & \cdots & r_{1n} \\ 0 & r_{22} & r_{23} & \cdots & r_{2n} \\ 0 & 0 & & & \\ \vdots & \vdots & & A^{(2)'} & \\ 0 & 0 & & & \end{pmatrix}.$$

Έτσι στο n βήμα θα έχουμε

$$A^{(n)} = U_n A^{(n-1)} = U_n U_{n-1} \dots U_1 A = \begin{pmatrix} r_{11} & r_{12} & \cdots & r_{1n} \\ & r_{22} & \cdots & r_{2n} \\ \mathbf{0} & & \ddots & \vdots \\ & & & r_{nn} \end{pmatrix} = \begin{pmatrix} R \\ \mathbf{0} \end{pmatrix}$$

όπου ο R είναι άνω τριγωνικός.

1.7 Hessenberg αναγωγή πινάκων και η πραγματική μορφή Schur

Ένας πίνακας $H = (h_{ij})$ θα λέγεται πως είναι άνω Hessenberg πίνακας εάν $h_{ij} = 0$ για $i > j + 1$. Δηλαδή ο άνω Hessenberg πίνακας $H = (h_{ij})$ θα έχει την μορφή :

$$H = \begin{pmatrix} h_{11} & h_{12} & \cdots & h_{1n} \\ h_{21} & h_{22} & \cdots & h_{2n} \\ 0 & h_{32} & \cdots & h_{3n} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & h_{n,n-1} & h_{nn} \end{pmatrix}.$$

Αν ισχύει πως $h_{ij} = 0$ για $j > i + 1$ τότε ο πίνακας $H = (h_{ij})$ θα λέγεται πως είναι κάτω Hessenberg πίνακας, ενώ ένας τετράγωνος πίνακας που είναι ταυτόχρονα και άνω και κάτω Hessenberg πίνακας θα λέγεται τριδιαγώνιος πίνακας.

Ένας άνω Hessenberg πίνακας H θα είναι μη αναγωγίσιμος εάν $h_{i,i-1} \neq 0$ για $i = 2, 3, \dots, n$.

Ομοίως ένας κάτω Hessenberg πίνακας H θα είναι μη αναγωγίσιμος εάν $h_{i,i+1} \neq 0$ για $i = 1, 2, \dots, n-1$.

Ένας πίνακας A διάστασης $n \times n$ θα μπορεί πάντοτε να αναχθεί σε άνω Hessenberg πίνακα H_u μέσω ορθογωνίων μετασχηματισμών. Δηλαδή δοθέντος ένα πίνακα A διάστασης $n \times n$, υπάρχει πίνακας ορθογώνιος P τέτοιο ώστε να ισχύει $PAP^T = H_u$.

Και πάλι μπορούμε να χρησιμοποιήσουμε επίπεδες περιστροφές ή στοιχειώδεις αντανακλάσεις για να πάρουμε την Hessenberg μορφή ενός πίνακα. Εδώ θα συζητήσουμε μονάχα την μέθοδο αναγωγής που χρησιμοποιεί στοιχειώδεις αντανακλάσεις.

Η μέθοδος αυτή είναι μια επέκταση της QR παραγοντοποίησης. Θα έχουμε τον πίνακα P ο οποίος θα είναι γινόμενο $n-2$ στοιχειωδών πινάκων αντανάκλασης P_1 έως P_{n-2} . Ο πίνακας P_1 κατασκευάζεται έτσι ώστε να δημιουργήσει μηδενικά στην πρώτη στήλη του πίνακα A κάτω από το στοιχείο στην θέση $(2,1)$. Ο πίνακας P_2 κατασκευάζεται έτσι ώστε να δημιουργήσει μηδενικά στην δεύτερη στήλη του πίνακα $P_1AP_1^T$ κάτω από το στοιχείο στην θέση $(3,2)$ και συνεχίζουμε έτσι μέχρι το τέλος. Στο τέλος του $n-2$ βήματος ο πίνακας $A^{(n-2)}$ θα είναι ένας πίνακας Hessenberg H_u ορθογώνια όμοιος με τον πίνακα A .

Θα ισχύει

$$\begin{aligned} H_u = A^{(n-2)} &= P_{n-2} A^{(n-3)} P_{n-2}^T = P_{n-2} \left(P_{n-3} A^{(n-4)} P_{n-3}^T \right) P_{n-2}^T \\ &= \dots = \left(P_{n-2} P_{n-3} \dots P_1 \right) A \left(P_1^T \dots P_{n-3}^T P_{n-2}^T \right) \end{aligned}$$

ενώ αν

**Αριθμητική επίλυση των εξίσωσεων Sylvester – Η εξίσωση
Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab**

$$P = P_{n-2}P_{n-3} \dots P_1$$

τότε θα έχουμε

$$H_u = PAP^T.$$

Επίσης επειδή ο κάθε P_i είναι ορθογώνιος άρα και ο πίνακας P θα είναι ορθογώνιος. Επίσης θα θέλαμε να πούμε πως ο πολλαπλασιασμός από δεξιά με τον πίνακα P_i^T δεν καταστρέφει τα μηδενικά που ήδη υπάρχουν στον πίνακα $P_i A^{(i-1)}$.

Η διαδικασία αυτή χρειάζεται περίπου $\frac{10}{3}n^3$ flops, ενώ αν θέλουμε να υπολογίσουμε και τον πίνακα P τότε θα χρειαστούν επιπλέον και αλλά $\frac{4}{3}n^3$ flops.

Τώρα ας είναι A ένας $n \times n$ πραγματικός πίνακας. Τότε θα υπάρχει ένας $n \times n$ ορθογώνιος πίνακας Q τέτοιος ώστε

$$Q^T A Q = R = \begin{pmatrix} R_{11} & R_{12} & \dots & R_{1k} \\ 0 & R_{22} & \dots & R_{2k} \\ \vdots & & \ddots & \vdots \\ 0 & \dots & 0 & R_{nn} \end{pmatrix}$$

όπου ο πίνακας R είναι η πραγματική μορφή Schur του πίνακα A . Κάθε R_{ii} είναι ή ένας πραγματικός αριθμός ή ένας 2×2 πραγματικός πίνακας. Οι πραγματικοί αριθμοί, οι οποίοι βρίσκονται στην διαγώνιο, αντιστοιχούν σε πραγματικές ιδιοτιμές και κάθε 2×2 πραγματικός πίνακας, ο οποίος βρίσκεται στην διαγώνιο, αντιστοιχεί σε ζευγάρι συζυγών ιδιοτιμών.

Έστω τώρα ένας $n \times n$ πραγματικός πίνακας A . Θα παρουσιάσουμε την βασική QR επαναληπτική μέθοδο.

Θέτουμε $A_0 \equiv A$.

for $k = 1, 2, \dots$ *do*

Βρες την QR παραγοντοποίηση του $A_{k-1} : A_{k-1} = Q_k R_k$

Υπολόγισε $A_k = R_k Q_k$

end

Έτσι θα έχουμε μια ακολουθία από πίνακες $\{A_k\}$ όπου ο κάθε πίνακας είναι ορθογώνια όμοιος με τον αρχικό πίνακα. Έτσι στην ουσία θα ισχύει πως

$$A_k = (Q_1 Q_2 \dots Q_k)^T A_0 (Q_1 Q_2 \dots Q_k).$$

Μπορεί ναδειχτεί ότι κάτω από ήπιους περιορισμούς αυτή η ακολουθία συγκλίνει στην πραγματική μορφή Schur. Έτσι μπορούμε να χρησιμοποιήσουμε την μέθοδο αυτή αν θέλουμε να υπολογίσουμε την πραγματική μορφή Schur ενός πίνακα.

Βέβαια η παραπάνω μέθοδος δεν θα είναι και τόσο πρακτική αν ο πίνακας που θέλουμε να αναγάγουμε σε Schur μορφή είναι γεμάτος και πυκνός, δηλαδή λίγα έως και κανένα από τα στοιχεία του δεν είναι μηδέν. Αυτό γιατί θα χρειάζονται περίπου $O(n^4)$ flops. Αυτό γιατί η QR παραγοντοποίηση ενός πίνακα χρειάζεται $O(n^3)$ flops και αν ο πίνακας μας έχει διάσταση $n \times n$ τότε n τέτοιες παραγοντοποιήσεις θα χρειαστούμε.

Ευτυχώς μπορούμε να αναγάγουμε τον πίνακα μας A σε μορφή Hessenberg και μετά να εφαρμόσουμε τον QR αλγόριθμο. Η αναγωγή του πίνακα μας σε Hessenberg μορφή θα γίνει μια φορά και θα χρειαστούν $\frac{10}{3}n^3$ flops. Από εκεί και ύστερα η QR παραγοντοποίηση ενός Hessenberg πίνακα χρειάζεται $O(n^2)$ flops. Άρα η συνολική διαδικασία θα μας κοστίσει τελικά $O(n^3)$ flops εάν γίνει με αυτόν τον τρόπο.

Τώρα το μόνο μειονέκτημα της παραπάνω τεχνικής είναι ότι η σύγκλιση της είναι σχετικά αργή, με την έννοια ότι χρειάζεται πολλές επαναλήψεις. Για την αποφυγή του

Αριθμητική επίλυση των εξίσωσεων Sylvester – Η εξίσωση Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab

μειονεκτήματος αυτού θα χρησιμοποιήσουμε κατάλληλη μετατόπιση. Η ιδέα αφορά την χρήση του αλγορίθμου QR στον μετατοπισμένο πίνακα $\hat{H} = H - \hat{\lambda}_i I$ όπου $\hat{\lambda}_i$ είναι μια προσεγγισμένη ιδιοτιμή του πίνακα H . Επειδή όμως οι μιγαδικές ιδιοτιμές ενός πραγματικού πίνακα υπάρχουν ως συζυγείς ζευγάρια στην ουσία εμείς τότε αντί για μια μετατόπιση χρησιμοποιούμε δυο μετατοπίσεις και έτσι θα ισχύει

```
for i = 1, 2, ... do
    διάλεξε δυο μετατοπίσεις  $k_1$  και  $k_2$ 
    βρες την QR παραγοντοποίηση  $H - k_1 I = Q_1 R_1$ 
    σχημάτισε  $H_1 = Q_1 R_1 + k_2 I$ 
    βρες την QR παραγοντοποίηση  $H_1 - k_2 I = Q_2 R_2$ 
    σχημάτισε  $H_2 = Q_2 R_2 + k_1 I$ 
end .
```

Η παραπάνω τεχνική όμως θα μας οδηγήσει σε μιγαδική αριθμητική κάτι το οποίο δεν θέλουμε. Για αυτό τον λόγο οδηγούμαστε στην λεγόμενη πεπλεγμένη τεχνική διπλής μετατόπισης της οποίας παρουσιάζουμε ένα βήμα

1. Υπολόγισε την πρώτη στήλη n_1 του πίνακα

$$N = (H - k_1 I)(H - k_2 I) = H^2 - (k_1 + k_2)H + k_1 k_2 I$$

2. Βρες έναν πίνακα στοιχειώδους αντανάκλασης P_0 τέτοιος ώστε $P_0 n_1$ να είναι πολλαπλάσιο του e_1 .

3. Βρες στοιχειώδεις πίνακες αντανάκλασης P_1 έως P_{n-2} :

$$H_2 = (P_0 P_1 \dots P_{n-2})^T H (P_0 P_1 \dots P_{n-2})$$

να είναι άνω Hessenberg πίνακας.

Έτσι μετά από δυο ή τρεις εφαρμογές της πεπλεγμένης τεχνικής διπλής μετατόπισης θα έχουμε την πραγματική μορφή Schur που ζητάμε.

Αν τώρα οι ιδιοτιμές του πίνακα μας διαφέρουν αρκετά μεταξύ τους η μια με την άλλη τότε αυτό θα έχει ως αντίκτυπο στην ακρίβεια του QR αλγορίθμου μας. Υπάρχει τρόπος αντιμετώπισης του προβλήματος αυτού αλλά δεν θα συζητηθεί εδώ πέρα.

Η αναγωγή ενός πίνακα σε πραγματική μορφή Schur απαιτεί περίπου $26n^3$ flops .

1.8 Αναγωγή σε Hessenberg-τριγωνική μορφή

Η αναγωγή σε Hessenberg-τριγωνική μορφή θεωρείται στην ουσία το πρώτο βήμα του QZ αλγορίθμου. Ο QZ αλγόριθμος δοθέντος δυο $n \times n$ πραγματικούς πίνακες A και B , υπάρχουν ορθογώνιοι πίνακες Q και Z έτσι ώστε ο πίνακας $Q^T A Z$ είναι άνω πραγματικός πίνακας Schur και ο πίνακας $Q^T B Z$ είναι άνω τριγωνικός.

Η αναγωγή σε Hessenberg-τριγωνική μορφή θα έχει ως εξής :

Ας είναι A και B δυο $n \times n$ πίνακες. Τότε

Βήμα 1. Βρες έναν ορθογώνιο πίνακα U τέτοιο ώστε ο

$$B \equiv U^T B$$

να είναι ένας άνω τριγωνικός πίνακας βρίσκοντας την QR παραγοντοποίηση του B .

Σχημάτισε τον πίνακα

$$A \equiv U^T A.$$

Αριθμητική επίλυση των εξίσωσεων Sylvester – Η εξίσωση Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab

Βήμα 2. Κάνε αναγωγή του πίνακα A σε Hessenberg μορφή καθώς κατά την διαδικασία αυτή διατήρησε την τριγωνική μορφή του πίνακα B .

Στην αρχή θα έχουμε

$$A \equiv U^T A = \begin{pmatrix} * & * & \dots & * \\ * & * & \dots & * \\ \vdots & \vdots & & \vdots \\ * & * & \dots & * \\ * & * & \dots & * \end{pmatrix}$$

$$B \equiv U^T B = \begin{pmatrix} * & * & \dots & \dots & * \\ 0 & * & \dots & \dots & * \\ 0 & 0 & \ddots & \dots & * \\ \vdots & & & \ddots & \vdots \\ 0 & 0 & 0 & \dots 0 & * \end{pmatrix}.$$

Πρώτα το στοιχείο του πίνακα A που βρίσκεται στην θέση $(n,1)$ μηδενίζεται αν εφαρμόσουμε σε αυτόν μια επίπεδη περιστροφή στο $(n-1, n)$ επιπέδου :

$$A \equiv Q_{n-1,n} A = \begin{pmatrix} * & * & \dots & * \\ * & * & \dots & * \\ \vdots & \vdots & & \vdots \\ * & * & \dots & * \\ 0 & * & \dots & * \end{pmatrix}.$$

Αυτός ο μετασχηματισμός, όταν εφαρμοστεί από τα αριστερά στον πίνακα B , θα μας δώσει ένα στοιχείο στην θέση $(n, n-1)$ που πριν ήταν μηδέν:

$$B \equiv Q_{n-1,n} B = \begin{pmatrix} * & * & \dots & \dots & * \\ 0 & * & \dots & \dots & * \\ 0 & 0 & \ddots & \dots & * \\ \vdots & & & \ddots & \vdots \\ 0 & 0 \dots & 0 & * & * \end{pmatrix}.$$

Τώρα εφαρμόζεται μια κατάλληλη επίπεδη περιστροφή $Z_{n-1,n}$ από τα δεξιά του B η οποία μηδενίζει το στοιχείο του πίνακα B που βρίσκεται στην θέση $(n, n-1)$. Αυτός ο μετασχηματισμός όταν εφαρμοστεί και στον πίνακα A από τα δεξιά ευτυχώς δεν καταστρέφει το μηδενικό που δημιουργήθηκε πριν. Έτσι θα έχουμε :

$$B \equiv B Z_{n-1,n} = \begin{pmatrix} * & * & \dots & \dots & * \\ 0 & * & \dots & \dots & * \\ 0 & 0 & \ddots & \dots & * \\ \vdots & & & \ddots & \vdots \\ 0 & 0 & 0 & \dots 0 & * \end{pmatrix}$$

$$A \equiv A Z_{n-1,n} = \begin{pmatrix} * & * & \dots & * \\ * & * & \dots & * \\ \vdots & \vdots & & \vdots \\ * & * & \dots & * \\ 0 & * & \dots & * \end{pmatrix}$$

Έτσι τα στοιχεία του A που βρίσκονται στην θέση $(n-1,1), (n-2,1), \dots, (3,1)$ γίνονται επιτυχώς μηδέν κάθε φορά που εφαρμόζουμε κατάλληλη επίπεδη περιστροφή στα αριστερά του πίνακα A , η οποία ακολουθείται από μια άλλη επίπεδη περιστροφή που εφαρμόζεται στα δεξιά του πίνακα B για να μηδενίσει τα ανεπιθύμητα μη μηδενικά στοιχεία που εμφανίζονται στον πίνακα αυτόν σε ανεπιθύμητες θέσεις. Έτσι στο τέλος του πρώτου βήματος, ο πίνακας A θα είναι πίνακας Hessenberg μονάχα στην πρώτη στήλη καθώς ο πίνακας B παραμένει άνω τριγωνικός.

Αριθμητική επίλυση των εξισώσεων Sylvester – Η εξίσωση Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab

$$A = \begin{pmatrix} * & * & \dots & * \\ * & * & \dots & * \\ 0 & * & \dots & * \\ \vdots & \vdots & \ddots & \vdots \\ 0 & * & \dots & * \end{pmatrix}, \quad B = \begin{pmatrix} * & * & \dots & \dots & * \\ 0 & * & \dots & \dots & * \\ 0 & 0 & \ddots & \dots & * \\ \vdots & & \ddots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 0 & * \end{pmatrix}.$$

Τώρα σαν επόμενο βήμα προσπαθούμε να δημιουργήσουμε μηδενικά στην δεύτερη στήλη του πίνακα A , στις κατάλληλες θέσεις καθώς θα προσπαθούμε να διατηρήσουμε τον πίνακα B σε άνω τριγωνικό με ανάλογο τρόπο όπως προηγουμένως.

Η διαδικασία τερματίζεται όταν έχουμε μετασχηματίσει πλήρως τον πίνακα A σε πίνακα Hessenberg ενώ συγχρόνως ο B έχει διατηρηθεί σε άνω τριγωνική μορφή.

Τέλος να πούμε ότι ο συγγραφέας της διπλωματικής αυτής εργασίας, έχει φτιάξει σε συνάρτηση την αναγωγή σε Hessenberg-τριγωνική μορφή. Την ονόμασε **myqzpart1** και είναι αναρτημένη εκτενώς στο παράρτημα , σαν βοηθητική συνάρτηση, ενώ εκεί πέρα γίνεται και λόγος για την εγγύτητα των αποτελεσμάτων που μας δίνει.

1.9 Ελεγχιμότητα και παρατηρησιμότητα

Υπάρχουν πολλοί ορισμοί και πολλά αποτελέσματα για την ελεγχιμότητα και την παρατηρησιμότητα ενός συστήματος και για το πότε ένα σύστημα είναι ελέγξιμο ή παρατηρήσιμο. Εδώ βεβαίως θα παρουσιάσουμε χωρίς απόδειξη μονάχα όσα αποτελέσματα μας αφορούν και μας ενδιαφέρουν.

Ας θεωρήσουμε το σύστημα

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t) \\ y(t) &= Cx(t) + Du(t) \end{aligned} \quad (1.1)$$

όπου A, B, C, D είναι πίνακες διάστασης $n \times n, n \times m, r \times n, r \times m$ αντίστοιχα με $m \leq n$

Ορισμός 1.9.1

Το σύστημα

$$\begin{aligned}\dot{x}(t) &= Ax(t) + Bu(t) \\ y(t) &= Cx(t) + Du(t)\end{aligned}$$

θα λέγεται ότι είναι ελέγξιμο (controllable) εάν για κάθε αρχική κατάσταση $x(0)$, το σύστημα μπορεί να οδηγηθεί σε οποιαδήποτε τελική κατάσταση $x_1 = x(t_1)$ σε κάποιο πεπερασμένο χρόνο t_1 , επιλέγοντας κατάλληλα την μεταβλητή εισόδου $u(t)$ για $0 \leq t \leq t_1$.

Στο παρακάτω θεώρημα παρουσιάζουμε κριτήρια που μας δείχνουν πότε ένα σύστημα της παραπάνω μορφής είναι ελέγξιμο. Ορισμένα από αυτά έχουν ως εξής :

Θεώρημα 1.9.1 Οι επόμενες προτάσεις είναι ισοδύναμες:

- (i) το ζεύγος (A, B) είναι ελέγξιμο .
- (ii) Ο $n \times nm$ πίνακας $C_M = (B, AB, A^2B, \dots, A^{n-1}B)$ έχει πλήρη τάξη n .
- (iii) $\text{rank}(A - \lambda I, B) = n$ για κάθε ιδιοτιμή λ του πίνακα A .

Ο πίνακας $C_M = (B, AB, A^2B, \dots, A^{n-1}B)$ λέγεται πίνακας ελεγχιμότητας .

Η έννοια της ελεγχιμότητας είναι δυϊκή αυτής της παρατηρησιμότητας.

Ορισμός 1.9.2

Το σύστημα της μορφής (1.1) θα λέγεται ότι είναι παρατηρήσιμο (observable) εάν υπάρχει $t_1 > 0$ τέτοιο ώστε κάθε αρχική κατάσταση $x(0)$ μπορεί να προσδιοριστεί μοναδικά από την είσοδο $u(t)$ και την έξοδο $y(t)$ για όλα τα t , όπου $0 \leq t \leq t_1$.

**Αριθμητική επίλυση των εξισώσεων Sylvester – Η εξίσωση
Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab**

Θεώρημα 1.9.2 Οι επόμενες προτάσεις είναι ισοδύναμες:

- (i) το ζεύγος (A, C) είναι παρατηρήσιμο.
- (ii) Ο πίνακας παρατηρησιμότητας

$$O_M = \begin{pmatrix} C \\ CA \\ CA^2 \\ \vdots \\ CA^{n-1} \end{pmatrix} \quad \text{έχει πλήρη τάξη } n.$$

- (iii) $\text{rank} \begin{pmatrix} \lambda I - A \\ C \end{pmatrix} = n$ για κάθε ιδιοτιμή λ του πίνακα A .

2. ΑΡΙΘΜΗΤΙΚΗ ΕΠΙΛΥΣΗ ΚΑΙ ΚΑΤΑΣΤΑΣΗ ΤΗΣ ΕΞΙΣΩΣΗΣ SYLVESTER ΚΑΙ ΤΗΣ ΓΕΝΙΚΕΥΜΕΝΗΣ ΕΞΙΣΩΣΗΣ SYLVESTER

2.1 Εισαγωγή

Οι λύσεις της εξίσωσης Sylvester έχουν μεγάλη χρησιμότητα, αφού χρησιμοποιούνται για την κατασκευή παρατηρητών κατάστασης, καθώς επίσης και για την επίλυση του προβλήματος ανατοποθέτησης των ιδιοτιμών. Οι λύσεις της εξίσωσης Sylvester μπορούν να χρησιμοποιηθούν στην διαδικασία μπλοκ διαγωνοποίησης πινάκων μέσω μετασχηματισμών ομοιότητας. Όταν ένας πίνακας μετασχηματιστεί σε μπλοκ διαγώνια μορφή χρησιμοποιώντας ένα μετασχηματισμό ομοιότητας τότε η μπλοκ διαγώνια αυτή μορφή μπορεί άνετα να χρησιμοποιηθεί για να υπολογίσουμε τον πίνακα e^{At} .

Οι λύσεις της εξίσωσης Sylvester έχουν χρησιμότητα και σε άλλους τομείς των μαθηματικών. Για παράδειγμα η αριθμητική επίλυση προβλημάτων ελλειπτικών τροχιών μπορεί να αναχθεί σε πρόβλημα επίλυσης της εξίσωσης Sylvester.

Τελείως εγκυκλοπαιδικά θα θέλαμε να πούμε πως η συνεχούς χρόνου εξίσωση Lyapunov

$$XA + A^T X = C$$

είναι μια ειδικότερη μορφή της εξίσωσης συνεχούς χρόνου Sylvester που είναι η

$$XA + BX = C. \quad (2.1)$$

Ομοίως για τα συστήματα διακριτού χρόνου έχουμε την εξίσωση Sylvester που είναι η

$$BXA - X = C$$

Αριθμητική επίλυση των εξισώσεων Sylvester – Η εξίσωση Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab

όπου οι πίνακες A, B, C , δεν χρειάζεται απαραίτητα να είναι οι πίνακες του συστήματος.

Επίσης έχουμε και την γενικευμένη εξίσωση Sylvester συνεχούς χρόνου που είναι η εξής

$$AXB^T + CXD^T = E$$

Σε αυτό το κεφάλαιο μελετάμε πρώτα τις βασικές θεωρίες της ύπαρξης και μοναδικότητας λύσης της εξίσωσης Sylvester. Στην συνέχεια ασχολούμαστε με τις θεωρίες διαταραχών και τέλος με τις αριθμητικές μεθόδους επίλυσης των εξισώσεων Sylvester.

Γενικά όπου χρειάζεται να αναφερθούμε στην εξίσωση συνεχούς χρόνου Sylvester θα αναφερόμαστε σε αυτήν απλά ως εξίσωση Sylvester.

2.2 Ύπαρξη και μοναδικότητα των λύσεων της εξίσωσης Sylvester

Αν $A_{n \times n}, B_{m \times m}, C_{m \times n}$, τότε έχουμε το παρακάτω θεώρημα ύπαρξης μοναδικότητας λύσης της εξίσωσης Sylvester.

Θεώρημα 2.2.1. Αν $\lambda_1, \dots, \lambda_n$ είναι οι ιδιοτιμές του πίνακα A και μ_1, \dots, μ_m οι ιδιοτιμές του πίνακα B τότε η εξίσωση Sylvester έχει μοναδική λύση X αν-ν $\lambda_i + \mu_j \neq 0$ για κάθε $i = 1, \dots, n$ και για κάθε $j = 1, \dots, m$ αν-ν δηλαδή οι πίνακες A και $-B$ δεν έχουν ίδιες ιδιοτιμές.

Απόδειξη. Η $XA + BX = C$ είναι ισοδύναμη με το γραμμικό σύστημα $Px = c$, όπου

$$P = (I_n \otimes B) + (A^T \otimes I_m),$$

$$x = \text{vec}(X) = (x_{11}, \dots, x_{m1}, x_{12}, x_{22}, \dots, x_{m2}, \dots, x_{1n}, x_{2n}, \dots, x_{mn})^T,$$

$$c = \text{vec}(C) = (c_{11}, \dots, c_{m1}, c_{12}, c_{22}, \dots, c_{m2}, \dots, c_{1n}, c_{2n}, \dots, c_{mn})^T.$$

Έτσι η εξίσωση Sylvester έχει μοναδική λύση εάν-ν ο πίνακας P είναι αντιστρέψιμος (non-singular). Επίσης το γινόμενο $W \otimes Z$ ονομάζεται το γινόμενο του Kronecker των W και Z . Ενδεικτικά να πούμε ότι εάν $W_{p \times p}$ και $Z_{r \times r}$ τότε το γινόμενο του Kronecker ορίζεται ως

$$W \otimes Z = \begin{pmatrix} w_{11}Z & w_{12}Z & \cdots & w_{1p}Z \\ w_{21}Z & w_{22}Z & \cdots & w_{2p}Z \\ \vdots & \vdots & \ddots & \vdots \\ w_{p1}Z & w_{p2}Z & \cdots & w_{pp}Z \end{pmatrix} \in \mathbb{R}^{pr \times pr}.$$

Έτσι η εξίσωση $Px = c$, έχει μοναδική λύση εάν-ν ο πίνακας P είναι non-singular.

Οι ιδιοτιμές του πίνακα P είναι οι nm αριθμοί $\lambda_i + \mu_j$ όπου $i = 1, \dots, n$ και $j = 1, \dots, m$

Συνεπώς ο πίνακας P είναι non-singular αν-ν $\lambda_i + \mu_j \neq 0$ για κάθε $i = 1, \dots, n$ και $j = 1, \dots, m$. ▲

2.3 Ανάλυση διαταραχής της εξίσωσης Sylvester

Ένας πολύ σημαντικός παράγοντας όσων αφορά τα αποτελέσματα που παίρνουμε από έναν αριθμητικό αλγόριθμο είναι η ακρίβεια των λύσεων μας. Για τον λόγο αυτό στην ενότητα αυτή θα μελετήσουμε την ανάλυση διαταραχής της εξίσωσης Sylvester.

Παρακάτω ορίζουμε την συνάρτηση sep η οποία εμφανίζεται στα αποτελέσματα της ανάλυσης διαταραχής της εξίσωσης Sylvester καθώς και στον αριθμό κατάστασης της εξίσωσης αυτής που θα μελετηθεί αργότερα.

**Αριθμητική επίλυση των εξίσωσεων Sylvester – Η εξίσωση
Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab**

Ορισμός 2.3.1 Ο διαχωρισμός δυο πινάκων A και B μπορεί να γραφτεί ως $sep(A, B)$ και ορίζεται ως :

$$sep(A, B) = \min_{X \neq 0} \frac{\|AX - XB\|_F}{\|X\|_F}.$$

Έστω λοιπόν οι πίνακες A, B, C και \hat{X} και οι διαταραχές τους $\Delta A, \Delta B, \Delta C$ και ΔX αντίστοιχα. Αν \hat{X} είναι η λύση του διαταραγμένου προβλήματος τότε θα έχουμε ότι το \hat{X} ικανοποιεί την εξής σχέση

$$\hat{X}(A + \Delta A) + (B + \Delta B)\hat{X} = C + \Delta C. \quad (2.2)$$

Από (Higham 1996) έχουμε το παρακάτω θεώρημα.

Θεώρημα 2.3.1

Αν $\|\Delta A\|_F \leq \varepsilon\alpha$, $\|\Delta B\|_F \leq \varepsilon\beta$, $\|\Delta C\|_F \leq \varepsilon\gamma$ και α, β, γ κάποια προκαθορισμένα όρια τότε θα έχουμε

$$\varepsilon = \max \left\{ \frac{\|\Delta A\|_F}{\alpha}, \frac{\|\Delta B\|_F}{\beta}, \frac{\|\Delta C\|_F}{\gamma} \right\}$$

Αν τώρα η εξίσωση Sylvester έχει μοναδική λύση X για $C \neq 0$ τότε θα ισχύει ότι

$$\frac{\|\Delta X\|_F}{\|X\|_F} = \frac{\|\hat{X} - X\|_F}{\|X\|_F} \leq \sqrt{3}\varepsilon\delta, \quad (2.3)$$

όπου

$$\delta = \|P^{-1}\|_2 \frac{(\alpha + \beta)\|X\|_F + \gamma}{\|X\|_F}. \quad \blacktriangle$$

Όμως ισχύει

$$\|P^{-1}\|_2 = \frac{1}{\text{sep}(B, -A)}$$

και άρα η παραπάνω ανισότητα (2.3) γράφεται:

$$\frac{\|\Delta X\|_F}{\|X\|_F} \leq \sqrt{3}\varepsilon \frac{1}{\text{sep}(B, -A)} \frac{(\alpha + \beta)\|X\|_F + \gamma}{\|X\|_F}. \quad (2.4)$$

όπου εδώ το δεξί μέλος της ανισότητας (2.4) εξαρτάται από την λύση X .

Το δεξί μέλος της ανισότητας (2.4) (Golub et al.1979) δε θα εξαρτάται από την λύση X αν-ν όλες οι διαταραχές των πινάκων A, B, C είναι τάξης ίσης με την ακρίβεια της μηχανής, έστω μ αυτή. Δηλαδή να ισχύει $\|\Delta A\|_F \leq \mu\|A\|_F$, $\|\Delta B\|_F \leq \mu\|B\|_F$, και $\|\Delta C\|_F \leq \mu\|C\|_F$, καθώς και να ισχύει ότι

$$\mu \frac{\|A\|_F + \|B\|_F}{\text{sep}(B, -A)} \leq \frac{1}{2}.$$

Τότε θα έχουμε ότι

$$\frac{\|\Delta X\|_F}{\|X\|_F} = \frac{\|\hat{X} - X\|_F}{\|X\|_F} \leq 4\mu \frac{\|A\|_F + \|B\|_F}{\text{sep}(B, -A)}.$$

Παράδειγμα 2.3.1

Εάν

$$A = \begin{pmatrix} 1 & 1 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{pmatrix} \quad B = \begin{pmatrix} -0.9888 & 0 & 0 \\ 0 & -0.9777 & 0 \\ 0 & 0 & -0.9666 \end{pmatrix}$$

**Αριθμητική επίλυση των εξίσωσεων Sylvester – Η εξίσωση
Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab**

$$C = \begin{pmatrix} 0.0112 & 1.0112 & 2.0112 \\ 0.0223 & 1.0223 & 2.0223 \\ 0.0334 & 1.0334 & 2.0334 \end{pmatrix}$$

τότε η λύση της εξίσωσης Sylvester $BX + XA = C$ είναι

$$X = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix}$$

Εάν τώρα διαταράξουμε μονάχα το στοιχείο του πίνακα A που βρίσκεται στην θέση (1,1) και από 1 γίνει 0.999999, τότε η λύση με ακρίβεια 4 δεκαδικών ψηφίων είναι η

$$\hat{X} = \begin{pmatrix} 1.0001 & 0.9920 & 1.7039 \\ 1.0000 & 0.9980 & 1.0882 \\ 1.0000 & 0.9991 & 1.0259 \end{pmatrix}.$$

Έτσι μια σχετικά πολύ μικρή αλλαγή στον πίνακα A και συνεπώς στην νόρμα του

$$\frac{\|\Delta A\|_F}{\|A\|_F} = 4.0825 \times 10^{-7}$$

μας αποφέρει στην λύση της εξίσωσης Sylvester σχετικό σφάλμα

$$\frac{\|\hat{X} - X\|_F}{\|X\|_F} = 0.2366.$$

Βλέπουμε δηλαδή ότι σχετικές διαταραχές στα δεδομένα μας έχουν μεγεθυνθεί κατά 10^6 στην λύση μας.

Αυτό γίνεται γιατί ισχύει ότι ο αριθμός $\text{sep}(B, -A) = 1.4207 \times 10^{-6}$ είναι μικρός.

Αν επίσης $\alpha = \|A\|_F$, $\beta = \|B\|_F$, $\gamma = \|C\|_F$ τότε $\varepsilon = 4.0825 \times 10^{-7}$ και το δεξί μέλος της ανισότητας (2.4) θα είναι ίσο με 2.7133. Άρα η ανισότητα θα ικανοποιείται.

2.4 Ο αριθμός κατάστασης της εξίσωσης Sylvester

Εδώ θα μελετήσουμε τον αριθμό κατάστασης της εξίσωσης Sylvester έτσι ώστε να είμαστε σε θέση να γνωρίζουμε εάν τα προβλήματα μας είναι σε καλή κατάσταση ή όχι.

2.4.1 Ο αριθμός κατάστασης της εξίσωσης Sylvester

Εάν έχουμε σχετικές μ -διαταραχές στους πίνακες A, B, C , της εξίσωσης Sylvester (2.1) και ισχύει ότι

$$\delta = \mu \frac{\|A\|_F + \|B\|_F}{\text{sep}(B, -A)} < 1$$

τότε υπάρχει θεώρημα που μας λέει ότι θα ισχύει

$$\frac{\|\hat{X} - X\|_F}{\|X\|_F} \leq \frac{2\mu}{1 - \delta} \frac{\|A\|_F + \|B\|_F}{\text{sep}(B, -A)}. \quad (2.5)$$

Η ποσότητα

$$\frac{\|A\|_F + \|B\|_F}{\text{sep}(B, -A)}$$

θεωρείται ως ο αριθμός κατάστασης της εξίσωσης Sylvester.

Στην ανισότητα (2.5) μπορούμε να δούμε ότι αν ο διαχωρισμός μεταξύ των πινάκων είναι πολύ μικρός τότε η αλλαγή στον X είναι μεγάλη όπως πράγματι είδαμε στο παράδειγμα 2.3.1.

Αριθμητική επίλυση των εξίσωσεων Sylvester – Η εξίσωση Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab

Επίσης υπάρχει ένα θεώρημα το οποίο μας δίνει και αυτό τον αριθμό κατάστασης της εξίσωσης Sylvester αλλά εκμεταλλεύεται την μορφή Kronecker του P . Ο αριθμός αυτός είναι ο

$$\Psi = \frac{\|P^{-1}[\beta(X^T \otimes I_m), a(I_n \otimes X), -\gamma I_{mn}]\|_2}{\|X\|_F},$$

όπου ισχύει ότι

$$\frac{\|\Delta X\|_F}{\|X\|_F} \leq \sqrt{3}\Psi \varepsilon. \quad (2.6)$$

Τα $\varepsilon, \alpha, \beta, \gamma$ ορίζονται όπως και στο Θεώρημα 2.3.1. Το όριο διαταραχής που μας δίνεται στην ανισότητα (2.3) καμία φορά μπορεί να υπερεκτιμήσει τα αποτελέσματα των διαταραχών όταν διαταράσσονται μονάχα οι πίνακες A και B . Για αυτό και συστήνεται το όριο διαταραχής που μας δίνεται στην ανισότητα (2.6).

2.4.2 Καθορισμός αρρωστημένης κατάστασης από τις ιδιοτιμές

Καθώς ο υπολογισμός του $\|P^{-1}\| = 1/\text{sep}(B, -A)$ δεν είναι καθόλου εύκολος και επειδή $\text{sep}(B, -A) > 0$ αν-ν ο A και ο B δεν έχουν κοινές ιδιοτιμές, θα αναρωτιόταν κανείς εάν θα μπορούσαμε να προσδιορίσουμε μέσω των ιδιοτιμών των πινάκων A και B αν ο P^{-1} και άρα η εξίσωση Sylvester είναι σε αρρωστημένη κατάσταση.

Σαν απάντηση έχουμε το παρακάτω θεώρημα.

Θεώρημα 2.4.1 Η εξίσωση Sylvester $XA + BX = C$ είναι σε αρρωστημένη κατάσταση αν και οι δυο πίνακες A και B είναι σε αρρωστημένη κατάσταση όσον αφορά την αντιστροφή τους.

Παράδειγμα 2.4.1 Έστω οι πίνακες

$$A = \begin{pmatrix} 1 & 1 & 1 \\ 0 & 0 & 1 \\ 1 & 1 & 0.001 \end{pmatrix}, \quad B = \begin{pmatrix} 1 & 2 & 3 \\ 0 & 0.0001 & 1 \\ 0 & 0 & 0.0001 \end{pmatrix}$$

και

$$C = \begin{pmatrix} 8 & 8 & 8.001 \\ 3.0001 & 3.0001 & 3.0011 \\ 2.0001 & 2.0001 & 2.0011 \end{pmatrix}.$$

Η ακριβής λύση είναι η

$$X = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix}.$$

Αν τώρα αλλάξουμε το στοιχείο a_{31} από 1 σε 0.99999 και τα υπόλοιπα στοιχεία τα αφήσουμε απείραχτα τότε η λύση του διαταραγμένου προβλήματος είναι η

$$\hat{X} = \begin{pmatrix} 908.1970 & -905.2944 & -906.2015 \\ -452.6722 & 454.2208 & 454.6745 \\ 1.0476 & 0.9524 & 0.9524 \end{pmatrix}.$$

Όπως βλέπουμε υπάρχει μεγάλη διαφορά μεταξύ της λύσης X από την \hat{X} . Συγκεκριμένα έχουμε ότι

$$\frac{\|\hat{X} - X\|_2}{\|X\|_2} = 585.4190$$

ενώ η σχετική διαταραχή στα δεδομένα μας είναι :

Αριθμητική επίλυση των εξισώσεων Sylvester – Η εξίσωση Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab

$$\frac{\|\hat{A} - A\|_2}{\|A\|_2} = 4.5964 \times 10^{-6}.$$

Βέβαια κάτι τέτοιο συμβαίνει επειδή και οι δυο πίνακες βρίσκονται σε αρρωστημένη κατάσταση αφού ισχύει ότι

$$\text{Cond}(A) = 6.1918 \times 10^{16}, \quad \text{Cond}(B) = 8.5602 \times 10^8.$$

Το αντίστροφο του θεωρήματος γενικά δεν ισχύει.

Παράδειγμα 2.4.2 Ας πάρουμε τους πίνακες

$$A = \begin{pmatrix} 1 & 1 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{pmatrix}, \quad B = \begin{pmatrix} -0.9888 & 0 & 0 \\ 0 & -0.9777 & 0 \\ 0 & 0 & -0.9666 \end{pmatrix},$$

$$\text{και } C = \begin{pmatrix} 0.0112 & 1.0112 & 2.0112 \\ 0.0223 & 1.0223 & 2.0223 \\ 0.0334 & 1.0334 & 2.0334 \end{pmatrix}.$$

Τότε η υπολογισμένη λύση της εξίσωσης $BX + XA = C$ (υπολογισμένη από την συνάρτηση της Matlab **lyap**) είναι

$$\hat{X} = \begin{pmatrix} 1.0001 & 0.9920 & 1.7039 \\ 1.0000 & 0.9980 & 1.0882 \\ 1.0000 & 0.9991 & 1.0259 \end{pmatrix}.$$

Το σχετικό σφάλμα από την πραγματική λύση θα είναι

$$\frac{\|\hat{X} - X\|_F}{\|X\|_F} = 0.2366.$$

Βλέπουμε δηλαδή ότι η εξίσωση Sylvester έχοντας τα δεδομένα αυτού του παραδείγματος βρίσκεται σε αρρωστημένη κατάσταση. Θα πρέπει να τονίσουμε όμως ότι ούτε ο πίνακας A ούτε ο πίνακας B είναι σε αρρωστημένη κατάσταση διότι

$$\text{Cond}(A) = 4.0489, \quad \text{Cond}(B) = 1.0230.$$

Σε λίγες γραμμές θα θέλαμε να πούμε πως για κάθε nonsingular πίνακα, έστω A , το γινόμενο $\|A\| \|A^{-1}\|$ θα λέγετε πως είναι ο αριθμός κατάστασης του πίνακα A δηλαδή $\text{Cond}(A) = \|A\| \|A^{-1}\|$. Τον αριθμό αυτό μπορούμε να τον υπολογίσουμε με οποιαδήποτε νόρμα. Αναλόγως θα έχουμε και διαφορετικά αποτελέσματα ανάλογα με το ποια νόρμα θα χρησιμοποιήσουμε. Εδώ έχει υπολογιστεί με την χρήση της 2-νόρμας.

Τώρα να προσθέσουμε ότι επειδή ισχύει $\|I\| = \|A A^{-1}\| \leq \|A\| \|A^{-1}\|$ θα έχουμε ότι θα ισχύει $\text{Cond}(A) = \|A\| \|A^{-1}\| \geq 1$. Έτσι αν ο αριθμός αυτός βρίσκεται κοντά στην μονάδα τότε ο πίνακας A θα βρίσκεται σε καλή κατάσταση. Εάν δεν βρίσκεται κοντά στην μονάδα τότε ο πίνακας θα βρίσκεται σε αρρωστημένη κατάσταση (γενικά όσον αφορά την επίλυση γραμμικών συστημάτων), συγκεκριμένα εδώ όσον αφορά την αντιστροφή του.

Τέλος να πούμε ότι εάν ο πίνακας βρίσκεται σε αρρωστημένη κατάσταση με μια νόρμα, τότε θα βρίσκεται σε αρρωστημένη κατάσταση και με τις άλλες νόρμες, ενώ παρομοίως μπορούμε να πούμε και όταν ο πίνακας βρίσκεται σε καλή κατάσταση. Αυτό γιατί οι αριθμοί κατάστασης των διαφόρων νορμών συσχετίζονται μεταξύ τους.

2.5 Αριθμητικές μέθοδοι για την επίλυση της εξίσωσης Sylvester

Ένας φανερός τρόπος για να λύσουμε την εξίσωση Sylvester $XA + BX = C$ είναι να χρησιμοποιήσουμε απαλοιφή Gauss με μερική οδήγηση στο σύστημα $Px = c$. Αλλά, εκτός εάν μπορούμε να εκμεταλλευτούμε την ειδική μορφή του πίνακα P , η απαλοιφή

Αριθμητική επίλυση των εξίσωσεων Sylvester – Η εξίσωση Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab

Gauss με μερική οδήγηση για την εξίσωση Sylvester θα είναι υπολογιστικά απαγορευτική, καθώς χρειαζόμαστε $O(n^3 m^3)$ flops υπολογισμού και $O(n^2 m^2)$ flops αποθήκευσης. Ένας τρόπος για να εκμεταλλευτούμε την ειδική δομή του πίνακα P είναι να μετασχηματίσουμε τους πίνακες A και B σε πιο απλές μορφές χρησιμοποιώντας μετασχηματισμούς ομοιότητας.

Έτσι αν U και V είναι nonsingular πίνακες τέτοιοι ώστε

$$U^{-1}AU = \hat{A}, \quad V^{-1}BV = \hat{B} \quad \text{και} \quad V^{-1}CU = \hat{C},$$

τότε η εξίσωση $XA + BX = C$ μετασχηματίζεται στην

$$Y\hat{A} + \hat{B}Y = \hat{C} \tag{2.7}$$

όπου $Y = V^{-1}XU$.

Εάν \hat{A} και \hat{B} είναι σε απλές μορφές, τότε η εξίσωση $Y\hat{A} + \hat{B}Y = \hat{C}$ μπορεί εύκολα να λυθεί, και την λύση X μπορούμε να την πάρουμε από τον πίνακα Y . Έτσι θα έχουμε τα παρακάτω βήματα:

Βήμα 1. Μετασχημάτισε τους πίνακες A και B σε απλές μορφές όπως διαγωνιοποίηση, Jordan και συνοδεύων μορφή, Hessenberg, Schur, πραγματική μορφή Schur κ.τ.λ χρησιμοποιώντας μετασχηματισμούς ομοιότητας:

$$\hat{A} = U^{-1}AU, \quad \hat{B} = V^{-1}BV.$$

Βήμα 2. Υπολόγισε τον πίνακα: $\hat{C} = V^{-1}CU$.

Βήμα 3. Λύσε ως προς Y την μετασχηματισμένη εξίσωση: $Y\hat{A} + \hat{B}Y = \hat{C}$.

Βήμα 4. Πάρε το X από το Y λύνοντας το σύστημα : $XU = VY$.

Στο βήμα 2 και στο βήμα 4 έχουμε να λύσουμε στην ουσία γραμμικά συστήματα εξισώσεων, όπου βασικά στο βήμα 2 έχουμε το σύστημα $V\hat{C} = CU$ και θέλουμε να λύσουμε ως προς \hat{C} . Θα θέλαμε να τονίσουμε ότι δεν προτιμούμε να λύσουμε τα γραμμικά αυτά συστήματα, εάν οι μετασχηματισμοί ομοιότητας δεν είναι σε καλή κατάσταση, χρησιμοποιώντας έστω και απαλοιφή Gauss με μερική οδήγηση επειδή αναμένουμε πως η υπολογισμένη λύση \hat{X} που θα πάρουμε θα περιέχει σχετικά σφάλματα τάξης

$$\mu[\text{cond}(U) + \text{cond}(V)].$$

2.5.1 Αριθμητική αστάθεια της διαγωνιοποίησης, της Jordan κανονικής μορφής, και των τεχνικών σχηματισμού συνοδεύων πινάκων

Είναι αλήθεια πως οι δομές των πινάκων Jordan και των συνοδεύων πινάκων μπορούν να χρησιμοποιηθούν ωραία για να λύσουμε την ελαττωμένη εξίσωση Sylvester (2.7). Παρόλα αυτά τις μορφές Jordan και των συνοδεύων πινάκων δεν μπορούμε να τις δημιουργήσουμε χωρίς να χάσουμε σε αριθμητική ευστάθεια. Έτσι οι μετασχηματισμένοι πίνακες θα είναι σε μερικές περιπτώσεις σε αρρωστημένη κατάσταση, και αυτό με την σειρά του θα επηρεάσει τον υπολογισμό των πινάκων $\hat{A}, \hat{B}, \hat{C}$ και X , αφού για να προκύψουν αυτοί οι πίνακες θα χρειαστεί να υπολογιστούν οι αντίστροφοι πίνακες των μετασχηματισμών. Έτσι αυτού του είδους οι μέθοδοι θα μας δώσουν ανακριβή αποτελέσματα και γενικά θα πρέπει να αποφεύγονται για αριθμητικούς υπολογισμούς. Παρακάτω παρουσιάζουμε ένα παράδειγμα για το πώς η τεχνική της διαγωνιοποίησης μας δίνει μη-ακριβή αποτελέσματα.

Παράδειγμα 2.5.1 Ας είναι

$$A = \begin{pmatrix} -2.60 & 12.45 \\ -0.50 & 2.39 \end{pmatrix}, \quad B = \begin{pmatrix} 4.60 & 21.54 \\ -0.60 & -2.59 \end{pmatrix}$$

και

Αριθμητική επίλυση των εξισώσεων Sylvester – Η εξίσωση Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab

$$C = \begin{pmatrix} 13.85 & 28.80 \\ 5.89 & 20.84 \end{pmatrix}.$$

Εάν οι πίνακες A και B ελαττωθούν σε διαγώνια μορφή $\hat{A} = U^{-1}AU$ και $\hat{B} = V^{-1}BV$ αντίστοιχα, τότε μπορούμε να πάρουμε τον πίνακα Y από

$$y_{ij} = \frac{\hat{c}_{ij}}{\hat{a}_{ii} + \hat{b}_{jj}} \quad i = 1, 2 \quad j = 1, 2$$

Χρησιμοποιώντας αριθμητική κινητού σημείου με $\mu = 2^{-31} \approx 4.66 \times 10^{-10}$ έχουμε μέχρι 10 δεκαδικά ψηφία

$$Y_{diag} = \begin{pmatrix} 0.9995926065 & 0.9975505071 \\ 0.9999319419 & 0.9995908819 \end{pmatrix}$$

ενώ η ακριβής λύση είναι

$$Y = \begin{pmatrix} 1.000000000 & 1.000000000 \\ 1.000000000 & 1.000000000 \end{pmatrix}.$$

Για αυτό το παράδειγμα έχουμε ότι

$$\text{cond}_2(U) = 1.6773 \times 10^6 \quad \text{και} \quad \text{cond}_2(V) = 4.9025 \times 10^6$$

και έτσι τα σφάλματα που εμφανίζονται στη λύση είναι τάξης

$$\mu [\text{cond}_2(U) + \text{cond}_2(V)] = 0.003.$$

Από την άλλη έχουμε

$$\frac{1}{\text{sep}(A, -B)} = 787.97.$$

Άρα βλέπουμε ότι η εξίσωση Sylvester είναι σχετικά σε καλή κατάσταση. Άρα η μέθοδος που χρησιμοποιούμε είναι αριθμητικά ασταθής.

Έτσι από τα παραπάνω βλέπουμε ότι οι απλές μορφές με τις οποίες θα ασχοληθούμε είναι οι μορφές Hessenberg και οι (πραγματικές) μορφές Schur, αφού ξέρουμε ότι οι πίνακες μετασχηματισμού U, V σε αυτές τις περιπτώσεις μπορούν να διαλεχτούν έτσι ώστε να είναι ορθογώνιοι, οι οποίοι ορθογώνιοι μετασχηματισμοί είναι σε τέλεια καλή κατάσταση.

2.5.2 Λύνοντας την εξίσωση Sylvester με την μέθοδο Schur (Bartels-Stewart)

Η πρώτη μέθοδος για να λύσουμε την εξίσωση Sylvester με ορθογώνιους μετασχηματισμούς είναι η μέθοδος Schur που χρησιμοποιείται στον αλγόριθμο των Bartels-Stewart. Συγκεκριμένα έχουμε ότι ο πίνακας A μετασχηματίζεται σε

$$\tilde{A} = U^T A U = \begin{pmatrix} \tilde{A}_{11} & 0 & \cdots & 0 \\ \tilde{A}_{21} & \tilde{A}_{22} & \ddots & \vdots \\ \vdots & \vdots & \ddots & 0 \\ \tilde{A}_{p1} & \tilde{A}_{p2} & \cdots & \tilde{A}_{pp} \end{pmatrix}$$

όπου ο κάθε πίνακας \tilde{A}_{ii} είναι τάξης ένα ή δυο.

Ομοίως και ο B μετασχηματίζεται σε άνω μορφή Schur

$$\tilde{B} = V^T B V = \begin{pmatrix} \tilde{B}_{11} & \tilde{B}_{12} & \cdots & \tilde{B}_{1q} \\ 0 & \tilde{B}_{22} & \cdots & \tilde{B}_{2q} \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & \tilde{B}_{qq} \end{pmatrix}$$

όπου τα \tilde{B}_{ii} είναι το πολύ τάξης δυο.

**Αριθμητική επίλυση των εξισώσεων Sylvester – Η εξίσωση
Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab**

Εάν

$$\tilde{C} = U^T C V = \begin{pmatrix} \tilde{C}_{11} & \cdots & \tilde{C}_{1q} \\ \vdots & & \vdots \\ \tilde{C}_{p1} & \cdots & \tilde{C}_{pq} \end{pmatrix} \quad \text{και} \quad \tilde{X} = U^T X V = \begin{pmatrix} \tilde{X}_{11} & \cdots & \tilde{X}_{1q} \\ \vdots & & \vdots \\ \tilde{X}_{p1} & \cdots & \tilde{X}_{pq} \end{pmatrix}$$

τότε η εξίσωση $AX + XB = C$ γίνεται $\tilde{A}\tilde{X} + \tilde{X}\tilde{B} = \tilde{C}$.

Εάν τα κομμάτια των $\tilde{A}, \tilde{B}, \tilde{C}, \tilde{X}$ είναι σύμμορφα (conformal), τότε ισχύει

$$\tilde{A}_{kk} \tilde{X}_{kl} + \tilde{X}_{kl} \tilde{B}_{ll} = \tilde{C}_{kl} - \sum_{j=1}^{k-1} \tilde{A}_{kj} \tilde{X}_{jl} - \sum_{i=1}^{l-1} \tilde{X}_{ki} \tilde{B}_{il} \quad \begin{cases} k=1, 2, \dots, p \\ l=1, 2, \dots, q \end{cases}$$

Αυτές οι εξισώσεις μπορούν να λυθούν για $\tilde{X}_{11}, \tilde{X}_{21}, \dots, \tilde{X}_{p1}, \tilde{X}_{12}, \tilde{X}_{22}, \dots, \tilde{X}_{pq}$. Την λύση την παίρνουμε από τον μετασχηματισμό $X = U \tilde{X} V^T$. Η μετατροπή του πίνακα A σε κάτω πραγματικό πίνακα Schur μπορεί να γίνει εάν μετατρέψουμε τον ανάστροφο του A σε άνω πίνακα Schur και μετά αναστρέψουμε το αποτέλεσμα.

Παρακάτω ακολουθεί ο αλγόριθμος των Bartels-Stewart για την επίλυση της εξίσωσης Sylvester $AX + XB = C$. Στον αλγόριθμο αυτό αναγάγουμε τους πίνακες A και B σε κάτω πραγματική μορφή Schur και σε άνω πραγματική μορφή Schur αντίστοιχα.

Αλγόριθμος 2.5.1

Ελάττωσε τον πίνακα A σε κάτω πραγματική μορφή Schur

$$\tilde{A} = U^T A U .$$

Ελάττωσε τον πίνακα B σε άνω πραγματική μορφή Schur

$$\tilde{B} = V^T B V .$$

Δημήτριος Χ. Κατωτοκίδης

Υπολόγισε τον

$$\tilde{C} = U^T C V .$$

For $l = 1, 2, \dots, q$

$$\tilde{C}_{kl} \leftarrow \tilde{C}_{kl} - \sum_{i=1}^{l-1} \tilde{X}_{ki} \tilde{B}_{il} \quad k = 1, 2, \dots, p$$

For $k = 1, 2, \dots, p$

$$\tilde{C}_{kl} \leftarrow \tilde{C}_{kl} - \sum_{j=1}^{k-1} \tilde{A}_{kj} \tilde{X}_{jl}$$

$$\text{Λύσε την } \tilde{A}_{kk} \tilde{X}_{kl} + \tilde{X}_{kl} \tilde{B}_{ll} = \tilde{C}_{kl} \text{ για } \tilde{X}_{kl}$$

Υπολόγισε την $X = U \tilde{X} V^T$.

Υποθέτοντας ότι η μετατροπή των πινάκων A και B σε Schur μορφή χρειάζεται $26n^3$ και $26m^3$ flops αντίστοιχα τότε ο αλγόριθμος μας απαιτεί

$$26(n^3 + m^3) + 5(nm^2 + mn^2)$$

flops. Ο απαιτούμενος αριθμός flops για αποθήκευση είναι $4n^2 + 4m^2 + 2nm$, υποθέτοντας ότι τα δεδομένα μας γράφτηκαν πάνω σε προηγούμενα δεδομένα.

Σε αυτόν τον αλγόριθμο έχουμε το εξής:

Έχουμε τους πίνακες $\tilde{A} = U^T A U$ και $\tilde{B} = V^T B V$ και ο πίνακας Y στην εξίσωση $\tilde{A}Y + Y\tilde{B} = \tilde{C}$, όπου $\tilde{C} = U^T C V$ υπολογίζεται στην ουσία χρησιμοποιώντας την απαλοιφή Gauss και εμπρός αντικατάσταση για να λύσουμε το σύστημα

$$Ty = d .$$

Όπου εδώ ο $T = I_m \otimes R + S^T \otimes I_n$ είναι ένας κάτω περίπου τριγωνικός πίνακας και

**Αριθμητική επίλυση των εξίσωσεων Sylvester – Η εξίσωση
Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab**

$$y = \text{vec}(Y), \quad d = \text{vec}(D).$$

Εδώ σε αυτόν τον αλγόριθμο θα έχουμε στην λύση μας \bar{X} σφάλματα τάξης

$$\mu \frac{\|A\|_F + \|B\|_F}{\text{sep}(A, -B)}$$

αφού ισχύει ότι

$$\frac{\|\bar{X} - X\|_F}{\|X\|_F} \leq (8\varepsilon + 4\varepsilon^2) \frac{\|A\|_F + \|B\|_F}{\text{sep}(A, -B)}$$

όπου μ η ακρίβεια της μηχανής και ε μικρό πολλαπλάσιο αυτής.

Για το σχετικό υπόλοιπο θα έχουμε ότι ισχύει

$$\frac{\|A\bar{X} + \bar{X}B - C\|_F}{\|X\|_F} \leq (8\varepsilon + 4\varepsilon^2) (\|A\|_F + \|B\|_F)$$

που είναι ανεξάρτητο του $\text{sep}(A, -B)$

Παράδειγμα 2.5.2. Έστω ότι έχουμε να λύσουμε την εξίσωση Sylvester $AX + XB = C$

όπου

$$A = \begin{pmatrix} -5 & 2 & 7 & 1 & -8 \\ 3 & -6 & 2 & -2 & -4 \\ -1 & 4 & -2 & 6 & 3 \\ -4 & -7 & -1 & 5 & -2 \\ -2 & 3 & 1 & 0 & 9 \end{pmatrix} \quad B = \begin{pmatrix} -13 & -14 \\ 4 & 5 \end{pmatrix}$$

$$C = \begin{pmatrix} 4 & 7 \\ -52 & -27 \\ -2 & -9 \\ 15 & 36 \\ -3 & -46 \end{pmatrix}.$$

Η ακριβής λύση είναι η

$$X = \begin{pmatrix} -1 & 2 \\ 3 & 1 \\ -2 & -3 \\ -6 & -4 \\ 1 & -2 \end{pmatrix}.$$

Η λύση που πήραμε από τον αλγόριθμο 2.5.1 με αριθμητική κινητού σημείου $\mu = 2^{-56} \approx 1.39 \times 10^{-17}$ είναι ακριβής σε 13 ψηφία και έχει σχετικό σφάλμα

$$\frac{\|\bar{X} - X\|_F}{\|X\|_F} = 0.661 \times 10^{-14}.$$

Θα έχουμε επίσης $\text{sep}(A, -B) = 0.0568$ και έτσι το σφάλμα θα είναι τάξης

$$\mu \frac{\|A\|_F + \|B\|_F}{\text{sep}(A, -B)} = 1.02 \times 10^{-14}.$$

Το σχετικό υπόλοιπο θα είναι

$$\frac{\|A\bar{X} + \bar{X}B - C\|_F}{\|X\|_F} = 2.28 \times 10^{-15}$$

που είναι τάξης

$$\mu (\|A\|_F + \|B\|_F) = 5.80 \times 10^{-16}.$$

Αριθμητική επίλυση των εξίσωσεων Sylvester – Η εξίσωση Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab

Στο παράρτημα παραθέτουμε ένα πρόγραμμα ,την συνάρτηση **SchrblockSchr**, που έφτιαξε ο συγγραφέας αυτής της διπλωματικής ,που επιλύει την εξίσωση Sylvester με την Bartels-Stewart μέθοδο.

2.5.3 Η Hessenberg-Schur μέθοδος για την εξίσωση Sylvester

Σε αυτήν την μέθοδο για την λύση της εξίσωσης Sylvester θα μπορούσαμε να πάρουμε τους πίνακες A και B και να μετασχηματίσουμε τον πίνακα A σε κάτω πραγματικό Schur πίνακα ενώ τον πίνακα B σε άνω πραγματικό Schur πίνακα, και μετά να χρησιμοποιήσουμε πίσω αντικατάσταση για να λύσουμε το ελαττωμένο Schur πρόβλημα όπως κατά κάποιο τρόπο κάναμε παραπάνω. Μπορούμε όμως να λιγοστέψουμε την αριθμητική προσπάθεια εάν τον B , τον έστω μεγαλύτερο σε διάσταση από τους πίνακες A και B , τον αφήσουμε σε Hessenberg μορφή, ενώ τον πίνακα A , τον έστω μικρότερο από τους δυο πίνακες, τον μετασχηματίσουμε παραπέρα σε πραγματικό Schur πίνακα. Ο λόγος που συμβαίνει αυτό είναι πως ένας πίνακας πρέπει σαν πρωταρχικό βήμα να μετασχηματιστεί σε Hessenberg μορφή για να ελαττωθεί αργότερα σε πραγματική μορφή Schur. Το καλό είναι ότι και εδώ είναι δυνατή η χρήση της πίσω αντικατάστασης για την επίλυση του Hessenberg-Schur προβλήματος.

Βήμα 1. *Αναγωγή στο Hessenberg-Schur πρόβλημα.* Ας υποθέσουμε ότι το m είναι μεγαλύτερο του n , όπου $n \times n$ διάσταση του πίνακα A και $m \times m$ διάσταση του πίνακα B . Ας είναι $R = U^T A^T U$ και $H = V^T B V$ η άνω πραγματική Schur μορφή και η άνω Hessenberg μορφή των A και B αντίστοιχα. Τότε η

$$XA + BX = C \quad \text{γίνεται} \quad YR^T + HY = \hat{C},$$
$$\text{όπου } Y = V^T X U, \quad \hat{C} = V^T C U.$$

Βήμα 2. *Επίλυση του αναγομένου Hessenberg-Schur προβλήματος.* Στο αναγόμενο πρόβλημα $HY + YR^T = \hat{C}$ ας έχουμε ότι $Y = (y_1, y_2, \dots, y_n)$ και $\hat{C} = (c_1, c_2, \dots, c_n)$. Τότε

ας υποθέσουμε ότι οι στήλες y_{k+1}, \dots, y_n έχουν ήδη υπολογιστεί, οι στήλες y_k (ή οι στήλες y_k και y_{k+1}) μπορούν να υπολογιστούν σύμφωνα με τις δυο παρακάτω περιπτώσεις.

Περίπτωση 1. Εάν $r_{k,k-1} = 0$, η στήλη y_k υπολογίζεται λύνοντας το $m \times m$ Hessenberg σύστημα:

$$(H + r_{kk}I)y_k = c_k - \sum_{j=k+1}^n r_{kj}y_j.$$

Περίπτωση 2. Εάν $r_{k,k-1} \neq 0$, τότε εξισώνοντας τις στήλες $k-1$ και k στο $HY + YR^T = \hat{C}$, βλέπουμε ότι οι στήλες y_{k-1} και y_k μπορούν να υπολογιστούν ταυτόχρονα λύνοντας το $2m \times 2m$ γραμμικό σύστημα:

$$\begin{aligned} H(y_{k-1}, y_k) + (y_{k-1}, y_k) \begin{pmatrix} r_{k-1,k-1} & r_{k,k-1} \\ r_{k-1,k} & r_{kk} \end{pmatrix} = \\ = (c_{k-1}, c_k) - \sum_{j=k+1}^n (r_{k-1,j}y_j, r_{kj}y_j) = (d_{k-1}, d_k) \end{aligned}$$

Σημείωση: Ο πίνακας του συστήματος μπορεί να γίνει άνω τριγωνικός με δυο μη μηδενικές υποδιαγωνίους, αναδιατάσσοντας κατάλληλα τις μεταβλητές. Το άνω αυτό τριγωνικό σύστημα μπορεί να λυθεί χρησιμοποιώντας πίσω αντικατάσταση.

Βήμα 3. Παίρνοντας την αυθεντική λύση X . Η λύση X προκύπτει από την Y ως εξής:

$$X = VYU^T.$$

Αλγόριθμος 2.5.2. Ο Hessenberg-Schur αλγόριθμος για την $XA + BX = C$

Είσοδος: Οι πίνακες A, B, C , διάστασης $n \times n, m \times m$ και $m \times n$ αντίστοιχα, με $n \leq m$.

Έξοδος: Ο πίνακας X που ικανοποιεί την $XA + BX = C$.

**Αριθμητική επίλυση των εξίσωσεων Sylvester – Η εξίσωση
Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab**

Βήμα 1. Μετασχημάτισε τον A^T σε έναν πραγματικό πίνακα Schur R , και τον B σε ένα άνω πίνακα Hessenberg H με τον εξής ορθογώνιο μετασχηματισμό :

$$U^T A^T U = R, \quad V^T B V = H.$$

Σχημάτισε τον $\hat{C} = V^T C U$, και διαμέρισε τον σε στήλες, δηλαδή $\hat{C} = (c_1, \dots, c_n)$.

Βήμα 2. Λύσε την $HY + YR^T = \hat{C}$:

For $k = n, \dots, 1$ do μέχρι οι στήλες του Y να υπολογιστούν

If $r_{k,k-1} = 0$, τότε υπολόγισε την στήλη y_k λύνοντας το Hessenberg σύστημα:

$$(H + r_{kk}I)y_k = c_k - \sum_{j=k+1}^n r_{kj}y_j$$

Else, υπολόγισε τις στήλες y_k και y_{k-1} λύνοντας το σύστημα:

$$\begin{pmatrix} H + r_{k-1,k-1}I & r_{k-1k}I \\ r_{k,k-1}I & H + r_{kk}I \end{pmatrix} \begin{pmatrix} y_{k-1} \\ y_k \end{pmatrix} = \begin{pmatrix} d_{k-1} \\ d_k \end{pmatrix},$$

όπου

$$(d_{k-1}, d_k) = (c_{k-1}, c_k) - \sum_{j=k+1}^n (r_{k-1,j}y_j, r_{kj}y_j).$$

Βήμα 3. Πάρε την λύση X : $X = V Y U^T$.

Παράδειγμα 2.5.3 Ας υποθέσουμε ότι έχουμε να λύσουμε την εξίσωση $XA + BX = C$ χρησιμοποιώντας τον αλγόριθμο 2.5.2. Έστω λοιπόν

$$A = \begin{pmatrix} 1 & -1 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 2 \end{pmatrix}, \quad B = \begin{pmatrix} 1 & 2 & 3 & 4 \\ 4 & 5 & 6 & 7 \\ 7 & 8 & 9 & 1 \\ 10 & 0 & 0 & 0 \end{pmatrix}$$

και

$$C = \begin{pmatrix} 12 & 10 & 12 \\ 24 & 22 & 24 \\ 27 & 25 & 27 \\ 12 & 10 & 12 \end{pmatrix}$$

οι πίνακες μας.

Βήμα 1. Ελάττωσε τον A^T σε πραγματική Schur μορφή και τον B σε μορφή Hessenberg:

$$U^T A^T U = R = \begin{pmatrix} 1 & 1 & 0 \\ -1 & 1 & 0 \\ 0 & 0 & 2 \end{pmatrix},$$

$$V^T B V = H = \begin{pmatrix} 1 & -5.3766 & -0.3709 & -0.0886 \\ -12.8452 & 7.6545 & 5.3962 & -0.7695 \\ 0 & 10.6689 & 4.7871 & -0.2737 \\ 0 & 0 & -5.3340 & 1.5584 \end{pmatrix}.$$

$$U = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \quad V = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & -3.114 & -0.7398 & -0.5965 \\ 0 & -5.449 & -0.3752 & 0.7498 \\ 0 & -0.7785 & 0.5585 & -0.2863 \end{pmatrix}$$

Υπολόγισε τον

$$\hat{C} = \begin{pmatrix} 12 & 10 & 12 \\ -31.5292 & -28.2595 & -31.5292 \\ -21.1822 & -20.0693 & -21.1822 \\ 2.4949 & 2.7608 & 2.4949 \end{pmatrix}.$$

Βήμα 2. Λύση του ελαττωμένου προβλήματος: $HY + YR^T = \hat{C}$.

**Αριθμητική επίλυση των εξίσωσεων Sylvester – Η εξίσωση
Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab**

Περίπτωση 1. Καθώς $r_{32} = 0$, την στήλη y_3 την παίρνουμε λύνοντας το σύστημα $(H + r_{33}I)y_3 = c_3$. Άρα

$$y_3 = (1 \quad -1.6348 \quad -0.5564 \quad -0.1329)^T.$$

Περίπτωση 2. Καθώς $r_{21} \neq 0$, τις στήλες y_1 και y_2 τις υπολογίζουμε ταυτόχρονα λύνοντας το σύστημα:

$$\begin{pmatrix} H + r_{11}I & r_{12}I \\ r_{21}I & H + r_{22}I \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \end{pmatrix} = \begin{pmatrix} d_1 \\ d_2 \end{pmatrix},$$

όπου $(d_1, d_2) = (c_1 - r_{13}y_3, c_2 - r_{23}y_3)$.

$$y_1 = \begin{pmatrix} 1 \\ -1.6348 \\ -0.5564 \\ -0.1329 \end{pmatrix}, \quad y_2 = \begin{pmatrix} 1 \\ -1.6348 \\ -0.5564 \\ -0.1329 \end{pmatrix}.$$

Οπότε,

$$Y = (y_1, y_2, y_3) = \begin{pmatrix} 1 & 1 & 1 \\ -1.6348 & -1.6348 & -1.6348 \\ -0.5564 & -0.5564 & -0.5564 \\ -0.1329 & -0.1329 & -0.1329 \end{pmatrix}.$$

Βήμα 3.

$$X = VYU^T = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix}.$$

Αριθμός Flops:

1. Ελάττωση σε πραγματική μορφή Schur και σε μορφή Hessenberg:

$$\frac{10}{3}m^3 + 26n^3.$$

2. Υπολογισμός του \hat{C} : $2m^2n + 2mn^2$.
3. Υπολογισμός του Y : $6m^2n + mn^2$.
4. Υπολογισμός του X : $2m^2n + 2mn^2$.

Συνολικά Flops :Περίπου $(10m^3/3 + 26n^3 + 10m^2n + 5mn^2)$.

Αριθμητική ευστάθεια της Hessenberg – Schur μεθόδου: Η ανάλυση του σφάλματος στρογγύλευσης της Hessenberg – Schur μεθόδου για την εξίσωση Sylvester $XA + BX = C$ που έγινε από τον Golub et al.(1979) μας δείχνει ότι ‘σφάλματα όχι μεγαλύτερα από $O(\|\phi^{-1}\|_2 \varepsilon)$ θα υπάρχουν μέσα στην υπολογισμένη λύση \hat{X} , όπου $\|\phi^{-1}\|_2 = 1/\text{sep}(B, -A)$, και ε είναι ένα μικρό πολλαπλάσιο της ακρίβειας της μηχανής μ .

Συγκεκριμένα εάν

$$\frac{\varepsilon(2 + \varepsilon)(\|A\|_2 + \|B\|_2)}{\text{sep}(B, -A)} < \frac{1}{2},$$

τότε,

$$\frac{\|X - \hat{X}\|_F}{\|X\|_F} \leq \frac{(9\varepsilon + 2\varepsilon^2)(\|A\|_F + \|B\|_F)}{\text{sep}(B, -A)}.$$

Το παραπάνω αποτέλεσμα δείχνει ότι η ποσότητα $\text{sep}(B, -A)$ πράγματι επηρεάζει την αριθμητική ακρίβεια της υπολογιζόμενης λύσης που πήραμε από τον αλγόριθμο Hessenberg – Schur για την εξίσωση Sylvester.

Στο παράρτημα παραθέτουμε ένα πρόγραμμα ,την συνάρτηση **HessSch**, που έφτιαξε ο συγγραφέας αυτής της διπλωματικής ,που επιλύει την εξίσωση Sylvester με την Hessenberg-Schur μέθοδο.

Αριθμητική επίλυση των εξίσωσεων Sylvester – Η εξίσωση Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab

2.5.4 Η Hessenberg μέθοδος για την εξίσωση Sylvester $AX+XB=C$

Αν και η Hessenberg – Schur μέθοδος είναι αριθμητικά σταθερή για την εξίσωση Sylvester και χρησιμοποιείται ευρέως, θα ήταν ωραίο να υπάρχει και μια μέθοδος η οποία θα απαιτεί την ελάττωση των πινάκων A και B σε μορφή Hessenberg μονάχα. Έτσι θα έχουμε λιγότερες πράξεις, αφού είπαμε ότι για να μετατρέψουμε έναν πίνακα σε πραγματική μορφή Schur θα πρέπει πρώτα να ελαττωθεί σε Hessenberg μορφή, και η μέθοδος θα είναι πιο αποδοτική. Η αριθμητική ευστάθεια της μεθόδου αυτής δεν έχει ερευνηθεί ακόμα και έτσι η μέθοδος αυτή έχει κυρίως θεωρητικό ενδιαφέρον.

Βήμα 1. Αναγωγή του προβλήματος σε πρόβλημα Hessenberg. Μετασχημάτισε τον πίνακα A σε κάτω πίνακα Hessenberg H_1 , και τον B πάλι σε κάτω πίνακα Hessenberg H_2 .

$$U^T AU = H_1, \quad V^T BV = H_2.$$

Ας υποθέσουμε ότι οι πίνακες H_1 και H_2 δεν ανάγονται παραπέρα.

Τότε η $AX + XB = C$ γίνεται

$$H_1 Y + Y H_2 = C', \quad \text{όπου } Y = U^T X V, \quad C' = U^T C V.$$

Βήμα 2. Επίλυση του αναγόμενου προβλήματος. $H_1 Y + Y H_2 = C'$. Ας είναι $Y = (y_1, y_2, \dots, y_n)$ και $H_2 = (h_{ij})$.

Τότε η εξίσωση $H_1 Y + Y H_2 = C'$ είναι ισοδύναμη με

$$\begin{aligned} H_1 y_n + h_{n-1,n} y_{n-1} + h_{nn} y_n &= c'_n, \\ H_1 y_{n-1} + h_{n-2,n-1} y_{n-2} + h_{n-1,n-1} y_{n-1} + h_{nn-1} y_n &= c'_{n-1}, \\ &\vdots \\ H_1 y_1 + h_{11} y_1 + h_{21} y_2 + \dots + h_{n1} y_n &= c'_1. \end{aligned}$$

Βγάζοντας τα y_1 έως y_{n-1} έχουμε

$$Ry_n = d,$$

όπου

$$R = \frac{1}{\prod_{i=2}^n h_{i-1,i}} \phi(-H_1),$$

όπου $\phi(x)$, είναι το χαρακτηριστικό πολυώνυμο του H_1 και το διάνυσμα d ορίζεται παρακάτω στο βήμα 4.

Έτσι αν λύσουμε το σύστημα $Ry_n = d$, μπορούμε να υπολογίσουμε τα y_{n-1} έως y_1 αναδρομικά ως εξής:

$$y_{i-1} = -\frac{1}{h_{i-1,i}} \left(H_1 y_i + \sum_{j=i}^n h_{ji} y_j - c'_i \right), \quad i = n, n-1, \dots, 2.$$

Βήμα 3. Υπολογίζοντας τον πίνακα R του βήματος 2. Είναι γνωστό ότι αν γνωρίζουμε μια γραμμή ή μια στήλη ενός πολυωνυμικού πίνακα μέσα σε έναν μη ελαττώσιμο πίνακα Hessenberg τότε και οι υπόλοιπες γραμμές ή στήλες του πίνακα μπορούν να βρεθούν αναδρομικά.

Ο πίνακας R είναι ένας πολυωνυμικός πίνακας μέσα σε έναν κάτω Hessenberg πίνακα H_1 και ο υπολογισμός του έχει ως εξής:

Εάν $R = (r_1, \dots, r_n)$, τότε αν ξέρουμε το r_n , τα r_{n-1}, \dots, r_1 μπορούν να βρεθούν αναδρομικά ως εξής:

**Αριθμητική επίλυση των εξίσωσεων Sylvester – Η εξίσωση
Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab**

$$r_{k-1} = \frac{1}{h'_{k-1,k}} \left(H_1 r_k - \sum_{i=k}^n h'_{ik} r_i \right),$$

όπου $H_1 = (h'_{ij})$, $k = n, n-1, \dots, 2$.

Άρα τώρα μένει μονάχα να μάθουμε πως θα υπολογίσουμε το r_n . Αυτό μπορεί να γίνει ως εξής. Θέτουμε $\theta_n = e_n = (0, 0, 0, \dots, 0, 1)^T$ και μετά υπολογίζουμε το θ_{n-1} έως θ_0 αναδρομικά χρησιμοποιώντας τον τύπο

$$\theta_{i-1} = -\frac{1}{h_{i-1,i}} \left(H_1 \theta_i + \sum_{j=i}^n h_{ji} \theta_j \right), \quad i = n, n-1, \dots, 1.$$

Τότε θα ισχύει ότι

$$r_n = \theta_0, \quad \text{θέτοντας } h_{01} = 1.$$

Βήμα 4. Υπολογίζοντας το διάνυσμα d του βήματος 2. Το διάνυσμα d μπορεί και αυτό να υπολογιστεί από την παραπάνω αναδρομή. Έτσι ξεκινώντας με $z_n = 0$ (μηδενικό διάνυσμα), εάν τα z_{n-1} έως z_0 υπολογίζονται αναδρομικά χρησιμοποιώντας τον τύπο

$$z_{i-1} = -\frac{1}{h_{i-1,i}} \left(H_1 z_i + \sum_{j=i}^n h_{ji} z_j - c'_i \right), \quad i = n, \dots, 2, 1,$$

τότε $d = -z_0$.

Βήμα 5. Παίρνοντας την πραγματική λύση X από την Y .

$$X = U Y V^T.$$

Σχόλιο

- Θέλουμε ξανά να τονίσουμε ότι αυτή η μέθοδος όπως παρουσιάστηκε παραπάνω έχει προς το παρόν μονάχα θεωρητικό ενδιαφέρον. Υπάρχουν πιθανές αριθμητικές δυσκολίες όπως, για παράδειγμα, εάν μια ή περισσότερες από τις εισόδους των υποδιαγωνίων του πίνακα Hessenberg H_2 είναι μικρές, τότε στον υπολογισμό του y_{i-1} του βήματος 2 αναμένεται να υπάρχει μεγάλο σφάλμα στρογγυλοποίησης. Πρέπει πρώτα να γίνει μια λεπτομερής έρευνα της αριθμητικής συμπεριφοράς της μεθόδου αυτής για να χρησιμοποιηθεί και πρακτικά. Είναι σίγουρο πως θα χρειαστεί να γίνουν και μερικές αλλαγές για την αριθμητική υλοποίηση του αλγορίθμου.

Παράδειγμα 2.5.4. Ας πούμε ότι θέλουμε να λύσουμε την εξίσωση $AX + BX = C$ και έχουμε τα παρακάτω δεδομένα

$$A = \begin{pmatrix} 1 & 2 & 3 & 4 \\ 4 & 5 & 6 & 7 \\ 7 & 8 & 9 & 1 \\ 10 & 0 & 0 & 0 \end{pmatrix}, \quad B = \begin{pmatrix} 1 & -1 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 2 \end{pmatrix}, \quad C = \begin{pmatrix} 12 & 10 & 12 \\ 24 & 22 & 24 \\ 27 & 25 & 27 \\ 12 & 10 & 12 \end{pmatrix},$$

Βήμα 1. Ελάττωση των πινάκων A και B σε κάτω Hessenberg πίνακες:

$$H_1 = \begin{pmatrix} 1.0000 & -5.3852 & 0 & 0 \\ -12.8130 & 8.7241 & 5.1151 & 0 \\ 0.8337 & 10.3127 & 4.6391 & 0.1586 \\ 0.3640 & 1.3595 & -4.8552 & 0.6368 \end{pmatrix},$$

$$U = \begin{pmatrix} 1.0000 & 0 & 0 & 0 \\ 0 & -0.3714 & -0.6009 & -0.7078 \\ 0 & -0.5571 & -0.4657 & 0.6876 \\ 0 & -0.7428 & 0.6497 & -0.1618 \end{pmatrix},$$

**Αριθμητική επίλυση των εξίσωσεων Sylvester – Η εξίσωση
Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab**

$$H_2 = \begin{pmatrix} 1 & 1 & 0 \\ -1 & 1 & 0 \\ 0 & 0 & 2 \end{pmatrix}, \quad V = \begin{pmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{pmatrix},$$

$$C' = \begin{pmatrix} 12.0000 & -10.0000 & 12.0000 \\ -32.8681 & 29.5256 & -32.8681 \\ -19.1978 & 18.3641 & -19.1978 \\ -0.3640 & -0.0000 & -0.3640 \end{pmatrix}.$$

Βήμα 2. Επίλυση του ελαττωμένου προβλήματος: Καθώς ο πίνακας H_2 είναι ελαττωμένος ($h_{23} = 0$), επειδή θα έχουμε κατάρρευση του αλγόριθμου θα προσπαθήσουμε καλύτερα να λύσουμε απευθείας το σύστημα:

$$\begin{aligned} H_1 y_3 + h_{33} y_3 &= c'_3, \\ H_1 y_2 + h_{12} y_1 + h_{22} y_2 &= c'_2 - h_{32} y_3 = \hat{c}_2, \\ H_1 y_1 + h_{11} y_1 + h_{21} y_2 &= c'_1 - h_{31} y_3 = \hat{c}_1. \end{aligned}$$

Το διάνυσμα y_3 το παίρνουμε σαν λύση του πρώτου συστήματος, και αφού ξέρουμε το y_3 και ξέρουμε τα \hat{c}_2 και \hat{c}_1 εύκολα υπολογίζουμε τα y_1 και y_2 .

$$y_3 = \begin{pmatrix} 1.0000 \\ -1.6713 \\ -0.4169 \\ -0.1820 \end{pmatrix}, \quad \hat{c}_2 = \begin{pmatrix} -10.0000 \\ 29.5256 \\ 18.3641 \\ -0.0000 \end{pmatrix}, \quad \hat{c}_1 = \begin{pmatrix} 12.0000 \\ -32.8681 \\ -19.1978 \\ -0.3640 \end{pmatrix}.$$

Ξεκινώντας από το βήμα 3 για τον υπολογισμό των διανυσματων z_i, d και προχωρώντας στη συνέχεια στο βήμα 4 οδηγούμαστε στον υπολογισμό των y_2 και y_1 .

Βήμα 3. Υπολογισμός του διανύσματος d . Ξεκινάμε με $z_2 = (0 \ 0 \ 0 \ 0)^T$,

Δημήτριος Χ. Κατωτοκίδης

$$z_1 = -\frac{1}{h_{12}}(H_1 z_2 + h_{22} z_2 - \hat{c}_2) = (-10.0000 \quad 29.5256 \quad 18.3641 \quad -0.0000)^T,$$

$$d = -z_0 = \frac{1}{1}(H_1 z_1 + h_{11} z_1 + h_{21} z_2 - \hat{c}_1) = (-191.0000 \quad 542.0447 \quad 418.9050 \quad -52.2985)^T.$$

Βήμα 4. Υπολογισμός του πίνακα R . Ξεκινώντας από το $\theta_2 = (0 \quad 0 \quad 0 \quad 1)^T$,

$$\theta_1 = -\frac{1}{h_{12}}(H_1 \theta_2 + h_{22} \theta_2) = (0 \quad 0 \quad -0.1586 \quad -1.6368)^T,$$

$$\theta_0 = -\frac{1}{1}(H_1 \theta_1 + h_{11} \theta_1 + h_{21} \theta_2) = (0 \quad 0.8112 \quad 1.1538 \quad 2.9092)^T$$

και θα έχουμε

$$r_4 = \theta_0$$

$$r_3 = \frac{1}{h'_{34}}(H_1 r_4 - h'_{44} r_4) \\ = (-27.5462 \quad 78.5858 \quad 84.7805 \quad -28.3717)^T,$$

$$r_2 = \frac{1}{h'_{23}}(H_1 r_3 - h'_{33} r_3 - h'_{43} r_4) \\ = (-63.1364 \quad 217.3103 \quad 154.1618 \quad -36.5856)^T,$$

$$r_1 = \frac{1}{h'_{12}}(H_1 r_2 - h'_{22} r_2 - h'_{32} r_3 - h'_{42} r_4) \\ = (74.0000 \quad -145.9565 \quad -125.7098 \quad -20.1428)^T,$$

Άρα έχουμε

$$R = \begin{pmatrix} 74.0000 & -63.1364 & -27.5462 & 0 \\ -145.9565 & 217.3103 & 78.5858 & 0.8112 \\ -125.7098 & 154.1618 & 84.7805 & 1.1538 \\ -20.1428 & -36.5856 & -28.3717 & 2.9092 \end{pmatrix}.$$

Από την εξίσωση $Ry_2 = d$ παίρνουμε

**Αριθμητική επίλυση των εξίσωσεων Sylvester – Η εξίσωση
Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab**

$$y_2 = (-1.0000 \quad 1.6713 \quad 0.4169 \quad 0.1820)^T$$

και τέλος υπολογίζουμε

$$y_1 = (1.0000 \quad -1.6713 \quad -0.4169 \quad -0.1820)^T.$$

έτσι η λύση του αναγόμενου ή αλλιώς του ελαττωμένου προβλήματος (reduced problem) Y θα είναι

$$Y = \begin{pmatrix} 1.0000 & -1.0000 & 1.0000 \\ -1.6713 & 1.6713 & -1.6713 \\ -0.4169 & 0.4169 & -0.4169 \\ -0.1820 & 0.1820 & -0.1820 \end{pmatrix}.$$

Η πραγματική λύση X είναι

$$X = U Y V^T = \begin{pmatrix} 1.0000 & 1.0000 & 1.0000 \\ 1.0000 & 1.0000 & 1.0000 \\ 1.0000 & 1.0000 & 1.0000 \\ 1.0000 & 1.0000 & 1.0000 \end{pmatrix}.$$

Επίσης θα ισχύει $\|AX + XB - C\|_2 = 5.6169 \times 10^{-14}$.

2.5.5 Η Hessenberg-Schur μέθοδος για την διακριτή εξίσωση Sylvester

Σε μερικές εφαρμογές μας είναι χρήσιμο να λύσουμε την διακριτή εξίσωση Sylvester:

$$BXA + C = X.$$

Ας υποθέσουμε ότι η τάξη του πίνακα A είναι μικρότερη από αυτήν του πίνακα B .

$A \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{m \times m}$. Ας μετατρέψουμε τους πίνακες A^T και B , αντίστοιχα, σε έναν πραγματικό άνω Schur πίνακα και σε έναν άνω Hessenberg πίνακα με ορθογώνιους μετασχηματισμούς:

$$\begin{aligned}U^T A^T U &= R, \\V^T B V &= H.\end{aligned}$$

Τότε η $BXA + C = X$ γίνεται $HYR^T + \hat{C} = Y$, όπου $Y = V^T XU$, $\hat{C} = V^T CU$.

Έστω επίσης

$$Y = (y_1, \dots, y_n) \text{ και } \hat{C} = (c_1, c_2, \dots, c_n).$$

Η ελαττωμένη εξίσωση τώρα μπορεί να λυθεί με τον ίδιο ακριβώς τρόπο όπως με τον Hessenberg-Schur αλγόριθμο της εξίσωσης Sylvester (αλγόριθμος 2.5.2). Ένας αλγόριθμος επίλυσης αυτής της εξίσωσης παρουσιάζεται παρακάτω στις εφαρμογές καθώς και η συνάρτηση **discreteHessSch** που έφτιαξε ο συγγραφέας της διπλωματικής αυτής για την επίλυση της εξίσωσης αυτής.

2.6 Σύγκριση διαφορετικών μεθόδων και συμπεράσματα

Ο υπολογισμός της λύσης της εξίσωσης Sylvester μέσω του γινόμενου Kronecker και της χρήσης της απαλοιφής Gauss στο προκύπτον σύστημα $Px = c$, αν δεν μπορούμε να εκμεταλλευτούμε την ειδική δομή του πίνακα P , δεν συνίσταται στην πράξη μιας και απαιτεί πολλά flops υπολογισμού και αποθήκευσης. Επίσης το γινόμενο του Kronecker είναι αριθμητικά αρκετά ευαίσθητο.

Οι μέθοδοι που βασίζονται σε αναγωγή σε πίνακα Jordan και σε συνοδεύοντες πίνακες θα δώσουν γενικά ανακριβή αποτελέσματα αν οι πίνακες μετασχηματισμού είναι σε αρρωστημένη κατάσταση. Έτσι οι μέθοδοι αυτοί θα πρέπει να αποφεύγονται για αριθμητικούς υπολογισμούς.

Αριθμητική επίλυση των εξίσωσεων Sylvester – Η εξίσωση Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab

Έτσι από αριθμητικής άποψης, και επειδή η μέθοδος Hessenberg για την επίλυση της εξίσωσης Sylvester βρίσκεται ακόμα σε θεωρητικό στάδιο, οι μέθοδοι επιλογής μας για την λύση της εξίσωσης Sylvester, είναι η μέθοδος Schur , δηλαδή ο αλγόριθμος 2.5.1 των Bartels-Stewart , και η μέθοδος Hessenberg-Schur. Και οι δυο αυτές μέθοδοι είναι σε τέλεια θα λέγαμε καλή κατάσταση. Έχουν τις ίδιες ικανοποιητικές αριθμητικές ιδιότητες και το ίδιο ισχύει για τα σφάλματα στρογγυλοποίησης. Όσον αφορά την ακρίβεια των λύσεων, και στις δυο μεθόδους βλέπουμε ότι είναι πρακτικά ίδιες.

Όμως ο αλγόριθμος Hessenberg-Schur έχει ένα πλεονέκτημα από τον αλγόριθμο Bartels-Stewart όταν $n \geq m$. Αν για παράδειγμα $n = 4m$, τότε ο αλγόριθμος Hessenberg-Schur χρειάζεται το $\frac{1}{4}$ της δουλειάς που χρειάζεται ο αλγόριθμος των Bartels-Stewart για να επιλύσει το πρόβλημα. Εάν πάλι $n < m$ τότε μπορούμε να χρησιμοποιήσουμε πάλι τον αλγόριθμο Hessenberg-Schur αλλά στο ανάστροφο πρόβλημα

$$\boxed{B^T X^T + X^T A^T = C^T}$$

αντί του $AX + XB = C$.

Ο απαραίτητος χώρος αποθήκευσης του Hessenberg-Schur αλγορίθμου είναι σχετικά λιγάκι πιο μεγάλος από αυτόν του Bartels-Stewart αλγορίθμου, αλλά αυτό είναι κάτι το οποίο δεν μας πειράζει και πολύ.

Έτσι κανείς θα έλεγε πως ο αλγόριθμος Hessenberg-Schur είναι πιο αποτελεσματικός, πιο αποδοτικός από αυτόν του Bartels-Stewart και είναι αυτός που προς το παρόν ,ανάμεσα από άλλους αλγορίθμους, συστήνεται στην πράξη για αριθμητικούς υπολογισμούς. Αυτό βέβαια δεν σημαίνει ότι δεν προτείνουμε τον αλγόριθμο του Bartels-Stewart που είναι εξίσου καλός.

2.7 Επίλυση της γενικευμένης εξίσωσης Sylvester $AXB^T + CXD^T = E$

2.7.1 Η γενικευμένη εξίσωση Sylvester

Μια εξίσωση πινάκων που θα μας κινούσε το ενδιαφέρον στην θεωρία έλεγχου είναι η εξίσωση τύπου Sylvester

$$AXB^T + CXD^T = E \quad (2.8)$$

στην οποία έχουμε τους πίνακες A και C να είναι διάστασης $m \times m$, τους πίνακες B και D να είναι διάστασης $n \times n$. Ο πίνακας E καθώς και η επιθυμητή λύση X έχουν διάσταση $m \times n$. Η εξίσωση (2.8) είναι μια ειδική περίπτωση της γενικής γραμμικής εξίσωσης

$$\sum_{i=1}^N A_i X B_i = E$$

την οποία πρώτος μελέτησε ο Sylvester.

Η εξίσωση (2.8) είναι γνωστή ως *γενικευμένη εξίσωση Sylvester*.

Η εξίσωση (2.8) έχει μοναδική λύση αν και μόνο εάν οι πίνακες pencil $A - \lambda C$ και $D - \lambda B$ είναι κανονικοί δηλαδή να ισχύει $\det(A - \lambda C) \neq 0$ και $\det(D - \lambda B) \neq 0$ και το φάσμα των ιδιοτιμών του ενός πίνακα pencil είναι διαφορετικό από το αρνητικό του φάσματος ιδιοτιμών του άλλου πίνακα pencil. Έτσι οποιοσδήποτε από τους πίνακες A, B, C, D μπορεί να είναι singular χωρίς να υπάρχει κάποιο ιδιαίτερο πρόβλημα, ενώ θα πρέπει να ισχύουν ορισμένες συνθήκες για να είναι ένας δεύτερος από αυτούς τους πίνακες singular. Οι αλγόριθμοι μας μπορούν να χειριστούν τέτοιες περιπτώσεις.

2.7.2 Αριθμός κατάστασης της γενικευμένης εξίσωσης Sylvester

Αριθμητική επίλυση των εξίσωσεων Sylvester – Η εξίσωση Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab

Μια εκτίμηση του αριθμού κατάστασης k της γενικευμένης εξίσωσης Sylvester είναι

$$k = \|G\| \|G\|^{-1} \approx (\|A\| \|B\| + \|C\| \|D\|) / sep$$

όπου $G = B \otimes A + D \otimes C$ και το sep εδώ ορίζεται ως

$$sep = \min_{Z \neq 0} \|AZB^T + CZD^T\| / \|Z\|$$

2.7.3 Προσπαθώντας να επιλύσουμε την γενικευμένη εξίσωση Sylvester

Ένας εύκολος τρόπος για να λύσουμε την εξίσωση (2.8) είναι να γράψουμε την εξίσωση μας στην μορφή

$$Gy = h$$

όπου

$$G = B \otimes A + D \otimes C$$

και

$$y = (x_{11}, x_{21}, \dots, x_{m1}, x_{12}, \dots, x_{mn})^T$$

$$h = (e_{11}, e_{21}, \dots, e_{m1}, e_{12}, \dots, e_{mn})^T.$$

Υπενθυμίζουμε πως αν έχουμε π.χ δυο πίνακες M, N τότε ο πίνακας $M \otimes N$ ορίζεται ως το Kronecker γινόμενο των πινάκων M, N . Το block στοιχείο του block πίνακα $M \otimes N$ στην θέση (i, j) είναι $m_{ij}N$. Η εξίσωση $Gy = h$ μπορεί να λυθεί με απαλοιφή Gauss. Παρόλα αυτά ο πίνακας G θα έχει διάσταση $mn \times mn$, κάτι το οποίο κάνει την μέθοδο αυτή μη-πρακτική όταν έχουμε μεγάλης διάστασης συστήματα.

Μια δεύτερη επιλογή είναι να πάρουμε την εξίσωση (2.8) και να την γράψουμε στην μορφή $FX + XG = H$ όπου για την επίλυση της στην μορφή αυτή υπάρχουν ήδη

αποτελεσματικοί αλγόριθμοι. Συγκεκριμένα θα έχουμε ότι η εξίσωση (2.8) μπορεί να γραφτεί ως

$$\underbrace{C^{-1}A}_F X + X \underbrace{D^T B^{-T}}_G = \underbrace{C^{-1}EB^{-T}}_H$$

όπου B, C είναι nonsingular πίνακες.

Η εξίσωση (2.8) μπορεί τέλος να λυθεί αν γραφτεί ως ζευγάρι δυο εξισώσεων

$$\begin{aligned} AR - LD^T &= E \\ CR + LB^T &= 0 \end{aligned}$$

όπου $XB^T = R$ και $CX = -L$. Θα πρέπει εδώ τουλάχιστον ένας από τους πίνακες B, C να είναι nonsingular έτσι ώστε να μπορούμε να πάρουμε την λύση X μέσω των πινάκων L, R .

Αυτές όμως οι μέθοδοι δεν είναι γενικά ικανοποιητικές στην περίπτωση που οι πίνακες B, C είναι singular ή βρίσκονται σε αρρωστημένη κατάσταση όσον αφορά την αντιστροφή. Έτσι βλέπουμε ότι χρειαζόμαστε μια πιο γενική μέθοδο. Έτσι παρακάτω δεν έχουμε παρά να παρουσιάζουμε δυο ικανοποιητικές μεθόδους, την μέθοδο των Bartels-Stewart και την μέθοδο Hessenberg-Schur, για την επίλυση της γενικευμένης εξίσωσης Sylvester $AXB^T + CXD^T = E$.

Η Bartels-Stewart και η Hessenberg-Schur μέθοδος αναμένεται να έχουν ικανοποιητικές αριθμητικές ιδιότητες γιατί και οι δυο μέθοδοι χρησιμοποιούν τον QZ αλγόριθμο και την απαλοιφή Gauss δυο αλγορίθμους που είναι αριθμητικά σταθεροί.

2.7.4 Η μέθοδος Bartels-Stewart για την επίλυση της γενικευμένης εξίσωσης Sylvester

Αριθμητική επίλυση των εξισώσεων Sylvester – Η εξίσωση Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab

Η μέθοδος Bartels-Stewart όπως είδαμε είναι μια μέθοδος που δημιουργήθηκε για την εξίσωση $AX + XB = C$. Η μέθοδος αυτή μπορεί να επεκταθεί και να χρησιμοποιηθεί για την επίλυση της γενικευμένης εξίσωσης Sylvester $AXB^T + CXD^T = E$.

Συγκεκριμένα ξαναγράφουμε την εξίσωση (2.8) ως εξής:

$$(Q_1AZ_1)(Z_1^T XZ_2)(Z_2^T B^T Q_2^T) + (Q_1CZ_1)(Z_1^T XZ_2)(Z_2^T D^T Q_2^T) = Q_1EQ_2^T$$

όπου οι πίνακες Q_1, Z_1, Q_2, Z_2 είναι ορθογώνιοι πίνακες. Ο QZ αλγόριθμος μας επιτρέπει να διαλέξουμε τους πίνακες αυτούς έτσι ώστε η παραπάνω εξίσωση να ξαναγραφτεί ως

$$PYR^T + SYT^T = F$$

όπου οι πίνακες $Q_1AZ_1 = P$, $Q_2DZ_2 = T$ είναι σχεδόν πάνω τριγωνικοί, και οι πίνακες $Q_1CZ_1 = S$, $Q_2BZ_2 = R$ είναι πάνω τριγωνικοί. Επίσης θα έχουμε ότι $Z_1^T XZ_2 = Y$ και $Q_1EQ_2^T = F$.

Συγκεκριμένα ο QZ αλγόριθμος εφαρμόζεται σε καθένα από τα ζευγάρια (A,C) και (D,B) . Ο πρώτος πίνακας του ζευγαριού ανάγεται σε σχεδόν άνω τριγωνικό πίνακα, ενώ ο δεύτερος πίνακας του ζευγαριού ανάγεται σε άνω τριγωνικό.

Υπενθυμίζουμε πως ένας σχεδόν τριγωνικός πίνακας είναι ένας block τριγωνικός πίνακας με 1×1 και 2×2 block διαγώνια στοιχεία, τα οποία αναφέρονται αντίστοιχα στις πραγματικές και μιγαδικές ιδιοτιμές.

Επειδή η εξίσωση (2.8) γράφεται υπό την ειδική μορφή $PYR^T + SYT^T = F$, εμείς μπορούμε να πάρουμε την εξίσωση $PYR^T + SYT^T = F$ και να την λύσουμε με την τεχνική της πίσω αντικατάστασης.

Παρακάτω παρουσιάζουμε τον αλγόριθμο των Bartels-Stewart για την επίλυση της γενικευμένης εξίσωσης Sylvester $AXB^T + CXD^T = E$.

Αλγόριθμος 2.7.1. Ο Bartels-Stewart αλγόριθμος για την $AXB^T + CXD^T = E$

Είσοδος: Οι πίνακες A, B, C, D, E διάστασης $m \times m, n \times n, m \times m, n \times n$, και $m \times n$ αντίστοιχα.

Έξοδος: Ο πίνακας X που ικανοποιεί την $AXB^T + CXD^T = E$.

Βήμα 1. Μετασχημάτισε τους A, D σε πραγματικούς πίνακες Schur (σχεδόν άνω τριγωνικούς) P, T αντίστοιχα, και τους B, C σε άνω τριγωνικούς πίνακες R, S αντίστοιχα, με τους εξής ορθογώνιους πίνακες Q_1, Z_1, Q_2, Z_2 , που παίρνουμε αν εφαρμόσουμε τον αλγόριθμο QZ στα ζευγάρια (A, C) και (D, B) :

$$\begin{aligned} Q_1 A Z_1 &= P, & Q_2 D Z_2 &= T \\ Q_1 C Z_1 &= S, & Q_2 B Z_2 &= R. \end{aligned}$$

Σχημάτισε τον $F = Q_1 E Q_2^T$, και διαμέρισε τον σε στήλες, δηλαδή $F = (f_1, \dots, f_n)$.

Βήμα 2. Λύσε την $PYR^T + SYT^T = F$:

For $k = n, \dots, 1$ do μέχρι οι στήλες του Y να υπολογιστούν

If $t_{k,k-1} = 0$, τότε υπολόγισε την στήλη y_k λύνοντας το σχεδόν άνω τριγωνικό σύστημα:

$$(r_{kk}P + t_{kk}S)y_k = f_k - \sum_{j=k+1}^n (r_{kj}P + t_{kj}S)y_j$$

Else, υπολόγισε τις στήλες y_k και y_{k-1} λύνοντας το σύστημα:

$$\begin{pmatrix} r_{k-1,k-1}P + t_{k-1,k-1}S & r_{k-1,k}P + t_{k-1,k}S \\ t_{k,k-1}S & r_{kk}P + t_{kk}S \end{pmatrix} \begin{pmatrix} y_{k-1} \\ y_k \end{pmatrix} = \begin{pmatrix} d_{k-1} \\ d_k \end{pmatrix},$$

Αριθμητική επίλυση των εξίσωσεων Sylvester – Η εξίσωση Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab

όπου

$$(d_{k-1}, d_k) = (f_{k-1}, f_k) - \sum_{j=k+1}^n \left((r_{k-1,j}P + t_{k-1,j}S)y_j, (r_{kj}P + t_{kj}S)y_j \right).$$

Βήμα 3. Πάρε την λύση $X : X = Z_1 Y Z_2^T$.

Τα παραπάνω γραμμικά συστήματα μπορούν να λυθούν χρησιμοποιώντας απαλοιφή Gauss με μερική οδήγηση.

Στο παράρτημα παραθέτουμε ένα πρόγραμμα ,την συνάρτηση **genSchSch**, που έφτιαξε ο συγγραφέας αυτής της διπλωματικής ,που επιλύει την γενικευμένη εξίσωση Sylvester με την Bartels-Stewart μέθοδο.

2.7.5 Η Hessenberg-Schur μέθοδος για την επίλυση της γενικευμένης εξίσωσης Sylvester

Όπως για την εξίσωση $AX + XB = C$ εκτός από την μέθοδο των Bartels-Stewart έχουμε και την Hessenberg-Schur μέθοδο έτσι και για την γενικευμένη εξίσωση Sylvester εκτός από την Bartels-Stewart μέθοδο, η οποία έχει επεκταθεί κατάλληλα για να λύσει την γενικευμένη μας εξίσωση Sylvester, έχουμε την αντίστοιχη Hessenberg-Schur μέθοδο η οποία έχει και αυτή επεκταθεί κατάλληλα έτσι ώστε να εφαρμόζει στο πρόβλημα μας.

Η μέθοδος αυτή ονομάζεται Hessenberg-Schur μέθοδος επειδή ο μεγαλύτερος σε διάσταση από τους πίνακες A και D της εξίσωσης $AXB^T + CXD^T = E$, ας πούμε ο πίνακας A , ανάγεται μέσω του QZ αλγορίθμου σε άνω Hessenberg μορφή, ενώ ο D θα αναχθεί σε σχεδόν άνω τριγωνική μορφή γνωστή αλλιώς και ως πραγματική Schur μορφή. Οι πίνακες C και B θα αναχθούν και πάλι, όπως και στην Bartels-Stewart μέθοδο, σε άνω τριγωνικούς πίνακες.

Συγκεκριμένα θα έχουμε ότι όταν θα εφαρμόσουμε τον QZ αλγόριθμο (ο οποίος αποτελείται από δυο βήματα) στο πρώτο ζευγάρι πινάκων (A,C) τότε θα εφαρμοστεί μονάχα το πρώτο βήμα του αλγορίθμου. Έτσι ο πίνακας C θα αναχθεί σε άνω τριγωνική μορφή, αλλά ο πίνακας A θα αναχθεί μονάχα σε άνω Hessenberg μορφή.

Από την άλλη όταν εφαρμόσουμε τον QZ αλγόριθμο στο δεύτερο μας ζευγάρι πινάκων (D,B) τότε ο αλγόριθμος θα εφαρμοστεί κανονικά και ο πίνακας D θα αναχθεί σε πραγματική μορφή Schur ενώ ο πίνακας B θα αναχθεί σε άνω τριγωνικό πίνακα.

Συγκεκριμένα ο αλγόριθμος θα έχει ως εξής :

Αλγόριθμος 2.7.2. Ο Hessenberg-Schur αλγόριθμος για την $AXB^T + CXD^T = E$

Είσοδος: Οι πίνακες A, B, C, D, E διάστασης $m \times m, n \times n, m \times m, n \times n$, και $m \times n$ αντίστοιχα, με $n \leq m$.

Έξοδος: Ο πίνακας X που ικανοποιεί την $AXB^T + CXD^T = E$.

Βήμα 1. Μετασχημάτισε το ζευγάρι (A,C) στο ζευγάρι (P,S) , όπου ο πίνακας P είναι ένας άνω Hessenberg πίνακας και S είναι άνω τριγωνικός πίνακας, εφαρμόζοντας το πρώτο βήμα του QZ αλγορίθμου και μετασχημάτισε το ζευγάρι (D,B) στο ζευγάρι (T,R) όπου ο πίνακας T είναι σε πραγματική Schur μορφή ενώ ο πίνακας R είναι σε τριγωνική μορφή :

$$\begin{aligned} Q_1 A Z_1 &= P, & Q_2 D Z_2 &= T \\ Q_1 C Z_1 &= S, & Q_2 B Z_2 &= R. \end{aligned}$$

Σχημάτισε τον $F = Q_1 E Q_2^T$, και διαμέρισε τον σε στήλες, δηλαδή $F = (f_1, \dots, f_n)$.

Βήμα 2. Λύσε την $P Y R^T + S Y T^T = F$:

For $k = n, \dots, 1$ do μέχρι οι στήλες του Y να υπολογιστούν

If $t_{k,k-1} = 0$, τότε υπολόγισε την στήλη y_k λύνοντας το άνω Hessenberg σύστημα:

**Αριθμητική επίλυση των εξισώσεων Sylvester – Η εξίσωση
Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab**

$$(r_{kk}P + t_{kk}S)y_k = f_k - \sum_{j=k+1}^n (r_{kj}P + t_{kj}S)y_j$$

Else, υπολόγισε τις στήλες y_k και y_{k-1} λύνοντας το σύστημα:

$$\begin{pmatrix} r_{k-1,k-1}P + t_{k-1,k-1}S & r_{k-1,k}P + t_{k-1,k}S \\ t_{k,k-1}S & r_{kk}P + t_{kk}S \end{pmatrix} \begin{pmatrix} y_{k-1} \\ y_k \end{pmatrix} = \begin{pmatrix} d_{k-1} \\ d_k \end{pmatrix},$$

όπου

$$(d_{k-1}, d_k) = (f_{k-1}, f_k) - \sum_{j=k+1}^n ((r_{k-1,j}P + t_{k-1,j}S)y_j, (r_{kj}P + t_{kj}S)y_j).$$

Βήμα 3. Πάρε την λύση $X : X = Z_1 Y Z_2^T$.

Βεβαίως τα παραπάνω γραμμικά συστήματα μπορούν να λυθούν χρησιμοποιώντας απαλοιφή Gauss με μερική οδήγηση.

Εάν $n > m$ τότε μπορούμε να εφαρμόσουμε τον ίδιο αλγόριθμο στο ανάστροφο πρόβλημα

$$BX^T A^T + DX^T C^T = E^T.$$

Στο παράρτημα παραθέτουμε ένα πρόγραμμα ,την συνάρτηση **genHessSch**, που έφτιαξε ο συγγραφέας αυτής της διπλωματικής ,που επιλύει την γενικευμένη εξίσωση Sylvester με την Hessenberg-Schur μέθοδο.

2.7.6 Σύγκριση των δυο μεθόδων

Σε αυτήν την παράγραφο γίνεται σύγκριση των δυο μεθόδων που έχουμε για την επίλυση της γενικευμένης εξίσωσης Sylvester, της Hessenberg-Schur μεθόδου και της Bartels-Stewart μεθόδου.

Όσον αφορά την ακρίβεια των αποτελεσμάτων μας, θα θέλαμε να πούμε ότι η Hessenberg-Schur μέθοδος δεν έχει κάποιο ιδιαίτερο πλεονέκτημα απέναντι στην Bartels-Stewart μέθοδο. Θα λέγαμε πως στον τομέα της ακρίβειας οι δυο αυτές μέθοδοι είναι πανομοιότυπες.

Τώρα θα δώσουμε μια flop μέτρηση των δυο αυτών αλγορίθμων μας.

Στο πρώτο κάλεσμα του QZ αλγορίθμου για το ζευγάρι (A, C) , στην Hessenberg-Schur μέθοδο αναγάγουμε τον πίνακα A σε άνω μορφή Hessenberg και τον C σε άνω τριγωνική μορφή κάτι το οποίο απαιτεί $17.6 m^3 \text{ flops}$, ενώ στην Bartels-Stewart μέθοδο έχουμε αναγωγή του πίνακα A σε άνω πραγματική Schur μορφή και τον C σε άνω τριγωνική μορφή κάτι το οποίο απαιτεί $66 m^3 \text{ flops}$. Αυτό αν $m \geq n$. Ειδιάλλως θα έχουμε το ανάστροφο πρόβλημα.

Στο δεύτερο κάλεσμα του QZ αλγορίθμου για το ζευγάρι (D, B) , στην Hessenberg-Schur μέθοδο αναγάγουμε τον πίνακα D σε άνω πραγματική μορφή Schur και τον B σε άνω τριγωνική μορφή κάτι το οποίο απαιτεί $66 n^3 \text{ flops}$, ενώ ομοίως στην Bartels-Stewart μέθοδο έχουμε αναγωγή του πίνακα D σε άνω πραγματική Schur μορφή και τον B σε άνω τριγωνική μορφή κάτι το οποίο απαιτεί $66 n^3 \text{ flops}$.

Η αναγωγή του πίνακα E στον πίνακα $Q_1 E Q_2^T$ απαιτεί $2m^2 n + 2n^2 m \text{ flops}$.

Στο βήμα της πίσω αντικατάστασης όπου έχουμε την λύση των γραμμικών συστημάτων με την Hessenberg-Schur μέθοδο, αν ο πίνακας T , ο σχεδόν άνω τριγωνικός πίνακας, στην χειρότερη περίπτωση έχει τα στοιχεία στην πρώτη υποδιαγώνιο του μη μηδενικά τότε απαιτούνται $(6 + 5.5)m^2 n + 2mn^2 \text{ flops}$, ενώ αν τα

Αριθμητική επίλυση των εξισώσεων Sylvester – Η εξίσωση Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab

στοιχεία αυτά είναι μηδενικά τότε απαιτούνται $6m^2n + 2mn^2$ flops. Εδώ για την Bartels-Stewart μέθοδο χρειαζόμαστε $2m^2n + 2n^2m$ flops.

Για τον υπολογισμό της λύσης $X = Z_1 Y Z_2^T$ χρειαζόμαστε $2m^2n + 2n^2m$ flops.

Έτσι συνολικά για τον Hessenberg-Schur αλγόριθμο θα έχουμε $17.6m^3 + 66n^3 + (10 + 5.5)m^2n + 6mn^2$ flops, ενώ για την Bartels-Stewart μέθοδο θα έχουμε $66m^3 + 66n^3 + 6m^2n + 6mn^2$ flops.

Γενικά βλέπουμε ότι όπως και στην εξίσωση Sylvester έτσι και στην γενικευμένη εξίσωση Sylvester η Hessenberg-Schur μέθοδος έχει σημαντικό πλεονέκτημα απέναντι στην Bartels-Stewart μέθοδο ως προς τον αριθμό των flops που χρειάζονται για τον υπολογισμό της λύσης. Αυτό γιατί η Hessenberg-Schur μέθοδος χρησιμοποιεί μονάχα το πρώτο βήμα του QZ αλγορίθμου.

Αυτό δείχνει ότι η Hessenberg-Schur μέθοδος γίνεται ολοένα και πιο αποτελεσματική όσο το m ξεπερνά το n ή το n ξεπερνά το m (στο ανάστροφο πρόβλημα).

Τέλος παρουσιάζουμε ένα μικρό παράδειγμα όπου δυο από τους πίνακες της εξίσωσης μας είναι singular.

Παράδειγμα 2.7.1 Έχουμε

$$A = \begin{pmatrix} 0 & 1 \\ 0 & 2 \end{pmatrix}, \quad B = (2), \quad C = \begin{pmatrix} 3 & 4 \\ 0 & 0 \end{pmatrix}, \quad D = (1), \quad E = \begin{pmatrix} 9 \\ 4 \end{pmatrix}$$

Τότε θα έχουμε $(2A + C)X = E$, δηλαδή

$$2A + C = \begin{pmatrix} 0 & 2 \\ 0 & 4 \end{pmatrix} + \begin{pmatrix} 3 & 4 \\ 0 & 0 \end{pmatrix} = \begin{pmatrix} 3 & 5 \\ 0 & 2 \end{pmatrix}.$$

Η λύση που παίρνουμε από τους δυο αλγορίθμους είναι $X = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$, η οποία είναι συγχρόνως και η πραγματική λύση.

2.8 Εφαρμογές

Εδώ γίνεται υλοποίηση των αλγορίθμων στην Matlab με προγράμματα που έχει κάνει ο συγγραφέας αυτής της διπλωματικής, για την επίλυση της εξίσωσης Sylvester.

Εφαρμογή 2.8.1

Εάν $A = (a_{ij})$ και $B = (b_{ij})$ είναι άνω τριγωνικοί πίνακες διάστασης $m \times m$ και $n \times n$ αντίστοιχα, τότε ναδειχθεί ότι η λύση $X = (x_{ij})$ της εξίσωσης Sylvester $AX + XB = C$ μπορεί να βρεθεί από τον παρακάτω τύπο

$$x_{ij} = \frac{c_{ij} - \sum_{k=i+1}^m a_{ik}x_{kj} - \sum_{k=1}^{j-1} x_{ik}b_{kj}}{a_{ii} + b_{jj}}.$$

Απάντηση:

Εάν $A = (a_{ij})$ και $B = (b_{ij})$ είναι άνω τριγωνικοί πίνακες διάστασης $m \times m$ και $n \times n$, δηλαδή έχουμε

$$A = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1m} \\ 0 & a_{22} & \cdots & a_{2m} \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & a_{mm} \end{pmatrix}$$

και

**Αριθμητική επίλυση των εξισώσεων Sylvester – Η εξίσωση
Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab**

$$B = \begin{pmatrix} b_{11} & b_{12} & \cdots & b_{1n} \\ 0 & b_{22} & \cdots & b_{2n} \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & b_{nn} \end{pmatrix}$$

τότε εάν πάρουμε την εξίσωση $AX + XB = C$ και κάνουμε στοιχείο με στοιχείο της πράξης θα μας προκύψει ένα σύστημα εξισώσεων τάξης $m \times n$. Εάν πάρουμε τις m αυτές εξισώσεις θα παρατηρήσουμε ότι όντως κάθε μια από αυτές επαληθεύονται από τον παραπάνω τύπο.

Μάλιστα παρακάτω παρουσιάζουμε και ένα πρόγραμμα, μια συνάρτηση, που χρησιμοποιεί τον παραπάνω τύπο για να υπολογίσει την λύση X της εξίσωσης Sylvester $AX + XB = C$. Θα θέλαμε να τονίσουμε όμως πως ο παραπάνω τύπος, και άρα με την σειρά της η παρακάτω συνάρτηση, ισχύει μονάχα όταν οι πίνακες μας A, B είναι σε άνω τριγωνική μορφή. Αν αυτό δεν ισχύει τότε θα έχουμε εσφαλμένα αποτελέσματα.

Ακολουθεί λοιπόν η παρακάτω συνάρτηση, ενώ γίνεται επαλήθευση των αποτελεσμάτων της συνάρτησης, χρησιμοποιώντας ένα δικό μας παράδειγμα που αυτό με την σειρά του επαληθεύεται μέσω της συνάρτησης **lyap** της Matlab.

Έχουμε:

```
function xx=uptrig(A,B,C)
% Αυτή η συνάρτηση λύνει την εξίσωση AX+XB=C
% έχοντας τους πίνακες A,B ως άνω τριγωνικούς
% πίνακες.
% Είσοδος :A(m,m),B(n,n),C(m,n) σε αυτήν την σειρά.
%-----ΚΑΤΩΤΟΙΚΙΔΗΣ ΔΗΜΗΤΡΙΟΣ Χ.-----
[m1,m2]=size(A);
[n1,n2]=size(B);
[m3,n3]=size(C);
```

```
% A,B πρέπει να είναι ακριβώς άνω τριγωνικοί πίνακες
% Αν αυτό δεν γίνει τότε ο τύπος αυτός δεν θα ισχυεί
% και θα έχουμε λάθος αποτέλεσμα
disp('A,B πρέπει να είναι ακριβώς άνω τριγωνικοί πίνακες')
disp(' Αν αυτό δεν γίνει τότε ο τύπος αυτός δεν θα ισχυεί και θα έχουμε λάθος
αποτέλεσμα')
if (m1~=m2) || (m2~=m3) || (n1~=n2) || (n2~=n3)
    disp('wrong dimensions,A is mxm,B is nxn,C is mxn')
else
    m=m1;
    n=n1;
    CC=C;
    P=0;
    PP=0;
    for i=m:-1:1
        for j=1:n
            P=0;
            PP=0;
            if j>1
                for k=1:j-1
                    P=P+X(i,k)*B(k,j);
                end
            end
            if i+1<=m
                for k=i+1:m
                    PP=PP+A(i,k)*X(k,j);
                end
            end
            CC(i,j)=CC(i,j)-P-PP;
            X(i,j)=CC(i,j)/(A(i,i)+B(j,j));
        end
    end
end
```

Αριθμητική επίλυση των εξισώσεων Sylvester – Η εξίσωση Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab

```
xx=X;  
end  
end
```

Αν τώρα εφαρμόσουμε στο παραπάνω πρόγραμμα τους εξής άνω τριγωνικούς πίνακες

$$A = \begin{pmatrix} 3 & 2 & 8 \\ 0 & 12 & 3 \\ 0 & 0 & 7 \end{pmatrix},$$

$$B = \begin{pmatrix} 9 & 17 & 2 \\ 0 & 3 & 8 \\ 0 & 0 & 6 \end{pmatrix}$$

και

$$C = \begin{pmatrix} 23 & 12 & 5 \\ 7 & 19 & 6 \\ 13 & 2 & 7 \end{pmatrix}$$

τότε σαν λύση της εξίσωσης Sylvester $AX + XB = C$ από το πρόγραμμα μας θα πάρουμε την λύση xx

$$xx = \begin{pmatrix} 1.33878968253968 & -0.63713293650794 & -0.09164401784772 \\ 0.21726190476190 & 1.25668650793651 & -0.43939831773165 \\ 0.81250000000000 & -1.18125000000000 & 1.14038461538462 \end{pmatrix},$$

ενώ η πραγματική λύση X μέσω της εξίσωσης **lyap** είναι η

$$X = \begin{pmatrix} 1.33878968253968 & -0.63713293650794 & -0.09164401784772 \\ 0.21726190476190 & 1.25668650793651 & -0.43939831773165 \\ 0.81250000000000 & -1.18125000000000 & 1.14038461538462 \end{pmatrix}.$$

Βλέπουμε ότι έχουμε ακρίβεια 14 δεκαδικών ψηφίων. Αναμέναμε τέτοια ακρίβεια καθώς και θα αναμένουμε ακρίβεια παρόμοιας περίπου τάξης και για δεδομένα μεγαλύτερης τάξης, καθώς είναι τέτοια η φύση του αλγορίθμου έτσι ώστε να έχουμε όσο το δυνατόν λιγότερα σφάλματα. Πάντως δεν θα θέλαμε να γενικεύσουμε το παραπάνω σχόλιο.

Εφαρμογή 2.8.2

Δημιούργησε έναν αλγόριθμο για την επίλυση της εξίσωσης Sylvester $XA + BX = C$, που να βασίζεται στην αναγωγή και των δυο πινάκων σε πραγματικές Schur μορφές.

Δώσε μια flop-μέτρηση αυτού του αλγορίθμου και σύγκρινε τον με τον Hessenberg-Schur αλγόριθμο.

Απάντηση

Έχουμε ήδη έναν αλγόριθμο για την επίλυση της εξίσωσης Sylvester $AX + XB = C$, που να βασίζεται στην αναγωγή και των δυο πινάκων σε πραγματικές Schur μορφές, τον αλγόριθμο των Bartels-Stewart. Παρόλα αυτά έχουμε δημιουργήσει και έναν ακόμα αλγόριθμο που να βασίζεται στην αναγωγή και των δυο πινάκων σε πραγματικές Schur μορφές και βασίζεται πιστά στον Hessenberg-Schur αλγόριθμο με την διαφορά εδώ βεβαίως ότι έχουμε αναγωγή των πινάκων A, B σε πραγματικές Schur μορφές, και όχι σε Hessenberg και Schur μορφές. Κατά τα άλλα ο αλγόριθμος αυτός είναι πανομοιότυπος με τον Hessenberg-Schur.

Θα έχουμε δηλαδή

Αλγόριθμος 2.8.1. *Ο Schur-Schur αλγόριθμος για την $AX + XB = C$*

Είσοδος: *Οι πίνακες A, B, C , διάστασης $p \times p, q \times q$ και $p \times q$ αντίστοιχα.*

Έξοδος: *Ο πίνακας X που ικανοποιεί την $AX + XB = C$.*

Βήμα 1. *Μετασχημάτισε τον B^T σε έναν πραγματικό πίνακα Schur RB , και τον A σε ένα άνω πίνακα Schur RA με τον εξής ορθογώνιο μετασχηματισμό :*

$$U^T B^T U = RB, \quad V^T A V = RA.$$

Αριθμητική επίλυση των εξίσωσεων Sylvester – Η εξίσωση Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab

Σχημάτισε τον $\hat{C} = V^T C U$, και διαμέρισε τον σε στήλες, δηλαδή $\hat{C} = (c_1, \dots, c_q)$.

Βήμα 2. Λύσε την $RA \cdot Y + Y \cdot RB^T = \hat{C}$:

For $k = q, \dots, 1$ do μέχρι οι στήλες του Y να υπολογιστούν

If $rb_{k,k-1} = 0$, τότε υπολόγισε την στήλη y_k λύνοντας το σύστημα:

$$(RA + rb_{kk}I)y_k = c_k - \sum_{j=k+1}^q rb_{kj}y_j$$

Else, υπολόγισε τις στήλες y_k και y_{k-1} λύνοντας το σύστημα:

$$\begin{pmatrix} RA + rb_{k-1,k-1}I & rb_{k-1k}I \\ rb_{k,k-1}I & RA + rb_{kk}I \end{pmatrix} \begin{pmatrix} y_{k-1} \\ y_k \end{pmatrix} = \begin{pmatrix} d_{k-1} \\ d_k \end{pmatrix},$$

όπου

$$(d_{k-1}, d_k) = (c_{k-1}, c_k) - \sum_{j=k+1}^q (rb_{k-1,j}y_j, rb_{kj}y_j).$$

Βήμα 3. Πάρε την λύση $X : X = V Y U^T$.

Παρακάτω κάναμε και μια συνάρτηση που να βασίζεται στον παραπάνω Schur-Schur αλγόριθμο. Η παρακάτω συνάρτηση χρησιμοποιεί για την επίλυση του γραμμικού συστήματος $Tx = d$ όπου χρειάζεται, μια άλλη συνάρτηση την οποία κατασκευάσαμε, την συνάρτηση **mygaussel**, η οποία μας βρίσκει το $x = mygaussel(T, d)$, και την οποία παρουσιάζουμε αναλυτικά στο παράρτημα.

Έτσι θα έχουμε

```
function xx=schrschr(A,B,C)
% Αυτή η συνάρτηση λύνει την εξίσωση AX+XB=C
% μετατρέποντας τους πίνακες A,B σε πραγματικούς
```

```
% πίνακες Schur.
% Είσοδος :A(p,p),B(q,q),C(p,q) σε αυτήν την σειρά.
% Η λύση είναι χχ
%-----ΚΑΤΩΤΟΙΚΙΔΗΣ ΔΗΜΗΤΡΙΟΣ Χ.-----
[p1,p2]=size(A);
[q1,q2]=size(B);
[p3,q3]=size(C);
if (p1~=p2) || (p2~=p3) || (q1~=q2) || (q2~=q3)
    disp('wrong dimensions,A is pxp,B is qxq,C is pxq')
else
    p=p1;
    q=q1;
[V,RA]=schur(A);
[U,RB]=schur(transpose(B));
CC=V'*C*U;
    I=eye(p);
if q==1
    F=(RA+RB(1,1)*I);
    Z=CC(:,1);
    Y=mygaussel(F,Z);
    for i=1:p
        YY(i,1)=Y(i);
    end
    xx=V*YY*U';
else
    for k=q:-1:2
        if RB(k,k-1)==0
            F=(RA+RB(k,k)*I);
            P=0;
            if k+1<=q
                for j=k+1:q
                    PP=RB(k,j)*YY(:,j);
                    P=P+PP;
                end
            end
        end
    end
end
```



```
end
end
Z=CC(:,1)-P;
Y=mygaussel(F,Z);
for i=1:p
    YY(i,1)=Y(i);
end
end
xx=V*YY*U';
end
end
```

Ο αλγόριθμος μας σίγουρα χρησιμοποιεί περισσότερα flops από τον Hessenberg-Schur αλγόριθμο αφού για ελάττωση σε πραγματική μορφή Schur και σε μορφή Hessenberg χρειάζεται:

$$\frac{10}{3}p^3 + 26q^3, \text{ flops}$$

ενώ για ελάττωση σε πραγματικές μορφές Schur χρειάζεται:

$$26p^3 + 26q^3 \text{ flops.}$$

Τώρα θα εφαρμόσουμε ένα δικό μας παράδειγμα στην συνάρτηση αυτή και τα αποτελέσματα μας θα τα συγκρίνουμε με τα αποτελέσματα της συνάρτησης **lyap** της Matlab.

Έστω οι πίνακες

$$A = \begin{pmatrix} 3 & 19 & 2 \\ 18 & 2 & 3 \\ 1 & 5 & 3 \end{pmatrix}, \quad B = \begin{pmatrix} 2 & 15 \\ 10 & 3 \end{pmatrix}$$

Αριθμητική επίλυση των εξισώσεων Sylvester – Η εξίσωση Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab

και

$$C = \begin{pmatrix} 10 & 2 \\ 12 & 5 \\ 3 & 19 \end{pmatrix}.$$

Τότε η υπολογισμένη λύση που θα πάρουμε από την συνάρτηση μας θα είναι η

$$xx = \begin{pmatrix} 1.00972237747314 & 0.97265082768502 \\ -0.42949942585385 & -0.95350584816120 \\ 1.69268446350361 & -0.43256475657220 \end{pmatrix},$$

ενώ η λύση X από την συνάρτηση **lyap** της Matlab θα είναι

$$X = \begin{pmatrix} 1.00972237747314 & 0.97265082768502 \\ -0.42949942585385 & -0.95350584816120 \\ 1.69268446350361 & -0.43256475657220 \end{pmatrix}.$$

Βλέπουμε ότι έχουμε ακρίβεια 14 δεκαδικών ψηφίων.

Εφαρμογή 2.8.3

Δημιούργησε έναν Hessenberg-Schur αλγόριθμο για την επίλυση της διακριτής εξίσωσης Sylvester $BXA + C = X$.

Απάντηση

Παρακάτω παρουσιάζουμε έναν Hessenberg-Schur αλγόριθμο που δημιουργήσαμε για την επίλυση της διακριτής εξίσωσης Sylvester $BXA + C = X$, καθώς και μια συνάρτηση που δημιουργήσαμε για την επίλυση της διακριτής εξίσωσης Sylvester $BXA + C = X$, που βασίζεται σε αυτόν τον αλγόριθμο.

Αλγόριθμος 2.8.2. Ένας Hessenberg-Schur αλγόριθμος για την $BXA + C = X$

Είσοδος: Οι πίνακες A, B, C , διάστασης $n \times n, m \times m$ και $m \times n$ αντίστοιχα, με $n \leq m$.

Έξοδος: Ο πίνακας X που ικανοποιεί την $BXA + C = X$.

Βήμα 1. Μετασχημάτισε τον A^T σε έναν πραγματικό πίνακα Schur R , και τον B σε ένα άνω πίνακα Hessenberg H με τον εξής ορθογώνιο μετασχηματισμό :

$$U^T A^T U = R, \quad V^T B V = H.$$

Σχημάτισε τον $\hat{C} = V^T C U$, και διαμέρισε τον σε στήλες, δηλαδή $\hat{C} = (c_1, \dots, c_n)$.

Βήμα 2. Λύσε την $HYR^T + \hat{C} = Y$

For $k = n, \dots, 1$ do μέχρι οι στήλες του Y να υπολογιστούν

If $r_{k,k-1} = 0$, τότε υπολόγισε την στήλη y_k λύνοντας το Hessenberg σύστημα:

$$(Hr_{kk} - I)y_k = -c_k - H \sum_{j=k+1}^n r_{kj} y_j$$

Else, υπολόγισε τις στήλες y_k και y_{k-1} λύνοντας το σύστημα:

$$\begin{pmatrix} Hr_{k-1,k-1} - I & r_{k-1k} H \\ r_{k,k-1} H & Hr_{kk} - I \end{pmatrix} \begin{pmatrix} y_{k-1} \\ y_k \end{pmatrix} = \begin{pmatrix} d_{k-1} \\ d_k \end{pmatrix},$$

όπου

$$(d_{k-1}, d_k) = (-c_{k-1}, -c_k) - \sum_{j=k+1}^n (Hr_{k-1,j} y_j, Hr_{kj} y_j).$$

Βήμα 3. Πάρε την λύση $X : X = V Y U^T$.

function X=discreteHessSch(B,A,C)

% Η συνάρτηση αυτή μας λύνει την διακριτή εξίσωση Sylvester με την Hessenberg-Schur

**Αριθμητική επίλυση των εξίσωσεων Sylvester – Η εξίσωση
Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab**

```

% μέθοδο. Η εξίσωση είναι η  $B \cdot X \cdot A + C = X$ .
% Ο πίνακας A είναι ένας  $n \times n$  πίνακας.
% Ο πίνακας B είναι ένας  $m \times m$  πίνακας.
% Ο πίνακας C είναι ένας  $m \times n$  πίνακας.
% Είσοδος: B,A,C σε αυτή την σειρά.
% Ο μεγαλύτερος από τους πίνακες A,B διαλέγεται να μετασχηματιστεί σε
% Hessenberg μορφή, ενώ ο άλλος σε Schur μορφή.
% Η λύση είναι η X
%-----ΚΑΤΩΤΟΙΚΙΔΗΣ ΔΗΜΗΤΡΙΟΣ X.-----

[n1,n2]=size(A);
[m1,m2]=size(B);
[m3,n3]=size(C);
if (n1~=n2) || (n2~=n3) || (m1~=m2) || (m2~=m3)
    disp('wrong dimensions,A is nxn,B is mxm,C is mxn')
else
    m=m1;
    n=n1;
    if m>=n
        [U,R]=schur(transpose(A));
        [V,H]=hess(B);
        CC=transpose(V)*C*U;
        I=eye(m);
        if n==1
            F=(H*R(1,1)-I);
            Z=-CC(:,1);
            Y=mygaussel(F,Z);
            for i=1:m
                YY(i,1)=Y(i);
            end
            X=V*YY*U';
        else
            for k=n:-1:2
                if R(k,k-1)==0

```

```

F=(H*R(k,k)-I);
P=0;
if k+1<=n
    for j=k+1:n
        PP=R(k,j)*H*YY(:,j);
        P=P+PP;
    end
end
Z=-CC(:,k)-P;
Y=mygaussel(F,Z);
for i=1:m
    YY(i,k)=Y(i);
end
else
FF=[H*R(k-1,k-1)-I R(k-1,k)*H;R(k,k-1)*H H*R(k,k)-I];
PPP=0;
if k+1<=n
    for j=k+1:n
        PPPP=[H*R(k-1,j)*YY(:,j);H*R(k,j)*YY(:,j)];
        PPP=PPP+PPPP;
    end
end
ZZ=[-CC(:,k-1);-CC(:,k)]-PPP;
G=mygaussel(FF,ZZ);
for i=1:m
    YY(i,k-1)=G(i);
end
for i=m+1:2*m
    YY(i-m,k)=G(i);
end
end
end
if R(2,1)==0

```

**Αριθμητική επίλυση των εξισώσεων Sylvester – Η εξίσωση
Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab**

```

F=(H*R(1,1)-I);
P=0;
if 2<=n
    for j=2:n
        PP=R(1,j)*H*YY(:,j);
        P=P+PP;
    end
end
Z=-CC(:,1)-P;
Y=mygaussel(F,Z);
for i=1:m
    YY(i,1)=Y(i);
end
end
X=V*YY*U';
end
else
    [U,R]=schur(B);
[V,H]=hess(transpose(A));
CC=transpose(V)*C*U;
I=eye(n);
if m==1
    F=(H*R(1,1)-I);
    Z=-CC(:,1);
    Y=mygaussel(F,Z);
    for i=1:n
        YY(i,1)=Y(i);
    end
    Xtran=V*YY*U';
    X=Xtran';
else
    for k=m:-1:2
        if R(k,k-1)==0

```

```

F=(H*R(k,k)-I);
P=0;
if k+1<=m
    for j=k+1:m
        PP=R(k,j)*H*YY(:,j);
        P=P+PP;
    end
end
Z=-CC(:,k)-P;
Y=mygaussel(F,Z);
for i=1:n
    YY(i,k)=Y(i);
end
else
FF=[H*R(k-1,k-1)-I R(k-1,k)*H;R(k,k-1)*H H*R(k,k)-I];
PPP=0;
if k+1<=m
    for j=k+1:m
        PPPP=[H*R(k-1,j)*YY(:,j);H*R(k,j)*YY(:,j)];
        PPP=PPP+PPPP;
    end
end
ZZ=[-CC(:,k-1);-CC(:,k)]-PPP;
G=mygaussel(FF,ZZ);
for i=1:n
    YY(i,k-1)=G(i);
end
for i=n+1:2*n
    YY(i-n,k)=G(i);
end
end
end
if R(2,1)==0

```

**Αριθμητική επίλυση των εξίσωσεων Sylvester – Η εξίσωση
Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab**

```
F=(H*R(1,1)-I);
P=0;
if 2<=m
    for j=2:m
        PP=R(1,j)*H*YY(:,j);
        P=P+PP;
    end
end
Z=-CC(:,1)-P;
Y=mygaussel(F,Z);
for i=1:n
    YY(i,1)=Y(i);
end
end
Xtran=V*YY*U';
X=Xtran';
end
end
end
```

Τώρα θα εφαρμόσουμε ένα δικό μας παράδειγμα στην συνάρτηση αυτή και τα αποτελέσματά μας θα τα συγκρίνουμε κάνοντας επαλήθευση των πράξεων καθώς δεν υπάρχει συνάρτηση της Matlab που να υπολογίζει την λύση της διακριτής εξίσωσης Sylvester.

Ας έχουμε τους πίνακες

$$A = \begin{pmatrix} 12 & 10 \\ 8 & 11 \end{pmatrix},$$

Δημήτριος Χ. Κατωτοκίδης

$$B = \begin{pmatrix} 3 & 8 & 12 \\ 2 & 1 & 4 \\ 8 & 1 & 6 \end{pmatrix} \quad \text{και} \quad C = \begin{pmatrix} 11 & 19 \\ 13 & 2 \\ 8 & 7 \end{pmatrix}.$$

Τότε ο αλγόριθμος μας θα μας δώσει την υπολογισμένη λύση X

$$X = \begin{pmatrix} 0.51518494556661 & -0.49651943926524 \\ 1.58130371006622 & -1.61191893226432 \\ -1.11787180285350 & 0.99183529746873 \end{pmatrix}.$$

Σαν επαλήθευση των αποτελεσμάτων για το παραπάνω X στην εξίσωση μας $BXA + C = X$ θα πάρουμε

$$X - BXA = \begin{pmatrix} 11.00000000000000 & 19.00000000000001 \\ 12.99999999999999 & 2.00000000000000 \\ 8.00000000000000 & 7.00000000000001 \end{pmatrix} \approx C$$

Έχουμε καλή ακρίβεια.

3. Η ΕΞΙΣΩΣΗ SYLVESTER – ΠΑΡΑΤΗΡΗΤΗΣ ΣΑΝ ΕΚΤΙΜΗΤΗΣ ΚΑΤΑΣΤΑΣΗΣ ΚΑΙ ΤΟ ΠΡΟΒΛΗΜΑ ΑΝΑΤΟΠΟΘΕΤΗΣΗΣ ΤΩΝ ΙΔΙΟΤΙΜΩΝ

3.1 Εισαγωγή

Σε όλα τα προβλήματα ανάδρασης-κατάστασης όπως, η ευστάθεια μέσω ανάδρασης κατάστασης, ανατοποθέτησης ιδιοτιμών, το LQR πρόβλημα καθώς και τα προβλήματα ανάδρασης κατάστασης H_∞ - ελέγχου αλλά και σε άλλα προβλήματα είναι απαραίτητο να διαθέτουμε με σαφήνεια το διάνυσμα κατάστασης $x(t)$. Παρόλα αυτά στις πιο πολλές περιπτώσεις οι καταστάσεις δεν είναι πλήρως γνωστές και συνήθως γνωρίζουμε μονάχα την είσοδο $u(t)$ και την έξοδο $y(t)$. Οι μη-διαθέσιμες καταστάσεις θα πρέπει κάπως να υπολογιστούν με ακρίβεια χρησιμοποιώντας τους πίνακες A, B και C καθώς επίσης και την είσοδο $u(t)$ και την έξοδο $y(t)$.

Στο κεφάλαιο αυτό θα ασχοληθούμε κυρίως με το πώς μπορούμε να εκτιμήσουμε τις καταστάσεις ενός συστήματος συνεχούς χρόνου με την εξίσωση Sylvester-παρατηρητή. Έτσι θα ασχοληθούμε μόνο με την περίπτωση του συνεχούς χρόνου μιας και αυτά τα οποία θα παρουσιάσουμε εδώ ισχύουν και για την περίπτωση διακριτού χρόνου, ίσως βέβαια με μερικές αλλαγές.

Θα παρουσιάσουμε την διαδικασία εκτίμησης κατάστασης μέσω της εξίσωσης Sylvester-παρατηρητή και θα παρουσιάσουμε σε γενικό πλαίσιο την αντίστοιχη διαδικασία εκτίμησης κατάστασης μέσω των ιδιοτιμών, για σύγκριση και για καλύτερη κατανόηση της διαδικασίας εκτίμησης κατάστασης μέσω της εξίσωσης Sylvester-παρατηρητή.

Τέλος θα δούμε πως εμπλέκεται η εξίσωση Sylvester στο πρόβλημα ανατοποθέτησης των ιδιοτιμών.

Μπορούμε για την αριθμητική επίλυση της εξίσωσης Sylvester-παρατηρητή να χρησιμοποιήσουμε την Hessenberg-Schur μέθοδο για την εξίσωση Sylvester.

3.2 Εκτίμηση κατάστασης μέσω των ιδιοτιμών

Εδώ σε αυτό το κομμάτι του κεφαλαίου παρουσιάζουμε εν συντομία και μονάχα για ενημερωτικούς σκοπούς την εκτίμηση κατάστασης μέσω των ιδιοτιμών.

Ας υποθέσουμε ότι έχουμε το γραμμικό και συνεχούς χρόνου αναλλοίωτο σύστημα

$$\begin{aligned}\dot{x}(t) &= Ax(t) + Bu(t), \\ y(t) &= Cx(t),\end{aligned}\tag{3.1}$$

όπου $A \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{n \times m}$, και $C \in \mathbb{R}^{r \times n}$.

Ας είναι $\hat{x}(t)$ ένας εκτιμητής του διανύσματος κατάστασης $x(t)$. Προφανώς θα θέλαμε να κατασκευάσουμε ένα διάνυσμα $\hat{x}(t)$ τέτοιο ώστε το σφάλμα $e(t) = x(t) - \hat{x}(t)$ να πλησιάζει το μηδέν όσο το δυνατόν γρηγορότερα, για όλες τις αρχικές καταστάσεις και για κάθε είσοδο $u(t)$. Ας υποθέσουμε ότι σχεδιάζουμε ένα δυναμικό σύστημα χρησιμοποιώντας όπως είπαμε τα στοιχεία τα οποία διαθέτουμε, δηλαδή την είσοδο $u(t)$ και την έξοδο $y(t)$ καθώς επίσης και τους πίνακες A, B και C που ικανοποιούν

$$\dot{\hat{x}}(t) = (A - KC)\hat{x}(t) + Ky(t) + Bu(t),$$

όπου εδώ αναζητούμε το $K \in \mathbb{R}^{n \times r}$. Τότε,

Αριθμητική επίλυση των εξισώσεων Sylvester – Η εξίσωση Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab

$$\begin{aligned}\dot{e}(t) &= \dot{x}(t) - \dot{\hat{x}}(t) = Ax(t) + Bu(t) - A\hat{x}(t) + KC\hat{x}(t) - Ky(t) - Bu(t), \\ &= (A - KC)x(t) - (A - KC)\hat{x}(t) = (A - KC)e(t).\end{aligned}$$

Η λύση του συστήματος αυτού των διαφορικών εξισώσεων είναι $e(t) = e^{(A-KC)t}e(0)$, η οποία μας δείχνει ότι ο ρυθμός κατά τον οποίο τα στοιχεία του διανύσματος σφάλματος $e(t)$ φτάνουν στο μηδέν μπορεί να ελεγχθεί από τις ιδιοτιμές του πίνακα $A - KC$. Εάν για παράδειγμα οι ιδιοτιμές του $A - KC$ έχουν αρνητικά πραγματικά μέρη μικρότερα από έναν αριθμό $-a$, τότε το σφάλμα $e(t)$ θα φτάσει το μηδέν γρηγορότερα από το $e^{-at}e(0)$.

Από τα παραπάνω βλέπουμε ότι το πρόβλημα της εκτίμησης κατάστασης μπορεί να λυθεί εάν βρεθεί ένας τέτοιος πίνακας K έτσι ώστε ο πίνακας $A - KC$ να έχει τις κατάλληλες επιθυμητές ιδιοτιμές.

Παραθέτουμε ότι εάν το ζευγάρι (A, C) είναι παρατηρήσιμο τότε μπορούμε να βρούμε πάντοτε ένα τέτοιο K , αφού η παρατηρησιμότητα του (A, C) συνεπάγεται την ελεγχσιμότητα του (A^T, C^T) . Μάλιστα στην συγκεκριμένη περίπτωση αν (A^T, C^T) είναι ελέγξιμο τότε υπάρχει πίνακας $L \in \mathbb{R}^{r \times n}$ τέτοιος ώστε ο πίνακας $(A^T + C^T L)$ να έχει τις κατάλληλες ιδιοτιμές. Έτσι αν $K = -L^T$ τότε οι ιδιοτιμές του πίνακα $A^T - C^T K^T$ θα είναι οι ίδιες με τον πίνακα $A - KC$.

Θεώρημα 3.2.1

Εάν (A, C) είναι παρατηρήσιμο, τότε οι καταστάσεις $x(t)$ του συστήματος

$$\begin{aligned}\dot{x}(t) &= Ax(t) + Bu(t), \\ y(t) &= Cx(t),\end{aligned}$$

όπου $A \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{n \times m}$, και $C \in \mathbb{R}^{r \times n}$, μπορούν να εκτιμηθούν από το

$$\dot{\hat{x}}(t) = (A - KC)\hat{x}(t) + Ky(t) + Bu(t), \quad (3.2)$$

όπου το K έχει κατασκευαστεί τέτοιο ώστε ο πίνακας $A - KC$ είναι ένας ευσταθής πίνακας. Ένας πίνακας λέγεται ευσταθής εάν όλες οι ιδιοτιμές του πίνακα αυτού έχουν αρνητικά πραγματικά μέρη.

Για το σφάλμα $e(t) = x(t) - \hat{x}(t)$ θα έχουμε

$$\dot{e}(t) = (A - KC)e(t)$$

και $e(t) \rightarrow 0$ καθώς $t \rightarrow \infty$.

3.3 Εκτίμηση κατάστασης μέσω της εξίσωσης Sylvester

Και τώρα θα δούμε πως μπορεί να γίνει εκτίμηση κατάστασης μέσω της εξίσωσης Sylvester. Ξέροντας λοιπόν την είσοδο $u(t)$ και την έξοδο $y(t)$ καθώς επίσης και τους πίνακες A, B και C , κατασκευάζουμε το σύστημα

$$\dot{z}(t) = Fz(t) + Gy(t) + Pu(t), \quad (3.3)$$

όπου F, G , και P είναι αντίστοιχα πίνακες διάστασης $n \times n$, $n \times r$, και $n \times m$, τέτοιοι ώστε να υπάρχει σταθερός nonsingular πίνακας X διάστασης $n \times n$, έτσι ώστε το διάνυσμα σφάλματος να τείνει στο μηδέν δηλαδή

$$e(t) = z(t) - X \cdot x(t) \rightarrow 0$$

για κάθε αρχική κατάσταση $x(0), z(0)$, και για κάθε είσοδο $u(t)$. Έτσι αν ισχύουν τα παραπάνω θα λέμε ότι το διάνυσμα $z(t)$ θα είναι εκτιμητής του $X \cdot x(t)$. Το σύστημα

Αριθμητική επίλυση των εξίσωσεων Sylvester – Η εξίσωση Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab

(3.3) θα λέγεται τότε ότι είναι παρατηρητής κατάστασης του συστήματος (3.1). Η ιδέα αυτή ήταν του D.Luenberger (1964) και για αυτό τον λόγο το σύστημα (3.3) είναι γνωστό στην θεωρία έλεγχου ως παρατηρητής Luenberger.

Το επόμενο θεώρημα παίζει βασικό ρόλο στην δημιουργία παρατηρητών κατάστασης. Μας λέει ότι οι πίνακες X, F, G και P πρέπει να ικανοποιούν ορισμένες προϋποθέσεις έτσι ώστε το σύστημα (3.3) να είναι παρατηρητής κατάστασης του συστήματος (3.1).

Θεώρημα 3.3.1 Το σύστημα (3.3) θα λέγεται ότι είναι παρατηρητής κατάστασης του συστήματος (3.1), δηλαδή, το $z(t)$ θα είναι εκτιμητής του $X \cdot x(t)$ με την έννοια ότι το σφάλμα $e(t) = z(t) - X \cdot x(t) \rightarrow 0$ καθώς το $t \rightarrow \infty$ για κάθε αρχική κατάσταση $x(0), z(0)$, και για κάθε είσοδο $u(t)$ εάν

- (i) $XA - FX = GC$,
- (ii) $P = XB$,
- (iii) F είναι ευσταθής πίνακας.

Απόδειξη Θα πρέπει να δείξουμε ότι εάν οι συνθήκες (i)-(iii) ικανοποιούνται, τότε $e(t) \rightarrow 0$ καθώς το $t \rightarrow \infty$.

Από την $e(t) = z(t) - X \cdot x(t)$, έχουμε

$$\begin{aligned}\dot{e}(t) &= \dot{z}(t) - X \cdot \dot{x}(t) \\ &= Fz(t) + Gy(t) + Pu(t) - X(Ax(t) + Bu(t)).\end{aligned}$$

Αν στην παραπάνω σχέση αντικαταστήσουμε το $y(t)$ με $Cx(t)$ και προσθαιρέσουμε το $F \cdot X \cdot x(t)$ θα έχουμε

$$\dot{e}(t) = Fe(t) + (FX - XA + GC)x(t) + (P - XB)u(t)$$

από την παραπάνω σχέση βλέπουμε ότι αν ισχύουν οι συνθήκες (i) και (ii) τότε έχουμε

$$\dot{e}(t) = Fe(t).$$

Αν τώρα ισχύει και η συνθήκη (iii) τότε όντως το $e(t) \rightarrow 0$ καθώς $t \rightarrow \infty$ για κάθε αρχική κατάσταση $x(0), z(0)$, και για κάθε είσοδο $u(t)$.

Άρα το $z(t)$ θα είναι εκτιμητής του $X \cdot x(t)$. ▲

Από τις εξισώσεις (3.2) και (3.3) βλέπουμε ότι το σύστημα (3.2) θα είναι παρόμοιο με το σύστημα (3.3) αν-ν

$$F = A - KC, \quad G = K, \quad P = B.$$

Η εξίσωση (3.2) μπορεί να γραφτεί στην μορφή

$$\begin{aligned} \dot{\hat{x}}(t) &= A\hat{x}(t) + Bu(t) + K(y(t) - C\hat{x}(t)) \\ &= A\hat{x}(t) + Bu(t) + K(y(t) - \hat{y}(t)) \end{aligned}$$

που μας δείχνει ότι ο παρατηρητής είναι ένα μοντέλο του συστήματος (3.1) με έναν ακόμα όρο ανάλογο του σφάλματος $y(t) - \hat{y}(t)$ όπου $\hat{y}(t) = C\hat{x}(t)$ είναι η ανακατασκευασμένη έξοδος.

Η εξίσωση Sylvester παρατηρητής

Ορισμός 3.3.1 Η εξίσωση πινάκων

$$XA - FX = GC$$

όπου έχουμε ως γνωστούς τους πίνακες A και C και ψάχνουμε τους πίνακες X, F και G θα λέγεται εξίσωση **Sylvester παρατηρητής**.

Αριθμητική επίλυση των εξισώσεων Sylvester – Η εξίσωση Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab

Δικαίως η παραπάνω εξίσωση ονομάζεται *εξίσωση Sylvester παρατηρητής* μιας και η εξίσωση αυτή παίζει μεγάλο ρόλο για την κατασκευή ενός παρατηρητή, ενώ μάλιστα αποτελεί παραλλαγή της κλασσικής εξίσωσης Sylvester

$$AX + XB = C$$

όπου είναι γνωστοί οι πίνακες A, B και C ενώ αναζητούμε τον άγνωστο πίνακα X .

Έτσι θα έχουμε τον παρακάτω αλγόριθμο:

Αλγόριθμος 3.3.1 Σχεδιασμός παρατηρητή πλήρους τάξης μέσω της εξίσωσης Sylvester παρατηρητή.

Είσοδοι. Οι πίνακες του συστήματος A, B και C διάστασης $n \times n$, $n \times m$, και $r \times n$, αντίστοιχα.

Έξοδος. Μια εκτίμηση $\hat{x}(t)$ του διανύσματος κατάστασης $x(t)$.

Υποθέσεις. Το ζευγάρι (A, C) είναι παρατηρήσιμο.

Βήμα 1. Βρες μια nonsingular λύση X της εξίσωσης Sylvester παρατηρητή

$$XA - FX = GC$$

διαλέγοντας τον F να είναι ευσταθής πίνακας και διαλέγοντας τον G με τέτοιο τρόπο έτσι ώστε η λύση X να είναι nonsingular.

Βήμα 2. Υπολόγισε τον πίνακα $P = XB$.

Βήμα 3. Δημιούργησε τον παρατηρητή $z(t)$ λύνοντας το σύστημα διαφορικών εξισώσεων :

$$\dot{z}(t) = Fz(t) + Gy(t) + Pu(t), \quad z(0) = z_0.$$

Βήμα 4. Βρες έναν εκτιμητή $\hat{x}(t)$ του $x(t)$: $\hat{x}(t) = X^{-1}z(t)$.

Σε ενότητα που θα παρουσιάσουμε παρακάτω αποδεικνύεται ότι αν θέλουμε να έχουμε μια non-singular λύση X από την εξίσωση Sylvester παρατηρητή

$$XA - FX = GC$$

θα πρέπει να έχουμε τις εξής απαραίτητες συνθήκες : το ζευγάρι πινάκων (A, C) να είναι παρατηρήσιμο και το ζευγάρι πινάκων (F, G) να είναι ελέγξιμο. Έτσι θα λέγαμε πως θα πρέπει να επιλέξουμε τους πίνακες F, G έτσι ώστε το ζευγάρι πινάκων (F, G) να είναι ελέγξιμο.

Παράδειγμα 3.3.1 Ας έχουμε τους πίνακες

$$A = \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix}, \quad B = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \quad C = (1 \ 0).$$

Το ζευγάρι (A, C) είναι παρατηρήσιμο.

Βήμα 1. Διαλέγουμε $G = \begin{pmatrix} 1 \\ 3 \end{pmatrix}$, $F = \text{diag}(-1, -3)$.

Τότε η πραγματική λύση X (υπολογισμένη από την συνάρτηση **lyap** της Matlab) θα είναι

$$X = \begin{pmatrix} 0.6667 & -0.3333 \\ 0.8000 & -0.2000 \end{pmatrix}.$$

Ο πίνακας X είναι nonsingular.

**Αριθμητική επίλυση των εξίσωσεων Sylvester – Η εξίσωση
Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab**

Βήμα 2.

$$P = XB = \begin{pmatrix} 0.6667 \\ 0.8000 \end{pmatrix}.$$

Βήμα 3. Θα έχουμε

$$\dot{z}(t) = \begin{pmatrix} -1 & 0 \\ 0 & -3 \end{pmatrix} z(t) + \begin{pmatrix} 1 \\ 3 \end{pmatrix} y(t) + \begin{pmatrix} 0.6667 \\ 0.8000 \end{pmatrix} u(t), \quad z(0) = z_0$$

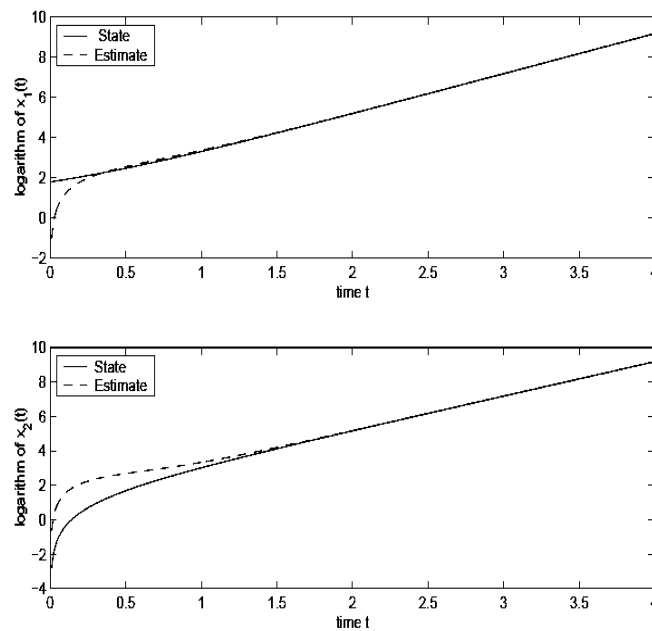
όπου

$$z(t) = \begin{pmatrix} z_1(t) \\ z_2(t) \end{pmatrix}$$

οπότε ένας εκτιμητής $\hat{x}(t)$ του $x(t)$ είναι

$$\hat{x}(t) = X^{-1}z(t) = \begin{pmatrix} -1.5 & 2.5 \\ -6 & 5 \end{pmatrix} \begin{pmatrix} z_1(t) \\ z_2(t) \end{pmatrix} = \begin{pmatrix} -1.5z_1(t) + 2.5z_2(t) \\ -6z_1(t) + 5z_2(t) \end{pmatrix}.$$

Σύγκριση της κατάστασης $x(t)$ και του εκτιμητή κατάστασης $\hat{x}(t)$ για το παράδειγμα 3.3.1 : Στον πίνακα 3.1 συγκρίνουμε το εκτιμώμενο $\hat{x}(t)$ που πήραμε από τον αλγόριθμο 3.3.1 με την κατάσταση $x(t)$ που βρέθηκε απευθείας λύνοντας το σύστημα (3.1) με $u(t)$ την μοναδιαία βηματική συνάρτηση και $x(0) = (6, 0)^T$. Η διαφορική συνάρτηση στο βήμα 3 λύθηκε με αρχικές συνθήκες $z(0) = 0$ από την εξίσωση **ode23** της Matlab. Η συνεχόμενη γραμμή αναφέρεται στην πραγματική κατάσταση $x(t)$ ενώ η διακεκομμένη γραμμή αναφέρεται στην εκτιμώμενη κατάσταση $\hat{x}(t)$.



Εικόνα 3.1: Οι πρώτες και οι δεύτερες μεταβλητές των καταστάσεων $x(t)$ και $\hat{x}(t)$ για το παράδειγμα 3.3.1

Αν τώρα προσπαθήσουμε να εφαρμόσουμε το δικό μας πρόγραμμα, την συνάρτηση μας **HessSch** (που δημιούργησε ο συγγραφέας αυτής της διπλωματικής) και που βασίζεται πιστά στον αλγόριθμο Hessenberg-Schur και η οποία συνάρτηση παρατίθεται λεπτομερώς στο παράρτημα, για να πάρουμε την υπολογιζόμενη λύση της εξίσωσης Sylvester-παρατηρητή στο βήμα 1, ας την ονομάσουμε, \bar{X} θα δούμε πως η λύση που μας δίνει το πρόγραμμα μας συμφωνεί απολύτως με την πραγματική λύση που μας δίνει η συνάρτηση **lyap** της Matlab και στην περίπτωση που έχουμε ακρίβεια 4 δεκαδικών ψηφίων αλλά και στην περίπτωση που έχουμε ακρίβεια 14 δεκαδικών ψηφίων.

Έτσι θα έχουμε

$$\bar{X} = X_{lyap} = \begin{pmatrix} 0.6667 & -0.3333 \\ 0.8000 & -0.2000 \end{pmatrix}$$

με ακρίβεια 4 δεκαδικών ψηφίων, και

Αριθμητική επίλυση των εξίσωσεων Sylvester – Η εξίσωση Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab

$$\bar{X} = X_{lyap} = \begin{pmatrix} 0.666666666666667 & -0.333333333333333 \\ 0.800000000000000 & -0.200000000000000 \end{pmatrix}$$

με ακρίβεια 14 δεκαδικών ψηφίων.

3.4 Εκτίμηση κατάστασης ελαττωμένης τάξης

Σε αυτήν την παράγραφο θα δείξουμε ότι αν ο πίνακας C διάστασης $r \times n$ έχει πλήρη τάξη r δηλαδή ισχύει $\text{rank } C = r$ τότε το πρόβλημα εύρεσης ενός εκτιμητή κατάστασης πλήρους τάξης n για το σύστημα

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t), \\ y(t) &= Cx(t), \end{aligned}$$

ανάγεται ή καλύτερα μπορεί να αναχθεί στο πρόβλημα εύρεσης ενός εκτιμητή κατάστασης τάξης $(n-r)$. Ένας τέτοιος εκτιμητής είναι γνωστός ως *εκτιμητής κατάστασης ελαττωμένης τάξης*.

Συγκεκριμένα έχουμε r στον αριθμό μεταβλητές εξόδου $y(t)$ που είναι διαθέσιμες για μέτρηση αν $\text{rank } C = r$ και χρειαζόμαστε και έναν εκτιμητή κατάστασης ελαττωμένης τάξης $(n-r)$ έτσι ώστε να εκτιμήσουμε τις υπόλοιπες $(n-r)$ στον αριθμό καταστάσεις $x(t)$ που δεν είναι διαθέσιμες από την εξίσωση $y(t) = Cx(t)$.

Έτσι μόλις κατασκευάσουμε έναν εκτιμητή κατάστασης τάξης $(n-r)$, και όχι n , οι πλήρεις καταστάσεις του συστήματος μπορούν να γίνουν γνωστές από τις $(n-r)$ μεταβλητές κατάστασης του παρατηρητή κατάστασης και τις r μεταβλητές που είναι διαθέσιμες για μετρήσεις.

Υπάρχουν, όπως και στην περίπτωση του πλήρους τάξεως εκτιμητή κατάστασης, δυο τρόποι για την εύρεση εκτιμητή κατάστασης ελαττωμένης τάξης. Έχουμε την εκτίμηση κατάστασης ελαττωμένης τάξης μέσω της ανατοποθέτησης των ιδιοτιμών, μια μέθοδος με την οποία δεν θα ασχοληθούμε εδώ, μονάχα γίνεται αναφορά στην ύπαρξη της, και ο άλλος τρόπος είναι με την επίλυση της Sylvester-παρατηρητή εξίσωσης που θα παρουσιάσουμε ευθύς αμέσως. Για χάριν απλότητας σε αυτήν την παράγραφο αντί για $x(t)$ και $\dot{x}(t)$ γράφουμε απλά x και \dot{x} αντίστοιχα καθώς και για τα άλλα διανύσματα.

Η διαδικασία έχει ως εξής

Αλγόριθμος 3.4.1 Σχεδιασμός παρατηρητή ελαττωμένης τάξης μέσω της Sylvester-παρατηρητή εξίσωσης.

Είσοδοι: Οι πίνακες A, B, C διάστασης αντίστοιχα $n \times n, n \times m, r \times n$.

Έξοδος: Μια εκτίμηση \hat{x} του διανύσματος κατάστασης x .

Υποθέσεις: (i) (A, C) είναι παρατηρήσιμο

(ii) ο πίνακας C έχει πλήρη τάξη

Βήμα 1: Διάλεξε έναν ευσταθή $(n-r) \times (n-r)$ πίνακα F .

Βήμα 2: Λύσε την ελαττωμένη εξίσωση Sylvester-παρατηρητή για να βρεθεί η λύση X , ένας πίνακας διάστασης $(n-r) \times n$, από την εξίσωση:

$$XA - FX = GC$$

διαλέγοντας κατάλληλα τον πίνακα G διάστασης $(n-r) \times n$.

Βήμα 3: Υπολόγισε τον πίνακα $P = XB$.

**Αριθμητική επίλυση των εξισώσεων Sylvester – Η εξίσωση
Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab**

Βήμα 4: Βρες τον $(n-r)$ ελαττωμένης-τάξης παρατηρητή z λύνοντας το σύστημα διαφορικών εξισώσεων :

$$\dot{z} = Fz + Gy + Pu, \quad z(0) = z_0$$

Βήμα 5: Βρες έναν εκτιμητή \hat{x} του x :

$$\hat{x} = \begin{pmatrix} C \\ X \end{pmatrix}^{-1} \begin{pmatrix} y \\ z \end{pmatrix}$$

Θα θέλαμε να πούμε πως η εξίσωση

$$\hat{x} = \begin{pmatrix} C \\ X \end{pmatrix}^{-1} \begin{pmatrix} y \\ z \end{pmatrix}$$

μπορεί να γραφτεί στην μορφή

$$\hat{x} = \bar{S}_1 y + \bar{S}_2 z$$

αν

$$\begin{pmatrix} C \\ X \end{pmatrix}^{-1} = (\bar{S}_1, \bar{S}_2).$$

Επίσης θα θέλαμε να ξαναπούμε ότι σε ενότητα που θα παρουσιάσουμε παρακάτω αποδεικνύεται ότι αν θέλουμε να έχουμε μια non-singular λύση X από την εξίσωση Sylvester παρατηρητή

$$XA - FX = GC$$

θα πρέπει να έχουμε τις εξής απαραίτητες συνθήκες : το ζευγάρι πινάκων (A, C) να είναι παρατηρήσιμο και το ζευγάρι πινάκων (F, G) να είναι ελέγξιμο. Έτσι θα λέγαμε

πως θα πρέπει να επιλέξουμε τους πίνακες F, G έτσι ώστε το ζευγάρι πινάκων (F, G) να είναι ελέγξιμο.

Παράδειγμα 3.4.1 Ας πάρουμε το πρόβλημα σχεδιασμού ενός παρατηρητή ελαττωμένης τάξης για το γραμμικό μοντέλο γνωστό και ως πρόβλημα ελικοπτέρου, που έχει τα εξής στοιχεία :

$$A = \begin{pmatrix} -0.02 & 0.005 & 2.4 & -32 \\ -0.14 & 0.44 & -1.3 & -30 \\ 0 & 0.018 & -1.6 & 1.2 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

$$B = \begin{pmatrix} 0.14 & -0.12 \\ 0.36 & -8.6 \\ 0.35 & 0.009 \\ 0 & 0 \end{pmatrix}$$

$$C = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 57.3 \end{pmatrix}$$

όπου έχουμε εδώ $\text{rank}(C) = 2, r = 2$.

Βήμα 1: Διαλέγουμε $F = \begin{pmatrix} -1 & 0 \\ 0 & -2 \end{pmatrix}$.

Βήμα 2: Διαλέγουμε $G = \begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix}$.

Η λύση X της εξίσωσης Sylvester-παρατηρητή εξίσωσης $XA - FX = GC$ είναι

$$X = \begin{pmatrix} -0.0117 & -0.0822 & 62.1322 & 37.2007 \\ -0.1364 & -1.9296 & 428.2711 & -173.4895 \end{pmatrix}.$$

Βήμα 3: $P = XB = \begin{pmatrix} 21.7151 & 1.2672 \\ 149.1811 & 20.4653 \end{pmatrix}$

**Αριθμητική επίλυση των εξίσωσεων Sylvester – Η εξίσωση
Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab**

Βήμα 4: θα έχουμε τον ελαττωμένης τάξης παρατηρητή

$$\dot{z}(t) = \begin{pmatrix} -1 & 0 \\ 0 & -2 \end{pmatrix} z(t) + \begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix} y(t) + \begin{pmatrix} 21.7151 & 1.2672 \\ 149.1811 & 20.4653 \end{pmatrix} u(t)$$

Ένας εκτιμητής \hat{x} του x είναι

$$\hat{x} = \begin{pmatrix} C \\ X \end{pmatrix}^{-1} \begin{pmatrix} y \\ z \end{pmatrix} = \begin{pmatrix} -24.5513 & -135.1240 & 124.1400 & -18.0098 \\ 1 & 0 & 0 & 0 \\ -0.0033 & -0.0360 & 0.0395 & -0.0034 \\ 0 & 0.0175 & 0 & 0 \end{pmatrix} \begin{pmatrix} y \\ z \end{pmatrix},$$

όπου αν

$$\hat{x} = (\hat{x}_1 \quad \hat{x}_2 \quad \hat{x}_3 \quad \hat{x}_4)^T \quad \text{και} \quad y = \begin{pmatrix} y_1 \\ y_2 \end{pmatrix}$$

τότε θα έχουμε

$$\hat{x} = (\hat{x}_1 \quad y_1 \quad \hat{x}_3 \quad 0.0175y_2)^T.$$

Σύγκριση της κατάστασης $x(t)$ και του εκτιμητή κατάστασης $\hat{x}(t)$ για το παράδειγμα

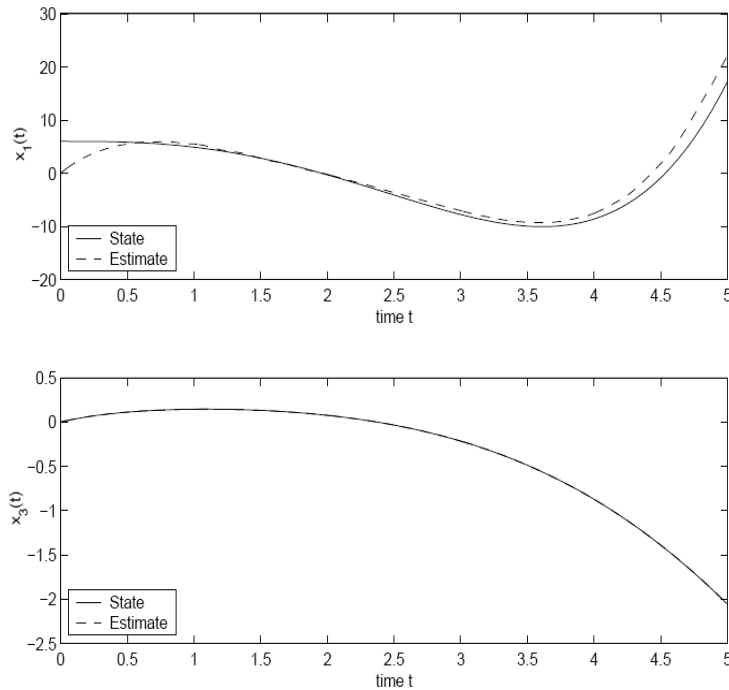
3.4.1 : Στον πίνακα 3.2 συγκρίνουμε το εκτιμώμενο $\hat{x}(t)$ που πήραμε από τον αλγόριθμο

3.4.1 με την κατάσταση $x(t)$ που βρέθηκε απευθείας λύνοντας το σύστημα (3.1) με

$u(t) = H(t) \begin{pmatrix} 1 \\ 1 \end{pmatrix}$ όπου $H(t)$ η μοναδιαία βηματική συνάρτηση και $x(0) = (6, 0, 0, 0)^T$. Η

διαφορική συνάρτηση στο βήμα 3 λύθηκε με αρχικές συνθήκες $z(0) = 0$ από την εξίσωση

ode23 της *Matlab*. Η συνεχόμενη γραμμή αναφέρεται στην πραγματική κατάσταση $x(t)$ ενώ η διακεκομμένη γραμμή αναφέρεται στην εκτιμώμενη κατάσταση $\hat{x}(t)$.



Εικόνα 3.2 : Οι πρώτες και οι τρίτες μεταβλητές των καταστάσεων $x(t)$ και $\hat{x}(t)$ για το παράδειγμα 3.4.1

Αν τώρα προσπαθήσουμε να εφαρμόσουμε το δικό μας πρόγραμμα, την συνάρτηση μας **HessSch** (που δημιούργησε ο συγγραφέας αυτής της διπλωματικής) και που βασίζεται πιστά στον αλγόριθμο Hessenberg-Schur και η οποία συνάρτηση παρατίθεται λεπτομερώς στο παράρτημα, για να πάρουμε την υπολογιζόμενη λύση της εξίσωσης Sylvester-παρατηρητή στο βήμα 1, ας την ονομάσουμε, \bar{X} θα δούμε πως η λύση που μας δίνει το πρόγραμμα μας συμφωνεί σχεδόν απόλυτα με την πραγματική λύση που μας δίνει η συνάρτηση **lyap** της *Matlab* στην περίπτωση που έχουμε ακρίβεια 4 δεκαδικών ψηφίων αλλά στην περίπτωση που έχουμε ακρίβεια 12 δεκαδικών ψηφίων η λύση μας συμφωνεί με την πραγματική στα 11 δεκαδικά ψηφία σίγουρα ενώ συμφωνεί και στην ακρίβεια των 12 δεκαδικών ψηφίων σε πολλά από τα στοιχεία του πίνακα που εκπροσωπεί την λύση.

Έτσι θα έχουμε

Αριθμητική επίλυση των εξίσωσεων Sylvester – Η εξίσωση Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab

$$\bar{X} = X_{lyap} = \begin{pmatrix} -0.0117 & -0.0822 & 62.1322 & 37.2007 \\ -0.1364 & -1.9296 & 428.2711 & -173.4895 \end{pmatrix}$$

με ακρίβεια 4 δεκαδικών ψηφίων, και

$$\bar{X} = \begin{pmatrix} -0.011738221579 & -0.082167551053 & 62.132220256919 & 37.200686069572 \\ -0.136435587629 & -1.929589025034 & 428.271077723355 & -173.489451411576 \end{pmatrix}$$

$$X_{lyap} = \begin{pmatrix} -0.011738221579 & -0.082167551053 & 62.132220256919 & 37.200686069572 \\ -0.136435587629 & -1.929589025033 & 428.271077723357 & -173.489451411576 \end{pmatrix}$$

με ακρίβεια 12 δεκαδικών ψηφίων.

3.5 Non-singular λύσεις τις εξίσωσης Sylvester-παρατηρητή

Έχουμε δει ότι ο σχεδιασμός ενός παρατηρητή μέσω της εξίσωσης Sylvester-παρατηρητής χρειάζεται μια non-singular λύση X είτε είχαμε σχεδιασμό παρατηρητή πλήρους τάξης είτε είχαμε σχεδιασμό παρατηρητή ελαττωμένης τάξης. Σε αυτήν την παράγραφο θα περιγράψουμε μερικές απαραίτητες συνθήκες έτσι ώστε η μοναδική λύση της εξίσωσης Sylvester να έχει τέτοιες ιδιότητες. Για χάρη ευκολίας θα παρουσιάσουμε μονάχα την περίπτωση της πλήρους τάξης. Βεβαίως τα αποτελέσματα ισχύουν και για την περίπτωση ελαττωμένης τάξης.

Το παρακάτω θεώρημα έχει αποδειχθεί από τον Bhattacharyya και DeSousa (1981). Η απόδειξη αυτή έχει παρθεί από τον Chen (1984).

Θεώρημα 3.5.1. *Απαραίτητες συνθήκες για να έχουμε μια nonsingular λύση της εξίσωσης Sylvester-παρατηρητή. Ας είναι A, F, G και C , αντίστοιχα διάστασης $n \times n, n \times n, n \times r$, και $r \times n$. Ας είναι X μια μοναδική λύση της εξίσωσης Sylvester-παρατηρητή*

$$XA - FX = GC.$$

Τότε απαραίτητες συνθήκες για την λύση X να είναι non-singular είναι: το ζευγάρι πινάκων (A, C) να είναι παρατηρήσιμο και το ζευγάρι πινάκων (F, G) να είναι ελέγξιμο.

Απόδειξη : Από την εξίσωση $XA - FX = GC$, θα έχουμε

$$\begin{aligned} XA^0 - F^0 X &= 0, \quad (A^0 = I_{n \times n}, \quad F^0 = I_{n \times n}.) \\ XA - FX &= GC, \\ XA^2 - F^2 X &= GCA + FGC, \\ &\vdots \\ XA^n - F^n X &= GCA^{n-1} + FGCA^{n-2} + \dots + F^{n-1}GC. \end{aligned}$$

Ας είναι $a(\lambda) = \lambda^n + a_1 \lambda^{n-1} + \dots + a_n$ το χαρακτηριστικό πολυώνυμο του πίνακα A , και ας ορίσουμε τον πίνακα ελεγχιμότητας του ζευγαριού (F, G) ως C_{FG} , και τον πίνακα παρατηρησιμότητας του ζευγαριού (A, C) ως O_{AC} .

Πρώτα από όλα, τονίζουμε ότι η μοναδικότητα της λύσης X προϋποθέτει ότι ο πίνακας $a(F)$ είναι non-singular και το αντίστροφο. Αυτό ισχύει ως εξής : Ξέρουμε ότι η λύση X είναι μοναδική λύση αν και μόνο αν A, F δεν έχουν όμοιες ιδιοτιμές. A, F δεν έχουν όμοιες ιδιοτιμές εάν και μόνο εάν ο πίνακας $a(F)$ είναι non-singular επειδή οι ιδιοτιμές του πίνακα $a(F)$ είναι n αριθμοί

$$\prod_{j=1}^n (\mu_i - \lambda_j), i = 1, \dots, n$$

**Αριθμητική επίλυση των εξισώσεων Sylvester – Η εξίσωση
Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab**

όπου λ_j είναι οι ιδιοτιμές του πίνακα A και μ_i οι ιδιοτιμές του πίνακα F . Έτσι $a(F)$ είναι non-singular αν και μόνο αν X είναι μοναδική λύση της εξίσωσης $XA - FX = GC$.

Τώρα αν πολλαπλασιάσουμε τις παραπάνω εξισώσεις αντίστοιχα με $a_n, a_{n-1}, \dots, 1$, και χρησιμοποιήσουμε το θεώρημα των Cayley-Hamilton, μετά από αρκετές αλγεβρικές πράξεις θα έχουμε ότι :

$$X = -[a(F)]^{-1} C_{FG} R O_{AC}, \quad (3.4)$$

όπου

$$R = \begin{pmatrix} a_{n-1}I & a_{n-2}I & \dots & a_1I & I \\ a_{n-2}I & a_{n-3}I & \dots & I & 0 \\ \vdots & \vdots & & \vdots & \vdots \\ a_1I & I & \dots & 0 & 0 \\ I & 0 & \dots & 0 & 0 \end{pmatrix}.$$

Από την εξίσωση (3.4) έχουμε ότι για να είναι το X non-singular λύση οι πίνακες C_{FG} και O_{AC} πρέπει να έχουν πλήρη τάξη, δηλαδή το ζευγάρι πινάκων (A, C) να είναι παρατηρήσιμο και το ζευγάρι πινάκων (F, G) να είναι ελέγξιμο. ▲

Πόρισμα 3.5.1 Εάν G είναι πίνακας διάστασης $n \times 1$ και C πίνακας διάστασης $1 \times n$ τότε ικανές και αναγκαίες συνθήκες για να έχουμε μια nonsingular λύση της εξίσωσης Sylvester-παρατηρητή είναι το ζευγάρι πινάκων (A, C) να είναι παρατηρήσιμο και το ζευγάρι πινάκων (F, G) να είναι ελέγξιμο.

Απόδειξη. Σε αυτήν την περίπτωση οι πίνακες C_{FG} και O_{AC} θα είναι τετραγωνικοί πίνακες και από την εξίσωση (3.4) προκύπτει τότε ότι για να έχουμε μια nonsingular

λύση της εξίσωσης Sylvester-παρατηρητή θα πρέπει τότε το ζευγάρι πινάκων (A, C) να είναι παρατηρήσιμο και το ζευγάρι πινάκων (F, G) να είναι ελέγξιμο.

3.6 Συνδυασμός ανάδρασης κατάστασης και σχεδιασμού παρατηρητή

Όταν αντί για τις πραγματικές καταστάσεις x εμείς χρησιμοποιούμε τις εκτιμώμενες καταστάσεις \hat{x} τότε θα έχουμε τον νόμο ανάδρασης

$$u = s(t) - K \hat{x}(t) \quad (3.5)$$

και τότε αναρωτιόμαστε αν θα υπάρξει κάποιο πρόβλημα στην ανατοποθέτηση των ιδιοτιμών.

Γίνεται αναφορά μονάχα στην περίπτωση ελαττωμένης τάξης, ενώ βεβαίως ισχύουν τα ίδια αποτελέσματα και για την περίπτωση της πλήρους τάξης.

Από τις εξισώσεις (3.5) και (3.1) θα έχουμε

$$\begin{aligned} \dot{x} &= Ax + B(s - K\hat{x}) \\ &= Ax + B(s - K\bar{S}_1 y - K\bar{S}_2 z) \\ &= Ax + B(s - K\bar{S}_1 Cx - K\bar{S}_2 z) \\ &= (A - BK\bar{S}_1 C)x - BK\bar{S}_2 z + Bs \end{aligned}$$

αν θυμηθούμε πως το \hat{x} μπορεί να γραφτεί στην μορφή

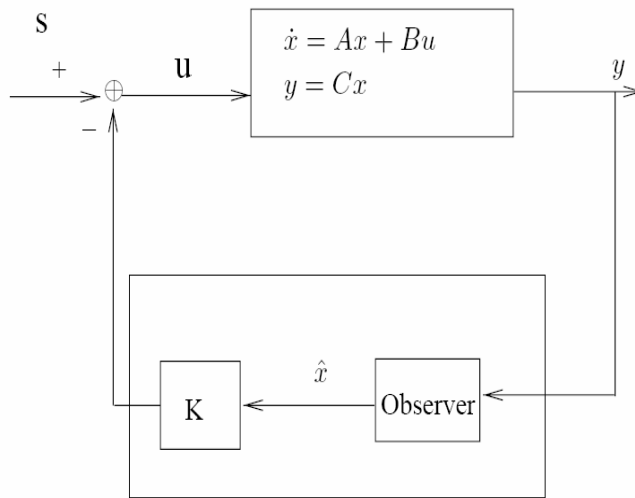
$$\hat{x} = \bar{S}_1 y + \bar{S}_2 z.$$

Θα έχουμε επίσης πως

Αριθμητική επίλυση των εξίσωσεων Sylvester – Η εξίσωση Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab

$$\begin{aligned} \dot{z} &= Fz + Gy + Pu \\ &= Fz + GCx + P(s - K\bar{S}_1 y - K\bar{S}_2 z) \\ &= (GC - PK\bar{S}_1 C)x + (F - PK\bar{S}_2)z + Ps \end{aligned}$$

Έτσι το συνδυασμένο σύστημα (εικόνα 3.3)



εικόνα 3.3 ανάδραση κατάστασης βασισμένη σε παρατηρητή

Θα δίνεται από το σύστημα

$$\begin{aligned} \begin{pmatrix} \dot{x} \\ \dot{z} \end{pmatrix} &= \begin{pmatrix} A - BK\bar{S}_1 C & -BK\bar{S}_2 \\ GC - PK\bar{S}_1 C & F - PK\bar{S}_2 \end{pmatrix} \begin{pmatrix} x \\ z \end{pmatrix} + \begin{pmatrix} B \\ P \end{pmatrix} s \\ y &= (C, 0) \begin{pmatrix} x \\ z \end{pmatrix}. \end{aligned}$$

Εφαρμόζοντας στο σύστημα αυτό τον μετασχηματισμό ομοιότητας, που δίνεται από τον non-singular πίνακα και ξέροντας επίσης ότι

$$e = z - Xx, \quad XA - FX = GC, \quad P = XB$$

και μετά από αρκετές αλγεβρικές πράξεις θα καταλήξουμε από το παραπάνω σύστημα στο σύστημα

$$\begin{pmatrix} \dot{x} \\ \dot{e} \end{pmatrix} = \begin{pmatrix} A - BK & -BK\bar{S}_2 \\ 0 & F \end{pmatrix} \begin{pmatrix} x \\ e \end{pmatrix} + \begin{pmatrix} B \\ 0 \end{pmatrix} s$$
$$y = (C, 0) \begin{pmatrix} x \\ e \end{pmatrix}.$$

Έτσι έχουμε πως οι ιδιοτιμές του συνδυασμένου συστήματος είναι η ένωση των ιδιοτιμών του πίνακα $A - BK$ και του πίνακα παρατηρητή F .

Έτσι συμπεραίνουμε ότι ο σχεδιασμός παρατηρητή και ανάδρασης μπορούν να γίνουν ανεξάρτητα ο ένας από τον άλλο και ο υπολογισμός του κέρδους ανάδρασης δεν επηρεάζεται εάν χρησιμοποιήσουμε είτε την πραγματική κατάσταση είτε την εκτιμημένη.

Αυτή η ιδιότητα είναι γνωστή ως ιδιότητα διαχωρισμού.

3.7 Αριθμητική επίλυση της περιορισμένης Sylvester-παρατηρητή εξίσωσης

Σε αυτόν τον τομέα θεωρούμε το πρόβλημα επίλυσης της περιορισμένης, ελαττωμένης τάξης, εξίσωσης Sylvester-παρατηρητή. Συγκεκριμένα έχουμε το ακόλουθο πρόβλημα:

Λύσε την ελαττωμένης τάξης εξίσωση Sylvester-παρατηρητή

$$XA - FX = GC$$

έτσι ώστε να ισχύει

$$XB = 0$$

καθώς επίσης και να ισχύει ο πίνακας

**Αριθμητική επίλυση των εξίσωσεων Sylvester – Η εξίσωση
Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab**

$$\begin{bmatrix} X \\ C \end{bmatrix}$$

να έχει πλήρη τάξη.

Η σημαντικότητα επίλυσης της ελαττωμένης εξίσωσης Sylvester κείται στο γεγονός ότι εάν ισχύει ο περιορισμός

$$XB = 0$$

εάν δηλαδή ο περιορισμός αυτός ικανοποιείται, τότε το σύστημα ανάδρασης με τον ελαττωμένης τάξης παρατηρητή έχει τις ίδιες εύρωστες ιδιότητες με αυτές που θα είχε το ευθύ σύστημα ανάδρασης (Tsui 1988).

Παρακάτω παρουσιάζουμε τον αλγόριθμο για το πώς θα επιλύσουμε μια τέτοιου είδους εξίσωση (Barlow *et al* 1992).

Αλγόριθμος 3.7.1 Ένας αλγόριθμος για την επίλυση της περιορισμένης ελαττωμένης τάξης εξίσωση Sylvester

Είσοδοι: οι πίνακες του συστήματος A, B και C διάστασης $n \times n, n \times m$, και $r \times n$, αντίστοιχα καθώς και ένας πίνακας F διάστασης $(n-r) \times (n-r)$.

Έξοδοι: Ένας πίνακας X διάστασης $(n-r) \times n$ και ένας πίνακας G διάστασης

$(n-r) \times r$, έτσι ώστε να ικανοποιείται η εξίσωση $XA - FX = GC$, ο πίνακας $\begin{bmatrix} X \\ C \end{bmatrix}$ να

έχει πλήρη τάξη, να είναι δηλαδή *nonsingular*, και τέλος να ισχύει $XB = 0$.

Υποθέσεις: το ζευγάρι (A, C) είναι παρατηρήσιμο, θα ισχύει $n > r > m$, και

$$\text{rank}(CB) = \text{rank}(B) = m.$$

Βήμα 1. Βρες την QR παραγοντοποίηση του B :

$$B = W \begin{pmatrix} S \\ 0 \end{pmatrix},$$

όπου S είναι διάστασης $m \times m$, άνω τριγωνικός και έχει πλήρη τάξη, και W είναι ορθογώνιος και διάστασης $n \times n$.

Χώρισε τον W σε $W = (W_1, W_2)$, όπου W_1 είναι διάστασης $n \times m$ και W_2 είναι διάστασης $n \times (n - m)$.

Βήμα 2. Θέσε

$$A_1 = W_2^T A W_1, \quad A_2 = W_2^T A W_2, \quad C_1 = C W_1, \quad C_2 = C W_2$$

Βήμα 3. Βρες μια QR παραγοντοποίηση του πίνακα C_1 :

$$C_1 = (Q_1, Q_2) \begin{pmatrix} R \\ 0 \end{pmatrix},$$

όπου Q_1 έχει διάσταση $r \times m$, Q_2 έχει διάσταση $r \times (r - m)$, και R είναι ένας πίνακας διάστασης $m \times m$, άνω τριγωνικός και έχει πλήρη τάξη.

Βήμα 4. Όρισε το E ως εξής:

$$E = \begin{pmatrix} E_1 \\ E_2 \end{pmatrix} = Q^T C_2$$

όπου E_1 είναι διάστασης $m \times (n - m)$, E_2 είναι διάστασης $(r - m) \times (n - m)$, και $Q = (Q_1, Q_2)$.

Αριθμητική επίλυση των εξισώσεων Sylvester – Η εξίσωση Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab

Βήμα 5. Σχημάτισε τον πίνακα $\hat{A} = A_2 - A_1 R^{-1} E_1$. Λύσε την εξίσωση Sylvester :

$$Z\hat{A} - FZ = G_2 E_2$$

διαλέγοντας τον πίνακα G_2 τυχαία. Ο πίνακας Z έχει διάσταση $(n-r) \times (n-m)$.

Βήμα 6. Θέσε

$$G_1 = Z A_1 R^{-1} = Z J, \quad G = (G_1, G_2) Q^T, \quad \text{και} \quad X = ZW_2^T.$$

Ο πίνακας $J = A_1 R^{-1}$ υπολογίζεται λύνοντας το άνω τριγωνικό σύστημα $JR = A_1$.

Υλοποίηση του αλγορίθμου αυτού και εφαρμογή αυτού στο παράδειγμα που ακολουθεί έχουμε στο παράρτημα στην συνάρτηση που έφτιαξε ο συγγραφέας της διπλωματικής αυτής με όνομα **constrainedSylvEq**.

Παράδειγμα 3.7.1 Ας λύσουμε την εξίσωση $XA - FX = GC$ χρησιμοποιώντας τον παραπάνω αλγόριθμο με

$$A = \begin{pmatrix} -0.02 & 0.005 & 2.4 & -3.2 \\ -0.14 & 0.44 & -1.3 & -3 \\ 0 & 0.018 & -1.6 & 1.2 \\ 0 & 0 & 1 & 0 \end{pmatrix}, \quad B = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \end{pmatrix},$$

$$C = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 57.3 \end{pmatrix}, \quad \text{και} \quad F = \begin{pmatrix} -1 & 0 \\ 1 & -2 \end{pmatrix}.$$

Τότε θα έχουμε $n = 4, r = 2, m = 1$.

Βήμα 1. $W_1 = (-0.5 \quad -0.5 \quad -0.5 \quad -0.5)^T$

$$W_2 = \begin{pmatrix} -0.5 & -0.5 & -0.5 \\ 0.8333 & -0.1667 & -0.1667 \\ -0.1667 & 0.8333 & -0.1667 \\ -0.1667 & -0.1667 & 0.8333 \end{pmatrix}, \quad S = -2$$

Βήμα 2.

$$A_1 = W_2^T A W_1 = \begin{pmatrix} 1.5144 \\ -0.2946 \\ -0.9856 \end{pmatrix},$$

$$A_2 = W_2^T A W_2 = \begin{pmatrix} 0.9015 & -1.6430 & 0.5596 \\ 0.1701 & -2.5976 & 2.9907 \\ -0.4185 & -0.2230 & 1.5604 \end{pmatrix},$$

$$C_1 = C W_1 = \begin{pmatrix} -0.5 \\ -28.65 \end{pmatrix}, \quad C_2 = C W_2 = \begin{pmatrix} 0.8333 & -0.1667 & -0.1667 \\ -9.55 & -9.55 & 47.75 \end{pmatrix}$$

Βήμα 3.

$$Q = \begin{pmatrix} -0.0174 & -0.9998 \\ -0.9998 & 0.0174 \end{pmatrix}, \quad R = 28.6544$$

Βήμα 4.

$$E = Q^T C_2 = \begin{pmatrix} 9.5340 & 9.5515 & -47.7398 \\ -0.9998 & 0 & 0.9998 \end{pmatrix}$$

$$E_2 = (-0.9998 \quad 0 \quad 0.9998)$$

Βήμα 5.

$$\hat{A} = A_2 - A_1 R^{-1} E_1 = \begin{pmatrix} 0.3976 & -2.1478 & 1.96 \\ -0.0721 & -2.4944 & 2.4999 \\ -0.0905 & 0.1056 & -0.0817 \end{pmatrix}$$

**Αριθμητική επίλυση των εξίσωσεων Sylvester – Η εξίσωση
Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab**

Διαλέγουμε

$$G_2 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}.$$

Η λύση Z της εξίσωσης Sylvester : $Z\hat{A} - FZ = G_2E_2$ είναι

$$Z = \begin{pmatrix} -0.6715 & 0.9589 & -0.0860 \\ -0.2603 & -0.5957 & 0.9979 \end{pmatrix}$$

Βήμα 6. $G_1 = ZJ = \begin{pmatrix} -0.0424 \\ -0.0420 \end{pmatrix}$

$$G = (G_1, G_2)Q^T = \begin{pmatrix} -0.9991 & 0.0598 \\ 0.0007 & 0.0419 \end{pmatrix}$$

$$X = \begin{pmatrix} -0.1007 & -0.7050 & 0.9254 & -0.1196 \\ -0.0709 & -0.2839 & -0.6199 & 0.9743 \end{pmatrix}$$

Έτσι επιβεβαιώνουμε ότι

(i) $\|XA - FX - GC\| = O(10^{-15})$

(ii) $XB = 10^{-16} \begin{pmatrix} 0.4163 \\ 0.4163 \end{pmatrix}$

(iii) $\text{rank} \begin{pmatrix} X \\ C \end{pmatrix} = 4$

Εάν είχαμε μεγαλύτερη ακρίβεια τα στοιχεία του πίνακα XB θα πλησίαζαν ακόμα περισσότερο το μηδέν.

Να πούμε επίσης πως αν διαλέξουμε ως G το

$$G_2 = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

τότε η λύση X και συνεπώς ο πίνακας $\begin{bmatrix} X \\ C \end{bmatrix}$ δεν θα έχουν πλήρη τάξη. Σε αυτήν την περίπτωση η λύση X θα έχει τάξη 1 και θα είναι η

$$X = \begin{pmatrix} -0.1006 & -0.7044 & 0.9246 & -0.1195 \\ -0.1006 & -0.7044 & 0.9246 & -0.1195 \end{pmatrix}.$$

3.8 Η εξίσωση Sylvester-παρατηρητής και το πρόβλημα ανατοποθέτησης των ιδιοτιμών.

Ας υποθέσουμε ότι το διάνυσμα κατάστασης $x(t)$ μας είναι γνωστό, ή έχει εκτιμηθεί χρησιμοποιώντας την μέθοδο Luenberger ή κάποια άλλη μέθοδο. Τότε μπορούμε να σταθεροποιήσουμε επιτυχώς ένα ασταθές σύστημα ως εξής :

Ας έχουμε $u = Fx(t)$. Τότε το σύστημα

$$\dot{x}(t) = Ax(t) + Bu(t)$$

γίνεται

$$\dot{x}(t) = (A + BF)x(t).$$

Το πρόβλημα είναι να διαλέξουμε τον πίνακα F έτσι ώστε το κλειστό σύστημα

$$\dot{x}(t) = (A + BF)x(t)$$

Αριθμητική επίλυση των εξίσωσεων Sylvester – Η εξίσωση Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab

να γίνει ασυμπτωτικά ευσταθές, δηλαδή να βρεθεί πίνακας F έτσι ώστε ο πίνακας $A + BF$ να έχει όλες τις ιδιοτιμές του με αρνητικά πραγματικά μέρη. Αυτό είναι ένα γνωστό πρόβλημα σταθεροποίησης ανάδρασης κατάστασης.

Σε μερικές περιπτώσεις μάλιστα δεν είναι απλά αρκετό να σταθεροποιήσουμε ένα σύστημα. Σε πολλά πρακτικά προβλήματα θα μας ήταν χρήσιμο να διαλέξουμε εμείς ποιες θα είναι η ιδιοτιμές του κλειστού συστήματος έτσι ώστε το κλειστό σύστημα μας να έχει και διάφορες ακόμα ιδιότητες εκτός από το να είναι ευσταθές, ιδιότητες όπως ευρωστία, επιθυμητή χρονική απόκριση και πολλά άλλα.

Αυτές οι απαιτήσεις μας οδηγούν στο αρκετά γνωστό πρόβλημα ανατοποθέτησης ιδιοτιμών: Δοθέντος του ζευγαριού (A, B) και των επιθυμητών ιδιοτιμών $\Omega = \{\lambda_1, \dots, \lambda_n\}$, κλειστό ως προς τη μιγαδική συζυγία, βρες έναν πίνακα F έτσι ώστε ο πίνακας $A + BF$ του κλειστού συστήματος να έχει ως ιδιοτιμές τις ιδιοτιμές του συνόλου $\Omega = \{\lambda_1, \dots, \lambda_n\}$. Είναι γνωστό πως το πρόβλημα αυτό έχει λύση αν και μόνο αν το ζεύγος πινάκων (A, B) είναι ελέγξιμο.

Και τώρα θα δείξουμε πως οι nonsingular λύσεις της εξίσωσης Sylvester-παρατηρητή μπορούν να δώσουν λύση στο πρόβλημα ανατοποθέτησης ιδιοτιμών.

Ας θεωρήσουμε ότι έχουμε την Sylvester-παρατηρητή εξίσωση

$$XT - AX = BG.$$

Οι πίνακες A, B είναι οι πίνακες του συστήματος

$$\dot{x}(t) = Ax(t) + Bu(t)$$

διάστασης αντίστοιχα $n \times n$ και $n \times m$. Ο πίνακας T είναι τέτοιος ώστε οι ιδιοτιμές του πίνακα αυτού να είναι το επιθυμητό σύνολο ιδιοτιμών $\Omega = \{\lambda_1, \dots, \lambda_n\}$.

Ας υποθέσουμε ότι η εξίσωση $XT - AX = BG$ έχει μια nonsingular λύση X . Τότε σαν επακόλουθο θα έχουμε τον πίνακα ανάδρασης F που τον παίρνουμε αν λύσουμε το σύστημα

$$FX = G$$

ως προς F . Αν έχουμε μια nonsingular λύση X μπορούμε να λύσουμε το σύστημα $FX = G$ ως $F = GX^{-1}$, αλλά γενικά για καλύτερα αποτελέσματα καλύτερα να προτιμήσουμε μια άλλη μέθοδο επίλυσης του $FX = G$, όπως με την μέθοδο απαλοιφής Gauss με μερική οδήγηση.

Εφόσον πάρουμε τον πίνακα F από το σύστημα $FX = G$ δημιουργούμε τον πίνακα $A + BF$ όπου οι ιδιοτιμές του πίνακα αυτού θα έχουν το επιθυμητό σύνολο ιδιοτιμών $\Omega = \{\lambda_1, \dots, \lambda_n\}$.

Όπως είπαμε όμως αν θέλουμε μια nonsingular λύση X από την εξίσωση

$$XT - AX = BG$$

τότε σύμφωνα με τον Bhattacharya και Desouza (1981), οι οποίοι πρώτοι ασχολήθηκαν με την μέθοδο αυτήν, απαραίτητη συνθήκη για να έχουμε μια nonsingular λύση X : θα πρέπει το ζεύγος πινάκων (A, B) είναι ελέγξιμο και το ζεύγος πινάκων (G, T) να είναι παρατηρήσιμο.

Ο πίνακας T μπορεί να έχει διδιαγώνια μορφή, δηλαδή έχει τις επιθυμητές ιδιοτιμές στην διαγώνιο του και κάτω ή πάνω από αυτήν κάποια αλλά επιθυμητά στοιχεία, ενώ αν θέλουμε παραδείγματος χάριν να έχουμε στο σύστημα μας τις μιγαδικές ιδιοτιμές $x \pm yi$ μπορούμε σαν πίνακα T μπορούμε να έχουμε ένα πίνακα τις μορφής

$$\begin{bmatrix} x & -y \\ y & x \end{bmatrix}.$$

Αριθμητική επίλυση των εξισώσεων Sylvester – Η εξίσωση Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab

Βεβαίως ο T μπορεί να έχει διαγώνια μορφή ή τριγωνική μορφή, αλλά γενικά θέλουμε να πούμε ότι δεν έχουμε κανένα περιορισμό πάνω σε αυτό το θέμα. Ο T μπορεί να έχει οποιαδήποτε μορφή.

Δημιουργία της παραπάνω μεθόδου σε συνάρτηση με όνομα **SylvEqInPolePlacement** και εφαρμογή αυτής σε δικό μας παράδειγμα παρουσιάζεται εκτενώς στο παράρτημα.

ΠΑΡΑΡΤΗΜΑ

Εδώ γίνεται υλοποίηση των αλγορίθμων στην Matlab με προγράμματα που έχει κάνει ο συγγραφέας αυτής της διπλωματικής για την επίλυση της εξίσωσης Sylvester και της γενικευμένης εξίσωσης Sylvester.

Συνάρτηση επίλυσης του συστήματος $AX=B$ που βασίζεται στην μέθοδο απαλοιφής Gauss με μερική οδήγηση.

```
function xx=mygaussel(A,b)
% A*X=b  A(n,n),b(n,1)
% P*A=L*U
% L*y=P*b
% U*x=y
%-----ΚΑΤΩΤΟΙΚΙΔΗΣ ΔΗΜΗΤΡΙΟΣ Χ.-----
[n1,n2]=size(A);
[n3,ena]=size(b);
if (n1~=n2) || (n2~=n3) || (ena~=1)
    disp('wrong dimensions,A(n,n),b(n,1)')
else
    n=n1;
    [L,U,P]=lu(A);
    bb=P*b;
    y(1)=bb(1)/L(1,1);
    for i=2:n
        sum=0;
        for j=i-1:-1:1
            sum=sum+L(i,j)*y(j);
        end
        y(i)=(1/L(i,i))*(bb(i)-sum);
    end
end
```

Αριθμητική επίλυση των εξίσωσεων Sylvester – Η εξίσωση Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab

```
yy=y';
x(n)=yy(n)/U(n,n);
for i=n-1:-1:1
    sum=0;
    for j=i+1:n
        sum=sum+U(i,j)*x(j);
    end
    x(i)=(1/U(i,i))*(yy(i)-sum);
end
xx=x';
end
```

Η συνάρτηση αυτή φτιάχτηκε για να χρησιμοποιηθεί στα προγράμματα μας.

Συνάρτηση επίλυσης της εξίσωσης Sylvester $BX+XA=C$ με την Hessenberg-Schur μέθοδο.

Αυτή η συνάρτηση βασίζεται στον αλγόριθμο 2.5.2, ενώ γίνεται εφαρμογή μέσω αυτής του παραδείγματος 2.5.3. Η συνάρτηση μας θα είναι η :

```
function X=HessSch(B,A,C)
% Η συνάρτηση αυτή μας λύνει την εξίσωση Sylvester με την Hessenberg-Schur
% μέθοδο. Η εξίσωση είναι η  $B*X+X*A=C$ .
% Ο πίνακας A είναι ένας  $n*n$  πίνακας.
% Ο πίνακας B είναι ένας  $m*m$  πίνακας.
% Ο πίνακας C είναι ένας  $m*n$  πίνακας.
% Είσοδος: B,A,C σε αυτή την σειρά.
% Ο μεγαλύτερος από τους πίνακες A,B διαλέγεται να μετασχηματιστεί σε
% Hessenberg μορφή, ενώ ο άλλος σε Schur μορφή.
%-----ΚΑΤΩΤΟΙΚΙΔΗΣ ΔΗΜΗΤΡΙΟΣ X.-----
[n1,n2]=size(A);
```

```
[m1,m2]=size(B);
[m3,n3]=size(C);
if (n1~=n2) || (n2~=n3) || (m1~=m2) || (m2~=m3)
    disp('wrong dimensions,A is nxn,B is mxm,C is mxn')
else
    n=n1;
    m=m1;
    if m>=n
        [U,R]=schur(transpose(A));
        [V,H]=hess(B);
        CC=transpose(V)*C*U;
        I=eye(m);
        if n==1
            F=(H+R(1,1)*I);
            Z=CC(:,1);
            Y=mygaussel(F,Z);
            for i=1:m
                YY(i,1)=Y(i);
            end
            X=V*YY*U';
        else
            for k=n:-1:2
                if R(k,k-1)==0
                    F=(H+R(k,k)*I);
                    P=0;
                    if k+1<=n
                        for j=k+1:n
                            PP=R(k,j)*YY(:,j);
                            P=P+PP;
                        end
                    end
                    Z=CC(:,k)-P;
                    Y=mygaussel(F,Z);
                end
            end
        end
    end
end
```

**Αριθμητική επίλυση των εξίσωσεων Sylvester – Η εξίσωση
Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab**

```

for i=1:m
    YY(i,k)=Y(i);
end
else
FF=[H+R(k-1,k-1)*I R(k-1,k)*I;R(k,k-1)*I H+R(k,k)*I];
PPP=0;
if k+1<=n
    for j=k+1:n
        PPPP=[R(k-1,j)*YY(:,j);R(k,j)*YY(:,j)];
        PPP=PPP+PPPP;
    end
    end
    ZZ=[CC(:,k-1);CC(:,k)]-PPP;
    G=mygaussel(FF,ZZ);
    for i=1:m
        YY(i,k-1)=G(i);
    end
    for i=m+1:2*m
        YY(i-m,k)=G(i);
    end
end
end
if R(2,1)==0
    F=(H+R(1,1)*I);
    P=0;
    if 2<=n
        for j=2:n
            PP=R(1,j)*YY(:,j);
            P=P+PP;
        end
    end
    Z=CC(:,1)-P;
    Y=mygaussel(F,Z);

```

```
    for i=1:m
        YY(i,1)=Y(i);
    end
end
X=V*YY*U';
end
else
    [U,R]=schur(B);
[V,H]=hess(transpose(A));
CC=transpose(V)*C'*U;
I=eye(n);
if m==1
    F=(H+R(1,1)*I);
    Z=CC(:,1);
    Y=mygaussel(F,Z);
    for i=1:n
        YY(i,1)=Y(i);
    end
    Xtran=V*YY*U';
    X=Xtran';
else
    for k=m:-1:2
        if R(k,k-1)==0
            F=(H+R(k,k)*I);
            P=0;
            if k+1<=m
                for j=k+1:m
                    PP=R(k,j)*YY(:,j);
                    P=P+PP;
                end
            end
            Z=CC(:,k)-P;
            Y=mygaussel(F,Z);
```

**Αριθμητική επίλυση των εξίσωσεων Sylvester – Η εξίσωση
Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab**

```

for i=1:n
    YY(i,k)=Y(i);
end
else
FF=[H+R(k-1,k-1)*I R(k-1,k)*I;R(k,k-1)*I H+R(k,k)*I];
PPP=0;
if k+1<=m
    for j=k+1:m
        PPPP=[R(k-1,j)*YY(:,j);R(k,j)*YY(:,j)];
        PPP=PPP+PPPP;
    end
    end
    ZZ=[CC(:,k-1);CC(:,k)]-PPP;
    G=mygaussel(FF,ZZ);
    for i=1:n
        YY(i,k-1)=G(i);
    end
    for i=n+1:2*n
        YY(i-n,k)=G(i);
    end
end
end
if R(2,1)==0
    F=(H+R(1,1)*I);
    P=0;
    if 2<=m
        for j=2:m
            PP=R(1,j)*YY(:,j);
            P=P+PP;
        end
    end
    Z=CC(:,1)-P;
    Y=mygaussel(F,Z);

```

Δημήτριος Χ. Κατωτοκίδης

```
    for i=1:n
        YY(i,1)=Y(i);
    end
end
Xtran=V*YY*U';
X=Xtran';
end
end
end
```

Αν δώσουμε σε αυτήν τα δεδομένα του παραδείγματος 2.5.3, δηλαδή :

```
>> A=[1 -1 0;1 1 0;0 0 2]
```

$$A = \begin{pmatrix} 1 & -1 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 2 \end{pmatrix}$$

```
>> B=[1 2 3 4;4 5 6 7;7 8 9 1;10 0 0 0]
```

$$B = \begin{pmatrix} 1 & 2 & 3 & 4 \\ 4 & 5 & 6 & 7 \\ 7 & 8 & 9 & 1 \\ 10 & 0 & 0 & 0 \end{pmatrix}$$

```
>> C=[12 10 12;24 22 24;27 25 27;12 10 12]
```

$$C = \begin{pmatrix} 12 & 10 & 12 \\ 24 & 22 & 24 \\ 27 & 25 & 27 \\ 12 & 10 & 12 \end{pmatrix}$$

Τότε σαν αποτέλεσμα θα πάρουμε

**Αριθμητική επίλυση των εξίσωσεων Sylvester – Η εξίσωση
Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab**

$$X = \begin{pmatrix} 1.000000000000000 & 1.000000000000000 & 1.000000000000000 \\ 1.000000000000000 & 1.000000000000000 & 1.000000000000000 \\ 1.000000000000000 & 1.000000000000000 & 1.000000000000000 \\ 1.000000000000000 & 1.000000000000000 & 1.000000000000000 \end{pmatrix}$$

που συμφωνεί με το αποτέλεσμα του παραδείγματος 2.5.3.

Μπορούμε επίσης να ζητήσουμε και τους πίνακες R, U, V, H, \hat{C} (εδώ ως CC) και Y (εδώ ως YY).

$$R = \begin{pmatrix} 1 & 1 & 0 \\ -1 & 1 & 0 \\ 0 & 0 & 2 \end{pmatrix}$$

$$U = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

$$V = \begin{pmatrix} 1.000000000000000 & 0 & 0 & 0 \\ 0 & -0.31139957766461 & -0.73977073283302 & -0.59646422011216 \\ 0 & -0.54494926091307 & -0.37519218659465 & 0.74984073385528 \\ 0 & -0.77849894416152 & 0.55854282374946 & -0.28630282565383 \end{pmatrix}$$

$$H = \begin{pmatrix} 1.000000000000000 & -5.37164271471451 & -0.37094673045214 & -0.08861754127381 \\ -12.84523257866513 & 7.65454545454546 & 5.39618856048896 & -0.76948892583061 \\ 0 & 10.66893226628042 & 4.78709671968580 & -0.27371950888108 \\ 0 & 0 & -5.33396264593099 & 1.55835782576875 \end{pmatrix}$$

$$CC = \begin{pmatrix} 12.000000000000000 & 10.000000000000000 & 12.000000000000000 \\ -31.52920723854169 & -28.25951167306329 & -31.52920723854169 \\ -21.18217274105446 & -20.06933254969805 & -21.18217274105446 \\ 2.49492462355485 & 2.76077724737627 & 2.49492462355485 \end{pmatrix}$$

$$YY = \begin{pmatrix} 1.000000000000000 & 1.000000000000000 & 1.000000000000000 \\ -1.63484778273920 & -1.63484778273920 & -1.63484778273920 \\ -0.55642009567820 & -0.55642009567820 & -0.55642009567820 \\ -0.13292631191071 & -0.13292631191070 & -0.13292631191071 \end{pmatrix}.$$

Συνάρτηση επίλυσης της εξίσωσης Sylvester $AX+XB=C$ με την Schur μέθοδο.(Bartels-Stewart)

Αυτή η συνάρτηση βασίζεται στον αλγόριθμο 2.5.1 των Bartels και Stewart. Η συνάρτηση αυτή έχει δυο μεθόδους . Στην πρώτη μέθοδο χρησιμοποιούμε πίνακες τάξης ένα για να βρούμε την λύση X . Στην δεύτερη μέθοδο χρησιμοποιούμε πίνακες τάξης δυο για να βρούμε την λύση X .Κανονικά χρησιμοποιείται η πρώτη μέθοδος. Όταν όμως οι πίνακες μας A, B δεν μετατρέπονται σε κανονικούς πίνακες Schur ,δηλαδή κάποια στοιχεία των πινάκων αυτών θα έπρεπε να ήταν μηδέν αλλά δεν είναι. Στην δεύτερη μέθοδο απαιτείται να επέμβει και ο χρήστης έτσι ώστε να χωρίσει τους πίνακες Schur καθώς και τον μετασχηματισμένο πίνακα C σε 2×2 πίνακες με την εντολή `mat2cell`. Αν οι πίνακες μας αυτοί έχουν περιττή διάσταση τότε το πρόγραμμα μας τους χωρίζει σε 2×2 πίνακες ξανά ,αφού πρώτα προσθέσει μια μηδενική γραμμή ή στήλη όπου χρειάζεται έτσι ώστε να έχουμε άρτιες διαστάσεις. Για να δουλέψει σωστά θα πρέπει την συνάρτηση αυτή να την καλέσουμε σαν `m-file` καθώς θα πρέπει επίσης ο χρήστης να έχει εξοικειωθεί με την χρήση της εντολής `mat2cell`. Γίνεται εφαρμογή του παραδείγματος 2.5.2 με την δεύτερη μέθοδο, ενώ δημιουργούμε και ένα δικό μας παράδειγμα για να εφαρμόσουμε και την πρώτη μέθοδο. Την ορθότητα του αποτελέσματος του δικού μας παραδείγματος μας την επαληθεύει η Matlab. Η συνάρτηση μας θα είναι η :

```
function xx=schrblockschr(A,B,C)
```

**Αριθμητική επίλυση των εξίσωσεων Sylvester – Η εξίσωση
Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab**

```

% Αυτη η συναρτηση λυνει την εξίσωση AX+XB=C
% μετατρεποντας τους πινακες A,B σε κατω και ανω πραγματικους
% πινακες Schur.
% Εισοδος :A(p,p),B(q,q),C(p,q) σε αυτην την σειρα.
%.....
% Εδω εχουμε δυο μεθοδους:
% 1.Εχουμε πινακες ταξης 1.Λυση γραμμικων συστηματων ταξης 1 με εμπρος
αντικατασταση.
% 2.Εχουμε πινακες ταξης 2.Λυση γραμμικων συστηματων ταξης 4 με Kronecker
γινομενο.
%.....
% οι πινακες A,B,C μπορούν να λυθουν με την πρωτη μεθοδο αν
% η μετατροπη τους σε Schur πινακες ειναι κανονικη.(T==1)
% Αν οχι τοτε η λυση βρισκεται με την δευτερη μεθοδο.(T==2)
% Και οι δυο μεθοδοι βασιζονται πιστα στον αλγοριθμο των Bartels και Stewart.
%-----ΚΑΤΩΤΟΙΚΙΔΗΣ ΔΗΜΗΤΡΙΟΣ Χ.-----

[p1,p2]=size(A);
[q1,q2]=size(B);
[p3,q3]=size(C);
if (p1~=p2) || (q1~=q2) || (p2~=p3) || (q2~=q3)
    disp('wrong dimensions,A is pxp,B is qxq,C is pxq')
else
    p=p1;
    q=q1;
[U,RA]=schur(transpose(A));
RA=RA';
[V,RB]=schur(B);
CC=U'*C*V;
%.....εξεταση της Schur μετατροπης των δυο πινακων.....
%..... και επιλογη μεθοδου.....
T=1;
if nnz(RB-triu(RB))~=0
    T=2;

```

```
end
if nnz(RA-tril(RA))~=0
    T=2;
end
%.....
if T==1
    P=0;
    PP=0;
    for k=1:p
        for L=1:q
            P=0;
            PP=0;
            if L>1
                for i=1:L-1
                    P=P+X(k,i)*RB(i,L);
                end
            end
            if k>1

                for j=1:k-1
                    PP=PP+RA(k,j)*X(j,L);
                end
            end
            CC(k,L)=CC(k,L)-P-PP;
            X(k,L)=CC(k,L)/(RA(k,k)+RB(L,L));
        end
    end
xx=U*X*V';
else
    [p,p]=size(A);
    [q,q]=size(B);
    p11=p;
    q11=q;
```

**Αριθμητική επίλυση των εξίσωσεων Sylvester – Η εξίσωση
Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab**

```
if mod(p,2)~=0
    p=p+1;
    for i=1:p
        A(i,p)=0;
        A(p,i)=0;
    end
    for i=1:q
        C(p,i)=0;
    end
end
if mod(q,2)~=0
    q=q+1;
    for i=1:q
        B(i,q)=0;
        B(q,i)=0;
    end
    for i=1:p
        C(i,q)=0;
    end
end
[U,RA]=schur(transpose(A));
RA=RA';
[V,RB]=schur(B);
CC=U'*C*V;
display(RA)
display(RB)
display(CC)
I2=eye(2);
RAA=input('Χωρισε σε 2x2 κομματια τον RA με την mat2cell εντολη: RAA=');
RBB=input('Χωρισε σε 2x2 κομματια τον RB με την mat2cell εντολη: RBB=');
RCC=input('Χωρισε σε 2x2 κομματια τον CC με την mat2cell εντολη: RCC=');
P=zeros(2);
PP=zeros(2);
```

```
for k=1:p/2
for L=1:q/2
    P=zeros(2);
    PP=zeros(2);
    if L>1
        for i=1:L-1
            P=P+CX{k,i}*RBB{i,L};
        end
    end
    if k>1
        for j=1:k-1
            PP=PP+RAA{k,j}*CX{j,L};
        end
    end
    RCC{k,L}=RCC{k,L}-P-PP;
    T=kron(I2,RAA{k,k})+kron(RBB{L,L}',I2);
    d=reshape(RCC{k,L},[4 1]);
    RCX=mygaussel(T,d);
    CX{k,L}=reshape(RCX,[2 2]);
end
end
X=cell2mat(CX);
xxtest=U*X*V';
xx=xxtest(1:p11,1:q11);
end
end
```

Αν εφαρμόσουμε το παράδειγμα 2.5.2 τότε με την δεύτερη μέθοδο για τους πίνακες

```
>> A=[-5 2 7 1 -8;3 -6 2 -2 -4;-1 4 -2 6 3;-4 -7 -1 5 -2;-2 3 1 0 9]
```

**Αριθμητική επίλυση των εξίσωσεων Sylvester – Η εξίσωση
Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab**

$$A = \begin{pmatrix} -5 & 2 & 7 & 1 & -8 \\ 3 & -6 & 2 & -2 & -4 \\ -1 & 4 & -2 & 6 & 3 \\ -4 & -7 & -1 & 5 & -2 \\ -2 & 3 & 1 & 0 & 9 \end{pmatrix}$$

>> B=[-13 -14;4 5]

$$B = \begin{pmatrix} -13 & -14 \\ 4 & 5 \end{pmatrix}$$

>> C=[4 7;-52 -27;-2 -9;15 36;-3 -46]

$$C = \begin{pmatrix} 4 & 7 \\ -52 & -27 \\ -2 & -9 \\ 15 & 36 \\ -3 & -46 \end{pmatrix}$$

Χωρίζουμε τους πίνακες Schur των A,B καθώς και τον μετασχηματισμένο πίνακα του C, εδώ ως CC ως εξής:

RAA=mat2cell(RA,[2 2 2],[2 2 2]);

RBB=mat2cell(RB,[2]);

RCC=mat2cell(CC,[2 2 2]);

Και θα πάρουμε σαν αποτέλεσμα

$$xx = \begin{pmatrix} -1.00000000000000 & 2.00000000000001 \\ 3.00000000000000 & 1.00000000000000 \\ -2.00000000000000 & -2.99999999999999 \\ -6.00000000000000 & -4.00000000000000 \\ 1.00000000000000 & -1.99999999999999 \end{pmatrix}$$

Δημήτριος Χ. Κατωτοκίδης

ενώ η λύση που μας δίνει το παράδειγμα 2.5.2 είναι

$$X = \begin{pmatrix} -1 & 2 \\ 3 & 1 \\ -2 & -3 \\ -6 & -4 \\ 1 & -2 \end{pmatrix}$$

Πολύ καλή ακρίβεια. Ο μετασχηματισμένος C και οι άνω και κάτω Schur πίνακες των πινάκων A και B αντίστοιχα είναι

RA =

Columns 1 through 3

-8.08124756092100	-1.20908078508019	0
2.18306660379438	-8.08124756092100	0
5.82473103816832	1.93728989590828	4.00474581954500
-0.14891551832865	-4.91339834215884	8.75088140616510
4.66200190262518	-6.68426283501860	0.16060375306769
0	0	0

Columns 4 through 6

0	0	0
0	0	0
-1.95337851367168	0	0
4.00474581954500	0	0
2.57458497615018	9.15300348275201	0
0	0	0

$$RB = \begin{pmatrix} -9.00000000000000 & -18.00000000000000 \\ 0 & 1.00000000000000 \end{pmatrix}$$

**Αριθμητική επίλυση των εξίσωσεων Sylvester – Η εξίσωση
Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab**

$$CC = \begin{pmatrix} -37.56756288188821 & -36.12628033825931 \\ 0.21316197455794 & 4.04495728900706 \\ -10.25968871819843 & -51.41780619440419 \\ 1.54952943479863 & -31.42597557787879 \\ 20.34301972551386 & -18.52780424272021 \\ & 0 & 0 \end{pmatrix}$$

Βλέπουμε πως ο πίνακας A δεν μετασχηματίστηκε ακριβώς σε κάτω πραγματική μορφή Schur. Για αυτό τον λόγο το παράδειγμα αυτό λύθηκε με την δεύτερη μέθοδο του προγράμματος μας. Αν χρησιμοποιούνταν η πρώτη μέθοδος για τα παραπάνω δεδομένα θα είχαμε τότε ανακριβή αποτελέσματα.

Τώρα ας κάνουμε εφαρμογή της πρώτης μεθόδου σε ένα δικό μας παράδειγμα. Έστω οι πίνακες :

>> A=[1 2 3 4;4 5 6 7;7 8 9 1;10 0 0 0]

$$A = \begin{pmatrix} 1 & 2 & 3 & 4 \\ 4 & 5 & 6 & 7 \\ 7 & 8 & 9 & 1 \\ 10 & 0 & 0 & 0 \end{pmatrix}$$

>> B=[1 -1 1;1 1 2;3 5 2]

$$B = \begin{pmatrix} 1 & -1 & 1 \\ 1 & 1 & 2 \\ 3 & 5 & 2 \end{pmatrix}$$

>> C=[12 10 12;24 22 24;27 25 27;12 10 12]

$$C = \begin{pmatrix} 12 & 10 & 12 \\ 24 & 22 & 24 \\ 27 & 25 & 27 \\ 12 & 10 & 12 \end{pmatrix}$$

Το πρόγραμμα μας χρησιμοποιεί την πρώτη μέθοδο, αφού οι πίνακες A και B μετασχηματίστηκαν επιτυχώς σε κάτω και άνω πραγματικούς πίνακες Schur.

$$RA = \begin{pmatrix} 17.11147212530491 & 0 & 0 & 0 \\ -5.21546524896087 & -6.11200014609071 & 0 & 0 \\ -4.48769002032916 & 6.03755603177135 & 3.19166312415674 & 0 \\ 0.93745481589933 & 1.70214731843237 & -4.73617468597196 & 0.80886489662907 \end{pmatrix}$$

$$RB = \begin{pmatrix} 4.99999999999999 & 2.09761769634030 & 3.36242209618917 \\ & 0 & 1.00000000000000 & -1.13759291798904 \\ & 0 & & 0 & -2.00000000000000 \end{pmatrix}$$

Το πρόγραμμα μας λοιπόν μας δίνει :

$$xx = \begin{pmatrix} 1.08943904116318 & 0.73409961685824 & 1.27383829452795 \\ 0.86208861381275 & 0.19050987326849 & 2.38836820905786 \\ 1.22235976029080 & 2.42198644267611 & -0.71230965713725 \\ -0.03143727281658 & -1.09873268493958 & 0.74525984870813 \end{pmatrix}$$

Η συνάρτηση $X_{lyap} = lyap(A, B, -C)$ της Matlab μας δίνει την ακριβή λύση, που είναι

$$X_{lyap} = \begin{pmatrix} 1.08943904116318 & 0.73409961685824 & 1.27383829452795 \\ 0.86208861381275 & 0.19050987326849 & 2.38836820905786 \\ 1.22235976029080 & 2.42198644267610 & -0.71230965713724 \\ -0.03143727281658 & -1.09873268493958 & 0.74525984870813 \end{pmatrix}$$

Βλέπουμε ότι η λύση μας έχει πολύ καλή ακρίβεια 13 δεκαδικών ψηφίων.

Αριθμητική επίλυση των εξίσωσεων Sylvester – Η εξίσωση Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab

Συνάρτηση επίλυσης της γενικευμένης εξίσωσης Sylvester με την Bartels-Stewart μέθοδο

Εδώ παρουσιάζουμε την συνάρτηση την οποία φτιάξαμε για την επίλυση της γενικευμένης εξίσωσης Sylvester η οποία ακολουθεί πιστά τον αλγόριθμο Bartels-Stewart για την γενικευμένη εξίσωση Sylvester. Την συνάρτηση αυτή την εφαρμόζουμε σε ένα δικό μας παράδειγμα, ενώ επαληθεύουμε την εγγύτητα των αποτελεσμάτων μας κάνοντας τις πράξεις, δηλαδή επαληθεύουμε την εξίσωση Sylvester για την υπολογισμένη λύση X που θα πάρουμε.

Έτσι θα έχουμε την παρακάτω συνάρτηση,

```
function X=genSchSch(A,B,C,D,E)
% Η συνάρτηση αυτή μας λύνει την γενική εξίσωση Sylvester με την
% Bartels-Stewart μέθοδο.
% Η εξίσωση είναι η  $A*X*B'+C*X*D'=E$ 
.....
% Ο πίνακας A είναι ένας m*m πίνακας.
% Ο πίνακας B είναι ένας n*n πίνακας.
% Ο πίνακας C είναι ένας m*m πίνακας.
% Ο πίνακας D είναι ένας n*n πίνακας.
% Ο πίνακας E είναι ένας m*n πίνακας.
% Είσοδος: A,B,C,D,E σε αυτή την σειρά.
%-----AA*Y*transpose(BB)+CC*Y*transpose(DD)=EE-----
%-----The solution is X-----
%-----ΚΑΤΩΤΟΙΚΙΔΗΣ ΔΗΜΗΤΡΙΟΣ X.-----

[m1,m2]=size(A);
[n1,n2]=size(B);
[m3,m4]=size(C);
[n3,n4]=size(D);
[m5,n5]=size(E);
if (n1~=n2) || (n2~=n3) || (n3~=n4) || (n4~=n5) || (m1~=m2) || (m2~=m3) || (m3~=m4)
    || (m4~=m5)
```

```
disp('matrix A,C must have n*n dimensions , matrix B,D must have m*m ,matrix E
must have n*m')

else
    n=n1;
    m=m1;
[AA,CC,U1,R1]=qz(A,C,'real'); % U1*A*R1=AA , U1*C*R1=CC
[DD,BB,U2,R2]=qz(D,B,'real'); % U2*B*R2=BB , U2*D*R2=DD
EE=U1*E*transpose(U2);
if n==1
    F=(BB(1,1)*AA+DD(1,1)*CC);
    Z=EE(:,1);
    Y=mygaussel(F,Z);
    for i=1:m
        YY(i,1)=Y(i);
    end
    X=R1*YY*R2';
else
    for k=n:-1:2
        if DD(k,k-1)==0
            F=(BB(k,k)*AA+DD(k,k)*CC);
            P=0;
            if k+1<=n
                for j=k+1:n
                    PP=(BB(k,j)*AA+DD(k,j)*CC)*YY(:,j);
                    P=P+PP;
                end
            end
            Z=EE(:,k)-P;
            Y=mygaussel(F,Z);
            for i=1:m
                YY(i,k)=Y(i);
            end
        end
    end
else
```

**Αριθμητική επίλυση των εξίσωσεων Sylvester – Η εξίσωση
Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab**

```

FF=[BB(k-1,k-1)*AA+DD(k-1,k-1)*CC BB(k-1,k)*AA+DD(k-1,k)*CC;
                                     DD(k,k-1)*CC BB(k,k)*AA+DD(k,k)*CC];

PPP=0;
if k+1<=n
    for j=k+1:n
        PPPP=[(BB(k-1,j)*AA+DD(k-1,j)*CC)*YY(:,j);
              (BB(k,j)*AA+DD(k,j)*CC)*YY(:,j)];

        PPP=PPP+PPPP;
    end
end
ZZ=[EE(:,k-1);EE(:,k)]-PPP;
G=mygaussel(FF,ZZ);
for i=1:m
    YY(i,k-1)=G(i);
end
for i=m+1:2*m
    YY(i-m,k)=G(i);
end
end
end
if DD(2,1)==0
    F=(BB(1,1)*AA+DD(1,1)*CC);
    P=0;
    if 2<=n
        for j=2:n
            PP=(BB(1,j)*AA+DD(1,j)*CC)*YY(:,j);
            P=P+PP;
        end
    end
    Z=EE(:,1)-P;
    Y=mygaussel(F,Z);
    for i=1:m
        YY(i,1)=Y(i);
    end
end

```

Δημήτριος Χ. Κατωτοκίδης

```
end  
end  
X=R1*YY*R2';  
end  
end
```

Λοιπόν για τους πίνακες

```
>> A=[1 2;3 4]
```

$$A = \begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix}$$

```
>> B=[6 3 8;2 9 1;7 2 4]
```

$$B = \begin{pmatrix} 6 & 3 & 8 \\ 2 & 9 & 1 \\ 7 & 2 & 4 \end{pmatrix}$$

```
>> C=[6 1;8 4]
```

$$C = \begin{pmatrix} 6 & 1 \\ 8 & 4 \end{pmatrix}$$

```
>> D=[4 2 7;12 7 8;7 15 3]
```

$$D = \begin{pmatrix} 4 & 2 & 7 \\ 12 & 7 & 8 \\ 7 & 15 & 3 \end{pmatrix}$$

```
>> E=[21 23 12;13 17 20]
```

**Αριθμητική επίλυση των εξίσωσεων Sylvester – Η εξίσωση
Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab**

$$E = \begin{pmatrix} 21 & 23 & 12 \\ 13 & 17 & 20 \end{pmatrix}$$

παίρνουμε την υπολογισμένη λύση X

$$X = \begin{pmatrix} -0.67404235868105 & 0.31739825477626 & 1.13439830339412 \\ -0.92235587343585 & 0.84826516703287 & -0.53426285183511 \end{pmatrix}$$

επαληθεύουμε την λύση αυτή στην γενικευμένη εξίσωση Sylvester, θα έχουμε

$$AXB^T + CXD^T = \begin{pmatrix} 21.000000000000002 & 23.000000000000001 & 12.000000000000002 \\ 13.000000000000005 & 17.000000000000004 & 20.000000000000005 \end{pmatrix} \cong E$$

Συνάρτηση επίλυσης της γενικευμένης εξίσωσης Sylvester με την Hessenberg-Schur μέθοδο

Εδώ παρουσιάζουμε την συνάρτηση την οποία φτιάξαμε για την επίλυση της γενικευμένης εξίσωσης Sylvester η οποία ακολουθεί πιστά τον αλγόριθμο Hessenberg-Schur για την γενικευμένη εξίσωση Sylvester. Την συνάρτηση αυτή την εφαρμόζουμε σε ένα δικό μας παράδειγμα, ενώ επαληθεύουμε την εγγύτητα των αποτελεσμάτων μας κάνοντας τις πράξεις, δηλαδή επαληθεύουμε την εξίσωση Sylvester για την υπολογισμένη λύση X που θα πάρουμε.

Η συνάρτηση μας αυτή χρησιμοποιεί μια άλλη συνάρτηση μας την **myqzpart1**, την οποία φτιάξαμε, η οποία θεωρείται στην ουσία το πρώτο βήμα του αλγορίθμου QZ. Η συνάρτηση αυτή παίρνει ως πούμε το ζευγάρι πινάκων (A,B) και επιστρέφει το ζευγάρι πινάκων ως πούμε (HA,TB) όπου HA είναι ένας πίνακας σε Hessenberg μορφή και TB ένας πίνακας σε άνω τριγωνική μορφή. Η συνάρτηση αυτή μας δίνει και πίνακες Q, Z , έτσι ώστε να ισχύει ταυτοχρόνως $HA=Q'*A*Z$ και $TB=Q'*B*Z$.

Έτσι πρώτα θα έχουμε την βοηθητική συνάρτηση **myqzpart1**

```
function [HA,TB,Q,Z]=myqzpart1(A,B)
% αυτή η συνάρτηση παίρνει το ζευγάρι (A,B) και αναγάγει τον HA σε άνω πίνακα
hessenberg
% και τον πίνακα TB σε άνω τριγωνικό πίνακα ενώ μας δίνει και τους πίνακες Q
% και Z έτσι ώστε να ισχύει HA=Q'*A*Z και TB=Q'*B*Z.
% Αυτό στην ουσία θεωρείται το πρώτο βήμα του αλγορίθμου QZ
% Βεβαίως και οι δυο πίνακες είναι τάξης nxn
%-----ΚΑΤΩΤΟΙΚΙΔΗΣ ΔΗΜΗΤΡΙΟΣ Χ.-----
[n1,n2]=size(A);
[n3,n4]=size(B);
if (n1~=n2) || (n2~=n3) || (n3~=n4)
    disp('error kai oi dyo pinakes prepei na einai nxn')
else
    n=n1;
    Ia=eye(n);
    Ib=eye(n);
    [U,Tb]=qr(B);% B=U*Tb
    Ha=U'*A;
    Ta=Ha;
    TTb=Tb;
    .....
    if n<=2
        Ra=eye(n);
        Ta=Ra*Ta; % πολλαπλασιασμός από αριστερά
        TTb=Ra*TTb; % update B
        Ia=Ra*Ia;
        Rb=eye(n);
        TTb=TTb*Rb; % πολλαπλασιασμός από δεξιά
        Ta=Ta*Rb; % update A
        Ib=Ib*Rb;
        Q=U*Ia';
```

**Αριθμητική επίλυση των εξίσωσεων Sylvester – Η εξίσωση
Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab**

```

Z=Ib;
HA=Ta;
TB=TTb;
end
.....
if n>2
for j=1:n-2
    for i=1:n-1-j
[GA,ya]=planerot([Ta(n-i,j) Ta(n-i+1,j)]');
Ra=eye(n);
Ra(n-i,n-i)=GA(1,1);
Ra(n-i,n-i+1)=GA(1,2);
Ra(n-i+1,n-i)=GA(2,1);
Ra(n-i+1,n-i+1)=GA(2,2);
Ta=Ra*Ta; % πολλαπλασιασμός από αριστερά
TTb=Ra*TTb; % update B
Ia=Ra*Ia;
[GB,yb]=planerot([TTb(n+1-j,n-i+1),TTb(n+1-j,n-i)]');
Rb=eye(n);
Rb(n-i,n-i)=GB(1,1);
Rb(n-i,n-i+1)=GB(1,2);
Rb(n-i+1,n-i)=GB(2,1);
Rb(n-i+1,n-i+1)=GB(2,2);
TTb=TTb*Rb; % πολλαπλασιασμός από δεξιά
Ta=Ta*Rb; % update A
Ib=Ib*Rb;
end
end
Q=U*Ia';
Z=Ib;
HA=Ta;
TB=TTb;
end

```


end

Ενώ παρουσιάζουμε τώρα και την συνάρτηση την οποία φτιάξαμε για την επίλυση της γενικευμένης εξίσωσης Sylvester η οποία ακολουθεί πιστά τον αλγόριθμο Hessenberg-Schur για την γενικευμένη εξίσωση Sylvester. Αν $m \geq n$ όλα λειτουργούν πιστά σύμφωνα με τον αλγόριθμο. Αν $n > m$ τότε η συνάρτηση κάνει χρήση του αλγορίθμου λύνοντας το ανάστροφο πρόβλημα, όπως και θα έπρεπε.

```
function X=genHessSch(A,B,C,D,E)
% Η συνάρτηση αυτή μας λύνει την γενικευμένη εξίσωση Sylvester με την Hessenberg-
Schur
% μέθοδο. Η εξίσωση είναι η  $A*X*B'+C*X*D'=E$ .
% Ο πίνακας A είναι ένας m*m πίνακας.
% Ο πίνακας B είναι ένας n*n πίνακας.
% Ο πίνακας C είναι ένας m*m πίνακας.
% Ο πίνακας D είναι ένας n*n πίνακας.
% Είσοδος: A,B,C,D,E σε αυτή την σειρά.
% Ο μεγαλύτερος από τους πίνακες A,B από τα ζευγάρια (A,C),(B,D) διαλέγεται να
μετασχηματιστεί σε
% Hessenberg μορφή, ενώ ο άλλος σε Schur μορφή. Οι υπόλοιποι δυο πίνακες C,D
% μετασχηματίζονται σε άνω τριγωνικούς πίνακες.
%-----AA*Y*transpose(BB)+CC*Y*transpose(DD)=EE-----
%----The solution is X-----
%-----ΚΑΤΩΤΟΙΚΙΔΗΣ ΔΗΜΗΤΡΙΟΣ Χ.-----

[m1,m2]=size(A);
[n1,n2]=size(B);
[m3,m4]=size(C);
[n3,n4]=size(D);
[m5,n5]=size(E);
if (n1~=n2) || (n2~=n3) || (n3~=n4) || (n4~=n5) || (m1~=m2) || (m2~=m3) || (m3~=m4)
```

**Αριθμητική επίλυση των εξίσωσεων Sylvester – Η εξίσωση
Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab**

|| (m4~m5)

disp('matrix A,C must have n*n dimensions , matrix B,D must have m*m ,matrix E
must have n*m')

else

n=n1;

m=m1;

if m>=n

[AA,CC,U1,R1]=myqzpart1(A,C);

U1=U1'; % U1*A*R1=AA , U1*C*R1=CC

[DD,BB,U2,R2]=qz(D,B,'real'); % U2*B*R2=BB , U2*D*R2=DD

EE=U1*E*transpose(U2);

if n==1

F=(BB(1,1)*AA+DD(1,1)*CC);

Z=EE(:,1);

Y=mygaussel(F,Z);

for i=1:m

YY(i,1)=Y(i);

end

X=R1*YY*R2';

else

for k=n:-1:2

if DD(k,k-1)==0

F=(BB(k,k)*AA+DD(k,k)*CC);

P=0;

if k+1<=n

for j=k+1:n

PP=(BB(k,j)*AA+DD(k,j)*CC)*YY(:,j);

P=P+PP;

end

end

Z=EE(:,k)-P;

Y=mygaussel(F,Z);

```

for i=1:m
    YY(i,k)=Y(i);
end
else
    FF=[BB(k-1,k-1)*AA+DD(k-1,k-1)*CC    BB(k-1,k)*AA+DD(k-1,k)*CC;
        DD(k,k-1)*CC    BB(k,k)*AA+DD(k,k)*CC];
    PPP=0;
    if k+1<=n
        for j=k+1:n
            PPPP=[(BB(k-1,j)*AA+DD(k-1,j)*CC)*YY(:,j);
                (BB(k,j)*AA+DD(k,j)*CC)*YY(:,j)];
            PPP=PPP+PPPP;
        end
    end
    ZZ=[EE(:,k-1);EE(:,k)]-PPP;
    G=mygaussel(FF,ZZ);
    for i=1:m
        YY(i,k-1)=G(i);
    end
    for i=m+1:2*m
        YY(i-m,k)=G(i);
    end
end
end
if DD(2,1)==0
    F=(BB(1,1)*AA+DD(1,1)*CC);
    P=0;
    if 2<=n
        for j=2:n
            PP=(BB(1,j)*AA+DD(1,j)*CC)*YY(:,j);
            P=P+PP;
        end
    end
end
end

```

**Αριθμητική επίλυση των εξίσωσεων Sylvester – Η εξίσωση
Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab**

```

    Z=EE(:,1)-P;
    Y=mygaussel(F,Z);
    for i=1:m
        YY(i,1)=Y(i);
    end
end
X=R1*YY*R2';
end
.....
else
.....το ανάστροφο πρόβλημα.....
    [AA,CC,U1,R1]=myqzpart1(D,B);
    U1=U1'; % U1*B'*R1=CC , U1*D'*R1=AA
    [DD,BB,U2,R2]=qz(A,C,'real'); % U2*A'*R2=DD , U2*C'*R2=BB

    EE=U1*E'*transpose(U2);
    if m==1
        F=(BB(1,1)*AA+DD(1,1)*CC);
        Z=EE(:,1);
        Y=mygaussel(F,Z);
        for i=1:n
            YY(i,1)=Y(i);
        end
        Xtrans=R1*YY*R2';
        X=Xtrans';
    else
        for k=m:-1:2
            if DD(k,k-1)==0
                F=(BB(k,k)*AA+DD(k,k)*CC);
                P=0;
                if k+1<=m
                    for j=k+1:m
                        PP=(BB(k,j)*AA+DD(k,j)*CC)*YY(:,j);
                    end
                end
            end
        end
    end
end

```

```

        P=P+PP;
    end
end
    Z=EE(:,k)-P;
    Y=mygaussel(F,Z);
    for i=1:n
        YY(i,k)=Y(i);
    end
else
    FF=[BB(k-1,k-1)*AA+DD(k-1,k-1)*CC  BB(k-1,k)*AA+DD(k-1,k)*CC;
        DD(k,k-1)*CC  BB(k,k)*AA+DD(k,k)*CC];

    PPP=0;
    if k+1<=m
        for j=k+1:m
            PPPP=[(BB(k-1,j)*AA+DD(k-1,j)*CC)*YY(:,j);
                (BB(k,j)*AA+DD(k,j)*CC)*YY(:,j)];

            PPP=PPP+PPPP;
        end
    end
    ZZ=[EE(:,k-1);EE(:,k)]-PPP;
    G=mygaussel(FF,ZZ);
    for i=1:n
        YY(i,k-1)=G(i);
    end
    for i=n+1:2*n
        YY(i-n,k)=G(i);
    end
end
end
end
    if DD(2,1)==0
        F=(BB(1,1)*AA+DD(1,1)*CC);
        P=0;
        if 2<=m

```

**Αριθμητική επίλυση των εξίσωσεων Sylvester – Η εξίσωση
Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab**

```
for j=2:m
    PP=(BB(1,j)*AA+DD(1,j)*CC)*YY(:,j);
    P=P+PP;
end
end
Z=EE(:,1)-P;
Y=mygaussel(F,Z);
for i=1:n
    YY(i,1)=Y(i);
end
end
Xtrans=R1*YY*R2';
X=Xtrans';
end
end
end
```

Για τους πίνακες πάλι

```
>> A=[1 2;3 4]
```

$$A = \begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix}$$

```
>> B=[6 3 8;2 9 1;7 2 4]
```

$$B = \begin{pmatrix} 6 & 3 & 8 \\ 2 & 9 & 1 \\ 7 & 2 & 4 \end{pmatrix}$$

```
>> C=[6 1;8 4]
```

$$C = \begin{pmatrix} 6 & 1 \\ 8 & 4 \end{pmatrix}$$

>> D=[4 2 7;12 7 8;7 15 3]

$$D = \begin{pmatrix} 4 & 2 & 7 \\ 12 & 7 & 8 \\ 7 & 15 & 3 \end{pmatrix}$$

>> E=[21 23 12;13 17 20]

$$E = \begin{pmatrix} 21 & 23 & 12 \\ 13 & 17 & 20 \end{pmatrix}$$

θα έχουμε

$$X = \begin{pmatrix} -0.67404235868105 & 0.31739825477626 & 1.13439830339412 \\ -0.92235587343585 & 0.84826516703287 & -0.53426285183511 \end{pmatrix}$$

Επαληθεύουμε την λύση αυτή στην γενικευμένη εξίσωση Sylvester, θα έχουμε

$$AXB^T + CXD^T = \begin{pmatrix} 21.000000000000002 & 22.999999999999999 & 12.000000000000001 \\ 13.000000000000002 & 16.999999999999999 & 20.000000000000001 \end{pmatrix} \cong E$$

Επίσης θα θέλαμε να δείξουμε για την βοηθητική δική μας συνάρτηση **myqzpart1** ότι στο συγκεκριμένο παράδειγμα η συνάρτηση παίρνει το ζεύγος πινάκων (D,B), επειδή $n > m$ και το μετατρέπει στο συγκεκριμένο παράδειγμα στο ζεύγος (AA,CC), δηλαδή ο πίνακας D ανάγεται στην άνω Hessenberg μορφή :

**Αριθμητική επίλυση των εξίσωσεων Sylvester – Η εξίσωση
Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab**

$$AA = \begin{pmatrix} -10.28197943606169 & -7.03700464476891 & -14.60905150426337 \\ 10.16272103702569 & 1.29283181177245 & 7.77087551861361 \\ 0 & -8.64308932786502 & -0.54338405065618 \end{pmatrix}$$

ενώ ο πίνακας B ανάγεται στην παρακάτω άνω τριγωνική μορφή :

$$CC = \begin{pmatrix} -9.43398113205660 & 0.09038252219556 & -9.82045157830987 \\ 0 & 7.24792590617632 & 3.21093831983032 \\ 0 & 0 & 3.96333280996892 \end{pmatrix}$$

Επίσης παίρνουμε τους πίνακες U_1, R_1 (στην συνάρτηση μας **myqzpart1** είναι Q^T, Z)
έτσι ώστε να ισχύει

$$U_1 D R_1 = AA$$

$$U_1 B R_1 = CC$$

Οι πίνακες αυτοί είναι οι

$$U_1 = \begin{pmatrix} -0.63599872800382 & -0.21199957600127 & -0.74199851600445 \\ -0.24986672697648 & 0.96629875830727 & -0.06191387925081 \\ 0.73011796083071 & 0.14602359216614 & -0.66753642133093 \end{pmatrix}$$

$$R_1 = \begin{pmatrix} 1.00000000000000 & 0 & 0 \\ 0 & 0.83687687926316 & 0.54739116631049 \\ 0 & -0.54739116631049 & 0.83687687926316 \end{pmatrix}$$

Πράγματι επαληθεύουμε ότι

$$U_1 D R_1 = \begin{pmatrix} -10.28197943606169 & -7.03700464476891 & -14.60905150426337 \\ 10.16272103702569 & 1.29283181177245 & 7.77087551861361 \\ 0 & -8.64308932786502 & -0.54338405065618 \end{pmatrix} = AA$$

και

$$U_1 B R_1 = \begin{pmatrix} -9.43398113205660 & 0.09038252219556 & -9.82045157830987 \\ & 0 & 7.24792590617632 & 3.21093831983032 \\ & 0 & & 0 & 3.96333280996892 \end{pmatrix} = CC$$

Συνάρτηση επίλυσης της περιορισμένης εξίσωσης Sylvester-παρατηρητή

Παρακάτω παρουσιάζουμε την συνάρτηση που επιλύει αριθμητικά την περιορισμένη εξίσωση Sylvester-παρατηρητή, ενώ γίνεται και εφαρμογή του παραδείγματος 3.7.1 για την εγγύτητα των αποτελεσμάτων.

```
function X = constrainedSylvEq(A,B,C,F)
% Αυτή η συνάρτηση μας λύνει την εξίσωση XA-FX=G*C έτσι ώστε να ισχυεί
% X*B=0 και ο πίνακας [ X ; C ] να έχει πλήρη τάξη n
%
% Λύνοντας την εξίσωση με αυτόν τον τρόπο το σύστημα ανάδρασης με τον
% ελαττωμένης τάξη παρατηρητή θα έχει τις ίδιες εύρωστες ιδιότητες με αυτές
% του ευθύ συστήματος ανάδρασης.
%
%----ΚΑΤΩΤΟΙΚΙΔΗΣ ΔΗΜΗΤΡΙΟΣ Χ.-----
[n1,n2]=size(A);
[n3,m]=size(B);
[r,n4]=size(C);
if (n1~=n2) || (n2~=n3) || (n3~=n4)
    disp('wrong dimensions A is nxn,B is nxm,C is rxn')
else
    n=n1;
    q1=n-r;
[q2,q3]=size(F); %q=n-r
    if (q1~=q2) || (q2~=q3)
```

**Αριθμητική επίλυση των εξισώσεων Sylvester – Η εξίσωση
Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab**

```

disp('wrong dimensions F is (n-r)x(n-r)')
else
    q=q1;
if (rank(observ(A,C))~=n)
    disp('the pair (A,C) is not observable')
elseif (rank(C*B)~=m) || (rank(B)~=m)
    disp('(rank(C*B)~=m) || (rank(B)~=m)')
elseif (r>=n) || (m>=n) || (r>=n)
    disp('(r>=n) || (m>=n) || (r>=n)')
else
[W,S]=qr(B);
W1=W(:,1:m);
W2=W(:,m+1:n);
A1=W2'*A*W1;
A2=W2'*A*W2;
C1=C*W1;
C2=C*W2;
[Q,R]=qr(C1);
R=R(1:m,1:m);
Q1=Q(:,1:m);
Q2=Q(:,m+1:r);
E=Q'*C2;
E1=E(1:m,:);
E2=E(m+1:r,:);

%-----
%-solve for J J*R=A1--
%-----

TrA1=A1';
TrR=R';
for i=1:(n-m)
    JJ=mygaussel(TrR,TrA1(:,i));
    Jtrans(:,i)=JJ;

```

```

end
J=Jtrans';
%-----

Ahat=A2-J*E1;
disp('n-r=')
disp(n-r)
disp('r-m=')
disp(r-m)
disp('G2 is (n-r)x(r-m)')
G2=input('please give a, correct in dimensions, desirable G2. G2=')
G2E2=G2*E2;
FF=-F;
Z=HessSch(FF,Ahat,G2E2);
G1=Z*J;
G=[G1 G2]*Q';
X=Z*W2';
end
end
end

```

Ας κάνουμε εφαρμογή στο παράδειγμα 3.7.1. Αν έχουμε τους πίνακες

$$A = \begin{pmatrix} -0.02 & 0.005 & 2.4 & -3.2 \\ -0.14 & 0.44 & -1.3 & -3 \\ 0 & 0.018 & -1.6 & 1.2 \\ 0 & 0 & 1 & 0 \end{pmatrix}, \quad B = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \end{pmatrix},$$

$$C = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 57.3 \end{pmatrix}, \quad \text{και} \quad F = \begin{pmatrix} -1 & 0 \\ 1 & -2 \end{pmatrix}.$$

Τότε θα έχουμε $n = 4, r = 2, m = 1$.

Αριθμητική επίλυση των εξισώσεων Sylvester – Η εξίσωση Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab

Τότε για G_2

$$G_2 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$$

θα έχουμε:

$$X = \begin{pmatrix} -0.10072035163287 & -0.70504246143007 & 0.92536823062697 & -0.11960541756403 \\ -0.07094522558486 & -0.28393710732251 & -0.61939366600667 & 0.97427599891404 \end{pmatrix}$$

ενώ το παράδειγμα μας μας έδινε λύση $X_{\text{paradeigmatos}}$

$$X_{\text{paradeigmatos}} = \begin{pmatrix} -0.1007 & -0.7050 & 0.9254 & -0.1196 \\ -0.0709 & -0.2839 & -0.6199 & 0.9743 \end{pmatrix}$$

Επίσης θα έχουμε

$$\text{rank} \begin{pmatrix} X \\ C \end{pmatrix} = 4$$

ενώ

$$XB = 10^{-16} \begin{pmatrix} -0.69388939039072 \\ 0 \end{pmatrix}$$

Συνάρτηση ανατοποθέτησης ιδιοτιμών μέσω της εξίσωσης Sylvester – παρατηρητή

Εδώ έχουμε δημιουργήσει μια συνάρτηση η οποία καταρχήν μας λύνει την εξίσωση $XT - AX = BG$, για τυχαίο G και επιθυμητό T , μέσω της συνάρτησης που έχουμε ήδη φτιάξει, την συνάρτηση **HessSch**, και εφόσον το ζεύγος πινάκων (A, B) είναι ελέγξιμο και το ζεύγος πινάκων (G, T) είναι παρατηρήσιμο μας δίνει μια nonsingular λύση X

που την χρησιμοποιούμε για να λύσουμε ως προς F το σύστημα $FX = G$. Ο πίνακας T θα πρέπει να έχει τις επιθυμητές ιδιοτιμές που θα θέλαμε να έχει το κλειστό σύστημα μας. Έτσι δημιουργούμε τον επιθυμητό T και τον εισαγάγουμε ως δεδομένο. Σαν έξοδο θα πάρουμε τον πίνακα ανάδρασης F έτσι ώστε το κλειστό μας σύστημα, δηλαδή ο πίνακας $A + BF$ να έχει τις επιθυμητές ιδιοτιμές.

Το σύστημα $FX = G$ επιλέγουμε να λυθεί με την συνάρτηση μας **mygaussel** για καλύτερα αποτελέσματα.

Στην συνάρτηση αυτήν εφαρμόζουμε ένα δικό μας παράδειγμα για να δείξουμε πως δουλεύει.

```
function F=SylvEqInPolePlacement(A,B,T,G)
% Αυτή η συνάρτηση βρίσκει από την εξίσωση  $X^*T - A^*X = B^*G$  την λύση  $X$  όπου
%  $T$  είναι ένας πίνακας τέτοιος ώστε να έχει τις επιθυμητές ιδιοτιμές που
% θέλουμε να έχει ο πίνακας  $A+B^*F$ , ενώ ο  $G$  είναι ένας τυχαίος πίνακας
% Ο πίνακας  $F$  θα δίνεται αν λύσουμε το σύστημα  $F^*X=G$ 
% Εν τέλει η συνάρτηση αυτή μας δίνει αυτόν το πίνακα  $F$ .
%-----ΚΑΤΩΤΟΙΚΙΔΗΣ ΔΗΜΗΤΡΙΟΣ Χ.-----
[n1,n2]=size(A);
[n3,m1]=size(B);
[n4,n5]=size(T);
[m2,n6]=size(G);
if (n1~=n2) || (n2~=n3) || (n3~=n4) || (n4~=n5) || (n5~=n6) || (m1~=m2)
    disp('wrong dimensions,A is nxn,B is nxm,T is nxn,G is mxn')
else
    n=n1;
    m=m1;
    BG=B*G;
    disp('rank (A,B)')
    Rab=rank(ctrb(A,B))
    disp('rank(G,T)')
    Rtg=rank(observ(G,T))
```

**Αριθμητική επίλυση των εξισώσεων Sylvester – Η εξίσωση
Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab**

```
if (Rab~=n)
    disp('the problem has no solution')
end
if (Rtg~=n)
    disp(' X is singular ')
end
X=HessSch(-A,T,BG);
disp('the eigenvalues of T are :')
disp(eig(T))
%-----
% επίλυση του F*X=G
%-----
[g1,g2]=size(G);
TrX=X';
TrG=G';
for i=1:g1
    xx=mygaussel(TrX,TrG(:,i));
    TrF(:,i)=xx;
end
F=TrF';
%-----
disp('F is:')
disp(F)
disp('the eigenvalues of A+BF are :')
disp(eig(A+B*F))
end
```

Αν έχουμε σαν είσοδο τους πίνακες

$$A = \begin{pmatrix} 3 & 19 & 2 \\ 18 & 2 & 3 \\ 1 & 5 & 3 \end{pmatrix}$$

$$B = \begin{pmatrix} 9.50129285147175 & 4.85982468709300 & 4.56467665168341 \\ 2.31138513574288 & 8.91298966148902 & 0.18503643248224 \\ 6.06842583541787 & 7.62096833027395 & 8.21407164295253 \end{pmatrix}$$

$$T = \begin{pmatrix} -1 & 0 & 0 \\ 5 & -2 & 0 \\ 0 & 7 & -3 \end{pmatrix}$$

$$G = \begin{pmatrix} 4.44703364353194 & 9.21812970744802 & 4.05706213062096 \\ 6.15432348100095 & 7.38207245810665 & 9.35469699107605 \\ 7.91937037427035 & 1.76266144494618 & 9.16904439913408 \end{pmatrix}$$

Συγκεκριμένα αν θέλουμε να δούμε τις ιδιοτιμές του συστήματος πριν την ανάδραση δεν έχουμε παρά να πάρουμε τις ιδιοτιμές του πίνακα A που εδώ είναι:

$$\text{eig}(A) = \begin{pmatrix} -16.12124632650563 \\ 21.78657182413286 \\ 2.33467450237278 \end{pmatrix}$$

Από την συνάρτηση θα έχουμε ότι το ζεύγος πινάκων (A, B) είναι ελέγξιμο και το ζεύγος πινάκων (G, T) είναι παρατηρήσιμο ενώ θα έχουμε και

$$F = \begin{pmatrix} -4.15053952638447 & 1.94734639883989 & -0.47953540453813 \\ 2.01381067015951 & -2.86069663820170 & 0.08679719179506 \\ 7.62057734327776 & -6.22232656901345 & 0.64029740887236 \end{pmatrix}$$

ενώ

$$\text{eig}(A + BF) = \begin{pmatrix} -0.99999999999947 \\ -2.00000000000147 \\ -2.99999999999906 \end{pmatrix} \cong \begin{pmatrix} -1 \\ -2 \\ -3 \end{pmatrix}$$

* * *

**Αριθμητική επίλυση των εξίσωσεων Sylvester – Η εξίσωση
Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab**

Επιλεγμένες συναρτήσεις (της Matlab)

Εκτός από τις κλασικές (βασικές) συναρτήσεις τις Matlab έχουμε χρησιμοποιήσει και τις παρακάτω συναρτήσεις :

lyap : Αν δοθεί ως $\text{lyap}(A, B, C)$ λύνει την εξίσωση Sylvester $AX + XB = -C$.

qz : Χρησιμοποιεί τον αλγόριθμο qz και μας ανάγει τον πίνακα A σε σχεδόν τριγωνική μορφή και τον B σε τριγωνική μορφή

qr : Χρησιμοποιεί τον αλγόριθμο qr και μας ανάγει τον πίνακα A σε τριγωνική μορφή

hess : Ανάγει έναν πίνακα σε μορφή Hessenberg

schur : Ανάγει έναν πίνακα σε μορφή Schur

lu : lu παραγοντοποίηση. Δοθέντος ενός πίνακα X μας δίνει πίνακες L, U έτσι ώστε να ισχύει $X = L*U$. L κάτω τριγωνικός πίνακας και U άνω τριγωνικός πίνακας.

ctrb : υπολογίζει τον πίνακα ελεγκσιμότητας

obsv : υπολογίζει τον πίνακα παρατηρησιμότητας

mat2cell : σε γενικές γραμμές με αυτήν την συνάρτηση μπορούμε να χωρίσουμε έναν πίνακα σε κελιά (block) πινάκων.

cell2mat : κάνει την αντίθετη ακριβώς δουλειά από την mat2cell συνάρτηση

planerot : συνάρτηση επίπεδης περιστροφής για ένα 2×1 διάνυσμα

nnz : αριθμός μη μηδενικών στοιχείων ενός πίνακα

triu : μας δίνει το άνω τριγωνικό μέρος ενός πίνακα

tril : μας δίνει το κάτω τριγωνικό μέρος ενός πίνακα

Δημήτριος Χ. Κατωτοκίδης

reshape : ανασχηματίζει έναν πίνακα

kron : μας δίνει το γινόμενο Kronecker 2 πινάκων

Για περισσότερες πληροφορίες για τις παραπάνω συναρτήσεις μπορεί κανείς να ανατρέξει στα help files της Matlab.

**Αριθμητική επίλυση των εξίσωσεων Sylvester – Η εξίσωση
Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab**

Συναρτήσεις που έφτιαξε ο συγγραφέας της διπλωματικής αυτής

discreteHessSch : συνάρτηση επίλυσης της διακριτής εξίσωσης Sylvester
 $BXA + C = X$

schrshr : συνάρτηση επίλυσης της εξίσωσης Sylvester $AX + XB = C$ που βασίζεται
στον αλγόριθμο Schur – Schur

uptrig : συνάρτηση επίλυσης της εξίσωσης Sylvester $AX + XB = C$ σύμφωνα με
τον τύπο

$$x_{ij} = \frac{c_{ij} - \sum_{k=i+1}^m a_{ik} x_{kj} - \sum_{k=1}^{j-1} x_{ik} b_{kj}}{a_{ii} + b_{jj}}$$

όπου $A = (a_{ij})$ και $B = (b_{ij})$ είναι άνω τριγωνικοί πίνακες διάστασης
 $m \times m$ και $n \times n$ αντίστοιχα .

genHessSch : συνάρτηση επίλυσης της γενικευμένης εξίσωσης Sylvester που
βασίζεται στον αλγόριθμο Hessenberg – Schur

myqzpart1 : Θεωρείται στην ουσία το πρώτο βήμα του αλγορίθμου QZ. Η
συνάρτηση αυτή παίρνει ως πούμε το ζευγάρι πινάκων (A,B) και
επιστρέφει το ζευγάρι πινάκων ως πούμε (HA,TB) όπου HA είναι ένας
πίνακας σε Hessenberg μορφή και TB ένας πίνακας σε άνω τριγωνική
μορφή. Η συνάρτηση αυτή μας δίνει και πίνακες Q, Z, έτσι ώστε να
ισχύει ταυτοχρόνως $HA=Q'*A*Z$ και $TB=Q'*B*Z$.

SchrblockSchr : συνάρτηση επίλυσης της εξίσωσης Sylvester που
βασίζεται στον αλγόριθμο των Bartels-Stewart

SylvEqInPolePlacement : συνάρτηση ανατοποθέτησης ιδιοτιμών μέσω της εξίσωσης
Sylvester-παρατηρητή

constrainedSylvEq : επιλύει την περιορισμένη εξίσωση Sylvester-παρατηρητή

genSchSch : συνάρτηση επίλυσης της γενικευμένης εξίσωσης Sylvester που
βασίζεται στον αλγόριθμο των Bartels - Stewart

HessSch : συνάρτηση επίλυσης της εξίσωσης Sylvester που
βασίζεται στον αλγόριθμο Hessenberg – Schur

mygaussel : συνάρτηση επίλυση του συστήματος $AX = B$ που βασίζεται στην
μέθοδο απαλοιφής Gauss με μερική οδήγηση.

ΒΙΒΛΙΟΓΡΑΦΙΑ

- Barlow J.B., Monahemi M.M., and O’Leary D.P. “Constrained matrix Sylvester Equations,” *SIAM J. Matrix Anal. Appl.*, Vol. 13, pp.1-9, 1992.
- Bartels R.H. and Stewart G.W. “Algorithm 432: solution of the matrix equation $AX + XB = C$,” *Comm. ACM*, Vol. 15, pp. 820-826, 1972
- Bhattacharyya S.P. and Desouza E, “Controllability, observability and the solution of $AX - XB = C$,” *Lin. Alg. Appl.*, Vol. 39, pp.167-188, 1981.
- Bhattacharyya S.P. and Desouza E, “Pole assignment via Sylvester equation”, *Systems Control Lett.* 1(4):261-263 (1982).
- Chen C.-T. *Linear System Theory and Design*, CBS College Publishing, New York, 1984
- Γουσίδου-Κουτίτα Μ., *Αριθμητικές Μέθοδοι με Εφαρμογές στην Θεωρία Ελέγχου (Στοιχεία από το βιβλίο Computational Methods for Linear Control Systems των Petkov, Christov και Konstantinov)*. Θεσσαλονίκη, 2002.
- Datta B. N., *Linear and numerical linear algebra in control theory: Some research problems*, *Linear Algebra Appl.* 197/198:755-790 1994.
- Datta B. N., *Numerical Methods for Linear Control Systems Design and Analysis*. Illinois, 2003.
- Gardiner, J. D., Laub, A. L., Amato, J. A., and Moler, C. B. 1992b. “Solution of the Sylvester matrix equation $AXB^T + CXD^T = E$.” *ACM Trans. Math. Softw.* 18, 2, pp 223–231.
- Golub G.H., Nash S., and Van Loan C.F. “A Hessenberg-Schur method for the problem $AX + XB = C$,” *IEEE Trans. Autom. Control*, Vol. AC-24, pp. 909-913, 1979
- Higham N.J.(1996), *Accuracy and Stability of Numerical Algorithms*, SIAM Philadelphia.
- Luenberger D.G. (1964), “Observing the state of a linear system,” *IEEE Trans. Mil. Electr.*, Vol. 8, 74-80.
- Luenberger D.G. (1966), “Observers for multivariable systems,” *IEEE Trans. Autom. Control*, Vol. 11, pp. 190-197.
- Luenberger D.G. (1971), “An introduction to observers,” *IEEE Trans. Autom.*

**Αριθμητική επίλυση των εξίσωσεων Sylvester – Η εξίσωση
Sylvester στη θεωρία έλεγχου – Εφαρμογές με Matlab**

Control, Vol. AC-16, pp. 596-602.

Luenberger D.G. (1979), *Introduction to Dynamic Systems; Theory, Models, and Applications*, John Wiley & Sons, New York.

Matlab 6.5 - Help Files

Petkov P., Christov N.D., and Konstantinov M.M. *Computational Methods for Linear Control Systems*, Prentice Hall, London, 1991.

Tsui C.-C. (1988), “A new approach to robust observer design,” *Int. J. Control*, Vol. 47, pp.745-751.

ΧΡΗΣΙΜΑ SITES

www.ams.org/mathscinet/search

anemos.math.auth.gr/datta.zip

www.springerlink.com

* * *