



**ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ
ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ
ΔΙΑΤΜΗΜΑΤΙΚΟ ΠΡΟΓΡΑΜΜΑ
ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ
ΠΛΗΡΟΦΟΡΙΚΗ ΚΑΙ ΥΠΟΛΟΓΙΣΤΙΚΗ ΒΙΟΙΑΤΡΙΚΗ**

**Αλγόριθμοι φασματικής ανάλυσης σε πίνακες μοριακών
δεδομένων μεγάλης διάστασης**

Δέσποινα Δ. Καβαλιέρου

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

Επιβλέπουσα

Μαρία Αδάμ

Καθηγήτρια Π.Θ.

Λαμία, 2025



UNIVERSITY OF THESSALY
SCHOOL OF SCIENCE
INFORMATICS AND COMPUTATIONAL
BIOMEDICINE

**Algorithms for the spectral analysis
of high-dimensional molecular data**

Despoina D. Kavalierou

Master thesis

Supervisor: Maria Adam
Professor UTH

Lamia, 2025



**ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ
ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ
ΔΙΑΤΜΗΜΑΤΙΚΟ ΜΕΤΑΠΤΥΧΙΑΚΟ ΠΡΟΓΡΑΜΜΑ
ΠΛΗΡΟΦΟΡΙΚΗ ΚΑΙ ΥΠΟΛΟΓΙΣΤΙΚΗ ΒΙΟΙΑΤΡΙΚΗ
ΚΑΤΕΥΘΥΝΣΗ**

«ΥΠΟΛΟΓΙΣΤΙΚΗ ΙΑΤΡΙΚΗ ΚΑΙ ΒΙΟΛΟΓΙΑ»

**Αλγόριθμοι φασματικής ανάλυσης σε πίνακες μοριακών
δεδομένων μεγάλης διάστασης**

Δέσποινα Δ. Καβαλιέρου

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

Επιβλέπουσα

Μαρία Αδάμ

Καθηγήτρια Π.Θ.

Λαμία, 2025

«Υπεύθυνη Δήλωση μη λογοκλοπής και ανάληψης προσωπικής ευθύνης»

Με πλήρη επίγνωση των συνεπειών του νόμου περί πνευματικών δικαιωμάτων, και γνωρίζοντας τις συνέπειες της λογοκλοπής, δηλώνω υπεύθυνα και ενυπογράφως ότι η παρούσα εργασία με τίτλο «Αλγόριθμοι φασματικής ανάλυσης σε πίνακες μοριακών δεδομένων μεγάλης διάστασης» αποτελεί προϊόν αυστηρά προσωπικής εργασίας και όλες οι πηγές από τις οποίες χρησιμοποίησα δεδομένα, ιδέες, φράσεις, προτάσεις ή λέξεις, είτε επακριβώς (όπως υπάρχουν στο πρωτότυπο ή μεταφρασμένες) είτε με παράφραση, έχουν δηλωθεί κατάλληλα και ευδιάκριτα στο κείμενο με την κατάλληλη παραπομπή και η σχετική αναφορά περιλαμβάνεται στο τμήμα των βιβλιογραφικών αναφορών με πλήρη περιγραφή. Αναλαμβάνω πλήρως, ατομικά και προσωπικά, όλες τις νομικές και διοικητικές συνέπειες που δύναται να προκύψουν στην περίπτωση κατά την οποία αποδειχθεί, διαχρονικά, ότι η εργασία αυτή ή τμήμα της δεν μου ανήκει διότι είναι προϊόν λογοκλοπής.

Η ΔΗΛΟΥΣΑ

Ημερομηνία

Υπογραφή

**Αλγόριθμοι φασματικής ανάλυσης σε πίνακες μοριακών
δεδομένων μεγάλης διάστασης**

**Algorithms for the spectral analysis
of high-dimensional molecular data**

Δέσποινα Δ. Καβαλιέρου

Τριμελής Επιτροπή:

Μαρία Αδάμ, Καθηγήτρια Τμήματος Πληροφορικής με Εφαρμογές στη Βιοϊατρική του
Π.Θ. (επιβλέπουσα)

Σπυρίδων Γεωργακόπουλος, Επίκουρος Καθηγητής, Τμήματος Μαθηματικών του Π.Θ.

Παναγιώτα Κοντού, Επίκουρος Καθηγήτρια, Τμήματος Μαθηματικών του Π.Θ.

Ευχαριστίες

Θα ήθελα να ευχαριστήσω θερμά την επιβλέπουσα καθηγήτρια, κ. Μαρία Αδάμ, αρχικά γιατί δέχθηκε να αναλάβει την επίβλεψη της παρούσας διπλωματικής, κυρίως όμως διότι η καθοδήγηση της υπήρξε καθοριστική σε όλα τα στάδια της: επιλογή θέματος, μελέτη αντικειμένου, επεξεργασία στοιχείων και εξαγωγή συμπερασμάτων.

Τα μέλη της τριμελούς επιτροπής, τους ευχαριστώ για τον πολύτιμο χρόνο που αφιέρωσαν στη μελέτη της διπλωματικής καθώς και για τις παρατηρήσεις και τα σχόλια τους που συνέβαλλαν στη διαμόρφωση της τελικής μορφής της.

Θα ήταν παράλειψη να μην ευχαριστήσω όλους τους διδάσκοντες που είχα κατά τη διάρκεια των μεταπτυχιακών μου σπουδών, καθώς η παρούσα διπλωματική αποτέλεσε το επιστέγασμα όλων των γνώσεων που μου προσέφεραν. Επίσης, θα ήθελα να ευχαριστήσω τους συναδέλφους μεταπτυχιακούς φοιτητές, για την ενθάρρυνση καθώς και για τις ανταλλαγές απόψεων και τη σημαντική βοήθειά τους σε όλα τα στάδια των κοινών σπουδών μας.

Τέλος, δε μπορώ να μην αναφερθώ στην οικογένεια μου που ήταν δίπλα μου σε όλη την διάρκεια των μεταπτυχιακών μου σπουδών. Τους χρωστάω ένα μεγάλο Ευχαριστώ!

Περίληψη

Η παρούσα μεταπτυχιακή διπλωματική εργασία επικεντρώνεται στη χρήση υπολογιστικών μεθόδων για τη μελέτη των ιδιοτιμών και ιδιοδιανυσμάτων ενός συμμετρικού πίνακα αποστάσεων, ο οποίος προκύπτει από μοριακά δεδομένα τοποθετημένα σε πίνακα μεγάλων διαστάσεων. Στόχος της μελέτης είναι η κατανόηση της γεωμετρικής δομής και της δυναμικής συμπεριφοράς των πρωτεϊνικών μορίων. Η μελέτη των ιδιοτιμών συμβάλλει στην κατανόηση των φυσικών ιδιοτήτων των πρωτεϊνών, ενώ η μελέτη των ιδιοδιανυσμάτων αποκαλύπτει τους βασικούς τρόπους κατανομής των ατόμων στον τρισδιάστατο χώρο, δίνοντας πληροφορίες για την κινητική συμπεριφορά των μοριακών δομών.

Η εργασία περιλαμβάνει τη θεωρητική ανάλυση τεσσάρων αλγορίθμων: του κλασικού αλγορίθμου Lanczos, δύο βελτιωμένων εκδόσεών του, καθώς και ενός νέου, προτεινόμενου αλγορίθμου. Όλοι οι αλγόριθμοι υλοποιήθηκαν σε περιβάλλον Matlab και αξιολογήθηκαν συγκριτικά ως προς την απόδοσή τους. Βασικό χαρακτηριστικό των μεθόδων αυτών είναι η κατασκευή ενός τριδιαγώνιου πίνακα μικρότερης διάστασης από τον αρχικό πίνακα αποστάσεων, του οποίου οι ιδιοτιμές αποτελούν προσεγγίσεις των ιδιοτιμών του αρχικού πίνακα.

Τα αποτελέσματα της μελέτης δείχνουν ότι ο προτεινόμενος αλγόριθμος υπερέχει έναντι των υπολοίπων, επιτυγχάνοντας τον μικρότερο χρόνο εκτέλεσης. Παράλληλα, προσφέρει ιδιαίτερα ακριβείς προσεγγίσεις των ακραίων ιδιοτιμών (φασματικής ακτίνας και ελάχιστης ιδιοτιμής) του πίνακα αποστάσεων, υπό την προϋπόθεση ότι επιλέγεται κατάλληλη διάσταση για τον τριδιαγώνιο πίνακα. Με τη σωστή επιλογή διάστασης, το σφάλμα της προσέγγισης είναι της τάξης μεγέθους τουλάχιστον 10^{-10} .

Λέξεις-κλειδιά:

Ιδιοτιμές και ιδιοδιανύσματα, Αλγόριθμος Lanczos, Αριθμητική ακρίβεια, Υλοποίηση σε Matlab, Πίνακας απόστασης πρωτεϊνών, Φασματική προσέγγιση, Υπολογιστική αποδοτικότητα.

Abstract

This master thesis investigates computational methods for the spectral analysis of large symmetric distance matrices derived from molecular data, with particular emphasis on protein structures. The primary objective is to elucidate the geometric configuration and dynamic behavior of proteins by examining the eigenvalues and eigenvectors of these matrices. The spectral information provides critical insights into the intrinsic physical properties of proteins, while the corresponding eigenvectors reveal principal atomic displacement patterns in three-dimensional space, contributing to a deeper understanding of molecular motion and flexibility.

The study presents a comprehensive theoretical analysis of four algorithms: the classical Lanczos algorithm, two improved variants, and a newly proposed algorithm. All methods were implemented in Matlab and their computational efficiency and accuracy were systematically evaluated. The core computational strategy involves the construction of a tridiagonal matrix of reduced dimension, whose eigenvalues serve as approximations to those of the original distance matrix.

The study's results show that the proposed algorithm outperforms the others, achieving the shortest execution time. In addition, it provides highly accurate approximations of the extreme eigenvalues (spectral radius and smallest eigenvalue) of the distance matrix, provided that an appropriately sized tridiagonal matrix is selected. When the dimension of the tridiagonal matrix is properly chosen, the approximation error is on the order of at least 10^{-10} .

Keywords :

Eigenvalues and Eigenvectors, Lanczos algorithm, Numerical Accuracy, Matlab implementation, Protein distance matrix, Spectral Approximation, Computational Efficiency.

Περιεχόμενα

Περίληψη	8
Abstract.....	9
Περιεχόμενα.....	10
Πρόλογος.....	11
Κεφάλαιο 1°	15
Εισαγωγικές Έννοιες	15
1.1 Μέθοδος των Δυνάμεων	15
1.1.1 Αλγόριθμος	15
1.1.2 Εφαρμογή της Μεθόδου των Δυνάμεων.....	25
1.2 Μέθοδοι υποχώρων.....	28
1.2.1 Επανάληψη Υποχώρου (Subspace Iteration).....	28
1.2.2 Μέθοδος υποχώρων Krylov.....	28
Κεφάλαιο 2°	38
Αλγόριθμος Lanczos.....	38
2.1 Προέλευση αλγορίθμου και εφαρμογές.....	38
2.2 Κλασσικός Αλγόριθμος Lanczos.....	40
2.3 Βελτιωμένοι Αλγόριθμοι Lanczos	49
Κεφάλαιο 3°	56
Προσέγγιση ιδιοποσών του πίνακα αποστάσεων και Κλασσικός Αλγόριθμος Lanczos	56
3.1 Φασματική ανάλυση με τον Κλασσικό Αλγόριθμο Lanczos	56
3.2 Συναρτήσεις (functions).....	61
3.2.1 Συνάρτηση Lanczos_classic.....	61
3.2.2 Συνάρτηση analyze_eigenvectors.....	64
3.2.4 Συνάρτηση visualize_all	65
3.2.3 Συνάρτηση compare_eigenvalues	70
Κεφάλαιο 4°	78
Σύγκριση τεσσάρων αλγορίθμων	78
4.1 Χρόνος εκτέλεσης αλγορίθμων.....	78
4.2 Ανάλυση σφαλμάτων προσέγγισης των ιδιοτιμών	81
Συμπεράσματα	102
Βιβλιογραφία	104
Ιστοσελίδες	105
Παράρτημα Ι	107
Παράρτημα ΙΙ.....	113
Παράρτημα ΙΙΙ.....	121

Πρόλογος

Η αρχική ιδέα για την παρούσα διπλωματική εργασία προέκυψε από το ενδιαφέρον για το πώς η επίλυση ιδιοπροβλημάτων μπορεί να αξιοποιηθεί στον τομέα της βιοπληροφορικής, συμβάλλοντας στην εξαγωγή ουσιαστικών βιολογικών συμπερασμάτων. Το έναυσμα αυτό δόθηκε από τη μελέτη σχετικών επιστημονικών εργασιών, οι οποίες ανέδειξαν τη σημασία της φασματικής ανάλυσης ως εργαλείο για την ερμηνεία πολύπλοκων μοριακών δεδομένων τοποθετημένων σε πίνακες μεγάλης διάστασης [10], [11], [18], [19], [20]. Ενδεικτικά, οι συγγραφείς στην εργασία [10] μελετώντας πίνακες αποστάσεων μέσω της φασματικής ανάλυσης προβλέπουν τη δομή πρωτεϊνών, καταδεικνύοντας ότι η φασματική ανάλυση ενός πίνακα αποστάσεων μπορεί να παράγει πληροφορίες για βασικά χαρακτηριστικά της δευτεροταγούς και τριτοταγούς δομής τους. Παρότι η προσέγγιση αυτή παρουσιάζει σημαντικό ερευνητικό ενδιαφέρον, εστιάζει σε περιορισμένο φάσμα πληροφορίας και παραβλέπει πιο σύνθετες, αλλά κρίσιμης σημασίας, σχέσεις που επηρεάζουν τη λειτουργία της πρωτεΐνης. Αντίστοιχα, στην εργασία [11] επιχειρείται η αντιμετώπιση του προβλήματος επίλυσης ιδιοπροβλημάτων σε πολύ μεγάλα συστήματα μέσω coarse-grained προσεγγίσεων, όπου οι λεπτομέρειες μικρής κλίμακας αγνοούνται προκειμένου να επιτευχθεί μεγαλύτερη υπολογιστική αποδοτικότητα. Αν και η μεθοδολογία αυτή μειώνει σημαντικά το υπολογιστικό κόστος, επικεντρώνεται αποκλειστικά στην αναζήτηση της αποδοτικότερης αριθμητικής επίλυσης του προβλήματος, χωρίς να παρέχει βιολογική ερμηνεία. Στην εργασία [18] προτείνεται ένα μοντέλο φασματικής ομαδοποίησης (multi-way spectral clustering) για την ανάλυση δικτύων, το οποίο χρησιμοποιεί έναν μικρό αριθμό ιδιοτιμών ώστε να εντοπίσει σύνθετα μοτίβα σύνδεσης σε βιολογικά και κοινωνικά δίκτυα. Αν και η μέθοδος αυτή διευκολύνει την κατανόηση της δομής των δικτύων, εστιάζει κυρίως στις συνδέσεις μεταξύ κόμβων και δεν λαμβάνει υπόψη σημαντικές πληροφορίες, όπως οι δυναμικές κινήσεις ή οι φυσικοχημικές ιδιότητες που μπορούν να προκύψουν μέσω της φασματικής ανάλυσης. Στο ίδιο πλαίσιο, η εργασία [19] παρουσιάζει το εργαλείο iGNM, το οποίο δημιουργεί μια βάση δεδομένων GNM-ανάλυσης και παρέχει πληροφορίες για τις συλλογικές κινήσεις χιλιάδων πρωτεϊνικών δομών. Ωστόσο, η μέθοδος βασίζεται σε συγκεκριμένα απλοποιημένα μοντέλα και δεν προσφέρει ευελιξία προσαρμογής σε πρωτεΐνες με πολύπλοκη ή ανομοιογενή δομή. Επίσης, στην [20] παρουσιάζεται ένας αλγόριθμος Subspace Iteration (επανάληψη υποχώρου) για την ανάλυση των βασικών τρόπων κίνησης των πρωτεϊνών, επιτυγχάνοντας σημαντική μείωση του χρόνου και του υπολογιστικού κόστους. Παρ' όλα αυτά, η προσέγγιση αυτή δεν εξηγεί με σαφήνεια πώς τα φασματικά αποτελέσματα συνδέονται με τη λειτουργία της πρωτεΐνης σε πραγματικές βιολογικές συνθήκες.

Οι παραπάνω ερευνητικές εργασίες συνέβαλαν καθοριστικά στην κατανόηση του υφιστάμενου επιστημονικού πεδίου και στον εντοπισμό των κύριων ερευνητικών κενών. Καμία από τις εξεταζόμενες μεθοδολογίες δεν καλύπτει επαρκώς την ανάγκη για φασματική ανάλυση, που να συνδυάζει υψηλή υπολογιστική αποδοτικότητα, δυνατότητα εφαρμογής σε ετερογενή και πολύπλοκα βιολογικά δίκτυα, καθώς και ουσιαστική, βιολογικά ερμηνεύσιμη σύνδεση μεταξύ των ιδιοποσών (ιδιοτιμών και ιδιοδιανυσμάτων

κατάλληλου πίνακα) και λειτουργικών ιδιοτήτων των πρωτεϊνικών δομών. Αυτοί οι περιορισμοί αποτέλεσαν τη βάση για την ανάπτυξη της συγκεκριμένης εργασίας, η οποία στοχεύει στη βελτιστοποίηση των υπάρχουσών μεθόδων και στην ανάπτυξη νέας μεθόδου φασματικής ανάλυσης, ώστε αυτές να συμβάλλουν στην ταχύτερη ερμηνεία ιδιοπροβλημάτων, που θέτει η βιοπληροφορική.

Προς αυτήν την κατεύθυνση, είναι κρίσιμο να μελετηθεί το μαθηματικό υπόβαθρο της φασματικής ανάλυσης των πινάκων, και να κατανοηθούν θέματα που σχετίζονται με την αριθμητική επίλυση ιδιοπροβλημάτων μη αρνητικών πινάκων. Ένα από τα σημαντικότερα θέματα που διαπραγματεύεται η Αριθμητική Γραμμική Άλγεβρα είναι ο υπολογισμός των ιδιοτιμών, λ , για τετραγωνικούς $n \times n$ πίνακες A , δηλαδή η εύρεση των $n \times 1$ μη τετριμμένων λύσεων x της εξίσωσης $Ax = \lambda x \Leftrightarrow (\lambda I_n - A)x = 0$, όπου I_n σημειώνει το μοναδιαίο $n \times n$ πίνακα. Η τελευταία εξίσωση είναι γραμμική ως προς το άγνωστο διάνυσμα x , όταν η τιμή του λ είναι γνωστή. Ωστόσο, όταν αναζητούνται ταυτόχρονα οι τιμές των λ και x , το πρόβλημα γίνεται πολύπλοκο, εξαιτίας της αλληλεξάρτησης των δύο μεταβλητών, με αποτέλεσμα να καταλήγουμε σε ένα σύνθετο υπολογιστικό πρόβλημα. Ο υπολογισμός των ιδιοτιμών πραγματοποιείται μέσω της επίλυσης της χαρακτηριστικής εξίσωσης $\det(\lambda I_n - A) = 0$. Λαμβάνοντας υπόψιν την πολυπλοκότητα επίλυσης μιας πολυωνυμικής εξίσωσης μεγάλου βαθμού, αναζητούνται αριθμητικές μέθοδοι εύρεσης των ριζών, μέθοδος του Νεύτωνα, μέθοδος διχοτόμησης και άλλες, οι οποίες να μην παρουσιάζουν αποκλίσεις από τις αναμενόμενες τιμές και να μην είναι αριθμητικά χρονοβόρες. Οι μέθοδοι για την επίλυση του ιδιοπροβλήματος ενός πίνακα χωρίζονται σε δύο κατηγορίες: αυτές που μεταβάλλουν τον αρχικό πίνακα μέσω διαδοχικών μετασχηματισμών και αυτές που διατηρούν τον αρχικό πίνακα αναλλοίωτο. Στην πρώτη κατηγορία (μεταβολή πίνακα) ανήκουν:

- ο Αλγόριθμος Francis
- ο Μέθοδος Διαίρει και Βασίλευε
- ο Μέθοδος Διχοτόμησης για Ιδιοτιμές
- ο Αλγόριθμος Jacobi
- ο Αλγόριθμος Laguerre
- ο Αλγόριθμος MRRR

Στη δεύτερη κατηγορία (αρχικός πίνακας αναλλοίωτος) ανήκουν:

- ο Μέθοδος των Δυνάμεων (Power Method)
- ο Επανάληψη υποχώρου (Subspace Iteration)
- ο Μέθοδοι υποχώρων Krylov

Οι μέθοδοι, που διατηρούν τον αρχικό πίνακα αναλλοίωτο, είναι εξαιρετικά σημαντικές για την επίλυση ιδιοπροβλημάτων μεγάλης κλίμακας. Αυτές οι μέθοδοι είναι προτιμότερες, επειδή διατηρούν την αραιότητα και τη δομή του αρχικού πίνακα, καθιστώντας τις πιο αποδοτικές σε υπολογιστικό επίπεδο.

Η παρούσα διπλωματική εργασία για την αναζήτηση ιδιοτιμών και ιδιοδιανυσμάτων εστιάζει σε μεθόδους που ανήκουν στη δεύτερη προαναφερόμενη κατηγορία, όπως είναι ο αλγόριθμος Lanczos και ο αλγόριθμος που αναπτύχθηκε στο πλαίσιο αυτής της εργασίας, έχοντας ως κίνητρο τη συμβολή της αναπαράστασης των ιδιοτιμών και των

αντίστοιχων ιδιοδιανυσμάτων ενός συμμετρικού πίνακα αποστάσεων στην εξαγωγή χρήσιμων βιολογικών συμπερασμάτων. Ειδικότερα, από τις $n \times 3$ χωρικές συντεταγμένες των ατόμων των πρωτεϊνικών δομών ορίζεται ένας $n \times n$ τετραγωνικός πίνακας αποστάσεων, από τον οποίο παράγεται μικροτέρων διαστάσεων $m \times m$ τριδιαγώνιος πίνακας και μέσω της φασματικής ανάλυσης του τριδιαγώνιου πίνακα, παρέχονται πληροφορίες για τη χωρική οργάνωση, τη συνοχή και τη λειτουργική διαφοροποίηση της πρωτεΐνης.

Η αναπαράσταση των ιδιοδιανυσμάτων που αντιστοιχούν στη μεγαλύτερη ιδιοτιμή ανέδειξε γεωμετρικά μοτίβα, ενώ η ερμηνεία των μεγαλύτερων κατά απόλυτη τιμή ιδιοτιμών επέτρεψε την αναγνώριση δομικών και λειτουργικών αλληλεπιδράσεων. Βασικό κριτήριο για την επιλογή της μεθόδου αποτέλεσε η εύρεση του αποδοτικότερου αλγορίθμου, τόσο ως προς τον χρόνο εκτέλεσης όσο και ως προς τον υπολογισμό των ιδιοποσών του τριδιαγώνιου πίνακα για τη βέλτιστη προσέγγιση των ιδιοτιμών του πίνακα αποστάσεων. Για την επίτευξη αυτού του στόχου, ο προτεινόμενος αλγόριθμος συγκρίνεται με τον κλασσικό αλγόριθμο Lanczos και δύο βελτιωμένες εκδόσεις του.

Η παρούσα διπλωματική εργασία οργανώνεται σε τέσσερα κεφάλαια, το καθένα εκ των οποίων επικεντρώνεται σε διαφορετικές πτυχές του θέματος. Στο Κεφάλαιο 1 παρουσιάζονται τα θεωρητικά θεμέλια της φασματικής ανάλυσης, με έμφαση στη Μέθοδο των Δυνάμεων και την έννοια των υποχώρων Krylov. Αναλύονται οι βασικές έννοιες των μεθόδων αυτών, καθώς και οι μαθηματικές τους προσεγγίσεις, προκειμένου να γίνει κατανοητός ο ρόλος τους στην προσέγγιση ιδιοτιμών και ιδιοδιανυσμάτων ενός τετραγωνικού πίνακα.

Στο Κεφάλαιο 2 μελετάται το θεωρητικό και αλγοριθμικό υπόβαθρο του κλασσικού αλγορίθμου Lanczos (Classic Algorithm Lanczos) και των βελτιωμένων εκδόσεών του (Modified Lanczos Algorithm και Reorthogonalization Lanczos Algorithm) και δίνονται εφαρμογές. Παρουσιάζονται τα πλεονεκτήματα των μεθόδων που αναπτύσσονται, σε προβλήματα που μοντελοποιούνται σε συμμετρικούς μη αρνητικούς πίνακες μεγάλης διάστασης και εξετάζονται θέματα αριθμητικής ακρίβειας των αποτελεσμάτων, που συνδέονται με την ορθογωνοποίηση των διανυσμάτων.

Το Κεφάλαιο 3 είναι αφιερωμένο στην έρευνα και την υλοποίηση που πραγματοποιήθηκε στο πλαίσιο της παρούσας διπλωματικής εργασίας. Βασιζόμενοι στον κλασσικό αλγόριθμο Lanczos, παρουσιάζονται αναλυτικά οι συναρτήσεις που αναπτύχθηκαν και υλοποιήθηκαν σε περιβάλλον Matlab για την προσέγγιση των ιδιοτιμών και ιδιοδιανυσμάτων των πινάκων αποστάσεων μεγάλης διάστασης. Στο ίδιο κεφάλαιο προτείνεται ένας νέος αλγόριθμος, ο οποίος συνδυάζει τα πλεονεκτήματα του κλασσικού αλγορίθμου Lanczos με εκείνα της Μεθόδου των Δυνάμεων. Μοριακά δεδομένα που καταχωρούνται σε πίνακες μεγάλων διαστάσεων χρησιμοποιούνται για να υλοποιηθεί ο προτεινόμενος αλγόριθμος και να γίνει η φασματική ανάλυση κατάλληλων τριδιαγώνιων πινάκων, που παράγονται από συμμετρικούς πίνακες αποστάσεων μεγάλης διάστασης.

Στο Κεφάλαιο 4 παρουσιάζεται η συγκριτική αξιολόγηση των αποτελεσμάτων που προέκυψαν από την υλοποίηση του κλασσικού αλγορίθμου και των δύο βελτιωμένων εκδόσεών του, του Modified Lanczos Algorithm και του Reorthogonalization Lanczos Algorithm, σε σύγκριση με τον προτεινόμενο αλγόριθμο (Proposed Algorithm). Η εν

λόγω αξιολόγηση σχετίζεται με τον χρόνο εκτέλεσης των υλοποιήσεων και με την ακρίβεια προσέγγισης των ιδιοτιμών του αρχικού πίνακα αποστάσεων, μέσω των ιδιοτιμών που υπολογίζονται από τους αντίστοιχους τριδιαγώνιους πίνακες.

Η ερευνητική διαδικασία ολοκληρώνεται με την παρουσίαση των συμπερασμάτων που προέκυψαν από αυτήν. Γίνεται αποτίμηση της αποτελεσματικότητας των υπό μελέτη αλγορίθμων, επισημαίνοντας τα πλεονεκτήματά και τους περιορισμούς τους, προτείνοντας πιθανές κατευθύνσεις για μελλοντική έρευνα στον τομέα της φασματικής ανάλυσης πινάκων μεγάλης διάστασης, οι οποίοι προέρχονται από μοριακά δεδομένα. Στο τέλος της διπλωματικής εργασίας ακολουθεί η βιβλιογραφία με τα σχετικά συγγράμματα, οι ιστοσελίδες που χρησιμοποιήθηκαν, καθώς και τα Παραρτήματα I, II και III.

Επιπλέον, ο αναγνώστης θα πρέπει να λάβει υπόψη του ότι κάθε κεφάλαιο αυτής της διπλωματικής εργασίας αριθμείται και υποδιαιρείται σε ενότητες οι οποίες αριθμούνται με δύο ή τρεις αριθμούς ανάλογα με την υποδιαίρεση. Ο πρώτος αριθμός αντιπροσωπεύει το κεφάλαιο, ο δεύτερος την ενότητα και ο τρίτος αριθμός την υποενότητα. Επιπλέον, τα ακόλουθα στοιχεία αριθμούνται με σύστημα διπλής αρίθμησης, κατά το οποίο ο πρώτος αριθμός αντιστοιχεί στο κεφάλαιο και ο δεύτερος στον αύξοντα αριθμό εμφάνισής τους: (i) μαθηματικές σχέσεις, (ii) Ορισμοί, Αλγόριθμοι, Κώδικες, Παραδείγματα, Παρατηρήσεις, καθώς και τα Σχόλια των Παραδειγμάτων, και (iii) Σχήματα και Πίνακες σύγκρισης.

Κεφάλαιο 1^ο

Εισαγωγικές Έννοιες

1.1 Μέθοδος των Δυνάμεων

1.1.1 Αλγόριθμος

Η Μέθοδος των Δυνάμεων είναι μια απλή και αποτελεσματική αριθμητική μέθοδος, η οποία χρησιμοποιείται για τον υπολογισμό της μεγαλύτερης (κατά απόλυτη τιμή) ιδιοτιμής και του αντίστοιχου ιδιοδιανύσματος ενός τετραγωνικού πίνακα με πραγματικά στοιχεία, που στη συνέχεια ο πίνακας σημειώνεται $A \in \mathbb{R}^{n \times n}$, [6], [7], [17], [22], [26], [27], [30]. Εφαρμόζεται σε συμμετρικούς και μη συμμετρικούς πίνακες, αν και στην περίπτωση των μη συμμετρικών πινάκων μπορεί να επιστρέψει ζεύγος συζύγων μιγαδικών ιδιοτιμών αντί για τη μέγιστη πραγματική ιδιοτιμή.

Έστω ένας τυχαίος πίνακας $A \in \mathbb{R}^{n \times n}$ με n γραμμικά ανεξάρτητα ιδιοδιανύσματα u_i τα οποία αποτελούν μια βάση του $\mathbb{R}^n$ και λ_i , $i = 1, 2, \dots, n$ οι αντίστοιχες ιδιοτιμές για τις οποίες ισχύει ότι:

$$|\lambda_1| \geq |\lambda_2| \geq \dots \geq |\lambda_n| \quad (1.1)$$

Αν $v_0 \neq 0$ ένα διάνυσμα του $\mathbb{R}^n$, τότε επειδή τα ιδιοδιανύσματα του A αποτελούν μια βάση το v_0 μπορεί να γραφεί ως γραμμικός συνδυασμός

$$v_0 = c_1 u_1 + c_2 u_2 + \dots + c_n u_n = \sum_{i=1}^n c_i u_i \quad (1.2)$$

όπου οι συντελεστές c_i , $i = 1, 2, \dots, n$, είναι πραγματικοί αριθμοί όχι όλοι μηδέν. Στη συνέχεια επιλέγουμε αυθαίρετα ένα αρχικό διάνυσμα v_0 , μέσω της (1.2), οπότε σύμφωνα με τον ορισμό της ιδιοτιμής του πίνακα, $Ax = \lambda x$, και διαδοχικών αντικαταστάσεων των παραγόμενων διανυσμάτων, παράγεται

$$v_1 = Av_0 = c_1 Au_1 + c_2 Au_2 + \dots + c_n Au_n = A \sum_{i=1}^n c_i u_i = \sum_{i=1}^n c_i Au_i = \sum_{i=1}^n c_i \lambda_i u_i$$

$$v_2 = Av_1 = A \sum_{i=1}^n c_i \lambda_i u_i = \sum_{i=1}^n c_i \lambda_i Au_i = \sum_{i=1}^n c_i \lambda_i \lambda_i u_i = \sum_{i=1}^n c_i \lambda_i^2 u_i$$

Γενικά για κάθε $j = 0, 1, 2, \dots$ έχουμε :

$$v_j = Av_{j-1} = A^j v_0 = c_1 \lambda_1^j u_1 + c_2 \lambda_2^j u_2 + \dots + c_n \lambda_n^j u_n = \sum_{i=1}^n c_i \lambda_i^j u_i$$

η οποία γράφεται:

$$\begin{aligned} v_j &= A^j v_0 = \lambda_1^j \left[c_1 u_1 + c_2 \left(\frac{\lambda_2}{\lambda_1} \right)^j u_2 + \dots + c_n \left(\frac{\lambda_n}{\lambda_1} \right)^j u_n \right] \\ &= \lambda_1^j \left[c_1 u_1 + \sum_{i=2}^n c_i \left(\frac{\lambda_i}{\lambda_1} \right)^j u_i \right] = \lambda_1^j (c_1 u_1 + \varepsilon_j) \end{aligned} \quad (1.3)$$

όπου

$$\varepsilon_j = \sum_{i=2}^n c_i \left(\frac{\lambda_i}{\lambda_1} \right)^j u_i. \quad (1.4)$$

Από την (1.1) είναι φανερό ότι ισχύει

$$\left| \frac{\lambda_i}{\lambda_1} \right| < 1, \quad i = 2, 3, \dots, n,$$

οπότε στην (1.4) ο όρος ε_j συγκλίνει στο μηδέν, επομένως για το διάνυσμα v_j στην (1.3) μπορούμε να γράψουμε :

$$\lim_{j \rightarrow \infty} v_j = \lim_{j \rightarrow \infty} \lambda_1^j c_1 u_1 = \left(\lim_{j \rightarrow \infty} \lambda_1^j \right) c_1 u_1 \quad (1.5)$$

Υποθέτουμε ότι $c_1 \neq 0$ και θέτουμε στην (1.5), όπου j το $j-1$, οπότε έχουμε

$$\lim_{j \rightarrow \infty} v_{j-1} = \left(\lim_{j \rightarrow \infty} \lambda_1^{j-1} \right) c_1 u_1 \quad (1.6)$$

Επειδή οι σχέσεις (1.5) και (1.6) ισχύουν για οποιαδήποτε συνιστώσα των αναφερόμενων διανυσμάτων, θεωρώντας ότι η k -οστή συνιστώσα ενός διανύσματος για κάθε $k = 1, 2, \dots, n$ σημειώνεται ως εκθέτης του αντίστοιχου διανύσματος μέσα σε παρένθεση, (k) , είναι φανερό ότι οι σχέσεις (1.5) και (1.6) μπορούν να εκφράσουν και τις αντίστοιχες συνιστώσες των διανυσμάτων, οπότε για κάθε $k = 1, 2, \dots, n$ μπορούμε να γράψουμε:

$$\lim_{j \rightarrow \infty} \frac{v_j^{(k)}}{v_{j-1}^{(k)}} = \frac{\left(\lim_{j \rightarrow \infty} \lambda_1^j \right) c_1 u_1^{(k)}}{\left(\lim_{j \rightarrow \infty} \lambda_1^{j-1} \right) c_1 u_1^{(k)}} = \lim_{j \rightarrow \infty} \frac{\lambda_1^j}{\lambda_1^{j-1}} = \lambda_1 \quad (1.7)$$

Επομένως στην (1.7) προσδιορίζεται η μεγαλύτερη κατά μέτρο ιδιοτιμή. Τώρα, από την (1.5) μπορούμε να παράγουμε το αντίστοιχο ιδιοδιάνυσμα, ως ακολούθως:

$$\lim_{j \rightarrow \infty} v_j = \left(\lim_{j \rightarrow \infty} \lambda_1^j \right) c_1 u_1 \Rightarrow \frac{\lim_{j \rightarrow \infty} v_j}{\left(\lim_{j \rightarrow \infty} \lambda_1^j \right)} = c_1 u_1 \Rightarrow \frac{\lim_{j \rightarrow \infty} v_j}{c_1 \cdot \lambda_1} = u_1 \quad (1.8)$$

Παράδειγμα 1.1

Θεωρούμε τον 3×3 πραγματικό πίνακα $A = \begin{pmatrix} 1 & 0 & 1 \\ 1 & 6 & 0 \\ 1 & 2 & -1 \end{pmatrix}$ και το αρχικό διάνυσμα

$$v_0 = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} = (1 \quad 1 \quad 1)^T.$$

Κάνοντας τέσσερις επαναλήψεις και διατηρώντας τέσσερα ψηφία στους υπολογισμούς μας έχουμε:

$$v_1 = Av_0 = \begin{pmatrix} 1 & 0 & 1 \\ 1 & 6 & 0 \\ 1 & 2 & -1 \end{pmatrix} \cdot \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} = \begin{pmatrix} 2 \\ 7 \\ 2 \end{pmatrix} = 7 \begin{pmatrix} 0.2857 \\ 1 \\ 0.2857 \end{pmatrix}$$

$$v_2 = Av_1 = \begin{pmatrix} 1 & 0 & 1 \\ 1 & 6 & 0 \\ 1 & 2 & -1 \end{pmatrix} \cdot \begin{pmatrix} 2 \\ 7 \\ 2 \end{pmatrix} = \begin{pmatrix} 4 \\ 44 \\ 14 \end{pmatrix} = 44 \begin{pmatrix} 0.0909 \\ 1 \\ 0.3182 \end{pmatrix}$$

$$v_3 = Av_2 = \begin{pmatrix} 1 & 0 & 1 \\ 1 & 6 & 0 \\ 1 & 2 & -1 \end{pmatrix} \cdot \begin{pmatrix} 4 \\ 44 \\ 14 \end{pmatrix} = \begin{pmatrix} 18 \\ 268 \\ 78 \end{pmatrix} = 268 \begin{pmatrix} 0.0672 \\ 1 \\ 0.2910 \end{pmatrix}$$

$$v_4 = Av_3 = \begin{pmatrix} 1 & 0 & 1 \\ 1 & 6 & 0 \\ 1 & 2 & -1 \end{pmatrix} \cdot \begin{pmatrix} 18 \\ 268 \\ 78 \end{pmatrix} = \begin{pmatrix} 96 \\ 1626 \\ 476 \end{pmatrix} = 1626 \begin{pmatrix} 0.059 \\ 1 \\ 0.2927 \end{pmatrix}$$

Χρησιμοποιώντας $k = 2$ στην (1.7) από τις παραπάνω ισότητες προκύπτει:

$$\frac{v_2^{(2)}}{v_1^{(2)}} = \frac{44}{7} = 6.2857, \quad \frac{v_3^{(2)}}{v_2^{(2)}} = \frac{268}{44} = 6.0909, \quad \frac{v_4^{(2)}}{v_3^{(2)}} = \frac{1626}{268} = 6.0672$$

Συνεπώς προσεγγιστικά η μέγιστη (κατά απόλυτη τιμή) ιδιοτιμή είναι

$$\lambda_1 = 6.0672$$

με αντίστοιχο ιδιοδιάνυσμα

$$u_1 = \begin{pmatrix} 0.059 \\ 1 \\ 0.2927 \end{pmatrix},$$

το οποίο ορθοκανονικοποιημένο με την $\|\cdot\|_2$ είναι

$$\widetilde{u}_1 = \frac{u_1}{\|u_1\|_2} = \begin{pmatrix} 0.0565 \\ 0.9582 \\ 0.2805 \end{pmatrix}.$$

Παρατήρηση 1.2

- i. Η ταχύτητα σύγκλισης της μεθόδου των δυνάμεων εξαρτάται από τις σταθερές c_i και τους λόγους $\left| \frac{\lambda_2}{\lambda_1} \right|, \left| \frac{\lambda_3}{\lambda_1} \right|, \dots, \left| \frac{\lambda_n}{\lambda_1} \right|$. Όσο μικρότεροι είναι οι λόγοι, δηλαδή όσο το δυνατόν πιο κοντά στο μηδέν, τόσο πιο γρήγορη είναι η σύγκλιση. Αν οι λόγοι είναι κοντά στη μονάδα, τότε η σύγκλιση πολύ αργή.
- ii. Στην περίπτωση που $|\lambda_1| > 1$, τότε από τη σχέση (1.5) έχουμε ότι

$$\lim_{j \rightarrow \infty} v_j = \lim_{j \rightarrow \infty} \lambda_1^j c_1 u_1 = \pm \infty,$$

ενώ για $|\lambda_1| < 1$, τότε

$$\lim_{j \rightarrow \infty} v_j = \lim_{j \rightarrow \infty} \lambda_1^j c_1 u_1 = 0.$$

Έτσι δημιουργείται πρόβλημα αύξησης των σφαλμάτων στρογγυλοποίησης στους υπολογισμούς, διότι θα πρέπει να κάνουμε πράξεις με απόλυτα που έχουν πάρα πολύ μεγάλους ή μικρούς αριθμούς, εκτός εάν $|\lambda_1| \simeq 1$. Αυτό το πρόβλημα μπορούμε να το αποφύγουμε κάνοντας μια τροποποίηση της προαναφερόμενης Μεθόδου των Δυνάμεων, ακολουθώντας έναν αλγόριθμο τριών βημάτων, που περιγράφεται στη συνέχεια και προτείνεται [22], [27], [30].

Αλγόριθμος 1.3

Βήμα 1^ο: Ορίζουμε ένα μη μηδενικό αρχικό διάνυσμα u_0 και μία μικρή θετική τιμή, που σημειώνεται με tol , και ορίζει την ακρίβεια των προσεγγίσεων.

Βήμα 2^ο: Εκτελούμε μια επαναληπτική διαδικασία, για $j = 0, 1, 2, \dots$, για να υπολογιστούν

$$v_{j+1} = A u_j, \quad \lambda_{j+1} = \|v_{j+1}\|_{\infty} \quad \text{και} \quad u_{j+1} = \frac{v_{j+1}}{\|v_{j+1}\|_{\infty}}, \quad (1.9)$$

όπου $\|v_j\|_{\infty}$ συμβολίζει την άπειρη νόρμα του διανύσματος v_j , δηλαδή τη μέγιστη (κατά απόλυτη τιμή) συντεταγμένη του διανύσματος v_j .

Η επαναληπτική διαδικασία τερματίζεται, όταν

$$\|u_{j+1} - u_j\|_{\infty} < \text{tol} \quad (1.10)$$

Βήμα 3^ο: Η τελευταία υπολογισμένη λ_{j+1} είναι η μέγιστη (κατά απόλυτη τιμή) ιδιοτιμή του πίνακα A με το αντίστοιχα υπολογισμένο u_{j+1} είναι το αντίστοιχο ιδιοδιάνυσμά της.

Παράδειγμα 1.4

Θεωρούμε τον πίνακα $A = \begin{pmatrix} 1 & 0 & 1 \\ 1 & 6 & 0 \\ 1 & 2 & -1 \end{pmatrix}$ του Παραδείγματος 1.1, αρχικό διάνυσμα

$u_0 = (1 \ 1 \ 1)^T$ και κριτήριο τερματισμού $\text{tol} = 0.01$ στη συνέχεια υλοποιούμε τον Αλγόριθμο 1.3 για να εντοπίσουμε τις προσεγγιστικές τιμές των ιδιοποσών.

➤ Θέτουμε $j = 0$ στην (1.9)

$$v_1 = Au_0 = \begin{pmatrix} 1 & 0 & 1 \\ 1 & 6 & 0 \\ 1 & 2 & -1 \end{pmatrix} \cdot \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} = \begin{pmatrix} 2 \\ 7 \\ 2 \end{pmatrix}$$

όπου $\|v_1\|_\infty = \max\{2, 7, 2\} = 7$.

Άρα, η προσέγγιση της ιδιοτιμής είναι $\lambda_1 = 7$ και η προσέγγιση του ιδιοδιανύσματος είναι:

$$u_1 = \frac{v_1}{\|v_1\|_\infty} = \frac{(2 \ 7 \ 2)^T}{7} = (0.2857 \ 1 \ 0.2857)^T$$

Έλεγχος τερματισμού:

$$\|u_1 - u_0\|_\infty = \|(0.2857 \ 1 \ 0.2857)^T - (1 \ 1 \ 1)^T\|_\infty = 0.7143 > \text{tol}$$

Επομένως, η επαναληπτική διαδικασία συνεχίζεται.

➤ Θέτουμε $j = 1$ στην (1.9)

$$v_2 = Au_1 = \begin{pmatrix} 1 & 0 & 1 \\ 1 & 6 & 0 \\ 1 & 2 & -1 \end{pmatrix} \cdot \begin{pmatrix} 0.2857 \\ 1 \\ 0.2857 \end{pmatrix} = \begin{pmatrix} 0.5714 \\ 6.2857 \\ 2 \end{pmatrix}$$

όπου $\|v_2\|_\infty = \max\{0.5714, 6.2857, 2\} = 6.2857$.

Άρα, η προσέγγιση της ιδιοτιμής είναι $\lambda_2 = 6.2857$ και η προσέγγιση του ιδιοδιανύσματος είναι:

$$u_2 = \frac{v_2}{\|v_2\|_\infty} = \frac{(0.5714 \ 6.2857 \ 2)^T}{6.2857} = (0.0909 \ 1 \ 0.3182)^T$$

Έλεγχος τερματισμού:

$$\begin{aligned} \|u_2 - u_1\|_\infty &= \|(0.0909 \ 1 \ 0.3182)^T - (0.2857 \ 1 \ 0.2857)^T\|_\infty \\ &= 0.1948 > \text{tol} \end{aligned}$$

Επομένως, η επαναληπτική διαδικασία συνεχίζεται.

➤ Θέτουμε $j = 2$ στην (1.9)

$$v_3 = Au_2 = \begin{pmatrix} 1 & 0 & 1 \\ 1 & 6 & 0 \\ 1 & 2 & -1 \end{pmatrix} \cdot \begin{pmatrix} 0.0909 \\ 1 \\ 0.3182 \end{pmatrix} = \begin{pmatrix} 0.4091 \\ 6.0909 \\ 1.7727 \end{pmatrix}$$

όπου $\|v_3\|_\infty = \max\{0.4091, 6.0909, 1.7727\} = 6.0909$.

Άρα, η προσέγγιση της ιδιοτιμής είναι $\lambda_3 = 6.0909$ και η προσέγγιση του ιδιοδιανύσματος είναι:

$$u_3 = \frac{v_3}{\|v_3\|_\infty} = \frac{(0.4091 \ 6.0909 \ 1.7727)^T}{6.0909} = (0.0672 \ 1 \ 0.2910)^T$$

Έλεγχος τερματισμού:

$$\begin{aligned} \|u_3 - u_2\|_\infty &= \|(0.0672 \ 1 \ 0.2910)^T - (0.0909 \ 1 \ 0.3182)^T\|_\infty \\ &= 0.0271 > \text{tol} \end{aligned}$$

Επομένως, η επαναληπτική διαδικασία συνεχίζεται.

➤ Θέτουμε $j = 3$ στην (1.9)

$$v_4 = Au_3 = \begin{pmatrix} 1 & 0 & 1 \\ 1 & 6 & 0 \\ 1 & 2 & -1 \end{pmatrix} \cdot \begin{pmatrix} 0.0672 \\ 1 \\ 0.2910 \end{pmatrix} = \begin{pmatrix} 0.3582 \\ 6.0672 \\ 1.7761 \end{pmatrix}$$

όπου $\|v_4\|_\infty = \max\{0.3582, 6.0672, 1.7761\} = 6.0672$. Άρα η προσέγγιση της ιδιοτιμής είναι $\lambda_4 = 6.0672$ και η προσέγγιση του ιδιοδιανύσματος είναι:

$$u_4 = \frac{v_4}{\|v_4\|_\infty} = \frac{(0.3582 \quad 6.0672 \quad 1.7761)^T}{6.0672} = (0.0590 \quad 1 \quad 0.2927)^T$$

Έλεγχος τερματισμού:

$$\begin{aligned} \|u_4 - u_3\|_\infty &= \|(0.0590 \quad 1 \quad 0.2927)^T - (0.0672 \quad 1 \quad 0.2910)^T\|_\infty \\ &= 0.0081 < 0.01 = \text{tol} \end{aligned}$$

Επομένως, η επαναληπτική διαδικασία τερματίζεται.

Συνεπώς, η μέγιστη ιδιοτιμή είναι

$$\lambda = 6.0672$$

με αντίστοιχο ιδιοδιάνυσμα

$$u = (0.0590 \quad 1 \quad 0.2927)^T.$$

Συγκρίνοντας τις προσεγγιστικές τιμές που προέκυψαν ακολουθώντας τη Μέθοδο των Δυνάμεων και τον Αλγόριθμο 1.3, παρατηρούμε ότι οι τιμές των προσεγγίσεων ταυτίζονται, παρόλο που υπάρχει διαφορά στις δύο διαδικασίες, η οποία εντοπίζεται στο κριτήριο τερματισμού. Συγκεκριμένα, στη Μέθοδο των Δυνάμεων το κριτήριο τερματισμού σχετίζεται με τη διαφορά δύο διαδοχικών τιμών των μέγιστων συντεταγμένων των διανυσμάτων v_j , ενώ στον Αλγόριθμο 1.3 από την (1.10) συμπεραίνουμε ότι το κριτήριο τερματισμού σχετίζεται με τη διαφορά των αντίστοιχων διανυσμάτων u_j , που υπολογίζονται, για κατάλληλες τιμές j .

Τροποποιώντας τον Αλγόριθμο 1.3, λαμβάνοντας υπόψη τη Μέθοδο των Δυνάμεων και τον Αλγόριθμο, που περιγράφεται [26, Ενότητα 5.5], στη συνέχεια προτείνεται ένας αλγόριθμος, που συνδυάζει τις προαναφερόμενες διαδικασίες και στη συνέχεια αποκαλείται **τροποποιημένη Μέθοδος των Δυνάμεων**. Ο αλγόριθμος αναπτύσσεται στη συνάρτηση `MPowerMeth`, που υλοποιείται σε περιβάλλον Matlab, η οποία, εφαρμόζοντας την τροποποιημένη Μέθοδο των Δυνάμεων, υπολογίζει μέσω της (1.9) τη μέγιστη (κατά απόλυτη τιμή) ιδιοτιμή και το αντίστοιχο ιδιοδιάνυσμά της καθώς και το μέγιστο πλήθος των επαναλήψεων, προκειμένου να επιτευχθεί η προσέγγιση των ιδιοποσών με ακρίβεια εκ των προτέρων γνωστή, η οποία ορίζεται από τον χρήστη σύμφωνα με τις απαιτήσεις ακρίβειας του προβλήματος και σημειώνεται με `tol`. Είσοδος στη συνάρτηση `MPowerMeth` είναι ο πίνακας, A , το σφάλμα της προσέγγισης, `tol`, καθώς και το επιτρεπτό μέγιστο πλήθος επαναλήψεων, `int`, και στην έξοδό της δίνονται η μέγιστη ιδιοτιμή, `lmax`, το αντίστοιχο ιδιοδιάνυσμα, `umax`, καθώς και το

πλήθος των επαναλήψεων, *iterations*, που απαιτήθηκαν για να επιτευχθούν οι προαναφερόμενοι υπολογισμοί.

Κώδικας 1.5

```
function [lmax, umax] = MPowerMeth(A, tol, int)
% Modified Power Method
% tol: the error of accuracy
% int: maximal number of iterations

[m,n]=size(A);
u0 = ones(n,1);
error = 1;
interat = 0;
x0=1;

while error > tol && interat < int
    interat = interat + 1;
    v = A * u0;
    [lm,xx]= max(abs(v));
    u = v/lm;
    error = abs(lm-x0);
    u0 = u;
    x0=lm;
end
iterations=interat;
disp(['number of iterations:' , num2str(iterations)]);
umax=u0/norm(u0);
lmax=umax'*A*umax;
end
```

Στη συνέχεια, εφαρμόζοντας την τροποποιημένη Μέθοδο των Δυνάμεων και υλοποιώντας τον Κώδικα 1.5 η μέγιστη ιδιοτιμή και το αντίστοιχο ιδιοδιάνυσμά της υπολογίζονται.

Παράδειγμα 1.6

Η υλοποίηση του Κώδικα 1.5 για τον πίνακα $A = \begin{pmatrix} 1 & 0 & 1 \\ 1 & 6 & 0 \\ 1 & 2 & -1 \end{pmatrix}$ του Παραδείγματος 1.1, με αρχικό διάνυσμα $u_0 = (1 \ 1 \ 1)^T$, κριτήριο τερματισμού $tol=0.01$ και μέγιστο πλήθος 100 επαναλήψεων δίνει:

- Επειδή $error=1 > tol$ && $interat < int$
 - $interat = 1$

$$v_1 = Au_0 = (2 \ 7 \ 2)^T$$

$$\lambda_1 = \|v_1\|_\infty = \max\{2, 7, 2\} = 7$$

$$u_1 = \frac{v_1}{\|v_1\|_\infty} = \frac{(2 \ 7 \ 2)^T}{7} = (0.2857 \ 1 \ 0.2857)^T$$

Επειδή

$$\text{error} = |\lambda_1 - x_0| = |7 - 1| = 6 > \text{tol} \quad \text{και} \quad \text{interat} < 100$$

η επαναληπτική διαδικασία συνεχίζεται θέτοντας

$$u_0 = u_1$$

$$x_0 = \lambda_1 = 7$$

- $\text{interat} = 2$

$$v_2 = Au_1 = (0.5714 \ 6.2857 \ 2)^T$$

$$\lambda_2 = \|v_2\|_\infty = \max\{0.5714, 6.2857, 2\} = 6.2857$$

$$u_2 = \frac{v_2}{\|v_2\|_\infty} = \frac{(0.5714 \ 6.2857 \ 2)^T}{6.2857} = (0.0909 \ 1 \ 0.3182)^T$$

Επειδή

$$\text{error} = |\lambda_2 - 7| = |6.2857 - 7| = 0.7143 > \text{tol} \quad \text{και} \quad \text{interat} < 100$$

η επαναληπτική διαδικασία συνεχίζεται θέτοντας

$$u_0 = u_2$$

$$x_0 = \lambda_2 = 6.2857;$$

- $\text{interat} = 3$

$$v_3 = Au_2 = (0.4091 \ 6.0909 \ 1.7727)^T$$

$$\lambda_3 = \|v_3\|_\infty = \max\{0.4091, 6.0909, 1.7727\} = 6.0909$$

$$u_3 = \frac{v_3}{\|v_3\|_\infty} = \frac{(0.4091 \ 6.0909 \ 1.7727)^T}{6.0909} = (0.0672 \ 1 \ 0.2910)^T$$

Επειδή

$$\text{error} = |\lambda_3 - 6.2857| = |6.0909 - 6.2857| = 0.1948 > \text{tol} \quad \text{και} \quad \text{interat} < 100$$

η επαναληπτική διαδικασία συνεχίζεται θέτοντας

$$u_0 = u_3$$

$$x_0 = \lambda_3 = 6.0909;$$

- $\text{interat} = 4$

$$v_4 = Au_3 = (0.3582 \ 6.0672 \ 1.7761)^T$$

$$\lambda_4 = \|v_4\|_\infty = \max\{0.3582, 6.0672, 1.7761\} = 6.0672$$

$$u_4 = \frac{v_4}{\|v_4\|_\infty} = \frac{(0.3582 \ 6.0672 \ 1.7761)^T}{6.0672} = (0.0590 \ 1 \ 0.2927)^T$$

Επειδή

$$\text{error} = |\lambda_4 - 6.2857| = |6.0672 - 6.0909| = 0.0237 > \text{tol} \quad \text{και} \quad \text{interat} < 100$$

η επαναληπτική διαδικασία συνεχίζεται θέτοντας

$$u_0 = u_4$$

$$x_0 = \lambda_4 = 6.0672;$$

- $\text{interat} = 5$

$$v_5 = Au_4 = (0.3518 \ 6.0590 \ 1.7663)^T$$

$$\lambda_5 = \|v_5\|_\infty = \max\{0.3518, 6.0590, 1.7663\} = 6.0590$$

$$u_5 = \frac{v_5}{\|v_5\|_\infty} = \frac{(0.3518 \ 6.0590 \ 1.7663)^T}{6.0590} = (0.0581 \ 1 \ 0.2915)^T$$

Επειδή
 $\text{error} = |\lambda_5 - 6.2857| = |6.0590 - 6.0672| = 0.0081 < 0.01 = \text{tol} \ \&\& \ \text{interat} < 100$
 η επαναληπτική διαδικασία τερματίζεται θέτοντας
 $u_0 = u_5 = (0.0581 \ 1 \ 0.2915)^T$
 $x_0 = \lambda_5 = 6.0590$;

Επειδή $\|u_5\|_2 = \sqrt{1.0884} = 1.0432$, το ορθοκανονικοποιημένο αντίστοιχο ιδιοδιάνυσμα είναι:

$$u_{\max} = \frac{u_5}{\|u_5\|_2} = (0.0557 \ 0.9586 \ 0.2794)^T$$

Συνεπώς, η μέγιστη ιδιοτιμή υπολογίζεται

$$\lambda = \text{lmax} = u_{\max}^T A u_{\max} = 6.0581.$$

Σημειώστε, ότι το φάσμα του πίνακα, δηλαδή το σύνολο ιδιοτιμών¹ (με στρογγυλοποίηση και εμφάνιση τεσσάρων δεκαδικών ψηφίων) του πίνακα A, είναι:

$$\sigma(A) = \{-1.3140, 1.2563, 6.0576\}$$

οπότε στη συνέχεια οι διατεταγμένες ιδιοτιμές του A σημειώνονται:

$$\ell_1 = 6.0576 > \ell_2 = 1.2563 > \ell_3 = -1.3140$$

Επομένως, υλοποιώντας τον Κώδικα 1.5, η μέγιστη ιδιοτιμή $\lambda_{\max} = 6.0576$ του πίνακα προσεγγίζεται μετά από πέντε επαναλήψεις από την

$$\lambda = \text{lmax} = 6.0581$$

η οποία εμφανίζει ακρίβεια δύο δεκαδικών ψηφίων με την πραγματική ιδιοτιμή ℓ_1 , δηλαδή, η ακρίβεια είναι όση υποδεικνύεται από το tol.

Η παραπάνω προσέγγιση είναι η **βέλτιστη**, αν συγκρίνουμε με τα αποτελέσματα, που υπολογίστηκαν κατά την εφαρμογή του αλγορίθμου της Μεθόδου των Δυνάμεων (βλέπε, το αποτέλεσμα του Παραδείγματος 1.1) και του Αλγορίθμου 1.4 (βλέπε, το αποτέλεσμα του Παραδείγματος 1.4) και δίνεται από την (1.11), που είναι η βέλτιστη προσέγγιση.

Παρατήρηση 1.7

- i. Το αρχικό διάνυσμα $u_0 = [1 \ 1 \ 1 \ \dots \ 1]^T$, που αναφέρεται στον Κώδικα 1.5 μπορεί να είναι οποιοδήποτε μη μηδενικό διάνυσμα. Εδώ χρησιμοποιήθηκε το συγκεκριμένο για να προκύψουν συγκρίσιμα αποτελέσματα με αυτά των Παραδειγμάτων 1.1 και 1.4.
- ii. Η ταχύτητα σύγκλισης της τροποποιημένης Μεθόδου των Δυνάμεων είναι γραμμική. Όπως αναφέρθηκε η ταχύτητα σύγκλισης εξαρτάται από τους λόγους $\left|\frac{\lambda_2}{\lambda_1}\right|, \left|\frac{\lambda_3}{\lambda_1}\right|, \dots, \left|\frac{\lambda_n}{\lambda_1}\right|$ και ιδιαίτερα από τον λόγο $\left|\frac{\lambda_2}{\lambda_1}\right|^j$ για $j = 1, 2, \dots, n$, δηλαδή, η τάξη σύγκλισης είναι $O\left(\left|\frac{\lambda_2}{\lambda_1}\right|^j\right)$.

¹ Οι υπολογισμοί γίνονται χρησιμοποιώντας το λογισμικό της Matlab την έκδοση R2023.

- iii. Ο υπολογισμός της συνολικής πολυπλοκότητας της υλοποίησης του Κώδικα 1.5 για έναν πίνακα $A \in \mathbb{R}^{n \times n}$ και για m επαναλήψεις δίνεται [6, Section 7.3] και ισούται με

$$\sum_{k=1}^m (2n^2 + 2n) = m(2n^2 + 2n) = 2mn^2 + 2mn \approx O(mn^2). \quad (1.11)$$

Από την (1.11) συμπεραίνουμε ότι όταν ο αριθμός των επαναλήψεων m παίρνει μικρές τιμές σε σχέση με το n , n σημειώνει τη διάσταση του συμμετρικού πίνακα A , τότε η πολυπλοκότητα της τροποποιημένης Μεθόδου των Δυνάμεων είναι $O(mn^2)$, δηλαδή είναι αρκετά καλή και θεωρείται αποδοτική. Όσο πλησιάζει η τιμή του m το n ο αλγόριθμος γίνεται πιο αργός, διότι η πολυπλοκότητα προσεγγίζει το $O(n^3)$.

- iv. Με τη Μέθοδο των Δυνάμεων μπορούμε να υπολογίσουμε και την ελάχιστη ιδιοτιμή ενός πίνακα με την προϋπόθεση ότι το μηδέν δεν είναι ιδιοτιμή του πίνακα. Σε αυτήν την περίπτωση, ο πίνακας είναι αντιστρέψιμος, το δε ιδιοπρόβλημα $Ax = \lambda x$ συνεπάγεται, [23] :

$$A^{-1}x = \frac{1}{\lambda}x$$

Επομένως, για έναν πραγματικό τετραγωνικό πίνακα A με ιδιοτιμές λ_i , $i = 1, 2, \dots, n$, για τις οποίες ισχύει: $|\lambda_1| \geq |\lambda_2| \geq \dots \geq |\lambda_{n-1}| > |\lambda_n| > 0$, μπορούμε να γράψουμε:

$$\frac{1}{|\lambda_n|} > \frac{1}{|\lambda_{n-1}|} \geq \dots \geq \frac{1}{|\lambda_2|} \geq \frac{1}{|\lambda_1|} > 0$$

Εφαρμόζοντας την τροποποιημένη Μέθοδο των Δυνάμεων στον αντίστροφο πίνακα A^{-1} από την προαναφερόμενη ανισοτική σχέση υπολογίζεται η ποσότητα $\frac{1}{|\lambda_n|}$, οπότε και η τιμή με την μικρότερη απόλυτη τιμή, [17]. Παρόλα αυτά, ο υπολογισμός του αντίστροφου πίνακα είναι μια αρκετά χρονοβόρα διαδικασία και για το λόγο αυτό η μέθοδος αυτή αποφεύγεται, [17], [22].

Παίρνοντας αφορμή από την Παρατήρηση 1.7 (iii), τη μεθοδολογία, που αναπτύσσεται [26] και στηρίζεται στον υπολογισμό των ιδιοποσών των πινάκων που προκύπτουν από τη μετατόπιση της αρχής των αξόνων (βλέπε, τη σχετική θεωρία που αναπτύσσεται [26, Υποενότητα 5.6.4]) καθώς και την αναγκαιότητα να υπολογίζονται όλες οι ιδιοτιμές του πίνακα A , στη συνέχεια παραθέτουμε τον αντίστοιχο κώδικα, που υλοποιείται σε περιβάλλον Matlab, προκειμένου να υπολογιστούν όλες οι ιδιοτιμές του πίνακα και στηρίζεται στον Κώδικα 1.3.

Κώδικας 1.8

```
function d = Powers(A, tol, int)
% Modified Power Method for all eigenvalues
% tol: the error of accuracy
% int: maximal number of iterations

[m,n]=size(A);
for i = 1:n
    [lmax, umax] = MPowerMeth(A, tol, int);
```



```

        A = A - lmax * umax * umax';
        d(i) = lmax;
        u(:, i) = umax;
    end
d = sort(d);
end

```

Συνέχεια του Παραδείγματος 1.6. Υλοποιώντας τον Κώδικα 1.8 με τα δεδομένα του Παραδείγματος 1.6 προκύπτουν τα παρακάτω αποτελέσματα:

```

d =
    -1.3098
     1.2526
     6.0581

```

Δηλαδή οι προσεγγιστικές τιμές, που υπολογίζει ο προαναφερόμενος αλγόριθμος, είναι :

$$\lambda_1 = 6.0581 > \lambda_2 = 1.2526 > \lambda_3 = -1.3098$$

υπενθυμίζοντας ότι οι ιδιοτιμές, που υπολογίζονται σε περιβάλλον Matlab έκδοσης R2023, είναι:

$$\ell_1 = 6.0576 > \ell_2 = 1.2563 > \ell_3 = -1.3140$$

Εδώ παρατηρούμε ότι, στις δύο μεγαλύτερες ιδιοτιμές που υπολογίστηκαν υπάρχει ακρίβεια μέχρι και τα δυο πρώτα δεκαδικά ψηφία όση υποδεικνύεται από το tol, ενώ στην τελευταία ιδιοτιμή η ακρίβεια είναι μικρότερη από το tol.

1.1.2 Εφαρμογή της Μεθόδου των Δυνάμεων

Η Μέθοδος των Δυνάμεων χρησιμοποιείται συχνά για τον υπολογισμό της μεγαλύτερης κατά απόλυτη τιμή ιδιοτιμής και του αντίστοιχου ιδιοδιανύσματος ενός πίνακα, ιδιαίτερα όταν ο πίνακας είναι μεγάλων διαστάσεων με στοιχεία μη αρνητικούς πραγματικούς αριθμούς. Ένας τέτοιος πίνακας είναι ο Google Matrix.

Συγκεκριμένα, ο **Google Matrix G** είναι ένας στοχαστικός πίνακας, που τα στοιχεία G_{ij} του δίνονται

$$G_{ij} = a S_{ij} + (1 - a) \frac{1}{N}, \quad 1 \leq i, j \leq N \quad (1.12)$$

όπου

- S_{ij} : Τα στοιχεία του στοχαστικού πίνακα μετάβασης S , που προκύπτει από τον πίνακα γειτνίασης του δικτύου.
- a : Ο παράγοντας απόσβεσης με τιμές $0 < a < 1$, (συνήθως $a = 0.85$).
- N : Ο αριθμός των κόμβων στο δίκτυο.

Ο όρος $a \cdot S_{ij}$ αναπαριστά τη διαδικασία πλοήγησης ενός "τυχαίου περιηγητή", που ακολουθεί τους συνδέσμους του δικτύου, ενώ ο όρος $(1 - a) \frac{1}{N}$ εισάγει την πιθανότητα ο περιηγητής να μεταβεί σε οποιονδήποτε κόμβο με ίση πιθανότητα, καλύπτοντας έτσι τους "dangling nodes" (κόμβους χωρίς εξερχόμενους συνδέσμους), [4], [5].

Η (1.12) ισοδύναμα γράφεται :

$$G = a S + (1 - a) \frac{1}{N} \mathbf{1}, \quad (1.13)$$

όπου $\mathbf{1}$ είναι ο $N \times N$ πίνακας με όλα τα στοιχεία του ίσα με 1 και S είναι ο στοχαστικός πίνακας μετάβασης, που προκύπτει από τον πίνακα γειτνίασης του δικτύου, [4].

Από την (1.12) είναι προφανές ότι το άθροισμα των στοιχείων κάθε στήλης ισούται με 1 και σύμφωνα με τη θεωρία Perron-Frobenius όλες οι ιδιοτιμές του στοχαστικού πίνακα G εντοπίζονται εντός του μοναδιαίου κυκλικού δίσκου, με τη μεγαλύτερη ιδιοτιμή να ισούται με τη μονάδα και να είναι απλή (να εμφανίζεται μία φορά) και το αντίστοιχο μοναδικό ιδιοδιάνυσμά της να έχει συντεταγμένες μη αρνητικούς πραγματικούς αριθμούς, [9, chapter 8].

Στην περίπτωση του Google Matrix, στόχος είναι η εφαρμογή αριθμητικής μεθόδου για τον υπολογισμό του μοναδικού ιδιοδιανύσματος, που αντιστοιχεί στη μονάδα, γνωστό στην αρθρογραφία ως **PageRank vector**, επειδή το διάνυσμα αυτό δίνει τις στατικές πιθανότητες ενός τυχαίου περιηγητή να βρίσκεται σε κάθε κόμβο του δικτύου. Η Μέθοδος των Δυνάμεων θεωρείται ως η καταλληλότερη μέθοδος για τον εντοπισμό του PageRank vector, λόγω της απλότητάς της και της προσαρμοστικότητάς της σε αραιούς πίνακες, πίνακες που προκύπτουν κατά την απεικόνιση τεράστιων σε μέγεθος δικτύων, [2], [4], [5].

➤ Εφαρμογή της Μεθόδου των Δυνάμεων μέσω του Αλγορίθμου 1.3

- ο Ξεκινάμε με ένα αρχικό θετικό διάνυσμα u_0 , συνήθως τυχαίο ή με ίσες πιθανότητες για όλους τους κόμβους και έναν μικρό προκαθορισμένο αριθμό, tol , το οποίο καθορίζει και την επιθυμητή ακρίβεια των προσεγγιστικών λύσεων του προβλήματος.
- ο Επαναλαμβάνουμε τη διαδικασία: $v_{j+1} = G \cdot u_j$, $j=0,1,2,\dots$
- ο Κανονικοποιούμε το u_{j+1} σε κάθε βήμα, ώστε να διατηρεί τη συνολική πιθανότητα στο 1 ως εξής: $u_{j+1} = \frac{v_{j+1}}{\|v_{j+1}\|_\infty}$
- ο Η διαδικασία συνεχίζεται έως ότου το u_j συγκλίνει, δηλαδή η διαφορά $\|u_{j+1} - u_j\|_\infty$ να γίνει μικρότερη από το tol .

➤ Αποτελεσματικότητα της Μεθόδου των Δυνάμεων:

- ο Η σύγκλιση είναι γρήγορη λόγω της απόσβεσης a , που διασφαλίζει ότι ο πίνακας G είναι **αυστηρά στοχαστικός** και έχει **μοναδικό** κυρίαρχο ιδιοδιάνυσμα το PageRank vector, που υπολογίζεται γρήγορα.

- ο Για τυπικές τιμές $a = 0.85$, η μέθοδος των δυνάμεων συγκλίνει μετά από περίπου 100 - 150 επαναλήψεις για δίκτυα μεγάλου μεγέθους έως και εκατομμυρίων κόμβων.
- ο Η διαδικασία πολλαπλασιασμού του πίνακα G με ένα διάνυσμα έχει πολυπλοκότητα $O(N)$, λόγω της αραιότητας του πίνακα.
- ο Οι "dangling nodes" (κόμβοι χωρίς εξερχόμενους συνδέσμους) αντιμετωπίζονται μέσω της προσθήκης της στοχαστικής συνιστώσας με τύπο $(1 - a) \frac{1}{N}$, διατηρώντας την αποτελεσματικότητα της μεθόδου.

1.2 Μέθοδοι υποχώρων

1.2.1 Επανάληψη Υποχώρου (Subspace Iteration)

Η μέθοδος Επανάληψης Υποχώρου είναι μια γενίκευση της Μεθόδου των Δυνάμεων [7], [28], [29]. Η βασική ιδέα της μεθόδου ξεκινάει με την επιλογή ενός αρχικού υποχώρου αντί για ένα διάνυσμα, δηλαδή ενός συνόλου διανυσμάτων $S \subset \mathbb{R}^n$ διάστασης k , όπου τα διανύσματα αυτού του υποχώρου είναι γραμμικά ανεξάρτητα. Στη συνέχεια υπολογίζεται μια ακολουθία $S, AS, A^2S, A^3S, \dots$, όπου τα στοιχεία αυτής της ακολουθίας συγκλίνουν στον παραγόμενο υποχώρο από τα k μεγαλύτερα ιδιοδιανύσματα του πίνακα A .

Αναλυτικότερα ακολουθούμε την εξής διαδικασία:

1. Επιλέγουμε τυχαία έναν αρχικό υπόχωρο $S \subset \mathbb{R}^n$, συνήθως ορθοκανονικό, με διάσταση k , όπου k είναι ο αριθμός των μεγαλύτερων ιδιοτιμών που αναζητούμε.
2. Πολλαπλασιάζουμε τον αρχικό υπόχωρο με τον πίνακα A για να δημιουργήσουμε έναν νέο υπόχωρο.
3. Εκτελούμε ορθοκανονικοποίηση, που μπορεί να γίνει με διάφορους τρόπους, π.χ. μέσω της διαδικασίας του Gram-Schmidt για να δημιουργήσουμε έναν ορθοκανονικό υπόχωρο.
4. Αντικαθιστούμε τον αρχικό υπόχωρο με τον νέο υπόχωρο.
5. Επαναλαμβάνουμε τα βήματα 2 και 3 μέχρι τη σταθεροποίηση του παραγόμενου υποχώρου, δηλαδή να μην αλλάζει σημαντικά ο υπόχωρος από επανάληψη σε επανάληψη. Στο σημείο αυτό, ο υπόχωρος περιέχει προσεγγίσεις για τα ιδιοδιανύσματα που αντιστοιχούν στις μεγαλύτερες ιδιοτιμές.
6. Επειδή ο υπόχωρος σταθεροποιήθηκε, άρα ο αλγόριθμος συγκλίνει, οι μεγαλύτερες ιδιοτιμές του πίνακα A μπορούν να υπολογιστούν από τις ιδιοτιμές ενός πίνακα που προκύπτει από την προβολή του αρχικού πίνακα A πάνω στον τελικό υπόχωρο.

Καταλήγουμε λοιπόν ότι η μέθοδος της Επανάληψης Υποχώρου είναι σημαντική στην Αριθμητική Ανάλυση, ιδιαίτερα όταν μελετούμε μεγάλα μηχανικά και φυσικά συστήματα, όπου η επίλυση ιδιοτιμών είναι απαραίτητη για την κατανόηση των δυναμικών ιδιοτήτων του συστήματος.

Ωστόσο, η παραπάνω μέθοδος χρησιμοποιείται σπανιότερα, μια και έχει αντικατασταθεί από μεθόδους των υποχώρων Krylov, οι οποίες προσεγγίζουν βέλτιστα τις μεγαλύτερες ιδιοτιμές ενός πίνακα μεγάλων διαστάσεων και ιδιαίτερα όταν είναι αραιός.

1.2.2 Μέθοδος υποχώρων Krylov

Στην ενότητα αυτή θα παρουσιάσουμε μία μέθοδο που βασίζεται σε ιδέες που είναι κατάλληλες για να εφαρμοστούν σε πίνακες τυχαίους (μη συμμετρικούς) μεγάλης διάστασης, αραιούς, που επιλύουν προβλήματα ιδιοτιμών.

Θεωρούμε ότι χρειάζεται να επιλυθεί ένα γραμμικό σύστημα $Au = b$, όπου ο πίνακας $A \in \mathbb{R}^{n \times n}$ είναι ένας αραιός και πολύ μεγάλης διάστασης πίνακας (π.χ. $n > 10^4$). Η ιδέα

που εισήγαγε η μέθοδος Krylov είναι η μείωση της διάστασης του πίνακα του συστήματος, ώστε η λύση του γραμμικού συστήματος να εντοπίζεται σε έναν άλλο χώρο πολύ μικρότερης διάστασης από τη διάσταση του αρχικού πίνακα A . Επειδή ένα ιδιοπρόβλημα μπορεί να πάρει τη μορφή του παραπάνω γραμμικού προβλήματος αρκεί $b = \ell u$, στη συνέχεια η αναζήτηση του ιδιοζεύγους (ℓ, u) είναι ισοδύναμη με την επίλυση του αντίστοιχου γραμμικού συστήματος. Οπότε εδώ επικεντρωνόμαστε στη μελέτη των υποχώρων Krylov με σκοπό να επιλύσουμε μεγάλων διαστάσεων γραμμικά συστήματα μέσω επαναληπτικών μεθόδων παράγοντας χώρους μικρότερης διάστασης μια και στηρίζονται σε προβολικές διαδικασίες, [17, Section 4.3].

Οι προβολικές διαδικασίες διακρίνονται σε δύο κατηγορίες :

- στην ορθή προβολική, και
- στην πλάγια προβολική.

Στην **ορθή προβολική διαδικασία** έχουμε να επιλύσουμε ένα πρόβλημα ιδιοτιμών με $A \in \mathbb{R}^{n \times n}$ και έστω (ℓ, u) να είναι το ζεύγος που ικανοποιεί το ιδιοπρόβλημα με $\ell \in \mathbb{R}$ και $u \in \mathbb{R}^n$. Επίσης έχουμε τον υπόχωρο $K \subset \mathbb{R}^n$ διάστασης m (με $m < n$) και τον $n \times m$ πίνακα $Q = (q_1 \ q_2 \ \dots \ q_m)$, όπου οι στήλες του αποτελούν ορθοκανονική βάση του υποχώρου K . Εφαρμόζοντας τη συνθήκη ορθογωνιότητας Petrov-Galerkin εξάγουμε μια προσέγγιση του ζεύγους (ℓ, u) , που τη συμβολίζουμε (λ, u) με $\lambda \in \mathbb{R}$ και $u \in K$. Άρα το αρχικό ιδιοπρόβλημα διάστασης $n \times n$ έχει μετασχηματιστεί σε ένα πρόβλημα διάστασης $m \times m$

$$Q^T A Q x = \lambda x, \quad \forall x \in K \quad (1.14)$$

Έπειτα εφαρμόζουμε τη διαδικασία Rayleigh-Ritz για τον υπολογισμό των προσεγγίσεων, όπου οι προσεγγιστικές ιδιοτιμές και ιδιοδιανύσματα που προκύπτουν καλούνται τιμές και διανύσματα Ritz, αντίστοιχα. Στην ειδική περίπτωση που ο υπόχωρος K είναι αναλλοίωτος υπό τον A , τότε τα προσεγγιστικά ιδιοζεύγη ταυτίζονται με τα ιδιοζεύγη του A .

Όσο για την **πλάγια προβολική διαδικασία** θεωρούμε ότι έχουμε επιλύσουμε ένα πρόβλημα ιδιοτιμών όπως παραπάνω και έστω (ℓ, u) το ζεύγος που ικανοποιεί το ιδιοπρόβλημα. Έστω ότι έχουμε δύο υπόχωρους $K \subset \mathbb{R}^n$, $L \subset \mathbb{R}^n$ και τους $n \times m$ πίνακες $Q = (q_1 \ q_2 \ \dots \ q_m)$, $G = (g_1 \ g_2 \ \dots \ g_m)$, όπου οι στήλες κάθε πίνακα αποτελούν ορθοκανονική βάση του αντίστοιχου υποχώρου K και L . Εφαρμόζοντας τη συνθήκη Petrov-Galerkin στον υπόχωρο L παράγουμε μια προσέγγιση του ζεύγους (ℓ, u) , που τη συμβολίζουμε (λ, u) . Άρα όπως και προηγούμενα το αρχικό ιδιοπρόβλημα διάστασης $n \times n$ μετασχηματίζεται σε ένα πρόβλημα διάστασης $m \times m$ που διατυπώνεται ανάλογα με την (1.14).

Στη συνέχεια εφαρμόζουμε τη διαδικασία Rayleigh-Ritz για τον υπολογισμό των προσεγγίσεων, [17, Algorithm 4.5]. Προφανώς εάν οι υπόχωροι K και L είναι αναλλοίωτοι υπό τον A , τότε τα προσεγγιστικά ιδιοζεύγη ταυτίζονται με τα ιδιοζεύγη του A .

Ορισμός 1.9 (Υπόχωρος Krylov)

Έστω ένας πίνακας $A \in \mathbb{R}^{n \times n}$ και $x \in \mathbb{R}^n$ ένα μη μηδενικό διάνυσμα, η ακολουθία

$$\{x, Ax, A^2x, \dots, A^{n-1}x\}$$

ονομάζεται ακολουθία Krylov και ο πίνακας

$$K_n(A, x) = (x, Ax, A^2x, \dots, A^{n-1}x)$$

ονομάζεται πίνακας Krylov.

Ο υπόχωρος

$$K_m(A, x) = \text{span}\{x, Ax, A^2x, \dots, A^{m-1}x\}$$

ονομάζεται υπόχωρος Krylov διάστασης m , υποθέτοντας ότι τα διανύσματα $(x, Ax, A^2x, \dots, A^{m-1}x)$ είναι γραμμικώς ανεξάρτητα.

Οι βασικές μέθοδοι υπολογισμού του υποχώρου Krylov είναι η μέθοδος Arnoldi [3], [16] και η μέθοδος Lanczos που εφαρμόζεται μόνο για συμμετρικούς πίνακες (βλέπε αναλυτικά στο Κεφάλαιο 2). Στη περίπτωση που ο πίνακας A είναι συμμετρικός και η επιλογή του αρχικού διανύσματος είναι ίδια, τότε οι δυο προαναφερθείσες μέθοδοι ταυτίζονται.

Μέθοδος Arnoldi

Η διαδικασία Arnoldi χρησιμοποιείται για την επίλυση του ιδιοπροβλήματος ενός τυχαίου πίνακα $A \in \mathbb{R}^{n \times n}$ μεγάλης διάστασης και αραιού, με σκοπό τον υπολογισμό ορισμένων τιμών που να προσεγγίζουν ορισμένες ιδιοτιμές του πίνακα A χωρίς να απαιτείται η χρήση των πολύπλοκων μεθόδων εντοπισμού των ιδιοτιμών του πίνακα. Η καινοτομία της μεθόδου είναι η «προβολή» του προβλήματος διάστασης n σε έναν μικρότερης διάστασης χώρο (τον υπόχωρο Krylov), ο οποίος περιέχει την πληροφορία που χρειάζεται για τον εντοπισμό των ιδιοτιμών.

Η μέθοδος βασίζεται κυρίως στο μετασχηματισμό του πίνακα A σε έναν πίνακα H άνω Hessenberg μέσω της δημιουργίας ενός ορθογώνιου πίνακα Q τέτοιου ώστε $Q^T A Q = H$. Συγκεκριμένα, αν $Q = (q_1 \ q_2 \ \dots \ q_n)$ ένας ορθογώνιος πίνακας, όπου κάθε στήλη δημιουργεί έναν υπόχωρο Krylov διάστασης k της μορφής

$$K_k(A, q_1) = \text{span}\{q_1, Aq_1, A^2q_1, \dots, A^{k-1}q_1\} \quad (1.15)$$

τότε μέσω της ισότητας $AQ = QH$, η στήλη Aq_k εκφράζεται ως γραμμικός συνδυασμός των ήδη υπολογισμένων στηλών q_i με συντελεστές h_{ik} , όπως φαίνεται στο παρακάτω τύπο:

$$Aq_k = \sum_{i=1}^{k+1} h_{ik} q_i, \quad 1 \leq k \leq n-1 \quad (1.16)$$

Λύνοντας ως προς τον τελευταίο όρο του αθροίσματος έχουμε:

$$h_{k+1,k} q_{k+1} = Aq_k - \sum_{i=1}^k h_{ik} q_i \equiv r_k, \quad (1.17)$$

όπου $h_{ik} = q_i^T Aq_k$ για $i = 1, 2, \dots, k$.

Αν $r_k \neq 0$, τότε το q_{k+1} προσδιορίζεται ως

$$q_{k+1} = \frac{r_k}{h_{k+1,k}} = \frac{r_k}{\|r_k\|_2}, \quad (1.18)$$

όπου $\|r_k\|_2$ σημειώνει τη 2-νόρμα ή Ευκλείδεια νόρμα του διανύσματος r_k , δηλαδή την τετραγωνική ρίζα του αθροίσματος των τετραγώνων όλων των συντεταγμένων του r_k .

Από την (1.18) είναι φανερό ότι το q_{k+1} είναι μοναδιαίο διάνυσμα.

Η διαδικασία επαναλαμβάνεται μέχρι να υπολογιστούν όλες οι στήλες του πίνακα Q και τα στοιχεία του πίνακα H . Ο επόμενος αλγόριθμος υλοποιεί τα παραπάνω ξεκινώντας με ένα μοναδιαίο διάνυσμα q_1 .

Αλγόριθμος 1.10 (Μέθοδος Arnoldi)

Έστω $A \in \mathbb{R}^{n \times n}$ και $q_1 \in \mathbb{R}^n$ με $\|q_1\|_2 = 1$. Ο ακόλουθος αλγόριθμος προσδιορίζει έναν πίνακα $Q_t = (q_1 \ q_2 \ \dots \ q_t) \in \mathbb{R}^{n \times t}$ με ορθοκανονικές στήλες και ένα άνω Hessenberg πίνακα $H_t = (h_{ij}) \in \mathbb{R}^{t \times t}$ με την ιδιότητα ότι $AQ_t = Q_t H_t$. Για τον ακέραιο t ισχύει: $1 \leq t \leq n$

```

    k = 0, r_0 = q_1, h_10 = 1
    while (h_{k+1,k} ≠ 0)
        q_{k+1} = r_k / h_{k+1,k}
        k = k + 1
        r_k = Aq_k
        for i = 1: k
            h_{ik} = q_i^T r_k
            r_k = r_k - h_{ik} q_i
        end
        h_{k+1,k} = \|r_k\|_2
    end
    t = k

```

Τα διανύσματα q_k , ονομάζονται **διανύσματα Arnoldi** και δημιουργούν μια ορθοκανονική βάση για τον υπόχωρο Krylov της μορφής (1.15). Έπειτα από k βήματα καταλήγουμε στην εξίσωση

$$AQ_k = Q_k H_k + r_k e_k^T, \quad (1.19)$$

όπου $Q_k = (q_1 \ q_2 \ \dots \ q_k)$, $e_k = I_k(:, k)$ δηλαδή, το e_k αναπαριστά τη k -οστή στήλη του $k \times k$ μοναδιαίου πίνακα I_k και

$$H_k = \begin{pmatrix} h_{11} & h_{12} & h_{13} & \dots & h_{1k} \\ h_{21} & h_{22} & h_{23} & \dots & h_{2k} \\ 0 & h_{32} & \ddots & \dots & h_{3k} \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \dots & h_{k,k-1} & h_{kk} \end{pmatrix} \quad (1.20)$$

Ο πίνακας H_k στην (1.20) είναι σχεδόν άνω τριγωνικός, διότι έχει και ένα μη μηδενικό στοιχείο, το οποίο βρίσκεται κάτω από το στοιχείο της κύριας διαγωνίου του πίνακα, μορφή H_k που καθιστά την επίλυση του ιδιοπροβλήματος ευκολότερη από την επίλυση του προβλήματος έχοντας τον πίνακα A στην αρχική του μορφή. Κάθε ανάλυση της μορφής (1.19) είναι μια διαδικασία Arnoldi k βημάτων, εάν ισχύει ότι ο πίνακας Q_k έχει ορθοκανονικές στήλες για τον οποίο ισχύει $Q_k^T \Gamma_k = 0$, και ο πίνακας H_k είναι ένας άνω Hessenberg. Συνδυάζοντας τις ανωτέρω ιδιότητες των πινάκων Q_k, H_k, Γ_k συμπεραίνουμε ότι η Μέθοδος Arnoldi k βημάτων ολοκληρώνεται όταν παράγονται πίνακες που την ισότητα στην (1.19) την διαμορφώνουν

$$AQ_k = Q_k H_k \quad (1.21)$$

Επιπλέον, εάν το y είναι ιδιοδιάνυσμα του πίνακα H_k με ιδιοτιμή λ , τότε μπορούμε να προσεγγίσουμε τις ιδιοτιμές του A μέσω της εξίσωσης:

$$(A - \lambda I)x = (e_k^T y) r_k, \quad (1.22)$$

όπου $x = Q_k y$. Με την προϋπόθεση ότι το διάνυσμα r_k είναι κάθετο σε όλα τα διανύσματα του υποχώρου $Krylov$. Με αυτόν τον τρόπο καταλήγουμε σε ένα ζεύγος Ritz (λ, x) .

Στην περίπτωση που $u = (e_k^T y) r_k$, τότε

$$(A + E)x = \lambda x, \quad (1.23)$$

όπου $E = -ux^T$ με $\|E\|_2 = |y_k| \|r_k\|_2$.

Δύο χαρακτηριστικά της Μεθόδου Arnoldi που τη διακρίνουν είναι:

- Η αναφορά σε όλα τα διανύσματα Arnoldi $q_1, q_2, \dots, q_k$ πρέπει να γίνεται στο k -στό βήμα και ο υπολογισμός του q_{k+1} περιλαμβάνει $O(kn)$ flops, αν εξαιρέσουμε το γινόμενο Aq_k . Συνεπώς, υπάρχει σημαντικό υπολογιστικό κόστος που σχετίζεται με τη δημιουργία ακολουθιών Arnoldi μεγάλου μήκους.
- Οι πληροφορίες για τις ακραίες ιδιοτιμές δεν είναι τόσο βοηθητικές όσο στη συμμετρική περίπτωση, όπως θα δούμε στη συνέχεια. Δεν υπάρχει μη συμμετρική θεωρία σύγκλισης αντίστοιχη της θεωρίας Kaniel-Paige-Saad, [7].

Παράδειγμα 1.11

Η υλοποίηση του Αλγορίθμου 1.10 βήμα-βήμα για τον πίνακα $A = \begin{pmatrix} 1 & 0 & 1 \\ 1 & 6 & 0 \\ 1 & 2 & -1 \end{pmatrix}$ του

Παραδείγματος 1.6 και με αρχικό διάνυσμα το $q_1 = (1 \ 1 \ 1)^T$ παρουσιάζεται στη συνέχεια.

Βήμα 1^ο: Κανονικοποίηση του αρχικού διανύσματος:

$$q_1 = \frac{q_1}{\|q_1\|_2} = \frac{1}{\sqrt{3}} \cdot \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} = \left(\frac{1}{\sqrt{3}} \quad \frac{1}{\sqrt{3}} \quad \frac{1}{\sqrt{3}} \right)^T = \frac{1}{\sqrt{3}} (1 \ 1 \ 1)^T$$

Βήμα 2^ο: Υπολογισμός του γινομένου :

$$r_1 = Aq_1 = \frac{1}{\sqrt{3}} \begin{pmatrix} 1 & 0 & 1 \\ 1 & 6 & 0 \\ 1 & 2 & -1 \end{pmatrix} \cdot \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} = \frac{1}{\sqrt{3}} \begin{pmatrix} 2 \\ 7 \\ 2 \end{pmatrix}$$

Βήμα 3^ο: Η προβολή του Aq_1 πάνω στο q_1 δίνει το h_{11} , που είναι το πρώτο στοιχείο του άνω Hessenberg πίνακα H_k , ως εξής:

$$h_{11} = q_1^T r_1 = \frac{1}{\sqrt{3}} (1 \ 1 \ 1)^T \cdot \frac{1}{\sqrt{3}} \begin{pmatrix} 2 \\ 7 \\ 2 \end{pmatrix} = \frac{1}{3} (2 + 7 + 2) = \frac{11}{3} \cong 3.6667$$

Στη συνέχεια υπολογίζεται το δεύτερο διάνυσμα από την αφαίρεση της προβολής από το r_1 :

$$u_2 = r_1 - h_{11}q_1 = \frac{1}{\sqrt{3}} \begin{pmatrix} 2 \\ 7 \\ 2 \end{pmatrix} - \frac{11}{3} \frac{1}{\sqrt{3}} \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} = \frac{-5}{3\sqrt{3}} \begin{pmatrix} 1 \\ -2 \\ 1 \end{pmatrix}$$

Το μέτρο του u_2 χρειάζεται για τον υπολογισμό του στοιχείου h_{21} , επειδή

$$h_{21} = \|u_2\|_2 = \frac{5}{3\sqrt{3}} \sqrt{1^2 + (-2)^2 + 1^2} = \frac{5}{3\sqrt{3}} \sqrt{6} = \frac{5\sqrt{2}}{3}$$

Η κανονικοποίηση του u_2 χρειάζεται για τον υπολογισμό του δεύτερου διανύσματος

$$q_2 = \frac{u_2}{\|u_2\|_2} = \frac{3}{5\sqrt{2}} \frac{-5}{3\sqrt{3}} \begin{pmatrix} 1 \\ -2 \\ 1 \end{pmatrix} = -\frac{1}{\sqrt{6}} \begin{pmatrix} 1 \\ -2 \\ 1 \end{pmatrix}$$

Βήμα 4^ο: Υπολογισμός του γινομένου για $k=2$:

$$r_2 = Aq_2 = -\frac{1}{\sqrt{6}} \begin{pmatrix} 1 & 0 & 1 \\ 1 & 6 & 0 \\ 1 & 2 & -1 \end{pmatrix} \cdot \begin{pmatrix} 1 \\ -2 \\ 1 \end{pmatrix} = \frac{1}{\sqrt{6}} \begin{pmatrix} -2 \\ 11 \\ 4 \end{pmatrix}$$

Βήμα 5^ο: Η προβολή του Aq_2 πάνω στο q_1 δίνει το στοιχείο h_{12} του άνω Hessenberg πίνακα H_k , ως εξής:

$$h_{12} = q_1^T r_2 = \frac{1}{\sqrt{3}} (1 \ 1 \ 1) \cdot \frac{1}{\sqrt{6}} \begin{pmatrix} -2 \\ 11 \\ 4 \end{pmatrix} = \frac{1}{3\sqrt{2}} (-2 + 11 + 4) = \frac{13}{3\sqrt{2}}$$

Στη συνέχεια υπολογίζεται το τρίτο διάνυσμα από την αφαίρεση της προβολής από το r_1 :

$$u_3 = r_2 - h_{12}q_1 = \frac{1}{\sqrt{6}} \begin{pmatrix} -2 \\ 11 \\ 4 \end{pmatrix} - \frac{13}{3\sqrt{2}} \cdot \frac{1}{\sqrt{3}} \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} = -\frac{1}{3\sqrt{6}} \begin{pmatrix} 19 \\ -20 \\ 1 \end{pmatrix}$$

Καθώς και το στοιχείο h_{22}

$$h_{22} = q_2^T r_2 = -\frac{1}{\sqrt{6}} (1 \quad -2 \quad 1)^T \cdot \frac{1}{\sqrt{6}} \begin{pmatrix} -2 \\ 11 \\ 4 \end{pmatrix} = \frac{1}{6} (2 + 22 - 4) = \frac{20}{6} = \frac{10}{3}$$

Και το διάνυσμα

$$r_3 = r_2 - h_{22}q_2 = \frac{1}{\sqrt{6}} \begin{pmatrix} -2 \\ 11 \\ 4 \end{pmatrix} + \frac{10}{3} \frac{1}{\sqrt{6}} \begin{pmatrix} 1 \\ -2 \\ 1 \end{pmatrix} = \frac{1}{3\sqrt{6}} \begin{pmatrix} 4 \\ 13 \\ 22 \end{pmatrix}$$

Το μέτρο του u_3 χρειάζεται για τον υπολογισμό του στοιχείου h_{32} , επειδή

$$h_{32} = \|u_3\|_2 = \frac{1}{3\sqrt{6}} \sqrt{19^2 + (-20)^2 + 1^2} = \frac{1}{3\sqrt{6}} \sqrt{762} = \frac{\sqrt{127}}{3}$$

Η κανονικοποίηση του r_3 χρειάζεται για τον υπολογισμό του τρίτου διανύσματος

$$q_3 = \frac{u_3}{\|u_3\|_2} = \frac{3}{\sqrt{127}} \frac{1}{3\sqrt{6}} \begin{pmatrix} 4 \\ 13 \\ 22 \end{pmatrix} = -\frac{1}{\sqrt{762}} \begin{pmatrix} 19 \\ -20 \\ 1 \end{pmatrix}$$

Βήμα 6^ο: Υπολογισμός του γινομένου για $k=3$:

$$r_3 = Aq_3 = -\frac{1}{\sqrt{762}} \begin{pmatrix} 1 & 0 & 1 \\ 1 & 6 & 0 \\ 1 & 2 & -1 \end{pmatrix} \cdot \begin{pmatrix} 19 \\ -20 \\ 1 \end{pmatrix} = \frac{1}{\sqrt{762}} \begin{pmatrix} -20 \\ 101 \\ 22 \end{pmatrix}$$

Βήμα 7^ο: Η προβολή του Aq_3 πάνω στο q_1 δίνει το στοιχείο h_{13} του άνω Hessenberg πίνακα H_k , ως εξής:

$$h_{13} = q_1^T r_3 = \frac{1}{\sqrt{3}} (1 \quad 1 \quad 1) \cdot \frac{1}{\sqrt{762}} \begin{pmatrix} -20 \\ 101 \\ 22 \end{pmatrix} = \frac{1}{3\sqrt{254}} (-20 + 101 + 22) = \frac{103}{3\sqrt{254}}$$

Βήμα 8^ο: Η προβολή του Aq_3 πάνω στο q_2 δίνει το στοιχείο h_{23} του άνω Hessenberg πίνακα H_k , ως εξής:

$$h_{23} = q_2^T r_3 = \frac{1}{\sqrt{6}} (-1 \quad 2 \quad -1) \cdot \frac{1}{\sqrt{762}} \begin{pmatrix} -20 \\ 101 \\ 22 \end{pmatrix} = \frac{1}{6\sqrt{127}} (20 + 202 - 22) = \frac{200}{6\sqrt{127}}$$

Καθώς και το στοιχείο h_{22}

$$\begin{aligned} h_{33} = q_3^T r_3 &= -\frac{1}{\sqrt{762}} (19 \quad -20 \quad 1)^T \cdot \frac{1}{\sqrt{762}} \begin{pmatrix} -20 \\ 101 \\ 22 \end{pmatrix} = -\frac{1}{762} (-380 - 2020 + 22) \\ &= \frac{2378}{762} = \frac{1189}{381} \end{aligned}$$

Βήμα 9^ο: Ο τελικός άνω Hessenberg πίνακας H_k είναι:

$$H_k = \begin{pmatrix} \frac{11}{3} & \frac{13}{3\sqrt{2}} & \frac{103}{3\sqrt{254}} \\ \frac{5\sqrt{2}}{3} & \frac{10}{3} & \frac{200}{6\sqrt{127}} \\ 0 & \frac{127}{3} & \frac{1189}{381} \end{pmatrix}$$

Βήμα 10^ο: Υπολογίζουμε τις ιδιοτιμές του πίνακα H_k από την εξίσωση

$$\det(H_k - \lambda I) = 0 \Leftrightarrow \det \begin{pmatrix} \frac{11}{3} - \lambda & \frac{13}{3\sqrt{2}} & \frac{103}{3\sqrt{254}} \\ \frac{5\sqrt{2}}{3} & \frac{10}{3} - \lambda & \frac{200}{6\sqrt{127}} \\ 0 & \frac{127}{3} & \frac{1189}{381} - \lambda \end{pmatrix} = 0$$

από όπου έχουμε:

$$\lambda_1 = \frac{10101}{1250} = 8.0808, \lambda_2 = \frac{1179}{500} = 2.3580, \lambda_3 = -\frac{1591}{5000} = -0.3182$$

Που είναι οι προσεγγιστικές ιδιοτιμές του πίνακα A .

Βήμα 11^ο: Υπολογίζουμε τα ιδιοδιανύσματα του πίνακα H_k

$$(H_k - \lambda I)y = 0 \Rightarrow \begin{pmatrix} \frac{11}{3} - \lambda & \frac{13}{3\sqrt{2}} & \frac{103}{3\sqrt{254}} \\ \frac{5\sqrt{2}}{3} & \frac{10}{3} - \lambda & \frac{200}{6\sqrt{127}} \\ 0 & \frac{127}{3} & \frac{1189}{381} - \lambda \end{pmatrix} \cdot \begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$$

από όπου έχουμε:

$$\hat{h}_1 = \begin{pmatrix} -0.6468 \\ -0.6080 \\ -0.4605 \end{pmatrix}, \hat{h}_2 = \begin{pmatrix} 0.7539 \\ 0.1307 \\ -0.6439 \end{pmatrix}, \hat{h}_3 = \begin{pmatrix} 0.1196 \\ -0.6704 \\ 0.7323 \end{pmatrix}$$

Βήμα 9^ο: Μετατρέπουμε τα ιδιοδιανύσματα $\hat{h}_1, \hat{h}_2, \hat{h}_3$ του πίνακα H_k σε ιδιοδιανύσματα του πίνακα A .

Για να βρούμε τα ιδιοδιανύσματα του A , χρησιμοποιούμε τον πίνακα Q_k ο οποίος κατασκευάστηκε με τη Μέθοδο Arnoldi, (βλέπε, Αλγόριθμο 1.10) και αποτελείται από τα ορθοκανονικά διανύσματα q_1, q_2 και q_3 όπου:

$$q_1 = \frac{1}{\sqrt{3}} \cdot \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}, \quad q_2 = -\frac{1}{\sqrt{6}} \begin{pmatrix} 1 \\ -2 \\ 1 \end{pmatrix}, \quad q_3 = -\frac{1}{\sqrt{762}} \begin{pmatrix} 19 \\ -20 \\ 1 \end{pmatrix}$$

Τα ιδιοδιανύσματα του A θα είναι:

$$v_1 = Q_k \hat{h}_1 = q_1 \cdot \hat{h}_{1,1} + q_2 \cdot \hat{h}_{1,2} + q_3 \cdot \hat{h}_{1,3}$$

$$v_1 = \frac{-0.6468}{\sqrt{3}} \cdot \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} + \frac{0.6080}{\sqrt{6}} \begin{pmatrix} 1 \\ -2 \\ 1 \end{pmatrix} + \frac{0.4605}{\sqrt{762}} \begin{pmatrix} 19 \\ -20 \\ 1 \end{pmatrix} = \begin{pmatrix} 0.1917 \\ -1.2035 \\ -0.1085 \end{pmatrix}$$

και

$$v_2 = Q_k \hat{h}_2 = q_1 \cdot \hat{h}_{2,1} + q_2 \cdot \hat{h}_{2,2} + q_3 \cdot \hat{h}_{2,3}$$

$$v_2 = \frac{0.7539}{\sqrt{3}} \cdot \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} - \frac{0.1307}{\sqrt{6}} \begin{pmatrix} 1 \\ -2 \\ 1 \end{pmatrix} + \frac{0.6439}{\sqrt{762}} \begin{pmatrix} 19 \\ -20 \\ 1 \end{pmatrix} = \begin{pmatrix} 0.8251 \\ 0.0755 \\ 0.4052 \end{pmatrix}$$

και

$$v_3 = Q_k \hat{h}_3 = q_1 \cdot \hat{h}_{3,1} + q_2 \cdot \hat{h}_{3,2} + q_3 \cdot \hat{h}_{3,3}$$

$$v_3 = \frac{0.1196}{\sqrt{3}} \cdot \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} + \frac{0.6704}{\sqrt{6}} \begin{pmatrix} 1 \\ -2 \\ 1 \end{pmatrix} - \frac{0.7323}{\sqrt{762}} \begin{pmatrix} 19 \\ -20 \\ 1 \end{pmatrix} = \begin{pmatrix} -0.1613 \\ 0.0522 \\ 0.3162 \end{pmatrix}$$

Δηλαδή οι προσεγγιστικές τιμές που υπολογίζει ο προαναφερόμενος αλγόριθμος είναι :

$$\tilde{\lambda}_1 = 8.0808 > \tilde{\lambda}_2 = 2.3580 > \tilde{\lambda}_3 = -0.3182 \quad (1.22)$$

υπενθυμίζοντας ότι οι ιδιοτιμές, που υπολογίζονται σε περιβάλλον Matlab έκδοσης R2023, είναι:

$$\ell_1 = 6.0576 > \ell_2 = 1.2563 > \ell_3 = -1.3140$$

Συγκρίνοντας τις προσεγγιστικές τιμές $\tilde{\lambda}_1, \tilde{\lambda}_2, \tilde{\lambda}_3$ του πίνακα A , που αναγράφονται στην (1.22) και υπολογίζονται από τις ιδιοτιμές του πίνακα Hessenberg ακολουθώντας τη Μέθοδο Arnoldi μέσω του Αλγορίθμου 1.10 με τις προσεγγιστικές τιμές $\lambda_1, \lambda_2, \lambda_3$, που υπολογίζονται με την τροποποιημένη Μέθοδο των Δυνάμεων (βλέπε, Παράδειγμα 1.6 και

την συνέχεια του σελ. 25) συμπεραίνουμε ότι αυτές οι τιμές έχουν μεγάλη απόκλιση μεταξύ τους. Επομένως, οι προσεγγιστικές ιδιοτιμές του πίνακα Hessenberg, που αναγράφονται στην (1.22), δεν αποτελούν προσεγγιστική λύση των ιδιοτιμών του πίνακα A. Συνεπώς, η Μέθοδος Arnoldi χρειάζεται κάποια διόρθωση για παράδειγμα αύξηση των επαναλήψεων κ.α., το οποίο δεν αποτελεί αντικείμενο μελέτης της παρούσας διπλωματικής εργασίας.

Κεφάλαιο 2^ο

Αλγόριθμος Lanczos

2.1 Προέλευση αλγορίθμου και εφαρμογές

Οι αλγόριθμος Lanczos αποτελεί μια από τις σημαντικότερες υπολογιστικές καινοτομίες των αρχών της δεκαετίας του 1950 για την επίλυση γραμμικών συστημάτων εξισώσεων και ιδιοπροβλημάτων. Η μέθοδος αυτή άρχισε να χρησιμοποιείται ευρέως μόλις στα μέσα της δεκαετίας του 1970. Λίγο αργότερα, οι τεράστιες μνήμες υπολογιστών κατέστησαν δυνατή την χρήση αυτών των μεθόδων για την επίλυση προβλημάτων που δεν μπορούσαν να επιλυθούν με οποιονδήποτε άλλο προσεγγιστικό τρόπο και με μικρότερο ή τον ίδιο υπολογιστικό φόρτο. Από τότε, ο αλγόριθμος έχει βελτιωθεί περαιτέρω και έχει γίνει βασικό εργαλείο για την επίλυση μιας μεγάλης ποικιλίας προβλημάτων σε μια μεγάλη ποικιλία αρχιτεκτονικών υπολογιστών, [6], [7].

Ο δημιουργός του αλγορίθμου είναι ο Cornelius Lanczos ο οποίος, γεννήθηκε το 1893 στην Ουγγαρία. Μετά τις σπουδές του στη Βουδαπέστη, ειδικεύθηκε στις εξισώσεις του Maxwell και στη θεωρητική φυσική. Λόγω πολιτικών συνθηκών, μετανάστευσε πρώτα στη Γερμανία, όπου συνεργάστηκε με τον Αϊνστάιν, και αργότερα στις Ηνωμένες Πολιτείες. Μέχρι τις αρχές της δεκαετίας του 1940 ο Lanczos κατείχε τρεις καθηγητικούς τίτλους στο Πανεπιστήμιο Purdue. Ο τελευταίος τίτλος τον οδήγησε στην εταιρεία Boeing κατά τη διάρκεια του πολέμου, όπου έδωσε διαλέξεις σε μηχανικούς για διάφορα θέματα εφαρμοσμένων μαθηματικών. Το 1949 εντάχθηκε στο Ινστιτούτο Αριθμητικής Ανάλυσης του Εθνικού Γραφείου Προτύπων, με έδρα το UCLA. Κατά τη διάρκεια αυτής της περιόδου ανέπτυξε τη μεθοδό του για τον υπολογισμό ιδιοτιμών, η οποία θεωρείται μια από τις σημαντικότερες εργασίες της διεθνούς αρθρογραφίας αναφορικά με τους υπολογισμούς πινάκων. Αν και δεν κατάφερε να επιλύσει πλήρως το πρόβλημα της ορθογωνιότητας της μεθόδου του, ακόμη και αυτή η πτυχή της είναι επίσης σημαντική. Αυτή η απώλεια του προβλήματος της ορθογωνιότητας ενέπνευσε πολλούς ερευνητές να αναλύσουν περαιτέρω την μέθοδο, με αποτέλεσμα σημαντικές βελτιώσεις και συνακόλουθες ανακαλύψεις, [12].

Παρακάτω παραθέτουμε κάποιες εφαρμογές, όπου υλοποιείται ο αλγόριθμος Lanczos, για την επίλυση προβλημάτων που εμφανίζονται στη μηχανική [15], και στα συνελκτικά δίκτυα [13].

➤ Αλγόριθμος Lanczos-εφαρμογή στη μηχανική.

Στην εργασία [15] παρουσιάζεται μια νέα προσέγγιση για τη χρήση της αμφίπλευρης αναδρομής Lanczos στην επίλυση του προβλήματος εύρεσης ιδιοτιμής. Η μεθοδολογία που προτείνεται διατηρεί την αρχική μορφή των τετραγωνικών προβλημάτων, χωρίς να απαιτείται η χρήση της παραδοσιακής μεθόδου αύξησης του πίνακα, η οποία

μετασχηματίζει το πρόβλημα σε γραμμικό πρόβλημα ιδιοτιμών τάξης $2n$. Βασισμένη στην ιδέα της κατασκευής ενός «χαρακτηριστικού πίνακα» που οδηγεί σε έναν «τετραγωνικό πίνακα ιδιοτιμών», η αναδρομή Lanczos αναπτύχθηκε ως μέσο επίλυσης του εν λόγω προβλήματος. Στη συνέχεια, παρουσιάζεται η εφαρμογή του αλγορίθμου Lanczos σε ένα πρόβλημα υπολογισμού ιδιοσυχνοτήτων μιας απλά υποστηριζόμενης δοκού με απόσβεση. Το συγκεκριμένο πρόβλημα αφορά την επίλυση ενός προβλήματος ιδιοτιμών που σχετίζεται με δυναμικές ταλαντώσεις. Ο αλγόριθμος Lanczos εφαρμόζεται για τον υπολογισμό των ιδιοσυχνοτήτων του συστήματος. Τα αποτελέσματα δείχνουν ότι οι υπολογισμένες ιδιοσυχνότητες συμφωνούν σε μεγάλο βαθμό με τις αναλυτικές λύσεις, με μικρές αποκλίσεις που αποδίδονται σε σφάλματα διακριτοποίησης. Συνολικά, ο αλγόριθμος Lanczos αποδεικνύεται γρήγορος και ακριβής, ακόμη και σε συστήματα που περιλαμβάνουν αποσβεστικές δυνάμεις.

➤ **Αλγόριθμος Lanczos-εφαρμογή στα συνελκτικά δίκτυα.**

Στην εργασία [13] το LanczosNet είναι ένα συνελκτικό νευρωνικό δίκτυο γραφημάτων (GNN), το οποίο ενσωματώνει τον αλγόριθμο Lanczos για μια αποδοτική φασματική προσέγγιση πολλαπλής κλίμακας μοντελοποιώντας τη μελέτη σε κατάλληλο γράφημα. Το μοντέλο βασίζεται στο μετασχηματισμό του πίνακα του γραφήματος σε ισοδύναμο πίνακα τριδιαγωνίας μορφής, ο εν λόγω μετασχηματισμός παράγεται μέσω του αλγορίθμου Lanczos, επιτρέποντας μια προσέγγιση χαμηλής κατάταξης του πίνακα Laplacian. Με αυτόν τον τρόπο, διευκολύνεται η συλλογή πληροφορίας σε πολλαπλές κλίμακες με χαμηλό υπολογιστικό κόστος, γεγονός που το καθιστά κατάλληλο για την ανάλυση μεγάλων και πολύπλοκων γραφημάτων. Επιπλέον, το LanczosNet χρησιμοποιεί φασματικά φίλτρα με δυνατότητα εκμάθησης, ενισχύοντας τη χωρητικότητα του μοντέλου στην αναγνώριση δομικών χαρακτηριστικών του γραφήματος, σε αντίθεση με τα παραδοσιακά GNNs που βασίζονται σε σταθερά φίλτρα. Η αποδοτικότητα του μοντέλου αξιολογήθηκε στο σύνολο δεδομένων QM8, που αφορά την πρόβλεψη μοριακών ιδιοτήτων στον τομέα της κβαντικής χημείας. Το LanczosNet πέτυχε αποτελέσματα κορυφαίας απόδοσης στην πρόβλεψη των ενεργειών ηλεκτρονικών φασμάτων, αποδεικνύοντας ότι η ενσωμάτωση του Lanczos οδηγεί σε μοντέλα με υψηλή ακρίβεια και μειωμένη υπολογιστική πολυπλοκότητα. Η μελέτη καταλήγει ότι η φασματική προσέγγιση πολλαπλής κλίμακας που προτείνεται αποτελεί μια ιδανική λύση για εφαρμογές, όπως είναι η μοριακή πρόβλεψη και η ανάλυση γραφημάτων μεγάλης κλίμακας.

2.2 Κλασσικός Αλγόριθμος Lanczos

Έστω ένας συμμετρικός και αραιός πίνακας $A \in \mathbb{R}^{n \times n}$ μεγάλων διαστάσεων για τον οποίο αναζητούνται μερικές από τις μεγαλύτερες ή/και τις μικρότερες ιδιοτιμές του, δηλαδή η αναζήτηση αφορά στις ιδιοτιμές που βρίσκονται στα δύο άκρα του φάσματος, οι οποίες λέγονται ακραίες ιδιοτιμές. Υπενθυμίζεται ότι η ιδιότητα της συμμετρίας του πίνακα A διασφαλίζει πραγματικές ιδιοτιμές [9], [23], οι οποίες υποθέτουμε ότι διατάσσονται :

$$\ell_1 \geq \ell_2 \geq \dots \geq \ell_n \quad (2.1)$$

Το πρόβλημα αναζήτησης των ιδιοτιμών στη (2.1) αντιμετωπίζεται με την υλοποίηση του Κλασσικού Αλγορίθμου Lanczos ή τροποποιημένων εκδόσεών του που έχουν στόχο την βελτιστοποίηση της ακρίβειας και την ελαχιστοποίηση της πολυπλοκότητας του προβλήματος.

Ο κλασσικός αλγόριθμος παράγει, για τις διάφορες τιμές του m με $m < n$, μια ακολουθία τριδιαγώνιων συμμετρικών πινάκων $T_m \in \mathbb{R}^{m \times m}$ με την ιδιότητα οι ιδιοτιμές του T_m , που υπολογίζονται και στη συνέχεια σημειώνονται $\lambda_k(T_m) := \lambda_k$, $k = 1, 2, \dots, m$, με

$$\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_m, \quad (2.2)$$

να αποτελούν προσδευτικά όλο και καλύτερες εκτιμήσεις των πραγματικών και διατεταγμένων ιδιοτιμών του A , όπως αυτές σημειώνονται στη (2.1).

Ένας τρόπος για να κατανοήσουμε την αναγκαιότητα ανάπτυξης νέας μεθοδολογίας, αυτή που αναπτύχθηκε για τον αλγόριθμο Lanczos, είναι να θυμηθούμε τα μειονεκτήματα της Μεθόδου των Δυνάμεων, που εξετάστηκαν στην Ενότητα 1.2. Θυμίζουμε ότι, μπορούμε να χρησιμοποιήσουμε τη Μέθοδο των Δυνάμεων για να βρούμε τη μεγαλύτερη (κατά απόλυτη τιμή) ιδιοτιμή λ_1 και ένα αντίστοιχο ιδιοδιάνυσμα u_1 και στην περίπτωση αυτή ο ρυθμός σύγκλισης υπαγορεύεται από το λόγο $\left| \frac{\lambda_2}{\lambda_1} \right|^m$, όπου λ_2 η ιδιοτιμή με τη δεύτερη μεγαλύτερη απόλυτη τιμή. Η Μέθοδος των Δυνάμεων (απλή και τροποποιημένη) είναι πολύ αργή, εκτός και αν υπάρχει αρκετά μεγάλη διαφορά μεγέθους μεταξύ των δύο τιμών λ_1, λ_2 , (βλέπε, Παρατήρηση 1.2(i), Παρατήρηση 1.7(ii)). Μια μέθοδος η οποία προσφέρει ακριβέστερη και ταχύτερη σύγκλιση χρησιμοποιώντας πιο αποδοτικά τα διαθέσιμα δεδομένα είναι η μέθοδος Lanczos, [6], [7].

Υλοποιώντας τον Αλγόριθμο 1.10-Μέθοδος Arnoldi για έναν **συμμετρικό** πίνακα A , παράγεται ο άνω Hessenberg πίνακας H_m , που είναι πίνακας συμμετρικός, και μάλιστα αποδεικνύεται ότι είναι τριδιαγώνιος, γι' αυτό στη συνέχεια σημειώνεται T_m και είναι της μορφής:

$$T_m = \begin{pmatrix} \alpha_1 & \beta_1 & \dots & 0 \\ \beta_1 & \alpha_2 & \dots & \vdots \\ \vdots & \vdots & \ddots & \beta_{m-1} \\ 0 & \dots & \beta_{m-1} & \alpha_m \end{pmatrix} \quad (2.3)$$

Οπότε, σύμφωνα με την (1.21), η υλοποίηση του Αλγορίθμου 1.10 για έναν δοσμένο συμμετρικό πίνακα $A \in \mathbb{R}^{n \times n}$ και ένα αρχικό διάνυσμα $v \in \mathbb{R}^{n \times 1}$ προσδιορίζει έναν πίνακα $U_m = (u_1 \ u_2 \ \dots \ u_m) \in \mathbb{R}^{n \times m}$ με ορθοκανονικές στήλες και έναν τριδιαγώνιο πίνακα $T_m \in \mathbb{R}^{m \times m}$, όπως ορίζεται στη (2.3), ικανοποιώντας την ακόλουθη ισότητα:

$$AU_m = U_m T_m \Leftrightarrow$$

$$A (u_1 \ u_2 \ \dots \ u_m) = (u_1 \ u_2 \ \dots \ u_m) \begin{pmatrix} \alpha_1 & \beta_1 & \dots & 0 \\ \beta_1 & \alpha_2 & \dots & \vdots \\ \vdots & \vdots & \ddots & \beta_{m-1} \\ 0 & \dots & \beta_{m-1} & \alpha_m \end{pmatrix} \quad (2.4)$$

Για τον προσδιορισμό των πινάκων U_m και T_m επιλέγουμε το u_1 τέτοιο ώστε $u_1 = \frac{v}{\|v\|_2}$, όπου v είναι το δοσμένο αρχικό διάνυσμα. Προφανώς το u_1 είναι μοναδιαίο διάνυσμα, από την παραπάνω επιλογή του.

Βήμα 1^ο - Υπολογισμός των α_1 , β_1 και u_2 : Συγκρίνοντας την πρώτη στήλη και στα δυο μέλη της ισότητας στη (2.4) έχουμε:

$$Au_1 = \alpha_1 u_1 + \beta_1 u_2 \quad (2.5)$$

Στην παραπάνω ισότητα (2.5) θέτουμε $\tilde{u}_2 = Au_1$, οπότε αυτή γράφεται :

$$\alpha_1 u_1 + \beta_1 u_2 = \tilde{u}_2 \quad (2.6)$$

Επειδή το u_1 είναι μοναδιαίο διάνυσμα, πολλαπλασιάζοντας τη (2.6) αριστερά επί το u_1^T και θεωρώντας δεδομένη την ορθογωνιότητα των διανυσμάτων u_1, u_2 προκύπτει:

$$\alpha_1 = u_1^T \tilde{u}_2$$

Θέτουμε

$$\tilde{u}_2 = \tilde{u}_2 - \alpha_1 u_1, \quad \text{και} \quad \beta_1 = \|\tilde{u}_2\|_2 \quad (2.7)$$

Συνδυάζοντας τη (2.6) με την αριστερή ισότητα στη (2.7) προκύπτει :

$$\beta_1 u_2 = \tilde{u}_2 \Leftrightarrow u_2 = \frac{\tilde{u}_2}{\beta_1}, \text{ αν } \beta_1 \neq 0$$

Από την τελευταία ισότητα και τη δεξιά ισότητα στη (2.7) είναι φανερό ότι ισχύει $\|u_2\|_2 = 1$.

Βήμα 2^ο - Υπολογισμός των α_2 , β_2 και u_3 : Συγκρίνοντας τη δεύτερη στήλη και στα δυο μέλη της ισότητας στη (2.4) έχουμε:

$$Au_2 = \beta_1 u_1 + \alpha_2 u_2 + \beta_2 u_3 \quad (2.8)$$

Στη (2.8) θέτουμε $\tilde{u}_3 = Au_2 - \beta_1 u_1$, οπότε αυτή γράφεται :

$$\alpha_2 u_2 + \beta_2 u_3 = \tilde{u}_3 \quad (2.9)$$

Επειδή το u_2 είναι μοναδιαίο διάνυσμα, πολλαπλασιάζοντας τη (2.9) αριστερά επί το u_2^T και θεωρώντας δεδομένη την ορθογωνιότητα των διανυσμάτων u_2, u_3 προκύπτει:

$$\alpha_2 = u_2^T \tilde{u}_3$$

Θέτουμε

$$\tilde{u}_3 = u_3 - \alpha_2 u_2, \quad \text{και} \quad \beta_2 = \|\tilde{u}_3\|_2 \quad (2.10)$$

Συνδυάζοντας τη (2.9) με την αριστερή ισότητα στη (2.10) προκύπτει :

$$\beta_2 u_3 = \tilde{u}_3 \Leftrightarrow u_3 = \frac{\tilde{u}_3}{\beta_2}, \text{ αν } \beta_2 \neq 0$$

Από την τελευταία ισότητα και τη δεξιά ισότητα στη (2.10) είναι φανερό ότι ισχύει $\|u_3\|_2 = 1$.

Τα υπόλοιπα βήματα είναι ανάλογα.

Είναι φανερό από τη (2.4) ότι για κάθε στήλη ισχύει η ακόλουθη αναδρομική σχέση:

$$Au_m = \beta_{m-1}u_{m-1} + \alpha_m u_m + \beta_m u_{m+1} \quad (2.11)$$

Χρησιμοποιώντας τη (2.11) και τη λογική που ακολουθήθηκε στα βήματα που περιεγράφηκαν παραπάνω, μπορούν να προσδιοριστούν όλα τα στοιχεία α_k και β_k του τριδιαγώνιου πίνακα T_m καθώς και τα διανύσματα u_{k+1} , για κάθε $k = 1, 2, \dots, m$.

Αλγόριθμος 2.1 (Κλασσικός Αλγόριθμος Lanczos)

Έστω $A \in \mathbb{R}^{n \times n}$ συμμετρικός πίνακας, $v \in \mathbb{R}^{n \times 1}$ αρχικό διάνυσμα και m θετικός ακέραιος με $m < n$.

Ο ακόλουθος αλγόριθμος υπολογίζει ένα σύνολο από $m + 1$ διανύσματα $\{u_1, u_2, \dots, u_{m+1}\}$ του $\mathbb{R}^{n \times 1}$ καθώς και τα στοιχεία α_k και β_k του συμμετρικού και τριδιαγώνιου πίνακα T_m στη (2.3). Θέτουμε $u_0 = 0 \in \mathbb{R}^{n \times 1}$, $\beta_0 = 0$ και $u_1 = \frac{v}{\|v\|_2}$.

for $k = 1, 2, \dots, m$

$$\tilde{u}_{k+1} = Au_k - \beta_{k-1}u_{k-1}$$

$$\alpha_k = u_j^T \tilde{u}_{j+1}$$

$$\tilde{u}_{k+1} = \tilde{u}_{k+1} - \alpha_k u_k$$

$$\beta_k = \|\tilde{u}_{k+1}\|_2 \text{ if } \beta_k = 0 \text{ stop}$$

$$u_{k+1} = \frac{\tilde{u}_{k+1}}{\beta_k}$$

end

Παρατήρηση 2.2

Ο υπολογισμός της συνολικής πολυπλοκότητας του Αλγορίθμου 2.1 για έναν **συμμετρικό** πίνακα $A \in \mathbb{R}^{n \times n}$ δίνεται [25] και ισούται με

$$\sum_{k=1}^m (n^2 + 5n) = m(n^2 + 5n) = mn^2 + 5mn \approx O(mn^2). \quad (2.12)$$

Από τη (2.12) συμπεραίνουμε ότι όταν ο αριθμός των επαναλήψεων m παίρνει μικρές τιμές σε σχέση με το n , n σημειώνει τη διάσταση του συμμετρικού πίνακα A , τότε η πολυπλοκότητα του Κλασσικού Αλγορίθμου Lanczos είναι $O(mn^2)$, δηλαδή είναι αρκετά καλή και θεωρείται αποδοτική. Όσο πλησιάζει η τιμή του m το n τόσο πιο αργός γίνεται ο αλγόριθμος, διότι η πολυπλοκότητα προσεγγίζει το $O(n^3)$.

Παράδειγμα 2.3

Έστω ο συμμετρικός πίνακας $A = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 6 & 2 \\ 1 & 2 & -1 \end{pmatrix}$ και το αρχικό διάνυσμα $v = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}$.

Η υλοποίηση του Αλγορίθμου 2.1 παρουσιάζεται στη συνέχεια βήμα-βήμα για τον παραπάνω πίνακα και το αρχικό διάνυσμα.

Να σημειώσουμε ότι στο συγκεκριμένο παράδειγμα έχουμε $n = 3$, οπότε πρέπει $m = 2$, δηλαδή ο τριδιαγώνιος πίνακας $T_m := T_2$ θα είναι ένας 2×2 της μορφής :

$$T_2 = \begin{pmatrix} \alpha_1 & \beta_1 \\ \beta_1 & \alpha_2 \end{pmatrix}$$

Αρχικά μετατρέπουμε το αρχικό διάνυσμα v σε μοναδιαίο

$$u_1 = \frac{v}{\|v\|_2} = \frac{1}{\sqrt{3}} \cdot \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} = \begin{pmatrix} \frac{1}{\sqrt{3}} & \frac{1}{\sqrt{3}} & \frac{1}{\sqrt{3}} \end{pmatrix}^T.$$

Βήμα 1^ο : Αντιστοιχεί στο $k = 1$, όπου υπολογίζονται τα μεγέθη α_1 , β_1 και u_2 .

Θέτουμε

$$u_2 = Au_1 = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 6 & 2 \\ 1 & 2 & -1 \end{pmatrix} \cdot \begin{pmatrix} \frac{1}{\sqrt{3}} \\ \frac{1}{\sqrt{3}} \\ \frac{1}{\sqrt{3}} \end{pmatrix} = \begin{pmatrix} \frac{3}{\sqrt{3}} \\ \frac{9}{\sqrt{3}} \\ \frac{2}{\sqrt{3}} \end{pmatrix},$$

οπότε έχουμε ότι το α_1 στοιχείο του τριδιαγώνιου πίνακα είναι:

$$\alpha_1 = u_1^T u_2 = \begin{pmatrix} \frac{1}{\sqrt{3}} & \frac{1}{\sqrt{3}} & \frac{1}{\sqrt{3}} \end{pmatrix} \cdot \begin{pmatrix} \frac{3}{\sqrt{3}} \\ \frac{9}{\sqrt{3}} \\ \frac{2}{\sqrt{3}} \end{pmatrix} = \frac{1}{3} (3 + 9 + 2) = \frac{14}{3} \cong 4.6667$$

Επίσης θέτουμε

$$\tilde{u}_2 = u_2 - \alpha_1 u_1 = \begin{pmatrix} \frac{3}{\sqrt{3}} \\ \frac{9}{\sqrt{3}} \\ \frac{2}{\sqrt{3}} \end{pmatrix} - \frac{14}{3} \begin{pmatrix} \frac{1}{\sqrt{3}} \\ \frac{1}{\sqrt{3}} \\ \frac{1}{\sqrt{3}} \end{pmatrix} = \frac{1}{3\sqrt{3}} \begin{pmatrix} -5 \\ 13 \\ -8 \end{pmatrix}$$

και μέσω αυτού υπολογίζεται το στοιχείο β_1 του τριδιαγώνιου πίνακα:

$$\beta_1 = \|\tilde{u}_2\|_2 = \sqrt{\left(\frac{-5}{3\sqrt{3}}\right)^2 + \left(\frac{13}{3\sqrt{3}}\right)^2 + \left(\frac{-8}{3\sqrt{3}}\right)^2} = \frac{\sqrt{86}}{3} \cong 3.0912$$

Τέλος, μέσω των παραπάνω ποσοτήτων υπολογίζεται το διάνυσμα :

$$u_2 = \frac{\tilde{u}_2}{\beta_1} = \frac{3}{\sqrt{86}} \frac{1}{3\sqrt{3}} \begin{pmatrix} -5 \\ 13 \\ -8 \end{pmatrix} = \frac{1}{\sqrt{258}} \begin{pmatrix} -5 \\ 13 \\ -8 \end{pmatrix}$$

Τα διανύσματα u_1, u_2 είναι ορθογώνια, επειδή ισχύει:

$$u_2^T u_1 = \frac{1}{\sqrt{3}\sqrt{258}} (-5 \quad 13 \quad -8) \cdot \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} = 0 \quad (2.13)$$

Βήμα 2^ο: Αντιστοιχεί στο $\mathbf{k} = 2$, όπου υπολογίζονται τα μεγέθη α_2 , β_2 και u_2 .
Θέτουμε

$$\hat{u}_3 = A u_2 - \beta_1 u_1 = \frac{1}{\sqrt{258}} \begin{pmatrix} 0 \\ 57 \\ 29 \end{pmatrix} - \frac{\sqrt{86}}{3\sqrt{3}} \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} = \frac{1}{3\sqrt{258}} \begin{pmatrix} -86 \\ 85 \\ 1 \end{pmatrix}$$

οπότε έχουμε ότι το α_2 στοιχείο του τριδιαγώνιου πίνακα είναι:

$$\alpha_2 = u_2^T \hat{u}_3 = \frac{1}{\sqrt{258}} (-5 \quad 13 \quad -8) \cdot \frac{1}{3\sqrt{258}} \begin{pmatrix} -86 \\ 85 \\ 1 \end{pmatrix} = \frac{509}{258} \cong 1.9729$$

Επίσης θέτουμε

$$\begin{aligned} \tilde{u}_3 &= \hat{u}_3 - \alpha_2 u_2 = \frac{1}{3\sqrt{258}} \begin{pmatrix} -86 \\ 85 \\ 1 \end{pmatrix} - \frac{509}{258\sqrt{258}} \begin{pmatrix} -5 \\ 13 \\ -8 \end{pmatrix} = \frac{1}{258\sqrt{258}} \begin{pmatrix} -4851 \\ 693 \\ 4158 \end{pmatrix} \\ &= \frac{1}{86\sqrt{258}} \begin{pmatrix} -1617 \\ 231 \\ 1386 \end{pmatrix} = \frac{3}{86\sqrt{258}} \begin{pmatrix} -539 \\ 77 \\ 462 \end{pmatrix} \cong \begin{pmatrix} -1.1706 \\ 0.1672 \\ 1.0034 \end{pmatrix} \end{aligned}$$

και μέσω αυτού υπολογίζεται το στοιχείο β_2 του τριδιαγώνιου πίνακα:

$$\beta_2 = \|\tilde{u}_3\|_2 = \frac{3}{86\sqrt{258}} \sqrt{(-539)^2 + (77)^2 + (462)^2} = \frac{3\sqrt{254947}}{86\sqrt{129}} \cong 1.5508$$

Τέλος, μέσω των παραπάνω ποσοτήτων υπολογίζεται το διάνυσμα :

$$u_3 = \frac{\tilde{u}_3}{\beta_2} = \frac{1}{\sqrt{2} \cdot \sqrt{254947}} \begin{pmatrix} -539 \\ 77 \\ 462 \end{pmatrix} \cong \begin{pmatrix} -0.7548 \\ 0.1078 \\ 0.6470 \end{pmatrix}$$

Σημειώστε ότι τα διανύσματα u_3, u_2 δεν είναι ορθογώνια, επειδή ισχύει:

$$u_3^T u_2 = \frac{1}{\sqrt{2} \cdot \sqrt{254947} \cdot \sqrt{258}} (-539 \quad 77 \quad 462) \cdot \begin{pmatrix} -5 \\ 13 \\ -18 \end{pmatrix} = \frac{-2310}{\sqrt{129} \cdot \sqrt{254947}} \neq 0 \quad (2.14)$$

Συνεπώς, υλοποιώντας τον Αλγόριθμο 2.1 προκύπτουν τα στοιχεία του τριδιαγώνιου και συμμετρικού πίνακα της μορφής (2.3)

$$T_2 = \begin{pmatrix} \frac{14}{3} & \frac{\sqrt{86}}{3} \\ \frac{\sqrt{86}}{3} & \frac{509}{258} \end{pmatrix}$$

καθώς και τα διανύσματα u_1, u_2 , που αποτελούν τις ορθοκανονικές στήλες του αντίστοιχου πίνακα

$$U_2 = \begin{pmatrix} 1/\sqrt{3} & -5/\sqrt{258} \\ 1/\sqrt{3} & 13/\sqrt{258} \\ 1/\sqrt{3} & -18/\sqrt{258} \end{pmatrix}, \quad (2.15)$$

επειδή ισχύει:

$$U_2^T U_2 = \begin{pmatrix} \frac{1}{\sqrt{3}} & \frac{1}{\sqrt{3}} & \frac{1}{\sqrt{3}} \\ -5 & 13 & -8 \\ \sqrt{258} & \sqrt{258} & \sqrt{258} \end{pmatrix} \begin{pmatrix} 1/\sqrt{3} & -5/\sqrt{258} \\ 1/\sqrt{3} & 13/\sqrt{258} \\ 1/\sqrt{3} & -18/\sqrt{258} \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} = I_2 \quad (2.16)$$

Παρακάτω παρατίθεται ο αντίστοιχος κώδικας σε περιβάλλον Matlab για τον Κλασσικό Αλγόριθμο Lanczos, ο οποίος βασίζεται στον Αλγόριθμο 2.1, με ορισμένες βελτιώσεις σε σχέση με τη θεωρητική του διατύπωση. Οι βελτιώσεις αυτές καθιστούν τον αλγόριθμο αριθμητικά σταθερότερο και καταλληλότερο κατά την υλοποίησή του στις εφαρμογές. Συγκεκριμένα, οι βελτιώσεις συνίσταται στα ακόλουθα :

- Έλεγχος σύγκλισης μέσω της συνθήκης "if beta < 1e-10", με σκοπό την αριθμητική σταθερότητα.
- Αποθήκευση του τριδιαγώνιου πίνακα $T := T_m$.
- Υπολογισμός των ιδιοτιμών και των αντίστοιχων ιδιοδιανυσμάτων του πίνακα T .

Κώδικας 2.4

```
% Classic Lanczos Algorithm
function [T, vT1, eT1, eT1s]=Lanczos_classic(A, v, m)
n = length(A);
U = zeros(n, m);
T = zeros(m, m);
U(:, 1) = v / norm(v);
beta = 0;
u_prev = zeros(n, 1);

for k = 1:m
    w = A * U(:, k) - beta * u_prev;
    alpha = U(:, k)' * w;
    w = w - alpha * U(:, k);
    beta = norm(w);
    T(k, k) = alpha;
    if k < m
        T(k, k+1) = beta;
        T(k+1, k) = beta;
    end

    if beta < 1e-10
        break;
    end
    u_prev = U(:, k);
    U(:, k+1) = w / beta;
```

```

end
[vT1,eT1]=eig(T);
eT1s=sort(diag(eT1));
end

```

Συνέχεια του Παραδείγματος 2.3. Θεωρούμε τον τριδιαγώνιο και συμμετρικό πίνακα

$$T \equiv T_2 = \begin{pmatrix} \frac{14}{3} & \frac{\sqrt{86}}{3} \\ \frac{\sqrt{86}}{3} & \frac{509}{258} \end{pmatrix}$$

που υπολογίστηκε στο Παράδειγμα 2.3 με την υλοποίηση του Κλασσικού Αλγορίθμου Lanczos.

Οι ιδιοτιμές του πίνακα T υπολογίζονται ως ρίζες της εξίσωσης

$$\det(T - \lambda I) = 0 \Leftrightarrow \det \begin{pmatrix} \frac{14}{3} - \lambda & \frac{\sqrt{86}}{3} \\ \frac{\sqrt{86}}{3} & \frac{509}{258} - \lambda \end{pmatrix} = 0,$$

από όπου προκύπτουν οι τιμές (τοποθετημένες σε φθίνουσα σειρά) :

$$\lambda_1(T) := \lambda_1 = 6.6917 \text{ και } \lambda_2(T) := \lambda_2 = -0.0521 \quad (2.17)$$

Για κάθε ιδιοτιμή $\lambda_i(T) = \lambda_i$, $i = 1, 2$ υπολογίζονται τα αντίστοιχα ιδιοδιανύσματα t_i του πίνακα T ως λύση του γραμμικού συστήματος

$$(T - \lambda_i I)t_i = 0 \Leftrightarrow \begin{pmatrix} \frac{14}{3} - \lambda & \frac{\sqrt{86}}{3} \\ \frac{\sqrt{86}}{3} & \frac{509}{258} - \lambda \end{pmatrix} \cdot \begin{pmatrix} y_1 \\ y_2 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \text{ για κάθε } i = 1, 2.$$

- Για την ιδιοτιμή λ_1 προκύπτει ως αντίστοιχο ιδιοδιάνυσμα το $t_1 = \begin{pmatrix} -0.8365 \\ -0.5480 \end{pmatrix}$.
- Για την ιδιοτιμή λ_2 προκύπτει ως αντίστοιχο ιδιοδιάνυσμα το $t_2 = \begin{pmatrix} 0.5480 \\ -0.8365 \end{pmatrix}$.

Υλοποιώντας τον Κώδικα 2.4 σε Matlab (έκδοση R2023) για τα δεδομένα του Παραδείγματος 2.3 (βλέπε, τον συμμετρικό πίνακα και το αρχικό διάνυσμα) υπολογίζονται εκτός από τα στοιχεία του τριδιαγώνιου και συμμετρικού πίνακα T και τα ιδιοποσά του, τα οποία εμφανίζονται ως ακολούθως :

$vT1 =$

$$\begin{pmatrix} 0.5480 & -0.8365 \\ -0.8365 & -0.5480 \end{pmatrix}$$

$$eT1 =$$

$$\begin{pmatrix} -0.0521 & 0 \\ 0 & 6.6917 \end{pmatrix}$$

Οι ιδιοτιμές του T εμφανίζονται ως διαγώνια στοιχεία στον $eT1$ και τα αντίστοιχα ιδιοδιανύσματα εμφανίζονται ως στήλες στο $vT1$. Τα αποτελέσματα αυτά επαληθεύουν τα ιδιοποσά του πίνακα T , που υπολογίστηκαν παραπάνω με τη χρήση του ορισμού των αντίστοιχων μεγεθών.

Επιπλέον, τοποθετώντας ως στήλες τα ορθοκανονικά διανύσματα u_1, u_2 , που υπολογίζει ο Κώδικας 2.4, παράγεται ο πίνακας U_2 στη (2.15), ο οποίος οδηγεί σε μετασχηματισμό ομοιότητας τον πίνακα A μέσω της (2.4). Πράγματι, πολλαπλασιάζοντας αριστερά τη (2.4) επί U_2^T και λόγω της (2.16), $U_2^T U_2 = I_2$, καταλήγουμε :

$$T = U_2^T A U_2 \quad (2.18)$$

Συνεπώς, μέσω του μετασχηματισμού ομοιότητας στη (2.18) συμπεραίνουμε ότι οι ιδιοτιμές του πίνακα T στη (2.17) αποτελούν προσεγγιστικές τιμές των πραγματικών ιδιοτιμών του πίνακα A , γεγονός που επαληθεύεται στο συγκεκριμένο παράδειγμα, λαμβάνοντας υπόψη ότι οι διατεταγμένες ιδιοτιμές του πίνακα A είναι:

$$\ell_1 = 6.7922 > \ell_2 = 0.9367 > \ell_3 = -1.7289 \quad (2.19)$$

Συγκρίνοντας τις ιδιοτιμές των πινάκων A και T στις (2.19) και (2.17) αντίστοιχα δημιουργείται το ερώτημα ποια είναι η σχέση που συνδέει τις ιδιοτιμές των δύο πινάκων, ιδιαίτερα της μεγαλύτερης ιδιοτιμής τους, η απάντηση του οποίου δίνεται στην πρόταση που ακολουθεί.

Υπενθυμίζεται ότι η φασματική ακτίνα ενός πίνακα $A \in \mathbb{R}^{n \times n}$ ορίζεται :

$$\rho(A) = \max\{|\lambda|: \lambda \text{ είναι ιδιοτιμή του } A\} \quad (2.20)$$

Πρόταση 2.5

Έστω ένας συμμετρικός πίνακας $A \in \mathbb{R}^{n \times n}$, ένας πίνακας $U_m \in \mathbb{R}^{n \times m}$ με ορθοκανονικές στήλες με $m \leq n$ και $T_m \in \mathbb{R}^{m \times m}$ ένας συμμετρικός τριδιαγώνιος πίνακας για τον οποίο ισχύει $AU_m = U_m T_m$. Τότε

$$\rho(T_m) \leq \rho(A). \quad (2.21)$$

Η ισότητα στην (2.21) ισχύει μόνο στην περίπτωση που $m = n$.

Απόδειξη: Αρχικά πολλαπλασιάζοντας αριστερά την ισότητα $AU_m = U_m T_m$ επί U_m^T , επειδή $U_m^T U_m = I_m$, προκύπτει:

$$T_m = U_m^T A U_m \quad (2.22)$$

Από τους ορισμούς της φασματικής νόρμας πίνακα² και της φασματικής ακτίνας πίνακα στην (2.20) για τους συμμετρικούς πίνακες A, T_m αποδεικνύονται οι ακόλουθες ισότητες:

$$\rho(A) = \|A\|_2 \quad \text{και} \quad \rho(T_m) = \|T_m\|_2 \quad (2.23)$$

Συνδυάζοντας τις ισότητες στη (2.23) με την (2.22) και σύμφωνα με τον ορισμό της νόρμας αξιοποιώντας την υποπολλαπλασιαστική ιδιότητα της νόρμας πίνακα και $\|U_m\|_2 = \|U_m^T\|_2 = 1^3$, μπορούμε να γράψουμε

$$\rho(T_m) = \|T_m\|_2 = \|U_m^T A U_m\|_2 \leq \|U_m^T\|_2 \cdot \|A\|_2 \cdot \|U_m\|_2 = \|A\|_2 = \rho(A), \quad (2.24)$$

το οποίο αποδεικνύει την ανισοτική σχέση στη (2.21).

Για να ισχύει η ισότητα στη (2.24), άρα και στη (2.21), πρέπει ο πίνακας U_m να είναι ορθομοναδιαίος, το οποίο προϋποθέτει να είναι τετραγωνικός, $m = n$, με ορθοκανονικές στήλες, οπότε σε αυτήν την περίπτωση η φασματική νόρμα του πίνακα είναι ορθομοναδιαία αναλλοίωτη, που σημαίνει ότι ισχύει $\|U_n^T A U_n\|_2 = \|A\|_2$, [9, Exercise, p.296].

Παρατήρηση 2.6

- i. Οι κύριες διαφορές της Μεθόδου Arnoldi με την Μέθοδο Lanczos για συμμετρικό πίνακα είναι ότι ο πίνακας H_k αντιστοιχεί σε έναν τριδιαγώνιο πίνακα και η πιο σημαντική διαφορά έγκειται στην αποθήκευση μόνο τριών διανυσμάτων εκτός αν χρησιμοποιηθεί εκ νέου κάποια μορφή ορθογωνοποίησης, [21].
- ii. Ο Κλασσικός Αλγόριθμος Lanczos (βλέπε, Αλγόριθμο 2.1 και Κώδικα 2.4) εγγυάται, τουλάχιστον σε ακριβή αριθμητική, ότι οι στήλες του πίνακα U_m αποτελούν ορθοκανονική βάση του πραγματικού χώρου στηλών, δηλαδή τα διανύσματα $u_k, k = 1, 2, \dots, m$, είναι ορθογώνια. Πράγματι, η ακριβής ορθογωνιότητα αυτών των διανυσμάτων παρατηρείται μόνο κατά την αρχή της διαδικασίας. Από κάποιο k (σχετικά μικρό) και για όλες τις επόμενες τιμές τους, τα u_k δεν διατηρούν την ιδιότητα της ορθογωνιότητας, (βλέπε, Παράδειγμα 2.3, σχέσεις (2.13)-(2.14)). Έχει πραγματοποιηθεί αρκετή έρευνα που συνδέεται με την αναζήτηση τρόπων ανάκτησης της ορθογωνιότητας μελετώντας μεθόδους μερικής ή/και επιλεκτικής ορθογωνιότητας ή τουλάχιστον ορίζοντας τρόπους μείωσης των συνεπειών της.

Το πρόβλημα αυτό θα μας απασχολήσει στην επόμενη ενότητα, όπου θα μελετήσουμε και θα προτείνουμε βελτιωμένες εκδόσεις του Κλασσικού Αλγορίθμου Lanczos.

² Η φασματική νόρμα $\|\cdot\|_2$ του πίνακα $A \in \mathbb{R}^{m \times n}$ ορίζεται

$$\|A\|_2 = \max\{\sqrt{\lambda}: \lambda \text{ είναι ιδιοτιμή του } A^T A\},$$

βλέπε, [14, ορισμός στην (5.2.7), p. 281]. Στην περίπτωση που $A \in \mathbb{R}^{n \times n}$ ισχύει ο ίδιος ορισμός της $\|\cdot\|_2$, βλέπε [9, Definition 5.6.6].

³ Βλέπε [14, p. 283], όπου από την ιδιότητα στην (5.2.11) μπορούμε να γράψουμε

$$\|U_m\|_2^2 = \|U_m^T U_m\|_2 = \|I_m\|_2 = 1 \Rightarrow \|U_m\|_2 = 1$$

και μέσω της ιδιότητας στην (5.2.10) καταλήγουμε $\|U_m^T\|_2 = \|U_m\|_2 = 1$.

2.3 Βελτιωμένοι Αλγόριθμοι Lanczos

Τα σημαντικότερα χαρακτηριστικά και πλεονεκτήματα του αλγορίθμου Lanczos επηρεάζονται ή αναιρούνται όταν υλοποιείται σε περιβάλλοντα με αριθμητική πεπερασμένη ακρίβειας. Όπως είχε αναγνωρίσει και ο ίδιος ο Lanczos, τα διανύσματα της βάσης που υπολογίζονται από τον αλγόριθμο σταδιακά χάνουν την ορθογωνιότητά τους καθώς οι τιμές τους επηρεάζονται από σφάλματα στρογγυλοποιήσεων, με αποτέλεσμα να υπάρχει σφάλμα στον υπολογισμό των ιδιοτιμών και κατ' επέκταση να αποτυγχάνει η επίλυση του ιδιοπροβλήματος. Αυτό το γεγονός, μαζί με την έλευση της τελείως σταθερής μεθόδου τριδιαγωνοποίησης του Householder, εξηγεί γιατί οι ερευνητές της Αριθμητικής Ανάλυσης περιφρόνησαν τον αλγόριθμο Lanczos στις δεκαετίες του 1950 και του 1960. Ωστόσο, αυτό άλλαξε υπό την πίεση της αναγκαιότητας επίλυσης μεγάλων και αραιών ιδιοπροβλημάτων, μαζί με τις υπολογιστικές γνώσεις που εισήγαγε το 1971 ο Paige. Με πολύ λιγότερες επαναλήψεις από τις n , που χρειάζονται συνήθως για να βρεθούν οι βέλτιστες προσεγγιστικές τιμές των ακραίων ιδιοτιμών, η μέθοδος Lanczos έγινε περισσότερο δημοφιλής ως τεχνική αραιών πινάκων παρά ως ανταγωνιστής της προσέγγισης Householder [6], [7].

Το πρώτο βήμα για τη βελτίωση της συμπεριφοράς του Κλασσικού Αλγορίθμου Lanczos αφορά στον αριθμό των επαναλήψεων που απαιτούνται, στην ανάθεση των απολύτως απαραίτητων μεταβλητών διατηρώντας τα δεδομένα που αντιστοιχούν στις πραγματικές επαναλήψεις. Αυτό οδηγεί σε αποδοτικότερη διαχείριση μνήμης και μείωση περιττών υπολογισμών.

Αλγόριθμος 2.7 (Modified Lanczos Algorithm)

Έστω $A \in \mathbb{R}^{n \times n}$ συμμετρικός πίνακας, $v \in \mathbb{R}^{n \times 1}$ αρχικό διάνυσμα και m θετικός ακέραιος με $m < n$. Θέτουμε

$$u_1 = \frac{v}{\|v\|_2}$$

$$\beta_0 = 0$$

for $k = 1, 2, \dots, m$

$$u_{k+1} = Au_k - \beta_{k-1}u_{k-1}$$

$$\alpha_k = u_k^T u_{k+1}$$

$$u_{k+1} = u_{k+1} - \alpha_k u_k$$

$$\beta_k = \|u_{k+1}\|_2$$

if $\beta_k == 0$ then

"Invariant subspace under A found"

exit

end

$$u_{k+1} = \frac{u_{k+1}}{\beta_k}$$

end

Σε κάθε επανάληψη του αλγορίθμου, ο συντελεστής α_k υπολογίζεται ως προς ένα "τροποποιημένο" u_{k+1} . Παρόλο που αυτή η αλλαγή είναι χρήσιμη, δεν είναι αρκετή για να "θωρακίσει" τον αλγόριθμο από τα σφάλματα που προκαλούνται λόγω της πεπερασμένης ακρίβειας στους υπολογισμούς.

Στη συνέχεια παρατίθεται ο κώδικας σε περιβάλλον Matlab, που αποτελεί μια υλοποίηση του Αλγορίθμου 2.7 (Modified Lanczos Algorithm), διατηρώντας τη βασική μαθηματική του δομή, με ορισμένες βελτιώσεις ως προς το πλήθος των μεταβλητών, καθιστώντας τον κώδικα κατάλληλο για αριθμητικές εφαρμογές, οι οποίες μοντελοποιούνται σε συμμετρικούς πίνακες μεγάλης διάστασης και απαιτούν φασματική ανάλυση. Συγκεκριμένα, οι βελτιώσεις ως προς τον Κλασσικό Αλγόριθμο Lanczos (βλέπε, Αλγόριθμο 2.1 και Κώδικα 2.4) συνίσταται στα ακόλουθα:

- Αποθήκευση των ορθοκανονικών διανυσμάτων V .
- Εμφάνιση ενημερωτικού μηνύματος όταν εντοπιστεί αμετάβλητος υπόχωρος, προσφέροντας στον χρήστη πληροφορίες για το σημείο τερματισμού της διαδικασίας.
- Τροποποίηση των διαστάσεων των πινάκων T και V , διατηρώντας μόνο τα δεδομένα που αντιστοιχούν στις πραγματικά εκτελεσμένες επαναλήψεις. Αυτό οδηγεί σε αποδοτικότερη διαχείριση μνήμης και μείωση περιττών υπολογισμών.

Κώδικας 2.8

```
% Modified Lanczos Algorithm
function [T, vT, eT, eTs] = Lanczos_modified(A, v, m)
n = length(A);
V = zeros(n, m);
T = zeros(m, m);
V(:, 1) = v / norm(v);
beta = 0;
q_prev = zeros(n, 1);
for j = 1:m
    w = A * V(:, j) - beta * q_prev;
    alpha = V(:, j)' * w;
    w = w - alpha * V(:, j);
    T(j, j) = alpha;
    beta = norm(w);
    if beta == 0
        fprintf('Invariant subspace found at
iteration %d\n', j);
        T = T(1:j, 1:j);
        V = V(:, 1:j);
        return;
    end
end
```

```

    if j < m
        T(j, j+1) = beta;
        T(j+1, j) = beta;
    end
    q_prev = V(:, j);
    V(:, j+1) = w / beta;
end
[vT, eT] = eig(T);
eTs = sort(diag(eT));
end

```

Το δεύτερο βήμα για τη βελτίωση της συμπεριφοράς του Κλασσικού Αλγορίθμου Lanczos σε περιβάλλον με αριθμητική πεπερασμένης ακρίβειας αφορά στην αναδιοργάνωση των πράξεων που εκτελούνται σε κάθε επανάληψη. Συγκεκριμένα, σε κάθε βήμα το νέο διάνυσμα u_{k+1} ορθογωνοποιείται ως προς τα Au_k και u_{k-1} . Στον αλγόριθμο που ακολουθεί, η εφαρμογή της διαδικασίας επιτυγχάνεται ουσιαστικά μέσω δύο σταδίων της μεθόδου Gram–Schmidt, κατά τα οποία υπολογίζονται οι συντελεστές α_k και β_k . Ωστόσο, για την αντιμετώπιση σφαλμάτων στρογγυλοποίησης που ενδέχεται να προκύψουν κατά τη διάρκεια των υπολογισμών, προτείνεται η χρήση της τροποποιημένης εκδοχής της μεθόδου, γνωστής ως Modified Gram–Schmidt [28, Αλγόριθμος 4, Κεφ. 4.3.1]. Η συγκεκριμένη προσέγγιση διαφοροποιείται από τη βασική μέθοδο στο ότι οι υπολογισμένοι συντελεστές δεν αποθηκεύονται για μεταγενέστερη χρήση, αλλά εφαρμόζονται άμεσα. Ως αποτέλεσμα, η ορθογωνιότητα επιτυγχάνεται σταδιακά, με βάση ήδη διορθωμένα διανύσματα, γεγονός που συμβάλλει στη βελτίωση της αριθμητικής συμπεριφοράς του αλγορίθμου. Αν και θεωρητικά οι δύο εκδοχές είναι ισοδύναμες υπό συνθήκες ιδανικής αριθμητικής ακρίβειας, στην πράξη η Modified Gram–Schmidt προσφέρει αυξημένη σταθερότητα και αξιοπιστία σε περιβάλλοντα περιορισμένης αριθμητικής ακρίβειας, όπως αυτά που συναντώνται σε υπολογιστικά συστήματα.

Η απώλεια της ορθογωνιότητας παρατηρείται σε δύο σημεία της διαδικασίας:

- Στα διαδοχικά διανύσματα u_k και u_{k+1} , όπου ο αλγόριθμος προσπαθεί να διατηρήσει την καθετότητα μεταξύ αυτών των δύο διαδοχικών διανυσμάτων. Όμως, λόγω των σφαλμάτων πεπερασμένης ακρίβειας, αυτή η προσπάθεια αποτυγχάνει, ως αποτέλεσμα την παραγωγή διανυσμάτων που δεν ικανοποιούν τις ιδιότητες και τις σχέσεις που περιγράφονται θεωρητικά. Για την αντιμετώπιση αυτού του προβλήματος, μπορεί να χρησιμοποιηθεί η τεχνική της **τοπικής ορθογωνοποίησης**. Αυτή η τεχνική εφαρμόζεται σε κάθε επανάληψη του αλγορίθμου με στόχο τη βελτίωση της ορθογωνιότητας ανάμεσα στα δύο πιο πρόσφατα διανύσματα, u_k και u_{k+1} . Η υλοποίησή της βασίζεται σε ένα βήμα της διαδικασίας Gram - Schmidt, αξιοποιώντας ορθογώνιες προβολές. Η τεχνική αυτή δεν απαιτεί την αποθήκευση επιπλέον διανυσμάτων πέρα από αυτά που είναι ήδη διαθέσιμα στον αλγόριθμο. Συνεπώς, η επιβάρυνση περιορίζεται μόνο σε υπολογιστικό επίπεδο.

Προτείνεται όχι μόνο η ορθογωνοποίηση των u_k και u_{k+1} , αλλά και η ορθογωνοποίηση μεταξύ των u_{k+1} και u_{k-1} . Αυτή η προσέγγιση διασφαλίζει ακόμα μεγαλύτερη ακρίβεια. Ένα πλεονέκτημα της τοπικής ορθογωνοποίησης είναι ότι, εξασφαλίζει τα διαδοχικά διανύσματα της βάσης, που υπολογίζονται από τον αλγόριθμο, να είναι κάθετα μεταξύ τους μέχρι το όριο ακρίβειας της μηχανής, [28].

- Στα διανύσματα της βάσης, που έχουν υπολογιστεί από προηγούμενες επαναλήψεις. Θεωρητικά, τα διανύσματα u_k και u_{k+1} πρέπει να είναι ορθογώνια ως προς τα προηγούμενα διανύσματα βάσης u_j , $j = 1, 2, \dots, k-1$. Στη πράξη όμως, αυτή η ορθογωνιότητα χάνεται, ιδιαίτερα όταν ορισμένες τιμές Ritz αρχίζουν να συγκλίνουν. Το ζήτημα αυτό έχει σημαντική επίπτωση στην απόδοση του αλγορίθμου, καθώς αυξάνει σημαντικά τόσο τον υπολογιστικό φόρτο όσο και τη μνήμη που απαιτείται. Για την αντιμετώπιση αυτής της απώλειας ορθογωνιότητας, υπάρχουν διάφορες τεχνικές, όπως είναι: η **μερική ορθογωνοποίηση** και η **ολική ορθογωνοποίηση**.

- ο Η μερική ορθογωνοποίηση προτάθηκε από τον Simon και εστιάζει στη βελτίωση της ορθογωνιότητας των διανυσμάτων, u_j και u_{j+1} ως προς όλα τα προηγούμενα Lanczos διανύσματα, αλλά δεν εφαρμόζεται σε κάθε επανάληψη, [28]. Χρησιμοποιείται μια αναδρομική σχέση για την παρακολούθηση της απώλειας ορθογωνιότητας και πραγματοποιείται μόνο όταν κρίνεται απαραίτητη. Τα διανύσματα u_j και u_{j+1} ορθογωνοποιούνται, ώστε να διασφαλιστεί ότι το νέο διάνυσμα που παράγεται θα είναι ορθογώνιο στα προηγούμενα. Για την παρακολούθηση της ορθογωνιότητας των Lanczos διανυσμάτων ορίζονται οι μεταβλητές $\omega_{j,k} = u_j^T u_k$ και $\omega_j = \max_{1 \leq k < j} |\omega_{j,k}|$.

Όπως σημείωσαν οι Scott και Parlett, η διατήρηση της ημικαθετότητας, δηλαδή ένα επίπεδο ορθογωνιότητας κοντά στην ακρίβεια της μηχανής ϵ_s είναι συνήθως αρκετή για ακριβή αποτελέσματα. Η αναδρομική σχέση για τον υπολογισμό των $\omega_{j,k}$, χωρίς την ανάγκη υπολογισμού των εσωτερικών γινομένων $u_j^T u_k$ δίνεται από τη σχέση: με $1 \leq k < j$ και $\omega_{j,k+1} = \omega_{k+1,j}$.

$$\beta_{j+1} \omega_{j+1,k} = \beta_{k+1} \omega_{j,k+1} + (\alpha_k - \alpha_j) \omega_{j,k} + \beta_k \omega_{j,k-1} - \beta_j \omega_{j-1,k} + u_j^T f_k - u_k^T f_j$$

Επίσης οι όροι f_k και f_j σχετίζονται με τα σφάλματα στρογγυλοποίησης. Συνοψίζοντας, όταν η αναδρομή εντοπίσει απώλεια της ορθογωνιότητας, δηλαδή, $\omega_j > \sqrt{\epsilon_s}$ τα δύο πιο πρόσφατα Lanczos διανύσματα u_j και u_{j+1} ορθογωνοποιούνται ως προς όλα τα προηγούμενα, u_i , $i = 1, 2, \dots, j-1$.

- ο Η πλήρης επαναορθογωνοποίηση είναι η πιο απλή και αξιόπιστη μέθοδος για την αντιμετώπιση της απώλειας ορθογωνιότητας στον αλγόριθμο Lanczos. Σε αυτή την προσέγγιση, το διάνυσμα u_{i+1} ορθογωνοποιείται ρητά ως προς όλα τα προηγούμενα υπολογισμένα διανύσματα Lanczos. Αυτό περιλαμβάνει τον υπολογισμό των προβολών και την αφαίρεση των συνιστωσών κατά μήκος όλων των προηγούμενων διανυσμάτων, ακόμα και αυτών για τα οποία ο συντελεστής Fourier θα ήταν θεωρητικά μηδέν σε ιδανικές συνθήκες αριθμητικής. Το κύριο πλεονέκτημα της ολικής ορθογωνοποίησης είναι η ευρωστία της, αφού η ορθογωνιότητα διατηρείται με πλήρη ακρίβεια μηχανής. Σημειώστε ότι για να είναι αυτό αληθές μπορεί να είναι απαραίτητο να καταφύγουμε σε διπλή ορθογωνοποίηση. **Δύο μειονεκτήματα** αυτής

της τεχνικής είναι ότι όλα τα διανύσματα Lanczos πρέπει να διατηρούνται στην κύρια μνήμη (όχι στη δευτερεύουσα αποθήκευση όπως ορισμένες παλαιότερες υλοποιήσεις) και ότι το κόστος της ορθογωνοποίησης είναι υψηλό και αυξάνει όσο περισσότερα βήματα Lanczos εκτελούνται. Ένα άλλο μειονέκτημα εμφανίζεται όταν γίνεται παράλληλη υλοποίηση της μεθόδου, οπότε η απόδοση θα ήταν κακή αν μια τροποποιημένη παραλλαγή Gram-Schmidt χρησιμοποιείται για την ορθογωνοποίηση, [8].

Με βάση τα παραπάνω συμπεραίνουμε ότι, ο απλός αλγόριθμος Lanczos, όταν υλοποιείται σε υπολογιστικά συστήματα με πεπερασμένη αριθμητική ακρίβεια, αντιμετωπίζει σοβαρά προβλήματα.

Ένα από αυτά είναι ότι, ο αλγόριθμος συνεχίζει να εκτελείται, υπολογίζοντας συνεχώς επαναλαμβανόμενες ιδιοτιμές, ιδιαίτερα εκείνες που βρίσκονται στα άκρα του φάσματος του πίνακα. Αυτές οι ιδιοτιμές συνήθως έχουν ήδη υπολογιστεί σωστά στις αρχικές επαναλήψεις. Επίσης, καθώς η βάση του Krylov υποχώρου σταδιακά χάνει την ορθοκανονικότητά της, ο τριδιαγώνιος πίνακας T_j παύει να αντιπροσωπεύει σωστά την ορθογώνια προβολή του αρχικού πίνακα στον υπόχωρο $\text{span}(Q_j)$, με αποτέλεσμα τον υπολογισμό λανθασμένων ιδιοτιμών. Οι υλοποιήσεις του αλγορίθμου πρέπει οπωσδήποτε να λαμβάνουν υπόψιν και να αντιμετωπίζουν την απώλεια της ορθογωνιότητας των Lanczos διανυσμάτων. Μια υλοποίηση με χρήση της τοπικής και μερικής ορθογωνοποίησης παρουσιάζεται στον παρακάτω αλγόριθμο, [28].

Αλγόριθμος 2.9 (Lanczos με μερική και τοπική επαναορθογωνοποίηση)

Έστω $A \in \mathbb{R}^{n \times n}$ συμμετρικός πίνακας, $v \in \mathbb{R}^{n \times 1}$ αρχικό διάνυσμα και m θετικός ακέραιος με $m < n$. Θέτουμε

$$u_1 = \frac{v}{\|v\|_2}$$

$$\beta_0 = 0$$

for $j = 1, 2, \dots, m$

$$u_{j+1} = Au_j - \beta_{j-1}u_{j-1}$$

$$\alpha_j = u_j^T u_{j+1}$$

$$u_{j+1} = u_{j+1} - \alpha_j u_j$$

$$\beta_j = \|u_{j+1}\|_2$$

if $\beta_j == 0$ then

"Invariant subspace under A found"

exit

end if

$$u_{j+1} = \frac{u_{j+1}}{\beta_j}$$

Evaluate ω recurrence and perform partial reorthogonalization if necessary

Perform local reorthogonalization

end for

Η ενότητα ολοκληρώνεται με τον σχεδιασμένο στο πλαίσιο της παρούσας διπλωματικής εργασίας κώδικα σε Matlab, ο οποίος αποτελεί βελτιωμένη έκδοση του Κλασσικού Αλγορίθμου Lanczos βασισμένος στα οφέλη της μερικής και τοπικής επαναορθογωνοποίησης. Ο Κώδικας 2.10 αποτελεί μια υλοποίηση του Αλγορίθμου 2.9 (Lanczos με μερική και τοπική επαναορθογωνοποίηση), διατηρώντας τη βασική μαθηματική του δομή με στοχευμένες επεκτάσεις, που ενισχύουν την αριθμητική σταθερότητα και την αποδοτικότητα, με αποτέλεσμα να είναι κατάλληλος στην επίλυση προβλημάτων φασματικής ανάλυσης συμμετρικών πινάκων μεγάλης διάστασης.

Συγκεκριμένα, οι βελτιώσεις ως προς τον Κλασσικό Αλγόριθμο Lanczos και τον Modified Lanczos Αλγόριθμο (βλέπε, Αλγόριθμο 2.1, Κώδικα 2.4, Αλγόριθμο 2.7, Κώδικα 2.8) συνίσταται στα ακόλουθα:

- Υλοποίηση της διαδικασίας μερικής επαναορθογωνοποίησης, διορθώνοντας τη σταδιακή απώλεια ορθογωνιότητας μέσω προβολών του νέου διανύσματος πάνω στα προηγούμενα.
- Εφαρμογή κριτηρίου ανοχής "tol" για τον έλεγχο αναγκαιότητας της επαναορθογωνοποίησης, επιτυγχάνοντας ισορροπία μεταξύ ακρίβειας και υπολογιστικού κόστους.
- Αξιολόγηση του μεγέθους της παραμέτρου "beta", με χρήση το κριτήριο ανοχής αντί για απόλυτο μηδέν, ώστε να αντιμετωπίζονται τα σφάλματα κινητής υποδιαστολής και να επιτυγχάνεται ασφαλής τερματισμός της διαδικασίας.
- Κατασκευή του τριδιαγώνιου πίνακα T, αποθήκευση των ορθοκανονικών διανυσμάτων V και υπολογισμός των ιδιοποσών του πίνακα T.
- Περικοπή των πινάκων T και V πριν την έξοδο της συνάρτησης, διατηρώντας μόνο τα απαραίτητα δεδομένα, γεγονός που οδηγεί σε πιο αποδοτική χρήση μνήμης και επεξεργαστικών πόρων.

Κώδικας 2.10

```
% Reorthogonalization Lanczos Algorithm
function [T, vT, eT, eTs] =
Lanczos_reorthogonalization(A, v, m, tol)
n = length(A);
V = zeros(n, m);
T = zeros(m, m);
V(:, 1) = v / norm(v);
beta = 0;
q_prev = zeros(n, 1);
for j = 1:m
    w = A * V(:, j) - beta * q_prev;
    alpha = V(:, j)' * w;
    w = w - alpha * V(:, j);
    for k = 1:j
        reortho_correction = V(:, k)' * w;
```

```

        if abs(reortho_correction) > tol
            w = w - reortho_correction * V(:, k);
        end
    end
    beta = norm(w);
    T(j, j) = alpha;
    if j < m
        T(j, j+1) = beta;
        T(j+1, j) = beta;
    end
    if beta < tol
        T = T(1:j, 1:j);
        V = V(:, 1:j);
        break;
    end
    q_prev = V(:, j);
    V(:, j+1) = w / beta;
end
[vT, eT]=eig(T);
eTs = sort(diag(eT));
end

```

Κεφάλαιο 3^ο

Προσέγγιση ιδιοποσών του πίνακα αποστάσεων και Κλασσικός Αλγόριθμος Lanczos

3.1 Φασματική ανάλυση με τον Κλασσικό Αλγόριθμο Lanczos

Το μέρος του κώδικα, που συγκεντρωτικά παρατίθεται στο Παράρτημα Ι, που αναφέρεται στη φασματική ανάλυση ενός συμμετρικού πίνακα μεγάλων διαστάσεων, όπως είναι ο πίνακας αποστάσεων μεταξύ όλων των ατόμων στη δομή της πρωτεΐνης, αποτελείται από τα ακόλουθα μέρη:

- i. Εισαγωγή των δεδομένων από το RCSB (PDB ID)
- ii. Ανάγνωση των τρισδιάστατων συντεταγμένων των ατόμων
- iii. Υπολογισμός του πίνακα αποστάσεων μεταξύ των ατόμων
- iv. Υλοποίηση του Κλασσικού Αλγορίθμου Lanczos για να υπολογιστούν τα ιδιοποσά (ιδιοτιμές και ιδιοδιανύσματα) του τριδιαγώνιου πίνακα που παράγεται από τον πίνακα αποστάσεων, στοιχεία απαραίτητα για την ανάλυση της γεωμετρίας και της δομής της πρωτεΐνης

Στη συνέχεια αναλύονται τα βήματα που ακολουθούνται κατά τον σχεδιασμό των προαναφερόμενων μερών (i)-(iv) του κώδικα που παρατίθεται στο Παράρτημα Ι.

Βήμα 1^ο : Είσοδος από τον Χρήστη

Ζητείται από τον χρήστη να εισάγει το PDB ID (Protein Data Bank ID), έναν μοναδικό κωδικό που αντιστοιχεί σε μια πρωτεΐνη, π.χ. 3pl6, καθώς και να καθορίσει το πλήθος των ιδιοποσών m , που θα υπολογιστούν μέσω του Κλασσικού Αλγορίθμου Lanczos.

Βήμα 2^ο : Λήψη του αρχείου PDB από τον διακομιστή RCSB

Δημιουργείται η πλήρης διεύθυνση URL για την ανάκτηση του αρχείου από το API του RCSB PDB, με βάση το ID που εισήγαγε ο χρήστης και με την εντολή "websave", πραγματοποιείται η λήψη του αρχείου από το διαδίκτυο και η αποθήκευσή του σε τοπικό φάκελο. Η βάση δεδομένων PDB (Protein Data Bank) αποτελεί μια διεθνή πλατφόρμα αποθήκευσης τρισδιάστατων δομικών δεδομένων βιολογικών μακρομορίων, όπως οι πρωτεΐνες και τα νουκλεϊκά οξέα. Η πρόσβαση σε αυτά τα δεδομένα είναι ελεύθερη παγκοσμίως μέσω της ιστοσελίδας [31], παρέχοντας στους ερευνητές τη δυνατότητα μελέτης και ανάλυσης πλήθους βιομοριακών δομών.

Βήμα 3^ο : Ανάγνωση του αρχείου PDB και εξαγωγή συντεταγμένων

Μελετώντας το περιεχόμενο των αρχείων ".pdb", παρατηρείται ότι στις πρώτες γραμμές εμφανίζεται η κατηγορία στην οποία εμπίπτει το μόριο, βάσει κριτηρίων, όπως η λειτουργία, ο μεταβολικός ρόλος, η κυτταρική τοποθεσία, κ.ά., καθώς και ο τίτλος του μορίου, ο οποίος περιγράφει συνοπτικά το περιεχόμενο του αρχείου και βοηθά στη διάκριση του από παρόμοιες καταχωρήσεις μοριακών δομών. Αυτές οι πληροφορίες

σηματοδοτούνται στα αρχεία ".pdb" με τις λέξεις "HEADER" και "TITLE" αντίστοιχα. Συνεπώς, για να εξαχθεί ο τίτλος του μορίου, πρέπει να προσδιοριστούν οι γραμμές που αρχίζουν με τον όρο "TITLE", ώστε να αποθηκευτεί το τμήμα της γραμμής που ακολουθεί αυτή τη λέξη. Μετά τον τίτλο του μορίου, στις επόμενες γραμμές περιέχονται πρόσθετες πληροφορίες και πειραματικά δεδομένα, τα οποία δεν μας είναι απαραίτητα για τον στόχο του αλγορίθμου μας και επομένως παρακάμπτονται κατά την ανάγνωση των αρχείων ".pdb". Εκτός από τον τίτλο του μορίου, σημαντικές για την επίλυση του προβλήματος είναι οι πληροφορίες σχετικά με τη χημική σύσταση του βιομορίου, τα άτομα δηλαδή που συνθέτουν το μόριο, μαζί με τις αντίστοιχες συντεταγμένες τους και τον συνολικό αριθμό των ατόμων. Αυτές οι πληροφορίες περιέχονται σε ξεχωριστές γραμμές του αρχείου που αρχίζουν με τις λέξεις "ATOM" και "HETATM". Κάθε γραμμή που ξεκινά με την ένδειξη "ATOM" σε ένα αρχείο ".pdb" αντιπροσωπεύει τα πρότυπα άτομα που απαρτίζουν το βιομόριο και δίνονται απαραίτητες πληροφορίες για το συγκεκριμένο άτομο, όπως είναι οι συντεταγμένες των ατόμων στον τρισδιάστατο χώρο, ο τύπος του ατόμου, το υπόλειμμα στο οποίο ανήκει, η αλυσίδα κ.λ.π.

Αντίστοιχα, οι γραμμές "HETATM" αφορούν μη καθιερωμένα ή μη πολυμερή άτομα, όπως μόρια νερού ή άλλες μικρές ενώσεις, οι οποίες ενδέχεται να συνυπάρχουν με τις κύριες μοριακές δομές. Και οι δύο αυτές κατηγορίες παρέχουν, μεταξύ άλλων, πληροφορίες για το όνομα του ατόμου, το αμινοξικό κατάλοιπο στο οποίο ανήκει, το φορτίο του καθώς και τις συντεταγμένες του, [24].

Για την ανάγνωση δεδομένων δομής πρωτεΐνης από ένα αρχείο PDB στη Matlab, μπορεί να χρησιμοποιηθεί η συνάρτηση "**pdbread**", η οποία παρέχεται μέσω του εργαλείου Bioinformatics Toolbox. Αυτή η συνάρτηση επιτρέπει την ανάγνωση των δομικών δεδομένων μιας πρωτεΐνης και την αποθήκευσή τους σε μορφή κατάλληλη για περαιτέρω ανάλυση. Στο πλαίσιο της παρούσας διπλωματικής εργασίας, αξιοποιήθηκαν οι γραμμές του αρχείου που ξεκινούν με τη λέξη "ATOM", δηλαδή τα δεδομένα που αντιπροσωπεύουν τις συντεταγμένες των ατόμων, τα οποία βρίσκονται στο πεδίο "Model.Atom". Οι συντεταγμένες αυτές εξάγονται σε έναν πίνακα $n \times 3$, όπου n είναι ο αριθμός των ατόμων. Κάθε γραμμή του πίνακα περιέχει τις συντεταγμένες ενός ατόμου, ενώ κάθε στήλη αντιστοιχεί σε μία από τις τρεις χωρικές συντεταγμένες, που σημειώνονται στη συνέχεια X, Y, Z .

Βήμα 4ο : Υπολογισμός του πίνακα αποστάσεων

Ανάλογα με τη μορφή των δομικών δεδομένων της πρωτεΐνης, μπορεί να απαιτείται προ επεξεργασία, κανονικοποίηση ή εφαρμογή μετασχηματισμών. Στην παρούσα διπλωματική εργασία, τα δεδομένα μετατρέπονται σε πίνακα αποστάσεων, ο οποίος είναι θεμελιώδους σημασίας για την πρόβλεψη της δομής και της δυναμικής των πρωτεϊνών. Όταν ολοκληρωθεί η εξαγωγή των συντεταγμένων των ατόμων, αυτές χρησιμοποιούνται για τον υπολογισμό των αποστάσεων μεταξύ όλων των δυνατών ζευγών ατόμων στη δομή της πρωτεΐνης. Με αυτόν τον τρόπο δημιουργείται ο **πίνακας αποστάσεων**, ο οποίος σημειώνεται *distanceMatrix* και στους Κώδικες ως *A*, οι δε ιδιότητές του αναφέρονται στην συνέχεια, (βλέπε, Παρατήρηση 3.1).

Βήμα 5^ο : Υπολογισμός ιδιοποσών

Από τον πίνακα αποστάσεων, που έχει δημιουργηθεί για κάθε δομή πρωτεΐνης, και την υλοποίηση του Κλασσικού Αλγορίθμου Lanczos, που παρουσιάστηκαν στο Κεφάλαιο 2 της παρούσας διπλωματικής εργασίας (βλέπε, Αλγόριθμος 2.1-Classic και Κώδικας 2.4), υπολογίζονται οι ιδιοτιμές και τα αντίστοιχα ιδιοδιανύσματα ποικίλων διαστάσεων συμμετρικών τριδιαγώνιων πινάκων T_m με στόχο τη βαθύτερη κατανόηση και ανάλυση της πρωτεϊνικής δομής. Συγκεκριμένα, τα ιδιοδιανύσματα του πίνακα T_m αντιστοιχούν σε βασικές κατευθύνσεις στον τρισδιάστατο χώρο, κατά τις οποίες η πρωτεΐνη μπορεί να παραμορφωθεί ή να κινηθεί. Οι αντίστοιχες ιδιοτιμές παρέχουν πληροφορίες για τη σταθερότητα της δομής μιας πρωτεΐνης, καθώς κάθε μία από αυτές συνδέεται με ένα συγκεκριμένο ιδιοδιάνυσμα, δηλαδή μια κατεύθυνση στον τρισδιάστατο χώρο. Όπως αναφέρεται [10], μικρές ή μηδενικές ιδιοτιμές ενδέχεται να υποδεικνύουν περιοχές ευστάθειας ή δομικά τμήματα που μπορούν να παραμορφωθούν πιο εύκολα και ανάλογα τα ιδιοδιανύσματα αντιπροσωπεύουν τις κατευθύνσεις στις οποίες οι δομές αλλάζουν κατά τις κινήσεις ή τις παραμορφώσεις τους. Τέλος να υπενθυμίσουμε ότι γνωρίζοντας τα ιδιοποσά του T_m μέσω του μετασχηματισμού ομοιότητας (2.4) προσεγγίζονται τα ιδιοποσά του A , και σύμφωνα με την Πρόταση 2.5 όταν ενδιαφερόμαστε για την φασματική ακτίνα του A , για μεγάλες τιμές του m , η προσέγγισή της γίνεται βέλτιστη από την τιμή της φασματικής ακτίνας του T_m . Συνεπώς γίνεται φανερή η αναγκαιότητα κατασκευής αλγορίθμου που να υπολογίζει τις ιδιοτιμές και τα ιδιοδιανύσματα του T_m .

Παρατήρηση 3.1

Ο **πίνακας αποστάσεων**, distanceMatrix, είναι ένας συμμετρικός πίνακας διαστάσεων $n \times n$, όπου n είναι ο αριθμός των ατόμων. Κάθε στοιχείο του πίνακα στη θέση (i, j) με $1 \leq i, j \leq n$ εκφράζει την απόσταση μεταξύ δύο ατόμων, που το i -άτομο βρίσκεται στη θέση (X_i, Y_i, Z_i) και το j -άτομο βρίσκεται στη θέση (X_j, Y_j, Z_j) , η δε απόσταση μεταξύ τους σημειώνεται $d_{i,j}$ και δίνεται από την Ευκλείδεια νόρμα :

$$d_{i,j} = \sqrt{(X_i - X_j)^2 + (Y_i - Y_j)^2 + (Z_i - Z_j)^2} \quad (3.1)$$

Από την (3.1) είναι φανερό ότι ισχύουν: $d_{i,j} = d_{j,i} > 0$, για κάθε $i \neq j$, και $d_{i,i} = 0$. Συνεπώς, ο πίνακας αποστάσεων είναι ένας μη αρνητικός συμμετρικός πίνακας με μηδενικά διαγώνια στοιχεία. Λόγω συμμετρίας του πίνακα αποστάσεων οι ιδιοτιμές είναι πραγματικοί αριθμοί, που σημειώνονται ℓ_i , $i = 1, 2, \dots, n$ και είναι διατεταγμένοι σε αύξουσα σειρά ως ακολούθως :

$$\ell_1 \leq \ell_2 \leq \dots \leq \ell_n \quad (3.2)$$

Επειδή όλα τα στοιχεία του πίνακα $I_n + \text{distanceMatrix}$ είναι θετικοί αριθμοί, I_n σημειώνει τον $n \times n$ μοναδιαίο πίνακα, συμπεραίνουμε ότι ο πίνακας αποστάσεων είναι πίνακας μη αναγώγιμος (irreducible)⁴. Σύμφωνα με τη θεωρία Perron-Frobenius η μεγαλύτερη ιδιοτιμή του πίνακα αποστάσεων, η φασματική ακτίνα, που σημειώνεται

⁴ Ένας $n \times n$ πίνακας A με μη αρνητικά στοιχεία είναι μη αναγώγιμος αν και μόνο αν τα στοιχεία του πίνακα $(I_n + A)^{n-1}$ είναι θετικοί αριθμοί, [9, Theorem 6.2.23, Lemma 8.4.1].

$$\rho(\text{distanceMatrix}): = \ell_n,$$

είναι ένας πραγματικός θετικός αριθμός με αλγεβρική πολλαπλότητα ίση με ένα (ιδιοτιμή απλή) και το αντίστοιχο (μοναδικό) ιδιοδιάνυσμά της έχει συντεταγμένες μη αρνητικούς πραγματικούς αριθμούς, [9, Theorem 8.4.4]. Επομένως η (3.2) λόγω θετικότητας και μοναδικότητας της φασματικής ακτίνας γράφεται:

$$\ell_1 \leq \ell_2 \leq \dots \leq 0 < \dots < \ell_n \quad (3.3)$$

Επιπρόσθετα ο πίνακας αποστάσεων είναι πρωταρχικός (primitive)⁵, επειδή τα στοιχεία του πίνακα $(\text{distanceMatrix})^2$ είναι θετικοί αριθμοί. Επομένως, η ιδιότητα της πρωταρχικότητας⁶ του πίνακα αποστάσεων διασφαλίζει ότι για όλες τις υπόλοιπες ιδιοτιμές, ℓ_i , $i = 2, \dots, n-1$ εκτός της φασματικής ακτίνας του πίνακα ισχύει $-\ell_n < \ell_i < \ell_n$, από όπου συμπεραίνουμε ότι όλες οι ιδιοτιμές ανήκουν εντός του ανοικτού διαστήματος του πραγματικού άξονα με κέντρο το 0 και ακτίνα ίση με τη φασματική ακτίνα, δηλαδή

$$\ell_i \in (-\ell_n, \ell_n), \quad \text{για κάθε } i = 2, \dots, n-1. \quad (3.4)$$

Παράδειγμα 3.2

Θεωρούμε τα δεδομένα της δομής της πρωτεΐνης 141L, τα οποία ανακτώνται από [32] και εμφανίζονται σε γραμμές, για παράδειγμα:

ATOM	1	N	MET	A	1	44.099	-3.316	8.912	1.00	24.39	N
------	---	---	-----	---	---	--------	--------	-------	------	-------	---

Οι πληροφορίες που λαμβάνονται από την παραπάνω καταχώρηση είναι :

ATOM : Υποδεικνύει την καταχώρηση του ατόμου

1 : Σειριακός αριθμός του ατόμου

N : Τύπος του ατόμου (άτομο αζώτου σε αυτήν την περίπτωση)

MET : Όνομα του υπολείμματος (μεθειονίνη)

A : Αλυσίδα

1 : Αριθμός του υπολείμματος.

44.099 - 3.316 8.912 : Συντεταγμένες X, Y, Z του ατόμου στον $\mathbb{R}^3$

1.00 : Κατοχή (occupancy), η οποία δείχνει το ποσοστό του χρόνου που το άτομο καταλαμβάνει τη συγκεκριμένη θέση

24.39 : Παράγοντας θερμικής ταλάντωσης (B-factor), ο οποίος δείχνει την κινητικότητα ή την αβεβαιότητα της θέσης του ατόμου.

N : Συμβολισμός στοιχείου (άτομο αζώτου)

⁵ Ένας $n \times n$ πίνακας A με μη αρνητικά στοιχεία είναι πρωταρχικός (primitive) αν και μόνο αν τα στοιχεία του πίνακα A^k , $k \geq 1$, είναι θετικοί αριθμοί, [9, Theorem 8.5.2].

⁶ Ένας $n \times n$ πίνακας A με μη αρνητικά στοιχεία είναι πρωταρχικός (primitive) αν έχει μόνο μία ιδιοτιμή με μέτρο ίση με τη φασματική ακτίνα, [9, Definition 8.5.0]. Επομένως, οι υπόλοιπες ιδιοτιμές εκτός της φασματικής ακτίνας είναι εντός του κυκλικού δίσκου κέντρου 0 και ακτίνας ίσης με τη φασματική ακτίνα.

Συνολικά, οι συντεταγμένες των ατόμων της πρωτεΐνης 141L, που ανακτήθηκαν, εμφανίζονται στον ακόλουθο πίνακα `cord_141L` διάστασης 1293×3 ,

$$\text{cord_141L} = \begin{pmatrix} 44.099 & -3.316 & 8.912 \\ 43.663 & -1.947 & 8.96 \\ \vdots & \vdots & \vdots \\ 46.966 & -2.702 & 1.416 \\ 45.851 & -3.648 & 1.217 \end{pmatrix}$$

Υπενθυμίζεται ότι κάθε στήλη του `cord_141L` αντιστοιχεί στις αντίστοιχες συντεταγμένες X, Y, Z του τρισδιάστατου χώρου και κάθε σειρά αντιστοιχεί στις συντεταγμένες ενός ατόμου της πρωτεΐνης, έχοντας πρώτη γραμμή τις συντεταγμένες του ατόμου που περιεγράφηκε παραπάνω. Σύμφωνα με τις ιδιότητες που αναφέρονται στην Παρατήρηση 3.1, ο πίνακας αποστάσεων `distanceMatrix`, που δημιουργείται από τον `cord_141L`, είναι πίνακας διαστάσεων 1293×1293 με διαγώνια στοιχεία ίσα με μηδέν. Κάθε μη διαγώνιο στοιχείο του `distanceMatrix` εκφράζει την ευκλείδεια απόσταση μεταξύ ενός ζεύγους ατόμων της πρωτεΐνης και υπολογίζεται από την (3.1), δηλαδή:

$$d_{1,2} = \sqrt{(44.099 - 43.663)^2 + (-3.316 + 1.947)^2 + (8.912 - 8.96)^2} = 1.4376$$

$$d_{1,1292} = \sqrt{(44.099 - 46.966)^2 + (-3.316 + 2.702)^2 + (8.912 - 1.416)^2} = 8.049$$

$$d_{1,1293} = \sqrt{(44.099 - 45.851)^2 + (-3.316 + 3.648)^2 + (8.912 - 1.217)^2} = 7.8989$$

$$d_{2,1292} = \sqrt{(43.663 - 46.966)^2 + (-1.947 + 2.702)^2 + (8.96 - 1.416)^2} = 8.2699$$

$$d_{2,1293} = \sqrt{(43.663 - 45.851)^2 + (-1.947 + 3.648)^2 + (8.96 - 1.217)^2} = 8.224$$

$$d_{1292,1293} = \sqrt{(46.966 - 45.851)^2 + (-2.702 + 3.648)^2 + (1.416 - 1.217)^2} = 1.4757$$

...

Ανάλογα υπολογίζονται και τα υπόλοιπα μη διαγώνια στοιχεία του πίνακα αποστάσεων και λαμβάνοντας υπόψη τη συμμετρία του, καταλήγουμε στον 1293×1293 πίνακα :

$$\text{distanceMatrix} = \begin{pmatrix} 0 & 1.4376 & \dots & 8.049 & 7.8989 \\ 1.4376 & 0 & \dots & 8.2699 & 8.224 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 8.049 & 8.2699 & \dots & 0 & 1.4757 \\ 7.8989 & 8.224 & \dots & 1.4757 & 0 \end{pmatrix}$$

3.2 Συναρτήσεις (functions)

Στην παρούσα ενότητα παρουσιάζονται και σχολιάζονται αναλυτικά οι συναρτήσεις που δημιουργήθηκαν στο πλαίσιο της παρούσας διπλωματικής εργασίας προκειμένου η υλοποίησή τους να συμβάλει στη φασματική ανάλυση των πινάκων αποστάσεων και δίνεται εφαρμογή στα δεδομένα της πρωτεΐνης 3p16.

3.2.1 Συνάρτηση Lanczos_classic

Η συνάρτηση (function), που ονομάζεται "Lanczos_classic" είναι ένας αλγόριθμος, που υλοποιείται για τον υπολογισμό τιμών που παρουσιάζουν τη βέλτιστη προσέγγιση των ιδιοποσών των $n \times n$ συμμετρικών πινάκων, όπως είναι οι πίνακες αποστάσεων A και ιδιαίτερα σε περιπτώσεις πολύ μεγάλων τιμών του n . Ο σχεδιασμός του αλγορίθμου βασίζεται στη μείωση των διαστάσεων του αρχικού συμμετρικού πίνακα αποστάσεων A σε έναν $m \times m$ τριδιαγώνιο πίνακα $T := T_m$ με προκαθορισμένη από τον χρήστη την τιμή του m με $m < n$, και στον υπολογισμό των ιδιοποσών του πίνακα T .

Αντίστοιχα με τον Κώδικα 2.4 δημιουργήθηκε ο ακόλουθος Κώδικας 3.3⁷ ως συνάρτηση (function) της Matlab, η οποία λαμβάνει:

- Ως είσοδο τον πίνακα αποστάσεων A και τον αριθμό m των επαναλήψεων, που θα εκτελεστούν για τον υπολογισμό των μη μηδενικών στοιχείων του πίνακα T .
- Ως έξοδο, η συνάρτηση επιστρέφει τα ιδιοποσά του πίνακα T . Συγκεκριμένα, τα ιδιοδιανύσματα του τριδιαγώνιου πίνακα T , συμβολίζονται vT , και οι ταξινομημένες σε αύξουσα σειρά ιδιοτιμές του πίνακα T , συμβολίζονται eT .

Κώδικας 3.3

```
%Classic Algorithm Lanczos
function [vT, eT] = Lanczos_classic(A, m)
n = size(A, 1);
V = zeros(n, m);
T = zeros(m, m);
v1 = rand(n, 1);
v1 = v1 / norm(v1);
V(:, 1) = v1;
beta = 0;
v_prev = zeros(n, 1);
for k = 1:m
    w = A * V(:, k) - beta * v_prev;
    alpha = V(:, k)' * w;
    w = w - alpha * V(:, k);
    beta = norm(w);
```

⁷ Βλέπε στο Παράρτημα Ι στις γραμμές 66 – 67.

```

T(k, k) = alpha;
if k < m
    T(k, k+1) = beta;
    T(k+1, k) = beta;
end
if beta < 1e-10
    disp('Lanczos method stopped due to low
    beta. ');
    break;
end
v_prev = V(:, k);
V(:, k+1) = w / beta;
end
[vT, eT_matrix] = eig(T);
eT = diag(eT_matrix);
[eT, idx] = sort(eT, 'ascend');
vT = vT(:, idx);
end

```

Παράδειγμα 3.4

Θεωρούμε τα δεδομένα της πρωτεΐνης 3p16 από [33], τα οποία εμφανίζονται στο Παράρτημα ΙΙΙ ως ένας πίνακας, `cord_3p16`, διάστασης 6405×3 . Ακολουθώντας τη μεθοδολογία υπολογισμού που παρουσιάστηκε στο Παράδειγμα 3.2 από τον `cord_3p16` δημιουργείται ο πίνακας αποστάσεων A , που είναι ένας 6405×6405 μη αρνητικός συμμετρικός πίνακας. Για τον προαναφερόμενο A και για $m = 3$ υλοποιείται ο Κώδικας 3.3 σε Matlab (έκδοση R2023), ο οποίος αποτελεί τις γραμμές 66 – 67 του κύριου κώδικα του Παραρτήματος Ι, οπότε υπολογίζονται τα στοιχεία του τριδιαγώνιου και συμμετρικού πίνακα

$$T_3 = 10^5 \begin{pmatrix} 2.2422 & 1.3854 & 0 \\ 1.3854 & 0.8096 & 0.1233 \\ 0 & 0.1233 & -0.3342 \end{pmatrix}, \quad (3.5)$$

καθώς και τα ιδιοποσά του.

➤ Οι διατεταγμένες σε αύξουσα σειρά **ιδιοτιμές** του T_3 στην (3.5) είναι:

$eT =$

```

1.0e+05 *
-0.3685
-0.0006
3.0867

```

➤ Τα αντίστοιχα **ιδιοδιανύσματα** του T_3 εμφανίζονται ως στήλες στο vT στη συνέχεια:

$$\begin{aligned}
 \mathbf{vT} = & \\
 & \begin{array}{ccc}
 0.1409 & 0.5013 & -0.8537 \\
 -0.2655 & -0.8116 & -0.5204 \\
 0.9538 & -0.3000 & -0.0188
 \end{array}
 \end{aligned}$$

Σχόλια 3.5

Ερμηνεία ιδιοποσών Παραδείγματος 3.4

- i. Η πρώτη και δεύτερη εμφανιζόμενη ιδιοτιμή στον $\mathbf{eT}$, έστω $\lambda_1 = -0.3685 \cdot 10^5$, $\lambda_2 = -0.0006 \cdot 10^5$ είναι μικρότερες από την τρίτη εμφανιζόμενη ιδιοτιμή, $\lambda_3 = 3.0867 \cdot 10^5$, που είναι η φασματική ακτίνα του πίνακα T_3 . Σύμφωνα με τα σχόλια στο 5^ο βήμα της Ενότητας 3.1 η φασματική ακτίνα του πίνακα T_3 αντιπροσωπεύει τη σημαντικότερη συνιστώσα του πίνακα αποστάσεων. Η συνιστώσα αυτή εξηγεί τη μεγαλύτερη διακύμανση στη γεωμετρία της πρωτεΐνης.
- ii. Η πρώτη και δεύτερη στήλη του $\mathbf{vT}$, στη συνέχεια σημειώνονται ως $\mathbf{v}_1, \mathbf{v}_2$, αντίστοιχα, αποτελούν τα αντίστοιχα ιδιοδιανύσματα των ιδιοτιμών λ_1, λ_2 , τα οποία περιγράφουν δευτερεύουσες κατευθύνσεις διακύμανσης, δηλαδή η σημασία τους είναι μικρότερη, όπως φαίνεται και από τη διάταξη των αντίστοιχων ιδιοτιμών τους.
- iii. Η τρίτη στήλη του $\mathbf{vT}$, $\mathbf{v}_3 = [-0.8537 \quad -0.5204 \quad -0.0188]^T$, που αντιστοιχεί στην φασματική ακτίνα του T_3 , δείχνει την κύρια κατεύθυνση της πρωτεΐνης στον τρισδιάστατο χώρο. Έχει τη μεγαλύτερη συνεισφορά στην περιγραφή της γεωμετρίας της πρωτεΐνης και εξηγεί τη μεγαλύτερη κατανομή των δεδομένων.
- iv. Χρησιμοποιώντας την εντολή `eigs`⁸ της Matlab υπολογίζονται τρεις από τις 6405 ιδιοτιμές του πίνακα αποστάσεων A , οι οποίες αναφέρονται στη συνέχεια:

$$\ell_1 = -1.2833 \cdot 10^5 < \ell_2 = -0.3695 \cdot 10^5 < \ell_3 = 3.0867 \cdot 10^5 \quad (3.6)$$
 Παρατηρούμε ότι οι ιδιοτιμές ℓ_1, ℓ_2 ανήκουν στο διάστημα $(-\ell_3, \ell_3)$ επαληθεύοντας την (3.4).
- v. Από τις ιδιοτιμές των πινάκων T_3 και A , που υπολογίστηκαν παραπάνω στα (i) και (iv), υπολογίζεται το σφάλμα της προσέγγισης των ιδιοτιμών, το οποίο είναι:

$$|\ell_1 - \lambda_1| = 0.9148 \cdot 10^5, \quad |\ell_2 - \lambda_2| = 0.3689 \cdot 10^5, \quad |\ell_3 - \lambda_3| = 0$$
 Παρατηρούμε ότι στην περίπτωση της φασματικής ακτίνας των πινάκων δεν υπάρχει σφάλμα στην προσέγγιση, από όπου συμπεραίνουμε ότι η προσέγγιση της φασματικής ακτίνας του A από την φασματική ακτίνα του T_3 είναι αρκετά ικανοποιητική, ενώ δεν μπορούν να προκύψουν ασφαλή συμπεράσματα από την προσέγγιση των υπολοίπων ιδιοτιμών του A από τις λ_1, λ_2 του T_3 μια και το σφάλμα είναι διάφορο του μηδενός. Εδώ χρειάζεται να σημειώσουμε ότι τα σφάλματα, που αντιστοιχούν στις δύο μικρότερες ιδιοτιμές των πινάκων, έχουν τάξη μεγέθους όση και οι αντίστοιχες ιδιοτιμές, δηλαδή 10^5 .
- vi. Σύμφωνα με την (2.12) η υπολογιστική πολυπλοκότητα του αλγορίθμου εξαρτάται από την τιμή της παραμέτρου m και μάλιστα είναι ανάλογη της διάστασης του πίνακα και της m . Επομένως όσο μικρότερη τιμή έχει η παράμετρος m τόσο μικρότερη είναι η συνολική πολυπλοκότητα των υπολογισμών, γεγονός που εξασφαλίζει μικρότερο

⁸ Βλέπε, στην Ενότητα 4.2 της παρούσας διπλωματικής εργασίας για την εντολή στην Matlab.

χρόνο εκτέλεσης του αλγορίθμου και απαιτεί λιγότερο χώρο στη μνήμη του συστήματος. Συνεπώς, ο αλγόριθμος Lanczos εξοικονομεί υπολογιστική ισχύ.

3.2.2 Συνάρτηση `analyze_eigenvectors`

Ο ακόλουθος Κώδικας 3.6⁹ δημιουργήθηκε ως συνάρτηση (function) της Matlab, που ονομάζεται "analyze_eigenvectors", η οποία χρησιμοποιείται για την ανάλυση και την οπτικοποίηση των τριών ιδιοδιανυσμάτων του 3×3 τριδιαγώνιου πίνακα T_3 , συνδυάζοντας αριθμητικά αποτελέσματα με γραφική αναπαράσταση στον πραγματικό χώρο. Παρέχει στον χρήστη σαφή αριθμητική πληροφορία και δημιουργεί ένα καθαρό και επεξηγηματικό τρισδιάστατο διάγραμμα για την οπτική αναπαράσταση των ιδιοδιανυσμάτων του $\mathbb{R}^3$.

Κατά την εκτέλεση της συνάρτησης:

- Τα ιδιοδιανύσματα πολλαπλασιάζονται με έναν συντελεστή κλιμάκωσης "**scale_factor**", ώστε να απεικονίζονται πιο ευδιάκριτα στο γράφημα.
- Για την αναπαράσταση/οπτικοποίηση των ιδιοδιανυσμάτων δημιουργείται ένα τρισδιάστατο διάγραμμα, όπου:
 - ο Χρησιμοποιείται η εντολή "**quiver3**" για να σχεδιαστεί ένα προσανατολισμένο διάνυσμα με αρχή το σημείο (0, 0, 0) και τέλος τις συντεταγμένες του καθενός από τα τρία ιδιοδιανύσματα v_1, v_2, v_3 .
 - Κάθε ιδιοδιάνυσμα εμφανίζεται με διαφορετικό χρώμα, επιλεγμένο από την χρωματική παλέτα lines.
 - Κοντά στο τέλος κάθε διανύσματος προστίθεται ετικέτα με το όνομά του (π.χ., v_1, v_2 , ή v_3), διευκολύνοντας την αναγνώρισή του.

Κώδικας 3.6

```
% Οπτικοποίηση Ιδιοδιανυσμάτων
function analyze_eigenvectors(e, v, matrixName, m)
if m ~= 3
    fprintf('3D visualization of eigenvectors is not
possible for m = %d. Required m = 3.\n', m);
    return;
end
numEigenvalues = min(length(e), 10);
figure;
hold on;
colors = lines(numEigenvalues);
for i = 1:numEigenvalues
    vec = v(:, i) * sqrt(abs(e(i)));
```

⁹Βλέπε στο Παράρτημα I, στις γραμμές 68 – 90.


```

    quiver3(0, 0, 0, vec(1), vec(2), vec(3),
    'LineWidth', 2, ...
    'MaxHeadSize', 2, 'Color', colors(i, :));
    text(vec(1), vec(2), vec(3), ['v' num2str(i)],
    'FontSize', 8);
end
hold off;
xlabel('X');
ylabel('Y');
zlabel('Z');
view(3);
title(['3D Visualization of ', matrixName, '
Eigenvectors (m = ', num2str(m), ')']);
grid on;
end

```

3.2.4 Συνάρτηση visualize_all

Ο ακόλουθος Κώδικας 3.7¹⁰ δημιουργήθηκε ως συνάρτηση (function) της Matlab, που ονομάζεται "visualize_all", έχει ως στόχο τη δημιουργία οπτικοποιήσεων που περιγράφουν τη γεωμετρία και τα χαρακτηριστικά μιας πρωτεΐνης, συνδυάζοντας τρεις κύριες απεικονίσεις:

- 3D απεικόνιση των ατομικών συντεταγμένων της πρωτεΐνης
- Θερμικός χάρτης (heatmap) του πίνακα αποστάσεων
- 3D απεικόνιση των ιδιοδιανυσμάτων του τριδιαγώνιου πίνακα T_3

Η υλοποίηση της συνάρτησης δέχεται ως είσοδο τον πίνακα (cord) των συντεταγμένων X,Y,Z των ατόμων της πρωτεΐνης, τον πίνακα αποστάσεων (distanceMatrix) μεταξύ όλων των ατόμων τις ιδιοτιμές και τα αντίστοιχα ιδιοδιανύσματα του τριδιαγώνιου πίνακα, τα οποία έχουν υπολογιστεί μέσω του Κώδικα 3.3 (eT και vT αντίστοιχα), καθώς και το αναγνωριστικό της πρωτεΐνης σε μορφή PDB (pdbID) και τον αριθμό των επαναλήψεων m, που καθορίζει πόσες ιδιοτιμές του τριδιαγώνιου πίνακα υπολογίζονται.

Η έξοδος της συνάρτησης περιλαμβάνει τρία βασικά διαγράμματα:

- **3D απεικόνιση ατομικών συντεταγμένων της πρωτεΐνης**
Χρησιμοποιείται η εντολή "scatter3" για τη σχεδίαση των ατόμων της πρωτεΐνης στον τρισδιάστατο χώρο. Οι συντεταγμένες X,Y,Z καθορίζουν τη θέση κάθε σημείου, ενώ το χρώμα κάθε σημείου βασίζεται στην τιμή του άξονα Z (ύψος), με χρήση της παλέτας jet.

¹⁰Βλέπε στο Παράρτημα I, στις γραμμές 91 – 111.

Η συγκεκριμένη απεικόνιση παρουσιάζει τη χωρική κατανομή των ατόμων και αναδεικνύει τα γεωμετρικά χαρακτηριστικά της δομής της πρωτεΐνης, γι' αυτό και το διάγραμμα είναι ανεξάρτητο από τον αριθμό των επαναλήψεων m .

- **Θερμικός χάρτης (heatmap) του πίνακα αποστάσεων**

Το διάγραμμα αυτό απεικονίζει τις αποστάσεις μεταξύ όλων των ατόμων της πρωτεΐνης, επιτρέποντας τον εντοπισμό περιοχών με μικρές ή μεγάλες αποστάσεις. Για τη δημιουργία του χρησιμοποιείται η ενσωματωμένη συνάρτηση της Matlab "**imagesc**", όπου κάθε τιμή στον πίνακα αντιπροσωπεύεται από ένα χρώμα:

- ο Σκούρα χρώματα αντιστοιχούν σε μικρές αποστάσεις
- ο Ανοιχτά χρώματα σε μεγαλύτερες αποστάσεις.

Επειδή το διάγραμμα σχετίζεται με τον πίνακα αποστάσεων και όχι με τον τριδιαγώνιο πίνακα γι' αυτό και το διάγραμμα είναι ανεξάρτητο από τον αριθμό των επαναλήψεων m .

- **3D απεικόνιση των ιδιοδιανυσμάτων του τριδιαγώνιου πίνακα T_3**

Αυτή η απεικόνιση αναδεικνύει τις κύριες κατευθύνσεις του πίνακα αποστάσεων, δηλαδή τις γεωμετρικά σημαντικές συνιστώσες της δομής. Για την αναπαράσταση των ιδιοδιανυσμάτων του 3×3 τριδιαγώνιου πίνακα T_3 υλοποιείται ο Κώδικας 3.6.

Κώδικας 3.7

```
% ΟΠΤΙΚΟΠΟΙΗΣΕΙΣ
function visualize_all(cord, distanceMatrix, eT, vT,
pdbID, m)
% 3D ΟΠΤΙΚΟΠΟΙΗΣΗ Ατομικών Συντεταγμένων
figure;
scatter3(cord(:, 2), cord(:, 1), cord(:, 3), 20,
cord(:, 3), 'filled');
colormap(jet);
colorbar;
xlabel('X-axis'); % Ετικέτα για τον X-άξονα
ylabel('Y-axis'); % Ετικέτα για τον Y-άξονα
zlabel('Z-axis');
title(['3D Visualization of Protein Atomic Coordinates
for ', pdbID]);
grid on;
axis equal;
view(30, 30);
% Heatmap του Πίνακα Αποστάσεων
figure;
imagesc(distanceMatrix);
colorbar;
xlabel('Atom Index');
```

```

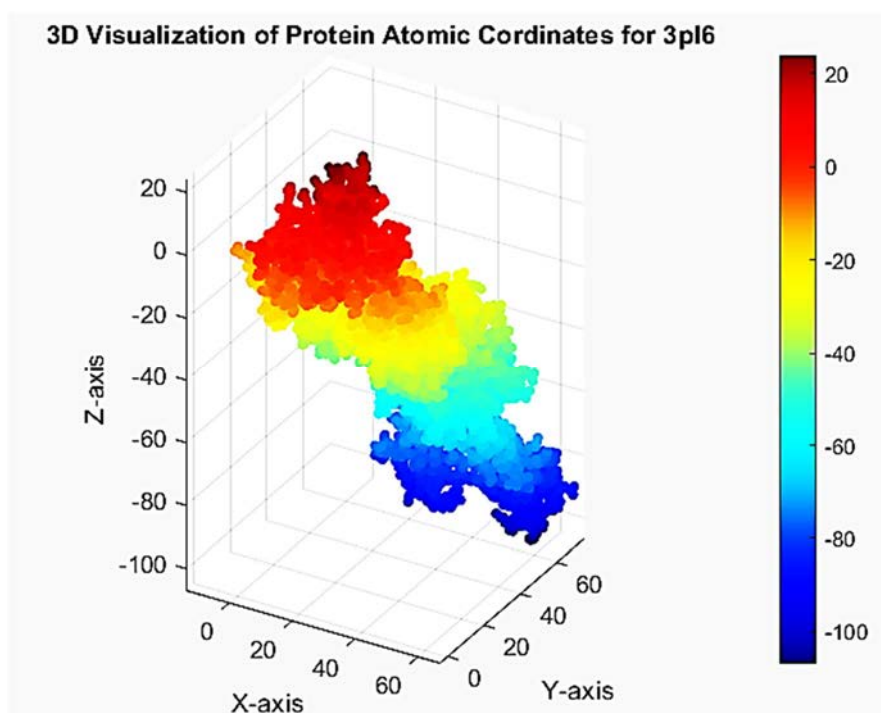
ylabel('Atom Index');
title(['Heatmap of Distance Matrix for ', pdbID]);
axis square;
% Οπτικοποίηση Ιδιοδιανυσμάτων
analyze_eigenvectors(eT, vT, ['Distance Matrix of ',
pdbID], m);
end

```

Παράδειγμα 3.8

Θεωρούμε τα δεδομένα της πρωτεΐνης 3pl6. Υλοποιώντας τον Κώδικα 3.7 που αποτελεί τις γραμμές 91 – 111 του κύριου κώδικα του Παραρτήματος Ι, σχεδιάζονται :

- Ο τα άτομα της πρωτεΐνης μέσω των συντεταγμένων τους στον τρισδιάστατο χώρο, τα οποία απεικονίζεται στο Σχήμα 3.1. Τα σημεία χρωματίζονται με βάση τις τιμές του Z-άξονα, με τη χρωματική κλίμακα να κυμαίνεται από μπλε (χαμηλές τιμές του Z) σε κόκκινο (υψηλές τιμές του Z).
- Ο οι αποστάσεις μεταξύ όλων των ατόμων της πρωτεΐνης αποτυπώνονται με χρώματα εμφανίζοντας τον θερμικό χάρτη, ο οποίος απεικονίζεται στο Σχήμα 3.2. Οι άξονες X και Y του θερμικού χάρτη αντιστοιχούν στους δείκτες των ατόμων της πρωτεΐνης και κάθε σημείο (X,Y) του θερμικού χάρτη αντιστοιχεί στην απόσταση των αντίστοιχων ατόμων X, Y.
- Ο στο Σχήμα 3.3 με χρωματικά βέλη τα ιδιοδιανύσματα v_1 , v_2 , και v_3 , του πίνακα T_3 στην (3.5), τα οποία υπολογίστηκαν στο Παράδειγμα 3.4 υλοποιώντας τον Κώδικα 3.3 και τα οποία αντιπροσωπεύουν τις τρεις κύριες κατευθύνσεις του τρισδιάστατου χώρου όπου κατανέμεται η πρωτεΐνη, σύμφωνα με το δεύτερο και τρίτο σχόλιο στα Σχόλια 3.5.

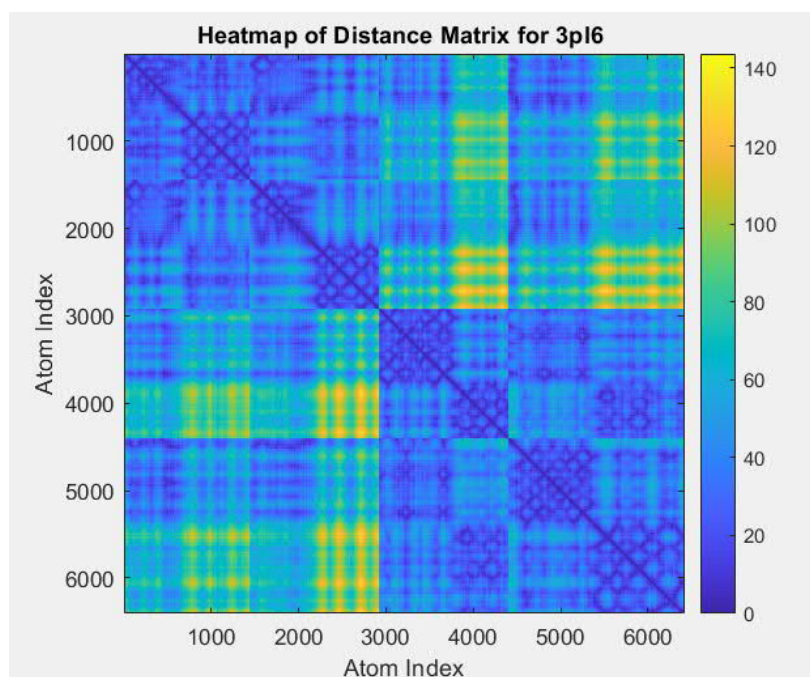


Σχήμα 3.1: 3D απεικόνιση των ατομικών συντεταγμένων της πρωτεΐνης 3pl6 με χρωματική κλίμακα ως προς τον άξονα Z.

Ερμηνεία Σχήματος 3.1:

- Η πρωτεΐνη έχει μια συμπαγής γεωμετρική δομή που εκτείνεται σε όλους τους άξονες.
- Η χρωματική κλίμακα δείχνει ότι τα περισσότερα άτομα βρίσκονται κοντά στο κέντρο, αλλά υπάρχει κάποια διακύμανση στο ύψος (Z).

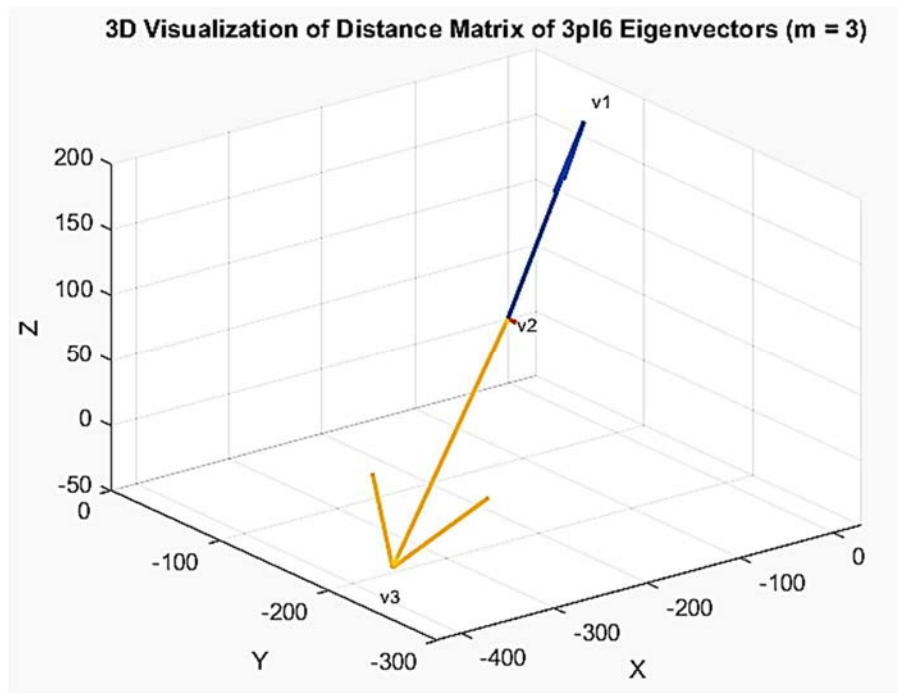
Συνεπώς, η δομή της πρωτεΐνης είναι καλά ορισμένη σε τρεις διαστάσεις, με ελαφρώς μεγαλύτερη έκταση στον Z-άξονα. Επιπλέον, η κατανομή των ατόμων είναι σχετικά ομοιογενής, χωρίς μεγάλες ανωμαλίες.



Σχήμα 3.2: Θερμικός χάρτης του πίνακα αποστάσεων της πρωτεΐνης 3p16.

Ερμηνεία Σχήματος 3.2:

- Οι πιο σκούρες περιοχές (μπλε) αντιπροσωπεύουν ομάδες ατόμων που βρίσκονται κοντά το ένα στο άλλο.
- Οι πιο φωτεινές περιοχές (κίτρινο/κόκκινο) δείχνουν άτομα που βρίσκονται πιο μακριά. Συνεπώς, από τον θερμικό χάρτη φαίνεται ότι το μεγαλύτερο μέρος των στοιχείων του πίνακα είναι μπλε ή πράσινα. Αυτό επιβεβαιώνει ότι οι περισσότερες ομάδες ατόμων βρίσκονται κοντά η μία στην άλλη. Οι κόκκινες περιοχές, αν και υπάρχουν, είναι περιορισμένες σε σχέση με το σύνολο. Αυτό σημαίνει ότι οι μεγάλες αποστάσεις είναι λιγότερες, αλλά σημαντικές. Επίσης οι περιοχές κοντά στη διαγώνιο είναι έντονο μπλε, που δείχνουν ότι τα περισσότερα άτομα έχουν γειτονικά άτομα σε μικρές αποστάσεις. Συνεπώς, ο θερμικός χάρτης του πίνακα αποστάσεων υποδεικνύει ότι η δομή της πρωτεΐνης περιλαμβάνει και μικρές τοπικές αποστάσεις, αλλά ελάχιστες μεγάλες αποστάσεις μεταξύ απομακρυσμένων περιοχών.



Σχήμα 3.3: 3D απεικόνιση των ιδιοδιανυσμάτων του πίνακα T_3 της πρωτεΐνης 3p16.

Ερμηνεία Σχήματος 3.3:

- Τα ιδιοδιανύσματα υποδεικνύουν τις κύριες κατευθύνσεις στις οποίες κατανέμεται η γεωμετρία της πρωτεΐνης.
- Το μέγεθος των βελών δείχνει προς τη σχετική σημασία κάθε κατεύθυνσης (οι μεγαλύτερες αντιστοιχούν στις κυριότερες συνιστώσες).

Από την 3D απεικόνιση των ιδιοδιανυσμάτων του πίνακα T_3 συμπεραίνουμε:

- Το v_3 συνδέεται με τη φασματική ακτίνα λ_3 , η οποία αντιπροσωπεύει τη σημαντικότερη κατεύθυνση κατανομής των δεδομένων. Γνωρίζουμε, ότι το μέγεθος των βελών είναι ανάλογο με τις αντίστοιχες ιδιοτιμές. Εδώ παρατηρείται ότι το μήκος του v_3 , είναι μεγαλύτερο από αυτό των v_1 και v_2 , υποδεικνύοντας τη σημασία του ως κύριο ιδιοδιάνυσμα. Η φασματική ακτίνα, που σχετίζεται με το v_3 , δείχνει ότι η πλειονότητα των ατόμων της πρωτεΐνης κινούνται σε αυτή την κατεύθυνση. Τα v_1 και v_2 , που συνδέονται με μικρότερες ιδιοτιμές, περιγράφουν τοπικές και δευτερεύουσες κινήσεις της πρωτεΐνης.
- Ο αλγόριθμος Lanczos είναι εξαιρετικά αποτελεσματικός, αλλά ευαίσθητος σε αριθμητικά σφάλματα, ειδικά όταν εφαρμόζεται σε μεγάλα προβλήματα όπως αυτό. Βελτιώσεις, όπως η χρήση επανεκκινήσεων, μπορούν να μειώσουν περαιτέρω τα αριθμητικά σφάλματα και να δώσουν ακόμα πιο ακριβή αποτελέσματα.

Σχόλια 3.9

- i. **Δομή Πρωτεΐνης:** Η πρωτεΐνη 3p16 είναι συμπαγής, με καλά καθορισμένες τρισδιάστατες κατανομές με ελαφρώς μεγαλύτερη έκταση στον Z-άξονα.
- ii. **Σχέσεις μεταξύ ατόμων:** Υπάρχουν μικρές τοπικές αποστάσεις, που υποδεικνύουν περιοχές της πρωτεΐνης που είναι στενά συνδεδεμένες και ελάχιστες μεγάλες αποστάσεις, που υποδεικνύουν απομακρυσμένες περιοχές, πιθανώς μεταξύ διαφορετικών περιοχών.
- iii. **Ανάλυση ιδιοτιμών/ιδιοδιανυσμάτων:** Η φασματική ακτίνα αντιπροσωπεύει τη σημαντικότερη συνιστώσα, η οποία σχετίζεται με τον κύριο άξονα κατανομής των δεδομένων. Το τρίτο ιδιοδιάνυσμα, το οποίο αντιστοιχεί στη φασματική ακτίνα δείχνει την κύρια κατεύθυνση κατανομής της πρωτεΐνης στον τρισδιάστατο χώρο. Το πρώτο και δεύτερο ιδιοδιάνυσμα αντιστοιχούν σε μικρότερες ιδιοτιμές από αυτές της φασματικής ακτίνας, επαληθεύοντας με αυτόν τον τρόπο τα Σχόλια 3.5 (ii) – (iii).
- iv. **Χρησιμότητα:** Τα ιδιοδιανύσματα επιτρέπουν τη μείωση της διάστασης, διατηρώντας τις βασικές πληροφορίες για τη γεωμετρία της πρωτεΐνης. Αυτό είναι ιδιαίτερα χρήσιμο για την ανάλυση μεγάλων πρωτεϊνικών συστημάτων.
- v. **Γεωμετρική Ανάλυση:** Μέσω του πίνακα αποστάσεων και τη μελέτη του τριδιαγώνιου πίνακα, αντλούνται πληροφορίες που αφορούν τη γεωμετρική και τοπολογική δομή της πρωτεΐνης, οι οποίες σχετίζονται με τη σταθερότητα, την ευελιξία και τις δυναμικές της ιδιότητες.

3.2.3 Συνάρτηση `compare_eigenvalues`

Ο ακόλουθος Κώδικας 3.10¹¹ δημιουργήθηκε ως συνάρτηση (function) της Matlab, που ονομάζεται " `compare_eigenvalues` ", η οποία δημιουργεί ένα διάγραμμα σύγκρισης των ιδιοτιμών του τριδιαγώνιου πίνακα, που υπολογίζονται, για δύο διαφορετικές τιμές της παραμέτρου m , συγκεκριμένα για τις τιμές m_1 και m_2 . Σκοπός της είναι να απεικονίσει τις ιδιοτιμές για τις δύο περιπτώσεις και να αναδείξει τις διαφορές μεταξύ τους, συμβάλλοντας στην κατανόηση του τρόπου με τον οποίο επηρεάζει η παράμετρος m την προσέγγιση των ιδιοτιμών.

Η συνάρτηση δέχεται ως είσοδο τις ιδιοτιμές που υπολογίστηκαν για την πρώτη τιμή m_1 (μεταβλητή `eT1`) και τις ιδιοτιμές για τη δεύτερη τιμή m_2 (μεταβλητή `eT2`). Ως έξοδο, παράγεται ένα διάγραμμα σύγκρισης, το οποίο περιλαμβάνει, δύο γραμμές (μία για κάθε τιμή των m_1, m_2) που απεικονίζουν τις ιδιοτιμές, τις ετικέτες που υποδεικνύουν ποια γραμμή αντιστοιχεί στην τιμή m_1 και ποια στην m_2 μία γραμμή μηδενικής βάσης, η οποία διαχωρίζει τις θετικές από τις αρνητικές ιδιοτιμές και διευκολύνει την οπτική σύγκριση.

¹¹Βλέπε στο Παράρτημα I, στις γραμμές 112 – 124.

Αρχικά, ο χρήστης εισάγει την πρώτη τιμή m_1 , δηλαδή τον αριθμό ιδιοτιμών και ιδιοδιανυσμάτων που θα υπολογιστούν μέσω του Κλασσικού Αλγορίθμου Lanczos, (βλέπε, Αλγόριθμος 2.1 και Κώδικας 2.4). Η μέθοδος αυτή είναι ιδιαίτερα αποδοτική για την εύρεση ιδιοτιμών και ιδιοδιανυσμάτων συμμετρικών πινάκων μεγάλων διαστάσεων. Στη συνέχεια, δίνεται η δυνατότητα στον χρήστη να εισάγει μια δεύτερη τιμή m_2 , απαντώντας "yes" στην αντίστοιχη ερώτηση (βλέπε, Παράρτημα I, γραμμή 34), ώστε να επαναληφθεί η διαδικασία υπολογισμού για διαφορετικό πλήθος ιδιοτιμών και ιδιοδιανυσμάτων. Εάν ο χρήστης επιλέξει να μην εκτελεστεί δεύτερη επανάληψη, ο κώδικας προχωρά με την ανάλυση μόνο για την τιμή m_1 . Σκοπός του βήματος αυτού είναι η εξαγωγή των ιδιοτιμών και των αντίστοιχων ιδιοδιανυσμάτων που παράγονται ανάλογα με την τιμή του m , τα οποία μπορούν να αξιοποιηθούν για σκοπούς οπτικοποίησης και περαιτέρω ανάλυσης. Επίσης, παρέχεται η δυνατότητα σύγκρισης των αποτελεσμάτων για τις δύο τιμές, m_1 και m_2 , με θετική απάντηση στην ερώτηση (βλέπε Παράρτημα I, γραμμή 58).

Κώδικας 3.10

```
% Σύγκριση ιδιοτιμών
function compare_eigenvalues(eT1, eT2, m1, m2)
figure;
plot(1:length(eT1), eT1, 'o', 'MarkerSize', 8,
'DisplayName', ['m1 = ', num2str(m1)]);
hold on;
plot(1:length(eT2), eT2, '*', 'MarkerSize', 8,
'DisplayName', ['m2 = ', num2str(m2)]);
title('Comparison of Eigenvalues between m1 and m2',
'FontSize', 14);
xlabel('Eigenvalue Number', 'FontSize', 12);
ylabel('Eigenvalue Price', 'FontSize', 12);
yline(0, '--', 'Zero Base', 'LineWidth', 1, 'Color',
'k');
grid on;
legend('show', 'FontSize', 12, 'Location', 'Best');
hold off;
end
```

Παράδειγμα 3.11

Θεωρούμε τα δεδομένα της πρωτεΐνης 3p16 όπως στα Παραδείγματα 3.4 και 3.8. Ακολουθώντας τη μεθοδολογία του Παραδείγματος 3.4 και υλοποιώντας τον Κώδικα 3.3 για τον 6405×6405 πίνακα αποστάσεων A και για $m = 5$ υπολογίζονται τα στοιχεία του τριδιαγώνιου και συμμετρικού πίνακα T_5

$$T_5 = 10^5 \begin{pmatrix} 2.2463 & 1.3842 & 0 & 0 & 0 \\ 1.3842 & 0.8021 & 0.1288 & 0 & 0 \\ 0 & 0.1288 & -0.3304 & 0.0367 & 0 \\ 0 & 0 & 0.0367 & -0.9436 & 0.5114 \\ 0 & 0 & 0 & 0.5114 & -0.5056 \end{pmatrix} \quad (3.7)$$

καθώς και τα ιδιοποσά του.

➤ Οι διατεταγμένες σε αύξουσα σειρά **ιδιοτιμές** του T_5 στην (3.7) είναι:

$eT =$

1.0e+05 *
-1.2819
-0.3692
-0.1667
-0.0001
3.0867

➤ Τα αντίστοιχα **ιδιοδιανύσματα** του T_5 εμφανίζονται στη συνέχεια ως στήλες στο vT :

$vT =$

0.0011	-0.1467	0.0330	0.4970	-0.8546
-0.0027	0.2772	-0.0575	-0.8066	-0.5189
0.0326	-0.9443	0.0780	-0.3176	-0.0196
-0.8346	0.0258	0.5495	-0.0274	-0.0002
0.5498	0.0967	0.8292	-0.0277	-0.0000

Εδώ χρειάζεται να υπενθυμίσουμε ότι τα διαγράμματα, που προκύπτουν στην έξοδο του Κώδικα 3.7 ως 3D απεικόνιση των ατομικών συντεταγμένων της πρωτεΐνης 3p16 και ως θερμικός χάρτης του πίνακα αποστάσεων της πρωτεΐνης 3p16 (βλέπε, Σχήμα 3.1, Σχήμα 3.2, αντίστοιχα), είναι ανεξάρτητα από τις τιμές του m . Συνεπώς τα δύο προαναφερόμενα διαγράμματα παραλείπονται ωστόσο ο σχολιασμός τους είναι ακριβώς ο ίδιος με αυτόν που αναφέρεται στο Παράδειγμα 3.8 και λαμβάνεται υπόψη κατά την ερμηνεία των ιδιοποσών του παρόντος παραδείγματος στη συνέχεια.

Σχόλια 3.12

Ερμηνεία ιδιοποσών και γραφημάτων Παραδείγματος 3.11

- i. Σημειώνοντας με $\tilde{\lambda}_1 = -1.2819 \cdot 10^5$, $\tilde{\lambda}_2 = -0.3692 \cdot 10^5$, $\tilde{\lambda}_3 = -0.1667 \cdot 10^5$, $\tilde{\lambda}_4 = -0.0001 \cdot 10^5$, και $\tilde{\lambda}_5 = 3.0867 \cdot 10^5$ τις ιδιοτιμές του T_5 , αυτές προφανώς παρουσιάζουν την ακόλουθη διάταξη $\tilde{\lambda}_1 < \tilde{\lambda}_2 < \tilde{\lambda}_3 < \tilde{\lambda}_4 < \tilde{\lambda}_5$. Η σημαντική διαφορά μεταξύ της φασματικής ακτίνας, $\tilde{\lambda}_5$, και των υπολοίπων ιδιοτιμών δείχνει ότι η πρωτεΐνη μπορεί να περιγραφεί κατά κύριο λόγο από μία κύρια συνιστώσα.

- ii. Η πέμπτη στήλη του $\mathbf{VT}$, $\tilde{\mathbf{V}}_5$, είναι το ιδιοδιάνυσμα, που αντιστοιχεί στη φασματική ακτίνα του $\mathbf{T}_5$, $\tilde{\lambda}_5$, συνεπώς το $\tilde{\mathbf{V}}_5$ είναι η κύρια συνιστώσα, που δείχνει την κύρια κατεύθυνση της πρωτεΐνης στον τρισδιάστατο χώρο. Επιπρόσθετα από την κύρια συνιστώσα ερμηνεύεται ταυτόχρονα και η μεγαλύτερη διακύμανση. Η μεγαλύτερη διακύμανση υποδεικνύει την κύρια κατεύθυνση κίνησης ή κατανομής της πρωτεΐνης στο χώρο. Αυτή η συνιστώσα έχει τη μεγαλύτερη συνεισφορά στην περιγραφή της γεωμετρίας της πρωτεΐνης και ερμηνεύει τη μεγαλύτερη κατανομή των δεδομένων ή τη συνολική δυναμική της πρωτεΐνης. Αυτό ακριβώς επαληθεύεται και από τα σχόλια στο 5^ο βήμα της Ενότητας 3.1, από όπου συμπεραίνεται ότι στην ιδιοτιμή $\tilde{\lambda}_5$ αντιστοιχεί η κύρια και σημαντικότερη συνιστώσα.
- iii. Τα ιδιοδιανύσματα $\tilde{\mathbf{V}}_1, \tilde{\mathbf{V}}_2, \tilde{\mathbf{V}}_3, \tilde{\mathbf{V}}_4$ (αντίστοιχα η πρώτη, δεύτερη κ.ο.κ. στήλη του $\mathbf{VT}$) αντιστοιχούν στις διατεταγμένες κατά αύξουσα σειρά ιδιοτιμές $\tilde{\lambda}_1, \tilde{\lambda}_2, \tilde{\lambda}_3, \tilde{\lambda}_4$ του $\mathbf{T}_5$. Οι ιδιοτιμές αυτές περιγράφουν δευτερεύουσες κατευθύνσεις που έχουν πολύ μικρότερη σημασία, καθώς σχετίζονται με τοπικές παραμορφώσεις ή λεπτομέρειες της δομής. Συγκεκριμένα, τα ιδιοδιανύσματα $\tilde{\mathbf{V}}_1$ και $\tilde{\mathbf{V}}_2$ έχουν μικρή συμβολή στη συνολική γεωμετρία, ενώ τα $\tilde{\mathbf{V}}_3$ και $\tilde{\mathbf{V}}_4$ περιγράφουν δευτερεύουσες δομικές παραμορφώσεις, παρέχοντας λεπτομέρειες για μικρές διαφοροποιήσεις στη γεωμετρία της πρωτεΐνης.
- iv. Σύμφωνα με την ερμηνεία του Σχήματος 3.1 στο Παράδειγμα 3.8, η δομή της πρωτεΐνης ερμηνεύεται από τη 3D απεικόνιση των ατομικών συντεταγμένων, η οποία αναπαριστά μια καλά ορισμένη δομή με ορισμένες τοπικές διαφοροποιήσεις στον άξονα-Z.
- v. Αλληλεπιδράσεις μεταξύ ατόμων της πρωτεΐνης: Στο Σχήμα 3.2, ο θερμικός χάρτης αναδεικνύει/υποδεικνύει ότι η πρωτεΐνη έχει τόσο τοπικές όσο και απομακρυσμένες αλληλεπιδράσεις. Οι τοπικές αλληλεπιδράσεις είναι περισσότερες από τις απομακρυσμένες, κάτι που είναι σύνηθες σε πρωτεΐνες με συμπαγή και καλά διπλωμένη δομή.
- vi. Χρησιμοποιώντας την εντολή `eigs`¹² της Matlab υπολογίζονται πέντε από τις 6405 ιδιοτιμές του πίνακα αποστάσεων $\mathbf{A}$, οι οποίες αναφέρονται στη συνέχεια:
 $\tilde{\ell}_1 = -1.2833 \cdot 10^5$, $\tilde{\ell}_2 = -0.3695 \cdot 10^5$, $\tilde{\ell}_3 = -0.2554 \cdot 10^5$, $\tilde{\ell}_4 = -0.1690 \cdot 10^5$,
 $\tilde{\ell}_5 = 3.0867 \cdot 10^5$, οι οποίες έχουν την ακόλουθη διάταξη

$$\tilde{\ell}_1 < \tilde{\ell}_2 < \tilde{\ell}_3 < \tilde{\ell}_4 < \tilde{\ell}_5 \quad (3.8)$$
 Παρατηρούμε ότι οι ιδιοτιμές $\tilde{\ell}_i$, $i = 1, 2, 3, 4$ ανήκουν στο διάστημα $(-\tilde{\ell}_5, \tilde{\ell}_5)$ επαληθεύοντας την (3.4).
 Επιπλέον, συγκρίνοντας τις ιδιοτιμές του πίνακα αποστάσεων $\mathbf{A}$ που εμφανίζονται στην (3.6) του Παραδείγματος 3.4 (βλέπε, Σχόλια 3.5 (iv)) και (3.8), παρατηρούμε ότι η φασματική ακτίνα του πίνακα $\mathbf{A}$ και οι δύο μικρότερες ιδιοτιμές ταυτίζονται, είτε μελετάται η περίπτωση $m = 3$, είτε $m = 5$, και στην (3.8) εμφανίζονται δύο νέες τιμές, που είναι: $\tilde{\ell}_3, \tilde{\ell}_4$
- vii. Συγκρίνοντας τις ιδιοτιμές των πινάκων $\mathbf{T}_5$ και $\mathbf{A}$, που υπολογίστηκαν στο (i) και (vi), υπολογίζεται το σφάλμα της προσέγγισής τους, που είναι:

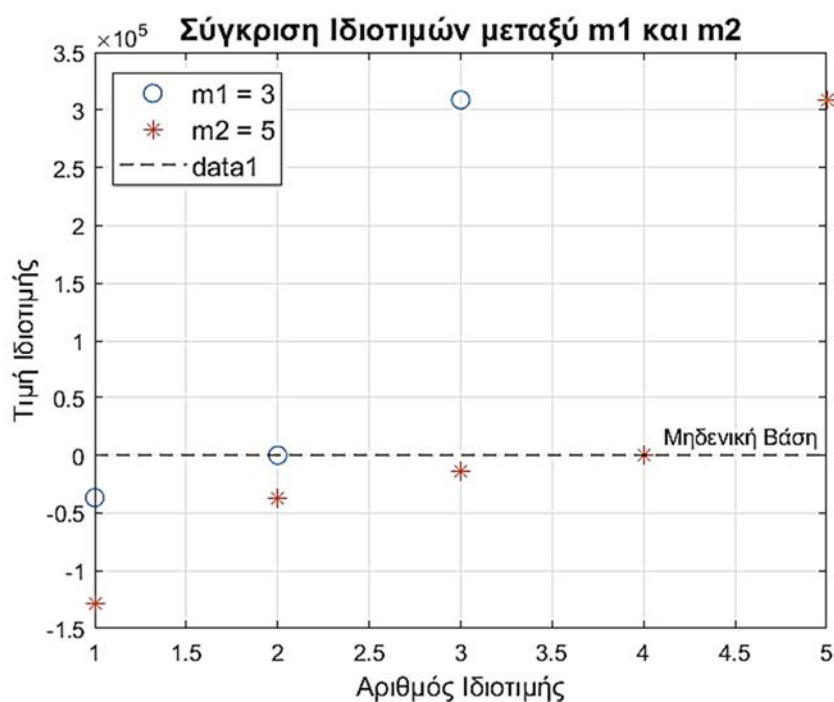
¹² Βλέπε, στην Ενότητα 4.2 της παρούσας διπλωματικής εργασίας για την εντολή στην Matlab.

$|\tilde{\ell}_1 - \tilde{\lambda}_1| = 140, |\tilde{\ell}_2 - \tilde{\lambda}_2| = 30, |\tilde{\ell}_3 - \tilde{\lambda}_3| = 8870, |\tilde{\ell}_4 - \tilde{\lambda}_4| = 16890, |\tilde{\ell}_5 - \tilde{\lambda}_5| = 0$
 Παρατηρούμε ότι στην περίπτωση της φασματικής ακτίνας των πινάκων δεν υπάρχει σφάλμα στην προσέγγιση, ενώ στις υπόλοιπες περιπτώσεις το σφάλμα είναι διάφορο του μηδενός και μάλιστα η τέταρτη διαφορά έχει τάξη μεγέθους όση και οι ιδιοτιμές, δηλαδή 10^5 . Αυτή η παρατήρηση γενικεύεται στο Σχόλιο 4.10 της Ενότητας 4.2 της παρούσης διπλωματικής εργασίας.

Παράδειγμα 3.13

Θεωρούμε τα δεδομένα της πρωτεΐνης 3p16. Υλοποιώντας τον Κώδικα 3.10, που αποτελεί τις γραμμές 112 – 124 του κύριου κώδικα του Παραρτήματος Ι, απεικονίζονται στο Σχήμα 3.4 οι ιδιοτιμές των τριδιαγώνιων πινάκων T_3, T_5 , που υπολογίστηκαν στα Παραδείγματα 3.4 και 3.11, αντίστοιχα.

Στον οριζόντιο άξονα του Σχήματος 3.4 απεικονίζεται ο δείκτης της ιδιοτιμής και στον κατακόρυφο η αριθμητική τιμή της (πολλαπλασιασμένη επί 10^5). Η διακεκομμένη γραμμή ονομαζόμενη Μηδενική Βάση είναι η ευθεία που χωρίζει το πραγματικό επίπεδο σε δύο ημιεπίπεδα, προκειμένου να οριοθετείται το θετικό από το αρνητικό επίπεδο, συνεπώς να γίνεται άμεσα αντιληπτό το πρόσημο των ιδιοτιμών, θετικές στο πάνω ημιεπίπεδο και αρνητικές στο κάτω. Οι ιδιοτιμές του T_3 σημειώνονται μπλε "ο" και η φασματική ακτίνα $\lambda_3 = 3.0867 \cdot 10^5$ επίσης με μπλε "ο". Οι ιδιοτιμές του T_5 σημειώνονται πορτοκαλί "*" και η φασματική ακτίνα $\tilde{\lambda}_5 = 3.0867 \cdot 10^5$ με το ίδιο σύμβολο.



Σχήμα 3.4: Αναπαράσταση των ιδιοτιμών των πινάκων T_3 και T_5

Σχόλια 3.14

Επειδή στις δύο περιπτώσεις ($m_1 = 3$ και $m_2 = 5$), οι φασματικές ακτίνες των πινάκων ταυτίζονται, $\lambda_3 = \tilde{\lambda}_5 = 3.0867 \cdot 10^5$, αποδεικνύεται ότι το αντίστοιχο ιδιοδιάνυσμα σε καθένα από τους τριδιαγώνιους πίνακες είναι η κυρίαρχη συνιστώσα. Αυτή η συνιστώσα έχει τη μεγαλύτερη συνεισφορά στην περιγραφή της γεωμετρίας της πρωτεΐνης και ερμηνεύει τη μεγαλύτερη κατανομή των δεδομένων ή τη συνολική δυναμική της πρωτεΐνης, βλέπε στο τρίτο σχόλιο στα ερμηνευτικά Σχόλια 3.5 του Παραδείγματος 3.4 και στο δεύτερο σχόλιο στα Σχόλια 3.12 του Παραδείγματος 3.11.

Συνεπώς, η υλοποίηση του Κώδικα 3.4 με $m_1 = 3$ είναι κατάλληλη για γρήγορη επισκόπηση των κύριων κατευθύνσεων ενώ η υλοποίηση με $m_2 = 5$

Εκτός από τη φασματική ακτίνα στις δύο περιπτώσεις υπολογίζονται επιπλέον ιδιοτιμές με τα αντίστοιχα ιδιοδιανύσματα τους, τα οποία προσφέρουν περισσότερες πληροφορίες για δευτερεύουσες παραμορφώσεις ή κινήσεις της πρωτεΐνης, ωστόσο η βασική κυριαρχία της συνιστώσας που αντιστοιχεί στη φασματική ακτίνα δεν αλλάζει. Συνεπώς, η κύρια συνιστώσα, που αντιστοιχεί στη φασματική ακτίνα, εξηγεί την πλειονότητα της γεωμετρικής δομής της πρωτεΐνης ανεξάρτητα από την επιλογή της τιμής m .

Ωστόσο, επειδή ο πίνακας αποστάσεων της πρωτεΐνης 3p16 είναι 6405×6405 , χρησιμοποιώντας τη σχέση στη (2.12) για την κάθε περίπτωση των $m_1 = 3$ και $m_2 = 5$, υπολογίζεται διαφορετικός υπολογιστικός φόρτος κατά την υλοποίηση του Κλασσικού Αλγορίθμου Lanczos. Η υπολογιστική πολυπλοκότητα είναι διαφορετική στις δύο περιπτώσεις μια και κάθε αλγόριθμος εκτελεί διαφορετικό πλήθος επαναλήψεων, (βλέπε, τον γενικό τύπο στον Πίνακα 3.5).

Συνεπώς, η υλοποίηση του Κώδικα 3.4 με $m_1 = 3$ είναι κατάλληλη για γρήγορη επισκόπηση των κύριων κατευθύνσεων ενώ η υλοποίηση με $m_2 = 5$ προσφέρει πληροφορίες για δευτερεύουσες κατευθύνσεις χωρίς να μεταβάλλεται η κύρια συνιστώσα. Τα συμπεράσματα της σύγκρισης των αποτελεσμάτων των δύο παραδειγμάτων παρουσιάζονται συγκεντρωτικά στον ακόλουθο Πίνακα 3.5.

Πίνακας 3.5: Σύγκριση αποτελεσμάτων των Παραδειγμάτων 3.4 και 3.11.

Χαρακτηριστικό	$m = 3$	$m = 5$
Πλήθος Ιδιοτιμών	3	5
Κύρια συνιστώσα- κύρια κατεύθυνση	Συνδέεται με το v_3 που αντιστοιχεί στη λ_3	Συνδέεται με το $\tilde{v}_5$ που αντιστοιχεί στη λ_5
Δευτερεύουσες Κατευθύνσεις	v_1, v_2	$\tilde{v}_1, \tilde{v}_2, \tilde{v}_3, \tilde{v}_4$
Υπολογιστική πολυπλοκότητα	$3n^2 + 15n$	$5n^2 + 25n$

Αξιοποιώντας τα πλεονεκτήματα της Μεθόδου των Δυνάμεων (βλέπε, Κώδικα 1.8) κατά την προσέγγιση των ιδιοτιμών ενός πίνακα, που είναι η ταχύτητα και η ακρίβεια, με τα πλεονεκτήματα της υλοποίησης του Κλασσικού Αλγορίθμου Lanczos (βλέπε, Κώδικα 3.3), που είναι η αριθμητική σταθερότητα, σκεφτήκαμε να συνδυάσουμε τους δύο αλγορίθμους προτείνοντας τον Κώδικα 3.15 προκειμένου να υπολογιστεί η βέλτιστη προσέγγιση των ιδιοτιμών συμμετρικών πινάκων μεγάλης διάστασης το δυνατόν ταχύτερα.

Συγκεκριμένα, χρησιμοποιούμε τη συνάρτηση της Μεθόδου των Δυνάμεων για να υπολογίσουμε τις ιδιοτιμές του πίνακα αποστάσεων. Ωστόσο, στους πίνακες μεγάλων διαστάσεων που χρησιμοποιούνται στα παραδείγματά μας, η Μέθοδος των Δυνάμεων δεν μπορεί να ανταποκριθεί επαρκώς λόγω του υπολογιστικού κόστους. Για τον λόγο αυτό, εφαρμόζουμε τη μέθοδο σε έναν μικρότερο πίνακα, ο οποίος είναι ο τριδιαγώνιος πίνακας που προκύπτει από τον Κλασσικό Αλγόριθμο Lanczos.

Κώδικας 3.15

```
%Proposed Algorithm
function [l,v]=Power(a,tol)
irep=0;
z=a*(ones(length(a(:,1)),1));
x=1;
xnew=x+10*tol;
while abs(x-xnew)>tol && irep<2000
    x=xnew;irep=irep+1;
    z=a*z/x;
    [xnew,xx]=max(abs(z));
    zprev=z(xx);
end

z=a*z;
l=z(xx)/zprev;
v=z/norm(z);
end

function d = Powers(a, tol)
n = length(a(:,1));
d = zeros(n, 1);
u = zeros(n, n);

for i = 1:n
    [l, v] = Power(a, tol);
```

```
    a = a - l * (v * v');  
    d(i) = l;  
    u(:,i) = v;  
end  
d = sort(d);  
end
```

Κεφάλαιο 4^ο

Σύγκριση τεσσάρων αλγορίθμων

Στόχος του παρόντος κεφαλαίου είναι η σύγκριση των αλγορίθμων μέσα από την παρουσίαση των αποτελεσμάτων που προκύπτουν μετά την υλοποίηση του Κλασσικού Αλγορίθμου Lanczos (βλέπε, Κώδικα 3.3-Classic Algorithm Lanczos) και των βελτιωμένων εκδόσεών του (βλέπε, Κώδικα 2.8-Modified Lanczos Algorithm και Κώδικα 2.10-Reorthogonalization Lanczos Algorithm) καθώς και τον προτεινόμενο Αλγόριθμο (βλέπε, Κώδικα 3.15-Proposed Algorithm) θεωρώντας στην είσοδό τους τον πίνακα αποστάσεων μεγάλων διαστάσεων (δεδομένα της πρωτεΐνης 3p16 και ιού Ζίκα-ZIKV-5IZ7, [33], [34]) προκειμένου να υπάρξει η βέλτιστη προσέγγιση των ιδιοτιμών του. Η σύγκριση σχετίζεται με τον χρόνο εκτέλεσης των αλγορίθμων και με τα σφάλματα προσέγγισης ορισμένων στο πλήθος ιδιοτιμών του αρχικού πίνακα αποστάσεων από τις ιδιοτιμές του τριδιαγώνιου πίνακα. Τα αποτελέσματα τη σύγκρισης παρουσιάζονται στις επόμενες δύο ενότητες.

4.1 Χρόνος εκτέλεσης αλγορίθμων

Ο χρόνος, σε δευτερόλεπτα, που απαιτείται για την εκτέλεση κάθε αλγορίθμου¹³, υπολογίζεται με χρήση της ενσωματωμένης συνάρτησης "timeit" της Matlab. Η συνάρτηση αυτή δημιουργεί μια ανώνυμη συνάρτηση, που περιλαμβάνει την κλήση του κώδικα που θέλουμε να μετρήσουμε και την εκτελεί επανειλημμένα για να εξασφαλίσει ακριβή αποτελέσματα, λαμβάνοντας υπόψη και τον θόρυβο από το σύστημα. Τα αριθμητικά αποτελέσματα συμβάλλουν στην αξιολόγηση της ταχύτητας απόκρισης των αλγορίθμων και στον εντοπισμό του χρονικά (ως προς την εκτέλεση) αποδοτικότερου αλγορίθμου.

Για την μεγαλύτερη ακρίβεια των αποτελεσμάτων κάθε αλγόριθμος μπορεί να υλοποιείται περισσότερες φορές από μια και ως χρόνος εκτέλεσης να θεωρείται ο μέσος χρόνος που υπολογίζεται κάθε φορά από την "timeit". Στα παραδείγματα που ακολουθούν κάθε αλγόριθμος υλοποιήθηκε 10 φορές και στους πίνακες εμφανίζεται ο μέσος χρόνος, που υπολογίστηκε από την "timeit".

Παράδειγμα 4.1

Θεωρούμε τον 6405×6405 πίνακα αποστάσεων A του Παραδείγματος 3.11, ο οποίος δημιουργήθηκε από τα δεδομένα της πρωτεΐνης 3p16, [33]. Υλοποιώντας διαδοχικά τον Κώδικα 3.3 (Classic Algorithm Lanczos), τον Κώδικα 2.8 (Modified Lanczos Algorithm), τον Κώδικα 2.10 (Reorthogonalization Lanczos Algorithm) και τον Κώδικα 3.15

¹³ Βλέπε, Παράρτημα II, γραμμές 26-28, 31-33, 36-38 και 41-43.

(Proposed Algorithm) για ποικίλες τιμές της παραμέτρου m , 10,20,50, 100, 150, και 200 και χρησιμοποιώντας τη συνάρτηση της Matlab "timeit" υπολογίζεται ο χρόνος εκτέλεσης σε δευτερόλεπτα καθενός από τους προαναφερθέντες αλγορίθμους (βλέπε, Παράρτημα II, γραμμές 26-28, 31-33, 36-38 και 41-43). Η διαδικασία επαναλαμβάνεται δέκα φορές και ως χρόνος εκτέλεσης κάθε αλγορίθμου εμφανίζεται στον Πίνακα 4.1 ο μέσος χρόνος αυτών των υλοποιήσεων.

Πίνακας 4.1: Μέσος χρόνος εκτέλεσης των αλγορίθμων για τα δεδομένα της πρωτεΐνης 3pl6.

Αλγόριθμοι	$m = 10$	$m = 20$	$m = 50$	$m = 100$	$m = 150$	$m = 200$
Classic-Lanczos	0.105007	0.178967	0.461693	1.219030	1.678124	2.300637
Modified-Lanczos	0.104481	0.177735	0.473566	1.405660	1.680726	2.569961
Reorthogonalization-Lanczos	0.114963	0.179807	0.505813	1.725260	2.130587	2.734176
Proposed	0.000289	0.000994	0.009263	0.162424	0.325514	0.467125

Παράδειγμα 4.2

Θεωρούμε ως δεδομένα τις ατομικές συντεταγμένες από τον ιό Ζίκα (ZIKV-5IZ7) από [34], οι οποίες τοποθετούνται σε έναν πίνακα `cord_5IZ7`, διάστασης 13226×3 , (βλέπε, Παράρτημα III). Ακολουθώντας τη μεθοδολογία υπολογισμού που παρουσιάστηκε στο Παράδειγμα 3.2 από τον `cord_5IZ7`, δημιουργείται ο πίνακας αποστάσεων A , που είναι ένας 13226×13226 μη αρνητικός συμμετρικός πίνακας. Υλοποιώντας διαδοχικά τον Κώδικα 3.3 (Classic Algorithm Lanczos), τον Κώδικα 2.8 (Modified Lanczos Algorithm), τον Κώδικα 2.10 (Reorthogonalization Lanczos Algorithm) και τον Κώδικα 3.15 (Proposed Algorithm) για ποικίλες τιμές της παραμέτρου m , 10,20,50, 100, 150, και 200 και χρησιμοποιώντας τη συνάρτηση της Matlab "timeit" υπολογίζεται ο χρόνος εκτέλεσης σε δευτερόλεπτα καθενός από τους προαναφερθέντες αλγορίθμους. Η διαδικασία επαναλαμβάνεται δέκα φορές και ως χρόνος εκτέλεσης κάθε αλγορίθμου εμφανίζεται στον Πίνακα 4.2 ο μέσος χρόνος αυτών των υλοποιήσεων ξεχωριστά για κάθε αλγόριθμο.

Πίνακας 4.2: Μέσος χρόνος εκτέλεσης των αλγορίθμων για τα δεδομένα του ιού ZIKV-5IZ7.

Αλγόριθμοι	$m = 10$	$m = 20$	$m = 50$	$m = 100$	$m = 150$	$m = 200$
Classic-Lanczos	0.447661	0.922441	2.132218	4.198956	5.822627	7.790310
Modified-Lanczos	0.445351	0.912521	2.148368	4.226151	5.839665	7.815961
Reorthogonalization-Lanczos	0.450688	0.933882	2.203846	4.360312	6.233680	8.481842
Proposed	0.003071	0.006017	0.013103	0.050150	0.240380	0.368503

Σχόλια 4.3

Συγκρίνοντας τα αποτελέσματα που καταγράφηκαν στους Πίνακες 4.1 και 4.2 συμπεραίνουμε ότι ανεξάρτητα από τη διάσταση του πίνακα αποστάσεων A έχουμε τα ακόλουθα:

- i. Ο χρόνος εκτέλεσης, των τεσσάρων αλγορίθμων που συγκρίνονται, αυξάνεται με την αύξηση της παραμέτρου m , γεγονός που ευθυγραμμίζεται με την αναμενόμενη αύξηση του υπολογιστικού φόρτου, επειδή αυτός εξαρτάται από τις παραμέτρους m, n σύμφωνα με (2.12) και (1.11).
- ii. Ο προτεινόμενος Αλγόριθμος (βλέπε, Κώδικα 3.15-Proposed Algorithm) έχει τον μικρότερο χρόνο εκτέλεσης από οποιαδήποτε άλλη υλοποίηση αλγορίθμου Lanczos (βλέπε, Κώδικα 3.3-Classic Algorithm Lanczos, Κώδικα 2.8-Modified Lanczos Algorithm και Κώδικα 2.10- Reorthogonalization Lanczos Algorithm).
- iii. Οι υλοποιήσεις του Κλασσικού Αλγορίθμου Lanczos και του Modified Lanczos Αλγορίθμου (βλέπε, Κώδικα 3.3 και Κώδικα 2.8) είναι ταχύτερες από την υλοποίηση του Κώδικα 2.10-Reorthogonalization Lanczos Algorithm. Το πρόσθετο βήμα της ορθογωνοποίησης, που επιτυγχάνεται στην εν λόγω υλοποίηση, την καθιστά αρκετά χρονοβόρα.
- iv. Για σχετικά μεγάλο πλήθος επαναλήψεων $m > 20$, ο χρόνος υλοποίησης του Κώδικα 3.3-Classic Algorithm Lanczos είναι μικρότερος από τον χρόνο υλοποίησης του Κώδικα 2.8-Modified Lanczos Algorithm, ενώ στις περιπτώσεις $m \leq 20$ ο Modified Lanczos Αλγόριθμος είναι ταχύτερος από τον Κλασσικό Αλγόριθμο Lanczos.

4.2 Ανάλυση σφαλμάτων προσέγγισης των ιδιοτιμών

Υπολογίζονται και απεικονίζονται τα σφάλματα μεταξύ των ιδιοτιμών του τριδιαγώνιου πίνακα που προσδιορίζονται από την υλοποίηση των αλγορίθμων και "ορισμένων στο πλήθος" ιδιοτιμών του αρχικού πίνακα αποστάσεων. Ο υπολογισμός των σφαλμάτων βασίζεται στον ακόλουθο τύπο:

$$\text{error} = |x_A - x_T| \quad (4.1)$$

όπου x_A είναι ορισμένες ιδιοτιμές (m στο πλήθος) του αρχικού πίνακα αποστάσεων A , που υπολογίζονται με τη χρήση της ενσωματωμένης συνάρτησης "eigs" της Matlab. Η "eigs" υπολογίζει τις ιδιοτιμές του πίνακα με την ίδια ακρίβεια που τις υπολογίζει και η συνάρτηση "eig", εστιάζοντας μόνο σε ένα υποσύνολο του φάσματος πλήθους m . Η χρήση της "eigs" συνιστάται στις περιπτώσεις πινάκων μεγάλης διάστασης, όπου η υλοποίηση της συνάρτησης "eig" λόγω μεγάλου υπολογιστικού φόρτου είναι απαγορευτική. Αντίστοιχα, το x_T αντιστοιχεί στις ιδιοτιμές του τριδιαγώνιου πίνακα T_m , που υπολογίζονται από τον εκάστοτε αλγόριθμο καθώς αυτός υλοποιείται κάνοντας m επαναλήψεις. Αυτή η ανάλυση επιτρέπει την εκτίμηση της ακρίβειας κάθε αλγορίθμου και την επιλογή του πιο αξιόπιστου αλγορίθμου. Η αναπαράσταση των σφαλμάτων πραγματοποιείται με δύο τύπους διαγραμμάτων:

- **Διαγράμματα σημείων (Scatter Plots).** Αυτά τα διαγράμματα¹⁴ παρέχουν πληροφορίες για τη σχετική θέση των υπολογισμένων ιδιοτιμών x_A ως προς τις x_T και επιτρέπουν την αξιολόγηση της οπτικής ταύτισής τους οδηγώντας κατά αυτόν τον τρόπο σε συμπεράσματα για την ακρίβεια κάθε αλγορίθμου. Στον οριζόντιο άξονα εμφανίζεται το πλήθος των επαναλήψεων m , το οποίο καθορίζει και τη διάσταση του εκάστοτε τριδιαγώνιου πίνακα και στον κατακόρυφο άξονα αποτυπώνεται η αντίστοιχη ιδιοτιμή του T_m .
- **Ραβδογράμματα (Bar Plots).** Αυτά τα διαγράμματα¹⁵ αποτυπώνουν την ποσοτικοποίηση των σφαλμάτων, σκιαγραφώντας μέσω της (4.1) το μέγεθος της αριθμητικής απόκλισης (αριθμητικό σφάλμα) μεταξύ των ιδιοτιμών, που υπολογίζονται. Το μέγεθος της απόκλισης εμφανίζεται ως ύψος της αντίστοιχης ράβδου. Τα ραβδογράμματα δίνουν σαφέστερη αναπαράσταση των σφαλμάτων, λόγω σκιαγράφησης μεγαλύτερης επιφάνειας (αυτής των ράβδων), συγκριτικά με τη σημειακή αναπαράσταση που εμφανίζεται μέσω των διαγραμμάτων σημείων με αποτέλεσμα να υπάρχει ακριβέστερη εκτίμηση των σφαλμάτων.

Παράδειγμα 4.6

Θεωρούμε τον 6405×6405 πίνακα αποστάσεων A του Παραδείγματος 4.1, ο οποίος δημιουργήθηκε από τα δεδομένα της πρωτεύουσας 3p16, [33].

Για τον προαναφερόμενο A υλοποιείται ο Κώδικας 3.3 και η συνάρτηση "eigs" της Matlab

¹⁴ Βλέπε, Παράρτημα II στις γραμμές 48-63.

¹⁵ Βλέπε, Παράρτημα II στις γραμμές 64-77.

- για $m = 3$. Χωρίς βλάβη της γενικότητας, θεωρούμε ότι υπολογίστηκε ο πίνακας T_3 του Παραδείγματος 3.4, ο οποίος έχει τις διατεταγμένες ιδιοτιμές:

$$\lambda_1 = -0.3685 \cdot 10^5 < \lambda_2 = -0.0006 \cdot 10^5 < \lambda_3 = 3.0867 \cdot 10^5$$

Οι τρεις διατεταγμένες ιδιοτιμές χ_A του πίνακα αποστάσεων A είναι

$$\ell_1 = -1.2833 \cdot 10^5 < \ell_2 = -0.3695 \cdot 10^5 < \ell_3 = 3.0867 \cdot 10^5. \quad (4.2)$$

- Για $m = 4$, έστω ότι υπολογίζεται ένας τριδιαγώνιος πίνακας T_4 με ιδιοτιμές:

$$\tilde{\lambda}_1 = -1.2762 \cdot 10^5, \tilde{\lambda}_2 = -0.3672 \cdot 10^5, \tilde{\lambda}_3 = -0.0005 \cdot 10^5, \tilde{\lambda}_4 = 3.0867 \cdot 10^5$$

Οι τέσσερις ιδιοτιμές χ_A του πίνακα αποστάσεων A είναι

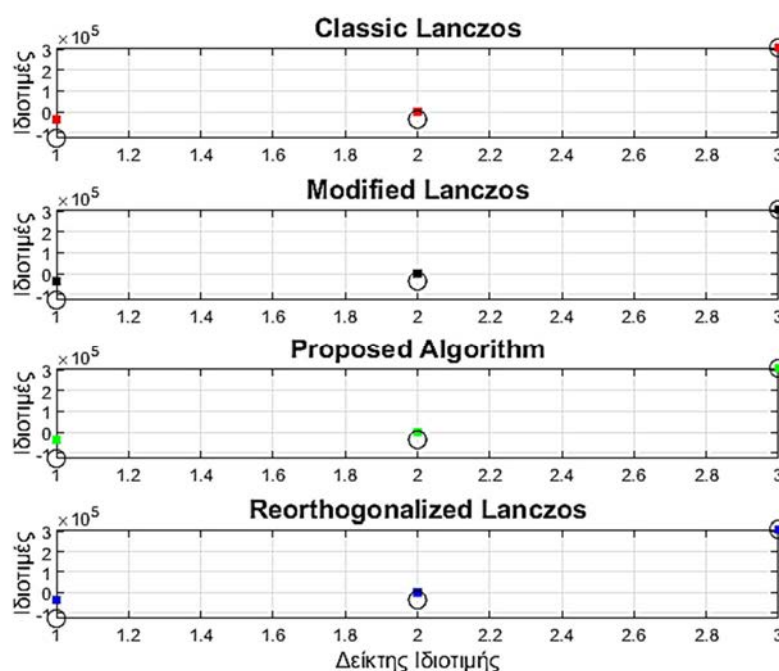
$$\tilde{\ell}_1 = -1.2833 \cdot 10^5, \tilde{\ell}_2 = -0.3695 \cdot 10^5, \tilde{\ell}_3 = -0.2554 \cdot 10^5, \tilde{\ell}_4 = 3.0867 \cdot 10^5 \quad (4.3)$$

Μέσω των γραμμών 53-90 του Παραρτήματος II γίνεται η υλοποίηση των Κωδίκων 2.8, 2.10, 3.3 και 3.15, οπότε απεικονίζονται οι ιδιοτιμές χ_T των προαναφερόμενων T_m , και οι m πλήθους ιδιοτιμές χ_A του A , για $m = 3$, και $m = 4$.

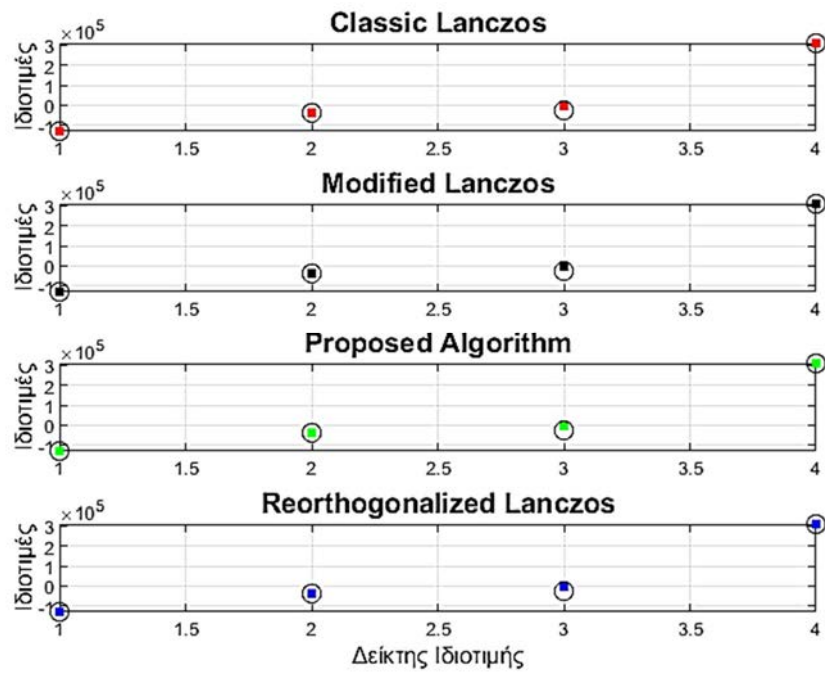
Οι αναπαραστάσεις γίνονται με

- ο τα **διαγράμματα σημείων (Scatter Plots)** στα Σχήματα 4.3-4.4, τα οποία σχεδιάζονται από την υλοποίηση του κώδικα στις γραμμές 53-73. Σε αυτά, οι m πλήθους ιδιοτιμές χ_A του πίνακα αποστάσεων A αποτυπώνονται με μαύρους κύκλους σε όλες τις υλοποιήσεις και οι ιδιοτιμές χ_T σημειώνονται με έγχρωμα τετράγωνα, ανάλογα με την υλοποίηση: κόκκινο (Κώδικας 3.3-Classic Algorithm Lanczos), μαύρο (Κώδικας 2.8-Modified Lanczos Algorithm), πράσινο (Κώδικας 3.15-Proposed Algorithm) και μπλε (Κώδικας 2.10-Reorthogonalization Lanczos Algorithm).
- ο τα **ραβδογράμματα (Bar Plots)** στα Σχήματα 4.5-4.6, τα οποία σχεδιάζονται από την υλοποίηση του κώδικα στις γραμμές 74-90. Σε αυτά, το ύψος κάθε ράβδου αντιστοιχεί στην τιμή της απόλυτης διαφοράς των αντίστοιχων ιδιοτιμών, χ_T, χ_A .

ο Διαγράμματα σημείων (Scatter Plots)



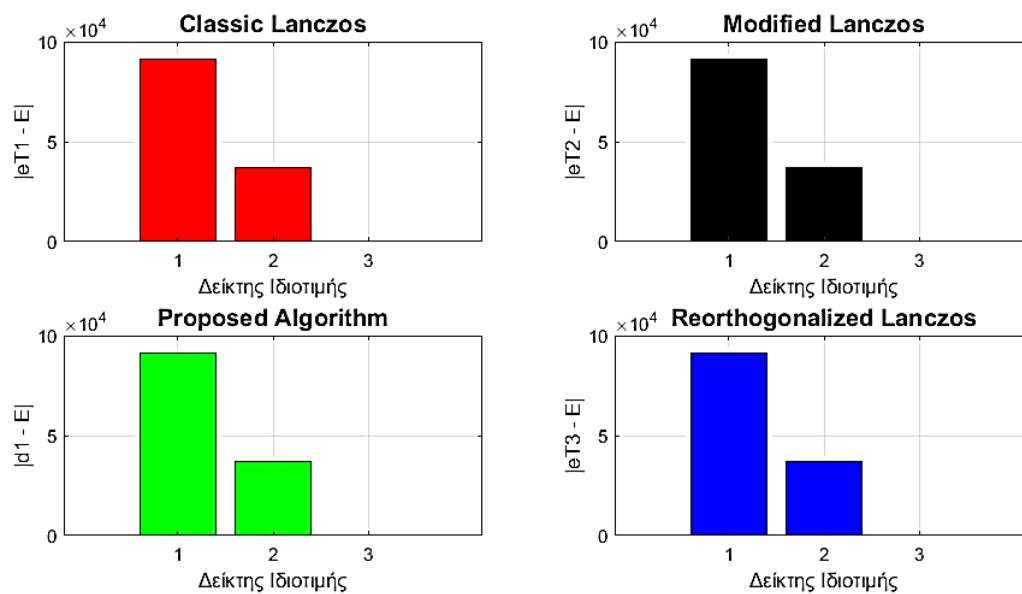
Σχήμα 4.3: Οι ιδιοτιμές χ_T του T_3 με έγχρωμα τετράγωνα και με κύκλους οι ιδιοτιμές χ_A του A για την πρωτεΐνη 3p16.



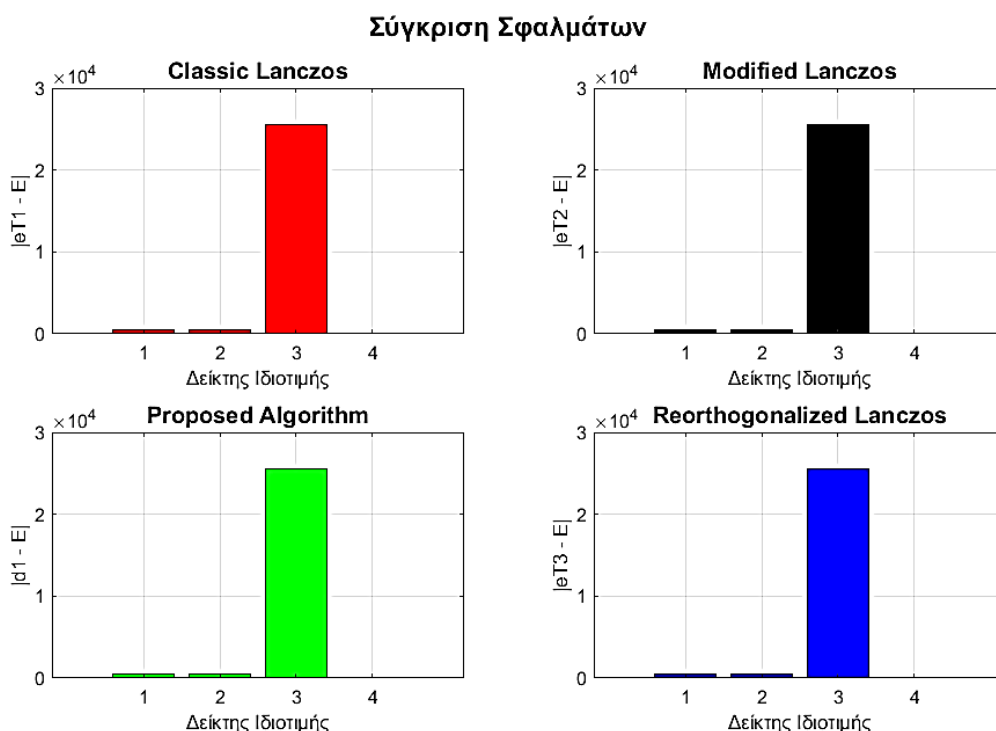
Σχήμα 4.4: Οι ιδιοτιμές χ_T του T_4 με έγχρωμα τετράγωνα και με κύκλους οι ιδιοτιμές χ_A του A για την πρωτεΐνη 3p16.

Ο Ραβδογράμματα (Bar Plots)

Σύγκριση Σφαλμάτων



Σχήμα 4.5: Η απόλυτη διαφορά των m ιδιοτιμών με $m = 3$ των δεδομένων της 3p16.



Σχήμα 4.6: Η απόλυτη διαφορά των m ιδιοτιμών με $m = 4$ των δεδομένων της 3p16.

Σχόλια 4.7

- i. Για τις ιδιοτιμές του πίνακα αποστάσεων A , που αναφέρονται στην (4.3), επαληθεύουν την (3.4), δηλαδή, οι ιδιοτιμές $\tilde{\ell}_i$, $i = 1, 2, 3$ ανήκουν στο διάστημα $(-\tilde{\ell}_4, \tilde{\ell}_4)$.
- ii. Επιπλέον, συγκρίνοντας τις ιδιοτιμές του πίνακα αποστάσεων A , που αναφέρονται στις (4.2) και (4.3), παρατηρούμε ότι ισχύουν:
 - $\ell_3 \equiv \tilde{\ell}_4$ ως φασματική ακτίνα του πίνακα A , και
 - $\ell_1 \equiv \tilde{\ell}_1$, $\ell_2 \equiv \tilde{\ell}_2$.

Συνεπώς, ορισμένες ιδιοτιμές του A παραμένουν ίδιες και καθώς αυξάνεται η τιμή της παραμέτρου m προστίθενται ενδιάμεσες ιδιοτιμές ανάμεσα στις προηγούμενες, που είχαν υπολογιστεί, και στη φασματική ακτίνα.

Στην τελευταία περίπτωση ενσφηνώθηκε μία νέα ιδιοτιμή, η οποία είναι η $\tilde{\ell}_3$. Δείτε και συγκρίνετε την ανάλογη παρατήρηση στο Σχόλιο 3.12(vi) για την περίπτωση $m = 5$.
- iii. Από τις αναπαραστάσεις των Σχημάτων 4.3-4.6 συμπεραίνουμε ότι στις περιπτώσεις $m = 3$, και $m = 4$, οι τέσσερις Αλγόριθμοι (Κώδικας 3.3-Classic Algorithm Lanczos, Κώδικας 2.8-Modified Lanczos Algorithm, Κώδικας 2.10-Reorthogonalization Lanczos Algorithm, Κώδικας 3.15-Proposed Algorithm) υπολογίζουν με πολύ μεγάλη ακρίβεια **μόνο** τη φασματική ακτίνα του πίνακα αποστάσεων A , γεγονός που μπορεί να διαπιστωθεί από τα διαγράμματα σημείων καθώς και από τα ραβδογράμματα.

- ο Στα **διαγράμματα σημείων** (βλέπε, Σχήματα 4.3 και 4.4) παρατηρούμε ότι: οι δύο ή τρεις μικρότερες ιδιοτιμές των τριδιαγώνιων πινάκων T_3 , T_4 , αντίστοιχα, που υπολογίζονται από τις υλοποιήσεις των τεσσάρων προαναφερόμενων αλγορίθμων, δεν αποτελούν ικανοποιητικές προσεγγίσεις των ιδιοτιμών του A , επειδή τα έγχρωμα τετράγωνα, τα οποία σημειώνουν τις ιδιοτιμές των T_3 , T_4 , δεν εσωκλείονται στους κύκλους, οι οποίοι σημειώνουν τις ιδιοτιμές του A . Ο εν λόγω εγκλεισμός ισχύει μόνο στην περίπτωση της φασματικής ακτίνας, όταν υλοποιείται οποιοσδήποτε από τους προαναφερόμενους αλγορίθμους.
- ο Επειδή το ύψος κάθε ράβδου στο ραβδόγραμμα ισούται με το σφάλμα της αντίστοιχης προσέγγισης ορισμένων (m στο πλήθος) ιδιοτιμών του πίνακα αποστάσεων A από τις ιδιοτιμές του τριδιαγώνιου πίνακα T_m , παρατηρώντας τα **ραβδογράμματα** (βλέπε, Σχήματα 4.5 και 4.6), συμπεραίνουμε ότι με την υλοποίηση οποιουδήποτε Κώδικα στις δύο ή τρεις μικρότερες ιδιοτιμές τα αντίστοιχα σφάλματα "φαίνεται ότι **δεν** είναι μηδενικά" και μόνο το σφάλμα της φασματικής ακτίνας "φαίνεται ότι είναι μηδενικό". Επομένως, η προσέγγιση μόνο της φασματικής ακτίνας του A από τη φασματική του T_3 (η οποία ταυτίζεται με τη φασματική ακτίνα του T_4) είναι ικανοποιητική, επειδή το αντίστοιχο σφάλμα είναι μηδενικό.

Ενδεικτικά επαληθεύουμε τα προαναφερόμενα με αντικατάσταση στην (4.1) των δεδομένων ιδιοτιμών του T_3 , που υπολογίστηκαν από την υλοποίηση του Κώδικα 3.3 και τις ιδιοτιμές του A από την (4.2), (βλέπε, Σχόλια 3.5(v)), από όπου έχουμε:

$$|\ell_1 - \lambda_1| = 0.9148 \cdot 10^5, \quad |\ell_2 - \lambda_2| = 0.3689 \cdot 10^5, \quad |\ell_3 - \lambda_3| = 0 \quad (4.4)$$

Ανάλογα για $m = 4$, με αντικατάσταση στην (4.1) των δεδομένων ιδιοτιμών του T_4 και τις ιδιοτιμές του A από την (4.2), υπολογίζονται τα ακόλουθα σφάλματα:

$$|\tilde{\ell}_1 - \tilde{\lambda}_1| = 710, \quad |\tilde{\ell}_2 - \tilde{\lambda}_2| = 230, \quad |\tilde{\ell}_3 - \tilde{\lambda}_3| = 25490, \quad |\tilde{\ell}_4 - \tilde{\lambda}_4| = 0 \quad (4.5)$$

Από τις (4.4) και (4.5) παρατηρούμε ότι μόνο στην περίπτωση της φασματικής ακτίνας δεν υπάρχει σφάλμα στην προσέγγιση, ενώ στις υπόλοιπες δύο ή τρεις προσεγγίσεις το σφάλμα είναι διάφορο του μηδενός και μάλιστα στην (4.4) το δεύτερο σφάλμα και στην (4.5) το τρίτο σφάλμα έχει τάξη μεγέθους όση και οι ιδιοτιμές, δηλαδή 10^5 .

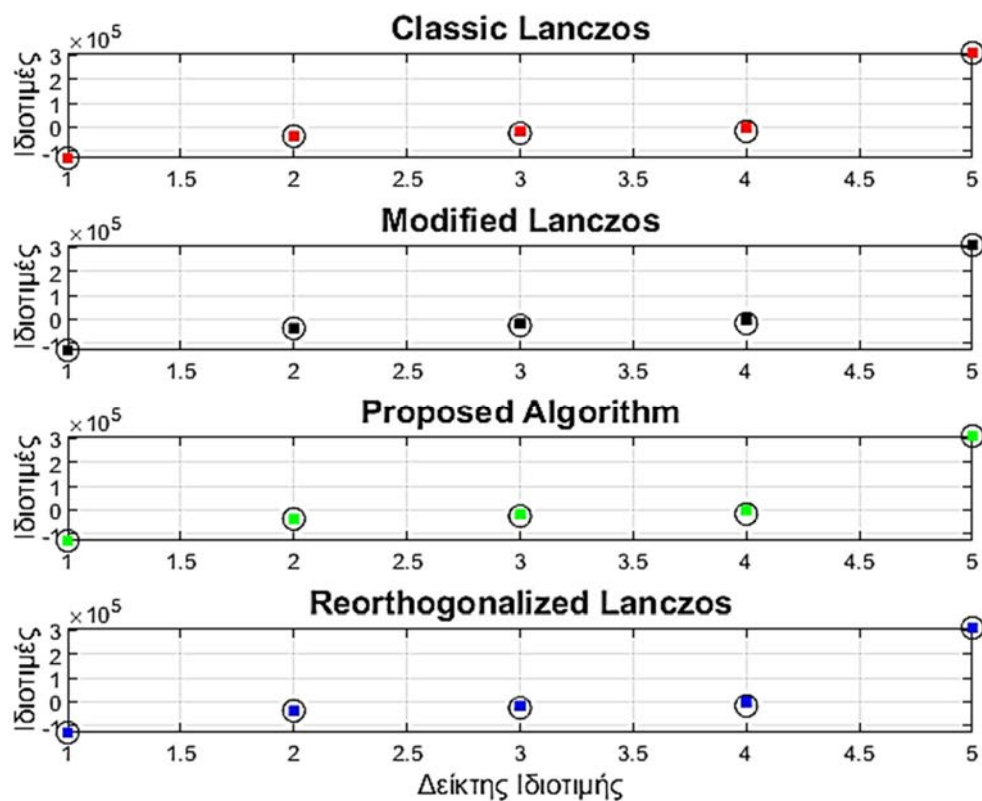
Παράδειγμα 4.8

Θεωρούμε τον 6405×6405 πίνακα αποστάσεων A του Παραδείγματος 4.1, ο οποίος δημιουργήθηκε από τα δεδομένα της πρωτεΐνης 3p16. Μέσω της υλοποίησης των Κωδίκων 2.8, 2.10, 3.3 και 3.15, βλέπε στις γραμμές 78-209 του Παραρτήματος II, υπολογίζονται και απεικονίζονται οι ιδιοτιμές χ_T των τριδιαγώνιων πινάκων T_m , και m πλήθους ιδιοτιμές χ_A του πίνακα αποστάσεων A , για ποικίλες τιμές της παραμέτρου m , 5,10,20,50,100,150, και 200. Μετά την υλοποίηση του καθενός από τους προαναφερθέντες Κώδικες από την (4.1) επιπρόσθετα υπολογίζονται τα σφάλματα των ακραίων ιδιοτιμών των τριδιαγώνιων πινάκων T_m και A (δηλαδή απόλυτη διαφορά φασματικών ακτίνων και απόλυτη διαφορά των ελαχίστων ιδιοτιμών των αντίστοιχων πινάκων), βλέπε στις γραμμές 210-228 του Παραρτήματος II. Η διαδικασία υπολογισμού των σφαλμάτων των ακραίων ιδιοτιμών επαναλαμβάνεται δέκα φορές και ο μέσος όρος των σφαλμάτων, που υπολογίζεται για κάθε τιμή της παραμέτρου m , αποτυπώνεται στον Πίνακα 4.21.

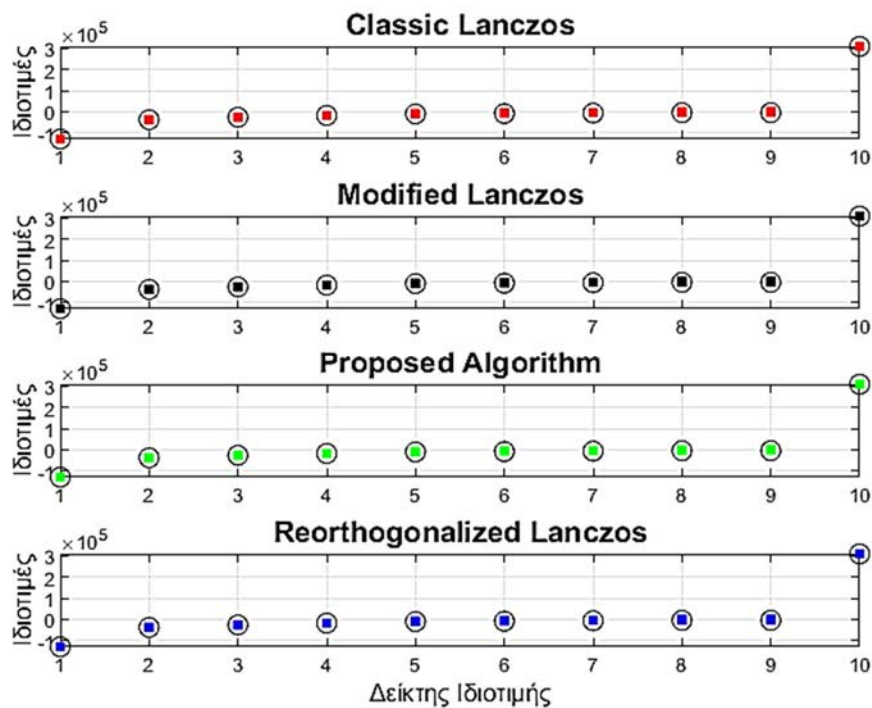
Οι αναπαραστάσεις των ιδιοτιμών γίνονται με

- ο τα **διαγράμματα σημείων (Scatter Plots)** στα Σχήματα 4.7-4.13, τα οποία σχεδιάζονται από την υλοποίηση του κώδικα στις γραμμές 48-63. Σε αυτά, οι m πλήθους ιδιοτιμές χ_A του πίνακα αποστάσεων A αποτυπώνονται με μαύρους κύκλους σε όλες τις υλοποιήσεις και οι ιδιοτιμές χ_T σημειώνονται με έγχρωμα τετράγωνα, ανάλογα με την υλοποίηση: κόκκινο (Κώδικας 3.3-Classic Algorithm Lanczos), μαύρο (Κώδικας 2.8-Modified Lanczos Algorithm), πράσινο (Κώδικας 3.15-Proposed Algorithm) και μπλε (Κώδικας 2.10-Reorthogonalization Lanczos Algorithm).
- ο τα **ραβδογράμματα (Bar Plots)** στα Σχήματα 4.14-4.20, τα οποία σχεδιάζονται από την υλοποίηση του κώδικα στις γραμμές 64-77. Σε αυτά, το ύψος κάθε ράβδου αντιστοιχεί στην τιμή της απόλυτης διαφοράς των αντίστοιχων ιδιοτιμών, χ_T, χ_A .

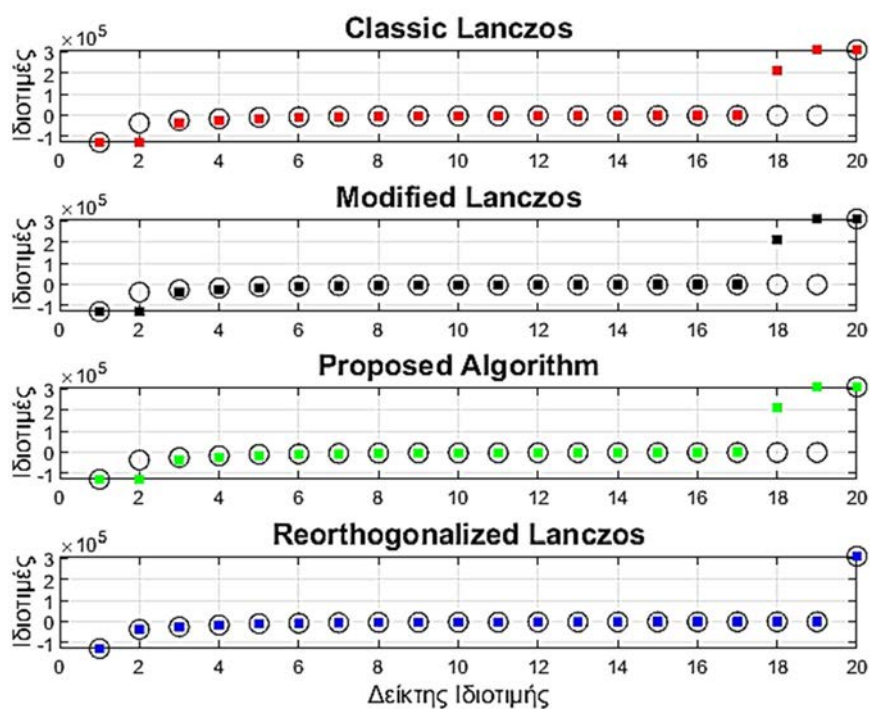
ο Διαγράμματα σημείων (Scatter Plots)



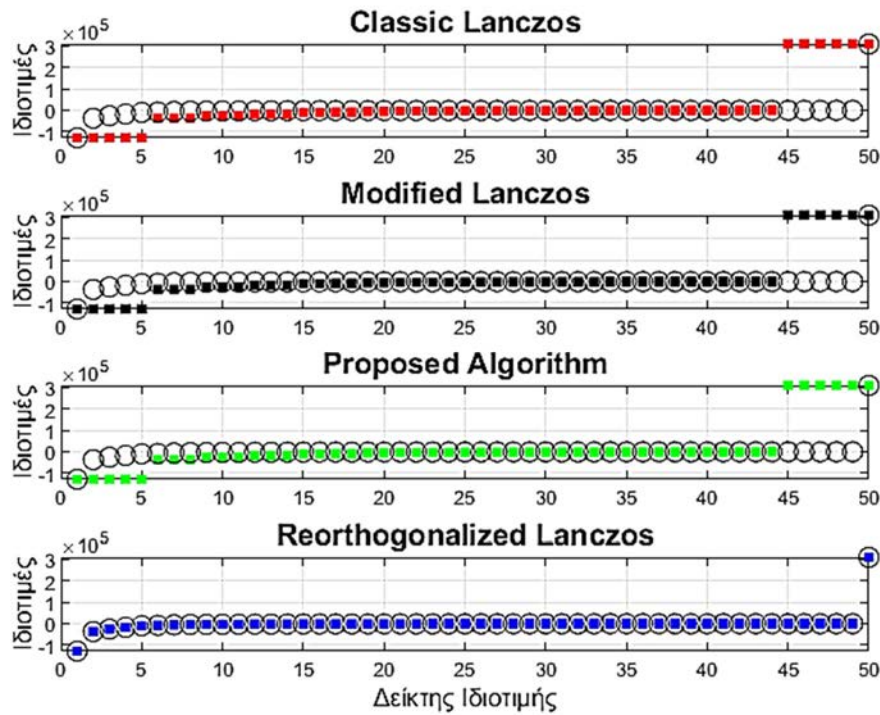
Σχήμα 4.7: Οι ιδιοτιμές χ_T του T_5 με έγχρωμα τετράγωνα και με κύκλους οι ιδιοτιμές χ_A του A για την πρωτεΐνη 3p16.



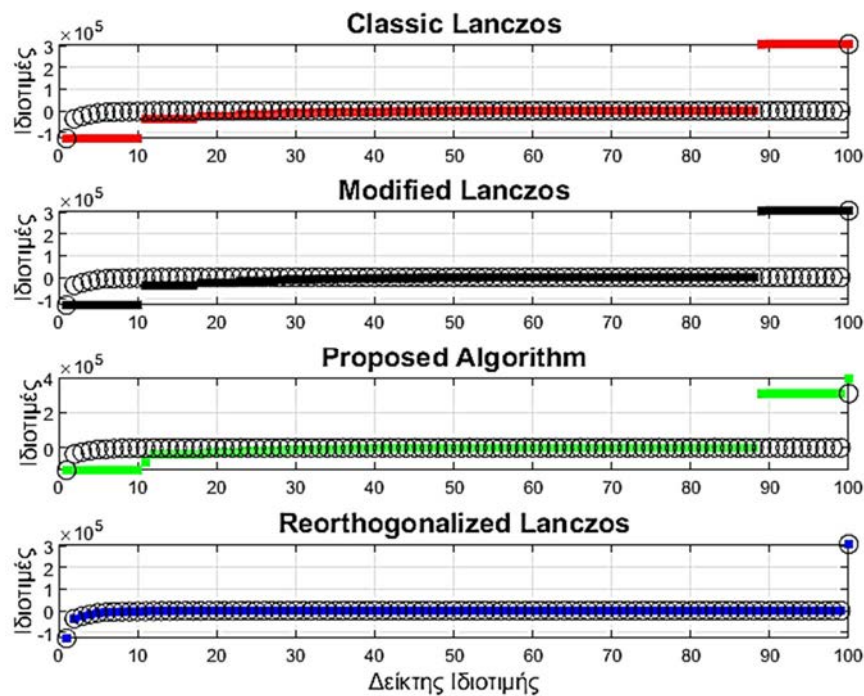
Σχήμα 4.8: Οι ιδιοτιμές χ_T του T_{10} με έγχρωμα τετράγωνα και με κύκλους οι ιδιοτιμές χ_A του A για την πρωτεΐνη 3p16.



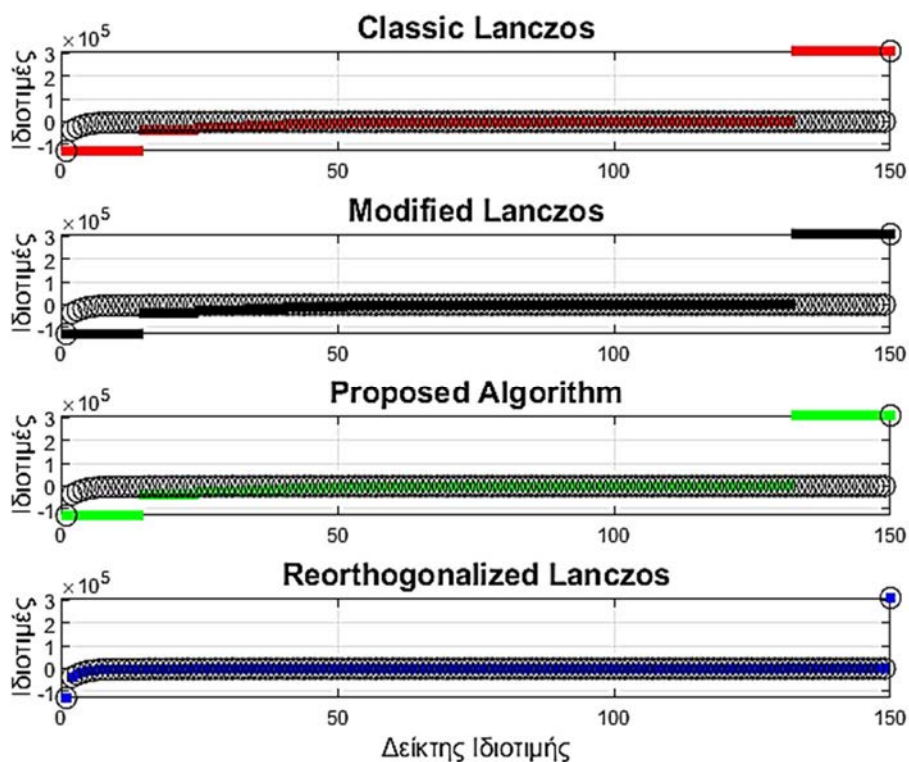
Σχήμα 4.9: Οι ιδιοτιμές χ_T του T_{20} με έγχρωμα τετράγωνα και με κύκλους οι ιδιοτιμές χ_A του A για την πρωτεΐνη 3p16.



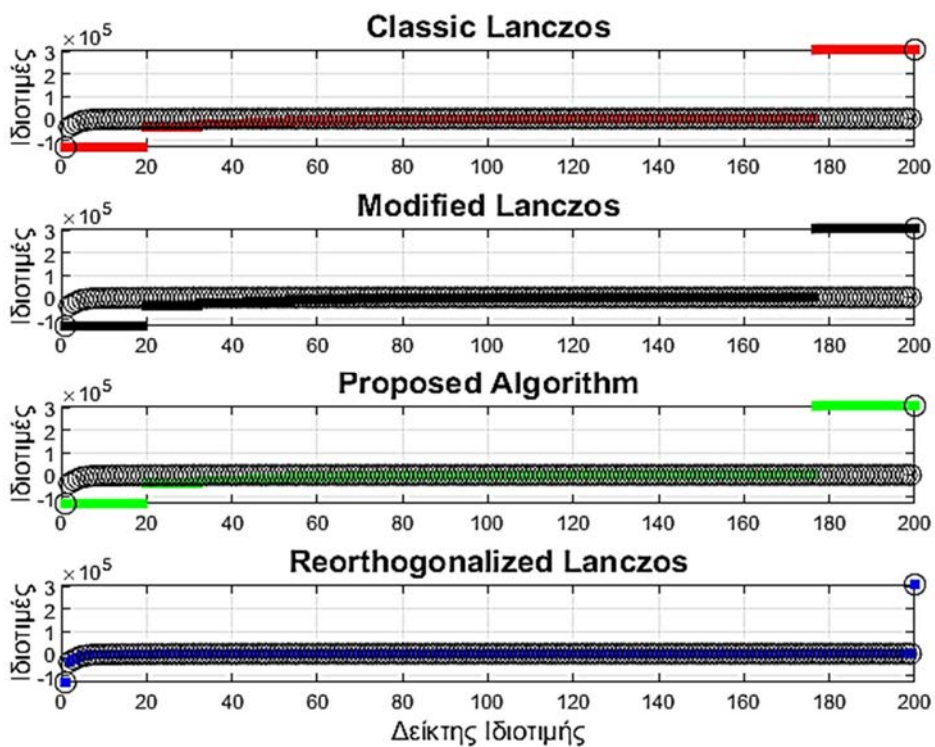
Σχήμα 4.10: Οι ιδιοτιμές χ_T του T_{50} με έγχρωμα τετράγωνα και με κύκλους οι ιδιοτιμές χ_A του A για την πρωτεΐνη 3p16.



Σχήμα 4.11: Οι ιδιοτιμές χ_T του T_{100} με έγχρωμα τετράγωνα και με κύκλους οι ιδιοτιμές χ_A του A για την πρωτεΐνη 3p16.

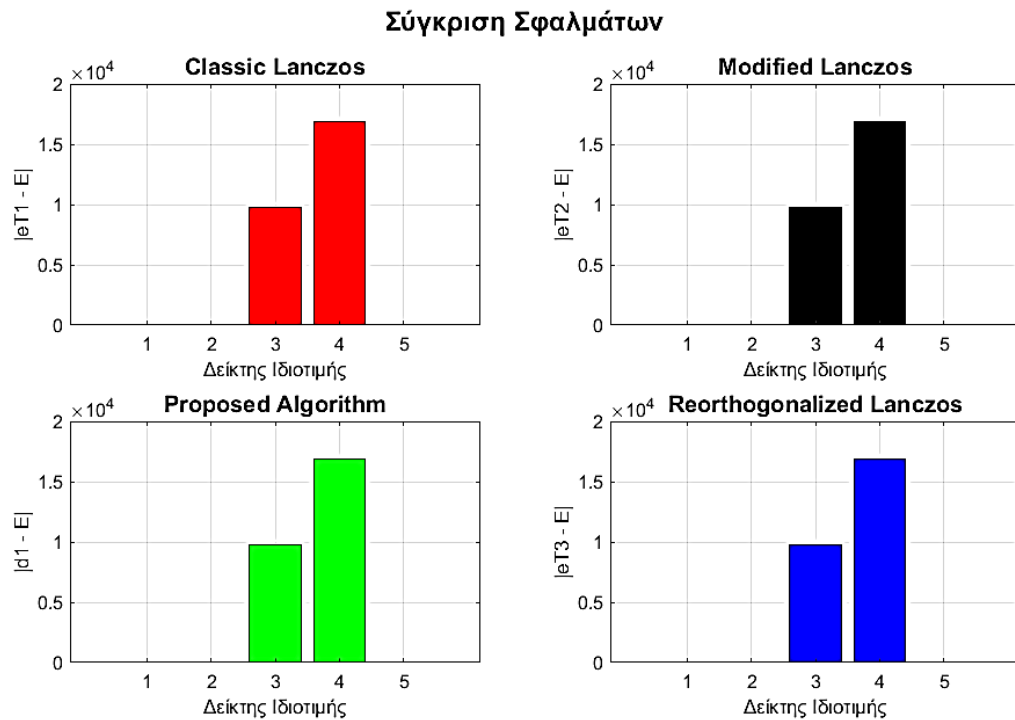


Σχήμα 4.12: Οι ιδιοτιμές χ_T του T_{150} με έγχρωμα τετράγωνα και με κύκλους οι ιδιοτιμές χ_A του A για την πρωτεΐνη 3p16.

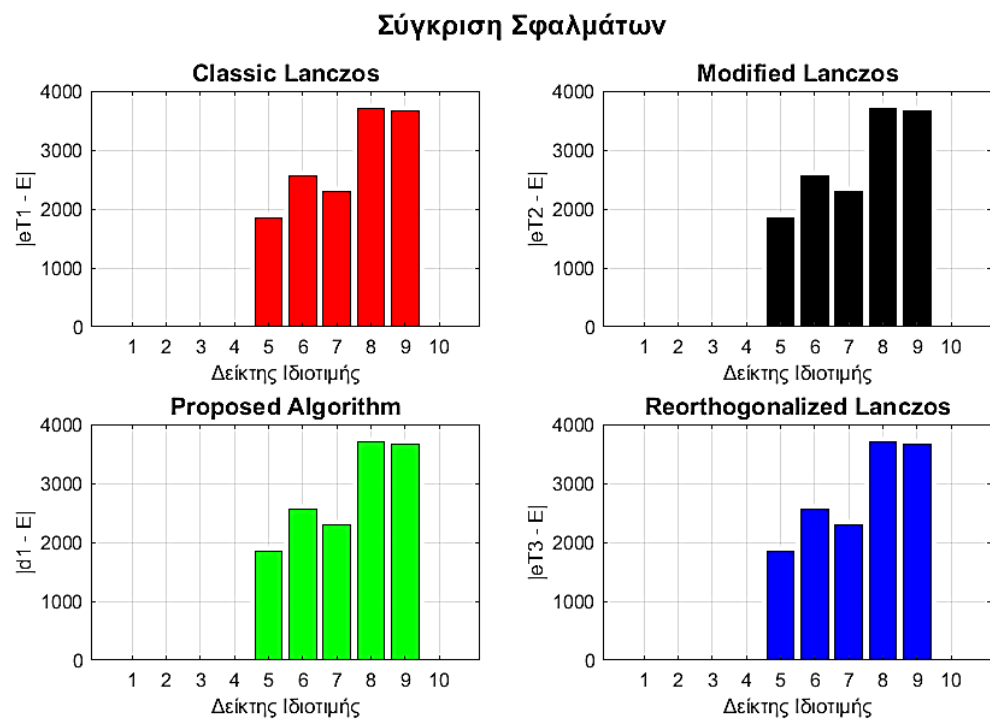


Σχήμα 4.13: Οι ιδιοτιμές χ_T του T_{200} με έγχρωμα τετράγωνα και με κύκλους οι ιδιοτιμές χ_A του A για την πρωτεΐνη 3p16.

ο Ραβδογράμματα (Bar Plots)

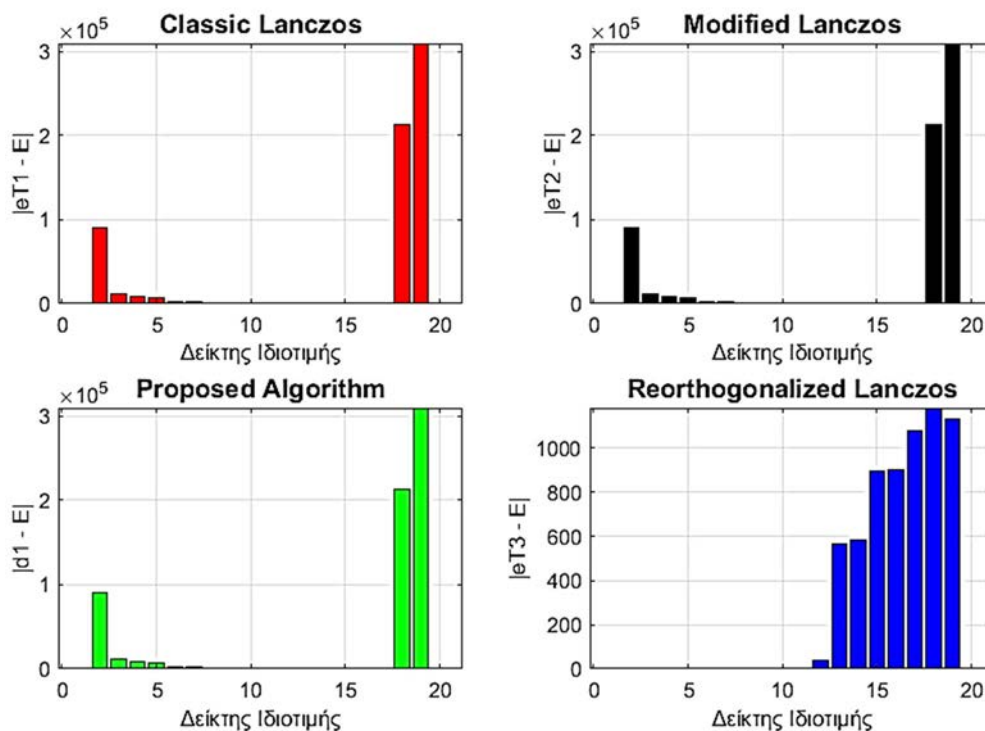


Σχήμα 4.14: Η απόλυτη διαφορά των m ιδιοτιμών με $m = 5$ των δεδομένων της 3p16.



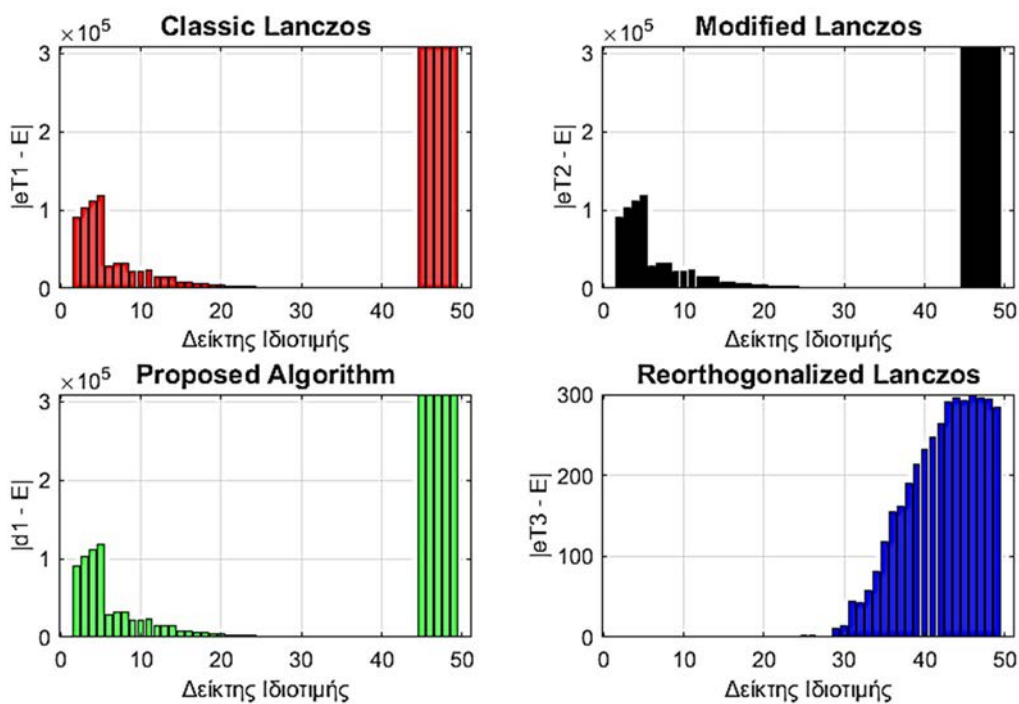
Σχήμα 4.15: Η απόλυτη διαφορά των m ιδιοτιμών με $m = 10$ των δεδομένων της 3p16.

Σύγκριση Σφαλμάτων



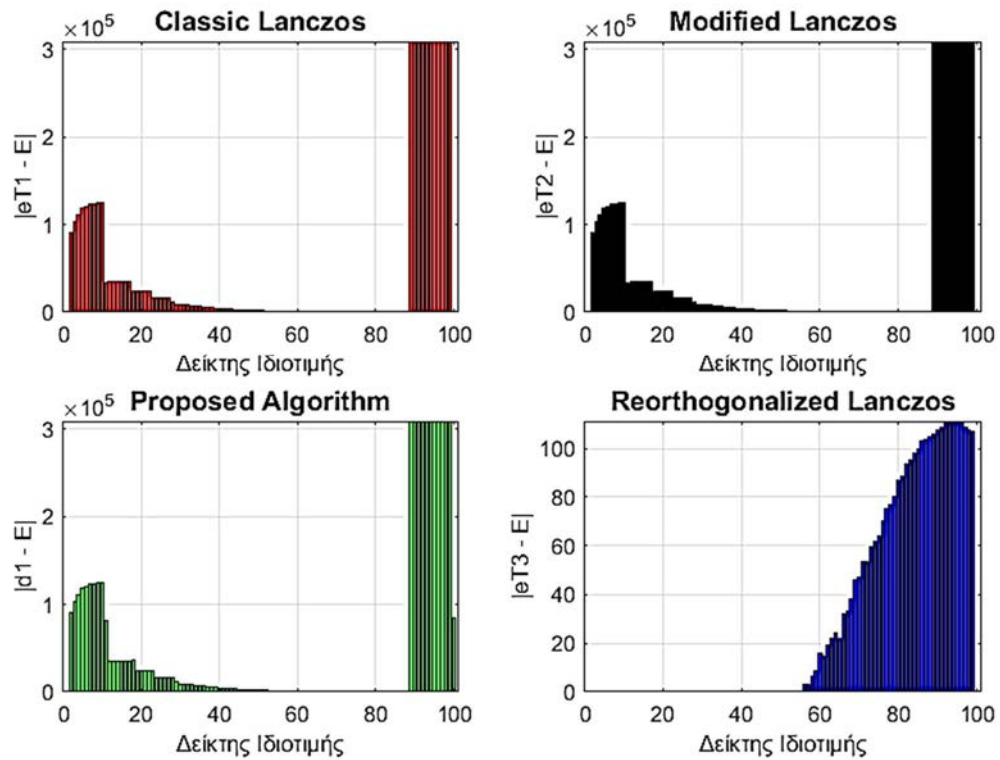
Σχήμα 4.16: Η απόλυτη διαφορά των m ιδιοτιμών με $m = 20$ των δεδομένων της 3p16.

Σύγκριση Σφαλμάτων



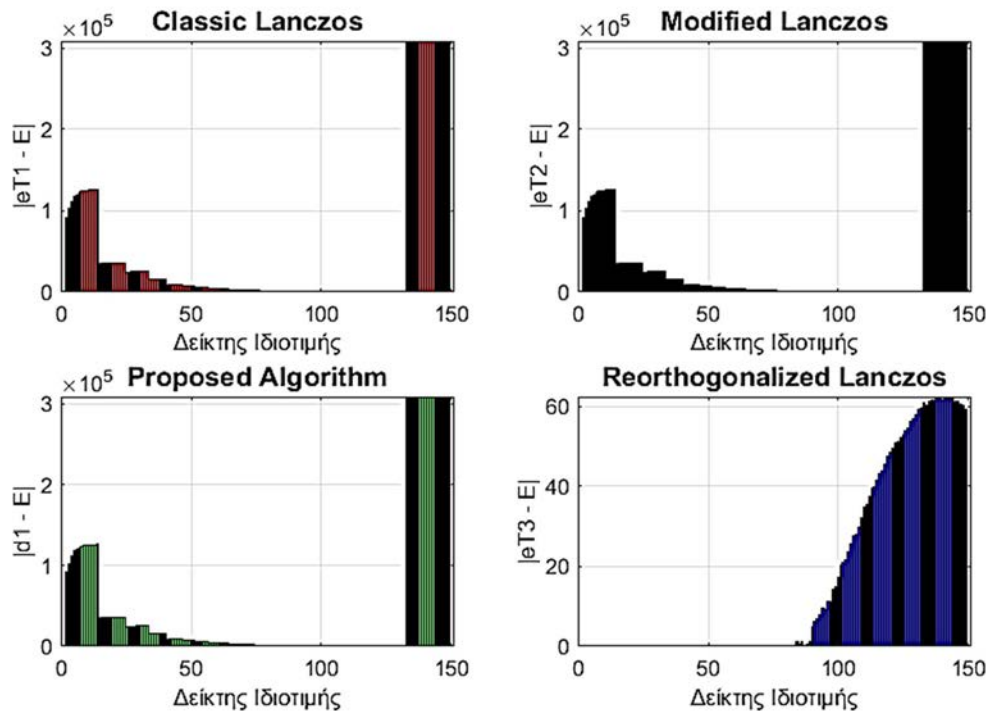
Σχήμα 4.17: Η απόλυτη διαφορά των m ιδιοτιμών με $m = 50$ των δεδομένων της 3p16.

Σύγκριση Σφαλμάτων



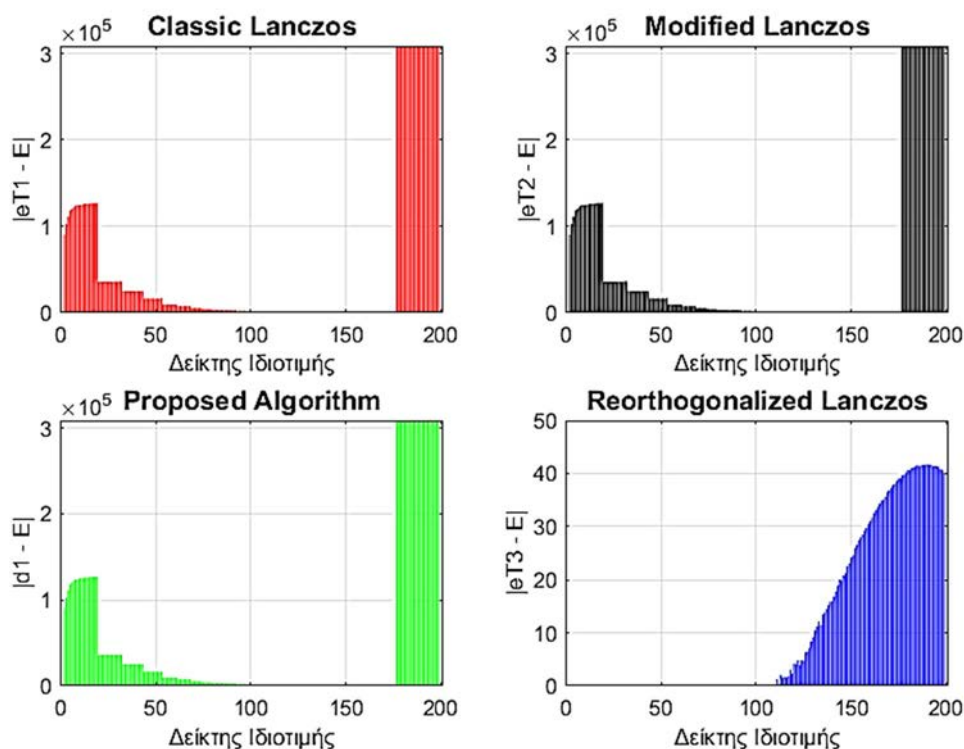
Σχήμα 4.18: Η απόλυτη διαφορά των m ιδιοτιμών με $m = 100$ των δεδομένων της 3p16.

Σύγκριση Σφαλμάτων



Σχήμα 4.19: Η απόλυτη διαφορά των m ιδιοτιμών με $m = 150$ των δεδομένων της 3p16.

Σύγκριση Σφαλμάτων



Σχήμα 4.20: Η απόλυτη διαφορά των m ιδιοτιμών με $m = 200$ των δεδομένων της 3p16.

Πίνακας 4.21: Σφάλματα ακραίων ιδιοτιμών των πινάκων T_m και A για την πρωτεΐνη 3p16.

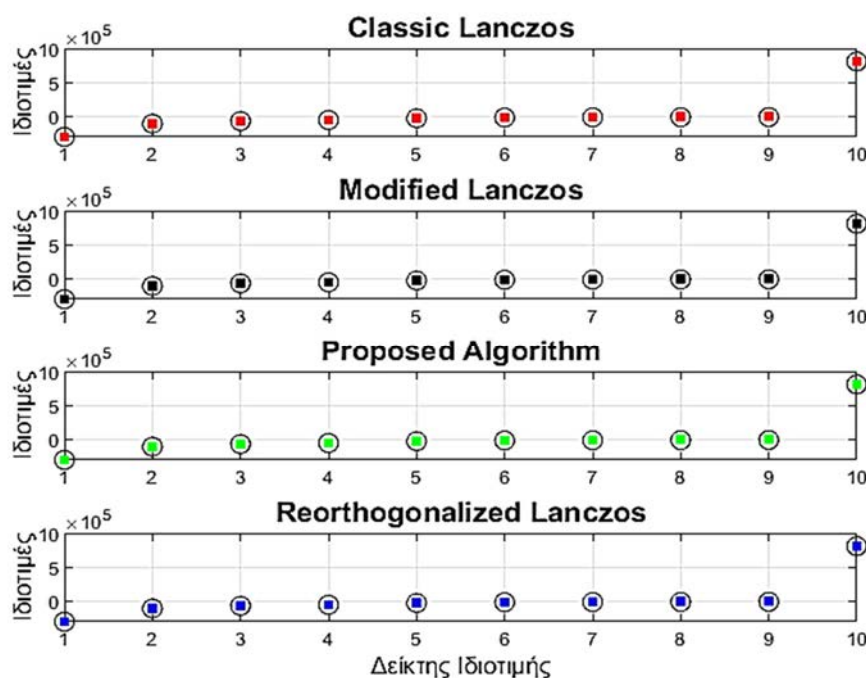
Φασματική ακτίνα							
Αλγόριθμοι	$m = 5$	$m = 10$	$m = 20$	$m = 50$	$m = 100$	$m = 150$	$m = 200$
Classic-Lanczos	$1.1 \cdot 10^{-7}$	$3.5 \cdot 10^{-10}$	$1.2 \cdot 10^{-10}$	$4.1 \cdot 10^{-10}$	$3.5 \cdot 10^{-10}$	$1.7 \cdot 10^{-10}$	$4.1 \cdot 10^{-10}$
Modified-Lanczos	$1.1 \cdot 10^{-7}$	$3.5 \cdot 10^{-10}$	$1.2 \cdot 10^{-10}$	$4.1 \cdot 10^{-10}$	$3.5 \cdot 10^{-10}$	$1.7 \cdot 10^{-10}$	$4.1 \cdot 10^{-10}$
Reorthogonalization-Lanczos	$1.1 \cdot 10^{-7}$	$2.3 \cdot 10^{-10}$	$5.8 \cdot 10^{-11}$	$5.8 \cdot 10^{-11}$	$5.8 \cdot 10^{-11}$	$5.8 \cdot 10^{-11}$	$5.8 \cdot 10^{-11}$
Proposed	$1.1 \cdot 10^{-7}$	$2.3 \cdot 10^{-10}$	0	0	$5.8 \cdot 10^{-11}$	$1.2 \cdot 10^{-10}$	$1.2 \cdot 10^{-10}$
Ελάχιστη ιδιοτιμή							
Αλγόριθμοι	$m = 5$	$m = 10$	$m = 20$	$m = 50$	$m = 100$	$m = 150$	$m = 200$
Classic-Lanczos	$1.4 \cdot 10^2$	$1.5 \cdot 10^{-10}$	$1.6 \cdot 10^{-10}$	$1.5 \cdot 10^{-11}$	$1.3 \cdot 10^{-10}$	$1.0 \cdot 10^{-10}$	$1.5 \cdot 10^{-11}$
Modified-Lanczos	$1.4 \cdot 10^2$	$1.5 \cdot 10^{-10}$	$1.6 \cdot 10^{-10}$	$1.5 \cdot 10^{-11}$	$1.3 \cdot 10^{-10}$	$1.0 \cdot 10^{-10}$	$1.5 \cdot 10^{-11}$
Reorthogonalization-Lanczos	$1.4 \cdot 10^2$	$1.2 \cdot 10^{-10}$	$1.5 \cdot 10^{-10}$	$1.5 \cdot 10^{-11}$	$5.8 \cdot 10^{-11}$	$8.7 \cdot 10^{-11}$	$2.9 \cdot 10^{-11}$
Proposed	$1.4 \cdot 10^2$	$1.2 \cdot 10^{-10}$	$1.4 \cdot 10^{-10}$	$1.5 \cdot 10^{-11}$	$1.2 \cdot 10^{-10}$	$8 \cdot 10^{-11}$	$1.5 \cdot 10^{-11}$

Παράδειγμα 4.9

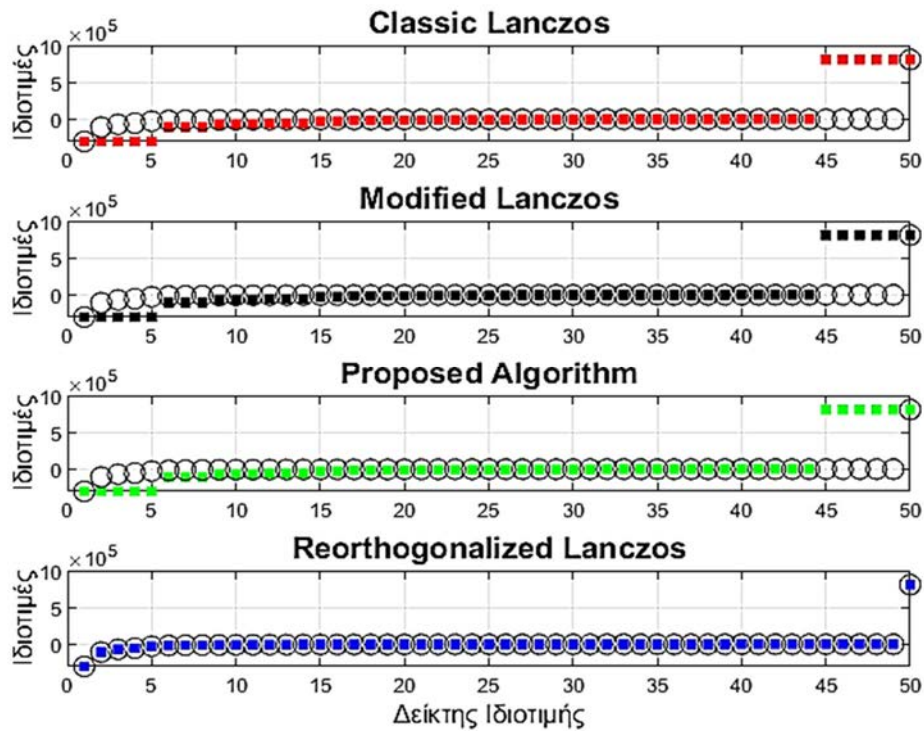
Θεωρούμε τον 13226×13226 πίνακα αποστάσεων A του Παραδείγματος 4.2, ο οποίος δημιουργήθηκε από τα δεδομένα του ιού Ζίκα (ZIKV-5IZ7), [34]. Μέσω της υλοποίησης των Κωδίκων 2.8, 2.10, 3.3 και 3.15, βλέπε στις γραμμές 78-209 του Παραρτήματος II, υπολογίζονται και απεικονίζονται οι ιδιοτιμές χ_T των τριδιαγώνιων πινάκων T_m , για m πλήθους ιδιοτιμές χ_A του A , όταν $m = 10$, $m = 50$, και $m = 100$. Μετά την υλοποίηση του καθενός από τους προαναφερθέντες Κώδικες από την (4.1) επιπρόσθετα υπολογίζονται τα σφάλματα των ακραίων ιδιοτιμών των τριδιαγώνιων πινάκων T_m και A (δηλαδή απόλυτη διαφορά φασματικών ακτίνων και απόλυτη διαφορά των ελαχίστων ιδιοτιμών των αντίστοιχων πινάκων), βλέπε στις γραμμές 210-228 του Παραρτήματος II. Η διαδικασία υπολογισμού των σφαλμάτων των ακραίων ιδιοτιμών επαναλαμβάνεται δέκα φορές και ο μέσος όρος των σφαλμάτων, που υπολογίζεται για κάθε τιμή της παραμέτρου m , αποτυπώνεται στον Πίνακα 4.28.

Οι αναπαραστάσεις των ιδιοτιμών γίνονται με

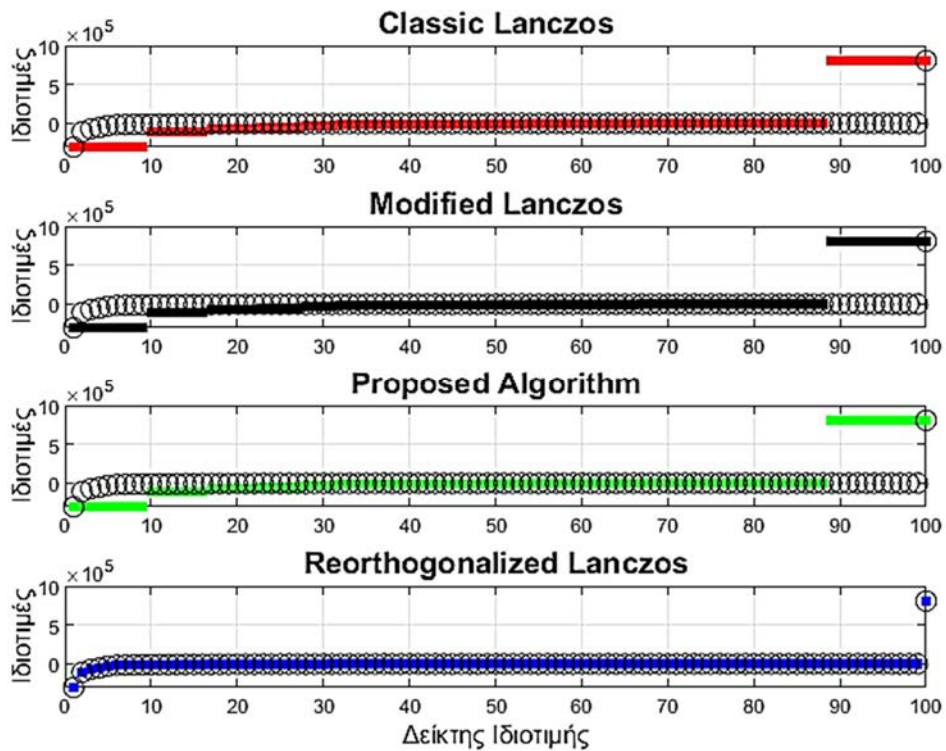
- ο τα **διαγράμματα σημείων (Scatter Plots)** στα Σχήματα 4.22-4.24. Σε αυτά, οι m πλήθους ιδιοτιμές χ_A του πίνακα αποστάσεων A αποτυπώνονται με μαύρους κύκλους σε όλες τις υλοποιήσεις και οι ιδιοτιμές χ_T σημειώνονται με έγχρωμα τετράγωνα, ανάλογα με την υλοποίηση: κόκκινο (Κώδικας 3.3-Classic Algorithm Lanczos), μαύρο (Κώδικας 2.8-Modified Lanczos Algorithm), πράσινο (Κώδικας 3.15-Proposed Algorithm) και μπλε (Κώδικας 2.10-Reorthogonalization Lanczos Algorithm).
- ο τα **ραβδογράμματα (Bar Plots)** στα Σχήματα 4.25-4.27. Σε αυτά, το ύψος κάθε ράβδου αντιστοιχεί στην τιμή της απόλυτης διαφοράς των αντίστοιχων ιδιοτιμών, χ_T, χ_A .
- ο **Διαγράμματα Σημείων (Scatter Plots)**



Σχήμα 4.22: Οι ιδιοτιμές χ_T του T_{10} με έγχρωμα τετράγωνα και με κύκλους οι ιδιοτιμές χ_A του A για τον ιό ZIKV-5IZ7.



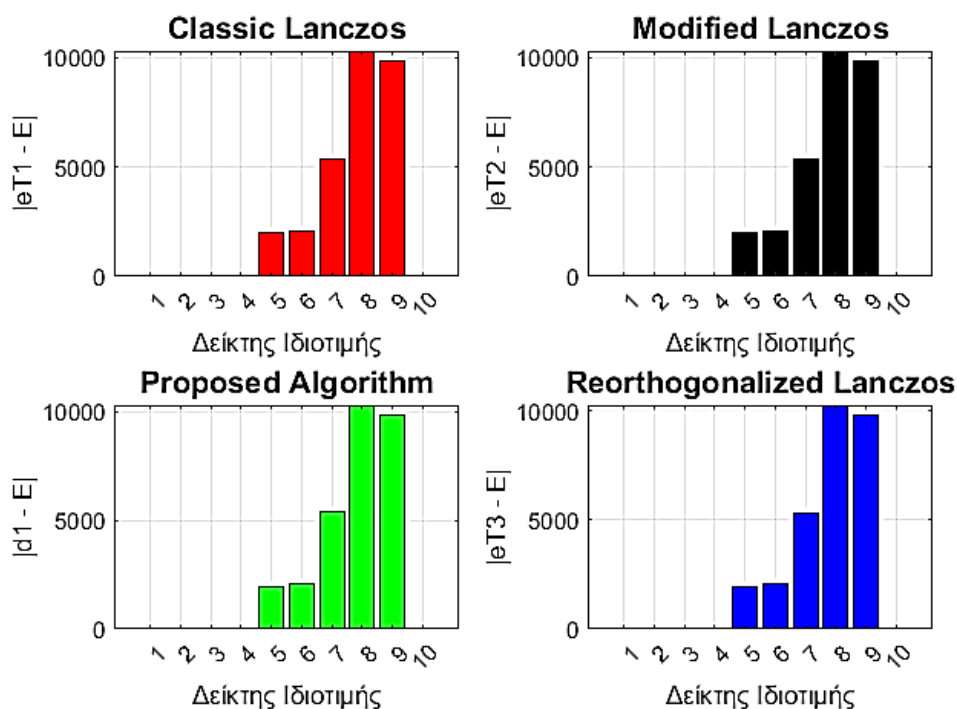
Σχήμα 4.23: Οι ιδιοτιμές χ_T του T_{50} με έγχρωμα τετράγωνα και με κύκλους οι ιδιοτιμές χ_A του A για τον ιό ZIKV-51Z7.



Σχήμα 4.24: Οι ιδιοτιμές χ_T του T_{100} με έγχρωμα τετράγωνα και με κύκλους οι ιδιοτιμές χ_A του A για τον ιό ZIKV-51Z7.

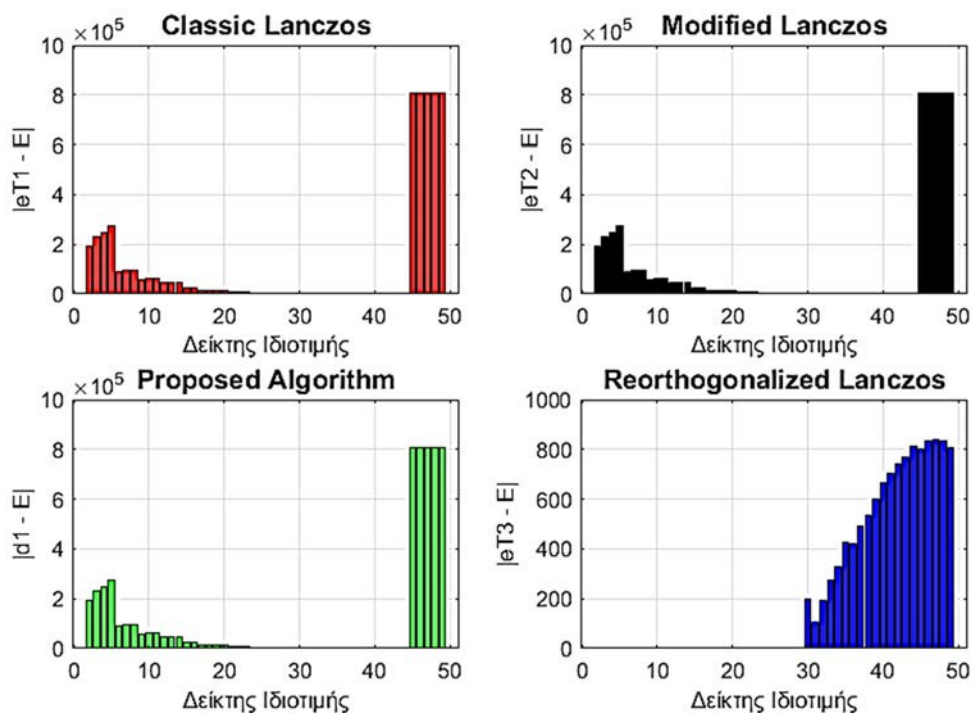
ο Ραβδογράμματα (Bar Plots)

Σύγκριση Σφαλμάτων



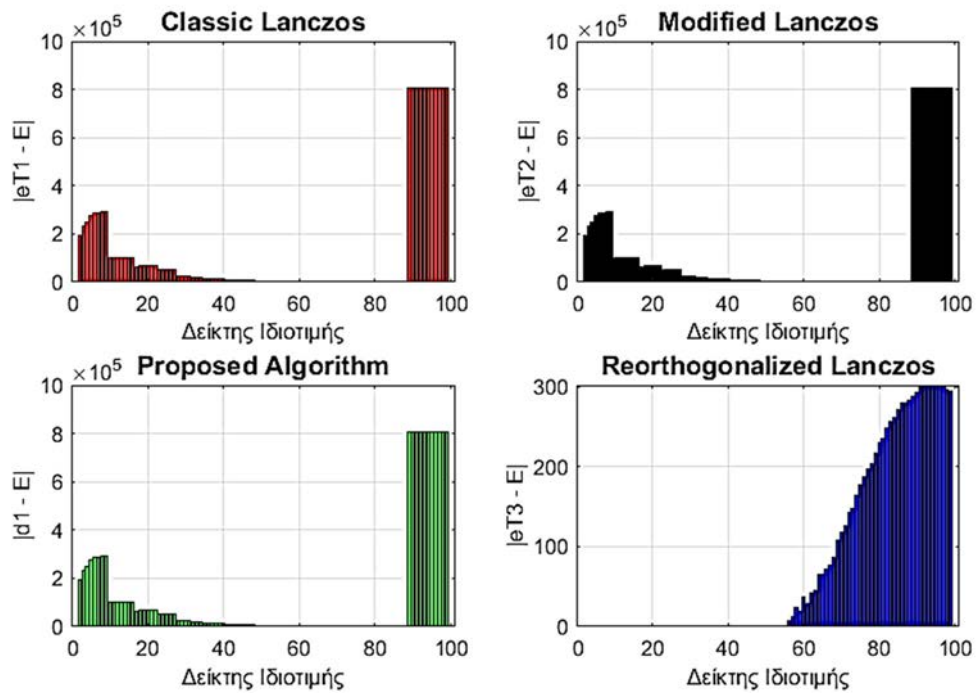
Σχήμα 4.25: Η απόλυτη διαφορά των m ιδιοτιμών με $m = 10$ των δεδομένων ZIKV-51Z7.

Σύγκριση Σφαλμάτων



Σχήμα 4.26: Η απόλυτη διαφορά των m ιδιοτιμών με $m = 50$ των δεδομένων ZIKV-51Z7.

Σύγκριση Σφαλμάτων



Σχήμα 4.27: Η απόλυτη διαφορά των m ιδιοτιμών με $m = 100$ των δεδομένων ZIKV-5IZ7.

Πίνακας 4.28: Σφάλματα ακραίων ιδιοτιμών των πινάκων T_m και A από δεδομένα ZIKV-5IZ7.

Αλγόριθμοι	Φασματική ακτίνα			Ελάχιστη ιδιοτιμή		
	$m = 10$	$m = 50$	$m = 100$	$m = 10$	$m = 50$	$m = 100$
Classic-Lanczos	$1.6 \cdot 10^{-9}$	$1.2 \cdot 10^{-10}$	$2.1 \cdot 10^{-9}$	$8.1 \cdot 10^{-10}$	$5.2 \cdot 10^{-10}$	$3.5 \cdot 10^{-10}$
Modified-Lanczos	$1.6 \cdot 10^{-9}$	$1.2 \cdot 10^{-10}$	$2.1 \cdot 10^{-9}$	$8.1 \cdot 10^{-10}$	$5.2 \cdot 10^{-10}$	$3.5 \cdot 10^{-10}$
Reorthogonalization-Lanczos	$1.5 \cdot 10^{-9}$	$1.2 \cdot 10^{-10}$	$2.3 \cdot 10^{-10}$	$7.6 \cdot 10^{-10}$	$5.8 \cdot 10^{-11}$	$1.7 \cdot 10^{-10}$
Proposed	$1.5 \cdot 10^{-9}$	$1.1 \cdot 10^{-10}$	$1.9 \cdot 10^{-10}$	$6.9 \cdot 10^{-10}$	$5.2 \cdot 10^{-11}$	$1.7 \cdot 10^{-10}$

Σχόλια 4.10

1. Παρατηρώντας όλα τα διαγράμματα σημείων και τα ραβδογράμματα των Παραδειγμάτων 4.8 και 4.9, τα οποία σχεδιάστηκαν στα Σχήματα 4.7-4.20 και 4.22-4.27 συμπεραίνουμε ότι ανεξάρτητα από τη διάσταση του πίνακα αποστάσεων A και για οποιαδήποτε τιμή της παραμέτρου $m \geq 5$, ο υπολογισμός της **φασματικής ακτίνας**, μέσω της υλοποίησης καθενός από τους τέσσερις αλγόριθμους (βλέπε, Κώδικα 2.8-Modified Lanczos Algorithm, Κώδικα 2.10-Reorthogonalization Lanczos Algorithm, Κώδικα 3.3-Classic Algorithm Lanczos, Κώδικα 2.8- και Κώδικα 3.15-Proposed Algorithm), είναι πολύ ακριβής, επειδή το αντίστοιχο σφάλμα, που υπολογίζεται από την (4.1), "φαίνεται ότι είναι μηδενικό". Στα ραβδογράμματα, (βλέπε, Σχήματα 4.14-4.20 και 4.25-4.27) παρατηρούμε ότι δεν υπάρχει ράβδος για τη μεγαλύτερη προς τα δεξιά ιδιοτιμή, άρα το σφάλμα της φασματικής ακτίνας είναι μηδενικό, στα δε διαγράμματα σημείων (βλέπε, Σχήματα 4.7-4.13 και 4.22-4.24) παρατηρούμε ότι το τελευταίο έγχρωμο τετράγωνο, το οποίο σημειώνει τη φασματική ακτίνα του τριδιαγώνιου πίνακα T_m , εσωκλείεται σε κύκλο, ο οποίος σημειώνει τη φασματική ακτίνα του A . Μελετώντας διεξοδικότερα τις τιμές των σφαλμάτων της φασματικής ακτίνας, που εμφανίζονται στους Πίνακες 4.21 και 4.28, γίνεται φανερό ότι :
 - i. Για μικρές τιμές της παραμέτρου m , δηλαδή για $5 \leq m < 10$, από τους τέσσερις αλγόριθμους, η προσέγγιση της φασματικής ακτίνας του A επιτυγχάνεται με σφάλμα τάξης μεγέθους 10^{-7} , (βλέπε, Πίνακα 4.21).
 - ii. Για $m \geq 10$ και μέσω της υλοποίησης των τεσσάρων αλγορίθμων η φασματική ακτίνα του A προσεγγίζεται από τη φασματική ακτίνα του T_m με πολύ μεγάλη ακρίβεια, που έχει τάξη μεγέθους τουλάχιστον 10^{-9} .
 - iii. Για $m \geq 10$, οι υλοποιήσεις του Proposed Αλγορίθμου (βλέπε, Κώδικα 3.15) και του Reorthogonalization Lanczos Αλγορίθμου (βλέπε, Κώδικα 2.10) υπολογίζουν τη βέλτιστη προσέγγιση της φασματικής ακτίνας του A , επειδή αυτοί οι αλγόριθμοι επιτυγχάνουν μικρότερης τάξης σφάλματα συγκριτικά με τις υλοποιήσεις των άλλων δύο αλγορίθμων.
Σε ορισμένες περιπτώσεις, η προσέγγιση της φασματικής ακτίνας μπορεί να παρουσιάζει σφάλμα της τάξης 10^{-11} , ο δε προτεινόμενος Αλγόριθμος μπορεί να παρουσιάζει ακόμη και μηδενικό σφάλμα, όπως συμβαίνει στις δύο περιπτώσεις, που εμφανίζονται στον Πίνακα 4.21, όπου $m = 20$, και $m = 50$.
2. Παρατηρώντας όλα τα διαγράμματα σημείων και τα ραβδογράμματα των Παραδειγμάτων 4.8 και 4.9, τα οποία σχεδιάστηκαν στα Σχήματα 4.7-4.20 και 4.22-4.27 συμπεραίνουμε ότι για οποιαδήποτε τιμή της παραμέτρου $m \geq 5$, ο υπολογισμός της **ελάχιστης ιδιοτιμής**, μέσω της υλοποίησης καθενός από τους τέσσερις Κώδικες, "φαίνεται ότι είναι ακριβής", επειδή το αντίστοιχο σφάλμα "φαίνεται ότι είναι μηδενικό". Στα ραβδογράμματα, (βλέπε, Σχήματα 4.14-4.20 και 4.25-4.27) παρατηρούμε ότι δεν υπάρχει ράβδος για τη μικρότερη προς τα αριστερά ιδιοτιμή, άρα το σφάλμα της ελάχιστης ιδιοτιμής είναι μηδενικό, στα δε διαγράμματα σημείων (βλέπε, Σχήματα 4.7-4.13 και 4.22-4.24) παρατηρούμε ότι το πρώτο έγχρωμο τετράγωνο, το οποίο σημειώνει την ελάχιστη ιδιοτιμή του T_m , εσωκλείεται σε κύκλο,

ο οποίος σημειώνει την ελάχιστη ιδιοτιμή του A . Ωστόσο μελετώντας τις τιμές των σφαλμάτων της ελάχιστης ιδιοτιμής, που εμφανίζονται στους Πίνακες 4.21 και 4.28, γίνεται φανερό ότι :

- i. Για μικρές τιμές της παραμέτρου m , δηλαδή για $5 \leq m < 10$, από τους τέσσερις αλγόριθμους, η ελάχιστη ιδιοτιμή του T_m , που υπολογίζεται από καθένα από τους τέσσερις Κώδικες, **δεν** προσεγγίζει την ελάχιστη ιδιοτιμή του A , επειδή το σφάλμα έχει τάξη μεγέθους 10^2 (βλέπε, στην πρώτη στήλη του Πίνακα 4.21). Σύμφωνα με τα Σχόλια 3.12(vii) για $m = 5$ και για τα δεδομένα του Παραδείγματος 3.11, το σφάλμα προσέγγισης της ελάχιστης ιδιοτιμής είναι μη μηδενικό, επειδή η ελάχιστη ιδιοτιμή του πίνακα αποστάσεων A είναι $\tilde{\ell}_1 = -1.2833 \cdot 10^5$ (βλέπε, Σχόλια 3.12(vi)) και η ελάχιστη ιδιοτιμή του T_5 είναι $\tilde{\lambda}_1 = -1.2819 \cdot 10^5$, γεγονός που επαληθεύει τα προαναφερόμενα.
 - ii. Για $m \geq 10$, ανεξάρτητα από τη διάσταση του πίνακα A και μέσω της υλοποίησης των τεσσάρων αλγορίθμων, η ελάχιστη ιδιοτιμή του A προσεγγίζεται από τη φασματική ακτίνα του T_m με μεγάλη ακρίβεια, που έχει τάξη μεγέθους τουλάχιστον 10^{-10} . Η ακρίβεια της προσέγγισης αυξάνει καθώς αυξάνει η τιμή της παραμέτρου m .
 - iii. Για $m \geq 10$, με τις υλοποιήσεις του Proposed Αλγορίθμου (βλέπε, Κώδικα 3.15) και του Reorthogonalization Lanczos Αλγορίθμου (βλέπε, Κώδικα 2.10) επιτυγχάνεται η βέλτιστη προσέγγιση της ελάχιστης ιδιοτιμής του A σχετικά με τις υλοποιήσεις των άλλων δύο αλγορίθμων, επειδή αυτοί οι αλγόριθμοι επιτυγχάνουν μικρότερης τάξης σφάλματα.
3. Επιπρόσθετα, εκτός από την προσέγγιση των ακραίων ιδιοτιμών (φασματική ακτίνα και ελάχιστη ιδιοτιμή) μπορούμε να εξάγουμε και ορισμένα συμπεράσματα για τις προσεγγίσεις των **ενδιάμεσων ιδιοτιμών**. Μελετώντας
- ο τα διαγράμματα σημείων**, που σχεδιάστηκαν στα Σχήματα 4.7-4.13 και 4.22-4.24, παρατηρούμε ότι:
- i. Για μικρές τιμές της παραμέτρου m , δηλαδή για $5 \leq m \leq 10$, τα αποτελέσματα των υλοποιήσεων με τη χρήση των τεσσάρων Κωδίκων "φαίνεται ότι δίνουν τις προσεγγίσεις των ενδιάμεσων ιδιοτιμών ακριβείς", επειδή τα σφάλματα "φαίνονται ότι είναι μηδενικά" μια και τα έγχρωμα τετράγωνα, τα οποία σημειώνουν τις ιδιοτιμές του T_m , εσωκλείονται στους κύκλους, οι οποίοι σημειώνουν τις ιδιοτιμές του A , παρατηρήστε στα Σχήματα 4.7, 4.8 και 4.22. Τα προαναφερόμενα επαληθεύονται από τις ιδιοτιμές χ_T του T_5 , που υπολογίστηκαν στο Παράδειγμα 3.11 με την υλοποίηση του Κώδικα 3.3-Classic Algorithm Lanczos και από τις πέντε ιδιοτιμές χ_A του A , που υπολογίστηκαν στα Σχόλια 3.12(vi) και τα σφάλματά τους σχολιάστηκαν στα Σχόλια 3.12(vii).
 - ii. Για $m > 10$, με την υλοποίηση του Reorthogonalization Lanczos Αλγορίθμου (βλέπε, Κώδικα 2.10), για κάθε τιμή της παραμέτρου m το αριθμητικό σφάλμα που υπολογίζεται από την (4.1) για κάθε ενδιάμεση ιδιοτιμή "φαίνεται ότι είναι μηδενικό" μια και τα έγχρωμα τετράγωνα (ιδιοτιμές του T_m) εσωκλείονται στους κύκλους (ιδιοτιμές του A), παρατηρήστε στα Σχήματα 4.9-4.13 και 4.23-4.24.

- iii. Για $m > 10$, οι υλοποιήσεις με τον Κώδικα 3.3-Classic Algorithm Lanczos, Κώδικα 2.8-Modified Lanczos Algorithm και Κώδικα 3.15-Proposed Algorithm "φαίνονται ότι παρουσιάζουν σφάλματα" για ένα μικρό πλήθος ενδιάμεσων ιδιοτιμών κοντά στην ελάχιστη ιδιοτιμή και στη φασματική ακτίνα και **δεν** παρουσιάζουν σφάλματα σε όλες τις υπόλοιπες ενδιάμεσες ιδιοτιμές. Το πλήθος των ενδιάμεσων ιδιοτιμών που παρουσιάζουν σφάλματα αυξάνεται καθώς αυξάνεται η τιμή της παραμέτρου m , παρατηρείστε στα Σχήματα 4.9-4.13 και 4.23-4.24.

Συνεπώς, ανεξάρτητα από τη διάσταση του πίνακα αποστάσεων A και την τιμή της παραμέτρου m , η υλοποίηση του Reorthogonalization Lanczos Αλγορίθμου (βλέπε, Κώδικα 2.10) δίνει τις πιο ακριβείς προσεγγίσεις των ενδιάμεσων ιδιοτιμών του A , επειδή τα αντίστοιχα σφάλματα "φαίνονται ότι είναι μηδενικά".

- ο Μελετώντας τα **ραβδογράμματα**, που σχεδιάστηκαν στα Σχήματα 4.14-4.20 και 4.25-4.27, αποκτούμε ακριβέστερη εκτίμηση των "μηδενικών σφαλμάτων των ενδιάμεσων ιδιοτιμών" που προσδιορίστηκαν από τα διαγράμματα σημείων. Συγκεκριμένα, από τα ραβδογράμματα παρατηρούμε ότι ανεξάρτητα από τη διάσταση του πίνακα αποστάσεων A :

- iv. Για μικρές τιμές της παραμέτρου m , δηλαδή για $5 \leq m \leq 10$, οι υλοποιήσεις των τεσσάρων Κωδίκων δίνουν τα ίδια σφάλματα των ενδιάμεσων ιδιοτιμών. Συγκεκριμένα, στις ιδιοτιμές¹⁶, που το ακέραιο μέρος του δείκτη τους ανήκει στο διάστημα $\left[2, \frac{m}{2} - 1\right]$ τα σφάλματα "φαίνονται ότι είναι μηδενικά" και στις υπόλοιπες ιδιοτιμές (εκτός από τη φασματική ακτίνα) εμφανίζονται σφάλματα, (βλέπε, Σχήματα 4.14, 4.15 και 4.25). Τα προαναφερόμενα επαληθεύονται από τις ενδιάμεσες ιδιοτιμές χ_T του T_5 , που υπολογίστηκαν στο Παράδειγμα 3.11 με την υλοποίηση του Κώδικα 3.3-Classic Algorithm Lanczos και από τις πέντε ιδιοτιμές χ_A του A , που υπολογίστηκαν στα Σχόλια 3.12(vi) και τα σφάλματά τους σχολιάστηκαν στα Σχόλια 3.12(vii).
- v. Για $m > 10$, με την υλοποίηση του Reorthogonalization Lanczos Αλγορίθμου (βλέπε, Κώδικα 2.10), το αριθμητικό σφάλμα της (4.1) "φαίνεται ότι είναι μηδενικό" **μόνο** για τις ιδιοτιμές, που το ακέραιο μέρος του δείκτη τους ανήκει στο διάστημα $\left[2, \frac{m}{2}\right]$, και αποτυπώνεται μη μηδενικό σφάλμα σε όλες τις ενδιάμεσες ιδιοτιμές, που το ακέραιο μέρος του δείκτη ανήκει στο διάστημα $\left[\frac{m}{2} + 1, m - 1\right]$.

Οι υλοποιήσεις με τους υπόλοιπους Κώδικες (Κώδικα 3.3-Classic Algorithm Lanczos, Κώδικα 2.8-Modified Lanczos Algorithm και Κώδικα 3.15-Proposed Algorithm) "εμφανίζουν μηδενικό σφάλμα" εκτός από τις ακραίες ιδιοτιμές

¹⁶ Υπενθυμίζεται ότι :

- i) Οι ιδιοτιμές των συμμετρικών πινάκων $A \in \mathbb{R}^{n \times n}$ είναι πραγματικοί αριθμοί, συνεπώς μπορούν να διαταχθούν σε αύξουσα σειρά, έστω $\lambda_1 \leq \lambda_2 \leq \dots \leq \lambda_n$.
- ii) Οι πίνακες T_m, A είναι συμμετρικοί, επομένως οι ιδιοτιμές τους διατάσσονται με ανάλογο τρόπο, όπως στο (i).

καθώς και στις ενδιάμεσες ιδιοτιμές, στις οποίες το ακέραιο μέρος του δείκτη ανήκει στο διάστημα $\left(\frac{m}{2} + 1, \frac{7m}{8}\right)$.

- vi. Οι τιμές των αριθμητικών σφαλμάτων, που σχεδιάστηκαν στα ραβδογράμματα από την υλοποίηση του Κώδικα 3.3-Classic Algorithm Lanczos, του Κώδικα 2.8-Modified Lanczos Algorithm και του Κώδικα 3.15-Proposed Algorithm, είναι της τάξης μεγέθους του 10^5 , που σημαίνει ότι η διαφορά (κατά απόλυτη τιμή) μεταξύ των ιδιοτιμών χ_T και χ_A είναι τάξης μεγέθους των ιδιοτιμών, γεγονός που δεν ισχύει με την υλοποίηση του Reorthogonalization Lanczos Αλγορίθμου, όπως επαληθεύεται από το ακόλουθο παράδειγμα. Θεωρώντας $m = 20$, η υλοποίηση του Κώδικα 2.10 υπολογίζει τις ιδιοτιμές του T_{20} , οι οποίες δίνονται σε αύξουσα σειρά ως εξής:

$$\lambda_1 \leq \lambda_2 \leq \dots \leq \lambda_{18} = -0.00045 \cdot 10^5 \leq \lambda_{19} \leq \lambda_{20} = 3.0867 \cdot 10^5$$

Η υλοποίηση του Κώδικα 3.3 υπολογίζει τις ιδιοτιμές του T_{20} ως εξής:

$$\hat{\lambda}_1 \leq \hat{\lambda}_2 \leq \dots \leq \hat{\lambda}_{18} = 2.87596 \cdot 10^5 \leq \hat{\lambda}_{19} \leq \hat{\lambda}_{20} = \lambda_{20} = 3.0867 \cdot 10^5$$

Οι ιδιοτιμές του A χρησιμοποιώντας την "eigs" υπολογίζονται ως εξής:

$$\ell_1 \leq \ell_2 \leq \dots \leq \ell_{18} = -0.01226 \cdot 10^5 \leq \ell_{19} < \ell_{20} = 3.0867 \cdot 10^5.$$

Προφανώς, τα αντίστοιχα σφάλματα, τα οποία σκιαγραφούνται στο Σχήμα 4.16 (βλέπε, Reorthogonalized Lanczos και Classic Lanczos), είναι

$$|\ell_{18} - \lambda_{18}| = 0.01181 \cdot 10^5 \cong 1200, \quad \text{και} \quad |\ell_{18} - \hat{\lambda}_{18}| = 2.88822 \cdot 10^5,$$

τα οποία επαληθεύουν το παραπάνω συμπέρασμα.

Συνεπώς, ο Reorthogonalization Lanczos Αλγόριθμος "φαίνεται" ότι υπολογίζει μικρότερης τάξης σφάλματα των ενδιάμεσων ιδιοτιμών συγκριτικά με τα σφάλματα από τις υλοποιήσεις των υπολοίπων αλγορίθμων (βλέπε, παραπάνω στο (vi)).

Συμπεράσματα

Στην παρούσα διπλωματική εργασία ασχοληθήκαμε με την ανάπτυξη και υλοποίηση αλγορίθμων για την μελέτη προβλημάτων φασματικής ανάλυσης πινάκων αποστάσεων, που δημιουργούνται από τις ατομικές συντεταγμένες μοριακών δεδομένων, οι οποίες είναι καταχωρημένες σε πίνακες μεγάλων διαστάσεων.

Η συμμετρικότητα του πίνακα αποστάσεων και η αξιοποίηση των πλεονεκτημάτων της Μεθόδου των Δυνάμεων και του Κλασσικού Αλγορίθμου Lanczos, οδήγησαν στην ανάπτυξη ενός αλγορίθμου (Proposed Algorithm), ο οποίος συμβάλλει αποτελεσματικά στον υπολογισμό των ιδιοτιμών συμμετρικών τριδιαγώνιων πινάκων ακόμη και μεγάλης διάστασης.

Η μελέτη των ιδιοτιμών και ιδιοδιανυσμάτων των τριδιαγώνιων πινάκων που προέκυψαν από την υλοποίηση των τεσσάρων αλγορίθμων (Classic Lanczos Algorithm, Modified Lanczos Algorithm, Reorthogonalization Lanczos Algorithm, Proposed Algorithm) οδήγησε σε μια σειρά από χρήσιμα και σημαντικά συμπεράσματα για την προσέγγιση των ιδιοποσών πινάκων αποστάσεων μεγάλης διάστασης, τα οποία συμβάλλουν καθοριστικά στην κατανόηση τόσο της γεωμετρικής δομής της πρωτεΐνης όσο και της αποδοτικότητας των επιμέρους υπολογιστικών προσεγγίσεων. Σε επίπεδο υπολογιστικής πολυπλοκότητας, το πλήθος m των ιδιοτιμών που επιθυμούμε να υπολογίσουμε επηρεάζει σημαντικά τον χρόνο εκτέλεσης, καθώς η υπολογιστική πολυπλοκότητα των υλοποιήσεων εξαρτάται από το m και τη διάσταση n του πίνακα αποστάσεων.

Από τη σύγκριση τεσσάρων υλοποιήσεων του Κλασσικού Αλγορίθμου Lanczos, των δύο βελτιωμένων εκδόσεών του Lanczos (Modified Lanczos Algorithm και Reorthogonalization Lanczos Algorithm) και του νέου αλγορίθμου που αναπτύχθηκε στο πλαίσιο της παρούσας διπλωματικής εργασίας, προκύπτουν σημαντικές διαφορές τόσο στην απόδοση όσο και στην ακρίβεια, οι οποίες συνοψίζονται στη συνέχεια:

- ο προτεινόμενος Αλγόριθμος εμφανίζει τον μικρότερο χρόνο εκτέλεσης, γεγονός που τον καθιστά ιδανικό για προβλήματα φασματικής ανάλυσης ενός συμμετρικού πίνακα μεγάλης διάστασης, στα οποία καθοριστικό ρόλο για την επίλυσή τους παίζει η ταχύτητα απόκρισης.
- Ανεξάρτητα από την τιμή της παραμέτρου m , τη διάσταση του πίνακα αποστάσεων A και από την επιλογή του αλγορίθμου, η φασματική ακτίνα του πίνακα αποστάσεων A προσεγγίζεται με πολύ μεγάλη ακρίβεια από τη φασματική ακτίνα του τριδιαγώνιου πίνακα T_m , επειδή το σφάλμα της προσέγγισής της έχει τάξη μεγέθους τουλάχιστον 10^{-7} , βλέπε στα Σχόλια 4.7(iii) και Σχόλια 4.10 στο 1(i)-(ii).
- Η φασματική ακτίνα ερμηνεύεται ως η κύρια συνιστώσα της πρωτεΐνης, αναδεικνύοντας την κυρίαρχη κατεύθυνση κατανομής των ατομικών συντεταγμένων της στον τρισδιάστατο χώρο. Το αντίστοιχο ιδιοδιάνυσμά της παίζει κομβικό ρόλο, καθώς ερμηνεύει τη μεγαλύτερη διακύμανση στα δεδομένα αποτελώντας τη σημαντικότερη πληροφορία για την περιγραφή της συνολικής γεωμετρίας ενός μακρομορίου, όπως είναι μια πρωτεΐνη.

- Μόνο όταν $m \geq 10$, η ελάχιστη ιδιοτιμή του A προσεγγίζεται με πολύ μεγάλη ακρίβεια από την ελάχιστη ιδιοτιμή του τριδιαγώνιου πίνακα T_m , επειδή το σφάλμα της προσέγγισής της έχει τάξη μεγέθους τουλάχιστον 10^{-10} , βλέπε στα Σχόλια 4.7(iii) και στα Σχόλια 4.10 στο 2(i)-(ii).
- Όταν $m \geq 10$, η βέλτιστη προσέγγιση των ακραίων ιδιοτιμών (φασματικής ακτίνας και ελάχιστης ιδιοτιμής) του A επιτυγχάνεται μετά τις υλοποιήσεις του Proposed Algorithm και του Reorthogonalization Lanczos Algorithm, επειδή το σφάλμα της προσέγγισής της έχει τάξη μεγέθους τουλάχιστον 10^{-10} , βλέπε αναλυτικά στα Σχόλια 4.10 στο 1(iii) και 2(iii).
- Μέσω των αναπαραστάσεων των διαγραμμάτων σημείων και των ραβδογραμμάτων αποδεικνύεται ότι ανεξάρτητα από τη διάσταση του πίνακα αποστάσεων A , οι ενδιάμεσες ιδιοτιμές του A παρουσιάζουν σημαντικό αριθμητικό σφάλμα στην προσέγγισή τους από τις αντίστοιχες ιδιοτιμές του τριδιαγώνιου πίνακα T_m , για οποιαδήποτε τιμή της παραμέτρου m , βλέπε στα Σχόλια 4.7(iii) και στα Σχόλια 4.10 στο 3. Το σφάλμα αυτό συχνά είναι της ίδιας τάξης μεγέθους με την τιμή της εκάστοτε ιδιοτιμής, γεγονός που καθιστά την ερμηνεία αυτών των ιδιοτιμών μη αξιόπιστη, βλέπε αναλυτικά στα Σχόλια 4.10 στο 3(iv)-(vi).

Συνοψίζοντας, ο υπολογισμός της φασματικής ακτίνας και της ελάχιστης ιδιοτιμής του πίνακα αποστάσεων αποτελεί τη σταθερότερη και πιο ακριβή πληροφορία που μπορούμε να εξάγουμε από τη μελέτη των ιδιοποσών των τριδιαγώνιων πινάκων, που κατασκευάζονται από τους τέσσερις αλγορίθμους Lanczos, οι οποίοι μελετήθηκαν στην παρούσα διπλωματική εργασία. Η επιλογή του καταλληλότερου αλγορίθμου και της παραμέτρου m εξαρτάται από το εκάστοτε ζητούμενο: είτε η ταχύτητα απόκρισης, είτε η ακρίβεια, είτε η ανάκτηση λεπτομερειών της δομής. Η παρούσα εργασία δεν περιορίστηκε στην απλή εφαρμογή τεχνικών, αλλά διαμόρφωσε ένα συστηματικό και αξιόπιστο μεθοδολογικό πλαίσιο, ικανό να αξιοποιηθεί για τη φασματική ανάλυση μοριακών συστημάτων με δεδομένα που τοποθετούνται σε πίνακες μεγάλων διαστάσεων. Η αξιοποίηση της φασματικής θεωρίας των μη αρνητικών πινάκων και ειδικά η μελέτη της φασματικής ακτίνας αναδείχθηκαν ως ισχυρά εργαλεία για την ανάδειξη βασικών χαρακτηριστικών των πρωτεϊνικών δομών. Μέσα από τη χρήση τριδιαγώνιων προσεγγίσεων, ιδιοτιμών και ιδιοδιανυσμάτων, παρουσιάστηκε ένας αποτελεσματικός τρόπος μελέτης της σταθερότητας και της δυναμικής της δομής, έχοντας υψηλή υπολογιστική αποδοτικότητα, δίνοντας το έναυσμα για μελλοντικές αναλύσεις παρόμοιων προβλημάτων.

Βιβλιογραφία

1. Andrilli, S., & Hecker, D. (2022). Elementary Linear Algebra (Vol. 6th edition). Academic Press. Retrieved from <https://www.sciencedirect.com/topics/mathematics/power-method>
2. Ascher Uri. M, & Greif Chen. (2011). A First Course in Numerical Methods (Vol. Computational Science and Engineering). SIAM.
3. Walter Edwin Arnoldi. The principle of minimizing iterations in the solution of the matrix eigenvalue problem. Quarterly of Applied Mathematics, 9(1): 17–29, 1951.
4. Ermann L, Frahm K, & Shepelyansky D. (2016). Google matrix. (Scholarpedia, Ed.) doi:10.4249/scholarpedia.30944
5. Ermann L, Frahm K. M, & Shepelyansky D.L. (2016). Google matrix analysis of directed networks. Reviews of Modern Physics, 88. doi:<https://doi.org/10.1103/RevModPhys.87.1261>
6. Gene H. Golub, & Dianne P. O’Leary. (Mar., 1989). Some History of the Conjugate Gradient and Lanczos Algorithms: 1948-1976. SIAM Review. Retrieved from <https://doi.org/10.1137/1031003>
7. Gene H. Golub, & Charles F. Van Loan. (2015). Θεωρία και υπολογισμοί μητρώων. (Α. Μπικουβαράκης, Μεταφρ.) Αθήνα.
8. Hernandez V, Roman J. E, Tomas A , & Vidal V. (2006). Lanczos Methods in SLEPc. Scalable Library for Eigenvalue Problem Computations. Retrieved from <https://slepc.upv.es/documentation/reports/str5.pdf>
9. Horn, R. A., & Johnson, C. R. (2013). Matrix Analysis (2nd ed.). Cambridge University Press.
10. Kloczkowski A., Jernigan R., Wu Z., Song G., Yang L., Kolinski A., & Pokarowski P. (2009, February 18). Distance matrix-based approach to protein structure prediction. Journal of Structural and Functional Genomics, p. 15. doi:10.1007/s10969-009-9062-2
11. Koehl P. (2018, Jun 20). Large Eigenvalue Problems in Coarse-Grained Dynamic Analyses of Supramolecular Systems. JCTC. Doi: 10.1021/acs.jctc.8b00338
12. Komzsik L. (2003). The Lanczos method: evolution and application. SIAM. Retrieved from <https://epubs.siam.org/doi/pdf/10.1137/1.9780898718188.bm>
13. Liao R., Zhao Z., Urtasun R., & Zemel R. (2019). LanczosNet: Multi-Scale Deep Graph Convolutional Networks. Conference paper at ICLR (pp. 1-17).
14. Meyer C.D. (2000), Matrix Analysis and Applied Linear Algebra, SIAM
15. Rajakumar C. (1993). Lanczos algorithm for the quadratic eigenvalue problem in engineering applications. Computer Methods in Applied Mechanics and Engineering.
16. Youcef Saad, Variations on Arnoldi’s Method for Computing Eigenelements of Large Unsymmetric Matrices, Linear Algebra and Applications, v. 34, 1980, p.269-295.
17. Youcef Saad. Numerical methods for large eigenvalue problems. SIAM, 2nd ed. 2011 https://www-users.cse.umn.edu/~saad/eig_book_2ndEd.pdf.
18. Sedeh R., Bathe M., & Bathe K. (2009). The subspace iteration method in protein normal mode analysis. NIH. doi:10.1002/jcc.21250
19. Symeonidis P., Iakovidou N., Mantas N., & Manolopoulos Y. (2013). From biological to social networks: Link prediction based on multi-way spectral clustering. Data & Knowledge Engineering.

20. Yang L., Liu X., Jursa C., Holliman M., Rader A., Karimi H., & Bahar I. (2005). i GNM: a database of protein functional motions based on Gaussian Network Model. *iscB*. doi:10.1093/bioinformatics/bti469
21. Αντωνέλλου Θ. (2016). Μέθοδοι Προβολής για την Αριθμητική Επίλυση. Διπλωματική Εργασία, τ. Μαθηματικών, Ε.Κ.Π.Α.
22. Γραββάνης Γ., & Γιαννουτάκης Κ. (2012). Προγραμματισμός με τη Χρήση MATLAB. Αθήνα: Παπασωτηρίου.
23. Δονάτος Γ., & Αδάμ Μ. (2008). Γραμμική Άλγεβρα θεωρία και εφαρμογές, Αθήνα. Αθήνα: Gutenberg.
24. Κωνσταντινίδης Κ. (2017). Ανάπτυξη λογισμικού φιλικού προς τον χρήστη για τον υπολογισμό αποκλειόμενου όγκου πρόσδεσης μικρών μορίων. Πτυχιακή Εργασία, τ. Μοριακή Βιολογία και Γενετική, Δημοκρίτειο Πανεπιστήμιο Θράκης.
25. Μητρούλη Μ. (2022). Αριθμητική Γραμμική Άλγεβρα και Επιστημονικοί Υπολογισμοί. Αθήνα: Ε.Κ.Π.Α.
26. Μισυρλής Ν. (2017). Αριθμητική Ανάλυση. (Μια αλγοριθμική προσέγγιση) (2η εκδ.). Αθήνα: Ε.Κ.Π.Α.
27. Παπαγεωργίου Γ. Σ, & Τσίτουρας Χ. Γρ. (2015). Αριθμητική Ανάλυση με εφαρμογές σε Mathematica και Matlab. Αθήνα: Τσότρας.
28. Παυλίδης Μ. (2022, Ιουνίου 15). GPU υλοποίηση του μπλόκ αλγόριθμου του Lanczos για τον υπολογισμό των μεγαλύτερων ιδιοτιμών και ιδιοδιανυσμάτων μεγάλων αραιών πινάκων. Διπλωματική Εργασία, τ. Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών, Α.Π.Θ.
29. Προύντσης Τ. (2022). Μέθοδοι υποχώρων Krylov για την επίλυση πολυωνυμικών προβλημάτων ιδιοτιμών. Διπλωματική Εργασία, τ. ΣΕΜΦΕ, Ε.Μ.Π.
30. Φαμέλης Ι. Θ.. (2021). Υπολογιστικά Μαθηματικά. Αθήνα: Κριτική.

Ιστοσελίδες

31. RCSB Protein Data Bank (RCSB PDB). Retrieved from: <http://www.rcsb.org/pdb/home/home.do>
32. 141L, Role of Backbone flexibility in the accommodation of variants that repack the core of t4 lysozyme. Retrieved from: <https://www.rcsb.org/structure/141L>
33. 3pl6, Structure of Autoimmune TCR Hy.1B11 in complex with HLA-DQ1 and MBP 85-99. Retrieved from: <https://www.rcsb.org/structure/3PL6>
34. 5IZ7, Cryo-EM structure of thermally stable Zika virus strain H/PF/2013. Retrieved from: <https://www.rcsb.org/structure/5IZ7>

Παράρτημα Ι

Εδώ παρουσιάζεται ο κύριος κώδικας υλοποίησης σε Matlab που αφορά τον Κλασσικό Αλγόριθμο Lanczos, όπως αυτός αναλύεται στο Κεφάλαιο 2. Ο αλγόριθμος αυτός εφαρμόζεται για την αναπαράσταση των ιδιοτιμών και των αντίστοιχων ιδιοδιανυσμάτων συμμετρικών πινάκων.

```
%% --- ΚΥΡΙΟΣ ΚΩΔΙΚΑΣ ---
```

```
% Ζήτηση PDB ID
```

```
1. pdbID = input('Enter the PDB ID (e.g., 3pl6): ', 's');
```

```
% Λήψη αρχείου PDB από το RCSB
```

```
2. url = ['https://files.rcsb.org/download/' pdbID  
        '.pdb'];
```

```
3. pdbFileName = [pdbID '.pdb'];
```

```
4. fprintf('Downloading PDB file for %s...\n', pdbID);
```

```
5. websave(pdbFileName, url);
```

```
% Ανάγνωση του αρχείου και εξαγωγή συντεταγμένων
```

```
6. pdbStruct = pdbread(pdbFileName);
```

```
7. atoms = pdbStruct.Model.Atom;
```

```
8. cord = [[atoms.X]; [atoms.Y]; [atoms.Z]]';
```

```
9. cord = cord(~any(isnan(cord), 2), :);
```

```
% Δημιουργία του πίνακα αποστάσεων
```

```
10. fprintf('Calculating distance matrix...\n');
```

```
11. distanceMatrix = squareform(pdist(cord));
```

```
12. % Ερώτηση για πρώτη τιμή m
```

```
13. m1 = input('Input the number of eigenvectors to compute  
for the first run (m1) greater than 2 = ');
```

```
14. if m1 < 3
```

```
15. error('The number of eigenvectors (m1) must be greater  
than or equal to 3.');
```

```
16. end
```

```
17. % Υπολογισμός Ιδιοτιμών του Πίνακα Αποστάσεων για m1
```

```
18. [V, E] = eigs(distanceMatrix, m1);
```

```
19. E = sort(diag(E));
```

```
20. fprintf('Eigenvalues of Distance matrix for m1 =  
%d...\n', m1);
```

```
21. disp(E);
```

```
22. % Εφαρμογή του Αλγορίθμου Lanczos για m1
```

```

23. fprintf('Running Lanczos algorithm for m1 = %d...\n',
    m1);
24. [T1,vT1,eT1] = Lanczos_classic(distanceMatrix, m1);
25. fprintf('Tridiagonal matrix T for m1 = %d...\n', m1);
26. disp(T1);
27. fprintf('Eigenvalues of Tridiagonal matrix for m1 =
    %d...\n', m1);
28. disp(eT1);
29. fprintf('Eigenvectors of Tridiagonal matrix for m1 =
    %d...\n', m1);
30. disp(vT1);

31. % Οπτικοποιήσεις για την πρώτη τιμή m1
32. visualize_all(cord, distanceMatrix, eT1, vT1, pdbID,
    m1);

33. % Ερώτηση για δεύτερη τιμή m
34. second_choice = input('Do you want to compute
    eigenvectors for a second value of m? (yes/no): ',
    's');

35. if strcmpi(second_choice, 'yes')
36. % Δεύτερη τιμή m
37. m2 = input('Input the number of eigenvectors to compute
    for the second run (m2) greater than 2 = ');
38. if m2 < 3
39. error('The number of eigenvectors (m2) must be greater
    than or equal to 3.');
```

```

40. end

41. % Υπολογισμός Ιδιοτιμών του Πίνακα Αποστάσεων για m2
42. [V, E] = eigs(A, m2);
43. E = sort(diag(E));
44. fprintf('Eigenvalues of Distance matrix for m2 =
    %d...\n', m2);
45. disp(E);
46. % Εφαρμογή του Αλγορίθμου Lanczos για m1
47. fprintf('Running Lanczos algorithm for m2 = %d...\n',
    m2);
48. [T2,vT2,eT2] = Lanczos_classic(distanceMatrix, m2);
49. fprintf('Tridiagonal matrix T for m2 = %d...\n', m2);
50. disp(T2);
51. fprintf('Eigenvalues of Tridiagonal matrix for m2 =
    %d...\n', m2);

```

```

52. disp(eT2);
53. fprintf('Eigenvectors of Tridiagonal matrix for m2 =
    %d...\n', m2);
54. disp(vT2);

55. % Οπτικοποιήσεις για τη δεύτερη τιμή m2
56. visualize_all(cord, distanceMatrix, eT2, vT2, pdbID,
    m2);

57. % Ερώτηση για σύγκριση
58. compare_choice = input('Do you want to compare the
    results of m1 and m2? (yes/no): ', 's');
59. if strcmpi(compare_choice, 'yes')
60. % Σύγκριση ιδιοτιμών
61. compare_eigenvalues(eT1, eT2, m1, m2);
62. end
63. else
64. disp('No second run requested. Proceeding with the
    first run only.');
```

```
%% --- ΣΥΝΑΡΤΗΣΕΙΣ ---
```

```
% Classic Algorithm Lanczos
```

```
66. function [vT, eT] = Lanczos_classic(A, m)
    n = size(A, 1);
    V = zeros(n, m);
    T = zeros(m, m);
    v1 = rand(n, 1);
    v1 = v1 / norm(v1);
    V(:, 1) = v1;
    beta = 0;
    v_prev = zeros(n, 1);

    for k = 1:m
        w = A * V(:, k) - beta * v_prev;
        alpha = V(:, k)' * w;
        w = w - alpha * V(:, k);

        beta = norm(w);

        T(k, k) = alpha;
        if k < m
            T(k, k+1) = beta;
            T(k+1, k) = beta;
        end

        if beta < 1e-10
            disp('Lanczos method stopped due to low
                beta. ');
            break;
        end

        v_prev = V(:, k);
        V(:, k+1) = w / beta;
    end
    [vT, eT_matrix] = eig(T);
    eT = diag(eT_matrix);
    [eT, idx] = sort(eT, 'ascend')
    vT = vT(:, idx);
67. end
```

```

% Συνάρτηση Οπτικοποίησης Ιδιοδιανυσμάτων
68. function analyze_eigenvectors(e, v, matrixName, m)
69. if m ~= 3
70. fprintf('3D visualization of eigenvectors is not
    possible for m = %d. Required m = 3.\n', m);
71. return;
72. end
73. numEigenvalues = min(length(e), 10);
74. figure;
75. hold on;
76. colors = lines(numEigenvalues); % Χρωματική
    διαφοροποίηση
77. for i = 1:numEigenvalues
78. vec = v(:, i) * sqrt(abs(e(i))); % Κλιμάκωση με βάση
    την ιδιοτιμή
79. quiver3(0, 0, 0, vec(1), vec(2), vec(3), 'LineWidth',
    2, ...
80. 'MaxHeadSize', 2, 'Color', colors(i, :));
81. text(vec(1), vec(2), vec(3), ['v' num2str(i)],
    'FontSize', 8);
82. end
83. hold off;
84. xlabel('X');
85. ylabel('Y');
86. zlabel('Z');
87. view(3); % Ρύθμιση τρισδιάστατης προβολής
88. title(['3D Visualization of ', matrixName, '
    Eigenvectors (m = ', num2str(m), ')]);
89. grid on;
90. end

% Οπτικοποιήσεις
91. function visualize_all(cord, distanceMatrix, eT, vT,
    pdbID, m)

% 3D Οπτικοποίηση Ατομικών Συντεταγμένων
92. figure;
93. scatter3(cord(:, 2), cord(:, 1), cord(:, 3), 20,
    cord(:, 3), 'filled');
94. colormap(jet);
95. colorbar;
96. xlabel('X-axis'); % Ετικέτα για τον X-άξονα
97. ylabel('Y-axis'); % Ετικέτα για τον Y-άξονα
98. zlabel('Z-axis');

```

```

99. title(['3D Visualization of Protein Atomic Coordinates
        for ', pdbID]);
100. grid on;
101. axis equal;
102. view(30, 30);

% Heatmap του Πίνακα Αποστάσεων
103. figure;
104. imagesc(distanceMatrix);
105. colorbar;
106. xlabel('Atom Index');
107. ylabel('Atom Index');
108. title(['Heatmap of Distance Matrix for ', pdbID]);
109. axis square;

% Οπτικοποίηση Ιδιοδιανυσμάτων
110. analyze_eigenvectors(eT, vT, ['Distance Matrix of ',
        pdbID], m);
111. end

% Σύγκριση Ιδιοτιμών
112. function compare_eigenvalues(eT1, eT2, m1, m2)
113. figure;
114. plot(1:length(eT1), eT1, 'o', 'MarkerSize', 8,
        'DisplayName', ['m1 = ', num2str(m1)]);
115. hold on;
116. plot(1:length(eT2), eT2, '*', 'MarkerSize', 8,
        'DisplayName', ['m2 = ', num2str(m2)]);
117. title('Comparison of Eigenvalues between m1 and m2',
        'FontSize', 14);
118. xlabel('Eigenvalue Number', 'FontSize', 12);
119. ylabel('Eigenvalue Price', 'FontSize', 12);
120. yline(0, '--', 'Zero Base', 'LineWidth', 1, 'Color',
        'k');
121. grid on;
122. legend('show', 'FontSize', 12, 'Location', 'Best');
123. hold off;
124. end

```


Παράρτημα II

Εδώ παρουσιάζονται οι κώδικες σε Matlab, που αφορούν τους αλγορίθμους του Κεφαλαίου 2, καθώς και ο προτεινόμενος αλγόριθμος που αναπτύχθηκε στο Κεφάλαιο 3.

```
%% --- Κύριος Κώδικας ---
```

```
% Ζήτηση PDB ID και αριθμού ιδιοτιμών από τον χρήστη
```

```
1. pdbID = input('Enter the PDB ID (e.g., 3pl6, 5IZ7): ',  
    's');  
2. m = input('Input the number of eigenvectors to compute  
    m = ');
```

```
% Λήψη και Ανάγνωση του αρχείου PDB από το RCSB
```

```
3. url = ['https://files.rcsb.org/download/' pdbID  
    '.pdb'];  
4. pdbFileName = [pdbID '.pdb'];  
5. fprintf('Downloading PDB file for %s...\n', pdbID);  
6. websave(pdbFileName, url);  
7.  
8. pdbStruct = pdbread(pdbFileName);
```

```
% Εξαγωγή των ατομικών συντεταγμένων
```

```
9. atoms = pdbStruct.Model.Atom;  
10. cord = [[atoms.X]; [atoms.Y]; [atoms.Z]]';  
11. cord = cord(~any(isnan(cord), 2), :); % Αφαίρεση μη  
    έγκυρων γραμμών
```

```
% Δημιουργία πίνακα αποστάσεων
```

```
12. fprintf('Calculating distance matrix...\n');  
13. distanceMatrix = squareform(pdist(cord));
```

```
% Παράμετροι
```

```
14. A = distanceMatrix;  
15. n = size(A, 1);  
16. % Πρώτο διάνυσμα Lanczos  
17. q = rand(n, 1);  
18. q = q / norm(q);  
19. tol = eps; % Ανοχή για σύγκλιση
```

```
% Υπολογισμός ιδιοτιμών του πίνακα αποστάσεων
```

```
20. [V, E] = eigs(A, m);  
21. E = sort(diag(E));
```

```

% Εκτέλεση των αλγορίθμων Lanczos και Proposed
22. algorithms = {'Classic Lanczos', 'Modified
    Lanczos', 'Proposed Algorithm', 'Reorthogonalized
    Lanczos'};
23. errors = cell(1, 3); % Προετοιμασία για αποθήκευση των
    σφαλμάτων για κάθε αλγόριθμο
24. times = zeros(1, 4); % Χρόνοι εκτέλεσης
25. d_vals = cell(1, 3); % Τιμές από Proposed Algorithm
% Classic Lanczos
26. f1 = @() Lanczos_classic(A, q, m);
27. times(1) = timeit(f1);
28. fprintf('Χρόνος εκτέλεσης Classic Lanczos: %.6f
    δευτερόλεπτα\n', times(1));
29. [T1, vT1, eT1, eT1s] = Lanczos_classic(A, q, m);
30. errors{1} = abs(eT1s - E);

% Modified Lanczos
31. f2 = @() Lanczos_modified(A, q, m);
32. times(2) = timeit(f2);
33. fprintf('Χρόνος εκτέλεσης Modified Lanczos: %.6f
    δευτερόλεπτα\n', times(2));
34. [T2, vT2, eT2, eT2s] = Lanczos_modified(A, q, m);
35. errors{2} = abs(eT2s - E);

% Reorthogonalized Lanczos
36. f3 = @() Lanczos_reorthogonalization(A, q, m, tol);
37. times(3) = timeit(f3);
38. fprintf('Χρόνος εκτέλεσης Reorthogonalized Lanczos:
    %.6f δευτερόλεπτα\n', times(3));
39. [T3, vT3, eT3, eT3s] = Lanczos_reorthogonalization(A,
    q, m, tol);
40. errors{3} = abs(eT3s - E);

% Proposed Algorithm
41. f4 = @() Powers(T1, tol);
42. times(4) = timeit(f4);
43. fprintf('Χρόνος εκτέλεσης Proposed Algorithm (T1):
    %.6f δευτερόλεπτα\n', times(4));
44. d1 = Powers(T1, tol);
45. d_vals{1} = d1;
46. d_vals{2} = Powers(T2, tol);
47. d_vals{3} = Powers(T3, tol);

```

```

%% --- Οπτικοποιήσεις ---
% Σύγκριση σφαλμάτων των αλγορίθμων
% Δημιουργία Διαγραμμάτων Σημείων
48. figure;
49. xp = 1:m;
50. colors = {'r', 'k', 'g', 'b'};
51. eigen_data = {eT1s, eT2s, d1, eT3s};
52. for i = 1:4
53.     subplot(4, 1, i);
54.     plot(xp, eigen_data{i}, '.', 'Color', colors{i},
'MarkerSize', 12, 'DisplayName', ['Υπολογισμένες - ',
algorithms{i}]);
55.     hold on;
56.     plot(xp, E, 'ko', 'MarkerSize', 8, 'DisplayName',
'Πραγματικές Ιδιοτιμές');
57.     title([algorithms{i}], 'FontSize', 12,
'FontWeight', 'bold');
58.     ylabel('Ιδιοτιμές', 'FontSize', 10);
59.     if i == 4
60.         xlabel('Δείκτης Ιδιοτιμής', 'FontSize', 10);
61.     end
62.     grid on;
63. end
% Δημιουργία Ραβδογραμμάτων
64. figure;
65. errors_ordered = {errors{1}, errors{2}, abs(E - d1),
errors{3}};
66. ylabel_text = {'|eT1 - E|', '|eT2 - E|', '|d1 - E|',
'|eT3 - E|'};
67. for i = 1:4
68.     subplot(2, 2, i);
69.     bar(1:length(errors_ordered{i}),
errors_ordered{i}, 'FaceColor', colors{i});
70.     title(algorithms{i}, 'FontSize', 12, 'FontWeight',
'bold');
71.     ylabel(ylabel_text{i}, 'FontSize', 10);
72.     xlabel('Δείκτης Ιδιοτιμής', 'FontSize', 10);
73.     grid on;
74. end
75.
76. sgtitle('Σύγκριση Σφαλμάτων', 'FontSize', 14,
'FontWeight', 'bold');

77. spectral_error_analysis(E, algorithms, eigen_data);

```

```
%% --- Συναρτήσεις ---
```

```
% Classic Lanczos Algorithm
```

```
78. function [T1,vT1,eT1,eT1s] = Lanczos_classic(A, v, m)
79. % Classic Lanczos algorithm to compute tridiagonal
    matrix T and orthogonal basis V
80. n = length(A);
81. V1 = zeros(n, m);
82. T1 = zeros(m, m);
83. V1(:, 1) = v / norm(v);
84.
85. beta = 0;
86. v_prev = zeros(n, 1);
87.
88. for k = 1:m
89.     w = A * V1(:, k) - beta * v_prev;
90.     alpha = V1(:, k)' * w;
91.     w = w - alpha * V1(:, k);
92.     beta = norm(w);
93.
94.     T1(k, k) = alpha;
95.     if k < m
96.         T1(k, k+1) = beta;
97.         T1(k+1, k) = beta;
98.     end
99.
100.    if beta < 1e-10
101.        break;
102.    end
103.
104.    v_prev = V1(:, k);
105.    V1(:, k+1) = w / beta;
106. end
107. [vT1,eT1]=eig(T1);
108. eT1s=sort(diag(eT1));
109. end
```

% Modified Lanczos Algorithm

```
110. function [T2,vT2,eT2,eT2s] = Lanczos_modified(A, v, m)
111. n = length(A);
112. V2 = zeros(n, m);
113. T2 = zeros(m, m);
114. V2(:, 1) = v / norm(v);
115. beta = 0;
116. q_prev = zeros(n, 1);
117.
118. for j = 1:m
119.     w = A * V2(:, j) - beta * q_prev;
120.     alpha = V2(:, j)' * w;
121.     w = w - alpha * V2(:, j);
122.     T2(j, j) = alpha;
123.     beta = norm(w);
124.
125.     if beta == 0
126.         fprintf('Invariant subspace found at iteration
%d\n', j);
127.         T2 = T2(1:j, 1:j);
128.         V2 = V2(:, 1:j);
129.         return;
130.     end
131.
132.     if j < m
133.         T2(j, j+1) = beta;
134.         T2(j+1, j) = beta;
135.     end
136.
137.     q_prev = V2(:, j);
138.     V2(:, j+1) = w / beta;
139. end
140. [vT2,eT2]=eig(T2);
141. eT2s=sort(diag(eT2));
142. end
```

% Reorthogonalization Lanczos Algorithm

```
143. function [T3,vT3,eT3,eT3s] =  
    Lanczos_reorthogonalization(A, v, m, tol)  
144. n = length(A);  
145. V3 = zeros(n, m);  
146. T3 = zeros(m, m);  
147. V3(:, 1) = v / norm(v);  
148. beta = 0;  
149. q_prev = zeros(n, 1);  
150.  
151. for j = 1:m  
152.     w = A * V3(:, j) - beta * q_prev;  
153.     alpha = V3(:, j)' * w;  
154.     w = w - alpha * V3(:, j);  
155.  
156.     for k = 1:j  
157.         reortho_correction = V3(:, k)' * w;  
158.         if abs(reortho_correction) > tol  
159.             w = w - reortho_correction * V3(:, k);  
160.         end  
161.     end  
162.  
163.     beta = norm(w);  
164.     T3(j, j) = alpha;  
165.     if j < m  
166.         T3(j, j+1) = beta;  
167.         T3(j+1, j) = beta;  
168.     end  
169.  
170.     if beta < tol  
171.         T3 = T3(1:j, 1:j);  
172.         V3 = V3(:, 1:j);  
173.         break;  
174.     end  
175.     q_prev = V3(:, j);  
176.     V3(:, j+1) = w / beta;  
177. end  
178. [vT3,eT3]=eig(T3);  
179. eT3s=sort(diag(eT3));  
180. end
```

% Proposed Algorithm

```
181. function [l,v]=Power(a,tol)
182. irep=0;
183. z=a*(ones(length(a(:,1)),1));
184. x=1;
185. xnew=x+10*tol;
186. while abs(x-xnew)>tol && irep<2000
187.     x=xnew;irep=irep+1;
188.     z=a*z/x;
189.     [xnew,xx]=max(abs(z));
190.     zprev=z(xx);
191. end
192. z=a*z;
193. l=z(xx)/zprev;
194. v=z/norm(z);
195. end
196. function d = Powers(a, tol)
197. n = length(a(:,1));
198. d = zeros(n, 1);
199. u = zeros(n, n);
200.
201. for i = 1:n
202.     [l, v] = Power(a, tol);
203.     a = a - l * (v * v');
204.     d(i) = l;
205.     u(:,i) = v;
206. end
207.
208. d = sort(d);
209. end
```

```

%Σφάλματα Φασματικής ακτίνας και Ελάχιστης ιδιοτιμής
210. function spectral_error_analysis(E, algorithms,
    eigen_data)
211. fprintf(' Ιδιοτιμές του πίνακα αποστάσεων\n');
212. maxA=max(E);
213. minA=min(E);
214. fprintf(' Φασματική ακτίνα: %.6f\n Ελάχιστη ιδιοτιμή:
    %.6f\n\n', maxA, minA);

215. for i = 1:length(algorithms)
216.     name = algorithms{i};
217.     e = eigen_data{i};
218.     maxT = max(e);
219.     minT = min(e);
220.     er_ra = abs(maxA - maxT);
221.     er_min = abs(minA - minT);
222.
223.     fprintf(' Ιδιοτιμές του %s:\n', name);
224.     fprintf(' Φασματική ακτίνα: %.6f\n Ελάχιστη
    ιδιοτιμή: %.6f\n', maxT, minT);
225.     fprintf(' Σφάλμα Φασματικής Ακτίνας: %.6e\n',
    er_ra);
226.     fprintf(' Σφάλμα Ελάχιστης Ιδιοτιμής: %.6e\n\n',
    er_min);
227. end
228. end

```


Παράρτημα III

Στη συνέχεια παρατίθενται οι πίνακες με τις ατομικές συντεταγμένες των δεδομένων που χρησιμοποιήθηκαν στην παρούσα έρευνα.

➤ Ατομικές συντεταγμένες της πρωτεΐνης **3pl6** από [33]:

$$\text{cord_3pl6} = \begin{pmatrix} 46.8270 & 67.7700 & -49.8070 \\ 47.1360 & 66.9100 & -50.9900 \\ 48.2050 & 65.8790 & -50.6170 \\ \vdots & \vdots & \vdots \\ 18.2570 & 20.9480 & 23.5640 \\ 18.5970 & 19.8230 & 23.1350 \\ 17.0970 & 21.1940 & 23.9600 \end{pmatrix}$$

➤ Ατομικές συντεταγμένες του ιού Ζίκα **5IZ7** από [34]:

$$\text{cord_5IZ7} = \begin{pmatrix} -84.8010 & -162.5370 & -107.1200 \\ -84.7930 & -163.0830 & -108.5060 \\ -85.5270 & -162.1490 & -109.4650 \\ \vdots & \vdots & \vdots \\ -96.0410 & -122.5110 & -115.6510 \\ -94.8460 & -121.1990 & -112.6600 \\ -95.3870 & -122.1490 & -111.7540 \end{pmatrix}$$