# ΑΞΙΟΛΟΓΗΣΗ ΠΟΛΥΩΝΥΜΙΚΩΝ ΥΠΟΛΟΓΙΣΤΙΚΩΝ ΠΑΚΕΤΩΝ

ΜΕΤΑΠΤΥΧΙΑΚΗ ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

**ΝΙΚΟΣ ΤΡΙΑΝΤΑΦΥΛΛΙΔΗΣ**

επιβλέπων: Ν. ΚΑΡΑΜΠΕΤΑΚΗΣ

ΜΕΤΑΠΤΥΧΙΑΚΟ ΠΡΟΓΡΑΜΜΑ ΣΠΟΥΔΩΝ
''ΘΕΩΡΗΤΙΚΗ ΠΛΗΡΟΦΟΡΙΚΗ
ΚΑΙ ΘΕΩΡΙΑ ΣΥΣΤΗΜΑΤΩΝ & ΕΛΕΓΧΟΥ''

ΤΜΗΜΑ ΜΑΘΗΜΑΤΙΚΩΝ
ΑΡΙΣΤΟΤΕΛΕΙΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΟΝΙΚΗΣ

Ιούνιος 2005

# ΑΞΙΟΛΟΓΗΣΗ ΠΟΛΥΩΝΥΜΙΚΩΝ ΥΠΟΛΟΓΙΣΤΙΚΩΝ ΠΑΚΕΤΩΝ

ΜΕΤΑΠΤΥΧΙΑΚΗ ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ


## ΝΙΚΟΣ ΤΡΙΑΝΤΑΦΥΛΛΙΔΗΣ

Επιβλέπων: Ν. Καραμπετάκης


Εγκρίθηκε από την τριμελή επιτροπή την .................

Ν. Καραμπετάκης .................

Μ.Γουσίδου Κουτίτα .................

Α.Ι.Βαρδουλάκης .................

Θεσσαλονίκη, Ιούνιος 2005

Τριανταφυλλίδης Νίκος ................
Πτυχιούχος Φυσικός Α.Π.Θ

# ΠΕΡΙΛΗΨΗ

Στα συστήματα αυτομάτου ελέγχου οι υπολογιστικές εφαρμογές στους Η/Υ κατέχουν κύρια θέση και η περαιτέρω εξέλιξη τους θεωρείται απαραίτητη για την ανάπτυξη του συγκεκριμένου κλάδου επιστήμης. Από τα τέλη της δεκαετίας του `80 υπάρχει ραγδαία αύξηση των λογισμικών ειδικευμένα στον έλεγχο προερχόμενα από ερευνητικές ομάδες ανά τον κόσμο.

Η γενικότερη έννοια της ανάπτυξης λογισμικού σχετίζεται άμεσα με την διαδικασία βελτιστοποίησης (optimization) του, της οποίας πρώτο βήμα είναι η αξιολόγηση (benchmarking). Συνεπώς, στην περίπτωση των υπολογιστικών προγραμμάτων συστημάτων ελέγχου το benchmarking μπορεί να χρησιμοποιηθεί ως στοιχείο αξιολόγησης και να συμβάλλει στην άμεση βελτιστοποίηση τους.

Στην παρούσα εργασία παρουσιάζεται ένα καινούργιο λογισμικό υπό το όνομα POLYBENCH το οποίο ειδικεύεται στην αυτόματη αξιολόγηση (benchmarking) πολυωνυμικών μεθόδων πάνω στον έλεγχο συστημάτων. Συγκεκριμένα, δώδεκα κώδικες οι οποίοι προσφέρουν λύση σε γνωστά προβλήματα της πολυωνυμικής θεωρίας όπως η διοφαντική εξίσωση πολυωνυμικών πινάκων ελέγχονται για την αποδοτικότητα και την αξιοπιστία τους. Ακόμα, παρουσιάζεται μια πρώτη συγκριτική ανάλυση των αποτελεσμάτων που προκύπτουν από μία συγκεκριμένη εφαρμογή του λογισμικού POLYBENCH.

# ABSTRACT

The computer applications of control systems play an important role in the development of the relevant branch of science. Since the late '80s there is a significant increase in the amount of this specific software produced around the world.

In general, the development of software, commercial or not, is strictly related with the optimizing procedure whose first step is the proceess of benchmarking. Consequently, in control systems software benchmarking can be used as an optimizing tool.

A software platform for automatic benchmarking of numerical codes for advanced polynomial methods is presented in the present report. Twelve important in control systems survey solvers from three well known control libraries are tested for their efficiency. The results are presented along with some comments on them.

Η εργασία πραγματοποιήθηκε στο Center for Applied Cybernetics (Control Engineering Department, Electrical Engineering Faculty, Czech Technical University).

Θα ήθελα να ευχαριστήσω προσωπικά για την δημιουργική συμβολή τους στην ολοκλήρωση της παρούσας εργασίας τους ερευνητές Zdenek Hurak, Petr Kujan και Martin Hromcik.

Τέλος, θα ήθελα να μνημονεύσω το πρόγραμμα ανταλλαγής φοιτητών ERASMUS και το ίδρυμα υποτροφιών Ι.Κ.Υ. για την υλική και όχι μόνο συμπαράσταση τους στην παραμονή μου στο εθνικό πολυτεχνείο της Τσεχίας.

# Περιεχόμενα

# Κεφάλαιο 1

# ΕΙΣΑΓΩΓΗ

## 1.1 Γενικά

Η έννοια της *αξιολόγησης* (*benchmarking*) στην επιστήμη των υπολογιστών ξεκινά από τις αρχές της δεκαετίας του '80. Μονάδες ηλεκτρονικών υπολογιστών όπως κάρτες γραφικών, ήχου ή κεντρικών μονάδων επεξεργασίας (CPU) υφίστανται την διαδικασία αξιολόγησης με σκοπό την περαιτέρω εξέλιξη τους. Στην ανάπτυξη λογισμικού το benchmarking θεωρείται ένα βασικό εργαλείο για τη βελτιστοποίηση της αποδοτικότητας και γενικά της ποιότητας υπολογιστικών προγραμμάτων. Για παράδειγμα, η βελτιστοποίηση ενός προγράμματος μπορεί να αξιοποιήσει πλήρως τις δυνατότητες ενός Η/Υ χωρίς να χρειάζεται η αντικατάσταση του τελευταίου με έναν νεώτερης τεχνολογίας.

Η αξιολόγηση επιτυγχάνεται μέσω της σύγκρισης της απόδοσης (performance) ορισμένων κοινά αποδεκτών μετρητών (metrics) κατά τη χρονική διάρκεια μιας υπολογιστικής διαδικασίας π.χ. ο υπολογισμός του γινομένου δυο πινάκων. Συνήθεις μετρητές είναι οι

- CPU time
- MFLOPS
- CLOCK time
- Cache memory misses
- Error

Οι παραπάνω μετρητές σχετίζονται με το ηλεκτρολογικό μέρος του Η/Υ (hardware) π.χ. CPU time είτε με το υπό εξέταση λογισμικό π.χ. Error το οποίο αναφέρεται στην ακρίβεια των αποτελεσμάτων. Οι μετρήσεις που λαμβάνουν χώρα οδηγούν σε αποτελέσματα που χρίζουν σχολιασμού ο οποίος αντικατοπτρίζει πολλές φορές αντικρουόμενες απόψεις. Τα αποτελέσματα της αξιολόγησης παρουσιάζονται συνήθως υπό την μορφή πινάκων και γραφικών παραστάσεων.

Η βελτιστοποίηση προγραμμάτων προορισμένων για χρήση από Η/Υ (optimization) μπορεί να αποφέρει πολλαπλά κέρδη με χαμηλό οικονομικό και όχι μόνο κόστος. Για παράδειγμα, η πλήρης αξιοποίηση ενός αλγορίθμου, η οποία μπορεί να προέλθει από την συνεχόμενη βελτιστοποίηση του αντίστοιχου υπολογιστικού κώδικα μπορεί να αναβάλλει την ανάγκη εύρεσης ενός θεωρητικά πιο γρήγορου νέου αλγορίθμου.

Η παρούσα εργασία έχει διττό στόχο. Ο πρώτος έγκειται στην ανάπτυξη ενός λογισμικού ειδικευμένου στην αξιολόγηση (benchmarking) υπολογιστικών βιβλιοθηκών που βασίζονται στην πολυωνυμική περιγραφή των συστημάτων ελέγχου. Το λογισμικό αυτό θα πρέπει να έχει τη δυνατότητα μεταφοράς μεταξύ των διάφορων λειτουργικών συστημάτων χωρίς ιδιαίτερη δυσκολία. Ο παραπάνω στόχος πετυχαίνεται σε μεγάλο βαθμό με την ανάπτυξη της πλατφόρμας με τη χρησιμοποίηση της γλώσσας PYTHON η οποία διαθέτει όλα τα απαραίτητα χαρακτηριστικά:

- ανεξάρτητη λειτουργικού συστήματος (platform independent)
- ανοικτού κώδικα (open source)
- υψηλού επιπέδου (high level)

Ο δεύτερος σκοπός της εργασίας είναι η δημιουργία ενός πλαισίου σχολιασμού των αποτελεσμάτων μιας συγκεκριμένης αξιολόγησης. Η ανάλυση των αποτελεσμάτων και η προσπάθεια εύρεσης των αιτιών της διακύμανσης τους σκοπό έχει να αποτελέσει εφαλτήριο για των περαιτέρω ανάπτυξη των πολυωνυμικών μεθόδων και των αντίστοιχων υπολογιστικών βιβλιοθηκών

## 1.2 Πολυωνυμικές Περιγραφές Συστημάτων

Τα συστήματα ελέγχου αποτελούν κυρίαρχο εργαλείο στην ολοένα αυξανόμενη ανάπτυξη της τεχνολογίας και παρουσιάζουν ένα εξαιρετικά ευρύ φάσμα εφαρμογών. Συγκεκριμένα, οι αναπαραστάσεις των συστημάτων ελέγχου σε ηλεκτρολογικό, μηχανολογικό, χημικό επίπεδο γνωρίζουν μεγάλη ανάπτυξη εδώ και δύο δεκαετίες.

Η περιγραφή ενός δυναμικού συστήματος είναι εφικτή μέσω των state space καταστάσεων όπου η μελέτη του οδηγεί στην λύση διαφορικών εξισώσεων και εξισώσεων διαφορών των οποίων η λύση προσεγγίζεται με αριθμητικές μεθόδους στις περισσότερες περιπτώσεις.

Από τα τέλη της δεκαετίας του `70 υφίσταται η χρησιμοποίηση πολυωνυμικών πινάκων για μελέτη προβλημάτων ελέγχου με θεωρητικό υπόβαθρο. Παρόλο αυτά, μόνο στις αρχές της δεκαετίας του `90 γνωστές αριθμητικές μέθοδοι της γραμμικής άλγεβρας όπως LU, QR, SVD decompositions συνδυάστηκαν με τους πολυωνυμικούς πίνακες όπου οι τελευταίοι παρουσιάζονταν υπό την μορφή πολυωνύμων με μέλη πίνακες οπότε άρχισαν να χρησιμοποιούνται σε πρακτικά προβλήματα ελέγχου.

Στις μέρες μας, οι πολυωνυμικές μέθοδοι αποτελούν βασικό θεωρητικό και υπολογιστικό εργαλείο στην σχεδίαση συστημάτων ελέγχου και στην επεξεργασία σήματος. Χαρακτηριστικό παράδειγμα είναι η περιγραφή γραμμικών συστημάτων σε κλασματική μορφή με τη βοήθεια συναρτήσεων μεταφοράς. Συγκεκριμένα, στα MIMO - Multiple Input Multiple Output - συστήματα το παραπάνω σκεπτικό γενικεύεται σε αριστερά και δεξιά κλάσματα πολυωνυμικών πινάκων των οποίων η μελέτη εναπόκειται στον κλάδο της πολυωνυμικής θεωρίας πινάκων. Οι υπολογιστικές μέθοδοι για την λύση γραμμικών εξισώσεων πολυωνυμικών πινάκων (π.χ. διοφαντικές εξισώσεις) και φασματικών παραγοντοποιήσεων ( spectral factorization ) αποτελούν βασικές συνιστώσες στην σχεδιάση συστημάτων ελέγχου. Από την πλευρά της αριθμητικής ανάλυσης τα περιγραφέντα προβλήματα οδηγούν συνήθως στην λύση δομημένων block Teoplitz γραμμικών συστημάτων.

Γενικά, οι πολυωνυμικές μέθοδοι αποτελούν μια διαφορετική προσέγγιση στις προυπάρχουσες state space περιγραφές οι οποίες βασίζονται κυρίως στη μελέτη των εξισώσεων Lyapunov και Riccati.

Οι παραπάνω δύο μέθοδοι αποτελούν αντικείμενο αντιπαραθέσεων μεταξύ επιστημόνων ειδικευμένων στον έλεγχο για την χρησιμότητά τους και για το ποια είναι η επικρατέστερη προσέγγιση. Συνολικά οι δύο μέθοδοι δίνουν την δυνατότητα ανάλυσης συστημάτων από διαφορετικές οπτικές γωνίες αφού χρησιμοποιούν διαφορετικά μαθηματικά εργαλεία. Συμπερασματικά, οι περιγραφές καταστάσεων χώρου state space και οι πολυωνυμικές αποτελούν συμπληρωματικές θεωρίας που σε συνδυασμό ανοίγουν νέες προοπτικές στην ανάπτυξη της θεωρίας ελέγχου.

Στην παρούσα εργασία δίνεται έμφαση στις πολυωνυμικές μεθόδους που έχουν αναπτυχθεί και έχουν πάρει πρακτική χροιά όντας σε κώδικες υπολογιστών. Η προσπάθεια έγκειται στην ανάδειξη των συγκεκριμένων μεθόδων διότι προσφέρουν λύση σε προβλήματα σχετικά με τα συστήματα ελέγχου. Ο σκοπός είναι η διαπίστωση της αξιοπιστίας της λύσης αυτής είτε μετρώντας την απόδοση συγκεκριμένων μετρητών που σχετίζονται με την ταχήτυτα της λύσης ή με την ακρίβεια της.

## 1.3 Λογισμικά Συστημάτων Ελέγχου

Οι εφαρμογές των συστημάτων ελέγχου οδηγούν στην ανάπτυξη σχετικών λογισμικών τα οποία δημιουργούν νέα δεδομένα και επεκτείνουν τις δυνατότητες της θεωρίας ελέγχου. Για παράδειγμα εξισώσεις με πίνακες πολύ μεγάλων διαστάσεων ή διαφορικές εξισώσεις χωρίς αναλυτική λύση χρειάζονται την βοήθεια Η/Υ για να αντιμετοπιστούν. Πολλές ερευνητικές ομάδες ανά τον κόσμο έχουν ασχοληθεί με την ανάπτυξη λογισμικού το οποίο παίζει σημαντικό ρόλο στην εφαρμογή της θεωρίας ελέγχου σε πρακτικά προβλήματα των οποίων θα αναζητείται η λύση. Τα πακέτα λογισμικών στον έλεγχο ειδικεύονται όπως και η αντίστοιχη θεωρία στην state space και πολυωνυμική περιγραφή των συστημάτων.

Οι πιο γνωστές γλώσσες που προσφέρουν πακέτα πάνω στον έλεγχο είναι:

- MATLAB
  (CONTROL SYSTEM TOOLBOX, POLYNOMIAL TOOLBOX)
- MATHEMATICA
  (CONTROL SYSTEMS PROFESSIONAL, ADVANCED NUMERICAL METHODS, POLYNOMIAL PACKAGE,
  DESCRIPTOR PACKAGE )
- SCILAB
- MUPAD
- SLICOT
- OCTAVE
- MAPLE

Το πρώτο βήμα στην ανάπτυξη λογισμικού με σκοπό την επεξεργασία πολυωνυμικών πινάκων συγκαταλέγεται στην προσπάθεια αποθήκευσης τους στην μνήμη του υπολογιστή. Οι πολυωνυμικοί πίνακες αποτελούν ένα τρισδιάστατο αντικείμενο όπου οι πρώτες δύο διαστάσεις είναι ο συνηθισμένος πίνακας (με μέλη αριθμούς) και η τρίτη διάσταση προστίθεται λόγω του πολυωνύμου που υφίσταται. Η διαδικασία αυτή γίνεται συνήθως σειριακά από την πρώτο στοιχείο του πίνακα μέχρι το τελευταίο. Ακόμα, σε πολλές περιπτώσεις οι πολυωνυμικοί πίνακες έχουν ως μέλη μεγάλο αριθμό μηδενικών και σε συγκριτικά λίγες θέσεις πολυώνυμα (sparse matrices). Οπότε, ο βέλτιστος τρόπος εισαγωγής στην μνήμη του υπολογιστή των πολυωνυμικών πινάκων πρέπει να λαμβάνει υπόψη τις παραπάνω ιδιαιτερότητες της φύσης τους.

Οι χρήστες της υπολογιστικής γλώσσας MATLAB έχουν την δυνατότητα της χρησιμοποίησης του πιο πλήρους λογισμικού πακέτου για πολυωνυμικούς πίνακες ειδικευμένο στον έλεγχο συστημάτων με την ονομασία POLYNOMIAL TOOLBOX. Το συγκεκριμένο λογισμικό διατίθεται εμπορικά από την εταιρία PolyX και περιλαμβάνει πάνω από 100 συναρτήσεις (solvers) που καλύπτουν το σύνολο των εφαρμογών συστημάτων ελέγχου που βασίζεται στην πολυωνυμική περιγραφή τους. Για παράδειγμα, υπάρχουν συναρτήσεις που δίνουν λύση σε πολυωνυμικές εξισώσεις πινάκων, βρίσκουν αντιστρόφους, ορίζουσες, .... Επιπλέον, σε γλώσσες όπως η MATHEMATICA, SCILAB υπάρχουν παρόμοιες πολυωνυμικές βιβλιοθήκες χωρίς όμως να παρουσιάζουν την πληρότητα του POLYNOMIAL TOOLBOX.

# Κεφάλαιο 2

# POLYBENCH

## 2.1 Γενικά

Ο κύριος στόχος της παρούσας εργασίας είναι η ανάπτυξη λογισμικού ειδικευμένου στο benchmarking πολυωνυμικών μεθόδων που εμφανίζονται στα παρακάτω πακέτα λογισμικού ελέγχου συστημάτων:

- POLYNOMIAL TOOLBOX version 3.0.19 [1]– MATLAB version 7.0.0.19920 (R14) [8]
- POLYNOMIAL PACKAGE version 2.1 [2]– MATHEMATICA version 5.1.0.0 [9]
- SCILAB unstable version 3.0-u-20050128 [3]

Τα παραπάνω πακέτα προσφέρουν μία σειρά από solvers σε προβλήματα πολυωνυμικού ελέγχου. Από το σύνολο αυτό έχουν επιλεγεί δώδεκα προβλήματα τα οποία απαντώνται συχνά στην πολυωνυμική θεωρία ελέγχου και η λύση τους αποτελεί αντικείμενο συνεχούς μελέτης. Τα πιο χαρακτηριστικά από αυτά είναι η γραμμική εξίσωση πολυωνυμικών πινάκων ή η συγγενής της διοφαντική, όπως και η φασματική παραγοντοποίηση. Η λύση τους απαιτείται για την εύρεση ελεγκτών σε δυναμικά συστήματα και στη μοντελοποίηση συστημάτων αντίστοιχα.

Ακόμα, για την σχεδίαση της πλατφόρμας POLYBENCH χρησιμοποιήθηκε το εξής λογισμικό:

- PYTHON 2.3.4 [4]
- Matplotlib 0.74 [5]
- Karrigell 2.1.2 [6]
- Papi 3.0 [7]

Η πλατφόρμα λογισμικού POLYBENCH έχει την δυνατότητα της αυτόματης διενέργειας της διαδικασίας αξιολόγησης (benchmarking) και στη συνέχεια της παρουσίασης των αποτελεσμάτων σε μορφή δυναμικής ιστοσελίδας αποτελούμενη από αντίστοιχους πίνακες και γραφικές παραστάσεις.

Οι συναρτήσεις που αξιολογήθηκαν είναι δώδεκα τον αριθμό σε όποιες βιβλιοθήκες υπάρχουν. Συγκεκριμένα:

- Πολλαπλασιασμός πολυωνυμικών πινάκων : $\mathbf{C(s)} = \mathbf{A(s)} \cdot \mathbf{B(s)}$
(Polynomial Toolbox, Polynomial Package, Scilab)

- Πρόσθεση πολυωνυμικών πινάκων : $\mathbf{C(s)} = \mathbf{A(s)} + \mathbf{B(s)}$
(Polynomial Toolbox, Polynomial Package, Scilab)

- Ορίζουσα πολυωνυμικού πίνακα : $\mathbf{C(s)} = \mathbf{Det}[\mathbf{A(s)}]$
(Polynomial Toolbox, Polynomial Package, Scilab)

- Αντίστροφος πολυωνυμικού πίνακα : $\mathbf{C(s)} = \mathbf{Inv}[\mathbf{A(s)}]$
(Polynomial Toolbox, Polynomial Package, Scilab)

- Γραμμική εξίσωση πολυωνυμικών πινάκων: $\mathbf{A(s)} \cdot \mathbf{X(s)} = \mathbf{B(s)}$
(Polynomial Toolbox, Polynomial Package)

- Διοφαντική εξίσωση πολυωνυμικών πινάκων: $\mathbf{A(s)} \cdot \mathbf{X(s)} + \mathbf{B(s)} \cdot \mathbf{Y(s)} = \mathbf{C(s)}$
(Polynomial Toolbox, Polynomial Package)

- Διοφαντική εξίσωση πολυωνύμων: $\mathbf{a(s)} \cdot \mathbf{x(s)} + \mathbf{b(s)} \cdot \mathbf{y(s)} = \mathbf{c(s)}$
(Polynomial Toolbox, Polynomial Package, Scilab)

- Διμερής συμμετρική εξίσωση πολυωνυμικών πινάκων: $\mathbf{A}'(\mathbf{s}) \cdot \mathbf{X(s)} + \mathbf{X}'(\mathbf{s}) \cdot \mathbf{A(s)} = \mathbf{B(s)}$
(Polynomial Toolbox)

- Διμερής συμμετρική εξίσωση πολυωνύμων: $\mathbf{a}'(\mathbf{s}) \cdot \mathbf{x(s)} + \mathbf{x}'(\mathbf{s}) \cdot \mathbf{a(s)} = \mathbf{b(s)}$
(Polynomial Toolbox, Polynomial Package)

- Φασματική παραγοντοποίηση πολυωνυμικών πινάκων: $\mathbf{X}'(\mathbf{s}) \cdot \mathbf{X(s)} = \mathbf{A(s)}$
(Polynomial Toolbox)

- J - Φασματική παραγοντοποίηση πολυωνυμικών πινάκων: $\mathbf{X}'(\mathbf{s}) \cdot \mathbf{J} \cdot \mathbf{X(s)} = \mathbf{A(s)}$
(Polynomial Toolbox)

- Δεξιά διαίρεση πολυωνυμικών πινάκων: $\mathbf{N(s)} = \mathbf{D(s)} \cdot \mathbf{Q(s)} + \mathbf{R(s)}$
(Polynomial Toolbox, Scilab)

## 2.2 Ανάλυση Λογισμικού

Η διαδικασία λειτουργίας της πλατφόρμας POLYBENCH λαμβάνει χώρα στο batch mode επίπεδο (επίπεδο κονσόλας) του λειτουργικού συστήματος WINDOWS. Συγκεκριμένα, χρησιμοποιήθηκαν τα WINDOWS XP SP1. Η γενική φιλοσοφία της όλης σχεδίασης είναι η εύκολη μεταφορά του λογισμικού μεταξύ των διάφορων λειτουργικών συστημάτων υπολογιστών. Με αυτό το σκεπτικό, επιλέχθηκε η ανοικτού κώδικα (open source) γλώσσα PYTHON για την ανάπτυξη της πλατφόρμας αξιολόγησης.

Υπάρχουν τρεις κατηγορίες αρχείων - προγραμμάτων (scripts) που τρέχουν στον ηλεκτρονικό υπολογιστή και αποτελούν την πλατφόρμα POLYBENCH.

### 2.2.1 1η κατηγορία αρχείων

Η πρώτη κατηγορία αποτελείται από τα batch scripts τα οποία είναι υπεύθυνα για την έναρξη της όλης διαδικασίας. Δηλαδή, καλούν τα benchmark scripts που είναι γραμμένα για τις γλώσσες υψηλού επιπέδου MATLAB, MATHEMATICA, SCILAB και στη συνέχεια καλούν τα αρχεία που είναι γραμμένα σε PYTHON τα οποία διαχειρίζονται τα αποτελέσματα της αξιολόγησης. Υπάρχουν δώδεκα τέτοια αρχεία με την παρακάτω μορφή (π.χ. πολλαπλασιασμός πινάκων):

```
:: Multiplication of Polynomial Matrices
cls
matlab -nosplash -nodesktop -minimize -r mat1
math <math1.txt
scilex -f sci1.sce
python bench1.py
```

Η πρώτη σειρά αποτελεί σχόλιο χαρακτηριστικό του αρχείου. Η δεύτερη σειρά αποτελεί εντολή στην γλώσσα της κονσόλας των WINDOWS και δίνει την εντολή για καθαρισμό της οθόνης από τυχόν προηγούμενες εντολές. Η τρίτη σειρά καλεί το MATLAB και τον MEX compiler του να τρέξουν το m-file mat1 σε batch mode δηλαδή χωρίς να ανοίξει όλο το γραφικό περιβάλλον που συνοδεύει τη συγκεκριμένη γλώσσα. Η τέταρτη γραμμή καλεί την γλώσσα MATHEMATICA στο περιβάλλον της κονσόλας και τον αντίστοιχο compiler της να τρέξουν το αρχείο math1. Χαρακτηριστικό στοιχείο αποτελεί το γεγονός ότι το αρχείο είναι text file και δεν έχει την γνωστή κατάληξη που έχουν τα αρχεία της MATHEMATICA. Η επόμενη σειρά είναι η τελευταία όσον αναφορά τα benchmark scripts. Η γλώσσα SCILAB καλείται, πάντα στο επίπεδο της κονσόλας, να τρέξει το αρχείο sci1 το οποίο έχει την χαρακτηριστική κατάληξη των προγραμμάτων γραμμένα στην συγκεκριμένη γλώσσα.

Τέλος, η τελευταία σειρά αποτελείται από την εντολή που καλεί την γλώσσα PYTHON και το αντίστοιχο αρχείο της το οποίο επεξεργάζεται τα αποτελέσματα των προηγούμενων εντολών.

Η φιλοσοφία η οποία χαρακτηρίζει τα παραπάνω αρχεία σχετίζεται με το γεγονός της χρησιμοποίησης της κονσόλας στο λειτουργικό σύστημα WINDOWS κατά τρόπο ανάλογο με την λειτουργία της περίφημης κονσόλας των λειτουργικών συστημάτων UNIX. Τα WINDOWS προσφέρουν κονσόλα μειωμένων δυνατοτήτων διότι συγκεντρώνουν της δυναμή τους σε εφαρμογές με γραφικό περιβάλλον.

Από τις τρεις γλώσσες υψηλού επιπέδου που περιλαμβάνει το λογισμικό POLYBENCH, το MATHEMATICA και το SCILAB προσφέρουν την δυνατότητα λειτουργίας σε περιβάλλον κονσολάς εκτός από την δεδομένη λειτουργία σε παραθυρικό περιβάλλον. Το MATLAB όμως, δεν έχει την δυνατότητα να λειτουργήσει στην κονσόλα των WINDOWS όπως τα άλλα δύο εξεταζόμενα πακέτα λογισμικού και καλείται απλά να ανοίξει με minimized το παραθυρικό του περιβάλλον.

### 2.2.2 2η κατηγορία αρχείων

Η δεύτερη κατηγορία των αρχείων είναι αυτά τα οποία εκτελούν την αξιολόγηση (benchmarking) και τα οποία καλούνται από τα batch αρχεία. Συγκεκριμένα, υπάρχουν αρχεία με κατάληξη .m, .txt, .sce γραμμένα στις γλώσσες MATLAB, MATHEMATICA, SCILAB αντίστοιχα.

Η βασική σχεδίαση τους έγκειται σε μια επαναλαμβανόμενη διαδικασία (loop) αυξανόμενου μεγέθους τετραγωνικών πολυωνυμικών πινάκων. Ο βαθμός των πολυωνύμων είναι σταθερός και ισούται με δύο. Ο βασικός λόγος για την επιλογή αυτή είναι ότι η φυσική περιγραφή δυναμικών συστημάτων γίνεται με διαφορικές εξισώσεις δευτέρου βαθμού στις περισσότερες περιπτώσεις. Ο βαθμός των πολυωνύμων στους πίνακες αυξάνεται συνήθως στην περίπτωση της εύρεσης ελεγκτών (controllers) ενός συστήματος ελέγχου.

Στα αρχεία αξιολόγησης υπάρχει μία ακόμη επαναλαμβανόμενη διαδικασία η οποία σταματάει το ‘‘τρέξιμο’’ του προγράμματος για ένα συγκεκριμένο χρονικό όριο λειτουργίας της ΚΜΕ (Κεντρικής Μονάδας Επεξεργασίας - CPU). Ο σκοπός της παραπάνω σχεδίασης είναι το γεγονός ότι όλοι οι κώδικες - (solvers) δεν είναι στο ίδιο επίπεδο αποδοτικότητας (efficiency) και οι δυνατότητες του εκάστοτε Η/Υ, όπου λαμβάνει χώρα το benchmarking, έχουν συγκεκριμένα υπολογιστικά όρια.

Οι μετρητές (counters) που χρησιμοποιούνται σχετίζονται με τη δυνατότητα που προσφέρεται από τις χρησιμοποιούμενες γλώσσες προγραμματισμού για την μέτρηση της απόδοσης προγραμμάτων γραμμένα σε αυτές. Το περιβάλλον του MATLAB όπου λειτουργεί το POLYNOMIAL TOOLBOX προσφέρει την δυνατότητα της μέτρησης της χρονικής διάρκειας - σε δευτερόλεπτα - κατά την οποία δουλεύει η CPU. Η παραπάνω δυνατότητα μέτρησης προσφέρεται και στις υπόλοιπες δύο υπολογιστικές πλατφόρμες δηλαδή στις MATHEMATICA, SCILAB. Επιπλέον, με την εισαγωγή μιας ειδικής benchmark βιβλιοθήκης στο MATLAB υπό το όνομα PAPI γίνεται δυνατή η μέτρηση του μέσου όρου των MFLOPS. Τέλος, το πακέτο λογισμικού για πολυωνυμικές μεθόδους POLYNOMIAL TOOLBOX προσφέρει την δυνατότητα υπολογισμού της norm ενός πολυωνυμικού πίνακα. Αυτό δίνει τη δυνατότητα του υπολογισμού της σχετικής norm του υπολοίπου, το οποίο θεωρείται σαν λάθος (ακρίβεια) - error του υπολογιστικού κώδικα (solvers).

Συγκεκριμένα η benchmarking platform POLYBENCH περιλαμβάνει τους παρακάτω μετρητές.

- CPU time
  (POLYNOMIAL TOOLBOX, POLYNOMIAL PACKAGE, SCILAB)
- MFLOPS
  (POLYNOMIAL TOOLBOX)
- $10^{10}$·[norm(residue)/norm(input)]
  (POLYNOMIAL TOOLBOX)

Το τελικό αποτέλεσμα που προκύπτει από αυτά τα αρχεία (scripts) είναι καινούργια αρχεία κειμένων (text files). Δηλαδή, είναι αρχεία όπου η πληροφορία δίδεται σαν strings υπό μορφή στηλών.

**Αρχεία αξιολόγησης στο MATLAB** Στη γλώσσα MATLAB ένα αρχείο benchmark, για παράδειγμα πολλαπλασιασμός πολυωνυμικών πινάκων έχει τις παρακάτω εντολές:

↪ εντολή εισαγωγής του πακέτου POLYNOMIAL TOOLBOX

*pinit*

↪ άνοιγμα εξωτερικού αρχείου για γράψιμο

$f = fopen('matlab.out','w');$

↪ τύπωση των μετρητών στο εξωτερικό αρχείο

$fprintf(f,'\%12s\ \%12s\ \%12s\ \backslash n \backslash n','Mflops','Size','CPUtime');$

↪ d=βαθμός του πολυωνυμικού πίνακα,fin=το μέγιστο μέγεθος του πολυωνυμικού πίνακα
↪ st=το βήμα της επαναληπτικής διαδικασίας, ao=αρχική τιμή του βήματος επανάληψης

$d = 2; fin = 2000; st = 100; ao = 100;$

$\hookrightarrow$ αρχή της επαναληπτικής διαδικασίας αξιολογησης

$for\ n = ao : st : fin,$

$\hookrightarrow$ δημιουργία δύο πολυωνυμικών πινάκων δευτέρου βαθμού με random coefficients

$A = prand(d, n); B = prand(d, n);$

$\hookrightarrow$ μηδενισμός των μετρητών των CPU time, MFLOPS

$t1 = cputime; flops(0);$

$\hookrightarrow$ διεργασία προς αξιολόγηση: Πολλαπλασιασμός πολυωνυμικών πινάκων

$C = mtimes(A, B);$

$\hookrightarrow$ μέτρηση της τιμής των μετρητών CPU time, MFLOPS

$t2 = cputime - t1; [ops, mflops] = flops;$

$\hookrightarrow$ η επαναληπτική διαδικασία αξιολόγησης σταματάει αν περάσει ένα προκαθορισμένο χρονικό όριο

$if\ t2 > 200$

$fprintf(f,' \%12.2f\ \%12d\ \%12.2f\ \backslash n', mflops, n, t2);$

$for\ n = n + st : st : fin,$

$t2 = 0; mflops = 0;$

$fprintf(f,' \%12d\ \%12d\ \%12d\ \backslash n', mflops, n, t2);$

$end$

$exit$

$end$

$\hookrightarrow$ τύπωση των αποτελεσμάτων των μετρήσεων

$fprintf(f,'\%12.2f\ \%12d\ \%12.2f\ \backslash n', mflops, n, t2);$

$\hookrightarrow$ τέλος της επαναληπτικής διαδικασίας

$end$

$exit$

**Αρχεία αξιολόγησης στο MATHEMATICA** Στη γλώσσα MATHEMATICA ένα αρχείο benchmark, για παράδειγμα πολλαπλασιασμός πολυωνυμικών πινάκων έχει τις παρακάτω εντολές:

$\hookrightarrow$ εντολή εισαγωγής του πακέτου POLYNOMIAL PACKAGE

$<< Polynomial`;$

$\hookrightarrow$ θέσιμο της διεύθυνσης εργασίας

$SetDirectory[dir];$

$\hookrightarrow$ άνοιγμα εξωτερικού αρχείου για γράψιμο

$strm = OpenWrite[mathematica.out, FormatType- > OutputForm];$

$\hookrightarrow$ γράψιμο των μετρητών στο εξωτερικό αρχείο

$Write[strm, "Degree\ \backslash t\ Size\ \backslash t\ CPUtime \backslash n"];$

$\hookrightarrow$ d=βαθμός του πολυωνυμικού πίνακα, fin=το μέγιστο μέγεθος του πολυωνυμικού πίνακα

$\hookrightarrow$ st=το βήμα της επαναληπτικής διαδικασίας, ao=αρχική τιμή του βήματος επανάληψης

$d = 2; fin = 2000; st = 100; ao = 100;$

$Do[$

$\hookrightarrow$ δημιουργία δύο τετραγωνικών πολυωνυμικών πινάκων με τυχαίους συντελεστές

$a = PMRandom[d, n]; b = PMRandom[d, n];$

$\hookrightarrow$ πολλαπλασιασμός δύο πολυωνυμικών πινάκων

$\hookrightarrow$ μέτρηση της CPU time που χρειάστηκε

$t = Timing[Dot[a, b];]\ /.\ Second- > 1;$

$\hookrightarrow$ η επαναληπτική διαδικασία αξιολόγησης σταματάει αν περάσει ένα προκαθορισμένο χρονικό όριο

$If[t[[1]]\ >\ 150,$

$Write[strm, d, "\ \backslash t\ ", n, "\ \backslash t\ ", t[[1]]];$

$\hookrightarrow$ αρχή της επαναληπτικής διαδικασίας αξιολόγησης

$Do[$

$Write[strm, d, " \backslash t ", n, " \backslash t ", 0];$

$, n, n + st, fin, st];$

$Close[strm];$

$Quit[];$

$\hookrightarrow$ τύπωση των τιμών του μετρητή

$Write[strm, d, " \backslash t ", n, " \backslash t ", t[[1]]];$

$, n, ao, fin, st];$

$\hookrightarrow$ τέλος της επαναληπτικής διαδικασίας αξιολόγησης

$Close[strm];$

$Quit[];$

$\square$

**Αρχεία αξιολόγησης στο SCILAB** Στη γλώσσα SCILAB ένα αρχείο benchmark, για παράδειγμα πολλαπλασιασμός πολυωνυμικών πινάκων έχει τις παρακάτω εντολές:

$\hookrightarrow$ άνοιγμα εξωτερικού αρχείου για γράψιμο

$f = mopen('scilab.out','w');$

$\hookrightarrow$ τύπωση των μετρητών στο εξωτερικό αρχείο

$fprintf(f,"Degree\ Size\ CPUtime\ \backslash n \backslash n")$

$\hookrightarrow$ δημιουργία ενός πολυωνύμου δευτέρου βαθμού με τυχαίους συντελεστές

$s = poly([1,2],\ 's');$

$\hookrightarrow$ d=βαθμός του πολυωνυμικού πίνακα, fin=το μέγιστο μέγεθος του πολυωνυμικού πίνακα

$\hookrightarrow$ st=το βήμα της επαναληπτικής διαδικασίας, ao=αρχική τιμή του βήματος επανάληψης

$d = 2; fin = 2000; st = 100; ao = 100;$

$\hookrightarrow$ αύξηση της χωρητικότητας της stack memory

$stacksize(100000006)$

$lines(0);$

$\hookrightarrow$ αρχή της επαναληπτικής διαδικασίας αξιολογησης

$for\ n = ao : st : fin,$

$\hookrightarrow$ δημιουργία δύο πολυωνυμικών πινάκων δευτέρου βαθμού με τυχαίους συντελεστές

$A = s * rand(n, n);\ B = s * rand(n, n);$

$\hookrightarrow$ έναρξη της λειτουργίας του μετρητή CPU time

$timer();$

$\hookrightarrow$ πολλαπλασιασμός δύο πολυωνυμικών πινάκων

$C = A * B;$

$\hookrightarrow$ καταγραφή της τιμής του μετρητή

$t = timer();$

$\hookrightarrow$ η επαναληπτική διαδικασία αξιολόγησης σταματάει αν περάσει ένα προκαθορισμένο χρονικό όριο

$if\ t > 200\ then$

$fprintf(f, "\%i \setminus t\ \%i \setminus t\ \%f \setminus n", d, n, t)$

$for\ n = n + st : st : fin,$

$t = 0;$

$fprintf(f, "\%i \setminus t\ \%i \setminus t\ \%i \setminus n", d, n, t)$

$end$
$mclose(f);$
$quit$

$end$

$\hookrightarrow$ τύπωση των τιμών του μετρητή

$fprintf(f, "\%i \backslash t \%i \backslash t \%f \backslash n", d, n, t)$

$\hookrightarrow$ τέλος της επαναληπτικής διαδικασίας

$end$

$mclose(f);$

$quit$

$\square$

**Σχολιασμός** Τα αρχεία αξιολόγησης -benchmark scripts- και στα τρία πολυωνυμικά πακέτα λογισμικού βασίζονται στη προαναφερθείσα σχεδίαση. Υπάρχουν διαφορές που αντιστοιχούν στην επιμέρους διαδικασία που αξιολογείται όπως και στις ιδιαιτερότητες που εμφανίζει το κάθε πακέτο.

Στην περίπτωση των πολυωνυμικών εξισώσεων στο POLYNOMIAL TOOLBOX υπάρχει ο μετρητής error πάνω στην ακρίβεια του αποτελέσματος εκτός από τους CPU time, MFLOPS.

Η εισαγωγή των δεδομένων προς επεξεργασία (input) γίνεται με μία αυτόματη διαδικασία η οποία δημιουργεί πολυωνυμικούς πίνακες δευτέρου βαθμού με τυχαίους συντελεστές. Ανάλογα με την διαδικασία που αξιολογείται θα πρέπει οι εισαγώμενοι πίνακες να τηρούν ορισμένες προϋποθέσεις ή συγκεκριμένες αλγεβρικές ιδιότητες. Για παράδειγμα στην διμερή συμμετρική εξίσωση πολυωνυμικών πινάκων

$$A'(s) \cdot X(s) + X'(s) \cdot A(s) = B(s)$$

ο πίνακας $B(s)$ πρέπει να ικανοποιεί την εξίσωση $B(s) = B(s) \cdot B'(s)$ δηλαδή να είναι συμμετρικός.

Στην περίπτωση της φασματικής παραγοντοποίησης (spectral factorization) το POLYNOMIAL TOOLBOX προσφέρει έναν κώδικα επίλυσης του συγκεκριμένου προβλήματος (εντολή spf) ο οποίος δίνει λύση και στο (J - spectral factorization) πρόβλημα. Οπότε έγκειται στον εισαγώμενο πίνακα - input - το πρόβλημα που θα λυθεί.

$$X' \cdot X = A$$

$$X' \cdot J \cdot X = A$$

Αν ο εισαγώμενος πίνακας είναι συμμετρικός θα οδηγήσει στην πρώτη εξίσωση ενώ εάν είναι της μορφής $B' \cdot J \cdot B$ όπου $J$ είναι ένας $diag[1, 1, ..., -1, 1]$ διαγώνιος πίνακας θα οδηγήσει στη δεύτερη.

### 2.2.3 3η κατηγορία αρχείων

Η 3η κατηγορία των αρχείων που αποτελούν το λογισμικό αξιολόγησης POLYBENCH είναι προγράμματα scripts γραμμένα στη γλώσσα PYTHON. Ο ρόλος τους είναι να διαβάζουν τα text αρχεία των αποτελεσμάτων του benchmark και να τα παρουσιάζουν σε μορφή πινάκων και γραφικών παραστάσεων. Η PYTHON όπως και η συγγενής της γλώσσα προγραμματισμού PERL διαθέτει μεγάλη δυνατότητα επεξεργασίας κειμένων μέσω των regular expressions -re-. Οπότε, γίνεται η εισαγωγή της βιβλιοθήκης re σε κάθε PYTHON script.

Η συγκεκριμένη γλώσσα είναι web-oriented και open source. Ακόμα προγράμματα γραμμένα σε αυτή μπορούν πολύ εύκολα να μεταφέρονται μεταξύ λειτουργικών συστημάτων. Υπάρχουν πολλά πακέτα γραμμένα για την γλώσσα PYTHON τα οποία επεκτείνουν τις ήδη μεγάλες δυνατότητές της. Η πλατφόρμα POLYBENCH χρησιμοποιεί τα πακέτα PYLAB και KARRIGELL.

Το PYLAB είναι ένα 2D λογισμικό γραφικών το οποίο κάνει γραφικές παραστάσεις σε στυλ MATLAB. Οι γραφικές παραστάσεις αυτές είναι των CPU time,Size και MFLOPS, Size, δηλαδή των δύο βασικών μετρητών με βάση το αυξανόμενο μέγεθος των τετραγωνικών πολυωνυμικών πινάκων. Είναι χαρακτηριστικό ότι οι εντολές αυτές παρόλο που είναι γραμμένες σε PYTHON και συγκεκριμένα σε μια βιβλιοθήκη της PYLAB, μοιάζουν πολύ με τις αντίστοιχες του MATLAB. Οπότε, το πρόγραμμα διαβάζει τα αρχεία που προκύπτουν από τα MATLAB, MATHEMATICA, SCILAB scripts τα οποία έχουν την πληροφορία σε μορφή στυλών.

**Αρχεία επεξεργασίας σε PYTHON.** Ακολουθεί παρουσίαση αρχείου .py.

$\hookrightarrow$ εισαγωγή των απαραίτητων βιβλιοθηκών re, pylab

*import re*

*from pylab import* $\star$

$\hookrightarrow$ ορισμός ενός αριθμού σε κάθε σειρά. Η κάθε σειρά περιέχει τρεις αριθμούς

$number = "(\backslash d\star\backslash.\{0,1\}\backslash d\star)"$

$pattern = re.compile('\%(NUM)s\backslash s+\%(NUM)s\backslash s+\%(NUM)s'$
$\%\{'NUM' : number\})$

$\hookrightarrow$ διάβασμα των γραμμών του αρχείου που προέκυψε από το MATLAB benchmark script

$datalines = open('matlab.out','r').readlines()$

$d1 = []$

$\hookrightarrow$ βάζοντας τα αποτελέσματα από το MATLAB σε μορφή λίστας

$for\ n\ in\ datalines :$

$line = n.strip()$

$if\ len(line) :$

$if\ 'degree'\ in\ line :\ continue$

$numbers = pattern.search(line)$

$if\ numbers :$

$d1.append(numbers.groups())$

↪ διάβασμα των γραμμών του αρχείου που προέκυψε από το MATHEMATICA benchmark script

$datalines = open('mathematica.out','r').readlines()$

$d2 = []$

↪ βάζοντας τα αποτελέσματα από το MATHEMATICA σε μορφή λίστας

$for\ n\ in\ datalines:$

$line = n.strip()$

$if\ len(line):$

$if\ 'degree'\ in\ line:\ continue$

$numbers = pattern.search(line)$

$if\ numbers:$

$d2.append(numbers.groups())$

↪ διάβασμα των γραμμών του αρχείου που προέκυψε από το SCILAB benchmark script

$datalines = open('scilab.out','r').readlines()$

$d3 = []$

↪ βάζοντας τα αποτελέσματα από το SCILAB σε μορφή λίστας

$for\ n\ in\ datalines:$

$line = n.strip()$

$if\ len(line):$

$if\ 'degree'\ in\ line:\ continue$

$numbers = pattern.search(line)$

$if\ numbers:$

$d3.append(numbers.groups())$

$\hookrightarrow$ μετατροπή των αριθμητικών αναπαραστάσεων από strings σε floats - αριθμούς κινητής υποδιαστολής για δημιουργία γραφικών παραστάσεων

$p11 = map(float, [x[1]\ for\ x\ in\ d1[1:]])$

$p12 = map(float, [x[2]\ for\ x\ in\ d1[1:]])$

$p10 = map(float, [x[0]\ for\ x\ in\ d1[1:]])$

$p21 = map(float, [x[1]\ for\ x\ in\ d2[1:]])$

$p22 = map(float, [x[2]\ for\ x\ in\ d2[1:]])$

$p31 = map(float, [x[1]\ for\ x\ in\ d3[1:]])$

$p32 = map(float, [x[2]\ for\ x\ in\ d3[1:]])$

$figure(1)$

$subplot(111, axisbg =' \sharp B1D3EC')$

$plot(p11, p12, 'rs', p21, p22, 'b', p31, p32, 'go')$

$legend(('Matlab{-}Poly.T.', 'Mathematica{-}Poly.', 'Scilab'),\ 'upper\ left')$

$xlabel('Size(nxn) - -- >')$

$$ylabel('ComputationalTime(s) - -- >')$$

$$t1 = title('CPUTIME',\ color =' blue')$$

$$savefig('graph1', dpi = 100)$$

$$savefig('graph2', dpi = 50)$$

$$figure(2)$$

$$subplot(111, axisbg =' \sharp B1D3EC')$$

$$plot(p11, p10,' rs')$$

$$legend(('Matlab - Poly.T.','' ),\ 'upper\ left')$$

$$xlabel('Size(nxn) - -- >')$$

$$ylabel('Mflops - -- >')$$

$$t1 = title('MFLOPS',\ color =' blue')$$

$$savefig('graph3', dpi = 100)$$

$$savefig('graph4', dpi = 50)$$

Το πακέτο KARRIGELL είναι ένα λογισμικό που υποστηρίζει την γλώσσα PYTHON και δημιουργεί δυναμικές ιστοσελίδες στυλ PHP, ASP. Συγκεκριμένα, αρχεία τύπου .pih δηλαδή Python In Html χρησιμοποιούνται για να φιλοξενήσουν τα αποτελέσματα της αξιολόγησης. Τα αποτελέσματα παρουσιάζονται με την μορφή πινάκων που δημιουργούνται από αρχεία (.pih) όπως το παραπάνω πρόγραμμα.

**Αρχεία δυναμικών σελίδων σε PYTHON.** Ακολουθεί παρουσίαση αρχείου (.pih) στην περίπτωση του πολλαπλασιασμού πολυωνυμικών πινάκων.

$< head >$

$< title > PolyBench\ --\ Multiplication < /title >$

$< base\ target = "\_blank" >$

$< link\ rel = "stylesheet"\ type = "text/css"\ href = "general1.css" / >$

$< /head >$

$< body >$

$< h1 > multiplication\ --\ A(s) \cdot B(s) < /h1 >$

$< hr\ noshade\ size = 1 >$

$< a\ href = "graphs/MaMatSc1.png" >$

$< img\ class =' cpu'\ src = "graphs/MaMatSc1sm.png" / >< /a >$

$< a\ href = "graphs/FMat1.png" >$

$< img\ class = "mflops"\ src = "graphs/FMat1sm.png" >< /a >$

$< \%Include("table1.py")\% >$

< table width = "50%" border = 1 >

< tr >

< th rowspan = 2 > Matrix < br > Size < /th >

< th colspan = 2 > MATLAB < br > PolynomialToolbox < /th >

< th > MATHEMATICA < br > Polynomial < /th >

< th > SCILAB < /th >

< tr >

< th > MFLOPS < /th >

< th > CPUtime < br > (sec) < /th >

< th > CPUtime < br > (sec) < /th >

< th > CPUtime < br > (sec) < /th >

< /tr >

< indent >

< %for i in range(len(p1)) : % >

< tr >

< td align = "center" >< % print p1[i] % >< /td >

< td align = "center" >< % print p0[i] % >< /td >

< td align = "center" >< % print p2[i] % >< /td >

< td align = "center" >< % print p4[i] % >< /td >

< td align = ”center” >< % print p6[i] % >< /td >

< /tr >

< /indent >

< /table >

< div class = ”library” >

< p  align = ”center” >

< b >< var > LibraryFunctions :< /var >< /b >

< var >< br > < /var >< /p >

< ul type = square >

< li >< var > MATLAB − Polynomial Toolbox < /var >
< ul >< li >< var > mtimes() < /var >< /ul >
< li >< var > MATHEMATICA − Polynomial < /var >< ul >
< li >< var > Dot() < /var >< /ul >

< li >< var > SCILAB < /var >< ul >

< li >< var > star(∗) < /var >< /ul >

< /ul >< /div >

< hr class = ”end” >

< /body >

# Κεφάλαιο 3

# ΑΝΑΛΥΣΗ ΑΞΙΟΛΟΓΗΣΗΣ

## 3.1 Γενικά

Ο δεύτερος στόχος της παρούσας εργασίας είναι η παρουσίαση αποτελεσμάτων αξιολόγησης (benchmarking) μιας σειράς κωδικών (solvers) που βρίσκονται σε τρία γνωστά πακέτα αριθμητικών πολυωνυμικών μεθόδων: POLYNOMIAL TOOLBOX, POLYNOMIAL PACKAGE, SCILAB. Οι δώδεκα (solvers) που αξιολογούνται απαντώνται εξαιτερικά συχνά στην σχεδίαση συστημάτων ελέγχου, στη μοντελοποίηση και στην επεξεργασία σήματος. Οι μετρητές (counters) η απόδοση των οποίων κρίνει την αποδοτικότητα των κωδικών - προγραμάτων είναι τρεις για το περιβάλλον του MATLAB και ένας για MATHEMATICA ,SCILAB. Η εκλογή των μετρητών σχετίζεται με τη δυνατότητα που προσφέρει η κάθε υπολογιστική πλατφόρμα.

Σε όλα τα εξεταζόμενα πακέτα υπήρχε η εντολή μέτρησης της χρονικής διάρκειας που δουλεύει η Κεντρική Μονάδα Επεξεργασίας ή ο μετρητής CPU time. Ο συγκεκριμένος μετρητής είναι ο χαρακτηριστικότερος όλων και συναντάνται συχνά σε κάθε είδους benchmark υπολογιστών. Τα συμπεράσματα που μπορούν να προκύψουν από την απόδοσή του είναι εύκολα ερμηνεύσιμα. Ο κώδικας θα πρέπει να τρέχει σε όσο το δυνατό λιγότερη CPU time. Η CPU time σχετίζεται με την Clock time που είναι η πραγματική ώρα που χρειάζεται για να εκτελεσθεί μια διαδικασία. Ο δεύτερος μετρητής εξαρτάται και από άλλους τρίτους παράγοντες εκτός την ποιότητα του κώδικα και της CPU όπως είναι η γενική κατάσταση του Η/Υ, ο ελεύθερος χώρος στον δίσκο, .... Συμπερασματικά, ο μετρητής CPU time αποτελεί στοιχειώδης εντολή κάθε πακέτου και για αυτό η παρουσία του είναι απαραίτητη σε κάθε μορφή αξιολόγησης προγραμμάτων.

Η αξιολόγηση προγραμμάτων Η/Υ απαιτεί και άλλες πληροφορίες πέρα από τη χρονική διάρκεια εκτέλεσης τους. Για να επιτευχθεί αυτό χρειάζεται και η κατάλληλη τεχνογνωσία δηλαδή κάποιο λογισμικό το οποίο να προσφέρει αυτή τη δυνατότητα.

Το λογισμικό αξιολόγησης PAPI το οποίο έχει αναπτυχθεί σε ινστιτούτο των Η.Π.Α. είναι ένα εργαλείο το οποίο επιτρέπει χρησιμοποιώντας διάφορες μεθόδους την αξιολόγηση benchmarking κωδικών γραμμένων σε Fortran, C κυρίως, αλλά παράλληλα προσφέρει μία βιβλιοθήκη αξιολόγησης ειδικά σχεδιασμένη για το περιβάλλον του MATLAB. Η εισαγωγή της συγκεκριμένης βιβλιοθήκης δίνει τη δυνατότητα της μέτρησης του αριθμού των MFLOPS - Mega Floating Operations Second που αντιστοιχεί στις διεργασίες κινητής υποδιαστολής ανά δευτερόλεπτο που πραγματοποιεί η αριθμητική μονάδα που βρίσκεται στην CPU κάθε Η/Υ.

Ο συγκεκριμένος μετρητής προσφέρει σημαντικές πληροφορίες για την απόδοση της Κεντρικής Μονάδας Επεξεργασίας κατά την διάρκεια εκτέλεσης ενός κώδικα. Συγκεκριμένα, ο ρυθμός των MFLOPS δηλώνει την αξιοποίηση ή μη της υπολογιστικής δυνατότητας της CPU κατά το παρακάτω μοντέλο: Ο μέσος όρος των μετρούμενων MFLOPS αυξάνει όσο αυξάνεται το μέγεθος της εισαγώμενης προς επεξεργασία πληροφορίας. Η συγκεκριμένη αύξηση δεν είναι αέναη αλλά σταματάει πέρα από κάποιο συγκεκριμένο υπολογιστικό όριο. Η σταθεροποίηση αυτή δηλώνει την μέγιστη ικανότητα επεξεργασίας που προσφέρει η CPU στον συγκεκριμένο κώδικα που εκτελείται. Δυστηχώς, η παραπάνω εικόνα είναι θεωρητική και δεν απαντάται στις πρακτικές εφαρμογές όπου στην πλειάδα των περιπτώσεων ο ρυθμός των MFLOPS μειώνεται δραστικά εώς και στα χαμηλότερα δυνατά επίπεδα σε ορισμένες περιπτώσεις. Η εξήγηση για την παραπάνω διακύμανση είναι η μη αποδοτική λειτουργία των cache memories: L1,L2,L3 όπου συγκεντρώνεται τμηματικά η πληροφορία (data) προτού επεξεργαστεί στην CPU. Δηλαδή ο κώδικας δεν χειρίζεται σωστά την αποθήκευση και διοχεύτευση της πληροφορίας στη κεντρική μονάδα επεξεργασίας του Η/Υ.

Οι κώδικες αξιολογούνται εκτός από την υπολογιστική τους ικανότητα και για την ακρίβεια του αποτελέσματος όταν εκτελούνται. Αυτή η πληροφορία είναι πολύ σημαντική και παίζει σπουδαίο ρόλο στην αξιολόγηση του κώδικα (solver). Ο πιο συνήθεις τρόπος μελέτης του βαθμού του υπολογιστικού λάθους εντοπίζεται στους κώδικες που δίνουν λύση σε εξισώσεις. Η ακρίβεια μπορεί να μετρηθεί υπολογίζοντας την σχετική νόρμα του υπολοίπου πολλαπλασιασμένη με μια σταθερά μεγάλου μεγέθους για πιο ευκρινή αποτελέσματα. Για παράδειγμα σε μία εξίσωση της μορφής

$$A \cdot X = B$$

ο μετρητής error είναι

$$10^{10} \cdot [norm(A \cdot X - B)/norm(A)]$$

Η τιμή του μετρητή αυτού θα πρέπει να είναι αμελητέα (μικρότερη της τάξης του $10^{-5}$ ) για να είναι το αποτέλεσμα αποδεκτό χωρίς ιδιαίτερη απόκλιση από το θεωρητικό αντίστοιχο.

## 3.2 Σχολιασμός

Ο πολλαπλασιασμός πινάκων θεωρείται από τις καλύτερες διαδικασίες στην επιστήμη των υπολογιστών για την αξιολόγηση της υπολογιστικής ικανότητας των κεντρικών μονάδων επεξεργασίας -CPU- των Η/Υ. Συγκεκριμένα αυτή η διεργασία μπορεί να φτάσει στο 60% του μέγιστου ρυθμού MFLOPS της CPU.

Παρόμοια απόδοση αναμένεται στην περίπτωση των πολυωνυμικών πινάκων. Με βάση τη γραφική παράσταση 3.1 παρατηρείται ότι η απόδοση των MFLOPS φτάνει τον αριθμό των 1280 όπου το μέγιστο δυνατό της συγκεκριμένης CPU είναι περίπου 2180 MFLOPS. Οπότε, επαληθεύεται η θεωρητική προσέγγιση. Στο συγκριτικό διάγραμμα της CPU time διακρίνεται ότι ο κώδικας που προσφέρει το POLYNOMIAL PACKAGE στην πλατφόρμα της MATHEMATICA κατέχει την πρώτη θέση στην ταχύτητα εύρεσης του αποτελέσματος. Ακολουθεί στην απόδοση το POLYNOMIAL TOOLBOX το οποίο προσφέρει και αυτό ένα ανταγωνιστικό κώδικα ο οποίος φτάνει την CPU στα υπολογιστικά της όρια όπως δείχνουν τα MFLOPS. Το διάγραμμα δείχνει την καλή χρήση των μνημών της CPU αφού τα MFLOPS παρουσιάζουν σταθερή διακύμανση. Το μέγεθος των πολυωνυμικών πινάκων, μέσα στην επαναληπτική διαδικασία ξεκινά από τετραγωνικό πίνακα $100 \times 100$ και φτάνει την τιμή των $2000 \times 2000$ που αποτελεί ένα πολύ μεγάλο μέγεθος εισαγώμενης πληροφορίας input data και συνεπώς ένα δύσκολο εγχείρημα για τους solvers. Ο κώδικας που προσφέρει το πακέτο SCILAB όπως προκύπτει από το διάγραμμα είναι εκτός ανταγωνισμού και συνεπώς παύει η αξιολόγηση του από ένα σημείο και πέρα. Οι συναρτήσεις βιβλιοθήκης που χρησιμοποιήθηκαν είναι:

- mtimes( ) - POLYNOMIAL TOOLBOX -
- Dot[ ] - POLYNOMIAL PACKAGE -
- ( $\star$ ) - SCILAB -

Η περίπτωση της πρόσθεσης πολυωνυμικών πινάκων δεν φημίζεται για την υπολογιστική της ικανότητα οπότε και αναμένεται χαμηλός ρυθμός των MFLOPS στην CPU όπως φαίνεται και στο διάγραμμα 3.2 όπου ο ρυθμός είναι της τάξης του 5% του μέγιστου ρυθμού των MFLOPS. Η διακύμανση τους χαρακτηρίζεται σταθερή και ικανοποιητική όπως κοντά στην ιδανική χωρίς ιδιαίτερες αλλαγές. Το συγκριτικό διάγραμμα της απόδοσης της CPU δείχνει ότι το SCILAB διαθέτει τον ταχύτερο κώδικα για πρόσθεση πολυωνυμικών πινάκων σε σχέση με τα πακέτα POLYNOMIAL TOOLBOX, MATHEMATICA. Γενικά η συγκεκριμένη διαδικασία δεν είναι ιδιαίτερα απαιτητική σε υπολογιστική ισχύ και δεν ενδείκνυται για εξαγωγή συμπερασμάτων. Οι συναρτήσεις βιβλιοθήκης που χρησιμοποιήθηκαν είναι:

- plus( ) - POLYNOMIAL TOOLBOX -
- Plus[ ] - POLYNOMIAL PACKAGE -
- ( + ) - SCILAB -

Η επόμενη αξιολόγηση περιλαμβάνει τους κώδικες που αφορούν την εύρεση ορίζουσας και αντιστρόφου, δύο σχετικά συγγενείς διαδικασίες των υπολογιστικών μαθηματικών. Στην περίπτωση αυτή το POLYNOMIAL PACKAGE για τη MATHEMATICA δεν έχει ανταγωνιστικό solver και η αξιολόγηση του σταματάει σύντομα αμέσως μετά την έναρξη της διαδικασίας σε μικρό μέγεθος τετραγωνικού πίνακα. Οι δύο ανταγωνιστικοί κώδικες είναι του POLYNOMIAL TOOLBOX και του SCILAB με τον πρώτο να υπερέχει του δεύτερου στην εύρεση του αντιστρόφου πολυωνυμικού πίνακα και το ανάποδο όσον αναφορά την απόδοση του μετρητή CPU time στην περίπτωση της ορίζουσας όπου ο κώδικας του POLYNOMIAL TOOLBOX παρουσιάζει μία αστάθεια στην απόδοσή του παρόλο που έχει σχετικά παρόμοια απόδοση με αυτόν του SCILAB. Η αστάθεια που φαίνεται στο διάγραμμα των MFLOPS στην ορίζουσα πολυωνυμικού πίνακα και η ραγδαία μείωση του ρυθμού των MFLOPS παρά την συνεχή αύξηση του εισαγώμενου μεγέθους (input data) δηλώνουν το πρόβλημα που υπάρχει στην περίπτωση του κώδικα του POLYNOMIAL TOOLBOX σε αντίθεση με την εξαιρετική απόδοση στη περίπτωση της εύρεσης αντιστρόφου πάλι από το ίδιο πολυωνυμικό πακέτο. Τα παραπάνω διακρίνονται στα διαγράμματα 3.3 και 3.4. Οι συναρτήσεις βιβλιοθήκης που χρησιμοποιήθηκαν είναι:

- det( ), inv( ) - POLYNOMIAL TOOLBOX -
- Det[ ], Inverse[ ] - POLYNOMIAL PACKAGE -
- determ( ), coffg( ) - SCILAB -

Οι επόμενοι αξιολογημένοι κώδικες αφορούν εξισώσεις πολυωνυμικών πινάκων και απλών πολυωνύμων. Σε πολλές περιπτώσεις απαιτείται η χρησιμοποίηση πολυωνύμων στον έλεγχο των οποίων οι υπολογιστικές διεργασίες είναι διαφορετικές από τους αντίστοιχους πίνακες και σε κάθε περίπτωση λιγότερο πολύπλοκες. Ο λόγος της παρουσίασης τους στην παρούσα εργασία είναι η ανάδειξη της φύσης τους έναντι των πολυωνυμικών πινάκων. Για παράδειγμα, για τον πολλαπλασιασμό των πολυωνύμων χρειάζεται μόνο η σύγκριση του βαθμού τους σε αντίθεση με την περίπτωση των πινάκων όπου απαιτείται μια σειρά από διαδικασίες.

Η πολυωνυμική εξίσωση πολυωνυμικών πινάκων και η συγγενής της διοφαντική αποτελούν τις σπουδαιότερες εξισώσεις που παρουσιάζονται στη πολυωνυμική θεωρία ελέγχου. Το POLYNOMIAL TOOLBOX προσφέρει δύο θεωρητικά διαφορετικούς κώδικες για την λύση των παραπάνω εξισώσεων που πρακτικά αποτελούν ένα κώδικα αφού ο κώδικας της διοφαντικής εξίσωσης καλεί τον αντίστοιχο της γραμμικής πολυωνυμικής εξίσωσης. Η ταύτιση αυτή προέρχεται από το γεγονός ότι η διοφαντική πολυωνυμική εξίσωση μπορεί πολύ εύκολα να γραφεί σαν γραμμική εξίσωση πινάκων και επομένως να αντιμετοπιστεί με παρόμοιο τρόπο. Η παρατήρηση αυτή μπορεί να δικαιλογηθεί από τα διαγράμματα 3.5 και 3.6 που δείχνουν ότι ο ρυθμός των MFLOPS είναι ίδιος. Το αποτέλεσμα του benchmarking δηλώνει τη μη αποδοτικό χειρισμό του συγκεκριμένου κώδικα για πίνακες μεγάλους μεγέθους (τάξης $200 \times 200$) αφού διαπιστώνεται πολύ μεγάλη μείωση του ρυθμού των MFLOPS. Αντίθετα, το πακέτο λογισμικού POLYNOMIAL PACKAGE διαθέτει δύο κώδικες εξαιρετικής απόδοσης από ότι δείχνουν τα διαγράμματα του μετρητή CPU time σε σύγκριση με αυτούς του POLYNOMIAL TOOLBOX. Ακόμα ένα σημαντικό πλεονέκτημα του πακέτου της MATHEMATICA είναι ότι διαθέτει για την εύρεση της βαθμού της λύσης της γραμμικής εξίσωσης δυαδική αναζήτηση με βάρος (weighted search) σε αντίθεση με το POLYNOMIAL TOOLBOX που έχει απλή δυαδική αναζήτηση κάτι που επιβραδύνει ακόμα περισσότερο την εξεύρεση λύσης. Για λόγους αξιοπιστίας των αποτελεσμάτων στο συγκεκριμένο benchmark που γίνεται αντικείμενο σχολιασμού ήταν προκαθορισμένος ο βαθμός λύσης των εξισώσεων. Τέλος, από το διάγραμμα 3.13 φαίνεται ότι η διακύμανση του μετρητή του λάθους (error) στην προσέγγιση της επιζητούμενης λύσης είναι σε πολύ χαμηλά επίπεδο οπότε και η λύση του POLYNOMIAL TOOLBOX είναι σχεδόν απόλυτα σωστή και στις δύο περιπτώσεις των εξισώσεων. Οι συναρτήσεις βιβλιοθήκης που χρησιμοποιήθηκαν είναι: axb( ) - POLYNOMIAL TOOLBOX - και AXBSolve[ ] - POLYNOMIAL PACKAGE.

Παρόμοια αποτελέσματα αναδείχτηκαν στην περίπτωση της διοφαντικής εξίσωσης πολυωνύμων όπου το POLYNOMIAL TOOLBOX δεν διαθέτει ξεχωριστό κώδικα αλλά χρησιμοποιεί τον υπάρχοντα από την αντίστοιχη περίπτωση των πολυωνυμικών πινάκων βάζοντας βαθμό του πίνακα την μονάδα. Συγκεκριμένα, το πακέτο αυτό δεν λαμβάνει υπόψη την απλότητα της πολυωνυμικής περίπτωσης σε σύγκριση με αυτήν των πινάκων οπότε είναι αναμενόμενη η μικρή αποδοτικότητα του συγκεκριμένου solver. Σε πλήρη αντίθεση το POLYNOMIAL PACKAGE της MATHEMATICA διαθέτει ειδικό κώδικα για λύση πολυωνυμικής διοφαντικής εξίσωσης και όπως φαίνεται στο διάγραμμα 3.7 είναι εξαιτερικά ταχύς. Η πολυωνυμική βιβλιοθήκη του πακέτου SCILAB προσφέρει έναν solver πολύ μειωμένης απόδοσης όπως φαίνεται από τη συγκριτική γραφική παράσταση της CPU time. Από το πρώτο διάγραμμα 3.14 προκύπτει ότι η ακρίβεια της λύσης είναι σχεδόν απόλυτη παρόμοια με αυτή της περίπτωσης των πινάκων. Οι συναρτήσεις βιβλιοθήκης που χρησιμοποιήθηκαν στη μελέτη των διοφαντικών εξισώσεων είναι:

- axbyc( ) - POLYNOMIAL TOOLBOX -
- DESolve[ ] , AXBYCSolve[ ] - POLYNOMIAL PACKAGE -
- diophant( ) - SCILAB -

Στη συνέχεια αξιολογούνται οι solvers για τη λύση της διμερούς συμμετρικής εξίσωσης πολυωνυμικών πινάκων και της περίπτωσης των απλών πολυωνύμων. Μόνο το POLYNOMIAL TOOLBOX, το οποίο είναι το πιο πλήρες υπολογιστικό πακέτο στην πολυωνυμική θεωρία ελέγχου, προσφέρει κώδικα για τη λύση και των δύο αυτών εξισώσεων ο οποίος παρουσιάζει σταθερή απόδοση στο ρυθμό των MFLOPS και CPU time όπως δείχνουν τα διαγράμματα 3.8 και 3.9. Παρόλο που η αποδοτικότητα του συγκεκριμένου κώδικα (είναι ο ίδιος για πολυωνυμικούς πίνακες και πολυώνυμα) κρίνεται ικανοποιητική, η ακρίβεια της λύσης υπολογισμένη από τον μετρητή Error (βλέπε διάγραμμα 3.14) δεν είναι καλή και παρουσιάζονται αρκετά σημαντικές αποκλίσεις. Ο κώδικας δεν προσφέρει ακριβή λύση στην εξίσωση των πολυωνυμικών πινάκων αφού η ακρίβεια του αγγίζει τη μονάδα. Στην περίπτωση των απλών πολυωνύμων το υπολογιζόμενο λάθος είναι σε επιτρεπτά όρια (βλέπε διάγραμμα 3.15) με ακρίβεια της τάξης του $10^{-10}$. Όπως στην περίπτωση της διοφαντικής εξίσωσης, το συγκεκριμένο πακέτο διαθέτει έναν κώδικα για λύση της εξισώσης με πολυώνυμα ή πολυωνυμικούς πίνακες. Το πακέτο POLYNOMIAL PACKAGE διαθέτει κώδικα επίλυσης της διμερής συμμετρικής εξίσωσης πολυωνύμων ο οποίος παρουσιάζει εξαιρετική απόδοση στην ταχύτητα εύρεσης του αποτελέσματος συγκριτικά με τον συγκρινόμενο κώδικα (βλέπε διάγραμμα 3.9). Οι συναρτήσεις βιβλιοθήκης που χρησιμοποιήθηκαν είναι οι axxab() - POLYNOMIAL TOOLBOX και AXXABSolve[] - POLYNOMIAL PACKAGE.

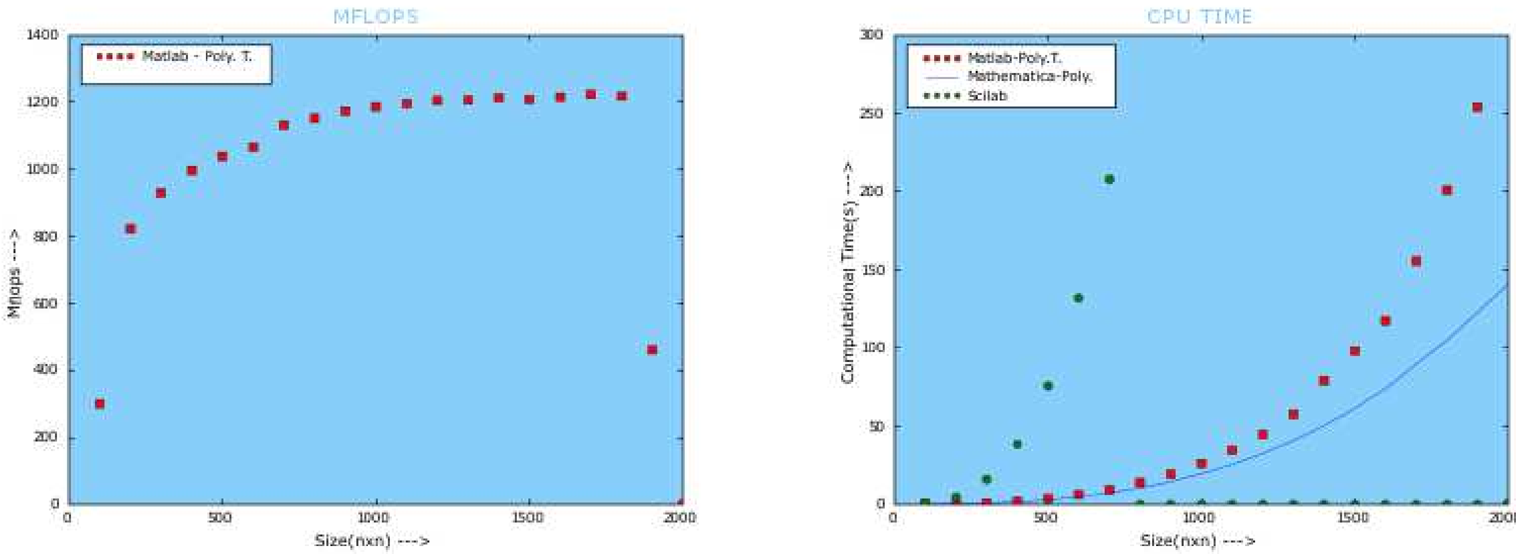
Οι φασματικές παραγοντοποιήσεις (spectral factorizations) παίζουν εξαιτερικά σημαντικό ρόλο στη σχεδίαση συστημάτων και στην επεξεργασία σήματος. Η λύση τους αποτελεί συνεχές αντικείμενο έρευνας και η ανάπτυξη κωδικών για την αυτόματη λήψη του συγκεκριμένου αποτελέσματος είναι εξαιρετικής σημασίας. Το POLYNOMIAL TOOLBOX προσφέρει έναν αξιόπιστο solver όπως φαίνεται από τα διαγράμματα απόδοσης 3.10 και 3.11 όπου ο ρυθμός των MFLOPS παρουσιάζει μία σταθερή διακύμανση. Η ακρίβεια της προσφερόμενης λύσης είναι πολύ σημαντική στη συγκεκριμένη περίπτωση. Τα διαγράμματα 3.15 και 3.16 δείχουν ότι ο solver παρουσιάζει μία αστάθεια στον ακριβή εντοπισμό της λύσης. Το πρόβλημα εντοπίζεται κυρίως στην περίπτωση της εξίσωσης $X'(s) \star J \star X(s) = A(s)$ όπου η ακρίβεια της λύσης φτάνει την τάξη $10^{-4}$ με μία συνεχόμενη πορεία που δηλώνει ότι η ακρίβεια θα μειώνεται δραστικά όσο μεγαλώνει το μέγεθος των πολυωνυμικών πινάκων. Η συνάρτηση βιβλιοθήκης που χρησιμοποιήθηκε είναι η spf() - POLYNOMIAL TOOLBOX.

Η τελευταία αξιολογημένη διαδικασία είναι η διαίρεση πινάκων η οποία είναι μία διαδικασία που εμφανίζεται πολύ συχνά στις πολυωνυμικές μεθόδους. Το POLYNOMIAL TOOLBOX και το SCILAB διαθέτουν τους σχετικούς κώδικες για λύση του συγκεκριμένου προβλήματος όπου το πρώτο πακέτο διαθέτει έναν solver πολύ ικανοποιητικής απόδοσης όπως δείχνει η διακύμανση των MFLOPS (διάγραμμα 3.12) αλλά και το συγκριτικό διάγραμμα της CPU time με τον solver του SCILAB ο οποίος για σχετικά μεγάλα μεγέθη εισαγωμένων πινάκων αδυνατεί να δώσει άμεσα λύση και η αξιολόγηση του σταματάει όπως δείχνει το διάγραμμα. Από το διάγραμμα 3.16 συμπεραίνεται ότι η ακρίβεια του αποτελέσματος είναι σε ικανοποιητικά μεγέθη (τάξης $10^{-7}$) για τον κώδικα του POLYNOMIAL TOOLBOX. Χρησιμοποιήθηκαν οι συναρτήσεις βιβλιοθήκης rdiv() - POLYNOMIAL TOOLBOX - και pdiv - SCILAB.

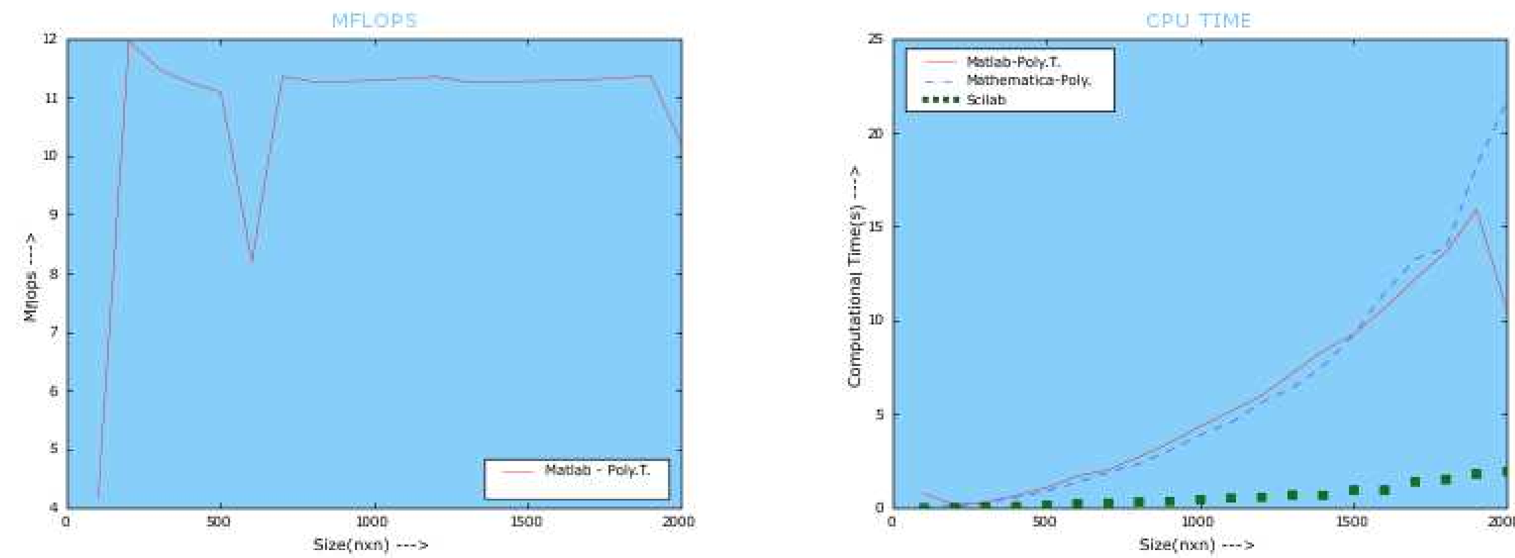
## 3.3 Αποτελέσματα

Ακολουθούν αποτελέσματα αξιολόγησης παρουσιασμένα σε γραφικές παραστάσεις των δώδεκα συναρτήσεων που περιλαμβάνει η πλατφόρμα POLYBENCH. Σε κάθε εξεταζόμενη διαδικασία αντιστοιχούν δύο διαγράμματα: το πρώτο αφορά τον ρυθμό των MFLOPS το οποίο μόνο στο περιβάλλον της υπολογιστικής γλώσσας MATLAB μπορεί να μετρηθεί και το δεύτερο το συγκριτικό διάγραμμα απόδοσης του μετρητή της CPU time μεταξύ των πακέτων που διαθέτουν solver για την αντίστοιχη διαδικασία. Επίσης, υπάρχουν διαγράμματα που αφορούν τον λεγόμενο μετρητή Error και σχετίζονται με την ακρίβεια των λύσεων των εξισώσεων που προσφέρουν οι κώδικες του POLYNOMIAL TOOLBOX.

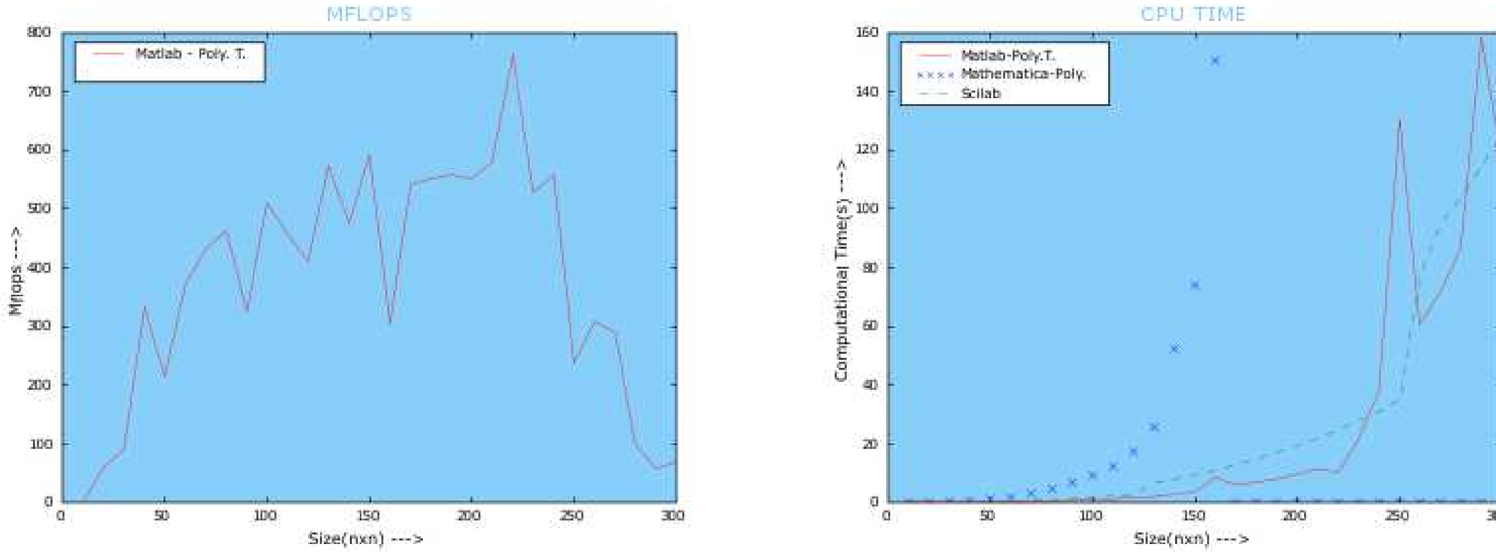
Ο υπολογιστής όπου έτρεξε η πλατφόρμα POLYBENCH είναι φορητού τύπου Intel(R) Pentium(R) M processor 1600 MHz 1MB L2 me 2181 MFLOPS μέγιστη λειτουργία της κεντρικής αριθμητικής μονάδας FPU - Floating Point Unit και κεντρική μνήμη 512 MB DDR - SDRAM.



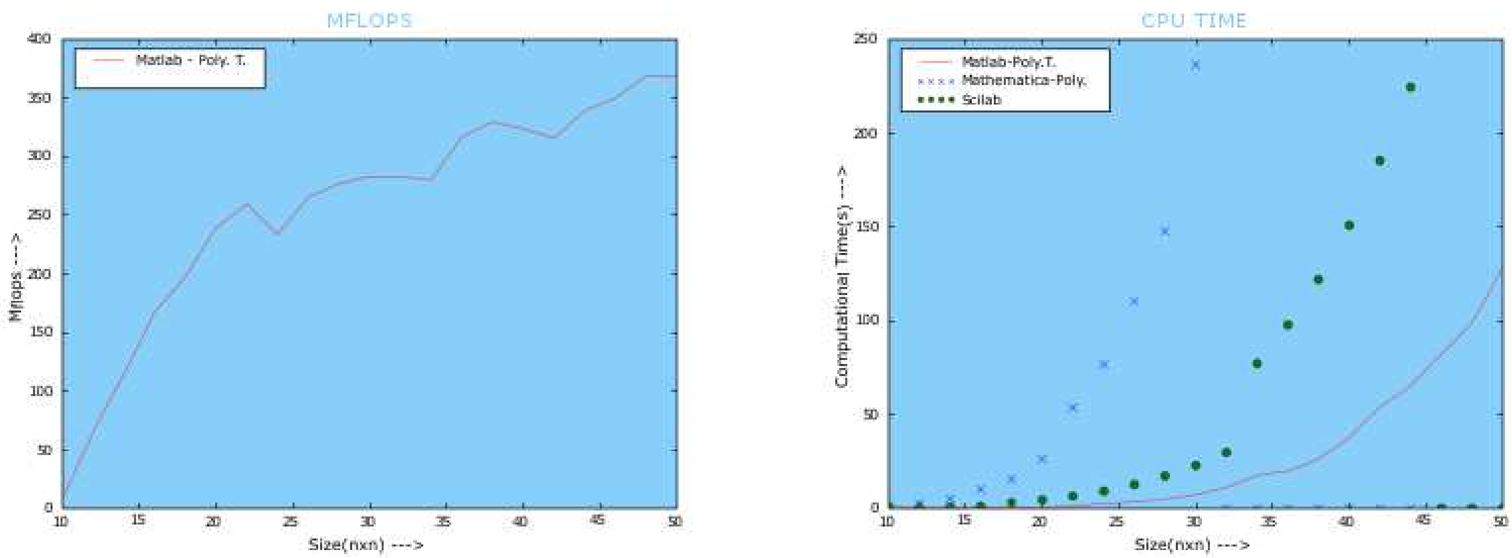

Σχήμα 3.1: Πολλαπλασιασμός πολυωνυμικών πινάκων δευτέρου βαθμού μιας μεταβλητής $\mathbf{A}(\mathbf{s}) \cdot \mathbf{B}(\mathbf{s})$.



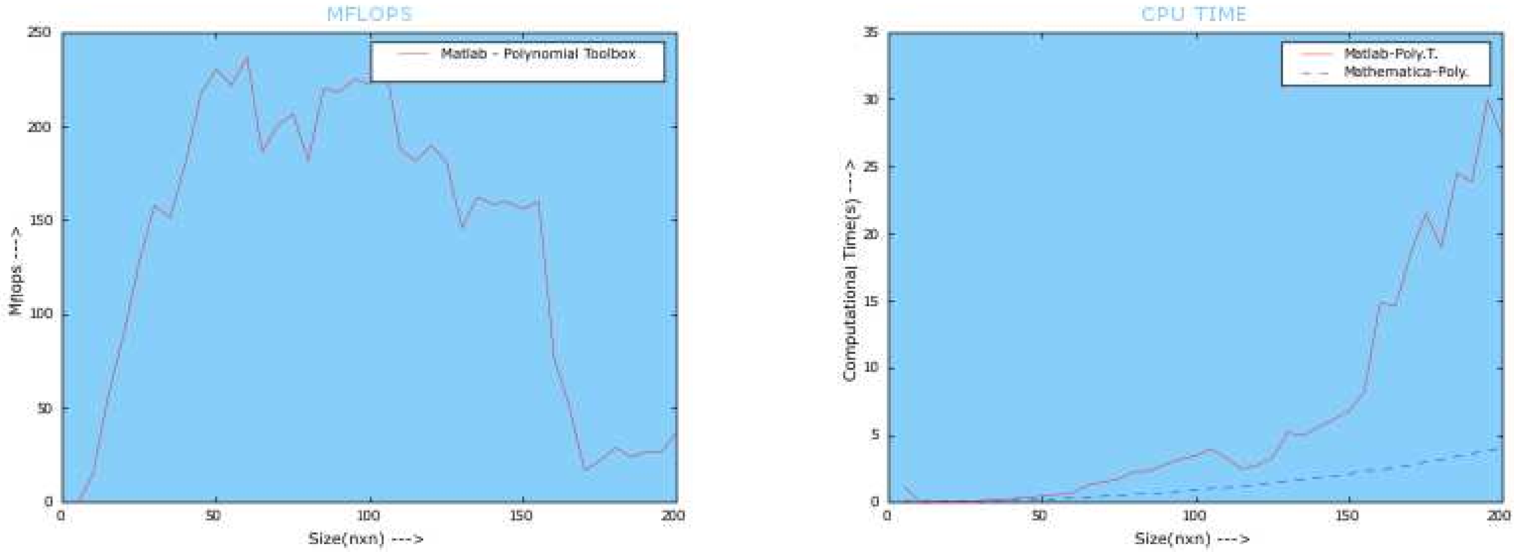

Σχήμα 3.2: Πρόσθεση πολυωνυμικών πινάκων δευτέρου βαθμού μιας μεταβλητής $\mathbf{A}(\mathbf{s}) + \mathbf{B}(\mathbf{s})$.



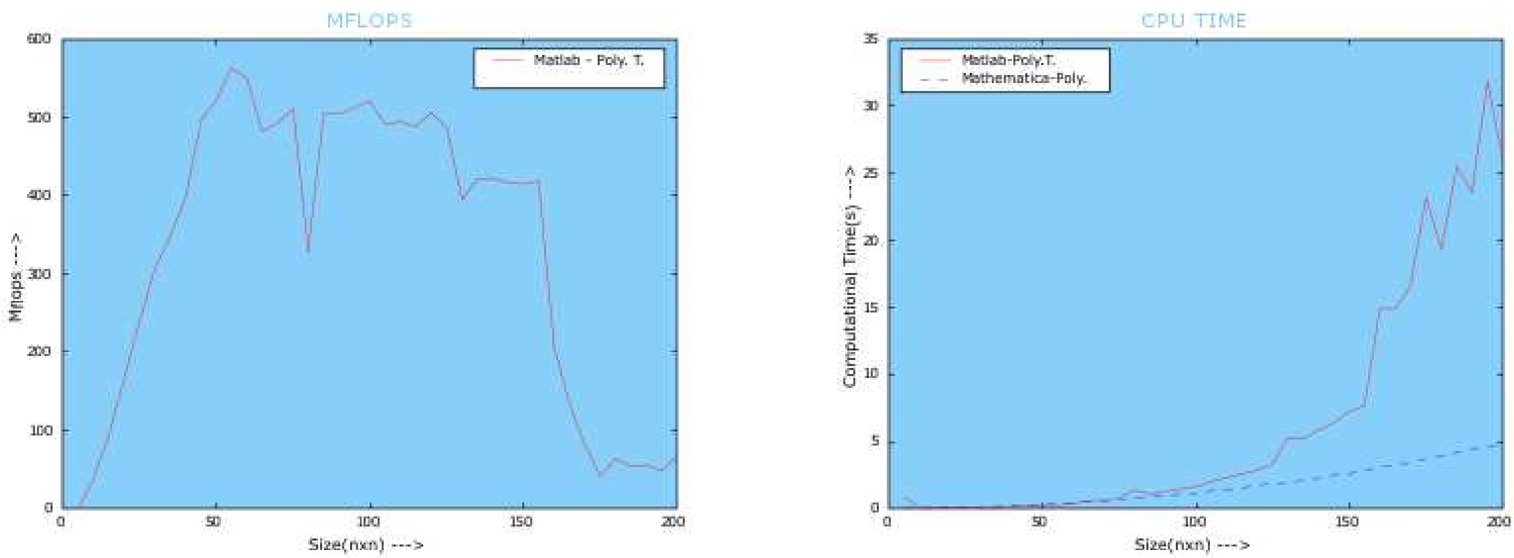

Σχήμα 3.3: Ορίζουσα πολυωνυμικού πίνακα δευτέρου βαθμού μιας μεταβλητής $\mathbf{Det}[\mathbf{A}(\mathbf{s})]$.



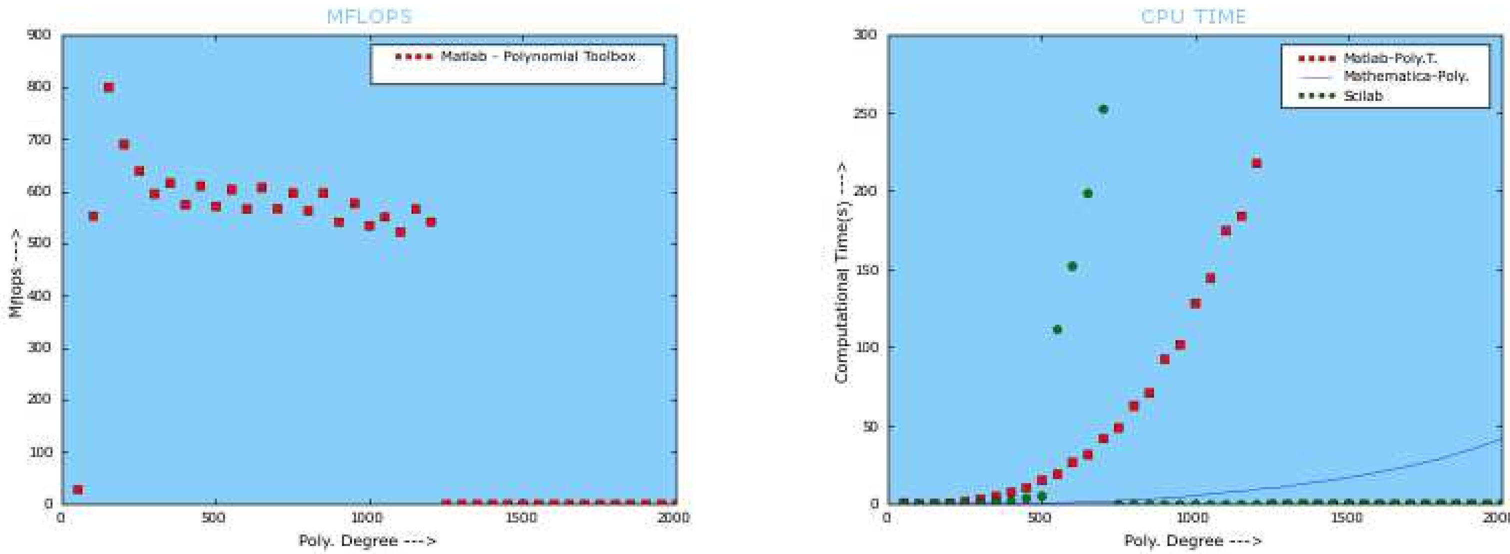

Σχήμα 3.4: Αντίστροφος πολυωνυμικού πίνακα δευτέρου βαθμού μιας μεταβλητής $\mathbf{Inv}[\mathbf{A}(\mathbf{s})]$.



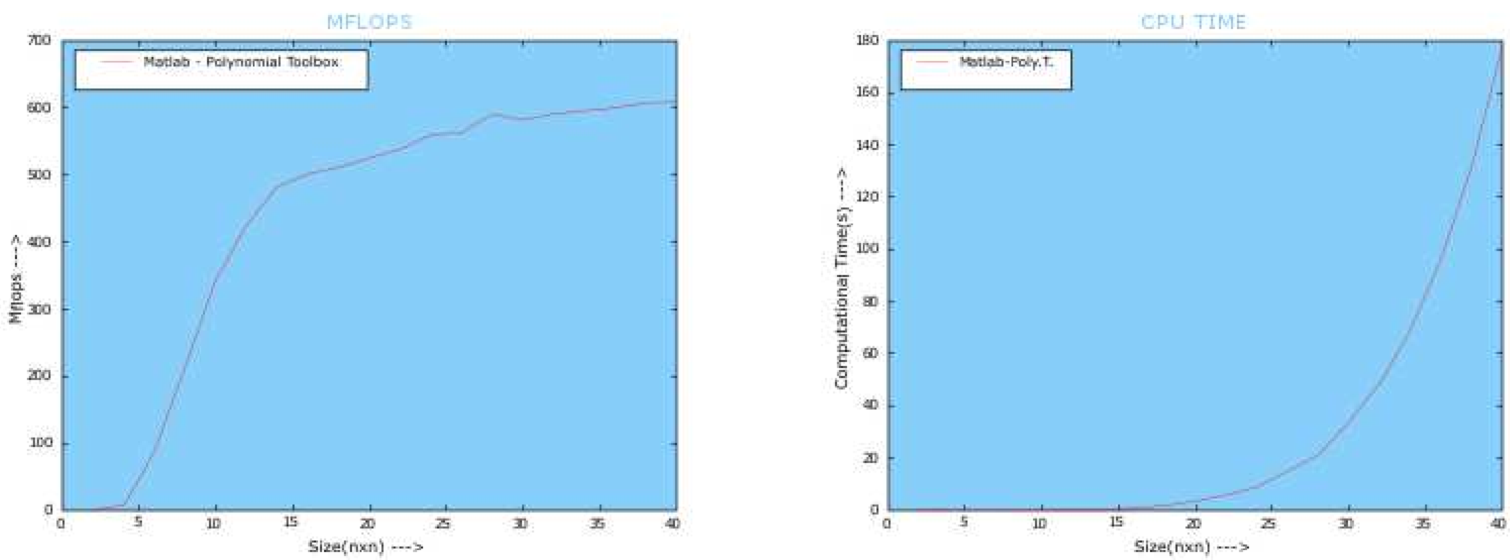

Σχήμα 3.5: Γραμμική εξίσωση πολυωνυμικών πινάκων δευτέρου βαθμού μιας μεταβλητής $\mathbf{A}(\mathbf{s}) \cdot \mathbf{X}(\mathbf{s}) = \mathbf{B}(\mathbf{s})$ .



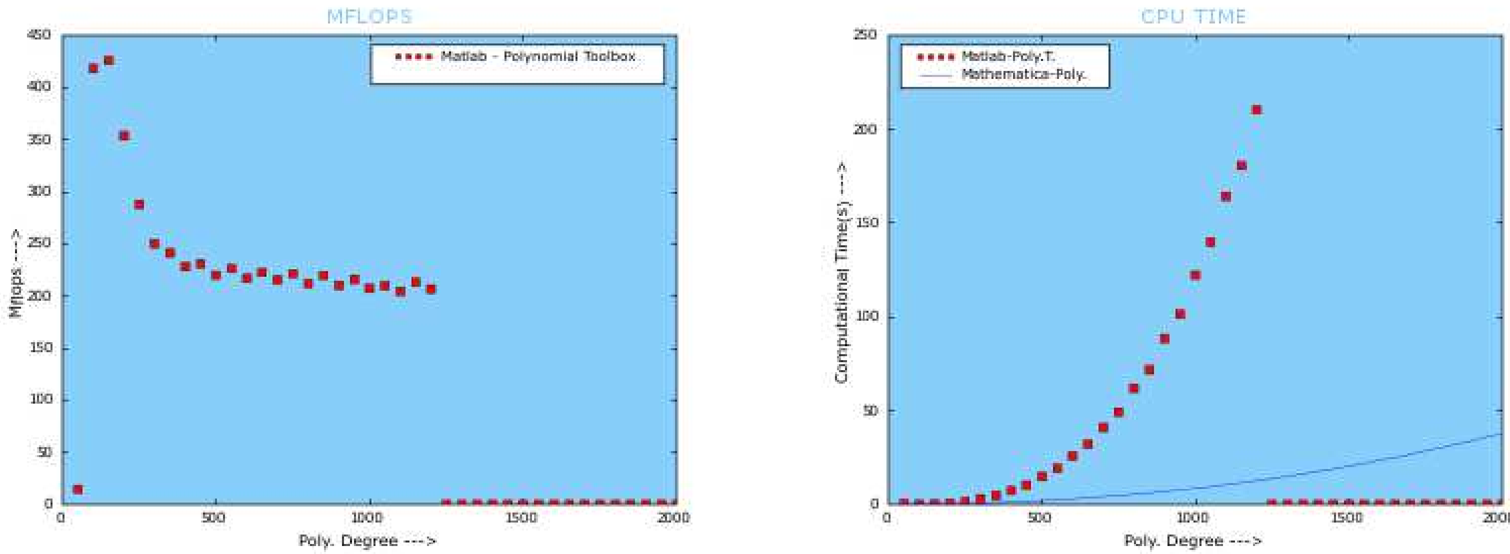

Σχήμα 3.6: Διοφαντική εξίσωση πολυωνυμικών πινάκων δευτέρου βαθμού μιας μεταβλητής $\mathbf{A}(\mathbf{s}) \cdot \mathbf{X}(\mathbf{s}) + \mathbf{B}(\mathbf{s}) \cdot \mathbf{Y}(\mathbf{s}) = \mathbf{C}(\mathbf{s})$ .



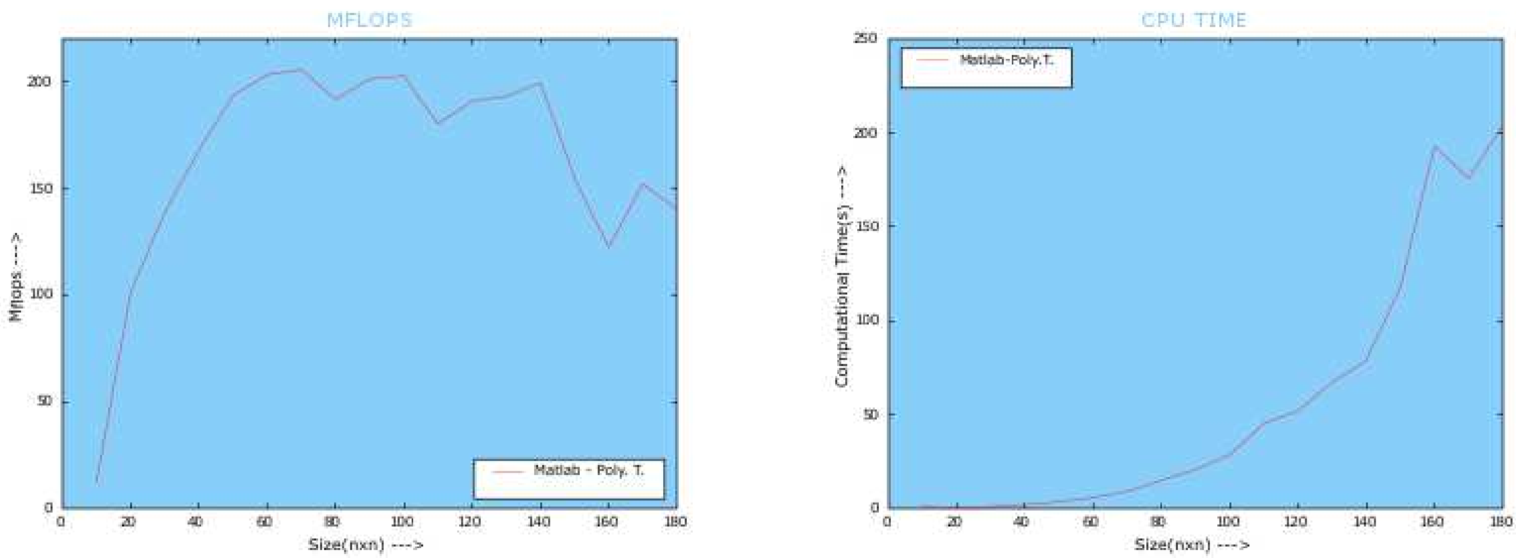

Σχήμα 3.7: Διοφαντική εξίσωση πολυωνύμων μιας μεταβλητής $\mathbf{a(s)\cdot x(s)+b(s)\cdot y(s)=c(s)}$.



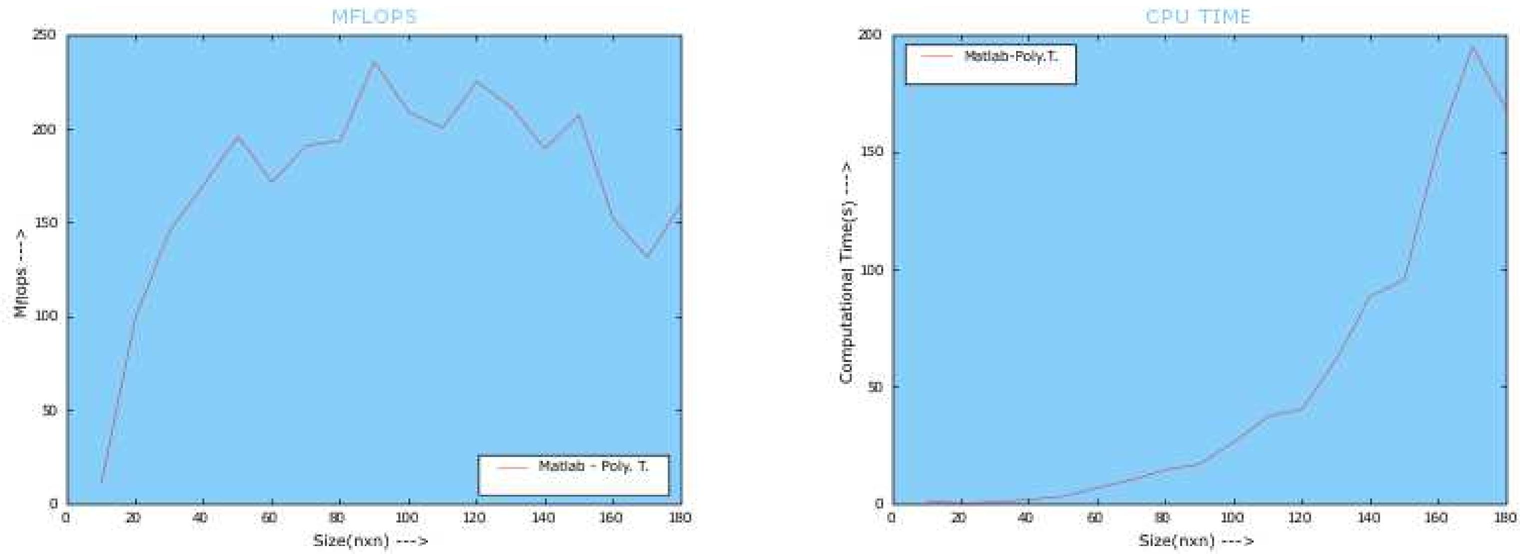

Σχήμα 3.8: Διμερής συμμετρική εξίσωση πολυωνυμικών πινάκων δευτέρου βαθμού μιας μεταβλητής $\mathbf{A'(s)\cdot X(s)+X'(s)\cdot A(s)=B(s)}$ .



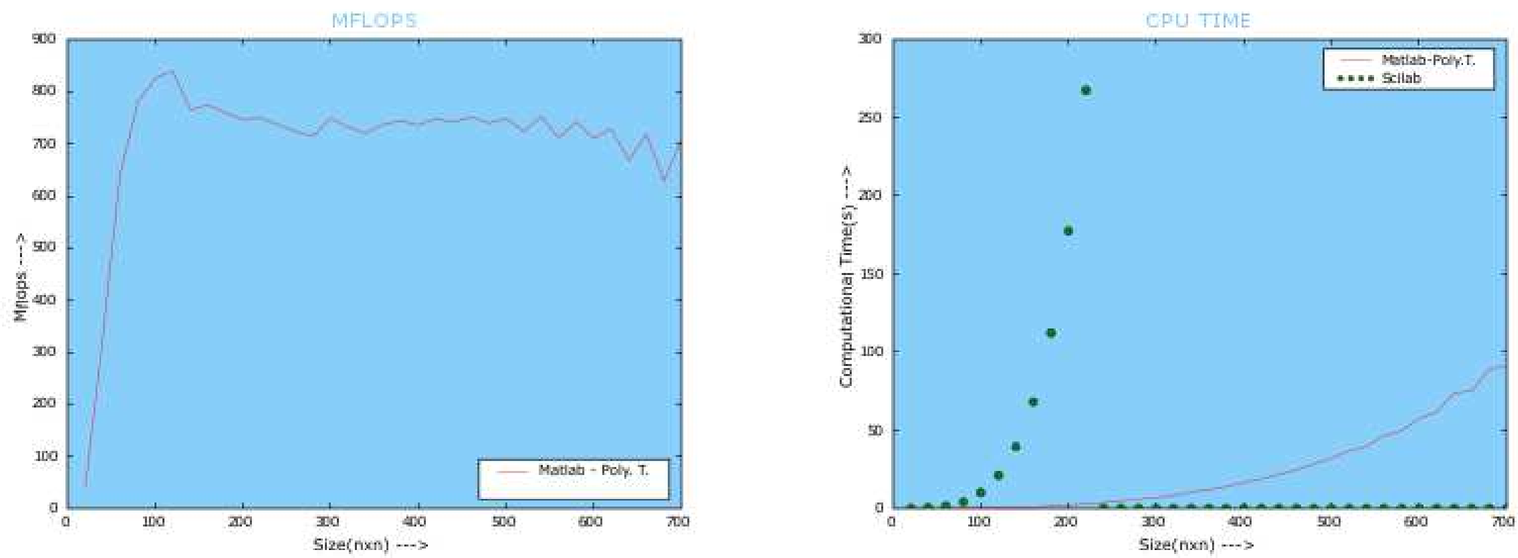

Σχήμα 3.9: Διμερής συμμετρική εξίσωση πολυωνύμων μιας μεταβλητής $\mathbf{a}'(\mathbf{s}) \cdot \mathbf{x}(\mathbf{s}) + \mathbf{x}'(\mathbf{s}) \cdot \mathbf{a}(\mathbf{s}) = \mathbf{b}(\mathbf{s})$ .



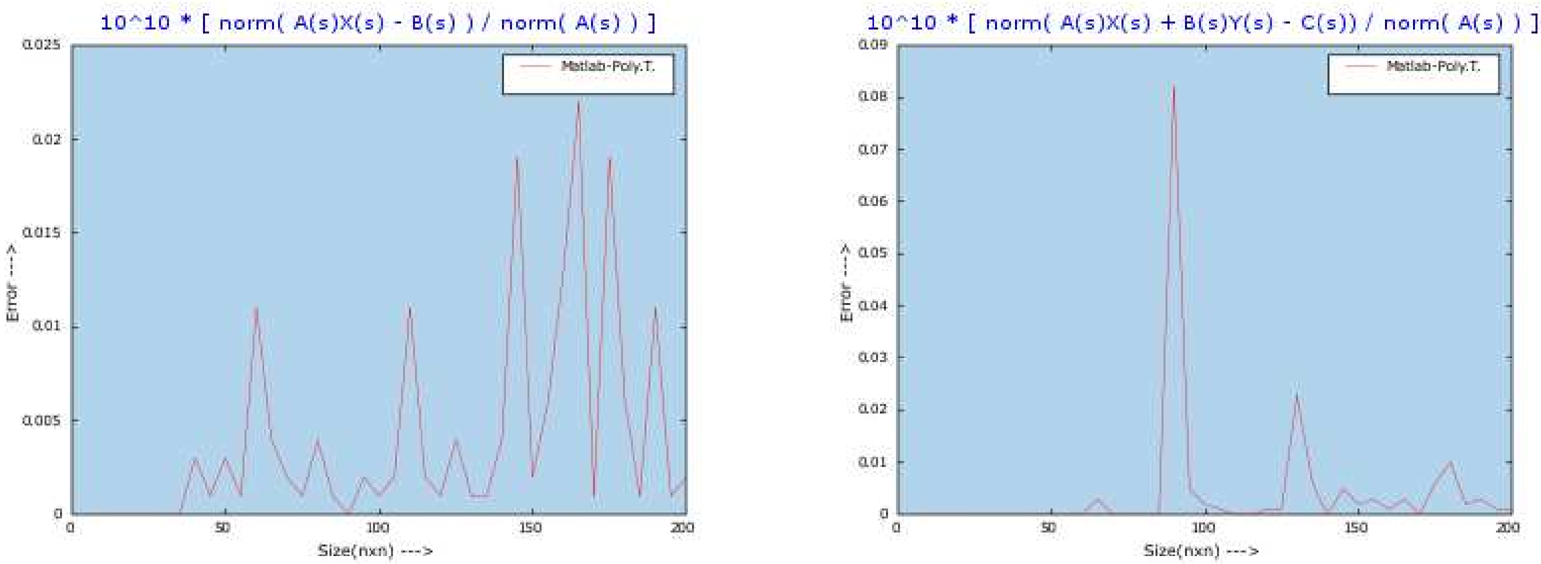

Σχήμα 3.10: Φασματική παραγοντοποίηση πολυωνυμικού πίνακα δευτέρου βαθμού μιας μεταβλητής $\mathbf{X}'(\mathbf{s}) \cdot \mathbf{X}(\mathbf{s}) = \mathbf{A}(\mathbf{s})$ .



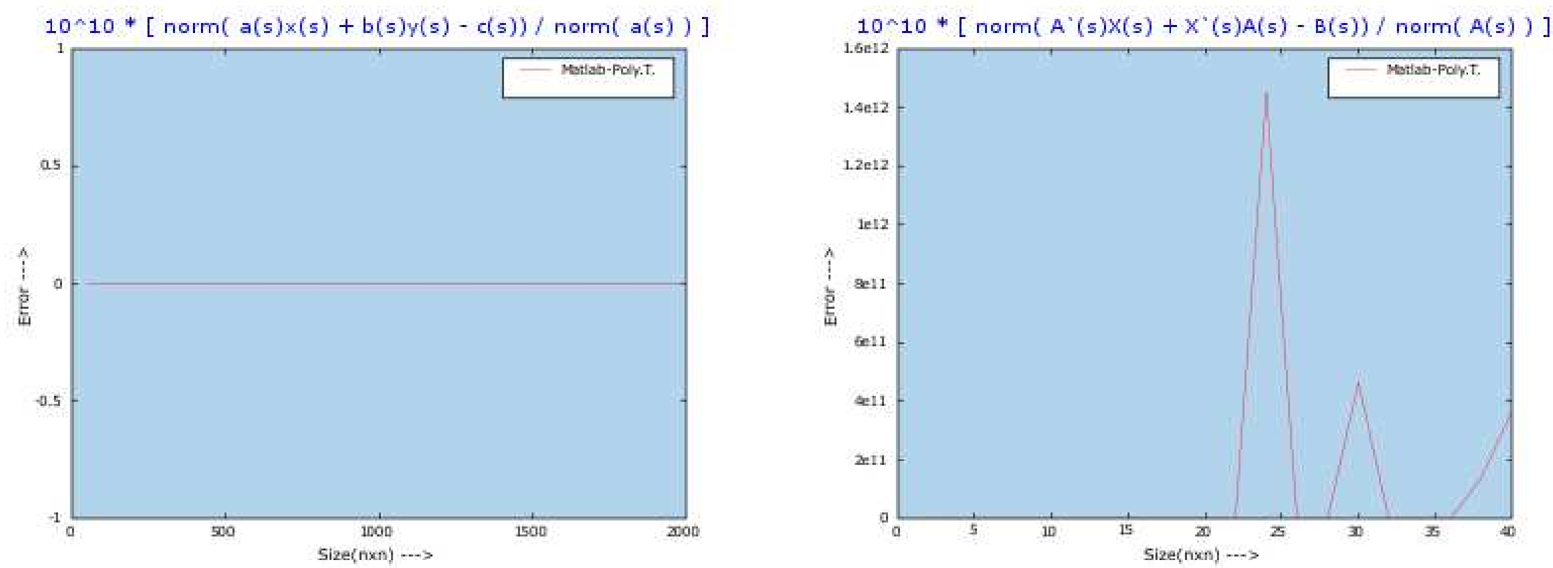

Σχήμα 3.11: J - Φασματική παραγοντοποίηση πολυωνυμικού πίνακα δευτέρου βαθμού μιας μεταβλητής $\mathbf{X}'(\mathbf{s}) \cdot \mathbf{J} \cdot \mathbf{X}(\mathbf{s}) = \mathbf{A}(\mathbf{s})$ .



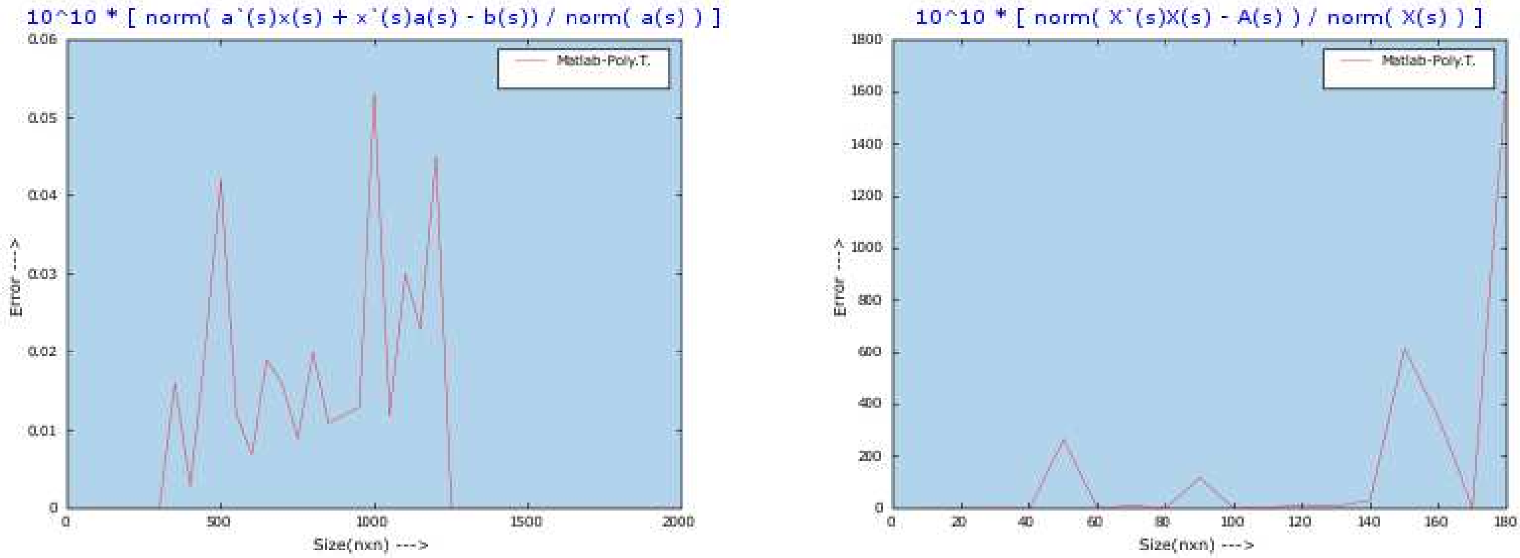

Σχήμα 3.12: Δεξιά διαίρεση πολυωνυμικών πινάκων δευτέρου βαθμού μιας μεταβλητής $\mathbf{N}(\mathbf{s}) = \mathbf{D}(\mathbf{s}) \cdot \mathbf{Q}(\mathbf{s}) + \mathbf{R}(\mathbf{s})$ αριαβλε.



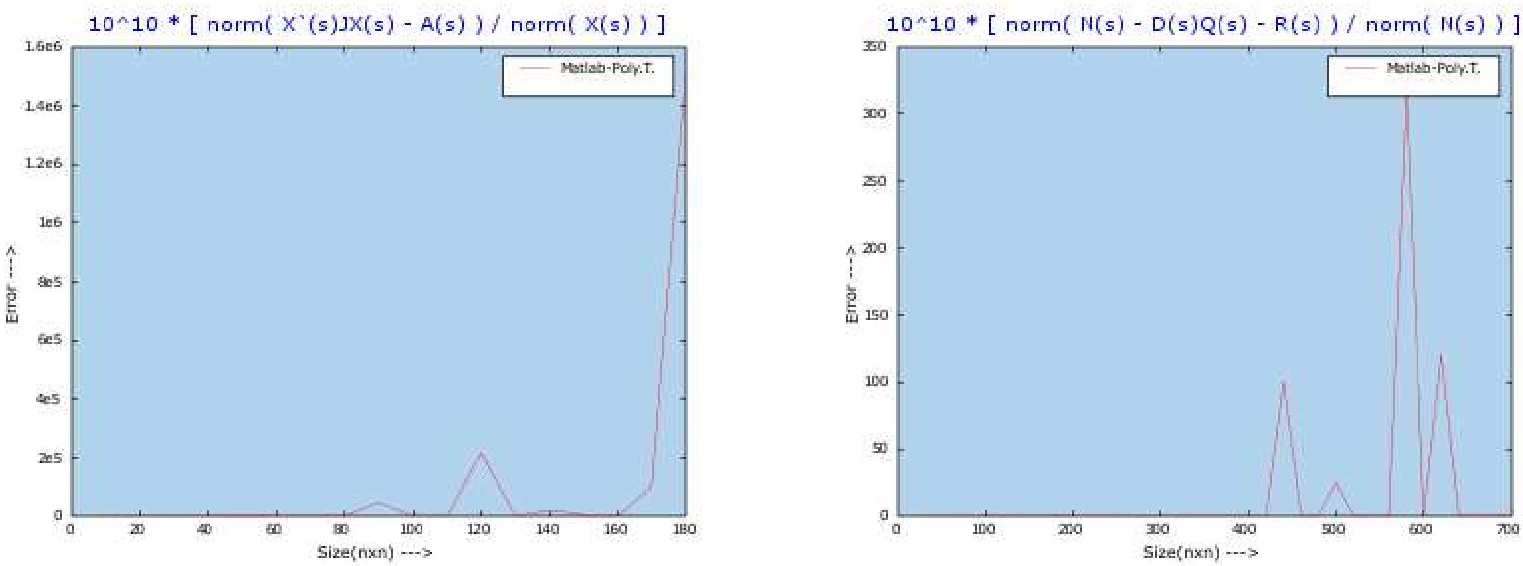

Σχήμα 3.13: Διακύμανση του μετρητή Error σε γραμμικές εξισώσεις πολυωνυμικών πινάκων




Σχήμα 3.14: Διακύμανση του μετρητή Error σε γραμμικές εξισώσεις πολυωνυμικών πινάκων




Σχήμα 3.15: Διακύμανση του μετρητή Error σε εξισώσεις πολυωνυμικών πινάκων




Σχήμα 3.16: Διακύμανση του μετρητή Error σε εξισώσεις πολυωνυμικών πινάκων

# Κεφάλαιο 4

# ΣΥΜΠΕΡΑΣΜΑΤΑ

Η παρούσα εργασία είχε σαν βασικό σκοπό την εισαγωγή της έννοιας του benchmarking στον χώρο των συστημάτων ελέγχου και συγκεκριμένα στον χώρο των λογισμικών ειδικευμένων στις πολυωνυμικές μεθόδους. Καθώς, αναπτύσσονται τα συγκεκριμένα προγράμματα όλο και περισσότερο είναι βασικής σημασίας η εισαγωγή εννοιών γνωστών και συνηθισμένων στη βιομηχανία λογισμικού όπως είναι το benchmarking. Μία τέτοια εξέλιξη προωθεί την ποιότητα των υπολογιστικών προγραμμάτων της επιστήμης του ελέγχου και τα καθιστά ανταγωνιστικά στο σύγχρονο πεδίο ανταγωνισμού.

Τα αποτελέσματα της αξιολόγησης των πιο χαρακτηριστικών λογισμικών όπου λειτουργούν οι πολυωνυμικές υπολογιστικές βιβλιοθήκες έδειξαν πολύ ενδιαφέροντα αποτελέσματα αφού συγκρίνονταν πακέτα βασισμένα στους ίδιους αλγόριθμους σε μεγάλο βαθμό αλλά προγραμματισμένα σε διαφορετικά περιβάλλοντα.

Βασικό στόχος του project ήταν ακόμα η ανάπτυξη ενός ανεξάρτητου λειτουργικής πλατφόρμας λογισμικού το οποίο θα εκτελεί και θα παρουσιάζει άμεσα τα αποτελέσματα των μετρήσεων. Αυτός ήταν ο λόγος δημιουργίας της πλατφόρμας αξιολόγησης POLYBENCH η σχεδίαση της οποίας αναλύθηκε στην παρούσα εργασία.

Τα αποτελέσματα της εργασίας παρουσιάζονται στο παγκόσμιο διαδικτυακό ιστό στη διεύθυνση anadrasis.math.auth.gr/polybench/index.htm.

# Βιβλιογραφία


[1] PolyX *www.polyx.com.*

[2] Polynomial Package *dce.felk.cvut.cz/kujan/polynomial*

[3] Scilab *scilabsoft.inria.fr*

[4] Python *www.python.org*

[5] Matplotlib *matplotlib.sourceforge.net*

[6] Karrigell *karrigell.sourceforge.net*

[7] Papi *icl.cs.utk.edu/papi*

[8] Matlab *www.mathworks.com*

[9] Mathematica *www.wolfram.com*

# Κεφάλαιο 5

# ΠΑΡΑΡΤΗΜΑ

Το συγκεκριμένο κεφάλαιο περιλαμβάνει όλα τα αρχεία αξιολόγησης benchmark scripts των δώδεκα πολυωνυμικών μεθόδων στα υπολογιστικά περιβάλλοντα του MATLAB, MATHEMATICA, SCILAB.