Αριστοτέλειο Πανεπιστήμιο Θεσσαλονίκης
Πολυτεχνική Σχολή

Τμήμα Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών

Μάθημα:Τεχνικές Βελτιστοποίησης

1η Εργαστηριακή Άσκηση

Ελεάνα Ζέρη AEM:10811

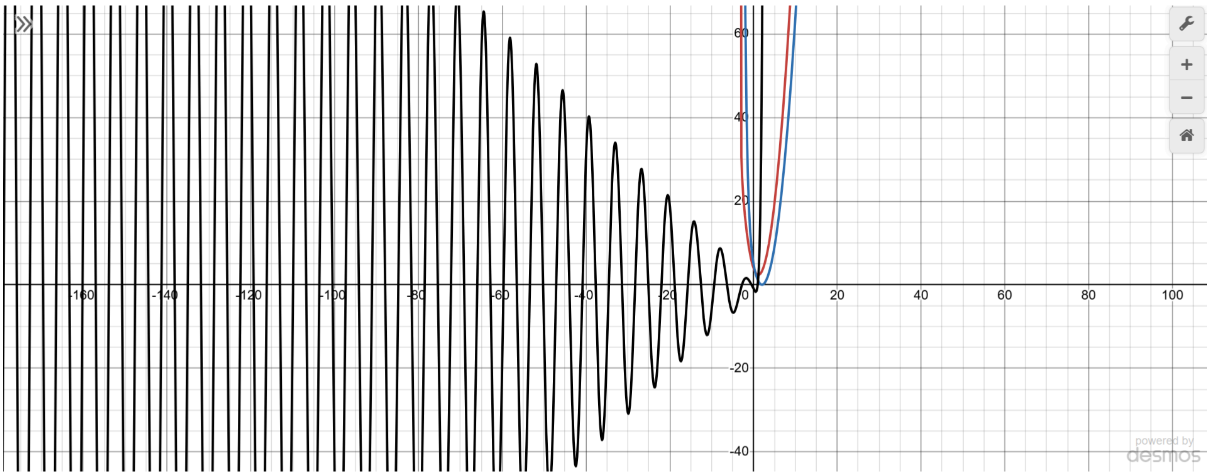
Η πρώτη εργαστηριακή άσκηση έχει ως ζητούμενο την ελαχιστοποίηση των παρακάτω κυρτών συναρτήσεων:

- f1(x) = (x - 2)^2 + x * ln(x + 3)
- f2(x) = e^(-2x) + (x - 2)^2
- f3(x) = e^x * (x^3 - 1) + (x - 1) * sin(x)   , [a,b]=[-1,3] (παρατηρούμε ότι όλες οι συναρτήσεις ορίζονται σε αυτό το διάστημα)

με την χρήση των ακόλουθων μεθόδων αναζήτησης ελάχιστου:

Θεμα 1:Μέθοδος της διχοτόμου
Θέμα 2:Μέθοδος του Χρυσού Τομέα
Θέμα 3:Μέθοδος Fibonacci
Θέμα 4:Μέθοδος της Διχοτόμου με χρήση παραγώγου

Οι γραφικές παραστάσεις είναι οι ακόλουθες:(f1(x)κόκκινο,f2(x) γαλάζιο,f3(x)μαύρο)

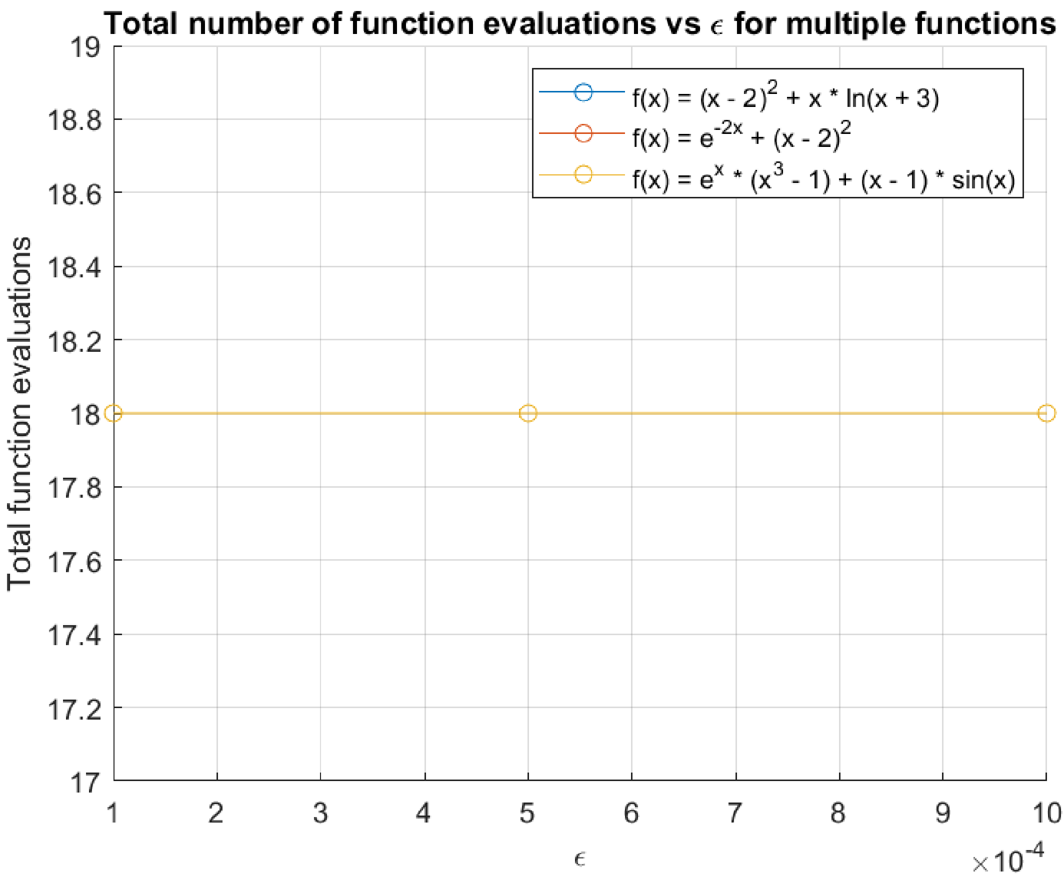


<u>Θέμα 1</u>:
Είχαμε να υλοποιήσουμε την μέθοδο της διχοτόμου κατά την οποία ξεκινάμε με ένα αρχικό διάστημα [a,b](εδώ [-1,3]) και στοχεύουμε να εντοπίσουμε το ελάχιστο σημείο της συνάρτησης σε αυτό το διάστημα. Οι συναρτήσεις που μας δίνονται είναι κυρτές οπότε το ελάχιστο που θα βρεθεί θα είναι και ολικό.Θέτουμε δύο παραμέτρους ε (μικρή απόσταση για τον υπολογισμό σημείων) και l(ακρίβεια για το πότε να

σταματήσουμε).Σε κάθε βήμα, υπολογίζουμε δύο σημεία κοντά στο κέντρο του διαστήματος,x1(αριστερά απο το κέντρο) και x2 (δεξιά από το κέντρο) και συγκρίνουμε τις τιμές της συνάρτησης στα σημεία αυτά.Αν η συνάρτηση έχει μικρότερη τιμή στο x1, διατηρούμε το αριστερό τμήμα του διαστήματος.Αν έχει μικρότερη τιμή στο x2, διατηρούμε το δεξί τμήμα.Συνεχίζουμε μέχρι το διάστημα να μικρύνει σε μέγεθος μικρότερο από l.Όταν το διάστημα είναι αρκετά μικρό (σύμφωνα με την ακρίβεια l), σταματάμε. Το ελάχιστο θεωρείται πως βρίσκεται στο τελικό, μικρό διάστημα που έχουμε.Με αυτή τη διαδικασία, η μέθοδος διχοτόμησης εντοπίζει ένα σημείο κοντά στο ελάχιστο της συνάρτησης.
Όσον αφορά την άσκηση:

- Κρατώντας σταθερό το τελικό εύρος αναζήτησης l=0.01,καθώς μεταβάλαμε την σταθερά ε > 0 (απόσταση από τη διχοτόμο),είχαμε να υλοποιήσουμε την μέθοδο της Διχοτόμου στο Matlab σε κάθε συνάρτηση και πιο συγκεκριμένα να εξετάσουμε το πώς μεταβάλλεται το πλήθος υπολογισμών των τιμών της fι(x),ι=1,2,3 συναρτήσει του ε.

Τα αντίστοιχα διαγράμματα είναι τα παρακάτω (αρχείο στο φάκελο matlab: *bisection_method1_all.m)* και έπεται ο σχολιασμός τους



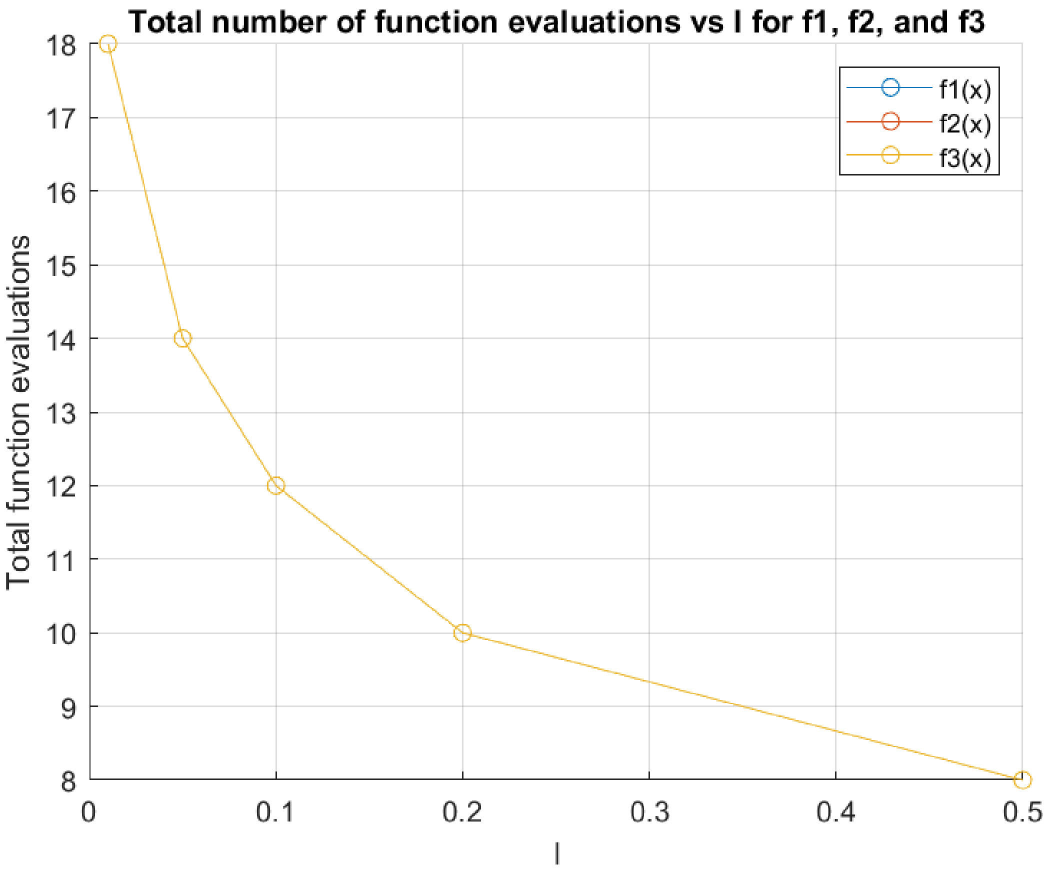
Παρατηρούμε οτι ο αριθμος που υπολογίσαμε την τιμη των f1,f2,f3 συναρτήσει διαφόρων τιμών του ε ταυτίζεται και είναι ακριβώς 18 στο διαστημα [-1,3].Αυτό είναι φυσιολογικό, καθώς στη μέθοδο της διχοτόμου ο αριθμός των υπολογισμών των συνάρτησεων fι(x) που απαιτούνται εξαρτάται από το αρχικό μήκος του διαστήματος και το τελικό επιθυμητό μήκος l,που είναι ίδια και για τις τρείς συναρτήσεις.Πιο συγκεκριμένα,ο αριθμός των επαναλήψεων k που απαιτούνται για να φτάσουμε σε ένα διάστημα μικρότερο ή ίσο με το l δίνεται από τη σχέση:
k~log(b-a/l)

Εδώ έχουμε:[a,b]=[−1,3], με μήκος b−a=4,l=0.0.Άρα:k~log(4/0.01)=log(400)=8.69(~9 άρα περίπου 18 επαναλήψεις)

Στη συνέχεια,

- Κρατώντας τώρα σταθερο το ε=0.001 είχαμε να μελετήσουμε τη μεταβολή του πλήθους των υπολογισμών της τιμής της κάθε συνάρτησης συναρτήσει του l και να δημιουργήσουμε τις αντίστοιχες γραφικές παραστάσεις από τις τιμές που προκύπτουν για τις τρεις συναρτήσεις.

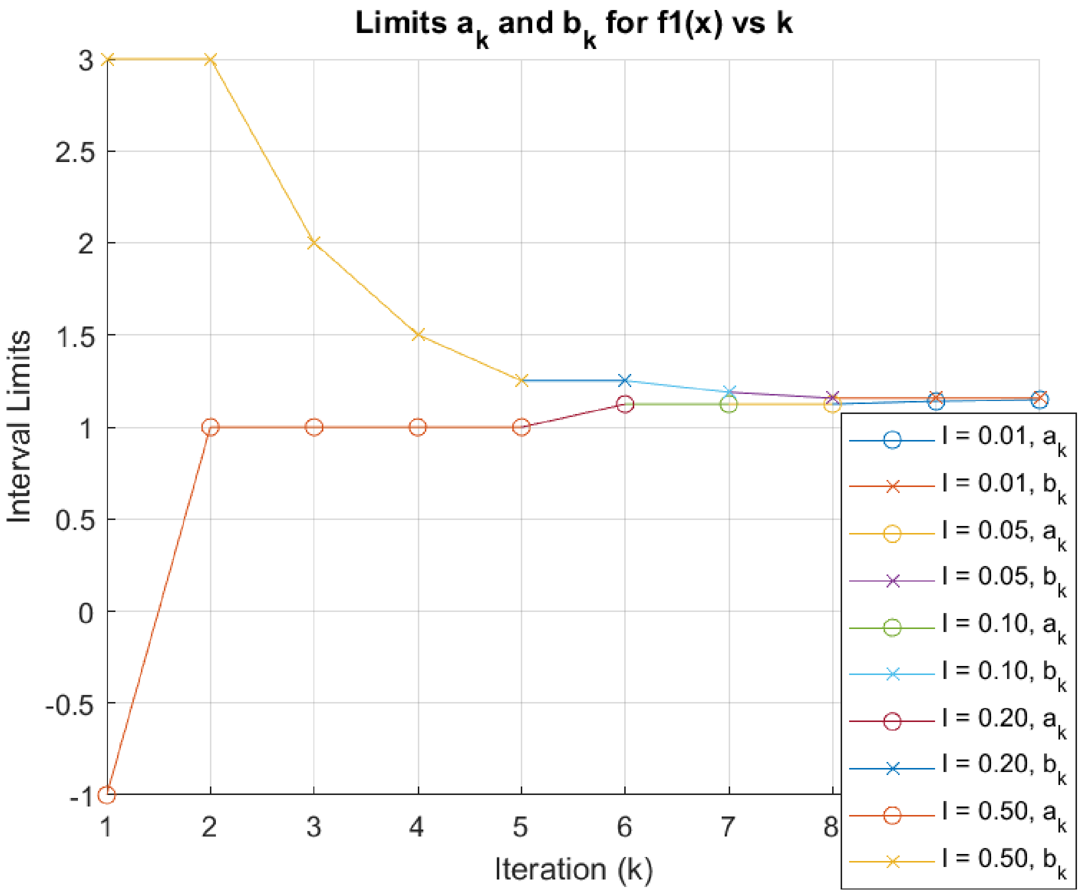
Το αντίστοιχο διαγράμμα είναι το παρακάτω (αρχείο στο φάκελο matlab: *bisection_method2_all.m)* και ακολουθεί ο σχολιασμός τους:



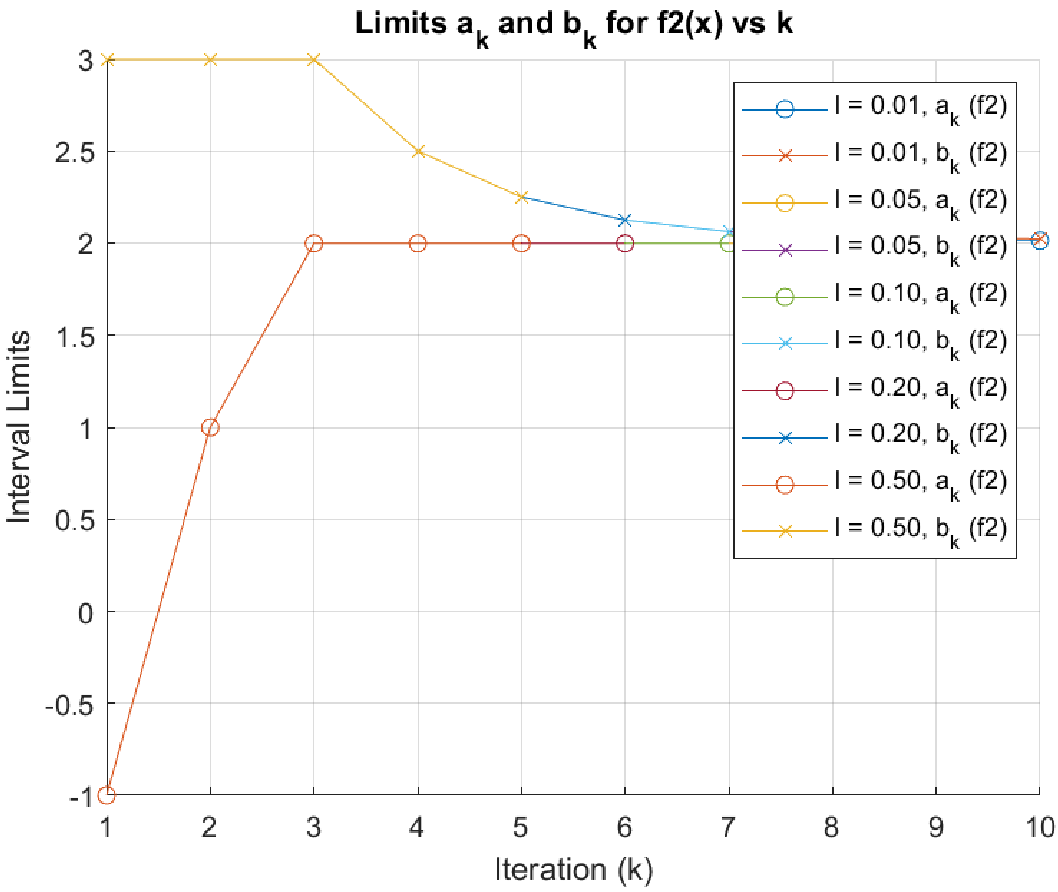
Βλέπουμε πάλι οτι η σχέση k~log(b-a/l) καθορίζει πόσες επαναλήψεις απαιτούνται, ανεξαρτήτως της μορφής της συνάρτησης fι(x). Αυτός είναι και ο λόγος που ο αριθμός επαναλήψεων ταυτίζεται για όλες τις συναρτήσεις, εφόσον τα a, b, l, και ε παραμένουν σταθερά.Επίσης βλέπουμε τον αριθμό επαναλήψεων να μειώνεται όσο αυξάνεται το l καθώς όσο μεγαλύτερο είναι το επιτρεπόμενο τελικό μήκος του διαστήματος (πιο χαλαρό l), τόσο λιγότερες επαναλήψεις χρειάζονται για να το πετύχουμε γιατί τόσο πιο γρήγορα θα πλησιάσουμε αυτό το μήκος l.

Τέλος,

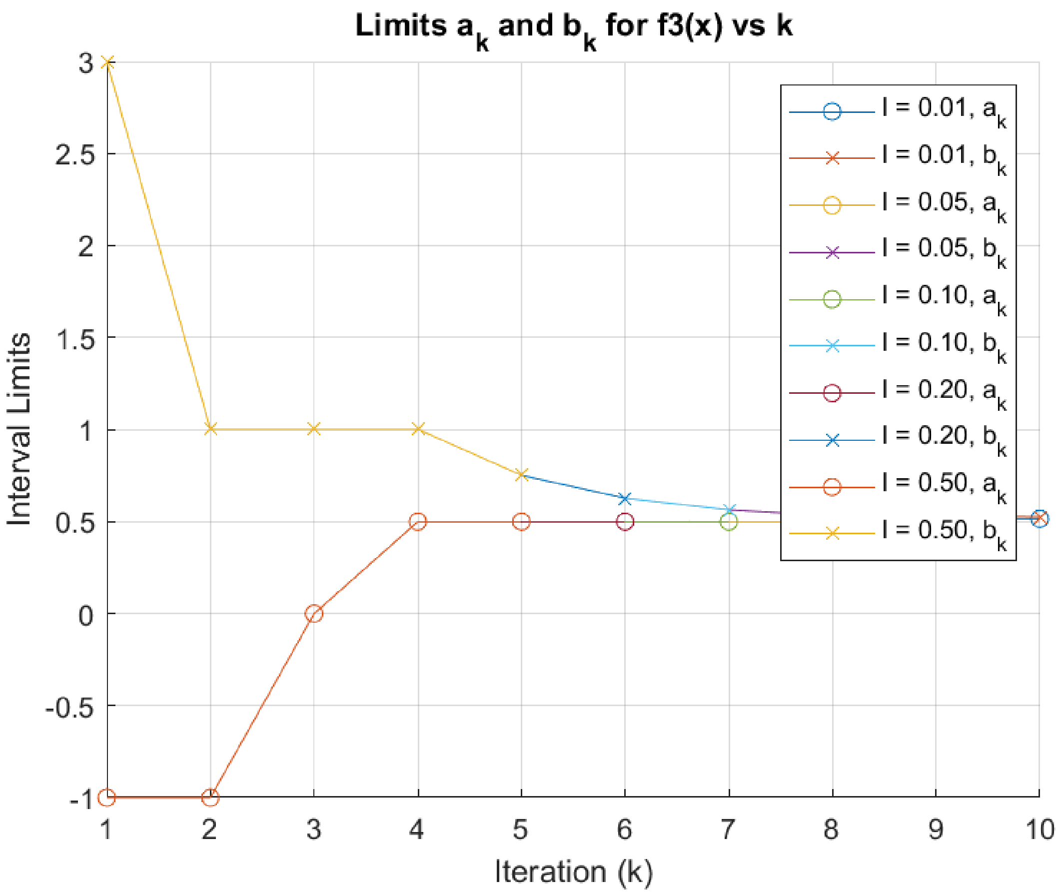
Σε τρία διαγράμματα ,ένα για κάθε συνάρτηση,είχαμε να σχεδιάσουμε τις γραφικές παραστάσεις των άκρων του διαστήματος [ak,bk] συναρτήσει του δείκτη επαναλήψεων k, δηλαδή (k,ak)και (k,bk),για διάφορες τιμές του τελικού εύρους αναζήτησης l.



Limits a_k and b_k for f1(x) vs k
Interval Limits
Iteration (k)
3
2.5
2
1.5
1
0.5
0
-0.5
-1
1
2
3
4
5
6
7
8
l = 0.01, a_k
l = 0.01, b_k
l = 0.05, a_k
l = 0.05, b_k
l = 0.10, a_k
l = 0.10, b_k
l = 0.20, a_k
l = 0.20, b_k
l = 0.50, a_k
l = 0.50, b_k



Limits a_k and b_k for f2(x) vs k
Interval Limits
Iteration (k)
I = 0.01, a_k (f2)
I = 0.01, b_k (f2)
I = 0.05, a_k (f2)
I = 0.05, b_k (f2)
I = 0.10, a_k (f2)
I = 0.10, b_k (f2)
I = 0.20, a_k (f2)
I = 0.20, b_k (f2)
I = 0.50, a_k (f2)
I = 0.50, b_k (f2)



Limits a_k and b_k for f3(x) vs k
Interval Limits
Iteration (k)
I = 0.01, a_k
I = 0.01, b_k
I = 0.05, a_k
I = 0.05, b_k
I = 0.10, a_k
I = 0.10, b_k
I = 0.20, a_k
I = 0.20, b_k
I = 0.50, a_k
I = 0.50, b_k

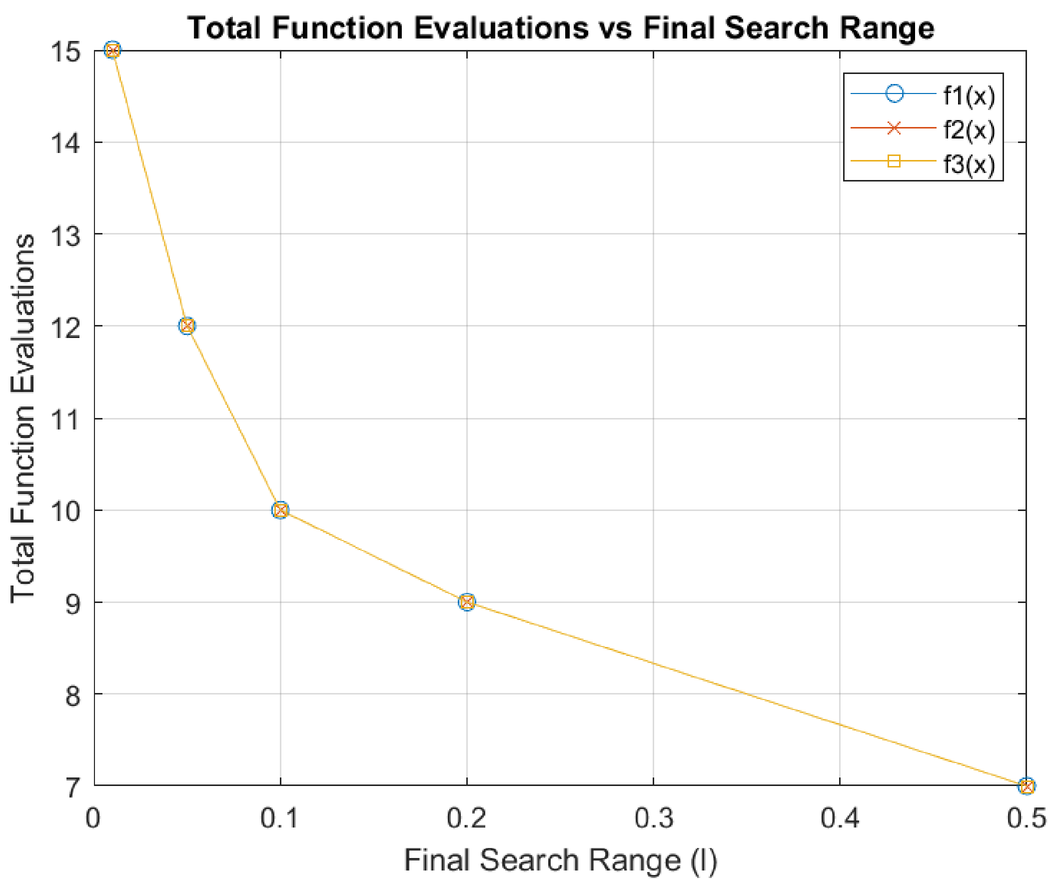
Παρατηρούμε ότι τα όρια ak και bk συγκλίνουν καθώς προχωράμε σε περισσότερες επαναλήψεις της διαδικασίας διχοτόμησης.Πιο συγκεκριμένα,οι τιμές ak τείνουν να γίνονται μεγαλύτερες ενώ οι bk γίνονται μικρότερες και οταν το ένα απο τα δυο άκρα μεταβάλλεται το άλλο παραμένει σταθερό. Αυτό συμβαδίζει με την αρχή της μεθόδου διχοτόμησης, η οποία επιλέγει το μεσαίο σημείο του διαστήματος και απορρίπτει το διάστημα στο οποίο η τιμή της συνάρτησης είναι μεγαλύτερη.Στα διάγραμματα, παρατηρούμε ακόμη ότι οι αλγόριθμοι με μεγαλύτερες τιμές του l (όπως l=0.5 και l=0.2) συρρικνώνουν τα όρια ak και bk πιο γρήγορα, αλλά η διαδικασία αυτή έχει ως αποτέλεσμα να φτάσουν σε ένα σημείο που θεωρούν ότι έχουν βρει το ελάχιστο. Αντίθετα, οι αλγόριθμοι με μικρότερα l (όπως l=0.1, l=0.05, και ιδιαίτερα l=0.01) συνεχίζουν να εστιάζουν στις μικρές μεταβολές των ορίων,διατηρώντας τη διαδικασία διχοτόμησης για περισσότερες επαναλήψεις και οδηγώντας τελικά σε μια καλύτερη και πιο ακριβή εκτίμηση του ελάχιστου.

(τα αρχεία στο φάκελο matlab είναι τα εξής: bisection_method_3o_f1.m , bisection_method_3o_f2.m , bisection_method_3o_f3.m)

<u>ΘΕΜΑ 2</u>:Είχαμε να υλοποιήσουμε την μέθοδο του Χρυσού Τομέα κατά την οποία:Ξεκινάμε με το αρχικό διάστημα [a,b] αναζήτησης και ορίζουμε ένα μικρό όριο ακρίβειας l, το οποίο καθορίζει το τελικό εύρος αναζήτησης.Το διάστημα χωρίζεται με βάση τον χρυσό λόγο(*γ~0.618*)σε δύο σημεία, x1 και x2. Στη συνέχεια υπολογίζουμε τις τιμές της συνάρτησης στα σημεία αυτά: f(x1) και f(x2).Στην k-οστή επανάληψη αν η συνάρτηση f έχει μικρότερη τιμή στο ένα σημείο (π.χ. στο x1), τότε απορρίπτουμε το κομμάτι του διαστήματος που περιέχει το x2.Αντίστοιχα, αν η τιμή είναι μικρότερη στο x2, απορρίπτουμε το κομμάτι που περιέχει το x1.Όταν περνάμε στην επόμενη επανάληψη, ο ένας από τους δύο υπολογισμούς της συνάρτησης f διατηρείται, καθώς μόνο το ένα από τα δύο σημεία, το x1 ή το x2, μετακινείται για τον περαιτέρω περιορισμό του διαστήματος.Έτσι, στην επανάληψη k+1, απαιτείται μόνο ένας νέος υπολογισμός της f στο νέο σημείο που αντικαθιστά το άλλο που απορρίφθηκε.Επαναλαμβάνουμε αυτή τη διαδικασία μέχρι το διάστημα [a,b] να γίνει μικρότερο από το επιθυμητό όριο ακρίβειας l. Στη συνέχεια θεωρούμε ότι βρήκαμε το ελάχιστο στο περιορισμένο διάστημα.Χάρη σε αυτή την προσέγγιση, η μέθοδος του χρυσού τομέα εξοικονομεί υπολογιστική προσπάθεια, απαιτώντας μόνο έναν νέο υπολογισμό ανά επανάληψη μετά το αρχικό βήμα.

- Κρατώντας σταθερό το τελικό εύρος αναζήτησης l=0.01,καθώς μεταβάλαμε την σταθερά ε > 0 (απόσταση από τη διχοτόμο),είχαμε να υλοποιήσουμε την μέθοδο του Χρυσού Τομέα στο Matlab σε κάθε συνάρτηση και πιο συγκεκριμένα να εξετάσουμε το πώς μεταβάλλεται το πλήθος υπολογισμών των τιμών της fι(x),ι=1,2,3 συναρτήσει του ε.

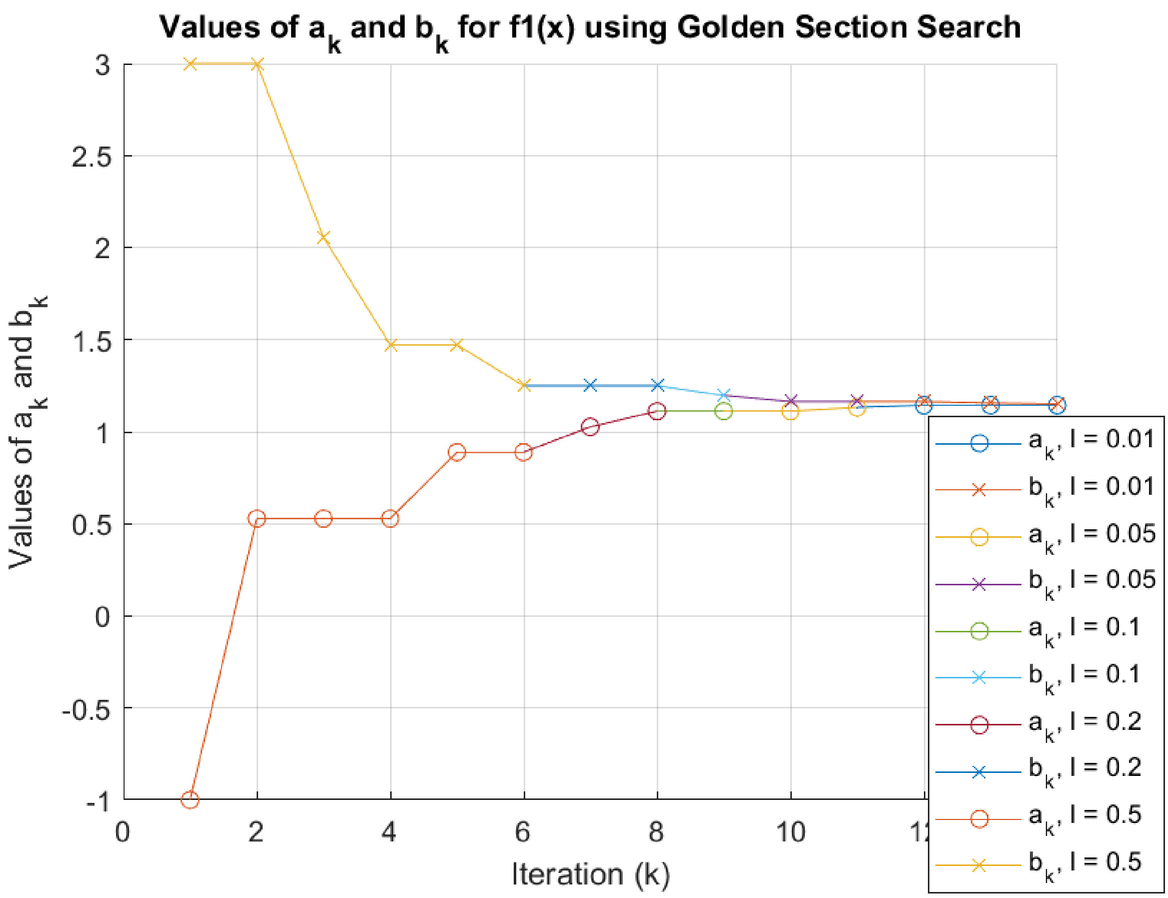
Το αντίστοιχο διαγράμμα είναι το παρακάτω (αρχείο στο φάκελο matlab: goldmeth1_all.m) και έπεται ο σχολιασμός τους

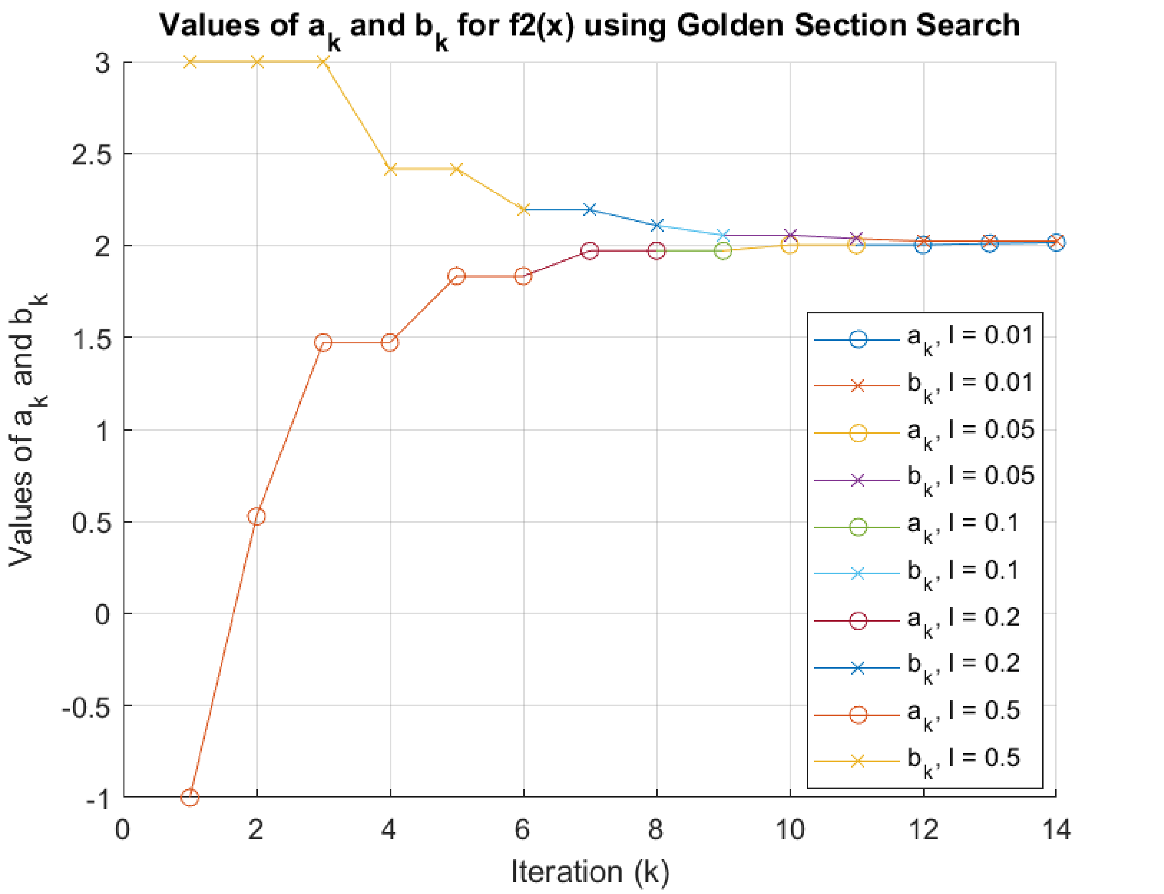


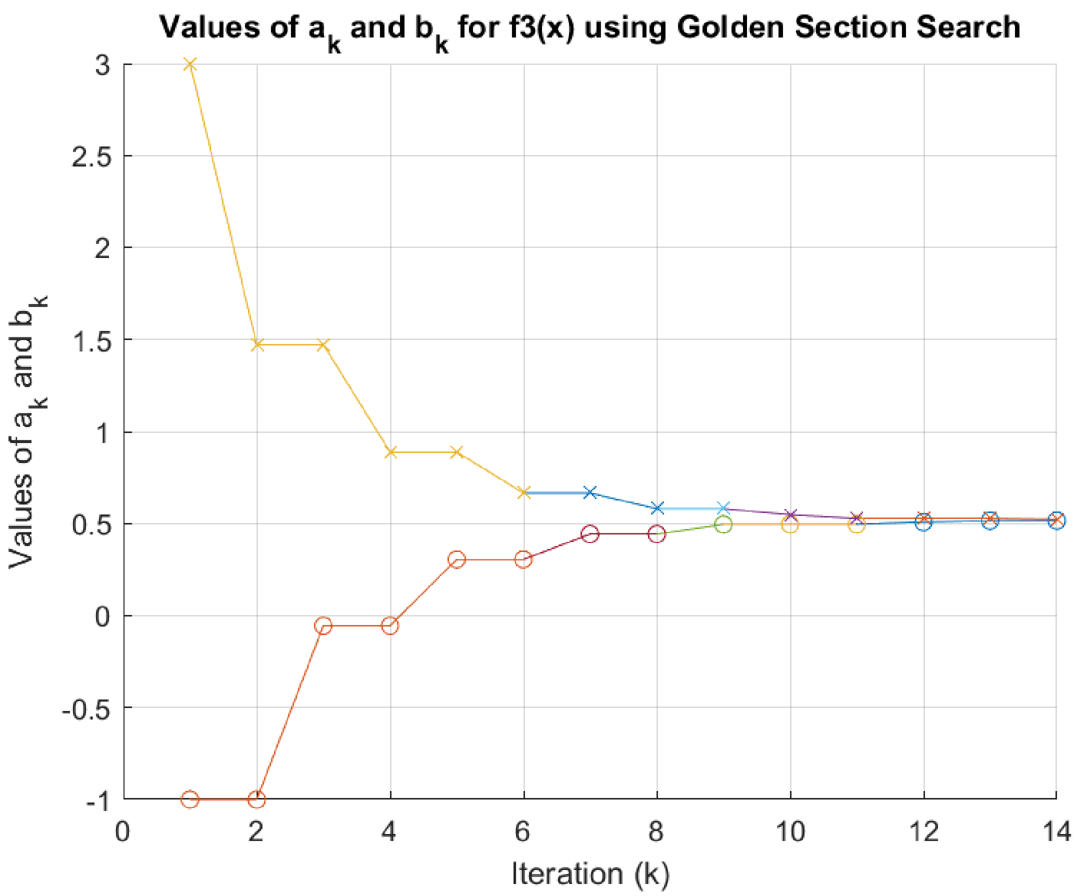
Βλέπουμε στη μέθοδο του Χρυσού Τομέα οπως και στη μέθοδο της Διχοτόμησης οτι η σχέση k~log(b-a/l) καθορίζει πόσες επαναλήψεις απαιτούνται, ανεξαρτήτως της μορφής της συνάρτησης fι(x). Έτσι ο αριθμός επαναλήψεων ταυτίζεται για όλες τις συναρτήσεις, εφόσον τα a, b, l, και ε παραμένουν σταθερά.Καταλήγουμε επίσης στα ίδια συμπεράσματα και όσον αφορα τη σχέση του πλήθους των επαναλήψεων σε σχέση με το l.

Τέλος,

- Σε τρία διαγράμματα ,ένα για κάθε συνάρτηση, σχεδιάστε τις γραφικές παραστάσεις των άκρων του διαστήματος [ak,bk] συναρτήσει του δείκτη επαναλήψεων k, δηλαδή (k,ak)και (k,bk),για διάφορες τιμές του τελικού εύρους αναζήτησης l. *(τα αρχεία στο φάκελο matlab είναι τα εξής: goldmeth2_f1.m , goldmeth2_f2.m , goldmeth2_f3.m , goldmethod2_all.m)*


Values of a_k and b_k for f1(x) using Golden Section Search
Values of a_k and b_k
Iteration (k)
a_k, l = 0.01
b_k, l = 0.01
a_k, l = 0.05
b_k, l = 0.05
a_k, l = 0.1
b_k, l = 0.1
a_k, l = 0.2
b_k, l = 0.2
a_k, l = 0.5
b_k, l = 0.5


Values of a_k and b_k for f2(x) using Golden Section Search
Values of a_k and b_k
Iteration (k)
a_k, l = 0.01
b_k, l = 0.01
a_k, l = 0.05
b_k, l = 0.05
a_k, l = 0.1
b_k, l = 0.1
a_k, l = 0.2
b_k, l = 0.2
a_k, l = 0.5
b_k, l = 0.5



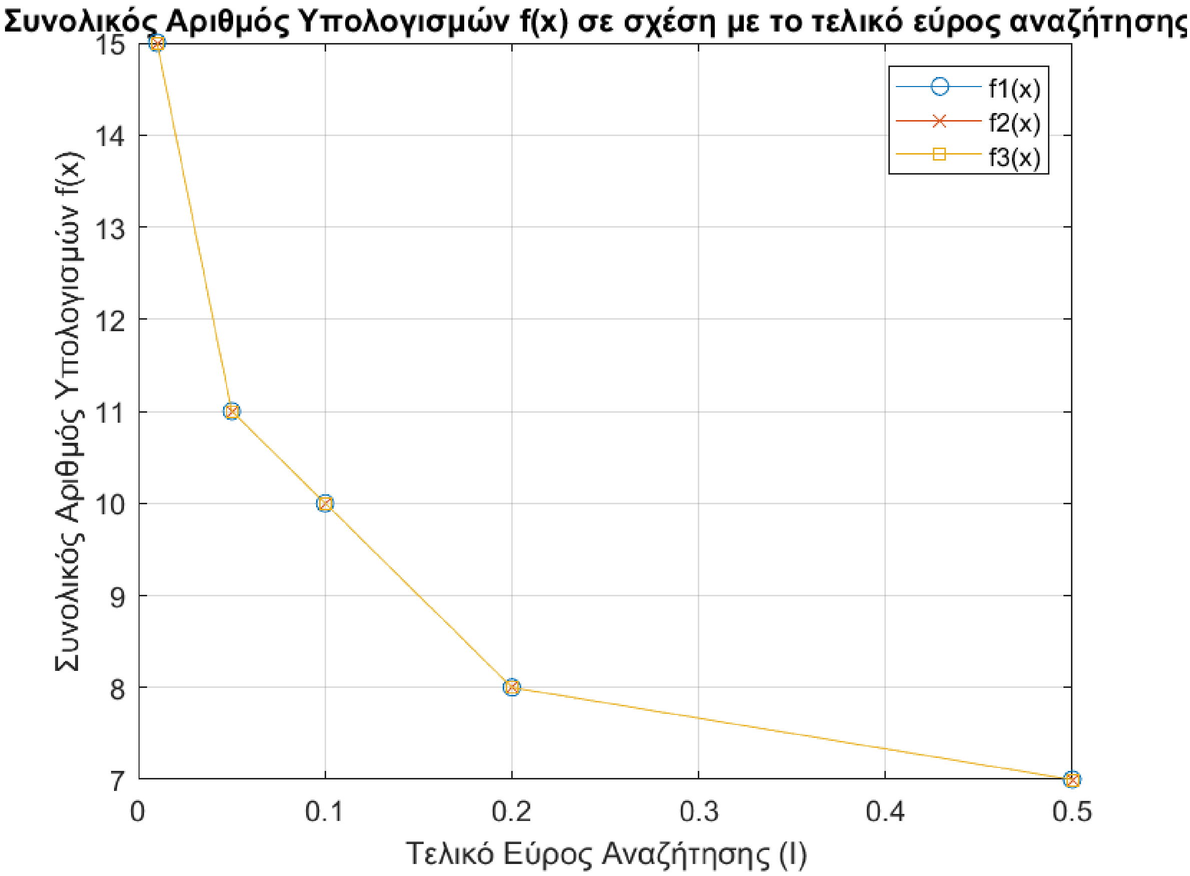
Εδώ αξίζει να σχολιάσουμε οτι συγκριτικά με τη μέθοδο της Διχοτόμου η μέθοδος του Χρυσού Τομέα συγκλίνει πιο αργά δηλάδη με περισσότερες επαναλήψεις του αλγορίθμου στο ελάχιστο των συναρτήσεων(μέθοδος διχοτόμησης σχεδόν 10 ενω μέθοδος Χρυσού Τομέα 14).Η μέθοδος Διχοτόμου αποδεικνύεται ταχύτερη και πιο άμεση από τη μέθοδο του Χρυσού Τομέα για τις συναρτήσεις που εξετάζουμε οι οποιες έχουν πολύπλοκες μορφες(ημίτονα,εκθετικά).Πιο συγκεκριμένα,διαιρεί το διάστημα σε δύο ίσα μέρη και επιλέγει το διάστημα όπου η συνάρτηση έχει την καλύτερη τιμή (μικρότερη), απορρίπτοντας γρήγορα το διάστημα αυτο και μειώνοντας ετσι τα όρια πολύ πιο γρήγορα.Η μέθοδος Χρυσού Τομέα απο την άλλη εξετάζει δύο σημεία σε αναλογία χρυσού τομέα και βασίζεται στην εξίσωση της βελτίωσης των ορίων. Τα σημεία αυτά επιλέγονται με βάση τη θέση των άκρων a και b.Ακόμη και όταν οι αποστάσεις μεταξύ των ακραίων σημείων γίνονται μικρότερες, η μέθοδος μπορεί να απαιτεί την εκτέλεση πολλών υπολογισμών για να επιβεβαιώσει ότι έχει φτάσει κοντά στο ελάχιστο.

Η μέθοδος του Χρυσού Τομέα απαιτεί έναν επιπλέον υπολογισμό στην πρώτη επανάληψη για να υπολογίσει τη συνάρτηση σε δύο σημεία. Από τη δεύτερη επανάληψη και μετά, χρειάζεται μόνο έναν νέο υπολογισμό ανά βήμα,μειώνει, λοιπον,τον συνολικό αριθμό υπολογισμών.Στη μέθοδο Διχοτόμου, διαιρείται πάντα το διάστημα σε δύο ίσα μέρη. Συμπερασματικά,καθώς ο υπολογισμός της f μπορεί να είναι χρονοβόρος σε σύνθετες συναρτήσεις, η μέθοδος του Χρυσού Τομέα εξοικονομεί υπολογιστικό χρόνο. Αυτός ο μειωμένος φόρτος υπολογισμού την καθιστά πιο αποδοτική για περιπτώσεις όπου η συνάρτηση f είναι ακριβής στον υπολογισμό της. Ετσι,παρά το γεγονός ότι μπορεί να απαιτήσει περισσότερες επαναλήψεις για να συγκλίνει, το γεγονός ότι η μέθοδος του Χρυσού Τομέα υπολογίζει τη συνάρτηση λιγότερες φορές συνολικά, την καθιστά συγκριτικά πιο αποδοτική από τη μέθοδο της Διχοτόμου σε επίπεδο συνολικών υπολογισμών της συνάρτησης.

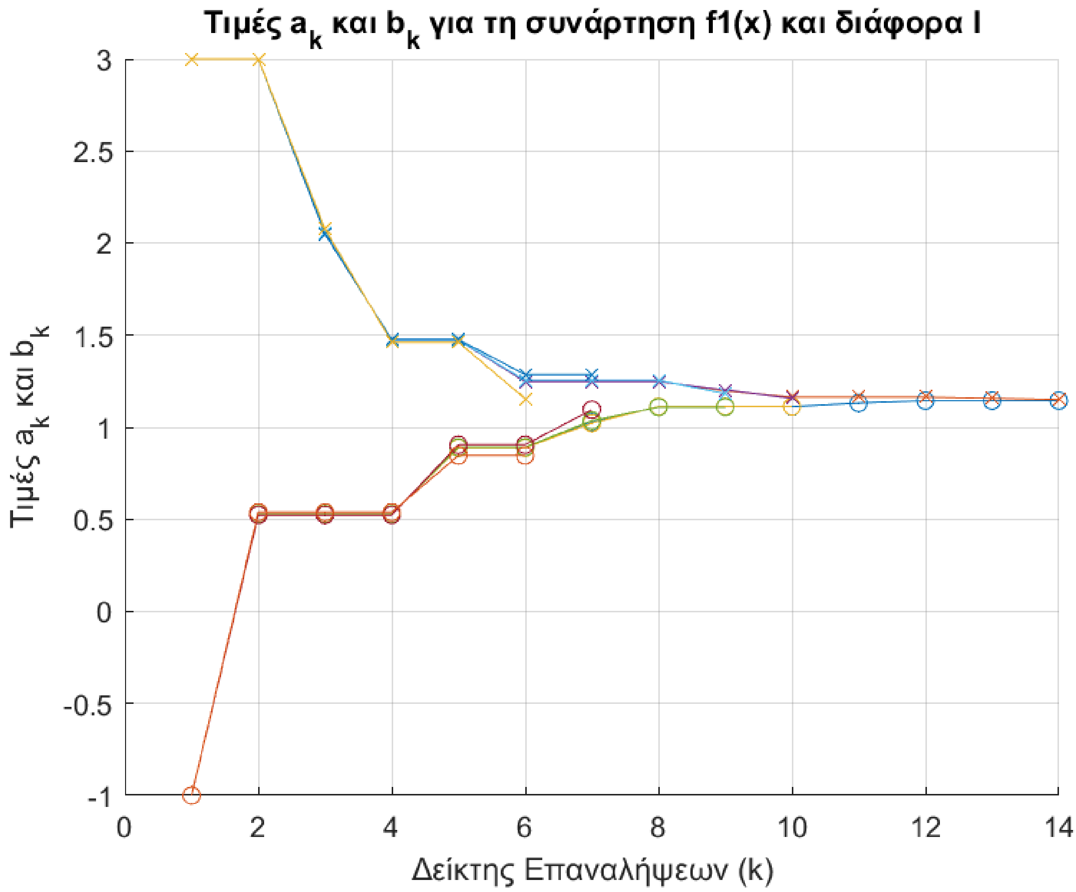
<u>ΘΕΜΑ 3</u>: Είχαμε να εφαρμόσουμε ακριβώς τα ίδια με το Θεμα 2 χρησιμοποιώνττας την μέθοδο Fibonacci σύμφωνα με την οποία:χρησιμοποιούμε τους αριθμούς Fibonacci για να διασφαλίσουμε ότι ο συνολικός αριθμός υπολογισμών της f(x) (δηλαδή ο αριθμός επαναλήψεων) καλύπτει την απαραίτητη ακρίβεια του διαστήματος.Αρχικά υπολογίζουμε δύο σημεία x1 και x2 μέσα στο διάστημα, με βάση τον λόγο Fibonacci, και αξιολογούμε την τιμή της f σε αυτά τα σημεία.Αν η f(x1) είναι μεγαλύτερη από την f(x2), τότε το ελάχιστο βρίσκεται στο δεξί υποδιάστημα, και ανανεώνουμε το διάστημα [x1,β], ορίζοντας το x2 ως το νέο x1.Αντίστοιχα, αν f(x1)<f(x2), το ελάχιστο βρίσκεται στο αριστερό υποδιάστημα, και το νέο διάστημα γίνεται [α,x2].Στην κάθε επόμενη επανάληψη k+1, χρειάζεται μόνο ένας νέος υπολογισμός της f, είτε στο x1k+1 είτε στο x2k+1, επειδή το ένα σημείο (είτε το x1 είτε το x2) διατηρείται από την προηγούμενη επανάληψη.Αυτό μειώνει σημαντικά τον συνολικό αριθμό των υπολογισμών της συνάρτησης.Όταν το διάστημα [α,β] γίνει μικρότερο από το επιθυμητό όριο l, η διαδικασία σταματά. Στη μέθοδο Fibonacci, το πλάτος του διαστήματος μειώνεται σταδιακά, και μετά από n−1 επαναλήψεις το διάστημα θα έχει μειωθεί κατά έναν παράγοντα που καθορίζεται από τον αριθμό Fibonacci. Συγκεκριμένα, ο αριθμός n πρέπει να ικανοποιεί τη σχέση:

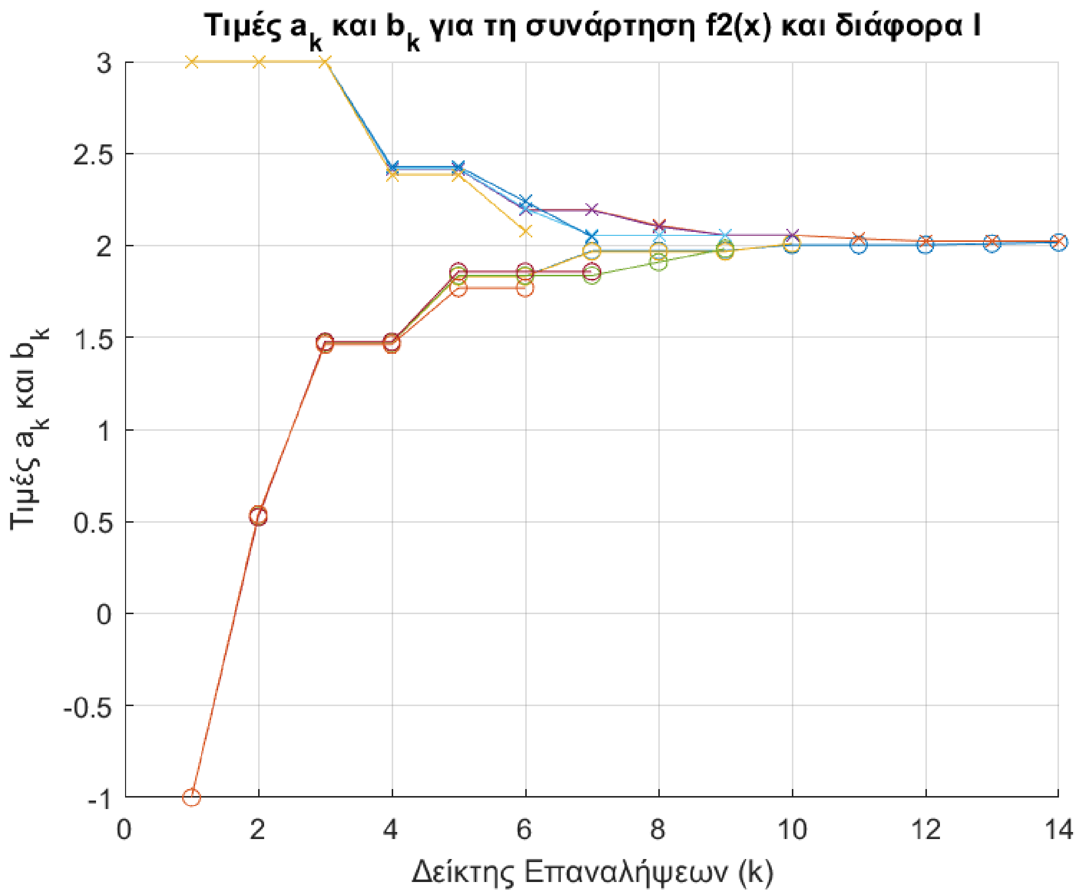
(β1-α1)/Fn<=l

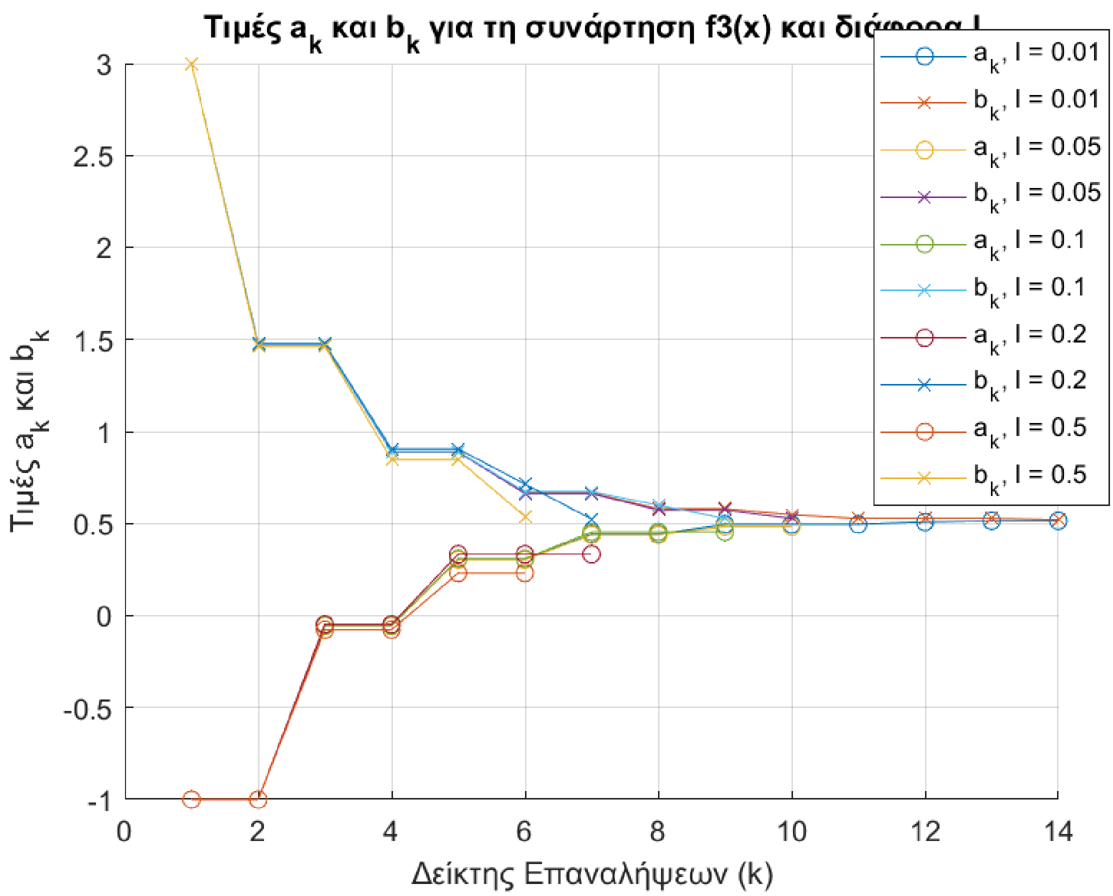
- Ο συνολικός αριθμός υπολογισμών της f(x) συναρτήσει του l σε κάθε μία απο τις συναρτήσεις δίνεται παρακάτω(αρχείο στο φάκελο matlab: *fibmeth1_all.m*):

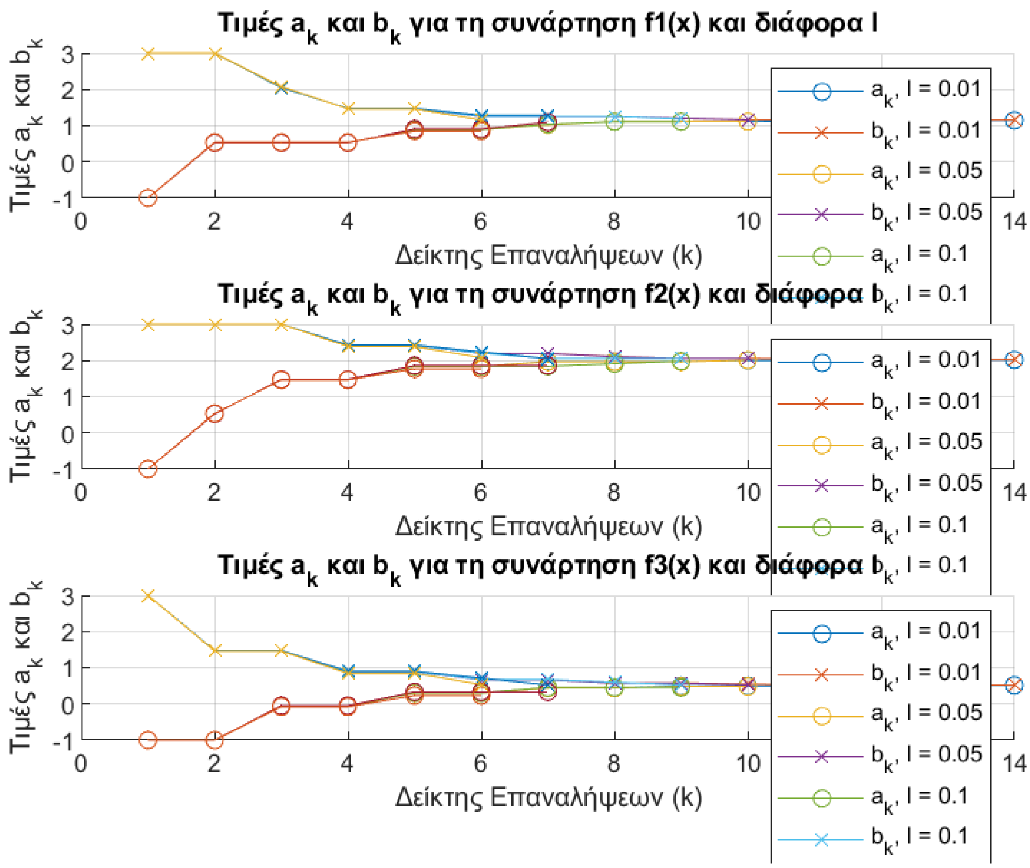


  - οι γραφικές παραστάσεις των άκρων του διαστήματος [ak,bk] συναρτήσει του δείκτη επαναλήψεων k, δηλαδή (k,ak)και (k,bk),για διάφορες τιμές του τελικού εύρους αναζήτησης l παρατίθενται παρακάτω*(τα αρχεία στο φάκελο matlab είναι τα εξής: fib_2_all.m , fibmeth2_f1.m , fibmeth2_f2.m , fibmeth2_f3.m)* :


Τιμές a_k και b_k για τη συνάρτηση f1(x) και διάφορα l
Τιμές a_k και b_k
Δείκτης Επαναλήψεων (k)
3
2.5
2
1.5
1
0.5
0
-0.5
-1
0
2
4
6
8
10
12
14


Τιμές a_k και b_k για τη συνάρτηση f2(x) και διάφορα l
Τιμές a_k και b_k
Δείκτης Επαναλήψεων (k)
3
2.5
2
1.5
1
0.5
0
-0.5
-1
0
2
4
6
8
10
12
14


Τιμές a_k και b_k για τη συνάρτηση f3(x) και διάφορα l
Τιμές a_k και b_k
Δείκτης Επαναλήψεων (k)
a_k, l = 0.01
b_k, l = 0.01
a_k, l = 0.05
b_k, l = 0.05
a_k, l = 0.1
b_k, l = 0.1
a_k, l = 0.2
b_k, l = 0.2
a_k, l = 0.5
b_k, l = 0.5


Τιμές a_k και b_k για τη συνάρτηση f1(x) και διάφορα l
Τιμές a_k και b_k για τη συνάρτηση f2(x) και διάφορα l
Τιμές a_k και b_k για τη συνάρτηση f3(x) και διάφορα l
Τιμές a_k και b_k
Δείκτης Επαναλήψεων (k)
a_k, l = 0.01
b_k, l = 0.01
a_k, l = 0.05
b_k, l = 0.05
a_k, l = 0.1
b_k, l = 0.1

Στη συγκεκριμένη εργασία για:
$a1=-1$
$b1=3$
Fn είναι ο n-οστός όρος της ακολουθίας Fibonacci
Έχουμε:
$b1-a1=3-(-1)=4$
$4/Fn \leq 0.01 \Rightarrow Fn \geq 4/0.01 \Rightarrow Fn \geq 400$
Άρα,χρειαζόμαστε τον πρώτο όρο της ακολουθίας Fibonacci που είναι μεγαλύτερος ή ίσος με 400.Ο μικρότερος όρος $Fn \geq 400$ είναι το $F15=610$.Τελικά,για να περιοριστεί το διάστημα αναζήτησης κάτω από το ζητούμενο $l=0.01$, χρειάζονται 15 επαναλήψεις.

Σύγκριση με τις προηγούμενες δύο μεθόδους:

Η μέθοδος Fibonacci ξεχωρίζει ως η πιο αποδοτική μέθοδος στον αριθμό υπολογισμών της συνάρτησης f για να επιτύχει ένα διάστημα ακρίβειας l. Αυτό οφείλεται στο γεγονός οτι αυτή η μέθοδος βασίζεται στη σειρά Fibonacci για τον καθορισμό των σημείων υποδιαίρεσης. Έτσι, ο αριθμός των επαναλήψεων n καθορίζεται εκ των προτέρων ώστε το διάστημα να ικανοποιεί την ακρίβεια l μετά από n−1 επαναλήψεις. Σε κάθε βήμα, η μείωση του διαστήματος ακολουθεί έναν συγκεκριμένο λόγο,τον Fn-k/Fn-k-1 και διαφοροποιείται ετσι από τη μέθοδο του Χρυσού Τομέα στο ότι το υπό-διάστημα αναζήτησης στην κ επανάληψη δεν συνδέεται μ' αυτό της κ – 1 επανάληψης με μια σταθερα(χρυσός λόγος ~0.618), αλλα μεταβαλλεται απο επαναληψη σε επαναληψη. Η μέθοδος του Χρυσού Τομέα, ενώ είναι κοντά στην αποδοτικότητα της Fibonacci, είναι λιγότερο αποδοτική από τη Fibonacci για τον ίδιο αριθμό επαναλήψεων. Στις γραφικές παραστάσεις των άκρων [ak,bk]συναρτήσει των επαναλήψεων k, φαίνεται ότι η μέθοδος Fibonacci μειώνει το διάστημα πιο ομαλά, η μέθοδος του Χρυσού Τομέα τείνει να συγκλίνει παρόμοια με την Fibonacci για μεγάλο n, ενώ η διχοτόμος έχει μεγαλύτερα άλματα στη μείωση του διαστήματος.

<u>ΘΕΜΑ 4</u>: Είχαμε να εφαρμόσουμε ακριβώς τα ίδια με το Θεμα 2 χρησιμοποιώνττας την μέθοδο της Διχοτόμου με τη χρήση παραγώγων κατά την οποία σε κάθε επανάληψη, υπολογίζουμε την παράγωγο της συνάρτησης f(x) στο μέσο του διαστήματος c=a+b/2. Η παράγωγος μας δείχνει αν η συνάρτηση αυξάνεται ή μειώνεται στο σημείο αυτό.

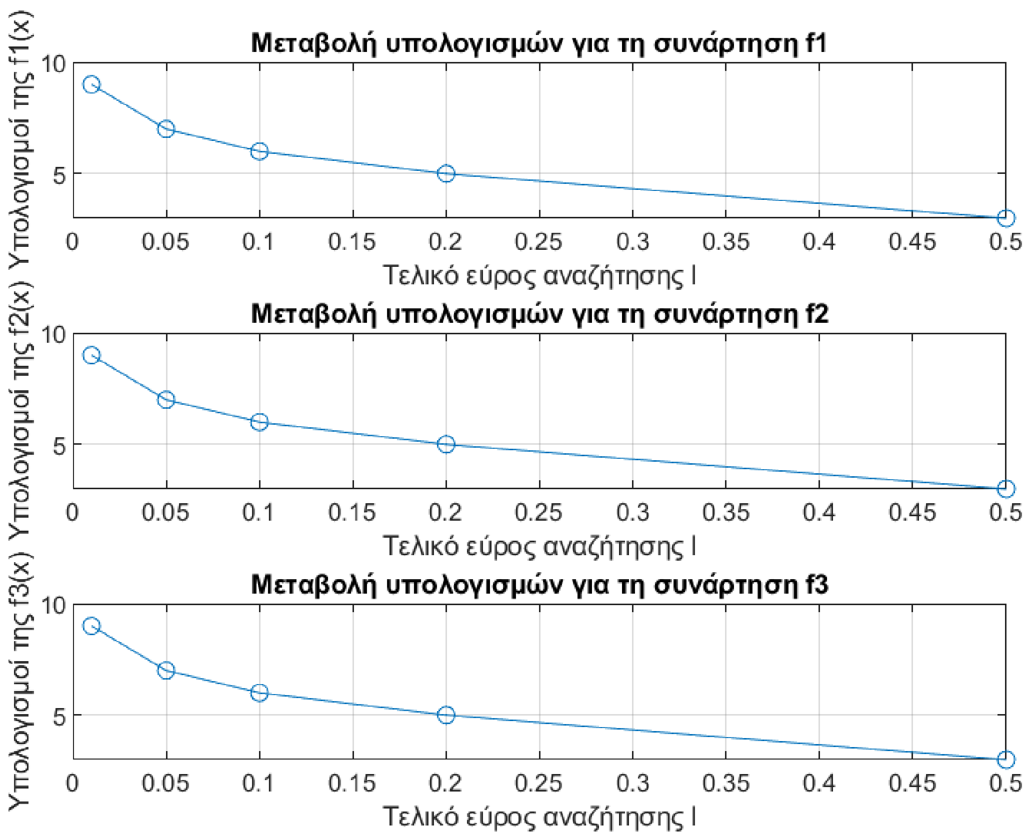
Αν f′(c)>0, τότε το ελάχιστο είναι προς τα αριστερά του c, και το διάστημα ενημερώνεται σε [a,c].

Αν f′(c)<0, τότε το ελάχιστο είναι προς τα δεξιά του c, και το διάστημα γίνεται [c,b].

Αν f′(c)=0, το σημείο c είναι στάσιμο και ενδέχεται να είναι το ελάχιστο (ή μέγιστο) σημείο της συνάρτησης,οπότε ο αλγόριθμος τερματίζει.

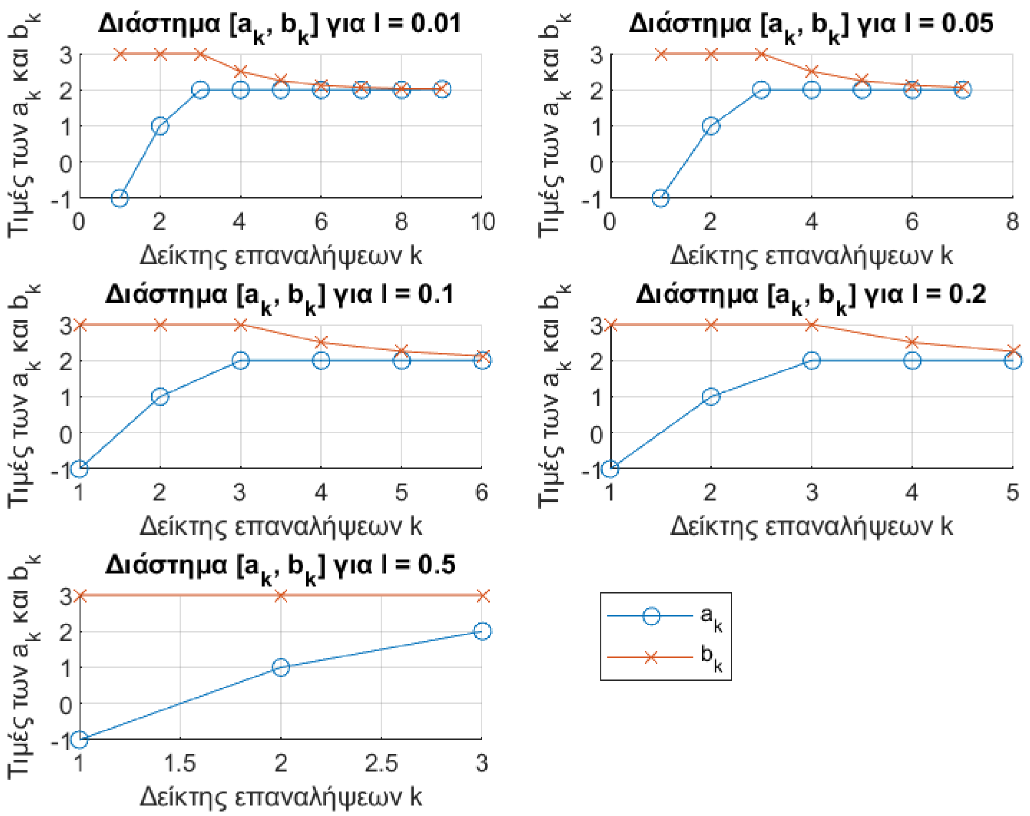
Η διαδικασία επαναλαμβάνεται έως ότου το μήκος του διαστήματος [a,b] γίνει μικρότερο από το προκαθορισμένο όριο ακρίβειας l. Στο σημείο αυτό, θεωρείται ότι το ελάχιστο βρίσκεται στο σημείο c του τελικού διαστήματος.

- Ο συνολικός αριθμός υπολογισμών της f(x) συναρτήσει του l σε κάθε μία απο τις συναρτήσεις δίνεται παρακάτω(αρχείο στο φάκελο matlab: dixmeparagogous1_all.m):

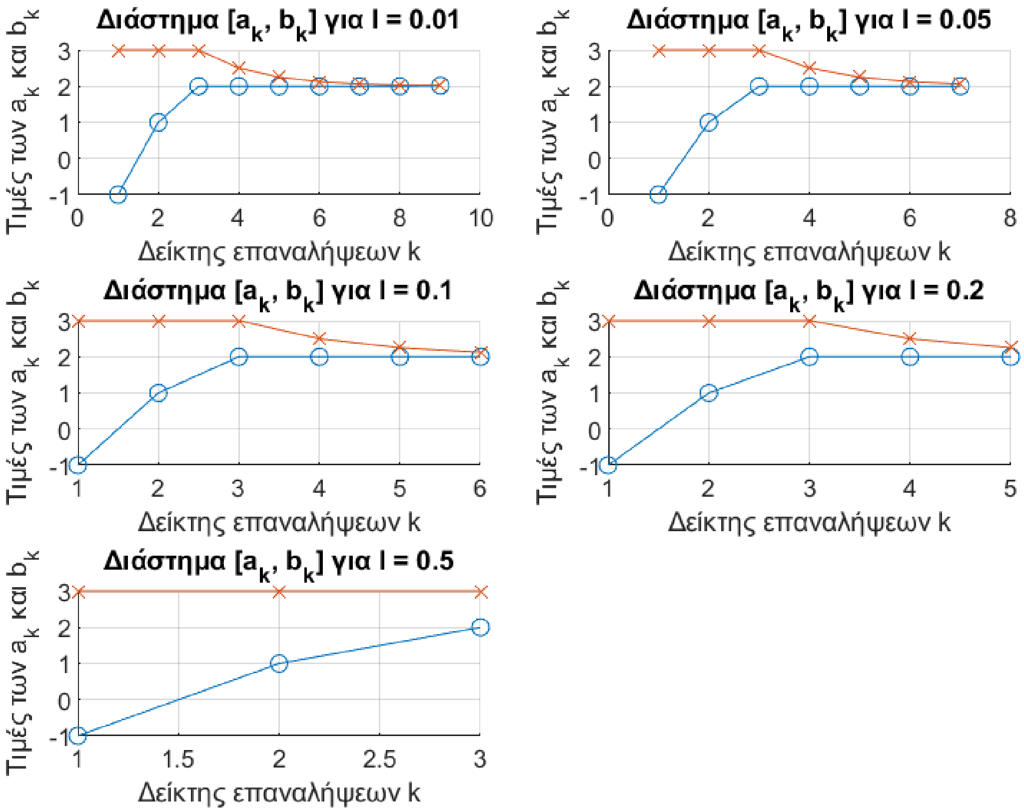


- οι γραφικές παραστάσεις των άκρων του διαστήματος [ak,bk] συναρτήσει του δείκτη επαναλήψεων k, δηλαδή (k,ak)και (k,bk),για διάφορες τιμές του τελικού εύρους αναζήτησης l παρατίθενται παρακάτω*(τα αρχεία στο φάκελο matlab είναι τα εξής: dixmeparagogous2_f1.m , dixmeparagogous2_f2.m , dixmeparagogous2_f3.m)* :

Για την f1(x):

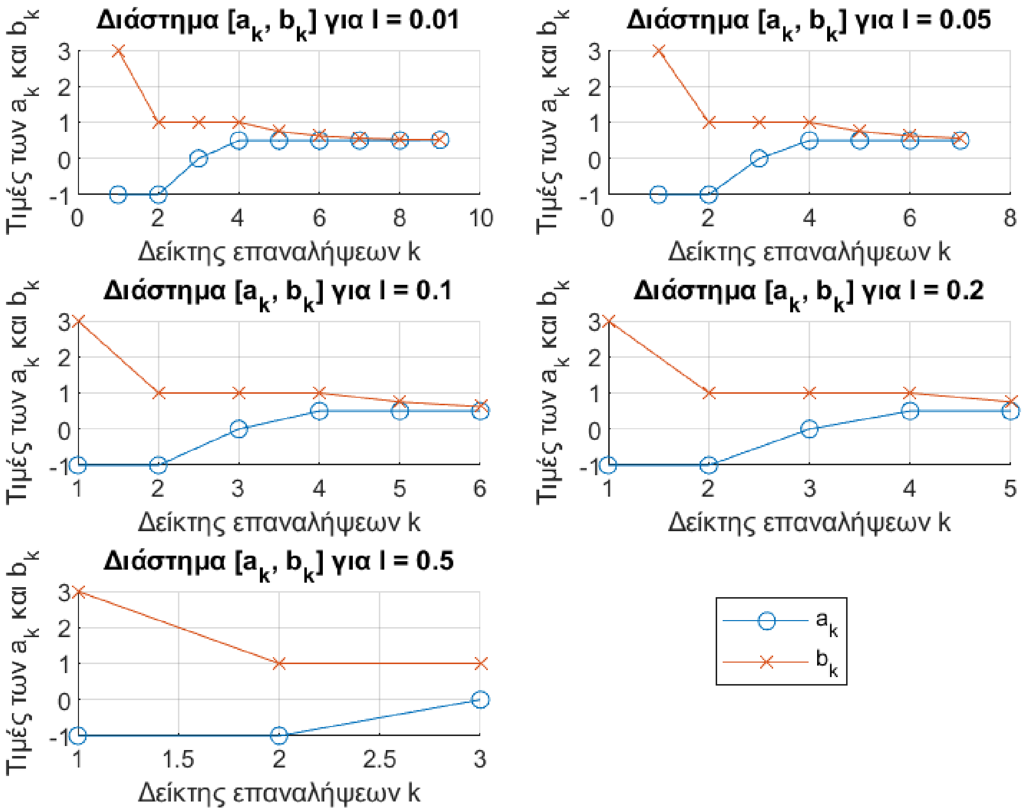


Για την f2(x):



Διάστημα [a_k, b_k] για l = 0.01
Διάστημα [a_k, b_k] για l = 0.05
Διάστημα [a_k, b_k] για l = 0.1
Διάστημα [a_k, b_k] για l = 0.2
Διάστημα [a_k, b_k] για l = 0.5
Τιμές των a_k και b_k
Δείκτης επαναλήψεων k

Για την f3(x):



Επειδή στην μέθοδο αυτή υπολογίζουμε την παράγωγο f′(x) σε ένα σημείο του διαστήματος και χρησιμοποιούμε το πρόσημο της παραγώγου για να καθορίσουμε αν το ελάχιστο βρίσκεται αριστερά ή δεξιά αυτού του σημείου,η διαδικασία αυτή επιτρέπει στην αναζήτηση να γίνεται πιο «στοχευμένα» και συνήθως απαιτεί λιγότερους υπολογισμούς.Η πληροφορία της παραγώγου μειώνει το διάστημα αναζήτησης πιο γρήγορα, καθώς κατευθύνει τον αλγόριθμο απευθείας προς την κατεύθυνση του ελαχίστου.Οι άλλες μέθοδοι (διχοτόμηση χωρίς παράγωγο, χρυσή τομή, Fibonacci) είναι μέθοδοι χωρίς παράγωγους, που σημαίνει ότι βασίζονται μόνο στην αξιολόγηση της τιμής της συνάρτησης σε δύο ή περισσότερα σημεία κάθε φορά,οπως είδαμε.Οι συναρτήσεις που μας δίνονται έχουν εύκολη παράγωγο,γεγονός που καθιστά την μέθοδο αυτή την πιο αποδοτική.Αυτό φαίνεται ξεκάθαρα απο το πλήθος των συνολικών υπολογισμών των fi(x) όπου στα διαγράμματα των άκρων [ak,bk], για τη μικρότερη τιμή του l(η ελάχιστη τιμη του l που χρησιμοποιώ στους κώδικες είναι η l=0.01) εχει λιγότερους απο 10 υπολογισμούς των fi(x),ενω η μεθοδος της διχοτόμου εχει μεχρι και 10 και οι μέθοδοι χρυσής τομής και Fibonacci ισοδύναμα σχεδον τείνουν στις 14 επαναλήψεις.Ετσι,σε περιπτώσεις όπου η παράγωγος της συνάρτησης είναι διαθέσιμη και υπολογίζεται εύκολα, η μέθοδος διχοτόμησης με παράγωγο είναι πολύ αποδοτική και προτιμάται. Βέβαια να σημειωθεί οτι η διχοτόμηση με παράγωγο απαιτεί υπολογισμό της παραγώγου σε κάθε βήμα, που είναι πιο δαπανηρός από έναν απλό υπολογισμό της συνάρτησης αν η παράγωγος είναι δύσκολη.