<u>ΤΕΧΝΙΚΕΣ ΒΕΛΤΙΣΤΟΠΟΙΗΣΗΣ</u>

1η Εργαστηριακή Άσκηση

Ονοματεπώνυμο: Φωτιάδης Αλέξανδρος

ΑΕΜ:10392

E-mail: afotiadis@ece.auth.gr

Σε αυτό το report παρουσιάζονται οι λύσεις για την 1η εργαστηριακή άσκηση του μαθήματος, τα ζητούμενα διαγράμματα και ο σχολιασμός και τα συμπεράσματα που εξάγονται από τα αποτελέσματα.

Αντικείμενο της εργασίας είναι η ελαχιστοποίηση μονοδιάστατων, κυρτών συναρτήσεων σε δοσμένο διάστημα με τη χρήση επαναληπτικών αλγορίθμων. Μας ζητήθηκε λοιπόν να ελαχιστοποιήσουμε στο δοσμένο διάστημα [-1,3] τις παρακάτω συναρτήσεις:

$$f_1(x) = (x-2)^2 + x * \ln(x+3)$$

$$f_2(x) = e^{(-2*x)} + (x-2)^2$$

$$f_3(x) = e^x * (x^3 - 1) + (x-1) * \sin(x)$$

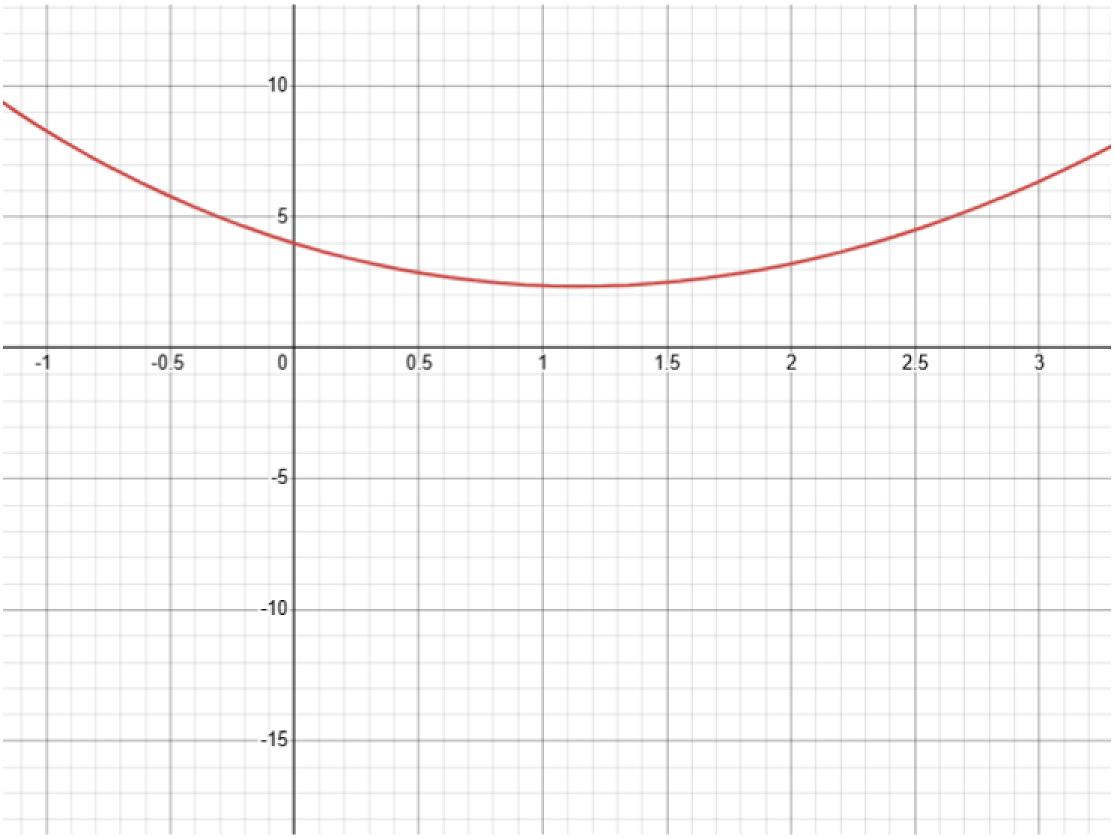
<u>ΣΗΜΕΙΩΣΗ**:</u>** Μόλις παρατήρησα ότι έχω κάνει λάθος στο latex και στους τίτλους στα διαγράμματα γράφω

$$f_3(x) = e^{(x^3-1)} + (x-1) * \sin(x)$$

αντί για την σωστή μορφή της f_3. Ωστόσο αυτό είναι τυπογραφικό λάθος, στο σώμα του κώδικα είναι ορισμένη σωστά!

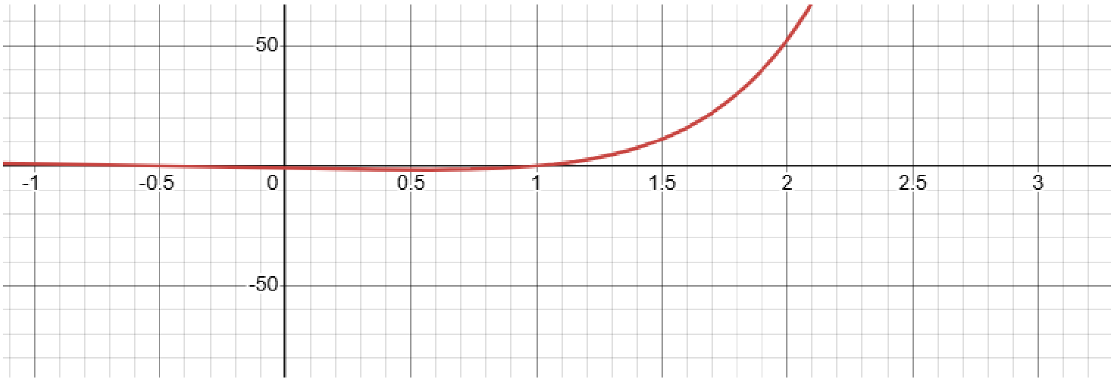
Παρακάτω, χρησιμοποιώντας το desmos graphing calculator έγινε η απεικόνιση των συναρτήσεων στο δοσμένο διάστημα, ώστε να οπτικοποιηθεί πιο εύκολα το πρόβλημα.

$f_1(x)$:



$f_2(x)$:

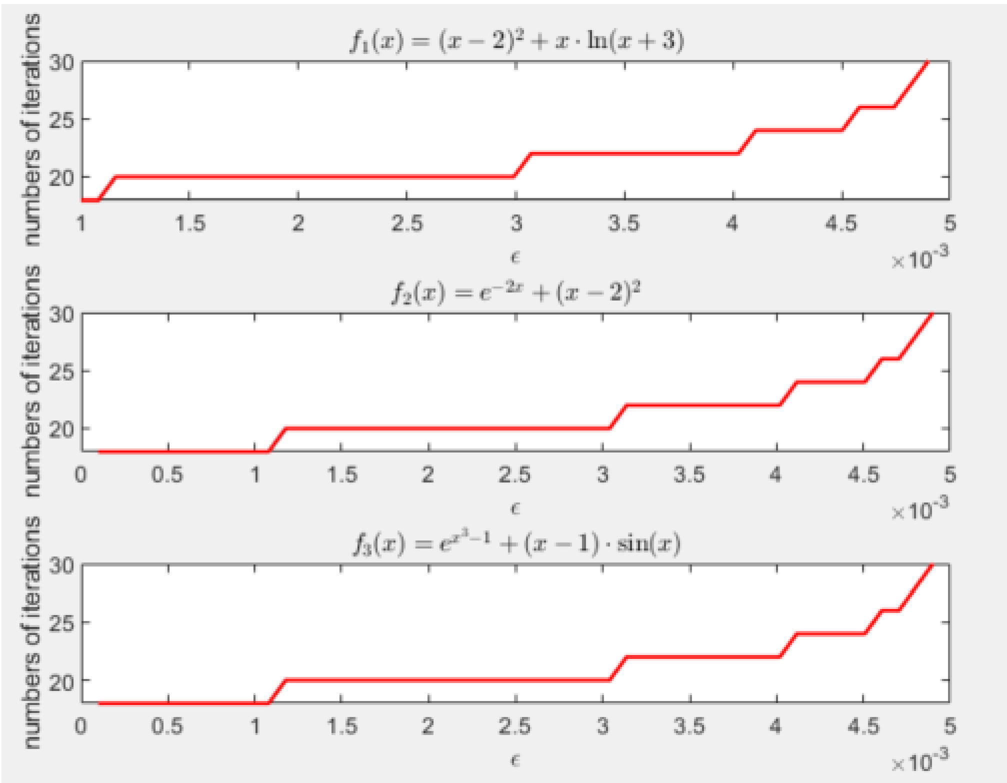
$f_3(x)$:



ΘΕΜΑ 1:

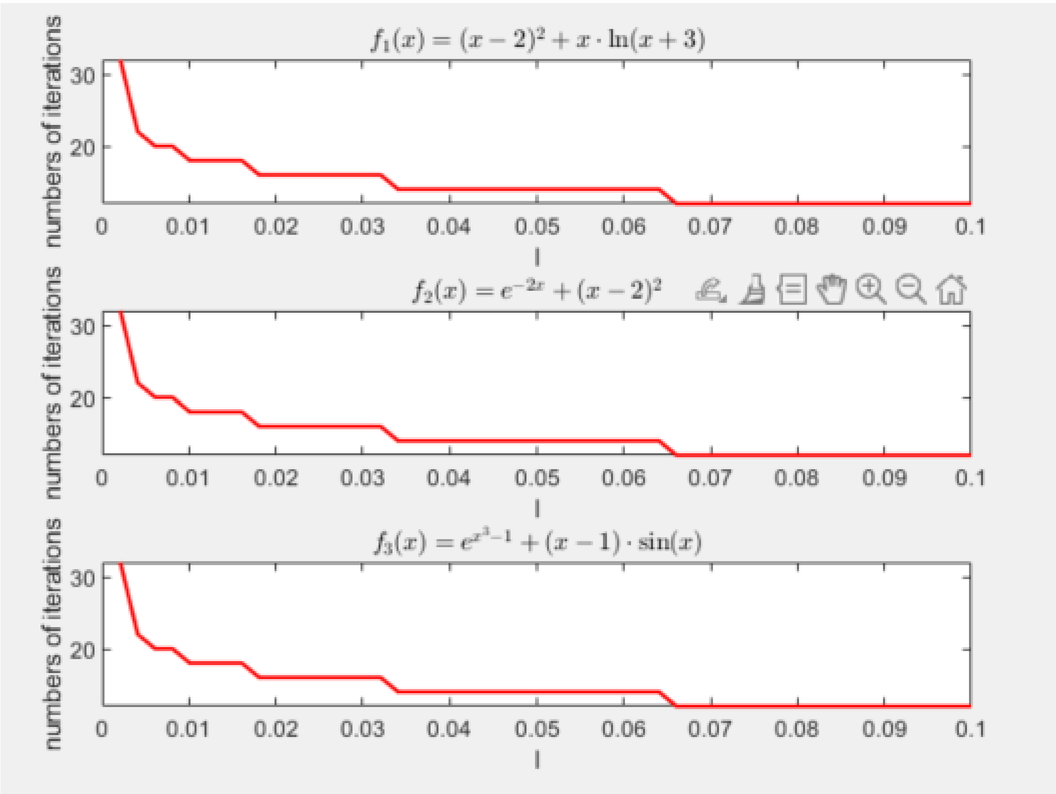
Μέθοδος της διχοτόμου:

➢ Κρατώντας σταθερό το τελικό εύρος αναζήτησης l=0.01, τρέχουμε προσομοιώσεις ώστε να βρούμε πόσες φορές υπολογίζεται η τιμή της συνάρτησης $f_i(x)$ (i=1,2,3) για διάφορες τιμές της σταθεράς e η οποία συμβολίζει την απόσταση από τη διχοτόμο. Το e διατρέχει τις τιμές του διαστήματος 0.0001 έως 0.0049, εξασφαλίζοντας έτσι την σύγκλιση της μεθόδου καθώς για να συγκλίνει η μέθοδος πρέπει να τηρείται η συνθήκη l>2e. (bisection_optimization.m)



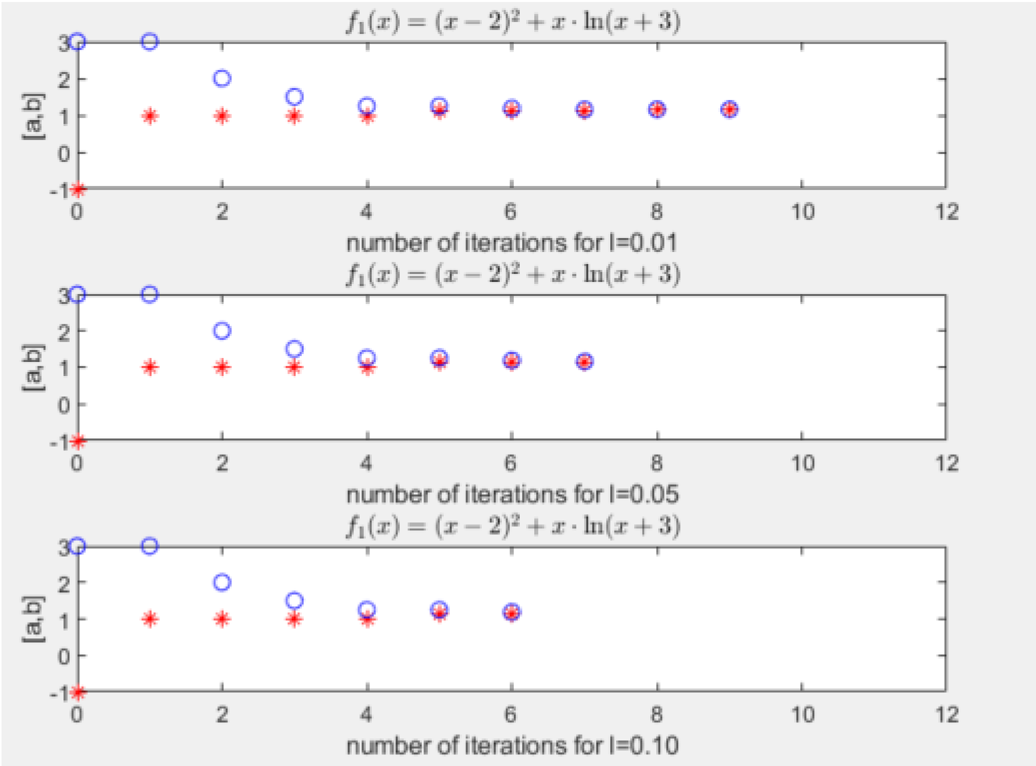
Όπως βλέπουμε τα γραφήματα, όσο αυξάνεται η τιμή του e, τόσο αυξάνονται και οι κλήσεις της εκάστοτε συνάρτησης $f_i(x)$. Αυτό συμβαίνει γιατί με την αύξηση της e που συμβολίζει την απόσταση από τη διχοτόμο, τα x1(k) και x2(k) υπολογίζονται πιο μακριά από τη διχοτόμο του διαστήματος [a(k),b(k)] και πιο κοντά στα άκρα του, με αποτέλεσμα το διάστημα [a(k+1),b(k+1)] που προκύπτει μετά από κάθε επανάληψη k να έχει μεγαλύτερο εύρος. Ως συνέπεια, αυξάνονται τα βήματα που απαιτούνται από τον αλγόριθμο για να φτάσουμε στο επιθυμητό εύρος διαστήματος l. Αυτό συνεπάγεται περισσότερες κλήσεις της $f_i(x)$.

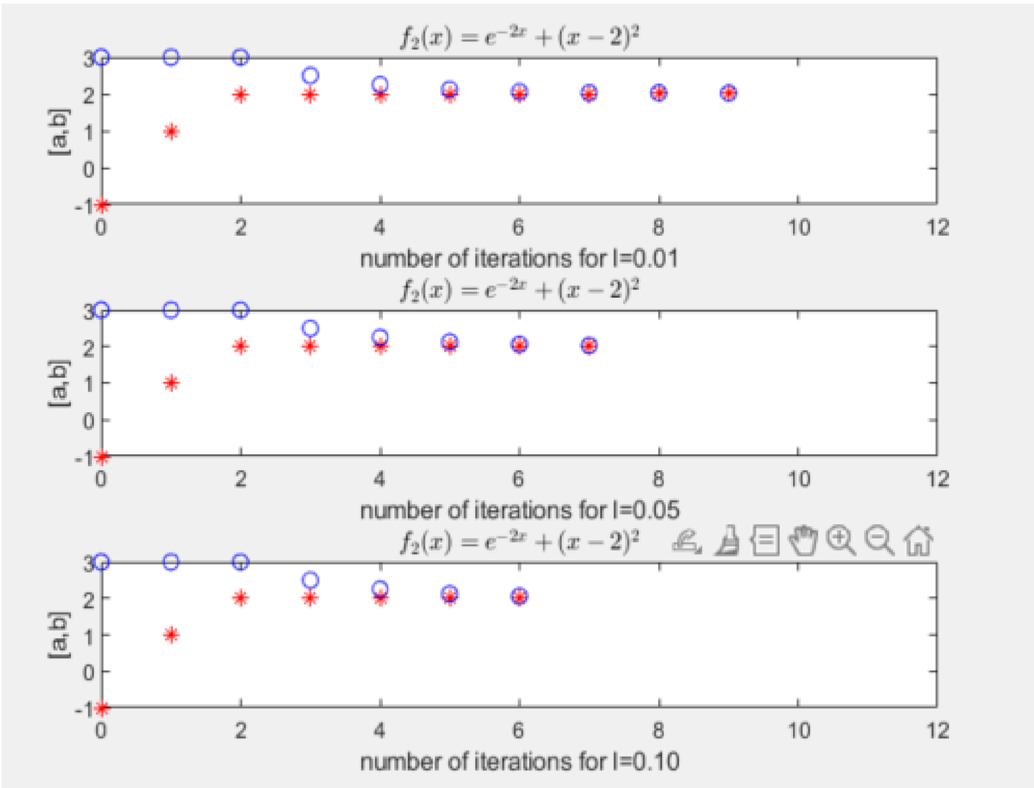
- Κρατώντας σταθερή την απόσταση από τη διχοτόμο e=0.001, τρέχουμε προσομοιώσεις ώστε να βρούμε πόσες φορές υπολογίζεται η τιμή της συνάρτησης $f_i(x)$ (i=1,2,3) για διάφορες τιμές του εύρους τελικού διαστήματος l. Το l διατρέχει τις τιμές του διαστήματος 0.0021 έως 0.1, εξασφαλίζοντας έτσι την σύγκλιση της μεθόδου καθώς για να συγκλίνει η μέθοδος πρέπει να τηρείται η συνθήκη l>2e. (bisection_optimization.m)

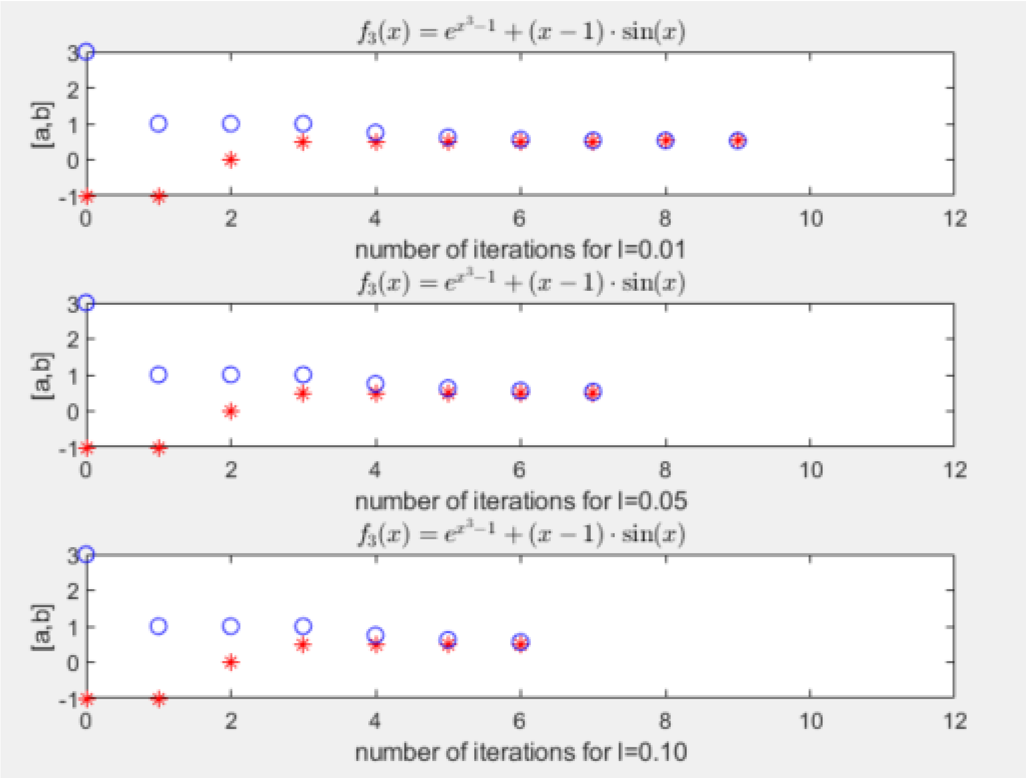


Όπως βλέπουμε τα γραφήματα, όσο αυξάνεται η τιμή του l, τόσο μειώνονται και οι κλήσεις της εκάστοτε συνάρτησης $f_i(x)$. Αυτό συμβαίνει γιατί με την αύξηση της l, αυξάνεται και το εύρος του τελικού διαστήματος το οποίο ορίζεται ως επιθυμητό για να σταματήσει να τρέχει ο αλγόριθμος. Συνεπώς, μια αύξηση του l είναι λογικό να οδηγεί σε λιγότερες διαδοχικές διαμερίσεις και κατ' επέκταση σε λιγότερες κλήσεις της αντικειμενικής συνάρτησης

➢ Στο τελευταίο ερώτημα μας ζητείται να μελετήσουμε πώς μεταβάλλονται τα άκρα του διαστήματος για διάφορες τιμές του εύρους l.





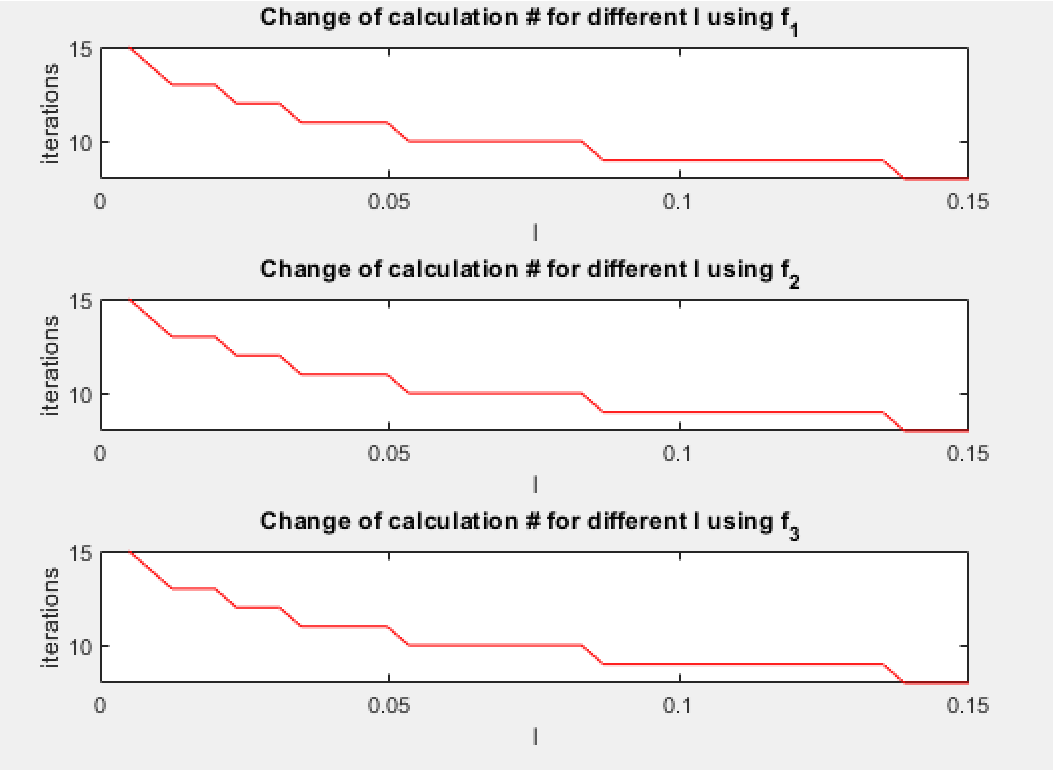


Όπως γίνεται αντιληπτό από τα γραφήματα, και όπως ήταν αναμενόμενο, όσο μικρότερο είναι το εύρος του τελικού διαστήματος l (και συνεπώς τόσο μεγαλύτερη ακρίβεια επιζητούμε), τόσο περισσότερες επαναλήψεις χρειάζονται για να το επιτύχουμε.

<u>ΘΕΜΑ 2:</u>

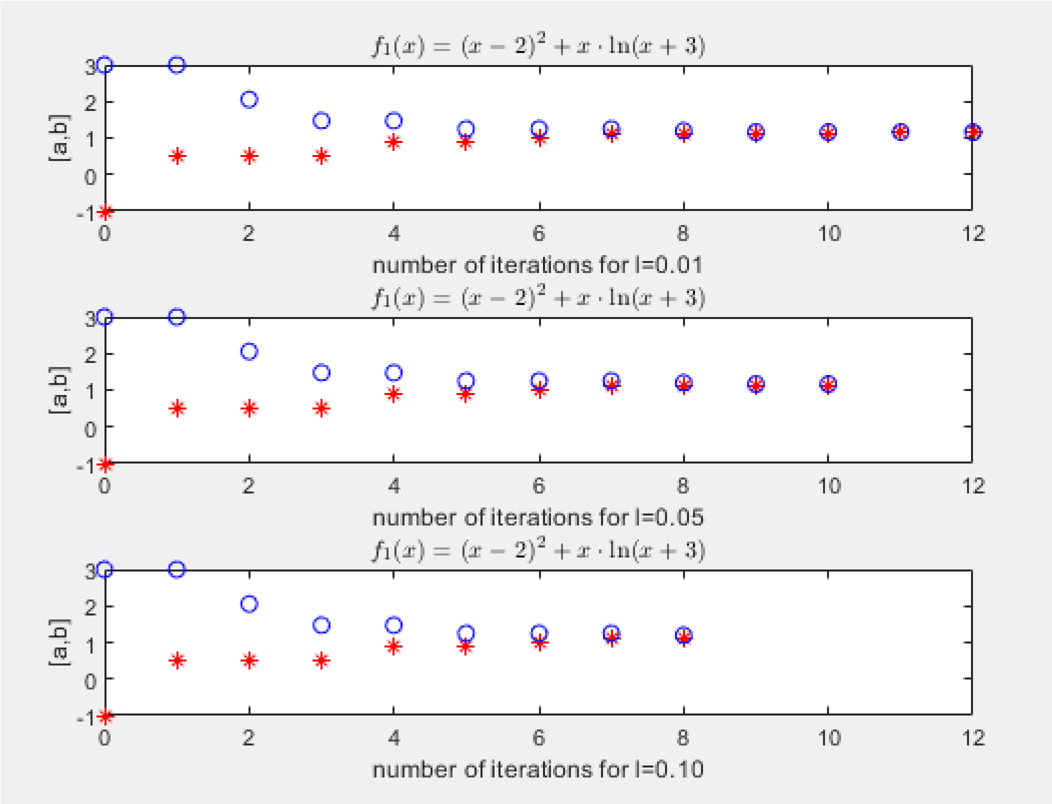
Μέθοδος χρυσού τομέα:

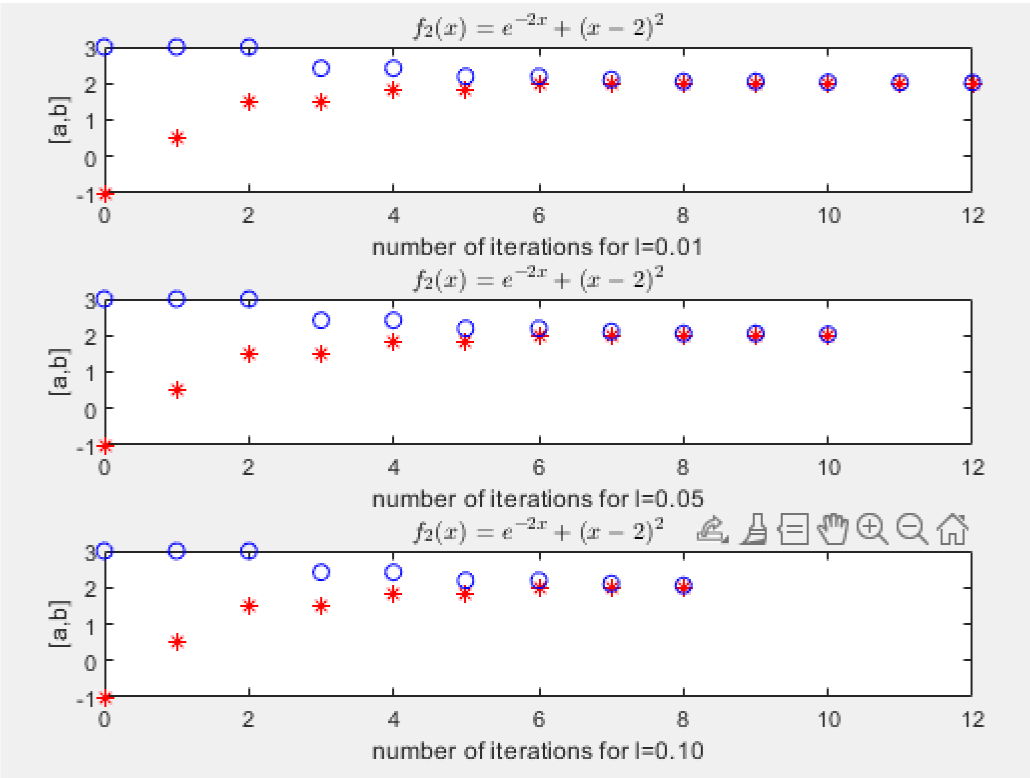
- Μελετάμε το πόσες φορές υπολογίζεται τιμή της αντικειμενικής συνάρτησης f για διάφορες τιμές του τελικού εύρους διαστήματος l. Το τελικό εύρος διαστήματος l μεταβάλλεται από 0.005 μέχρι 0.15 .(golden_section_optimization.m)

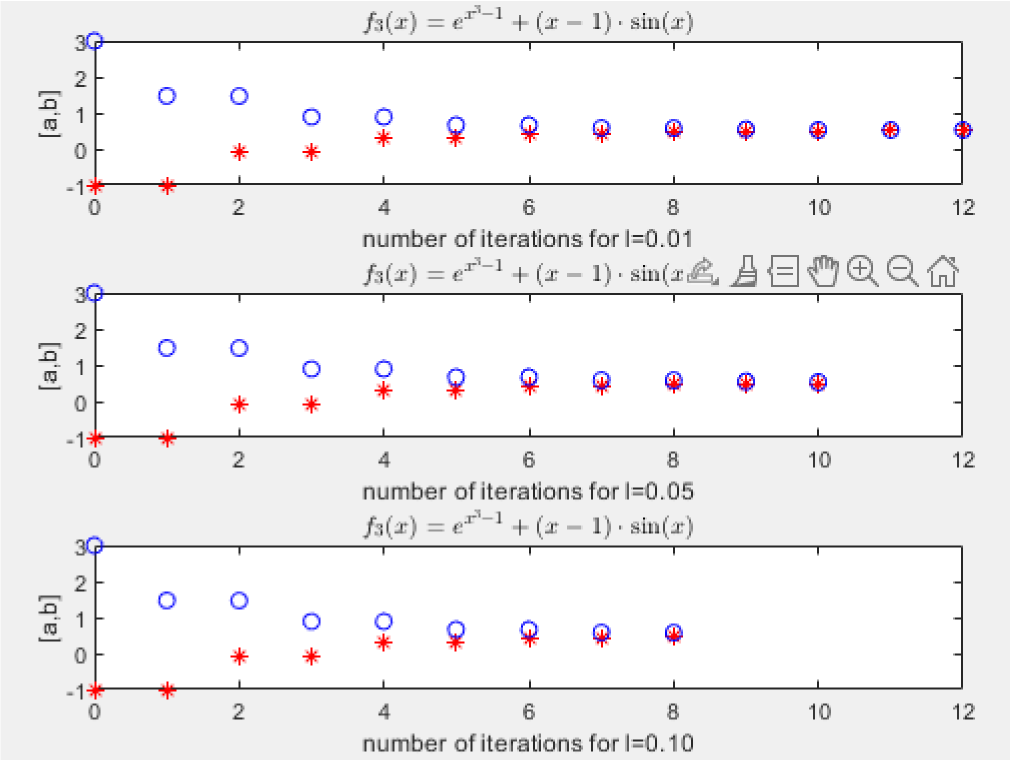


Όπως γίνεται αντιληπτό και από τα διαγράμματα, καθώς αυξάνεται το τελικό εύρος l μειώνεται ο αριθμός των κλήσεων και υπολογισμών της αντικειμενικής συνάρτησης $f_i(x)$ (i=1,2,3). Το αποτέλεσμα αυτό είναι αναμενόμενο καθώς η αύξηση του τελικού εύρους συνεπάγεται την μείωση της επιθυμητής ακρίβειας της τελικής λύσης και αυτό με τη σειρά του οδηγεί σε λιγότερους υπολογισμούς της αντικειμενικής συνάρτησης.

- Σε αυτό το ερώτημα μελετάμε τα τελικά άκρα του διαστήματος $[a_k,b_k]$ ως συνάρτηση του k-οστου βήματος, για 3 ενδεικτικές τιμές της παραμέτρου l=[0.01,0.05,0.1] (golden_section_optimization.m).


$f_1(x) = (x-2)^2 + x \cdot \ln(x+3)$
[a,b]
number of iterations for l=0.01
$f_1(x) = (x-2)^2 + x \cdot \ln(x+3)$
[a,b]
number of iterations for l=0.05
$f_1(x) = (x-2)^2 + x \cdot \ln(x+3)$
[a,b]
number of iterations for l=0.10


$f_2(x) = e^{-2x} + (x-2)^2$
[a,b]
number of iterations for l=0.01
$f_2(x) = e^{-2x} + (x-2)^2$
[a,b]
number of iterations for l=0.05
$f_2(x) = e^{-2x} + (x-2)^2$
[a,b]
number of iterations for l=0.10

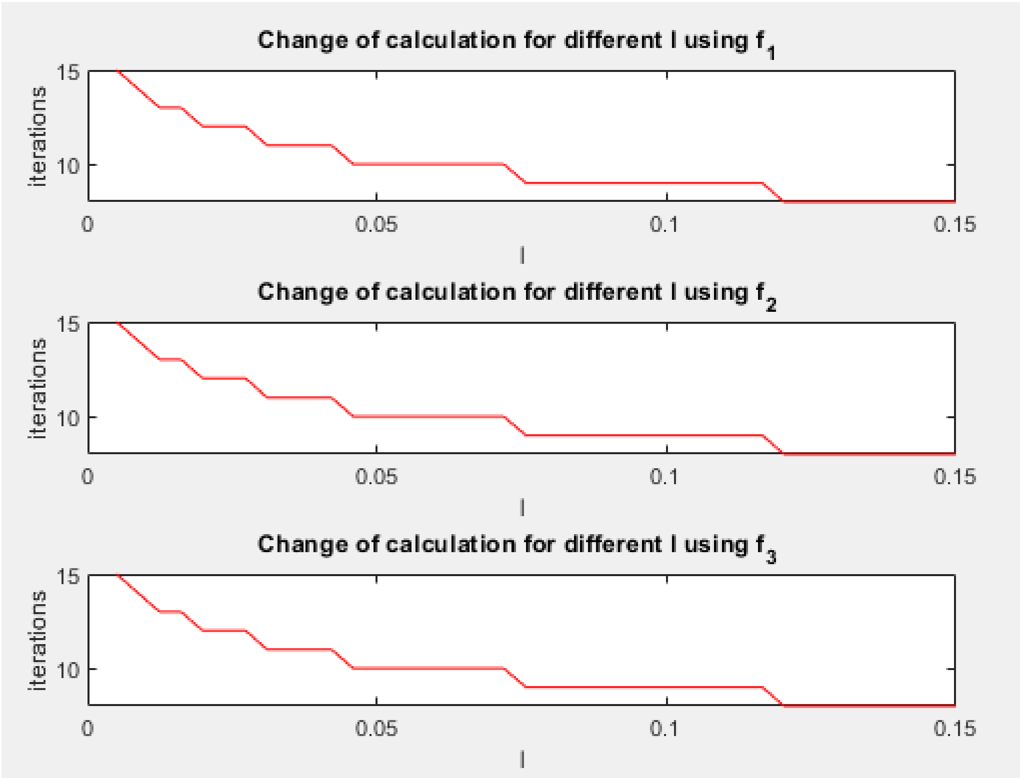


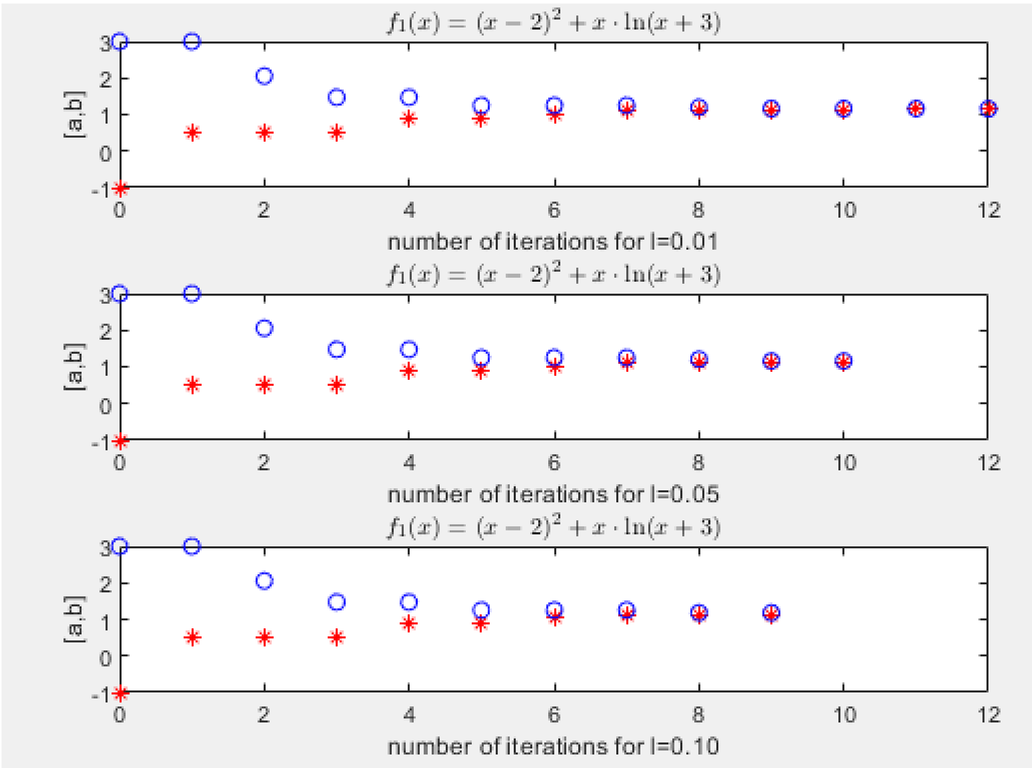
Και σε αυτή την περίπτωση καταλήγουμε στο συμπέρασμα ότι όσο παραπάνω μειώνεται το εύρος του τελικού διαστήματος l τόσες περισσότερες επαναλήψεις χρειάζεται η μέθοδος ώστε να καταλήξει στην επιθυμητή , ορισμένη από εμάς ακρίβεια.

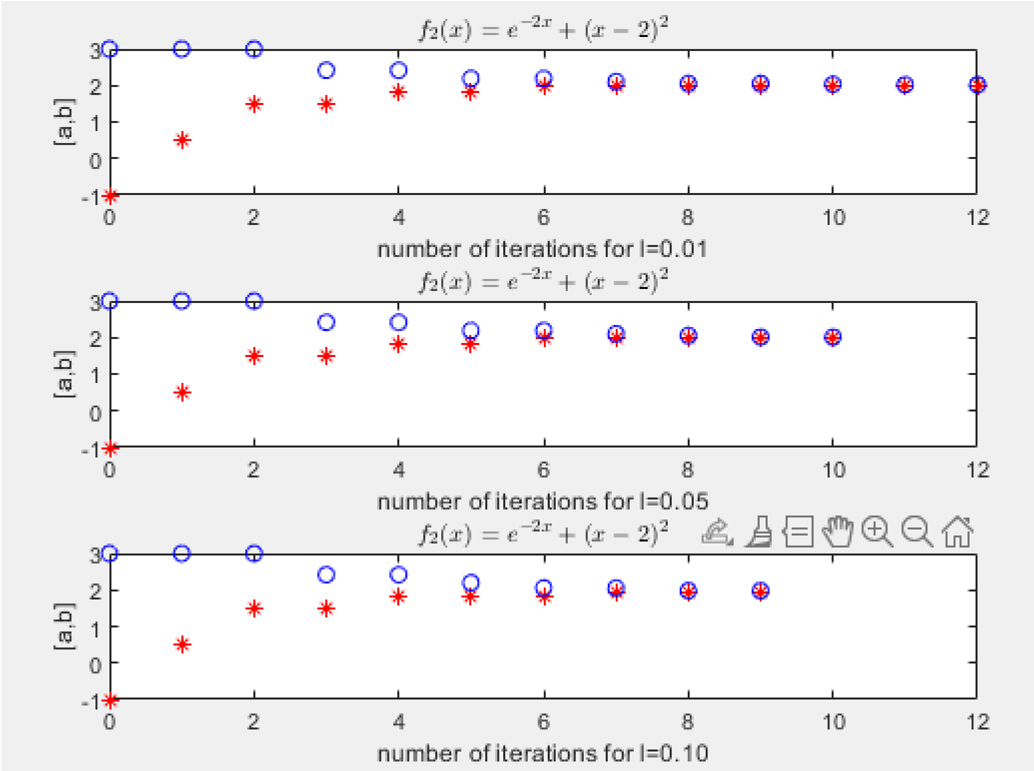
<u>ΘΕΜΑ 3:</u>

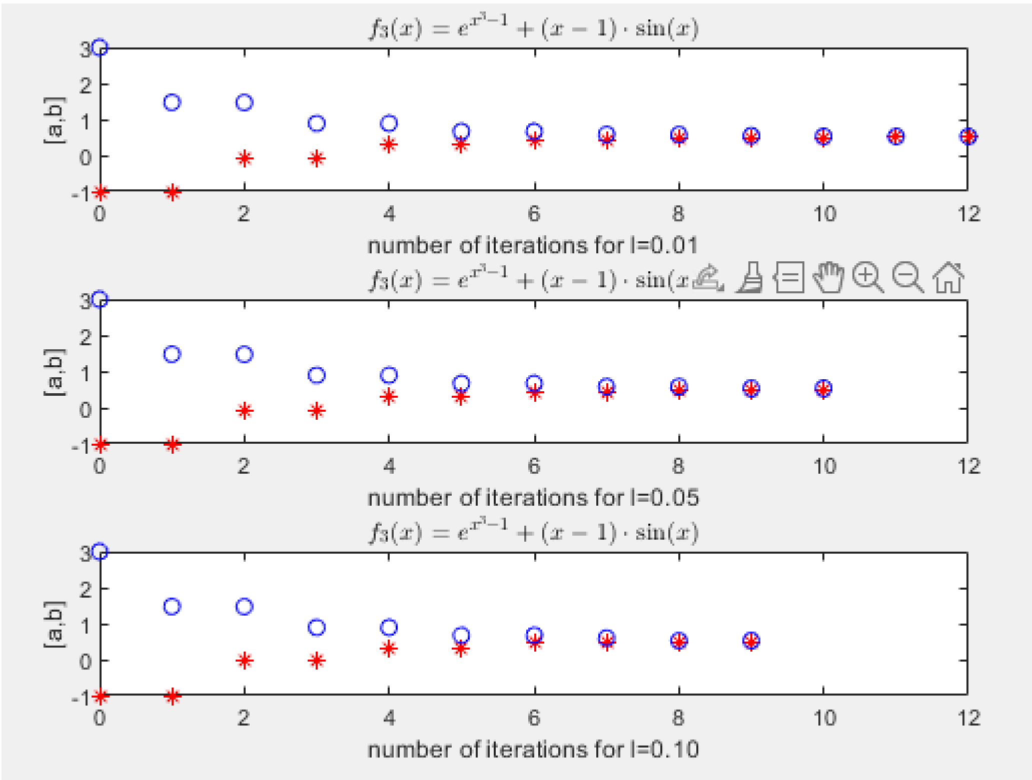
Μέθοδος Fibonacci:

Η μέθοδος αυτή μοιάζει αρκετά με την μέθοδο που παρουσιάστηκε στο προηγούμενο θέμα, παρόλα αυτά ο διαμερισμός των διαστημάτων που προκύπτουν κατά την εκτέλεσή της δεν συνδέονται με κάποια σταθερά. Αντιθέτως, μεταβάλλονται από επανάληψη σε επανάληψη βασιζόμενα στην ακολουθία αριθμών Fibonacci. Τα ζητούμενα σε αυτό το θέμα είναι τα ίδια με αυτά του θέματος 2 οπότε παρακάτω παρουσιάζονται τα αντίστοιχα διαγράμματα(fibonacci_optimization.m):


Change of calculation for different l using f_1
iterations
l
Change of calculation for different l using f_2
iterations
l
Change of calculation for different l using f_3
iterations
l


$f_1(x) = (x-2)^2 + x \cdot \ln(x+3)$
[a,b]
number of iterations for l=0.01
$f_1(x) = (x-2)^2 + x \cdot \ln(x+3)$
[a,b]
number of iterations for l=0.05
$f_1(x) = (x-2)^2 + x \cdot \ln(x+3)$
[a,b]
number of iterations for l=0.10





ΠΑΡΑΤΗΡΗΣΕΙΣ:

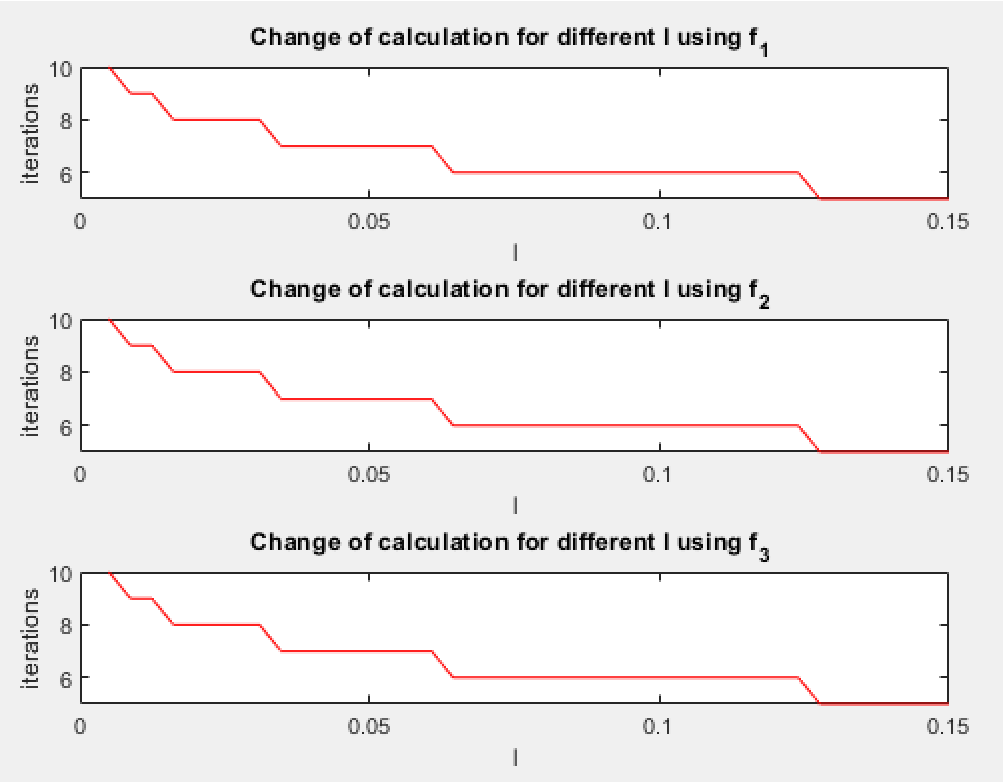
- Τα διαγράμματα που απεικονίζουν την μεταβολή του αριθμού κλήσεων της αντικειμενικής συνάρτησης $f_i(x)$ είναι πανομοιότυπα μεταξύ τους για κάθε i=1,2,3. Αυτό φυσικά δεν πρέπει να μας εκπλήσσει καθώς σε όλες τις περιπτώσεις οι υπολογισμοί των αντικειμενικών συναρτήσεων εξαρτώνται από τις ίδιες παραμέτρους (l και αρχικό διάστημα αναζήτησης).

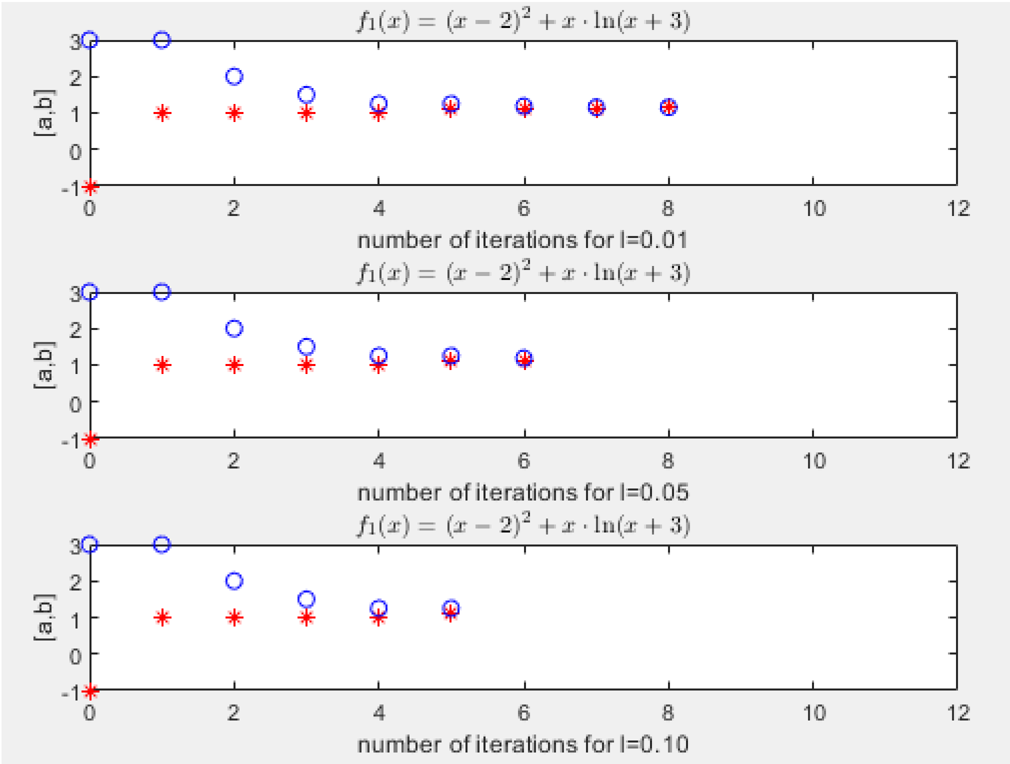
- Καθώς αυξάνουμε το l, ο αλγόριθμος συμπεριφέρεται όπως και στις προηγούμενες περιπτώσεις δηλαδή μειώνονται οι επαναλήψεις που απαιτούνται για να φτάσουμε στην τελική λύση.
- Ενδεικτικά, για να επιβεβαιώσουμε τον τύπο παραγράφου 5.1 από την βιβλιογραφία, παρατηρούμε ότι για περίπου l =0.04 έχουμε n = 12 υπολογισμούς της αντικειμενικής συνάρτησης.

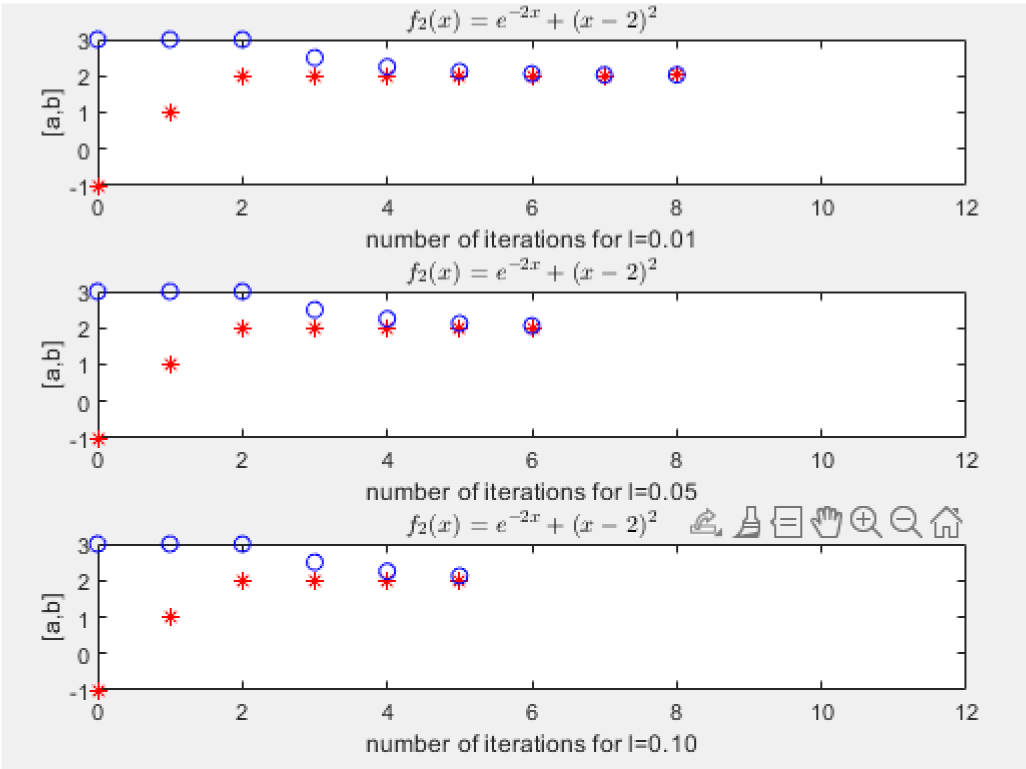
ΘΕΜΑ 4:

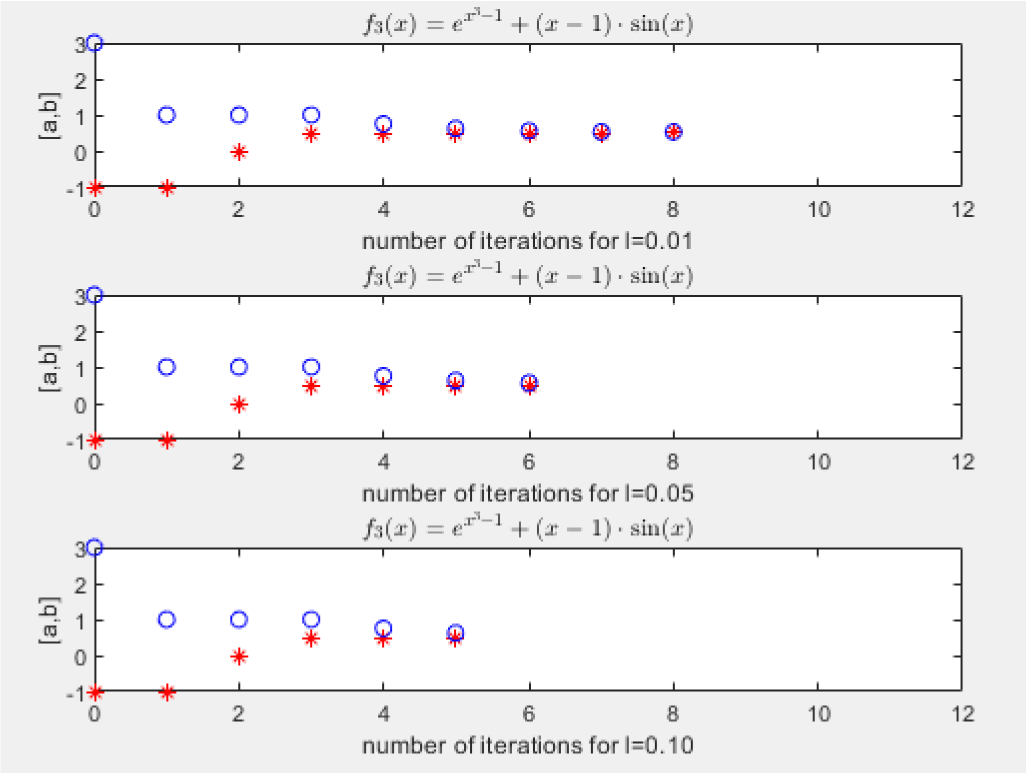
Μέθοδος διχοτόμου με χρήση παραγώγων:

Η μέθοδος αυτή ομοιάζει με αυτή που παρουσιάστηκε στο θέμα 1 καθώς βασίζεται στη διχοτόμηση του αρχικού διαστήματος αναζήτησης, ωστόσο διαφοροποιείται από το γεγονός ότι καταλήγει στην τελική λύση χρησιμοποιώντας την παράγωγο πρώτης τάξης για την $f_i(x)$. Τα ζητούμενα και εδώ είναι ίδια με αυτά των θεμάτων 2 και 3 οπότε προχωράμε κατευθείαν στα διαγράμματα (bisection_w_der_opt.m):




$f_1(x) = (x-2)^2 + x \cdot \ln(x+3)$
[a,b]
number of iterations for l=0.01
$f_1(x) = (x-2)^2 + x \cdot \ln(x+3)$
[a,b]
number of iterations for l=0.05
$f_1(x) = (x-2)^2 + x \cdot \ln(x+3)$
[a,b]
number of iterations for l=0.10

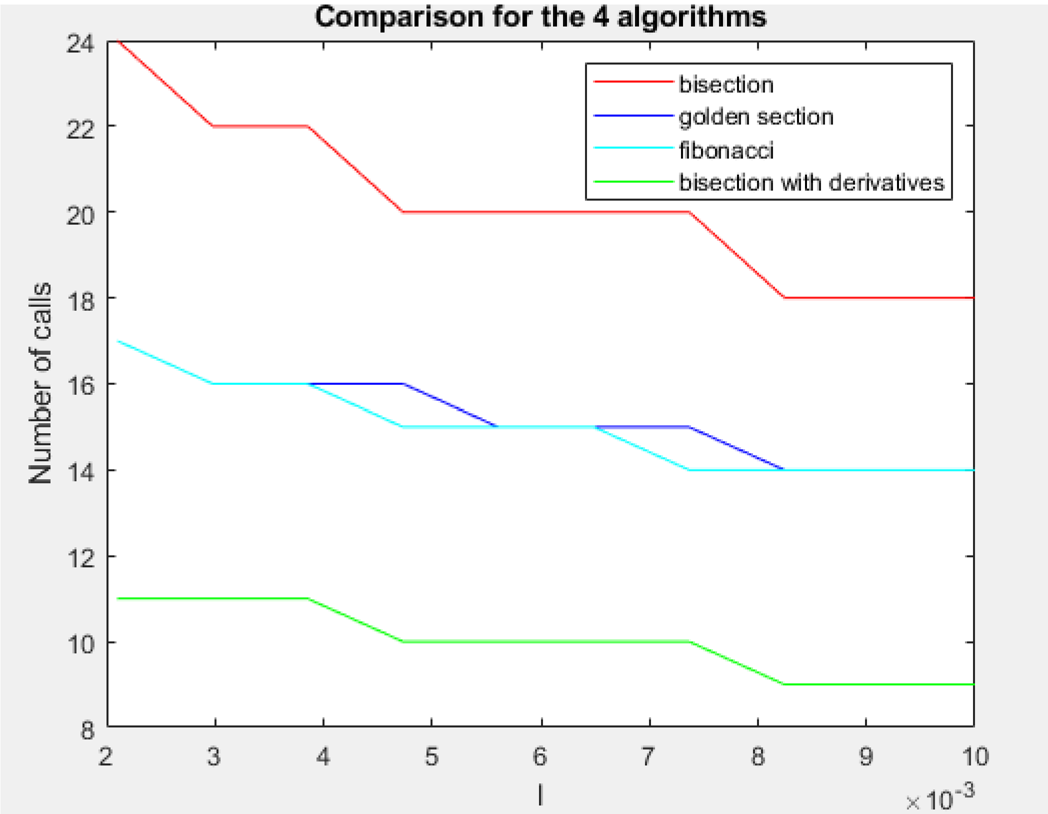

$f_2(x) = e^{-2x} + (x-2)^2$
[a,b]
number of iterations for l=0.01
$f_2(x) = e^{-2x} + (x-2)^2$
[a,b]
number of iterations for l=0.05
$f_2(x) = e^{-2x} + (x-2)^2$
[a,b]
number of iterations for l=0.10



- Καθώς αυξάνουμε το l, ο αλγόριθμος συμπεριφέρεται όπως και στις προηγούμενες περιπτώσεις δηλαδή μειώνονται οι επαναλήψεις που απαιτούνται για να φτάσουμε στην τελική λύση μιας και οι απαιτήσεις μας γίνονται πιο «ευέλικτες».

<u>ΣΥΜΠΕΡΑΣΜΑΤΑ:</u>

Για να οπτικοποιήσουμε και να κάνουμε πιο εύκολα αντιληπτά τα συμπεράσματα που προέκυψαν από την παραπάνω άσκηση, δημιουργήθηκε το πρόγραμμα comparing_the_methods.m από το οποίο προκύπτει το παρακάτω διάγραμμα που δείχνει τους υπολογισμούς της αντικειμενικής συνάρτησης f που κάνει ο κάθε αλγόριθμος μέχρι να φτάσει στην τελική λύση. Φυσικά, για να είναι σωστά τα συμπεράσματά μας, όλοι οι αλγόριθμοι δοκιμάστηκαν με την ίδια συνάρτηση και με τις ίδιες παραμέτρους.



Για τους υπολογισμούς που χρειάζεται κάθε μέθοδος λοιπόν ισχύει:

διχοτόμος με παράγωγο < Fibonacci ≈ χρυσός τομέας < διχοτόμος χωρίς παράγωγο

Παρατηρούμε ότι οι επαναλήψεις της μεθόδου Fibonacci ταυτίζονται με αυτές της μεθόδου του χρυσού τομέα. Αυτό το περιμέναμε καθώς χρησιμοποιούμε μικρές τιμές l και έτσι το n είναι μεγάλο. Από την θεωρία για n μεγάλο αυτές οι δύο μέθοδοι είναι σχεδόν ταυτόσημες. Ένα πλεονέκτημα όμως του αλγορίθμου Fibonacci είναι ότι απαιτεί προεπιλογή του αριθμού επαναλήψεων που θα χρειαστούν το οποίο με την σωστή χρήση κώδικα μπορεί να τον κάνει πιο αποδοτικό σε πολύ μεγάλα δείγματα (δέσμευση χώρου για μεταβλητές που αλλάζουν μέγεθος, απουσία μεταβλητής counter που αυξάνει με κάθε επανάληψη κλπ.).

Παρατηρούμε ακόμη ότι όσο το l αυξάνεται (κρατώντας το e fixed, όπου αυτό χρησιμοποιείται), οι απαιτούμενες επαναλήψεις μειώνονται. Για σταθερό l, στην μέθοδο της διχοτόμου με παράγωγο, όσο αυξάνουμε το e, οι απαιτούμενες επαναλήψεις αυξάνονται. Δηλαδή, όσο πιο απαιτητικοί είμαστε από άποψη ακρίβειας (l μικρό) τόσο περισσότερες επαναλήψεις θα χρειαστούν για την εύρεση του διαστήματος όπου η συνάρτηση ελαχιστοποιείται. Αντίστοιχα, όσο πιο μεγάλη είναι η απόστασή μας από τη διχοτόμο (e μεγάλο), ο αλγόριθμος θα προσεγγίσει πιο αργά το διάστημα ελαχίστου.