Τεχνικές Βελτιστοποίησης - 1η Εργαστηριακή Άσκηση

Ελαχιστοποίηση κυρτής συνάρτησης μιας μεταβλητής σε δοσμένο διάστημα

Δάφνη Νικολαΐδου ΑΕΜ:10546

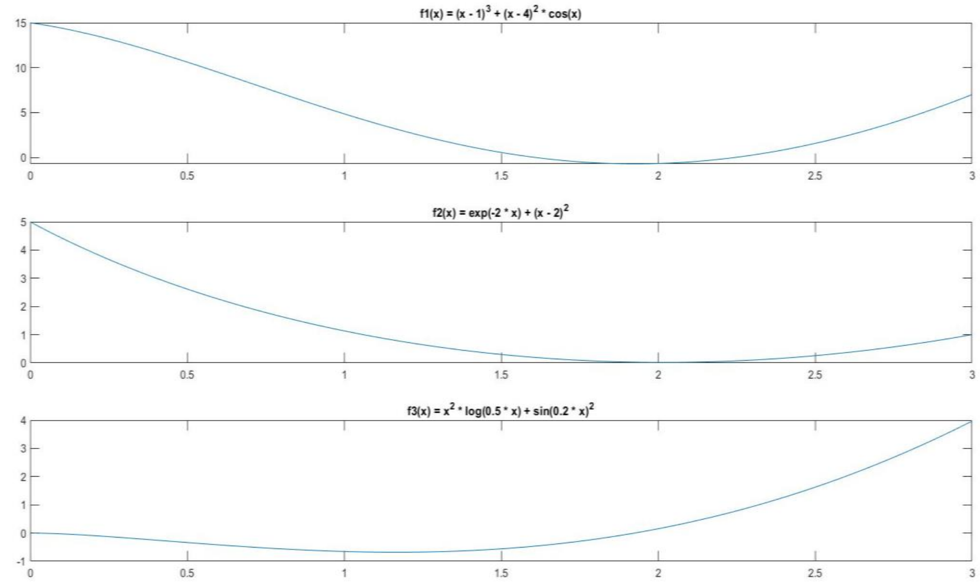
Σε αυτή την άσκηση υλοποιούνται στο MATLAB 4 μέθοδοι για την ελαχιστοποίηση των δοθέντων συναρτήσεων:

$$f1(x) = (x-1)^3 + (x-1)^2 \cos(x)$$

$$f2(x) = e^{-2x} + (x-2)^2$$

$$f2(x) = x^2 \ln(0.5x) + \sin^2(0.2x)$$

Πριν ξεκινήσουμε τη μελέτη, διαπιστώνουμε τη κυρτότητα των συναρτήσεων στο αρχικό διάστημα [0,3] από τις γραφικές τους παραστάσεις.



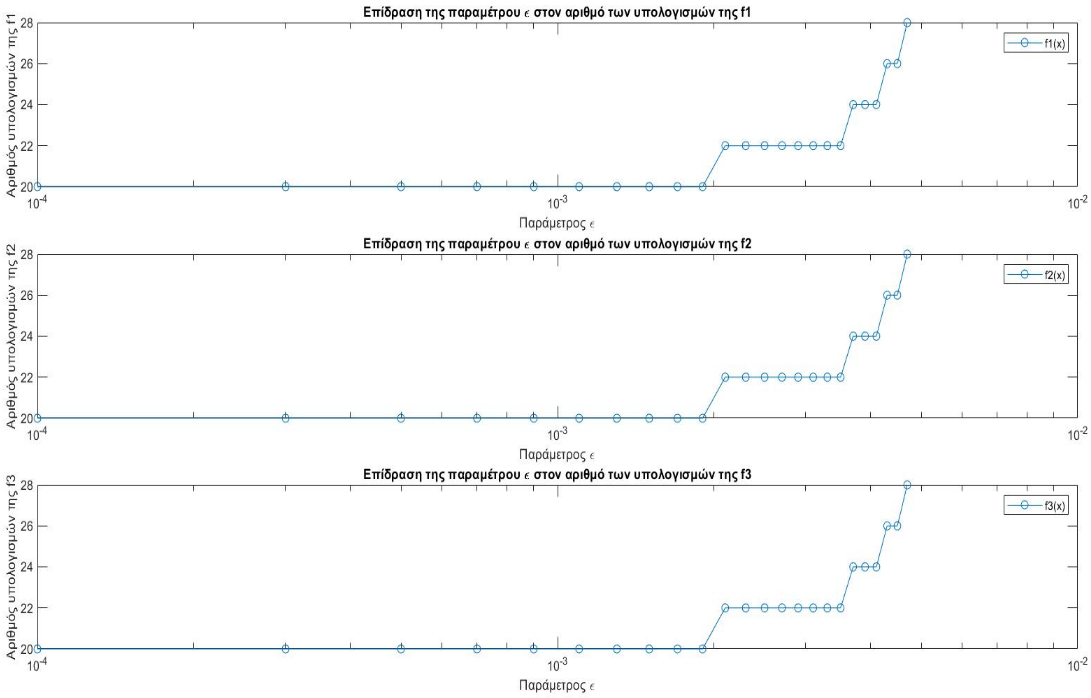
Μέθοδος της Διχοτόμου

Η μέθοδος αυτή χωρίζει το αρχικό διάστημα σε ίσα υποδιαστήματα $x1$, $x2$ συμμετρικά σε απόσταση ε > 0 από την διχοτόμο του. Έπειτα, περιορίζει το διάστημα αναζήτησης του ελαχίστου της συνάρτησης στο [α, $x2$) ή το ($x1$, β] ανάλογα με το αν $f(x1)$ < $f(x2)$ ή $f(x1)$ > $f(x2)$ αντίστοιχα, έως ότου το διάστημα να έχει εύρος μικρότερο από τη τελική ακρίβεια l.

Το αρχείο MATLAB για την υλοποίησή της έχει όνομα thema1.m

Μελετούμε τη μεταβολή των υπολογισμών της αντικειμενικής συνάρτησης για κάθε μια από τις f1,f2, f3 για τις εξής περιπτώσεις:

- Κρατώντας σταθερό το l = 0.01 και μεταβάλλοντας το ε το οποίο παίρνει τιμές ε = 0.0001 έως 0.0048 με βήμα 0.0002, καθώς πρέπει να τηρείται ο περιορισμός l >2ε, έχουμε ως αποτέλεσμα τις παρακάτω γραφικές παραστάσεις

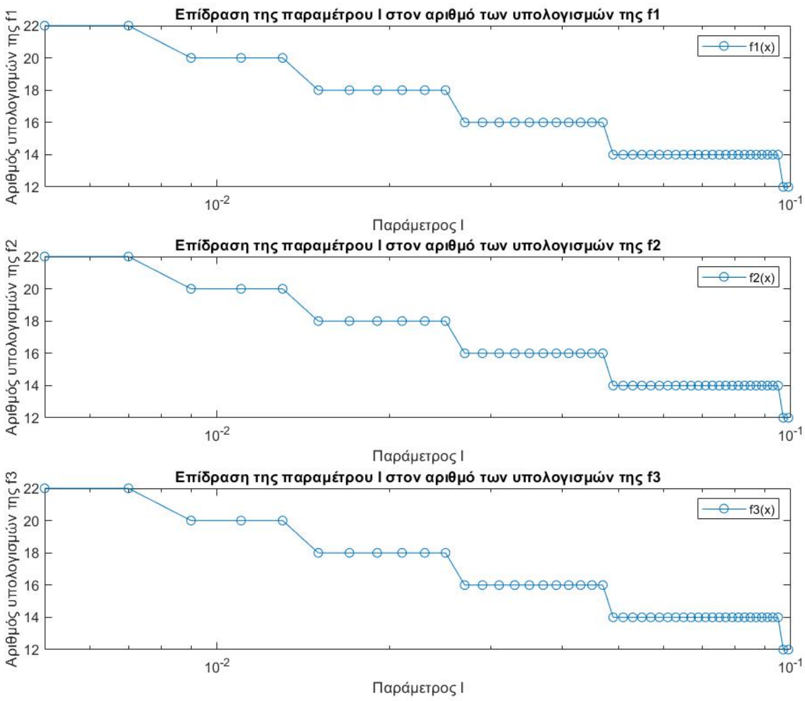


Ο οριζόντιος άξονας απεικονίζει τη μεταβολή του ε, ενώ ο κατακόρυφος τον αριθμό των υπολογισμών της f.

Παρατηρούμε πως όσο μεγαλώνει το ε τόσο περισσότερους υπολογισμούς της αντικειμενικής συνάρτησης χρειαζόμαστε για να καταλήξουμε στο επιθυμητό αποτέλεσμα.

Διαπιστώνουμε, επίσης, πως το γράφημα είναι το ίδιο και για τις τρεις συναρτήσεις. Δηλαδή, το κόστος σε κλήσεις της f δεν εξαρτάται από την ίδια τη συνάρτηση, επομένως ούτε και από τη θέση του ελαχίστου της. Και στις τρεις περιπτώσεις αυτά που επηρεάζουν τους υπολογισμούς της αντικειμενικής συνάρτησης είναι l, ε και διάστημα αναζήτησης.

- Κρατώντας τώρα σταθερό ε = 0.001 και μεταβάλλοντας το l το οποίο παίρνει τιμές από 0.005 έως 0.1 με βήμα 0.002 τηρώντας τον περιορισμό l >2ε, έχουμε ως αποτέλεσμα τα εξής γραφήματα:,

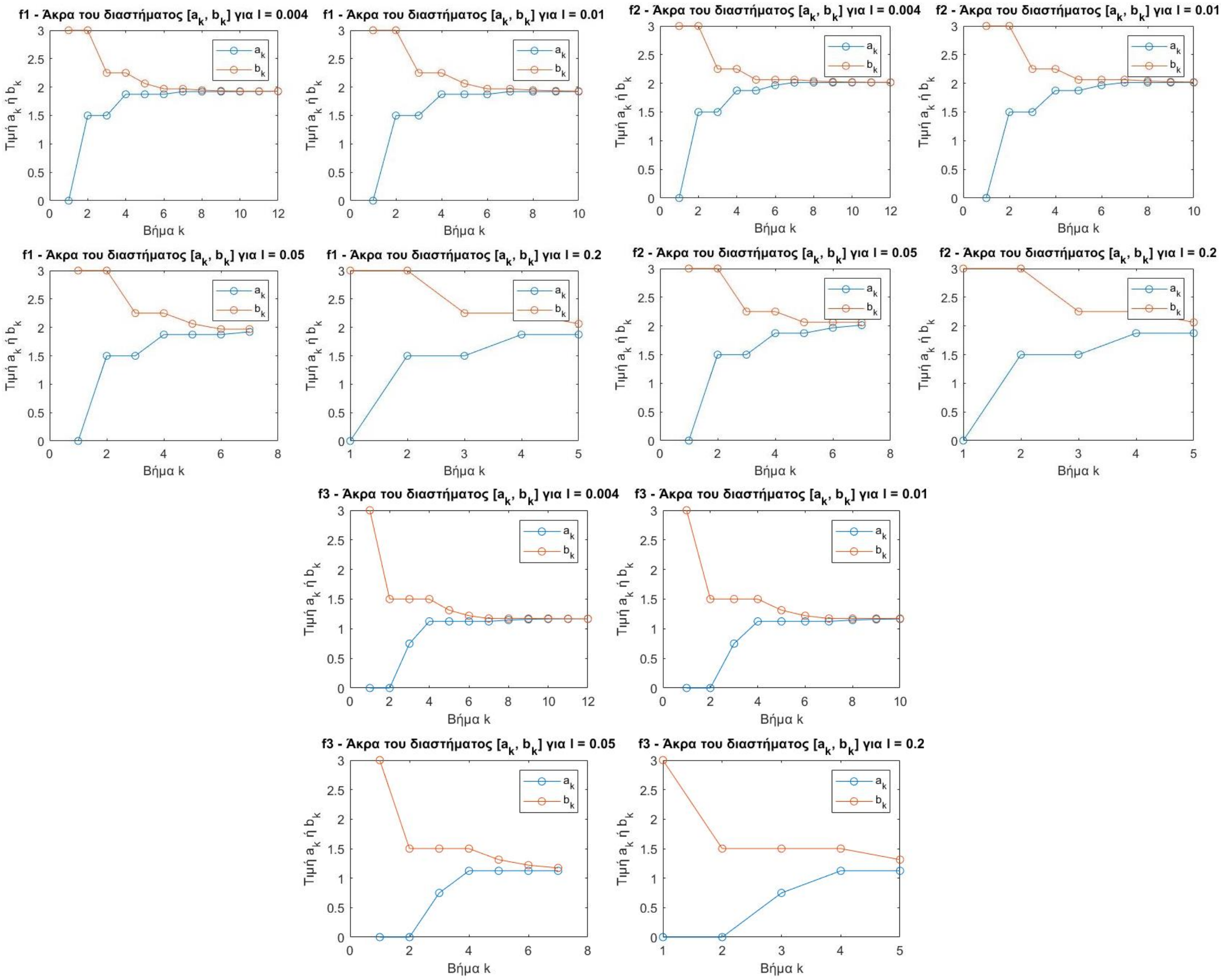


Αυτή τη φορά, ο οριζόντιος άξονας απεικονίζει τη μεταβολή του l, ενώ ο κατακόρυφος τον αριθμό των υπολογισμών της f.

Όπως θα δούμε, όσο μεγαλώνει το l, τόσο λιγότεροι υπολογισμοί της f χρειάζονται, κάτι λογικό αφού μεγαλύτερο l σημαίνει ότι απαιτείται λιγότερη ακρίβεια, με αποτέλεσμα να χρειαστούν λιγότεροι υπολογισμοί για να φτάσουμε σε ένα αποδεκτό εύρος.

Και σε αυτή τη περίπτωση, βλέπουμε ότι η μορφή της ίδιας της συνάρτησης δεν επηρεάζει το αποτέλεσμα.

Ας δούμε τώρα, πως διαμορφώνεται το διάστημα αναζήτησης μετά από κάθε επανάληψη k του αλγορίθμου συναρτήσει και αυτή τη φορά, του l. Δοκιμάζουμε για τιμές l = [0.004, 0.01, 0.05, 0.2]



Τα διαγράμματα $[\alpha\kappa, b\kappa]$ ανά επανάληψη κ , δείχνουν τον περιορισμό του εύρους του κάθε διαστήματος και τη σύγκλιση του αλγόριθμου στην λύση.

Φαίνεται, επιπλέον, ότι για μεγαλύτερο l (μικρότερη τελική ακρίβεια), η σύγκλιση γίνεται πιο γρήγορα (σε λιγότερες επαναλήψεις του αλγορίθμου)

Μέθοδος του Χρυσού Τομέα

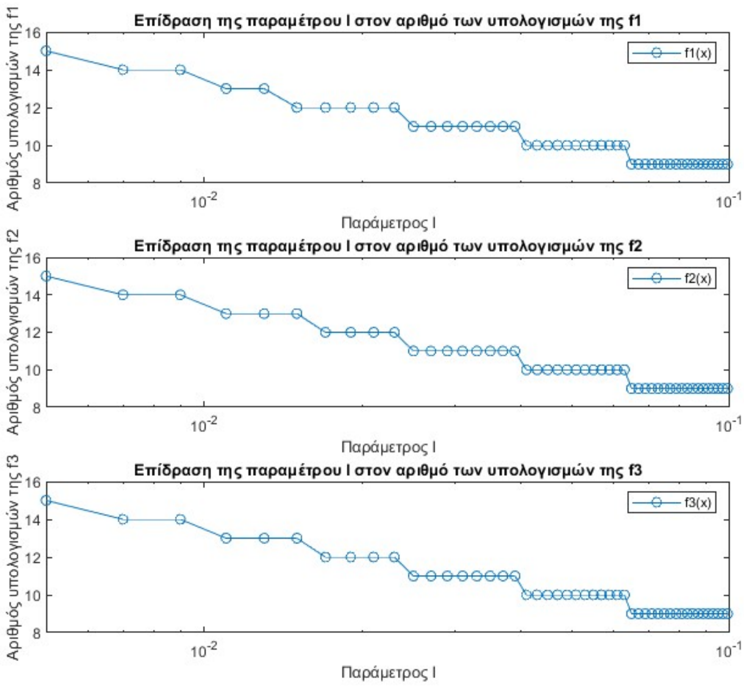
Σ’ αυτή την μέθοδο τα δύο διαστήματα που προκύπτουν από την k-οστή επανάληψη, ικανοποιούν την σχέση:

$$\frac{bk+1-ak+1}{bk-ak} = \gamma \text{ , όπου } \gamma = 0.618 \text{ η χρυσή αναλογία}$$

Επιλέγουμε $x1k = ak + (1-\gamma)(bk-ak)$ και $x2k = ak + \gamma(bk-ak)$. Ανάλογα με το αν $f(x1) < f(x2)$ ή $f(x1) > f(x2)$, σε κάθε νέα επανάληψη θα έχουμε $x1k+1$ και $x2k+1$ τέτοια ώστε$x1k+1 = x2k$ ή $x2k+1 = x1k$, ενώ το διάστημα θα περιορίζεται στο $[\alpha, x2)$ ή $(x1, \beta]$ αντίστοιχα.

Το αρχείο MATLAB για την υλοποίησή της έχει όνομα thema2.m

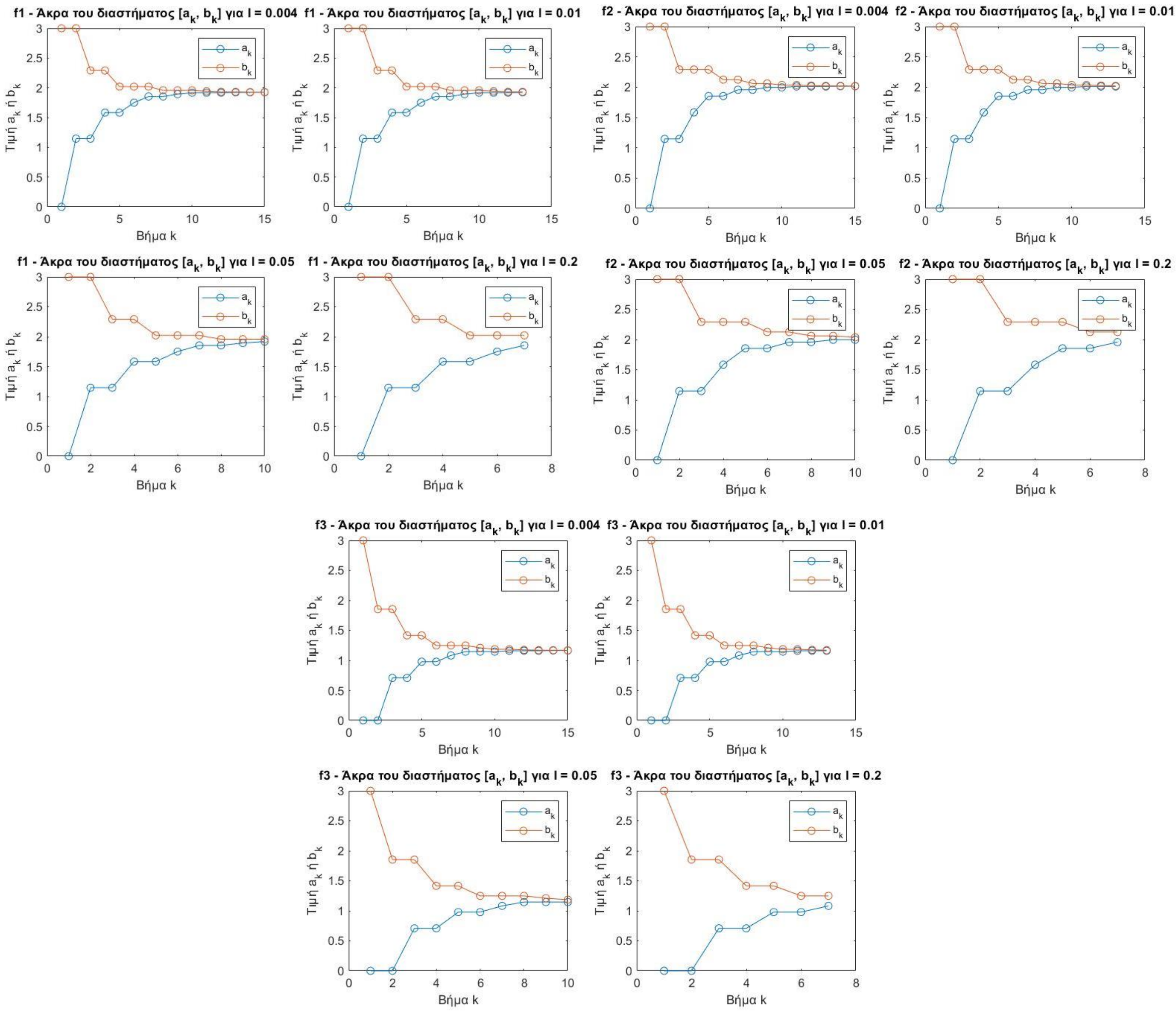
Μελετούμε τη μεταβολή των υπολογισμών της αντικειμενικής συνάρτησης για κάθε μια από τις f1,f2, f3, μεταβάλλοντας το l το οποίο παίρνει τιμές από 0.005 έως 0.1 με βήμα 0.002 και έχουμε ως αποτέλεσμα τα εξής γραφήματα:



Ομοίως με πριν, όσο μεγαλώνει το l, τόσο λιγότεροι υπολογισμοί της f χρειάζονται, κάτι λογικό αφού μεγαλύτερο l σημαίνει ότι απαιτείται λιγότερη ακρίβεια με αποτέλεσμα να χρειαστούν λιγότεροι υπολογισμοί για να φτάσουμε σε ένα αποδεκτό εύρος.

Και σε αυτή τη περίπτωση, βλέπουμε ότι η μορφή της ίδιας της συνάρτησης δεν επηρεάζει το αποτέλεσμα.

Ας δούμε τώρα, πως διαμορφώνεται το διάστημα αναζήτησης μετά από κάθε επανάληψη k του αλγορίθμου συναρτήσει και αυτή τη φορά, του l. Δοκιμάζουμε για τιμές l = [0.004, 0.01, 0.05, 0.02]



Τα διαγράμματα $[a\kappa, b\kappa]$ ανά επανάληψη κ, δείχνουν τον περιορισμό του εύρους του κάθε διαστήματος και τη σύγκλιση του αλγόριθμου στην λύση.

Φαίνεται, επιπλέον, ότι για μεγαλύτερο l (μικρότερη τελική ακρίβεια), η σύγκλιση γίνεται πιο γρήγορα, δηλαδή μετά από λιγότερες επαναλήψεις του αλγορίθμου.

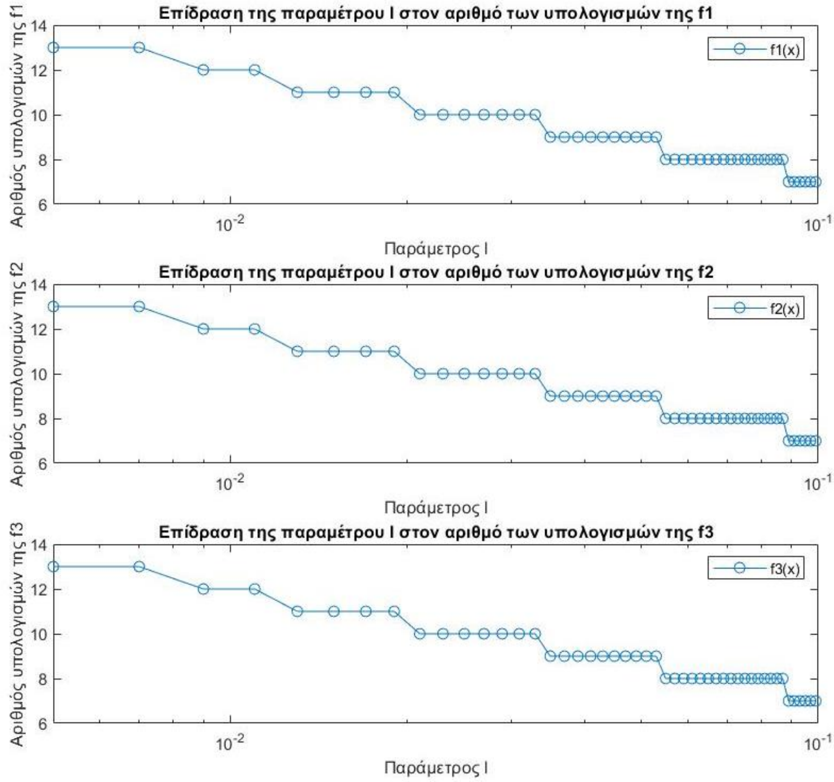
Μέθοδος Fibonacci

Η μέθοδος Fibonacci μοιάζει αρκετά με τη Μέθοδο του Χρυσού Τομέα. Αυτή τη φορά, όμως, τα υποδιαστήματα που προκύπτουν μετά από κάθε επανάληψη υπολογίζονται βάσει της ακολουθίας Fibonacci.

Ισχύει ο περιορισμός $Fn \leq \frac{b-a}{l} \leq Fn + 1$ για τον αριθμό των επαναλήψεων n.

Το αρχείο MATLAB για την υλοποίησή της έχει όνομα thema3.m

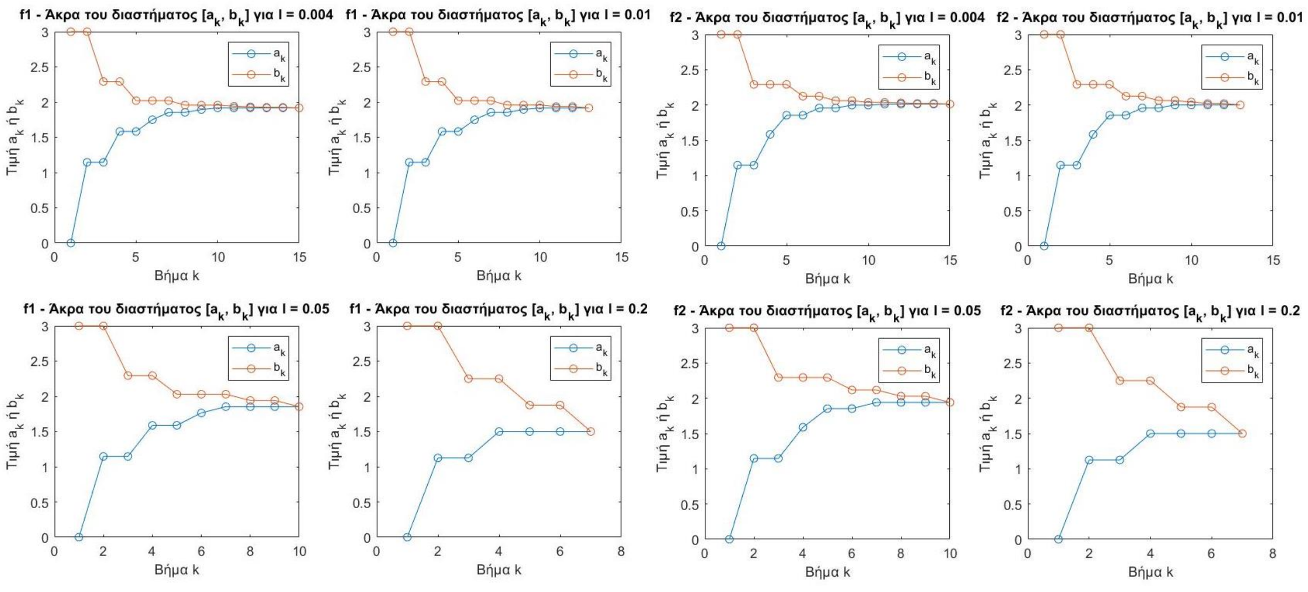
Μελετούμε τη μεταβολή των υπολογισμών της αντικειμενικής συνάρτησης για κάθε μια από τις f1,f2, f3, μεταβάλλοντας το l ώστε να παίρνει τιμές από 0.005 έως 0.1 με βήμα 0.002, έχουμε ως αποτέλεσμα τα εξής γραφήματα:

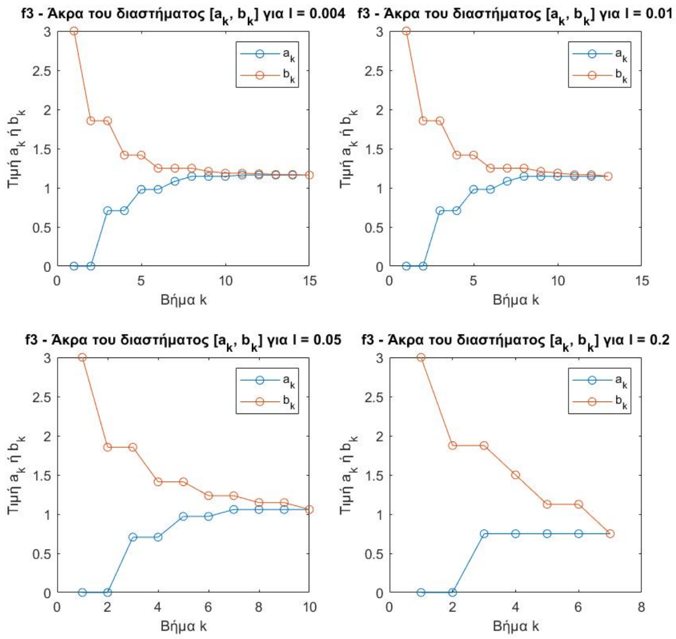


Όπως στις προηγούμενες 2 μεθόδους, όσο μεγαλώνει το l, τόσο λιγότεροι υπολογισμοί της f χρειάζονται, κάτι λογικό αφού μεγαλύτερο l σημαίνει ότι απαιτείται λιγότερη ακρίβεια με αποτέλεσμα να χρειαστούν λιγότεροι υπολογισμοί για να φτάσουμε σε ένα αποδεκτό εύρος.

Όπως και προηγουμένως, βλέπουμε ότι η μορφή της ίδιας της συνάρτησης δεν επηρεάζει το αποτέλεσμα.

Μελετούμε επιπλέον, πως διαμορφώνεται το διάστημα αναζήτησης μετά από κάθε επανάληψη k του αλγορίθμου συναρτήσει και αυτή τη φορά, του l. Δοκιμάζουμε για τιμές l = [0.004, 0.01, 0.05, 0.02]





Τα διαγράμματα $[\alpha\kappa, b\kappa]$ ανά επανάληψη κ , δείχνουν τον περιορισμό του εύρους του κάθε διαστήματος και τη σύγκλιση του αλγόριθμου στην λύση.

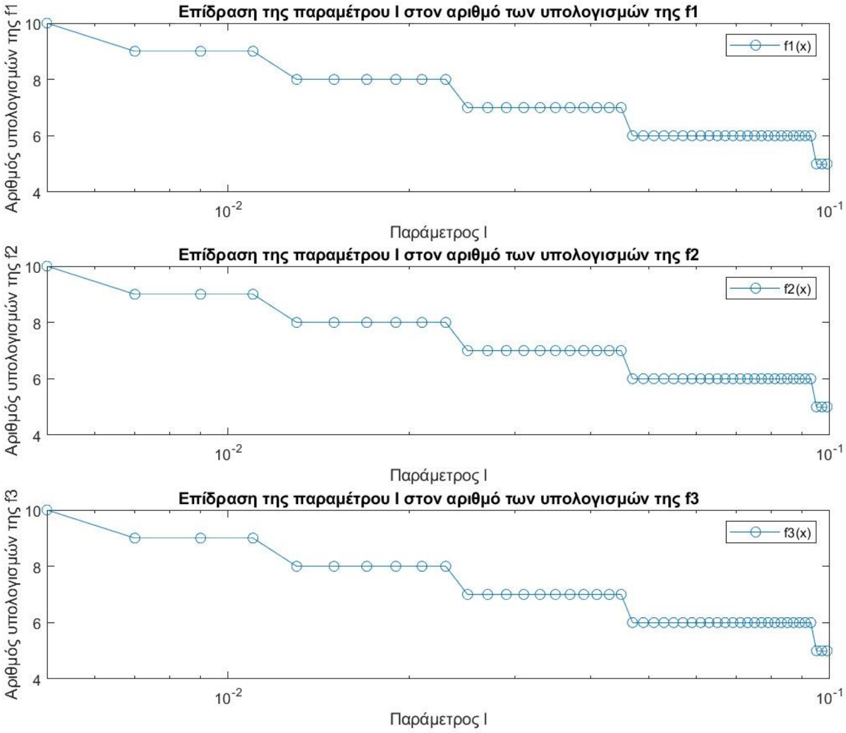
Φαίνεται και εδώ, ότι για μεγαλύτερο l (μικρότερη τελική ακρίβεια), η σύγκλιση γίνεται πιο γρήγορα (σε λιγότερες επαναλήψεις του αλγορίθμου)

Μέθοδος της Διχοτόμου με χρήση παραγώγων

Στη μέθοδο αυτή, κάνουμε χρήση της κλίσης της αντικειμενικής συνάρτησης σε κάποιο σημείο x*. Ως γνωστών, αν $\frac{df(x)}{dt} = 0$**,** στο σημείο x*, τότε αυτό αποτελεί ελάχιστο, Αλλιώς, αν $\frac{df(x)}{dt} > 0$ ή $\frac{df(x)}{dt} < 0$ συνεχίζουμε την αναζήτηση στο [α, x*) ή (x*, β] αντίστοιχα.

Το αρχείο MATLAB για την υλοποίησή της έχει όνομα thema4.m

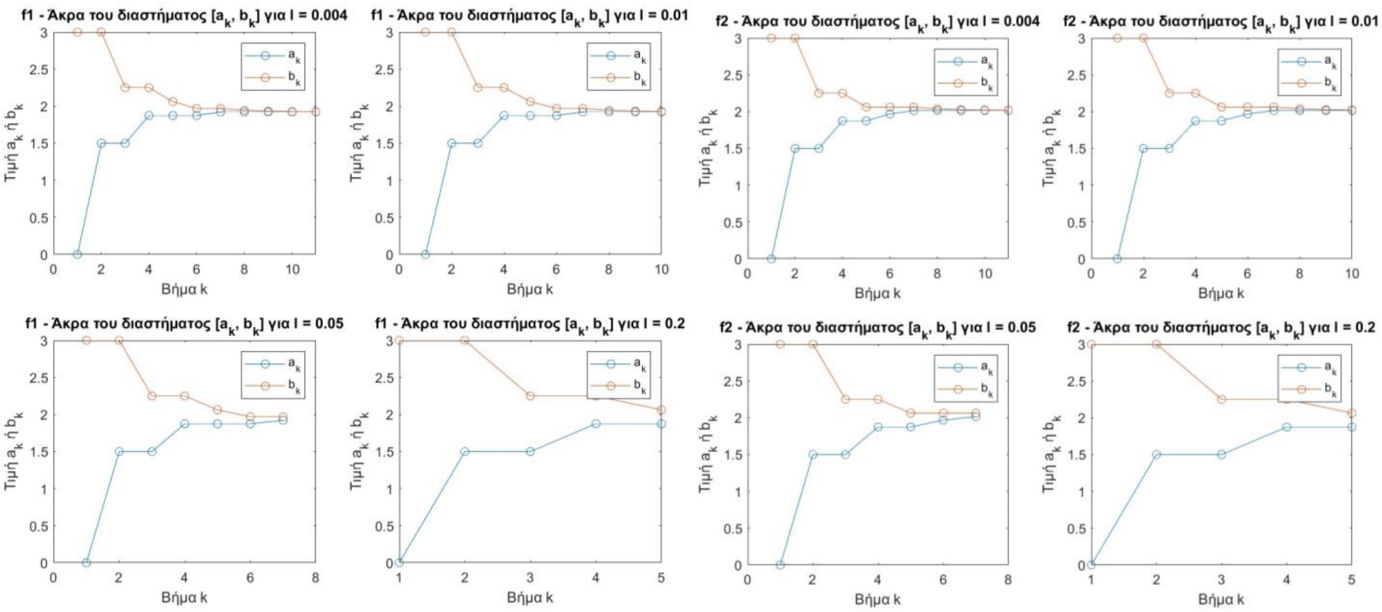
Μελετούμε τη μεταβολή των υπολογισμών της αντικειμενικής συνάρτησης για κάθε μια από τις f1,f2, f3, μεταβάλλοντας το l ώστε να παίρνει τιμές από 0.005 έως 0.1 με βήμα 0.002, έχουμε ως αποτέλεσμα τα εξής γραφήματα:

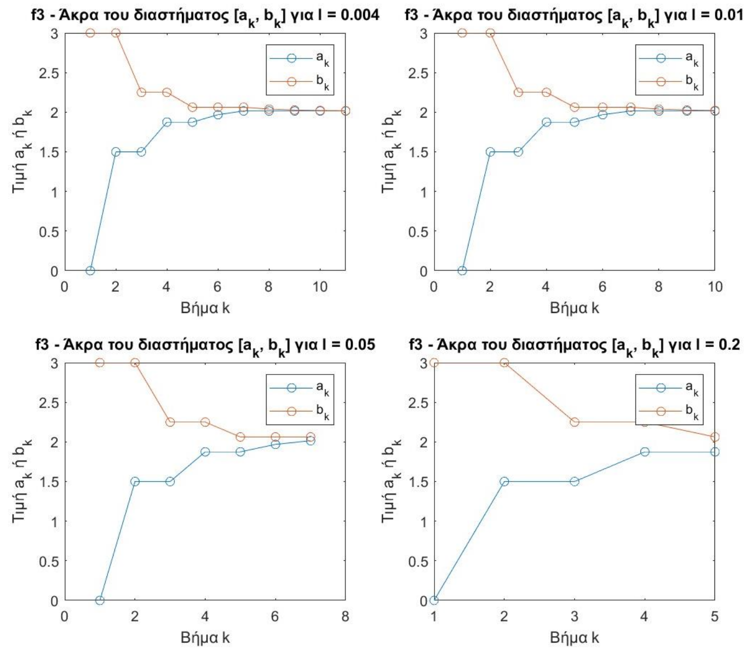


Για άλλη μια φορά, όσο μεγαλώνει το l, τόσο λιγότεροι υπολογισμοί της f χρειάζονται, αφού όπως προαναφέρθηκε, μεγαλύτερο l σημαίνει ότι απαιτείται λιγότερη ακρίβεια με αποτέλεσμα να χρειαστούν λιγότεροι υπολογισμοί για να φτάσουμε σε ένα αποδεκτό εύρος.

Όπως και προηγουμένως, βλέπουμε ότι η μορφή της ίδιας της συνάρτησης δεν επηρεάζει το αποτέλεσμα.

Εξετάζοντας τώρα, πως διαμορφώνεται το διάστημα αναζήτησης μετά από κάθε επανάληψη k του αλγορίθμου συναρτήσει και αυτή τη φορά, του l, δοκιμάζουμε και πάλι για τιμές l = [0.004, 0.01, 0.05, 0.02].





Τα διαγράμματα $[ακ, bκ]$ ανά επανάληψη $κ$, δείχνουν τον περιορισμό του εύρους του κάθε διαστήματος και τη σύγκλιση του αλγόριθμου στην λύση.

Φαίνεται και εδώ, ότι για μεγαλύτερο l (μικρότερη τελική ακρίβεια), η σύγκλιση γίνεται πιο γρήγορα (σε λιγότερες επαναλήψεις του αλγορίθμου)

Σύγκριση των Μεθόδων

Μιας και όπως διαπιστώσαμε η μορφή της ίδιας της αντικειμενικής συνάρτησης δεν επηρεάζει την αποτελεσματικότητα κανενός από τους παραπάνω αλγορίθμους, θα προχωρήσουμε στη σύγκριση μεταξύ αυτών σχολιάζοντας τα δεδομένα που προκύπτουν από την f1 μόνο και ως προς τη μεταβολή της τελικής ακρίβειας l.

- ***Μεταβολή των υπολογισμών της αντικειμενικής συνάρτησης f***

Είδαμε πως όλες οι μέθοδοι που προηγήθηκαν, επηρεάζονται από το l , την τελική ακρίβεια του εύρους αναζήτησης, ως προς τον αριθμό υπολογισμών της f, καθώς και ότι μικρότερο l, δηλαδή μεγαλύτερη ακρίβεια απαιτεί περισσότερο «κόπο» (περισσότερους υπολογισμούς) για τη λήψη αποτελέσματος. Αν συγκρίνουμε όμως, τις μεθόδους ως προς την αποδοτικότητά τους για κάποιο l βλέπουμε τα εξής:

Μέθοδος	**Πλήθος κλήσεων της f (για l = 0.005)**
Της Διχοτόμου	22
Του Χρυσού Τομέα	15
Fibonacci	13
Της Διχοτόμου με χρήση Παραγώγων	10

- ***Μεταβολή του διαστήματος αναζήτησης $[ακ, bκ]$ σε κάθε επανάληψη $κ$***

Η επίδραση της τελικής ακρίβειας l στις μεθόδους που μελετήσαμε, διακρίνεται και στο ρυθμό μείωσης του διαστήματος αναζήτησης $[\alpha\kappa, b\kappa]$. Φαίνεται, πως για μεγαλύτερο l (δηλαδή μικρότερη ακρίβεια), το διάστημα αναζήτησης $[\alpha\kappa, b\kappa]$ συγκλίνει γρηγορότερα, δηλαδή σε λιγότερες επαναλήψεις κ της μεθόδου, στο επιθυμητό διάστημα. Πραγματοποιώντας και πάλι σύγκριση μεταξύ των μεθόδων για δεδομένο l παρατηρούμε τα εξής:

Μέθοδος	**Πλήθος επαναλήψεων κ της μεθόδου (για l = 0.01)**
Της Διχοτόμου	10
Του Χρυσού Τομέα	13
Fibonacci	13
Της Διχοτόμου με χρήση Παραγώγων	10

Συμπέρασμα: Ο πιο αποδοτικός, ως προς και τα δύο σημεία που εξετάσαμε, αλγόριθμος για την εύρεση του ελαχίστου, σε κυρτή συνάρτηση μιας μεταβλητής, είναι ο αλγόριθμος της διχοτόμησης με χρήση παραγώγου.

Όσον αφορά τον αριθμό υπολογισμών της f ακολουθεί η μέθοδος Fibonacci και έπειτα οι Χρυσού Τομέα και Διχοτόμου, ενώ η απλή μέθοδος της Διχοτόμου παρουσιάζει καλύτερη επίδοση από τις άλλες δύο σε ότι αφορά τον αριθμό των επαναλήψεων.