2022-2023

ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ ΑΠΘ - Τεχνικές Βελτιστοποίησης

ΦΙΛΙΠΠΟΣ ΓΕΡΜΑΝΟΠΟΥΛΟΣ

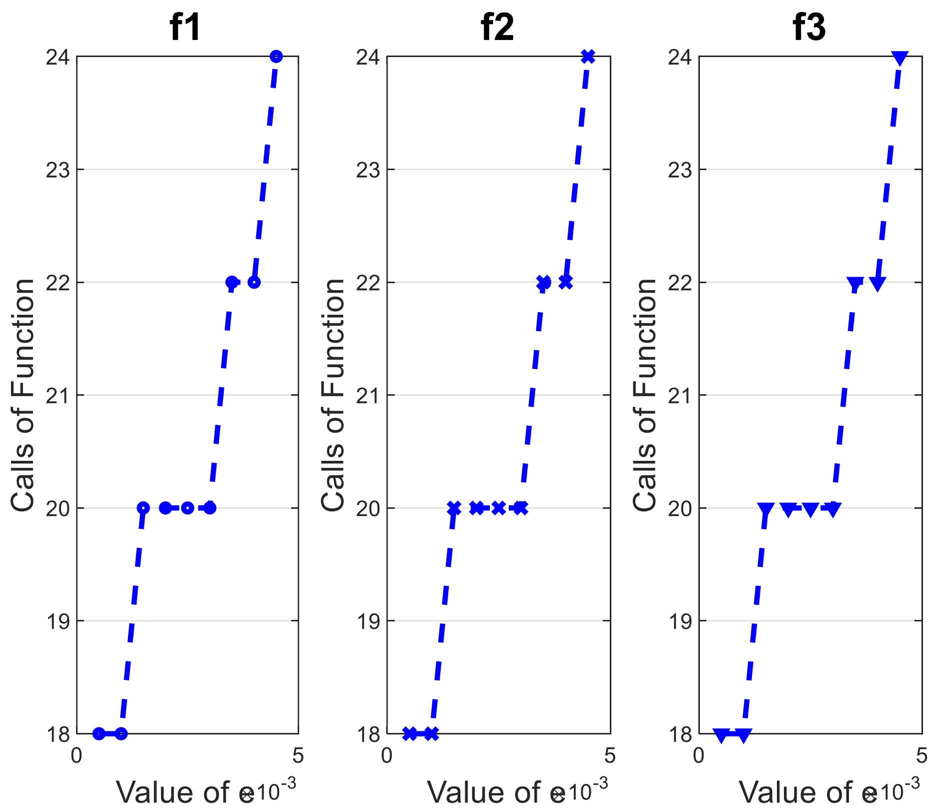
[1Η ΕΡΓΑΣΤΗΡΙΑΚΗ ΑΣΚΗΣΗ]

Ελαχιστοποίηση κυρτής συνάρτησης μίας μεταβλητής σε δοσμένο διάστημα.

ΘΕΜΑ 1

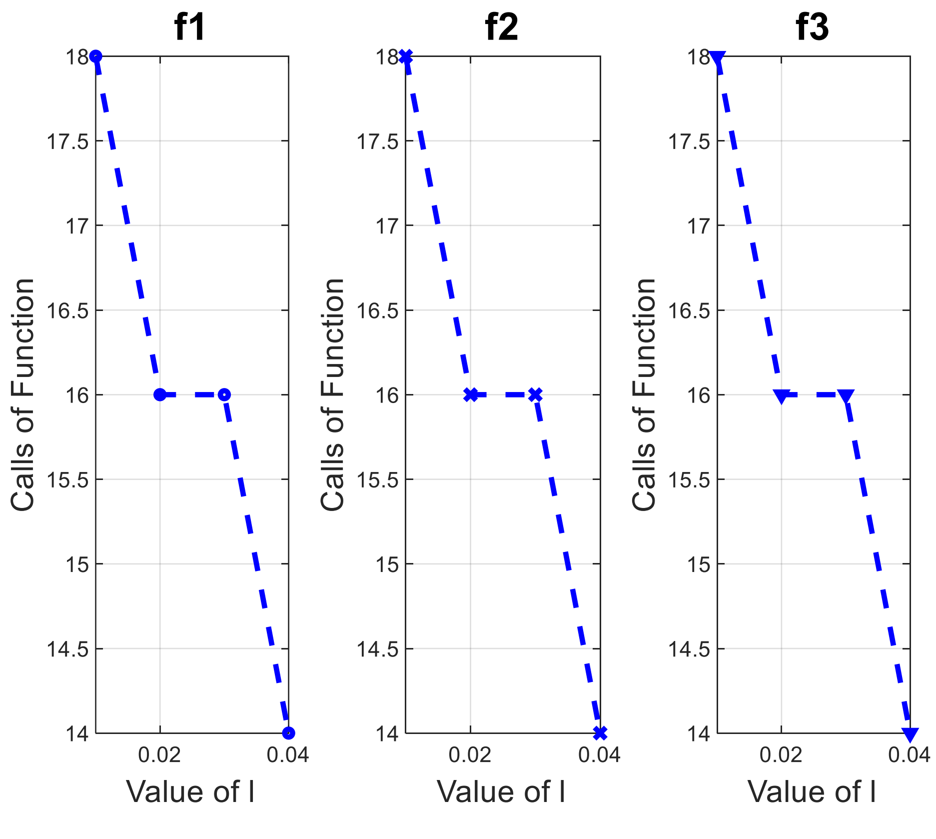
Οι τρείς δοθείσες συναρτήσεις είναι κυρτές στο πεδίο ορισμού που δίνεται στην εκφώνηση. Έτσι καθίσταται δυνατό να χρησιμοποιήσουμε όλες τις μεθόδους αναζήτησης ελαχίστου χωρίς τη χρήση παραγώγων, αλλά και αυτή με χρήση παραγώγων.

Το πρώτο θέμα πραγματεύεται την μέθοδο της διχοτόμου. Η υλοποίησή της στο MATLAB, χωρίζεται σε δύο φακέλους. Ο “l_constant” φάκελος αφορά την περίπτωση όπου το τελικό εύρος αναζήτησης l είναι σταθερό και ίσο με 0.01. Περιέχει ένα αρχείο script που συμπεριλαμβάνει όλες τις αντικειμενικές συναρτήσεις και ένα βοηθητικό αρχείο function όπου υλοποιεί τη διχοτόμηση. Μελετάται η μεταβολή υπολογισμών των αντικειμενικών συναρτήσεων, καθώς μεταβάλλεται ελαφρώς η απόσταση e από τη διχοτόμο. Παρακάτω η αντίστοιχη γραφική παράσταση:



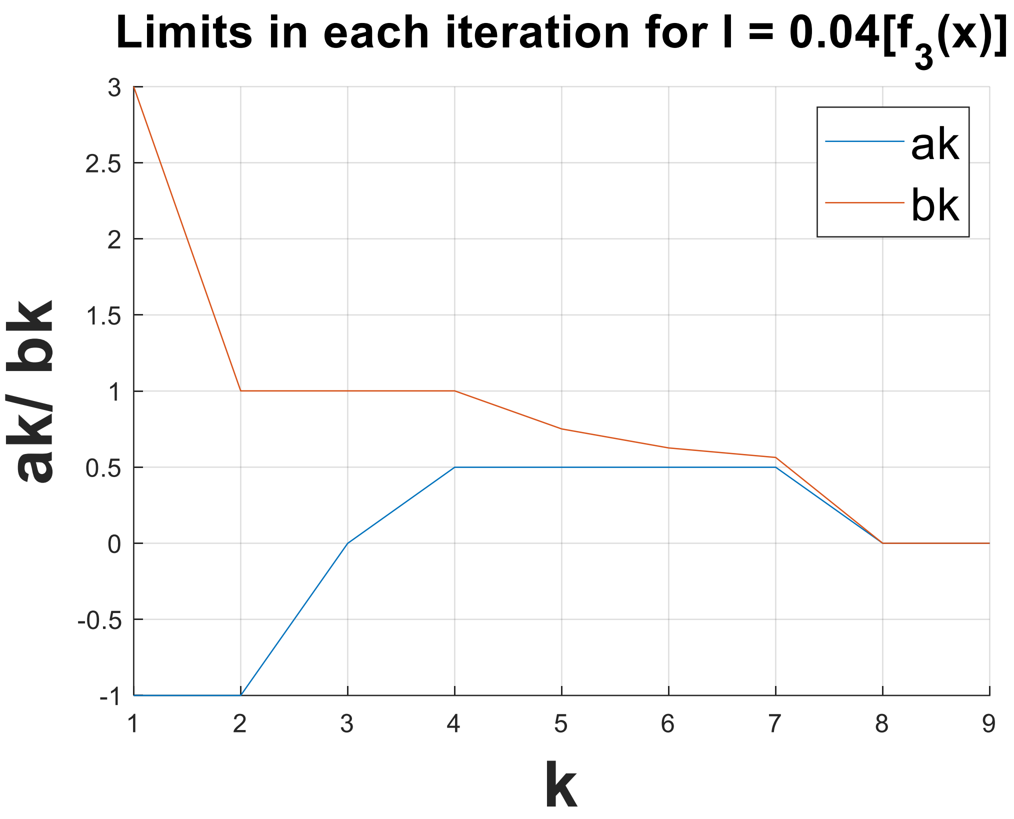
Παρατηρούμε ότι οι 3 αντικειμενικές συναρτήσεις καλούνται ακριβώς τις ίδιες φορές, για ίδια e. Επίσης, αυξανόμενο το e, αυξάνονται και οι φορές που θα κληθούν οι συναρτήσεις, όπως είναι προφανώς αναμενόμενο. Αρκεί αυτή η μεταβολή όμως να είναι επαρκώς μεγάλη, καθώς σε μερικές επαναλήψεις βλέπουμε πως οι κλήσεις τις συνάρτησης παραμένουν ίδιες, παρόλο που αυξήθηκε το e.

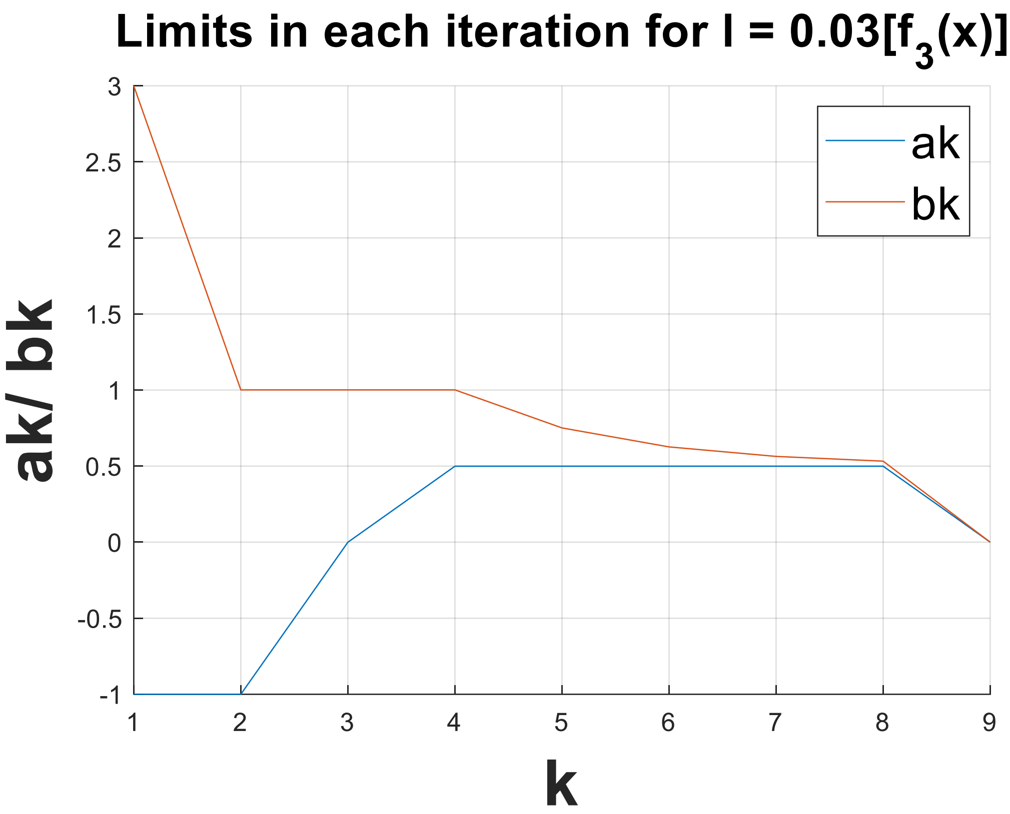
Ο δεύτερος φάκελος “e_constant” περιέχει την περίπτωση όπου το e παραμένει σταθερό και το l μεταβάλλεται. Διαθέτει 3 αρχεία, 1 τύπου function, όπως και στην προηγούμενη περίπτωση και 2 script. Το script “f_e_constant” μελετά τη μεταβολή των υπολογισμών των συναρτήσεων σε σχέση με το l.

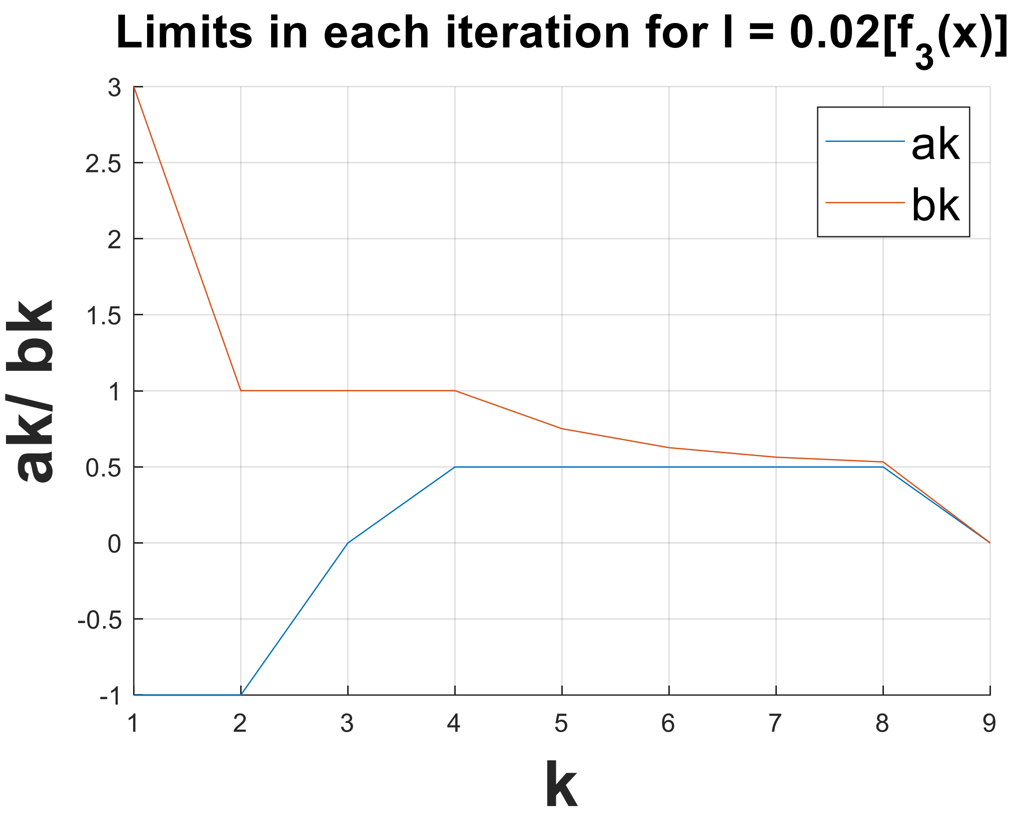


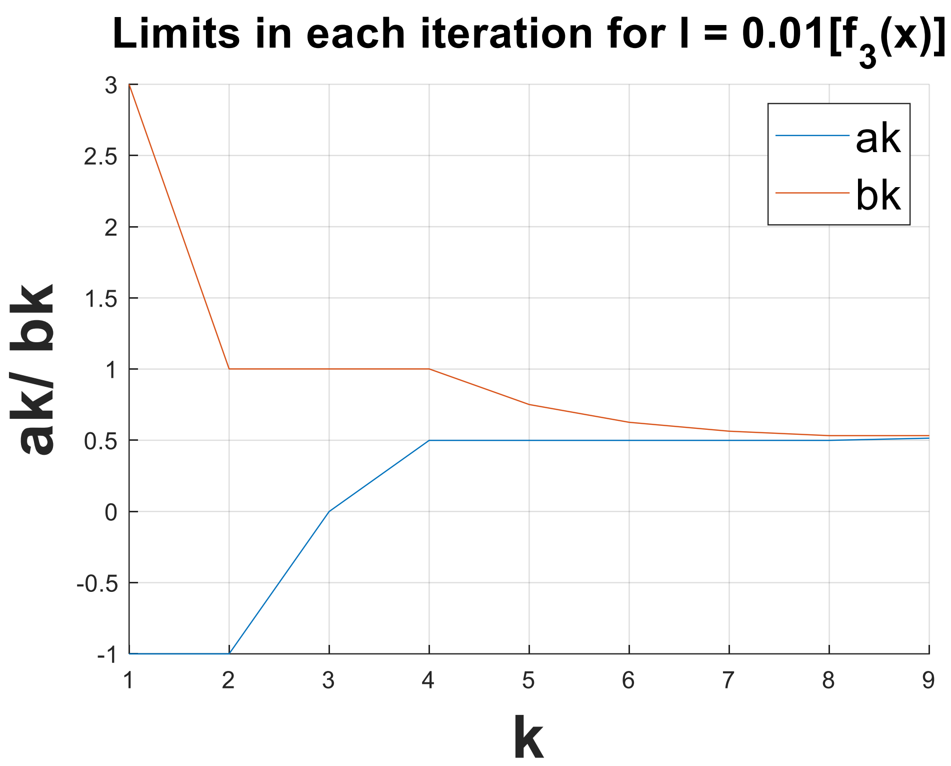
Παρατηρούμε ότι οι 3 αντικειμενικές συναρτήσεις καλούνται ακριβώς τις ίδιες φορές, για ίδια l. Όταν αυξάνεται το l, οι υπολογισμοί των συναρτήσεων τείνουν να μειώνονται, που είναι λογικό εφόσον ικανοποιείται η συνθήκη bk-ak>l για λιγότερες τιμές και έτσι ο αλγόριθμος τερματίζει. Η μεταβολή του l πρέπει επίσης να είναι αρκούντως μεγάλη.

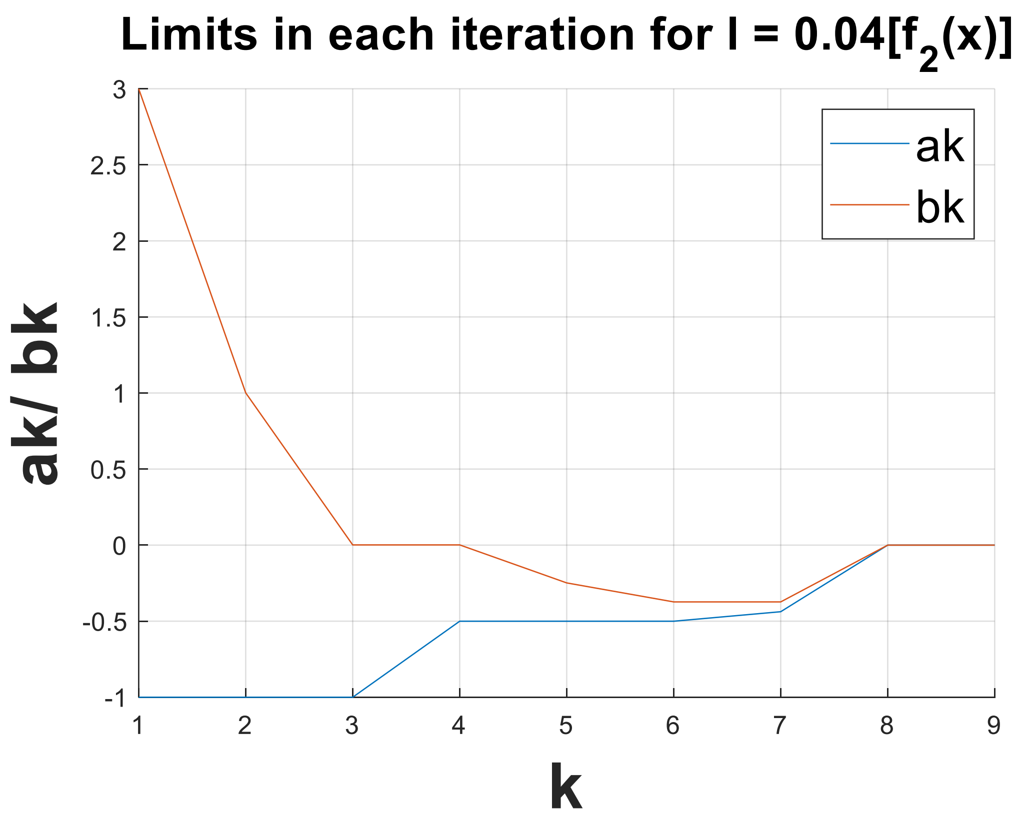
Στο δεύτερο αρχείο script “ak_bk” περιλαμβάνονται οι γραφικές παραστάσεις των άκρων ak, bk των αντικειμενικών συναρτήσεων συναρτήσει του δείκτη βήματος κ, καθώς και ένα παράθυρο που δείχνει τις αρχικές γραφικές παραστάσεις των συναρτήσεων στο πεδίο ορισμού που δίνεται.

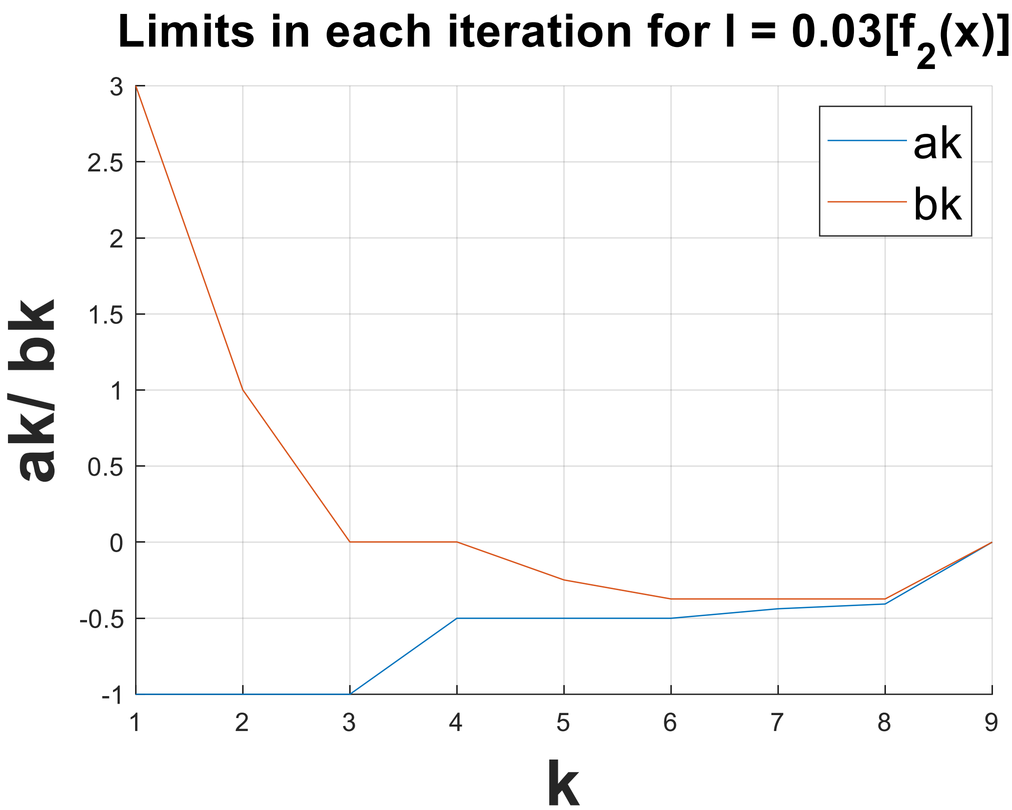



Limits in each iteration for l = 0.03[f3(x)]
ak
bk
ak/ bk
k
3
2.5
2
1.5
1
0.5
0
-0.5
-1
1
2
3
4
5
6
7
8
9

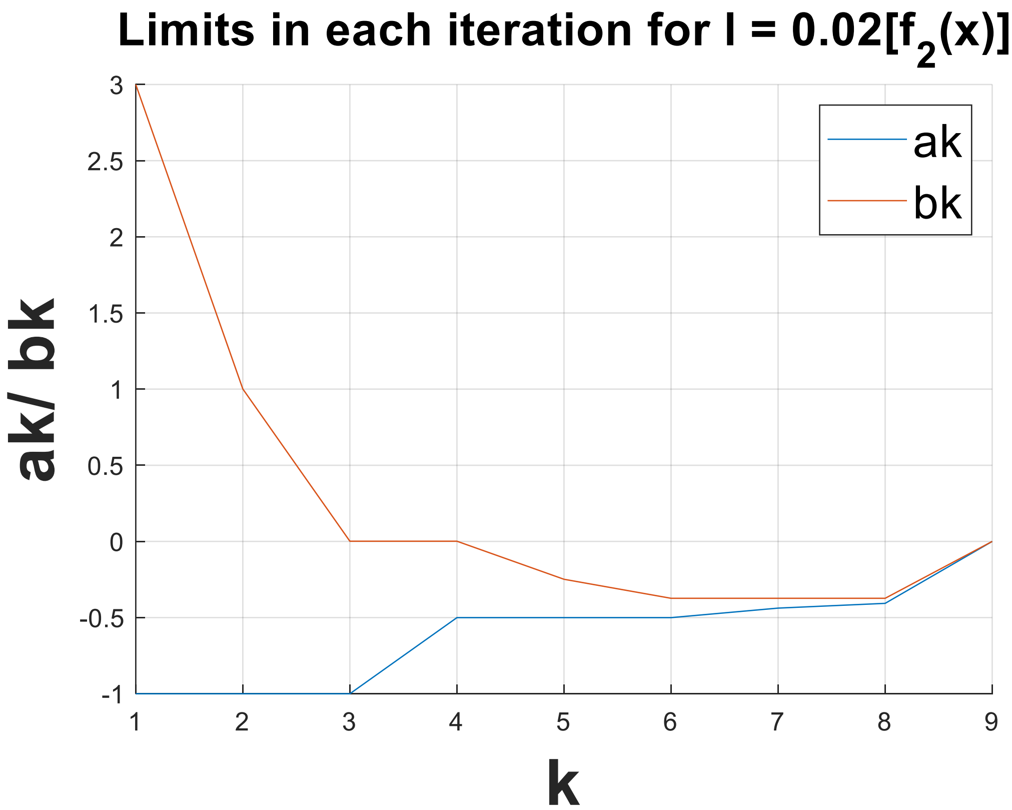

Limits in each iteration for l = 0.02[f3(x)]
ak
bk
ak/ bk
k
3
2.5
2
1.5
1
0.5
0
-0.5
-1
1
2
3
4
5
6
7
8
9


Limits in each iteration for l = 0.01[f3(x)]
ak
bk
ak/ bk
k

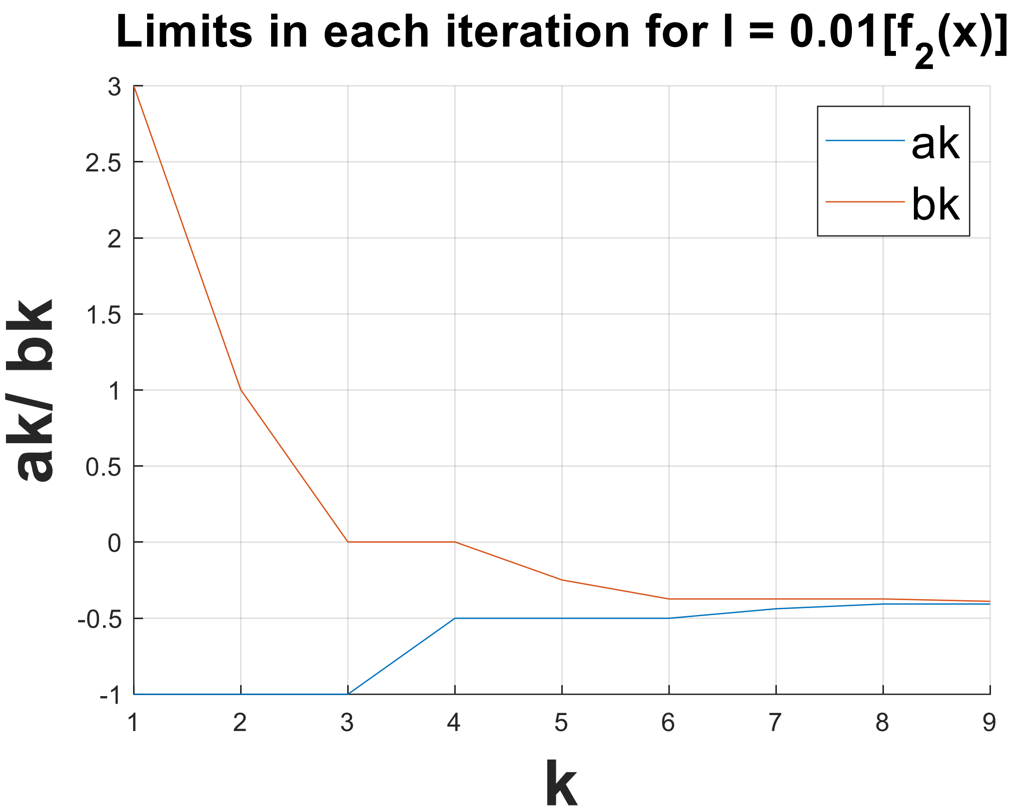

Limits in each iteration for l = 0.04[f2(x)]
ak
bk
ak/ bk
k



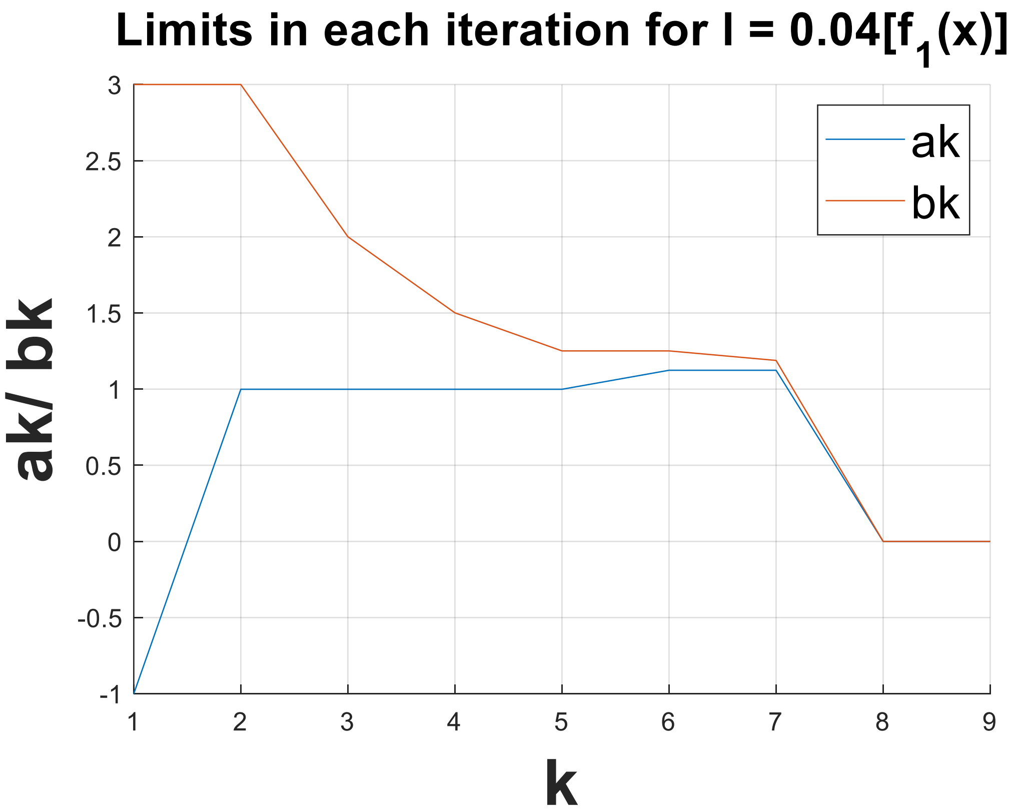
Limits in each iteration for l = 0.03[f2(x)]
ak/ bk
k
ak
bk
3
2.5
2
1.5
1
0.5
0
-0.5
-1
1
2
3
4
5
6
7
8
9



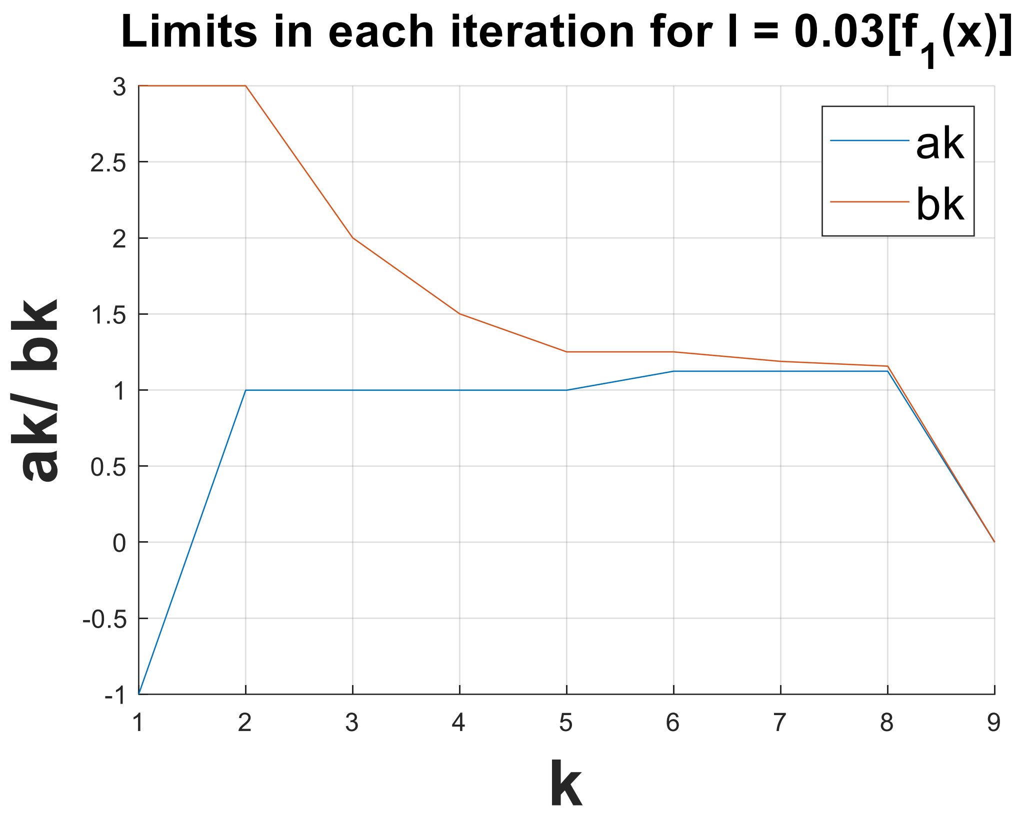
Limits in each iteration for l = 0.02[f2(x)]
ak/ bk
k
ak
bk
3
2.5
2
1.5
1
0.5
0
-0.5
-1
1
2
3
4
5
6
7
8
9

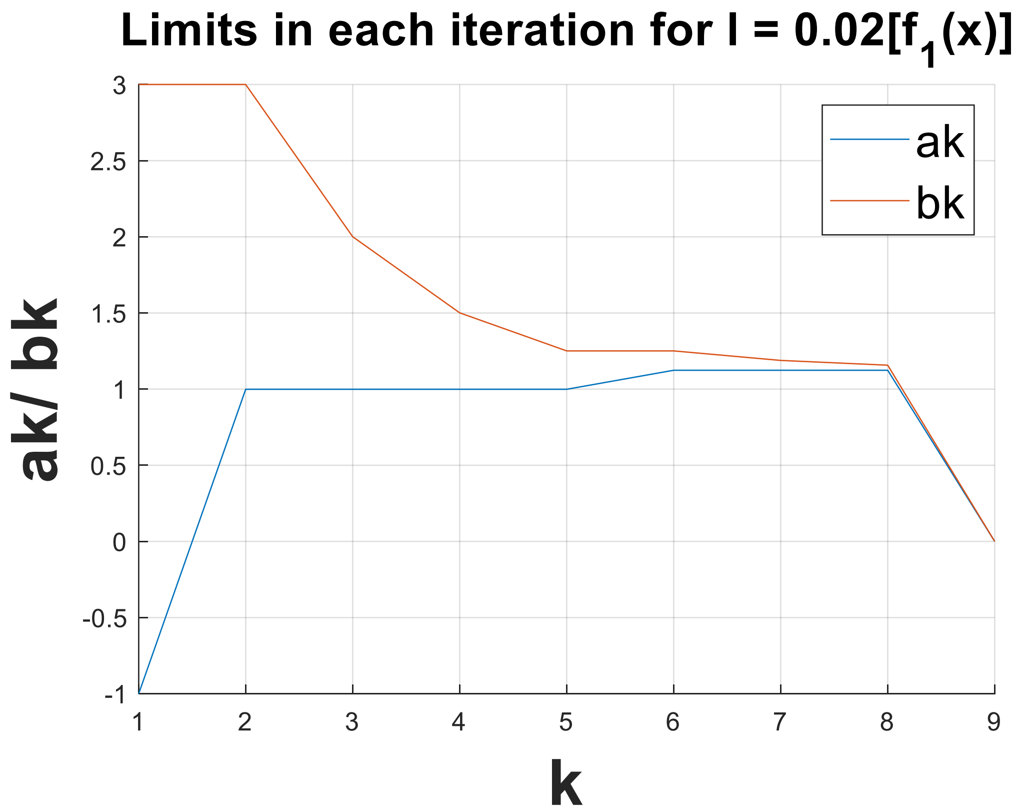


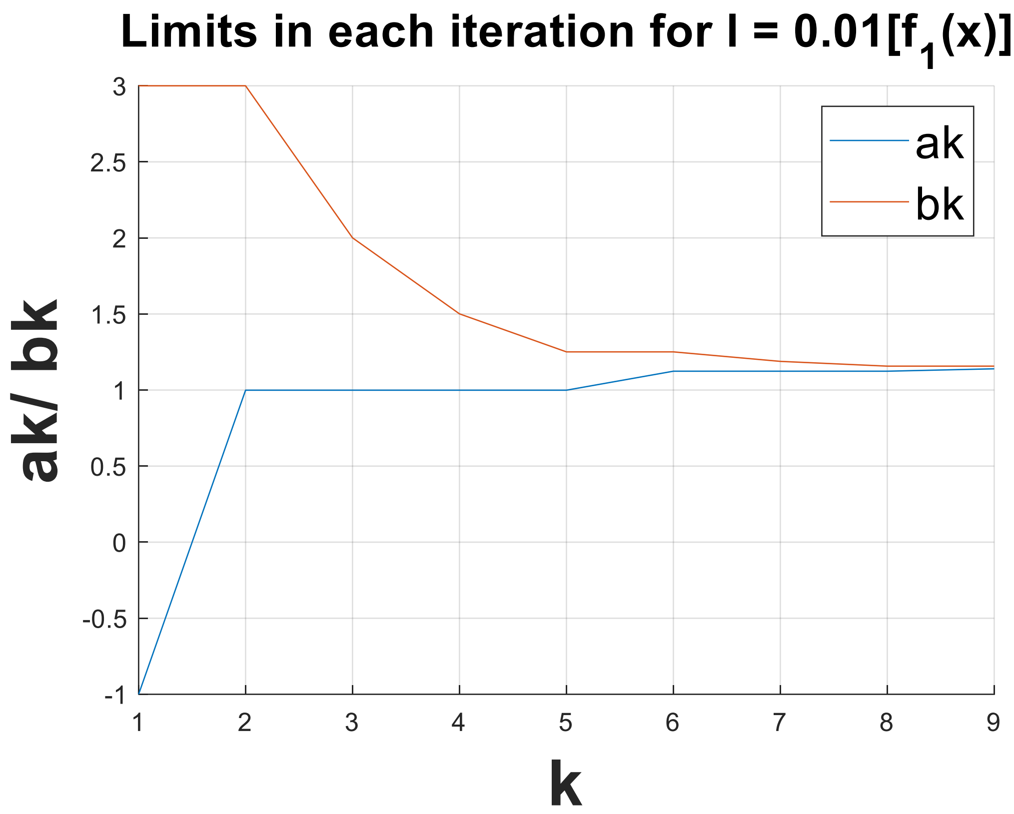
Limits in each iteration for l = 0.01[f2(x)]
ak/ bk
k
ak
bk
3
2.5
2
1.5
1
0.5
0
-0.5
-1
1
2
3
4
5
6
7
8
9

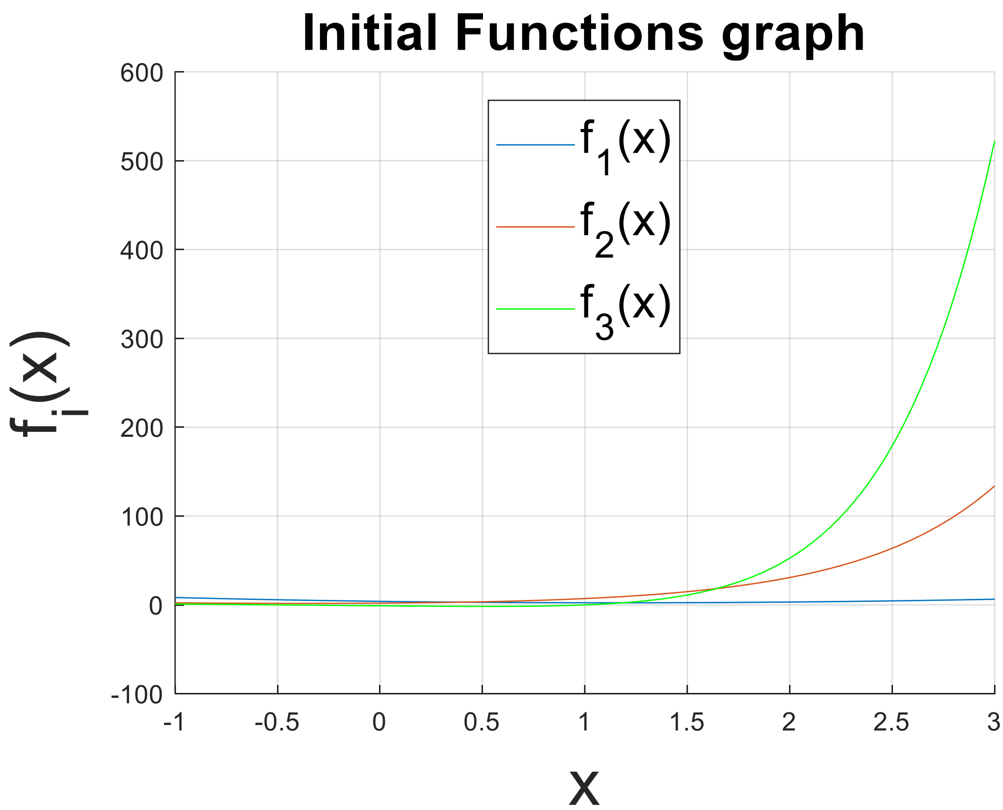


Limits in each iteration for l = 0.04[f1(x)]
ak/ bk
k
ak
bk
3
2.5
2
1.5
1
0.5
0
-0.5
-1
1
2
3
4
5
6
7
8
9


Limits in each iteration for l = 0.03[f1(x)]
ak
bk
ak/ bk
k
3
2.5
2
1.5
1
0.5
0
-0.5
-1
1
2
3
4
5
6
7
8
9


Limits in each iteration for l = 0.02[f1(x)]
ak
bk
ak/ bk
k
3
2.5
2
1.5
1
0.5
0
-0.5
-1
1
2
3
4
5
6
7
8
9

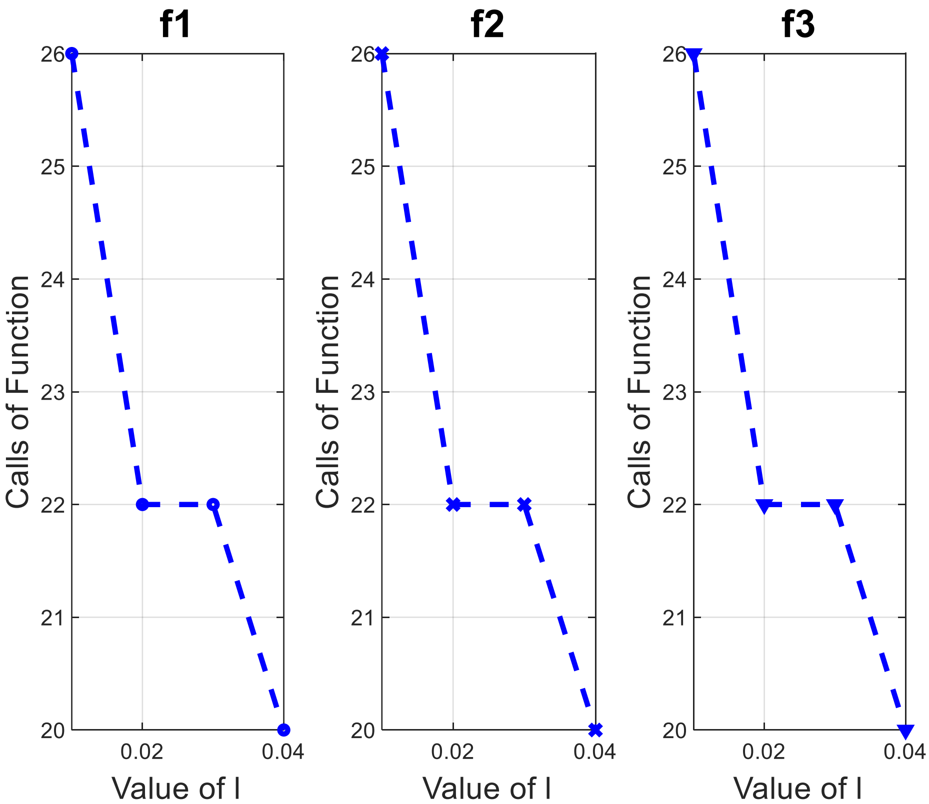




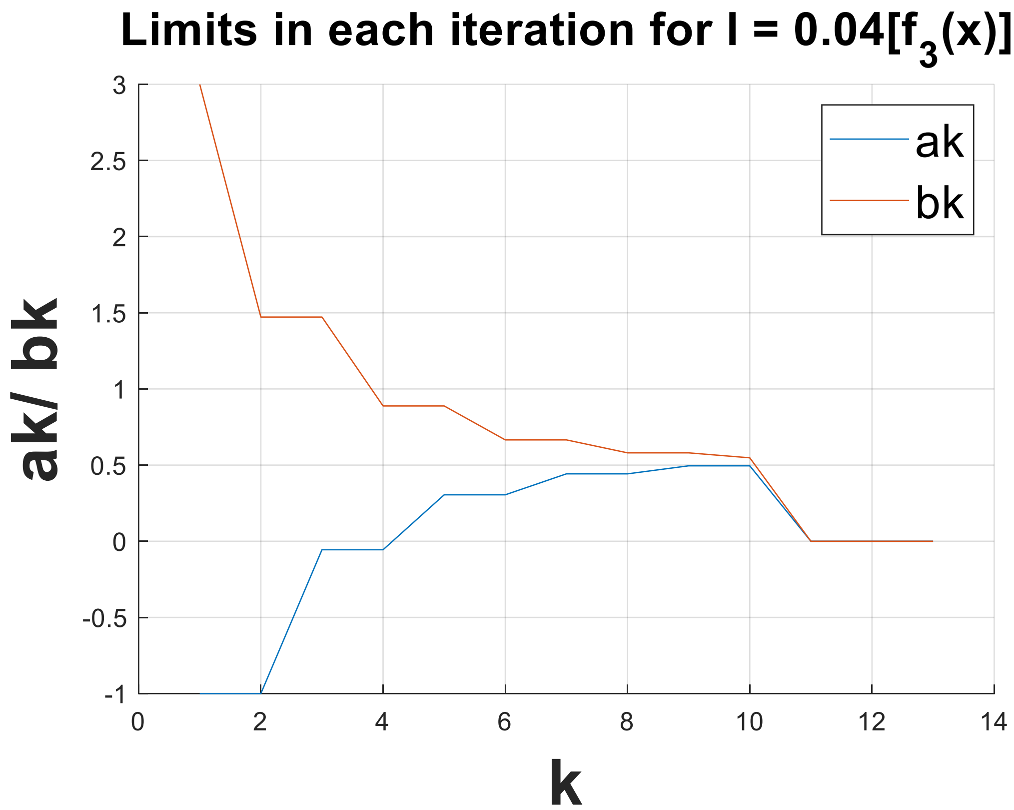
Βλέπουμε πως τα άκρα του διαστήματος για μία συνάρτηση, λαμβάνουν τις ίδιες ακριβώς τιμές , για διαφορετικά l. Ωστόσο, για μικρότερα l επιτρέπονται περισσότερες επαναλήψεις και έτσι το τελικό διάστημα [ak bk] θα πλησιάζει με μεγαλύτερη ακρίβεια την τιμή του ελαχίστου της συνάρτησης. Τούτο ισχύει για l=0.01 που είναι η μικρότερη τιμή του l. Για μεγαλύτερα l, όταν οι ο αριθμός των επαναλήψεων στο διάγραμμα ξεπερνά αυτές για τις οποίες μπορεί να τρέξει ο αλγόριθμος, τότε άμεσα μηδενίζονται οι δύο μεταβλητές ak και bk, ως ένδειξη τερματισμού του αλγορίθμου. Αυτό συμβαίνει λόγω της αρχικοποίησης των αντίστοιχων μεταβλητών που γίνονται plot, ως μηδενικοί πίνακες.

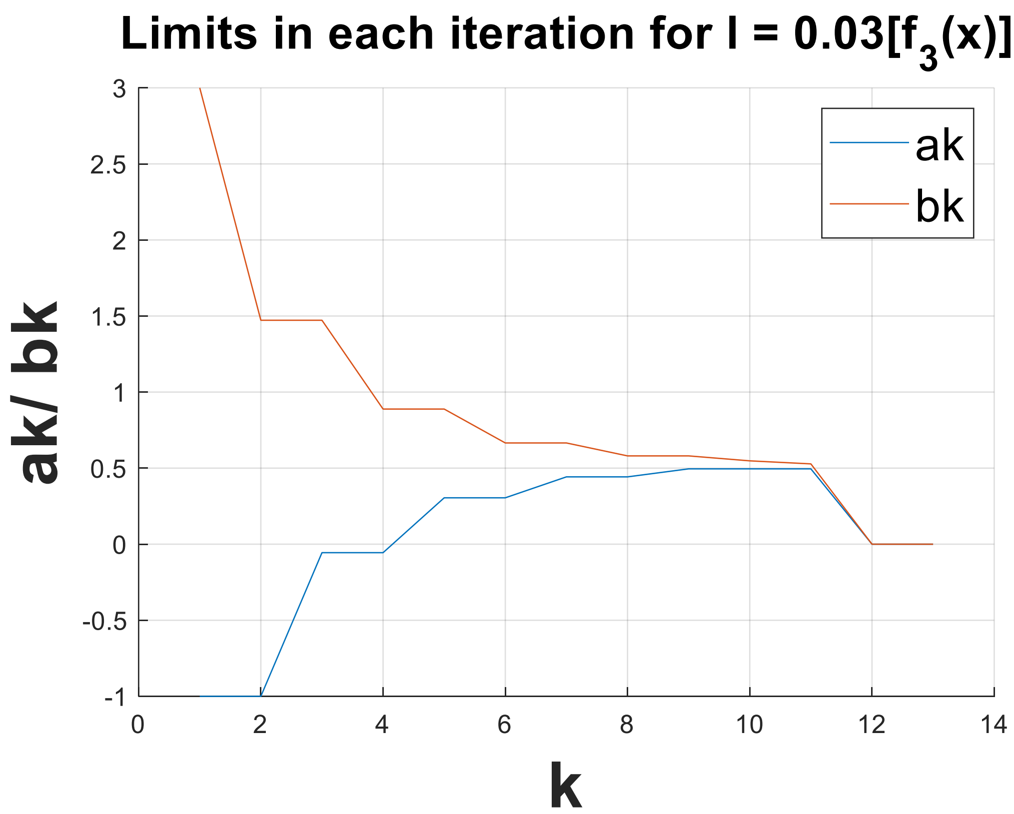
ΘΕΜΑ 2

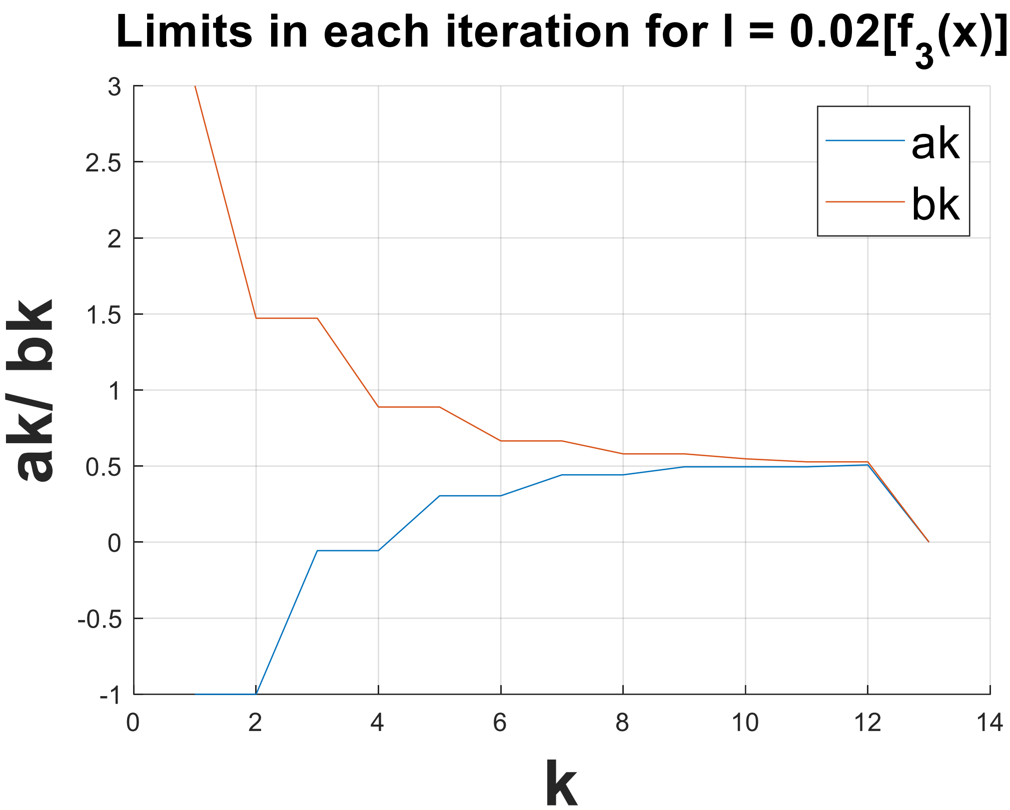
Ακολουθεί η μέθοδος του χρυσού τομέα. Όπως αποδεικνύεται και στο βιβλίο, επιλέγεται η τιμή γ=0.618 (όπου γ=gr στον κώδικα). Η μέθοδος αυτή χωρίζεται σε 2 αρχεία script. Το “f_e_constant” αφορά τη μεταβολή υπολογισμού των αντικειμενικών συναρτήσεων σε συνάρτηση με το μεταβαλλόμενο l.

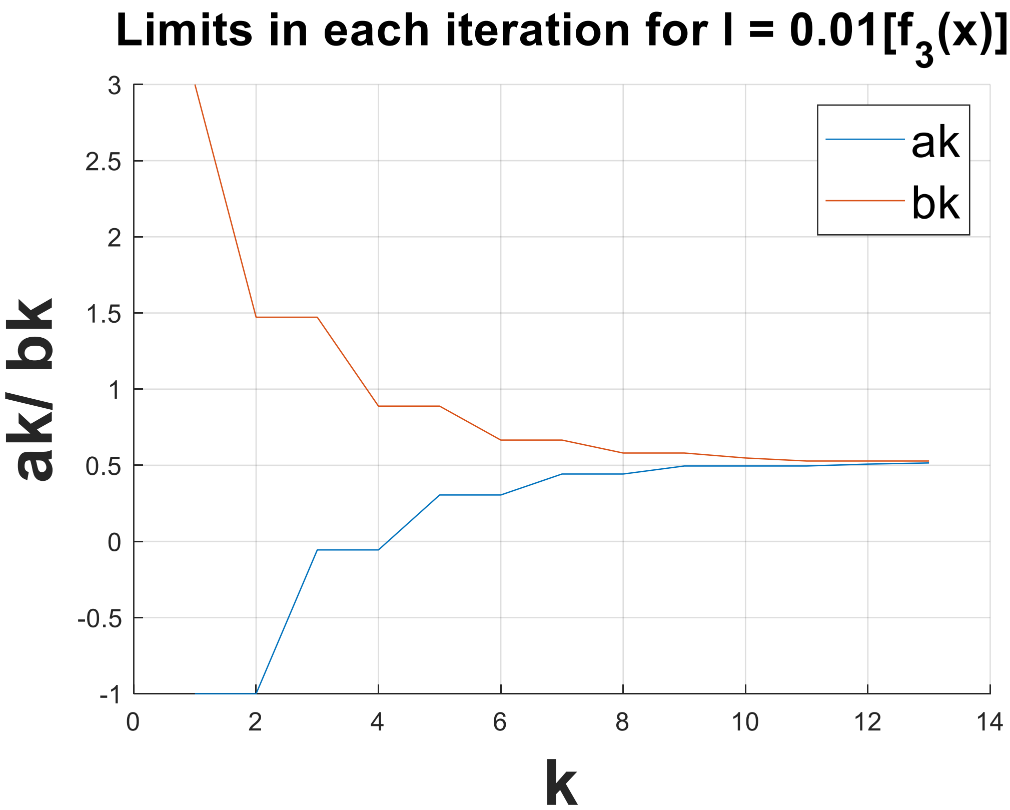


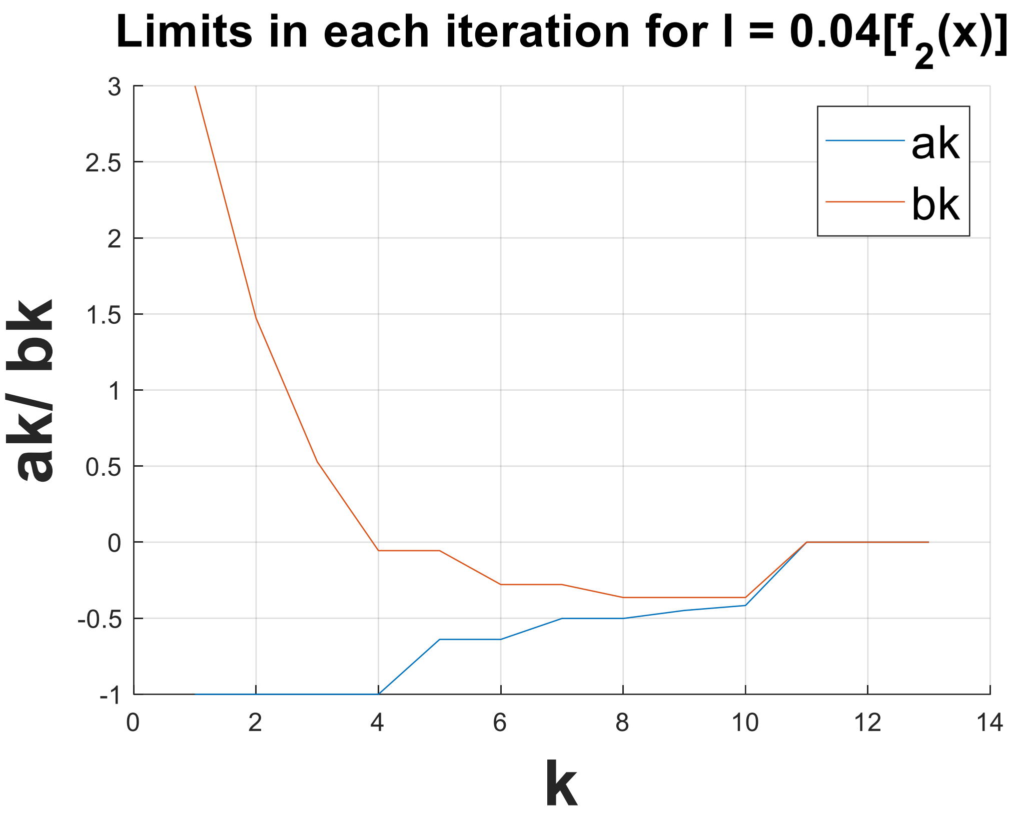
Το δεύτερο αρχείο script “ak_bk” σχεδιάζει τις γραφικές παραστάσεις των άκρων των συναρτήσεων σε κάθε επανάληψη και σχεδιάζει τις αρχικές συναρτήσεις στο δοθέν πεδίο ορισμού.

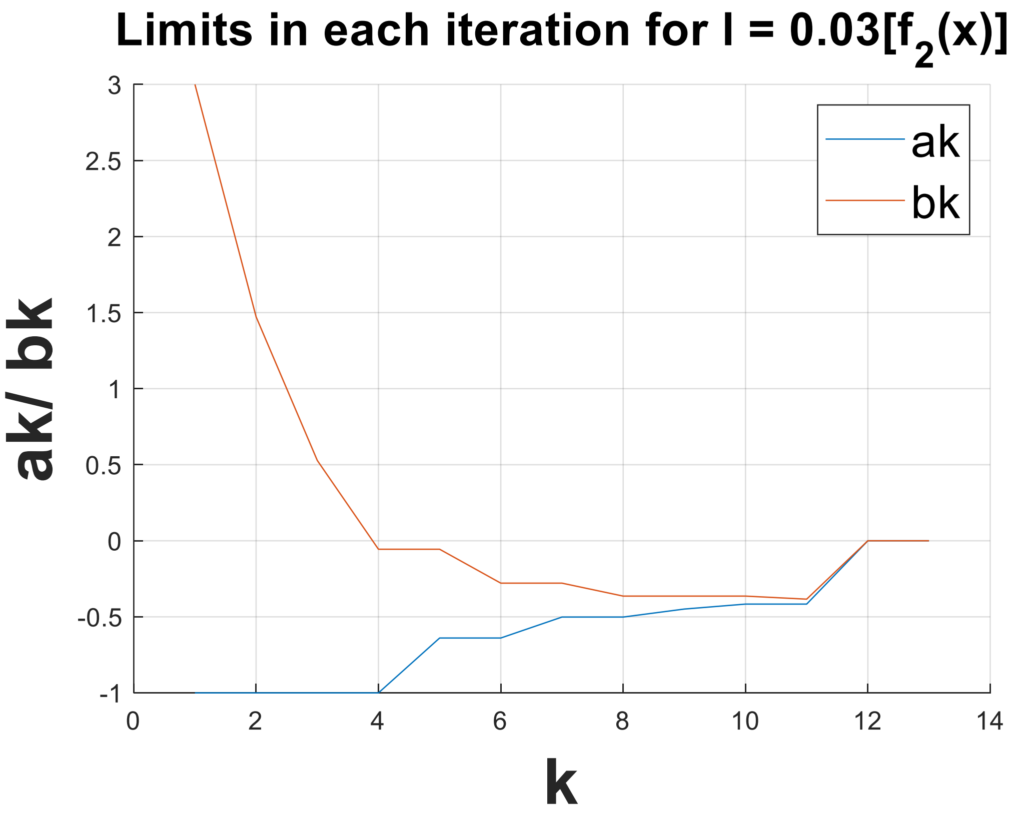


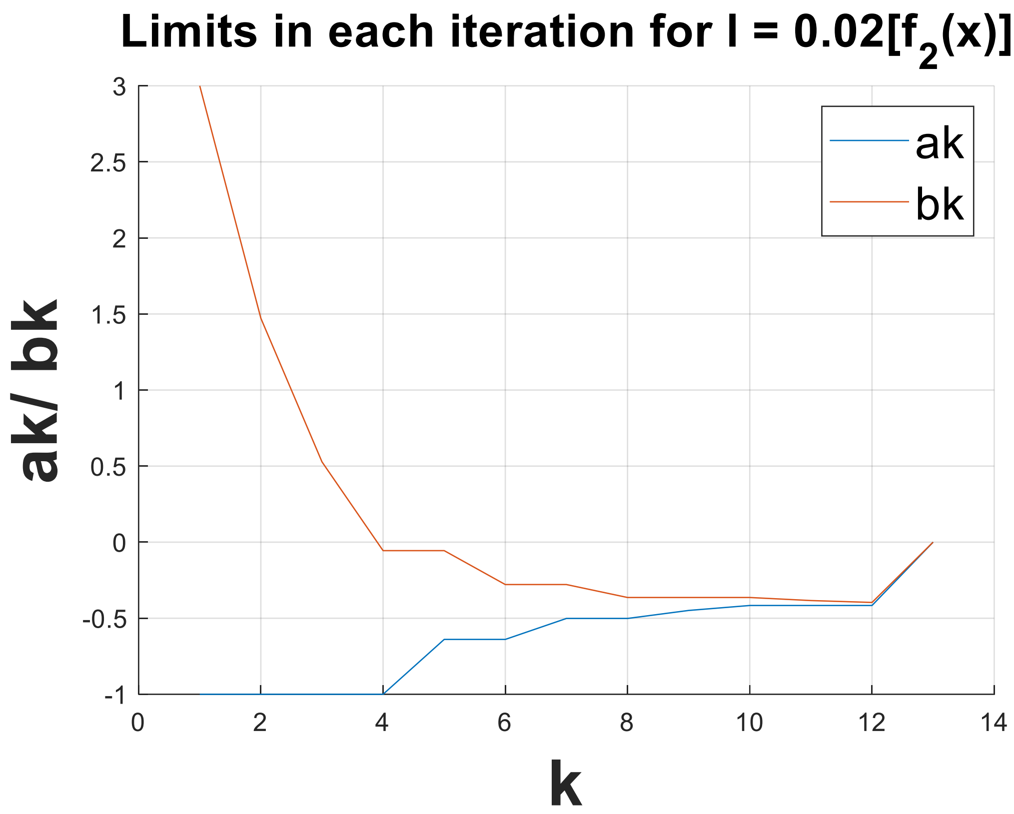


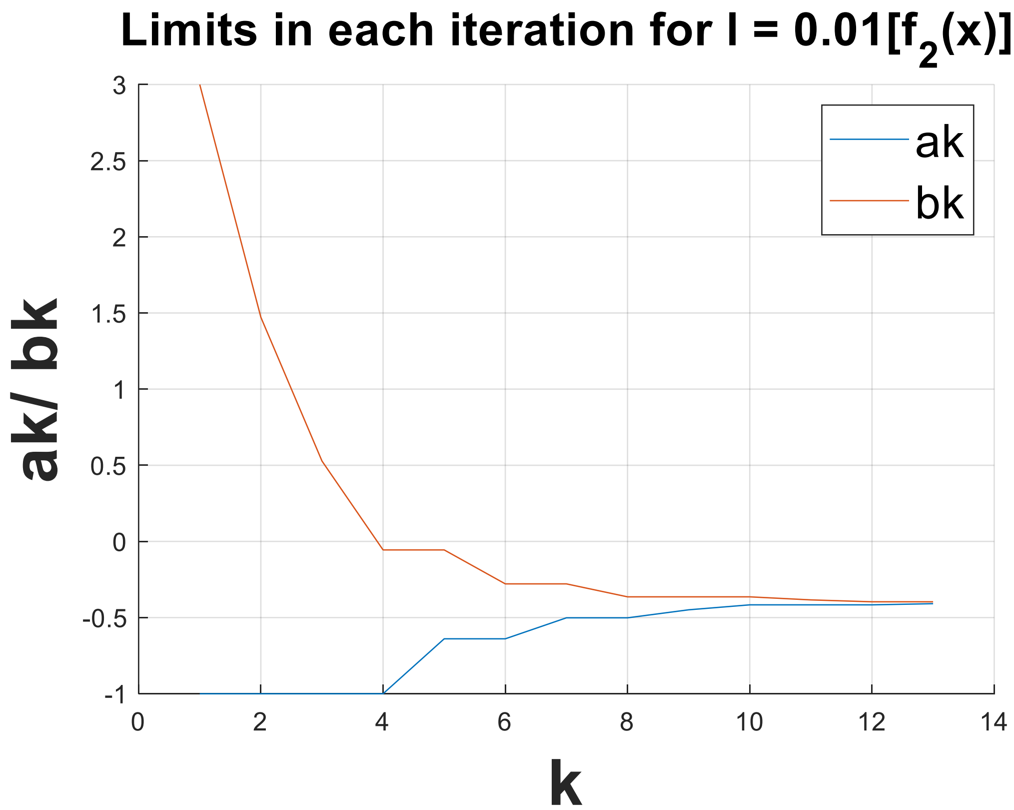

Limits in each iteration for l = 0.02[f3(x)]
ak
bk
ak/ bk
k
3
2.5
2
1.5
1
0.5
0
-0.5
-1
0
2
4
6
8
10
12
14

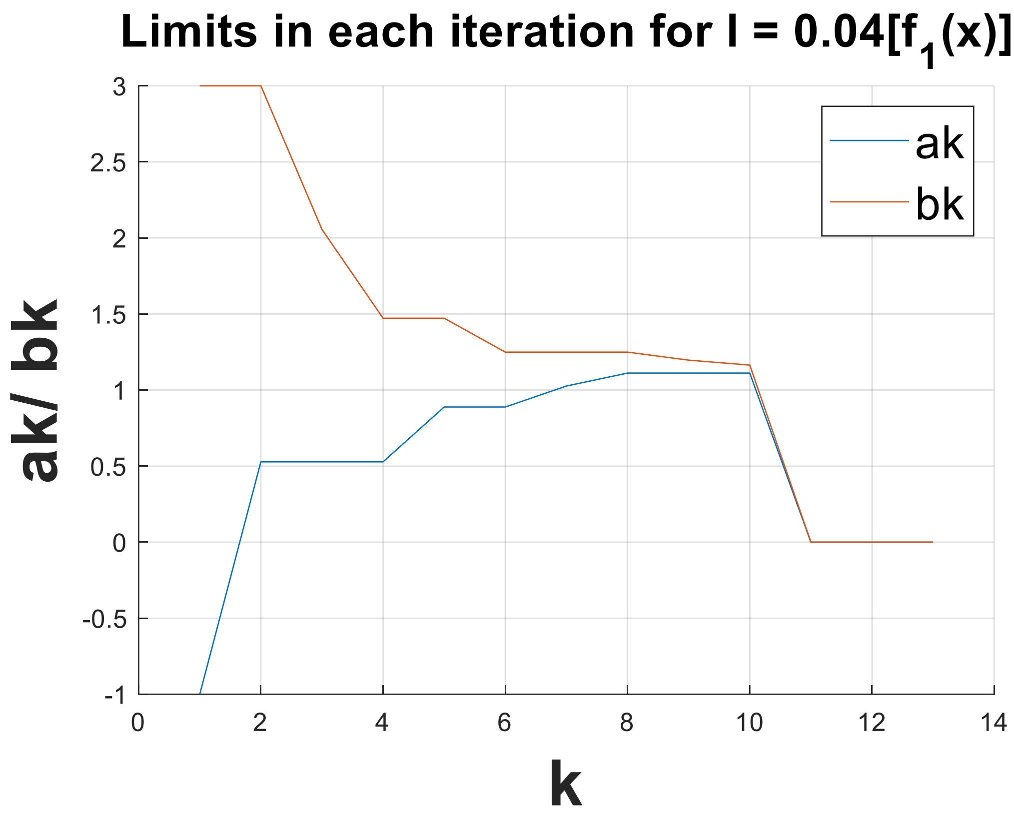

Limits in each iteration for l = 0.01[f3(x)]
ak
bk
ak/ bk
k
3
2.5
2
1.5
1
0.5
0
-0.5
-1
0
2
4
6
8
10
12
14

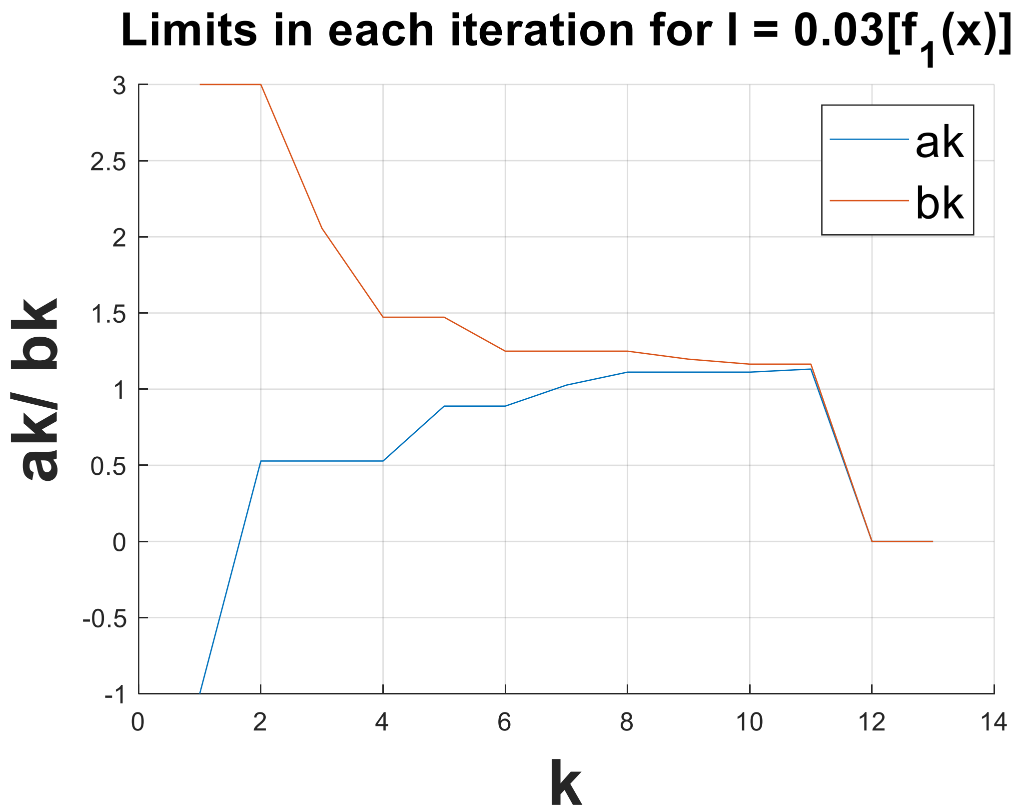

Limits in each iteration for l = 0.04[f2(x)]
ak/ bk
k
ak
bk

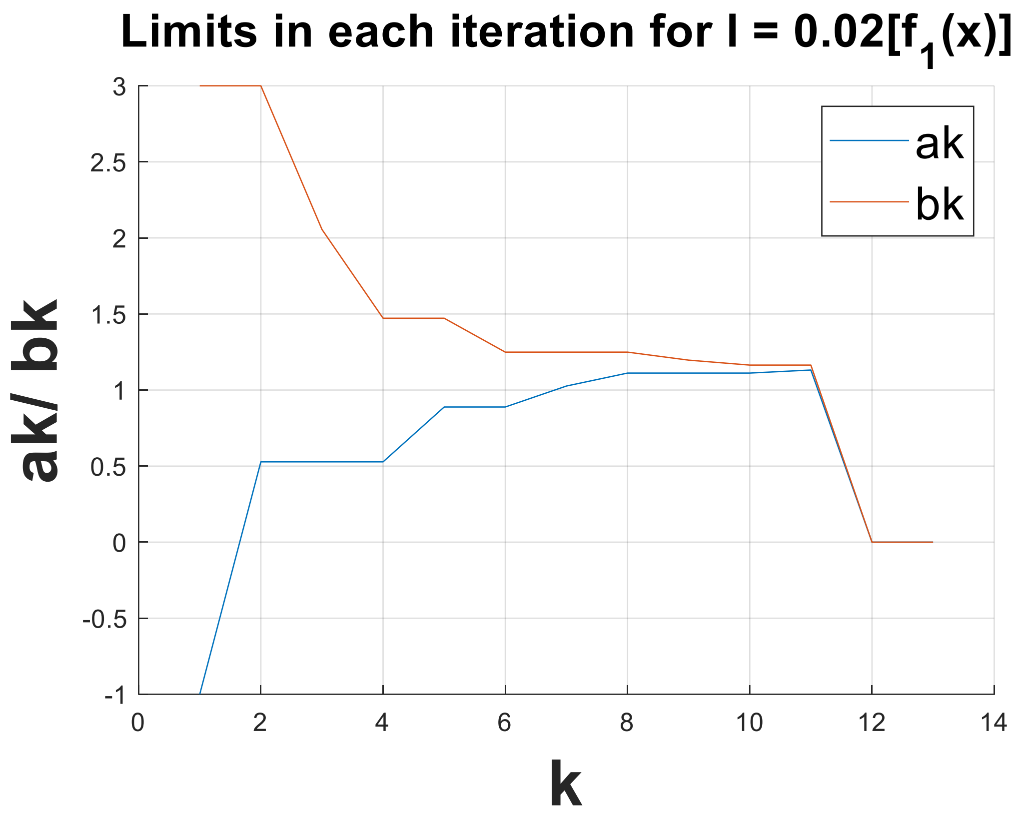

Limits in each iteration for l = 0.03[f2(x)]
ak/ bk
k
ak
bk

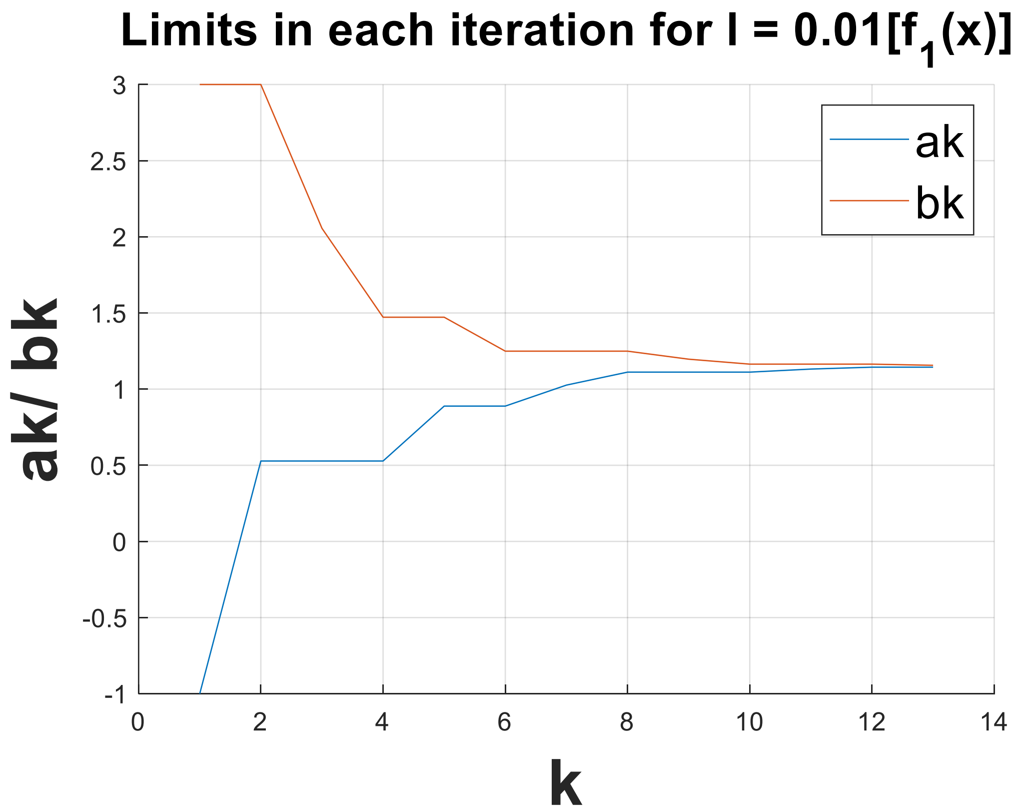

Limits in each iteration for l = 0.02[f2(x)]
ak/ bk
k
ak
bk
3
2.5
2
1.5
1
0.5
0
-0.5
-1
0
2
4
6
8
10
12
14

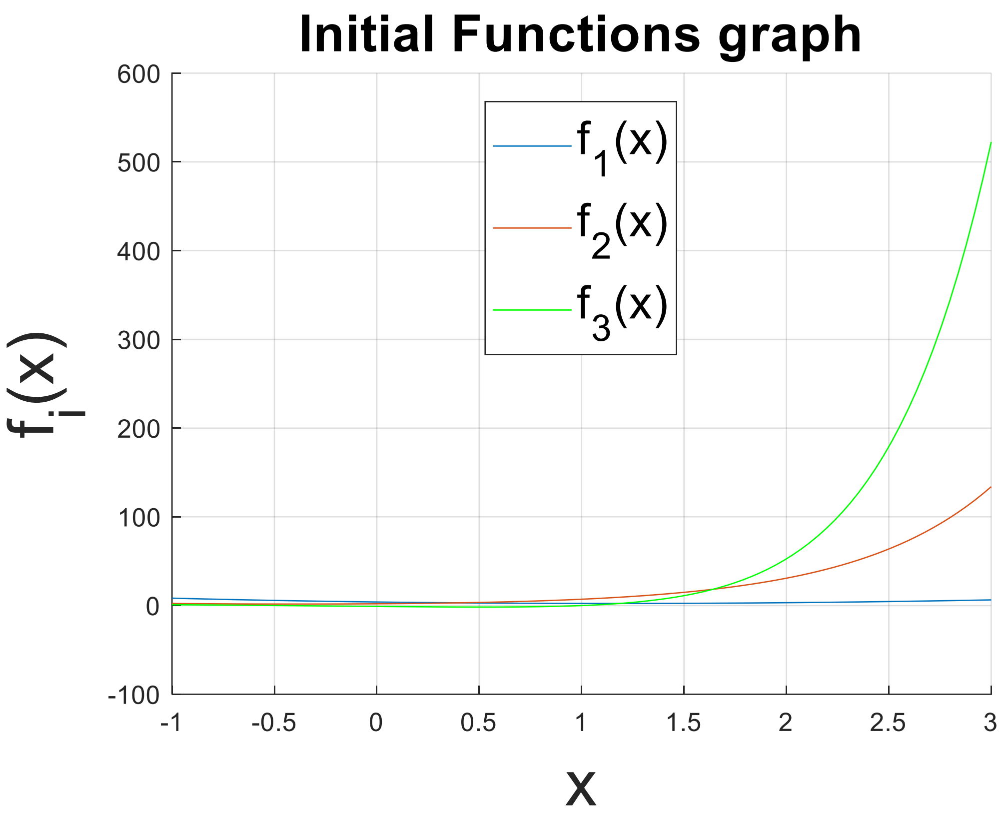

Limits in each iteration for l = 0.01[f2(x)]
ak/ bk
k
ak
bk
3
2.5
2
1.5
1
0.5
0
-0.5
-1
0
2
4
6
8
10
12
14


Limits in each iteration for l = 0.04[f1(x)]
ak/ bk
k
ak
bk
3
2.5
2
1.5
1
0.5
0
-0.5
-1
0
2
4
6
8
10
12
14


Limits in each iteration for l = 0.03[f1(x)]
ak/ bk
k
ak
bk
3
2.5
2
1.5
1
0.5
0
-0.5
-1
0
2
4
6
8
10
12
14


Limits in each iteration for l = 0.02[$f_1(x)$]
ak/ bk
k
ak
bk
3
2.5
2
1.5
1
0.5
0
-0.5
-1
0
2
4
6
8
10
12
14

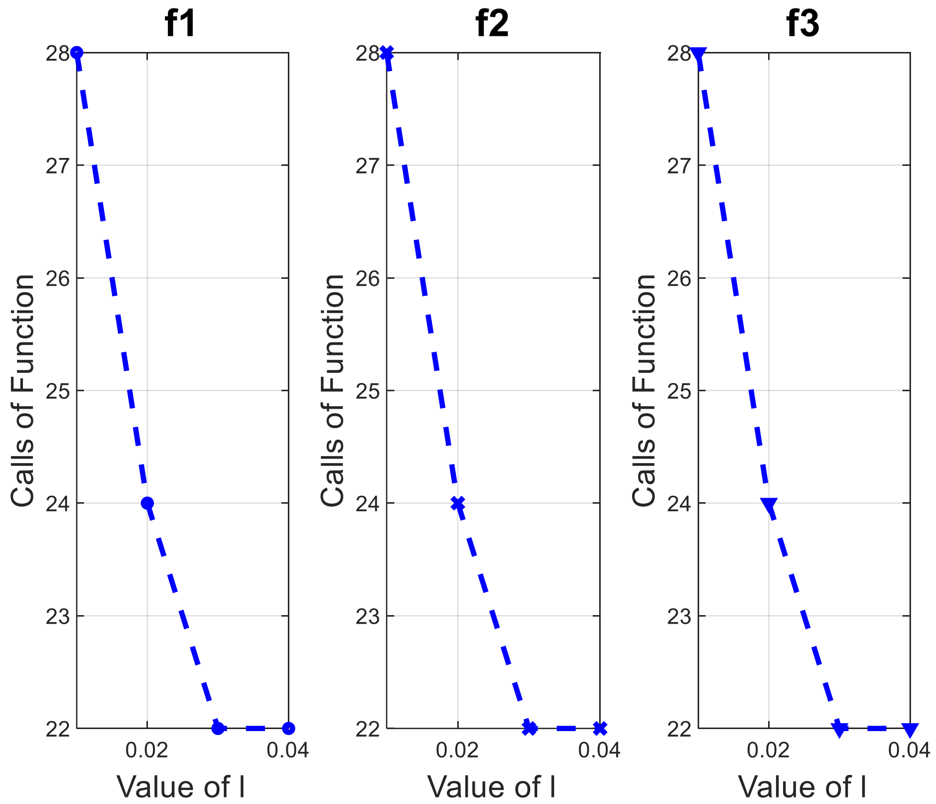

Limits in each iteration for l = 0.01[$f_1(x)$]
ak/ bk
k
ak
bk
3
2.5
2
1.5
1
0.5
0
-0.5
-1
0
2
4
6
8
10
12
14



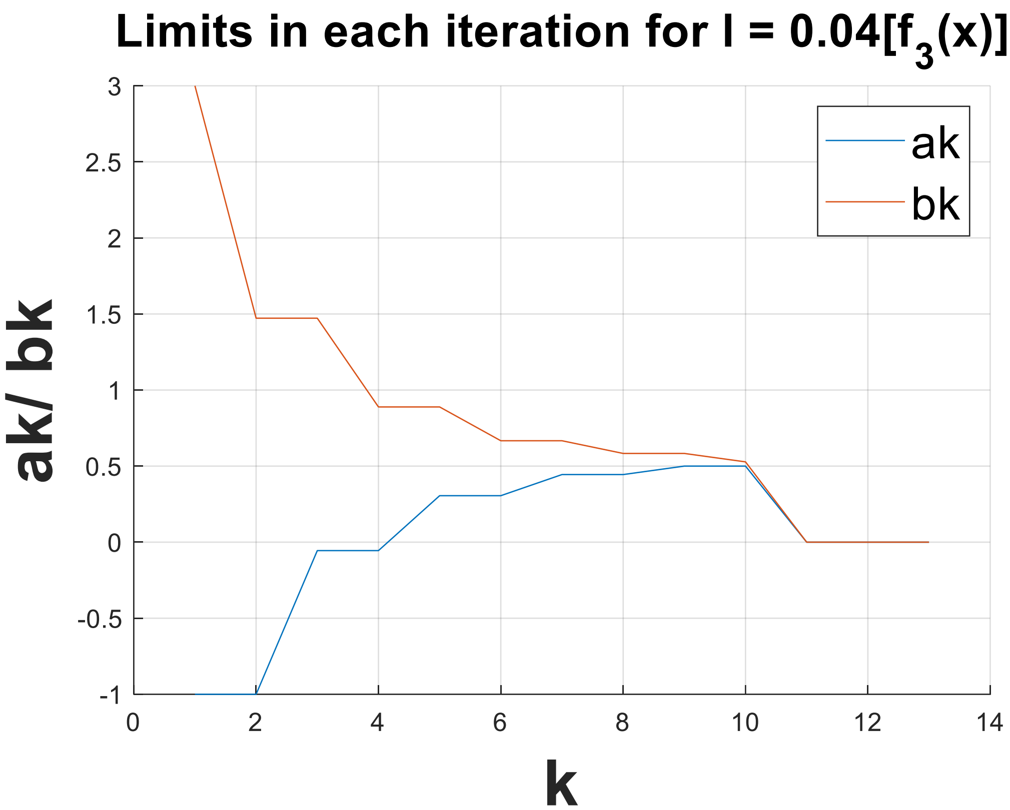
Τα προηγούμενα σχόλια που αφορούσαν τα άκρα του διαστήματος στην μέθοδο διχοτόμησης, ισχύουν και εδώ.

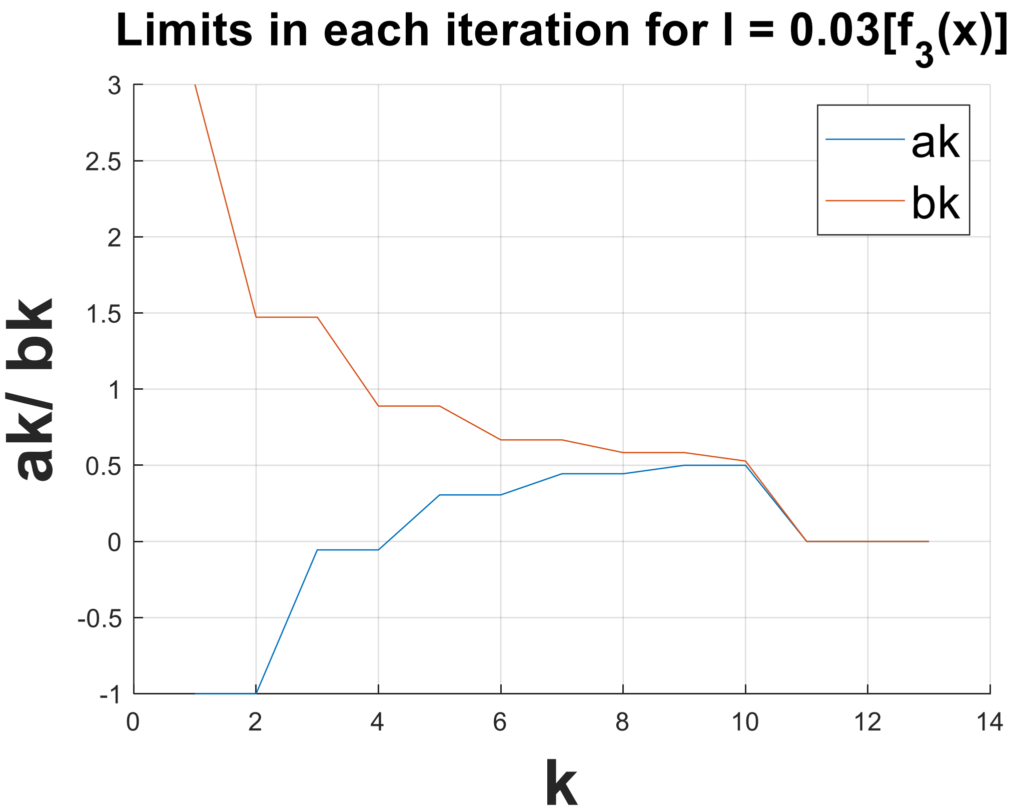
ΘΕΜΑ 3

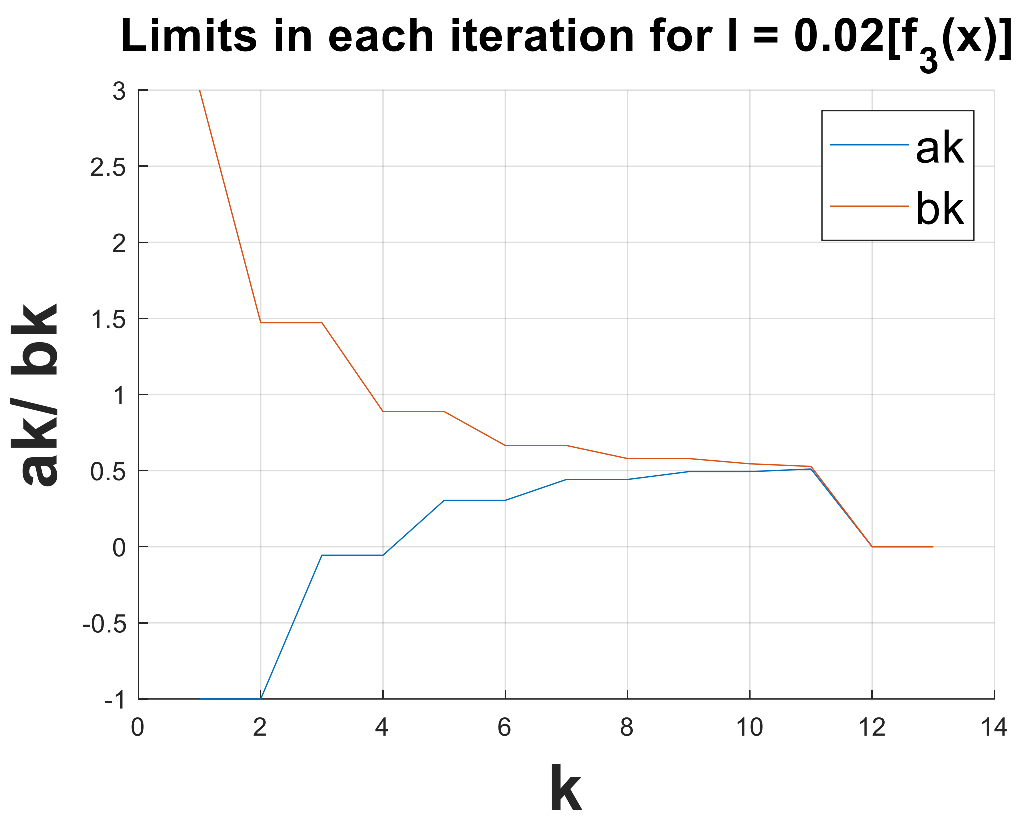
Το θέμα 3 χρησιμοποιεί τη μέθοδο Fibonacci. Ο φάκελος περιέχει 5 αρχεία, 3 τύπου function και 2 script. Το αρχείο τύπου function “Find_n” αφορά και τα δύο script και βρίσκει το βέλτιστο συνολικό αριθμό υπολογισμών των αντικειμενικών συναρτήσεων σε κάθε περίπτωση, ώστε να ικανοποιείται η συνθήκη του αλγορίθμου Fibonacci. Το δεύτερο αρχείο function “Fibonacci” καλείται μέσα στο αρχείο script “f_e_constant” και υλοποιεί την μέθοδο Fibonacci, έτσι ώστε τελικά να παρουσιασθεί η μεταβολή υπολογισμού των αντικειμενικών συναρτήσεων στο αρχείο “f_e_constant”.

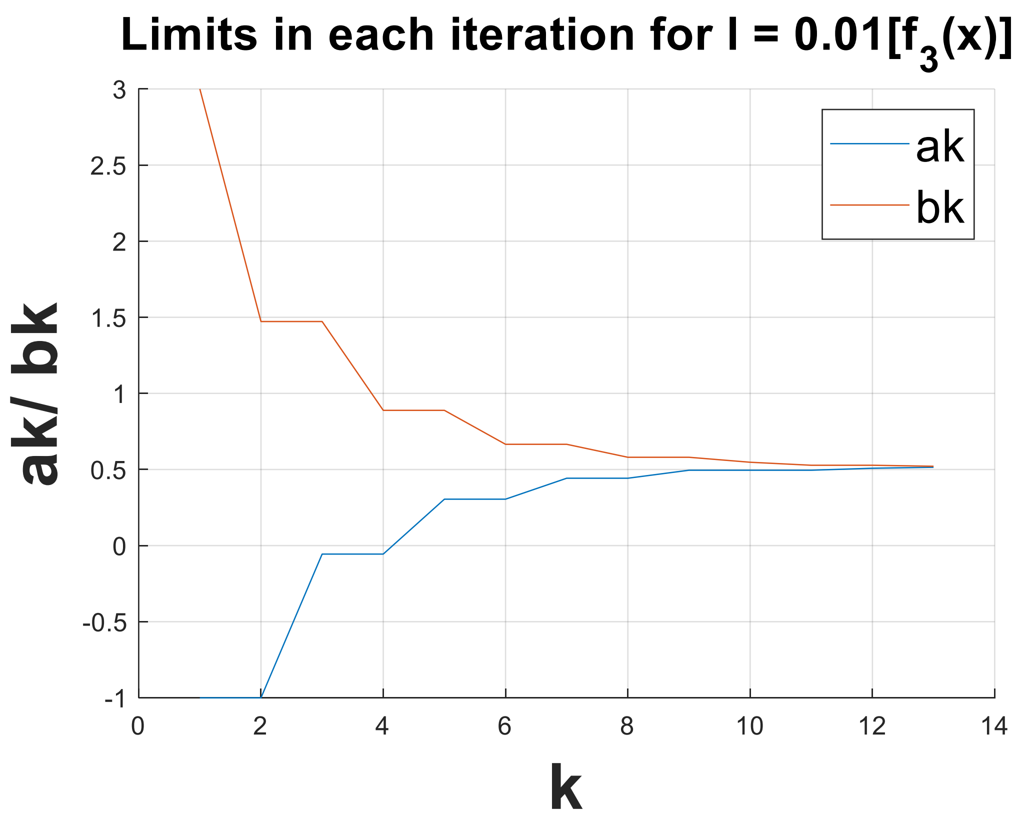


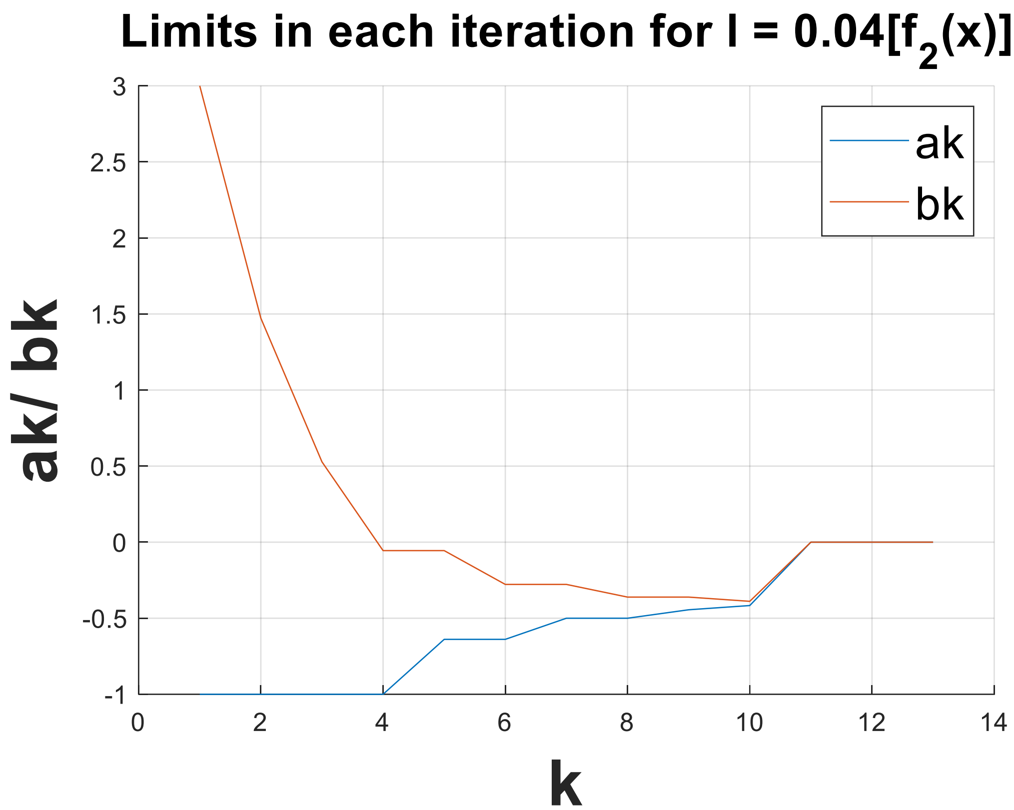
Το τελευταίο αρχείο function “Fibonacci Alternative” χρησιμοποιείται από το αρχείο script “ak_bk” και υλοποιεί επίσης την μέθοδο Fibonacci, αλλά δέχεται διαφορετικά ορίσματα και επιστρέφει άλλες μεταβλητές. Επιστρέφει τρεις δισδιάστατους πίνακες που αποθηκεύουν για κάθε l τις τιμές ak, bk , k ώστε να αναπαρασταθούν γραφικά. Το αρχείο “ak_bk” εμφανίζει τα διαγράμματα των άκρων των αντικειμενικών συναρτήσεων , συναρτήσει των επαναλήψεων κ, για διάφορες τιμές του τελικού εύρους αναζήτησης l. Επίσης σχεδιάζει τις αρχικές συναρτήσεις το πεδίο ορισμού της εκφώνησης.

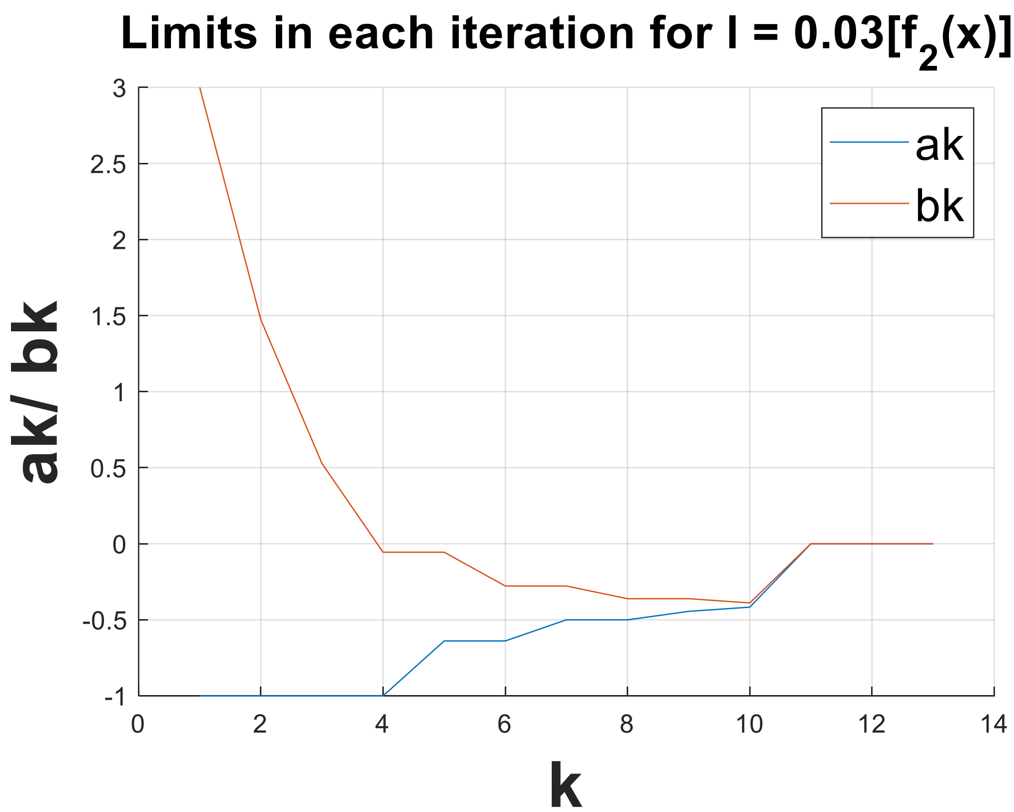

Limits in each iteration for l = 0.04[f3(x)]
ak
bk
ak/ bk
k
3
2.5
2
1.5
1
0.5
0
-0.5
-1
0
2
4
6
8
10
12
14

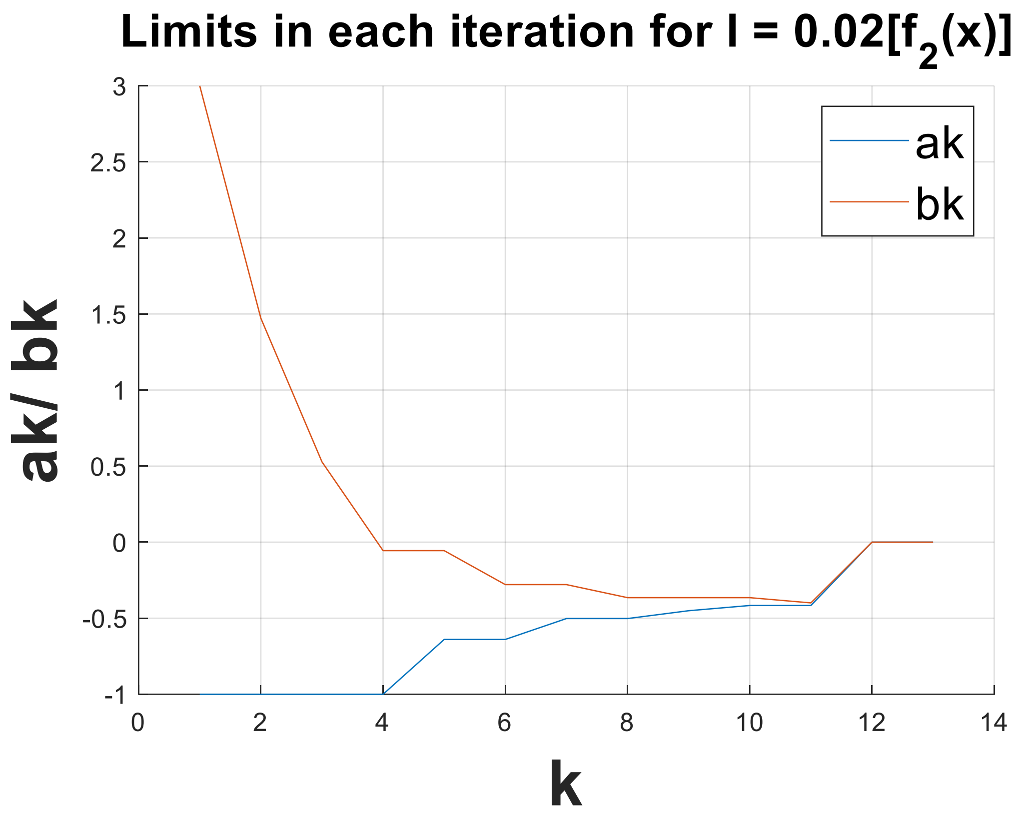

Limits in each iteration for l = 0.03[f3(x)]
ak
bk
ak/ bk
k
3
2.5
2
1.5
1
0.5
0
-0.5
-1
0
2
4
6
8
10
12
14

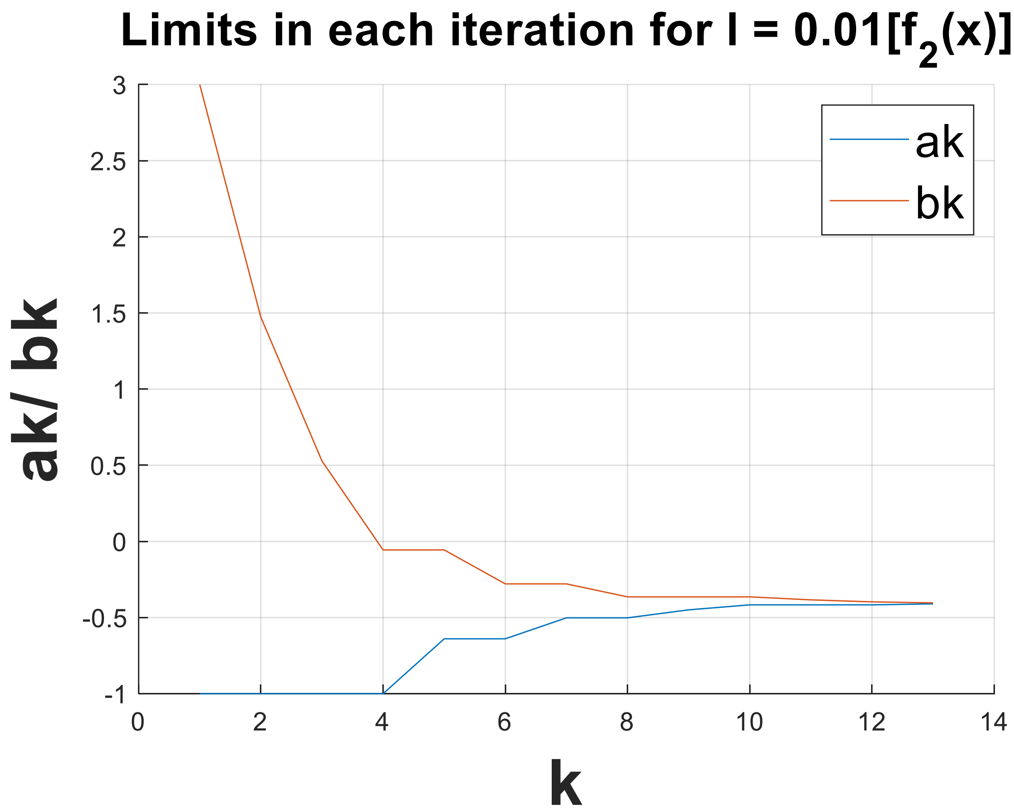

Limits in each iteration for l = 0.02[f3(x)]
ak/ bk
k
ak
bk

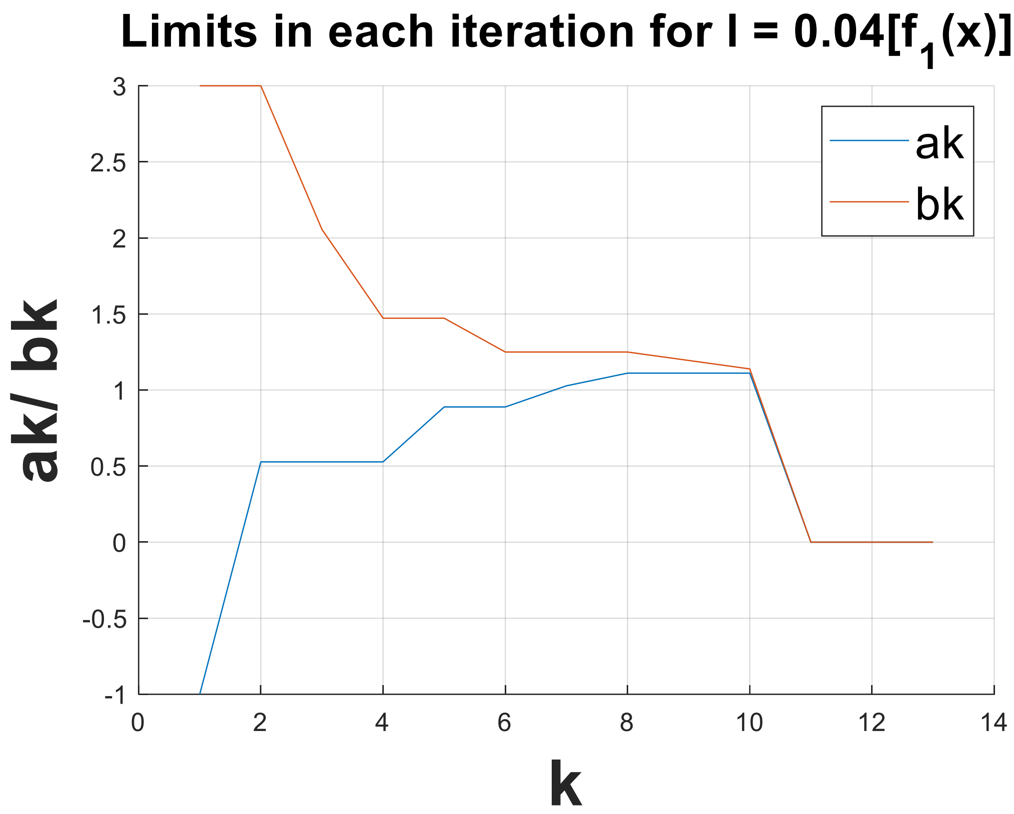

Limits in each iteration for l = 0.01[f3(x)]
ak/ bk
k
ak
bk

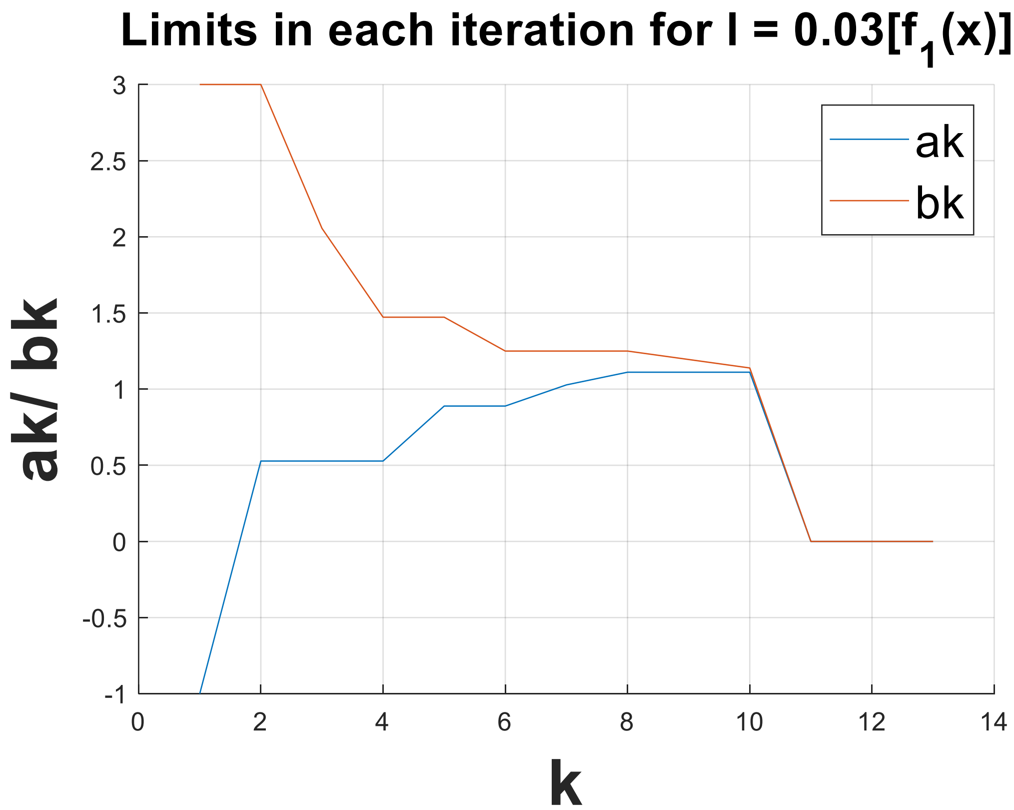

Limits in each iteration for l = 0.04[f2(x)]
ak/ bk
k
ak
bk
3
2.5
2
1.5
1
0.5
0
-0.5
-1
0
2
4
6
8
10
12
14

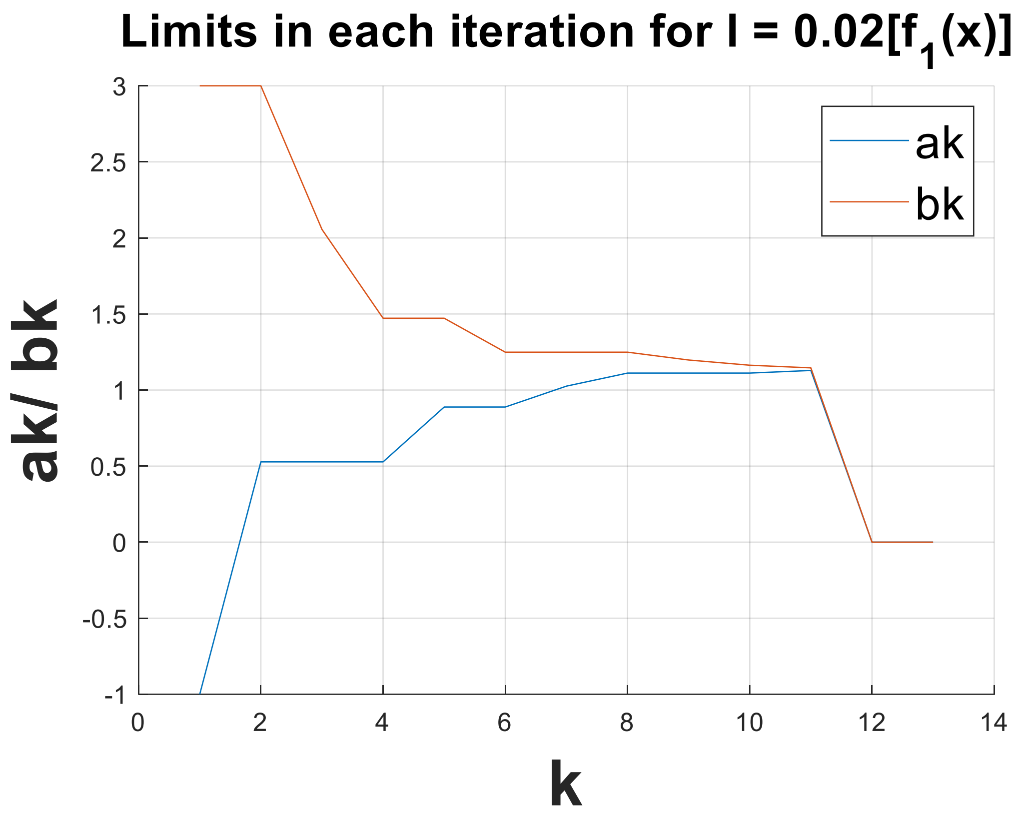

Limits in each iteration for l = 0.03[f2(x)]
ak/ bk
k
ak
bk
3
2.5
2
1.5
1
0.5
0
-0.5
-1
0
2
4
6
8
10
12
14


Limits in each iteration for l = 0.02[f2(x)]
ak
bk
ak/ bk
k
3
2.5
2
1.5
1
0.5
0
-0.5
-1
0
2
4
6
8
10
12
14

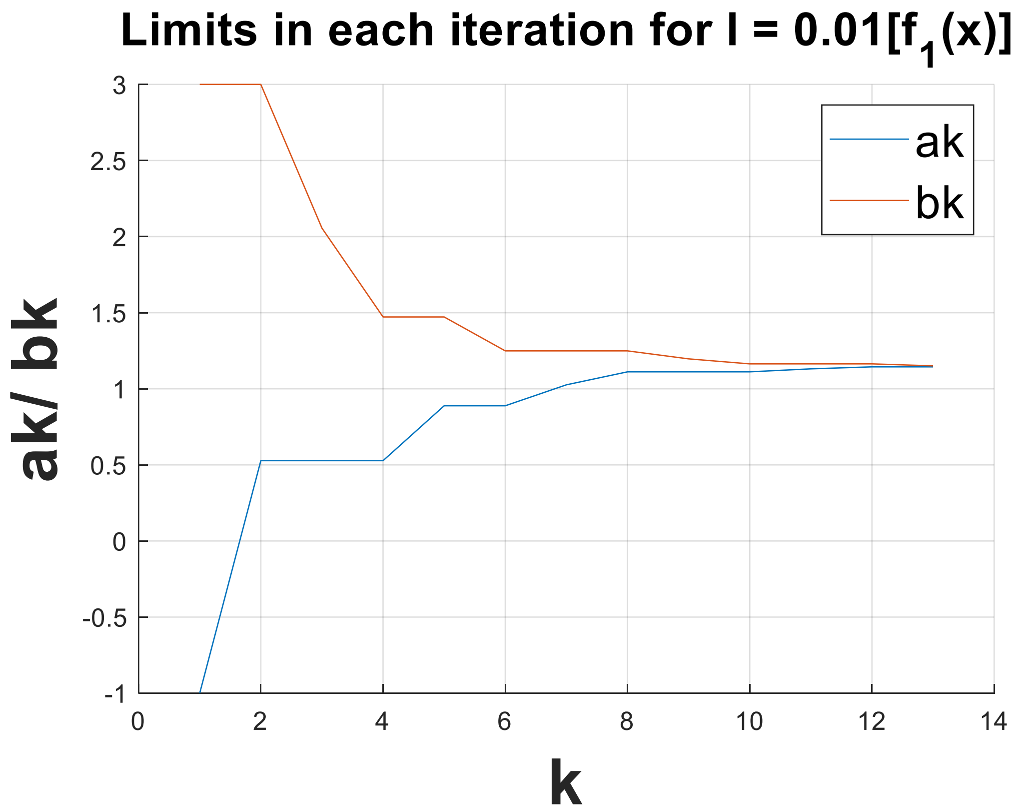

Limits in each iteration for l = 0.01[f2(x)]
ak
bk
ak/ bk
k
3
2.5
2
1.5
1
0.5
0
-0.5
-1
0
2
4
6
8
10
12
14


Limits in each iteration for l = 0.04[f1(x)]
ak/ bk
k
ak
bk
3
2.5
2
1.5
1
0.5
0
-0.5
-1
0
2
4
6
8
10
12
14

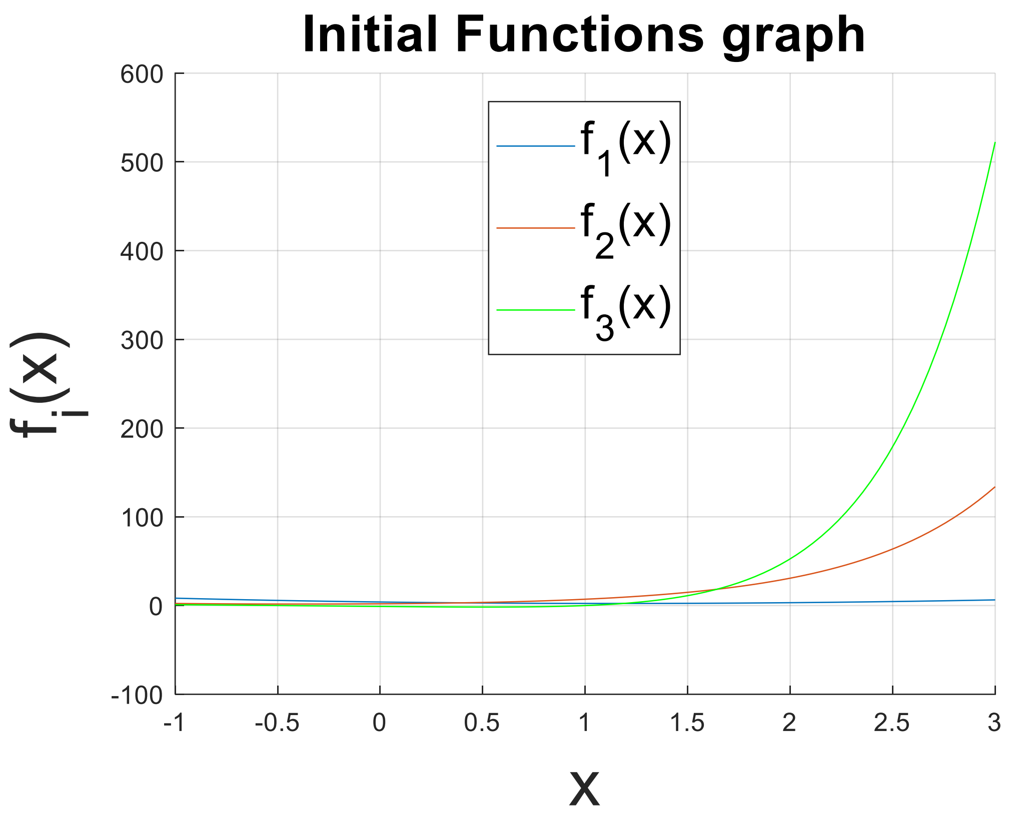

Limits in each iteration for l = 0.03[f1(x)]
ak/ bk
k
ak
bk
3
2.5
2
1.5
1
0.5
0
-0.5
-1
0
2
4
6
8
10
12
14



Limits in each iteration for l = 0.02[f_1(x)]
ak/ bk
k
ak
bk



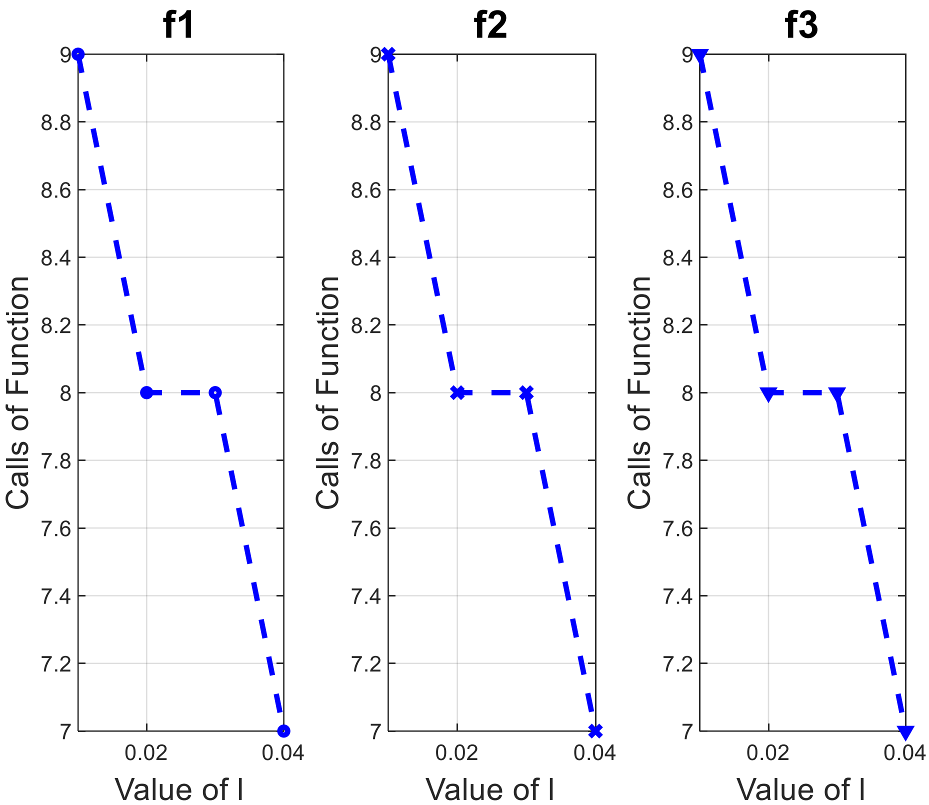
Limits in each iteration for l = 0.01[f_1(x)]
ak/ bk
k
ak
bk



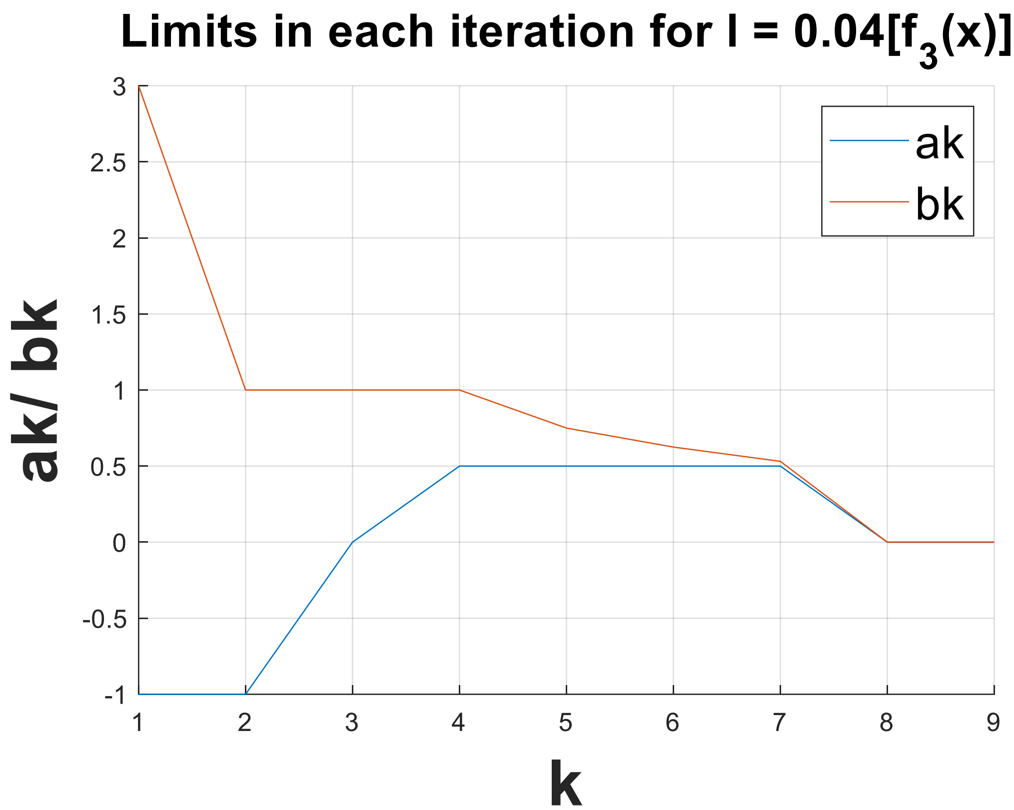
Οι παρατηρήσεις των μεθόδων της διχοτόμου και της Χρυσής Τομής ισχύουν και εδώ.

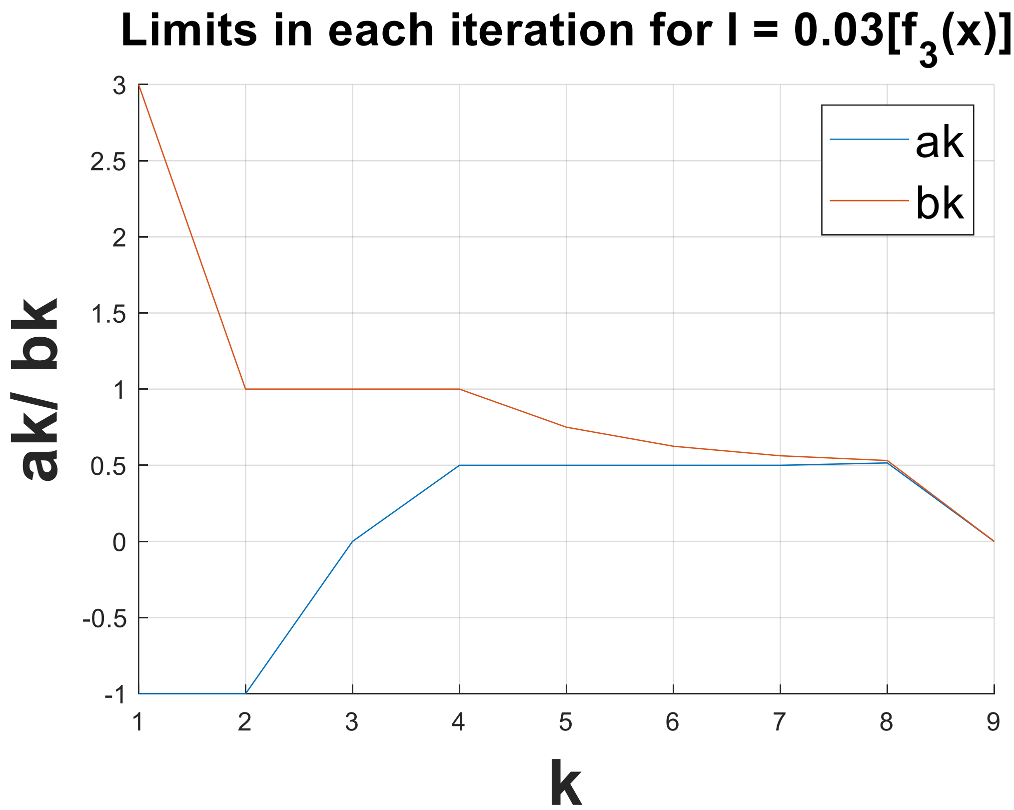
ΘΕΜΑ 4

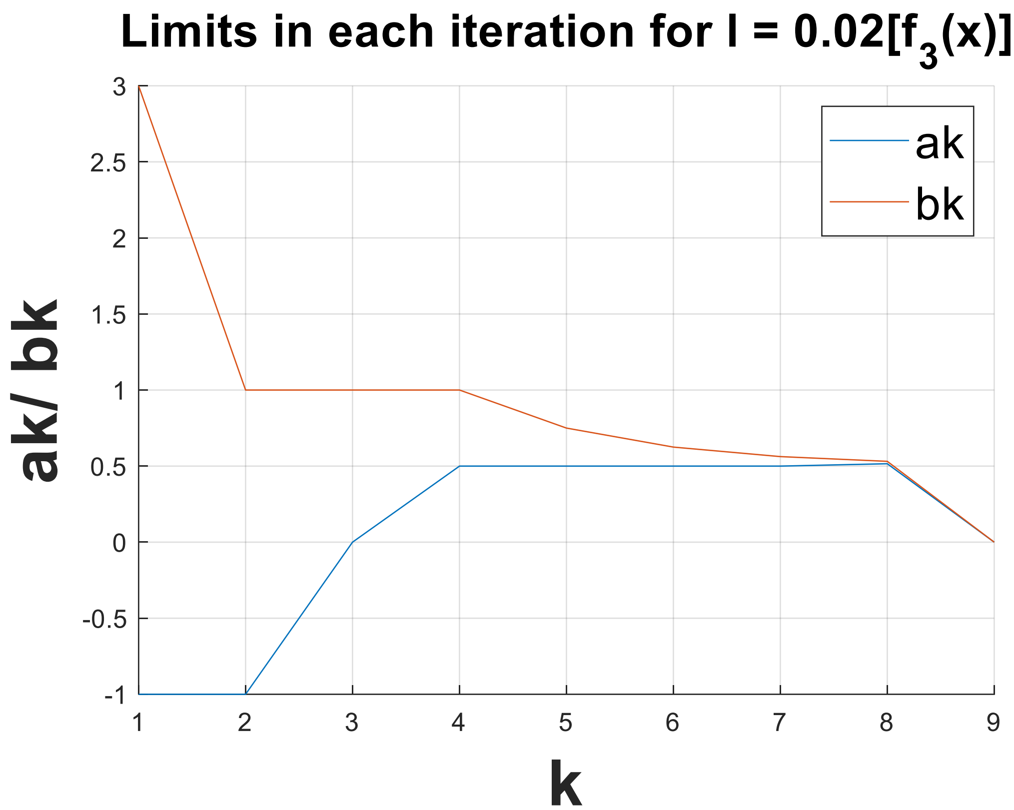
Υλοποίηση της Μεθόδου της διχοτόμου με χρήση παραγώγου. Ο φάκελος περιέχει 5 αρχεία, 3 τύπου function και 2 τύπου script. Το αρχείο function “Find_n” αφορά και τα δύο scripts και είναι υπεύθυνο για την εύρεση του αριθμού n υπολογισμών της αντικειμενικής συνάρτησης, δηλαδή τον μικρότερο θετικό ακέραιο που ικανοποιεί την προϋπόθεση της μεθόδου. Το αρχείο function “Bisection_Derivative” χρησιμοποιείται από το αρχείο script “f_e_constant” και υλοποιεί τη μέθοδο της διχοτόμου με χρήση παραγώγου και επιστρέφει τον αριθμό υπολογισμών της αντικειμενικής συνάρτησης. Το αρχείο script “f_e_constant” σχεδιάζει στη συνέχεια τη γραφική παράσταση συναρτήσει του τελικού εύρους αναζήτησης l.



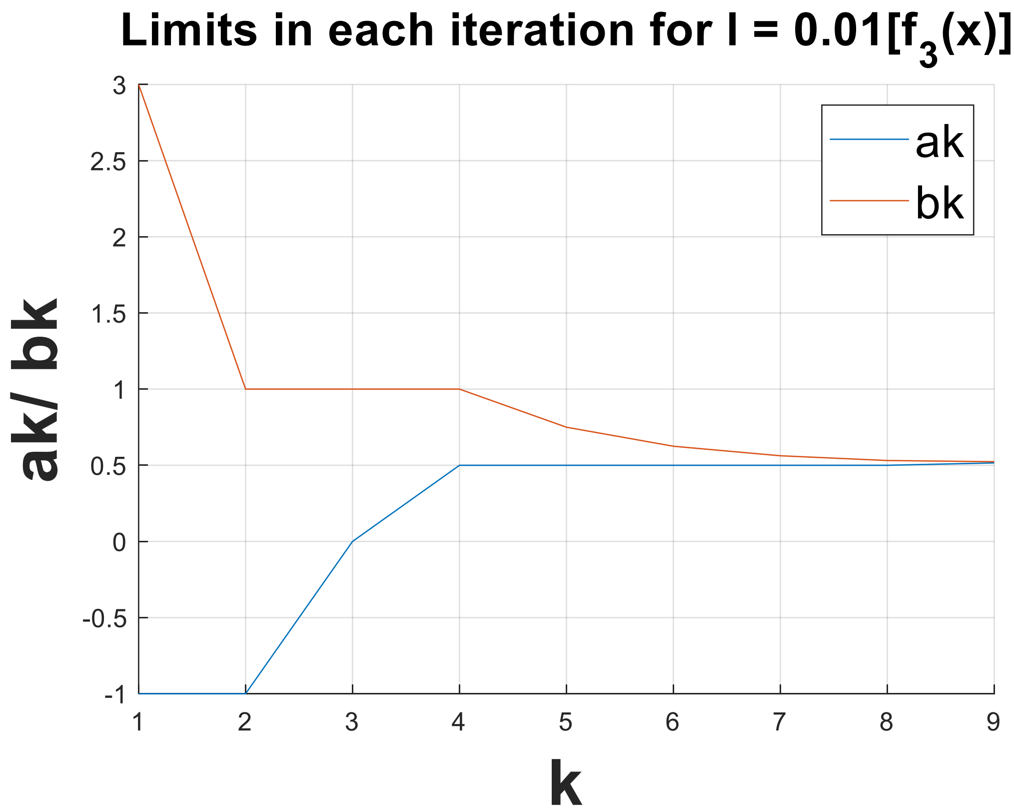
Το αρχείο function “Bisection_Derivative_Alternative” υλοποιεί επίσης την μέθοδο της διχοτόμου με χρήση παραγώγου, αλλά δέχεται διαφορετικά ορίσματα και επιστρέφει περισσότερες και διαφορετικές μεταβλητές. Επιστρέφει τρεις δισδιάστατους πίνακες που αποθηκεύουν για κάθε l τις τιμές ak, bk , k ώστε να αναπαρασταθούν γραφικά. Αξιοποιείται από το αρχείο script “ak_bk” , το οποίο μετά σχεδιάζε τις γραφικές παραστάσεις των άκρων των αντικειμενικών συναρτήσεων συναρτήσει των επαναλήψεων κ, για διάφορες τιμές του τελικού εύρους αναζήτησης l. Σχεδιάζει, τέλος, τις γραφικές παραστάσεις των αρχικών συναρτήσεων στο πεδίο ορισμού που δίνεται.


Limits in each iteration for l = 0.04[f3(x)]
ak/ bk
k
ak
bk
3
2.5
2
1.5
1
0.5
0
-0.5
-1
1
2
3
4
5
6
7
8
9

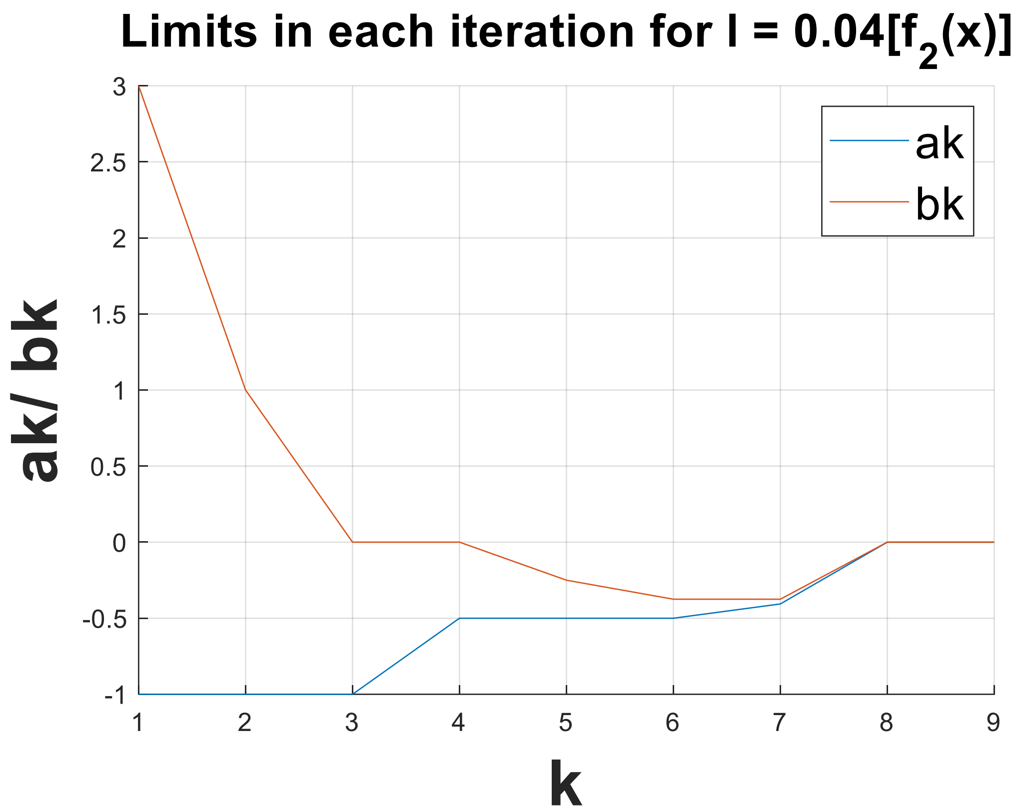

Limits in each iteration for l = 0.03[f3(x)]
ak/ bk
k
ak
bk
3
2.5
2
1.5
1
0.5
0
-0.5
-1
1
2
3
4
5
6
7
8
9

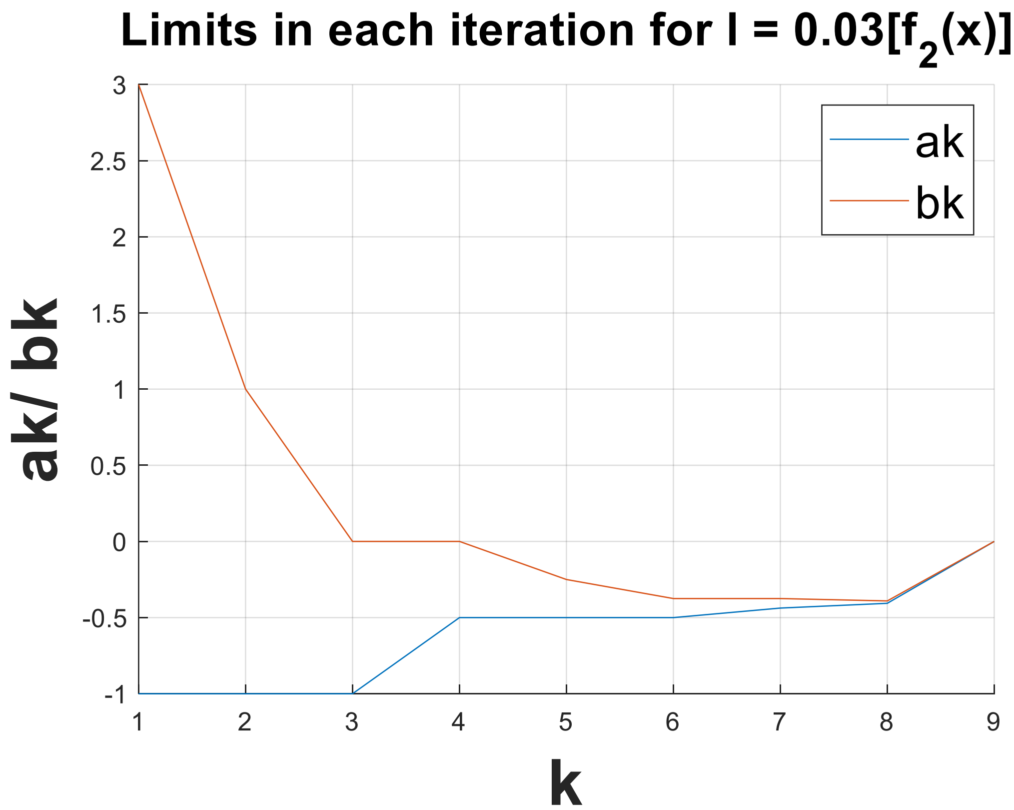


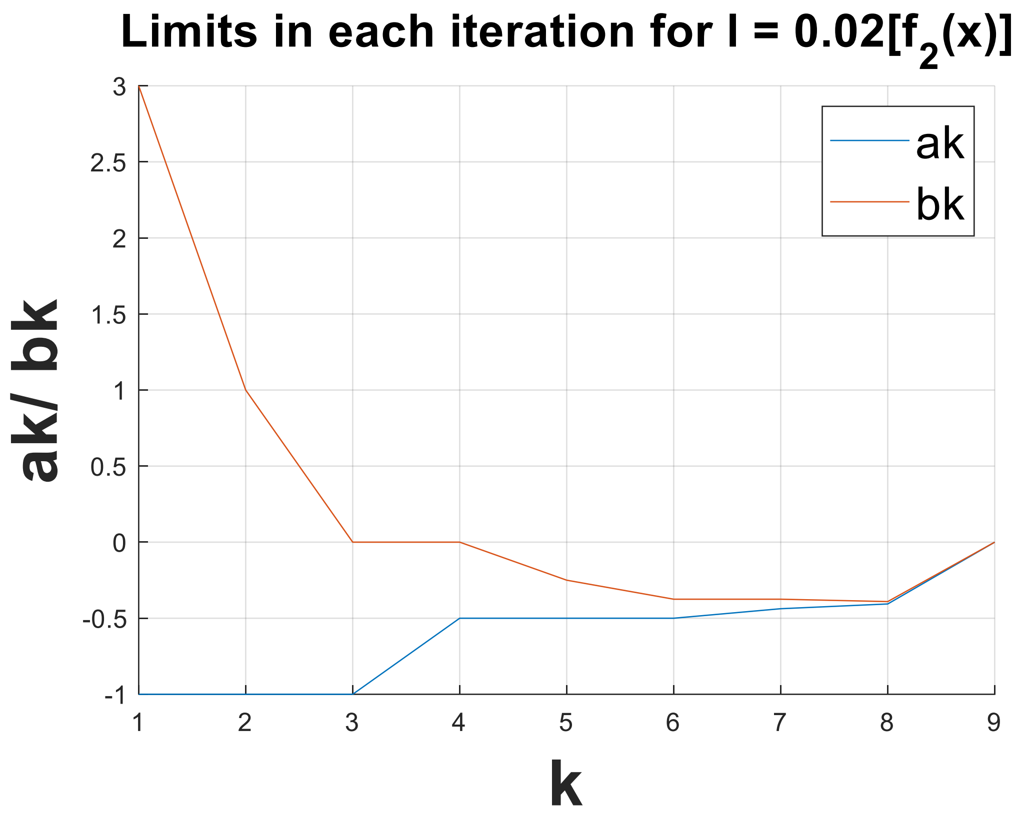
Limits in each iteration for l = 0.02[f3(x)]
ak
bk
ak/ bk
k
3
2.5
2
1.5
1
0.5
0
-0.5
-1
1 2 3 4 5 6 7 8 9

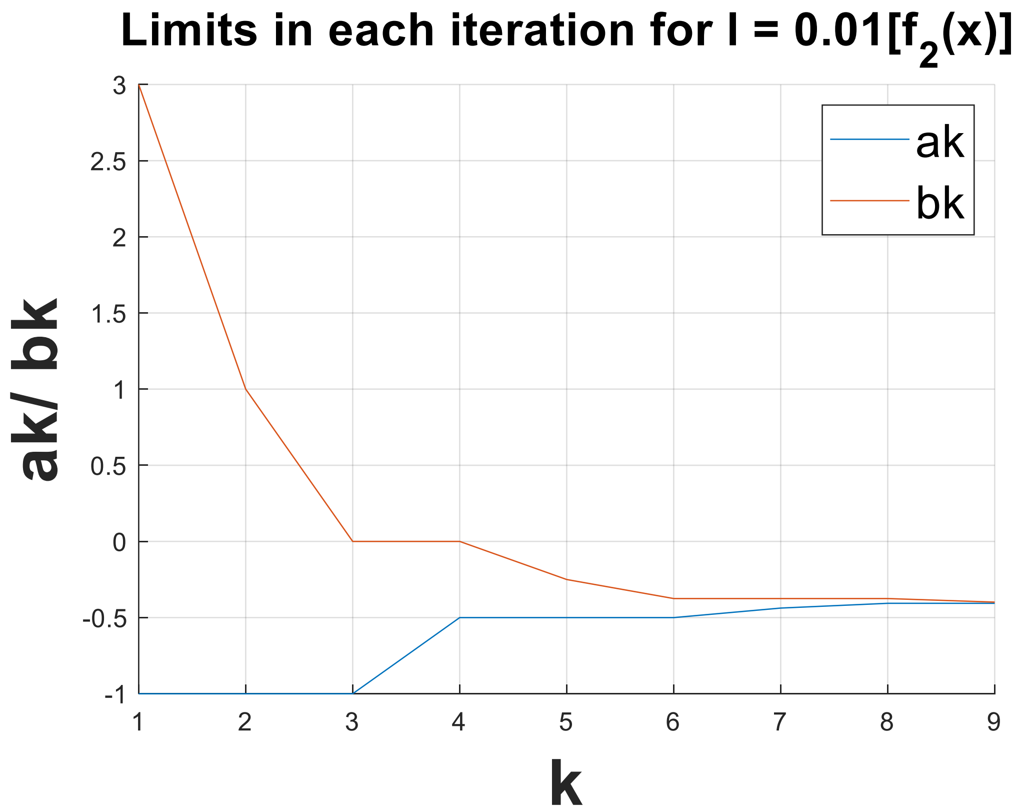


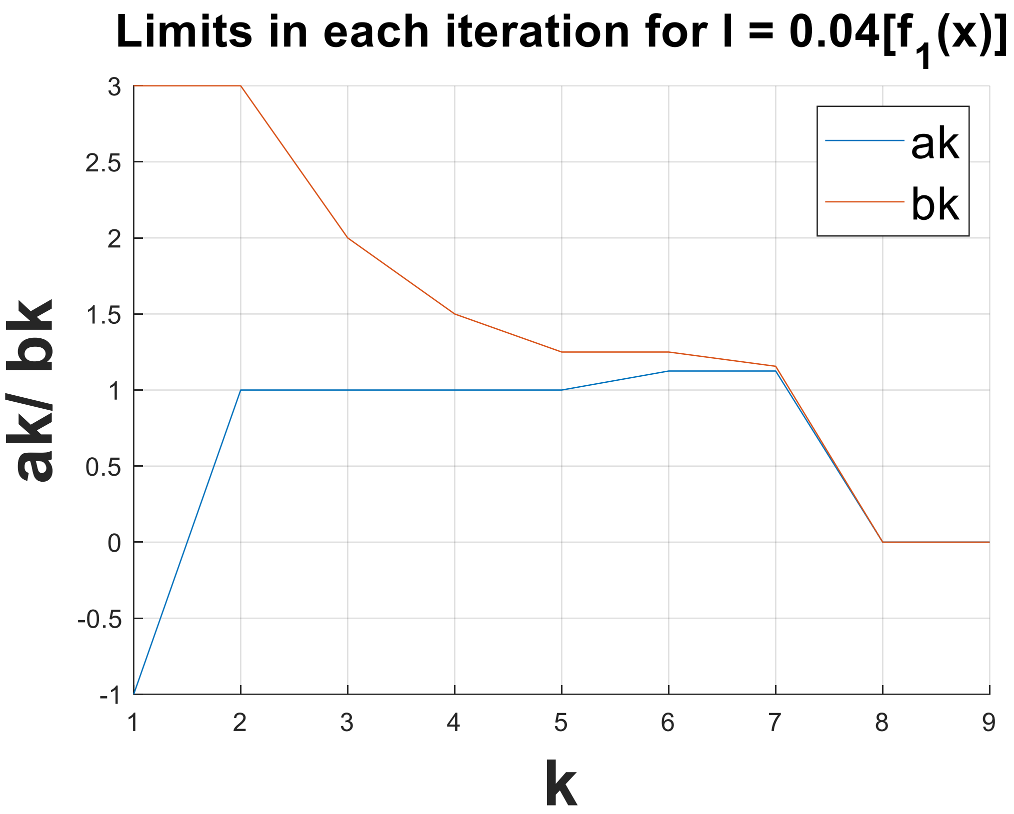
Limits in each iteration for l = 0.01[f3(x)]
ak
bk
ak/ bk
k
3
2.5
2
1.5
1
0.5
0
-0.5
-1
1 2 3 4 5 6 7 8 9

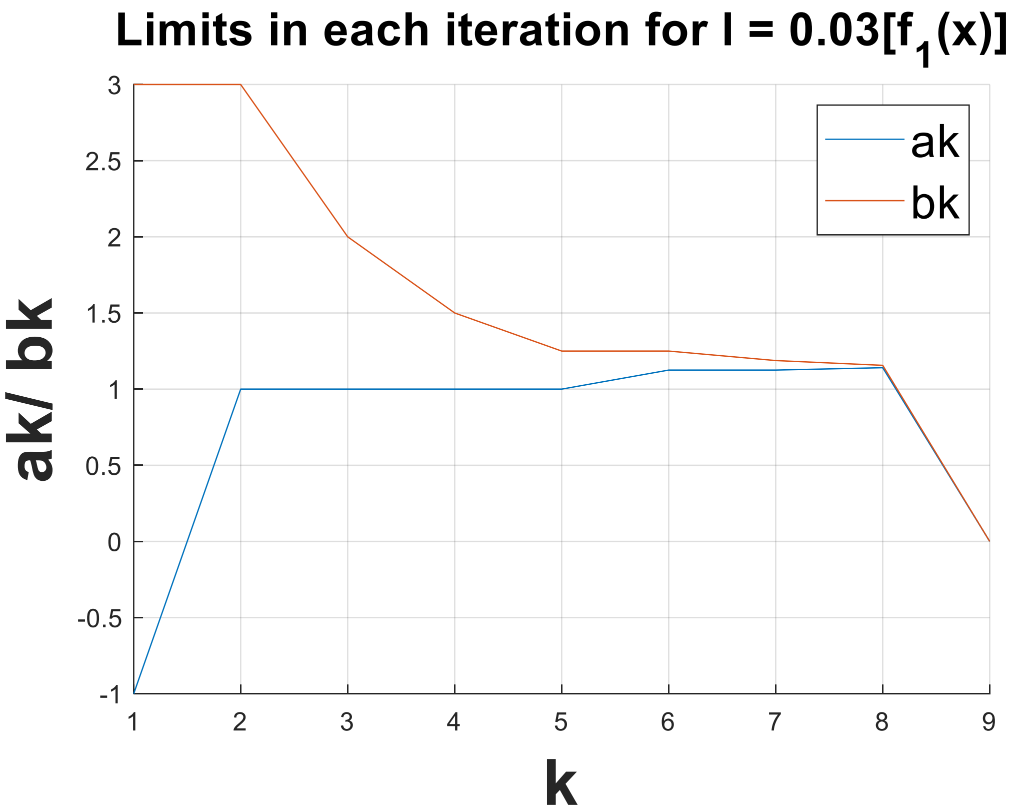

Limits in each iteration for l = 0.04[f_2(x)]
ak
bk
ak/ bk
k

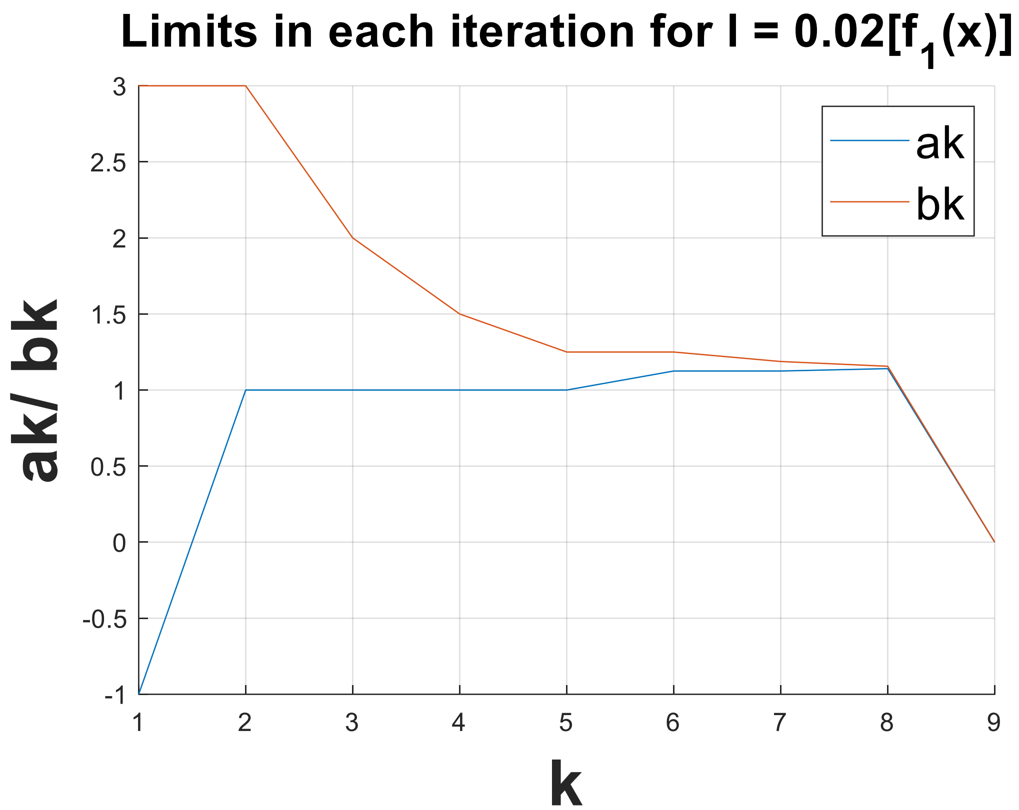

Limits in each iteration for l = 0.03[f_2(x)]
ak
bk
ak/ bk
k

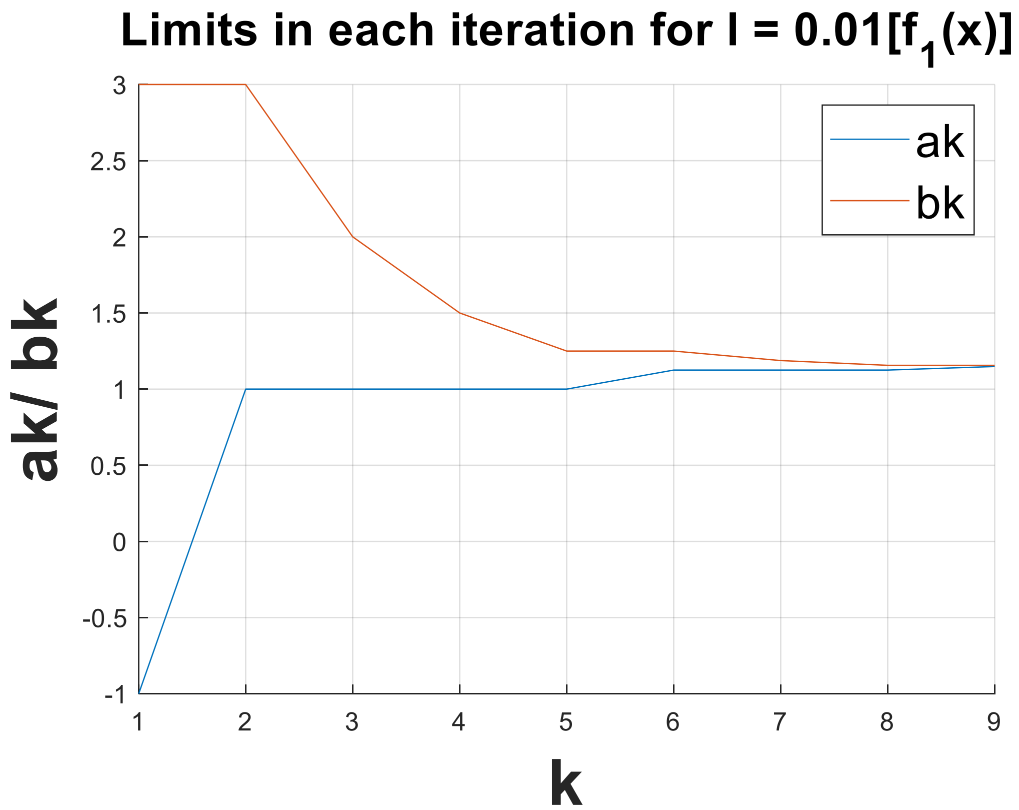

Limits in each iteration for l = 0.02[f_2(x)]
ak/ bk
k
ak
bk
3
2.5
2
1.5
1
0.5
0
-0.5
-1
1
2
3
4
5
6
7
8
9

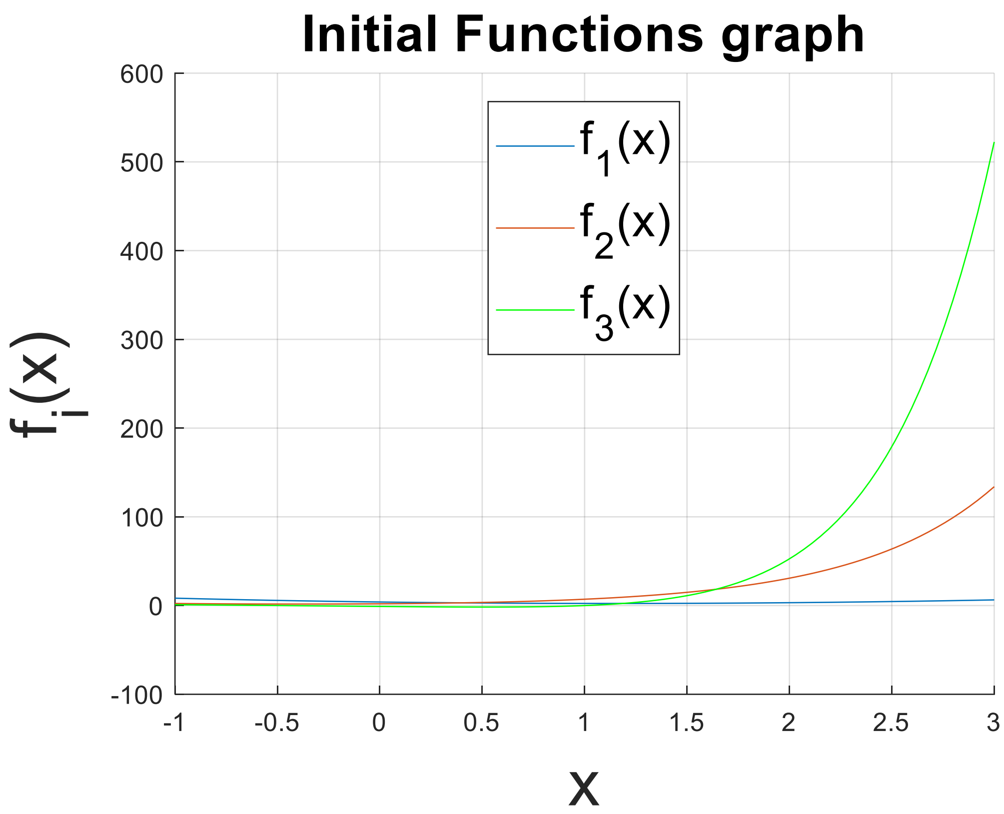

Limits in each iteration for l = 0.01[f_2(x)]
ak/ bk
k
ak
bk
3
2.5
2
1.5
1
0.5
0
-0.5
-1
1
2
3
4
5
6
7
8
9


Limits in each iteration for l = 0.04[f1(x)]
ak
bk
ak/ bk
k


Limits in each iteration for l = 0.03[f1(x)]
ak
bk
ak/ bk
k


Limits in each iteration for l = 0.02[f1(x)]
ak/ bk
k
ak
bk


Limits in each iteration for l = 0.01[f1(x)]
ak/ bk
k
ak
bk



Συμπεράσματα:

Η μέθοδος της διχοτόμου είναι απλοϊκή , δεν απαιτεί πολύπλοκους υπολογισμούς και όσο πιο πολλές επαναλήψεις υπάρχουν, τόσο πιο ακριβής γίνεται. Ωστόσο, σε μερικές συναρτήσεις αδυνατεί να βρει την επιθυμητή τιμή και πάντα διατηρεί ένα περιθώριο σφάλματος.

Η Μέθοδος του Χρυσού Τομέα πλεονεκτεί στο ότι έχει την δυνατότητα να βρίσκει το ακρότατο μίας συνάρτησης. Όμως η αποτελεσματικότητά της οφείλεται σε μεγάλο βαθμό στην επιλογή των σημείων υπολογισμού της αντικειμενικής συνάρτησης και συνήθως είναι πιο αργή από την μέθοδο της διχοτόμου.

Η μέθοδος Fibonacci μοιάζει με αυτή της Χρυσής τομής, αλλά δεν απαιτεί σταθερό λόγο με τον οποίο να αλλάζει τα διαστήματα. Επιπροσθέτως, απαιτεί εκ των προτέρων προσδιορισμό των συνολικών υπολογισμών της αντικειμενικής συνάρτησης. Η μέθοδος Fibonacci είναι η πιο αποτελεσματική, όπως φαίνεται και από τα διαγράμματα, όπου και στις 3 συναρτήσεις το τελικό παραγόμενο πεδίο μέσα στο οποίο βρίσκεται το ελάχιστο σημείο, ήταν το μικρότερο από όλες τις άλλες μεθόδους. Σημειωτέον πως για πολύ μεγάλο n, τότε η μέθοδος Fibonacci με την μέθοδο χρυσής τομής θα ήταν σχεδόν ταυτόσημες.

Η μέθοδος της διχοτόμου με χρήση παραγώγου απαιτεί τις λιγότερες επαναλήψεις k , ακολουθόμενη από την μέθοδο της διχοτόμου. Παρόλα αυτά η πρώτη παρουσιάζει ακόμη τις λιγότερες κλήσεις των αντικειμενικών συναρτήσεων.

Και οι 4 μέθοδοι είναι εξαιρετικά αποτελεσματικές για την εύρεση ελαχίστου μίας κυρτής συνάρτησης. Η Fibonacci υπερέχει των υπολοίπων σε όλες τις περιπτώσεις και ακολουθεί η μέθοδος της Χρυσής τομής στην περίπτωση της f1 και της f3,όπως ισχύει και γενικά .Για την f2 παρόλα αυτά, δεύτερη αποτελεσματικότερη είναι η μέθοδος της διχοτόμησης με χρήση παραγώγου. Η μέθοδος της διχοτόμησης είναι η πιο αναποτελεσματική από όλες τις μεθόδους σε όλες τις περιπτώσεις, εκτός από τη συνάρτηση f1 όπου είναι η δεύτερη χειρότερη, ξεπερνώντας την μέθοδο της διχοτόμησης με χρήση παραγώγου.