Τεχνικές Βελτιστοποίησης

Ιωάννης-Παναγιώτης Μπουντουρίδης

ΑΕΜ: 8872

work 1

Καθώς οι συναρτήσεις που μας δίνονται είναι κυρτές(σχεδόν κυρτές) και ορισμένες σε ένα διάστημα [a,b] είναι εμφανές οτι μπορούμε να χρησιμοποιήσουμε το θεώρημα 5.5.1 και κατ επέκταση, τους αλγορίθμους που βασίζονται σε αυτό το θεώρημα στη συγκεκριμένη εργασία.

Θέμα 1

Λίγα λόγια για τον αλγόριθμο

Σε αυτό το θέμα θα υλοποιήσουμε στο matlab τη μέθοδο της διχοτόμου την οποία θα εφαρμόσουμε στις συναρτήσεις και θα σχολιάσουμε τα αποτελέσματα.

Η μέθοδος της διχοτόμου αν και απλή ,οπως θα δούμε είναι μία αργή μέθοδος. Οι παράμετροι που απαιτούνται είναι το ε δηλαδή η απόσταση των x_1, x_2 από τη διχοτόμο και το l το οποίο δίνει τη συνθήκη τερματισμού δηλαδή, σε τι απόσταση πρέπει να βρίσκονται τα άκρα του διαστήματος ώστε να έχουμε μία ικανοποιητική προσέγγιση του ελαχίστου της συνάρτησης.

Περιγραφή αλγόριθμου

Ο αλγόριθμος της διχοτόμου δεδομένων των $\varepsilon > 0$ και $l > 0$ ξεκινάει με το αρχικό διάστημα $[a_1, b_1]$ και σε ένα βρόγχο (while) με συνθήκη τερματισμού $\boxed{b_k - a_k < l}$ επιλέγει τα σημεία

$$\boxed{x_{1k} = \frac{a_k + b_k}{2} - \varepsilon} \quad \boxed{x_{2k} = \frac{a_k + b_k}{2} + \varepsilon}$$

Αν $\boxed{f(x_{1k}) < f(x_{2k}) \Rightarrow a_{k+1} = a_k \;\; \kappa\alpha\iota \;\; b_{k+1} = x_{2k}}$

διαφορετικά $\boxed{f(x_{1k}) > f(x_{2k}) \Rightarrow a_{k+1} = x_{1k} \;\; \kappa\alpha\iota \;\; b_{k+1} = b_k}$

Υπολογισμοί της αντικειμενικής συνάρτησης

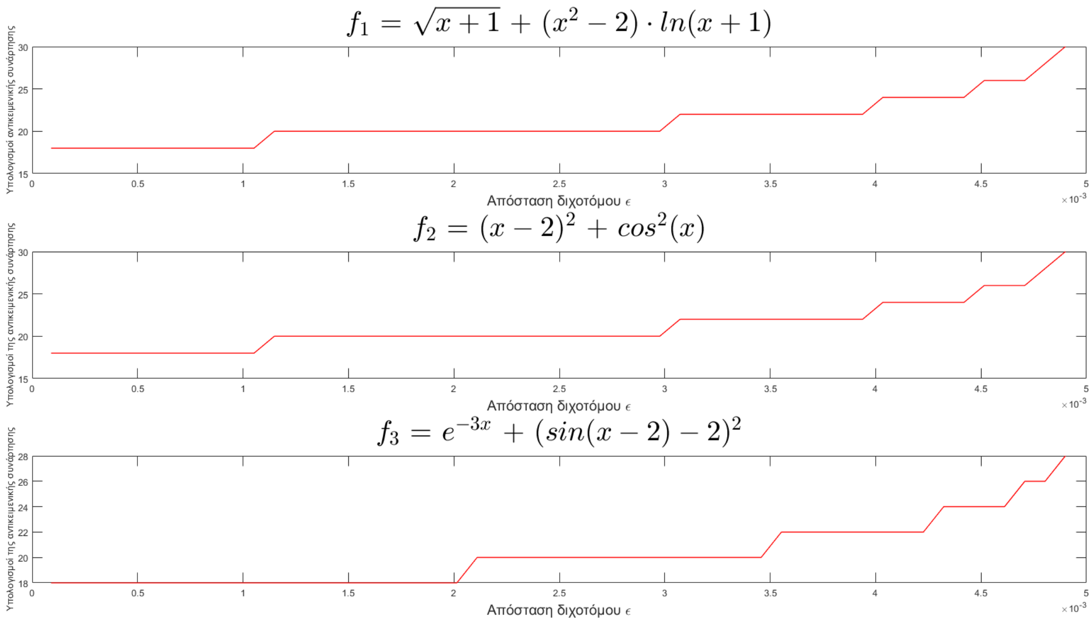
Όπως φαίνεται στην περιγραφή του αλγόριθμου σε κάθε επανάληψη υπολογίζουμε τις μεταβλητές x_{1k} και x_{2k} ώστε να προκύψει ανάλογα με την περίπτωση $f(x_{1k}) < f(x_{2k})$ ή $f(x_{1k}) > f(x_{2k})$ το νέο εύρος $[a_k, b_k]$. Για k=1 δεν έχουμε κανένα υπολογισμό της αντικειμενικής συνάρτησης ενω για k=2 έχουμε δύο υπολογισμούς $f(x_1)$ και $f(x_2)$ για τον έλεγχο της συνθήκης. Καταλήγουμε οτι οι υπολογισμοί της αντικειμενικής συνάρτησης συνδέονται απο την σχέση:

$$\boxed{calc(k) = 2(k-1)}$$

Σταθερό l

Για σταθερό τελικό εύρος αναζήτησης $\boxed{l = 0.01}$ μελετάμε τη μεταβολή των υπολογισμών της αντικειμενικής συνάρτησης καθώς μεταβάλλουμε την απόσταση απο τη διχοτόμο (δηλαδή την μεταβλητή ε)

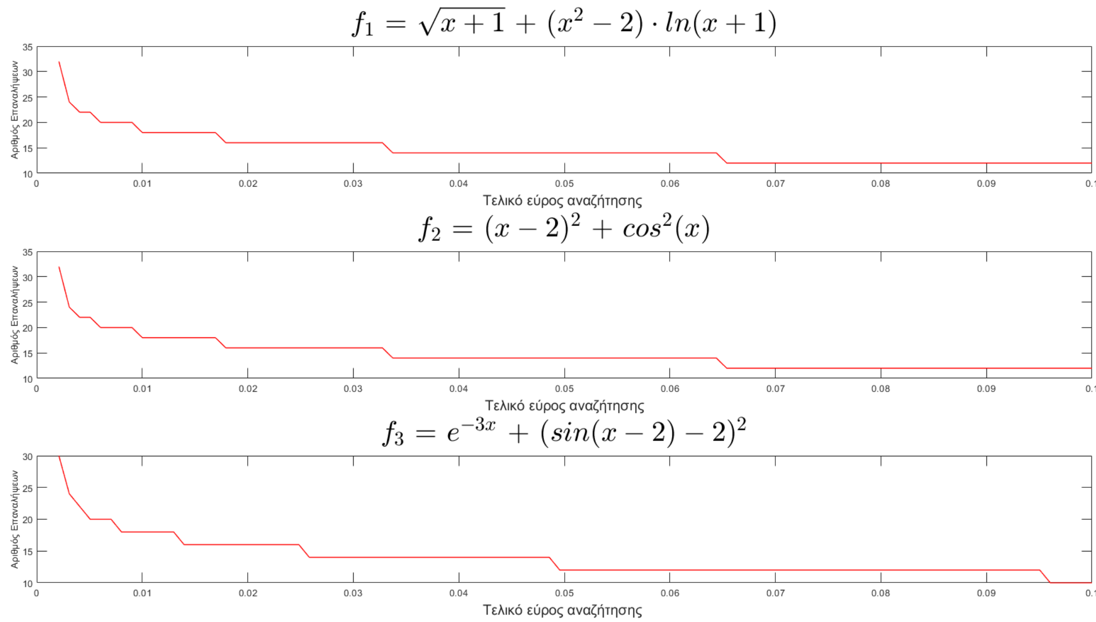
Για $\boxed{0.00009 \leq \varepsilon \leq 0.00049}$ επιλέγουμε 51 σημεία.



Παρατηρούμε οτι για $\boxed{\varepsilon \geq l/2}$ ο αλγόριθμος δεν τερματίζει ενώ για τιμές κοντά στο $\boxed{\varepsilon = l/2}$ η επαναλήψεις αυξάνονται σημαντικά.

Σταθερό ε

Αντίστοιχα για σταθερό $\boxed{\varepsilon = 0.001}$ οσο μεγαλώνει το l τόσο μειώνονται οι επαναλήψεις.

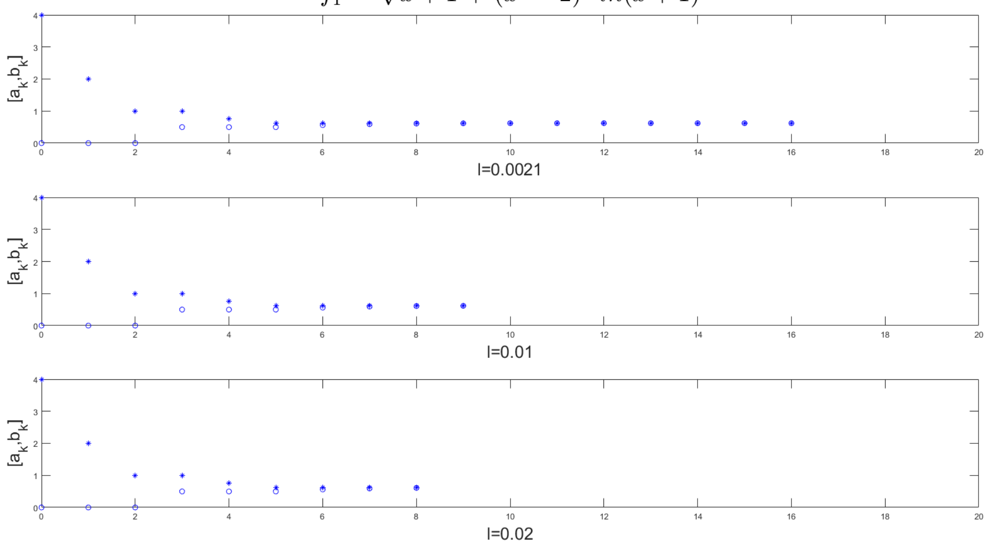


Ο αριθμός των υπολογισμών της αντικειμενικής συνάρτησης ικανοποιεί την σχέση: $\boxed{n \geq 2 \cdot log_2(\frac{b_1 - a_1}{l - \varepsilon})}$ ενώ οπως αποδείξαμε πιο πάνω ο αριθμός των κλήσεων n συνδέεται με τον αριθμό των επαναλήψεων k με την σχέση: $\boxed{n = 2(k-1)}$

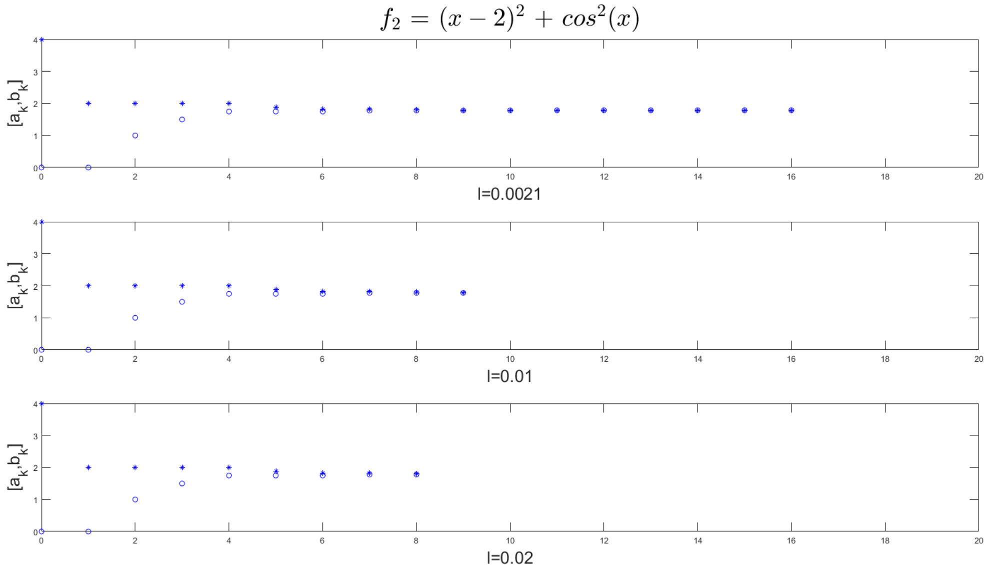
Μεταβολή των υποδιαστημάτων $[a_k, b_k]$

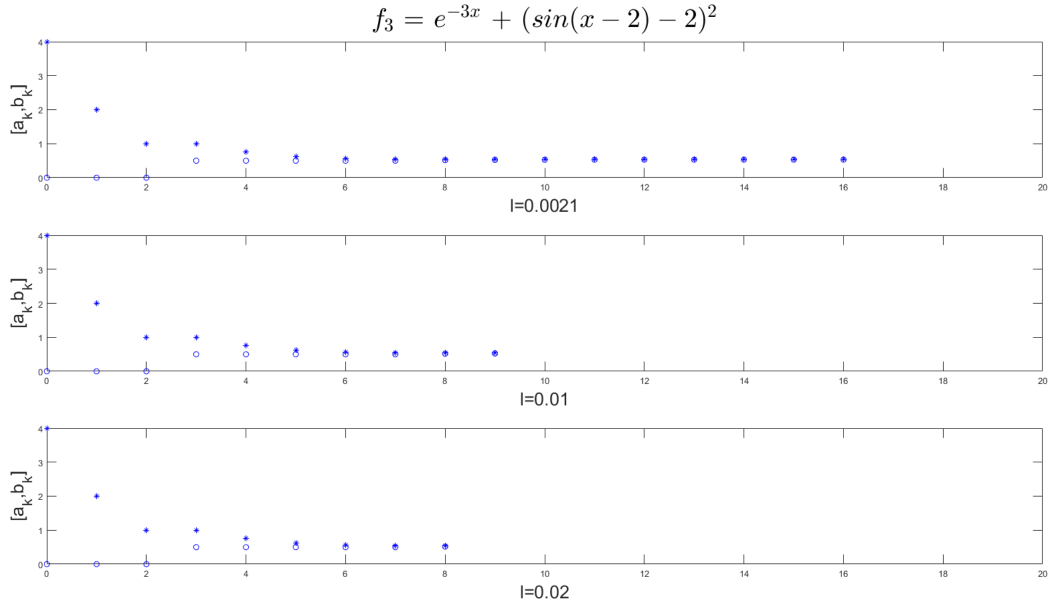
Τέλος παρουσιάζονται τα διαστήματα $[\alpha_k, b_k]$ ανα επανάληψη k για διάφορα l

$$f_1 = \sqrt{x+1} + (x^2-2) \cdot ln(x+1)$$



$$f_2 = (x-2)^2 + cos^2(x)$$





Όπως φαίνεται απο τα διαγράμματα ο αλγόριθμος συγκλίνει στο ελάχιστο πιο γρήγορα οσο το l μεγαλώνει με μικρότερη ακρίβεια για σταθερό $\varepsilon = 0.001$ Βλέπουμε ότι όσο πιο μικρό l δηλαδή όσο μεγαλύτερη ακρίβεια απαιτείται στην εύρεση του ελαχίστου, τόσο αυξάνονται και οι επαναλήψεις του αλγόριθμου μέχρι τον τερματισμό του, κάτι απολύτως λογικό.

Θέμα 2

Λίγα λόγια για τον αλγόριθμο

Σε αυτό το θέμα θα υλοποιήσουμε στο matlab τη μέθοδο του χρυσού τομέα την οποία θα εφαρμόσουμε στις συναρτήσεις και θα σχολιάσουμε τα αποτελέσματα.

Η μέθοδος του χρυσού τομέα βασίζεται και αυτή στη λογική ότι έχουμε ένα αρχικό διάστημα [a,b] και προσπαθούμε να το περιορίζουμε(συνεχώς μέχρι να τερματιστεί ο αλγόριθμος) και στο τελικό διάστημα(όπως και σε όλα τα προηγούμενα διαστήματα) να περιέχεται το ελάχιστο της συνάρτησης με ακρίβεια l(απόσταση των άκρων a, b)
Η βασική ιδιότητα που συνδέει το νέο υποδιάστημα με το προηγούμενο είναι $b_{k+1} - a_{k+1} = \gamma(b_k - a_k)$ όπου γ ίσο με 0.618 και a_k , b_k τα άκρα του διαστήματος

Περιγραφή αλγόριθμου

Ο αλγόριθμος του χρυσού τομέα δεδομένου του τελικού εύρους $l > 0$ για αρχικό διάστημα $[a_1, b_1]$ αρχικοποιεί τα σημεία

$$\boxed{x_{11} = a_1 + (1-\gamma)(b_1 - a_1)} \quad \boxed{x_{21} = a_1 + \gamma(b_1 - a_1)}$$

με γ = 0.618 σε ένα βρόγχο (while) με συνθήκη τερματισμού $\boxed{b_k - a_k < l}$ επιλέγει τα σημεία

αν $\boxed{f(x_{1k}) > f(x_{2k}) \Rightarrow a_{k+1} = x_{1k} \ \ \kappa\alpha\iota \ \ b_{k+1} = b_k}$

τα σημεία ανανεώνονται σε: $\boxed{x_{2k+1} = a_{k+1} + \gamma(b_{k+1} - a_{k+1}) \ \ \kappa\alpha\iota \ \ x_{1k+1} = x_{2k}}$

αν $\boxed{f(x_{1k}) < f(x_{2k}) \Rightarrow a_{k+1} = a_k \ \ \kappa\alpha\iota \ \ b_{k+1} = x_{2k}}$

τα σημεία ανανεώνονται σε: $\boxed{x_{1k+1} = a_{k+1} + (1-\gamma)(b_{k+1} - a_{k+1}) \ \ \kappa\alpha\iota \ \ x_{2k+1} = x_{1k}}$

Υπολογισμοί της αντικειμενικής συνάρτησης

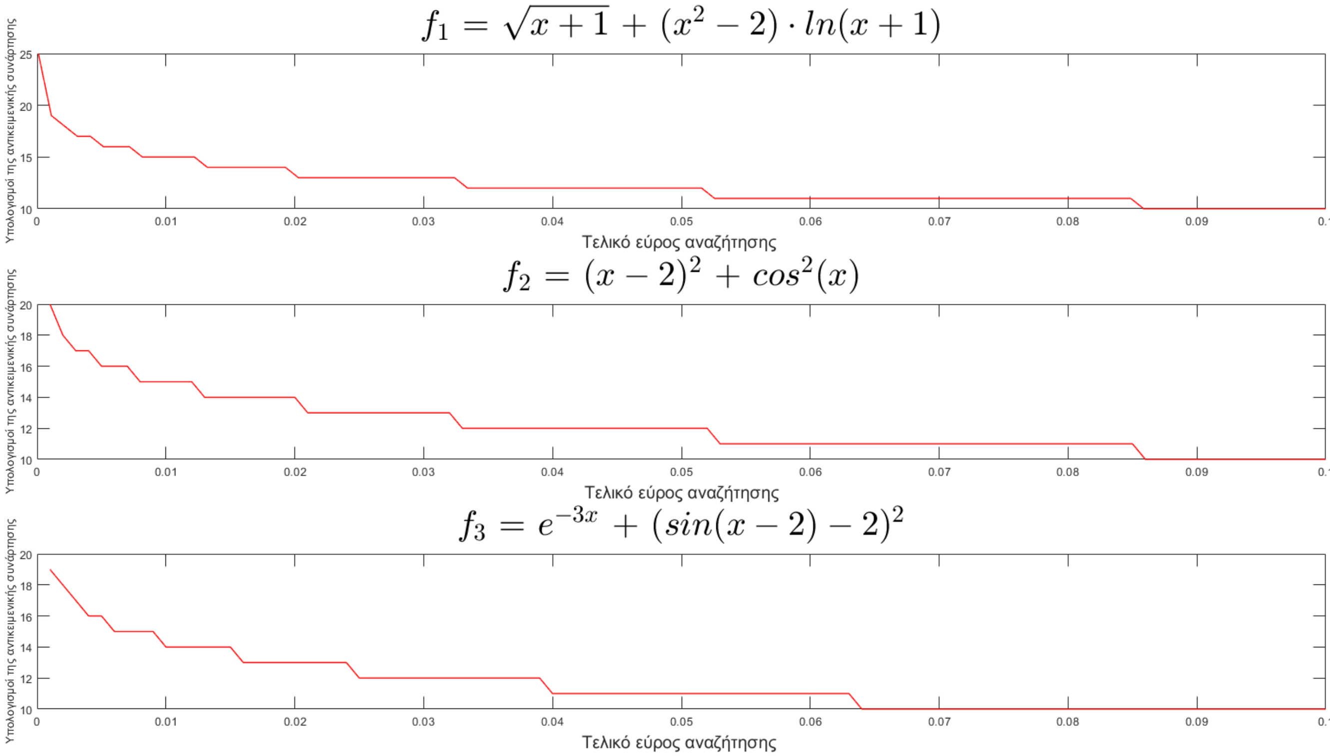
Οπως είδαμε στην παραπάνω περιγραφή στην αρχή του αλγορίθμου υπολογίζουμε αρχικά τα σημεία x_{11} και x_{21} αρα και κατ επέκταση τα $f(x_{11})$ και $f(x_{21})$ δηλαδή έχουμε δύο υπολογισμούς της αντικειμενικής συνάρτησης για την πρώτη επανάληψη ενώ για τις επόμενες επαναλήψεις προστίθεται μόνο ένας υπολογισμός της αντικειμενικής συνάρτησης ανάλογα με την περίπτωση $f(x_{1k}) < f(x_{2k})$ ή $f(x_{1k}) > f(x_{2k})$.
Καταλήγουμε οτι οι υπολογισμοί της αντικειμενικής συνάρτησης συνάρτηση του k συνδέονται απο την σχέση:

$$\boxed{calc(k) = k + 1}$$

Για k=1 έχουμε έχουμε δύο υπολογισμούς της αντικειμενικής συνάρτησης αφού υπολογίζουμε τα $f(x_{11})$ και $f(x_{21})$. Όμοια για k = 2 έχουμε τρείς υπολογισμούς της αντικειμενικής συνάρτησης αφού σε κάθε επόμενη επαναληψη μετά της πρώτης έχουμε εναν επιπλέον υπολογισμό.

Μεταβολή τελικού ευρους αναζήτησης

Παρακάτω μελετάμε τη μεταβολή των υπολογισμών της αντικειμενικής συνάρτησης $f_i(x)$ καθώς μεταβάλλουμε το τελικό εύρος αναζήτησης l.

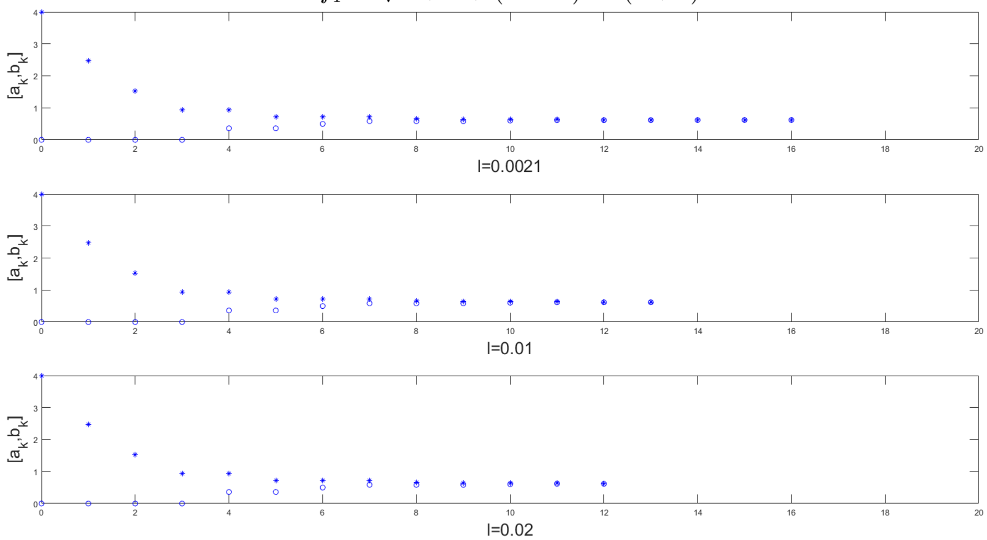


Βλέπουμε ότι όσο πιο μικρό l δηλαδή όσο μεγαλύτερη ακρίβεια απαιτείται στην εύρεση του ελαχίστου, τόσο αυξάνονται και οι επαναλήψεις του αλγόριθμου μέχρι τον τερματισμό του, κάτι απολύτως λογικό.

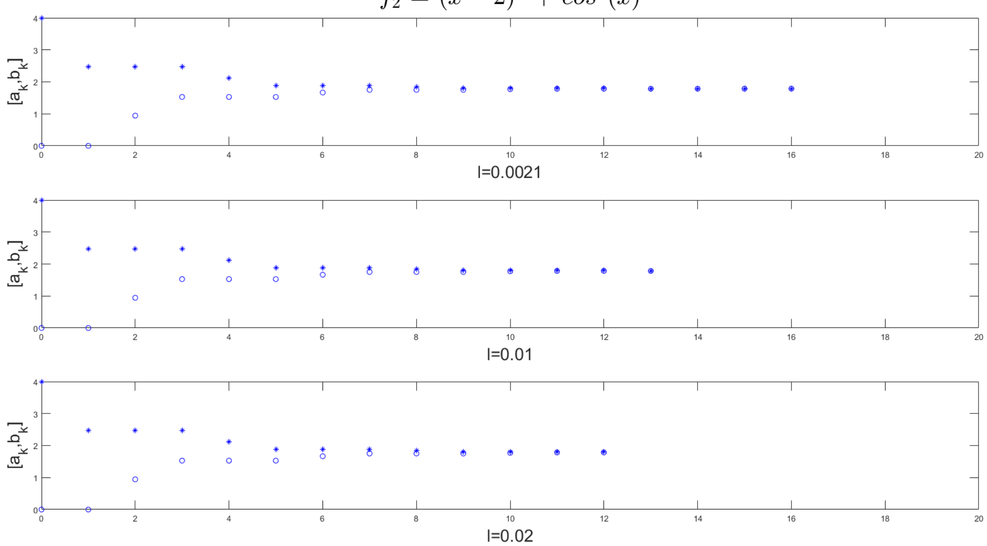
Μεταβολή των υποδιαστημάτων $[a_k, b_k]$

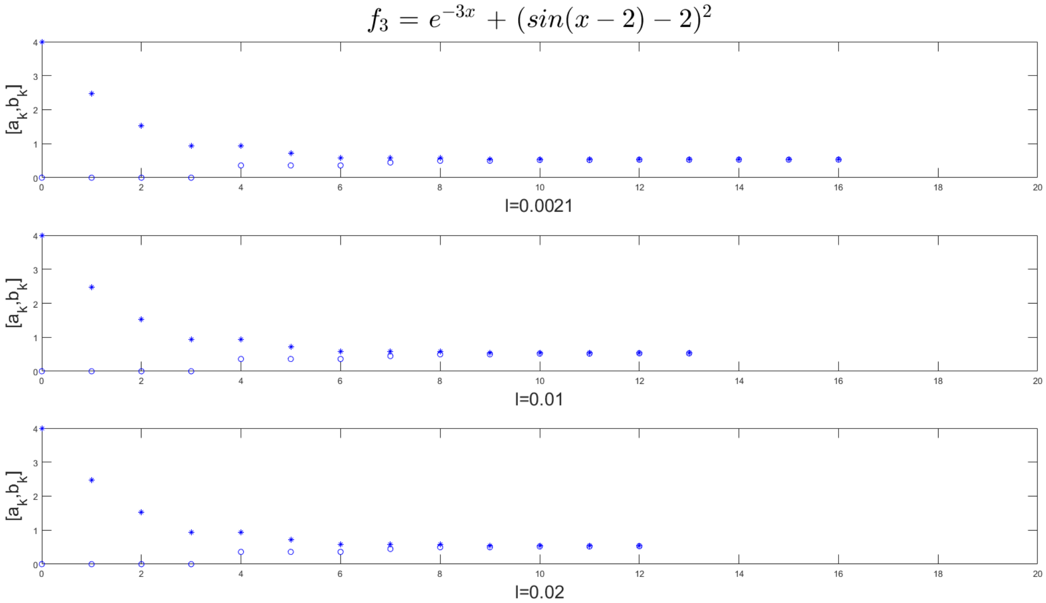
Τέλος παρουσιάζονται τα διαστήματα $[\alpha_k, b_k]$ ανα επανάληψη k για διάφορα l

$$f_1 = \sqrt{x+1} + (x^2 - 2) \cdot ln(x+1)$$



$$f_2 = (x-2)^2 + cos^2(x)$$





όπως φαίνεται απο τα διαγράμματα ο αλγόριθμος συγκλίνει στο ελάχιστο πιο γρήγορα οσο το l μεγαλώνει ωστόσο με μικρότερη ακρίβεια αφού το εύρος που υπάρχει το ελάχιστο μεγαλώνει

Θέμα 3

Λίγα λόγια για τον αλγόριθμο

Σε αυτό το θέμα θα υλοποιήσουμε στο matlab τη μέθοδο του fibonacci την οποία θα εφαρμόσουμε στις συναρτήσεις και θα σχολιάσουμε τα αποτελέσματα. Η μέθοδος fibonacci μοιάζει αρκετά με τη μέθοδο του χρυσού τομέα, βέβαια στη μέθοδο fibonacci το νέο διάστημα δεν μειώνεται σε σχέση με το προηγούμενο διάστημα με τη βοήθεια της σταθεράς γ αλλά με τη βοήθεια της ακολουθίας fibonacci. Ο συντελεστής αυτός δεν είναι σταθερός αλλά αλλάζει σε κάθε επανάληψη.
Σε αντίθεση με την σύμβαση του βιβλίου θα πάρουμε την ακουλουθία

$$F_n = \left\{ \begin{array}{ll} 0, & n = 0 \\ 1, & n = 1 \\ F_{n-1} + F_{n-2}, & n > 1 \end{array} \right\}$$

Περιγραφή αλγόριθμου

Δεδομένων των a_1 και b_1 για k=1 αρχικά υπολογίζουμε την ακολουθία Fibonacci έτσι ώστε $F_n > \frac{b_1 - a_1}{l}$ και στην συνέχεια τα αρχικά σημεία
$\boxed{x_{11} = a_1 + \frac{F_{n-k-1}}{F_{n-k+1}}(b_k - a_k)}$ και $\boxed{x_{21} = a_1 + \frac{F_{n-k}}{F_{n-k+1}}(b_k - a_k)}$
Στην συνέχεια σε έναν βρόγχο επανάληψης για n-1 φορές:

$$A\nu \;\; f(x1k) < f(x2k) \;\; \tau o \tau \varepsilon \;\; a_{k+1} = a_k \;\; \kappa \alpha \iota \;\; b_{k+1} = x_{2k}$$

$$x_{1k+1} = a_{k+1} + \frac{F_{n-k-2}}{F_{n-k}}(b_{k+1} - a_{k+1}) \;\; \kappa \alpha \iota \;\; x_{2k+1} = x_{1k}$$

$$A\nu \;\; f(x1k) > f(x2k) \;\; \tau o \tau \varepsilon \;\; a_{k+1} = x_{1k} \;\; \kappa \alpha \iota \;\; b_{k+1} = b_k$$

$$x_{2k+1} = a_{k+1} + \frac{F_{n-k-1}}{F_{n-k}}(b_{k+1} - a_{k+1}) \;\; \kappa \alpha \iota \;\; x_{1k+1} = x_{2k}$$

Υπολογισμοί της αντικειμενικής συνάρτησης

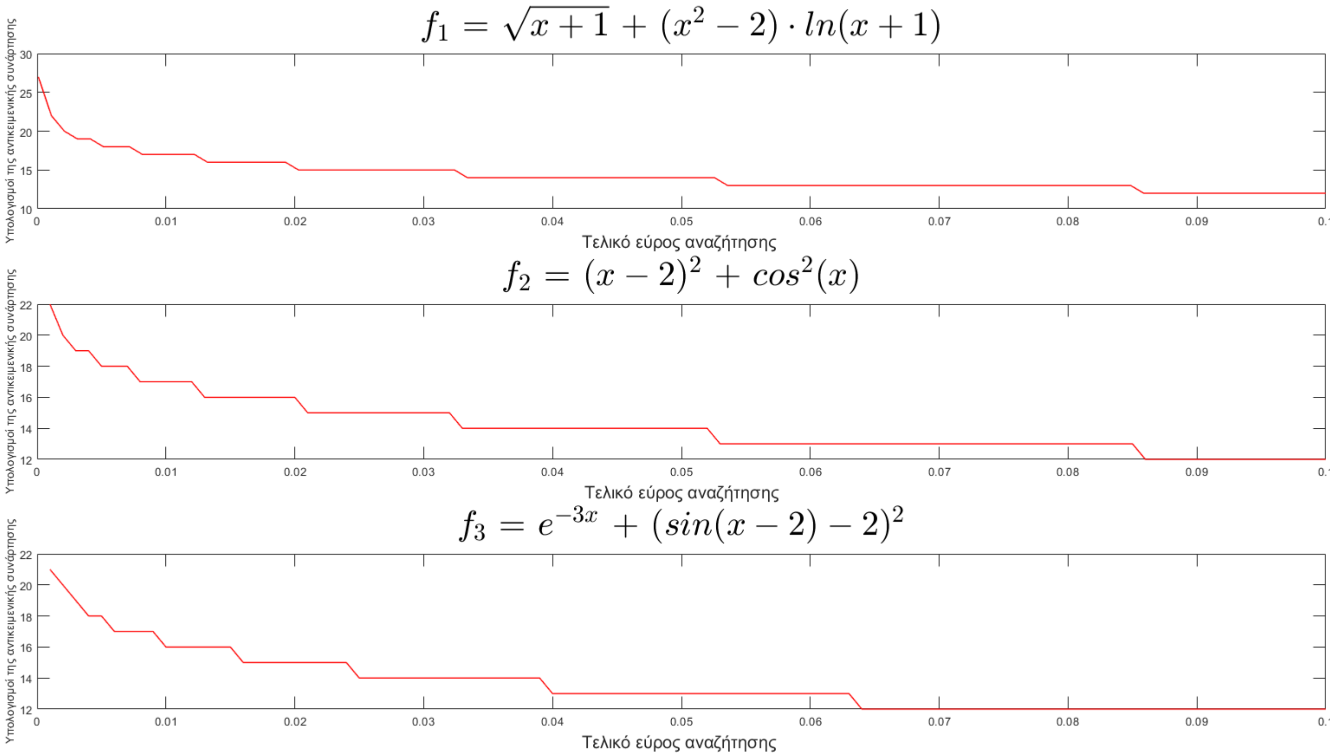
Στην πρώτη επανάληψη του αλγορίθμου έχουμε δύο υπολογισμούς της αντικειμενικής συνάρτησης $f(x_{11})$ και $f(x_{21})$ ενώ για κάθε επόμενη επανάληψη έχουμε ένα επιπλέον υπολογισμό. Ωστόσο επειδή ο αλγόριθμος τρέχει για n-1 επαναλήψεις καταλήγουμε οτι οι υπολογισμοί της αντικειμενικής συνάρτησης συνάρτηση του k συνδέονται απο την σχέση:

$$calc(1) = 2 \;\; \gamma \iota \alpha \;\; k = 1$$

$$\boxed{calc(k) = k} \;\; \gamma \iota \alpha \;\; k > 1$$

Μεταβολή τελικού ευρους αναζήτησης

Παρακάτω μελετάμε τη μεταβολή των υπολογισμών της αντικειμενικής συνάρτησης $f_i(x)$ καθώς μεταβάλλουμε το τελικό εύρος αναζήτησης l.

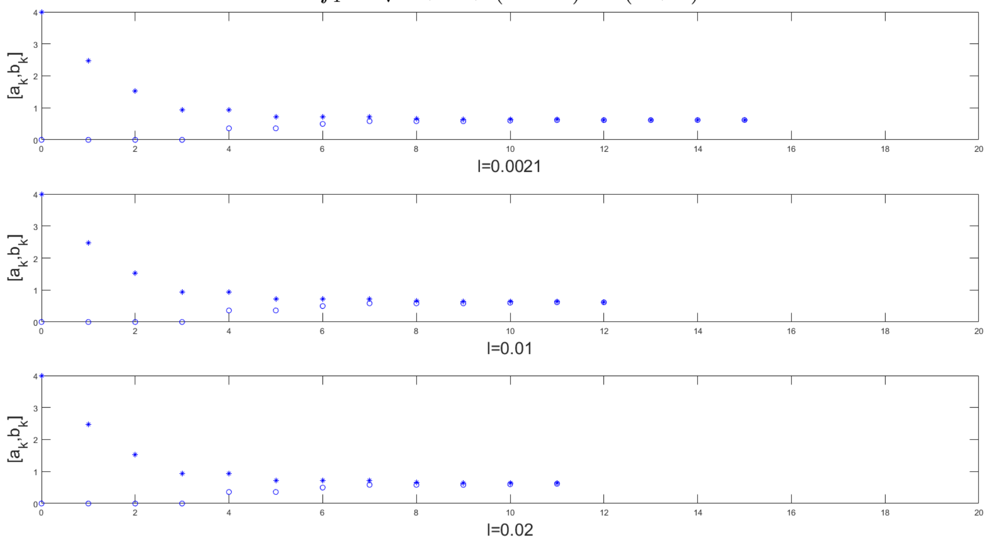


Βλέπουμε ότι όσο πιο μικρό l δηλαδή όσο μεγαλύτερη ακρίβεια απαιτείται στην εύρεση του ελαχίστου, τόσο αυξάνονται και οι επαναλήψεις του αλγόριθμου μέχρι τον τερματισμό του, κάτι απολύτως λογικό.

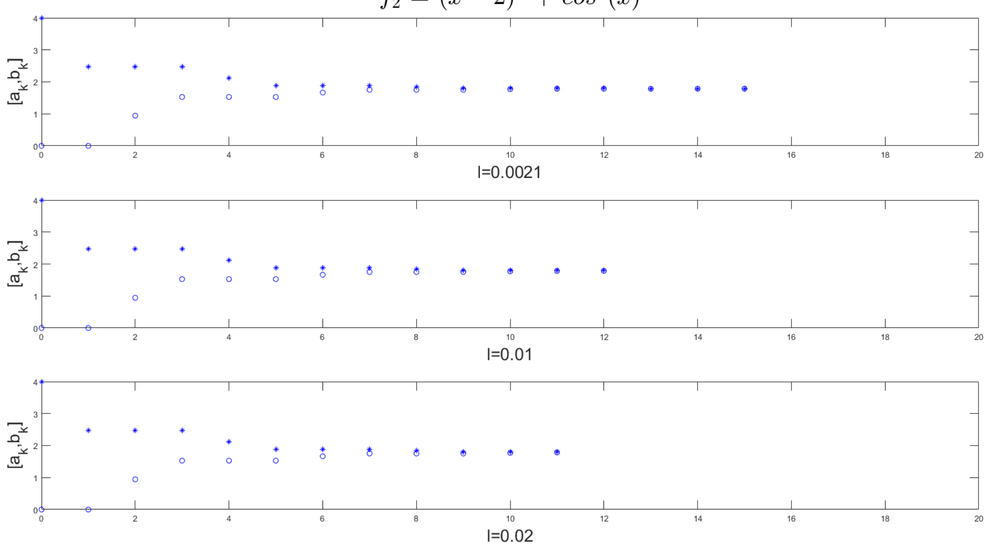
Μεταβολή των υποδιαστημάτων $[a_k, b_k]$

Τέλος παρουσιάζονται τα διαστήματα $[\alpha_k, b_k]$ ανα επανάληψη k για διάφορα l

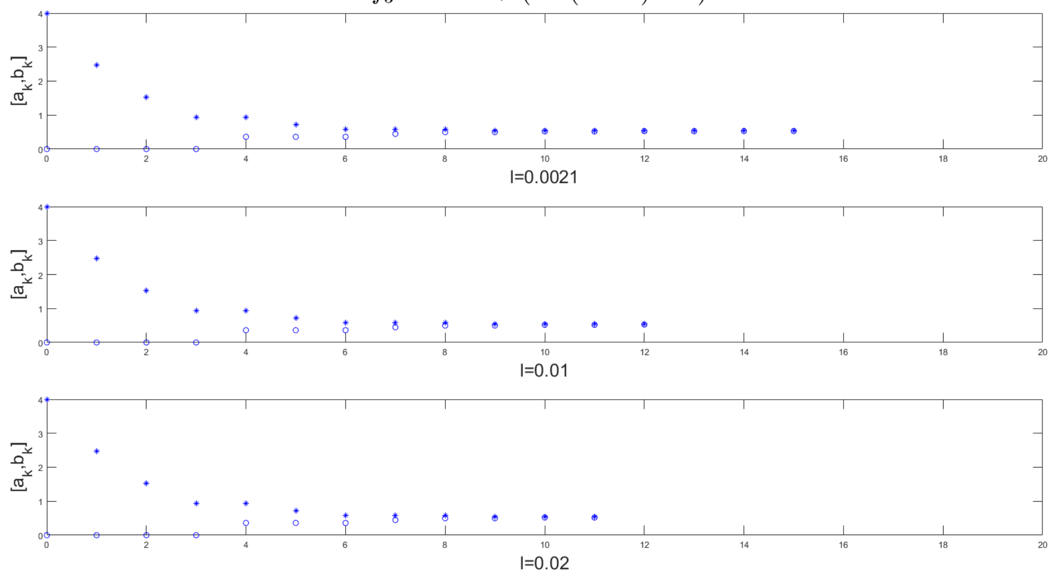
$$f_1 = \sqrt{x+1} + (x^2-2) \cdot ln(x+1)$$



$$f_2 = (x-2)^2 + cos^2(x)$$



$$f_3 = e^{-3x} + (sin(x-2)-2)^2$$



όπως φαίνεται απο τα διαγράμματα ο αλγόριθμος συγκλίνει στο ελάχιστο πιο γρήγορα οσο το l μεγαλώνει ωστόσο με μικρότερη ακρίβεια αφού το εύρος που υπάρχει το ελάχιστο μεγαλώνει

Θέμα 4

Λίγα λόγια για τον αλγόριθμο

Σε αυτό το θέμα θα υλοποιήσουμε στο matlab τη μέθοδο του μέθοδο της διχοτόμου με χρήση παραγώγων την οποία θα εφαρμόσουμε στις συναρτήσεις και θα σχολιάσουμε τα αποτελέσματα.
Επιλέγω x_k=$(a_k$+$b_k)/2$, όπου a_k, b_k τα άκρα του αρχικού διαστήματος και εξετάζω το πρόσημο της παραγώγου στο x_k. Αν είναι ίση με το μηδέν είμαι στο ελάχιστο, κάτι απολύτως λογικό γιατί στη μέθοδο θεωρήσαμε κυρτές(σχεδόν) συναρτήσεις. Αν η παράγωγος είναι μεγαλύτερη του μηδενός τότε το ελάχιστο βρίσκεται αριστερά του x_k και παίρνω σαν νέο διάστημα $[a_k$,$x_k]$ ενώ αν είναι αρνητική το ελάχιστο βρίσκεται δεξιά το x_k και παίρνω σαν νέο διάστημα το $[x_k$,$b_k]$.

Περιγραφή αλγόριθμου

$$A\nu\ \ \frac{df(x)}{dx} = 0\ \ \gamma\iota\alpha\ \ x = x_k\ \ \tau o\tau\varepsilon\ \ x_k\ \ \varepsilon\iota\nu\alpha\iota\ \ \varepsilon\lambda\alpha\chi\iota\sigma\tau o$$

$$A\nu\ \ \frac{df(x)}{dx} > 0\ \ \gamma\iota\alpha\ \ x = x_k\ \ \tau o\tau\varepsilon\ \ [a_{k+1}, b_{k+1}] = [a_k, x_k]$$

$$A\nu\ \ \frac{df(x)}{dx} < 0\ \ \gamma\iota\alpha\ \ x = x_k\ \ \tau o\tau\varepsilon\ \ [a_{k+1}, b_{k+1}] = [x_k, b_k]$$

Υπολογισμοί της αντικειμενικής συνάρτησης

Για k=1 έχουμε τον υπολογισμό $\frac{df(x_1)}{dx}$ $\gamma\iota\alpha\ x = x_1$ και ο αλγοριθμός ξεκινάει με έναν υπολογισμό αντικειμενικής συνάρτησης. Για κάθε επόμενη επανάληψη προστίθεται πάλι αλλος ένας υπολογισμός αντικειμενικής συνάρτησης συνεπώς καταλήγουμε οτι οι υπολογισμοί της αντικειμενικής συνάρτησης συνάρτηση του k συνδέονται απο την σχέση:

$$\boxed{calc(k) = k}$$

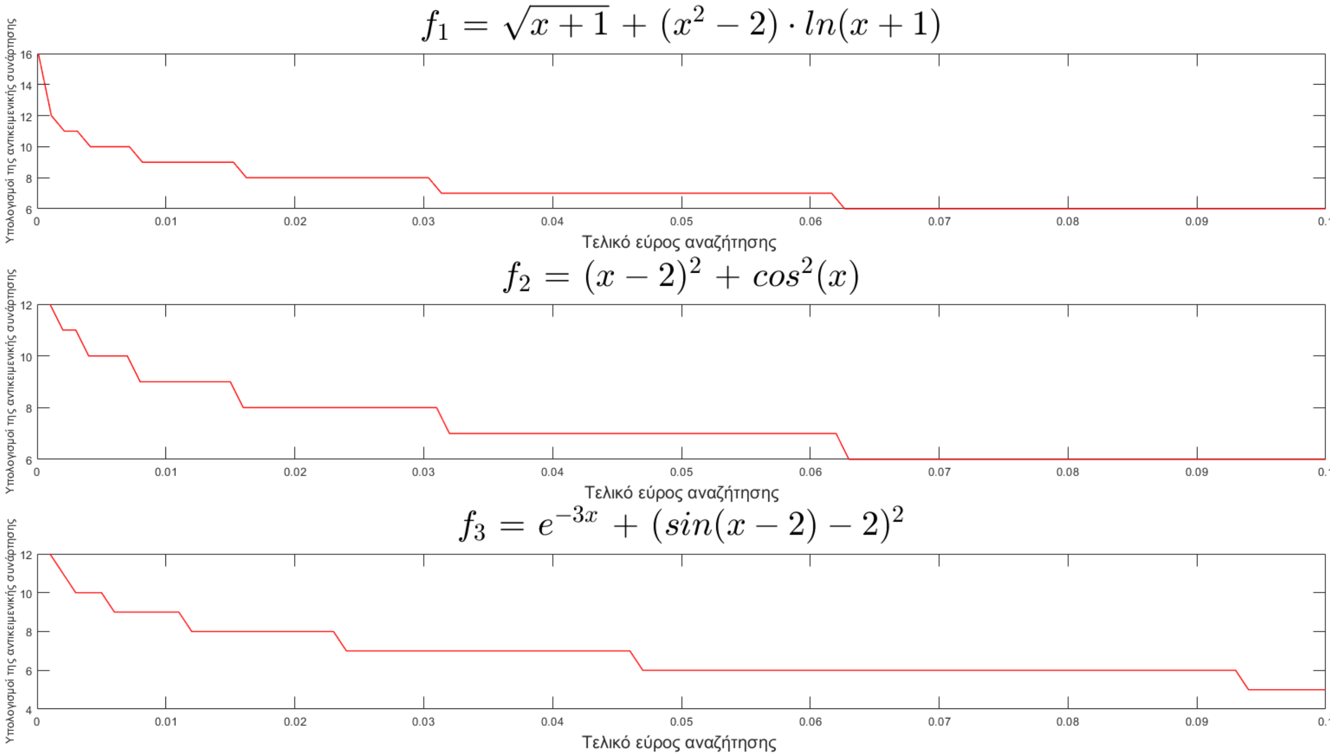
Να σημειωθεί πως ο αλγόριθμος έχει τον περιορισμό των επαναλήψεων

$$(\frac{1}{2})^n \leq \frac{l}{b_1 - a_1}$$

συνεπώς ο αλγόριθμος τερματίζει εως k = n υπολογισμούς

Μεταβολή τελικού ευρους αναζήτησης

Παρακάτω μελετάμε τη μεταβολή των υπολογισμών της αντικειμενικής συνάρτησης $f_i(x)$ καθώς μεταβάλλουμε το τελικό εύρος αναζήτησης l.

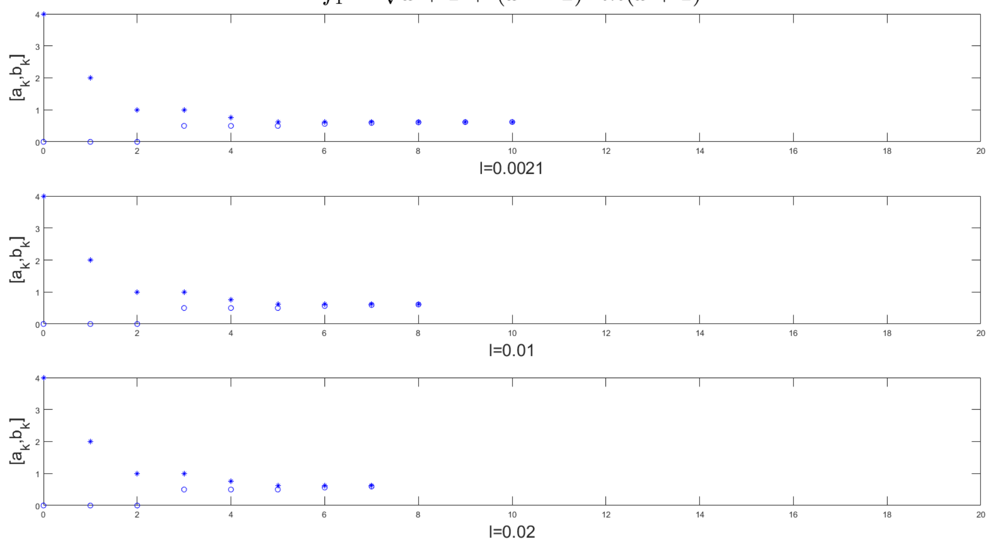


Βλέπουμε ότι όσο πιο μικρό l δηλαδή όσο μεγαλύτερη ακρίβεια απαιτείται στην εύρεση του ελαχίστου, τόσο αυξάνονται και οι επαναλήψεις του αλγόριθμου μέχρι τον τερματισμό του, κάτι απολύτως λογικό.

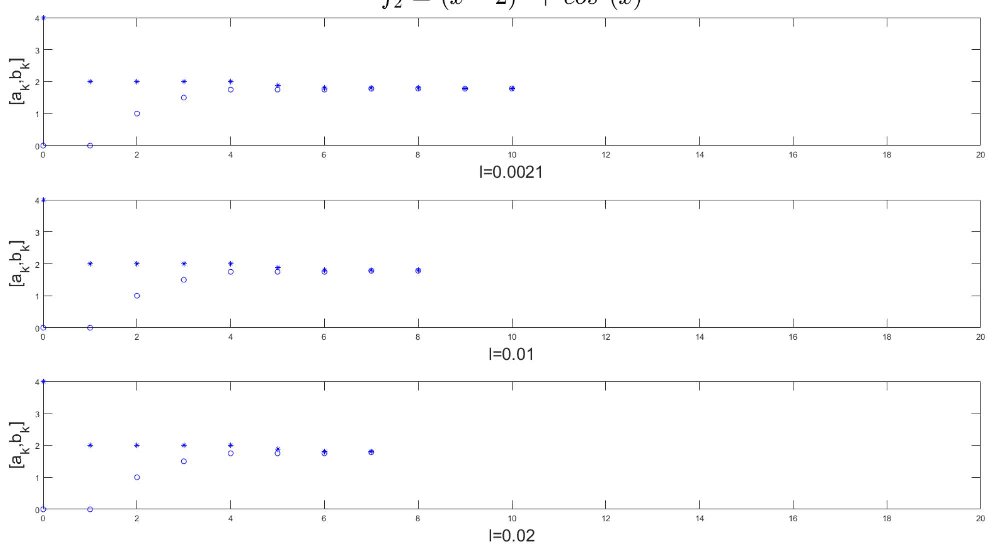
Μεταβολή των υποδιαστημάτων $[a_k, b_k]$

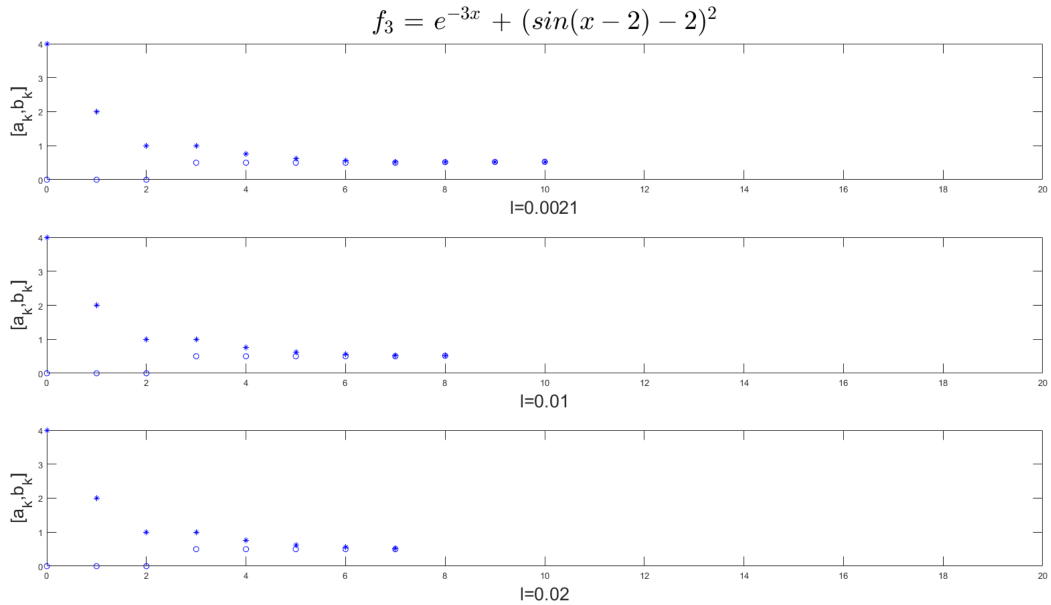
Τέλος παρουσιάζονται τα διαστήματα $[\alpha_k, b_k]$ ανα επανάληψη k για διάφορα l

$$f_1 = \sqrt{x+1} + (x^2 - 2) \cdot ln(x+1)$$



$$f_2 = (x-2)^2 + cos^2(x)$$





όπως φαίνεται απο τα διαγράμματα ο αλγόριθμος συγκλίνει στο ελάχιστο πιο γρήγορα οσο το l μεγαλώνει ωστόσο με μικρότερη ακρίβεια αφού το εύρος που υπάρχει το ελάχιστο μεγαλώνει

Σχολιασμός

Algorithm	l=0.0021	l=0.01	l=0.02
Διχοτόμησης	30	16	14
Χρυσής Τομής	17	14	13
Fibonacci	15	12	11
Παραγώγισης	10	8	7

Η μέθοδος διχοτόμου με χρήση παραγώγων είναι η περισσότερο αποτελεσματική,εν συνεχεία ακολουθεί η μέθοδος fibonacci, αμέσως μετά η μέθοδος της χρυσής τομής και τέλος η λιγότερο αποτελεσματική, μέθοδο της διχοτόμου. Τα αποτελέσματα αυτά συμφωνούν με το βιβλίο Τεχνικές Βελτιστοποίησης.