

Εργασία 4: Βελτιστοποίηση χωρίς περιορισμούς

Τα περισσότερα μαθηματικά και πρακτικά προβλήματα μπορούν να διατυπωθούν σαν προβλήματα βελτιστοποίησης συναρτήσεων με ή χωρίς περιορισμούς. Υπάρχουν αρκετοί μέθοδοι για την λύση αυτών των προβλημάτων. Στην εργασία αυτή μελετάμε βασικούς αλγόριθμους για βελτιστοποίηση συναρτήσεων χωρίς περιορισμούς.

1. Βελτιστοποίηση συναρτήσεων μιας μεταβλητής

Οι αλγόριθμοι για την κατηγορία αυτών των προβλημάτων κατηγοριοποιούνται ως εξής:

1. **Μέθοδοι δευτέρας παραγώγου:** Μέθοδος Newton
2. **Μέθοδοι πρώτης παραγώγου:** ημί- μέθοδοι Newton
3. **Μέθοδοι χωρίς παραγώγους :** χρυσής τομής

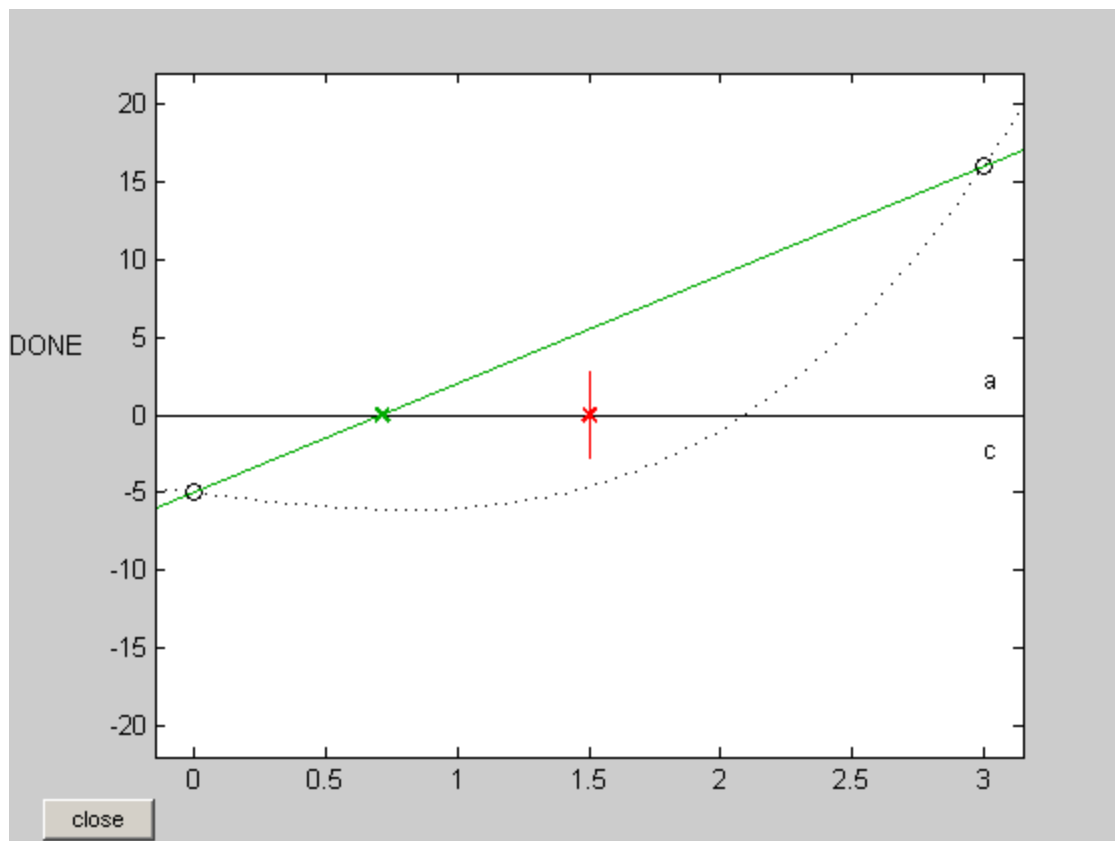
Υπάρχουν κατηγορίες μεθόδων που συνδιάζουν τα χαρακτηριστικά δυο οι περισσότερων μεθόδων. Η MATLAB `fzero()` συνάρτηση συνδιάζει μεθόδους των κατηγοριών 2 και 3 .

Άσκηση 1: `fzero` ντέμο

Η MATLAB συνάρτηση `fzero()` συνδιάζει 3 μεθόδους εύρεσης ριζών: διχοτόμησης, τέμνουσας και αντίστροφη παρεμβολή με δευτέρου βαθμού πολυώνυμα. Η `fzerogui.m` συνάρτηση μας παρέχει μια αλληλεπιδραστική δυνατότητα να οπτικοποιήσουμε την διαδικασία προσέγγισης ριζών:

- διαβάστε τα σχόλια της `fzerogui()` συνάρτησης και εκτελέστε την για τα παραδείγματα συναρτήσεων που περιλαμβάνει,
- πως λύνουμε προβλήματα βελτιστοποίησης χρησιμοποιώντας την συνάρτηση αυτή - δώστε 2 παραδείγματα.

ΒΑΛΤΕ ΤΟΝ ΚΩΔΙΚΑ ΣΑΣ ΕΔΩ:



Οι μέθοδοι που δεν χρησιμοποιούν παραγώγους – δηλαδή το ψαχούλεμα για το ελάχιστο της συνάρτησης βασίζεται μόνον στις τιμές της συνάρτησης. Η MATLAB συνάρτηση `fminbnd()` χρησιμοποιεί συνδιασμό της χρυσής τομής και παραβολική παρεμβολή για την εύρεση του ελαχίστου.

Άσκηση 2: MATLAB `fminbnd`

Εφαρμόστε `fminbnd()` για να προσεγγίσετε το ελάχιστο των συναρτήσεων :

$$f(x) = x^4 - 14x^3 + 60x^2 - 70x,$$

$$g(x) = \frac{1}{2}x^2 - \sin x,$$

$$h(x) = x^2 + 4 \cos x.$$

ΒΑΛΤΕ ΤΟΝ ΚΩΔΙΚΑ ΣΑΣ ΕΔΩ:

2. Βελτιστοποίηση συναρτήσεων με πολλές μεταβλητές

Στην περίπτωση αυτή υπάρχει παρόμοια κατηγοροποίηση των μεθόδων:

1. **Μέθοδοι δευτέρας παραγώγου:** Μέθοδος Newton
2. **Μέθοδοι πρώτης παραγώγου:** απότομης κατάβασης (steepest descent) , συζυγών κλίσεων (conjugate gradient), ημί- μέθοδοι Newton
3. **Μέθοδοι χωρίς παραγώγους :** simplex method

Η καθαρά εφαρμογή της μεθόδου Newton δεν συνιστάται, ιδιαίτερα όταν η συνάρτηση για μεγιστοποίηση έχει πολλές μεταβλητές – η δεύτερη παράγωγος είναι ο πίνακας Έσσιαν (*Hessian*). Η μέθοδος που συνήθως χρησιμοποιείται είναι της **πρώτης παραγώγου** όπως απότομης κατάβασης (steepest descent) , συζυγών κλίσεων (conjugate gradient), ημί- μέθοδοι Newton . Η επιλογή της μεθόδου εξαρτάται από την συμπεριφορά των συναρτήσεων που πρόκειται να ελαχιστοποιήσουμε.

Όπως και στη μία διάσταση, οι μέθοδοι χωρίς παραγώγους δεν εξαρτώνται από την ομαλότητα των παραγώγων της συνάρτησης. Η τιμή που πληρώνουμε για το πλεονέκτημα των μεθόδων αυτών είναι ο υπολογισμός ενός μεγάλου αριθμού τιμών της συνάρτησης.

Άσκηση 3: MATLAB `congrad()`, `nlnewton()`

Θεωρείστε το πρόβλημα ελαχιστοποίησης της συνάρτησης "μπανάνα συνάρτηση" με 2 μεταβλητές:

$$100(y - x^2)^2 + (a - x)^2,$$

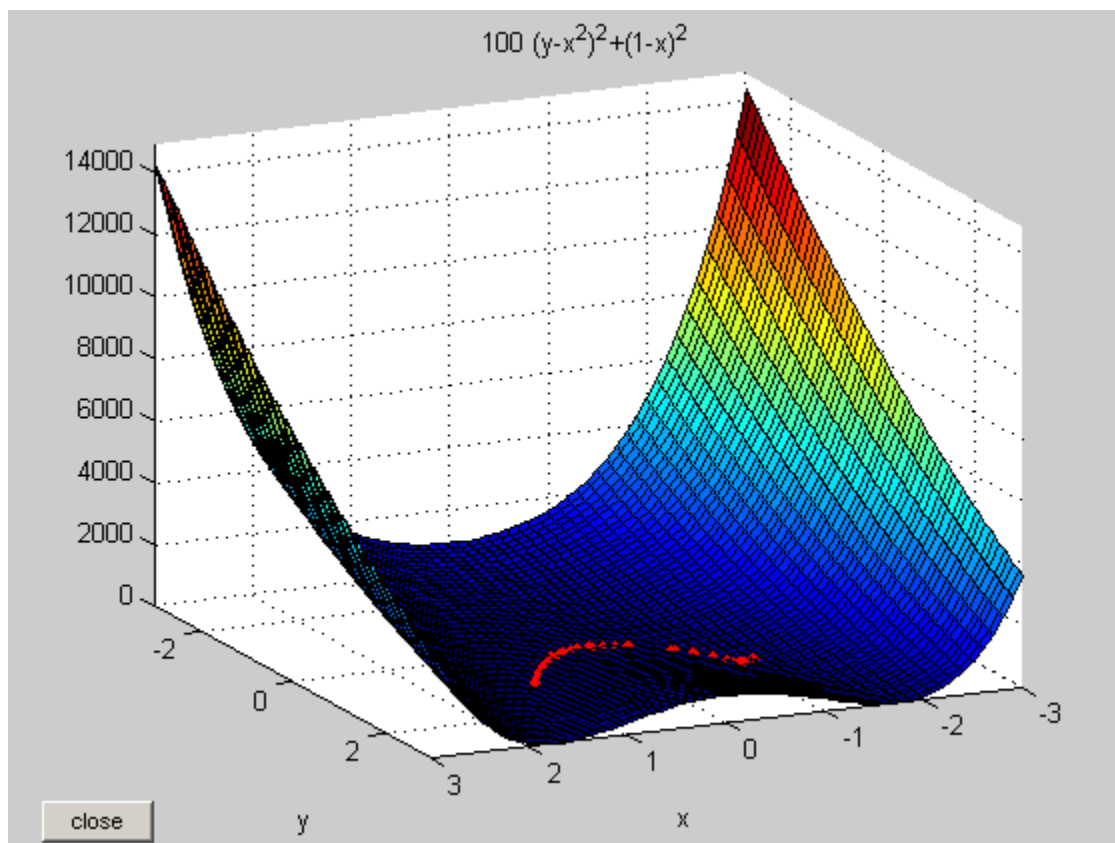
που έχει ελάχιστο στο σημείο $[a, a^2]$.

- τρέξτε τον κώδικα `congrad()` και `nlnewton()` για να βρείτε το ελάχιστο και κάντε την γραφική παράσταση της συνάρτησης χρησιμοποιώντας τον παρακάτω κώδικα

```
fbanana = @(x) 100*(x(2)-x(1)^2)^2+(1-x(1))^2; banana = @(x,y) 100*(y-x.^2).^2+(1-x).^2; ezsurf(banana, [-3 3]); view(155, 20); hold off
```

- Πόσες επαναλήψεις η `conjugate gradient`, `congrad()`, και μέθοδοι `nlnewton()`, απαιτούν?

ΒΑΛΤΕ ΤΟΝ ΚΩΔΙΚΑ ΕΔΩ:



Άσκηση 4:

- Τρέξτε τους παρακάτω κώδικες και εξηγήστε τα αποτελέσματα



newton_method.m



gradient_method_2.m



gradient_method_search.m



gradient_search.m



linear_search_demo.m