

Βελτιστοποίηση Περιβαλλοντικών Συστημάτων

Επίλυση προβλημάτων Γενετικών Αλγορίθμων

Ε. Τσέκερη, Δ. Κολοκοτσά

Απρίλιος 2021

1 Ασκήσεις γενετικών αλγορίθμων

1.1 Άσκηση 1

Να υπολογιστεί το μήκος της συμβολοσειράς για δυαδική κωδικοποίηση του κάθε χρωμοσώματος πληθυσμού για τη βελτιστοποίηση της συνάρτησης $f(x_1, x_2) = 3x_1^3 x_2 + 6x_2^2$ με χρήση γενετικών αλγορίθμων. Τα πεδία ορισμού των μεταβλητών είναι $x_1 \in [3, 6]$ και $x_2 \in [4, 10]$ και απαιτείται ακρίβεια δύο δεκαδικών ψηφίων και για τις δυο μεταβλητές.

1.2 Άσκηση 2

Σε ένα πρόβλημα μεγιστοποίησης μιας συνάρτησης $f(x)$, με $x \in [-1, 2]$, απαιτείται ακρίβεια 6 δεκαδικών ψηφίων για τη μεταβλητή x . Ζητείται η κωδικοποίηση, σε δυαδικό σύστημα, της μεταβλητής x . Στη συνέχεια να γίνει η αποκωδικοποίηση ενός τυχαίου χρωμοσώματος του αρχικού πληθυσμού.

Δίνονται:

1. Σχέση μετατροπής από το δυαδικό στο δεκαδικό σύστημα:

$$(b_{21}b_{20} \dots b_0)_2 = \left(\sum_{i=0}^{m-1} b_i 2^i \right)_2$$

όπου b_i τα δυαδικά ψηφία (bits) που αναπαριστούν τη μεταβλητή στο δυαδικό σύστημα.

2. Σχέση αναγωγής του δεκαδικού αριθμού που προκύπτει από τη δυαδική αναπαράσταση στο πραγματικό πεδίο ορισμού της μεταβλητής.

$$x_i = a_i + x_i' \frac{(b_i - a_i)}{(2^{m_i} - 1)}$$

όπου $x_i \in [a_i, b_i]$ και m_i ο αριθμός των δυαδικών ψηφίων που χρησιμοποιήθηκε για τη δυαδική αναπαράσταση της μεταβλητής x_i .

1.3 Άσκηση 3

Έστω 10 χρωμοσώματα με αντίστοιχες τιμές καταλληλότητας: 0.10, 1.75, 95.35, 0.04, 1.95, 1.82, 0.001, 33.65, 0.55 και 0.12. Ζητείται να παραχθούν 6 γονείς, μέσω επιλογής με τη μέθοδο της ρουλέτας.

Δίνονται:

Για να εφαρμοστεί η μέθοδος της ρουλέτας θα πρέπει οι παραπάνω τιμές καταλληλότητας να μπου σε αύξουσα σειρά.

1.4 Άσκηση 4

Εφαρμόζεται γενετικός αλγόριθμος για τη μεγιστοποίηση της συνάρτησης καταλληλότητας:

$$f(x_1, x_2) = 3x_1^2 x_2^2 - 6x_1 x_2 + 2x_1^4 x_2$$

με $x_1 \in [0, 10]$ και $x_2 \in [0, 20]$. Ως αρχικός πληθυσμός Π1 έχουν επιλεχθεί τα παρακάτω χρωμοσώματα όπου στο κάθε χρωμόσωμα οι μεταβλητές x_1 και x_2 αναπαρίστανται με 5 δυαδικά ψηφία (τα πρώτα 5 για τη x_1 και τα υπόλοιπα 5 για τη x_2).

0	0	1	0	0	1	0	1	1	0
0	1	0	1	0	0	1	0	1	1
0	1	0	1	1	0	0	1	0	0
1	0	0	0	0	0	1	0	0	1

(α) Να υπολογιστεί η απόδοση του πληθυσμού Π1

(β) Θεωρήστε ότι ο πληθυσμός δεξαμενής ζευγαρώματος είναι το 100% του Π1 και ότι τα ζεύγη είναι 1-4 και 2-3. Να εφαρμοστεί ο τελεστής διασταύρωσης στη θέση 6 για το 1^ο ζεύγος και στη θέση 4 για το 2^ο ζεύγος. Στη συνέχεια να πραγματοποιηθεί μετάλλαξη στις θέσεις 30 και 31 του πληθυσμού. Να δοθεί η απόδοση του του πληθυσμού Π2 που προκύπτει μετά τη διασταύρωση και μετάλλαξη.

1.5 Άσκηση 5

Εφαρμόζεται γενετικός αλγόριθμος για τη μεγιστοποίηση της συνάρτησης καταλληλότητας:

$$f(x_1, x_2) = 2x_1^2 - 2x_1x_2 + 5x_2^2$$

με $x_1 \in [-5, 10]$ και $x_2 \in [-2, 6]$. Ως αρχικός πληθυσμός Π1 έχουν επιλεχθεί τα παρακάτω χρωμοσώματα όπου στο κάθε χρωμόσωμα οι μεταβλητές x_1 και x_2 αναπαρίστανται με 5 δυαδικά ψηφία (τα πρώτα 5 για τη x_1 και τα υπόλοιπα 5 για τη x_2).

1	0	1	0	0	1	0	1	1	0
0	1	1	0	0	1	0	1	0	0
1	0	1	0	1	1	1	1	0	1
0	0	1	0	0	1	1	0	0	1
1	0	0	0	1	1	0	1	1	1

(α) Να υπολογιστεί η απόδοση του πληθυσμού Π1

(β) Να εφαρμοστεί η μέθοδος της ρουλέτας για την επιλογή υποψήφιων γονέων. Θεωρούμε ότι η πιθανότητα διασταύρωσης του πληθυσμού είναι 80%. Για την εφαρμογή της μεθόδου να χρησιμοποιηθούν οι τυχαίοι αριθμοί:

$$r_1=0,82$$

$$r_2=0,12$$

$$r_3=0,50$$

$$r_4=0,2$$

(γ) Ποιος είναι ο πληθυσμός της δεξαμενής ζευγαρώματος; Θεωρώντας ότι θα διασταυρωθούν τα άτομα της δεξαμενής ζευγαρώματος με τη σειρά με την οποία προκύπτουν, να εφαρμοστεί ο τελεστής διασταύρωσης ενός σημείου στη θέση 3 για το 1^ο ζεύγος και στη θέση 7 για το 2^ο ζεύγος.

Στη συνέχεια να πραγματοποιηθεί μετάλλαξη στις θέσεις 39 και 42 του πληθυσμού.

Να δοθεί ο πληθυσμός που προκύπτει μετά τη διασταύρωση και μετάλλαξη

1.6 ΕΦΑΡΜΟΓΗ

Ζητείται η μεγιστοποίηση της συνάρτησης 2 μεταβλητών:

$$f(x_1, x_2) = 21.5 + x_1 \sin(4\pi x_1) + x_2 \sin(20\pi x_2)$$

$$\text{με } -3.0 \leq x_1 \leq 12.1$$

$$\text{και } 4.1 \leq x_2 \leq 5.8$$

Η απαιτούμενη ακρίβεια είναι 4 δεκαδικά ψηφία για κάθε μεταβλητή. Ξεκινώντας από έναν τυχαίο αρχικό πληθυσμό, να κατασκευαστεί η 1^η γενιά. Να θεωρήσετε μέγεθος πληθυσμού pop_size=20 και πιθανότητες διασταύρωσης $p_c=0.25$ και μετάλλαξης $p_m=0.01$.

ΕΠΙΛΥΣΗ ΑΣΚΗΣΗΣ 4

Τα βήματα ενός ΓΑ είναι:

1. Αρχικοποίηση πληθυσμού
2. Αποκωδικοποίηση χρωμοσωμάτων
3. Αξιολόγηση πληθυσμού
4. Αναπαραγωγή

➤ Επιλογή γονέων

- Διασταύρωση
- Μετάλλαξη

Με το πέρας όλων των παραπάνω διεργασιών, καταλήγουμε στη δημιουργία της επόμενης γενιάς.

5. Επανάληψη από το βήμα (2) μέχρι την ικανοποίηση των κριτηρίων τερματισμού

ΑΡΧΙΚΟΠΟΙΗΣΗ ΠΛΗΘΥΣΜΟΥ – ΑΠΟΚΩΔΙΚΟΠΟΙΗΣΗ ΧΡΩΜΟΣΩΜΑΤΩΝ – ΑΞΙΟΛΟΓΗΣΗ ΠΛΗΘΥΣΜΟΥ

Το εύρος τιμών της μεταβλητής x_1 έχει μήκος 15.1, άρα το διάστημα $[-3.0, 12.1]$ θα πρέπει να διαχωριστεί σε $15.1 \cdot 10000 = 151000$ ίσα υποδιαστήματα. Αυτό σημαίνει ότι $m_1=18$ δυαδικά ψηφία απαιτούνται για τη δυαδική αναπαράσταση της x_1 (και τα οποία θα αποτελούν το πρώτο τμήμα του χρωμοσώματος), αφού:

$$131072 = 2^{17} < 151000 \leq 2^{18} = 262144.$$

Η ίδια διαδικασία για τη x_2 προσδιορίζει το $m_1=15$ ως τον απαιτούμενο αριθμό δυαδικών ψηφίων για την αναπαράσταση των δεκαδικών τιμών της. Επομένως, το συνολικό μήκος του χρωμοσώματος είναι:

$$m = \sum_{i=1}^k m_i = 18 + 15 = 33 \text{ δυαδικά ψηφία, όπου } k \text{ ο αριθμός των μεταβλητών (εδώ } k=2).$$

Έστω, για παράδειγμα, το άτομο:

(010001001011010000111110010100010).

Τα πρώτα 18 δυαδικά ψηφία (010001001011010000) αναπαριστούν τον αριθμό:

$$x_1 = -3.0 + 70352 \frac{15.1}{262143} = -3.0 + 4.0524 = 1.0524$$

και τα επόμενα 15 δυαδικά ψηφία (11110010100010) τον αριθμό $x_2 = 5.7553$ (υπενθυμίζουμε ότι ζητάμε ακρίβεια 4 δεκαδικών ψηφίων για τις μεταβλητές). Δηλαδή, το άτομο αυτό αντιστοιχεί στο σημείο:

$$(x_1, x_2) = (1.0524, 5.7553).$$

Η απόδοση για αυτό το χρωμόσωμα είναι $f(1.0524, 5.7553) = 20.2526$.

Το παραπάνω ήταν ένα παράδειγμα κωδικοποίησης, αρχικοποίησης, αποκωδικοποίησης και υπολογισμού της απόδοσης ενός ατόμου. Με όμοιο τρόπο ακολουθείται η ίδια διαδικασία για όλα τα χρωμοσώματα.

Ας υποθέσουμε ότι μετά τη διαδικασία της αρχικοποίησης (δηλαδή την επιλογή, με τυχαίο τρόπο, του αρχικού πληθυσμού) προκύπτει ο ακόλουθος πληθυσμός:

ΑΡΧΙΚΟΣ ΠΛΗΘΥΣΜΟΣ

$v1 = (100110100000001111111010011011111)$

$v2 = (111000100100110111001010100011010)$

v3 = (000010000011001000001010111011101)
v4 = (100011000101101001111000001110010)
v5 = (000111011001010011010111111000101)
v6 = (000101000010010101001010111111011)
v7 = (001000100000110101111011011111011)
v8 = (100001100001110100010110101100111)
v9 = (010000000101100010110000001111100)
v10 = (000001111000110000011010000111011)
v11 = (011001111110110101100001101111000)
v12 = (110100010111101101000101010000000)
v13 = (111011111010001000110000001000110)
v14 = (010010011000001010100111100101001)
v15 = (111011101101110000100011111011110)
v16 = (110011110000011111100001101001011)
v17 = (011010111111001111010001101111101)
v18 = (011101000000001110100111110101101)
v19 = (000101010011111111110000110001100)
v20 = (101110010110011110011000101111110)

Για να κατασκευάσουμε τη ρουλέτα με την οποία θα επιλεχθούν οι υποψήφιοι γονείς, θα πρέπει να γνωρίζουμε την απόδοσή κάθε χρωμοσώματος. Έτσι, αρχικά αποκωδικοποιούμε κάθε χρωμόσωμα και υπολογίζουμε την απόδοσή του. Έχουμε:

ΑΞΙΟΛΟΓΗΣΗ ΠΛΗΘΥΣΜΟΥ

eval(v1) = f(6.0845, 5.6522) = 26.019600
eval(v2) = f(10.3484, 4.3803) = 7.580015
eval(v3) = f(-2.5166, 4.3904) = 19.526329
eval(v4) = f(5.2786, 5.5935) = 17.406725

```

eval(v5) = f(-1.2552, 4.7345) = 25.341160
eval(v6) = f(-1.8117, 4.3919) = 18.100417
eval(v7) = f(-0.9915, 5.6803) = 16.020812
eval(v8) = f(4.9106, 4.7030) = 17.959701
eval(v9) = f(0.7954, 5.3815) = 16.127799
eval(v10) = f(-2.5549, 4.7937) = 21.278435
eval(v11) = f(3.1301, 4.9961) = 23.410669
eval(v12) = f(9.3562, 4.2395) = 15.011619
eval(v13) = f(11.1346, 5.3787) = 27.316702
eval(v14) = f(1.3359, 5.1514) = 19.876294
eval(v15) = f(11.0890, 5.0545) = 30.060205
eval(v16) = f(9.2116, 4.9938) = 23.867227
eval(v17) = f(3.3675, 4.5713) = 13.696165
eval(v18) = f(3.8430, 5.1582) = 15.414128
eval(v19) = f(-1.7466, 5.3956) = 20.095903
eval(v20) = f(7.9360, 4.7573) = 13.666916

```

Είναι εμφανές ότι το άτομο v_{15} είναι αυτό με την καλύτερη απόδοση και το άτομο v_2 αυτό με τη χειρότερη απόδοση στον αρχικό πληθυσμό.

ΚΑΤΑΣΚΕΥΗ ΡΟΥΛΕΤΑΣ – ΕΠΙΛΟΓΗ ΥΠΟΨΗΦΙΩΝ ΓΟΝΕΩΝ

Στην συνέχεια κατασκευάζουμε μια ρουλέτα (roulette wheel). Τα βήματα κατασκευής της συγκεκριμένης ρουλέτας είναι:

1. Υπολογίζουμε την απόδοση $eval(v_i)$ για κάθε χρωμόσωμα (έχει υπολογιστεί).
2. Υπολογίζουμε τη συνολική απόδοση του πληθυσμού:

$$F = \sum_{i=1}^{pop_size} eval(v_i) = \text{άθροισμα όλων των επιμέρους αποδόσεων}$$

3. Υπολογίζουμε την πιθανότητα επιλογής p_i για κάθε χρωμόσωμα

$$p_i = \frac{\text{eval}(v_i)}{F}$$

4. Τέλος, υπολογίζουμε τη συσσωρευμένη (cumulative) πιθανότητα q_i για κάθε χρωμόσωμα:

$$q_i = \sum_{j=1}^i p_j = \text{άθροισμα } p_j, \text{ όπου } j = 1, \dots, i$$

Η συνολική απόδοση (fitness) του πληθυσμού είναι:

$$F = \sum_{i=1}^{20} \text{eval}(v_i) = 387.776822$$

Η πιθανότητα επιλογής p_i κάθε μέλους το πληθυσμού v_i , $i=1,\dots,20$, είναι:

p1 = eval(v1)/F = 0.067099	p11 = eval(v11)/F = 0.060372
p2 = eval(v2)/F = 0.019547	p12 = eval(v12)/F = 0.038712
p3 = eval(v3)/F = 0.050355	p13 = eval(v13)/F = 0.070444
p4 = eval(v4)/F = 0.044889	p14 = eval(v14)/F = 0.051257
p5 = eval(v5)/F = 0.065350	p15 = eval(v15)/F = 0.077519
p6 = eval(v6)/F = 0.046677	p16 = eval(v16)/F = 0.061549
p7 = eval(v7)/F = 0.041315	p17 = eval(v17)/F = 0.035320
p8 = eval(v8)/F = 0.046315	p18 = eval(v18)/F = 0.039750
p9 = eval(v9)/F = 0.041590	p19 = eval(v19)/F = 0.051823
p10 = eval(v10)/F = 0.054873	p20 = eval(v20)/F = 0.035244

όπως αναφέρθηκε, οι συσσωρευμένες πιθανότητες (cumulative probabilities) q_i για κάθε άτομο

v_i , $i=1,\dots,20$ του πληθυσμού, υπολογίζονται από την:

$$q_i = \sum_{j=1}^i p_j$$

Για παράδειγμα:

Για $i=1 \rightarrow q_1=p_1$

Για $i=2 \rightarrow q_1=p_1+p_2$

...

Για $i=20 \rightarrow q_{20}=p_1+p_2+\dots+p_{20}$

Προκύπτουν έτσι οι παρακάτω τιμές:

q1 = 0.067099	q6 = 0.293917	q11 = 0.538381	q16 = 0.837863
q2 = 0.086647	q7 = 0.335232	q12 = 0.577093	q17 = 0.873182
q3 = 0.137001	q8 = 0.381546	q13 = 0.647537	q18 = 0.912932
q4 = 0.181890	q9 = 0.423137	q14 = 0.698794	q19 = 0.964756
q5 = 0.247240	q10 = 0.478009	q15 = 0.776314	q20 = 1.000000

Τώρα είμαστε έτοιμοι να περιστρέψουμε τη ρουλέτα $\text{pop_size}=20$ φορές. Σε κάθε περιστροφή επιλέγουμε από τον αρχικό πληθυσμό και ένα άτομο για τον πληθυσμό της δεξαμενής ζευγαρώματος. Πιο συγκεκριμένα, ακολουθούμε τα εξής βήματα:

1.Επιλέγουμε τυχαία έναν αριθμό r μεταξύ 0 και 1.

2.Αν $r < q_i$, τότε επιλέγουμε για τον πληθυσμό της δεξαμενής ζευγαρώματος το πρώτο χρωμόσωμα v_i , διαφορετικά επιλέγουμε το v_i για το οποίο ισχύει:

$$q_{i-1} < r \leq q_i, \quad \text{με } 2 \leq i \leq \text{pop_size}$$

Τα παραπάνω βήματα θα επαναληφθούν $\text{pop_size} = 20$ φορές. Παρακάτω γίνεται η εφαρμογή τους.

Υποθέτουμε ότι έχουμε παράγει την εξής ακολουθία $\text{pop_size}=20$ τυχαίων αριθμών στο διάστημα $[0, 1]$:

r₁ = 0.513870	r₂ = 0.175741	r₃ = 0.308652	r₄ = 0.534534
r₅ = 0.947628	r₆ = 0.171736	r₇ = 0.702231	r₈ = 0.226431
r₉ = 0.494773	r₁₀ = 0.424720	r₁₁ = 0.703899	r₁₂ = 0.389647
r₁₃ = 0.277226	r₁₄ = 0.368071	r₁₅ = 0.983437	r₁₆ = 0.005398
r₁₇ = 0.765682	r₁₈ = 0.646473	r₁₉ = 0.767139	r₂₀ = 0.780237

Ο πρώτος αριθμός $r_1=0.513870$ είναι μεγαλύτερος του q_{10} και μικρότερος του q_{11} , γεγονός που σημαίνει ότι το άτομο v_{11} επιλέγεται για να «περάσει» στον πληθυσμό της δεξαμενής ζευγαρώματος. Ο δεύτερος αριθμός $r_2 = 0.175741$ είναι μεγαλύτερος του q_3 και μικρότερος του q_4 , οπότε το άτομο v_4 επιλέγεται για το νέο πληθυσμό. Συνεχίζοντας με τον ίδιο τρόπο κατασκευάζουμε το νέο πληθυσμό:

ΔΕΞΑΜΕΝΗ ΖΕΥΓΑΡΩΜΑΤΟΣ

v1* = (011001111110110101100001101111000) (v11)
v2* = (100011000101101001111000001110010) (v4)
v3* = (001000100000110101111011011111011) (v7)
v4* = (011001111110110101100001101111000) (v11)
v5* = (00010101001111111110000110001100) (v19)
v6* = (100011000101101001111000001110010) (v4)
v7* = (111011101101110000100011111011110) (v15)
v8* = (000111011001010011010111111000101) (v5)
v9* = (011001111110110101100001101111000) (v11)
v10* = (000010000011001000001010111011101) (v3)
v11* = (111011101101110000100011111011110) (v15)
v12* = (010000000101100010110000001111100) (v9)
v13* = (00010100001001010100101011111011) (v6)
v14* = (100001100001110100010110101100111) (v8)
v15* = (101110010110011110011000101111110) (v20)
v16* = (100110100000001111111010011011111) (v1)
v17* = (000001111000110000011010000111011) (v10)
v18* = (111011111010001000110000001000110) (v13)
v19* = (111011101101110000100011111011110) (v15)
v20* = (110011110000011111100001101001011) (v16)

Ο παραπάνω πληθυσμός αποτελεί τη δεξαμενή ζευγαρώματος (πληθυσμός υποψήφιων γονέων). Στην τελευταία στήλη, μέσα σε παρέθεση, φαίνονται τα άτομα του αρχικού πληθυσμού που έχουν περάσει (αντιγραφεί) στη δεξαμενή ζευγαρώματος. Παρατηρούμε ότι ορισμένα άτομα έχουν αντιγραφεί περισσότερες από μία φορές, κάποια άλλα μόνο μία και μερικά δεν υπάρχουν καθόλου, ανάλογα με την απόδοσή τους.

ΕΠΙΛΟΓΗ ΓΟΝΕΩΝ - ΔΙΑΣΤΑΥΡΩΣΗ

Από τον παραπάνω πληθυσμό θα επιλεχθούν οι γονείς που θα συμμετέχουν τελικά στη διασταύρωση. Να σημειωθεί ότι ο τελεστής διασταύρωσης εφαρμόζεται σε κάποια (ή και όλα, ανάλογα με την τιμή της πιθανότητας διασταύρωσης p_c) μέλη του πληθυσμού. Ο θεωρητικά αναμενόμενος αριθμός γονέων ισούται με:

$$p_c \cdot \text{pop_size} = 0.25 \cdot 20 = 5 \text{ άτομα-γονείς.}$$

Ας δούμε πόσοι τελικά θα επιλεχθούν. Τα βήματα που ακολουθούνται είναι τα εξής:

1. Για κάθε χρωμόσωμα, επιλέγουμε τυχαία έναν αριθμό r μεταξύ 0 και 1.
2. Αν $r < p_c$, επιλέγουμε το χρωμόσωμα για διασταύρωση.

Επομένως, για κάθε άτομο στον παραπάνω πληθυσμό επιλέγουμε έναν τυχαίο αριθμό r από 0 ως 1. Επειδή $p_c = 0.25$, αν $r < 0.25$, επιλέγουμε το τρέχον άτομο για διασταύρωση. Διαφορετικά δεν το επιλέγουμε και εξετάζουμε την παραπάνω σχέση για το επόμενο άτομο.

Ας υποθέσουμε ότι η ακολουθία των τυχαίων αριθμών είναι:

$r_1 = 0.822951$	$r_2 = 0.151932$	$r_3 = 0.625477$	$r_4 = 0.314685$
$r_5 = 0.346901$	$r_6 = 0.911720$	$r_7 = 0.519760$	$r_8 = 0.401154$
$r_9 = 0.606758$	$r_{10} = 0.785402$	$r_{11} = 0.031523$	$r_{12} = 0.869921$
$r_{13} = 0.166525$	$r_{14} = 0.674520$	$r_{15} = 0.758400$	$r_{16} = 0.581893$
$r_{17} = 0.389248$	$r_{18} = 0.200232$	$r_{19} = 0.355635$	$r_{20} = 0.826927$

Αυτό σημαίνει ότι 4 άτομα, τα v_2 , v_{11} , v_{13} και v_{18} επιλέχθηκαν για διασταύρωση. Εάν κατά την παραπάνω διαδικασία προκύψουν περισσότεροι γονείς από τους αναμενόμενους, σύμφωνα με την τιμή του γινομένου ($\text{pop_size} \cdot p_c$), τότε συνήθως επιλέγουμε από αυτούς πλήθος ίσο με το αναμενόμενο. Μπορούμε όμως να επιλέξουμε να κρατήσουμε περισσότερα από τα αναμενόμενα άτομα-γονείς, εάν διαπιστώσουμε ότι αυτό θα βελτίωνε τα αποτελέσματά μας.

Ο αριθμός των επιλεχθέντων ατόμων στο παράδειγμά μας είναι ζυγός, οπότε το ταίριασμά τους είναι εύκολο. Στην περίπτωση που ο αριθμός τους ήταν μονός, θα έπρεπε είτε να επιλέξουμε ένα ακόμα άτομο από τον πληθυσμό ή να απορρίψουμε κάποιο από τα ήδη επιλεχθέντα.

Στη συνέχεια, ταιριάζουμε με τυχαίο τρόπο άτομα για ζευγάρωμα. Ας υποθέσουμε ότι ζευγαρώνουν τα πρώτα δύο (v_2 και v_{11}) και τα επόμενα δύο (v_{13} και v_{18}). Θα επιλέξουμε διαστύρωση 1 σημείου. Έστω ότι για το πρώτο ζεύγος επιλέχθηκε, με τυχαίο τρόπο, ως σημείο διασταύρωσης, η θέση $\text{pos} = 9$:

ΓΟΝΕΑΣ 1: $v_2 = (100011000 \mathbf{1} 101101001111000001110010)$

ΓΟΝΕΑΣ 2: $v_{11} = (111011101 \mathbf{1} 101110000100011111011110)$

Τότε αυτά θα αντικατασταθούν από τα άτομα–απογόνους:

ΑΠΟΓΟΝΟΣ 1: $v_2' = (100011000 \mid 101110000100011111011110)$

ΑΠΟΓΟΝΟΣ 2: $v_{11}' = (111011101 \mid 101101001111000001110010)$

Έστω ότι για το δεύτερο ζεύγος επιλέχθηκε, πάλι με τυχαίο τρόπο, ως σημείο διασταύρωσης, η θέση pos = 20:

ΓΟΝΕΑΣ 3: $v_{13} = (00010100001001010100 \mid 101011111011)$

ΓΟΝΕΑΣ 4: $v_{18} = (11101111101000100011 \mid 0000001000110)$

Τότε αυτά θα αντικατασταθούν από τα άτομα–απογόνους:

ΑΠΟΓΟΝΟΣ 3: $v_2' = (00010100001001010100 \mid 0000001000110)$

ΑΠΟΓΟΝΟΣ 4: $v_{11}' = (11101111101000100011 \mid 101011111011)$

Οπότε η τρέχουσα μορφή του πληθυσμού έχει ως εξής:

ΠΛΗΘΥΣΜΟΣ ΜΕΤΑ ΤΗ ΔΙΑΣΤΑΥΡΩΣΗ

$v_1 = (100110100000001111111010011011111)$

$v_2' = (100011000 \mid 101110000100011111011110)$

$v_3 = (000010000011001000001010111011101)$

$v_4 = (100011000101101001111000001110010)$

$v_5 = (000111011001010011010111111000101)$

$v_6 = (000101000010010101001010111111011)$

$v_7 = (001000100000110101111011011111011)$

$v_8 = (100001100001110100010110101100111)$

$v_9 = (010000000101100010110000001111100)$

```

v10 = (000001111000110000011010000111011)
v11' = (111011101 101101001111000001110010)
v12 = (110100010111101101000101010000000)
v13' = (00010100001001010100 0000001000110)
v14 = (010010011000001010100111100101001)
v15 = (111011101101110000100011111011110)
v16 = (110011110000011111100001101001011)
v17 = (011010111111001111010001101111101)
v18' = (11101111101000100011 010111111011)
v19 = (000101010011111111110000110001100)
v20 = (101110010110011110011000101111110)

```

Δηλαδή τα άτομα απόγονοι i αντικατέστησαν τα αντίστοιχα άτομα i στον αρχικό πληθυσμό (όχι στον πληθυσμό της δεξαμενής ζευγαρώματος).

ΜΕΤΑΛΛΑΞΗ

Η μετάλλαξη βλέπει το σύνολο των ψηφίων όλων των χρωμοσωμάτων σαν ένα συρμό από ψηφία, χωρίς να ξεχωρίζει χρωμοσώματα. Στο δυαδικό σύστημα, η μετάλλαξη έχει σαν αποτέλεσμα την αντιστροφή του ψηφίου (το 0 γίνεται 1 και το 1 γίνεται 0).

Στην περίπτωση μας, η πιθανότητα μετάλλαξης είναι $p_m = 0.01$, άρα αναμένουμε ότι το 1% περίπου όλων των δυαδικών ψηφίων του πληθυσμού θα αντιστραφούν. Ο πληθυσμός μας αποτελείται από:

$m \cdot pop_size = 33 \cdot 20 = 660$ δυαδικά ψηφία,

άρα αναμένουμε περίπου 6.6 μεταλλάξεις ανά γενιά. Ισχύει ό,τι αναφέρθηκε στη διασταύρωση, σε περίπτωση που από την παρακάτω διαδικασία προκύψει μεγαλύτερος αριθμός ψηφίων από τον αναμενόμενο.

Τα βήματα που ακολουθούνται είναι τα εξής:

1. Για κάθε ψηφίο επιλέγουμε τυχαία έναν αριθμό r μεταξύ 0 και 1.
2. Αν $r < p_m$, τότε αντιστρέφουμε το δυαδικό ψηφίο (το ψηφίο υφίσταται μετάλλαξη).

Συνεπώς, για κάθε δυαδικό ψηφίο πρέπει να παράγουμε ένα τυχαίο αριθμό από το διάστημα $[0, 1]$. Εάν $r < 0.01$, αντιστρέφουμε το δυαδικό ψηφίο. Έτσι, πρέπει να παράγουμε 660 τυχαίους αριθμούς. Για λόγους οικονομίας χώρου θα παρουσιάσουμε μόνο τα επιλεχθέντα προς μετάλλαξη γονίδια (δηλαδή ψηφία). Στον ακόλουθο πίνακα, φαίνονται, για μία τέτοια εκτέλεση, οι θέσεις στον πληθυσμό των δυαδικών ψηφίων που επιλέχθηκαν για μετάλλαξη, ο αριθμός του ατόμου που αντιστοιχούν, καθώς και η θέση τους στα αντίστοιχα άτομα:

α/α ΨΗΦΙΟΥ	ΑΡ. ΑΤΟΜΟΥ	ΘΕΣΗ ΨΗΦΙΟΥ
112	4	13
349	11	19
418	13	22
429	13	33
602	19	8

Δηλαδή, τέσσερα άτομα του πληθυσμού θα υποστούν μετάλλαξη. Ακολουθεί ο τελικός πληθυσμός για την τρέχουσα γενιά, όπου τα ψηφία που έχουν υποστεί μετάλλαξη φαίνονται πιο έντονα:

ΠΛΗΘΥΣΜΟΣ ΜΕΤΑ ΤΗ ΜΕΤΑΛΛΑΞΗ

```

v1 = (100110100000001111111010011011111)
v2' = (100011000101110000100011111011110)
v3 = (000010000011001000001010111011101)
v4'' = (100011000101001001111000001110010)
v5 = (000111011001010011010111111000101)
v6 = (00010100001001010100101011111011)
v7 = (00100010000011010111101101111011)
v8 = (100001100001110100010110101100111)
v9 = (010000000101100010110000001111100)
v10 = (000001111000110000011010000111011)
v11'' = (111011101101101001101000001110010)
v12 = (110100010111101101000101010000000)

```

v13'' = (00010100001001010100**1**00000100011**1**)

v14 = (010010011000001010100111100101001)

v15 = (111011101101110000100011111011110)

v16 = (110011110000011111100001101001011)

v17 = (011010111111001111010001101111101)

v18' = (11101111101000100011101011111011)

v19'' = (000101010011111111110000110001100)

v20 = (1011100**0**0110011110011000101111110)

Αυτή είναι η νέα γενιά και συγκεκριμένα η 1^η γενιά, η οποία προέκυψε από τον αρχικό πληθυσμό. Ακολουθώντας την ίδια επαναληπτική διαδικασία (αποκωδικοποίηση και αξιολόγηση των μεταβλητών, κατασκευή ρουλέτας επιλογής, εφαρμογή τελεστών διασταύρωσης και μετάλλαξης), κατασκευάζεται κάθε νέα γενιά από την προηγούμενη.

Κάθε φορά είναι ενδιαφέρον να ελέγχουμε την ολική απόδοση και τις επιμέρους αποδόσεις των ατόμων στο νέο πληθυσμό και να τις συγκρίνουμε με αυτές της προηγούμενης γενιάς.

Η διαδικασία τερματίζεται όταν ικανοποιηθεί το εκάστοτε κριτήριο τερματισμού, το οποίο μπορεί να είναι, για παράδειγμα, ένας συγκεκριμένος αριθμός γενεών, ένα συγκεκριμένο ποσοστό βελτίωσης του καλύτερου ατόμου ή της συνολικής απόδοσης του πληθυσμού, συνδυασμός των παραπάνω, κ.ο.κ.

Επειδή η καλύτερη γενιά δεν είναι αναγκαστικά η τελευταία, δηλαδή μπορεί κάποιο χρωμόσωμα προηγούμενης γενιάς να έχει καλύτερη απόδοση από όλα τα χρωμοσώματα της τελευταίας γενιάς, είναι σκόπιμο να αποθηκεύεται, σε κάθε επαναληπτική διαδικασία, δηλαδή σε κάθε γενιά, το «μέχρι τώρα καλύτερο άτομο», έτσι ώστε η βέλτιστη αυτή να μη χάνεται τελικά.

(Πηγή: Σημειώσεις «Γενετικοί Αλγόριθμοι και Εφαρμογές», Ελληνικό Ανοικτό Πανεπιστήμιο, Σπ. Λυκοθανάσης)

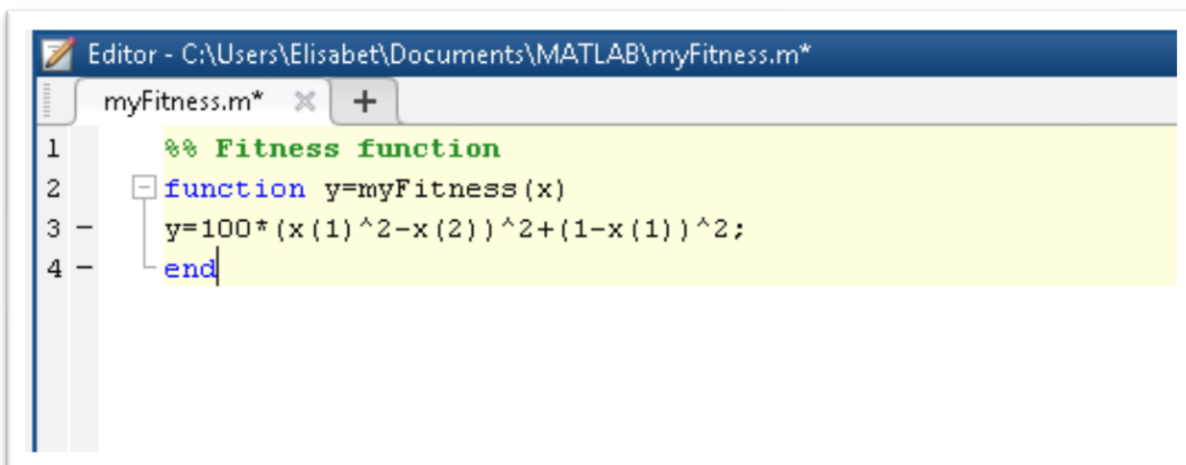
2 Παραδείγματα επίλυσης προβλημάτων με γενετικούς αλγόριθμους

2.1 Επίλυση απλού προβλήματος

Θέλουμε να ελαχιστοποιηθεί η παρακάτω αντικειμενική συνάρτηση:

$$f = 100x_1^2 - x_2^2 + (1 - x_1)^2$$

Ανοίγουμε τον editor στη MATLAB και η αντικειμενική συνάρτηση γράφεται ως:



```
Editor - C:\Users\Elisabet\Documents\MATLAB\myFitness.m*  
myFitness.m* x +  
1 %% Fitness function  
2 function y=myFitness(x)  
3 y=100*(x(1)^2-x(2))^2+(1-x(1))^2;  
4 end
```

Εικόνα 1. Δημιουργία αντικειμενικής συνάρτησης



```
Editor - C:\Users\Elisabet\Documents\MATLAB\maincode.m*  
myFitness.m x maincode.m* x +  
1 %%main function-minimization  
2 FitFcn=@myFitness;  
3 nvars=2;  
4 [x,fval]=ga(FitFcn,nvars);  
5  
6  
7  
8
```

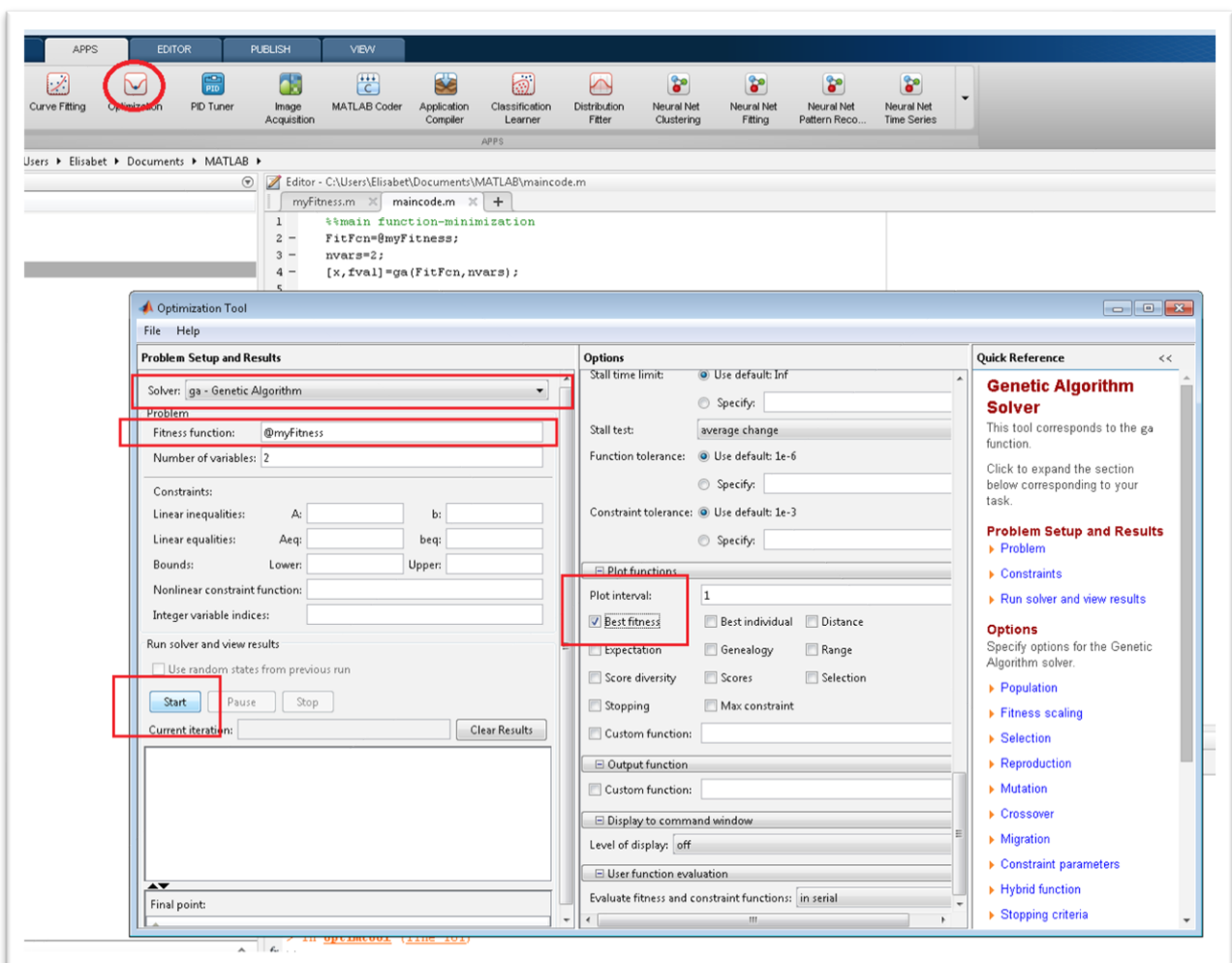
Εικόνα 2. Δημιουργία κύριου κώδικα

- Η αντικειμενική συνάρτηση στοm-fileκαλείται μέσω function handle(@myFitness) για τον υπολογισμό της τιμής της σε κάθε επανάληψη από τον ΓΑ.

- **nvars:** (number of variables) Είναι ο αριθμός των ανεξάρτητων μεταβλητών της αντικειμενικής συνάρτησης
- **x=ga (problem)** βρίσκει την τιμή που ελαχιστοποιεί το πρόβλημα (@myFitness)
- **[x,fval]=ga(FitFcn, nvars)** υπολογίζεται η fval η οποία είναι η τελική/βέλτιστη τιμή της αντικειμενικής συνάρτησης.
- Επιπλέον η χρήση της συνάρτησης **ga()** επιτρέπει την εξαγωγή:
 1. ενός πλήθους λύσεων σε μορφή πίνακα που περιλαμβάνει τις λύσεις της αντικειμενικής συνάρτησης
 2. της επίδοσης του τελικού πληθυσμού
- Γράφοντας στο command window την εντολή **maincode**, μας εμφανίζει στα δεξιά στο workspace τα αποτελέσματα του αλγορίθμου (x,fval)

2.2 Επίλυση του προβλήματος χρησιμοποιώντας Genetic Algorithm toolbox-MATLAB

Έχοντας δημιουργήσει την αντικειμενική συνάρτηση (Εικόνα 1), ανοίγουμε το toolbox GA, πατώντας το APPS=> Optimization όπως φαίνεται στην Εικόνα 3.



Εικόνα 3. GA Toolbox

2.3 Επίλυση προβλήματος με ΓΑ (κώδικας και εντολές στη MATLAB)

Να βρεθούν οι παράμετροι C_1 , C_2 ώστε να ικανοποιούνται τα παρακάτω κριτήρια:

$$y = C_1 + 0.5C_2 - 1 \quad (1)$$

$$C_1 + C_2 = 1 \quad (2)$$

$$2C_1 + C_2 = 2 \quad (3)$$

$$3C_1 + C_2 = 3 \quad (4)$$

LB= [-2,2], UB= [5,5]

Λύση

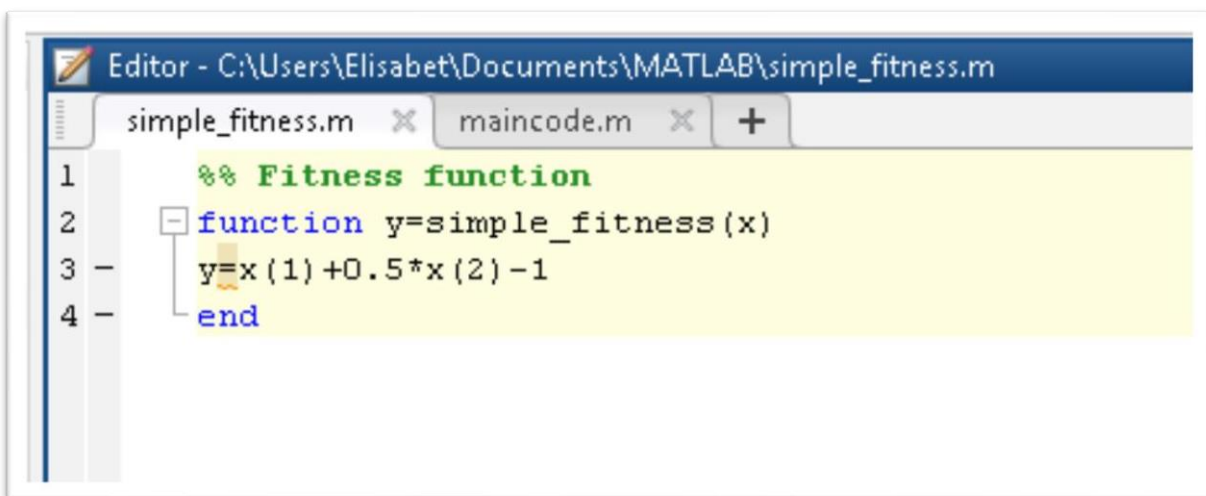
1. Θέλουμε οι περιορισμοί μας να είναι ίσοι με το 0, άρα οι (2),(3),(4) γράφονται:

$$C_1 + C_2 - 1 = 0$$

$$2C_1 + C_2 - 2 = 0$$

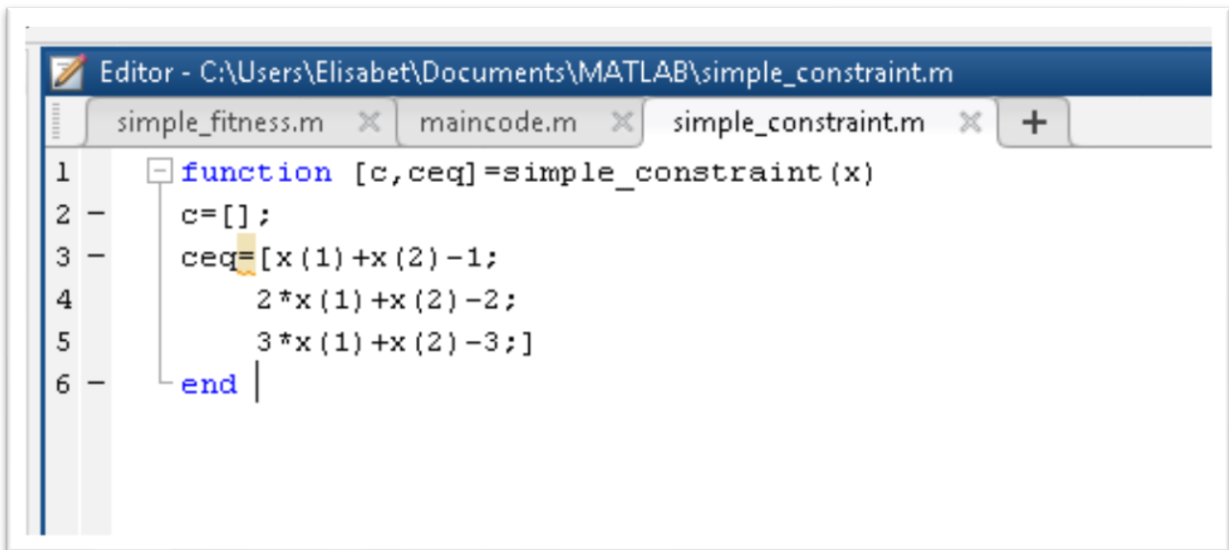
$$3C_1 + C_2 - 3 = 0$$

2. Δημιουργούμε τη fitness function στο MATLAB



```
Editor - C:\Users\Elisabet\Documents\MATLAB\simple_fitness.m
simple_fitness.m  x  maincode.m  x  +
1  %% Fitness function
2  function y=simple_fitness(x)
3  -   y=x(1)+0.5*x(2)-1
4  -   end
```

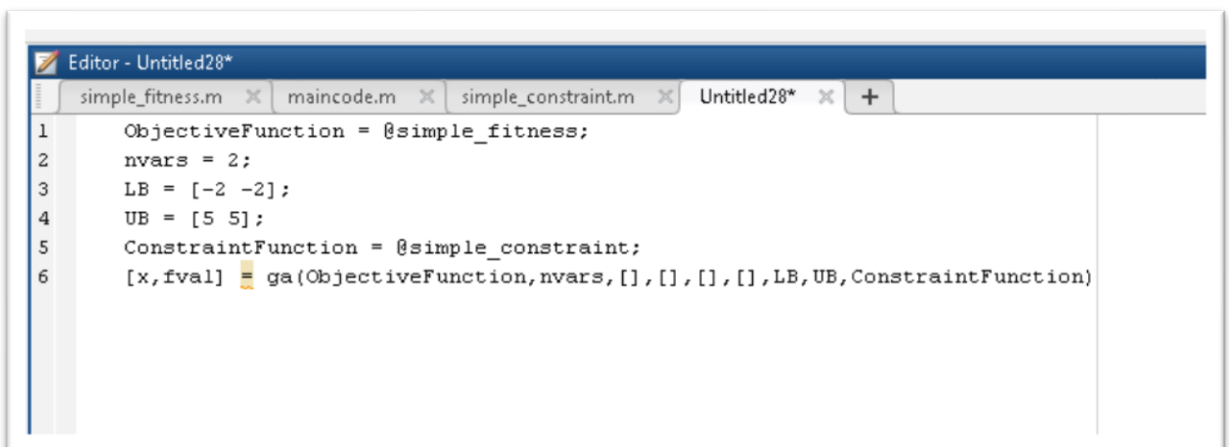
Εικόνα 4. Δημιουργία αντικειμενικής συνάρτησης



```
Editor - C:\Users\Elisabet\Documents\MATLAB\simple_constraint.m
simple_fitness.m x maincode.m x simple_constraint.m x +
1 function [c,ceq]=simple_constraint(x)
2     c=[];
3     ceq=[x(1)+x(2)-1;
4          2*x(1)+x(2)-2;
5          3*x(1)+x(2)-3;]
6 end
```

Εικόνα 5. Δημιουργία των περιορισμών ισότητας

Θέλουμε οι περιορισμοί μας να είναι ίσοι με 0, για αυτό τους γράφουμε μέσα στην μεταβλητή ceq, αλλιώς θα γραφότουσαν στην μεταβλητή c.



```
Editor - Untitled28*
simple_fitness.m x maincode.m x simple_constraint.m x Untitled28* x +
1 ObjectiveFunction = @simple_fitness;
2 nvars = 2;
3 LB = [-2 -2];
4 UB = [5 5];
5 ConstraintFunction = @simple_constraint;
6 [x,fval] = ga(ObjectiveFunction,nvars,[],[],[],[],LB,UB,ConstraintFunction)
```

6. Δομή προβλήματος

Μόλις τρέξει ο γενετικός, εμφανίζεται το παρακάτω μήνυμα:

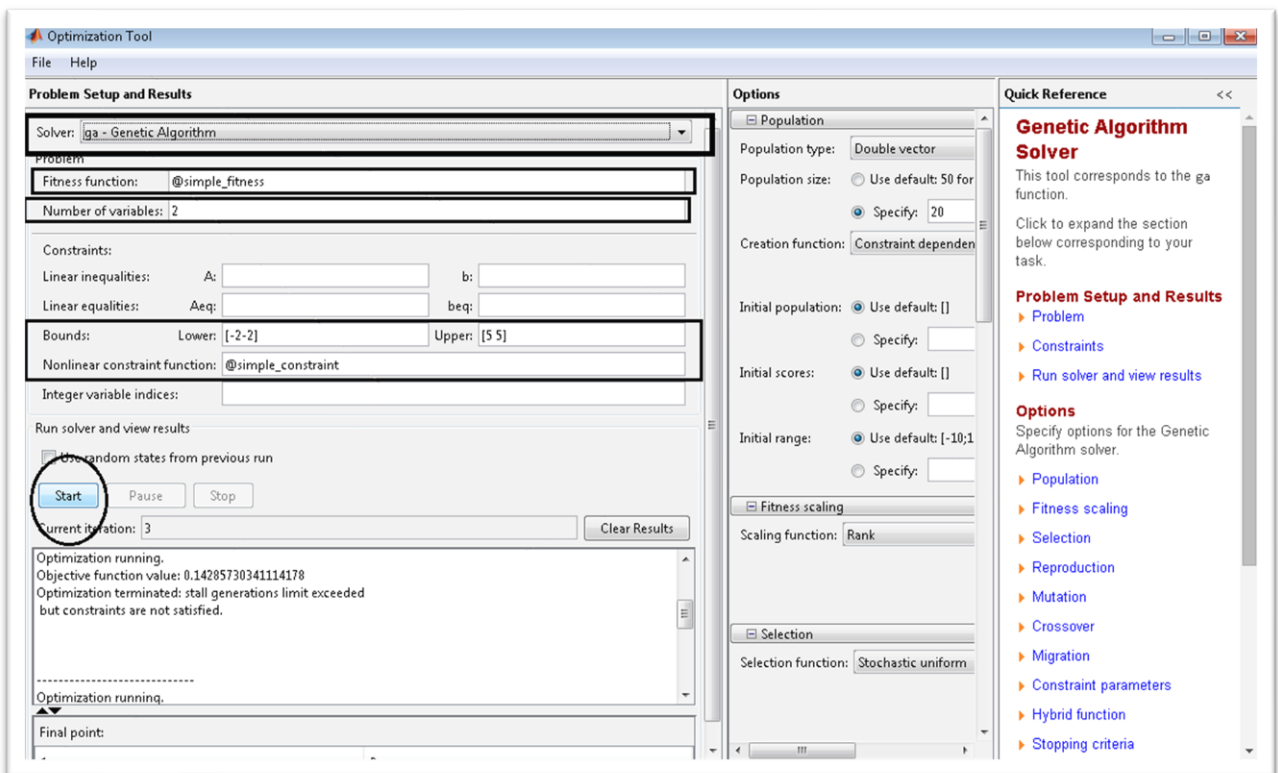
```
Command Window
New to MATLAB? See resources for Getting Started.

y =
-5.3362e-05
y =
-1.1440e-04
Optimization terminated: average change in the fitness value less than options.FunctionTolerance
and constraint violation is less than options.ConstraintTolerance.
x =
1.0000 0.0001
fval =
7.6728e-06
```

7. Αποτελέσματα τρεξίματος

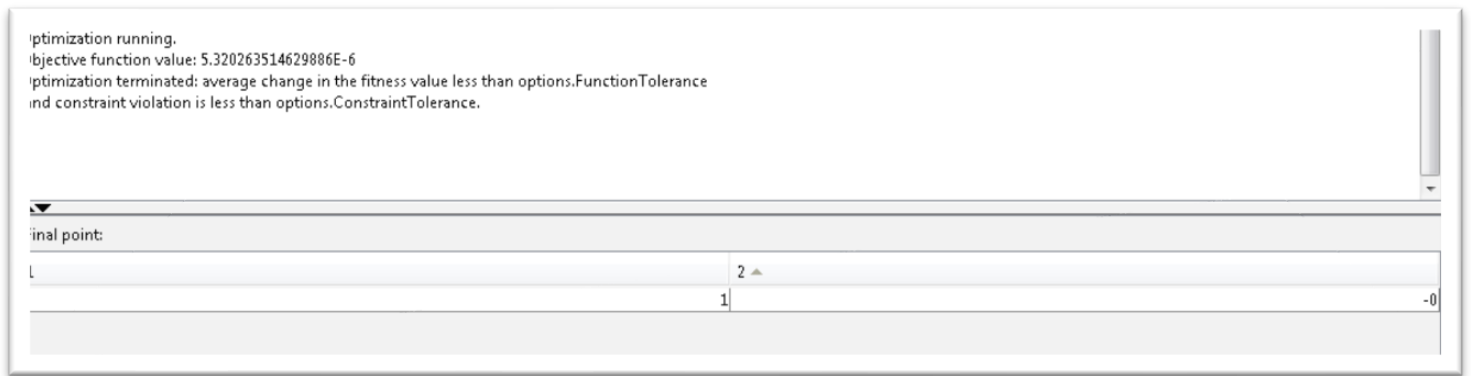
Δηλαδή $C1=1$ και $C2=0$

2.4 Επίλυση προβλήματος χρησιμοποιώντας Genetic Algorithm toolbox-MATLAB



Εικόνα 8. GA TOOLBOX

Ανοίγουμε πάλι το GA TOOLBOX (όπως κάναμε και στην εικόνα 3) με την διαφορά ότι εδώ προσθέτουμε και την συνάρτηση με τους περιορισμούς και τα άνω και κάτω όρια.



Εικόνα 9. Αποτελέσματα τρεξίματος μεσω GA toolbox

Πάλι καταλήγουμε στο αποτέλεσμα ότι **C1=1** και **C2=0**