



Αριστοτέλειο Πανεπιστήμιο Θεσσαλονίκης
Πολυτεχνική Σχολή

Τμήμα Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών
Τομέας Ηλεκτρονικής

Τεχνικές Βελτιστοποίησης - Project

Θεόδωρος Παπαφωτίου
ΑΕΜ: 9708
papafotit@ece.auth.gr

Περιεχόμενα

1 Περιγραφή Project	2
1.1 Αξιολόγηση Αποτελεσμάτων	2
1.2 Υλοποίηση	3
2 Γενετικοί Αλγόριθμοι	4
2.1 Αρχικός Πληθυσμός	5
2.2 Συνάρτηση Ικανότητας	5
2.3 Επιλογή	5
2.4 Διασταύρωση	5
2.5 Μετάλλαξη	6
3 Εφαρμογή Γενετικού Αλγορίθμου	8
3.1 Αρχικός πληθυσμός	8
3.2 Fitness Function	8
3.3 Crossover	10
3.4 Mutation	10
3.5 Εκτέλεση Αλγορίθμου	10
3.6 Δοκιμές	12
3.6.1 Μεταβολή του μεγέθους αρχικού πληθυσμού	12
3.6.2 Μεταβολή του αριθμού γκαουσιανών	13
3.6.3 Μεταβολή της πιθανότητας μετάλλαξης	14
3.6.4 Μεταβολή του ποσοστού επιλογής καλύτερων χρωμοσωμάτων	16
3.6.5 Μεταβολή του ποσοστού επιλογής τυχαίων χρωμοσωμάτων	17
3.7 Επιλεγμένο Χρωμόσωμα	18

Κεφάλαιο 1

Περιγραφή Project

Στόχος της παρούσας εργασίας είναι η εύρεση μιας αναλυτικής έκφρασης για την συνάρτηση f , που περιγράφει το στατικό σύστημα δύο εισόδων u_1, u_2 και μιας εξόδου y .

$$y = f(u_1, u_2) \quad (1.1)$$

Για την εξαγωγή της αναλυτικής έκφρασης της f αξιοποιείται ο **γραμμικός συνδυασμός Gaussian συναρτήσεων** της μορφής:

$$G(u_1, u_2) = \exp \left[-\left(\frac{(u_1 - c_1)^2}{2\sigma_1^2} \right) + \left(\frac{(u_2 - c_2)^2}{2\sigma_2^2} \right) \right] \quad (1.2)$$

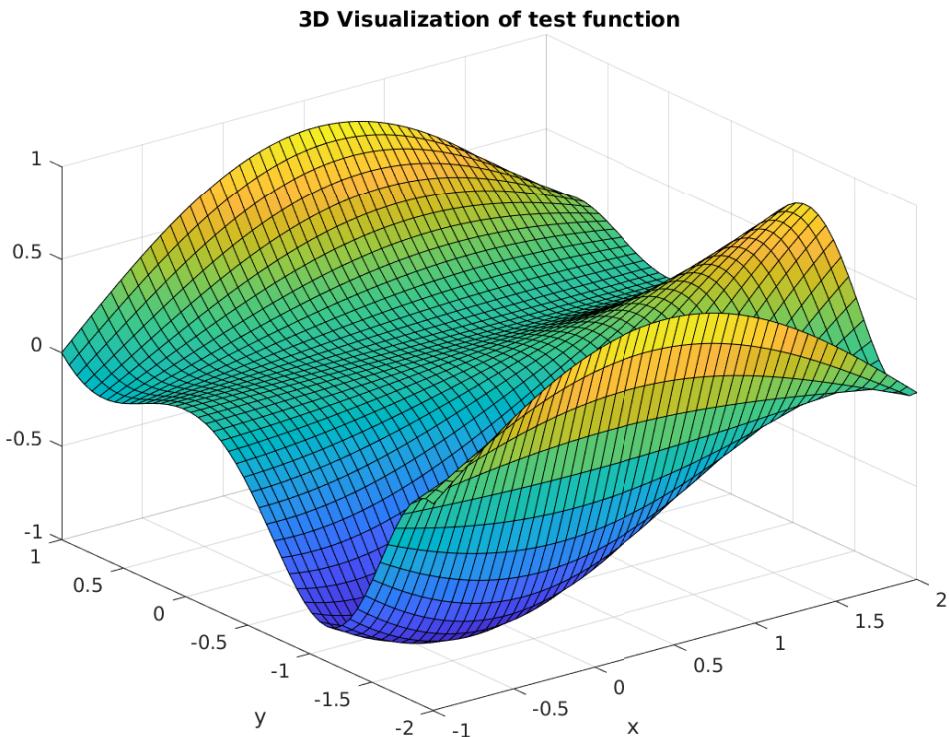
Με τη χρήση **το πολύ 15** Gaussian συναρτήσεων και τη χρήση **Γενετικού Αλγορίθμου**, καλούμαστε να προτείνουμε μια αναλυτική έκφραση της f .

1.1 Αξιολόγηση Αποτελεσμάτων

Για τις διαδικασίες της παραγωγής δεδομένων εισόδου-εξόδου καθώς και για την αξιολόγηση των αποτελεσμάτων που εξάγονται από τον αλγόριθμο, και μόνο, θεωρούμε τη συνάρτηση:

$$f(u_1, u_2) = \sin(u_1 + u_2) \cdot \sin u_2^2, \quad u_1 \in [-1, 2], \quad u_2 \in [-2, 1] \quad (1.3)$$

Στο διάγραμμα 1.1 μπορούμε να παρατηρήσουμε σε 3D οπτικοποίηση την παραπάνω συνάρτηση.



Σχήμα 1.1: 3D visualisation of test function

Για την αξιολόγηση της ποιότητας της αναλυτικής έκφρασης, χρησιμοποιείται **διαφορετικό σύνολο δεδομένων** εισόδου-εξόδου από αυτό που χρησιμοποιείται για τον προσδιορισμό της αναλυτικής έκφρασης.

1.2 Υλοποίηση

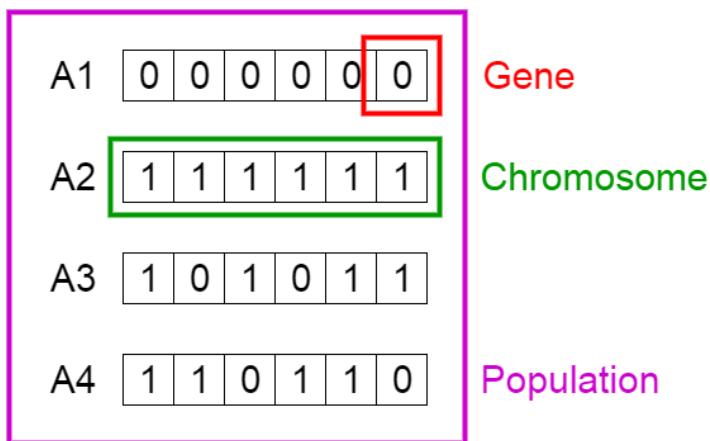
Η υλοποίηση της εργασίας πραγματοποιήθηκε με τη χρήση του λογισμικού **MATLAB** (v.2021a). Το κεντρικό αρχείο του project είναι το `main.m`, το οποίο αξιοποιεί τα συνοδευτικά αρχεία (`Genetic_Algorithm.m`, `mutate.m`, `crossover.m`, `limits_f.m`, `fitness_function.m`, `f_analytic.m`, `sort_fit.m`, `plot_3D.m`). Όλα τα αρχεία επισυνάπτονται με την παρούσα τεχνική αναφορά στο αποθετήριο της πλατφόρμας eLearning.

Κεφάλαιο 2

Γενετικοί Αλγόριθμοι

Σύμφωνα με τη **Θεωρία της Εξέλιξης**, οι οργανισμοί που είναι καλύτερα προσαρμοσμένοι στο περιβάλλον τους, μεταβιβάζουν χαρακτηριστικά στους απογόνους τους, σε αντίθεση με εκείνους που είναι λιγότερο προσαρμοσμένοι.

Από αυτήν την λογική πήραν την έμπνευσή τους οι Γενετικοί Αλγόριθμοι. Όπως οι οργανισμοί, έτσι και οι λύσεις ενός προβλήματος, μέσα από την εφαρμογή γενετικών τελεστών, **μεταβιβάζουν χαρακτηριστικά** από γενιά σε γενιά, οδηγώντας σταδιακά από γενιά σε γενιά στην εξαφάνιση των λιγότερο προσαρμοσμένων λύσεων και στη βελτίωση των καλύτερα προσαρμοσμένων λύσεων.



Σχήμα 2.1: Genetic Algorithm

Σε ένα γενετικό αλγόριθμο, μπορούμε να διακρίνουμε τα παρακάτω στάδια:

1. Αρχικός Πληθυσμός (Initialization)
2. Συνάρτηση Ικανότητας (Fitness function)
3. Επιλογή (Selection)
4. Διασταύρωση (Crossover)
5. Μετάλλαξη (Mutation)

2.1 Αρχικός Πληθυσμός

Για την έναρξη της διαδικασίας, ορίζεται **τυχαία** ένα set από ανεξάρτητα "πακέτα", που συνιστά τον αρχικό πληθυσμό. Το κάθε τέτοιο πακέτο ονομάζεται **Χρωμόσωμα**, συγκροτείται από έναν συγκεκριμένο αριθμό παραμέτρων που ονομάζονται **Γονίδια** (Genes) και αποτελεί μια ενδεχόμενη λύση στο πρόβλημα που θέλουμε να λύσουμε (σχήμα 2.1).

Συνήθως, τα γονίδια κωδικοποιούνται με δυαδικές τιμές, οπότε το χρωμόσωμα αποτελείται από Ο και 1 μόνο.

2.2 Συνάρτηση Ικανότητας

Η συνάρτηση ικανότητας (Fitness Function) προσδιορίζει σε κάθε χρωμόσωμα το πόσο ικανό/ταιριαστό (fit) είναι (ικανότητά του χρωμοσώματος να "συναγωνιστεί" με τα υπόλοιπα χρωμοσώματα). Η πιθανότητα να επιλεγεί ένα συγκεκριμένο χρωμόσωμα για να αναπαραγγίγεται βασίζεται στο **βαθμό ικανότητάς** του (fitness score).

2.3 Επιλογή

Ο γενετικός αλγόριθμος τρέχει μέσα σε ένα βρόχο (loop) έως ότου ξεπεραστεί ο μέγιστος αριθμός γενεών που έχουν οριστεί, ή μόλις βρεθεί το πιο ικανό χρωμόσωμα, σύμφωνα με το όριο που του έχουμε θέσει εμείς.

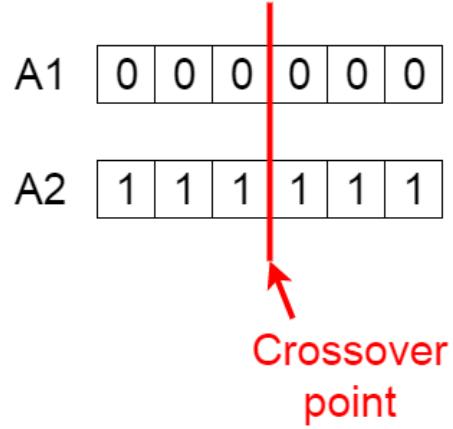
Σε κάθε επανάληψη/γενεά χρειάζεται να επιλέγουμε **τα πιο ικανά χρωμοσώματα** και να τους επιτρέψουμε να μεταφέρουν τα γονίδιά τους στην επόμενη γενιά.

Το ποσοστό των καλύτερων χρωμοσωμάτων που επιλέγονται σε κάθε γενιά καθορίζεται από το χρήστη.

Συγχρόνως επιλέγεται τυχαία κάποιο ποσοστό χρωμοσωμάτων εκ των υπολοίπων, **ανεξάρτητα** από το fitness score τους.

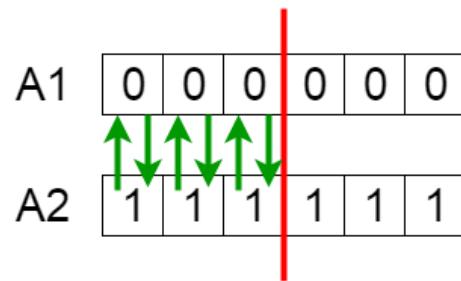
2.4 Διασταύρωση

Η Διασταύρωση (Crossover) θεωρείται το πιο σημαντικό κομμάτι ενός γενετικού αλγορίθμου (όπως και στη θεωρία της εξέλιξης). Για κάθε ζευγάρι χρωμοσωμάτων που επιλέγεται τυχαία (γονείς - parents) για διασταύρωση, ορίζεται τυχαία ένα **σημείο διασταύρωσης** (crossover point) μεταξύ των γονιδίων, κοινό και στα 2 χρωμοσώματα (σχήμα 2.2).



Σχήμα 2.2: Crossover Point

Στη συνέχεια, δημιουργούνται οι **απόγονοι** με την ανταλλαγή γονιδίων μεταξύ των γονέων μέχρι το σημείο διασταύρωσης (σχήμα 2.3). Οι 2 απόγονοι που σχηματίζονται προστίθενται στον πληθυσμό.



Σχήμα 2.3: Crossover Offsprings

2.5 Μετάλλαξη

Εφόσον σχηματιστεί ο πληθυσμός μιας γενεάς, κάποια γονίδια ορισμένων χρωμοσωμάτων που επιλέγονται με τυχαίο τρόπο και με **μικρή πιθανότητα**, μπορούν να υποστούν μετάλλαξη (mutation). Αυτό σημαίνει ότι οι τιμές των παραμέτρων τους μεταβάλλονται προς μια τυχαία τιμή.

Σε περίπτωση που τα γονίδια έχουν κωδικοποιηθεί με το δυαδικό σύστημα, κατά τη μετάλλαξη γίνεται flip ορισμένων bits στο χρωμόσωμα. Προφανώς ο βαθμός σημαντικότητας ενός bit στην κάθε κωδικοποιημένη παράμετρο παίζει ρολό στο κατά πόσο θα επηρεαστεί το χρωμόσωμα αν το συγκεκριμένο bit γίνει flip (σχήμα 2.4).

Before Mutation

A5	1	1	1	0	0	0
----	---	---	---	---	---	---

After Mutation

A5	1	1	0	1	1	0
----	---	---	---	---	---	---

Σχήμα 2.4: Mutation

Η διαδικασία των μεταλλάξεων συμβαίνει για να διατηρηθεί η **ποικιλομορφία** εντός του πληθυσμού, ενώ συγχρόνως εμποδίζει τον γενετικό αλγόριθμο από το να "κολλήσει" σε περιοχές του χώρου αναζήτησης (σε κάποιο τοπικά βέλτιστο σημείο πρακτικά) με το να **διαταράσσει ελαφρά** το χρωμόσωμα.

Όλες οι παραπάνω διαδικασίες (Crossover, Mutation, κωδικοποίηση γονιδίων του χρωμοσώματος και επιλογή της Fitness Function μπορούν να γίνουν με διάφορες τεχνικές και διαφέρουν από πρόβλημα σε πρόβλημα.

Κεφάλαιο 3

Εφαρμογή Γενετικού Αλγορίθμου

Στην εργασία που καλούμαστε να υλοποιήσουμε, στόχος είναι να προσεγγίσουμε την δοσμένη συνάρτηση f (test function) με μια αναλυτική έκφραση που θα προκύπτει από τον **γραμμικός συνδυασμός** Gaussian συναρτήσεων της μορφής 1.2. Συνεπώς θα έχουμε μια συνάρτηση της μορφής:

$$f_{\text{analytic}}(u_1, u_2) = A_1 \cdot G_1(u_1, u_2) + A_2 \cdot G_2(u_1, u_2) + \dots + A_n \cdot G_n(u_1, u_2) \quad (3.1)$$

όπου n ο συνολικός αριθμός των Gaussian που θα αξιοποιηθούν ($n_{\max} = 15$). Η τιμή της κάθε Gaussian ($G_1 \dots G_n$) εξαρτάται με τη σειρά της από τις τιμές των παραμέτρων:

$$s_{k_1} \quad s_{k_2} \quad c_{k_1} \quad c_{k_2} \quad (3.2)$$

όπου k ο αριθμός της εκάστοτε γκαουσιανής.

Συνεπώς, αν θεωρήσουμε **συντελεστές** της f_{analytic} το γινόμενο $A_k \cdot G_k$, έχουμε συνολικά **5** παραμέτρους/γονίδια (genes) που διαμορφώνουν την τιμή του κάθε συντελεστή, $A_k, s_{k_1}, s_{k_2}, c_{k_1}, c_{k_2}$. Τελικά, δηλαδή, το κάθε **χρωμόσωμα** αποτελείται από $5n$ γονίδια, όπου n ο συνολικός αριθμός των συντελεστών για τη δημιουργία της f_{analytic} .

Η εγγραφή των γονιδίων στα χρωμοσώματα δεν έγινε με δυαδική κωδικοποίηση, αλλά η κάθε παράμετρος αναπαριστάται με τη **δεκαδική** της μορφή.

3.1 Αρχικός πληθυσμός

Ο αρχικός πληθυσμός δεν πρέπει να είναι ούτε πολύ μικρός, για να υπάρχει μεγαλύτερο πλήθος λύσεων, αλλά ούτε και πολύ μεγάλος καθώς δημιουργούνται προβλήματα αποθήκευσης και διαχείρισης της μνήμης. Στις περισσότερες δοκιμές επιλέχθηκε πληθυσμός ίσος με 50.

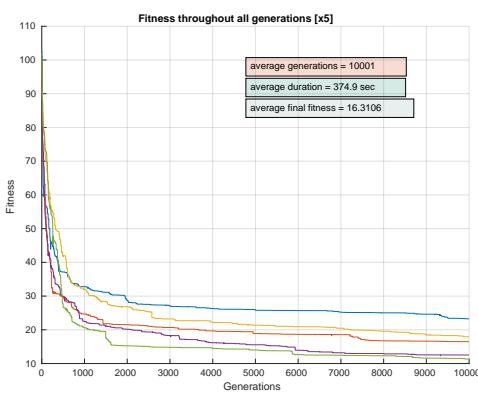
3.2 Fitness Function

Εφόσον στόχος της εργασίας είναι η προσέγγιση μιας συνάρτησης, η χρήση του σφάλματος, δηλαδή της απόκλισης από την πραγματική τιμή, ως συνάρτηση ικανότητας είναι ιδανική.

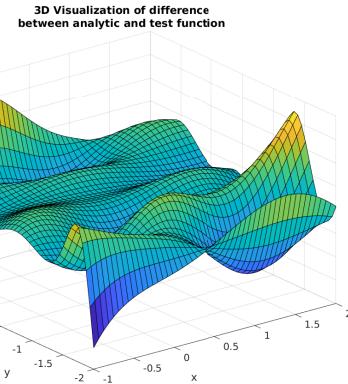
Ακολουθώντας αυτή τη λογική, μια αρχική δοκιμή ήταν με το **απόλυτο σφάλμα**:

$$e_{\text{abs}} = \sum |f(x_i) - \bar{f}(x_i)| \quad (3.3)$$

όπου $f(x_i)$ η πραγματική τιμή και $\bar{f}(x_i)$ η προσέγγισή της. Επειδή όμως οι τιμές του σφάλματος είναι υψηλές, ο αλγόριθμος **συγκλίνει αργά**, ωστόσο καταλήγει σε ένα ικανοποιητικό αποτέλεσμα, όπως μπορούμε να δούμε στα διαγράμματα 3.2 και 3.1.



Σχήμα 3.1: Fitness - Absolute Error



Σχήμα 3.2: $f_{analytic} - f$ -Absolute Error

Τελικά όμως χρησιμοποιήθηκε το **μέσο τετραγωνικό σφάλμα** (mean squared error)

$$e_{mse} = \frac{1}{n} \sum \left(f(x_i) - \bar{f}(x_i) \right)^2 \quad (3.4)$$

όπου n ο αριθμός των σημείων που χρησιμοποιήθηκαν για υπολογιστεί η προσέγγιση.

Ο χρήστης κατά την έναρξη εκτέλεσης του αλγορίθμου μπορεί αν επιλέξει ποιον τύπο error επιδιυμεί για το σχηματισμό της συνάρτησης πιθανότητας

Ποιο αναλυτικά, για τον υπολογισμό της συνάρτησης ικανότητας, δημιουργήθηκε ο παρακάτω αλγόριθμος:

$$X = \underbrace{x_{\min} + \frac{x_{\max} - x_{\min}}{s}, x_{\min} + 2 \cdot \frac{x_{\max} - x_{\min}}{s}, \dots, x_{\max}}_{s \text{ στοιχεία}}$$

$$Y = \underbrace{y_{\min} + \frac{y_{\max} - y_{\min}}{s}, y_{\min} + 2 \cdot \frac{y_{\max} - y_{\min}}{s}, \dots, y_{\max}}_{s \text{ στοιχεία}}$$

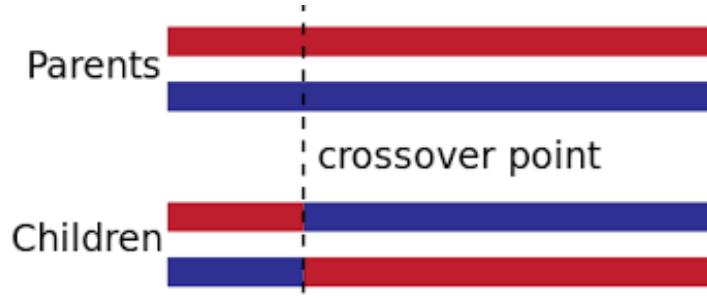
$$f_{fit} = \sum_{n \in X} \sum_{k \in Y} \left[f(n, k) - f_{analytic}(n, k, chromosome) \right]^2$$

$$f_{fit} \leftarrow \frac{f_{fit}}{s^2}$$

Η μεταβλητή s κατά τη διάρκεια των δοκιμών παρέμεινε **σταθερή** και ίση με $s = 20$. Αν θέλουμε να λάβουμε υπόψιν ακόμη περισσότερα σημεία της συνάρτησης για καλύτερη προσέγγιση, μπορούμε να αυξήσουμε το s αυξάνοντας φυσικά τις πράξεις που πραγματοποιούνται, άρα και τον χρόνο εκτέλεσης του αλγορίθμου. Το s ορίστηκε ίσο με 50 μόνο για το σχηματισμό του **τελικώς επιλεγμένου χρωμοσώματος** σε αυτήν την εργασία.

3.3 Crossover

Για τη διαδικασία του Crossover επιλέγεται τυχαία στα 2 χρωμοσώματα προς διαστά-ύρωση το Crossover Point, σχηματίζονται οι 2 απόγονοι και προστίθενται στον πληθυ-σμό, ακριβώς όπως εξηγήθηκε στο προηγούμενο κεφάλαιο.

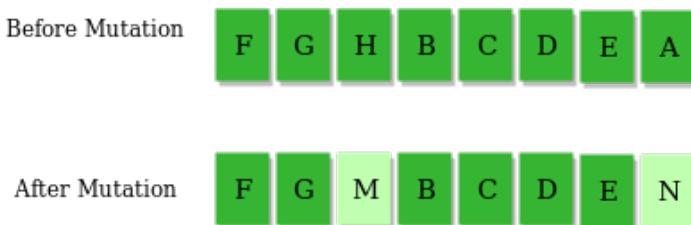


Σχήμα 3.3: Crossover - Split method

Οι τυχαίες τιμές προκύπτουν από ομοιόμορφη κατανομή εντός του δοσμένου εύρους τι-μών.

3.4 Mutation

Για τη διαδικασία του Mutation επιλέγονται **τυχαία** κάποια γονίδια από το δοσμένο χρωμόσωμα και **επαναπροσδιορίζονται** οι τιμές τους εντός του επιτρεπτού εύρους τιμών του εκάστοτε γονιδίου πάλι με τυχαίο τρόπο. Η διαδικασία φαίνεται και στο σχήμα 3.4.



Σχήμα 3.4: Mutation

Οι τυχαίες τιμές προκύπτουν από ομοιόμορφη κατανομή εντός του δοσμένου εύρους τι-μών.

3.5 Εκτέλεση Αλγορίθμου

Παρακάτω μπορούμε να δούμε με τη μορφή ψευδοκώδικα τη δομή του αλγορίθμου:

Αρχικά, πριν την εκτέλεση του αλγορίθμου, χρειάζεται να τεθούν τα όρια των μεταβλητών $A_k, s_{k_1}, s_{k_2}, c_{k_1}, c_{k_2}$, σύμφωνα με τον παρακάτω πίνακα:

Algorithm 1 Genetic Algorithm - Pseudocode

Generate Initial Population
Compute Fitness Function
while Population has not converged && Generation < MAX_generations **do**
 Selection Process
 Crossover Process
 Mutation Process
 Compute Fitness Function
 $Generation \leftarrow Generation + 1$
end while
 $result \leftarrow BestChromosome$

	min	max
A	$f_{\min} \approx -1$	$f_{\max} \approx 0.842$
c	-5	5
s	0.1	1.2

Οι τιμές των γονιδίων επιλέγονται από το αντίστοιχο εύρος τιμών.

- Στην περίπτωση των κέντρων c επιλέχθηκαν τα όρια έτσι ώστε να είναι **συμμετρικά** γύρω από το 0. Η απόλυτη τιμή των ορίων επιλέχθηκε τυχαία, αλλά μεγαλύτερη από την μεγαλύτερη, κατά απόλυτη τιμή, τιμή του πεδίου ορισμού (a, b) των x και y στην συνάρτηση f . Στην περίπτωση μας δηλαδή, $\text{abs}(c) > 2$.
- Στην περίπτωση της διασποράς s επιλέχθηκαν τα όρια έτσι ώστε η παράμετροι s_{k_1}, s_{k_2} να παίρνουν πάντα **τιμές μεγαλύτερες του 0**. Τα όρια επιλέχθηκαν τυχαία.

Στη συνέχεια χρειάζεται να καθορίσουμε τις παρακάτω παραμέτρους:

- Ακρίβεια [Min threshold του error που θέλουμε να πετύχουμε]
- Μέγιστο Αριθμό Γενεών (generations)
- Πλήθος Gaussian που χρησιμοποιούνται για το σχηματισμό της αναλυτικής έκφρασης της f
- Αρχικός Πληθυσμός (Population)
- Πιθανότητα Mutation σε κάθε γενεά
- Ποσοστό των καλύτερα προσαρμοσμένων χρωμοσωμάτων που μεταφέρονται ανέπαφα στην επόμενη γενεά και αξιοποιούνται για διασταυρώσεις.
- Ποσοστό των τυχαία επιλεγμένων χρωμοσωμάτων που μεταφέρονται ανέπαφα στην επόμενη γενεά

Το ποσοστό χρωμοσωμάτων από διασταυρώσεις (crossover) επιλέγεται έτσι ώστε ο πληθυσμός να παραμένει **σταθερός**.

Δοκιμάζοντας διαφορετικούς συνδυασμούς αυτών των παραμέτρων μπορούμε να πετύχουμε και πάλι βελτιστοποίηση της αναλυτικής έκφρασης της f , αλλά με διαφορετικό **συνολικό χρόνο, αριθμό generations** ή/και **ακρίβεια**.

Παρακάτω, μπορούμε να παρατηρήσουμε πώς η μεταβολή των παραπάνω παραμέτρων μπορεί να επηρεάσει τα αποτελέσματα του γενετικού αλγορίθμου.

3.6 Δοκιμές

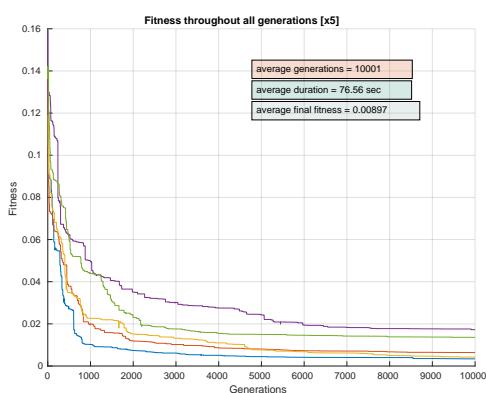
Σε όλες τις παρακάτω δοκιμές, διατηρήθηκε **σταθερός αριθμός επαναλήψεων** ίσος με **10000**, ενώ ο αλγόριθμος εκτελέστηκε από **5 φορές** για κάθε περίπτωση, έτσι ώστε να εξαλειφθεί κατά το δυνατόν η τυχαιότητα και ο πιθανοκρατικός χαρακτήρας των αποτελεσμάτων.

Για την παραγωγή των δεδομένων εισόδου-εξόδου, χρησιμοποιήθηκαν 20 σημεία για κάθε ένα από τα x, y στο εύρος που τα έχουμε ορίσει. Για την δημιουργία των γραφημάτων χρησιμοποιήθηκαν 50 σημεία για κάθε ένα από το x, y στο ίδιο διάστημα, έτσι ώστε να χρησιμοποιηθούν διαφορετικά σημεία για τις δύο αυτές φάσεις κατά την εκτέλεση του αλγορίθμου (διαφορετικό σύνολο δεδομένων εισόδου-εξόδου από αυτό που χρησιμοποιήθηκε για τον προσδιορισμό της αναλυτικής έκφρασης).

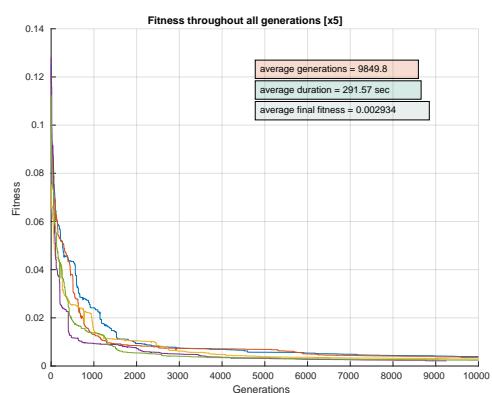
3.6.1 Μεταβολή του μεγέθους αρχικού πληθυσμού

	value
fit_limit	0.002
num_of_gaussians	15
max_generations	10000
select_best_probability	0.4
select_random_probability	0.1
mutation_probability	0.1

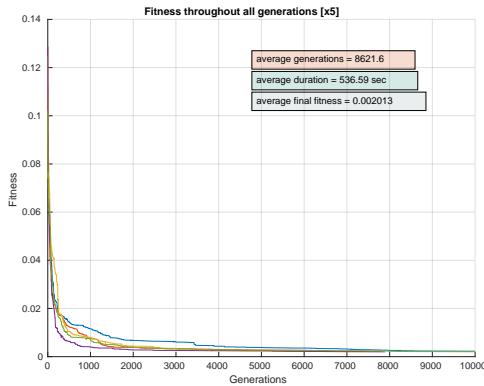
Παρατηρώντας τα παρακάτω διαγράμματα, μπορούμε να συμπεράνουμε ότι η αύξηση του πληθυσμού οδηγεί σε μείωση των απαιτούμενων γενεών για σύγκλιση στην ίδια καταλληλότητα, ωστόσο οδηγεί συγχρόνως σε αύξηση του απαιτούμενου χρόνου για την ολοκλήρωση. Αυτό συμβαίνει γιατί σε κάθε επανάληψη ο αλγόριθμος έχει να επεξεργαστεί παραπάνω χρωμοσώματα όταν ο πληθυσμός αυξάνεται.



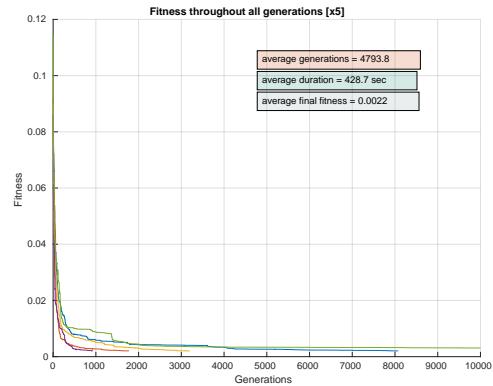
Σχήμα 3.5: Population = 10



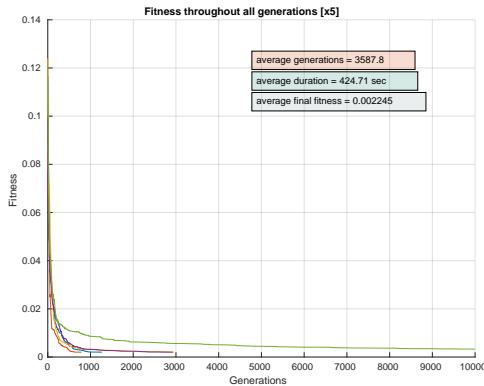
Σχήμα 3.6: Population = 40



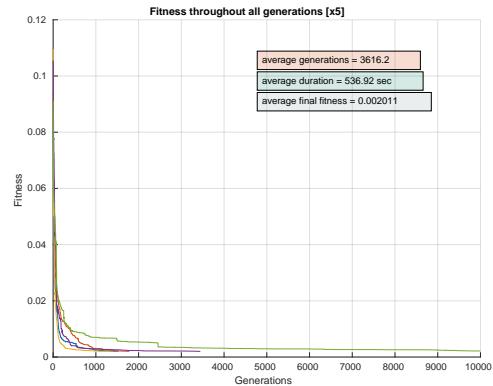
Σχήμα 3.7: Population = 80



Σχήμα 3.8: Population = 120



Σχήμα 3.9: Population = 160



Σχήμα 3.10: Population = 200

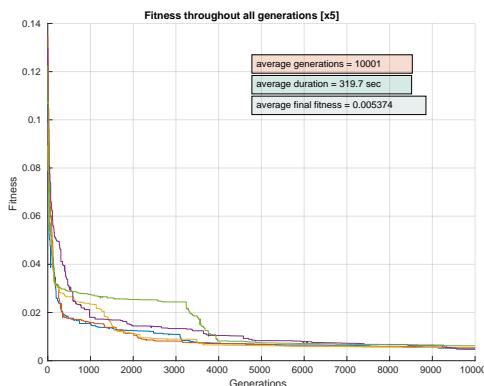
3.6.2 Μεταβολή του αριθμού γκαουσιανών

	value
fit_limit	0.002
mutation_probability	0.1
max_generations	10000
select_best	0.4
select_random	0.1
num_of_population	50

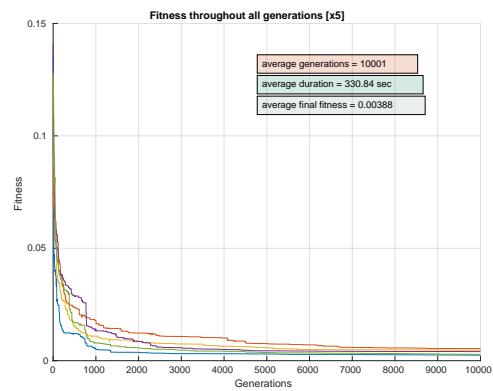
Από τα παρακάτω διαγράμματα μπορούμε να παρατηρήσουμε ότι ο αριθμός των Gaussian που χρησιμοποιούνται για την αναλυτική έκφραση της f επηρεάζει τον αριθμό των γενεών που απαιτούνται για να φτάσουμε την απαιτούμενη ακρίβεια. Συγκεκριμένα:

- Αν θέσουμε την ζητούμενη ακρίβεια ικανότητας σε τιμές > 0.1 , για παράδειγμα, παρατηρούμε ότι και με **μικρό αριθμό γκαουσιανών** μπορούμε να φτάσουμε στο ζητούμενο αποτέλεσμα και μάλιστα σε συντομότερο χρόνο, αφού τα γονίδια είναι λιγότερα σε κάθε χρωμόσωμα, άρα κατά επέκταση είναι λιγότεροι και οι υπολογισμοί που απαιτούνται.

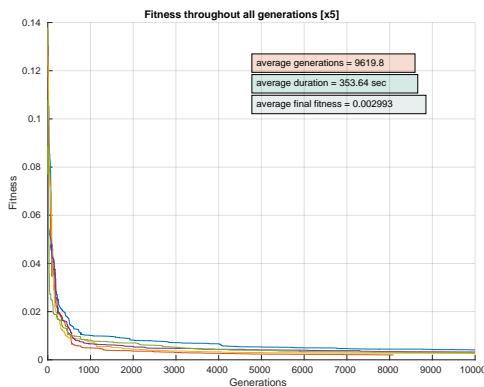
- Αν θέσουμε ωστόσο την ζητούμενη ακρίβεια ικανότητας σε τιμές < 0.1 , για παράδειγμα, παρατηρούμε ότι χρειαζόμαστε **μεγαλύτερο αριθμό γκαουσιανών** για να φτάσουμε συντομότερα στο επιθυμητό αποτέλεσμα. Αυτό είναι αναμενόμενο, καθώς με περισσότερες γκαουσιανές μπορείς να προσεγγίσεις "ευκολότερα" μια συνάρτηση με μεγάλη ακρίβεια. Παρατηρούμε επίσης ότι όσο αυξάνεται ο αριθμός των γκαουσιανών, η **πορεία** του γενετικού αλγορίθμου φαίνεται να είναι ίδια προς τη σύγκλιση όσες φορές και να τον τρέξουμε.



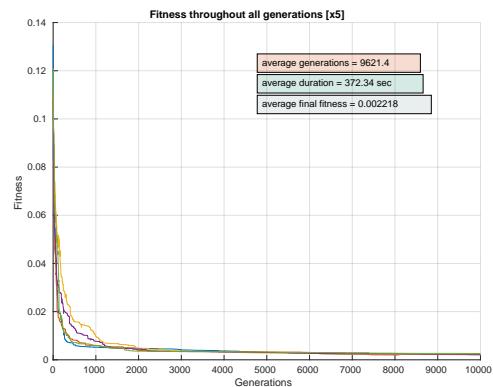
Σχήμα 3.11: Num_of_gaussians = 6



Σχήμα 3.12: Num_of_gaussians = 9



Σχήμα 3.13: Num_of_gaussians = 12



Σχήμα 3.14: Num_of_gaussians = 15

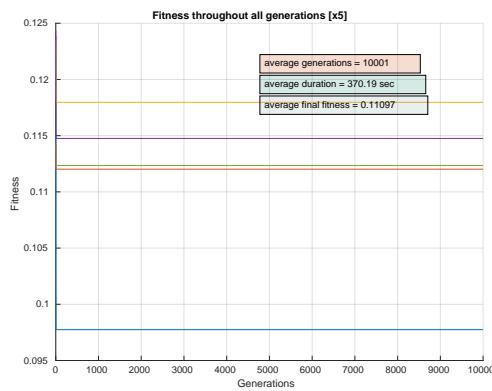
3.6.3 Μεταβολή της πιθανότητας μετάλλαξης

	value
fit_limit	0.002
num_of_gaussians	15
max_generations	10000
select_best	0.4
select_random	0.1
num_of_population	50

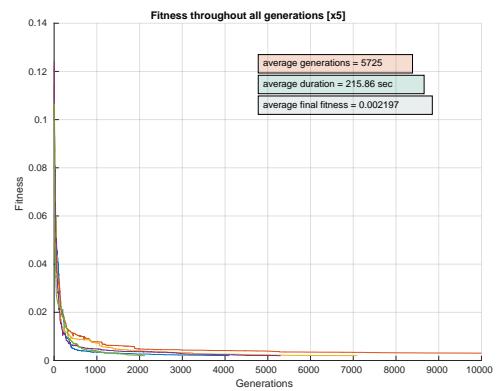
Με τη μεταβολή της πιθανότητας μετάλλαξης εντός ενός εύρους [5% - 60%] παρατηρούμε μικρές διαφορές στο συνολικό αριθμό γενεών που απαιτείται για επίτευξη του

threshold, ενώ, όσο αυξάνεται η πιθανότητα μετάλλαξης ($\geq 70\%$), παρατηρούμε περισσότερες "διαταράξεις" πάνω στην καμπύλη και αδυναμία σύγκλισης εντός του ορίου σύγκλισης (fit_limit). Η επιδόσεις του αλγορίθμου προφανώς χειροτερεύουν όσο πλησιάζουμε σε 100% πιθανότητα μετάλλαξης.

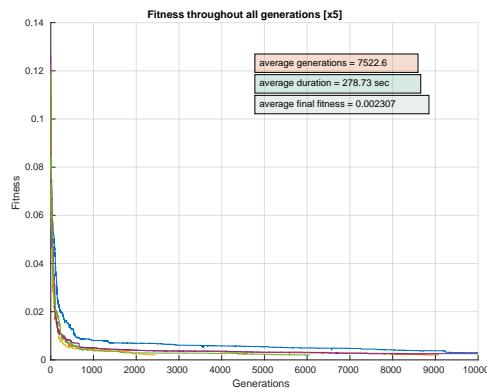
Ωστόσο πρέπει να τονίσουμε ότι ένα ποσοστό μετάλλαξης είναι απαραίτητο, καθώς, όπως μπορούμε να δούμε στο διάγραμμα 3.15, όταν ο αλγόριθμος τρέχει με μηδενική πιθανότητα μετάλλαξης εγκλωβίζεται σε μια τοπικά βέλτιστη λύση και χρειάζεται μια ελαφριά διαταραχή στα χρωμοσώματα.



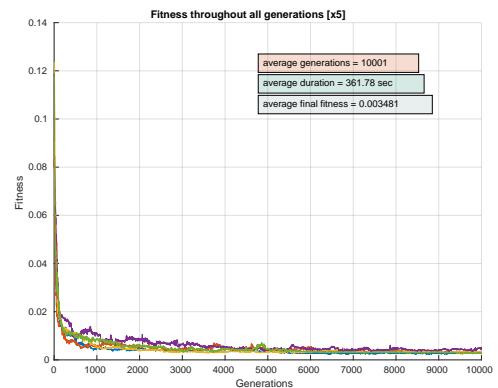
Σχήμα 3.15: Mutation Prob. = 0%



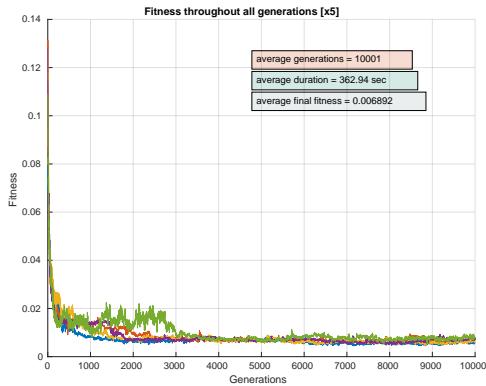
Σχήμα 3.16: Mutation Prob. = 30%



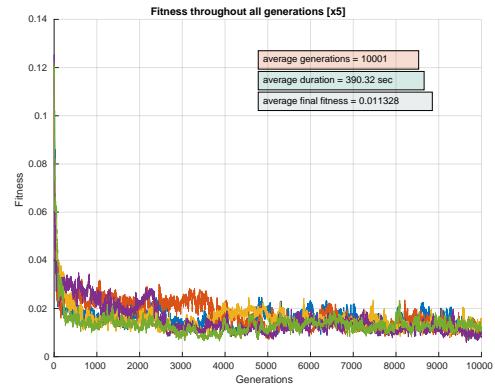
Σχήμα 3.17: Mutation Prob. = 50%



Σχήμα 3.18: Mutation Prob. = 70%



Σχήμα 3.19: Mutation Prob. = 90%



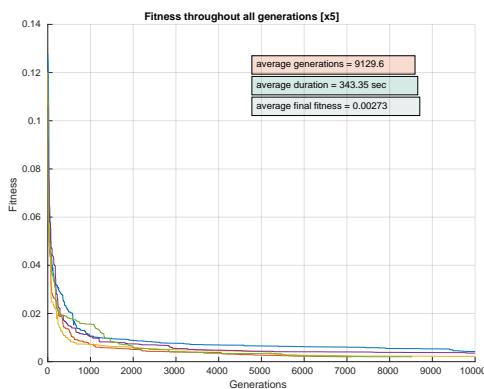
Σχήμα 3.20: Mutation Prob. = 100%

3.6.4 Μεταβολή του ποσοστού επιλογής καλύτερων χρωμοσωμάτων

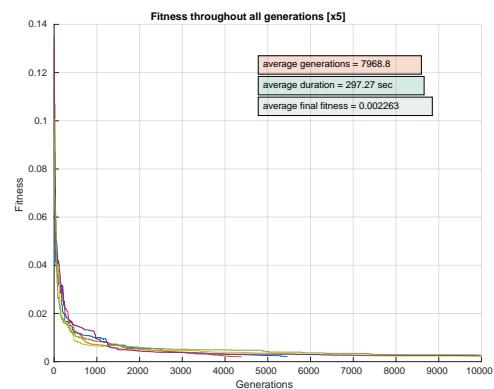
Αρχικά, να τονίσουμε ότι η μεταβολή του ποσοστού επιλογής καλύτερων χρωμοσωμάτων για να περάσουν ανέπαφα στην επόμενη γενεά, διατηρώντας σταθερό το ποσοστό επιλογής τυχαίων χρωμοσωμάτων και τον πληθυσμό, συνεπάγεται άμεσα **μεταβολή** του αριθμού των χρωμοσωμάτων που θα παραχθούν από **διασταύρωση** (αύξηση του ενός συνεπάγεται μείωση του άλλου).

Από τα διαγράμματα παρατηρούμε ότι εντός ενός "κεντρικού" εύρους [20% - 50%] οι μεταβολές είναι μικρές και μπορούμε να πετύχουμε σύγκλιση εντός του ορίου `max_generations`, ωστόσο όταν ξεπεράσουμε αυτό το όριο (50%) παρατηρούμε περισσότερες "διαταραχές" και αδυναμία σύγκλισης εντός του προβλεπόμενου αριθμού γενεών. Αυτό είναι λογικό, καθώς όσο αυξάνεται το ποσοστό επιλογής καλύτερων χρωμοσωμάτων, μειώνεται ο αριθμός χρωμοσωμάτων από διασταύρωση, οπότε δεν έχουμε σημαντικές μεταβολές/θελτιώσεις στην ικανότητα της κάθε γενεάς.

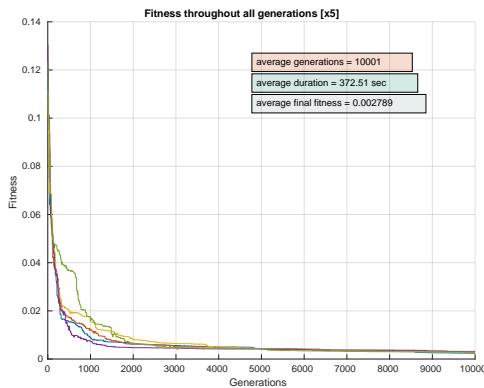
	value
<code>fit_limit</code>	0.002
<code>num_of_gaussians</code>	15
<code>max_generations</code>	10000
<code>mutation_prob</code>	0.1
<code>select_random</code>	0.1
<code>num_of_population</code>	50



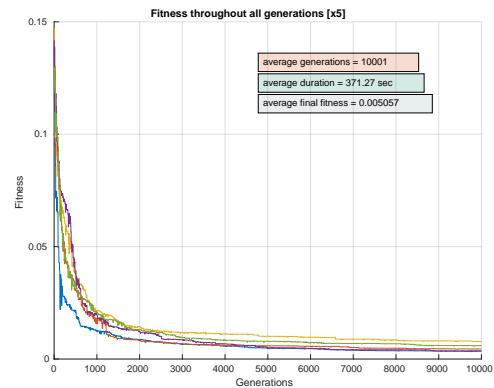
Σχήμα 3.21: Best Selection = 20%



Σχήμα 3.22: Best Selection = 40%



Σχήμα 3.23: Best Selection = 60%



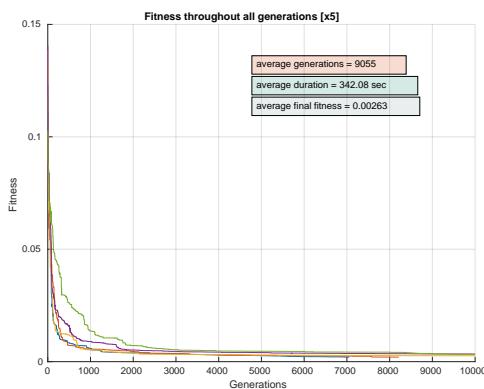
Σχήμα 3.24: Best Selection = 80%

3.6.5 Μεταβολή του ποσοστού επιλογής τυχαίων χρωμοσωμάτων

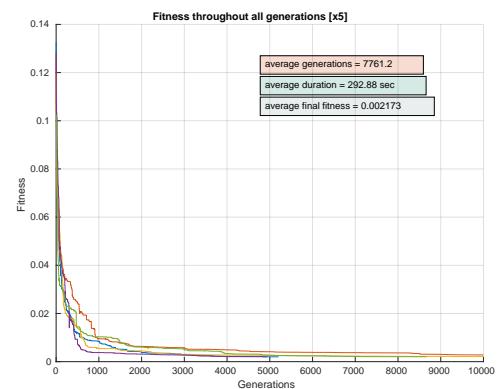
Όπως αναφέρθηκε και στο προηγούμενο section έτσι και με τη μεταβολή του ποσοστού επιλογής τυχαίων χρωμοσωμάτων έχουμε άμεση μεταβολή του αριθμού χρωμοσωμάτων από διασταύρωση, ώστε να διατηρηθεί σταθερός ο πληθυσμός.

Στα παρακάτω διαγράμματα είναι εμφανές ότι κινηθήκαμε σε πιο μικρά ποσοστά (κυρίως επειδή κρατήσαμε σταθερή την τιμή του ποσοστού επιλογής καλύτερων χρωμοσωμάτων), οπότε σε όλες τις περιπτώσεις φαίνεται να συγκλίνει ο αλγόριθμος, αλλά όσο αυξάνεται το ποσοστό των τυχαίων επιλεγόμενων χρωμοσωμάτων βλέπουμε ότι ο αλγόριθμος συγκλίνει πιο "άτακτα" (δηλαδή πιο τυχαία) στο όριο που έχουμε θέσει. Αυτό είναι λογικό, γιατί δίνουμε κατά αυτόν τον τρόπο μεγαλύτερο χώρο στην τυχαιότητα του αλγορίθμου.

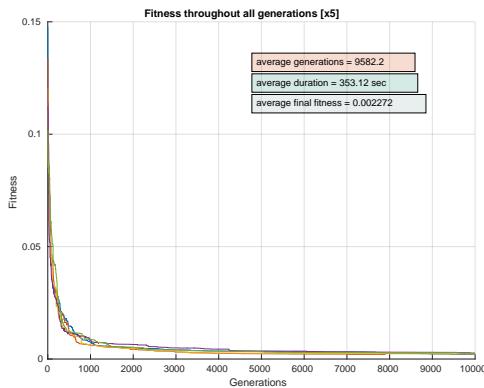
	value
fit_limit	0.002
num_of_gaussians	15
max_generations	10000
mutation_prob	0.1
select_best	0.4
num_of_population	50



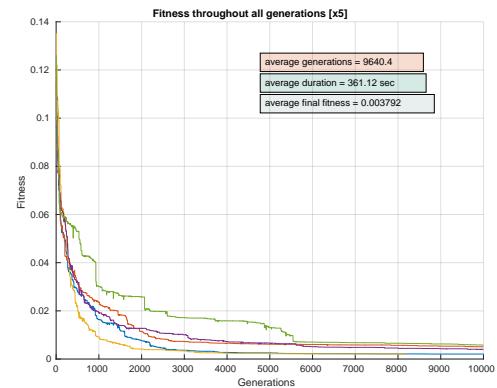
Σχήμα 3.25: Random Selection = 0%



Σχήμα 3.26: Random Selection = 10%



Σχήμα 3.27: Random Selection = 30%

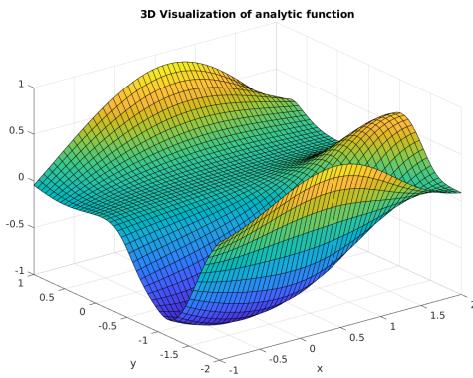


Σχήμα 3.28: Random Selection = 50%

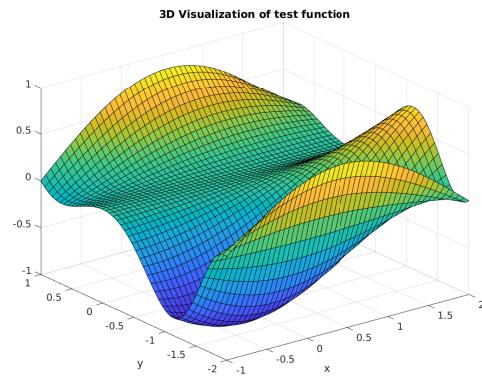
3.7 Επιλεγμένο Χρωμόσωμα

Το τελικώς επιλεγμένο χρωμόσωμα εξήχθη έχοντας θέσει τις παρακάτω παραμέτρους:

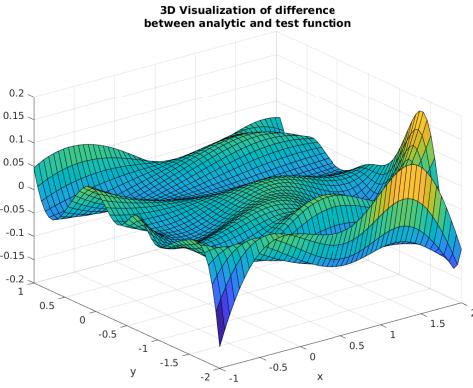
	value
fit_limit	0.001
num_of_gaussians	15
max_generations	20000
mutation_prob	0.1
select_best	0.4
select_random	0.1
num_of_population	100



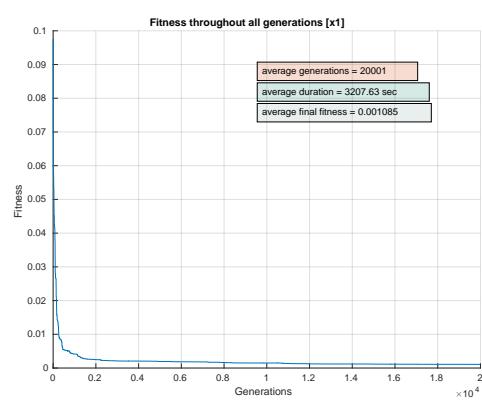
Σχήμα 3.29: $f_{analytic}$



Σχήμα 3.30: f_{real}



Σχήμα 3.31: Σφάλμα - $f_{analytic} - f$



Σχήμα 3.32: Fitness

Από το διάγραμμα 3.31 παρατηρείται η αδυναμία του αλγορίθμου να προσεγγίσει με μεγάλη ακρίβεια την συνάρτηση στα **άκρα του διαστήματος**, όπου και εμφανίζεται και η μεγαλύτερη απόκλιση (≈ 0.2) από την πραγματική τιμή. Ωστόσο μπορούμε να διακρίνουμε ότι η συνάρτηση προσεγγίζεται αρκετά καλά σε όλα τα υπόλοιπα σημεία.

Τα τελικά γονίδια φαίνονται στον παρακάτω πίνακα, **στρογγυλοποιημένα** στο 4ο δεκαδικό ψηφίο. Το επιλεγόμενο χρωμόσωμα χωρίς στρογγυλοποίηση βρίσκεται στο αρχείο `final_selected_chromosome.mat` το οποίο κατατίθεται μαζί με τον αλγόριθμο στο MATLAB.

k	c_{k_1}	c_{k_2}	s_{k_1}	s_{k_2}	A_k
1	1.7144	1.3456	1.1765	0.5044	0.4662
2	0.6556	-1.5436	0.7677	0.2084	-0.6239
3	-0.0304	-3.5174	0.7526	0.9931	0.7463
4	-2.6337	1.9950	1.1692	1.1976	-0.9909
5	0.2742	-1.1875	0.9252	0.2268	-0.2420
6	-0.4648	2.8013	0.5890	0.9231	0.7746
7	-0.5961	-1.2116	1.1393	0.3822	-0.9210
8	0.4028	-3.3510	0.7219	1.0737	0.8411
9	0.2898	-3.2665	0.7600	0.7024	0.8327
10	0.8530	-3.5718	0.4626	0.8622	0.8413
11	-2.1735	-1.8752	1.0737	1.0239	0.3160
12	2.8757	1.0534	0.9060	0.1565	-0.6858
13	0.5760	1.2776	0.8416	0.4242	0.8041
14	-2.4957	-0.6928	0.9664	0.2208	-0.7400
15	1.8321	-1.0769	0.7183	0.3924	0.7606

Πίνακας 3.1: Τελικώς επιλεγόμενο χρωμόσωμα

Στον παραπάνω πίνακα, με k συμβολίζεται ο αριθμός Gaussian συνάρτησης στην αναλυτική έκφραση της f .