Πολυτεχνείο Κρήτης

Τμήμα Ηλεκτρονικών Μηχανικών και Μηχανικών Ηλεκτρονικών Υπολογιστών

Διπλωματική εργασία:

“Βελτιστοποίηση υποστρώματος γονιδιακής ανάλυσης με χρήση γενετικών αλγορίθμων”

Συμεωνίδη Κάτια

Επιβλέπων Καθηγητής:
Ζερβάκης Μιχάλης

Τριμελής Εξεταστική Επιτροπή:
Ζερβάκης Μιχάλης
Σταυρακάκης Γεώργιος
Μπάλας Κωνσταντίνος

Περίληψη (Abstract)

Τα πρότυπα μείγματα αναφοράς έχουν πολλές εφαρμογές στον τομέα της Μοριακής Βιολογίας και κυρίως στις υβριδοποιήσεις σε μικροσυστοιχίες. Η δημιουργία των μειγμάτων αυτών γίνεται κατά κύριο λόγω από διάφορες κυτταρικές σειρές, οι οποίες περιέχουν γονιδιακές πληροφορίες. Ο κύριος λόγος για τον οποίων πραγματοποιούνται υβριδοποιήσεις μεταξύ προτύπων μειγμάτων και μικροσυστοιχιών είναι η εξαγωγή πληροφοριών για τα επίπεδα έκφρασης γονιδίων, καθώς και για τον προσδιορισμό των τμημάτων των γονιδίων που περιέχουν που περιέχουν χρήσιμη πληροφορία. Η δημιουργία μειγμάτων αναφοράς που κατά την υβριδοποίησή τους θα δημιουργούν γονίδια με έντονο σήμα ώστε να είναι εύκολη η περαιτέρω μελέτη τους καθώς και η προτυποποίηση των μειγμάτων αυτών αποτελεί ένα ανοικτό πεδίο έρευνας.

Σκοπός της εργασίας αυτής είναι η δημιουργία ενός μείγματος αναφοράς το οποίο προέρχεται από διάφορες κυτταρικές σειρές, η υβριδοποίηση του οποίου στην μικροσυστοιχία να επιτυγχάνει την μέγιστη δυνατή κάλυψη. Ως ποσοστό κάλυψης θεωρείται το σύνολο των γονιδίων που συμμετέχουν στις κυτταρικές σειρές να δίνει τιμή έντασης πάνω από ένα κατώφλι, το οποίο υπολογίζεται ανάλογα με τα δεδομένα πάνω στα οποία γίνεται η εκάστοτε έρευνα. Ταυτόχρονα, το μείγμα αναφοράς πρέπει να είναι ομοιόμορφο, δηλαδή οι εντάσεις των γονιδίων που συμμετέχουν στο μείγμα να έχουν κοντινές τιμές.

Το πρόβλημα που κληθήκαμε να αντιμετωπίσουμε ανήκει στην κατηγορία προβλημάτων πολυκριτηριακής βελτιστοποίησης (multiobjective optimization) και η μεθοδολογία που χρησιμοποιήσαμε ήταν αυτή των Γενετικών Αλγορίθμων. Μέσω των πειραμάτων που εκτελέσαμε προέκυψαν διάφορα μείγματα αναφοράς, διαφορετικών μεγεθών, και από τα μείγματα αυτά επιλέχθηκε το βέλτιστο, το οποίο αποτελείται από 20 κυτταρικές σειρές, καλύπτοντας τους περιορισμούς της ομοιομορφίας και της μέγιστης κάλυψης.

Table of Contents

Κεφάλαιο 1 – Εισαγωγή....5
1.1 Βιοπληροφορική....5
1.2 Μικροσυστοιχίες DNA....6
1.3 Μέτρηση της κατανομής γονιδιακής έκφρασης....6
1.4 Κατηγορίες μικροσυστοιχιών....9
1.5 Τυποποίηση της πληροφορίας που παράγεται από τα πειράματα με μικροσυστοιχίες....10
1.6 Πρότυπα μείγματα αναφοράς....11
1.7 Σκοπός εργασίας....12
Κεφάλαιο 2 – Περιγραφή Εργαλείων....15
2.1 Προβλήματα βελτιστοποίησης....15
2.2 Αλγόριθμοι Βελτιστοποίησης....16
2.2.1 Ντετερμινιστικοί και μη-Ντετερμινιστικοί αλγόριθμοι (παραδείγματα)....17
....27
2.3 Γενετικοί αλγόριθμοι....28
2.3.1 Ενδεικτική λειτουργία τυπικού ΓΑ....28
2.3.2 Αναπαράσταση υποψηφίων λύσεων....31
2.3.2.1 Δυαδική Κωδικοποίηση....31
2.3.2.2 Αναπαράσταση Ακεραίων....32
2.3.2.3 Δεκαδική Αναπαράσταση....33
2.3.2.3.1 Δυαδική Κωδικοποίηση....33
2.3.2.3.2 Άμεση Κωδικοποίηση....34
2.3.2.4 Αλφαβητική Αναπαράσταση....34
2.3.3 Συνάρτηση καταλληλότητας....35
2.3.4 Διαδικασία επιλογής γονέων....36
2.3.5 Αναπαραγωγή....38
2.3.6 Κριτήρια τερματισμού....40
2.3.6.1 Αριθμός επαναλήψεων....40
2.3.6.2 Προσέγγιση στόχου....41
2.3.6.3 Κριτήριο ομοιότητας....41
2.3.6.4 Διαφορά καλύτερου-χειρότερου....42
2.3.7 Το Πρόβλημα του πλανόδιου πωλητή....43
2.3.8 Τεχνικά εργαλεία....45
Κεφάλαιο 3 – Υλοποίηση....46
3.1 Περιγραφή δεδομένων....47
3.2 Προεπεξεργασία δεδομένων....48
3.2.1 Κατώφλι έκφρασης γονιδίου....48
3.2.2 Απόρριψη γονιδίων....49
3.3 Κανονικοποίηση δεδομένων....50
3.4 Ομοιομορφία....50
3.5 Προσαρμογή των ΓΑ στις ιδιαιτερότητες του προβλήματος....54
3.5.1 Αναπαράσταση υποψήφιων λύσεων ΓΑ....55
3.5.2 Παραγωγή γονιδίων (n-άδων κυτταρικών σειρών)....56
3.5.3 Παραγωγή γενεών....58

3.5.3.1 Πληθυσμός γενιάς (population size) 20 59
3.5.3.2 Πληθυσμός γενιάς (population size) 100.....60
3.5.4 Συναρτήσεις Καταλληλότητας.....61
3.5.4.1 Συνάρτηση ελαχιστοποίησης διασποράς με κατώφλι.....61
3.5.4.2 Συνάρτηση ελαχιστοποίησης διασποράς χωρίς κατώφλι.....62
3.5.4.3 'Υβριδική' συνάρτηση καταλληλότητας.....62
3.5.5 Τερματισμός ΓΑ.....63
Κεφάλαιο 4 – Πειράματα και Αποτελέσματα.....71
4.1 Πειράματα για την εύρεση της ομοιομορφίας.....71
4.1.1 Σύγκριση μεθοδολογιών και ΓΑ.....71
4.1.2 Πειράματα ΓΑ για την ομοιομορφία με χρήση κατωφλίων (αναλυτικά).....77
4.1.3 Σύγκριση ΓΑ με μικρό και μεγάλο πληθυσμό.....78
4.2 ΓΑ με “υβριδική” συνάρτηση καταλληλότητας.....86
4.2.1 Αποτελέσματα με μέγεθος γονιδίου ΓΑ 20.....86
4.2.2 Αποτελέσματα με μεταβαλλόμενο μέγεθος γονιδίου ΓΑ 89
.....92
.....92
4.3 ΓΑ vs. Genetic Algorithm Tool του Matlab.....95
Κεφάλαιο 5 – Συμπεράσματα και μελλοντική εργασία.....98
Βιβλιογραφία.....103
Παράρτημα.....105
Συγκεντρωτικά αποτελέσματα ΓΑ με κατώφλι.....105
Συγκεντρωτικά αποτελέσματα ΓΑ με “υβριδική” συνάρτηση καταλληλότητας.....107
Συγκεντρωτικά αποτελέσματα ΓΑ με “υβριδική” συνάρτηση καταλληλότητας και διαφορετικό μέγεθος n-άδας.....108
Προτεινόμενα μείγματα αναφοράς 200 ΓΑ με μέγεθος 20 κυτταρικές σειρές.....109
Ρουτίνες που προσαρμόζουν το gatool στο πρόβλημα.....112
clear; 112
Πηγαίος Κώδικας Matlab/Octave.....119

Index of Tables

Table 1: Παράδειγμα πληθυσμού χρωμοσωμάτων.....28
Table 2: Αναπαραγωγή χρωμοσωμάτων.....28
Table 3: Αναπαράσταση υποψήφιας λύσης.....34
Table 4: Συγκεντρωτικές τιμές μετρικής ομοιομορφίας για διαφορετικές μεθόδους.....60
Table 5: Συγκεντρωτικός πίνακας ελάχιστων τιμών διασποράς και κατωφλίων που επιτεύχθηκε.....61
Table 6: Μέγιστες, ελάχιστες και μέσες τιμές διασποράς και έκφρασης στα 200 πειράμτα.....68
Table 7: Ελάχιστες, μέγιστες και μέσες τιμές διασποράς και έκφρασης, καθώς και μέσα ποσοστά κάλυψης n-άδων.....71
Table 8: Αποτελέσματα gatool και ΓΑ.....76

Illustration Index

1 Σχηματικό διάγραμμα της αντιγραφής της πληροφορίας από τα γονίδια(DNA) σε αγγελιαφόρο RNA (mRNA) και τη μετάφραση του mRNA σε πρωτεΐνες. Το περιβάλλον και το DNA μεταβάλλουν την παραγωγή των πρωτεϊνών μέσα από ένα πολύπλοκο ανατροφοδοτούμενο δίκτυο. 8
2 Σχηματικό διάγραμμα ενός πειράματος με μικροσυστοιχία cDNA.....10
3 Ταξινόμηση αλγορίθμων βελτιστοποίησης με κριτήριο την τυχαιότητα.....18
4 Κατάταξη ντετερμινιστικών αλγορίθμων βελτιστοποίησης.....27
5 Κατάταξη μη-ντετερμινιστικών (σταχαστικών) αλγορίθμων βελτιστοποίησης.....28
6 Βασικός κύκλος γενετικού αλγορίθμου.....31
7 Ιστόγραμμα κανονικοποιημένων τιμών έντασης των γονιδίων 50
8 Διάγραμμα τιμών διασποράς κάθε γονιδίου στις 83 κυτταρικές σειρές (μη κανονικοποιημένα δεδομένα).....52
9 Διάγραμμα τιμών διασποράς κάθε γονιδίου στις 83 κυτταρικές σειρές σε φθίνουσα σειρά (μη κανονικοποιημένα δεδομένα).....53
10 Διάγραμμα τιμών διασποράς κάθε γονιδίου στις 83 κυτταρικές σειρές (κανονικοποιημένα δεδομένα) 54
11 Διάγραμμα τιμών διασποράς κάθε γονιδίου στις 83 κυτταρικές σειρές σε φθίνουσα σειρά(κανονικοποιημένα δεδομένα).....55
12 Πρόοδος του ΓΑ για 300 γενεές με population size 100.....65
13 Πρόοδος του ΓΑ για 300 γενιές με population size 20.....67
14 Συμμετοχή κυτ. σειρών στην εξέλιξη του ΓΑ (population 100, gene size 20).....68
15 Συμμετοχή κυτ. σειρών στην εξέλιξη του ΓΑ (population 20, gene size 20).....69
16 Διάγραμμα ροής προτεινόμενου αλγορίθμου.....71
17 Σύγκριση μεθόδων για προσδιορισμό ομοιομορφίας.....73
18 Διάγραμμα των καλύτερων σκορ 200 ΓΑ με πληθυσμό 20.....80
19 Οι καλύτερες 20-άδες κυτταρικών σειρών για κάθε ΓΑ με πληθυσμό 20.....81
20 Διάγραμμα των καλύτερων σκορ 200 ΓΑ με πληθυσμό 100.....82
21 Οι καλύτερες 20-άδες κυτταρικών σειρών για πληθυσμό 100.....83
22 Σύγκριση καλύτερων τιμών καταλληλότητας ΓΑ για πληθυσμό 20 και 100 84
23 Συμμετοχή κυτταρικών σειρών στα πειράματα με πληθυσμό 100.....85
24 Συμμετοχή κυτταρικών σειρών στα πειράματα με πληθυσμό 20.....85
25 Σύγκριση καλύτερων κυτταρικών σειρών για πληθυσμό γενιάς 20 και 100 20-άδων 86
26 Οι καλύτερες 20-άδες κυτταρικών σειρών με "υβριδική" συνάρτηση καταλληλότητας.....88
27 Συμμετοχή κυτταρικών σειρών στα 200 "υβριδικά" πειράματα με μέγεθος γονιδίου ΓΑ 20.....89
28 Εξέλιξη της έκφρασης των γονιδίων σε ΓΑ με διαφορετικό μέγεθος γονιδίου.....92
29 Εξέλιξη της διασποράς των γονιδίων σε ΓΑ με διαφορετικό μέγεθος γονιδίου.....93
30 Εξέλιξη του ποσοστού κάλυψης των γονιδίων σε ΓΑ με διαφορετικό μέγεθος γονιδίου.....93
31 Συμμετοχές κυτταρικών σειρών σε διαφορετικά μεγέθη "γονιδίου".....95

Κεφάλαιο 1 – Εισαγωγή

1.1 Βιοπληροφορική

Η Βιοπληροφορική αποτελεί ένα σύγχρονο τομέα έρευνας και ανάπτυξης, τόσο για τους βιολόγους όσο και για τους επιστήμονες της πληροφορικής. Είναι η επιστήμη η οποία παρέχει μεθόδους και εργαλεία τα οποία υποστηρίζουν την ανάγκη για εκμετάλλευση μεγάλης υπολογιστικής ισχύος για εξαγωγή γνώσης από βιολογικά δεδομένα, και επιλύει προβλήματα που προκύπτουν από τη Βιολογία χρησιμοποιώντας μεθοδολογία από την επιστήμη υπολογιστών.
Οι στόχοι της Βιοπληροφορικής μπορούν να ταξινομηθούν σε τρεις ομάδες. Σε πρώτο επίπεδο η Βιοπληροφορική επιτρέπει την αποδοτική οργάνωση των δεδομένων, ώστε να είναι δυνατή η αποθήκευση, ανάκτηση και ενημέρωσή τους (π.χ. η βάση δεδομένων της δομής τρισδιάστατων μορίων Protein Data Bank). Σε δεύτερο επίπεδο η Βιοπληροφορική περιλαμβάνει εργαλεία τα οποία επιτρέπουν την ανάλυση βιολογικών δεδομένων. Σε τρίτο επίπεδο θέτει ως στόχο την ανάπτυξη εργαλείων με σκοπό την ερμηνεία αποτελεσμάτων βιολογικής σημασίας. Οι κυριότερες εφαρμογές της Βιοπληροφορικής σήμερα αφορούν

- Υλοποίηση και σχεδιασμό υπολογιστικών εργαλείων για αυτόματη ανάκτηση γνώσης από Βάσεις Βιολογικών Δεδομένων
- Ανάλυση ακολουθιών Βιολογικών Δεδομένων
- Κατηγοριοποίηση Βιολογικών Δεδομένων
- Μοριακή μοντελοποίηση
- Ανάλυση πρωτεϊνών
- Σχεδιασμό φαρμάκων με βοήθεια ηλεκτρονικού υπολογιστή

Οι τεχνικές από τον χώρο της πληροφορικής που χρησιμοποιούνται είναι η τεχνολογία Βάσεων Δεδομένων για την οργάνωση, αποθήκευση και ανάκτηση βιολογικών δεδομένων, τεχνικές επεξεργασίας συμβολοσειρών (strring manipulation techniques) για την ανάλυση ακολουθιών, τεχνικές μηχανικής μάθησης και εξόρυξης δεδομένων (data mining) στην ανακάλυψη προτύπων και τέλος

αλγορίθμους τρισδιάστατων συγκρίσεων στην ανάλυση τρισδιάστατης δομής βιολογικών μορίων. Στις περισσότερες περιπτώσεις οι υπολογιστικές τεχνικές ενσωματώνουν και στατιστικούς ελέγχους των αποτελεσμάτων. [1]

1.2 Μικροσυστοιχίες DNA

Την τελευταία δεκαετία η ερευνητική κοινότητα που ασχολείται με το ανθρώπινο και όχι μόνο γονιδίωμα έχει στρέψει την προσοχή της σε τεχνολογίες με πολύ υψηλούς ρυθμούς απόδοσης ως προς τον όγκο πληροφορίας που χειρίζονται. Μία από αυτές τις τεχνολογίες είναι οι μικροσυστοιχίες DNA (DNA) εξαιτίας του μεγάλου πλήθους πειραμάτων, της παράλληλης χρήσης τους, καθώς του πλούτου των πληροφοριών που παρέχουν. Για τον λόγο αυτό έρχονται και σε αντίθεση με τις παραδοσιακές μεθόδους ανάλυσης γονιδιωμάτων, οι οποίες μπορούν να αναλύσουν έναν μικρό αριθμό γονιδίων τη φορά.

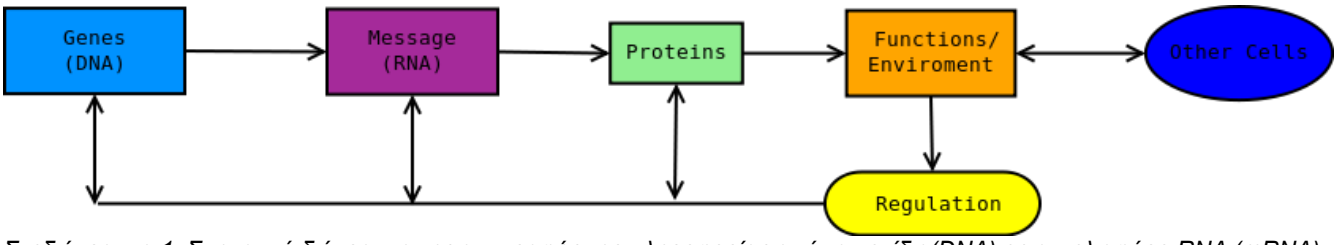
Οι μικροσυστοιχίες DNA χρησιμοποιούνται κυρίως για τη μέτρηση επιπέδου έκφρασης γονιδίων, δηλαδή θέτουν την μετρική της διαδικασίας μεταγραφής του DNA σε mRNA(messenger RNA). Η πληροφορία που μεταγράφεται σε mRNA με τη σειρά της μεταφράζεται σε πρωτεΐνες, τα μόρια που εκτελούν τις περισσότερες λειτουργίες των κυττάρων. Για το λόγο αυτό, μέσω της μέτρησης του επιπέδου έκφρασης των γονιδίων, οι επιστήμονες μπορούν να συμπεραίνουν κρίσιμες πληροφορίες για την λειτουργία των κυττάρων ή και ολόκληρου του προς εξέταση οργανισμού. Μία απόκλιση από το συνηθισμένο επίπεδο έκφρασης πολύ συχνά αποτελεί ένδειξη ασθένειας, καταστώντας τα πειράματα με μικροσυστοιχίες DNA πολύτιμο εργαλείο της έρευνας των γενετικών αιτιών των ασθενειών. Άλλωστε ένας από τους απώτερους σκοπούς της έρευνας και της ανάπτυξης αυτής της τεχνολογίας είναι η διάγνωση ασθενειών σε μοριακό επίπεδο και η δημιουργία εξατομικευμένων φαρμάκων. [2], [3]

1.3 Μέτρηση της κατανομής γονιδιακής έκφρασης

Τα κύτταρα παράγουν τις πρωτεΐνες που χρειάζονται για τη σωστή λειτουργία τους μετά από επίδραση εσωτερικών ή εξωτερικών ερεθισμάτων. Οι πρωτεΐνες είναι ταινίες αμινοξέων που συντίθεται στα ριβοσώματα με μία διαδικασία που καθορίζεται από το DNA και τα αντίστοιχα σε αυτά γονίδια. Το DNA βρίσκεται στον πυρήνα των κυττάρων και διαβιβάζεται στα ριβοσώματα με τη μορφή του mRNA. Τρεις διαδοχικές βάσεις, που καλούνται condon, καθορίζουν ποια από τα είκοσι αμινοξέα θα

χρησιμοποιηθούν για την σύνθεση της αλυσίδας των αμινοξέων που θα αποτελέσουν την πρωτεΐνη. Συνοπτικά η διαδικασία παραγωγής των πρωτεϊνών έχει ως εξής:

1. μεταγραφή των αντίστοιχων γονιδίων DNA σε mRNA
2. μετάφραση των μορίων του mRNA στις αντίστοιχες πρωτεΐνες και στην ποσότητα που χρειάζεται ο οργανισμός



Σχεδιάγραμμα 1: Σχηματικό διάγραμμα της αντιγραφής της πληροφορίας από τα γονίδια(DNA) σε αγγελιαφόρο RNA (mRNA) και τη μετάφραση του mRNA σε πρωτεΐνες. Το περιβάλλον και το DNA μεταβάλλουν την παραγωγή των πρωτεϊνών μέσα από ένα πολύπλοκο ανατροφοδοτούμενο δίκτυο.

Οι μικροσυστοιχίες παίρνουν ένα στιγμιότυπο της δραστηριότητας του κυττάρου, μετρώντας υπό ορισμένες συνθήκες τη συγκέντρωση των αντιγράφων του mRNA, το οποίο με τη σειρά του μας δίνει μία έμμεση εικόνα της δραστηριότητας των πρωτεϊνών. Ύστερα από έκθεση του κυττάρου σε κάποια εξωτερική διαταραχή, η γενετική δραστηριότητα μεταβάλλεται, καθώς κάποιες διαδικασίες σταματούν, άλλες αρχίζουν και κάποιες μένουν αμετάβλητες. Ανάλογα με τα ερεθίσματα “ενεργοποιούνται” και διαφορετικά γονίδια, μεταγράφονται σε mRNA και τελικά έχουμε την παραγωγή διαφορετικών πρωτεϊνών. Τα κύτταρα, δηλαδή, αλληλεπιδρούν με το περιβάλλον αλλά και με τα υπόλοιπα κύτταρα.

Αποκλίσεις γονιδίων από τα “φυσιολογικά” είναι δυνατόν να προήλθαν τόσο από κληρονομικότητα, όσο και από έκθεση σε περιβαλλοντικούς παράγοντες. Η αλλαγή σε μία και μόνο βάση, στις περισσότερες περιπτώσεις, δεν αλλάζει απαραιτήτως την λειτουργία μίας πρωτεΐνης, όπως επίσης αλλάζοντας ένα αμινοξύ σε μία πρωτεΐνη, δεν αλλάζει απαραιτήτως η λειτουργία της πρωτεΐνης. Κοιτάζοντας όμως ανάποδα, μπορούμε να πούμε ότι υπάρχει η περίπτωση σοβαρών επιπτώσεων στον οργανισμό. Ενδεικτικό παράδειγμα είναι η δρεπανοκυτταρική αναιμία, η οποία προκαλείται από μία αλλαγή σε μία βάση, ενός κατά τα άλλα φυσιολογικού DNA.

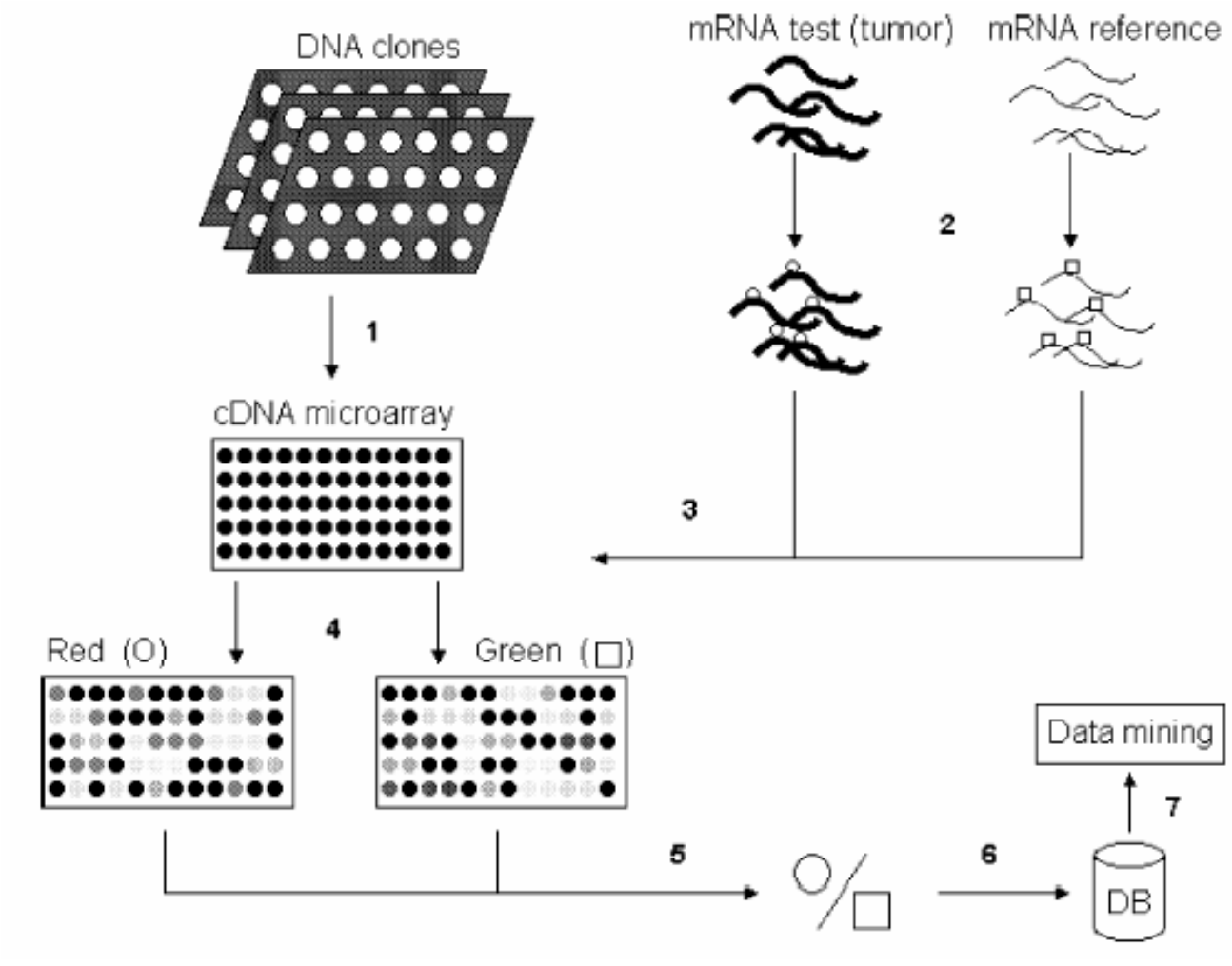
Πριν προχωρήσουμε στην περαιτέρω ανάλυση των μικροσυστοιχιών είναι απαραίτητο να αναφερθούμε στην ιδιότητα του υβριδισμού της διπλής έλικας του DNA. Όταν μία μονή κορδέλα DNA έρχεται σε επαφή με την συμπληρωματική της ακολουθία DNA, ενώνεται με αυτήν και σχηματίζει μία

διπλή κορδέλα DNA. Η διαδικασία αυτή καλείται υβριδισμός. Επειδή οι δύο μονές κορδέλες DNA έχουν αντίθετους προσανατολισμούς, η συμπληρωματική ακολουθία παράγεται συμπληρώνοντας τις βάσεις από το τέλος της κορδέλας αναφοράς προς την αρχή. Για το λόγο αυτό ο υβριδισμός επιτρέπει σε ένα "ερευνητικό" DNA να αναγνωρίσει ένα αντίγραφο της συμπληρωματικής του ακολουθίας που αποκτήθηκε από ένα βιολογικό δείγμα.

Μία μικροσυστοιχία αποτελείται από υποδείγματα που αναπαράχθηκαν από διάφορα DNA δείγματα προσκολλημένα σε ένα στερεό υποστήριγμα. Μετά την εξαγωγή του RNA από ένα βιολογικό δείγμα, ετοιμάζονται φθορίζοντα μαρκαρισμένα υποδείγματα συμπληρωματικού DNA (cDNA) ή συμπληρωματικού RNA (cRNA). Στη συνέχεια το φθορίζων δείγμα υβριδίζεται με το υπάρχον DNA στο πλακίδιο της μικροσυστοιχίας. Με την διαδικασία φθορισμού η ένταση του υβριδισμού, η οποία είναι ανάλογη του αριθμού των αντιγράφων κάθε είδους RNA που υπάρχει στο δείγμα, μπορεί να μετρηθεί με έναν σαρωτή laser και να μετατραπεί σε μία ποσοτική τελικά έξοδο. Με τον τρόπο αυτό οι μικροσυστοιχίες επιτρέπουν τη σύγχρονη μέτρηση της ομοιομορφίας γονιδιακών εκφράσεων χιλιάδων γονιδίων σε ένα και μόνο πείραμα υβριδισμού. [4], [5], [6], [7]

Σχηματικά, η περιγραφή ενός πειράματος μικροσυστοιχιών είναι η εξής:

1. Τοποθέτηση του προ-συντιθεμένου DNA που πρόκειται να ερευνηθεί (αυτό το DNA παρήχθηκε από τα γονίδια που θέλουμε να μελετήσουμε)
2. Μαρκάρισμα του συνόλου των mRNA του δείγματος δοκιμής (κόκκινο κανάλι) και του δείγματος αναφοράς (πράσινο κανάλι)
3. Εισαγωγή των δύο δειγμάτων και υβριδισμός
4. Παίρνουμε για κάθε κουκκίδα τις εντάσεις του φθορίζοντος υλικού ξεχωριστά για το κάθε κανάλι (στην ουσία μετράμε το ποσοστό του υβριδισμού στα δείγματα)
5. Υπολογισμός της σχετικής ομοιομορφίας των γονιδιακών εκφράσεων
6. Αποθήκευση των αποτελεσμάτων σε βάση δεδομένων
7. Εξόρυξη δεδομένων



Σχεδιάγραμμα 2: Σχηματικό διάγραμμα ενός πειράματος με μικροσυστοιχία cDNA

[3], [20]

1.4 Κατηγορίες μικροσυστοιχιών

Οι μικροσυστοιχίες δημιουργούνται όπως είδαμε από παρατάσσοντας δείγματα έρευνας πάνω σε έναν πίνακα και κάνοντας χρήση της ιδιότητας ζευγαρώματος των βάσεων του DNA, έτσι ώστε να συνδεθούν οι συμπληρωματικοί "στόχοι" μόρια του DNA και του mRNA. Διάφορες τεχνολογίες έχουν αναπτυχθεί για την παραγωγή μικροσυστοιχιών. Το πλακίδιο το οποίο συνήθως χρησιμοποιείται είναι γυαλί, nylon ή σιλικόνη. Τα προς έρευνα δείγματα μπορεί να είναι DNA, cDNA ή ολιγονουκλεοτίδια και αποτυπώνονται πάνω στο πλακίδιο με τη βοήθεια της ρομποτικής ή με την τεχνική της φωτολιθογραφίας ή με την τεχνική έκχυσης μελάνης. Ακόμη, μπορούν να κατηγοριοποιηθούν

περαιτέρω στις παρακάτω κατηγορίες:

1. Μικροσυστοιχίες cDNA: για την διευκόλυνση μεγάλης κλίμακας ανάλυσης της mRNA αφθονίας ως δείκτη της έκφρασης γονιδίου
2. Μικροσυστοιχίες SNP (Single Nucleotide Polymorphirsm) & mutation arrays (μετάλλαξη): για την ανίχνευση πολυμορφισμών ή μεταλλάξεων σε έναν πληθυσμό χρησιμοποιώντας SNP arrays ή παρατάξεις που είναι σχεδιασμένες για την ανίχνευση γνωστών μεταλλάξεων
3. Συγκριτικός Γονιδιακός Υβριδισμός (Comparative Genomic Hybridizition – CGH): για την έρευνα "απώλειας" ή "κέρδους", ή αλλαγές στον αριθμό αντιγραφής κάποιου συγκεκριμένου γονιδίου, που σχετίζεται με κάποια ασθένεια

Οι μικροσυστοιχίες cDNA είναι παραδείγματα "αντιγραφικής" παράθεσης, ενώ οι μικροσυστοιχίες SNP και οι μικροσυστοιχίες CGH είναι "γονιδιακές" παραθέσεις. Κάθε μία από τις παραπάνω κατηγορίες μπορεί να χρησιμοποιηθεί σε διαφορετικές περιπτώσεις, αναλύοντας δεκάδες χιλιάδες ακολουθίες DNA ή RNA σε ένα και μόνο υψηλής ανάλυσης πείραμα. [3], [20]

1.5 Τυποποίηση της πληροφορίας που παράγεται από τα πειράματα με μικροσυστοιχίες

Στον τομέα της τεχνολογίας των μικροσυστοιχιών, όπως και σε άλλους τομείς της βιολογίας και κατ' επέκταση της βιοπληροφορικής, δημιουργείται σύγχυση από την χρήση διαφορετικών μονάδων μέτρησης από διάφορους ερευνητές, για τον ίδιο σκοπό. Αυτό έχει σαν αποτέλεσμα η αύξηση ή η μείωση της γονιδιακής έκφρασης που παρατηρείται ανάμεσα σε δύο εργαστήρια μπορεί αν μην είναι η ίδια και να είναι αποτέλεσμα των διαφορετικών πρωτοκόλλων και μεθόδων ανάλυσης δεδομένων. Χωρίς έναν κοινό κανόνα είναι αδύνατον να κρίνει κανείς την ισχύ των αποτελεσμάτων, τόσο στην μέτρηση της γονιδιακής έκφρασης όσο και στην επιθεώρηση των "ωμών δεδομένων" (raw data). Για τον λόγο αυτό είναι επιτακτική η ανάγκη προτυποποίησης τόσο στην αναπαράσταση της ελάχιστης πληροφορίας που χρειάζεται να καταγράφεται για την αξιόπιστη παραγωγή και επαναληψιμότητα πειραμάτων, όσο και για την επιβεβαίωση της ισχύς του πειράματος και για την διευκόλυνση συγκρίσεων μεταξύ παρόμοιων πειραμάτων. [3], [11]

1.6 Πρότυπα μείγματα αναφοράς

Όπως είδαμε και παραπάνω, οι μικροσυστοιχίες χρησιμοποιούνται κυρίως για καταγραφή εκφράσεων γονιδίων και για συγκριτική υβριδοποίηση γονιδιώματος. Οι διαδικασίες αυτές απλοποιούνται και τα αποτελέσματά τους είναι πιο "ισχυρά" αν υπάρχει ένα κοινό πρότυπο μείγμα αναφοράς για την εκτέλεση των πειραμάτων. Ταυτόχρονα, το μείγμα αυτό θα πρέπει να μπορεί να παράγεται σε μεγάλες ποσότητες. Διάφορες έρευνες έχουν διεξαχθεί προς τον σκοπό αυτό. Η καταλληλότητα των ερευνών αυτών πιστοποιείται υπολογίζοντας το ποσοστό κάλυψης της μικροσυστοίχιας. Ως ποσοστό κάλυψης θεωρούμε το σύνολο των γονιδίων που βρίσκονται πάνω από ένα κατώφλι. Το κατώφλι αυτό υπολογίζεται ανάλογα με τα δεδομένα πάνω στα οποία έχει γίνει η εκάστοτε έρευνα. Η βασική ιδέα για την κατασκευή πρότυπων μειγμάτων αναφοράς είναι ότι συνδυάζοντας RNA που προέρχεται από περιορισμένες σε αριθμό, αλλά διαφορετικές κυτταρικές σειρές μπορούμε να καταλήξουμε σε βέλτιστα πρότυπα μείγματα. Η χρήση κυτταρικών σειρών γίνεται γιατί οι κυτταρικές σειρές αποτελούν ανανεώσιμη πηγή μεγάλων ποσοτήτων RNA. Κάθε κυτταρική σειρά εκφράζει μια μεγάλη ποικιλία τύπων RNA, με αποτέλεσμα να παρέχει ένα μετρήσιμο, βασικό υβριδισμό για ένα υποσύνολο γονιδίων σε ένα array. Οι κυτταρικές σειρές προέρχονται από μία ποικιλία βασικών ιστών και το RNA κατασκευάζεται από ένα μεγάλο μέγεθος (παραπάνω από 10) τέτοιων κυτταρικών σειρών. [3], [9], [11]

Ένα ευρέως χρησιμοποιούμενο πρότυπο μείγμα αναφοράς, εμπορικά διαθέσιμο, που χρησιμοποιείται συχνά σε πειράματα μικροσυστοιχιών είναι το διεθνές μείγμα ανθρώπινου RNA της Sratagene. Πρόκειται για για μείγμα που αποτελείται από 10 έως 14 κυτταρικές σειρές. Παράλληλα, έχουν γίνει προσπάθειες να κατασκευαστούν και άλλα πρότυπα μείγματα. Μια τέτοια προσπάθεια έγινε το 2002, όπου κατασκευάστηκαν δύο πρότυπα μείγματα αναφοράς αποτελούμενα από τρεις κυτταρικές σειρές το καθένα, σε ίσες αναλογίες, οι οποίες έχουν την μεγαλύτερη αναπαράσταση μοναδικών γονιδίων (κριτήριο "μοναδικότητας" γονιδίου είναι ένταση πάνω από 1000 και η ύπαρξη τέτοιας τιμής μόνο σε μία κυτταρική σειρά), τα οποία παρέχουν μεγάλες ποσότητες RNA. Στο πρώτο μείγμα αναμείχθηκαν δύο κυτταρικές σειρές που περιγράφουν καρκινώματα του παχέως εντέρου (CaCO2, KM12L4A) και μία κυτταρική σειρά που περιγράφει καρκίνωμα των ωοθηκών (OVCAR3), ενώ στο δεύτερο η OVCAR3 αντικαταστάθηκε από την σειρά U118MG, που περιγράφει καρκίνωμα του εγκεφάλου. Το

πρώτο δείγμα είχε ποσοστό κάλυψης 75%, συγκρίσιμο δηλαδή με το μείγμα της Stratagene, ενώ το δεύτερο 80%. Ενώ στο πείραμα που περιγράφηκε παραπάνω οι αναλογίες πρόσμιξης των κυτταρικών σειρών είναι 1:1, υπάρχουν έρευνες στις οποίες οι αναλογίες κυτταρικών σειρών είναι μεγαλύτερες. [12]

Τα πειράματα με μικροσυστοιχίες μπορούν να διεξαχθούν και χωρίς την χρήση πρότυπων μειγμάτων αναφοράς, αλλά με mRNA το οποίο παράγεται από τα ίδια τα κύτταρα του οργανισμού που τίθεται προς εξέταση. Η διαδικασία αυτή λέγεται self-self hybridization. Η τάση της επιστημονικής κοινότητας προς τα κοινά πρότυπα μείγματα αναφοράς έχει να κάνει με το γεγονός ότι στην περίπτωση του self-self hybridization τα αποτελέσματα που παράγονται δεν μπορούν να χρησιμοποιηθούν για συγκριτικές μελέτες μεταξύ διαφορετικών εργαστηριακών κέντρων, αφού παράγονται τοπικά και σε μικρή κλίμακα. Επίσης, ενώ μπορεί να έχουν καλές αποδόσεις σε ένα συγκεκριμένο πείραμα, η χρήση τους σε ευρύτερη κλίμακα πειραμάτων δεν έχει αποδειχθεί δόκιμη. [12]

1.7 Σκοπός εργασίας

Χρησιμοποιώντας ως μείγμα αναφοράς ένα μείγμα που προήλθε από μία κυτταρική σειρά, το δείγμα ελέγχου δεν παρέχει σήμα για όλες τις κηλίδες DNA που εξετάζονται, όπως προκύπτει και από την παραπάνω μελέτη. Για να ξεπεραστεί αυτό το πρόβλημα χρησιμοποιούνται μείγματα αναφοράς που προέρχονται από διάφορες κυτταρικές ώστε να παρέχουν τη μέγιστη κάλυψη. Όπως αναφέρθηκε και παραπάνω η μέγιστη κάλυψη δεν προέρχεται από τον αριθμό των κυτταρικών σειρών που συμμετέχουν στο μείγμα αναφοράς, αλλά από την δομή του μείγματος, δηλαδή από το ποιες είναι αυτές οι κυτταρικές σειρές.

Σκοπός της εργασίας αυτής είναι να δημιουργήσουμε ένα μείγμα αναφοράς, το οποίο προέρχεται από διάφορες κυτταρικές σειρές, η υβριδοποίηση του οποίου στη μικροσυστοιχία να επιτυγχάνει την μέγιστη κάλυψη. Οι κυτταρικές σειρές που χρησιμοποιούμε είναι 83 και αποτελούνται η καθεμία από 22284 γονίδια. Κάνοντας χρήση γενετικών αλγορίθμων θα επιλέξουμε τις κυτταρικές σειρές για τις οποίες η πλειοψηφία των γονιδίων δίνει σήμα πάνω από ένα κατώφλι, το οποίο θα εξαγάγουμε, ενώ παράλληλα θέλουμε τα σήματα των γονιδίων να είναι όσο το δυνατόν πιο ομοιόμορφα. Η ανάμιξη των κυτταρικών σειρών θα γίνει με αναλογία 1 προς 1.

Το πρόβλημα το οποίο καλούμαστε να επιλύσουμε μπορεί να περιγραφεί ως η εύρεση ολικού μεγίστου δύο συναρτήσεων. Αναλυτικά:

Η πρώτη συνάρτηση είναι η επιλογή κυτταρικών σειρών τέτοιων ώστε η ένταση των γονιδίων τους να είναι πάνω ένα threshold, το οποίο και θα υπολογίσουμε.

$f_{expression}: A \rightarrow R^1$, όπου το A είναι οι διατάξεις $\binom{l}{k}$, l είναι το πλήθος των κυτταρικών σειρών που διαθέτουμε και k είναι το πλήθος των κυτταρικών σειρών που περιέχει ένα πιθανό μείγμα αναφοράς.

Η δεύτερη συνάρτηση είναι η επιλογή κυτταρικών σειρών των οποίων οι τιμές των γονιδίων τους να είναι όσο το δυνατόν πιο ομοιόμορφες.

$f_{std}: A \rightarrow R^1$, όπου το A είναι οι διατάξεις $\binom{l}{k}$, l είναι το πλήθος των κυτταρικών σειρών που διαθέτουμε και k είναι το πλήθος των κυτταρικών σειρών που περιέχει ένα πιθανό μείγμα αναφοράς.

Δοθέντος των δύο συναρτήσεων $f_{expression}$ και f_{std} , να βρεθεί το στοιχείο $x = \{x_0, x_1, \ldots, x_k\} \in A$, τέτοιο ώστε $f_{expression}(x) \geq f_{expression}(x'), \forall x' \in A$ και αντίστοιχα $1/(1+f_{std}(x)) \geq 1/(1+f_{std}(x')), \forall x' \in A$.

Η εύρεση ολικού μεγίστου, όπως περιγράφεται παραπάνω, μας παραπέμπει σε προβλήματα βελτιστοποίησης.

Κανονικοποιώντας τις εντάσεις των όλων των γονιδίων μέσα στις 83 κυτταρικές σειρές, μπορέσαμε με χρήση ιστογράμματος των τιμών αυτών να προσδιορίσουμε το κατώφλι έκφρασης των γονιδίων στην τιμή έντασης 100. Στη συνέχεια, και αφού αφαιρέσαμε τα γονίδια εκείνα που σε καμία από τις κυτταρικές σειρές δεν έχουν τιμή ίση ή μεγαλύτερη από το κατώφλι, εφαρμόσαμε Γενετικούς Αλγορίθμους για να δημιουργήσουμε τα υποψήφια μείγματα. Τα μείγματα τα οποία δημιουργήσαμε ήταν διαφορετικών μεγεθών, σε μία προσπάθεια να βρούμε το πιο κατάλληλο με βάση τους περιορισμούς του προβλήματος, δηλαδή μεγάλο ποσοστό έκφρασης και ταυτόχρονα υψηλή τιμή ομοιομορφίας. Το βέλτιστο μείγμα στο οποίο καταλήξαμε αποτελείται από 20 κυτταρικές σειρές, παρέχοντας ποσοστό κάλυψης 82% στα δεδομένα που έχουμε μετά την απόρριψη των γονιδίων που δεν εκφράζονται σε καμία κυτταρική σειρά και 62.77% στο σύνολο των δεδομένων και μέση τιμή διασποράς 0.4431.

Όπως εμείς με την εργασία αυτή κάναμε μία προσπάθεια για να προτείνουμε ένα βέλτιστο μείγμα

αναφοράς, έχουν γίνει και εξακολουθούν να γίνονται αντίστοιχες προσπάθειες, είτε με χρήση υπολογιστικών μεθόδων είτε με χρήση πειραματικών διατάξεων σε εργαστήρια. Μία αντίστοιχη προσπάθεια για την βελτιστοποίηση πρότυπου μείγματος αναφοράς με χρήση υπολογιστικών μεθόδων αφορούσε 22 κυτταρικές σειρές στις οποίες συμμετείχαν 34772 γονίδια και η ανάμειξη των σειρών αυτών έγινε με αναλογία 1 προς 1. Η υπολογιστική μέθοδος που χρησιμοποιήθηκε ήταν η simulated annealing και το βέλτιστο μείγμα που προτάθηκε αποτελείται από 11 κυτταρικές σειρές, παρέχοντας ποσοστό κάλυψης 57.12%.[24] Η χρήση διαφορετικών δεδομένων και η έλλειψη προτυποποίησης σε αυτόν τον τομέα της έρευνας, όπως αναφέραμε προηγουμένως, δυστυχώς δεν μας επιτρέπει μια άμεση και απόλυτα τεκμηριωμένη σύγκριση αποτελεσμάτων. Παρόλα αυτά, θεωρούμε ότι τα αποτελέσματά μας είναι ικανοποιητικά, μιας και καλύπτουν σε μεγάλο βαθμό τις απαιτήσεις και ιδιότητες του μείγματος, όπως αυτές παρουσιάζονται στον σκοπό της εργασίας.

Κεφάλαιο 2 – Περιγραφή Εργαλείων

2.1 Προβλήματα βελτιστοποίησης

Στα μαθηματικά και τη επιστήμη υπολογιστών, ο όρος βελτιστοποίηση (optimization) αναφέρεται στο πρόβλημα της επιλογής του καλύτερου στοιχείου από ένα σύνολο διαθέσιμων πιθανών λύσεων. Στην απλούστερη των περιπτώσεων, αναφέρεται σε προβλήματα ελαχιστοποίησης ή μεγιστοποίησης πραγματικών συναρτήσεων, μέσω συστηματικής επιλογής τιμών πραγματικών ή ακεραίων μεταβλητών, μέσα από ένα επιτρεπόμενο σύνολο. Γενικεύοντας την παραπάνω περίπτωση, ανφέρεται στην εύρεση των "καλύτερων διαθέσιμων" τιμών μιας αντικειμενικής συνάρτησης (objective function), δοθέντος ενός ορισμένου πεδίου και περιέχοντας μια ποικιλία διαφορετικών τύπων αντικειμενικών συναρτήσεων και τύπων πεδίων.

Ορισμός 2.1 (Objective Function) : Μία αντικειμενική συνάρτηση (objective function) $f : X \rightarrow Y$, με $Y \subset R$ είναι μία μαθηματική συνάρτηση που υπόκειται σε βελτιστοποίηση.

Ένα πρόβλημα βελτιστοποίησης μπορεί να αναπαρασταθεί με τον ακόλουθο τρόπο:

Δοθέντος: μίας συνάρτησης $f : A \rightarrow R$, από κάποιο σύνολο A στους πραγματικούς αριθμούς

Να βρεθεί: ένα στοιχείο x_0 στο A, τέτοιο ώστε $f(x_0) \leq f(x)$, για όλα τα x στο A (ελαχιστοποίηση), ή τέτοιο ώστε $f(x_0) \geq f(x)$, για όλα τα x στο A (μεγιστοποίηση)

Ένας πιο τυπικός ορισμός για το πρόβλημα βελτιστοποίησης A είναι ένα τετραδικό σύνολο (I,f,m,g), όπου:

- I είναι ένα σύνολο από περιπτώσεις (instances)
- δεδομένου ενός instance x που ανήκει στο I, f(x) είναι ένα σύνολο εφικτών (feasible) λύσεων
- δεδομένου ενός instance x και μιας εφικτής λύσης y του x, m(x,y) συμβολίζει τη διάσταση του x, που είναι συνήθως ένας πραγματικός αριθμός
- g είναι η συνάρτηση "στόχου" (goal function), συνήθως ελαχιστοποίηση (min) ή μεγιστοποίηση (max)

Κατόπιν ο σκοπός είναι να βρεθεί για κάποιο instance x μια βέλτιστη λύση, η οποία να ανήκει στις

εφικτές λύσεις y, με

$$m(x,y)=g\{m(x,y'), \text{ όπου } \quad y \in f(x) \quad \}$$

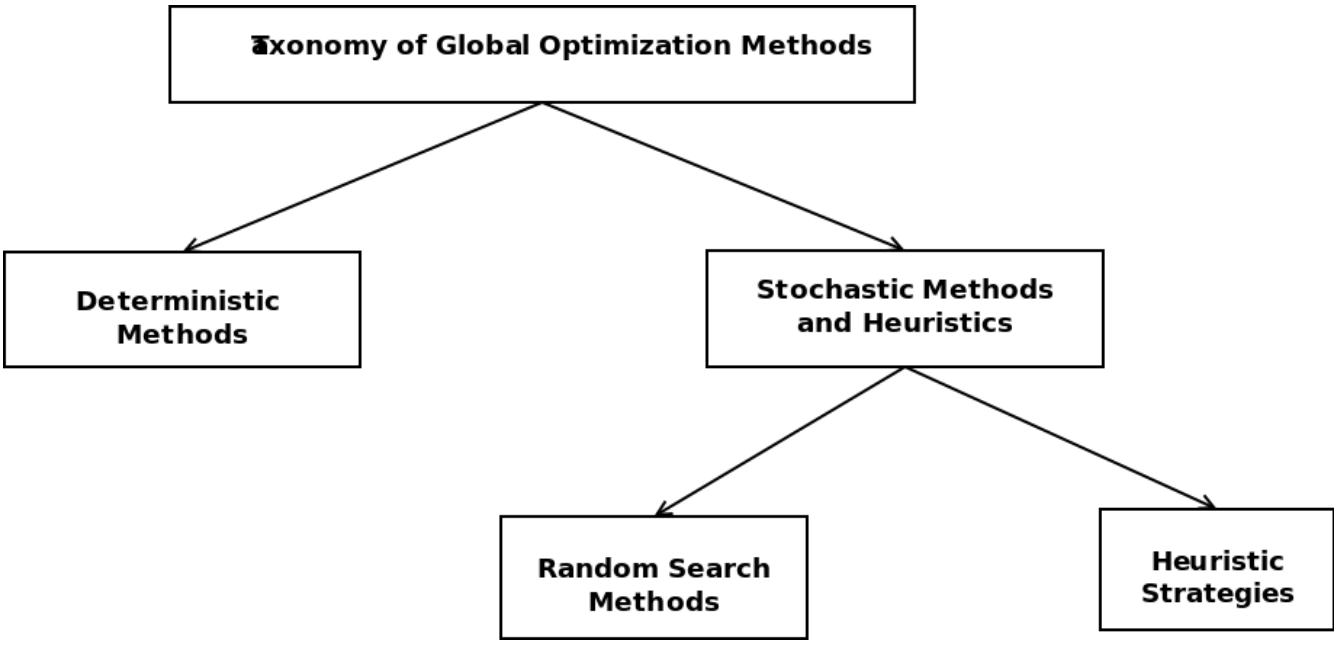
[21]

2.2 Αλγόριθμοι Βελτιστοποίησης

Ο χώρος ενός προβλήματος είναι το σύνολο όλων των πιθανών λύσεων. Σε πολλά προβλήματα για να έχουμε βέβαιη βελτιστοποίηση ο μόνος τρόπος να το επιτύχουμε, είναι να “σαρώσουμε” όλες τις πιθανές λύσεις. Ακόμα και σε πολύ απλά προβλήματα η εξαντλητική αυτή διερεύνηση είναι αδύνατη, λόγω απαγορευτικού υπολογιστικού κόστους. Οι περισσότερες μεθοδολογίες βελτιστοποίησης “δοκιμάζουν” ένα υποσύνολο, κατά πού μικρότερο του αρχικού χώρου, προσπαθώντας να συγκεράσουν τις δύο αντιφατικές ανάγκες, της ακρίβειας (ή βεβαιότητας) και της ταχύτητας επίλυσης.

Οι αλγόριθμοι βελτιστοποίησης μπορούν να κατηγοριοποιηθούν σε δύο βασικές κλάσεις, σύμφωνα με την τυχαιότητα, τους ντετερμινιστικούς (deterministic) και μη (probabilistic) αλγορίθμους. Οι ντετερμινιστικοί αλγόριθμοι χρησιμοποιούνται κυρίως όταν υπάρχει μία λιτή και ευδιάκριτη σχέση ανάμεσα στα χαρακτηριστικά των πιθανών λύσεων και την χρήση τους στο προς επίλυση πρόβλημα. Τότε, το πεδίο έρευνας μπορεί να εξερευνηθεί αποτελεσματικά με την χρήση, παραδείγματος χάριν, ενός “διαίρει και βασίλευε” σχήματος. Αντιθέτως, αν η σχέση ανάμεσα σε μία υποψήφια λύση και στην συνάρτηση καταλληλότητάς της δεν είναι προφανής, εν γένει είναι πολύπλοκη ή οι διαστάσεις του χώρου αναζήτησης (πεδίο ορισμού της fitness) είναι πολλές, προτιμούμε μη-ντετερμινιστικές (probabilistic) μεθόδους. [13]

Οι Ευριστικές Μέθοδοι (Heuristic Methods) ξεκίνησαν να αναπτύσσονται την δεκαετία του '80 ως ένα νέο ευρύ φάσμα νέων προσεγγίσεων στα προβλήματα βελτιστοποίησης, οι οποίες εξαρτώνται σε μεγάλο βαθμό στην ταχύτητα των υπολογιστών. Ορισμένες από αυτές τις μεθόδους είναι εμπνευσμένες από προσεκτικές παρατηρήσεις των φυσικών φαινομένων και κάποιες από αυτές έχουν αναπτυχθεί με την απλή εφαρμογή πρακτικών ιδεών και αποτελούν ένα υποσύνολο των μη-ντετερμινιστικών αλγορίθμων βελτιστοποίησης.



Σχεδιάγραμμα 3: Ταξινόμηση αλγορίθμων βελτιστοποίησης με κριτήριο την τυχαιότητα

2.2.1 Ντετερμινιστικοί και μη-Ντετερμινιστικοί αλγόριθμοι (παραδείγματα)

Το βασικό χαρακτηριστικό των ντετερμινιστικών αλγορίθμων είναι η προβλεπόμενη συμπεριφορά τους. Δοσμένης μίας δεδομένης εισόδου, θα παράγουν συνεχώς το ίδιο αποτέλεσμα, και ο μηχανισμός λειτουργίας τους θα περνάει συνέχεια από τις ίδιες καταστάσεις. Αντιθέτως, οι μη-ντετερμινιστικοί αλγόριθμοι είναι αλγόριθμοι που περιέχουν έναν βαθμό τυχαιότητας σαν μέρος της λογικής τους. Ένας μη-ντετερμινιστικός αλγόριθμος τυπικά χρησιμοποιεί ομοιόμορφα τυχαία bits σαν μία “βοηθητική” είσοδο για την καθοδήγηση της συμπεριφοράς του, με την ελπίδα να επιτύχει καλή απόδοση στην “μέση περίπτωση” για όλες τις πιθανές επιλογές των τυχαίων bits. Η επίδοση του αλγορίθμου θα είναι μια τυχαία μεταβλητή που καθορίζεται από τα τυχαία bits, έτσι ώστε είτε ο χρόνος λειτουργίας ή η έξοδος (ή και τα δύο) να είναι τυχαίες μεταβλητές. Και στις δύο περιπτώσεις, της τυχαίας εξόδου και της τυχαίας απόδοσης, ο όρος “αλγόριθμος” για την διαδικασία αυτή χρησιμοποιείται με μεγάλο βαθμό ελευθερίας, καθώς τυπικά δεν ισχύει. Παρόλα αυτά, σε πολλές περιπτώσεις, οι μη-ντετερμινιστικοί αλγόριθμοι είναι οι μόνοι πρακτικά εφαρμόσιμοι τρόποι επίλυσης. Παρακάτω, παρουσιάζουμε συνοπτικά μερικούς ντετερμινιστικούς και μη αλγορίθμους.

Παραδείγματα Ντετερμινιστικών Αλγορίθμων

Gradient Descend

Ο Gradient Descent είναι ένας αλγόριθμος βελτιστοποίησης που χρησιμοποιεί την παράγωγο της συνάρτησης προς βελτιστοποίηση και την ιδέα της απότομης καθόδου. Η παράγωγος μιας συνάρτησης είναι απλά η κλίση. Επομένως, αν γνωρίζουμε την κλίση μιας συνάρτησης, τότε είναι λογικό ότι το μόνο που χρειάζεται να κάνουμε είναι με κάποιο τρόπο να μετακινήσουμε την συνάρτηση στην αρνητική κατεύθυνση της κλίσης, έτσι ώστε να μειώσουμε την τιμή της συνάρτησης. Ο Gradient Descent είναι μια επαναληπτική μέθοδος και η βασικά ιδέα είναι η εξής :

- Υπολογισμός της παραγώγου της συνάρτησης σε σχέση με τις ανεξάρτητες μεταβλητές του. Η παράγωγος της συνάρτησης δηλώνεται ως dF(**x**), όπου F(**x**) είναι η συνάρτηση προς βελτιστοποίηση και **x** είναι το διάνυσμα των ανεξάρτητων μεταβλητών.
- Αλλαγή της τιμής του **x** ως εξής: $\boldsymbol{x}_{(n+1)} = \boldsymbol{x}_n - h\, dF(\boldsymbol{x}_n)$, όπου ο δείκτης n αναφέρεται στον αριθμό επανάληψης και h είναι το μέγεθος του βήματος που πρέπει να επιλεγεί, έτσι ώστε το βήμα να μην είναι πολύ μικρό αλλά ούτε και πολύ μεγάλο. Ένα πολύ μεγάλο βήμα θα αυξήσει υπερβολικά το ελάχιστο της συνάρτησης, ενώ ένα πολύ μικρό βήμα θα έχει σαν αποτέλεσμα μεγάλο χρόνο σύγκλισης.
- Επανάληψη των δύο παραπάνω βημάτων μέχρι να συγκλίνουν σε ένα ελάχιστο της συνάρτησης F(**x**).

Ο Gradient Descent είναι μία ελκυστική μέθοδος βελτιστοποίησης υπό υην έννοια ότι είναι εννοιολογικά απλή και συχνά συγκλίνει γρήγορα. Τα μειονεκτήματά της είναι πως η παράγωγος της συνάρτησης πρέπει να είναι διαθέσιμη και ότι συγκλίνει σε ένα τοπικό και όχι ολικό ελάχιστο. [20]

Newton's Method

Βασική ιδέα της μεθόδου είναι η προσέγγιση της προς ελαχιστοποίηση συνάρτησης f από μία τετραγωνική συνάρτηση και στη συνέχεια η ελαχιστοποίηση της τετραγωνικής συνάρτησης. Προκειμένου να επιτευχθεί η προσέγγιση αναπτύσσεται η f κατά truncated Taylor στην περιοχή του x_k .

$$f(x) \simeq f(x_k) + \nabla f(x_k)(x - x_k) + \frac{1}{2}(x - x_k)^T F(x_k)(x - x_k)$$

Το δεξί μέλος ελαχιστοποιείται για $x_{(k+1)} = x_k - F_k^{-1} g_k$ και αυτός είναι ο αναδρομικός τύπος της μεθόδου Newton. Αναφορικά με τις ικανές συνθήκες 2ης τάξης για την ύπαρξη ελαχίστου, θα θεωρηθεί ότι στο τοπικό ελάχιστο x* η Εσσιανή F(x*) είναι θετικά ορισμένη. Στη συνέχεια θα εξεταστεί αν η f έχει συνεχείς δεύτερες μερικές παραγώγους και αν η F(x) είναι θετικά ορισμένη στην περιοχή του x*, ώστε η μέθοδος να είναι καλά ορισμένη γύρω από τη λύση. Η μέθοδος Newton συγκεντρώνει πολλές επιθυμητές ιδιότητες αν ξεκινήσει από αρχικό σημείο κοντά στη λύση. [21]

Παραδείγματα Μη-Ντετερμινιστικών (Στοχαστικών) Αλγορίθμων

Ant Colony Algorithm

Ο αλγόριθμος βελτιστοποίησης Ant Colony (ACO) είναι μία μη-ντετερμινιστική τεχνική για την επίλυση υπολογιστικών προβλημάτων τα οποία μπορούν “περιγραφούν” με την εύρεση καλών διαδρομών μέσα από γράφους. Το φυσικό φαινόμενο που μοντελοποιεί ο συγκεκριμένος αλγόριθμος είναι η συμπεριφορά που παρουσιάζουν τα μυρμήγκια κατά τις μετακινήσεις τους από και προς τις φωλιές τους στην προσπάθεια συλλογής τροφής. Παρότι πρόκειται για σχεδόν τυφλά ζώα, τα μυρμήγκια, αξιοποιώντας τα ίχνη φερορμόνης που έχουν παραμείνει στο έδαφος από προηγούμενες μετακινήσεις, έχουν τη δυνατότητα εύρεσης της πιο σύντομης διαδρομής από τις φωλιές τους προς τα σημεία συλλογής τροφής ακολουθώντας πάντα τα μονοπάτια εκείνα στα οποία τα ίχνη είναι περισσότερο έντονα. Παράλληλα, κατά τη μετακίνησή τους, τα μυρμήγκια αφήνουν στο έδαφος νέες ποσότητες φερορμόνης, οι οποίες θα αξιοποιηθούν για τον ίδιο σκοπό από άλλα μυρμήγκια στο μέλλον. Με τον τρόπο αυτό όλα τα μυρμήγκια καταλήγουν στην επιλογή της πιο σύντομης διαδρομής, την οποία και εξακολουθούν να επιλέγουν εφόσον δεν εμφανιστούν νέα εμπόδια κατά μήκος αυτής, οπότε και η διαδικασία επιλογής του συντομότερου μονοπατιού ξεκινά από την αρχή.

Βασισμένοι στη συμπεριφορά αυτή των μυρμηγκιών, οι επιστήμονες επινόησαν τον “Ant Colony Search Algorithm”. Κατά την εφαρμογή του ACSA σε ένα πρόβλημα βελτιστοποίησης, ένα σύνολο “τεχνητών μυρμηγκιών” συνεργάζονται μεταξύ τους ανταλλάσσοντας πληροφορίες με στόχο την εύρεση μιας λύσης, η οποία προσεγγίζει τη βέλτιστη.

Η εφαρμογή του ACSA για την επίλυση οποιουδήποτε συνδυαστικού προβλήματος βελτιστοποίησης

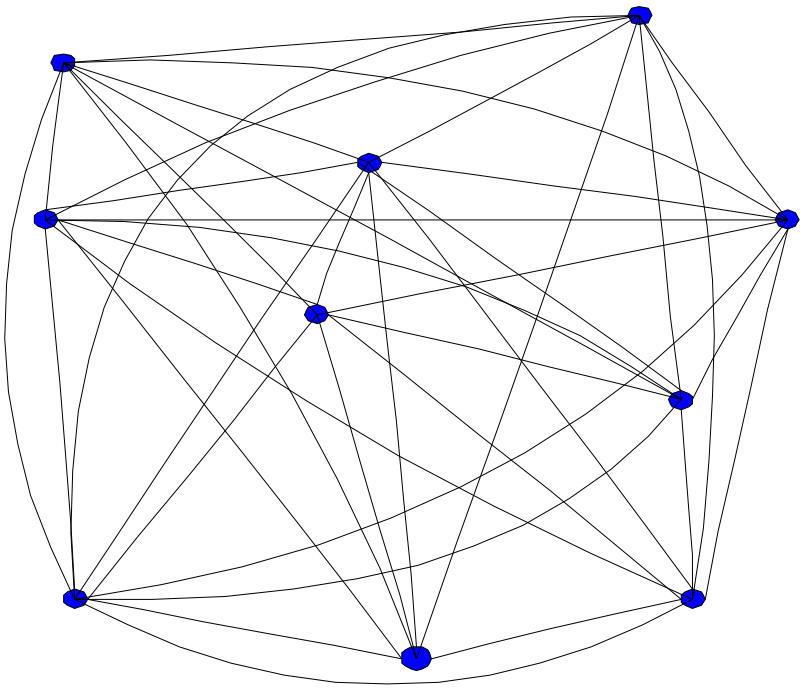
συνίσταται στην επαναληπτική εκτέλεση ορισμένων βασικών βημάτων. Το πρώτο βήμα, λοιπόν, του αλγορίθμου είναι η μοντελοποίηση και κατασκευή του πλήρους γράφου του προβλήματος. Οι κορυφές και οι ακμές του γράφου αντιπροσωπεύουν κάθε φορά μεγέθη τα οποία εξαρτώνται από τη φύση και τους περιορισμούς του εκάστοτε προβλήματος.

Κατά την αλγοριθμική διαδικασία, ένας προκαθορισμένος αριθμός μυρμηγκιών τοποθετείται τυχαία σε κάποιες από τις κορυφές του γράφου (ένα μυρμήγκι σε καθεμία από τις επιλεγόμενες κορυφές). Κατά τη διαδοχική μετάβαση των μυρμηγκιών από μία κορυφή σε μία επόμενη κατασκευάζονται τελικά πλήρεις εφικτές λύσεις του εξεταζόμενου προβλήματος, οι οποίες κάθε φορά αξιολογούνται και ανάλογα με το κατά πόσο θεωρούνται επιθυμητές ή όχι επηρεάζουν τις μετακινήσεις των μυρμηγκιών σε μελλοντικές επαναλήψεις της διαδικασίας αναζήτησης της βέλτιστης λύσης. Απώτερο στόχο φυσικά αποτελεί η εύρεση μιας τελικής εφικτής λύσης, η οποία θα προσεγγίζει αρκετά τη βέλτιστη.Κατά τη διάρκεια των μεταβάσεών τους από κορυφή σε κορυφή, τα μυρμήγκια δεν κινούνται αυτόνομα, αλλά συνεργάζονται σε μια παράλληλη διαδικασία αναζήτησης και εύρεσης της βέλτιστης λύσης. Η διαδικασία αυτή περιλαμβάνει συλλογή και ανταλλαγή πληροφοριών, τις οποίες τα μυρμήγκια συναντούν κατά τις μεταβάσεις τους και προέρχονται από προηγούμενες επαναλήψεις της αλγοριθμικής διαδικασίας.

Καταρχάς, σε κάθε κορυφή του γράφου αντιστοιχίζεται ένα μέγεθος γνωστό σαν “κόστος”, το οποίο παραμένει σταθερό καθ’ όλη τη διάρκεια εφαρμογής του αλγορίθμου και καθορίζει το κατά πόσον είναι επιθυμητή η μετάβαση στην κορυφή αυτή κατά τη διάρκεια κατασκευής μιας λύσης. Όσο πιο μικρό είναι το “κόστος” μιας κορυφής, τόσο περισσότερο επιθυμητή είναι η συμμετοχή αυτής σε μια πλήρη λύση του προβλήματος.

Επιπλέον, σε κάθε ακμή του γράφου του προβλήματος αντιστοιχίζεται ένα μέγεθος γνωστό σαν “φερορμόνη”, η οποία σε αντίθεση με το “κόστος”, μεταβάλλεται καθώς η επαναληπτική διαδικασία προχωρά και επηρεάζει την επιλογή της συγκεκριμένης ακμής από ένα μυρμήγκι. Όσο μεγαλύτερη είναι η τιμή της “φερορμόνης” μιας ακμής, τόσο πιο επιθυμητή είναι η συμμετοχή των κορυφών που συνδέει η συγκεκριμένη ακμή σε μια συνολική λύση του προβλήματος. Αξιοποιώντας τις πληροφορίες που προσφέρουν τα δυο αυτά μεγέθη, τα μυρμήγκια μετακινούνται από κορυφή σε κορυφή κατασκευάζοντας πλήθος εφικτών λύσεων, επιλέγοντας κάθε φορά τις διαδρομές εκείνες στις οποίες η

“φερορμόνη” εμφανίζει μεγαλύτερες τιμές και κατά συνέπεια το συνολικό “κόστος” των κορυφών τους είναι μικρό. Μετά από έναν ορισμένο αριθμό επαναλήψεων, όλα τα μυρμήγκια θα ακολουθήσουν την ίδια διαδρομή, παρέχοντας μια τελική λύση αρκετά κοντά στη βέλτιστη.



Γράφος εφαρμογής του ACSA

Έστω ότι n ο αριθμός των μυρμηγκιών που τοποθετούνται τυχαία σε κάποιες κορυφές του γράφου. Προφανώς ο αριθμός των μυρμηγκιών πρέπει να είναι το πολύ ίσος με το συνολικό αριθμό των κορυφών του γράφου, ενώ για την καλύτερη απόδοση του αλγορίθμου ο αριθμός των μυρμηγκιών πρέπει να είναι κατά πολύ μικρότερος από το συνολικό αριθμό κορυφών (συνήθως το n κυμαίνεται μεταξύ δέκα και είκοσι, ανάλογα πάντα με το μέγεθος του εξεταζόμενου προβλήματος).

Ας θεωρήσουμε, τώρα, ότι ενα μυρμήγκι που βρίσκεται σε μια κορυφή (έστω r) καλείται να επιλέξει την επόμενη κορυφή (έστω s) στην προσπάθεια κατασκευής μιας πλήρους λύσης. Αφού καθοριστούν, βάση των απαιτήσεων του προβλήματος, οι εφικτές μετακινήσεις που είναι δυνατό να πραγματοποιηθούν από κάθε μυρμήγκι (είναι διαφορετικές για καθένα από αυτά), υπολογίζεται για κάθε εφικτή μετακίνηση το γινόμενο:

$$t(r,s)[n(r,s)]^{\beta}$$

όπου t(r,s) είναι το επίπεδο “φερορμόνης” του μονοπατιού (r,s), n(r,s) είναι ένα μέγεθος γνωστό σαν “απόσταση”, το οποίο καθορίζεται από το “κόστος” των κορυφών r και s, ενώ β είναι μια σταθερά που καθορίζει τη βαρύτητα της “φερορμόνης” έναντι της “απόστασης” για την επιλογή της επόμενης κορυφής και δεν μεταβάλλεται καθ’όλη τη διάρκεια εφαρμογής του αλγορίθμου.

Παράλληλα, παράγεται ένας τυχαίος αριθμός q που ανήκει στο διάστημα [0,1] και ακολουθεί την ομοιόμορφη κατανομή. Εφόσον $q<q_o$ (όπου q_o σταθερά η οποία παραμένει αμετάβλητη καθ’όλη τη διάρκεια της διαδικασίας αναζήτησης και ανήκει στο διάστημα [0,1]), το κάθε μυρμήγκι θα επιλέξει την επόμενη κορυφή s, έτσι ώστε το μονοπάτι (r,s) που θα σχηματιστεί να εμφανίζει το μεγαλύτερο γινόμενο από όλα τα εφικτά μονοπάτια (ακμές που συνδέουν δύο κορυφές) που αντιστοιχούν στο συγκεκριμένο μυρμήγκι. Αν $q>q_o$, τότε το μυρμήγκι θα διαλέξει την επόμενη κορυφή βάση της παρακάτω πιθανότητας:

$$p_k(r,s) = \frac{t(r,s) * [n(r,s)]^{\beta}}{\sum t(r,s) * [n(r,s)]^{\beta}}$$

Έτσι, στην περίπτωση που $q>q_o$ το μυρμήγκι θα επιλέξει τυχαία την επόμενη κορυφή, με μεγαλύτερη πάντως πιθανότητα επιλογής του μονοπατιού που εμφανίζει το μεγαλύτερο γινόμενο. Με αυτό τον τρόπο επιλογής διαδοχικών κορυφών εξασφαλίζεται ως ένα βαθμό η δυνατότητα του αλγορίθμου να διαφύγει τον κίνδυνο παγίδευσης σε τοπικά σημεία ακροτάτων, καθώς πραγματοποιείται μια περισσότερο ενδελεχής εξερεύνηση του πεδίου λύσεων του προβλήματος. Μέσα από αυτή τη συνεχόμενη διαδικασία επιλογής και λαμβάνοντας κάθε φορά υπ’ όψιν τις πληροφορίες που παρέχουν η “φερορμόνη” και το “κόστος”, κάθε μυρμήγκι κατασκευάζει μια πλήρη εφικτή λύση του εξεταζόμενου προβλήματος, έτσι ώστε μετά το πέρας μιας επανάληψης του αλγορίθμου να έχουμε τόσες εφικτές λύσεις, όσες και τα μυρμήγκια που αρχικά τοποθετήσαμε στις κορυφές του γράφου.

Όπως προαναφέρθηκε, η “φερορμόνη” είναι ένα μέγεθος που μεταβάλλεται καθώς η αλγοριθμική διαδικασία επαναλαμβάνεται. Η μεταβολή αυτή πραγματοποιείται σε δύο στάδια κατά τη διάρκεια κάθε επανάληψης. Κατά τη διαδικασία δημιουργίας εφικτών λύσεων και αφού το μυρμήγκι μετακινηθεί από την κορυφή r στην κορυφή s, η “φερορμόνη” της ακμής (r,s) μεταβάλλεται βάση της

παρακάτω σχέσης:

$$t(r,s) = (1-a) * t(r,s) + a * t_o$$

όπου α είναι μια σταθερά που ανήκει στο διάστημα [0,1] και t_o είναι η αρχική ποσότητα “φερορμόνης” που έχουμε τοποθετήσει σε κάθε μονοπάτι πριν ξεκινήσει η διαδικασία αναζήτησης.

Το t_o δίνεται από τη σχέση: $t_o = \left(L_{nn} * n\right)^{-1}$

όπου L_{nn} είναι μια τυχαία αρχική εφικτή λύση, ενώ n είναι ο αριθμός των μυρμηγκιών που έχουμε χρησιμοποιήσει. Η σχέση $t(r,s) = (1-a) * t(r,s) + a * t_o$, η οποία επιφέρει μείωση της “φερορμόνης” των μονοπατιών όπου εφαρμόζεται, εξασφαλίζει ότι τα μυρμήγκια δεν θα κατασκευάσουν παρόμοιες διαδρομές, εξερευνώντας μεγαλύτερο κομμάτι του πεδίου λύσεων, αφού κάθε φορά που επιλέγεται ένα μονοπάτι η “φερορμόνη” του μειώνεται κάνοντας λιγότερο πιθανή την επιλογή του ίδιου μονοπατιού σε επόμενο στάδιο. Η διαδικασία αυτή επαναλαμβάνεται σε όλη τη διάρκεια κατασκευής μιας πλήρους εφικτής λύσης από οποιοδήποτε μυρμήγκι.

Το δεύτερο στάδιο μεταβολής της “φερορμόνης” πραγματοποιείται μετά το πέρας κάθε επανάληψης της αλγοριθμικής διαδικασίας και αφού όλα τα μυρμήγκια έχουν κατασκευάσει τόσες εφικτές λύσεις, όσες και ο αριθμός τους. Αφού εξεταστεί ποιά από τις λύσεις αυτές είναι η καλύτερη, επιλέγονται τα μονοπάτια της καλύτερης λύσης και μόνο σε αυτά αυξάνεται η “φερορμόνη”, έτσι ώστε τα συγκεκριμένα μονοπάτια να αποτυπωθούν ως επιθυμητά, με αποτέλεσμα να τα επισκεφθούν περισσότερα μυρμήγκια σε επόμενες επαναλήψεις. Η “φερορμόνη” των ακμών αυτών αυξάνεται σύμφωνα με τη σχέση: $t(r,s) = (1-a) * t(r,s) + a * \Delta t(r,s)$

όπου $\Delta t(r,s)=(L_{gb})^{-1}$ και L_{gb} είναι η τιμή της λύσης την οποία παρείχε η καλύτερη διαδρομή.

Οι επιλογές για την εφαρμογή της σχέσης παραπάνω είναι δύο. Είναι δυνατόν κάθε φορά να αυξάνεται η “φερορμόνη” των ακμών της συνολικά καλύτερης, μέχρι εκείνη τη στιγμή, λύσης ή να αυξάνεται η “φερορμόνη” στα μονοπάτια της καλύτερης λύσης κάθε επανάληψης. Η εμπειρία δείχνει ότι η πρώτη επιλογή αποδίδει καλύτερα αποτελέσματα. Επιπλέον, είναι δυνατό να αυξάνεται η “φερορμόνη” όχι μόνο του ενός, αλλά των δύο καλύτερων διαδρομών (σύνολο μονοπατιών που αποτελούν μια πλήρη

λύση), όμως και αυτή η επιλογή παρέχει αποτελέσματα αμφίβολης ποιότητας.

Μετά από ένα αριθμό επαναλήψεων και καθώς η “φερορμόνη” των καλύτερων μονοπατιών αυξάνεται, όλα τα μυρμήγκια τείνουν να ακολουθήσουν τη διαδρομή που παρουσιάζει τα μεγαλύτερα επίπεδα “φερορμόνης”, αξιοποιώντας τις πληροφορίες που προέρχονται απο προηγούμενες επαναλήψεις και παρέχοντας τελικά μια λύση αρκετά κοντά στη βέλτιστη.[22]

Simulated Anneling (Προσομοιωμένη Ανόπτηση)

Πρόκειται για μια γενική στοχαστική τεχνική βελτιστοποίησης, η οποία θεωρητικά συγκλίνει ασυμπτωτικά στη συνολικά βέλτιστη λύση. Το φυσικό φαινόμενο που μοντελοποιείται στη συγκεκριμένη περίπτωση είναι η αργή ψύξη ενός μετάλλου, η οποία επιτυγχάνεται με τη σταδιακή μείωση της θερμοκρασίας του περιβάλλοντος, αφού πρώτα η θερμοκρασία αυξηθεί απότομα μέχρι μια συγκεκριμένη υψηλή τιμή. Κατά τη διαδικασία αυτή, η ενέργεια του μετάλλου τείνει να πάρει μια ελάχιστη τιμή. Πιο συγκεκριμένα, σε κάθε στάδιο η θερμοκρασία διατηρείται σταθερή για αρκετό χρονικό διάστημα, ώστε το μέταλλο να αποκτήσει τη θερμική του ισορροπία. Στην κατάσταση αυτή, το μέταλλο μπορεί να εμφανίζει διαφορετικά ενεργειακά επίπεδα, τα οποία ανταποκρίνονται σε διαφορετικά spin των ηλεκτρονίων του μετάλλου και αντιστοιχούν σε διαφορετικές τιμές της ενέργειάς του. Η πιθανότητα P_{config} το μέταλλο να εμφανίζει, σε συνθήκες θερμικής ισορροπίας, συγκεκριμένο ενεργειακό επίπεδο δίνεται από τη σχέση: $P_{config} = K \exp\left(-E_{config} / C_p\right)$

όπου:

K	:	μια σταθερά
E_{config}	:	η τιμή της ενέργειας του μετάλλου για το συγκεκριμένο ενεργειακό επίπεδο
C_p	:	η θερμοκρασία του μετάλλου

Με τη σταδιακή μείωση της θερμοκρασίας επιτυγχάνεται η σύγκλιση σε όλο και χαμηλότερες τιμές ενέργειας του μετάλλου και καθώς η θερμοκρασία προσεγγίζει την ελάχιστη τιμή της, οι ενεργειακές καταστάσεις μικρότερης τιμής παρουσιάζουν μεγαλύτερη πιθανότητα εμφάνισης .

Εξομοιώνοντας την παραπάνω διαδικασία με το πρόβλημα της ελαχιστοποίησης συναρτήσεων,

είναι δυνατό να επιλυθεί ένας μεγάλος αριθμός συνδυαστικών προβλημάτων βελτιστοποίησης ακολουθώντας την ίδια διαδικασία ¨μετάβασης¨ από μία κατάσταση ισορροπίας σε μια άλλη και αναζητώντας την ελάχιστη ενεργειακή κατάσταση του συστήματος. Δημιουργώντας την αναλογία ανάμεσα στη διαδικασία της αργής ψύξης ενός μετάλλου και του εκάστοτε προβλήματος, διαπιστώνουμε ότι:

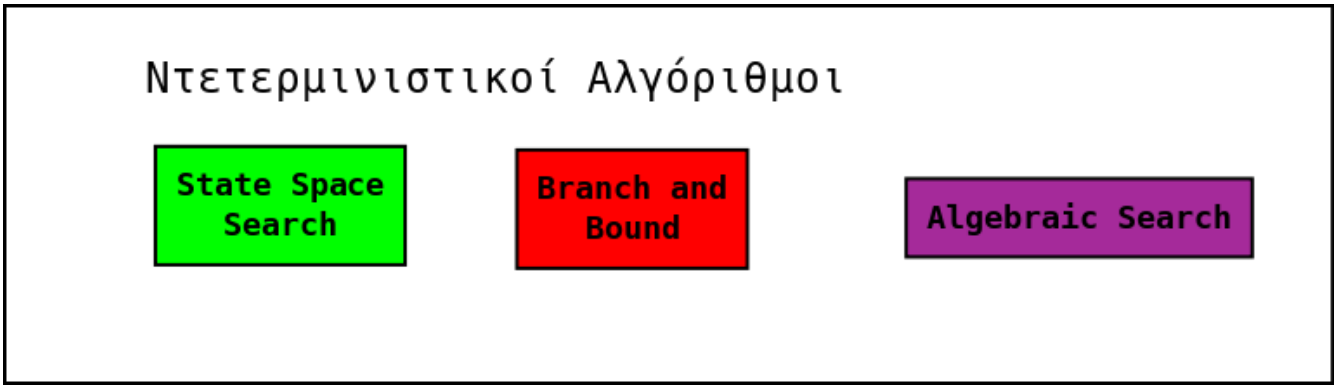
- Οι πιθανές λύσεις του προβλήματος αντιστοιχούν στις διάφορες ενεργειακές καταστάσεις του μετάλλου.
- Οι τιμές της συνάρτησης κόστους αντιστοιχούν στις τιμές της ενέργειας που εμφανίζει το μέταλλο σε κάθε κατάσταση.
- Η θερμοκρασία παίζει το ρόλο της παραμέτρου ελέγχου της όλης διαδικασίας, εξασφαλίζοντας ότι ικανοποιούνται οι απαιτήσεις και οι περιορισμοί του προβλήματος.

Ο αλγόριθμος της Προσομοιωμένης Ανόπτησης (ΠΑ), αρχίζει με την τυχαία επιλογή μιας εφικτής λύσης του προβλήματος. Στη συνέχεια πραγματοποιείται η αξιολόγηση νέων λύσεων, οι οποίες προκύπτουν από την αρχική μέσω τυχαίων αλλαγών που προκαλούνται σε αυτή. Η επιλογή αντικατάστασης της τρέχουσας λύσης από μια νέα υποψήφια λύση πραγματοποιείται εφόσον η τελευταία προσδίδει χαμηλότερη τιμή στην συνάρτηση προς ελαχιστοποίηση ή, αν δεν συμβαίνει αυτό, όταν ικανοποιείται η συνθήκη: $Exp\left(-\Delta E / C_p\right) \geq U$

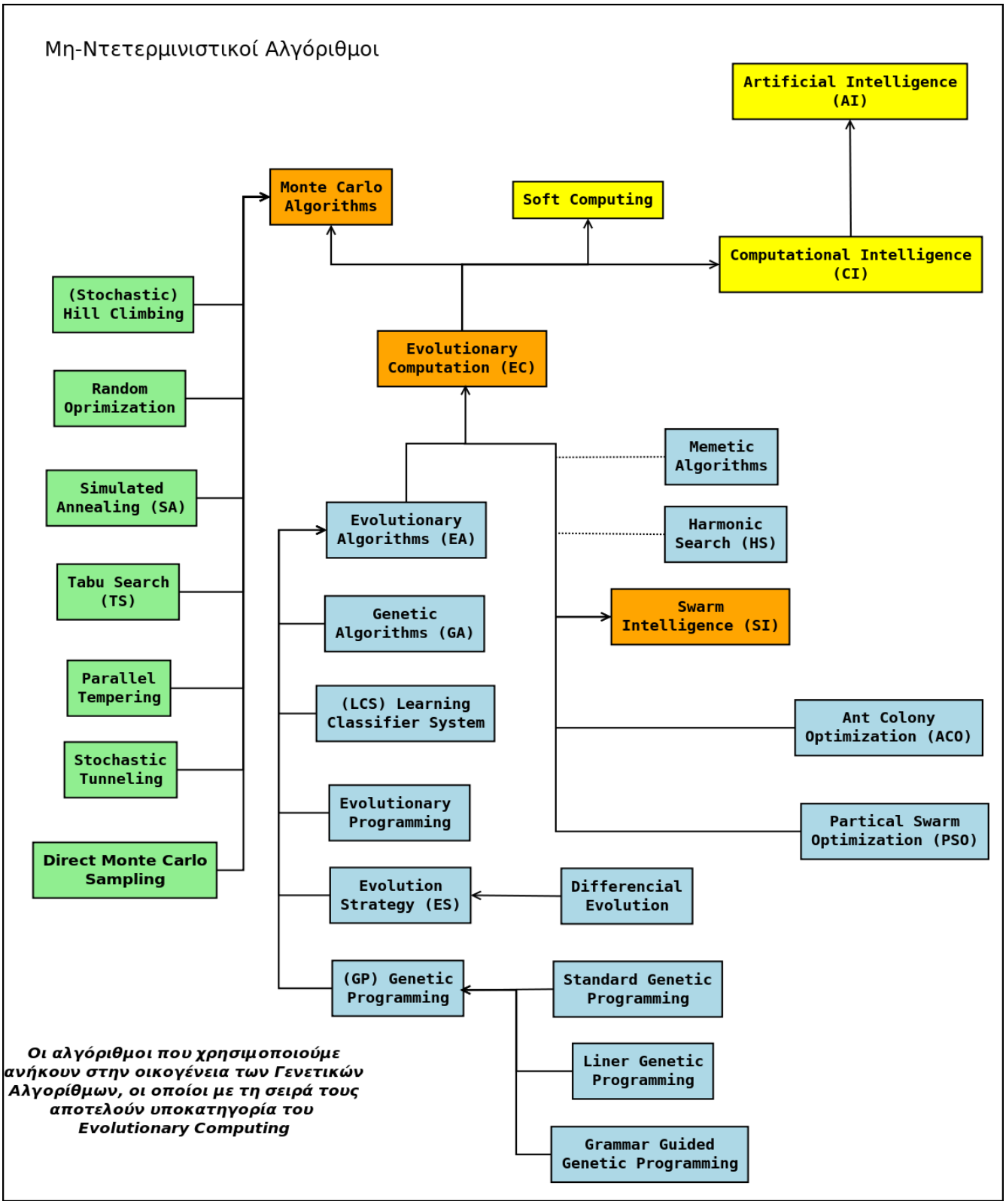
όπου ΔΕ είναι η αύξηση της αντικειμενικής συνάρτησης που εμφανίζει η νέα λύση σε σχέση με την αρχική και U είναι ένας τυχαίος αριθμός στο διάστημα [0,1], ο οποίος ακολουθεί την ομοιόμορφη κατανομή. Με τη σταδιακή μείωση της παραμέτρου C_p διασφαλίζεται ότι η πιθανότητα αποδοχής μιας λύσης που δίνει μεγαλύτερη τιμή της αντικειμενικής συνάρτησης μειώνεται καθώς η διαδικασία έρευνας του πεδίου λύσεων προχωρά.

Το γεγονός ότι μια λύση δεν απορρίπτεται αυτόματα όταν δεν αποδίδει χαμηλότερο λειτουργικό κόστος εξασφαλίζει την ενδελεχή εξερεύνηση του πεδίου λύσεων του προβλήματος και αποτρέπει το φαινόμενο της σύγκλισης σε σημεία τοπικών ελαχίστων της συνάρτησης κόστους, οδηγώντας την επίλυση προς το ολικό ελάχιστο. Η μέθοδος είναι αισθητά πιο γρήγορη από το δυναμικό προγραμματισμό, μπορεί εύκολα να αντιμετωπίσει πολύπλοκους περιορισμούς, δεν δεσμεύει μεγάλο κομμάτι της μνήμης του υπολογιστή για την εφαρμογή της και έχει αποδειχτεί θεωρητικά ότι συγκλίνει

στη βέλτιστη λύση. Επιπλέον, η επιλογή της αρχικής λύσης δεν επηρεάζει σημαντικά την όλη διαδικασία και γι΄ αυτόν η μέθοδος μπορεί να χρησιμοποιηθεί και σαν εργαλείο βελτίωσης λύσεων παρεχόμενων από άλλες μεθόδους επίλυσης. [23]



Σχεδιάγραμμα 4: Κατάταξη ντετερμινιστικών αλγορίθμων βελτιστοποίησης



Σχεδιάγραμμα 5: Κατάταξη μη-ντετερμινιστικών (σταχαστικών) αλγορίθμων βελτιστοποίησης

2.3 Γενετικοί αλγόριθμοι

Η φύση έχει έναν πολύ ισχυρό μηχανισμό εξέλιξης των οργανισμών, που βασίζεται στον ακόλουθο κανόνα της φυσικής επιλογής: οι οργανισμοί που δεν μπορού να να επιβιώσουν στο περιβάλλον τους πεθαίνουν, ενώ οι υπόλοιποι πολλαπλασιάζονται μέσω της αναπαραγωγής. Οι απόγονοι παρουσιάζουν μικρές αποκλίσεις από τους προγόνους τους, ενώ συνήθως υπερισχύουν αυτοί που συγκεντρώνουν τα καλύτερα χαρακτηριστικά. Αν το περιβάλλον μεταβάλλεται με αργούς ρυθμούς, τα διάφορα είδη μπορούν να εξελίσσονται σταδιακά, ώστε να προσαρμόζονται σε αυτό. Αν όμως συμβούν ραγδαίες μεταβολές, αρκετά είδη οργανισμών θα εξαφανιστούν. Σποραδικά, συμβαίνουν τυχαίες μεταλλάξεις, από τις οποίες οι περισσότερες οδηγούν τα μεταλλαγμένα άτομα στο θάνατο, αν και είναι πιθανό πολύ σπάνια να οδηγήσουν στη δημιουργία νέων “καλύτερων” οργανισμών.

Οι γενετικοί αλγόριθμοι είναι μία κατηγορία υπολογιστικών μοντέλων εμπνευσμένη από την θεωρία της εξέλιξης (evolution). Αυτοί οι αλγόριθμοι κωδικοποιούν μία πιθανή λύση ενός προβλήματος σε μία απλή δομή δεδομένων, με μορφή που προσομοιώνει αυτή του χρωμοσώματος, και εφαρμόζει συνδυαστικές μεθόδους στις δομές αυτές, ώστε να διατηρήσει κρίσιμες πληροφορίες που περιέχουν. [14], [15]

2.3.1 Ενδεικτική λειτουργία τυπικού ΓΑ

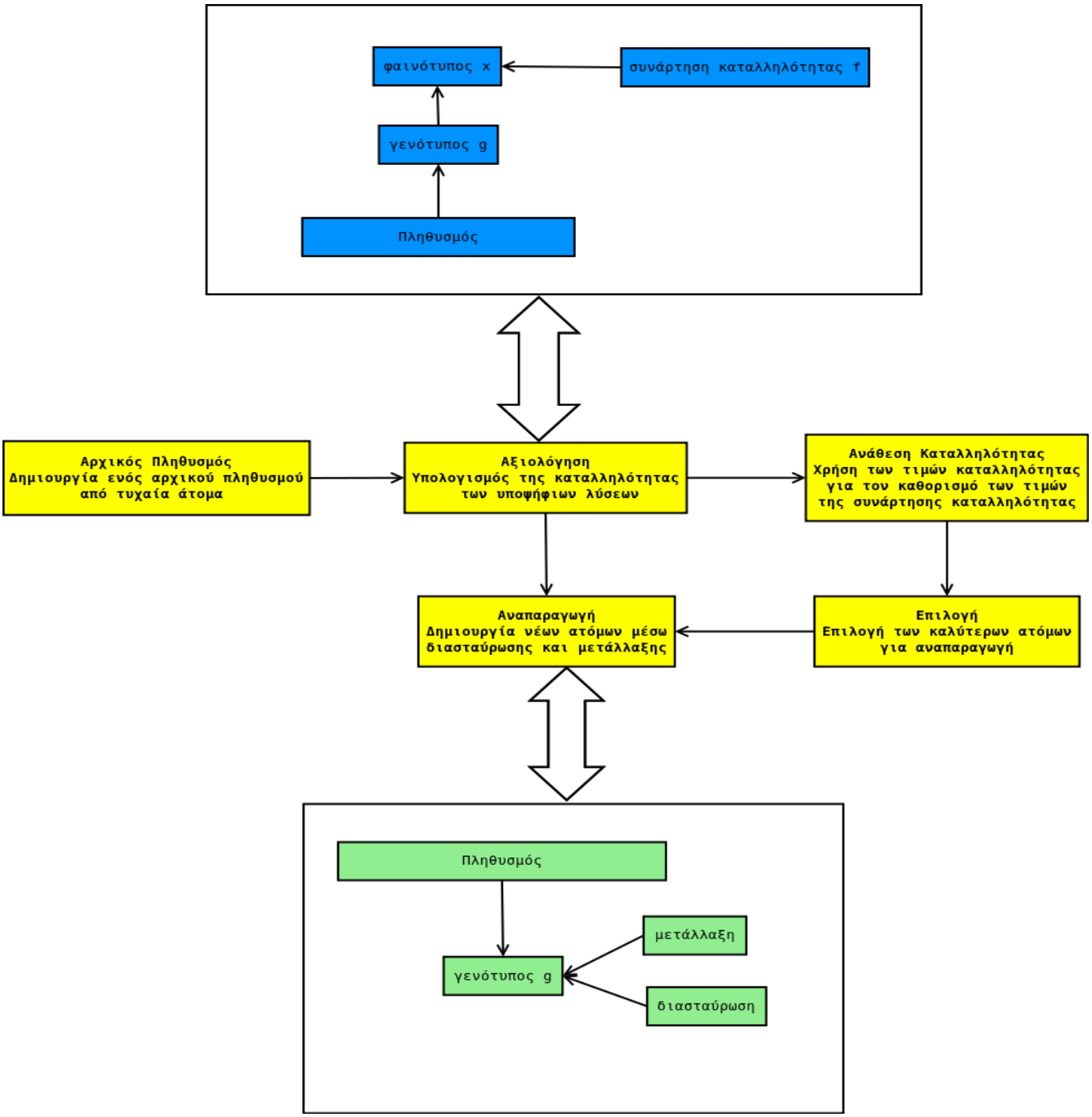
Η φιλοσοφία των γενετικών αλγορίθμων είναι η εξής: Αρχικά δημιουργείται με τυχαίο τρόπο ένα σύνολο υποψηφίων λύσεων P. Το σύνολο αυτό αποτελείται κατά κύριο λόγο από μη αποδεκτές λύσεις (π.χ. δεν είναι έγκυρες, δεν είναι βέλτιστες, κλπ). Έστω N το πλήθος του συνόλου P. Οι λύσεις αυτές βαθμολογούνται από μία συνάρτηση καταλληλότητας (fitness function). Η βαθμολόγησή τους συνίσταται στην αντιστοίχηση κάθε υποψήφιας λύσης με έναν αριθμό, ο οποίος υποδηλώνει την εγγύτητα της μη αποδεκτής αυτής λύσης ως προς κάποια αποδεκτή. Στη συνέχεια, από τον αρχικό πληθυσμό σχηματίζονται N/2 ζευγάρια όχι απαραίτητα μοναδικών γονέων, δίνοντας μεγαλύτερη προτεραιότητα στις πλέον κατάλληλες λύσεις. Κάθε ζευγάρι ζευγαρώνει (mates), δίνοντας δύο νέες λύσεις, τους απογόνους (offsprings). Ο νέος πληθυσμός P' αποτελείται από το σύνολο των απογόνων και συνήθως αποτελεί βελτίωση του αρχικού πληθυσμού. Η διαδικασία επαναλαμβάνεται για τον νέο

πληθυσμό P', ενώ οι πιο συνηθισμένες συνθήκες τερματισμού της είναι η εύρεση μιας τέλειας λύσης με βάση τη συνάρτηση καταλληλότητας και η σύγκλιση όλων των λύσεων σε μία. Συνοπτικά, για τη λειτουργία ενός τυπικού ΓΑ, θα μπορούσαμε να πούμε ότι περιγράφεται από τα εξής βήματα:

1. Δημιουργείται ένας αρχικός πληθυσμός P, με N υποψήφιες λύσεις
2. Υπολογίζεται η καταλληλότητα της κάθε λύσης
3. Όσο ο πληθυσμός P δεν συγκλίνει σε μία λύση επαναλαμβάνονται N φορές τα ακόλουθα βήματα:
   (a) Επιλογή δύο λύσεων από τον πληθυσμό P
   (b) Συνδυασμός των δύο λύσεων για παραγωγή δύο απογόνων
   (c) Υπολογισμός της καταλληλότητας των δύο απογόνων
   (d) Εισαγωγή των δύο απογόνων στον νέο πληθυσμό

Στο βήμα (1) έχουμε την δημιουργία ενός τυχαίου αρχικού πληθυσμού. Στο βήμα (2) τα μέλη του αρχικού πληθυσμού αξιολογούνται ανάλογα με την συνάρτηση καταλληλότητας. Όσο τα μέλη του πληθυσμού έχουν τιμές μικρότερες από την μέγιστη τιμή προχωράμε στο βήμα (3), το οποίο είναι η δημιουργία ενός νέου πληθυσμού, δηλαδή της επόμενης γενιάς. Στο βήμα (a) επιλέγονται μέλη του αρχικού πληθυσμού και στο βήμα (b) συνδυάζονται έτσι ώστε να παράγουν έναν απόγονο, δηλαδή ένα μέλος της επόμενης γενιάς. Στο βήμα (c) τα νέα μέλη αξιολογούνται και αν έχουν υψηλές τιμές καταλληλότητας τότε περνάνε στο βήμα (d), δηλαδή εισάγονται στην νέα γενιά. Ο κύκλος αυτός συνεχίζεται μέχρι ο αλγόριθμος να συγκλίνει.

Ο γενετικός αλγόριθμος εκτελεί μια αναζήτηση στο χώρο των υποψηφίων λύσεων, με στόχο την εύρεση κάποιας λύσης που μεγιστοποιεί την συνάρτηση καταλληλότητας. Η αναζήτηση είναι παράλληλη, καθώς σε κάθε υποψήφια λύση μπορεί να εκτελεστεί ξεχωριστή αναζήτηση. Η μέθοδος της αναζήτησης μπορεί να θεωρηθεί σαν αναρρίχηση λόφου (hill climbing), καθώς γίνονται μικρές αλλαγές στις υποψήφιες λύσεις του πληθυσμού και επιλέγονται πάντα οι καλύτερες, βάσει της συνάρτησης καταλληλότητας. Η αναζήτηση επικεντρώνεται στις περισσότερο κατάλληλες λύσεις, χωρίς όμως να αγνοούνται οι υπόλοιπες, καθώς πάντα υπάρχει ο κίνδυνος να παγιδευτεί η διαδικασία σε τοπικό μέγιστο (local maximum).



Σχεδιάγραμμα 6: Βασικός κύκλος γενετικού αλγορίθμου

Υπάρχουν τέσσερα βασικά θέματα που αφορούν την εφαρμογή των ΓΑ στην επίλυση προβλημάτων: η αναπαράσταση των λύσεων, η συνάρτηση καταλληλότητας, η επιλογή των ζευγαριών και η διαδικασία

της αναπαραγωγής. [16]

2.3.2 Αναπαράσταση υποψηφίων λύσεων

Στην κλασική περίπτωση των ΓΑ, κάθε υποψήφια λύση αναπαρίσταται με μία συμβολοσειρά (string) ενός πεπερασμένου αλφαβήτου. Κάθε στοιχείο της συμβολοσειράς ονομάζεται και γονίδιο (gene). Συνήθως η συμβολοσειρά αναφέρεται και σαν χρωμόσωμα (chromosome), δανειζόμενη όρους από τη βιολογία. Στο πραγματικό DNA το αλφάβητο έχει μήκος τέσσερα και αποτελείται από τα στοιχεία A (adenine), G (guanine), C (cytocine) και T (thymine), ενώ στους ΓΑ χρησιμοποιείται συνήθως το δυαδικό αλφάβητο, οπότε οι συμβολοσειρές ονομάζονται και bit-strings. Ωστόσο υπάρχουν περιπτώσεις ΓΑ με πιο σύνθετες μορφές αναπαράστασης. [17],[16]

2.3.2.1 Δυαδική Κωδικοποίηση

Η πρώτη μορφή κωδικοποίησης που εφαρμόστηκε στους Γ.Α. Ήταν η δυαδική κωδικοποίηση. Όταν χρησιμοποιείται η δυαδική κωδικοποίηση υπάρχουν δύο δυνατοί τρόποι χρήσης της:

1. Όλο το χρωμόσωμα είναι ένας ακέραιος αριθμός και τα γονίδιά του είναι δυαδικά στοιχεία (0 ή 1). Σε αυτήν την περίπτωση μπορούμε να συναντήσουμε αλγορίθμους όπου το μήκος του χρωμοσώματος είναι σταθερό, καθώς και αλγορίθμους όπου το μήκος του χρωμοσώματος δεν είναι σταθερό.
2. Το χρωμόσωμα αποτελείται από μία σειρά από δυαδικούς αριθμούς. Συνήθως το πλήθος των δυαδικών αριθμών σε ένα χρωμόσωμα είναι σταθερό αν και μπορεί να αυξομειώνεται το μήκος κάθε επιμέρους δυαδικού αριθμού.

Στα περισσότερα προβλήματα οι λύσεις περιγράφονται με μεταβλητές διαφόρων τύπων δεδομένων, επομένως η διαδικασία της κωδικοποίησης περιλαμβάνει τη μετατροπή των τιμών αυτών των μεταβλητών στις αντίστοιχες δυαδικές. Για παράδειγμα, αν μία λύση περιγράφεται από δύο μεταβλητές, μία boolean με τιμή 1 και μία long integer με τιμή 73 (01001001 σε binary), η κωδικοποίηση θα απαιτούσε 9 bits και το παραγόμενο bit-string θα ήταν το 101001001.

'Ενα μεγάλο πρόβλημα που υπάρχει με την δυαδική αναπαράσταση αριθμών είναι πως μία μικρή αλλαγή σε ένα χρωμόσωμα μπορεί να προκαλέσει μεγάλες αλλαγές στην επίδοση του χρωμοσώματος.

Για παράδειγμα, οι αριθμοί 15=01111 και 16=10000 αν και έχουν διαφορά μόνο μία μονάδα πρέπει να αλλάξουν 5 bits προκειμένου να μεταβούμε από τον ένα αριθμό στον άλλο. Προβλήματα αυτού του είδους μπορούν να λυθούν αν ακολουθηθεί η κωδικοποίηση Gray.

2.3.2.2 Αναπαράσταση Ακεραίων

Οι ακέραιοι αριθμοί αναπαρίστανται σχετικά εύκολα με τη χρήση χρωμοσωμάτων ενός γενετικού αλγορίθμου, ωστόσο και σε αυτήν την περίπτωση υπάρχουν κάποια προβλήματα. Έστω x ο αριθμός που θέλουμε να κωδικοποιήσουμε και N+1 τα διαθέσιμα δυαδικά ψηφία ενός χρωμοσώματος:

1. Αν ο αριθμός x είναι στο διάστημα $0..2^N-1$, τότε μπορεί να αναπαρασταθεί σχετικά εύκολα χρησιμοποιώντας ένα άμεσο σχήμα $e=[b_0, b_1, ..., b_N], b_i \in \{0,1\}$ έτσι ώστε $x=\sum b_i 2^i, i=0,..., N$.
2. Αν $x \in \{M, ..., M+2_N-1\}$, τότε η κωδικοποίηση γίνεται με τον ίδιο τρόπο όπως και πριν. Ωστόσο κατά την αποτίμηση του χρωμοσώματος, το χρωμόσωμα δεν είναι στο διάστημα $0..2^N-1$, αλλά σε ένα διάστημα ολισθημένο κατά M.
3. Αν $x \in \{0,1,..., L-1\}$, αλλά το L δεν είναι κάποια δύναμη του 2, τότε υπάρχει πρόβλημα, καθώς κάποιοι συνδυασμοί στο χρωμόσωμα μπορεί να οδηγήσουν σε άκυρες τιμές. Για παράδειγμα, έστω η κωδικοποίηση των αριθμών στο διάστημα {0,10}, για την οποία χρειάζονται 4 bits. Κατά την διάρκεια της εφαρμογής του αλγορίθμου μπορεί να προκύψει ο αριθμός 1011=11, ο οποίος είναι εκτός ορίων. Σε τέτοιες περιπτώσεις υπάρχουν τρεις δυνατοί τρόποι αντιμετώπισης:

   (a) Θέτουμε $N=\log_2(L)+1$ και αναπαριστούμε κάθε αριθμό από 0 έως L-2 με μία ακριβώς απεικόνιση και τον τελευταίο αριθμό L-1 με τις υπόλοιπες 2_N-L διαθέσιμες απεικονίσεις. Αυτή η μεθοδολογία ονομάζεται **αποκοπή** (clipping). Το πρόβλημα που έχει αυτή η μέθοδος είναι πως παρουσιάζεται μία πόλωση στην αναπαράσταση, καθώς ένας και μόνο αριθμός καταλαμβάνει περισσότερες αναπαραστάσεις από τους άλλους. Στο παράδειγμα για τον αριθμό 0-10, ο αριθμός 10 θα είχε 6 αναπαραστάσεις από τις 16 (ποσοστό 37.5%) και οι υπόλοιποι αριθμοί από μία αναπαράσταση (ποσοστό 6.25%), με

αποτέλεσμα να προτιμάται κατά την αρχικοποίηση ή και την εκτέλεση των γενετικών τελεστών.

(b) Θέτουμε $N=\log_2(L)+1$ και απεικονίζουμε το ακέραιο μέρος των αριθμών y(x), όπου $y(x)=(x/L-1)(2_N-1)$. Αυτή η διαδικασία είναι πιο δίκαιη από την προηγούμενη, αλλά και πάλι είναι πιθανό κάποιοι αριθμοί να αναπαρίστανται από περισσότερα τους ενός χρωμοσώματα.

(c) Η τρίτη διαδικασία δεν απαιτεί καμία μετατροπή, παρά μόνο έλεγχο για το αν μία αναπαράσταση είναι έγκυρη ή όχι. Ο έλεγχος θα πρέπει να γίνεται τόσο στην διαδικασία της αρχικοποίησης, όσο και στην διαδικασία της εφαρμογής των γενετικών τελεστών. Αν σε κάποια από αυτές τις φάσεις προκύψει χρωμόσωμα εκτός ορίων, τότε απορρίπτεται, είτε αναθέτοντάς του μία κακή τιμή καταλληλότητας ώστε να μην χρησιμοποιηθεί στην αναπαραγωγή, είτε απεικονίζεται με κάποιου είδους αποκοπή στο σύνολο δεδομένων. Αυτός είναι και τρόπος που χρησιμοποιείται πιο συχνά.

2.3.2.3 Δεκαδική Αναπαράσταση

Η αναπαράσταση δεκαδικών αριθμών δεν είναι το ίδιο απλή με την αναπαράσταση ακεραίων αριθμών, καθώς δεν υπάρχει άμεση αντιστοίχηση ανάμεσα σε δεκαδικούς αριθμούς και ακεραίους. Υπάρχουν γενικά δύο τρόποι να αναπαραστήσουμε δεκαδικούς αριθμούς με την χρήση χρωμοσωμάτων, η δυαδική και η άμεση αναπαράσταση.

2.3.2.3.1 Δυαδική Κωδικοποίηση

Για την κωδικοποίηση ενός δεκαδικού αριθμού $x\in[x_{min},x_{max}]$ με N δυαδικά ψηφία σε κάθε χρωμόσωμα, δεν είναι δυνατή η κωδικοποίηση δεκαδικών αριθμών που απέχουν απόσταση μικρότερη από $2_N-1/x_{man}-x_{min}$. Για παράδειγμα, έστω ότι θέλουμε να αναπαραστήσουμε αριθμούς στο διάστημα [-5, 5] και διαθέτουμε 10 bits. Ο πρώτος αριθμός αριθμός που μπορεί να αναπαρασταθεί είναι ο $b_0=[0,0,0,0,0,0,0,0,0,0]$. Ο επόμενος αριθμός που μπορεί να αναπαρασταθεί είναι ο αριθμός -5+10/1024=-4.990234375 με τον συνδυασμό $b_1=[0,0,0,0,0,0,0,0,0,1]$ και ο τρίτος αριθμός είναι ο -5+20/1024=-4.98046875 με τον συνδυασμό $b_2=[0,0,0,0,0,0,0,0,1,0]$. Όπως παρατηρούμε, υπάρχει

ένα σημαντικό κενό ανάμεσα στους δύο αριθμούς, παρόλη την χρήση μεγάλου αριθμού από bits που χρησιμοποιήθηκε στην κωδικοποίηση. Επιπλέον αυτός ο τρόπος αναπαράστασης δεν είναι κατάλληλος για αριθμητικές βελτιστοποιήσεις συναρτήσεων, όπου και οι πολύ μικρές διαφορές παίζουν εξαιρετικά σημαντικό ρόλο.

2.3.2.3.2 Άμεση Κωδικοποίηση

Είναι ο πιο καινά χρησιμοποιούμενος τρόπος αναπαράστασης. Αντί να χρησιμοποιούνται ακέραιοι ως χρωμοσώματα ή διανύσματα ακεραίων, χρησιμοποιούνται άμεσα οι δεκαδικοί αριθμοί που απαιτεί το πρόβλημα. Με αυτόν τον τρόπο αποφεύγονται τα όποια προβλήματα ακρίβειας προκαλεί η δυαδική αναπαράσταση. Η ακρίβεια στους υπολογισμούς περιορίζεται από την ακρίβεια των δεκαδικών αριθμών, ωστόσο το πρόβλημα αυτό θα υπήρχε ακόμα και αν χρησιμοποιούνταν ακέραιη αναπαράσταση.

2.3.2.4 Αλφαβητική Αναπαράσταση

Για την αναπαράσταση αλφαριθμητικών μεταβλητών με την χρήση χρωμοσωμάτων, η αναπαράστασή τους γίνεται με συνδυασμούς ακεραίων. Για παράδειγμα, αν έχουμε ένα πρόβλημα στο οποίο απαιτείται λύση στο σύνολο {a, b, c,d}, τότε μπορούμε να κωδικοποιήσουμε το a με τον αριθμό 1 (ή 01 στο δυαδικό σύστημα), το b με τον αριθμό 2, το c με το 3 και το d με το 4. Μετά το τέλος του Γ.Α. θα πρέπει να μετατραπεί η αναπαράσταση που θα προκύψει σε αυτήν του προβλήματος. Στην περίπτωση που το σύνολο αναπαράστασης του προβλήματος δεν είναι πεπερασμένο και επιτρέπονται λύσεις με συνδυασμούς πολλών ή λίγων γραμμάτων, τότε θα πρέπει να χρησιμοποιηθεί αλφαριθμητική και όχι ακέραια αναπαράσταση.

Στη γενετική, το σύνολο των παραμέτρων που αναπαρίσταται από ένα συγκεκριμένο χρωμόσωμα αναφέρονται σαν γονότυπος (genotype). Ο γονότυπος περιλαμβάνει όλες εκείνες τις πληροφορίες που είναι απαραίτητες για την αναπαράσταση του οργανισμού, ο οποίος αποτελεί τον φαινότυπο (phenotype) του χρωμοσώματος.

2.3.3 Συνάρτηση καταλληλότητας

Η συνάρτηση καταλληλότητας δέχεται σαν είσοδο ένα χρωμόσωμα και επιστρέφει ένα αριθμό, ο οποίος υποδηλώνει πόσο κατάλληλο είναι. Το πεδίο τιμών της συνάρτησης καταλληλότητας συνήθως είναι το διάστημα των πραγματικών τιμών από το 0 έως το 1, αν και ανάλογα με την υλοποίηση αυτό μπορεί να διαφέρει. Το 1 υποδηλώνει ότι το συγκεκριμένο χρωμόσωμα είναι τέλειο, δηλαδή ικανοποιεί όλες τις προϋποθέσεις του προβλήματος και αποτελεί αποδεκτή λύση. Ο τρόπος υλοποίησης της συνάρτησης εξαρτάται από το εκάστοτε πρόβλημα.

Ανάλογα με το πρόβλημα, η κατασκευή της συνάρτησης καταλληλότητας μπορεί να είναι από πολύ απλή έως εξαιρετικά πολύπλοκη. Η ιδανική συνάρτηση καταλληλότητας θα πρέπει να είναι συνεχής και μονότονη. Ωστόσο, επειδή αυτό σπάνια συμβαίνει, το ζητούμενο είναι μια συνάρτηση καταλληλότητας που δεν έχει πολλά τοπικά μέγιστα ή ένα απομονωμένο ολικό μέγιστο.

Ο γενικός κανόνας στην κατασκευή της συνάρτησης καταλληλότητας είναι ότι πρέπει να αντικατοπτρίζει ρεαλιστικά την αξία του χρωμοσώματος, Ωστόσο η έννοια της αξίας του χρωμοσώματος δεν είναι πάντα ιδιαίτερα χρήσιμη. Για παράδειγμα, σε προβλήματα βελτιστοποίησης με περιορισμούς υπάρχουν πολλά χρωμοσώματα που παραβιάζουν τους περιορισμούς και επομένως η αξία τους είναι μηδενική. Ενδεικτικό παράδειγμα αποτελεί το πρόβλημα της δημιουργίας ωρολογίου προγράμματος, όπου οι αίθουσες και οι διδάσκοντες πρέπει να κατανεμηθούν στα τμήματα, με περιορισμούς, όπως οι αίθουσες και οι διδάσκοντες δεν μπορούν να κατανεμηθούν σε παραπάνω από ένα τμήματα την ίδια ώρα. Σε αυτήν την περίπτωση η συνάρτηση καταλληλότητας δεν πρέπει να εκφράζει την ποιότητα της λύσης, αλλά το πόσο κοντά αυτή βρίσκεται σε μία έγκυρη λύση. Έτσι υπολογίζεται ο αριθμός των περιορισμών που παραβιάζει η λύση και γίνεται μια εκτίμηση για την προσπάθεια που απαιτείται για να διορθωθεί.

Μια προσέγγιση που ακολουθείται πολλές φορές είναι αυτή της προσεγγιστικής συνάρτησης καταλληλότητας (approximate fitness function). Το κρίσιμο θέμα είναι η επιθυμητή ακρίβεια της συνάρτησης καταλληλότητας και το υπολογιστικό κόστος που θεωρείται αποδεκτό για μία συνάρτηση καταλληλότητας που δίνει ενδεχομένως άριστα αποτελέσματα. Εάν λοιπόν υπάρχουν δύο διαθέσιμες συναρτήσεις καταλληλότητας, μια ακριβής και μία προσεγγιστική, ενδεχομένως να συμφέρει η χρήση

της προσεγγιστικής, αφού αυτή θα επιτρέψει την πραγματοποίηση δεκαπλάσιας ποσότητας υπολογισμών (δηλαδή αναπαραγωγών) στον ίδιο χρόνο, σε σχέση με την χρονοβόρα συνάρτηση ακριβείας. [17],[16]

2.3.4 Διαδικασία επιλογής γονέων

Η διαδικασία επιλογής γονέων σχετίζεται με την απόδοση πιθανοτήτων επιλογής προς αναπραγωγή στα μέλη ενός πληθυσμού υποψηφίων λύσεων. Κατά τη διαδικασία αυτή,κάποιοι γονείς με υψηλή τιμή στη συνάρτηση καταλληλότητας ενδέχεται να επιλεγούν για αναπαραγωγή περισσότερες από μία φορές, ενώ κάποιοι γονείς με χαμηλή τιμή ενδέχεται να μην επιλεγούν καθόλου.

Κατά τη διαδικασία επιλογής, αρχικά οι υποψήφιες λύσεις αντιγράφονται σε μία δεξαμενή ζευγαρώματος (mating pool). Η δεξαμενή αυτή έχει μέγεθος ίσο με τον αρχικό πληθυσμό, και σε αυτήν αντιγράφονται μέλη του αρχικού πληθυσμού, με πιθανότητα ανάλογη της καταλληλότητάς τους. Για την επιλογή των χρωμοσωμάτων που θα αντιγραφούν στην δεξαμενή ζευγαρώματος χρησιμοποιούνται αρκετές τεχνικές. Μία από τις πιο συνηθισμένες είναι η τεχνική της ρουλέτας. Η τεχνική της ρουλέτας συνοψίζεται στα ακόλουθα βήματα:

1. Παράγεται το άθροισμα όλων των τιμών αξιολόγησης των υποψηφίων λύσεων, έστω S
2. Επιλέγεται ένας τυχαίος αριθμός n, από το 0 μέχρι το S, χρησιμοποιώντας συνάρτηση ομοιόμορφης κατανομής για τη δημιουργία των τυχαίων αριθμών
3. Επαναληπτικά εξετάζεται κάθε υποψήφια λύση και η τιμή προστίθεται σε έναν καταχωρητή K
4. Αν η τιμή του K γίνει ίση ή μεγαλύτερη του n, η λύση επιλέγεται και ο K μηδενίζεται. Στην αντίθετη περίπτωση εκτελείται πάλι το 3
5. Αν δεν έχει επιλεγεί ικανοποιητικός αριθμός υποψηφίων λύσεων εκτελείται το 2, αλλιώς ο αλγόριθμος τερματίζει

Είναι φανερό από τον παραπάνω αλγόριθμο ότι ορισμένες υποψήφιες λύσεις μπορεί να επιλεγούν παραπάνω από μία φορές. Μετά την ολοκλήρωση της παραπάνω διαδικασίας, από τα μέλη της δεξαμενής ζευγαρώματος δημιουργούνται ζευγάρια με τυχαίο τρόπο και οι απόγονοι που προκύπτουν αποτελούν τον νέο πληθυσμό.

Ο τρόπος επιλογής των γονέων που θα ζευγαρώσουν είναι σημαντικός για την απόδοση του ΓΑ. Δύο

προβλήματα που εμφανίζονται συχνά στους ΓΑ και τα οποία μπορούν να αντιμετωπιστούν με τροποποίηση της διαδικασίας επιλογής, είναι η πρόωρη σύγκλιση (premature convergence) και η αργή σύγκλιση (slow convergence). Με τον όρο “σύγκλιση” εννοούμε την επικράτηση ενός χρωμοσώματος ή μικρών παραλλαγών του, σε μεγάλο ποσοστό στον πληθυσμό.

Με έναν αποδοτικό ΓΑ, ο πληθυσμός θα πρέπει μετά από αρκετές επαναλήψεις να συγκλίνει προς το ολικό μέγιστο. Η σύγκλιση αυτή αφορά είτε το καλύτερο στοιχείο του πληθυσμού ή το μέσο όρο αυτού. Ένα γονίδιο θεωρείται ότι συγκλίνει, όταν έχει την ίδια τιμή στο 95% των χρωμοσωμάτων. Ένας πληθυσμός θεωρείται ότι έχει συγκλίνει όταν όλα τα γονίδιά του έχουν συγκλίνει.

Κατά την πρόωρη σύγκλιση, ο πληθυσμός πολύ γρήγορα συγκλίνει γύρω από κάποιο χρωμόσωμα, το οποίο όμως αποτελεί τοπικό μέγιστο. Το αποτέλεσμα είναι ο ΓΑ να μην μπορεί πλέον να ξεφύγει από αυτό το τοπικό μέγιστο, παρά μόνο με την διαδικασία της μετάλλαξης, το οποίο όμως πρακτικά έχει μηδενική πιθανότητα να συμβεί. Το φαινόμενο αυτό εμφανίζεται σε περιπτώσεις που η συνάρτηση καταλληλότητας εμφανίζει πολύ απότομες μεταβολές και έντονα τοπικά μέγιστα. Η αντιμετώπιση του φαινομένου αυτού γίνεται με δύο τρόπους. Ο πρώτος τρόπος είναι η απεικόνιση της συνάρτησης καταλληλότητας σε νέα λιγότερο απότομη συνάρτηση (fitness remapping). Ο δεύτερος τρόπος είναι να καθοριστούν μέγιστα και ελάχιστα όρια, όσον αφορά το πόσες φορές επιλέγεται ένα χρωμόσωμα προς αναπαραγωγή σε κάθε κύκλο ανανέωσης του πληθυσμού.

Η αργή σύγκλιση είναι ουσιαστικά το αντίθετο φαινόμενο της πρόωρης σύγκλισης. Κατά αυτήν, μετά από έναν μεγάλο αριθμό επαναλήψεων, ο πληθυσμός εξακολουθεί να μην έχει συγκλίνει. Το φαινόμενο εμφανίζεται όταν η συνάρτηση καταλληλότητας έχει μικρές κλίσεις, με αποτέλεσμα τα μέγιστα και τα ελάχιστά της να έχουν μικρές διαφορές. Η λύση είναι και πάλι η απεικόνιση της συνάρτησης σε μία νέα, η οποία πρέπει να έχει πιο έντονες διακυμάνσεις.

Τέλος, μια εναλλακτική προσέγγιση στο θέμα της εξέλιξης του πληθυσμού είναι η μη ανανέωση ολόκληρου του πληθυσμού σε κάθε γενιά, αλλά ένα μέρος αυτού. Ως χάσμα γενεών (generation gap) ορίζεται το ποσοστό των χρωμοσωμάτων που ανανεώθηκε ως προς το σύνολο των χρωμοσωμάτων. Στους τυπικούς ΓΑ ο συντελεστής αυτός ισούται με την μονάδα. Ωστόσο, η τάση είναι η ανανέωση μέρους του πληθυσμού και μάλιστα, στην πιο ακραία μορφή, μόνο δύο μελών. Η μέθοδος αυτή ονομάζεται μέθοδος της μερικής ανανέωσης (steady-state replacement).

Η μέθοδος της μερικής ανανέωσης προσεγγίζει πιο πολύ την πραγματικότητα, αφού εκεί συνυπάρχουν πάντα σε κάποιο βαθμό οι διαφορετικές γενιές. Μάλιστα, δίνεται η δυνατότητα στους απογόνους να ανταγωνιστούν τους γονείς τους, επικρατώντας και πάλι ο καλύτερος.

Υπάρχουν δύο θέματα στην προσέγγιση της μερικής ανανέωσης: πώς θα επιλεγούν οι γονείς που θα ζευγαρώσουν και πώς θα επιλεγεί ισάριθμος αριθμός γονέων που θα αποχωρήσουν , ώστε να αφήσουν χώρο για τους απογόνους. Οι προσεγγίσεις που ακολουθούνται είναι οι ακόλουθες τρεις:

- Επιλογή των γονέων προς αναπαραγωγή με πιθανότητα ανάλογη προς την καταλληλότητά τους και τυχαία επιλογή των γονέων που θα αποχωρήσουν
- Επιλογή των γονέων προς αναπαραγωγή τυχαία και επιλογή των γονέων που θα αποχωρήσουν με πιθανότητα αντιστρόφως ανάλογη προς την καταλληλότητά τους
- Επιλογή των γονέων προς αναπαραγωγή με πιθανότητα ανάλογη προς την καταλληλότητά τους και επιλογή των γονέων που θα αποχωρήσουν με πιθανότητα αντιστρόφως ανάλογη προς την καταλληλότητά τους

[17],[16]

2.3.5 Αναπαραγωγή

Αναπαραγωγή είναι η διαδικασία δημιουργίας δύο απογόνων από δύο γονείς χρωμοσώματα. Η αναπαραγωγή μπορεί να γίνει με διαφορετικές τεχνικές. Η σημαντικότερη από τις τεχνικές αυτές είναι ο συνδυασμός της διασταύρωσης (cross-over) και της μετάλλαξης (mutation).

Κατά τη διασταύρωση, για κάθε ζευγάρι επιλεγμένων γονέων επιλέγεται ένα σημείο διασταύρωσης (cross-point) με τυχαίο τρόπο, το οποίο είναι ένας αριθμός N, από το 1 μέχρι το μήκος L των bit-strings. Με βάση το σημείο διασταύρωσης, από δύο γονείς A και B παράγονται δύο απόγονοι με τον εξής τρόπο: Ο πρώτος απόγονος παράγεται από τα πρώτα N γονίδια του A και τα L-N γονίδια του B, ενώ ο δεύτερος παράγεται από τα πρώτα N γονίδια του B και τα L-N γονίδια του A.

Έστω για παράδειγμα οι γονείς A=**011001** και B=110010 και σημείο διασταύρωσης το 4. Οι δύο απόγονοι θα είναι οι: **0110**10 και 1100**01**.

Η μετάλλαξη εκτελείται στους απογόνους που παράγονται από τη διασταύρωση. Με βάση μία πολύ

μικρή πιθανότητα, κάθε σύμβολο των bit-strings των απογόνων αλλάζει τιμή.

Για παράδειγμα, έστω ένας πληθυσμός που αποτελείται από τα χρωμοσώματα που δείχνει ο Πίνακας 1. Στην 2η στήλη του πίνακα φαίνονται οι τιμές της συνάρτησης καταλληλότητας για κάθε χρωμόσωμα, ενώ στην 3η φαίνονται οι πιθανότητες επιλογής προς αναπαραγωγή κάθε χρωμοσώματος στο διάστημα [0,1]. Οι τιμές της 3ης στήλης προκύπτουν από το λόγο της καταλληλότητας του κάθε χρωμοσώματος προς το άθροισμα των καταλληλοτήτων όλων των χρωμοσωμάτων του πληθυσμού. Έστω ότι το αποτέλεσμα της διαδικασίας επιλογής είναι το [Α,Β,Β,Γ] και τα ζευγάρια που σχηματίζονται είναι τα (Β,Α) και (Β,Γ). Ως σημείο διασταύρωσης του πρώτου ζευγαριού επιλέγεται τυχαία το 4, ενώ ως σημείο διασταύρωσης του δεύτερου το 9.

Table 1: Παράδειγμα πληθυσμού χρωμοσωμάτων

Υποψήφιες Λύσεις(bit-string)	Τιμή συνάρτησης καταλληλότητας	Πιθανότητα επιλογής
Α=000110010111	8	0.32
Β=111010101100	6	0.24
Γ=001110101001	6	0.24
Δ=111011011100	5	0.2

Τη διαδικασία αναπαραγωγής σκιαγραφεί ο Πίνακας 2. Στη γραμμή “Διασταύρωση” φαίνεται το αποτέλεσμα της διασταύρωσης, ενώ στην γραμμή “Μετάλλαξη” έχει εκτελεστεί και η μετάλλαξη. Τα σύμβολα που είναι πιο έντονα τυπωμένα στην πέμπτη γραμμή είναι αυτά που έχουν επηρεαστεί από τη μετάλλαξη.

Table 2: Αναπαραγωγή χρωμοσωμάτων

Γονέας 1	(Β)1110-10101100	(Β)1110-10101100	(Β)111010101-100	(Β)111010101-100
Γονέας 2	(Α)0001-10010111	(Α)0001-10010111	(Γ)001110101-001	(Γ)001110101-001
Απόγονος	1ος	2ος	3ος	4ος
Διασταύρωση	1110-10010111	0001-10101100	111010101-001	001110101-100
Μετάλλαξη	1110-10010111	0001-10101100	111**1**10101-001	001110101-10**1**

Από τα παραπάνω φαίνεται ότι η διασταύρωση είναι η πιο σημαντική τεχνική για την γρήγορη εξερεύνηση όλου του χώρου αναζήτησης. Από την άλλη πλευρά, η μετάλλαξη παρέχει μια πολύ μικρή πιθανότητα τυχαίας αναζήτησης, εξασφαλίζοντας έστω και με απειροελάχιστη πιθανότητα, ότι όλα τα σημεία του χώρου αναζήτησης έχουν πιθανότητα επίσκεψης. [17],[16]

2.3.6 Κριτήρια τερματισμού

Η βασική αρχή των αλγορίθμων, είναι ότι αποτελούν μία πεπερασμένη διαδικασία για την επίλυση ενός προβλήματος. Αυτό θα πρέπει επομένως να ισχύει και για τους Γενετικούς Αλγορίθμους. Το φυσικό τους υπόβαθρο μπορεί να εκτελείται για πάντα, αλλά δεν είναι δυνατόν στα προβλήματα που καλούνται να επιλύσουν ο προγραμματιστής να περιμένει για πάντα, ώστε να πάρει μία λύση. Επιπλέον, σαν προγραμματιστικές διαδικασίες, καταλαμβάνουν γενικά μεγάλο μέρος της μνήμης του υπολογιστή και απαιτούν σημαντικό μέρος της προγραμματιστικής του ισχύος. Επομένως θα πρέπει να υπάρχει κάποιος αποδοτικός τρόπος τερματισμού του αλγορίθμου, μέσω ενός έξυπνου κριτηρίου τερματισμού. Το κριτήριο τερματισμού θα πρέπει να είναι τέτοιο ώστε:

- Να μην επιτρέπει την εκτέλεση επαναλήψεων που δεν οδηγούν σε κάποιο καλύτερο αποτέλεσμα
- Να διακόπτουν τον αλγόριθμο όταν με κάποια βεβαιότητα έχει βρεθεί ο στόχος (ολικό μέγιστο/ελάχιστο)

Είναι προφανές ότι ένα κριτήριο δεν μπορεί με βεβαιότητα να ικανοποιεί απόλυτα και τους δύο παραπάνω στόχους, καθώς δεν μπορούμε με ακρίβεια την λύση του εκάστοτε προβλήματος. Συνήθως τα κριτήρια τερματισμού είναι τέτοια που προσπαθούν να ικανοποιούν στον μεγαλύτερο βαθμό το ένα ή και τα δύο από τα παραπάνω κριτήρια. [17],[16]

2.3.6.1 Αριθμός επαναλήψεων

Είναι ο πιο απλός και γενικός τρόπος τερματισμού των Γενετικών Αλγορίθμων. Εκ των προτέρων θεωρούμε πως ο μέγιστος αριθμός επαναλήψεων (γενιών) είναι ίσος με έναν αριθμό. Συνήθως

χρησιμοποιείται σαν όριο κάποιες εκατοντάδες γενιών. Μόλις ο αλγόριθμος φτάσει σε αυτό το όριο τερματίζει. Το κριτήριο αυτό, αν και απλό στην υλοποίησή του, έχει πολλά προβλήματα:

- Δεν χρησιμοποιεί καμία πληροφορία για το πρόβλημα προς επίλυση
- Αν ο αριθμός των επαναλήψεων ολοκληρωθεί, χωρίς να φτάσουμε στην απαιτούμενη λύση, τότε έχουμε πρόωρη διακοπή του αλγορίθμου
- Αν η λύση βρεθεί σε κάποια από τις αρχικές γενιές, τότε μεγάλο μέρος των υπολογιστικών πόρων ξοδεύεται σε μια άσκοπη διαδικασία

Αυτό το κριτήριο τερματισμού δεν χρησιμοποιείται σχεδόν ποτέ μόνο του, αλλά σε συνδυασμό με κάποιο άλλο έξυπνο κριτήριο, το οποίο λαμβάνει υπ' όψιν την φύση του προβλήματος. Αν το κριτήριο ικανοποιείται σε κάποια από τις αρχικές γενιές τότε ο αλγόριθμος τερματίζει, διαφορετικά αν το έξυπνο κριτήριο δεν ικανοποιείται και εξαντληθεί ο αριθμός των επαναλήψεων, τότε τερματίζει ο Γενετικός Αλγόριθμος.

2.3.6.2 Προσέγγιση στόχου

Σε πολλές περιπτώσεις ο στόχος είναι γνωστός, ή τουλάχιστον μία συνοπτική περιγραφή του. Στην περίπτωση αυτή μπορούμε να τερματίσουμε τον αλγόριθμο, όταν αυτός ο στόχος έχει προσεγγιστεί από τον Γενετικό Αλγόριθμο. Προφανώς το κριτήριο αυτό μπορεί να χρησιμοποιηθεί σε συγκεκριμένες περιπτώσεις, όπου υπάρχει σαφής γνώση του αντικειμενικού στόχου.

2.3.6.3 Κριτήριο ομοιότητας

Το κριτήριο ομοιότητας είναι ένα ακόμα κριτήριο που χρησιμοποιείται συχνά. Σε αυτό υπολογίζεται η μέση απόσταση μεταξύ όλων των χρωμοσωμάτων, και αν αυτή είναι μικρότερη από κάποιο προκαθορισμένο όριο, τότε ο αλγόριθμος τερματίζει. Με μαθηματικό συμβολισμό, μπορούμε να πούμε ότι ο αλγόριθμος τερματίζει όταν:

$\sum_{(i=0)}^{M} \sum_{(j \neq i,\ j=1)} \|x_i - x_j\| \leq e$, όπου M το πλήθος των χρωμοσωμάτων.

Αυτό το κριτήριο χρησιμοποιείται αρκετά συχνά και είναι αρκετά γενικό, αφού δεν εξαρτάται ούτε από αντικειμενικό πρόβλημα ούτε από την χρησιμοποιούμενη κωδικοποίηση, μιας και με την κατάλληλη επιλογή της νόρμας, μπορεί να χρησιμοποιηθεί σε δυαδική, ακέραια ή δεκαδική αναπαράσταση. Ωστόσο έχει τα ακόλουθα μειονεκτήματα:

- Έχει αργό υπολογισμό, ειδικά για μεγάλα σε μέγεθος χρωμοσώματα ή αρκετά σε πλήθος
- Σε πολλές περιπτώσεις είναι δυνατόν η ποσότητα της παραπάνω εξίσωσης να είναι μεγάλη, αλλά ο αλγόριθμος να έχει βρει την λύση που ψάχνει. Σε αυτήν την περίπτωση θα δαπανηθεί χρόνος για την ικανοποίηση του κριτηρίου τερματισμού
- Σε πολλές περιπτώσεις μπορεί το παραπάνω κριτήριο να μην ικανοποιηθεί σε εύλογο χρονικό διάστημα

2.3.6.4 Διαφορά καλύτερου-χειρότερου

Αυτό το κριτήριο χρησιμοποιείται αρκετά συχνά, ειδικά στην περίπτωση της δεκαδικής αναπαράστασης. Σε κάθε επανάληψη υπολογίζεται η καταλληλότητα του καλύτερου και του χειρότερου χρωμοσώματος (f_{max} και f_{min} αντίστοιχα). Ο αλγόριθμος τερματίζει όταν:

$$|f_{max} - f_{min}| \leq e$$

Το κριτήριο αυτό μοιάζει αρκετά με το κριτήριο ομοιότητας, αλλά είναι αρκετά πιο γρήγορο στον υπολογισμό, καθώς δεν απαιτεί επιπλέον πράξεις. Ωστόσο και αυτό πάσχει από τα μειονεκτήματα του κριτηρίου ομοιότητας.

Τα παραπάνω κριτήρια είναι “αντικειμενικά”, δηλαδή μπορούν να εφαρμοστούν σε όλους τους Γενετικούς Αλγορίθμους, ανεξαρτήτως του εκάστοτε αντικειμενικού προβλήματος και δεν λαμβάνουν υπ' όψιν το υπολογιστικό κόστος τόσο του αλγορίθμου όσο και του ίδιου του υπολογισμού τους, το οποίο όπως ειπώθηκε μπορεί να είναι αρκετά μεγάλο. Πολύ συχνά επίσης παρατηρείται το φαινόμενο ένας πληθυσμός να παραμείνει στατικός για έναν αριθμό γενεών προτού να βρεθεί ένα ανώτερο άτομο, με αποτέλεσμα η εφαρμογή των συμβατικών κριτηρίων τερματισμού να γίνεται προβληματική. Για τους λόγους αυτούς, σε πολλές περιπτώσεις χρησιμοποιούνται ποιοτικά κριτήρια του ίδιου του

αλγορίθμου, ώστε να οριστεί ο τερματισμός του. Τέτοιου είδους κριτήρια είναι τα εξής:

- Ο ΓΑ να τερματίζει όταν ο μέσος όρος της τιμής καταλληλότητας για όλα τα χρωμοσώματα ενός πληθυσμού και για έναν διαδοχικό αριθμό πληθυσμών δεν βελτιώνεται
- Ο ΓΑ να τερματίζει όταν το ποσοστό των χρωμοσωμάτων που δεν μεταβάλλονται έχει ξεπεράσει ένα συγκεκριμένο ποσοστό

2.3.7 Το Πρόβλημα του πλανόδιου πωλητή

Το πρόβλημα του πλανόδιου πωλητή συνίσταται στην εύρεση της σειράς με την οποία ένας πωλητής πρέπει να περάσει από όλες τις πόλεις ενός συνόλου πόλεων και να επιστρέψει στην αρχική, ώστε να έχει το μικρότερο κόστος, όπως αυτό εκφράζεται κάθε φορά (π.χ. χρόνος, χρήματα κλπ).

Το πρώτο θέμα που τίθεται για την αντιμετώπιση του συγκεκριμένου προβλήματος με γενετικούς αλγορίθμους είναι η μορφή των χρωμοσωμάτων. Για το πρόβλημα αυτό συνηθίζεται η χρήση διανυσμάτων ακεραίων αριθμών μήκους ίσου με το πλήθος των πόλεων. Έτσι για παράδειγμα ένα χρωμόσωμα θα έχει την μορφή $v=(i_1 i_2 ... i_n)$ όπου $i_1, i_2, ..., i_n \in 1...n$ και $i_j \neq i_k$ για $j \neq k$.

Ύστερα από την επιλογή της κωδικοποίησης, δημιουργείται ο αρχικός πληθυσμός χρωμοσωμάτων παίρνοντας τυχαία ένα σύνολο από συνδυασμούς των πόλεων. Η αξιολόγηση των χρωμοσωμάτων είναι απλή διαδικασία, αφού απαιτεί απλά την πρόσθεση των επιμέρους κόστων για την μετακίνηση μεταξύ των γειτονικών πόλεων, με την σειρά που αυτές περιγράφονται στο χρωμόσωμα.

Το δεύτερο θέμα που τίθεται είναι οι τεχνικές της διασταύρωσης και της μετάλλαξης.. Στο πρόβλημα αυτό μπορεί να προκύψουν μη-έγκυρα χρωμοσώματα, μιας και μπορεί στο ίδιο χρωμόσωμα να εμφανιστεί μία πόλη παραπάνω από μία φορές. Για την αντιμετώπιση αυτού του προβλήματος υπάρχουν δύο δυνατότητες : Είτε ύστερα από κάθε αναπαραγωγή να ελέγχονται οι απόγονοι και να επιδιορθώνονται όσα χρωμοσώματα είναι μη-έγκυρα ή να χρησιμοποιηθούν τροποποιημένες τεχνικές μετάλλαξης και διασταύρωσης, οι οποίες να δίνουν πάντα έγκυρα χρωμοσώματα.

Μια πρώτη σκέψη θα ήταν να πραγματοποιηθεί η αναπαραγωγή όχι από ζεύγη χρωμοσωμάτων, αλλά

από μεμονωμένα χρωμοσώματα. Για παράδειγμα αντιμεταθέτοντας μερικές πόλεις ενός χρωμοσώματος καταλήγουμε πάντα σε έγκυρη λύση. Ωστόσο, έχει αποδειχτεί ότι η χρήση ενός μόνο γονέα για την δημιουργία ενός απογόνου όχι μόνο δεν οδηγεί στην σύγκλιση του πληθυσμού στο βέλτιστο χρωμόσωμα, αλλά συχνά δεν παράγει καν καλά χρωμοσώματα.

Η δύναμη των γενετικών αλγορίθμων προκύπτει από την διασταύρωση γονέων με υψηλή τιμή καταλληλότητας. Μία δυνατή διασταύρωση για το πρόβλημα του πλανόδιου πωλητή είναι η εξής: Αρχικά επιλέγονται δύο τυχαίες ακολουθίες γονιδίων ίδιου μήκους από κάθε χρωμόσωμα, έστω οι ακόλουθες:

$$p_1 = (123 \vee 4567 \vee 89)$$

$$p_2 = (452 \vee 1876 \vee 93)$$

όπου τα σύμβολα | περιορίζουν τις επιλεγείσες ακολουθίες. Στη συνέχεια δημιουργείται η αρχική μορφή των απογόνων, κρατώντας μόνο τις επιλεγείσες ακολουθίες και αφήνοντας κενές τις υπόλοιπες θέσεις:

$$o_1 = (xxx \vee 4567 \vee xx)$$

$$o_2 = (xxx \vee 1876 \vee xx)$$

Στη συνέχεια, ξεκινώντας από το δεύτερο σύμβολο | του ενός γονέα, οι πόλεις του άλλου γονέα αντιγράφονται στις ίδιες θέσεις, παραλείποντας αυτές που υπάρχουν ήδη. Μόλις γεμίσουν οι θέσεις στο τέλος η συμπλήρωση συνεχίζεται από την αρχή. Για παράδειγμα, για την συμπλήρωση των θέσεων του απογόνου o_1, θα ληφθούν υπόψη τα στοιχεία του χρωμοσώματος p_2 με τη σειρά:

9 3 4 5 2 1 8 7

Αφαιρώντας όμως τις πόλεις που υπάρχουν ήδη στο o_1, απομένουν οι:

9 3 2 1 8

Αυτές συμπληρώνουν τις θέσεις του χρωμοσώματος o_1, ξεκινώντας αμέσως μετά το 7. Τελικά οι δύο απόγονοι που προκύπτουν είναι οι εξής:

$$o_1 = (218 \vee 4567 \vee 93)$$

$$o_2 = (345 \vee 1876 \vee 92)$$

Όσον αφορά την διαδικασία της μετάλλαξης και προκειμένου να οδηγεί σε έγκυρα αποτελέσματα, παίρνει την μορφή της αντιμετάθεσης δύο γειτονικών πόλεων. [16]

2.3.8 Τεχνικά εργαλεία

Η ανάπτυξη των Γενετικών Αλγορίθμων, καθώς και των ρουτινών για την επεξεργασία των αποτελεσμάτων έγινε στο Matlab 7.4.0 (R2007α) και σε λειτουργικό περιβάλλον GNU/Linux, διανομή Ubuntu και έκδοση 9.4. Η επιλογή των εργαλείων αυτών έχει άμεση σχέση με τις ιδιαιτερότητες της εργασίας, λόγω του ιδιαίτερα μεγάλου πλήθους υπολογισμών που χρειαζόταν καθώς και της εκτέλεσης των υπολογισμών αυτών σε οικιακό δίκτυο.

Για να υλοποιηθεί η εργασία αποτελεσματικά χρειάζονταν εργαλεία λογισμικού που να καλύπτουν τις ανάγκες μας, οι σημαντικότερες απ’ τις οποίες είναι:

- Χαμηλό κόστος και δυνατότητα για ελεύθερη αναδιανομή.
- Πολύ καλή και άμεση υποστήριξη διανυσμάτων και αλγεβρικών δομών.
- Μηχανισμούς για την αναλυτική καταγραφή (logging) των πειραμάτων.
- Μηχανισμούς για να μπορεί το λογισμικό μας να εκκινείται αυτόματα καθώς και να μπορεί να συνεχίζει μετά από βίαιη διακοπή.
- Αυξημένες δυνατότητες γραφικών απεικονίσεων των δεδομένων μας.
- Δυνατότητα παράλληλης εκτέλεσης, τόσο σε πολλά νήματα εκτέλεσης στον ίδιο υπολογιστή, όσο και σε πολλούς υπολογιστές.

Κεφάλαιο 3 – Υλοποίηση

Το ζητούμενο της εργασίας αυτής είναι η κατασκευή Γενετικών Αλγορίθμων, οι οποίοι θα προτείνουν βέλτιστα μείγματα αναφοράς, τα οποία αποτελούνται από κυτταρικές σειρές. Ως δεδομένα πάνω στα οποία θα κάνουμε την βελτιστοποίηση αυτή έχουμε 83 κυτταρικές σειρές, οι οποίες αποτελούνται από 22283 γονίδια η καθεμία. Τα γονίδια αυτά είναι ίδια σε κάθε κυτταρική σειρά και "εκφράζονται" με μία τιμή έντασης.

Ένα μείγμα αναφοράς (refernce mix) αποτελείται από έναν συνδυασμό κυτταρικών σειρών, συνήθως από 10 και παραπάνω. [10],[9] Το μείγμα που καλούμαστε να δημιουργήσουμε πρέπει να έχει δύο βασικά χαρακτηριστικά, αυτά της ομοιομορφίας και της μέγιστης δυνατής έκφρασης. Θεωρούμε ότι ένα γονίδιο εκφράζεται όταν η τιμή της έντασής του βρίσκεται πάνω από ένα κατώφλι. Το κατώφλι αυτό θα υπολογιστεί και στη συνέχεια τα γονίδια τα οποία σε καμία κυτταρική σειρά δεν ξεπερνάνε το κατώφλι αυτό δεν θα λαμβάνουν μέρος στη διαδικασία, απορρίπτονται. Στη συνέχεια, αφού έχουμε προσδιορίσει τα γονίδια τα οποία εκφράζονται έστω και σε μία κυτταρική σειρά, θα αρχίσουμε να "ψάχνουμε" τους συνδυασμούς κυτταρικών σειρών με την μέγιστη ομοιομορφία και έκφραση.

Ως ομοιομορφία νοείται η μικρή απόκλιση στις τιμές των εντάσεων των γονιδίων που συμμετέχουν στο υποψήφιο μείγμα. Ας θεωρήσουμε ότι ο παρακάτω πίνακας είναι ένα πρότυπο μείγμα. Οι στήλες περιέχουν τις κυτταρικές σειρές που συμμετέχουν στο μείγμα αυτό και οι γραμμές τα γονίδια, τα οποία έχουν παραμείνει μετά την απόρριψη. Κάθε ένα από τα γονίδια έχει μία τιμή έντασης σε κάθε κυτταρική σειρά.

Έστω s_n μία κυτταρική σειρά με $n=[1,83]$ και g_i ένα γονίδιο με $i=[1,22283]$. Τότε ένα μείγμα αναφοράς (reference mix) είναι ένας πίνακας $R=(r_i)$, όπου $n \geqslant 10$ και i ισούται με τον αριθμό των γονιδίων τα οποία έχουν ξεπεράσει το κατώφλι έντασης. Δηλαδή, κάθε υποψήφια λύση

$m_k=\sum_{n=1}^{N} r_n s_n$ για i= 1 έως τον αριθμό των γονιδίων που έχουν ξεπεράσει το κατώφλι έκφρασης.

	s_1	s_2	...	s_n
g_1				
g_2				
...				
g_i				

Table 3: Αναπαράσταση υποψήφιας λύσης

Η ομοιομορφία, επομένως, νοείται και μετράται για κάθε γονίδιο κατά μήκος των γραμμών του πίνακα. Ο τρόπος που μετράμε την ομοιομορφία είναι μέσω της διασποράς. Ταυτόχρονα, μας ενδιαφέρει και η όσο το δυνατόν μεγαλύτερη έκφραση, δηλαδή οι τιμές των εντάσεων των γονιδίων μέσα στις κυτταρικές σειρές που αποτελούν το μείγμα να είναι όσο το δυνατόν μεγαλύτερες, με σεβασμό στο θέμα της ομοιομορφίας. Επομένως, τα κριτήρια βάση των οποίων πρέπει να αξιολογούμε τα γονίδια του Γενετικού Αλγορίθμου είναι ανά στήλη (ανά γονίδιο μέσα στις κυτταρικές σειρές) και όχι ανά στήλη (ανά κυτταρική σειρά).

Ο Γενετικός Αλγόριθμος που κατασκευάζουμε έχει σαν σκοπό να μας δώσει αυτά τα καλύτερα μείγματα. Η λύση του προβλήματος θα είναι ένας πίνακας που θα έχει την μορφή του Πίνακα 3. Δηλαδή, θα είναι ένας πίνακας διαστάσεων i*n, όπου i-γραμμές θα είναι όλα τα γονίδια που έχουν περάσει το κατώφλι απόρριψης και n-στήλες θα είναι ο αριθμός των κυτταρικών σειρών που θα συμμετέχουν στο μείγμα. Για το λόγο αυτό, το κάθε “γονίδιο” του ΓΑ που θα παράγουμε και θα εξετάζουμε θα είναι μία n-άδα κυτταρικών σειρών.

3.1 Περιγραφή δεδομένων

Τα δεδομένα τα οποία μας δόθηκαν αποτελούνται από 83 κυτταρικές σειρές, οι οποίες περιλαμβάνουν 22283 γονίδια. Τα γονίδια αυτά αποτελούν το NCI-60 cancer cell panel, το οποίο αναπαριστά εννέα ιστούς και “περιλαμβάνει” καρκίνο του στήθους, των ωοθηκών, του προστάτη, του εντέρου, των πνευμόνων, του κεντρικού νευρικού συστήματος, λευχαιμία και μελάνωμα. Τα δεδομένα αυτά είναι δημοσίως διαθέσιμα και έχουν παραχθεί από την εταιρεία Affymetrix. [19]

3.2 Προεπεξεργασία δεδομένων

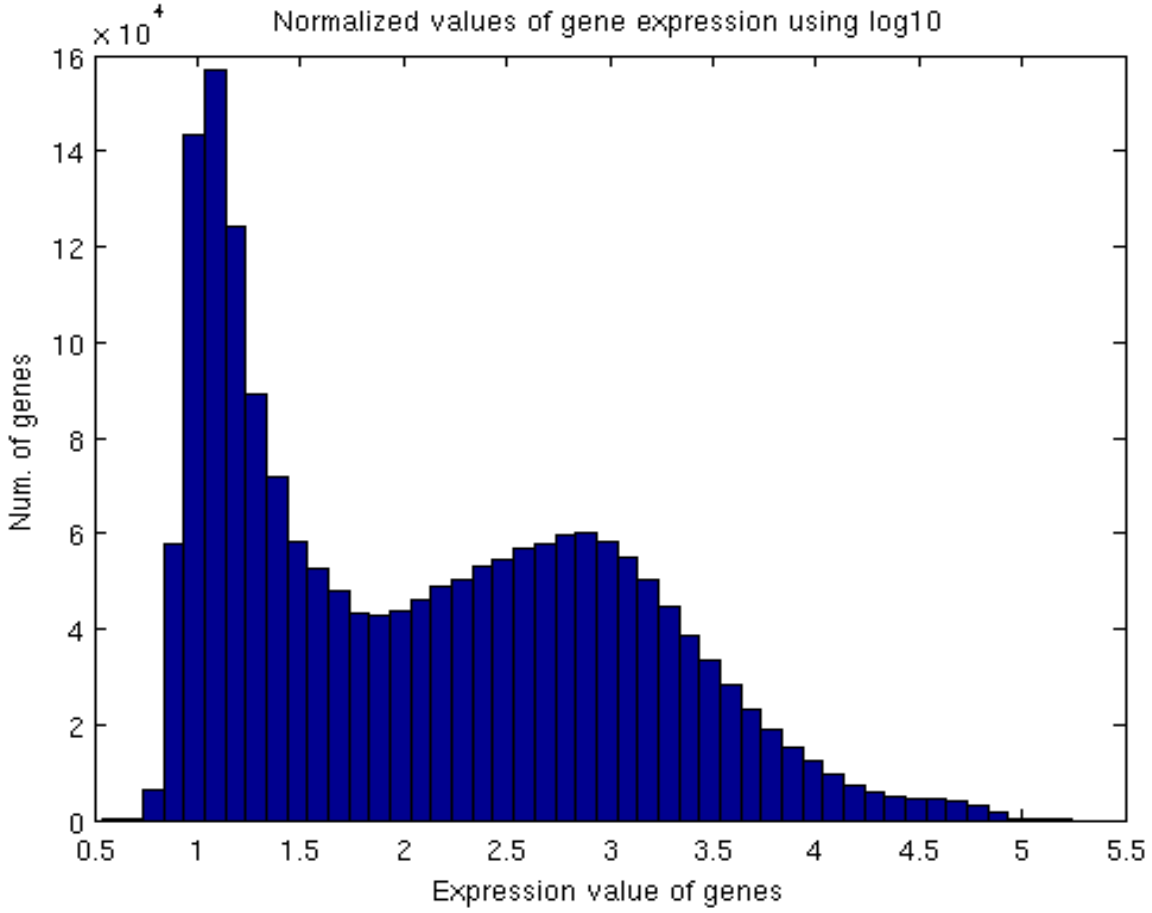
3.2.1 Κατώφλι έκφρασης γονιδίου

Όπως ειπώθηκε και παραπάνω σκοπός της εργασίας αυτής είναι να βρούμε τον συνδυασμό των κυτταρικών σειρών για τον οποίο η πλειοψηφία των γονιδίων δίνει σήμα πάνω από ένα κατώφλι. Τα γονίδια αυτά θεωρούμε ότι εκφράζονται πάνω στην μικροσυστοιχία, ενώ όσα βρίσκονται κάτω από το κατώφλι όχι. Το πρώτο ζήτημα το οποίο κληθήκαμε να αντιμετωπίσουμε είναι ο προσδιορισμός του κατωφλίου έκφρασης.

Όπως παρουσιάστηκε και παραπάνω, η a priori γνώση του ποια είναι η τιμή για την οποία θεωρούμε ότι τα γονίδια εκφράζονται δεν είναι δυνατή, αλλά εξαρτάται από τον τρόπο κατασκευής και την καθαρότητα του πλακιδίου. Για να προσδιορίσουμε την τιμή έκφρασης των γονιδίων στο dataset που μας δόθηκε, η διαδικασία που ακολουθήσαμε ήταν η εξής:

Αρχικά κανονικοποιήσαμε τις τιμές έντασης των όλων των γονιδίων μέσα σε όλες τις κυτταρικές σειρές (δηλαδή όλα τα δεδομένα μας), χρησιμοποιώντας τον λογάριθμο με βάση 10. Στη συνέχεια προσδιορίσαμε την μέγιστη και την ελάχιστη τιμή των εντάσεων των γονιδίων, και χρησιμοποιώντας τις τιμές αυτές σαν ανώτατο και κατώτατο όριο αντίστοιχα, πήραμε το ιστόγραμμα των τιμών των εντάσεων των γονιδίων στη μικροσυστοιχία. Το αποτέλεσμα αυτής της επεξεργασίας παρουσιάζεται στο Σχεδιάγραμμα 5.

Μέσω του ιστογράμματος αυτού βλέπουμε ότι στην τιμή 100 της έντασης των γονιδίων παρουσιάζεται ένα τοπικό ελάχιστο, κοντά στην τιμή 2 του άξονα x, το οποίο αποτελεί και το κρίσιμο σημείο (critical point) το οποίο θέλουμε να προσδιορίσουμε. Επειδή η κανονικοποίηση των τιμών έχει γίνει με την χρήση του λογαρίθμου με βάση 10, η τιμή 2 του άξονα των x αντιστοιχεί στην τιμή έντασης 100. Με την διαδικασία αυτή, για τα δεδομένα τα οποία έχουμε, καταφέραμε να προσδιορίσουμε ότι τα γονίδια εκφράζονται για τιμές έντασης άνω του 100, η οποία θα αποτελέσει και το κατώφλι απόρριψης γονιδίων, όπως αναφέραμε και σε προηγούμενη παράγραφο.



Σχεδιάγραμμα 7: Ιστόγραμμα κανονικοποιημένων τιμών έντασης των γονιδίων

3.2.2 Απόρριψη γονιδίων

Μέσω της διαδικασίας που περιγράψαμε στην προηγούμενη παράγραφο, διαπιστώσαμε ότι η τιμή της έντασης για την οποία θεωρούμε ότι ένα γονίδιο εκφράζεται είναι 100. Ένα ερώτημα το οποίο έπρεπε να απαντηθεί άμεσα, επομένως, είναι αν υπάρχουν γονίδια τα οποία δεν εκφράζονται σε καμία από τις 83 κυτταρικές σειρές.

Γονίδια τα οποία δεν εκφράζονται σε καμία από τις κυτταρικές σειρές μετά την εφαρμογή του γενετικού αλγορίθμου θα συνεχίζουν να μην εκφράζονται. Ο λόγος είναι ότι κατά την εφαρμογή του γενετικού κάθε κυτταρική σειρά που επιλέγεται, συμμετέχει μία μόνο φορά, με πολλαπλασιαστή

δηλαδή 1. Ο ίδιος ο αλγόριθμος άλλωστε έχει σαν σκοπό την εύρεση του καλύτερου συνδυασμού κυτταρικών σειρών, ώστε να πληρούνται οι συνθήκες έκφρασης και ομοιομορφίας, όπως έχουν προσδιοριστεί, και όχι η μείξη των κυτταρικών σειρών. Επομένως, η ύπαρξη γονιδίων τα οποία δεν εκφράζονται ποτέ δεν έχει κανένα νόημα για την εργασία αυτή, και επιπλέον ρίχνει την απόδοση του αλγορίθμου.

Μέσω του Matlab διαπιστώσαμε ότι από τις 22283 γονίδια που περιλαμβάνονται σε κάθε κυτταρική σειρά, εκφράζονται τα 17012, δηλαδή δεν εκφράζεται περίπου το 23% των γονιδίων. Τα γονίδια αυτά, αφού εντοπίστηκαν, αφαιρέθηκαν από τον πίνακα των δεδομένων.

3.3 Κανονικοποίηση δεδομένων

Η ανάγκη κανονικοποίησης των τιμών της έντασης των γονιδίων προέκυψε από την ανάγκη για μικρότερη πολυπλοκότητα υπολογισμών. Οι τιμές έντασης των γονιδίων στις δοσμένες κυτταρικές σειρές είναι της τάξης των 3ων, 4ων ακόμα και 5ων ψηφίων. Αριθμοί μικρότερης τάξης ψηφίων, όμως, μελετούνται με μεγαλύτερη ευκολία και με πιο γρήγορους υπολογισμούς.

Για τον λόγο αυτόν κανονικοποιούμε τις τιμές έντασης των γονιδίων στον νέο πίνακα δεδομένων, αυτόν δηλαδή που προέκυψε από την αφαίρεση των γονιδίων με ένταση μικρότερη από το κατώφλι που προσδιορίσαμε, χρησιμοποιώντας ως μέσο κανονικοποίησης τον δυαδικό λογάριθμο ($\log_2$). Από το σημείο αυτό και μετά, οι πράξεις και διαδικασίες που ακολουθούν αφορούν τα κανονικοποιημένα δεδομένα και μόνο.

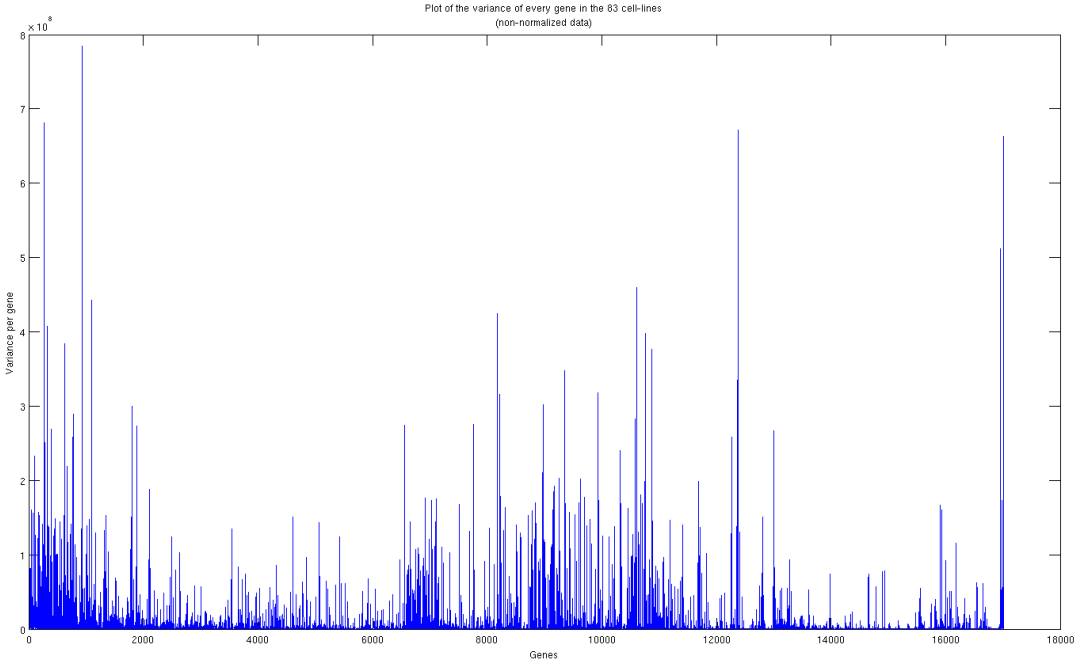
3.4 Ομοιομορφία

Όπως αναφέρθηκε και στην περιγραφή του σκοπού της εργασίας, ένα από τα κύρια στοιχεία που πρέπει να πληρεί το μείγμα αναφοράς είναι αυτό της ομοιομορφίας. Τα γονίδια τα οποία εκφράζονται, δηλαδή, πρέπει να εκφράζονται και με τον ίδιο τρόπο σε όλο το μείγμα. Η ομοιομορφία ανά γονίδιο συνεπάγεται μηδενική διασπορά τιμής έντασης του γονιδίου σε όλο το μείγμα. Η διασπορά αποτελεί ένα μέσο εκτίμησης για το πόσο κοντά ή μακριά κατανέμονται οι τιμές του μείγματος γύρω από την

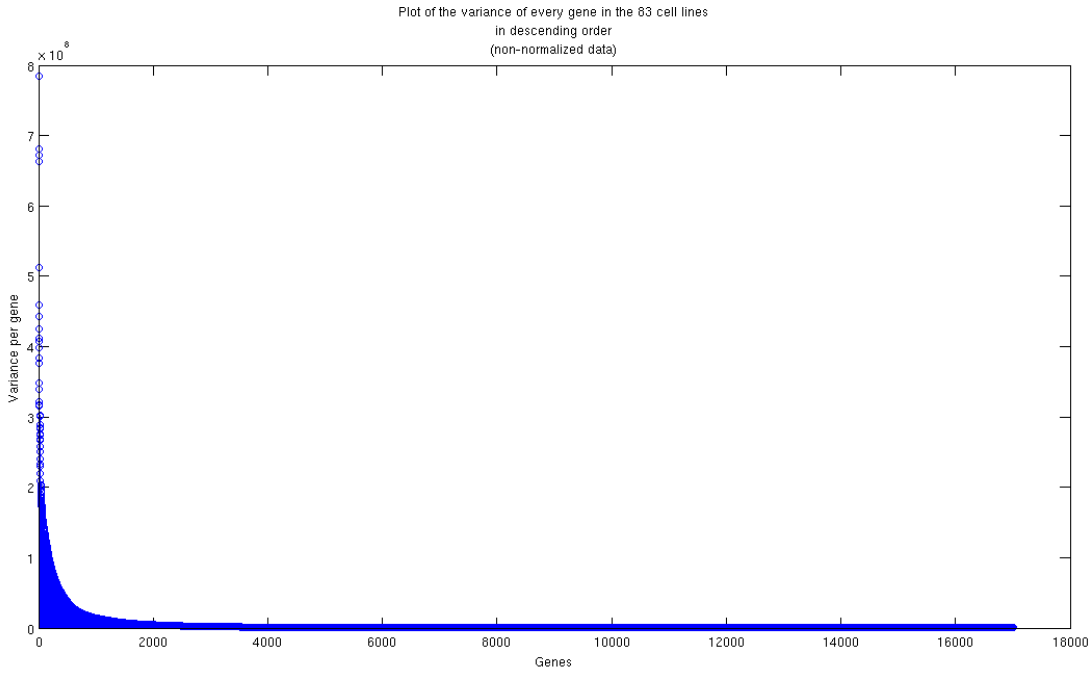
μέση τιμή. Ο τύπος της διασποράς (variance) είναι ο εξής:

$$\sigma^2 = mean(x^2) - (mean(x))^2 \forall g_i$$

Για παράδειγμα αναφέρουμε το γονίδιο 1007_s_at, η διασπορά του οποίου στις 83 κυτταρικές σειρές υπολογιζόμενη από το Matlab είναι var(data(1,:))=7.5814e+06. Επιλέγοντας τις πρώτες 20 από τις 83 κυτταρικές σειρές, για το γονίδιο αυτό η διασπορά πλέον υπολογίζεται var(data(1,1:20))=4.1367e+06, ενώ αν επιλέξουμε τις κυτταρικές σειρές 10 έως 30, τότε έχουμε var(data(1,10:30))=3.6504e+06. Βλέπουμε, επομένως, ότι ανάλογα με τις επιλογές κυτταρικών σειρών που κάνουμε η διασπορά των τιμών της έντασης του γονιδίου μεταβάλλεται. Σκοπός μας είναι, μέσω των κατάλληλων συναρτήσεων καταλληλότητας, ο ΓΑ να επιλέξει κυτταρικές σειρές που θα μειώνουν όσο γίνεται περισσότερο την διασπορά των γονιδίων, δηλαδή επιλογή των κυτταρικών σειρών στις οποίες η διασπορά της τιμής της έντασης των γονιδίων στο μείγμα να είναι κοντά στο 0.



Σχεδιάγραμμα 8: Διάγραμμα τιμών διασποράς κάθε γονιδίου στις 83 κυτταρικές σειρές (μη κανονικοποιημένα δεδομένα)



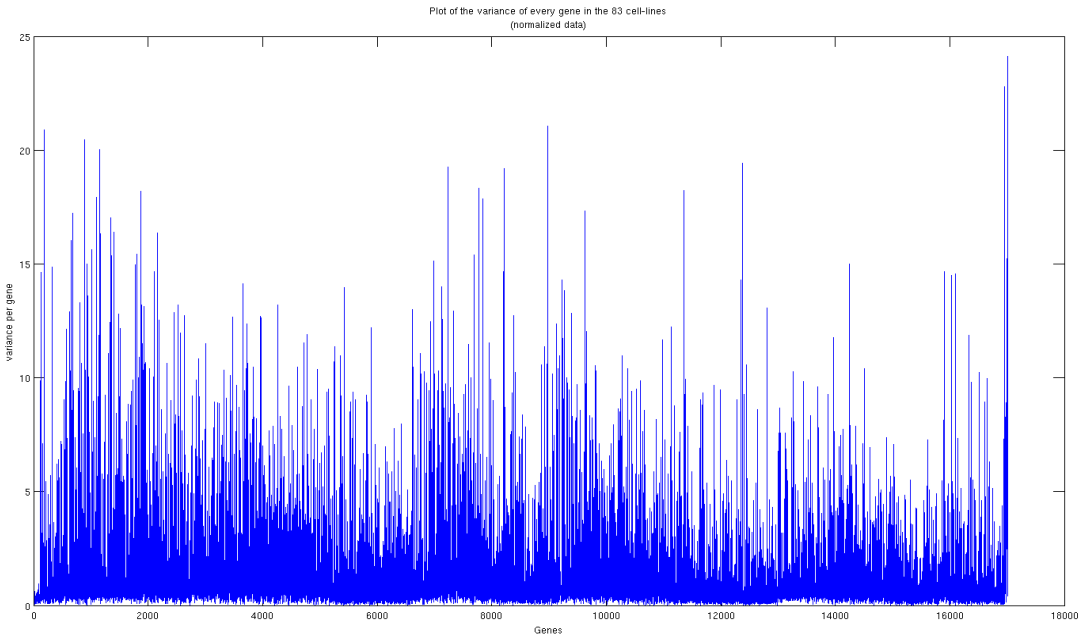
Σχεδιάγραμμα 9: Διάγραμμα τιμών διασποράς κάθε γονιδίου στις 83 κυτταρικές σειρές σε φθίνουσα σειρά (μη κανονικοποιημένα δεδομένα)

Στο Σχεδ.6, βλέπουμε για κάθε γονίδιο που πέρασε το κατώφλι έκφρασης την διασπορά του μέσα στις 83 κυτταρικές σειρές, ενώ στο Σχεδ.7 έχουμε σε φθίνουσα σειρά τις διασπορές των γονιδίων μέσα στις κυτταρικές σειρές. Και στις δύο προαναφερόμενες γραφικές παραστάσεις, οι τιμές της έντασης των γονιδίων βάση των οποίων υπολογίζουμε την διασπορά του κάθε γονιδίου δεν έχουν υποστεί κανενός είδους κανονικοποίηση.

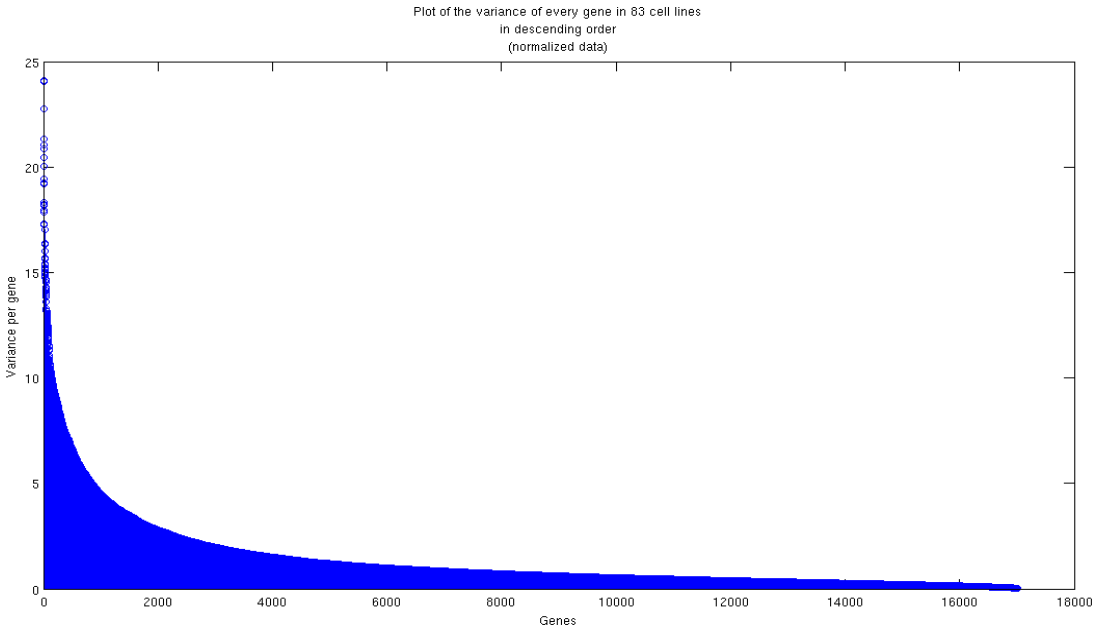
Οι τιμές της έντασης των γονιδίων μέσα στις κυτταρικές τιμές έχουν ένα μεγάλο φάσμα τιμών. Ενδεικτικό στοιχείο αυτού του γεγονότος είναι ότι η μέγιστη τιμή έκφρασης που καταγράφεται είναι 1.5664e+05 και η ελάχιστη είναι 4.7300, όπως μετρήθηκαν μετά την απόρριψη εκείνων των γονιδίων που δεν εκφράζονται. Στην μέτρηση της ομοιομορφίας μπορούμε να θεωρήσουμε ότι τιμές που βρίσκονται στην ίδια τάξη μεγέθους και με πολύ μικρή απόκλιση μεταξύ τους θεωρούνται ισοδύναμες (π.χ. οι τιμές 100 και 120 ή οι τιμές 1000 και 1100). Αυτός ήταν ένας επιπλέον λόγος που μας ώθησε στην λογαριθμική κανονικοποίηση των δεδομένων.

Στα παρακάτω σχεδιαγράμματα έχουμε και πάλι την τιμή της διασποράς του κάθε γονιδίου μέσα στις 83 κυτταρικές σειρές (Σχεδ. 9) και σε φθίνουσα σειρά (Σχεδ. 10), ο υπολογισμός της οποίας έγινε σε

κανονικοποιημένες τιμές έντασης. Παρατηρούμε ότι κανονικοποιώντας τις τιμές της έντασης, οι τιμές της διασποράς του κάθε γονιδίου είναι αρκετές τάξεις μεγέθους μικρότερες, κάνοντας τους πολύπλοκους και χρονοβόρους υπολογισμούς που θα ακολουθήσουν πιο γρήγορους, χωρίς όμως να χάνουμε πληροφορία.



Σχεδιάγραμμα 10: Διάγραμμα τιμών διασποράς κάθε γονιδίου στις 83 κυτταρικές σειρές (κανονικοποιημένα δεδομένα)



Σχεδιάγραμμα 11: Διάγραμμα τιμών διασποράς κάθε γονιδίου στις 83 κυτταρικές σειρές σε φθίνουσα σειρά(κανονικοποιημένα δεδομένα)

Όπως αναφέραμε και στην αρχή του κεφαλαίου, σκοπός μας είναι η κατασκευή ενός μείγματος αναφοράς το οποίο θα αποτελείται από ένα πλήθος κυτταρικών σειρών. Με βάση την βιβλιογραφία το μέγεθος του μείγματος είναι πάνω από 10 κυτταρικές σειρές. Μέσω των πειραμάτων που εκτελέσαμε, τα οποία αναλύονται και παρατίθενται παρακάτω, καταλήξαμε στο συμπέρασμα ότι το βέλτιστο πρότυπο μείγμα, το οποίο καλύπτει σε πολύ ικανοποιητικό βαθμό τόσο την παράμετρο της έκφρασης, όσο και την παράμετρο της ομοιομορφίας έχει μέγεθος 20 κυτταρικών σειρών. Επομένως, η μέτρηση και η αξιολόγηση της ομοιομορφίας γίνεται μέσα στα πλαίσια του μεγέθους των υποψηφίων λύσεων, δηλαδή των n-άδων, όπως θα εξηγήσουμε αναλυτικότερα παρακάτω.

3.5 Προσαρμογή των ΓΑ στις ιδιαιτερότητες του προβλήματος

Ο Γενετικός Αλγόριθμος ο οποίος κατασκευάσαμε στα πλαίσια αυτής της εργασίας δεν είναι ένας τυπικός ΓΑ, όπως αυτός παρουσιάστηκε στο Κεφάλαιο 2. Αν και η βασική ιδέα πίσω από την υλοποίηση, καθώς και οι τεχνικές παραγωγής γενεών, είναι εμπνευσμένα από την λειτουργία ενός τυπικού ΓΑ, υπάρχουν αρκετές διαφοροποιήσεις, οι οποίες ήταν επιβεβλημένες από την φύση του προς βελτιστοποίηση προβλήματος.

Όπως αναφέρθηκε και προηγουμένως, στην παράγραφο για την ομοιομορφία, σκοπός μας είναι να μειώσουμε όσο το δυνατόν περισσότερο την διασπορά της έντασης των γονιδίων μέσα στις κυτταρικές σειρές. Σε κάθε κυτταρική σειρά ένα γονίδιο εμφανίζεται μία και μόνο φορά. Επομένως αυτό που πρέπει να κάνουμε είναι να επιλέγουμε n-άδες κυτταρικών σειρών, να μετράμε την διασπορά όλων των γονιδίων μέσα σε αυτές τις n-άδες και αυτό είναι το μέγεθος που θέλουμε να ελαχιστοποιήσουμε. Είναι προφανές, επομένως, ότι η διερεύνηση μίας κυτταρικής σειράς από μόνη της δεν μπορεί να μας δώσει κάποιο αποτέλεσμα παρά μόνο ο συνδυασμός πολλών κυτταρικών σειρών, δηλαδή μίας n-άδας. Μιλώντας με τους όρους των ΓΑ, ένα γονίδιο του ΓΑ που κατασκευάσαμε αποτελείται από ένα πλήθος n-άδων και μία n-άδα αποτελεί ένα χρωμόσωμα του ΓΑ. Τελικά, οι λύσεις οι οποίες διερευνούμε είναι συνδυασμοί κυτταρικών σειρών, χωρίς διάταξη.

Οι τεχνικές της αναπαραγωγής που αναπτύξαμε στο προηγούμενο κεφάλαιο εφαρμόζονται στην παραγωγή νέων n-άδων από τις προηγούμενες γενιές του ΓΑ και τα καινούρια γονίδια του ΓΑ αποτελούνται από την συλλογή των νέων n-άδων. Παρακάτω θα αναπτύξουμε τους τρόπους με τους οποίους εφαρμόσαμε τις τεχνικές της αναπαραγωγής στα γονίδια του δικού μας ΓΑ, καθώς και το πώς προκύπτουν τα χρωμοσώματα του ΓΑ που έχουμε κατασκευάσει.

3.5.1 Αναπαράσταση υποψήφιων λύσεων ΓΑ

Το πρόβλημα το οποίο καλούμαστε να βελτιστοποιήσουμε, με την χρήση ΓΑ, είναι αυτό της επιλογής του καλύτερου υποσυνόλου από το αρχικό πλήθος κυτταρικών σειρών που μας έχει δοθεί, με βάση τα κριτήρια της ομοιομορφίας και της έκφρασης γονιδίων, όπως αυτά αναπτύχθηκαν στις προηγούμενες παραγράφους. Ο αρχικός μας πληθυσμός αποτελείται από 83 κυτταρικές σειρές, κάθε μία από τις οποίες “ονομάζεται” με έναν δεκαδικό αριθμό από το 1 έως το 83. Επομένως το σύνολο αναπαράστασης του προβλήματος είναι πεπερασμένο και η πιο απλή και αποδοτική αναπαράσταση είναι αυτή της άμεσης ακέραιας κωδικοποίησης. Κάθε γονίδιο του ΓΑ αποτελείται από μία n-άδα ακεραίων αριθμών από το 1 έως το 83, οι οποίοι αριθμοί αντιστοιχούν στην κυτταρική σειρά με αυτό το όνομα, χωρίς να επαναλαμβάνεται καμία τιμή μέσα στο διάστημα. Επειδή δύο πιθανές n-άδες με την ίδια τιμή αλλά διαφορετική διάταξη αναφέρονται στην ίδια πιθανή λύση, n-άδες είναι σε διατεταγμένη αύξουσα σειρά, ώστε με μία απλή όρο προς όρο σύγκριση να προκύπτει η ισότητα ή η διαφορά δύο n-άδων. Το ποσοστό διαφοράς δύο n-άδων μετράται ως το ποσοστό των διαφορετικών

κυτταρικών σειρών σε μία n-άδα προς την ίδια την n-άδα.

Στο σημείο αυτό και πριν προχωρήσουμε στην ανάλυση των μεθόδων δημιουργίας των γονιδίων των ΓΑ, αλλά και του τρόπου δημιουργίας του πληθυσμού της κάθε γενιάς των ΓΑ, είναι σκόπιμο να ορίσουμε την ορολογία που θα χρησιμοποιούμε. Με τον όρο n-άδα εννοούμε το γονίδιο του κάθε ΓΑ, δηλαδή τον αριθμό των κυτταρικών σειρών που αποτελούν την υποψήφια λύση. Κάθε γενιά ΓΑ αποτελείται από ένα πλήθος n-άδων, το σύνολο των οποίων και απαρτίζει τον πληθυσμό της κάθε γενιά των ΓΑ. Όπως θα αναλύσουμε και παρακάτω, οι ΓΑ που εξετάζουμε σε ότι αφορά το μέγεθος του πληθυσμού είναι δύο κατηγοριών, με στενό πληθυσμό που αποτελείται από 20 γονίδια (n-άδες ή αλλιώς υποψήφιες λύσεις) και με “φαρδύ” πληθυσμό που αποτελείται από 100 γονίδια (n-άδες ή αλλιώς υποψήφιες λύσεις). Σε ότι αφορά το μέγεθος των γονιδίων (n-άδων ή υποψηφίων λύσεων) έχουμε πειράματα με μέγεθος υποψηφίων λύσεων αποτελούμενων από 10 έως 30 κυτταρικών σειρών.

3.5.2 Παραγωγή γονιδίων (n-άδων κυτταρικών σειρών)

Όπως αναφέρεται στο 1ο κεφάλαιο αυτής της εργασίας, τα πρότυπα μείγματα αναφοράς δεν αποτελούνται από έναν συγκεκριμένο και σταθερό αριθμό κυτταρικών σειρών. Το πλήθος των κυτταρικών σειρών που συμμετέχουν διαφέρει από πείραμα σε πείραμα. Στις περισσότερες μελέτες που έχουν γίνει έχει χρησιμοποιηθεί πλήθος κυτταρικών σειρών ίσο και μεγαλύτερο των 10. Θεωρήσαμε σκόπιμο στην εργασία αυτή το μέγεθος της n-άδας κάθε ΓΑ να είναι ίσο με 20, δηλαδή ο συνδυασμός 20 κυτταρικών σειρών, και για κάθε ΓΑ το πείραμα να επαναληφθεί και για μέγεθος n-άδας μεγαλύτερο αλλά και μικρότερο του 20. Ο λόγος που επιλέξαμε την επανάληψη του κάθε πειράματος για μικρότερο μέγεθος n-άδας ήταν για να δούμε αν υπάρχει συσχέτιση της απόδοσης του αλγορίθμου με το μέγεθος της προτεινόμενης λύσης.

Μέσα σε μία συγκεκριμένη γενιά του ΓΑ, κάθε n-άδα κυτταρικών σειρών μπορεί να προκύψει με έναν από τους παρακάτω τρόπους:

- Carry-over : Αντιγραφή από την προηγούμενη γενιά στην τρέχουσα χωρίς καμία τροποποίηση. Με την μέθοδο αυτή εξασφαλίζουμε την “διαιώνιση” των καλύτερων n-άδων που έχουν συναντηθεί κατά την εκτέλεση του ΓΑ. Το πλήθος των n-άδων που προκύπτουν από το carry-over ως προς το πλήθος των n-άδων που απαρτίζουν όλη την γενιά είναι το “χάσμα γενεών”

(generation gap) που εμφανίζει ο εκάστοτε ΓΑ. Όσο μικρότερο είναι το χάσμα γενεών τόσο περισσότερο εξασφαλίζεται η προσήλωση του ΓΑ σε n-άδες προηγούμενων γενεών με καλά αποτελέσματα.

- Mutate : Παραγωγή n-άδας με περιορισμένη τροποποίηση προηγούμενης n-άδας. Δεδομένου ότι η κωδικοποίηση του ΓΑ αποκλίνει από τη συνήθη εκδοχή, ως μηχανισμός μετάλλαξης επιλέχθηκε ο παρακάτω τρόπος :

  Έστω $VS=\{1,2,\ldots,k\}$ το σύνολο των κυτταρικών σειρών, $G\subset VS, |G|=n$ μία n-άδα και $C=VS-G, |C|=k-n$. Η νέα n-άδα παράγεται αντιγράφοντας το σύνολο G και στη συνέχεια αντικαθιστώντας στοιχεία του συνόλου G με τον ίδιο αριθμό στοιχείων του συνόλου C. Η παραγωγή n-άδων μέσω του μηχανισμού της μετάλλαξης εντατικοποιεί τις τυχαίες αναζητήσεις κοντά σε καλές n-άδες. Επειδή ο ΓΑ επιλέγει να μεταλλάσσει από τις καλύτερες σε καταλληλότητα n-άδες της κάθε γενιάς, ώστε να τις εισαγάγει μεταλλαγμένες στην επόμενη, εξασφαλίζονται τόσο η προσήλωση σε n-άδες με καλά αποτελέσματα, όπως και με την μέθοδο του carry-over, ενώ ταυτόχρονα έχουμε και την εξερεύνηση όλου του χώρου αναζήτησης γύρω από τα καλά αποτελέσματα και σε συνδυασμό με αυτά. Ταυτόχρονα επιτυγχάνουμε τον μη-εγκλωβισμό σε τοπικά μέγιστα, μέσω της διαφοροποίησης των καλών n-άδων κατά ένα μικρό ποσοστό.

- Cross-over : Παραγωγή n-άδας μέσω “διασταύρωσης” δύο προηγούμενων n-άδων. Και πάλι, δεδομένου ότι η κωδικοποίηση του ΓΑ αποκλίνει από τη συνήθη εκδοχή, επιλέξαμε τον παρακάτω μηχανισμό διασταύρωση:

  Έστω $VS=\{1,2,\ldots,k\}$ το σύνολο των κυτταρικών σειρών και $G1\subset VS, G2\subset VS$ όπου $|G1|=|G2|=n$ οι δύο n-άδες που θέλουμε να διασταυρώσουμε. Η καινούρια n-άδα παράγεται ως εξής: Παίρνουμε την ένωση των συνόλων G1 και G2, $G1\cup G2$, το μέγεθος της οποίας βρίσκεται μέσα στο διάστημα $[n, 2*n]$. Στη συνέχεια “ανακατεύουμε” τα στοιχεία του συνόλου που έχει δημιουργηθεί από την ένωση των G1 και G2 (με χρήση της εντολής randperm του Matlab) και από το σύνολο αυτό επιλέγουμε τις n πρώτες κυτταρικές σειρές, οι οποίες και θα αποτελέσουν την καινούρια n-άδα.

- Τυχαία παραγωγή : Παραγωγή n-άδας με τυχαίο τρόπο. Από το σύνολο των κυτταρικών σειρών

που έχουμε διαλέγουμε τυχαία n στοιχεία. Η επιλογή γίνεται “ανακατεύοντας” τα στοιχεία του συνόλου $VS=\{1,2,\ldots,k\}$ (με χρήση της εντολής randperm του Matlab) και επιλέγουμε τα n πρώτα στοιχεία. Η τυχαία παραγωγή n-άδων εντατικοποιεί τις τυχαίες αναζητήσεις μέσα σε όλο το χώρο αναζήτησης, μειώνοντας με αυτόν τον τρόπο την πιθανότητα να “χαθεί” κάποιος καλός συνδυασμός εξαιτίας της προσήλωσης του γενετικού σε κάποιους καλούς συνδυασμούς που έχουν δοκιμαστεί και συνεχίζουν να δοκιμάζονται. Ταυτόχρονα, με την εισαγωγή τυχαίων n-άδων ελαχιστοποιούμε την πιθανότητα εγκλωβισμού σε τοπικά μέγιστα, αφού συνεχώς εισάγονται νέοι συνδυασμοί που δεν έχουν καμία φυσική σχέση με κάποιους άλλους που έχουν ήδη δοκιμαστεί και επιτύχει.

3.5.3 Παραγωγή γενεών

Η παραγωγή της πρώτης γενιάς κάθε ΓΑ γίνεται με τυχαίο τρόπο. Με τον μηχανισμό της τυχαίας παραγωγής που αναπτύξαμε παραπάνω παράγουμε m αριθμό n-άδων, που θα αποτελέσουν την πρώτη γενιά. Κάθε γενιά του ΓΑ έχει ένα συγκεκριμένο πληθυσμό υποψηφίων λύσεων. Η βασική φιλοσοφία των Γενετικών Αλγορίθμων, όπως αναφέραμε και στο Κεφ.2, είναι ότι ο πληθυσμός κάθε γενιάς αξιολογείται με βάση την συνάρτηση καταλληλότητας και εν συνεχεία τα μέλη του πληθυσμού που έχουν τις υψηλότερες τιμές καταλληλότητας αποτελούν την βάση για την επόμενη γενιά. Για τον λόγο αυτό, αξιολογούμε την κάθε n-άδα του αρχικού αυτού πληθυσμού με βάση την συνάρτηση καταλληλότητας και τις ταξινομούμε με φθίνουσα σειρά, έτσι ώστε οι n-άδες με την υψηλότερη τιμή καταλληλότητας να χρησιμοποιηθούν στην παραγωγή της επόμενης γενιάς, ενώ αυτές με τις χαμηλότερες τιμές να απορριφθούν. Επιπλέον για κάθε γενιά του γενετικού κρατάμε ιστορικό με την καλύτερη n-άδα της γενιάς, καθώς και το σκορ το οποίο “έπιασε”.

Ο πληθυσμός κάθε γενιάς είναι ένας αριθμός από n-άδες, τα μέλη του οποίου αξιολογούνται βάση της συνάρτησης καταλληλότητας, και με τις τεχνικές που έχουμε αναπτύξει και παρατίθενται παρακάτω, από κάθε γενιά παράγεται η επόμενη. Ένα ερώτημα το οποίο ανακύπτει είναι από πόσα μέλη θα αποτελείται η κάθε γενιά. Οι υλοποιήσεις που έχουμε κάνει έχουν δύο διαφορετικούς πληθυσμούς. Η πρώτη περιλαμβάνει 20 n-άδες ανά γενιά και η δεύτερη 100 n-άδες ανά γενιά. Το μέγεθος της γενιάς, ουσιαστικά, είναι εις βάθος αναζήτηση στο χώρο και όσο μεγαλύτερος είναι ο πληθυσμός τόσο πιο εξαντλητική είναι και η έρευνα. Αρχικά χρησιμοποιήσαμε πληθυσμό 20 “γονιδίων”, αλλά

παρατηρώντας τα αποτελέσματα (Κεφάλαιο 4), καταλήξαμε στο συμπέρασμα ότι ο πληθυσμός αυτός είναι ανεπαρκής και μέσω πειραμάτων είδαμε ότι ένας πληθυσμός 100 “γονιδίων” είναι αυτός που ταιριάζει καλύτερα στο πρόβλημά μας.

3.5.3.1 Πληθυσμός γενιάς (population size) 20

Για την παραγωγή των επόμενων γενεών χρησιμοποιούμε σαν βάση την πρώτη γενιά που έχουμε ήδη παραγάγει και εφαρμόζουμε τους μηχανισμούς παραγωγής γονιδίων που αναλύσαμε παραπάνω, για να δημιουργήσουμε την νέα γενιά.

- Το 10% της νέας γενιάς το παίρνουμε με την τεχνική του carry-over. Στην περίπτωση που ο πληθυσμός των γονιδίων, δηλαδή το χρωμόσωμα, του ΓΑ είναι 20 n-άδες, επιλέγουμε τις 2 καλύτερες n-άδες, δηλαδή αυτές που είχαν την υψηλότερη τιμή καταλληλότητας, για να συμμετάσχουν στην νέα γενιά. Ο λόγος που επιλέγονται οι δύο καλύτερες της προηγούμενης γενιάς να εισαχθούν στην επόμενη είναι ότι έτσι εξασφαλίζουμε πως κάθε καινούρια γενιά θα είναι καλύτερη από την επόμενη και ταυτόχρονα, όσο καλύτερη απόδοση έχει μία n-άδα κυτταρικών σειρών, τόσες περισσότερες φορές δοκιμάζεται.

- Το 25% της νέας γενιάς το διαλέγουμε με τυχαίο τρόπο. Στην περίπτωση που ο πληθυσμός των γονιδίων, δηλαδή το χρωμόσωμα, του ΓΑ είναι 20 n-άδες, επιλέγουμε 5 n-άδες για να συμμετάσχουν στον νέο πληθυσμό οι οποίες παράγονται με τον τυχαίο τρόπο που αναπτύξαμε παραπάνω.

- Το 15% της νέας γενιάς το παράγουμε με την τεχνική της μετάλλαξης. Στην περίπτωση που ο πληθυσμός των γονιδίων, δηλαδή το χρωμόσωμα, του ΓΑ είναι 20 n-άδες, επιλέγουμε 3 n-άδες που έχουν προκύψει μεταλλάσσοντας τις 3 n-άδες της προηγούμενης γενιάς που είχαν την υψηλότερη τιμή καταλληλότητας. Η μετάλλαξη γίνεται με τον μηχανισμό που αναλύσαμε παραπάνω, αντικαθιστώντας δηλαδή σε κάθε μία από τις τρεις n-άδες δύο κυτταρικές σειρές με τυχαίο τρόπο.

- Το 50% της νέας γενιάς το παράγουμε με την τεχνική της cross-over διασταύρωσης. Στην περίπτωση που ο πληθυσμός των γονιδίων, δηλαδή το χρωμόσωμα, του ΓΑ είναι 20 n-άδες, επιλέγουμε 10 n-άδες, οι οποίες παράγονται με την τεχνική του cross-over. Οι υποψήφιοι “γονείς” είναι οι 5 n-άδες της προηγούμενης γενιάς με την υψηλότερη τιμή καταλληλότητας.

Για την παραγωγή κάθε “απογόνου” επιλέγουμε με τυχαίο τρόπο δύο γονείς από τους 5 πιθανούς υποψήφιους και μετά ακολουθούμε την παραπάνω διαδικασία. Ο τρόπος επιλογής των γονέων γίνεται με πιθανότητας επιλογής ανάλογη με την καταλληλότητά τους.

3.5.3.2 Πληθυσμός γενιάς (population size) 100

Ένας δεύτερος ΓΑ που υλοποιήσαμε, χρησιμοποιεί τις ίδιες ρουτίνες και μεθόδους με αυτόν που περιγράψαμε παραπάνω, με την διαφορά ότι σε κάθε γενιά ο πληθυσμός των υποψηφίων λύσεων σε κάθε γενιά από 20 είναι 100. Πιο συγκεκριμένα, η πρώτη γενιά αποτελείται από 100 n-άδες που παράγονται με τυχαίο τρόπο. Στη συνέχεια, για κάθε επόμενη γενιά

- Το 2% της επόμενης γενιάς αποτελείται από τις καλύτερες n-άδες της προηγούμενης (carry-over), δηλαδή 2 από τις 100 n-άδες του πληθυσμού
- Το 8% της νέας γενιάς προέρχεται από n-άδες που έχουν επιλεγεί με την τεχνική της μετάλλαξης, δηλαδή 8 από τις 100 n-άδες του πληθυσμού
- Το 10% της νέας γενιάς επιλέγεται με τυχαίο τρόπο, δηλαδή 10 n-άδες του νέου πληθυσμού
- Το 80% της νέας γενιάς αποτελείται από n-άδες που έχουν επιλεγεί με την τεχνική της cross-over διασταύρωσης, δηλαδή 80 από τις 100 n-άδες του νέου πληθυσμού

Για τη διασταύρωση που εκτελείται σε αυτήν την υλοποίηση έχουμε αυξήσει κατά 10 φορές τους υποψήφιους γονιούς, δηλαδή οι υποψήφιοι γονείς είναι 50 ανά γενιά. Η επιλογή τους γίνεται διαλέγοντας τις 50 καλύτερες από άποψη τιμής καταλληλότητας n-άδες της προηγούμενης γενιάς. Η στρατηγική επιλογής γονέων που ακολουθούμε, σε αντίθεση με την προηγούμενη υλοποίηση, είναι ουσιαστικά ο αποκλεισμός από την δεξαμενή ζευγαρώματος (mating pool) των χειρότερων n-άδων, δηλαδή αποκλεισμός με πιθανότητα αντιστρόφως ανάλογη της καταλληλότητάς τους. Με τον τρόπο αυτό εξαντλούμε ακόμα περισσότερο τον χώρο αναζήτησης και ελαττώνουμε ακόμα περισσότερο την πιθανότητα εγκλωβισμού του αλγορίθμου σε τοπικά μέγιστα.

Μόλις η διαδικασία παραγωγής μίας γενιάς τελειώσει, τότε τα στοιχεία της γενιάς αξιολογούνται και ξεκινάει ο κύκλος παραγωγής της επόμενης. Η παραγωγή γενεών και κατ' επέκταση ο ΓΑ σταματάει όταν επιτευχθεί το κριτήριο τερματισμού του γενετικού αλγορίθμου που έχουμε θέσει και αναλύουμε

παρακάτω.

3.5.4 Συναρτήσεις Καταλληλότητας

Κάθε ΓΑ που κατασκευάσαμε ακολουθεί τις ίδιες μεθοδολογίες παραγωγής γενεών, κωδικοποίησης και τερματισμού. Αυτό που διαφοροποιεί τους ΓΑ μεταξύ τους είναι οι συναρτήσεις καταλληλότητας για την αξιολόγηση των υποψηφίων λύσεων, ο πληθυσμός των υποψηφίων λύσεων που παράγονται ανά γενιά, καθώς και ο πληθυσμός των υποψήφιων γονέων. Στις παραγράφους που προηγήθηκαν αναφερθήκαμε στις μεθοδολογίες παραγωγής γενεών, κωδικοποίησης και τερματισμού, καθώς και στους πληθυσμούς υποψηφίων λύσεων και γονέων. Στο σημείο αυτό θα αναλύσουμε τις συναρτήσεις καταλληλότητας, που αποτελούν και τον τρόπο αξιολόγησης των ίδιων των ΓΑ μεταξύ τους.

3.5.4.1 Συνάρτηση ελαχιστοποίησης διασποράς με κατώφλι

Η πρώτη προσέγγιση που κάναμε στο ζητούμενο της ελαχιστοποίησης της διασποράς της έντασης των γονιδίων ήταν με την χρήση κατωφλίου. Πιο συγκεκριμένα, μετράμε τη διασπορά κάθε γονιδίου κάθε υποψήφιας λύσης (n-άδας) και ζητάμε η διασπορά αυτή να είναι κάτω από ένα κατώφλι, κάτω δηλαδή από μία συγκεκριμένη τιμή. Στη συνέχεια κάθε n-άδα αξιολογείται, δηλαδή παίρνει μία τιμή από το 0 μέχρι το 1, όπου 1 σημαίνει απόλυτη ταύτιση με την τιμή του κατωφλίου, και οι n-άδες με τις υψηλότερες τιμές αποτελούν και την γενιά του γενετικού. Για να μπορέσουμε να προσδιορίσουμε το κατώφλι που ελαχιστοποιεί όσο το δυνατόν περισσότερο την διασπορά εκτελέσαμε πολλαπλά πειράματα με διαφορετικές τιμές κατωφλίωσης. Οι τιμές που δοκιμάσαμε ανήκαν στο διάστημα [0.0024, 24] με βήμα 0.0006, προσπαθώντας να καλύψουμε όσο το δυνατόν περισσότερες τιμές, τόσο σε μικρές όσο και σε αρκετά μεγάλες τιμές διασποράς. Όπως βλέπουμε και από το Σχεδ.9, η μέγιστη τιμή διασποράς που εμφανίζουν τα γονίδια μέσα στις 83 κυτταρικές σειρές είναι 25 και για το λόγο αυτό η μέγιστη τιμή κατωφλίου που δοκιμάζουμε είναι 24.

Όπως έχουμε αναπτύξει και παραπάνω, ο ΓΑ που έχουμε κατασκευάσει προσπαθεί να εντοπίσει τοπικά μέγιστα. Επομένως έπρεπε η συνάρτηση καταλληλότητας να δομηθεί με τέτοιο τρόπο ώστε αν και ο αλγόριθμος δουλεύει εντοπίζοντας τα μέγιστα, να μας δίνει αποτελέσματα που ελαττώνουν την

διασπορά. Ο τρόπος που το επιτυγχάνουμε αυτό είναι θέτοντας την μεγιστοποίηση της συνάρτησης ως εξής: $variance_{(n-\acute{\alpha}\delta\alpha\varsigma)} < variance_{threshold}$. Επειδή η συνάρτηση αυτή μεγιστοποιείται, μας ενδιαφέρουν κυρίων οι μικρές τιμές κατωφλίου.

3.5.4.2 Συνάρτηση ελαχιστοποίησης διασποράς χωρίς κατώφλι

Η δεύτερη προσέγγιση που κάναμε στο ζητούμενο της ελαχιστοποίησης της διασποράς της έντασης των γονιδίων ήταν χωρίς χρήση κατωφλίου. Αντίστοιχα με παραπάνω, μετράμε την διασπορά κάθε γονιδίου κάθε υποψήφιας λύσης (n-άδας). Αντί όμως να ζητάμε λύσεις μικρότερες από ένα κατώφλι, στην περίπτωση αυτή ψάχνουμε την n-άδας που μεγιστοποιεί την συνάρτηση $1/(1+variance_{(n-\acute{\alpha}\delta\alpha\varsigma)})$. Η συνάρτηση αυτή παίρνει τιμές στο διάστημα [0,1] και η μεγιστοποίησή της συνεπάγεται την την ελαχιστοποίηση της διασποράς της n-άδας. Το όφελος που έχουμε με την χρήση αυτής της συνάρτησης καταλληλότητας είναι ότι η βελτιστοποίηση του προβλήματος γίνεται με “αυτόματο” τρόπο, δηλαδή χωρίς να πρέπει να θέτουμε εξωτερικά κατώφλια, τα οποία και θα πρέπει συνέχεια να αξιολογούμε ως προς την απόδοσή τους, αλλά πάντα, όπως θα δείξουμε και παρακάτω, βρίσκει την βέλτιστη λύση.

3.5.4.3 'Υβριδική' συνάρτηση καταλληλότητας

Μέχρι το σημείο αυτό, οι συναρτήσεις καταλληλότητας που παρουσιάσαμε ασχολούνταν μόνο με το θέμα της μεγιστοποίησης της ομοιομορφίας. Ο σκοπός μας, όμως, είναι διττός. Ταυτόχρονα με την μεγιστοποίηση της ομοιομορφίας θέλουμε να μεγιστοποιήσουμε και την έκφραση των γονιδίων μέσα στο μείγμα των κυτταρικών σειρών που κατασκευάζουμε, ώστε τελικά να πετύχουμε την μέγιστη κάλυψη. Ως κάλυψη θεωρούμε το ποσοστό των γονιδίων το οποίο έχει τιμή έντασης πάνω από ένα κατώφλι. Το γεγονός αυτό σημαίνει ότι πρέπει να κατασκευαστεί μία συνάρτηση καταλληλότητας που θα συνδυάζει τα παραπάνω.

Αφού εξαντλήσαμε το θέμα της ομοιομορφίας, στο σημείο αυτό θα ασχοληθούμε με το θέμα της μεγιστοποίησης της έκφρασης των γονιδίων. Όπως έχουμε αναφέρει και σε προηγούμενες παραγράφους, ο πίνακας δεδομένων που δουλεύουμε έχει μειωθεί κατά 23%, αφού έχουν αφαιρεθεί όλα εκείνα τα γονίδια τα οποία δεν εκφράζονται, δηλαδή έχουν τιμή έντασης μικρότερη από 100, σε

καμία από τις 83 κυτταρικές σειρές. Αυτό βέβαια δεν συνεπάγεται ότι δεν υπάρχουν γονίδια τα οποία σε κάποιες κυτταρικές σειρές έχουν τιμή έντασης μικρότερη του κατωφλίου και σε κάποιες άλλες μεγαλύτερη του κατωφλίου. Επομένως, πρέπει μέσω της συνάρτησης καταλληλότητας να μπορέσουμε να συμπεριλάβουμε κυτταρικές σειρές στο μείγμα, στις οποίες τουλάχιστον το μεγαλύτερο ποσοστό των γονιδίων να έχει τιμή πάνω από το κατώφλι τουλάχιστον μία φορά μέσα στο δείγμα. Για να το επιτύχουμε αυτό πρέπει ουσιαστικά να "αναζητήσουμε" την μέγιστη μέση τιμή.

Σε αντίθεση με την ελαχιστοποίηση της διασποράς, η συνάρτηση καταλληλότητας της οποίας είναι φραγμένη στο [0,1], η μεγιστοποίηση της μέσης τιμής δεν μπορεί να έχει ένα φραγμένο και προκαθορισμένο φάσμα τιμών. Για να μπορέσουμε, επομένως, να συνδυάσουμε την μείωση της διασποράς και την μεγιστοποίηση της μέσης τιμής, χωρίς να υποτιμήσουμε καμία παράμετρο έναντι της άλλης, η αξιολόγηση των υποψηφίων λύσεων γίνεται πολλαπλασιάζοντας τα αποτελέσματα των δύο συναρτήσεων.

$$f_{hibrid} = f_{(variance)} . * f_{(expression)} = [1/(1+variance_{(n-\acute{\alpha}\delta\alpha\varsigma)})] . * [mean(expression_{(n-\acute{\alpha}\delta\alpha\varsigma)})]$$

Και οι τρεις συναρτήσεις καταλληλότητας που χρησιμοποιήσαμε στα πειράματά μας έχουν κριτήρια τα οποία εφαρμόζονται ανά σειρά του πίνακα δεδομένων, δηλαδή ανά γονίδιο, και κανένα από αυτά ανά στήλη. Ο λόγος που δουλέψαμε κατά αυτόν τον τρόπο ήταν ότι ανά στήλη, δηλαδή ανά κυτταρική σειρά, δεν ήταν δυνατόν να πάρουμε τις πληροφορίες που θέλαμε τόσο για την ομοιομορφία όσο και για την έκφραση των γονιδίων, μιας και στόχος μας ήταν η κατασκευή μειγμάτων σε σχέση με ιδιότητες των γονιδίων. Αν τα κριτήριά μας ήταν σε σχέση με ιδιότητες των κυτταρικών σειρών, τότε οι αντίστοιχες συναρτήσεις καταλληλότητας θα εφαρμόζονταν ανά στήλη.

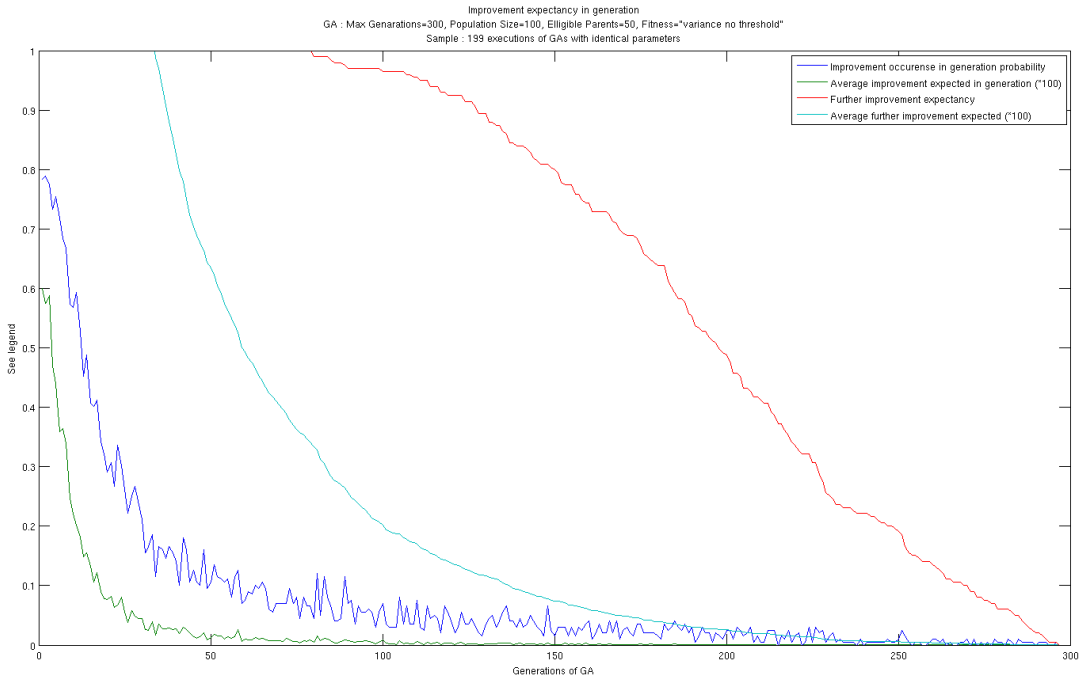
3.5.5 Τερματισμός ΓΑ

Με δεδομένη την πολυπλοκότητα του αλγορίθμου και το υπολογιστικό κόστος της κάθε γενιάς, η ανάγκη για βελτιστοποίηση των αποτελεσμάτων έπρεπε να γίνεται μέσα σε λογικό υπολογιστικό χρόνο. Για το λόγο αυτό το κριτήριο τερματισμού του αλγορίθμου αποφασίσαμε να είναι ο αριθμός των επαναλήψεων και όχι κάποιο από τα άλλα κριτήρια που αυξάνουν το υπολογιστικό κόστος. Για να είμαστε σίγουροι ότι δεν είμαστε στην περίπτωση εγκλωβισμού του αλγορίθμου γύρω από κάποιο τοπικό μέγιστο αποφασίσαμε κάθε γενετικός να τερματίζει αφού έχει επιτευχθεί σύγκλιση των

αποτελεσμάτων συν κάποιες γενιές ακόμα, θεωρώντας ότι αυξάνοντας των αριθμό των γενεών, ακόμα και αν είμαστε στην περίπτωση του εγκλωβισμού σε τοπικό μέγιστο, θα το ξεπερνά, ακόμα και μετά από πολλές γενεές μηδενικής προόδου.

Ο τρόπος αυτός για τον τερματισμό του ΓΑ έχει το πλεονέκτημα ότι δεν αυξάνει το υπολογιστικό κόστος του αλγορίθμου, το οποίο είναι ούτως ή άλλως αρκετά υψηλό, αλλά ταυτόχρονα έχει το μειονέκτημα ότι αλλάζοντας κάποιες από τις μεταβλητές του αλγορίθμου, όπως για παράδειγμα την συνάρτηση καταλληλότητας ή το βάθος του, δηλαδή τον πληθυσμό της κάθε γενιάς, ο αλγόριθμος παρουσιάζει σύγκλιση σε διαφορετικό αριθμό γενεών. Επομένως, μαζί με όλες τις άλλες παραμέτρους που αλλάζουμε πρέπει ενδεχομένως να αλλάζει και ο αριθμός των επαναλήψεων.

Στις γραφικές παραστάσεις που ακολουθούν γίνεται φανερό το παραπάνω μειονέκτημα του κριτηρίου τερματισμού που χρησιμοποιούμε:



Σχεδιάγραμμα 12: Πρόοδος του ΓΑ για 300 γενεές με population size 100

Στο παραπάνω διάγραμμα βλέπουμε την πρόοδο ενός ΓΑ, που έχουμε τρέξει 199 φορές. Ο ΓΑ αυτός χρησιμοποιεί σαν συνάρτηση καταλληλότητας την “ελαχιστοποίηση διασποράς χωρίς κατώφλι” και το μέγεθος του πληθυσμού ανά γενιά είναι 100 20-άδες, εκ των οποίων οι 80 προέρχονται από cross-over διασταύρωση με πληθυσμό πιθανών γονέων 50 20-άδες, 8 με την τεχνική της μετάλλαξης, 2 με την

τεχνική του carry-over και 10 έχουν εισαχθεί με τυχαίο τρόπο.

Με μπλε γραμμή παρουσιάζεται η πιθανότητα βελτίωσης του γονιδίου του ΓΑ από γενιά σε γενιά. Για κάθε γενιά του ΓΑ υπολογίζουμε την ποσοστιαία διαφορά ανάμεσα στο καλύτερο σκορ της γενιάς και της αμέσως προηγούμενής της για κάθε ένα από τα 199 πειράματα και στη συνέχεια υπολογίζουμε την μέση τιμή της διαφοράς αυτής, η οποία αποτελεί και την πιθανότητα βελτίωσης από την μία γενιά στην άλλη.

Με πράσινη γραμμή έχουμε το πραγματικό μέσο ποσοστό της βελτίωσης του καλύτερου γονιδίου από την μία γενιά στην αμέσως επόμενη. Για κάθε γενιά του ΓΑ υπολογίζουμε την διαφορά ανάμεσα στο καλύτερο σκορ της γενιάς και της αμέσως προηγούμενής της για κάθε ένα από τα 199 πειράματα και στη συνέχεια παίρνουμε την μέση τιμή του ποσοστού αυτού.

Η κόκκινη γραμμή παρουσιάζει τη μέση αναμενόμενη πιθανότητα (πρόβλεψη) να υπάρξει βελτίωση από την μία γενιά στην αμέσως επόμενη. Ο υπολογισμός της τιμής αυτής γίνεται παίρνοντας το μέσο συσσωρευτικό άθροισμα της πιθανότητας βελτίωσης.

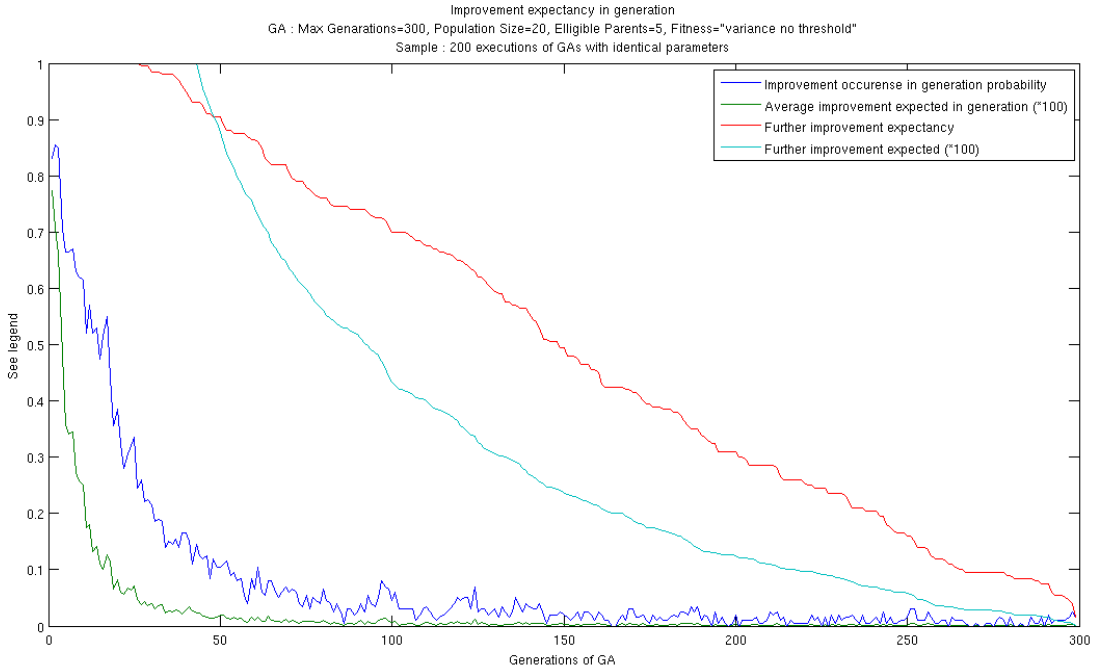
Με την γαλάζια γραμμή έχουμε το μέσο αναμενόμενο ποσοστό βελτίωσης (πρόβλεψη) του καλύτερου γονιδίου από την μία γενιά στην αμέσως επόμενη. Ο υπολογισμός της τιμής αυτής γίνεται παίρνοντας το μέσο συσσωρευτικό άθροισμα του ποσοστού βελτίωσης.

Μπορούμε να δούμε από το παραπάνω γράφημα ότι από την γενιά 240 και μετά η πιθανότητα βελτίωσης του καλύτερου γονιδίου από την μία γενιά στην αμέσως επόμενη τείνει στο 0 (μπλε γραμμή), καθώς επίσης και και το ποσοστό βελτίωσης (πράσινη γραμμή). Το αναμενόμενο ποσοστό βελτίωσης (γαλάζια γραμμή) τείνει και αυτό στο 0, ενώ η αναμενόμενη πιθανότητα βελτίωσης παρουσιάζει αρκετά μικρή τιμή. Μπορούμε επομένως να θεωρήσουμε με βεβαιότητα ότι ο αλγόριθμος μετά την γενιά 250 έχει συγκλίνει και το κριτήριο τερματισμού των 300 γενιών είναι ικανοποιητικό.

Το διάγραμμα που ακολουθεί είναι αντίστοιχο του Σχεδ.12. Η συνάρτηση καταλληλότητας του ΓΑ είναι και πάλι η “ελαχιστοποίηση διασποράς χωρίς κατώφλι”, αλλά αυτή την φορά το μάγεθος του πληθυσμού ανά γενιά είναι 20 20-άδες, εκ των οποίων οι 10 προέρχονται από cross-over διασταύρωση με πληθυσμό πιθανών γονέων 5 20-άδες, 3 με την τεχνική της μετάλλαξης, 2 με την τεχνική του carry-over και 5 έχουν εισαχθεί με τυχαίο τρόπο.

Στο σχεδιάγραμμα αυτό παρατηρούμε ότι μετά την γενιά 150, το ποσοστό βελτίωσης (πράσινη

γραμμή) του ΓΑ τείνει στο μηδέν, αλλά η πιθανότητα βελτίωσης (μπλε γραμμή), αν και έχει πολύ μικρές τιμές, ταλαντώνεται συνεχώς. Οι προβλέψεις, τόσο της πιθανότητας βελτίωσης (κόκκινη γραμμή) όσο και του ποσοστού βελτίωσης (γαλάζια γραμμή) φθίνουν με σχεδόν γραμμικό τρόπο, σε αντίθεση με το Σχεδ.12, όπου η μείωση ήταν εκθετική, και πλησιάζουν την τιμή 0 προς το τέλος των 300 γενεών του κριτηρίου τερματισμού.



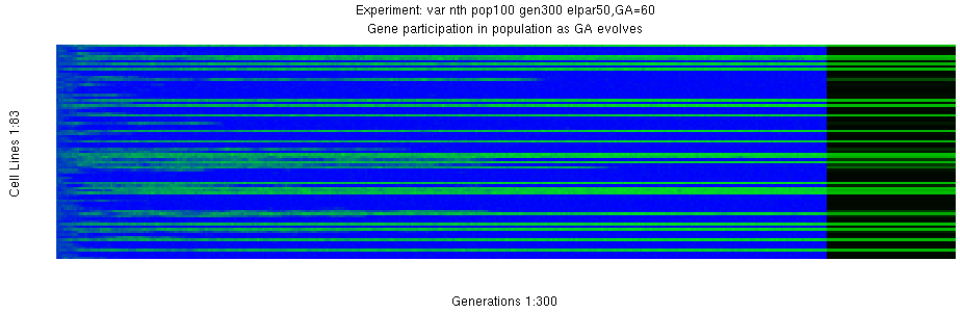
Σχεδιάγραμμα 13: Πρόοδος του ΓΑ για 300 γενιές με population size 20

Συγκρίνοντας τα Σχεδιαγράμματα 12 και 13 μπορούμε να πούμε ότι για τους ΓΑ με μεγάλο πληθυσμό το κριτήριο τερματισμού των 300 γενεών είναι επαρκές, ενώ για τους ΓΑ με μικρό πληθυσμό θα πρέπει τα κριτήριο τερματισμού να διευρυνθεί.

Στη συνέχεια εξετάζουμε τον τρόπο με τον οποίο οι ΓΑ εξελίσσονται μέσα στο κριτήριο τερματισμού και αν το κριτήριο που έχουμε θέσει είναι επαρκές για την σύγκλισή τους. Ενδεικτικά παρουσιάζουμε την εξέλιξη ενός ΓΑ με πληθυσμό γενιάς 100 20-άδες και στη συνέχεια έναν ΓΑ με πληθυσμό ανά γενιά 20 20-άδες.

Στο παρακάτω σχήμα έχουμε έναν από τους 199 ΓΑ με πληθυσμό γενιάς 100 20-άδες. Στον οριζόντιο άξονα του σχήματος αυτού είναι οι 300 γενιές του ΓΑ και στον κάθετο έχουμε τις 83 κυτταρικές σειρές. Κάθε κυτταρική σειρά η οποία συμμετέχει στην καλύτερη λύση κάθε γενιάς “παίρνει” πράσινο χρώμα.

Όταν η σειρά αυτή εμφανιστεί και στην καλύτερη λύση της επόμενης γενιάς “ανεβαίνει” κατά έναν τόνο η απόχρωση του πράσινου. Όσο πιο πράσινη είναι μία κυτταρική σειρά τόσες περισσότερες φορές και συμμετέχει στην καλύτερη λύση, δηλαδή αυξάνετε η βεβαιότητα του ΓΑ ότι η κυτταρική σειρά αυτή είναι στις καλύτερες. Παρατηρούμε εύκολα ότι κατά τις πρώτες γενιές έχουμε συμμετοχή πολλών διαφορετικών κυτταρικών σειρών (εμφάνιση με πολύ αχνό πράσινο), αλλά όσο εξελίσσεται ο ΓΑ, δηλαδή όσο προχωράμε προς το κριτήριο τερματισμού, αρχίζουν κυτταρικές σειρές που συμμετείχαν στις αρχικές γενιές να “εξαφανίζονται” , ενώ άλλες να ισχυροποιούνται. Στις 200 περίπου γενιές παρατηρούμε ότι έχει πλέον αποφασίσει ο ΓΑ τις καλύτερες κυτταρικές σειρές και από το σημείο αυτό και μετά έχουμε συμμετοχή των ίδιων κυτταρικών σειρών. Καταλήγουμε επομένως στο συμπέρασμα ότι οι 300 γενιές είναι ένα επαρκές κριτήριο τερματισμού.

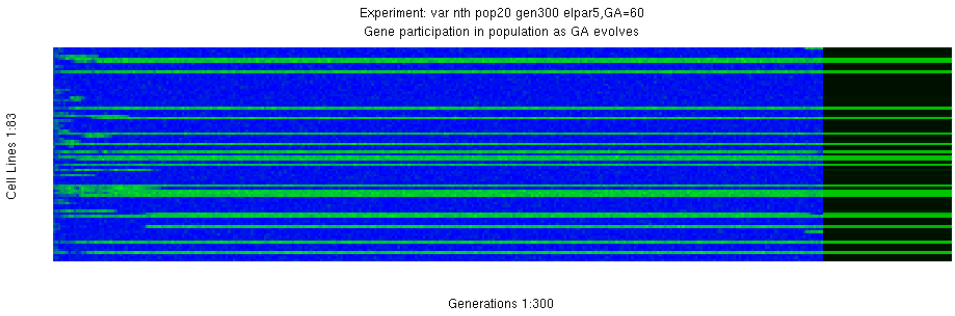


Σχεδιάγραμμα 14: Συμμετοχή κυτ. σειρών στην εξέλιξη του ΓΑ (population 100, gene size 20)

Στην δεξιά στήλη του σχεδιαγράμματος έχουμε την ολική συμμετοχή κυτταρικών σειρών στον ΓΑ.

Στο παρακάτω σχήμα έχουμε έναν από τους 200 ΓΑ με πληθυσμό γενιάς 20 20-άδες. Στον οριζόντιο

άξονα του σχήματος αυτού είναι όπως και πριν οι 300 γενιές του ΓΑ και στον κάθετο έχουμε τις 83 κυτταρικές σειρές. Σε αντιστοιχία με το Σχεδ.14, όσο πιο πράσινη είναι μία κυτταρική σειρά τόσες περισσότερες φορές και συμμετέχει στην καλύτερη λύση, δηλαδή αυξάνετε η βεβαιότητα του ΓΑ ότι η κυτταρική σειρά αυτή είναι στις καλύτερες. Και πάλι παρατηρούμε ότι κατά τις πρώτες γενιές έχουμε συμμετοχή πολλών διαφορετικών κυτταρικών σειρών (εμφάνιση με πολύ αχνό πράσινο), αλλά όσο εξελίσσεται ο ΓΑ, δηλαδή όσο προχωράμε προς το κριτήριο τερματισμού, αρχίζουν κυτταρικές σειρές που συμμετείχαν στις αρχικές γενιές να “εξαφανίζονται” , ενώ άλλες να ισχυροποιούνται. Ενώ φαίνεται αρχικά ότι όπως και στο προαναφερθέν διάγραμμα ο ΓΑ έχει αποφασίσει αρκετά νωρίς, τελικά λίγο πριν το τέλος των 300 γενεών κυτταρικές σειρές που συμμετείχαν κατά ένα πολύ μεγάλο ποσοστό γενεών και φαίνονται να έχουν αποφασιστεί, τελικά “χάνονται” και αντικαθίστανται από άλλες. Συμπερασματικά για αυτόν τον ΓΑ, το κριτήριο των 300 γενεών δεν είναι επαρκές.

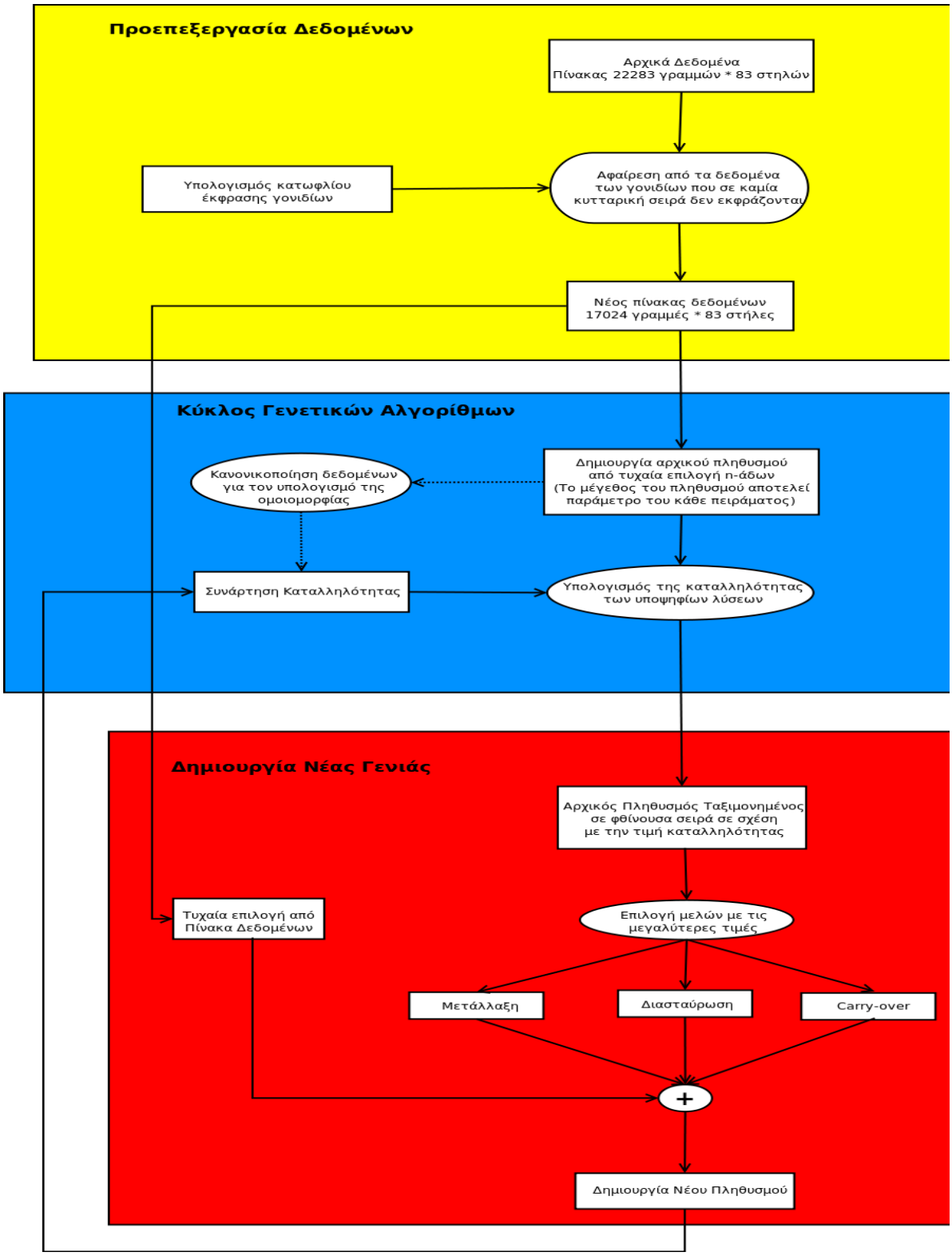


Σχεδιάγραμμα 15: Συμμετοχή κυτ. σειρών στην εξέλιξη του ΓΑ (population 20, gene size 20)
Στην δεξιά στήλη του σχεδιαγράμματος έχουμε την ολική συμμετοχή κυτταρικών σειρών στον ΓΑ.

Από τα παραπάνω προκύπτει ένα επιπλέον συμπέρασμα. Όσο μεγαλύτερο είναι το πλήθος των n-άδων των κυτταρικών σειρών που έχουμε ανά γενιά, τόσο πιο γρήγορα συγκλίνει ο ΓΑ. Δηλαδή, η έρευνα σε βάθος απαιτεί μικρότερη έρευνα σε πλάτος, θεωρώντας πλάτος τον αριθμό των γενεών και βάθος το

πλήθος των πιθανών λύσεων.

Κλείνοντας αυτήν την παράγραφο, μπορούμε να πούμε ότι το κριτήριο τερματισμού που εφαρμόζουμε είναι ένας συνδυασμός της σύγκλισης του αλγορίθμου μαζί με το κριτήριο του αριθμού επαναλήψεων, όπως αυτό περιγράφηκε στο Κεφάλαιο 2. Μπορεί η μέθοδός μας να έχει το ελάττωμα που αναφέρθηκε παραπάνω, αλλά συγκρίνοντας τον υπολογιστικό χρόνο που θα χρειαζόμασταν αν εφαρμόζαμε κάποια άλλη από τις προτεινόμενες μεθόδους με το κόστος των 4 ή 5 παραπάνω ΓΑ προκειμένου να βρούμε πότε συγκλίνει ο αλγόριθμος, θεωρούμε ότι η επιλογή αυτή ήταν πιο αποδοτική. Στο παρακάτω διάγραμμα έχουμε σχεδιάσει τα δομικά στοιχεία του προτεινόμενου αλγορίθμου που περιγράψαμε στο κεφάλαιο αυτό, καθώς και των τρόπο διασύνδεσής τους.



Σχεδιάγραμμα 16: Διάγραμμα ροής προτεινόμενου αλγορίθμου

Κεφάλαιο 4 – Πειράματα και Αποτελέσματα

4.1 Πειράματα για την εύρεση της ομοιομορφίας

4.1.1 Σύγκριση μεθοδολογιών και ΓΑ

Στο σημείο αυτό είναι χρήσιμο να αναφέρουμε μία πειραματική μέτρηση την οποία κάναμε, για να αξιολογήσουμε τους δύο Γενετικούς Αλγορίθμους που χρησιμοποιήσαμε για την μέτρηση της ομοιομορφίας, καθώς και να τους συγκρίνουμε με άλλες μεθόδους. Για να μπορέσει να γίνει αυτή η σύγκριση αρχικά έπρεπε να οριστεί μία μετρική, με την οποία θα αξιολογούνταν τα πειράματα, και η οποία θα έπρεπε να είναι άνω και κάτω φραγμένη. Για τον λόγο αυτό η μετρική που χρησιμοποιήθηκε ήταν η εξής:

$$metric = 1/(1+variance),\ metric \in [0,1]$$

Επειδή οι ΓΑ που έχουμε κατασκευάσει προσπαθούν να μεγιστοποιήσουν την την συνάρτηση καταλληλότητας, χρησιμοποιώντας την παραπάνω συνάρτηση, η μεγιστοποίησή της μας δίνει την ελαχιστοποίηση της διασποράς, δηλαδή την μεγιστοποίηση της ομοιομορφίας. Στην συνέχεια αυτής της παραγράφου θα αντιπαραβάλλουμε διαφορετικές μεθόδους και ΓΑ, που έχουν σαν σκοπό την αύξηση της ομοιομορφίας των γονιδίων, προκειμένου να καταλήξουμε στην αποδοτικότερη.

1. Τυχαία επιλογή κυτταρικών σειρών: Στο πείραμα αυτό διαλέξαμε με τυχαίο τρόπο 199 φορές εικοσάδες κυτταρικών σειρών και μετρήσαμε, με βάση την παραπάνω μετρική, την απόδοση της τυχαίας επιλογής.

2. “Αφελής” επιλογή προτύπου: Σε αυτό το πείραμα μετρήσαμε για κάθε γονίδιο στις κυτταρικές σειρές την μέση τιμή του. Στη συνέχεια, επιλέξαμε εκείνη την εικοσάδα κυτταρικών σειρών, για την οποία η μέση τιμή της έντασης των γονιδίων έχει την μικρότερη απόκλιση από το “αφελές” μας πρότυπο, δηλαδή την μέση τιμή της έντασης του κάθε γονιδίου.

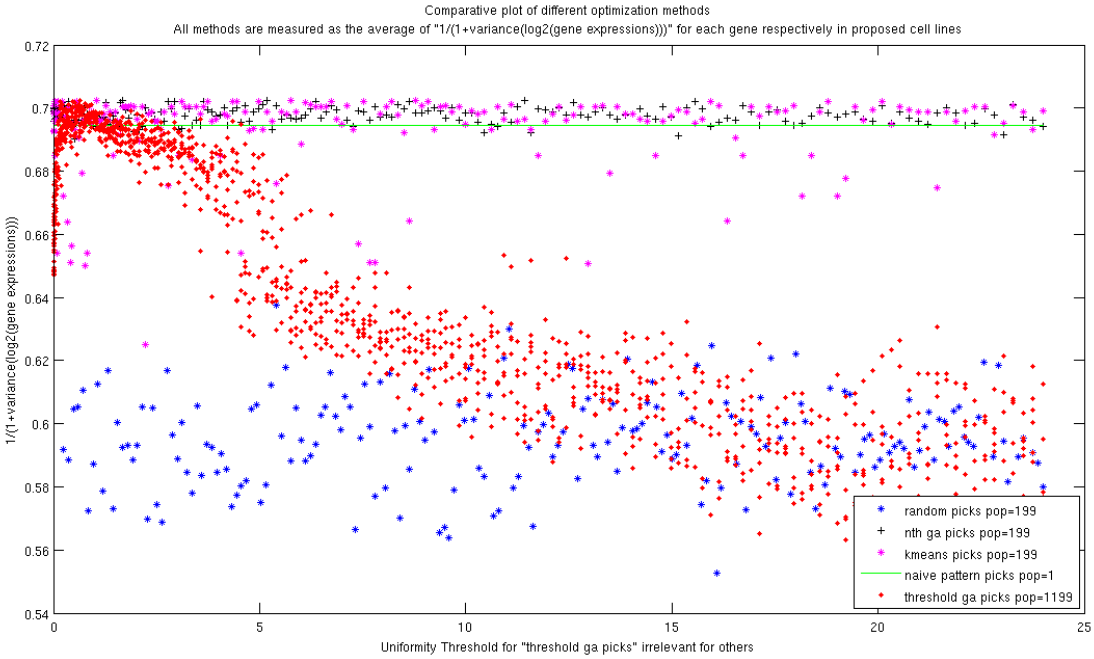
3. K-means clustering: Εφαρμόσαμε ένα απλό clustering με 10 clusters, που ελαχιστοποιεί το τετράγωνο της Ευκλείδειας απόστασης. Εφαρμόσαμε 199 φορές το clustering.

4. ΓΑ με κατωφλίωση (cutoff): Για το πείραμα αυτό χρησιμοποιήσαμε τον ΓΑ που κατασκευάσαμε με κατώφλι για την τιμή της διασποράς και κάναμε 199 πειράματα για

ισάριθμα διαφορετικά κατώφλια διασποράς.

5. ΓΑ χωρίς κατωφλίωση: Για το πείραμα αυτό χρησιμοποιήσαμε τον ΓΑ που κατασκευάσαμε χωρίς κατώφλι για την τιμή της διασποράς και κάναμε 1199 πειράματα.

Η γραφική παράσταση που ακολουθεί δείχνει συγκεντρωτικά τα αποτελέσματα των μεθόδων και πειραμάτων που περιγράφηκαν παραπάνω:



Σχεδιάγραμμα 17:Σύγκριση μεθόδων για προσδιορισμό ομοιομορφίας

Στον άξονα y του διαγράμματος βρίσκονται οι τιμές της μετρικής, ενώ στον άξονα x οι διαφορετικές τιμές κατωφλίων που δοκιμάσαμε στον ΓΑ με κατώφλι. Ο άξονας x για τα υπόλοιπα πειράματα είναι άνευ σημασίας.

Ο σκοπός του παραπάνω γραφήματος είναι να αξιολογήσουμε την αποδοτικότητα των αλγορίθμων μας, σε σχέση με την βελτιστοποίηση της ομοιομορφίας. Οι αλγόριθμοι που συγκρίνουμε είναι τα δύο είδη γενετικών που υλοποιήσαμε (με κατώφλίωση και χωρίς) καθώς και κάποιοι άλλοι αλγόριθμοι

επιλογής κυτταρικών σειρών, τους οποίους υλοποιήσαμε και αναφέρουμε πιο αναλυτικά παρακάτω. Σε κάθε αλγόριθμο το μέγεθος της υποψήφιας λύσης είναι το ίδιο, δηλαδή 20 κυτταρικές σειρές, και τα πειράματα επαναλήφθηκαν αρκετές φορές για καλύτερη τεκμηρίωση. Το συμπέρασμα στο οποίο καταλήξαμε ήταν ότι ο Γ.Α. χωρίς κατώφλι αποδείχθηκε πιο αποδοτικός, όπως αναλυτικά αναφέρουμε στην συνέχεια της παραγράφου.

Κάθε μία από τις μπλε κουκκίδες αντιπροσωπεύει και μία λύση που προτείνει ο μηχανισμός της τυχαίας επιλογής. Η επιλογή αυτή υλοποιείται με έναν πολύ απλό μηχανισμό. Ανακατεύουμε κάθε φορά τις 83 κυτταρικές σειρές που έχουμε στο data set με τυχαίο τρόπο (χρήση randperm()) και στη συνέχεια διαλέγουμε 20 από τις 83 σειρές. Αφού γίνει η επιλογή αξιολογούμε την κάθε 20-άδα με βάση την μετρική που αναφέρουμε παραπάνω. Όπως είναι εμφανές και από το γράφημα, το καλύτερο σκορ που πιάνει μία 20-άδα που επιλέγεται με αυτή την μέθοδο είναι 0.6330, ενώ το χειρότερο είναι 0.5482.

Η πράσινη γραμμή αντιπροσωπεύει την επιλογή 20-άδας με τον μηχανισμό της “αφελούς” επιλογής. Ο μηχανισμός αυτός συνίσταται στην δημιουργία ενός προτύπου. Το πρότυπό μας αποτελείται από ένα διάνυσμα-στήλη 17012 γραμμών, κάθε μία από τις οποίες έχει την μέση τιμή της έντασης κάθε γονιδίου στις 83 κυτταρικές σειρές. Αφού δημιουργήσαμε το πρότυπο αυτό, στη συνέχεια επιλέξαμε 20 κυτταρικές σειρές, για τις οποίες η μέση τιμή της έντασης του κάθε γονιδίου μέσα σε αυτήν την εικοσάδα έχει την μικρότερη δυνατή απόκλιση από το πρότυπο. Στη συνέχεια αξιολογήσαμε την προτεινόμενη λύση από την σκοπιά της ομοιομορφίας χρησιμοποιώντας την μετρική που αναφέρουμε παραπάνω. Το σκορ της 20-άδας που προτείνεται από την “αφελή” επιλογή είναι 0.6944, δηλαδή κατά πολύ μεγαλύτερο από αυτό της τυχαίας επιλογής. Η 20-άδα που επιλέγεται είναι αυτή που έχει την μικρότερη απόκλιση από το πρότυπο αυτό και επιλέγεται με ντετερμινιστικό τρόπο. Συγκρίνοντας την μέθοδο αυτή με την μέθοδο της τυχαίας επιλογής, μπορούμε να πούμε ότι η μέθοδος του “αφελούς” προτύπου είναι πιο αποδοτική, πράγμα το οποίο οφείλεται σε μεγάλο βαθμό στον ντετερμινισμό της μεθόδου.

Οι μοβ κουκκίδες του γραφήματος αντιπροσωπεύουν τις επιλεγόμενες από τον k-means cluster αλγόριθμο 20-άδες. Το καλύτερο σκορ των 20-άδων που επιλέγονται από τον cluster είναι 0.7024, ενώ το χειρότερο είναι 0.65. Τρέξαμε τον cluster 199 φορές και όπως φαίνεται από το γράφημα η πλειοψηφία των λύσεων βρίσκεται πάνω από την τιμή απόδοσης που “πιάνει” το “αφελές” πρότυπο,

αλλά υπάρχουν και αρκετές λύσεις οι οποίες βρίσκονται σε τιμές απόδοσης αρκετά μικρότερες, αλλά πάνω από την καλύτερη λύση της τυχαίας επιλογής. Πιο συγκεκριμένα 147 λύσεις βρίσκονται πάνω από την τιμή 0.6944, δηλαδή ποσοστό 75%. Τελικά, ο k-means είναι πιο αποδοτικός από την τυχαία επιλογή, αφού η χειρότερη πρότασή του είναι καλύτερη από την καλύτερη πρόταση του τυχαίου, αλλά και από το "αφελές" πρότυπο, αφού τα τρία τέταρτα των προτάσεών του βρίσκονται πάνω από την τιμή του προτύπου.

Οι κόκκινες κουκκίδες του γραφήματος παρουσιάζουν τα σκορ που πιάνουν οι προτεινόμενες 20-άδες του ΓΑ με συνάρτηση καταλληλότητας που θέτει κατώφλι στην τιμή της διασποράς της έντασης των γονιδίων στις κυτταρικές σειρές. Κάθε κόκκινη κουκκίδα αντιστοιχεί και σε διαφορετική προτεινόμενη 20-άδα, με διαφορετική τιμή κατωφλίου. Στη συνάρτηση καταλληλότητας που χρησιμοποιήσαμε, υπολογίζουμε πάνω στα λογαριθμημένα δεδομένα την variance των γονιδίων μέσα στις υποψήφιες λύσεις και επιλέγουμε αυτές τις λύσεις όπου η variance είναι μικρότερη από το κατώφλι. Εκτελέσαμε 199 πειράματα με διαφορετικά κατώφλια, που δίνονται από τον τύπο $uniformity_{threshold} = k^2 * 24$, *όπου* $k \in [0.01, 1]$ *με βήμα επανάληψης* $(0.005^2) * 24$ και τα οποία επαναλάβαμε 6 φορές.

Σε κάθε γενιά του ΓΑ, είχαμε πληθυσμό 20-άδων ανά γενιά 20. Η πρώτη γενιά παράγεται με τυχαίο τρόπο και για τις επόμενες έχουμε δύο 20-άδες να παράγονται με την τεχντική του carry-over, 3 με την τεχνική της μετάλλαξης, 5 να διαλέγονται με τυχαίο τρόπο και 10 με την τεχνική του cross-over. Ο πληθυσμός των υποψήφιων γονέων για την παραγωγή του cross-over ήταν 5. Ως κριτήριο τερματισμού των γενετικών έχουμε θέσει τις 300 γενιές. Όπως φαίνεται και στο γράφημα, για πολύ μικρές τιμές κατωφλίου (άξονας x), τα αποτελέσματα του αλγορίθμου ακολουθούν αύξουσα τιμή, με αφετηρία τιμή μετρικής 0.6471 και αποκορύφωμα την τιμή 0.7024, αλλά στη συνέχεια και όσο αυξάνονται οι τιμές των κατωφλίων τα σκορ των υποψηφίων λύσεων αρχίζουν να φθίνουν εκθετικά, και για πολύ μεγάλες τιμές κατωφλίων η συμπεριφορά του ΓΑ προσομοιάζει την συμπεριφορά της τυχαίας επιλογής. Βλέπουμε ότι ο ΓΑ με κατώφλι για κάποιες μικρές τιμές κατωφλίων είναι καλύτερος από το "αφελές" πρότυπο και δίνει αποτελέσματα συγκρίσιμα με αυτά του cluster, αλλά για μεγάλες τιμές κατωφλίου λειτουργεί σχεδόν με τυχαίο τρόπο. Επομένως, για να έχουμε καλά αποτελέσματα πρέπει η επιλογή κατωφλίου να γίνει με απόλυτο τρόπο, αφού κάθε απόκλιση από το βέλτιστο κατώφλι μας απομακρύνει και από τη βέλτιστη λύση.

Τέλος, με μαύρες κουκκίδες παρουσιάζονται στο γράφημα οι προτεινόμενες λύσεις των ΓΑ χωρίς κατωφλίωση. Η συνάρτηση καταλληλότητας των ΓΑ είναι η ίδια η μετρική, δηλαδή $metric = 1/(1+variance), metric \in [0,1]$. Η παραγωγή γενεών, καθώς και το κριτήριο τερματισμού γίνονται με τον ίδιο τρόπο όπως και στους ΓΑ με κατωφλίωση. Εκτελέσαμε 199 πειράματα και το χειρότερο σκορ προτεινόμενης λύσης είναι 0.6878, ενώ το καλύτερο είναι 0.7024. Παρατηρούμε ότι η ζώνη απόδοσης του ΓΑ χωρίς κατωφλίωση είναι μικρότερη από την αντίστοιχη του cluster και η πλειοψηφία των λύσεων βρίσκεται πάνω από το “αφελές” πρότυπο. Συγκρίνοντας τον ΓΑ αυτόν με τον ΓΑ χωρίς κατωφλίωση παρατηρούμε ότι και οι δύο πιάνουν το υψηλότερο σκορ, δηλαδή 0.7024 όπως και ο k-means, αλλά η έλλειψη κατωφλίου κάνει τον αλγόριθμο πιο ευέλικτο στη χρήση (μπορούμε για παράδειγμα να αλλάξουμε data set χωρίς να χρειαστεί να αλλάξουμε το παραμικρό στον αλγόριθμο) και λιγότερο επιρρεπή σε λάθη, όπως ο προσδιορισμός κακού κατωφλίου.

Το Σχεδ.15 αποτελεί επί της ουσίας έναν συγκεντρωτικό χάρτη αποτελεσμάτων των διαφόρων μεθόδων που δοκιμάσαμε, προκειμένου να μετρήσουμε την αποδοτικότητα των ΓΑ που κατασκευάσαμε, σε ότι αφορά την ελαχιστοποίηση της διασποράς. Σε όλα τα πειράματα που εκτελέσαμε το μέγεθος της υποψήφιας λύσης είναι 20 κυτταρικές σειρές. Η μέτρηση της διασποράς γίνεται με τον ίδιο μηχανισμό σε όλα τα πειράματα και σε όλες τις μεθοδολογίες, γιατί με αυτόν τον τρόπο μπορούμε να συγκρίνουμε την αποδοτικότητα των διαφορετικών μεθοδολογιών. Είναι προφανές από το παραπάνω διάγραμμα ότι οι ΓΑ, τόσο με κατώφλι όσο και χωρίς, δεν λειτουργούν με τυχαίο τρόπο, όπως θα μπορούσε κάποιος να ισχυριστεί, αφού τα αποτελέσματά τους (κόκκινο και μαύρο) είναι εντελώς διαφορετικά από την μέθοδο της τυχαιότητας (μπλε). Το αποτέλεσμα που παίρνουμε από το “αφελές” πρότυπο είναι σαφώς ανώτερο της τυχαίας επιλογής, αφού αποφασίζεται με ντετερμινιστικό τρόπο και λαμβάνοντας ως παράμετρο την μέση τιμή της έντασης των γονιδίων μέσα στις κυτταρικές σειρές, θέτοντας δηλαδή κάποιον περιορισμό σε ότι αφορά την επιλογή, κάτι που στην τυχαία επιλογή δεν συμβαίνει. Συγκεντρώνοντας όλες τις παρατηρήσεις που κάναμε παραπάνω, μπορούμε να ταξινομήσουμε τις μεθοδολογίες ως προς την απόδοσή τους στην ελαχιστοποίηση της διασποράς με κριτήριο την μετρική $metric = 1/(1+variance), metric \in [0,1]$, με αύξουσα σειρά καταλληλότητας σε σχέση με την ελαχιστοποίηση της ομοιομορφίας ως εξής:

1. Τυχαία Επιλογή
2. ΓΑ με μεγάλες τιμές κατωφλίου

3. Αφελές Πρότυπο
4. ΓΑ με μικρές τιμές κατωφλίου
5. k-means cluster
6. ΓΑ χωρίς κατώφλι

Ο λόγος για τον οποίο ο ΓΑ χωρίς κατώφλι θεωρούμε ότι είναι πιο αποδοτικός από τον cluster, είναι ότι παρουσιάζει μεγαλύτερο βαθμό συγκέντρωσης των αποτελεσμάτων του γύρω από την καλύτερη τιμή σε σχέση με τον cluster, δηλαδή παρουσιάζει μικρότερη διασπορά τιμών. Οι ΓΑ με κατώφλι παρουσιάζουν διαφορετική συμπεριφορά, ανάλογα με την τιμή κατωφλίου που θέτουμε. Μπορούμε να πούμε ότι για μικρές τιμές κατωφλίων η μετρική αυξάνεται λογαριθμικά, μέχρι το σημείο που επιτυγχάνει την μέγιστη τιμή της (0.7024) και μετά φθίνει εκθετικά, όσο οι τιμές των κατωφλίων αυξάνονται. Επομένως, ο ΓΑ με κατωφλίωση επιτυγχάνει την βέλτιστη λύση μόνο για μία σαφώς ορισμένη τιμή κατωφλίου και θεωρείται για τον λόγο αυτό λιγότερο αποδοτικός.

Table 4: Συγκεντρωτικές τιμές μετρικής ομοιομορφίας για διαφορετικές μεθόδους

Μέθοδος	Χειρότερη τιμή μετρικής	Καλύτερη τιμή μετρικής
Τυχαία επιλογή	0.5482	0.6330
“Αφελές” πρότυπο	-	0.6944
K-means cluster	0.6500	0.7024
ΓΑ με κατωφλίωση	0.5472	0.7024
ΓΑ χωρίς κατωφλίωση	0.6878	0.7024

Παρατηρούμε από τον παραπάνω πίνακα ότι τρεις από τις πέντε μεθόδους που χρησιμοποιήσαμε έχουν σαν υψηλότερη τιμή μετρικής την 0.7024. Το γεγονός αυτό μας ωθεί στο συμπέρασμα ότι αυτή είναι και η υψηλότερη τιμή γενικά της μετρικής και κατά συνέπεια η διασπορά στην ένταση των γονιδίων ελαχιστοποιείται, με ελάχιστη τιμή 0.4237. Αν και έχουμε τρεις μεθόδους που μας δίνουν το ίδιο αποτέλεσμα, ο ΓΑ χωρίς κατωφλίωση θεωρούμε ότι είναι ο πιο αποδοτικός, για τους λόγους τους οποίους αναπτύξαμε παραπάνω.

4.1.2 Πειράματα ΓΑ για την ομοιομορφία με χρήση κατωφλίων (αναλυτικά)

Στην παράγραφο αυτή θα παρουσιάσουμε αναλυτικότερα τα πειράματα που έγιναν με χρήση συνάρτησης καταλληλότητας για μείωση της διασποράς των τιμών έντασης των γονιδίων στις κυτταρικές σειρές με κατώφλι. Η συνάρτηση καταλληλότητας που χρησιμοποιήσαμε είναι η "Συνάρτηση ελαχιστοποίησης διασποράς χωρίς κατώφλι". Ο πληθυσμός των υποψηφίων λύσεων είναι 20 ανά γενιά και το μέγεθος των υποψηφίων λύσεων είναι 20. Εκτός από την πρώτη γενιά που δημιουργείται με τυχαίο τρόπο, οι επόμενες γενιές προκύπτουν διαλέγοντας τις 2 καλύτερες 20-άδες της προηγούμενης γενιάς για να "προχωρήσουν" στην επόμενη με την τεχνική του carry-over, 5 νέες 20-άδες εισάγονται με τυχαίο τρόπο, 3 20-άδες στην νέα γενιά είναι αποτέλεσμα μετάλλαξης και τέλος 10 20-άδες στην νέα γενιά είναι αποτέλεσμα cross-over διασταύρωσης. Το κριτήριο τερματισμού του κάθε ΓΑ είναι στις 300 γενιές.

Όπως αναφέρουμε και σε προηγούμενη παράγραφο, τα κατώφλια που δοκιμάσαμε ήταν από 0.0024 έως 24 και κάθε κατώφλι δοκιμάστηκε 6 φορές. Στον παρακάτω πίνακα, για κάθε ένα από τα έξι πειράματα που έγιναν, βλέπουμε την ελάχιστη τιμή διασποράς που επιτεύχθηκε, καθώς επίσης και το αντίστοιχο κατώφλι. Παρατηρούμε ότι αν και οι τιμές της διασποράς είναι παραπλήσιες δεν υπάρχει ταύτιση μεταξύ των κατωφλίων. Η ελάχιστη τιμή διασποράς είναι 0.4237 στο κατώφλι 0.6534, η οποία όμως επιτυγχάνεται μόνο μία φορά στα συνολικά έξι πειράματα. Επομένως το συμπέρασμα στο οποίο καταλήξαμε στην προηγούμενη παράγραφο, ότι δηλαδή η μέθοδος με την κατωφλίωση δεν είναι αποδοτική, επαληθεύεται.

Αριθμός Πειράματος	Ελάχιστη τιμή διασποράς	Κατώφλι διασποράς
1	0.4260	0.7776
2	0.4267	0.9126
3	0.4241	0.1734
4	0.4246	0.4704
5	0.4237	0.6534
6	0.4272	0.8664

Table 5: Συγκεντρωτικός πίνακας ελάχιστων τιμών διασποράς και κατωφλίων που επιτεύχθηκε

(Συνολικά τα αποτελέσματα των πειραμάτων βρίσκονται στο Παράρτημα)

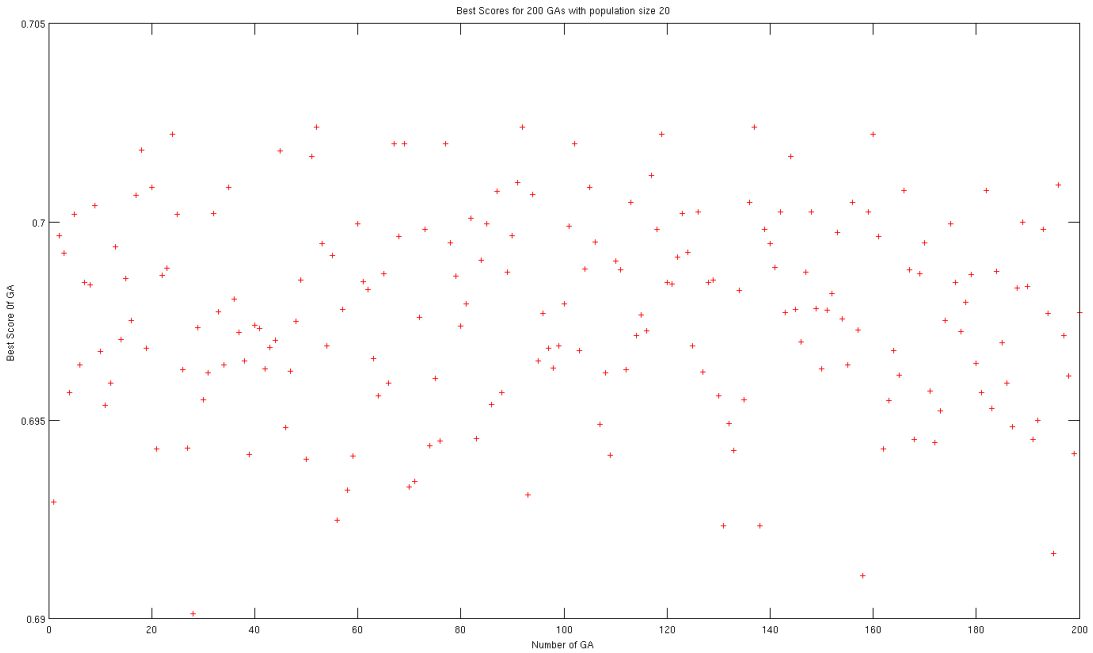
4.1.3 Σύγκριση ΓΑ με μικρό και μεγάλο πληθυσμό

Σε προηγούμενη παράγραφο δοκιμάσαμε διαφορετικές μεθόδους και ΓΑ για να προσδιορίσουμε ποια μέθοδος δίνει την βέλτιστη λύση στο πρόβλημα της ομοιομορφίας. Όπως φαίνεται η μέθοδος αυτή είναι ο ΓΑ χωρίς κατώφλι. Στα πειράματα που ακολουθούν δοκιμάσαμε ΓΑ χωρίς κατώφλι με στενό αλλά και φαρδύ πληθυσμό.

Ο πρώτος ΓΑ έχει πληθυσμό 20-άδων (population size) 20, οι οποίες, εκτός από την πρώτη γενιά όπου ο πληθυσμός δημιουργείται με τυχαίο τρόπο, δημιουργούνται ως εξής:

- Το 10%, 2 20-άδες, της νέας γενιάς το παίρνουμε με την τεχνική του carry-over.
- Το 25%, 5 20-άδες, της νέας γενιάς το διαλέγουμε με τυχαίο τρόπο.
- Το 15%, 3 20-άδες, της νέας γενιάς το παράγουμε με την τεχνική της μετάλλαξης.
- Το 50%, 10 20-άδες της νέας γενιάς το παράγουμε με την τεχνική της cross-over διασταύρωσης, όπου οι υποψήφιοι “γονείς” είναι οι 5 20-άδες της προηγούμενης γενιάς.

Εκτελέσαμε 200 ΓΑ, με 300 γενιές ανά γενετικό και συνάρτηση καταλληλότητας την “Συνάρτηση ελαχιστοποίησης της διασποράς χωρίς κατώφλι”. Στο γράφημα που ακολουθεί παρουσιάζονται για κάθε έναν από τους ΓΑ το καλύτερο σκορ που κατάφερε στην συνάρτηση αυτή.

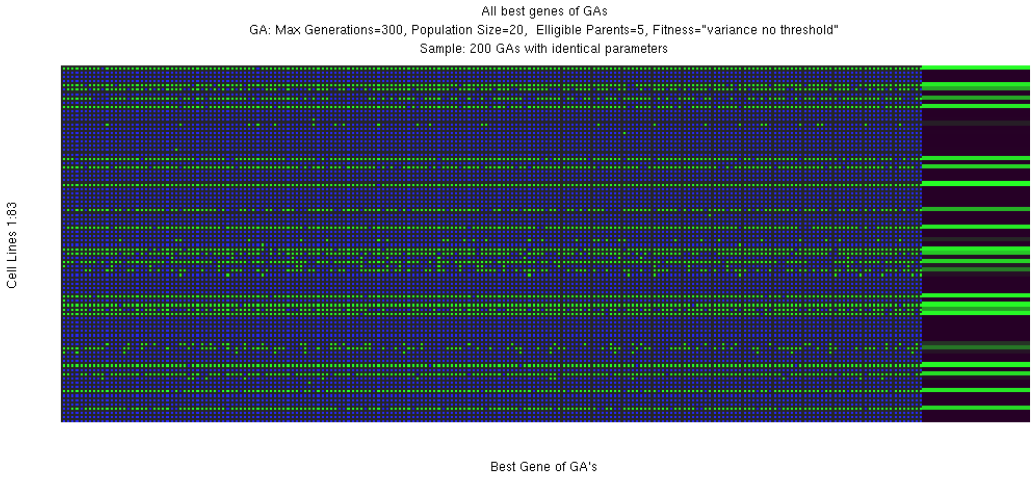


Σχεδιάγραμμα 18: Διάγραμμα των καλύτερων σκορ 200 ΓΑ με πληθυσμό 20

Στον x-άξονα έχουμε κάθε έναν από τους 200 Γ.Α. που εκτελέσαμε και στον y-άξονα έχουμε την επιτυχία του γενετικού. Εύκολα μπορούμε να παρατηρήσουμε ότι για κάθε γενετικό έχουμε και διαφορετική τιμή επιτυχίας και επομένως διαφορετική λύση. Ταυτόχρονα, βλέπουμε ότι υπάρχει αρκετά μεγάλη απόκλιση στην “καλύτερη” και στην “χειρότερη” τιμή επιτυχίας.

Στο παρακάτω σχήμα (Σχεδ. 17) αποτυπώνονται για κάθε έναν από τους παραπάνω ΓΑ η καλύτερη 20-άδα κυτταρικών σειρών, δηλαδή η 20-άδα που σημείωσε το καλύτερο σκορ, την καλύτερη τιμή καταλληλότητας. Στον y-άξονα του παρακάτω σχήματος έχουμε τις 83 κυτταρικές σειρές (με αύξουσα αρίθμηση από πάνω προς τα κάτω) και στον x-άξονα τα 200 πειράματα που εκτελέσαμε. Στο σχήμα αποτυπώνουμε επίσης το πόσο ισχυρά συμμετέχει μία κυτταρική σειρά στο σύνολο των πειραμάτων, με την στήλη που έχουμε στα δεξιά του γραφήματος. Κάθε πράσινο pixel δηλώνει τη συμμετοχή της κυτταρικής σειράς στον ΓΑ. Οι πράσινες γραμμές στα δεξιά του σχήματος δηλώνουν πόσο ισχυρά συμμετέχει μία κυτταρική σειρά στα πειράματα. Όσο πιο πράσινη είναι η γραμμή, τόσο πιο πολύ συμμετέχει και η κυτταρική σειρά, ενώ οι μαύρες γραμμές δηλώνουν ότι οι κυτταρικές σειρές στις οποίες αντιστοιχούν δεν συμμετέχουν ως υποψήφιες λύσεις στα πειράματα. Το σημείο της απότομης μεταβολής του γραφήματος είναι το σημείο όπου οι 200 Γ.Α. έχουν ολοκληρωθεί και από το σημείο

αυτό και στα δεξιά έχουμε την αποτύπωση της ισχυρής ή μη συμμετοχής των κυτταρικών σειρών στα πειράματα.



Σχεδιάγραμμα 19: Οι καλύτερες 20-άδες κυτταρικών σειρών για κάθε ΓΑ με πληθυσμό 20

Όπως βλέπουμε από το παραπάνω σχεδιάγραμμα, υπάρχουν αρκετές κυτταρικές σειρές οι οποίες συμμετέχουν σε σχεδόν όλα τα πειράματα, αλλά ταυτόχρονα υπάρχουν και κάποιες κυτταρικές σειρές που η συμμετοχή τους είναι σποραδική. Η σποραδικότητα αυτή που εμφανίζουν οι κυτταρικές σειρές στη συμμετοχή και η μη ισχυρή συνέπεια του Γ.Α. γύρω από συγκεκριμένες σειρές έχει σαν αποτέλεσμα την μεγάλη διακύμανση στην επιτυχία του γενετικού που παρατηρήθηκε στο Σχεδ. 16.

Στη συνέχεια εκτελέσαμε τα ίδια πειράματα, κρατώντας όλες τις παραμέτρους ίδιες αλλάζοντας μόνο το μέγεθος του πληθυσμού, έτσι ώστε να δούμε πώς το μέγεθος του πληθυσμού ενός Γ.Α. επηρεάζει την συμπεριφορά του και να μπορέσουμε να καταλήξουμε σε Γ.Α. που να “ταιριάζουν” καλύτερα στο πρόβλημα που καλούμαστε να βελτιστοποιήσουμε.

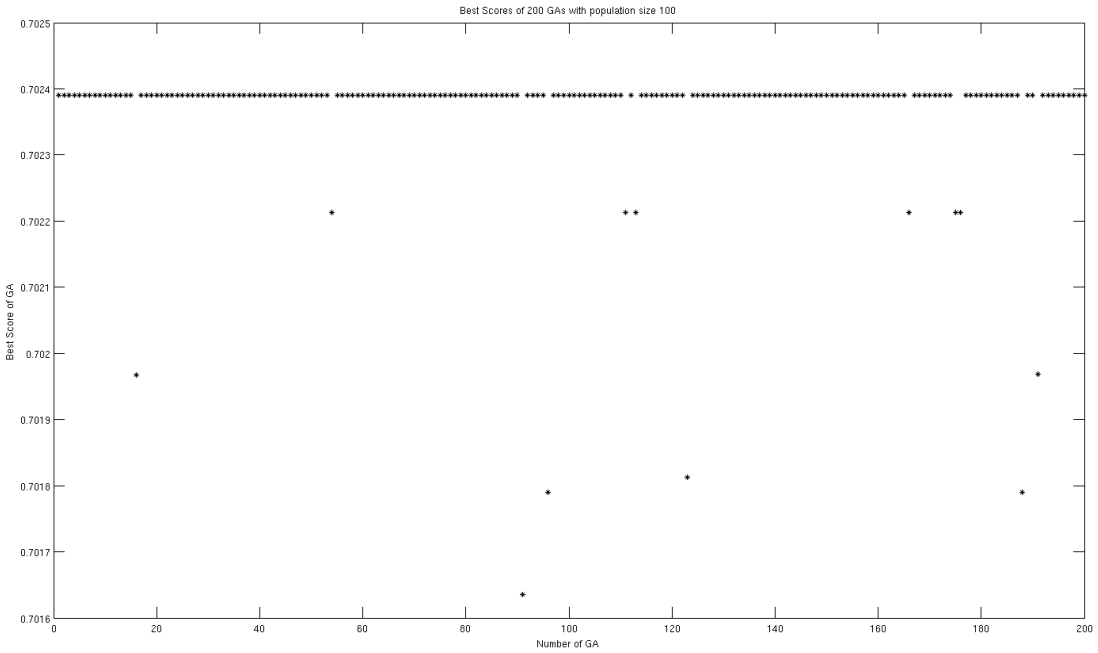
Ο δεύτερος ΓΑ έχει πληθυσμό 20-άδων (population size) 100, οι οποίες, εκτός από την πρώτη γενιά όπου ο πληθυσμός δημιουργείται με τυχαίο τρόπο, δημιουργούνται ως εξής:

- Το 2% της επόμενης γενιάς αποτελείται από τις καλύτερες n-άδες της προηγούμενης (carry-

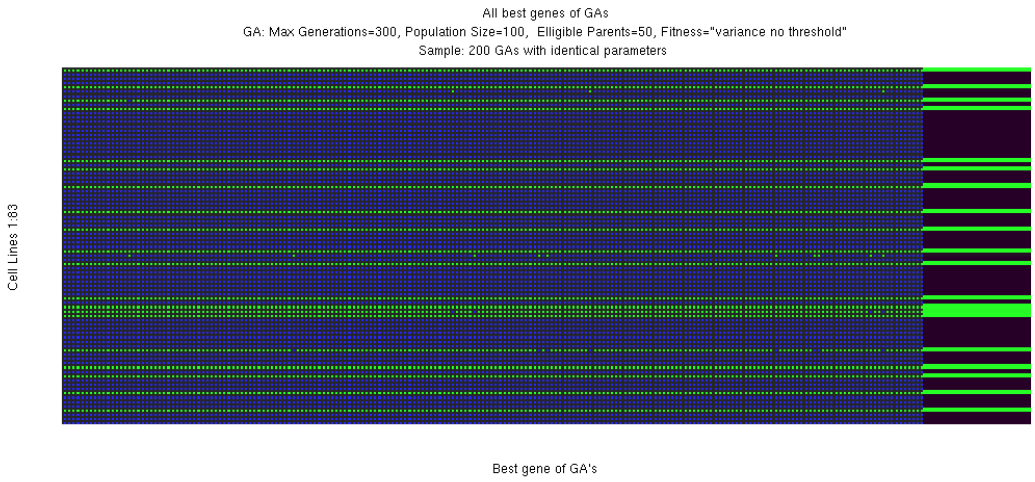
over), δηλαδή 2 από τις 100 20-άδες του πληθυσμού.

- Το 8% της νέας γενιάς προέρχεται από n-άδες που έχουν επιλεγεί με την τεχνική της μετάλλαξης, δηλαδή 8 από τις 100 20-άδες του πληθυσμού.
- Το 10% της νέας γενιάς επιλέγεται με τυχαίο τρόπο, δηλαδή 10 20-άδες του νέου πληθυσμού.
- Το 80% της νέας γενιάς αποτελείται από 20-άδες που έχουν επιλεγεί με την τεχνική της cross-over διασταύρωσης, δηλαδή 80 από τις 100 20-άδες του νέου πληθυσμού. Στην περίπτωση αυτού του αλγορίθμου οι "υποψήφιοι γονείς" είναι 50.

Εκτελέσαμε και πάλι 200 ΓΑ, με συνάρτηση καταλληλότητας και μέγιστο αριθμό γενεών όπως και παραπάνω. Τα αποτελέσματα που πήραμε ήταν τα εξής:



Σχεδιάγραμμα 20: Διάγραμμα των καλύτερων σκορ 200 ΓΑ με πληθυσμό 100



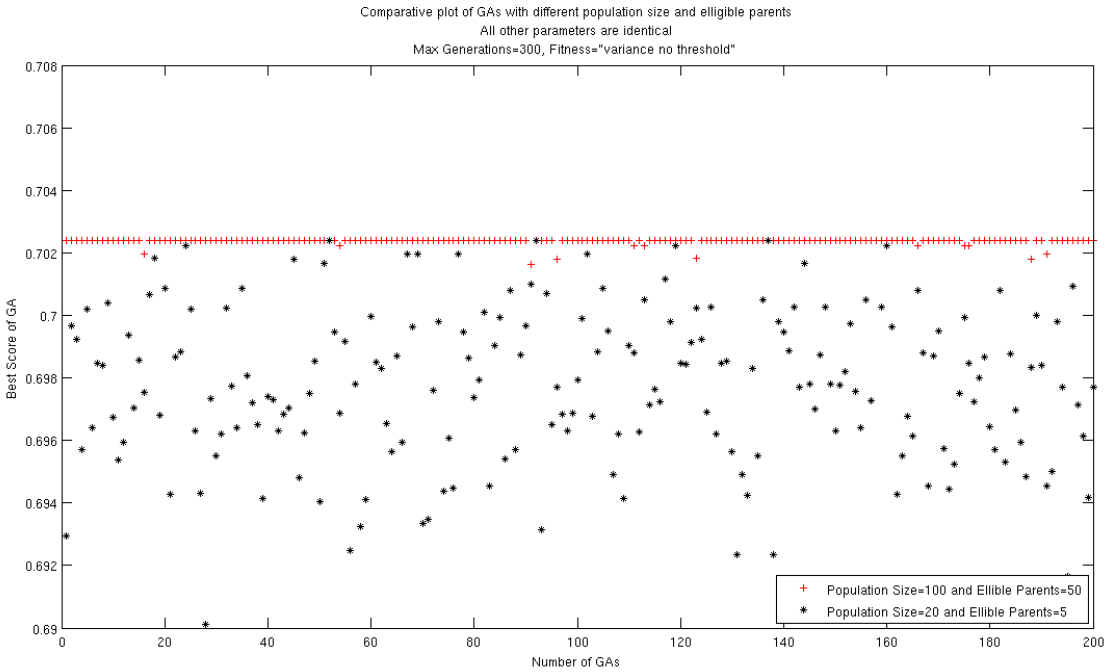
Σχεδιάγραμμα 21: Οι καλύτερες 20-άδες κυτταρικών σειρών για πληθυσμό 100

Όπως παρατηρούμε από τα παραπάνω διαγράμματα, ο ΓΑ με μεγάλο πληθυσμό πετυχαίνει για την πλειοψηφία των πειραμάτων την ίδια υψηλή τιμή της συνάρτησης καταλληλότητας, καθώς επίσης αποφασίζει, θα μπορούσαμε να πούμε σχεδόν ντετερμινιστικά, την καλύτερη 20-άδα κυτταρικών σειρών. Αντίθετα, στον ΓΑ με μικρό πληθυσμό παρατηρούμε ότι υπάρχει αρκετά μεγάλη διακύμανση στην καλύτερη τιμή της συνάρτησης καταλληλότητας που επιτυγχάνει, όπως επίσης και στις 20-άδες κυτταρικών σειρών που επιλέγει.

Με βάση όλα τα παραπάνω, μπορούμε με βεβαιότητα να πούμε ότι οι Γ.Α. έχουν καλύτερη συμπεριφορά όταν έχουμε μεγάλο πληθυσμό σε σχέση με αυτούς που έχουν μικρό πληθυσμό. Βλέπουμε ότι κρατώντας όλες τις παραμέτρους ίδιες και αλλάζοντας μόνο το μέγεθος του πληθυσμού οι Γ.Α. με μεγάλο πληθυσμό εμφανίζουν μία συνέπεια γύρω από το ποσοστό επιτυχίας και την προτεινόμενη λύση, εκεί που οι Γ.Α. με μικρότερο πληθυσμό εμφανίζουν μεγάλες διαφορές στις προτάσεις τους.

Στο γράφημα που ακολουθεί με κόκκινα '+' παρουσιάζονται οι καλύτερες τιμές καταλληλότητας κάθε ΓΑ με πληθυσμό 100, ενώ με μαύρα '*' οι καλύτερες τιμές καταλληλότητας κάθε ΓΑ με πληθυσμό 20.

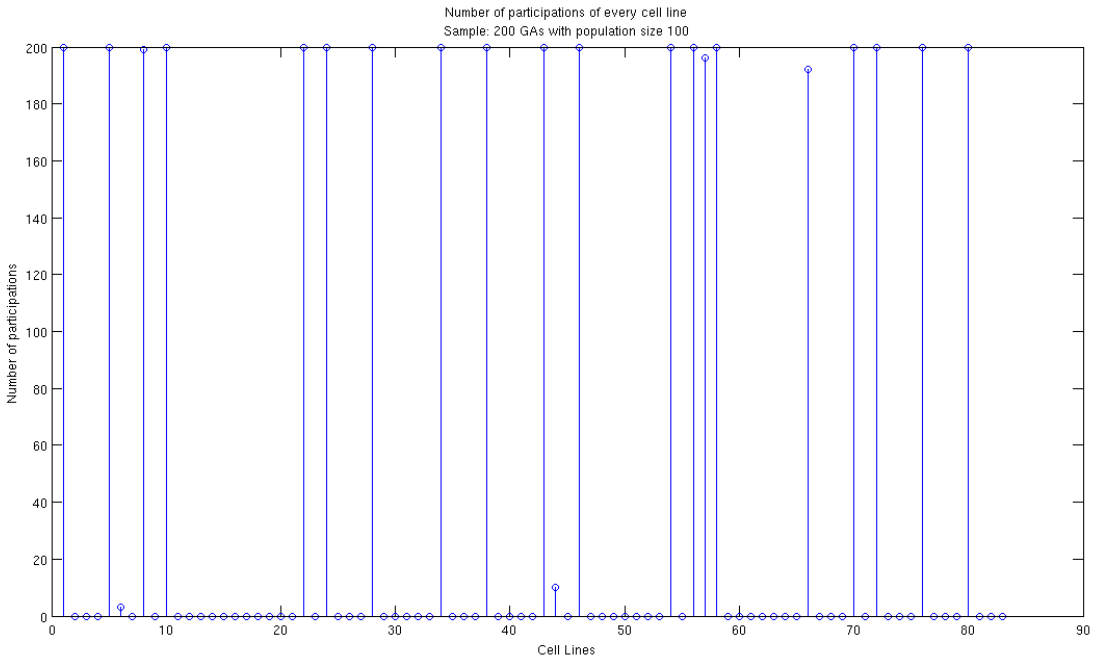
Και για τους δύο ΓΑ εκτελέσαμε 200 πειράματα. Στον άξονα των x έχουμε τον αριθμό του κάθε πειράματος και στον άξονα των y την βέλτιστη λύση που προτείνει ο κάθε ΓΑ, έχοντας ως μετρική $metric = 1/(1+variance), metric \in [0,1]$. Εύκολα παρατηρούμε ότι οι ΓΑ με μεγάλο πληθυσμό στην πλειοψηφία τους συμφωνούν σε ότι αφορά την βέλτιστη λύση, ενώ οι ΓΑ με μικρό πληθυσμό εμφανίζουν μεγάλες διακυμάνσεις στις τιμές απόφασης με μόνο τρεις από τους 200 να επιτυγχάνουν την βέλτιστη τιμή. Επίσης, στο γράφημα αυτό γίνεται ακόμα πιο ξεκάθαρος ο σχεδόν ντετερμινιστικός τρόπος με τον οποίο οι ΓΑ με μεγάλο πληθυσμό "αποφασίζουν" την βέλτιστη 20-άδα, σε αντίθεση με τους ΓΑ με μικρό πληθυσμό.



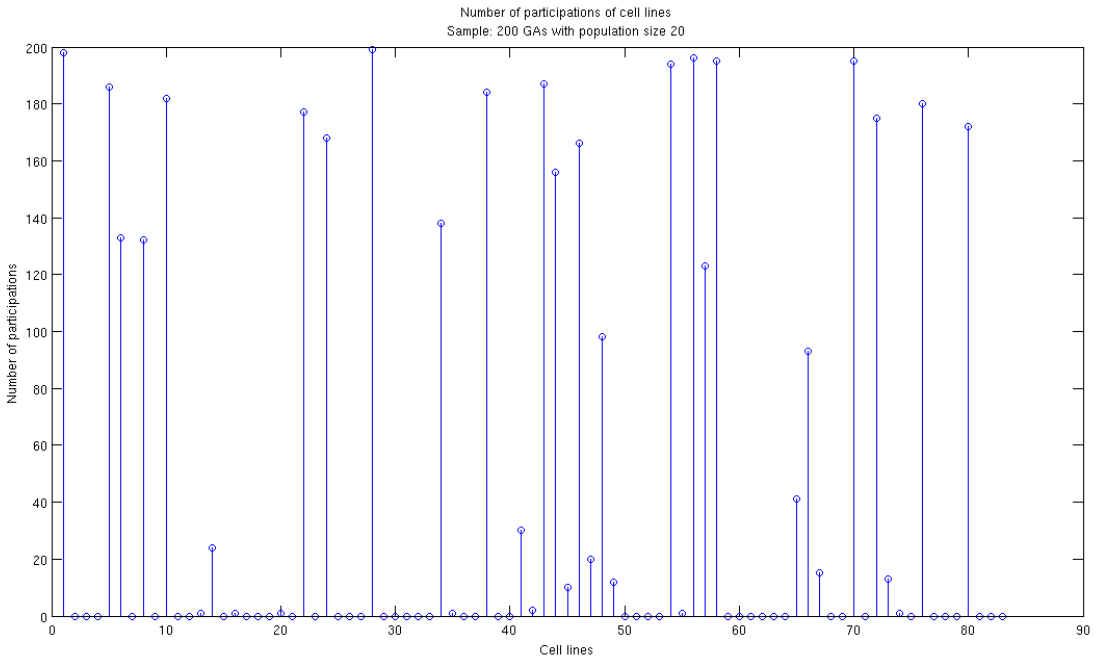
Σχεδιάγραμμα 22: Σύγκριση καλύτερων τιμών καταλληλότητας ΓΑ για πληθυσμό 20 και 100

Στις δύο γραφικές που ακολουθούν βλέπουμε για κάθε κυτταρική σειρά πόσες φορές συμμετέχει στα 200 πειράματα που έχουμε τρέξει, τόσο για τον ΓΑ με πληθυσμό 100, όσο και για τον ΓΑ με πληθυσμό 20. Στον x-άξονα έχουμε τις 83 κυτταρικές σειρές και στον y-άξονα έχουμε το πόσες φορές μία κυτταρική σειρά συμμετέχει στην πρόταση των 200 Γ.Α. που έχουμε εκτελέσει. Είναι ακόμα πιο φανερός ο ντετερμινιστικός σχεδόν τρόπος που αποφασίζει ο ένας ΓΑ σε σχέση με τον άλλον, καθώς βλέπουμε ότι η πλειοψηφία των κυτταρικών σειρών που συμμετέχουν στους ΓΑ με πληθυσμό 100 συμμετέχει και στα 200 πειράματα. Αντίθετα, στους ΓΑ με πληθυσμό 20 μόνο μία κυτταρική σειρά

συμμετέχει και στα 200 πειράματα.



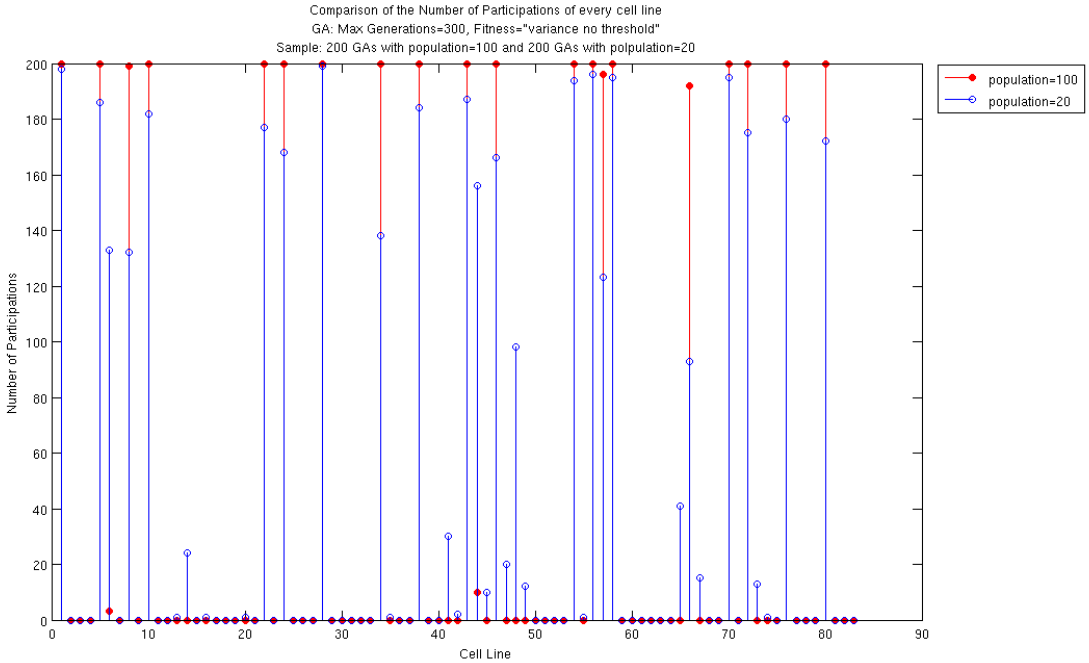
Σχεδιάγραμμα 23: Συμμετοχή κυτταρικών σειρών στα πειράματα με πληθυσμό 100



Σχεδιάγραμμα 24: Συμμετοχή κυτταρικών σειρών στα πειράματα με πληθυσμό 20

Συμπερασματικά, μπορούμε να πούμε ότι ένας ΓΑ με μεγάλο πληθυσμό είναι προτιμότερος από έναν ΓΑ με μικρό, τουλάχιστον σε ότι αφορά το θέμα της βελτιστοποίησης της ομοιομορφίας.

Το παρακάτω γράφημα αποτελεί εναπόθεση των δύο παραπάνω γραφημάτων. Και πάλι, στον x-άξονα έχουμε τις κυτταρικές σειρές και στον y-άξονα έχουμε τον αριθμό συμμετοχών κάθε κυτταρικής σειράς στην προτεινόμενη λύση, για 200 πειράματα. Με κόκκινο χρώμα είναι τα αποτελέσματα για πληθυσμό γενιάς 100 20-άδων και με μπλε χρώμα τα αποτελέσματα για πληθυσμό γενιάς 20 20-άδων.



Σχεδιάγραμμα 25: Σύγκριση καλύτερων κυτταρικών σειρών για πληθυσμό γενιάς 20 και 100 20-άδων

Παρατηρούμε από το παραπάνω γράφημα ότι οι προτεινόμενες κυτταρικές σειρές που προτείνουν οι ΓΑ με πληθυσμό 100 έχουν πολύ μεγάλη “προτίμηση” και από τους ΓΑ με πληθυσμό 20. Παράλληλα, οι κυτταρικές σειρές που προτείνονται λιγότερες φορές από τους ΓΑ με πληθυσμό 20 στην πλειοψηφία τους δεν προτείνονται καθόλου από τους ΓΑ με πληθυσμό 100, ενώ υπάρχουν και δύο κυτταρικές σειρές (οι 6 και 44) που αν και οι ΓΑ με πληθυσμό 20 τις επιλέγουν αρκετές φορές, οι ΓΑ με πληθυσμό 100 τις επιλέγουν ελάχιστες φορές. Μπορούμε επομένως να βγάλουμε το συμπέρασμα ότι οι “καλές” κυτταρικές σειρές επιλέγονται ανεξάρτητα από το μέγεθος του πληθυσμού του ΓΑ.

Στους ΓΑ με μεγάλο πληθυσμό έχουμε αύξηση των πιθανοτήτων να επιλεγούν υποψήφιες λύσεις με υψηλές τιμές καταλληλότητας και κατά την διάρκεια της εξέλιξης του αλγορίθμου οι λύσεις αυτές να

εξελιχθούν σε ακόμα πιο κατάλληλες. Στους ΓΑ με στενό πληθυσμό οι πιθανότητες αυτές είναι σαφώς μικρότερες, αφού από 100 υποψήφιες λύσεις που δοκιμάζονται σε κάθε γενιά στους ΓΑ με μεγάλο πληθυσμό πέφτουμε στις 20 ανά γενιά στους ΓΑ με στενό πληθυσμό. Επίσης, έχοντας 100 υποψήφιες λύσεις εκ των οποίων οι 80 προκύπτουν από cross-over διασταυρώσεις τα πλέον κατάλληλα μέλη κάθε γενιάς εξελίσσονται πιο γρήγορα σε αντίθεση με τις 20 υποψήφιες λύσεις όπου μόνο 10 από αυτές προκύπτουν από την ίδια διαδικασία. Αποτέλεσμα αυτής της διαδικασίας είναι ταυτόχρονα το ξεκαθάρισμα κυτταρικών σειρών που έχουν θόρυβο από τις υποψήφιες λύσεις πιο αποτελεσματικά όταν έχουμε μεγάλο πληθυσμό, ενώ στην περίπτωση του μικρού πληθυσμού οι σειρές αυτές είναι πιο δύσκολο να απομονωθούν.

Η παράμετρος των ΓΑ που είναι διαφορετική στα πειράματα αυτά είναι μόνο το μέγεθος του πληθυσμού. Το κριτήριο τερματισμού είναι στις 300 γενιές, που όπως έχουμε τεκμηριώσει προηγουμένως δεν είναι επαρκές για τους ΓΑ με μικρό πληθυσμό. Ο λόγος που δεν αλλάξαμε το κριτήριο τερματισμού για τους ΓΑ με πληθυσμό 20 ήταν ότι μπορούμε ουσιαστικά να έχουμε καλή εποπτεία των αποτελεσμάτων και της συμπεριφοράς των ΓΑ μόνο όταν αλλάζουμε μία παράμετρο στο κάθε πείραμα.

4.2 ΓΑ με “υβριδική” συνάρτηση καταλληλότητας

Από την μελέτη που παρουσιάσαμε στις προηγούμενες παραγράφους καταλήξαμε στο συμπέρασμα ότι για την μεγιστοποίηση της ομοιομορφίας ο πιο κατάλληλος ΓΑ είναι αυτός που έχει μεγάλο πληθυσμό και η ελαχιστοποίηση της διασποράς γίνεται χωρίς κατωφλίωση. Πάνω σε αυτό το συμπέρασμα εφαρμόσαμε την “υβριδική” συνάρτηση καταλληλότητας, όπως αυτή παρουσιάστηκε στο κεφάλαιο 3. Ο ΓΑ έχει πληθυσμό 100 n-άδες, εκ των οποίων 2 έχουν προέλθει με την τεχνική του carry-over, 10 έχουν εισαχθεί με τυχαίο τρόπο, 8 έχουν προέλθει από μετάλλαξη και 80 από cross-over διασταύρωση, από 50 “γονείς”.

4.2.1 Αποτελέσματα με μέγεθος γονιδίου ΓΑ 20

Ο ΓΑ που εφαρμόσαμε αρχικά έχει μέγεθος γονιδίου 20 κυτταρικές σειρές, θεωρώντας έπειτα από τη σχετική μελέτη που παρουσιάστηκε στο κεφάλαιο 1, ότι 20 κυτταρικές σειρές είναι ένα ικανοποιητικό μείγμα. Εκτελέσαμε 200 πειράματα με πληθυσμό 100 και κριτήριο τερματισμού τις 300 γενιές. Στον

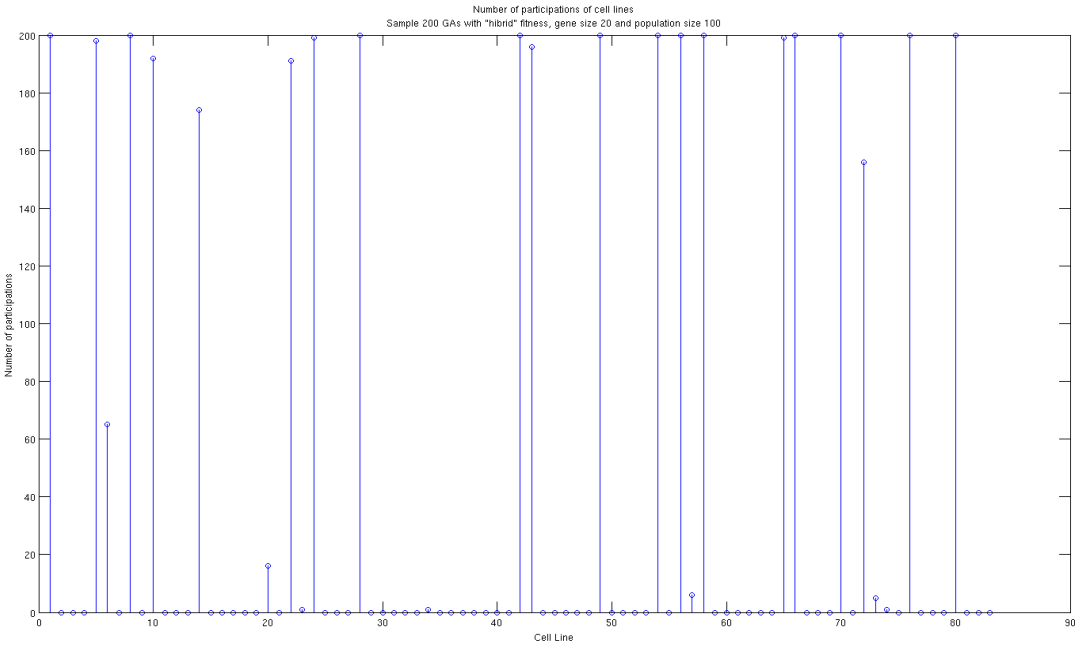
παρακάτω πίνακα παρουσιάζονται οι μέγιστες, οι ελάχιστες και ο μέσος όρος των τιμών τόσο της διασποράς όσο και της έκφρασης των γονιδίων μέσα στις κυτταρικές σειρές στα 200 πειράματα που εκτελέσαμε. Ο υπολογισμός της διασποράς γίνεται πάνω σε λογαριθμημένα δεδομένα, ενώ η έκφραση υπολογίζεται σε δεδομένα που δεν έχουν υποστεί κανενός είδους κανονικοποίηση.

	Min	Max	Mean
Variance	0.4380	0.4543	0.4426
Expression	1.8546e+03	1.8755e+03	1.8628e+03

Table 6: Μέγιστες, ελάχιστες και μέσες τιμές διασποράς και έκφρασης στα 200 πειράμτα

Στο παρακάτω γράφημα απεικονίζονται για κάθε ένα από τα 200 πειράματα οι κυτταρικές σειρές που θεωρούνται οι καλύτερες από τον ΓΑ. Μπορούμε να παρατηρήσουμε ότι η πλειοψηφία των κυτταρικών σειρών που αποφασίζουν οι ΓΑ αποφασίζονται από όλους τους ΓΑ.

Σχεδιάγραμμα 26: Οι καλύτερες 20-άδες κυτταρικών σειρών με "υβριδική" συνάρτηση καταλληλότητας



Σχεδιάγραμμα 27: Συμμετοχή κυτταρικών σειρών στα 200 "υβριδικά" πειράματα με μέγεθος γονιδίου ΓΑ 20

Στην παραπάνω γραφική βλέπουμε για κάθε κυτταρική σειρά πόσες φορές συμμετέχει στα 200 πειράματα που έχουμε τρέξει.

Αν και έχουμε υπολογίσει την μέση τιμή της έκφρασης των γονιδίων στις 20-άδες των κυτταρικών σειρών που κατέληξαν οι ΓΑ, μένει να δούμε το ποσοστό κάλυψης των γονιδίων. Πως αναφέρουμε και σε προηγούμενα κεφάλαια, ως ποσοστό κάλυψης νοείται το ποσοστό των γονιδίων μέσα στο μείγμα το οποίο έχει τιμή μεγαλύτερη από το κατώφλι έκφρασης το οποίο έχουμε υπολογίσει και στην περίπτωση των δεδομένων που έχουμε η τιμή αυτή είναι 100. Αν και έχουμε αφαιρέσει τα γονίδια τα οποία σε καμία από τις κυτταρικές σειρές δεν έχουν τιμή πάνω από το κατώφλι, στα δεδομένα μας υπάρχουν γονίδια τα οποία σε κάποιες κυτταρικές σειρές υποεκφράζονται. Ο μέσος αριθμός γονιδίων που εκφράζονται στα 200 πειράματα που εκτελέσαμε είναι 13987 από τα 17012 γονίδια που έμειναν μετά την αφαίρεση αυτών που δεν εκφράζονται πουθενά, δηλαδή ποσοστό έκφρασης 82.1652%. Αρχικά είχαμε 22283 γονίδια, άρα από αυτά εκφράζεται τελικά ποσοστό της τάξης του 62.7698%.

(Αναλυτικά τα αποτελέσματα των 200 πειραμάτων βρίσκονται στο παράρτημα)

4.2.2 Αποτελέσματα με μεταβαλλόμενο μέγεθος γονιδίου ΓΑ

Στην εργασία αυτή, ασχοληθήκαμε πολύ με μείγμα μεγέθους 20 κυτταρικών σειρών, μιας και ένα τέτοιο δείγμα είναι με βάση την βιβλιογραφική έρευνα του 1ου κεφαλαίου είναι αποδεκτό, αν και μεγάλο σε μέγεθος.

Στη συνέχεια θελήσαμε να δούμε την συμπεριφορά της μεθόδου μας, αλλάζοντας το μέγεθος της n-άδας των κυτταρικών σειρών. Για τον λόγο αυτό, εκτελέσαμε πειράματα όπου το μέγεθος της n-άδας ήταν από 10 έως 30 κυτταρικές σειρές. Για κάθε n-άδα εκτελέσαμε 5 ΓΑ, όπου όλα τα υπόλοιπα στοιχεία του πειράματος είναι ίδια με το προηγούμενο πείραμα. Συγκεντρωτικά, τα αποτελέσματα που λάβαμε είναι τα εξής:

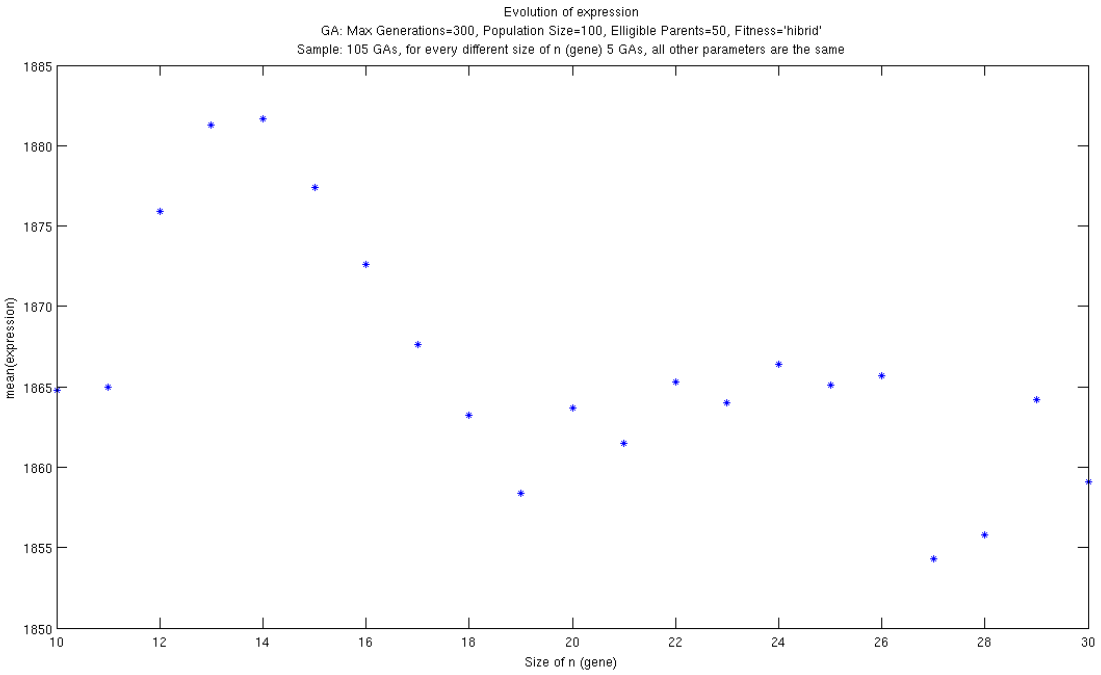
Μέγεθος n-άδας	Variance			Expression			Ποσοστό κάλυψης μειωμένου πίνακα	Ποσοστό κάλυψης αρχικού πίνακα
	Min	Max	Mean	Min	Max	Mean	Mean	Mean
10-άδα	0.3477	0.3618	0.3505	1.8627e+03	1.8733e+03	1.8648e+03	77.1408%	58.8933%
11-άδα	0.3640	0.3640	0.3640	1.865e+03	1.865e+03	1.865e+03	77.8215%	59.4130%
12-άδα	0.3825	0.3846	0.3829	1.8756e+03	1.876e+03	1.8759e+03	78.3870%	59.8447%
13-άδα	0.3963	0.3963	0.3963	1.8813e+03	1.8813e+03	1.8813e+03	78.8972%	60.2343%
14-άδα	0.4060	0.4060	0.4060	1.8817e+03	1.8817e+03	1.8817e+03	79.4380%	60.6471%
15-άδα	0.4135	0.4135	0.4135	1.8774e+03	1.8774e+03	1.8774e+03	80.0024%	61.0780%
16-άδα	0.4184	0.4184	0.4184	1.8726e+03	1.8726e+03	1.8726e+03	80.6960%	61.6075%
17-άδα	0.4234	0.4234	0.4234	1.8676e+03	1.8676e+03	1.8676e+03	80.9194%	61.7780%
18-άδα	0.4278	0.4278	0.4278	1.8632e+03	1.8632e+03	1.8632e+03	81.1662%	61.9665%
19-άδα	0.4314	0.4314	0.4314	1.8584e+03	1.8584e+03	1.8584e+03	81.4543%	62.1864%
20-άδα	0.4413	0.4500	0.4431	1.8616e+03	1.8719e+03	1.8637e+03	82.2149%	62.7671%
21-άδα	0.4471	0.4497	0.4476	1.8615e+03	1.8615e+03	1.8615e+03	82.5053%	62.9888%
22-άδα	0.4519	0.4656	0.4593	1.8572e+03	1.8723e+03	1.8653e+03	82.9567%	63.3335%
23-άδα	0.4613	0.4678	0.4637	1.8615e+03	1.8699e+03	1.8640e+03	83.3811%	63.6575%
24-άδα	0.4652	0.4803	0.4729	1.857e+03	1.8726e+03	1.8664e+03	83.8044%	63.9806%
25-άδα	0.4684	0.4829	0.4790	1.8525e+03	1.8702e+03	1.8651e+03	84.1818%	64.2687%
26-άδα	0.4809	0.4890	0.4840	1.861e+03	1.8703e+03	1.8657e+03	84.4768%	64.4940%
27-άδα	0.4724	0.4898	0.4827	1.8415e+03	1.8624e+03	1.8543e+03	84.6520%	64.6277%
28-άδα	0.4805	0.4962	0.4889	1.8438e+03	1.8658e+03	1.8558e+03	84.9800%	64.8782%
29-άδα	0.4969	0.5068	0.5022	1.8579e+03	1.8708e+03	1.8642e+03	85.3362%	65.1501%
30-άδα	0.4965	0.5087	0.5041	1.8468e+03	1.8645e+03	1.8591e+03	85.6666%	65.4023%

Table 7: Ελάχιστες, μέγιστες και μέσες τιμές διασποράς και έκφρασης, καθώς και μέσα ποσοστά κάλυψης n-άδων

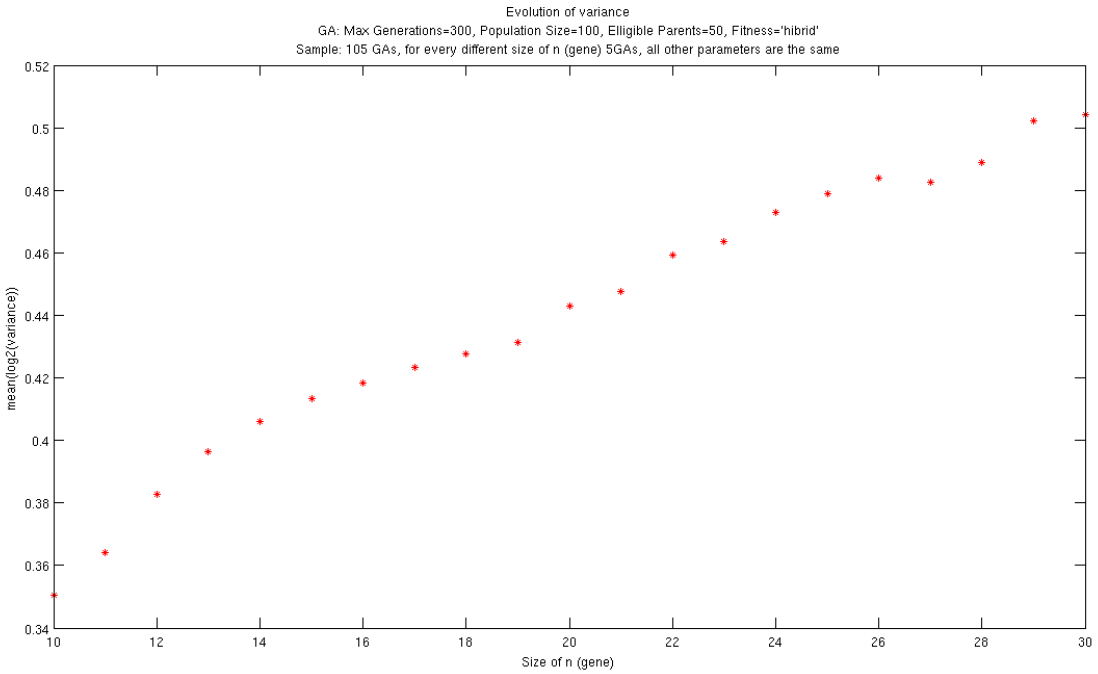
Από τον παραπάνω πίνακα μπορούμε να παρατηρήσουμε ότι όσο “μεγαλώνει σε μέγεθος” η υποψήφια λύση, αυξάνεται η διασπορά των γονιδίων μέσα στο μείγμα, ενώ η μέση τιμή της έντασης ταλαντώνεται γύρω από την τιμή 1.8663e+03. Ταυτόχρονα, παρατηρείται αύξηση του ποσοστού

κάλυψης των γονιδίων.

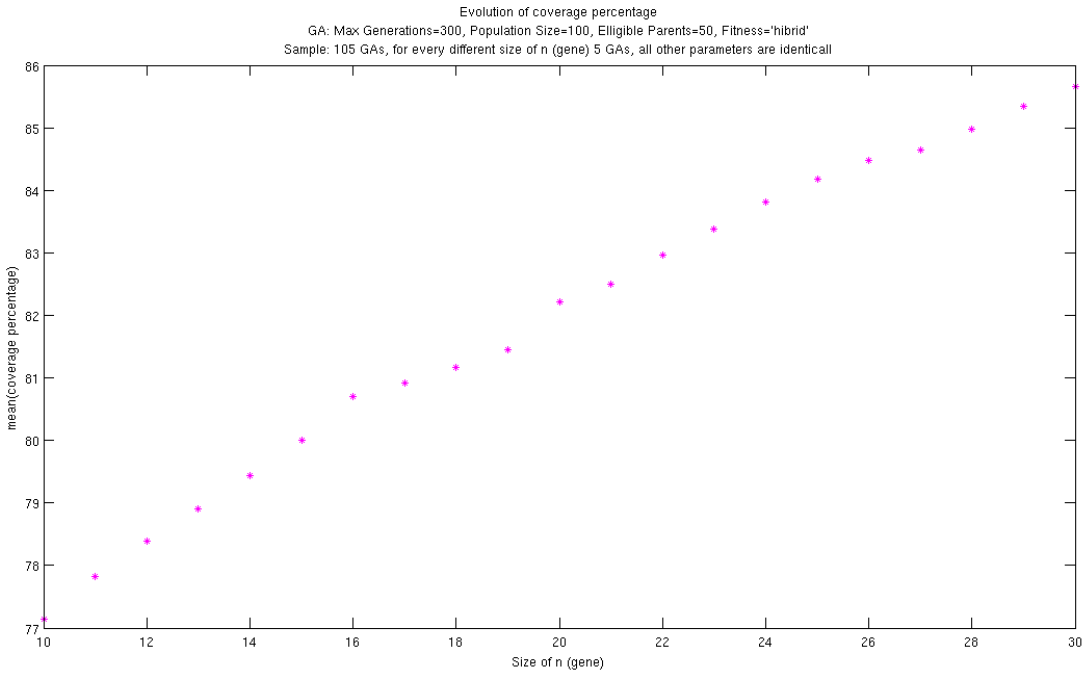
Στις παρακάτω γραφικές βλέπουμε για κάθε διαφορετικό μέγεθος n-άδας (γονιδίου) στον άξονα των x, την μέση τιμή της έκφρασης των γονιδίων για τους 5 ΓΑ που τρέξαμε (Σχεδ. 28), την μέση τιμή της διασποράς των γονιδίων (Σχεδ. 29) και το μέσο ποσοστό κάλυψης του μείγματος (Σχεδ. 30).



Σχεδιάγραμμα 28: Εξέλιξη της έκφρασης των γονιδίων σε ΓΑ με διαφορετικό μέγεθος γονιδίου



Σχεδιάγραμμα 29: Εξέλιξη της διασποράς των γονιδίων σε ΓΑ με διαφορετικό μέγεθος γονιδίου



Σχεδιάγραμμα 30: Εξέλιξη του ποσοστού κάλυψης των γονιδίων σε ΓΑ με διαφορετικό μέγεθος γονιδίου

Αυτό που μπορούμε να παρατηρήσουμε στις τρεις παραπάνω γραφικές παραστάσεις, είναι ότι όσο αυξάνουμε το μέγεθος της n-άδας, δηλαδή όσο περισσότερες κυτταρικές σειρές συμμετέχουν στο μείγμα, τόσο αυξάνονται τόσο η διασπορά των γονιδίων μέσα στο μείγμα, όσο και το ποσοστό κάλυψης του μείγματος. Οι αυξήσεις αυτές είναι σχεδόν γραμμικές. Αντίθετα, η τιμή της έκφρασης των γονιδίων δείχνει να ταλαντώνεται. Η μέση τιμή της έκφρασης όλων των παραπάνω ΓΑ είναι 1.8663e+03.

Τα παραπάνω αποτελέσματα ήταν αναμενόμενα. Η αύξηση των κυτταρικών σειρών που συμμετέχουν σημαίνει ταυτόχρονα και αύξηση της πιθανότητας γονίδια που δεν εκφράζονταν να εκφράζονται, άρα και αύξηση του ποσοστού κάλυψης. Ταυτόχρονα, αφού έχουμε περισσότερα γονίδια να συμμετέχουν περιμένουμε να υπάρξει και αύξηση της διασποράς, αφού είναι προφανές ότι τα γονίδια αυτά λειτουργούν προσθετικά στο μέγεθος της διασποράς, όντας σχετικά “χειρότερης ποιότητας” μιας και τα καλύτερα έχουν ήδη επιλεγεί. Σε ότι αφορά το μέγεθος της έκφρασης των γονιδίων, δηλαδή της έντασής τους μέσα στις κυτταρικές σειρές, θα περιμέναμε να υπάρχει μία μείωση της τιμής αυτής όσο αυξάνεται το μέγεθος του μείγματος. Ενώ αυτή η μείωση στην πλειοψηφία των πειραμάτων συμβαίνει, υπάρχουν και πειράματα όπου η τιμή αυτή αυξάνεται σε σχέση με το αμέσως προηγούμενο πείραμα, όπου το μείγμα ήταν μικρότερο κατά μία κυτταρική σειρά. Και αυτό το φαινόμενο, όμως, εξηγείται αν λάβουμε υπ' όψιν ότι κάθε κυτταρική σειρά για κάθε γονίδιο έχει και διαφορετική τιμή έντασης. Αυτός είναι άλλωστε και ο λόγος που περιμέναμε αύξηση της διασποράς. Επομένως, μία κυτταρική σειρά μπορεί να έχει υψηλή τιμή έκφρασης των γονιδίων αλλά και υψηλή διασπορά αυτών. Όσο μικρότερο είναι το μείγμα, τόσο οι κυτταρικές σειρές αυτές απορρίπτονται. Όσο όμως αυξάνεται σε μέγεθος, τόσο αυξάνονται και οι πιθανότητες συμμετοχής τους στο μείγμα.

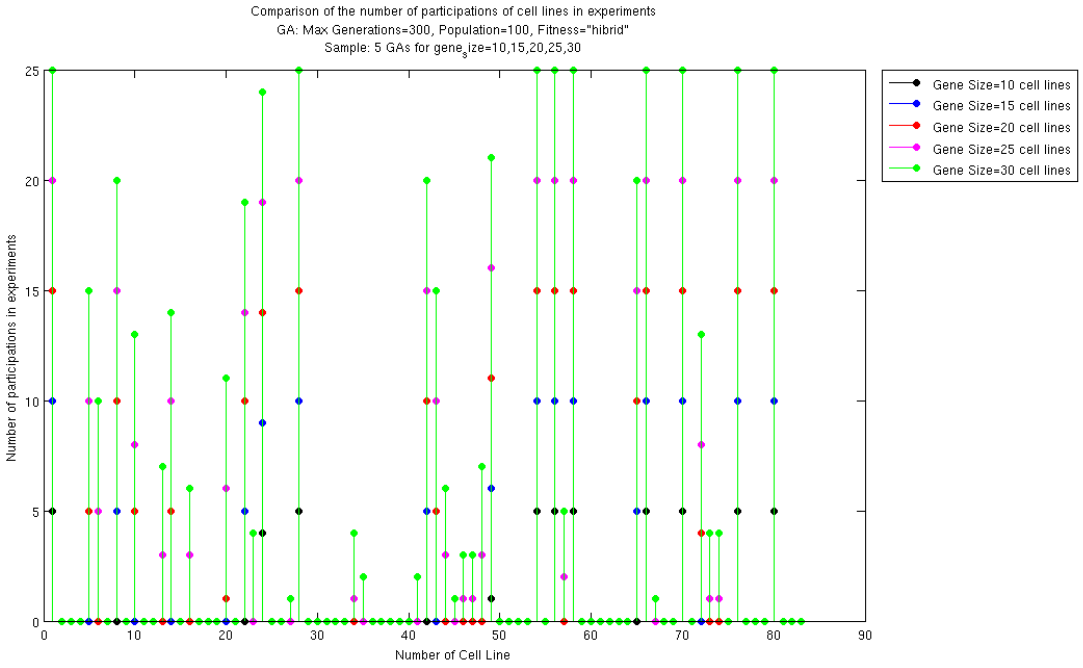
Τελικά, μπορούμε να εξάγουμε το συμπέρασμα ότι η αύξηση του αριθμού των κυτταρικών σειρών που συμμετέχουν στο μείγμα οδηγεί σε πολύ μικρό ποσοστό την αύξηση του ποσοστού κάλυψης των γονιδίων, ενώ ταυτόχρονα έχουμε αρκετά μεγάλη αύξηση της διασποράς. Η έκφραση των γονιδίων μέσα στις κυτταρικές σειρές μπορούμε να θεωρήσουμε λόγω της πολύ μικρής αυξομείωσης που παρουσιάζει ότι δεν εμφανίζει πρόοδο.

Η μείωση του αριθμού των κυτταρικών σειρών, οδηγεί σε μείωση της διασποράς κατά ένα αρκετά μεγάλο ποσοστό, και κυρίως όταν έχουμε μείωση του αριθμού των κυτταρικών σειρών κατά 50%, δηλαδή μείγμα 10 κυτταρικών σειρών. Η αύξηση της έκφρασης παραμένει σε πολύ μικρό ποσοστό,

οπότε και πάλι μπορούμε να την θεωρήσουμε αμελητέα, ενώ η μείωση του ποσοστού κάλυψης μπορεί να θεωρηθεί σημαντική ως ποσοστό.

Όπως βλέπουμε από τις τρεις παραπάνω γραφικές παραστάσεις, το βέλτιστο μείγμα κυτταρικών σειρών είναι αυτό που αποτελείται από 20 κυτταρικές σειρές. Από τα πειραματικά δεδομένα που έχουμε βλέπουμε ότι στα πειράματα με 20 κυτταρικές σειρές το ποσοστό κάλυψης είναι αρκετά υψηλό, όπως και η μέση τιμή έκφρασης, ενώ ταυτόχρονα η μέση τιμή διασποράς μένει σε πολύ χαμηλές τιμές. Για τα μείγματα με λιγότερες από 20 κυτταρικές σειρές αν και έχουμε πολύ χαμηλές μέσες τιμές διασποράς και υψηλές μέσες τιμές έκφρασης “χάνουμε” πολύ από το ποσοστό της κάλυψης, ενώ για περισσότερες από 20 κυτταρικές σειρές αν και η μέση τιμή έκφρασης δεν παρουσιάζει μεγάλες αποκλίσεις και υπάρχει αξιόλογο κέρδος στο ποσοστό της κάλυψης, οι μέσες τιμές διασποράς αυξάνονται αρκετά ώστε να θεωρούμε το μείγμα ακατάλληλο.

Στην παρακάτω γραφική παράσταση έχουμε τις προτεινόμενες κυτταρικές σειρές των ΓΑ για μέγεθος “γονιδίου” 10, 15, 20, 25 και 30 κυτταρικών σειρών. Κάθε ένας από αυτούς τους ΓΑ έχει εκτελεστεί 5 φορές.



Σχεδιάγραμμα 31: Συμμετοχές κυτταρικών σειρών σε διαφορετικά μεγέθη "γονιδίου"

Στον άξονα των x έχουμε τους αριθμούς των κυτταρικών σειρών και στον άξονα των y τον αριθμό συμμετοχών στα πειράματα. Με μαύρο χρώμα σημειώνονται οι συμμετοχές σε ΓΑ με μέγεθος "γονιδίου" 10 κυτταρικών σειρών, με μπλε οι συμμετοχές σε ΓΑ με μέγεθος "γονιδίου" 15 κυτταρικών σειρών, με κόκκινο οι συμμετοχές σε ΓΑ με μέγεθος "γονιδίου" 20 κυτταρικών σειρών, με μωβ οι συμμετοχές σε ΓΑ με μέγεθος "γονιδίου" 25 κυτταρικών σειρών και τέλος με πράσινο οι συμμετοχές σε ΓΑ με μέγεθος "γονιδίου" 30 κυτταρικών σειρών. Βλέπουμε ότι οι κυτταρικές σειρές :

1 – 28 – 54 – 56 – 58 – 66 – 70 – 76 – 80

συμμετέχουν σε όλα τα πειράματα, συνολικά 25 δηλαδή, ανεξαρτήτως του μεγέθους του "γονιδίου" του ΓΑ. Όσο αυξάνεται ο αριθμός των κυτταρικών σειρών στο "γονίδιο", παρατηρούμε ότι οι νέες κυτταρικές σειρές που επιλέγονται, επιλέγονται με "επιμονή", δηλαδή όχι μόνο από τον ΓΑ με μεγαλύτερο μέγεθος "γονιδίου" αλλά και από τον επόμενο. Αν θέλουμε, όμως, να πούμε ποιες είναι οι καλές κυτταρικές σειρές ανεξαρτήτων του μεγέθους του "γονιδίου", τότε αυτές είναι οι 9 κυτταρικές σειρές που έχουμε παραπάνω.

Λαμβάνοντας υπ' όψιν όλα τα παραπάνω δεδομένα που εξαγάγαμε από τα πειράματα, θεωρούμε ότι τελικά το καλύτερο μείγμα αναφοράς αποτελείται από 20 κυτταρικές σειρές. Το μείγμα των 20 κυτταρικών έχει αρκετά χαμηλό ποσοστό διασποράς και εμφανίζει ικανοποιητική κάλυψη σε ότι αφορά τα γονίδια που συμμετέχουν σε αυτό, ενώ ταυτόχρονα βρίσκεται μέσα στα πλαίσια που έχουν προταθεί και από άλλα πειράματα, σε σχέση με το μέγεθος. Μείγματα με παραπάνω κυτταρικές σειρές, αν και εμφανίζουν μεγαλύτερα ποσοστά κάλυψης, χάνουν σε πολύ μεγάλο βαθμό την ομοιομορφία τους, η οποία αποτελεί ένα ποιοτικό κριτήριο ιδιαιτέρως σημαντικό. Αντίθετα, μείγματα με μικρότερο αριθμό κυτταρικών σειρών, παρουσιάζουν μεγαλύτερου βαθμού ομοιομορφία, αλλά δεν παρέχουν ικανοποιητική κάλυψη. Για τους λόγους αυτούς μπορούμε να ισχυριστούμε ότι το μείγμα των 20 κυτταρικών σειρών αποτελεί την "χρυσή τομή", δηλαδή παρέχει επαρκή κάλυψη και ικανοποιητικό βαθμό ομοιομορφίας.

4.3 ΓΑ vs. Genetic Algorithm Tool του Matlab

Ο ΓΑ που κατασκευάσαμε, όπως αναφέρουμε και σε προηγούμενο κεφάλαιο, έχει αρκετές διαφοροποιήσεις από ένα τυπικό Γενετικό Αλγόριθμο. Για τον λόγο αυτό, για να μπορέσουμε να συγκρίνουμε την αποδοτικότητα και επιτυχία του δικού μας ΓΑ με το εργαλείο για γενετικούς του

Matlab, έπρεπε να προσαρμόσουμε το toolbox του Matlab στην δική μας μεθοδολογία. Προσαρμόσαμε τις ρουτίνες παραγωγής γονιδίων του γενετικού, έτσι ώστε τα υποψήφια γονίδια να είναι μία n-άδα, τις ρουτίνες μετάλλαξης και cross-over και υλοποιήσαμε μία δική μας ρουτίνα για να κρατάμε την ιστορία του κάθε γενετικού, ώστε να μπορούμε να συλλέγουμε όλα τα δεδομένα που μας είναι απαραίτητα για την αξιολόγηση της μεθόδου. Σαν κριτήριο τερματισμού χρησιμοποιήσαμε και πάλι έναν μέγιστο αριθμό γενεών.

Επίσης έπρεπε να κατασκευάσουμε και μία συνάρτηση καταλληλότητας, ώστε να την εφαρμόσουμε και στο toolbox του Matlab, αλλά και στον δικό μας ΓΑ, ώστε να μπορέσει να γίνει η σύγκριση. Επειδή δεν θέλαμε να αναλώσουμε πολύ χρόνο στην διαδικασία προτιμήσαμε μία απλή συνάρτηση καταλληλότητας. Η συνάρτηση αυτή μετράει την επιτυχία του γενετικού να επιλέξει n-άδες κυτταρικών σειρών, για τις οποίες οι τιμές έντασης όλων των γονιδίων βρίσκονται πάνω από ένα κατώφλι. Πιο συγκεκριμένα, εκτελέσαμε 10 φορές έναν ΓΑ, τόσο με την δική μας μεθοδολογία όσο και με το gatool, με πληθυσμό 20, μέγεθος n-άδας 20, και γενιές που παράγονται η μεν πρώτη με τυχαίο τρόπο, οι δε επόμενες έχοντας 10 νέες 20-άδες από cross-over διασταύρωση με 5 “υποψήφιους γονείς”, 3 νέες 20-άδες που προέρχονται από μετάλλαξη, 2 carry-over και 5 με τυχαία επιλογή. Το κατώφλι που χρησιμοποιήσαμε στην συνάρτηση καταλληλότητας ήταν 200 και τα δεδομένα δεν είχαν υποστεί κανονικοποίηση. Μία ακόμα διαφορά που υπάρχει είναι πως ενώ η δική μας μέθοδος εντοπίζει ολικά μέγιστα, η μέθοδος του Matlab εντοπίζει ολικά ελάχιστα και αυτή ήταν άλλη μια τροποποίηση που έπρεπε να κάνουμε, δηλαδή να παίρνουμε την μέγιστη τιμή καταλληλότητας που υπολόγιζε το toolbox και να μετράμε το συμπληρωματικό της. Αυτός ήταν και ένας από τους λόγους που αποφασίσαμε να μην χρησιμοποιήσουμε το εργαλείο αυτό, δηλαδή ότι υπολογίζει ελάχιστα και όχι μέγιστα.

Αφού εκτελέσαμε τα 10 πειράματα, μετρήσαμε την μέση τιμή επιτυχίας και των δύο μεθόδων, καθώς και την απόκλιση των τιμών επιτυχίας τους. Ακολουθούν τα αποτελέσματα:

	Μέση τιμή επιτυχίας	Απόκλιση (std)
gatool	0.6555	0.018
Γενετικός Αλγ.	0.6576	0.011

Table 8: Αποτελέσματα gatool και ΓΑ

Η διαφορές τόσο στην μέση τιμή επιτυχίας, όσο και στην απόκλιση των τιμών επιτυχίας βλέπουμε ότι είναι πάρα πολύ μικρή και επίσης παρατηρούμε ότι αν και ο δικός μας ΓΑ έχει κατά λίγο μικρότερη μέση τιμή, έχει και μικρότερη απόκλιση στις τιμές αυτές. Όσο σε σχέση με τον χρόνο εκτέλεσης, και οι δύο μεθοδολογίες έχουν παραπλήσιους χρόνους.

Κεφάλαιο 5 – Συμπεράσματα και μελλοντική εργασία

Στην εργασία αυτή δημιουργήσαμε ένα πρότυπο μείγμα αναφοράς 20 κυτταρικών σειρών. Το μείγμα αυτό προέρχεται από ένα ευρύτερο data set, το οποίο περιλαμβάνει 83 κυτταρικές σειρές. Από αυτές τις 83 κυτταρικές σειρές, με την χρήση Γενετικών Αλγορίθμων, επιλέξαμε 20, τις οποίες και “αναμείξαμε” με αναλογία 1:1. Τα χαρακτηριστικά τα οποία επιθυμούσαμε να έχει το μείγμα αυτό ήταν αυτό της ομοιομορφίας και της όσο το δυνατόν μεγαλύτερης κάλυψης.

Για τον παράγοντα της ομοιομορφίας χρησιμοποιήσαμε ως κριτήριο ελαχιστοποίηση της διασποράς των γονιδίων μέσα στις κυτταρικές σειρές και εν συνεχεία στο μείγμα, ενώ για να έχουμε την μέγιστη δυνατή κάλυψη χρησιμοποιήσαμε ως κριτήριο την μεγιστοποίηση της μέσης τιμής κάθε γονιδίου μέσα στις κυτταρικές σειρές. Ο υπολογισμός της διασποράς, που ήταν το βασικό μας στοιχείο για την μεγιστοποίηση της ομοιομορφίας, έγινε σε κανονικοποιημένα δεδομένα, ενώ η μέτρηση της έκφρασης και κατ' επέκταση η μεγιστοποίησή της σε μη-κανονικοποιημένα. Στα πλαίσια της προεπεξεργασίας των δεδομένων, εντοπίσαμε το κατώφλι έκφρασης των γονιδίων και αφαιρέσαμε από τα δεδομένα μας τα γονίδια αυτά που σε καμία από τις 83 κυτταρικές σειρές δεν παρουσίαζε ένταση πάνω από το κατώφλι αυτό. Ο λόγος της αφαίρεσής τους ήταν ότι αφού κάθε κυτταρική σειρά μπορεί να συμμετάσχει στο μείγμα με συντελεστή βαρύτητας 1 και μόνο, τα γονίδια αυτά ποτέ δεν θα μπορούσαν να έχουν τιμή στο μείγμα πάνω από το κατώφλι αποκοπής. Ο συνδυασμός των δύο αυτών παραμέτρων, της μεγιστοποίησης της έκφρασης και της ομοιομορφίας, αποτελούσε και την συνάρτηση καταλληλότητας των Γενετικών Αλγορίθμων. Και εκτελέσαμε πολλούς και σε πολλές περιπτώσεις διαφορετικούς ΓΑ στην προσπάθειά μας να “συντονίσουμε” όλες τις παραμέτρους των ΓΑ, με τρόπο τέτοιο ώστε να μας δίνουν τις βέλτιστες λύσεις, μέσα σε εύλογο υπολογιστικό χρόνο. Ταυτόχρονα, εκτελέστηκαν πολλαπλοί ΓΑ με ίδιες παραμέτρους, έτσι ώστε να μπορέσουμε να βγάλουμε ένα ασφαλές συμπέρασμα για την ποιότητα των αποτελεσμάτων, μιας και οι ΓΑ δεν ανήκουν στην κατηγορία των ντετερμινιστικών αλγορίθμων βελτιστοποίησης.

Στα πλαίσια της εργασίας αυτής και λόγω των ιδιαιτεροτήτων του προβλήματος που κληθήκαμε να επιλύσουμε, ξεφύγαμε από την κλασσική υλοποίηση ενός ΓΑ. Στην υλοποίησή μας, για κάθε υποψήφιο μείγμα κυτταρικών σειρών εξετάζαμε για κάθε γονίδιο ξεχωριστά την διασπορά της έντασής του μέσα στο μείγμα και αυτήν τη διασπορά θέλαμε να ελαχιστοποιήσουμε, ενώ παράλληλα μεγιστοποιούσαμε την μέση τιμή της έντασής του μέσα στο μείγμα. Τα γονίδια, επομένως, του ΓΑ, δεν ήταν

μονοδιάστατα arrays, όπως συνηθίζεται, αλλά δισδιάστατοι πίνακες. Οι ΓΑ που εκτελέσαμε διέφεραν μεταξύ τους σε σχέση με την συνάρτηση καταλληλότητάς τους, τον πληθυσμό τους και το μέγεθος των γονιδίων τους. Για την βελτιστοποίηση της ομοιομορφίας δοκιμάσαμε δύο διαφορετικές συναρτήσεις καταλληλότητας, δύο διαφορετικές τεχνικές. Η πρώτη συνάρτηση έθετε κατώφλι στην τιμή της διασποράς και αναζητούσαμε λύσεις οι οποίες είχαν τιμές κάτω από το κατώφλι αυτό. Αντιθέτων στην δεύτερη συνάρτηση, δεν υπήρχε κάποιο κατώφλι, αλλά μία μετρική, και αναζητούσαμε λύσεις που είχαν όσο το δυνατόν μεγαλύτερη τιμή μετρικής. Συγκρίνοντας τις δύο αυτές μεθόδους καταλήξαμε στο συμπέρασμα ότι η δεύτερη συνάρτηση καταλληλότητας, χωρίς κατώφλι, ήταν αυτή που έδινε καλύτερα αποτελέσματα. Επειδή ταυτόχρονα θέλαμε το μείγμα μας να έχει και την μεγαλύτερη δυνατή κάλυψη, συνδυάσαμε αυτήν τη συνάρτηση με την συνάρτηση μεγιστοποίησης της έκφρασης και έτσι καταλήξαμε στην “υβριδική” συνάρτηση καταλληλότητας, την οποία και χρησιμοποιήσαμε για τα τελικά μας αποτελέσματα.

Το πρόβλημα το οποίο κληθήκαμε να βελτιστοποιήσουμε ανήκει στην κατηγορία της πολυκριτηριακής βελτιστοποίησης (multiobjective optimization), όπου έχουμε δύο κριτήρια βελτιστοποίησης τα οποία συγκρούονται μεταξύ τους, καθώς η αύξηση του ποσοστού κάλυψης συνεπάγεται σε μείωση της ομοιομορφίας και αντίστροφα. Η φύση αυτή του προβλήματος μας ώθησε στην χρήση Γενετικών Αλγορίθμων, οι οποίοι είναι ιδιαίτερα δημοφιλής για την λύση τέτοιου είδους προβλημάτων. Συνοψίζοντας τις διαφοροποιήσεις των Γενετικών Αλγορίθμων που χρησιμοποιήσαμε σε σχέση με την κοινή πρακτική μπορούμε να πούμε ότι είναι οι εξής:

- Χρήση άμεσης δεκαδικής κωδικοποίησης στην αναπαράσταση των υποψηφίων λύσεων
- Πεδίο τιμών των Γενετικών Αλγορίθμων διατάξεις
- Τροποποίηση των γενετικών τελεστών και προσαρμογή τους στις ιδιαιτερότητες του προβλήματος
- Σπονδυλωτή εφαρμογή (modular implementation) του αλγορίθμου, ώστε να παραμετροποιείται εξωτερικά για την άμεση προσαρμογή του και σε διαφορετικά δεδομένα

Το επόμενο βήμα ήταν ο καθορισμός των παραμέτρων του αλγορίθμου σε σχέση με τον πληθυσμό της γενιάς. Εκτελέσαμε μία σειρά ΓΑ με πληθυσμό ανά γενιά 20 n-άδων και στη συνέχεια 100 n-άδων. Οι γενεές παράγονταν με τις ίδιες ακριβώς διαδικασίες και το διαφορετικό στοιχείο αυτών ήταν το

μέγεθος του πληθυσμού και κατ' επέκταση το ποσοστό κάθε γενιάς που παραγόταν από τις διαδικασίες αυτές (τυχαία επιλογή, μετάλλαξη, διασταύρωση, carry-over). Από τα αποτελέσματα που πήραμε καταλήξαμε στο συμπέρασμα ότι οι ΓΑ με μεγάλο πληθυσμό γενιάς έχουν καλύτερη συμπεριφορά, επειδή η σύγκλισή τους γίνεται σε μικρότερο αριθμό γενεών (κριτήριο τερματισμού) και συμπεριφέρονται με τρόπο σχεδόν ντετερμινιστικό, αφού η πλειοψηφία των διακοσίων αυτών ΓΑ "συμφωνούσε" στην καλύτερη 20άδα κυτταρικών σειρών.

Έχοντας πλέον "συντονίσει" κατάλληλα τους ΓΑ τόσο σε σχέση με την συνάρτηση καταλληλότητας όσο και σε σχέση με την παραγωγή και τον πληθυσμό των γενεών, τα πειράματα που ακολούθησαν είχαν σαν σκοπό να μας δώσουν το βέλτιστο μέγεθος το υποψήφιου μείγματος. Δοκιμάσαμε μέγεθος μείγματος αποτελούμενο από 10 έως 30 κυτταρικές σειρές. Από τα αποτελέσματα που πήραμε διαπιστώσαμε ότι το καλύτερο μείγμα, αυτό δηλαδή που έχει υψηλό ποσοστό ομοιομορφίας και υψηλές τιμές έκφρασης, ενώ ταυτόχρονα δεν έρχεται σε αντίθεση με την υπάρχουσα βιβλιογραφία, αποτελείται από 20 κυτταρικές σειρές. Εδώ πρέπει να αναφέρουμε ότι μείγμα 30 κυτταρικών σειρών είναι πάρα πολύ μεγάλο, όπως αναφέρουν σχετικές μελέτες, και τα αποτελέσματά μας συμφωνούν, αφού τόσο μεγάλο μείγμα δεν βελτιώνει σημαντικά την έκφραση ενώ ταυτόχρονα δεν έχει υψηλή τιμή ομοιομορφίας, όπως το δείγμα των 20 κυτταρικών σειρών.

Μέσα από όλα τα πειράματα και τις διαδικασίες που περιγράψαμε σε προηγούμενα κεφάλαια, καταλήξαμε σε ένα μείγμα που αποτελείται από 20 κυτταρικές σειρές. Οι κυτταρικές σειρές που συμμετέχουν σε αυτό το μείγμα είναι συνολικά οι:

1 5 6 8 10 14 20 22 24 42 43 49 54 56 57 58 65 66 70 72 73 76 80

Ο λόγος για τον οποίο οι κυτταρικές σειρές που προτείνονται είναι περισσότερες από 20, είναι ότι οι ΓΑ, ακριβώς επειδή δεν λειτουργούν ντετερμινιστικά, μπορεί να οδηγήσουν σε διαφορετικές λύσεις. Και στους 200, όμως, ΓΑ που τρέξαμε με μέγεθος γονιδίου 20, καταλήξαμε στις παραπάνω κυτταρικές σειρές. Συνολικά για κάθε πείραμα οι προτεινόμενες λύσεις είναι στο Παράρτημα.

Η κάλυψη του μείγματος που επιτύχαμε ήταν της τάξης του 82%, μετά την απόρριψη των γονιδίων εκείνων που δεν μπορούν να εκφραστούν σε καμία από τις 83 κυτταρικές σειρές. Κάνοντας αναγωγή του ποσοστού αυτού στα αρχικά μας δεδομένα, το ποσοστό κάλυψης είναι της τάξης του 62.77% και η μέση τιμή διασποράς σε αρκετά χαμηλό βαθμό, δηλαδή 0.4431.

Συνοπτικά, τα αποτελέσματα τα οποία είχαμε ήταν τα εξής :

1. Για τους Γενετικούς Αλγορίθμους :
   - Χρήση ΓΑ με μεγάλους πληθυσμούς ανά γενιά
   - Κριτήριο τερματισμού στις 300 γενεές
   - Συνάρτηση καταλληλότητας για την μεγιστοποίηση της ομοιομορφίας χωρίς κατώφλι
2. Για το υποψήφιο μείγμα :
   - Βέλτιστο μείγμα αποτελούμενο από 20 κυτταρικές σειρές
   - Μέση τιμή διασποράς βέλτιστου μείγματος 0.4431
   - Μέση τιμή έκφρασης γονιδίων στο βέλτιστο μείγμα 1.8637e+03
   - Μέσο ποσοστό κάλυψης δεδομένων μετά την αφαίρεση των μη-εκφραζόμενων σε καμία κυτταρική σειρά γονιδίων 82.2%
   - Μέσο ποσοστό κάλυψης στο σύνολο των δεδομένων 62.76%

Στο σημείο αυτό μπορούμε να πούμε ότι τα αποτελέσματα που προέκυψαν από την εργασία αυτή είναι ικανοποιητικά. Παρόλα αυτά, αποτελούν μία πρόταση. Θα ήταν επομένως χρήσιμο να χρησιμοποιηθούν σε μία πειραματική διάταξη στα εργαστήρια Μοριακής Βιολογίας, ώστε να δούμε την πραγματική τους απόδοση πάνω στις μικροσυστοιχίες. Τοποθετώντας τα μείγματα αυτά στις μικροσυστοιχίες και εν συνεχεία υβριδοποιώντας τα με τα γονίδια που βρίσκονται διάστικτα πάνω σε αυτές, λαμβάνουμε την απαραίτητη πληροφορία για τα γονίδια αυτά. Εάν τα γονίδια είναι πλήρως ευδιάκριτα και μετά την υβριδοποίηση και έχουν κοινές τιμές έντασης, τότε το μείγμα είναι ομοιόμορφο και κατάλληλο για χρήση. Σε αντίθετη περίπτωση, μπορούμε μέσα από τα δεδομένα που θα λάβουμε από την πειραματική διαδικασία αυτή, να δούμε που τελικά υπήρξε πρόβλημα και να ρυθμίσουμε κατάλληλα τον Γενετικό Αλγόριθμο, με τέτοιο τρόπο ώστε να βελτιώσουμε την πρότασή μας.

Ο ΓΑ που υλοποιήσαμε για την εργασία αυτή μπορεί να χρησιμοποιηθεί και για μελέτη περαιτέρω δεδομένων. Η χρήση του σε datasets με διαφορετικό μέγεθος, δηλαδή με περισσότερες ή λιγότερες κυτταρικές σειρές, μπορεί να γίνει άμεσα και το μόνο που ίσως χρειαστεί να αλλάξει είναι οι ρυθμίσεις

του ΓΑ σε σχέση με το μέγεθος του “γονιδίου”, του κριτηρίου τερματισμού, του πληθυσμού της γενιάς και του τρόπου σχηματισμού της, δηλαδή πόσα μέλη στον νέο πληθυσμό θα παράγονται από την κάθε μέθοδο. Ο αριθμός των γονιδίων στις κυτταρικές σειρές επηρεάζει μόνο σε ότι αφορά τον χρόνο εκτέλεσης και δεν χρειάζεται καμία απολύτων αλλαγή για την εφαρμογή του σε δεδομένα με περισσότερα γονίδια.

Ο αλγόριθμός μας την στιγμή αυτή έχει σαν παραμέτρους στην συνάρτηση καταλληλότητας την ομοιομορφία και την ένταση. Μία επέκταση η οποία θα μπορούσε να υλοποιηθεί θα ήταν και η εισαγωγή του μεγέθους του μείγματος ως επιπλέον παράγοντα. Δηλαδή, αντί να ορίζεται παραμετρικά εξωτερικά το μέγεθος, να γίνεται αυτόματα από την συνάρτηση καταλληλότητας. Πιο συγκεκριμένα, θα μπορούσε να γίνει μία υλοποίηση, η οποία θα υπολόγιζε την καταλληλότητα της υποψήφιας λύσης με ένα δεδομένο μέγεθος καθώς και την καταλληλότητα λύσεων κοντά στο μέγεθος αυτό. Δηλαδή αν το μέγεθος ήταν 20 κυτταρικές σειρές, ο αλγόριθμος θα εξέταζε στον ίδιο κύκλο υποψήφιες λύσεις 18, 19 και 21 σειρών και θα επέλεγε τις πιο κατάλληλες από αυτές, ανεξαρτήτως μεγέθους, για τον σχηματισμό της επόμενης γενιάς.

Μένοντας στο ίδιο πεδίο έρευνας, δηλαδή στον Εξελικτικό Προγραμματισμό (Evolutionary Computing) και στον κύριο εκφραστή του, τους Γενετικούς Αλγορίθμους, μία ακόμα πιθανή επέκταση θα ήταν η δημιουργία “χωριών” ΓΑ που συναποφασίζουν. Δηλαδή, δύο ή και παραπάνω ΓΑ με παρόμοιους αλλά όχι ίδιους τρόπους παραγωγής γενεών, οι οποίοι καθώς εξελίσσονται ανά τακτά διαστήματα “παντρεύουν” τα αποτελέσματά τους για την παραγωγή νέων γενεών.

Βιβλιογραφία

[1] Αικατερίνη Γ. Περδικούρη & Αθανάσιος Κ. Τσακαλίδης, “Εισαγωγή στην Βιοπληροφορική”, Μάρτιος 2004, Τμήμα Μηχανικών Ηλεκτρονικών Υπολογιστών και Πληροφορικής, Πανεπιστήμιο Πατρών

[2] Haris Vikalo, Babak Hassibi & Arjang Hassibi, “A Statistical Model for Microarrays, Optimal Estimation Algorithms, and Limits of Performance”, IEEE Transactions on Signal Processing, Vol.54, No 6, June 2006

[3] Τσακανίκας Παναγιώτης, “Αναγνώριση Γονιδιακών Εκφράσεων Νεοπλασιών σε Microarrays”, Ιούνιος 2005, Τμήμα Μηχανικών Ηλεκτρονικών Υπολογιστών και Πληροφορικής, Πανεπιστήμιο Πατρών

[4] Gordon K. Smyth, Joelle Michaud & Hamish S. Scott, “Use of within-array replicate spots for assesing differential expression in microarray experiments”, December 2004, Bioinformatics Vol. 21 no. 9 2005, pages 2067-2075

[5] Gordon K. Smyth & Terry Speed, “Normalization of cDNA Microarray Data”, December 2003, Methods: Selecting Candidate Genes from DNA Array Screens: Application to Neuroscience, 31(4):265-73

[6] Yee Hwa Yang, Sandrine Dudoit, Percy Luu & Terence P. Speed, “Normalization for cDNA Microarray Data”, Nucleic Acids Res. 2002 Feb 15;30(4):e15

[7] John Quackenbush, “Microarray Data Normalization and Transformation”, December 2002, Nature Genetics Supplements, volume 32

[8] Vitoantonio Bevilaqua, Giusepe Mastronardi, Filippo Menolascina, Angelo Paradiso & Stefania Tommasi, “Genetic Algorithms and Artificial Neural Networks in Microarray Data Analysis: a Distributed Approach”, November 2006, Engineering Letters, 13:3, EL_13_3_14

[9] Natalia Novoradovskya, Michael L Whitfield, Lee S Basehore, Alexey Novoradovsky, Robert Pesich, Jerry Usary, Mehmet Karaca, Winston K Wong, Olga Aprelikova, Michael Fero, Charles M Perou, David Botstein & Jeff Braman, “Universal Reference RNA as a standard for microarray experiments”, March 2004, BMC Genomics

[10] Charles M. Perou, Therese Sorlie, Michael B. Eisen, Matt van de Rijn, Stefanie S. Jeffrey, Christian A. Rees, Jonathan R. Pollack, Douglas T. Ross, Hilde Johnsen, Lars A. Akslen, Oystein Fluge, Alexander Pergamenschikov, Cheryl Williams, Shirley X. Zhu, Per E. Lonning, Anne-Lise Borresen-Dale, Patrick O. Brown, David Bostein, “Molecular portraits of human breast tumors”, Nature 406, 747-752, 17 August 2000

[11] Maureen Cronin, Krishna Ghosh, Frank Sistare, John Quackenbush, Vincent Vilker & Catherine O'Connell, “Universal RNA Reference Materials for Gene Expression”, Clinical Chemistry 50, No 8, 2004

[12] Ivana V Yang, Emily Chen, Jeremy P Hasseman, Wei Liang, Bryan C Frank, Shuibang Wang, Vasily Sharov, Alexander I Saeed, Joseph White, Jerry Li, Norman H Lee, Timothy J Yeatman & John Quackenbush, “Within the fold: assessing differential expression measures and reproducibility in microarray assays”, Genone Biology, Vol. 3, No II, 2002

[13] Thomas Weise, “Global Optimization Algoritms – Theory and Application”,e-book 2nd edition, Nature Inspired Computation and Applications Laboratory (NICAL), University of Science and Technology of China, 26-06-09

[14] Whitley Darrel, “A Genetic Algorithm Tutorial”, Computer Science Department, Colorado State University, Statistics and Computing (1994) 4, 65-85

[15] Olivier de Weck, “Heuristic Techniques: A Basic Introduction to Genetic Algorithms”, March 2004, Lecture 11, Massachausetts Institute of Technology, Engineering Systems Divisionand Dept. Of Aeronautics and Astronautics

[16] Ι. Βλαχαβάς, Π. Κεφαλάς, Ν. Βασιλειάδης, Φ. Κόκκορας, Η. Σακελλαρίου, “Τεχνητή Νοημοσύνη”, Γ' Έκδοση, ISBN: 960-387-431-0, Εκδόσεις Β. Γκιούρδας Εκδοτική, 2006

[17] Ιωάννης Γ. Τσούλιος, “Γενετικοί Αλγόριθμοι”, Σεπτέμβριος 2009, Πανεπιστήμιο Ιωαννίνων, Τμήμα Πληροφορικής

[18] Andrew D. Martin, Kevin M. Quinn, “A Review Of Discrete Optimization Algorithms”, 1996, The Political Methodologist, vol. 7, no. 2

[19] Uma T. Shankavaram, William C. Reinhold, Satoshi Nishizuka, Sylvia Major, Daisaku Morita, Krishna K. Chary, Mark A. Remers, Uwe Scherf, Ari Kahn, Douglas Dolginow, Jeffrey Cossman, Eric P. Kaldjian, Dominic A. Scudiero, Emanuel Petricoin, Lance Liotta, Jae K. Lee, John N. Weinstein, “Transcript and protein expression profiles of the NCI-60 cancer cell panel: an integromic microarray study”, Molecular Cancer Therapeutics, 2007;6(3):820–32

[20] Dan Simon “Gradient Descent Optimization of Fuzzy Functions”, May 2007, Innovatia

[21] Μυλωνά Ειρήνη, “Επισκόπηση Σύγχρονων Μεθόδων για την Βελτιστοποίηση Μη Γραμμικών Συναρτήσεων”, Μάιος 2008, Τμήμα Μαθηματικών, Πανεπιστήμιο Πατρών

[22] Vittorio Maniezzo, Luca Maria Gambardella, and Fabio de Luigi (2004) ”Ant Colony Optimization”, New Optimization Techniques in Engineering, by Onwubolu, G. C., and B. V. Babu, Springer-Verlag Berlin Heidelberg, page 101-117.

[23]G. K. Purusothama, L. Jenkins, “Simulated annealing with local search – a hybrid algorithm for unit commitment”, IEEE Transactions on Power Systems, Vol. 18, No. 1, 2003.

[24]Georgia Tsiliki, Sofa Kaforou, Manouela Kapsetaki, Dimitris Kafetzopoulos, “A computational approach to microarray universal reference”, Bioinformatics and BioEngineering, 2008. BIBE 2008. 8th IEEE International Conference

[25] http://en.wikipedia.org/wiki/DNA_microarray

[26] http://en.wikipedia.org/wiki/Optimization

[27] http://en.wikipedia.org/wiki/Deterministic_algorithm

[28] http://en.wikipedia.org/wiki/Probabilistic_algorithms

Παράρτημα

Συγκεντρωτικά αποτελέσματα ΓΑ με κατώφλι

Οι παρακάτω πίνακες δείχνουν για τους ΓΑ με κατώφλι στην διασπορά στην πρώτη στήλη τον αριθμό του πειράματος, στην δεύτερη στήλη το κατώφλι που δοκιμάστηκε, στην τρίτη στήλη την μέση τιμή της διασποράς που πέτυχαν οι ΓΑ στα 6 πειράματα που έγιναν για το αντίστοιχο κατώφλι και την τέταρτη στήλη την μέση τιμή της συνάρτησης καταλληλότητας για τα 6 πειράματα ανά κατώφλι.

ΑΑ	Ραλιαρλ Ομομο μμίας	Μβοηψμή διοοηρόίς	Μβοηψμή επιχίας
1	0.00240	0.52600	0.03644
2	0.00540	0.52400	0.04949
3	0.00960	0.52040	0.05517
4	0.01500	0.51632	0.07535
5	0.02160	0.50637	0.09240
6	0.02940	0.49751	0.10642
7	0.03840	0.47816	0.12465
8	0.04860	0.49080	0.13653
9	0.06000	0.48360	0.15310
10	0.07260	0.46578	0.17033
11	0.08640	0.46386	0.18614
12	0.10140	0.46514	0.20421
13	0.11760	0.46520	0.22529
14	0.13500	0.45259	0.24940
15	0.15360	0.44150	0.27697
16	0.17340	0.43441	0.30703
17	0.19440	0.44664	0.32856
18	0.21660	0.44326	0.36152
19	0.24000	0.43843	0.39886
20	0.26460	0.43908	0.42819
21	0.29040	0.43967	0.46888
22	0.31740	0.44347	0.48520
23	0.34560	0.43629	0.51957
24	0.37500	0.43431	0.54609
25	0.40560	0.44061	0.56752
26	0.43740	0.43994	0.59125
27	0.47040	0.43070	0.62182
28	0.50460	0.43732	0.63811
29	0.54000	0.43469	0.66049
30	0.57660	0.43759	0.67644
31	0.61440	0.43252	0.69867
32	0.65340	0.43637	0.70982
33	0.69360	0.43466	0.72729
34	0.73500	0.44020	0.73866
35	0.77760	0.43637	0.75328
36	0.82140	0.43286	0.76854
37	0.86640	0.43664	0.77369
38	0.91260	0.43241	0.78791
39	0.96000	0.43399	0.79806
40	1.00860	0.43668	0.80751
41	1.05840	0.43700	0.81661
42	1.10940	0.44241	0.82191
43	1.16160	0.44116	0.83056
44	1.21500	0.44194	0.83660
45	1.26960	0.44617	0.84396
46	1.32540	0.44288	0.85149
47	1.38240	0.44348	0.85772
48	1.44060	0.44257	0.86198
49	1.50000	0.44610	0.86730
50	1.56060	0.44542	0.87349
51	1.62240	0.44586	0.87663
52	1.68540	0.44353	0.88384
53	1.74960	0.44667	0.88986
54	1.81500	0.44709	0.89016
55	1.88160	0.44655	0.89433
56	1.94940	0.45090	0.89741
57	2.01840	0.44653	0.90305
58	2.08860	0.45704	0.90459
59	2.16000	0.44414	0.91063
60	2.23260	0.45144	0.91346
61	2.30640	0.45390	0.91648
62	2.38140	0.44747	0.92001
63	2.45760	0.45357	0.92363
64	2.53500	0.45466	0.92800
65	2.61360	0.45197	0.92836
66	2.69340	0.45201	0.93152
67	2.77440	0.45706	0.93318
68	2.85660	0.46668	0.93492
69	2.94000	0.44634	0.93863

ΑΑ	Ραλιαρλ Ομομομρ φίας	Μβοηψμή διοοηρόίς	Μβοηψμή επιχίας
70	3.02460	0.45791	0.94114
71	3.11040	0.45545	0.94302
72	3.19740	0.47055	0.94374
73	3.28560	0.45403	0.94800
74	3.37500	0.45797	0.94861
75	3.46560	0.46522	0.95090
76	3.55740	0.47976	0.95213
77	3.65040	0.46640	0.95455
78	3.74460	0.47902	0.95584
79	3.84000	0.48079	0.95874
80	3.93660	0.47256	0.95988
81	4.03440	0.47100	0.96111
82	4.13340	0.47634	0.96263
83	4.23360	0.49057	0.96377
84	4.33500	0.46686	0.96611
85	4.43760	0.50250	0.96679
86	4.54140	0.51753	0.96801
87	4.64640	0.51911	0.96922
88	4.75260	0.54159	0.97031
89	4.86000	0.50785	0.97090
90	4.96860	0.49550	0.97189
91	5.07840	0.51053	0.97310
92	5.18940	0.53648	0.97414
93	5.30160	0.53170	0.97509
94	5.41500	0.55512	0.97623
95	5.52960	0.52502	0.97681
96	5.64540	0.54046	0.97836
97	5.76240	0.57500	0.97883
98	5.88060	0.56665	0.97959
99	6.00000	0.55868	0.98073
100	6.12060	0.57094	0.98127
101	6.24240	0.56360	0.98231
102	6.36540	0.57665	0.98336
103	6.48960	0.58070	0.98383
104	6.61500	0.57280	0.98465
105	6.74160	0.55877	0.98533
106	6.86940	0.57342	0.98620
107	6.99840	0.58929	0.98692
108	7.12860	0.57857	0.98737
109	7.26000	0.57861	0.98810
110	7.39260	0.58286	0.98871
111	7.52640	0.58749	0.98889
112	7.66140	0.59419	0.98969
113	7.79760	0.57749	0.99016
114	7.93500	0.59767	0.99065
115	8.07360	0.59310	0.99100
116	8.21340	0.59800	0.99160
117	8.35440	0.59961	0.99183
118	8.49660	0.59283	0.99241
119	8.64000	0.59737	0.99265
120	8.78460	0.60742	0.99318
121	8.93040	0.60229	0.99333
122	9.07740	0.59854	0.99391
123	9.22560	0.60905	0.99416
124	9.37500	0.60286	0.99460
125	9.52560	0.60346	0.99471
126	9.67740	0.60497	0.99517
127	9.83040	0.61999	0.99553
128	9.98460	0.62204	0.99564
129	10.14000	0.61785	0.99586
130	10.29700	0.62466	0.99617
131	10.45400	0.60523	0.99646
132	10.61300	0.61162	0.99672
133	10.77400	0.61668	0.99699
134	10.93500	0.61509	0.99707
135	11.09800	0.59801	0.99728
136	11.26100	0.63862	0.99737
137	11.42600	0.64800	0.99746
138	11.59300	0.59830	0.99775

ΑΑ	Ραλιαρλ Ομομομρ φίας	Μβοηψμή διοοηρόίς	Μβοηψμή επιχίας
138	11.59300	0.59830	0.99775
139	11.76000	0.64989	0.99769
140	11.92900	0.62141	0.99810
141	12.09800	0.65316	0.99798
142	12.26900	0.63667	0.99831
143	12.44200	0.60736	0.99839
144	12.61500	0.63462	0.99868
145	12.79000	0.63363	0.99850
146	12.96500	0.61719	0.99877
147	13.14200	0.64112	0.99877
148	13.32100	0.63337	0.99897
149	13.50000	0.64026	0.99895
150	13.68100	0.63979	0.99921
151	13.86200	0.62706	0.99920
152	14.04600	0.65494	0.99912
153	14.23000	0.64799	0.99923
154	14.41500	0.63152	0.99930
155	14.60200	0.64914	0.99932
156	14.78900	0.63089	0.99948
157	14.97800	0.66413	0.99955
158	15.16900	0.67648	0.99946
159	15.36000	0.65039	0.99946
160	15.55300	0.65099	0.99968
161	15.74600	0.67994	0.99964
162	15.94100	0.67354	0.99965
163	16.13800	0.67587	0.99971
164	16.33500	0.66739	0.99982
165	16.53400	0.68368	0.99980
166	16.73300	0.67428	0.99983
167	16.93400	0.66080	0.99987
168	17.13700	0.68566	0.99986
169	17.34000	0.68391	0.99990
170	17.54600	0.69134	0.99991
171	17.75000	0.67811	0.99990
172	17.95700	0.69443	0.99996
173	18.16600	0.69084	0.99997
174	18.37500	0.69718	0.99998
175	18.58600	0.70455	0.99998
176	18.79700	0.69708	1.000000
177	19.01000	0.69455	0.99999
178	19.22500	0.71873	0.99999
179	19.44000	0.69174	1.000000
180	19.65700	0.69228	1.000000
181	19.87400	0.69718	1.000000
182	20.09300	0.67941	1.000000
183	20.31400	0.68873	1.000000
184	20.53600	0.67826	1.000000
185	20.75800	0.68881	1.000000
186	20.98100	0.68865	1.000000
187	21.20600	0.70071	1.000000
188	21.43300	0.66424	1.000000
189	21.66000	0.66436	1.000000
190	21.88900	0.68238	1.000000
191	22.11800	0.66563	1.000000
192	22.34900	0.71346	1.000000
193	22.58200	0.68339	1.000000
194	22.81500	0.67986	1.000000
195	23.05000	0.68033	1.000000
196	23.28500	0.68089	1.000000
197	23.52200	0.68941	1.000000
198	23.76100	0.66287	1.000000
199	24.00000	0.72313	1.000000

Συγκεντρωτικά αποτελέσματα ΓΑ με "υβριδική" συνάρτηση καταλληλότητας

Το μέγεθος του γονιδίου των ΓΑ είναι 20 κυτταρικές σειρές.

ΑΑ	Διασπορά	Έκφραση	Ποσοστό έκφρασης
1	0.43914	1858.4	81.925
2	0.44408	1863.7	82.218
3	0.44132	1861.6	82.148
4	0.45007	1871.2	82.506
5	0.44180	1859.0	81.954
6	0.44132	1861.6	82.148
7	0.44132	1861.6	82.148
8	0.44132	1861.6	82.148
9	0.43914	1858.4	81.925
10	0.44442	1862.1	82.271
11	0.44132	1861.6	82.148
12	0.44132	1861.6	82.148
13	0.44625	1865.1	82.342
14	0.44132	1861.6	82.148
15	0.44132	1861.6	82.148
16	0.43797	1854.6	81.825
17	0.44132	1861.6	82.148
18	0.44132	1861.6	82.148
19	0.45004	1871.9	82.483
20	0.45267	1874.1	82.518
21	0.44132	1861.6	82.148
22	0.44467	1865.9	82.248
23	0.44132	1861.6	82.148
24	0.45004	1871.9	82.483
25	0.43914	1858.4	81.925
26	0.44467	1865.9	82.248
27	0.44132	1861.6	82.148
28	0.44132	1861.6	82.148
29	0.44467	1865.9	82.248
30	0.44467	1865.9	82.248
31	0.45256	1872.7	82.124
32	0.45007	1871.2	82.506
33	0.44467	1865.9	82.248
34	0.44132	1861.6	82.148
35	0.43914	1858.4	81.925
36	0.44132	1861.6	82.148
37	0.44408	1863.7	82.218
38	0.44132	1861.6	82.148
39	0.44132	1861.6	82.148
40	0.44132	1861.6	82.148
41	0.44467	1865.9	82.248
42	0.44132	1861.6	82.148
43	0.44583	1864.5	82.001
44	0.44467	1865.9	82.248
45	0.44132	1861.6	82.148
46	0.44132	1861.6	82.148
47	0.44132	1861.6	82.148
48	0.44132	1861.6	82.148
49	0.44132	1861.6	82.148
50	0.44132	1861.6	82.148
51	0.44132	1861.6	82.148
52	0.44467	1865.9	82.248
53	0.44132	1861.6	82.148
54	0.44132	1861.6	82.148
55	0.44132	1861.6	82.148
56	0.43914	1858.4	81.925
57	0.44096	1857.3	82.224
58	0.44132	1861.6	82.148
59	0.44132	1861.6	82.148
60	0.44055	1858.5	81.925
61	0.44132	1861.6	82.148
62	0.44132	1861.6	82.148
63	0.44132	1861.6	82.148
64	0.44132	1861.6	82.148
65	0.44132	1861.6	82.148
66	0.44132	1861.6	82.148
67	0.44132	1861.6	82.148
68	0.43914	1858.4	81.925
69	0.44132	1861.6	82.148
70	0.44467	1865.9	82.248
71	0.44132	1861.6	82.148
72	0.44132	1861.6	82.148
73	0.44491	1864.5	82.107
74	0.44132	1861.6	82.148
75	0.44467	1865.9	82.248
76	0.44132	1861.6	82.148
77	0.44132	1861.6	82.148
78	0.44408	1863.7	82.218
79	0.44132	1861.6	82.148
80	0.43914	1858.4	81.925
81	0.43914	1858.4	81.925
82	0.44132	1861.6	82.148
83	0.44132	1861.6	82.148
84	0.45314	1873.3	82.559
85	0.44132	1861.6	82.148
86	0.44132	1861.6	82.148
87	0.44467	1865.9	82.248
88	0.44467	1865.9	82.248
89	0.44132	1861.6	82.148
90	0.43914	1858.4	81.925
91	0.44467	1865.9	82.248
92	0.44132	1861.6	82.148
93	0.44132	1861.6	82.148
94	0.43914	1858.4	81.925
95	0.44132	1861.6	82.148
96	0.44132	1861.6	82.148
97	0.44132	1861.6	82.148
98	0.44320	1857.7	81.719
99	0.44467	1865.9	82.248
100	0.44132	1861.6	82.148
101	0.44467	1865.9	82.248
102	0.44132	1861.6	82.148
103	0.43914	1858.4	81.925
104	0.44132	1861.6	82.148
105	0.43914	1858.4	81.925
106	0.44408	1863.7	82.218
107	0.45434	1875.5	82.624
108	0.44132	1861.6	82.148
109	0.44132	1861.6	82.148
110	0.45007	1871.2	82.506
111	0.44132	1861.6	82.148
112	0.45366	1875.5	82.601
113	0.44132	1861.6	82.148
114	0.44132	1861.6	82.148
115	0.44132	1861.6	82.148
116	0.43914	1858.4	81.925
117	0.44132	1861.6	82.148
118	0.45004	1871.9	82.483
119	0.43914	1858.4	81.925
120	0.44467	1865.9	82.248
121	0.43914	1858.4	81.925
122	0.44467	1865.9	82.248
123	0.43914	1858.4	81.925
124	0.44132	1861.6	82.148
125	0.44467	1865.9	82.248
126	0.44132	1861.6	82.148
127	0.44132	1861.6	82.148
128	0.44132	1861.6	82.148
129	0.44182	1855.4	81.936
130	0.44132	1861.6	82.148
131	0.44132	1861.6	82.148
132	0.44132	1861.6	82.148
133	0.44132	1861.6	82.148
134	0.43914	1858.4	81.925
135	0.43914	1858.4	81.925
136	0.45004	1871.9	82.483
137	0.45004	1871.9	82.483
138	0.44132	1861.6	82.148
139	0.44935	1867.8	82.377
140	0.44132	1861.6	82.148
141	0.44132	1861.6	82.148
142	0.44132	1861.6	82.148
143	0.44132	1861.6	82.148
144	0.44132	1861.6	82.148
145	0.44408	1863.7	82.218
146	0.44442	1862.1	82.271
147	0.44132	1861.6	82.148
148	0.44583	1864.5	82.001
149	0.44467	1865.9	82.248
150	0.44467	1865.9	82.248
151	0.44132	1861.6	82.148
152	0.44132	1861.6	82.148
153	0.44132	1861.6	82.148
154	0.44132	1861.6	82.148
155	0.44132	1861.6	82.148
156	0.44467	1865.9	82.248
157	0.44132	1861.6	82.148
158	0.44132	1861.6	82.148
159	0.44132	1861.6	82.148
160	0.44467	1865.9	82.248
161	0.44408	1863.7	82.218
162	0.44132	1861.6	82.148
163	0.44132	1861.6	82.148
164	0.44132	1861.6	82.148
165	0.44467	1865.9	82.248
166	0.44132	1861.6	82.148
167	0.44132	1861.6	82.148
168	0.44132	1861.6	82.148
169	0.44132	1861.6	82.148
170	0.44132	1861.6	82.148
171	0.45004	1871.9	82.483
172	0.44132	1861.6	82.148
173	0.44132	1861.6	82.148
174	0.44467	1865.9	82.248
175	0.44132	1861.6	82.148
176	0.44703	1865.2	82.060
177	0.43914	1858.4	81.925
178	0.44132	1861.6	82.148
179	0.44132	1861.6	82.148
180	0.44132	1861.6	82.148
181	0.44467	1865.9	82.248
182	0.44132	1861.6	82.148
183	0.44132	1861.6	82.148
184	0.44132	1861.6	82.148
185	0.44132	1861.6	82.148
186	0.44132	1861.6	82.148
187	0.44132	1861.6	82.148
188	0.45057	1869.5	82.454
189	0.44467	1865.9	82.248
190	0.44132	1861.6	82.148
191	0.44467	1865.9	82.248
192	0.44467	1865.9	82.248
193	0.44132	1861.6	82.148
194	0.44132	1861.6	82.148
195	0.44132	1861.6	82.148
196	0.44132	1861.6	82.148
197	0.44467	1865.9	82.248
198	0.44132	1861.6	82.148
199	0.44132	1861.6	82.148
200	0.45007	1871.2	82.506

Συγκεντρωτικά αποτελέσματα ΓΑ με "υβριδική" συνάρτηση καταλληλότητας και διαφορετικό μέγεθος n-άδας

ΑΑ	Μέγεθος n-άδας	Διασπορά	Έκφραση	Εκφραζόμενων Γονιδίων
1	10	0.34771	1862.7	77.0870
2		0.34771	1862.7	77.0870
3		0.36184	1873.3	77.3570
4		0.34771	1862.7	77.0870
5		0.34771	1862.7	77.0870
6	11	0.36400	1865.0	77.8220
7		0.36400	1865.0	77.8220
8		0.36400	1865.0	77.8220
9		0.36400	1865.0	77.8220
10		0.36400	1865.0	77.8220
11	12	0.38251	1876.0	78.3680
12		0.38251	1876.0	78.3680
13		0.38251	1876.0	78.3680
14		0.38455	1875.6	78.4620
15		0.38251	1876.0	78.3680
16	13	0.39627	1881.3	78.8970
17		0.39627	1881.3	78.8970
18		0.39627	1881.3	78.8970
19		0.39627	1881.3	78.8970
20		0.39627	1881.3	78.8970
21	14	0.40600	1881.7	79.4380
22		0.40600	1881.7	79.4380
23		0.40600	1881.7	79.4380
24		0.40600	1881.7	79.4380
25		0.40600	1881.7	79.4380
26	15	0.41354	1877.4	80.0020
27		0.41354	1877.4	80.0020
28		0.41354	1877.4	80.0020
29		0.41354	1877.4	80.0020
30		0.41354	1877.4	80.0020
31	16	0.41840	1872.6	80.6960
32		0.41840	1872.6	80.6960
33		0.41840	1872.6	80.6960
34		0.41840	1872.6	80.6960
35		0.41840	1872.6	80.6960
36	17	0.42343	1867.6	80.9190
37		0.42343	1867.6	80.9190
38		0.42343	1867.6	80.9190
39		0.42343	1867.6	80.9190
40		0.42343	1867.6	80.9190
41	18	0.42775	1863.2	81.1660
42		0.42775	1863.2	81.1660
43		0.42775	1863.2	81.1660
44		0.42775	1863.2	81.1660
45		0.42775	1863.2	81.1660
46	19	0.43137	1858.4	81.4540
47		0.43137	1858.4	81.4540
48		0.43137	1858.4	81.4540
49		0.43137	1858.4	81.4540
50		0.43137	1858.4	81.4540
51	20	0.44132	1861.6	82.1480
52		0.44132	1861.6	82.1480
53		0.44132	1861.6	82.1480

ΑΑ	Μέγεθος n-άδας	Διασπορά	Έκφραση	Ποσοστό Έκφρασης
54		0.44132	1861.6	82.1480
55		0.45004	1871.9	82.4830
56	21	0.44974	1861.5	82.5710
57		0.44710	1861.5	82.4890
58		0.44710	1861.5	82.4890
59		0.44710	1861.5	82.4890
60		0.44710	1861.5	82.4890
61	22	0.46551	1870.4	82.9830
62		0.46560	1872.3	83.2710
63		0.45588	1859.9	82.7710
64		0.45741	1866.8	82.9830
65		0.45187	1857.2	82.7770
66	23	0.46135	1861.5	83.3290
67		0.46572	1863.2	83.5350
68		0.46187	1862.6	83.2650
69		0.46187	1862.6	83.2650
70		0.46782	1869.9	83.5120
71	24	0.48029	1872.6	84.0580
72		0.47270	1867.6	83.8110
73		0.46515	1857.0	83.5650
74		0.47280	1868.4	83.6590
75		0.47365	1866.2	83.9290
76	25	0.48081	1865.9	84.1580
77		0.48209	1870.2	84.2640
78		0.48072	1868.6	84.3400
79		0.48285	1868.4	84.2930
80		0.46838	1852.5	83.8530
81	26	0.48469	1866.2	84.5230
82		0.48086	1861.0	84.3700
83		0.48414	1865.6	84.4700
84		0.48904	1870.3	84.5760
85		0.48123	1865.5	84.4460
86	27	0.48374	1857.0	84.6230
87		0.48983	1861.3	84.8400
88		0.47882	1849.3	84.5170
89		0.48885	1862.4	84.7580
90		0.47243	1841.5	84.5230
91	28	0.49624	1863.2	85.2100
92		0.49524	1865.8	85.1630
93		0.48227	1843.8	84.6930
94		0.48049	1844.0	84.8750
95		0.49009	1862.0	84.9580
96	29	0.50296	1862.7	85.4460
97		0.50684	1867.9	85.3220
98		0.49858	1861.9	85.3400
99		0.49685	1857.9	85.2100
100		0.50583	1870.8	85.3630
101	30	0.50160	1858.2	85.5690
102		0.50869	1864.1	85.7920
103		0.50730	1864.5	85.8570
104		0.50641	1861.9	85.7920
105		0.49652	1846.8	85.3220

Προτεινόμενα μείγματα αναφοράς 200 ΓΑ με μέγεθος 20 κυτταρικές σειρές

Αρ. Πειράματος																				
1	1	5	6	8	10	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
2	1	5	6	8	10	14	24	28	42	43	49	54	56	58	65	66	70	72	76	80
3	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
4	1	5	8	14	20	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
5	1	5	8	10	22	24	28	42	43	49	54	56	57	58	65	66	70	73	76	80
6	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
7	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
8	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
9	1	5	6	8	10	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
10	1	5	8	10	14	22	24	28	42	43	49	54	56	57	58	65	66	70	76	80
11	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
12	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
13	1	6	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
14	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
15	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
16	1	5	8	10	22	24	28	42	43	49	54	56	57	58	65	66	70	72	76	80
17	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
18	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
19	1	5	8	10	14	20	22	24	28	42	43	49	54	56	58	65	66	70	76	80
20	1	5	6	8	10	14	20	24	28	42	43	49	54	56	58	65	66	70	76	80
21	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
22	1	5	6	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	76	80
23	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
24	1	5	8	10	14	20	22	24	28	42	43	49	54	56	58	65	66	70	76	80
25	1	5	6	8	10	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
26	1	5	6	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	76	80
27	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
28	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
29	1	5	6	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	76	80
30	1	5	6	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	76	80
31	1	5	6	8	10	14	20	24	28	42	49	54	56	58	65	66	70	72	76	80
32	1	5	8	14	20	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
33	1	5	6	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	76	80
34	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
35	1	5	6	8	10	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
36	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
37	1	5	6	8	10	14	24	28	42	43	49	54	56	58	65	66	70	72	76	80
38	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
39	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
40	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
41	1	5	6	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	76	80
42	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
43	1	5	6	8	10	14	22	24	28	42	49	54	56	58	65	66	70	72	76	80
44	1	5	6	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	76	80
45	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
46	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
47	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
48	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
49	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
50	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
51	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
52	1	5	6	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	76	80
53	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
54	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
55	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
56	1	5	6	8	10	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
57	1	5	6	8	10	14	22	24	28	42	43	49	54	56	58	66	70	72	76	80
58	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
59	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
60	1	5	8	10	22	24	28	42	43	49	54	56	58	65	66	70	72	73	76	80

61	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
62	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
63	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
64	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
65	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
66	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
67	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
68	1	5	6	8	10	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
69	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
70	1	5	6	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	76	80
71	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
72	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
73	1	5	8	10	20	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
74	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
75	1	5	6	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	76	80
76	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
77	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
78	1	5	6	8	10	14	24	28	42	43	49	54	56	58	65	66	70	72	76	80
79	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
80	1	5	6	8	10	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
81	1	5	6	8	10	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
82	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
83	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
84	1	5	6	8	14	20	24	28	42	43	49	54	56	58	65	66	70	72	76	80
85	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
86	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
87	1	5	6	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	76	80
88	1	5	6	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	76	80
89	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
90	1	5	6	8	10	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
91	1	5	6	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	76	80
92	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
93	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
94	1	5	6	8	10	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
95	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
96	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
97	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
98	1	5	8	10	22	24	28	42	49	54	56	57	58	65	66	70	72	73	76	80
99	1	5	6	8	2	6	7	9	3	42	43	49	54	56	58	65	66	70	76	80
100	1	5	8	10	2	6	9	11	3	43	49	54	56	58	65	66	70	72	76	80
101	1	5	6	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	76	80
102	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
103	1	5	6	8	10	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
104	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
105	1	5	6	8	10	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
106	1	5	6	8	10	14	24	28	42	43	49	54	56	58	65	66	70	72	76	80
107	1	6	8	10	14	20	22	24	28	42	43	49	54	56	58	65	66	70	76	80
108	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
109	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
110	1	5	8	14	20	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
111	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
112	1	5	6	8	14	20	22	24	28	42	43	49	54	56	58	65	66	70	76	80
113	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
114	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
115	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
116	1	5	6	8	10	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
117	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
118	1	5	8	10	14	20	22	24	28	42	43	49	54	56	58	65	66	70	76	80
119	1	5	6	8	10	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
120	1	5	6	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	76	80

121	1	5	6	8	10	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
122	1	5	6	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	76	80
123	1	5	6	8	10	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
124	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
125	1	5	6	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	76	80
126	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
127	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
128	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
129	1	5	6	8	10	22	28	34	42	43	49	54	56	58	65	66	70	72	76	80
130	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
131	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
132	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
133	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
134	1	5	6	8	10	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
135	1	5	6	8	10	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
136	1	5	8	10	14	20	22	24	28	42	43	49	54	56	58	65	66	70	76	80
137	1	5	8	10	14	20	22	24	28	42	43	49	54	56	58	65	66	70	76	80
138	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
139	1	5	8	10	14	22	23	24	28	42	43	49	54	56	58	65	66	70	76	80
140	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
141	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
142	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
143	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
144	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
145	1	5	6	8	10	14	24	28	42	43	49	54	56	58	65	66	70	72	76	80
146	1	5	8	10	14	22	24	28	42	43	49	54	56	57	58	65	66	70	76	80
147	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
148	1	5	6	8	10	14	22	24	28	42	49	54	56	58	65	66	70	72	76	80
149	1	5	6	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	76	80
150	1	5	6	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	76	80
151	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
152	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
153	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
154	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
155	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
156	1	5	6	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	76	80
157	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
158	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
159	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
160	1	5	6	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	76	80
161	1	5	6	8	10	14	24	28	42	43	49	54	56	58	65	66	70	72	76	80
162	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
163	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
164	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
165	1	5	6	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	76	80
166	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
167	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
168	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
169	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
170	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
171	1	5	8	10	14	20	22	24	28	42	43	49	54	56	58	65	66	70	76	80
172	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
173	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
174	1	5	6	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	76	80
175	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
176	1	5	8	22	24	28	42	43	49	54	56	57	58	65	66	70	73	74	76	80
177	1	5	6	8	10	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
178	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
179	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
180	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80

181	1	5	6	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	76	80
182	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
183	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
184	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
185	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
186	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
187	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
188	1	5	6	8	14	22	24	28	42	43	49	54	56	58	65	66	70	73	76	80
189	1	5	6	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	76	80
190	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
191	1	5	6	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	76	80
192	1	5	6	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	76	80
193	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
194	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
195	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
196	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
197	1	5	6	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	76	80
198	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
199	1	5	8	10	14	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80
200	1	5	8	14	20	22	24	28	42	43	49	54	56	58	65	66	70	72	76	80

Ρουτίνες που προσαρμόζουν το gatool στο πρόβλημα

Ρουτίνα αρχικοποίησης (m_init.m)

```
clear;
load ../data/gatool_experiment.mat;
%Tropopoioume to dataset wste na periexei mono ta gonidia ta opoia
%ekfrazontai estw kai se mia kyttarikh seira
%b=experiment.full_data>=experiment.config.data_threshold;
b=experiment.full_data;
c=sum(b,2);
d=c>0;
experiment.tmp.data=experiment.full_data(d,:);
clear c;
clear d;
clear b;
if not(experiment.config.interactive)
   experiment.tmp.plot_fcns={};
else
   experiment.tmp.plot_fcns={ @gaplotbestf,@gaplotstopping};
end
```

```
experiment.tmp.plot_interval=1;

experiment.config.generations=5;

experiment.tmp.problem.options=gaoptimset('CreationFcn',@m_create_gene,'MutationFcn',@m_mutate,'CrossoverFcn',@
m_crossover_gene,'EliteCount',experiment.config.elite,'Generations',experiment.config.generations,'PopulationSize',experi
ment.config.population_size,'PopulationType','custom','FitnessLimit',-
Inf,'StallGenLimit',experiment.config.generations,'StallTimeLimit',Inf,'PlotFcns',experiment.tmp.plot_fcns,'PlotInterval',exp
eriment.tmp.plot_interval);

experiment.tmp.problem.nvars=experiment.config.gene_size;

experiment.tmp.problem.fitnessfcn=experiment.config.fitness;

global experiment

clc;
```

Ρουτίνα δημιουργίας γονιδίου Γενετικού Αλγορίθμου

```
function res=m_create_gene(gene_size,fitness,m_ga_params)

   %disp 'create_gene start'

   global experiment

   nb_gene=m_ga_params.PopulationSize;

   [height,width]=size(experiment.tmp.data);

   res=zeros(nb_gene,gene_size);

   for k=1:nb_gene

   gene=randperm(width);

   gene=gene(1:gene_size)';

   res(k,:)=sort(gene(:));

   end

   %disp 'create_gene done'

end
```

Ρουτίνα υλοποίησης γενετικού τελεστή μετάλλαξης

```
function res=m_mutate(parents,m_ga_params,gene_size,fitness,cur_state,cur_score,cur_population)

   %disp 'mutate start'

   global experiment

   [height, width]=size(experiment.tmp.data);

   res=zeros(length(parents),gene_size);

   count=1;

   for k=parents
```

```
    gene_compl=setxor(cur_population(k,:),1:width);
    gene_compl=gene_compl(1:gene_size);
    mutate=rand(size(cur_population(k,:)))<experiment.config.mutate_element_chance;
    res(count,:)=cur_population(k,:);
    res(count,mutate)=gene_compl(mutate);
    count=count+1;
    end
   % disp 'mutate done'
end
```

Ρουτίνα υλοποίησης γενετικού τελεστή διασταύρωσης

```
function res=m_crossover_gene(parents,m_ga_params,gene_size,fitness,unused,cur_population)
    %disp 'crossover start'
    res=zeros(numel(parents)/2,gene_size);
    for k=1:numel(parents)/2
    %gene2.PopInitRange
   % g_pos=parents(randperm(numel(parents)));
    gene1=cur_population(parents(2*k-1),:);
    gene2=cur_population(parents(2*k),:);
    tmp=unique([gene1,gene2]);
    gene=tmp(randperm(numel(tmp)));
    res(k,:)=gene(1:gene_size);
    end
    %disp 'crossover done'

   % gene=mod((cur_population+1),80)+1
end
```

Ρουτίνα για την εκκίνηση του ΓΑ

```
function [best_score,history]=run_ga(exper_name,ga_num)
clear experiment;
load ../data/gatool_experiment.mat

fname=sprintf('../work/%s/z_ga_config_%d.mat',exper_name,ga_num);
load(fname);
```

```
experiment.config=config;
%Tropopoioume to dataset wste na periexei mono ta gonidia ta opoia
%ekfrazontai estw kai se mia kyttarikh seira
b=experiment.full_data>=experiment.config.data_threshold;
b=experiment.full_data;
c=sum(b,2);
d=c>0;
experiment.tmp.data=experiment.full_data(d,:);
clear c;
clear d;
clear b;

if not(experiment.config.interactive)
   experiment.tmp.plot_fcns={};
else
   experiment.tmp.plot_fcns={ @gaplotbestf,@gaplotstopping};
end
experiment.tmp.plot_interval=1;
experiment.tmp.problem.options=gaoptimset('CreationFcn',@m_create_gene,'MutationFcn',@m_mutate,'CrossoverFcn',@
m_crossover_gene,'EliteCount',experiment.config.elite,'Generations',experiment.config.generations,'PopulationSize',experi
ment.config.population_size,'PopulationType','custom','FitnessLimit',-
Inf,'StallGenLimit',experiment.config.generations,'StallTimeLimit',Inf,'PlotFcns',experiment.tmp.plot_fcns,'PlotInterval',exp
eriment.tmp.plot_interval,'OutputFcns',@m_get_history);
experiment.tmp.problem.nvars=experiment.config.gene_size;
experiment.tmp.problem.fitnessfcn=experiment.config.fitness;
global experiment
ga(experiment.tmp.problem);
history=experiment.history;
best_score=history.best_genes_scores(end);
fname=sprintf('../work/%s/z_ga_results_%d.mat',exper_name,ga_num);
save(fname,'history','config')
finishedfname=sprintf('../work/%s/z_finished.mat',exper_name);
load(finishedfname);
finished(end+1)=ga_num;
todo=todo(todo~=ga_num);
save(finishedfname,'todo','finished');
```

Ρουτίνα για την καταγραφή του ιστορικού του ΓΑ

```
function[state, options,optchanged] = m_get_history(options,state,flag)
%GAOUTPUTFCNTEMPLATE Template to write custom OutputFcn for GA.
%   [STATE, OPTIONS, OPTCHANGED] = GAOUTPUTFCNTEMPLATE(OPTIONS,STATE,FLAG)
%   where OPTIONS is an options structure used by GA.
%
%   STATE: A structure containing the following information about the state
%   of the optimization:
%           Population: Population in the current generation
%                Score: Scores of the current population
%           Generation: Current generation number
%            StartTime: Time when GA started
%             StopFlag: String containing the reason for stopping
%            Selection: Indices of individuals selected for elite,
%                       crossover and mutation
%          Expectation: Expectation for selection of individuals
%                 Best: Vector containing the best score in each generation
%       LastImprovement: Generation at which the last improvement in
%                       fitness value occurred
%   LastImprovementTime: Time at which last improvement occurred
%   FLAG: Current state in which OutputFcn is called. Possible values are:
%        init: initialization state
%        iter: iteration state
%   interrupt: intermediate state
%        done: final state
%
%   STATE: Structure containing information about the state of the
%         optimization.
%   OPTCHANGED: Boolean indicating if the options have changed.
%          See also PATTERNSEARCH, GA, GAOPTIMSET
%   Copyright 2004-2005 The MathWorks, Inc.
%   $Revision: 1.1.6.4 $  $Date: 2006/11/11 22:42:34 $
global experiment
optchanged = false;
switch flag
```

```
 case 'init'
        experiment.history.generation=0;
        experiment.history.best_genes=zeros(experiment.config.generations,experiment.config.gene_size);
        experiment.history.all_genes=zeros(experiment.config.generations,experiment.config.population_size,experiment.confi
g.gene_size);
        experiment.history.best_genes_scores=zeros(experiment.config.generations,1);
        experiment.history.all_genes_scores=zeros(experiment.config.generations,experiment.config.population_size);

        experiment.history.generation=experiment.history.generation+1;
        experiment.history.all_genes(experiment.history.generation,:,:)=state.Population;
        experiment.history.all_genes_scores(experiment.history.generation,:)=state.Score(:);
        [best_score,best_pos]=min(state.Score);
        best_gene=state.Population(best_pos,:);
        experiment.history.best_genes_scores(experiment.history.generation)=best_score;
        experiment.history.best_genes(experiment.history.generation,:)=best_gene;

        disp(sprintf('Initializing GA: %s',experiment.config.name));
    case {'iter','interrupt'}

        experiment.history.generation=experiment.history.generation+1;
        experiment.history.all_genes(experiment.history.generation,:,:)=state.Population;
        experiment.history.all_genes_scores(experiment.history.generation,:)=state.Score(:);
        [best_score,best_pos]=min(state.Score);
        best_gene=state.Population(best_pos,:);
        experiment.history.best_genes_scores(experiment.history.generation)=best_score;
        experiment.history.best_genes(experiment.history.generation,:)=best_gene;
        disp(sprintf('Running GA: %s, Gen= %d',experiment.config.name,experiment.history.generation));
    case 'done'
        disp('Performing final task');
          experiment.history.generation=experiment.history.generation+1;
        experiment.history.all_genes(experiment.history.generation,:,:)=state.Population;
        experiment.history.all_genes_scores(experiment.history.generation,:)=state.Score(:);
        [best_score,best_pos]=min(state.Score);
        best_gene=state.Population(best_pos,:);
        experiment.history.best_genes_scores(experiment.history.generation)=best_score;
        experiment.history.best_genes(experiment.history.generation,:)=best_gene;
```

```
    disp(sprintf('Running GA: %s, Gen= %d',experiment.config.name,experiment.history.generation));
end
```

Ρουτίνα για την δημιουργία πειράματος

```
function m_create_experiment(experiment_name)
  experiment_folder=sprintf('../work/%s/',experiment_name);
  mkdir(experiment_folder);
  experiment_fnames={};
  load ../data/gatool_experiment.mat
  config=experiment.config;
  experiment
  for k=1:10
    experiment_fnames{k}=sprintf('%sz_ga_config_%d.mat',experiment_folder,k);
    config.generations=300;
    config.population_size=20;
    config.name=sprintf('test_by_gen %d',k);
    save(experiment_fnames{k},'config')
  end
  todo=1:10;
  finished=[];
  finishedfname=sprintf('../work/%s/z_finished.mat',experiment_name);
  save(finishedfname,'todo','finished');
```

Ρουτίνα για την επανεκκίνηση του πειράματος και την συνέχειά του μετά από βίαιη διακοπή

```
function m_resume_experiment(exper_name)
  experiment_folder=sprintf('../work/%s/',exper_name);
  files=dir(experiment_folder);
  configs={};
  config_count=0;
  configs={};
  config_count=0;
  finishedfname=sprintf('../work/%s/z_finished.mat',exper_name);
  load(finishedfname);
  experiment_fnames={};
  for k=todo
```

```
        %exper_name=sprintf('%sz_ga_config_%d.mat',experiment_folder,k);
%       load(experiment_fnames{k});
        [best_score history]=m_run_ga(exper_name,k);
 %      save(experiment_fnames{k});
    end
```

Συνάρτηση Καταλληλότητας

```
function res=m_fitness_expression(gene)
    %disp 'fitness start'
      global experiment
     [height, width]=size(experiment.tmp.data);
        tmp_data=double(experiment.tmp.data(:,gene)>=experiment.config.fitness_expression.expression_threshold);
                res=sum(tmp_data,2)>0;
                res=sum(res)/height;
        res=1-res;
        %disp(sprintf('height=%d',height));
       % disp 'fitness done'
end
```

Πηγαίος Κώδικας Matlab/Octave

Ρουτίνα αφαίρεσης γονιδίων που δεν περνάνε το κατώφλι έκφρασης

```
function k_init(expression_cutoff)
%Tropopoioume to dataset wste na periexei mono ta gonidia ta opoia
%ekfrazontai estw kai se mia kyttarikh seira
clear global data
global data
load ../data/globals.mat
if nargin==0
    expression_cutoff=100;
    keep=data>=expression_cutoff;
    g_keep=sum(keep,2);
    data=data(g_keep>0,:);
else
```

```
    keep=data>=expression_cutoff;
    g_keep=sum(keep,2);
    data=data(g_keep>0,:);
end
global d_height d_width
[d_height,d_width]=size(data);
end
```

Ρουτίνα υλοποίησης Γενετικού Αλγορίθμου

```
function [best_gene best_score full_history]=k_ga_core(genetic_params)
    clear global ga_params
    global ga_params
    ga_params=genetic_params;
    function nb_people=persons_in_generation()
        nb_people=genetic_params.best_children+genetic_params.new_in_generation+genetic_params.best_mutated+genetic_
params.best_carry_over;
    end
%paragwgi paidiou
    function child=produce_child(gene1,gene2)
        res=union(gene1,gene2);
        rnd_idx=randperm(numel(res));
        child=res(rnd_idx(1:genetic_params.gene_size));
        child=sort(child);
    end
%syn8etoume to kainourio gonidio kratwntas ola-mutation_chance kai
%pros8etontas mutation_chance stoixeia ap'to pedio timwn(kanonika 1:83)
    function new_gene=mutate(old_gene)
        new_values=setdiff(genetic_params.gene_allowed_values,old_gene);
        new_rnd_idx=randperm(numel(new_values));
        new_rnd_idx=new_rnd_idx(1:genetic_params.mutation_chance);
        old_rnd_idx=randperm(numel(old_gene));
        old_rnd_idx=old_rnd_idx(1:end-genetic_params.mutation_chance);
        new_gene=[old_gene(old_rnd_idx),new_values(new_rnd_idx)];
        new_gene=sort(new_gene);
    end
%xrhsimopoieitai gia paragwgi prwths genias h gia ta tyxaia atoma ka8e
```

```
%genias
  function new_gene=create_random_gene()
    tmp=randperm(numel(genetic_params.gene_allowed_values));
    tmp=genetic_params.gene_allowed_values(tmp(1:genetic_params.gene_size));
    new_gene=tmp;
    new_gene=sort(new_gene);
  end

  function [genes scores]=get_genes_sorted_by_fitnes(generation)
    generation.persons(1);
    generation.persons(1).gene;
    generation.persons(1).score;
    numel(generation.persons);
    %%pause
    scores=zeros([numel(generation.persons)]);
    genes=zeros([numel(generation.persons),genetic_params.gene_size]);
    for person_num=1:numel(generation.persons)
      scores(person_num)=generation.persons(person_num).score;
      genes(person_num,:)=generation.persons(person_num).gene;
    end
  end

  function new_generation=eliminate_duplicates(cur_generation)
    for iter=1:numel(cur_generation)
      cur_generation.persons(iter).gene=sort(cur_generation.persons(iter).gene);
    end
    new_generation(1).persons=cur_generation(1).persons;
    for iter=2:numel(cur_generation)
      person=cur_generation.persons(iter).gene;
      found_same=0;
      for k=1:numel(new_generation)
        found_same=found_same+(sum(person==new_generation.persons(k).gene));
      end
      while found_same>0
        person=create_random_gene;
```

```
            found_same=0;
            for k=1:numel(new_generation)
                found_same=found_same+(sum(person==new_generation.persons(k).gene));
            end
        end
        new_generation.persons(iter).gene=person;
    end
    new_generation.best_in_generation=[];
    new_generation.best_score_in_generation=-1;
    new_generation.best_in_history=[];
    new_generation.best_score_in_history=-1;
end


  function new_generation=produce_generation(old_generation)
    new_generation.best_in_generation=[];
    new_generation.best_score_in_generation=-1;
    new_generation.best_in_history=[];
    new_generation.best_score_in_history=-1;
    if nargin==1
        %is not the first generation
        [old_genes,old_scores]=get_genes_sorted_by_fitnes(old_generation);
        new_person_counter=1;
        for i=1:genetic_params.best_carry_over

            new_generation.persons(new_person_counter).gene=old_genes(i,:);
            new_person_counter=new_person_counter+1;
        end
        for i=1:genetic_params.best_children
            rnd_idx=randperm(genetic_params.elligible_parents);
             new_generation.persons(new_person_counter).gene=produce_child(old_genes(rnd_idx(1),:),old_genes(rnd_idx(2
),:));
            new_person_counter=new_person_counter+1;
        end

         for i=1:genetic_params.best_mutated
```

```
            new_generation.persons(new_person_counter).gene=mutate(old_genes(i,:));
            new_person_counter=new_person_counter+1;
        end
        for i=1:genetic_params.new_in_generation
            new_generation.persons(new_person_counter).gene=create_random_gene();
            new_person_counter=new_person_counter+1;
        end
        new_generation.persons;
        new_person_counter;
        new_generation.best_in_history=old_generation.best_in_history;
        new_generation.best_score_in_history=old_generation.best_score_in_history;
    else
        %paragwgi prwths genias
        for i=1:persons_in_generation()
            new_generation.persons(i).gene=create_random_gene();
        end
    end
   % new_generation=eliminate_duplicates(new_generation);
end

function eval_generation=evaluate(generation)
    for i=1:numel(generation.persons)
        generation.persons(i).score=rand;
        generation.persons(i).score=genetic_params.fitness(generation.persons(i).gene);
        scores(i)=generation.persons(i).score;
    end
    [tmp,sorted_indexes]=sort(scores,'descend');

    for i=1:numel(sorted_indexes)
        eval_generation.persons(i).gene=generation.persons(sorted_indexes(i)).gene;
        eval_generation.persons(i).score=generation.persons(sorted_indexes(i)).score;
    end
    eval_generation.persons(i).score;
    eval_generation.persons(i).gene;
    %%pause
```

```
        eval_generation.best_in_generation=eval_generation.persons(1).gene;
        eval_generation.best_score_in_generation=eval_generation.persons(1).score;
        eval_generation.persons(1).score;
        %pause
        if eval_generation.best_score_in_generation>=generation.best_score_in_history
          eval_generation.best_score_in_history=eval_generation.best_score_in_generation;
          eval_generation.best_in_history=eval_generation.best_in_generation;
        else
           eval_generation.best_score_in_history=generation.best_score_in_history;
           eval_generation.best_in_history=generation.best_in_history;
        end
         eval_generation.best_score_in_generation;
    end
    tmp_generations(1)=produce_generation();
    generations(1)=evaluate(tmp_generations(1));
    scores_(1)=generations(1).best_score_in_generation;
    scores;
    for gen_count=2:genetic_params.max_generations
        tmp_generations(gen_count)=produce_generation(generations(gen_count-1));
        generations(gen_count)=evaluate(tmp_generations(gen_count));
        %clc
        generations(gen_count);
        %%pause
        scores_(gen_count)=generations(gen_count).best_score_in_generation;
    end
    full_history=generations;
    best_gene=generations(end).best_in_history;
    best_score=generations(end).best_score_in_history;
    genetic_params.max_generations;
    gen_count;
    scores_;
    %pause
end
```

Ρουτίνα δημιουργίας φακέλου του πειράματος

```
function k_experiment_create(ga_params_vec,dir_name)
dir_path=sprintf('../work/%s',dir_name);
mkdir(dir_path);
iter=0;
experiment_fname=sprintf('%s/experiment.mat',dir_path);
for l=1:numel(ga_params_vec)
        results(l).ga_hist_fname=sprintf('%s/z_%d.mat',dir_path,l);
end
save(experiment_fname,'ga_params_vec','iter','results');
```

Ρουτίνα διαίρεσης του πειράματος σε κομμάτια, ώστε να τρέχει παράλληλα σε πολλαπλούς πυρήνες και μηχανήματα

```
function k_break_experiment(dir_name,pieces)
  dir_path=sprintf('../work/%s',dir_name);
  big_experiment_fname=sprintf('%s/experiment.mat',dir_path);
load(big_experiment_fname,'ga_params_vec','results');
ga_params_vect=ga_params_vec;
total_results=results;
  iter=0;
  for piece=1:pieces
    dir_path=sprintf('../work/%s/piece_%d_of_%d',dir_name,piece,pieces);
    mkdir(dir_path);
    experiment_fname=sprintf('%s/experiment.mat',dir_path);
    count=1;
    clear ga_params_vec;
    clear results;
    for l=1:numel(ga_params_vect)
      if mod(l,pieces)+1==piece
        ga_params_vec(count)=ga_params_vect(l);
        results(count).ga_hist_fname=sprintf('%s/z_%d.mat',dir_path,count);
        total_results(l).ga_hist_fname=results(count).ga_hist_fname;
        count=count+1;
      end
    end
```

```
        save(experiment_fname,'ga_params_vec','iter','results');
    end
    ga_params_vec=ga_params_vect;
    results=total_results;
    save(big_experiment_fname,'ga_params_vec','iter','results','pieces');
end
```

Ρουτίνα για “επανένωση” των κομματιών του πειράματος

```
function k_merge_experiment(dir_name)
    iter=0;
    out_dir_path=sprintf('../work/%s',dir_name);
    out_experiment_fname=sprintf('%s/experiment.mat',out_dir_path);
    load (out_experiment_fname);
    total_ga_params_vec=ga_params_vec;
    total_results=results;
    for piece=1:pieces
        in_dir_path=sprintf('../work/%s/piece_%d_of_%d',dir_name,piece,pieces);
        in_experiment_fname=sprintf('%s/experiment.mat',in_dir_path);
        load (in_experiment_fname);
        if numel(ga_params_vec)~=iter
            disp(sprintf('merge failed, %s is not complete',in_experiment_fname));
            return
        end
        count=1;
        for l=1:numel(total_ga_params_vec)
            if mod(l,pieces)+1==piece
                total_results(l).ga_hist_fname=results(count).ga_hist_fname;
                total_results(l).best_gene=results(count).best_gene;
                total_results(l).best_score=results(count).best_score;
                count=count+1;
            end
        end

    end
    load (out_experiment_fname);
```

```
   iter=numel(ga_params_vec);
   results=total_results;
   save(out_experiment_fname,'ga_params_vec','iter','results','pieces');
end
```

Ρουτίνα για επανεκκίνηση του πειράματος μετά από βίαιο τερματισμό

```
function k_experiment_resume(dir_name)
dir_path=sprintf('../work/%s',dir_name);
experiment_fname=sprintf('%s/experiment.mat',dir_path);
load ../data/globals.mat;
load (experiment_fname);
tic;

  while iter<numel(ga_params_vec)

   disp(sprintf('Starting experiment %d of %d totall execution time=%d',iter,numel(ga_params_vec),toc ))
          iter=iter+1;
   k_init(ga_params_vec(iter).expression_cutoff);
          [best_gene best_score full_history]=k_ga_core(ga_params_vec(iter));
   ga_params_save=ga_params_vec(iter);
          save(results(iter).ga_hist_fname,'full_history','best_score','best_gene','ga_params_save');
          results(iter).best_gene=best_gene;
          results(iter).best_score=best_score;
          save(experiment_fname,'ga_params_vec','iter','results');
  end
```

Συναρτήσεις καταλληλότητας

```
function res=k_fitness_variance_nth(gene)
   global data d_height d_width ga_params
   global  k_fitness_variance_th_log_data
   if(numel( k_fitness_variance_th_log_data)==0)
      k_fitness_variance_th_log_data=log2(data);
   end
   tmp_data=k_fitness_variance_th_log_data(:,gene);
   r_gene_unif=(std((tmp_data),0,2)).^2;
```

```
    %ga_params
    res=1./(1+r_gene_unif);
    %res=r_gene_unif<0.5;
    res=sum(res)./d_height;

end

function res=k_fitness_variance_th(gene)
    global data d_height d_width ga_params
    global  k_fitness_variance_th_log_data
    if(numel( k_fitness_variance_th_log_data)==0)
        k_fitness_variance_th_log_data=log2(data);
    end
    tmp_data=k_fitness_variance_th_log_data(:,gene);
    r_gene_unif=(std((tmp_data),0,2)).^2;
    %ga_params
    res=r_gene_unif<ga_params.uniformity_threshold;
    %res=r_gene_unif<0.5;
    res=sum(res)./d_height;
end

function res=k_fitness_hibrid_nth(gene)
    global data d_height d_width ga_params
    global  k_fitness_variance_th_log_data
    if(numel( k_fitness_variance_th_log_data)==0)
        k_fitness_variance_th_log_data=log2(data);
    end
    tmp_data=k_fitness_variance_th_log_data(:,gene);
    r_gene_unif=(std((tmp_data),0,2)).^2;

    res=(((1./(1+r_gene_unif))')');

    res=sum(res)/d_height;
    res=res.*mean(mean(data(:,gene)));
end
```

Συνάρτηση δημιουργίας διαφορετικών μεθόδων επιλογής μείγματος και συγκρισής τους

```
function r_plot_compaire_varnth_2kmeans_naive(recalculate,fname)
if nargin==0
   recalculate=1;
   fname='../work/ga_nth-kmeans-ga_th-naive';
end
if nargin==1
   fname='../work/ga_nth-kmeans-ga_th-naive';
end
if recalculate
k_init(100)
global data
%make cutoff data
img_res=k_ga_plot_gene_of_experiment('unif_cutoff/var_uniformity_test_cutoff');
for k=1:199;ga_cut_resc1(k)=k_fitness_variance_nth(find(img_res(:,k)));end
ga_cut_resc_idx=(0.01:0.005:1).^2*24;
img_res=k_ga_plot_gene_of_experiment('unif_cutoff/var_uniformity_cutoff_control');
for k=1:199;ga_cut_resc2(k)=k_fitness_variance_nth(find(img_res(:,k)));end
img_res=k_ga_plot_gene_of_experiment('unif_cutoff/var_uniformity_cutoff_control2');
for k=1:199;ga_cut_resc3(k)=k_fitness_variance_nth(find(img_res(:,k)));end
img_res=k_ga_plot_gene_of_experiment('unif_cutoff/var_uniformity_cutoff_control3');
for k=1:199;ga_cut_resc4(k)=k_fitness_variance_nth(find(img_res(:,k)));end
img_res=k_ga_plot_gene_of_experiment('unif_cutoff/var_uniformity_cutoff_control4');
for k=1:199;ga_cut_resc5(k)=k_fitness_variance_nth(find(img_res(:,k)));end
img_res=k_ga_plot_gene_of_experiment('unif_cutoff/var_uniformity_cutoff_control5');
for k=1:199;ga_cut_resc6(k)=k_fitness_variance_nth(find(img_res(:,k)));end
%cutoff data made
%plot data =
%ga_cut_resc6,ga_cut_resc5,ga_cut_resc4,ga_cut_resc3,ga_cut_resc2,ga_cut_re
%sc1
%idx_data = ga_cut_resc_idx
%make nth pop20 data
img_res=k_ga_plot_gene_of_experiment('var_nth_pop20_gen300_elpar5');
for k=1:199;ga_res(k)=k_fitness_variance_nth(find(img_res(:,k)));end
ga_res_idx=ga_cut_resc_idx;
```

```
%made nth pop20 data
%plot_data=ga_res
%idx_data=ga_res_idx
%make naive pattern data
cell_pattern=mean(log2(data),2);
for cl=1:83;res(:,cl)=abs(cell_pattern-log2(data(:,cl)));end
cell_error_sum=sum(res);
[a,b]=sort(cell_error_sum);
cell_pattern_gene=b(1:20);
simple_pat_res=ones(199,1)*k_fitness_variance_nth(cell_pattern_gene);
clear a b res
simple_pat_res_idx=ga_cut_resc_idx;
%made naive pattern data
%plot_data=simple_pat_res
%idx_data=simple_pat_res_idx
%%make random data
for k=1:199;a=randperm(83);rand_res(k)=k_fitness_variance_nth(a(1:20));end
rand_res_idx=(0.01:0.005:1)*24;
%%made random data
%plot_data=random_data
%idx_data=rand_res_idx
%%%%%%%%%%%%%%%%%%%% k-means %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
tic
for k_t=1:199
   k_t
   toc
   clear clust larger_clust c good_cell_lines kmeans_cell_pattern cell_error_sum
   clust=kmeans(log2(data)',10);
   larger_clust=1;
   for c=2:10
    if(sum(clust==c)>sum(clust==larger_clust))
       larger_clust=c;
    end
   end
   good_cell_lines=(clust==larger_clust);
```

```
    kmeans_cell_pattern=mean(log2(data(:,good_cell_lines)),2);
    for cl=1:83;res(:,cl)=abs(kmeans_cell_pattern-log2(data(:,cl)));end
    cell_error_sum=sum(res);
    [a,b]=sort(cell_error_sum);
    kmeans_res(k_t)=k_fitness_variance_nth(b(1:20));
    kmeans_genes(k_t,:)=b(1:20);
end
kmeans_res_idx=ga_cut_resc_idx;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
save(fname,'kmeans_res','kmeans_res_idx','rand_res','rand_res_idx','ga_res','ga_res_idx','simple_pat_res','simple_pat_res_idx','ga_cut_resc_idx','ga_cut_resc6','ga_cut_resc5','ga_cut_resc4','ga_cut_resc3','ga_cut_resc2','ga_cut_resc1')
clear global data
end
load (fname)
hold off;
plot(rand_res_idx,rand_res,'*','color','blue');
hold on;
plot(ga_res_idx,ga_res,'+','color','black');
plot(kmeans_res_idx,kmeans_res,'*','color','magenta');
plot(simple_pat_res_idx,simple_pat_res,'green');
plot(ga_cut_resc_idx,ga_cut_resc1,'.','color','red');
plot(ga_cut_resc_idx,ga_cut_resc2,'.','color','red');
plot(ga_cut_resc_idx,ga_cut_resc3,'.','color','red');
plot(ga_cut_resc_idx,ga_cut_resc4,'.','color','red');
plot(ga_cut_resc_idx,ga_cut_resc5,'.','color','red');
plot(ga_cut_resc_idx,ga_cut_resc6,'.','color','red');
legend('random picks pop=199','nth ga picks pop=199','kmeans picks pop=199','naive pattern picks pop=1','threshold ga picks pop=1199','Location','SouthEast');
xlabel('Uniformity Threshold for "threshold ga picks" irrelevant for others');
ylabel('1/(1+variance(log2(gene expressions)))');
title({'Comparative plot of different optimization methods';'All methods are measured as the average of "1/(1+variance(log2(gene expressions)))" for each gene respectively in proposed cell lines'});
hold off
```