



ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

UNIVERSITY OF PIRAEUS

Ακαδημαϊκό έτος: 2020-2021

Εαρινό Εξάμηνο

Ονοματεπώνυμο: Πατάκης Ανδρέας
Αριθμός Μητρώου: Π17103

Τεχνητή Νοημοσύνη και Έμπειρα Συστήματα
Εργασία 2

Σκέλος 2.1:

Γενετικός Αλγόριθμος

Τρόπος δράσης του γενετικού αλγορίθμου και ανάλυση προσέγγισης του προβλήματος

Κατά την γενική μορφή του ένας γενετικός αλγόριθμος προσπαθεί να προσεγγίσει την λύση ενός προβλήματος χρησιμοποιώντας τεχνικές βασισμένες στην θεωρία της εξέλιξης. Αυτό σημαίνει ότι κάθε λύση(γονίδιο) συνυπάρχει εντός ενός γενικευμένου σύνολου λύσεων το οποίο θεωρούμε ως γενιά, ενώ για να παραχθεί μια νέα γενιά λαμβάνουν χώρο μια σειρά απο διασταυρώσεις και μεταλλάξεις (μεταξύ των διαθέσιμων λύσεων).

Προσέγγιση

Το ζητούμενο πρόβλημα προς λύση ήταν να δημιουργηθεί το γράμμα “Π” σε ένα grid 11x7. Αυτό σημαίνει πως το παραπάνω grid περιέχει συνολικά 77 πλαίσια. Έτσι η μορφή της λύσης μας αποτελείται από μια λίστα θέσεων οι οποίες όταν ενεργοποιηθούν σχηματίζουν το γράμμα “Π”

Το grid έχει την παρακάτω μορφή:

Genetic Algorithm Execution

file:///C:/Users/Jojo/Documents/PBX Funicular Intaglio Zone/Al_Search_Algorithms/GeneticAlgo/index.html

Press start and let the genetic algorithm begin!

Total generations: 0

Average fitness score: 0%

Total population: 100

Elitism rate: 10%

Mutation rate: 10%

Start

Type here to search

2:02 AM 5/22/2021

Στην περίπτωση μας οι θέσεις αυτές αντιστοιχούν στις παρακάτω.

```
solution_arr = [71, 64, 57, 50, 43, 36, 29, 22, 15, 8, 9, 10, 11, 12, 19, 26, 33, 40, 47, 54, 61, 68, 75]
```

Επομένως το πρόβλημα χρωματισμού του παραπάνω γράφου υπάγεται στο γενικότερο πρόβλημα ταύτισης μιας ακολουθίας.

Σειρά εκτέλεσης ενεργειών

1. Αρχικά ο αλγόριθμος μας ξεκινάει δημιουργώντας την πρώτη, τυχαία παραγμένη, γενιά λύσεων. Το πλήθος των λύσεων, δηλαδή ο πληθυσμός(population) της γενιάς, θα έχει την τιμή την οποία θα επιλέξει ο χρήστης πριν την εκκίνηση του αλγορίθμου. Οι επιλογές οι οποίες του δίνονται είναι οι εξής:

Total population:	<input type="text" value="100"/>	▼
	100	
	200	
Elitism rate:	<input type="text" value="10%"/>	300
		400
Mutation rate:	<input type="text" value="10%"/>	500
		600
<input type="button" value="Start"/>		700
		800
		900
		1000

Επομένως κάθε γενιά έχει N λύσεις(γονίδια), όσες και ο πληθυσμός, και κάθε λύση έχει την αναπαράσταση που δόθηκε παραπάνω. Αυτό σημαίνει ότι κάθε λύση είναι ουσιαστικά ένας πίνακας 23 θέσεων. Το πρόγραμμα είναι υλοποιημένο με τέτοιο τρόπο έτσι ώστε να είναι ευέλικτο, επομένως αν θέλουμε να αναπαραστήσουμε κάποιο άλλο γράμμα αρκεί να παραθέσουμε στην αρχή του προγράμματος ένα άλλο `solution_arr` οι θέσεις του οποίου θα αντιστοιχούν στη επιθυμητή λύση-γράμμα.

2. Αφού δημιουργηθεί μια γενιά, σειρά έχει η αξιολόγηση κάθε λύσης της. Προκειμένου να αξιολογήσουμε μια λύση χρησιμοποιούμε μια **συνάρτηση καταλληλότητας**(fitness function). Η συνάρτηση καταλληλότητας που χρησιμοποιήθηκε στο συγκεκριμένο πρόβλημα είχε την εξής λογική:

- Ψάξε για 1-1 αντιστοιχία κάθε θέσης του πίνακα μίας λύσης με τις αντίστοιχες θέσεις του πίνακα της πραγματικής λύσης.
- Αν οι θέσεις διαφέρουν, αύξησε κατά 1 έναν μετρητή ο οποίος μετράει την απόκλιση.

Η παραπάνω διαδικασία γίνεται για ολόκληρο των πληθυσμό μίας γενιάς και με αυτών των τρόπο κάθε λύση αξιολογείται με βάση την απόκλιση της απο την επιθυμητή λύση. Αυτό σημαίνει οτι όσο μικρότερο το score μια λύσης, τόσο καλύτερη θεωρείται αφού μετράμε την απόκλιση. Τέλος όταν κάθε λύση έχει βαθμολογηθεί, γίνεται μια αύξουσα ταξινόμηση στον πληθυσμό προκειμένου να έχει την μορφή: καλύτερη λύση → χειρότερη λύση.

3. Αφού ολοκληρωθούν οι παραπάνω διαδικασίες σειρά έχει η αναπαραγωγή της επόμενης γενιάς, κατά την οποία λαμβάνουν χώρα διάφορες διαδικασίες τις οποίες θα εξηγήσουμε τώρα.

Αρχικά η γενιά η οποία πρόκειται να αναπαραχθεί αντιγράφεται ολόκληρη σε μια “δεξαμενή αναπαραγωγής” (mating pool) προκειμένου να γίνουν οι απαραίτητες ενέργειες (διασταυρώσεις, μεταλλάξεις, μερική ανανέωση).

Όσων αφορά τις διασταυρώσεις θα πρέπει να επιλέξουμε ένα μηχανισμό επιλογής γονιδίων(λύσεων) ο οποίος θα διαλέγει τα γονίδια τα οποία θα διασταυρώνονται κάθε φορά. Ο μηχανισμός ο οποίος επιλέχθηκε είναι αυτός της Ανάλογης Καταλληλότητας(fitness proportionate selection) ο οποίος δίνει πιθανότητα επιλογής ενός γονιδίου ανάλογη με το score καταλληλότητας τους.

Επομένως το πρώτο πράγμα το οποίο συμβαίνει στην δεξαμενή αναπαραγωγής είναι η ανάθεση πιθανοτήτων επιλογής σε κάθε ένα από τα γονίδια της τρέχουσας γενιάς, μέσω της συνάρτησης `calculate_odds(generation)`.

ΣΗΜΕΙΩΣΗ: ο τρόπος με τον οποίο αναθέτει τις πιθανότητες η παραπάνω συνάρτηση βασίζεται αποκλειστικά στο score και ΟΧΙ στην συχνότητα εμφάνισης μιας λύσης. Δηλαδή αν μια κακή λύση κατέχει το 95% μια γενιάς ενώ το υπόλοιπο 5% μια καλή(έστω ότι είναι μονάχα δυο), η καλύτερη θα έχει μεγαλύτερη πιθανότητα επιλογής(πάντα ανάλογη του score της) παραβλέποντας την συχνότητα εμφάνισης της.

Αφού ανατεθούν οι πιθανότητες επιλογής σε κάθε γονίδιο, σειρά έχει η διαδικασία της *μερικής ανανέωσης*. Κατά την μερική ανανέωση ένα ποσοστό του πληθυσμού(το καλύτερο) αντιγράφεται απευθείας στην επόμενη γενιά(ελιτισμός). Στην περίπτωση μας το ποσοστό αυτό το επιλέγει ο χρήστης πριν την εκτέλεση του αλγορίθμου έχοντας τις εξής επιλογές:

Elitism rate: 10% ▼

Mutation rate: 10%

20%

30%

40%

50%

60%

70%

80%

90%

100%

Start

Όταν η μερική ανανέωση ολοκληρωθεί, θα ξεκινήσουν οι διασταυρώσεις προκειμένου να αναπαραχθεί το υπόλοιπο ποσοστό της γενιάς που πρέπει να συμπληρωθεί (πχ. Αν ο χρήστης επιλέξει 100% μερική ανανέωση, δεν θα γίνει καμία διασταύρωση). Η επιλογή των γονέων γίνεται με βάση των πιθανοτήτων οι οποίες αναλύθηκαν νωρίτερα, έως ότου συμπληρωθεί ο κατάλληλος αριθμός γονιδίων για την επόμενη γενιά. Για την αναπαραγωγή χρησιμοποιήθηκε διασταύρωση ενός σημείου (single-point crossover).

Τέλος, έχοντας ολοκληρώσει την δημιουργία της επόμενης γενιάς θα παρουσιάσουμε μια μικρή μετάλλαξη (σε μονάχα ένα σημείο μιας λύσης) σε πλήθος γονιδίων το οποίο θα επιλέξει ο χρήστης. Το ποσοστό του πληθυσμού στο οποίο θα γίνει μετάλλαξη κυμαίνεται στα εξής:

Mutation rate:

- 10%
- 20%
- 30%
- 40%
- 50%
- 60%
- 70%
- 80%
- 90%
- 100%

Σύνοψη

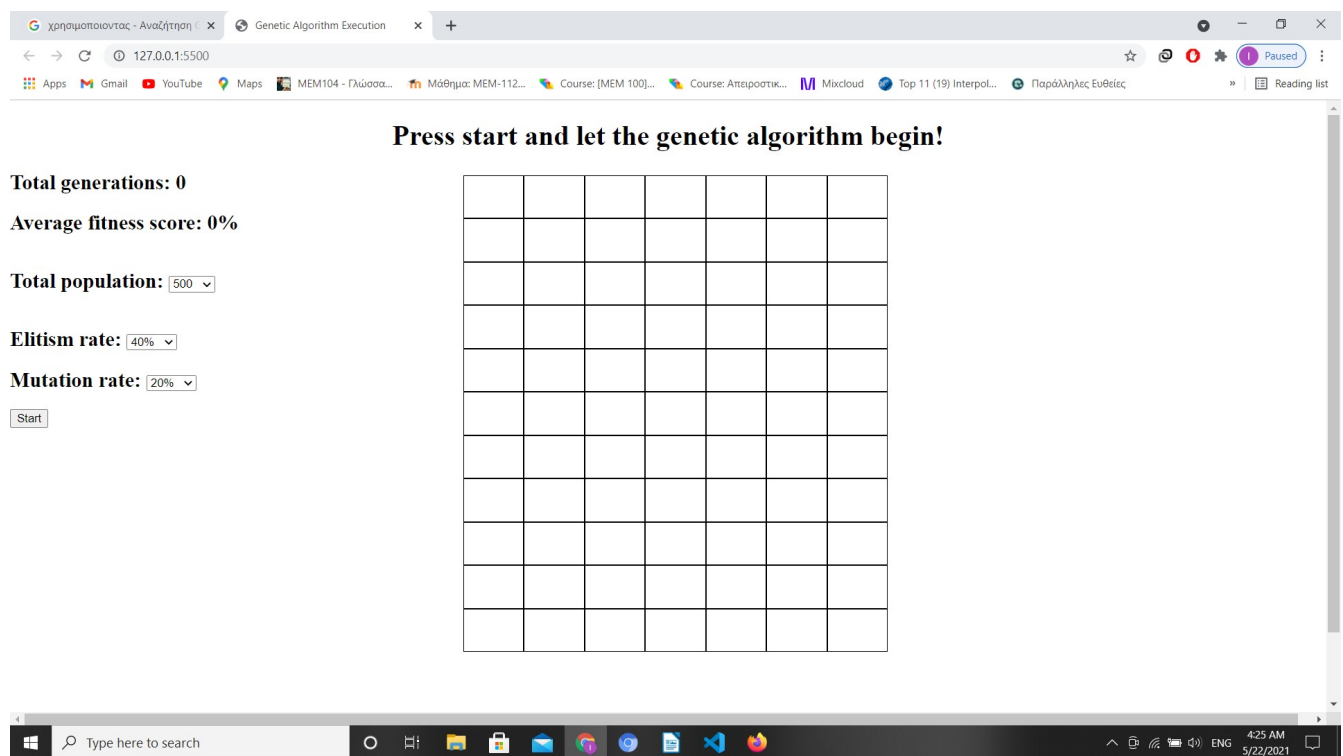
Η παραπάνω διαδικασία επαναλαμβάνεται συνεχώς, δημιουργώντας καινούριες γενιές οι οποίες συγκλίνουν όλο και περισσότερο προς την επιθυμητή λύση. Ανάλογα με τις επιλογές τις οποίες θα κάνει ο χρήστης τόσο για τον συνολικό αριθμό πληθυσμού όσο και για τα ποσοστά μερικής ανανέωσης(ελιτισμού) και μετάλλαξης, η λύση αυτή μπορεί να βρεθεί πολύ σύντομα έως και καθόλου.

Το πρόγραμμα είναι γραμμένο σε javascript προκειμένου να είναι εύκολο στον οποιοδήποτε να το ανοίξει και να τρέξει τα δικά του παραδείγματα με τις δικές τους επιλογές. Για εκτέλεση απλώς ανοίξτε το αρχείο index στον browser της επιλογής σας. Ακολουθούν κάποια screenshots από εκτελέσεις του αλγορίθμου.

Στο αρχείο που σας παρέδωσα υπάρχει και ένα βίντεο λίγων δευτερολέπτων το οποίο δείχνει την εκτέλεση του αλγορίθμου και πως σιγά σιγά βρίσκει την λύση.

******Στο βίντεο για κάποιο λόγο δεν εμφανίζονται τα dropdownlist των επιλογών και δεν φαίνεται τι επιλέγω, δεν γνωρίζω γιατί. Προφανώς αν ανοίξετε το index στον browser σας όλα λειτουργούν κανονικά.

Screenshots



χρησιμοποιώντας - Αναζήτηση x Genetic Algorithm Execution x +

127.0.0.1:5500

Apps Gmail YouTube Maps MEM104 - Γλώσσα... Μάθημα: MEM-112... Course: [MEM 100]... Course: Απειροστικ... Mixcloud Top 11 (19) Interpol... Παράλληλες Ευθείες Reading list

Press start and let the genetic algorithm begin!

Total generations: 3

Average fitness score: 8%

Total population: 500 ▾

Elitism rate: 40% ▾

Mutation rate: 20% ▾

Start

χρησιμοποιώντας - Αναζήτηση x Genetic Algorithm Execution x +

127.0.0.1:5500

Apps Gmail YouTube Maps MEM104 - Γλώσσα... Μάθημα: MEM-112... Course: [MEM 100]... Course: Απειροστικ... Mixcloud Top 11 (19) Interpol... Παράλληλες Ευθείες Reading list

Press start and let the genetic algorithm begin!

Total generations: 42

Average fitness score: 77%

Total population: 500 ▾

Elitism rate: 40% ▾

Mutation rate: 20% ▾

Start

Press start and let the genetic algorithm begin!

Total generations: 140

Average fitness score: 95%

Total population: 500

Elitism rate: 40%

Mutation rate: 20%

Start

