

# ΠΜΣ ΕΠΙΣΤΗΜΗ ΚΑΙ ΜΗΧΑΝΙΚΗ ΔΕΔΟΜΕΝΩΝ

ΕΛΑΧΙΣΤΟΠΟΙΗΣΗ ΤΟΥ ΔΥΝΑΜΙΚΟΥ LENNARD-JONES ΜΕ  
ΧΡΗΣΗ ΓΕΝΕΤΙΚΩΝ ΑΛΓΟΡΙΘΜΩΝ ΚΑΙ ΝΟΗΜΟΣΤΥΝΗΣ  
ΣΜΗΝΩΝ

Γεώργιος Βαρδάκας Α.Μ. 432



Υπεύθυνος Καθηγητής:  
Παρσόπουλος Κωνσταντίνος

Τμήμα Μηχανικών Η/Υ και Πληροφορικής  
Πολυτεχνική Σχολή Πανεπιστημίου Ιωαννίνων

# Περιεχόμενα

<b>1</b>	<b>Εισαγωγή</b>	<b>2</b>
<b>2</b>	<b>Εξελικτικοί αλγόριθμοι και νοημοσύνη σμηνών</b>	<b>2</b>
2.1	Γενετικός αλγόριθμος δυαδικής αναπαράστασης . . . . .	3
2.2	Γενετικός αλγόριθμος πραγματικής αναπαράστασης . . . . .	4
2.3	Particle Swarm Optimization . . . . .	5
<b>3</b>	<b>Πειράματα</b>	<b>6</b>
3.1	Αποτελέσματα για πλήθος ατόμων $N = 4$ . . . . .	6
3.2	Αποτελέσματα για πλήθος ατόμων $N = 5$ . . . . .	8
3.3	Αποτελέσματα για πλήθος ατόμων $N = 6$ . . . . .	9
3.4	Αποτελέσματα για πλήθος ατόμων $N = 7$ . . . . .	11
3.5	Αποτελέσματα για πλήθος ατόμων $N = 8$ . . . . .	13
3.6	Αποτελέσματα για πλήθος ατόμων $N = 9$ . . . . .	14
3.7	Αποτελέσματα για πλήθος ατόμων $N = 10$ . . . . .	16
3.8	Αποτελέσματα για πλήθος ατόμων $N = 15$ . . . . .	17
3.9	Αποτελέσματα για πλήθος ατόμων $N = 20$ . . . . .	19
3.10	Συμπεράσματα . . . . .	20
<b>4</b>	<b>Δομή αναπτυσσόμενου κώδικα</b>	<b>20</b>
<b>5</b>	<b>Παρατηρήσεις για την ταχύτητα του κώδικα</b>	<b>23</b>
<b>6</b>	<b>Οδηγίες για την εκτέλεση του κώδικα</b>	<b>23</b>

# 1 Εισαγωγή

Με τον όρο συστάδες ατόμων (atom clusters) χαρακτηρίζουμε τις ευσταθείς (συνήθως) ομάδες ατόμων που σχηματίζονται διαμέσου φυσικών και χημικών δεσμών μεταξύ των ατόμων παράγοντας νέα στοιχεία. Οι φυσικές και χημικές ιδιότητες των συστάδων αλλάζουν σημαντικά σε σχέση με τον αριθμό των ατόμων που τις απαρτίζουν. Οι ιδιότητες αυτές παίζουν πολύ σημαντικό ρόλο στην κατασκευή νέων υλικών με συγκεκριμένα χαρακτηριστικά αγωγιμότητας, αποθήκευσης ενέργειας κ.λ.π.

Μια από τις μεθόδους μελέτης ατομικών συστάδων έγκειται στη μελέτη του ενεργειακού προφίλ της συστάδας διαμέσου εξισώσεων δυναμικού (potential) που περιγράφουν τις αλληλεπιδράσεις μεταξύ των ατόμων. Ένα τέτοιο μοντέλο συστάδων, το οποίο αποδεικνύεται ιδιαίτερα ακριβές για την περίπτωση των αδρανών αερίων, είναι εκείνες των Lennard-Jones (Lennard-Jones clusters). Η δομή αυτών των συστάδων, περιγράφει τη διάταξη των ατόμων στο χώρο επηρεάζουν άμεσα την ελάχιστη τιμή της δυναμικής ενέργειας του συστήματος.

Η ελάχιστη τιμή της δυναμικής ενέργειας μιας τέτοιας συστάδας ατόμων συνήθως αντιστοιχεί στη θεμελιώδη κατάσταση του συστήματος κι επηρεάζεται άμεσα από τη δομή, δηλαδή τη διάταξη των ατόμων της στον χώρο.

**Σκοπός της παρούσας εργασίας** είναι η εύρεση της βέλτιστης διάταξης των ατόμων στο χώρο με στόχο την εύρεση εκείνης της δομής που ελαχιστοποιεί τη δυναμική ενέργεια της συστάδας όπως έχουν ορίσει οι Lennard-Jones. Η συνάρτηση δυναμικού Lennard-Jones που υπολογίζει την δυναμική ενέργεια μιας συστάδας  $N$  ατόμων προκύπτει ως εξής:

Έστω ότι μια συστάδα  $X$  αποτελείται από  $N$  άτομα,  $X = \{x_1, \dots, x_N\}$  και κάθε  $x_i$  ορίζει τις συντεταγμένες του στον χώρο ως  $x_i = (x_{i1}, x_{i2}, x_{i3})^T$ . Κατά συνέπεια ορίζεται ως  $X = \{x_{11}, x_{12}, x_{13}, \dots, x_{N1}, x_{N2}, x_{N3}\}$ . Συμβολίζουμε ως  $r_{ij} = \|x_i - x_j\|_2$  την Ευκλείδεια απόσταση των ατόμων  $x_i$  και  $x_j$  για τα οποία η ενέργεια που προκύπτει για το ζεύγος αυτό δίνεται από την σχέση  $u_{ij} = (\frac{\sigma}{r_{ij}})^{12} - (\frac{\sigma}{r_{ij}})^6$ . Τέλος η δυναμική ενέργεια ολόκληρης της συστάδας των  $N$  ατόμων, που καλούμαστε να ελαχιστοποιήσουμε, δίνεται από τη συνάρτηση δυναμικού Lennard-Jones:  $E_{LJ} = 4\epsilon \sum_{i=1}^{N-1} \sum_{j=i+1}^N u_{ij}$ , όπου  $\sigma$  (equilibrium pair separation) και  $\epsilon$  (pair well depth) είναι παράμετροι του δυναμικού. Η παρούσα εργασία θα επικεντρωθεί σε συστάδες με παραμέτρους  $\sigma = \epsilon = 1$ .

## 2 Εξελικτικοί αλγόριθμοι και νοημοσύνη σμηνών

Οι αλγόριθμοι που θα εξεταστούν ανήκουν στην γενική κατηγορία των μεθόδων βελτιστοποίησης χωρίς παραγώγους και συγκεκριμένα στην κατηγορία των εξελικτικών αλγορίθμων και νοημοσύνης σμηνών. Πιο συγκεκριμένα οι μεθοδολογίες που εξετάστηκαν είναι οι εξής:

- Γενετικός αλγόριθμος δυαδικής αναπαράστασης με επιλογή roulette wheel.
- Γενετικός αλγόριθμος δυαδικής αναπαράστασης με επιλογή tournament selection.
- Γενετικός αλγόριθμος πραγματικής αναπαράστασης με επιλογή tournament selection.
- Μέθοδος Particle Swarm Optimization με global best μοντέλο.
- Μέθοδος Particle Swarm Optimization με local best μοντέλο

## 2.1 Γενετικός αλγόριθμος δυαδικής αναπαράστασης

Ο τρόπος λειτουργίας των Γενετικών Αλγορίθμων είναι εμπνευσμένος από τη βιολογία. Χρησιμοποιεί την ιδέα της εξέλιξης ως φυσικής επιλογής, διασταύρωσης και γενετικής μετάλλαξης. Αρχικά, στην διαδικασία του Γενετικού Αλγορίθμου αρχικοποιείται ένα τυχαίο σύνολο πληθυσμού, συνήθως με τυχαίες τιμές, δημιουργώντας ένα αρχικό σύνολο λύσεων. Στην περίπτωση της δυαδικής αναπαράστασης ο πληθυσμός αυτός είναι αρχικοποιημένος με τιμές 0 ή 1. Έπειτα ο πληθυσμός από δυαδική αναπαράσταση μετατρέπεται σε πραγματική (decoding), έτσι ώστε κάθε μέλος του πληθυσμού να αξιολογηθεί με βάση την συναρτησιακή του τιμή. Στη συνέχεια παρατίθεται το κύριο μέρος του αλγορίθμου που καλείται επαναληπτικά με τους εξής κεντρικούς άξονες:

**Επιλογή:** είναι το πρώτο βήμα του γενετικού αλγορίθμου, όπου από τον πληθυσμό επιλέγονται με βάση την συναρτησιακή τους τιμή τα καλύτερα υποψήφια μέλη του. Οι δύο μέθοδοι της επιλογής που εξετάστηκαν στην εργασία είναι η επιλογή με roulette wheel καθώς και με tournament selection.

Στην επιλογή με roulette wheel αναθέτουμε σε όλα τα μέλη του πληθυσμού ένα fitness score με βάση την θέση που κατέχει το κάθε μέλος λαμβάνοντας υπόψη τη συναρτησιακή του τιμή (εφόσον οι συναρτησιακές τιμές είναι ταξινομημένες σε φθίνουσα σειρά). Στην συνέχεια το fitness score του μέλους  $p_i$  υπολογίζεται ως εξής:  $\phi_i = 2 - s + 2(s - 1)\frac{\rho_i - 1}{N - 1}$ ,  $i = 1, \dots, N$ , όπου  $\rho_i$  η θέση του μέλους  $p_i$  και  $s$  παράμετρος επιλεκτικής πίεσης ( $s \in [1, 2]$ ). Στην συνέχεια κάνοντας χρήση των fitness score του πληθυσμού υπολογίζουμε τις πιθανότητες επιλογής του κάθε μέλους κανονικοποιώντας τα fitness score. Έπειτα δειγματοληπτούμε ομοιόμορφα κάνοντας χρήση της ομοιόμορφης κατανομής  $N$  τυχαίους αριθμούς στο διάστημα  $[0, 1]$  όπου και επιλέγονται μέλη των οποίων οι τυχαίοι αριθμοί βρίσκονται κατανεμημένοι εντός των εκάστοτε διαστημάτων πιθανότητας.

Στην επιλογή με tournament selection σημαντικό ρόλο παίζει η παράμετρος που ονομάζεται μέγεθος τουρνουά (tournament size). Το μέγεθος τουρνουά λαμβάνει τιμές από  $[1, N]$ . Με την διαδικασία αυτή επιλέγονται τυχαία μέλη από τον πληθυσμό, όσα και το μέγεθος του τουρνουά σε πλήθος, και διαλέγουμε αυτόν με την καλύτερη συναρτησιακή τιμή. Αυτή η διαδικασία επαναλαμβάνεται  $N$  φορές.

**Διασταύρωση:** αποτελεί το δεύτερο βήμα του γενετικού αλγορίθμου, όπου από τον επιλεγμένο πληθυσμό διαλέγονται με κάποια πιθανότητα (crossover rate) μέλη με σκοπό να κάνουν απογόνους. Στην συνέχεια το "DNA" των γονέων διαχωρίζεται σε ένα πλήθος σημείων (crossover points) τα οποία συνιστούν δυο ακόμη μέλη που ονομάζουμε απογόνους (παιδιά). Οι απόγονοι λαμβάνουν εναλλάξ κομμάτια "DNA" από τους δύο γονείς τους. Στο τέλος της διαδικασίας οι απόγονοι αντικαθιστούν τους γονείς στον πληθυσμό.

**Μετάλλαξη:** συνιστά το τελευταίο βασικό βήμα του γενετικού αλγορίθμου, όπου από τον διασταυρωμένο πληθυσμό επιλέγονται με κάποια πιθανότητα συνιστώσες μελών (mutation rate) για μετάλλαξη. Στην διαδικασία αυτή η αρχική τιμή των υπο μετάλλαξη συνιστωσών αλλάζει/μεταβάλλεται από  $i \rightarrow j$ ,  $i, j = 0, 1$ , δηλαδή εάν έχουν τιμή 1 γίνονται 0 και αντίστροφα.

Τέλος ανάμεσα στον αρχικό πληθυσμό και στον υπό μετάλλαξη πληθυσμό διαλέγονται οι  $N$  καλύτεροι με βάση την συναρτησιακή τους τιμή ως αρχική τιμή πληθυσμού για την επόμενη επανάληψη.

Parameter	Value
Population size	1000
Selective pressure	1.3
Tournament size	100
Crossover rate	0.5
Crossover points	6
Mutation rate	0.1
$\epsilon$	$10^{-3}$

Πίνακας 1: Τιμές παραμέτρων του γενετικού αλγορίθμου δυαδικής αναπαράστασης

## 2.2 Γενετικός αλγόριθμος πραγματικής αναπαράστασης

Οι αρχές λειτουργίας αυτής της μεθόδου είναι ίδιες με αυτές που περιγράφηκαν στην προηγούμενη [παράγραφο](#). Η κύρια διαφορά της είναι ότι η συγκεκριμένη μέθοδος ακολουθεί πραγματική και όχι δυαδική αναπαράσταση και δεν απαιτείται η διαδικασία του decoding. Πλέον ο πληθυσμός αρχικοποιείται τυχαία με πραγματικές τιμές και αξιολογείται με βάση τις συναρτησιακές τους τιμές δημιουργώντας το αρχικό σύνολο λύσεων. Όπως και στην προηγούμενη μέθοδο οι βασικές διαδικασίες είναι η επιλογή, η διασταύρωση και η μετάλλαξη. Η διαδικασία της επιλογής είναι ακριβώς η ίδια με εκείνη που περιγράφηκε παραπάνω.

**Πραγματική Διασταύρωση:** αποτελεί το δεύτερο βήμα του γενετικού αλγορίθμου με πραγματική αναπαράσταση, όπου από τον επιλεγμένο πληθυσμό διαλέγονται με κάποια πιθανότητα (crossover rate) μέλη του ώστε να να δημιουργήσουν απογόνους. Στην συνέχεια το "D-NA" των γονέων συνδυάζεται με την παρακάτω διαδικασία για την δημιουργία ενός απογόνου: Έστω  $o = (o_1, \dots, o_n)$  απόγονος του  $p_{i1}, p_{i2}$ , τότε η συνιστώσα  $o_j = r_j p_{i1,j} + (1 - r_j) p_{i2}$  με  $r_j \sim U(-\delta, 1 + \delta)$  τυχαίος αριθμός με  $\delta > 0$ .

**Πραγματική Μετάλλαξη:** συνιστά το τελευταίο βασικό βήμα του γενετικού αλγορίθμου με πραγματική αναπαράσταση, όπου από τον διασταυρωμένο πληθυσμό επιλέγονται με κάποια πιθανότητα συνιστώσες μελών (mutation rate) για να υποστούν μετάλλαξη. Στην διαδικασία αυτή στις υπό μετάλλαξη συνιστώσες, προστίθεται ένας τυχαίος αριθμός  $z \sim N(0, \sigma^2)$ , όπου το  $3\sigma$  είναι ίσο με το πλησιέστερο (άνω ή κάτω) όριο του χώρου αναζήτησης, ούτως ώστε μόνο ένα πολύ μικρό ποσοστό της τάξης του 0.03% των επιλεγμένων προς μετάλλαξη συνιστωσών να αποκλίνει ελάχιστα εκτός του χώρου αναζήτησης.

Τέλος ανάμεσα στον αρχικό πληθυσμό και σε εκείνον που προκύπτει μετά το πέρας της μετάλλ-λαξης πληθυσμό επιλέγονται από το δείγμα οι  $N$  καλύτεροι με βάση την συναρτησιακή τους τιμή ως αρχική τιμή συνόλου πληθυσμού για την επόμενη επανάληψη.

Parameter	Value
Population size	1000
Tournament size	100
Crossover rate	0.5
Mutation rate	0.1
$\delta$	0.25

Πίνακας 2: Τιμές παραμέτρων του γενετικού αλγορίθμου πραγματικής αναπαράστασης

## 2.3 Particle Swarm Optimization

Ο τρόπος λειτουργίας της νοημοσύνης σμήνων (PSO) είναι εμπνευσμένος από τη συλλογική συμπεριφορά μη κατανεμημένων, αυτοοργανωμένων φυσικών συστημάτων. Η νοημοσύνη σμήνους είναι μια ιδιότητα των συστημάτων που επιδεικνύουν συλλογικά ευφύη συμπεριφορά. Αρχικά, στην διαδικασία της μεθόδου Particle Swarm Optimization αρχικοποιείται τυχαία ένα σμήνος δημιουργώντας το αρχικό σύνολο λύσεων καθώς και μία αρχική τυχαία ταχύτητα για κάθε μέλος του σμήνους. Επίσης για το κάθε μέλος του σμήνους υπάρχει η καλύτερη θέση που έχει επισκεφθεί καθώς και η γειτονιά του η οποία και καθορίζει με ποια άλλα σωματίδια θα ανταλλάξει πληροφορία. Πρακτικά κάθε γειτονιά αποτελείται από ένα πλήθος δεικτών άλλων σωματιδίων με τα οποία ανταλλάσσεται η πληροφορία καλύτερης θέσης. Οι διάφορες τοπολογίες των γειτονιών των σωματιδίων κατασκευάζουν και διαφορετικά μοντέλα του αλγορίθμου. Τα μοντέλα που μελετήθηκαν στην εργασία είναι το μοντέλο global best και local best. Στο μοντέλο global best το κάθε σωματίδιο ανταλλάσσει πληροφορία με όλα τα υπόλοιπα του πληθυσμού. Στην περίπτωση του local best μοντέλου το  $i$ -σωματίδιο ανταλλάσσει πληροφορία μόνο με τα σωματίδια με δείκτες  $i - \rho, \dots, \rho + i$  όπου  $\rho$  είναι η ακτίνα της γειτονιάς. Τελικά το σμήνος αξιολογείται με βάση την συναρτησιακή του τιμή και ακολουθεί το κύριο μέρος του αλγορίθμου που καλείται επαναληπτικά.

**Υπολογισμός ταχύτητας:** σε αυτή τη διαδικασία, υπολογίζεται η καινούργια ταχύτητα του κάθε σωματιδίου του σμήνους με βάση τον τύπο:

$$v_{ij}^{t+1} = \chi[v_{ij}^t + \text{rand}()c_1(p_{ij}^t - x_{ij}^t) + \text{rand}()c_2(p_{g_{ij}}^t - x_{ij}^t)]$$

όπου  $g_i$  δείκτης του καλύτερου σωματιδίου στην γειτονιά συγκεκριμένης ακτίνας,  $\chi, c_1, c_2$  θετικοί παράμετροι και  $\text{rand}()$  τυχαίοι αριθμοί στο διάστημα  $[0, 1]$ .

**Έλεγχος ταχύτητας:** σε αυτό το βήμα ελέγχεται εάν η ταχύτητα έχει ξεπεράσει κάποια μέγιστη τιμή (καθορίζεται από την παράμετρο  $\alpha$ ), έτσι ώστε να μειωθεί στο μέγιστο δυνατό όριο της. Η διαδικασία αυτή γίνεται προληπτικά για να μην έχουμε φαινόμενα έκρηξης του σμήνους. Κύριο στόχο έχει να αποτρέψει τα σωματίδια να αναπτύξουν τόσο μεγάλες ταχύτητες που θα τα φέρουν εκτός του χώρου αναζήτησης.

**Υπολογισμός θέσης:** σε αυτό το βήμα υπολογίζουμε τις καινούργιες θέσεις του σμήνους με βάση τη σχέση:  $x_{ij}^{t+1} = x_{ij}^t + v_{ij}^{t+1}$ , μεταβάλλοντας τα δηλαδή κατ' αυτό το τρόπο στον χώρο αναζήτησης.

**Έλεγχος σμήνους:** στο βήμα αυτό ελέγχεται εάν τα μέλη του σμήνους έχουν βγει εκτός του χώρου αναζήτησης. Εάν κάποιο μέλος έχει ξεφύγει τότε τοποθετείται στο όριο του χώρου αναζήτησης.

Τέλος οι καινούργιες λύσεις αξιολογούνται με βάση την συναρτησιακή τους τιμή και πραγματοποιείται η ανανέωση των καλύτερων θέσεων του πληθυσμού έως την συγκεκριμένη χρονική στιγμή.

Parameter	Value
Swarm size	1000
local best $\rho$	5
global best $\rho$	500
Alpha	0.5
$c_1$	2.05
$c_2$	2.05
$\chi$	0.729

Πίνακας 3: Τιμές παραμέτρων της PSO

### 3 Πειράματα

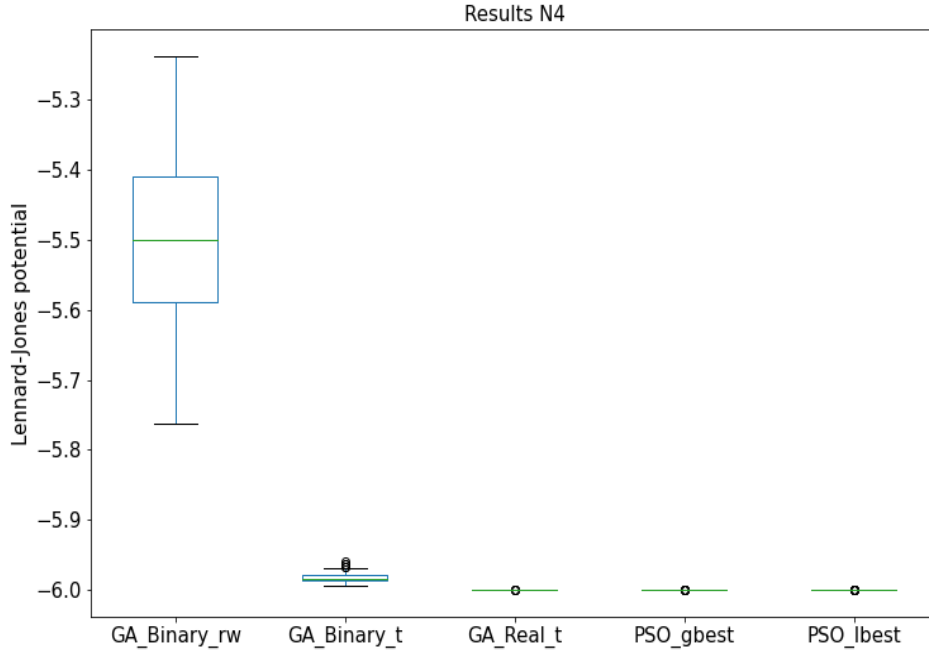
Στην τρέχουσα ενότητα παρατίθενται τα αποτελέσματα των πειραμάτων για τα διαφορετικά πλήθη ατόμων  $N$ . Για κάθε πείραμα το επιτρεπτό πλήθος συναρτησιακών υπολογισμών προσδιορίζεται από την σχέση  $T = N \times 10^5$  και  $N$  το πλήθος των ατόμων. Σε όλα τα πειράματα έχει πραγματοποιηθεί το στατιστικό τεστ Wilcoxon με επίπεδο σημαντικότητας  $\alpha = 0.05$ . Σαν πρωτεύον κριτήριο χρησιμοποιήθηκε η ποιότητα της λύσης των μεθόδων και σαν δευτερεύον κριτήριο η ταχύτητα εντοπισμού της λύσης. Στα αποτελέσματα οι μέθοδοι του Γενετικού Αλγορίθμου συμβολίζονται ως GA ενώ ακολουθεί αντίστοιχα το Binary/Real στην περίπτωση δυαδικής ή πραγματικής αναπαράστασης καθώς και το rw/t για τις μεθόδους επιλογής roulette wheel/tournament αντίστοιχα. Επίσης η Particle Swarm Optimization συμβολίζεται ως PSO και ακολουθεί το είδος του μοντέλου ως gbest/lbest. Οι πίνακες στατιστικών των μεθόδων έχουν στρογγυλοποιηθεί στο τρίτο δεκαδικό ψηφίο.

#### 3.1 Αποτελέσματα για πλήθος ατόμων $N = 4$

Στο [boxplot](#) που ακολουθεί παρατίθενται οι λύσεις των μεθόδων για τα 30 πειράματα που πραγματοποιήθηκαν. Είναι εμφανές ότι η μέθοδος του γενετικού αλγορίθμου με πραγματική αναπαράσταση καθώς και οι μέθοδοι PSO πέτυχαν τα καλύτερα αποτελέσματα καθώς στην μέση περίπτωση ([πίνακας](#)) βρέθηκε το ολικό ελάχιστο της συνάρτησης, σύμφωνα με τους πίνακες της επίσημης [σελίδας](#). Αυτό πρακτικά σημαίνει ότι βρέθηκε η δομή των ατόμων που ελαχιστοποιούν το δυναμικό Lennard-Jones.

Στην συνέχεια πραγματοποιήθηκε στατιστικός έλεγχος με τεστ Wilcoxon ώστε να διαπιστωθεί εάν οι κατανομές των δειγμάτων των λύσεων έχουν στατιστικά σημαντική διαφορά μεταξύ τους. Τα αποτελέσματα δείχνουν ότι όλες οι μέθοδοι έχουν στατιστικά σημαντική διαφορά μεταξύ τους ([πίνακας](#)), ωστόσο είναι τόσο μικρή που δεν είναι αντιληπτή λόγω στρογγυλοποίησης των αποτελεσμάτων τους στους αντίστοιχους πίνακες.

Σαν δεύτερο κριτήριο έχει χρησιμοποιηθεί η ταχύτητα εύρεσης της λύσης. Εκ των τριών καλύτερων μεθόδων ταχύτερη είναι η PSO με μοντέλο το global best ([πίνακας](#)). Η διαφορά τους είναι στατιστικά σημαντική συγκρινόμενη με την άλλη εκδοχή της PSO αλλά όχι με τον γενετικό αλγόριθμο ([πίνακας](#)). Με βάση τα αποτελέσματα οδηγούμαστε στο συμπέρασμα ότι η καλύτερη μέθοδος για την επίλυση του συγκεκριμένου προβλήματος είναι η PSO με μοντέλο global best, ως πιο γρήγορη, καθώς οι δειγματοληπτικές μέσες τιμές και των τριών έχουν πετύχει το ολικό ελάχιστο.



Σχήμα 1: Boxplot λύσεων των μεθόδων για  $N = 4$ .

Method	Mean	Median	St.Dev.	Min	Max
GA_Binary_rw	-5.496	-5.5	0.144	-5.762	-5.238
GA_Binary_t	-5.981	-5.983	0.009	-5.994	-5.96
GA_Real_t	-6	-6	0	-6	-5.999
PSO_gbest	-6	-6	0	-6	-5.999
PSO_lbest	-6	-6	0	-6	-5.999

Πίνακας 4: Στατιστικά λύσεων των μεθόδων για  $N = 4$ .

	GAB_rw	GAB_t	GAR_t	PSO_gbest	PSO_lbest
GAB_rw	-	1.73e-6	1.73e-6	1.73e-6	1.73e-6
GAB_t	-	-	1.73e-6	1.73e-6	1.73e-6
GAR_t	-	-	-	1.73e-6	1.89e-4
PSO_gbest	-	-	-	-	2.84e-5

Πίνακας 5: Wilcoxon Test Pvalues με κριτήριο τις ποιότητες των λύσεων για  $N = 4$ .

Method	Mean	Median	St.Dev.	Min	Max
GA_Binary_rw	291566	307000	79236	98000	396000
GA_Binary_t	303333	309000	67882	77000	400000
GA_Real_t	341366	367500	67098	87000	400000
PSO_gbest	318766	310000	53339	218000	400000
PSO_lbest	397533	399000	3234	390000	400000

Πίνακας 6: Στατιστικά της ταχύτητας εύρεσης των λύσεων για  $N = 4$ .



	GAB_rw	GAB_t	GAR_t	PSO_gbest	PSO_lbest
GAB_rw	-	0.688	0.018	0.171	1.73e-6
GAB_t	-	-	0.015	0.443	2.1e-6
GAR_t	-	-	-	0.0999	1e-5
PSO_gbest	-	-	-	-	4.2e-6

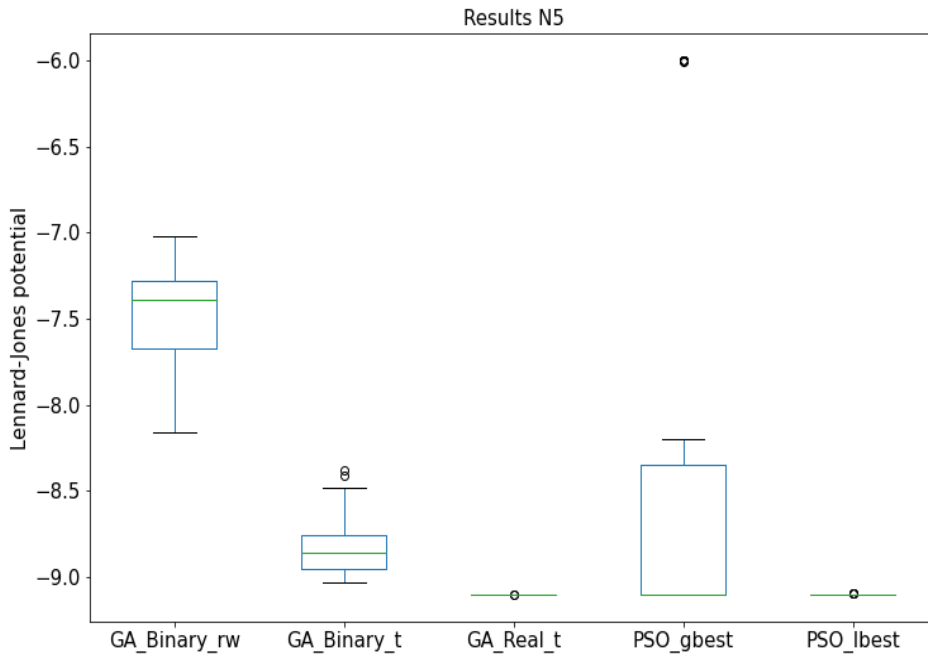
Πίνακας 7: Wilcoxon's Test Pvalues με κριτήριο την ταχύτητας εύρεσης των λύσεων για  $N = 4$ .

### 3.2 Αποτελέσματα για πλήθος ατόμων $N = 5$

Στο [boxplot](#) που ακολουθεί παραθέτονται οι λύσεις των μεθόδων για τα 30 πειράματα που πραγματοποιήθηκαν. Ο γενετικός αλγόριθμος με πραγματική αναπαράσταση καθώς και η μέθοδος PSO με μοντέλο local best πέτυχαν τα καλύτερα αποτελέσματα καθώς στην μέση περίπτωση ([πίνακας](#)) βρήκαν το ολικό ελάχιστο της συνάρτησης καθώς και οι διακυμάνσεις τους είναι σχεδόν μηδενικές.

Ακολουθεί ο στατιστικός έλεγχος με τεστ Wilcoxon στις κατανομές των δειγμάτων των λύσεων στον [πίνακα](#). Ο γενετικός αλγόριθμος με πραγματική αναπαράσταση και με την μέθοδο PSO local best έχει στατιστικά σημαντικές διαφορές καθώς το  $pvalue = 1.92e-6$ . Εκ των δύο η καλύτερη θεωρείται αυτή του γενετικού αλγορίθμου καθώς έχει τη χαμηλότερη μέση τιμή και τη μικρότερη διακύμανση στα αποτελέσματα της.

Σαν δεύτερο κριτήριο έχει χρησιμοποιηθεί η ταχύτητα εύρεσης της λύσης. Εκ των δύο καλύτερων μεθόδων ταχύτερη είναι ο γενετικός αλγόριθμος ([πίνακας](#)). Ωστόσο η διαφορά τους δεν είναι στατιστικά σημαντική ([πίνακας](#)). Με βάση τα αποτελέσματα οδηγούμαστε στο συμπέρασμα ότι η καλύτερη μέθοδος για το συγκεκριμένο πρόβλημα είναι ο γενετικός αλγόριθμος με πραγματική αναπαράσταση.



Σχήμα 2: Boxplot λύσεων των μεθόδων για  $N = 5$ .

Method	Mean	Median	St.Dev.	Min	Max
GA_Binary_rw	-7.472	-7.392	0.282	-8.160	-7.019
GA_Binary_t	-8.821	-8.860	0.176	-9.032	-8.376
GA_Real_t	-9.104	-9.104	0	-9.104	-9.104
PSO_gbest	-8.409	-9.104	1.245	-9.105	-6
PSO_lbest	-9.102	-9.103	0.003	-9.104	-9.092

Πίνακας 8: Στατιστικά των λύσεων για  $N = 5$ .

	GAB_rw	GAB_t	GAR_t	PSO_gbest	PSO_lbest
GAB_rw	-	1.73e-6	1.28e-3	1.73e-6	1.73e-6
GAB_t	-	-	1.73e-6	0.91	1.73e-6
GAR_t	-	-	-	0.033	1.92e-6
PSO_gbest	-	-	-	-	0.339

Πίνακας 9: Wilcoxon's Test Pvalues με κριτήριο τις ποιότητες των λύσεων για  $N = 5$ .

Method	Mean	Median	St.Dev.	Min	Max
GA_Binary_rw	379533	405000	97052	160000	496000
GA_Binary_t	420900	437000	59776	300000	498000
GA_Real_t	488166	498500	19138	429000	500000
PSO_gbest	494466	500000	22335	381000	500000
PSO_lbest	497500	499000	2933	491000	500000

Πίνακας 10: Στατιστικά της ταχύτητας εύρεσης των λύσεων για  $N = 5$ .

	GAB_rw	GAB_t	GAR_t	PSO_gbest	PSO_lbest
GAB_rw	-	0.128	5.7e-6	1.1e-5	2e-6
GAB_t	-	-	1.24e-5	5.74e-6	1.73e-6
GAR_t	-	-	-	0.014	0.094
PSO_gbest	-	-	-	-	0.039

Πίνακας 11: Wilcoxon's Test Pvalues με κριτήριο την ταχύτητας εύρεσης των λύσεων για  $N = 5$ .

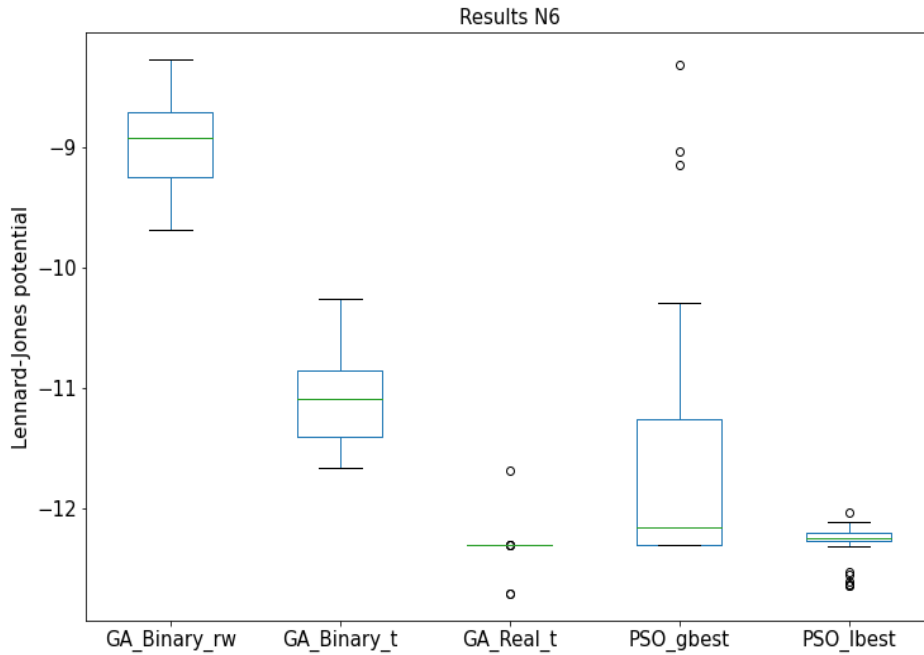
### 3.3 Αποτελέσματα για πλήθος ατόμων $N = 6$

Στο [boxplot](#) που ακολουθεί παρατίθενται οι λύσεις των μεθόδων για τα 30 πειράματα που πραγματοποιήθηκαν. Ο γενετικός αλγόριθμος με πραγματική αναπαράσταση καθώς και η μέθοδος PSO με μοντέλο local best πέτυχαν τα καλύτερα αποτελέσματα καθώς στην μέση περίπτωση ([πίνακας](#)) βρήκαν το ολικό ελάχιστο της συνάρτησης και οι διακυμάνσεις τους είναι μικρότερες των υπόλοιπων μεθόδων.

Ακολουθεί ο στατιστικός έλεγχος με τεστ Wilcoxon στις κατανομές των δειγμάτων των λύσεων στον [πίνακα](#). Ο γενετικός αλγόριθμος με πραγματική αναπαράσταση και με την μέθοδο PSO local best δεν έχουν στατιστικά σημαντική διαφορά καθώς το  $pvalue = 0.245$ . Εκ των δύο η καλύτερη θεωρείται αυτή του γενετικού αλγορίθμου καθώς έχει τη χαμηλότερη μέση τιμή και τη μικρότερη διακύμανση στα αποτελέσματα της.

Σαν δεύτερο κριτήριο έχει χρησιμοποιηθεί η ταχύτητα εύρεσης της λύσης. Εκ των δύο καλύτερων μεθόδων ταχύτερη είναι ο γενετικός αλγόριθμος ([πίνακας](#)). Ωστόσο η διαφορά τους δεν

είναι στατιστικά σημαντική (πίνακας). Με βάση τα ανωτέρω οδηγούμαστε στο συμπέρασμα ότι η καλύτερη μέθοδος για το συγκεκριμένο πρόβλημα είναι ο γενετικός αλγόριθμος με πραγματική αναπαράσταση.



Σχήμα 3: Boxplot λύσεων των μεθόδων για  $N = 6$ .

Method	Mean	Median	St.Dev.	Min	Max
GA_Binary_rw	-8.945	-8.922	0.396	-9.687	-8.263
GA_Binary_t	-11.062	-11.088	0.391	-11.659	-10.259
GA_Real_t	-12.31	-12.303	0.156	-12.712	-11.691
PSO_gbest	-11.537	-12.162	1.110	-12.303	-8.313
PSO_lbest	-12.292	-12.245	0.169	-12.648	-12.037

Πίνακας 12: Στατιστικά των λύσεων για  $N = 6$ .

	GAB_rw	GAB_t	GAR_t	PSO_gbest	PSO_lbest
GAB_rw	-	1.73e-6	1.73e-6	2.13e-6	1.73e-6
GAB_t	-	-	1.73e-6	0.018	1.73e-6
GAR_t	-	-	-	1e-5	0.245
PSO_gbest	-	-	-	-	3.9e-4

Πίνακας 13: Wilcoxon's Test Pvalues με κριτήριο τις ποιότητες των λύσεων για  $N = 6$ .

Method	Mean	Median	St.Dev.	Min	Max
GA_Binary_rw	434700	464500	127806	85000	593000
GA_Binary_t	511266	532000	75092	297000	598000
GA_Real_t	593866	600000	16336	541000	600000
PSO_gbest	599466	600000	2556	586000	600000
PSO_lbest	598266	600000	2923	589000	600000

Πίνακας 14: Στατιστικά της ταχύτητας εύρεσης των λύσεων για  $N = 6$ .

	GAB_rw	GAB_t	GAR_t	PSO_gbest	PSO_lbest
GAB_rw	-	0.018	3.18e−6	1.73e−6	1.73e−6
GAB_t	-	-	6.33e−6	1.73e−6	1.92e−6
GAR_t	-	-	-	0.092	0.732
PSO_gbest	-	-	-	-	0.017

Πίνακας 15: Wilcoxon's Test Pvalues με κριτήριο την ταχύτητας εύρεσης των λύσεων για  $N = 6$ .

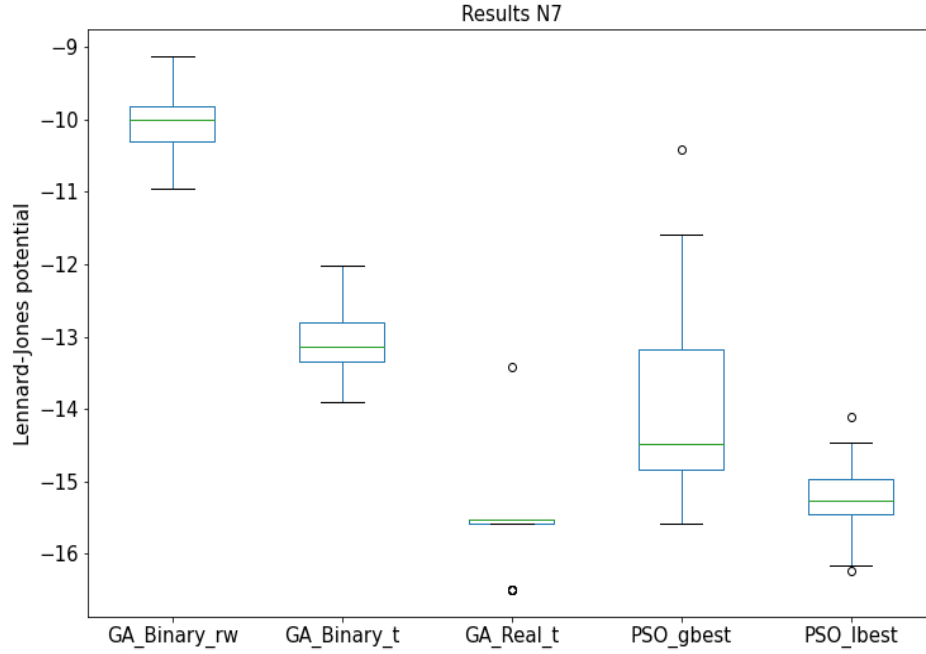
### 3.4 Αποτελέσματα για πλήθος ατόμων $N = 7$

Στο [boxplot](#) που ακολουθεί παρατίθενται οι λύσεις των μεθόδων για τα 30 πειράματα που πραγματοποιήθηκαν. Ο γενετικός αλγόριθμος με πραγματική αναπαράσταση και με τη μέθοδο PSO με μοντέλο local best πέτυχαν τα καλύτερα αποτελέσματα καθώς είναι αυτές με την μικρότερη μέση τιμή ([πίνακας](#)) και επίσης βρίσκονται κοντά στο ολικό ελάχιστο. Ο γενετικός αλγόριθμος πραγματικής αναπαράστασης κατάφερε να πετύχει τον ολικό ελαχιστοποιητή της συνάρτησης τουλάχιστον μία φορά όπως βλέπουμε στην στήλη min του στατιστικού πίνακα.

Ακολουθεί ο στατιστικός έλεγχος με τεστ Wilcoxon στις κατανομές των δειγμάτων των λύσεων στον [πίνακα](#). Ο γενετικός αλγόριθμος με πραγματική αναπαράσταση και με τη μέθοδο PSO local best έχουν στατιστικά σημαντική διαφορά καθώς το  $pvalue = 1.11e-3$ . Εκ των δύο η βέλτιστη θεωρείται αυτή του γενετικού αλγορίθμου καθώς έχει μικρότερη μέση τιμή.

Σαν δεύτερο κριτήριο έχει χρησιμοποιηθεί η ταχύτητα εύρεσης της λύσης. Οι δύο μέθοδοι δεν έχουν στατιστικά σημαντικές διαφορές στην ταχύτητα τους. Με βάση τα παραπάνω οδηγούμεστε στο συμπέρασμα ότι η καλύτερη μέθοδος για το συγκεκριμένο πρόβλημα είναι ο γενετικός αλγόριθμος με πραγματική αναπαράσταση.

Τα αποτελέσματα των πειραμάτων για πλήθος ατόμων με  $N = 8, 9, 10, 15, 20$  είναι ίδια με αυτά αυτού του συγκεκριμένου πειράματος και για αυτό και ως προς τη μέθοδο δεν τίθεται σκόπιμη η περαιτέρω ανάλυση τους. Η γραφική απεικόνιση των αποτελεσμάτων και η στατιστική ανάλυση παρατίθεται στις ενότητες που ακολουθούν.



Σχήμα 4: Boxplot λύσεων των μεθόδων για  $N = 7$ .

Method	Mean	Median	St.Dev.	Min	Max
GA_Binary_rw	-10.06	-9.992	0.424	-10.95	-9.121
GA_Binary_t	-13.056	-13.133	0.47	-13.912	-12.019
GA_Real_t	-15.665	-15.533	0.577	-16.505	-13.415
PSO_gbest	-14.001	-14.489	1.29	-15.586	-10.405
PSO_lbest	-15.224	-15.267	0.473	-16.246	-14.104

Πίνακας 16: Στατιστικά των λύσεων για  $N = 7$ .

	GAB_rw	GAB_t	GAR_t	PSO_gbest	PSO_lbest
GAB_rw	-	1.73e-6	1.73e-6	1.73e-6	1.73e-6
GAB_t	-	-	1.92e-6	1.71e-3	1.73e-6
GAR_t	-	-	-	1.73e-6	1.11e-3
PSO_gbest	-	-	-	-	1.8e-5

Πίνακας 17: Wilcoxon's Test Pvalues με κριτήριο τις ποιότητες των λύσεων για  $N = 7$ .

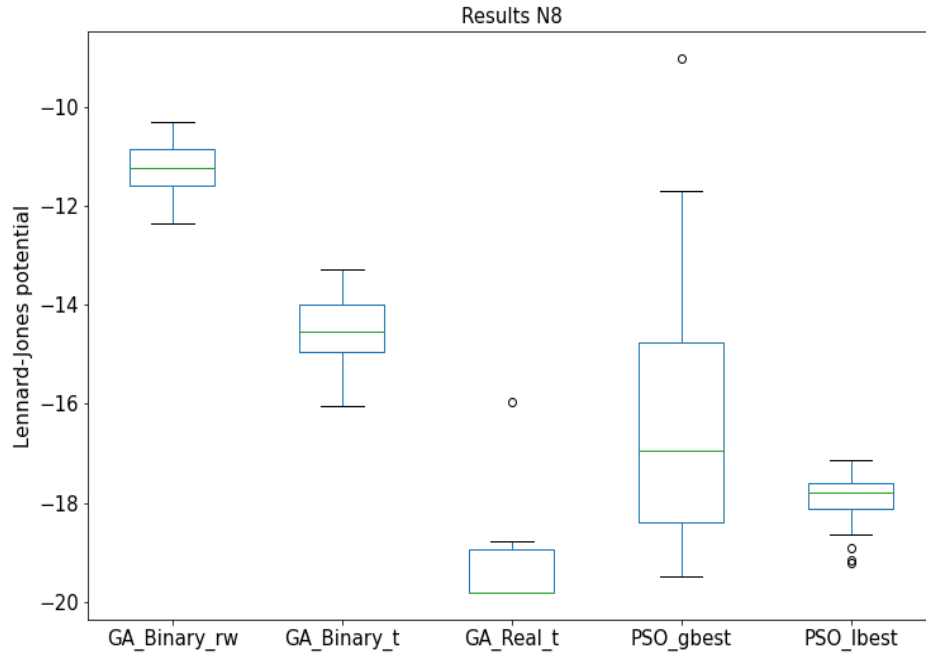
Method	Mean	Median	St.Dev.	Min	Max
GA_Binary_rw	448233	471000	171788	96000	685000
GA_Binary_t	521433	525000	116132	186000	694000
GA_Real_t	697600	700000	6483	674000	700000
PSO_gbest	700000	700000	0	700000	700000
PSO_lbest	697200	698000	3652	687000	700000

Πίνακας 18: Στατιστικά της ταχύτητας εύρεσης των λύσεων για  $N = 7$ .

	GAB_rw	GAB_t	GAR_t	PSO_gbest	PSO_lbest
GAB_rw	-	0.066	1.73e-6	1.73e-6	1.73e-6
GAB_t	-	-	1.73e-6	1.73e-6	1.92e-6
GAR_t	-	-	-	0.042	0.101
PSO_gbest	-	-	-	-	8.27e-5

Πίνακας 19: Wilcoxon's Test Pvalues με κριτήριο την ταχύτητας εύρεσης των λύσεων για  $N = 7$ .

### 3.5 Αποτελέσματα για πλήθος ατόμων $N = 8$



Σχήμα 5: Boxplot λύσεων των μεθόδων για  $N = 8$ .

Method	Mean	Median	St.Dev.	Min	Max
GA_Binary_rw	-11.227	-11.242	0.517	-12.348	-10.291
GA_Binary_t	-14.456	-14.532	0.722	-16.028	-13.27
GA_Real_t	-19.4	-19.821	0.785	-19.821	-15.954
PSO_gbest	-16.368	-16.939	2.581	-19.481	-9.011
PSO_lbest	-17.936	-17.8	0.537	-19.225	-17.127

Πίνακας 20: Στατιστικά των λύσεων για  $N = 8$ .

	GAB_rw	GAB_t	GAR_t	PSO_gbest	PSO_lbest
GAB_rw	-	1.73e-6	1.73e-6	2.6e-6	1.73e-6
GAB_t	-	-	1.73e-6	1.04e-3	1.73e-6
GAR_t	-	-	-	5.75e-6	3.88e-6
PSO_gbest	-	-	-	-	3.61e-3

Πίνακας 21: Wilcoxon's Test Pvalues με κριτήριο τις ποιότητες των λύσεων για  $N = 8$ .

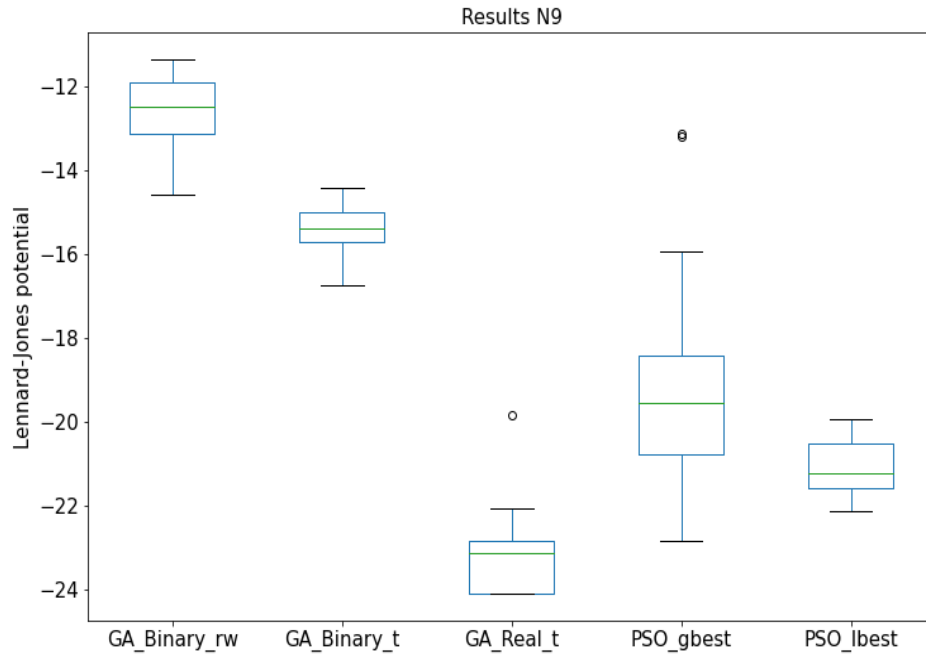
Method	Mean	Median	St.Dev.	Min	Max
GA_Binary_rw	515266	563500	208540	115000	772000
GA_Binary_t	603033	617500	129925	230000	794000
GA_Real_t	799033	800000	2697	788000	800000
PSO_gbest	798000	800000	9273	749000	800000
PSO_lbest	798000	800000	3912	784000	800000

Πίνακας 22: Στατιστικά της ταχύτητας εύρεσης των λύσεων για  $N = 8$ .

	GAB_rw	GAB_t	GAR_t	PSO_gbest	PSO_lbest
GAB_rw	-	0.088	1.73e-6	1.73e-6	1.73e-6
GAB_t	-	-	1.73e-6	1.73e-6	1.73e-6
GAR_t	-	-	-	0.661	0.185
PSO_gbest	-	-	-	-	0.11

Πίνακας 23: Wilcoxon's Test Pvalues με κριτήριο την ταχύτητας εύρεσης των λύσεων για  $N = 8$ .

### 3.6 Αποτελέσματα για πλήθος ατόμων $N = 9$



Σχήμα 6: Boxplot λύσεων των μεθόδων για  $N = 9$ .

Method	Mean	Median	St.Dev.	Min	Max
GA_Binary_rw	-12.66	-12.491	0.903	-14.596	-11.367
GA_Binary_t	-15.464	-15.397	0.6289	-16.756	-14.442
GA_Real_t	-23.2067	-23.134	0.98	-24.113	-19.847
PSO_gbest	-19.365	-19.547	2.277	-22.856	-13.134
PSO_lbest	-21.088	-21.25	0.649	-22.133	-19.931

Πίνακας 24: Στατιστικά των λύσεων για  $N = 9$ .

	GAB_rw	GAB_t	GAR_t	PSO_gbest	PSO_lbest
GAB_rw	-	1.73e-6	1.73e-6	1.73e-6	1.73e-6
GAB_t	-	-	1.73e-6	5.75e-6	1.73e-6
GAR_t	-	-	-	3.52e-6	5.22e-6
PSO_gbest	-	-	-	-	5.3e-4

Πίνακας 25: Wilcoxon's Test Pvalues με κριτήριο τις ποιότητες των λύσεων για  $N = 9$ .

Method	Mean	Median	St.Dev.	Min	Max
GA_Binary_rw	662733	682000	153144	308000	880000
GA_Binary_t	700333	719500	124653	349000	900000
GA_Real_t	897533	900000	7181	864000	900000
PSO_gbest	899933	900000	253	899000	900000
PSO_lbest	897900	899000	3835	885000	900000

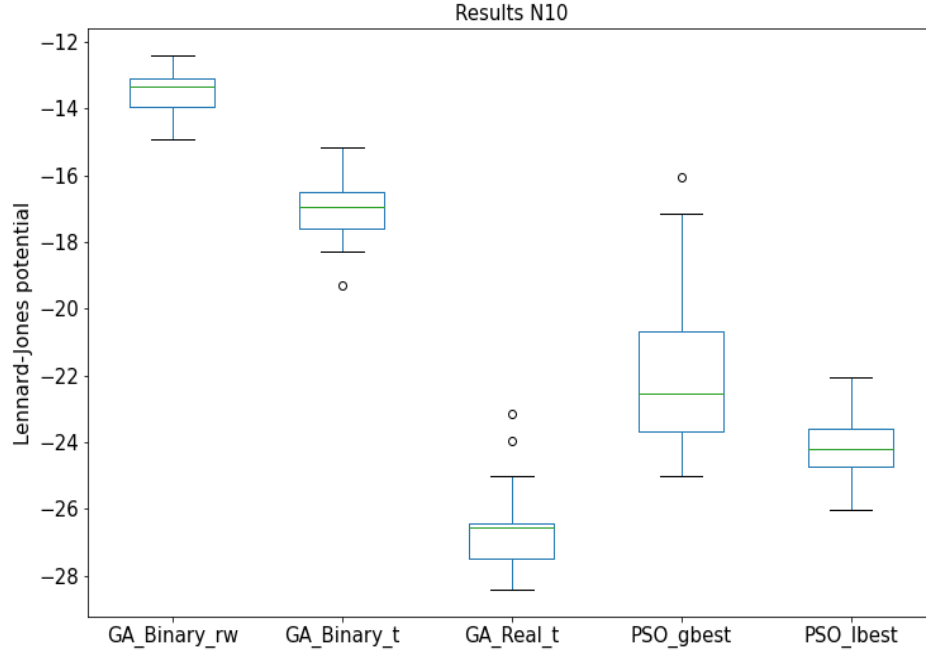
Πίνακας 26: Στατιστικά της ταχύτητας εύρεσης των λύσεων για  $N = 9$ .

	GAB_rw	GAB_t	GAR_t	PSO_gbest	PSO_lbest
GAB_rw	-	0.465	1.73e-6	1.73e-6	1.73e-6
GAB_t	-	-	2.56e-6	2.56e-6	2.12e-6
GAR_t	-	-	-	0.026	0.406
PSO_gbest	-	-	-	-	5.5e-5

Πίνακας 27: Wilcoxon's Test Pvalues με κριτήριο την ταχύτητας εύρεσης των λύσεων για  $N = 9$ .



### 3.7 Αποτελέσματα για πλήθος ατόμων $N = 10$



Σχήμα 7: Boxplot λύσεων των μεθόδων για  $N = 10$ .

Method	Mean	Median	St.Dev.	Min	Max
GA_Binary_rw	-13.497	-13.347	0.629	-14.909	-12.396
GA_Binary_t	-17.033	-16.933	0.885	-19.274	-15.17
GA_Real_t	-26.531	-26.558	1.127	-28.422	-23.163
PSO_gbest	-22.013	-22.553	2.177	-24.987	-16.032
PSO_lbest	-24.062	-24.179	0.936	-26.0126	-22.0611

Πίνακας 28: Στατιστικά των λύσεων για  $N = 10$ .

	GAB_rw	GAB_t	GAR_t	PSO_gbest	PSO_lbest
GAB_rw	-	1.73e-6	1.73e-6	1.73e-6	1.73e-6
GAB_t	-	-	1.73e-6	2.6e-6	1.73e-6
GAR_t	-	-	-	1.92e-4	4.73e-6
PSO_gbest	-	-	-	-	1.6e-4

Πίνακας 29: Wilcoxon's Test Pvalues με κριτήριο τις ποιότητες των λύσεων για  $N = 10$ .

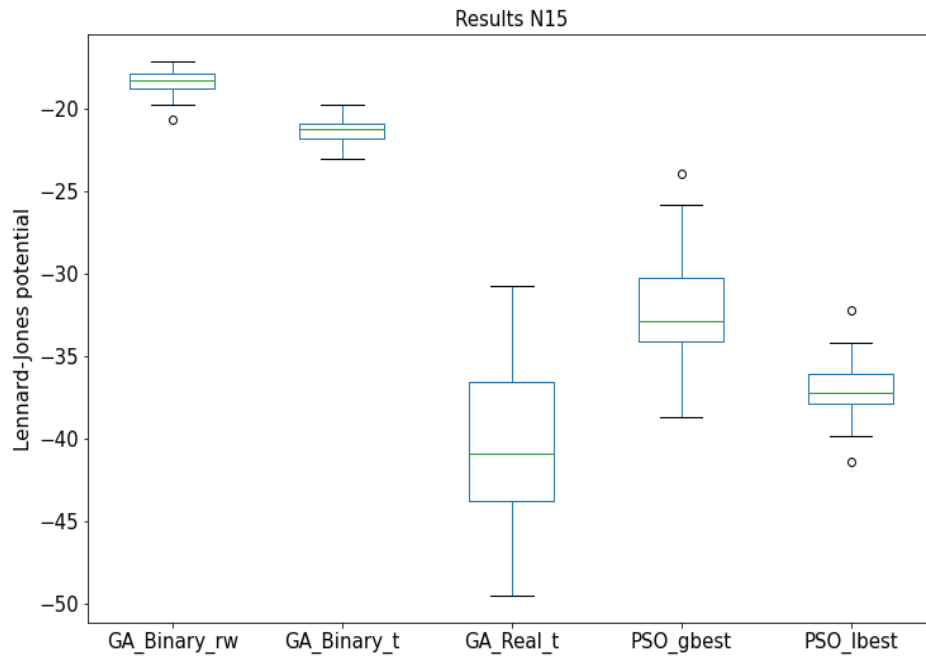
Method	Mean	Median	St.Dev.	Min	Max
GA_Binary_rw	682500	719000	247140	164000	996000
GA_Binary_t	718400	796000	235180	159000	987000
GA_Real_t	999266	1000000	2753	985000	1000000
PSO_gbest	999966	1000000	182	999000	1000000
PSO_lbest	998700	999000	1235	996000	1000000

Πίνακας 30: Στατιστικά της ταχύτητας εύρεσης των λύσεων για  $N = 10$ .

	GAB_rw	GAB_t	GAR_t	PSO_gbest	PSO_lbest
GAB_rw	-	0.673	1.73e-6	1.73e-6	1.73e-6
GAB_t	-	-	1.73e-6	1.73e-6	1.73e-6
GAR_t	-	-	-	0.047	0.007
PSO_gbest	-	-	-	-	6.81e-5

Πίνακας 31: Wilcoxon's Test Pvalues με κριτήριο την ταχύτητας εύρεσης των λύσεων για  $N = 10$ .

### 3.8 Αποτελέσματα για πλήθος ατόμων $N = 15$



Σχήμα 8: Boxplot λύσεων των μεθόδων για  $N = 15$ .

Method	Mean	Median	St.Dev.	Min	Max
GA_Binary_rw	-18.448	-18.311	0.827	-20.705	-17.152
GA_Binary_t	-21.406	-21.247	0.766	-23.102	-19.802
GA_Real_t	-40.3126	-40.921	4.721	-49.566	-30.797
PSO_gbest	-32.123	-32.871	3.389	-38.747	-23.979
PSO_lbest	-37.002	-37.2112	1.881	-41.434	-32.245

Πίνακας 32: Στατιστικά των λύσεων για  $N = 15$ .

	GAB_rw	GAB_t	GAR_t	PSO_gbest	PSO_lbest
GAB_rw	-	1.73e-6	1.73e-6	1.73e-6	1.73e-6
GAB_t	-	-	1.73e-6	1.73e-6	1.73e-6
GAR_t	-	-	-	7.69e-6	1.83e-3
PSO_gbest	-	-	-	-	2.88e-6

Πίνακας 33: Wilcoxon's Test Pvalues με κριτήριο τις ποιότητες των λύσεων για  $N = 15$ .

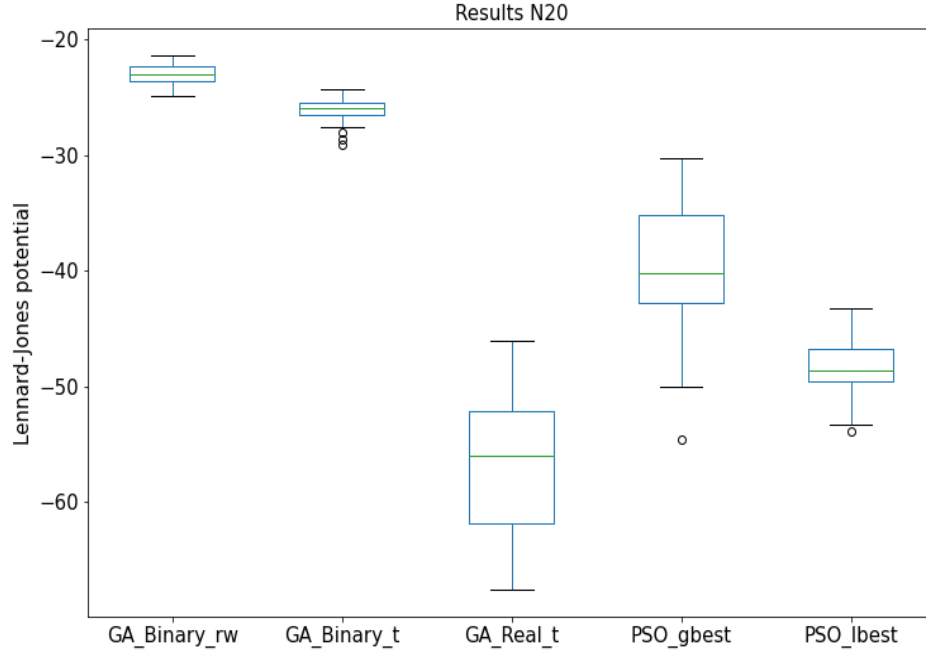
Method	Mean	Median	St.Dev.	Min	Max
GA_Binary_rw	946733	999000	328535	140000	1401000
GA_Binary_t	916500	943500	311664	255000	1394000
GA_Real_t	1498300	1499000	3852	1479000	1500000
PSO_gbest	1499933	1500000	253	1499000	1500000
PSO_lbest	1497800	1498500	2708	1491000	1500000

Πίνακας 34: Στατιστικά της ταχύτητας εύρεσης των λύσεων για  $N = 15$ .

	GAB_rw	GAB_t	GAR_t	PSO_gbest	PSO_lbest
GAB_rw	-	0.789	1.73e-6	1.73e-6	1.73e-6
GAB_t	-	-	1.73e-6	1.73e-6	1.73e-6
GAR_t	-	-	-	5.67e-5	0.244
PSO_gbest	-	-	-	-	1.7e-5

Πίνακας 35: Wilcoxon's Test Pvalues με κριτήριο την ταχύτητας εύρεσης των λύσεων για  $N = 15$ .

### 3.9 Αποτελέσματα για πλήθος ατόμων $N = 20$



Σχήμα 9: Boxplot λύσεων των μεθόδων για  $N = 20$ .

Method	Mean	Median	St.Dev.	Min	Max
GA_Binary_rw	-22.949	-22.951	0.969	-24.915	-21.354
GA_Binary_t	-26.116	-25.96	1.101	-29.046	-24.245
GA_Real_t	-56.759	-55.992	5.780	-67.595	-46.007
PSO_gbest	-39.836	-40.165	5.262	-54.593	-30.246
PSO_lbest	-48.422	-48.653	2.361	-53.865	-43.22

Πίνακας 36: Στατιστικά των λύσεων για  $N = 20$ .

	GAB_rw	GAB_t	GAR_t	PSO_gbest	PSO_lbest
GAB_rw	-	1.17e-6	1.17e-6	1.17e-6	1.17e-6
GAB_t	-	-	1.17e-6	1.17e-6	1.17e-6
GAR_t	-	-	-	1.3e-6	2.82e-6
PSO_gbest	-	-	-	-	5e-6

Πίνακας 37: Wilcoxon's Test Pvalues με κριτήριο τις ποιότητες των λύσεων για  $N = 20$ .

Method	Mean	Median	St.Dev.	Min	Max
GA_Binary_rw	1289233	1348000	414877	527000	1912000
GA_Binary_t	1169500	1296500	478946	180000	1874000
GA_Real_t	1997166	1998500	3312	1988000	2000000
PSO_gbest	1999766	2000000	504	1998000	2000000
PSO_lbest	1998433	1999000	2284	1990000	2000000

Πίνακας 38: Στατιστικά της ταχύτητας εύρεσης των λύσεων για  $N = 20$ .

	GAB_rw	GAB_t	GAR_t	PSO_gbest	PSO_lbest
GAB_rw	-	0.189	1.73e−6	1.17e−6	1.17e−6
GAB_t	-	-	1.17e−6	1.17e−6	1.17e−6
GAR_t	-	-	-	9.91e−5	0.037
PSO_gbest	-	-	-	-	6.7e−5

Πίνακας 39: Wilcoxon's Test Pvalues με κριτήριο την ταχύτητας εύρεσης των λύσεων για  $N = 20$ .

### 3.10 Συμπεράσματα

Μετά την συγκεντρωτική ανάλυση των αποτελεσμάτων οδηγούμαστε στο συμπέρασμα ότι στο συγκεκριμένο πρόβλημα η καλύτερη μέθοδος είναι αυτή του γενετικού αλγορίθμου με πραγματική αναπαράσταση. Αυτό είναι ιδιαίτερα εμφανές καθώς αυξάνεται το πλήθος των ατόμων. Ωστόσο σημαντικό ρόλο στα αποτελέσματα έχουν οι επιλογές που έγιναν στις παραμέτρους των αλγορίθμων. Το σύνολο των παραμέτρων που επιλέχθηκαν ελέγχθηκε αρχικά ως προς την βέλτιστή τους συμπεριφορά στο πρόβλημα και έπειτα αξιοποιήθηκαν για την διενέργεια των πειραμάτων.

Επίσης κρίνεται σκόπιμο να αναφερθεί ότι καθώς το πλήθος των ατόμων αυξάνεται οι μέθοδοι απέχουν περισσότερο από το ολικό ελάχιστο. Μία πιθανή αιτία για αυτό είναι ότι καθώς ο χώρος αναζήτησης αυξάνεται λόγω της μεγαλύτερης διάστασης, ίσως η επιλογή μεγαλύτερου πληθυσμού/σμήνους θα επέφερε καλύτερα αποτελέσματα. Υπάρχει δηλαδή το trade off ανάμεσα στο μέγεθος του πληθυσμού/σμήνους και στις επαναλήψεις του αλγορίθμου. Με μεγαλύτερο πληθυσμό/σμήνος υπάρχει η δυνατότητα εξερεύνησης μεγαλύτερης περιοχής του χώρου αναζήτησης με κόστος τη διενέργεια λιγότερων επαναλήψεων, ενώ με περισσότερες επαναλήψεις ο αλγόριθμος θα τρέξει περισσότερες εποχές με κόστος την εξερεύνηση μικρότερης περιοχής του χώρου αναζήτησης.

## 4 Δομή αναπτυσσόμενου κώδικα

Στην παρούσα ενότητα περιγράφεται η λειτουργία του κώδικα που αναπτύχθηκε για την τρέχουσα εργασία. Ακολουθεί μια σύντομη περιγραφή των μεθόδων και των συναρτήσεων που χρησιμοποιήθηκαν.

- **main():** Το πρόγραμμα εκκινεί μέσα από τη συνάρτηση main. Στο αρχείο seeds.csv υπάρχουν 30 αριθμοί(seeds) που τα χρησιμοποιούνται για την αρχικοποίηση της γεννήτριας τυχαίων αριθμών σε κάθε πείραμα. Σε αυτή την συνάρτηση αρχικοποιείται ο χώρος αναζήτησης  $U = [-2.5, 2.5]$  και εκκινούν οι μέθοδοι. Στην συνέχεια τα αποτελέσματα των μεθόδων αποθηκεύονται σε ένα αρχείο excel. Για κάθε μέθοδο αποθηκεύεται η ελάχιστη συναρτησιακή τιμή καθώς και η επανάληψη που βρέθηκε η τιμή αυτή.

- **Genetic\_Algorithm\_Binary(Umin, Umax, number\_of\_atoms, selection\_method):** η συνάρτηση υλοποιεί τον γενετικό αλγόριθμο αναζήτησης με δυαδική αναπαράσταση. Δέχεται σαν ορίσματα το διάστημα αναζήτησης, το πλήθος των ατόμων καθώς και την μέθοδο επιλογής (roulette wheel, tournament selection). Επιστρέφει την ελάχιστη συναρτησιακή τιμή που υπολόγισε καθώς και σε ποια επανάληψη την πέτυχε ενώ οι υπόλοιπες επιστρεφόμενες τιμές δεν αξιοποιούνται στην πράξη.
- **Genetic\_Algorithm\_Real(Umin, Umax, number\_of\_atoms, selection\_method):** υλοποιεί τον γενετικό αλγόριθμο αναζήτησης με πραγματική αναπαράσταση. Δέχεται σαν ορίσματα το διάστημα αναζήτησης, το πλήθος των ατόμων καθώς και τη μέθοδο επιλογής (roulette wheel, tournament selection). Επιστρέφει την ελάχιστη συναρτησιακή τιμή που υπολόγισε καθώς και σε ποιά επανάληψη την πέτυχε ενώ οι υπόλοιπες επιστρεφόμενες τιμές δεν αξιοποιούνται στην πράξη.
- **Particle\_Swarm\_Optimization(Umin, Umax, number\_of\_atoms, model):** υλοποιεί, την μέθοδο νοημοσύνης σμηνών (PSO). Δέχεται σαν ορίσματα το διάστημα αναζήτησης, το πλήθος των ατόμων καθώς και το μοντέλο της γειτονιάς (global best, local best). Επιστρέφει την ελάχιστη συναρτησιακή τιμή που υπολόγισε καθώς και σε ποια επανάληψη την πέτυχε (οι υπόλοιπες επιστρεφόμενες τιμές δεν αξιοποιούνται στην πράξη).
- **evaluate\_population(population, number\_of\_atoms):** δέχεται σαν ορίσματα τον πληθυσμό/σμήνος και το πλήθος των ατόμων. Στο εσωτερικό της καλεί την συνάρτηση mylib.evaluate που γράφηκε σαν βιβλιοθήκη σε C. Επιστρέφει την συναρτησιακή τιμή κάθε μέλους του πληθυσμού/σμήνους, το δείκτη του καλύτερου σε ολόκληρο τον πληθυσμό/σμήνος καθώς και την ελάχιστη συναρτησιακή τιμή του.
- **roulette\_wheel\_selection(population, evaluations, selective\_pressure):** δέχεται σαν ορίσματα τον πληθυσμό/σμήνος, τις συναρτησιακές τιμές κάθε μέλους του πληθυσμού/σμήνους καθώς και την τιμή της επιλεκτικής πίεσης. Επιστρέφει τον επιλεγθέν πληθυσμό μετά την διαδικασία του roulette wheel.
- **tournament\_selection(population, evaluations, tournament\_size):** δέχεται σαν ορίσματα τον πληθυσμό/σμήνος, τις συναρτησιακές τιμές κάθε μέλους του πληθυσμού/σμήνους καθώς και το μέγεθος του τουρνουά. Επιστρέφει τον επιλεγθέν πληθυσμό μετά την διαδικασία του tournament selection.
- **new\_population\_top\_N(population, mutated\_population, population\_evaluations, mutated\_population\_evaluations):** δέχεται σαν ορίσματα τον αρχικό πληθυσμό, υπό μετάλλαξη πληθυσμό καθώς και τις συναρτησιακές του τιμές αντίστοιχα και επιστρέφει τα  $N$  καλύτερα άτομα με βάση αυτές.
- **calculate\_number\_of\_bits(Umin, Umax, error):** δέχεται σαν ορίσματα το διάστημα του χώρου αναζήτησης καθώς και το σφάλμα της λύσης και επιστρέφει το πλήθος των bits, που χρειάζεται η δυαδική αναπαράσταση.
- **calculate\_base\_10(binary\_number):** δέχεται σαν όρισμα έναν δυαδικό αριθμό και επιστρέφει τον αριθμό σε βάση του δέκα.
- **calculate\_number\_base\_10\_in\_feasible\_space(Umin, Umax, n\_bits, number\_base\_10):** δέχεται σαν ορίσματα τα όρια του χώρου αναζήτησης, το πλήθος των bits και έναν αριθμό σε βάση δέκα και επιστρέφει έναν αριθμό μέσα στο χώρο αναζήτησης.
- **decoder(population, ...):** δέχεται σαν ορίσματα τον πληθυσμό σε δυαδική αναπαράσταση, και τον επιστρέφει σε πραγματική αναπαράσταση.

- **initialize\_binary\_population(population\_size, number\_of\_atoms, dimensionality, n\_bits):** δέχεται σαν ορίσματα το μέγεθος του πληθυσμού, το πλήθος των ατόμων, το πλήθος των bits και επιστρέφει τον αρχικό τυχαίο δυαδικό πληθυσμό.
- **crossover\_binary\_population(selected\_population, crossover\_rate, crossover\_points):** δέχεται σαν ορίσματα τον επιλεγθέν δυαδικό πληθυσμό, τις παραμέτρους crossover\_rate, crossover\_points και επιστρέφει τον ανασυνδυασμένο δυαδικό πληθυσμό.
- **mutation\_binary\_population(crossover\_population, mutation\_rate):** δέχεται σαν όρισμα τον ανασυνδυασμένο δυαδικό πληθυσμό καθώς και την παράμετρο mutation\_rate και επιστρέφει τον μεταλλαγμένο δυαδικό πληθυσμό.
- **initialize\_real\_population(Umin, Umax, population\_size, number\_of\_atoms, dimensionality):** δέχεται σαν ορίσματα το χώρο αναζήτησης, το μέγεθος του πληθυσμού, το πλήθος των ατόμων και επιστρέφει τον αρχικό τυχαίο πραγματικό πληθυσμό (δεκαδική αναπαράσταση).
- **crossover\_real\_population(selected\_population, crossover\_rate, delta=0.25):** δέχεται σαν ορίσματα τον επιλεγθέν πραγματικό πληθυσμό, και τις παραμέτρους crossover\_rate, delta και επιστρέφει τον ανασυνδυασμένο πραγματικό πληθυσμό.
- **mutation\_real\_population(crossover\_population, mutation\_rate, ...):** δέχεται σαν όρισμα τον ανασυνδυασμένο πραγματικό πληθυσμό καθώς και την παράμετρο mutation\_rate και επιστρέφει τον μεταλλαγμένο πραγματικό πληθυσμό.
- **initialize\_velocity(swarm\_size, number\_of\_atoms, dimensionality, max\_velocity):** δέχεται σαν ορίσματα το μέγεθος του σμήνους, το πλήθος των ατόμων, και την μέγιστη επιτρεπτή ταχύτητα, και επιστρέφει τις αρχικές τυχαίες ταχύτητες του σμήνους.
- **create\_neighborhoods(swarm\_size, neighborhood\_radius):** δέχεται σαν όρισμα το μέγεθος του σμήνους και την ακτίνα της γειτονίας και επιστρέφει την τυπολογική γειτονιά του σμήνους.
- **update\_velocity(swarm, velocity, best\_positions, best\_positions\_evaluations, neighborhoods, c1=2.05, c2=2.05, x=0.729):** επιστρέφει τις νέες ταχύτητες του σμήνους.
- **update\_particles(swarm, velocity):** επιστρέφει τις νέες θέσεις του σμήνους.
- **check\_velocity\_bounds(velocity, max\_velocity):** δέχεται σαν ορίσματα τις ταχύτητες του σμήνους και την μέγιστη επιτρεπτή ταχύτητα. Επιστρέφει τις ταχύτητες του σμήνους σε επιτρεπτές τιμές.
- **check\_particles\_bounds(swarm, Umin, Umax):** δέχεται σαν ορίσματα το σμήνος και τα όρια του χώρου αναζήτησης, και το επιστρέφει σε επιτρεπτές τιμές του χώρου αναζήτησης.
- **update\_best\_positions(best\_positions, best\_positions\_evaluations, swarm, swarm\_evaluation):** επιστρέφει τις καινούργιες καλύτερες θέσεις του σμήνους.
- **lennard\_jones\_function(double \*atoms\_position, int n, double epsilon, double sigma):** δέχεται σαν ορίσματα τις θέσεις και το πλήθος των ατόμων επιστρέφει το δυναμικό των lennard-jones .
- **calculate\_dynamic\_energy(double euclidean\_distance, double sigma):** δέχεται σαν ορίσματα την ευκλείδεια απόσταση δύο ατόμων και επιστρέφει το δυναμικό τους.

- `calculate_euclidean_distance(double *atom_i, double *atom_j)`: δέχεται σαν ορίσματα την θέση δύο ατόμων και επιστρέφει την ευκλείδεια απόσταση τους..

## 5 Παρατηρήσεις για την ταχύτητα του κώδικα

Αρχικά όλος ο κώδικας κατασκευάστηκε σε γλώσσα προγραμματισμού Python. Κατά την διάρκεια των πειραμάτων παρατηρήθηκε μεγάλη καθυστέρηση εκτέλεσης του προγράμματος. Ύστερα από εξέταση των διαφορετικών μεθόδων διαπιστώθηκε ότι η συνάρτηση `evaluate` που περιγράφεται στο προηγούμενο κεφάλαιο ήταν η κύρια πηγή αυτή της καθυστέρησης. Για να αντιμετωπιστεί αυτό το πρόβλημα μεταφέρθηκε σε γλώσσα C αυτή η συνάρτηση μαζί με τις συναρτήσεις `lennard_jones_function()`, `calculate_dynamic_energy()`, `calculate_euclidean_distance()`. Στην συνέχεια έγινε χρήση της βιβλιοθήκης `OpenMP` ώστε να γίνει αυτό το κομμάτι του κώδικα παράλληλο. Ο χρόνος σε απλή python για να αξιολογηθεί σμήνος/πληθυσμός ίσο(ς) 1000 και αριθμός ατόμων ίσος με 50 χρειαζόταν κατά μέσο όρο 11 δευτερόλεπτα. Σε απλή C χωρίς την χρήση παραλληλίας χρειαζόταν κατά μέσο όρο 0.1 δευτερόλεπτα, ενώ και με παραλληλία 0.01 δευτερόλεπτα. Η διαδικασία αυτή έδωσε επιτάχυνση που ξεπερνάει την τιμή 1000 στην συγκεκριμένη συνάρτηση. Η μέθοδος της PSO local best και ο γενετικός αλγόριθμος με πραγματική αναπαράσταση είναι οι δύο μέθοδοι που έχουν κερδίσει περισσότερο σε ταχύτητα λόγω αυτής της μετατροπής του κώδικα καθώς δεν έχουν τόσο έντονες καθυστερήσεις από άλλες χρονοβόρες συναρτήσεις όπως ο decoder.

## 6 Οδηγίες για την εκτέλεση του κώδικα

Η υλοποίηση των μεθόδων έγινε σε γλώσσα Python 3.8. Οι βιβλιοθήκες που έχουν χρησιμοποιηθεί είναι οι `numpy`, `pandas`, `matplotlib`, `sys`, `scipy` και `ctypes`.

- Για τους μαθηματικούς υπολογισμούς χρησιμοποιήθηκε η βιβλιοθήκη `numpy`.
- Για την εγγραφή των αποτελεσμάτων σε αρχεία η βιβλιοθήκη `pandas`.
- Η βιβλιοθήκη `scipy` χρησιμοποιήθηκε για την πραγματοποίηση του Wilcoxon τεστ.
- Η βιβλιοθήκη `matplotlib` χρησιμοποιήθηκε για τα boxplots.
- Το module που χρησιμοποιήθηκε είναι το `sys` για να διαβαστεί είσοδος από το terminal.

Για να γίνει compile ο κώδικας C την πρώτη φορά μόνο χρειάζεται στο τερματικό να τρέξει η εντολή **make**. Στην συνέχεια η εντολή εκτέλεσης του προγράμματος είναι η εξής:

```
python3 optimization.py n_atoms
```

Π.χ. Πειράματα με αριθμό ατόμων ίσο με 4: `python3 optimization.py 4`