

Πανεπιστήμιο Πειραιώς – Τμήμα Πληροφορικής

Πρόγραμμα Μεταπτυχιακών Σπουδών

«Προηγμένα Συστήματα Πληροφορικής»

Μεταπτυχιακή Διατριβή

Τίτλος Διατριβής	Εφαρμογές του αλγορίθμου της νυχτερίδας σε πολυκριτηριακά προβλήματα βελτιστοποίησης Bat algorithm applications in multi-objective optimization problems
Όνοματεπώνυμο Φοιτητή	Δημήτριος Γονιδάκης
Πατρώνυμο	Κωνσταντίνος
Αριθμός Μητρώου	ΜΠΣΠ/10020
Επιβλέπων	Γεώργιος Τσιχριντζής, Καθηγητής

Ημερομηνία Παράδοσης **Ιανουάριος 2014**

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

Τριμελής Εξεταστική Επιτροπή

(υπογραφή)

(υπογραφή)

(υπογραφή)

Γεώργιος Τσιχριντζής
Καθηγητής

Δημήτριος Αποστόλου
Επίκουρος Καθηγητής

Χαράλαμπος Κωνσταντόπουλος
Επίκουρος Καθηγητής

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

ΕΥΧΑΡΙΣΤΙΕΣ

Θα ήθελα να εκφράσω τις ευχαριστίες μου σε όλους εκείνους που συνέβαλαν με οποιονδήποτε τρόπο στην υλοποίηση της παρούσας μεταπτυχιακής διατριβής.

Αρχικά, θα ήθελα να ευχαριστήσω τον Καθηγητή του τμήματος Πληροφορικής του Πανεπιστημίου Πειραιώς, κ. Γεώργιο Τσιχριντζή για την ευκαιρία που μου έδωσε να εκπονήσω την παρούσα διατριβή υπό την επίβλεψή του και να ερευνήσω ένα ιδιαίτερα ενδιαφέρον θέμα.

Πολύ καθοριστική υπήρξε και η συμβολή τού Δρ. Πανεπιστημίου Πειραιώς, κ. Αριστείδη Βλάχου. Οφείλω να τον ευχαριστήσω καθώς η μεγάλη εμπειρία του στο συγκεκριμένο επιστημονικό πεδίο, οι συμβουλές του και η καθοδήγησή του έπαιξαν πολύ σημαντικό ρόλο στην ολοκλήρωση της διπλωματικής διατριβής.

Επίσης, ευχαριστώ τη μητέρα μου Πάτρα και την αδερφή μου Κατερίνα για την ηθική τους συμπαράσταση κατά τη διάρκεια εκπόνησης της διατριβής.

Τέλος, θα ήθελα να ευχαριστήσω το φίλο μου Σαράντη Κορώνη, κτηνίατρο Αριστοτελείου Πανεπιστημίου Θεσσαλονίκης, για τις πληροφορίες που μου παρείχε σχετικά με τη συμπεριφορά και τα χαρακτηριστικά των νυχτερίδων.

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

ΠΕΡΙΕΧΟΜΕΝΑ

ΠΕΡΙΛΗΨΗ.....	5
ABSTRACT	5
1. ΕΙΣΑΓΩΓΗ	7
2. ΜΕΤΑΕΥΡΕΤΙΚΕΣ ΜΕΘΟΔΟΙ ΒΕΛΤΙΣΤΟΠΟΙΗΣΗΣ	11
2.1. Βασικές αρχές.....	11
2.2. Γενετικοί αλγόριθμοι.....	12
2.2.1. Απλός γενετικός αλγόριθμος.....	13
2.2.2. Ρύθμιση παραμέτρων	17
2.3. Διαφορική εξέλιξη.....	18
2.3.1. Περιγραφή του αλγορίθμου	19
2.3.2. Παραλλαγές του αλγορίθμου	22
2.3.3. Διαφορές και ομοιότητες με άλλους εξελικτικούς αλγορίθμους.....	23
2.4. Προσομοιωμένη ανόπτηση.....	24
2.4.1. Περιγραφή του αλγορίθμου	24
2.4.2. Ρύθμιση παραμέτρων	25
2.5. Βελτιστοποίηση σμήνους σωματιδίων	28
2.5.1. Βασική μορφή του αλγορίθμου	28
2.5.2. Ρύθμιση παραμέτρων	29
2.5.3. Συντελεστής αδράνειας	30
2.5.4. Συντελεστές περιορισμού.....	32
2.5.5. Πλήρως ενημερωμένο σμήνος σωματιδίων	32
2.5.6. Τοπολογία πληθυσμού	33
2.6. Αναζήτηση αρμονίας.....	34
2.6.1. Περιγραφή του αλγορίθμου	36
2.6.2. Βελτιωμένες μορφές του αλγορίθμου.....	38
2.7. Αποικία τεχνητών μελισσών	40
2.7.1. Περιγραφή του αλγορίθμου	40
2.8. Άλλοι μεταευρετικοί αλγόριθμοι	43
2.8.1. Εξελικτικές στρατηγικές.....	44
2.8.2. Γενετικός προγραμματισμός	44
2.8.3. Βελτιστοποίηση αποικίας μυρμηγκιών.....	44
2.8.4. Βελτιστοποίηση αναζήτησης τροφής των βακτηρίων	45
2.8.5. Αλγόριθμος σμήνους ψαριών	45
2.8.6. Αλγόριθμος της πυγολαμπίδας.....	46
2.8.7. Αναζήτηση του κούκου.....	46
2.8.8. Αναζήτηση βαρύτητας.....	47

3. Ο ΑΛΓΟΡΙΘΜΟΣ ΤΗΣ ΝΥΧΤΕΡΙΔΑΣ	49
3.1. Ο ηχοεντοπισμός των νυχτερίδων.....	49
3.1.1. Γενικά στοιχεία για τις νυχτερίδες.....	49
3.1.2. Το φαινόμενο του ηχοεντοπισμού.....	50
3.1.3. Οι ακουστικές ιδιότητες του ηχοεντοπισμού	50
3.2. Περιγραφή του αλγορίθμου	52
3.2.1. Η κίνηση των ψηφιακών νυχτερίδων	53
3.2.2. Ένταση ήχου και εκπομπή παλμών.....	54
3.3. Ανάλυση χαρακτηριστικών και σύγκριση με άλλους αλγορίθμους	55
3.4. Βελτιωμένες μορφές του αλγορίθμου της νυχτερίδας.....	56
3.4.1. Υβριδικός αλγόριθμος νυχτερίδας.....	56
3.4.2. Εισαγωγή τελεστή μετάλλαξης.....	57
3.4.3. Άλλες μορφές του αλγορίθμου της νυχτερίδας	58
4. ΤΟ ΠΡΟΒΛΗΜΑ ΤΗΣ ΣΥΝΔΥΑΣΜΕΝΗΣ ΚΑΤΑΝΟΜΗΣ ΟΙΚΟΝΟΜΙΚΟΥ ΚΟΣΤΟΥΣ ΚΑΙ ΕΚΠΟΜΠΗΣ ΡΥΠΩΝ	59
4.1. Οικονομική κατανομή φορτίου	60
4.2. Κατανομή εκπομπής ρύπων.....	61
4.3. Συνδυασμένη κατανομή οικονομικού κόστους και εκπομπής ρύπων	62
4.4. Βελτιστοποίηση κατά Pareto.....	62
4.5. Εισαγωγή παράγοντα ποιότητος	64
4.6. Μέθοδος σταθμισμένου αθροίσματος.....	67
5. ΕΦΑΡΜΟΓΕΣ ΚΑΙ ΑΠΟΤΕΛΕΣΜΑΤΑ.....	69
5.1. Σύστημα παραγωγής I	69
5.2. Σύστημα παραγωγής II	72
5.3. Σύστημα παραγωγής III	75
6. ΣΥΝΟΨΗ – ΣΥΜΠΕΡΑΣΜΑΤΑ.....	89
ΠΑΡΑΡΤΗΜΑ	91
Π.1. Κατάλογος ακρωνύμιων.....	91
Π.2. Κατάλογος εικόνων	91
Π.3. Κατάλογος πινάκων	92
Π.4. Κατάλογος ψευδοκωδίκων.....	93
ΒΙΒΛΙΟΓΡΑΦΙΑ	95

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

ΠΕΡΙΛΗΨΗ

Τις τελευταίες τέσσερις δεκαετίες παρουσιάζεται μεγάλη πρόοδος στην ανάπτυξη μεταευρετικών μεθόδων δηλαδή αλγορίθμων που μπορούν να εφαρμοστούν σε δύσκολα προβλήματα βελτιστοποίησης για τα οποία είναι ακατάλληλη η χρήση αιτιοκρατικών προσεγγίσεων. Τέτοιες προηγμένες τεχνικές βασίζονται συνήθως στη μοντελοποίηση φυσικών, βιολογικών και κοινωνικών φαινομένων. Μεταευρετικές μέθοδοι όπως οι γενετικοί αλγόριθμοι, η διαφορική εξέλιξη, η προσομοιωμένη απόπτηση, η βελτιστοποίηση σμήνους σωματιδίων, η αναζήτηση αρμονίας και η αποικία τεχνητών μελισσών συνδυάζουν με επιτυχία την εκμετάλλευση γύρω από μία καλή λύση και την εξερεύνηση του χώρου λύσεων. Μία πολύ πρόσφατη και ιδιαίτερα αποδοτική μεταευρετική μέθοδος είναι ο αλγόριθμος της νυχτερίδας που είναι εμπνευσμένος από το φαινόμενο του ηχοεντοπισμού που παρατηρείται στις μικρές νυχτερίδες. Ο αλγόριθμος της νυχτερίδας χρησιμοποιείται για την επίλυση της συνδυασμένης κατανομής οικονομικού κόστους και εκπομπής ρύπων, ενός πολύπλοκου και πολύ σημαντικού προβλήματος βελτιστοποίησης σε ένα σύστημα παραγωγής ενέργειας. Στόχος είναι η ελαχιστοποίηση του κόστους καυσίμων και της ποσότητας των ρυπογόνων ουσιών που εκπέμπονται, δύο αντικειμενικών συναρτήσεων που συγκρούονται μεταξύ τους ενώ ταυτόχρονα πρέπει να ικανοποιείται μία σειρά λειτουργικών περιορισμών. Ο αλγόριθμος της νυχτερίδας εφαρμόζεται σε τρία συστήματα παραγωγής ενέργειας και τα αποτελέσματά του συγκρίνονται με αυτά που προέρχονται από άλλους αλγορίθμους.

ABSTRACT

Last four decades there is a high progress level in development of metaheuristics, i.e. algorithms which can be applied to NP-hard optimization problems where deterministic approaches can't be used. Such advanced techniques are usually based on modeling of natural, biological and social behaviours. Metaheuristics such as Genetic Algorithms, Differential Evolution, Simulated Annealing, Particle Swarm Optimization, Harmony Search and Artificial Bee Colony successfully combine exploitation around a good solution and exploration of the solution space. Bat Algorithm is a very recent and efficient metaheuristic, inspired by the echolocation of microbats. It is used for solving the combined economic and emission dispatch, a very important complex optimization problem in a power generation system. Main purpose is the minimization of two conflicting objectives, the fuel cost and pollutant emission, while satisfying all operational constraints. Bat Algorithm is used for three different generating test systems and obtaining results are compared to those obtained by other algorithms.

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

1. ΕΙΣΑΓΩΓΗ

Τα προβλήματα βελτιστοποίησης παρουσιάζονται σε κάθε τομέα της ανθρωπίνης δραστηριότητας, από την καθημερινή ζωή μέχρι την επιστήμη και την τεχνολογία. Προβλήματα που συναντούμε συχνά όπως η αναζήτηση της συντομότερης διαδρομής ή πιο σύνθετα θέματα όπως η κατάρτιση ενός επιχειρηματικού πλάνου ή ο διαμοιρασμός πόρων (εφαρμογών, συσκευών, αρχείων) σε ένα δίκτυο υπολογιστών είναι δραστηριότητες στις οποίες καλούμαστε να βελτιστοποιήσουμε δηλαδή να μεγιστοποιήσουμε ή να ελαχιστοποιήσουμε ένα μέγεθος στα πλαίσια των δυνατοτήτων του πραγματικού κόσμου. Ο στόχος κάθε φορά είναι η εύρεση της βέλτιστης λύσης μέσα από ένα σύνολο διαθέσιμων λύσεων. Κάτι τέτοιο καθίσταται πολύ δύσκολο και πολλές φορές απαγορευτικό όταν το πρόβλημα προς επίλυση είναι ιδιαίτερα πολύπλοκο και ο χώρος λύσεων τεράστιος. Πρέπει λοιπόν να αναζητήσουμε τεχνικές οι οποίες μπορούν να εντοπίζουν όσο το δυνατό καλύτερες λύσεις, ικανοποιώντας παράλληλα τους περιορισμούς που τίθενται κάθε φορά.

Ένα πρόβλημα βελτιστοποίησης περιγράφεται μαθηματικά από την ακόλουθη γενική μορφή:

$$\text{minimize } f_i(x), \quad i = 0, 1, 2, \dots, I \quad (1.1)$$

έτσι ώστε

$$g_j(x) = 0, \quad j = 0, 1, 2, \dots, J \quad (1.2)$$

και

$$h_k(x) \leq 0, \quad k = 0, 1, 2, \dots, K \quad (1.3)$$

Στις παραπάνω σχέσεις, $x \in \mathbb{R}^n$ είναι η μεταβλητή βελτιστοποίησης με $x = (x_1, x_2, \dots, x_n)$. Τα στοιχεία x_1, x_2, \dots, x_n της μεταβλητής x καλούνται μεταβλητές απόφασης (decision variables) και μπορούν να είναι διακριτές, συνεχείς ή συνδυασμός των δύο. Ο αριθμός των διαστάσεων του προβλήματος είναι ίσος με τον αριθμό n των μεταβλητών απόφασης.

Οι συναρτήσεις $f: \mathbb{R}^n \rightarrow \mathbb{R}$ καλούνται αντικειμενικές συναρτήσεις (objective functions ή objectives) ή συναρτήσεις κόστους (cost functions). Μπορούμε να θεωρήσουμε ως αντικειμενική συνάρτηση την $-f$ και να προκύψει πρόβλημα μεγιστοποίησης. Οι αντικειμενικές συναρτήσεις μπορούν να είναι γραμμικές ή μη γραμμικές. Αν $I = 1$ υπάρχει μόνο μία αντικειμενική συνάρτηση και το πρόβλημα βελτιστοποίησης ονομάζεται μονοκριτηριακό (single objective) ενώ αν $I > 1$ το πρόβλημα ονομάζεται πολυκριτηριακό (multi-objective). Στην ακραία περίπτωση όπου $I = 0$ δηλαδή δεν υπάρχει αντικειμενική συνάρτηση παρά μόνο περιορισμοί, έχουμε ένα πρόβλημα σκοπιμότητας (feasibility problem).

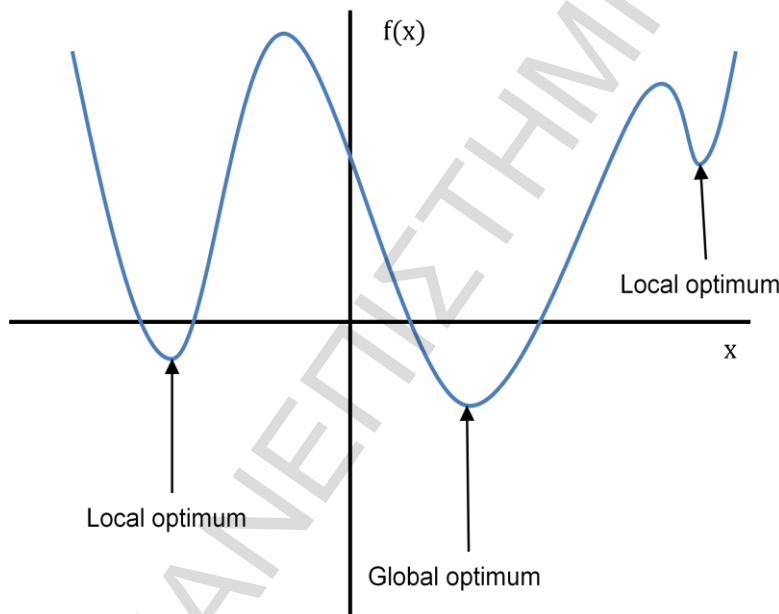
Οι σχέσεις (1.2) και (1.3) καλούνται περιορισμοί ισότητας (equality constraints) και ανισότητας (inequality constraints). Παρόμοια, οι συναρτήσεις $g: \mathbb{R}^n \rightarrow \mathbb{R}$ και $h: \mathbb{R}^n \rightarrow \mathbb{R}$ είναι οι συναρτήσεις ισότητας και ανισότητας αντίστοιχα. Λαμβάνοντας την αντίθετη συνάρτηση της h , η σχέση (1.3) μετατρέπεται σε ανισότητα μεγαλύτερη ή ίση του μηδενός. Αν $J = K = 0$ προκύπτει πρόβλημα βελτιστοποίησης χωρίς περιορισμούς (unconstrained optimization problem). Όπως και οι αντικειμενικές συναρτήσεις, έτσι και οι συναρτήσεις των περιορισμών μπορούν να είναι γραμμικές ή μη γραμμικές. Αν όλες οι συναρτήσεις (αντικειμενικές και περιορισμών) είναι γραμμικές προκύπτει πρόβλημα γραμμικού προγραμματισμού.

Ένα διανυσμα $s \in \mathbb{R}^n$ είναι μία εφικτή λύση (feasible solution) του προβλήματος όταν ικανοποιεί τους περιορισμούς ισότητας και ανισότητας. Το σύνολο $S \subseteq \mathbb{R}^n$ όλων των εφικτών

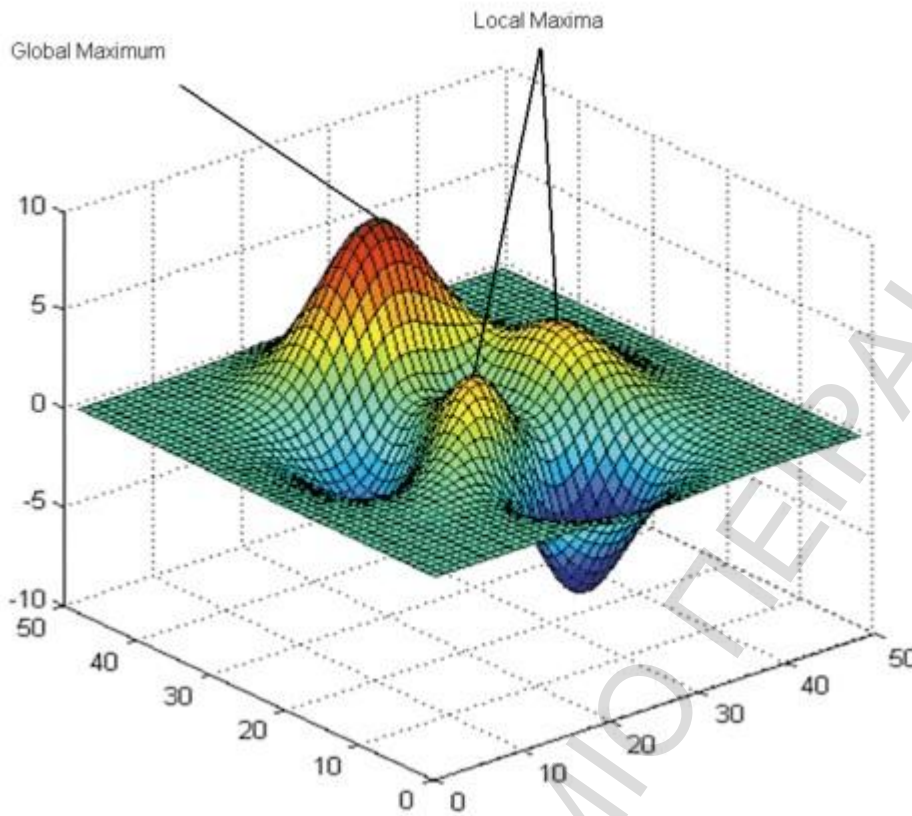
λύσεων του προβλήματος βελτιστοποίησης λέγεται χώρος αναζήτησης (search space) ή χώρος λύσεων (solution space). Όταν ένα πρόβλημα βελτιστοποίησης έχει μία τουλάχιστον εφικτή λύση ονομάζεται εφικτό. Στην αντίθετη περίπτωση ονομάζεται ανέφικτο. Για ένα διάνυσμα s_l του χώρου αναζήτησης μπορούμε να θεωρήσουμε ένα νέο διάνυσμα $\varphi_l \in \mathbb{R}^I$ με στοιχεία, τις τιμές των αντικειμενικών συναρτήσεων που αντιστοιχούν στο s_l δηλαδή $\varphi_l = [f_1(s_l), f_2(s_l), \dots, f_I(s_l)]$. Το σύνολο $\Phi \subseteq \mathbb{R}^I$ όλων διανυσμάτων των τιμών των αντικειμενικών συναρτήσεων που αντιστοιχούν στις εφικτές λύσεις του προβλήματος ονομάζεται χώρος απόκρισης (response space) ή χώρος αντικειμενικών συναρτήσεων (objective space).

Μία λύση ονομάζεται βέλτιστη όταν ελαχιστοποιεί την αντικειμενική συνάρτηση και ταυτόχρονα ικανοποιεί τους περιορισμούς ισότητας και ανισότητας. Για δύο εφικτές λύσεις s_1 και s_2 , η s_1 είναι καλύτερη της s_2 αν ισχύει η σχέση $f(s_1) < f(s_2)$ ενώ η s_1 είναι τουλάχιστον τόσο καλή όσο η s_2 αν ισχύει η σχέση $f(s_1) \leq f(s_2)$. Η βέλτιστη λύση σε ένα σύνολο γειτονικών εφικτών λύσεων ονομάζεται τοπικό βέλτιστο (local optimum). Η βέλτιστη λύση στο σύνολο όλων των εφικτών λύσεων ονομάζεται ολικό βέλτιστο (global optimum) (Εικόνες 1.1 και 1.2). Προφανώς, η βέλτιστη λύση ενός προβλήματος ταυτίζεται με το ολικό βέλτιστο. Σε ένα πρόβλημα σκοπιμότητας κάθε εφικτή λύση είναι βέλτιστη.

Με βάση την ανωτέρω περιγραφή γίνεται εύκολα αντιληπτό ότι σε προβλήματα βελτιστοποίησης, ιδιαίτερα αν υπάρχουν περιορισμοί και μη γραμμικές συναρτήσεις, δεν μπορούν να χρησιμοποιηθούν οι κλασικές αιτιοκρατικές (deterministic) μέθοδοι για την επίλυσή τους. Αιτιοκρατικοί αλγόριθμοι όπως η μέθοδος των πολλαπλασιαστών Lagrange μπορεί να δίνουν εγγυημένες λύσεις, ακολουθούν όμως μια συγκεκριμένη διαδικασία που επαναλαμβάνεται συνεχώς και με την εισαγωγή συγκεκριμένων δεδομένων παράγουν πάντοτε το ίδιο αποτέλεσμα.



Εικόνα 1.1. Τοπικά βέλτιστα και ολικό βέλτιστο σε πρόβλημα ελαχιστοποίησης μίας διαστάσης.



Εικόνα 1.2. Τοπικά βέλτιστα και ολικό βέλτιστο σε πρόβλημα μεγιστοποίησης δύο διαστάσεων.

Έτσι δημιουργήθηκε η ανάγκη ανάπτυξης στοχαστικών μεθόδων οι οποίες δε θα εγκλωβίζονται σε τοπικά βέλτιστα. Οι ερευνητές, μελετώντας φυσικά, βιολογικά και κοινωνικά φαινόμενα δημιούργησαν αλγόριθμους οι οποίοι λαμβάνοντας υπόψη και τον παράγοντα της τυχαιότητας μπορούν να εξερευνούν ένα μεγάλο μέρος του χώρου λύσεων και με επαναλαμβανόμενες εκτιμήσεις της αντικειμενικής συνάρτησης να παράγουν ικανοποιητικά αποτελέσματα.

Ο σκοπός της παρούσας μεταπτυχιακής διατριβής είναι η θεωρητική ανάλυση και η εφαρμογή του αλγορίθμου της νυχτερίδας (Bat Algorithm) καθώς και η σύγκρισή του με άλλους αλγόριθμους που μοντελοποιούν φαινόμενα ή συμπεριφορές που συναντώνται στην φύση. Πρόκειται για έναν σύγχρονο μεταερευνητικό αλγόριθμο, εμπνευσμένο από την ικανότητα των νυχτερίδων να εντοπίζουν το θήραμά τους με την εκπομπή και λήψη ηχητικών κυμάτων. Μπορεί να εφαρμοστεί με επιτυχία σε ένα μεγάλο εύρος προβλημάτων βελτιστοποίησης και συνδυαζόμενος με άλλους μεταερευνητικούς αλγόριθμους μπορεί να παράγει ακόμα καλύτερα αποτελέσματα.

Το πρόβλημα που επιλύεται με τη χρήση του αλγορίθμου της νυχτερίδας είναι η συνδυασμένη κατανομή οικονομικού κόστους και εκπομπής ρύπων (Combined Economic and Emission Dispatch). Στα σύγχρονα συστήματα παραγωγής ενέργειας απαιτείται η ελαχιστοποίηση του οικονομικού κόστους καυσίμου και της ποσότητας εκπομπής ρύπων, δύο μεγεθών που συγκρούονται μεταξύ τους αφού μειώνοντας το ένα αυξάνεται το άλλο. Ταυτόχρονα πρέπει να ληφθούν υπ' όψη και οι περιορισμοί χωρητικότητας της κάθε γεννήτριας και η σχέση παραγωγής και ζήτησης του συνολικού δικτύου. Συνεπώς προκύπτει ένα πολυκριτηριακό πρόβλημα βελτιστοποίησης με συνεχείς μεταβλητές το οποίο γίνεται πολυπλοκότερο όταν αυξάνεται το πλήθος των σταθμών παραγωγής ενέργειας του δικτύου.

Μετά τα εσαγωγικά στοιχεία που παρουσιάσαμε, η υπόλοιπη διπλωματική διατριβή ακολουθεί την παρακάτω δομή:

- Στο δεύτερο κεφάλαιο περιγράφονται οι βασικές αρχές των μεταερευτικών αλγορίθμων. Στη συνέχεια παρουσιάζεται συνοπτικά ένα πλήθος επιτυχημένων μεταερευτικών αλγορίθμων.
- Στο τρίτο κεφάλαιο γίνεται μία αναλυτική περιγραφή του αλγορίθμου της νυχτερίδας. Αναλύεται το φαινόμενο του ηχοεντοπισμού των νυχτερίδων, περιγράφεται η βασική μορφή του αλγορίθμου και ακολουθούν προτεινόμενες βελτιωμένες εκδοχές του.
- Στο τέταρτο κεφάλαιο αναπτύσσεται το θεωρητικό μέρος του προς επίλυση προβλήματος δηλαδή της συνδυασμένης κατανομής οικονομικού κόστους και εκπομπής ρύπων.
- Στο πέμπτο κεφάλαιο εξετάζονται τρία δίκτυα γεννητριών παραγωγής ενέργειας στα οποία εφαρμόζονται οι μορφές του αλγορίθμου της νυχτερίδας που παρουσιάστηκαν στο τρίτο κεφάλαιο. Τα αποτελέσματα συγκρίνονται με αυτά που έχουν προκύψει από άλλους αλγόριθμους.
- Τέλος συνοψίζονται τα συμπεράσματα που προκύπτουν από την παρούσα μελέτη και προτείνονται θέματα για μελλοντική έρευνα.

2. ΜΕΤΑΕΥΡΕΤΙΚΕΣ ΜΕΘΟΔΟΙ ΒΕΛΤΙΣΤΟΠΟΙΗΣΗΣ

Οι μεταευρετικές μέθοδοι (metaheuristics) είναι αυτές που θα μας απασχολήσουν στην παρούσα μελέτη. Σε αυτό το κεφάλαιο περιγράφονται τα βασικά χαρακτηριστικά των μεταευρετικών μεθόδων και ακολουθεί η παρουσίαση αντιπροσωπευτικών αλγορίθμων οι οποίοι έχουν εφαρμοστεί σε προβλήματα βελτιστοποίησης με επιτυχημένα αποτελέσματα.

2.1. Βασικές αρχές

Οι συμβατικοί αιτιοκρατικοί αλγόριθμοι εγγυώνται την εύρεση της βέλτιστης λύσης στα προβλήματα που καλούνται να επιλύσουν. Όταν όμως χρησιμοποιηθούν σε δύσκολα (NP-hard) προβλήματα βελτιστοποίησης απαιτούνται τόσο μεγάλοι υπολογιστικοί χρόνοι για την παραγωγή αποτελεσμάτων ώστε πρακτικά είναι αδύνατη η εφαρμογή τους. Έτσι, η έρευνα στράφηκε στην ανάπτυξη στοχαστικών μεθόδων οι οποίες μπορούν να βρουν μια πολύ καλή λύση (όχι πάντα τη βέλτιστη) σε εύλογο χρονικό διάστημα.

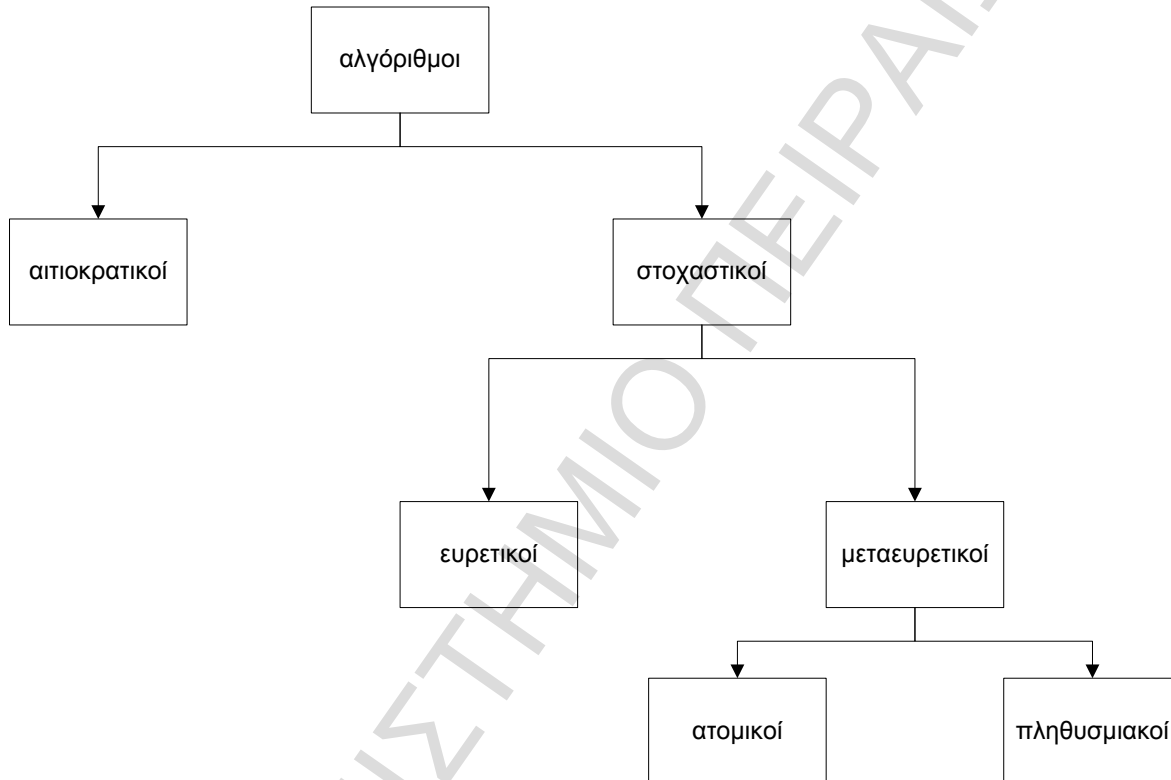
Οι στοχαστικές μέθοδοι διακρίνονται δύο κατηγορίες με μικρές διαφορές μεταξύ τους (Yang, 2010c), τις ευρετικές (heuristics) και τις μεταευρετικές (metaheuristics). Με τη σειρά τους οι ευρετικοί αλγόριθμοι διακρίνονται στους κατασκευαστικούς αλγορίθμους (constructive algorithms) και στους αλγορίθμους τοπικής αναζήτησης (local search algorithms) (Bianchi κ.ά., 2009). Οι κατασκευαστικοί αλγόριθμοι δημιουργούν μια λύση ενώνοντας τα διάφορα τμήματα της λύσης το ένα μετά το άλλο μέχρι η λύση να ολοκληρωθεί. Οι αλγόριθμοι τοπικής αναζήτησης ξεκινούν από μία υπάρχουσα λύση και προσπαθούν να τη βελτιώσουν μεταβάλλοντας ορισμένα τμήματά της. Είναι προφανές ότι ο όρος *heuristic* προέρχεται από το ρήμα *ευρίσκω*. Παρά την ετυμολογία τους, οι ευρετικές μέθοδοι δεν εγγυώνται ότι έχουν προσεγγίσει τη βέλτιστη λύση αν και μπορούν να βρискουν ποιοτικές λύσεις σε εύλογο χρονικό διάστημα. Αυτό μας αρκεί όταν θέλουμε απλώς να βρούμε όχι τις καλύτερες λύσεις αλλά απλώς καλές λύσεις οι οποίες εντοπίζονται εύκολα. Οι ευρετικές μέθοδοι σχεδιάζονται για την επίλυση συγκεκριμένων προβλημάτων (Talbi, 2009).

Το επίθεμα *meta-* στον όρο *metaheuristic* προέρχεται επίσης από την ελληνική γλώσσα και σημαίνει «πιο πέρα» ή «σε ένα υψηλότερο επίπεδο». Δηλαδή οι μεταευρετικές μέθοδοι είναι υψηλού επιπέδου τεχνικές, αποδοτικότερες από τις απλές ευρετικές (Yang, 2010c) και μπορούν να χρησιμοποιηθούν για την επίλυση σχεδόν όλων των προβλημάτων βελτιστοποίησης (Talbi, 2009). Αξίζει να σημειωθεί ότι αρκετές φορές στη βιβλιογραφία οι όροι *heuristics* και τα *metaheuristics* συγχέονται και δεν ορίζονται ξεκάθαρες διαφορές μεταξύ τους.

Δύο είναι τα βασικά χαρακτηριστικά όλων των μεταευρετικών μεθόδων: η εκμετάλλευση (exploitation) και η εξερεύνηση (exploration). Η εκμετάλλευση αφορά την αναζήτηση σε μια μικρή περιοχή στην οποία έχει ήδη εντοπιστεί μια καλή λύση. Η εξερεύνηση σχετίζεται με την αναζήτηση σε όλο το χώρο λύσεων με τη βοήθεια της τυχαιότητας η οποία δεν επιτρέπει στον αλγόριθμο να εγκλωβίζεται σε τοπικά βέλτιστα. Ένας αποδοτικός μεταευρετικός αλγόριθμος πρέπει να επιτυγχάνει έναν ισορροπημένο συνδυασμό μεταξύ αυτών των δύο διαδικασιών. Αυτή η «χρυσή τομή» είναι απαραίτητη ώστε σε σύντομο χρονικό διάστημα να προσδιορίζονται περιοχές του χώρου αναζήτησης με ποιοτικές λύσεις και παράλληλα να μη χάνεται χρόνος σε περιοχές που έχουν ήδη εξερευνηθεί ή δεν προσφέρουν καλές λύσεις.

Οι μεταευρετικοί αλγόριθμοι μπορούν να ταξινομηθούν σε δύο κατηγορίες (Yang, 2010c). Η πρώτη είναι οι πληθυσμιακοί αλγόριθμοι (population-based algorithms) οι οποίοι χρησιμοποιούν ένα πλήθος ατόμων ή πρακτόρων που αλληλεπιδρούν μεταξύ τους όπως κάνουν για παράδειγμα οι γενετικοί αλγόριθμοι και η βελτιστοποίηση σμήνους σωματιδίων. Η δεύτερη κατηγορία είναι οι ατομικοί αλγόριθμοι (individual-based algorithms) ή αλγόριθμοι τροχιάς (trajectory-based algorithms). Εδώ υπάρχει ένας πράκτορας που κινείται στο χώρο αναζήτησης για την εύρεση λύσεων. Χαρακτηριστικό παράδειγμα αλγορίθμου τροχιάς είναι η προσομοιωμένη απόπτηση. Η ταξινόμηση των αλγορίθμων που περιγράψαμε φαίνεται στην Εικόνα 2.1.

Πρέπει να επισημάνουμε ότι στους μεταευρετικούς αλγόριθμους, όπως και στους ευρετικούς, μεγάλο ρόλο παίζουν η εμπειρία και ο πειραματισμός. Η τιμές των παραμέτρων σε έναν αλγόριθμο ρυθμίζονται με βάση τα χαρακτηριστικά του πρόβληματος που πρόκειται να επιλυθεί. Επιπλέον σε ένα πρόβλημα βελτιστοποίησης με πολλές παραμέτρους ή τεράστιο χώρο λύσεων (όπως είναι τα συνεχή προβλήματα) είναι πιθανό ακόμα και οι μεταευρετικοί αλγόριθμοι να μη φτάνουν στη βέλτιστη λύση καθώς είναι αδύνατον να εξεταστούν όλες οι πιθανές λύσεις. Μπορούν όμως να βρουν μία περιοχή γύρω από τη βέλτιστη λύση ή μία λύση πολύ κοντά στη βέλτιστη που μπορεί να γίνει αποδεκτή. Έτσι, πολλές φορές ο όρος *βέλτιστη λύση* στη βιβλιογραφία προσδιορίζει μία εφικτή λύση που είναι αποδεκτή ή βρίσκεται πολύ κοντά στη βέλτιστη.



Εικόνα 2.1. Ταξινόμηση των αλγορίθμων

2.2. Γενετικοί αλγόριθμοι

Η θεωρία της εξέλιξης των ειδών που διατυπώθηκε αρχικά από τον Κάρολο Δαρβίνο, αποτέλεσε πηγή έμπνευσης για ένα πλήθος μεθόδων που εφαρμόζονται σε προβλήματα βελτιστοποίησης. Η εξέλιξη ενός είδους στη φύση βασίζεται στη φυσική επιλογή (οι οργανισμοί που είναι περισσότερο προσαρμοσμένοι στο περιβάλλον τους επιβιώνουν), στην αναπαραγωγή (δημιουργία απογόνων που κληρονομούν χαρακτηριστικά των γονέων τους) και στη μετάλλαξη (τυχαία μεταβολή στο γενετικό υλικό ενός οργανισμού). Οι ερευνητές, παρατηρώντας κοινά χαρακτηριστικά μεταξύ της βιολογικής εξέλιξης και της βελτιστοποίησης, οδηγήθηκαν στην ανάπτυξη του εξελικτικού υπολογισμού (Evolutionary Computation), ενός πολύ σημαντικού κλάδου της υπολογιστικής νοημοσύνης (Computational Intelligence).

Ο εξελικτικός υπολογισμός περιλαμβάνει ένα ευρύ φάσμα τεχνικών που ξεκίνησαν να εφαρμόζονται στις αρχές της δεκαετίας του 1950 (Das, 2011). Υποσύνολο του εξελικτικού υπολογισμού είναι οι εξελικτικοί αλγόριθμοι (Evolutionary Algorithms). Πρόκειται για πληθυσμιακούς αλγόριθμους οι οποίοι έχουν αναπτυχθεί ανεξάρτητα αλλά χρησιμοποιούν παρόμοιους μηχανισμούς, εμπνευσμένους από τη βιολογική εξέλιξη. Οι εξελικτικοί αλγόριθμοι που εφαρμόζονται συχνότερα σε προβλήματα βελτιστοποίησης είναι οι γενετικοί αλγόριθμοι (Genetic Algorithms), ο γενετικός προγραμματισμός (Genetic Programming), οι εξελικτικές στρατηγικές (Evolutionary Strategies), ο εξελικτικός προγραμματισμός (Evolutionary Programming) και η διαφορική εξέλιξη (Differential Evolution) (Fister, Fister και Yang, 2013).

Οι γενετικοί αλγόριθμοι (Genetic Algorithms – GA) είναι από τις πιο ευρέως χρησιμοποιούμενες τεχνικές στη σύγχρονη βελτιστοποίηση. Προτάθηκαν από τον Holland (1975) που ήταν ο πρώτος που χρησιμοποίησε διασταύρωση (crossover) / αναδιάταξη (recombination), μετάλλαξη (mutation) και επιλογή (selection) για τη μελέτη τεχνητών και προσαρμοστικών συστημάτων. Μέχρι και σήμερα έχουν αναπτυχθεί πολλές εκδοχές γενετικών αλγορίθμων με εφαρμογές σε ένα ευρύτατο φάσμα προβλημάτων βελτιστοποίησης από κλασικά διακριτά προβλήματα όπως το πρόβλημα του περιοδεύοντος πωλητή (Travel Salesman Problem) μέχρι και πιο πολύπλοκα συνεχή προβλήματα όπως η αποδοτική σχεδίαση αεροτομών στη μηχανική αεροσκαφών (Yang, 2010a).

2.2.1. Απλός γενετικός αλγόριθμος

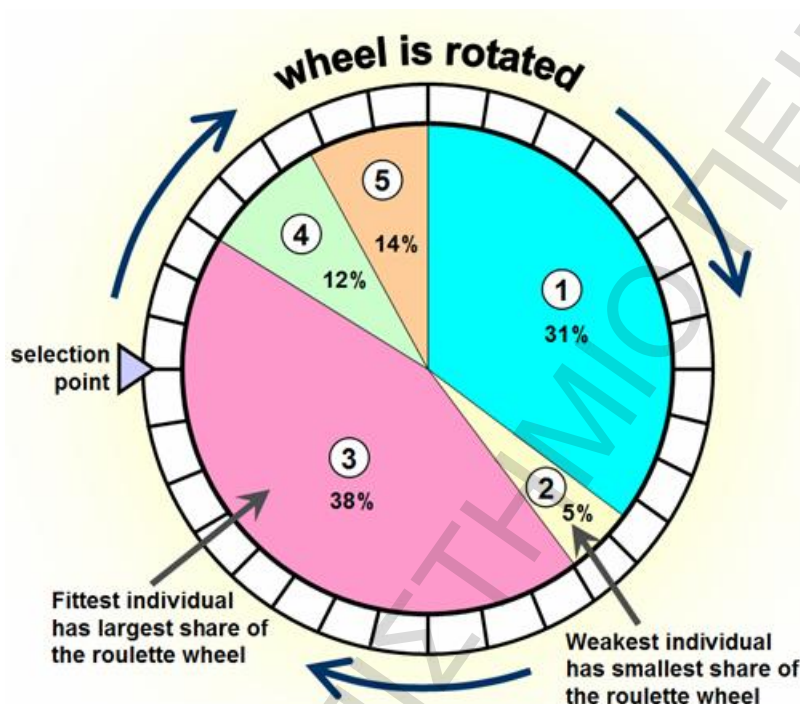
Ο γενετικός αλγόριθμος που παρουσιάστηκε από τον Holland (1975) αναφέρεται στην βιβλιογραφία ως απλός γενετικός αλγόριθμος (Simple Genetic Algorithm – SGA). Το στοιχειώδες χαρακτηριστικό σε έναν γενετικό αλγόριθμο είναι ένας πληθυσμός δυαδικών συμβολοσειρών που παίζουν το ρόλο των χρωμοσωμάτων. Κάθε συμβολοσειρά που αποτελείται από τα ψηφία 0 και 1, είναι η κωδικοποιημένη μορφή μιας λύσης σε ένα πρόβλημα βελτιστοποίησης. Με τη χρήση γενετικών τελεστών (διασταύρωση και μετάλλαξη) στα άτομα του τρέχοντος πληθυσμού, ο αλγόριθμος δημιουργεί την επόμενη γενιά. Στους εξελικτικούς αλγορίθμους, γενιά (generation) ονομάζεται μία επανάληψη στην οποία δημιουργείται ένας νέος πληθυσμός. Ο κύκλος των γενεών επαναλαμβάνεται μέχρι να επιτευχθεί κάποιο κριτήριο τερματισμού. Ο απλός γενετικός αλγόριθμος αποτελείται από τα εξής στοιχεία (Srinivas και Patnaik, 1994):

- έναν πληθυσμό δυαδικών συμβολοσειρών
- ένα μηχανισμό κωδικοποίησης των λύσεων σε δυαδικές συμβολοσειρές
- παραμέτρους ελέγχου
- μια συνάρτηση καταλληλότητας
- ένα μηχανισμό επιλογής

Βασικό στοιχείο στη δομή ενός γενετικού αλγόριθμου είναι ο μηχανισμός κωδικοποίησης για την αναπαράσταση των μεταβλητών του προβλήματος βελτιστοποίησης. Ο μηχανισμός κωδικοποίησης εξαρτάται από τη φύση του προβλήματος. Για παράδειγμα σε ένα πρόβλημα με συνεχείς μεταβλητές, μία συνηθισμένη μέθοδος κωδικοποίησής τους είναι η χρήση της ακέραιας αναπαράστασής τους (Srinivas και Patnaik, 1994). Κάθε συνεχής μεταβλητή αντιστοιχίζεται με γραμμικό τρόπο σε έναν ακέραιο αριθμό ο οποίος ορίζεται σε ένα συγκεκριμένο διάστημα και ο ακέραιος μετατρέπεται σε μία σειρά δυαδικών bit. Οι δυαδικές αναπαραστάσεις όλων των μεταβλητών ενώνονται και δημιουργείται μία ενιαία δυαδική συμβολοσειρά.

Κάθε συμβολοσειρά αξιολογείται από την αντικειμενική συνάρτηση. Το εύρος των τιμών της αντικειμενικής συνάρτησης διαφέρει από πρόβλημα σε πρόβλημα. Για να διατηρείται ομοιομορφία μεταξύ διαφορετικών προβλημάτων βελτιστοποίησης, χρησιμοποιούμε τη συνάρτηση καταλληλότητας (fitness function) και κανονικοποιούμε την αντικειμενική συνάρτηση σε ένα διάστημα μεταξύ 0 και 1. Η κανονικοποιημένη τιμή της αντικειμενικής συνάρτησης για μία συμβολοσειρά είναι η τιμή καταλληλότητάς της.

Ο μηχανισμός της επιλογής χρησιμοποιεί την τιμή καταλληλότητας για να αξιολογήσει μία λύση. Οι καταλληλότερες λύσεις επιβιώνουν και οι πιο αδύναμες αφανίζονται. Στον SGA όσο πιο κατάλληλη είναι μία συμβολοσειρά, τόσο πιο πολλούς απογόνους έχει. Η επιλογή των χρωμοσωμάτων / συμβολοσειρών για την δημιουργία της επόμενης γενιάς στον SGA γίνεται με τη μέθοδο της ρουλέτας (roulette wheel selection). Κάθε χρωμόσωμα τοποθετείται σε έναν τομέα (σχισμή) της ρουλέτας. Η επίκεντρη γωνία που αντιστοιχεί σε κάθε τομέα ισούται με $2\pi fit_i / \sum fit_i$ όπου fit_i η τιμή της συνάρτησης καταλληλότητας του χρωμοσώματος. Ένα χρωμόσωμα επιλέγεται για τη δημιουργία απογόνου αν ένας τυχαίος αριθμός που παράγεται στο διάστημα μεταξύ 0 και 2π βρίσκεται στον τομέα που αντιστοιχεί στο συγκεκριμένο χρωμόσωμα. Ο αλγόριθμος επιλέγει χρωμοσώματα με αυτό τον τρόπο έως ότου συμπληρωθεί ένας αριθμός χρωμοσωμάτων ίσος με τον πληθυσμό που έχει οριστεί. Στην Εικόνα 2.2 (Dalton, 2007) βλέπουμε ένα απλό σχήμα ρουλέτας με 5 χρωμοσώματα.

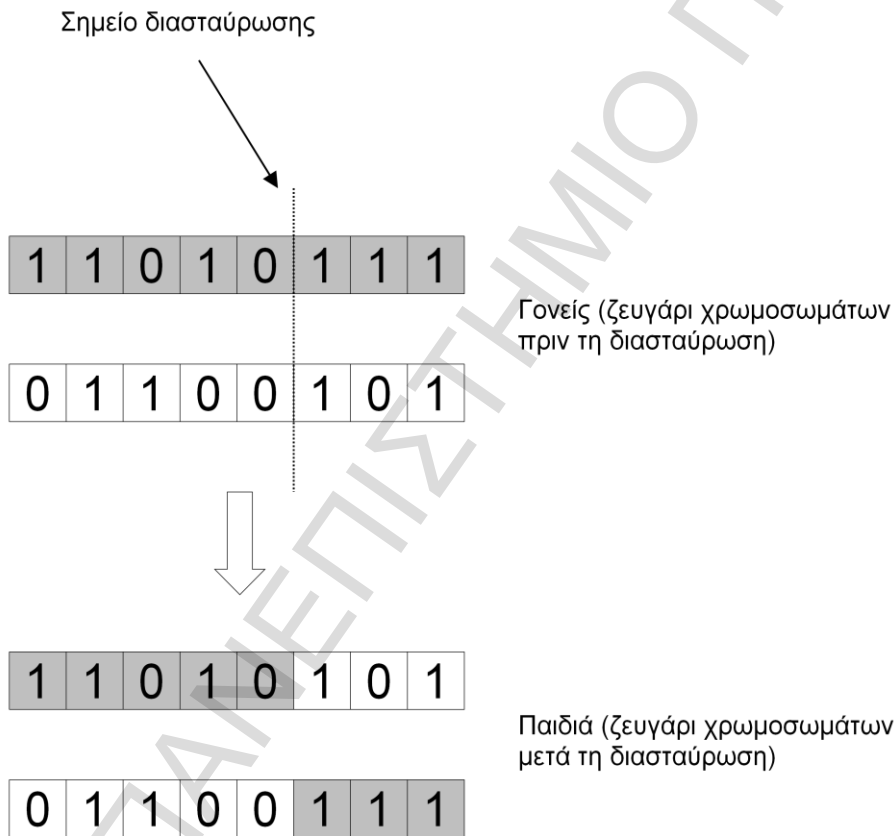


Εικόνα 2.2. Απεικόνιση της μεθόδου της ρουλέτας. Κάθε ένα από τα πέντε χρωμοσώματα τοποθετείται σε ένα τομέα μεγέθους ανάλογου με την τιμή καταλληλότητάς του δηλαδή όσο μεγαλύτερη είναι η τιμή καταλληλότητας τόσο μεγαλύτερη είναι η σχισμή της ρουλέτας. Σε κάθε τομέα αναγράφεται το ποσοστό της καταλληλότητας για το αντίστοιχο χρωμόσωμα δηλαδή η τιμή $100\% \times fit_i / \sum fit_i$. Το καταλληλότερο χρωμόσωμα είναι το 3^ο και το λιγότερο κατάλληλο το 2^ο. Ο αριθμός των περιστροφών της ρουλέτας είναι ίσος με το μέγεθος του πληθυσμού. Όσο πιο κατάλληλο είναι ένα χρωμόσωμα τόσο πιθανότερο είναι να επιλεγεί μετά από κάθε περιστροφή.

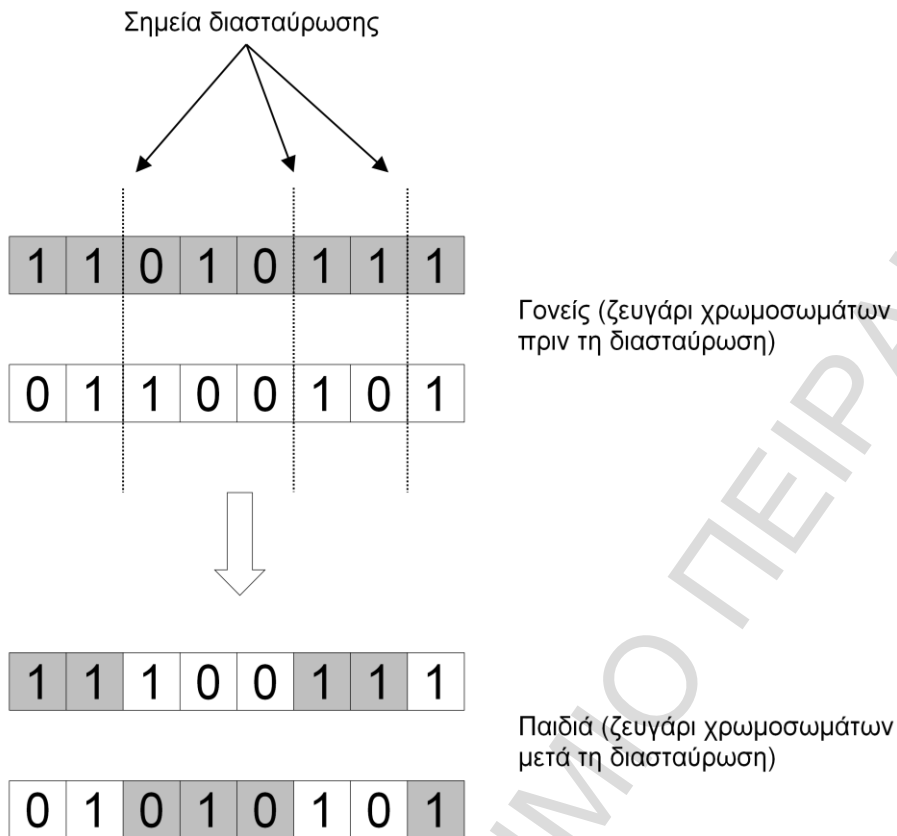
Μετά την επιλογή, τα χρωμοσώματα αναπαράγονται με τη λειτουργία της διασταύρωσης (ή αναδιάταξης). Ζευγάρια χρωμοσωμάτων λαμβάνονται τυχαία από τον πληθυσμό και διασταυρώνονται. Ο SGA χρησιμοποιεί τη διασταύρωση ενός σημείου (one point crossover) που είναι ο απλούστερος και ο συνηθέστερος τύπος διασταύρωσης στους γενετικούς αλγόριθμους (Thengade και Dondal, 2012). Αν υποθέσουμε ότι το μήκος ενός χρωμοσώματος είναι ίσο με l , επιλέγεται τυχαία ένα σημείο διασταύρωσης το οποίο παίρνει τιμές στο διάστημα από 1 έως και

$1 - l$. Τα τμήματα που έχουν δημιουργηθεί στα δύο χρωμοσώματα μετά τον καθορισμό του σημείου διασταύρωσης, ανταλλάσσουν θέσεις. Το τμήμα του ενός χρωμοσώματος αντικαθιστά το αντίστοιχο τμήμα του άλλου χρωμοσώματος και δημιουργούνται δύο νέα χρωμοσώματα (Εικόνα 2.3). Το σημείο διασταύρωσης μπορεί να θεωρηθεί οποιαδήποτε από τις $1 - l$ τιμές με την ίδια πιθανότητα. Η διασταύρωση πραγματοποιείται με μια πιθανότητα p_c . Αρκετοί γενετικοί αλγόριθμοι χρησιμοποιούν διασταύρωση πολλαπλών σημείων (Εικόνα 2.4) για να αυξήσουν την αποδοτικότητά τους.

Όταν ολοκληρωθεί η διασταύρωση, τα χρωμοσώματα υποβάλλονται σε μετάλλαξη. Η διαδικασία της μετάλλαξης περιλαμβάνει τη μεταβολή ενός bit της συμβολοσειράς από 0 σε 1 ή αντίστροφα (Εικόνα 2.5). Όπως η παράμετρος p_c ελέγχει την πιθανότητα διασταύρωσης, στη μετάλλαξη μία άλλη παράμετρος p_m δίνει την πιθανότητα αλλαγής ενός bit. Τα bit μιας συμβολοσειράς μεταλλάσσονται ανεξάρτητα δηλαδή η μεταβολή ενός bit δεν επηρεάζει την πιθανότητα μετάλλαξης άλλου bit. Στο SGA η μετάλλαξη αντιμετωπίζεται ως δευτερεύων γενετικός τελεστής έναντι της διασταύρωσης. Αν για παράδειγμα ένα bit σε ένα χρωμόσωμα πρέπει να μεταβληθεί ώστε να προκύψει η βέλτιστη λύση και αυτό δεν μπορεί να συμβεί με διασταύρωση με κάποιο άλλο χρωμόσωμα, εφαρμόζεται μετάλλαξη (Srinivas και Patnaik, 1994).



Εικόνα 2.3. Σχηματική αναπαράσταση διασταύρωσης ενός σημείου. Το σημείο διασταύρωσης επιλέγεται τυχαία και εδώ έχει την τιμή 5.



Εικόνα 2.4. Σχηματική αναπαράσταση διασταύρωσης τριών σημείων. Τα σημεία διασταύρωσης είναι 2, 5 και 7. Δύο bit απογόνων που συναντάται στα σημεία διασταύρωσης προέρχονται από διαφορετικούς γονείς.



Εικόνα 2.5. Σχηματική αναπαράσταση της μετάλλαξης. Το τρίτο bit μεταβάλλεται από 1 σε 0.

2.2.2. Ρύθμιση παραμέτρων

Σε κάθε γενετικό αλγόριθμο πρέπει να καθοριστεί ο μηχανισμός για την επιλογή των χρωμοσωμάτων που θα διασταυρωθούν. Πέραν της μεθόδου της ρουλέτας που περιγράψαμε, οι πιο κλασικοί μηχανισμοί επιλογής που έχουν χρησιμοποιηθεί είναι οι παρακάτω (Sivaraj και Ravichandran, 2011):

- Επιλογή βαθμονόμησης (ranking selection): Τα άτομα κάθε πληθυσμού ταξινομούνται από το χειρότερο στο καλύτερο με βάση την τιμή καταλληλότητάς τους. Στη συνέχεια κάθε άτομο λαμβάνει τιμή καταλληλότητας ίση με τη θέση κατάταξής του. Δηλαδή το χειρότερο άτομο έχει τιμή 1 και το καλύτερο N (μέγεθος πληθυσμού). Με αυτές τις τιμές καταλληλότητας τα άτομα επιλέγονται όπως και στη μέθοδο της ρουλέτας.
- Επιλογή τουρνουά (tournament selection): Δημιουργούνται υποσύνολα k ατόμων του πληθυσμού και από κάθε υποσύνολο (τουρνουά) το άτομο με τη μεγαλύτερη τιμή καταλληλότητας (νικητής του τουρνουά) επιλέγεται για διασταύρωση. Σε αυτή τη μέθοδο πρέπει να επιλέγεται το κατάλληλο μέγεθος του τουρνουά δηλαδή η τιμή k του πλήθους των ατόμων.

Ένα πολύ σημαντικό θέμα για έναν γενετικό αλγόριθμο είναι η διαμόρφωση ή η επιλογή μιας εύστοχης συνάρτησης καταλληλότητας η οποία καθορίζει το κριτήριο επιλογής σε ένα συγκεκριμένο πρόβλημα. Για την ελαχιστοποίηση μιας συνάρτησης με τη χρήση γενετικών αλγορίθμων ένας πολύ απλός τρόπος κατασκευής μιας συνάρτησης καταλληλότητας είναι η χρήση της ακόλουθης μορφής (Yang, 2010a):

$$fit = A - f(x) \quad (2.1)$$

όπου A είναι ένας πολύ μεγάλος σταθερός αριθμός αν θέλουμε η fit να παίρνει μόνο θετικές τιμές. Στις περισσότερες περιπτώσεις έχουμε $A = 0$, δηλαδή ο στόχος μας είναι η μεγιστοποίηση της συνάρτησης καταλληλότητας και επομένως η ελαχιστοποίηση της αντικειμενικής συνάρτησης f . Ένας άλλος τρόπος είναι να χρησιμοποιήσουμε μια συνάρτηση καταλληλότητας η οποία συσχετίζει την αντικειμενική συνάρτηση κάθε ατόμου του πληθυσμού με το σύνολο του πληθυσμού ως εξής (Yang, 2010a):

$$fit(x_i) = \frac{f(\xi_i)}{\sum_{i=1}^N f(\xi_i)} \quad (2.2)$$

όπου N είναι το μέγεθος του πληθυσμού και ξ_i είναι το χρωμόσωμα (συμβολοσειρά) που αναπαριστά τη λύση i . Η κατάλληλη μορφή της συνάρτησης fit θα εγγραφεί ότι οι πιο κατάλληλες λύσεις θα επιλεγούν. Με μία ανεπαρκή συνάρτηση καταλληλότητας μπορεί να προκύψουν εσφαλμένες ή ακατάλληλες λύσεις.

Καθοριστικό ρόλο για την απόδοση ενός γενετικού αλγορίθμου παίζει η επιλογή των τιμών για τις παραμέτρους της διασταύρωσης και της μετάλλαξης. Η πιθανότητα διασταύρωσης p_c είναι συνήθως πολύ μεγάλη, με τιμές μεταξύ 0.7 και 1 (Yang, 2010a). Αν η p_c είναι πολύ μικρή, έχουμε λίγες διασταυρώσεις και η εξέλιξη του πληθυσμού παραγματοποιείται με αργούς ρυθμούς με αποτέλεσμα ο αλγόριθμος να μην είναι αποδοτικός. Η διασταύρωση δε συμβαίνει πάντα ώστε να μπορούν κάποια άτομα του πληθυσμού να επιβιώνουν στην επόμενη γενιά. Σε αντίθεση με τη διασταύρωση, η πιθανότητα μετάλλαξης p_m για κάθε bit χρωμοσώματος συνήθως λαμβάνει πολύ μικρές τιμές συνήθως μεταξύ 0.001 και 0.05 (Yang, 2010a). Αν η p_m είναι πολύ μεγάλη, ένα μεγάλο τμήμα του χρωμοσώματος μεταβάλλεται και ο αλγόριθμος μετατρέπεται σε μια μέθοδο τυχαίας

αναζήτησης, επομένως δεν μπορεί να προσεγγίσει τη βέλτιστη λύση. Εάν όμως η μετάλλαξη παραληφθεί, ο αλγόριθμος κινδυνεύει να εγκλωβιστεί σε τοπικά βέλτιστα.

Η επιλογή ενός κατάλληλου μεγέθους πληθυσμού είναι επίσης πολύ σημαντική. Αν ο πληθυσμός είναι πολύ μικρός, η εξέλιξη είναι σύντομη και υπάρχει κίνδυνος εξαφάνισης όλου του πληθυσμού, όπως συμβαίνει και στον πραγματικό κόσμο σε είδη με μικρούς πληθυσμούς. Ένα τέτοιο σύστημα κινδυνεύει να συγκλίνει πρόωρα. Σε ένα μικρό πληθυσμό, αν ένα χρωμόσωμα μεγάλης καταλληλότητας εμφανιστεί νωρίς, μπορεί να αναπαράγει πολλούς απογόνους οι οποίοι θα κατακλύσουν τον πληθυσμό. Έτσι το σύστημα οδηγείται σε τοπικό βέλτιστο και όχι σε ολικό (Yang, 2010a). Από την άλλη πλευρά, αν το μέγεθος του πληθυσμού είναι πολύ μεγάλο, χρειάζονται περισσότερες αξιολογήσεις της αντικειμενικής συνάρτησης με συνέπεια αυξάνεται ο υπολογιστικός χρόνος.

Σε απλές μορφές γενετικών αλγορίθμων, μετά τη δημιουργία απογόνων οι γονείς αγνοούνται και υπάρχει το ενδεχόμενο να χαθούν οι καλύτερες λύσεις που έχουν βρεθεί μέχρι στιγμής. Η διατήρηση των καλύτερων λύσεων μέχρι την τρέχουσα χρονική γενιά ονομάζεται ελιτισμός (elitism) και παίζει σημαντικό ρόλο στη σύγκλιση του αλγορίθμου στο ολικό βέλτιστο (Das, 2011). Επομένως ένα κριτήριο για την επιλογή των καλύτερων λύσεων είναι πολύ σημαντικό. Στη βασική μορφή ελιτισμού σε έναν γενετικό αλγόριθμο επιλέγεται το πιο κατάλληλο άτομο κάθε γενιάς το οποίο θα διατηρηθεί στην επόμενη γενιά χωρίς να υποστεί καμία μεταβολή από τους γενετικούς τελεστές. Με αυτόν τρόπο η καλύτερη λύση επιτυγχάνεται γρηγορότερα. Ο ελιτισμός είναι φαινόμενο των περισσότερων εξελικτικών αλγορίθμων.

Τα βήματα ενός τυπικού γενετικού αλγορίθμου φαίνονται στον Ψευδοκώδικα που ακολουθεί.

Ψευδοκώδικας 2.1. Genetic Algorithm

```

Objective function  $f(x)$ ,  $x = (x_1, x_2, \dots, x_n)^T$ 
Encode the solutions into chromosomes (binary strings)
Define fitness function
Generate the initial population
Define probabilities of crossover ( $p_c$ ) and mutation ( $p_m$ )
  while (t<maximum number of generations)
    Select solutions for next generation
    Generate new solution by crossover and mutation
    if  $p_c > rand$  crossover end if
    if  $p_m > rand$  mutate end if
    Evaluate population
    Select the current best for new generation (elitism)
  end while
Decode the results and visualization

```

2.3. Διαφορική εξέλιξη

Ο αλγόριθμος της διαφορικής εξέλιξης (Differential Evolution – DE) (Storn και Price, 1995) (Storn και Price, 1997) έχει αναδειχθεί για παραπάνω από μία δεκαετία σε μία από τις πιο ανταγωνιστικές μορφές του εξελικτικού υπολογισμού. Παρουσιάζει σημαντικά χαρακτηριστικά που τον καθιστούν κατάλληλο για προβλήματα βελτιστοποίησης που συναντώνται σε πολλούς τομείς όπως η αναγνώριση προτύπων, η επεξεργασία σήματος και η ρομποτική.

2.3.1. Περιγραφή του αλγορίθμου

Η διαφορική εξέλιξη ακολουθεί τα ίδια υπολογιστικά βήματα με αυτά που ακολουθεί ένας τυπικός εξελικτικός αλγόριθμος. Το κύριο χαρακτηριστικό της, που τη διαφοροποιεί από τους παραδοσιακούς εξελικτικούς αλγόριθμους, είναι ότι λειτουργεί με διαφορές μεταξύ των διανυσμάτων που συγκροτούν τον πληθυσμό. Τα νέα διανύσματα προκύπτουν από πράξεις αφαίρεσης μεταξύ διανυσμάτων του τρέχοντος πληθυσμού.

Κάθε λύση καλείται διάνυσμα παραμέτρων (parameter vector) ή γονιδίωμα (genome). Ο αρχικός πληθυσμός των διανυσμάτων επιλέγεται τυχαία και πρέπει να καλύπτει όσο το δυνατό μεγαλύτερο μέρος του χώρου αναζήτησης. Συνήθως θεωρούμε μία ομοιόμορφη κατανομή πιθανότητας για όλες τις τυχαίες επιλογές. Ο αλγόριθμος DE παράγει νέα διανύσματα παραμέτρων προσθέτοντας τη σταθμισμένη διαφορά μεταξύ δύο διανυσμάτων του πληθυσμού σε ένα τρίτο διάνυσμα. Αυτή η λειτουργία καλείται μετάλλαξη (mutation). Οι παράμετροι του μεταλλαγμένου διανύσματος που ονομάζεται διάνυσμα δότης (donor vector) (Das, 2011) συνδυάζονται με τις παραμέτρους ενός άλλου προκαθορισμένου διανύσματος που ονομάζεται διάνυσμα στόχος (target vector) και προκύπτει ένα νέο διάνυσμα, το διάνυσμα δοκιμής (trial vector). Η διαδικασία της ανάμιξης των παραμέτρων ονομάζεται διασταύρωση (crossover) ή αναδιάταξη (recombination). Αν το διάνυσμα δοκιμής δίνει χαμηλότερη τιμή αντικειμενικής συνάρτησης απ' ό,τι το διάνυσμα στόχος, το διάνυσμα δοκιμής αντικαθιστά το διάνυσμα στόχο στην επόμενη γενιά. Η τελευταία λειτουργία ονομάζεται επιλογή (selection). Κάθε διάνυσμα του πληθυσμού παίζει το ρόλο του διανύσματος στόχου, μία φορά σε κάθε γενιά. Συνεπώς αν ο πληθυσμός αποτελείται από NP διανύσματα, θα έχουμε και NP συγκρίσεις.

Πιο συγκεκριμένα κατά τη λειτουργία της μετάλλαξης για κάθε διάνυσμα x_i με $i = 1, 2, \dots, NP$ της G -οστής γενιάς (επανάληψης) παράγεται ένα νέο, μεταλλαγμένο διάνυσμα σύμφωνα με την εξίσωση:

$$v_i^{(G)} = x_{r_1}^{(G)} + F \cdot (x_{r_2}^{(G)} - x_{r_3}^{(G)}) \quad (2.3)$$

όπου:

$x_{r_1}, x_{r_2}, x_{r_3}$: τρία τυχαία διανύσματα του πληθυσμού διαφορετικά μεταξύ τους και διαφορετικά από το διάνυσμα x_i . Το x_{r_1} ονομάζεται διάνυσμα βάσης (base vector).

F : ένας πραγματικός σταθερός συντελεστής που ανήκει στο διάστημα $[0, 2]$ (Storn και Price, 1997) και ελέγχει την επίδραση της διαφοράς $x_{r_2} - x_{r_3}$. Ένα εύρος τιμών του F που δίνει καλή απόδοση στον αλγόριθμο είναι μεταξύ του 0.4 και του 1 (Das, 2011).

Στην Εικόνα 2.6 φαίνεται η λειτουργία της μετάλλαξης σε δισδιάστατο χώρο αναζήτησης.

Για να αυξηθεί την ποικιλομορφία του πληθυσμού, μετά τη μετάλλαξη και την παραγωγή του διανύσματος δότη εφαρμόζεται η λειτουργία της διασταύρωσης (Εικόνα 2.7). Για κάθε παράμετρο v_{ij} με $j = 1, 2, \dots, D$ του διανύσματος v_i εφαρμόζεται η σχέση:

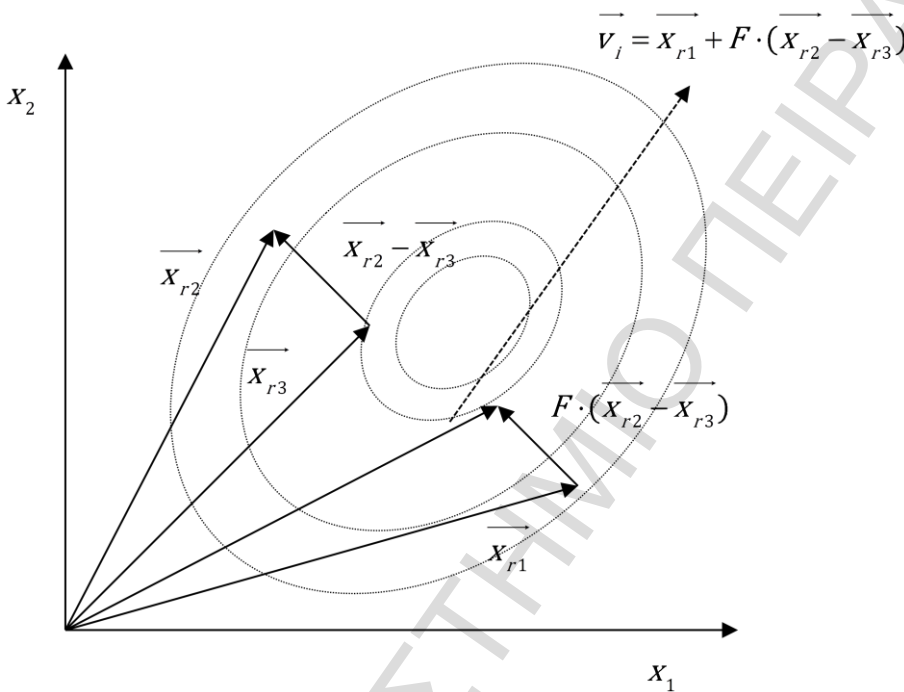
$$z_{ij}^{(G)} = \begin{cases} v_{ij}^{(G)}, & randb(j) \leq CR \vee j = j_{rand} \\ x_{ij}^{(G)}, & randb(j) > CR \wedge j \neq j_{rand} \end{cases} \quad (2.4)$$

όπου:

$randb(j)$: ένας ομοιόμορφα κατανεμημένος αριθμός που ανήκει στο διάστημα $[0,1]$ και υπολογίζεται εκ νέου για κάθε παράμετρο j και για κάθε διάνυσμα i .

CR : ένας σταθερός πραγματικός αριθμός που ανήκει στο διάστημα $[0,1]$ και ονομάζεται σταθερά διασταύρωσης. Για μεγαλύτερες τιμές του CR λαμβάνονται περισσότερες παράμετροι από το v_i . Το προτεινόμενο εύρος τιμών του CR εξαρτάται από διάφορους παράγοντες όπως το μέγεθος του πληθυσμού και το είδος της αντικειμενικής συνάρτησης (Rönkkönen, Kukkonen και Price, 2005) (Zaharie, 2007) (Das, 2011).

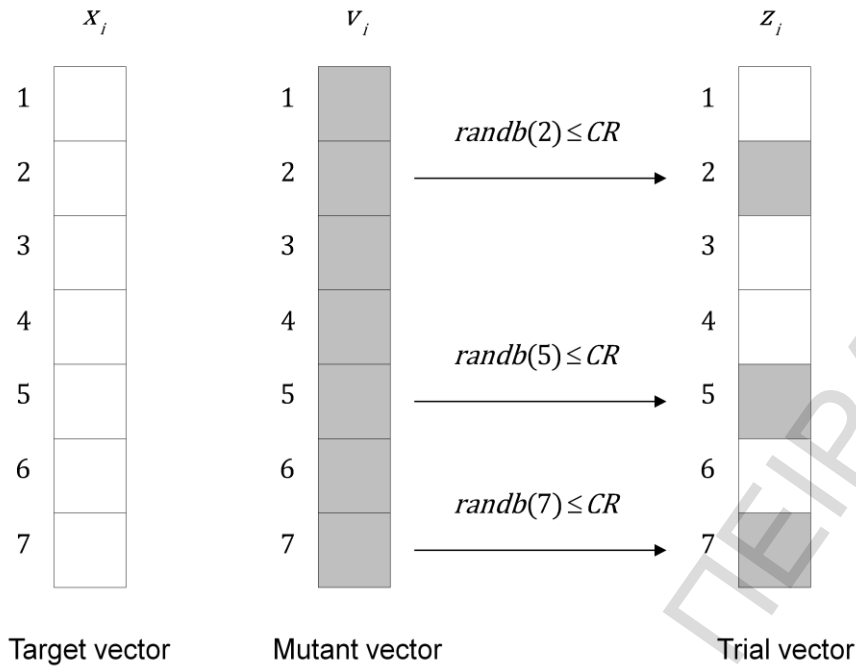
j_{rand} : μία διάσταση (παράμετρος) του διανύσματος που επιλέγεται τυχαία, ώστε να διασφαλιστεί ότι το διάνυσμα που προκύπτει έχει τουλάχιστον μία διάσταση διαφορετική από το διάνυσμα x_i .



Εικόνα 2.6. Απεικόνιση της μετάλλαξης του DE σε χώρο αναζήτησης 2 διαστάσεων

Στη διαδικασία της επιλογής ο αλγόριθμος DE αποφασίζει αν το διάνυσμα δοκιμής $z_i^{(G)}$ θα είναι μέλος της γενιάς $G + 1$. Αν το διάνυσμα $z_i^{(G)}$ δίνει μικρότερη τιμή στην αντικειμενική συνάρτηση f τότε παίρνει τη θέση του διανύσματος στόχου $x_i^{(G)}$ διαφορετικά το $x_i^{(G)}$ παραμένει στην επόμενη γενιά. Συνεπώς:

$$x_i^{(G+1)} = \begin{cases} z_i^{(G)}, & f(x_i^{(G)}) > f(z_i^{(G)}) \\ x_i^{(G)}, & f(x_i^{(G)}) \leq f(z_i^{(G)}) \end{cases} \quad (2.5)$$



Εικόνα 2.7. Απεικόνιση της διασταύρωσης για $D=7$ παραμέτρους σε κάθε διάνυσμα.

Ακολουθεί ο Ψευδοκώδικας του αλγορίθμου DE.

Ψευδοκώδικας 2.2. Differential Evolution

Objective function $f(x)$, $x = (x_1, x_2, \dots, x_d)^T$

Set constants F , CR and number of population size NP

Initialize vectors x_i

while (criterion)

for $i \leftarrow 1$ to NP

 Mutation step: generate a donor vector $v_i^{(G)}$ corresponding to the i -th target vector $x_i^{(G)}$

 Crossover step: generate a trial vector $z_i^{(G)}$ for the i -th target vector $x_i^{(G)}$

 Selection step: evaluate the trial vector

if ($z_i^{(G)}$ better than $x_i^{(G)}$)

 replace $x_i^{(G)}$ with $z_i^{(G)}$

end if

$G \leftarrow G + 1$

end for

end while

2.3.2. Παραλλαγές του αλγορίθμου

Στην προηγούμενη υποενότητα περιγράψαμε τα βήματα της απλής μορφής του αλγορίθμου DE. Η μετάλλαξη στην εξίσωση (2.3) χρησιμοποιεί ένα τυχαία επιλεγμένο διάνυσμα x_{r_1} και μόνο ένα σταθμισμένο διάνυσμα διαφοράς, $F \cdot (x_{r_2} - x_{r_3})$. Στη σχέση (2.4) της διασταύρωσης, κάθε παράμετρος του διανύσματος δοκιμής z_i επιλέγεται με πιθανότητα CR από το διάνυσμα δότη v_i και με πιθανότητα $1 - CR$ από το διάνυσμα στόχο x_i δηλαδή οι παράμετροι που λαμβάνονται από το v_i ακολουθούν τη διωνυμική κατανομή. Σε αυτή την περίπτωση έχουμε μία διωνυμική διασταύρωση (Zaharie, 2007).

Όταν η μετάλλαξη υλοποιείται σύμφωνα με την εξίσωση (2.3), αναφέρεται ως DE/rand/1. Επιπλέον, όταν χρησιμοποιείται διωνυμική διασταύρωση, η διαδικασία γράφεται DE/rand/1/bin. Μπορούμε πλέον εύκολα να αντιληφθούμε πως χαρακτηρίζονται οι διάφορες μορφές του DE. Η γενική σύμβαση που χρησιμοποιείται είναι η DE/x/y/z. Το x αντιπροσωπεύει μία συμβολοσειρά που προσδιορίζει τον τρόπο επιλογής του διανύσματος βάσης. Είναι “rand” όταν το διάνυσμα βάσης επιλέγεται τυχαία και “best” όταν επιλέγεται από τον πληθυσμό το διάνυσμα που δίνει τη χαμηλότερη τιμή στην αντικειμενική συνάρτηση. Το y είναι ο αριθμός των διανυσμάτων διαφοράς που προστίθενται στο διάνυσμα βάσης. Το z είναι επίσης μία συμβολοσειρά που προσδιορίζει τον τύπο της διασταύρωσης που χρησιμοποιείται. Είναι “bin” όταν χρησιμοποιείται διωνυμική διασταύρωση και “exp” όταν χρησιμοποιείται εκθετική.

Άλλες τέσσερις προτεινόμενες μορφές μετάλλαξης είναι οι εξής (Das, 2011):

“DE/best/1”:

$$v_i^{(G)} = x_{\text{best}}^{(G)} + F \cdot (x_{r_1}^{(G)} - x_{r_2}^{(G)}) \quad (2.6)$$

“DE/target-to-best/1”:

$$v_i^{(G)} = x_i^{(G)} + F \cdot (x_{\text{best}}^{(G)} - x_i^{(G)}) + F \cdot (x_{r_1}^{(G)} - x_{r_2}^{(G)}) \quad (2.7)$$

“DE/best/2”:

$$v_i^{(G)} = x_{\text{best}}^{(G)} + F \cdot (x_{r_1}^{(G)} - x_{r_2}^{(G)}) + F \cdot (x_{r_3}^{(G)} - x_{r_4}^{(G)}) \quad (2.8)$$

“DE/rand/2”:

$$v_i^{(G)} = x_{r_1}^{(G)} + F \cdot (x_{r_2}^{(G)} - x_{r_3}^{(G)}) + F \cdot (x_{r_4}^{(G)} - x_{r_5}^{(G)}) \quad (2.9)$$

Οι δείκτες r_1, r_2, r_3, r_4 και r_5 είναι φυσικοί αριθμοί διάφοροι μεταξύ τους, επιλέγονται τυχαία από το $[1, NP]$ και είναι διάφοροι του δείκτη i . $x_{\text{best}}^{(G)}$ είναι το διάνυσμα του πληθυσμού στη γενιά G που δίνει τη μικρότερη τιμή της αντικειμενικής συνάρτησης σε σχέση με τα υπόλοιπα διανύσματα.

Η διασταύρωση του αλγορίθμου εκτός από διωνυμική μπορεί να είναι και εκθετική. Στην εκθετική διασταύρωση επιλέγεται τυχαία ένας ακέραιος n από το διάστημα $[1, D]$. Ο n αποτελεί ένα σημείο εκκίνησης της διασταύρωσης στο διάστημα στόχο. Ακόμα, επιλέγεται ένας άλλος ακέραιος L από το διάστημα $[1, D]$ ο οποίος υποδηλώνει τον αριθμό των παραμέτρων που προσφέρονται από το διάνυσμα δότη. Μετά την επιλογή των παραμέτρων n και L το διάνυσμα δοκιμής προκύπτει ως εξής:

$$z_{ij}^{(G)} = \begin{cases} v_{ij}^{(G)} & \text{for } j = \langle n \rangle_D, \langle n+1 \rangle_D, \dots, \langle n+L-1 \rangle_D, \\ x_{ij}^{(G)} & \text{for all other } j \in [1, D] \end{cases} \quad (2.10)$$

όπου οι γωνιακές παρενθέσεις $\langle \cdot \rangle_D$ υποδηλώνουν μία πράξη modulo με διαιρέτη το D . Ο ακέραιος L λαμβάνεται από το διάστημα $[1, D]$ σύμφωνα με τον Ψευδοκώδικα:

```

L ← 1
while (rand(0,1) ≤ CR AND (L ≤ D))
L ← L+1
end while

```

Η πιθανότητα να ληφθούν h παράμετροι από το διάνυσμα δότη είναι $P(L = h) = (CR)^{h-1}$ για κάθε θετικό ακέραιο $h \in [1, D]$. Για κάθε διάνυσμα δότη πρέπει να επιλεγούν νέες τιμές n και L (Das, 2011).

2.3.3. Διαφορές και ομοιότητες με άλλους εξελικτικούς αλγορίθμους

Ο αλγόριθμος DE παρουσιάζει κοινά στοιχεία με τους άλλους εξελικτικούς αλγορίθμους ιδιαίτερα στη βασική του δομή. Η διαφορά του έγκειται στον τρόπο εκτέλεσης του κάθε βήματος.

Στους γενετικούς αλγορίθμους και γενικά στους εξελικτικούς αλγορίθμους, η μετάλλαξη εκλαμβάνεται ως μία τυχαία αλλαγή κάποιων παραμέτρων του διανύσματος. Η διαφορική εξέλιξη διαφέρει σημαντικά γιατί μεταλλάσσει τα διανύσματα βάσης προσθέτοντας σταθμισμένα διανύσματα διαφοράς που δημιουργούνται από διανύσματα διαφορετικά από το διάνυσμα βάσης. Στις εξελικτικές στρατηγικές και στον εξελικτικό προγραμματισμό απαιτείται πλήρης καθορισμός του μηχανισμού της μετάλλαξης για κάθε μεταβλητή. Αντίθετα, στη διαφορική εξέλιξη απαιτείται μόνο ο καθορισμός της σταθεράς F .

Η διαφορική εξέλιξη και οι εξελικτικές στρατηγικές χρησιμοποιούν διασταύρωση για να παράγουν ένα διάνυσμα δοκιμής, ενώ οι περισσότεροι γενετικοί αλγόριθμοι διασταυρώνουν δύο διανύσματα για να παράγουν δύο διανύσματα απογόνους. Όπως είδαμε στην προηγούμενη υποενοότητα μία συνηθισμένη τεχνική διασταύρωσης στους γενετικούς αλγορίθμους είναι η διασταύρωση των n σημείων. Να σημειώσουμε επίσης ότι στον εξελικτικό προγραμματισμό οι απόγονοι δημιουργούνται μόνο με μετάλλαξη και δεν υπάρχει τελεστής διασταύρωσης.

Η διαδικασία της επιλογής σε μία εξελικτική διαδικασία μπορεί να εφαρμοστεί σε δύο στάδια. Το πρώτο στάδιο είναι η *επιλογή του γονέα (parent selection)* που αποφασίζει ποια διανύσματα του πληθυσμού θα διασταυρωθούν, ενώ το δεύτερο στάδιο είναι η *επιλογή του επιζώντα (survivor selection)* που επιλέγει τα διανύσματα από τους πληθυσμούς των γονέων και των απογόνων που θα επιβιώσουν στην επόμενη γενιά. Αντίθετα με τους γενετικούς αλγορίθμους που επιλέγουν τους γονείς με βάση την καταλληλότητά τους, η διαφορική εξέλιξη και οι εξελικτικές στρατηγικές δίνουν σε

όλα τα άτομα ίδες ευκαιρίες να επιλεγούν ως γονείς και να δημιουργήσουν απογόνους με τη χρήση γενετικών τελεστών. Αν ο γονέας είναι καλύτερος από τον απόγονο παραμένει στον πληθυσμό συνεπώς το φαινόμενο του ελιτισμού παρουσιάζεται και στη διαφορική εξέλιξη.

2.4. Προσομοιωμένη ανόπτηση

Πρόκειται για έναν από τους κλασικότερους και πιο δημοφιλείς μεταερευτικούς αλγορίθμους. Η προσομοιωμένη ανόπτηση (Simulated Annealing – SA) προτάθηκε για πρώτη φορά από τους Kirkpatrick, Gelatt και Vecchi (1983) και από τότε έχει χρησιμοποιηθεί σχεδόν σε όλους τους τομείς στους οποίους απαιτείται βελτιστοποίηση.

Το όνομα και η λειτουργία του αλγορίθμου προέρχονται από τη φυσική διαδικασία της ανόπτησης των στερεών στη μεταλλουργία. Το στερεό (μέταλλο ή γυαλί) θερμαίνεται αρχικά σε μια υψηλή θερμοκρασία και ψύχεται με αργό ρυθμό μέχρι να φτάσει σε κάποια τελική θερμοκρασία όπου έχει επιτευχθεί η ελάχιστη ενεργειακή του κατάσταση. Κατά τη διάρκεια της ψύξης του υλικού η ενέργειά του κινείται σε χαμηλότερη στάθμη, δεν αποκλείεται όμως να αυξηθεί για μικρό χρονικό διάστημα μέχρι να βρει μία νέα κατάσταση με χαμηλότερα ενεργειακά επίπεδα. Αν η αρχική θερμοκρασία είναι χαμηλή, η διαδικασία ψύξης πραγματοποιείται πολύ γρήγορα και δεν επιτυγχάνεται η ελάχιστη ενεργειακή κατάσταση με αποτέλεσμα να παραμορφώνεται η κρυσταλλική δομή του στερεού.

Η συμπεριφορά του μετάλλου κατά τη διαδικασία της ανόπτησης αντιστοιχίζεται με τα προβλήματα βελτιστοποίησης. Οι ενεργειακές καταστάσεις του μετάλλου αντιπροσωπεύουν εφικτές λύσεις, ενώ η ενέργεια αντιπροσωπεύει την αντικειμενική συνάρτηση του εξεταζόμενου προβλήματος. Συνεπώς οι τιμές τις ενέργειας σε κάθε κατάσταση αντιστοιχούν στις τιμές της αντικειμενικής συνάρτησης και η ελάχιστη ενεργειακή κατάσταση αντιστοιχεί στη βέλτιστη τιμή της αντικειμενικής συνάρτησης. Η θερμοκρασία λειτουργεί ως μια παράμετρος ελέγχου της διαδικασίας. Στον Πίνακα 2.1 δίνεται η αντιστοιχία της θερμοδυναμικής συμπεριφοράς ενός μετάλλου με ένα πρόβλημα βελτιστοποίησης.

Πίνακας 2.1. Αναλογία της ανόπτησης μετάλλου με την προσομοιωμένη ανόπτηση

Ανόπτηση μετάλλου	Προσομοιωμένη ανόπτηση
Θερμοδυναμική συμπεριφορά μετάλλου	Πρόβλημα βελτιστοποίησης
Ενεργειακές καταστάσεις μετάλλου	Εφικτές λύσεις
Ενέργεια	Αντικειμενική συνάρτηση
Αλλαγή κατάστασης	Γειτονικές λύσεις
Θερμοκρασία	Παράμετρος ελέγχου
Ελάχιστη ενεργειακή κατάσταση	Βέλτιστη λύση

2.4.1. Περιγραφή του αλγορίθμου

Η βασική ιδέα του αλγορίθμου της προσομοιωμένης ανόπτησης είναι η χρήση της τυχαίας αναζήτησης με τέτοιο τρόπο ώστε να γίνονται αποδεκτές όχι μόνο αλλαγές κατάστασης που βελτιώνουν την αντικειμενική συνάρτηση αλλά να διατηρούνται και κάποιες μεταβολές που δε δίνουν καλύτερα αποτελέσματα.

Έστω s μια λύση που έχει βρεθεί και $N(s)$ ένα σύνολο γειτονικών λύσεων της s . Οι επιλογές των εφικτών λύσεων γίνονται τυχαία μέσα από αυτό το σύνολο. Αν s' είναι μία γειτονική λύση τότε η μετακίνηση από την s στην s' είναι αποδεκτή αν ισχύει:

$$\Delta f = f(s') - f(s) \leq 0 \quad (2.11)$$

όπου f η αντικειμενική συνάρτηση του προβλήματος. Αν δεν ισχύει η παραπάνω σχέση, τότε η μετάβαση στην s' γίνεται με πιθανότητα:

$$p(\Delta f, T) = e^{-\Delta f/kT} \quad (2.12)$$

όπου:

k : η σταθερά Boltzmann η οποία συνήθως τίθεται ίση με 1 (Yang, 2010c).

T : η θερμοκρασία ελέγχου της διαδικασίας.

Για να διαπιστώσουμε αν γίνεται αποδεκτή η μεταβολή κατάστασης θεωρούμε έναν αριθμό r που έχει ληφθεί από μία ομοιόμορφη κατανομή τυχαίων αριθμών στο διάστημα $[0, 1]$. Αν ισχύει η σχέση

$$p = e^{-\Delta f/T} > r \quad (2.13)$$

η μετακίνηση στη λύση s' είναι αποδεκτή. Η p ονομάζεται πιθανότητα μετάβασης.

Η παράμετρος θερμοκρασίας T αρχικά έχει μεγάλη τιμή επιτρέποντας την ανάδειξη των καλύτερων λύσεων ενώ οι λύσεις οι οποίες επιδέχονται βελτίωση γίνονται πιο συχνά αποδεκτές. Έτσι ο αλγόριθμος δεν εγκλωβίζεται σε τοπικά βέλτιστα. Καθώς η παράμετρος T μειώνεται σταδιακά, μειώνονται και οι πιθανότητες αποδοχής των χειρότερων λύσεων και αλγόριθμος προσεγγίζει το ολικό βέλτιστο.

2.4.2. Ρύθμιση παραμέτρων

Υπάρχουν τέσσερις παράμετροι οι οποίες επηρεάζουν άμεσα τη σύγκλιση και την απόδοση του αλγορίθμου SA:

- Η αρχική θερμοκρασία
- Το χρονοπρόγραμμα ψύξης
- Το πλήθος των επαναλήψεων σε κάθε θερμοκρασία
- Το κριτήριο τερματισμού

Η επιλογή μιας καλής αρχικής θερμοκρασίας είναι ζωτικής σημασίας για τον αλγόριθμο. Δεδομένης μιας διαφοράς Δf , αν η θερμοκρασία είναι πολύ μεγάλη ($T \rightarrow \infty$), τότε $p \rightarrow 1$ που σημαίνει ότι σχεδόν όλες οι μεταβολές θα γίνουν αποδεκτές. Σε αυτήν την περίπτωση το σύστημα βρίσκεται σε πολύ υψηλή ενεργειακή κατάσταση. Ο αλγόριθμος αναγκάζεται να εκτελέσει ένα πολύ μεγάλο πλήθος επαναλήψεων και δυσκολεύεται κατά την αναζήτηση της βέλτιστης λύση. Αν η θερμοκρασία είναι πολύ μικρή ($T \rightarrow 0$), τότε σχεδόν καμία χειρότερη κατάσταση δε θα γίνεται αποδεκτή αφού $p \rightarrow 0$ και η ποικιλομορφία του αλγορίθμου περιορίζεται σημαντικά. Ο αλγόριθμος εγκλωβίζεται σε τοπικό βέλτιστο (το οποίο δεν είναι απαραίτητα ολικό βέλτιστο) και δεν μπορεί να εξερευνησει άλλες περιοχές για την αναζήτηση του ολικού βέλτιστου. Αν είναι γνωστή η μέγιστη διαφορά Δf έχει προταθεί ο ακόλουθος τύπος για τον υπολογισμό της αρχικής θερμοκρασίας (Suman και Kumar, 2006):

$$T_0 = \frac{\max(\Delta f)}{\ln p_0} \quad (2.14)$$

όπου p_0 μία πιθανότητα που παίρνει τιμές μεταξύ του 0.5 και του 0.95.

Το χρονοπρόγραμμα ψύξης ελέγχει τη διαδικασία της απόψησης προσδιορίζοντας τον τρόπο με τον οποίο μειώνεται η θερμοκρασία. Έχουν προταθεί πολλά χρονοπρογράμματα ψύξης με πιο συνηθισμένα το γεωμετρικό, το γραμμικό και το λογαριθμικό.

Το γεωμετρικό χρονοπρόγραμμα είναι ένας από τους πιο απλούς και συνηθισμένους κανόνες και μειώνει τη θερμοκρασία με τον ακόλουθο τύπο (Kirkpatrick, Gelatt και Vecchi, 1983):

$$T(t) = T_0 \cdot a^t \quad (2.15)$$

όπου:

a : μια θετική σταθερά που ονομάζεται παράγοντας ψύξης (cooling factor) και παίρνει τιμές μικρότερες της μονάδας αλλά κοντά σε αυτή. Συνήθως οι τιμές της κυμαίνονται μεταξύ του 0,8 και του 0,99.

Το μεγάλο πλεονέκτημα αυτής της μεθόδου είναι ότι για $t \rightarrow \infty$ προκύπτει ότι $T \rightarrow 0$. Άρα δεν είναι αναγκαίο να καθορίσουμε το μέγιστο αριθμό επαναλήψεων. Χρειάζεται όμως να επιτύχουμε μία διαδικασία ψύξης αρκετά αργή ώστε να επιτρέψει στο σύστημα να σταθεροποιηθεί ομαλά.

Για τη γραμμική μέθοδο έχουμε:

$$T(t) = T_0 - \beta t \quad (2.16)$$

όπου β μία σταθερή τιμή που ονομάζεται ρυθμός ψύξης και συνήθως ισούται με $(T_0 - T_f)/N$ (Yang, 2010c) δηλαδή η τελική θερμοκρασία T_f έχει καθορισθεί.

Στη λογαριθμική μέθοδο, η τιμή της θερμοκρασίας κατά την επανάληψη t ισούται με (German και German, 1984):

$$T(t) = \frac{c}{\log(t + d)} \quad (2.17)$$

όπου d μια σταθερά συνήθως ίση με 1 και $c \geq d^*$ όπου d^* ένας αριθμός τέτοιος ώστε (Hajek, 1988)

$$\sum_{t=1}^{\infty} \exp(-d^*/T(t)) = +\infty \quad (2.18)$$

Η απόφαση για τον πλήθος των επαναλήψεων σε κάθε θερμοκρασία είναι πολύ σημαντική. Συνήθως ορίζεται ένας σταθερός αριθμός επαναλήψεων για κάθε θερμοκρασία ο οποίος πρέπει να είναι αρκετά μεγάλος ώστε να εξετάζονται όλες οι καταστάσεις και ο αλγόριθμος να μπορεί να εντοπίσει τη βέλτιστη λύση.

Μια άλλη μέθοδος που έχει προταθεί (Lundy και Mess, 1986) είναι η εκτέλεση μίας μόνο επανάληψης σε κάθε θερμοκρασία και η μείωση της θερμοκρασίας να γίνεται πολύ αργά με τον ακόλουθο τύπο:

$$T(t) = \frac{t}{1 + \beta t} \quad (2.19)$$

όπου β μία μικρή θετική τιμή.

Ένας εναλλακτικός τρόπος είναι η δυναμική αλλαγή του πλήθους των επαναλήψεων καθώς ο αλγόριθμος εκτελείται. Σε χαμηλότερες θερμοκρασίες πραγματοποιείται μεγάλος αριθμός επαναλήψεων ώστε το τοπικό βέλτιστο να μπορεί να εντοπιστεί εύκολα. Σε υψηλότερες θερμοκρασίες ο αριθμός των επαναλήψεων είναι μικρότερος.

Για τον τερματισμό του αλγορίθμου έχουν προταθεί διάφορα κριτήρια. Ένα από αυτά μπορεί να είναι η τελική θερμοκρασία του συστήματος. Κατά την εκτέλεση του αλγορίθμου συνήθως αφήνουμε τη θερμοκρασία να μειωθεί μέχρι να φτάσει στο μηδέν. Αυτό όμως μπορεί να αναγκάσει τον αλγόριθμο να εκτελείται για πολύ μεγαλύτερο χρονικό διάστημα από όσο χρειάζεται. Τελικά, το κριτήριο τερματισμού μπορεί να είναι μία κατάλληλη χαμηλή θερμοκρασία στην οποία το σύστημα έχει ψυχθεί αρκετά.

Άλλο κριτήριο τερματισμού που χρησιμοποιείται είναι ο συνολικός αριθμός των επαναλήψεων του αλγορίθμου. Όταν έχουμε μεγάλο αριθμό επαναλήψεων, ο αλγόριθμος μπορεί να εκτελείται για μεγάλο χρονικό διάστημα χωρίς να βελτιώνει την αντικειμενική συνάρτηση ενώ με λίγες επαναλήψεις κινδυνεύει να μην προσεγγίσει τη βέλτιστη λύση.

Παρά τις μεθόδους που έχουν προταθεί για τις παραμέτρους του αλγορίθμου SA, απαιτείται πειραματισμός για την εύρεση των κατάλληλων τιμών τους. Οι παράμετροι καθορίζονται από τη φύση του εκάστοτε προβλήματος.

Ακολουθεί ο Ψευδοκώδικας του αλγορίθμου (Yang, 2010c).

Ψευδοκώδικας 2.3. Simulated Annealing

Objective function $f(x)$, $x = (x_1, x_2, \dots, x_n)^T$
 Set initial temperature T_0 and initial guess $x^{(0)}$
 Set final temperature T_f and max number of iterations N
 Define cooling schedule
while ($T > T_f$ and $n < N$)
 Move randomly to new locations: $x_{n+1} \leftarrow x_n + \varepsilon$ (random walk)
 Calculate $\Delta f = f(x_{n+1}) - f(x_n)$
 Accept the new solution if better
 if not improved
 Generate a random number r
 Accept if $p = \exp(-\Delta f/T) > r$
 end if
 Update the best x_* and f_*
 $n \leftarrow n + 1$
end while

2.5. Βελτιστοποίηση σμήνους σωματιδίων

Η βελτιστοποίηση σμήνους σωματιδίων (Particle Swarm Optimization – PSO) είναι ένας από τους πιο χαρακτηριστικούς πληθυσμιακούς αλγόριθμους. Αναπτύχθηκε από τους Kennedy και Eberhart (1995) και βασίζεται στη νοημοσύνη σμήνους (swarm intelligence) δηλαδή στη συλλογική συμπεριφορά αυτο-οργανωμένων συστημάτων (self-organized systems) που συναντάται στη φύση όπως σμήνη πτηνών και εντόμων ή κοπάδια ψαριών. Οι αλγόριθμοι νοημοσύνης σμήνους είναι μία ευρεία κατηγορία αλγορίθμων που εμπνέονται από τη φύση στην οποία, εκτός του PSO, ανήκουν δημοφιλείς τεχνικές όπως η βελτιστοποίηση αποικίας μυρμηγκιών (Ant Colony Optimization), οι αλγόριθμοι των μελισσών (Bee Algorithms), ο αλγόριθμος βελτιστοποίησης αναζήτησης τροφής των βακτηρίων (Bacterial Foraging Optimization Algorithm), ο αλγόριθμος της πυγολαμπίδας (Firefly Algorithm), η αναζήτηση του κούκου (Cuckoo Search) και ο αλγόριθμος της νυχτερίδας (Das, 2011) (Fister, Fister και Yang, 2013).

Ο PSO δε χρησιμοποιεί τελεστές όπως η μετάλλαξη και η διασταύρωση των εξελικτικών αλγορίθμων. Αντί αυτών βασίζεται στην τυχαιότητα των πραγματικών αριθμών και στην επικοινωνία μεταξύ των ατόμων του πληθυσμού.

2.5.1. Βασική μορφή του αλγορίθμου

Στον αλγόριθμο PSO, ένα πλήθος οντοτήτων που ονομάζονται σωματίδια (particles) ή πράκτορες (agents), τοποθετούνται στο χώρο αναζήτησης και κάθε ένα αξιολογεί την αντικειμενική συνάρτηση στην παρούσα θέση του. Κάθε σωματίδιο κινείται στο χώρο αναζήτησης και η θέση του κάθε φορά καθορίζεται τόσο από τη δική του εμπειρία όσο και από την εμπειρία του υπόλοιπου σμήνους. Ταυτόχρονα λαμβάνεται υπόψη και ο παράγοντας της τυχαιότητας. Η επόμενη επανάληψη πραγματοποιείται όταν όλα τα σωματίδια έχουν κινηθεί. Τελικά το σμήνος κινείται πολύ κοντά σε μία βέλτιστη λύση.

Κάθε πράκτορας του σμήνους περιγράφεται από τρία διανύσματα: την τρέχουσα τοποθεσία του x_i , την καλύτερη τοποθεσία p_i στην οποία έχει βρεθεί μέχρι την τρέχουσα χρονική στιγμή και την ταχύτητά του v_i .

Σε κάθε επανάληψη του αλγορίθμου, η τρέχουσα τοποθεσία αντιπροσωπεύει μία λύση του προβλήματος. Αν αυτή η θέση είναι καλύτερη από κάθε άλλη θέση στην οποία έχει βρεθεί το σωματίδιο i μέχρι εκείνη τη στιγμή, οι συντεταγμένες της αποθηκεύονται στο διάνυσμα p_i . Για κάθε επανάληψη t υπάρχει τρέχουσα καλύτερη θέση για όλα τα σωματίδια. Ο στόχος είναι να βρεθεί το ολικό βέλτιστο g μεταξύ των καλύτερων θέσεων όλων των σωματιδίων μέχρις ότου η αντικειμενική συνάρτηση να μη βελτιώνεται άλλο ή μέχρι να ολοκληρωθεί ένας καθορισμένος αριθμός επαναλήψεων.

Ο αλγόριθμος ρυθμίζει την ταχύτητα v_i και τα νέα σημεία καθορίζονται προσθέτοντας το v_i στο x_i . Ο τύπος που καθορίζει την ταχύτητα για κάθε σωματίδιο είναι ο εξής:

$$v_i(t+1) = v_i(t) + c_1 \cdot r_1 \odot [p_i(t) - x_i(t)] + c_2 \cdot r_2 \odot [g(t) - x_i(t)] \quad (2.20)$$

όπου:

c_1, c_2 : σταθερές που ονομάζονται συντελεστές επιτάχυνσης (accelerator factors).

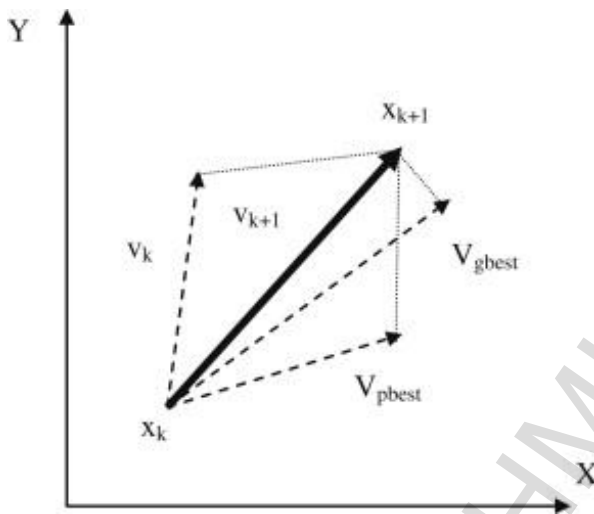
r_1, r_2 : δύο διανύσματα τυχαιών αριθμών ομοιόμορφα κατανεμημένων στο διάστημα $[0,1]$.

\odot : ο τελεστής του γινομένου Hadamard. Μεταξύ δύο $m \times n$ πινάκων $A = (a_{ij})$ και $B = (b_{ij})$ το γινόμενο Hadamard είναι ένας $m \times n$ πίνακας $C = (c_{ij})$ με $c_{ij} = a_{ij} \cdot b_{ij}$.

Οι αρχικές θέσεις των σωματιδίων πρέπει να κατανέμονται σχεδόν ομοιόμορφα ώστε να καλύπτουν πολλές περιοχές, γεγονός πολύ σημαντικό για τα πολυκριτηριακά προβλήματα. Οι αρχικές ταχύτητες όλων των σωματιδίων μπορούν να είναι ίσες με μηδέν. Η νέα θέση υπολογίζεται ως εξής:

$$x_i(t + 1) = x_i(t) + v_i(t + 1) \quad (2.21)$$

Αν και οι ταχύτητες μπορούν να πάρουν οποιαδήποτε τιμή, συνήθως ορίζεται μία ελάχιστη και μία μέγιστη τιμή. Η κίνηση ενός σωματιδίου στις δύο διαστάσεις φαίνεται στην Εικόνα 2.8 (Siddhartha, Sharma και Varun, 2012). Ο Ψευδοκώδικας 2.4 περιγράφει τη βασική μορφή του αλγορίθμου PSO (Yang, 2010a).



Εικόνα 2.8. Κίνηση σωματιδίου σε 2 διαστάσεις. Από την άλγεβρα διανυσμάτων ισχύει $v_{k+1} = x_{k+1} - x_k$ δηλαδή $x_{k+1} = x_k + v_{k+1}$.

2.5.2. Ρύθμιση παραμέτρων

Η βασική μορφή του αλγορίθμου PSO περιλαμβάνει έναν αριθμό παραμέτρων που χρειάζεται να καθοριστούν. Μία παραμέτρος είναι το μέγεθος του σμήνους δηλαδή το πλήθος των σωματιδίων. Πολλά σωματίδια αυξάνουν τον υπολογιστικό χρόνο του αλγορίθμου χωρίς απαραίτητα να βελτιώνουν την απόδοσή του. Το μέγεθος του πληθυσμού καθορίζεται εμπειρικά με βάση τις διαστάσεις και τη δυσκολία του προβλήματος. Τιμές μεταξύ των 20 και των 50 σωματιδίων είναι συνηθισμένες (Poli, Kennedy και Blackwell, 2007).

Οι συντελεστές επιτάχυνσης c_1 και c_2 της εξίσωσης (2.20) μπορούν να εκληφθούν ως παράγοντες εμπιστοσύνης που έχει κάθε σωματίδιο στον εαυτό του (c_1) και στο υπόλοιπο σμήνος (c_2). Αν $c_1 > 0$ και $c_2 = 0$ χάνεται η κοινωνική συνοχή και το κάθε σωματίδιο αναζητά αυτόνομα τη λύση. Αν $c_1 = 0$ και $c_2 > 0$ χάνεται η ατομική πρωτοβουλία και όλα τα σωματίδια κινούνται σύμφωνα με το καλύτερο σωματίδιο του πληθυσμού. Οι τιμές που παίρνουν οι συντελεστές είναι συνήθως κοντά στο 2, δηλαδή $c_1 \approx c_2 \approx 2$ (Yang, 2010a).

Είναι προφανές ότι η αλλαγή των τιμών του συντελεστών επιτάχυνσης επιφέρει δραστική αλλαγή στη συμπεριφορά του PSO. Ο αλγόριθμος μπορεί να φτάσει μέχρι και σε κατάσταση

αστάθειας, με τα σωματίδια να αυξάνουν τις ταχύτητές τους ανεξέλεγκτα. Αυτό μπορεί να συμβεί και με τις τιμές κοντά στο 2 που συνήθως επιλέγονται. Για να είναι εφικτός ο έλεγχος της κίνησης των σωματιδίων τίθενται κάποια όρια στις τιμές των ταχυτήτων. Σε κάποιες έρευνες υπάρχουν διανύσματα μέγιστης και ελάχιστης ταχύτητας και ισχύει $v_{\min} = -v_{\max}$ (Poli, Kennedy και Blackwell, 2007). Σε κάποιες άλλες υπάρχει μόνο ένα διάνυσμα v_{\max} (Eberhart και Shi, 2001) των μέγιστων τιμών των διαστάσεων της ταχύτητας.

Ψευδοκώδικας 2.4. Particle Swarm Optimization

Objective function $f(x)$, $x = (x_1, x_2, \dots, x_d)^T$

Initialize locations x_i and velocity of n particles.

$t \leftarrow t + 1$

Find g from $\min \{f(x_1), f(x_2), \dots, f(x_n)\}$

while (criterion)

$t \leftarrow t + 1$ (pseudo time or iteration counter)

for loop over all n particles and all d dimensions

Generate new velocity $v_i(t + 1)$ using equation (2.20)

Calculate new locations $x_i(t + 1) = x_i(t) + v_i(t + 1)$

Evaluate objective functions at new locations $x_i(t + 1)$

Find the current best for each particle p_i

end for

Find the current global best g

end while

Output the final results p_i and g

Αν το εύρος τιμών της ταχύτητας είναι πολύ μεγάλο ή για πολύ μεγάλες τιμές του v_{\max} , τα σωματίδια κινδυνεύουν να βρεθούν πολύ μακριά από καλές λύσεις που πιθανώς να έχουν βρει. Για πολύ μικρό εύρος τιμών ή πολύ μικρές τιμές της v_{\max} τα σωματίδια κάνουν τοπικές αναζητήσεις και δεν εξετάζουν το χώρο αναζήτησης επαρκώς. Η εύρεση των κατάλληλων ορίων εξαρτάται και πάλι από το πρόβλημα. Συνήθως η v_{\max} περιορίζεται στο 10-20% του δυναμικού εύρους των συντεταγμένων θέσης (Eberhart και Shi, 2001).

Το πλήθος των επαναλήψεων είναι επίσης μία παράμετρος που πρέπει να καθοριστεί. Με λίγες επαναλήψεις υπάρχει κίνδυνος πρόωρου τερματισμού του αλγορίθμου χωρίς να έχει βρεθεί μια ικανοποιητική λύση, ενώ με πολλές επαναλήψεις αυξάνεται ο υπολογιστικός χρόνος χωρίς απαραίτητα να προκύπτουν καλύτερες λύσεις.

2.5.3. Συντελεστής αδράνειας

Η μέγιστη ταχύτητα v_{\max} λειτουργεί ως παράγοντας περιορισμού για τον έλεγχο της σφαιρικής εξερεύνησης ενός σημείου σωματιδίων. Όπως είδαμε νωρίτερα, μία μεγαλύτερη τιμή της v_{\max} διευκολύνει τη σφαιρική αναζήτηση (εξερεύνηση), ενώ μια μικρότερη τιμή ενισχύει την τοπική αναζήτηση (εκμετάλλευση). Η ιδέα ενός συντελεστή αδράνειας (inertia weight) (Shi και Eberhart 1998), υλοποιείται για καλύτερο έλεγχο της εξερεύνησης και της εκμετάλλευσης ώστε να περιοριστεί η ανάγκη χρήσης της v_{\max} .

Η ακόλουθη εξίσωση προσδιορίζει την ταχύτητα συμπεριλαμβανομένου και του συντελεστή αδράνειας w :

$$v_i(t+1) = w \cdot v_i(t) + c_1 \cdot r_1 \odot [p_i(t) - x_i(t)] + c_2 \cdot r_2 \odot [g(t) - x_i(t)] \quad (2.22)$$

Η νέα θέση σε κάθε επανάληψη υπολογίζεται με την εξίσωση:

$$x_i(t+1) = x_i(t) + v_i(t+1) \quad (2.23)$$

Άρα η διαφορά με τη βασική μορφή του PSO είναι ότι η ταχύτητα της προηγούμενης επανάληψης πολλαπλασιάζεται με το συντελεστή w .

Στην απλούστερη περίπτωση, ο συντελεστής αδράνειας είναι μία σταθερή τιμή. Όμως οι έρευνες έχουν δείξει ότι μπορεί να επιτευχθεί καλύτερη απόδοση όταν ο συντελεστής μειώνεται γραμμικά ξεκινώντας από μία μεγάλη τιμή, για παράδειγμα 0.9 και καταλήγοντας σε μία μικρότερη για παράδειγμα 0.4 (Poli, Kennedy και Blackwell, 2007). Η γραμμική εξίσωση που υπολογίζει την τιμή του w για κάθε επανάληψη είναι (Bansal κ.ά., 2011):

$$w_{iter} = w_{\max} - \frac{w_{\max} - w_{\min}}{iter_{\max}} \times iter \quad (2.24)$$

όπου:

w_{\max} : η μέγιστη (αρχική) τιμή του w

w_{\min} : η ελάχιστη (τελική) τιμή του w

$iter$: ο αριθμός των επαναλήψεων σε μια δεδομένη χρονική στιγμή

$iter_{\max}$: ο μέγιστος αριθμός των επαναλήψεων

Για μεγάλες τιμές του w τα σωματίδια κάνουν εκτενείς εξερευνήσεις στο χώρο αναζήτησης, ενώ για μικρότερες τιμές περιορίζονται στην αναζήτηση τοπικών βέλτιστων. Η κατάλληλη επιλογή ενός συντελεστή αδράνειας επιτυγχάνει ισορροπία μεταξύ της εκμετάλλευσης και της εξερεύνησης ώστε το σμήνος να χρειάζεται λιγότερες επαναλήψεις για να βρει μια ικανοποιητική βέλτιστη λύση.

Εκτός από τη γραμμική μείωση του συντελεστή αδράνειας έχουν προταθεί κι άλλες στρατηγικές για τη ρύθμισή του (Poli, Kennedy και Blackwell, 2007) (Bansal κ.ά., 2011) όπως η χρήση ασαφών συστημάτων, η τυχαία επιλογή του μέσα από ένα σύνολο ομοιόμορφα κατανεμημένων τιμών και η σταδιακή αύξηση του συντελεστή αδράνειας.

Μετά από πολλές εφαρμογές του PSO με τη χρήση του συντελεστή αδράνειας βρέθηκε ότι ο παράγοντας της μέγιστης ταχύτητας v_{\max} δεν μπορεί να εξαλειφθεί, αλλά ο αλγόριθμος λειτουργεί καλά αν οι διαστάσεις της v_{\max} περιοριστούν στις τιμές των διαστάσεων της x_{\max} δηλαδή στις μέγιστες τιμές του δυναμικού εύρους για κάθε συντεταγμένη θέσης (Poli, Kennedy και Blackwell, 2007) (Ratnaweera, Halgamuge και Watson, 2004). Με την εισαγωγή του συντελεστή αδράνειας δε χρειάζεται να πειραματιζόμαστε με την μέγιστη ταχύτητα για να επιτύχουμε καλύτερη απόδοση του αλγορίθμου.

2.5.4. Συντελεστές περιορισμού

Οι συντελεστές περιορισμού (constriction factors) αναπτύχθηκαν στα πλαίσια μιας προσέγγισης (Clerc, 1999) για τη εξασφάλιση της σύγκλισης του αλγορίθμου σε κάποια λύση χωρίς την ανάγκη χρήσης της παραμέτρου v_{\max} . Οι εξισώσεις ταχύτητας και θέσης είναι αντίστοιχα:

$$v_i(t+1) = \chi \cdot [v_i(t) + \varphi_1 \odot [p_i(t) - x_i(t)] + \varphi_2 \odot (g(t) - x_i(t))] \quad (2.25)$$

και

$$x_i(t+1) = x_i(t) + v_i(t+1) \quad (2.26)$$

όπου

$$\chi = \frac{2\kappa}{|2 - \varphi - \sqrt{\varphi(\varphi - 4)}|} \quad (2.27)$$

με

$$\varphi = \varphi_1 + \varphi_2 \quad (2.28)$$

όπου

$$\varphi_1 = c_1 r_1 \quad \text{και} \quad \varphi_2 = c_2 r_2 \quad (2.29)$$

Η παράμετρος κ της εξίσωσης (2.27) ανήκει στο διάστημα $[0,1]$ και ελέγχει την εξερεύνηση και την εκμετάλλευση του PSO. Για $\kappa \approx 0$ προκύπτει τοπική εκμετάλλευση και γρήγορη σύγκλιση. Για $\kappa \approx 1$ προκύπτει σφαιρική εξερεύνηση και αργή σύγκλιση. Έχει αποδειχθεί ότι η σύγκλιση του αλγορίθμου είναι εγγυημένη για $\varphi \geq 4$ (Clerc και Kennedy, 2002) (Yang, 2010a).

Αντίθετα με τη μέθοδο της αδράνειας, η μέθοδος του περιορισμού εξασφαλίζει ότι το σμήνος θα συγκλίνει χωρίς τη χρήση της v_{\max} . Εντούτοις, μετά από μια σειρά πειραματισμών και εφαρμογών προέκυψε το συμπέρασμα ότι μια καλύτερη μέθοδος είναι ο περιορισμός των διαστάσεων της v_{\max} στις τιμές των διαστάσεων της x_{\max} δηλαδή στις μέγιστες τιμές του δυναμικού εύρους για κάθε συντεταγμένη θέσης (Eberhart και Shi, 2001).

Αξίζει να σημειώσουμε ότι η το μοντέλο του περιορισμού είναι αλγεβρικά ισοδύναμο με το μοντέλο του συντελεστή αδράνειας. Στην εξίσωση (2.25) αν θέσουμε $\chi \equiv w$, $\chi\varphi_1 \equiv c_1 r_1$ και $\chi\varphi_2 \equiv c_2 r_2$ προκύπτει η εξίσωση (2.22).

2.5.5. Πλήρως ενημερωμένο σμήνος σωματιδίων

Στον κλασικό PSO, η κίνηση κάθε σωματιδίου καθορίζεται από προηγούμενες δικές του κινήσεις και από κινήσεις του γειτονικού σωματιδίου που έχει βρει την ολικά καλύτερη λύση. Οι πληροφορίες από τα υπόλοιπα γειτονικά σωματίδια παραμένουν αχρησιμοποίητες. Οι Mendes, Kennedy και Neves (2003) παρουσίασαν μία παραλλαγή του αλγορίθμου PSO όπου κάθε σωματίδιο επηρεάζεται από όλα τα γειτονικά του σωματίδια. Ένας πληθυσμός σωματιδίων τα οποία επηρεάζονται από γειτονικά τους, συγκροτεί ένα πλήρως ενωμερωμένο σμήνος σωματιδίων (Fully

Informed Particle Swarm – FIPS). Οι εξισώσεις της ταχύτητας και της θέσης των σωματιδίων του σμήνους είναι οι εξής (Poli, Kennedy και Blackwell, 2007):

$$v_i(t + 1) = \chi \cdot [v_i(t) + \frac{1}{K_i} \sum_{n=1}^{K_i} c \cdot r \odot (p_n(t) - x_i(t))] \quad (2.30)$$

και

$$x_i(t + 1) = x_i(t) + v_i(t + 1) \quad (2.31)$$

όπου

K_i : ο αριθμός των γειτονικών σωματιδίων για το σωματίδιο i

p_n : η καλύτερη θέση του n -οστού γειτονικού σωματιδίου για το σωματίδιο i

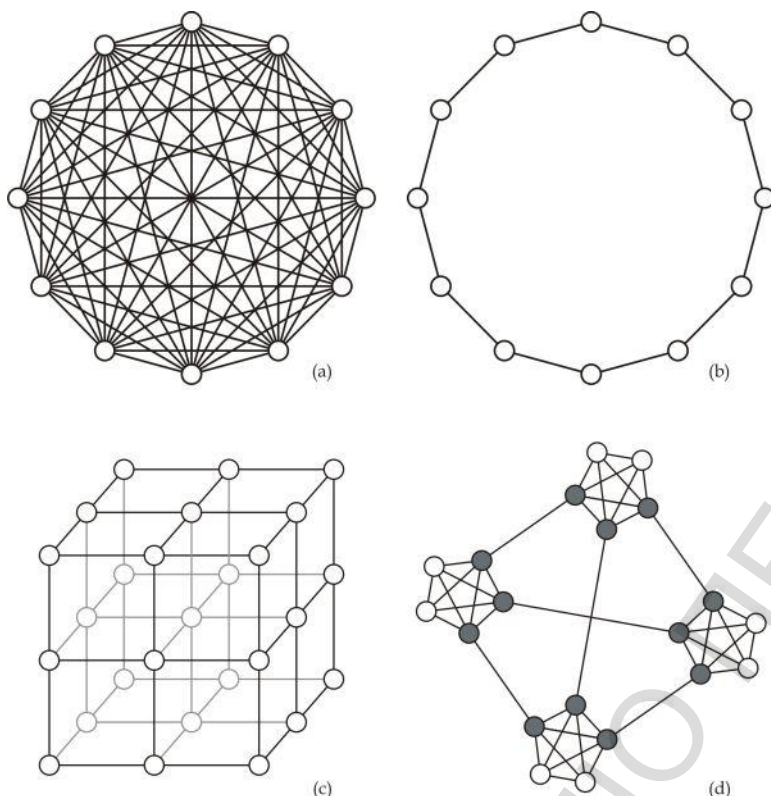
Ένα πλήρως ενημερωμένο σμήνος σωματιδίων μπορεί να βρει καλύτερες λύσεις και σε λιγότερες επαναλήψεις από τις προηγούμενες μορφές του αλγορίθμου αλλά εξαρτάται από την τοπολογία του πληθυσμού.

2.5.6. Τοπολογία πληθυσμού

Η τοπολογία σε ένα σμήνος σωματιδίων περιγράφει τον τρόπο με τον οποίο οργανώνονται οι γειτονίες των σωματιδίων στο σμήνος. Λέμε ότι δύο σωματίδια βρίσκονται στην ίδια γειτονιά ή είναι γείτονες όταν αλληλεπιδρούν μεταξύ τους, δηλαδή η κίνηση του ενός εξαρτάται από την κίνηση του άλλου. Η εξάρτηση αυτή δεν είναι πάντα αμφίδρομη, δηλαδή μπορεί ένα σωματίδιο i να εξαρτάται από ένα σωματίδιο j αλλά το j να μην εξαρτάται από το i . Η τοπολογία παίζει μεγάλο ρόλο στην απόδοση του PSO.

Οι τοπολογίες που συναντώνται συχνότερα έχουν αμφίδρομες εξαρτήσεις και είναι οι εξής:

- g best ή πλήρως συνδεδεμένη τοπολογία (fully connected topology): Κάθε σωματίδιο έχει γείτονες όλα τα υπόλοιπα σωματίδια (Εικόνα 2.9a) και η μετακίνησή του προς τη βέλτιστη λύση επηρεάζεται από το σύνολο του σμήνους. Με την πλήρως συνδεδεμένη τοπολογία ο αλγόριθμος συγκλίνει ταχύτερα και εγκλωβίζεται σε τοπικά βέλτιστα ευκολότερα.
- l best ή τοπολογία δακτυλίου (ring topology): Κάθε σωματίδιο επικοινωνεί με τους n άμεσους γείτονές του. Στην Εικόνα 2.9.b έχουμε $n = 2$. Η κίνηση των σωματιδίων προς τη βέλτιστη λύση επηρεάζεται μόνο από τους πιο κοντινούς τους γείτονες. Εδώ η σύγκλιση είναι πιο αργή αλλά γίνεται μεγαλύτερη εξερεύνηση του χώρου αναζήτησης.
- Τοπολογία Von Neumann: Τα σωματίδια επικοινωνούν με τέτοιο τρόπο που οι γειτονιές τους δημιουργούν ένα πλέγμα όπως φαίνεται στην Εικόνα 2.9.c.
- Τοπολογία συστοιχιών (cluster topology): Τα σωματίδια ενώνονται μεταξύ τους και οργανώνονται σε συστοιχίες σωματιδίων. Υπάρχουν σωματίδια που συνδέουν τη συστοιχία τους με άλλες. Στην Εικόνα 2.9.d φαίνονται 4 συστοιχίες.



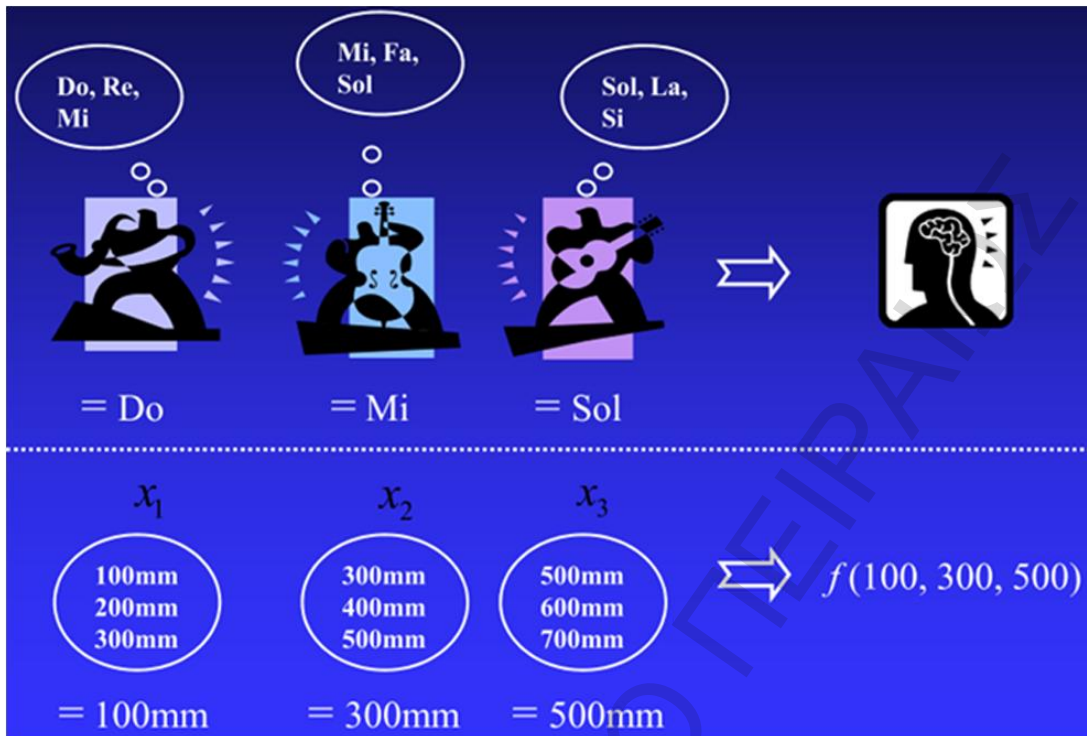
Εικόνα 2.9. Τοπολογίες σωματιδίων. (α) Πλήρως συνδεδεμένη τοπολογία, (β) Τοπολογία δακτυλίου, (γ) Τοπολογία Von Neumann, (δ) Τοπολογία συστοιχιών

2.6. Αναζήτηση αρμονίας

Η αναζήτηση αρμονίας (Harmony Search – HS) είναι ένας σχετικά νέος μεταερευνητικός πληθυσμιακός αλγόριθμος που αναπτύχθηκε από τους Geem, Kim και Loganathan (2001). Μιμείται τη διαδικασία του μουσικού «αυτοσχεδιασμού» όπου σε μία μουσική πρόβα, κάθε μουσικός μεταβάλλει τον τόνο του μουσικού οργάνου του έτσι ώστε όλοι μαζί οι τόνοι των μουσικών να επιτύχουν μία τέλεια κατάσταση αρμονίας η οποία καθορίζεται από αισθητικά κριτήρια. Πρόκειται για έναν πολύ επιτυχημένο αλγόριθμο με εφαρμογές σε ένα πλήθος προβλημάτων όπως η δρομολόγηση στόλου οχημάτων, η σχεδίαση δικτύων διανομής νερού και η κατασκευή ωρολογίων προγραμμάτων (Alia και Mandava, 2011).

Η διαδικασία αναζήτησης της βέλτιστης λύσης σε ένα πρόβλημα βελτιστοποίησης μπορεί να παραλληλιστεί με τη διαδικασία του αυτοσχεδιασμού των μουσικών, όπως φαίνεται και στην Εικόνα 2.10 (Gao κ.ά., 2012). Η αναλογία μεταξύ αυτοσχεδιασμού και βελτιστοποίησης είναι η ακόλουθη (Alia και Mandava, 2011):

- Κάθε μουσικός αντιστοιχεί σε μία μεταβλητή απόφασης.
- Ο τόνος ενός μουσικού οργάνου αντιστοιχεί στην τιμή μίας μεταβλητής απόφασης.
- Η μουσική αρμονία σε συγκεκριμένη χρονική στιγμή αντιστοιχεί σε ένα διάνυσμα λύσης σε μια συγκεκριμένη επανάληψη άρα η τέλεια αρμονία αντιστοιχεί στη βέλτιστη λύση.
- Η αισθητική του κοινού αντιστοιχεί την αντικειμενική συνάρτηση του προβλήματος βελτιστοποίησης.



Εικόνα 2.10. Αναλογία μεταξύ μουσικού αυτοσχεδιασμού και βελτιστοποίησης.

Μπορούμε να περιγράψουμε πιο εύκολα τον αλγόριθμο HS όταν γνωρίζουμε πως συμπεριφέρεται ένας μουσικός κατά τη διαδικασία του αυτοσχεδιασμού. Όταν ένας μουσικός αυτοσχεδιάζει, έχει τρεις πιθανές επιλογές:

- Να παίξει ένα μουσικό κομμάτι (μία σειρά τόνων σε αρμονία) που ήδη γνωρίζει, δηλαδή που βρίσκεται στη μνήμη του.
- Να παίξει κάτι παρόμοιο με ένα κομμάτι που γνωρίζει δηλαδή να μεταβάλλει ελάχιστα τον τόνο
- Να συνθέσει ένα νέο κομμάτι ή να παίξει τυχαίες νότες

Διαμορφώνοντας αυτές τις επιλογές για ένα πρόβλημα βελτιστοποίησης, έχουμε αντίστοιχα τις παρακάτω ενέργειες:

- τη χρήση της μνήμης αρμονίας (harmony memory – HM)
- την προσαρμογή του τόνου (pitch adjustment)
- την τυχαία επιλογή

Η χρήση της μνήμης αρμονίας είναι πολύ σημαντική και είναι παρόμοια με την επιλογή των ατόμων με τη μεγαλύτερη καταλληλότητα που συναντάται στους εξελικτικούς αλγόριθμους. Έτσι εξασφαλίζεται ότι οι καλύτερες αρμονίες θα μεταφερθούν στη νέα μνήμη αρμονίας. Για να χρησιμοποιήσουμε αυτή τη μνήμη πιο αποτελεσματικά ορίζεται μία παράμετρος που ονομάζεται βαθμός θεώρησης της μνήμης αρμονίας (Harmony Memory Considering Rate – HMCR). Αν αυτή η παράμετρος είναι πολύ χαμηλή, πολύ λίγες καλές αρμονίες διατηρούνται στη μνήμη και η σύγκλιση μπορεί να είναι πολύ αργή. Αντίθετα αν είναι πολύ υψηλή (κοντά στο 1), σχεδόν όλες οι αρμονίες παραμένουν στη μνήμη άρα είναι δύσκολο να εξερευνηθούν νέες αρμονίες.

Στη δεύτερη ενέργεια, έχουμε μία μικρή μεταβολή μίας μεταβλητή της μνήμης αρμονίας. Θεωρητικά η μεταβολή του τόνου μπορεί να είναι γραμμική ή μη γραμμική αλλά πρακτικά χρησιμοποιείται η γραμμική μεταβολή. Η μεταβολή του τόνου είναι λειτουργία παρόμοια με τον τελεστή μετάλλαξης στους γενετικούς αλγόριθμους. Μπορούμε να θεωρήσουμε μία παράμετρο ελέγχου που ονομάζεται βαθμός προσαρμογής του τόνου (pitch adjusting rate – PAR). Αν η παράμετρος είναι πολύ χαμηλή, σπάνια θα υπάρχει αλλαγή. Αν είναι πολύ υψηλή ο αλγόριθμος ενδέχεται να μη συγκλίνει.

Η τρίτη ενέργεια είναι η τυχαία επιλογή, η οποία αυξάνει την ποικιλία των λύσεων. Αν και η προσαρμογή του τόνου έχει παρόμοια λειτουργία, περιορίζεται σε ένα συγκεκριμένο εύρος τιμών άρα αντιστοιχεί στην τοπική αναζήτηση. Η τυχαία επιλογή μπορεί να οδηγήσει το σύστημα σε εξερεύνηση περισσότερων περιοχών ώστε να φτάσει στη βέλτιστη λύση.

2.6.1. Περιγραφή του αλγορίθμου

Ο αλγόριθμος HS αποτελείται από πέντε βήματα (Geem κ.ά., 2005):

1. Αρχικοποίηση των παραμέτρων του αλγορίθμου
2. Αρχικοποίηση της μνήμης αρμονίας
3. Αυτοσχεδιασμός μιας νέας αρμονίας
4. Ενημέρωση της μνήμης αρμονίας
5. Έλεγχος των κριτηρίων τερματισμού

Στο πρώτο βήμα, οι παράμετροι που πρέπει να καθοριστούν είναι:

- το μέγεθος της μνήμης αρμονίας (harmony memory size – HMS) δηλαδή ο αριθμός των διανυσμάτων λύσης στη μνήμη
- ο βαθμός θεώρησης της μνήμης αρμονίας (HMCR) που ανήκει στο διάστημα $[0,1]$ και συνήθως παίρνει τιμές μεταξύ 0.7 και 0.95 (Yang, 2010a)
- ο βαθμός προσαρμογής του τόνου (PAR) που ανήκει στο διάστημα $[0,1]$ και συνήθως παίρνει τιμές μεταξύ 0.1 και 0.5 (Yang, 2010a)
- το κριτήριο τερματισμού δηλαδή ο αριθμός των αυτοσχεδιασμών NI .

Η μνήμη αρμονίας (HM) είναι μία μήτρα μεγέθους HMS όπου κάθε διάνυσμά της αντιπροσωπεύει μία λύση όπως βλέπουμε στη σχέση (2.32). Στο δεύτερο βήμα οι λύσεις δημιουργούνται τυχαία και ταξινομούνται με βάση της τιμής της αντικειμενικής συνάρτησης του προβλήματος. Για $f(x^1) \leq f(x^2) \leq \dots \leq f(x^{HMS-1}) \leq f(x^{HMS})$ έχουμε:

$$HM = \begin{bmatrix} x_1^1 & x_2^1 & \dots & x_{N-1}^1 & x_N^1 & f(x^1) \\ x_1^2 & x_2^2 & \dots & x_{N-1}^2 & x_N^2 & f(x^2) \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_1^{HMS-1} & x_2^{HMS-1} & \dots & x_{N-1}^{HMS-1} & x_N^{HMS-1} & f(x^{HMS-1}) \\ x_1^{HMS} & x_2^{HMS} & \dots & x_{N-1}^{HMS} & x_N^{HMS} & f(x^{HMS}) \end{bmatrix} \quad (2.32)$$

Το τρίτο βήμα αποτελεί τον ακρογωνιαίο λίθο του αλγορίθμου. Σε αυτό το βήμα, ο αλγόριθμος HS παράγει (αυτοσχεδιάζει) ένα νέο διάνυσμα αρμονίας $x' = (x'_1, x'_2, \dots, x'_N)$ ακολουθώντας τρεις κανόνες: τη θεώρηση της HM, την τυχαία θεώρηση και την προσαρμογή του τόνου. Στη θεώρηση της μνήμης, οι τιμές ενός νέου διανύσματος αρμονίας λαμβάνονται τυχαία από τις τιμές που είναι αποθηκευμένες στη HM με πιθανότητα $HMCR$. Επομένως η τιμή της μεταβλητής απόφασης x'_1 επιλέγεται τυχαία από τις τιμές $\{x_1^1, x_1^2, \dots, x_1^{HMS}\}$. Ομοίως οι επόμενες μεταβλητές απόφασης

x'_2, x'_3, \dots, x'_N επιλέγονται με τον ίδιο τρόπο και με πιθανότητα $HMCR$. Η χρήση της HM είναι παρόμοια με το στάδιο όπου ο μουσικός χρησιμοποιεί τη μνήμη του για να παράγει έναν εύηχο τόνο. Αυτό το βήμα εξασφαλίζει ότι οι καλές αρμονίες θα περιλαμβάνονται στα νέα διανύσματα αρμονίας.

Οι μεταβλητές απόφασης x'_i που δε λαμβάνονται από την HM σύμφωνα με την παράμετρο $HMCR$, επιλέγονται τυχαία ανάλογα με το εύρος τιμών τους, X_i . Συνεπώς η τυχαία επιλογή γίνεται με πιθανότητα $1 - HMCR$ και αυξάνει την ποικιλία των λύσεων ώστε το σύστημα να εξερευνήσει περισσότερες περιοχές και να φθάσει στη βέλτιστη λύση.

Οι θεωρήσεις της HM και η τυχαία θεωρήσεις συνοψίζονται στην παρακάτω εξίσωση:

$$x'_i \leftarrow \begin{cases} x'_i \in \{x_i^1, x_i^2, \dots, x_i^{HMS}\} & \text{με πιθανότητα } HMCR \\ x'_i \in X_i & \text{με πιθανότητα } 1 - HMCR \end{cases} \quad (2.33)$$

Η τυχαία επιλογή μπορεί να γίνει με τη βοήθεια της εξίσωσης:

$$x'_i = x_{i,\min} + rand \cdot (x_{i,\max} - x_{i,\min}) \quad (2.34)$$

όπου $rand$ ένας τυχαίος αριθμός στο διάστημα $(0,1)$ και $x_{i,\min}$, $x_{i,\max}$ η ελάχιστη και η μέγιστη τιμή αντίστοιχα του εύρους τιμών X_i .

Για να εξερευνηθεί περισσότερο ο χώρος αναζήτησης, εξετάζεται η προσαρμογή του τόνου για κάθε μεταβλητή απόφασης του νέου διανύσματος αρμονίας $x' = (x'_1, x'_2, \dots, x'_N)$ η οποία προέρχεται από την HM. Για κάθε τέτοια μεταβλητή απόφασης x'_i , ο τόνος μεταβάλλεται με πιθανότητα PAR όπως βλέπουμε στην παρακάτω σχέση.

$$x'_i \leftarrow \begin{cases} \text{μεταβολή του τόνου με πιθανότητα } PAR \\ \text{διατήρηση του τόνου με πιθανότητα } 1 - PAR \end{cases} \quad (2.35)$$

Συνεπώς η πιθανότητα προσαρμογής του τόνου για μια οποιαδήποτε μεταβλητή απόφασης x'_i του νέου διανύσματος αρμονίας x' ισούται με $HMCR \times PAR$.

Κάθε τόνος (δηλαδή κάθε μεταβλητή απόφασης) προσαρμόζεται ως εξής:

$$x'_{i,new} = x'_{i,old} \pm rand \cdot bw \quad (2.36)$$

όπου

$rand$: μία τυχαία τιμή που λαμβάνεται από μία ομοιόμορφη κατανομή στο διάστημα $(0,1)$.

bw : το εύρος ζώνης (bandwidth) που ελέγχει την προσαρμογή του τόνου και η τιμή του εξαρτάται αν το πρόβλημα είναι διακριτό ή συνεχές.

Στην πραγματικότητα το bw καθορίζει το μέγεθος των μεταβολών στα στοιχεία του νέου διανύσματος. Ο τρόπος με τον οποίο μεταβάλλονται τα στοιχεία του νέου διανύσματος αρμονίας είναι ανάλογος με τη συμπεριφορά των μουσικών όταν αλλάζουν ελάχιστα τη συχνότητα του τόνου για να παράγουν καλύτερες αρμονίες. Ουσιαστικά, η προσαρμογή του τόνου της εξίσωσης (2.36) είναι μία τυχαία διαδρομή.

Για να ανανεωθεί η HM αξιολογείται κάθε νέο διάνυσμα x' . Αν η τιμή της αντικειμενικής συνάρτησης $f(x')$ είναι καλύτερη από την τιμή του χειρότερου διανύσματος αρμονίας που είναι αποθηκευμένο στην HM τότε το χειρότερο διάνυσμα αντικαθίσταται από το νέο διάνυσμα αρμονίας x' .

Τα βήματα 4 και 5 επαναλαμβάνονται έως ότου ικανοποιηθεί το κριτήριο τερματισμού που συνήθως είναι η συμπλήρωση του μέγιστου αριθμού αυτοσχεδιασμών (επανάληψεων) NI .

Ακολουθεί ο Ψευδοκώδικας του αλγορίθμου HS.

Ψευδοκώδικας 2.5. Harmony Search

Objective function $f(x)$, $x = (x_1, x_2, \dots, x_N)^T$

Define parameters HMS , $HMCR$, PAR

Define number of variables N

Define maximum number of iterations NI

Define initial HM

while (*iteration number* $\leq NI$)

while ($i < N$)

if ($rand(0,1) \leq HMCR$)

$x'_i \leftarrow x_i^j$, where $j = rand[1, HMS]$ and $j \in \mathbb{N}$

if ($rand(0,1) \leq PAR$)

$x'_i \leftarrow x'_i + rand(0,1) \times bw$

end if

else

$x'_i \leftarrow x_{i,min} + rand(0,1) \times (x_{i,max} - x_{i,min})$

end if

end while

if ($f(\text{new harmony vector}) \leq f(\text{worst harmony vector})$)

 replace the worst harmony vector with new harmony vector

end if

end while

Find the best solution

2.6.2. Βελτιωμένες μορφές του αλγορίθμου

Η μεταβολή ορισμένων χαρακτηριστικών της αρμονικής αναζήτησης ή η εισαγωγή στοιχείων από άλλες μεθόδους βελτιστοποίησης έχουν βελτιώσει σημαντικά την απόδοση της βασικής μορφής του αλγορίθμου που μόλις περιγράψαμε. Μία από τις πιο δημοφιλείς αναβαθμισμένες μορφές του HM είναι η βελτιωμένη αναζήτηση αρμονίας (Improved Harmony Search – IHS) (Mahdavi, Fesanghary και Damangir, 2007). Η διαφορά με τη βασική μορφή που παρουσιάστηκε είναι ότι ο βαθμός προσαρμογής του τόνου (PAR) και το εύρος ζώνης (bw) μεταβάλλονται δυναμικά. Η τιμή της παραμέτρου PAR αυξάνεται γραμμικά σε κάθε επανάληψη με την ακόλουθη εξίσωση:

$$PAR(gn) = PAR_{\min} + \frac{PAR_{\max} - PAR_{\min}}{NI} \times gn \quad (2.37)$$

όπου

gn : ο αριθμός των γενεών (επαναλήψεων)

$PAR(gn)$: η τιμή της παραμέτρου PAR για κάθε γενιά gn

PAR_{\min} , PAR_{\max} : ο ελάχιστος και ο μέγιστος βαθμός προσαρμογής του τόνου αντίστοιχα.

Το εύρος ζώνης bw μειώνεται εκθετικά σε κάθε επανάληψη του IHS με την εξίσωση:

$$bw(gn) = bw_{\max} \exp(c \cdot gn) \quad (2.38)$$

όπου

$$c = \frac{\ln(bw_{\min}/bw_{\max})}{NI} \quad (2.39)$$

$bw(gn)$: η τιμή της παραμέτρου bw για κάθε γενιά (επανάληψη) gn

bw_{\min} , bw_{\max} : η ελάχιστη και η μέγιστη τιμή του εύρους ζώνης αντίστοιχα.

Μία άλλη βελτιωμένη εκδοχή είναι η αναζήτηση αρμονίας του ολικού βέλτιστου (Global-best Harmony Search – GHS) (Omran και Mahdavi, 2008) και αναπτύχθηκε ώστε να μην είναι αναγκαίος ο καθορισμός ορίων για το bw που συμβαίνει στον IHS. Ο GHS είναι εμπνευσμένος από τον αλγόριθμο PSO όπου η κίνηση κάθε σωματιδίου εξαρτάται από την θέση του καλύτερου σωματιδίου του πληθυσμού. Λειτουργεί όπως ο IHS, με τη διαφορά ότι δε χρησιμοποιείται η παράμετρος bw και η νέα μεταβλητή απόφασης λαμβάνεται τυχαία από την καλύτερη αρμονία της HM όπως φαίνεται και στον Ψευδοκώδικα 2.6 του τρίτου βήματος.

Ψευδοκώδικας 2.6. Τρίτο βήμα του αλγορίθμου Global-best Harmony Search

```

while ( $i < N$ )
  if ( $rand(0,1) \leq HMCR$ )
     $x'_i \leftarrow x_i^j$ , where  $j = rand[1, HMS]$  and  $j \in \mathbb{N}$ 
    if ( $rand(0,1) \leq PAR(gn)$ )
       $x'_i \leftarrow x_k^{best}$ , where  $best$  is the index of the best harmony,  $k = rand[1, N]$  and  $k \in \mathbb{N}$ 
    end if
  else
     $x'_i \leftarrow x_{i,min} + rand(0,1) \times (x_{i,max} - x_{i,min})$ 
  end if
end while

```

Η διαφορική αναζήτηση αρμονίας (Differential Harmony Search – DHS) (Chakraborty κ.ά., 2009) είναι ακόμη μία αποδοτικότερη μορφή του HS όπου η προσαρμογή του τόνου αντικαθίσταται με την λειτουργία της μετάλλαξης της κλασικής διαφορικής εξέλιξης (DE/rand/1/bin). Το διάνυσμα βάσης είναι το νέο διάνυσμα που παράγεται από τη θεώρηση μνήμης και την τυχαία θεώρηση που είδαμε στην εξίσωση (2.33). Στη συνέχεια, στο νέο διάνυσμα προστίθεται η σταθμισμένη διαφορά δύο τυχαίων διανυσμάτων που βρίσκονται στην HM. Η παράμετρος PAR παραλείπεται και όλα τα νέα διανύσματα παράγονται σύμφωνα με τη σχέση:

$$x'_{i,new} = x'_{i,old} + F(x_{r1} - x_{r2}) \quad (2.40)$$

2.7. Αποικία τεχνητών μελισσών

Οι αλγόριθμοι μελισσών (Bee Algorithms) αποτελούν μία ξεχωριστή κατηγορία μεταερευτικών μεθόδων που είναι εμπνευσμένες από τον αξιοθαύμαστο τρόπο που αναζητούν την τροφή τους οι πραγματικές μέλισσες. Ήδη από τη δεκαετία του 1990 έχουν προταθεί τεχνικές (Yonezawa και Kikuchi, 1996) (Sato και Hagiwara, 1997) που περιλαμβάνουν στοιχεία της συμπεριφοράς των πραγματικών μελισσών. Η ανάπτυξή τους ως ένα ξεχωριστό πεδίο μεταερευτικών αλγορίθμων ξεκίνησε στα μέσα της δεκαετίας του 2000 με τον αλγόριθμο των μελιποφόρων μελισσών (Honey Bee Algorithm – HBA) (Nakrani και Tovey, 2004). Μέχρι και σήμερα έχει αναπτυχθεί μία ποικιλία αλγορίθμων των μελισσών μεταξύ των οποίων η αποικία τεχνητών μελισσών (Artificial Bee Colony – ABC).

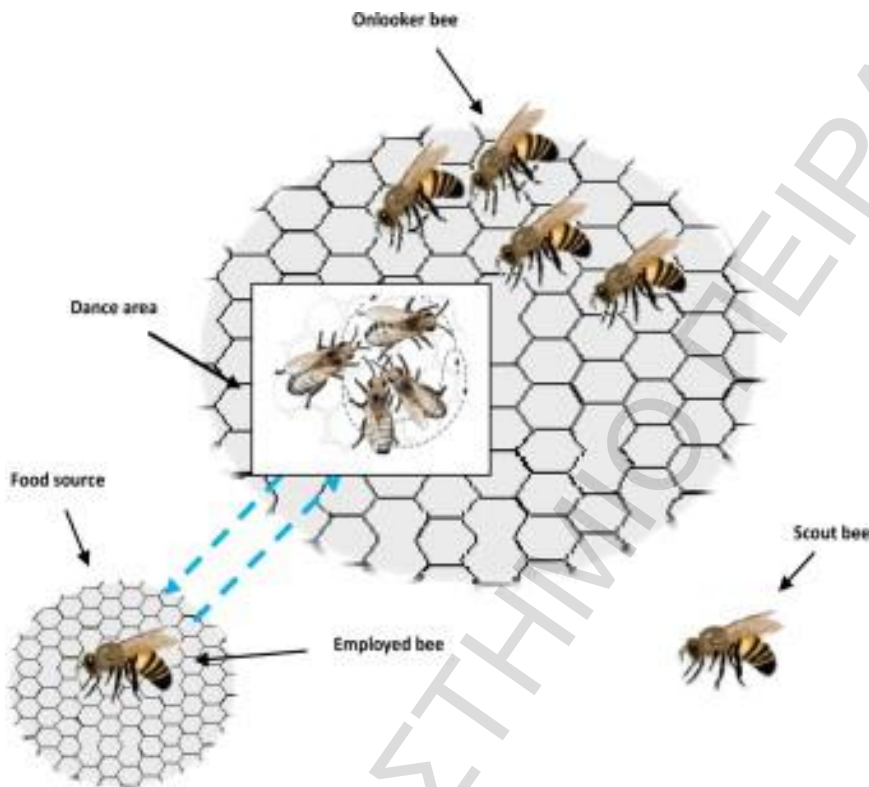
2.7.1. Περιγραφή του αλγορίθμου

Στον αλγόριθμο ABC (Karaboga και Basturk, 2007), η αποικία των τεχνητών μελισσών περιλαμβάνει τρεις ομάδες: τις μέλισσες εργάτριες (employed bees), τις παρατηρήτριες (onlookers) και τις ανιχνεύτριες (scouts). Παρατηρήτρια είναι η μέλισσα που αποφασίζει αν θα επιλέξει μία πηγή τροφής στην οποία έχει φθάσει προηγουμένως η μέλισσα εργάτρια. Ανιχνεύτρια είναι η μέλισσα που εκτελεί μία τυχαία αναζήτηση. Στον αλγόριθμο ABC, το ½ των ατόμων του πληθυσμού της αποικίας αποτελούν τις μέλισσες εργάτριες και το υπόλοιπο ½ αποτελούν τις παρατηρήτριες. Για κάθε πηγή τροφής, υπάρχει μόνο μία μέλισσα εργάτρια. Μία εργάτρια, της οποίας η πηγή τροφής έχει εξερευνηθεί πολλές φορές, γίνεται ανιχνεύτρια. Το μοντέλο μιας αποικίας μελισσών φαίνεται στην Εικόνα 2.11 (Kaviani κ.ά., 2013).

Κάθε κύκλος (επανάληψη) αναζήτησης στον αλγόριθμο ABC αποτελείται από τρία βήματα (Karaboga και Basturk, 2007): α) την αποστολή των εργατριών στις πηγές τροφής ώστε να υπολογίσουν τις ποσότητες του νέκταρ, β) την επιλογή των πηγών τροφής από τις παρατηρήτριες αφού πρώτα ενημερωθούν από τις εργάτριες και τον υπολογισμό της ποσότητας του νέκταρ στις τροφές, γ) τον καθορισμό των ανιχνεύτριων μελισσών και την αποστολή τους σε άλλες πιθανές πηγές τροφής.

Στο στάδιο αρχικοποίησης, οι μέλισσες επιλέγουν τυχαία ένα πλήθος τοποθεσιών τροφής και υπολογίζονται οι ποσότητες του νέκταρ για τις τοποθεσίες. Στη συνέχεια, αυτές οι μέλισσες εισέρχονται στην κυψέλη και μοιράζονται τις πληροφορίες για τις ποσότητες του νέκταρ με τις μέλισσες που περιμένουν στην κυψέλη. Η διαδικασία της μετάδοσης των πληροφοριών ονομάζεται χορός (dance) και το τμήμα της κυψέλης στο οποίο λαμβάνει χώρα ονομάζεται περιοχή χορού (dance area). Μετά το στάδιο της αρχικοποίησης, κάθε μέλισσα εργάτρια πηγαίνει στην περιοχή της καλύτερης πηγής τροφής που έχει επισκεφθεί η οποία είναι αποθηκευμένη στη μνήμη της. Εκεί, ανάλογα με τις πληροφορίες που συλλέγει, επιλέγει μία γειτονική πηγή τροφής και τη συγκρίνει με αυτή που βρίσκεται στη μνήμη της. Στο επόμενο στάδιο, μία παρατηρήτρια μέλισσα επιλέγει μία περιοχή τροφής με κριτήριο την πληροφορία για την ποσότητα του νέκταρ που έχει λάβει από τις εργάτριες μέλισσες στην περιοχή χορού. Όσο μεγαλύτερη είναι η ποσότητα του νέκταρ σε μια πηγή

τροφής, τόσο μεγαλύτερη είναι η πιθανότητα προτίμησής της από μία παρατηρήτρια. Συνεπώς ο χορός των μελισσών εργατριών οδηγεί τις παρατηρήτριες σε περιοχές με μεγαλύτερες ποσότητες νέκταρ. Μόλις η παρατηρήτρια μέλισσα φτάσει στην περιοχή, επιλέγει μία νέα πηγή τροφής, γειτονική της πηγής τροφής για την οποία έχει πληροφορηθεί από την εργάτρια. Η επιλογή της γειτονικής πηγής εξαρτάται από την ποσότητα του νέκταρ. Όταν μία συγκεκριμένη πηγή τροφής επιλεγεί πολλές φορές από τις μέλισσες και εξαντλείται το νέκταρ τότε η πηγή εγκαταλείπεται και καθορίζεται τυχαία από την ανιχνεύτρια μέλισσα μία νέα πηγή τροφής η οποία αντικαθιστά την εγκαταλελειμμένη.



Εικόνα 2.11. Τυπικό μοντέλο αποικίας μελισσών

Στον αλγόριθμο ABC, η θέση μιας πηγής τροφής αντιπροσωπεύει μία πιθανή λύση του προβλήματος βελτιστοποίησης και η ποσότητα του νέκταρ στην πηγή τροφής αντιστοιχεί στην ποιότητα (καταλληλότητα) της συγκεκριμένης λύσης. Ο αριθμός των εργατριών (ή των παρατηρητριών) μελισσών είναι ίσος με τον αριθμό των λύσεων στον πληθυσμό. Στο βήμα αρχικοποίησης, ο αλγόριθμος ABC παράγει έναν τυχαία κατανεμημένο αρχικό πληθυσμό SN λύσεων (θέσεων που βρίσκονται οι πηγές τροφής) άρα ο SN υποδηλώνει το μέγεθος του πληθυσμού. Κάθε λύση (τοποθεσία τροφής) x_i ($i = 1, 2, \dots, SN$) είναι ένα διάνυσμα D διαστάσεων. Μετά την αρχικοποίηση του πληθυσμού, σε κάθε κύκλο C του αλγορίθμου ($C = 1, 2, \dots, MCN$) γίνονται νέες αναζητήσεις λύσεων από τις εργάτριες, τις παρατηρήτριες και τις ανιχνεύτριες μέλισσες. Στην περίπτωση των πραγματικών μελισσών, η εύρεση των νέων τοποθεσιών τροφής γίνεται με μία διαδικασία σύγκρισης των πηγών τροφής που βασίζεται σε πληροφορίες που συλλέγουν οι μέλισσες με τις αισθήσεις τους (τέτοιες πληροφορίες μπορεί να είναι το χρώμα, το σχήμα ή το άρωμα των λουλουδιών). Στο τεχνητό μοντέλο, η παραγωγή νέων πηγών τροφής βασίζεται επίσης σε σύγκριση των τοποθεσιών που βρίσκεται η τροφή. Μία τεχνητή εργάτρια

μέλισσα παράγει μία νέα τοποθεσία (λύση) μεταβάλλοντας την τοποθεσία που ήδη υπάρχει στη μνήμη της, σύμφωνα με την εξίσωση:

$$v_{ij} = x_{ij} + \varphi_{ij} \cdot (x_{ij} - x_{kj}) \quad (2.41)$$

όπου $k \in \{1, 2, \dots, SN\}$ και $j \in \{1, 2, \dots, D\}$ τυχαία επιλεγμένοι δείκτες με $k \neq i$. φ_{ij} είναι ένας τυχαίος αριθμός που ανήκει στο διάστημα $[-1, 1]$ και ελέγχει την παραγωγή τοποθεσιών τροφής γύρω από την τοποθεσία x_i . Η τροποποίηση της τοποθεσίας x_i αντιστοιχεί στη οπτική σύγκριση των πηγών τροφής από τις πραγματικές μέλισσες.

Με αυτόν τον τρόπο, χρησιμοποιώντας την τοπική πληροφορία (αντί της πληροφορίας που βασίζεται στις αισθήσεις) βρίσκει μία νέα πηγή τροφής (λύση) στην οποία υπολογίζει την ποσότητα του νέκταρ που αντιστοιχεί στην καταλληλότητα μιας λύσης, ως εξής (Kiran και Gündüz, 2012):

$$fit_i = \begin{cases} \frac{1}{1 + f(x_i)} & \text{αν } f(x_i) \geq 0 \\ 1 + |f(x_i)| & \text{αν } f(x_i) < 0 \end{cases} \quad (2.42)$$

Στην παραπάνω εξίσωση, fit είναι η συνάρτηση καταλληλότητας και f είναι η αντικειμενική συνάρτηση του προβλήματος. Αν η ποσότητα του νέκταρ (συνάρτηση καταλληλότητας) στη νέα τοποθεσία είναι μεγαλύτερη, η μέλισσα απομακρύνει από τη μνήμη της την παλιά τοποθεσία και στη θέση της τοποθετεί την καινούρια. Διαφορετικά, η παλιά τοποθεσία παραμένει. Επομένως εφαρμόζεται ένας μηχανισμός άπληστης επιλογής (greedy selection) μεταξύ της λύσης που ήδη υπάρχει στη μνήμη της μέλισσας και της νέας υποψήφιας λύσης.

Όταν όλες οι εργάτριες μέλισσες ολοκληρώσουν τη διαδικασία αναζήτησης, μεταφέρουν τις πληροφορίες για τις τοποθεσίες τροφής στις παρατηρήτριες οι οποίες στη συνέχεια επιλέγουν μία νέα λύση με μία πιθανότητα που εξαρτάται από την ποσότητα του νέκταρ και υπολογίζεται ως εξής (Karaboga και Basturk, 2007):

$$p_i = \frac{fit_i}{\sum_{j=1}^{SN} fit_j} \quad (2.43)$$

Όπως μία εργάτρια μέλισσα, έτσι και μία παρατηρήτρια επιλέγει νέα τοποθεσία τροφής με την εξίσωση (2.41) και αξιολογεί την ποσότητα του νέκταρ με την εξίσωση (2.42). Αν η νέα τοποθεσία τροφής έχει ίσο ή περισσότερο νέκταρ αποθηκεύεται στη μνήμη της μέλισσας στη θέση της παλιάς τοποθεσίας, διαφορετικά η παλιά τοποθεσία παραμένει.

Μία τοποθεσία (λύση) εγκαταλείπεται όταν δεν μπορεί να βελτιωθεί περαιτέρω μετά από έναν προκαθορισμένο αριθμό κύκλων. Η τιμή των προκαθορισμένων κύκλων ονομάζεται όριο εγκατάλειψης (limit for abandonment) (Karaboga και Ozturk, 2011). Αν υπάρξει μια εγκαταλελειμμένη πηγή τροφής και $j \in \{1, 2, \dots, D\}$, η ανιχνεύτρια μέλισσα την αντικαθιστά με μία νέα πηγή τροφής x_{new} της οποίας οι διαστάσεις υπολογίζονται τυχαία σύμφωνα με την εξίσωση:

$$x_j = x_j^{\min} + rand \cdot (x_j^{\max} - x_j^{\min}) \quad (2.44)$$

όπου $rand$ ένας τυχαίος αριθμός στο διάστημα $(0, 1)$. Στη βασική μορφή του αλγορίθμου, δεν υπάρχει πάνω από μία ανιχνεύτρια μέλισσα σε κάθε κύκλο. Αν προκύψουν παραπάνω από μία

εγκαταλελειμμένες λύσεις αντικαθίσταται αυτή που έχει παραμείνει για περισσότερους κύκλους στον πληθυσμό. Σε άλλες μορφές του αλγορίθμου μπορούν να αντικατασταθούν παραπάνω από μία ή και όλες οι λύσεις που ξεπερνάνε το όριο εγκατάλειψης.

Ακολουθεί ο Ψευδοκώδικας του αλγορίθμου.

Ψευδοκώδικας 2.7. Artificial Bee Colony

Objective function $f(x)$, $x = (x_1, x_2, \dots, x_D)^T$

Define number of population (food sources) SN

Define maximum number of cycles MCN

Define limit of abandonment

Evaluate the population

$cycle \leftarrow 1$

while ($cycle \leq MCN$)

for ($i \leftarrow 1$ to SN)

 Produce new solution v_i for employed bee and evaluate it

 Apply the greedy selection process

end for

for ($i \leftarrow 1$ to SN)

 Calculate the probability value p_i for the solution v_i

if ($rand \leq p_i$)

 Produce new solution v_i for onlooker bee and evaluate it

 Apply the greedy selection process

end if

end for

Determine the abandoned solution for the scout, if exists and replace it with a new randomly produced solution

Memorize the best solution achieved so far

$cycle \leftarrow cycle + 1$

end while

Στον αλγόριθμο ABC η εκμετάλλευση του χώρου αναζήτησης ελέγχεται από τις εργάτριες και τις παρατηρήτριες μέλισσες οι οποίες εκτελούν τοπικές αναζητήσεις γύρω από τις λύσεις που βρίσκονται στη μνήμη τους. Η εξερεύνηση ελέγχεται από τις ανιχνεύτριες μέλισσες που κάνουν τυχαίες αναζητήσεις για την εύρεση νέων λύσεων. Οι παράμετροι ελέγχου του αλγορίθμου είναι ο αριθμός των πηγών τροφής που είναι ίσος με τον αριθμό των εργατριών ή των παρατηρητριών μελισσών (SN), ο μέγιστος αριθμός κύκλων (MCN) και η τιμή του ορίου εγκατάλειψης.

2.8. Άλλοι μεταερευνητικοί αλγόριθμοι

Εκτός των αλγορίθμων που περιγράψαμε, έχει αναπτυχθεί ένα μεγάλο πλήθος μεταερευνητικών μεθόδων οι οποίες έχουν εφαρμοστεί με επιτυχία σε προβλήματα βελτιστοποίησης. Σημαντικοί αλγόριθμοι που είναι εμπνευσμένοι από τη φύση και τη βιολογία είναι και οι παρακάτω.

2.8.1. Εξελικτικές στρατηγικές

Οι εξελικτικές στρατηγικές (Evolutionary Strategies – ES) αναπτύχθηκαν για πρώτη φορά το 1964 από τους Bienenert, Rechenberg, Schwefel (Biniitha και Siva Sathya, 2012). Είναι εμπνευσμένες από την εξέλιξη των ειδών και συγκεκριμένα στο επίπεδο της μακροεξέλιξης (φαινότυπος, κληρονομικότητα) αντίθετα από τους γενετικούς αλγορίθμους που μοντελοποιούν το επίπεδο της μικροεξέλιξης (γονίδιο, γονιδίωμα, χρωμόσωμα). Ένα πολύ σημαντικό χαρακτηριστικό τους είναι η χρήση αυτοπροσαρμοστικών μηχανισμών για τον έλεγχο της λειτουργίας της μετάλλαξης. Τέτοιοι μηχανισμοί προσπαθούν να βελτιώσουν όχι μόνο τις λύσεις αλλά και τις παραμέτρους για τη μετάλλαξη αυτών των λύσεων. Πολύ συνηθισμένα σχήματα εξελικτικών στρατηγικών είναι τα ακόλουθα (Biniitha και Siva Sathya, 2012):

- $(1+1)$ – ES: Πρόκειται για έναν απλό μηχανισμό επιλογής στον οποίο κάθε διάνυσμα γονέας παράγει ένα νέο διάνυσμα μεταβλητών με την πρόσθεση μιας ομαλά κατανομημένης τυχαίας τιμής σε κάθε μεταβλητή του. Το νέο διάνυσμα συγκρίνεται με το γονέα του και το καλύτερο από τα δύο διανύσματα γίνεται γονέας της επόμενης γενιάς ενώ το άλλο εγκαταλείπεται.
- $(\mu + \lambda)$ – ES: Επιλέγονται μ γονείς από μία γενιά και παράγουν λ απογόνους με τη χρήση τελεστών αναδιάταξης και μετάλλαξης. Από την ένωση των $(\mu + \lambda)$ γονέων και απογόνων διατηρούνται τα μ καλύτερα άτομα για την επόμενη γενιά. Η συγκεκριμένη τεχνική περιλαμβάνει ελιτισμό.
- (μ, λ) – ES: Επιλέγονται μ γονείς από μία γενιά και παράγουν λ απογόνους με $\lambda \geq \mu$. Οι μ καλύτεροι απόγονοι δημιουργούν την νέα γενιά και οι γονείς εγκαταλείπονται. Εδώ, δεν παρουσιάζεται ελιτισμός.

2.8.2. Γενετικός προγραμματισμός

Ο γενετικός προγραμματισμός (Genetic Programming – GP) (Koza, 1993) διαφέρει από τους γενετικούς αλγορίθμους στον τρόπο αναπαράστασης των λύσεων. Ο GP συνήθως χρησιμοποιεί ένα δένδρο για να κωδικοποιήσει μία υποψήφια λύση η οποία μπορεί να είναι ένα πρόγραμμα υπολογιστή. Η δεύτερη θεμελιώδης διαφορά είναι το μεταβλητό μήκος της αναπαράστασης της λύσης στον GP σε αντίθεση με τους GA όπου το μήκος της κωδικοποιημένης λύσης είναι σταθερό. Τα τρία βασικά βήματα του GP για την παραγωγή ενός προγράμματος υπολογιστή είναι τα εξής (Biniitha και Siva Sathya, 2012):

1. Παραγωγή ενός αρχικού πληθυσμού προγραμμάτων που περιλαμβάνουν τερματικούς όρους (terminals) και συναρτήσεις.
2. Εκτέλεση κάθε προγράμματος του πληθυσμού και ανάθεση μιας τιμής καταλληλότητας σύμφωνα με την ικανότητα που εμφανίζει για την επίλυση του προβλήματος
3. Δημιουργία νέου πληθυσμού προγραμμάτων:
 - a) Αντιγραφή των καλύτερων προγραμμάτων
 - b) Δημιουργία νέων προγραμμάτων με τη λειτουργία της μετάλλαξης
 - c) Δημιουργία νέων προγραμμάτων με τη λειτουργία της διασταύρωσης

2.8.3. Βελτιστοποίηση αποικίας μυρμηγκιών

Η βελτιστοποίηση αποικίας μυρμηγκιών (Ant Colony Optimization – ACO) είναι εμπνευσμένη από τον τρόπο αναζήτησης και συλλογής τροφής των μυρμηγκιών και συγκεκριμένα από τον φαινόμενο της στιγμεργίας. Ο όρος *στιγμεργία* (stigmery) εισήχθη από τον Grasse το 1959 (Biniitha και Siva Sathya, 2012) και προέρχεται από τις λέξεις στίγμα και έργο. Αναφέρεται στην έμμεση εικονομία σε ένα αυτο-οργανωμένο σύστημα η οποία πραγματοποιείται με φυσικές μεταβολές του περιβάλλοντος από τα άτομα του συστήματος. Πολλά είδη μυρμηγκιών έχουν την ικανότητα να ανακαλύπτουν τη συντομότερη διαδρομή μεταξύ της φωλιάς τους και της πηγής τροφής ακολουθώντας μονοπάτια που έχουν δημιουργηθεί από άλλα μυρμηγκία με την απελευθέρωση μιας ορμόνης που ονομάζεται

φερομόνη. Ένα μονοπάτι επιλέγεται με κάποια πιθανότητα ανάλογη με την ποσότητα φερομόνης δηλαδή όσο περισσότερη φερομόνη περιέχει ένα μονοπάτι τόσο πιθανότερο είναι να επιλεγεί. Ένα μυρμήγκι που εντοπίζει ένα ίχνος φερομόνης και αποφασίζει να το ακολουθήσει, εναποθέτει και τη δική του φερομόνη κάνοντας τη συγκεκριμένη διαδρομή πιο ελκυστική για τα επόμενα μυρμήγκια. Το συγκεκριμένο χαρακτηριστικό συμπεριφοράς ονομάζεται ενίσχυση (reinforcement). Τα χαρακτηριστικά που περιγράψαμε μοντελοποιούνται για την εύρεση λύσεων σε προβλήματα βελτιστοποίησης.

Ο πρώτος αλγόριθμος μυρμηγκιών ήταν ο Ant System (AS) που παρουσιάστηκε από τους Dorigo, Maniezzo και Coloni (1996) και εφαρμόστηκε με επιτυχία στο πρόβλημα του περιοδεύοντος πωλητή. Οι Dorigo and Gambardella (1997), με τον αλγόριθμο Ant Colony System (ACS) βελτίωσαν σημαντικά την απόδοση του AS. Ο ACO αναπτύχθηκε από τους Dorigo, Di Caro και Gambardella (1999) για να γενικεύσει τις μεθόδους επίλυσης προβλημάτων συνδυαστικής βελτιστοποίησης που βασίζονται στη συμπεριφορά των πραγματικών μυρμηγκιών (Binitha και Siva Sathya, 2012). Τρεις βασικές συνάρτησεις συγκροτούν τον ACO (Binitha και Siva Sathya, 2012):

- AntSolutionConstruct: Εκτελεί τη διαδικασία κατασκευής λύσεων όπου τα τεχνητά μυρμήγκια μετακινούνται στις επόμενες καταστάσεις σύμφωνα με έναν κανόνα μετάβασης.
- PheromoneUpdate: Εκτελεί την ανανέωση των μονοπατιών φερομόνης. Η φερομόνη μπορεί να ανανεώνεται μετά από την κατασκευή μιας λύσης ή μετά από κάθε επανάληψη του αλγορίθμου. Εκτός από ενίσχυση, ο ACO περιλαμβάνει και εξάτμιση μονοπατιών φερομόνης ώστε τα μυρμήγκια να αποφεύγουν τις κακές λύσεις που βρήκαν νωρίτερα.
- DemonActions: Προαιρετικό βήμα του αλγορίθμου που περιλαμβάνει επιπρόσθετες ανανεώσεις όπως για παράδειγμα επιπλέον ενίσχυση φερομόνης για την καλύτερη λύση που έχει βρεθεί.

2.8.4. Βελτιστοποίηση αναζήτησης τροφής των βακτηρίων

Η αναζήτηση τροφής (foraging) σε μία αποικία βακτηρίων είναι ένα σύνθετο φαινόμενο με χαρακτηριστικά όπως η χημειοταξία (μετακίνηση κυττάρων), ο μεταβολισμός και η διακυτταρική επικοινωνία (quorum sensing) που έχουν μοντελοποιηθεί για την επίλυση πολύπλοκων προβλημάτων βελτιστοποίησης. Ο πιο γνωστός αλγόριθμος που έχει εμπνευστεί από αυτό το φαινόμενο είναι η βελτιστοποίηση αναζήτησης τροφής των βακτηρίων (Bacteria Foraging Optimization – BFO) (Passino, 2002).

Οι βασικότεροι μηχανισμοί του αλγορίθμου BFO είναι οι εξής (Binitha και Siva Sathya, 2012):

- Χημειοταξία ή χημειοτακτισμός (chemotaxis): Είναι η κίνηση και συγκέντρωση των βακτηρίων σε περιοχές πλούσιες σε θρεπτικά συστατικά (π.χ. αμινοξέα). Ένα βακτήριο μπορεί να κινείται προς μία συγκεκριμένη κατεύθυνση (swimming) ή να αλλάζει προσανατολισμό (tumbling) καθώς αναζητά την τροφή του (Thomas, 2013).
- Αναπαραγωγή: Μετά από έναν αριθμό χημειοτακτικών βημάτων, το 1/2 του πληθυσμού με τα καλύτερα άτομα υποβάλλεται σε αναπαραγωγή ενώ τα υπόλοιπα βακτήρια αποβάλλονται (Thomas, 2013).
- Γεγονότα αποβολής / διασποράς: Για την απόδραση από τοπικά βέλτιστα, τυχαία βακτήρια του πληθυσμού αποβάλλονται με πολύ μικρή πιθανότητα. Στη συνέχεια αντικαθίστανται από νέα βακτήρια τα οποία διασκορπίζονται τυχαία στο περιβάλλον (χώρο αναζήτησης).

2.8.5. Αλγόριθμος σμήνους ψαριών

Ο αλγόριθμος σμήνους ψαριών (Fish Swarm Algorithm – FSA) παρουσιάστηκε από τους Li, Shao και Qian (2002) και έχει εμπνευστεί από την κοινωνική συμπεριφορά των ψαριών τα οποία είναι οργανωμένα σε κοπάδια και κάθε κοπάδι κολυμπά συντονισμένα προς την ίδια κατεύθυνση (fish schooling) (Mahale και Chavan, 2012). Ένα ψάρι χαρακτηρίζεται από μια θέση x_i με d διαστάσεις. Ο βαθμός ικανοποίησης ενός ψαριού από την τροφή του συμβολίζεται με FS_i . Η σχέση μεταξύ δύο

ψαριών x_i και x_j υποδηλώνεται με την ευκλίδεια απόστασή τους $d_{ij} = \|x_i - x_j\|$. Ο FSA μοντελοποιεί τρία βασικά χαρακτηριστικά της συμπεριφοράς των ψαριών (Mahale και Chavan, 2012):

- Αναζήτηση (searching): Το ψάρι αναζητά την τροφή του κάνοντας τυχαίες κινήσεις ώστε να βρεθεί σε περιοχή με μεγάλη ποσότητα τροφής.
- Κίνηση κατά σμήνη (swarming): Ένα ψάρι που βρίσκεται στη θέση x_i έχει γειτονικά ψάρια και x_c είναι το κέντρο της γειτονιάς. Αν η θέση x_c έχει μεγαλύτερη συγκέντρωση τροφής από την x_i , το ψάρι θα κινηθεί προς το κέντρο της γειτονιάς του.
- Ακολουθία (following): Όταν ένα ψάρι βρει τροφή, τα γειτονικά του ψάρια θα το ακολουθήσουν. Κάθε ψάρι, προκειμένου να ικανοποιήσει την ανάγκη του για τροφή στο μέγιστο βαθμό, προσπαθεί να ακολουθήσει το ψάρι το οποίο βρίσκεται στο οπτικό του πεδίο και βρίσκει τη μεγαλύτερη ποσότητα τροφής (x_{min}). Πρέπει όμως να αποφεύγεται ο συνωστισμός των ψαριών σε μια τοποθεσία.

Οι τρεις βασικοί παράμετροι του FSA είναι η ακτίνα του οπτικού πεδίου (visual distance), το μέγιστο μήκος του βήματος και ένας παράγοντας συνωστισμού. Η αποτελεσματικότητα του FSA επηρεάζεται περισσότερο από τις δύο πρώτες παραμέτρους.

2.8.6. Αλγόριθμος της πυγολαμπίδας

Αναπτύχθηκε από τον Yang το 2007 (Yang, 2010a) και είναι εμπνευσμένος από τη συμπεριφορά των πυγολαμπίδων όταν εκπέμπουν ή όταν έλκονται από λάμπες. Οι πυγολαμπίδες χρησιμοποιούν τη λάμψη ως μέσο επικοινωνίας για να προσελκύσουν άλλες πυγολαμπίδες. Για λόγους απλότητας ο αλγόριθμος της νυχτερίδας (Firefly Algorithm – FA) ακολουθεί τρεις κανόνες (Sureja, 2012) (Yang, 2010a):

- Οι πυγολαμπίδες είναι μονογενείς. Μια πυγολαμπίδα μπορεί να προσελκύσει οποιαδήποτε άλλη χωρίς να εξετάζεται το φύλο τους.
- Η ελκυστικότητα των πυγολαμπίδων είναι ανάλογη με τη φωτεινότητά τους. Η φωτεινότητα μιας πυγολαμπίδας μειώνεται καθώς αυξάνεται η απόστασή της από άλλες πυγολαμπίδες. Μία πυγολαμπίδα θα κινηθεί προς μία άλλη, φωτεινότερη. Αν δεν υπάρχει φωτεινότερη, η πυγολαμπίδα θα κινηθεί προς τυχαία κατεύθυνση.
- Μία πυγολαμπίδα δεν μπορεί σε καμία περίπτωση να προσελκύσει μια άλλη που είναι φωτεινότερη.

Η φωτεινότητα μιας πυγολαμπίδας σχετίζεται με την αντικειμενική συνάρτηση του προβλήματος. Σε ένα πρόβλημα μεγιστοποίησης η φωτεινότητα είναι ανάλογη με την τιμή της αντικειμενικής συνάρτησης. Μπορούν να καθοριστούν και άλλες μορφές φωτεινότητας παρόμοιες με τη συνάρτηση καταλληλότητας των γενετικών αλγορίθμων. Για τον υπολογισμό της απόστασης μεταξύ των πυγολαμπίδων μπορεί να χρησιμοποιηθεί η καρτεσιανή απόσταση. Επίσης διαμορφώνεται μία συνάρτηση ελκυστικότητας (attractiveness function) από τη φωτεινότητα, την απόσταση κι έναν παράγοντα απορρόφησης της λάμψης. Η ελκυστικότητα, η τρέχουσα θέση και μία τυχαία διαδρομή συνθέτουν τη συνάρτηση κίνησης (movement function) για μία πυγολαμπίδα (Sureja, 2012).

2.8.7. Αναζήτηση του κούκου

Ο αλγόριθμος της αναζήτησης του κούκου (Cuckoo Search – CS) (Yang και Deb, 2009) είναι εμπνευσμένος από την παρασιτική συμπεριφορά που εμφανίζεται στη διαδικασία αναπαραγωγής ορισμένων ειδών κούκου τα οποία τοποθετούν τα αυγά τους σε φωλιές άλλων πουλιών. Τα αυγά κάποιων ειδών κούκου μοιάζουν με τα αυγά άλλων ειδών πουλιών που χρησιμοποιούνται ως

ξενιστές. Αυτή η ιδιότητα μειώνει την πιθανότητα της απομάκρυνσης των αβγών του κούκου από το πουλί - ξενιστή. Ενίοτε υπάρχει σύγκρουση μεταξύ του ξενιστή και του κούκου και ο ξενιστής είτε απομακρύνει τα αβγά του κούκου, είτε καταστρέφει τη φωλιά του και χτίζει μία νέα σε άλλο μέρος. Συνήθως ο κούκος επιλέγει μία φωλιά στην οποία ο ξενιστής έχει μόλις τοποθετήσει τα αβγά του. Τα αβγά του κούκου συνηθίζουν να εκκολάπτονται νωρίτερα από αυτά του ξενιστή. Όταν ο πρώτος νεοσσός κούκος εκκολαφθεί, απωθεί ενστικτωδώς τα αβγά του ξενιστή από τη φωλιά. Με αυτό τον τρόπο ο κούκος καταφέρνει να λάβει μεγαλύτερη ποσότητα τροφής από τον ξενιστή.

Το αβγό σε μια φωλιά αναπαριστά μία λύση, ενώ το αβγό του κούκου αναπαριστά μία νέα λύση. Για να επιτευχθούν καλύτερες λύσεις, τα αβγά που βρίσκονται στις φωλιές αντικαθίσταται με καλύτερα αβγά που προέρχονται από τους κούκους. Ο CS βασίζεται σε τρεις κανόνες (Yang και Deb, 2010), (Sureja, 2012):

- Κάθε κούκος τοποθετεί ένα αβγό κάθε φορά σε μία τυχαία φωλιά.
- Οι φωλιές με τα καλύτερα αβγά (λύσεις) μεταφέρονται στην επόμενη γενιά.
- Ο αριθμός των φωλιών είναι προκαθορισμένος και ο ξενιστής μπορεί να ανακαλύψει ένα ξένο αβγό με μια πιθανότητα p_a . Σε αυτή την περίπτωση το πουλί ξενιστής μπορεί είτε να απομακρύνει το αβγό είτε να χτίσει μία νέα φωλιά σε άλλη θέση. Για απλότητα, ένα κλάσμα p_a επί του συνόλου των φωλιών αντικαθίσταται με νέες φωλιές που περιλαμβάνουν νέες λύσεις.

Σε ένα πρόβλημα μεγιστοποίησης η καταλληλότητα μιας λύσης είναι ανάλογη με την τιμή της αντικειμενικής συνάρτησης. Μπορούν να καθοριστούν και άλλες μορφές καταλληλότητας παρόμοιες με τη συνάρτηση καταλληλότητας των γενετικών αλγορίθμων. Ένα ακόμη αξιοσημείωτο χαρακτηριστικό είναι ότι ο CS αντί της απλής τυχαίας διαδρομής χρησιμοποιεί πτήσεις Lévy.

2.8.8. Αναζήτηση βαρύτητας

Ο αλγόριθμος της αναζήτησης βαρύτητας (Gravitational Search Algorithm – GSA) (Rashedi, Nezamabadi-Pour και Saryazdi, 2009) είναι ένας πληθυσμιακός αλγόριθμος, βασισμένος στο νόμο της βαρύτητας και στις αλληλεπιδράσεις των μαζών. Η ελκτική δύναμη μεταξύ δύο σωμάτων είναι ευθέως ανάλογη του γινομένου της μάζας τους και αντιστρόφως ανάλογη του τετραγώνου της απόστασής τους (Rashedi, Nezamabadi-Pour και Saryazdi, 2009).

Στον GSA οι πράκτορες αναζήτησης είναι μία συλλογή αντικειμένων που αλληλεπιδρούν μεταξύ τους με βάση τη νευτώνεια βαρύτητα και τους νόμους της κίνησης. Η απόδοση κάθε αντικειμένου καθορίζεται από τη μάζα του. Όλα τα αντικείμενα έλκονται μεταξύ τους λόγω της ελκτικής δύναμης. Πιο συγκεκριμένα, τα αντικείμενα κινούνται προς άλλα αντικείμενα με μεγαλύτερες μάζες. Οι μεγάλες μάζες (δηλαδή βαρύτερα αντικείμενα) αντιστοιχούν σε καλές λύσεις ενός προβλήματος. Η θέση ενός αντικειμένου αντιστοιχεί σε μία λύση και η μάζα του καθορίζεται από τη συνάρτηση καταλληλότητας. Μετά το πέρας του αλγορίθμου, η βέλτιστη λύση αντιστοιχεί στη θέση του αντικειμένου με τη μεγαλύτερη μάζα δηλαδή του αντικειμένου που έλκει όλα τα υπόλοιπα.

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

3. Ο ΑΛΓΟΡΙΘΜΟΣ ΤΗΣ ΝΥΧΤΕΡΙΔΑΣ

Ο αλγόριθμος της νυχτερίδας (Bat Algorithm – BA) είναι ένας από τους πιο πρόσφατους μεταερευτικούς αλγόριθμους. Αναπτύχθηκε από τον Xin-She Yang (2010b) και είναι επηρεασμένος από το σύστημα ηχοεντοπισμού που χρησιμοποιούν οι νυχτερίδες κατά την αναζήτηση της τροφής τους. Έχει αναδειχθεί σε έναν ιδιαίτερα αποδοτικό πληθυσμιακό αλγόριθμο νοημοσύνης σμήνους με την απόδοσή του να μπορεί να αυξάνεται παιρετέρω μετά από κατάλληλη τροποποίηση ορισμένων παραμέτρων ή ένταξη στοιχείων από άλλους μεταερευτικούς αλγόριθμους.

3.1. Ο ηχοεντοπισμός των νυχτερίδων

3.1.1. Γενικά στοιχεία για τις νυχτερίδες

Οι νυχτερίδες υπολογίζεται ότι εμφανίστηκαν στη γη πριν από 65 με 100 εκατομμύρια χρόνια (Airas, 2003). Συγκροτούν την τάξη των χειρόπτερων (Chiroptera) που είναι η δεύτερη μεγαλύτερη σε αριθμό ειδών τάξη των θηλαστικών μετά τα τρωκτικά (Σοφινίδου, 2000). Έχουν καταγραφεί περισσότερα από 1200 διαφορετικά είδη τα όποια αποτελούν το 1/5 του συνόλου των θηλαστικών του πλανήτη. Οι νυχτερίδες είναι τα μόνα θηλαστικά που πετούν πραγματικά, ανεμίζοντας τα πολύ μακριά δάχτυλα που βρίσκονται στα άνω άκρα τους και είναι καλυμμένα με μία λεπτή μεμβράνη. Αναφορικά με το μέγεθός τους, παρουσιάζουν μεγάλη ποικιλία ξεκινώντας από τη νυχτερίδα μέλισσα (bumblebee bat) που είναι και το μικρότερο θηλαστικό του πλανήτη με βάρος μόλις 1.5 με 2 γραμμάρια μέχρι τις γιγάντιες ιπτάμενες αλεπούδες (giant flying foxes) που το βάρος τους μπορεί να ξεπεράσει το 1 χιλιόγραμμα και το άνοιγμα των πτερύγων τους μπορεί να πλησιάσει τα 2 μέτρα (Batcon.org, 2011) (Yang, 2010b). Σχεδόν τα 2/3 των νυχτερίδων είναι εντομοφάγες και το υπόλοιπο 1/3 τρέφεται με φρούτα, γύρη και νέκταρ από τα λουλούδια ενώ περίπου το 1% τρέφεται με άλλα ζώα όπως όπως ψάρια και ποντίκια (Batcon.org, 2011).

Υπάρχουν δύο υποτάξεις στις οποίες διαιρούνται οι νυχτερίδες:

- Megachiroptera (megabats)
- Microchiroptera (microbats)

Οι όροι είναι κάπως παραπλανητικοί καθώς ορισμένα μεγαχειρόπτερα είναι μικρότερα από κάποια μικροχειρόπτερα (Σοφινίδου, 2000). Στην πρώτη υποτάξη ανήκουν οι φρουτοφάγες νυχτερίδες. Η δεύτερη υποτάξη περιλαμβάνει τις νυχτερίδες οι οποίες προσανατολίζονται με τη χρήση του ηχοεντοπισμού, μία λειτουργία που θα περιγραφεί αναλυτικά. Όλες οι νυχτερίδες βλέπουν πολύ καλά αλλά η όραση χρησιμοποιείται για προσανατολισμό μόνο από τα μεγαχειρόπτερα (Σοφινίδου, 2000).

Πολλά είδη νυχτερίδων είναι ιδιαίτερα κοινωνικά και φωλιάζουν ακόμα και κατά εκατομμύρια. Φωλιές δημιουργούν συνήθως σε σκοτεινά μέρη όπως σπηλιές, ορυχεία, σχισμές βράχων και κοιλώματα δένδρων. Σε μία φωλιά μπορούν να συνυπάρχουν νυχτερίδες από διάφορα είδη. Ένα άλλο εντυπωσιακό χαρακτηριστικό είναι η μακροβιότητά τους. Κάποια είδη νυχτερίδας ζουν μέχρι και 30 χρόνια. Έχουν όμως χαμηλό ρυθμό αναπαραγωγής αφού γεννούν 1-3 μωρά το χρόνο (Ελληνική Σπηλαιολογική Εταιρεία).

Παρά τα εκπληκτικά τους χαρακτηριστικά και τον πολύ σημαντικό ρόλο τους στη διατήρηση της ισορροπίας στη φύση, οι νυχτερίδες είναι τα πιο παρεξηγημένα θηλαστικά του πλανήτη. Στις δυτικές κοινωνίες είναι συνδεδεμένες με τον Κόμη Δράκουλα και τους βρικόλακες, φανταστικά πλάσματα της λογοτεχνίας και του κινηματογράφου που τρέφονται με ανθρώπινο αίμα. Η αλήθεια είναι ότι υπάρχουν μόλις 3 είδη νυχτερίδων που τρέφονται με αίμα και ονομάστηκαν «βαμπίροι» (vampire bats) από το μυθιστόρημα «Δράκουλας» του Bram Stoker. Μόνο ένα είδος βαμπίρου είναι επικίνδυνο για τον άνθρωπο γιατί είναι φορέας του ιού της λύσσας. Όλα τα υπόλοιπα είδη νυχτερίδας είναι τελείως ακίνδυνα και πολλές φορές ευεργετικά γιατί περιορίζουν τα βλαβερά

έντομα και μεταφέρουν σπόρους βοηθώντας στην αναγέννηση των δασών (Ελληνική Σπηλαιολογική Εταιρεία).

3.1.2. Το φαινόμενο του ηχοεντοπισμού

Ένα από τα αξιοσημείωτα χαρακτηριστικά των νυχτερίδων είναι ο τρόπος προσανατολισμού τους που γίνεται με ένα φαινόμενο γνωστό ως ηχοεντοπισμός. Η διαδικασία του ηχοεντοπισμού παρατηρείται και σε άλλα ζώα (Airas, 2003). Τα κήτη (φάλαινες, δελφίνια) μαζί με τις νυχτερίδες είναι τα θηλαστικά με τις πιο προηγμένες ικανότητες ηχοεντοπισμού αλλά υπάρχουν και ορισμένοι μικροί μυρμηγκοφάγοι που διαθέτουν υποτυπώδεις ικανότητες υπερηχητικού ηχοεντοπισμού. Επίσης αρκετά σπηλαιόβια πτηνά χρησιμοποιούν ήχους χαμηλών συχνοτήτων ώστε να εντοπίζουν τις φωλιές τους μέσα σε σκοτεινές σπηλιές. Στις νυχτερίδες, ο ηχοεντοπισμός παρουσιάζεται σε όλες τις μικρές νυχτερίδες (microbats) ενώ από τις μεγάλες νυχτερίδες (megabats) μόνο σε ένα είδος, τις αιγυπτιακές φρουτοφάγες (Egyptian fruit bats).

Οι μικρές νυχτερίδες χρησιμοποιούν ένα είδος σόναρ (ηχητικού εντοπιστή) για να εντοπίζουν τη λεία τους, να αποφεύγουν εμπόδια και να εντοπίζουν στο σκοτάδι τις χαραμάδες σπηλιών ή δένδρων που χρησιμοποιούν ως φωλιές. Οι συγκεκριμένες νυχτερίδες καθώς πετούν, εκπέμπουν από το στόμα ή τα ρουθούνια (Σοφιανίδου, 2000) πολύ δυνατούς ηχητικούς παλμούς που παράγονται στο λάρυγγα (Airas, 2003) με σκοπό να ακούσουν τις ηχητικές ανακλάσεις που προέρχονται από τα γύρω αντικείμενα (Εικόνα 3.1). Οι αιγυπτιακές φρουτοφάγες νυχτερίδες παράγουν ήχους κροταλίζοντας τη γλώσσα τους αλλά αυτό το είδος ηχοεντοπισμού δε θα μας απασχολήσει.

Οι μελέτες δείχνουν ότι οι μικρές νυχτερίδες υπολογίζουν τη χρονική καθυστέρηση μεταξύ της εκπομπής του παλμού και της λήψης της ανάκλασής του, τη χρονική διαφορά μεταξύ των ήχων που φτιάνουν στα δύο αφτιά και τη διακυμάνση της έντασης των ανακλάσεων του ήχου ώστε να δημιουργήσουν μία τρισδιάστατη Εικόνα του χώρου που τις περιβάλλει (Yang, 2010b). Μπορούν να εντοπίσουν την απόσταση και την κατεύθυνση του στόχου τους, το είδος του θηράματός τους ακόμα και την ταχύτητα με την οποία κινείται το θήραμά τους. Οι έρευνες επίσης δείχνουν ότι οι νυχτερίδες είναι ικανές να ξεχωρίζουν τους στόχους τους με τη βοήθεια των διακυμάνσεων του φαινομένου Doppler¹ που προκαλείται από τις κινήσεις των φτερών των εντόμων που αποτελούν τους στόχους.

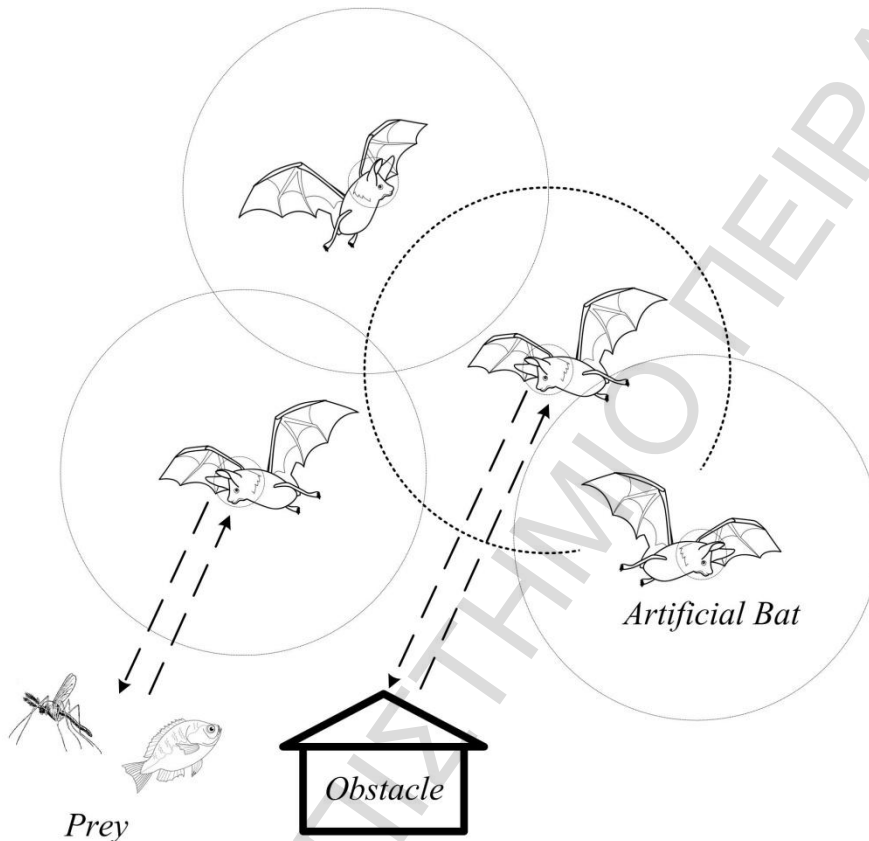
3.1.3. Οι ακουστικές ιδιότητες του ηχοεντοπισμού

Η συχνότητα ενός ηχητικού παλμού που εκπέμπεται από μία νυχτερίδα βρίσκεται συνήθως στην περιοχή μεταξύ 25 kHz και 100 kHz, ενώ ενώ μερικά είδη μπορούν να εκπέμπουν συχνότητες μέχρι και 150 kHz (Airas, 2003) (Yang, 2010b). Οι παλμοί ποικίλλουν, εξαρτώνται από το είδος και σχετίζονται με τη στρατηγική που χρησιμοποιεί μια νυχτερίδα καθώς κυνηγά τη λεία της. Οι περισσότερες οικογένειες νυχτερίδων χρησιμοποιούν σύντομους ήχους διαμορφωμένης συχνότητας (frequency-modulated – FM) που σαρώνονται κατά μία οκτάβα. Το εύρος ζώνης του σήματος διαφέρει ανά είδος και συχνά αυξάνεται με τη χρήση περισσότερων αρμονικών. Πολύ συνηθισμένα είναι και τα σήματα σταθερής συχνότητας (frequency-constant – CM).

Τα περισσότερα είδη εκπέμπουν σύντομους παλμούς που διαρκούν 8 με 10 ms αλλά υπάρχουν και νυχτερίδες με παλμούς που ξεπερνούν σε διάρκεια τα 10 ms και φθάνουν μέχρι 100 ms (Airas, 2003). Κάθε υπερηχητική ριπή, τυπικά μπορεί να διαρκέσει 5 έως 20 ms και οι μικρές νυχτερίδες εκπέμπουν περίπου 10 με 20 τέτοιες ηχητικές ριπές ανά δευτερόλεπτο καθώς αναζητούν τη λεία τους. Όταν πετούν κοντά στο θήραμά τους, ο ρυθμός εκπομπής παλμών

¹ Φαινόμενο Doppler (Dopple effect ή Doppler shift) ονομάζεται το φαινόμενο κατά το οποίο όταν μία πηγή εκπομπής κυμάτων και ένας παρατηρητής βρίσκονται σε σχετική κίνηση μεταξύ τους, το μήκος κύματος και η συχνότητα που αντιλαμβάνεται ο παρατηρητής διαφέρουν από το πραγματικό μήκος κύματος και την πραγματική συχνότητα του κύματος που εκπέμπεται από την πηγή.

αυξάνεται και μπορεί να φθάσει τους 200 παλμούς ανά δευτερόλεπτο και κάθε παλμός να διαρκεί μόλις ένα κλάσμα του ms (Airas, 2003) (Yang, 2010b). Όταν μία νυχτερίδα βρίσκεται μακριά από το στόχο της μπορεί να εκπέμπει παλμούς μεγάλης διάρκειας και να αφήνει επαρκή χρονικά διαστήματα μεταξύ των παλμών. Όσο πλησιάζει προς το στόχο της, οι παλμοί πρέπει να γίνονται συντομότεροι (και περισσότεροι) ώστε να να αποφεύγεται η επικάλυψη μεταξύ εκπεμπόμενου ήχου και ανάκλασης και η νυχτερίδα να φτάνει στο στόχο γρηγορότερα. Οι πολύ σύντομες ηχητικές ριπές δείχνουν την εκπληκτική ικανότητα που έχουν οι νυχτερίδες στην επεξεργασία σημάτων. Στην πραγματικότητα, οι έρευνες δείχνουν ότι ο χρόνος ολοκλήρωσης (άθροισης των σημάτων) του αψιού της νυχτερίδας είναι περίπου 300 με 400 μ s (Yang, 2010b).



Εικόνα 3.1. Απεικόνιση του ηχοεντοπισμού. Οι νυχτερίδες εκπέμπουν ηχητικούς παλμούς και αντιλαμβάνονται την ηχώ τους που προέρχεται από τα θηράματά τους ή εμπόδια που βρίσκονται γύρω τους.

Καθώς η ταχύτητα του ήχου στον αέρα είναι περίπου $v = 340$ m/s σε θερμοκρασία δωματίου, το μήκος κύματος λ των υπερηχητικών ριπών για μία καθορισμένη συχνότητα f δίνεται από την εξίσωση²:

$$\lambda = \frac{v}{f} \quad (3.1)$$

² Στη συνέχεια, η συχνότητα θα συμβολίζεται με Q για να αποφευχθεί σύγχυση με το συμβολισμό της αντικειμενικής συνάρτησης.

Άρα για το τυπικό εύρος συχνοτήτων 25 kHz έως 150 kHz, το μήκος κύματος βρίσκεται μεταξύ 2 mm και 14 mm. Η σειρά των τιμών του μήκους κύματος είναι αντίστοιχη με τη σειρά του μεγέθους των θηραμάτων τους.

Η ένταση του ηχητικού παλμού διαφέρει ανά είδος και μπορεί να έχει τιμή μέχρι και 120 dB (Airas, 2003) ωστόσο ο παλμός δε γίνεται αντιληπτός από το ανθρώπινο αφτί γιατί βρίσκεται στην περιοχή των υπερήχων (ultrasonic region). Η ένταση του ήχου (loudness) είναι πολύ μεγάλη όταν η νυχτερίδα αναζητά την τροφή της και μικρή όταν πλησιάζει το θήραμά της (Yang, 2010b). Τέτοιοι σύντομοι παλμοί συνήθως διανύουν αποστάσεις λίγων μέτρων, οι οποίες εξαρτώνται από τις πραγματικές συχνότητες. Οι μικρές νυχτερίδες καταφέρνουν να αποφεύγουν εμπόδια, τόσο μικρά όσο μία ανθρώπινη τρίχα.

Όλες οι νυχτερίδες έχουν καλή όραση και οι περισσότερες έχουν πολύ ευαίσθητη όσφρηση. Στην πραγματικότητα, συνδυάζουν όλες τις αισθήσεις ώστε να ανιχνεύουν τη λεία τους και να κινούνται ομαλότερα (Yang, 2010b). Όμως, στην παρούσα μελέτη μας ενδιαφέρει μόνο η συμπεριφορά τους κατά τη διαδικασία του ηχοεντοπισμού.

Η λειτουργία του ηχοεντοπισμού των μικρών νυχτερίδων μπορεί να συσχετισθεί με την επίση ενός προβλήματος βελτιστοποίησης, γεγονός που καθιστά δυνατή την ανάπτυξη νέων αλγορίθμων. Μία νέα μεταερευτική τεχνική είναι ο αλγόριθμος της νυχτερίδας που παρουσιάζουμε στην τρέχουσα ενότητα.

3.2. Περιγραφή του αλγορίθμου

Ο αλγόριθμος της νυχτερίδας είναι βασισμένος σε ορισμένα από τα χαρακτηριστικά του ηχοεντοπισμού που περιγράψαμε. Για απλότητα, θεωρούμε τους ακόλουθους κανόνες (Yang, 2010b):

- Όλες οι νυχτερίδες χρησιμοποιούν ηχοεντοπισμό για να εντοπίσουν σημεία και γνωρίζουν τη διαφορά μεταξύ τροφής/θηράματος και εμποδίων.
- Οι νυχτερίδες αναζητούν το θήραμά τους πετώντας τυχαία με ταχύτητα v_i σε μια θέση x_i με μία καθορισμένη ελάχιστη συχνότητα Q_{min} , με μία ένταση ήχου A_0 , και με το μήκος κύματος να ποικίλλει. Μπορούν να προσαρμόσουν αυτόματα το μήκος κύματος (ή τη συχνότητα) των παλμών και να ρυθμίσουν το ρυθμό εκπομπής παλμών $r \in [0,1]$ ανάλογα με την απόσταση από το στόχο τους.
- Αν και η ένταση του ήχου μπορεί να ποικίλλει με πολλούς τρόπους, υποθέτουμε ότι κυμαίνεται μεταξύ ενός πολύ μεγάλου θετικού αριθμού A_0 και μιας ελάχιστης τιμής A_{min} .

Είναι προφανές ότι δεν υπάρχει υπολογισμός της απόστασης ούτε και εκτίμηση του τρισδιάστατου χώρου με τη βοήθεια της χρονικής καθυστέρησης. Επίσης για λόγους απλότητας, η συχνότητα Q έχει ένα εύρος $[Q_{min}, Q_{max}]$ που αντιστοιχεί σε ένα εύρος μήκους κύματος $[\lambda_{min}, \lambda_{max}]$. Για παράδειγμα, μία συχνότητα εύρους [20 kHz, 500 kHz] αντιστοιχεί σε ένα εύρος μήκους κύματος από 0.7 mm έως και 17 mm.

Για ένα δεδομένο πρόβλημα μπορούμε για ευκολία να χρησιμοποιήσουμε οποιοδήποτε μήκος κύματος. Στις πραγματικές εφαρμογές μπορούμε να προσαρμόσουμε το εύρος μεταβάλλοντας τα μήκη κύματος (ή τις συχνότητες). Το εύρος (ή το μέγιστο μήκος κύματος) πρέπει να επιλέγεται έτσι ώστε να είναι κατάλληλο για την πολυπλοκότητα και το μέγεθος του εκάστοτε προβλήματος. Για πολυπλοκότερα προβλήματα μπορούν να επιλέγονται μεγαλύτερα εύρη μήκους κύματος. Επίσης, δεν είναι αναγκαίο να χρησιμοποιούμε μόνο τα μήκη κύματος αλλά μπορούμε να μεταβάλλουμε τις συχνότητες καθορίζοντας έτσι και το μήκος κύματος λ . Αυτό συμβαίνει επειδή τα λ και Q σχετίζονται μεταξύ τους, καθώς το γινόμενο λQ είναι σταθερό.

Επίσης για απλότητα, μπορούμε να υποθέσουμε ότι $Q \in [0, Q_{max}]$. Είναι γνωστό ότι υψηλότερες συχνότητες έχουν μικρότερα μήκη κύματος και διανύουν συντομότερες αποστάσεις. Για

τις νυχτερίδες, αυτές οι αποστάσεις είναι λίγα μέτρα. Ο ρυθμός εκπομπής των παλμών μπορεί να ανήκει στο διάστημα $[0,1]$ όπου 0 σημαίνει ότι δεν εκπέμπεται παλμός και το 1 δηλώνει το μέγιστο ρυθμό εκπομπής παλμών.

Με βάση τις παραπάνω περιγραφές και προσεγγίσεις, τα βασικά βήματα του αλγορίθμου της νυχτερίδας μπορούν να συνοψιστούν στον Ψευδοκώδικα 3.1 (Yang, 2010b).

Ψευδοκώδικας 3.1. Bat Algorithm

```

Initialize the bat population,  $x_i$  ( $i = 1, 2, \dots, n$ ) and  $v_i$ 
Initialize frequencies  $Q_i$ , pulse rates  $r_i$  and the loudness  $A_i$ 
while ( $t < \text{Maximum number of iterations}$ )
    Generate new solutions by adjusting frequency
    Updating velocities and locations/solutions
    if ( $\text{rand} > r_i$ )
        Select a solution among the best solutions
        Generate a local solution around the selected best solution
    end if
    Generate a new solution by flying randomly
    if ( $\text{rand} < A_i \ \& \ f(x_i) < f(x_{\text{best}})$ )
        Accept the new solutions
        Increase  $r_i$  and decrease  $A_i$ 
    end if
    Rank the bats and find the current best  $x_{\text{best}}$ 
end while

```

3.2.1. Η κίνηση των ψηφιακών νυχτερίδων

Στις προσομοιώσεις χρησιμοποιούμε «ψηφιακές» νυχτερίδες (virtual bats). Πρέπει λοιπόν να καθορίσουμε τους κανόνες με τους οποίους ανανεώνονται οι θέσεις x_i και οι ταχύτητες v_i σε ένα χώρο αναζήτησης d διαστάσεων. Οι νέες λύσεις x_i^t και οι ταχύτητες v_i^t στην επανάληψη t δίνονται από τις ακόλουθες σχέσεις:

$$Q_i = Q_{\min} + (Q_{\max} - Q_{\min})\beta \quad (3.2)$$

$$v_i^t = v_i^{t-1} + (x_i^{t-1} - x_{\text{best}}) Q_i \quad (3.3)$$

$$x_i^t = x_i^{t-1} + v_i^t \quad (3.4)$$

όπου:

β : ένας τυχαίος αριθμός που λαμβάνεται από μία ομοιόμορφη κατανομή στο διάστημα $[0,1]$.

x_{best} : η ολικά καλύτερη τοποθεσία (λύση) μέχρι την τρέχουσα χρονική στιγμή η οποία εντοπίζεται μετά από σύγκριση όλων των θέσεων μεταξύ των n νυχτερίδων.

Όπως αναφέραμε προηγουμένως, μπορούμε να χρησιμοποιήσουμε τον παράγοντα Q_i (ή τον λ_i) ώστε να μεταβάλλεται η ταχύτητα ενώ ρυθμίζεται και ο άλλος παράγοντας λ_i (ή Q_i), ανάλογα με το είδος του προς εξέταση προβλήματος. Συνήθως ισχύει $Q_{\min} = 0$ και στο Q_{\max} δίνεται μία τιμή η οποία εξαρτάται από το μέγεθος του προβλήματος. Αρχικά, η κάθε νυχτερίδα έχει μία τυχαία συχνότητα από μια ομοιόμορφη κατανομή στο διάστημα $[Q_{\min}, Q_{\max}]$.

Για την τοπική αναζήτηση, αφού επιλεγεί μία λύση μεταξύ των καλύτερων λύσεων της τρέχουσας χρονικής στιγμής, δημιουργείται μια νέα λύση για κάθε νυχτερίδα με τη χρήση της τυχαίας διαδρομής. Δηλαδή:

$$x_{\text{new}} = x_{\text{old}} + \epsilon A^t \quad (3.5)$$

όπου $\epsilon \in [-1,1]$ είναι ένας τυχαίος αριθμός, και $A^t = \langle A_i^t \rangle$ είναι η μέση τιμή των εντάσεων του ήχου για όλες τις νυχτερίδες στην επάληψη t . Συνήθως, αντί της (3.5) χρησιμοποιείται τοπική αναζήτηση γύρω από την τρέχουσα καλύτερη λύση δηλαδή (Fister, Fister and Yang, 2013):

$$x_{\text{new}} = x_{\text{best}} + \epsilon A^t \quad (3.6)$$

Οι υπολογισμοί των ταχυτήτων και των τοποθεσιών σε κάθε επανάληψη έχουν ομοιότητες με τις αντίστοιχες διαδικασίες στον αλγόριθμο PSO καθώς το Q_i ελέγχει το διάστημα της κίνησης των σωματιδίων του σμήνους. Σε κάποιο βαθμό, ο BA μπορεί να θεωρηθεί συνδυασμός του βασικού PSO και μιας τοπικής αναζήτησης που ελέγχονται από την ένταση του ήχου και το ρυθμό εκπομπής παλμών.

3.2.2. Ένταση ήχου και εκπομπή παλμών

Εκτός από τη συχνότητα, τη θέση και την ταχύτητα, σε κάθε επανάληψη πρέπει να ανανεώνονται η ένταση A_i και ο ρυθμός r_i της εκπομπής παλμών. Καθώς μία νυχτερίδα κινείται προς τη λεία της, η ένταση του ήχου μειώνεται ενώ ο ρυθμός εκπομπής παλμών αυξάνεται. Μπορούν να επιλεγούν οποιοσδήποτε τιμές για την ένταση του ήχου, για παράδειγμα $A_0 = 100$ και $A_{\min} = 1$. Για απλότητα μπορούμε να επιλέξουμε $A_0 = 1$ και $A_{\min} = 0$. Η ελάχιστη τιμή $A_{\min} = 0$ δηλώνει ότι μια νυχτερίδα μόλις έχει εντοπίσει τη λεία της και προσωρινά σταματά να εκπέμπει ήχους. Έτσι, έχουμε (Yang, 2010b):

$$A_i^{t+1} = \alpha A_i^t \quad (3.7)$$

και

$$r_i^{t+1} = r_i^0 [1 - \exp(-\gamma t)] \quad (3.8)$$

όπου α και γ είναι σταθερές. Για κάθε $0 < \alpha < 1$ και $\gamma > 0$, έχουμε:

$$A_i^t \rightarrow 0, \quad r_i^t \rightarrow r_i^0 \quad \text{όταν } t \rightarrow \infty \quad (3.9)$$

Στην απλούστερη περίπτωση ισχύει $\alpha = \gamma$.

Όπως και στις άλλες μεταερευτικές μεθόδους, έτσι και εδώ η επιλογή των παραμέτρων γίνεται μετά από πειραματισμό. Αρχικά, κάθε νυχτερίδα πρέπει να έχει διαφορετικές τιμές έντασης του

ήχου και ρυθμό εκπομπής παλμών και αυτό μπορεί να επιτευχθεί με τη βοήθεια της τυχαιότητας. Για παράδειγμα, η αρχική τιμή A_i^0 της έντασης μπορεί να ανήκει στο διάστημα $[1,2]$, ενώ η αρχική τιμή r_i^0 του ρυθμού εκπομπής παλμών μπορεί να είναι κοντά στο μηδέν ή να λαμβάνει τιμή στο διάστημα $[0,1]$. Οι τιμές της έντασης του ήχου και της εκπομπής παλμών των νυχτερίδων θα ενημερώνονται μόνο όταν οι νέες λύσεις είναι βελτιωμένες δηλαδή οι νυχτερίδες κινούνται προς τη βέλτιστη λύση (Yang, 2010b).

3.3. Ανάλυση χαρακτηριστικών και σύγκριση με άλλους αλγόριθμους

Ο αλγόριθμος της νυχτερίδας είναι ένας ευέλικτος και εύρωστος αλγόριθμος. Έρευνες και πειράματα έχουν δείξει ότι ο αλγόριθμος της νυχτερίδας είναι πιο αποδοτικός από τις απλές μορφές γενετικών αλγορίθμων και τη βασική μορφή του PSO (Yang, 2010b) (Khan και Sahai, 2012).

Η βελτιστοποίηση σμήνους σωματιδίων και η αναζήτηση αρμονίας είναι ειδικές περιπτώσεις του αλγορίθμου της νυχτερίδας. Αν αντικαταστήσουμε τις διακυμάνσεις της συχνότητας Q_i με τυχαία παράμετρο και θέσουμε $A_i = 0$ και $r_i = 1$, ο BA ουσιαστικά μετατρέπεται στη βασική μορφή του αλγορίθμου PSO. Παρομοίως, αν παραλείψουμε τις ταχύτητες και οι παράμετροι της έντασης του ήχου και του ρυθμού εκπομπής παλμών είναι σταθερές, για παράδειγμα $A_i = r_i = 0.7$, ο BA μετατρέπεται στον απλό HS. Η αλλαγή της συχνότητας (ή του μήκους κύματος) αντιστοιχεί στην προσαρμογή του τόνου και ο ρυθμός εκπομπής παλμών παίζει το ρόλο του βαθμού θεώρησης της μνήμης αρμονίας.

Ο BA χρησιμοποιεί ηχοεντοπισμό και ρύθμιση συχνοτήτων για την επίλυση προβλημάτων. Έστω και αν ο ηχοεντοπισμός του αλγορίθμου δε μιμείται πιστά τη διαδικασία εντοπισμού των πραγματικών νυχτερίδων, ο BA χρησιμοποιεί διακυμάνσεις θέσεων, ταχυτήτων και συχνοτήτων. Έτσι, ο τρόπος λειτουργίας του μοιάζει με αυτούς των αλγορίθμων PSO και HS. Ο BA εμφανίζει όλα τα πλεονεκτήματα των αλγορίθμων νοημοσύνης σμήνους.

Αντίθετα με άλλες μεταερευτικές μεθόδους όπου οι παράμετροι ελέγχου είναι προκαθορισμένες και σταθερές, στον BA οι τιμές των A και r διαφέρουν για κάθε νυχτερίδα και μεταβάλλονται σε κάθε επανάληψη. Η παράμετρος A ελέγχει την αποδοχή μιας νέας λύσης, ενώ η r ελέγχει την τοπική αναζήτηση. Η αλλαγή των παραμέτρων οδηγεί αυτόματα τον αλγόριθμο από την εξερεύνηση του χώρου λύσεων σε μια διεξοδική τοπική εκμετάλλευση καθώς κινείται προς τη βέλτιστη λύση. Με αυτό τον τρόπο επιτυγχάνεται, σε σύντομο χρονικό διάστημα, εστίαση σε περιοχές όπου έχουν βρεθεί καλές λύσεις με αποτέλεσμα ο αλγόριθμος να συγκλίνει πολύ γρήγορα (Yang, 2013). Η ταχύτερη σύγκλιση είναι ένα ξεχωριστό πλεονέκτημα του BA έναντι των άλλων μεταερευτικών μεθόδων. Κάνει τον αλγόριθμο πολύ αποδοτικό σε εφαρμογές όπως η ταξινόμηση όπου η λύση πρέπει να βρίσκεται πολύ γρήγορα. Ελλοχεύει όμως ο κίνδυνος της πρόωρης σύγκλισης και ελλιπούς εξερεύνησης του χώρου λύσεων. Για να αντιμετωπισθούν αυτές οι αδυναμίες έχουν αναπτυχθεί βελτιωμένες μορφές του αλγορίθμου που παρουσιάζονται παρακάτω.

Η σύγκλιση μπορεί να βελτιωθεί και με την κατάλληλη ρύθμιση των παραμέτρων α και γ (Yang, 2010b). Ουσιαστικά η παράμετρος α έχει παρόμοιο ρόλο με τον παράγοντα ψύξης σε ένα χρονοπρόγραμμα ψύξης της προσομοιωμένης ανόπτωσης. Η λειτουργία του BA είναι απλή αλλά η ρύθμιση των παραμέτρων πολυπλοκότερη σε σχέση με άλλους μεταερευτικούς αλγορίθμους. Καταφέρνει όμως να αξιοποιήσει πολλά από τα πλεονεκτήματα επιτυχημένων αλγορίθμων βελτιστοποίησης, εμπνευσμένος από τη διαδικασία του ηχοεντοπισμού των νυχτερίδων. Οι νέες λύσεις προκύπτουν από τις προσαρμογές των συχνοτήτων, της έντασης του ήχου και του ρυθμού εκπομπής παλμών. Η αποδοχή μία νέας λύσης εξαρτάται από την ποιότητα των λύσεων οι οποίες ελέγχονται από την ένταση του ήχου και το ρυθμό εκπομπής παλμών.

Η φύση του αλγορίθμου της νυχτερίδας μας δίνει πολλές δυνατότητες πειραματισμού. Στην κίνηση των νυχτερίδων, μία ενδιαφέρουσα επέκταση θα μπορούσε να είναι η χρήση διαφορετικών, μη γραμμικών σχημάτων για το μήκος κύματος (ή τη συχνότητα) ή ο περιορισμός στις τιμές των ταχυτήτων (όπως στον PSO). Μπορούμε να πειραματιστούμε ακόμα περισσότερο και να

μεταβάλλουμε με ένα τυχαίο τρόπο τις τιμές των παραμέτρων α και γ κατά τη διάρκεια της εκτέλεσης ή να θεωρήσουμε σταθερές τις παραμέτρους ελέγχου A και r . Ο αλγόριθμος της νυχτερίδας μπορεί ακόμα να συνδυαστεί με άλλες επιτυχημένες μεταερευτικές μεθόδους και να προκύψουν υβριδικές τεχνικές που παρουσιάζουν ακόμα καλύτερα αποτελέσματα.

3.4. Βελτιωμένες μορφές του αλγορίθμου της νυχτερίδας

Ο αλγόριθμος της νυχτερίδας πραγματοποιεί ισχυρή εκμετάλλευση (τοπική αναζήτηση) αλλά υστερεί εν μέρει στην εξερεύνηση του χώρου των λύσεων και μπορεί να εγκλωβιστεί σε τοπικά βέλτιστα. Για να αντιμετωπιστούν αυτές οι αδυναμίες, έχει προταθεί ένα πλήθος μεθόδων και στρατηγικών ώστε να αυξηθεί η ποικιλία των λύσεων και να βελτιωθεί η απόδοση του αλγορίθμου. Βελτιωμένες εκδοχές του αλγορίθμου που έχουν προκύψει, παρουσιάζονται παρακάτω.

3.4.1. Υβριδικός αλγόριθμος νυχτερίδας

Οι Fister, Fister και Yang (2013) σε πρόσφατη έρευνά τους προτείνουν μία νέα μορφή του BA, τον υβριδικό αλγόριθμο νυχτερίδας (Hybrid Bat Algorithm – HBA). Ο νέος αλγόριθμος συνδυάζει τον BA με τον DE που είδαμε στο προηγούμενο κεφάλαιο. Πιο συγκεκριμένα, στον HBA αντί της εξίσωσης (3.6) που χρησιμοποιείται για την τοπική αναζήτηση, εφαρμόζονται οι τελεστές μετάλλαξης, διασταύρωσης και επιλογής του αλγορίθμου DE. Ο Ψευδοκώδικας 3.2 περιγράφει τον HBA.

Ψευδοκώδικας 3.2. Hybrid Bat Algorithm

Objective function $f(x)$, $x = (x_1, x_2, \dots, x_d)^T$

Initialize the bat population x_i and velocities v_i for $i = 1, 2, \dots, n$

Initialize pulse frequency $Q_i \in [Q_{min}, Q_{max}]$

Initialize pulse rates r_i and the loudness A_i

while ($t < \text{Maximum number of iterations}$)

 Generate new solutions by adjusting frequency

 Updating velocities and locations/solutions

if ($\text{rand} > r_i$)

 Modify the solution using “DE/rand/1/bin”

end if

 Generate a new solution by flying randomly

if ($\text{rand} < A_i \ \& \ f(x_i) < f(x_{\text{best}})$)

 Accept the new solutions

 Increase r_i and decrease A_i

end if

 Rank the bats and find current best

end while

Postprocess results and visualization

Η διαφορά του HBA από τον απλό BA εντοπίζεται στο στάδιο της τοπικής αναζήτησης. Στον απλό BA ο ρυθμός εκπομπής παλμών r ελέγχει την τυχαία αναζήτηση. Στον HBA ο r ελέγχει την εφαρμογή των τελεστών της μορφής “DE/rand/1/bin” του αλγορίθμου DE.

3.4.2. Εισαγωγή τελεστή μετάλλαξης

Σε αυτή τη μορφή εκτελείται ο απλός ΒΑ με την προσθήκη ενός τελεστή μετάλλαξης. Με αυτό τον τρόπο γίνεται μεγαλύτερη εξερεύνηση στο χώρο λύσεων και αποφεύγεται η πρόωρη σύγκλιση και ο εγκλωβισμός σε τοπικά βέλτιστα. Η μέθοδος μετάλλαξης που χρησιμοποιούμε περιγράφεται με την ακόλουθη εξίσωση (Azizirpanah-Abarghoosee και Niknam, 2012):

$$x_{\text{new}} = x_{r_1} + \beta_1 \odot (x_{r_2} - x_{r_3}) + \beta_2 \odot (x_{\text{best}} - x_{r_4}) \quad (3.10)$$

Όπου

$x_{r_1}, x_{r_2}, x_{r_3}, x_{r_4}$: 4 λύσεις που επιλέγονται τυχαία από τον πληθυσμό και είναι διάφορες μεταξύ τους και διάφορες της λύσης x_i που εξετάζεται.

β_1, β_2 : δύο διανύσματα τυχαίων αριθμών ομοιόμορφα κατανομημένων στο διάστημα (0,1).

Ο μεταλλαγμένος αλγόριθμος νυχτερίδας (Mutated Bat Algorithm – MBA) αξιολογεί τις λύσεις που έχουν προκύψει από τις εξισώσεις (3.4), (3.6) και (3.10) και επιλέγει την καλύτερη. Η λειτουργία του MBA παρουσιάζεται στον Ψευδοκώδικα 3.3.

Ψευδοκώδικας 3.3. Mutated Bat Algorithm

```

Initialize the bat population,  $x_i$  ( $i = 1, 2, \dots, n$ ) and  $v_i$ 
Initialize frequencies  $Q_i$ , pulse rates  $r_i$  and the loudness  $A_i$ 
while ( $t < \text{Maximum number of iterations}$ )
    Generate new solutions by adjusting frequency
    Updating velocities and locations/solutions  $x_i$ 
    if ( $\text{rand} > r_i$ )
        Select a solution among the best solutions
        Generate a local solution  $x_{\text{new}}$  around the selected best solution
        if ( $f(x_{\text{new}}) < f(x_i)$ )
             $x_i \leftarrow x_{\text{new}}$ 
        end if
    end if
    Generate a solution  $x_m$  using mutation operator
    if ( $f(x_m) < f(x_i)$ )
         $x_i \leftarrow x_m$ 
    end if
    if ( $\text{rand} < A_i \ \& \ f(x_i) < f(x_{\text{best}})$ )
        Accept the new solutions
        Increase  $r_i$  and decrease  $A_i$ 
    end if
    Rank the bats and find the current  $x_{\text{best}}$ 
end while

```

3.4.3. Άλλες μορφές του αλγορίθμου της νυχτερίδας

Πέρα από τις δύο εκδοχές που παρουσιάσαμε και θα εφαρμόσουμε στην παρούσα διατριβή, άλλες παραλλαγές του BA που συναντώνται στη βιβλιογραφία είναι οι εξής:

- Fuzzy Logic Bat Algorithm (FLBA): Οι Khan, NIKON και Sahai (2011) εισάγουν ασαφή λογική στον BA για την επίλυση ενός προβλήματος ομαδοποίησης (clustering).
- Bat Algorithm with Mutation (BAM): Στον αλγόριθμο του Wang κ.ά. (2012) ανάλογα με την τιμή του r εκτελείται τυχαία αναζήτηση ή μετάλλαξη του απλού DE. Ο BAM εφαρμόζεται στο σχεδιασμό μονοπατιών μη επανδρωμένων εναέριων οχημάτων μάχης (Uninhabited Combat Aerial Vehicle Path Planning – UCAV Path Planning).
- Binary Bat Algorithm (BBA): Ο Nakamura κ.ά. (2012) προτείνουν μία διακριτή μορφή του BA για την επίλυση προβλημάτων ταξινόμησης και επιλογής χαρακτηριστικών.
- Improved Bat Algorithm I (IBA): Οι Azizipanah-Abarghoee και Niknam (2012) προτείνουν δυο εξισώσεις για την τοπική αναζήτηση. Κάθε εξίσωση επιλέγεται ανάλογα με την τιμή της παραμέτρου r . Για καλύτερη εξερεύνηση του χώρου λύσεων χρησιμοποιούν τον τελεστή μετάλλαξης της εξίσωσης (3.10). Για μία νυχτερίδα, μία νέα λύση μπορεί να γίνει δεκτή με πιθανότητα A_i αν δεν είναι καλύτερη από τη λύση που είχε βρει στην προηγούμενη επανάληψη.
- Harmony Search / Bat Algorithm (HS/BA): Οι Wang και Guo (2013) προτείνουν έναν υβριδικό αλγόριθμο που συνδυάζει τον BA με τον HS. Η συχνότητα Q και η ένταση A έχουν σταθερές τιμές.
- Differential Operator and Lévy Flights Bat Algorithm (DLBA): Στον αλγόριθμο που προτείνει ο Xie κ.ά. (2013), για την παραγωγή νέων λύσεων χρησιμοποιείται ένας τελεστής μετάλλαξης παρόμοιος με αυτόν του σχήματος “DE/best/2” του αλγορίθμου DE ενώ οι νυχτερίδες εκτελούν πτήσεις Lévy (Lévy flights). Επίσης εκτελούνται δύο τοπικές αναζητήσεις, μία που ελέγχεται από τον r και μία από τον A .
- Improved Bat Algorithm II (IBA): Οι Yilmaz και Kucuksille (2013) χρησιμοποιούν συντελεστή αδράνειας που μειώνεται γραμμικά όπως και στον PSO. Οι συχνότητες διαμορφώνονται με χρήση των αποστάσεων μεταξύ των τρέχουσων λύσεων και της καλύτερης λύσης. Για μεγαλύτερη εξερεύνηση του χώρου των λύσεων, εισάγεται το στάδιο ανίχνευσης που είδαμε στον αλγόριθμο ABC. Ο IBA εφαρμόζεται σε συνεχή προβλήματα βελτιστοποίησης.

4. ΤΟ ΠΡΟΒΛΗΜΑ ΤΗΣ ΣΥΝΔΥΑΣΜΕΝΗΣ ΚΑΤΑΝΟΜΗΣ ΟΙΚΟΝΟΜΙΚΟΥ ΚΟΣΤΟΥΣ ΚΑΙ ΕΚΠΟΜΠΗΣ ΡΥΠΩΝ

Τα τελευταία χρόνια, τα συστήματα παραγωγής ενέργειας έχουν γίνει πιο ογκώδη και πιο πολύπλοκα. Ένα από τα σημαντικότερα προβλήματα των οργανισμών παραγωγής ηλεκτρικής ενέργειας είναι η εύρεση λύσεων με σκοπό την ικανοποίηση της ζήτησης μαζί με την ελαχιστοποίηση του κόστους παραγωγής. Η οικονομική κατανομή φορτίου (Economic Load Dispatch – ELD) ή πιο απλά οικονομική κατανομή είναι ένα βασικότατο πρόβλημα σε ένα σύστημα παραγωγής ενέργειας αφού αφορά το λειτουργικό κόστος του συστήματος. Στις παραδοσιακές οικονομικές μελέτες, το λειτουργικό κόστος μπορεί να μειωθεί με την κατάλληλη τοπολογία της ποσότητας της παραχθείσας ενέργειας από διαφορετικές μονάδες παραγωγής. Ο στόχος της οικονομικής κατανομής φορτίου είναι να καθοριστεί η κατανομή του οικονομικού φόρτου των γεννητριών με τέτοιο τρόπο ώστε να ελαχιστοποιηθεί το κόστος παραγωγής ενέργειας ενώ ταυτόχρονα πρέπει να ικανοποιείται η συνολική ζήτηση του δικτύου και να μην υπερβαίνονται τα λειτουργικά όρια του κάθε σταθμού παραγωγής. Πρόκειται για ένα κλασικό πρόβλημα βελτιστοποίησης για το οποίο έχουν προταθεί παραδοσιακές τεχνικές όπως η μέθοδος της επανάληψης λάμδα (Lambda Iteration Method) και ο τετραγωνικός προγραμμαρισμός (Quadratic Programming) ενώ μεταγενέστερες τεχνικές, όπως οι γενετικοί αλγόριθμοι και ο PSO έχουν δώσει πολύ καλά αποτελέσματα (Krishnan και Krishnan, 2011).

Ωστόσο, η ελαχιστοποίηση του κόστους παραγωγής δεν είναι πλέον το μοναδικό κριτήριο που πρέπει να ληφθεί υπόψη. Πολλές χώρες παγκοσμίως έχουν επικεντρωθεί στη μείωση της εκπομπής ρυπογόνων ουσιών από τις μονάδες παραγωγής που χρησιμοποιούν ορυκτά καύσιμα. Η παραγωγή ηλεκτρικής ενέργειας με τη χρήση ορυκτών καυσίμων, εκτός από τους στερεούς (σε μορφή σωματιδίων) ρύπους εκλύει στην ατμόσφαιρα αέριους ρύπους όπως το διοξείδιο του άνθρακα (CO_2), οξείδια του θείου (SO_x) και οξείδια του αζώτου (NO_x) (Basu, 2011). Οι συγκεκριμένες ρυπογόνες ουσίες είναι ιδιαίτερα βλαβερές για τη χλωρίδα, την πανίδα και την ανθρώπινη υγεία. Επομένως, προκύπτει ένα δεύτερο πρόβλημα, η κατανομή εκπομπής ρύπων (Emission Dispatch – ED) που έχει ως σκοπό να ελαχιστοποιήσει την ποσότητα των ρυπογόνων ουσιών που απελευθερώνονται στο περιβάλλον. Άρα κατά την ελαχιστοποίηση του οικονομικού κόστους πρέπει να εξετάζεται η ατμοσφαιρική ρύπανση.

Λαμβάνοντας υπόψη μόνο τον περιορισμό της επιβάρυνσης του περιβάλλοντος προκύπτει μεγάλη αύξηση στο κόστος παραγωγής. Από την άλλη πλευρά, η λειτουργία του συστήματος παραγωγής ενέργειας με το ελάχιστο οικονομικό κόστος επιφέρει αύξηση της εκπομπής των ρύπων σε μη αποδεκτά επίπεδα. Είναι αναγκαίο λοιπόν να καταρτιστεί μια στρατηγική κατανομής της λειτουργίας των θερμικών μονάδων παραγωγής του συστήματος έτσι ώστε να επιτυγχάνεται η ταυτόχρονη ελαχιστοποίηση του κόστους λειτουργίας και της εκπομπής ρύπων.

Έτσι, προκύπτει το πρόβλημα της συνδυασμένης κατανομής οικονομικού κόστους και εκπομπής ρύπων (Combined Economic and Emission Dispatch – CEED)³. Πρόκειται για ένα πολυκριτηριακό πρόβλημα βελτιστοποίησης όπου ζητείται η ελαχιστοποίηση δύο μεγεθών που «συγκρούονται» μεταξύ. Τα σύγχρονα συστήματα παραγωγής ηλεκτρικής ενέργειας αποτελούνται από ένα μεγάλο πλήθος γεννητριών και κατασκευάζονται με μη γραμμικά χαρακτηριστικά. Επομένως απαιτείται η βελτιστοποίηση πολύπλοκων αντικειμενικών συναρτήσεων υπό μία σειρά πρακτικών περιορισμών. Οι μεταβλητές του προβλήματος λαμβάνουν πραγματικές τιμές που σημαίνει ότι παρουσιάζουν απειρία τιμών σε οποιοδήποτε διάστημα του πεδίου ορισμού τους. Άρα αναζητούνται λύσεις σε ένα συνεχές πρόβλημα βελτιστοποίησης. Η σπουδαιότητα του προβλήματος CEED είναι πολύ μεγάλη καθώς επηρεάζει τη σταθερότητα των συστημάτων παραγωγής, γεγονός που έχει οδηγήσει μέχρι και σήμερα τους ερευνητές στην ανάπτυξη ενός μεγάλου πλήθους μεθόδων για την επίλυσή του.

³ Το πρόβλημα CEED αναφέρεται στην βιβλιογραφία και ως «περιβαλλοντική και οικονομική κατανομή (Environmental / Economic Dispatch – EED)».

4.1. Οικονομική κατανομή φορτίου

Το πρόβλημα της οικονομικής κατανομής φορτίου είναι από τα κυρίαρχα όσον αφορά τα συστήματα λειτουργίας και σχεδιασμού δικτύων μονάδων παραγωγής ενέργειας. Μπορεί να περιγραφεί ως η κατανομή της ενέργειας που θα παραχθεί από κάθε θερμική μονάδα, ώστε να ελαχιστοποιηθεί το κόστος καυσίμων και να ικανοποιηθούν οι περιορισμοί ζήτησης και ορίων παραγωγής.

Η αντικειμενική συνάρτηση του οικονομικού κόστους καυσίμων είναι διωνυμική και δίνεται από τη σχέση:

$$F = \sum_{i=1}^n (a_i P_i^2 + b_i P_i + c_i) \quad (4.1)$$

Όταν σε μία γεννήτρια υπάρχουν πολλαπλές βαλβίδες εισαγωγής ατμού (valve points), η καμπύλη του κόστους δεν είναι ομαλή όπως φαίνεται και στην Εικόνα 4.1. Η επίδραση των βαλβίδων ατμού μπορεί να ληφθεί υπόψη με την πρόσθεση μιας ημιτονοειδούς συνάρτησης στη διωνυμική (Kothari και Dhillon, 2011) (Basu, 2011). Τότε, το κόστος καυσίμων υπολογίζεται ως εξής:

$$F = \sum_{i=1}^n \left[a_i P_i^2 + b_i P_i + c_i + \left| d_i \sin \left(e_i (P_{i,\min} - P_i) \right) \right| \right] \quad (4.2)$$

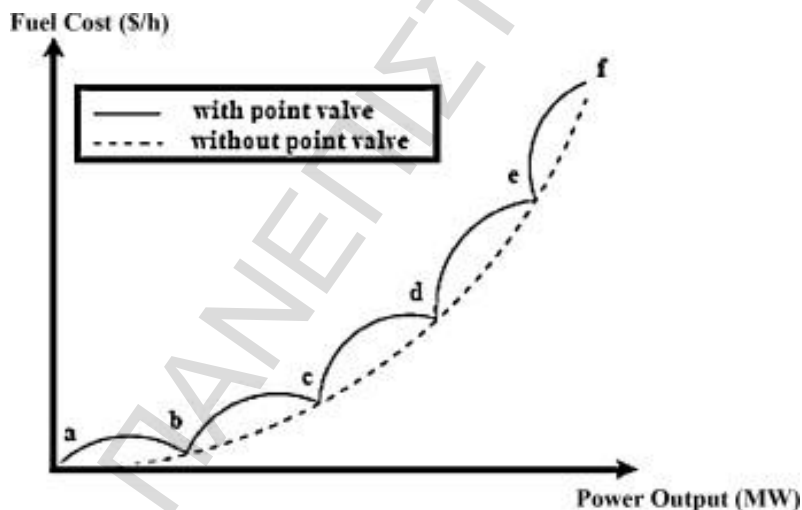
όπου

F : το συνολικό κόστος καυσίμων στο σύστημα

n : ο αριθμός των θερμικών μονάδων παραγωγής ενέργειας

P_i : η ισχύς (ο ρυθμός παραγωγής ενέργειας) της i -οστής μονάδας παραγωγής

a_i, b_i, c_i, d_i, e_i : οι συντελεστές του κόστους καυσίμων της i -οστής μονάδας παραγωγής



Εικόνα 4.1. Καμπύλες των συναρτήσεων του οικονομικού κόστους. Όταν δε λαμβάνονται υπόψη οι βαλβίδες ατμού η καμπύλη είναι ομαλή. Η επίδραση των βαλβίδων απεικονίζεται με το σχηματισμό μιας μη ομαλής καμπύλης.

Η αντικειμενική συνάρτηση κόστους υποτάσσεται στους παρακάτω περιορισμούς:

- Περιορισμός ζήτησης (περιορισμός ισότητας):

$$\sum_{i=1}^n P_i - P_D - P_L = 0 \quad (4.3)$$

όπου:

P_D : η συνολική ζήτηση του δικτύου

P_L : η συνολική απώλεια του δικτύου στις γραμμές μεταφοράς.

Η απώλεια του δικτύου υπολογίζεται ως εξής:

$$P_L = \sum_{i=1}^n \sum_{j=1}^n P_i B_{ij} P_j \quad (4.4)$$

όπου B_{ij} ο συντελεστής απώλειας από το μέσο μεταφοράς μεταξύ των σταθμών παραγωγής i και j .

- Όρια δυνατότητας παραγωγής (περιορισμοί ανισότητας):

$$P_{i,\min} \leq P_i \leq P_{i,\max} \quad (4.5)$$

όπου

$P_{i,\min}$: η ελάχιστη ποσότητα ισχύος που μπορεί να παράγει η i -οστή γεννήτρια

$P_{i,\max}$: η μέγιστη ποσότητα ισχύος που μπορεί να παράγει η i -οστή γεννήτρια

4.2. Κατανομή εκπομπής ρύπων

Ο στόχος της κατανομής εκπομπής ρύπων είναι η ρύθμιση της λειτουργίας για κάθε γεννήτρια ώστε να ελαχιστοποιηθεί η ποσότητα της εκπομπής ρύπων που οφείλεται στην καύση των ορυκτών καυσίμων για την παραγωγή ενέργειας, ικανοποιώντας ταυτόχρονα του περιορισμούς ζήτησης και ορίων παραγωγής. Οι ατμοσφαιρικοί ρύποι όπως το διοξείδιο του άνθρακα (CO_2), τα οξειδία του θείου (SO_x) και τα οξειδία του αζώτου (NO_x) μπορούν να μοντελοποιηθούν ξεχωριστά. Για λόγους απλότητας, μπορεί να ληφθεί υπόψη μόνο η ποσότητα των οξειδίων του αζώτου ως η βλαβερότερη όλων. Η εκπομπή NO_x προσεγγίζεται με μία διωνυμική εξίσωση, δηλαδή:

$$E = \sum_{i=1}^n (\alpha_i P_i^2 + \beta_i P_i + \gamma_i) \quad (4.6)$$

Ο υπολογισμός της εκπομπής μπορεί να συμπεριλαμβάνει όλα τα ρυπογόνα αέρια που απελευθερώνονται στο περιβάλλον με το άθροισμα μιας εκθετικής συνάρτησης στη διωνυμική (Basu, 2011). Σε αυτήν την περίπτωση, η εξίσωση της εκπομπής ρύπων διαμορφώνεται ως εξής:

$$E = \sum_{i=1}^n (\alpha_i P_i^2 + \beta_i P_i + \gamma_i + \eta_i \exp(\delta_i P_i)) \quad (4.7)$$

όπου:

E : η συνολική εκπομπή ρύπων

$\alpha_i, \beta_i, \gamma_i, \eta_i, \delta_i$: οι συντελεστές εκπομπής ρύπων της i -οστής μονάδας παραγωγής

Οι περιορισμοί των εξισώσεων (4.3) και (4.5) πρέπει να ικανοποιούνται και στην κατανομή εκπομπής ρύπων.

4.3. Συνδυασμένη κατανομή οικονομικού κόστους και εκπομπής ρύπων

Οι δύο κατανομές που περιγράψαμε διαφέρουν σημαντικά. Η οικονομική κατανομή ελαχιστοποιεί το λειτουργικό κόστος (κόστος καυσίμων) ενώ η κατανομή εκπομπής ρύπων ελαχιστοποιεί την ποσότητα των ρύπων. Η συνδυασμένη κατανομή οικονομικού κόστους και εκπομπής ρύπων στοχεύει στην ταυτόχρονη ελαχιστοποίηση και των δύο μεγεθών.

Το πρόβλημα CEED μπορεί να παρουσιαστεί ως εξής:

$$\min\{G = (F, E)\} \quad (4.8)$$

και υποτάσσεται στους περιορισμούς (4.3) και (4.5).

Το πολυκριτηριακά προβλήματα βελτιστοποίησης (όπως το CEED), αντίθετα με τα προβλήματα μιας αντικειμενικής συνάρτησης, δεν έχουν πάντα μία βέλτιστη λύση η οποία ελαχιστοποιεί ταυτόχρονα όλες τις αντικειμενικές συναρτήσεις. Συχνά, υπάρχουν αντικειμενικές συναρτήσεις που συγκρούονται μεταξύ τους και οι βέλτιστες παράμετροι μιας συνάρτησης δε βελτιστοποιούν πάντα τις υπόλοιπες συναρτήσεις. Αντιθέτως μπορεί να χειροτερεύουν μερικές από αυτές. Στο πρόβλημα CEED θέλουμε να ελαχιστοποιήσουμε το κόστος καυσίμων και τις εκπομπές ρύπων ταυτόχρονα. Τα πολύ φθηνά καύσιμα όμως προκαλούν μεγαλύτερη περιβαλλοντική επιβάρυνση που σημαίνει ότι η αντικειμενική συνάρτηση οικονομικού κόστους συγκρούεται με την αντικειμενική συνάρτηση εκπομπής ρύπων.

Επομένως, πρέπει να αναζητήσουμε ένα σημείο ισορροπίας μεταξύ των συγκρουόμενων αντικειμενικών συναρτήσεων. Είναι αναγκαίο να συγκρίνουμε τις αντικειμενικές συναρτήσεις και να προβούμε σε ορισμένους συμβιβασμούς μεταξύ τους. Γίνεται εύκολα αντιληπτό ότι η άμεση επίλυση ενός πολυκριτηριακού προβλήματος είναι πολύ δύσκολη ιδιαίτερα όταν οι αντικειμενικές συναρτήσεις ανταγωνίζονται μεταξύ τους. Πρέπει λοιπόν να υιοθετηθεί ένα διαφορετικό μοντέλο που καθιστά δυνατή την επίλυση του πολυκριτηριακού προβλήματος.

4.4. Βελτιστοποίηση κατά Pareto

Ένα διάνυσμα $u = (u_1, u_2, \dots, u_n)^T$ θεωρούμε ότι κυριαρχεί ή υπερिशχύει ενός άλλου διανύσματος $v = (v_1, v_2, \dots, v_n)^T$ αν και μόνο αν:

$$\forall i \in \{1, 2, \dots, n\}: u_i \leq v_i \wedge \exists i \in \{1, 2, \dots, n\}: u_i < v_i \quad (4.9)$$

Δηλαδή δεν υπάρχει στοιχείο του u το οποίο είναι μεγαλύτερο από το αντίστοιχο στοιχείο του v και τουλάχιστον ένα στοιχείο του u είναι μικρότερο από το αντίστοιχο στοιχείο του v . Η συγκεκριμένη σχέση συμβολίζεται ως εξής:

$$u < v \quad (4.10)$$

Παρομοίως, μπορούμε να ορίσουμε μία άλλη σχέση κυριαρχίας \leq ως εξής:

$$u \leq v \Leftrightarrow u < v \vee u = v \quad (4.11)$$

Οι σχέσεις κυριαρχίας που μόλις είδαμε, ισχύουν για προβλήματα ελαχιστοποίησης. Για προβλήματα μεγιστοποίησης η σχέση $<$ αντικαθίσταται με την $>$.

Μία λύση $s \in S$ ενός προβλήματος βελτιστοποίησης καλείται κατά Pareto βέλτιστη (Pareto optimal solution) ή μη κατώτερη (non-inferior) ή μη κυριαρχούμενη (non-dominated) αν και μόνο αν δεν υπάρχει άλλη λύση $s' \in S$ τέτοια ώστε $f_i(s') \leq f_i(s)$ για κάθε $i = 1, 2, \dots, I$. Με άλλα λόγια, η s είναι μία λύση Pareto αν και μόνο αν δεν υπάρχει εφικτή λύση s' η οποία μπορεί να βελτιώσει κάποιες αντικειμενικές συναρτήσεις χωρίς ταυτόχρονα να υποβαθμίσει τουλάχιστον μία άλλη αντικειμενική συνάρτηση. Κάθε λύση Pareto δεν μπορεί να κυριαρχηθεί από άλλη εφικτή λύση.

Σε αντίθεση με τη μονοκριτηριακή βελτιστοποίηση όπου συνήθως υπάρχει μία μόνο βέλτιστη λύση, η πολυκριτηριακή βελτιστοποίηση οδηγείται σε ένα σύνολο βέλτιστων λύσεων. Το σύνολο P^* όλων των κατά Pareto βέλτιστων λύσεων ονομάζεται κατά Pareto βέλτιστο σύνολο (Pareto optimal set) ή πιο απλά σύνολο Pareto (Pareto set) (Yang, 2010a). Δηλαδή, το σύνολο των βέλτιστων λύσεων στο χώρο αναζήτησης συγκροτούν το σύνολο Pareto. Έτσι έχουμε:

$$P^* = \{s \in S \mid \nexists s' \in S: \varphi' < \varphi\} \quad (4.12)$$

όπου $\varphi = [f_1(s), f_2(s), \dots, f_I(s)]$ και $\varphi' = [f_1(s'), f_2(s'), \dots, f_I(s')]$

Προφανώς τα διανύσματα του συγκεκριμένου συνόλου δεν κυριαρχούνται από άλλα διανύσματα.

Η απεικόνιση του συνόλου Pareto στο χώρο απόκρισης ονομάζεται μέτωπο Pareto (Pareto front) (Yang, 2010a). Δηλαδή τα διανύσματα των τιμών των αντικειμενικών συναρτήσεων που προέρχονται από τις λύσεις του συνόλου Pareto συγκροτούν το μέτωπο Pareto P . Έχουμε:

$$P = \{\varphi \in \Phi \mid \nexists \varphi' \in \Phi: \varphi' < \varphi\} \quad (4.13)$$

Όταν υπάρχουν παραπάνω από μία λύσεις Pareto είναι αδύνατο να προτιμήσουμε μία λύση έναντι των άλλων χωρίς να έχουμε επιπρόσθετες πληροφορίες για το πρόβλημα. Επιπλέον, ο προσδιορισμός του μετώπου Pareto δεν είναι πάντα εύκολη υπόθεση και χρειάζεται ανάλυση των παραμέτρων του προβλήματος. Στην Εικόνα 4.2 βλέπουμε ένα παράδειγμα μετώπου Pareto για ένα πρόβλημα δύο αντικειμενικών συναρτήσεων.

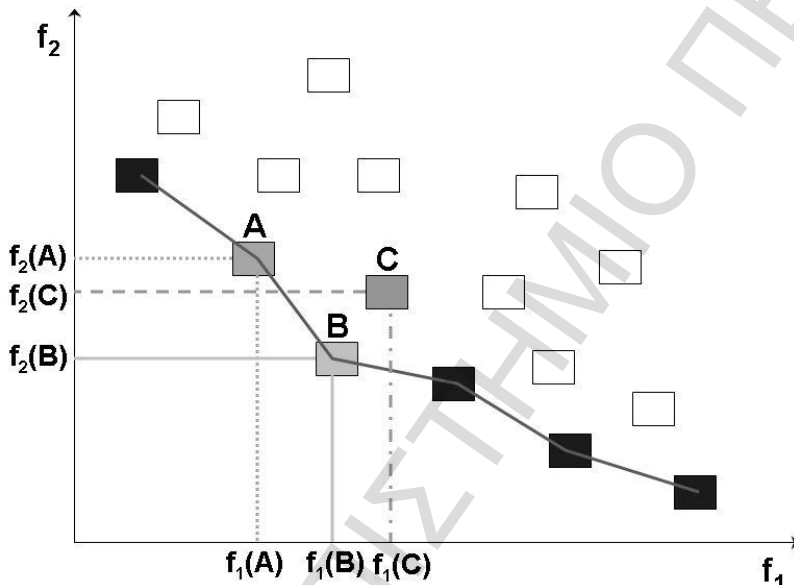
Η πολυκριτηριακή βελτιστοποίηση λοιπόν, είναι ένα δύσκολο και πολύπλοκο θέμα. Για την επίλυση προβλημάτων βελτιστοποίησης με παραπάνω από μία αντικειμενικές συναρτήσεις, η πιο διαδεδομένη μέθοδος που έχει προταθεί είναι ο μετασχηματισμός του πολυκριτηριακού

προβλήματος σε μονοκριτηριακό. Αυτό επιτυγχάνεται με τη δημιουργία μιας συνάρτησης που αποτελεί ένα σταθμισμένο συνδυασμό των αντικειμενικών συναρτήσεων. Η συνάρτηση αυτή καλείται συνάρτηση χρησιμότητας (utility function) και ένας απλός τρόπος κατασκευής της είναι η χρήση του σταθμισμένου αθροίσματος, δηλαδή (Yang, 2010a):

$$u(f_1(x), \dots, f_l(x)) = \sum_{i=1}^l a_i f_i(x) \quad (4.14)$$

όπου a_i είναι οι συντελεστές στάθμισης. Με αυτόν τον τρόπο οι μέθοδοι βελτιστοποίησης που έχουν προταθεί μπορούν να εφαρμοστούν όμως οι λύσεις εξαρτώνται από την επιλογή των βαρών a_i .

Για το μετασχηματισμό του προβλήματος CEED που εξετάζουμε από πολυκριτηριακό σε μονοκριτηριακό, δύο μέθοδοι εφαρμόζονται πολύ συχνά: η εισαγωγή παράγοντα ποινής και η μέθοδος σταθμισμένου αθροίσματος.



Εικόνα 4.2. Μέτωπο Pareto για πρόβλημα βελτιστοποίησης δύο αντικειμενικών συναρτήσεων. Τα τετράγωνα αναπαριστούν τις εφικτές λύσεις του προβλήματος. Η λύση B κυριαρχεί επί της C καθώς δίνει μικρότερες τιμές και στις δύο αντικειμενικές συναρτήσεις f_1 και f_2 . Οι A και B δεν κυριαρχούνται από άλλες λύσεις άρα είναι βέλτιστες κατά Pareto (μη κυριαρχούμενες).

4.5. Εισαγωγή παράγοντα ποινής

Για το μετασχηματισμό του CEED σε πρόβλημα μίας αντικειμενικής συνάρτησης, οι Palanichamy και Srikrishna (1991) πρότειναν μια μέθοδο στην οποία γίνεται χρήση ενός παράγοντα ποινής (penalty factor) h . Η αντικειμενική συνάρτηση που θα ελαχιστοποιηθεί διαμορφώνεται ως εξής:

$$G = F + h \cdot E \quad (4.15)$$

Η τιμή του h καθορίζεται από μία διαδικασία που ακολουθεί τα παρακάτω βήματα:

1. Υπολογισμός του κόστους καυσίμων της κάθε θερμικής μονάδας κατά τη μέγιστη παραγωγή ενέργειας, δηλαδή:

$$F_i(P_{i,\max}) = a_i P_{i,\max}^2 + b_i P_{i,\max} + c_i \quad (4.16)$$

ή

$$F_i(P_{i,\max}) = a_i P_{i,\max}^2 + b_i P_{i,\max} + c_i + \left| d_i \sin \left(e_i (P_{i,\min} - P_{i,\max}) \right) \right| \quad (4.17)$$

όπου F_i το κόστος της μονάδας παραγωγής i .

2. Υπολογισμός της ποσότητας των εκπεμπόμενων ρύπων της κάθε θερμικής μονάδας κατά τη μέγιστη παραγωγή ενέργειας, δηλαδή:

$$E_i(P_{i,\max}) = \alpha_i P_{i,\max}^2 + \beta_i P_{i,\max} + \gamma_i \quad (4.18)$$

ή

$$E_i(P_{i,\max}) = \alpha_i P_{i,\max}^2 + \beta_i P_{i,\max} + \gamma_i + \eta_i \exp(\delta_i P_{i,\max}) \quad (4.19)$$

όπου E_i η εκπομπή ρύπων της μονάδας παραγωγής i .

3. Υπολογισμός του λόγου h_i του κόστους $F_i(P_{i,\max})$ προς την ποσότητα $E_i(P_{i,\max})$ των εκπεμπόμενων ρύπων για κάθε μονάδα παραγωγής i , δηλαδή:

$$h_i = \frac{F_i(P_{i,\max})}{E_i(P_{i,\max})} \quad (4.20)$$

4. Ταξινόμηση των λόγων h_i κατά αύξουσα σειρά.
5. Η μέγιστη ισχύς $P_{i,\max}$ της κάθε μονάδας παραγωγής προστίθεται μία κάθε φορά, ξεκινώντας από τη μονάδα με το μικρότερο h_i έως ότου $\sum P_{i,\max} \geq P_D$.
6. Το h_i που αντιστοιχεί στη μονάδα παραγωγής του τελευταίου $P_{i,\max}$ που προστέθηκε στο βήμα 5 αποτελεί τον παράγοντα ποινής h για τη δεδομένη ζήτηση ενέργειας P_D .

Μια άλλη προσέγγιση αφορά τον υπολογισμό του παράγοντα ποινής για κάθε γεννήτρια ξεχωριστά. Σε αυτή την περίπτωση, η αντικειμενική συνάρτηση είναι η εξής:

$$G = \sum_{i=1}^n (F_i + h_i \cdot E_i) \quad (4.21)$$

όπου n ο αριθμός των γεννητριών. Έχουν προταθεί 5 διαφορετικοί τρόποι υπολογισμού του h_i (Krishnamurthy και Tzoneva, 2012):

- Min-Max penalty factor:

$$h_i = \frac{F_i(P_{i,\min})}{E_i(P_{i,\max})} \quad (4.22)$$

- Max-Max penalty factor:

$$h_i = \frac{F_i(P_{i,\max})}{E_i(P_{i,\max})} \quad (4.23)$$

- Min-Min penalty factor:

$$h_i = \frac{F_i(P_{i,\min})}{E_i(P_{i,\min})} \quad (4.24)$$

- Max-Min penalty factor:

$$h_i = \frac{F_i(P_{i,\max})}{E_i(P_{i,\min})} \quad (4.25)$$

- Average penalty factor:

$$h_i = \frac{\frac{F_i(P_{i,\min})}{E_i(P_{i,\max})} + \frac{F_i(P_{i,\max})}{E_i(P_{i,\max})} + \frac{F_i(P_{i,\min})}{E_i(P_{i,\min})} + \frac{F_i(P_{i,\max})}{E_i(P_{i,\min})}}{4} \quad (4.26)$$

Ένας ακόμη τρόπος υπολογισμού για τον παράγοντα ποινής δίνεται από την παρακάτω εξίσωση (Krishnamurthy και Tzoneva, 2012):

$$h = \frac{\sum_{i=1}^n \frac{\frac{F_i(P_{i,\min})}{E_i(P_{i,\max})} + \frac{F_i(P_{i,\max})}{E_i(P_{i,\max})} + \frac{F_i(P_{i,\min})}{E_i(P_{i,\min})} + \frac{F_i(P_{i,\max})}{E_i(P_{i,\min})}}{4}}{n} \quad (4.27)$$

Όταν πλέον γνωρίζουμε την τιμή του h ή των h_i μπορούμε να επιλύσουμε το πρόβλημα CEED ελαχιστοποιώντας την τιμή της συνάρτησης της εξίσωσης (4.15) ή (4.21) υποκείμενης στους περιορισμούς (4.3) και (4.5).

4.6. Μέθοδος σταθμισμένου αθροίσματος

Το CEED μπορεί να μετατραπεί σε μονοκριτηριακό πρόβλημα με τη χρήση της μεθόδου σταθμισμένων συναρτήσεων. Η αντικειμενική συνάρτηση που θα ελαχιστοποιηθεί έχει την ακόλουθη μορφή:

$$G = w_1 F + w_2 E \quad (4.28)$$

με

$$w_1 + w_2 = 1 \quad (4.29)$$

και

$$w_1, w_2 \geq 0 \quad (4.30)$$

Προφανώς για $w_2 = 0$ προκύπτει το πρόβλημα ELD και για $w_1 = 0$ προκύπτει το πρόβλημα ED. Ο καλύτερος συνδυασμός τιμών των w_1 και w_2 καθορίζεται με τη βοήθεια των ασαφών συνόλων. Η μέθοδος που ακολουθείται ονομάζεται cardinal priority ranking (Kothari και Dhillon, 2011) (Manteaw και Odero, 2012).

Τα ασαφή σύνολα περιγράφονται από εξισώσεις που ονομάζονται συναρτήσεις συμμετοχής (membership functions). Η συνάρτηση συμμετοχής μ απεικονίζει το βαθμό επάρκειας της αντικειμενικής συνάρτησης f_i και παίρνει τιμές από 0 έως και 1, με $\mu(f_i) = 0$ η αντικειμενική συνάρτηση να μην είναι καθόλου επαρκής και $\mu(f_i) = 1$ η συνάρτηση να είναι πλήρως επαρκής. Με αυτό τον τρόπο οι αντικειμενικές συναρτήσεις διαβαθμίζονται με κλίμακα από 0 έως και 1. Η συνάρτηση συμμετοχής υπολογίζεται ως εξής:

$$\mu(f_i) = \begin{cases} 1, & f_i \leq f_{i,\min} \\ \frac{f_{i,\max} - f_i}{f_{i,\max} - f_{i,\min}}, & f_{i,\min} < f_i < f_{i,\max} \\ 0, & f_i \geq f_{i,\max} \end{cases} \quad (4.31)$$

όπου $f_{i,\min}$ είναι η καλύτερη τιμή της αντικειμενικής συνάρτησης f_i και υπολογίζεται θέτοντας το βάρος της ίσο με 1 και τα βάρη των υπόλοιπων αντικειμενικών συναρτήσεων ίσα με 0 (Singh, Dhillon και Chauhan, 2006). Αντίστοιχα η χειρότερη τιμή $f_{i,\max}$ βρίσκεται αν τεθεί το βάρος της i -οστής αντικειμενικής συνάρτησης ίσο με 0. Η τιμή της συνάρτησης συμμετοχής υποδηλώνει το βαθμό (με κλίμακα από 0 έως και 1) στον οποίο μία μη κυριαρχούμενη λύση ικανοποιεί την i -οστή αντικειμενική συνάρτηση. Το άθροισμα των τιμών της συνάρτησης συμμετοχής όλων των αντικειμενικών συναρτήσεων μπορεί να αξιολογήσει το βαθμό επιτυχίας κάθε λύσης. Για κάθε λύση κανονικοποιούμε αυτό το βαθμό επιτυχίας ως εξής:

$$R_k = \frac{\sum_{i=1}^L \mu_k(f_i)}{\sum_{k=1}^M \sum_{i=1}^L \mu_k(f_i)} \quad (4.32)$$

όπου:

R_k : η τιμή της κανονικοποιημένης συνάρτησης συμμετοχής

L : το πλήθος των αντικειμενικών συναρτήσεων

M : το πλήθος των Pareto λύσεων

Η λύση στην οποία αντιστοιχεί η μεγαλύτερη τιμή του R_k είναι αυτή που επιλέγεται ως η καλύτερη συμβιβαστική λύση (best compromise solution). Παρακάτω βλέπουμε τον Ψευδοκώδικα της μεθόδου επιλογής βαρών που μόλις περιγράψαμε (Dhillon και Jain, 2011).

Ψευδοκώδικας 4.1. Cardinal Priority Raking

for $k \leftarrow 1$ to M

for $i \leftarrow 1$ to L

if $f_i^{(k)} \geq f_{i,\max}$

$\mu_k(f_i) = 0$

else if $f_i^{(k)} > f_{i,\min} \vee f_i^{(k)} < f_{i,\max}$

$\mu_k(f_i) = (f_{i,\max} - f_i^{(k)}) / (f_{i,\max} - f_{i,\min})$

else $\mu_k(f_i) = 1$

end if

end for

end for

for $k \leftarrow 1$ to M

$R_k = \sum_{i=1}^L \mu_k(f_i) / \sum_{k=1}^M \sum_{i=1}^L \mu_k(f_i)$

end for

Select solution x_k with $\max \{R_k : k = 1, 2, \dots, M\}$

5. ΕΦΑΡΜΟΓΕΣ ΚΑΙ ΑΠΟΤΕΛΕΣΜΑΤΑ

Θα μελετήσουμε τρία διαφορετικά συστήματα παραγωγής ενέργειας. Στα δύο πρώτα εφαρμόζεται ο απλός ΒΑ και στο τρίτο εφαρμόζονται οι δύο βελτιωμένες μορφές που παρουσιάσαμε αναλυτικότερα στο κεφάλαιο 3.4. Τα αποτελέσματα που προκύπτουν συγκρίνονται με τα αποτελέσματα άλλων μεταερευτικών μεθόδων που έχουν χρησιμοποιηθεί. Οι προσομοιώσεις εκτελέστηκαν στο προγραμματιστικό περιβάλλον MATLAB 8 (R2012b), σε υπολογιστή με επεξεργαστή AMD Phenom II, 3GHz.

5.1. Σύστημα παραγωγής Ι

Το πρώτο σύστημα παραγωγής ενέργειας αποτελείται από 6 γεννήτριες και εξετάζεται για 7 διαφορετικές τιμές ζήτησης δικτύου οι οποίες είναι 500, 600, 700, 800, 900, 1000 και 1100 MW. Το κόστος καυσίμων και η εκπομπή ρύπων υπολογίζονται από τις δυνωμικές εξισώσεις (4.1) και (4.6) αντίστοιχα. Αντικειμενική συνάρτηση του προβλήματος είναι η (4.21) με παράγοντα ποινής τον max-max της σχέσης (4.23). Συνεπώς το πρόβλημα CEED περιγράφεται από τη σχέση:

$$\min G = \sum_{i=1}^n [a_i P_i^2 + b_i P_i + c_i + h_i (\alpha_i P_i^2 + \beta_i P_i + \gamma_i)] \quad (5.1)$$

υποκείμενη στους περιορισμούς (4.3) και (4.5) με G να είναι το συνολικό κόστος του συστήματος (σε \$/hr) και ο h_i (σε \$/kg) για κάθε γεννητρία i να υπολογίζεται από την σχέση:

$$h_i = \frac{a_i P_{i,\max}^2 + b_i P_{i,\max} + c_i}{\alpha_i P_{i,\max}^2 + \beta_i P_{i,\max} + \gamma_i} \quad (5.2)$$

Οι απώλειες είναι αμελητέες δηλαδή $P_L = 0$. Άρα ο περιορισμός ισότητας διαμορφώνεται ως εξής:

$$\sum_{i=1}^n P_i - P_D = 0 \quad (5.3)$$

Τα δεδομένα του συστήματος έχουν ληφθεί από τον Güneş (2010). Οι συντελεστές κόστους καυσίμων και το όρια παραγωγής για κάθε μονάδα παραγωγής του συστήματος δίνονται στον Πίνακα 5.1 και οι συντελεστές εκπομπής ρύπων στον Πίνακα 5.2.

Εικόνα 5.1. Συντελεστές κόστους καυσίμων και όρια παραγωγής των 6 γεννητριών.

Generating unit	a_i (\$/MW ² hr)	b_i (\$/MWhr)	c_i (\$/hr)	$P_{i,\min}$ (MW)	$P_{i,\max}$ (MW)
1	0.1525	38.540	756.800	10	125
2	0.1060	46.160	451.325	10	150
3	0.0280	40.400	1050.000	35	225
4	0.0355	38.310	1243.530	35	210
5	0.0211	36.328	1658.570	130	325
6	0.0180	38.270	1356.660	125	315

Πίνακας 5.2. Συντελεστές εκπομπής ρύπων των 6 γεννητριών.

Generating unit	α_i (kg/MW ² hr)	β_i (kg/MW hr)	γ_i (kg/hr)
1	0.0042	0.3300	13.860
2	0.0042	0.3300	13.860
3	0.0068	-0.5455	40.267
4	0.0068	-0.5455	40.267
5	0.0046	-0.5112	42.900
6	0.0046	-0.5112	42.900

Για την επίλυση του προβλήματος CEED χρησιμοποιείται η βασική μορφή του BA. Οι τιμές των παραμέτρων του αλγορίθμου δίνονται στον Πίνακα 5.3. Οι τιμές των παραμέτρων A και r είναι διαφορετικές για κάθε νυχτερίδα και ανήκουν στα διαστήματα (1,2) και (0,1) αντίστοιχα.

Πίνακας 5.3. Τιμές των παραμέτρων του BA για το σύστημα παραγωγής 6 γεννητριών.

Παράμετρος	Τιμή
Μέγεθος πληθυσμού	15
Μέγιστος αριθμός επαναλήψεων	500
α	0.9
γ	0.01
Ελάχιστη συχνότητα	0
Μέγιστη συχνότητα	2

Τα βήματα του BA για το πρόβλημα CEED στο σύστημα των 6 γεννητριών είναι τα ακόλουθα:

1. Αρχικοποίηση των ταχυτήτων v_i , των συχνοτήτων Q_i , των εντάσεων ήχου A_i και των ρυθμών εκπομπής παλμών r_i για κάθε νυχτερίδα i με $i = 1, 2, \dots, n$. Αρχικοποίηση των θέσεων / λύσεων x_i για κάθε νυχτερίδα σύμφωνα με όρια παραγωγής της κάθε μονάδας. Προφανώς, κάθε θέση αναπαριστά ένα διάνυσμα με τις τιμές παραγωγής ενέργειας των γεννητριών.
2. Στην τρέχουσα επανάληψη t , για κάθε λύση x_i , παραγωγή νέων λύσεων σύμφωνα με τις εξισώσεις:

$$Q_i^t = Q_{\min} + (Q_{\max} - Q_{\min}) \cdot \text{rand}(0,1) \quad (5.4)$$

$$v_i^t = v_i^{t-1} + (x_i^{t-1} - x_{\text{best}}) \cdot Q_i^t \quad (5.5)$$

$$x_{\text{new}} = x_i^{t-1} + v_i^t \quad (5.6)$$

3. Αν $\text{rand}(0,1) > r_i$

$$x_{\text{new}} = x_{\text{best}} + \text{rand}n \quad (5.7)$$

όπου $randn$ ένα διάνυσμα d τυχαίων αριθμών που προέρχονται από την τυποποιημένη κανονική κατανομή. Εδώ έχουμε $d = 6$.

4. Έλεγχος των περιορισμών ισοτήτας και ανισότητας και αξιολόγηση της λύσης x_{new} .

5. Αν $(f(x_{new}) \leq f(x_i^{t-1}) \& rand(0,1) < A_i)$

αποδοχή των νέων λύσεων δηλαδή $x_i^t \leftarrow x_{new}$

αύξηση του r_i δηλαδή $r_i^{t+1} \leftarrow r_i^t [1 - \exp(-0.01t)]$

μείωση του A_i δηλαδή $A_i^{t+1} \leftarrow 0.9A_i^t$

6. Αξιολόγηση των θέσεων των νυχτερίδων και εύρεση της καλύτερης λύσης του πληθυσμού x_{best} .

7. Επανάληψη της διαδικασίας από το βήμα 2 έως ότου συμπληρωθεί ο μέγιστος αριθμός επαναλήψεων.

Τα αποτελέσματα που προκύπτουν φαίνονται στον Πίνακα 5.4. Στους Πίνακες 5.5 και 5.6 συγκρίνονται οι τιμές του κόστους καυσίμων και εκπομπής ρύπων του BA με τα αποτελέσματα άλλων 6 αλγορίθμων που δίνονται από τον Gύνενς (2010).

Για όλες τις τιμές ζήτησης ενέργειας ο BA δίνει χαμηλότερες τιμές και στις δύο αντικειμενικές συναρτήσεις (κόστους καυσίμων και εκπομπής ρύπων) από τους αλγορίθμους γ -iteration, Recursive, PSO, DE και Simplified recursive. Σε σύγκριση με το γενετικό αλγόριθμο που προτείνει ο Gύνενς (2010), για ζήτηση ενέργειας ίση με 600 και 1100 MW ο αλγόριθμος νυχτερίδας δίνει χαμηλότερες τιμές και στις δύο αντικειμενικές συναρτήσεις. Όταν η ζήτηση είναι 700, 800 ή 1000 MW, ο BA δίνει χαμηλότερες τιμές κόστους καυσίμων και υψηλότερες τιμές εκπομπής ρύπων. Το αντίστροφο συμβαίνει στις περιπτώσεις ζήτησης 500 και 900 MW. Εκεί ο BA δίνει υψηλότερες τιμές οικονομικού κόστους και χαμηλότερες τιμές εκπομπής ρύπων. Παρατηρούμε ότι δεν υπάρχει περίπτωση στην οποία ο γενετικός αλγόριθμος να κυριαρχεί επί του BA. Αντίθετα ο BA υπερισχύει του γενετικού αλγορίθμου σε δύο περιπτώσεις ζήτησης του δικτύου.

Πίνακας 5.4. Λύσεις του προβλήματος CEED για το σύστημα παραγωγής 6 γεννητριών με 7 διαφορετικές τιμές ζήτησης ενέργειας.

Output	Power demand (MW)						
	500	600	700	800	900	1000	1100
Unit 1 (MW)	20.1666	31.8708	43.5821	55.2051	66.6866	78.3329	90.2634
Unit 2 (MW)	14.4411	28.3164	42.0450	55.8449	69.6366	83.6692	97.3149
Unit 3 (MW)	93.0387	108.3706	123.6064	138.8468	154.4775	169.9744	185.0326
Unit 4 (MW)	90.1316	103.8710	117.8394	131.8451	145.8501	159.6377	173.5770
Unit 5 (MW)	143.8169	166.6947	189.6915	212.3683	235.2374	257.9507	280.6470
Unit 6 (MW)	138.4051	160.8764	183.2356	205.8897	228.1118	250.4351	273.1651
Fuel cost (\$/hr)	27089.87	31626.09	36310.39	41143.89	46125.14	51260.01	56541.70
Emission (kg/hr)	261.2313	338.4188	433.6513	546.8795	678.2238	827.3304	994.4646
CPU time (sec)	0.442695	0.442210	0.436435	0.435394	0.434396	0.429029	0.434389

Πίνακας 5.5. Σύγκριση των τιμών του κόστους καυσίμων για το σύστημα παραγωγής των 6 γεννητριών

Algorithm	Power demand (MW)						
	500	600	700	800	900	1000	1100
γ-iteration	27092.5	31628.7	36314.0	41148.4	46131.8	51264.6	56546.4
Recursive	27092.5	31628.6	36313.9	41148.3	46131.8	51264.5	56546.2
PSO	27097.5	31634.9	36314.2	41160.3	46160.6	51269.6	56556.7
DE	27098.1	31629.2	36314.0	41152.6	46152.6	51264.6	56546.6
Simplified recursive	27092.5	31628.6	36313.9	41148.3	46131.8	51264.6	56546.2
SC-GA	27089.45	31626.79	36310.80	41144.47	46124.54	51262.31	56542.01
Bat Algorithm	27089.87	31626.09	36310.39	41143.89	46125.14	51260.01	56541.70

Πίνακας 5.6. Σύγκριση των τιμών της εκπομπής ρύπων για το σύστημα παραγωγής των 6 γεννητριών

Algorithm	Power demand (MW)						
	500	600	700	800	900	1000	1100
γ-iteration	261.635	338.993	434.380	547.797	679.241	828.720	996.224
Recursive	261.634	338.992	434.380	547.796	679.241	828.715	996.218
PSO	262.225	339.820	434.605	547.844	679.724	828.863	996.672
DE	261.859	339.065	434.453	547.802	679.283	828.715	996.222
Simplified recursive	261.634	338.992	434.380	547.796	679.241	828.715	996.218
SC-GA	261.3307	338.4397	433.6409	546.7831	678.2906	827.2612	994.5205
Bat Algorithm	261.2313	338.4188	433.6513	546.8795	678.2238	827.3304	994.4646

5.2. Σύστημα παραγωγής II

Το δεύτερο σύστημα παραγωγής ενέργειας αποτελείται από 11 γεννήτριες και εξετάζεται για 7 διαφορετικές τιμές ζήτησης δικτύου οι οποίες είναι 1000, 1250, 1500, 1750, 2000, 2250 και 2500 MW. Η αντικειμενική συνάρτηση είναι ίδια με αυτή του προηγούμενου συστήματος. Οι απώλειες είναι επίσης μηδενικές. Οι συντελεστές κόστους καυσίμων και το όρια παραγωγής για κάθε μονάδα παραγωγής του συστήματος δίνονται στον Πίνακα 5.7. Οι συντελεστές εκπομπής ρύπων δίνονται στον Πίνακα 5.8 (Güvenç, 2010).

Για την επίλυση του προβλήματος CEED χρησιμοποιείται η βασική μορφή του BA. Οι τιμές των παραμέτρων του αλγορίθμου για το δεύτερο σύστημα παραγωγής ενέργειας είναι ίδιες με αυτές που είδαμε στον Πίνακα 5.3. Ο αλγόριθμος εκτελείται όπως και στο προηγούμενο σύστημα αλλά εδώ οι λύσεις έχουν 11 διαστάσεις.

Τα αποτελέσματα που προκύπτουν φαίνονται στον Πίνακα 5.9. Στους Πίνακες 5.10 και 5.11 συγκρίνονται οι τιμές του κόστους καυσίμων και εκπομπής ρύπων του BA με τα αποτελέσματα άλλων 6 αλγορίθμων που δίνονται από τον Güvenç (2010).

Παρατηρούμε ότι σε κάθε περίπτωση ο Bat Algorithm δίνει χαμηλότερες τιμές και στις δύο αντικειμενικές συναρτήσεις (κόστους καυσίμων και εκπομπής ρύπων) από τους αλγορίθμους γ-iteration, Recursive, PSO, DE και Simplified recursive. Σε σχέση με το γενετικό αλγόριθμο που προτείνει ο Güvenç (2010), ο BA παρουσιάζει χαμηλότερες τιμές κόστους καυσίμων και υψηλότερες τιμές εκπομπής ρύπων για ζήτηση δικτύου ίση με 1500 και 2000 MW. Για τιμές ζήτησης 1000,

1250, 1750 και 2500 MW ο BA παρουσιάζει υψηλότερο κόστος καυσίμων και χαμηλότερη εκπομπή ρύπων. Για ζήτηση ίση με 2250 MW, ο BA παρουσιάζει χαμηλότερες τιμές από το γενετικό αλγόριθμο και στις δύο αντικειμενικές συναρτήσεις. Δηλαδή δεν υπάρχει περίπτωση στην οποία ο γενετικός αλγόριθμος να κυριαρχεί επί του BA. Αντίθετα ο BA υπερισχύει του γενετικού αλγορίθμου για μία τιμή ζήτησης του δικτύου.

Πίνακας 5.7. Συντελεστές κόστους καυσίμων και όρια παραγωγής των 11 γεννητριών

Generating unit	a_i (\$/MW ² hr)	b_i (\$/MWhr)	c_i (\$/hr)	$P_{i,min}$ (MW)	$P_{i,max}$ (MW)
1	0.00762	1.92699	387.85	20	250
2	0.00838	2.11969	441.62	20	210
3	0.00523	2.19196	422.57	20	250
4	0.00140	2.01983	552.50	60	300
5	0.00154	2.22181	557.75	20	210
6	0.00177	1.91528	562.18	60	300
7	0.00195	2.10681	568.39	20	215
8	0.00106	1.99138	682.93	100	455
9	0.00117	1.99802	741.22	100	455
10	0.00089	2.12352	617.83	110	460
11	0.00098	2.10487	674.61	110	465

Πίνακας 5.8. Συντελεστές εκπομπής ρύπων των 11 γεννητριών

Generating unit	α_i (kg/MW ² hr)	β_i (kg/MWhr)	γ_i (kg/hr)
1	0.00419	-0.67767	33.93
2	0.00461	-0.69044	24.62
3	0.00419	-0.67767	33.93
4	0.00683	-0.54551	27.14
5	0.00751	-0.40060	24.15
6	0.00683	-0.54551	27.14
7	0.00751	-0.40006	24.15
8	0.00355	-0.51116	30.45
9	0.00417	-0.56228	25.59
10	0.00355	-0.41116	30.45
11	0.00417	-0.56228	25.59

Πίνακας 5.9. Λύσεις του προβλήματος CEED για το σύστημα παραγωγής 11 γεννητριών με 7 διαφορετικές τιμές ζήτησης ενέργειας.

Output	Power demand (MW)						
	1000	1250	1500	1750	2000	2250	2500
Unit 1 (MW)	85.5712	94.4293	103.4139	112.5168	121.6655	130.2063	139.1689
Unit 2 (MW)	76.3053	82.5571	88.2812	94.6160	100.4250	106.8349	112.9796
Unit 3 (MW)	87.4725	97.3146	107.0910	116.6109	126.3312	136.2092	145.7241
Unit 4 (MW)	77.8843	102.0759	125.6952	149.5845	173.1616	197.7158	221.2563
Unit 5 (MW)	48.1905	63.2788	77.9868	93.2745	107.8951	122.4630	137.2000
Unit 6 (MW)	79.0293	101.7733	125.6055	148.6524	172.4953	195.3247	217.8049
Unit 7 (MW)	50.7031	65.3963	80.1620	95.3320	110.4565	125.6297	140.5225
Unit 8 (MW)	130.1014	165.5617	202.0057	237.0730	273.4170	310.0522	346.3838
Unit 9 (MW)	122.5129	157.9796	191.2557	226.3588	260.0998	294.4109	329.4509
Unit 10 (MW)	119.3012	159.1890	201.2850	241.6588	282.1500	322.9586	363.3025
Unit 11 (MW)	122.9284	160.4447	197.2181	234.3223	271.9030	308.1948	346.2065
Fuel cost (\$/hr)	8502.07	9108.34	9733.11	10377.59	11040.72	11723.01	12424.42
Emission (kg/hr)	205.0772	339.6104	540.4952	806.8678	1139.728	1538.194	2002.950
CPU time (sec)	0.526657	0.491473	0.516921	0.514418	0.551570	0.511232	0.511030

Πίνακας 5.10. Σύγκριση των τιμών του κόστους καυσίμων για το σύστημα παραγωγής των 11 γεννητριών

Algorithm	Power demand (MW)						
	1000	1250	1500	1750	2000	2250	2500
γ -iteration	8502.30	9108.38	9733.54	10377.78	11041.08	11723.47	12424.94
Recursive	8502.29	9108.38	9733.54	10377.77	11041.08	11723.47	12424.94
PSO	8508.24	9114.42	9737.33	10380.82	11041.09	11725.68	12428.63
DE	8505.81	9117.63	9736.22	10377.86	11041.08	11723.54	12425.06
Simplified recursive	8502.29	9108.38	9733.54	10377.77	11041.08	11723.47	12424.94
SC-GA	8501.85	9107.99	9733.22	10377.01	11040.84	11723.25	12423.77
Bat Algorithm	8502.07	9108.34	9733.11	10377.59	11040.72	11723.01	12424.42

Πίνακας 5.11. Σύγκριση των τιμών της εκπομπής ρύπων για το σύστημα παραγωγής των 11 γεννητριών

Algorithm	Power demand (MW)						
	1000	1250	1500	1750	2000	2250	2500
γ-iteration	205.205	339.870	540.545	807.220	1139.912	1538.600	2003.301
Recursive	205.204	339.870	540.544	807.220	1139.911	1538.600	2003.300
PSO	208.012	345.669	545.347	812.263	1142.182	1540.465	2003.720
DE	205.206	339.935	544.298	807.236	1139.911	1538.659	2003.350
Simplified recursive	205.204	339.870	540.544	807.220	1139.911	1538.600	2003.300
SC-GA	205.175	339.706	539.493	807.214	1138.279	1538.319	2003.030
Bat Algorithm	205.0772	339.6104	540.4952	806.8678	1139.728	1538.194	2002.950

5.3. Σύστημα παραγωγής III

Το τρίτο σύστημα παραγωγής ενέργειας που μελετάμε αποτελείται από 10 σταθμούς παραγωγής ενέργειας. Η συνολική ζήτηση του δικτύου είναι 2000 MW. Το κόστος καυσίμων και οι εκπομπές ρύπων υπολογίζονται από τις εξισώσεις (4.2) και (4.7) αντίστοιχα. Αντικειμενική συνάρτηση του προβλήματος είναι η (4.28). Συνεπώς το πρόβλημα CEED περιγράφεται από τη σχέση:

$$\min G = \sum_{i=1}^n \left[w_1 \left(a_i P_i^2 + b_i P_i + c_i + \left| d_i \sin \left(e_i (P_{i,\min} - P_i) \right) \right| \right) + w_2 \left(\alpha_i P_i^2 + \beta_i P_i + \gamma_i + \eta_i \exp(\delta_i P_i) \right) \right] \quad (5.8)$$

υποκείμενη στους περιορισμούς (4.3) και (4.5). Οι τιμές των w_1 και w_2 καθορίζονται με τη βοήθεια των ασαφών συνόλων που είδαμε στην ενότητα 4.6.

Τα δεδομένα του συστήματος έχουν ληφθεί από τους Manteaw και Odero (2012). Ακολουθούν οι συντελεστές κόστους καυσίμου με τα όρια παραγωγής (Πίνακας 5.12) και οι συντελεστές εκπομπής ρύπων (Πίνακας 5.13) για κάθε μονάδα παραγωγής.

Πίνακας 5.12. Συντελεστές κόστους καυσίμων και όρια παραγωγής των 10 γεννητριών

Generating unit	a_i (\$/MW ² hr)	b_i (\$/MWhr)	c_i (\$/hr)	d_i (\$/hr)	e_i (rad/MW)	$P_{i,min}$ (MW)	$P_{i,max}$ (MW)
1	0.12951	40.5407	1000.403	33	0.0174	10	55
2	0.10908	39.5804	950.606	25	0.0178	20	80
3	0.12511	36.5104	900.705	32	0.0162	47	120
4	0.12111	39.5104	800.705	30	0.0168	20	130
5	0.15247	38.5390	756.799	30	0.0148	50	160
6	0.10587	46.1592	451.325	20	0.0163	70	240
7	0.03546	38.3055	1243.531	20	0.0152	60	300
8	0.02803	40.3965	1049.998	30	0.0128	70	340
9	0.02111	36.3278	1658.569	60	0.0136	135	470
10	0.01799	38.2704	1356.659	40	0.0141	150	470

Πίνακας 5.13. Συντελεστές εκπομπής ρύπων των 10 γεννητριών

Generating unit	α_i (lb/MW ² hr)	β_i (lb/MWhr)	γ_i (lb/hr)	η_i (lb/hr)	δ_i (1/MW)
1	0.04702	-3.9864	360.0012	0.25475	0.01234
2	0.04652	-3.9524	350.0012	0.25475	0.01234
3	0.04652	-3.9023	330.0056	0.25163	0.01215
4	0.04652	-3.9023	330.0056	0.25163	0.01215
5	0.00420	0.3277	13.8593	0.24970	0.01200
6	0.00420	0.3277	13.8593	0.24970	0.01200
7	0.00680	-0.5455	40.2699	0.24800	0.01290
8	0.00680	-0.5455	40.2699	0.24990	0.01203
9	0.00460	-0.5112	42.8955	0.25470	0.01234
10	0.00460	-0.5112	42.8955	0.25470	0.01234

Η μήτρα των συντελεστών απώλειας είναι

$$B = 10^{-5} \times \begin{bmatrix} 4.9 & 1.4 & 1.5 & 1.5 & 1.6 & 1.7 & 1.7 & 1.8 & 1.9 & 2 \\ 1.4 & 4.5 & 1.6 & 1.6 & 1.7 & 1.5 & 1.5 & 1.6 & 1.8 & 1.8 \\ 1.5 & 1.6 & 3.9 & 1 & 1.2 & 1.2 & 1.4 & 1.4 & 1.6 & 1.6 \\ 1.5 & 1.6 & 1 & 4 & 1.4 & 1 & 1.1 & 1.2 & 1.4 & 1.5 \\ 1.6 & 1.7 & 1.2 & 1.4 & 3.5 & 1.1 & 1.3 & 1.3 & 1.5 & 1.6 \\ 1.7 & 1.5 & 1.2 & 1 & 1.1 & 3.6 & 1.2 & 1.2 & 1.4 & 1.5 \\ 1.7 & 1.5 & 1.4 & 1.1 & 1.3 & 1.2 & 3.8 & 1.6 & 1.6 & 1.8 \\ 1.8 & 1.6 & 1.4 & 1.2 & 1.3 & 1.2 & 1.6 & 4 & 1.5 & 1.6 \\ 1.9 & 1.8 & 1.6 & 1.4 & 1.5 & 1.4 & 1.6 & 1.5 & 4.2 & 1.9 \\ 2 & 1.8 & 1.6 & 1.5 & 1.6 & 1.5 & 1.8 & 1.6 & 1.9 & 4.4 \end{bmatrix}$$

Στο τρίτο σύστημα παραγωγής ενέργειας χρησιμοποιούνται οι HBA και MBA που είδαμε στην ενότητα 3.4 και αναζητούνται λύσεις στα προβλήματα ELD, ED και CEED. Οι τιμές των παραμέτρων του HBA δίνονται στον Πίνακα 5.14.

Πίνακας 5.14. Τιμές των παραμέτρων του HBA για το σύστημα παραγωγής 10 γεννητριών

Παράμετρος	Τιμή
Μέγεθος πληθυσμού	15
Μέγιστος αριθμός επαναλήψεων	500
Ένταση του ήχου (A)	0.95
Ρυθμός εκπομπής παλμών (r)	0.5
Ελάχιστη συχνότητα	0
Μέγιστη συχνότητα	2
Διαφορικός συντελεστής (F)	0.5
Σταθερά διασταύρωσης (CR)	0.8

Παρατηρούμε ότι οι παράμετροι A και r είναι σταθερές. Επιπλέον θέτουμε όρια στις τιμές των ταχυτήτων. Για κάθε διάσταση του διανύσματος των ταχυτήτων έχουμε:

$$v_{i,\min} = -0.1 \cdot P_{i,\min} \quad (5.9)$$

και

$$v_{i,\max} = 0.1 \cdot P_{i,\max} \quad (5.10)$$

Τα βήματα που ακολουθεί ο HBA είναι τα εξής:

1. Αρχικοποίηση των ταχυτήτων v_i και των συχνοτήτων Q_i για κάθε νυχτερίδα i με $i = 1, 2, \dots, n$. Αρχικοποίηση των θέσεων / λύσεων x_i για κάθε νυχτερίδα σύμφωνα με όρια παραγωγής της κάθε μονάδας. Κάθε θέση είναι ένα διάνυσμα με τις τιμές παραγωγής ενέργειας των γεννητριών.
2. Στην τρέχουσα επανάληψη t , για κάθε λύση x_i , παραγωγή νέων σύμφωνα με τις εξισώσεις:

$$Q_i^t = Q_{\min} + (Q_{\max} - Q_{\min}) \cdot rand(0,1) \quad (5.11)$$

$$v_i^t = v_i^{t-1} + (x_{\text{best}} - x_i^{t-1}) \cdot Q_i^t \quad (5.12)$$

$$x_{\text{new}} = x_i^{t-1} + v_i^t \quad (5.13)$$

3. Αν $rand(0,1) > r$

Εφαρμογή των τελεστών μετάλλαξης, διασταύρωσης και επιλογής του αλγορίθμου DE για την παραγωγή νέας λύσης x_{new} .

4. Έλεγχος των περιορισμών ισοτήτας και ανισότητας και αξιολόγηση της λύσης x_{new} .
5. Αν $(f(x_{new}) \leq f(x_i^{t-1}) \& rand(0,1) < A)$
αποδοχή των νέων λύσεων δηλαδή $x_i^t \leftarrow x_{new}$
6. Αξιολόγηση των θέσεων των νυχτερίδων και εύρεση της καλύτερης λύσης του πληθυσμού x_{best} .
7. Επανάληψη της διαδικασίας από το βήμα 2 έως ότου συμπληρωθεί ο μέγιστος αριθμός επαναλήψεων.

Οι τιμές των παραμέτρων του MBA δίνονται στον Πίνακα 5.15.

Πίνακας 5.15. Τιμές των παραμέτρων του MBA για το σύστημα παραγωγής 10 γεννητριών

Παράμετρος	Τιμή
Μέγεθος πληθυσμού	15
Μέγιστος αριθμός επαναλήψεων	500
α	0.9
γ	0.01
Ελάχιστη συχνότητα	0
Μέγιστη συχνότητα	2

Τα βήματα που ακολουθεί ο MBA είναι τα εξής:

1. Αρχικοποίηση των ταχυτήτων v_i , των συχνοτήτων Q_i , των εντάσεων ήχου A_i και των ρυθμών εκπομπής παλμών r_i για κάθε νυχτερίδα i με $i = 1, 2, \dots, n$. Αρχικοποίηση των θέσεων / λύσεων x_i για κάθε νυχτερίδα σύμφωνα με όρια παραγωγής της κάθε μονάδας. Κάθε θέση αναπαριστά ένα διάνυσμα με τις τιμές παραγωγής ενέργειας των γεννητριών.
2. Στην τρέχουσα επανάληψη t , για κάθε λύση x_i , παραγωγή νέων σύμφωνα με τις εξισώσεις:

$$Q_i^t = Q_{\min} + (Q_{\max} - Q_{\min}) \cdot rand(0,1) \quad (5.14)$$

$$v_i^t = v_i^{t-1} + (x_i^{t-1} - x_{best}) \cdot Q_i^t \quad (5.15)$$

$$x_{new} = x_i^{t-1} + v_i^t \quad (5.16)$$

3. Έλεγχος των περιορισμών ισοτήτας και ανισότητας και αξιολόγηση της λύσης x_{new}

4. Αν $rand(0,1) > r_i$

$$x_{local} = x_{best} + 0.1 \cdot randn \quad (5.17)$$

όπου $randn$ ένα διάνυσμα d τυχαίων αριθμών που προέρχονται από την τυποποιημένη κανονική κατανομή. Εδώ έχουμε $d = 10$.

5. Έλεγχος των περιορισμών ισοτήτας και ανισότητας και αξιολόγηση της λύσης x_{local}

$$\text{Αν } f(x_{local}) < f(x_i^t)$$

$$x_{new} = x_{local} \quad (5.18)$$

6. Επιλογή 4 τυχαίων διανυσμάτων $x_{r1}, x_{r2}, x_{r3}, x_{r4}$ διάφορων μεταξύ τους και διάφορων του x_i και παραγωγή νέας λύσης με την εφαρμογή ενός τελεστή μετάλλαξης ως εξής:

$$x_m = x_{r1} + rand(0,1) \odot (x_{r2} - x_{r3}) + rand(0,1) \odot (x_{best} - x_{r4}) \quad (5.19)$$

7. Έλεγχος των περιορισμών ισοτήτας και ανισότητας και αξιολόγηση της λύσης x_m

$$\text{Αν } f(x_m) < f(x_{new})$$

$$x_{new} = x_m \quad (5.20)$$

8. Αν $(f(x_{new}) \leq f(x_i^{t-1}) \& rand(0,1) < A_i)$

αποδοχή των νέων λύσεων δηλαδή $x_i^t \leftarrow x_{new}$

αύξηση του r_i δηλαδή $r_i^{t+1} \leftarrow r_i^t [1 - \exp(-0.01t)]$

μείωση του A_i δηλαδή $A_i^{t+1} \leftarrow 0.9A_i^t$

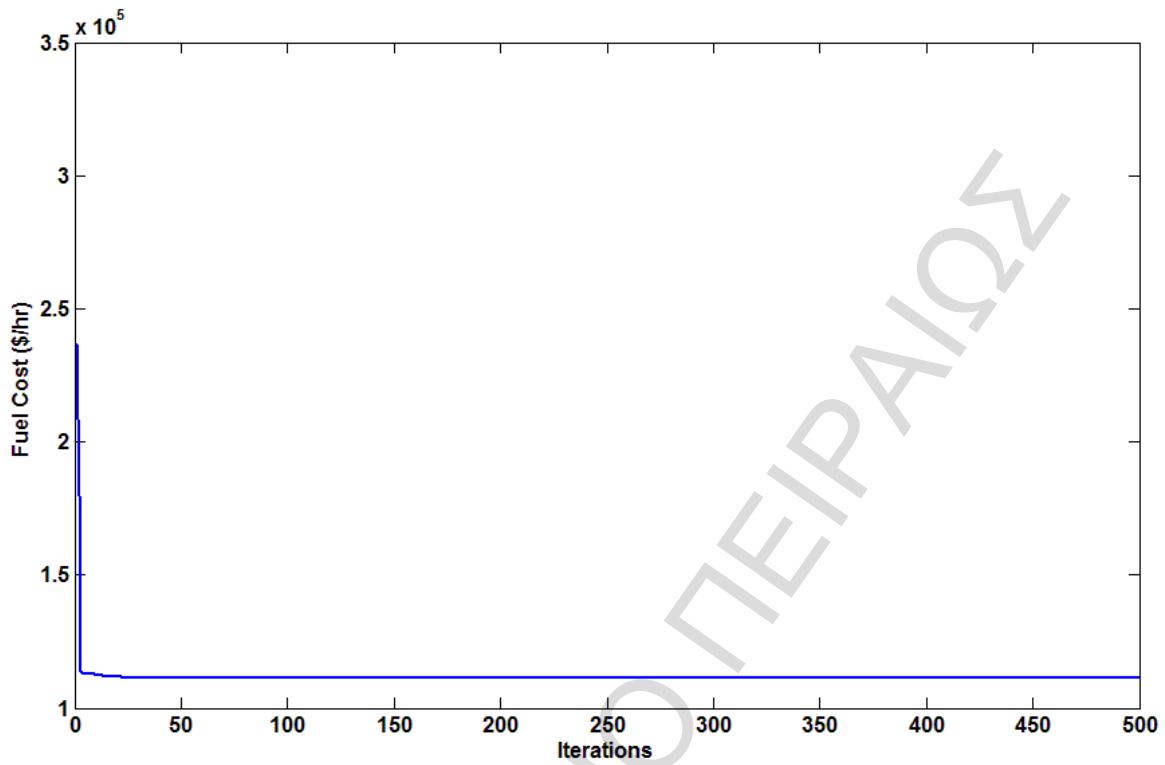
9. Αξιολόγηση των θέσεων των νυχτερίδων και εύρεση της καλύτερης λύσης του πληθυσμού x_{best} .
10. Επανάληψη της διαδικασίας από το βήμα 2 έως ότου συμπληρωθεί ο μέγιστος αριθμός επαναλήψεων.

Τα αποτελέσματα για την οικονομική κατανομή (ELD) φαίνονται στον Πίνακα 5.16 και συγκρίνονται με αυτά του υβριδικού αλγορίθμου ABC_PSO και του DE που παρουσιάζονται από τους Manteaw και Odego (2012). Ακολουθούν οι γραφικές παραστάσεις της σύγκλισης των αλγορίθμων HBA και MBA.

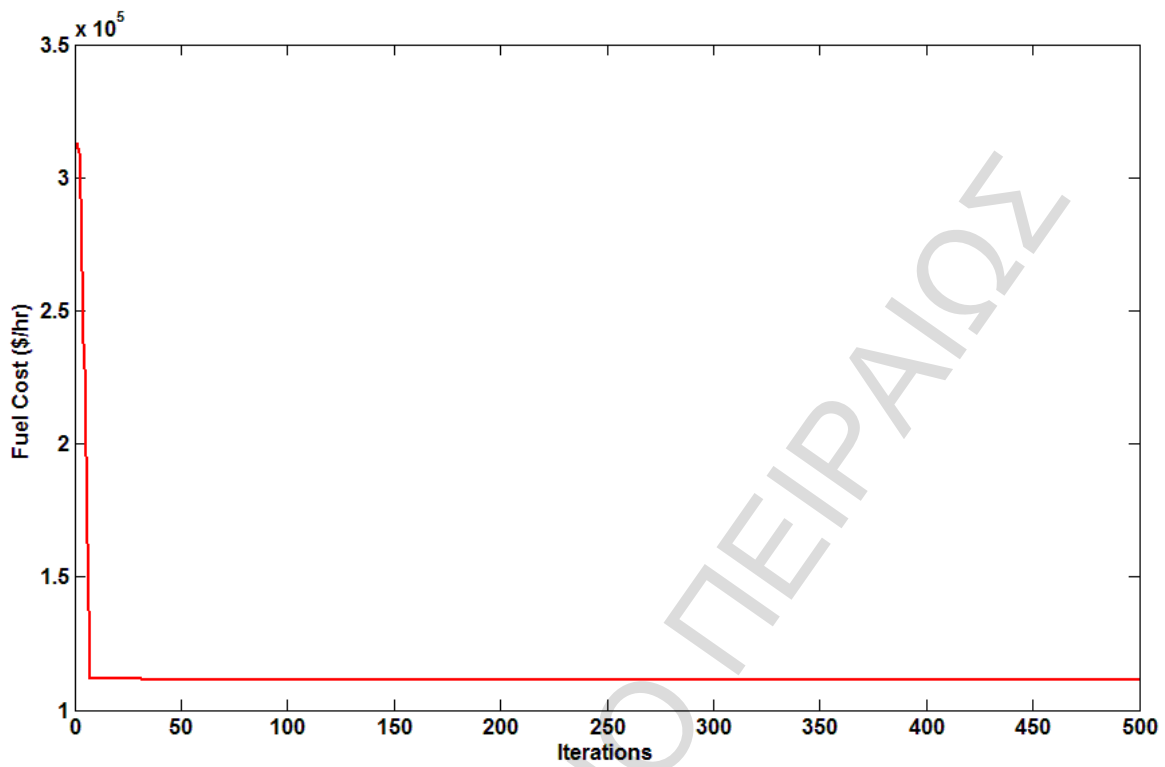
Πίνακας 5.16. Σύγκριση των αποτελεσμάτων της οικονομικής κατανομής φορτίου (ELD) για το σύστημα παραγωγής των 10 γεννητριών με $P_D = 2000$ MW

Output	Algorithm			
	HBA	MBA	ABC_PSO	DE
Unit 1 (MW)	55	55	55	55
Unit 2 (MW)	80	80	80	79.89
Unit 3 (MW)	106.6250	106.0842	106.93	106.8253
Unit 4 (MW)	99.2860	99.4944	100.5668	102.8307
Unit 5 (MW)	82.1004	82.4612	81.49	82.2418
Unit 6 (MW)	84.0278	83.9977	83.011	80.4352
Unit 7 (MW)	300	300	300	300
Unit 8 (MW)	339.9983	340	340	340
Unit 9 (MW)	470	470	470	470
Unit 10 (MW)	469.9999	469.9973	470	469.8975
Power losses (MW)	87.0374	87.0349	87.0344	-
Fuel cost (\$/hr)	111498	111498	111500	111500
Emission (lb/hr)	4564.97	4563.17	4571.2	4581

Ο χρόνος εκτέλεσης των HBA και MBA είναι 1.993011 και 2.076821 sec αντίστοιχα. Με τη χρήση των αλγορίθμων νυχτερίδας μειώνεται το οικονομικό κόστος καυσίμων και η εκπομπή ρύπων σε σύγκριση με τους αλγόριθμους ABC_PSO και DE. Οι δύο αλγόριθμοι νυχτερίδας δίνουν ίδιο κόστος καυσίμων. Ο MBA δίνει χαμηλότερες εκπομπές ρύπων όμως ο HBA συγκλίνει ταχύτερα. Στις Εικόνες 5.1 και 5.2 βλέπουμε τις γραφικές παραστάσεις της σύγκλισης των δύο αλγορίθμων.



Εικόνα 5.1. Σύγκλιση του HBA για την οικονομική κατανομή φορτίου



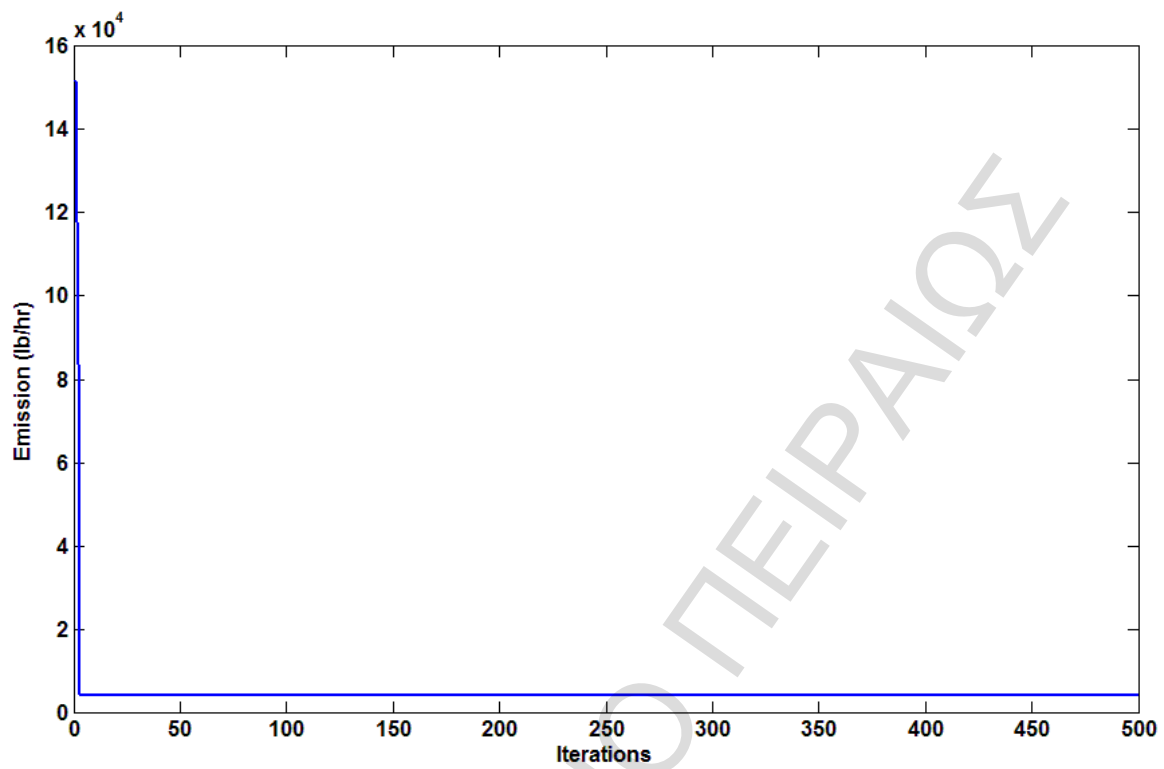
Εικόνα 5.2. Σύγκλιση του MBA για την οικονομική κατανομή φορτίου

Τα αποτελέσματα για την κατανομή εκπομπής ρύπων (ED) φαίνονται στον Πίνακα 5.17 και συγκρίνονται με αυτά του υβριδικού αλγορίθμου ABC_PSO και του DE (Manteaw και Odero, 2012). Ακολουθούν οι γραφικές παραστάσεις της σύγκλισης των αλγορίθμων HBA και MBA.

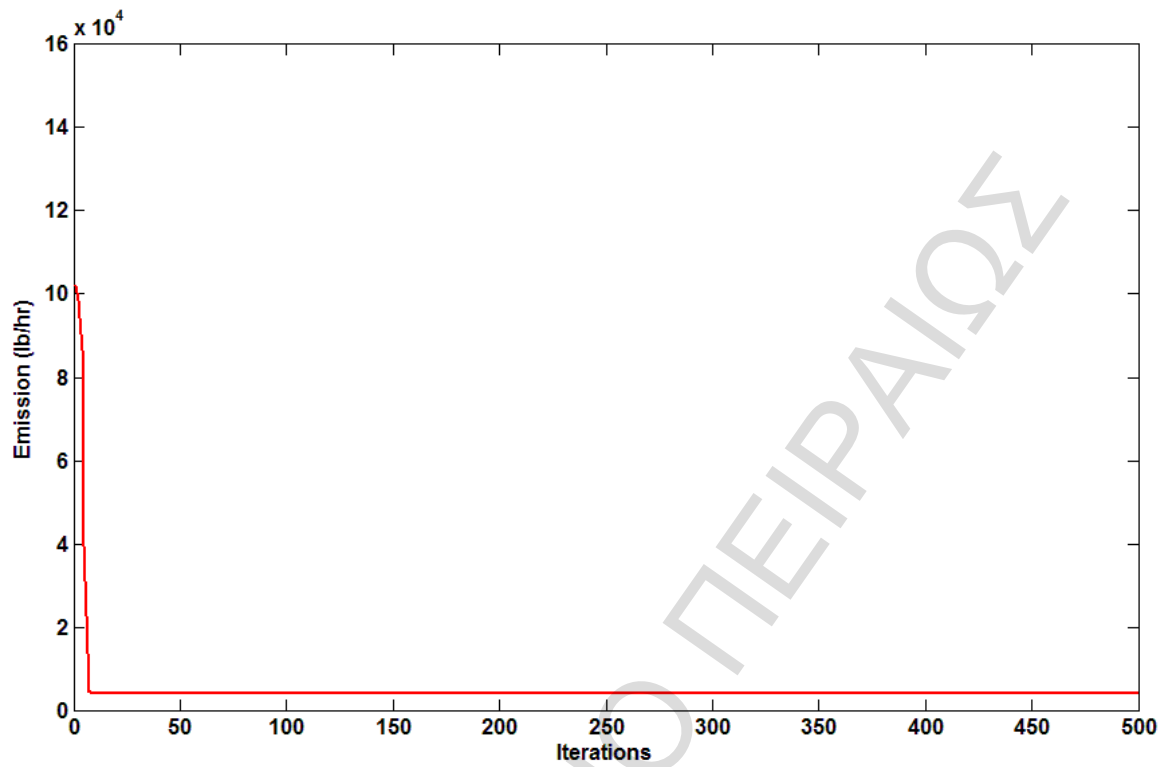
Πίνακας 5.17. Σύγκριση των αποτελεσμάτων της κατανομής εκπομπής ρύπων (ED) για το σύστημα παραγωγής των 10 γεννητριών με $P_D = 2000$ MW

Output	Algorithm			
	HBA	MBA	ABC_PSO	DE
Unit 1 (MW)	55	55	55	55
Unit 2 (MW)	80	80	80	80
Unit 3 (MW)	81.1342	81.1391	81.9604	80.5924
Unit 4 (MW)	81.3638	81.3567	78.8216	81.0233
Unit 5 (MW)	160	160	160	160
Unit 6 (MW)	240	240	240	240
Unit 7 (MW)	294.4856	294.4801	300	292.7434
Unit 8 (MW)	297.2685	297.2822	292.78	299.1214
Unit 9 (MW)	396.7662	396.7657	401.8478	394.5147
Unit 10 (MW)	395.5769	395.5713	391.2096	398.6383
Power losses (MW)	81.5952	81.5951	81.5879	-
Fuel cost (\$/hr)	116412	116412	116420	116400
Emission (lb/hr)	3932.24	3932.24	3932.3	3923.4

Ο χρόνος εκτέλεσης των HBA και MBA είναι 1.993287 και 2.249674 sec αντίστοιχα. Οι προτεινόμενοι αλγόριθμοι παρουσιάζουν χαμηλότερες εκπομπές ρύπων και χαμηλότερο κόστος καυσίμων από τον ABC_PSO. Υστερούν όμως σε σχέση με τον DE. Και πάλι μπορούμε να διακρίνουμε στις Εικόνες 5.3 και 5.4 ότι ο HBA συγκλίνει ταχύτερα από τον MBA.



Εικόνα 5.3. Σύγκλιση του HBA για την κατανομή εκπομπής ρύπων



Εικόνα 5.4. Σύγκλιση του MBA για την κατανομή εκπομπής ρύπων

Για τη συνδυασμένη κατανομή εκτελούμε τους αλγόριθμους για 5 συνδυασμούς τιμών των βαρών. Οι τιμές των βαρών και η αξιολόγηση κάθε λύσης φαίνονται στους Πίνακες 5.18 και 5.19 για τους HBA και MBA αντίστοιχα.

Πίνακας 5.18. Αξιολόγηση των βαρών για τις λύσεις Pareto του HBA

w_1	w_2	Fuel cost	Emission	R_k
0.095	0.905	113389	4117.55	0.2363
0.295	0.705	111614	4450.18	0.2073
0.495	0.505	111523	4510.11	0.1937
0.695	0.305	111503	4555.64	0.1815
0.895	0.105	111499	4559.24	0.1807

Πίνακας 5.19. Αξιολόγηση των βαρών για τις λύσεις Pareto του MBA

w_1	w_2	Fuel cost	Emission	R_k
0.0955	0.9045	113375	4119	0.2351
0.2955	0.7045	111632	4441.89	0.2072
0.4955	0.5045	111529	4502.42	0.1939
0.6955	0.3045	111504	4538.7	0.1845
0.8955	0.1045	111499	4558.1	0.1793

Άρα για τον αλγόριθμο HBA επιλέγεται η λύση με $w_1 = 0.095$ και $w_2 = 0.905$ ενώ για τον MBA επιλέγεται η λύση με $w_1 = 0.0955$ και $w_2 = 0.9045$. Στον Πίνακα 5.20 συγκρίνονται οι λύσεις των αλγορίθμων HBA, MBA, ABC_PSO, Multi-objective Differential Evolution (MODE), Nondominated Sorting Genetic Algorithm – II (NSGA-II) και Strength Pareto Evolutionary Algorithm – 2 (SPEA-2) για το πρόβλημα CEED.

Πίνακας 5.20. Σύγκριση των αποτελεσμάτων της συνδυασμένης κατανομής (CEED) για το σύστημα παραγωγής των 10 γεννητριών με $P_D = 2000$ MW

Output	Algorithm					
	HBA	MBA	ABC_PSO	MODE	NSGA-II	SPEA-2
Unit 1 (MW)	55	55	55	54.9487	51.9515	52.9761
Unit 2 (MW)	80	80	80	74.5821	67.2584	72.813
Unit 3 (MW)	85.0157	85.0378	81.14	79.4294	73.6879	78.1128
Unit 4 (MW)	83.6358	83.6548	84.216	80.6875	91.3554	83.6088
Unit 5 (MW)	141.5894	141.3312	138.3377	136.8551	134.0522	137.2432
Unit 6 (MW)	161.6944	161.3887	167.5086	172.6393	174.9504	172.9188
Unit 7 (MW)	300	299.9998	296.8338	283.8233	289.435	287.2023
Unit 8 (MW)	315.4369	315.4383	311.5824	316.3407	314.0556	326.4023
Unit 9 (MW)	429.7206	429.9759	420.3363	448.5923	455.6978	448.8814
Unit 10 (MW)	431.8299	432.1132	449.1598	436.4287	431.8054	423.9025
Power losses (MW)	83.9228	83.9395	84.1736	-	-	-
Fuel cost (\$/hr)	113389	113375	113420	113480	113540	113520
Emission (lb/hr)	4117.55	4119	4120.1	4124.9	4130.2	4109.1

Ο χρόνος εκτέλεσης των HBA και MBA είναι 2.300691 και 2.933911 sec αντίστοιχα. Οι αλγόριθμοι HBA και MBA δίνουν χαμηλότερες τιμές κόστους και εκπομπής ρύπων από τους αλγόριθμους ABC_PSO, MODE και NSGA-II. Σε σύγκριση με τον αλγόριθμο SPEA-2 το κόστος καυσίμων είναι μικρότερο και η εκπομπή ρύπων μεγαλύτερη.

Συγκρίνοντας τους HBA και MBA παρατηρούμε ότι ο HBA συγκλίνει ταχύτερα, γεγονός που αποδίδεται στην ισχυρότερη τοπική αναζήτηση που πραγματοποιείται στον HBA. Αντίθετα, στον MBA με την εισαγωγή του τελεστή μετάλλαξης γίνεται μεγαλύτερη εξερεύνηση του χώρου λύσεων και στην περίπτωση του ELD η λύση του MBA δίνει χαμηλότερη εκπομπή ρύπων. Επίσης, ο HBA απαιτεί μικρότερο χρόνο εκτέλεσης από τον MBA. Ένα μειονέκτημα του HBA είναι η ρύθμιση πολλών παραμέτρων αφού συνδυάζονται δύο αλγόριθμοι. Μάλιστα στο πρόβλημα που μελετήσαμε έπρεπε να ρυθμίσουμε όρια και για τις ταχύτητες ώστε να επιτύχουμε καλύτερες λύσεις. Στον MBA οι παράμετροι είναι σαφώς λιγότεροι.

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

6. ΣΥΝΟΨΗ – ΣΥΜΠΕΡΑΣΜΑΤΑ

Στην παρούσα μεταπτυχιακή διατριβή αναφέραμε τις βασικές αρχές των προβλημάτων βελτιστοποίησης και των μεταερευτικών αλγορίθμων. Περιγράψαμε συνοπτικά μεταερευτικούς αλγορίθμους που έχουν δώσει επιτυχημένα αποτελέσματα σε πολλά προβλήματα, ενώ κάναμε μια αναλυτική περιγραφή του αλγορίθμου της νυχτερίδας, ενός από τους νεότερους μεταερευτικούς αλγορίθμους που ακολουθεί τη νοημοσύνη σμήνους. Επίσης αναλύσαμε το πρόβλημα της συνδυασμένης κατανομής οικονομικού κόστους και εκπομπής ρύπων, ένα πολυκριτηριακό πρόβλημα βελτιστοποίησης με περιορισμούς και συναρτήσεις που ανταγωνίζονται μεταξύ τους.

Ο αλγόριθμος της νυχτερίδας υλοποιήθηκε προκειμένου να βρεθούν λύσεις του προβλήματος σε τρία διαφορετικά συστήματα παραγωγής ενέργειας. Στα δύο πρώτα χρησιμοποιήθηκε ο απλός BA για 7 τιμές ζήτησης ενέργειας και τα αποτελέσματά του συγκρίθηκαν με αυτά που είχαν προκύψει από άλλους 5 αλγορίθμους. Στην πλειοψηφία των περιπτώσεων, με τη χρήση του BA προέκυψαν καλύτερες τιμές και για τα δύο μεγέθη δηλαδή για το οικονομικό κόστος και για την εκπομπή ρύπων. Σε πολύ λίγες περιπτώσεις ο BA υστερούσε στο ένα μέγεθος αλλά υπερτερούσε στο άλλο. Ποτέ όμως δεν παρουσίασε χειρότερες τιμές και για τα δύο μεγέθη.

Για το τρίτο σύστημα παραγωγής χρησιμοποιήθηκαν δύο προηγμένες μορφές του αλγορίθμου, ο HBA και ο MBA. Τα αποτελέσματα συγκρίθηκαν με αυτά που είχαν ληφθεί από άλλους 4 αλγορίθμους. Και οι 2 αλγόριθμοι δίνουν καλύτερες τιμές για τα δύο μεγέθη, με εξαίρεση μόλις μία περίπτωση όπου προκύπτει χαμηλότερο κόστος και μεγαλύτερη εκπομπή ρύπων.

Από τα παραπάνω συμπεραίνουμε αρχικά ότι και οι 3 μορφές του BA είναι ιδιαίτερα αποδοτικές αφού στη συντριπτική πλειοψηφία των περιπτώσεων, οι λύσεις που παράγουν δίνουν χαμηλότερες τιμές και στις δύο αντικειμενικές συναρτήσεις του προβλήματος σε σχέση με τους άλλους αλγορίθμους. Σε λίγες περιπτώσεις η μία αντικειμενική συνάρτηση έχει μικρότερη τιμή και η άλλη μεγαλύτερη αλλά κανένας από τους άλλους αλγόριθμους δε δίνει χαμηλότερη τιμή και στις δύο αντικειμενικές συναρτήσεις. Εκτός από αποδοτικός, ο BA είναι και εύρωστος. Το δεύτερο σύστημα παραγωγής έχει περισσότερες γεννήτριες από το πρώτο, γεγονός που δεν επηρέασε σημαντικά τον αλγόριθμο ο οποίος συνέχισε να παρουσιάζει αποδοτικές λύσεις. Στο τρίτο σύστημα μελετήσαμε τη σύγκλιση των αλγορίθμων στα προβλήματα ELD και ED και είδαμε ότι όχι μόνο βρίσκουν πολύ καλές λύσεις αλλά φτάνουν πολύ γρήγορα σε αυτές. Για όλες τις προσομοιώσεις υπολογίσαμε και το χρόνο εκτέλεσης του κάθε αλγορίθμου και τα αποτελέσματά μας έδειξαν ότι δεν απαιτείται μεγάλος υπολογιστικός χρόνος για την παραγωγή των λύσεων. Όπως ήταν αναμενόμενο, ο χρόνος εκτέλεσης αυξάνεται όταν αυξάνεται η πολυπλοκότητα του αλγορίθμου ή του προβλήματος βελτιστοποίησης.

Συμπεραίνουμε ότι ο αλγόριθμος της νυχτερίδας εμφανίζει ιδιότητες που τον καθιστούν μία ιδιαίτερα ανταγωνιστική μεταερευτική μέθοδο. Αν και έχει παρέλθει μικρό χρονικό διάστημα από την παρουσίασή του, δεν μπορεί να θεωρηθεί απλώς μία ελπιδοφόρα μεταερευτική μέθοδος αφού είναι αποδοτικότερος από άλλες επιτυχημένες τεχνικές βελτιστοποίησης. Πέρα από τα πλεονεκτήματά του παρουσιάζει ορισμένες αδυναμίες, με κυριότερες τον κίνδυνο σύγκλισης σε τοπικά βέλτιστα και την ανάγκη ρύθμισης πολλών παραμέτρων. Όμως η φύση του παρέχει πολλές δυνατότητες βελτίωσης ή και υβριδοποίησής του με άλλους αλγορίθμους. Άλλωστε, βελτιωμένες μορφές του αλγορίθμου συνεχίζουν να παρουσιάζονται και μάλιστα με ταχύτατο ρυθμό.

Παρόλα αυτά, υπάρχουν θέματα τα οποία χρήζουν περαιτέρω έρευνας. Το πιο σημαντικό θέμα είναι η ρύθμιση και ο συντονισμός των παραμέτρων του αλγορίθμου. Οι παράμετροι σε έναν αλγόριθμο καθορίζονται κυρίως από το κάθε πρόβλημα αλλά συνήθως μετά από μελέτες διαμορφώνεται ένας προτεινόμενος τρόπος καθορισμού των παραμέτρων. Στον αλγόριθμο της νυχτερίδας δεν έχει βρεθεί κάποια γενικότερη στρατηγική για τον καθορισμό των παραμέτρων ελέγχου. Ακόμα, δεν έχουν προταθεί κάποια διαστήματα τιμών για τις υπόλοιπες παραμέτρους που θα μπορούσαν μάλιστα να συντονιστούν μεταξύ τους ώστε να αυξηθεί η απόδοση του αλγορίθμου. Άρα ο αλγόριθμος της νυχτερίδας αν και είναι μία πολύ ισχυρή μεταερευτική μέθοδος με πολλές εφαρμογές, μπορεί στο μέλλον με τις κατάλληλες μελέτες να γίνει ακόμα ισχυρότερος και με ένα ευρύτερο πεδίο εφαρμογών.

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

ΠΑΡΑΡΤΗΜΑ

Π.1. Κατάλογος ακρωνύμιων

ABC – Artificial Bee Colony
ACO – Ant Colony Optimization
AS – Ant System
BA – Bat Algorithm
BFO – Bacteria Foraging Optimization
CEED – Combined Economic and Emission Dispatch
CS – Cuckoo Search
DE – Differential Evolution
DHS – Differential Harmony Search
ED – Emission Dispatch
ELD – Economic Load Dispatch
ES – Evolutionary Strategies
FA – Firefly Algorithm
FIPS – Fully Informed Particle Swarm
FSA – Fish Swarm Algorithm
GA – Genetic Algorithm
GHS – Global-based Harmony Search
GP – Genetic Programming
GS – Gravitational Search
HBA – Hybrid Bat Algorithm
HM – Harmony Memory
HMCR – Harmony Memory Considering Rate
HMS – Harmony Memory Size
HS – Harmony Search
IBA – Improved Bat Algorithm
IHS – Improved Harmony Search
MBA – Mutated Bat Algorithm
MODE – Multi-objective Differential Evolution
NSGA-II – Nondominated Sorting Genetic Algorithm – II
PAR – Pitch Adjusting Rate
PSO – Particle Swarm Optimization
SA – Simulated Annealing
SGA – Simple Genetic Algorithm
SPEA-2 – Strength Pareto Evolutionary Algorithm – 2

Π.2. Κατάλογος εικόνων

Εικόνα 1.1. Τοπικά βέλτιστα και ολικό βέλτιστο σε πρόβλημα ελαχιστοποίησης μίας διαστάσης 8

Εικόνα 1.2. Τοπικά βέλτιστα και ολικό βέλτιστο σε πρόβλημα μεγιστοποίησης δύο διαστάσεων 9

Εφαρμογές του αλγορίθμου της νυχτερίδας σε πολυκριτηριακά προβλήματα βελτιστοποίησης

Εικόνα 2.1. Ταξινόμηση των αλγορίθμων	12
Εικόνα 2.2. Απεικόνιση της μεθόδου της ρουλέτας	14
Εικόνα 2.3. Σχηματική αναπαράσταση διασταύρωσης ενός σημείου	15
Εικόνα 2.4. Σχηματική αναπαράσταση διασταύρωσης τριών σημείων	16
Εικόνα 2.5. Σχηματική αναπαράσταση της μετάλλαξης	16
Εικόνα 2.6. Απεικόνιση της μετάλλαξης του DE σε χώρο αναζήτησης 2 διαστάσεων	20
Εικόνα 2.7. Απεικόνιση της διασταύρωσης για $D=7$ παραμέτρους σε κάθε διάνυσμα	21
Εικόνα 2.8. Κίνηση σωματιδίου σε 2 διαστάσεις	29
Εικόνα 2.9. Τοπολογίες σωματιδίων	34
Εικόνα 2.10. Αναλογία μεταξύ μουσικού αυτοσχεδιασμού και βελτιστοποίησης	35
Εικόνα 2.11. Τυπικό μοντέλο αποικίας μελισσών	41
Εικόνα 3.1. Απεικόνιση του ηχοεντοπισμού	51
Εικόνα 4.1. Καμπύλες των συναρτήσεων του οικονομικού κόστους	60
Εικόνα 4.2. Μέτωπο Pareto για πρόβλημα βελτιστοποίησης δύο αντικειμενικών συναρτήσεων	64
Εικόνα 5.1. Σύγκλιση του HBA για την οικονομική κατανομή φορτίου	81
Εικόνα 5.2. Σύγκλιση του MBA για την οικονομική κατανομή φορτίου	82
Εικόνα 5.3. Σύγκλιση του HBA για την κατανομή εκπομπής ρύπων	84
Εικόνα 5.4. Σύγκλιση του MBA για την κατανομή εκπομπής ρύπων	85

Π.3. Κατάλογος πινάκων

Πίνακας 2.1. Αναλογία της ανόπτησης μετάλλου με την προσομοιωμένη ανόπτηση	24
Πίνακας 5.1. Συντελεστές κόστους καυσίμων και όρια παραγωγής των 6 γεννητριών	69
Πίνακας 5.2. Συντελεστές εκπομπής ρύπων των 6 γεννητριών	70
Πίνακας 5.3. Τιμές των παραμέτρων του BA για το σύστημα παραγωγής 6 γεννητριών	70
Πίνακας 5.4. Λύσεις του προβλήματος CEED για το σύστημα παραγωγής 6 γεννητριών με 7 διαφορετικές τιμές ζήτησης ενέργειας	71
Πίνακας 5.5. Σύγκριση των τιμών του κόστους καυσίμων για το σύστημα παραγωγής των 6 γεννητριών	72
Πίνακας 5.6. Σύγκριση των τιμών της εκπομπής ρύπων για το σύστημα παραγωγής των 6 γεννητριών	72
Πίνακας 5.7. Συντελεστές κόστους καυσίμων και όρια παραγωγής των 11 γεννητριών	73
Πίνακας 5.8. Συντελεστές εκπομπής ρύπων των 11 γεννητριών	73
Πίνακας 5.9. Λύσεις του προβλήματος CEED για το σύστημα παραγωγής 11 γεννητριών με 7 διαφορετικές τιμές ζήτησης ενέργειας	74
Πίνακας 5.10. Σύγκριση των τιμών του κόστους καυσίμων για το σύστημα παραγωγής των 11 γεννητριών	74
Πίνακας 5.11. Σύγκριση των τιμών της εκπομπής ρύπων για το σύστημα παραγωγής των 11 γεννητριών	75
Πίνακας 5.12. Συντελεστές κόστους καυσίμων και όρια παραγωγής των 10 γεννητριών	76
Πίνακας 5.13. Συντελεστές εκπομπής ρύπων των 10 γεννητριών	76
Πίνακας 5.14. Τιμές των παραμέτρων του HBA για το σύστημα παραγωγής 10 γεννητριών	77
Πίνακας 5.15. Τιμές των παραμέτρων του MBA για το σύστημα παραγωγής 10 γεννητριών	78

Πίνακας 5.16. Σύγκριση των αποτελεσμάτων της οικονομικής κατανομής φορτίου (ELD) για το σύστημα παραγωγής των 10 γεννητριών με $P_D = 2000$ MW	80
Πίνακας 5.17. Σύγκριση των αποτελεσμάτων της κατανομής εκπομπής ρύπων (ED) για το σύστημα παραγωγής των 10 γεννητριών με $P_D = 2000$ MW	83
Πίνακας 5.18. Αξιολόγηση των βαρών για τις λύσεις Pareto του HBA	86
Πίνακας 5.19. Αξιολόγηση των βαρών για τις λύσεις Pareto του MBA	86
Πίνακας 5.20. Σύγκριση των αποτελεσμάτων της συνδυασμένης κατανομής (CEED) για το σύστημα παραγωγής των 10 γεννητριών με $P_D = 2000$ MW	87

Π.4. Κατάλογος ψευδοκωδίκων

Ψευδοκώδικας 2.1. Genetic Algorithm	18
Ψευδοκώδικας 2.2. Differential Evolution	21
Ψευδοκώδικας 2.3. Simulated Annealing	27
Ψευδοκώδικας 2.4. Particle Swarm Optimization	30
Ψευδοκώδικας 2.5. Harmony Search	38
Ψευδοκώδικας 2.6. Τρίτο βήμα του αλγορίθμου Global-best Harmony Search	39
Ψευδοκώδικας 2.7. Artificial Bee Colony	43
Ψευδοκώδικας 3.1. Bat Algorithm	53
Ψευδοκώδικας 3.2. Hybrid Bat Algorithm	56
Ψευδοκώδικας 3.3. Mutated Bat Algorithm	57
Ψευδοκώδικας 4.1. Cardinal Priority Raking	68

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

ΒΙΒΛΙΟΓΡΑΦΙΑ

1. Airas, M. 2003. "Echolocation in bats", in *Proceedings of Spatial sound perception and reproduction*.
2. Alia, O. M. and Mandava, R. 2011. The variants of the harmony search algorithm: an overview. *Artificial Intelligence Review*, 36 (1), pp. 49-68.
3. Azizipanah-Abarghooee, R. and Niknam, T. 2012. "A new improved Bat Algorithm for fuzzy interactive multiobjective economic/emission dispatch with load and wind power uncertainty", in C. Kahraman, E. E. Kerre and F. T. Bozbura (eds.), *Uncertainty Modeling in Knowledge Engineering and Decision Making: Proceedings of the 10th International FLINS Conference*, World Scientific Publishing, Singapore, pp. 388-393.
4. Bansal, J. C., Singh, P.K., Saraswat, M., Verma, A., Jadon, S. S., Abraham, A. 2011. "Inertia weight strategies in particle swarm optimization", in *Proceedings 2011 Third World Congress on Nature and Biologically Inspired Computing*, pp. 640-647.
5. Basu, M. 2011. Economic environmental dispatch using multi-objective differential evolution. *Applied Soft Computing*, 11 (2), pp. 2845-2853.
6. Batcon.org. 2011. *Intro to Bats*. [online] Available at: <http://www.batcon.org/index.php/all-about-bats/intro-to-bats.html> [Accessed: 27 Sep 2013]
7. Bianchi, L., Dorigo, M., Gambardella, L. M. and Gutjahr, W. J. 2009. A survey on metaheuristics for stochastic combinatorial optimization. *Natural Computing: an international journal*, 8 (2), pp. 239-287.
8. Binitha, S. and Siva Sathya, S. 2012. A Survey of Bio inspired Optimization Algorithms. *International Journal of Soft Computing and Engineering*, 2 (2), pp. 137-151.
9. Chakraborty, P., Roy, G. G., Das, S., Jain, D. and Abraham, A. 2009. An Improved Harmony Search Algorithm with Differential Mutation Operator. *Fundamenta Informaticae - Swarm Intelligence*, 95 (4), pp. 401-426.
10. Clerc, M. 1999. "The swarm and the queen: towards a deterministic and adaptive particle swarm optimization", *Proceedings of the Congress on Evolutionary Computation, 1999. CEC 99*, pp. 1951-1957.
11. Clerc, M. and Kennedy, J. 2002. The Particle Swarm - Explosion, Stability, and Convergence in a Multidimensional Complex Space. *IEEE Transactions on Evolutionary Computation*, 6 (1), pp. 58-73.
12. Dalton, J. 2007. *Genetic Algorithms (GAs)*. [online] Available at: <http://www.edc.ncl.ac.uk/highlight/rhjanuary2007.php/> [Accessed: 18 Aug 2013]
13. Das, S. 2011. Differential Evolution: A Survey of the State-of-the-Art. *IEEE Transactions on Evolutionary Computation*, 15 (1), pp. 4-31.
14. Dhillon, J. and Jain, S. K. 2011. Multi-Objective Generation and Emission Dispatch Using NSGA-II. *International Journal of Engineering and Technology*, 3 (5), pp. 460-466.
15. Dorigo, M., Di Caro, G. and Gambardella, L. M. 1999. Ant Algorithms for Discrete Optimization. *Artificial Life*, 5 (2), pp. 137-172.
16. Dorigo, M. and Gambardella, L. M. 1997. Ant colonies for the travelling salesman problem. *Biosystems*, 43 (2), pp. 73-81.
17. Dorigo, M., Maniezzo, V. and Colorni, A. 1996. Ant system: optimization by a colony of cooperating agents. *IEEE Transactions on Systems, Man and Cybernetics, Part B: Cybernetics*, 26 (1), pp. 29-41.
18. Eberhart, R. C. and Shi, Y. 2001. Particle swarm optimization: developments, applications and resources. *Proceedings of the 2001 Congress on Evolutionary Computation (CEC 2001)*, 1, pp. 81-86.
19. Fister, I., Fister, D., Yang, X. S. 2013. A hybrid bat algorithm. *Elektrotehniški vestnik*, 80 (1-2), pp. 1-7.
20. Gao, M. L., He, X. H., Luo, D. S. and Yu, Y. M. 2012. Object tracking based on harmony search: comparative study. *Journal of Electronic Imaging*, 21 (4), pp. 043001-1 -043001-13.

21. Geem, Z. W, Kim, J. H. and Loganathan, G. V. 2001. A new heuristic optimization algorithm: Harmony search. *Simulation*, 76 (2), pp. 60-68.
22. Geem, Z. W., Tseng, C. L. and Park, Y. 2005. "Harmony Search for Generalized Orienteering Problem: Best Touring in China", in L. Wag, K. Chen and Y. S. Ong (eds.), *Advances in Natural Computation: ICNC'05 Proceedings of the First international conference on Advances in Natural Computation - Part III*, Springer Berlin-Heidelberg, Germany, pp. 741-750.
23. German, S. and German, D. 1984. Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *IEEE Transactions Pattern Analysis and Machine Intelligence*, 6 (6), pp. 721-741.
24. Güvenç, U. 2010. Combined economic emission dispatch using genetic algorithm based on similarity crossover. *Scientific Research and Essays*, 5 (17), pp. 2451-2456.
25. Hajek, B. 1988. Cooling schedules for optimal annealing. *Mathematics of operation research*, 13 (2), pp. 311-329.
26. Holland, J. H. 1975. *Adaptation in Natural and Artificial Systems*, University of Michigan Press, Ann Arbor.
27. Karaboga, D. and Basturk, B. 2007. A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm. *Journal of Global Optimization*, 39 (3), pp. 459-471.
28. Karaboga, D. and Ozturk, C. 2011. A novel clustering approach: Artificial Bee Colony (ABC) algorithm. *Applied Soft Computing*, 11 (1), pp. 652-657.
29. Kaviani, Y. S., Rashedi, A., Mahani, A. and Ghassemloooy, Z. 2013. Routing and wavelength assignment in optical networks using Artificial Bee Colony algorithm. *Optik - International Journal for Light and Electron Optics*, 124 (12), p. 1243-1249.
30. Kennedy, J. and Eberhart, R. 1995. "Particle swarm optimization". *Proceedings of the IEEE International Conference on Neural Networks*, 4, pp. 1942-1948.
31. Khan, K., Nikov, A., Sahai, A., 2011. "A fuzzy bat clustering method for ergonomic screening of office workplaces", in D. Dicheva, Z. Markov and E. Stefanova (eds.), *Advances in Intelligent and Soft Computing: Proceedings on Third International Conference on Software, Services and Semantic Technologies S3T 2011*, Springer Berlin-Heidelberg, Germany, pp. 59-66.
32. Khan, K. and Sahai, A. 2012. A Comparison of BA, GA, PSO, BP and LM for Training Feed forward Neural Networks in e-Learning Context. *International Journal of Intelligent Systems and Applications*, 4 (7), pp. 23-29.
33. Kiran, M. S. and Gündüz, M. 2012. A Novel Artificial Bee Colony - based Algorithm for Solving the Numerical Optimization Problems. *International Journal of Innovative Computing, Information and Control*, 8 (9), pp. 6107-6121.
34. Kirkpatrick, S., Gelatt, C. D. and Vecchi, M. P. 1983. Optimization by Simulated Annealing. *Science*, 220 (4598), pp. 671-680.
35. Kothari, D. P. and Dhillon, J. S. 2011. *Power system optimization*. 2nd ed. New Delhi: Prentice-Hall of India.
36. Koza, J. R. 1992. *Genetic Programming: On the Programming of Computers by Means of Natural Selection*. Cambridge, MA: The MIT Press.
37. Krishnamurthy, S. and Tzoneva, R. 2012. Multi Objective Dispatch Problem with Valve Point Effect Loading of Fuel Cost and Emission Criterion. *International Journal of Computer and Electrical Engineering*, 4 (5), pp. 775-784.
38. Krishnan, G. and Krishnan, A. 2011. Study on Techniques for Combined Economic and Emission Dispatch. *Global Journal of Researches in Engineering, Electrical and Electronic Engineering*, 11 (5), pp. 20-28.
39. Li, X. L., Shao, Z. J. and Qian, J. X. 2002. An Optimizing Method based on Autonomous Animate: Fish Swarm Algorithm. *System Engineering Theory and Practice*, 22 (11), pp. 32-38.
40. Lundy, M. and Mees, A. 1986. Convergence of an annealing algorithm. *Mathematical Programming*, 34 (1), pp. 111-124.
41. Mahale, R. and Chavan, S. 2012. A Survey: Evolutionary and Swarm Based Bio – Inspired Optimization Algorithms. *International Journal of Scientific and Research Publications*, 2 (12).

42. Mahdavi, M., Fesanghary, M. and Damangir, E. 2007. An improved harmony search algorithm for solving optimization problems. *Applied Mathematics and Computation*, 188 (2), pp. 1567-1579.
43. Manteaw, E. D. and Odero, N. A. 2012. Combined Economic and Emission Dispatch Solution Using ABC_PSO Hybrid Algorithm with Valve Point Loading Effect. *International Journal of Scientific and Research Publications*, 2 (12).
44. Mendes, R., Kennedy, J., and Neves, J. 2003. "Watch thy neighbor or how the swarm can learn from its environment", in *Proceedings of the 2003 IEEE swarm intelligence symposium (SIS '03)*, pp. 88–94.
45. Nakamura, R. Y. M., Pereira, L. A. M., Costa, K. A., Rodrigues, D., Papa, J. P. and Yang, X. S. 2012. "BBA: A binary bat algorithm for feature selection", in: *Proceedings of the 25th SIBGRAPI Conference on Graphics, Patterns and Images*, pp. 291-297.
46. Nakrani, S. and Tovey, C. 2004. On Honey Bees and Dynamic Server Allocation in Internet Hosting Centers. *Adaptive Behavior - Animals, Animats, Software Agents, Robots, Adaptive Systems*, 12 (3-4), pp. 223-240.
47. Omran, M. G. H. and Mahdavi, M. 2008. Global-best harmony search. *Applied Mathematics and Computation*, 198 (2), pp. 643–656.
48. Palanichamy, C. and Srikrishna, K. 1991. Economic Thermal Power Dispatch with Emission Constraint. *Journal of the Institution of Engineers*, 72, pp. 11-18.
49. Passino, K. M. 2002. Biomimicry of bacterial foraging for distributed optimization and control. *IEEE Control Systems*, 22 (2), pp. 52-67.
50. Poli, R., Kennedy, J. and Blackwell, T. 2007. Particle swarm optimization: An Overview. *Swarm Intelligence*, 1 (1), pp.33-57.
51. Rashedi, E., Nezamabadi-Pour, H. and Saryazdi, S. 2009. GSA: A Gravitational Search Algorithm. *Information Sciences*, 179 (13), pp. 2232–2248.
52. Ratnaweera, A., Halgamuge, S., Watson, H. C. 2004. Self-Organizing Hierarchical Particle Swarm Optimizer with Time-Varying Acceleration Coefficients. *IEEE Transactions on Evolutionary Computation*, 8 (3), pp. 240-255.
53. Rönkkönen, J., Kukkonen, S., Price, K.V., 2005. "Real parameter optimization with differential evolution", in *Proceedings of the 2005 IEEE Congress on Evolutionary Computation*, 1, pp. 506-513.
54. Sato, T. and Hagiwara, M. 1997. "Bee system: finding solution by a concentrated search", in *Computational Cybernetics and Simulation: Proceedings of 1997 IEEE International Conference on Systems, Man and Cybernetics*, 4, pp. 3954-3959.
55. Shi, Y. and Eberhart, R. C. 1998. "Parameter selection in particle swarm optimization", in V. W. Porto, M. Saravanan, D. Waagen and A. E. Eiben (eds.), *Evolutionary Programming VII: Proceedings of the 7th International Conference*, Springer Berlin Heidelberg pp. 591-600.
56. Siddhartha, Sharma, N. and Varun. 2012. A particle swarm optimization algorithm for optimization of thermal performance of a smooth flat plate solar air heater. *Energy*, 38 (1), pp. 406–413.
57. Singh, L., Dhillon, J. and Chauhan, R. 2006. Evaluation of Best Weight Pattern for Multiple Criteria Load Dispatch. *Electric Power Component and Systems*, 34 (1), pp. 21-35.
58. Sivaraj, R. and Ravichandran, T. 2011. A Review of Selection Methods in Genetic Algorithm. *International Journal of Engineering Science and Technology*, 3 (5), pp. 3792-3797.
59. Srinivas, M. and Patnaik, L. M. 1994. Genetic Algorithms: A Survey. *Computer*, 27 (6), pp. 17-26.
60. Storn, R. and Price, K. 1995. *Differential evolution: A simple and efficient adaptive scheme for global optimization over continuous spaces* [report TR-95-012] ICSI, USA.
61. Storn, R. and Price, K. 1997. Differential Evolution – A Simple and Efficient Heuristic for Global Optimization over Continuous Spaces. *Journal of Global Optimization*, 11 (4), p. 341–359.
62. Suman, B. and Kumar, P. 2006. A survey of simulated annealing as a tool for single and multiobjective optimization. *Journal of the Operational Research Society*, 57 (10), p.1143–1160.

63. Sureja, N. 2012. New Inspirations in Nature: A Survey. *International Journal of Computer Applications and Information Technology*, 1 (3), 21-24.
64. Talbi, E. G. 2009. *Metaheuristics: From Design to Implementation*. Hoboken, New Jersey: Wiley.
65. Thengade, A. and Dondal, R. 2012. "Genetic Algorithm - Survey Paper", in *IJCA Proceedings on National Conference on Recent Trends in Computing*, NCRTC (5), pp. 25-29.
66. Thomas, R. M. 2013. Survey of Bacterial Foraging Optimization Algorithm. *International Journal of Innovative Science and Modern Engineering*, 1 (4), pp. 11-12.
67. Wang, G. and Guo, L. 2013. A Novel Hybrid Bat Algorithm with Harmony Search for Global Numerical Optimization, *Journal of Applied Mathematics*, 2013, Article ID 696491, 21 pages, 2013. doi:10.1155/2013/696491
68. Wang, G. Guo, L. Duan, H. Liu, L. and Wang, H. 2012. A Bat Algorithm with Mutation for UCAV Path Planning. *The Scientific World Journal*, 2012, Article ID 418946, 15 pages. doi:10.1100/2012/418946
69. Xie, J., Zhou, Y. and Chen, H. 2013. A Novel Bat Algorithm Based on Differential Operator and Lévy Flights Trajectory. *Computational Intelligence and Neuroscience*, 2013, Article ID 453812, 13 pages, 2013. doi:10.1155/2013/453812
70. Yang, X. S. 2010a. *Engineering optimization: an introduction with metaheuristic applications*. Hoboken, New Jersey: Wiley.
71. Yang X. S. 2010b. "A New Metaheuristic Bat-Inspired Algorithm", in J. R. Gonzalez, D. A. Pelta, C. Cruz, G. Terrazas and N. Krasnogor (eds.), *Nature Inspired Cooperative Strategies for Optimization (NISCO 2010)*, Springer Berlin-Hedelberg, Germany, pp. 65-74.
72. Yang, X. S. 2010c. *Nature-Inspired Metaheuristic Algorithms*. 2nd ed. Frome: Luniver Press.
73. Yang, X. S. 2013. Bat Algorithm: Literature Review and Applications. *International Journal of Bio-Inspired Computation*, 5 (3), pp. 141-149.
74. Yang, X. S. and Deb, S., 2009. "Cuckoo search via Lévy flights", in *Proceedings of World Congress on Nature & Biologically Inspired Computing (NaBIC 2009)*, pp. 210-214.
75. Yang, X. S. and Deb, S., 2010. Engineering Optimisation by Cuckoo Search. *International Journal of Mathematical Modelling and Numerical Optimisation*, 1 (4), pp. 330-343.
76. Yilmaz, S. and Kucuksille, E. U. 2013. Improved Bat Algorithm (IBA) on Continuous Optimization Problems. *Lecture Notes on Software Engineering*, 1 (3), pp. 279-283.
77. Yonezawa, Y. and Kikuchi, T. 1996. "Ecological algorithm for optimal ordering used by collective honey bee behavior", in *Proceedings of the Seventh International Symposium on Micro Machine and and Human Science (MHS '96)*, pp. 249-256.
78. Zaharie, D. 2007. "A comparative analysis of crossover variants in differential evolution", *Proceedings of International Multiconference on Computer Science and Information Technology*, 2, pp. 171-181.
79. Ελληνική Σπηλαιολογική εταιρεία. *Γενικά για τις Νυχτερίδες*. [online] Διαθέσιμο: http://www.ese.edu.gr/default.asp?V_DOC_ID=2237 [Ανάκτηση: 30 Ιουνίου 2013].
80. Σοφιανίδου, Θ. Σ. 2000. *Συστηματική των τετραπόδων*. 2^η ανατυπωμένη βελτιωμένη έκδοση. Θεσσαλονίκη: Εκδόσεις Γιαχούδη – Γιαπούλη.