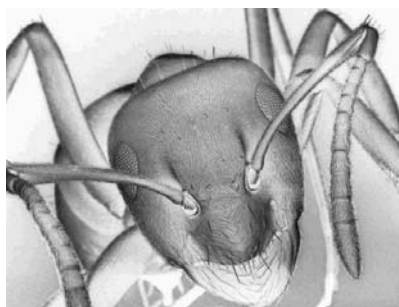




**ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ**  
**ΤΜΗΜΑ ΜΗΧΑΝΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ**  
**ΤΟΜΕΑΣ ΡΕΥΣΤΩΝ**  
**ΕΡΓΑΣΤΗΡΙΟ ΘΕΡΜΙΚΩΝ ΣΤΡΟΒΙΛΟΜΗΧΑΝΩΝ**



**Η ΜΕΘΟΔΟΣ ΒΕΛΤΙΣΤΟΠΟΙΗΣΗΣ ΜΕ ΑΠΟΙΚΙΕΣ ΜΥΡΜΗΓΚΙΩΝ:**  
**ΕΦΑΡΜΟΓΗ ΣΕ ΔΙΑΚΡΙΤΑ ΚΑΙ ΣΥΝΕΧΗ ΠΡΟΒΛΗΜΑΤΑ**

Διπλωματική εργασία του:

**Γιώργου Ε. Φαινέκου**

Επιβλέπων: Κ. Χ. Γιαννάκογλου

Αθήνα, Ιούλιος 2001



## ΠΡΟΛΟΓΟΣ

Τις τρεις τελευταίες δεκαετίες υπάρχει μια έντονη δραστηριοποίηση στο χώρο των υπολογιστικών αλγορίθμων που βασίζονται στη μοντελοποίηση διαφόρων φυσικών, βιολογικών ή κοινωνικών φαινομένων με σκοπό τη δημιουργία μη αιτιοκρατικών ευριστικών αλγορίθμων για την επίλυση NP – δύσκολων Συνδυαστικών Προβλημάτων Βελτιστοποίησης (NP – hard Combinatorial Optimization Problems – COP) αλλά και για την εύρεση του ελαχίστου ή του μεγίστου συνεχών συναρτήσεων ή προβλημάτων.

Ένας από τους πιο πρόσφατους αλγορίθμους που αναπτύχθηκαν στα πλαίσια της παραπάνω έρευνας είναι και η Μέθοδος Βελτιστοποίησης με Αποικίες Μυρμηγκιών (Ant Colony Optimization – ACO) η οποία βασίζεται στη συλλογική συμπεριφορά που εμφανίζουν τα μυρμήγκια κατά την αναζήτηση και τη συγκομιδή τροφής.

Τα αποτελέσματα από τη χρήση του ACO σε διακριτά προβλήματα είναι πολύ ενθαρρυντικά και συγκρίσιμα με μεθόδους που έχουν ερευνηθεί και αναπτυχθεί για πολύ μεγαλύτερο χρονικό διάστημα (πχ Γενετικοί Αλγόριθμοι).

Στα πλαίσια της παρούσης διπλωματικής εργασίας θα γίνει εκτενής ανάλυση των αλγορίθμων ACO αλλά και θα παρουσιασθεί ένας νέος αλγόριθμος (Extended Ant Colony Optimization – EACO) που ανήκει στην κατηγορία ACO και επιλύει συνεχή προβλήματα με κύρια εφαρμογή στην Αντίστροφη Σχεδίαση Αεροτομών (*Inverse Design of Aerodynamic Shapes – IDS*).

Ολοκληρώνοντας τον πρόλογο, θα ήθελα να ευχαριστήσω τον καθηγητή μου κ. Κ. Γιαννάκογλου που μου ανέθεσε αυτή την εργασία και μου έδωσε την ευκαιρία να εμβαθύνω στο επιστημονικό πεδίο την τεχνητής νοημοσύνης καθώς και για την αμέριστη συμπαράστασή του κατά την εκπόνησή της.



<b><u>1</u></b>	<b><u>ΕΙΣΑΓΩΓΗ</u></b>	<b><u>1</u></b>
<b><u>2</u></b>	<b><u>ΚΟΙΝΩΝΙΚΑ ΎΝΤΟΜΑ</u></b>	<b><u>9</u></b>
2.1	ΕΙΣΑΓΩΓΙΚΑ ΣΤΟΙΧΕΙΑ	9
2.2	ΑΥΤΟ-ΟΡΓΑΝΩΣΗ	10
2.3	ΣΤΙΓΜΕΡΓΙΑ	13
2.4	ΑΝΑΖΗΤΗΣΗ ΚΑΙ ΣΥΓΚΟΜΙΔΗ ΤΡΟΦΗΣ ΣΤΙΣ ΑΠΟΙΚΙΕΣ ΜΥΡΜΗΓΚΙΩΝ	15
<b><u>3</u></b>	<b><u>ΑΠΟ ΤΑ ΠΡΑΓΜΑΤΙΚΑ ΣΤΑ «ΨΗΦΙΑΚΑ» ΜΥΡΜΗΓΚΙΑ</u></b>	<b><u>21</u></b>
<b><u>4</u></b>	<b><u>ΔΙΑΚΡΙΤΟΙ ΑΛΓΟΡΙΘΜΟΙ ΒΕΛΤΙΣΤΟΠΟΙΗΣΗΣ</u></b>	<b><u>25</u></b>
4.1	ANT SYSTEM (AS)	25
4.2	ANT COLONY SYSTEM (ACS)	32
4.3	MAX-MIN ANT SYSTEM (MMAS)	34
4.4	ΑΠΟΤΕΛΕΣΜΑΤΑ	37
<b><u>5</u></b>	<b><u>ΑΝΤΙΣΤΡΟΦΗ ΣΧΕΔΙΑΣΗ ΑΕΡΟΤΟΜΩΝ</u></b>	<b><u>43</u></b>
5.1	ΕΙΣΑΓΩΓΗ	43
5.2	ΠΑΡΑΜΕΤΡΟΠΟΙΗΣΗ ΤΗΣ ΑΕΡΟΤΟΜΗΣ	43
5.3	ΣΧΟΛΙΑ	44
<b><u>6</u></b>	<b><u>ΣΥΝΕΧΕΙΣ ΑΛΓΟΡΙΘΜΟΙ ΒΕΛΤΙΣΤΟΠΟΙΗΣΗΣ</u></b>	<b><u>45</u></b>
6.1	CONTINUOUS ANT COLONY OPTIMIZATION (CACO)	45
6.2	ΑΛΓΟΡΙΘΜΟΣ ΕΜΠΝΕΥΣΜΕΝΟΣ ΑΠΟ ΤΑ ΜΥΡΜΗΓΚΙΑ RACHYCONDYLA APICALIS (API)	54
<b><u>7</u></b>	<b><u>EXTENDED ANT COLONY OPTIMIZATION (EACO)</u></b>	<b><u>57</u></b>
7.1	ΕΙΣΑΓΩΓΗ	57

<b>7.2</b>	<b>ΠΕΡΙΓΡΑΦΗ</b>	<b>57</b>
<b>7.3</b>	<b>ΑΝΑΛΥΣΗ</b>	<b>63</b>
<b>8</b>	<b><u>ΑΠΟΤΕΛΕΣΜΑΤΑ ΧΡΗΣΗΣ ΤΟΥ ΕΑΣΟ</u></b>	<b><u>77</u></b>
<b>8.1</b>	<b>ΕΙΣΑΓΩΓΗ</b>	<b>77</b>
<b>8.2</b>	<b>ΡΥΘΜΙΣΗ ΤΟΥ ΑΛΓΟΡΙΘΜΟΥ ΕΑΣΟ</b>	<b>77</b>
8.2.1	ΑΡΧΙΚΟΠΟΙΗΣΗ ΤΟΥ ΑΛΓΟΡΙΘΜΟΥ	78
8.2.2	ΑΡΙΘΜΟΣ ΚΟΜΒΩΝ ΠΛΕΓΜΑΤΟΣ	79
8.2.3	ΠΡΟΣΔΙΟΡΙΣΜΟΣ ΤΩΝ ΠΑΡΑΜΕΤΡΩΝ $A, B, C_E, C_V$	80
8.2.4	ΠΡΟΣΔΙΟΡΙΣΜΟΣ ΤΗΣ ΠΑΡΑΜΕΤΡΟΥ $R$	82
8.2.5	ΔΙΑΔΙΚΑΣΙΑ ΕΠΙΛΟΓΗΣ ΘΕΣΗΣ ΤΗΣ ΦΩΛΙΑΣ	83
8.2.6	ΕΝΑΠΟΘΕΣΗ ΦΕΡΟΜΟΝΗΣ	86
8.2.7	ΠΡΟΣΔΙΟΡΙΣΜΟΣ ΤΩΝ ΠΑΡΑΜΕΤΡΩΝ $N_{AR}$ ΚΑΙ $F_{GL}$	88
8.2.8	ΕΞΑΤΜΙΣΗ ΦΕΡΟΜΟΝΗΣ	88
8.2.9	ΟΜΑΛΟΠΟΙΗΣΗ ΦΕΡΟΜΟΝΗΣ	89
8.2.10	ΕΠΑΝΑΡΧΙΚΟΠΟΙΗΣΗ ΤΟΥ ΑΛΓΟΡΙΘΜΟΥ	90
8.2.11	ΣΥΝΔΥΑΣΜΕΝΗ ΔΡΑΣΗ ΟΜΑΛΟΠΟΙΗΣΗΣ ΤΗΣ ΦΕΡΟΜΟΝΗΣ ΚΑΙ ΕΠΑΝΑΡΧΙΚΟΠΟΙΗΣΗΣ	90
8.2.12	ΧΡΗΣΙΜΟΠΟΙΗΣΗ ΟΡΙΩΝ ΣΤΗ ΦΕΡΟΜΟΝΗ	91
<b>8.3</b>	<b>ΠΑΡΟΥΣΙΑΣΗ ΑΠΟΤΕΛΕΣΜΑΤΩΝ</b>	<b>93</b>
<b>9</b>	<b><u>ΣΥΜΠΕΡΑΣΜΑΤΑ</u></b>	<b><u>105</u></b>
<b>10</b>	<b><u>ΒΙΒΛΙΟΓΡΑΦΙΑ</u></b>	<b><u>107</u></b>

## 1 ΕΙΣΑΓΩΓΗ

Τα προβλήματα που εμφανίζονται στη φύση μπορεί να είναι δύσκολα Συνδυαστικά Προβλήματα Βελτιστοποίησης (hard Combinatorial Optimization Problems – COP) ή να μοντελοποιούνται από συνεχείς συναρτήσεις που εμφανίζουν ένα μεγάλο πλήθος τοπικά βέλτιστων λύσεων κάνοντας κατά αυτόν τον τρόπο δύσκολη, αν όχι αδύνατη, την επίλυσή τους με χρήση κλασσικών αιτιοκρατικών μεθόδων. Για το λόγο αυτό η έρευνα κατευθύνθηκε προς τους στοχαστικούς ευριστικούς αλγορίθμους και ειδικότερα σε αυτούς που εμπνέονται από φυσικά, κοινωνικά ή βιολογικά φαινόμενα [16], [38]. Οι αλγόριθμοι αυτοί έχουν την ιδιότητα να μην εγκλωβίζονται σε τοπικά βέλτιστες λύσεις και μάλιστα μπορούν να εντοπίσουν την περιοχή της ολικά βέλτιστης λύσης συνήθως μετά από ένα μεγάλο αριθμό εκτιμήσεων της αντικειμενικής συνάρτησης.

Γενικά σε ένα πρόβλημα βελτιστοποίησης, ο σκοπός μας να προσδιορίσουμε ένα διάλυμα λύσεων  $x^* \in S$  το οποίο θα ικανοποιεί μια *συνάρτηση κόστους* ή ένα *αντικειμενικό κριτήριο*  $\varphi: S \rightarrow \mathcal{R}$  για το οποίο θα ισχύει  $\forall x \in S: \varphi(x^*) \leq \varphi(x)$  (πρόβλημα ελαχιστοποίησης) ή  $\varphi(x^*) \geq \varphi(x)$  (πρόβλημα μεγιστοποίησης).

Η διαδικασία εύρεσης λύσεων στους στοχαστικούς ευριστικούς αλγορίθμους πραγματοποιείται πάντοτε μέσω μιας επαναληπτικής διαδικασίας. Σε κάθε επανάληψη δημιουργούνται από τον αλγόριθμο μια (πχ Simulated Annealing) ή περισσότερες (πχ Genetic Algorithms και Ant System) λύσεις από τον αντίστοιχο αριθμό ατόμων (*individuals*) που ανήκουν στο χώρο των πιθανών λύσεων. Η καταλληλότητα των λύσεων αυτών εκτιμάται μέσω της συναρτήσεως κόστους. Πολλές φορές οι αλγόριθμοι αυτοί κατά τη διαδικασία εύρεσης λύσεων προσαρμόζονται συνεχώς στο χώρο των λύσεων μεταβάλλοντας τις παραμέτρους τους. Η παραπάνω γενική διαδικασία συνεχίζεται μέχρι έως ότου ολοκληρωθεί ένας συγκεκριμένος αριθμός επαναλήψεων ή ικανοποιηθεί ένα κριτήριο σύγκλισης.

Τα βασικότερα χαρακτηριστικά των αλγορίθμων που εμπνέονται από τη φύση είναι:

- μοντελοποιούν σε κάποιο βαθμό ένα φυσικό (πχ η ανόπτηση στα μέταλλα), κοινωνικό (πχ η αλληλεπίδραση μεταξύ των ανθρώπων και του πολιτισμού τους ή γενικότερα μεταξύ των ζώντων όντων και περιβάλλοντός τους) ή βιολογικό φαινόμενο (πχ η εξέλιξη των ειδών).

- έχουν στοιχεία στοχαστικότητας (μη αιτιοκρατικοί), με κατευθυνόμενη στοχαστικότητα.
- πολλές φορές παρουσιάζουν παράλληλη δομή, δηλαδή πράκτορες (*agents*) ή άτομα (*individuals*) που δραστηριοποιούνται ταυτόχρονα και δημιουργούν πολλαπλές λύσεις του προβλήματος. Το γεγονός αυτό επιτρέπει την παράλληλη εκτέλεσή τους σε παράλληλους ηλεκτρονικούς υπολογιστές. Όταν χρησιμοποιείται ένας πληθυσμός ατόμων τότε τα άτομα μπορεί να επικοινωνούν είτε άμεσα (πχ Genetic Algorithms) είτε έμμεσα (πχ Ant System) μεταξύ τους.
- είναι προσαρμόσιμοι σε διαφορετικά «περιβάλλοντα» παρουσιάζοντας «στιβαρότητα». Συνεπώς, μπορούν να λειτουργήσουν αγνοώντας το θόρυβο αλλά και να αξιοποιηθούν σε διάφορα είδη προβλημάτων με ελάχιστες αλλαγές στον κώδικα.

Πρέπει να επισημανθεί ότι βασικά υπάρχουν δύο μηχανισμοί στους στοχαστικούς αλγορίθμους:

- η **εκμετάλλευση** (*exploitation*) του χώρου των λύσεων κατά την οποία πραγματοποιείται τοπική αναζήτηση βέλτιστων λύσεων στις περιοχές που έχουν ήδη δώσει καλά αποτελέσματα.
- η **εξερεύνηση** (*exploration*) όλου του δυνατού χώρου των λύσεων για την αποτελεσματικότερη αναζήτηση νέων προσοδοφόρων περιοχών.

Οι δύο αυτές διαδικασίες είναι ανταγωνιστικές μεταξύ τους και ένας σωστά ρυθμισμένος ευριστικός αλγόριθμος πρέπει να λειτουργεί στη «χρυσή τομή» των δύο.

Τα τρία κυριότερα μειονεκτήματα των στοχαστικών αλγορίθμων είναι τα εξής:

- πολλές φορές δυσκολεύονται να προσεγγίσουν την ακριβή τιμή της βέλτιστης λύσης του προβλήματος αν και συνήθως εντοπίζουν γρήγορα την περιοχή στην οποία αυτή βρίσκεται.
- απαιτούν μεγάλο αριθμό εκτιμήσεων της αντικειμενικής συνάρτησης – και επομένως έχουν υψηλό υπολογιστικό κόστος – προκειμένου να δώσουν ένα καλό αποτέλεσμα.
- συνήθως, έχουν ένα μεγάλο αριθμό παραμέτρων ο οποίος κάνει δύσκολη τη βέλτιστη ρύθμιση του αλγορίθμου.

Η σχέση μεταξύ μέγιστου αριθμού επαναλήψεων και ποιότητας της βέλτιστης λύσης είναι άμεση και πρέπει να λαμβάνουμε υπόψη μας το πρόβλημα που θέλουμε να επιλύσουμε. Προφανώς, θα πρέπει να γνωρίζουμε εκ των προτέρων αν απλά επιθυμούμε μια καλή λύση ή είναι απαραίτητος ο εντοπισμός της βέλτιστης λύσης.

Ένα σημαντικό χαρακτηριστικό των στοχαστικών ευριστικών αλγορίθμων που πρέπει να λαμβάνεται υπόψη από το χρήστη είναι η επαναληψιμότητα της απόδοσης του αλγορίθμου. Λόγω της στοχαστικής φύσης των αλγορίθμων κάθε εκτέλεσή τους με διαφορετικές αρχικές συνθήκες δίνει και διαφορετικό αποτέλεσμα. Υπολογίζοντας τη μέση τιμή της βέλτιστης λύσης και την τυπική απόκλιση αυτής για ένα μεγάλο αριθμό εκτελέσεων του αλγορίθμου, θα πρέπει σε έναν καλό στοχαστικό ευριστικό αλγόριθμο η τυπική απόκλιση να είναι σχετικά μικρή. Το γεγονός αυτό προσφέρει τη σιγουριά στο χρήστη ότι ο αλγόριθμος που χρησιμοποιεί θα έχει την αναμενόμενη απόδοση στις περισσότερες εκτελέσεις του. Πολλές φορές είναι επιθυμητή η μείωση της απόδοσης του αλγορίθμου με σκοπό την αύξηση της επαναληψιμότητά της ή και το αντίστροφο ελπίζοντας ότι μια τυχαία εκτέλεση του αλγορίθμου μπορεί να δώσει ένα πάρα πολύ καλό αποτέλεσμα.

Μερικές από τις γνωστότερες ευριστικές μεθόδους που εμπνεύστηκαν από τη φύση (*heuristics from nature* [8], [15]) είναι:

- οι **Γενετικοί Αλγόριθμοι - Genetic Algorithms (GA)**: Holland (1975)

Οι Γενετικοί Αλγόριθμοι αποτελούν μια υποκατηγορία των Εξελικτικών Αλγορίθμων οι οποίοι βασίζονται στη θεωρία εξέλιξης των ειδών του Δαρβίνου. Σύμφωνα με τη θεωρία του Δαρβίνου, η φυσική εξέλιξη των ατόμων ενός πληθυσμού βασίζεται στη *φυσική διαλογή* (σε όλα τα είδη των ζώων πιθανότερο είναι να επιβιώσουν τα πιο ικανά, γρήγορα κτλ), την *αναπαραγωγή* (τα ζώα αναπαράγονται μεταφέροντας τα γονιδιά τους στα παιδιά τους) και τη *μετάλλαξη* (κατά το πέρασμα των αιώνων όλα τα είδη των ζώων υφίστανται μεταλλάξεις που διαφοροποιούν ορισμένα χαρακτηριστικά τους όχι πάντοτε προς το καλύτερο).

Όμοια με το βιολογικό ανάλογο, οι Γενετικοί Αλγόριθμοι βασίζονται στην ύπαρξη ενός πληθυσμού ατόμων, τα οποία αποτελούν και λύσεις του προβλήματος. Ο πληθυσμός αυτός των ατόμων μεταβάλλεται από γενιά σε γενιά (δηλαδή σε κάθε επανάληψη του αλγορίθμου) καθώς υφίσταται: (α) *διαλογή (sorting)* ανάλογα με την καταλληλότητά του, (β) *διασταύρωση (crossover)* έτσι ώστε να μεταφέρει τα χαρακτηριστικά του στην επόμενη γενιά και (γ) *μετάλλαξη (mutation)* έτσι ώστε να καλύπτεται όλος ο χώρος των λύσεων. Να αναφέρουμε, δε, ότι με το μηχανισμό της διαλογής πραγματοποιείται *εκμετάλλευση (exploitation)* του χώρου των

λύσεων ενώ με τους μηχανισμούς της διασταύρωσης και της μετάλλαξης *εξερεύνηση* (*exploration*).

- οι **Εξελικτικές Στρατηγικές – Evolutionary Strategies** (ES): Rochenberg (1973)

Η Εξελικτική Στρατηγική αποτελεί και αυτή μια υποκατηγορία των Εξελικτικών Αλγορίθμων. Οι διαφοροποιήσεις της με τους Γενετικούς αλγορίθμους εντοπίζονται σε παραλλαγές των μηχανισμών διαλογής, διασταύρωσης και μετάλλαξης, στον τρόπο απεικόνισης των ατόμων – λύσεων αλλά και στις διαστάσεις του πληθυσμού των γονέων και των απογόνων.

- η **Προσομοιωμένη Ανόπτηση – Simulated Annealing** (SA): Kirkpatrick, Gelatt, Vecchi (1983)

Η Προσομοιωμένη Ανόπτηση (SA) βασίζεται στο φυσικό φαινόμενο της αποκατάστασης της ισορροπίας ενός μετάλλου σε φυσικοχημικό επίπεδο ή επίπεδο δομής κατά την αργή απόψυξή του από μια αρχική θερμοκρασία  $T_s$  σε μια τελική  $T_a$ . Σύμφωνα με τον Boltzman η ενέργεια  $E$  ενός μετάλλου σε θερμοκρασία  $T$  κατανέμεται στις διαφορετικές καταστάσεις του με πιθανότητα:  $p(E, T) \approx \exp(-E/kT)$ , όπου  $k$  η σταθερά Boltzman. Όταν ένα μέταλλο αποψύχεται, τότε, εν γένει η ενέργειά του κινείται προς τη χαμηλότερη στάθμη, αλλά δεν αποκλείεται με κάποια πιθανότητα να αυξηθεί για μικρό χρονικό διάστημα ώσπου να βρει μια νέα κατάσταση με χαμηλότερα ενεργειακά επίπεδα. Η πιθανότητα να αυξηθεί η ενεργειακή στάθμη του μετάλλου μειώνεται όσο μειώνεται και η θερμοκρασία  $T$ .

Ο αλγόριθμος SA είναι πρακτικά μια βελτίωση του Random Hill Climbing (RHC). Στον SA κάθε λύση (ενεργειακή στάθμη) που βελτιώνει την καταλληλότητα της συνάρτησης κόστους (μείωση της ενεργειακής στάθμης) είναι αποδεκτή. Επίσης, αποδεκτές αλλά με κάποια πιθανότητα είναι και οι λύσεις που μειώνουν την καταλληλότητα της συνάρτησης (αύξηση της ενεργειακής στάθμης). Η πιθανότητα αποδοχής της λύσης εξαρτάται από την μείωση της καταλληλότητας και από τον τρέχον αριθμό επανάληψης (θερμοκρασίας) του SA.

- **Tabu search** (TS): Glover (1989)

Ο TS είναι ένας αλγόριθμος τοπικής έρευνας (*μετα-ευριστικός – metaheuristic*) που χρησιμοποιείται για να καθοδηγήσει την έρευνα άλλων αλγορίθμων, όπως για παράδειγμα του RHC και του SA. Πρακτικά εισάγει μια μορφή «μνήμης» στον βασικό αλγόριθμο η οποία χρησιμοποιείται για την απομνημόνευση των προηγούμενων βημάτων της διαδικασίας βελτιστοποίησης και ιδιαίτερα αυτών που βελτίωσαν τη λύση του προβλήματος. Στην βασική έκδοση του αλγορίθμου δεν επιτρέπεται η χρήση των στοιχείων που βρίσκονται στη λίστα της

μνήμης. Με αυτόν τον τρόπο ουσιαστικά πραγματοποιείται καλύτερη εξερεύνηση του χώρου των λύσεων και αποφυγή των τοπικά βέλτιστων λύσεων. Να σημειώσουμε ότι ο αλγόριθμος TS στη μορφή που αναφέρθηκε παραπάνω είναι αιτιοκρατικός.

- τα **Νευρωνικά Δίκτυα – Neural Nets (NN)**: Hopfield, Tank (1985)

Τα Τεχνητά Νευρωνικά Δίκτυα (Artificial Neural Nets) εμπνεύστηκαν από τη δομή και λειτουργία του ανθρώπινου εγκεφάλου. Περιληπτικά, ο ανθρώπινος εγκέφαλος περιέχει ένα μεγάλο πλήθος, περίπου  $10^{12}$ , διασυνδεδεμένων νευρώνων. Ο κάθε νευρώνας δέχεται σήματα από τους άλλους με τους οποίους είναι συνδεδεμένος και όταν τα σήματα αυτά ξεπεράσουν ένα συγκεκριμένο όριο τότε και αυτός με τη σειρά του εκπέμπει ένα νέο σήμα.

Τα Τεχνητά Νευρωνικά Δίκτυα είναι αρκετά παρόμοια με τα φυσικά. Ένα απλό Τεχνητό Νευρωνικό Δίκτυο αποτελείται από τουλάχιστον τρία επίπεδα νευρώνων εκ των οποίων ένα είναι για την είσοδο και ένα για την έξοδο. Η επεξεργασία της πληροφορίας γίνεται σε πολλές απλές μονάδες, τους νευρώνες. Ο κάθε νευρώνας δέχεται σήματα από τους νευρώνες με τους οποίους επικοινωνεί μέσω των συνάψεων. Κάθε σήμα που εκπέμπουν οι νευρώνες πολλαπλασιάζεται με τον συντελεστή βαρύτητας που αντιστοιχεί σε κάθε σύνδεση. Ο κάθε νευρώνας αθροίζει τα σήματα που δέχεται και εκπέμπει ένα νέο με βάση τη συνάρτηση δραστηριοποίησής του. Τέλος, κάθε νευρώνας εκπέμπει μόνο ένα σήμα σε κάθε χρονική στιγμή, αλλά προς πολλούς άλλους νευρώνες.

- η **Βελτιστοποίηση με Αποικίες Μυρμηγκιών – Ant System<sup>1</sup> (AS)**: Dorigo, Coloni, Maniezzo (1991)

Ο Dorigo και οι συνεργάτες του εμπνευσμένοι από τη δουλειά του Deneubourg [17], [18], [29] πάνω στα **κοινωνικά έντομα** (*social insects*), και ειδικότερα τα μυρμηγκία, δημιούργησαν ένα πρωτοποριακό αλγόριθμο (Ant System – AS [19], [20]) βασισμένο στην έμμεση επικοινωνία (*stigmergy*) των μυρμηγκιών μέσω χημικών ουσιών – της φερομόνης<sup>2</sup> – κατά την αναζήτηση και συλλογή τροφής (*foraging*). Αναλυτικότερα, τα περισσότερα είδη μυρμηγκιών κατά την αναζήτηση και συγκομιδή τροφής εναποθέτουν φερομόνη στο πέρασμά τους. Η φερομόνη αυτή είναι ανιχνεύσιμη και από τα υπόλοιπα μυρμηγκία της αποικίας τους τα οποία αποφασίζουν με

---

<sup>1</sup> Ο πρώτος αλγόριθμος που αναπτύχθηκε από τον Dorigo ονομάζονταν Ant System (AS). Αργότερα με την εξέλιξη του χώρου δημιουργήθηκε ο όρος Ant Colony Optimization – ACO ο οποίος περιλαμβάνει όλους του αλγορίθμους (AS, Ant Colony System – ACS, Max-Min Ant System – MMAS, Continuous Ant Colony Optimization – CACO κτλ) που βασίζονται στις αρχές του κεφ. 3.

<sup>2</sup> Η **φερομόνη** (*pheromone*) είναι μια χημική ουσία την οποία εκλύουν τα ζώα και τα έντομα για να επικοινωνήσουν έμμεσα με τα άλλα μέλη είδους τους.

κάποια πιθανότητα αν θα ακολουθήσουν το ίχνος αυτό. Η ιδιότητα αυτή των μυρμηγκιών, όπως θα δούμε παρακάτω, οδηγεί μέσω του φαινομένου της αυτο-οργάνωσης στην εύρεση της συντομότερης διαδρομής μεταξύ της φωλιάς τους και της τοποθεσίας της τροφής.

Ο αλγόριθμος AS που βασίζεται στο υπολογιστικό ανάλογο της παραπάνω διαδικασίας επιλύει διακριτά προβλήματα (με πρώτη και συνηθέστερη εφαρμογή το Traveling Salesman Problem – TSP) με πάρα πολύ καλά αποτελέσματα οπότε και γέννησε μια σειρά βελτιωμένων εκδοχών του – Ant Colony System (ACS) [22], ANT-Q [21], Max-Min Ant System (MMAS) [49], AS-rank [9] – οι οποίοι βρίσκουν πολλαπλές εφαρμογές σε πληθώρα θεωρητικών και τεχνικών προβλημάτων όπως [6], [23]:

- Το Πρόβλημα του Περιπλανώμενου Πωλητή (Traveling Salesman Problem – TSP): Στο πρόβλημα αυτό πρέπει να βρεθεί η συντομότερη διαδρομή που συνδέει  $n$  πόλεις. Κάθε πόλη περιλαμβάνεται μόνο μια φορά στη διαδρομή και ο πωλητής πρέπει να επιστρέψει στην πόλη από όπου ξεκίνησε. Το πρόβλημα αυτό λόγω της απλότητάς του χρησιμοποιείται για να γίνονται συγκρίσεις της απόδοσης των διαφόρων αλγορίθμων.
- Το Πρόβλημα της Τετραγωνικής Ανάθεσης (Quadratic Assignment Problem – QAP): Στο πρόβλημα αυτό πρέπει να γίνει ανάθεση  $n$  οντοτήτων σε  $n$  τοποθεσίες έτσι ώστε το κόστος της ανάθεσης να είναι ελάχιστο.
- Το Πρόβλημα Προγραμματισμού Εργασιών (Job-Shop Scheduling Problem – JSP): Στο πρόβλημα αυτό υπάρχουν  $M$  εργαλειομηχανές και  $J$  εργασίες που πρέπει να γίνουν. Για τις  $J$  εργασίες υπάρχει μια καθορισμένη διαδοχή στις εργαλειομηχανές. Ο στόχος του προβλήματος είναι η ελαχιστοποίηση του μέγιστου χρόνου που απαιτείται για την ολοκλήρωση των εργασιών με περιορισμό να μην υπάρχουν δύο εργασίες την ίδια χρονική στιγμή στην ίδια εργαλειομηχανή.

Ξεκινώντας από προβλήματα όπως τα παραπάνω ακολούθησε ένα πλήθος δημοσιεύσεων στα διεθνή περιοδικά όπου οι αλγόριθμοι ACO χρησιμοποιούνται για την επίλυση προβλημάτων όπως:

- εκτίμηση βλαβών σε συστήματα ισχύος (fault section estimation in power systems) [13]
- καθορισμός συχνοτήτων ραδιοφωνικών εκπομπών (Frequency Assignment Problem – FAP) [36]

- έλεγχος δικτύων επικοινωνίας (Communications Network Control) [12]
- Shortest Common Supersequence (SCS) problem [39]
- Flow Shop Problem – FSP [48]
- έλεγχος της στάθμευσης των πλοίων στο λιμάνι (Ship Berthing Problem – SBP) [51]

Αργότερα, ο Bilchev [3], [4] κατάφερε να δώσει μια πρώτη εκδοχή του AS για την επίλυση προβλημάτων στο συνεχές πεδίο. Την αρχική του ιδέα βελτίωσαν οι Wodrich [52] και Mathur [37] και κατόπιν ακολούθησε μια πληθώρα εφαρμογών κυρίως στο χώρο της χημικής βιομηχανίας [32], [33] αλλά και αλλού (πχ έλεγχος στη παραγωγή ηλεκτρικής ενέργειας και θερμότητας σε σταθμούς συνδυασμένης παραγωγής – combined heat and power economic dispatch [46]).

Στην παρούσα εργασία θα παρουσιαστεί ένας νέος αλγόριθμος που βασίζεται στον AS και επιλύει συνεχή προβλήματα με επίκεντρο την αντίστροφη σχεδίαση (*inverse design*) αεροτομών.

Για την καλύτερη και σχολαστικότερη ανάπτυξη της θεωρίας ακολουθείται η παρακάτω ανάπτυξη. Στο Κεφάλαιο 2 παρουσιάζονται αναλυτικά τα βιολογικά φαινόμενα πάνω στα οποία στηρίζεται η μέθοδος και στο 3 γίνεται μια πρώτη προσέγγιση για τη μετάβαση από το φυσικό μοντέλο στο ψηφιακό. Ακολούθως, στο Κεφάλαιο 4 παρουσιάζονται οι αλγόριθμοι που επιλύουν διακριτά προβλήματα με χρήση του μοντέλου των αποικιών των μυρμηγκιών ACO. Αντίστοιχα, στο Κεφάλαιο 6 αναλύονται σύντομα οι ήδη υπάρχοντες αλγόριθμοι ACO για την επίλυση προβλημάτων στο συνεχές πεδίο.

Η ανάπτυξη των Κεφαλαίων 4 και 6 κρίνεται απαραίτητη για τους παρακάτω λόγους:

(α) παρουσιάζεται η σχετική έρευνα στο χώρο των αλγορίθμων ACO.

(β) γίνεται εύκολα κατανοητή η μετάβαση από το φυσικό πρόβλημα (= συγκομιδή τροφής αξιοποιώντας τη συντομότερη διαδρομή) στο ψηφιακό (= εύρεση της συντομότερης διαδρομής στο TSP) και από εκεί σε άλλα προβλήματα όπου ο στόχος δεν είναι η ελαχιστοποίηση του μήκους αλλά κάποια άλλη αντικειμενική συνάρτηση.

(γ) αναλύονται στοιχεία και ιδέες που θα αξιοποιηθούν τελικά στο νέο αλγόριθμο που παρουσιάζεται σε αυτήν την εργασία.

Στο Κεφάλαιο 7 αναπτύσσεται ο νέος αλγόριθμος που προτείνεται σε αυτή την εργασία, στο Κεφάλαιο 8 θα γίνει παράθεση, ανάλυση και σύγκριση των αποτελεσμάτων και τέλος, στο Κεφάλαιο 9 θα συζητηθούν τα συμπεράσματα της όλης διαδικασίας.



## 2 ΚΟΙΝΩΝΙΚΑ ΈΝΤΟΜΑ

### 2.1 ΕΙΣΑΓΩΓΙΚΑ ΣΤΟΙΧΕΙΑ

Τα κοινωνικά έντομα<sup>3</sup> – όπως τα μυρμήγκια, οι τερμίτες, οι μέλισσες και οι σφήγκες – πάντοτε εντυπωσίαζαν τους ανθρώπους, επιστήμονες και μη, με τη δομή και την οργάνωσή τους. Δεν είναι τυχαίο ότι υπάρχει πληθώρα πηγών στο *internet* με λεπτομέρειες γύρω από τη ζωή τους, για το πως μπορεί κανείς να τα παρατηρήσει ή και ακόμη να τα εκθρέψει ως κατοικίδια.

Ειδικότερα, τα μυρμήγκια αποτελούν ένα από τα πιο επιτυχημένα είδη ζωής πάνω στον πλανήτη μας έχοντας αποικήσει κάθε μέρος του. Ζούνε εδώ και 100 εκατομμύρια χρόνια και ο πληθυσμός τους υπολογίζεται γύρω στα  $10^{16}$  άτομα [25].

Κάθε μεμονωμένο έντομο, από όλα τα είδη των εντόμων, εμφανίζει αρκετά περίπλοκη δομή χρησιμοποιώντας πληθώρα αισθητηρίων οργάνων ώστε να συγκεντρώνει πληροφορίες από το περιβάλλον του και να λαμβάνει τις ανάλογες αποφάσεις. Για παράδειγμα, είναι αξιοσημείωτο πως τα μυρμήγκια *Cataglyphis fortis* καταφέρνουν να εντοπίσουν τον προορισμό τους (τοποθεσία με φαγητό) – όταν αλλάξουμε την τρέχουσα θέση τους – χρησιμοποιώντας οπτικά δεδομένα από τη μορφολογία του περιβάλλοντος, τον ήλιο ως πυξίδα, τη μυρουδιά του στόχου, την περιορισμένη μνήμη τους αλλά και ένα αθροιστή που διαθέτουν προκειμένου να υπολογίζουν αποστάσεις [14].

Άλλο χαρακτηριστικό παράδειγμα της δομής και λειτουργίας των εντόμων είναι ο τρόπος με τον οποίο πετούν και επικοινωνούν οι μέλισσες. Έχει αποδειχθεί [14] ότι οι μέλισσες καθορίζουν την ταχύτητά τους και μετρούν την απόσταση που έχουν διανύσει ανάλογα με την «οπτική ροή» των πληροφοριών στον αμφιβληστροειδή: πχ πετώντας χαμηλά και διανύοντας μικρή απόσταση δημιουργείται η ίδια ροή πληροφοριών με πετώντας ψηλά και διανύοντας μεγάλη απόσταση. Χρησιμοποιώντας τον ίδιο μηχανισμό και συνδυάζοντας το ρυθμό καθόδου τους με την πρόσω ταχύτητά τους καταφέρνουν να επιτυγχάνουν πάντα ομαλές προσγειώσεις σε οριζόντια επίπεδα χωρίς να έχουν σαφή γνώση τόσο του ύψους τους όσο και της ταχύτητάς τους. Τέλος, είναι γνωστό ότι η επικοινωνία μεταξύ των μελισσών επιτυγχάνεται μέσω του περίφημου

---

<sup>3</sup> **Κοινωνικά έντομα** (*social insects*): έντομα που ζουν σε αποικίες και η γενικότερη συμπεριφορά τους κατευθύνεται από την ανάγκη επιβίωσης της ομάδας και όχι του μεμονωμένου ατόμου.

χορού των μελισσών (*waggle dance of honeybees*) ο οποίος κωδικοποιεί την απόσταση και τη διεύθυνση της θέσης της τροφής.

## 2.2 ΑΥΤΟ-ΟΡΓΑΝΩΣΗ

Τα παραπάνω παραδείγματα παρουσιάζουν μόνο μερικές από τις ατομικές ικανότητες κάθε εντόμου. Είναι φανερό, όμως, ότι αυτές οι ικανότητες των εντόμων, μεμονωμένα, δεν είναι δυνατό να εμφανίσουν ένα συνολικό αρμονικά δομημένο αποτέλεσμα, όπως το κτίσιμο των φωλιών στους τερμίτες ή την εύρεση της συντομότερης διαδρομής μεταξύ δύο σημείων στα μυρμήγκια, και μάλιστα χωρίς την ύπαρξη κεντρικής διοίκησης. Επιπρόσθετα, είναι πολύ σημαντικό το γεγονός ότι τα κοινωνικά έντομα επιτελούν αυτές τις εργασίες με μεγάλη **ευελιξία** και **προσαρμοστικότητα** (*flexibility*): μπορούν να προσαρμόζονται στις εκάστοτε συνθήκες του περιβάλλοντος, αλλά και **στιβαρότητα** (*robustness*): μπορούν να ολοκληρώσουν οποιαδήποτε συλλογική εργασία ακόμα και αν κάποια άτομα αποτύχουν.

Το μυστικό της «επιτυχημένης» διεκπεραίωσης συλλογικών εργασιών και μάλιστα με βέλτιστο τρόπο βρίσκεται στην **αυτο-οργάνωση** (*self-organization* SO) των εντόμων. Στα SO μοντέλα γίνεται η βασική θεώρηση ότι το κάθε άτομο-έντομο είναι ένας απλός «πράκτορας» (*agent*) που μπορεί να διεκπεραιώσει μόνο απλές λειτουργίες. Η θεώρηση αυτή δεν λαμβάνει υπόψη την περίπλοκη δομή του μεμονωμένου εντόμου.

Για παράδειγμα, στα SO μοντέλα αυτό που έχει σημασία είναι η μετακίνηση της μέλισσας μεταξύ δύο σημείων και όχι πως αυτό πραγματοποιήθηκε, δηλαδή πια αισθητήρια μέσα χρησιμοποίησε η μέλισσα για τον εντοπισμό της θέσης της και του στόχου της, αν έχει περιορισμένη ικανότητα μνήμης κτλ. Ομοίως, στις αποικίες των μυρμηγκιών το βασικό χαρακτηριστικό που λαμβάνουν υπόψη τα SO μοντέλα είναι ότι τα μυρμήγκια μπορούν να ακολουθήσουν με κάποια πιθανότητα ένα ίχνος φερομόνης και να το ενισχύσουν χωρίς να ενδιαφέρει πως ανιχνεύεται το ίχνος αυτό, πως λαμβάνεται η απόφαση από το μυρμήγκι να το ακολουθήσει ή όχι και τέλος με ποιον τρόπο ενισχύει το ήδη υπάρχον ίχνος φερομόνης.

Τα SO μοντέλα που έχουν αναπτυχθεί με βάση την παραπάνω θεώρηση δείχνουν ότι είναι δυνατόν να εμφανιστούν περίπλοκες συλλογικές συμπεριφορές σε μια αποικία εντόμων. Αναλυτικότερα, η **αυτο-οργάνωση** [6] είναι ένα σύνολο δυναμικών μηχανισμών με τους οποίους σχηματίζονται δομές σε ένα σύστημα από αλληλεπιδράσεις των συνιστωσών του. Οι αλληλεπιδράσεις αυτές στηρίζονται καθαρά σε τοπικές πληροφορίες χωρίς να λαμβάνουν υπόψη τους τη συνολική εικόνα.

Η αυτο-οργάνωση χαρακτηρίζεται από 4 βασικά στοιχεία. Τη θετική ανάδραση, την αρνητική ανάδραση, την ενίσχυση των τυχαίων διακυμάνσεων και την αλληλεπίδραση μεταξύ των συνιστωσών του συστήματος.

Η **θετική ανάδραση**<sup>4</sup> (*positive feedback*) ή **αυτοκατάλυση** είναι ένα σύνολο κανόνων το οποίο είναι υπεύθυνο για τη δημιουργία των βασικών δομών. Κατά τη συγκεκριμένη διεργασία ένα έντομο αντιδρά στην ενεργοποίηση από ένα ερέθισμα (πχ εύρεση τροφής ή εντοπισμός ίχνους φερομόνης) φροντίζοντας για την προσέλκυση και άλλων εντόμων ή την ενίσχυση του ερεθίσματος. Γενικά, η θετική ανάδραση περιλαμβάνει δυο μηχανισμούς τη **στρατολόγηση** (*recruitment*) και την **ενίσχυση** (*reinforcement*).

Χαρακτηριστικά παραδείγματα στρατολόγησης είναι: (α) Έστω ότι ένα μυρμήγκι εντοπίζει μια τοποθεσία με τροφή. Κατά την επιστροφή του στη φωλιά εναποθέτει φερομόνη στη διαδρομή του σε ποσότητα ανάλογη με την ποιότητα και την ποσότητα της τροφής που έχει εντοπίσει. Το ίχνος αυτό της φερομόνης είναι ανιχνεύσιμο από τα υπόλοιπα μυρμήγκια της αποικίας τα οποία είναι πιθανό να προτιμήσουν να ακολουθήσουν την ίδια διαδρομή με μεγαλύτερη πιθανότητα από μια άλλη νέα-τυχαία διαδρομή. Με αυτόν τον τρόπο ένα μυρμήγκι καταφέρνει να «στρατολογήσει» και άλλα μυρμήγκια με σκοπό να εκμεταλλευθούν την ίδια **πηγή τροφής** (*food source*). (β) Έστω ότι μια μέλισσα εντοπίζει μια τοποθεσία με τροφή. Τότε επιστρέφει στη φωλιά με κάποια ποσότητα τροφής και κατόπιν αν η ποιότητα και η ποσότητα της τροφής στη συγκεκριμένη τοποθεσία είναι ικανοποιητική, πραγματοποιεί τον λεγόμενο χορό των μελισσών ο οποίος εμπεριέχει πληροφορίες για την τοποθεσία της τροφής (κατεύθυνση και απόσταση) αλλά και την ποσότητα και ποιότητα αυτής. Οι άλλες μέλισσες της αποικίας που παρακολουθούν το χορό αυτό είναι πολύ πιθανό να «στρατολογηθούν» για την εκμετάλλευση της τροφής στην τοποθεσία αυτή.

Ένα χαρακτηριστικό παράδειγμα ενίσχυσης είναι το εξής: Έστω ένα μυρμήγκι εντοπίζει ένα ίχνος φερομόνης. Με κάποια πιθανότητα αποφασίζει να το ακολουθήσει. Αν το ακολουθήσει τότε ενισχύει το ίχνος αυτό εναποθέτοντας και την δική του φερομόνη. Με τον τρόπο αυτό η διαδρομή που ενισχύεται με επιπλέον φερομόνη γίνεται πιο ελκυστική για τα επόμενα μυρμήγκια.

---

<sup>4</sup> Η **θετική ανάδραση** θα πρέπει να χρησιμοποιείται με ιδιαίτερη προσοχή καθώς μπορεί να οδηγήσει σε εγκλωβισμό σε **στάσιμες καταστάσεις** (*stagnation*) και να μην μπορεί να βελτιστοποιηθεί το σύστημα. Τέτοιες καταστάσεις υπάρχει περίπτωση να εμφανιστούν σε τοπικά βέλτιστα ή όταν μια αρχική τυχαία λύση δώσει μια πάρα πολύ καλή τιμή και εγκλωβίσει τους υπόλοιπους πράκτορες (*agents*) χωρίς να υπάρχει δυνατότητα να ξεφύγουν.

Η **αρνητική ανάδραση** (*negative feedback*) αναιρεί τα αποτελέσματα της θετικής ανάδρασης και βοηθάει στη σταθεροποίηση σε κάποια συγκεκριμένη δομή. Η αρνητική ανάδραση μπορεί να μοντελοποιεί:

- την εξάντληση κάποιας πηγής τροφής. Για παράδειγμα, οι μέλισσες σταματούν να εκτελούν το χορό τον μελισσών για μια τοποθεσία στην οποία η τροφή έχει εξαντληθεί και επομένως δεν στρατολογούνται άλλες μέλισσες για εκμετάλλευση της περιοχής αυτής.
- την εγκατάλειψη κάποιου μονοπατιού από τα μυρμήγκια. Για παράδειγμα, έχει παρατηρηθεί ότι η φερομόνη που εναποθέτουν τα μυρμήγκια στη διαδρομή τους εξατμίζεται με κάποιο ρυθμό. Αυτό έχει ως αποτέλεσμα την εγκατάλειψη των διαδρομών που για διάφορους εξωγενείς λόγους<sup>5</sup> δεν χρησιμοποιούνται συχνά
- τον ανταγωνισμό μεταξύ δύο διαφορετικών πηγών τροφής. Για παράδειγμα, όταν δύο μέλισσες έχουν εντοπίσει δύο διαφορετικές πηγές τροφής, εκτελούν το χορό των μελισσών στη φωλιά τους. Η μέλισσα με την καλύτερη ποιοτικά και ποσοτικά πηγή τροφής «στρατολογεί» και τα περισσότερα μέλη της αποικίας απομακρύνοντας το ενδιαφέρον από την άλλη περιοχή.

Η **ενίσχυση των τυχαίων διακυμάνσεων** (*amplification of random fluctuations*) είναι σημαντικότερη διαδικασία καθώς μπορεί να οδηγήσει στη δημιουργία νέων βελτιωμένων λύσεων ή και ακόμα και της συνολικά βέλτιστης λύσης ξεφεύγοντας ενδεχόμενα από τοπικά βέλτιστες συμπεριφορές. Οι τυχαίοι παράγοντες που υπεισέρχονται στα πραγματικά βιολογικά συστήματα είναι πολλοί και καθοριστικοί. Για παράδειγμα, είναι πολύ πιθανό να χαθεί ένα μυρμήγκι κατά την αναζήτηση και συγκομιδή τροφής και τελικά να ανακαλύψει μια καινούργια καλύτερη πηγή τροφής «στρατολογώντας» και άλλα μέλη της αποικίας του για την εκμετάλλευση της πηγής αυτής.

Κάθε μοντέλο αυτό-οργάνωσης βασίζεται στην **αλληλεπίδραση μεταξύ των ατόμων του**. Κάθε άτομο-έντομο πρέπει να είναι σε θέση να δημιουργήσει κάποια σταθερή δομή (πχ ένα μονοπάτι φερομόνης<sup>6</sup>) αλλά και να ανιχνεύσει της δομές που δημιούργησαν άλλα άτομα της κοινότητάς του. Για να έχει αποτέλεσμα η αλληλεπίδραση θα πρέπει να υπάρχει ένας ελάχιστος πληθυσμός ατόμων-εντόμων, διαφορετικά η εμφάνιση οργανωμένων δομών μέσω της

---

<sup>5</sup> πχ ανταγωνισμός ή/και πολεμική σύρραξη με άλλη αποικία μυρμηγκιών ή εξάντληση της πηγής τροφής.

<sup>6</sup> Ως **μονοπάτι φερομόνης** (*pheromone trail*) ορίζεται κάθε διαδρομή στην οποία έχει εναποτεθεί κάποια ποσότητα φερομόνης από ένα ή/και περισσότερα μυρμήγκια.

συλλογικής δράσης είναι αδύνατη διότι υπάρχουν τα φαινόμενα αρνητικής ανάδρασης. Έχει ήδη αναφερθεί ότι η φερομόνη εξατμίζεται με κάποιο ρυθμό. Συνεπώς, αν ένα μυρμήγκι δημιουργήσει μια νέα επιτυχημένη διαδρομή αλλά ο πληθυσμός των μυρμηγκιών δεν είναι αρκετός ώστε να ενισχύσει τη διαδρομή αυτή, τότε το νέο αυτό μονοπάτι θα εγκαταλειφθεί.

Τέλος, μερικά από τα χαρακτηριστικά που παρατηρούνται σε συστήματα αυτο-οργάνωσης είναι:

- Η εμφάνιση συγκεκριμένων δομών σε αρχικώς ομοιογενή μέσα. Για παράδειγμα, η δημιουργία μονοπατιών φερομόνης από τα μυρμήγκια (βλ. 2.4) ή η κατασκευή φωλιάς με εναπόθεση σφαιριδίων χόματος από τους τερμίτες (βλ. 2.3).
- Η εν δυνάμει εμφάνιση πολλών αποδεκτών-σταθερών λύσεων (*multistability*) ανάλογα με τις αρχικές συνθήκες. Για παράδειγμα, έστω ότι υπάρχουν δύο τοποθεσίες τροφής A και B σε ίση απόσταση από τη φωλιά των μυρμηγκιών αλλά και με την ίδια ποιότητα και ποσότητα τροφής. Η τελική αξιοποίηση μιας εκ των δύο πηγών τροφής εξαρτάται κατά κύριο λόγο από τις αρχικές τυχαίες διακυμάνσεις. Αλλά και οι δύο λύσεις, δηλαδή η συγκομιδή τροφής είτε από την τοποθεσία A είτε από την B, είναι αποδεκτές.
- Η ύπαρξη δύο καταστάσεων συμπεριφοράς των εντόμων κατά τη διεκπεραίωση μιας συλλογικής εργασίας. Για παράδειγμα, η μετάβαση από την τυχαία εναπόθεση σφαιριδίων χόματος στην οργανωμένη κατά το κτίσιμο φωλιάς από τους τερμίτες (βλ. 2.3).

### 2.3 ΣΤΙΓΜΕΡΓΙΑ

Πολλές φορές τα κοινωνικά έντομα προκειμένου να επιτύχουν τις συλλογικές εργασίες τους πρέπει να επικοινωνήσουν είτε άμεσα είτε έμμεσα. Η άμεση επικοινωνία βασίζεται στην οπτική ή χημική επαφή των εντόμων, στις ανταλλαγές τροφής ή υγρών μεταξύ των εντόμων κτλ. Αντίθετα, η έμμεση επικοινωνία βασίζεται στις μεταβολές του περιβάλλοντος από τα έντομα και την ανίχνευση των μεταβολών αυτών από τα ίδια ή άλλα έντομα της αποικίας. Ένα χαρακτηριστικό παράδειγμα τέτοιας διεργασίας είναι η εναπόθεση φερομόνης από τα μυρμήγκια στο έδαφος κατά τη μεταφορά τροφής στη φωλιά. Η φερομόνη αυτή είναι εντοπίσιμη και από τα υπόλοιπα μέλη της αποικίας τα οποία δρουν ανάλογα.

Η παραπάνω διαδικασία έμμεσης επικοινωνίας των εντόμων χαρακτηρίζεται από τον όρο **στιγμεργία** (*stigmergy*) [50] ο οποίος προέρχεται από τις ελληνικές λέξεις<sup>7</sup> **στίγμα** (*stigma*) και **έργο** (*ergo*) και εισήχθη για πρώτη φορά από τον Grassé στο έργο του για τους τερμίτες *Bellicositermes Natalensis* και *Cubitermes*. Βασικά, στο έργο αυτό είχε ορίσει τον όρο στιγμεργία (*stigmergy*) ως εξής: «διέγερση των εργατών<sup>8</sup> από την απόδοση που έχουν επιτύχει». Ο Grassé παρατήρησε ότι τα έντομα διεγείρονται εκτελώντας μια γενετικά προκαθορισμένη αντίδραση όταν δεχτούν ένα σημαντικό ερέθισμα. Η αντίδραση αυτή μπορεί να αποτελεί ένα νέο σημαντικό ερέθισμα τόσο για το ίδιο το έντομο που την ενεργοποίησε αρχικά όσο και για τα υπόλοιπα μέλη της αποικίας του.

Πιο συγκεκριμένα, ο Grassé παρατήρησε τη διαδικασία δημιουργίας φωλιάς στους τερμίτες *Bellicositermes Natalensis* και *Cubitermes*. Η διαδικασία έχει ως εξής ([6], [23], [25]): αρχικά, οι τερμίτες εναποθέτουν τυχαία στην περιοχή που πρόκειται να χτιστεί η φωλιά τους χωμάτινα σφαιρίδια εμπλουτισμένα με την φερομόνη τους. Κάθε επόμενος τερμίτης ανιχνεύει τα σφαιρίδια αυτά και εναποθέτει το δικό του δίπλα τους. Με αυτό τον τρόπο σχηματίζονται σωροί από χωμάτινα σφαιρίδια. Όταν για τυχαίους λόγους κάποιος από αυτούς τους σωρούς ξεπεράσει ένα ορισμένο μέγεθος, δηλαδή υπάρξει μια συγκεκριμένη συσσώρευση ποσότητας φερομόνης, τότε το γεγονός αυτό αποτελεί ένα νέο σημαντικό ερέθισμα που κινητοποιεί τους τερμίτες να εναποθέσουν τα επόμενα σφαιρίδια στο σωρό αυτό ώστε να σχηματιστεί μια στήλη. Αργότερα, θα σχηματιστούν τα τόξα που ενώνουν τις στήλες μεταξύ τους και τελικά θα δημιουργηθεί η φωλιά. Αν όμως ο πληθυσμός των τερμιτών δεν είναι αρκετά μεγάλος, τότε λόγω της εξάτμισης της φερομόνης δεν συσσωρεύεται ποτέ η απαιτούμενη ποσότητα σφαιριδίων (και άρα φερομόνης) που να ενεργοποιεί την επόμενη φάση στο κτίσιμο της φωλιάς.

Για να κάνουμε μια παράθεση του παραδείγματος αυτού με το Κεφάλαιο 2.2, θα επιστημόνουμε τα εξής [23]: υπάρχει θετική ανάδραση (η δημιουργία σωρών από τα χωμάτινα σφαιρίδια και στη συνέχεια στηλών από τους σωρούς), αρνητική ανάδραση (εξάτμιση φερομόνης), ενίσχυση των τυχαίων διακυμάνσεων (οι στήλες μπορούν να δημιουργηθούν οπουδήποτε), έμμεση αλληλεπίδραση των ατόμων της αποικίας (στιγμεργία), η δημιουργία δομών (στήλες) μέσα σε ένα ομοιογενές μέσο (το έδαφος με αρχικά τυχαία ομοιόμορφη κατανομή σφαιριδίων με φερομόνη), πολλαπλές αποδεκτές λύσεις (οι στήλες μπορούν να

---

<sup>7</sup> Προφανώς, η επιλογή των συγκεκριμένων ελληνικών λέξεων από τον Grassé δεν ήταν η καταλληλότερη.

<sup>8</sup> Οι **εργάτες** (*workers*) είναι μια από τις κοινωνικές τάξεις (*castes*) των τερμιτών, οι άλλες δύο είναι οι πολεμιστές (*warriors*) και η τάξη που ασχολείται με την αναπαραγωγή (*reproductives*).

εμφανιστούν οπουδήποτε) και, τέλος, μετάβαση από τη μια κατάσταση (τυχαία εναπόθεση σφαιριδίων) στην άλλη (οργανωμένη εναπόθεση σφαιριδίων).

Οι Bonabeau, Dorigo και Theraulaz στο [23] συμπλήρωσαν τον ορισμό του Grassé και του έδωσαν την εξής μορφή: Στιγμεργετική (*stigmergetic*) επικοινωνία είναι κάθε έμμεση επικοινωνία η οποία πραγματοποιείται με φυσικές μεταβολές στο περιβάλλον οι οποίες είναι μόνο τοπικά προσπελάσιμες από τα επικοινωνούντα άτομα (*agents*).

Τέλος, θα πρέπει να επισημανθεί ότι η στιγμεργετική επικοινωνία συμβάλλει στην ευελιξία ή/και προσαρμοστικότητα του συστήματος στο περιβάλλον. Κάθε εξωγενής αλλαγή στις παραμέτρους του περιβάλλοντος μπορεί να ληφθεί από τα άτομα της αποικίας ως ενδογενής αλλαγή από κάποια άλλα άτομα της αποικίας και να αντιμετωπισθεί ανάλογα<sup>9</sup>.

#### 2.4 ΑΝΑΖΗΤΗΣΗ ΚΑΙ ΣΥΓΚΟΜΙΔΗ ΤΡΟΦΗΣ ΣΤΙΣ ΑΠΟΙΚΙΕΣ ΜΥΡΜΗΓΚΙΩΝ

Έχει ήδη αναφερθεί ότι τα μυρμήγκια έχουν τη δυνατότητα να δημιουργούν μονοπάτια με φερομόνη (*pheromone trails*) όταν ψάχνουν για τοποθεσίες με τροφή ή όταν επιστρέφουν στη φωλιά μεταφέροντας τροφή ή και τα δύο<sup>10</sup> ανάλογα με το είδος του μυρμηγκιού [29]. Τα άλλα μυρμήγκια της αποικίας έχουν τη δυνατότητα να ανιχνεύουν τα μονοπάτια αυτά και να αποφασίσουν<sup>11</sup> με κάποια πιθανότητα αν θα τα ακολουθήσουν.

Η διαδικασία κατά την οποία τα μυρμήγκια προτρέπονται από κάποια άλλα να εκμεταλλευτούν μια συγκεκριμένη πηγή τροφής ονομάζεται **στρατολόγηση** (*recruitment*). Όταν αυτή η διαδικασία βασίζεται μόνο σε χημικά ίχνη (φερομόνη) τότε ονομάζεται **μαζική στρατολόγηση** (*mass recruitment*).

Γενικά, η διαδικασία δημιουργίας βέλτιστης διαδρομής μεταξύ δύο σημείων έχει ως εξής (Εικόνα 1 και Εικόνα 2): έστω δύο σημεία T (τοποθεσία τροφής) και Φ (τοποθεσία φωλιάς) που συνδέονται με δύο διαδρομές μια μικρού και μια μεγάλου μήκους και έστω ότι τα μυρμήγκια πρόκειται να κινηθούν μεταξύ αυτών των δύο σημείων. Τα μυρμήγκια ξεκινώντας την πορεία τους επιλέγουν μια από τις δύο διαδρομές εντελώς τυχαία. Όταν τα μυρμήγκια που έχουν φτάσει στην τροφή μέσω της σύντομης διαδρομής πρόκειται να επιστρέψουν στη φωλιά, τότε

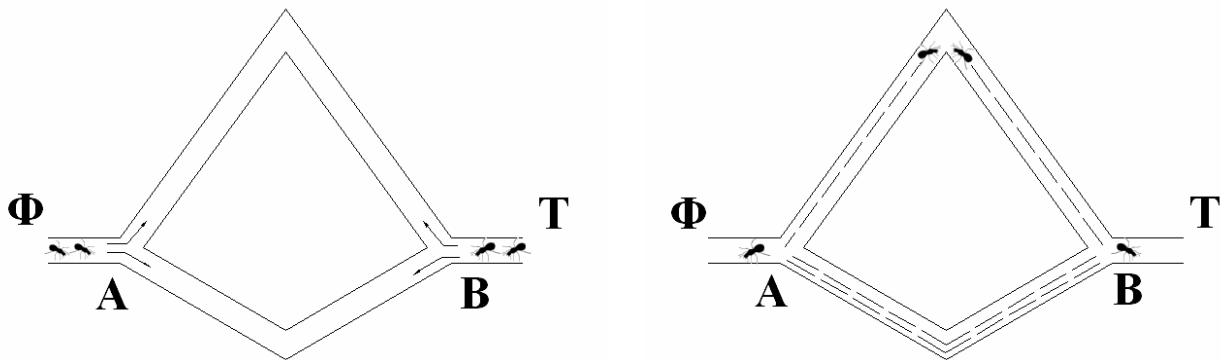
---

<sup>9</sup> Αυτό βεβαίως διευκολύνει το προγραμματισμό τέτοιων μεθόδων καθώς δεν χρειάζεται να λαμβάνονται υπόψη υποπεριπτώσεις αλλά όλες αντιμετωπίζονται από το βασικό αλγόριθμο.

<sup>10</sup> Χωρίς να αποκλείονται και άλλες περιπτώσεις πχ να εναποθέτουν λίγη φερομόνη κατά την ανίχνευση τροφής και πολύ κατά την επιστροφή στη φωλιά με τροφή ή το ποσό της φερομόνης που εναποθέτουν να είναι ανάλογο με την ποσότητα ή/και την ποιότητα της τροφής.

<sup>11</sup> Προφανώς σε κάποια επίπεδα συγκεντρωμένης φερομόνης οι πράκτορες (*agents*) δέχονται ένα σημαντικό ερέθισμα που τους προτρέπει να ακολουθήσουν το μονοπάτι αυτό.

χρησιμοποιούν και πάλι τη σύντομη διαδρομή καθώς αυτή είναι εμπλουτισμένη με τη φερομόνη που έχουν ήδη εναποθέσει. Επιπλέον, τα μυρμήγκια που φτάνουν στην τροφή έχοντας ακολουθήσει τη μεγάλη διαδρομή όταν πρόκειται να αποφασίσουν πια πορεία θα επιλέξουν για την επιστροφή τους θα έχουν μεγαλύτερη πιθανότητα να επιλέξουν τη σύντομη διαδρομή η οποία και θα έχει περισσότερη φερομόνη (αν αγνοήσουμε το ότι μπορεί να διαθέτουν και περιορισμένη μνήμη ή και άλλες μεθόδους προσανατολισμού). Προφανώς, τα νέα μυρμήγκια που ξεκινούν από τη φωλιά θα επιλέγουν με μεγαλύτερη πιθανότητα τη μικρού μήκους διαδρομή. Τελικά, με αυτόν τον τρόπο, δηλαδή με τη βοήθεια της θετικής ανάδρασης (εναπόθεση φερομόνης), τα μυρμήγκια επιτυγχάνουν να επιλέξουν την συντομότερη (βέλτιστη) διαδρομή.



Εικόνα 1. Έστω ότι τα μυρμήγκια κινούνται από το Φ στο T και αντίστροφα. Όταν φτάσουν στη διακλάδωση A ή B θα πρέπει να επιλέξουν μεταξύ της μικρής και της μεγάλης διαδρομής. Η επιλογή γίνεται τυχαία καθώς και στις δυο διαδρομές δεν υπάρχει φερομόνη.

Εικόνα 2. Οι διακεκομμένες γραμμές δείχνουν την κατανομή των επιπέδων της φερομόνης αφού τα μυρμήγκια που επέλεξαν τη σύντομη διαδρομή ολοκληρώσουν την μετακίνηση από το A στο B και αντίστροφα. Είναι φανερό ότι τα νέα μυρμήγκια που θα φτάσουν στη διακλάδωση A ή B θα επιλέξουν τη σύντομη διαδρομή με μεγαλύτερη πιθανότητα.

Ο Deneubourg με τους συνεργάτες του πραγματοποίησαν μια σειρά πειραμάτων [2], [29] με τα οποία αποδεικνύουν πειραματικά την παραπάνω διαδικασία και μάλιστα καταλήγουν και σε κάποια θεωρητικά μοντέλα τα οποία και επιβεβαιώνουν με προσομοιώσεις Monte-Carlo. Παρακάτω, θα παρουσιάσουμε συνοπτικά μερικά από τα πιο σημαντικά αποτελέσματα και συμπεράσματα αυτών των μελετών. Πιο συγκεκριμένα, ο Deneubourg εργάστηκε με τα μυρμήγκια *Iridomyrmex humilis* από την Αργεντινή τα οποία έχουν περιορισμένες ικανότητες προσανατολισμού και επιπλέον η φερομόνη τους έχει σχετικά μεγάλο χρόνο ημιζωής ( $\approx 30\text{min}$ )

σε σχέση με τη συνολική χρονική διάρκεια των πειραμάτων με συνέπεια να αγνοείται η επίδρασή της (δηλαδή δεν υφίσταται στο μοντέλο αρνητική ανάδραση).

Στο [18] αποδείχθηκε ότι όταν οι δύο διαδρομές που συνδέουν τη φωλιά με την τοποθεσία της τροφής είναι ίσης απόστασης (Εικόνα 3), τότε ενώ αρχικά τα μυρμήγκια επιλέγουν με πιθανότητα 50% μια από τις δύο διαδρομές, αργότερα λόγω τυχαίων διακυμάνσεων (που συνεπάγονται αύξηση της φερομόνης σε μια από τις δυο διαδρομές) καταλήγουν να χρησιμοποιούν σχεδόν όλα τα μυρμήγκια τη μια από αυτές τις διαδρομές (Εικόνα 4). Το συγκεκριμένο πείραμα δείχνει το αποτέλεσμα της θετικής ανάδρασης και το ρόλο που παίζουν οι τυχαίες διακυμάνσεις.

Στο [29] οι συγγραφείς πραγματοποίησαν παρόμοια πειράματα με το προηγούμενο χρησιμοποιώντας διαδρομές με διαφορετικά μήκη (Εικόνα 5). Κατέληξαν στο συμπέρασμα ότι η μικρού μήκους διαδρομή είναι πιθανότερο να επιλεγθεί όσο μεγαλύτερος είναι ο πληθυσμός των μυρμηγκιών και όσο μεγαλύτερος είναι ο λόγος των μηκών των δύο διαδρομών – εκτός και αν τυχαίος θόρυβος καθυστερεί ή αλλοιώνει τη διαδικασία επιλογής της συντομότερης διαδρομής. Επίσης, σημαντικό είναι το γεγονός ότι αν η μικρού μήκους διαδρομή παρουσιασθεί στα μυρμήγκια μετά από ένα ορισμένο χρονικό διάστημα και αφού έχουν κατασταλάξει στη μακρά διαδρομή τότε είναι σχεδόν αδύνατο τα μυρμήγκια να εντοπίσουν και να αξιοποιήσουν τη νέα συντομότερη διαδρομή. Η κατάσταση αυτή στην οποία τα μυρμήγκια δεν είναι πλέον ικανά να εντοπίσουν νέες συντομότερες διαδρομές ονομάζεται **στασιμότητα** (*stagnation*). Προφανώς, για να αποφευχθούν τέτοιες καταστάσεις πρέπει να υιοθετηθούν άλλοι μηχανισμοί, όπως για παράδειγμα η ταχεία εξάτμιση της φερομόνης (αρνητική ανάδραση)<sup>12</sup>.

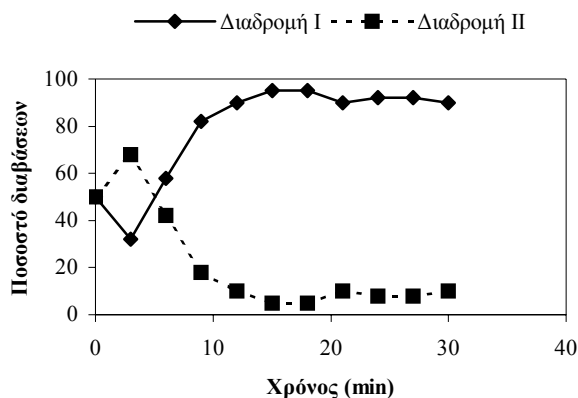
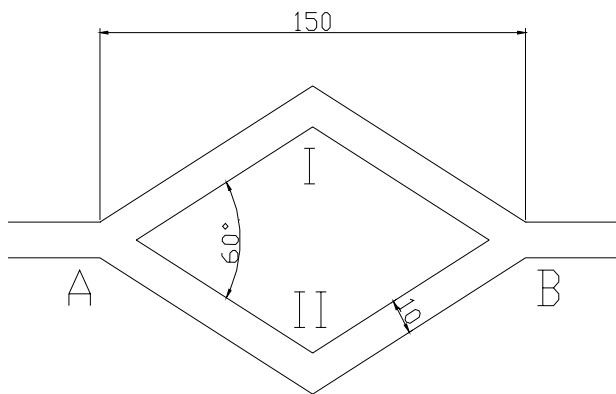
Ένα άλλο φυσικό παράδειγμα είναι η συμπεριφορά των μυρμηγκιών *Lasius niger* [2] τα οποία συγκρατούν στη μνήμη τους την τοποθεσία της τροφής και μπορούν να αντιληφθούν αν κινούνται σε μεγάλη γωνία ως προς την επιθυμητή πορεία. Έτσι, λοιπόν, αν το ερέθισμα (επίπεδα φερομόνης) δεν είναι ικανοποιητικό ώστε να συνεχίσουν την πορεία τους στο συγκεκριμένο μονοπάτι, κάνουν αναστροφή και επιστρέφουν στο αρχικό σημείο της διακλάδωσης αναζητώντας μια καλύτερη διαδρομή.

Επίσης, τα μυρμήγκια *Myrmica rubra* [2], αξιοποιώντας ένα παρόμοιο μηχανισμό με τον παραπάνω, εναποθέτουν μικρότερη ποσότητα φερομόνης όσο μεγαλύτερη είναι η απόκλισή τους

---

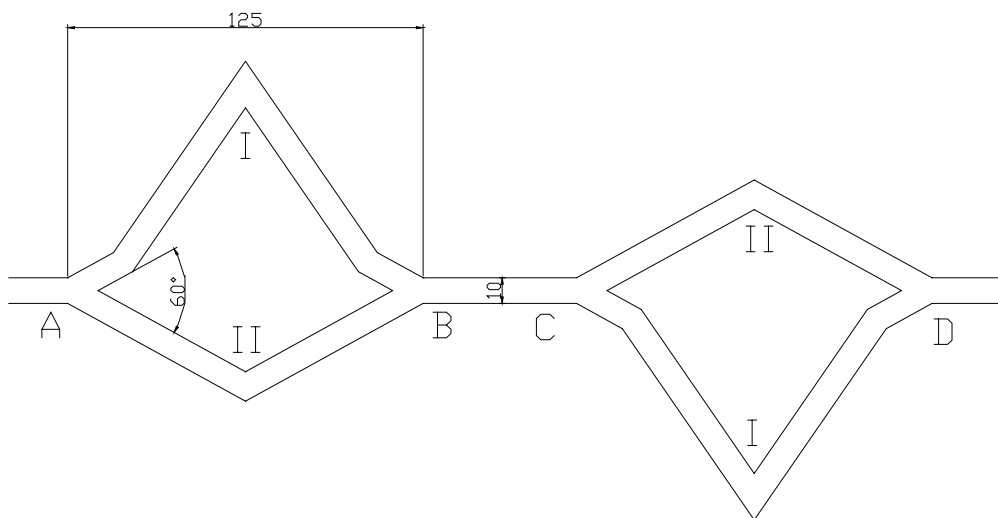
<sup>12</sup> Αν και στα πραγματικά βιολογικά συστήματα η εξάτμιση της φερομόνης είναι αργή, στα προγραμματιστικά μοντέλα που δημιουργούνται κάτι τέτοιο μπορεί να αποφευχθεί.

από το βασικό νοητό άξονα που συνδέει τη φωλιά με την τοποθεσία της τροφής. Με τον τρόπο αυτό ευνοούνται οι διαδρομές που βρίσκονται κοντά στην ευθεία φωλιάς-πηγής τροφής.

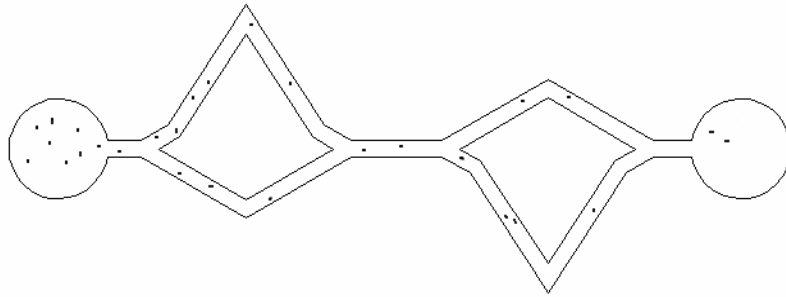


Εικόνα 3. Η πειραματική διάταξη (γέφυρα) του [18] που τοποθετήθηκε μεταξύ της φωλιάς και της πηγής τροφής.

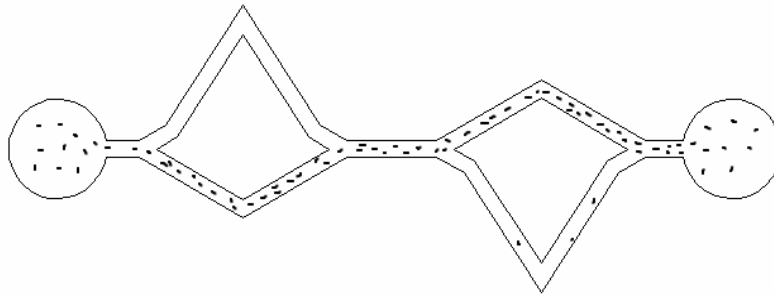
Εικόνα 4. Το ποσοστό των μυρμηγκιών που χρησιμοποίησαν τις διαδρομές I και II.



Εικόνα 5. Η πειραματική διάταξη (γέφυρα) που χρησιμοποιήθηκε στο [29]. Παρατηρούμε ότι η είσοδος (η οποία είναι και έξοδος κατά την αντίστροφη κίνηση) κάθε διαδρομής σε όλες τις διακλαδώσεις είναι στις 30° σε σχέση με την κύρια διαδρομή. Με τον τρόπο αυτό, τα μυρμήγκια κατά την έναρξη του πειράματος δείχνουν την ίδια προτίμηση και για τις δύο διαδρομές σε κάθε διακλάδωση. Επίσης, με την γωνία των 60° που σχηματίζεται αποφεύγονται κατά το δυνατόν οι αναστροφές των μυρμηγκιών (δηλαδή ενώ ένα μυρμήγκι κινείται από το A στο B μέσω της διαδρομής I μόλις φτάσει στο B να κινηθεί από το B στο A μέσω της διαδρομής II).



Εικόνα 6. Σχηματική κατανομή των μυρμηγκιών *Iridomyrmex humilis* σε μικρό χρονικό διάστημα από την έναρξη του πειράματος.



Εικόνα 7. Σχηματική κατανομή των μυρμηγκιών *Iridomyrmex humilis* σε σχετικά μεγάλο χρονικό διάστημα από την έναρξη του πειράματος.



### 3 ΑΠΟ ΤΑ ΠΡΑΓΜΑΤΙΚΑ ΣΤΑ «ΨΗΦΙΑΚΑ» ΜΥΡΜΗΓΚΙΑ

Προκειμένου να μετατρέψουμε τα φυσικά βιολογικά μοντέλα σε τεχνητά «αλγοριθμικά», δηλαδή σε υπολογιστικά εργαλεία για την επίλυση προβλημάτων αριστοποίησης, θα πρέπει να κάνουμε ορισμένες παραδοχές. Προφανώς, τα «ψηφιακά» μυρμήγκια<sup>13</sup> θα αποτελούν μια απλοποίηση των πραγματικών μυρμηγκιών αλλά και θα εμπλουτιστούν με επιπρόσθετες ιδιότητες που δεν εμφανίζονται στα φυσικά μυρμήγκια και θα καθορίζονται από το αντίστοιχο πρόβλημα προς λύση ή από τις ανάγκες του χρήστη για γρήγορη σύγκλιση κτλ.

Οι ομοιότητες των «ψηφιακών» μυρμηγκιών με τα φυσικά έχουν ως εξής [23]:

**Η αποικία.** Όπως στα φυσικά μυρμήγκια έτσι και στα «ψηφιακά» υπάρχει μια αποικία ατόμων τα οποία συνεργάζονται προκειμένου να δημιουργήσουν μια συγκεκριμένη δομή στο σύστημα τους. Η δομή αυτή στα υπό μελέτη προβλήματα στην παρούσα εργασία αντιστοιχεί στην εύρεση βέλτιστης λύσης. Ο πληθυσμός των «ψηφιακών» μυρμηγκιών είναι μια παράμετρος η οποία είναι σε άμεση συνάρτηση με το είδος του προβλήματος.

**Η φερομόνη.** Ορισμένες «ποικιλίες» πραγματικών μυρμηγκιών εναποθέτουν φερομόνη στο έδαφος κατά την κίνησή τους. Ομοίως και τα «ψηφιακά» μυρμήγκια θα εναποθέτουν στις διάφορες θέσεις τις οποίες επισκέπτονται μια συγκεκριμένη αριθμητική πληροφορία, την «ψηφιακή» φερομόνη<sup>14</sup>, οπότε και θα δημιουργούνται **μονοπάτια φερομόνης** (*pheromone trails*). Αυτή η αριθμητική πληροφορία θα εξαρτάται από την επίδοση<sup>15</sup> και την προηγούμενη ιστορία του «ψηφιακού» μυρμηγκιού και θα είναι προσβάσιμη και από τα υπόλοιπα μέλη της αποικίας εγκαθιδρύοντας με αυτόν τον τρόπο στιγμαρκετική (*stigmergetic*) επικοινωνία. Τέλος, η «ψηφιακή» φερομόνη όπως και η τεχνητή θα πρέπει να υφίσταται εξάτμιση έτσι ώστε να υπάρχει αρνητική ανάδραση και να μην εγκλωβίζονται τα άτομα σε τοπικά βέλτιστα.

**Η απόφαση.** Όπως τα φυσικά έτσι και τα «ψηφιακά» μυρμήγκια αποφασίζουν την επόμενη κίνησή τους με κάποια στοχαστικότητα. Τα δεδομένα που έχουν στη διάθεσή τους είναι καθαρά τοπικά στο χώρο και το χρόνο. Αναλυτικότερα, τα μυρμήγκια, είτε πραγματικά είτε «ψηφιακά»,

---

<sup>13</sup> Στη βιβλιογραφία ο αντίστοιχος όρος είναι **artificial ants** (τεχνητά μυρμήγκια) – συνήθως, στο κείμενο της εργασίας τα «ψηφιακά» μυρμήγκια θα αναφέρονται απλά ως *μυρμήγκια* ή *άτομα*.

<sup>14</sup> **Artificial pheromone** – ακριβής μετάφραση: *τεχνητή φερομόνη* – στην παρούσα εργασία προτιμούμε την αδόκιμη χρήση του όρου «ψηφιακή» φερομόνη.

<sup>15</sup> Όπως έχει ήδη αναφερθεί υπάρχουν ποικιλίες μυρμηγκιών που εναποθέτουν φερομόνη ανάλογα με την ποσότητα και την ποιότητα της τροφής που έχουν εντοπίσει.

δεν έχουν γνώση του τι συμβαίνει σε κάποιο άλλο σημείο όπου δραστηριοποιούνται μυρμήγκια τις αποικίας αλλά ούτε και τη γενικότερη κατάσταση της αποικίας. Επίσης, δεν γνωρίζουν τι έγινε τις προηγούμενες χρονικές στιγμές στο σημείο όπου βρίσκονται αλλά ούτε μπορούν να προβλέψουν μελλοντικές καταστάσεις.

Τα τοπικά δεδομένα θα μπορούσαν να χωριστούν σε δύο κατηγορίες: (α) τα δεδομένα που είναι γνωστά εκ των προτέρων (πχ. στο φυσικό μοντέλο η μορφολογία του εδάφους ενώ στο ψηφιακό η **ευριστική πληροφορία**<sup>16</sup> – *heuristic information*) και (β) τα δεδομένα από τις τοπικές αλλαγές στο περιβάλλον (πχ τα μονοπάτια φερομόνης).

Ο Deneubourg θέλοντας να επιβεβαιώσει τα αποτελέσματα των πειραμάτων του (βλέπε Κεφάλαιο 2.4), δημιούργησε ένα μαθηματικό μοντέλο [18] της συμπεριφοράς των πραγματικών μυρμηγκιών του οποίου οι εξισώσεις διέπουν την **απόφαση** της επιλογή μιας εκ των δύο διαδρομών σε μια διακλάδωση (πχ στη διακλάδωση Α Εικόνα 1).

Έστω  $L_i$  και  $R_i$  τα μυρμήγκια που έχουν επιλέξει την αριστερή και τη δεξιά διαδρομή αντίστοιχα μετά το πέρας της κίνησης του  $i$ -οστού μυρμηγκιού (δηλαδή  $i = L_i + R_i$  ο συνολικός αριθμός των μυρμηγκιών που έχουν περάσει το σημείο επιλογής διακλάδωσης Α). Τότε η πιθανότητα  $P_L$  το επόμενο μυρμήγκι ( $i+1$ ) να επιλέξει την αριστερή διαδρομή δίνεται από τη σχέση:

$$P_L = \frac{(k + L_i)^n}{(k + L_i)^n + (k + R_i)^n} = 1 - P_R \quad \text{Εξ. 1}$$

όπου:

$n$  η παράμετρος που καθορίζει το βαθμό μη γραμμικότητας της συνάρτησης επιλογής. Μεγάλη τιμή του  $n$  συνεπάγεται ότι μικρή διαφοροποίηση στα επίπεδα φερομόνης στις δύο διαδρομές αυξάνει κατά πολύ την πιθανότητα επιλογής ενός κλάδου εκ των δύο από το επόμενο μυρμήγκι.

$k$  η παράμετρος που ποσοτικοποιεί το βαθμό ελκυστικότητας μιας διαδρομής που δεν έχει εναποτεθεί φερομόνη. Όσο μεγαλύτερη τιμή έχει η παράμετρος  $k$  τόσο μεγαλύτερη ποσότητα φερομόνης απαιτείται ώστε η επιλογή ενός κλάδου να μην είναι πλέον τυχαία.

Η επιλογή μιας εκ των διακλαδώσεων γίνεται ως εξής:

---

<sup>16</sup> Για παράδειγμα, η *ευριστική πληροφορία* στο πρόβλημα TSP είναι συνάρτηση της απόστασης των πόλεων, ενώ στο IDS της διαφοράς πίεσεων μεταξύ του στόχου και της υπό δοκιμή λύσης.

$$L_{i+1} = \begin{cases} L_i + 1 & \alpha\nu \delta \leq P_L \\ L_i & \alpha\nu \delta > P_L \end{cases} \text{ και } R_{i+1} = \begin{cases} R_i + 1 & \alpha\nu \delta > P_L \\ R_i & \alpha\nu \delta \leq P_L \end{cases} \quad \text{Εξ. 2}$$

όπου  $\delta$  είναι μια τυχαία μεταβλητή ομοιόμορφα κατανομημένη στο διάστημα  $[0, 1]$ .

Ενδεικτικά θα αναφέρουμε ότι στο [29] χρησιμοποιούνται οι παράμετροι  $n=2$  και  $k=20$  (Εικόνα 5). Οι παράμετροι αυτοί μετά από εξομοιώσεις με τη μέθοδο Monte-Carlo έδωσαν αποτελέσματα που είναι σε πλήρη αρμονία με τα πειραματικά.

Ιδιαίτερη προσοχή πρέπει να δοθεί στην Εξ. 1 καθώς θα χρησιμοποιηθεί παρακάτω σε μια διαφορετική μορφή για την υλοποίηση των αλγορίθμων που βασίζονται στις αποικίες των μυρμηγκιών (Ant Colony Optimization).

Οι **επιπλέον ικανότητες** που προαιρετικά μπορούν να προστεθούν στα «ψηφιακά» μυρμήγκια έχουν ως εξής:

**Ύπαρξη «μνήμης»:** για να θυμούνται τις προηγούμενες πράξεις τους (πχ Tabu list στο TSP) ή άλλες χρήσιμες πληροφορίες.

**Ύπαρξη «όρασης» ή άλλων «αισθητηρίων»:** ώστε να εκμεταλλεύονται καλύτερα τις πληροφορίες του περιβάλλοντος (πχ. να μπορούν να υπολογίσουν εκ των προτέρων τις αποστάσεις των πόλεων στο TSP και όχι αφού έχουν ολοκληρώσει μια διαδρομή).

**Διακριτοποίηση του χώρου και του χρόνου:** για την επίλυση διακριτών προβλημάτων. Για παράδειγμα, οι κινήσεις των μυρμηγκιών στο TSP είναι διακριτές από πόλη σε πόλη και πραγματοποιούνται σε διακριτές χρονικές στιγμές.

**Εναπόθεση φερομόνης σε διαφορετικές χρονικές στιγμές:** ενώ τα φυσικά μυρμήγκια εναποθέτουν συνεχώς φερομόνη, τα «ψηφιακά» το κάνουν συνήθως αφού ολοκληρώσουν τις κινήσεις<sup>17</sup> τους.

**Χρησιμοποίηση αλγορίθμων για τη βελτίωση της απόδοσης του συστήματος:** μπορεί να χρησιμοποιηθούν αλγόριθμοι τοπικής βελτιστοποίησης κτλ.

---

<sup>17</sup> Ο Dorigo στο [19] εξετάζει δύο περιπτώσεις εναπόθεσης φερομόνης από τα μυρμήγκια στο πρόβλημα TSP: εναπόθεση φερομόνης μετά από κάθε βήμα (= μετακίνηση από πόλη σε πόλη) του μυρμηγκιού και εναπόθεση φερομόνης από το μυρμήγκι αφού έχει ολοκληρώσει το ταξίδι του (= έχει επισκευφθεί όλες τις πόλεις). Από τα αποτελέσματα προκύπτει ότι η δεύτερη διαδικασία βελτιώνει σαφέστατα την απόδοση του αλγορίθμου.



## 4 ΔΙΑΚΡΙΤΟΙ ΑΛΓΟΡΙΘΜΟΙ ΒΕΛΤΙΣΤΟΠΟΙΗΣΗΣ

### 4.1 ANT SYSTEM (AS)

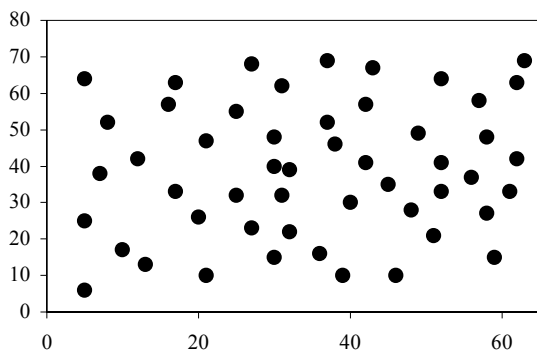
Ο πρώτος αλγόριθμος που αναπτύχθηκε με βάση τις αρχές του Κεφαλαίου 3 είναι ο Ant System (AS) του Dorigo [6], [19]. Ο αλγόριθμος αυτός χρησιμοποιήθηκε για πρώτη φορά για την επίλυση του **προβλήματος του περιπλανώμενου πωλητή** (Traveling Salesman Problem - TSP) καθώς το πρόβλημα αυτό είναι συναφές με τη πραγματική συμπεριφορά των μυρμηγκιών κατά την ανίχνευση και συγκομιδή τροφής, δηλαδή της εύρεσης της γρηγορότερης διαδρομής. Για το λόγο αυτό ο αλγόριθμος AS θα παρουσιασθεί στη μορφή για την επίλυση του TSP.

Πιο συγκεκριμένα, στο TSP το ζητούμενο είναι η εύρεση της συντομότερης διαδρομής που συνδέει  $n$  πόλεις. Οι περιορισμοί που τίθενται είναι:

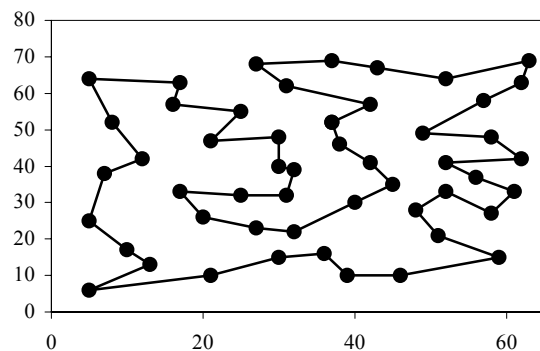
(α) κάθε πόλη πρέπει να περιλαμβάνεται μια φορά στη διαδρομή

(β) η διαδρομή πρέπει να είναι κλειστή, δηλαδή ο «πωλητής» πρέπει να επιστρέψει στην πόλη από όπου ξεκίνησε.

Οι αποστάσεις μεταξύ των πόλεων δεν είναι απαραίτητο να είναι συμμετρικές καθώς η απόσταση από την πόλη  $i$  στη  $j$  μπορεί να είναι διαφορετική από την απόσταση από την  $j$  πόλη στην  $i$  (τότε ορίζεται το ATSP: Asymmetric Traveling Salesman Problem).



Εικόνα 8. Η κατανομή των πόλεων στο πρόβλημα Eilon's 51 – town TSP.



Εικόνα 9. Η βέλτιστη διαδρομή του προβλήματος Eilon's 51 – town TSP.

Η κάθε λύση για το TSP σχηματίζεται με τη διαδοχική μετάβαση των «ψηφιακών» μυρμηγκιών από τη μια πόλη στην άλλη με κάποια πιθανότητα. Το «ταξίδι» ολοκληρώνεται όταν το «ψηφιακό» μυρμήγκι επιστέψει στην αρχική πόλη. Τότε βαθμολογείται η λύση που έχει επιτύχει με κριτήριο το συνολικό μήκος της διαδρομής και προστίθεται η ανάλογη φερομόνη στο μονοπάτι που ακολούθησε. Η παραπάνω διαδικασία επαναλαμβάνεται για κάθε άτομο της αποικίας μέχρι να συμπληρωθεί ο ζητούμενος αριθμός επαναλήψεων<sup>18</sup>.

Αναλυτικότερα, ο αλγόριθμος σχηματίζεται με την παρακάτω διαδικασία [6]. Έστω  $m$  ο αρχικός πληθυσμός των «ψηφιακών» μυρμηγκιών. Μετά από πειράματα [20], ο Dorigo διαπίστωσε ότι  $m=n$ , δηλαδή ο αρχικός πληθυσμός πρέπει να είναι ίσος με τον αριθμό των πόλεων του προβλήματος. Αν είναι πολύ μεγάλος ο πληθυσμός της αποικίας τότε πραγματοποιείται σύγκλιση σε τοπικά βέλτιστα, ενώ αν είναι πολύ μικρός δεν λειτουργεί η στιγμεργετική επικοινωνία μεταξύ των μυρμηγκιών.

Τα μυρμήγκια τοποθετούνται είτε τυχαία στις πόλεις είτε το καθένα σε μια διαφορετική πόλη ως αφετηρία της διαδρομής τους. Η πόλη αυτή καταγράφεται ως νούμερο «ένα» στη **λίστα της μνήμης** (*tabu list*) κάθε μυρμηγκιού. Η *tabu list* είναι μια λίστα που περιέχει τις πόλεις που έχει ήδη επισκεφθεί το μυρμήγκι έτσι ώστε αυτές να αποκλειστούν από τις επόμενες δυνατές επιλογές του.

Κάθε μυρμήγκι  $k$  κατά την επανάληψη  $t$  που βρίσκεται στην πόλη  $i$  επιλέγει την επόμενη πόλη  $j$  που θα επισκεφθεί με βάση τον **τυχαίο-αναλογικό κανόνα μετάβασης**<sup>19</sup> (*random-proportional transition rule*):

$$p_{ij}^k(t) = \frac{[\tau_{ij}(t)]^\alpha \cdot [\eta_{ij}(t)]^\beta}{\sum_{l \in J_i^k} [\tau_{il}(t)]^\alpha \cdot [\eta_{il}(t)]^\beta} \quad \alpha \nu \quad j \notin J_i^k$$

$$0 \quad \alpha \nu \quad j \in J_i^k$$
Εξ. 3

όπου:

$J_i^k$  είναι οι πόλεις που έχει ήδη επισκεφθεί το μυρμήγκι  $k$  όταν βρίσκεται στην πόλη  $i$  (*Tabu list*).

<sup>18</sup> Μια **επανάληψη** (*iteration*) πραγματοποιείται όταν όλα τα μυρμήγκια ολοκληρώσουν τα **ταξίδια** τους (*tours*). Επίσης, η ολοκλήρωση μιας επανάληψης ισοδυναμεί με τη διέλευση μιας χρονικής μονάδας. Συνεπώς, όταν στο κείμενο εμφανίζεται η χρονική στιγμή  $t$  εννοείται ότι έχουν ολοκληρωθεί  $t$  επαναλήψεις.

<sup>19</sup> Εκφράζει την πιθανότητα το μυρμήγκι  $k$  να μετακινηθεί από την πόλη  $i$  στην  $j$  τη στιγμή  $t$ .

$\eta_{ij}(t)$  είναι η **ορατότητα** (*visibility*) και ορίζεται ως το αντίστροφο της απόστασης μεταξύ των πόλεων  $i$  και  $j$ , δηλαδή  $\eta_{ij}=1/d_{ij}$ . Η ορατότητα εκφράζει την **ευριστική προτίμηση** (*heuristic desirability*) ως επόμενου σταθμού της πόλης  $j$  όταν το μυρμήγκι βρίσκεται στην πόλη  $i$  και βασίζεται καθαρά σε τοπικές πληροφορίες. Προφανώς, η πληροφορία αυτή είναι αναλλοίωτη στο TSP καθώς δεν μεταβάλλεται κατά την επίλυση του προβλήματος, δηλαδή  $\eta_{ij}(t)=\eta_{ij}$ .

$\tau_{ij}(t)$  είναι η ποσότητα της φερομόνης που βρίσκεται στην ακμή που ενώνει τις πόλεις  $i$  και  $j$ . Η πληροφορία αυτή έχει γενικό χαρακτήρα καθώς εκφράζει την τρέχουσα γνώση από την «εμπειρία» όλης της αποικίας. Κατά τη χρονική στιγμή  $t=0$  (πρώτη επανάληψη), η φερομόνη τίθεται σε κάθε ακμή  $ij$  σε μια αρχική τιμή  $\tau_0$  πάρα πολύ μικρή.

$\alpha, \beta$  είναι παράμετροι που καθορίζονται από τον χρήστη και εξαρτώνται από το πρόβλημα προς επίλυση. Όταν το  $\alpha=0$  τότε ο αλγόριθμος δεν χρησιμοποιεί την εμπειρία της αποικίας (δηλαδή τη φερομόνη) και επομένως εκφυλλίζεται σε έναν **πρακτικό στοχαστικό** (*greedy stochastic*) αλγόριθμο με πολλαπλά σημεία εκκίνησης, αφού κάθε μυρμήγκι ξεκινάει το ταξίδι του από διαφορετική πόλη. Αν  $\beta=0$  τότε η αναζήτηση της βέλτιστης λύσης στηρίζεται μόνο στη θετική ανάδραση και όπως έχει ήδη αναφερθεί (Κεφάλαιο 2.2) ο αλγόριθμος θα συγκλίνει γρήγορα σε τοπικά βέλτιστα.

Αφού όλα τα μυρμήγκια ολοκληρώσουν τα «ταξίδια» τους κατά την επανάληψη  $t$ , τότε προστίθεται φερομόνη σε κάθε μονοπάτι ανάλογα με την επίδοση του κάθε μυρμηγκιού. Σημαντικός παράγοντας στην όλη διαδικασία είναι και η αρνητική ανάδραση η οποία υλοποιείται με την εξάτμιση της φερομόνης. Κατά αυτόν τον τρόπο αποφεύγεται η ενίσχυση των αρχικών τυχαίων διακυμάνσεων.

Η σχέση που διέπει την εναπόθεση – και την ανανέωση – της φερομόνης κατά τη χρονική στιγμή  $t$  (που ουσιαστικά είναι η ποσότητα της φερομόνης σε κάθε ακμή κατά την επανάληψη  $t+1$ ) του μυρμηγκιού  $k$  είναι (**κανόνας ανανέωσης της φερομόνης – pheromone update rule**):

$$\tau_{ij}(t+1) = (1 - \rho)\tau_{ij}(t) + \Delta\tau_{ij}(t) \quad \text{Εξ. 4}$$

όπου:

$\rho$  με  $0 \leq \rho < 1$  ο συντελεστής εξάτμισης της φερομόνης.

$$\Delta\tau_{ij}(t) = \sum_{k=1}^m \Delta\tau_{ij}^k(t) \quad \text{Εξ. 5}$$

$$\Delta\tau_{ij}^k(t) = \begin{cases} Q/L^k(t) & \alpha\nu (i, j) \in T^k(t) \\ 0 & \alpha\nu (i, j) \notin T^k(t) \end{cases} \quad \text{Εξ. 6}$$

$T^k(t)$  το ταξίδι του μυρμηγκιού  $k$ , δηλαδή η σειρά των πόλεων που επισκέφθηκε κατά την επανάληψη  $t$ .

$L^k(t)$  το συνολικό μήκος του ταξιδιού  $T^k(t)$ .

$Q$  μια παράμετρος καθοριζόμενη από το χρήστη, η οποία όμως δεν διαφοροποιεί την επίδοση του αλγορίθμου (για το λόγο αυτό στις βελτιωμένες εκδόσεις του αλγορίθμου παραλείπεται).

Ο Dorigo στο [20] αναπαράγοντας και προσαρμόζοντας την έννοια του ελιτισμού όπως αυτή χρησιμοποιείται στους εξελικτικούς αλγορίθμους εισήγαγε την ιδέα των «**ΕΚΛΕΚΤΩΝ**» **μυρμηγκιών** (*elitist ants*) στον αλγόριθμο AS δημιουργώντας τον αλγόριθμο AS<sub>e</sub>. Σε αυτή την περίπτωση ένας επιπλέον αριθμός μυρμηγκιών, έστω  $e$ , ενισχύει την βέλτιστη διαδρομή  $T_{best}$  που έχει βρεθεί έως και την τρέχουσα επανάληψη  $t$  με επιπλέον φερομόνη ίση με  $eQ/L_{best}$ . Συνεπώς, η Εξ. 4 μετασχηματίζεται στην (*elitist pheromone update rule*):

$$\tau_{ij}(t+1) = (1 - \rho)\tau_{ij}(t) + \Delta\tau_{ij}(t) + e\Delta\tau_{ij}^e(t) \quad \text{Εξ. 7}$$

όπου:

$$\Delta\tau_{ij}^e(t) = \begin{cases} Q/L_{best}(t) & \alpha\nu (i, j) \in T_{best}(t) \\ 0 & \alpha\nu (i, j) \notin T_{best}(t) \end{cases} \quad \text{Εξ. 8}$$

με  $L_{best}(t)$  το μήκος της ολικά βέλτιστης διαδρομής μέχρι τη χρονική στιγμή  $t$  και  $T_{best}(t)$  η αντίστοιχη σειρά των πόλεων.

Ο AS λόγω της ευελιξίας του φυσικού μοντέλου στο οποίο βασίζεται είναι ευπροσάρμοστος. Το γεγονός αυτό τον κάνει άμεσα αξιοποιήσιμο σε δυναμικώς μεταβαλλόμενα προβλήματα, δηλαδή σε προβλήματα στα οποία η ευριστική πληροφορία (ορατότητα) μεταβάλλεται.

Η **στασιμότητα** (*stagnation*) στους διακριτούς αλγορίθμους ACO ορίζεται ως η κατάσταση κατά την οποία όλα τα μυρμηγκία ακολουθούν την ίδια διαδρομή. Κατά τη στασιμότητα το χρησιμοποιούμενο μονοπάτι ενισχύεται συνεχώς με νέα φερομόνη εγκλωβίζοντας όλα τα μυρμηγκία στην ίδια διαδρομή χωρίς δυνατότητα διαφυγής. Η στασιμότητα θα μπορούσε να θεωρηθεί ως επιθυμητή στην περίπτωση που τα μυρμηγκία εντόπιζαν την ολικά βέλτιστη

διαδρομή. Σε κάθε άλλη περίπτωση, πάντως, θα πρέπει να γίνονται προσπάθειες αποτροπής εμφάνισης στασιμότητας κατά την επίλυση ενός προβλήματος.

Αν αποφευχθεί η εμφάνιση στασιμότητας χρησιμοποιώντας κατάλληλες μεθόδους (πχ υψηλό ρυθμό εξάτμισης της φερομόνης) τότε ο αλγόριθμος ψάχνει πάντοτε για νέες λύσεις όπως προκύπτει και από τον **αριθμό διακλάδωσης των κόμβων** (*node branching number*) [6]. Ο αριθμός διακλάδωσης των κόμβων μιας πόλης είναι ο αριθμός των ακμών που συνδέουν την πόλη αυτή με τις υπόλοιπες και που έχουν ποσότητα φερομόνης μεγαλύτερη από ένα όριο. Όταν έχει επέλθει στασιμότητα τότε ο αριθμός αυτός ισούται με 2 σε κάθε πόλη, δηλαδή η μια ακμή είναι αυτή που οδηγεί το μυρμήγκι στην πόλη και η άλλη αυτή που το απομακρύνει.

Ένα άλλο πολύ σημαντικό στοιχείο του αλγορίθμου AS είναι η άμεση και εύκολη εφαρμογή της παράλληλης επεξεργασίας βελτιώνοντας κατά πολύ την απόδοσή του. Η διαδρομή κάθε μυρμηγκιού ή μιας ομάδας μυρμηγκιών είναι δυνατόν να υπολογίζεται από διαφορετικούς επεξεργαστές οι οποίοι ανταλλάσσουν δεδομένα ανά τακτά χρονικά διαστήματα. Στο [40] μελετάτε η βέλτιστη πολιτική ανταλλαγής πληροφοριών μεταξύ αποικιών των μυρμηγκιών (Multi Colony Ant Algorithms) που δραστηριοποιούνται παράλληλα σε διαφορετικούς επεξεργαστές.

Επίσης, ο Stützle στο [47] μελετάει την παράλληλη εκτέλεση πολλών ανεξάρτητων αλγορίθμων MMAS σε σχέση με μια μεγάλη εκτέλεση του ίδιου αλγορίθμου στον ίδιο υπολογιστικό χρόνο. Στην εργασία του αυτή καταλήγει στο συμπέρασμα ότι η παράλληλη εκτέλεση δίνει καλύτερα αποτελέσματα στον ίδιο υπολογιστικό χρόνο και ιδιαίτερος αν κάθε ανεξάρτητος αλγόριθμος MMAS εκτελείται με διαφορετικές παραμέτρους. Αυτή η τακτική κάνει ιδιαίτερα «στιβαρό» τον αλγόριθμο εμφανίζοντας βέλτιστη απόδοση σε διαφορετικά προβλήματα. Ακόμη ένα συμπέρασμα που εξάγεται από αυτή την εργασία είναι ότι για την καλύτερη απόδοση των αλγορίθμων ACO είναι απαραίτητη η επαναρχικοποίησή τους σε περίπτωση που δεν εντοπίζουν νέα βέλτιστη λύση. Το τελευταίο ισοδυναμεί με παράλληλη εκτέλεση του αλγορίθμου.

Κλείνοντας το κεφάλαιο αυτό παρουσιάζουμε συνοπτικά τη διαδικασία υλοποίησης του αλγορίθμου για την επίλυση του TSP όπως επίσης και τον αντίστοιχο ψευτοκώδικα (Κώδικας 1) για άμεση προγραμματιστική ανάπτυξη του αλγορίθμου.

## Έναρξη Αλγορίθμου

### *Βήμα 1<sup>ο</sup>:*

Θέτουμε τη φερομόνη στην αρχική της τιμή (είτε σε μια πολύ μικρή τιμή είτε σε μια τυχαία τιμή) σε όλα τα μονοπάτια που συνδέουν τις πόλεις μεταξύ τους.

Υπολογίζουμε όλες τις αποστάσεις των πόλεων.

Επιλέγουμε (τυχαία) την πόλη-αφετηρία της διαδρομής κάθε μυρμηγκιού και την τοποθετούμε στην λίστα της μνήμης του.

### *Βήμα 2<sup>ο</sup>:*

Για κάθε μυρμήγκι, επιλέγουμε την επόμενη πόλη που θα επισκεφτεί με βάση τον *τυχαίο αναλογικό κανόνα μετάβασης* (Εξ. 3) μέχρι να ολοκληρώσει τη διαδρομή του επισκεπτόμενο κάθε πόλη μια μόνο φορά και τελικά να επιστρέψει στην αφετηρία του.

### *Βήμα 3<sup>ο</sup>:*

Καταγράφουμε την καλύτερη διαδρομή που βρέθηκε.

### *Βήμα 4<sup>ο</sup>:*

Ανανεώνουμε τη φερομόνη στα μονοπάτια που επισκέφτηκαν τα μυρμήγκια με βάση τον *κανόνα ανανέωσης της φερομόνης* (Εξ. 7, με  $e=0$  σε περίπτωση που δεν θέλουμε να χρησιμοποιήσουμε τα «εκλεκτά» μυρμήγκια).

### *Βήμα 5<sup>ο</sup>:*

Επαναλαμβάνουμε τη διαδικασία από το *βήμα 2* μέχρι έως ότου ολοκληρωθεί ένας συγκεκριμένος αριθμός επαναλήψεων ή επιτευχθεί ένα κριτήριο σύγκλισης.

## **Ολοκλήρωση Αλγορίθμου.**

**Κώδικας 1. Ψευτοκώδικας Ant System (AS) για το TSP****Start:**

// Initialize

**For**  $i, j: 1$  to  $n$     Set  $\tau_{ij}(0) = \tau_0$ **End for****For**  $i, j: 1$  to  $n$     Set  $\eta_{ij} = 1/d_{ij}$ **End for****For**  $k: 1$  to  $m$     Set (randomly) 1<sup>st</sup> town in each ant's Tabu list:  $T^k(0)$ **End for****Initialize minimum tour length:**  $L_{best}$ 

// Main Loop

**For**  $t: 1$  to *maximum iterations*    **For**  $k: 1$  to  $m$         **For**  $i: 1$  to  $n-1$             Select next town by applying *random proportional transition rule*            Place next town in Tabu list:  $T^k(t)$         **End for**        **Compute tour length:**  $L^k(t)$         **If**  $L < L_{best}$  set  $L_{best} = L$  and keep edges of best tour:  $T_{best} = T^k(t)$     **End for**    **For**  $i, j: 1$  to  $n$         Update pheromone trails by applying (*elitist*) *pheromone update rule*    **End for****End for****Print**  $L_{best}$  and  $T_{best}$ **Stop**

## 4.2 ANT COLONY SYSTEM (ACS)

Ο αλγόριθμος αυτός εισήχθη από τον Dorigo [21], [22] και αποτελεί βελτίωση του AS. Μάλιστα, τα αποτελέσματα του ACS είναι σαφώς ανώτερα του AS για το TSP. Οι βασικές διαφορές των δύο αλγορίθμων μπορούν να συνοψιστούν στις εξής:

**Κανόνας μετάβασης (transition rule):** Τώρα πλέον, ένα μυρμήγκι  $k$  που βρίσκεται στην πόλη  $i$  επιλέγει την επόμενη πόλη  $j$  που θα επισκεφθεί με τον εξής κανόνα (*pseudo-random-proportional transition rule*):

$$j = \begin{cases} \max_{u \in J_i^k} \{ \tau_{iu}(t) \cdot [\eta_{iu}(t)]^\beta \} & \text{αν } q \leq q_o \\ J & \text{αν } q > q_o \end{cases} \quad \text{Εξ. 9}$$

όπου:

$q$  είναι μια τυχαία μεταβλητή ομοιόμορφα κατανομημένη στο διάστημα  $[0, 1]$ .

$q_o$  είναι μια καθοριζόμενη παράμετρος με τιμή στο διάστημα  $[0, 1]$ .

$J$  είναι μια πόλη που επιλέγεται τυχαία σύμφωνα με την παρακάτω εξίσωση:

$$p_{ij}^k(t) = \frac{\tau_{ij}(t) \cdot [\eta_{ij}(t)]^\beta}{\sum_{l \in J_i^k} \tau_{il}(t) \cdot [\eta_{il}(t)]^\beta} \quad \text{αν } J \notin J_i^k \quad \text{Εξ. 10}$$

Όταν το  $q < q_o$  τότε τα μυρμήγκια εκμεταλλεύονται (*exploit*) τον ήδη εξερευνημένο χώρο, δηλαδή επιλέγουν τις τοπικά βέλτιστες λύσεις χρησιμοποιώντας τη συσσωρευμένη εμπειρία (επίπεδα φερομόνης και ορατότητας). Προφανώς, η επιλογή των τοπικά βέλτιστων λύσεων δεν είναι σίγουρο πως οδηγεί και σε ολικό βέλτιστο. Αντίθετα, για  $q > q_o$  πραγματοποιείται επιπλέον εξερεύνηση του χώρου των λύσεων. Όπως στον AS, έτσι και εδώ, τα μυρμήγκια επιλέγουν την καλύτερη έως τώρα λύση με κάποια πιθανότητα, ενώ δεν αποκλείεται να «χαθούν» με ενδεχόμενο, αργότερα, να ανακαλύψουν ένα νέο ολικό βέλτιστο.

**Ανανέωση της φερομόνης (pheromone update):** Η εναπόθεση φερομόνης γίνεται μόνο από το μυρμήγκι που μέχρι αυτήν τη χρονική στιγμή έχει κάνει την καλύτερη διαδρομή (αντί από όλα τα μυρμήγκια όπως στον AS) σε μια προσπάθεια να κατευθυνθεί η εξερεύνηση του χώρου στην περιοχή του ολικού βέλτιστου.

Συνεπώς, η σχέση που διέπει την εναπόθεση της φερομόνης μετασχηματίζεται στην εξής (*κανόνας εναπόθεσης φερομόνης στη βέλτιστη διαδρομή*):

$$\tau_{ij}(t+1) = (1 - \rho)\tau_{ij}(t) + \rho \cdot \Delta\tau_{ij}(t) \quad \text{Εξ. 11}$$

όπου:

$$\Delta\tau_{ij}(t) = \begin{cases} 1/L_{best}(t) & \text{αν } (i, j) \in T_{best}(t) \\ 0 & \text{αν } (i, j) \notin T_{best}(t) \end{cases} \quad \text{Εξ. 12}$$

με  $L_{best}(t)$  το μήκος της ολικά βέλτιστης διαδρομής μέχρι τη χρονική στιγμή  $t$  και  $T_{best}(t)$  η αντίστοιχη σειρά των πόλεων.

**Τοπική ανανέωση της φερομόνης (*local pheromone update*):** Για να αποφευχθεί ο εγκλωβισμός των μυρμηγκιών σε μια και μόνο διαδρομή<sup>20</sup>, κάτι που θα έκανε τη χρήση των  $m$  μυρμηγκιών ατελέσφορη, πρέπει κάθε μυρμήγκι  $k$  που επισκέπτεται την πόλη  $j$  από την πόλη  $i$ , να αφαιρεί ένα ποσό φερομόνης από το μονοπάτι που συνδέει αυτές τις δύο πόλεις. Με αυτόν τον τρόπο, η εξερεύνηση για νέες λύσεις κατευθύνεται μακριά από την καλύτερη που έχει υπολογισθεί έως αυτή τη στιγμή. Η σχέση που ισχύει σε αυτή την περίπτωση είναι:

$$\tau_{ij}(t+1) = (1 - \rho)\tau_{ij}(t) + \rho \cdot \tau_o \quad \text{Εξ. 13}$$

όπου:  $\tau_o$  η ποσότητα της φερομόνης με την οποία γίνεται η αρχικοποίηση. Έχει βρεθεί πειραματικά [21] ότι:  $\tau_o = (nL_{nn})^{-1}$  με  $n$  τον αριθμό των πόλεων και  $L_{nn}$  το μήκος μιας διαδρομής που δημιουργείται με τον αλγόριθμο Nearest Neighbor Heuristic.

**Χρήση λίστας υποψηφίων πόλεων (*candidate list*):** Στα προβλήματα με μεγάλο αριθμό πόλεων – και ειδικά σε προβλήματα ATSP – είναι χρονοβόρο να εξετάζονται όλες οι πόλεις. Για το λόγο αυτό η λίστα αυτή περιέχει ένα αριθμό (έστω  $cl$ ) από τις πιο «επιθυμητές», δηλαδή κοντινές, πόλεις ταξινομημένες από την κοντινότερη προς την πιο μακρινή. Μόνο όταν οι πόλεις που βρίσκονται στη λίστα των υποψηφίων πόλεων ανήκουν και στην Tabu list είναι δυνατόν τα μυρμήγκια να κατευθυνθούν σε άλλες πόλεις.

**Άλλες διαφοροποιήσεις:** στο [6] προτείνεται να δοκιμαστούν τα εξής:

- (α) ανανέωση της φερομόνης από τα δύο μυρμήγκια με τις καλύτερες διαδρομές
- (β) αφαίρεση φερομόνης από τις κακές διαδρομές και, τέλος
- (γ) χρήση αλγορίθμων για τοπική αναζήτηση λύσεων, πχ. 2-opt, 3-opt, Lin-Kernighan (LK).

---

<sup>20</sup> Αφού επιτρέπεται μόνο στο μυρμήγκι που έχει βρει τη βέλτιστη έως τώρα διαδρομή να εναποθέσει φερομόνη.

### 4.3 MAX-MIN ANT SYSTEM (MMAS)

Ο πιο πρόσφατος αλγόριθμος που έχει βελτιώσει σημαντικά τον AS και μάλιστα δίνει και καλύτερα αποτελέσματα από τον ACS στο TSP είναι ο MMAS του Stützle [49]. Επιγραμματικά οι διαφοροποιήσεις από τον AS είναι οι εξής:

**Μόνο ένα μυρμήγκι εναποθέτει φερομόνη σε κάθε επανάληψη:** αυτό που έχει βρει την ολική βέλτιστη λύση (*global best ant*) ή αυτό που βρίσκει την καλύτερη λύση σε κάθε επανάληψη (*iteration best ant*). Προφανώς είναι δυνατή η χρήση διαφόρων συνδυασμών μεταξύ αυτών των δύο. Για παράδειγμα, στην αρχή μπορεί να χρησιμοποιείται η καλύτερη λύση κάθε επανάληψης για να εξερευνηθεί (*exploration*) όσο το δυνατόν καλύτερα όλος ο χώρος των λύσεων και καθώς αυξάνεται ο αριθμός των επαναλήψεων να χρησιμοποιείται όλο και περισσότερο η ολική βέλτιστη λύση έτσι ώστε να γίνει εκμετάλλευση (*exploitation*) της περιοχής της λύσης αυτής.

Η ανανέωση της φερομόνης δίνεται από την εξής σχέση:

$$\tau_{ij}(t+1) = (1 - \rho)\tau_{ij}(t) + \Delta\tau_{ij}(t) \quad \text{Εξ. 14}$$

όπου:

$$\Delta\tau_{ij}(t) = \begin{cases} 1/L^*(t) & \alpha\nu(i, j) \in T^*(t) \\ 0 & \alpha\nu(i, j) \notin T^*(t) \end{cases} \quad \text{Εξ. 15}$$

με  $L^*(t)$  το μήκος είτε της ολικά βέλτιστης διαδρομής μέχρι τη χρονική στιγμή  $t$  ( $L_{best}(t)$ ) είτε της καλύτερης λύσης κατά την επανάληψη  $t$  ( $L_{iter}(t)$ ) και  $T^*(t)$  η αντίστοιχη σειρά των πόλεων.

**Χρήση ορίων στα επίπεδα της φερομόνης:** στον MMAS χρησιμοποιείται τόσο άνω όριο  $\tau_{max}$  όσο και κάτω  $\tau_{min}$  για τον περιορισμό της φερομόνης. Ο σκοπός του άνω ορίου είναι να μην επιτρέψει να εμφανιστεί στασιμότητα (*stagnation*) κατά την επίλυση του προβλήματος καθώς κανένα μονοπάτι δεν θα μπορεί να συγκεντρώσει τόση φερομόνη – και κατά συνέπεια να παρουσιάσει μεγάλη πιθανότητα επιλογής – ώστε να έλκει πάντοτε όλα τα μυρμήγκια. Επιπλέον, το κάτω όριο της φερομόνης διασφαλίζει το ότι καμιά διαδρομή δεν θα έχει μηδενική ή περίπου μηδενική πιθανότητα εκλογής.

Τα άνω και κάτω όρια της φερομόνης ορίζονται με τις παρακάτω σχέσεις:

$$\tau_{max} = \frac{1}{\rho \cdot L_{opt}} \quad \text{Εξ. 16}$$

$$\tau_{\min} = \frac{\tau_{\max} \left(1 - \sqrt[n]{p_{best}}\right)}{(n/2 - 1) \sqrt[n]{p_{best}}} \quad \text{Εξ. 17}$$

όπου:

$L_{opt}$  το μήκος της πραγματικής – ολικής βέλτιστης λύσης. Προφανώς, επειδή αυτό δεν είναι γνωστό εκ των προτέρων χρησιμοποιούμε στη θέση του το  $L_{best}(t)$ . Συνεπώς, το  $\tau_{max}$  είναι δυναμικά μεταβαλλόμενο.

$p_{best}$  η πιθανότητα δημιουργίας της ολικά βέλτιστης λύσης. Η παράμετρος αυτή καθορίζεται από το χρήστη. Σημειώνεται ότι όταν  $p_{best}=1$  τότε  $\tau_{min}=0$ . Επίσης, αν η τιμή του  $p_{best}$  είναι πολύ μικρή τότε υπάρχει πιθανότητα να ισχύει  $\tau_{min} > \tau_{max}$ . Στην περίπτωση αυτή, θέτουμε  $\tau_{min} = \tau_{max}$  οπότε και ο αλγόριθμος χρησιμοποιεί μόνο την ευριστική πληροφορία (ορατότητα) για τη λύση του προβλήματος.

Απόδειξη της σχέσης του άνω ορίου (Εξ. 16):

Σε κάθε επανάληψη η μέγιστη ποσότητα φερομόνης που μπορεί να εναποτεθεί ισούται με  $1/L_{opt}$ . Συνεπώς, από τη σχέση ανανέωσης της φερομόνης (Εξ. 14) θα έχουμε:

$$\begin{aligned} \tau_{ij}(1) &= (1-\rho)\tau_{ij}(0) + 1/L_{opt} \\ \tau_{ij}(2) &= (1-\rho)\tau_{ij}(1) + 1/L_{opt} = (1-\rho)^2\tau_{ij}(0) + (1-\rho)/L_{opt} + 1/L_{opt} \\ \tau_{ij}(3) &= (1-\rho)\tau_{ij}(2) + 1/L_{opt} = (1-\rho)^3\tau_{ij}(0) + (1-\rho)^2/L_{opt} + (1-\rho)/L_{opt} + 1/L_{opt} \\ &\dots \end{aligned}$$

$$\tau_{ij}(t) = (1-\rho)^t \tau_{ij}(0) + 1/L_{opt} \sum_{i=0}^{t-1} (1-\rho)^i$$

όμως αφού  $\rho < 1$ , θα είναι:

$$\lim_{t \rightarrow \infty} (1-\rho)^t = 0$$

και:

$$\lim_{t \rightarrow \infty} \sum_{i=0}^{t-1} (1-\rho)^i = \frac{1}{\rho}$$

επομένως:

$$\lim_{t \rightarrow \infty} \tau_{ij}(t) = \lim_{t \rightarrow \infty} \left\{ (1-\rho)^t \tau_{ij}(0) + 1/L_{opt} \sum_{i=0}^{t-1} (1-\rho)^i \right\} = \frac{1}{\rho L_{opt}}$$

δηλαδή για κάθε  $\tau_{ij}$  θα ισχύει:

$$\tau_{ij}(t) \leq \frac{1}{\rho L_{opt}}$$

Απόδειξη της σχέσης του κάτω ορίου (Εξ. 17):

Όταν ο αλγόριθμος έχει συγκλίνει<sup>21</sup> ή έχει φτάσει κοντά στη σύγκλιση τότε η βέλτιστη λύση του προβλήματος δημιουργείται με πιθανότητα  $p_{best}$  που είναι αρκετά μεγαλύτερη του μηδενός. Η βέλτιστη λύση του προβλήματος δημιουργείται αν σε κάθε σημείο επιλογής – δηλαδή στις πόλεις για το TSP – το μυρμήγκι επιλέξει το μονοπάτι με τη μέγιστη φερομόνη  $\tau_{max}$ . Έστω ότι η πιθανότητα επιλογής του σωστού μονοπατιού είναι σε όλες τις πόλεις ίδια και μάλιστα ίση με  $p_{dec}$ .

Τότε θα ισχύει:

$$p_{best} = p_{dec}^n \Rightarrow p_{dec} = \sqrt[n]{p_{best}}$$

Σε κάθε πόλη το μυρμήγκι έχει να επιλέξει κατά μέσο όρο  $n/2$  ακμές που συνδέουν την πόλη που βρίσκεται με τις υπόλοιπες. Η πιθανότητα  $p_{dec}$  το μυρμήγκι να επιλέξει τη σωστή ακμή προκύπτει από την Εξ. 3 για ορατότητα ίση με τη μονάδα και για πλήθος ακμών  $n/2$ , δηλαδή:

$$p_{dec} = \frac{\tau_{max}}{\tau_{max} + (n/2 - 1)\tau_{min}}$$

Επομένως:

$$\tau_{min} = \frac{\tau_{max} (1 - \sqrt[n]{p_{best}})}{(n/2 - 1)\sqrt[n]{p_{best}}}$$

**Αρχικοποίηση της φερομόνης στην τιμή  $\tau_{max}$ , δηλαδή  $\tau_o = \tau_{max}$ :** Η διαφοροποίηση αυτή βοηθάει στην καλύτερη εξερεύνηση (*exploration*) του χώρου των λύσεων στις πρώτες επαναλήψεις του αλγορίθμου, καθώς η σχετική διαφορά μεταξύ των επιπέδων της φερομόνης των καλών και των κακών λύσεων είναι μικρή. Στο γεγονός αυτό συνεισφέρουν και τα όρια  $\tau_{max}$  και  $\tau_{min}$ .

---

<sup>21</sup> Η **σύγκλιση** (*convergence*) στον MMAS ορίζεται ως η κατάσταση στην οποία σε κάθε πόλη ένα από τα μονοπάτια που την συνδέουν με τις άλλες πόλεις έχει τιμή φερομόνης ίση με  $\tau_{max}$ , ενώ όλα τα υπόλοιπα έχουν τιμή ίση με  $\tau_{min}$ . Η κατάσταση αυτή διαφέρει από τη στασιμότητα καθώς η ύπαρξη κάτω ορίου στην φερομόνη δεν κάνει απίθανη την επιλογή ενός μονοπατιού με επίπεδα φερομόνης ίσα με  $\tau_{min}$ .

Αντιθέτως, όταν προστίθεται φερομόνη σε ένα αρχικά ομοιόμορφο χώρο με πολύ χαμηλά επίπεδα φερομόνης, η σχετική διαφορά των επιπέδων της φερομόνης των καλών και των κακών λύσεων είναι πολύ μεγάλη. Με αυτό τον τρόπο, ο αλγόριθμος οδηγείται στην εκμετάλλευση (*exploitation*) του χώρου των καλών λύσεων αγνοώντας, κατά κάποιον τρόπο, την εξερεύνηση (*exploration*) όλου του χώρου των λύσεων.

**Ομαλοποίηση της φερομόνης (*smoothing of the pheromone trail*):** Όταν ο αλγόριθμος έχει συγκλίνει ή έχει φτάσει κοντά στη σύγκλιση τότε επιχειρείται ομαλοποίηση των επιπέδων της φερομόνης. Ο μηχανισμός αυτός πραγματοποιεί αύξηση των επιπέδων της φερομόνης ανάλογα με τη διαφορά τους από το  $\tau_{max}$  έτσι ώστε να αυξηθεί η πιθανότητα επιλογής των μονοπατιών με χαμηλά επίπεδα φερομόνης.

Η σχέση που υλοποιεί την παραπάνω διαδικασία έχει ως εξής:

$$\tau_{ij}(t) \leftarrow \tau_{ij}(t) + \delta \cdot (\tau_{max}(t) - \tau_{ij}(t)) \quad \text{Εξ. 18}$$

όπου το  $\delta$  είναι μια παράμετρος καθοριζόμενη από το χρήστη με  $0 \leq \delta \leq 1$ . Για  $\delta=1$  έχουμε επαναρχικοποίηση των επιπέδων της φερομόνης, ενώ για  $\delta=0$  απενεργοποιείται ο μηχανισμός αυτός.

Η διαδικασία αυτή χρησιμοποιείται κυρίως σε εκτελέσεις του αλγορίθμου με πολύ μεγάλο αριθμό επαναλήψεων έτσι ώστε να εξερευνείται καλύτερα ο χώρος των λύσεων. Επίσης, ο μηχανισμός αυτός κάνει πιο «στιβαρό» τον αλγόριθμο στις διάφορες τιμές του  $\tau_{min}$ .

#### 4.4 ΑΠΟΤΕΛΕΣΜΑΤΑ

Για την εκτενέστερη παρουσίαση των διακριτών αλγορίθμων ACO, στο παρόν κεφάλαιο θα παρουσιάσουμε μια σειρά εφαρμογών του αλγορίθμου  $AS_e$  σε προβλήματα TSP. Η επιλογή του  $AS_e$  έγινε καθώς είναι ο πιο απλός αλγόριθμος ACO με αξιόλογη απόδοση. Η μετατροπή του κώδικα  $AS_e$  σε κώδικα ACS ή MMAS είναι άμεση και απλή, αλλά η παρουσίαση εκτενών αποτελεσμάτων από όλους του διακριτούς αλγορίθμους ACO είναι εκτός των πλαισίων της παρούσης διπλωματικής εργασίας. Παρά ταύτα, για λόγους πληρότητας της ανάπτυξης, θα παρουσιάσουμε τα συγκριτικά αποτελέσματα σε προβλήματα TSP των γνωστών αλγορίθμων ACO ( $AS$ ,  $AS_e$ , ACS,  $AS_{rank}$ , MMAS) όπως αυτά δημοσιεύτηκαν από τον Stützle στο [49].

Ο κώδικας του αλγορίθμου  $AS_e$  αναπτύχθηκε σε Borland C++ για να εκμεταλλευτούμε το γρήγορο μεταφραστή της γλώσσας αλλά και το πλήρες προγραμματιστικό περιβάλλον. Η πιο ενημερωμένη βιβλιοθήκη με προβλήματα TSP βρίσκεται στη διεύθυνση του διαδικτύου: <http://www.iwr.uni-heidelberg.de/iwr/comopt/soft/TSPLIB/TSPLIB.html>. Από εκεί πήραμε και

τα προβλήματα (eil51, berlin52 και kroA100) που παρουσιάζονται παρακάτω. Οι παράμετροι του AS<sub>e</sub> που χρησιμοποιήσαμε έχουν ως εξής [6], [49]:  $\alpha=1$ ,  $\beta=5$ ,  $\rho=0.5$ ,  $m=e=n$ ,  $Q=100$ ,  $\tau_o=10^{-6}$  και μέγιστος αριθμός επαναλήψεων 5000.

Συνοπτικά, τα αποτελέσματα εκτέλεσης του AS<sub>e</sub> έχουν ως εξής:

- Για το πρόβλημα eil51 – ένα θεωρητικό πρόβλημα 51 πόλεων (Εικόνα 10): Η συντομότερη διαδρομή που υπολογίσθηκε (Εικόνα 11) έχει μήκος 431.32 και βρέθηκε σε 1960 επαναλήψεις ή περίπου σε 99960 «ταξίδια» μυρμηγκιών. Η βέλτιστη λύση του προβλήματος eil51 έχει μήκος 426 (Εικόνα 9).
- Για το πρόβλημα berlin52 – ένα πραγματικό πρόβλημα με 52 θέσεις στο Βερολίνο (Εικόνα 13): Η συντομότερη διαδρομή που υπολογίσθηκε (Εικόνα 14) έχει μήκος 7548.99 και βρέθηκε σε 23 επαναλήψεις ή περίπου σε 1196 «ταξίδια» μυρμηγκιών.
- Για το πρόβλημα kroA100 – ένα θεωρητικό πρόβλημα 100 πόλεων (Εικόνα 16): Η συντομότερη διαδρομή που υπολογίσθηκε (Εικόνα 17) έχει μήκος 21483.79 και βρέθηκε σε 1563 επαναλήψεις ή περίπου σε 156300 «ταξίδια» μυρμηγκιών. Η βέλτιστη λύση του προβλήματος kroA100 έχει μήκος 21282.

Από τα παραπάνω αποτελέσματα και από αυτά που παρουσιάζει ο Stützle στο [49] (Πίνακας 1) βλέπουμε ότι ο αλγόριθμος AS<sub>e</sub> έχει την αναμενόμενη απόδοση και ποιότητα λύσης.

Ο Stützle συγκρίνει τους διακριτούς αλγορίθμους ACO (AS, AS<sub>e</sub>, ACS, AS<sub>rank</sub> και MMAS) εφαρμόζοντάς τους σε προβλήματα TSP (eil51, kroA100, d198) και ATSP (ry48p, ft70, kro124p, ftv170) που προτάθηκαν στον πρώτο διεθνή διαγωνισμό για την εξελικτική βελτιστοποίηση (First International Contest on Evolutionary Optimization). Η σύγκριση γίνεται χρησιμοποιώντας για όλους τους αλγορίθμους τον ίδιο αριθμό «ταξιδιών» από τα μυρμηγκία. Ο αριθμός αυτός ορίζεται ίσος με  $kn10000$ , όπου:  $k=1$  για τα προβλήματα TSP,  $k=2$  για τα προβλήματα ATSP και  $n$  ο αριθμός των πόλεων κάθε προβλήματος.

Για τον κάθε αλγόριθμο χρησιμοποιήθηκαν οι εξής παράμετροι:

- AS:  $\alpha=1$ ,  $\beta=5$ ,  $\rho=0.5$ ,  $m=n$ ,  $Q=100$ ,  $\tau_o=10^{-6}$ .
- AS<sub>e</sub>:  $\alpha=1$ ,  $\beta=5$ ,  $\rho=0.5$ ,  $m=e=n$ ,  $Q=100$ ,  $\tau_o=10^{-6}$ . Όταν χρησιμοποιήθηκε και ο μηχανισμός ομαλοποίησης της φερομόνης (+pts) τότε:  $\beta=1$  για τα προβλήματα TSP ή  $\beta=2$  για τα προβλήματα ATSP και  $\delta=0.5$ .

- ACS<sup>22</sup>:  $\alpha=1, \beta=2, \rho=0.1, q_o=0.9, cl=15, \tau_o=(nL_{mn})^{-1}, m=10$ .
- AS<sub>rank</sub>:  $\alpha=1, \beta=2, e=6$ . Όταν χρησιμοποιήθηκε και ο μηχανισμός ομαλοποίησης της φερομόνης (+pts) τότε:  $\beta=1$  για τα προβλήματα TSP ή  $\beta=2$  για τα προβλήματα ATSP και  $\delta=0.5$ .
- MMAS:  $\alpha=1, \beta=2, \rho=0.02, m=n, p_{best}=0.05, \delta=0.5$ , χρήση λίστας υποψηφίων πόλεων και, τέλος, ανανέωση της φερομόνης μόνο από το μυρμήγκι με την καλύτερη λύση σε κάθε επανάληψη.

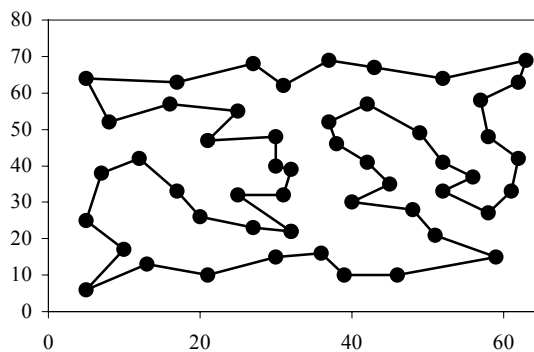
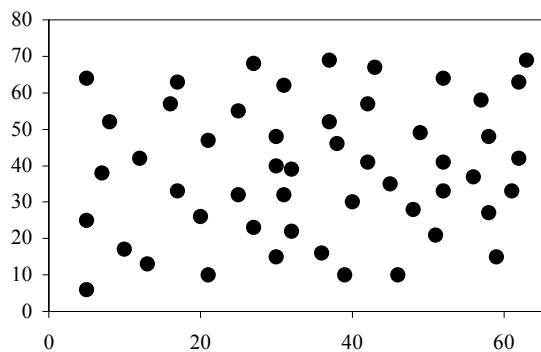
Προβλ.	Βελτ.	MMAS +pts	MMAS	ACS	AS <sub>rank</sub>	AS <sub>rank</sub> +pts	AS <sub>e</sub>	AS <sub>e</sub> +pts	AS
eil51	426	<b>427.1</b>	427.6	428.1	434.5	428.8	428.3	427.4	437.3
kroA100	21282	<b>21291.6</b>	21320.3	21420	21746	21394.9	21522.8	21431.9	22471.4
d198	15780	<b>15956.8</b>	15972.5	16054	16199.1	16025.2	16205	16140.8	16702.1
ry48p	14422	14523.4	14553.2	14565.4	<b>14511.4</b>	14644.6	14685.2	14657.9	15296.4
ft70	38673	<b>38922.7</b>	39040.2	39099	39410.1	39199.2	39261.8	39161	39596.3
kro124p	36230	<b>36573.6</b>	36773.5	36857	36973.5	37218	37510.2	37417.7	38733.1
ftv170	2755	<b>2817.7</b>	2828.8	2826.5	2854.2	2915.6	2952.4	2908.1	3154.5

Πίνακας 1. Σύγκριση των αλγορίθμων ACO σε διάφορα προβλήματα TSP («Προβλ.»). Στον πίνακα εμφανίζονται οι μέσοι όροι ενός αριθμού εκτελέσεων κάθε αλγορίθμου. Τέλος, στη στήλη «Βελτ.» παρουσιάζονται οι βέλτιστες λύσεις κάθε προβλήματος.

<sup>22</sup> Ο αριθμός των μυρμηγκιών  $m$  προκύπτει από τη σχέση [21]:

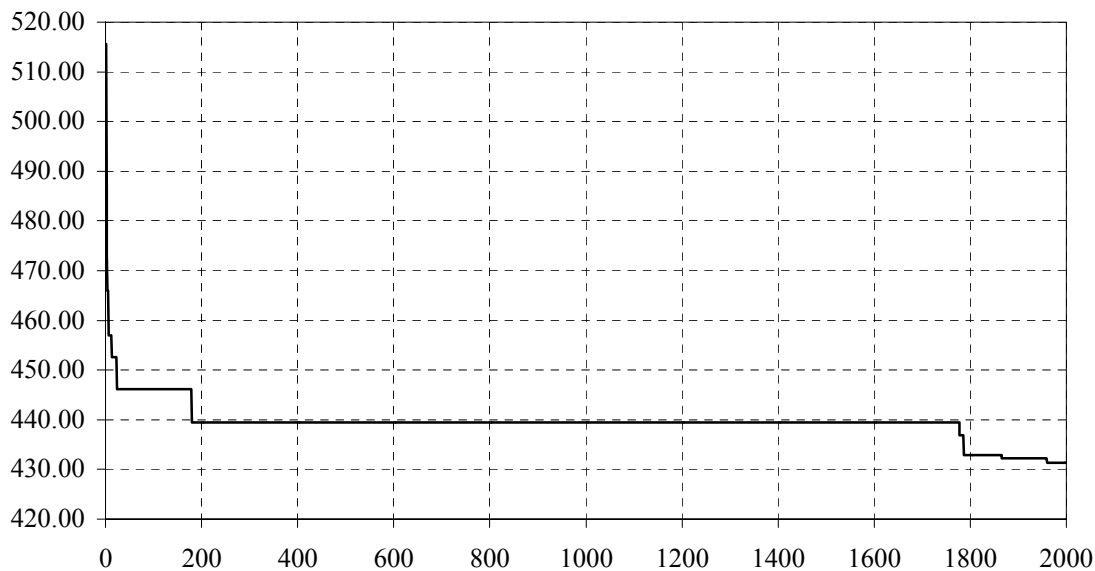
$$m = \frac{\log(\varphi_1 - 1) - \log(\varphi_2 - 1)}{q_o \log(1 - \rho)}$$

όπου:  $\varphi_1 \tau_o$  και  $\varphi_2 \tau_o$  είναι το μέσο επίπεδο φερομόνης πριν και μετά την ανανέωση της φερομόνης από το μυρμήγκι που έχει βρει την ολικά βέλτιστη διαδρομή μέχρι τη χρονική στιγμή  $t$ .

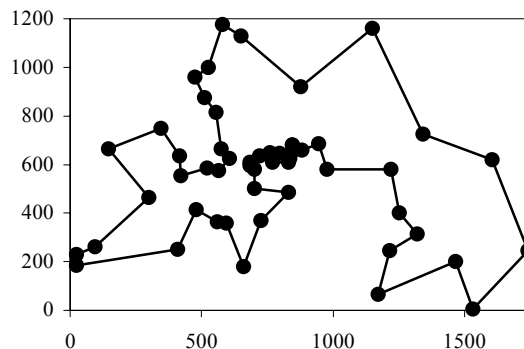
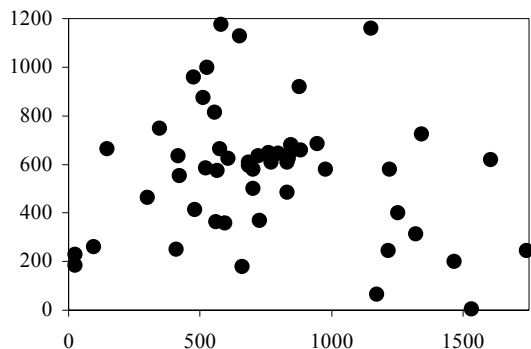


Εικόνα 10. Η κατανομή των 51 πόλεων στο πρόβλημα eil51.

Εικόνα 11. Η καλύτερη λύση (συντομότερη διαδρομή) του προβλήματος eil51 όπως αυτή υπολογίστηκε από τον αλγόριθμο  $AS_e$  σε 5000 επαναλήψεις. Η διαδρομή αυτή έχει μήκος 431.32.

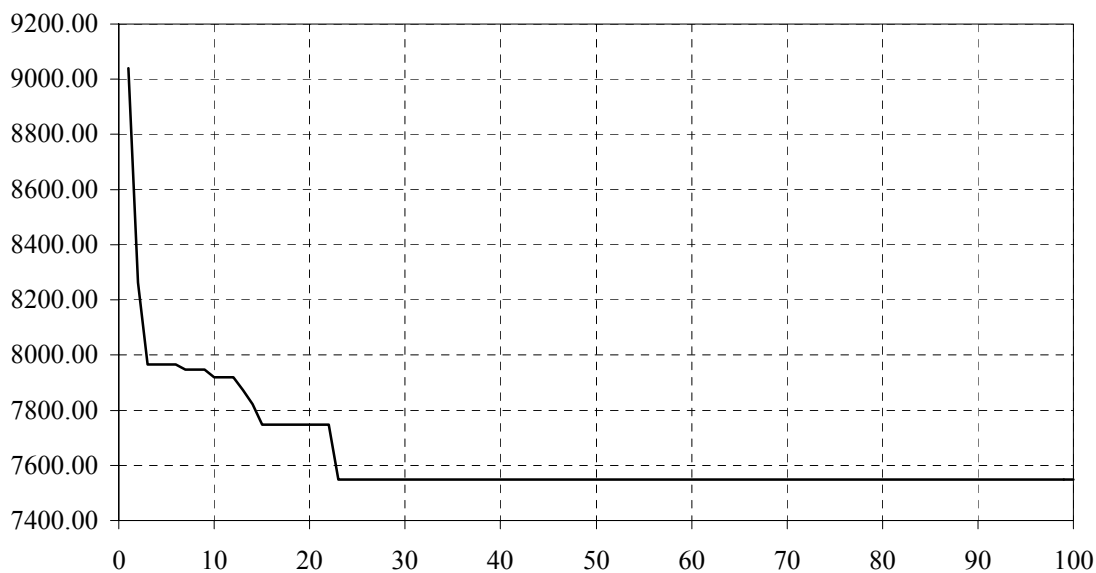


Εικόνα 12. Η σύγκλιση του αλγορίθμου  $AS_e$  για το πρόβλημα eil51 μέχρι την επανάληψη 2000.

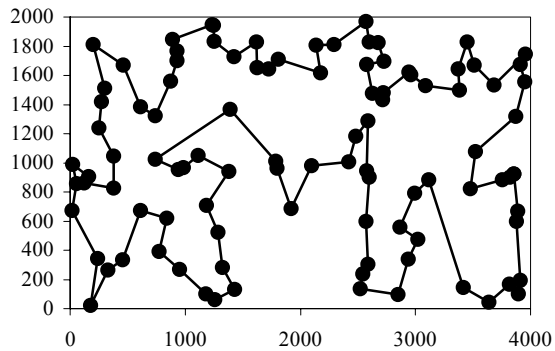
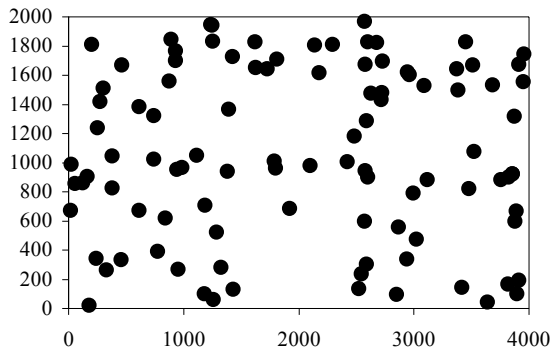


Εικόνα 13. Η κατανομή των 52 πόλεων στο πρόβλημα berlin52.

Εικόνα 14. Η καλύτερη λύση (συντομότερη διαδρομή) του προβλήματος berlin52 όπως αυτή υπολογίστηκε από τον αλγόριθμο  $AS_e$  σε 5000 επαναλήψεις. Η διαδρομή αυτή έχει μήκος 7548.99.

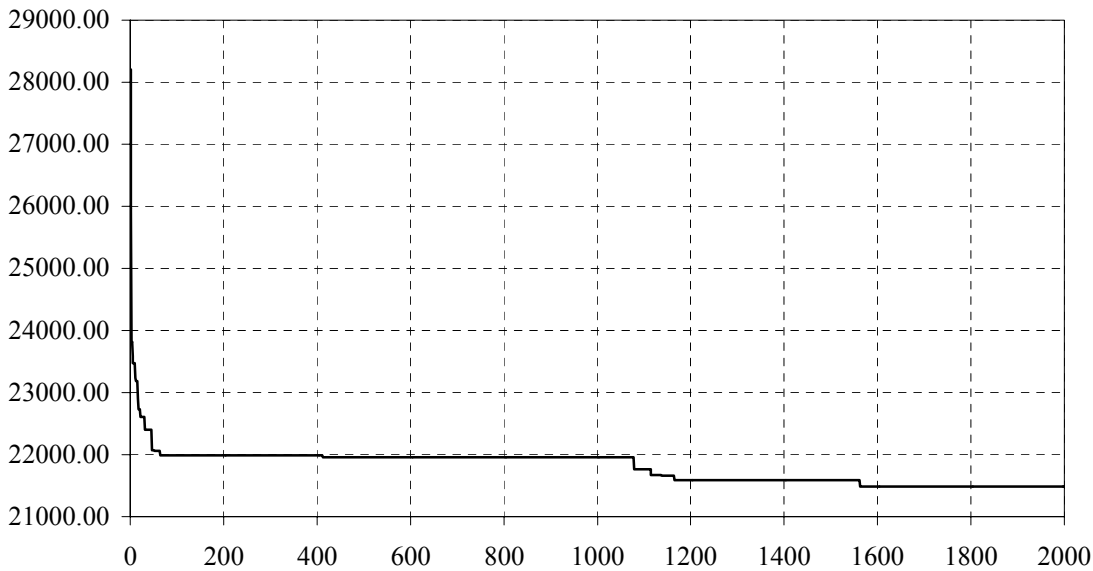


Εικόνα 15. Η σύγκλιση του αλγορίθμου  $AS_e$  για το πρόβλημα berlin52 μέχρι την επανάληψη 100.



Εικόνα 16. Η κατανομή των 100 πόλεων στο πρόβλημα kroA100.

Εικόνα 17. Η καλύτερη λύση (συντομότερη διαδρομή) του προβλήματος kroA100 όπως αυτή υπολογίστηκε από τον αλγόριθμο  $AS_e$  σε 5000 επαναλήψεις. Η διαδρομή αυτή έχει μήκος 21483.79.



Εικόνα 18. Η σύγκλιση του αλγορίθμου  $AS_e$  για το πρόβλημα kroA100 μέχρι την επανάληψη 2000.

## 5 ΑΝΤΙΣΤΡΟΦΗ ΣΧΕΔΙΑΣΗ ΑΕΡΟΤΟΜΩΝ

### 5.1 ΕΙΣΑΓΩΓΗ

Η Αντίστροφη Σχεδίαση Αεροτομών (*Inverse Design of Aerodynamic Shapes – IDS*) είναι ένα πρόβλημα βελτιστοποίησης. Σκοπός μας είναι να σχεδιάσουμε μια αεροτομή που να ικανοποιεί ορισμένες απαιτήσεις από την πλευρά της αεροδυναμικής συμπεριφοράς (τουλάχιστον στην παρούσα εργασία). Συγκεκριμένα, τις περισσότερες φορές δίνεται η κατανομή πιέσεων γύρω από μια υποθετική αεροτομή και το ζητούμενο είναι να προσδιορισθεί η μορφή αυτής της αεροτομής.

Για να υλοποιηθεί η παραπάνω διαδικασία είναι απαραίτητη η χρήση κάποιων εργαλείων. Πρώτον, θα πρέπει να γίνει παραμετροποίηση του περιγράμματος της αεροτομής. Αυτό συνήθως επιτυγχάνεται με χρήση των πολυωνύμων Bezier-Bernstein. Δεύτερον, πρέπει να χρησιμοποιηθούν κάποια υπολογιστικά εργαλεία της υπολογιστικής ρευστομηχανικής (*Computational Fluid Dynamics – CFD*), όπως η μέθοδος των ιδιόμορφων σημείων (*Panel method*) ή η επίλυση των εξισώσεων Euler ή Navier-Stokes. Τέλος, προφανώς, είναι απαραίτητη η χρήση μιας μεθόδου βελτιστοποίησης, όπως για παράδειγμα των γενετικών αλγορίθμων (GAs) [28] ή της μεθόδου που προτείνεται στο Κεφάλαιο 7 της παρούσας εργασίας.

### 5.2 ΠΑΡΑΜΕΤΡΟΠΟΙΗΣΗ ΤΗΣ ΑΕΡΟΤΟΜΗΣ

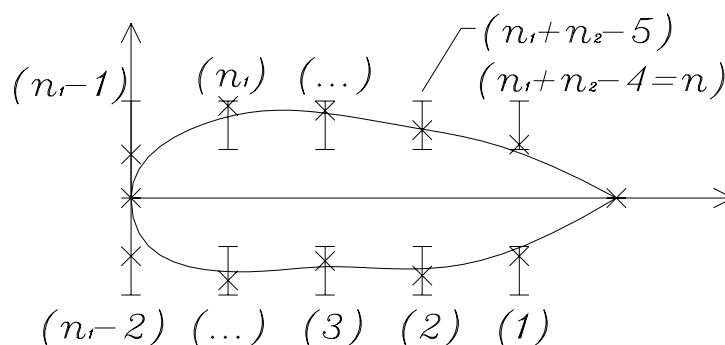
Για την παραμετροποίηση των αεροτομών στις σύγχρονες μεθόδους βελτιστοποίησης γίνεται χρήση των πολυωνύμων Bezier-Bernstein. Η εφαρμογή αυτών των πολυωνύμων μπορεί να γίνει με πολλούς τρόπους. Στο παρόν κεφάλαιο θα παρουσιάσουμε περιληπτικά μια αντιπροσωπευτική δομή η οποία και θα χρησιμοποιηθεί παρακάτω.

Μια αεροτομή είναι δυνατόν να παραμετροποιηθεί με δύο καμπύλες Bezier. Μια στην πλευρά πίεσης (*Pressure Side – PS*) της αεροτομής με έστω  $n_1$  σημεία ελέγχου και μια στην πλευρά υποπίεσης (*Suction Side – SS*) με  $n_2$  σημεία ελέγχου. Η αρίθμηση αυτών των σημείων ξεκινάει από την κάτω πλευρά (έστω πλευρά πίεσης) από το σημείο εκφυγής (*Trailing Edge – TE*) προς το σημείο προσβολής (*Leading Edge – LE*), ενώ συνεχίζεται αντίστροφα στην πάνω πλευρά, δηλαδή από το σημείο προσβολής προς το σημείο εκφυγής (Εικόνα 19).

Τα σημεία *LE* και *TE* τα θεωρούμε σταθερά. Επίσης θεωρούμε σταθερές κατά  $x$  τις συντεταγμένες όλων των υπολοίπων σημείων ( $n_1$  και  $n_2$ ) επιτρέποντας την μετακίνησή τους

μόνο κατά  $y$ . Συνεπώς, δημιουργούνται  $n=n_1+n_2-4$  βαθμοί ελευθερίας (*degrees of freedom – dof*).

Η απαραίτητη καμπυλότητα της αεροτομής στο χείλος προσβολής επιτυγχάνεται με τη χρήση δύο σημείων ελέγχου των καμπυλών Bezier στην περιοχή αυτή (Εικόνα 19). Ομοίως, αν απαιτείται καμπυλότητα στο χείλος εκφυγής χρησιμοποιούμε δύο σημεία ελέγχου και στην περιοχή του χείλους εκφυγής.



**Εικόνα 19.** Μια «τυχαία» αεροτομή όπως προκύπτει από τυχαία εκλογή των θέσεων των σημείων ελέγχου των καμπυλών που δημιουργούνται με χρήση των πολυωνύμων Bezier-Bernstein. Για τη συγκεκριμένη αεροτομή έχουμε  $n_1=n_2=7$ . Με (x) συμβολίζονται τα σημεία ελέγχου ενώ με (I) οι περιοχές στις οποίες μπορούν να κινηθούν αυτά (βαθμοί ελευθερίας).

### 5.3 ΣΧΟΛΙΑ

Για την επίλυση του προβλήματος IDS θα χρησιμοποιήσουμε μια νέα μέθοδο την Extended Ant Colony Optimization (EACO) η οποία βασίζεται στους διακριτούς αλγόριθμους ACO και παρουσιάζεται παρακάτω (Κεφάλαιο 7). Πριν την παρουσίαση του EACO θα γίνει μια βιβλιογραφική αναδρομή στους δύο υπάρχοντες αλγόριθμους που επιλύουν συνεχή προβλήματα – χωρίς εφαρμογή, όμως, στο IDS – και που βασίζονται στη συμπεριφορά των μυρμηγκιών κατά την αναζήτηση και συγκομιδή τροφής. Η ύπαρξη μικρού αριθμού τέτοιων μεθόδων για την επίλυση συνεχών προβλημάτων και ιδιαίτερα προβλημάτων IDS επιβεβαιώνει τα στοιχεία πρωτοτυπίας της νέας μεθόδου (EACO) που παρουσιάζεται στην παρούσα διπλωματική εργασία.

## 6 ΣΥΝΕΧΕΙΣ ΑΛΓΟΡΙΘΜΟΙ ΒΕΛΤΙΣΤΟΠΟΙΗΣΗΣ

### 6.1 CONTINUOUS ANT COLONY OPTIMIZATION (CACO)

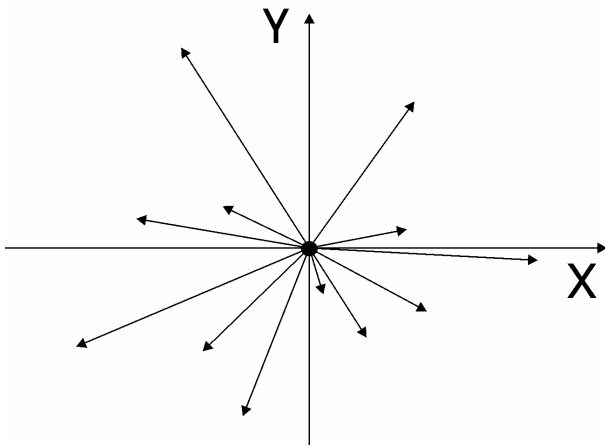
Η πρώτη προσπάθεια δημιουργίας ενός αλγορίθμου που να βασίζεται στο μοντέλο των αποικιών των μυρμηγκιών και να επιλύει συνεχή προβλήματα έγινε από τους Bilchev και Parmee [3], [4]. Ο αλγόριθμος αυτός, όμως, επικεντρωνόταν κυρίως στην τοπική έρευνα (*local search*). Αργότερα, οι Wodrich και Bilchev [52] ανέπτυξαν περαιτέρω τον αρχικό αλγόριθμο δημιουργώντας τον Continuous Ant Colony Optimization (CACO) για ολική βελτιστοποίηση προβλημάτων (*global optimization*). Τον CACO βελτίωσε ο Mathur με τους συνεργάτες του [37] εισάγοντας τον αλγόριθμο στη χημική βιομηχανία όπου φαίνεται ότι βρίσκει εφαρμογή [32], [33].

Ο CACO έχει βασικά δύο επίπεδα: την τοπική έρευνα (*local search*) και την ολική έρευνα (*global search*). Η τοπική έρευνα, η οποία χρησιμοποιείται για την εκμετάλλευση (*exploitation*) του χώρου των λύσεων, αξιοποιεί οποιονδήποτε αλγόριθμο μπορεί να λύσει συνεχή προβλήματα και χρησιμοποιεί μηχανισμούς προερχόμενους από τις αποικίες των μυρμηγκιών για να κατευθύνει την έρευνα στις περιοχές με υψηλή καταλληλότητα. Η ολική έρευνα, η οποία χρησιμοποιείται για την αποφυγή των τοπικών βέλτιστων και συνεπώς για την καλύτερη εξερεύνηση (*exploration*) του χώρου των λύσεων, αξιοποιεί μηχανισμούς προερχόμενους από τους γενετικούς αλγορίθμους. Το γεγονός αυτό κάνει δύσκολη την ένταξη του CACO στους αλγορίθμους ACO και από τους Wodrich και Bilchev προτείνεται κυρίως η ονομασία Ant-Genetic-Algorithm. Παρά ταύτα, η ολική έρευνα μπορεί να αντικατασταθεί από οποιαδήποτε άλλη μέθοδο κάνοντας τον CACO προσαρμόσιμο σε ειδικά προβλήματα.

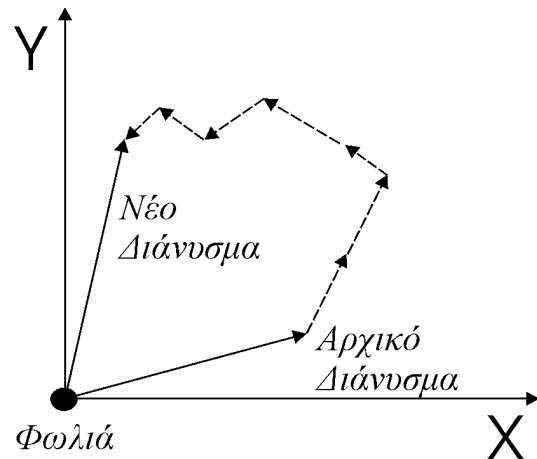
Το βασικό πρόβλημα στο πέρασμα από τους διακριτούς αλγορίθμους ACO στους συνεχείς είναι η διακριτοποίηση του χώρου των λύσεων. Αυτό στον CACO επιτυγχάνεται με τη δημιουργία ενός πεπερασμένου αριθμού προορισμών, έστω  $N$ , για τα μυρμηγκία. Οι προορισμοί αυτοί ονομάζονται περιοχές (*regions*) και δείχνουν τη γενική κατεύθυνση της αναζήτησης των λύσεων. Κάθε τέτοια περιοχή αποτελεί και μια ενδεχόμενη λύση του υπό εξέταση προβλήματος.

Οι  $N$  περιοχές αρχικά δημιουργούνται τυχαία στο χώρο των λύσεων (Εικόνα 20) και καθώς η έρευνα εξελίσσεται μετακινούνται ανάλογα με την επίδοση του μυρμηγκιού και τελικά συγκλίνουν σε περιοχές υψηλής καταλληλότητας. Η θέση κάθε περιοχής προσδιορίζεται από ένα

διάνυσμα με αρχή ένα σταθερό σημείο το οποίο είναι ίδιο για όλες τις περιοχές και ονομάζεται φωλιά<sup>23</sup> (*nest*).



Εικόνα 20. Η αρχική τυχαία κατανομή των διανυσμάτων των περιοχών γύρω από τη φωλιά σε διδιάστατο πρόβλημα (με δύο βαθμούς ελευθερίας).



Εικόνα 21. Η πορεία ενός μυρμηγκιού κατά την τοπική έρευνα μετά από ένα αριθμό επαναλήψεων. Αρχικά η περιοχή βρίσκεται στη θέση του αρχικού διανύσματος και τελικά καταλήγει στη θέση του τελικού διανύσματος.

Για την εξερεύνηση (*exploration*) και την εκμετάλλευση (*exploitation*) αυτών των περιοχών χρησιμοποιείται πληθυσμός  $A$  μυρμηγκιών εκ των οποίων  $L$  αξιοποιούνται στην τοπική έρευνα και  $G$  στην ολική ( $A=L+G$ ). Ο λόγος μεταξύ του πληθυσμού των μυρμηγκιών για τοπική και ολική έρευνα είναι μια από τις σημαντικότερες παραμέτρους του CACO.

Στην **τοπική έρευνα** (*local search*), όπως έχει ήδη αναφερθεί, μπορεί να χρησιμοποιηθεί οποιοσδήποτε αλγόριθμος επιλύει συνεχή προβλήματα. Στη γενική εφαρμογή του CACO προτιμάτε μια μέθοδος η οποία βασίζεται στον αλγόριθμο Τυχαίας Ανάβασης Λόφου (Random Hill Climbing<sup>24</sup> – RHC) λόγω της απλότητάς του. Βεβαίως, είναι δυνατόν να χρησιμοποιηθούν

<sup>23</sup> Στους αρχικούς αλγόριθμους του Bilchev [3], [4] η θέση της φωλιάς ήταν πρωταρχικής σημασίας και τοποθετούνταν σε μια θέση η οποία προσδιορίζονταν με χρήση ενός άλλου αλγορίθμου (πχ γενετικοί αλγόριθμοι GA) με κριτήριο πάντα το επίπεδο καταλληλότητάς της. Στον CACO, η επιλογή της θέσης της φωλιάς δεν είναι σημαντική παράμετρος.

<sup>24</sup> Ο αλγόριθμος αυτός είναι στοχαστικός και ιδιαίτερα εύκολος στην υλοποίησή του. Έστω  $f$  η συνάρτηση που θέλουμε να βελτιστοποιήσουμε και  $S$  ο  $n$ -διάστατος χώρος στον οποίον ορίζεται αυτή. Επίσης, θεωρούμε ότι θέλουμε να επιλύσουμε ένα πρόβλημα ελαχιστοποίησης. Παρακάτω παρουσιάζουμε τα βασικά βήματά του αλγορίθμου:

1. Αρχικοποίηση: Δημιουργούμε ένα τυχαίο σημείο  $s$  στο χώρο των λύσεων  $S$
2. Δημιουργούμε ένα τυχαίο σημείο  $s'$  στη γειτονιά του  $s$
3. Αν  $f(s') < f(s)$  αντικαθιστούμε το  $s$  με το  $s'$
4. Πηγαίνουμε στο βήμα 2 εκτός και αν ικανοποιείται το κριτήριο σύγκλησης

και πιο περίπλοκοι αλγόριθμοι αλλά τίποτα δεν εγγυάται ότι ο CACO θα έχει καλύτερη απόδοση η οποία μάλιστα να δικαιολογεί και το επιπλέον υπολογιστικό κόστος χρήσης τους.

Κατά την τοπική έρευνα, κάθε μυρμήγκι (από τα  $L$  που απασχολούνται στην τοπική έρευνα) επιλέγει μια περιοχή<sup>25</sup> (από τις  $N$ ) την οποία και θα εκμεταλλευτεί (*exploit*) ψάχνοντας για καλύτερες λύσεις. Η επιλογή της περιοχής γίνεται ανάλογα με την ποσότητα φερομόνης που αυτή έχει με βάση τον *τυχαίο-αναλογικό κανόνα μετάβασης*<sup>26</sup>:

$$p_i(t) = \frac{[\tau_i(t)]^\alpha \cdot [\eta_i(t)]^\beta}{\sum_{k=1}^N [\tau_k(t)]^\alpha \cdot [\eta_k(t)]^\beta} \quad \text{Εξ. 19}$$

όπου  $\tau_i$  η ποσότητα της φερομόνης και  $\eta_i$  η ευριστική πληροφορία<sup>27</sup> στην  $i$  περιοχή ( $i \in N$ ).

Αφού επιλέξει μια περιοχή, το μυρμήγκι πραγματοποιεί τοπική έρευνα χρησιμοποιώντας τον αντίστοιχο αλγόριθμο. Στην περίπτωση που χρησιμοποιείται ο αλγόριθμος που βασίζεται στον RHC τότε το μυρμήγκι πραγματοποιεί μια μικρή μετακίνηση με βήμα που καθορίζεται από την ηλικία (*age*) της περιοχής.

Το μήκος βήματος μεταβάλλεται γραμμικά από τη μέγιστη τιμή του για μηδενική ηλικία έως την ελάχιστη για την μέγιστη τιμή της ηλικίας. Με αυτόν τον τρόπο κάθε μυρμήγκι προσαρμόζει την έρευνά του στο χώρο επιτυγχάνοντας καλύτερη σύγκλιση. Το μέγιστο και το ελάχιστο μήκος βήματος είναι δύο από τις παραμέτρους του CACO, οι οποίες εξαρτώνται από τα όρια των μεταβλητών του προβλήματος.

Μια σχέση με την οποία θα μπορούσαμε να προσδιορίσουμε το μήκος βήματος του μυρμηγκιού ανάλογα με την ηλικία της περιοχής όπου βρίσκεται είναι η εξής:

$$step_{ij}(t) = c_j - m_j \cdot age_i(t) \quad \text{Εξ. 20}$$

όπου:

$step_{ij}(t)$  το μήκος βήματος της  $i$  περιοχής του  $j$  βαθμού ελευθερίας κατά την επανάληψη  $t$ .

<sup>25</sup> Στον αλγόριθμο που παρουσιάζει ο Mathur [37] αντί για το σύνολο των περιοχών  $N$  χρησιμοποιείται ένα ποσοστό  $S$  από αυτές επιλέγοντας από τις περιοχές με υψηλή καταλληλότητα. Με αυτόν τον τρόπο παρατηρείται βελτίωση της απόδοσης του CACO.

<sup>26</sup> Παρατηρούμε ότι η σχέση αυτή είναι όμοια με τον τυχαίο-αναλογικό κανόνα του AS με τη μόνη διαφορά ότι κάθε «ψηφιακό» μυρμήγκι έχει τη δυνατότητα να επιλέξει μια περιοχή  $i$  από όλο το σύνολο των περιοχών  $N$ , δηλαδή δεν υπάρχει λίστα μνήμης (*Tabu list*).

<sup>27</sup> Η ευριστική πληροφορία εξαρτάται από το είδος του προβλήματος. Κατά την εύρεση του μεγίστου ή του ελαχίστου μιας συνάρτησης είναι  $\eta_i = 1$ .

$c_j, m_j > 0$  παράμετροι που ορίζονται από το χρήστη για κάθε βαθμό ελευθερίας (μεταβλητή)  $j$  του προβλήματος.

$age_i(t)$  η ηλικία της  $i$  περιοχής κατά την επανάληψη  $t$ . Προφανώς, για να είναι  $step_{ij} > 0$ , πρέπει  $age_i < \min(c_j/m_j)$ . Με τον τρόπο αυτό ορίζεται η μέγιστη δυνατή ηλικία της περιοχής.

Η διεύθυνση κατά την οποία μετακινείται το μυρμήγκι στη νέα περιοχή είναι ίδια με την προηγούμενη αν κατά την προηγούμενη επανάληψη στην ίδια περιοχή το ίδιο ή άλλο μυρμήγκι είχε βρει μια βελτιωμένη τιμή καταλληλότητας. Διαφορετικά, το μυρμήγκι επιλέγει μια νέα τυχαία διεύθυνση.

Τέλος, αν το μυρμήγκι μετά το πέρας της έρευνας έχει ανακαλύψει μια νέα θέση με καλύτερη τιμή καταλληλότητας, τότε προσθέτει φερομόνη στην περιοχή αυτή ανάλογα με τη βελτίωση της καταλληλότητας. Επίσης, μεταφέρει το διάνυσμα της περιοχής της έρευνας στη νέα θέση του και μειώνει την ηλικία της περιοχής. Διαφορετικά, απλά αυξάνει την ηλικία της περιοχής.

Η **ολική έρευνα (global search)** έχει σκοπό να δημιουργεί  $G$  νέες περιοχές έρευνας (*παιδιά – children*) οι οποίες θα προέρχονται από τις καλύτερες περιοχές (*γονείς – parents*) και θα αντικαθιστούν της ασθενέστερες. Το ανάλογο με τους γενετικούς αλγορίθμους είναι προφανές και οι μηχανισμοί που αξιοποιούνται είναι η διασταύρωση (*crossover*) και η μετάλλαξη (*mutation*). Στον CACO οι μηχανισμοί που χρησιμοποιούνται στην ολική έρευνα είναι η τυχαία διαδρομή (*random walk*) και η διάχυση μονοπατιών (*trail diffusion*).

Μια ακόμα σημαντική παράμετρος του CACO είναι το πλήθος των μυρμηγκιών από τον  $G$  πληθυσμό αυτών που δραστηριοποιούνται στην τυχαία διαδρομή ( $RW$ ) και στη διάχυση των μονοπατιών ( $TD$ ). Προφανώς, θα είναι:  $G = RW + TD$ .

Κατά την **τυχαία διαδρομή (random walk)** τα μυρμήγκια καλύπτουν όλο το χώρο των λύσεων. Το φυσικό ανάλογο θα μπορούσε να είναι η τυχαία αναζήτηση τροφής. Η τυχαία διαδρομή υλοποιείται σε δύο στάδια. Στο πρώτο, το οποίο βασίζεται στο μηχανισμό της διασταύρωσης, δημιουργείται μια νέα περιοχή (*παιδί*) συνδυάζοντας στοιχεία<sup>28</sup> από περιοχές (*γονείς*) με υψηλή καταλληλότητα. Αναλυτικότερα, επιλέγεται ένας γονέας ο οποίος δίνει και το πρώτο στοιχείο στο παιδί. Κατόπιν τα υπόλοιπα στοιχεία λαμβάνονται από άλλους γονείς με βάση την πιθανότητα διασταύρωσης (*Crossover Probability – CP*). Όταν το  $CP=1$  τότε κάθε

---

<sup>28</sup> Οι συντεταγμένες του διανύσματος θέσης μιας περιοχής αποτελούν τα στοιχεία της περιοχής αυτής.

στοιχείο του παιδιού έχει διαφορετικό γονέα ενώ αν  $CP=0$  το παιδί είναι πανομοιότυπο με τον πρώτο γονέα. Ο κάθε γονέας έχει ίση πιθανότητα να επιλεγθεί για τη διασταύρωση.

Στο δεύτερο στάδιο επανατροποποιείται η νεοδημιουργηθείσα περιοχή μεταβάλλοντας τυχαία διάφορα στοιχεία της (μετάλλαξη). Κατά αυτόν τον τρόπο διασφαλίζεται η κάλυψη όλου του χώρου των λύσεων. Η μετάλλαξη σε κάθε στοιχείο της νεοδημιουργηθείσας περιοχής πραγματοποιείται με βάση την πιθανότητα μετάλλαξης (*Mutation Probability – MP*) η οποία είναι και σταθερή. Προφανώς, κάθε στοιχείο του παιδιού θα μεταβάλλεται αν ικανοποιείται η πιθανότητα μετάλλαξης.

Σημαντικός παράγοντας στη μετάλλαξη είναι το μέγιστο επιτρεπόμενο βήμα με βάση το οποίο μπορεί να μεταβληθεί κάθε στοιχείο του παιδιού. Προφανώς αυτό θα πρέπει να είναι μεγάλο στην αρχή της έρευνας και καθώς αυξάνεται ο αριθμός των επαναλήψεων να μειώνεται ταχέως. Η σχέση που ικανοποιεί τις παραπάνω απαιτήσεις είναι:

$$\Delta(T, R) = R \cdot (1 - r^{(1-T)^b}) \quad \text{Εξ. 21}$$

όπου:

$R$  η μέγιστη επιτρεπόμενη ακτίνα μέσα στην οποία δημιουργούνται οι περιοχές (*regions*)

$T$  το ποσοστό στο διάστημα  $[0, 1]$  που εκφράζει την επανάληψη  $t$  επί του συνολικού αριθμού των επαναλήψεων (= τρέχουσα επανάληψη / συνολικό αριθμό επαναλήψεων)

$r$  ένας τυχαίος αριθμός στο διάστημα  $[0, 1]$

$b$  μια παράμετρος που εκφράζει τη μη γραμμικότητα της σχέσης

Αφού ολοκληρωθεί η διαδικασία του τυχαίου περιπάτου θα πρέπει να επανατοποθετηθεί φερομόνη στη νεοδημιουργηθείσα περιοχή. Προφανώς, αν υπολογιστεί η καταλληλότητα της συγκεκριμένης περιοχής και εναποτεθεί η ανάλογη ποσότητα φερομόνης, υπάρχει μεγάλη πιθανότητα τα επίπεδα φερομόνης να μην είναι συγκρίσιμα με αυτά των γονέων καθώς δεν θα έχει πραγματοποιηθεί τοπική έρευνα στην περιοχή αυτή με αποτέλεσμα να αγνοηθεί. Για να αποφευχθεί το πρόβλημα αυτό, στην νέα περιοχή που δημιουργείται εναποτίθεται ποσότητα φερομόνης που αντιστοιχεί στο μέσο όρο του καλύτερου και χειρότερου γονέα. Με τον τρόπο αυτό αυξάνεται η πιθανότητα της περιοχής προς εκμετάλλευση (*exploitation*) αλλά και δεν μειώνεται η πιθανότητα εκμετάλλευσης (*exploitation*) των περιοχών που ήδη δίνουν καλά αποτελέσματα.

Κατά την **διάχυση μονοπατιών (trail diffusion)** αντί να επιλέγεται κάθε φορά ένας γονέας, επιλέγεται ένα ζεύγος γονέων ή πολλές φορές μια ομάδα γονέων (περιοχών με υψηλή καταλληλότητα). Κάθε ζεύγος γονέων έχει ίση πιθανότητα να επιλεγεί. Η διαδικασία ξεκινάει με την τυχαία επιλογή ενός ζεύγους γονέων. Κατόπιν, προσδιορίζεται το πρώτο στοιχείο του παιδιού από αυτό το ζεύγος γονέων σύμφωνα με τις παρακάτω επιλογές:

1. Το αντίστοιχο στοιχείο από το γονέα 1
2. Το αντίστοιχο στοιχείο από το γονέα 2
3. Ένας ζυγισμένος μέσος όρος από τα στοιχεία των δύο γονέων (με  $a$  μια τυχαία ομοιόμορφα κατανομημένη μεταβλητή στο  $[0, 1]$ ):

$$x_i(\text{child}) = a \cdot x_i(\text{parent}_1) + (1 - a) \cdot x_i(\text{parent}_2) \quad \text{Εξ. 22}$$

Η πιθανότητα επιλογής της περίπτωσης 3 είναι ίση με  $MP$ , ενώ οι πιθανότητες επιλογής των περιπτώσεων 1 και 2 είναι ίσες μεταξύ τους (δηλαδή αν  $MP=0.4$ , τότε η πιθανότητα εκλογής της περίπτωσης 3 είναι 40% ενώ των περιπτώσεων 1 και 2 είναι 30% έκαστη).

Το επόμενο στοιχείο του παιδιού μπορεί να προέρχεται από ένα νέο ζεύγος γονέων με πιθανότητα ίση με  $CP$ . Η παραπάνω διαδικασία επαναλαμβάνεται έως ότου καλυφθούν όλα τα στοιχεία του παιδιού (δηλαδή της νέας περιοχής). Τέλος, η φερομόνη που εναποθέτεται στη νέα περιοχή, όπως και στον τυχαίο περίπατο, ισούται με το μέσο όρο των επιπέδων της φερομόνης των γονέων.

Για την άρτια λειτουργία του CACO είναι απαραίτητη η ύπαρξη αρνητικής ανάδρασης όπως εξάλλου και στους διακριτούς αλγορίθμους ACO. Η αρνητική ανάδραση υλοποιείται μέσω της **εξάτμισης της φερομόνης** μετά το πέρας της τοπικής και της ολικής έρευνας. Η σχέση που διέπει την εξάτμιση της φερομόνης (κανόνας ανανέωσης της φερομόνης - *pheromone update rule*) είναι:

$$\tau_i(t+1) = (1 - \rho)\tau_i(t) \quad \text{Εξ. 23}$$

όπου:

$\tau_i$  η ποσότητα της φερομόνης στην  $i$  περιοχή ( $i \in N$ ).

$\rho$  με  $0 \leq \rho < 1$  ο συντελεστής εξάτμισης της φερομόνης.

Κλείνοντας το κεφάλαιο αυτό παρουσιάζουμε συνοπτικά τη διαδικασία υλοποίησης του CACO για την επίλυση συνεχών προβλημάτων όπως επίσης και τον αντίστοιχο ψευτοκώδικα (Κώδικας 2) για άμεση προγραμματιστική ανάπτυξη του αλγορίθμου.

## Έναρξη Αλγορίθμου

*Βήμα 1<sup>ο</sup>:*

Δημιουργούμε τυχαία τις  $N$  νέες περιοχές και υπολογίζουμε την καταλληλότητά τους.

Θέτουμε τη φερομόνη στην αρχική της τιμή σε όλες τις περιοχές.

Μηδενίζουμε την ηλικία κάθε περιοχής.

*Βήμα 2<sup>ο</sup>:*

$G$  μυρμήγκια απασχολούνται στην ολική έρευνα αντικαθιστώντας τις  $G$  ασθενέστερες από άποψη καταλληλότητας περιοχές.

Από το  $G$  πλήθος των μυρμηγκιών,  $RW$  δραστηριοποιούνται στην *τυχαία διαδρομή* και  $TD$  στη *διάχυση μονοπατιών*.

*Βήμα 3<sup>ο</sup>:*

$L$  μυρμήγκια απασχολούνται στην τοπική έρευνα. Κάθε μυρμήγκι επιλέγει μια περιοχή για να ερευνήσει με βάση τον *τυχαίο-αναλογικό κανόνα μετάβασης* (Εξ. 19). Αν το μυρμήγκι βρει μια θέση με υψηλό βαθμό καταλληλότητας τότε μεταφέρει το διάνυσμα της περιοχής στη θέση αυτή και εναποθέτει φερομόνη στην περιοχή ανάλογα με την αύξηση του βαθμού καταλληλότητας. Τέλος, ανανεώνεται η ηλικία της περιοχής.

*Βήμα 4<sup>ο</sup>:*

Καταγράφουμε το ολικό βέλτιστο και την περιοχή στην οποία αυτό βρέθηκε.

*Βήμα 5<sup>ο</sup>:*

Ανανεώνουμε τη φερομόνη στις περιοχές που επισκέφτηκαν τα μυρμήγκια με βάση τον *κανόνα ανανέωσης της φερομόνης* (Εξ. 23).

*Βήμα 6<sup>ο</sup>:*

Επαναλαμβάνουμε τη διαδικασία από το *βήμα 2* μέχρι έως ότου ολοκληρωθεί ένας συγκεκριμένος αριθμός επαναλήψεων ή επιτευχθεί ένα κριτήριο σύγκλισης.

*Βήμα 7<sup>ο</sup>:*

Εκτυπώνουμε το ολικό βέλτιστο και την περιοχή στην οποία αυτό βρέθηκε.

**Ολοκλήρωση Αλγορίθμου.**

**Κώδικας 2. Ψευτοκώδικας CACO για συνεχείς συναρτήσεις**

```
Start:  
  
// Initialize  
For  $i$ : 1 to  $N$   
  Create Randomly a Region  
  Calculate Fitness of Region  
  Set Pheromone Trail of Region to  $\tau_i(0) = \tau_0$   
  Set  $age_i(0)=0$   
End for  
  
// Main Loop  
For  $t$ : 1 to maximum iterations  
  
  // Global Search  
  For  $k$ : 1 to  $RW$   
    Perform Random Walk  
    Update Global Best  
  End for  
  For  $k$ : 1 to  $TD$   
    Perform Trail Diffusion  
    Update Global Best  
  End for  
  
  // Local Search  
  For  $k$ : 1 to  $L$   
    Perform Local Search  
    Update Global Best  
  End for  
  
  // Evaporate Pheromone Trails  
  For  $i$ : 1 to  $N$   
     $\tau_i(t+1) = \rho\tau_i(t)$   
  End for  
  
End for  
  
Print Best Fitness and Best Region  
  
Stop
```

## 6.2 ΑΛΓΟΡΙΘΜΟΣ ΕΜΠΝΕΥΣΜΕΝΟΣ ΑΠΟ ΤΑ ΜΥΡΜΗΓΚΙΑ *PACHYCONDYLA APICALIS* (API)

Στα πλαίσια της εκτενής παρουσίασης των αλγορίθμων που βασίζονται σε μοντέλα ανίχνευσης και συγκομιδής τροφής από τα μυρμήγκια, θα παρουσιάσουμε τον αλγόριθμο που βασίζεται στη συμπεριφορά των μυρμηγκιών *Pachycondyla Apicalis* (API) [42]. Ο αλγόριθμος αυτός βασίζεται σε μια άλλη πρωτόγονη ποικιλία μυρμηγκιών (*Pachycondyla Apicalis*) η οποία δεν στηρίζεται στη μαζική συγκομιδή τροφής και συνεπώς δεν χρησιμοποιεί ίχνη φερομόνης.

Κάθε αποικία από αυτό το είδος μυρμηγκιών έχει 20 με 100 άτομα εκ των οποίων μόνο το 20% με 30% βρίσκεται έξω από τη φωλιά για αναζήτηση λείας. Τα χαρακτηριστικά κάθε μυρμηγκιού πρακτικά είναι δύο: (1) δεν επικοινωνεί με τα άλλα μυρμήγκια της ομάδας του και (2) συγκρατεί στη μνήμη του τις περιοχές του κυνηγιού του. Τα μυρμήγκια αυτά δημιουργούν περιοχές κυνηγιού γύρω από τη φωλιά τους σε ακτίνα περίπου 10m. Κάθε τέτοια περιοχή έχει ακτίνα περίπου 2.5m. Όταν ένα μυρμήγκι βρει λεία στην περιοχή του, τότε επιστρέφει σε αυτή για περαιτέρω εκμετάλλευση-εξερεύνηση. Αν αποτύχει τότε πάει σε νέα περιοχή ή επιστρέφει σε παλιά στην οποία είχε επιτύχει στην αναζήτηση λείας. Ανά κάποια χρονικά διαστήματα (κατά πάσα πιθανότητα όταν δεν υπάρχει αρκετή λεία στην ευρύτερη περιοχή της φωλιάς) μια ειδική ομάδα μυρμηγκιών ψάχνει να βρει νέα τοποθεσία για τη φωλιά.

Η υλοποίηση της παραπάνω διαδικασίας σε υπολογιστικό αλγόριθμο έχει ως εξής:

1. Αρχικοποίηση: Τυχαία επιλογή της αρχικής θέσης της φωλιάς στο χώρο των λύσεων  $S$ .
2. Για κάθε μυρμήγκι:
  - Αν έχει λιγότερες περιοχές κυνηγιού στη μνήμη του από αυτές που μπορεί να συγκρατήσει, τότε αναζητεί λεία σε νέα περιοχή στο χώρο των λύσεων  $S$
  - Διαφορετικά:
    - i. αν η προηγούμενη αναζήτηση ήταν επιτυχημένη, ξαναψάχνει την ίδια τοποθεσία
    - ii. διαφορετικά επιλέγει τυχαία μια περιοχή από τη μνήμη του και την εξερευνεί (*explore*)
3. Αφαιρούνται από τη μνήμη του μυρμηγκιού όλες οι τοποθεσίες για τις οποίες το κυνήγι ήταν αποτυχημένο για ένα συγκεκριμένο αριθμό επαναλήψεων

4. Αντιγραφή μεταξύ δύο τυχαία επιλεγμένων μυρμηγκιών της καλύτερης έως τώρα τοποθεσίας
5. Αν έχει περάσει ένα συγκεκριμένος αριθμός επαναλήψεων τότε μετακινείται η φωλιά στην καλύτερη έως τώρα τοποθεσία και επαναρχικοποιούνται οι τοποθεσίες στη μνήμη των μυρμηγκιών.
6. Πηγαίνουμε στο βήμα 2 εκτός και αν ικανοποιείται το κριτήριο σύγκλισης οπότε και το πρόγραμμα τερματίζει.

Η αναζήτηση λείας σε κάθε περιοχή κυνηγιού γίνεται επιλέγοντας τυχαία ένα σημείο στη περιοχή κυνηγιού. Οι παράμετροι που υπεισέρχονται στον αλγόριθμο εκτός της περιοχής του χώρου των λύσεων  $S$ , είναι:

- ο αριθμός των περιοχών κυνηγιού που συγκρατεί κάθε μυρμήγκι στη μνήμη του
- ο συνολικός αριθμός των μυρμηγκιών
- το μέγεθος της περιοχής κυνηγιού σε σχέση με το χώρο γύρω από τη φωλιά (χώρος των λύσεων)
- ο μέγιστος αριθμός βημάτων του αλγορίθμου
- ο αριθμός των βημάτων πριν την αλλαγή της τοποθεσίας της φωλιάς
- ο αριθμός των βημάτων πριν τη διαγραφή ενός τόπου κυνηγιού από τη μνήμη του μυρμηγκιού

και τέλος:

- το κριτήριο σύγκλισης



## 7 EXTENDED ANT COLONY OPTIMIZATION (EACO)

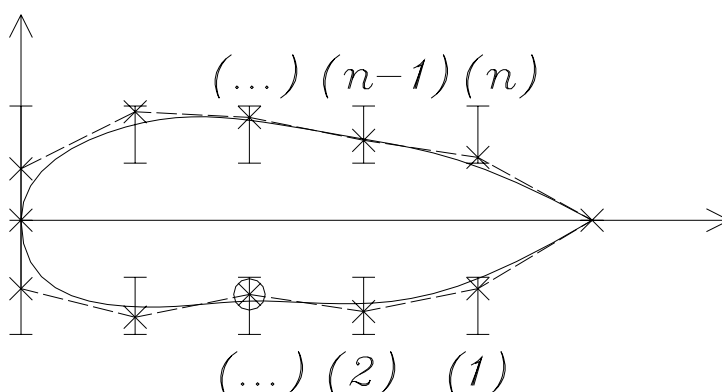
### 7.1 ΕΙΣΑΓΩΓΗ

Στα πλαίσια της έρευνας του Εργαστηρίου Θερμικών Στροβιλομηχανών του ΕΜΠ δοκιμάζονται και αναπτύσσονται νέοι αλγόριθμοι βελτιστοποίησης για την εφαρμογή τους στην Αντίστροφη Σχεδίαση Αεροτομών. Καθώς, το συγκεκριμένο πρόβλημα μπορεί να παρασταθεί γραφικά με περιοχές μέσα στις οποίες κινούνται τα σημεία ελέγχου των καμπυλών Bezier προτάθηκε η χρήση των αλγορίθμων ACO και συγκεκριμένα του AS που επιλύει προβλήματα που έχουν γραφική απεικόνιση. Όμως, η άμεση εφαρμογή του AS στο πρόβλημα IDS δεν είναι δυνατή καθώς ο AS έχει σχεδιασθεί για διακριτά προβλήματα. Επομένως, έγινε αναγκαία η επέκταση του αλγορίθμου και στο συνεχές χώρο. Το αποτέλεσμα αυτής της εργασίας ήταν ο αλγόριθμος Extended Ant Colony Optimization (EACO) [53], ο οποίος ενσωματώνει στοιχεία από όλους τους αλγορίθμους ACO.

### 7.2 ΠΕΡΙΓΡΑΦΗ

Έστω λοιπόν ότι έχουμε μια αεροτομή η οποία παραμετροποιείται με τον τρόπο που αναφέρεται στο Κεφάλαιο 5.2. Η βασική ιδέα είναι να χρησιμοποιηθεί κάθε περιοχή μέσα στην οποία μετακινούνται τα σημεία ελέγχου της καμπύλης Bezier ως μια πόλη – ισοδύναμη με τις πόλεις των διακριτών ACO – μέσα στην οποία όμως μπορεί να κινηθεί το «ψηφιακό» μυρμήγκι. Προφανώς, κάθε μυρμήγκι ξεκινάει από μια τέτοια πόλη και μετακινείται διαδοχικά στις γειτονικές του πόλεις μέχρι να επιστρέψει στη αρχική όπου έχει και την «φωλιά» του. Η διαδρομή κάθε μυρμηγκιού, δηλαδή η σειρά διαδοχής των πόλεων, είναι γνωστή και ταυτίζεται με την αύξουσα σειρά των σημείων ελέγχου των καμπυλών Bezier.

Αναλυτικότερα, επειδή οι καμπύλες Bezier περιλαμβάνουν και σταθερά σημεία ελέγχου και επιπλέον η κίνηση των υπόλοιπων σημείων ελέγχου είναι δυνατή μόνο κατά τον άξονα των  $y$ , είναι δυνατή η ταύτιση κάθε βαθμού ελευθερίας των καμπυλών με μια πόλη. Η αρίθμηση, συνεπώς, των πόλεων-βαθμών ελευθερίας γίνεται από το 1 έως το  $n$ . Τέλος, η διαδρομή κάθε μυρμηγκιού είναι κλειστή. Το πρόβλημα αυτό, συνεπώς, μπορεί να παρομοιασθεί με το TSP με τη διαφορά ότι η διαδοχή των πόλεων είναι γνωστή όχι όμως και η ακριβής θέση του μυρμηγκιού σε κάθε πόλη αφού τα μυρμήγκια μπορούν να κινηθούν μέσα στις πόλεις.



**Εικόνα 22.** Παρουσίαση των πόλεων-βαθμών ελευθερίας. Με διακεκομμένη γραμμή παρουσιάζεται η κλειστή διαδρομή ενός μυρμηγκιού. Με (x) συμβολίζονται τα σημεία που επισκέπτεται το μυρμήγκι σε κάθε πόλη, με (I) οι περιοχές στις οποίες μπορεί να κινηθεί το μυρμήγκι μέσα σε κάθε πόλη και με (O) η φωλιά του μυρμηγκιού.

Το ταξίδι κάθε μυρμηγκιού ξεκινάει από το σημείο της πόλης που έχει τη μέγιστη πιθανότητα επιλογής. Το σημείο αυτό θα το ονομάζουμε **φωλιά** (*nest*) και αντιπροσωπεύει τη συγκεντρωμένη γνώση (εμπειρία) της αποικίας όσον αφορά πιο είναι το καλύτερο σημείο σε κάθε πόλη. Η εμπειρία εξαρτάται από τα επίπεδα της φερομόνης αλλά και της ορατότητας, οι οποίες ποσότητες θα ορισθούν παρακάτω.

Το «ψηφιακό» μυρμήγκι συνεχίζει το ταξίδι του μετακινούμενο από πόλη σε πόλη ακολουθώντας τον αύξοντα αριθμό των πόλεων. Η επιλογή της θέσης που θα επισκεφθεί σε κάθε πόλη γίνεται σύμφωνα με τον γνωστό *τυχαίο-αναλογικό κανόνα μετάβασης* με κάποιες τροποποιήσεις για να ανταποκρίνεται στα συνεχή προβλήματα.

Όταν ένα μυρμήγκι ολοκληρώσει το ταξίδι του, δηλαδή επιστρέψει στη φωλιά του, τότε δημιουργούμε το περίγραμμα της αεροτομής που προκύπτει με χρήση των πολωνύμων Bezier με σημεία ελέγχου τις θέσεις που επισκέφθηκε το «ψηφιακό» μυρμήγκι στις πόλεις. Η αεροτομή αυτή (Εικόνα 24) πρέπει να αξιολογηθεί και να συγκριθεί με το στόχο μας (Εικόνα 25). Επομένως, υπολογίζουμε με τα κατάλληλα υπολογιστικά εργαλεία της ρευστομηχανικής (βλέπε Κεφάλαιο 5) την κατανομή πιέσεων γύρω από την αεροτομή που δημιούργησε το μυρμήγκι.

Συνεπώς έχουμε δύο κλειστές καμπύλες, μια του στόχου (Εικόνα 25) και μια της νέας αεροτομής (Εικόνα 26), με συντεταγμένες  $x$  τη θέση στη χορδή της αεροτομής και  $y$  την τιμή της πίεσης. Οι καμπύλες αυτές δεν συμπίπτουν απόλυτα δημιουργώντας ανάμεσά τους κλειστά χωρία. Προφανώς, για να επιτύχουμε σύγκλιση των δύο καμπυλών θα πρέπει να μηδενίσουμε το

εμβαδόν που περικλείεται ανάμεσά τους και άρα το πρόβλημα αυτό είναι ένα πρόβλημα ελαχιστοποίησης εμβαδού.

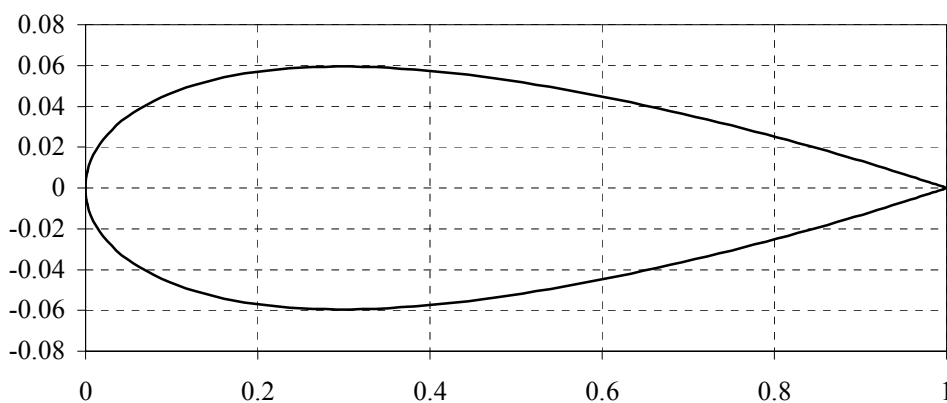
Η φερομόνη που εναποθέτει κάθε μυρμήγκι στις πόλεις από όπου πέρασε ισούται με την αντίστροφη τιμή του εμβαδού που περικλείεται από τις δύο καμπύλες. Ομοίως, θέτουμε την ορατότητα ίση με την αντίστροφη τιμή του εμβαδού που περικλείεται από τις δύο καμπύλες στη γειτονιά της πόλης. Η εναπόθεση της φερομόνης στην πόλη από τα μυρμήγκια γίνεται με εκθετική (ή άλλη φθίνουσα) κατανομή πάνω στην πόλη με κέντρο το σημείο που επισκέφτηκε το μυρμήγκι. Η απομνημόνευση της ορατότητας γίνεται συγκρατώντας τη μέγιστη ορατότητα ή προφανώς το ελάχιστο εμβαδόν.

Η ορατότητα στο IDS, εν αντιθέσει με το TSP, είναι μια δυναμικώς μεταβαλλόμενη ποσότητα καθώς εξαρτάται από τη σχετική θέση των σημείων ελέγχου. Η παρατήρηση αυτή οδήγησε σε μια νέα προσέγγιση του προβλήματος η οποία τελικά έδωσε και καλύτερα αποτελέσματα.

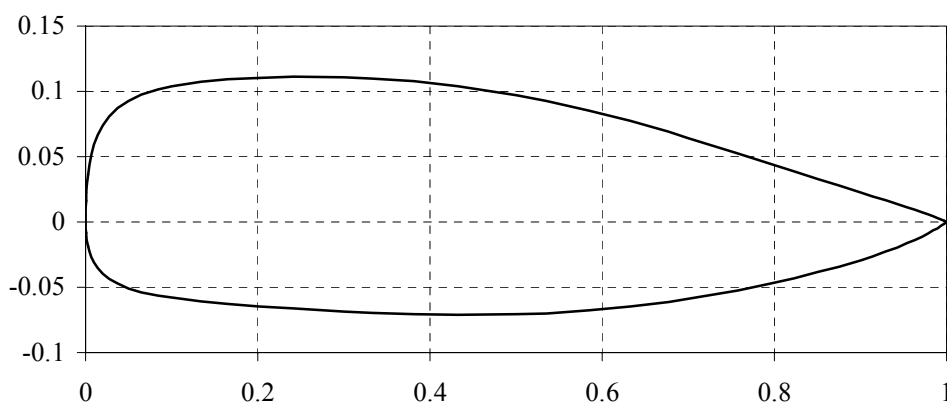
Έστω οι βαθμοί ελευθερίας  $f$  και  $f+1$ , με  $f=1,2,\dots,n-1$  (Εικόνα 29). Με βάση αυτούς μπορούμε να δημιουργήσουμε μια τετράγωνη διδιάστατη περιοχή με τετμημένη τον  $f$  βαθμό ελευθερίας και τεταγμένη τον  $f+1$  (Εικόνα 28). Κάθε τέτοια περιοχή αντιπροσωπεύει πλέον μια πόλη μέσα στην οποία είναι δυνατόν να σταθμεύσουν τα μυρμήγκια ταξιδεύοντας από πόλη σε πόλη. Ο συνολικός αριθμός των πόλεων είναι  $n$ . Η τελευταία πόλη, δηλαδή η  $n$ , έχει τετμημένη τον  $n$  βαθμό ελευθερίας και τεταγμένη τον  $1^o$  βαθμό ελευθερίας του προβλήματος.

Το μυρμήγκι συνεχίζει το ταξίδι του στη γειτονική πόλη έχοντας, όμως, πλέον γνωστή τουλάχιστον την τετμημένη ή την τεταγμένη της επόμενης θέσης του καθώς κάθε πόλη έχει στον άξονα των  $x$  τον βαθμό ελευθερίας που είχε η προηγούμενη στον άξονα των  $y$  (Εικόνα 30). Η διαδικασία αυτή συνεχίζεται μέχρι το «ψηφιακό» μυρμήγκι να επιστρέψει στη φωλιά του. Όταν το ταξίδι ολοκληρωθεί, υπολογίζεται η συνολική «απόσταση» του, εναποτίθεται η ανάλογη ποσότητα φερομόνης και απομνημονεύεται η ορατότητα.

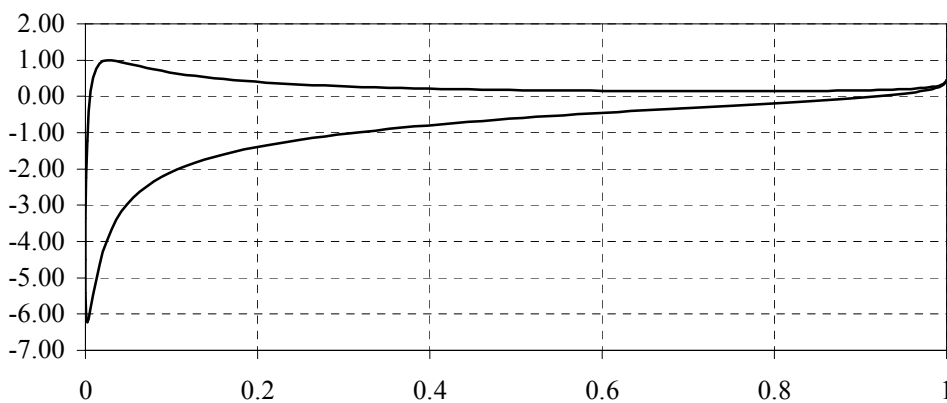
Το πρόβλημα της μετάβασης από τους διακριτούς αλγορίθμους στους συνεχείς εντοπίζεται βασικά στη διακριτοποίηση του χώρου. Στον EACO αυτό επιτυγχάνεται με τη διακριτοποίηση είτε του μονοδιάστατου είτε του διδιάστατου χώρου με ένα συγκεκριμένο αριθμό κόμβων. Πάνω στους κόμβους αυτούς εναποτίθεται η φερομόνη, αλλά και απομνημονεύεται η ορατότητα. Ενώ, όμως, η αποθήκευση των μεταβλητών γίνεται διακριτά, η χρήση τους γίνεται στο συνεχές πεδίο καθώς πραγματοποιούνται αναλυτικές προσεγγίσεις με πολυώνυμα (βλέπε παρακάτω Κεφάλαια 7.3 και 8.2.5).



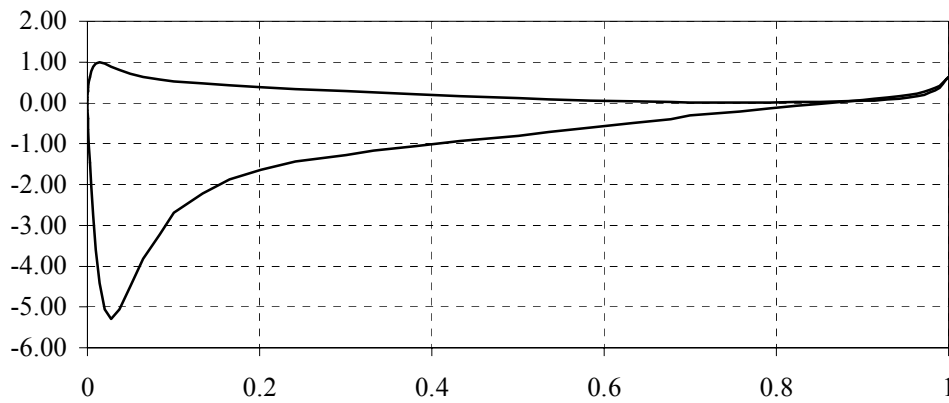
Εικόνα 23. Η αεροτομή NACA 0012.



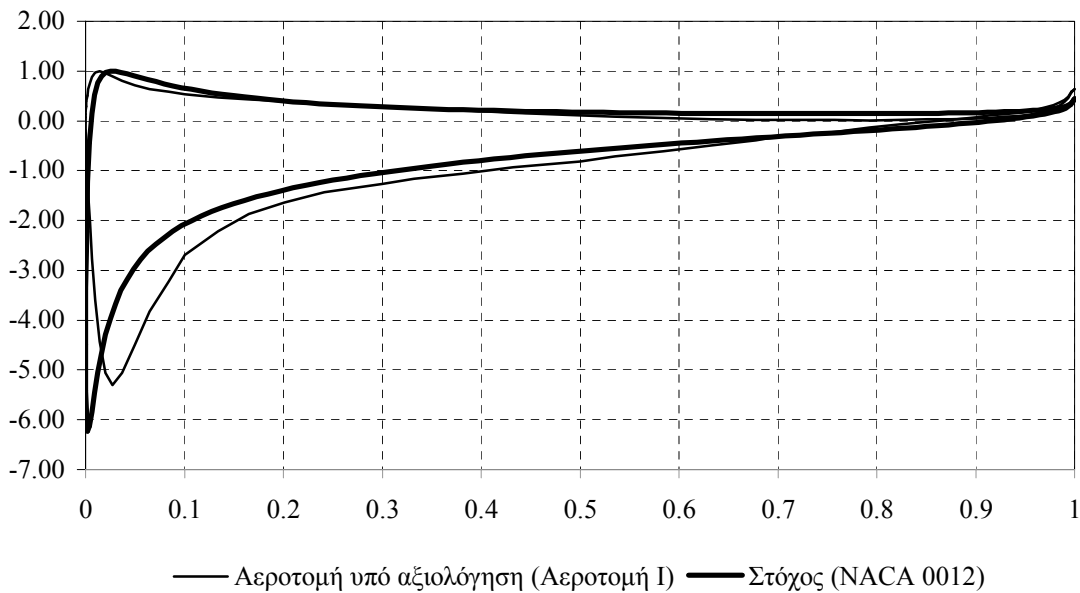
Εικόνα 24. Η αεροτομή (έστω αεροτομή I) που προκύπτει από τη «διαδρομή» του μυρμηγκιού που εντόπισε την καλύτερη λύση κατά τη δεύτερη επανάληψη του αλγορίθμου σε μια εκτέλεση του EACO.



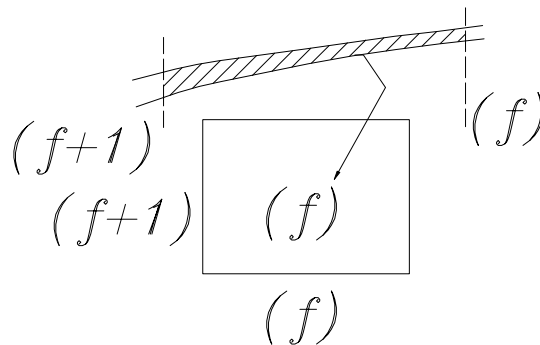
Εικόνα 25. Η κατανομή πιέσεων για αστρόβιλη και ασυμπίεστη ροή γύρω από την αεροτομή NACA 0012 (Εικόνα 23). Η γωνία προσβολής είναι  $10^\circ$  και η αδιάστατη ταχύτητα στο  $\infty$  είναι ίση με τη μονάδα.



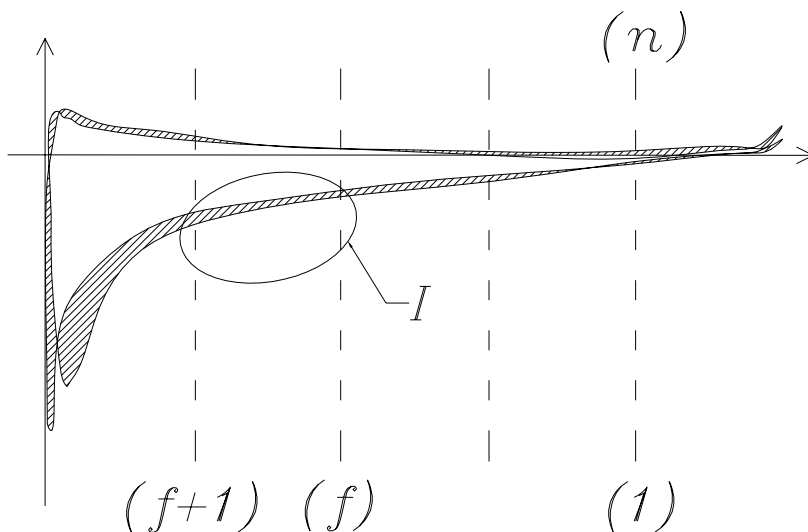
Εικόνα 26. Η κατανομή πιέσεων για αστρόβιλη και ασυμπιεστή ροή γύρω από την αεροτομή I (Εικόνα 24). Η γωνία προσβολής είναι  $10^\circ$  και η αδιάστατη ταχύτητα στο  $\infty$  είναι ίση με τη μονάδα.



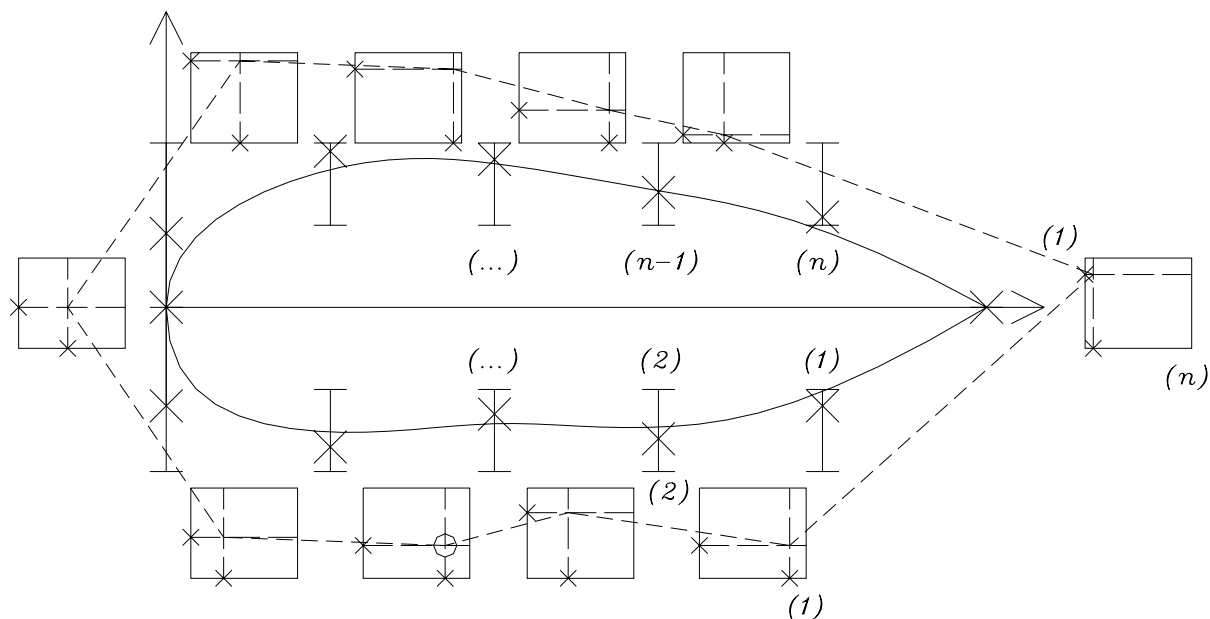
Εικόνα 27. Οι κατανομές πιέσεων των δύο αεροτομών (I και NACA 0012). Σκοπός μας είναι η ελαχιστοποίηση του εμβαδού που περικλείεται από αυτές τις δύο καμπύλες.



Εικόνα 28. Λεπτομέρεια I (Εικόνα 29). Η πόλη  $(f)$  που δημιουργείται από τους βαθμούς ελευθερίας  $(f)$  και  $(f+1)$ . Το γραμμοσκιασμένο εμβαδό αντιστοιχεί στην ορατότητα της πόλεως  $(f)$ .



Εικόνα 29. Οι κατανομές πιέσεων των δύο αεροτομών (I και NACA 0012). Με γραμμοσκίαση εμφανίζεται το εμβαδό που πρέπει να ελαχιστοποιηθεί. Με διακεκομμένη γραμμή εμφανίζονται οι περιοχές πάνω στις οποίες μπορούν να κινηθούν τα σημεία ελέγχου των καμπυλών Bezier (βαθμοί ελευθερίας).

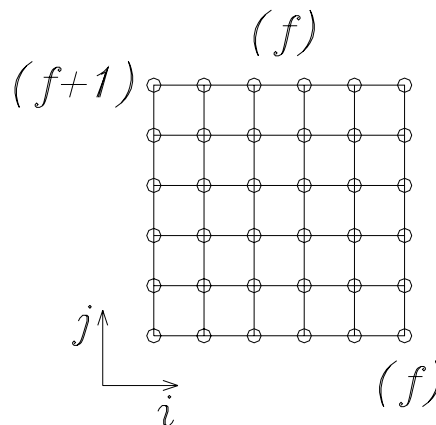


Εικόνα 30. Παρουσίαση των πόλεων που δημιουργούνται με δύο βαθμούς ελευθερίας. Με διακεκομμένη γραμμή παρουσιάζεται η κλειστή διαδρομή ενός μυρμηγκιού. Με (x) συμβολίζονται τα σημεία ελέγχου των καμπυλών Bezier, με (I) οι περιοχές στις οποίες μπορούν να κινηθούν αυτά (βαθμοί ελευθερίας) και με (O) η φωλιά του μυρμηγκιού. Το ψηφιακό μυρμηγκί μπορεί να κινηθεί μέσα σε κάθε τετράγωνο, αλλά πάντα με γνωστή τη μια συνιστώσα της θέσης του. Η διαδικασία που παρουσιάζεται στην παραπάνω εικόνα δίνει το ίδιο αποτέλεσμα με τη διαδικασία που παρουσιάζεται στην Εικόνα 22.

### 7.3 ΑΝΑΛΥΣΗ

Μετά τη γενική περιγραφή του EACO, ακολουθεί αναλυτικότερη παρουσίαση των μηχανισμών του αλγορίθμου.

Καταρχήν, για την υλοποίηση του αλγορίθμου απαιτούνται τρία πλέγματα για κάθε πόλη. Σε αυτά τα πλέγματα θα εναποτίθεται η φερομόνη, θα αποθηκεύεται η πιθανότητα και θα απομνημονεύεται η ορατότητα. Ο αριθμός  $N$  των **κόμβων** (*nodes*) κάθε βαθμού ελευθερίας που χρησιμοποιούνται για τη δημιουργία των πλεγμάτων είναι μια από τις σημαντικές παραμέτρους του αλγορίθμου.



Εικόνα 31. Το πλέγμα μιας πόλης  $f$  με  $N=6$  κόμβους σε κάθε βαθμό ελευθερίας ( $f$  και  $f+1$ ).

Η **αρχικοποίηση** (*initialization*) των επιπέδων της φερομόνης γίνεται με μια τιμή  $\tau_0$ , ενώ της ορατότητας με τη μηδενική τιμή καθώς το πρόβλημα IDS είναι δυναμικό και η ορατότητα δεν είναι γνωστή εκ των προτέρων. Επομένως, είναι απαραίτητο τα μυρμηγκία να πραγματοποιήσουν μερικά τελείως τυχαία ταξίδια και με αυτό τον τρόπο να καθορίσουν τις πρώτες τιμές της ορατότητας καθώς και της φερομόνης. Μια από τις παραμέτρους, λοιπόν, του αλγορίθμου είναι ο αριθμός των μυρμηγκιών ( $m_{init}$ ) που θα κάνουν τυχαία ταξίδια. Προφανώς, ο αριθμός αυτός δεν μπορεί να είναι ούτε πολύ μεγάλος, καθώς για παράδειγμα ο υπολογισμός της κατανομής των πιέσεων με χρήση των εξισώσεων Navier-Stokes είναι χρονοβόρος, αλλά ούτε και πολύ μικρός γιατί τότε θα αργήσει ο αλγόριθμος να εντοπίσει κάποιες καλές λύσεις.

Αφού ολοκληρωθεί η αρχικοποίηση ξεκινούν τα ταξίδια των μυρμηγκιών. Ο συνολικός **αριθμός των μυρμηγκιών** που ταξιδεύουν ανά επανάληψη - *iteration* (ή κύκλο - *cycle*) επιλέγεται να είναι ίσος με τον αριθμό των πόλεων, δηλαδή  $m=n$ . Η επιλογή αυτή βασίζεται σε προηγούμενες εργασίες [20], [22]. Εξάλλου, ειδικά στο πρόβλημά μας, η επιλογή αυτή είναι η

ενδεδειγμένη καθώς κάθε μυρμήγκι ξεκινάει εκμεταλλευόμενο την εμπειρία της πόλης του, γεγονός που θα έκανε δύο ή περισσότερα μυρμήγκια να ξεκινούν και να τελειώνουν το ταξίδι τους στο ίδιο σημείο (φωλιά) σε κάθε επανάληψη πραγματοποιώντας μειωμένη εξερεύνηση (*exploration*) του χώρου και άρα σπατάλη υπολογιστικού χρόνου. Επίσης, σημαντικό είναι το γεγονός ότι τα υπολογιστικά εργαλεία της ρευστομηχανικής είναι εξαιρετικά χρονοβόρα με αποτέλεσμα μεγάλος αριθμός μυρμηγκιών να οδηγεί σε χρονοβόρες εκτελέσεις του προγράμματος.

Στην αρχή κάθε επανάληψης (*iteration*) του αλγορίθμου υπολογίζεται η **πιθανότητα εκλογής κάθε κόμβου** όλων των πόλεων με βάση τη συγκεντρωμένη εμπειρία της αποικίας, δηλαδή την ορατότητα και τα επίπεδα της φερομόνης. Η πιθανότητα αυτή, όπως και σε όλους τους αλγορίθμους ACO, υπολογίζεται μέσω του *τυχαίου-αναλογικού κανόνα μετάβασης* και αποθηκεύεται στους κόμβους του αντίστοιχου πλέγματος της πόλης. Ο *τυχαίος-αναλογικός κανόνας μετάβασης* στον EACO είναι:

$$p_{cij}(t) = \frac{[\tau_{cij}(t)]^\alpha \cdot [\eta_{cij}(t)]^\beta}{\sum_{l,k=1}^N ([\tau_{clk}(t)]^\alpha \cdot [\eta_{clk}(t)]^\beta)} \quad \text{Εξ. 24}$$

όπου:

$p_{cij}(t)$  είναι η πιθανότητα του κόμβου  $(i,j)_c$  της πόλης  $c$  κατά την επανάληψη  $t$ .

$\tau_{cij}(t)$  είναι η φερομόνη του κόμβου  $(i,j)_c$  της πόλης  $c$  κατά την επανάληψη  $t$ .

$\eta_{cij}(t)$  είναι η ορατότητα του κόμβου  $(i,j)_c$  της πόλης  $c$  κατά την επανάληψη  $t$ .

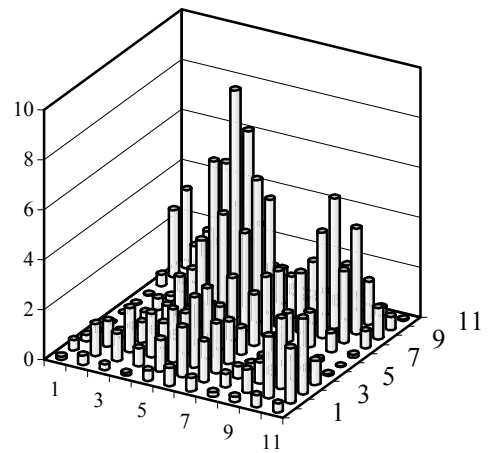
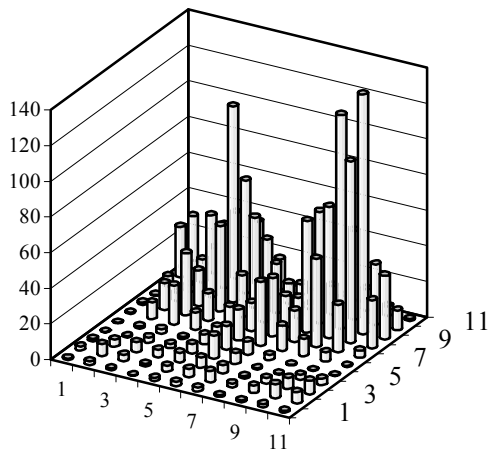
$\alpha, \beta$  είναι παράμετροι που εκφράζουν το ποσοστό επίδρασης της φερομόνης και της ορατότητας στον υπολογισμό της πιθανότητας του κόμβου  $(i,j)_c$ .

Η **θέση της φωλιάς** (*nest*) των μυρμηγκιών σε κάθε πόλη επιλέγεται στο σημείο με τη μέγιστη πιθανότητα. Η επιλογή<sup>29</sup> αυτή γίνεται έτσι ώστε το μυρμήγκι να εκμεταλλευθεί τη βέλτιστη τοπική λύση και κατόπιν να εξερευνήσει τον υπόλοιπο χώρο. Η διαδικασία έχει ως εξής, επιλέγεται ο κόμβος της πόλεως με τη μεγαλύτερη πιθανότητα (*κανόνας μέγιστης πιθανότητας – maximum probability rule*):

<sup>29</sup> Κάτι αντίστοιχο γίνεται όπως έχουμε ήδη δει και στον ACS.

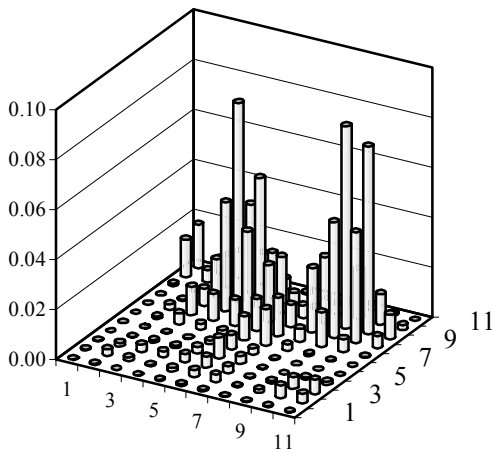
$$(i, j)_c^{best} = \max([\tau_{cij}(t)]^\alpha \cdot [\eta_{cij}(t)]^\beta) \tag{Εξ. 25}$$

και γίνεται παρεμβολή (βλέπε Κεφάλαιο 8.2.5) με τα γύρω σημεία έτσι ώστε η καμπύλη ή η επιφάνεια που θα προκύψει να δίνει το μέγιστο της περιοχής.

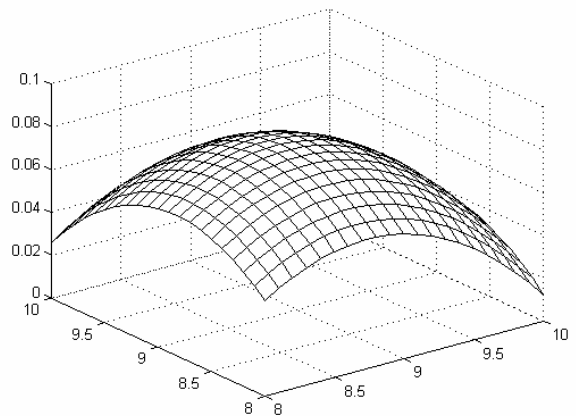


Εικόνα 32. Η κατανομή της ορατότητας στους κόμβους του πλέγματος στην 1<sup>η</sup> πόλη κατά την 2<sup>η</sup> επανάληψη του αλγορίθμου σε μια εκτέλεση του EACO.

Εικόνα 33. Η κατανομή της φερομόνης στους κόμβους του πλέγματος στην 1<sup>η</sup> πόλη κατά την 2<sup>η</sup> επανάληψη του αλγορίθμου σε μια εκτέλεση του EACO.

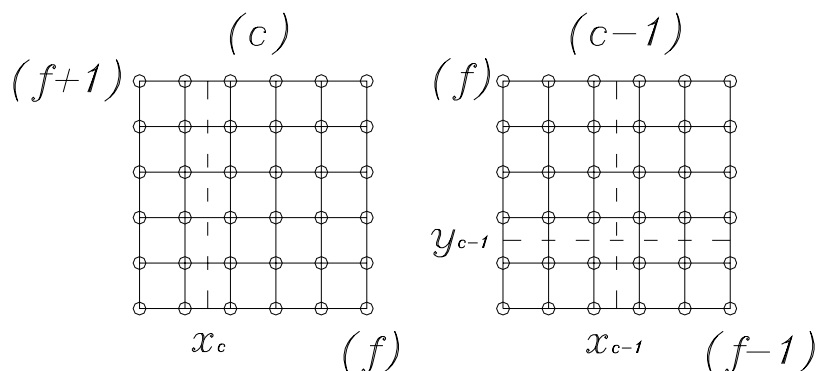


Εικόνα 34. Η κατανομή της πιθανότητας στους κόμβους του πλέγματος στην 1<sup>η</sup> πόλη κατά την 2<sup>η</sup> επανάληψη του αλγορίθμου σε μια εκτέλεση του EACO. Ο κόμβος με τη μέγιστη πιθανότητα εκλογής είναι ο (9,9) με τιμή 0.0811.



Εικόνα 35. Παράδειγμα παρεμβολής επιφάνειας σε 5 σημεία γύρω από τον κόμβο με τη μέγιστη πιθανότητα εκλογής – κόμβος (9,9) (Εικόνα 34). Το μέγιστο της επιφάνειας έχει τιμή ίση με 0.0831 και βρίσκεται στο σημείο (8.768,8.877). Το σημείο αυτό αποτελεί τη «φωλιά» των μυρμηγκιών στη συγκεκριμένη πόλη (1<sup>η</sup> πόλη) κατά τη 2<sup>η</sup> επανάληψη του αλγορίθμου.

Παρακάτω, περιγράφεται η επιλογή της θέσης του μυρμηγκιού στην επόμενη πόλη του ταξιδιού του. Έστω ότι το μυρμήγκι  $k$  βρίσκεται στη θέση  $(x,y)_{c-1}$  στην πόλη<sup>30</sup>  $c-1$  με τους βαθμούς ελευθερίας  $f-1$  (άξονας  $x$ ) και  $f$  (άξονας  $y$ ) να προσδιορίζουν την επιφάνειά της. Τότε, οι βαθμοί ελευθερίας που θα ορίζουν την επιφάνεια της επόμενης πόλης  $c$  θα είναι οι  $f$  (άξονας  $x$ ) και  $f+1$  (άξονας  $y$ ).



Εικόνα 36. Το πρώτο στάδιο της μετακίνησης του «ψηφιακού» μυρμηγκιού από την πόλη  $c-1$  στην πόλη  $c$ .

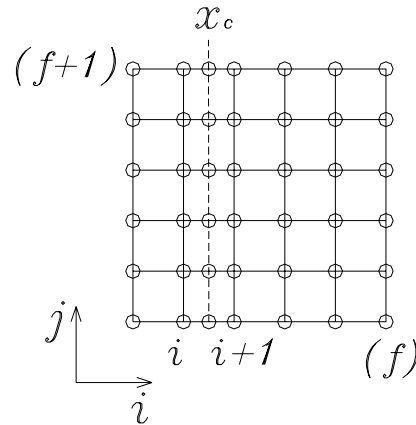
Επομένως, θα είναι  $x_c=y_{c-1}$ , δηλαδή είναι γνωστή η τετμημένη της νέας θέσης του μυρμηγκιού στην πόλη  $c$ , και μένει να προσδιορισθεί η τεταγμένη  $y_c$ . Για να μπορούμε να επιλέξουμε ένα σημείο πάνω στην ευθεία  $x=x_c$ , θα πρέπει να έχουμε κάποια δεδομένα για τα επίπεδα της φερομόνης και της ορατότητας πάνω στην ευθεία αυτή. Όμως, η συντεταγμένη  $x_c$  καθώς είναι ένα συνεχές μέγεθος σπάνια συμπίπτει με κάποιον από τους κόμβους του πλέγματος. Συνεπώς, πρέπει να δημιουργήσουμε κόμβους πάνω στην ευθεία  $x=x_c$  όπου και θα προσδιορίζεται η πιθανότητα εκλογής τους.

Έστω, λοιπόν, ότι η συντεταγμένη  $x_c$  βρίσκεται ανάμεσα στους κόμβους  $(i,l)_c$  και  $(i+1,l)_c$ . Τότε για κάθε  $j$  με  $j=1,2,\dots,N$  εκτελούμε γραμμική παρεμβολή μεταξύ των κόμβων  $(i,j)_c$  και  $(i+1,j)_c$  και του σημείου  $(x_c,j)_c$  έτσι ώστε να προσδιορίσουμε την πιθανότητα στους νέους κόμβους που δημιουργούνται πάνω στην ευθεία  $x=x_c$  (Εικόνα 37). Η σχέση που καθορίζει την πιθανότητα  $p_j^1$  του κόμβου  $j$  πάνω στην ευθεία  $x=x_c$  είναι η εξής:

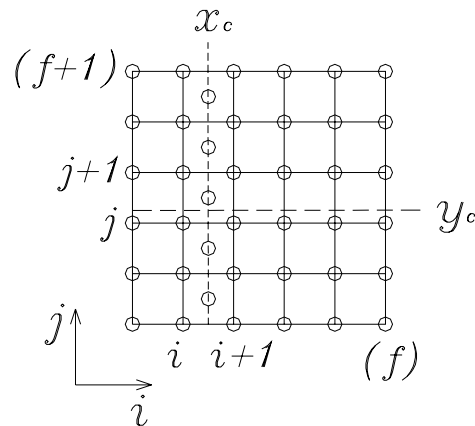
<sup>30</sup> Σημείωση: η πόλη  $c$  θα έχει τον ίδιο αύξοντα αριθμό με αυτόν του βαθμού ελευθερίας του άξονα  $x$  της πόλης, δηλαδή τον  $f$ . Συνεπώς  $c=f$ .

$$p_j^1 = (x_c - i) \cdot p_{c(i+1)j}(t) + [(i+1) - x_c] \cdot p_{cij}(t)$$

Εξ. 26



Εικόνα 37. Το δεύτερο στάδιο της μετακίνησης του «ψηφιακού» μυρμηγκιού από την πόλη  $c-1$  στην πόλη  $c$ . Παρουσιάζεται η ευθεία  $x=x_c$  και οι προσωρινοί κόμβοι που δημιουργούνται για την αποθήκευση της πιθανότητας  $p_j^1$  πάνω στην ευθεία  $x=x_c$ .



Εικόνα 38. Το τρίτο στάδιο της μετακίνησης του «ψηφιακού» μυρμηγκιού από την πόλη  $c-1$  στην πόλη  $c$ . Παρουσιάζονται οι προσωρινοί κόμβοι που δημιουργούνται για την αποθήκευση της πιθανότητας  $p_j^2$  πάνω στην ευθεία  $x=x_c$  και η τυχαία επιλογή της συντεταγμένης  $y_c$  στην περιοχή  $(x_cj)_c$  με  $(x_cj+1)_c$ .

Κατόπιν, υπολογίζουμε την πιθανότητα  $p_j^2$  στις θέσεις που βρίσκονται στο μέσο της απόστασης των κόμβων  $(x_cj)_c$  και  $(x_cj+1)_c$  με  $j=1,2,\dots,N-1$  χρησιμοποιώντας τη σχέση:

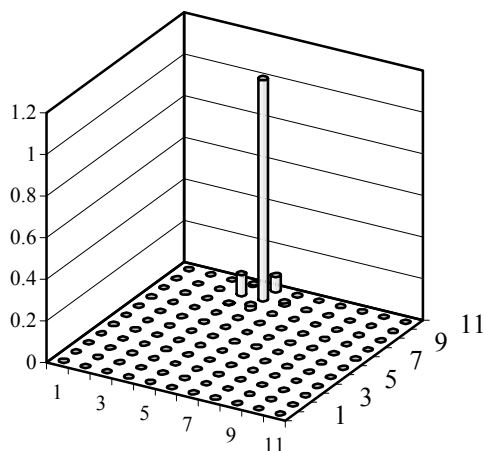
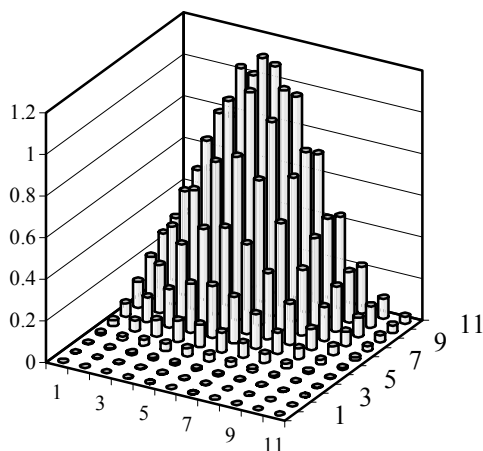
$$p_j^2 = \frac{p_j^1 + p_{j+1}^1}{2}$$

Εξ. 27

Με αυτόν τον τρόπο δημιουργούνται νέοι κόμβοι (Εικόνα 38). Αφού, λοιπόν, επιλεγεί τυχαία ένας κόμβος  $(x_cj)_c$  με βάση τις πιθανότητες  $p_j^2$ , στη συνέχεια με μια γεννήτρια τυχαίων

αριθμών επιλέγουμε μια θέση στην περιοχή που ορίζεται από τους κόμβους  $(x_{c,j})_c$  και  $(x_{c,j+1})_c$  η οποία μετασχηματίζεται στην τεταγμένη  $y_c$ .

Όταν ολοκληρώσει το ταξίδι του κάθε μυρμήγκι, θα πρέπει να εναποθέσει τη φερομόνη αλλά και να απομνημονεύσει την ορατότητα. Όπως έχει ήδη αναφερθεί κάθε πόλη διακριτοποιείται με ένα αριθμό κόμβων  $N$  σε κάθε πλευρά. Στους κόμβους αυτούς εναποτίθεται η φερομόνη και απομνημονεύεται η ορατότητα έτσι ώστε αργότερα να είναι δυνατή η παρεμβολή μεταξύ των κόμβων για τον υπολογισμό των πιθανοτήτων.



Εικόνα 39. Εναπόθεση φερομόνης  $T^1(0)=1.174$  στο σημείο  $(8.196, 10.095)$  με  $r=0.3$  σε πλέγμα με  $N=11$ .

Εικόνα 40. Εναπόθεση φερομόνης  $T^1(0)=1.174$  στο σημείο  $(8.196, 10.095)$  με  $r=0.05$  σε πλέγμα με  $N=11$ .

Για την εναπόθεση της φερομόνης επιλέχθηκε η χρήση εκθετικής κατανομής (κανόνας εναπόθεσης της φερομόνης – *pheromone deposit rule*):

$$\Delta \tau_{cij}^k(t) = T^k(t) \cdot \exp\left(-\left(d_{cij}^k\right)^2 / r\right) \tag{Εξ. 28}$$

όπου:

$T^k(t)$  η ποσότητα της φερομόνης που εναποθέτει το μυρμήγκι  $k$  κατά την επανάληψη  $t$ .

$$T^k(t) = 1 / (c_f + E^k(t)) \tag{Εξ. 29}$$

$E^k(t)$  το εμβαδόν που περικλείεται μεταξύ της καμπύλης πίεσεως του στόχου και της καμπύλης πίεσεως της αεροτομής που δημιούργησε το μυρμήγκι  $k$  κατά την επανάληψη  $t$  (το γραμμοσκιασμένο εμβαδό Εικόνα 29).

$c_f$  μια παράμετρος που λαμβάνει συνήθως μικρές τιμές. Η χρήση της έχει σκοπό να αποφύγει τα υπολογιστικά σφάλματα που οφείλονται σε πολύ μικρές τιμές εμβαδού.

$d_{cij}^k$  η απόσταση στην πόλη  $c$  μεταξύ της θέσης  $(x,y)_c$  που στάθμευσε το μυρμήγκι  $k$  και του κόμβου  $(i,j)_c$ .

$r$  μια παράμετρος που εκφράζει το ρυθμό απόσβεσης της εκθετικής κατανομής.

Η **απομνημόνευση της ορατότητας** (*visibility*) για τη χρονική στιγμή (επανάληψη)  $t+1$  διέπεται από την εξής σχέση (*κανόνας ανανέωσης της ορατότητας – visibility update rule*):

$$\eta_{cij}(t+1) = \max(\eta_{cij}^k(t), \eta_{cij}(t)) \quad \text{Εξ. 30}$$

όπου:

$\eta_{cij}(t)$  η ορατότητα που υπάρχει στους κόμβους  $(i,j)_c$  της πόλης  $c$  στην αρχή της επανάληψης  $t$ .

$\eta_{cij}^k(t)$  η νέα ορατότητα που υπολογίζεται από το μυρμήγκι  $k$  και κατανέμεται εκθετικά στους κόμβους  $(i,j)_c$  της πόλης  $c$  κατά την επανάληψη  $t$ .

$$\eta_{cij}^k(t) = \eta_c^k(t) \cdot \exp\left(-\left(d_{cij}^k\right)^2 / r\right) \quad \text{Εξ. 31}$$

$\eta_c^k(t)$  η ορατότητα που υπολογίζει το μυρμήγκι  $k$  στην πόλη  $c$  κατά την επανάληψη  $t$ .

$$\eta_c^k(t) = 1 / (c_v + E_c^k(t)) \quad \text{Εξ. 32}$$

$E_c^k(t)$  το εμβαδό που περικλείεται στη γειτονιά της πόλης  $c$  μεταξύ της καμπύλης πίεσεως του στόχου και της καμπύλης πίεσεως της αεροτομής που δημιουργήσε το μυρμήγκι  $k$  κατά την επανάληψη  $t$ . Πιο συγκεκριμένα το εμβαδό αυτό ισούται με το εμβαδό που περικλείεται από τις δύο καμπύλες πιέσεων μεταξύ δύο βαθμών ελευθερίας των πολυωνύμων Bezier (το γραμμοσκιασμένο εμβαδό Εικόνα 28).

$c_v$  μια παράμετρος που λαμβάνει συνήθως μικρές τιμές. Η χρήση της έχει σκοπό να αποφύγει τα υπολογιστικά σφάλματα που οφείλονται σε πολύ μικρές τιμές εμβαδού.

Όταν όλα τα μυρμήγκια ολοκληρώσουν τα ταξίδια τους, θα πρέπει να εφαρμοσθεί ο **κανόνας ανανέωσης της φερομόνης** (*pheromone update rule*). Ο κανόνας αυτός στην περίπτωση του EACO γίνεται:

$$\tau_{cij}(t+1) = (1 - \rho)\tau_{cij}(t) + \Delta\tau_{cij}(t) \quad \text{Εξ. 33}$$

όπου:

$\rho$  με  $0 \leq \rho < 1$  ο συντελεστής εξάτμισης της φερομόνης.

$$\Delta \tau_{cij}(t) = \sum_{k=1}^m \Delta \tau_{cij}^k(t) \quad \text{Εξ. 34}$$

Η επέκταση της παραπάνω σχέσης ώστε να περιλαμβάνει «εκλεκτά» μυρμήγκια (*elitist ants*) όπως στον AS<sub>e</sub> είναι άμεση. Όπως επίσης, άμεση είναι και η αλλαγή της σχέσης ώστε να επιτρέπει μόνο στο μυρμήγκι που έχει βρει τη βέλτιστη έως τώρα λύση (*global best ant*) να εναποθέτει φερομόνη (ACS) ή σε συνδυασμό του μυρμηγκιού που έχει βρει τη βέλτιστη έως τώρα λύση και του μυρμηγκιού που βρίσκει τη βέλτιστη λύση κάθε επανάληψης (*iteration best ant*) (MMAS).

Όταν για ένα συγκεκριμένο αριθμό επαναλήψεων ο αλγόριθμος δεν επιτυγχάνει να εντοπίσει νέες βέλτιστες λύσεις, θα πρέπει να ενεργοποιήσουμε τους μηχανισμούς ομαλοποίησης της φερομόνης ή/και επαναρχικοποίησης της φερομόνης και της ορατότητας.

Η **ομαλοποίηση της φερομόνης** (*smoothing of pheromone trail*) έχει ως σκοπό την αύξηση των επιπέδων της φερομόνης μέσα σε κάθε πόλη ανάλογα με τη διαφορά τους από το  $\tau_{gl}$  έτσι ώστε να αυξηθεί η πιθανότητα επιλογής θέσεων με χαμηλά επίπεδα φερομόνης. Σημαντική παράμετρος του μηχανισμού είναι ο αριθμός των επαναλήψεων, έστω  $n_{ts}$ , χωρίς την εύρεση νέας βέλτιστης τιμής για τον οποίο ενεργοποιείται ο μηχανισμός.

Η σχέση που υλοποιεί την παραπάνω διαδικασία έχει ως εξής:

$$\tau_{cij}(t) \leftarrow \tau_{cij}(t) + \delta \cdot (\tau_{gl}(t) - \tau_{cij}(t)) \quad \text{Εξ. 35}$$

όπου:

$\delta \in [0, 1]$ . Για  $\delta=1$ , γίνεται  $\tau_{cij}=\tau_{gl}$  για κάθε  $c, i, j$  και επομένως έχουμε επαναρχικοποίηση των επιπέδων της φερομόνης σε μια μεγάλη τιμή, ενώ για  $\delta=0$  απενεργοποιείται ο μηχανισμός αυτός.

$\tau_{gl}$  είναι μια δυναμικώς μεταβαλλόμενη ποσότητα και μπορεί να αντιπροσωπεύει τη μέγιστη ποσότητα φερομόνης σε κάθε πόλη (ή τη μέγιστη ποσότητα φερομόνης όλων των πόλεων), τη ποσότητα φερομόνης που εναποθέτει στη διαδρομή του το μυρμήγκι που έχει βρει τη βέλτιστη έως τώρα λύση (*global best ant* ή *global leader*) ή τέλος το άνω όριο της φερομόνης  $\tau_{max}$  σε περίπτωση που χρησιμοποιείται ο MMEACO.

Η **επαναρχικοποίηση** (*reinitialization*) των επιπέδων της φερομόνης και της ορατότητας γίνεται με σκοπό την επανεκκίνηση του αλγορίθμου ελπίζοντας ότι μια νέα τυχαία αλληλουχία

αριθμών θα δώσει καλύτερα αποτελέσματα στην έρευνα. Προφανώς, ο μηχανισμός υλοποιείται με τη σχέση:  $\tau_{cij}=\tau_o$  (σε μια διαφορετική έκδοση μπορεί να χρησιμοποιηθεί και η σχέση:  $\eta_{cij}=0$ ).

Σε μια προσπάθεια βελτίωσης της ποιότητας του αλγορίθμου EACO χρησιμοποιήθηκαν **άνω και κάτω όρια στην ποσότητα της φερομόνης**. Θα ονομάζουμε αυτή την έκδοση του EACO ως MMEACO (Max Min Extended Ant Colony Optimization). Ο σκοπός του άνω ορίου είναι να μην επιτρέψει σε κάποιες περιοχές να συγκεντρώσουν πολύ φερομόνη με συνέπεια να παρουσιάζουν μεγάλη πιθανότητα επιλογής και άρα να ελκύουν το μεγαλύτερο μέρος του πληθυσμού των μυρμηγκιών. Επιπλέον, το κάτω όριο της φερομόνης διασφαλίζει το ότι καμία περιοχή στις πόλεις δεν θα έχει μηδενική ή περίπου μηδενική πιθανότητα εκλογής. Με τις αλλαγές αυτές διασφαλίζεται η καλύτερη εξερεύνηση (*exploration*) του χώρου των λύσεων χωρίς όμως να μειώνεται η εκμετάλλευση (*exploitation*) των περιοχών που έχουν δώσει καλές λύσεις.

Τα άνω όριο της φερομόνης στην περίπτωση του MMEACO ορίζεται με την παρακάτω σχέση:

$$\tau_{\max} = \frac{1}{\rho} T_{best}(t) \quad \text{Εξ. 36}$$

όπου  $T_{best}(t)$  η ποσότητα της φερομόνης της βέλτιστης έως τώρα λύσης. Συνεπώς, το  $\tau_{\max}$  είναι δυναμικώς μεταβαλλόμενο.

Το κάτω όριο της φερομόνης δεν μπορεί να ορισθεί με σχέση παρόμοια με αυτήν του MMAS (Εξ. 17) καθώς το πρόβλημα της Αντίστροφης Σχεδίασης Αεροτομών είναι συνεχές και όχι διακριτό. Συνεπώς, το κάτω όριο  $\tau_{\min}$  θα πρέπει να καθορίζεται από το χρήστη είτε ως τιμή είτε ως ποσοστό επί του άνω ορίου της φερομόνης  $\tau_{\max}$ .

Κλείνοντας το κεφάλαιο αυτό παρουσιάζουμε συνοπτικά τη διαδικασία υλοποίησης του EACO για την επίλυση του IDS όπως επίσης και τον αντίστοιχο ψευτοκώδικα (Κώδικας 3) για άμεση προγραμματιστική ανάπτυξη του αλγορίθμου.

## Έναρξη Αλγορίθμου

### Βήμα 1<sup>ο</sup>:

Θέτουμε τη φερομόνη και την ορατότητα στις αρχικές τους τιμές σε όλες τις πόλεις.

Κάθε μυρμήγκι από τα  $m_{init}$  πραγματοποιεί ένα τυχαίο ταξίδι.

Με το πέρας του ταξιδιού του, κάθε μυρμήγκι εναποθέτει φερομόνη (Εξ. 33 με  $\rho=0$ ) και υπολογίζει την ορατότητα (Εξ. 30) στις περιοχές που επισκέφθηκε.

Υπολογίζουμε την πιθανότητα εκλογής κάθε κόμβου σε κάθε πόλη  $p_{cij}$  (Εξ. 24).

### Βήμα 2<sup>ο</sup>:

Για κάθε μυρμήγκι από τα  $m$  επιλέγουμε τη «φωλιά» του στην αρχική πόλη.

Η «φωλιά» βρίσκεται στο σημείο με τη μέγιστη πιθανότητα εκλογής στην πόλη.

Το σημείο αυτό εντοπίζεται πάνω στην επιφάνεια που προσδιορίζει την κατανομή της πιθανότητας εκλογής κάθε σημείου και δημιουργείται με παρεμβολή των κόμβων που βρίσκονται στη γειτονία του κόμβου που προκύπτει από την Εξ. 25.

### Βήμα 3<sup>ο</sup>:

Κάθε μυρμήγκι μετακινείται στην επόμενη πόλη μέχρι να ολοκληρώσει το «ταξίδι» του, δηλαδή να επιστρέψει στη «φωλιά» του.

Σε κάθε τέτοια μετακίνηση το μυρμήγκι γνωρίζει την τετμημένη της επόμενης θέσης του η οποία είναι ίδια με την τεταγμένη της προηγούμενης.

Την τεταγμένη στη νέα θέση την επιλέγει τυχαία στη γειτονία του κόμβου που προκύπτει από την Εξ. 27.

### Βήμα 4<sup>ο</sup>:

Για κάθε «ταξίδι» των μυρμηγκιών δημιουργούμε το περίγραμμα της αεροτομής που προκύπτει με χρήση των πολυωνύμων Bezier- Bernstein.

Υπολογίζουμε την κατανομή των πιέσεων γύρω από κάθε αεροτομή με χρήση της μεθόδου των ιδιόμορφων σημείων ή με επίλυση των εξισώσεων Euler ή Navier-Stokes.

Συγκρίνουμε την κατανομή των πιέσεων κάθε νέας αεροτομής με αυτή του στόχου.

### Βήμα 5<sup>ο</sup>:

Συγκρατούμε το ολικό βέλτιστο και την αντίστοιχη διαδρομή του μυρμηγκιού.

*Βήμα 6<sup>ο</sup>:*

Πραγματοποιούμε τοπική έρευνα αν χρειάζεται.

*Βήμα 7<sup>ο</sup>:*

Ανανεώνουμε τη φερομόνη στις περιοχές που επισκέφτηκαν τα μυρμηγκία με βάση τον κανόνα ανανέωσης της φερομόνης (Εξ. 33).

Ανανεώνουμε την ορατότητα στις περιοχές που επισκέφτηκαν τα μυρμηγκία με βάση τον κανόνα ανανέωσης της ορατότητας (Εξ. 30).

Υπολογίζουμε την πιθανότητα εκλογής κάθε κόμβου σε κάθε πόλη  $p_{cij}$  (Εξ. 24).

*Βήμα 8<sup>ο</sup>:*

Πραγματοποιούμε ομαλοποίηση της φερομόνης (Εξ. 35) αν χρειάζεται.

Πραγματοποιούμε επαναρχικοποίηση του αλγορίθμου αν χρειάζεται.

*Βήμα 9<sup>ο</sup>:*

Επαναλαμβάνουμε τη διαδικασία από το βήμα 2 μέχρι έως ότου ολοκληρωθεί ένας συγκεκριμένος αριθμός επαναλήψεων ή επιτευχθεί κάποιο κριτήριο σύγκλισης.

*Βήμα 10<sup>ο</sup>:*

Εκτυπώνουμε τη βέλτιστη αεροτομή που βρέθηκε καθώς και την αντίστοιχη κατανομή των πιέσεων.

**Ολοκλήρωση Αλγορίθμου.**

## Κώδικας 3. Ψευτοκώδικας EACO για το IDS

```

Start:

// Initialize
For  $c: 1$  to  $n$ 
  For  $i, j: 1$  to  $N$ 
    Set  $\tau_{cij}(0) = \tau_o$ 
    Set  $\eta_{cij}(0) = 0$ 
  End for
End for
Initialize minimum tour “length”:  $E_{best}$ 
For  $k: 1$  to  $m_{init}$ 
  Perform Random Tour
  For  $c: 1$  to  $n$ 
    For  $i, j: 1$  to  $N$ 
      Update pheromone trails  $\tau_{cij}$  by applying pheromone update rule with  $\rho = 0$ 
      Update visibility  $\eta_{cij}$  with visibility update rule
    End for
  End for
End for
For  $c: 1$  to  $n$ 
  For  $i, j: 1$  to  $N$ 
    Calculate probability  $p_{cij}$  with random proportional transition rule
  End for
End for

// Main Loop
For  $t: 1$  to maximum iterations

  // Create Tour
  For  $k: 1$  to  $m$ 
    Select nest  $(x_c, y_c)_c$  of ant  $k$  in town  $c = k$  by applying maximum probability rule and by
    interpolating around best node
    For counter:  $1$  to  $n-1$ 
      Set  $x_{c+1} = y_c$ 
      Select next town  $c = c+1$  (if  $c > n$  then  $c = 1$ )
      Select node  $(x_c, j)_c$  in town  $c$  by applying roulette wheel with probabilities  $p_j^2$ 
      Select randomly a position  $(x_c, y_c)_c$  in the neighborhood of  $(x_c, j)_c$ 
    End for
  End for

  // Evaluate tour “length”
  For  $k: 1$  to  $m$ 
    Construct airfoil contour with Bezier polynomials
    Calculate pressure distribution by applying Panel Method or by solving Euler or
    Navier-Stokes Equations
    Compare with target => Calculate  $E^k(t)$  and  $E_c^k(t)$ 
    If  $E^k(t) < E_{best}$  set  $E_{best} = E^k(t)$  and keep town positions of best tour
  End for
  Perform Local Search if necessary

```

```
// Update variables
For  $k: 1$  to  $m$ 
  For  $c: 1$  to  $n$ 
    For  $i, j: 1$  to  $N$ 
      Update pheromone trails  $\tau_{cij}$  by applying pheromone update rule
      Update visibility  $\eta_{cij}$  with visibility update rule
    End for
  End for
  For  $c: 1$  to  $n$ 
    For  $i, j: 1$  to  $N$ 
      Calculate probability  $p_{cij}$  with random proportional transition rule
    End for
  End for
  Apply Trail Smoothing if necessary
  Apply Trail Reinitialization if necessary

End for

Print  $E_{best}$  and best tour town positions

Stop
```



## 8 ΑΠΟΤΕΛΕΣΜΑΤΑ ΧΡΗΣΗΣ ΤΟΥ ΕΑCO

### 8.1 ΕΙΣΑΓΩΓΗ

Στο παρόν κεφάλαιο θα γίνει παράθεση, ανάλυση και σύγκριση των αποτελεσμάτων όπως αυτά προκύπτουν από διάφορες εκτελέσεις του αλγορίθμου σε διάφορες αεροτομές με διαφορετικές οριακές συνθήκες. Σε πρώτη φάση θα γίνει ρύθμιση (*tuning*) του αλγορίθμου ΕΑCO έτσι ώστε να προσδιορισθούν με σχετική ακρίβεια οι τιμές των παραμέτρων που θα δίνουν τα βέλτιστα αποτελέσματα και, τέλος, θα παρουσιασθούν τα αποτελέσματα των εκτελέσεων του αλγορίθμου για ένα αριθμό αεροτομών.

### 8.2 ΡΥΘΜΙΣΗ ΤΟΥ ΑΛΓΟΡΙΘΜΟΥ ΕΑCO

Για τη ρύθμιση (*tuning*) του ΕΑCO χρησιμοποιείται η υπόθεση *ceteris paribus* με βάση την οποία μεταβάλλεται πάντοτε μόνο ένα χαρακτηριστικό του αλγορίθμου και επομένως οποιαδήποτε αλλαγή στην απόδοσή του μπορεί να αποδοθεί μόνο σε αυτό το χαρακτηριστικό. Για την ορθότερη εκτίμηση των αποτελεσμάτων, θα πραγματοποιούνται 20 εκτελέσεις του αλγορίθμου και θα λαμβάνεται ο μέσος όρος αυτών. Ο μέγιστος αριθμός των επαναλήψεων (*maximum iterations*) κάθε εκτέλεσης του αλγορίθμου τίθεται ίσος με 200.

Τα αποτελέσματα στους πίνακες δείχνουν το μέσο όρο του δεκαδικού λογάριθμου του ελαχίστου εμβαδού που υπολογίστηκε, το μέσο όρο των επαναλήψεων (*iterations*) στο οποίο βρέθηκε, καθώς και τον αριθμό των εκτιμήσεων (*evaluations*), που στην περίπτωσή μας αντιστοιχούν στον αριθμό εκτελέσεων της μεθόδου των ιδιόμορφων σημείων. Τέλος, σε αρκετούς πίνακες εμφανίζεται και η τυπική απόκλιση (Τ.Α.) από το μέσο όρο του δεκαδικού λογάριθμου του ελαχίστου εμβαδού που υπολογίστηκε.

Για τη ρύθμιση του αλγορίθμου θα χρησιμοποιήσουμε την αεροτομή NACA 0012 (Εικόνα 23) με γωνία προσβολής στις  $0^\circ$  και αδιάστατη ταχύτητα στο άπειρο ίση με τη μονάδα. Για την επίλυση του ρευστοδυναμικού προβλήματος χρησιμοποιούμε τη μέθοδο των ιδιόμορφων σημείων, η οποία μας δίνει την κατανομή των πιέσεων γύρω από την πτέρυγα υποθέτοντας ιδεώδη ροή. Η επιλογή αυτή έγινε διότι η μέθοδος των ιδιόμορφων σημείων επιλύεται ταχύτητα στους σημερινούς ηλεκτρονικούς υπολογιστές. Εφαρμόζοντας τη μέθοδο των ιδιόμορφων σημείων στην παραπάνω αεροτομή λαμβάνουμε μια κατανομή πιέσεων η οποία είναι και ο στόχος μας.

Για να πραγματοποιήσουμε την Αντίστροφη Σχεδίαση Μεμονωμένης Αεροτομής και να επιτύχουμε τον παραπάνω στόχο, χρησιμοποιούμε τους εξής βαθμούς ελευθερίας για τα σημεία ελέγχου των πολυωνύμων Bezier:

$x$	$y_{min}$	$y_{max}$	$x$	$y_{min}$	$y_{max}$
<b>Bezier Curve 1</b>			<b>Bezier Curve 2</b>		
1.0	0.0	0.0	0.0	0.0	0.0
0.9	-0.2	0.0	0.0	0.0	0.2
0.7	-0.2	0.0	0.1	0.0	0.2
0.5	-0.2	0.0	0.3	0.0	0.2
0.3	-0.2	0.0	0.5	0.0	0.2
0.1	-0.2	0.0	0.7	0.0	0.2
0.0	-0.2	0.0	0.9	0.0	0.2
0.0	0.0	0.0	1.0	0.0	0.0

Πίνακας 2. Σημεία ελέγχου των πολυωνύμων Bezier

Στα επόμενα κεφάλαια γίνεται αναλυτική παρουσίαση της διαδικασίας ρύθμισης του αλγορίθμου EACO.

### 8.2.1 Αρχικοποίηση του αλγορίθμου

Μια από τις πρώτες παραμέτρους του EACO που αναφέρθηκαν είναι ο αριθμός των μυρμηγκιών που πραγματοποιούν τυχαία ταξίδια έτσι ώστε να αρχικοποιηθεί το μητρώο της ορατότητας, αλλά και αυτό της φερομόνης. Είναι φανερό (Πίνακας 3) ότι η απόδοση του αλγορίθμου πρακτικά παραμένει σταθερή για  $m_{init} \geq 5$ . Επομένως για να μειώσουμε το υπολογιστικό κόστος κατά την έναρξη του αλγορίθμου επιλέγουμε  $m_{init}=5$ .

Σημείωση: Αν όμως το ζητούμενο είναι η γρηγορότερη σύγκλιση του αλγορίθμου και κατά συνέπεια η γενική εξοικονόμηση υπολογιστικού χρόνου, τότε ίσως είναι καλύτερο να επιλέξουμε την τιμή  $m_{init}=30$  (η επιλογή αυτή δεν γίνεται με βεβαιότητα καθώς υπεισέρχονται φαινόμενα στοχαστικότητας στον αλγόριθμο). Είναι εμφανές (Πίνακας 3) ότι για αυτή την τιμή ο αλγόριθμος συγκλίνει πολύ γρηγορότερα από τις άλλες περιπτώσεις και μάλιστα στην καλύτερη τιμή που παρατηρήθηκε. Παρά ταύτα, όταν υιοθετείται μεγάλο πλήθος αρχικών μυρμηγκιών και άρα εναποτίθεται αρχικά τυχαία υψηλή ποσότητα φερομόνης υπάρχει ο κίνδυνος να εγκλωβίσουμε την έρευνα των επόμενων μυρμηγκιών σε τοπικά βέλτιστα.

Επίσης, σημαντική παράμετρος της αρχικοποίησης είναι η αρχική τιμή  $\tau_o$  της φερομόνης στις πόλεις. Από τα αποτελέσματα (Πίνακας 4 και Πίνακας 5) είναι εμφανές ότι η χρήση μικρής τιμής  $\tau_o$  είναι ευεργετική σε σχέση με τη μηδενική τιμή. Συνεπώς, επιλέγουμε  $\tau_o=10^{-10}$ .

Μ.Ο. Εμβαδού	Μ.Ο. Επαναλήψεων	Μ.Ο. Εκτιμήσεων	$m_{init}$
-1.33358	144.80	1732.6	1
-1.38315	137.90	1653.8	5
-1.37465	134.70	1620.4	10
-1.37451	129.75	1571.0	20
<b>-1.40444</b>	126.30	1539.6	30
-1.39190	133.10	1631.2	40

Πίνακας 3. Επίδραση  $m_{init}$  (EACO - 1 -b4,  $N=21$ ,  $\rho=0.5$ ,  $\alpha=0.8$ ,  $\beta=0.8$ ,  $r=0.1$ ,  $\tau_o=0.0$ ,  $c_f=0.001$ ,  $c_v=0.001$ )

Μ.Ο. Εμβαδού	Μ.Ο. Επαναλήψεων	Μ.Ο. Εκτιμήσεων	$\tau_o$
-1.70974	162.10	1944.2	0.0
<b>-1.72613</b>	154.15	1848.8	$10^{-10}$

Πίνακας 4. Επίδραση  $\tau_o$  (EACO -5 -b1 +ts1,  $N=31$ ,  $\rho=0.9$ ,  $\alpha=1$ ,  $\beta=11$ ,  $r=0.05$ ,  $c_f=0.001$ ,  $c_v=0.001$ ,  $\delta=0.75$ ,  $n_{ts}=10$ ,  $m_{init}=5$ )

Μ.Ο. Εμβαδού	Μ.Ο. Επαναλήψεων	Μ.Ο. Εκτιμήσεων	$\tau_o$
-1.72982	144.05	1727.6	0.0
<b>-1.75262</b>	150.50	1805.0	$10^{-10}$

Πίνακας 5. Επίδραση  $\tau_o$  (EACO -3 -b1 +ts2,  $N=31$ ,  $\rho=0.9$ ,  $\alpha=1$ ,  $\beta=11$ ,  $r=0.05$ ,  $c_f=0.001$ ,  $c_v=0.001$ ,  $\delta=0.75$ ,  $n_{ts}=15$ ,  $m_{init}=5$ ,  $n_{ar}=4$ ,  $f_{gl}=5$ )

### 8.2.2 Αριθμός κόμβων πλέγματος

Ο αριθμός των κόμβων  $N$  (*nodes*) κάθε πλευράς του πλέγματος που χρησιμοποιείται για την αποθήκευση των μεταβλητών του προβλήματος (φερομόνη, ορατότητα και πιθανότητα) παίζει καθοριστικό ρόλο στην απόδοση του αλγορίθμου. Μικρός αριθμός κόμβων μειώνει την ακρίβεια των πράξεων, ενώ μεγάλος αριθμός κόμβων αυξάνει κατά πολύ το υπολογιστικό κόστος του αλγορίθμου. Συνεπώς, πρέπει να βρεθεί μια ενδιάμεση τιμή που να δίνει ικανοποιητικά αποτελέσματα.

Από τα αποτελέσματα (Πίνακας 6) επιλέγουμε  $N=31$ , αν και για  $N>21$  η διαφοροποιήσεις είναι μικρές.

Μ.Ο. Εμβαδού	Μ.Ο. Επαναλήψεων	Μ.Ο. Εκτιμήσεων	<i>N</i>
-1.31110	119.45	1447.4	11
-1.41296	136.85	1656.2	21
<b>-1.42738</b>	129.20	1564.4	31
-1.41423	137.45	1663.4	41
-1.41730	127.45	1543.4	51

Πίνακας 6. Αριθμός κόμβων (EACO - 1 -b5,  $\rho=0.5$ ,  $\alpha=0.8$ ,  $\beta=0.8$ ,  $r=0.1$ ,  $\tau_o=0.0$ ,  $c_f=0.001$ ,  $c_v=0.001$ ,  $m_{ini}=20$ )

### 8.2.3 Προσδιορισμός των παραμέτρων $\alpha$ , $\beta$ , $c_f$ , $c_v$

Ο προσδιορισμός κατάλληλου συνδυασμού των  $\alpha$ ,  $\beta$  είναι αρκετά δύσκολος. Συγκρίνοντας τα αποτελέσματα (Πίνακας 7 και Πίνακας 8) παρατηρούμε ότι οι τιμές που λαμβάνουμε για  $\alpha, \beta < 1$  είναι πολύ χειρότερες από αυτές για  $\alpha, \beta \geq 1$ .

Θέτοντας  $\alpha=1$  και μεταβάλλοντας μόνο το  $\beta$ , βλέπουμε ότι ο αλγόριθμος εμφανίζει σταθερή συμπεριφορά σε ένα μεγάλο εύρος τιμών του  $\beta$  (από  $\beta=6$  μέχρι  $\beta=16$ ). Επειδή οι διαφορές των αποτελεσμάτων σε αυτό το διάστημα είναι μικρές σε σχέση με τα αποτελέσματα εκτός του διαστήματος αυτού, επιλέγουμε μια τιμή στο μέσο περίπου του διαστήματος. Έστω λοιπόν  $\beta=11$  (η τιμή αυτή μας έδωσε και τα καλύτερα αποτελέσματα).

Θέτοντας, τώρα,  $\beta=11$  και μεταβάλλοντας το  $\alpha$  δεν λαμβάνουμε σωστά αποτελέσματα τουλάχιστον κατά το 50% των εκτελέσεων του αλγορίθμου. Αυτό συμβαίνει διότι μετά από κάποιο αριθμό επαναλήψεων λόγω της μεγάλης τιμής του  $\alpha$  μηδενίζεται η πιθανότητα εκλογής μεγάλου μέρους των κόμβων της πόλεως δίνοντας εσφαλμένα αποτελέσματα. Προφανώς, η περίπτωση αυτή μπορεί να αποφευχθεί αν χρησιμοποιηθούν κάτω όρια στην ποσότητα της φερομόνης (αναφορικά με τη χρήση των ορίων της φερομόνης βλέπε παρακάτω).

Επιπλέον, για να μελετήσουμε την αναγκαιότητα ύπαρξης τόσο της φερομόνης όσο και της ορατότητας στον αλγόριθμο, θέτουμε  $\alpha=0 - \beta=1$  και  $\alpha=1 - \beta=0$  και συγκρίνουμε τα αποτελέσματα (Πίνακας 9). Είναι φανερό ότι για την καλύτερη απόδοση του αλγορίθμου είναι απαραίτητη τόσο η φερομόνη όσο και η ορατότητα. Σημαντικό είναι το γεγονός ότι με τη χρήση μόνο της φερομόνης λαμβάνουμε πολύ καλύτερα αποτελέσματα από ότι μόνο με την ορατότητα.

Οι παράμετροι  $c_f$  και  $c_v$  εξαρτώνται άμεσα από τις παραμέτρους  $\alpha$  και  $\beta$  και το αντίστροφο. Μετά από ένα σχετικά μεγάλο αριθμό δοκιμών που έγιναν (Πίνακας 10 και Πίνακας 11) και σε συνδυασμό με τα προηγούμενα δεδομένα (Πίνακας 8) καταλήξαμε στις τιμές  $c_f=c_v=0.001$ .

Λαμβάνοντας αφορμή από τα παραπάνω αποτελέσματα αντί της χρήσης του τυχαίου αναλογικού κανόνα μετάβασης του AS χρησιμοποιούμε αυτόν του ACS που θεωρεί  $\alpha=1$ . Δηλαδή, καταλήγουμε στην εξής σχέση:

$$P_{cij}(t) = \frac{\tau_{cij}(t) \cdot [\eta_{cij}(t)]^\beta}{\sum_{l,k} \tau_{clk}(t) \cdot [\eta_{clk}(t)]^\beta} \quad \text{Εξ. 37}$$

με  $\beta=11$  και  $c_f=c_v=0.001$ .

<b>Μ.Ο. Εμβαδού</b>	<b>Μ.Ο. Επαναλήψεων</b>	<b>Μ.Ο. Εκτιμήσεων</b>	<b><math>\alpha</math></b>	<b><math>\beta</math></b>
<b>-1.52484</b>	103.00	1250.0	1.0	1.0
-1.48331	133.60	1617.2	0.5	1.0
-1.37494	106.60	1293.2	0.1	1.0
-1.50200	121.60	1473.2	0.8	0.8
-1.50631	119.75	1451.0	1.0	0.5
-1.42653	130.65	1581.8	1.0	0.1

Πίνακας 7.  $0 < \alpha, \beta \leq 1$  (EACO - 5 -b1,  $N=31, \rho=0.9, r=0.1, \tau_o=0.0, c_f=0.001, c_v=0.001, m_{init}=20$ )

<b>Μ.Ο. Εμβαδού</b>	<b>Μ.Ο. Επαναλήψεων</b>	<b>Μ.Ο. Εκτιμήσεων</b>	<b><math>\alpha</math></b>	<b><math>\beta</math></b>
-1.52484	103.00	1250.0	1.0	1.0
-1.58037	136.80	1656.6	1.0	2.0
-1.63873	138.85	1680.2	1.0	4.0
-1.67140	152.50	1844.0	1.0	6.0
-1.67855	148.60	1797.2	1.0	10.0
<b>-1.70055</b>	141.85	1716.2	1.0	11.0
-1.68977	163.05	1970.6	1.0	12.0
-1.68529	152.85	1848.2	1.0	16.0
-1.65747	148.10	1791.2	1.0	19.0
-1.64172	163.85	1980.2	1.0	21.0

Πίνακας 8.  $\alpha, \beta \geq 1$  (EACO - 5 -b1,  $N=31, \rho=0.9, r=0.1, \tau_o=0.0, c_f=0.001, c_v=0.001, m_{init}=20$ )

<b>Μ.Ο. Εμβαδού</b>	<b>Μ.Ο. Επαναλήψεων</b>	<b>Μ.Ο. Εκτιμήσεων</b>	<b><math>\alpha</math></b>	<b><math>\beta</math></b>
<b>-1.52484</b>	103.00	1250.0	1.0	1.0
-1.13488	118.75	1439.0	0.0	1.0
-1.43553	130.80	1583.6	1.0	0.0

Πίνακας 9. Απενεργοποίηση φερομόνης και ορατότητας (EACO - 5 -b1,  $N=31, \rho=0.9, r=0.1, \tau_o=0.0, c_f=0.001, c_v=0.001, m_{init}=20$ )

Μ.Ο. Εμβαδού	Μ.Ο. Επαναλήψεων	Μ.Ο. Εκτιμήσεων	$c_f$	$c_v$
-1.55537	140.85	1704.2	0.000	0.000
-1.52484	103.00	1250.0	0.001	0.001
-1.52653	101.40	1230.8	0.000	0.001
-1.44950	133.80	1619.6	0.000	1.000
-1.55317	145.40	1758.8	0.001	0.000
-1.56008	114.40	1386.8	1.0	0.000
<b>-1.56749</b>	110.95	1345.4	100.0	0.000
-1.56222	116.25	1409.0	$10^4$	0.000
-1.54121	132.55	1604.6	$10^7$	0.000

Πίνακας 10. Επίδραση  $c_f$  και  $c_v$  με σταθερά τα  $\alpha$ ,  $\beta$  (EACO - 5 -b1,  $N=31$ ,  $\rho=0.9$ ,  $r=0.1$ ,  $\tau_o=0.0$ ,  $m_{init}=20$ ,  $\alpha=1$ ,  $\beta=1$ )

Μ.Ο. Εμβαδού	Μ.Ο. Επαναλήψεων	Μ.Ο. Εκτιμήσεων	$\alpha$	$\beta$
-1.62007	149.25	1790.0	1.0	0.5
-1.67151	153.50	1841.0	1.0	1.0
-1.66394	152.05	1823.6	1.0	1.3
<b>-1.69732</b>	159.00	1907.0	1.0	1.5
-1.66853	144.75	1736.0	1.0	1.6
-1.68693	148.40	1779.8	1.0	1.7
-1.65825	149.40	1791.8	1.0	2.0
-1.62191	151.40	1815.8	1.0	5.0

Πίνακας 11. Επίδραση των  $\alpha$ ,  $\beta$  συναρτήσει των νέων τιμών των  $c_f$  και  $c_v$  που προσδιορίστηκαν (Πίνακας 10) (EACO - 5 -b1,  $N=31$ ,  $\rho=0.9$ ,  $r=0.05$ ,  $\tau_o=0.0$ ,  $c_f=100$ ,  $c_v=0.0$ ,  $m_{init}=5$ )

#### 8.2.4 Προσδιορισμός της παραμέτρου $r$

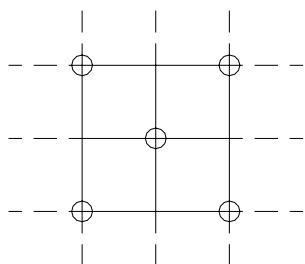
Η απόδοση του αλγορίθμου εξαρτάται σε μεγάλο βαθμό από την τιμή της παραμέτρου αυτής. Για  $r < 0.05$  η τυπική απόκλιση της εκθετικής κατανομής είναι πολύ μικρή με αποτέλεσμα να μην αποθηκεύονται οι μεταβλητές (φερομόνη και ορατότητα) στους κόμβους του πλέγματος. Η απόδοση του αλγορίθμου είναι σαφέστατα καλύτερη για  $r=0.05$  (Πίνακας 12).

Μ.Ο. Εμβαδού	Μ.Ο. Επαναλήψεων	Μ.Ο. Εκτιμήσεων	$r$
-0.74879	118.00	1415.0	1.00
-0.86441	135.45	1624.4	0.50
-1.38315	137.90	1653.8	0.10
<b>-1.49177</b>	143.15	1716.8	0.05

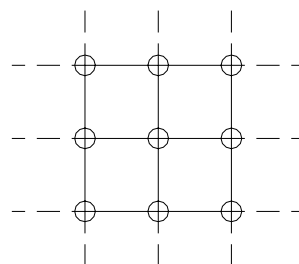
Πίνακας 12. Επίδραση  $r$  (EACO - 1 -b4,  $N=21$ ,  $\rho=0.5$ ,  $\alpha=0.8$ ,  $\beta=0.8$ ,  $\tau_o=0.0$ ,  $c_f=0.001$ ,  $c_v=0.001$ ,  $m_{init}=5$ )

### 8.2.5 Διαδικασία επιλογής θέσης της φωλιάς

Όπως έχει ήδη αναφερθεί, θέλοντας να εκμεταλλευτούμε τη συσσωρευμένη εμπειρία της αποικίας επιθυμούμε κάθε μυρμήγκι να ξεκινάει το ταξίδι του από το σημείο της πόλης με τη μέγιστη πιθανότητα να επιλεγεί. Για το λόγο αυτό επιλέγουμε τον κόμβο σε κάθε πόλη με τη μέγιστη πιθανότητα εκλογής και κάνουμε παρεμβολή μιας επιφάνειας χρησιμοποιώντας τους γειτονικούς κόμβους. Κατόπιν, υπολογίζουμε το σημείο όπου εμφανίζεται το μέγιστο της επιφάνειας και τοποθετούμε τη φωλιά στο σημείο αυτό.



Εικόνα 41. Οι 5 κόμβοι που χρησιμοποιούνται για την παρεμβολή της επιφάνειας για τον προσδιορισμό του σημείου με τη μέγιστη πιθανότητα.



Εικόνα 42. Οι 9 κόμβοι που χρησιμοποιούνται για την παρεμβολή της επιφάνειας για τον προσδιορισμό του σημείου με τη μέγιστη πιθανότητα.

Οι επιφάνειες που μπορούμε να υπολογίσουμε είναι οι εξής:

1. Επιφάνεια που δημιουργείται με παρεμβολή 5 κόμβων (Εικόνα 41):

$$S(x,y) = a_1 + a_2x + a_3y + a_4xy + a_5(x^2 + y^2) \quad \text{Εξ. 38}$$

και της οποίας το μέγιστο είτε υπολογίζεται αναλυτικά (EACO –b1) από τις σχέσεις:

$$x_{nest} = \frac{2a_2a_5 - a_3a_4}{a_2^2 - 4a_5^2} \quad \text{Εξ. 39}$$

$$y_{nest} = \frac{2a_3a_5 - a_2a_4}{a_2^2 - 4a_5^2} \quad \text{Εξ. 40}$$

είτε προσεγγίζεται υπολογιστικά (EACO –b2) με κάποιον αλγόριθμο. Στην περίπτωση μας χρησιμοποιούμε τον Multiple Restart Random Hill Climbing – MRRHC που ανήκει στην κατηγορία των στοχαστικών αλγορίθμων και είναι απλός στην υλοποίηση. Ο MRRHC χρησιμοποιείται με μέγιστο αριθμό επαναλήψεων 1000, επαναρχικοποίηση ανά 100 επαναλήψεις χωρίς βελτίωση και ποσοστό περιοχής της τοπικής έρευνας 20% του συνολικού χώρου των λύσεων.

2. Επιφάνεια που δημιουργείται με παρεμβολή 9 κόμβων (Εικόνα 42):

$$S(x,y) = a_1 + a_2x + a_3y + a_4xy + a_5x^2 + a_6y^2 + a_7x^2y + a_8xy^2 + a_9x^2y^2 \quad \text{Εξ. 41}$$

και της οποίας το μέγιστο μπορεί να προσεγγιστεί μόνο υπολογιστικά (χρησιμοποιούμε και πάλι τον MRRHC) καθώς η αναλυτική λύση είναι πολύ δύσκολη αν όχι αδύνατη.

3. Επιφάνεια που δημιουργείται από 9 (EACO -b5) ή περισσότερα σημεία εφαρμόζοντας το νόμο των ελαχίστων τετραγώνων σε επιφάνεια που προκύπτει με παρεμβολή 5 κόμβων:

$$E = \sum_{i=1}^{\mu} (z_i - S(x_i, y_i))^2 \quad \text{Εξ. 42}$$

όπου  $(x_i, y_i, z_i)$  με  $i=1, 2, \dots, \mu$  τα σημεία στα θέλουμε να παρεμβάλουμε την καμπύλη  $S(x,y)$  (Εξ. 38). Στόχος μας είναι να ελαχιστοποιήσουμε το  $E$ . Εφαρμόζοντας τη γνωστή μεθοδολογία των ελαχίστων τετραγώνων καταλήγουμε στο  $5 \times 5$  σύστημα εξισώσεων το οποίο και μας δίνει τους συντελεστές της  $S(x,y)$ :

$$\mu a_1 + \left( \sum_{i=1}^{\mu} x_i \right) a_2 + \left( \sum_{i=1}^{\mu} y_i \right) a_3 + \left( \sum_{i=1}^{\mu} x_i y_i \right) a_4 + \left( \sum_{i=1}^{\mu} x_i^2 + \sum_{i=1}^{\mu} y_i^2 \right) a_5 = \sum_{i=1}^{\mu} z_i \quad \text{Εξ. 43}$$

$$\left( \sum_{i=1}^{\mu} x_i \right) a_1 + \left( \sum_{i=1}^{\mu} x_i^2 \right) a_2 + \left( \sum_{i=1}^{\mu} x_i y_i \right) a_3 + \left( \sum_{i=1}^{\mu} x_i^2 y_i \right) a_4 + \left( \sum_{i=1}^{\mu} x_i^3 + \sum_{i=1}^{\mu} x_i y_i^2 \right) a_5 = \sum_{i=1}^{\mu} x_i z_i \quad \text{Εξ. 44}$$

$$\left( \sum_{i=1}^{\mu} y_i \right) a_1 + \left( \sum_{i=1}^{\mu} x_i y_i \right) a_2 + \left( \sum_{i=1}^{\mu} y_i^2 \right) a_3 + \left( \sum_{i=1}^{\mu} x_i y_i^2 \right) a_4 + \left( \sum_{i=1}^{\mu} x_i^2 y_i + \sum_{i=1}^{\mu} y_i^3 \right) a_5 = \sum_{i=1}^{\mu} y_i z_i \quad \text{Εξ. 45}$$

$$\begin{aligned} & \left( \sum_{i=1}^{\mu} x_i y_i \right) a_1 + \left( \sum_{i=1}^{\mu} x_i^2 y_i \right) a_2 + \left( \sum_{i=1}^{\mu} x_i y_i^2 \right) a_3 + \left( \sum_{i=1}^{\mu} x_i^2 y_i^2 \right) a_4 + \\ & + \left( \sum_{i=1}^{\mu} x_i^3 y_i + \sum_{i=1}^{\mu} x_i y_i^3 \right) a_5 = \sum_{i=1}^{\mu} x_i y_i z_i \end{aligned} \quad \text{Εξ. 46}$$

$$\begin{aligned} & \left( \sum_{i=1}^{\mu} x_i^2 + \sum_{i=1}^{\mu} y_i^2 \right) a_1 + \left( \sum_{i=1}^{\mu} x_i^3 + \sum_{i=1}^{\mu} x_i y_i^2 \right) a_2 + \left( \sum_{i=1}^{\mu} x_i^2 y_i + \sum_{i=1}^{\mu} y_i^3 \right) a_3 + \\ & + \left( \sum_{i=1}^{\mu} x_i^3 y_i + \sum_{i=1}^{\mu} x_i y_i^3 \right) a_4 + \left( \sum_{i=1}^{\mu} x_i^4 + 2 \sum_{i=1}^{\mu} x_i^2 y_i^2 + \sum_{i=1}^{\mu} y_i^4 \right) a_5 = \sum_{i=1}^{\mu} (x_i^2 + y_i^2) z_i \end{aligned} \quad \text{Εξ. 47}$$

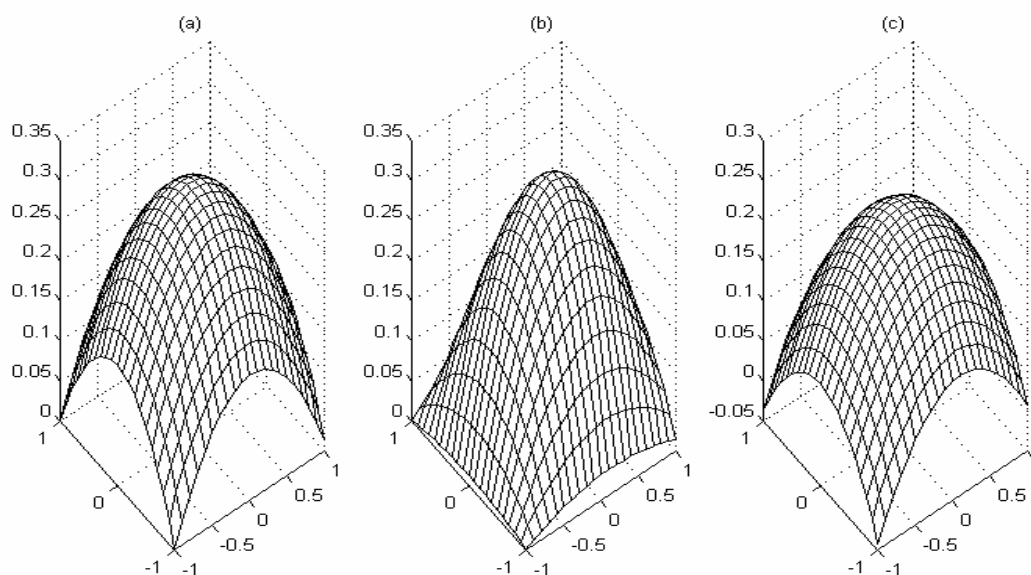
Το μέγιστο αυτής της καμπύλης, δηλαδή η θέση της φωλιάς, υπολογίζεται όπως και στην περίπτωση 1.

Τέλος, στις πρώτες δοκιμές του αλγορίθμου που έγιναν, χρησιμοποιήσαμε και τυχαία σημεία στη γειτονιά του καλύτερου κόμβου ως φωλιές των μυρμηγκιών. Αυτή η παραλλαγή του αλγορίθμου συμβολίζεται με (EACO –b4).

Από τα αποτελέσματα (Πίνακας 13) φαίνεται ότι ο ακριβής προσδιορισμός της θέσης της φωλιάς (b1, b2, b3, b5) είναι σαφώς καλύτερος από την τυχαία επιλογή μιας θέσης (b4). Όπως ήταν αναμενόμενο, το καλύτερο αποτέλεσμα το λαμβάνουμε από την επιφάνεια που παρεμβάλλεται σε 9 κόμβους (b3). Παρά ταύτα, αν θέλουμε να εξοικονομήσουμε υπολογιστικό χρόνο θα επιλέξουμε τη μέθοδο b1 η οποία εν τέλει δεν διαφοροποιείται και πολύ από την b3 όσον αφορά το τελικό αποτέλεσμα.

Αλγόριθμος	M.O. Εμβαδού	M.O. Επαναλήψεων	M.O. Εκτιμήσεων
EACO - 1 -b1	-1.41702	124.10	1503.2
EACO - 1 -b2	-1.41763	119.65	1449.8
EACO - 1 -b3	<b>-1.41843</b>	111.55	1352.6
EACO - 1 -b4	-1.37451	129.75	1571.0
EACO - 1 -b5	-1.41296	136.85	1656.2

Πίνακας 13. Επιλογής θέσης φωλιάς ( $\rho=0.5$ ,  $N=21$ ,  $\alpha=0.8$ ,  $\beta=0.8$ ,  $r=0.1$ ,  $\tau_o=0.0$ ,  $c_f=0.001$ ,  $c_v=0.001$ ,  $m_{ini}=20$ )



Εικόνα 43. Οι τρεις επιφάνειες που προκύπτουν με παρεμβολή γύρω από τον κόμβο με τη μέγιστη πιθανότητα εκλογής: (a) επιφάνεια που προκύπτει με παρεμβολή 5 κόμβων, (b) επιφάνεια που προκύπτει με παρεμβολή 9 κόμβων και (c) επιφάνεια που δημιουργείται από 9 σημεία εφαρμόζοντας το νόμο των ελαχίστων τετραγώνων σε επιφάνεια που προκύπτει με παρεμβολή 5 κόμβων.

### 8.2.6 Εναπόθεση φερομόνης

Εξετάσθηκαν οι εξής περιπτώσεις εναπόθεσης φερομόνης:

- Από όλα τα μυρμήγκια (EACO –1).
- Από το μυρμήγκι που βρίσκει την καλύτερη λύση σε κάθε επανάληψη και από το μυρμήγκι που έχει εντοπίσει τη βέλτιστη λύση (EACO –2).
- Με σταδιακή μετάβαση από το μυρμήγκι που βρίσκει την καλύτερη λύση σε κάθε επανάληψη στο μυρμήγκι που έχει βρει τη βέλτιστη λύση (EACO –3).

Στη συγκεκριμένη περίπτωση στον αλγόριθμο υπεισέρχονται και άλλες δύο παράμετροι:

(α) η αρχική συχνότητα ( $f_{gl}$ ) της χρήσης του μυρμηγκιού που έχει εντοπίσει τη βέλτιστη λύση. Για παράδειγμα, αν  $f_{gl}=3$  τότε για τρεις επαναλήψεις χρησιμοποιείται το μυρμήγκι που εντοπίζει την καλύτερη λύση κάθε επανάληψης και κατά την 4<sup>η</sup> επανάληψη το μυρμήγκι που έχει εντοπίσει τη βέλτιστη λύση μέχρι την τρέχουσα χρονική στιγμή. Κατά την εκτέλεση του αλγορίθμου η συχνότητα αυτή μεταβάλλεται γραμμικά από την αρχική της τιμή στη μηδενική τιμή στις τελευταίες επαναλήψεις του αλγορίθμου. Δηλαδή, προς το τέλος της εκτέλεσης του αλγορίθμου χρησιμοποιείται μόνο το μυρμήγκι που έχει βρει τη βέλτιστη λύση.

(β) ο αριθμός ( $n_{ar}$ ) των περιοχών στο χρόνο στις οποίες θα χρησιμοποιείται διαφορετική συχνότητα. Για παράδειγμα, αν  $f_{gl}=3$ ,  $n_{ar}=4$  και *maximum iterations*=200, τότε θα πραγματοποιηθούν τρεις αλλαγές στην τιμή της συχνότητας κατά τις επαναλήψεις 50, 100 και 150. Οι συχνότητες στις χρονικές περιοχές (επαναλήψεις) 1-50, 51-100, 101-150 και 151-200 θα είναι αντίστοιχα 3, 2, 1 και 0.

- Μόνο από το μυρμήγκι που βρίσκει την καλύτερη λύση σε κάθε επανάληψη (EACO –4).
- Μόνο από το μυρμήγκι που έχει εντοπίσει την ολικά βέλτιστη λύση μέχρι την τρέχουσα χρονική στιγμή (EACO –5).

Είναι προφανές (Πίνακας 14) ότι τα καλύτερα αποτελέσματα τα δίνει η ανανέωση φερομόνης από το μυρμήγκι που έχει βρει την ολικά βέλτιστη λύση (EACO –5). Λόγω του μικρού αριθμού των επαναλήψεων που χρησιμοποιούμε (200) είμαστε αναγκασμένοι να

αξιοποιήσουμε μια μέθοδο εναπόθεσης φερομόνης (-5) που να κατευθύνει την έρευνα προς την περιοχή που έχει δώσει τα καλύτερα αποτελέσματα. Αντιθέτως, αν είχαμε την ευχέρεια να χρησιμοποιήσουμε ένα μεγαλύτερο αριθμό επαναλήψεων θα προτιμούσαμε μία από τις μεθόδους (-2), (-3) και (-4) οι οποίες πραγματοποιούν καλύτερη εξερεύνηση (*exploration*) και εκμετάλλευση (*exploitation*) του χώρου των λύσεων.

Στη βιβλιογραφία παρατηρούμε τα εξής:

- Ο αλγόριθμος ACS [21] αξιοποιεί το μυρμήγκι που έχει εντοπίσει την ολικά βέλτιστη λύση για την εναπόθεση της φερομόνης (-5).
- Ο αλγόριθμος MMAS [49] αξιοποιεί το μυρμήγκι που βρίσκει την καλύτερη λύση σε κάθε επανάληψη (-4) αλλά επειδή η μέθοδος αυτή συγκλίνει πολύ αργά τελικά χρησιμοποιείται η σταδιακή μετάβαση από το μυρμήγκι που βρίσκει την καλύτερη λύση σε κάθε επανάληψη στο μυρμήγκι που έχει βρει την ολικά βέλτιστη λύση (-3). Η μέθοδος αυτή (-3) βοηθάει στην καλή αρχική εξερεύνηση (*exploration*) του χώρου των λύσεων και κατόπιν στη εκμετάλλευση (*exploitation*) της περιοχής που έχει δώσει τα καλύτερα αποτελέσματα.
- Ο Borczka στο [7] δείχνει πειραματικά ότι η εναπόθεση φερομόνης από το μυρμήγκι που έχει εντοπίσει τη βέλτιστη λύση (-5) (*global best ant* ή *global leader*) επιτυγχάνει πολύ καλύτερα αποτελέσματα από τις υπόλοιπες μεθόδους (δεν λαμβάνει υπόψη του, όμως, τις περιπτώσεις -2 και -3).

Αλγόριθμος	M.O. Εμβαδού	M.O. Επαναλήψεων	M.O. Εκτιμήσεων	$n_{ar}$	$f_{gl}$
EACO -1 -b1	-1.55973	155.95	1870.4		
EACO -2 -b1	-1.72099	144.60	1736.2		
EACO -3 -b1	-1.71617	148.70	1783.4	4	5
EACO -4 -b1	-1.74590	159.90	1917.8		
EACO -5 -b1	<b>-1.74901</b>	148.20	1777.4		

Πίνακας 14. Εναπόθεση φερομόνης ( $\rho=0.9$ ,  $N=31$ ,  $\alpha=1$ ,  $\beta=11$ ,  $r=0.05$ ,  $\tau_o=10^{-10}$ ,  $c_f=0.001$ ,  $c_v=0.001$ ,  $m_{ini}=5$ )

### 8.2.7 Προσδιορισμός των παραμέτρων $n_{ar}$ και $f_{gl}$

Οι παράμετροι αυτοί παίζουν σημαντικό ρόλο στην απόδοση του EACO -3. Από τα αποτελέσματα (Πίνακας 15 και Πίνακας 16) βλέπουμε ότι οι βέλτιστες τιμές είναι οι:  $n_{ar}=4$  και  $f_{gl}=5$ .

Μ.Ο. Εμβαδού	Μ.Ο. Επαναλήψεων	Μ.Ο. Εκτιμήσεων	$n_{ar}$
-1.69559	155.95	1870.4	2
-1.66486	169.40	2031.8	3
<b>-1.73597</b>	159.65	1914.8	4
-1.67992	167.90	2013.8	5
-1.63005	156.40	1875.8	6

Πίνακας 15. Επίδραση  $n_{ar}$  (EACO -3 -b1,  $N=31$ ,  $\rho=0.9$ ,  $\alpha=1$ ,  $\beta=11$ ,  $r=0.05$ ,  $c_f=0.001$ ,  $c_v=0.001$ ,  $f_{gl}=5$ ,  $m_{init}=5$ )

Μ.Ο. Εμβαδού	Μ.Ο. Επαναλήψεων	Μ.Ο. Εκτιμήσεων	$f_{gl}$
-1.70687	163.00	1955.0	1
-1.68257	158.25	1898.0	2
-1.70433	153.75	1844.0	3
-1.70342	152.75	1832.0	4
<b>-1.73597</b>	159.65	1914.8	5
-1.67145	160.90	1929.8	6
-1.69304	160.65	1926.8	7
-1.69793	158.60	1902.2	8

Πίνακας 16. Επίδραση  $f_{gl}$  (EACO -3 -b1,  $N=31$ ,  $\rho=0.9$ ,  $\alpha=1$ ,  $\beta=11$ ,  $r=0.05$ ,  $c_f=0.001$ ,  $c_v=0.001$ ,  $n_{ar}=4$ ,  $m_{init}=5$ )

### 8.2.8 Εξάτμιση φερομόνης

Όπως αναλύθηκε εκτενώς στο Κεφάλαιο 2 η αρνητική ανάδραση, δηλαδή η εξάτμιση της φερομόνης, είναι βασικός παράγοντας στα συστήματα αυτο-οργάνωσης και κατά προέκταση και στους αλγορίθμους που εμπνέονται από αυτά. Είναι φανερό (Πίνακας 17) ότι η βέλτιστη τιμή για την εξάτμιση της φερομόνης είναι  $\rho=0.9$ .

Μ.Ο. Εμβαδού	Μ.Ο. Επαναλήψεων	Μ.Ο. Εκτιμήσεων	$\rho$
-1.33646	121.80	1475.6	0.10
-1.37451	129.75	1571.0	0.50
-1.37849	123.80	1499.6	0.85
<b>-1.39270</b>	100.80	1223.6	0.90
-1.38900	104.20	1264.4	0.95

Πίνακας 17. Εξάτμιση φερομόνης (EACO - 1 -b4,  $N=21$ ,  $\alpha=0.8$ ,  $\beta=0.8$ ,  $r=0.1$ ,  $\tau_o=0.0$ ,  $c_f=0.001$ ,  $c_v=0.001$ ,  $m_{init}=20$ )

### 8.2.9 Ομαλοποίηση φερομόνης

Χρησιμοποιήθηκε ο εξής μηχανισμός ομαλοποίησης της φερομόνης (ts): ο μηχανισμός ενεργοποιείται μετά από  $n_{ts}$  επαναλήψεις χωρίς την εύρεση νέας βέλτιστης λύσης. Κατόπιν κάθε νέα ενεργοποίηση του μηχανισμού πραγματοποιείται μετά από αριθμό επαναλήψεων ίσο με τον αριθμό των επαναλήψεων που χρειάστηκαν από την έναρξη του αλγορίθμου μέχρι την πρώτη ενεργοποίηση του μηχανισμού ομαλοποίησης.

Στους σχετικούς πίνακες – Πίνακας 18 και Πίνακας 19 – παρουσιάζονται τα αποτελέσματα από τη διερεύνηση της επίδρασης της ομαλοποίησης της φερομόνης. Είναι εμφανές ότι ο μηχανισμός αυτός επιφέρει μια μικρή βελτίωση στην γενικότερη απόδοση του αλγορίθμου. Όσον αφορά το  $n_{ts}$  μπορούμε να κατασταλάξουμε στην τιμή 10. Η επιλογή του  $\delta$ , όμως, δεν είναι τόσο εύκολη. Παρατηρούμε ότι σε ένα εύρος τιμών 0.50-0.90 ο αλγόριθμος παρουσιάζει περίπου την ίδια απόδοση. Συνεπώς, επιλέγουμε την τιμή  $\delta=0.75$  στο μέσο του διαστήματος που τυχαίνει να έχει δώσει και τη βέλτιστη λύση που υπολογίστηκε.

Μια καλύτερη λύση για το μηχανισμό ομαλοποίησης της φερομόνης μπορεί να είναι η εξής: ο μηχανισμός ενεργοποιείται μετά από  $n_{ts}$  επαναλήψεις χωρίς την εύρεση νέας βέλτιστης λύσης. Κατόπιν, συγκρατούμε την καλύτερη λύση που υπολογίζεται μετά την ομαλοποίηση της φερομόνης και επαναενεργοποιούμε το μηχανισμό όταν αυτή δεν βελτιώνεται περαιτέρω για αριθμό επαναλήψεων ίσο με  $n_{ts}$ .

Δεν έγινε αναλυτική παρουσίαση αποτελεσμάτων με τη χρήση του παραπάνω μηχανισμού καθώς ο μικρός συνολικός αριθμός επαναλήψεων επιτρέπει την ενεργοποίηση του μηχανισμού της ομαλοποίησης της φερομόνης μόνο μια ή δύο φορές κατά την εκτέλεση του αλγορίθμου με αποτέλεσμα να μην μπορούμε να συγκρίνουμε τους δύο μηχανισμούς.

Αλγόριθμος	M.O. Εμβαδού	M.O. Επαναλήψεων	M.O. Εκτιμήσεων	$n_{ts}$
EACO - 5 -b4	-1.47443	127.60	1545.2	
EACO - 5 -b4 +ts	-1.46223	126.25	1529.0	5
EACO - 5 -b4 +ts	<b>-1.47631</b>	116.20	1408.4	10
EACO - 5 -b4 +ts	-1.46694	116.40	1410.8	15

Πίνακας 18. Ομαλοποίηση φερομόνης – επίδραση  $n_{ts}$  ( $N=21$ ,  $\rho=0.5$ ,  $\alpha=0.8$ ,  $\beta=0.8$ ,  $r=0.1$ ,  $\tau_o=0.0$ ,  $c_f=0.001$ ,  $c_v=0.001$ ,  $m_{init}=20$ ,  $\delta=0.5$ )

Μ.Ο. Εμβαδού	Μ.Ο. Επαναλήψεων	Μ.Ο. Εκτιμήσεων	$\delta$
-1.46327	117.75	1427.0	0.40
-1.47631	116.20	1408.4	0.50
-1.47567	113.35	1374.2	0.60
-1.47857	115.55	1400.6	0.70
<b>-1.48497</b>	118.15	1431.8	0.75
-1.48216	105.25	1277.0	0.85
-1.48004	110.10	1335.2	0.90
-1.47592	118.15	1431.8	0.95

Πίνακας 19. Ομαλοποίηση φερομόνης – επίδραση  $\delta$  (EACO - 5 -b4 +ts,  $N=21$ ,  $\rho=0.5$ ,  $\alpha=0.8$ ,  $\beta=0.8$ ,  $r=0.1$ ,  $\tau_o=0.0$ ,  $c_f=0.001$ ,  $c_v=0.001$ ,  $m_{ini}=20$ ,  $n_s=10$ )

### 8.2.10 Επαναρχικοποίηση του αλγορίθμου

Η επαναρχικοποίηση (tri) πραγματοποιείται θέτοντας τα επίπεδα της φερομόνης στην αρχική τους τιμή  $\tau_o$  και κατόπιν εκτελώντας τυχαία ταξίδια με σκοπό να επαναπροσδιορισθούν τα επίπεδα της φερομόνης. Η επαναενεργοποίηση της επαναρχικοποίησης του αλγορίθμου γίνεται με τον ίδιο τρόπο όπως και στην περίπτωση του μηχανισμού ομαλοποίησης της φερομόνης. Για την καλύτερη ανάλυση του μηχανισμού εξετάστηκε μια πληθώρα περιπτώσεων (Πίνακας 20). Από τα αποτελέσματα παρατηρούμε ότι η επαναρχικοποίηση του αλγορίθμου έχει θετική επίδραση στην απόδοσή του αλγορίθμου.

Αλγόριθμος	Μ.Ο. Εμβαδού	Μ.Ο. Επαναλήψεων	Μ.Ο. Εκτιμήσεων	$n_{tri}$
EACO -1 -b1	-1.60331	150.00	1799.0	
EACO -1 -b1 +tri	<b>-1.63175</b>	149.05	1787.6	30
EACO -2 -b1	-1.70843	165.70	1987.4	
EACO -2 -b1 +tri	<b>-1.72337</b>	158.85	1905.2	15
EACO -3 -b1	-1.73597	159.65	1914.8	
EACO -3 -b1 +tri	<b>-1.74124</b>	167.15	2004.8	25
EACO -4 -b1	-1.64890	161.45	1936.4	
EACO -4 -b1 +tri	<b>-1.73976</b>	157.55	1889.6	20
EACO -5 -b1	-1.68643	163.65	1962.8	
EACO -5 -b1 +tri	<b>-1.70157</b>	163.65	1962.8	25

Πίνακας 20. Επαναρχικοποίηση – επίδραση  $n_{tri}$  – παρουσιάζονται τα καλύτερα αποτελέσματα για κάθε εκδοχή του αλγορίθμου ( $N=31$ ,  $\alpha=1$ ,  $\beta=11$ ,  $r=0.05$ ,  $\rho=0.9$ ,  $\tau_o=0.0$ ,  $c_f=0.001$ ,  $c_v=0.001$ ,  $m_{ini}=5$ )

### 8.2.11 Συνδυασμένη δράση ομαλοποίησης της φερομόνης και επαναρχικοποίησης

Από τα αποτελέσματα (Πίνακας 21) παρατηρούμε ότι η συνδυασμένη δράση ομαλοποίησης της φερομόνης και επαναρχικοποίησης του αλγορίθμου είναι θετική προς την απόδοση του

αλγόριθμου πλην του (EACO -4 -b1). Στον (EACO -4 -b1), αντιθέτως, μειώνεται η απόδοση τόσο στον μέσο όρο του αποτελέσματος (εμβαδού) όσο και στην τυπική απόκλιση αυτού.

Το σημαντικό είναι ότι τώρα πλέον οποιαδήποτε εκδοχή του αλγόριθμου εμφανίζει περίπου την ίδια απόδοση και μάλιστα με τις εξής τιμές παραμέτρων:  $\delta=0.75$ ,  $n_{ts}=10$  και  $n_{tri}=30$ .

Μια ακόμη σημαντική παρατήρηση που προκύπτει είναι ότι ο αλγόριθμος (EACO -5 -b1) εμφανίζει τη μικρότερη τυπική απόκλιση με ένα αρκετά ικανοποιητικό μέσο όρο τελικού αποτελέσματος και επομένως είναι αρκετά αξιόπιστος ως προς την απόδοσή του. Αντιθέτως, ο αλγόριθμος (EACO -5 -b1 +ts +tri) εμφανίζει τη μεγαλύτερη τυπική απόκλιση αλλά με το δεύτερο καλύτερο μέσο όρο.

Σύμφωνα με τα τελευταία δεδομένα (Πίνακας 21), η καλύτερη λύση φαίνεται να είναι ο αλγόριθμος (EACO -2 -b1 +ts +tri) που εμφανίζει το μικρότερο μέσο όρο εμβαδού σε συνδυασμό με μια αρκετά καλή τυπική απόκλιση.

Αλγόριθμος	Μ.Ο. Εμβαδού	Μ.Ο. Επαναλήψεων	Μ.Ο. Εκτιμήσεων	T.A.
EACO -2 -b1	-1.72099	144.60	1734.2	0.110
EACO -2 -b1 +ts +tri	<b>-1.75546</b>	164.20	1969.4	0.095
EACO -3 -b1	-1.71617	148.70	1783.4	0.109
EACO -3 -b1 +ts +tri	<b>-1.73490</b>	151.90	1821.8	0.077
EACO -4 -b1	<b>-1.74590</b>	159.90	1917.8	0.105
EACO -4 -b1 +ts +tri	-1.72492	165.80	1988.6	0.120
EACO -5 -b1	-1.74901	148.20	1777.4	0.070
EACO -5 -b1 +ts +tri	<b>-1.75422</b>	155.20	1861.4	0.133

Πίνακας 21. Συνδυασμένη δράση ομαλοποίησης της φερομόνης και επαναρχικοποίησης – παρουσιάζονται τα καλύτερα αποτελέσματα για κάθε εκδοχή του αλγόριθμου ( $N=31$ ,  $\alpha=1$ ,  $\beta=11$ ,  $r=0.05$ ,  $\rho=0.9$ ,  $\tau_o=10^{-10}$ ,  $c_f=0.001$ ,  $c_v=0.001$ ,  $m_{init}=5$ ,  $\delta=0.75$ ,  $n_{ts}=10$ ,  $n_{tri}=30$ ,  $n_{ar}=4$ ,  $f_{gt}=5$ ).

### 8.2.12 Χρησιμοποίηση ορίων στη φερομόνη

Όπως έχει ήδη αναφερθεί, το άνω όριο της ποσότητας της φερομόνης υπολογίζεται από την Εξ. 36. Για τον προσδιορισμό του κάτω ορίου χρησιμοποιούμε 2 μεθόδους:

- MMEACO ( $\tau_{min}$ ): προσδιορισμό από το χρήστη της τιμής του κάτω ορίου  $\tau_{min}$ .
- MMEACO ( $\% \tau_{max}$ ): προσδιορισμό από το χρήστη μιας παραμέτρου  $c$  τέτοιας ώστε  $\tau_{min}=c\tau_{max}$ .

Όταν χρησιμοποιήθηκε μόνο το άνω όριο ( $\tau_{max}$ ) και έγινε αρχικοποίηση με  $\tau_o=0$  τότε ο MMEACO παρουσίασε τα ίδια αποτελέσματα με τον EACO. Αντιθέτως, όταν χρησιμοποιήθηκε

κάτω όριο στην ποσότητα της φερομόνης και έγινε αρχικοποίηση είτε με  $\tau_o=0$  είτε με  $\tau_o=10^{-10}$  τότε ο MMEACO εμφάνισε χειρότερη συμπεριφορά από τον EACO. Όταν, όμως, έγινε αρχικοποίηση σε μια μεγάλη τιμή φερομόνης ( $\tau_o=10^{10}$ ) τότε ο MMEACO εμφάνισε σαφέστατα καλύτερη απόδοση από τον EACO (Πίνακας 22). Σημαντικό είναι το γεγονός ότι εκτός από την βελτίωση στην απόδοση του αλγορίθμου παρατηρούμε και μείωση της τυπικής απόκλισης του μέσου όρου του ελάχιστου εμβαδού. Συνεπώς, μπορούμε να είμαστε πιο σίγουροι για την ακρίβεια των αποτελεσμάτων μιας εκτέλεσης του αλγορίθμου.

Λαμβάνοντας υπόψη τη σημαντική βελτίωση της ποιότητας της λύσης (μείωση μέσου όρου και απόκλισης) που παρουσίασε ο MMEACO μελετήσαμε και πάλι την επίδραση των μηχανισμών της επαναρχικοποίησης και της ομαλοποίησης της φερομόνης (Πίνακας 23). Τα αποτελέσματα και πάλι είναι θετικά. Η σημαντική διαφοροποίηση που παρατηρείται, όμως, είναι ότι στο περιορισμένο συνολικό αριθμό επαναλήψεων δεν είναι δυνατή η συνδυασμένη δράση των δύο μηχανισμών.

Τα καλύτερα αποτελέσματα (Μ.Ο. εμβαδού και τυπική απόκλιση) τα έδωσαν οι αλγόριθμοι MMEACO -3 ( $\% \tau_{\max}$ ) +ts και MMEACO -4 ( $\% \tau_{\max}$ ) +tri. Θα προτιμήσουμε τον δεύτερο καθώς παρουσιάζει τη μικρότερη τυπική απόκλιση και άρα και μειωμένο κίνδυνο εμφάνισης μιας «κακής» λύσης.

Τέλος, στην νέα αυτή εκδοχή του αλγορίθμου επανεξετάστηκαν οι επιδράσεις των παραμέτρων  $\alpha$ ,  $\beta$  και  $\rho$ . Τα αποτελέσματα έδειξαν ότι οι βέλτιστες τιμές παραμένουν οι ίδιες, δηλαδή  $\alpha=1$ ,  $\beta=11$  και  $\rho=0.9$ .

Αλγόριθμος	Μ.Ο. Εμβαδού	Μ.Ο. Επανάληψεων	Μ.Ο. Εκτιμήσεων	T.A.	$t_{min}$	$c$	$\tau_o$
EACO -2	-1.72099	144.60	1734.20	0.111			$10^{-10}$
MMEACO -2 ( $\tau_{min}$ )	<b>-1.77349</b>	155.20	1861.40	0.100	$10^{-3}$		$10^{10}$
MMEACO -2 ( $\% \tau_{max}$ )	-1.74641	160.70	1927.40	0.125		$10^{-3}$	$10^{10}$
EACO -3	-1.73597	159.65	1914.80	0.083			0
MMEACO -3 ( $\tau_{min}$ )	-1.78502	149.60	1794.20	0.066	$10^{-4}$		$10^{10}$
MMEACO -3 ( $\% \tau_{max}$ )	<b>-1.78907</b>	152.85	1833.20	0.069		$10^{-3}$	$10^{10}$
EACO -4	-1.74590	159.90	1917.80	0.105			$10^{-10}$
MMEACO -4 ( $\tau_{min}$ )	-1.77605	151.80	1820.60	0.074	$10^{-3}$		$10^{10}$
MMEACO -4 ( $\% \tau_{max}$ )	<b>-1.77768</b>	137.15	1644.80	0.041		$10^{-4}$	$10^{10}$
EACO -5	-1.74901	148.20	1777.40	0.070			$10^{-10}$
MMEACO -5 ( $\tau_{min}$ )	-1.75947	151.30	1814.60	0.103	$10^{-7}$		$10^{10}$
MMEACO -5 ( $\% \tau_{max}$ )	<b>-1.77503</b>	130.70	1567.40	0.079		$10^{-4}$	$10^{10}$

Πίνακας 22. Παρουσίαση των καλύτερων αποτελεσμάτων για κάθε εκδοχή του αλγορίθμου MMEACO (-b1,  $N=31$ ,  $\alpha=1$ ,  $\beta=11$ ,  $r=0.05$ ,  $\rho=0.9$ ,  $c_f=0.001$ ,  $c_v=0.001$ ,  $m_{init}=5$ ,  $n_{ar}=4$ ,  $f_{gl}=5$ ).

Αλγόριθμος	M.O. Εμβαδού	M.O. Επαναλήψεων	M.O. Εκτιμήσεων	T.A.	$\tau_{min}$ ή $c$	$n_{ts}$	$n_{tri}$
MMEACO -2 ( $\tau_{min}$ )	<b>-1.77349</b>	155.20	1861.40	0.100	$10^{-3}$		
MMEACO -2 ( $\tau_{min}$ )	-1.77054	149.00	1787.00	0.109	$10^{-3}$	50	
MMEACO -2 ( $\tau_{min}$ ) +tri	-1.76179	155.85	1869.20	0.103	$10^{-3}$		30
MMEACO -3 ( $\% \tau_{max}$ )	-1.78907	152.85	1833.20	0.069	$10^{-3}$		
MMEACO -3 ( $\% \tau_{max}$ ) +ts	-1.79520	153.30	1838.60	0.063	$10^{-3}$	50	
MMEACO -3 ( $\% \tau_{max}$ ) +tri	<b>-1.79529</b>	136.10	1632.20	0.082	$10^{-3}$		50
MMEACO -4 ( $\% \tau_{max}$ )	-1.77768	137.15	1644.80	0.041	$10^{-4}$		
MMEACO -4 ( $\% \tau_{max}$ ) +ts	<b>-1.79122</b>	152.85	1833.20	0.087	$10^{-4}$	20	
MMEACO -4 ( $\% \tau_{max}$ ) +tri	-1.78117	146.05	1751.60	0.038	$10^{-4}$		60
MMEACO -4 ( $\% \tau_{max}$ ) +ts +tri	-1.79114	138.10	1656.20	0.088	$10^{-4}$	20	60
MMEACO -5 ( $\% \tau_{max}$ )	-1.77503	130.70	1567.40	0.079	$10^{-4}$		
MMEACO -5 ( $\% \tau_{max}$ ) +ts	<b>-1.78514</b>	132.55	1589.60	0.089	$10^{-4}$	20	
MMEACO -5 ( $\% \tau_{max}$ ) +tri	-1.77958	141.90	1701.80	0.091	$10^{-4}$		40
MMEACO -5 ( $\% \tau_{max}$ ) +ts +tri	-1.78359	130.75	1568.00	0.098	$10^{-4}$	20	40

Πίνακας 23. Ομαλοποίηση της φερομόνης, επαναρχικοποίηση του αλγορίθμου και συνδυασμένη δράση αυτών στη περίπτωση του MMEACO – παρουσίαση των καλύτερων αποτελεσμάτων για κάθε εκδοχή του αλγορίθμου ( $-b1$ ,  $N=31$ ,  $\alpha=1$ ,  $\beta=11$ ,  $r=0.05$ ,  $\rho=0.9$ ,  $c_f=0.001$ ,  $c_v=0.001$ ,  $m_{init}=5$ ,  $n_{ar}=4$ ,  $f_{gl}=5$ ,  $\delta=0.75$ ,  $\tau_o=10^{10}$ ).

### 8.3 ΠΑΡΟΥΣΙΑΣΗ ΑΠΟΤΕΛΕΣΜΑΤΩΝ

Μετά τη διαδικασία ρύθμισης (*tuning*) του αλγορίθμου EACO είμαστε πλέον σε θέση να παρουσιάσουμε διάφορα αποτελέσματα για το πρόβλημα της Αντίστροφης Σχεδίασης Μεμονωμένων Αεροτομών.

Για τους υπολογισμούς χρησιμοποιήθηκε ο εξής αλγόριθμος:

**MMEACO -4 ( $\% \tau_{max}$ ) -b1 +tri** με  $m_{init}=5$ ,  $N=31$ ,  $\alpha=1$ ,  $\beta=11$ ,  $r=0.05$ ,  $c_f=0.001$ ,  $c_v=0.001$ ,  $\rho=0.9$ ,  $n_{tri}=60$ ,  $\tau_o=10^{10}$ ,  $c=10^{-4}$  και μέγιστο αριθμό επαναλήψεων **200**.

Για τον υπολογισμό των κατανομών της πίεσης γύρω από τις αεροτομές χρησιμοποιήθηκε η μέθοδος των ιδιόμορφων σημείων (Panel Method) για αστρόβιλη και ασυμπίεστη ροή.

Οι δοκιμές πραγματοποιήθηκαν με τους εξής στόχους:

1. Αεροτομή NACA 0012 με γωνία προσβολής  $0^\circ$ .
2. Αεροτομή NACA 0012 με γωνία προσβολής  $3^\circ$ .
3. Αεροτομή NACA 0012 με γωνία προσβολής  $7^\circ$ .
4. Αεροτομή NACA 4412 με γωνία προσβολής  $-3^\circ$ .
5. Αεροτομή NACA 4412 με γωνία προσβολής  $0^\circ$ .

6. Αεροτομή NACA 4412 με γωνία προσβολής  $3^\circ$ .
7. Αεροτομή NACA 4412 με γωνία προσβολής  $7^\circ$ .
8. Πτερύγωση CRISP με γωνία προσβολής  $36^\circ$ , στερεότητα  $\sigma=0.73475$  και γωνία κλίσης της πτερύγωσης  $\gamma=-27.9^\circ$ .

Για τις αεροτομές NACA 0012 και NACA 4412 χρησιμοποιήθηκαν τα σημεία ελέγχου των πολυωνύμων Bezier που παρουσιάζονται στον παρακάτω πίνακα.

$x$	$y_{min}$	$y_{max}$	$x$	$y_{min}$	$y_{max}$
<b>Bezier Curve 1</b>			<b>Bezier Curve 2</b>		
1.0	0.0	0.0	0.0	0.0	0.0
0.9	-0.2	0.0	0.0	0.0	0.2
0.7	-0.2	0.0	0.1	0.0	0.2
0.5	-0.2	0.0	0.3	0.0	0.2
0.3	-0.2	0.0	0.5	0.0	0.2
0.1	-0.2	0.0	0.7	0.0	0.2
0.0	-0.2	0.0	0.9	0.0	0.2
0.0	0.0	0.0	1.0	0.0	0.0

**Πίνακας 24.** Σημεία ελέγχου των πολυωνύμων Bezier για τις αεροτομές NACA 0012 και NACA 4412.

Από τα αποτελέσματα για τη συμμετρική αεροτομή NACA 0012 (Εικόνα 44-Εικόνα 46) παρατηρούμε ότι η απόδοση του αλγορίθμου είναι πάρα πολύ καλή και μάλιστα συγκλίνει αρκετά γρήγορα. Ο αλγόριθμος καταφέρνει να προσεγγίσει την κατανομή της πίεσης του στόχου κάθε δοκιμής με μεγάλη ακρίβεια και επομένως και τη γεωμετρία.

Παρόμοια αποτελέσματα λαμβάνουμε και για την ασύμμετρη αεροτομή NACA 4412 (Εικόνα 47-Εικόνα 50). Ο αλγόριθμος σε όλες τις περιπτώσεις επιτυγχάνει να προσεγγίσει με αρκετή ακρίβεια τη γεωμετρία της αεροτομής.

Για την πτερύγωση CRISP αρχικά χρησιμοποιήσαμε τα σημεία ελέγχου των Bezier του πίνακα της διπλανής σελίδας (Πίνακας 25). Το εύρος μετακίνησης κάθε σημείου είναι ίδιο με αυτό των σημείων που χρησιμοποιήθηκαν στις αεροτομές NACA 0012 και NACA 4412 (Πίνακας 24) για να είναι κατά κάποιον τρόπο συγκρίσιμα τα αποτελέσματα. Οι πρώτες δοκιμές με στόχο την πτερύγωση CRISP δεν έδωσαν αρκετά ικανοποιητικά αποτελέσματα (Εικόνα 51). Ο αλγόριθμος σε κάθε περίπτωση δεν κατάφερε να υπολογίσει μια τιμή ελαχίστου εμβαδού που να προσεγγίζει τις αντίστοιχες τιμές των δοκιμών με τις αεροτομές NACA 0012 και NACA 4412.

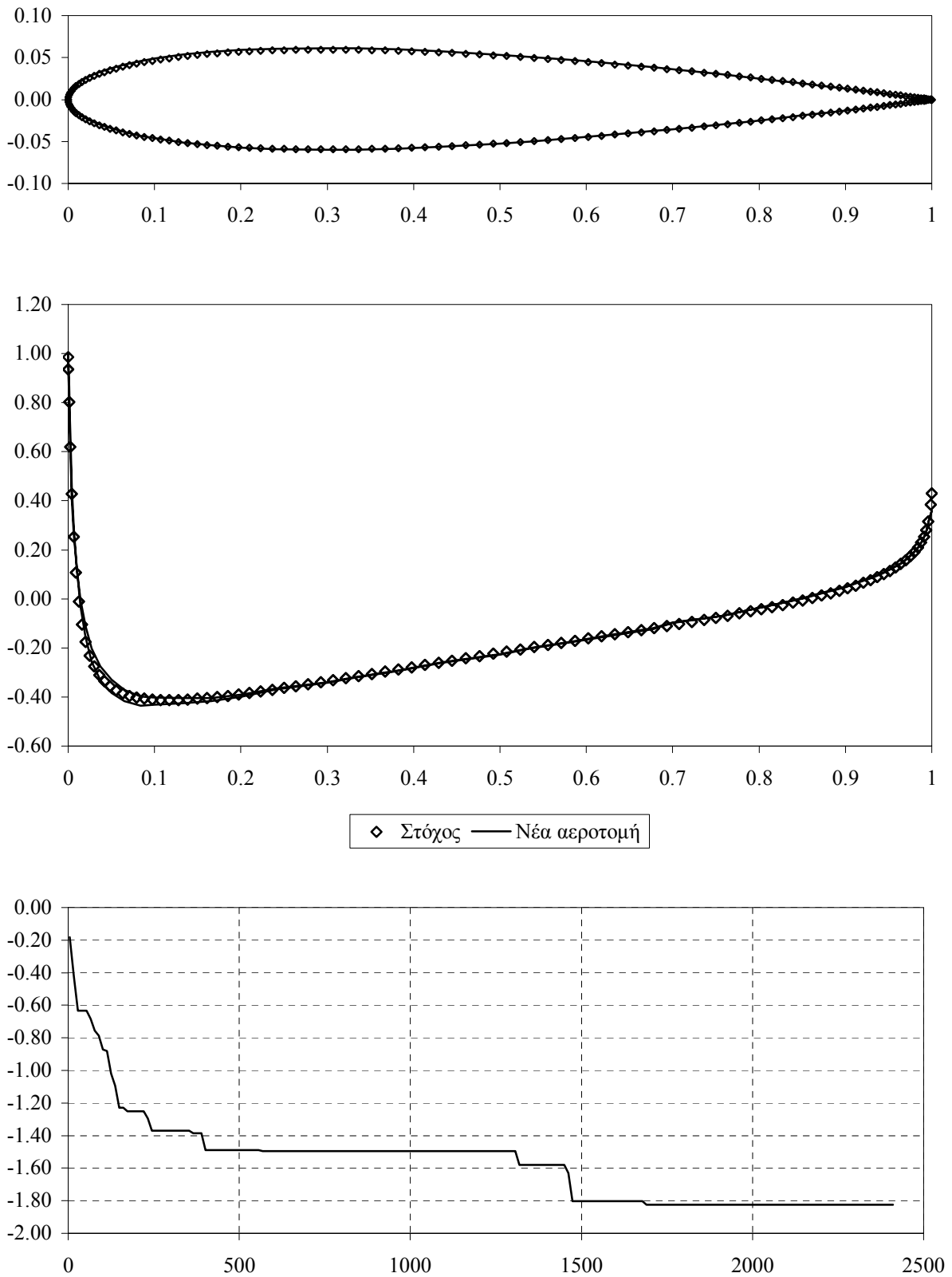
Για να βελτιώσουμε την απόδοση του MMEACO μειώσαμε το εύρος των δυνατών τιμών των σημείων ελέγχου των πολυωνύμων Bezier (Πίνακας 26). Με βάση τις νέες αυτές τιμές επαναλάβαμε την εκτέλεση του αλγορίθμου ο οποίος και έδωσε λίγο καλύτερα αποτελέσματα (Εικόνα 52).

$x$	$y_{min}$	$y_{max}$	$x$	$y_{min}$	$y_{max}$
<b>Bezier Curve 1</b>			<b>Bezier Curve 2</b>		
1.00	0.00	0.00	0.00	0.00	0.00
1.00	-0.20	0.00	0.00	0.00	0.20
0.98	-0.10	0.10	0.02	0.00	0.20
0.90	-0.10	0.10	0.10	0.00	0.20
0.70	-0.10	0.10	0.30	0.00	0.20
0.50	-0.10	0.10	0.50	0.00	0.20
0.30	-0.10	0.10	0.70	0.00	0.20
0.10	-0.10	0.10	0.90	0.00	0.20
0.02	-0.10	0.10	0.98	0.00	0.20
0.00	-0.20	0.00	1.00	0.00	0.20
0.00	0.00	0.00	1.00	0.00	0.00

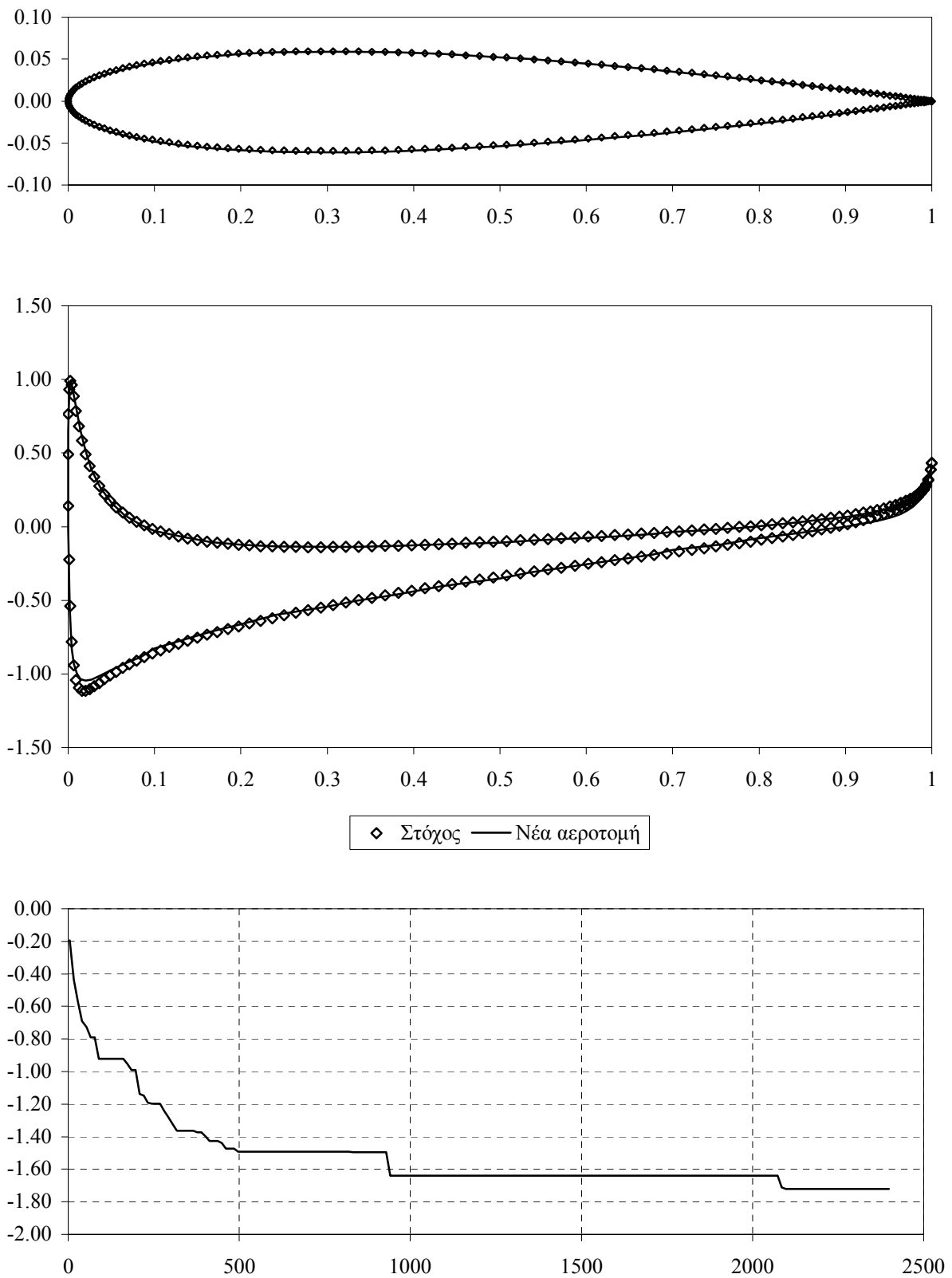
Πίνακας 25. Τα σημεία ελέγχου των πολυωνύμων Bezier της 1<sup>ης</sup> δοκιμής για την πετύγωση CRISP.

$x$	$y_{min}$	$y_{max}$	$x$	$y_{min}$	$y_{max}$
<b>Bezier Curve 1</b>			<b>Bezier Curve 2</b>		
1.00	0.00	0.00	0.00	0.00	0.00
1.00	-0.03	0.00	0.00	0.00	0.07
0.98	-0.03	0.03	0.02	0.00	0.07
0.90	-0.03	0.03	0.10	0.00	0.07
0.70	-0.03	0.03	0.30	0.00	0.07
0.50	-0.03	0.03	0.50	0.00	0.07
0.30	-0.03	0.03	0.70	0.00	0.07
0.10	-0.03	0.03	0.90	0.00	0.07
0.02	-0.03	0.03	0.98	0.00	0.07
0.00	-0.03	0.00	1.00	0.00	0.07
0.00	0.00	0.00	1.00	0.00	0.00

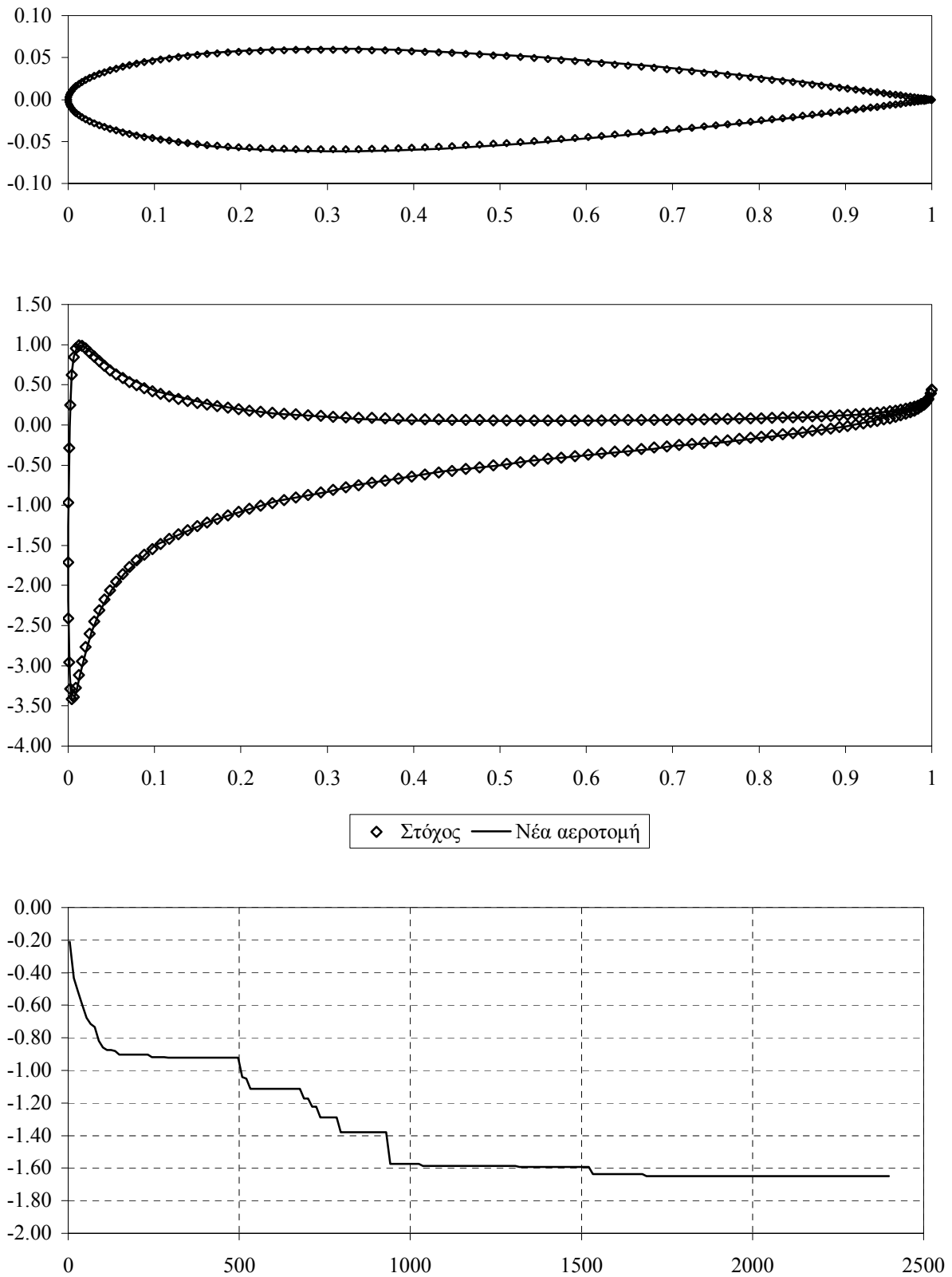
Πίνακας 26. Τα σημεία ελέγχου των πολυωνύμων Bezier της 2<sup>ης</sup> δοκιμής για την πετύγωση CRISP.



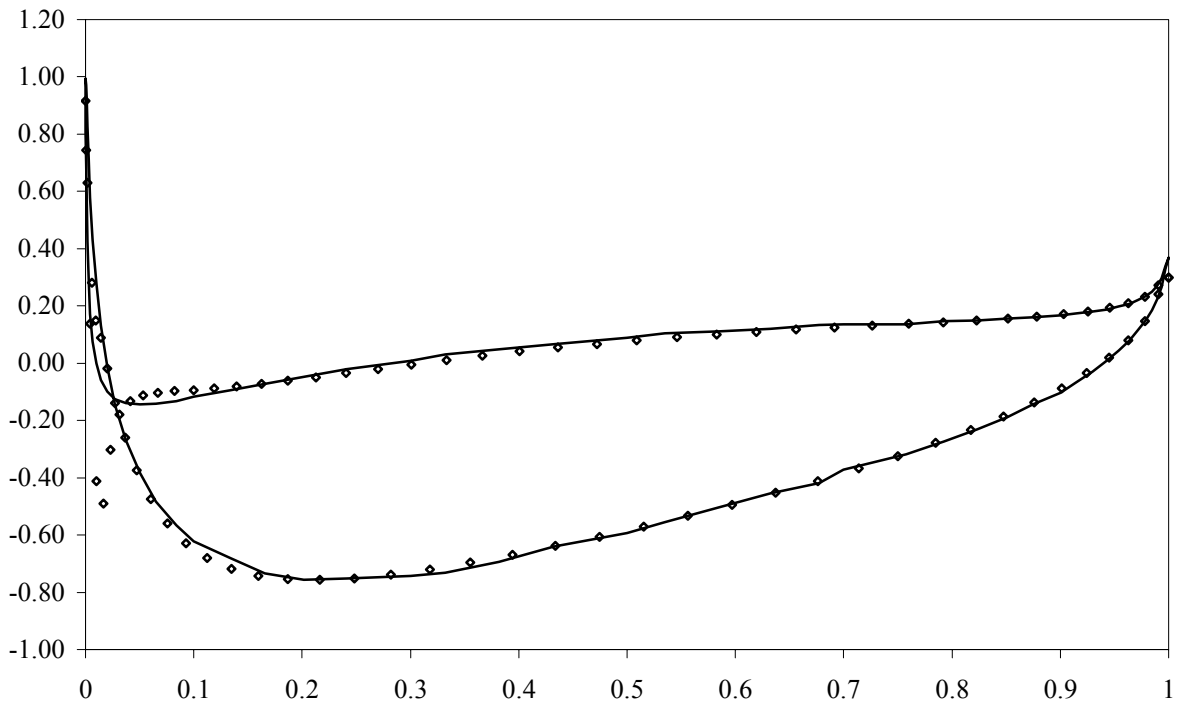
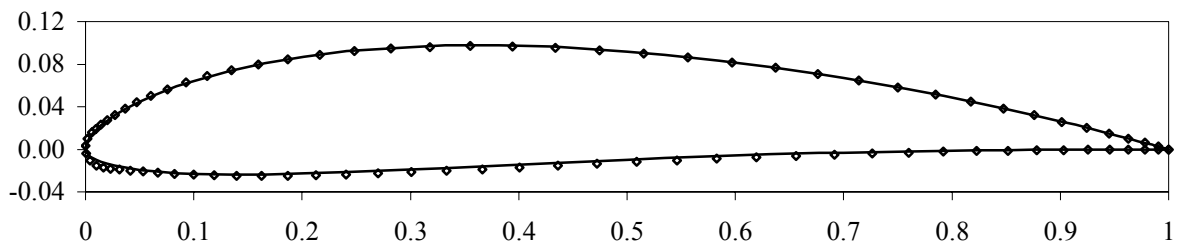
Εικόνα 44. Αεροτομή NACA 0012 με  $0^\circ$  γωνία προσβολής (α) Γεωμετρία της αεροτομής ( $y/c - x/c$ ) (β) Κατανομή πιέσεων ( $c_p - x/c$ ) (γ) Σύγκλιση του αλγορίθμου (Ο λογάριθμος του ελάχιστου εμβαδού που υπολογίστηκε συναρτήσει του αριθμού των εκτελέσεων της μεθόδου των ιδιόμορφων σημείων).



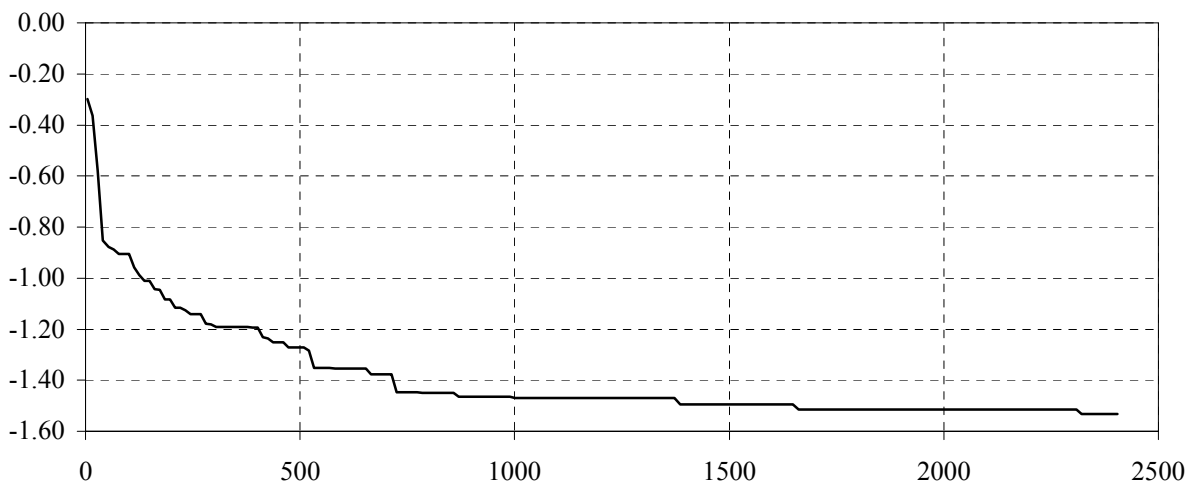
Εικόνα 45. Αεροτομή NACA 0012 με 3° γωνία προσβολής (α) Γεωμετρία της αεροτομής ( $y/c - x/c$ ) (β) Κατανομή πιέσεων ( $c_p - x/c$ ) (γ) Σύγκλιση του αλγορίθμου (Ο λογάριθμος του ελάχιστου εμβαδού που υπολογίσθηκε συναρτήσει του αριθμού των εκτελέσεων της μεθόδου των ιδιόμορφων σημείων).



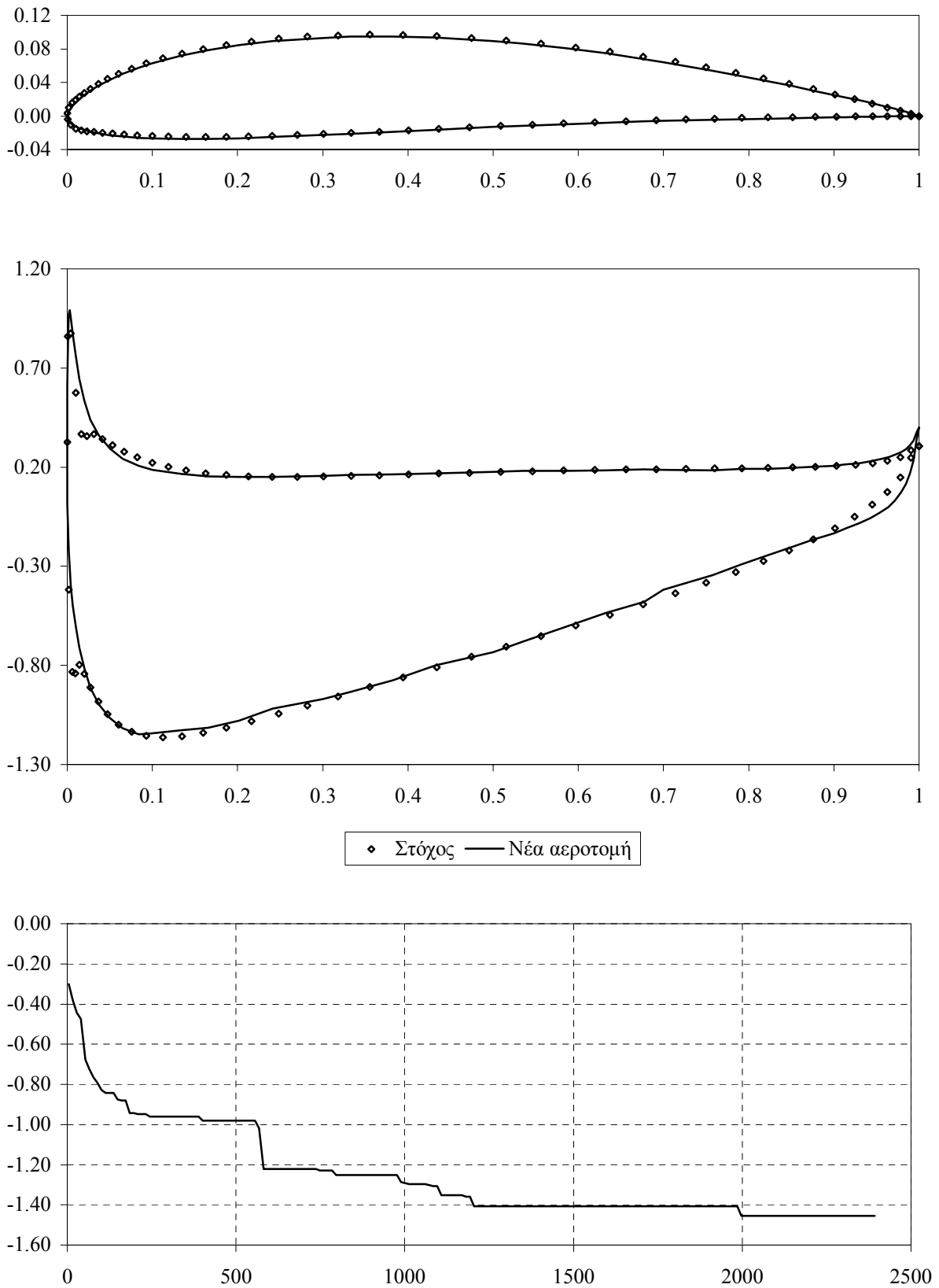
Εικόνα 46. Αεροτομή NACA 0012 με  $7^\circ$  γωνία προσβολής (α) Γεωμετρία της αεροτομής ( $y/c - x/c$ ) (β) Κατανομή πιέσεων ( $c_p - x/c$ ) (γ) Σύγκλιση του αλγορίθμου (Ο λογάριθμος του ελάχιστου εμβαδού που υπολογίστηκε συναρτήσει του αριθμού των εκτελέσεων της μεθόδου των ιδιόμορφων σημείων).



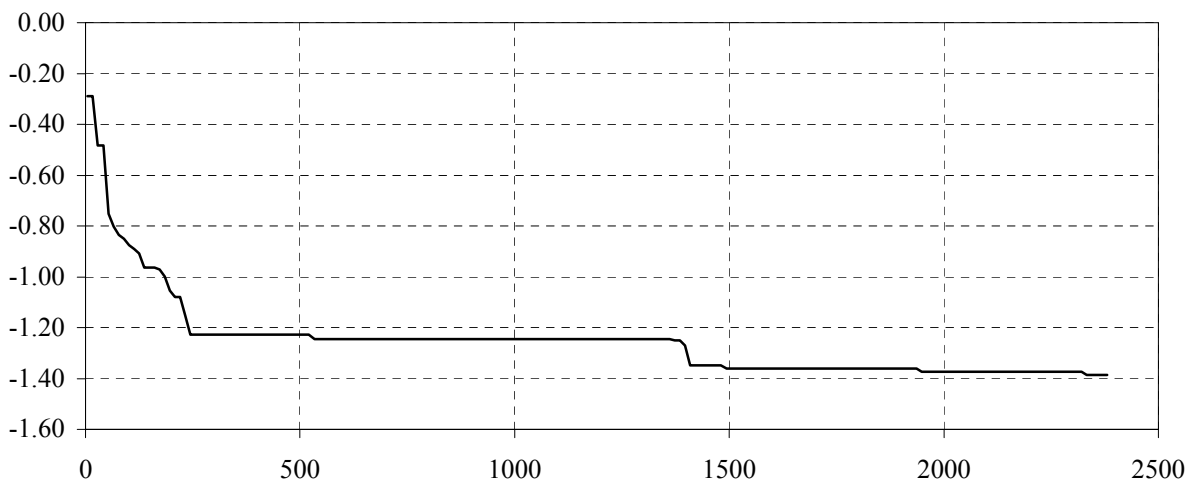
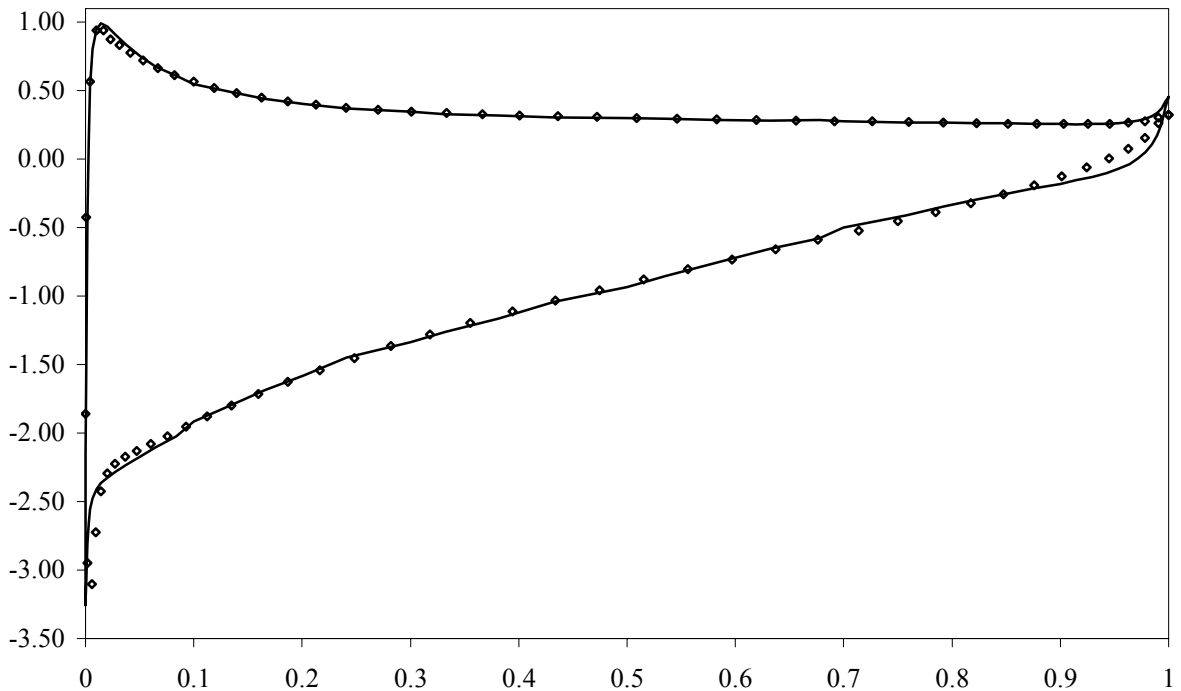
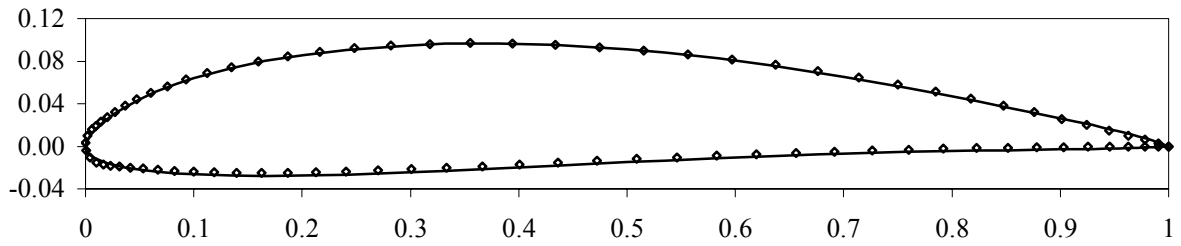
◆ Στόχος — Νέα αεροτομή



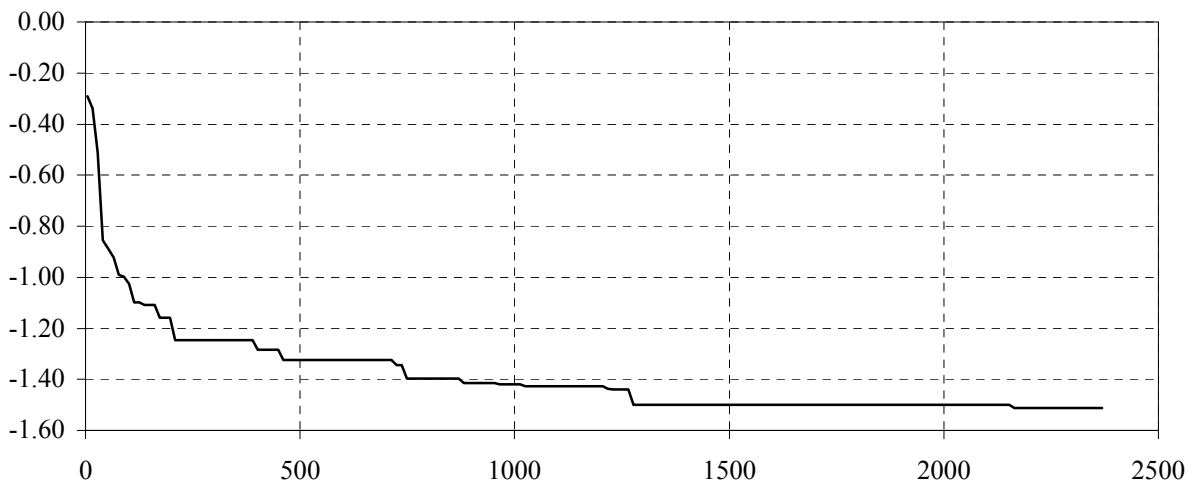
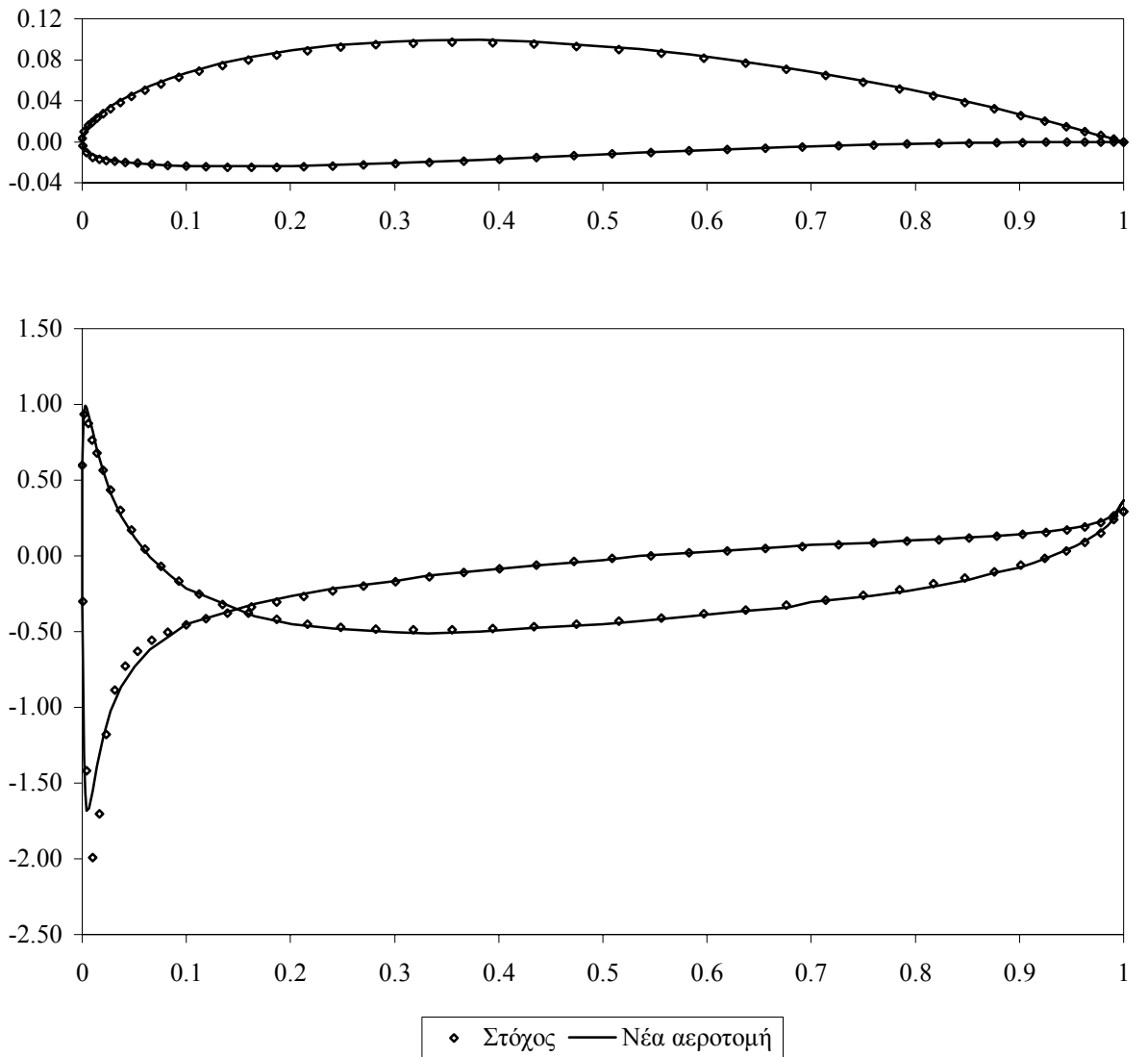
Εικόνα 47. Αεροτομή NACA 4412 με  $0^\circ$  γωνία προσβολής (α) Γεωμετρία της αεροτομής ( $y/c - x/c$ ) (β) Κατανομή πιέσεων ( $C_p - x/c$ ) (γ) Σύγκλιση του αλγορίθμου (Ο λογάριθμος του ελάχιστου εμβαδού που υπολογίστηκε συναρτήσει του αριθμού των εκτελέσεων της μεθόδου των ιδιόμορφων σημείων).



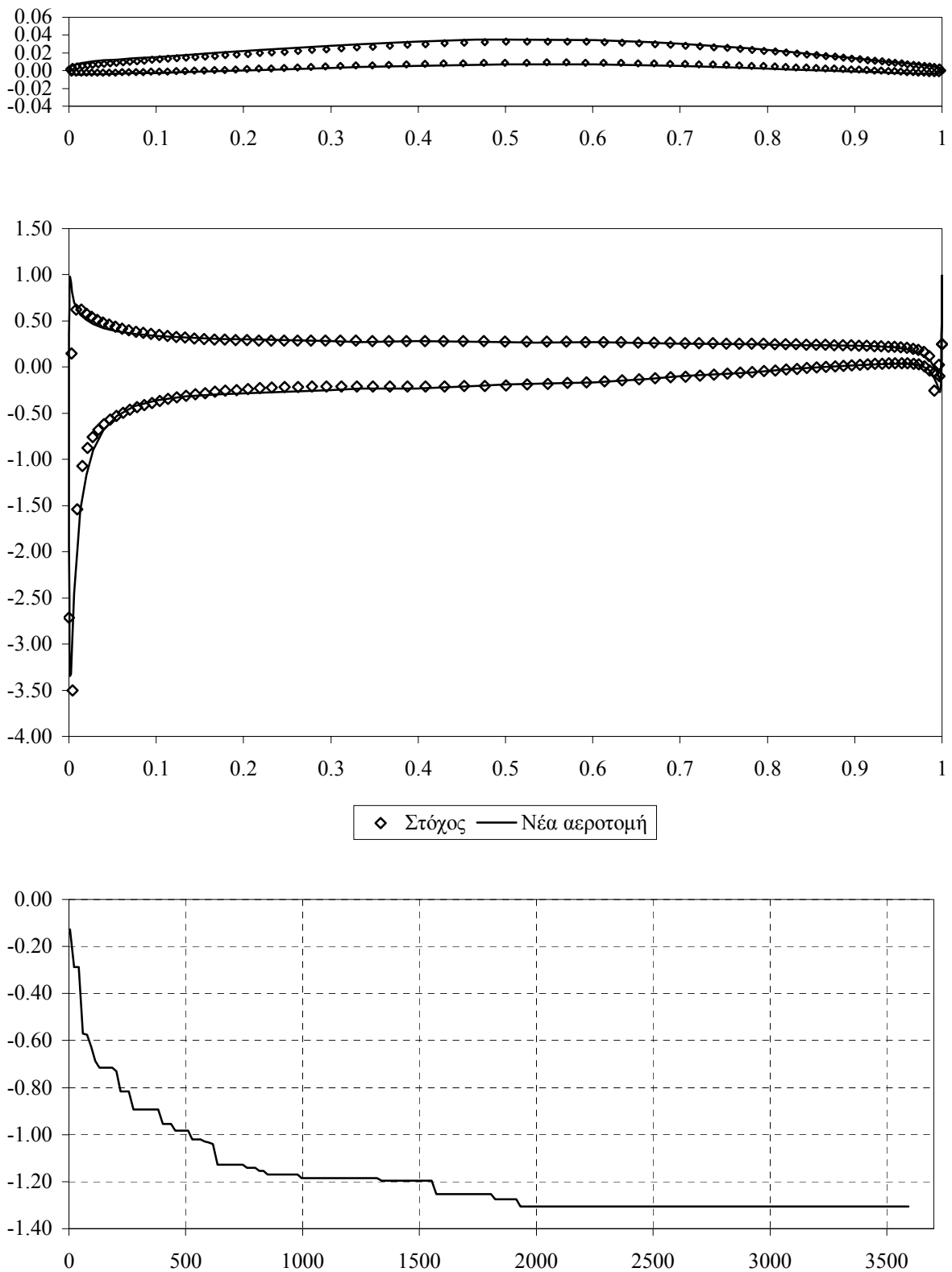
Εικόνα 48. Αεροτομή NACA 4412 με 3° γωνία προσβολής (α) Γεωμετρία της αεροτομής ( $y/c - x/c$ ) (β) Κατανομή πιέσεων ( $c_p - x/c$ ) (γ) Σύγκλιση του αλγορίθμου (Ο λογάριθμος του ελάχιστου εμβαδού που υπολογίστηκε συναρτήσει του αριθμού των εκτελέσεων της μεθόδου των ιδιόμορφων σημείων).



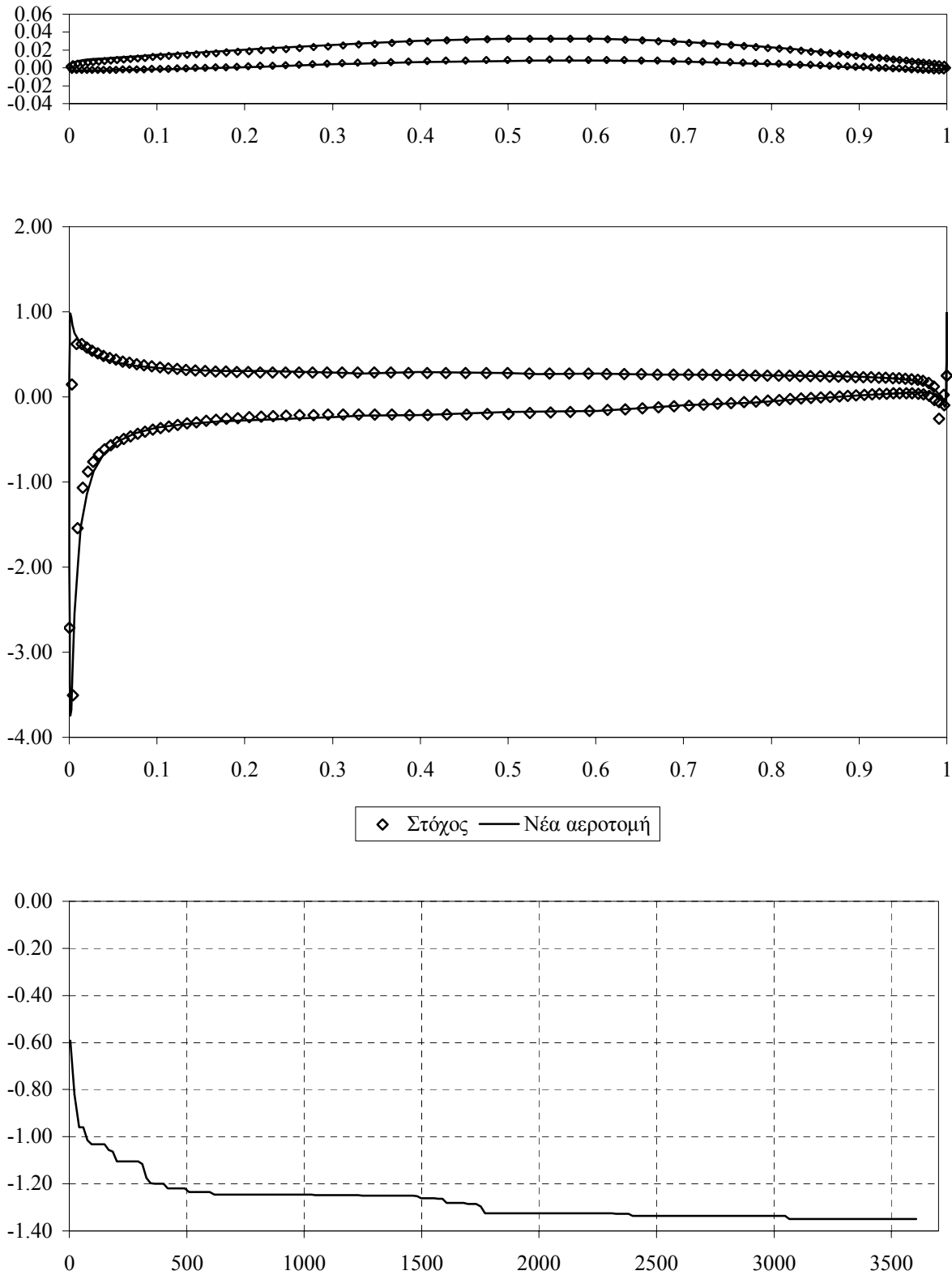
Εικόνα 49. Αεροτομή NACA 4412 με  $7^\circ$  γωνία προσβολής (α) Γεωμετρία της αεροτομής ( $y/c - x/c$ ) (β) Κατανομή πιέσεων ( $c_p - x/c$ ) (γ) Σύγκλιση του αλγορίθμου (Ο λογάριθμος του ελάχιστου εμβαδού που υπολογίστηκε συναρτήσει του αριθμού των εκτελέσεων της μεθόδου των ιδιόμορφων σημείων).



Εικόνα 50. Αεροτομή NACA 4412 με  $-3^\circ$  γωνία προσβολής (α) Γεωμετρία της αεροτομής ( $y/c - x/c$ ) (β) Κατανομή πιέσεων ( $c_p - x/c$ ) (γ) Σύγκλιση του αλγορίθμου (Ο λογάριθμος του ελάχιστου εμβαδού που υπολογίσθηκε συναρτήσει του αριθμού των εκτελέσεων της μεθόδου των ιδιόμορφων σημείων).



Εικόνα 51. Η πτερύγωση CRISP που υπολογίζεται χρησιμοποιώντας τα σημεία ελέγχου των πολυωνύμων Bezier της 1<sup>ης</sup> δοκιμής (Πίνακας 25) – (α) Γεωμετρία της αεροτομής ( $y/c - x/c$ ) (β) Κατανομή πιέσεων ( $c_p - x/c$ ) (γ) Σύγκλιση του αλγορίθμου (Ο λογάριθμος του ελάχιστου εμβαδού που υπολογίσθηκε συναρτήσει του αριθμού των εκτελέσεων της μεθόδου των ιδιόμορφων σημείων).



Εικόνα 52. Η περύγωση CRISP που υπολογίζεται χρησιμοποιώντας τα σημεία ελέγχου των πολωνόμων Bezier της  $2^{15}$  δοκιμής (Πίνακας 26) – (α) Γεωμετρία της αεροτομής  $(y/c - x/c)$  (β) Κατανομή πιέσεων  $(c_p - x/c)$  (γ) Σύγκλιση του αλγορίθμου (Ο λογάριθμος του ελάχιστου εμβαδού που υπολογίστηκε συναρτήσει του αριθμού των εκτελέσεων της μεθόδου των ιδιόμορφων σημείων).

## 9 ΣΥΜΠΕΡΑΣΜΑΤΑ

Στην παρούσα διπλωματική εργασία επιχειρήθηκε η εκτενής ανάλυση των αλγορίθμων βελτιστοποίησης που εμπνέονται από τη συμπεριφορά των μυρμηγκιών κατά την αναζήτηση και συγκομιδή τροφής. Το συγκεκριμένο ερευνητικό πεδίο αυτή τη στιγμή βρίσκεται υπό συνεχή ανάπτυξη και με πολλές εφαρμογές τόσο σε διακριτά όσο και σε συνεχή προβλήματα βελτιστοποίησης.

Ο νέος αλγόριθμος που παρουσιάστηκε (EACO) έχει δώσει αρκετά καλά αποτελέσματα στο πρόβλημα του Αντίστροφου Σχεδιασμού Αεροτομών τα οποία είναι συγκρίσιμα με αυτά πιο ανεπτυγμένων αλγορίθμων όπως για παράδειγμα των Γενετικών Αλγορίθμων. Σίγουρα, όμως, χρειάζεται περαιτέρω ανάπτυξη για να μπορέσει να αξιοποιηθεί σε ερευνητικές ή παραγωγικές διαδικασίες.

Η ανάλυση του EACO έδειξε ότι όπως και στην περίπτωση των διακριτών αλγορίθμων ACO τα καλύτερα αποτελέσματα λαμβάνονται όταν εναποτίθεται φερομόνη μόνο από το μυρμήγκι που έχει υπολογίσει τη βέλτιστη λύση του προβλήματος ή/και τη βέλτιστη λύση κάθε επανάληψης. Η ταχεία εξάτμιση της φερομόνης είναι απαραίτητη για την επιτάχυνση της σύγκλισης του αλγορίθμου. Σημαντικό είναι το γεγονός ότι ενώ τόσο η πληροφορία της φερομόνης όσο και της ορατότητας είναι απαραίτητες για την καλή απόδοση του αλγορίθμου, τα καλύτερα αποτελέσματα λαμβάνονται όταν ενισχύουμε την ορατότητα και μειώνουμε την επιρροή της φερομόνης. Επίσης, είναι φανερό η ευεργετική χρήση των ορίων στη φερομόνη όπως και στην περίπτωση του MMAS. Τέλος, κατά την ανάλυση επιβεβαιώθηκε η θετική δράση των μηχανισμών ομαλοποίησης της φερομόνης και επαναρχικοποίησης του αλγορίθμου.

Μερικές προτάσεις για μελλοντική εργασία γύρω από την βελτίωση της απόδοσης του αλγορίθμου EACO είναι:

- Εφαρμογή **τοπικής έρευνας** (*local search*) έχοντας ως αφετηρία την διαδρομή που έχει ήδη κατασκευάσει ένα μυρμήγκι. Για την εξοικονόμηση υπολογιστικού χρόνου καλό είναι η τοπική έρευνα να επικεντρώνεται στο ταξίδι του μυρμηγκιού που έχει βρει τη βέλτιστη έως τώρα λύση (*global best ant* ή *global leader*) ή/και του μυρμηγκιού που βρίσκει τη βέλτιστη λύση κάθε επανάληψης (*iteration best ant*).
- Χρήση **τοπικής ανανέωσης της φερομόνης** όπως στην περίπτωση του ACS (Κεφάλαιο 4.2).

- Συνδυασμός του αλγορίθμου EACO με διάφορους άλλους, όπως για παράδειγμα TS, GA και SA.
- Πραγματοποίηση **παράλληλης επεξεργασίας** είτε αφιερώνοντας έναν επεξεργαστή σε κάθε μυρμήγκι είτε δημιουργώντας ανεξάρτητες αποικίες μυρμηγκιών (distributed colonies) οι οποίες ανταλλάσσουν δεδομένα ανά τακτά χρονικά διαστήματα.
- Χρησιμοποίηση διαφορετικών μηχανισμών ομαλοποίησης της φερομόνης και επαναρχικοποίησης του αλγορίθμου. Για παράδειγμα, θα μπορούσαμε να κάνουμε ομαλοποίηση της φερομόνης ανά συγκεκριμένο αριθμό επαναλήψεων. Επίσης, φαίνεται καλύτερο μετά την επαναρχικοποίηση να γίνεται εναπόθεση φερομόνης κυρίως από το μυρμήγκι που έχει βρει τη βέλτιστη λύση μετά την επαναρχικοποίηση.

Το βασικότερο μειονέκτημα του αλγορίθμου EACO είναι το μεγάλο πλήθος των παραμέτρων που πρέπει να ρυθμιστούν έτσι ώστε να είναι βέλτιστη η απόδοσή του. Η βέλτιστη ρύθμιση των παραμέτρων του αλγορίθμου είναι και αυτή ένα δύσκολο πρόβλημα βελτιστοποίησης και επομένως ο χρήστης θα πρέπει να συμβιβαστεί με μια περιοχή τιμών που θα δίνει καλά αποτελέσματα.

Ένα επίσης σημαντικό πρόβλημα του EACO είναι η δύσκολη εφαρμογή του σε άλλα προβλήματα βελτιστοποίησης. Το γεγονός ότι η εφαρμογή του περιορίζεται σε προβλήματα που έχουν γραφική αναπαράσταση κάνει δύσκολη την ευρεία χρήση του και σε άλλες μορφές προβλημάτων. Αναφορικά με τη χρήση του αλγορίθμου και σε άλλα προβλήματα βελτιστοποίησης, έγινε μια πρώτη προσπάθεια εφαρμογής του EACO στην εύρεση ελαχίστου ή μεγίστου συνεχών συναρτήσεων, αλλά δεν έδωσε ενθαρρυντικά αποτελέσματα. Προφανώς, απαιτείται μια βαθύτερη μελέτη του συγκεκριμένου ζητήματος.

Σε κάθε περίπτωση πάντως ο EACO (ή ο MMEACO) είναι ένας αλγόριθμος που έδωσε ικανοποιητικά αποτελέσματα στο πρόβλημα της Αντίστροφης Σχεδίασης Αεροτομών και σίγουρα μπορεί μελλοντικά να επιτελέσει ένα χρήσιμο εργαλείο για την έρευνα και την παραγωγή.

## 10 ΒΙΒΛΙΟΓΡΑΦΙΑ

- [1] I.H. Abbot, A.E. Doenhoff, Theory of Wing Sections, Dover Publications Inc, New York, 1959.
- [2] R. Beekers, J.L. Deneubourg, S. Goss, Trails and U-turns in the Selection of a Path by the Ant *Lasius niger*, J. theor. Biol. (1992) 159, 397-415.
- [3] G. Bilchev, I.C. Parmee, The ant colony metaphor for searching continuous design spaces, In: T. Fogarty, ed. Lecture Notes in Computer Science, 993, Springer Verlag, (1995) 25-39.
- [4] G. Bilchev, I.C. Parmee, Constrained optimization with an ant colony search model, Proc. of the ACEDC'96, Plymouth, UK, 145-151.
- [5] E. Bonabeau, G. Theraulaz, J. Deneubourg, The synchronization of recruitment-based activities in ants, BioSystems 45 (1998), 195–211.
- [6] E. Bonabeau, M. Dorigo, G. Theraulaz, Swarm Intelligence: From Natural to Artificial Systems, Oxford University Press, Oxford, 1999.
- [7] U. Boryczka, Learning with Delayed Rewards in Ant Systems for the Job-Shop Scheduling Problem, L. Polkowski and A. Skowron (Eds.): RSCTC'98, LNAI 1424, pp. 271-274, Springer-Verlag Berlin Heidelberg, 1998.
- [8] D.G. Bounds, New optimization methods from physics and biology, Nature, Vol. 329, 17 (1987), pp. 215-219.
- [9] B. Bullnheimer, R.F. Hartl, C. Strauss, A new rank-based version of the Ant System: a computational study, Central Eur. J. Oper. Res. Econom. 7 (1) (1999) 25–38.
- [10] Β. Γκόγκος, Ρ. Ε. Κινγκ, Εισαγωγή σε μεθόδους αριστοποίησης της Υπολογιστικής Νοημοσύνης – Ι. Εξελικτικός Υπολογισμός, Τεχνικά Χρονικά, Ιαν. – Φεβρουάριος '98.
- [11] Β. Γκόγκος, Ρ. Ε. Κινγκ, Εισαγωγή σε μεθόδους αριστοποίησης της Υπολογιστικής Νοημοσύνης – ΙΙ. Προσομοιωμένη Ανόπτηση, Τεχνικά Χρονικά, Ιαν. – Φεβρουάριος '98.
- [12] G. D. Caro, M. Dorigo, AntNet: Distributed Stigmergetic Control for Communication Networks, Journal of Artificial Intelligence Research 9 (1998) 317-365.
- [13] C.S. Chang, L. Tian, F.S. Wen, A new approach to fault section estimation in power systems using Ant system, Electric Power Systems Research 49 (1999) 63–70.
- [14] T.S. Collet, M. Collet, Path integration in insects, Current opinion in Neurobiology (2000) 10:757-762.

- [15] A. Colomi, M. Dorigo, F. Maffioli, V. Maniezzo, G. Righini and M. Trubian, Heuristics from Nature for Hard Combinatorial Optimization Problems, *International Transactions in Operational Research* Vol. 3, No. 1, pp. 1-21, 1996.
- [16] D. Corne, M. Dorigo, F. Glover, *New Ideas in Optimization*, McGraw-Hill, 1999.
- [17] J.L. Deneubourg, S. Goss, N. Franks, J.M. Pasteels, The Blind Leading the Blind: Modeling Chemically Mediated Army Ant Raid Patterns, *Journal of Insect Behavior*, Vol. 2, No. 5, 1989, 719-725.
- [18] J.L. Deneubourg, S. Aron, S. Goss, J.M. Pasteels, The Self-Organizing Exploratory Pattern of the Argentine Ant, *Journal of Insect Behavior*, Vol. 3, No. 2, 1990, 159-169.
- [19] M. Dorigo, V. Maniezzo, A. Colomi, *Ant System: An Autocatalytic Optimizing Process*, Technical Report 91-016, Dip. di Elettronica, Politecnico di Milano, Italy, 1991.
- [20] M. Dorigo, V. Maniezzo, A. Colomi, The ant system: optimization by a colony of cooperating agents, *IEEE Trans. Systems Man Cybernet. B* 26 (1) (1996) 29–41.
- [21] M. Dorigo, L.M. Gambardella, Ant colony system: a cooperative learning approach to the traveling salesman problem, *IEEE Trans. Evol. Comput.* 1 (1) (1997) 53–66.
- [22] M. Dorigo, L.M. Gambardella, Ant colonies for the traveling salesman problem, *BioSystems* 43 (1997) 73–81.
- [23] M. Dorigo, G. Di Caro, L.M. Gambardella, Ant algorithms for discrete optimization, *Artificial Life* 5 (2) (1999) 137–172.
- [24] M. Dorigo, G. Di Caro, T. Stützle, Guest Editorial: Ant algorithms, *Future Generation Computer Systems* 16 (2000) v-vii.
- [25] M. Dorigo, E. Bonabeau, G. Theraulaz, Ant algorithms and stigmergy, *Future Generation Computer Systems* 16 (2000) 851-871.
- [26] G. Forsythe, C.B. Moler, *Computer Solution of Linear Algebraic Systems*, Prentice-Hall, 1967
- [27] K.C. Giannakoglou, Acceleration of genetic algorithms using artificial neural networks – theoretical background, von Karman Institute LS 2000-07, May 2000
- [28] K.C. Giannakoglou, A.P. Giotis, Acceleration of genetic algorithms using artificial neural networks – application of the method, von Karman Institute LS 2000-07, May 2000
- [29] S. Goss, S. Aron, J.L. Deneubourg, J.M. Pasteels, Self-organized shortcuts in the Argentine ant, *Naturwissenschaften* 76 (1989) 579–581.
- [30] W.J. Gutjahr, A Graph-based Ant System and its convergence, *Future Generation Comput. Systems* 16 (2000) 873–888.

- [31] B. A. Huberman, R. M. Lukose, T. Hogg, An Economics Approach to Hard Computational Problems, Science, Vol. 275, 3 January 1997.
- [32] V.K. Jayaraman, B.D. Kulkarni, S. Karale, P. Shelokar, Ant colony framework for optimal design and scheduling of batch plants, Computers and Chemical Engineering 24 (2000) 1901–1912.
- [33] V.K. Jayaraman, B.D. Kulkarni, K. Gupta, J. Rajesh, H.S. Kusumaker, Dynamic optimisation of fed-batch bioreactors using the ant algorithm, Biotechnology Progress, Volume 17, Issue 1, 2001, Pages 81-88.
- [34] Σ. Ν. Καπλάνη, Η Μέθοδος Monte-Carlo, Θεσσαλονίκη, 1978.
- [35] J. L. Kerrebrock, Aircraft Engines and Gas Turbines, The MIT Press, 5<sup>th</sup> printing, 1983.
- [36] V. Maniezzo, A. Carbonaro, An ANTS heuristic for the frequency assignment problem, Future Generation Computer Systems 16 (2000) 927-935.
- [37] M. Mathur, S.B. Karale, S. Priye, V.K. Jayaraman, B.D. Kulkarni, Ant Colony Approach to Continuous Function Optimization, Ind. Eng. Chem. Res. Vol. 39, No 10, (2000), 3814-3822
- [38] Z. Michalewicz, D. B. Fogel, How to solve it: Modern Heuristics, Springer, 2000.
- [39] R. Michel, M. Middendorf, An island model based Ant System with lookahead for the shortest supersequence problem, in: A.E. Eiben, T. Back, M. Schoenauer, H.P. Schwefel (Eds.), Proceedings of PPSN-V, Fifth International Conference on Parallel Problem Solving from Nature, Springer, Berlin, Germany, 1998, pp. 692–701.
- [40] M. Middendorf, F. Reischle, H. Schmeck, Information Exchange in Multi Colony Ant Algorithms, J. Rolim et al. (Eds.): IPDPS 2000 Workshops, LNCS 1800, pp. 645-652, Springer-Verlag Berlin Heidelberg, 2000.
- [41] M.M. Millonas, A Connectionist Type Model of Self-organized Foraging and Emergent Behavior in Ant Swarms, J. theor. Biol. (1992) 159, 529-552.
- [42] N. Monmarché, G. Venturini, M. Slimane, On how *Pachycondyla apicalis* ants suggest a new search algorithm, Future Generation Computer Systems 16 (2000) 937–946.
- [43] Κ. Δ. Παπαηλίου, Κ. Μ. Μαθιουδάκης, Κ. Χ. Γιαννάκογλου, Εισαγωγή στις Θερμικές Στροβιλομηχανές, Εκδόσεις ΕΜΠ, Αθήνα, 1997.
- [44] C. Peterson, B. Soderberg, Artificial Neural Networks, Local Search in Combinatorial Optimization, Edited by E. Aarts and J. K. Lenstra, John Wiley and Sons Ltd., 1997.
- [45] J.N. Reddy, An Introduction to the Finite Element Method, McGraw-Hill, 1986.

- [46] Y.H. Song, C.S. Chou, T.J. Stonham, Combined heat and power economic dispatch by improved ant colony search algorithm, *Electric Power Systems Research* 52 (1999) 115–121.
- [47] T. Stützle, Parallelization Strategies for Ant Colony Optimization, A.E. Eiben et al. (Eds.): *PPSN V, LNCS 1498*, pp. 722-731, Springer-Verlag Berlin Heidelberg, 1998.
- [48] T. Stützle, An ant approach to the flow shop problem, *Proceedings of the Sixth European Congress on Intelligent Techniques and Soft Computing (EUFIT'98)*, Vol. 3, Verlag Mainz, Aachen, Germany, 1998, 1560–1564.
- [49] T. Stützle, H. Hoos, MAX-MIN Ant System, *Future Generation Computer Systems* 16 (2000) 889-914.
- [50] G. Theraulaz, E. Bonabeau, A brief history of stigmergy, *Artificial Life* 5 (1999) 97–116.
- [51] C.J. Tong, H.C. Lau, A. Lim, Ant Colony Optimization for the Ship Berthing Problem, P.S. Thiagarajan, R. Yap (Eds.): *ASIAN'99, LNCS 1742*, 359-370, Springer-Verlag Berlin Heidelberg, 1999.
- [52] M. Wodrich, G. Bilchev, Cooperative distributed search: the ants' way. *Control and Cybernetics*, Vol. 26 No. 3 (1997) 413–445.
- [53] T. Zervogiannis, V. Assouti, K. Gagas, A. Kaounis, K.C. Giannakoglou, Inverse Design of Aerodynamic Shapes using Ant Colony Optimization, 6<sup>ο</sup> Συνέδριο Μηχανικής, Θεσσαλονίκη, Ιούλιος 2001.