



Ειδική Υπηρεσία Διαχείρισης και Εφαρμογής Δράσεων
στους τομείς Έρευνας, Τεχνολογικής Ανάπτυξης και Καινοτομίας

ΕΝΙΑΙΑ ΔΡΑΣΗ ΚΡΑΤΙΚΩΝ ΕΝΙΣΧΥΣΕΩΝ
ΕΡΕΥΝΑΣ, ΤΕΧΝΟΛΟΓΙΚΗΣ ΑΝΑΠΤΥΞΗΣ
& ΚΑΙΝΟΤΟΜΙΑΣ

ΕΡΕΥΝΩ – ΔΗΜΙΟΥΡΓΩ – ΚΑΙΝΟΤΟΜΩ

ΕΝΟΤΗΤΑ ΕΡΓΑΣΙΑΣ 1: ΔΙΕΡΕΥΝΗΣΗ ΑΛΓΟΡΙΘΜΩΝ ΒΕΛΤΙΣΤΟΠΟΙΗΣΗΣ

Π1.1 Έκθεση προόδου σχετικά με τους αλγορίθμους

Κωδικός Έργου:	T1ΕΔΚ-05603
Τίτλος Έργου:	Ολιστική Υπολογιστική Πλατφόρμα Βελτιστοποίησης Υψηλών Επιδόσεων – HP-OCP
Συντάκτες:	Νίκος Δ. Λαγαρός, Γιώργος Καζάκης, Στέφανος Σωτηρόπουλος, Σταύρος Ξυνογαλάς, Χαρά Μητροπούλου, Νίκος Καλλιώρας, Σταύρος Χατζηελευθερίου, Σπύρος Δαμικούκας
Φορείς:	ΕΜΠ-ISAAR, ACE HELLAS ΑΕ
Χρόνος Παράδοσης:	Από το πρόγραμμα: M18 Υλοποιήθηκε: M14 (Οκτώβριος 2019)



Εισαγωγή

Στο συγκεκριμένο παραδοτέο γίνεται διερεύνηση αλγορίθμων που εντάσσονται στην υπολογιστική πλατφόρμα βελτιστοποίησης. Προηγείται παρουσίαση του θεωρητικού υποβάθρου, ακολουθεί σύντομη περιγραφή των αλγορίθμων και στην συνέχεια παραδείγματα εφαρμογής (μαθηματικά και μηχανικού), στα οποία αξιολογείται η αποδοτικότητα της υλοποίησης και φαίνεται η επίδοση του καθένα από αυτούς.

Οι αλγόριθμοι χωρίζονται σε εκείνους που αντιμετωπίζουν προβλήματα με μία αντικειμενική συνάρτηση και σε εκείνους με περισσότερες.

1 ΒΕΛΤΙΣΤΟΣ ΣΧΕΔΙΑΣΜΟΣ

1.1 Εισαγωγή

Ο στόχος των μηχανικών είναι η ανάλυση και ο σχεδιασμός συστημάτων με έμφαση σε μηχανολογικές, αεροδιαστημικές, αστικές, βιο-ιατρικές, ηλεκτρολογικές και βιομηχανικές εφαρμογές. Εξαιτίας του μεγάλου ανταγωνισμού, σήμερα, δεν είναι πια αρκετός ένας σχεδιασμός ο οποίος απλά ικανοποιεί τις προδιαγραφές που έχουν τεθεί για το σύστημα το οποίο περιγράφει, αλλά ένας σχεδιασμός ο οποίος είναι ο καλύτερος δυνατός. Με τον όρο «καλύτερος» μπορεί να χαρακτηριστεί ένας σχεδιασμός ο οποίος είναι αποδοτικός, ευπροσάρμοστος, μοναδικός, και οικονομικός, δηλαδή, να χαρακτηρίζεται από χαμηλό κόστος και υψηλή πρακτική αξία.

Για το βέλτιστο σχεδιασμό συστημάτων απαιτούνται αναλυτικές και αριθμητικές μέθοδοι, η υλοποίηση των οποίων σε μορφή λογισμικού δίνει στο μηχανικό ένα πανίσχυρο εργαλείο στο σχεδιασμό συστημάτων. Η διαδικασία που ακολουθείται στο βέλτιστο σχεδιασμό συστημάτων είναι παρόμοια σε όλους τους τομείς της μηχανικής. Οι αναλυτικές και οι αριθμητικές μέθοδοι μπορεί να διαφέρουν μεταξύ τους, και ο ορισμός του προβλήματος μπορεί να χρησιμοποιεί μεγέθη που να αφορούν το συγκεκριμένο τομέα της μηχανικής. Μόλις όμως τεθεί το πρόβλημα στη μαθηματική του διατύπωση, τότε όλα τα προβλήματα από όλους τους τομείς της μηχανικής παρουσιάζουν σημαντικές ιδιότητες.

Οι βάσεις για τις διαδικασίες ανάλυσης και σχεδιασμού πολύπλοκων συστημάτων έχουν τεθεί εδώ και δεκαετίες. Μέχρι πριν από μερικά χρόνια, κύριο μέλημα των μηχανικών στο σχεδιασμό συστημάτων ήταν να παρουσιάσουν ένα σχεδιασμό ο οποίος να ικανοποιεί πλήρως τις απαιτούμενες προδιαγραφές και περιορισμούς. Μόλις πριν από τρεις περίπου δεκαετίες, με την αλματώδη αύξηση της διαθέσιμης επεξεργαστικής ισχύος των υπολογιστικών συστημάτων, έγινε εφικτός ο βέλτιστος σχεδιασμός συστημάτων, που είναι φτηνότερα και πιο αποδοτικά κατά τη λειτουργία τους.

Με τη χρήση διαδικασιών βέλτιστου σχεδιασμού, έγινε εφικτός ο σχεδιασμός κατασκευών οι οποίες είναι οι βέλτιστες δυνατές ως προς κριτήρια όπως το βάρος τους ή και το κόστος τους. Η εύρεση του βέλτιστου σχεδιασμού μιας κατασκευής, επιτυγχάνεται μέσω της διαδικασίας βελτιστοποίησης, κατά την οποία οι μηχανικοί προσπαθούν να βρουν ένα σετ ανεξάρτητων μεταξύ τους μεταβλητών, οι οποίες ονομάζονται παράμετροι ή μεταβλητές σχεδιασμού (design variables), τέτοιο ώστε η αντικειμενική συνάρτηση (fitness function) που περιγράφει το κριτήριο βελτιστοποίησης, να λάβει την ελάχιστη (ή τη μέγιστη) τιμή.

1.2 Μαθηματική διατύπωση

Η διατύπωση ενός προβλήματος βέλτιστου σχεδιασμού, μετατρέπει την περιγραφή του προβλήματος και τις προδιαγραφές του προς βελτιστοποίηση συστήματος, σε μια καλώς ορισμένη μαθηματική πρόταση. Η διατύπωση ξεκινά με την επιλογή των μεταβλητών σχεδιασμού, εκείνων δηλαδή των μεταβλητών οι οποίες περιγράφουν πλήρως ένα σχεδιασμό. Τα περισσότερα συστήματα σχεδιάζονται έτσι ώστε να ικανοποιούν και συγκεκριμένους περιορισμούς, οι οποίοι εκφράζονται μέσω των μεταβλητών σχεδιασμού. Αν ένας σχεδιασμός ικανοποιεί τους περιορισμούς που έχουν τεθεί για το πρόβλημα, τότε ονομάζεται «εφικτός». Είναι αναγκαίο κατά τη διαδικασία βέλτιστου σχεδιασμού, να μπορούν να συγκριθούν μεταξύ τους δύο ή περισσότεροι εφικτοί σχεδιασμοί. Αυτό επιτυγχάνεται με την επιλογή ενός ή περισσότερων κριτηρίων με βάση τα οποία θα κριθούν οι εφικτοί σχεδιασμοί. Το κριτήριο περιγράφεται από μια συνάρτηση των μεταβλητών σχεδιασμού η οποία ονομάζεται αντικειμενική συνάρτηση.

Συνοψίζοντας τα παραπάνω, η μαθηματική διατύπωση ενός προβλήματος βέλτιστου σχεδιασμού, με μια αντικειμενική συνάρτηση και με συνεχείς μεταβλητές σχεδιασμού δίνεται από την παρακάτω μαθηματική πρόταση:

$$\begin{aligned}
& \text{minimize } F \mathbf{x} = F(x_1, x_2, \dots, x_n) \\
& \text{subject to: } h_j \mathbf{x} \equiv h_j(x_1, x_2, \dots, x_n) = 0, \quad j = 1, \dots, p \\
& \quad \quad \quad g_i \mathbf{x} \equiv g_i(x_1, x_2, \dots, x_n) \leq 0, \quad i = 1, \dots, m \\
& \quad \quad \quad x_i^L \leq x_i \leq x_i^U, \quad i = 1, \dots, n
\end{aligned} \tag{1.1}$$

όπου F είναι η αντικειμενική συνάρτηση, \mathbf{x} είναι το διάνυσμα μεταβλητών σχεδιασμού, $h_j \mathbf{x}$ είναι η j συνάρτηση ισοτικών περιορισμών, $g_i \mathbf{x}$ είναι η i συνάρτηση ανισοτικών περιορισμών, και x_i^L, x_i^U είναι το κάτω και άνω όριο της i μεταβλητής σχεδιασμού αντίστοιχα. Η αντικειμενική συνάρτηση F πρέπει να εξαρτάται από όλες τις μεταβλητές σχεδιασμού. Οι συναρτήσεις περιορισμών h και g πρέπει να εξαρτώνται από μια τουλάχιστον μεταβλητή σχεδιασμού, αλλιώς δεν έχουν νόημα για το συγκεκριμένο πρόβλημα.

Ο αριθμός p των συναρτήσεων ισοτικών περιορισμών πρέπει να είναι μικρότερος ή το πολύ ίσος με τον αριθμό των μεταβλητών σχεδιασμού, δηλαδή πρέπει $p \leq n$. Αν $p > n$, τότε το πρόβλημα είναι αόριστο (over-determined). Σε αντίθεση με τους ισοτικούς περιορισμούς, στους ανισοτικούς περιορισμούς δεν υπάρχει όριο στο πλήθος τους. Αν \mathbf{x}^* είναι το βέλτιστο διάνυσμα μεταβλητών σχεδιασμού, τότε ο μέγιστος αριθμός ενεργών περιορισμών, δηλαδή ο μέγιστος αριθμός των συναρτήσεων ανισοτικών περιορισμών για τις οποίες ισχύει $g_i \mathbf{x}^* = 0$ δεν πρέπει να είναι μεγαλύτερος από το πλήθος των μεταβλητών σχεδιασμού n . Θα πρέπει να σημειωθεί εδώ πως κάποια προβλήματα δεν έχουν συναρτήσεις περιορισμών.

Συνοψίζοντας τα παραπάνω, για να εφαρμοστεί μια μέθοδος βελτιστοποίησης σε κάποιο πρόβλημα, πρέπει να προηγηθεί η μοντελοποίησή του, ώστε να αποτελείται από τα παρακάτω τρία βασικά στοιχεία:

1. Τις μεταβλητές σχεδιασμού που ορίζουν τους σχεδιασμούς.
2. Την αντικειμενική συνάρτηση που περιγράφει μαθηματικά το στόχο της βελτιστοποίησης
3. Τους περιορισμούς που διέπουν τις μεταβλητές σχεδιασμού, οι οποίοι καθοδηγούν την έρευνα σε εφικτούς σχεδιασμούς.

1.2.1 Μεταβλητές σχεδιασμού

Τα μεγέθη στα οποία αν δοθούν τιμές, χαρακτηρίζουν πλήρως έναν σχεδιασμό, ονομάζονται μεταβλητές σχεδιασμού. Το διάνυσμα σχεδιασμού, είναι εκείνο το διάνυσμα που έχει ως στοιχεία του τις μεταβλητές σχεδιασμού. Για παράδειγμα, για ένα δικτύωμα του οποίου οι κόμβοι έχουν τοποθετηθεί εξαρχής σε συγκεκριμένες θέσεις, ένας σχεδιασμός μπορεί να οριστεί πλήρως από τις διατομές των ράβδων του δικτύωματος, επομένως αυτές μπορούν να αποτελέσουν τις μεταβλητές σχεδιασμού. Αν οι ράβδοι είναι κυκλικής διατομής με εσωτερική διάμετρο d_{in} , εξωτερική διάμετρο d_{out} και πάχος $t = d_{out} - d_{in} / 2$, οι μεταβλητές σχεδιασμού θα μπορούσαν να είναι η εξωτερική και η εσωτερική διάμετρος, η εσωτερική διάμετρος και το πάχος, η εξωτερική διάμετρος και το πάχος, ή η μέση διάμετρος $d_m = d_{out} + d_{in} / 2$ και το πάχος.

Οι μεταβλητές σχεδιασμού πρέπει να είναι ανεξάρτητες μεταξύ τους, επομένως στο παράδειγμα του δικτύωματος οι μεταβλητές σχεδιασμού δεν θα μπορούσε να είναι η εξωτερική διάμετρος d_{out} , η εσωτερική διάμετρος d_{in} και το πάχος t , γιατί το πάχος συνδέεται άμεσα με την εξωτερική και εσωτερική διάμετρο. Μια μεταβλητή σχεδιασμού μπορεί να αντιπροσωπεύει πολλά μέρη μιας κατασκευής, όπως στην περίπτωση των δικτυωτών φορέων, που οι ράβδοι συνήθως ομαδοποιούνται. Με αυτόν τον τρόπο επιταχύνεται η διαδικασία βελτιστοποίησης και περιορίζεται ο κίνδυνος εμφάνισης προβλημάτων αριθμητικής αστάθειας.

Η επιλογή των μεταβλητών σχεδιασμού στη διατύπωση του προβλήματος είναι πολύ σημαντική, καθώς αν δεν επιλεγούν οι σωστές μεταβλητές, η διατύπωση μπορεί να είναι λανθασμένη ή επίλυση του προβλήματος αδύνατη. Κάποιες φορές είναι δυνατόν να προστεθούν επιπλέον μεταβλητές σχεδιασμού στο πρόβλημα, ώστε η διατύπωσή του να είναι πιο ευέλικτη. Κατά τη διαδικασία της βελτιστοποίησης οι επιπλέον μεταβλητές σχεδιασμού μπορούν να αφαιρεθούν αφού λάβουν συγκεκριμένες τιμές. Θα πρέπει να τονιστεί, πως υπάρχει ένας ελάχιστος αριθμός μεταβλητών σχεδιασμού ώστε η διατύπωση του προβλήματος να είναι σωστή.

1.2.2 Συναρτήσεις περιορισμών

Όπως προαναφέρθηκε, ένας σχεδιασμός χαρακτηρίζεται πλήρως από τις μεταβλητές σχεδιασμού. Ακόμη και αν κάποιες μεταβλητές σχεδιασμού λάβουν τιμές που δεν έχουν νόημα (πχ αρνητικές επιφάνειες), πάλι ο σχεδιασμός ορίζεται πλήρως. Για να μη λάβουν οι μεταβλητές σχεδιασμού μη επιτρεπόμενες γενικά τιμές, χρησιμοποιούνται οι λεγόμενες συναρτήσεις περιορισμών, τις οποίες και θα πρέπει να ικανοποιεί κάθε διάλυση σχεδιασμού, ώστε να θεωρηθεί «εφικτός» ο σχεδιασμός τον οποίο περιγράφει. Αν ένα διάλυση σχεδιασμού δεν ικανοποιεί τις συναρτήσεις περιορισμών, τότε ο σχεδιασμός καλείται «μη εφικτός».

Οι περιορισμοί επιβάλλονται στις μεταβλητές σχεδιασμού είτε άμεσα, είτε έμμεσα. Η άμεση επιβολή ενός περιορισμού σε μια μεταβλητή σχεδιασμού, υλοποιείται με την οριοθέτηση του πεδίου ορισμού της μεταβλητής σχεδιασμού μεταξύ μιας κατώτερης και μιας ανώτερης τιμής. Η έμμεση επιβολή των περιορισμών γίνεται με το συνδυασμό των μεταβλητών σχεδιασμών μέσω μιας συνάρτησης περιορισμών.

Οι συναρτήσεις περιορισμών διακρίνονται σε συναρτήσεις ανισοτικών περιορισμών που είναι της μορφής $g_i \mathbf{x} \equiv g_i x_1, x_2, \dots, x_n \leq 0$ και σε συναρτήσεις ισοτικών περιορισμών της μορφής $h_j \mathbf{x} \equiv h_j x_1, x_2, \dots, x_n = 0$, όπου \mathbf{x} είναι το διάνυσμα των μεταβλητών σχεδιασμού. Οι συναρτήσεις περιορισμών πρέπει να εξαρτώνται από τουλάχιστον μία μεταβλητή σχεδιασμού ώστε ο περιορισμός να έχει νόημα. Ο ανισοτικός περιορισμός i , καλείται «ενεργός» όταν για κάποιο διάνυσμα σχεδιασμού \mathbf{x}^* ισχύει $g_i \mathbf{x}^* = 0$, σε διαφορετική περίπτωση ονομάζεται «ανεργός». Ένας ισοτικός περιορισμός h_j , ονομάζεται «ενεργός» για ένα διάνυσμα σχεδιασμού \mathbf{x}^* αν ισχύει $h_j \mathbf{x}^* = 0$, σε διαφορετική περίπτωση, αν $h_j \mathbf{x}^* \neq 0$ ο περιορισμός παραβιάζεται. Επομένως, κάθε εφικτός σχεδιασμός ορίζεται από ενεργούς ή ανεργούς ανισοτικούς περιορισμούς και από ενεργούς ισοτικούς περιορισμούς. Σε κάθε διαφορετική περίπτωση, ο σχεδιασμός θεωρείται μη εφικτός.

Οι περιορισμοί που συνήθως επιβάλλονται στο βέλτιστο σχεδιασμό των κατασκευών είναι περιορισμοί τάσεων και μετατοπίσεων. Στις περισσότερες περιπτώσεις, οι περιορισμοί αυτοί δεν είναι δυνατόν να εκφραστούν αναλυτικά ως προς τις μεταβλητές σχεδιασμού. Επομένως, κατά τη διαδικασία της βελτιστοποίησης, πραγματοποιείται σε κάθε βήμα ανάλυση πεπερασμένων στοιχείων του κάθε πιθανού σχεδιασμού της κατασκευής, και από τα αποτελέσματα ελέγχεται αν αυτός ικανοποιεί τους περιορισμούς τάσεων και μετατοπίσεων ή όχι.

1.2.3 Αντικειμενική συνάρτηση

Ο όρος «βέλτιστος σχεδιασμός» υπονοεί την ύπαρξη πολλών σχεδιασμών, οι οποίοι δεν έχουν την ίδια πρακτική αξία. Το κριτήριο με βάση το οποίο συγκρίνονται οι σχεδιασμοί εκφράζεται από μια συνάρτηση των μεταβλητών σχεδιασμού, η οποία καλείται αντικειμενική συνάρτηση. Δίχως να βλάπτεται η γενικότητα, θεωρούμε πως σε ένα πρόβλημα βέλτιστου σχεδιασμού, ο στόχος είναι η εύρεση του ολικού ελαχίστου της αντικειμενικής συνάρτησης. Ένα πρόβλημα εύρεσης του μέγιστου της αντικειμενικής συνάρτησης μπορεί να μετατραπεί σε ένα πρόβλημα εύρεσης του ελαχίστου αν αντί να μεγιστοποιηθεί η συνάρτηση $F \mathbf{x}$, ελαχιστοποιηθεί η συνάρτηση $-F \mathbf{x}$. Στη βιβλιογραφία, μια αντικειμενική συνάρτηση που ελαχιστοποιείται, συχνά καλείται και συνάρτηση κόστους.

Η επιλογή της αντικειμενικής συνάρτησης είναι από τις πιο σημαντικές επιλογές σε όλη τη διαδικασία εύρεσης του βέλτιστου σχεδιασμού. Ο καθορισμός της αντικειμενικής συνάρτησης εξαρτάται από το είδος του προς επίλυση προβλήματος, τη μεθοδολογία ή αλγόριθμο βελτιστοποίησης και την απαιτούμενη αριθμητική ακρίβεια των υπολογισμών. Συνήθως, η αντικειμενική συνάρτηση εξισώνεται με κάποια οικονομικά μεγέθη, όπως το κόστος παραγωγής ή το κόστος λειτουργίας ενός συστήματος. Τις περισσότερες φορές, κάτι τέτοιο είναι πολύ δύσκολο ή και αδύνατο να συμβεί, καθώς τα οικονομικά μεγέθη εξαρτώνται από ευμετάβλητες παραμέτρους όπως οι μισθοί, τα επιτόκια, το κόστος των πρώτων υλών, κλπ. Η πλέον συνηθισμένη επιλογή της αντικειμενικής συνάρτησης στο βέλτιστο σχεδιασμό των κατασκευών, είναι η ελαχιστοποίηση της μάζας ή ισοδύναμα του βάρους της κατασκευής.

Σε πολλές περιπτώσεις προβλημάτων μπορεί να υπάρχουν περισσότερες από μια αντικειμενικές συναρτήσεις, οι οποίες συχνά αντιπαλεύονται η μία τις υπόλοιπες. Για παράδειγμα, σε ένα πρόβλημα μπορεί να θέλουμε να ελαχιστοποιήσουμε το βάρος μιας κατασκευής και ταυτόχρονα ελαχιστοποιήσουμε τη μετατόπισή της σε συγκεκριμένα σημεία. Τέτοια προβλήματα

ονομάζονται προβλήματα βέλτιστου σχεδιασμού με πολλές αντικειμενικές συναρτήσεις και μια πληρέστερη περιγραφή τους όπως και των τρόπων αντιμετώπισής τους δίνεται στο τέταρτο κεφάλαιο του παραδοτέου Π1.1.

1.2.4 Χώρος έρευνας

Σε ορισμένα προβλήματα βέλτιστου σχεδιασμού, οι μεταβλητές σχεδίασης έχουν νόημα όταν αυτές ένα πεπερασμένο σύνολο τιμών. Ένα τέτοιο πρόβλημα είναι αυτό της ελαχιστοποίησης του βάρους ενός δικτυωτού φορέα με μεταβλητές σχεδίασης τις διατομές των ράβδων του. Οι διατομές των ράβδων, εξαιτίας της τυποποίησής τους επιβάλλεται να λαμβάνουν τιμές από ένα διακριτό σύνολο τιμών.

Ένα πρόβλημα βέλτιστου σχεδιασμού με διακριτές μεταβλητές σχεδιασμού, διατυπώνεται μαθηματικά όπως ακριβώς και το πρόβλημα με συνεχείς μεταβλητές σχεδιασμού, με μοναδική διαφορά πως οι μεταβλητές σχεδιασμού μπορούν να λαμβάνουν τιμές από ένα πεπερασμένο σύνολο τιμών. Ένα τέτοιο σύνολο τιμών θα μπορούσε να είναι για παράδειγμα, ένα υποσύνολο του n -διάστατου χώρου των πραγματικών δηλαδή $\mathbf{x} \in \mathbb{R}^d \subset \mathbb{R}^n$, ένα υποσύνολο του n -διάστατου χώρου των ακέραιων αριθμών δηλαδή $\mathbf{x} \in \mathbb{Z}^d \subset \mathbb{Z}^n$ ή και ακόμη οι μεταβλητές σχεδιασμού να είναι δυαδικοί αριθμοί, δηλαδή $\mathbf{x} \in B^n$.

Η επίλυση ενός προβλήματος βέλτιστου σχεδιασμού με διακριτές μεταβλητές σχεδιασμού είναι πολύ δυσκολότερη από την επίλυση ενός προβλήματος με συνεχείς μεταβλητές σχεδιασμού. Αρχικά, κάποιος θα μπορούσε να πει πως αφού οι δυνατές λύσεις σε ένα πρόβλημα βελτιστοποίησης με διακριτές μεταβλητές σχεδιασμού είναι πεπερασμένες, άρα και λιγότερες από αυτές ενός προβλήματος με συνεχείς μεταβλητές σχεδιασμού, τότε το πρόβλημα θα είναι και ευκολότερο στη λύση. Η δυσκολία όμως σε διακριτά προβλήματα αυξάνει κατακόρυφα, επειδή ο χώρος λύσης είναι ασύνδετος και μή-κυρτός (non-convex). Επομένως, οι αλγόριθμοι που χρησιμοποιούνται στην περίπτωση της συνεχούς βελτιστοποίησης, δεν είναι δυνατόν να χρησιμοποιηθούν απευθείας σε προβλήματα βελτιστοποίησης που ορίζονται σε διακριτούς χώρους.

Ένα διακριτό πρόβλημα θα μπορούσε να λυθεί δοκιμάζοντας όλες τις δυνατές λύσεις που θα μπορούσαν να προκύψουν. Αυτό είναι δυνατόν να γίνει για διακριτούς χώρους με πολύ λίγα σημεία-λύσεις και για προβλήματα με λίγες μεταβλητές σχεδιασμού. Για παράδειγμα, ένα πρόβλημα με 10 μεταβλητές σχεδιασμού, κάθε μία από τις οποίες μπορεί να πάρει 10 διακριτές τιμές έχει 10^{10} δυνατές λύσεις, οι οποίες απαιτούν τεράστιο υπολογιστικό χρόνο για την εύρεσή τους και τεράστιο αποθηκευτικό χώρο, αφού χρειάζονται κάτι περισσότερο από 100 GB για την αποθήκευσή τους. Αν ληφθεί υπόψιν και το υπολογιστικό κόστος της ανάλυσης πεπερασμένων στοιχείων σε ένα τέτοιο πρόβλημα βέλτιστου σχεδιασμού κατασκευών, η εύρεση του βέλτιστου σχεδιασμού κρίνεται σχεδόν ακατόρθωτη ακόμη και με τη σημερινή υπολογιστική δύναμη.

Εκτός από τα προβλήματα με συνεχείς ή διακριτές μεταβλητές σχεδιασμού, υπάρχουν και προβλήματα βέλτιστου σχεδιασμού στα οποία κάποιες από τις μεταβλητές σχεδιασμού λαμβάνουν συνεχείς τιμές, ενώ οι υπόλοιπες λαμβάνουν διακριτές τιμές. Τα προβλήματα αυτά ονομάζονται μικτά προβλήματα, και για την αντιμετώπισή τους έχουν αναπτυχθεί αρκετές μέθοδοι, οι οποίες οδηγούν σε μια «καλή» λύση, αλλά όχι στη βέλτιστη δυνατή.

1.3 Βέλτιστος σχεδιασμός των κατασκευών

Ο βέλτιστος σχεδιασμός των κατασκευών χωρίζεται στις παρακάτω τρεις μεγάλες κατηγορίες:

- Στο βέλτιστο σχεδιασμό της τοπολογίας.
- Στο βέλτιστο σχεδιασμό του σχήματος.
- Στο βέλτιστο σχεδιασμό των διατομών.

Με το βέλτιστο σχεδιασμό των διατομών επιδιώκεται συνήθως η ελαχιστοποίηση του βάρους των κατασκευών, με περιορισμούς τάσεων και μετατοπίσεων στα δομικά τους μέλη. Η βελτιστοποίηση σχήματος έχει ως στόχο την ελαχιστοποίηση του όγκου μιας τρισδιάστατης κατασκευής, ή της επιφάνειάς της αν αυτή έχει μοντελοποιηθεί στις δύο διαστάσεις, υπό περιορισμούς. Η βελτιστοποίηση τοπολογίας έχει ως στόχο τη βελτιστοποίηση της συνδεσμολογίας της κατασκευής ώστε αυτή να γίνει οικονομικότερη. Στις επόμενες

παραγράφους δίνεται μια σύντομη περιγραφή των κατηγοριών βέλτιστου σχεδιασμού των κατασκευών.

1.3.1 Βέλτιστος σχεδιασμός τοπολογίας

Ο καθορισμός της τοπολογίας μιας κατασκευής είναι πολύ σημαντικός ώστε αυτή να είναι οικονομική και να παραλαμβάνει αποτελεσματικά τα φορτία. Η τοπολογία συνήθως καθορίζεται από τις προδιαγραφές της κατασκευής. Αφού οριστούν τα φορτία της κατασκευής και ο τρόπος στήριξής της, και με τη χρήση κατάλληλων αλγορίθμων, ακολουθείται μια επαναληπτική διαδικασία η οποία οδηγεί σε μια βέλτιστη κατανομή του υλικού της κατασκευής. Αφού επιτευχθεί η βελτιστοποίηση της τοπολογίας μιας κατασκευής, στη συνέχεια βελτιστοποιείται το σχήμα της με κατάλληλους αλγόριθμους ώστε να ολοκληρωθεί η εύρεση της βέλτιστης δομής της κατασκευής.

1.3.2 Βέλτιστος σχεδιασμός σχήματος

Στο βέλτιστο σχεδιασμό του σχήματος των κατασκευών μιας συγκεκριμένης τοπολογίας, ο βασικός στόχος είναι ο καθορισμός του σχήματος των διατομών ράβδων ή ολόσομων κατασκευών ώστε να είναι πιο οικονομική και πιο αποδοτική η χρήση τους. Κατά τον βέλτιστο σχεδιασμό του σχήματος των κατασκευών, καθορίζεται στην αρχή της διαδικασίας το αρχικό σχήμα των κατασκευών. Στη συνέχεια, δημιουργείται το πλέγμα των πεπερασμένων στοιχείων, και ολοκληρώνεται η ανάλυση, προκειμένου να εκτιμηθούν κρίσιμα μεγέθη όπως το τασικό, το παραμορφωσιακό και το θερμοκρασιακό πεδίο. Στη συνέχεια, με κάποιον αιτιοκρατικό ή στοχαστικό αλγόριθμο βελτιστοποίησης, δημιουργείται ένας νέος σχεδιασμός, δημιουργείται πάλι ένα νέο πλέγμα πεπερασμένων στοιχείων, και η διαδικασία συνεχίζεται όπως πριν.

Κατά τη διαδικασία βελτιστοποίησης του σχήματος των κατασκευών, επειδή το σχήμα τους μεταβάλλεται συνεχώς, απαιτείται αναπροσαρμογή του πλέγματος των πεπερασμένων στοιχείων ώστε να βελτιωθεί η ακρίβεια της ανάλυσης. Η αναπροσαρμογή του πλέγματος γίνεται με πυκνωση τύπου-h είτε με την δημιουργία από την αρχή ενός πλέγματος πυκνότερου στα σημεία συγκέντρωσης τάσεων, είτε με αυξομείωση του μεγέθους των στοιχείων του πλέγματος και μετατόπιση των κόμβων του, είτε με αναπροσαρμογή όλου του πλέγματος.

1.3.3 Βελτιστοποίηση των μεγεθών των διατομών

Ο βέλτιστος σχεδιασμός των μεγεθών των διατομών, αφορά κυρίως διατομές χωρικών πλαισίων και δικτυωτών κατασκευών. Αρχικά, ορίζεται το μοντέλο της κατασκευής δηλαδή η γεωμετρία της, τα ασκούμενα φορτία και το δομοστατικό προσομοίωμά της. Στη συνέχεια, διατυπώνεται το πρόβλημα της βελτιστοποίησης, με κατάλληλη επιλογή μεταβλητών σχεδιασμού, της αντικειμενικής συνάρτησης και των περιορισμών. Έπειτα, γίνεται η ανάλυση του σχεδιασμού της κατασκευής με τη μέθοδο των πεπερασμένων στοιχείων και υπολογίζονται οι τάσεις και οι μετατοπίσεις του φορέα. Από τα αποτελέσματα της ανάλυσης, ελέγχεται αν παραβιάζονται ή όχι οι περιορισμοί. Αν ως αλγόριθμος βελτιστοποίησης χρησιμοποιηθεί κάποια αιτιοκρατική μαθηματική μέθοδος, απαιτείται στη συνέχεια ανάλυση της ευαισθησίας του σχεδιασμού. Τέλος, ελέγχονται τα κριτήρια τερματισμού του αλγόριθμου, και αν ικανοποιούνται, η διαδικασία τερματίζεται, διαφορετικά ο αλγόριθμος βελτιστοποίησης επιλέγει νέους πιθανούς βέλτιστους σχεδιασμούς και η διαδικασία συνεχίζεται.

2 ΒΕΛΤΙΣΤΟΣ ΣΧΕΔΙΑΣΜΟΣ ΜΕ ΕΝΑ ΚΡΙΤΗΡΙΟ

2.1 Εισαγωγή

Οι κατηγορίες προβλημάτων βελτιστοποίησης, για ένα πρόβλημα βέλτιστου σχεδιασμού ενός κριτηρίου που διατυπώνεται από τη σχέση (1.1) του πρώτου κεφαλαίου είναι πολλές. Για παράδειγμα, ένα πρόβλημα μπορεί να κατηγοριοποιηθεί ανάλογα με τη φύση της αντικειμενικής συνάρτησης και των περιορισμών (γραμμική, μή-γραμμική, κυρτή), τον αριθμό των μεταβλητών σχεδιασμού (μικρός ή μεγάλος), την ομαλότητα της αντικειμενικής συνάρτησης (ομαλή ή μή-ομαλή), κλπ. Ο βασικότερος ίσως διαχωρισμός των προβλημάτων βελτιστοποίησης, είναι μεταξύ *προβλημάτων χωρίς περιορισμούς*, και *προβλημάτων με περιορισμούς*.

Τα προβλήματα χωρίς περιορισμούς εμφανίζονται σε πολλές πρακτικές εφαρμογές. Τις περισσότερες φορές δημιουργούνται από προβλήματα με περιορισμούς, τα οποία μετασχηματίζονται με κάποια μεθοδολογία σε προβλήματα χωρίς περιορισμούς. Τα προβλήματα με περιορισμούς προκύπτουν από τη μοντελοποίηση προβλημάτων η οποία συμπεριλαμβάνει συγκεκριμένους περιορισμούς των μεταβλητών σχεδιασμού. Οι περιορισμοί μπορεί να είναι απλά όρια στις μεταβλητές σχεδιασμού (πχ, $0 \leq x_3 \leq 7$), πιο γενικοί γραμμικοί περιορισμοί όπως:

$$\sum_i x_i \leq 10$$

ή μή-γραμμικοί ανισοτικοί περιορισμοί που αντιπροσωπεύουν πολύπλοκες σχέσεις ανάμεσα στις μεταβλητές σχεδιασμού.

Όταν η αντικειμενική συνάρτηση και οι συναρτήσεις των περιορισμών είναι γραμμικές συναρτήσεις των μεταβλητών σχεδιασμού x_i , το πρόβλημα καλείται *πρόβλημα γραμμικού προγραμματισμού*. Τέτοια προβλήματα συναντώνται κυρίως στις οικονομικές επιστήμες. Τα προβλήματα στα οποία η αντικειμενική συνάρτηση ή κάποιες από τις συναρτήσεις περιορισμών είναι μή-γραμμικές συναρτήσεις των μεταβλητών σχεδιασμού, ονομάζονται *προβλήματα μή-γραμμικού προγραμματισμού*. Το πρόβλημα του βέλτιστου σχεδιασμού των κατασκευών, είναι ένα πρόβλημα μή-γραμμικού προγραμματισμού.

2.2 Προβλήματα βελτιστοποίησης χωρίς περιορισμούς

Στα προβλήματα βελτιστοποίησης χωρίς περιορισμούς, πρέπει να ελαχιστοποιηθεί μια αντικειμενική συνάρτηση F , που είναι συνάρτηση πραγματικών μεταβλητών σχεδιασμού. Η μαθηματική διατύπωση ενός προβλήματος βελτιστοποίησης χωρίς περιορισμούς, δίνεται από τη σχέση:

$$\min F \mathbf{x} \quad (2.1)$$

όπου $\mathbf{x} \in \mathbb{R}^n$ είναι το διάνυσμα των μεταβλητών σχεδιασμού και $F: \mathbb{R}^n \rightarrow \mathbb{R}$ είναι η αντικειμενική συνάρτηση.

Ο στόχος σε ένα πρόβλημα βελτιστοποίησης, είναι η εύρεση του ολικού ελαχίστου της αντικειμενικής συνάρτησης. Ένας σχεδιασμός \mathbf{x}^* αποτελεί *ολικό ελάχιστο* της αντικειμενικής συνάρτησης F , όταν ισχύει η σχέση:

$$F \mathbf{x}^* \leq F \mathbf{x}, \quad \forall \mathbf{x} \in \mathbb{R}^n \quad (2.2)$$

Η εύρεση του ολικού ελαχίστου είναι συνήθως πολύ δύσκολη. Οι περισσότεροι αλγόριθμοι βρίσκουν συνήθως ένα τοπικό ελάχιστο της αντικειμενικής συνάρτησης. Ένας σχεδιασμός \mathbf{x}^* αποτελεί *τοπικό ελάχιστο* της αντικειμενικής συνάρτησης F όταν ισχύει η σχέση:

$$F \mathbf{x}^* \leq F \mathbf{x}, \quad \forall \mathbf{x} \in N \quad (2.3)$$

όπου N είναι μια περιοχή γύρω από το σχεδιασμό \mathbf{x}^* και η οποία περιέχει το σχεδιασμό \mathbf{x}^* .

Για μια ομαλή και δύο φορές παραγωγίσιμη αντικειμενική συνάρτηση F , είναι δυνατόν να εξαχθεί ένα συμπέρασμα για τον αν ο σχεδιασμός \mathbf{x}^* είναι τοπικό ελάχιστο από την κλίση $\nabla F \mathbf{x}^*$ και την εσσιανή $\nabla^2 F \mathbf{x}^*$. Ισχύουν τα παρακάτω θεωρήματα:

Θεώρημα 1: Έστω F μια παραγωγίσιμη αντικειμενική συνάρτηση στην περιοχή του σχεδιασμού \mathbf{x}^* η οποία συμπεριλαμβάνει το σχεδιασμό \mathbf{x}^* . Αν ο σχεδιασμός \mathbf{x}^* είναι τοπικό ελάχιστο της συνάρτησης F , τότε ισχύει:

$$\nabla F \mathbf{x}^* = 0 \quad (2.4)$$

Θεώρημα 2: Αν ο σχεδιασμός \mathbf{x}^* είναι τοπικό ελάχιστο της αντικειμενικής συνάρτησης F και οι δεύτερες παράγωγοι της F είναι συνεχείς στην περιοχή του σχεδιασμού \mathbf{x}^* , τότε $\nabla F \mathbf{x}^* = 0$ και $\nabla^2 F \mathbf{x}^*$ θετικά ημιορισμένο μητρώο.

Θεώρημα 3: Αν η αντικειμενική συνάρτηση F έχει συνεχείς δεύτερες παραγώγους, $\nabla F \mathbf{x}^* = 0$ και $\nabla^2 F \mathbf{x}^*$ θετικά ορισμένο μητρώο, τότε ο σχεδιασμός \mathbf{x}^* αποτελεί τοπικό ελάχιστο της συνάρτησης F .

Θεώρημα 4: Αν η αντικειμενική συνάρτηση F είναι κυρτή, τότε κάθε σχεδιασμός \mathbf{x}^* που είναι τοπικό ελάχιστο της F , είναι και ολικό ελάχιστο της αντικειμενικής συνάρτησης F .

Οι περισσότεροι αλγόριθμοι βελτιστοποίησης αναζητούν συνήθως ένα τοπικό ελάχιστο της αντικειμενικής συνάρτησης, επομένως δεν βρίσκουν πάντα τον καλύτερο σχεδιασμό, δηλαδή το ολικό ελάχιστο της αντικειμενικής συνάρτησης. Τα ολικά ελάχιστα, αν και επιθυμητά στα περισσότερα προβλήματα, είναι πολύ δύσκολο να βρεθούν. Μια ειδική περίπτωση, είναι οι κυρτές αντικειμενικές συναρτήσεις για τις οποίες κάθε τοπικό ελάχιστο είναι ταυτόχρονα και ολικό ελάχιστο.

Μια συνάρτηση ονομάζεται *κυρτή* αν για κάθε δύο σημεία του πεδίου ορισμού της, η γραφική της παράσταση βρίσκεται κάτω από την ευθεία που ενώνει αυτά τα δύο σημεία, δηλαδή:

$$F(a\mathbf{x} + (1-a)\mathbf{y}) \leq aF\mathbf{x} + (1-a)F\mathbf{y} \quad \forall a \in [0,1] \quad (2.5)$$

όπου \mathbf{x}, \mathbf{y} είναι δύο σημεία-σχεδιασμοί και a παράμετρος που μπορεί να πάρει τιμές στο διάστημα $[0,1]$. Αν η αντικειμενική συνάρτηση είναι κυρτή, ο αλγόριθμος βελτιστοποίησης θα συγκλίνει πάντα σε ολικό βέλτιστο.

Στα προβλήματα του γραμμικού προγραμματισμού η αντικειμενική συνάρτηση είναι κυρτή. Γενικά όμως, τα μή-γραμμικά προβλήματα, όπως αυτό του βέλτιστου σχεδιασμού των κατασκευών, είναι μή-κυρτά και συνήθως έχουν πολλά τοπικά ελάχιστα τα οποία δεν είναι ολικά ελάχιστα.

2.3 Κλασσικές μέθοδοι αντιμετώπισης προβλημάτων χωρίς περιορισμούς

Οι πλέον γνωστές μέθοδοι επίλυσης προβλημάτων χωρίς περιορισμούς, είναι οι μέθοδοι της μέγιστης καθόδου, των συζυγών κλίσεων και οι μέθοδοι Newton και quasi-Newton. Οι αλγόριθμοι αυτοί ακολουθούν ένα επαναληπτικό σχήμα της μορφής:

$$\mathbf{x}^g = \mathbf{x}^{g-1} + a^g \mathbf{s}^g \quad (2.6)$$

όπου a είναι μια παράμετρος που ονομάζεται μήκος βήματος, \mathbf{s} είναι η διεύθυνση στην οποία γίνεται η έρευνα για το βέλτιστο στην επανάληψη g και $\mathbf{x}^{g-1}, \mathbf{x}^g$ είναι η προσέγγιση της βέλτιστου σχεδιασμού κατά την επανάληψη $g-1$ και g αντίστοιχα.

Καθένας από αυτούς τους αλγόριθμους καθορίζει με κάποια μεθοδολογία το μήκος βήματος a και τη διεύθυνση έρευνας \mathbf{s} . Για να γίνει αυτό, πρέπει προηγουμένως να υπολογιστεί ή να εκτιμηθεί το διάνυσμα κλίσης της αντικειμενικής συνάρτησης $\nabla F \mathbf{x}$ και το εσσιανό μητρώο $\nabla^2 F \mathbf{x}$. Το διάνυσμα κλίσης για έναν σχεδιασμό \mathbf{x} , δίνεται από τη σχέση:

$$\nabla F \mathbf{x} = \left[\frac{\partial F \mathbf{x}}{\partial x_1}, \frac{\partial F \mathbf{x}}{\partial x_2}, \dots, \frac{\partial F \mathbf{x}}{\partial x_n} \right] \quad (2.7)$$

όπου n ο αριθμός των μεταβλητών σχεδιασμού. Το εσσιανό (Hessian) μητρώο της αντικειμενικής συνάρτησης για το σχεδιασμό \mathbf{x} , δίνεται από τη σχέση:

$$\mathbf{H}_{\mathbf{x}} = \nabla^2 F_{\mathbf{x}} = \begin{pmatrix} \frac{\partial^2 F}{\partial x_1^2} & \frac{\partial^2 F}{\partial x_1 \partial x_2} & \cdots & \frac{\partial^2 F}{\partial x_1 \partial x_n} \\ \frac{\partial^2 F}{\partial x_1 \partial x_2} & \ddots & & \vdots \\ \vdots & & \ddots & \vdots \\ \frac{\partial^2 F}{\partial x_1 \partial x_n} & \cdots & \cdots & \frac{\partial^2 F}{\partial x_n^2} \end{pmatrix} \quad (2.8)$$

2.3.1 Η μέθοδος της μέγιστης καθόδου (Steepest Descent)

Η μέθοδος της μέγιστης καθόδου ανήκει στις μεθόδους πρώτης τάξης, αφού χρησιμοποιεί ως πληροφορία το διάνυσμα κλίσης της αντικειμενικής συνάρτησης. Η διεύθυνση \mathbf{s}^g της έρευνας (βλ. σχέση (2.6)) υπολογίζεται από τη σχέση:

$$\mathbf{s}^g = -\nabla F_{\mathbf{x}^{g-1}} \quad (2.9)$$

Η ροή του αλγόριθμου απεικονίζεται συνοπτικά στο Σχήμα 2.1.

Algorithm: Steepest Descent

- 1 Choose initial \mathbf{x}^0
 - 2 **repeat**
 - 3 $g = g + 1$
 - 3 $\mathbf{s}^g = -\nabla F_{\mathbf{x}^{g-1}}$
 - 4 **choose** a **to minimize** $F_{\mathbf{x}^{g-1}} + a\mathbf{s}^g$
 - 5 $\mathbf{x}^g = \mathbf{x}^{g-1} + a^g \mathbf{s}^g$
 - 6 **until** termination_criterion
-

Σχήμα 2.1: Ο αλγόριθμος Steepest Descent

Η παράμετρος a συνήθως υπολογίζεται με κάποια μέθοδο έρευνας γραμμής (line search), όπως η μέθοδος της χρυσής τομής (golden section).

Ο αλγόριθμος της μέγιστης καθόδου δεν χρησιμοποιεί πληροφορία προηγούμενων επαναλήψεων και γενικά συγκλίνει πολύ αργά στην τελική λύση.

2.3.2 Η μέθοδος των συζυγών κλίσεων (Conjugate Gradient)

Η μέθοδος των συζυγών κλίσεων ανήκει στις μεθόδους πρώτης τάξης. Η διεύθυνση \mathbf{s}^g της έρευνας (βλ. σχέση (2.6)) υπολογίζεται από τη σχέση:

$$\mathbf{s}^g = \begin{cases} -\nabla F_{\mathbf{x}^0}, & \text{if } g=1 \\ -\nabla F_{\mathbf{x}^{g-1}} + \beta^g \mathbf{s}^{g-1}, & \text{if } g \neq 1 \end{cases} \quad (2.10)$$

όπου β^g παράμετρος. Η βασική διαφορά της μεθόδου των συζυγών κλίσεων από τη μέθοδο της μέγιστης καθόδου, είναι πως στη μέθοδο των συζυγών κλίσεων οι διευθύνσεις s είναι \mathbf{H} -συζυγείς, όπου \mathbf{H} είναι το εσσιανό μητρώο. Δηλαδή ισχύει:

$$\mathbf{s}_j^T \mathbf{H} \mathbf{s}_i = 0 \quad (2.11)$$

για δύο οποιεσδήποτε διευθύνσεις $\mathbf{s}_i, \mathbf{s}_j$. Η παράμετρος β^g υπολογίζεται από τη σχέση:

$$\beta^g = \frac{|\nabla F \mathbf{x}^{g-1}|^2}{|\nabla F \mathbf{x}^g|^2} \quad (2.12)$$

Ο αλγόριθμος των συζυγών κλίσεων συγκλίνει σε ακριβώς n επαναλήψεις, αν η αντικειμενική συνάρτηση είναι τετραγωνικής μορφής. Σε διαφορετική περίπτωση η μέθοδος συγκλίνει σε «περίπου» n επαναλήψεις.

2.3.3 Οι μέθοδοι Newton και Quasi-Newton

Η μέθοδος Newton ανήκει στις μεθόδους δεύτερης τάξης, οι οποίες κάνουν χρήση του εσσιανού μητρώου \mathbf{H} . Η διεύθυνση s της έρευνας (βλ. σχέση (2.6)) υπολογίζεται από τη σχέση:

$$\mathbf{s} = -\mathbf{H} \mathbf{x}^{-1} \nabla F \mathbf{x} \quad (2.13)$$

όπου \mathbf{H} είναι ο Hessian πίνακας της αντικειμενικής συνάρτησης.

Αν η αντικειμενική συνάρτηση είναι τετραγωνικής μορφής, τότε η μέθοδος Newton συγκλίνει σε μια επανάληψη. Δυστυχώς απαιτείται ο υπολογισμός του Hessian πίνακα, κάτι που είναι χρονοβόρο για πολλές μεταβλητές σχεδιασμού. Επίσης, αν κάποια μεταβλητή σχεδιασμού είναι γραμμικά εξαρτημένη από κάποια άλλη, τότε ενδέχεται να μην υπάρχει ο αντίστροφος του Hessian πίνακα.

Αντίθετα με τη μέθοδο Newton, στις μεθόδους quasi-Newton δεν είναι αναγκαίο να υπολογίζεται ο Hessian πίνακας σε κάθε επανάληψη. Η διεύθυνση \mathbf{s}^g της έρευνας (βλ. σχέση (2.6)) υπολογίζεται από τη σχέση:

$$\mathbf{s}^g = -\mathbf{A}^g \nabla F \mathbf{x}^{g-1} \quad (2.14)$$

όπου \mathbf{A}^g είναι ένας $n \times n$ πίνακας συμμετρικός και θετικά ορισμένος, που προσεγγίζει τον αντίστροφο του Hessian πίνακα \mathbf{H}^{-1} . Ο πίνακας \mathbf{A}^g αρχικά τίθεται ίσος με τον μοναδιαίο πίνακα \mathbf{I} , επομένως η πρώτη διεύθυνση υπολογίζεται όπως στη μέθοδο της μέγιστης καθόδου. Στις επόμενες επαναλήψεις υπολογίζεται από τη σχέση:

$$\mathbf{A}^{g+1} = \mathbf{A}^g + \mathbf{D}^g \quad (2.15)$$

όπου \mathbf{D}^g είναι ένας συμμετρικός $n \times n$ πίνακας που δίνεται από μια σχέση της μορφής:

$$\mathbf{D}^g = function \mathbf{x}^g - \mathbf{x}^{g-1}, \nabla F \mathbf{x}^g - \nabla F \mathbf{x}^{g-1}, \mathbf{A}^g \quad (2.16)$$

Υπάρχουν διάφορες μέθοδοι υπολογισμού του πίνακα \mathbf{D}^g . Η πιο γνωστές από αυτές είναι η μέθοδος DFP (Davidon-Fletcher-Powell) και η BFGS (Broydon-Fletcher-Goldfarb-Shanno) η οποία θεωρείται και η καλύτερη.

2.4 Προβλήματα βελτιστοποίησης με περιορισμούς

Η μαθηματική διατύπωση ενός προβλήματος βελτιστοποίησης με ανισοτικούς και ισοτικούς περιορισμούς, δίνεται από τη σχέση:

$$\begin{aligned}
& \text{minimize } F \mathbf{x} = F(x_1, x_2, \dots, x_n) \\
& \text{subject to: } \mathbf{h}_j \mathbf{x} \equiv \mathbf{h}_j(x_1, x_2, \dots, x_n) = 0, \quad j = 1, \dots, p \\
& \mathbf{g}_i \mathbf{x} \equiv \mathbf{g}_i(x_1, x_2, \dots, x_n) \leq 0, \quad i = 1, \dots, m \\
& x_i^L \leq x_i \leq x_i^U, \quad i = 1, \dots, n
\end{aligned} \tag{2.17}$$

όπου F είναι η αντικειμενική συνάρτηση, \mathbf{x} είναι το διάνυσμα μεταβλητών σχεδιασμού, $\mathbf{h}_j \mathbf{x}$ είναι η j συνάρτηση ισοτικών περιορισμών, $\mathbf{g}_i \mathbf{x}$ είναι η i συνάρτηση ανισοτικών περιορισμών, και x_i^L, x_i^U είναι το κάτω και άνω όριο της i μεταβλητής σχεδιασμού αντίστοιχα.

Ορίζεται επίσης ο χώρος S των εφικτών σχεδιασμών, τέτοιος ώστε:

$$S = \mathbf{x} \in \mathbb{R}^n \mid \mathbf{g} \mathbf{x} \leq 0, \mathbf{h} \mathbf{x} = 0, \mathbf{x}^L \leq \mathbf{x} \leq \mathbf{x}^U \tag{2.18}$$

Επομένως, το πρόβλημα με περιορισμούς μπορεί να διατυπωθεί ως εξής:

$$\min F \mathbf{x} \quad \mathbf{x} \in S \tag{2.19}$$

Ο χώρος S αποτελεί την εφικτή περιοχή του προβλήματος, στην οποία όλοι οι περιορισμοί ικανοποιούνται. Σε πολλά προβλήματα, ο βέλτιστος σχεδιασμός βρίσκεται στο όριο μεταξύ της περιοχής των εφικτών σχεδιασμών, και της μή-επιτρεπόμενης περιοχής, στην οποία τουλάχιστον ένας περιορισμός παραβιάζεται.

2.5 Κλασσικές μέθοδοι αντιμετώπισης προβλημάτων με περιορισμούς

Οι κλασσικές μεθοδολογίες επίλυσης ενός προβλήματος βελτιστοποίησης με περιορισμούς, χωρίζονται στις λεγόμενες «άμεσες» (direct) μεθόδους, οι οποίες προσπαθούν να επιλύσουν το πρόβλημα κινούμενες μεταξύ των ορίων που θέτουν οι περιορισμοί, και στις «έμμεσες» (indirect) μεθόδους, οι οποίες μετατρέπουν το πρόβλημα με περιορισμούς σε ένα πρόβλημα χωρίς περιορισμούς.

Το πρόβλημα βελτιστοποίησης με περιορισμούς, όπως διατυπώνεται από τη σχέση (2.17) μπορεί να μετατραπεί σε ένα πρόβλημα βελτιστοποίησης χωρίς περιορισμούς, από τη σχέση:

$$L \mathbf{x}, \lambda = F \mathbf{x} + \sum_{j=1}^p \lambda_j \mathbf{h}_j \mathbf{x} + \sum_{i=1}^m \lambda_i \mathbf{g}_i \mathbf{x} \tag{2.20}$$

όπου $L \mathbf{x}, \lambda$ είναι η λαγκραζιανή της F , και λ_i, λ_j είναι οι συντελεστές Lagrange.

Μία αναγκαία συνθήκη αλλά όχι ικανή, για να είναι ένας σχεδιασμός \mathbf{x}^* ελάχιστο (τοπικό ή ολικό), είναι:

$$\nabla L \mathbf{x}^*, \lambda = \nabla F \mathbf{x}^* + \sum_{j=1}^p \lambda_j \nabla \mathbf{h}_j \mathbf{x}^* + \sum_{i=1}^m \lambda_i \nabla \mathbf{g}_i \mathbf{x}^* = 0 \tag{2.21}$$

με $\lambda_j \in \mathbb{R}_+$ και $\lambda_i \in \mathbb{R}$.

Αν ο σχεδιασμός \mathbf{x}^* είναι τοπικό βέλτιστο της αντικειμενικής συνάρτησης, τότε ισχύουν οι παρακάτω συνθήκες:

1. Ο σχεδιασμός \mathbf{x}^* είναι εφικτός.
2. $\lambda_i \nabla \mathbf{g}_i \mathbf{x}^* = 0, i = 1, \dots, m$ και $\lambda_i \geq 0$.
3. $\nabla F \mathbf{x}^* + \sum_{j=1}^p \lambda_j \nabla \mathbf{h}_j \mathbf{x}^* + \sum_{i=1}^m \lambda_i \nabla \mathbf{g}_i \mathbf{x}^* = 0$ με $\lambda_i \geq 0$.

Οι τιμές των συντελεστών Lagrange λ_j μπορούν να πάρουν οποιοσδήποτε τιμές.

Η πρώτη συνθήκη, σημαίνει πως η βέλτιστη λύση ικανοποιεί τους περιορισμούς. Η δεύτερη συνθήκη σημαίνει πως αν ένας περιορισμός δεν ικανοποιείται πλήρως, τότε ο αντίστοιχος συντελεστής Lagrange είναι μηδενικός. Τέλος, η τρίτη συνθήκη σημαίνει πως το διάνυσμα κλίσης της λανγκραζιανής είναι μηδενικό στο βέλτιστο σημείο.

Οι παραπάνω συνθήκες ονομάζονται συνθήκες *Kuhn-Tucker*, και είναι αναγκαίες συνθήκες για βέλτιστο. Στην περίπτωση που ο εφικτός χώρος που ορίζεται από την αντικειμενική συνάρτηση και τους περιορισμούς, είναι κυρτός, οι συνθήκες *Kuhn-Tucker* είναι και ικανές. Αν και τα περισσότερα προβλήματα βελτιστοποίησης δεν είναι κυρτά, η θεωρία των κυρτών προβλημάτων είναι σημαντική καθώς πολλές φορές τα μή-κυρτά προβλήματα προσεγγίζονται από μια σειρά κυρτών προβλημάτων.

2.5.1 Η μέθοδος των συναρτήσεων ποινής (Penalty Functions)

Οι μέθοδοι των συναρτήσεων ποινής διακρίνονται στις μεθόδους των εξωτερικών και των εσωτερικών συναρτήσεων ποινής. Στις μεθόδους των εξωτερικών συναρτήσεων ποινής, το πρόβλημα βελτιστοποίησης με περιορισμούς μπορεί να διατυπωθεί από τη σχέση:

$$\min R \mathbf{x} , \quad R \mathbf{x} = F \mathbf{x} + \sum_i a_i G |g_i \mathbf{x}|^- + \sum_j b_j H |h_j \mathbf{x}| \quad (2.22)$$

όπου $R \mathbf{x}$ είναι η σύνθετη αντικειμενική συνάρτηση, G, H είναι συναρτήσεις των ανισοτικών και ισοτικών περιορισμών και a_i, b_j πραγματικοί αριθμοί με $a_i, b_j \geq 0$. Για τη συνάρτηση $|g_i \mathbf{x}|^-$, ισχύει:

$$|g_i \mathbf{x}|^- = \begin{cases} |g_i \mathbf{x}|, & \text{if } g_i \mathbf{x} < 0 \\ 0, & \text{otherwise} \end{cases}$$

Η σχέση (2.22) αντικαθιστά την αντικειμενική συνάρτηση και τους περιορισμούς σε όλο το πεδίο των τιμών (δυνατή και μή-δυνατή περιοχή). Οι διαμόρφωση της μή-δυνατής περιοχής εξαρτάται από τις παραμέτρους a_i, b_j και από τις συναρτήσεις G, H . Για γραμμικές συναρτήσεις G, H , η σύνθετη αντικειμενική συνάρτηση δίνεται από τη σχέση:

$$R \mathbf{x} = F \mathbf{x} + \sum_i a_i |g_i \mathbf{x}|^- + \sum_j b_j |h_j \mathbf{x}| \quad (2.23)$$

Για τετραγωνικές συναρτήσεις G, H , η σύνθετη συνάρτηση δίνεται από τη σχέση:

$$R \mathbf{x} = F \mathbf{x} + \sum_i a_i |g_i \mathbf{x}|^{-2} + \sum_j b_j |h_j \mathbf{x}|^2 \quad (2.24)$$

Η σχέση (2.24) μπορεί να γραφεί και ως:

$$R \mathbf{x} = F \mathbf{x} + r \left(\sum_i |g_i \mathbf{x}|^{-2} + \sum_j |h_j \mathbf{x}|^2 \right) \quad (2.25)$$

όπου οι παράμετροι a_i, b_j έχουν αντικατασταθεί από μια παράμετρο r . Το ελάχιστο της σύνθετης αντικειμενικής συνάρτησης εξαρτάται από την παράμετρο r και αυτό είναι ένα μειονέκτημα των μεθόδων των συναρτήσεων ποινής.

Στην περίπτωση που το πρόβλημα βελτιστοποίησης διέπεται από ανισοτικές συναρτήσεις περιορισμών, τότε μια σύνθετη συνάρτηση που μετατρέπει το πρόβλημα σε ένα πρόβλημα χωρίς περιορισμούς, μπορεί να διατυπωθεί από τις σχέσεις:

$$\begin{aligned}
R \mathbf{x} &= F \mathbf{x} + \sum_i a_i \frac{1}{g_i \mathbf{x}} \\
R \mathbf{x} &= F \mathbf{x} + r \sum_i \frac{1}{g_i \mathbf{x}} \\
R \mathbf{x} &= F \mathbf{x} + \sum_i \log \left(\frac{1}{g_i \mathbf{x}} \right)
\end{aligned} \tag{2.26}$$

Η σύνθετη συνάρτηση $R \mathbf{x}$ όπως ορίζεται από την παραπάνω σχέση, ορίζεται μόνο στην εφικτή περιοχή του προβλήματος. Για την εφαρμογή των εσωτερικών συναρτήσεων ποινής, πρέπει να είναι γνωστή αρχικά μια εφικτή λύση, διαφορετικά οι σύνθετες συναρτήσεις όπως διατυπώνονται από την σχέση (2.26), απειρίζονται.

Οι εσωτερικές και εξωτερικές συναρτήσεις ποινής μπορούν να συνδυαστούν και να προκύψει μια σύνθετη αντικειμενική συνάρτηση της μορφής:

$$R \mathbf{x} = F \mathbf{x} + r \sum_i \frac{1}{g_i \mathbf{x}} + \frac{1}{\sqrt{r}} \sum_j |h_j \mathbf{x}|^2 \tag{2.27}$$

Η απόδοση της μεθόδου με συνδυασμό εξωτερικών και εσωτερικών συναρτήσεων ποινής δεν αλλάζει σημαντικά.

2.6 Αλγόριθμοι βελτιστοποίησης

Οι αλγόριθμοι βελτιστοποίησης είναι επαναληπτικοί. Συνήθως, ξεκινούν από μια αρχική εκτίμηση των βέλτιστων σχεδιασμού και στη συνέχεια δημιουργούν μια αλληλουχία βελτιωμένων σχεδιασμών μέχρι να καταλήξουν στο βέλτιστο. Η στρατηγική μέσω της οποίας από μια επανάληψη προκύπτει η επόμενη, κατηγοριοποιεί και τον αλγόριθμο βελτιστοποίησης.

Κάθε αλγόριθμος βελτιστοποίησης πρέπει να έχει τις παρακάτω ιδιότητες:

- *Σθεναρότητα (Robustness)*: Ένας αλγόριθμος πρέπει να μπορεί να αντιμετωπίσει μια πληθώρα προβλημάτων.
- *Απόδοση (Efficiency)*: Ο αλγόριθμος δεν θα πρέπει να απαιτεί πολύ μεγάλη υπολογιστική ισχύ ή χρόνο, ώστε να βρει τη βέλτιστη λύση.
- *Ακρίβεια (Accuracy)*: Ένας αλγόριθμος βελτιστοποίησης θα πρέπει να μπορεί να αναγνωρίζει μια λύση με ακρίβεια, χωρίς να είναι υπερευαίσθητος σε αριθμητικής ακρίβειας σφάλματα.

Οι παραπάνω απαιτήσεις είναι αλληλοσυγκρουόμενες. Για παράδειγμα, ένας πολύ γρήγορος (στη σύγκλιση) αλγόριθμος μπορεί να απαιτεί υπερβολικά μεγάλο αποθηκευτικό χώρο ή μνήμη για προβλήματα με πολλές μεταβλητές σχεδιασμού. Αντίθετα, ένας αλγόριθμος με μεγάλη σθεναρότητα (Robustness) μπορεί να απαιτεί πολλές επαναλήψεις και μεγάλο υπολογιστικό χρόνο ώστε να βρει το βέλτιστο σχεδιασμό.

Μερικοί αλγόριθμοι διατηρούν ένα ποσοστό της πληροφορίας από τους σχεδιασμούς των προηγούμενων σχεδιασμών, ενώ άλλοι χρησιμοποιούν μόνο πληροφορία του τρέχοντος σχεδιασμού. Οι αλγόριθμοι, σχετικά με το είδος της πληροφορίας, διακρίνονται στους αλγόριθμους μηδενικής, πρώτης και δεύτερης τάξης. Οι αλγόριθμοι μηδενικής τάξης χρησιμοποιούν μόνο την τιμή της αντικειμενικής συνάρτησης στην έρευνα για το βέλτιστο, οι αλγόριθμοι πρώτης τάξης χρησιμοποιούν και την πρώτη παράγωγο της αντικειμενικής συνάρτησης, ενώ οι αλγόριθμοι δεύτερης τάξης χρησιμοποιούν και τη δεύτερη παράγωγο, εκτός από την τιμή της αντικειμενικής συνάρτησης και την πρώτη της παράγωγο.

Οι αλγόριθμοι μηδενικής τάξης διακρίνονται σε *αιτιοκρατικούς* και *στοχαστικούς* αλγόριθμους, ανάλογα με τον τρόπο που σχηματίζονται οι νέοι σχεδιασμοί. Οι αιτιοκρατικοί αλγόριθμοι προσεγγίζουν επαναληπτικά το βέλτιστο σχεδιασμό πολύ γρήγορα. Το βασικό τους μειονέκτημα είναι πως εγκλωβίζονται εύκολα σε τοπικά ελάχιστα. Οι στοχαστικοί αλγόριθμοι αναζητούν με έναν τυχαμιακό τρόπο νέους σχεδιασμούς καλύτερους από τους υπάρχοντες ώστε να οδηγηθούν

στο βέλτιστο. Δεν εγκλωβίζονται τόσο εύκολα σε τοπικά ελάχιστα όπως οι αιτιοκρατικοί αλγόριθμοι, αλλά απαιτούν μεγαλύτερη υπολογιστική ισχύ και χρόνο από αυτούς για να συγκλίνουν στο βέλτιστο.

Οι αλγόριθμοι βελτιστοποίησης μπορούν να διαχωριστούν σε αλγόριθμους που διαχειρίζονται έναν σχεδιασμό τη φορά, και σε αυτούς που διαχειρίζονται έναν πληθυσμό από σχεδιασμούς ταυτόχρονα. Όλοι οι αιτιοκρατικοί αλγόριθμοι διαχειρίζονται έναν σχεδιασμό τη φορά. Από τους στοχαστικούς αλγόριθμους, ο γνωστότερος αλγόριθμος που χρησιμοποιεί έναν σχεδιασμό τη φορά, είναι ο αλγόριθμος της Προσομιούμενης Ανόπτωσης (Simulated Annealing).

Οι στοχαστικοί αλγόριθμοι βελτιστοποίησης που διαχειρίζονται έναν πληθυσμό σχεδιασμών είναι γνωστοί και ως εξελικτικοί αλγόριθμοι. Οι εξελικτικοί αλγόριθμοι μοντελοποιούν συνήθως ένα φαινόμενο, φυσικό, κοινωνικό ή βιολογικό. Η λειτουργία τους είναι συνήθως παράλληλη, δηλαδή δημιουργούνται πολλοί σχεδιασμοί-λύσεις ταυτόχρονα. Οι εξελικτικοί αλγόριθμοι χαρακτηρίζονται από σθεναρότητα (robustness) και έχουν την ικανότητα να εντοπίζουν την περιοχή του ολικά βέλτιστου σχεδιασμού μετά από ένα μεγάλο αριθμό εκτιμήσεων της αντικειμενικής συνάρτησης. Η γνωστότερη στοχαστική μέθοδος που χρησιμοποιεί πληθυσμό σχεδιασμών, είναι οι Γενετικοί Αλγόριθμοι (Genetic Algorithms). Πληθυσμό σχεδιασμών επίσης χρησιμοποιούν οι Αποικίες Μυρμηγκιών (Ant Colonies) και οι Στρατηγικές Εξέλιξης (Evolution Strategies). Τα τελευταία χρόνια οι εξελικτικοί αλγόριθμοι έχουν εφαρμοστεί σε ερευνητικό επίπεδο, στο πεδίο του βέλτιστου σχεδιασμού των κατασκευών, και έχουν αποδειχθεί από τις πλέον αξιόπιστες μεθόδους.

2.7 Στρατηγικές εξέλιξης

Οι στρατηγικές εξέλιξης ανήκουν στις επονομαζόμενες Δαρβίνειες μεθόδους, επειδή προσομοιάζουν τη διαδικασία εξέλιξης των ειδών, όπως την παρουσίασε πρώτος ο Δαρβίνος. Οι στρατηγικές εξέλιξης, σε αντίθεση με τους αιτιοκρατικούς αλγόριθμους, διαχειρίζονται έναν πληθυσμό από σχεδιασμούς-λύσεις ταυτόχρονα. Οι λύσεις αυτές είναι ανεξάρτητες η μία από την άλλη, επομένως είναι δυνατή η υλοποίηση των στρατηγικών εξέλιξης σε παράλληλο υπολογιστικό περιβάλλον. Αφού δημιουργηθεί ένας αρχικός πληθυσμός λύσεων με τυχαίο τρόπο, στη συνέχεια θα δράσουν σε αυτόν οι τελεστές *μετάλλαξης*, *ανασυνδυασμού* και *επιλογής* ώστε να εξελιχθεί ο πληθυσμός και να πετύχει τον εντοπισμό της βέλτιστης λύσης.

Το βασικό πλεονέκτημα των στρατηγικών εξέλιξης, είναι πως λόγω του τυχηματικού χαρακτήρα τους δεν εγκλωβίζονται εύκολα σε τοπικά ελάχιστα, κατά τη διάρκεια της έρευνας για τη βέλτιστη λύση, επομένως έχουν μεγαλύτερες πιθανότητες να εντοπίζουν την περιοχή της ολικά βέλτιστης λύσης. Σε προβλήματα με πάρα πολλές μεταβλητές σχεδιασμού, ή και πολλές αντικειμενικές συναρτήσεις, οι εξελικτικοί αλγόριθμοι είναι οι μόνοι που μπορούν να δώσουν αποδεκτή λύση. Το βασικό μειονέκτημα των στρατηγικών εξέλιξης, όπως και των περισσότερων εξελικτικών αλγόριθμων, είναι πως απαιτείται μεγάλος αριθμός υπολογισμών της αντικειμενικής συνάρτησης, για τον εντοπισμό της βέλτιστης λύσης.

Ο στόχος των στρατηγικής εξέλιξης, όπως και κάθε αλγόριθμου βελτιστοποίησης, σε ένα πρόβλημα βελτιστοποίησης με μια αντικειμενική συνάρτηση, είναι η βελτιστοποίηση μιας αντικειμενικής συνάρτησης F ως προς το διάνυσμα των παραμέτρων της $\mathbf{y} = y_1, y_2, \dots, y_n$, το οποίο ονομάζεται διάνυσμα σχεδιασμού. Ο στόχος δηλαδή, είναι η εύρεση ενός διανύσματος \mathbf{y} τέτοιο ώστε:

$$F \mathbf{y} \rightarrow \text{optimum, } \mathbf{y} \in Y \quad (2.28)$$

Ο χώρος Y μπορεί να είναι ο n -διάστατος χώρος των πραγματικών αριθμών \mathbf{R}^n , ο n -διάστατος χώρος των ακέραιων αριθμών \mathbf{Z}^n , ο χώρος των δυαδικών αριθμών \mathbf{B}^n ή και οποιοσδήποτε συνδυασμός τους.

Οι στρατηγικές εξέλιξης, όπως προαναφέρθηκε, δρουν σε άτομα που αποτελούν ένα πληθυσμό. Ο πληθυσμός συνήθως συμβολίζεται με \mathbf{B} ενώ τα άτομα με a . Το k άτομο ενός πληθυσμού, συμβολίζεται με a_k και αποτελείται από το διάνυσμα σχεδιασμού \mathbf{y}_k , την τιμή της αντικειμενικής συνάρτησης $F_k = F \mathbf{y}_k$ και συνήθως από ένα σετ ενδογενών παραμέτρων \mathbf{s}_k , που ονομάζονται ενδογενείς παράμετροι στρατηγικής, έτσι ώστε:

$$a_k = \mathbf{y}_k, \mathbf{s}_k, F_k \quad (2.29)$$

Οι ενδογενείς παράμετροι στρατηγικής ελέγχουν διάφορους τελεστές των στρατηγικών εξέλιξης, όπως για παράδειγμα τον τελεστή της *μετάλλαξης* και του *ανασυνδυασμού*, και μπορούν να προσαρμόζονται κατά τη διάρκεια της εξέλιξης. Η μορφή των παραμέτρων στρατηγικής δεν είναι η ίδια σε όλους τους αλγόριθμους στρατηγικής εξέλιξης, αλλά διαφοροποιείται από αλγόριθμο σε αλγόριθμο, όπως θα δειχθεί στο τρίτο κεφάλαιο.

Γενικά, κατά τη διάρκεια μιας γενιάς στρατηγικής εξέλιξης, δημιουργούνται λάτομα απογόνων από ένα πληθυσμό μ γονέων. Οι παράμετροι μ, λ καθώς και μία ακόμη παράμετρος, που συμβολίζεται με ρ και ελέγχει τον τελεστή ανασυνδυασμού για τον οποίο θα γίνει αναφορά στη συνέχεια, ονομάζονται *εξώγενείς παράμετροι στρατηγικής*, και διατηρούνται σταθερές κατά τη διάρκεια της εξέλιξης. Ο τρόπος με τον οποίο δημιουργούνται οι απόγονοι από τον πληθυσμό των γονέων συμβολίζεται συνοπτικά με $\mu / \rho \pm \lambda$. Η παράμετρος ανασυνδυασμού ρ δηλώνει τον αριθμό των γονέων που θα συνδυαστούν για τη δημιουργία ενός απογόνου πριν δράσει σε αυτόν ο τελεστής μετάλλαξης. Προφανώς, πρέπει να ισχύει $\rho \leq \mu$. Ο συμβολισμός \pm περιγράφει τον τρόπο με τον οποίο θα δράσει ο τελεστής επιλογής στον πληθυσμό. Στην περίπτωση του «+», η επιλογή των ατόμων που θα συνεχίσουν στην επόμενη γενιά γίνεται από ολόκληρο τον πληθυσμό, δηλαδή θα επιλεγούν τα μ άτομα από το συνολικό πληθυσμό $\mu + \lambda$ των γονέων και των απογόνων. Στην περίπτωση του «,» η επιλογή θα γίνει μόνο από τον πληθυσμό των απογόνων, δηλαδή θα επιλεγούν τα μ καλύτερα άτομα από τους λ απόγονους, χωρίς να ληφθεί υπόψιν ο πληθυσμός των γονέων. Σε αυτήν την περίπτωση θα πρέπει προφανώς να ισχύει $\mu < \lambda$.

Στο επόμενο κεφάλαιο περιγράφονται οι αλγόριθμοι στρατηγικής εξέλιξης που χρησιμοποιήθηκαν στα πλαίσια του του παραδοτέου Π1.1, και θα δίνεται μια πλήρης περιγραφή του τρόπου λειτουργίας των τελεστών μετάλλαξης, ανασυνδυασμού και επιλογής σε αυτούς τους αλγόριθμους.

3 ΣΤΡΑΤΗΓΙΚΕΣ ΕΞΕΛΙΞΗΣ

3.1 Ο αλγόριθμος A-ES

Ο αλγόριθμος A-ES (Adaptive Evolution Strategy) είναι ο απλούστερος αλγόριθμος στρατηγικής εξέλιξης με αυτοπροσαρμοζόμενες ενδογενείς παραμέτρους στρατηγικής. Κάθε άτομο γονέα ή απογόνου αποτελείται από το διάνυσμα σχεδιασμού, το διάνυσμα ενδογενών παραμέτρων στρατηγικής και την τιμή της αντικειμενικής συνάρτησης, όπως ακριβώς περιγράφεται από τη σχέση (3.1). Στο σχήμα 3.1 φαίνεται σε μορφή ψευδοκώδικα η ροή του αλγόριθμου $\mu/\rho^{\pm}\lambda$ - A-ES, όπως αυτός υλοποιήθηκε.

Αρχικά, στη γενιά $g = 0$, δημιουργείται ο αρχικός πληθυσμός γονέων \mathbf{B}_p^0 ο οποίος αποτελείται από μ άτομα όπως φαίνεται στη γραμμή #3. Στη συνέχεια υπάρχει ο κύριος βρόχος του αλγόριθμου A-ES. Σε κάθε γενιά εξέλιξης, αρχικά δημιουργείται ένας πληθυσμός από απογόνους όπως φαίνεται στις γραμμές #5-12. Για να συμβεί αυτό, δημιουργείται αρχικά ένας υποπληθυσμός \mathbf{D}_i από ρ άτομα του πληθυσμού των γονέων \mathbf{B}_p^g (γραμμή #6). Σε αυτόν τον υποπληθυσμό \mathbf{D}_i θα δράσει ο τελεστής ανασυνδυασμού ώστε να προκύψει ένας αρχικός απόγονος με διάνυσμα ενδογενών στρατηγικών παραμέτρων \mathbf{s}_i και διάνυσμα σχεδιασμού \mathbf{y}_i (γραμμές #7-8). Σε αυτόν ακριβώς τον απόγονο, θα δράσει ο τελεστής μετάλλαξης ώστε να προκύψει ο τελικός απόγονος.

Πρώτα, ο τελεστής μετάλλαξης θα δράσει στο διάνυσμα των παραμέτρων στρατηγικής, όπως φαίνεται στη γραμμή #9. Στη συνέχεια, θα μεταλλαχθεί το διάνυσμα των παραμέτρων σχεδιασμού, στη γραμμή #10. Θα πρέπει εδώ να σημειωθεί, πως πρέπει να τηρηθεί αυστηρά αυτή η σειρά μετάλλαξης, δηλαδή πρώτα των παραμέτρων στρατηγικής και έπειτα του διανύσματος των μεταβλητών σχεδιασμού. Σαν τελικό βήμα στη δημιουργία ενός απογόνου, υπολογίζεται η τιμή της αντικειμενικής συνάρτησης (γραμμή #11). Αφού έχει δημιουργηθεί ο πληθυσμός απογόνων \mathbf{B}_o^g , θα δράσει σε αυτόν ο τελεστής επιλογής, ώστε να προκύψει ο πληθυσμός γονέων της επόμενης γενιάς. Αυτό συμβαίνει στις γραμμές #14-17. Ανάλογα με το αν ο τελεστής επιλογής είναι τύπου «,» ή «+», θα εκτελεστεί η γραμμή #15 ή #16. Αν ο τελεστής είναι τύπου «,», τότε η επιλογή των νέων γονέων θα βασιστεί μόνο στον πληθυσμό των απογόνων, από τον οποίο θα επιλεγούν ως νέοι γονείς οι μ απόγονοι με τις μικρότερες τιμές της αντικειμενικής συνάρτησης.

Η διαδικασία της εξέλιξης συνεχίζεται μέχρι να ικανοποιηθεί το κριτήριο τερματισμού (γραμμή #19). Ως κριτήριο τερματισμού μπορούν να χρησιμοποιηθούν :

- Ο μέγιστος αριθμός γενιών g
- Ο μέγιστος υπολογιστικός χρόνος εκτέλεσης της διαδικασίας εξέλιξης
- Η σύγκλιση των τιμών της αντικειμενικής συνάρτησης
- Η σύγκλιση των τιμών των παραμέτρων σχεδιασμού
- Η σύγκλιση των τιμών των ενδογενών παραμέτρων στρατηγικής

3.1.1 Ο τελεστής επιλογής

Σε μια στρατηγική εξέλιξης, η έρευνα θα πρέπει να οδηγείται με κάποιον τρόπο προς καλύτερους χώρους ως προς την αντικειμενική συνάρτηση και των μεταβλητών σχεδιασμού. Αυτός ακριβώς είναι ο ρόλος του τελεστή επιλογής, να οδηγήσει δηλαδή την έρευνα κατά τη διάρκεια της εξέλιξης σε χώρους οι οποίοι βρίσκονται εγγύτερα του ολικού ελαχίστου της αντικειμενικής συνάρτησης. Η δράση του τελεστή επιλογής είναι μια ντετερμινιστική διαδικασία κατά την οποία θα επιλέγουν τα καλύτερα άτομα του πληθυσμού, ως οι γονείς της επόμενης γενιάς. Συμβολικά, αυτό μπορεί να γραφεί ως εξής:

$$\mathbf{B}_p^{g+1} := a_{1,\gamma}, a_{2,\gamma}, \dots, a_{\mu,\gamma} \quad (3.1)$$

Ο συμβολισμός $a_{\mu,\gamma}$ σημαίνει πως από έναν πληθυσμό γ ατόμων στον οποίο θα δράσει ο τελεστής επιλογής, θα επιλεγούν τα μ καλύτερα άτομα, ως προς την τιμή της αντικειμενικής τους συνάρτησης, και θα αποτελέσουν τον πληθυσμό γονέων της επόμενης γενιάς.

Όπως αναφέρθηκε στην προηγούμενη παράγραφο, υπάρχουν δύο μορφές του τελεστή επιλογής, ο τελεστής «+» και ο τελεστής «,». Ο τελεστής επιλογής «+» ονομάζεται συνήθως και τελεστής επιλογής $\mu + \lambda$, ενώ ο τελεστής επιλογής «,» ονομάζεται συνήθως και τελεστής επιλογής μ, λ . Στην περίπτωση τελεστή επιλογής μ, λ , ο πληθυσμός από στον οποίο θα δράσει αυτός αποτελείται από $\gamma = \lambda$ άτομα, τους απογόνους της τωρινής γενιάς. Οι γονείς επομένως, ακόμη και αν είναι καλύτεροι από τους απογόνους, δεν λαμβάνουν μέρος στη διαδικασία της επιλογής.

Ο τελεστής επιλογής τύπου μ, λ χρησιμοποιείται κυρίως σε άνευ ορίων χώρους Y , όπως ο $Y = \mathcal{R}^n$.

Αντίθετα με τον τελεστή επιλογής μ, λ , ο τελεστής επιλογής $\mu + \lambda$ επιτρέπει στον γονείς της τωρινής γενιάς να λάβουν μέρος στη διαδικασία της επιλογής επομένως, ο πληθυσμός στον οποίο θα δράσει ο τελεστής αποτελείται από $\gamma = \mu + \lambda$ άτομα. Σε αυτήν την περίπτωση, δίνεται η δυνατότητα στους γονείς να επιβιώνουν στις επόμενες γενιές εξέλιξης, όταν αυτοί είναι καλύτεροι από τους απογόνους γι' αυτό και ο τελεστής επιλογής «+» ονομάζεται «ελιτιστικός», διατηρεί δηλαδή την ελίτ των ατόμων από γενιά σε γενιά. Η χρήση του τελεστή επιλογής «+» προτείνεται ιδιαίτερα όταν η έρευνα γίνεται σε διακριτούς χώρους.

Procedure $\mu/\rho \pm \lambda$ -A-ES

1. Begin
 2. $g := 0$
 3. initialize $\mathbf{B}_p^0 := \mathbf{y}_m^0, \mathbf{s}_m^0, F \mathbf{y}_m^0$, $m = 1, \dots, \mu$
 4. Repeat
 5. **For** $l := 1$ **To** λ **Do Begin**
 6. $\mathbf{D}_l := \text{marriage } \mathbf{B}_p^g$
 7. $\mathbf{s}_l := \text{s_recombination } \mathbf{D}_l$
 8. $\mathbf{y}_l := \text{y_recombination } \mathbf{D}_l$
 9. $\tilde{\mathbf{s}}_l := \text{s_mutation } \mathbf{s}_l$
 10. $\tilde{\mathbf{y}}_l := \text{y_mutation } \mathbf{y}_l$
 11. $\tilde{F}_l := F \tilde{\mathbf{y}}_l$
 12. **End**
 13. $\mathbf{B}_o^g := \mathbf{y}_l^g, \mathbf{s}_l^g, F \mathbf{y}_l^g$, $l = 1, \dots, \lambda$
 14. **Case selection_type Of**
 15. μ, λ : $\mathbf{B}_p^{g+1} := \text{selection } \mathbf{B}_o^g, \mu$
 16. $\mu + \lambda$: $\mathbf{B}_p^{g+1} := \text{selection } \mathbf{B}_o^g, \mathbf{B}_p^g, \mu$
 17. **End**
 18. $g := g + 1$
 19. **Until** termination_condition
 20. End
-

3.1.2 Ο τελεστής μετάλλαξης

Ο τελεστής μετάλλαξης συναντάται σε όλες τις μορφές του εξελικτικού προγραμματισμού. Αν και δεν υπάρχει κάποιος τελεστής μετάλλαξης με κοινά αποδεκτό τρόπο δράσης, κάθε τέτοιος τελεστής πρέπει να έχει τις τρεις παρακάτω ιδιότητες:

- *Δυνατότητα προσέγγισης (Reachability)*: Ένας τελεστής μετάλλαξης, θα πρέπει να μπορεί να μεταλλάξει ένα άτομο $a_p = \mathbf{y}_p, \mathbf{s}_p, F_p$, σε ένα άλλο $\tilde{a}_p = \tilde{\mathbf{y}}_p, \tilde{\mathbf{s}}_p, \tilde{F}_p$, μέσα από έναν πεπερασμένο αριθμό μεταλλάξεων.
- *Αμεροληψία (Unbiasedness)*: Η Δαρβίνεια Θεωρία περί της εξέλιξης των ειδών απαιτεί έναν αμερόληπτο τελεστή μετάλλαξης. Για να δράσει ο τελεστής επιλογής σε έναν πληθυσμό, θα πρέπει να λάβει υπόψη τις τιμές της αντικειμενικής συνάρτησης κάθε ατόμου του πληθυσμού, οπότε η δράση του μεροληπτεί υπέρ των ατόμων με τις

μικρότερες τιμές της αντικειμενικής συνάρτησης. Αντίθετα, ο τελεστής μετάλλαξης δεν πρέπει να χρησιμοποιεί μια τέτοια πληροφορία. Η απαίτηση για μια τέτοια ιδιότητα οδηγεί στην περίπτωση έρευνας στο n -διάστατο χώρο των πραγματικών αριθμών \mathbb{R}^n σε μεταλλάξεις που βασίζονται στην κανονική κατανομή.

- *Εξελιξιμότητα (Scalability)*: Η δύναμη της μετάλλαξης (mutation strength) θα πρέπει να είναι τέτοια, ώστε να είναι πιθανή η βελτίωση του πληθυσμού από γενιά σε γενιά. Αυτό σημαίνει, πως κατά τη διάρκεια της εξέλιξης, η δύναμη μετάλλαξης θα πρέπει να προσαρμόζεται ανάλογα με τον χώρο στον οποίο βρίσκεται η έρευνα .

Θα πρέπει να τονιστεί πως οι παραπάνω τρεις βασικές ιδιότητες δεν αποτελούν αναγκαία συνθήκη για την επίτευξη της εύρεσης του βέλτιστου.

Στο παραδοτέο Π1.1, τα προβλήματα τα οποία αντιμετωπίστηκαν αφορούσαν κυρίως αντικειμενικές συναρτήσεις με συνεχείς μεταβλητές σχεδιασμού. Επομένως η έρευνα του βέλτιστου έγινε στο n -διάστατο συνεχή χώρο \mathbb{R}^n που ορίζουν οι n μεταβλητές σχεδιασμού. Για κάθε μεταβλητή σχεδιασμού, υπάρχει και μια ενδογενής παράμετρος στρατηγικής η οποία ορίζει τη δύναμη της μετάλλαξης για τη συγκεκριμένη μεταβλητή.

Για ένα διάνυσμα μεταβλητών σχεδιασμού \mathbf{y} , σχέση που δίνει το μεταλλαγμένο διάνυσμα $\tilde{\mathbf{y}}$, αφού δράσει ο τελεστής μετάλλαξης, είναι:

$$\tilde{\mathbf{y}} := \mathbf{y} + \mathbf{z} \quad (3.2)$$

Στη σχέση (3.2), το διάνυσμα \mathbf{z} είναι το διάνυσμα των μεταλλάξεων το οποίο δίνεται από τη σχέση:

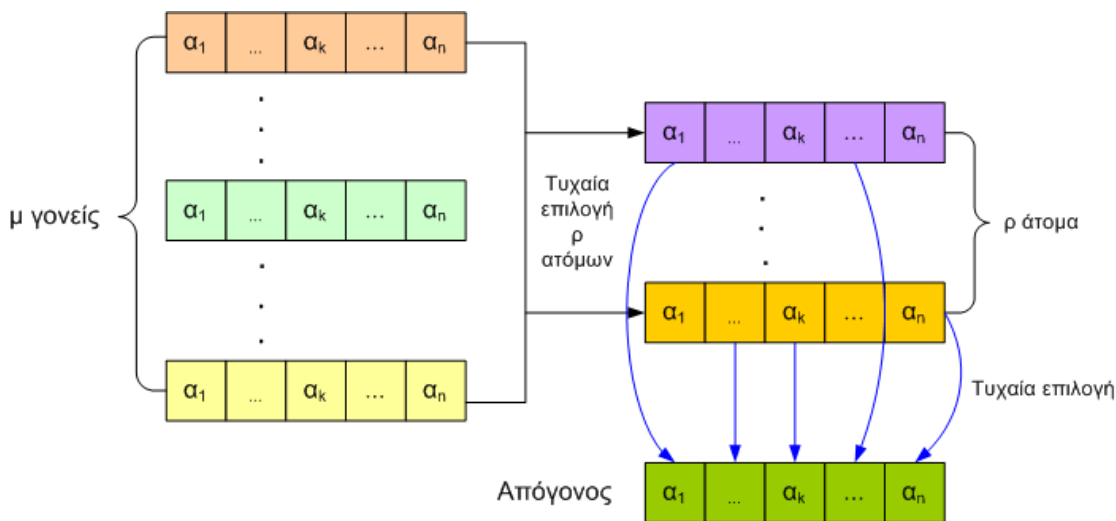
$$\mathbf{z} := \tilde{\sigma}_1 N_1 \ 0,1, \tilde{\sigma}_2 N_2 \ 0,1, \dots, \tilde{\sigma}_n N_n \ 0,1 \quad (3.3)$$

όπου $\tilde{\sigma}_1, \tilde{\sigma}_2, \dots, \tilde{\sigma}_n$ είναι οι μεταλλαγμένες ενδογενείς παράμετροι στρατηγικής (δυνάμεις μετάλλαξης) κάθε μιας από τις n μεταβλητές σχεδιασμού, και $N_1 \ 0,1, N_2 \ 0,1, \dots, N_n \ 0,1$ είναι ανεξάρτητοι μεταξύ τους τυχαίοι αριθμοί από κανονική κατανομή.

3.1.3 Ο τελεστής ανασυνδυασμού

Ο τελεστής ανασυνδυασμού είναι το ανάλογο του τελεστή διασταύρωσης (crossover operator) των γενετικών αλγορίθμων. Σε αντίθεση με τον τελευταίο ο οποίος δρα σε δύο γονείς από τους οποίους προκύπτουν τελικά δύο απόγονοι, ο τελεστής ανασυνδυασμού μπορεί να δράσει σε $\rho \leq \mu$ γονείς, από τους οποίους θα προκύψει τελικά ένας απόγονος.

Στις στρατηγικές εξέλιξης υπάρχουν δύο βασικές μορφές του τελεστή ανασυνδυασμού, ο διακριτός ανασυνδυασμός και ο ενδιάμεσος ανασυνδυασμός. Στο σχήμα 3.2 φαίνεται ο τρόπος δράσης του διακριτού ανασυνδυασμού στην περίπτωση που το πρόβλημα βελτιστοποίησης ορίζεται σε συνεχή χώρο.



Σχήμα 3.2: Διακριτός ανασυνδυασμός

Από τους μ γονείς επιλέγονται με τυχαίο τρόπο ρ άτομα τα οποία θα αποτελέσουν την οικογένεια από την οποία θα προκύψει ένας αρχικός απόγονος. Κάθε μεταβλητή σχεδιασμού ή ενδογενής παράμετρος στρατηγικής του απόγονου, επιλέγεται να είναι ίση με την ίδια μεταβλητή

σχεδιασμού ή ενδογενή παράμετρο στρατηγικής ενός μέλους της οικογένειας των ρ ατόμων, με τυχαίο πάλι τρόπο.

Αντίθετα με τον διακριτό ανασυνδυασμό, ο ενδιάμεσος ανασυνδυασμός λαμβάνει υπόψην κάθε μέλος της οικογένειας των ρ ατόμων. Σε αυτήν την περίπτωση, απλά η k μεταβλητή σχεδιασμού ή παράμετρος στρατηγικής r_k δίνεται από τη σχέση:

$$r_k = \frac{1}{\rho} \sum_{m=1}^{\rho} a_{m \ k} \quad (3.4)$$

Το στοιχείο $a_{m \ k}$ αφορά την k μεταβλητή σχεδιασμού (ή παράμετρο στρατηγικής) του m ατόμου που λαμβάνουν μέρος στον ανασυνδυασμό. Θα πρέπει να σημειωθεί πως δεν έχει σημασία αν ο ανασυνδυασμός γίνει πρώτα στις μεταβλητές σχεδιασμού ή στις παραμέτρους στρατηγικής.

Στην περίπτωση προβλημάτων σε διακριτούς χώρους, θα πρέπει να ληφθεί μέριμνα ώστε το προκύπτον άτομο να ανήκει μετά τον ανασυνδυασμό στο διακριτό χώρο του προβλήματος.

Στο παραδοτέο Π1.1, ο αλγόριθμος A-ES χρησιμοποιήθηκε στις περισσότερες εφαρμογές χωρίς τελεστή ανασυνδυασμού ($\rho = 1$). Στην περιπτώσεις που χρησιμοποιήθηκε ο τελεστής ανασυνδυασμού, επιλέχθηκε η μορφή του ενδιάμεσου συνδυασμού.

3.1.4 Αυτοπροσαρμογή των ενδογενών παραμέτρων στρατηγικής

Όπως έχει προαναφερθεί, κάθε άτομο στον αλγόριθμο A-ES, εκτός από το διάνυσμα των μεταβλητών σχεδιασμού, έχει και το δικό του διάνυσμα ενδογενών παραμέτρων στρατηγικής που παίζουν τον ρόλο της δύναμης μετάλλαξης (mutation strength) κάθε μεταβλητής σχεδιασμού. Αυτές οι παράμετροι στρατηγικής, κατά τη διάρκεια της εξέλιξης, μπορούν προσαρμόζονται (μεταλλάσσονται) και οι ίδιες, ώστε να είναι πιθανή η βελτίωση του πληθυσμού από γενιά σε γενιά.

Έστω $\mathbf{s} = \sigma_1, \sigma_2, \dots, \sigma_n$ το διάνυσμα των ενδογενών παραμέτρων στρατηγικής. Το μεταλλαγμένο διάνυσμα $\tilde{\mathbf{s}}$ των παραμέτρων στρατηγικής, προκύπτει από τη σχέση:

$$\tilde{\mathbf{s}} = e^{\tau_0 N_0 \cdot 0,1} \sigma_1 e^{\tau N_1 \cdot 0,1}, \sigma_2 e^{\tau N_2 \cdot 0,1}, \dots, \sigma_n e^{\tau N_n \cdot 0,1} \quad (3.5)$$

Οι παράμετροι τ_0, τ ονομάζονται παράμετροι μάθησης και αποτελούν εξωγενείς παραμέτρους στρατηγικής. Δίνονται από τις σχέσεις:

$$\tau_0 = \frac{c}{\sqrt{2n}} \quad (3.6)$$

$$\tau = \frac{c}{\sqrt{2\sqrt{n}}} \quad (3.7)$$

Η σταθερά c στις παραπάνω σχέσεις συνήθως τίθεται ίση με τη μονάδα ενώ n είναι το πλήθος των μεταβλητών σχεδιασμού του προβλήματος. Οι $N_0 \cdot 0,1, N_1 \cdot 0,1, N_2 \cdot 0,1, \dots, N_n \cdot 0,1$ είναι τυχαίοι αριθμοί από κανονική κατανομή με μηδενική μέση τιμή και διασπορά ίση με τη μονάδα και ανεξάρτητοι μεταξύ τους.

Αφού δράσει ο τελεστής μετάλλαξης όπως περιγράφηκε στο διάνυσμα \mathbf{s} των παραμέτρων στρατηγικής ενός ατόμου, αυτό θα μεταλλαχθεί στο $\tilde{\mathbf{s}} = \tilde{\sigma}_1, \tilde{\sigma}_2, \dots, \tilde{\sigma}_n$. Με αυτό το διάνυσμα παραμέτρων στρατηγικής, και από τις σχέσεις (3.2) και (3.3) προκύπτει το μεταλλαγμένο διάνυσμα μεταβλητών σχεδιασμού.

3.2 Ο αλγόριθμος CMA-ES

Ο αλγόριθμος CMA-ES¹ προτάθηκε από τον Hansen [?] και γενικεύει τον τρόπο δράσης του τελεστή μετάλλαξης, εισάγοντας συσχετιζόμενες μεταξύ τους μεταλλάξεις. Ο αλγόριθμος A-ES που περιγράφηκε στην προηγούμενη παράγραφο, εισάγει n παραμέτρους στρατηγικής που παίζουν τον ρόλο της δύναμης μετάλλαξης για κάθε μια από τις n μεταβλητές σχεδιασμού του προβλήματος. Οι n αυτές παράμετροι στρατηγικής είναι ανεξάρτητες μεταξύ τους, δηλαδή η δύναμη μετάλλαξης της j μεταβλητής σχεδιασμού δεν συσχετίζεται με οποιοδήποτε τρόπο με την

¹ CMA: Covariance Matrix Adaptation

δύναμη μετάλλαξης της k μεταβλητής σχεδιασμού. Αντίθετα, για ένα πρόβλημα με n μεταβλητές σχεδιασμού, ο αλγόριθμος CMA-ES χρησιμοποιεί:

$$\frac{n^2 - n}{2}$$

επιπλέον παραμέτρους στρατηγικής οι οποίες ορίζουν τις συσχετίσεις μεταξύ των αρχικών n παραμέτρων στρατηγικής.

Η δεύτερη βασική διαφορά του αλγόριθμου CMA-ES από τον αλγόριθμο A-ES, είναι πως στον τελευταίο κάθε άτομο του πληθυσμού έχει το δικό του διάνυσμα ενδογενών παραμέτρων στρατηγικής. Αντίθετα, στον αλγόριθμο CMA-ES οι

$$\frac{n^2 + n}{2}$$

συνολικά παράμετροι στρατηγικής αφορούν το σύνολο του πληθυσμού, είναι δηλαδή καθολικές, επομένως τα άτομα δεν έχουν δικές τους παραμέτρους στρατηγικής.

3.2.1 Ο πίνακας συνδιασποράς \mathbf{C}

Η εισαγωγή των επιπλέον ενδογενών παραμέτρων στρατηγικής γίνεται μέσω ενός πίνακα συνδιασποράς \mathbf{C} . Ο πίνακας συνδιασποράς \mathbf{C} είναι συμμετρικός και θετικά ορισμένος, δηλαδή, για κάθε $\mathbf{x} \in \mathbb{R}^n$ με $\mathbf{x} \neq \mathbf{0}$ ισχύει $\mathbf{x}^T \mathbf{C} \mathbf{x} > 0$. Για τον πίνακα \mathbf{C} υπάρχει μια ορθοκανονική βάση ιδιοδιανυσμάτων $\mathbf{B} = \mathbf{b}_1, \mathbf{b}_2, \dots, \mathbf{b}_n$ με αντίστοιχες ιδιοτιμές $\lambda_1, \lambda_2, \dots, \lambda_n > 0$. Οι ιδιοτιμές είναι θετικές επειδή ο πίνακας \mathbf{C} είναι συμμετρικός και θετικά ορισμένος.

Ο πίνακας \mathbf{C} μπορεί να παραγοντοποιηθεί στη μορφή:

$$\mathbf{C} = \mathbf{B} \mathbf{L} \mathbf{B}^T \quad (3.8)$$

όπου το μητρώο \mathbf{B} είναι ένα ορθοκανονικό μητρώο τέτοιο ώστε $\mathbf{B} \mathbf{B}^T = \mathbf{I}$ και περιέχει ως στήλες τα ιδιοδιανύσματα του πίνακα \mathbf{C} . Σε μια κανονική κατανομή $N(\mathbf{m}, \mathbf{C})$ με μέση τιμή \mathbf{m} και πίνακα συνδιασποράς \mathbf{C} , τα ιδιοδιανύσματα του μητρώου \mathbf{B} αποτελούν τους κύριους άξονες της κατανομής αυτής. Το μητρώο \mathbf{L} στη σχέση (3.8) είναι ένα διαγώνιο μητρώο με στοιχεία τις ιδιοτιμές του πίνακα \mathbf{C} .

Γεωμετρικά, κάθε πίνακας συνδιασποράς χαρακτηρίζεται από ένα υπερ-ελλειψοειδές με εξίσωση:

$$\mathbf{x}^T \mathbf{C}^{-1} \mathbf{x} = 1, \quad \mathbf{x} \in \mathbb{R}^n \quad (3.9)$$

Ο αντίστροφος του πίνακα συνδιασποράς \mathbf{C}^{-1} δίνεται από τη σχέση:

$$\mathbf{C}^{-1} = \mathbf{B} \mathbf{L}^{-1} \mathbf{B}^T \quad (3.10)$$

Οι διευθύνσεις των κύριων αξόνων του ελλειψοειδούς αντιστοιχούν στα ιδιοδιανύσματα του πίνακα \mathbf{B} . Αντίστοιχα, τα μήκη των αξόνων του ελλειψοειδούς αντιστοιχούν στις τετραγωνικές ρίζες των ιδιοτιμών του πίνακα συνδιασποράς \mathbf{C} .

Η τετραγωνική ρίζα του πίνακα συνδιασποράς, επειδή αυτός είναι συμμετρικός προκύπτει από τη σχέση:

$$\sqrt{\mathbf{C}} = \mathbf{B} \sqrt{\mathbf{L}} \mathbf{B}^T \quad (3.11)$$

όπου $\sqrt{\mathbf{L}}$ είναι ένα διαγώνιο μητρώο με στοιχεία τις τετραγωνικές ρίζες των ιδιοτιμών του πίνακα \mathbf{C} .

Ένα διάνυσμα τυχαίων αριθμών από μια κανονική κατανομή $N(\mathbf{m}, \mathbf{C})$ μπορεί να προκύψει με χρήση της (3.11) από τις παρακάτω σχέσεις:

$$\begin{aligned}
\mathbf{x} &\sim N \mathbf{m}, \mathbf{C} \Rightarrow \\
\mathbf{x} - \mathbf{m} &\sim N \mathbf{0}, \mathbf{C} = \sqrt{\mathbf{C}} N \mathbf{0}, \mathbf{I} \Rightarrow \\
\mathbf{x} - \mathbf{m} &\sim \mathbf{B} \sqrt{\mathbf{L}} \mathbf{B}^T N \mathbf{0}, \mathbf{I} \Rightarrow \\
\mathbf{x} - \mathbf{m} &\sim \mathbf{B} \sqrt{\mathbf{L}} N \mathbf{0}, \mathbf{I}
\end{aligned}$$

Επομένως, ένα διάνυσμα \mathbf{x} τυχαίων αριθμών από μια κανονική κατανομή $N \mathbf{m}, \mathbf{C}$ μπορεί να προκύψει από τη σχέση:

$$\mathbf{x} \sim \mathbf{m} + \mathbf{B} \sqrt{\mathbf{L}} N \mathbf{0}, \mathbf{I} \quad (3.12)$$

όπου \mathbf{I} είναι ο μοναδιαίος $n \times n$ πίνακας, $N \mathbf{0}, \mathbf{I}$ ένα διάνυσμα τυχαίων αριθμών ανεξάρτητων μεταξύ τους, από κανονική κατανομή με μηδενική μέση τιμή και μοναδιαία διασπορά, και $\sqrt{\mathbf{L}}$ είναι το διαγώνιο μητρώο με στοιχεία τις τετραγωνικές ρίζες των ιδιοτιμών του πίνακα \mathbf{C} . Η σχέση (3.12) είναι πολύ εύκολο να υλοποιηθεί σε πρόγραμμα υπολογιστή.

3.2.2 Οι τελεστές επιλογής και ανασυνδυασμού

Ο αλγόριθμος CMA-ES δεν είναι ελιτιστικός, δηλαδή ο τελεστής επιλογής είναι της μορφής μ, λ . Αυτό σημαίνει πως θα επιλεγούν οι μ καλύτεροι απόγονοι της τωρινής γενιάς ως γονείς της επόμενης γενιάς. Η δημιουργία του k απόγονου από ένα σύνολο λ απογόνων στον αλγόριθμο CMA-ES γίνεται από τη σχέση:

$$\mathbf{x}_k^{g+1} \sim N \mathbf{m}^g, \sigma^{g^2} \mathbf{C}^g \sim \mathbf{m}^g + \sigma^g N \mathbf{0}, \mathbf{C}^g \quad (3.13)$$

όπου:

- $\mathbf{x}_k^{g+1} \in \mathcal{R}^n$ είναι ο k απόγονος της γενιάς g , $k = 1, \dots, \lambda$.
- $N \mathbf{m}, \mathbf{C}$ είναι μία κανονική κατανομή με μέση τιμή \mathbf{m} και πίνακα συνδιασποράς \mathbf{C} .
- $\mathbf{m}^g \in \mathcal{R}^n$ είναι το διάνυσμα της μέσης τιμής της κανονικής κατανομής.
- $\sigma^g \in \mathcal{R}_+$ είναι το καθολικό μήκος βήματος.
- $\mathbf{C}^g \in \mathcal{R}^n$ είναι ο πίνακας συνδιασποράς.

Για να οριστεί λοιπόν μια πλήρης γενιά εξέλιξης του αλγόριθμου CMA-ES, θα πρέπει να υπολογιστεί η μέση τιμή \mathbf{m}^{g+1} , το καθολικό μήκος βήματος σ^g και ο πίνακας συνδιασποράς \mathbf{C}^{g+1} .

Το διάνυσμα \mathbf{m}^{g+1} αυτό θα προκύψει όταν ο τελεστής ανασυνδυασμού δράσει στους μ καλύτερους απόγονους της προηγούμενης γενιάς g . Η δράση του τελεστή ανασυνδυασμού δίνεται από τη σχέση:

$$\mathbf{m}^{g+1} = \sum_{i=1}^{\mu} w_i \mathbf{x}_{i:\lambda}^{g+1} \quad (3.14)$$

Το στοιχείο $\mathbf{x}_{i:\lambda}^{g+1}$ συμβολίζει το i -ιοστό καλύτερο άτομο ως προς την αντικειμενική συνάρτηση των λ απογόνων. Προφανώς, στην περίπτωση της ελαχιστοποίησης της αντικειμενικής συνάρτησης ισχύει:

$$F x_{1:\lambda}^{g+1} \leq F x_{2:\lambda}^{g+1} \leq \dots \leq F x_{\lambda:\lambda}^{g+1} \quad (3.15)$$

όπου F η αντικειμενική συνάρτηση. Τονίζεται εδώ, πως στον αλγόριθμο CMA-ES όλα τα άτομα γονέων πάντοτε ανασυνδυάζονται σε ένα άτομο, το οποίο θα παίζει το ρόλο της μέσης τιμής της κανονικής κατανομής, από την οποία θα προκύψουν οι απόγονοι της επόμενης γενιάς.

Ο τελεστής ανασυνδυασμού του αλγόριθμου CMA-ES, μοιάζει με τον ενδιάμεσο τελεστή ανασυνδυασμού του αλγόριθμου A-ES, με μόνη διαφορά πως η μέση τιμή είναι σταθμισμένη με τα βάρη w_1, w_2, \dots, w_μ . Οι τιμές των βαρών w_i θα πρέπει να είναι τέτοιες ώστε να ισχύει:

$$\sum_{i=1}^{\mu} w_i = 1, w_i > 0, i = 1, \dots, \mu \quad (3.16)$$

Επίσης, θα πρέπει να σημειωθεί πως στον αλγόριθμο CMA-ES, η παράμετρος ανασυνδυασμού ρ είναι πάντα ίση με τον αριθμό γονέων μ .

3.2.3 Προσαρμογή του πίνακα συνδιασποράς \mathbf{C}

Ο πίνακας συνδιασποράς μπορεί να εκτιμηθεί ξανά, αφού έχουν δημιουργηθεί οι λ απόγονοι, από τη σχέση:

$$\mathbf{C}_\lambda^{g+1} = \frac{1}{\lambda} \sum_{i=1}^{\lambda} \mathbf{x}_i^{g+1} - \mathbf{m}^g \quad \mathbf{x}_i^{g+1} - \mathbf{m}^g \quad T \quad (3.17)$$

Η παραπάνω σχέση δίνει μια καλή εκτίμηση του πίνακα αν $\lambda \geq 10n$, όπου n το πλήθος των μεταβλητών σχεδιασμού. Η εκτίμηση του πίνακα συνδιασποράς μπορεί να γίνει και αφού δράσει ο τελεστής επιλογής στον πληθυσμό των απογόνων, από τη σχέση:

$$\mathbf{C}_\mu^{g+1} = \sum_{i=1}^{\mu} w_i \mathbf{x}_{i:\lambda}^{g+1} - \mathbf{m}^g \quad \mathbf{x}_{i:\lambda}^{g+1} - \mathbf{m}^g \quad T \quad (3.18)$$

όπου w_i είναι τα ίδια βάρη που χρησιμοποιεί ο τελεστής ανασυνδυασμού. Ο πίνακας \mathbf{C}_λ^{g+1} εκτιμά τον πίνακα συνδιασποράς αφού έχουν δημιουργηθεί οι λ απόγονοι, ενώ ο πίνακας \mathbf{C}_μ^{g+1} τον εκτιμά από τους μ καλύτερους απογόνους, αφού δηλαδή έχει δράσει ο τελεστής επιλογής.

Επειδή γενικά ο πληθυσμός των απογόνων πρέπει να σχετικά μικρός ώστε η έρευνα του βέλτιστου να είναι γρήγορη, από τη σχέση (3.18) δεν είναι δυνατόν να εκτιμηθεί σωστά ο πίνακας συνδιασποράς σε μία μόνον γενιά. Επομένως, πρέπει να χρησιμοποιηθεί πληροφορία από προηγούμενες γενιές. Η μέση τιμή των εκτιμώμενων από τη σχέση (3.18) μητρώων συνδιασποράς είναι αμερόληπτη εκτιμήτρια του πίνακα συνδιασποράς \mathbf{C}^{g+1} και δίνεται από τη σχέση:

$$\mathbf{C}^{g+1} = \frac{1}{g+1} \sum_{i=0}^g \frac{1}{\sigma^{i^2}} \mathbf{C}_\mu^{i+1} \quad (3.19)$$

Στη σχέση (3.19) χρησιμοποιείται και το καθολικό μήκος βήματος σ^i ώστε να γίνουν οι πίνακες \mathbf{C}_μ συγκρίσιμοι από διαφορετικές γενιές.

Επιλέγοντας $\mathbf{C}^0 = \mathbf{I}$ και μια παράμετρο μάθησης c_{cov} τέτοια ώστε $0 < c_{\text{cov}} \leq 1$, είναι εφικτός με χρήση επαναληπτικού σχήματος ο υπολογισμός του πίνακα συνδιασποράς \mathbf{C}^{g+1} από τη σχέση:

$$\mathbf{C}^{g+1} = 1 - c_{\text{cov}} \mathbf{C}^g + c_{\text{cov}} \sum_{i=1}^{\mu} w_i \text{OP} \left(\frac{\mathbf{x}_{i:\lambda}^{g+1} - \mathbf{m}^g}{\sigma^g} \right) \quad (3.20)$$

όπου $\text{OP } \mathbf{y} = \mathbf{y}\mathbf{y}^T$ είναι το εξωτερικό γινόμενο του ορίσματος με τον εαυτό του. Στη σχέση (3.20), αν θέσουμε $c_{\text{cov}} = 0$ τότε δεν γίνεται καμία εκτίμηση του \mathbf{C}^{g+1} , επομένως ισχύει πως $\mathbf{C}^{g+1} = \mathbf{C}^0$. Στην περίπτωση που $c_{\text{cov}} = 1$, δεν διατηρείται καμία πληροφορία από τις προηγούμενες γενιές και ο πίνακας \mathbf{C}^{g+1} δίνεται από τη σχέση:

$$\mathbf{C}^{g+1} = \frac{1}{\sigma^{g^2}} \mathbf{C}_\mu^{g+1}.$$

Η σχέση (3.20) έχει το μειονέκτημα, πως επειδή $\text{OP } \mathbf{y} = \text{OP } -\mathbf{y}$, το πρόσημο του βήματος:

$$\frac{\mathbf{x}_{i:\lambda}^{g+1} - \mathbf{m}^g}{\sigma^g}$$

δεν παίζει ρόλο στον υπολογισμό του πίνακα \mathbf{C}^{g+1} . Για να εξαλειφθεί αυτό το μειονέκτημα, πρέπει να ληφθούν υπόψη τα βήματα εξέλιξης κατά τη διάρκεια αυτής. Έτσι, ορίζεται ως $\mathbf{p}_c^g \in \mathcal{R}^n$ το διάνυσμα πορείας της εξέλιξης, το οποίο δίνεται από το επαναληπτικό σχήμα:

$$\mathbf{p}_c^{g+1} = 1 - c_c \mathbf{p}_c^g + H_\sigma^{g+1} \sqrt{c_c} \frac{\mathbf{m}^{g+1} - \mathbf{m}^g}{\sigma^g} \quad (3.21)$$

όπου c_c είναι η παράμετρος μάθησης του διανύσματος \mathbf{p}_c με όρια τιμών $0 < c_c \leq 1$. Η παράμετρος H_σ^{g+1} σταματάει την ανανέωση του \mathbf{p}_c αν η ευκλείδια νόρμα $\|\mathbf{p}_\sigma^{g+1}\|$ του διανύσματος συζευγμένης πορείας της εξέλιξης \mathbf{p}_σ (βλ. παρακάτω) είναι πολύ μεγάλη. Η τιμή της H_σ^{g+1} δίνεται από τη σχέση:

$$H_\sigma^{g+1} = \begin{cases} 1, & \text{if } \frac{\|\mathbf{p}_\sigma^{g+1}\|}{\sqrt{1 - 1 - c_\sigma^{2g+1}}} < \left(1.5 + \frac{1}{n - 0.5}\right) E\|N \mathbf{0}, \mathbf{I}\| \\ 0, & \text{otherwise} \end{cases} \quad (3.22)$$

όπου $E\|N \mathbf{0}, \mathbf{I}\|$ είναι η αναμενόμενη τιμή της ευκλείδιας νόρμας ενός διανύσματος με στοιχεία τυχαίους αριθμούς από κανονική κατανομή με μηδενική μέση τιμή και μοναδιαία διασπορά.

Το αρχικό διάνυσμα πορείας της εξέλιξης είναι $\mathbf{p}_c^0 = \mathbf{0}$. Η παράμετρος μ_{eff} δίνεται από τη σχέση:

$$\mu_{eff} = \left(\sum_{i=1}^{\mu} w_i^2 \right)^{-1} \quad (3.23)$$

όπου w_i είναι τα βάρη που χρησιμοποιεί ο τελεστής ανασυνδυασμού.

Ο υπολογισμός του πίνακα συνδιασποράς \mathbf{C}^{g+1} , με χρήση του διανύσματος πορείας της εξέλιξης, δίνεται από τη σχέση:

$$\mathbf{C}^{g+1} = 1 - c_{cov} \mathbf{C}^g + c_{cov} \mathbf{p}_c^{g+1} \mathbf{p}_c^{g+1 T} \quad (3.24)$$

Οι σχέσεις (3.20) και (3.24), χρησιμοποιώντας μια παράμετρο μ_{cov} η οποία καθορίζει τη σχετική τους βαρύτητα, δίνουν την τελική σχέση υπολογισμού του πίνακα συνδιασποράς \mathbf{C}^{g+1} :

$$\begin{aligned} \mathbf{C}^{g+1} = & 1 - c_{cov} \mathbf{C}^g + \frac{c_{cov}}{\mu_{cov}} \mathbf{p}_c^{g+1} \mathbf{p}_c^{g+1 T} + \\ & + c_{cov} \left(1 - \frac{1}{\mu_{cov}} \right) \sum_{i=1}^{\mu} w_i \text{OP} \left(\frac{\mathbf{x}_{i:\lambda}^{g+1} - \mathbf{m}^g}{\sigma^g} \right) \end{aligned} \quad (3.25)$$

Η σχέση (3.25) δίνει την (3.24) αν $\mu_{cov} = 1$ και την (3.20) αν $\mu_{cov} = \infty$. Η καλύτερη επιλογή είναι $\mu_{cov} = \mu_{eff}$.

Ο υπολογισμός του πίνακα συνδιασποράς, ισοδυναμεί με τον υπολογισμό του αντίστροφου του Hessian μητρώου, που είναι απαραίτητος στις αιτιοκρατικές quasi-Newton μεθόδους.

3.2.4 Προσαρμογή του μήκους βήματος σ

Το καθολικό μήκος βήματος σ χρησιμοποιείται από τον αλγόριθμο CMA-ES για να επιταχύνει τη διαδικασία της εύρεσης του βέλτιστου. Για τον έλεγχο της τιμής του καθολικού μήκους βήματος σ , χρησιμοποιείται το διάνυσμα συζευγμένης πορείας της εξέλιξης, το οποίο δίνεται από το επαναληπτικό σχήμα:

$$\mathbf{p}_\sigma^{g+1} = 1 - c_\sigma \mathbf{p}_\sigma^g + \sqrt{c_\sigma} \frac{2 - c_\sigma}{\mu_{eff}} \mathbf{C}^g \frac{\mathbf{m}^{g+1} - \mathbf{m}^g}{\sigma^g} \quad (3.26)$$

όπου \mathbf{p}_σ^{g+1} το διάνυσμα συζευγμένης πορείας της εξέλιξης, και c_σ μία παράμετρος μάθησης με πεδίο τιμών $0 < c_\sigma \leq 1$. Ο πίνακας \mathbf{C}^g δίνεται από τη σχέση:

$$\mathbf{C}^{g \frac{1}{2}} = \mathbf{B}^g \mathbf{D}^{g-1} \mathbf{B}^{g T} \quad (3.27)$$

Η σχέση (3.27) προκύπτει, αν θεωρηθεί πως η παραγοντοποίηση του πίνακα \mathbf{C}^g δίνεται από τη σχέση:

$$\mathbf{C}^g = \mathbf{B}^g \mathbf{D}^{g^2} \mathbf{B}^{g T} \quad (3.28)$$

όπου \mathbf{B}^g είναι ο πίνακας με τα ορθοκανονικά ιδιοδιανύσματα του πίνακα \mathbf{C}^g , ενώ ο πίνακας \mathbf{D}^g είναι ένας διαγώνιος πίνακας με στοιχεία διαγωνίου τις τετραγωνικές ρίζες των ιδιοτιμών του πίνακα \mathbf{C}^g .

Το καθολικό μήκος βήματος σ^{g+1} δίνεται από τη σχέση:

$$\sigma^{g+1} = \sigma^g \exp \left(\frac{c_\sigma}{d_\sigma} \left(\frac{\|\mathbf{P}_\sigma^{g+1}\|}{E\|\mathbf{N} \mathbf{0}, \mathbf{I}\|} - 1 \right) \right) \quad (3.29)$$

όπου d_σ είναι μια παράμετρος απόσβεσης με τιμή κοντά στη μονάδα, και $E\|\mathbf{N} \mathbf{0}, \mathbf{I}\|$ είναι η αναμενόμενη τιμή της ευκλείδειας νόρμας ενός διανύσματος με στοιχεία τυχαίους αριθμούς από κανονική κατανομή με μηδενική μέση τιμή και μοναδιαία διασπορά.

3.2.5 Περιγραφή της ροής του αλγόριθμου $(\mu/\mu, \lambda)$ -CMA-ES

Σε αυτήν την παράγραφο, θα δώσουμε περιληπτικά τη ροή του αλγόριθμου $\mu/\mu, \lambda$ -CMA-ES. Έστω ένα πρόβλημα βέλτιστου σχεδιασμού με n μεταβλητές σχεδιασμού. Οι προεπιλεγμένες τιμές των εξωγενών παραμέτρων στρατηγικής $\lambda, w_i, c_\sigma, d_\sigma, c_c, \mu_{\text{cov}}, c_{\text{cov}}$ δίνονται στις παρακάτω εξισώσεις:

$$\lambda = 4 + 3 \ln n, \quad \mu = \frac{\lambda}{2} \quad (3.30)$$

$$w_i = \frac{\ln \mu + 1 - \ln i}{\sum_{j=1}^{\mu} \ln \mu + 1 - \ln j}, \quad i = 1, \dots, \mu \quad (3.31)$$

Οι σχέσεις (3.30) και (3.31) αφορούν τις εξωγενείς παραμέτρους στρατηγικής των τελεστών επιλογής και ανασυνδυασμού. Στις σχέσεις (3.32) και (3.33) δίνονται οι παράμετροι ελέγχου του καθολικού μήκους βήματος:

$$c_\sigma = \frac{\mu_{\text{eff}} + 2}{n + \mu_{\text{eff}+3}} \quad (3.32)$$

$$d_\sigma = 1 + 2 \max \left(0, \sqrt{\frac{\mu_{\text{eff}-1}}{n+1}} - 1 \right) + c_\sigma \quad (3.33)$$

όπου η παράμετρος μ_{eff} δίνεται από τη σχέση (3.23). Οι επόμενες σχέσεις αφορούν τις προεπιλεγμένες τιμές των παραμέτρων που αφορούν την προσαρμογή του πίνακα συνδιασποράς:

$$c_c = \frac{4}{4+n}, \quad \mu_{\text{cov}} = \mu_{\text{eff}} \quad (3.34)$$

$$c_{\text{cov}} = \frac{1}{\mu_{\text{cov}}} \frac{2}{n + \sqrt{2}} + \left(1 - \frac{1}{\mu_{\text{cov}}} \right) \min \left(1, \frac{2\mu_{\text{eff}} - 1}{n + 2 + \mu_{\text{eff}}} \right) \quad (3.35)$$

Τα βήματα του αλγόριθμου $\mu/\mu, \lambda$ -CMA-ES είναι τα εξής:

1. *Εναρκτήριο βήμα:* Αρχικοποίηση των διανυσμάτων $\mathbf{p}_c^0 = \mathbf{0}, \mathbf{p}_\sigma^0 = \mathbf{0}$, του πίνακα συνδιασποράς $\mathbf{C}^0 = \mathbf{0}$, του καθολικού μήκους βήματος σ^0 και της μέσης τιμής \mathbf{m}^0 της κανονικής κατανομής. Οι αρχικές τιμές των σ^0 και \mathbf{m}^0 εξαρτώνται από το πρόβλημα.
2. Δημιουργία λ απογόνων με χρήση της σχέσης (3.13).
3. Υπολογισμός της νέας μέσης τιμής \mathbf{m}^{g+1} από τη σχέση (3.14).
4. Υπολογισμός του νέου διανύσματος συζευγμένης πορείας της εξέλιξης \mathbf{P}_σ^{g+1} από τη σχέση (3.26).
5. Υπολογισμός του νέου καθολικού βήματος σ^{g+1} από τη σχέση (3.29). Η $E\|\mathbf{N} \mathbf{0}, \mathbf{I}\|$ δίνεται από τη σχέση:

$$E\|\mathbf{N} \mathbf{0}, \mathbf{I}\| \approx \sqrt{n} \left(1 - \frac{1}{4n} + \frac{1}{21n^2} \right)$$

όπου n το πλήθος των μεταβλητών σχεδιασμού.

6. Υπολογισμός του νέου διανύσματος πορείας της εξέλιξης \mathbf{P}_c^{g+1} από τη σχέση (3.26).
7. Υπολογισμός του νέου πίνακα συνδιασποράς \mathbf{C}^{g+1} από τη σχέση (3.25).
8. Έλεγχος κριτηρίου τερματισμού. Αν το κριτήριο τερματισμού ικανοποιείται, ο αλγόριθμος σταματάει. Διαφορετικά, επιστρέφουμε στο βήμα 2.

Ως κριτήρια τερματισμού του αλγόριθμου μπορούν να χρησιμοποιηθούν τα κριτήρια τερματισμού που αναφέρθηκαν στην περίπτωση του αλγόριθμου A-ES.

Από τις εξωγενείς παραμέτρους στρατηγικής, το πλήθος των απογόνων λ μπορεί να επιλεγεί μεγαλύτερο από την τιμή που δίνει η εξίσωση (3.30). Οι υπόλοιπες παράμετροι θα προκύψουν από την τιμή του λ . Η αύξηση του πληθυσμού των απογόνων, αυξάνει την ικανότητα έρευνας του αλγόριθμου αλλά έχει ως αποτέλεσμα τη μείωση της ταχύτητας εκτέλεσης του αλγόριθμου. Στη χειρότερη περίπτωση πάντως, η ταχύτητα μειώνεται γραμμικά με την αύξηση του πληθυσμού των απογόνων λ .

3.2.6 Πλεονεκτήματα-μειονεκτήματα του αλγόριθμου CMA-ES

Ο αλγόριθμος CMA-ES υπερνικά τα τυπικά προβλήματα που αντιμετωπίζουν οι περισσότεροι εξελικτικοί αλγόριθμοι, όπως:

1. Την κακή απόδοση σε μή-διαχωρίσιμες (non-separable) αντικειμενικές συναρτήσεις.
2. Την ανάγκη χρήση μεγάλων πληθυσμών. Η χρήση της παραμέτρου μάθησης c_{cov} έχει ως αποτέλεσμα την αξιόπιστη απόδοση του αλγόριθμου ακόμη και με μικρούς πληθυσμούς απογόνων (πχ, $\lambda = 10$).
3. Την πρόωρη σύγκλιση σε σημείο που δεν αποτελεί ολικό ή τοπικό ελάχιστο. Αυτό επιτυγχάνεται με τη χρήση του καθολικού μήκους βήματος σ .

Μία πολύ σημαντική ιδιότητα που είναι θεμιτό να έχει ένας αλγόριθμος βελτιστοποίησης, είναι αυτή της σταθερότητας (invariance). Έστω μια συνάρτηση f τέτοια ώστε $f: \mathbb{R}^n \rightarrow \mathbb{R}, \mathbf{x} \mapsto f(\mathbf{x})$ και $\mathbf{x}^0 = \mathbf{a} \in \mathbb{R}^n$. Τότε για παράδειγμα, αν ένας αλγόριθμος χαρακτηρίζεται από σταθερότητα κλίμακας (scale invariance), η απόδοσή του θα είναι η ίδια σε μια άλλη αντικειμενική συνάρτηση της μορφής $f_a: \mathbb{R}^n \rightarrow \mathbb{R}, \mathbf{x} \mapsto f(a \cdot \mathbf{x})$ με $\mathbf{x}^0 = -\mathbf{a} \in \mathbb{R}^n$ και $a \neq 0$. Ο αλγόριθμος CMA-ES χαρακτηρίζεται από τις παρακάτω μορφές σταθερότητας:

- Σταθερότητα ως προς μονότονους μετασχηματισμούς της αντικειμενικής συνάρτησης.
- Σταθερότητα ως προς τους μετασχηματισμούς περιστροφής, αντανάκλασης και μετατόπισης του χώρου έρευνας, αν μετασχηματιστεί αναλόγως και το αρχικό σημείο έρευνας \mathbf{m}^0 .
- Σταθερότητα ως προς αλλαγές της κλίμακας, αν αυτές οι αλλαγές εφαρμοστούν στο καθολικό βήμα σ^0 και στο αρχικό σημείο έρευνας \mathbf{m}^0 .
- Σταθερότητα ως προς κάθε αντιστρεπτό μετασχηματισμό \mathbf{A} του χώρου έρευνας, αν ο αρχικός πίνακας συνδιασποράς μετασχηματιστεί από τη σχέση $\mathbf{C}^0 = \mathbf{A}^{-1} \mathbf{A}^{-1}$ όπως και το αρχικό σημείο έρευνας \mathbf{m}^0 .

Ως μειονεκτήματα του αλγόριθμου CMA-ES θεωρείται η απώλεια αριθμητικής ακρίβειας και η εμφάνιση προβλημάτων αριθμητικής αστάθειας κατά τη διάρκεια της εξέλιξης. Ο αριθμός κατάστασης (condition number) του πίνακα συνδιασποράς \mathbf{C} αυξάνει συνεχώς και αυτό έχει ως αποτέλεσμα η έρευνα για το βέλτιστο να σταματήσει πρόωρα εξαιτίας αριθμητικής αστάθειας. Για να αποφευχθούν αριθμητικά σφάλματα, η μέγιστη δυνατή τιμή του αριθμού κατάστασης του πίνακα \mathbf{C} περιορίζεται σε 10^{14} . Σε περίπτωση που ο αριθμός κατάστασης ξεπεράσει τη μέγιστη επιτρεπόμενη τιμή 10^{14} , τότε ο πίνακας \mathbf{C} μετασχηματίζεται από τη σχέση $\mathbf{C} \leftarrow \mathbf{C} + \max \lambda_i / 10^{14} - \min \lambda_i \mathbf{I}$ όπου $\max \lambda_i, \min \lambda_i$ είναι η μέγιστη και η ελάχιστη ιδιοτιμή του πίνακα \mathbf{C} αντίστοιχα.

Μέριμνα επίσης θα πρέπει να ληφθεί, ώστε να υπάρχει μια ελάχιστη επιτρεπόμενη δύναμη μετάλλαξης. Αυτό μπορεί να επιτευχθεί ορίζοντας ένα ελάχιστο όριο για το γινόμενο $\sigma \sqrt{\min \lambda_i}$ όπου σ το καθολικό μήκος βήματος και $\sigma \sqrt{\min \lambda_i}$ η τετραγωνική ρίζα της ελάχιστης ιδιοτιμής του πίνακα συνδιασποράς \mathbf{C} . Η τιμή του ορίου αυτού εξαρτάται από το εκάστοτε πρόβλημα.

Ένα άλλο βασικό μειονέκτημα του αλγόριθμου CMA-ES είναι η πολυπλοκότητά του η οποία είναι της τάξης $O(n^3)$ όπου n το πλήθος των μεταβλητών σχεδιασμού. Η πολυπλοκότητα μπορεί να μειωθεί σε $O(n^{2.5})$ ή σε $O(n^2)$, αν ο υπολογισμός των πινάκων $\mathbf{B}^g, \mathbf{D}^g$ (βλ, σχέση (3.28)) γίνεται κάθε \sqrt{n} ή $n/10$ γενιές.

3.3 Ο αλγόριθμος E-CMA-ES

Ο αλγόριθμος E-CMA-ES προτάθηκε από τους Hansen et al και αποτελεί την ελιτιστική εκδοχή του αρχικού αλγόριθμου CMA-ES. Στον αλγόριθμο E-CMA-ES, ο τελεστής επιλογής είναι πάντα της μορφής $1 + \lambda$ όπου λ είναι ο αριθμός των απογόνων που προκύπτουν από έναν και μοναδικό γονέα ($\mu = 1$).

Στον αλγόριθμο $1 + \lambda$ -E-CMA-ES, κάθε άτομο a αποτελείται από ένα σύνολο πέντε στοιχείων, δηλαδή:

$$a = \mathbf{x}, \bar{p}_{succ}, \sigma, \mathbf{p}_c, \mathbf{C}$$

όπου \mathbf{x} είναι το διάνυσμα μεταβλητών σχεδιασμού, \bar{p}_{succ} είναι μια παράμετρος η οποία δηλώνει τον ρυθμό επιτυχιών κατά τη διάρκεια της έρευνας (βλ. παρακάτω), σ είναι το μήκος βήματος, \mathbf{p}_c είναι το διάνυσμα πορείας της εξέλιξης και \mathbf{C} είναι ο πίνακας συνδιασποράς. Σε αντίθεση με τον αλγόριθμο CMA-ES, στον αλγόριθμο E-CMA-ES κάθε άτομο έχει το δικό του μήκος βήματος σ , το δικό του διάνυσμα πορείας της εξέλιξης \mathbf{p}_c και το δικό πίνακα συνδιασποράς \mathbf{C} .

Στο Σχήμα 3.3 δίνεται βασική υπορουτίνα του αλγόριθμου $1 + \lambda$ -E-CMA-ES σε μορφή ψευδοκώδικα.

Algorithm: $1 + \lambda$ -E-CMA-ES

```

1  $\mathbf{g} = \mathbf{0}$ , initialize  $a_{parent}^g$ 
2 repeat
3    $a_{parent}^{g+1} \leftarrow a_{parent}^g$ 
4   for  $k = 1, \dots, \lambda$  do
5      $\mathbf{x}_k^{g+1} \sim N(\mathbf{x}_{parent}^g, \sigma^{g^2} \mathbf{C}^g)$ 
6   end do
7   UpdateStepSize  $\left( a_{parent}^{g+1}, \frac{\lambda_{succ}^{g+1}}{\lambda} \right)$ 
8   if  $f(\mathbf{x}_{1:\lambda}^{g+1}) < f(\mathbf{x}_{parent}^g)$  then
9      $\mathbf{x}_{parent}^{g+1} \leftarrow \mathbf{x}_{1:\lambda}^{g+1}$ 
10    UpdateCovariance  $\left( a_{parent}^{g+1}, \frac{\mathbf{x}_{parent}^{g+1} - \mathbf{x}_{parent}^g}{\sigma_{parent}^g} \right)$ 
11  end if
12 until stopping criterion is met
```

Σχήμα 3.3: Ψευδοκώδικας του αλγόριθμου E-CMA-ES

Στη βασική ρουτίνα, αρχικά δημιουργείται ο γονέας a_{parent}^g από τον οποίο θα ξεκινήσει η διαδικασία της εξέλιξης. Στις γραμμές #4-6 δημιουργούνται οι λ απόγονοι της γενιάς g . Για τη δημιουργία των απογόνων χρησιμοποιείται η σχέση (3.13) της προηγούμενης παραγράφου με $\mathbf{m}^g = \mathbf{x}_{parent}^g$.

Αφού δημιουργηθούν οι απόγονοι, καλείται η υπορουτίνα UpdateStepSize με ορίσματα τον γονέα a_{parent}^{g+1} και το ποσοστό των επιτυχημένων απογόνων:

$$\frac{\lambda_{succ}^{g+1}}{\lambda},$$

όπου λ_{succ}^{g+1} είναι ο αριθμός των απογόνων με τιμή αντικειμενικής συνάρτησης μικρότερη του γονέα a_{parent} .

Στο Σχήμα 3.4 δίνεται ο ψευδοκώδικας της υπορουτίνας UpdateStepSize. Ο υπολογισμός του νέου μήκους βήματος βασίζεται στον κανόνα επιτυχιών 1/5 [1]. Σύμφωνα με τον κανόνα αυτόν, το μήκος βήματος θα πρέπει να αυξάνεται αν ο ρυθμός επιτυχιών:

$$p_{succ} = \frac{\lambda_{succ}}{\lambda}$$

είναι μεγάλος, ενώ θα πρέπει να μειώνεται όταν αυτός είναι μικρός.

Procedure: UpdateStepSize $a = \mathbf{x}, \bar{p}_{succ}, \sigma, \mathbf{p}_c, \mathbf{C}, p_{succ}$

- 1 $\bar{p}_{succ} \leftarrow 1 - c_p \bar{p}_{succ} + c_p p_{succ}$
 - 2 $\sigma \leftarrow \sigma \exp\left(\frac{1}{d} \left(\bar{p}_{succ} - \frac{p_{succ}^{target}}{1 - p_{succ}^{target}} (1 - \bar{p}_{succ}) \right)\right)$
-

Σχήμα 3.4: Ψευδοκώδικας της υπορουτίνας UpdateStepSize

Η παράμετρος p_{succ}^{target} είναι μια προεπιλεγμένη εξωγενής παράμετρος στρατηγικής για την οποία θα γίνει αναφορά στη συνέχεια.

Αν ο καλύτερος απόγονος έχει μικρότερη τιμή αντικειμενικής συνάρτησης (γραμμή #8) από τον γονέα, τότε τίθεται αυτός ως γονέας της επόμενης γενιάς (γραμμή #9), και ταυτόχρονα υπολογίζεται ένας νέος πίνακας συνδιασποράς από την υπορουτίνα UpdateCovariance με ορίσματα τον γονέα a_{parent}^{g+1} και το βήμα:

$$\frac{x_{parent}^{g+1} - x_{parent}^g}{\sigma_{parent}^g}$$

όπως φαίνεται από τον ψευδοκώδικα που δίνεται στο Σχήμα 3.5.

Procedure: UpdateCovariance $a = \mathbf{x}, \bar{p}_{succ}, \sigma, \mathbf{p}_c, \mathbf{C}, \mathbf{x}_{step} \in \mathbb{R}^n$

- 1 **if** $\bar{p}_{succ} < p_{thresh}$ **then**
 - 2 $\mathbf{p}_c \leftarrow 1 - c_c \mathbf{p}_c + \sqrt{c_c} \cdot 2 - c_c \mathbf{x}_{step}$
 - 3 $\mathbf{C} \leftarrow 1 - c_{cov} \mathbf{C} + c_{cov} \mathbf{p}_c \mathbf{p}_c^T$
 - 4 **else**
 - 5 $\mathbf{p}_c \leftarrow 1 - c_c \mathbf{p}_c$
 - 6 $\mathbf{C} \leftarrow 1 - c_{cov} \mathbf{C} + c_{cov} \mathbf{p}_c \mathbf{p}_c^T + c_c \cdot 2 - c_c \mathbf{C}$
 - 7 **end if**
-

Σχήμα 3.5: Ψευδοκώδικας της υπορουτίνας UpdateCovariance

Όπως φαίνεται από την υπορουτίνα UpdateCovariance, ο υπολογισμός του νέου διανύσματος πορείας της εξέλιξης και του νέου πίνακα συνδιασποράς εξαρτάται από το ρυθμό επιτυχιών:

$$p_{succ} = \frac{\lambda_{succ}}{\lambda}$$

Αν $\bar{p}_{succ} < p_{thresh}$, τότε αρχικά υπολογίζεται το νέο διάνυσμα πορείας στη γραμμή #3 της υπορουτίνας, και στη συνέχεια στη γραμμή #4 υπολογίζεται ο νέος πίνακας συνδιασποράς. Αντίθετα, αν $\bar{p}_{succ} \geq p_{thresh}$, το βήμα:

$$\frac{x_{parent}^{g+1} - x_{parent}^g}{\sigma_{parent}^g}$$

δεν λαμβάνεται υπόψιν στον υπολογισμό του νέου διανύσματος \mathbf{p}_c .

Ο αριθμός απογόνων λ επιλέγεται από τον χρήστη. Οι προεπιλεγμένες τιμές των υπόλοιπων εξωγενών παραμέτρων στρατηγικής, δίνονται από τις παρακάτω σχέσεις:

$$d = 1 + \frac{n}{2\lambda} \quad (3.36)$$

$$p_{succ}^{target} = \frac{1}{5 + \frac{\sqrt{\lambda}}{2}} \quad (3.37)$$

$$c_p = \frac{p_{succ}^{target} \lambda}{2 + p_{succ}^{target} \lambda} \quad (3.38)$$

όπου d είναι η παράμετρος απόσβεσης στην υπορουτίνα UpdateStepSize, n είναι το πλήθος των μεταβλητών σχεδίασης, c_p είναι η παράμετρος εξομάλυνσης του \bar{p}_{succ} στην υπορουτίνα UpdateStepSize, και p_{succ}^{target} μια σταθερά που εξαρτάται μόνον από το πλήθος των απογόνων. Οι σχέσεις (3.36) ως (3.38) αφορούν παραμέτρους που σχετίζονται με τον υπολογισμό του μήκους βήματος σ .

Οι προεπιλεγμένες τιμές των παραμέτρων που σχετίζονται με τον υπολογισμό του πίνακα συνδιασποράς \mathbf{C} δίνονται από τις σχέσεις:

$$c_c = \frac{2}{n+2} \quad (3.39)$$

$$c_{cov} = \frac{2}{n^2+6} \quad (3.40)$$

$$p_{thresh} = 0.44 \quad (3.41)$$

Τα πεδία τιμών των παραμέτρων μάθησης c_c, c_{cov} του πίνακα συνδιασποράς είναι $0 \leq c_{cov} < c_c \leq 1$.

Κατά την αρχικοποίηση του αλγόριθμου $1 + \lambda$ -E-CMA-ES, στα στοιχεία του αρχικού γονέα δίδονται οι αρχικές τιμές $\mathbf{C} = \mathbf{I}$, $\mathbf{p}_c = \mathbf{0}$ και $\bar{p}_{succ} = p_{succ}^{target}$.

Ο ελκυστικός αλγόριθμος $1 + \lambda$ -E-CMA-ES συγκρινόμενος με τον αλγόριθμο $\mu/\mu, \lambda$ -CMA-ES είναι γρηγορότερος όσον αφορά τον απαιτούμενο υπολογιστικό χρόνο για την εξέλιξη. Αντίθετα, ο αλγόριθμος $\mu/\mu, \lambda$ -CMA-ES είναι λιγότερο ευαίσθητος σε τοπικά ελάχιστα και έχει μια πιο σθεναρή απόδοση. Θα πρέπει να σημειωθεί πως ο αλγόριθμος $1 + \lambda$ -E-CMA-ES διατηρεί όλα τα πλεονεκτήματα και μειονεκτήματα του αλγόριθμου CMA-ES (βλ. Παράγραφο 3.2.5).

Ο βασικός λόγος που δημιουργήθηκε ο αλγόριθμος $1 + \lambda$ -E-CMA-ES δεν είναι τόσο για να αντιμετωπίσει προβλήματα βέλτιστου σχεδιασμού με μία αντικειμενική συνάρτηση, αλλά να ικανοποιήσει την απαίτηση για ελιτιστικούς αλγόριθμους σε προβλήματα βελτιστοποίησης με πολλές αντικειμενικές συναρτήσεις (βλ. Κεφάλαιο 5).

4 ΒΕΛΤΙΣΤΟΠΟΙΗΣΗ ΠΟΛΛΑΠΛΩΝ ΚΡΙΤΗΡΙΩΝ

4.1 Εισαγωγή

Η ορθότητα της μαθηματικής διατύπωσης του προβλήματος του βέλτιστου σχεδιασμού, όπως αυτή δόθηκε στο πρώτο κεφάλαιο του παραδοτέου Π1.1, συνήθως απέχει αρκετά από την περιγραφή του προς επίλυση προβλήματος. Σπάνια το πρόβλημα μπορεί να περιγραφεί ορθά από μια αντικειμενική συνάρτηση και πολλές συναρτήσεις περιορισμών. Στη συνηθέστερη περίπτωση υπάρχουν περισσότερες από μια αντικειμενικές συναρτήσεις οι οποίες αντιπαλεύονται η μία την άλλη.

Στο βέλτιστο σχεδιασμό των κατασκευών, το σύνηθες κριτήριο ως προς το οποίο γίνεται η βελτιστοποίηση είναι το βάρος μιας κατασκευής. Στην πραγματικότητα όμως, το βάρος μιας κατασκευής καθορίζει μέχρι ένα ποσοστό το πόσο βέλτιστος είναι ένας σχεδιασμός. Για να βρεθεί ο πραγματικά βέλτιστος σχεδιασμός, ο μηχανικός θα πρέπει να λάβει υπόψιν του πολλά αλληλοσυγκρουόμενα κριτήρια, και στο τέλος να αναζητήσει έναν καλό συμβιβασμό ανάμεσά τους. Καθώς όμως ο αριθμός των αντικειμενικών συναρτήσεων αυξάνει, ώστε να μοντελοποιηθεί ένα πρόβλημα καλύτερα, αυξάνει ταυτόχρονα και η δυσκολία επίλυσης του προβλήματος και γίνεται δυσκολότερη η οπτικοποίηση και η εκτίμηση των αποτελεσμάτων. Είναι στην κρίση του μηχανικού ποια κριτήρια θα επιλέξει ώστε από τη μια η μοντελοποίηση του προβλήματος να είναι σωστή και ρεαλιστική, και από την άλλη η επίλυσή του να είναι δυνατή.

4.2 Βέλτιστος σχεδιασμός κατά Pareto

Στη βελτιστοποίηση πολλαπλών κριτηρίων, πρέπει να ελαχιστοποιηθούν m αντικειμενικές συναρτήσεις, οι οποίες υπόκεινται σε περιορισμούς. Η μαθηματική διατύπωση του προβλήματος βελτιστοποίησης πολλαπλών κριτηρίων είναι η εξής:

$$\begin{aligned} & \text{minimize } \mathbf{F} \mathbf{x} \\ & \text{subject to: } h_i \mathbf{x} = 0, \quad i = 1, \dots, p \\ & \quad \quad \quad g_j \mathbf{x} \leq 0, \quad j = 1, \dots, r \\ & \quad \quad \quad \mathbf{x}_l \leq \mathbf{x} \leq \mathbf{x}_u \end{aligned} \tag{4.1}$$

όπου $\mathbf{x} \in \mathbb{R}^n$ είναι το διάνυσμα των μεταβλητών σχεδιασμού, $h_i \mathbf{x}$ είναι η i συνάρτηση ισοτικών περιορισμών από ένα σύνολο p ισοτικών περιορισμών, $g_j \mathbf{x}$ είναι η j συνάρτηση ανισοτικών περιορισμών από ένα σύνολο r ανισοτικών περιορισμών και $\mathbf{x}_l, \mathbf{x}_u$ είναι τα κάτω και άνω όρια τιμών του διανύσματος σχεδιασμού. Επειδή η $\mathbf{F} \mathbf{x}$ είναι ένα διάνυσμα συναρτήσεων και τα στοιχεία του είναι αλληλοσυγκρουόμενα, δεν υπάρχει μοναδική λύση στο πρόβλημα.

Το εφικτό πεδίο τιμών σε ένα πρόβλημα βελτιστοποίησης με πολλαπλά κριτήρια, καθορίζεται όπως και στη βελτιστοποίηση με ένα κριτήριο, από τους ανισοτικούς και ισοτικούς περιορισμούς. Αν S είναι το σύνολο των εφικτών λύσεων, αυτό ορίζεται ως εξής:

$$S = \{ \mathbf{s} \in \mathbb{R}^n \mid \mathbf{g} \mathbf{s} \leq 0, \mathbf{h} \mathbf{s} = 0 \} \tag{4.2}$$

όπου \mathbf{s} είναι όλοι οι εφικτοί σχεδιασμοί, οι σχεδιασμοί δηλαδή που δεν παραβιάζουν τους ανισοτικούς και ισοτικούς περιορισμούς.

Η απεικόνιση του συνόλου S στο χώρο V των κριτηρίων, δίνεται από τη σχέση:

$$V = \{ \mathbf{z} \in \mathbb{R}^m \mid \mathbf{z} = \mathbf{F} \mathbf{s}, \mathbf{s} \in S \} \tag{4.3}$$

Ο χώρος V είναι το σύνολο τιμών των κριτηρίων που αντιστοιχούν σε βέλτιστους σχεδιασμούς. Επειδή η έννοια του μοναδικού βέλτιστου σχεδιασμού δεν υπάρχει σε προβλήματα βελτιστοποίησης πολλαπλών αντικειμενικών συναρτήσεων, χρησιμοποιείται η έννοια της κατά Pareto βέλτιστης λύσης. Ο V-Pareto (1848-1923), ένας Ιταλός οικονομολόγος, ήταν ο πρώτος που ασχολήθηκε με τη βελτιστοποίηση πολλών αντικειμενικών συναρτήσεων σε οικονομικά προβλήματα.

Παρακάτω, δίνονται τρεις βασικοί ορισμοί στα προβλήματα βέλτιστου σχεδιασμού με πολλές αντικειμενικές συναρτήσεις.

Ορισμός 1: Έστω δύο διανύσματα σχεδιασμού $\mathbf{a}, \mathbf{b} \in S$, όπου S είναι ο χώρος των εφικτών λύσεων, και το πρόβλημα βέλτιστου σχεδιασμού όπως περιγράφεται από τη σχέση (4.1). Το διάνυσμα σχεδιασμού \mathbf{a} λέγεται πως υπερಿಸχύει του διανύσματος σχεδιασμού \mathbf{b} , αν ισχύει:

$$\begin{aligned} \forall i \in 1, \dots, n : \mathbf{F}_i \mathbf{a} \leq \mathbf{F}_i \mathbf{b} \quad \wedge \\ \exists j \in 1, \dots, n : \mathbf{F}_j \mathbf{a} < \mathbf{F}_j \mathbf{b} \end{aligned} \quad (4.4)$$

όπου n είναι το πλήθος των μεταβλητών σχεδιασμού και \mathbf{F}_i είναι η i αντικειμενική συνάρτηση. Αν το διάνυσμα σχεδιασμού \mathbf{a} υπερισχύει του διανύσματος σχεδιασμού \mathbf{b} , αυτό συμβολίζεται με $\mathbf{a} \prec \mathbf{b}$.

Ορισμός 2: Έστω $\mathbf{a} \in S$, ένας εφικτός σχεδιασμός. Ο σχεδιασμός \mathbf{a} , αποκαλείται επικρατών (non-dominated) στο χώρο $S' \subseteq S$, αν και μόνον αν δεν υπάρχει κανένας εφικτός σχεδιασμός $\mathbf{b} \in S'$ τέτοιος ώστε $\mathbf{b} \prec \mathbf{a}$. Δηλαδή:

$$\nexists \mathbf{b} \in S' : \mathbf{b} \prec \mathbf{a} \quad (4.5)$$

Το διάνυσμα σχεδιασμού \mathbf{a} ονομάζεται βέλτιστο κατά Pareto στον χώρο S , αν και μόνον αν δεν υπάρχει άλλος εφικτός σχεδιασμός ο οποίος να επικρατεί του \mathbf{a} .

Οι Pareto βέλτιστοι σχεδιασμοί, είναι αδύνατον να βελτιωθούν ως προς κάποιο κριτήριο, χωρίς να χειροτερέψουν ως προς κάποιο άλλο κριτήριο. Με άλλα λόγια, όλοι οι κατά Pareto βέλτιστοι σχεδιασμοί είναι ισοδύναμοι. Όπως και στα προβλήματα με μία αντικειμενική συνάρτηση υπάρχουν τοπικά και ολικά βέλτιστα, έτσι και στα προβλήματα με πολλές αντικειμενικές συναρτήσεις, είναι δυνατόν ένας σχεδιασμός να ανήκει σε έναν χώρο που είναι τοπικά ή ολικά βέλτιστος κατά Pareto.

Ορισμός 3: Έστω το υποσύνολο των εφικτών λύσεων $S' \subseteq S$. Το υποσύνολο S' ονομάζεται τοπικά βέλτιστο κατά Pareto, αν και μόνον αν ισχύει:

$$\forall \mathbf{a}' \in S' : \nexists \mathbf{a} \in S : \mathbf{a} \prec \mathbf{a}' \wedge \|\mathbf{a} - \mathbf{a}'\| < \varepsilon \wedge \|\mathbf{F} \mathbf{a} - \mathbf{F} \mathbf{a}'\| < \delta \quad (4.6)$$

όπου $\varepsilon, \delta > 0$ και $\|\cdot\|$ μία μετρική αποστάσεων. Το υποσύνολο S' ονομάζεται ολικά βέλτιστο κατά Pareto, αν και μόνον αν:

$$\forall \mathbf{a}' \in S' : \nexists \mathbf{a} \in S : \mathbf{a} \prec \mathbf{a}' \quad (4.7)$$

Για να γίνει κατανοητός ο ορισμός του Pareto βέλτιστου σχεδιασμού, έστω το παρακάτω πρόβλημα βελτιστοποίησης με δύο αντικειμενικές συναρτήσεις:

$$\begin{aligned} \text{minimize } \mathbf{F} \mathbf{x} = \mathbf{F}_1 \mathbf{x}, \mathbf{F}_2 \mathbf{x}^T \\ \text{where: } \mathbf{F}_1 \mathbf{x} = x^2, \mathbf{F}_2 \mathbf{x} = x - 2^2 \end{aligned} \quad (4.8)$$

όπου x η μοναδική μεταβλητή σχεδιασμού. Τα γραφήματα των δύο αντικειμενικών συναρτήσεων φαίνονται στο Σχήμα 4.1.

Από το Σχήμα 4.1, φαίνεται καθαρά πως τα βέλτιστα κατά Pareto σημεία βρίσκονται στο διάστημα $[0, 2]$. Έξω από αυτό το διάστημα, και οι δύο αντικειμενικές συναρτήσεις αυξάνουν, ενώ μέσα σε αυτό το διάστημα υπάρχει ένας συμβιβασμός μεταξύ τους, αφού η μια αυξάνει και η άλλη μειώνεται.

Στο Σχήμα 4.2 απεικονίζεται γραφικά το ολικά βέλτιστο Pareto σύνολο σχεδιασμών.

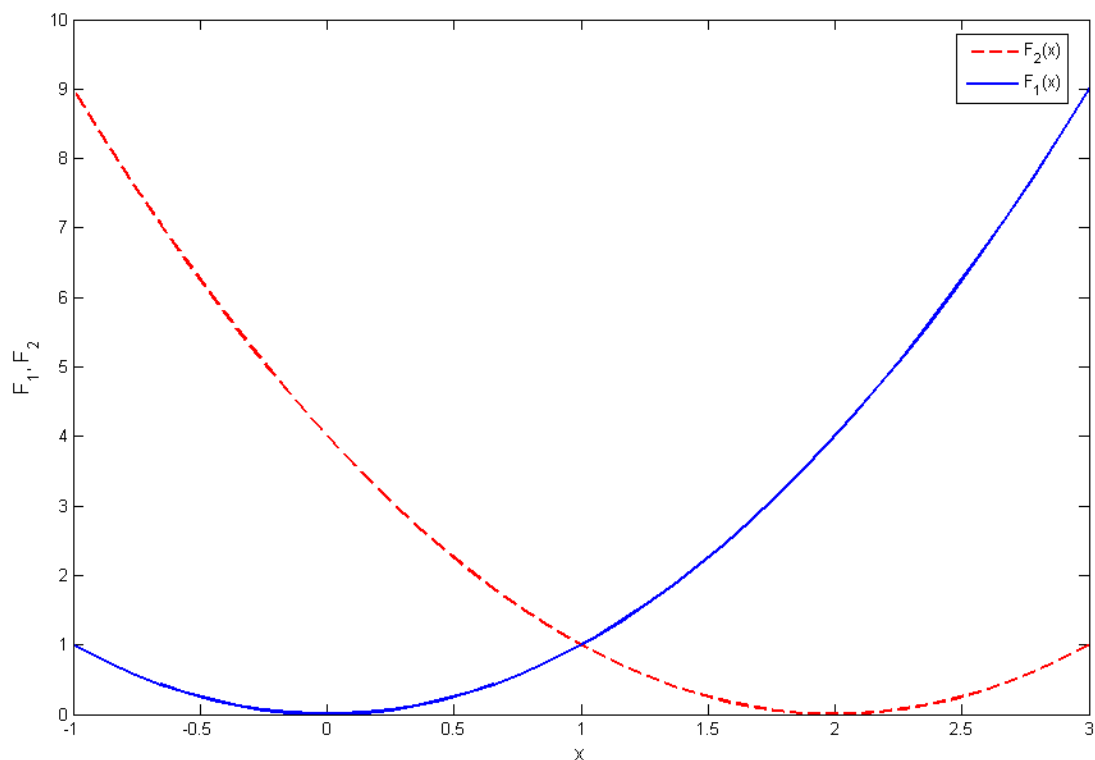
Όπως φαίνεται, σε αυτήν την περίπτωση υπάρχουν άπειρα σημεία-σχεδιασμοί οι οποίοι είναι βέλτιστοι. Αυτό το σύνολο λύσεων όμως δεν έχει καμία πρακτική εφαρμογή, αφού τελικά θα πρέπει να καθοριστεί από τον μηχανικό η σχετική βαρύτητα του κάθε κριτηρίου, ώστε να επιλεγεί τελικά ο βέλτιστος σχεδιασμός.

4.3 Κλασσικές μέθοδοι αντιμετώπισης προβλημάτων πολλαπλών κριτηρίων

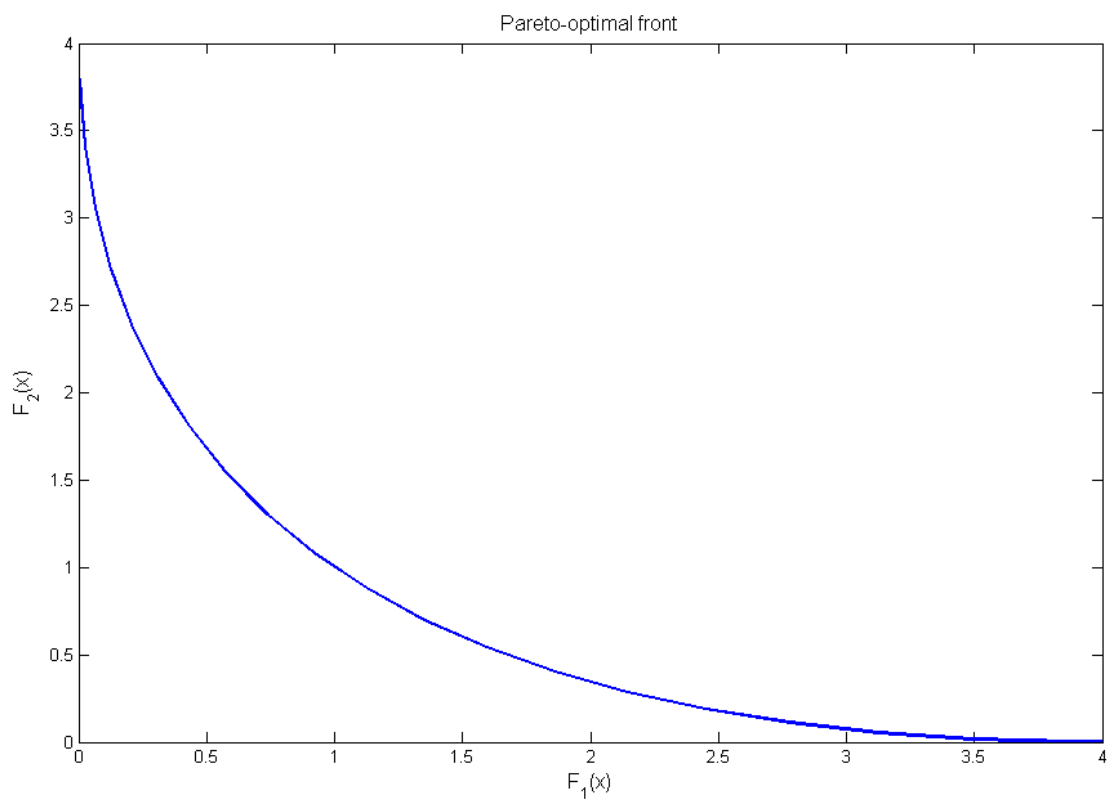
Τα τελευταία χρόνια έχουν αναπτυχθεί διάφορες μέθοδοι αντιμετώπισης προβλημάτων με πολλές αντικειμενικές συναρτήσεις. Όλες οι λεγόμενες κλασσικές μέθοδοι μετατρέπουν με κάποια μεθοδολογία το πρόβλημα των πολλαπλών κριτηρίων σε ένα πρόβλημα με ένα κριτήριο. Το βασικό πλεονέκτημα των κλασσικών μεθόδων, είναι πως υπάρχει μια πληθώρα αλγορίθμων

βελτιστοποίησης για προβλήματα με μία αντικειμενική συνάρτηση. Το βασικό μειονέκτημά τους είναι πως χρειάζονται η διαδικασία βελτιστοποίησης να πραγματοποιηθεί πολλές φορές για να βρεθεί το σύνολο των κατά Pareto βέλτιστων λύσεων.

Στις επόμενες παραγράφους δίνεται μια περιγραφή των κλασικών μεθόδων αντιμετώπισης τέτοιων προβλημάτων.



Σχήμα 4.1: Γραφική απεικόνιση των αντικειμενικών συναρτήσεων F_1 και F_2



Σχήμα 4.2: Βέλτιστο σύνολο σχεδιασμών κατά Pareto.

4.3.1 Γραμμική μέθοδος των βαρών

Η γραμμική μέθοδος των βαρών, μετατρέπει το πρόβλημα των πολλών αντικειμενικών συναρτήσεων, σε ένα πρόβλημα με μία αντικειμενική συνάρτηση. Αυτό γίνεται με τον συνδυασμό όλων των αντικειμενικών συναρτήσεων σε μία, με τη χρήση κατάλληλα επιλεγμένων βαρών. Έτσι η διατύπωση του προβλήματος γίνεται:

$$\text{minimize } f \mathbf{x} = \sum_{i=1}^m w_i F_i \mathbf{x}, \mathbf{x} \in S \quad (4.9)$$

όπου w_i είναι οι συντελεστές βαρών, $F_i \mathbf{x}$ είναι η i αντικειμενική συνάρτηση του αρχικού προβλήματος, \mathbf{x} είναι το διάνυσμα των μεταβλητών σχεδιασμού, S ο χώρος των εφικτών σχεδιασμών και $f \mathbf{x}$ είναι η συνδυασμένη αντικειμενική συνάρτηση. Οι συντελεστές βαρών w_i αντιστοιχούν στη βαρύτητα που θέλει να δώσει ο μηχανικός σε κάθε επιμέρους κριτήριο του αρχικού προβλήματος. Κάθε συνδυασμός των βαρών w_i οδηγεί σε μία κατά Pareto βέλτιστη λύση, οπότε με επιλογή των κατάλληλων συνδυασμών είναι δυνατό να παραχθεί μέρος ή και το σύνολο των κατά Pareto βέλτιστων σχεδιασμών.

Η γραμμική μέθοδος των βαρών είναι μια αξιόπιστη και με καλή συμπεριφορά μέθοδος. Ένα σοβαρό μειονέκτημά της είναι πως σε μή-κυρτά προβλήματα δεν υπάρχει εγγύηση της εύρεσης των βέλτιστων σχεδιασμών. Ένα άλλο μειονέκτημά της είναι πως θα πρέπει συνεχώς να δοκιμάζονται νέοι συνδυασμοί βαρών, ώστε να προκύψουν νέοι βέλτιστοι σχεδιασμοί.

4.3.2 Μέθοδος ελαχιστοποίησης της απόστασης από σημείο

Η μέθοδος αυτή μετατρέπει το πρόβλημα των πολλαπλών κριτηρίων σε ελαχιστοποίηση της απόστασης των σχεδιασμών από ένα σημείο αναφοράς. Το πρόβλημα διατυπώνεται ως:

$$\text{minimize } d_p \mathbf{x}, \mathbf{x} \in S \quad (4.10)$$

όπου S είναι ο χώρος των εφικτών λύσεων, \mathbf{x} είναι το διάνυσμα μεταβλητών συνδυασμού, και $d_p \mathbf{x}$ είναι η συνάρτηση απόστασης, η οποία ορίζεται από τη σχέση:

$$d_p \mathbf{x} = \left\{ \sum_{i=1}^m w_i \left(F_i \mathbf{x} - \mathbf{z}_i \right)^p \right\}^{1/p} \quad (4.11)$$

όπου m είναι ο αριθμός των αντικειμενικών συναρτήσεων του αρχικού προβλήματος, w_i είναι συντελεστές βαρών, F_i είναι η i αντικειμενική συνάρτηση, p είναι ένας ακέραιος αριθμός και \mathbf{z}_i είναι το i στοιχείο ενός διανύσματος αναφοράς $\mathbf{z} \in \mathbb{R}^m$. Το σημείο αναφοράς \mathbf{z} πρέπει να επιλεγεί από τον μηχανικό. Συνήθως, ως σημείο αναφοράς επιλέγεται το σημείο:

$$\mathbf{z} = \min F_1, \min F_2, \dots, \min F_m \quad (4.12)$$

όπου $\min F_i$ είναι οι βέλτιστες λύσεις των αντικειμενικών συναρτήσεων F_i , αν αυτές βελτιστοποιηθούν ξεχωριστά. Επομένως, αρχικά θα πρέπει να επιλυθούν m προβλήματα βελτιστοποίησης με ένα κριτήριο, ώστε να υπολογιστεί το διάνυσμα αναφοράς \mathbf{z} .

Στην περίπτωση που $p = 1$, το πρόβλημα μετασχηματίζεται σε αυτό της γραμμικής μεθόδου των βαρών, αν το διάνυσμα αναφοράς είναι το $\mathbf{z} = \mathbf{0}$, ενώ αν $p = 2$ η μέθοδος ονομάζεται *τετραγωνική μέθοδος των βαρών*, και φαίνεται να αποτελεί την επικρατέστερη μέθοδο στη βιβλιογραφία. Αν $p = \infty$, τότε το πρόβλημα (4.10) μετασχηματίζεται στο σταθμισμένο πρόβλημα ελαχιστοποίησης των μεγίστων, που διατυπώνεται ως εξής:

$$\min \max_i w_i F_i \mathbf{x}, \mathbf{x} \in S, i = 1, \dots, m \quad (4.13)$$

Η διατύπωση (4.13) μπορεί να δώσει όλες τις κατά Pareto βέλτιστες λύσεις, τόσο σε κυρτά, όσο και σε μή-κυρτά προβλήματα βελτιστοποίησης πολλαπλών κριτηρίων.

4.3.3 Μέθοδος περιορισμών

Στη μέθοδο των περιορισμών, όπως και στις προαναφερθείσες μεθόδους της ελαχιστοποίησης της απόστασης από σημείο, και της γραμμικής μεθόδου των βαρών, το αρχικό πρόβλημα πολλαπλών κριτηρίων μετασχηματίζεται σε ένα πρόβλημα βελτιστοποίησης με ένα κριτήριο. Αυτό επιτυγχάνεται επιλέγοντας μία από τις αντικειμενικές συναρτήσεις του αρχικού

προβλήματος ως μοναδική αντικειμενική συνάρτηση στο μετασχηματισμένο πρόβλημα. Οι υπόλοιπες αντικειμενικές συναρτήσεις του αρχικού προβλήματος θεωρούνται περιορισμοί στη διατύπωση του νέου προβλήματος.

Η διατύπωση του νέου προβλήματος δίνεται από τη σχέση:

$$\begin{aligned} & \text{minimize } F_p \mathbf{x}, \mathbf{x} \in S \\ & \text{subject to: } F_i \mathbf{x} - \varepsilon_i \leq 0, i = 1, \dots, m \quad \wedge \quad i \neq p \end{aligned} \quad (4.14)$$

όπου S είναι ο χώρος των εφικτών σχεδιασμών, $F_p \mathbf{x}$ είναι η αντικειμενική συνάρτηση του αρχικού προβλήματος που επιλέγεται προς βελτιστοποίηση στο νέο πρόβλημα, $F_i \mathbf{x}$ οι υπόλοιπες αντικειμενικές συναρτήσεις του αρχικού προβλήματος και ε_i παράμετροι που πρέπει να επιλεγούν από το μηχανικό.

Για κάθε συνδυασμό των παραμέτρων ε_i , προκύπτει από τη λύση του προβλήματος όπως διατυπώνεται από τη σχέση (4.14) μία νέα κατά Pareto βέλτιστη λύση. Με κατάλληλους συνδυασμούς των παραμέτρων ε_i , είναι δυνατόν να προκύψουν του συνόλου των κατά Pareto βέλτιστων λύσεων σε κυρτά και μή-κυρτά προβλήματα.

Το βασικό μειονέκτημα αυτής της μεθόδου είναι πως ο μηχανικός πρέπει να ιεραρχήσει τα αρχικά κριτήρια σε πρωτεύον, και δευτερεύοντα, και να επιλέγει κατάλληλες τιμές των παραμέτρων ε_i ώστε οι τελικές λύσεις να ανήκουν στον χώρο εφικτό λύσεων S .

4.3.4 Μέθοδος προγραμματισμού στόχων

Για να λειτουργήσει η μέθοδος αυτή, πρέπει τα κριτήρια να ιεραρχηθούν από το μηχανικό ως προς τη σπουδαιότητά τους. Ο μηχανικός, εκτός από την ταξινόμηση των κριτηρίων, πρέπει να θέσει και ένα «στόχο» για κάθε αντικειμενική συνάρτηση του προβλήματος. Το πρόβλημα μπορεί να διατυπωθεί ως εξής:

$$\begin{aligned} & \text{minimize } \gamma, \gamma \in \mathbb{R} \\ & \text{subject to: } F_i \mathbf{x} - w_i \gamma \leq F_i^*, \mathbf{x} \in S, i = 1, \dots, m \end{aligned} \quad (4.15)$$

όπου S είναι ο χώρος των εφικτών σχεδιασμών, w_i είναι συντελεστές βαρών, $F_i \mathbf{x}$ είναι η i αντικειμενική συνάρτηση του αρχικού προβλήματος και F_i^* είναι ο στόχος της i αντικειμενικής συνάρτησης. Η ιεράρχηση των αντικειμενικών συναρτήσεων γίνεται με κατάλληλη επιλογή των συντελεστών των βαρών w_i .

4.4 Οι εξελικτικοί αλγόριθμοι στη βελτιστοποίηση πολλαπλών κριτηρίων

Οι εξελικτικοί αλγόριθμοι φαίνεται να ταιριάζουν περισσότερο σε προβλήματα βέλτιστου σχεδιασμού πολλαπλών κριτηρίων, σε σχέση με τις κλασικές μεθόδους, αφού εξαιτίας της παράλληλης επεξεργασίας των πιθανών βέλτιστων σχεδιασμών, είναι δυνατό να βρεθεί το σύνολο των Pareto-βέλτιστων λύσεων σε μια μόνο διαδικασία βελτιστοποίησης-εξέλιξης.

Στην εφαρμογή των εξελικτικών αλγόριθμων σε προβλήματα βέλτιστου σχεδιασμού πολλαπλών κριτηρίων, δύο είναι τα βασικά προβλήματα που αντιμετωπίζονται:

1. Πώς πρέπει να δράσει ο τελεστής επιλογής, και με ποιον τρόπο πρέπει να υπολογίζεται η καταλληλότητα κάθε σχεδιασμού, ώστε η έρευνα να οδηγηθεί στο Pareto-βέλτιστο σύνολο σχεδιασμών.
2. Πώς θα διατηρηθεί μια ποικιλομορφία στους εφικτούς σχεδιασμούς ώστε να αποφευχθεί μια πρόωρη σύγκλιση μακριά από το Pareto-βέλτιστο σύνολο σχεδιασμών.

Οι αλγόριθμοι που έχουν προταθεί ως σήμερα, ακολουθούν γενικά διαφορετικές μεθοδολογίες για να αντιμετωπίσουν τα παραπάνω δύο προβλήματα. Στόχος κάθε εξελικτικού αλγόριθμου όμως, είναι να βρει πολλούς και διαφορετικούς μεταξύ τους σχεδιασμούς, να διατηρήσει μια «απόσταση» μεταξύ των παραγόμενων σχεδιασμών, και να μεγιστοποιήσει τις διαφορές των σχεδιασμών, ώστε να καλυφθεί τελικά όσο το δυνατόν μεγαλύτερο εύρος του Pareto-βέλτιστου συνόλου.

Όταν αρχικά χρησιμοποιήθηκαν οι εξελικτικοί αλγόριθμοι στη βελτιστοποίηση προβλημάτων με πολλές αντικειμενικές συναρτήσεις, δέχθηκαν κριτική εξαιτίας της υπολογιστικής τους πολυπλοκότητας, του μή-επιτιμικού τρόπου δράσης τους και της ανάγκης χρήσης επιπλέον

παραμέτρων ώστε να διατηρηθεί η ποικιλομορφία του πληθυσμού των σχεδιασμών. Στο του παραδοτέο Π1.1, επιλέχθηκαν τρεις state-of-the-art εξελικτικοί αλγόριθμοι βέλτιστου σχεδιασμού πολλαπλών κριτηρίων, και εφαρμόστηκαν σε μια πληθώρα μαθηματικών και πραγματικών προβλημάτων (βλ. Κεφάλαιο 6). Δύο από τους επιλεγέντες αλγορίθμους, ο NSGA-II και ο SPEA-2 (βλ. Κεφάλαιο 5), προέρχονται από το πεδίο των γενετικών αλγορίθμων και αποτελούν σημαντικές βελτιώσεις των προκατόχων τους, NSGA και SPEA αντίστοιχα, ενώ ο τρίτος αλγόριθμος είναι μια γενίκευση της ελιστιστικής E-CMA στρατηγικής εξέλιξης (βλ. Κεφάλαιο 3) ώστε να μπορεί να αντιμετωπίσει προβλήματα βελτιστοποίησης με πολλές αντικειμενικές συναρτήσεις, και ονομάζεται c-MO-CMA-ES.

4.5 Προβλήματα δομοστατικής πολλαπλών κριτηρίων

Στη διεθνή βιβλιογραφία, η επιλογή των κριτηρίων στη βελτιστοποίηση των κατασκευών διαφέρει. Το βάρος της κατασκευής ή ο όγκος της παραμένει η βασική επιλογή ως το πρώτο κριτήριο. Συνήθως, ως δεύτερο κριτήριο επιλέγεται μια έκφραση της ευκαμψίας της κατασκευής όπως κάποια συνάρτηση των μετατοπίσεών της, ή κάποια ενεργειακή έκφραση.

Στην περίπτωση των δικτυωτών φορέων, συνήθως επιλέγονται ως κριτήρια το βάρος της κατασκευής και κάποιες μετατοπίσεις ή και ιδιοσυχνότητες ως αλληλοσυγκρουόμενα κριτήρια. Στην περίπτωση των πλαισιωτών κατασκευών, συνήθως επιλέγονται ως κριτήρια το βάρος τους και κάποιες συναρτήσεις των τάσεων ή και των παραμορφώσεων.

5 ΕΞΕΛΙΚΤΙΚΟΙ ΑΛΓΟΡΙΘΜΟΙ ΒΕΛΤΙΣΤΟΠΟΙΗΣΗΣ ΠΟΛΛΑΠΛΩΝ ΚΡΙΤΗΡΙΩΝ

5.1 Ο αλγόριθμος NSGA-II

Ο αλγόριθμος NSGA-II (Nondominated Sorting Genetic Algorithm) προτάθηκε από τους Deb et al και αποτελεί μια σημαντική βελτίωση σε σχέση με τον προκάτοχό του, NSGA. Στον αλγόριθμο NSGA-II, γίνεται χρήση μιας νέας τεχνικής ταξινόμησης του πληθυσμού των σχεδιασμών που ονομάζεται Fast-nondominated-Sort. Παρουσιάζεται επίσης ένας νέος τελεστής επιλογής, ο οποίος θα επιλέξει ανάμεσα στον πληθυσμό των μ γονέων και των λ απογόνων τους μ καλύτερους σχεδιασμούς, με τέτοιο τρόπο ώστε να διατηρηθεί η ποικιλομορφία του πληθυσμού.

5.1.1 Μέθοδος ταξινόμησης Fast-nondominated-sort

Η μέθοδος Fast-nondominated-sort ταξινομεί έναν πληθυσμό ατόμων-σχεδιασμών σε επίπεδα «ισχύος» (nondomination levels). Η υπολογιστική πολυπλοκότητα είναι της τάξης $O m\gamma^2$, όπου m είναι το πλήθος αντικειμενικών συναρτήσεων, και γ είναι το μέγεθος του πληθυσμού. Στο Σχήμα 5.1 απεικονίζεται σε μορφή ψευδοκώδικα η υπορουτίνα Fast-nondominated-sort.

Για έναν σχεδιασμό p υπολογίζουμε δύο στοιχεία: i) τον αριθμό n_p των ατόμων τα οποία υπερिशύουν του σχεδιασμού p , και ii) το σύνολο S_p το οποίο αποτελείται από τους σχεδιασμούς των οποίων υπερिशύει ο σχεδιασμός p . Αυτό γίνεται στις γραμμές #2-3. Για να υπολογιστούν τα n_p και S_p κάθε σχεδιασμού, απαιτούνται υπολογισμοί τάξεως $O m\gamma^2$, όπου m είναι το πλήθος των αντικειμενικών συναρτήσεων και γ το μέγεθος του πληθυσμού \mathbf{P} , που είναι και το όρισμα της υπορουτίνας.

Κάθε σχεδιασμός p του πρώτου επιπέδου «ισχύος» F_1 , του επιπέδου δηλαδή του οποίου οι σχεδιασμοί υπερिशύουν όλων των άλλων σχεδιασμών, θα έχει $n_p = 0$, δηλαδή κανένας άλλος σχεδιασμός δεν θα υπερिशύει του σχεδιασμού p . Στις γραμμές #11-14, ελέγχεται αν ένας σχεδιασμός πρέπει να ανήκει στο πρώτο επίπεδο ισχύος F_1 . Αν κανείς άλλος σχεδιασμός δεν υπερिशύει του σχεδιασμού p τότε αυτός προστίθεται στο πρώτο επίπεδο ισχύος στη γραμμή #13.

Στη συνέχεια, μειώνουμε το δείκτη n_q κατά ένα (γραμμή #20), κάθε σχεδιασμού q ο οποίος ανήκει στο σύνολο των σχεδιασμών S_p , των σχεδιασμών δηλαδή των οποίων υπερिशύει ο σχεδιασμός q . Αν ο δείκτης n_q μηδενιστεί, τότε τοποθετούμε το σχεδιασμό q στη λίστα \mathbf{Q} . Οι σχεδιασμοί για τους οποίους θα μηδενιστεί ο δείκτης n_q , αποτελούν το δεύτερο επίπεδο ισχύος του πληθυσμού. Η όλη διαδικασία επαναλαμβάνεται μέχρι να βρεθούν όλα τα επίπεδα ισχύος (nondomination fronts) του πληθυσμού (γραμμές #18-29).

Θα πρέπει να σημειωθεί πως, αν και η υπολογιστική πολυπλοκότητα της μεθόδου είναι $O m\gamma^2$, οι απαιτήσεις αποθηκευτικού χώρου είναι της τάξης $O \gamma^2$, δηλαδή αρκετά μεγάλες. Ο προκάτοχος του NSGA-II, NSGA είχε απαιτήσεις αποθηκευτικού χώρου τάξης $O \gamma$ και υπολογιστική πολυπλοκότητα ταξινόμησης του πληθυσμού τάξης $O m\gamma^3$. Επομένως, αν και μειώθηκε η πολυπλοκότητα της ταξινόμησης στον NSGA-II, αυξήθηκαν οι απαιτήσεις αποθηκευτικού χώρου.

5.1.2 Μέθοδος συντήρησης της ποικιλομορφίας του πληθυσμού

Εκτός από τη σύγκλιση στο Pareto-βέλτιστο σύνολο σχεδιασμών, απαιτείται από έναν εξελικτικό αλγόριθμο βελτιστοποίησης να διατηρεί μια ποικιλομορφία στους σχεδιασμούς. Στον προκάτοχο του NSGA-II, τον NSGA, γινόταν χρήση μιας παραμέτρου σ_{share} η οποία δήλωνε τη μεγαλύτερη δυνατή απόσταση, εκφραζόμενη από κάποια μετρική, για την οποία δύο σχεδιασμοί θεωρούνταν ισοδύναμοι. Η παράμετρος σ_{share} έπρεπε να επιλεγεί από το χρήστη. Ένα βασικό μειονέκτημα χρήσης της παραμέτρου σ_{share} , ήταν πως η ποικιλομορφία του τελικά παραγόμενου από τον αλγόριθμο NSGA, συνόλου σχεδιασμών, είχε άμεση εξάρτηση από την παράμετρο σ_{share} . Επιπλέον, για κάθε σχεδιασμό θα έπρεπε να υπολογιστεί η απόστασή του από όλους τους άλλους, επομένως η πολυπλοκότητα της μεθοδολογίας με την παράμετρο σ_{share} ήταν τάξεως $O \gamma^2$, όπου γ το πλήθος των σχεδιασμών.

Ο αλγόριθμος NSGA-II δεν χρησιμοποιεί την παράμετρο σ_{share} του προκάτοχου του, αλλά έναν τελεστή ο οποίος συγκρίνει τους σχεδιασμούς ως προς τη συσσώρευση άλλων σχεδιασμών γύρω τους, και επιλέγει αυτούς με τη μικρότερη συσσώρευση. Για να συμβεί βέβαια αυτό, πρέπει

πρώτα να οριστεί μια κατάλληλη μετρική, βάσει της οποίας θα υπολογίζεται η πυκνότητα των σχεδιασμών γύρω από κάποιον άλλο σχεδιασμό.

Procedure: Fast– nondominated– sort **P**

```

1  For each  $p \in P$ 
2       $S_p = \emptyset$ 
3       $n_p = 0$ 
4      For each  $q \in P$ 
5          If  $p \prec q$  Then
6               $S_p = S_p \cup q$ 
7          Else if  $q \prec p$ 
8               $n_p = n_p + 1$ 
9          End
10     End
11     If  $n_p = 0$  Then
12          $p_{rank} = 1$ 
13          $F_1 = F_1 \cup p$ 
14     End
15      $i = 1$ 
16     While  $F_i \neq \emptyset$  Do
17          $Q = \emptyset$ 
18         For each  $p \in F_i$ 
19             For each  $q \in S_p$ 
20                  $n_q = n_q - 1$ 
21             If  $n_q = 0$  Then
22                  $q_{rank} = i + 1$ 
23                  $Q = Q \cup q$ 
24             End
25         End
26     End
27      $i = i + 1$ 
28      $F_i = Q$ 
29 End

```

Σχήμα 5.1: Ψευδοκώδικας της υπορουτίνας Fast-nondominated-sort

Για να υπολογιστεί η ζητούμενη πυκνότητα των σχεδιασμών γύρω από σχεδιασμό p , υπολογίζεται αρχικά η μέση απόσταση των δύο σχεδιασμών που βρίσκονται εγγύτερα του σχεδιασμού p . Η απόσταση αυτή ονομάζεται απόσταση συσσώρευσης (crowding distance), συμβολίζεται με $i_{distance}$ και αποτελεί μια εκτίμηση της περιμέτρου του κυβοειδούς που σχηματίζεται αν θεωρηθούν ως κορυφές του οι δύο εγγύτεροι σχεδιασμοί του p .

Στο Σχήμα 5.2 φαίνεται η υπορουτίνα crowding-distance-assignment σε μορφή ψευδοκώδικα. Ο υπολογισμός της απόστασης συσσώρευσης απαιτεί τη ταξινόμηση του πληθυσμού ως προς κάθε αντικειμενική συνάρτηση κατά αύξουσα τιμή της. Η ταξινόμηση αυτή γίνεται στη γραμμή #3. Στη συνέχεια, για κάθε αντικειμενική συνάρτηση, οι συνοριακοί σχεδιασμοί, οι σχεδιασμοί δηλαδή με τη μικρότερη και τη μεγαλύτερη τιμές της κάθε αντικειμενικής συνάρτησης, θεωρείται

πως έχουν άπειρη απόσταση συσσώρευσης (γραμμή #5). Η απόσταση συσσώρευσης για έναν ενδιαμέσο σχεδιασμό \mathbf{p} και για την αντικειμενική συνάρτηση m , ισούται με την απόλυτη διαφορά των δύο γειτονικών σχεδιασμών, κανονικοποιημένη ως προς τη διαφορά της ελάχιστης από τη μέγιστη τιμή της αντικειμενικής συνάρτησης m (γραμμή #7). Η ολική απόσταση συσσώρευσης κάθε σχεδιασμού υπολογίζεται ως το άθροισμα των επιμέρους αποστάσεων συσσώρευσης για κάθε αντικειμενική συνάρτηση. Θα πρέπει να τονιστεί, πως οι αποστάσεις υπολογίζονται στο χώρο των αντικειμενικών συναρτήσεων \mathbb{R}^m .

Ο συμβολισμός $\mathbf{I}^i . m$ αναφέρεται στην τιμή της m αντικειμενικής συνάρτησης, του i σχεδιασμού, στο επίπεδο ισχύος \mathbf{I} . Τα μεγέθη f_m^{\max} , f_m^{\min} είναι η μέγιστη και η ελάχιστη τιμή της m αντικειμενικής συνάρτησης. Στην υλοποίηση του αλγόριθμου, τα f_m^{\max} και f_m^{\min} υπολογίστηκαν βάσει όλου του πληθυσμού, και όχι θεωρώντας μόνον το επίπεδο ισχύος \mathbf{I} . Η υπολογιστική πολυπλοκότητα της όλης διαδικασίας υπολογισμού της απόστασης συσσώρευσης, εξαρτάται κυρίως από τον αλγόριθμο ταξινόμησης που χρησιμοποιείται στη γραμμή #4. Στη χειρότερη περίπτωση η πολυπλοκότητα είναι της τάξης $O(m\gamma \log \gamma)$, όπου γ το μέγεθος του πληθυσμού.

Μόλις ολοκληρωθεί η διαδικασία του υπολογισμού της απόστασης συσσώρευσης για κάθε σχεδιασμό του πληθυσμού, μπορούν να συγκριθούν μεταξύ τους δύο σχεδιασμοί ως προς την εγγύτητά τους με άλλους σχεδιασμούς. Ένας σχεδιασμός με μικρή τιμή απόστασης συσσώρευσης, θεωρείται πως είναι πιο συνωστισμένος σε σχέση με κάποιον άλλον σχεδιασμό που έχει μεγαλύτερη απόσταση συσσώρευσης. Η σύγκριση μεταξύ των σχεδιασμών του πληθυσμού γίνεται με τον τελεστή σύγκρισης των αποστάσεων συσσώρευσης (crowded comparison operator). Ο τελεστής σύγκρισης των αποστάσεων συσσώρευσης, η δράση του οποίου συμβολίζεται με \prec_n οδηγεί την έρευνα σε ένα ποικιλόμορφο Pareto-βέλτιστο σύνολο σχεδιασμών.

Αν υποθέσουμε πως κάθε σχεδιασμός i έχει έναν βαθμό ισχύος (nondomination rank) i_{rank} και μια απόσταση συσσώρευσης $i_{distance}$, τότε σύγκριση με έναν σχεδιασμό j γίνεται από τη σχέση:

$$i \prec_n j \text{ if } i_{rank} < j_{rank} \vee i_{rank} = j_{rank} \wedge i_{distance} > j_{distance} \quad (5.1)$$

Δηλαδή, μεταξύ δύο σχεδιασμών i, j επιλέγουμε εκείνον το σχεδιασμό με το μικρότερο (καλύτερο) επίπεδο ισχύος. Αν και οι δύο σχεδιασμοί ανήκουν στο ίδιο επίπεδο ισχύος, επιλέγουμε εκείνον το σχεδιασμό που είναι λιγότερο συνωστισμένος.

Procedure: crowding– distance– assignment \mathbf{I}

```

1  $l = |\mathbf{I}|$ 
2 For each  $i$ , set  $\mathbf{I}^i_{distance} = 0$ 
3 For each objective  $m$ 
4    $\mathbf{I} = \text{Sort } \mathbf{I}, m$ 
5    $\mathbf{I}^i_{distance} = \mathbf{I}^l_{distance} = \infty$ 
6   For  $i = 2$  to  $l - 1$ 
7      $\mathbf{I}^i_{distance} = \mathbf{I}^i_{distance} + \mathbf{I}^{i+1}.m - \mathbf{I}^{i-1}.m / f_m^{\max} - f_m^{\min}$ 
8   End
9 End

```

Σχήμα 5.2: Ψευδοκώδικας της μεθόδου crowding-distance-assignment

5.1.3 Περιγραφή του αλγόριθμου

Ο αλγόριθμος NSGA-II, όπως διατυπώθηκε αρχικά, ανήκει στο πεδίο των γενετικών αλγορίθμων. Στο παραδοτέο Π1.1, κρίθηκε σκόπιμο να τροποποιηθεί ο αλγόριθμος και να συνδυαστεί με τον αλγόριθμο A-ES (βλ. Κεφάλαιο 3).

Στο Σχήμα 5.3 απεικονίζεται η ροή του αλγόριθμου NSGA-II, όπως αυτός τροποποιήθηκε. Το πρώτο βήμα του αλγορίθμου, είναι να δημιουργηθεί ο αρχικός πληθυσμός γονέων \mathbf{B}_p^0 (γραμμή #3). Στη συνέχεια, από τον πληθυσμό των μ γονέων δημιουργούνται λ απόγονοι (γραμμές #5-12). Στη γραμμή #6, επιλέγεται τυχαία ένας ακέραιος αριθμός $k \in \{1, \dots, \mu\}$. Επιλέγεται στη συνέχεια ο γονέας \mathbf{p}_l , που είναι ο k γονέας του πληθυσμού \mathbf{B}_p^g , από τον οποίο θα δημιουργηθεί ένας νέος απόγονος, στις γραμμές #10-11.

Algorithm: $\mu + \lambda$ -A-ES / NSGA-II

```

1 Begin
2  $g := 0$ 
3 initialize  $\mathbf{B}_p^0 := \mathbf{y}_m^0, \mathbf{s}_m^0, \mathbf{F} \mathbf{y}_m^0$ ,  $m = 1, \dots, \mu$ 
4   Repeat
5     For  $l := 1$  To  $\lambda$  Do Begin
6        $k = \text{random\_index } \mu$ 
7        $\mathbf{p}_l := \text{SelectParent } \mathbf{B}_p^g, k$ 
8        $\mathbf{s}_l := \text{GetStrategy } \mathbf{p}_l$ 
9        $\mathbf{y}_l := \text{GetDesign } \mathbf{p}_l$ 
10       $\tilde{\mathbf{s}}_l := \text{s\_mutation } \mathbf{s}_l$ 
11       $\tilde{\mathbf{y}}_l := \text{y\_mutation } \mathbf{y}_l$ 
12    End
13     $\mathbf{B}_o^g := \mathbf{y}_l^g, \mathbf{s}_l^g, \mathbf{F} \mathbf{y}_l^g$ ,  $l = 1, \dots, \lambda$ 
14     $\mathbf{R}^g = \mathbf{B}_p^g \cup \mathbf{B}_o^g$ 
15     $\mathbf{S}^g = \text{fast-nondominated-sort } \mathbf{R}^g$ 
16     $\mathbf{B}_p^{g+1} = \emptyset$ ,  $i = 1$ 
17    Until  $|\mathbf{B}_p^{g+1}| + |\mathbf{S}_i| \leq \mu$  Do
18      crowding-distance-assignment  $\mathbf{S}_i$ 
19       $\mathbf{B}_p^{g+1} = \mathbf{B}_p^{g+1} \cup \mathbf{S}_i$ 
20       $i = i + 1$ 
21    End
22    crowded-comparison-operator  $\mathbf{S}_i$ 
23     $\mathbf{B}_p^{g+1} = \mathbf{B}_p^{g+1} \cup \mathbf{S}_i \left[ 1 : \mu - |\mathbf{B}_p^{g+1}| \right]$ 
24     $g := g + 1$ 
25.  Until termination_condition
26. End

```

Σχήμα 5.3: Ψευδοκώδικας του αλγόριθμου $(\mu+\lambda)$ -A-ES / NSGA-II.

Η διαδικασία των γραμμών #6-11 επαναλαμβάνεται λ φορές, ώστε να δημιουργηθεί ο πληθυσμός των απογόνων \mathbf{B}_o^g στη γραμμή #13. Στη γραμμή #14, οι πληθυσμοί γονέων και απογόνων ενώνονται στον πληθυσμό \mathbf{R}^g . Ο πληθυσμός \mathbf{R}^g αποτελείται από $\gamma = \mu + \lambda$ σχεδιασμούς. Στη συνέχεια, καλείται η συνάρτηση Fast-nondominated-sort η οποία θα ταξινομήσει τον πληθυσμό \mathbf{R}^g ως σε επίπεδα ισχύος, όπως περιγράφηκε προηγουμένα. Επειδή και οι απόγονοι

καί οι γονείς συμπεριλαμβάνονται στον πληθυσμό \mathbf{R}^g που ταξινομείται, η ελίτ των σχεδιασμών θα διατηρηθεί, επομένως ο αλγόριθμος NSGA-II είναι ελιτιστικός. Στο σύνολο \mathbf{F}_1 , με το πέρας της ταξινόμησης, θα ανήκουν οι υπερισχύωντες όλων των άλλων σχεδιασμοί του πληθυσμού \mathbf{R}^g . Στη γραμμή #16, δημιουργείται ένα κενό σύνολο σχεδιασμών \mathbf{B}_p^{g+1} το οποίο στη συνέχεια θα αποτελέσει τον πληθυσμό γονέων της νέας γενιάς. Αν το πλήθος των σχεδιασμών που ανήκουν στον σύνολο \mathbf{F}_1 είναι μικρότερο από μ , τότε όλοι αυτοί οι σχεδιασμοί υπεισέρχονται στο σύνολο \mathbf{B}_p^{g+1} . Τα υπόλοιπα μέλη του συνόλου \mathbf{B}_p^{g+1} επιλέγονται διαδοχικά από τα επόμενα επίπεδα ισχύος. Έτσι, επιλέγονται στη συνέχεια οι σχεδιασμοί του επιπέδου ισχύος \mathbf{F}_2 , μετά του επιπέδου \mathbf{F}_3 , κλπ. Η διαδικασία αυτή (γραμμές #17-21) συνεχίζεται μέχρι το πλήθος των σχεδιασμών στον πληθυσμό \mathbf{B}_p^{g+1} να γίνει τουλάχιστον ίσο με μ .

Σε περίπτωση που συμβεί το πλήθος των σχεδιασμών να γίνει ακριβώς ίσο με μ , η διαδικασία σταματάει και ο αλγόριθμος συνεχίζει με την επόμενη γενιά εξέλιξης. Στη συνηθέστερη όμως περίπτωση, το πλήθος των σχεδιασμών θα είναι μεγαλύτερο από μ . Επειδή πρέπει να επιλεγούν ακριβώς μ σχεδιασμοί ως πληθυσμός γονέων για την επόμενη γενιά εξέλιξης, ταξινομείται το τελευταίο επίπεδο ισχύος που εισήλθε στον πληθυσμό \mathbf{B}_p^{g+1} ως προς τον τελεστή σύγκρισης της απόστασης συσσώρευσης \prec_n (crowded-comparison-operator). Ουσιαστικά, επειδή οι σχεδιασμοί του τελευταίου επιπέδου ισχύος που χρησιμοποιήθηκε ανήκουν στο ίδιο επίπεδο ισχύος, θα ταξινομηθούν ως προς την απόσταση συσσώρευσης σε φθίνουσα σειρά (γραμμή #22). Από τους ταξινομημένους σχεδιασμούς, θα επιλεγούν τόσσοι ώστε ο πληθυσμός \mathbf{B}_p^{g+1} να αποτελείται από μ ακριβώς σχεδιασμούς, όπως φαίνεται στη γραμμή #23.

Η όλη διαδικασία επαναλαμβάνεται μέχρι να ικανοποιηθεί το κριτήριο τερματισμού του αλγόριθμου. Συνήθως, ως κριτήριο τερματισμού επιλέγεται ο μέγιστος αριθμός των γενεών g .

Η υπολογιστική πολυπλοκότητα των βασικών διαδικασιών του αλγόριθμου NSGA-II, στη χειρότερη των περιπτώσεων είναι:

1. Η ταξινόμηση των σχεδιασμών με τη μέθοδο Fast-nondominated-sort είναι της τάξης $O(m(\mu + \lambda))$.
2. Ο υπολογισμός της απόστασης συσσώρευσης για ολόκληρο τον πληθυσμό είναι της τάξης $O(m(\mu + \lambda) \log(\mu + \lambda))$.
3. Η ταξινόμηση όλου του πληθυσμού με τον τελεστή σύγκρισης \prec_n είναι της τάξης $O((\mu + \lambda) \log(\mu + \lambda))$.

Επομένως η ολική υπολογιστική πολυπλοκότητα του αλγόριθμου NSGA-II είναι της τάξης $O(m(\mu + \lambda)^2)$ στην οποία κυριαρχεί υπολογιστικά η μέθοδος ταξινόμησης Fast-nondominated-sort.

Θα πρέπει εδώ να σημειωθεί, πως δεν είναι αναγκαίο να ταξινομηθούν όλοι οι σχεδιασμοί σε επίπεδα ισχύος. Η ταξινόμηση μπορεί να σταματήσει όταν το άθροισμα των σχεδιασμών των επιπέδων ισχύος που έχουν βρεθεί είναι τουλάχιστον ίσο με το πλήθος των γονέων μ .

5.2 Ο αλγόριθμος SPEA-2

Ο εξελικτικός αλγόριθμος SPEA-2 (Strength Pareto Evolutionary Algorithm 2) προτάθηκε από τους Zitzler et al, και ανήκει και αυτός, όπως και ο αλγόριθμος NSGA-II στο πεδίο των γενετικών αλγορίθμων (GA). Ο αλγόριθμος SPEA-2 αποτελεί εξέλιξη και βελτίωση του προκατόχου του SPEA. Τα καινοτόμα στοιχεία του αλγόριθμου SPEA-2 σε σχέση με τον SPEA, είναι:

1. Μια βελτιωμένη μέθοδος υπολογισμού της καταλληλότητας (fitness) κάθε ατόμου-σχεδιασμού του πληθυσμού. Η νέα μέθοδος λαμβάνει υπόψιν πόσοι σχεδιασμοί υπερισχύουν ενός σχεδιασμού, και πόσων σχεδιασμών υπερισχύει αυτός.
2. Μια νέα μέθοδος υπολογισμού της πυκνότητας σχεδιασμών γύρω από άλλους σχεδιασμούς, η οποία οδηγεί με μεγαλύτερη ακρίβεια τη διαδικασία έρευνας στο Pareto-βέλτιστο σύνολο σχεδιασμών.
3. Μια νέα μέθοδος διατήρησης και αποκοπής (truncation) του «αρχείου» των σχεδιασμών, η οποία εγγυάται τη διατήρηση της ποικιλομορφίας του πληθυσμού.

Στις επόμενες παραγράφους δίνεται μια πλήρης περιγραφή του αλγόριθμου SPEA-2, όπως αυτός χρησιμοποιήθηκε στο παραδοτέο Π1.1.

5.2.1 Περιγραφή του αλγορίθμου SPEA-2

Ο αλγόριθμος SPEA-2, όπως και ο αλγόριθμος NSGA-2, συνδυάστηκε και αυτός με τον αλγόριθμο A-ES (βλ. Κεφάλαιο 3). Επομένως, δεν έγινε χρήση των γενετικών τελεστών διασταύρωσης, επιλογής και μετάλλαξης των γενετικών αλγορίθμων, αλλά η γένεση νέων απογόνων έγινε από τον αλγόριθμο A-ES. Ο αλγόριθμος SPEA-2 περιγράφεται στο Σχήμα 5.4.

Αρχικά, δημιουργείται ένας πληθυσμός \mathbf{B}_p^0 πλήθους μ σχεδιασμών (γραμμή #3). Επίσης δημιουργείται ένα κενό «αρχείο» σχεδιασμών \mathbf{A}^g . Ο μέγιστος αριθμός σχεδιασμών που μπορεί να υπάρχουν στο εξωτερικό αρχείο \mathbf{A}^g , στο τέλος κάθε γενιάς εξέλιξης είναι ίσος με λ .

Στη συνέχεια, στις γραμμές #6-7, ενώνονται ο πληθυσμός \mathbf{B}_p^g με το εξωτερικό αρχείο \mathbf{A}^g στο συνολικό πληθυσμό σχεδιασμών \mathbf{R}^g και υπολογίζεται η καταλληλότητα κάθε σχεδιασμού από την υπορουτίνα AssignFitness. Στη γραμμή #8, καλείται η υπορουτίνα EnvironmentalSelection στην οποία θα επιλεγούν οι σχεδιασμοί οι οποίοι θα παραμείνουν στο εξωτερικό αρχείο \mathbf{A}^g .

Algorithm: $\mu + \lambda$ -A-ES / SPEA-2

```

1 Begin
2    $g := 0$ 
3   initialize  $\mathbf{B}_p^0 := \mathbf{y}_m^0, \mathbf{s}_m^0, \mathbf{F} \mathbf{y}_m^0$  ,  $m = 1, \dots, \mu$ 
4    $\mathbf{A}^g = \emptyset$ 
5   Repeat
6      $\mathbf{R}^g = \mathbf{B}_p^g \cup \mathbf{A}^g$ 
7     AssignFitness  $\mathbf{R}^g$ 
8      $\mathbf{A}^{g+1} = \text{EnvironmentalSelection } \mathbf{R}^g$ 
9     For  $l := 1$  To  $\mu$  Do Begin
10       $k = \text{random\_index } \lambda$ 
11       $\mathbf{p}_l := \text{SelectParent } \mathbf{A}^{g+1}, k$ 
12       $\mathbf{s}_l := \text{GetStrategy } \mathbf{p}_l$ 
13       $\mathbf{y}_l := \text{GetDesign } \mathbf{p}_l$ 
14       $\tilde{\mathbf{s}}_l := \text{s\_mutation } \mathbf{s}_l$ 
15       $\tilde{\mathbf{y}}_l := \text{y\_mutation } \mathbf{y}_l$ 
16       $\mathbf{B}_p^g \ l := \tilde{\mathbf{y}}_l, \tilde{\mathbf{s}}_l, \mathbf{F} \tilde{\mathbf{y}}_l$ 
17     End
18      $g := g + 1$ 
19   Until termination\_condition
20 End

```

Σχήμα 5.4: Ψευδοκώδικας του αλγορίθμου $(\mu + \lambda)$ -A-ES /SPEA-2.

Έπειτα, στις γραμμές #9-17 δημιουργείται ο νέος πληθυσμός \mathbf{B}_p^g . Αρχικά, επιλέγεται τυχαία ένα άτομο \mathbf{p}_l από το εξωτερικό αρχείο σχεδιασμών \mathbf{A}^g (γραμμές #10-11). Αφού δράσει στο διάνυσμα παραμέτρων στρατηγικής \mathbf{s}_l και σχεδιασμού \mathbf{y}_l ο τελεστής μετάλλαξης του αλγορίθμου A-ES, δημιουργείται ένας νέος σχεδιασμός με διάνυσμα σχεδιασμού $\tilde{\mathbf{y}}_l$ και διάνυσμα

παραμέτρων στρατηγικής \tilde{s}_i (γραμμή #16). Αυτή η διαδικασία επαναλαμβάνεται μ φορές, ώστε να προκύψει ένας νέος πληθυσμός \mathbf{B}_p^g .

Στη γραμμή #19 ελέγχεται το κριτήριο τερματισμού του αλγόριθμου. Ως κριτήριο τερματισμού συνήθως χρησιμοποιείται ο μέγιστος αριθμός γενεών όπως και στον αλγόριθμο $\mu + \lambda$ -A-ES/NSGA-II.

Το εξωτερικό αρχείο σχεδιασμών \mathbf{A} , θα αποτελείται από τους Pareto-βέλτιστους σχεδιασμούς, με το πέρας της διαδικασίας εξέλιξης. Επειδή το αρχείο σχεδιασμών \mathbf{A} διατηρεί πάντα τους καλύτερους σχεδιασμούς που έχουν βρεθεί, ο εξελικτικός αλγόριθμος SPEA-2 είναι ελιτιστικός.

5.2.2 Μέθοδος ταξινόμησης των σχεδιασμών

Για να υπολογιστεί η καταλληλότητα (fitness) των σχεδιασμών ενός πληθυσμού, για κάθε σχεδιασμό i υπολογίζεται η δύναμή του S_i , η οποία ισούται με τον αριθμό των σχεδιασμών του πληθυσμού των οποίων υπερिσχύει ο σχεδιασμός i . Δηλαδή:

$$S_i = | \{ j \mid j \in \mathbf{B} \cup \mathbf{A} \wedge i \prec j \} | \quad (5.2)$$

όπου $\mathbf{B} \cup \mathbf{A}$ είναι η ένωση του πληθυσμού των γονέων \mathbf{B} με το εξωτερικό αρχείο σχεδιασμών \mathbf{A} .

Με βάση τις τιμές δύναμης S , υπολογίζεται στη συνέχεια η αρχική καταλληλότητα (raw fitness) R_i κάθε σχεδιασμού, από τη σχέση:

$$R_i = \sum_{j \in \mathbf{B} \cup \mathbf{A} \wedge i \prec j} S_j \quad (5.3)$$

Δηλαδή, η αρχική καταλληλότητα R_i του σχεδιασμού i , ισούται με το άθροισμα των τιμών δύναμης S_j όλων των σχεδιασμών j , οι οποίοι υπερिσχύουν του σχεδιασμού i . Προφανώς, δεν υπάρχει σχεδιασμός που να υπερिσχύει του σχεδιασμού i , τότε $R_i = 0$. Μια μεγάλη τιμή R_i , υποδηλώνει πως υπάρχουν πολλοί σχεδιασμοί που υπερिσχύουν του σχεδιασμού i .

Για να μπορούν να συγκριθούν σχεδιασμοί με ίδιες τιμές αρχικής καταλληλότητας R , χρησιμοποιείται μια τεχνική εκτίμησης της πυκνότητας σχεδιασμών γύρω από άλλους. Η τεχνική εκτίμησης της πυκνότητας στον αλγόριθμο SPEA-2 βασίζεται στη μέθοδο του k κοντινότερου σημείου (k -th nearest neighbor). Η πυκνότητα σε κάθε σχεδιασμό θεωρείται πως είναι μια συνάρτηση της απόστασης αυτού του σχεδιασμού από το k κοντινότερο σημείο. Πιο συγκεκριμένα, για κάθε σχεδιασμό $i \in \mathbf{B} \cup \mathbf{A}$, υπολογίζεται το διάλυσμα των αποστάσεων του, distance j από όλους τους άλλους σχεδιασμούς j , στο χώρο των αντικειμενικών συναρτήσεων \mathbb{R}^m . Οι αποστάσεις του σχεδιασμού i από τους υπόλοιπους σχεδιασμούς, ταξινομούνται στη συνέχεια σε αύξουσα σειρά. Η απόσταση σ_i^k του k κοντινότερου σχεδιασμού από το σχεδιασμό i , δίνεται από τη σχέση:

$$\sigma_i^k = \text{sorted_distance } k, \quad k = \sqrt{\mu + \lambda} \quad (5.4)$$

όπου μ το μέγεθος του πληθυσμού γονέων \mathbf{B} , λ το μέγεθος του εξωτερικού αρχείου \mathbf{A} και sorted_distance k είναι το k στοιχείο του ταξινομημένου κατά αύξουσα απόσταση διανύσματος distance.

Η πυκνότητα D_i του σχεδιασμού i δίνεται από τη σχέση:

$$D_i = \frac{1}{\sigma_i^k + 2} \quad (5.5)$$

Τέλος, για να υπολογιστεί η καταλληλότητα F_i του σχεδιασμού i , χρησιμοποιείται η σχέση:

$$F_i = R_i + D_i \quad (5.6)$$

Η υπολογιστική πολυπλοκότητα του υπολογισμού της πυκνότητας D , είναι της τάξης $O(\mu + \lambda \log(\mu + \lambda))$, ενώ η πολυπλοκότητα του υπολογισμού των τιμών της δύναμης S και της αρχικής καταλληλότητας R είναι της τάξης $O(\mu + \lambda)$.

Ο υπολογισμός της καταλληλότητας των σχεδιασμών γίνεται στην υπορουτίνα AssignFitness (βλ. Σχήμα 5.4, γραμμή #7). Στο Σχήμα 5.5 φαίνεται η υπορουτίνα AssignFitness σε μορφή ψευδοκώδικα.

Στις γραμμές #2-10 υπολογίζεται η δύναμη κάθε σχεδιασμού, όπως περιγράφεται από τη σχέση (5.2). Στη συνέχεια, στις γραμμές #12-18 υπολογίζεται η αρχική καταλληλότητα των σχεδιασμών, όπως περιγράφεται από τη σχέση (5.3). Έπειτα, στις γραμμές #20-25 υπολογίζεται η απόσταση του k κοντινότερου σχεδιασμού για όλους τους σχεδιασμούς στον πληθυσμό. Στις

γραμμές #21-23 υπολογίζεται η απόσταση του σχεδιασμού \mathbf{p} από κάθε σχεδιασμό \mathbf{q} , και αποθηκεύεται στη λίστα Distance (γραμμή #22). Στη γραμμή #24, η λίστα αυτή ταξινομείται σε αύξουσα σειρά, και μετά υπολογίζεται η απόσταση του k κοντινότερου σχεδιασμού στο σχεδιασμό \mathbf{p} .

Τέλος, στη γραμμή #26, υπολογίζεται η πυκνότητα γύρω από τον σχεδιασμό \mathbf{p} βάσει της σχέσης (5.5) και έπειτα υπολογίζεται η καταλληλότητα του σχεδιασμού \mathbf{p} από τη σχέση (5.6).

Procedure: AssignFitness \mathbf{R}^g

```

1   ! Calculate strength S
2   For each  $\mathbf{p} \in \mathbf{R}^g$ 
3        $i = 0$ 
4       For each  $\mathbf{q} \in \mathbf{R}^g$ 
5           If  $\mathbf{p} \prec \mathbf{q}$  Then
6                $i = i + 1$ 
7           End
8        $S_{\mathbf{p}} = i$ 
9   End
10  End
11  ! Calculate raw fitness R
12  For each  $\mathbf{p} \in \mathbf{R}^g$ 
13      For each  $\mathbf{q} \in \mathbf{R}^g$ 
14          If  $\mathbf{q} \prec \mathbf{p}$  Then
15               $R_{\mathbf{p}} = R_{\mathbf{p}} + S_{\mathbf{q}}$ 
16          End
17      End
18  End
19  ! Calculate k-th nearest neighbor & fitness
20  For each  $\mathbf{p} \in \mathbf{R}^g$ 
21      For each  $\mathbf{q} \in \mathbf{R}^g$ 
22          Distance  $\mathbf{q} = \|\mathbf{p} - \mathbf{q}\|$ 
23      End
24      Sort Distance, 'Ascending'
25       $\sigma_p^k = \text{Distance } k, \quad k = \text{Integer } \sqrt{\mu + \lambda}$ 
26       $D_{\mathbf{p}} = \frac{1}{\sigma_p^k + 2}$ 
27       $F_{\mathbf{p}} = R_{\mathbf{p}} + D_{\mathbf{p}}$ 
28  End

```

Σχήμα 5.5: Ψευδοκώδικας της υπορουτίνας AssignFitness

5.2.3 Μέθοδος συντήρησης της ποικιλομορφίας των σχεδιασμών

Η υπορουτίνα EnvironmentalSelection (βλ. Σχήμα 5.4, γραμμή #8), καλείται ώστε να δημιουργηθεί το εξωτερικό αρχείο σχεδιασμών \mathbf{A}^{g+1} της επόμενης γενιάς. Στο Σχήμα 5.6 φαίνεται σε μορφή ψευδοκώδικα η υπορουτίνα EnvironmentalSelection.

Procedure: $\mathbf{A}^{g+1} = \text{EnvironmentalSelection } \mathbf{R}^g$

```

1   SortByFitness  $\mathbf{R}^g$ , 'ascending',  $\mathbf{A}^{g+1} = \emptyset$ 
2    $\mathbf{A}^{g+1} = \{i \mid i \in \mathbf{R}^g \wedge F(i) < 1\}$ 
3   If  $|\mathbf{A}^{g+1}| = \lambda$  Then
4       Return
5   End
6   If  $|\mathbf{A}^{g+1}| < \lambda$  Then
7        $\mathbf{A}^{g+1} = \mathbf{R}^g[1:\lambda]$ 
8   Else
9       For each  $\mathbf{p} \in \mathbf{A}^{g+1}$ 
10          For each  $\mathbf{q} \in \mathbf{A}^{g+1}$ 
11              distance  $\mathbf{p}, \mathbf{q} = \|\mathbf{p} - \mathbf{q}\|$ 
12          End
13      End
14      SortByRow distance, 'ascending'
15       $k = \text{Integer } \sqrt{|\mathbf{A}^{g+1}|}$ 
16       $\mathbf{A}^{g+1} = \text{SelectByDistance } \mathbf{A}^{g+1}, \text{distance } :, k$ 
17  End
```

Σχήμα 5.6: Ψευδοκώδικας της υπορουτίνας EnvironmentalSelection.

Αρχικά, γίνεται η ταξινόμηση του πληθυσμού \mathbf{R}^g ως προς την τιμή καταλληλότητας σε αύξουσα σειρά (γραμμή #1). Στη συνέχεια, αντιγράφονται στο εξωτερικό αρχείο όλοι οι σχεδιασμοί, οι οποίοι έχουν τιμή καταλληλότητας μικρότερη της μονάδας (γραμμή #2). Αν το πλήθος των σχεδιασμών στο εξωτερικό αρχείο \mathbf{A}^{g+1} είναι ίσο με λ , τότε επιστρέφουμε στο κυρίως πρόγραμμα. Ο έλεγχος αυτός γίνεται στις γραμμές #3-5. Αν το πλήθος των σχεδιασμών στο εξωτερικό αρχείο \mathbf{A}^{g+1} δεν είναι ίσος με λ , τότε υπάρχουν δύο περιπτώσεις:

- $|\mathbf{A}^{g+1}| < \lambda$: Στην περίπτωση αυτή, αντιγράφονται επιπλέον σχεδιασμοί από τον πληθυσμό \mathbf{R}^g , ώστε το πλήθος των σχεδιασμών στο εξωτερικό αρχείο \mathbf{A}^{g+1} να γίνει ίσο με λ (γραμμές #6-8).
- $|\mathbf{A}^{g+1}| > \lambda$: Στην περίπτωση αυτή ακολουθείται η διαδικασία που περιγράφεται στις γραμμές #8-18. Αρχικά, υπολογίζεται η απόσταση κάθε σχεδιασμού \mathbf{p} του εξωτερικού αρχείου \mathbf{A}^{g+1} , από τους υπόλοιπους σχεδιασμούς \mathbf{p} του \mathbf{A}^{g+1} (γραμμές #9-13). Οι αποστάσεις αποθηκεύονται στον πίνακα distance. Έπειτα, ταξινομείται κάθε σειρά του πίνακα distance, κατά αυξανόμενη απόσταση (γραμμή #14). Η k στήλη του πίνακα distance, περιέχει τις αποστάσεις του k κοντινότερου σχεδιασμού από τους πληθυσμούς του \mathbf{A}^{g+1} . Με βάση τη στήλη αυτή, επιλέγονται στη γραμμή #16 οι λ σχεδιασμοί με τις μεγαλύτερες αποστάσεις από τον k κοντινότερο σχεδιασμό.

Η υλοποίηση της υπορουτίνας EnvironmentalSelection διαφέρει από την αρχικά προτεινόμενη των Zitzler et al.₃ Η αρχικά προτεινόμενη υπορουτίνα είχε υπολογιστική πολυπλοκότητα της τάξης του $O(\mu + \lambda)$. Η υπολογιστική πολυπλοκότητα της υπορουτίνας EnvironmentalSelection όπως υλοποιήθηκε, έχει υπολογιστική πολυπλοκότητα τάξης $O(\mu + \lambda)$.

5.3 Ο αλγόριθμος c-MO-CMA-ES

Ο εξελικτικός αλγόριθμος c-MO-CMA-ES προτάθηκε από τους Hansen et al και ανήκει στο πεδίο των στρατηγικών εξέλιξης. Στον αλγόριθμο c-MO-CMA-ES, συνδυάζεται η ελιτιστική στρατηγική εξέλιξης $(1 + \lambda)$ -E-CMA-ES (βλ. Κεφάλαιο 3) με τις μεθόδους ταξινόμησης και διατήρησης της ποικιλομορφίας των σχεδιασμών του αλγόριθμου NSGA-II, ώστε να είναι δυνατή η αντιμετώπιση προβλημάτων βέλτιστου σχεδιασμού πολλαπλών αντικειμενικών κριτηρίων. Στην επόμενη παράγραφο δίνεται μια πλήρης περιγραφή του αλγόριθμου c-MO-CMA-ES.

5.3.1 Περιγραφή του αλγόριθμου c-MO-CMA-ES

Στον αλγόριθμο c-MO-CMA-ES, εκτελούνται παράλληλα λ_{MO} στρατηγικές εξέλιξης $(1 + \lambda)$ -E-CMA-ES (βλ. Κεφάλαιο 3), όπου λ είναι ο αριθμός των απογόνων που δημιουργούνται από κάθε μια στρατηγική εξέλιξης ξεχωριστά.

Ένας σχεδιασμός a_k στον αλγόριθμο c-MO-CMA-ES, αποτελείται από το διάνυσμα μεταβλητών σχεδιασμού \mathbf{x} , την παράμετρο \bar{p}_{succ} η οποία δηλώνει τον ρυθμό επιτυχιών κατά τη διάρκεια της έρευνας, το μήκος βήματος σ , το διάνυσμα πορείας της εξέλιξης \mathbf{p}_c και τον πίνακα συνδιασποράς \mathbf{C} , δηλαδή:

$$a_k^g = \mathbf{x}_k^g, \bar{p}_{succ,k}^g, \sigma_k^g, \mathbf{p}_{c,k}^g, \mathbf{C}_k^g \quad (5.7)$$

όπου g είναι η γενιά εξέλιξης. Στο βλ. Σχήμα 5.7 φαίνεται ο ψευδοκώδικας του εξελικτικού αλγόριθμου c-MO-CMA-ES.

Αρχικά, δημιουργούνται λ_{MO} σχεδιασμοί, οι οποίοι θα παίξουν τον ρόλο των γονέων κάθε μιας από τις λ_{MO} στρατηγικές εξέλιξης (γραμμή #1). Στη συνέχεια, από κάθε γονέα δημιουργούνται λ απόγονοι (γραμμές #4-7). Ο j απόγονος του k γονέα συμβολίζεται με a_{kj} και το διάνυσμα σχεδιασμού του με \mathbf{x}_{kj} . Για τη δημιουργία των απογόνων χρησιμοποιείται η σχέση (3.15), όπως και στον αλγόριθμο E-CMA-ES.

Στη γραμμή #9, δημιουργείται το σύνολο \mathbf{Q}^g το οποίο αποτελείται από όλους τους σχεδιασμούς, γονείς και απογόνους. Το σύνολο \mathbf{Q}^g , ταξινομείται στη συνέχεια από τη μέθοδο Fast-nondominated-sort (βλ. Σχήμα 5.7) του αλγόριθμου NSGA-II, ώστε να κατανομηθούν οι σχεδιασμοί στα επίπεδα ισχύος \mathbf{F}^g (γραμμή #11). Έπειτα, στις γραμμές #12-13 υπολογίζεται η απόσταση συσσώρευσης κάθε σχεδιασμού, και ταξινομείται κάθε επίπεδο ισχύος με τον τελεστή σύγκρισης της απόστασης συσσώρευσης (βλ. Σχήμα 5.7).

Στις γραμμές #14-19, υπολογίζονται το νέο μήκος βήματος των γονέων, και ο νέος πίνακας συνδιασποράς και το νέο διάνυσμα της πορείας της εξέλιξης για κάθε απόγονο. Στη γραμμή #15, υπολογίζεται το νέο μήκος βήματος κάθε γονέα, από την υπορουτίνα UpdateStepSize (βλ. Σχήμα 5.7). Ως αριθμός επιτυχιών $\lambda_{succ,k}$ του γονέα k , θεωρείται ο αριθμός των απογόνων που προήλθαν από τον γονέα k , και είτε βρίσκονται σε καλύτερο επίπεδο ισχύος σε σχέση με αυτόν, ή αν βρίσκονται στο ίδιο

επίπεδο ισχύος, βρίσκονται σε καλύτερη θέση, όπως ταξινομήθηκαν από τον τελεστή σύγκρισης αποστάσεων στη γραμμή #13. Στις γραμμές #16-19, υπολογίζεται το νέο μήκος βήματος (γραμμή #17), και νέος πίνακας συνδιασποράς (γραμμή #18) των απογόνων. Ο νέος πίνακας συνδιασποράς, υπολογίζεται από τη ρουτίνα UpdateCovariance.

Τέλος, στη γραμμή επιλέγονται οι πρώτοι λ_{MO} σχεδιασμοί του συνόλου \mathbf{Q}^g , ως οι νέοι γονείς της επόμενης γενιάς (γραμμή #22). Στη γραμμή #25, ελέγχεται αν ικανοποιείται το κριτήριο τερματισμού του αλγόριθμου. Αν το κριτήριο ικανοποιείται, η διαδικασία εξέλιξης σταματά, με τους νέους γονείς να αποτελούν την προσέγγιση του Pareto-βέλτιστου συνόλου σχεδιασμών. Ως κριτήριο τερματισμού, συνήθως επιλέγεται ο μέγιστος αριθμός γενεών εξέλιξης.

Ο εξελικτικός αλγόριθμος c-MO-CMA-ES διατηρεί όλα τα πλεονεκτήματα και μειονεκτήματα του αλγόριθμου στρατηγικής εξέλιξης E-CMA-ES.

Algorithm: $\lambda_{MO} \times 1 + \lambda$ -c-MO-CMA-ES

```
1    $g = 0$ , initialize  $a_k^g$  for  $k = 1, \dots, \lambda_{MO}$ 
2   repeat
3     for  $k = 1, \lambda_{MO}$  do
4       for  $j = 1, \lambda$  do
5          $a_{kj}^g = a_k^g$ 
6          $\mathbf{x}_{kj}^{g+1} \sim N(\mathbf{x}_k^g, \sigma_k^{g^2} \mathbf{C}_k^g)$ 
7       end
8     end
9      $\mathbf{Q}^g =$  whole population
10    ! NSGA-II Routines
11     $\mathbf{F}^g =$  fast-nondomination-sort  $\mathbf{Q}^g$ 
12    crowding-distance-assignment  $\mathbf{F}^g$ 
13    crowded-comparison-operator  $\mathbf{F}^g$ 
14    for  $k = 1, \lambda_{MO}$  do
15      UpdateStepSize  $\left( a_k^g, \frac{\lambda_{succ,k}^{g+1}}{\lambda} \right)$ 
16      for  $j = 1, \lambda$  do
17        UpdateStepSize  $\left( a_{kj}^g, \frac{\lambda_{succ,k}^{g+1}}{\lambda} \right)$ 
18        UpdateCovariance  $\left( a_{kj}^{g+1}, \frac{\mathbf{x}_{kj}^{g+1} - \mathbf{x}_k^g}{\sigma_k^g} \right)$ 
19      end
20    end
21    for  $k = 1, \lambda_{MO}$  do
22       $a_k^{g+1} = Q^g \ k$ 
23    end
24     $g = g + 1$ 
25  until termination_condition
```

Σχήμα 5.7: Ψευδοκώδικας του αλγόριθμου c-MO-CMA-ES.

6 ΕΦΑΡΜΟΓΕΣ

6.1 Προβλήματα βελτιστοποίησης με ένα κριτήριο

6.1.1 Μαθηματικά προβλήματα

6.1.1.1 Πρόβλημα S-6ACT

Το πρόβλημα βελτιστοποίησης S-6ACT διατυπώνεται από την παρακάτω σχέση:

$$\begin{aligned} \min F(\mathbf{x}) = & x_1^2 + x_2^2 + x_1x_2 - 14x_1 - 16x_2 + x_3 - 10x_4^2 + 4x_4 - 5x_5^2 \\ & + x_5 - 3x_6^2 + 2x_6 - 1x_7^2 + 5x_7^2 + 7x_8 - 11x_8^2 + 2x_9 - 10x_9^2 \\ & + x_{10} - 7x_{10}^2 + 45 \end{aligned} \quad (6.1)$$

με τους παρακάτω ανισοτικούς περιορισμούς:

$$\begin{aligned} \text{subject to: } g_1(\mathbf{x}) = & 105 - 4x_1 - 5x_2 + 3x_7 - 9x_8 \geq 0 \\ g_2(\mathbf{x}) = & -10x_1 + 8x_2 + 17x_7 - 2x_8 \geq 0 \\ g_3(\mathbf{x}) = & 8x_1 - 2x_2 + 17x_7 - 2x_8 \geq 0 \\ g_4(\mathbf{x}) = & -3x_1 - 2x_2 - 4x_3 - 3x_4^2 - 2x_5^2 + 7x_4 + 120 \geq 0 \\ g_5(\mathbf{x}) = & -5x_1^2 - 8x_2 - x_3 - 6x_4^2 + 2x_4 + 40 \geq 0 \\ g_6(\mathbf{x}) = & -x_1^2 - 2(x_2 - 2)^3 + 2x_1x_2 - 14x_5 + 6x_6 \geq 0 \\ g_7(\mathbf{x}) = & -0.5(x_1 - 8)^3 - 2(x_2 - 4)^3 + 3x_5^2 + x_6 + 30 \geq 0 \\ g_8(\mathbf{x}) = & 3x_1 - 6x_2 - 12(x_3 - 8)^3 + 7x_{10} \geq 0 \\ & -10 \leq x_i \leq 10, i = 1, \dots, 10 \end{aligned} \quad (6.2)$$

Πρόκειται δηλαδή, για ένα πρόβλημα με δέκα μεταβλητές σχεδιασμού και οκτώ συναρτήσεις ανισοτικών περιορισμών. Στον πίνακα Π 6.1, δίνονται οι παράμετροι των εξελικτικών αλγορίθμων A-ES, CMA-ES και ECMA-ES, όπως χρησιμοποιήθηκαν για την επίλυση του προβλήματος S-6ACT. Για τους αλγόριθμους A-ES και ECMA-ES, χρησιμοποιήθηκε πληθυσμός με έναν γονέα και 14 απόγονους, ενώ για τον αλγόριθμο CMA-ES χρησιμοποιήθηκε πληθυσμός 10 απογόνων και 5 γονέων.

Π 6.1: Σύγκριση των αποτελεσμάτων για το πρόβλημα S-6ACT

Algorithm	μ	λ	Selection	Generations	Objective function		
					Best	Mean	Std. Dev
A-ES	1	14	+	667	14.962	49.0379	76.46656
ECMA-ES	1	14	+	667	14.4361	15.1669	0.784223
CMA-ES	5	10	,	1000	14.25713	14.26806	0.012222

Για να προκύψουν τα αναφερόμενα στον πίνακα Π 6.1 στατιστικά αποτελέσματα (μέση τιμή και τυπική απόκλιση), η διαδικασία εξέλιξης εκτελέστηκε 20 φορές. Στον πίνακα Π 6.2, συγκρίνονται τα αποτελέσματα με το καλύτερο αποτέλεσμα που αναφέρεται στη βιβλιογραφία [DEB00].

Όπως φαίνεται από τον πίνακα Π 6.2, οι τρεις αλγόριθμοι βρήκαν καλύτερη λύση από την έως τώρα βέλτιστη θεωρούμενη λύση. Από τους τρεις αλγόριθμους που χρησιμοποιήθηκαν στο παραδοτέο Π1.1, την καλύτερη λύση βρήκε ο CMA-ES.

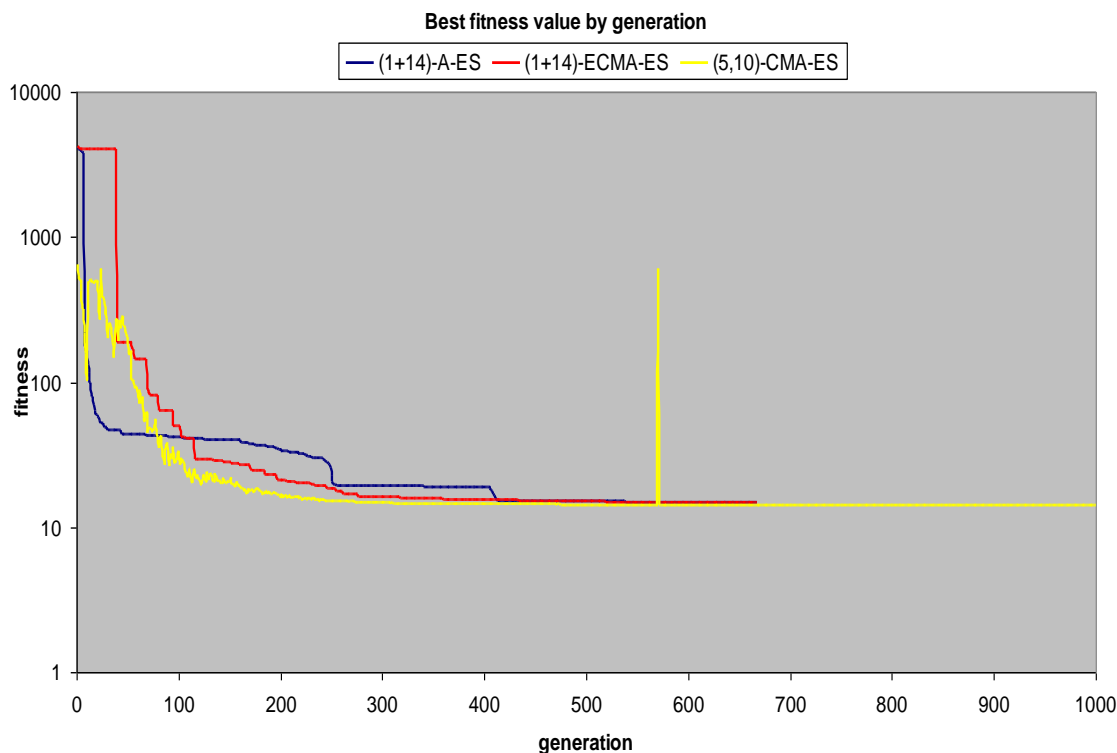
Ο αλγόριθμος CMA-ES είχε την καλύτερη συμπεριφορά βάσει των στατιστικών στοιχείων του πίνακα Π 6.1 αφού εκτός από τη μικρότερη μέση τιμή της αντικειμενικής συνάρτησης, η τυπική απόκλιση των τιμών αυτής ήταν τάξεις μεγέθους μικρότερη από αυτές των υπόλοιπων δύο

αλγόριθμων. Θα πρέπει να σημειωθεί, πως και οι 20 διαδικασίες εξέλιξης που εκτελέστηκαν για κάθε αλγόριθμο, οδήγησαν τελικά σε εφικτό σχεδιασμό.

Π 6.2: Σύγκριση των αποτελεσμάτων με τη βιβλιογραφία

x	[DEB00]	A-ES	ECMA-ES	CMA-ES
x ₁	2.171996	1.5859	1.69969025	1.5760762
x ₂	2.363683	2.8712	2.69470857	2.73198686
x ₃	8.773926	8.7952	8.78324484	8.79176326
x ₄	5.095984	5.0471	4.9932193	5.05953094
x ₅	0.9906548	1.1745	1.06756136	0.97675319
x ₆	1.430574	1.9129	1.60724838	1.43642955
x ₇	1.321644	0.74894	0.87381671	0.78377817
x ₈	9.828726	9.6163	9.70543789	9.70967668
x ₉	8.280092	9.7648	9.76549622	9.77448852
x ₁₀	8.375927	7.1255	6.92903178	7.06425525
F	24.306209	14.962	14.4361	14.25713

Στο Σχήμα 6.1 δίνεται η γραφική παράσταση της αντικειμενικής συνάρτησης της καλύτερης γενιάς εξέλιξης, για κάθε έναν από τους τρεις αλγόριθμους που χρησιμοποιήθηκαν.



Σχήμα 6.1: Μεταβολή της αντικειμενικής συνάρτησης στο πρόβλημα S-6ACT

Από το Σχήμα 6.1 φαίνεται πως ο εξελικτικός αλγόριθμος CMA-ES συγκλίνει περίπου 100 γενιές γρηγορότερα στην περιοχή του βέλτιστου, σε σχέση με τους αλγόριθμους A-ES και ECMA-ES. Πριν τις 600 γενιές εξέλιξης, φαίνεται μια στιγμιαία απόκλιση της τιμής της αντικειμενικής συνάρτησης στην εξέλιξη με τον αλγόριθμο CMA-ES. Αυτό οφείλεται στο γεγονός πως ο αλγόριθμος CMA-ES εν είναι ελιτιστικός, οπότε είναι δυνατό κατά την διάρκεια μιας γενιάς εξέλιξης, να μεταβούμε σε λύση χειρότερη της προηγούμενης γενιάς. Αυτό δεν συμβαίνει με τους ελιστικούς αλγόριθμους A-ES και ECMA-ES.

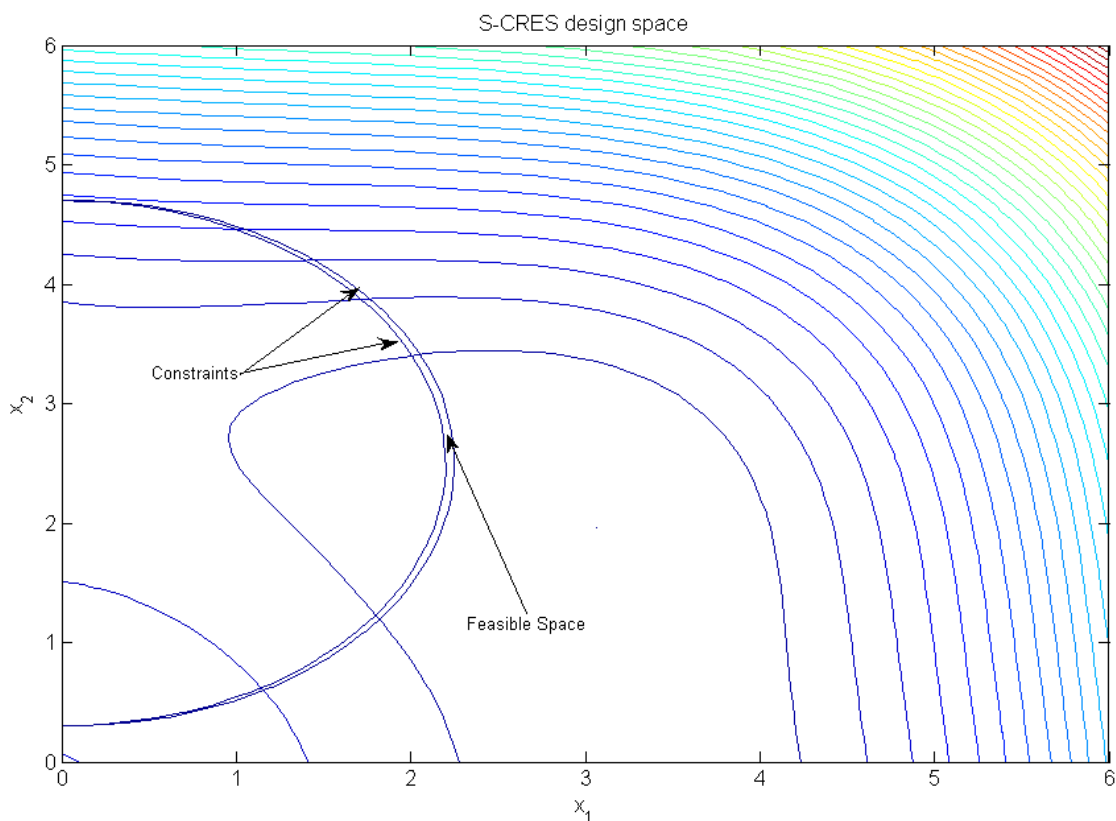
Παρατηρούμε επίσης, πως αν και ο αλγόριθμος ECMA-ES δεν ξεκινά όσο καλά ξεκινά ο αλγόριθμος A-ES τη διαδικασία εξέλιξης, στη συνέχεια συγκλίνει κοντά στο βέλτιστο γρηγορότερα.

6.1.1.2 Πρόβλημα S-CRES

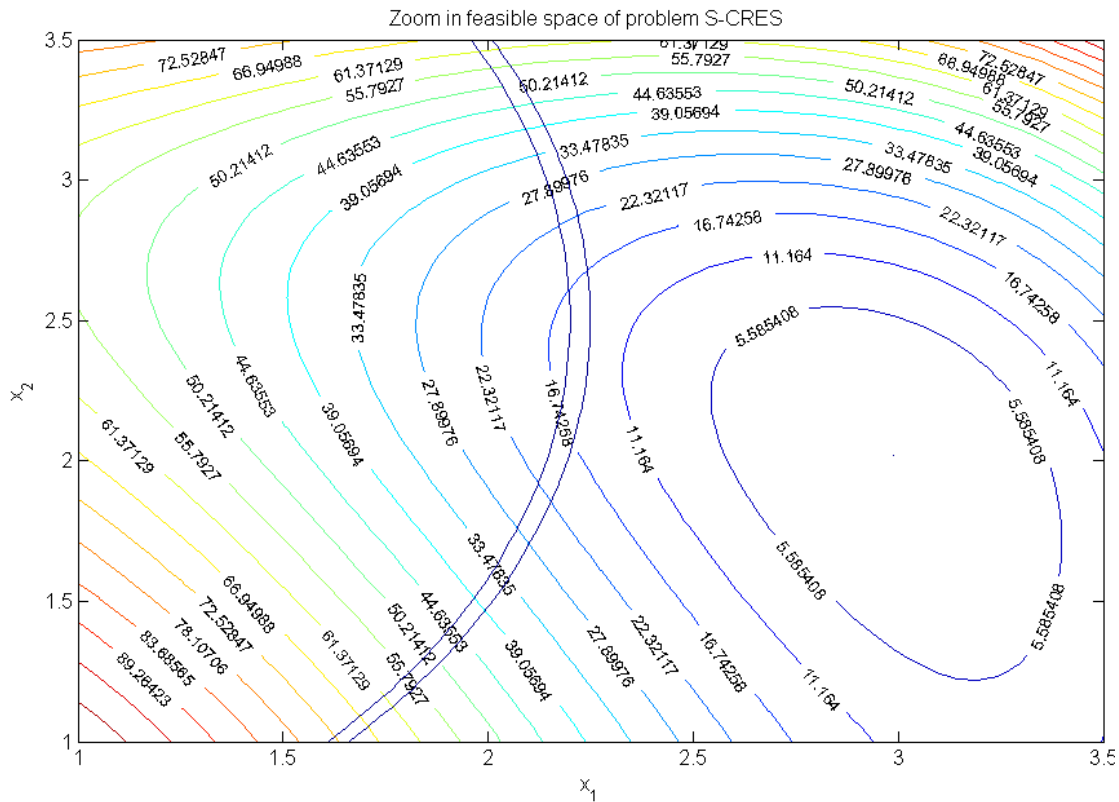
Το πρόβλημα S-CRES έχει προταθεί από τον Deb [DEB00]. Χαρακτηρίζεται από δύο μεταβλητές σχεδιασμού και δύο συναρτήσεις περιορισμών, και διατυπώνεται από την παρακάτω σχέση:

$$\begin{aligned} \min F \mathbf{x} &= x_1^2 + x_2 - 11^2 + x_1 + x_2^2 - 7^2 \\ \text{subject to: } g_1(\mathbf{x}) &= 4.84 - (x_1 - 0.05)^2 - (x_2 - 2.5)^2 \geq 0 \\ g_2(\mathbf{x}) &= x_1^2 + (x_2 - 2.5)^2 - 4.84 \geq 0 \\ 0 &\leq x_1 \leq 6, 0 \leq x_2 \leq 6 \end{aligned} \quad (6.3)$$

Στο Σχήμα 6.2 απεικονίζεται ο εφικτός και ο μη εφικτός χώρος του προβλήματος. Ο εφικτός χώρος στο πρόβλημα βελτιστοποίησης S-CRES, αντιστοιχεί στο 0.7% του συνολικού χώρου που ορίζουν οι δύο μεταβλητές σχεδιασμού. Στο Σχήμα 6.2 φαίνεται η περιοχή γύρω από το βέλτιστο σχεδιασμό και οι τιμές των ισογραμμών της αντικειμενικής συνάρτησης.



Σχήμα 6.2: Εφικτός και μή-εφικτός χώρος του προβλήματος S-CRES



Σχήμα 6.3: Μεγεθυμένος χώρος γύρω από το βέλτιστο σημείο.

Στον πίνακα Π 6.3 δίνονται οι παράμετροι των εξελικτικών αλγόριθμων A-ES, CMA-ES και ECMA-ES, όπως χρησιμοποιήθηκαν για την επίλυση του προβλήματος S-CRES. Για τους αλγόριθμους A-ES και ECMA-ES, χρησιμοποιήθηκε πληθυσμός με έναν γονέα και 8 απόγονους, ενώ για τον αλγόριθμο CMA-ES χρησιμοποιήθηκε πληθυσμός με 6 απόγονους και 3 γονείς.

Π 6.3: Σύγκριση των αποτελεσμάτων για το πρόβλημα S-6ACT

Algorithm	μ	λ	Selection	Generations	Objective function		
					Best	Mean	Std. Dev
A-ES	1	8	+	112	13.59085	13.6897	0.346575
ECMA-ES	1	8	+	112	13.59087	13.60961	0.053559
CMA-ES	3	6	,	167	13.59084	13.59088	0.000182

Για να προκύψουν τα αναφερόμενα στον πίνακα Π 6.3 στατιστικά αποτελέσματα (μέση τιμή και τυπική απόκλιση), η διαδικασία εξέλιξης εκτελέστηκε 50 φορές. Θα πρέπει να σημειωθεί πως όλες οι διαδικασίες εξέλιξης για κάθε αλγόριθμο, οδήγησαν τελικά σε εφικτό σχεδιασμό.

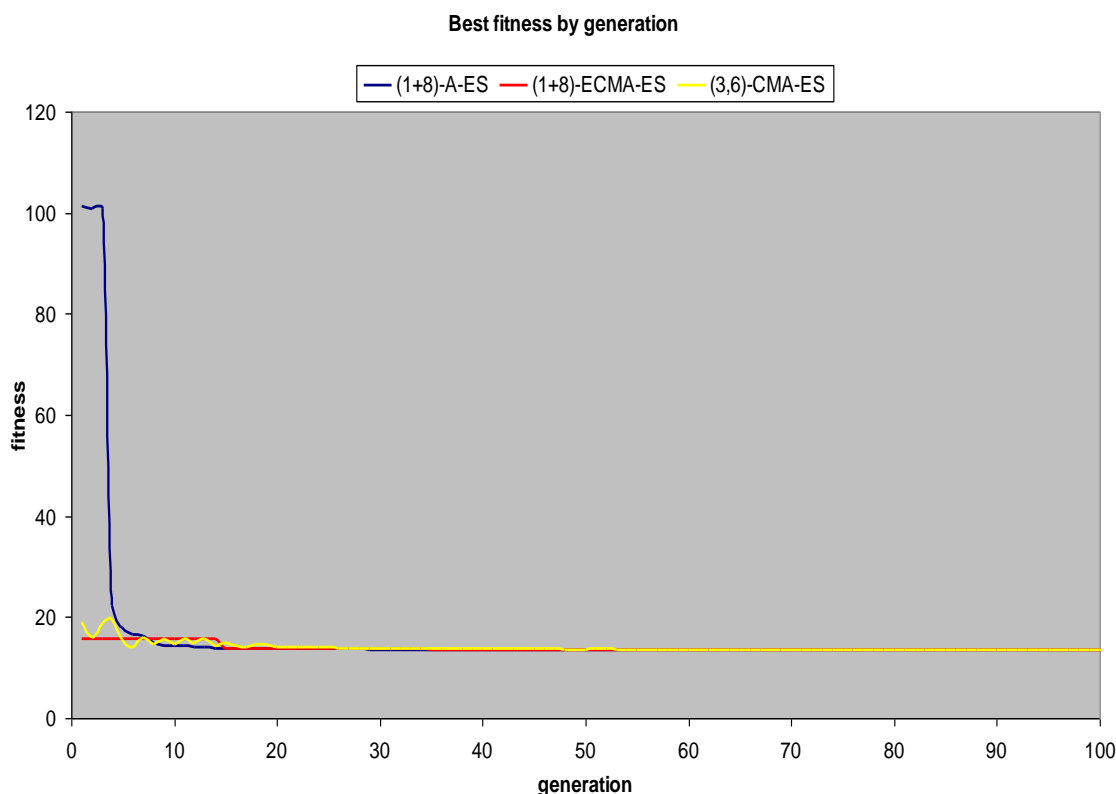
Στον πίνακα Π 6.4, συγκρίνονται τα αποτελέσματα των αλγόριθμων A-ES, ECMA-ES και CMA-ES, με το καλύτερο αποτέλεσμα που αναφέρεται στη βιβλιογραφία [DEB00].

Π 6.4: Σύγκριση των αποτελεσμάτων με τη βιβλιογραφία

x	[DEB00]	A-ES	ECMA-ES	CMA-ES
x_1	2.246826	2.24684071	2.24681071	2.24682593
x_2	2.381865	2.38214132	2.38159684	2.38186456
F	13.59085	13.5908459	13.5908676	13.5908406

Από τον πίνακα Π6.4 προκύπτει πως οι τρεις αλγόριθμοι που χρησιμοποιήθηκαν βρήκαν την περιοχή του βέλτιστου σχεδιασμού. Ο αλγόριθμος CMA-ES είχε την καλύτερη συμπεριφορά, αφού η τυπική απόκλιση των τιμών της αντικειμενικής συνάρτησης στο τέλος κάθε διαδικασίας εξέλιξης ήταν τάξεις μεγέθους μικρότερη από αυτή των υπόλοιπων δύο αλγόριθμων.

Στο Σχήμα 6.4 δίνεται η γραφική παράσταση της μεταβολής της αντικειμενικής συνάρτησης της καλύτερης διαδικασίας εξέλιξης, για κάθε έναν από τους αλγόριθμους A-ES, ECMA-ES και CMA-ES, στις πρώτες 100 γενιές εξέλιξης.



Σχήμα

6.4: Μεταβολή της αντικειμενικής συνάρτησης στο πρόβλημα S-6ACT

6.1.1.3 Πρόβλημα S-0.5F

Το πρόβλημα S-0.5F [DEB00] χαρακτηρίζεται από 7 μεταβλητές σχεδιασμού και 4 συναρτήσεις περιορισμών, και διατυπώνεται από την παρακάτω σχέση:

$$\begin{aligned}
 \min F \mathbf{x} &= x_1 - 10^2 + 5x_2 - 12^2 + x_3^4 + 3x_4 - 11^2 + 10x_5^6 \\
 &\quad + 7x_6^2 + x_7^4 - 4x_6x_7 \\
 \text{subject to: } g_1 \mathbf{x} &= 127 - 2x_1^2 - 3x_2^2 - x_3 - 4x_4^2 - 5x_5^2 \geq 0 \\
 g_2 \mathbf{x} &= 282 - 7x_1 - 3x_2 - 10x_3^2 - x_4 - x_5 \geq 0 \\
 g_3 \mathbf{x} &= 196 - 23x_1 - x_2^2 - 6x_6^2 + 8x_7 \geq 0 \\
 g_4 \mathbf{x} &= -4x_1^2 - x_2^2 + 3x_1x_2 - 2x_3^2 - 5x_6 + 11x_7 \geq 0 \\
 &\quad -10 \leq x_i \leq 10, i = 1, \dots, 7
 \end{aligned} \tag{6.4}$$

Στο συγκεκριμένο πρόβλημα, μόνο το 0.5% του χώρου που ορίζεται από τις μεταβλητές σχεδιασμού είναι εφικτός.

Π 6.5: Σύγκριση των αποτελεσμάτων για το πρόβλημα S-0.5F

Algorithm	μ	λ	Selection	Generations	Objective function		
					Best	Mean	Std. Dev
A-ES	1	12	+	2000	680.7721	705.7945	87.6404
ECMA-ES	1	12	+	2000	680.6848	681.5228	0.431397
CMA-ES	4	9	,	1000	680.6301	680.6301	8.19E-07

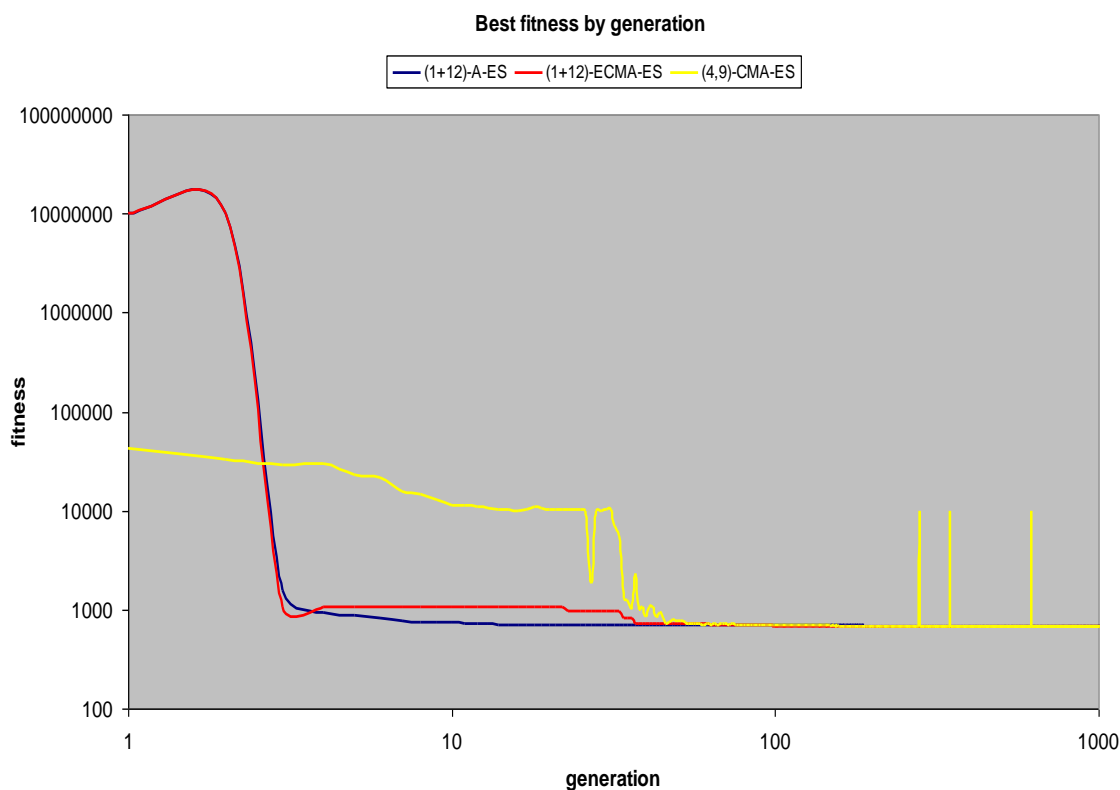
Στον πίνακα Π6.5 δίνονται οι παράμετροι των εξελικτικών αλγόριθμων A-ES, CMA-ES και ECMA-ES, όπως χρησιμοποιήθηκαν για την επίλυση του προβλήματος S-0.5F. Για τους αλγόριθμους A-ES και ECMA-ES, χρησιμοποιήθηκε πληθυσμός με έναν γονέα και 12 απόγονους, ενώ για τον αλγόριθμο CMA-ES χρησιμοποιήθηκε πληθυσμός με 9 απόγονους και 4 γονείς.

Για να προκύψουν τα αναφερόμενα στον πίνακα Π6.5 στατιστικά αποτελέσματα (μέση τιμή και τυπική απόκλιση), η διαδικασία εξέλιξης εκτελέστηκε 50 φορές. Θα πρέπει να σημειωθεί πως όλες οι διαδικασίες εξέλιξης για κάθε αλγόριθμο, οδήγησαν τελικά σε εφικτό σχεδιασμό.

Στον πίνακα Π6.6, συγκρίνονται τα αποτελέσματα των αλγόριθμων A-ES, ECMA-ES και CMA-ES, με το καλύτερο αποτέλεσμα που αναφέρεται στη βιβλιογραφία [DEB00].

Π 6.6: Τελικά διανύσματα σχεδιασμού

x	[DEB00]	A-ES	ECMA-ES	CMA-ES
x ₁	2.330499	2.32037995	2.2994298	2.3305007
x ₂	1.951372	1.96762456	1.94707593	1.95137345
x ₃	-0.4775414	-0.28180306	-0.46874672	-0.477539016
x ₄	4.3657226	4.31912891	4.38280688	4.36572268
x ₅	-0.624487	-0.61579912	-0.611883291	-0.624484386
x ₆	1.038131	1.05747015	1.00182303	1.03812497
x ₇	1.594227	1.56075925	1.54160756	1.59422479
F	680.63	680.7721	680.6848	680.6301



Σχήμα 6.5: Μεταβολή της αντικειμενικής συνάρτησης στο πρόβλημα S-0.5F

Από τον πίνακα Π6.6 προκύπτει πως αν και οι τρεις αλγόριθμοι που χρησιμοποιήθηκαν βρήκαν την περιοχή του βέλτιστου σχεδιασμού, μόνο ο αλγόριθμος CMA-ES βρήκε τον ολικά βέλτιστο σχεδιασμό. Ο αλγόριθμος CMA-ES είχε και σε αυτό το πρόβλημα την καλύτερη συμπεριφορά, αφού η τυπική απόκλιση των τιμών της αντικειμενικής συνάρτησης στο τέλος κάθε διαδικασίας εξέλιξης είναι σχεδόν μηδενική. Στο ?? δίνεται η γραφική παράσταση της μεταβολής της αντικειμενικής συνάρτησης της καλύτερης διαδικασίας εξέλιξης, για κάθε έναν από τους αλγόριθμους A-ES, ECMA-ES και CMA-ES. Οι άξονες του σχήματος είναι σε λογαριθμική κλίμακα.

6.1.1.4 Πρόβλημα S-HIM

Το πρόβλημα S-0.5F [DEB00] χαρακτηρίζεται από 5 μεταβλητές σχεδιασμού και 6 συναρτήσεις ανισοτικών περιορισμών, και διατυπώνεται από την παρακάτω σχέση:

$$\begin{aligned}
 \min F \mathbf{x} &= 5.3578547x_3^2 + 0.8356891x_1x_5 + 37.293239x_1 - 40792.141 \\
 \text{subject to:} \\
 g_1(\mathbf{x}) &= 85.334407 + 0.0056858x_2x_5 + 0.0006262x_1x_4 \\
 &\quad - 0.0022053x_3x_5 \geq 0 \\
 g_2(\mathbf{x}) &= 92 - g_1(\mathbf{x}) \geq 0 \\
 g_3(\mathbf{x}) &= 80.51249 + 0.0071317x_2x_5 + 0.0029955x_1x_2 \\
 &\quad + 0.0021813x_3^2 - 90 \geq 0 \\
 g_4(\mathbf{x}) &= 20 - g_3(\mathbf{x}) \geq 0 \\
 g_5(\mathbf{x}) &= 9.300961 + 0.0047026x_3x_5 + 0.0012547x_1x_3 \\
 &\quad + 0.0019085x_3x_4 - 20 \geq 0 \\
 g_6(\mathbf{x}) &= 5 - g_4(\mathbf{x}) \geq 0 \\
 78 \leq x_1 \leq 102, \quad 33 \leq x_2 \leq 45, \quad 27 \leq x_i \leq 45, i = 3, 4, 5
 \end{aligned} \tag{6.5}$$

Στον πίνακα Π6.7 δίνονται οι παράμετροι των εξελικτικών αλγόριθμων A-ES, CMA-ES και ECMA-ES, όπως χρησιμοποιήθηκαν για την επίλυση του προβλήματος S-HIM. Για τους αλγόριθμους A-ES και ECMA-ES, χρησιμοποιήθηκε πληθυσμός με έναν γονέα και 11 απόγονους, ενώ για τον αλγόριθμο CMA-ES χρησιμοποιήθηκε πληθυσμός με 8 απόγονους και 4 γονείς.

Π 6.7: Αποτελέσματα για το πρόβλημα S-0.5F.

Algorithm	μ	λ	Selection	Generations	Objective function		
					Best	Mean	Std. Dev
A-ES	1	11	+	417	-30665.5	-30190	546.1736
ECMA-ES	1	11	+	417	-30665.5	-30477.6	257.4024
CMA-ES	4	8	,	625	-25273.7	-24258.6	5023.252

Για να προκύψουν τα αναφερόμενα στον πίνακα Π6.7 στατιστικά αποτελέσματα (μέση τιμή και τυπική απόκλιση), η διαδικασία εξέλιξης εκτελέστηκε 50 φορές. Θα πρέπει να σημειωθεί πως όλες οι διαδικασίες εξέλιξης για κάθε αλγόριθμο, οδήγησαν τελικά σε εφικτό σχεδιασμό.

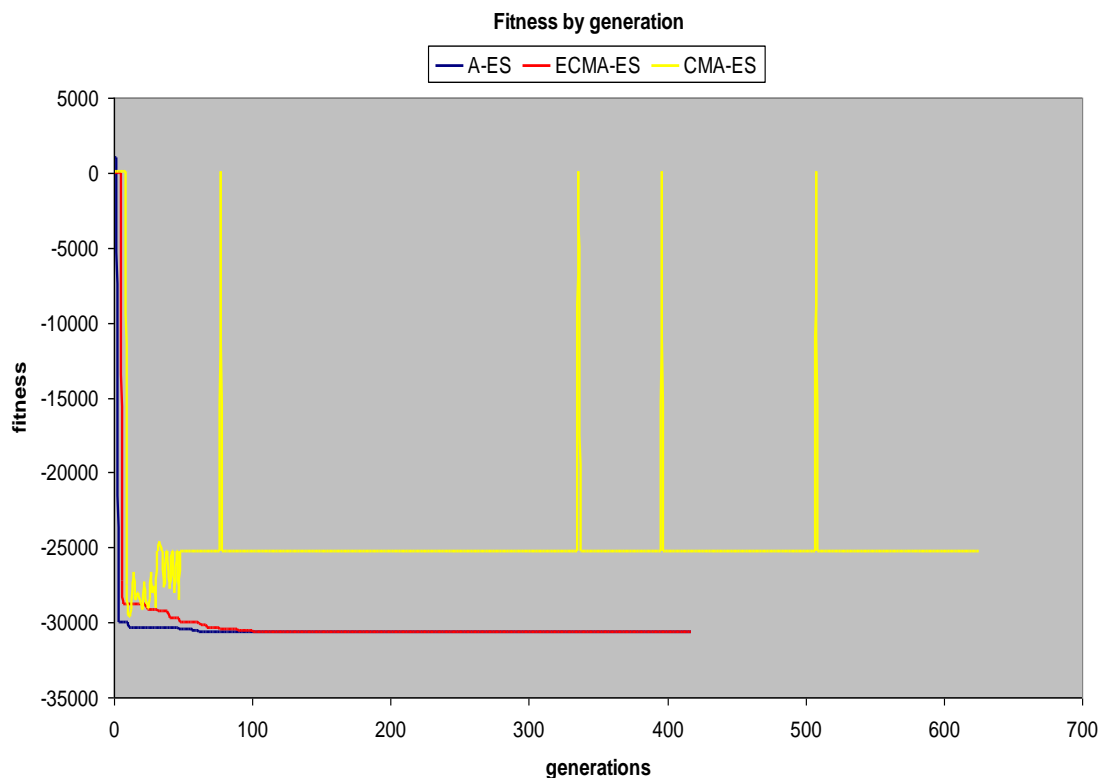
Στον πίνακα Π6.8, τα διανύσματα σχεδιασμού της καλύτερης γενιάς εξέλιξης κάθε αλγόριθμου.

Π 6.8: Τελικά διανύσματα σχεδιασμού.

x	A-ES	ECMA-ES	CMA-ES
x_1	78	78	78
x_2	33	33	33
x_3	29.9955587	29.9952498	45
x_4	45	45	45
x_5	36.7755144	36.7758306	27
F	-30665.4602	-30665.5	-25273.7

Στη βιβλιογραφία [COE02] αναφέρεται ως βέλτιστη τιμή της αντικειμενικής συνάρτησης η τιμή -31005.7966, η οποία επιτεύχθηκε με υπολογιστικό κόστος 900000 υπολογισμούς της αντικειμενικής συνάρτησης. Από τον πίνακα Π6.8 προκύπτει πως από τους τρεις αλγόριθμους που χρησιμοποιήθηκαν, οι αλγόριθμοι A-ES και ECMA-ES έφτασαν σχετικά κοντά στη βέλτιστη λύση, με δεδομένο το μικρό αριθμό γενεών εξέλιξης που χρησιμοποιήθηκαν. Αντίθετα, ο αλγόριθμος CMA-ES απέτυχε σε αυτό το πρόβλημα να βρει την περιοχή της βέλτιστης λύσης,

όπως φαίνεται και από το ?? στο οποίο απεικονίζεται η γραφική παράσταση της μεταβολής της αντικειμενικής συνάρτησης της καλύτερης διαδικασίας εξέλιξης, για κάθε έναν από τους αλγόριθμους A-ES, ECMA-ES και CMA-ES.



Σχήμα 6.6: Μεταβολή της αντικειμενικής συνάρτησης στο πρόβλημα S-0.5F

Από το Σχήμα 6.6 φαίνεται πως αρχικά ο αλγόριθμος CMA-ES προσεγγίζει την περιοχή του βέλτιστου, αλλά στη συνέχεια αποκλίνει από την περιοχή αυτή. Η συμπεριφορά αυτή οφείλεται στην ανεξέλεγκτη αύξηση της τιμής του μήκους βήματος σ , που φτάνει σε τιμές της τάξης του 10^{80} . Η αύξηση αυτή του καθολικού βήματος σ έχει ως αποτέλεσμα να «κολλήσει» η έρευνα στα όρια των μεταβλητών σχεδιασμού (βλ. σχέση (6.5)), όπως φαίνεται και από το τελικό διάγραμμα σχεδιασμού στον πίνακα Π6.8.

6.2 Προβλήματα βελτιστοποίησης με πολλά κριτήρια

6.2.1 Μαθηματικά προβλήματα

6.2.1.1 Πρόβλημα KUR

Η διατύπωση του προβλήματος βελτιστοποίησης KUR δίνεται από τη σχέση:

$$\min \begin{cases} f_1(x_1, x_2, x_3) = \sum_{i=1}^2 -10 \exp(-0.2 \sqrt{x_1^2 + x_{i+1}^2}) \\ f_2(x_1, x_2, x_3) = \sum_{i=1}^3 |x_i|^{0.8} + 5 \sin x_i^3 \end{cases} \quad (6.6)$$

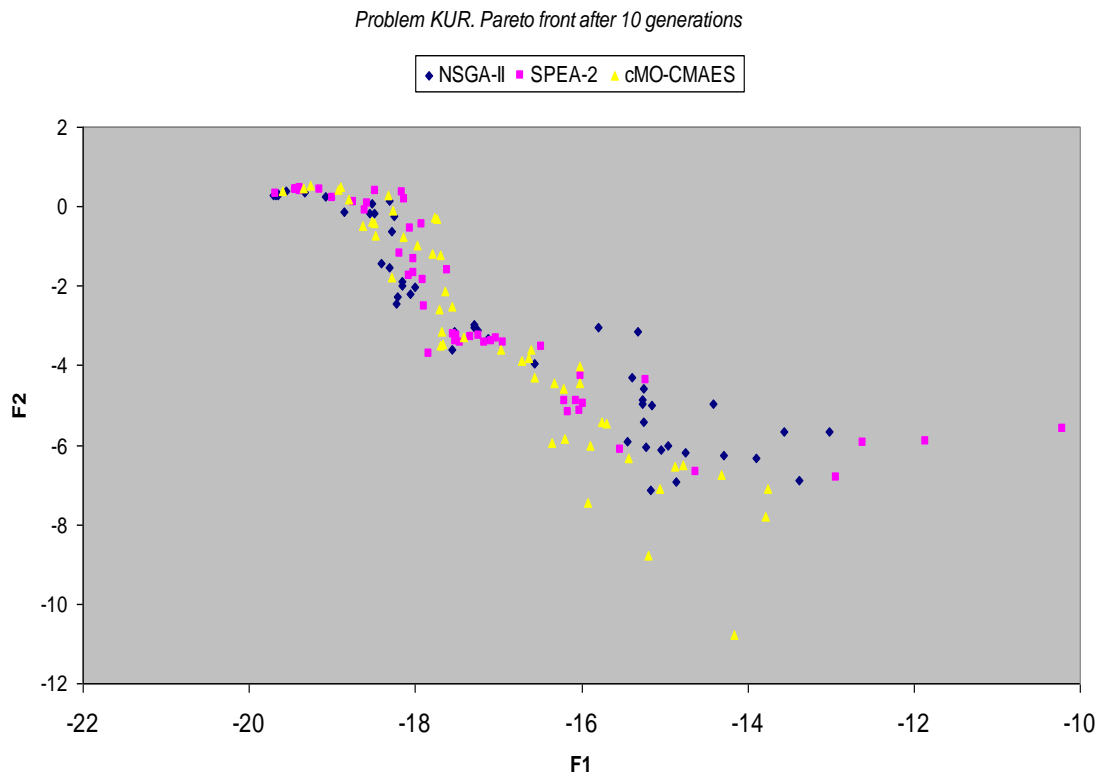
subject to: $-5 \leq x_i \leq 5, i = 1, 2, 3$

Το πρόβλημα KUR χαρακτηρίζεται από δύο αντικειμενικές συναρτήσεις και τρεις μεταβλητές σχεδιασμού και είναι μή-κυρτό (non-convex). Επίσης, το Pareto-βέλτιστο σύνολο λύσεων του προβλήματος παρουσιάζει και ασυνέχεια.

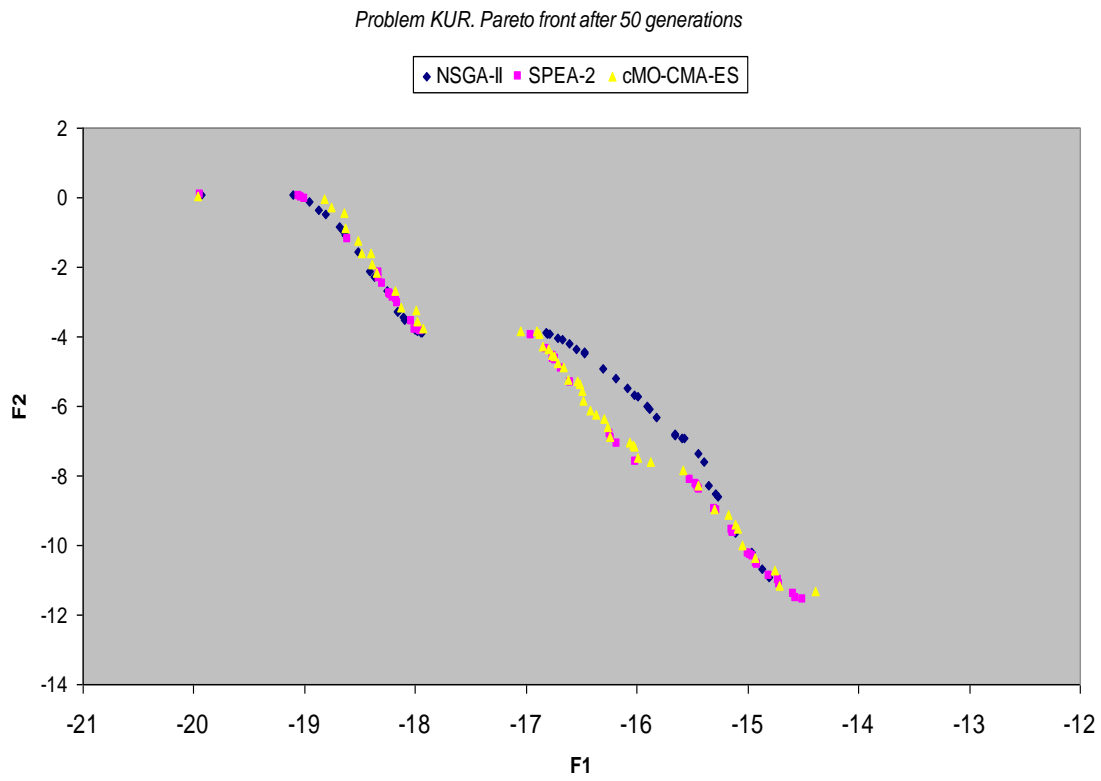
Για την εύρεση του Pareto-βέλτιστου συνόλου των σχεδιασμών, εκτελέστηκαν 500 γενιές εξέλιξης, χρησιμοποιώντας τους αλγόριθμους A-ES/NSGA-II, A-ES/SPEA-2 και cMO-CMA-ES (βλ. Κεφάλαιο 5). Ο αλγόριθμος A-ES/NSGA-II εκτελέστηκε με πληθυσμό γονέων $\mu = 50$, και

πληθυσμό απογόνων $\lambda = 50$. Ο αλγόριθμος A-ES/SPEA-2 εκτελέστηκε με πληθυσμό 50 σχεδιασμών και μέγεθος εξωτερικού αρχείου 50 ατόμων. Ομοίως, ο αλγόριθμος cMO-CMA-ES εκτελέστηκε με αριθμό παράλληλων στρατηγικών εξέλιξης $\lambda_{MO} = 50$ της μορφής 1+1 -E-CMA-ES. Και οι τρεις αλγόριθμοι εκτελέστηκαν για 50000 υπολογισμούς των αντικειμενικών συναρτήσεων.

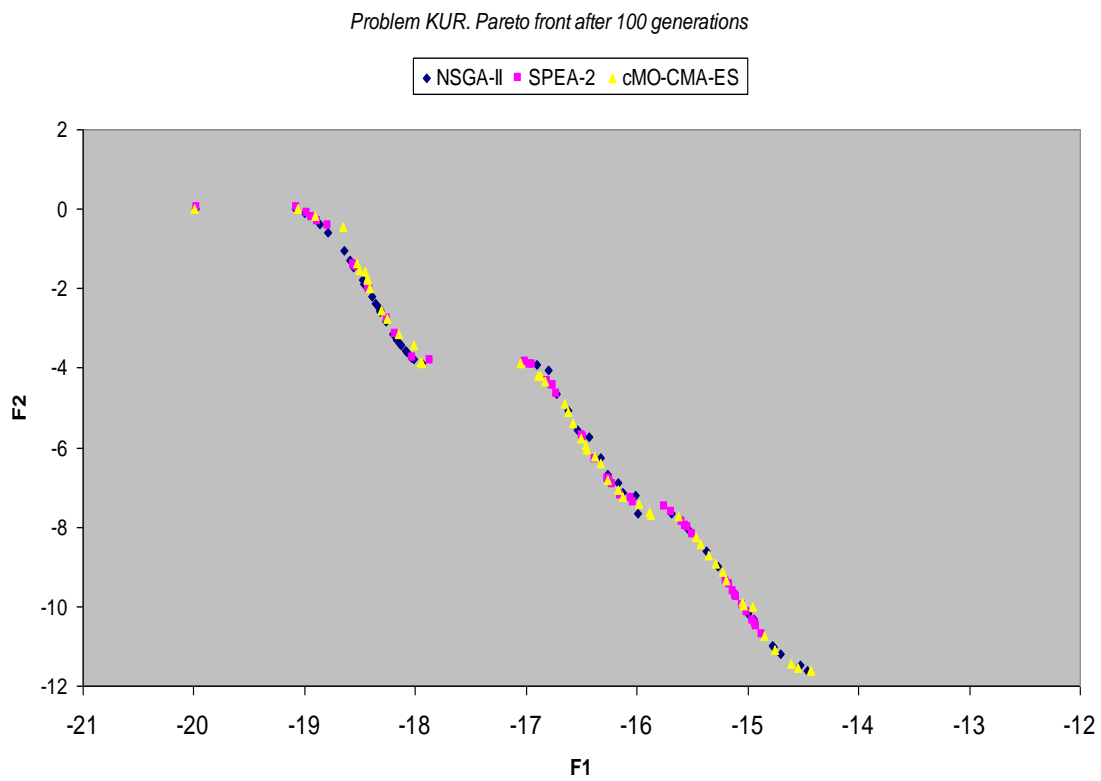
Στα σχήματα 6.7 ως 6.11 απεικονίζεται η σύγκλιση στο Pareto-βέλτιστο σύνολο των σχεδιασμών, των τριών αλγόριθμων, για 10, 50, 100, 200 και 500 γενιές εξέλιξης.



Σχήμα 6.7: Μέτωπα Pareto.

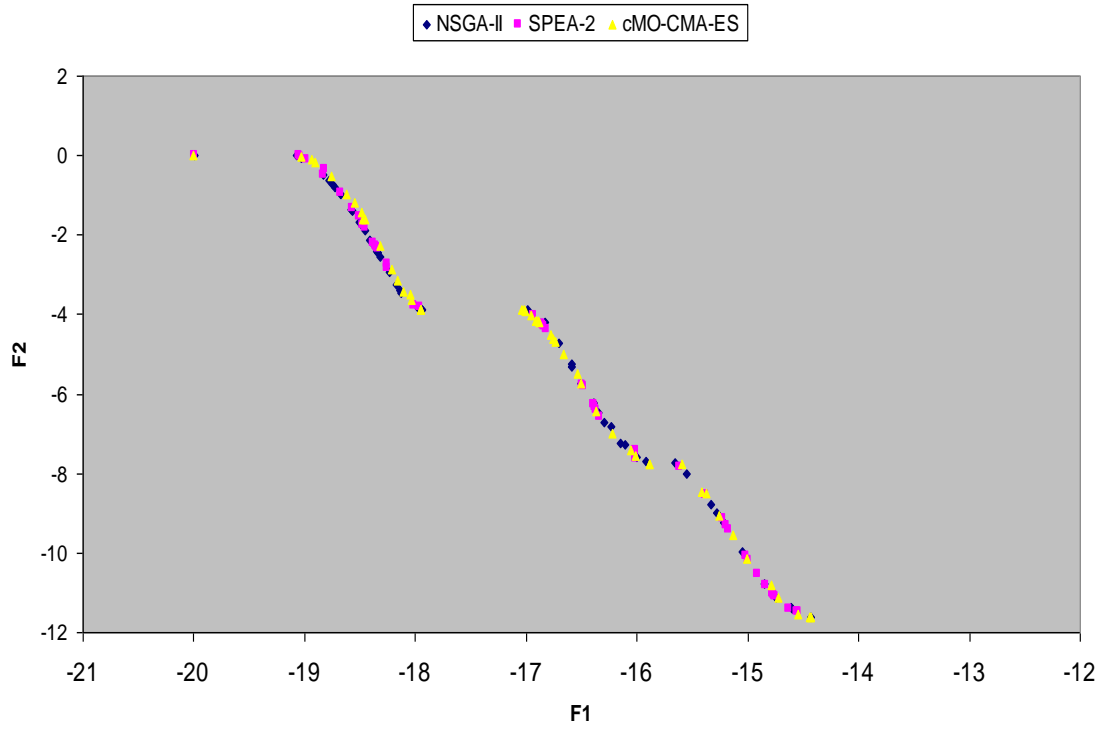


Σχήμα 6.8: Μέτωπα Pareto.



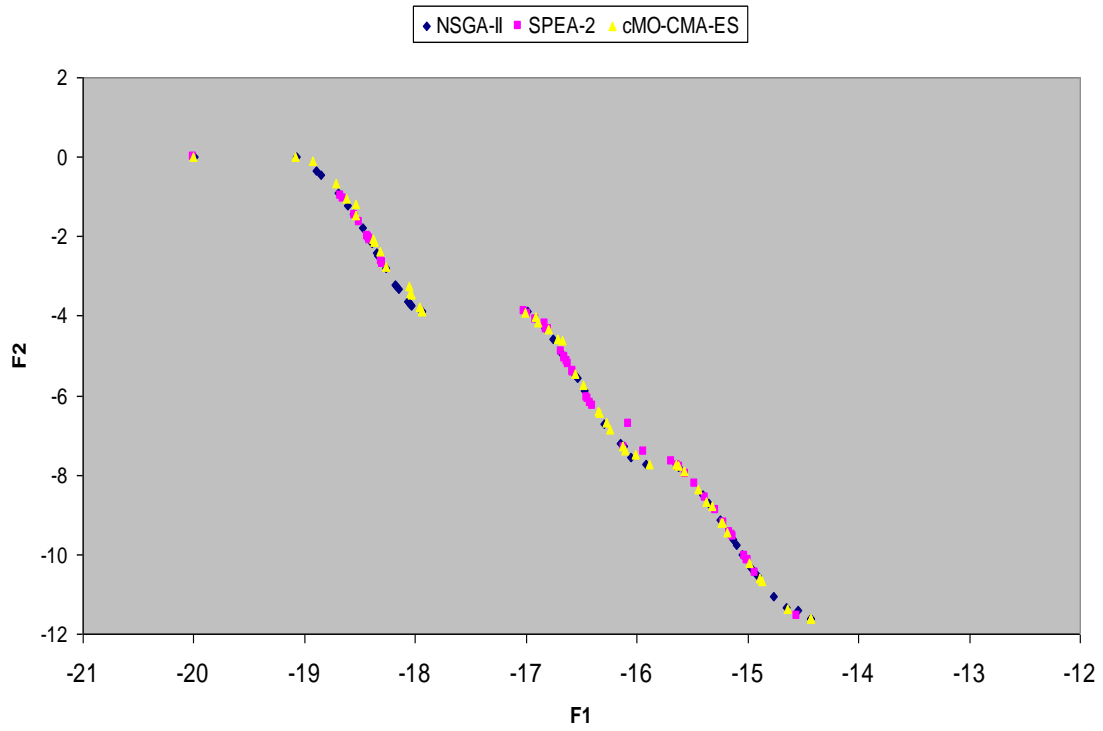
Σχήμα 6.9: Μέτωπα Pareto.

Problem KUR. Pareto front after 200 generations



Σχήμα 6.10: Μέτωπα Pareto.

Problem KUR. Pareto front after 500 generations



Σχήμα 6.11: Μέτωπα Pareto.

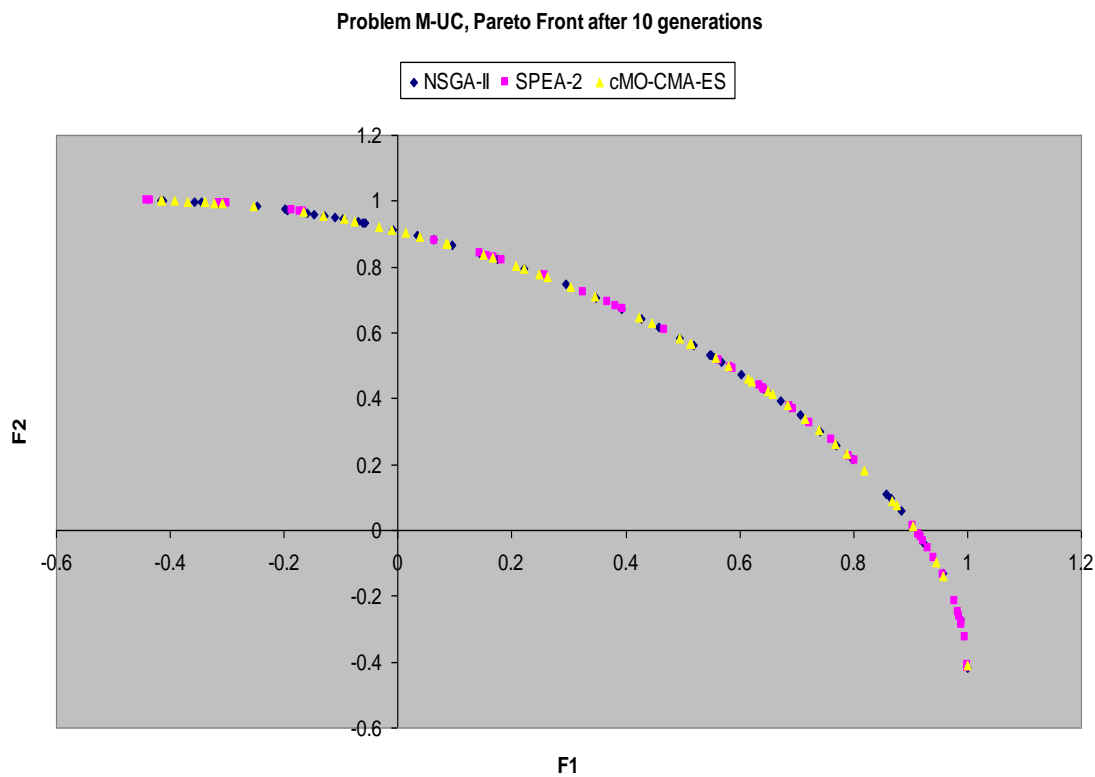
6.2.1.2 Πρόβλημα M-UC

Το πρόβλημα βελτιστοποίησης M-UC χαρακτηρίζεται από δύο αντικειμενικές συναρτήσεις και δύο μεταβλητές σχεδιασμού. Διατυπώνεται από τη σχέση:

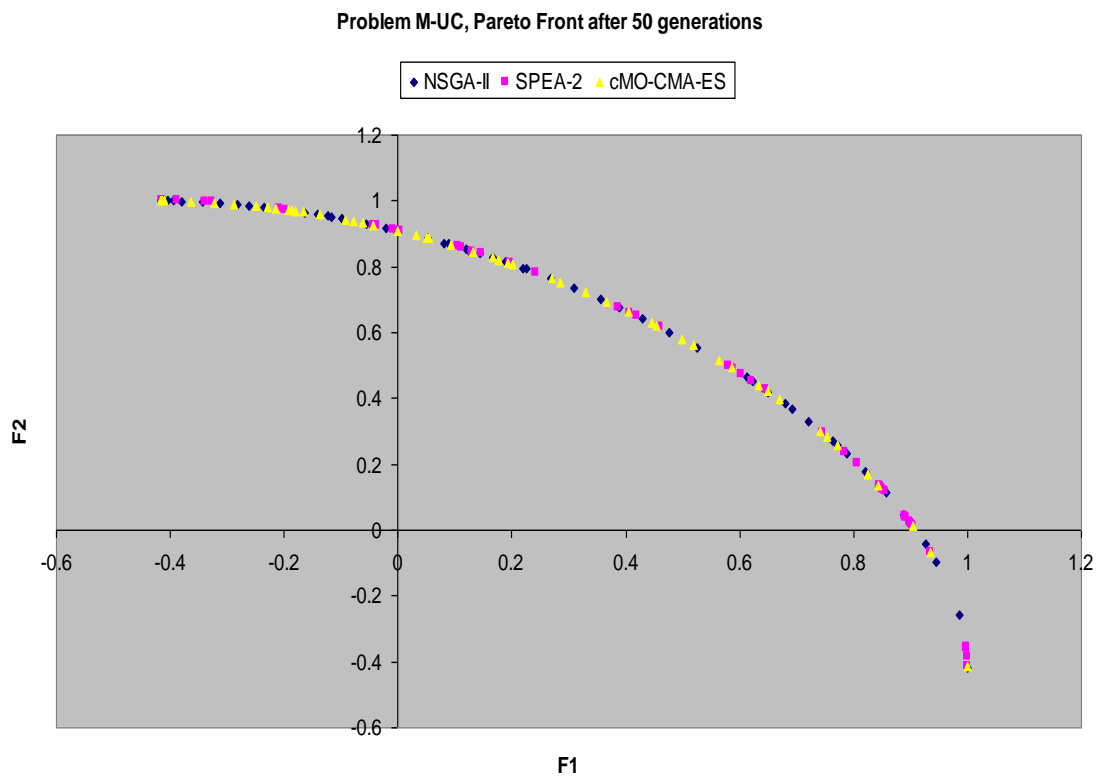
$$\begin{aligned} \max \begin{cases} f_1(x_1, x_2) = \sin(x_1^2 + x_2^2) - 1 \\ f_2(x_1, x_2) = \sin(x_1^2 + x_2^2) + 1 \end{cases} \\ \text{subject to: } 0 \leq x_i \leq \frac{3\pi}{4}, i = 1, 2 \end{aligned} \quad (6.7)$$

Το Pareto-βέλτιστο σύνολο σχεδιασμών του προβλήματος M-UC, δεν παρουσιάζει ασυνέχεια όπως αυτό του προηγούμενου προβλήματος. Για την εύρεση του Pareto-βέλτιστου συνόλου σχεδιασμών, ακολουθήθηκε η διαδικασία του προηγούμενου προβλήματος.

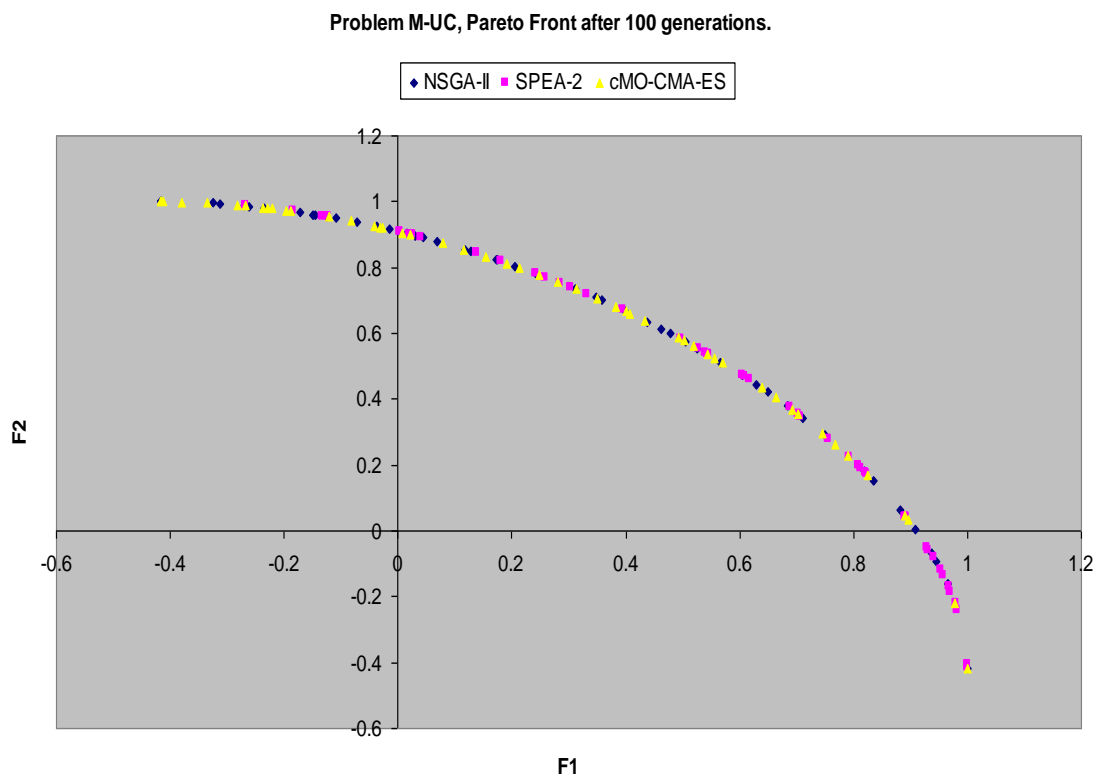
Στα σχήματα 6.12 ως 6.14 απεικονίζεται η σύγκλιση στο Pareto-βέλτιστο σύνολο των σχεδιασμών, των τριών αλγόριθμων, για 10, 50 και 100 γενιές εξέλιξης. Παρατηρούμε πως οι λύσεις έχουν συγκλίνει στο Pareto-βέλτιστο σύνολο λύσεων μέσα σε 10 γενιές εξέλιξης.



Σχήμα 6.12: Μέτωπα Pareto.



Σχήμα 6.13: Μέτωπα Pareto.



Σχήμα 6.14: Μέτωπα Pareto.

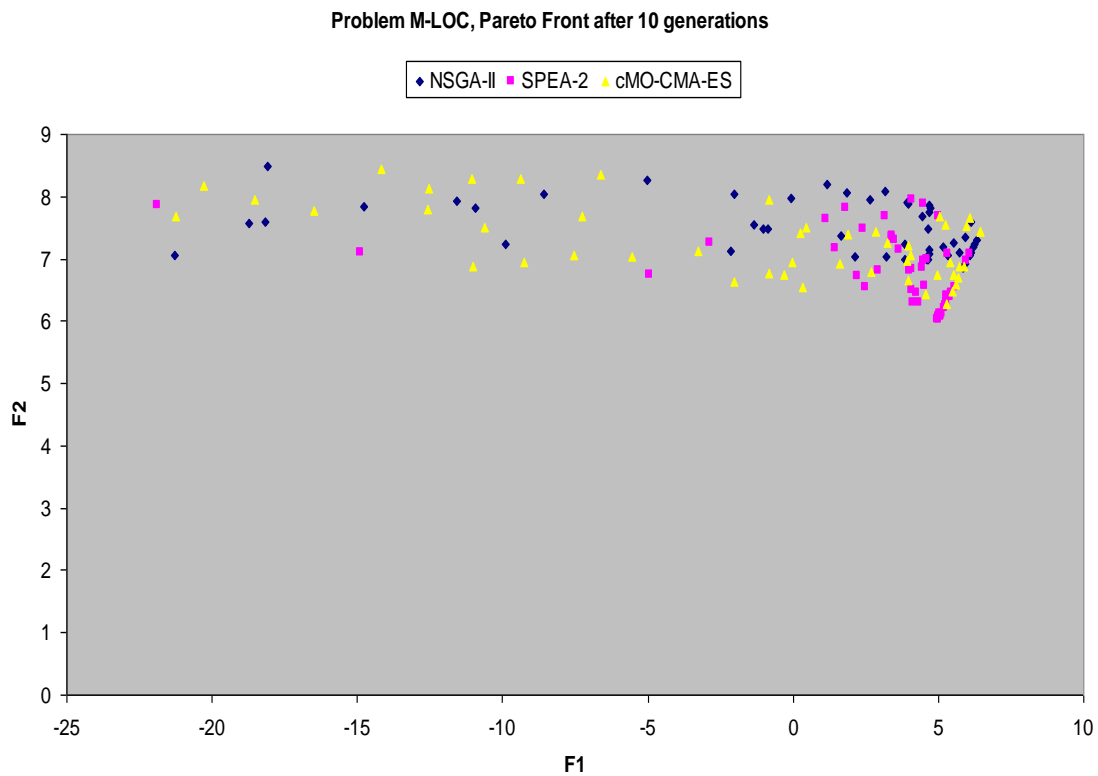
6.2.1.3 Πρόβλημα M-LOC

Το πρόβλημα βελτιστοποίησης M-LOC διατυπώνεται από τη σχέση:

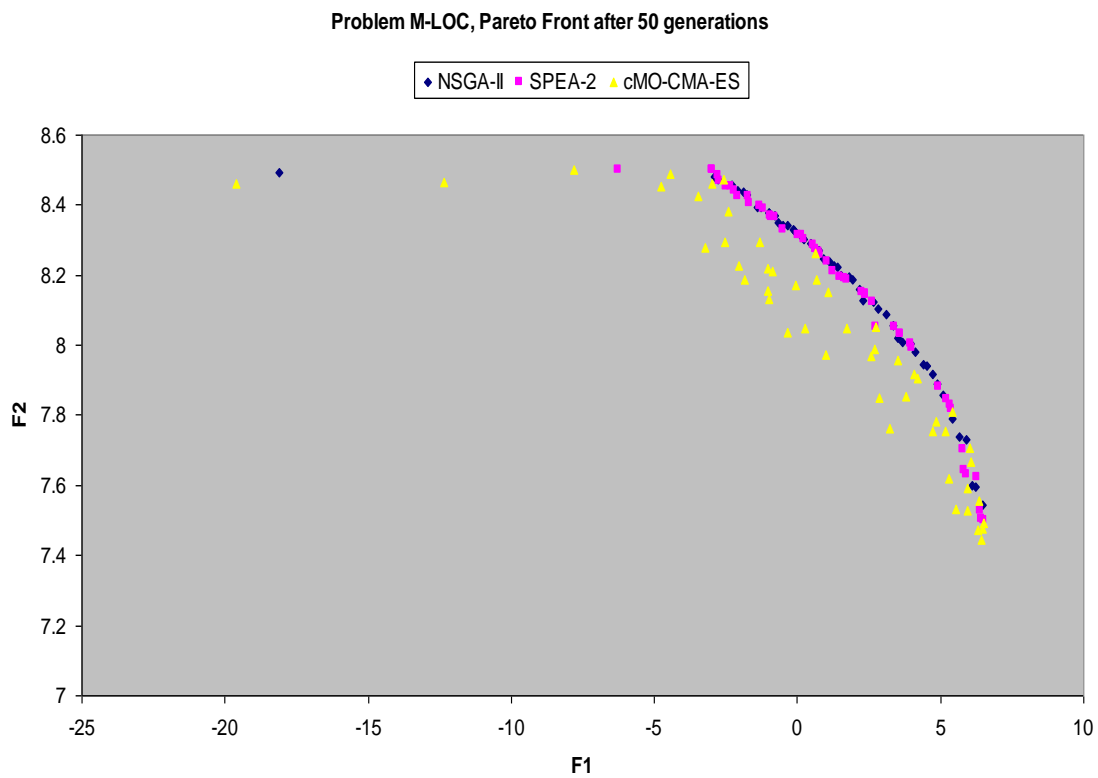
$$\begin{aligned}
& \max \begin{cases} f_1 & x_1, x_2 = -x_1^2 + x_2 \\ f_2 & x_1, x_2 = \frac{x_1}{2} + x_2 + 1 \end{cases} \\
& \text{subject to: } g_1(\mathbf{x}) = \frac{x_1}{6} + x_2 - \frac{13}{2} \leq 0 \\
& \quad g_2(\mathbf{x}) = \frac{x_1}{2} + x_2 - \frac{15}{2} \leq 0 \\
& \quad g_3(\mathbf{x}) = 5x_1 + x_2 - 30 \leq 0 \\
& \quad 0 \leq x_i \leq 100, i = 1, 2
\end{aligned} \tag{6.8}$$

Το πρόβλημα M-LOC χαρακτηρίζεται από δύο μεταβλητές σχεδιασμού, δύο αντικειμενικές συναρτήσεις και τρεις συναρτήσεις ανισοτικών περιορισμών. Για την εύρεση του Pareto-βέλτιστου συνόλου σχεδιασμών, ακολουθήθηκε η διαδικασία του προηγούμενου προβλήματος.

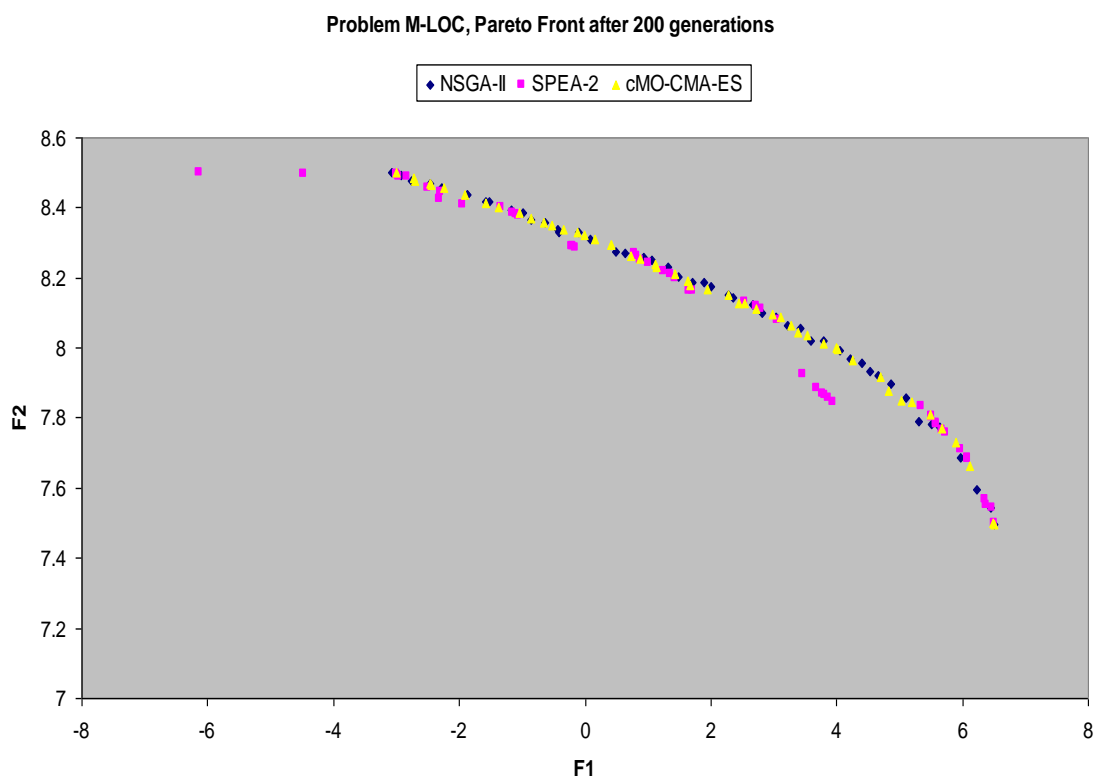
Στα σχήματα 6.15 ως 6.19 απεικονίζεται η σύγκλιση στο Pareto-βέλτιστο σύνολο των σχεδιασμών, των τριών αλγόριθμων, για 10, 50, 100, 200 και 500 γενιές εξέλιξης.



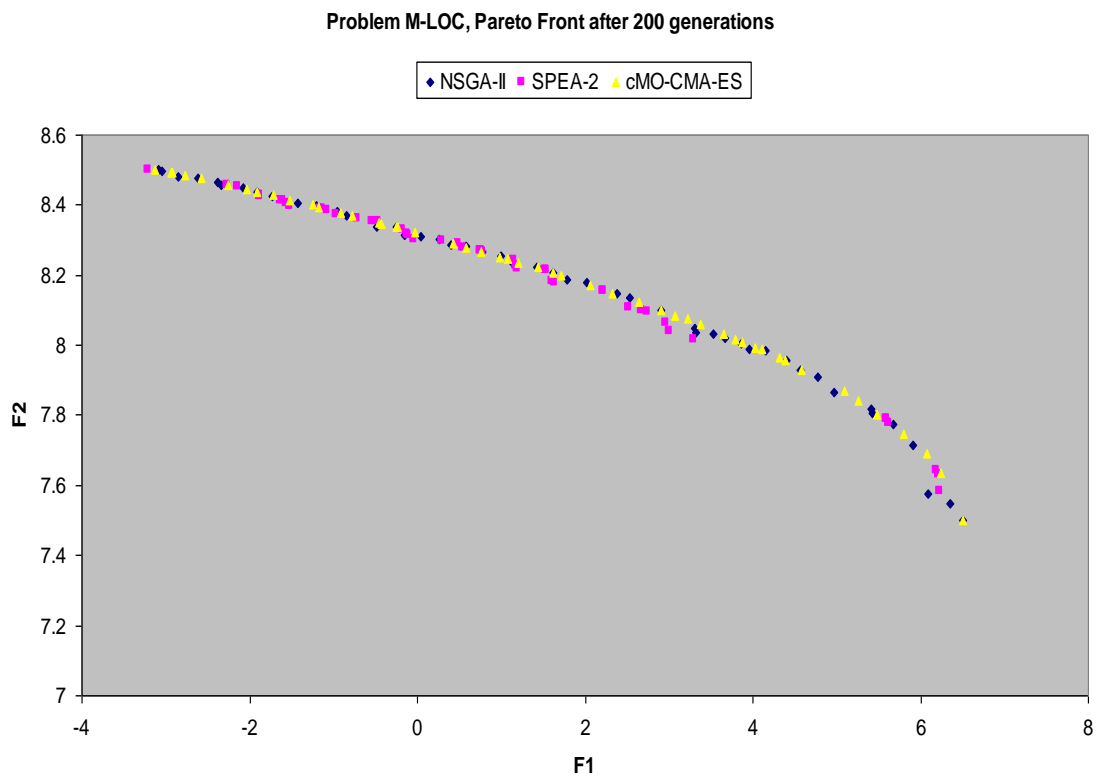
Σχήμα 6.15: Μέτωπα Pareto.



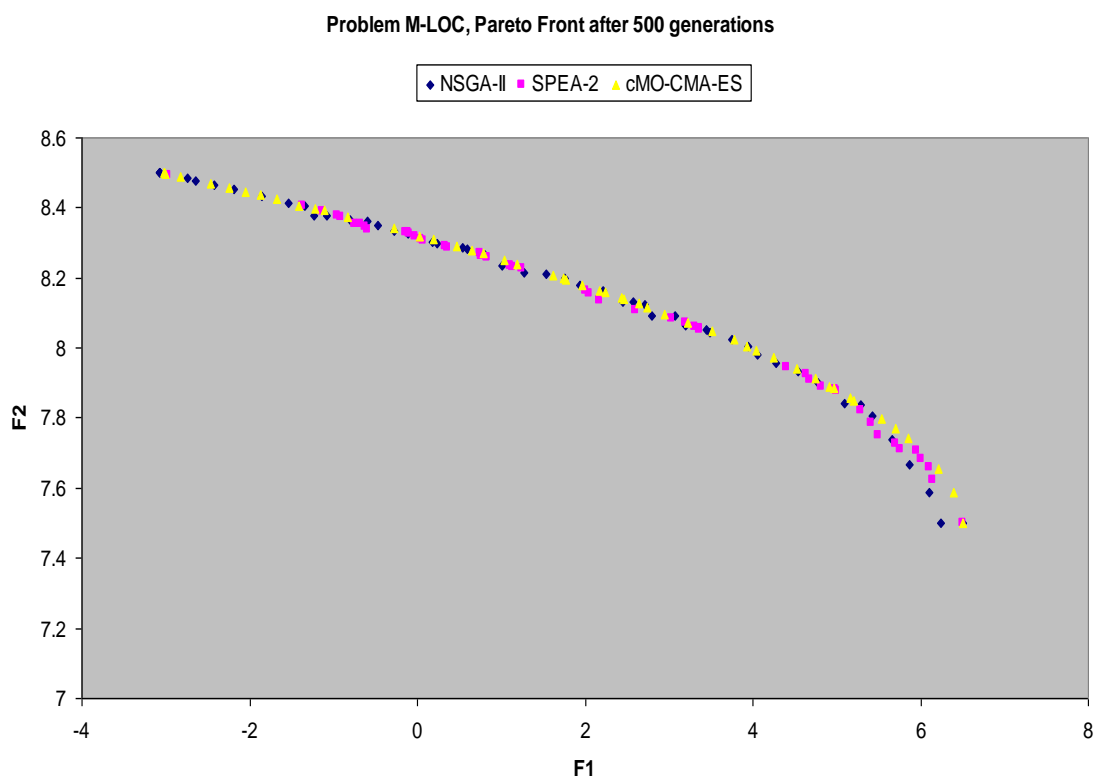
Σχήμα 6.16: Μέτωπα Pareto.



Σχήμα 6.17: Μέτωπα Pareto.



Σχήμα 6.18: Μέτωπα Pareto.



Σχήμα 6.19: Μέτωπα Pareto.

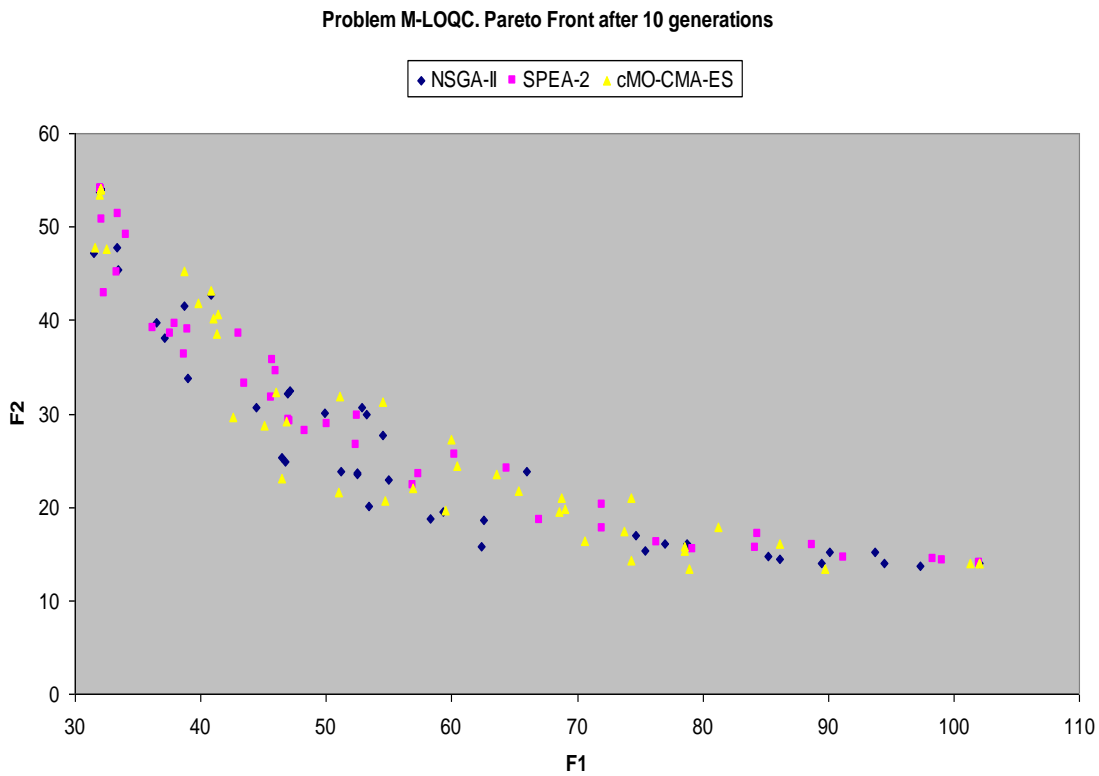
6.2.1.4 Πρόβλημα M-LOQC

Το πρόβλημα βελτιστοποίησης M-LOQC διατυπώνεται από τη σχέση:

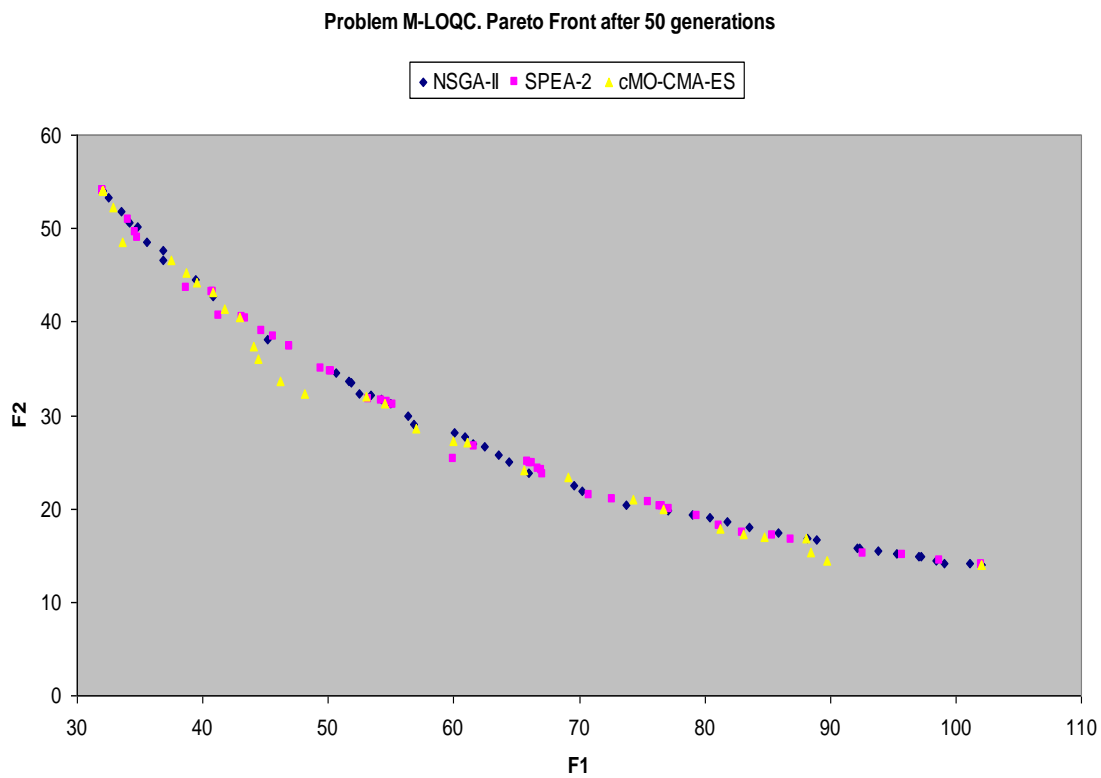
$$\begin{aligned} \max \quad & \begin{cases} f_1 \quad x_1, x_2 = x_1 + x_2^2 \\ f_2 \quad x_1, x_2 = x_1^2 + x_2 \end{cases} \\ \text{subject to:} \quad & g_1(\mathbf{x}) = 12 - x_1 - x_2 \geq 0 \\ & g_2(\mathbf{x}) = x_1^2 + 10x_1 - x_2^2 + 16x_2 - 80 \geq 0 \\ & 2 \leq x_1 \leq 5, \quad 7 \leq x_2 \leq 10 \end{aligned} \quad (6.9)$$

Το πρόβλημα M-LOQC χαρακτηρίζεται από δύο μεταβλητές σχεδιασμού, δύο αντικειμενικές συναρτήσεις και δύο συναρτήσεις ανισοτικών περιορισμών. Για την εύρεση του Pareto-βέλτιστου συνόλου σχεδιασμών, ακολουθήθηκε η διαδικασία του προηγούμενου προβλήματος.

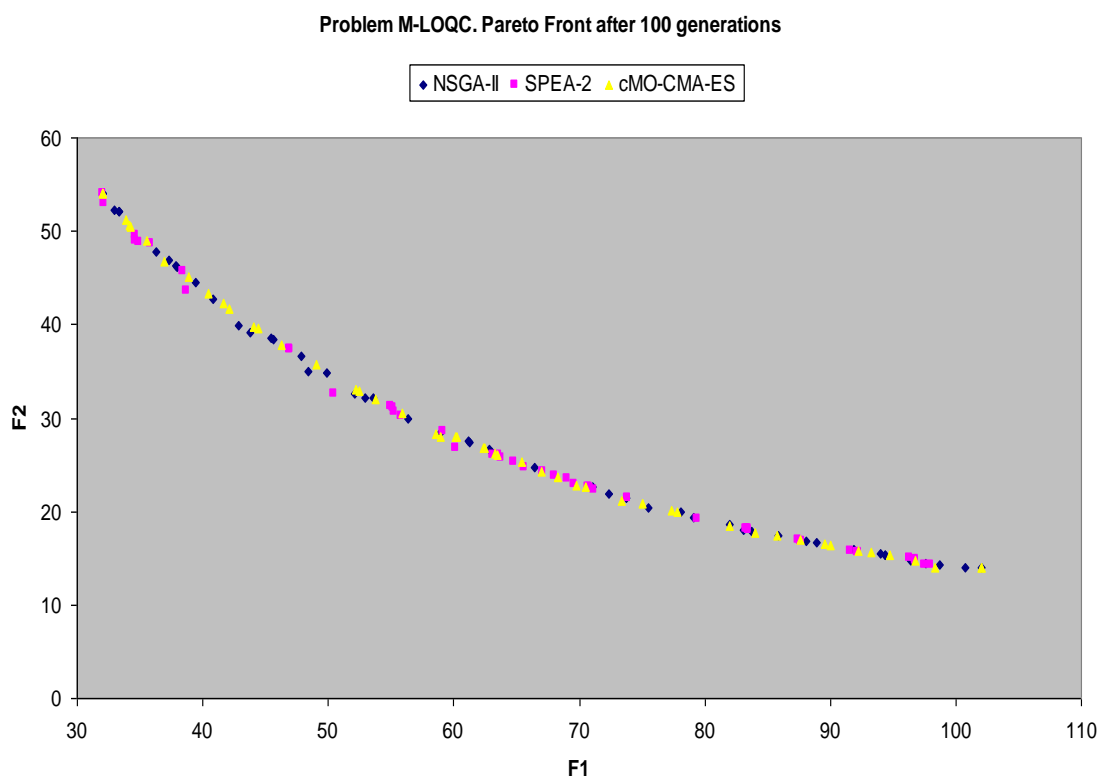
Στα σχήματα 6.20 ως ?? απεικονίζεται η σύγκλιση στο Pareto-βέλτιστο σύνολο των σχεδιασμών, των τριών αλγόριθμων, για 10, 50, 100, 200 και 500 γενιές εξέλιξης.



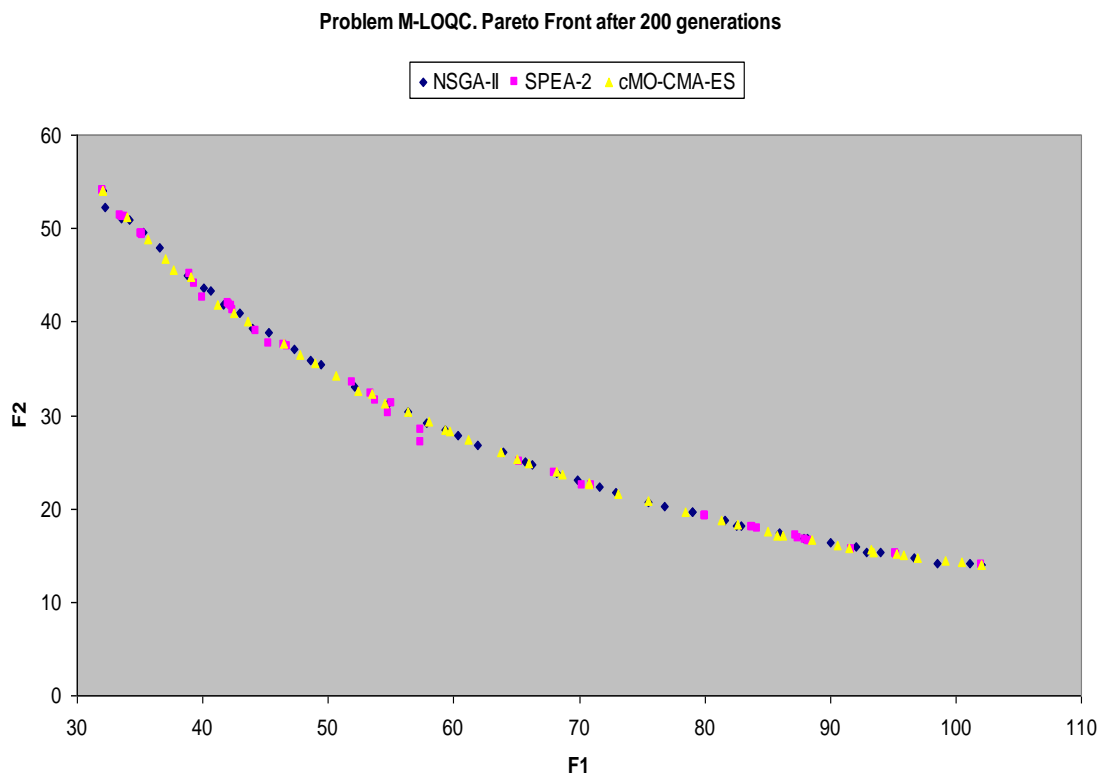
Σχήμα 6.20: Μέτωπα Pareto.



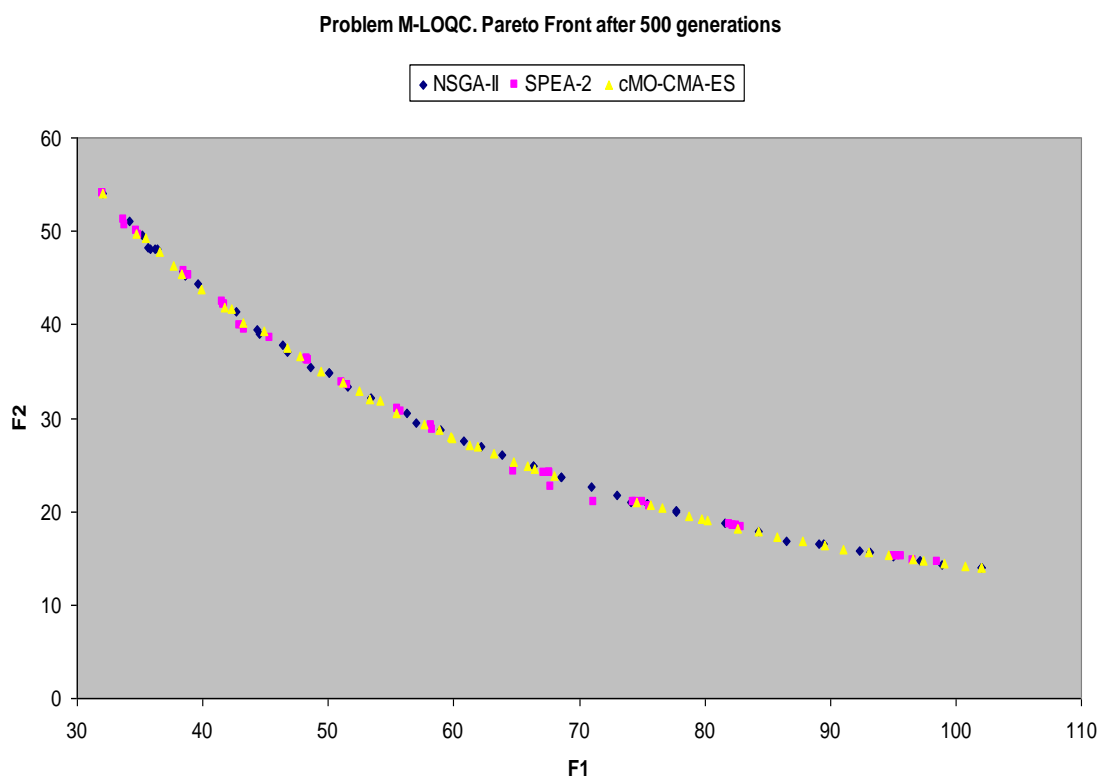
Σχήμα 6.21: Μέτωπα Pareto.



Σχήμα 6.22: Μέτωπα Pareto.



Σχήμα 6.23: Μέτωπα Pareto.



Σχήμα 6.24: Μέτωπα Pareto.

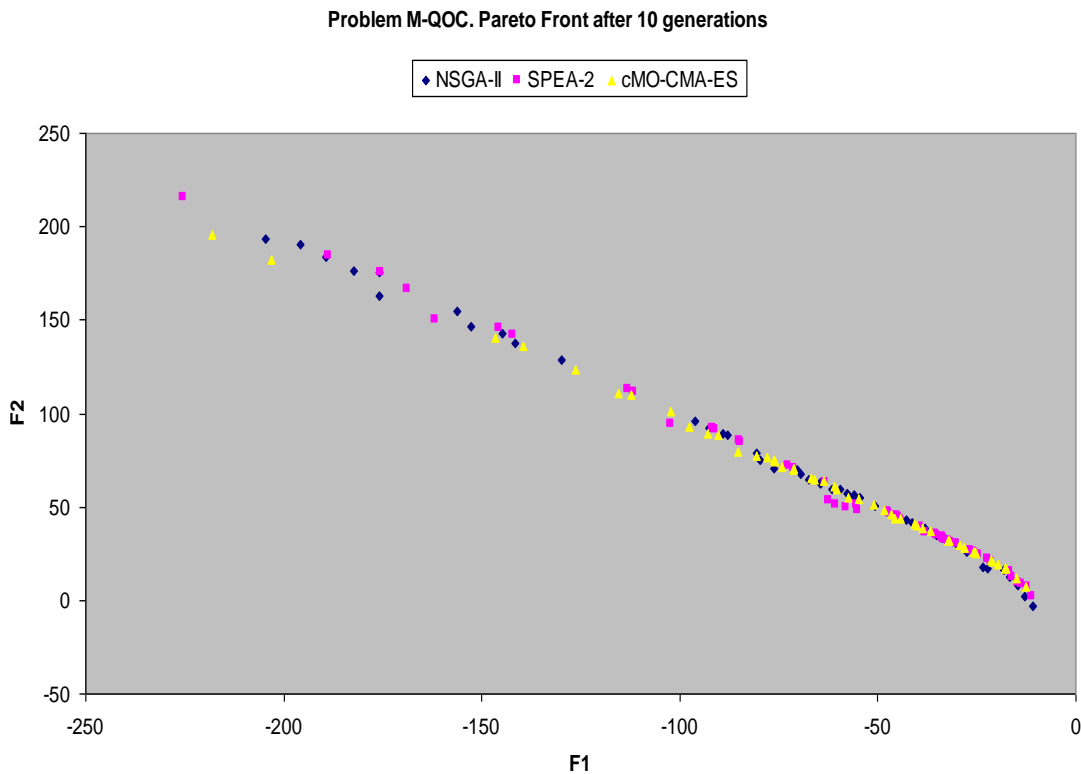
6.2.1.5 Πρόβλημα M-QOC

Το πρόβλημα βελτιστοποίησης M-QOC προτάθηκε από τους Srinivas και Deb, και διατυπώνεται από τη σχέση:

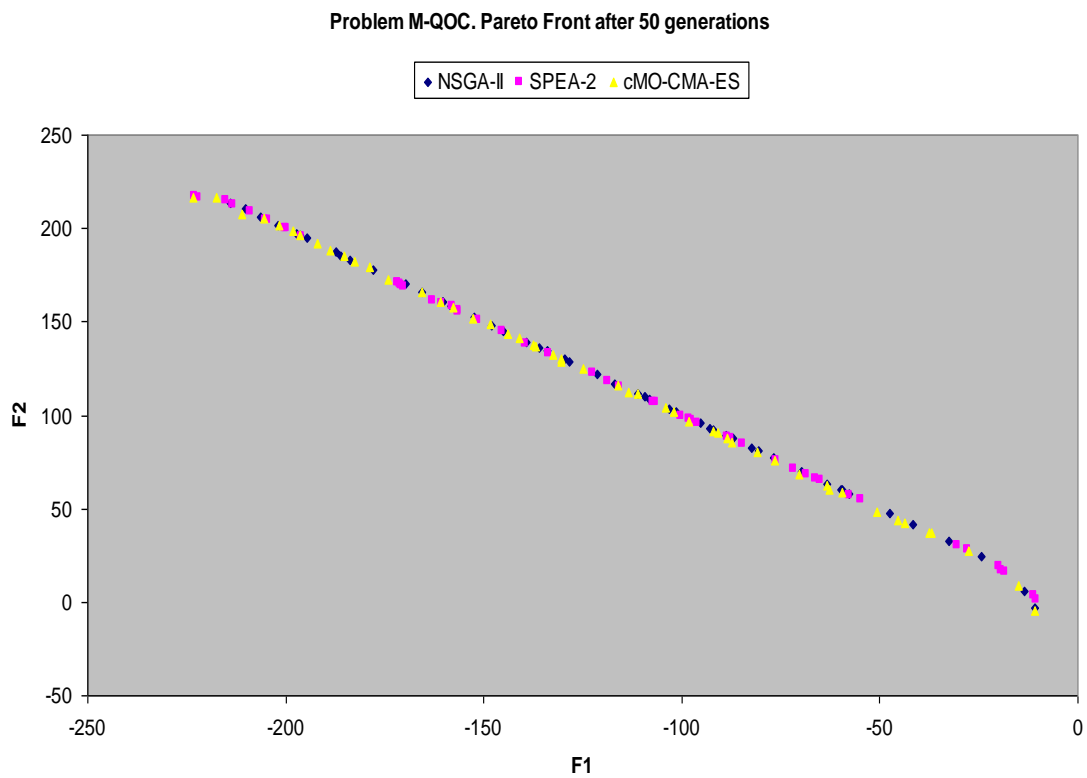
$$\begin{aligned}
& \max \begin{cases} f_1(x_1, x_2) = -[2 + x_1 - 2^2 + x_2 - 1^2] \\ f_2(x_1, x_2) = -[9x_1 - x_2 - 1^2] \end{cases} \\
& \text{subject to: } g_1(x) = x_1^2 + x_2^2 - 225 \leq 0 \\
& \quad g_2(x) = x_1 - 3x_2 + 10 \leq 0 \\
& \quad -20 \leq x_i \leq 20, i = 1, 2
\end{aligned} \tag{6.10}$$

Το πρόβλημα M-QOC χαρακτηρίζεται από δύο μεταβλητές σχεδιασμού, δύο αντικειμενικές συναρτήσεις και δύο συναρτήσεις ανισοτικών περιορισμών. Για την εύρεση του Pareto-βέλτιστου συνόλου σχεδιασμών, ακολουθήθηκε η διαδικασία του προηγούμενου προβλήματος.

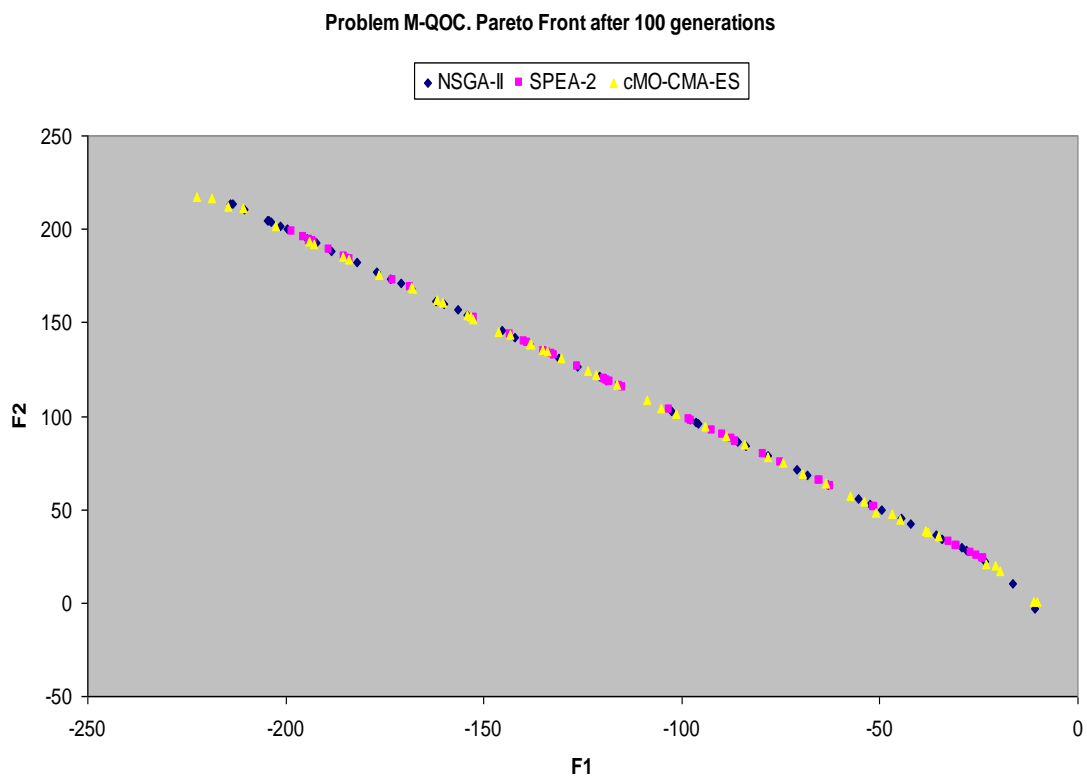
Στα σχήματα 6.25 ως 6.29 απεικονίζεται η σύγκλιση στο Pareto-βέλτιστο σύνολο των σχεδιασμών, των τριών αλγόριθμων, για 10, 50, 100, 200 και 500 γενιές εξέλιξης.



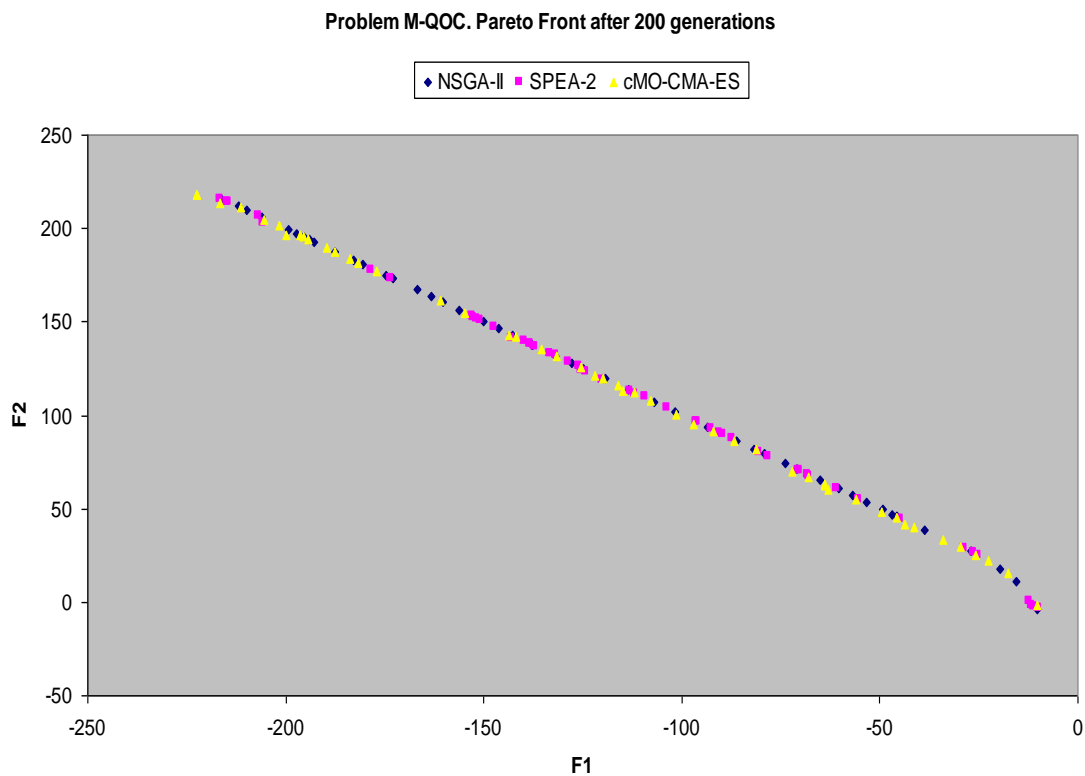
Σχήμα 6.25: Μέτωπα Pareto.



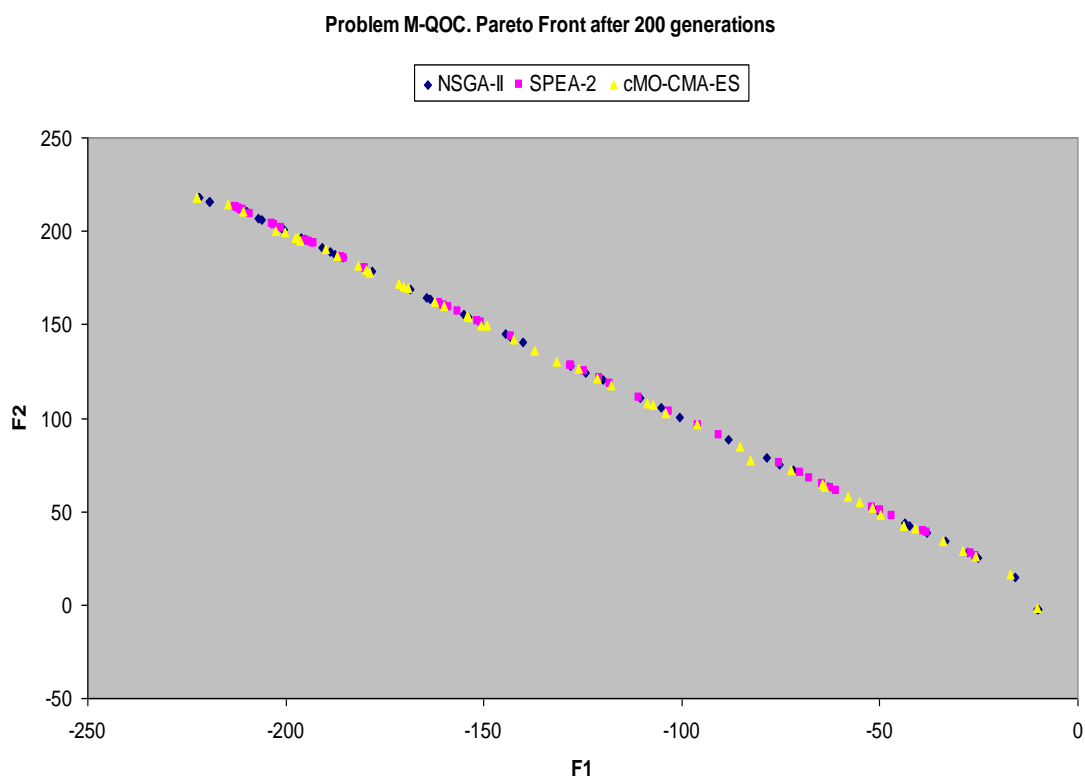
Σχήμα 6.26: Μέτωπα Pareto.



Σχήμα 6.27: Μέτωπα Pareto.



Σχήμα 6.28: Μέτωπα Pareto.



Σχήμα 6.29: Μέτωπα Pareto.

6.2.1.6 Πρόβλημα M-3OU

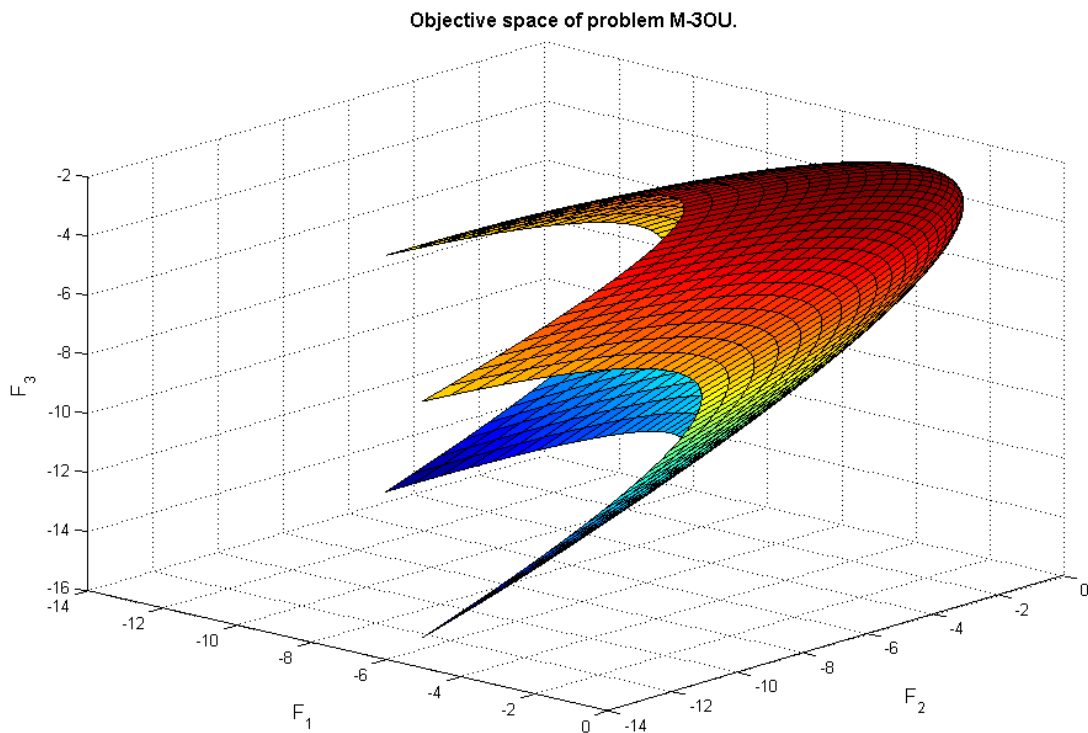
Το πρόβλημα βελτιστοποίησης M-3OU διατυπώνεται από τη σχέση:

$$\max \begin{cases} f_1(x_1, x_2) = x_1^2 + x_2 - 1^2 \\ f_2(x_1, x_2) = x_1^2 + x_2 + 1^2 + 1 \\ f_3(x_1, x_2) = x_1 - 1^2 + x_2^2 + 2 \end{cases} \quad (6.11)$$

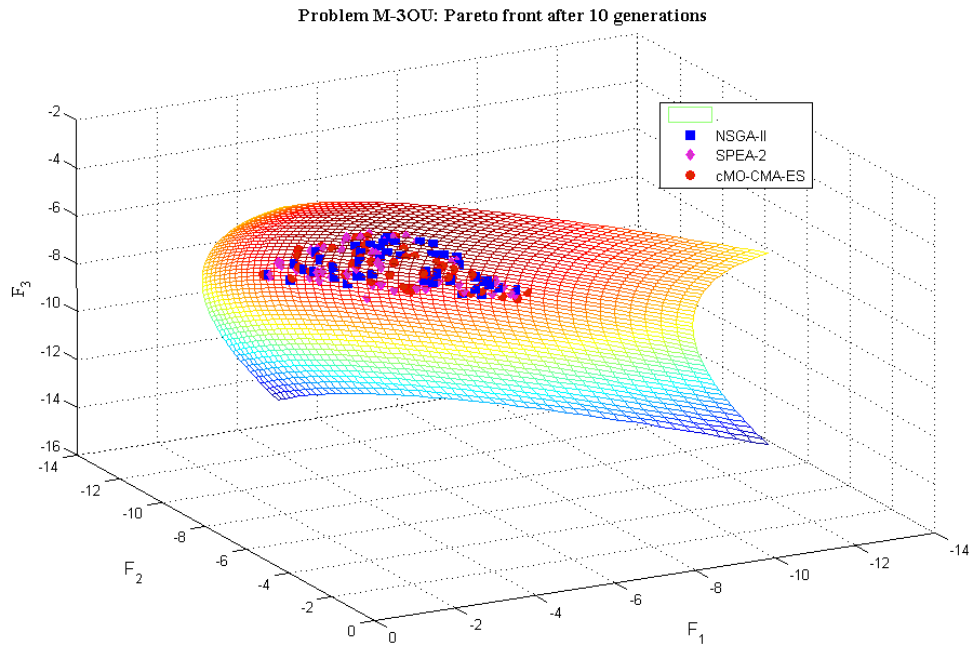
subject to: $-2 \leq x_i \leq 2, i = 1, 2$

Το πρόβλημα M-QOC χαρακτηρίζεται από δύο μεταβλητές σχεδιασμού και τρεις αντικειμενικές συναρτήσεις. Για την εύρεση του Pareto-βέλτιστου συνόλου σχεδιασμών, ακολουθήθηκε η διαδικασία του προηγούμενου προβλήματος.

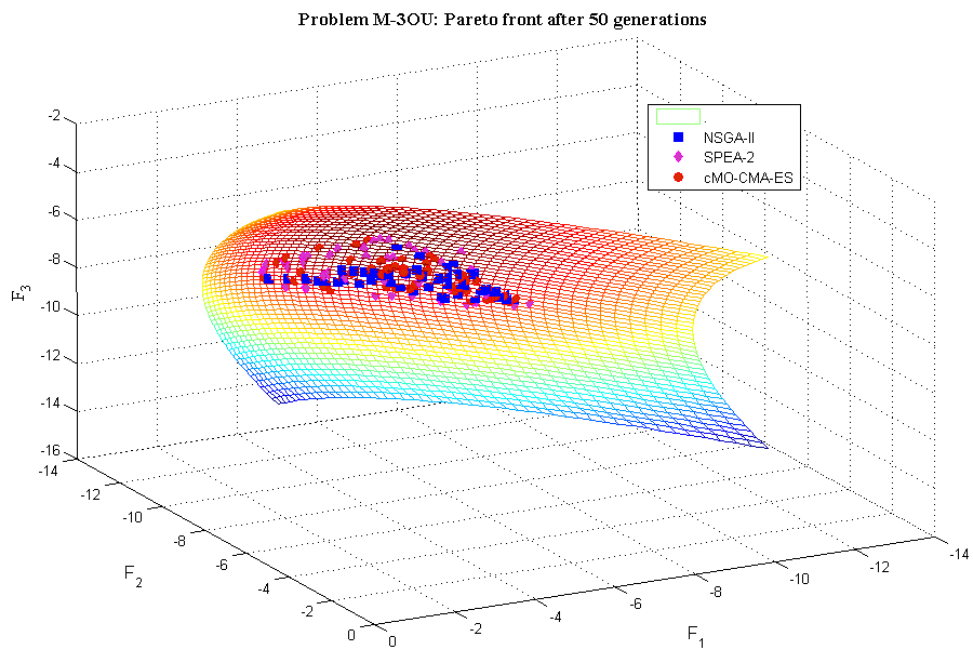
Στο σχήμα 6.30, απεικονίζεται η επιφάνεια εφικτών σχεδιασμών στο χώρο που ορίζεται από τις αντικειμενικές συναρτήσεις. Στα σχήματα 6.31 ως 6.35 απεικονίζεται η σύγκλιση στο Pareto-βέλτιστο σύνολο των σχεδιασμών, των τριών αλγόριθμων, για 10, 50, 100, 200 και 500 γενιές εξέλιξης.



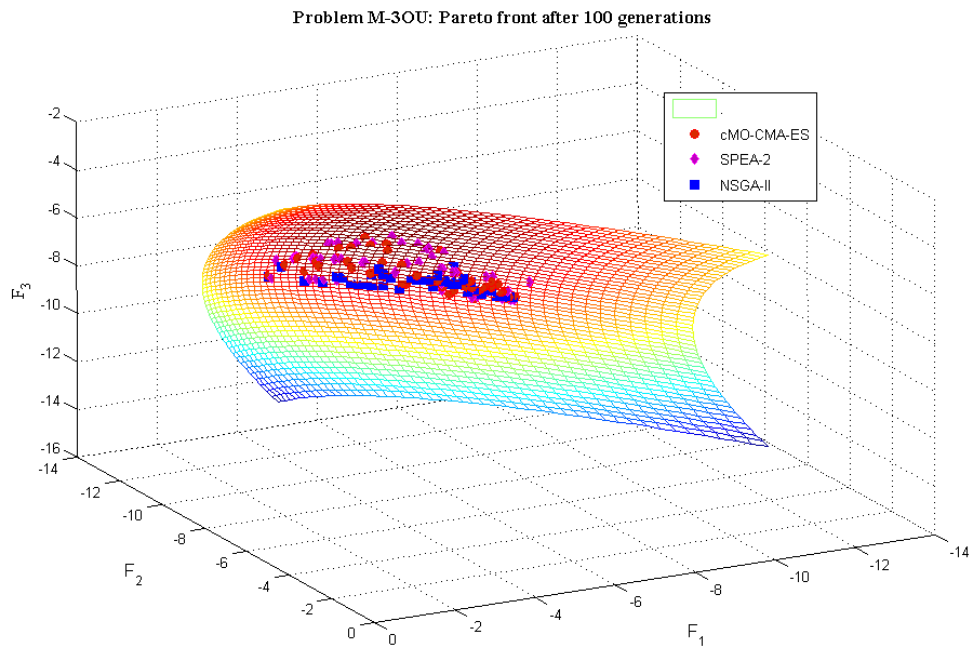
Σχήμα 6.30: Απεικόνιση της επιφάνειας των εφικτών σχεδιασμών στο χώρο των αντικειμενικών συναρτήσεων του προβλήματος M-30U.



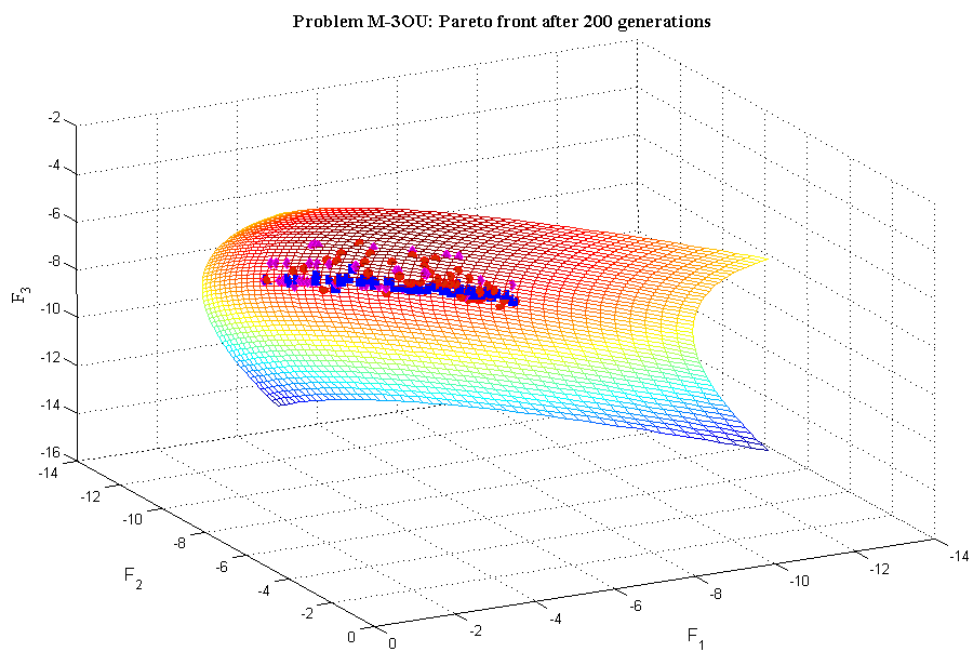
Σχήμα 6.31: Μέτωπα Pareto.



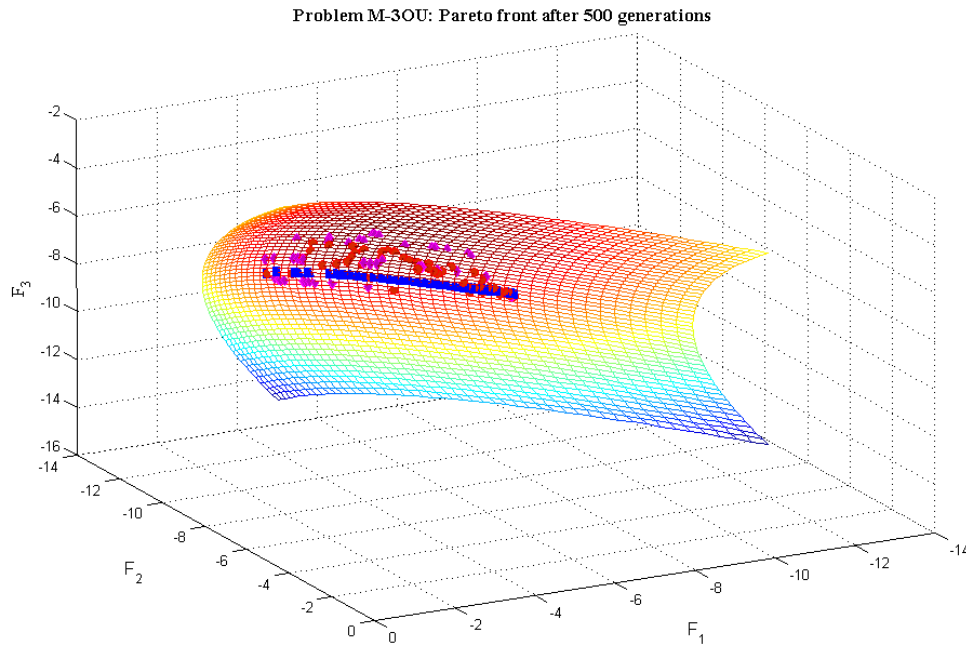
Σχήμα 6.32: Μέτωπα Pareto.



Σχήμα 6.33: Μέτωπα Pareto.



Σχήμα 6.34: Μέτωπα Pareto.



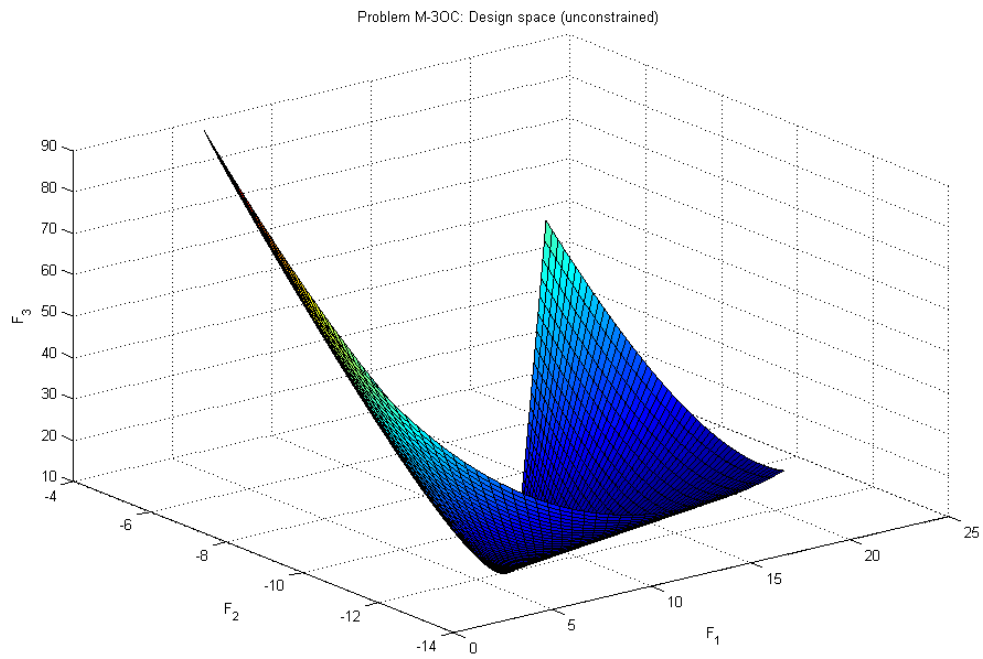
Σχήμα 6.35: Μέτωπα Pareto.

6.2.1.7 Πρόβλημα M-3OC

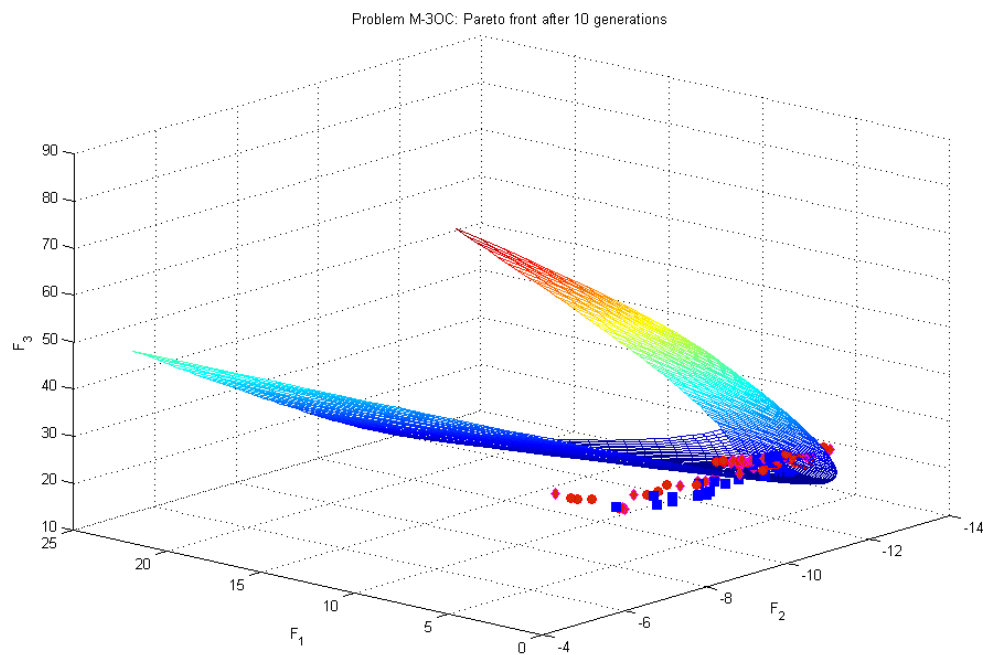
Το πρόβλημα βελτιστοποίησης M-3OU διατυπώνεται από τη σχέση:

$$\begin{aligned}
 \max \left\{ \begin{array}{l} f_1(x_1, x_2) = x_1 - 2^2/2 + x_2 + 1^2/13 + 3 \\ f_2(x_1, x_2) = x_1 + x_2 - 3^2/175 + 2x_2 - x_1^2/17 - 13 \\ f_3(x_1, x_2) = 3x_1 - 2x_2 + 4^2/8 + x_1 - x_2 + 1^2/27 + 15 \end{array} \right. \\
 \text{subject to: } g_1(\mathbf{x}) = -4x_1 - x_2 + 4 > 0 \\
 g_2(\mathbf{x}) = x_1 + 1 > 0 \\
 g_3(\mathbf{x}) = x_2 - x_1 + 2 > 0 \\
 -4 \leq x_i \leq 4, i = 1, 2
 \end{aligned} \tag{6.12}$$

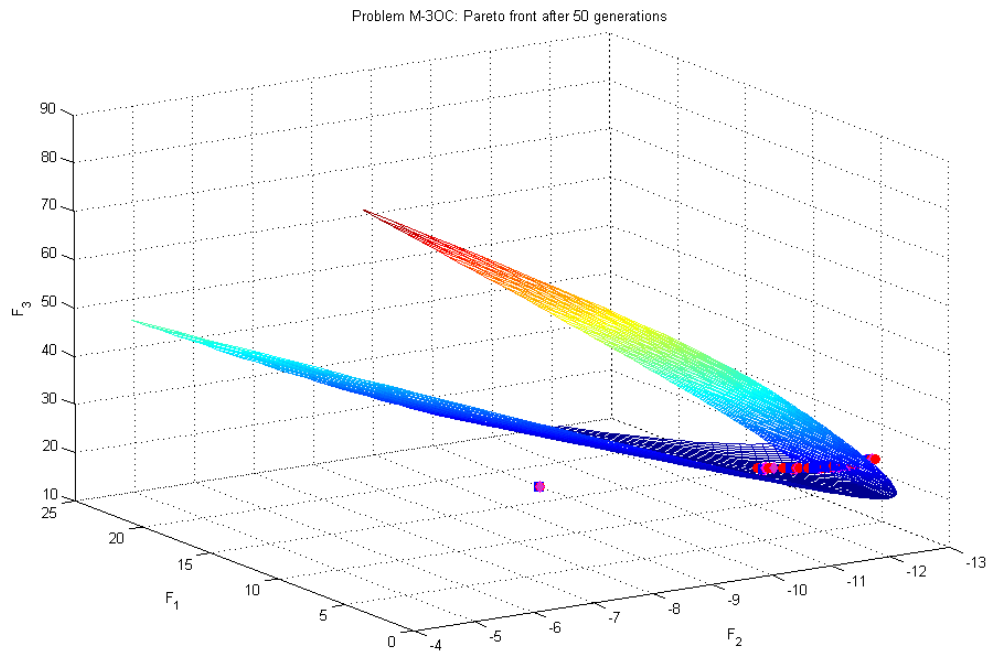
Το πρόβλημα M-3OC χαρακτηρίζεται από δύο μεταβλητές σχεδιασμού, τρεις αντικειμενικές συναρτήσεις και τρεις συναρτήσεις ανισοτικών περιορισμών. Για την εύρεση του Pareto-βέλτιστου συνόλου σχεδιασμών, ακολουθήθηκε η διαδικασία του προηγούμενου προβλήματος. Στο σχήμα 6.36, απεικονίζεται η επιφάνεια σχεδιασμών στο χώρο που ορίζεται από τις αντικειμενικές συναρτήσεις. Στα σχήματα 6.37 ως 6.41 απεικονίζεται η σύγκλιση στο Pareto-βέλτιστο σύνολο των σχεδιασμών, των τριών αλγόριθμων, για 10, 50, 100, 200 και 500 γενιές εξέλιξης.



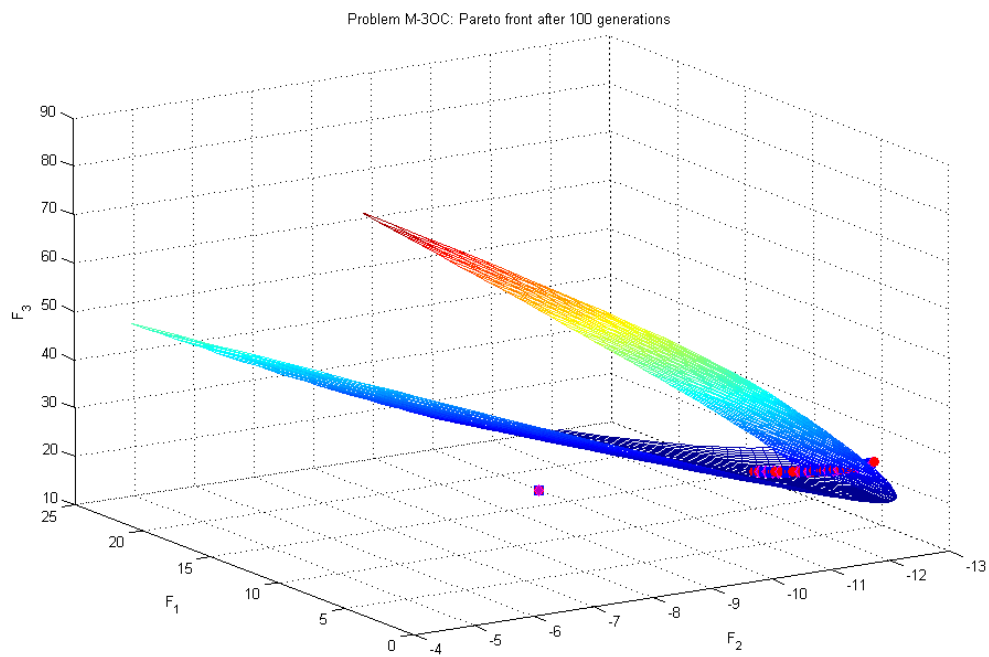
Σχήμα 6.36: Επιφάνεια σχεδιασμών.



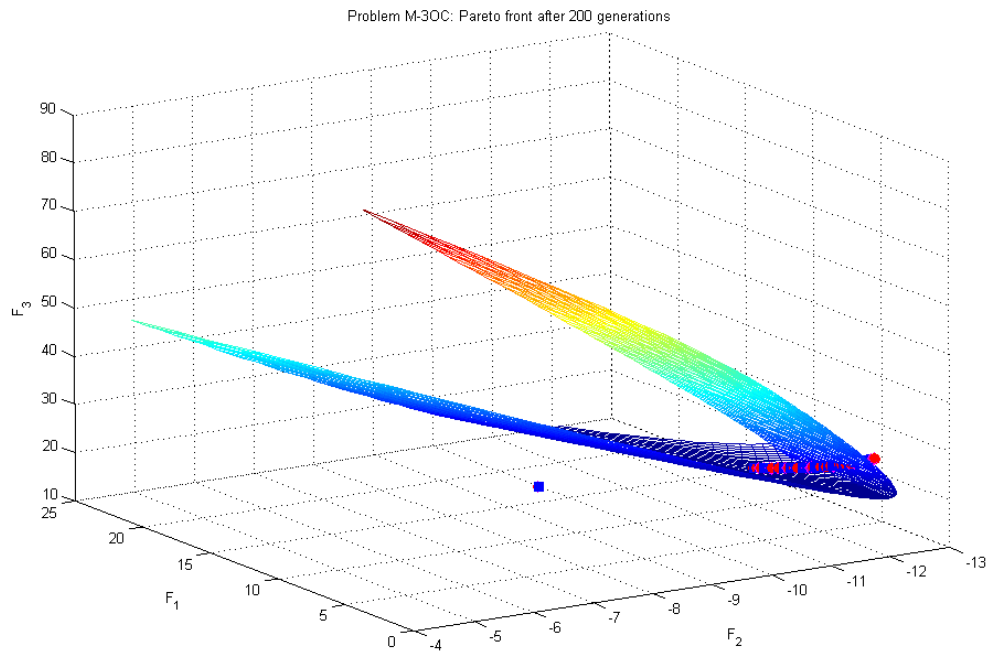
Σχήμα 6.37: Μέτωπα Pareto.



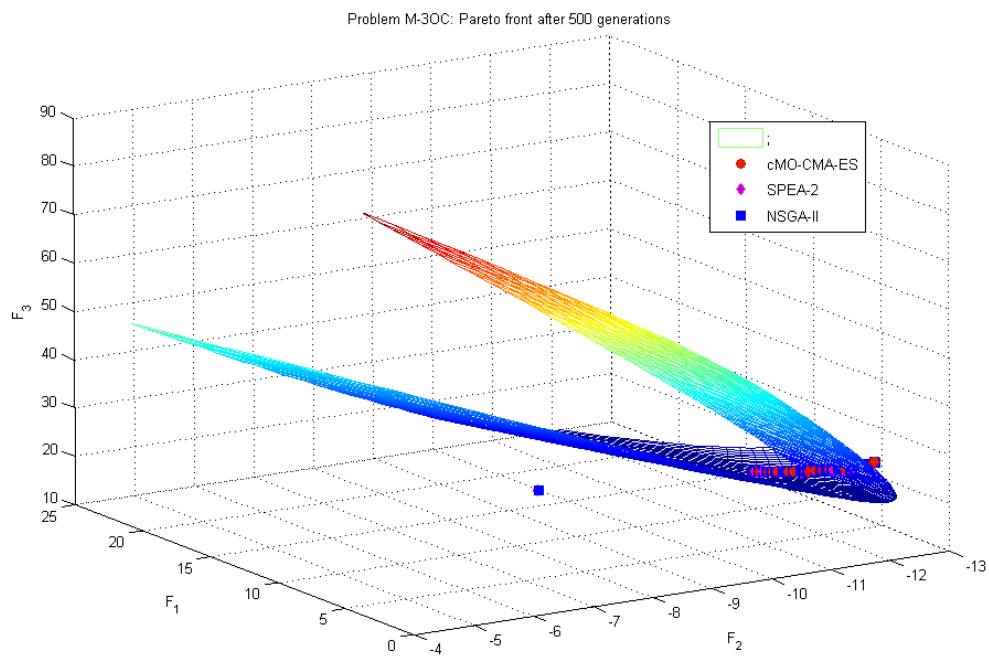
Σχήμα 6.38: Μέτωπα Pareto.



Σχήμα 6.39: Μέτωπα Pareto.



Σχήμα 6.40: Μέτωπα Pareto.



Σχήμα 6.41: Μέτωπα Pareto.

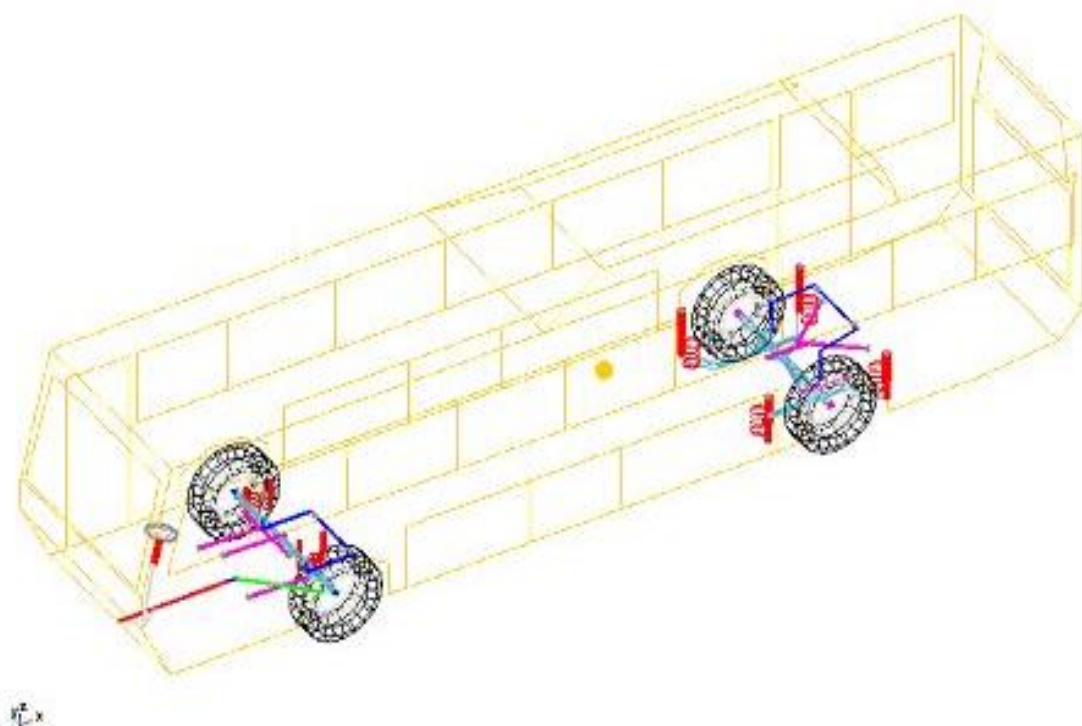
7 ΕΦΑΡΜΟΓΕΣ ΣΕ ΠΡΟΒΛΗΜΑΤΑ ΚΑΤΑΣΚΕΥΩΝ

7.1 Εισαγωγή

Το κεφάλαιο αυτό αναφέρεται στην εφαρμογή της μεθόδου των στρατηγικών εξέλιξης, που αναπτύχθηκε σε προηγούμενο κεφάλαιο, για τη βελτιστοποίηση τριών κατασκευών. Οι κατασκευές αυτές είναι: ένα λεωφορείο, ένας μεταλλικός πύργος και ένα μεταλλικό στέγαστρο. Για κάθε εφαρμογή επιλέχθηκαν τυχαία 6-6, 6-12 και 12-12 γονείς και απόγονοι αντίστοιχα. Η μέθοδος διαχείρισης περιορισμών που χρησιμοποιήθηκε είναι η θανατική καταδίκη. Τελικά, για κάθε περίπτωση εφαρμογής, προέκυψε το ελάχιστο βάρος κάθε κατασκευής, μέσω μιας επαναληπτικής διαδικασίας, η πορεία της οποίας φαίνεται σε ακόλουθα διαγράμματα

7.2 Εφαρμογή 1 – Λεωφορείο (bus)

Η εφαρμογή αυτή αναφέρεται σε ένα λεωφορείο μεταφοράς προσωπικού DAIMLER/ΕΛ.Β.Ο. 829.98.



Εικόνα 7.1.Λεωφορείο Μεταφοράς Προσωπικού 829.98

Αρχικά έγινε προσομοίωση του λεωφορείου με μεταλλικές ράβδους ορθογωνικής διατομής (beam elements), πακτωμένες μεταξύ τους. Το πλήθος των μεταβλητών σχεδιασμού είναι 15. Το πλήθος των πεπερασμένων στοιχείων είναι 1269 ενώ οι βαθμοί ελευθερίας είναι 4512 και ο αριθμός των κόμβων είναι 753. Η βάση δεδομένων των διατομών των στοιχείων είναι η ακόλουθη:

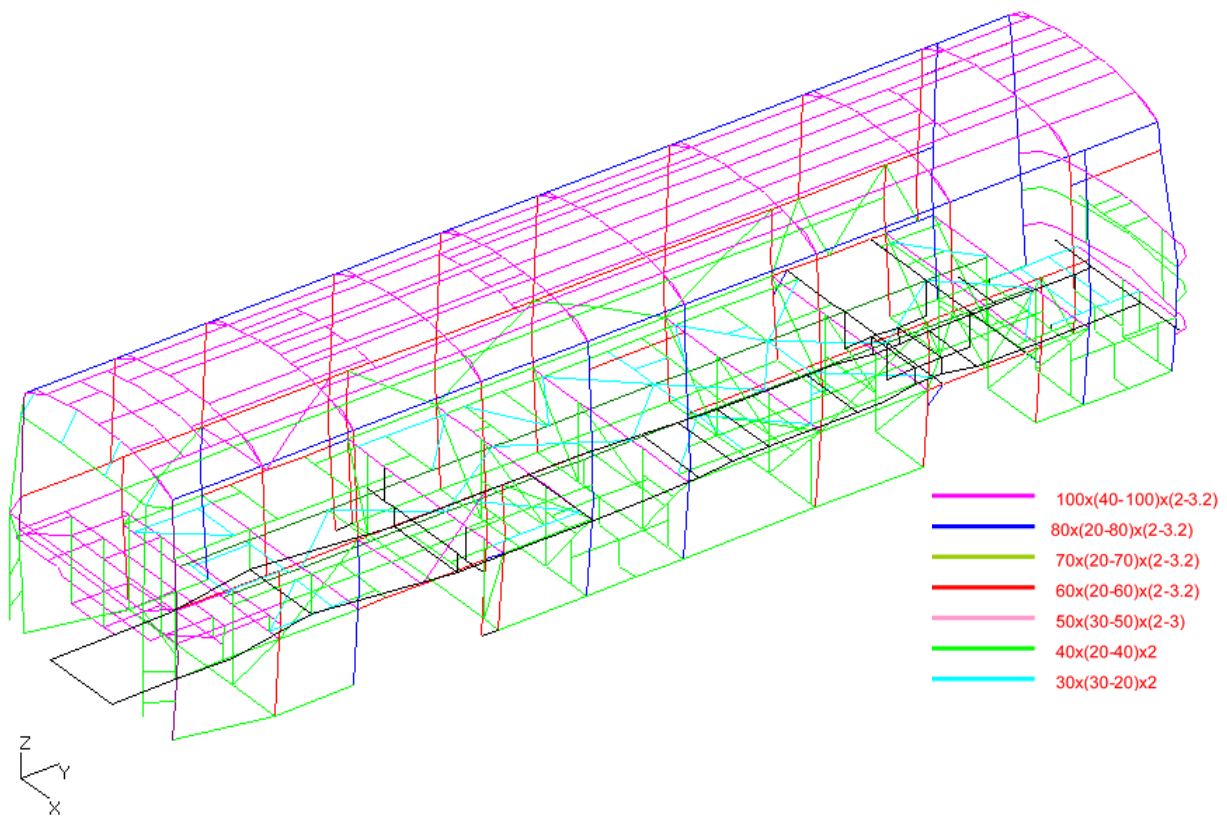
Για τον σχεδιασμό λαμβάνεται υπόψη η στατική φόρτιση (θέση οδηγού, κινητήρας, κλιματιστικό μηχανήμα, μηχανολογικός εξοπλισμός κατά μήκος της κεντρικής δοκού, παράθυρα) και το φρενάρισμα (φόρτιση κατά τον άξονα y), δηλαδή μάζα x αρνητική επιτάχυνση (η αρνητική επιτάχυνση είναι ίση με $1g$). Επίσης, πρέπει να ληφθεί υπόψη και η αντισυμμετρική κατακόρυφη φόρτιση (φόρτιση κατά τον άξονα z), δηλαδή μάζα x επιτάχυνση (η επιτάχυνση είναι ίση με $1,47g$). Για την περίπτωση αυτής της φόρτισης, το όχημα στηρίζεται στους αντιδιαμετρικά αντίθετους τροχούς και παρατηρείται όταν το όχημα διέρχεται από άσχημο οδόστρωμα. Στους τροχούς του ίδιου άξονα εμφανίζεται διαφορετική αντίδραση από το οδόστρωμα. Για τη λύση του προβλήματος, χρησιμοποιείται ο συνδυασμός όλων των παραπάνω φορτίσεων.

Πίνακας 7.1. Βάση Δεδομένων

20x20x2	40x30x2	70x40x3,2
---------	---------	-----------

30x30x2	50x30x2	80x20x2
40x40x2	50x40x2	80x40x2
50x50x2	50x40x2,5	80x40x2,5
60x60x2	50x40x3	80x40x3,2
70x70x3	60x20x2	100x40x3
80x80x3	60x40x3	100x40x3,2
90x90x3	60x40x3,2	100x50x3
100x100x3,2	70x20x2	100x50x3,2
30x20x2	70x30x2	
40x20x2	70x40x3	

Πριν την έναρξη της επαναληπτικής διαδικασίας μέσω του αλγορίθμου, γίνεται η επιλογή ενός αρχικού μοντέλου προσομοίωσης. Η ομαδοποίηση των στοιχείων του λεωφορείου φαίνεται στο παρακάτω σχήμα.



Εικόνα 7.2. Αρχικό Μοντέλο

Μετά την τυχαία επιλογή των 15 μεταβλητών σχεδιασμού για τον κάθε γονέα, υπολογίζεται το αρχικό βάρος του λεωφορείου, το οποίο ισούται με τον μέσο όρο των αντικειμενικών συναρτήσεων κάθε γονέα. Επίσης, ο αρχικός σχεδιασμός προκύπτει από τον γονέα του οποίου η τιμή της αντικειμενικής συνάρτησης είναι η μικρότερη. Άρα:

Πίνακας 7.2. Αρχικό Βάρος

	Αλγόριθμος		
	ES(6,6)	ES(6,12)	ES(12,12)
Αρχικό Βάρος(t)	3,5726	3,5726	3,4315

Πίνακας 7.3. Αρχικός Σχεδιασμός

ES(6,6)	ES(6,12)	ES(12,12)
70x40x3	70x40x3	50x40x2,5
70x20x2	70x20x2	70x40x3

50x40x2,5	50x40x2,5	50x40x2
80x20x2	80x20x2	50x40x2
80x80x3	80x80x3	70x70x3
60x40x3	60x40x3	60x20x2
50x50x2	50x50x2	70x70x3
100x100x3,2	100x100x3,2	20x20x2
100x40x3,2	100x40x3,2	80x40x2,5
70x20x2	70x20x2	40x30x2
70x40x3	70x40x3	70x20x2
70x30x2	70x30x2	50x30x2
100x50x3,2	100x50x3,2	50x40x2
80x20x2	80x20x2	70x20x2
100x50x3,2	100x50x3,2	50x30x2

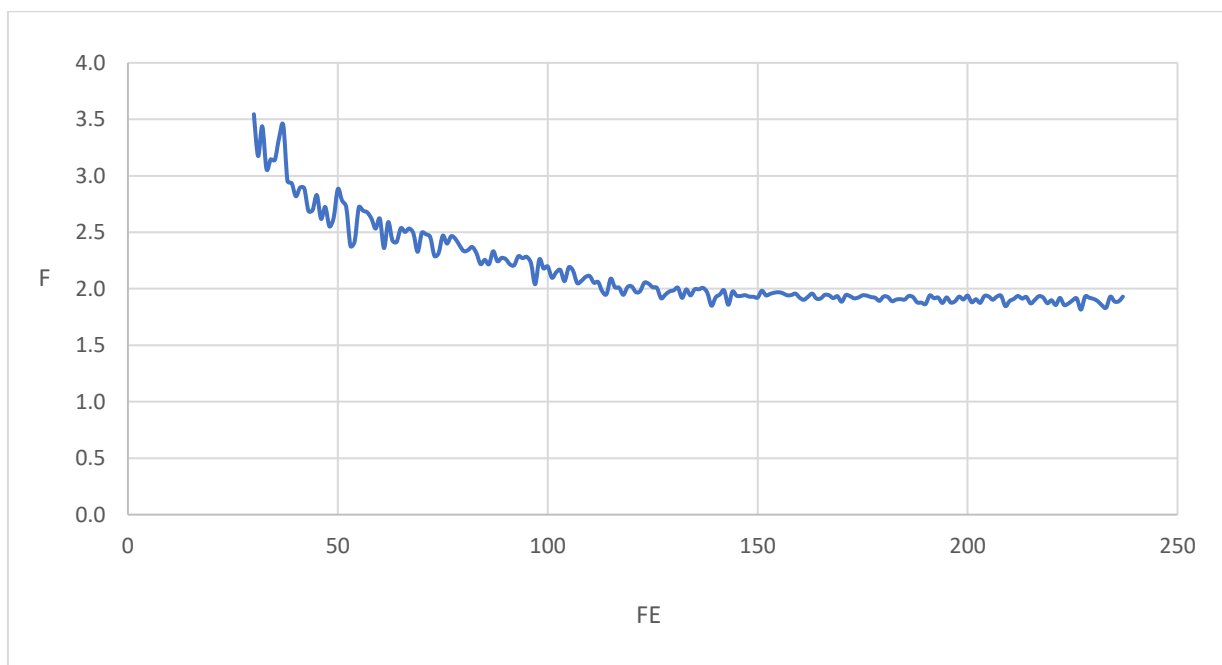
Τελικά, μετά από μια σειρά επαναλήψεων προκύπτει:

Πίνακας 7.4.

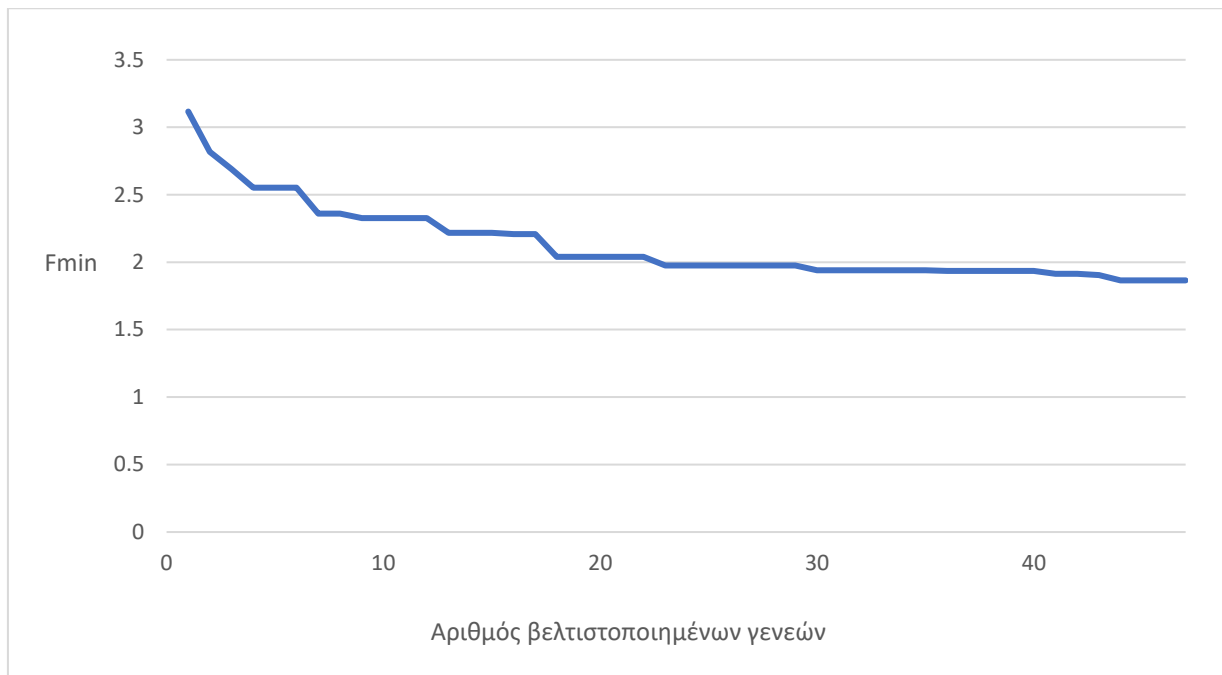
	Αλγόριθμος		
	ES(6,6)	ES(6,12)	ES(12,12)
Ελάχιστο Βάρος (Fmin)	1,8652	2,0788	1,7837
Αριθμός Γενεών	48	42	89
Αριθμός Αναλύσεων (FE)	237	180	536

Στον παραπάνω πίνακα, για 6-6, 6-12 και 12-12 γονείς-απογόνους, φαίνεται ο αριθμός των επαναλήψεων που έγινε ώστε να επιτευχθεί η βέλτιστη λύση, ο αριθμός των γενεών που παρουσίασαν βελτιστοποίηση, καθώς και το βέλτιστο βάρος του λεωφορείου. Παρακάτω φαίνονται διαγράμματα, που δείχνουν την πορεία του βάρους συναρτήσει του αριθμού των αναλύσεων και την πορεία του βελτιστοποιημένου βάρους συναρτήσει των βελτιστοποιημένων γενεών.

Για 6-6 γονείς-απογόνους:

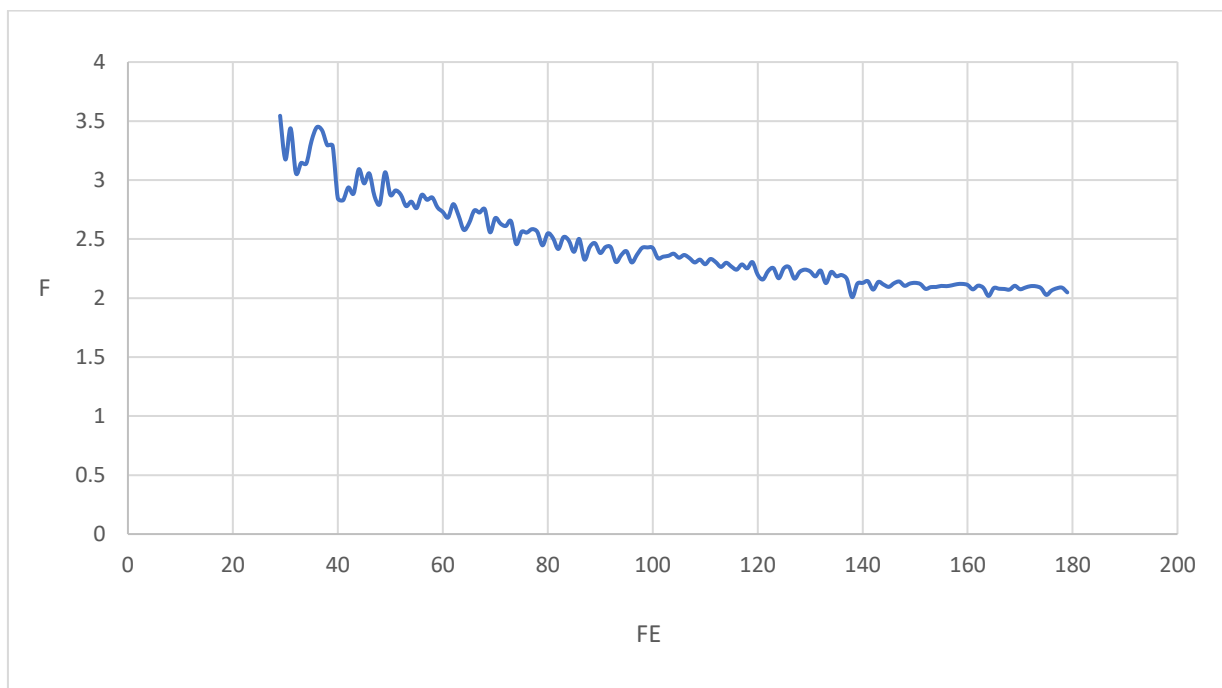


Διάγραμμα 7.1. Βάρους(F)- Αριθμού Αναλύσεων

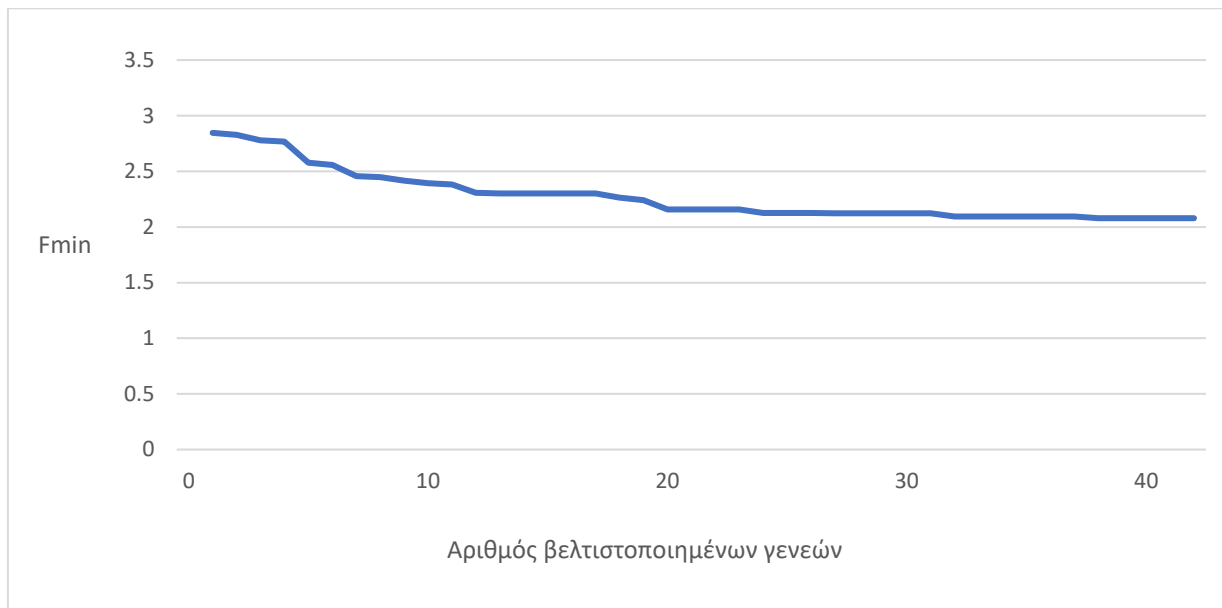


Διάγραμμα 7.2.Βελτ. Βάρους(Fmin)-Βελτ. Αριθμού Γενεών

Για 6-12 γονείς-απογόνους:

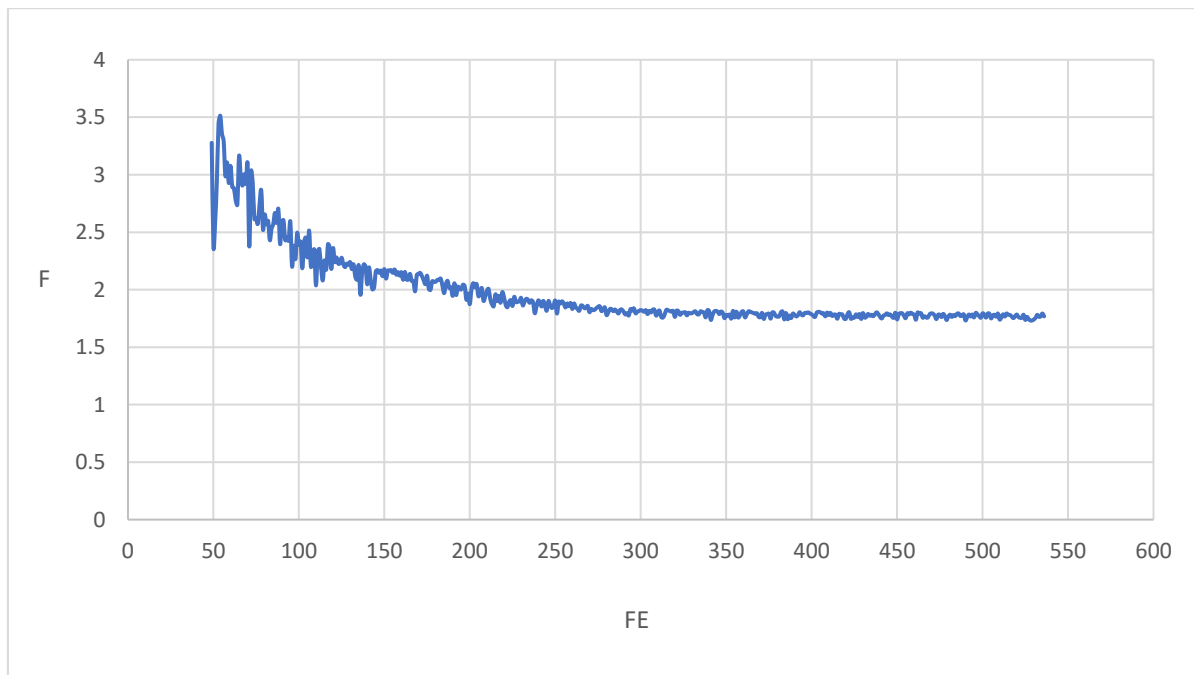


Διάγραμμα 7.3. Βάρους(F)- Αριθμού Αναλύσεων

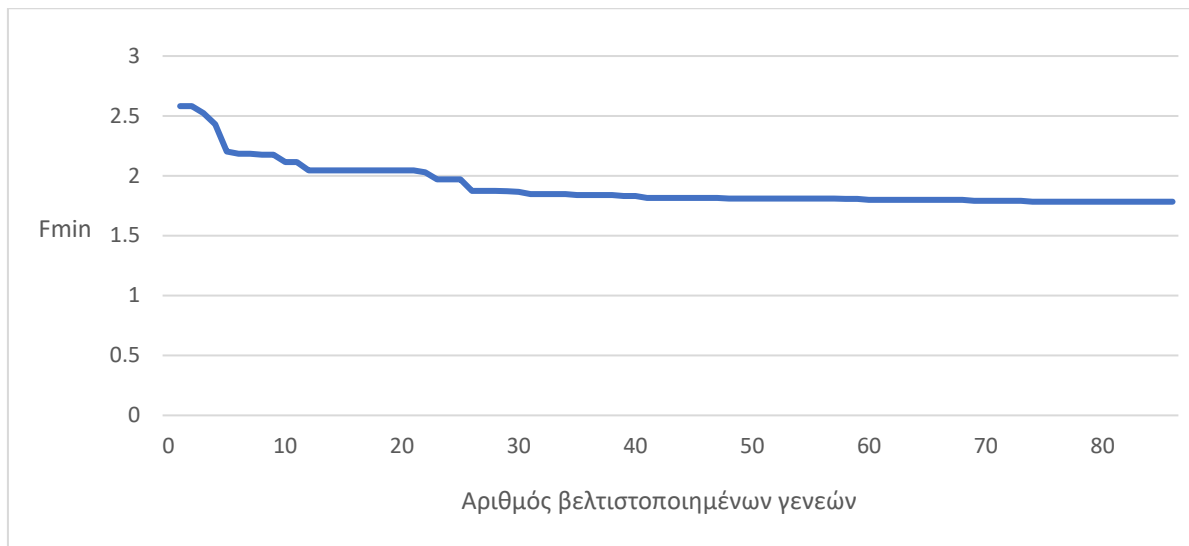


Διάγραμμα 7.4.Βελτ. Βάρους(Fmin)-Βελτ. Αριθμού Γενεών

Για 12-12 γονείς-απογόνους:



Διάγραμμα 7.5. Βάρους(F)- Αριθμού Αναλύσεων



Διάγραμμα 7.6. Βελτ. Βάρους(Fmin)-Βελτ. Αριθμού Γενεών

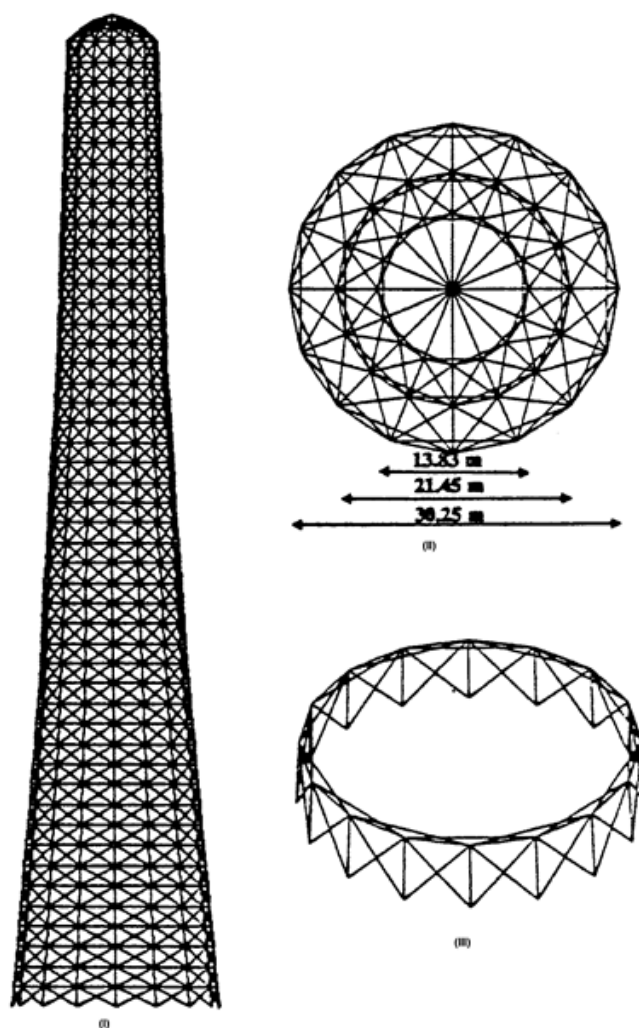
Ο τελικός σχεδιασμός που προέκυψε είναι:

Πίνακας 7.5. Τελικός Σχεδιασμός

ES(6,6)	ES(6,12)	ES(12,12)
100x40x3	50x40x3	50x40x2
70x20x2	70x20x2	70x20x2
60x20x2	70x20x2	30x20x2
80x20x2	80x20x2	30x20x2
30x30x2	20x20x2	20x20x2
70x20x2	60x20x2	60x20x2
30x30x2	40x40x2	30x30x2
20x20x2	30x30x2	20x20x2
30x20x2	80x20x2	80x20x2
70x40x3	80x40x2	50x30x2
80x20x2	80x20x2	70x20x2
70x20x2	70x20x2	30x20x2
100x40x3	100x50x3	30x20x2
80x20x2	70x20x2	70x20x2
100x40x3	50x40x2,5	60x20x2

7.3 Εφαρμογή 2 – Μεταλλικός πύργος (frame)

Η δεύτερη εφαρμογή αναφέρεται σε ένα μεταλλικό πύργο 147 ορόφων, με ύψος 564,06m. Οι 147 όροφοι αποτελούνται από 49 υπομονάδες όπως φαίνεται στο παρακάτω σχήμα.



Εικόνα 7.3. (I) Μεταλλικός πύργος, (II) Οι 3 πάνω όροφοι, (III) Μια υπομονάδα

Η κατασκευή αποτελείται από 2416 μέλη (beam members), διατομής διπλού L, που συνδέονται μεταξύ τους με αρθρώσεις. Το πλήθος των μεταβλητών σχεδιασμού είναι 26, οι βαθμοί ελευθερίας είναι 4806 και το πλήθος των κόμβων είναι 817.

Η φόρτιση που λαμβάνεται υπόψη για το σχεδιασμό του πύργου είναι 20 kN/m^2 , η οποία περιλαμβάνει τα μόνιμα φορτία, τα κινητά και το ίδιο βάρος. Επίσης θεωρείται ότι η κατασκευή βρίσκεται σε περιοχή με άνεμο 8 kN/m^2 . Για τη λύση του προβλήματος, χρησιμοποιείται ο συνδυασμός όλων των παραπάνω φορτίσεων.

Όμοια με την 1^η εφαρμογή, πριν την έναρξη της επαναληπτικής διαδικασίας μέσω του αλγορίθμου, γίνεται η επιλογή ενός αρχικού μοντέλου προσομοίωσης.

Πίνακας 7.6. Βάση Δεδομένων

30x3-0	45x4,5-5	80x8-8	120x12-14
35x3,5-0	50x5-5	90x9-8	150x15-14
40x4-0	60x6-5	100x10-8	180x18-14
45x4,5-0	30x3-6	120x12-8	200x20-14
50x5-0	35x3,5-6	50x5-10	90x9-16
60x6-0	40x4-6	60x6-10	100x10-16
70x7-0	45x4,5-6	70x7-10	120x12-16
80x8-0	50x5-6	80x8-10	150x15-16

90x9-0	60x6-6	90x9-10	180x18-16
100x10-0	70x7-6	100x10-10	200x20-16
120x12-0	40x4-7	120x12-10	100x10-18
150x15-0	45x4,5-7	150x15-10	120x12-18
180x18-0	50x5-7	60x6-12	150x15-18
200x20-0	60x6-7	70x7-12	180x18-18
30x3-4	70x7-7	80x8-12	200x20-18
35x3,5-4	80x8-7	90x9-12	100x10-20
40x4-4	90x9-7	100x10-12	120x12-20
45x4,5-4	40x4-8	120x12-12	150x15-20
50x5-4	45x4,5-8	150x15-12	180x18-20
30x3-5	50x5-8	80x8-14	200x20-20
35x3,5-5	60x6-8	90x9-14	
40x4-5	70x7-8	100x10-14	

Το αρχικό βάρος και ο αρχικός σχεδιασμός είναι:

Πίνακας 7.7. Αρχικό Βάρος

	Αλγόριθμος		
	ES(6,6)	ES(6,12)	ES(12,12)
Αρχικό Βάρος(t)	3154,914	3154,914	3086,7194

Πίνακας 7.8. Αρχικός Σχεδιασμός

ES(6,6)	ES(6,12)	ES(12,12)
180x18-20	180x18-20	200x20-20
180x18-20	180x18-20	200x20-20
120x12-20	120x12-20	120x12-18
200x20-20	200x20-20	150x15-20
120x12-20	120x12-20	200x20-20
200x20-20	200x20-20	120x12-20
200x20-18	200x20-18	150x15-20
180x18-20	180x18-20	100x10-20
100x10-20	100x10-20	200x20-20
200x20-18	200x20-18	200x20-20
200x20-20	200x20-20	150x15-20
180x18-20	180x18-20	180x18-18
120x12-20	120x12-20	200x20-20
100x10-20	100x10-20	200x20-20
120x12-20	120x12-20	150x15-16
150x15-20	150x15-20	120x12-20
120x12-20	120x12-20	150x15-18
150x15-20	150x15-20	200x20-16
200x20-20	200x20-20	100x20-18
200x20-16	200x20-16	200x20-20
180x18-18	180x18-18	200x20-18
200x20-16	200x20-16	100x10-20
180x18-20	180x18-20	100x10-20
100x10-18	100x10-18	150x15-20
120x12-18	120x12-18	150x15-20
200x20-20	200x20-20	120x12-20

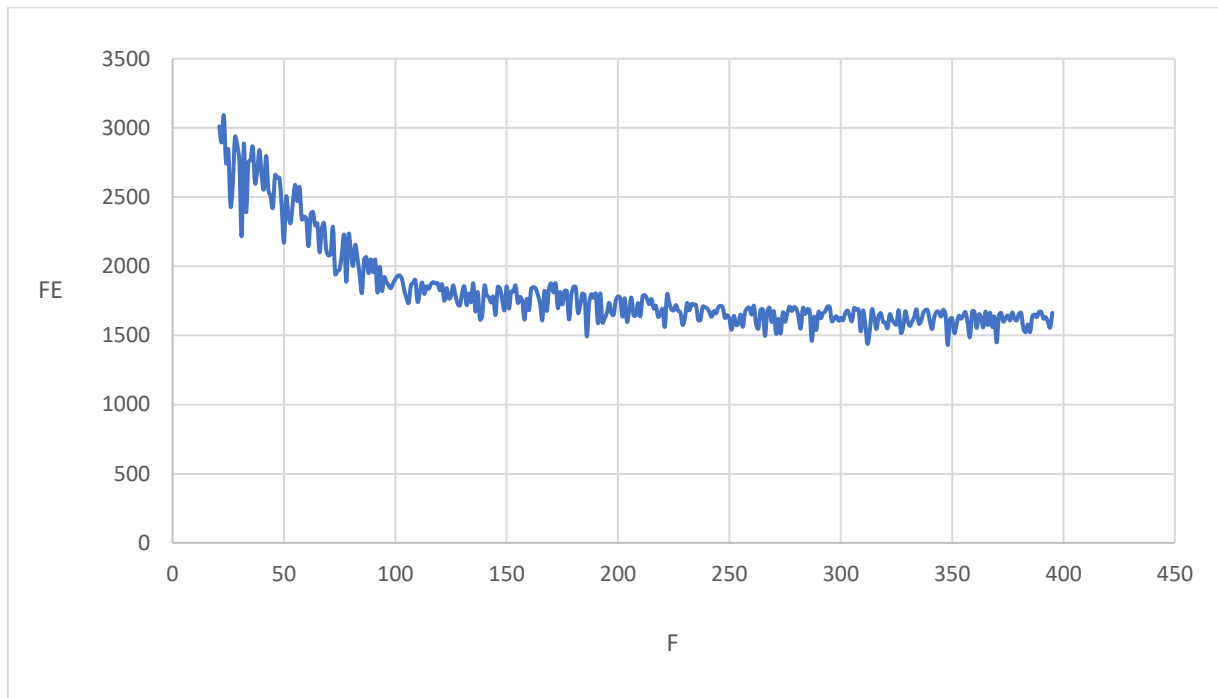
Τελικά, μετά από μια σειρά επαναλήψεων προκύπτει:

Πίνακας 7.9.

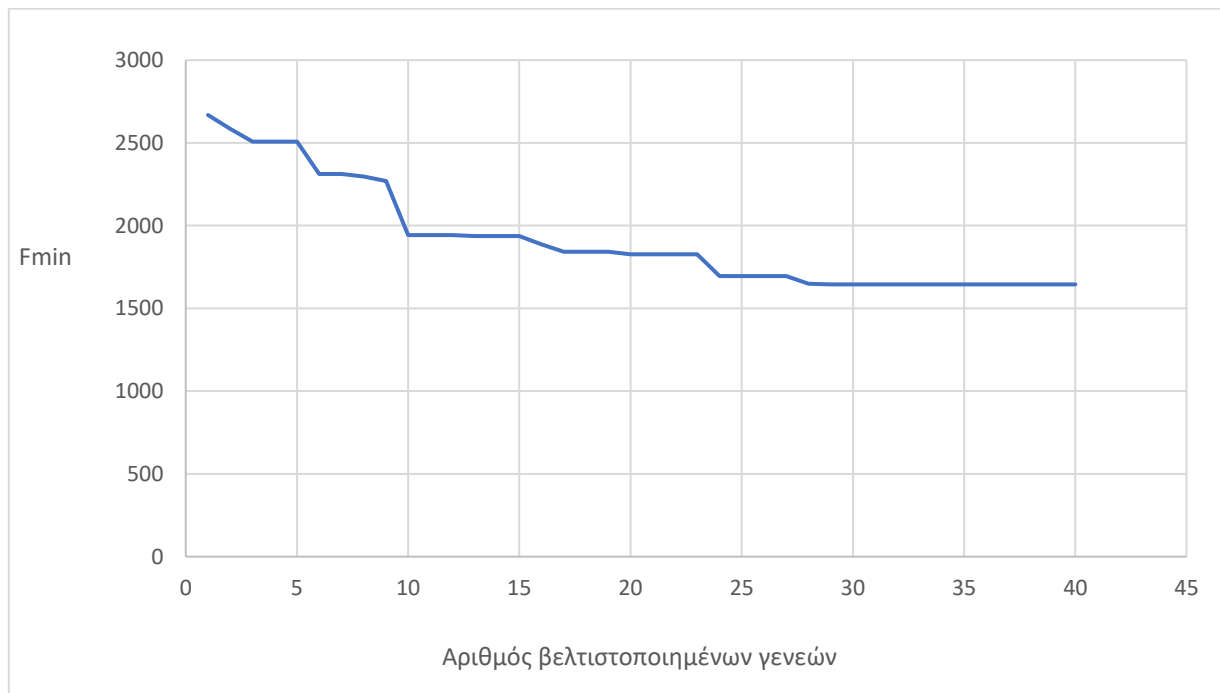
	Αλγόριθμος		
	ES(6,6)	ES(6,12)	ES(12,12)
Ελάχιστο Βάρος (Fmin)	1644,9903	1657,981	1547,9421
Αριθμός Γενεών	41	45	123
Αριθμός Αναλύσεων (FE)	395	600	2093

Παρακάτω φαίνονται διαγράμματα, που δείχνουν την πορεία του βάρους συναρτήσει του αριθμού των αναλύσεων και την πορεία του βελτιστοποιημένου βάρους συναρτήσει των βελτιστοποιημένων γενεών καθώς και ο τελικός σχεδιασμός.

Για 6-6 γονείς-απογόνους:

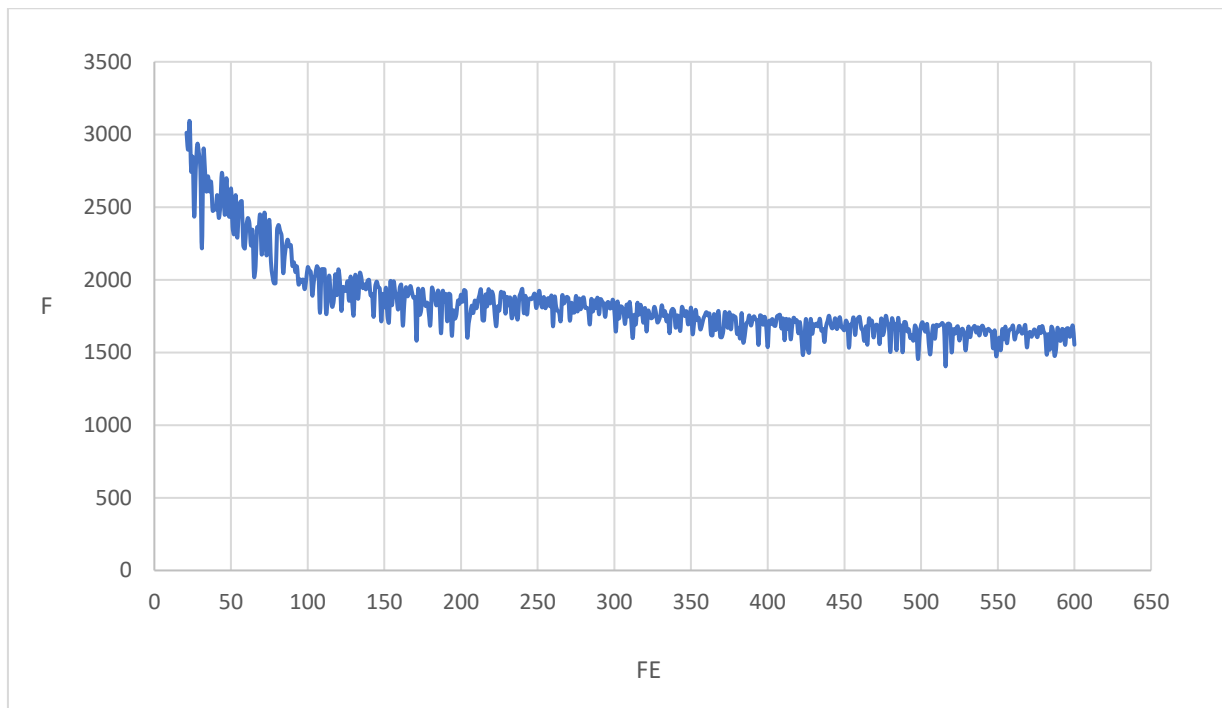


Διάγραμμα 7.7. Βάρους(F)- Αριθμού Αναλύσεων

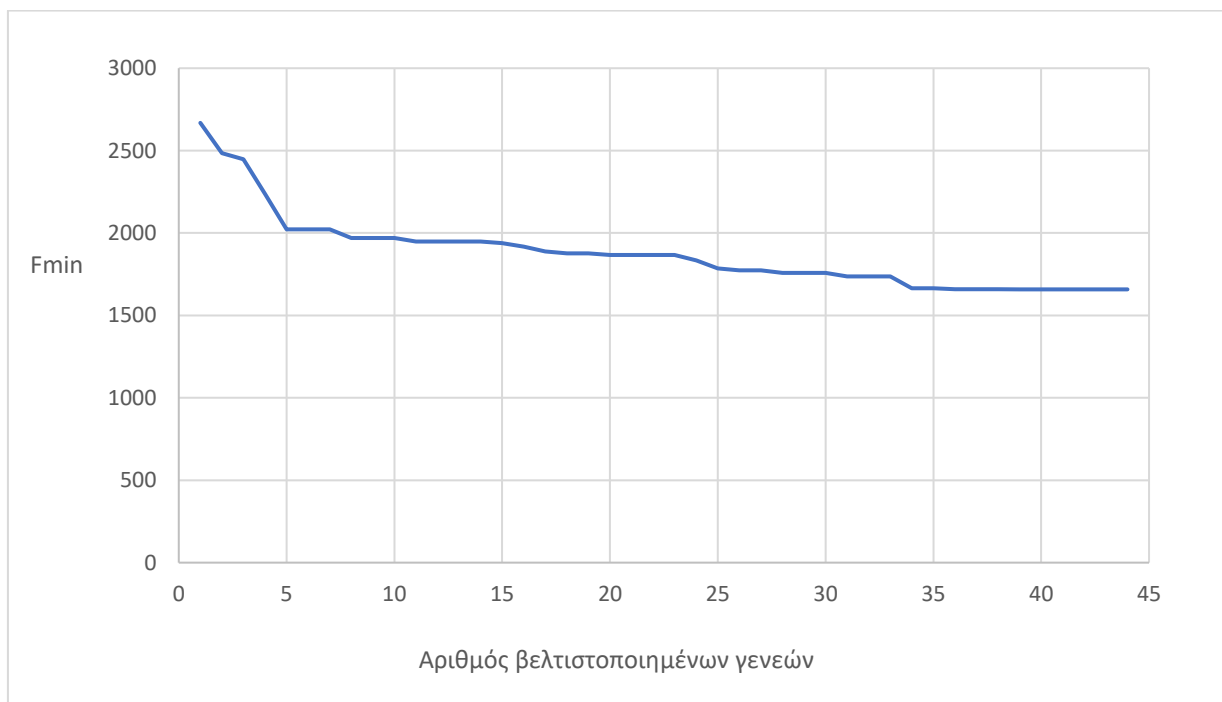


Διάγραμμα 7.8.Βελτ. Βάρους(Fmin)-Βελτ. Αριθμού Γενεών

Για 6-12 γονείς-απογόνους:

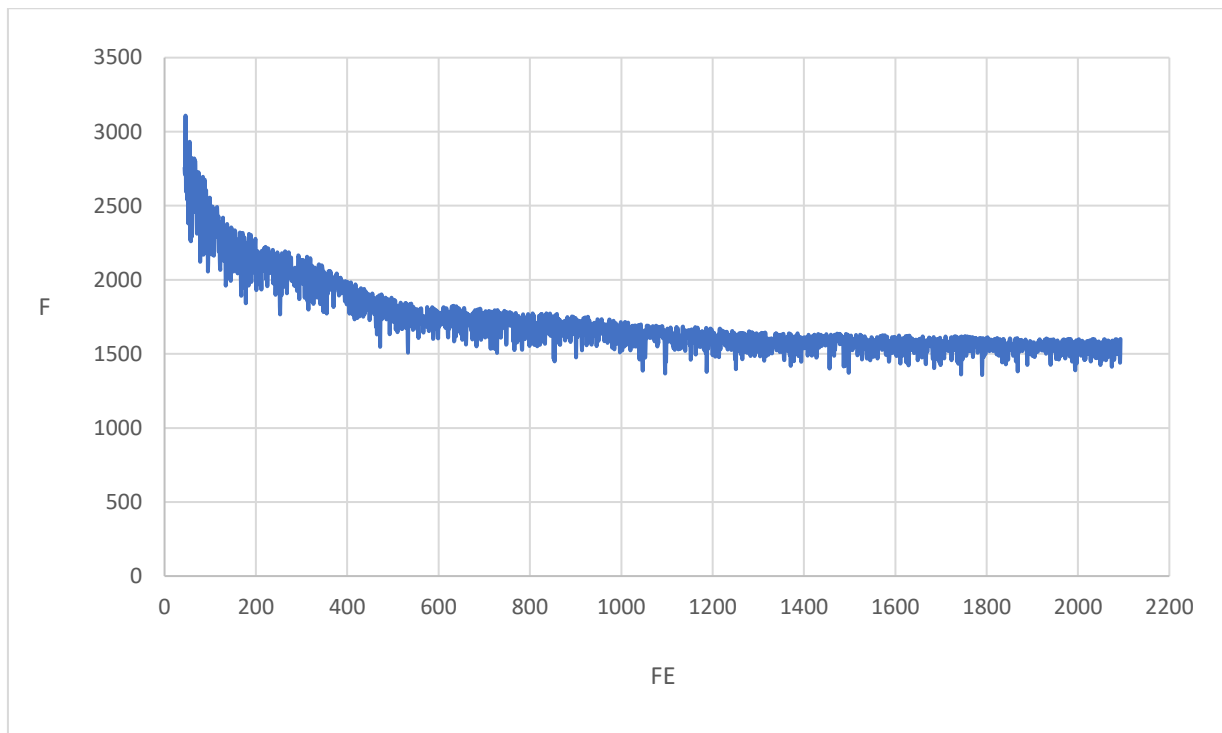


Διάγραμμα 7.9. Βάρους(F)- Αριθμού Αναλύσεων

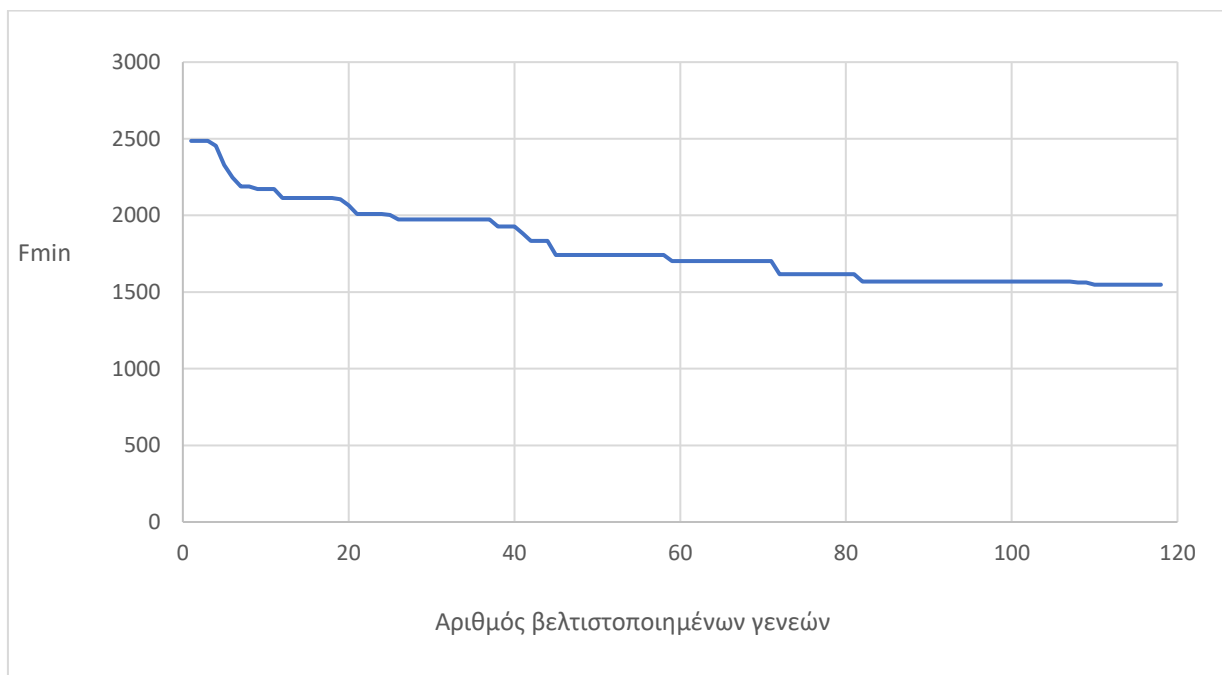


Διάγραμμα 7.10.Βελτ. Βάρους(Fmin)-Βελτ. Αριθμού Γενεών

Για 12-12 γονείς-απογόνους:



Διάγραμμα 7.11. Βάρους(F)- Αριθμού Αναλύσεων



Διάγραμμα 7.12. Βελτ. Βάρους(Fmin)-Βελτ. Αριθμού Γενεών

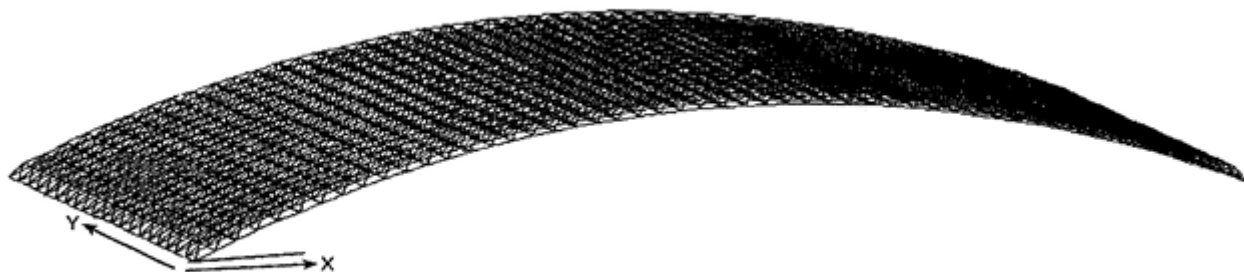
Πίνακας 7.10. Τελικός Σχεδιασμός

ES(6,6)	ES(6,12)	ES(12,12)
150x15-20	150x15-20	150x15-20
150x15-20	150x15-20	200x20-20
120x12-20	150x15-20	120x12-18
180x18-18	180x18-20	150x15-18
150x15-20	150x15-20	150x15-20
100x10-20	100x10-20	100x10-20
100x10-20	100x10-18	120x12-20
100x10-16	150x15-20	90x9-16
200x20-18	120x12-20	150x15-20
180x18-20	200x20-18	150x15-20

150x15-20	150x15-20	120x12-20
180x18-20	150x15-20	150x15-20
100x10-20	120x12-20	100x10-20
100x10-18	100x10-20	100x10-20
120x12-20	120x12-20	120x12-16
120x12-20	150x15-20	150x15-20
100x10-20	100x10-20	100x10-20
180x18-20	150x15-20	100x10-18
150x15-20	150x15-20	150x15-18
180x18-16	150x15-16	200x20-20
120x12-18	100x10-18	120x12-18
100x10-20	100x10-16	100x10-20
150x15-20	120x12-20	100x10-20
100x10-18	100x10-16	100x10-20
100x10-18	100x10-20	120x12-20
180x18-18	200x20-20	150x15-20

7.4 Εφαρμογή 3 – Μεταλλικό στέγαστρο (truss)

Η τρίτη εφαρμογή αναφέρεται σε ένα κυρτό μεταλλικό στέγαστρο που φαίνεται στο παρακάτω σχήμα.



Εικόνα 7.4. Μεταλλικό κυρτό στέγαστρο

Η κατασκευή αποτελείται από 42768 μέλη κυκλικής διατομής, που συνδέονται μεταξύ τους με αρθρώσεις. Το πλήθος των μεταβλητών σχεδιασμού είναι 8, οι βαθμοί ελευθερίας είναι 32418 και το πλήθος των κόμβων είναι 10846. Η βάση δεδομένων των διατομών των στοιχείων είναι η ακόλουθη:

Πίνακας 7.11. Βάση Δεδομένων

ΤΥΠΟΠΟΙΗΜΕΝΕΣ ΔΙΑΤΟΜΕΣ	
60,30X3,25	152,40X5,40
76,10X4,50	165,10X5,40
88,90X4,85	193,70X5,90
114,30X5,40	219,10X5,90
139,70X5,40	273,00X5,90

Η φόρτιση που λαμβάνεται υπόψη για το σχεδιασμό του στεγάστρου είναι 50 kN/m^2 , η οποία περιλαμβάνει τα μόνιμα φορτία, τα κινητά και το ίδιο βάρος. Επίσης στο κέντρο υπάρχει συγκεντρωμένο φορτίο 50 kN , λόγω γερανού. Για τη λύση του προβλήματος, χρησιμοποιείται ο συνδυασμός όλων των παραπάνω φορτίσεων.

Όμοια με τις άλλες εφαρμογές, πριν την έναρξη της επαναληπτικής διαδικασίας μέσω του αλγορίθμου, γίνεται η επιλογή ενός αρχικού μοντέλου προσομοίωσης. Η ομαδοποίηση των στοιχείων του στεγάστρου φαίνεται παρακάτω:

Το αρχικό βάρος και ο αρχικός σχεδιασμός είναι:

Πίνακας 7.12. Αρχικό Βάρος

	Αλγόριθμος		
	ES(6,6)	ES(6,12)	ES(12,12)
Αρχικό Βάρος(t)	4532,3630	4532,3630	4393,5240

Πίνακας 7.13. Αρχικός Σχεδιασμός

ES(6,6)	ES(6,12)	ES(12,12)
273,00x5,90	273,00x5,90	273,00x5,90
273,00x5,90	273,00x5,90	273,00x5,90
273,00x5,90	273,00x5,90	139,70x5,40
273,00x5,90	273,00x5,90	219,10x5,90
273,00x5,90	273,00x5,90	273,00x5,90
152,40x5,40	152,40x5,40	219,10x5,90

273,00x5,90	273,00x5,90	273,00x5,90
152,40x5,40	152,40x5,40	152,40x5,40

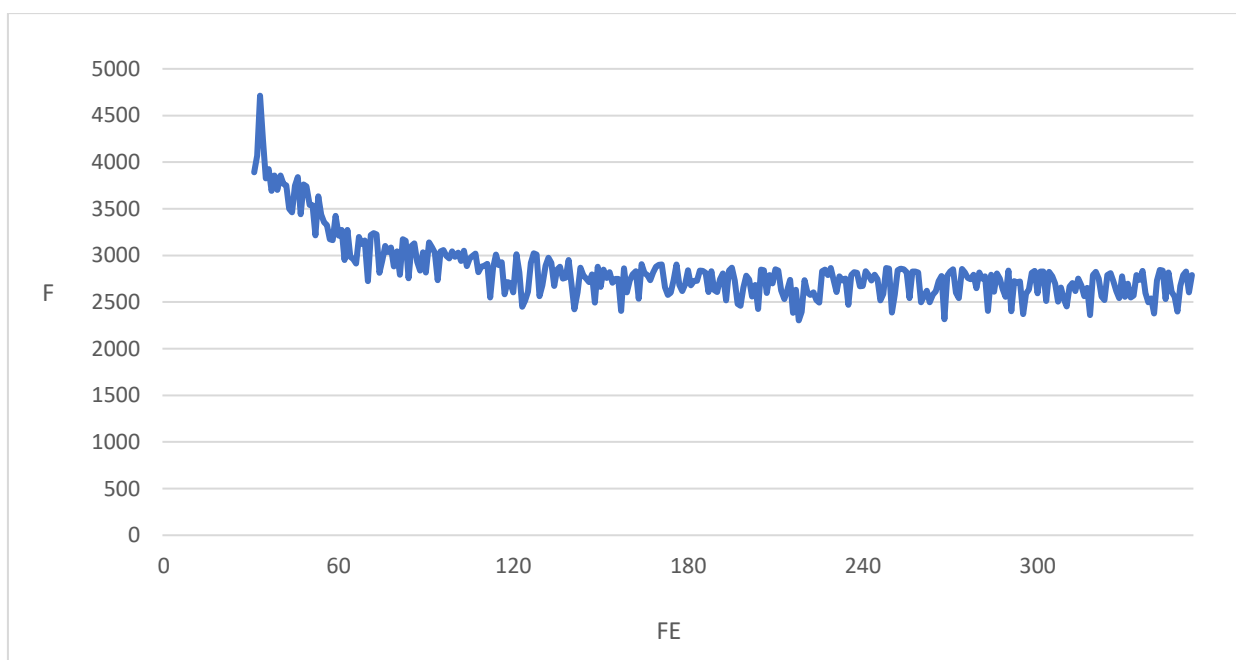
Τελικά, μετά από μια σειρά επαναλήψεων προκύπτει:

Πίνακας 7.14.

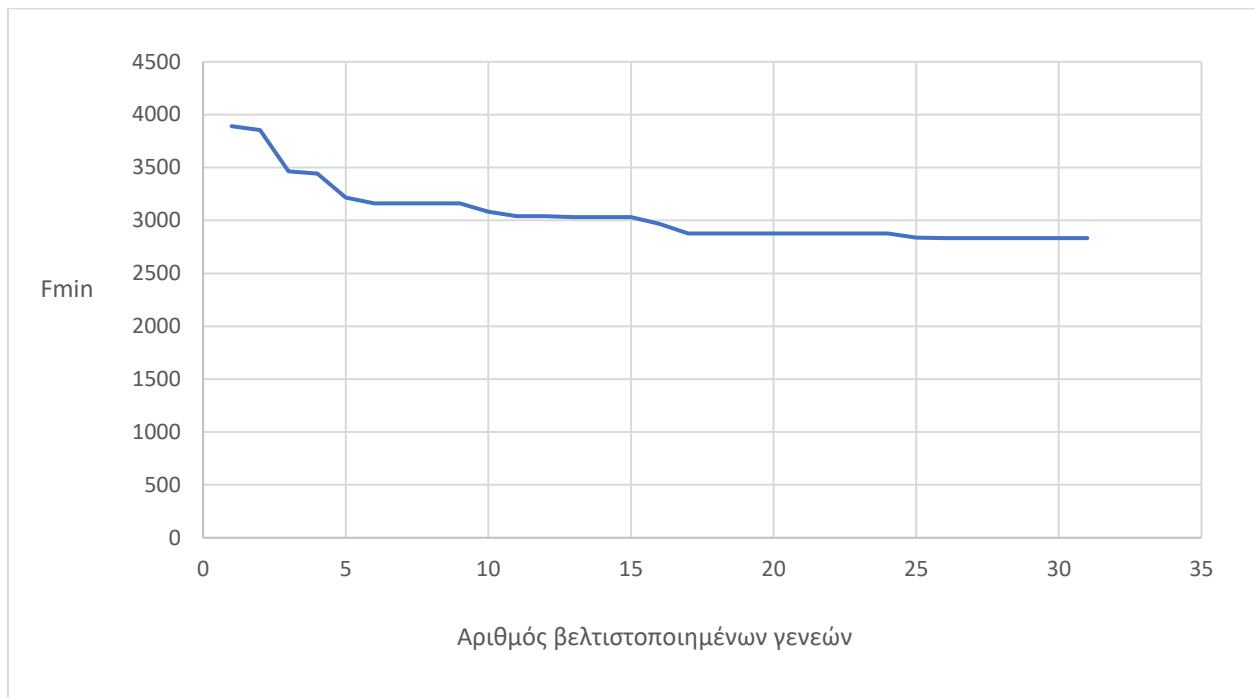
	Αλγόριθμος		
	ES(6,6)	ES(6,12)	ES(12,12)
Ελάχιστο Βάρος (Fmin)	2832,9604	2613,8496	2378,0922
Αριθμός Γενεών	31	49	104
Αριθμός Αναλύσεων (FE)	354	939	2099

Παρακάτω φαίνονται διαγράμματα, που δείχνουν την πορεία του βάρους συναρτήσει του αριθμού των αναλύσεων και την πορεία του βελτιστοποιημένου βάρους συναρτήσει των βελτιστοποιημένων γενεών καθώς και ο τελικός σχεδιασμός.

Για 6-6 γονείς-απογόνους:

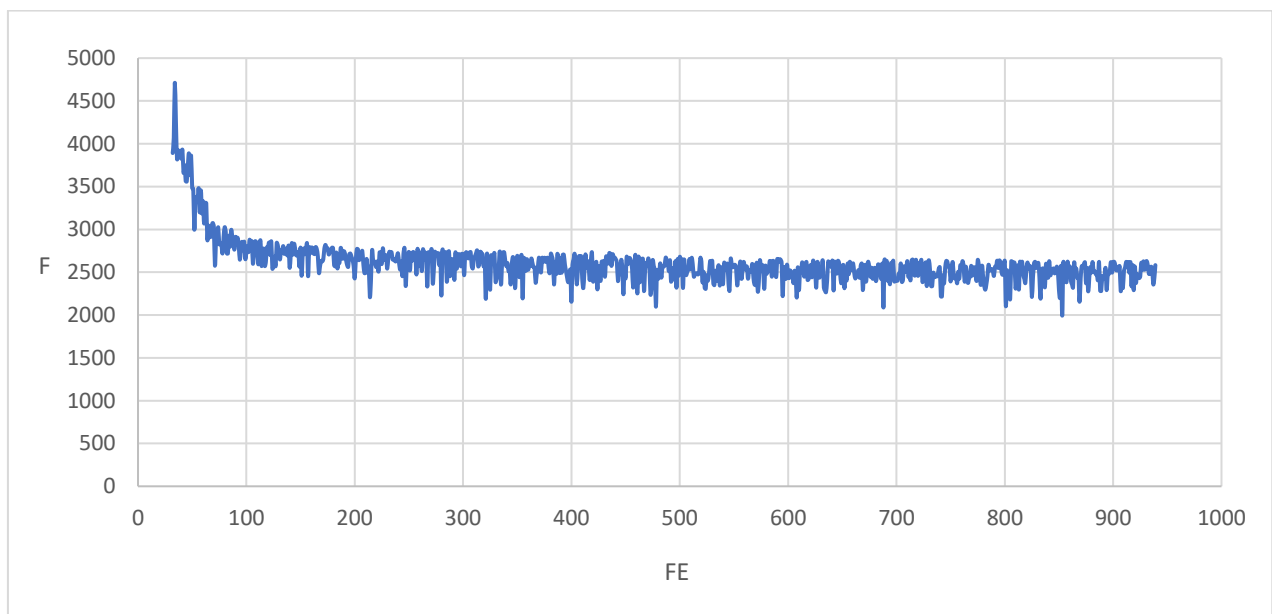


Διάγραμμα 7.13. Βάρους(F)- Αριθμού Αναλύσεων

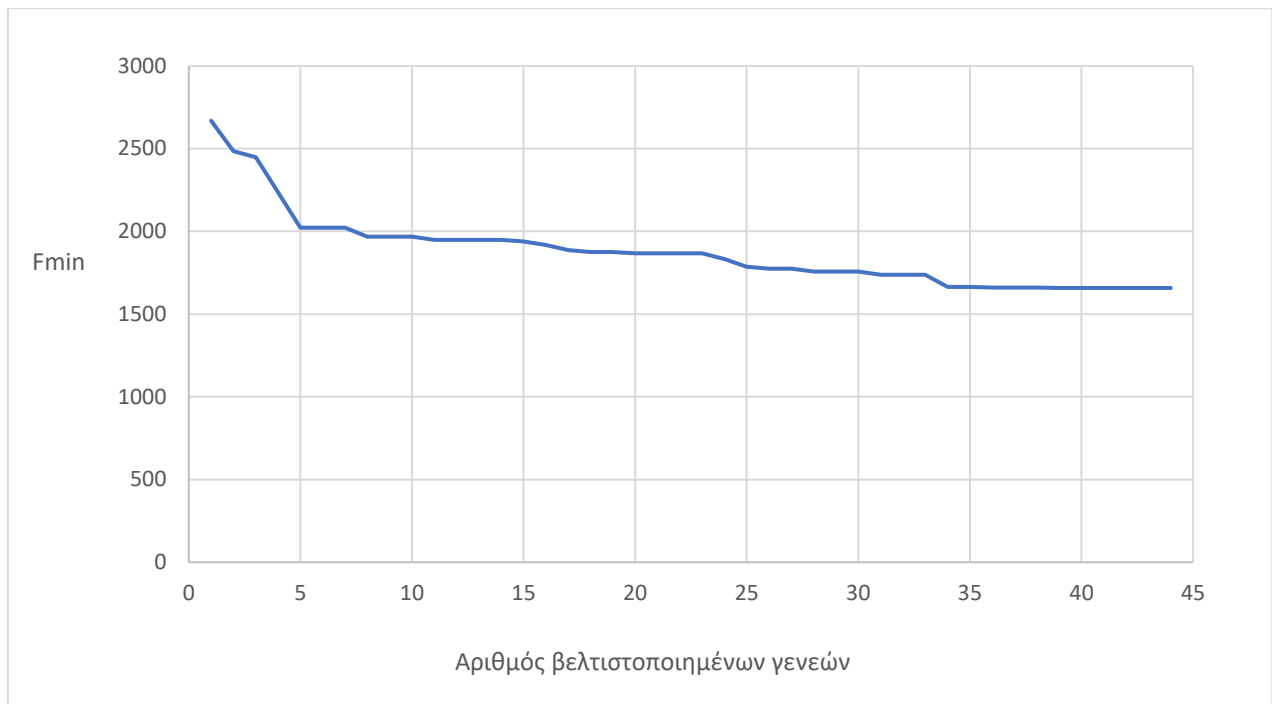


Διάγραμμα 7.14. Βελτ. Βάρους(Fmin)-Βελτ. Αριθμού Γενεών

Για 6-12 γονείς-απογόνους:

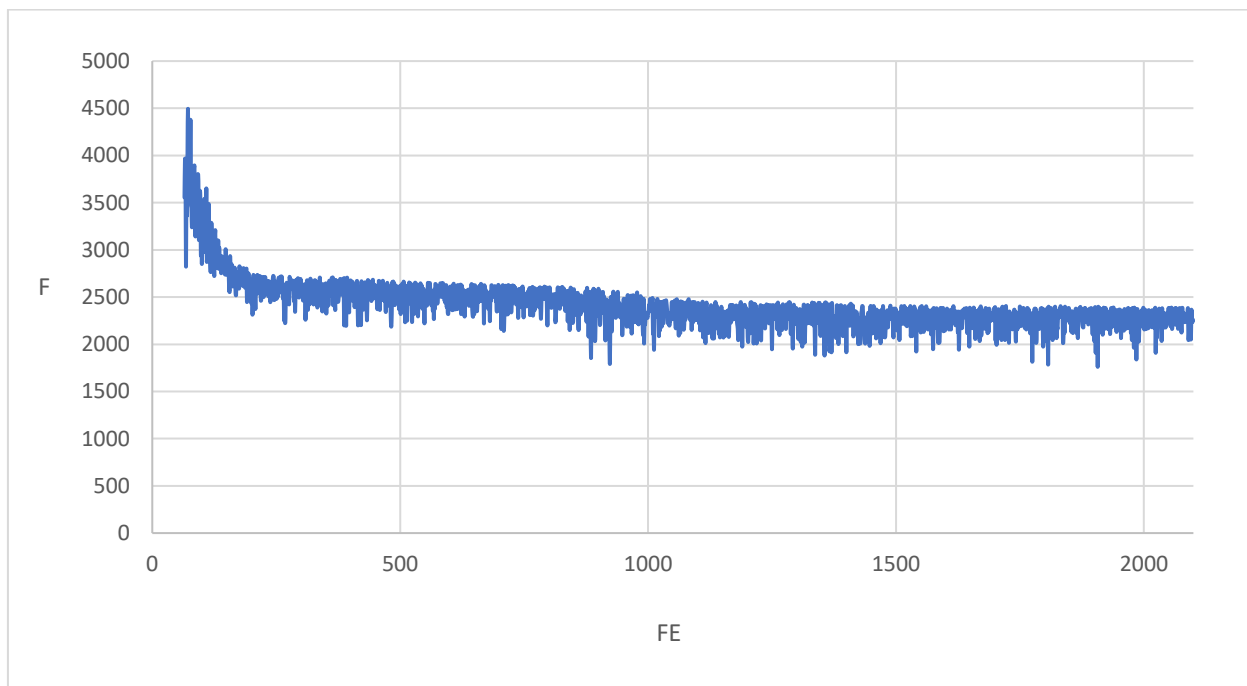


Διάγραμμα 7.15. Βάρους(F)- Αριθμού Αναλύσεων

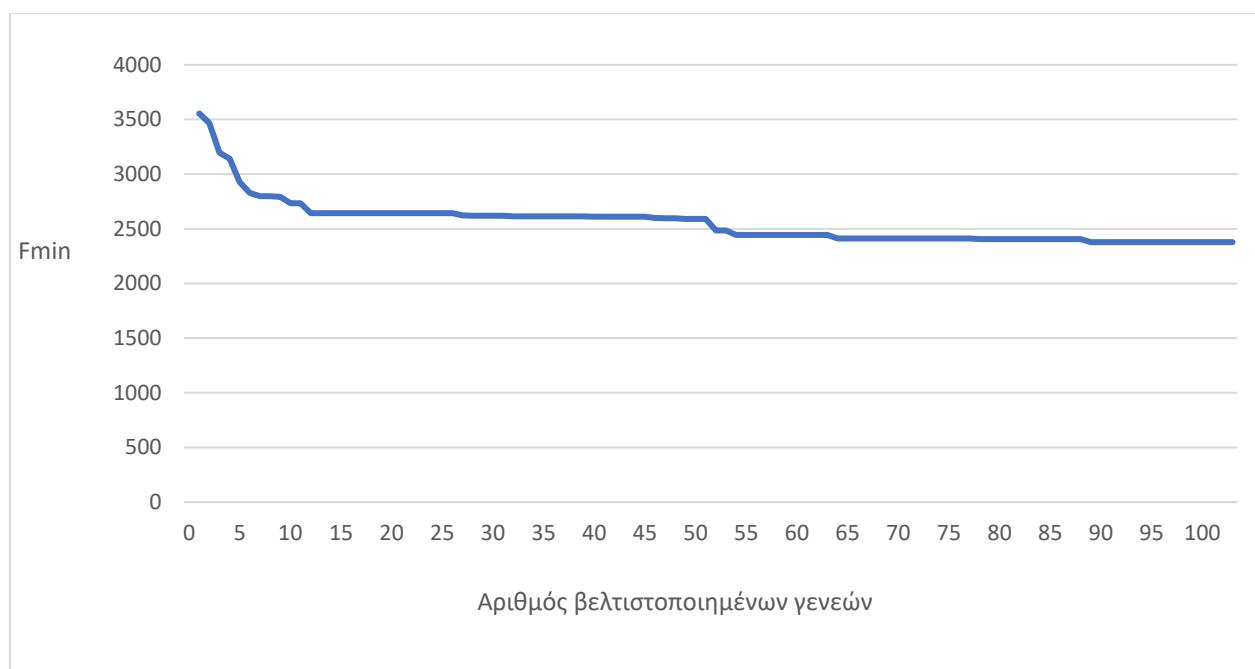


Διάγραμμα 7.16.Βελτ. Βάρους(Fmin)-Βελτ. Αριθμού Γενεών

Για 12-12 γονείς-απογόνους:



Διάγραμμα 7.17. Βάρους(F)- Αριθμού Αναλύσεων



Διάγραμμα 7.18. Βελτ. Βάρους(Fmin)-Βελτ. Αριθμού Γενεών

Πίνακας 7.15. Τελικός Σχεδιασμός

6-6	6-12	12-12
273,00x5,90	273,00x5,90	273,00x5,90
76,10x4,50	139,70x5,40	76,10x4,50
193,70x5,90	139,70x5,40	139,70x5,40
165,10x5,40	219,10x5,90	165,10x5,40
193,70x5,90	114,30x5,40	60,30x3,25
114,30x5,40	114,30x5,40	114,30x5,40
273,00x5,90	273,00x5,90	273,00x5,90
114,30x5,40	165,10x5,40	60,30x3,25

8 ΒΙΒΛΙΟΓΡΑΦΙΑ

- Adeli H. and Cheng N.T., 'Augmented Lagrangian Genetic Algorithm for structural optimization', *Journal of Aerospace Engineering, ASCE*, 7(1), 104-118, (1994).
- Adeli H. and Cheng N.T., 'Concurrent Genetic Algorithms for optimization of large structures', *Journal of Aerospace Engineering, ASCE*, 7(3), 276-296, (1994).
- Adeli H. and Cheng N.T., 'Integrated Genetic Algorithm for optimization of space structures', *Journal of Aerospace Engineering, ASCE*, 6(4), 315-328, (1993).
- Adeli H. and Hyo Seon Park 'Neural dynamics model for structural optimisation. - Theory', *Computers & Structures*, 57(3), pp. 383-399, 1995.
- Adeli H. and Hyo Seon Park 'Optimization of space structures by Neural dynamics', *Neural Networks*, 8(5), pp. 769-781, 1995.
- AISC-ASD *Manual of Steel Construction-allowable Stress Design*, 9th Edn. American Institute of Steel Construction, Chicago, 1989.
- Allen M, Maute K. Reliability-based shape optimization of structures undergoing fluid-structure interaction phenomena. *Computer Methods in Applied Mechanics and Engineering*; 194(30-33): 3472-3495, (2005).
- Altair OptiStruct, <http://www.altairhyperworks.com/Product.19.OptiStruct.aspx>, (last accessed September 2013).
- Ang, A.H.-S., Tang, W.H., *Probabilistic concepts in engineering planing and design. Decision, risk and reliability*, Vol. II. (New York, NY): Wiley; 1984.
- Arora J.S., (1990) Computational Design Optimization: A review and future directions, *Structural Safety*, 7, 131-148.
- Arslan M.A. and Hajela P. 'Counterpropagation Neural Networks in decomposition based optimal design', *Computers & Structures*, 65(5), pp. 641-650, 1997.
- ARUP, <http://www.arup.com/>, (last accessed September 2013).
- ASCE/SEI Standard 41-06, *Seismic Rehabilitation of Existing Buildings*, prepublication edition, Structural Engineering Institute, American Society of Civil Engineers, 2006.
- ATC-58, *Guidelines for Seismic Performance Assessment of Buildings*, Applied Technology Council, Redwood City, 2009.
- Back T., Hoffmeister F. and Schwefel H.-P., 'A survey of evolution strategies', in R. K. Belew and L. B. Booker (eds.), *Proceedings of the 4th International Conference on Genetic Algorithms*, Morgan Kaufmann, 2-9, (1991).
- Bäck, T, Schwefel H-P. An overview of evolutionary algorithms for parameter optimization, *Journal of Evolutionary Computation*; 1(1): 1-23, (1993).
- Baker J., 'Adaptive selection methods for genetic algorithms', in J. J. Grefenstette (ed.), *Proc. International Conf. On Genetic Algorithms and their applications*, Lawrence Erlbaum, (1985).
- Balling, R.J., Yao, X., Optimization of reinforced concrete frames, *Journal of Structural Engineering* 1997; **123**(2): 193-202.
- Barakat, S., Bani-Hani, K., Taha, M.Q., Multi-objective reliability-based optimization of prestressed concrete beams, *Structural Safety* 2004; **26**(3): 311-342.
- Barricelli N.A., 'Numerical testing of evolution theories', *ACTA Biotheoretica*, 16, 69-126, (1962).
- Barricelli, N.A., Numerical testing of evolution theories, *ACTA Biotheoretica*; **16**: 69-126, 1962.
- Belegundu A.D. and Arora J.S., 'A computational study of transformation methods for optimal design', *AIAA Journal*, 22(4), 535-542, (1984).
- Bendsoe M.P., and Kikuchi N., Generating optimal topologies in structural design using a homogenization method, *Computer Methods in Applied Mechanics and Engineering*, 71, 197-224, 1988.
- Berke L. and Hajela P., (1990) Applications of Artificial Neural Nets in Structural Mechanics, NASA 331-348 TM-102420.
- Berke L., Patnaik S.N. and Murthy P.L.N., 'Optimum Design of Aerospace Structural Components using Neural Networks', *Computers & Structures*, 48, pp. 1001-1010, 1993.

- Beyer H-G, Sendhoff B. Robust optimization - A comprehensive survey. *Computer Methods in Applied Mechanics and Engineering*; 196(33-34): 3190-3218, (2007).
- Bitoulas N. and Papadrakakis, M., An optimised computer implementation of the incomplete Cholesky factorization, *Comp. Systems in Engineering*, 5(3), 265-274, 1994.
- Bletzinger K.U., Kimmich S., and Ramm E., (1991) Efficient modelling in shape optimal design, *Computing Systems in Engineering*, 2(5/6), 483-495.
- BS5950: *Structural use of steelwork in building*, Part 1: Code of Practice for Design - Rolled and Welded Sections, British Standards Institution, 2000.
- Cai J. and Thierauf G., 'Discrete structural optimization using evolution strategies', in B.H.V. Topping and A.I. Khan (eds.), *Neural networks and combinatorial in civil and structural engineering*, Edinburgh, Civil-Comp Limited, 95-10, (1993).
- Chan, C.-M., Wang, Q., Nonlinear stiffness design optimization of tall reinforced concrete buildings under service loads, *Journal of Structural Engineering* 2006; 132(6): 978-990.
- Chan, C.-M., Zou, X.-K., Elastic and inelastic drift performance optimization for reinforced concrete buildings under earthquake loads, *Earthquake Engineering Structural Dynamics* 2004; **33**(8): 929-950.
- Charpis, D.C., Lagaros, N.D., Papadrakakis, M., Multi-database exploration of large design spaces in the framework of cascade evolutionary structural sizing optimization, *Computer Methods in Applied Mechanics and Engineering*; **194**(30-33): 3315-3330, 2005.
- Coello Coello C.A., An updated survey of GA-based multi-objective optimization techniques. *ACM Computing Surveys* 2000; **32**(2): 109-143.
- Coello Coello CA. An updated survey of GA-based multi-objective optimization techniques. *ACM Computing Surveys*; 32(2): 109-143, (2000).
- Daskalakis, K., <http://www.konstantiosdaskalakis.com>, (last accessed September 2013).
- Deb K, Pratap A, Agarwal S, Meyarivan T. A fast and elitist multi-objective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*; 6(2): 182-197, (2002).
- Der Kiureghian, A., First- and second-order reliability methods. Chapter 14 in *Engineering design reliability handbook*, Nikolaidis, E., Ghiocel, D.M., Singhal, S. (Eds.), CRC Press, Boca Raton, FL, 2005.
- Dorigo, M., Stützle, T., *Ant Colony Optimization*, The MIT Press, 2004.
- Ebrahimi, M., Farmani, M.R., Roshanian, J., Multidisciplinary design of a small satellite launch vehicle using particle swarm optimization, *Structural and Multidisciplinary Optimization*, **44**(6): 773-784, 2011.
- EC2. Eurocode 2. *Design of Concrete Structures-Part 1: General Rules and Rules for Buildings*. European Committee for Standardisation: Brussels, Belgium, The European Standard EN 1992-1-1: 2004.
- EC3. Eurocode 3: *Design of steel structures, Part 1.1: General rules and rules for buildings*. European Committee for Standardisation: Brussels, Belgium, The European Standard EN 1993-1-1: 2005.
- EC8. Eurocode 8: *Design of Structures for Earthquake Resistance*. European Committee for Standardisation: Brussels, Belgium, The European Standard EN 1998-1: 2004.
- Embrechts M.J., Metaneural-Version 4.1, Rensselaar Polytechnique Institute, 1994.
- Eschenauer H.A., Schumacher A., and Viator T., Decision makings for initial designs made of advanced materials, in Bendsoe M.P. and Soares C.A.M. (eds.), NATO ARW 'Topology design of structures', Sesimbra, Portugal, Kluwer Academic Publishers, Dordrecht, Netherlands, 469-480, 1993.
- Eurocode 3, Design of steel structures, Part1.1: General rules for buildings, CEN, ENV 1993-1-1/1992.
- Eurocode 8. Design provisions for earthquake resistance of structures. ENV1998, CEN European Committee for standardization, Brussels, 1996.
- European Committee for Standardization. Draft for Development of Eurocode 2: Design of concrete structures. ENV 1992-1-1, 1992.
- Farhat C. and Roux F.-X., Implicit Parallel Processing in Structural Mechanics, *Comp. Mec. Adv.*, 2, 1-124, 1994.
- Farhat C., Crivelli L. and Roux F.-X., Extending Substructure Based Iterative Solvers to Multiple Load and Repeated Analyses *Comp. Meth. Appl. Mech. Eng.*, 195-209, 1994.

- FEMA-356: *Prestandard and commentary for the seismic rehabilitation of buildings*. Federal Emergency Management Agency, Washington DC, SAC Joint Venture, 2000.
- FEMA-445: *Next-generation Performance-based Seismic Design Guidelines*, Program Plan for New and Existing Buildings. Federal Emergency Management Agency: Washington, DC, 2006.
- Fleury C., (1993) Dual methods for convex separable problems, in Rozvany G.I.N. (ed), NATO/DFG ASI 'Optimization of large structural systems', Berchtesgaden, Germany, Kluwer Academic Publishers, Dordrecht, Netherlands, 509-530.
- Fogel L.J., Owens A.J., and Walsh M.J., (1966) *Artificial intelligence through simulated evolution*, Wiley, New York.
- Frangopol, D.M., Computer-automated design of structural systems under reliability-based performance constraints, *Engineering Computations* 1986; **3**(2): 109-115.
- Gallagher R.H. and Zienkiewicz O.C., *Optimum Structural Design: Theory and Applications*, John Wiley & Sons, New York, 1973.
- Ganzerli, S., Pantelides, C.P., Reaveley, L.D. Performance-based design using structural optimization. *Earthquake Engineering Structural Dynamics* 2000; **29**(11): 1677-1690.
- Gasparini D.A. and Vanmarke E.H., "Simulated earthquake motions compatible with prescribed response spectra", Massachusetts Institute of Technology (MIT), Department of Civil Engineering, Publication No. R76-4, January 1976.
- Gasparini D.A., SIMQKE - A program for artificial motion generation, User's manual and documentation, Massachusetts Institute of Technology (MIT), Department of Civil Engineering, November 1976.
- Geem, Z.W., *Recent Advances in Harmony Search Algorithm*. Studies in Computational Intelligence, Springer, 2010.
- Genesis, <http://www.vrand.com/Genesis.html>, (last accessed September 2013).
- Ghobarah, A., Abou-Elfath, H., Biddah, A., Response-based damage assessment of structures, *Earthquake Engineering and Structural Dynamics*; **28**(1): 79-104, 1999.
- Gill P.E, Murray W., and Wright M.H., (1981) *Practical Optimization*, Academic Press.
- Gill P.E, Murray W., Saunders M.A., and Wright M.H., (1986) User's guide for NPSOL (Version 4.0): A Fortran Package for Nonlinear Programming, Technical Report SOL 86-2, Dept. of Operations Research, Stanford University.
- Giovenale, P., Cornell, C.A., Esteva, L., Comparing the adequacy of alternative ground motion intensity measures for the estimation of structural responses, *Earthquake Engineering and Structural Dynamics*; **33**(8): 951-979, 2004.
- Goldberg D., 'A note on boltzmann tournament selection for Genetic Algorithms and population-oriented Simulated Annealing', TCGA 90003, Engineering Mechanics, Alabama University, (1990).
- Goldberg D.E., (1989) *Genetic algorithms in search, optimization and machine learning*, Addison-Wesley Publishing Co., Inc., Reading, Massachusetts.
- Goldberg D.E., 'Sizing populations for serial and parallel genetic algorithms', TCGA Report No 88004, University of Alabama, (1988).
- Gunaratnam D.J. and Gero J.S. 'Effect of representation on the performance of Neural Networks in structural engineering applications', *Microcomputers in Civil Engineering*, Vol. 9, pp. 97-108, 1994.
- Hajela, P. and Berke, L., 'Neurobiological computational models in structural analysis and design', *Computers & Structures*, Vol. 41, pp. 657-667, 1991.
- Hansen, N., Ostermeier, A., Completely derandomized self-adaptation in evolution strategies. *Evolutionary Computations*; **9**(2): 159-195, 2001.
- Haug E.J. and Arora J.S., 'Optimal mechanical design techniques based on optimal control methods', ASME paper No 64-DTT-10, Proceedings of the 1st ASME design technology transfer conference, 65-74, New York, October, 1974.
- Herzog & De Meuron, <http://www.herzogdeameuron.com>, (last accessed September 2013).
- Hinton E., and Hassani B., Some experiences in structural topology optimization, in B.H.V. Topping (ed.) *Developments in Computational Techniques for Structural Engineering*, CIVIL-COMP Press, Edinburgh, 323-331, 1995.

- Hinton E., and Sienz J., (1994) Aspects of adaptive finite element analysis and structural optimization, in Topping B.H.V. and Papadrakakis M. (eds) *Advances in Structural Optimization*, CIVIL-COMP Press, Edinburgh, 1-26.
- Hinton E., and Sienz J., Fully stressed topological design of structures using an evolutionary procedure, *Journal of Engineering Computations*, 12, 229-244, 1993.
- Hinton, E. and Sienz, J., (1995) Studies with a robust and reliable structural shape optimization tool, in B.H.V. Topping (ed) *Developments in Computational Techniques for Structural Engineering*, CIVIL-COMP Press, Edinburgh, 343-358.
- Hoffmeister F., and Back T., (1991) Genetic Algorithms and Evolution Strategies-Similarities and Differences, in Schwefel, H.P. and Manner, R. (eds.), *Parallel Problems Solving from Nature*, Springer-Verlag, Berlin, Germany, 455-469.
- Holland J., (1975) *Adaptation in natural and artificial systems*, University of Michigan Press, Ann Arbor.
- Holland, J., *Adaptation in natural and artificial systems*, University of Michigan Press, Ann Arbor, Michigan, USA, 1975.
- Hooke R. and Jeeves T.A., 'Direct Search Solution of Numerical and Statistical Problems', *J. ACM*, Vol. 8, (1961).
- Hornlein, H.R.E.M., Schittkowski, K., *Software Systems for Structural Optimization*, International Series of Numerical Mathematics, Birkhauser, 1993.
- Igel, C., Hansen, N., Roth, S., Covariance matrix adaptation for multi-objective optimization. *Evolutionary Computations*; **15**(1): 1-28, 2007.
- Joines J. and Houck C., 'On the use of non-stationary penalty functions to solve non-linear constrained optimization problems with GA', in Z. Michalewicz, J. D. Schaffer, H.-P. Schwefel, D. B. Fogel, and H. Kitano (eds.), *Proceedings of the First IEEE International Conference on Evolutionary Computation*, 579-584. IEEE Press, (1994).
- Kanagasundaram, S., Karihaloo, B.L., (1990) Minimum cost design of reinforced concrete structures, *Computers & Structures* 1991; **41**(6), Pages 1357-1364.
- Karaboga, D., Basturk, B., On the performance of artificial bee colony algorithm, *Applied Soft Computing*; **8**(1): 687-697, 2008.
- Kennedy, J., Eberhart, R.C., Shi, Y., *Swarm Intelligence*, Series in Evolutionary Computation, Morgan Kaufmann Publishers, 2001.
- Khan, A. I., Topping, B. H. V. and Bahreininejad, A. 'Parallel training of neural networks for finite element mesh generation', *Neural Networks and Combinatorial Optimisation in Civil and Structural Engineering*, B. H. V. Topping and A. I. Khan, eds, Civil-Comp Press, pp. 81-94, 1993.
- Kirkpatrick S., Gelatt C.D., and Vecchi M.P., (1983) Optimization by simulated annealing, *Science*, Vol. 220, 671-680.
- Koskisto, O.J., Ellingwood, B.R., Reliability-based optimization of plant precast concrete structures, *Journal of Structural Engineering* 1997; **123**(3): 298-304.
- Kotinis, M., Kulkarni, A., Multi-objective shape optimization of transonic airfoil sections using swarm intelligence and surrogate models, *Structural and Multidisciplinary Optimization*, **45**(5): 747-758, 2012.
- Krawinkler, H., Zareian, F., Medina, R.A., Ibarra, L.F., Decision support for conceptual performance-based design, *Earthquake Engineering Structural Dynamics* 2006; **35**(1), 115-133.
- Krishnakumar K., 'Micro genetic algorithms for stationary and non stationary function optimization', in *SPIE Proceedings Intelligent control and adaptive systems*, 1196, (1989).
- Lagaros ND, Fragiadakis M, Papadrakakis M. Optimum design of shell structures with stiffening beams. *AIAA Journal*; 42(1): 175-184, (2004).
- Lagaros ND, Papadopoulos V. Optimum design of shell structures with random geometric, material and thickness imperfections, *International Journal of Solids and Structures*; 43(22-23): 6948-6964, (2006).
- Lagaros, N.D., Karlaftis, M.G., A critical assessment of metaheuristics for scheduling emergency infrastructure inspections, *Swarm and Evolutionary Computation*; **1**(3): 147-163, 2011.
- Lagaros, N.D., Papadrakakis, M., Applied soft computing for optimum design of structures, *Structural and Multidisciplinary Optimization*; **45**: 787-799, 2012.

- Lasdon L.S., Warren A.D., Jain A., and Ratner R., (1978) Design and testing of a generalized reduced gradient code for nonlinear programming, *ACM Trans. Math. Softw.*, 4 (1), 34-50.
- Le Riche R. G., Knopf-Lenoir C. and Haftka R. T., 'A segregated genetic algorithm for constrained structural optimization', in L. J. Eshelman (ed.), *Proceedings of the 6th International Conference on Genetic Algorithms*, 558-565, (1995).
- Le Riche, R., Haftka, R.T., On global optimization articles in SMO, *Structural and Multidisciplinary Optimization*, **46**(5): 627-629, 2012.
- Li, G., Cheng, G., Optimal decision for the target value of performance-based structural system reliability, *Structural and Multidisciplinary Optimization* 2001; **22**(4): 261-267.
- Maa C. and Shanblatt M., 'A two-phase optimization neural network', *IEEE Transactions Neural Networks*, 3(6), 1003-1009, (1992).
- Marler R.T., Arora J.S., Survey of multi-objective optimization methods for engineering, *Structural and Multidisciplinary Optimization* 2004; **26**(6): 369-395.
- Messac A, Ismail-Yahaya A. Multiobjective robust design using physical programming, *Structural Multidisciplinary Optimization*; 23: 357-371, (2002).
- Michalewicz Z., 'Genetic algorithms, numerical optimization and constraints', in L. J. Eshelman (ed.), *Proceedings of the 6th International Conference on Genetic Algorithms*, 151-158, Morgan Kaufmann, (1995).
- Michalewicz, Z., *Genetic Algorithms + Data Structures = Evolution Programs*, 3rd Edition, Springer, 2012.
- Missoum S, Ramu P, Haftka RT. A convex hull approach for the reliability-based design optimization of nonlinear transient dynamic problems. *Computer Methods in Applied Mechanics and Engineering*; 196(29-30): 2895-2906, (2007).
- Mitropoulou, Ch.Ch., Fourkiotis, Y., Lagaros, N.D., Karlaftis, M.G., Metaheuristics in structural design optimization, in *Metaheuristic Applications in Structures and Infrastructures*, Amir Hossein Gandomi, Xin-She Yang, Siamak Talatahari and Amir Hossein Alavi (Eds.), Elsevier, 79-102, 2013.
- Moharrami, H., Grierson, D.E., Computer-automated design of reinforced concrete frameworks, *Journal of Structural Engineering* 1993; **119**(7): 2036-2058.
- Moses F., 'Mathematical programming methods for structural optimization' ASME Structural Optimisation Symposium AMD Vol. 7, 35-48, 1974.
- Myung H., Kim J.-H. and Fogel D., 'Preliminary investigation into a two-stage method of evolutionary optimization on constrained problems', in J. R. McDonnell, R. G. Reynolds and D. B. Fogel (eds.), *Proceedings of the 4th Annual Conference on Evolutionary Programming*, 449-463, MIT Press, (1995).
- NAG, (1988) Software manual, NAG Ltd, Oxford, UK.
- Olhoff N., Rasmussen J., and Lund E., (1992) Method of exact numerical differentiation for error estimation in finite element based semi-analytical shape sensitivity analyses, Special Report No. 10, Institute of mechanical Engineering, Aalborg University, Aalborg, DK.
- Papadrakakis M., Lagaros N. D., Thierauf G. and Cai J., 'Advanced Solution Methods in Structural Optimization Based on Evolution Strategies', *J. Engng. Comput.*, 15(1), 12-34, (1998).
- Papadrakakis M., Lagaros N.D. and Tsompanakis Y., "Optimization of large-scale 3D trusses using evolution strategies and neural networks", *Special Issue of the International Journal of Space structures*, Vol. 14(3), pp. 211-223, 1999.
- Papadrakakis M., Lagaros N.D., and Tsompanakis Y., 'Structural optimization using evolution strategies and neural networks', *Comp. Meth. Appl. Mechanics & Engrg.*, Vol. 156, pp. 309-333, 1998.
- Papadrakakis M., Tsompanakis Y. and Lagaros N.D., Structural shape optimization using Evolution Strategies, *Engineering Optimization Journal*, Vol. 31, pp 515-540, 1999.
- Park G-J, Lee T-H, Lee KH, Hwang K-H, Robust design: An overview. *AIAA Journal*; 44(1): 181-191, (2006).
- Pedersen P., Topology optimization of three dimensional trusses, in Bendsoe M.P., and Soares C.A.M. (eds.), NATO ARW 'Topology design of structures', Sesimbra, Portugal, Kluwer Academic Publishers, Dordrecht, Netherlands, 19-31, 1993.
- Ponsich, A., Coello, C.A.C., Differential Evolution performances for the solution of mixed-integer constrained process engineering problems, *Applied Soft Computing Journal*; **11**(1): 399-409, 2011.

- Pope G.G. and Schmit L.A. (eds.) *Structural Design Applications of Mathematical Programming Techniques*, AGARDograph 149, Technical Editing and Reproduction Ltd., London, February, 1971.
- Ramm E., Bletzinger K.-U., Reitingner R., and Maute K., (1994) The challenge of structural optimization, in Topping B.H.V. and Papadrakakis M. (eds) *Advances in Structural Optimization*, CIVIL-COMP Press, Edinburgh, 27-52.
- Rechenberg I., (1973) *Evolution strategy: optimization of technical systems according to the principles of biological evolution*, Frommann-Holzboog, Stuttgart.
- Rechenberg, I., *Evolutionsstrategie – Optimierung technischer systeme nach prinzipien der biologischen Evolution*. Fromman-Holzboog, 1973.
- Rozvany G.I.N., and Zhou M., Layout and generalised shape optimization by iterative COC methods, in Rozvany G.I.N. (ed.), *NATO/DFG ASI `Optimization of large structural systems'*, Berchtesgaden, Germany, Kluwer Academic Publishers, Dordrecht, Netherlands, 103-120, 1993.
- Rummelhart D.E., and McClelland J.L., *Parallel Distributed Processing, Volume 1: Foundations*, The MIT Press, Cambridge, 1986.
- Sarma, K.C., Adeli, H., Cost optimization of concrete structures, *Journal of Structural Engineering* 1998; **124**(5): 570-578.
- Schittkowski K., Zillober C., and Zotemantel R., (1994) Numerical comparison of non-linear algorithms for structural optimization, *Structural Optimization*, 7, 1-19.
- Schwefel H.-P. and Rudolph G., 'Contemporary Evolution Strategies. in F. Morgan, A. Moreno, J. J. Merelo and P. Chacion (eds.). *Advances in Artificial Life*, Proc. Third European Conf. on Artificial Life Granada, Spain, June 4-6, Springer, Berlin, 893-907, (1995).
- Schwefel H.P., (1981) *Numerical optimization for computer models*, Wiley & Sons, Chichester, UK.
- Sheu C.Y. and Prager W., 'Recent development in optimal structural design', *Applied Mechanical Reviews*, 21(10), 985-992, 1968.
- Shieh R.C., 'Massively parallel structural design using stochastic optimization and mixed neural net/finite element analysis methods', *Computing Systems in Engineering*, Vol. 5, No. 4-6, pp. 455-467, 1994.
- Sir. Thomson, W., On the division of space with minimum partitional area. *Acta Mathematica*, 11(1-4): 121-134, 1887.
- Spunt L., *Optimum Structural Design*, Prentice-Hall, Englewood Cliffs, New Jersey, 41-42, 1971.
- Stephens, J.E., and VanLuchene, D., 'Integrated assessment of seismic damage in structures', *Microcomputers in Civil Engineering*, Vol. 9(2), pp. 119-128, 1994.
- Storn, R.M., Price, K.V., Differential evolution - a simple and efficient heuristic for global optimization over continuous spaces. *Journal of Global Optimization*; **11**: 341-359, 1997.
- Suzuki K., and Kikuchi N., Layout optimization using the homogenization method, in Rozvany G.I.N. (ed.), *NATO/DFG ASI `Optimization of large structural systems'*, Berchtesgaden, Germany, Kluwer Academic Publishers, Dordrecht, Netherlands, 157-175, 1993.
- Svanberg K. (1987) The method of moving asymptotes, a new method for structural optimization, *Int. J. Num. Meth. Eng.*, 23, 359-373.
- Taylor C.A., EQSIM, A program for generating spectrum compatible earthquake ground acceleration time histories, Reference Manual, Bristol Earthquake Engineering Data Acquisition and Processing System, December 1989.
- Thanedar P.B., Arora J.S., Tseng C.H., Lim O.K., and Park G.J., (1986) Performance of some SQP methods on structural optimization problems, *Inter. Journal of Num. Meth. Engng*, 23, 2187-2203.
- Thierauf G., and Cai J., (1995) A two level parallel evolution strategy for solving mixed-discrete structural optimization problems. The 21th ASME Design Automation Conference, Boston MA, September 17-221.
- Thierauf G., and Cai J., (1996) Structural optimization based on Evolution Strategy, in Papadrakakis M. and Bugeda G. (eds), *Advanced computational methods in structural mechanics*, CIMNE, Barcelona, 266-280.
- Topping, B.H.V. and Bahreininejad, A. *Neural computing for structural mechanics*, Saxe Coburg, Uk. 1997.
- Tsombanakis Y., Lagaros N.D., Papadrakakis M., *Structural Optimization Considering Uncertainties*, Taylor & Francis, 2007.

- Van Keulen F. and Hinton E., Topology design of plate and shell structures using the hard kill method, in B.H.V. Topping (ed.) *Advances in optimization for Structural Engineering*, CIVIL-COMP Press, Edinburgh, 177-188, 1996.
- van Laarhoven, P.J., Aarts, E.H., *Simulated Annealing: Theory and Applications (Mathematics and Its Applications)*, Kluwer Academic Publishers, 2010.
- Vanderplaats G.N., *Numerical optimisation techniques for engineering design: with applications*, McGraw-Hill, New York, 1984.
- Venkayya V.B., Khot N.S. and Berke L., 'Application of optimality criteria approaches to automated design of large practical structures', 2nd Symposium on Structural Optimisation AGARD-CP-123, Milan, Italy, April, 1973.
- Waagen D., Diercks P. and McDonnell J., 'The stochastic direction set algorithm: A hybrid technique for finding function extrema', in D. B. Fogel and W. Atmar (eds.), *Proceedings of the 1st Annual Conference on Evolutionary Programming*, pp. 35-42, Evolutionary Programming Society, (1992).
- Weaire, D., Phelan, R., A counter-example to Kelvin's conjecture on minimal surfaces, *Philosophical Magazine Letters*; **69**: 107-110, 1994.
- Xie Y.M., and Steven G.P., A simple evolutionary procedure for structural optimization, *Computers & Structures*, Vol. 49, 885-896, 1993.
- Xie Y.M., and Steven G.P., Optimal design of multiple load case structures using an evolutionary procedure, *Journal of Engineering Computations*, Vol. 11, 295-302, 1994.
- Yang, X.-S., Metaheuristic optimization: algorithm analysis and open problems, *Experimental Algorithms, Lecture Notes in Computer Science*, **6630**: 21-32, 2011.
- Yang, X.S., *Nature-Inspired Metaheuristic Algorithms*. Frome: Luniver Press, 2008.
- Zitzler E, Laumanns M, Thiele L. SPEA 2: improving the strength Pareto evolutionary algorithm. In: Giannakoglou K, Tsahalis D, Periaux J, Papailou P, Fogarty T. (Eds.), EUROGEN 2001, evolutionary methods for design, optimization and control with applications to industrial problems, Athens, Greece, 2001, pp. 95-100.

Σχόλια - Προβλήματα - Παρατηρήσεις

Δεν υπήρχαν παρατηρήσεις

	Επιστημονικός Υπεύθυνος Έργου	Συντονιστής Έργου
Υπογραφή:		
Όνοματεπώνυμο :	N. ΛΑΓΑΡΟΣ	Χ. ΚΩΣΤΟΠΑΝΑΓΙΩΤΗΣ
Ημ/νία :	31/01/2020	31/01/2020