ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΜΗΧΑΝΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ
ΤΟΜΕΑΣ ΡΕΥΣΤΩΝ
ΕΡΓΑΣΤΗΡΙΟ ΘΕΡΜΙΚΩΝ ΣΤΡΟΒΙΛΟΜΗΧΑΝΩΝ

ΜΕΛΕΤΕΣ ΒΕΛΤΙΣΤΟΠΟΙΗΣΗΣ ΤΟΥ ΣΥΣΤΗΜΑΤΟΣ ΠΡΟΩΣΗΣ ΑΕΡΟΣΚΑΦΩΝ

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

ΓΕΡΑΣΙΜΟΣ ΓΙΑΧΟΣ

Επίβλεψη:
Επ. Καθηγητής Ν. Αρετάκης

ΑΘΗΝΑ
ΦΕΒΡΟΥΑΡΙΟΣ 2019

ΠΕΡΙΛΗΨΗ

Αντικείμενο της παρούσας εργασίας είναι η εύρεση της πιο κατάλληλης τεχνικής ή μεθοδολογίας για την βελτιστοποίηση του προωστήριου συστήματος αεροσκάφους με σκοπό την μείωση της κατανάλωσης καυσίμου κατά την αποστολή του. Για το σκοπό αυτό γίνεται χρήση του εργαλείου βελτιστοποίησης Isight.

Αρχικά αναλύεται η αρχή λειτουργίας του Isight και γίνεται περιγραφή των τεχνικών βελτιστοποίησης που περιλαμβάνει. Ύστερα από εκτενή βιβλιογραφική ανασκόπηση, σχετικά με τις μεθόδους που προτείνονται για την προσέγγιση τέτοιου είδους προβλημάτων και λαμβάνοντας υπόψη τις τεχνικές που περιλαμβάνονται στο λογισμικό, επιλέχθηκαν κάποιες από αυτές ως οι πιο ιδανικές. Οι τεχνικές που επιλέχθηκαν είναι δύο από την κατηγορία Design of Experiments, η Optimal Latin Hypercube και η Parameter Study, τέσσερις αλγόριθμοι βελτιστοποίησης, ο ASA, ο Downhill Simplex, ο PSO και ο MIGA, ενώ επιλέχθηκαν και δύο τεχνικές δημιουργίας μεταπροτύπων η Kriging και τα RBF.

Στην συνέχεια γίνεται εφαρμογή των παραπάνω μεθόδων και σύγκριση των αποτελεσμάτων τους σε μια απλή περίπτωση για την βελτιστοποίηση του κινητήρα CFM56, που έχει μοντελοποιηθεί στο PROOSIS από το ΕΘΣ, με στόχο την μείωση της ειδικής κατανάλωσης καυσίμου του κατά την κρουαζιέρα. Σκοπός της εφαρμογής αυτής ήταν η εξοικείωση με το λογισμικό και η προσπάθεια ερμηνείας των αποτελεσμάτων των μεθόδων σε μια απλή περίπτωση.

Στη συνέχεια γίνεται εφαρμογή των ίδιων μεθόδων σε μια πιο πολύπλοκη περίπτωση για την βελτιστοποίηση του προωστήριου συστήματος ενός αεροσκάφους με στόχο την μείωση της κατανάλωσης καυσίμου σε ολόκληρη την αποστολή. Για το σκοπό αυτό χρησιμοποιείται το λογισμικό αξιολόγησης TERA που περιλαμβάνει μοντέλο του κινητήρα, μοντέλο αεροσκάφους, υπολογισμό βάρους αεροσκάφους - κινητήρα, υπολογισμό ήχου και ρύπων καθώς και υπολογισμό χρόνου μεταξύ διαστημάτων συντήρησης κινητήρα (TBO). Εκτός από τις παραπάνω μεθόδους, στην περίπτωση αυτή, έγινε επίσης χρήση του λογισμικού βελτιστοποίησης EASY που έχει αναπτυχθεί από το ΕΘΣ. Τέλος αναλύονται τα αποτελέσματα της βελτιστοποίησης και συγκρίνονται οι τεχνικές με βάση τα αποτελέσματα τους και το χρόνο σύγκλισής τους.

Από τα αποτελέσματα των παραπάνω εφαρμογών προέκυψε ότι οι αλγόριθμοι βελτιστοποίησης PSO και ASA παρουσιάζουν την καλύτερη συμπεριφορά, ενώ το εργαλείο βελτιστοποίησης EASY επιτυγχάνει πολύ καλά αποτελέσματα με πολύ μικρότερο αριθμό αξιολογήσεων.

Ακρωνύμια

ΕΜΠ	Εθνικό Μετσόβιο Πολυτεχνείο
ΕΘΣ	Εργαστήριο Θερμικών Στροβιλομηχανών

DOE	Design of Experiments
TERA	Techno-economic Environmental and Risk Assessment
PSO	Particle Swarm Optimizer
ASA	Adaptive Simulated Annealing
HPC	High Pressure Compressor
LPC	Low Pressure Compressor
FPR	Fan Pressure Ratio
BPR	Bypass Ratio
GTCRSR	Geared Turbofan with Counter-Rotating Core for Short Range Application

ΠΕΡΙΕΧΟΜΕΝΑ

1 ΕΙΣΑΓΩΓΗ 1.1

1.1 Σκοπός και δομή της εργασίας 1.4

2 Γενική Περιγραφή του Λογισμικού Isight 2.1

2.1 Περιγραφή 2.1

2.2 Αρχή λειτουργίας του προγράμματος 2.2

2.3 Προσομοίωση της ροής της διαδικασίας 2.5

2.4 Χαρακτηριστικά των components 2.7

2.5 Εφαρμογές του Isight σε πολυτομεακά προβλήματα βελτιστοποίησης 2.9

3 Τεχνικές που διαθέτει το Isight 3.1

3.1 Design of Experiments 3.1
3.1.1 Full-Factorial Design Technique 3.1
3.1.2 Fractional Factorial Design Technique 3.2
3.1.3 Parameter Study Technique 3.2
3.1.4 Central Composite Design Technique 3.2
3.1.5 Box-Behnken Design 3.3
3.1.6 Orthogonal Arrays 3.4
3.1.7 Latin Hypercube 3.5
3.1.8 Optimal Latin Hypercube 3.7

3.2 Optimization Algorithms 3.9
3.2.1 Adaptive Simulated Annealing (ASA) 3.10
3.2.2 Downhill Simplex 3.11
3.2.3 Hooke-Jeeves 3.13
3.2.4 Large Scale Generalized Reduced Gradient (LSGRG) 3.15
3.2.5 MISQP - Mixed Integer Sequential Quadratic Programming 3.15
3.2.6 Multi-Objective Particle Swarm Technique 3.15
3.2.7 Multi-Island Genetic Algorithm 3.16
3.2.8 Non-dominated Sorting Genetic Algorithm (NSGA-II) 3.17
3.2.9 Neighborhood Cultivation Genetic Algorithm (NCGA) 3.18
3.2.10 Evolutionary Optimization Algorithm (EVOL) 3.19
3.2.11 Archive based Micro Genetic Algorithm (AMGA) 3.20
3.2.12 Sequential Quadratic Programming (NLPQL) 3.21

3.2.13 Multifunction Optimization System Tool (MOST) 3.21
3.2.14 Pointer 3.21
3.2.15 Modified Method of Feasible Directions (MMFD) 3.22

3.3 Approximations 3.22
3.3.1 Kriging Model 3.23
3.3.2 RBF and EBF Models 3.24
3.3.3 Response Surface Model 3.25

3.4 Τεχνικές που προτείνονται για την βελτιστοποίηση του προβλήματος 3.25

4 Εφαρμογή στον κινητήρα CFM56 ..... 4.1

4.1 Ο αεροπορικός κινητήρας CFM56 ..... 4.1

4.2 Ενσωμάτωση στο Isight 4.4

4.3 Εφαρμογή των τεχνικών για την βελτιστοποίηση του προβλήματος 4.5
4.3.1 Δημιουργία DOE με την χρήση της Optimal Latin Hypercube τεχνικής 4.5
4.3.2 Ο αλγόριθμος βελτιστοποίησης Adaptive Simulated Annealing 4.7
4.3.3 Ο αλγόριθμος βελτιστοποίησης Downhill Simplex 4.10
4.3.4 Ο αλγόριθμος βελτιστοποίησης Multi-Island Genetic Algorithm 4.11
4.3.5 Ο αλγόριθμος βελτιστοποίησης Particle Swarm Optimizer 4.13
4.3.6 Χρήση μεταπρότυπων Kriging και RBF 4.14

4.4 Σύγκριση και Ανάλυση των αποτελεσμάτων 4.22

5 Εφαρμογή κατά την αποστολή του αεροσκάφους 5.1

5.1 Το εργαλείο TERA και η διάταξη GTCRSR 5.1

5.2 Εφαρμογή των τεχνικών βελτιστοποίησης 5.5
5.2.1 Optimal Latin Hypercube και Sensitivity Analysis 5.6

5.3 Αλγόριθμοι Βελτιστοποίησης 5.9
5.3.1 Adaptive Simulated Annealing 5.9
5.3.2 Downhill Simplex 5.12
5.3.3 Multi-Island Genetic Algorithm 5.13
5.3.4 Particle Swarm Optimizer 5.15

5.4 EASY 5.17

5.5 Χρήση των Kriging και RBF μεταπροτύπων 5.19

5.6 Σύγκριση και Ανάλυση των αποτελεσμάτων.....5.30

6 Ανακεφαλαίωση-Συμπεράσματα-Προτάσεις 6.1

6.1 Ανακεφαλαίωση.....6.1

6.2 Συμπεράσματα 6.2

7 Βιβλιογραφία 6.1

ΠΡΟΛΟΓΟΣ

Από τη θέση αυτή θα ήθελα να ευχαριστήσω τον επιβλέποντα καθηγητή μου κ. Αρετάκη Νίκο, που μου ανέθεσε ένα τόσο ενδιαφέρον θέμα και η συμβολή του οποίου ήταν καθοριστική για την εκπόνηση της παρούσας εργασίας. Επίσης, θέλω να ευχαριστήσω θερμά τον κ. Αλεξίου Αλέξη του οποίου οι συμβουλές και η καθοδήγηση ήταν πολύτιμη σε όλη τη διαδικασία της μελέτης. Επιπλέον θέλω να ευχαριστήσω τον καθηγητή κ. Γιαννάκογλου Κυριάκο για τις πολύτιμες συμβουλές του και τον Καψούλη Δημήτρη για ον χρόνο που αφιέρωσε για την χρήση του EASY.

Στους γονείς μου στους οποίους δεν θα ήμουν εδώ. Στους φίλους μου, ιδιαίτερα στον Ηλία, που ήταν σε κάθε βήμα παρόντες.

Στον Ιβάν.

1 ΕΙΣΑΓΩΓΗ

Οι τρέχουσες τάσεις στην πολιτική αεροπορία υπαγορεύουν συνεχώς αυξανόμενες απαιτήσεις για τη μείωση της κατανάλωσης καυσίμου και των περιβαλλοντικών επιπτώσεων που έχουν τα παραγόμενα καυσαέρια. Το Συμβούλιο για την Αεροναυτική Έρευνα στην Ευρώπη (ACARE) έχει θέσει ορισμένους περιβαλλοντικούς στόχους [1] που πρέπει να επιτευχθούν μέχρι το 2020. Στους στόχους αυτούς περιλαμβάνονται η μείωση των εκπομπών διοξειδίου του άνθρακα (CO_2) και οξειδίων του αζώτου (NOx) της τάξης του 50% και 80% αντίστοιχα. Για αυτόν τον λόγο, προκειμένου να διαχειριστούν αποτελεσματικά οι μακροπρόθεσμες επιπτώσεις της πολιτικής αεροπορίας στο περιβάλλον, λαμβάνοντας παράλληλα υπόψη την αναμενόμενη αύξηση της εναέριας κυκλοφορίας, επιβάλλεται η αύξηση της απόδοσης των αεροπορικών κινητήρων.

Για να επιτευχθούν οι παραπάνω στόχοι χρειάζονται να πραγματοποιηθούν μελέτες βελτιστοποίησης του προωστήριου συστήματος αεροσκάφους, με χρήση υπολογιστικών μοντέλων, λαμβάνοντας υπόψη διάφορα κριτήρια και περιορισμούς, δημιουργώντας έτσι ένα πολυτομεακό πρόβλημα βελτιστοποίησης.

Υπάρχουν διάφορα μοντέλα προσομοίωσης της λειτουργίας ενός αεροπορικού κινητήρα και των επιμέρους συνιστωσών που αποτελούν το πολυτομεακό πρόβλημα βελτιστοποίησης. Συγκεκριμένα, η μοντελοποίηση του κινητήρα στην βιομηχανία γίνεται με την χρήση κατάλληλων λογισμικών που έχουν αναπτύξει εσωτερικά οι εταιρείες ή και με ευρέως διαδεδομένα εμπορικά λογισμικά, όπως Proosis, NPSS, GasTurb, Turbomatch, Matlab/Simulink, και GSP. Για τις υπόλοιπες συνιστώσες όπως μοντέλο αεροσκάφους, υπολογισμό βάρους, ήχου, ρύπων και χρόνου ζωής εξαρτημάτων κινητήρα έχουν αναπτυχθεί πιο εξειδικευμένα λογισμικά τα οποία δεν είναι ευρέως διαδεδομένα.

Η βελτιστοποίηση ενός προωστήριου συστήματος περιλαμβάνει τη βελτιστοποίηση διαφόρων παραμέτρων του, μεταξύ των οποίων είναι διάφορες αεροθερμοδυναμικές παράμετροι, το βάρος του, η αεροδυναμική σχεδίαση του, οι εκπομπές ρύπων και το επίπεδο θορύβου του. Επιπλέον υπάρχουν και άλλες παράμετροι που επηρεάζουν το συνολικό κόστος του κινητήρα, όπως το κόστος κατασκευής και συντήρησης και η διάρκεια ζωής των εξαρτημάτων του.

Στη βιβλιογραφία παρουσιάζονται διάφορες μελέτες του πολυτομεακού προβλήματος βελτιστοποίησης, για τη μείωση της συνολικής κατανάλωσης καυσίμου με την ύπαρξη περιορισμών. Πιο συγκεκριμένα, οι Celis et al. [2] έδειξαν την δυνατότητα των μεθόδων καθολικής βελτιστοποίησης, για τη βελτιστοποίηση της τροχιάς του αεροσκάφους, για μείωση της κατανάλωσης καυσίμου και των εκπεμπόμενων ρύπων. Η μελέτη τους επικεντρώθηκε σε δισδιάστατο προφίλ ανόδου ενός γενικού, δικινητήριου,

μεγάλου αεροσκάφους μεταφοράς. Η μεθοδολογία τους περιελάμβανε τη χρήση μοντέλων σταθερής κατάστασης (steady-state) για τις επιδόσεις του αεροσκάφους και των κινητήρων και γενετικούς αλγόριθμους. Τα αποτελέσματα της βελτιστοποίησης έδειξαν μια εφικτή μείωση στην κατανάλωση καυσίμου και παραγωγή CO_2 της τάξης του 6.7% σε σχέση με μια μη-βέλτιστη γενική τροχιά.

Οι Goulos et al. [3] επικεντρώθηκαν στην εφαρμογή μια υβριδικής στρατηγικής βελτιστοποίησης στην μελέτη που περιγράφεται στην [2] χρησιμοποιώντας ίδια μεθοδολογία μοντελοποίησης. Εφάρμοσαν έναν γενετικό αλγόριθμο για να εξερευνήσει όλο τον χώρο σχεδιασμού και ύστερα ολοκλήρωσαν την διαδικασία βελτιστοποίησης με μια τοπική μέθοδο κλίσης (local-gradient). Έτσι προέκυψαν βελτιωμένες τροχιές πτήσεων που χρειάζονται 8% λιγότερο καύσιμο σε σχέση με την μελέτη που έγινε προηγουμένως [2].

Μια διαφορετική προσέγγιση, σε ένα τόσο πολύπλοκο πρόβλημα, είναι η δημιουργία μεταπροτύπων τα οποία, έχουν τη δυνατότητα να μειώσουν τον υπολογιστικό χρόνο για την βελτιστοποίηση του προβλήματος, δίνοντας παράλληλα ικανοποιητικά αποτελέσματα. Οι Kestner et al. [4] μελέτησαν την αποτελεσματικότητα των μεταπροτύπων και συνέκριναν τα αποτελέσματα της βελτιστοποίησης με και χωρίς την χρήση μεταπροτύπων. Οι ερευνητές χρησιμοποίησαν μια DOE τεχνική την Latin Hypercube με 12000 εκτελέσεις. Για την δημιουργία μεταπροτύπου χρησιμοποιήθηκαν τεχνητά νευρωνικά δίκτυα (ANN) ενώ ο χώρος σχεδιασμού δημιουργήθηκε ύστερα από 300 εκτελέσεις. Τα αποτελέσματα που προέκυψαν ήταν παρόμοια με αυτά χωρίς την χρήση μεταπροτύπων, κάτι που δείχνει ότι η χρήση τους είναι βιώσιμη και προτιμητέα για την αντιμετώπιση μεγάλων και περίπλοκων προβλημάτων.

Οι D'Ippolito et al. [5] εργάστηκαν στην βελτιστοποίηση μιας μανούβρας απογείωσης Κατηγορίας Α ενός ελικοπτέρου για την επίδραση της κατανάλωσης και του ήχου. Η στρατηγική βελτιστοποίησης περιελάμβανε την χρήση μιας μεθόδου DOE σε συνδυασμό με μοντέλα απόκρισης επιφάνειας (Response Surface Models) ή (RSMs). Προέκυψε μια τροχιά που ελαχιστοποιούσε την κατανάλωση καυσίμου, CO_2, NOx, και την επίδραση του θορύβου χρησιμοποιώντας και αυτοί μια υβριδική τεχνική βελτιστοποίησης όπως στις [3], [6].

Οι Fakhre Ali et al. [7] εργάστηκαν στην ελαχιστοποίηση της συνολικής κατανάλωσης καυσίμου και των ρύπων, κατά την διάρκεια της αποστολής ενός αεροσκάφους. Ο χώρος σχεδιασμού που αντιστοιχεί στις θερμοδυναμικές παραμέτρους, καθώς επίσης το βάρος της μηχανής, η κατανάλωση καυσίμου και οι ρύποι μελετήθηκαν με την χρήση της τεχνικής DOE, Latin Hypercube και στην συνέχεια δημιουργήθηκαν μεταπρότυπα RSMs με την χρήση της τεχνικής Kriging. Με την δημιουργία μεταπροτύπων έγινε η βελτιστοποίηση του προβλήματος με την χρήση ενός πολυκριτηριακού αλγόριθμου βελτιστοποίησης τον Particle Swarm Optimizer για να παραχθεί το μέτωπο Pareto, έτσι ώστε να βρεθεί η βέλτιστη αλληλεξάρτηση μεταξύ της κατανάλωσης καυσίμου στην αποστολή και των εκπομπών NOx. Τα αποτελέσματα ήταν η μείωση του καυσίμου κατά 27.02% και οι εκπομπές NOx κατά 7.02%.

Ειδικότερα οι Kyprianidis et al. [8]–[11] εργάστηκαν στην βελτιστοποίηση του αεροπορικού κινητήρα για την συνολική μείωση καυσίμου, κατά την αποστολή του αεροσκάφους. Στην μελέτη [9] διερεύνησαν την αλληλεπίδραση των κριτηρίων σχεδιασμού, ενός αεροπορικού κινητήρα με κιβώτιο ταχυτήτων (Geared turbofan) στον οποίο υπάρχει και ένας ενδιάμεσος ψύκτης (intercooler), έτσι ώστε να προκύψει σημαντική γνώση σχεδιασμού στο επίπεδο του συστήματος κινητήρα-αεροσκάφος. Προηγούμενοι ερευνητές διαφοροποιούσαν τις παραμέτρους μια την φορά κατά την βελτιστοποίηση ενώ σε αυτήν την μελέτη διαφοροποιούνται όλες ταυτόχρονα, για την επίτευξη της ελαχιστοποίησης της συνολικής κατανάλωσης καυσίμου κατά την διάρκεια της αποστολής του αεροσκάφους, με την ύπαρξη περιορισμών. Επειδή το πρόβλημα είναι μη-γραμμικό και ασυνεχές επιλέχθηκε ο αλγόριθμος βελτιστοποίησης Adaptive Simulated Annealing (ASA) και στην συνέχεια οι λύσεις που βρέθηκαν, εξετάστηκαν με την τεχνική Διαδοχικού Τετραγωνικού Προγραμματισμού (Sequential Quadratic Programming). Κατά την βελτιστοποίηση, η ASA επανεκλήθηκε από διάφορα σημεία στον χώρο σχεδιασμού για την αποφυγή της εγκλώβισης του αλγόριθμου σε τοπικό ακρότατο. Η βελτιστοποίηση συμπληρώθηκε από την δημιουργία ενός μεγάλου χώρου σχεδιασμού (DOE) αρκετών χιλιάδων σημείων και η τεχνική που επιλέχθηκε ήταν η Latin Hypercube. Η Latin Hypercube είχε τροποποιηθεί έτσι ώστε ο συνδυασμός των επιπέδων των παραγόντων για κάθε παράμετρο να έχει βελτιστοποιηθεί, παρά να έχει επιλεχθεί τυχαία. Τα αποτελέσματα του DOE επαλήθευσαν κατά προσέγγιση την θέση του βέλτιστου σχεδιασμού όπως πρότεινε ο αλγόριθμος βελτιστοποίησης. Τα αποτελέσματα της μελέτης έδειξαν σημαντική μείωση της κατανάλωσης με κάποιες υποθέσεις.

Σε άλλη πολυτομεακή εφαρμογή [12] σύγκριναν έναν geared open rotor turbofan με έναν ultra-high bypass ratio geared turbofan κινητήρα. Σκοπός της εφαρμογής ήταν, η σύγκριση των δύο κινητήρων όσον αφορά την κατανάλωση καυσίμου και ρύπων κατά την αποστολή του αεροσκάφους με την επιλογή κατάλληλης ειδικής ώσης. Σχεδιαστικοί περιορισμοί, όσον αφορά τις απαιτήσεις του κατασκευαστή, λήφθηκαν υπόψιν κάνοντας το πολυτομεακό πρόβλημα, πολυκριτηριακό με περιορισμούς. Για την μοντελοποίηση των κινητήρων χρησιμοποιήθηκε ο κώδικας EVA [10], ενώ για την μοντελοποίηση των διαστάσεων και του βάρους των συνιστωσών του κάθε κινητήρα ο Weico. Στη συνέχεια, τα συγκεκριμένα εργαλεία ενσωματώθηκαν σε ένα υπολογιστικό περιβάλλον το Isight [13]. Το λογισμικό Isight δίνει την δυνατότητα της ενσωμάτωσης των διάφορων συνιστωσών του πολυτομεακού προβλήματος και παρέχει κατάλληλες τεχνικές για την βελτιστοποίηση του. Οι ερευνητές αφού εκτέλεσαν πολυκριτηριακή βελτιστοποίηση, μελέτες σχεδιασμού και ανάλυση ευαισθησίας κατέληξαν στο ότι ο open rotor κινητήρας έχει κατά περίπου 15% λιγότερη συνολική κατανάλωση από τον ultra-high bypass ratio geared turbofan.

Σε μια άλλη μελέτη τους [8] ερεύνησαν τον χώρο σχεδιασμού ενός recuperated αεροπορικού κινητήρα με ενδιάμεσο ψύκτη, χρησιμοποιώντας λεπτομερή μοντέλα προσομοίωσης για τις επιδόσεις του κινητήρα και του αεροσκάφους, το βάρος και τις

διαστάσεις του. Αφού αναγνωρίστηκαν οι σημαντικότερες παράμετροι που επηρεάζουν την συνολική κατανάλωση καυσίμου, δημιουργήθηκε ένας χώρος σχεδιασμού των πειραμάτων για μια πρόχειρη κατανόηση του χώρου σχεδιασμού. Στη συνέχεια, έγινε η βελτιστοποίηση με διάφορους αλγόριθμους για την εύρεση του ιδανικού σχεδιασμού. Τέλος, διεξήχθη μια παραμετρική μελέτη στο βέλτιστο σημείο σχεδιασμού για την καλύτερη κατανόηση των αποτελεσμάτων. Οι διάφορες συνιστώσες του προβλήματος μοντελοποιήθηκαν στα εργαλεία TERA [14] και EVA [10], ενώ στην συνέχεια ενσωματώθηκαν στο λογισμικό Isight όπου εφαρμόστηκαν οι διάφορες τεχνικές βελτιστοποίησης. Τα αποτελέσματα, έδειξαν περίπου 5,56% μείωση της συνολικής κατανάλωσης καυσίμου.

Όπως φαίνεται και από τις προηγούμενες μελέτες υπάρχουν διάφορες προσεγγίσεις για την αντιμετώπιση του προβλήματος ωστόσο παρουσιάζεται ένα κενό στην γνώση για το ποια τεχνική είναι η ιδανικότερη. Οι ερευνητές χρησιμοποιούν τεχνικές οι οποίες εμπειρικά παράγουν ικανοποιητικά αποτελέσματα, ωστόσο μπορεί αυτές να μην είναι οι βέλτιστες, δηλαδή να μη δίνουν ικανοποιητικά αποτελέσματα με το λιγότερο υπολογιστικό κόστος. Επομένως, η εύρεση της αποδοτικότερης τεχνικής βελτιστοποίησης ενός τόσο περίπλοκου προβλήματος είναι καίριας σημασίας για την εξοικονόμηση υπολογιστικού κόστους.

1.1 Σκοπός και δομή της εργασίας

Ο σκοπός της εργασίας είναι η εύρεση της πιο κατάλληλης τεχνικής βελτιστοποίησης σε πολυτομεακά προβλήματα βελτιστοποίησης προωστήριου συστήματος αεροσκαφών. Για το σκοπό αυτό θα χρησιμοποιηθεί το λογισμικό Isight στο οποίο είναι εύκολη η ανάλυση ενός τέτοιου προβλήματος, παρέχοντας ένα πλήθος τεχνικών, οι οποίες προσεγγίζουν το πρόβλημα από διαφορετικές γωνίες, δίνοντας τη δυνατότητα εύρεσης της βέλτιστης τεχνικής για την βελτιστοποίηση του. Η αξιολόγηση των διαφόρων συνιστωσών του προβλήματος πραγματοποιείται από κατάλληλα υπολογιστικά μοντέλα τα οποία έχουν αναπτυχθεί από το ΕΘΣ σε συνεργασία με άλλα ευρωπαϊκά πανεπιστήμια [15].

Πιο συγκεκριμένα στα πλαίσια της εργασίας αυτής πραγματοποιούνται και αναλύονται δύο εφαρμογές, η πρώτη περιλαμβάνει την βελτιστοποίηση του κινητήρα CFM56 και η δεύτερη την βελτιστοποίηση ενός concept κινητήρα του Geared Turbofan with Counter-Rotating Core for short range application (GTCRC). Η εφαρμογή για τον CFM56 τροποποιεί τις αεροθερμοδυναμικές παραμέτρους του κινητήρα με στόχο την ελαχιστοποίηση της ειδικής κατανάλωσης του κινητήρα κατά την κρουαζιέρα. Στην δεύτερη εφαρμογή ο στόχος είναι η μείωση της συνολικής κατανάλωσης καυσίμου της αποστολής του αεροσκάφους, λαμβάνοντας υπόψη τις αλληλεπιδράσεις των διαφορετικών συνιστωσών του προβλήματος.

Παράλληλα με το λογισμικό Isight γίνεται χρήση και ενός επιπλέον λογισμικού βελτιστοποίησης που έχει αναπτυχθεί από την Μονάδα Παράλληλης Υπολογιστικής Ρευστομηχανικής & Βελτιστοποίησης του ΕΘΣ [16], το EASY, με στόχο την αξιολόγηση της καταλληλότητας του για τέτοιου είδους προβλήματα.

Η δομή της εργασίας έχει ως εξής:

- Στο Κεφάλαιο 2 περιγράφεται το λογισμικό βελτιστοποίησης Isight.
- Στο Κεφάλαιο 3 αναπτύσσονται τα βασικά χαρακτηριστικά των τεχνικών για βελτιστοποίηση, που διαθέτει το Isight και προτείνονται κάποιες από αυτές.
- Στο Κεφάλαιο 4 εφαρμόζονται οι τεχνικές που προτάθηκαν στο μοντέλο του κινητήρα CFM56 και αναλύονται τα αποτελέσματα που προκύπτουν.
- Στο Κεφάλαιο 5 εφαρμόζονται οι ίδιες τεχνικές για τη βελτιστοποίηση του κινητήρα GTCRC. Επίσης χρησιμοποιείται και το εργαλείο βελτιστοποίησης EASY. Τέλος, αναλύονται τα αποτελέσματα της εφαρμογής.
- Στο Κεφάλαιο 6 γίνεται η σύνοψη της εργασίας και εξάγονται συμπεράσματα σχετικά με τις τεχνικές που χρησιμοποιήθηκαν.

2 Γενική Περιγραφή του Λογισμικού Isight

Στο κεφάλαιο αυτό γίνεται η περιγραφή του προγράμματος βελτιστοποίησης Isight, το οποίο χρησιμοποιήθηκε στα πλαίσια της εργασίας. Επιπλέον γίνεται αναφορά στις τεχνικές που περιλαμβάνει και παρουσιάζονται παραδείγματα χρήσης του προγράμματος στην βιομηχανία.

2.1 Περιγραφή

Για την ανάπτυξη νέων προϊόντων, σχεδιαστές και μηχανικοί χρησιμοποιούν ένα μεγάλο εύρος προγραμμάτων για την σχεδίαση και προσομοίωση τους. Πολλές φορές, απαιτείται η χρήση δύο ή παραπάνω λογισμικών για την δημιουργία των προϊόντων. Συχνά, τα αποτελέσματα ενός λογισμικού απαιτούνται ως είσοδοι στα άλλα προγράμματα με τον χρήστη να είναι υποχρεωμένος να τα εισάγει χειροκίνητα, κάνοντας έτσι την διαδικασία της ανάπτυξης προϊόντων, χρονοβόρα και μη αποδοτική.

Το Isight παρέχει σε σχεδιαστές, μηχανικούς και ερευνητές ένα ανοιχτό σύστημα για την ενσωμάτωση μοντέλων σχεδιασμού και προσομοίωσης, που δημιουργούνται με διάφορα προγράμματα CAD, CFD, Matlab και πολλά άλλα, για την αυτοματοποίηση της εκτέλεσης χιλιάδων προσομοιώσεων. Το πρόγραμμα επιτρέπει στον χρήστη να εξοικονομήσει χρόνο και να βελτιώσει τα προϊόντα του βελτιστοποιώντας τα, με μετρήσεις απόδοσης ή κόστους μέσω στατιστικών μεθόδων όπως Design of Experiments (DOE) και την χρήση διάφορων αλγόριθμων βελτιστοποίησης. Ο ακρογωνιαίος λίθος του λογισμικού Isight είναι το αρχείο περιγραφής που περιέχει όλες τις πληροφορίες που σχετίζονται με ένα συγκεκριμένο πρόβλημα σχεδίασης. Η βασική συντακτική μονάδα του αρχείου περιγραφής Isight ονομάζεται εργασία (task) στην οποία ορίζονται όλες οι παράμετροι, η ανάλυση, η εκτέλεση του προγράμματος και οι επιλογές βελτιστοποίησης. Μια task μπορεί να περιέχει έναν απεριόριστο αριθμό επιμέρους εργασιών στην οποία η task ενεργεί ως συντονιστής. Η εύκολη χρήση του λογισμικού συμβάλλει στον συντονισμό του αυτοματισμού, μέσω της δημιουργίας ροών προσομοίωσης που επιτρέπουν την εκτέλεση των προσομοιώσεων χωρίς παρέμβαση από τον χρήστη. Στη συνέχεια, ενσωματώνοντας τα αποτελέσματα των πολυτομεακών αναλύσεων, εξετάζονται ταυτόχρονα από διάφορες μηχανικές πλευρές, του εκάστοτε προβλήματος, όπως η αεροδυναμική και η μηχανική καταπόνηση, η θερμοδυναμική και το κόστος ή το βάρος. Η μελέτη των διάφορων τομέων γίνεται με την χρήση διάφορων μεθόδων όπως Design of Experiments, χρήση αλγορίθμων βελτιστοποίησης και δημιουργία μεταπροτύπων, για την βελτιστοποίηση όλου του προβλήματος. Τέλος, τα

αποτελέσματα μπορούν να παρουσιαστούν σε διάφορες μορφές όπως πίνακες και 3Δ γραφήματα για την καλύτερη κατανόηση τους.

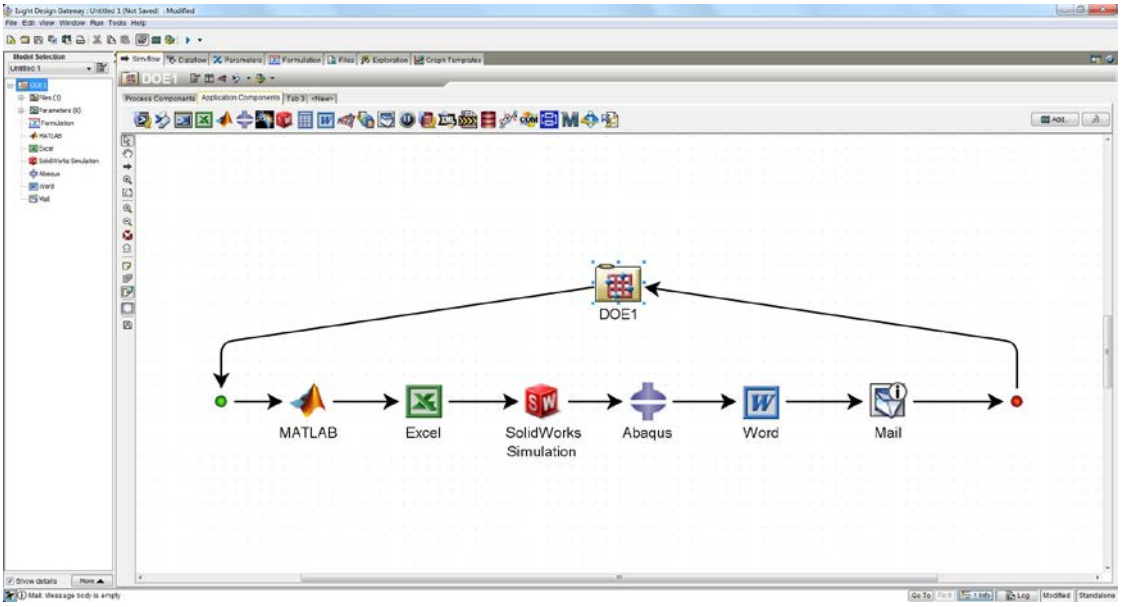
Το πρόγραμμα συνδυάζει πολυτομεακά μοντέλα και εφαρμογές μαζί σε μια προσομοίωση της ροής εργασίας, αυτοματοποιεί την εκτέλεσή τους, διερευνά τον προκύπτοντα χώρο σχεδιασμού και προσδιορίζει τις βέλτιστες παραμέτρους σχεδιασμού με βάση τους απαιτούμενους περιορισμούς. Η ικανότητα του Isight να χειρίζεται και να χαρτογραφεί παραμετρικά δεδομένα μεταξύ των βημάτων της διαδικασίας και να αυτοματοποιεί πολλαπλές προσομοιώσεις βελτιώνει σημαντικά την αποτελεσματικότητα, μειώνει τα ανθρώπινα σφάλματα και επιταχύνει την αξιολόγηση των εναλλακτικών λύσεων σχεδιασμού του προϊόντος.

Στην ουσία το λογισμικό αποτελεί ένα black-box στο οποίο δεν απαιτείται η γνώση πως εφαρμόζονται οι τεχνικές που χρησιμοποιεί, για να επιτευχθούν οι στόχοι που έχουν τεθεί. Ουσιαστικά το Isight δίνει την δυνατότητα στον χρήστη, να δώσει ως εισόδους μια πληθώρα από τύπους αρχείων, να επιλέξει την τεχνική η οποία ταιριάζει καλύτερα στο πρόβλημα και με το πάτημα ενός κουμπιού να προκύψουν τα επιθυμητά αποτελέσματα με ακρίβεια στους υπολογισμούς.

2.2 Αρχή λειτουργίας του προγράμματος

Το Isight έχει ένα φιλικό προς τον χρήστη γραφικό περιβάλλον διεπαφής όπου λειτουργεί με την μέθοδο των στοιχείων (components). Αυτά τα στοιχεία αποτελούν τα δομικά στοιχεία της προσομοίωσης της ροής της διαδικασίας. Κάθε component αποτελεί ένα βήμα για την δημιουργία, από τον χρήστη, προσομοίωσης της ροής διαδικασίας του μοντέλου του. Τα components του Isight είναι δύο ειδών: διεργασίας και εφαρμογής.

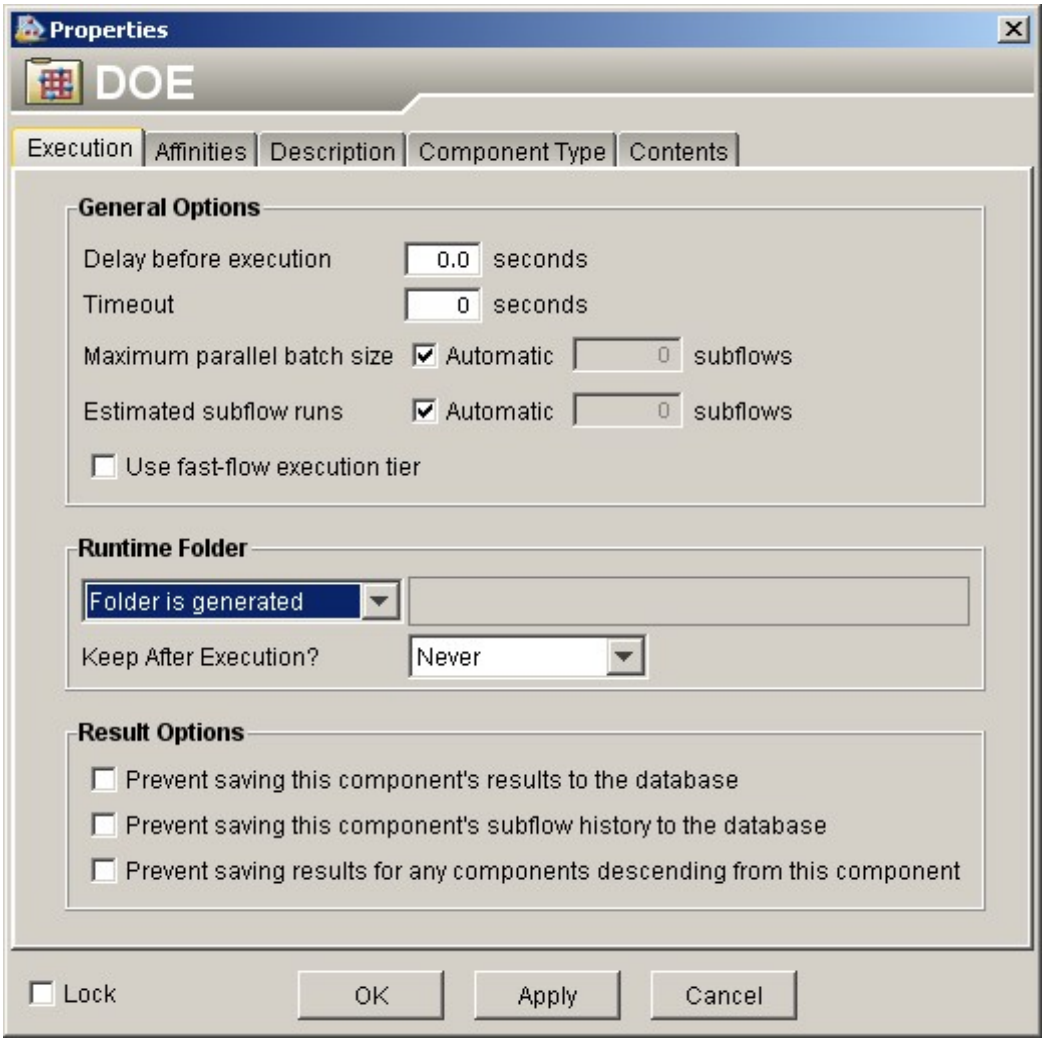
Η βιβλιοθήκη των components του Isight περιέχει components διεργασίας όπως DOE, Loop και Optimization καθώς και components εφαρμογών όπως Excel™, Word™, Calculator, CATIA V5™, Ansys™, MATLAB®, Ls-Dyna, εφαρμογές κειμένων I/O, Java, Python Scripting, Data Exchanger και βάσεις δεδομένων για την ενσωμάτωση και τη λειτουργία ενός μοντέλου ή μιας προσομοίωσης, ακόμα και Email, δίνοντας έτσι την δυνατότητα εύκολης ενσωμάτωσης των δεδομένων του προβλήματος σε οποιαδήποτε μορφή αυτά βρίσκονται. Η άμεση σύνδεση μεταξύ του Isight και των components επιτρέπει την εύκολη τροποποίηση ενός κομματιού που περιέχει τις εισόδους αναφοράς, την εκτέλεση του component και την εξαγωγή των πληροφοριών εξόδου. Στο σχήμα 2.1 φαίνεται μια προσομοίωση της ροής της διαδικασίας, που αποτελείται από διάφορα είδη components.



Σχήμα 2.1 Διάφορα components εφαρμογών

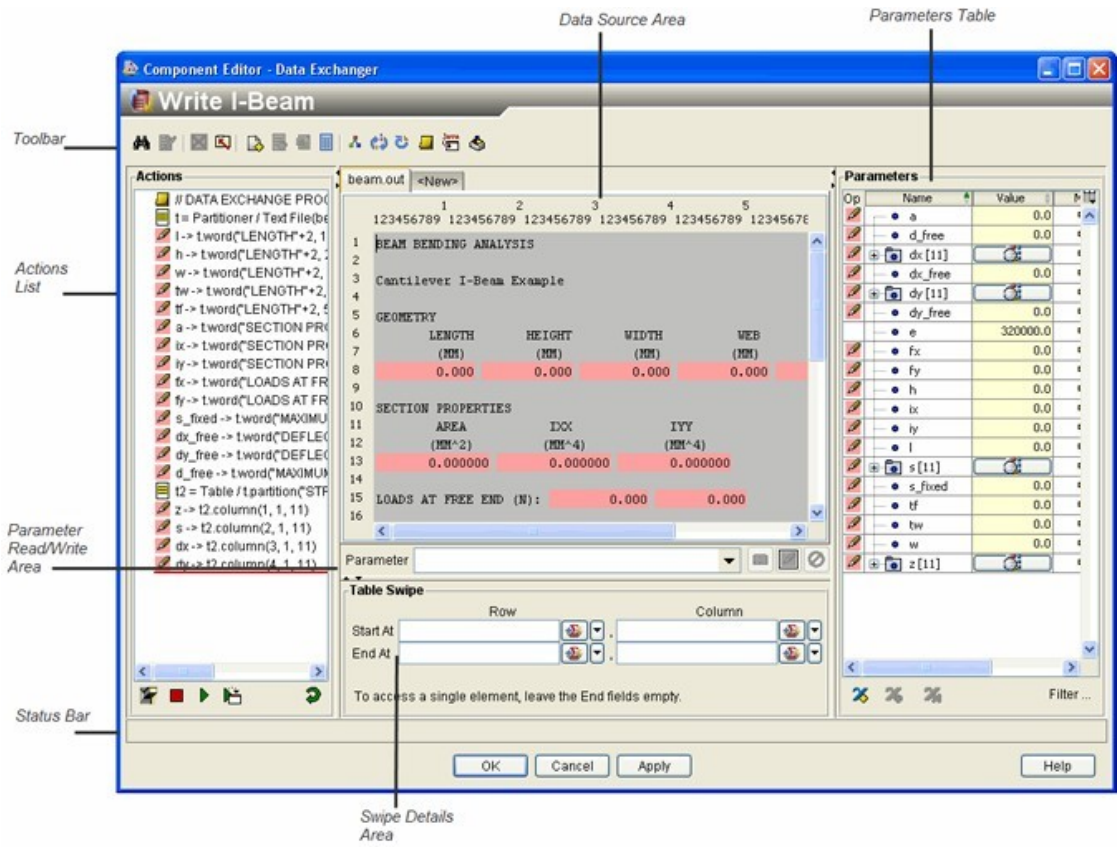
Τα components διεργασίας έχουν σχεδιαστεί για να περιέχουν μια προσομοίωση της ροής της διαδικασίας. Ανάλογα με τα χαρακτηριστικά κάθε component η ροή μπορεί να εκτελεστεί αρκετές φορές, όπως συμβαίνει με το component Loop το οποίο καθορίζει των αριθμών των βρόγχων μιας εκτέλεσης. Ουσιαστικά καθορίζουν την εκτέλεση όλης της διαδικασίας. Με την δημιουργία ενός component διεργασίας δίνεται η δυνατότητα επιλογής διαφόρων παραμέτρων εισόδου και εξόδου τις οποίες επιθυμεί ο χρήστης να τροποποιήσει ή να μελετήσει. Αυτό έχει ως αποτέλεσμα την δημιουργία αντίστοιχων παραμέτρων στο κύριο component κάθε υπορροής και την δημιουργία αντίστοιχων συνδέσεων στα υπόλοιπα component της υπορροής.

Κάθε component διεργασίας έχει μια σειρά από επιλογές οι οποίες είναι διαφορετικές σε κάθε component. Δίνεται η δυνατότητα στον χρήστη να ρυθμίσει αυτός τις επιλογές που επιθυμεί, ανάλογα με τις ιδιαιτερότητες του κάθε προβλήματος για την καλύτερη επίλυσή του. Κάποιες από τις επιλογές που μπορεί να ρυθμίσει ο χρήστης φαίνονται στο Σχήμα 2.2 .



Σχήμα 2.2 Διάφορες επιλογές στο component DOE

Τα components εφαρμογής είναι στοιχεία που έχουν σχεδιαστεί για να εκτελούν κάποια τελική λειτουργία, μερικές φορές αλληλεπιδρούν με μια εφαρμογή που είναι εξωτερική του Isight. Στην ουσία η κύρια ικανότητα των components εφαρμογής είναι να δίνουν την δυνατότητα εισαγωγής των δεδομένων του εκάστοτε προβλήματος με την χρήση του κατάλληλου component. Η επιλογή του κατάλληλου component γίνεται, ανάλογα την μορφή των δεδομένων του προβλήματος, έτσι ώστε να γίνει η επεξεργασία τους με την χρήση των τεχνικών του Isight. Στο Σχήμα 2.3 παρουσιάζεται η καρτέλα επιλογής παραμέτρων του component Data Exchanger.



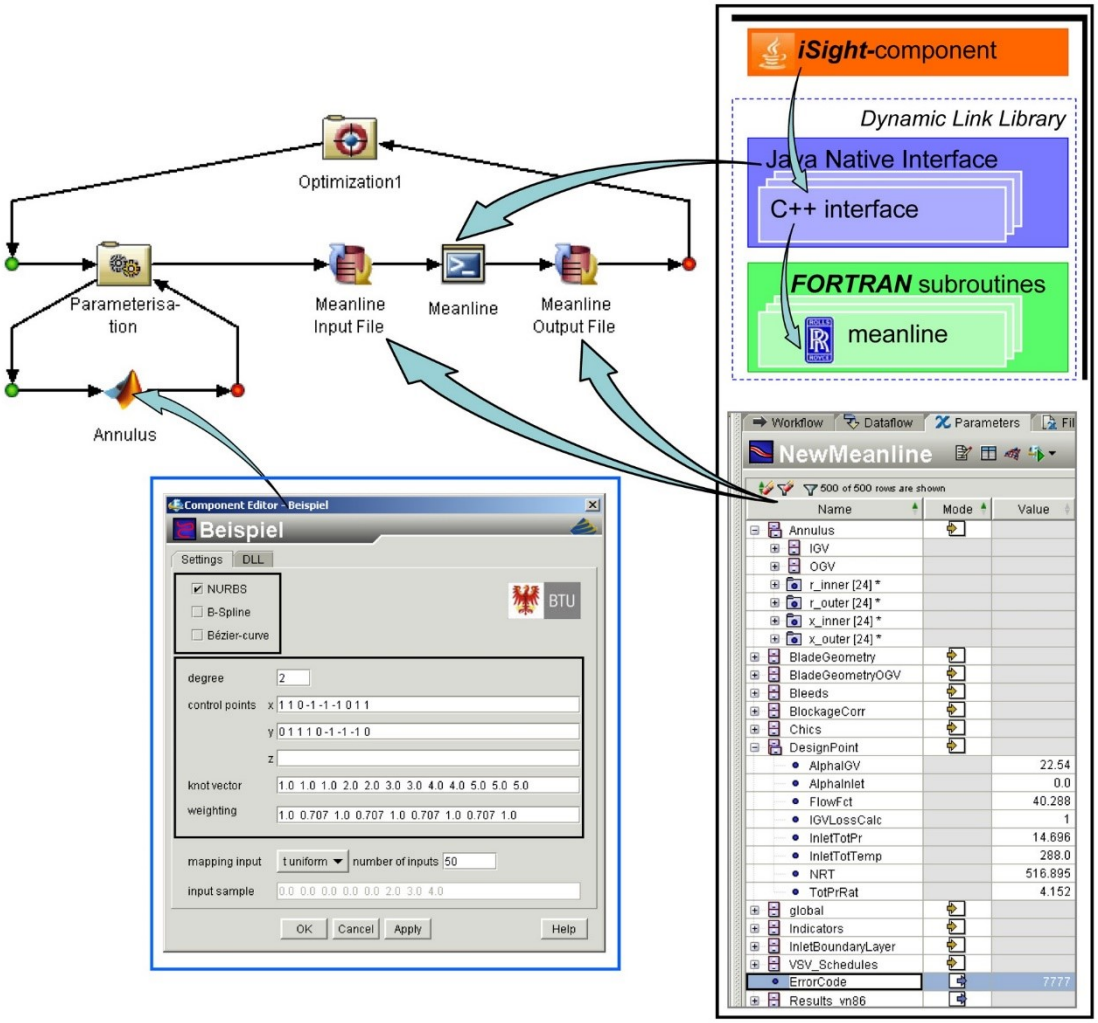
Σχήμα 2.3 Οι διάφορες επιλογές στο component Data Exchanger

2.3 Προσομοίωση της ροής της διαδικασίας

Το φιλικό προς τον χρήστη και απλό, γραφικό περιβάλλον του προγράμματος δίνει την δυνατότητα της αξιοποίησής του και από χρήστες που δεν έχουν ιδιαίτερη γνώση από προγραμματισμό. Το Isight παρέχει την δημιουργία της ροής διαδικασίας με την μέθοδο drag-and-drop των components, την χαρτογράφηση των παραμέτρων και την διατύπωση του προβλήματος. Το πρόγραμμα υποστηρίζει παραμέτρους αρχείων που μπορούν να αντιπροσωπεύουν μοντέλα προσομοίωσης ως μεταβλητές καθώς επίσης και δυναμικούς μεγάλους πίνακες τόσο ως είσοδοι αλλά και ως έξοδοι. Επιπλέον παρέχει διακλαδώσεις στην ροή της διαδικασίας, επαναλήψεις, εκτελέσεις υπό όρους μεγάλης πολυπλοκότητας με οποιαδήποτε παράμετρο και άλλες τεχνικές εκτέλεσης. Σε οποιοδήποτε σημείο της αυτοματοποιημένης διαδικασίας, μπορεί να σπάσει η ροή με την παρέμβαση του χρήστη. Αυτή η ευελιξία, σε συνδυασμό με Jython/Python scripts μπορεί να αλλάξει την συμπεριφορά της εκτέλεσης του μοντέλου ανάλογα με τις απαιτήσεις του κάθε προβλήματος.

Μια προσομοίωση της ροής της διαδικασίας είναι η συλλογή των εργασιών και υπολογισμών που υπάρχουν για τον υπολογισμό ενός μοντέλου, διατεταγμένα με λογική σειρά. Ένα μοντέλο μπορεί να περιέχει μόνο μία ροή που μπορεί να είναι ιεραρχική.

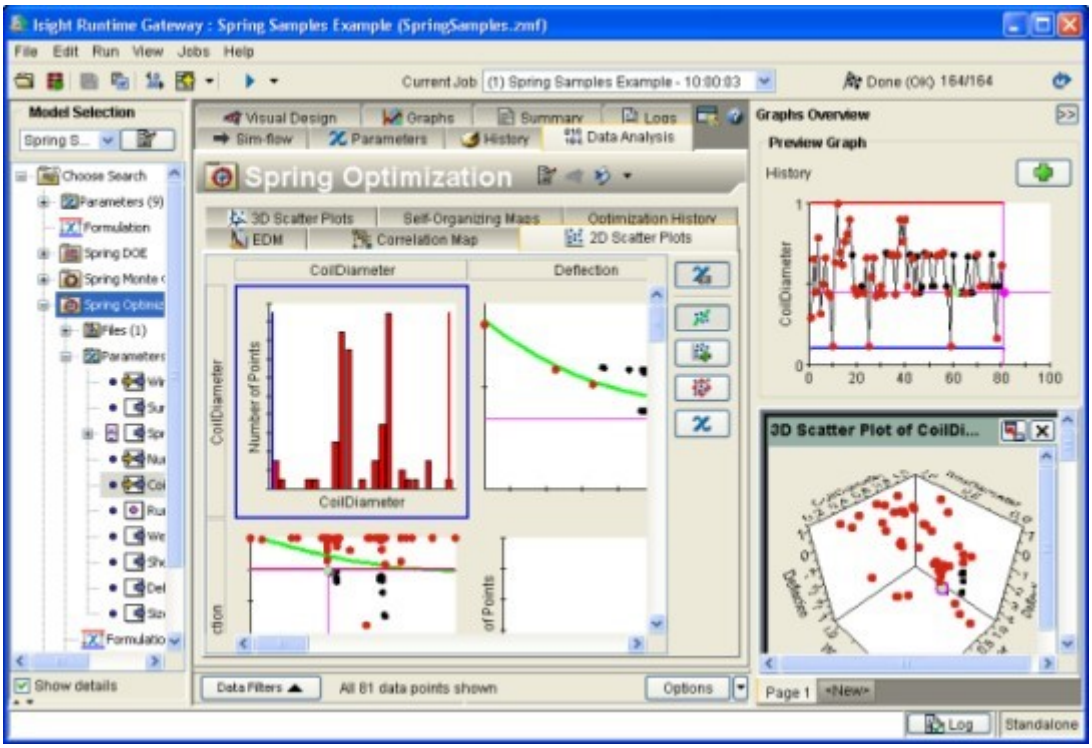
Κάθε βήμα της ροής αποτελεί ένα component είτε διεργασίας είτε εφαρμογής. Οι υποροές μπορούν να ενσωματωθούν σε οποιοδήποτε βαθμό, δηλαδή ένα component διεργασίας μπορεί να περιέχει ένα άλλο component διεργασίας το οποίο έχει τη δική του δευτερεύουσα ροή. Στο σχήμα 2.4 παρουσιάζεται μια ροή διαδικασίας, η οποία περιέχει διάφορα components διεργασίας και εφαρμογής, με ένα μάλιστα component να περιέχει και μια υποροή.



Σχήμα 2.4 Παράδειγμα ροής διαδικασίας που περιέχει και υποροή

Μόλις οριστεί η ροή της διαδικασίας, το πρόγραμμα επιτρέπει την εύκολη εισαγωγή εξωτερικών παραμέτρων. Βοηθητικά χαρακτηριστικά όπως η αναζήτηση μοντέλων, η προβολή του περιεχομένου του μοντέλου, η αναζήτηση παραμέτρων και η ομαδοποίηση παραμέτρων είναι επίσης διαθέσιμα. Με το πέρας του υπολογισμού, το Isight παρέχει δυνατότητα απεικόνισης των προβλημάτων σε γραφήματα αλλά και πίνακες. Δίνεται η δυνατότητα διαγραμμάτων 2Δ και 3Δ, χάρτες και πίνακες συσχετισμού, χάρτες αυτο-οργάνωσης και στατιστική ανάλυση. Τέλος, τα δεδομένα εκτέλεσης μπορούν να φιλτραριστούν και να ταξινομηθούν με ένα ευέλικτο σύνολο

κριτηρίων. Στο σχήμα 2.5 φαίνονται οι επιλογές που δίνονται στον χρήστη για την απεικόνιση των αποτελεσμάτων.



Σχήμα 2.5 Απεικόνιση αποτελεσμάτων στο Isight

Το πρόγραμμα προσφέρει πολλά πακέτα που χρησιμοποιούνται για την βελτιστοποίηση της προσομοίωσης της συμπεριφοράς των χαρακτηριστικών του μοντέλου όσον αφορά τις επιδόσεις, την διακύμανση της απόδοσης και την αξιοπιστία. Οι τεχνικές βελτιστοποίησης του σχεδιασμού μπορούν να χρησιμοποιηθούν μεμονωμένα ή να ενσωματωθούν στην ροή. Οι περισσότερες τεχνικές αξιοποιούν παράλληλες εκτελέσεις και επιλογές υποστήριξης για το χειρισμό αποτυχημένων εκτελέσεων. Τα χαρακτηριστικά των κυρίων components αναφέρονται στην συνέχεια.

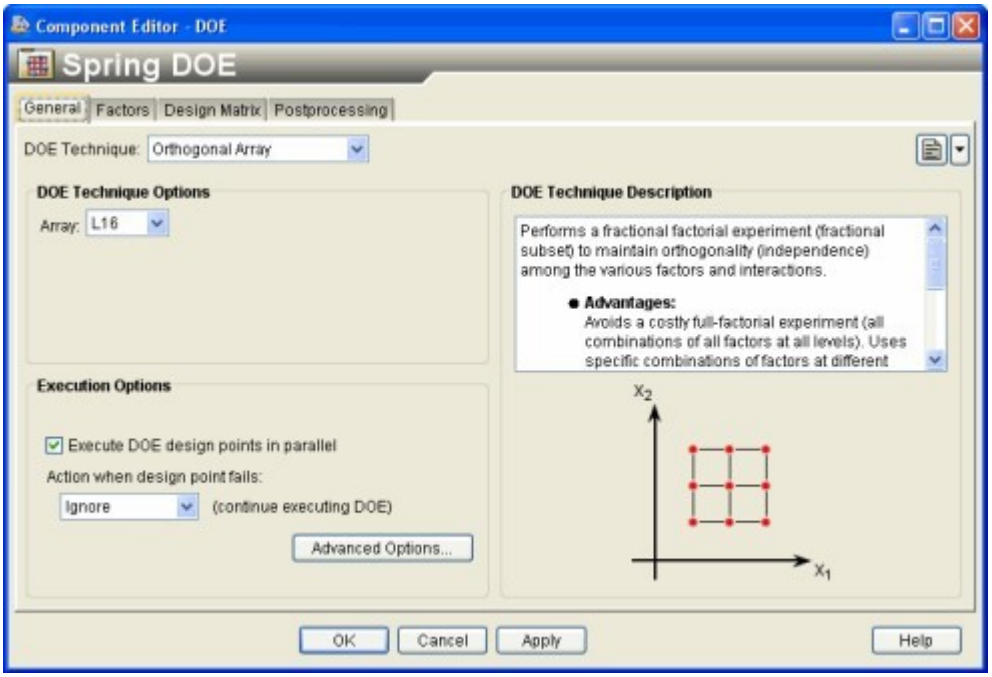
2.4 Χαρακτηριστικά των components

Το Isight προσφέρει διάφορες τεχνικές βελτιστοποίησης με την χρησιμοποίηση των components διεργασίας. Κάθε component διεργασίας έχει μια πληθώρα επιλογών σχετικά με τις τεχνικές που μπορούν να χρησιμοποιηθούν για την αντιμετώπιση του εκάστοτε προβλήματος. Επίσης, κάθε component έχει ξεχωριστές επιλογές ανάλογα με την τεχνική που επιλέγεται να χρησιμοποιηθεί.

Το component DOE επιτρέπει στους μηχανικούς να αξιολογήσουν γρήγορα την επίδραση διαφόρων μεταβλητών σχεδίασης, βάσει ενός συνόλου στόχων και να εντοπίσουν σημαντικές αλληλεπιδράσεις ανάμεσα στις παραμέτρους του προβλήματος.

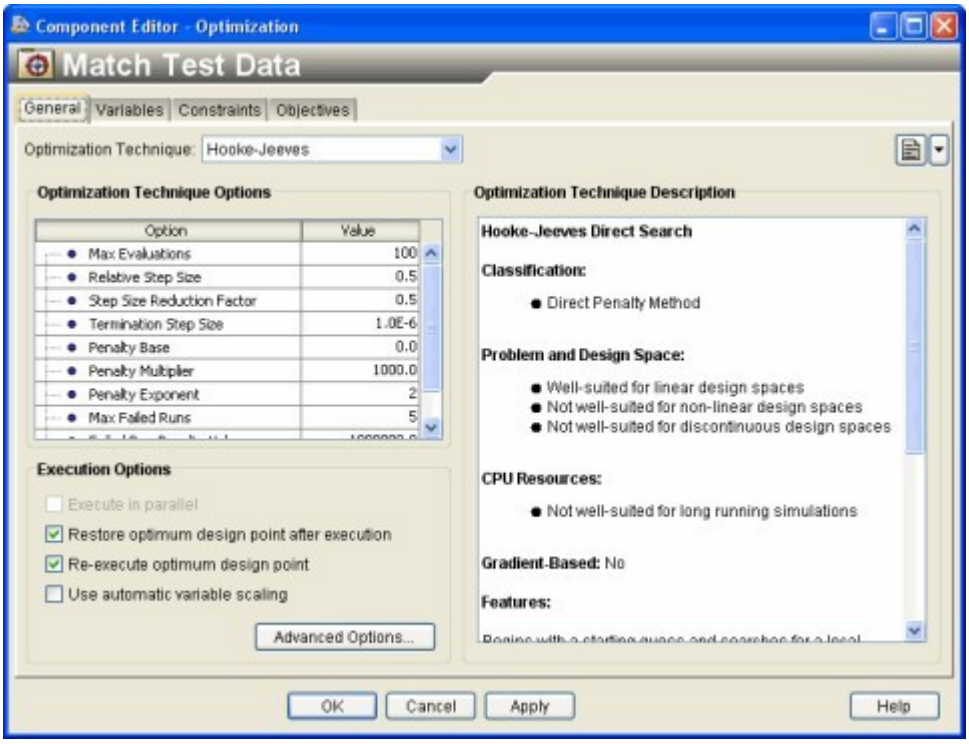
Να αναλύσουν έναν χώρο σχεδιασμού, ο οποίος παρέχει μια χονδρική εκτίμηση του βέλτιστου σχεδιασμού, το οποίο στην συνέχεια μπορεί να χρησιμοποιηθεί ως αρχικό σημείο για την αριθμητική βελτιστοποίηση. Τα δεδομένα σχεδιασμού που παράγονται από τις δοκιμές DOE μπορούν επίσης να χρησιμοποιηθούν με μοντέλα προσέγγισης του χώρου σχεδιασμού (μεταπρότυπα).

Υπάρχει μεγάλη ποικιλία DOE τεχνικών οι οποίες επιλέγονται στην επεξεργασία του component DOE. Κάποιες τεχνικές του component είναι οι Central Composite, Data File, Full Factorial, Fractional-Factorial, Box-Behnken, Latin Hypercube, Optimal Latin Hypercube, Orthogonal Array, Dependent Variable Sampling και Parameter Study με την κάθε τεχνική να έχει διάφορες επιλογές για την εκτέλεσή της. Στο σχήμα 2.6 φαίνεται η καρτέλα ρύθμισης του component DOE.



Σχήμα 2.6: DOE Component Editor

Το component Optimization παρέχει μια μεγάλη ποικιλία παράλληλων τεχνικών βελτιστοποίησης που μπορούν να εφαρμοστούν σε διάφορα προβλήματα. Επίσης περιλαμβάνει τεχνικές που μπορούν να χειριστούν πολυκριτηριακά προβλήματα βελτιστοποίησης. Κάθε τεχνική έχει και ξεχωριστές επιλογές για την καλύτερη ρύθμιση της, με βάση τις ανάγκες του κάθε προβλήματος. Στο σχήμα 2.7 φαίνεται η αντίστοιχη καρτέλα ρύθμισης του component Optimization.



Σχήμα 2.7 Optimization Component Editor

Κάποιες από τις τεχνικές που περιλαμβάνει το Isight και οι οποίες θα μελετηθούν εκτενώς αργότερα είναι:

- με βάση την κλίση (gradient-based) όπως οι NLPQL, MMFD, και LSGRG
- Αναζήτησης μοτίβου (Pattern-search) όπως Hooke-Jeeves, Downhill Simplex και Adaptive Simulated Annealing
- Mixed Integer/Real όπως οι MISQP, MOST
- γενετικοί αλγόριθμοι όπως Evol, Multi-Island GA
- πολυκριτηριακοί όπως οι AMGA, NSGA II, NCGA και Particle Swarm.
- και έναν γενικό αλγόριθμο τον Pointer.

2.5 Εφαρμογές του Isight σε πολυτομεακά προβλήματα βελτιστοποίησης

Το Isight έχει χρησιμοποιηθεί από ερευνητές για την αντιμετώπιση δύσκολων και σύνθετων προβλημάτων που περιέχουν συνιστώσες από διαφορετικά λογισμικά. Λόγω της εύκολης χρήσης του και την τεράστια ποικιλία των τεχνικών βελτιστοποίησης που διαθέτει, χρησιμοποιείται ευρέως για την αντιμετώπιση οποιοδήποτε προβλήματος στην αεροπορική βιομηχανία και όχι μόνο. Κάποιες εφαρμογές του παρουσιάζονται στην συνέχεια.

Ο Alexiou [17] εργάστηκε στην βελτιστοποίηση ενός κινητήρα διπλού ρεύματος με κιβώτιο ταχυτήτων με αντίστροφα περιστρεφόμενο πυρήνα σε σχέση με τον Fan για μικρής εμβέλειας πτήσεις (Geared Turbofan with Counter-Rotating core for Short Range

application, GTCRSR) με την χρήση του Isight. Τα αποτελέσματα έδειξαν μείωση της κατανάλωσης καυσίμου τουλάχιστον 7%. Η μοντελοποίηση των κινητήρων έγινε στο λογισμικό Proosis και στη συνέχεια το μοντέλο ενσωματώθηκε στο Isight για βελτιστοποίηση. Η μέθοδος βελτιστοποίησης του προβλήματος περιελάβανε την διαφοροποίηση κάποιων αεροθερμοδυναμικών παραμέτρων και αφού βρέθηκε το βέλτιστο σημείο σχεδίασης δημιουργήθηκε ένας χώρος σχεδιασμού DOE με πλήρη παραγοντική μελέτη (full factorial study). Οι παράμετροι σχεδιασμού διαφοροποιούνταν ανά δύο την φορά γύρω από την βέλτιστη τιμή τους και δημιουργήθηκε ένα μεταπρότυπο παρεμβάλλοντας τα γνωστά σημεία σχεδιασμού μεταξύ τους χρησιμοποιώντας τετραγωνικά πολυώνυμα.

Στην μελέτη των Deng, Guo et al. [18] για την αερομηχανική βελτιστοποίηση ενός διηχητικού ανεμιστήρα μεταξύ 95% και 105% της ταχύτητας περιστροφής. Ελήφθησαν υπόψη εννέα περιορισμοί μεταξύ των οποίων, για το περιθώριο του choke, για την παροχή μάζας, για τον λόγο πίεσης και για τις τάσεις στα πτερύγια. Η διαδικασία βελτιστοποίησης βασίστηκε πρώτα στην δημιουργία βάσης δεδομένων DOE για την αρχικοποίηση ενός Kriging μεταπροτύπου. Στη συνέχεια, ο μη κυριαρχικός γενετικός αλγόριθμος ταξινόμησης (NSGA-II) χρησιμοποιήθηκε για την απόκτηση μετώπου Pareto. Η βελτιστοποίηση του σχεδιασμού πέτυχε 0.2% κέρδος απόδοσης στο σημείο σχεδιασμού και 0.67% βελτίωση της παροχής μάζας στο 95%.

Παρόμοια μελέτη έγινε και από την Naz [19], για την πολυκριτηριακή βελτιστοποίηση των πτερυγίων ενός διηχητικού ανεμιστήρα. Οι επιμέρους συνιστώσες του προβλήματος δημιουργήθηκαν από το ANSYS για το CAD μοντέλο, ANSYS CFX για το υπολογιστικό ρευστομηχανικό μοντέλο και ANSYS PDL για την μηχανική καταπόνηση. Η μέθοδος βελτιστοποίησης περιελάβανε τους αλγόριθμους βελτιστοποίησης ASA, Pointer και MIGA χωρίς περαιτέρω επεξεργασία του καθενός. Η βελτιστοποίηση είχε ως αποτέλεσμα την αύξηση της απόδοσης κατά περίπου 1%.

Οι Larsson et al. [20] μελέτησαν την βελτιστοποίηση ενός συμβατικού και ενός Geared turbofan κινητήρα, ελαχιστοποιώντας τις εκπομπές CO_2 για μια δεδομένη αποστολή αεροσκάφους. Για την μοντελοποίηση του αεροσκάφους χρησιμοποιήθηκε το λογισμικό GISMO, για την μοντελοποίηση των κινητήρων το πρόγραμμα GESTPAN και για την προκαταρκτική σχεδίαση των κινητήρων σχετικά με τις διαστάσεις και τα βάρη τους το WEICO. Ύστερα ενσωμάτωσαν τις διάφορες συνιστώσες του προβλήματος στο Isight όπου πραγματοποιήθηκε η βελτιστοποίηση, οδηγώντας σε μείωση της SFC κατά 2.2% και συνολικής μείωσης του καυσίμου για όλη την αποστολή του αεροσκάφους κατά 3%, στον Geared turbofan σε σχέση με τον συμβατικό κινητήρα.

Όπως φαίνεται και από την βιβλιογραφία το Isight αποτελεί χρήσιμο εργαλείο στους ερευνητές για την αντιμετώπιση σύνθετων πολυτομεακών προβλημάτων. Η εύκολη ενσωμάτωση διάφορων τομέων, από σχεδόν όλα τα διάφορα λογισμικά που χρησιμοποιούνται στην αεροναυπηγική και η επιλογή των τεχνικών βελτιστοποίησης του το κάνουν ιδανικό πρόγραμμα για την αντιμετώπιση τέτοιου είδους προβλημάτων.

3 Τεχνικές που διαθέτει το Isight

Στο κεφάλαιο αυτό γίνεται περιγραφή των διαφόρων τεχνικών βελτιστοποίησης που περιλαμβάνει το πρόγραμμα Isight. Επίσης προτείνονται κάποιες από αυτές ως πιο κατάλληλες για την αντιμετώπιση του μη-γραμμικού ασυνεχή χώρου σχεδιασμού, πολυτομεακού προβλήματος της αποστολής του αεροσκάφους.

3.1 Design of Experiments

Το DOE είναι μια συλλογή στατιστικών τεχνικών που παρέχουν έναν συστηματικό και αποτελεσματικό τρόπο για την εξερεύνηση ενός χώρου σχεδιασμού. Είναι πολύ χρήσιμο για την αντιμετώπιση ενός προβλήματος για το οποίο ο χρήστης γνωρίζει ελάχιστα τον χώρο σχεδιασμού, παρέχοντας σημαντικές πληροφορίες που έχουν ως αποτέλεσμα την μείωση του χώρου σχεδιασμού. Επίσης, μελετούν τις επιδράσεις πολλαπλών μεταβλητών εισόδου σε μία ή περισσότερες παραμέτρους εξόδου. Επιπλέον, χρησιμοποιούνται συχνά πριν από τη δημιουργία ενός τυπικού προβλήματος βελτιστοποίησης, προκειμένου να προσδιοριστούν οι βασικοί παράγοντες μεταξύ των πιθανών μεταβλητών σχεδιασμού, των κατάλληλων μεταβλητών εύρους σχεδίασης και των επιτεύξιμων τιμών της αντικειμενικής συνάρτησης, υποστηρίζοντας τις παραδοσιακές διαδικασίες βελτιστοποίησης. Τέλος, τα δεδομένα DOE μπορούν να χρησιμοποιηθούν για την δημιουργία μεταπροτύπων.

Είναι επομένως σημαντικό, η επιλογή μιας κατάλληλης τεχνικής DOE, η οποία θα παρέχει με συστηματικό τρόπο τις μέγιστες δυνατές πληροφορίες για την απόκριση του συστήματος. Το Isight διαθέτει ένα πλήθος από τεχνικές DOE, κάποιες από τις οποίες παρουσιάζονται παρακάτω.

3.1.1 Full-Factorial Design Technique

Η τεχνική του πλήρους παραγοντικού σχεδιασμού (Full - Factorial Design) αξιολογεί όλους τους συνδυασμούς όλων των παραγόντων σε όλα τα επίπεδα. Συνήθως, η τυπική τεχνική πρακτική είναι να αξιολογηθεί συστηματικά ένα πλέγμα σημείων που απαιτεί $n_1 * n_2 * n_3 * \ldots . * n_i$ (i = αριθμός συντελεστών, n_i = αριθμός σημείων για τον παράγοντα i) εκτιμήσεις σημείων σχεδιασμού. Αυτή η πρακτική παρέχει εκτεταμένες πληροφορίες για την ακριβή εκτίμηση των επιδράσεων κάθε παράγοντα και της αλληλεπίδρασης τους. Ωστόσο, η χρήση της γίνεται απαγορευτική ως προς το υπολογιστικό κόστος λόγω του υψηλού αριθμού των απαιτούμενων αναλύσεων.

3.1.2 Fractional Factorial Design Technique

Η τεχνική κλασματικού παραγοντικού σχεδιασμού είναι ένα ορισμένο κλασματικό υποσύνολο 1/2, 1/4, 1/8, κ.λπ. για συντελεστές δύο επιπέδων και 1/3, 1/9, 1/27 κλπ. για παράγοντες τριών επιπέδων, του πλήρες παραγοντικού πειράματος. Ένας σχεδιασμός κλασματικού παράγοντα αποφεύγει ένα πλήρες παραγοντικό πείραμα στο οποίο μελετώνται όλοι οι συνδυασμοί όλων των παραγόντων (inputs) σε διαφορετικά επίπεδα (*p^n* για *n* παράγοντες, καθένας στα επίπεδα *p*). Σε κλασματικούς παραγοντικούς σχεδιασμούς ο αριθμός των στηλών στον πίνακα σχεδιασμού είναι μικρότερος από τον αριθμό που απαιτείται για να αντιπροσωπεύει κάθε παράγοντα και όλες τις αλληλεπιδράσεις αυτών των παραγόντων. Αντίθετα, οι στήλες "μοιράζονται" αυτές τις ποσότητες, γεγονός που είναι γνωστό ως σύγχυση (confounding), δηλαδή μια μεταβλητή μπορεί να επηρεάσει μια άλλη μεταβλητή που κανονικά είναι ανεξάρτητη αυτής. Το αποτέλεσμα της σύγχυσης είναι ότι δεν μπορεί κάποιος να προσδιορίσει ποια ποσότητα σε μια δεδομένη στήλη παρήγαγε την επίδραση στα outputs που αποδίδονται στη συγκεκριμένη στήλη.

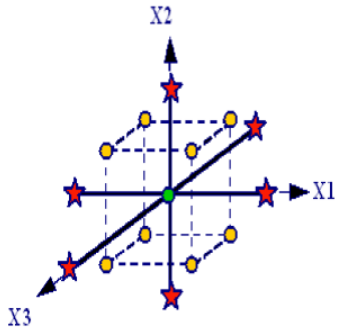
3.1.3 Parameter Study Technique

Η Παραμετρική Μελέτη (Parameter Study) μελετά κάθε παράγοντα σε όλα τα καθορισμένα επίπεδα ενώ όλοι οι άλλοι παράγοντες διατηρούνται σταθεροί στην αρχική τους τιμή. Ο όρος "Parameter Study", στο Isight, χρησιμοποιείται για να αναφερθεί σε μια πραγματική μελέτη της ευαισθησίας του σχεδιασμού (sensitivity analysis) σε κάθε παράγοντα ανεξάρτητο από όλους τους άλλους παράγοντες. Κάθε παράγοντας μελετάται σε όλα τα καθορισμένα επίπεδα, τιμές, ενώ όλοι οι άλλοι παράγοντες διατηρούνται σταθεροί στην αρχική τους τιμή. Αν και αυτό δεν παρέχει πληροφορίες αλληλεπίδρασης, επιτρέπει την ανάλυση πολλών παραγόντων σε πολλά επίπεδα με σχετικά λίγες αξιολογήσεις σημείων σχεδιασμού. Εάν οι αλληλεπιδράσεις είναι ασήμαντες τότε, τα αποτελέσματα είναι ένας καλός δείκτης των επιδράσεων των επιμέρους παραγόντων. Επειδή όμως οι παράγοντες που μελετώνται στο πρόβλημα του αεροπορικού κινητήρα επηρεάζει ο ένας τον άλλον, η τεχνική δεν ενδείκνυται. Ωστόσο στο παρελθόν έχει χρησιμοποιηθεί με ικανοποιητικά αποτελέσματα [17].

3.1.4 Central Composite Design Technique

Ο σχεδιασμός κεντρικής σύνθεσης (Central Composite Design), είναι μια στατιστικά βασισμένη τεχνική στην οποία ένα πείραμα πλήρους παραγόντων δύο επιπέδων αυξάνεται με επιπλέον σημεία. Ένα κεντρικό σημείο και δύο επιπλέον για κάθε παράγοντα, που ονομάζονται «αστέρια». Έτσι πέντε επίπεδα ορίζονται για κάθε παράγοντα, και για την μελέτη *n* παραγόντων χρειαζόμαστε $2^n + 2n + 1$ αξιολογήσεις (Σχήμα 3.1). Παρόλο που απαιτεί ένα μεγάλο αριθμό αξιολογήσεων είναι δημοφιλής

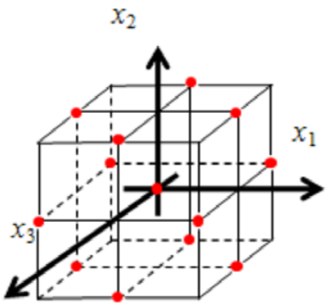
μέθοδος για την συλλογή δεδομένων για μοντέλα απόκρισης επιφάνειας (response surface modeling) εξαιτίας της μεγάλης έκτασης του χώρου σχεδιασμού που καταλαμβάνει και της υψηλής πληροφορίας που λαμβάνεται.



Σχήμα 3.1 Γραφική απεικόνιση των κόμβων της τεχνικής CCD

3.1.5 Box-Behnken Design

Ο Box-Behnken σχεδιασμός είναι μια παραλλαγή του σχεδιασμού κεντρικής σύνθεσης. Είναι ένας ελλιπής τρισδιάστατος παραγοντικός σχεδιασμός, χτισμένος συνδυάζοντας παραγοντικούς σχεδιασμούς δυο επιπέδων με ατελή σχεδιασμούς μπλοκ, εκτός των ακρογωνιαίων σημείων. Ο σχεδιασμός Box-Behnken εισήχθη προκειμένου να περιοριστεί το μέγεθος του δείγματος καθώς ο αριθμός των παραμέτρων αυξάνεται. Στον σχεδιασμό Box-Behnken, ένα μπλοκ δειγμάτων που αντιστοιχεί σε ένα παραγοντικό σχεδιασμό δυο επιπέδων επαναλαμβάνεται σε διαφορετικά σύνολα παραμέτρων. Οι παράμετροι που δεν περιλαμβάνονται στον παραγοντικό σχεδιασμό παραμένουν στο μέσο επίπεδο τους σε όλο το μπλοκ. Εφόσον ο σχεδιασμός Box-Behnken δεν περιλαμβάνει τα ακρογωνιαία σημεία, ο σχεδιασμός είναι αρκετός χρήσιμος σε περιπτώσεις όπου τα ακρογωνιαία σημεία είναι είτε αριθμητικά ασταθή ή ακατόρθωτα. Ωστόσο είναι διαθέσιμος μόνο για 3-21 παράγοντες και ανάλογα με το πρόβλημα που αντιμετωπίζεται πιθανόν να μην γίνεται να χρησιμοποιηθούν. Το σχήμα 3.2 δείχνει τις θέσεις δειγμάτων για τρεις παράγοντες.



Σχήμα 3.2 Θέσεις δειγματων για τρεις παράγοντες της τεχνικής Box-Behnken

Οι τεχνικές Central Composite και Box-Behnken έχουν χρησιμοποιηθεί κατά καιρούς για την δημιουργία DOE. Ωστόσο οι τεχνικές αυτές έχουν αποδειχθεί μη επαρκής και χρησιμοποιούνται όλο και πιο αραιά. Ειδικότερα σε ένα πολυτομεακό πρόβλημα, ο αριθμός των παραγόντων που υπάρχουν κάνουν τις τεχνικές κοστοβόρες, λόγω του μεγάλου αριθμού των αναλύσεων που απαιτούνται για την δημιουργία χώρου σχεδιασμού.

3.1.6 Orthogonal Arrays

Οι ορθογώνιες συστοιχίες (Orthogonal Arrays) είναι ένας ειδικός τύπος κλασματικού παραγοντικού πειράματος επιλεγμένου προσεκτικά για να διατηρηθεί η ορθογωνικότητα (ανεξαρτησία), μεταξύ των διαφόρων παραγόντων και των αλληλεπιδράσεων τους. Αυτή η ορθογωνικότητα επιτρέπει την ανεξάρτητη εκτίμηση των επιδράσεων των παραγόντων και της αλληλεπίδρασης τους από το σύνολο των πειραματικών αποτελεσμάτων.

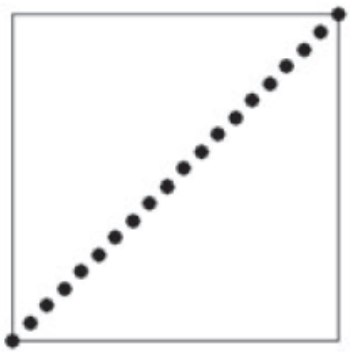
Χρησιμοποιώντας Orthogonal Arrays για τον κλασματικό παραγοντικό σχεδιασμό μειώνεται ο αριθμός των αξιολογήσεων για την ανάλυση των αποτελεσμάτων. Ωστόσο, η σημαντική μείωση του απαιτούμενου αριθμού πειραμάτων μπορεί συχνά να δικαιολογήσει αυτήν την απώλεια της ανάλυσης εφόσον μερικές από τις αλληλεπιδράσεις θεωρούνται αμελητέες. Ο εκάστοτε μηχανικός πρέπει να κάνει μια παραδοχή ως προς ποιες ποσότητες είναι ασήμαντες (συνήθως οι αλληλεπιδράσεις υψηλότερης τάξης) ανάλογα με το πρόβλημα, έτσι ώστε να μπορεί να προσδιοριστεί μια ενιαία συνεισφορά. Στην ουσία, για έναν ορθογώνιο πίνακα ενός δεδομένου μεγέθους, όσο περισσότεροι παράγοντες και αλληλεπιδράσεις μελετώνται, τόσο μεγαλύτερη είναι η σύγχυση (confounding).

Η σύγχυση συμβαίνει όταν οι στήλες "μοιράζονται" τις ποσότητες. Το αποτέλεσμα της σύγχυσης είναι ότι δεν μπορεί να καθοριστεί ποια ποσότητα σε μια δεδομένη στήλη παρήγαγε την επίδραση στις εκροές που αποδίδονται στη στήλη αυτή, από την ανάλυση μεταγενέστερης επεξεργασίας. Αυτό έχει ως αποτέλεσμα χαμηλότερη εμπιστοσύνη στα αποτελέσματα της ανάλυσης, επειδή πρέπει να γίνουν περισσότερες παραδοχές για τους ασήμαντους παράγοντες. Οι συστοιχίες τριών επιπέδων επιτρέπουν επίσης, την εκτίμηση των επιπτώσεων δεύτερης τάξης, δηλαδή της καμπυλότητας του σχεδιασμού.

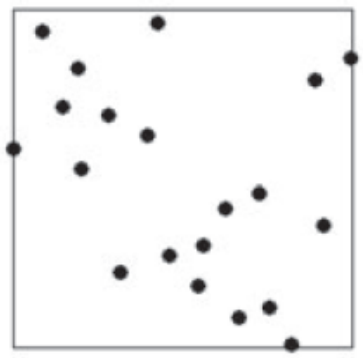
Η χρήση Orthogonal Arrays παρέχει μια συστηματική και αποτελεσματική μέθοδο μελέτης του χώρου σχεδιασμού και παρέχει προτάσεις για τη βελτίωση του σχεδιασμού. Ωστόσο, τα πραγματικά καθήκοντα επιλογής των κατάλληλων ορθογώνιων συστοιχιών για χρήση και ανάθεση των παραγόντων και των αλληλεπιδράσεων στις στήλες μπορεί να είναι κουραστική και επίπονη. Η αυτοματοποίηση αυτής της διαδικασίας στο Isight επιτρέπει την αποδοτική και αποτελεσματική μελέτη του χώρου σχεδιασμού χωρίς να χρειάζεται από τον χρήστη να έχει γνώση της μεθόδου.

3.1.7 Latin Hypercube

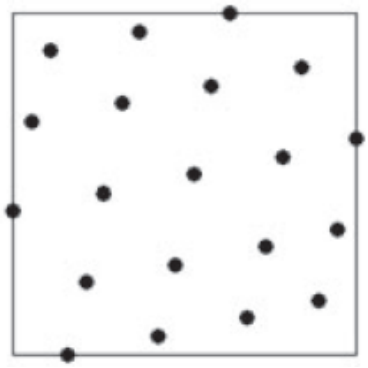
Έστω, ένας πειραματικός σχεδιασμός με p σημεία σε d διαστάσεις, που γράφεται σαν ένας πίνακας $p \times d$ με $\mathbf{X}=[\mathbf{x}_1\ \mathbf{x}_2\ ..\ \mathbf{x}_p]^T$, όπου κάθε στήλη αντιπροσωπεύει μια μεταβλητή και κάθε σειρά $\mathbf{x}_i= [x_i^{(1)}\ x_i^{(2)}\ ..\ x_i^{(d)}]$ αντιπροσωπεύει ένα δείγμα. Ένας Latin Hypercube σχεδιασμός κατασκευάζεται, με τέτοιον τρόπο έτσι ώστε για καθεμιά από τις d διαστάσεις που χωρίζεται σε p ίσα επίπεδα να υπάρχει ένα μόνο σημείο σε κάθε επίπεδο. Μια τυχαία διαδικασία χρησιμοποιείται για να καθορίσει τις θέσεις των σημείων. Στο σχήμα 3.3 φαίνονται τρία παραδείγματα της Latin Hypercube με d=2 διαστάσεις και p=20 σημεία. Όπως φαίνεται από την ακραία περίπτωση του Σχ.3.3 (α) υπάρχει ανεπαρκής κάλυψη του χώρου. Η τυχαιοποίηση μόνη της μπορεί να βελτιώσει τον χώρο σχεδιασμού των πειραμάτων όπως φαίνεται στο Σχ. 3.3 (β), ωστόσο η βελτιστοποίηση της θέσης των σημείων θα μας δώσει καλύτερα αποτελέσματα όπως φαίνεται στο Σχ.3.3 (γ) όπου τα σημεία είναι πιο ομοιόμορφα κατανεμημένα.



Σχήμα 3.3 (α) Σχεδιασμός με κακή κάλυψη του χώρου



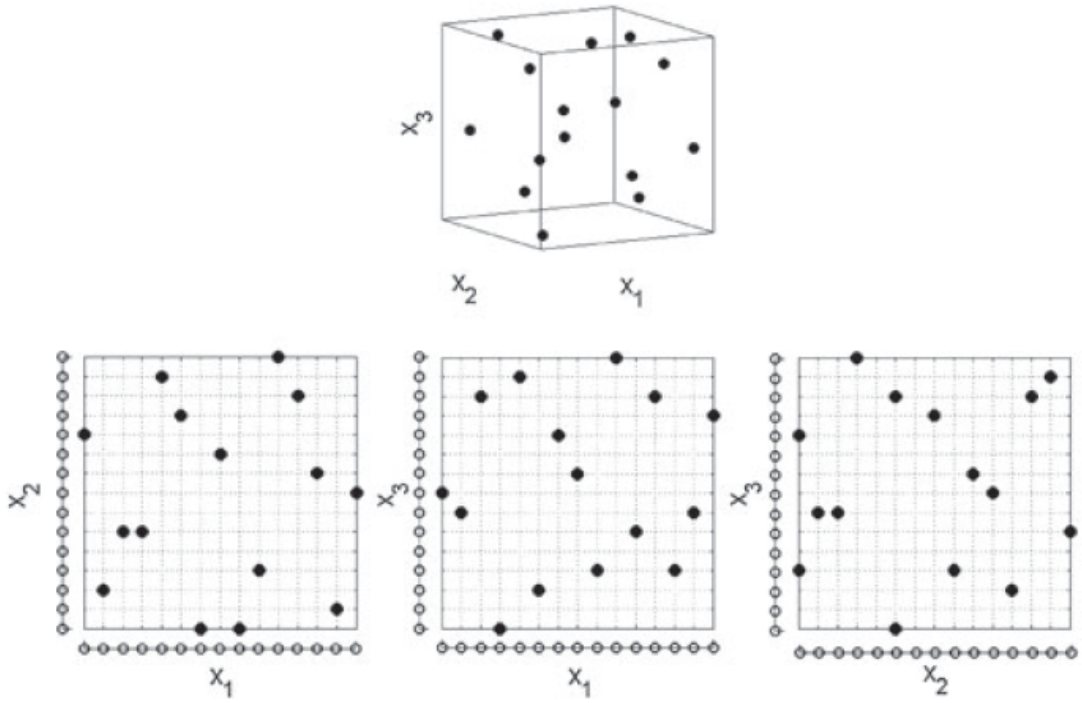
Σχήμα 3.3 (β) Τυχαίος σχεδιασμός



Σχήμα 3.3 (γ) Σχεδιασμός με καλή κάλυψη του χώρου

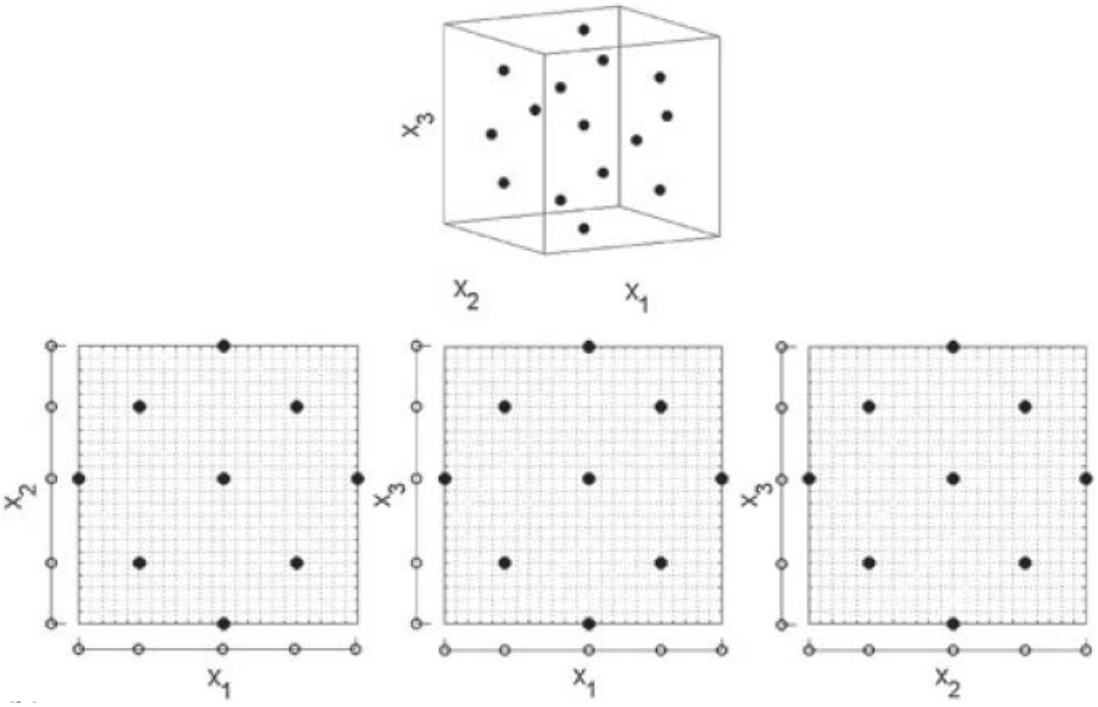
Η διαδεδομένη χρήση της Latin Hypercube οφείλεται στην ευελιξία της μεθόδου. Για παράδειγμα, εάν λίγες διαστάσεις απορριφθούν, ο τελικός σχεδιασμός παραμένει Latin Hypercube με όλα τα πλεονεκτήματα της, όπως ότι τα δείγματα δεν «καταρρέουν». Το Σχ. 3.4 δείχνει την περίπτωση ενός σχεδιασμού Latin Hypercube με d=3 διαστάσεις και p=15 σημεία. Κάθε μία από τις δισδιάστατες προβολές είναι ακόμα ένας σχεδιασμός Latin Hypercube με τα ίδια p=15 σημεία, στην συγκεκριμένη περίπτωση όμως , η x_1-x_2 προβολή είναι η καλύτερη στο να καλύπτει χώρο (space filling). Έτσι, εάν το υπολογιστικό κόστος για την παραγωγή νέου σετ δεδομένων που είναι ιδανικό για μικρότερο χώρο είναι μεγάλο, τα υπάρχοντα δεδομένα μπορούν να επαναχρησιμοποιηθούν χωρίς μείωση στον αριθμό των σημείων δειγματοληψίας. Αυτό δεν συμβαίνει στην περίπτωση όπου central composite ή παραγοντικά σχέδια

χρησιμοποιούνται . Σε αυτές τις περιπτώσεις, όταν κάποιες διαστάσεις απορριφθούν, τα σημεία «καταρρέουν» το ένα μέσα στο άλλο, μειώνοντας το μέγεθος του δείγματος.



Σχήμα 3.4 Σχεδιασμός Latin Hypercube με 15 σημεία

Το Σχ. 3.5 δείχνει την περίπτωση ενός central composite με *d=3* διαστάσεις, που προκύπτει έτσι με *p=15* σημεία. Ενώ οι δισδιάστατες, είναι ακόμα central composite σχέδια, τα σημεία «καταρρέουν» το ένα στο άλλο, και ο ουσιαστικός χώρος σχεδιασμού μειώνεται στα *p=9* σημεία.



Σχήμα 3.5 Σχεδιασμός Central Composite με 15 σημεία

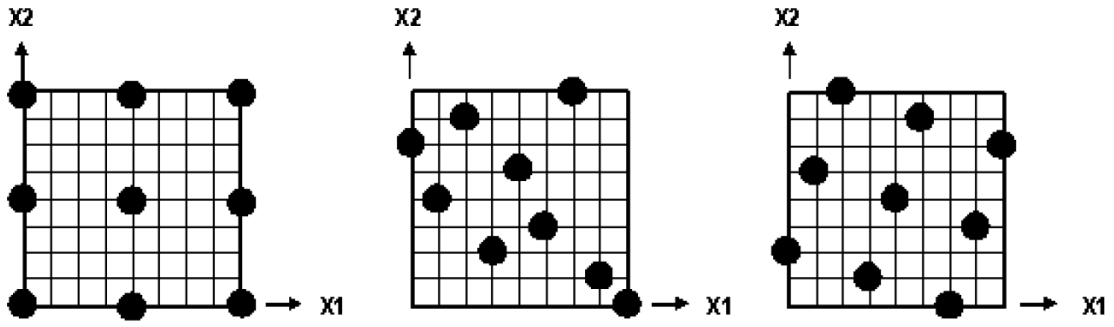
Δυστυχώς, η ίδια διαδικασία δεν μπορεί να χρησιμοποιηθεί στην περίπτωση της πρόσθεσης μεταβλητών σχεδιασμού στο πρόβλημα. Επειδή, όταν προστίθεται μια ή περισσότερες διαστάσεις στο πρόβλημα αυτό, λογικά προκύπτει ότι τα σημεία στο υπάρχον σχέδιο έχουν αυτές τις νέες διαστάσεις σταθερές σε ένα συγκεκριμένο επίπεδο. Αυτό διαταράσσει την ιδιότητα της μη- κατάρρευσης των Latin Hypercube σχεδιασμών, ή γενικά οποιουδήποτε σχεδιασμού. Στην πραγματικότητα, θα πρέπει κάποιος να εκτελέσει νέες προσομοιώσεις για να καλύψει τις νέες διαστάσεις. Για περιπτώσεις σαν αυτές, η καλύτερη πρακτική είναι να χρησιμοποιηθεί βελτιστοποίηση έτσι ώστε να καθοδηγήσει, που θα τοποθετηθούν τα νέα σημεία σχεδιασμού.

3.1.8 Optimal Latin Hypercube

Η βελτιστοποίηση της Latin Hypercube είναι μια δύσκολη διαδικασία επειδή είναι ένα συνδυαστικό πρόβλημα βελτιστοποίησης με χώρο αναζήτησης της τάξης $(p!)^d$. Για παράδειγμα, για να βελτιστοποιήσουμε την περιοχή 20 δειγμάτων σε δυο διαστάσεις, ο αλγόριθμος πρέπει να επιλέξει τον καλύτερο σχεδιασμό ανάμεσα από 10^{36} πιθανά σχέδια. Εάν ο αριθμός των μεταβλητών αυξηθεί σε 3 τότε, ο αριθμός των πιθανών σχεδίων είναι 10^{55}.

Στο Isight η βελτιστοποίηση της Latin Hypercube ξεκινάει με την δημιουργία ενός τυχαίου Latin Hypercube σαν DOE και στην συνέχεια βελτιστοποιείται. Με την εναλλαγή της τάξης παραγόντων δύο επιπέδου σε μια στήλη του πίνακα, ένας νέος πίνακας παράγεται όπου τα σημεία κατανέμονται όσο πιο ομοιόμορφα γίνεται μέσα στον χώρο σχεδιασμού, που ορίζεται από το κατώτατο και ανώτερο επίπεδο κάθε παράγοντα.

Η ιδέα Optimal Latin Hypercube απεικονίζεται στο Σχήμα 3.6 για μια διαμόρφωση με δύο παράγοντες (X1, X2) και εννέα σημεία σχεδιασμού. Στο Σχήμα 3.6 (α), εμφανίζεται ένας τυποποιημένος ορθογώνιος πίνακας τριών επιπέδων. Ενώ αυτός ο πίνακας έχει εννέα σημεία σχεδιασμού, υπάρχουν μόνο τρία επίπεδα για κάθε παράγοντα. Το Σχήμα 3.6 (β) δείχνει ένα τυχαίο Latin Hypercube. Αυτός ο πίνακας περιλαμβάνει επίσης εννέα σημεία σχεδιασμού για τους δύο παράγοντες, ωστόσο δεν κατανέμονται ομοιόμορφα στο χώρο σχεδιασμού. Για παράδειγμα, υπάρχουν ελάχιστες πληροφορίες στην άνω δεξιά και κάτω αριστερή γωνία του χώρου σχεδιασμού. Στο Σχήμα 3.6 (γ) εμφανίζεται ένας πίνακας Optimal Latin Hypercube. Με αυτό το πλέγμα, τα εννέα σημεία σχεδιασμού καλύπτουν εννέα επίπεδα κάθε παράγοντα και κατανέμονται ομοιόμορφα στο χώρο σχεδιασμού.



Σχήμα 3.6 (α) Orthogonal Arrays (β) Latin Hypercube (γ) Optimal Latin Hypercube

Ένα πλεονέκτημα της χρήσης της Latin Hypercube από την Orthogonal Arrays είναι ότι μπορούν να μελετηθούν περισσότερα σημεία και περισσότεροι συνδυασμοί για κάθε παράγοντα. Η τεχνική Latin Hypercube επιτρέπει στον σχεδιαστή πλήρη ελευθερία στην επιλογή του αριθμού των σχεδιασμών που πρέπει να εκτελεστούν (εφόσον είναι μεγαλύτεροι από τον αριθμό των παραγόντων). Οι διαμορφώσεις είναι πιο περιοριστικές χρησιμοποιώντας την τεχνική Orthogonal Arrays (L4, L8, κ.λπ.).

Για την βελτιστοποίηση της Latin Hypercube διάφορα κριτήρια βελτιστοποίησης μπορούν να χρησιμοποιηθούν για την κατανομή των σημείων. Το Isight χρησιμοποιεί ένα ευρέως χρησιμοποιούμενο κριτήριο βελτιστοποίησης σε αυτή την εφαρμογή: το φ_p κριτήριο που βασίζεται στο *maximin distance criterion.* Όπως περιγράφεται στο [21]:
Ένας σχεδιασμός λέγεται σχεδιασμός maximin απόστασης [22] εάν μεγιστοποιεί την ελάχιστη απόσταση μεταξύ των θέσεων:

$$\min\ d(x_i, x_j),$$
$$1 \le i, j \le n, i \ne j$$

Όπου $d(x_i,x_j)$, είναι η απόσταση μεταξύ δύο σημείων δειγματοληψίας x_i και x_j:

$$d(x_i, x_j) = d_{ij} = \left[\sum_{k=1}^{m} |x_{ik} - x_{jk}|^t\right]^{1/t} \quad , t = 1\ or\ 2$$

Οι Morris και Mitchell [23] πρότειναν μια διαισθητική επέκταση του κριτηρίου maxmin απόστασης. Για κάθε δεδομένο σχεδιασμό, ταξινομώντας όλες τις μεταξύ των σημείων απόσταση $d_{ij}(1 \le i,j \le n,\ i \ne j)$, μια λίστα απόστασης, $(d_1,d_2,\ldots d_s)$ και μια λίστα ευρετηρίου $(J_1,J_2,..,J_s)$ μπορεί να αποκτηθεί, όπου d_i είναι διακριτές τιμές απόστασης με $d_1<d_2<\ldots<d_s$, J_i ο αριθμός των ζευγών θέσεων στον σχεδιασμό που χωρίζονται από d_i, και s είναι ο αριθμός των ξεχωριστών τιμών απόστασης. Ένας σχεδιασμός ονομάζεται φ_p-βέλτιστος σχεδιασμός αν ελαχιστοποιεί:

$$\varphi_p = \left[\sum_{i=1}^{s} J_i d_i^{-p}\right]^{1/p}$$

Όπου p είναι ένας θετικός ακέραιος. Με ένα πολύ μεγάλο p, το φ_p κριτήριο είναι ισοδύναμο με το κριτήριο maxmin απόστασης.

Υπάρχουν δύο σημαντικές εξελίξεις αυτού του αλγορίθμου, σε σύγκριση με άλλους αλγόριθμους βέλτιστου DOE, οι οποίοι αυξάνουν την αποδοτικότητα του αλγορίθμου:

• Ανάπτυξη ενός αποτελεσματικού αλγόριθμου αναζήτησης ολικού βέλτιστου, "ενισχυμένος στοχαστικός εξελικτικός αλγόριθμος" (ESE)

• Αποτελεσματικοί αλγόριθμοι για την αξιολόγηση των κριτηρίων βελτιστοποίησης (σημαντική μείωση των υπολογισμών πινάκων για την αξιολόγηση νέων/τροποποιημένων σχεδίων κατά την διάρκεια της αναζήτησης)

Σύμφωνα και με παλαιότερες μελέτες [7], [4] για την δημιουργία χώρου σχεδιασμού έχει χρησιμοποιηθεί ευρέως η τεχνική Latin Hypercube λόγω των πλεονεκτημάτων που έχει, σε σχέση με τις υπόλοιπες μεθόδους. Το Isight προσφέρει την δυνατότητα της βελτιστοποίησης της Latin Hypercube με την μορφή της επιλογής της Optimal Latin Hypercube έχοντας περισσότερα πλεονεκτήματα με μικρό υπολογιστικό κόστος.

3.2 Optimization Algorithms

Η διαδικασία βελτιστοποίησης χρησιμοποιώντας μεθόδους βασισμένες στην κλίση (Gradient-based) χρειάζονται πληροφορίες για τις παραγώγους για τις αντικειμενικές και περιοριστικές συναρτήσεις, για να αποκτήσουν μια κατεύθυνση αναζήτησης. Η απόδοση μιας μεθόδου gradient-based εξαρτάται σε μεγάλο βαθμό από τις αρχικά παρεχόμενες μεταβλητές σχεδιασμού. Παρόλο που είναι μέθοδοι ταχείας βελτιστοποίησης και απαιτούν μικρό αριθμό αξιολογήσεων, δεν είναι οι ιδανικότερες. Αυτές οι μέθοδοι είναι κατάλληλες για συνεχή χώρο σχεδιασμού αλλά μπορεί να μην είναι κατάλληλες για την αναζήτηση του ολικού βέλτιστου ενός πολυτομεακού προβλήματος, επειδή μπορούν εύκολα να παγιδευτούν σε τοπικό ακρότατο.

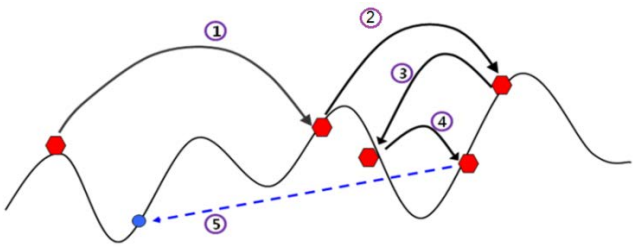
Οι non-gradient μέθοδοι δεν απαιτούν παραγώγους της αντικειμενικής συνάρτησης για την αναζήτηση του βέλτιστου σημείου. Ένα από τα πλεονεκτήματα αυτών των μεθόδων είναι, ότι είναι πιο πιθανό να βρουν το βέλτιστο σημείο και να μην παγιδευτούν σε τοπικό ακρότατο. Αυτές οι μέθοδοι είναι κατάλληλες για μη γραμμικά, πολύπλευρα προβλήματα και ασυνεχείς χώρους σχεδιασμού. Ωστόσο, οι μη gradient μέθοδοι χρειάζονται ένα μεγάλο αριθμό αξιολογήσεων και αυτό οδηγεί σε μεγάλο υπολογιστικό κόστος. Υπάρχουν πολλές μέθοδοι που ανήκουν σε αυτήν την κατηγορία μεθόδων όπως ο ASA, ο Evol και mPSO κλπ. Οι πιο διαδεδομένοι είναι οι εξελικτικοί αλγόριθμοι αλλά κάθε αλγόριθμος έχει τα πλεονεκτήματα και μειονεκτήματα του.

3.2.1 Adaptive Simulated Annealing (ASA)

Η Προσαρμοστική Προσομοίωση Ανόπτησης (Adaptive Simulated Annealing) είναι μια γενική πιθανολογική τεχνική για την προσέγγιση του ολικού βέλτιστου μιας δεδομένης συνάρτησης σε ένα μεγάλο χώρο αναζήτησης. Το όνομα και το πνεύμα της, προέρχονται από την ανόπτηση στη μεταλλουργία. Για την καλύτερη κατανόηση πως ο αλγόριθμος της μεθόδου λειτουργεί, βοηθά η απεικόνιση των προβλημάτων που παρουσιάζονται από τέτοια πολύπλοκα συστήματα ως ένα γεωγραφικό έδαφος. Για παράδειγμα, έστω μια οροσειρά, με δύο "παραμέτρους" όπως οι κατευθύνσεις Βορρά προς Νότο και Ανατολή προς Δύση. Σκοπός του προβλήματος είναι η εύρεση της χαμηλότερης κοιλάδας σε αυτό το έδαφος. Ο αλγόριθμος ASA προσεγγίζει αυτό το πρόβλημα παρόμοια με τη χρήση μιας μπάλας που μπορεί να αναπηδήσει πάνω από τα βουνά από κοιλάδα σε κοιλάδα. Αρχίζει με μια υψηλή θερμοκρασία, όπου θερμοκρασία είναι μια παράμετρος της ASA που μιμείται την επίδραση ενός ταχέως κινούμενου σωματιδίου σε ένα θερμό αντικείμενο όπως ένα θερμό τετηγμένο μέταλλο. Η μπάλα επιτρέπεται να κάνει πολύ υψηλές αναπηδήσεις και είναι σε θέση να αναπηδήσει πάνω από οποιοδήποτε βουνό για να αποκτήσει πρόσβαση σε οποιαδήποτε κοιλάδα, εάν της επιτραπούν αρκετές αναπηδήσεις. Καθώς η θερμοκρασία γίνεται σχετικά χαμηλή, η σφαίρα δεν μπορεί να αναπηδήσει τόσο ψηλά και μπορεί να καθίσει, παγιδευμένη σε σχετικά μικρότερες σειρές κοιλάδων.

Η «οροσειρά» περιγράφεται εύστοχα από μια "συνάρτηση κόστους" όπου στο Isight είναι η παράμετρος Objective and Penalty. Ορίζονται οι κατανομές πιθανοτήτων των δύο κατευθυντήρων παραμέτρων (παραγωγικές κατανομές), δημιουργώντας πιθανές κοιλάδες ή καταστάσεις που πρέπει να διερευνηθούν. Ορίζεται μια άλλη κατανομή, που ονομάζεται κατανομή αποδοχής, η οποία εξαρτάται από τη διαφορά των συναρτήσεων κόστους της τωρινής παραγόμενης κοιλάδας που πρόκειται να εξερευνήσει και της τελευταίας αποθηκευμένης χαμηλότερης κοιλάδας. Η κατανομή αποδοχής αποφασίζει πιθανολογικά αν θα παραμείνει σε μια νέα χαμηλότερη κοιλάδα ή θα αναπηδήσει από αυτήν. Όλες οι κατανομές παραγωγής και αποδοχής εξαρτώνται από τη θερμοκρασία. Η παράμετρος αυτή μπορεί να αναπροσαρμόζεται περιοδικά ή να αυξάνεται σε σχέση με τις προηγούμενες τιμές της, χρησιμοποιώντας τις σχετικές πρώτες παραγώγους με σεβασμό στη συνάρτηση κόστους, για να κατευθύνει την αναζήτηση «δίκαια» μεταξύ των παραμέτρων.

Το σχήμα 3.7 δείχνει τη διαδικασία αναζήτησης της μεθόδου. Στην αρχή, όταν η θερμοκρασία αυξάνεται, τα άτομα μετακινούνται σε υψηλή ενεργειακή κατάσταση (βήματα 1 και 2.) Όταν ψύχονται αργά, τα άτομα μετακινούνται μόνο στην κατάσταση ισορροπίας (βήματα 3 & 4). Εάν είναι υψηλότερο από το άτομο, το άτομο μετακινείται σε χαμηλότερη ενέργεια.



Σχήμα 3.7 Διαδικασία αναζήτησης της ASA

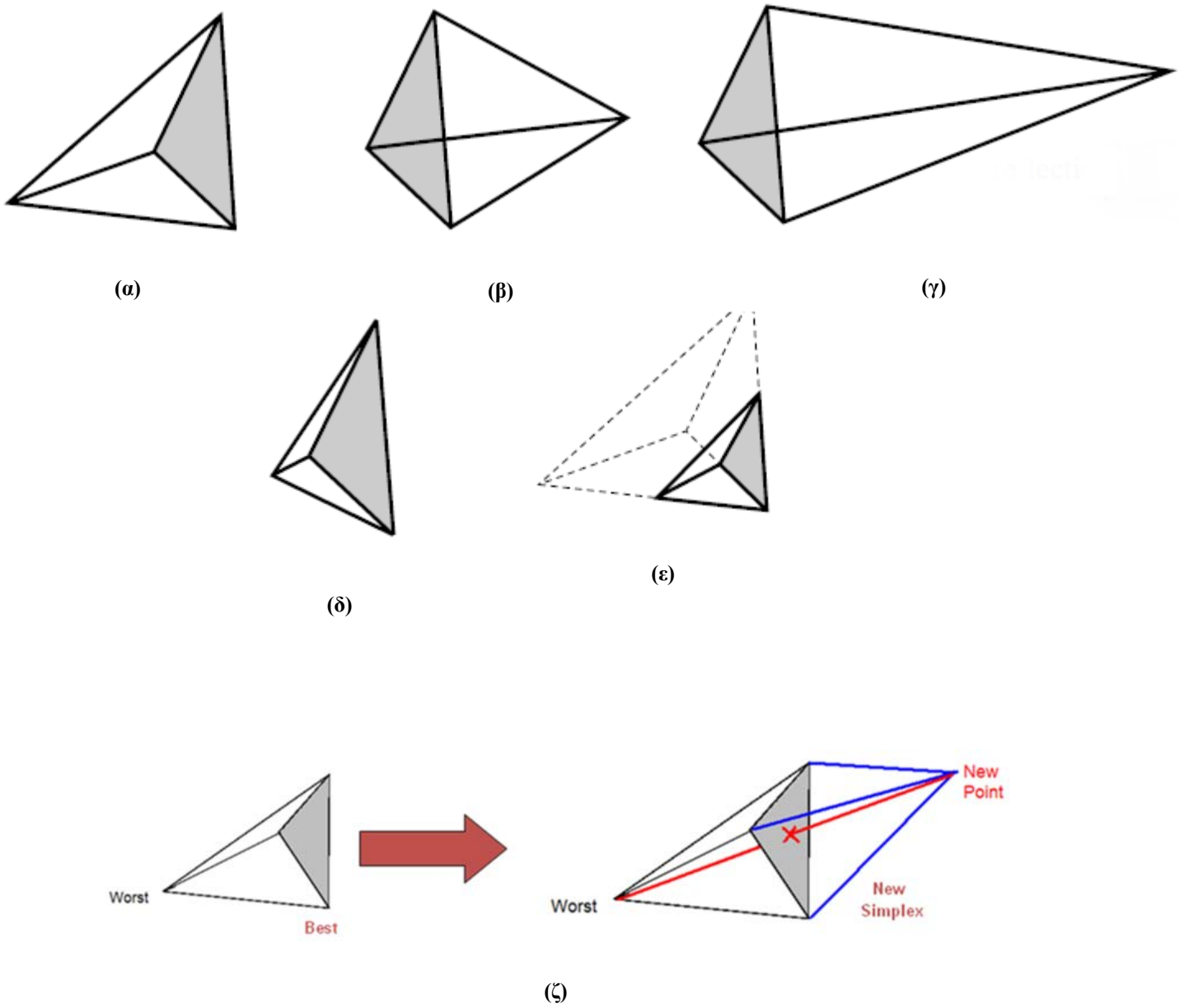
Το κύριο πλεονέκτημα της ASA είναι ότι μπορεί να χρησιμοποιηθεί σε οποιοδήποτε πρόβλημα και χρησιμοποιείται κυρίως όταν η εύρεση της καλύτερη τιμής είναι πιο σημαντική από την εύρεση μιας «γρήγορης» λύσης που δεν είναι και η βέλτιστη.

3.2.2 Downhill Simplex

Σαν "simplex" αποκαλείται ένα γεωμετρικό σχήμα που στις N διαστάσεις έχει $N+1$ γωνίες και τις επιφάνειες και ακμές που οι γωνίες ορίζουν. Στις 2 διαστάσεις το simplex είναι ένα τρίγωνο, ενώ στις 3 διαστάσεις ένα τετράεδρο. Η μέθοδος αρχικοποιείται με $N+1$ σημεία. Αν θεωρηθεί ένα σημείο αρχικοποίησης P_o, τότε τα υπόλοιπα N σημεία προκύπτουν ως εξής:

$$P_i = P_o + \lambda e_i$$

όπου e_i είναι N μοναδιαία διανύσματα, και η παράμετρος λ μπορεί να λάβει την ίδια ή διαφορετική τιμή για κάθε διάνυσμα. Η μέθοδος ξεκινάει με μια σειρά βημάτων όπου η γωνία του simplex με τη μεγαλύτερη τιμή μετακινείται κάθε φορά διαπερνώντας την απέναντι έδρα προς ένα σημείο με μικρότερη τιμή. Αυτά τα βήματα ονομάζονται αντανακλάσεις και πραγματοποιούνται με τέτοιο τρόπο ώστε να διατηρείται ο όγκος του simplex. Όπου είναι εφικτό, η μέθοδος εκτείνει τη γωνία του simplex με τέτοιο τρόπο ώστε να εκτελεί μεγαλύτερα βήματα. Όταν βρίσκεται σε επίπεδες περιοχές (όπου τα σημεία έχουν παρόμοιες τιμές και απουσιάζουν τοπικά ελάχιστα), η μέθοδος συμπτύσσεται προς μια αντίθετη κατεύθυνση και επιχειρεί να ξεφύγει από την επίπεδη περιοχή. Όταν ένα σημείο ελαχίστου βρίσκεται ανάμεσα σε σημεία με μεγάλες τιμές, το simplex δεν μπορεί να το προσεγγίσει διατηρώντας τον όγκο του. Τότε συμπτύσσεται προς το καλύτερο σημείο του, το σημείο δηλαδή με τη χαμηλότερη τιμή. Στο simplex έχει αποδοθεί ο χαρακτηρισμός "αμοιβάδα" που περιγράφει την ιδιαιτερότητα της κίνησής του. Οι βασικές κινήσεις του περιγράφονται στη Σχήμα 3.8.



Σχήμα 3.8 Πιθανές εκδοχές ενός βήματος της μεθόδου Downhill Simplex. (α) Το simplex για τρισδιάστατες εφαρμογές, στην αρχή του αλγόριθμου, (β) μία αντανάκλαση προς την αντίθετη κατεύθυνση του προηγούμενου μεγίστου, (γ) αντανάκλαση και επέκταση μακριά από το προηγούμενο μέγιστο, (δ) σύμπτυξη σε μια διάσταση από το μέγιστο και (ε) σύμπτυξη προς όλες τις κατευθύνσεις προς το χαμηλότερο σημείο. Μια κατάλληλη αλληλουχία τέτοιων βημάτων οδηγεί πάντα στην εύρεση του ελαχίστου μιας συνάρτηση. (ζ) αντανάκλαση προς την αντίθετη κατεύθυνση, του νέου καλύτερου σημείου

Σαν συνθήκη τερματισμού του αλγόριθμου μπορεί να οριστεί μία ελάχιστη απόσταση για το μέτρο του διανύσματος της μετακίνησης του simplex ή ο μέγιστος αριθμός αξιολογήσεων.

Όπως φαίνεται στην απεικόνιση το simplex ψάχνει μέσω των βημάτων του ένα ελάχιστο, ωστόσο μπορεί να εγκλωβιστεί σε τοπικό ελάχιστο και να μην βρει το ολικό ελάχιστο.

3.2.3 Hooke-Jeeves

Αυτή η τεχνική χρησιμοποιεί ένα συνδυασμό αντικειμενικών και penalty περιορισμών σαν αντικειμενική συνάρτηση *f(x)*. Η μέθοδος δεν προϋποθέτει να είναι διαφορίσιμη η αντικειμενική συνάρτηση, επειδή δεν χρησιμοποιούνται παράγωγοι της *f(x)*. Αντ ' αυτού, ο αλγόριθμος εξετάζει τα σημεία κοντά στο τωρινό σημείο διαταράσσοντας τις μεταβλητές σχεδιασμού, έναν άξονα κάθε φορά, μέχρι να βρεθεί ένα βελτιωμένο σημείο. Στο Isight η τεχνική έχει μια παράμετρο σύγκλισης, με την οποία μπορεί να καθοριστεί ο αριθμός των αξιολογήσεων που χρειάζεται για μεγαλύτερη πιθανότητα σύγκλισης.

Πιο αναλυτικά η Hooke-Jeeves είναι μια τεχνική αναζήτησης μοτίβων και περιλαμβάνει δύο είδη *κινήσεων*:

- Μια *διερευνητική αναζήτηση*. Αυτή είναι μια τοπική αναζήτηση, η οποία ψάχνει για να βελτιώσει την κατεύθυνση στην οποία θα κινηθεί ο αλγόριθμος.
- Μια *κίνηση μοτίβου*. Αυτή είναι μια μεγαλύτερη αναζήτηση στην βελτιωμένη κατεύθυνση.

Διερευνητική Αναζήτηση

Η κύρια ιδέα της *κίνησης* είναι να βρει μια βελτιωμένη κατεύθυνση, όχι απαραίτητα την βέλτιστη. Αυτό γίνεται με την διατάραξη του τρέχοντος σημείου κατά πολύ μικρές ποσότητες σε κάθε μια από τις μεταβλητές κατεύθυνσης και παρατηρώντας εάν η τιμή της αντικειμενικής συνάρτησης βελτιώνεται ή χειροτερεύει. Δίνεται η δυνατότητα να καθοριστεί το μέγεθος της διαταραχής καθώς και την penalty τιμή σε κάθε αξιολόγηση.

Εάν κάποια διαταραχή επιτύχει στο να βελτιώσει την τιμή της αντικειμενικής συνάρτησης, τότε η διερευνητική αναζήτηση έχει επιτύχει και η βελτιωμένη κατεύθυνση δίνεται από το διάνυσμα μεταξύ του αρχικού σημείου $x^{(0)}$ και του τελικού σημείου $x^{(1)}$. Ύστερα, από την εύρεση της βελτιωμένης κατεύθυνσης ακολουθεί η κίνηση μοτίβου.

Κίνηση Μοτίβου

Το μόνο που χρειάζεται η κίνηση μοτίβου είναι δύο σημεία, το τρέχον $x^{(0)}$ και κάποιο άλλο $x^{(1)}$ που έχει καλύτερη τιμή αντικειμενικής συνάρτησης. Αυτό δίνει στην κίνηση μοτίβου μια βελτιωμένη κατεύθυνση για να προχωρήσει. Ένα νέο σημείο $x^{(2)}$ παράγεται, μετακινώντας από το $x^{(0)}$ μέσω του $x^{(1)}$ όπως φαίνεται παρακάτω:

$$x\hat{}((2)) = x^{(0)} + a[x\hat{}((1)) - x\hat{}((0))]$$

Όπου α είναι ένας θετικός *παράγοντας επιτάχυνσης* ο οποίος πολλαπλασιάζει το μήκος του διανύσματος της βελτιωμένης κατεύθυνσης που δίνεται από: $x^{(1)} - x^{(0)}$. Ο αλγόριθμος σταματάει με βάση το κριτήριο του τερματισμού ή τον αριθμό των αξιολογήσεων.

Στο Σχήμα 3.9 παρουσιάζεται το τρόπος με τον οποίο 'κινείται' ο αλγόριθμος.

(α) **(β)** **(γ)**

Σχήμα 3.9. Διαδικασία του αλγορίθμου Hooke-Jeeves. (α) Διερευνητική κίνηση. Διατάραξη του X1 με βάση το μέγεθος του βήματος. Αποθήκευση της καλύτερης τιμής. (β) Διερευνητική κίνηση. Από το προηγούμενο καλύτερο σημείο, διατάραξε το X2 με βάση το μέγεθος του βήματος. Επανάλαβε: Συνέχισε την διαδικασία της διερευνητικής αναζήτησης για όλες τις μεταβλητές (X1 .. Xn). (γ) Κίνηση μοτίβου. Μόλις όλες οι διερευνητικές κινήσεις ολοκληρωθούν, να εξαχθεί το αρχικό σημείο βάσης μέσω του καλύτερου σημείου για να δημιουργηθεί νέο σημείο βάσης. Πίσω στην διερευνητική αναζήτηση.

Χαρακτηριστικά των Downhill Simplex και Hooke Jeeves

Οι αλγόριθμοι Downhill Simplex και Hooke Jeeves είναι τεχνικές αναζήτησης μοτίβου (Pattern Search). Οι οποίες μπορούν να χρησιμοποιηθούν για όλους τους τύπους των μεταβλητών όπως πραγματικές, ακέραιες και διακριτές. Επιπλέον με την κατάλληλη ρύθμιση του αρχικού βήματος δίνεται η δυνατότητα να απεγκλωβιστούν από τοπικά ελάχιστα. Μπορούν να εφαρμοστούν και σε μη συνεχή προβλήματα και έχουν εξαιρετική απόδοση αναζήτησης σε τοπικές περιοχές, στον χώρο σχεδιασμού. Ωστόσο, για να ξεπεραστεί το κυριότερο τους πρόβλημα, που είναι ο εγκλωβισμός σε τοπικό ακρότατο απαιτείται μεγάλος αριθμός αξιολογήσεων, γι' αυτό το λόγο μπορεί να γίνουν αρκετά χρονοβόρες. Επιπλέον, το υπολογιστικό κόστος εξαρτάται από τον αριθμό των μεταβλητών, όσο μεγαλύτερος ο αριθμός των μεταβλητών τόσο μεγαλύτερος ο αριθμός των αξιολογήσεων που απαιτούνται για ικανοποιητικά αποτελέσματα. Τέλος επειδή ξεκινάνε από ένα τυχαίο σημείο υπάρχει πιθανότητα να εγκλωβιστούν σε τοπικό ακρότατο νωρίς κατά την εκτέλεση, γι' αυτό πολλές φορές συνιστάται να επαναλαμβάνονται οι τεχνικές αυτές και το αρχικό σημείο να είναι τυχαίο.

3.2.4 Large Scale Generalized Reduced Gradient (LSGRG)

Η LSGRG τεχνική χρησιμοποιεί τον reduced gradient αλγόριθμο για την επίλυση περιορισμένων μη-γραμμικών προβλημάτων βελτιστοποίησης. Χρησιμοποιεί μια κατεύθυνση αναζήτησης έτσι ώστε όλοι οι ενεργοί περιορισμοί να παραμένουν ενεργοί για μια πολύ μικρή κίνηση προς αυτήν την κατεύθυνση. Ο αλγόριθμος λειτουργεί ακολουθώντας τους περιορισμούς στον χώρο σχεδιασμού έναν την φορά. Η σειρά επιλογής των περιορισμών μπορεί να επηρεάσει την διαδικασία εκτέλεσης. Ο generalized reduced αλγόριθμος είναι μια επέκταση μιας προηγούμενης reduced gradient μεθόδου, η οποία επιλύει προβλήματα με ίσους περιορισμούς μόνο. Η τροποποίηση περιλαμβάνει την πρόσθεση μιας χαλαρής μεταβλητής σε κάθε άνισο περιορισμό.

Η βασική ιδέα της generalized gradient μεθόδου για κάθε περιορισμό ισότητας, είναι ο προσδιορισμός μιας εξαρτημένης μεταβλητής σχεδιασμού, μειώνοντας έτσι τον συνολικό αριθμό των εξαρτημένων μεταβλητών σχεδιασμού. Ο GRG αλγόριθμος πρώτα βρίσκει μια κατεύθυνση αναζήτησης στις μεταβλητές απόφασης, δηλαδή τις ανεξάρτητες μεταβλητές και μετά για κάθε προτεινόμενη κίνηση προς την κατεύθυνση αναζήτησης, ενημερώνει το διάνυσμα των εξαρτημένων μεταβλητών έτσι ώστε οι περιορισμοί να ικανοποιούνται.

3.2.5 MISQP - Mixed Integer Sequential Quadratic Programming

Είναι μια μέθοδος που βασίζεται σε περιοχή εμπιστοσύνης για την επίλυση προβλημάτων που περιλαμβάνουν ακέραιες και διακριτές μεταβλητές. Η τεχνική υποθέτει ότι η αντικειμενική συνάρτηση και οι περιορισμοί είναι συνεχείς και διαφορίσιμοι. Επιπλέον υποθέτει ότι η αντικειμενική συνάρτηση και οι περιορισμοί είναι smooth σε σχέση με τις ακέραιες μεταβλητές.

3.2.6 Multi-Objective Particle Swarm Technique

Η τεχνική μιμείται την κοινωνική συμπεριφορά ζωικών ομάδων όπως τα σμήνη πουλιών ή τα κοπάδια ψαριών. Η διαδικασία της αναζήτησης βέλτιστου σημείου σχεδιασμού μοιάζει με την διαδικασία εύρεσης τροφής αυτών των οργανισμών. Η particle swarm τεχνική είναι μια διαδικασία αναζήτησης με βάση των πληθυσμό όπου τα άτομα (ονομαζόμενα σωματίδια) συνεχώς αλλάζουν θέση (κατάσταση) μέσα στην περιοχή αναζήτησης. Με άλλα λόγια, τα σωματίδια ‘πετούν’ τριγύρω στον χώρο σχεδιασμού ψάχνοντας την καλύτερη θέση. Η καλύτερη θέση που βρίσκει ένα σωματίδιο και τα γειτονικά του μαζί με την τρέχουσα ταχύτητα και την αδράνεια του χρησιμοποιούνται για να καθορίσουν την νέα θέση του σωματιδίου. Η κατάσταση ενός

σωματιδίου είναι η τιμή για όλες τις μεταβλητές σχεδιασμού στο πρόβλημα της βελτιστοποίησης και η ταχύτητα του στον χώρο σχεδιασμού. Κάθε κίνηση παράγει και μία νέα γενιά.

Ο αλγόριθμος ξεκινάει με την παραγωγή τυχαίων σωματιδίων που καθιστούν τον αρχικό πληθυσμό, και σε κάθε σωματίδιο δίνεται μια αρχική ταχύτητα. Σε κάθε αξιολόγηση , το σωματίδιο κινείται χρησιμοποιώντας τις δύο καλύτερες τιμές: 1) την καλύτερη θέση που έχει υπάρξει το συγκεκριμένο σωματίδιο ($\bar{P}_{Best}$) και 2) την καλύτερη θέση που έχει βρεθεί από όλα τα σωματίδια ανάμεσα σε όλες τις γενιές μέχρι στιγμής ($\bar{G}_{Best}$).

Έστω $\bar{X}_i^j$ και $\bar{V}_i^j$, τα οποία είναι η θέση και η ταχύτητα του σωματιδίου *i* κατά την διάρκεια της επανάληψης, *j* η ταχύτητα και η θέση του σωματιδίου στην επόμενη επανάληψη *(j+1)*, υπολογίζεται χρησιμοποιώντας τις ακόλουθες εξισώσεις:

$$\bar{V}_i^{j+1} = I\,\bar{V}_i^j + r_1 P_{Incr}(\bar{P}_{Best} - \bar{X}_i^j) + r_2 G_{Incr}(\bar{G}_{Best} - \bar{X}_i^j)$$
$$\bar{X}_i^{j+1} = \bar{X}_i^j + \bar{V}_i^{j+1}$$

Όπου, r_1 και r_2 δύο τυχαίοι αριθμοί μεταξύ του 0 και 1, *I* είναι η αδράνεια του σωματιδίου και P_{Incr}, G_{Incr} οι μέγιστες αυξήσεις εξαιτίας particle-best και global-best. P_{Incr} και G_{Incr} ονομάζονται και «Learning Rates» ή «Learning Factors».

Στην περίπτωση πολλαπλών στόχων κάθε σημείο του Pareto είναι υποψήφιο για αρχηγός. Η κύρια διαφορά μεταξύ τους, είναι πως η επιλογή του $\bar{G}_{Best}$ γίνεται από ένα εξωτερικό αρχείο του ενημερωμένου μετώπου Pareto, που περιέχει τα νέα άτομα. Η επιλογή από το αρχείο γίνεται με μια διαδικασία επιλογής ρουλέτας, που ωστόσο κάθε άτομο έχει ξεχωριστή βαρύτητα ανάλογα με την τοπική πυκνότητα του μετώπου. Το Isight χρησιμοποιεί επίσης έναν παράγοντα που διαταράζει την ταχύτητα μικρού αριθμού σωματιδίων έτσι ώστε να καλυφθεί πιο καλά ο χώρος σχεδιασμού. Μια μεγάλη τιμή αδράνειας επιτρέπει στο σωματίδιο να εξερευνήσει ένα μεγαλύτερο τμήμα του χώρου σχεδιασμού αυξάνοντας παράλληλα τον αριθμό των επαναλήψεων για σύγκλιση στο βέλτιστο σημείο.

Η τεχνική έχει χρησιμοποιηθεί και στο παρελθόν για την βελτιστοποίηση της διαδρομής ανόδου του αεροσκάφους. Όπως επισημαίνεται και στην [24] η χρήση του αλγόριθμου, δίνει την δυνατότητα της απλοποίησης της διαδικασίας αλληλοεπίδρασης των μοντέλων προσομοίωσης επιδόσεων του κινητήρα/αεροσκάφους. Τα αποτελέσματα της μελέτης είναι αρκετά ικανοποιητικά και παρόμοια διαδικασία χρησιμοποιείται και στο Isight.

3.2.7 Multi-Island Genetic Algorithm

Όπως και σε άλλους γενετικούς αλγόριθμους, κάθε σημείο σχεδιασμού αντιλαμβάνεται σαν ένα άτομο το οποίο έχει μια συγκεκριμένη τιμή καταλληλότητας

(fitness), με βάση της τιμής της αντικειμενικής συνάρτησης και την ποινή περιορισμού. Όπως είναι προφανές ένα άτομο με καλύτερη τιμή αντικειμενικής συνάρτησης και καλύτερη τιμή ποινής έχει υψηλότερη τιμή καταλληλότητας.

Το κύριο χαρακτηριστικό του αλγόριθμου είναι ότι διαχωρίζει τον πληθυσμό σε διάφορα νησιά και εκτελεί τις παραδοσιακές γενετικές διαδικασίες σε κάθε νησί ξεχωριστά. Έπειτα κάποια άτομα επιλέγονται από κάθε νησί και μεταναστεύουν σε άλλο σε κάθε γενιά με την διαδικασία επιλογής ρουλέτας.

Κατά την διαδικασία της εξέλιξης δύο παράμετροι ελέγχουν την μετανάστευση:

- Διάστημα μετανάστευσης (migration interval) όπου καθορίζεται ο αριθμός των γενιών μεταξύ κάθε μετανάστευσης και
- Ρυθμός μετανάστευσης (migration rate) όπου καθορίζει το ποσοστό των ατόμων που θα μεταναστεύσουν από κάθε νησί.

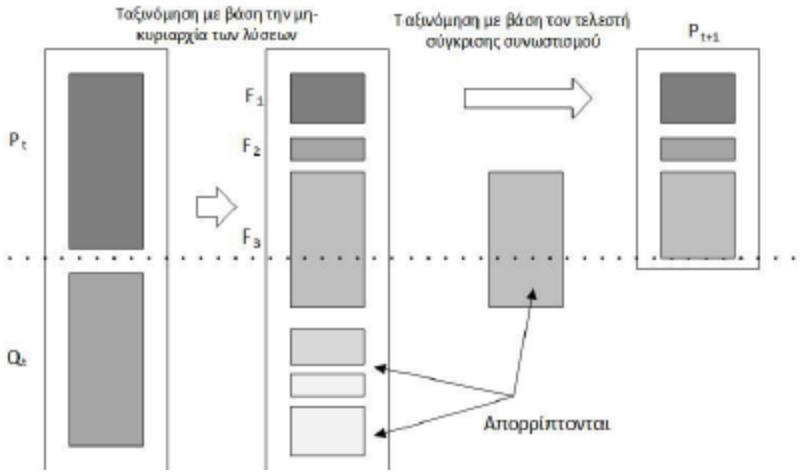
Στην περίπτωση των γενετικών αλγόριθμων που έχουν ως βάση το Island Model έχει παρατηρηθεί, ότι έχοντας πολλαπλούς υποπληθυσμούς βοηθάει στο να διατηρηθεί η γενετική ποικιλομορφία, καθώς κάθε νησί μπορεί, να ακολουθήσει διαφορετική τροχιά στον χώρο αναζήτησης.

Ο αλγόριθμος έχει χρησιμοποιηθεί στο παρελθόν με επιτυχία μέσω του Isight. Ένα χαρακτηριστικό παράδειγμα είναι στην [25], όπου με την χρήση του Isight εφαρμόσθηκε ο αλγόριθμος MIGA για την βελτιστοποίηση του διαχωριστικού συστήματος ενός δορυφόρου. Στο Isight ενσωματώθηκε το πρόγραμμα ADAMS ως Simcode Component και το Isight με την χρήση του Optimization Component, έτρεξε το πρόβλημα της βελτιστοποίησης με αλγόριθμο τον MIGA. Ο αλγόριθμος δεν εγκλωβίστηκε σε τοπικά ακρότατα και τα εξαιρετικά αποτελέσματα, επαληθεύτηκαν και από πειραματικά.

3.2.8 Non-dominated Sorting Genetic Algorithm (NSGA-II)

Ο NSGA-II αποτελεί έναν εξελικτικό αλγόριθμο, ο οποίος βασίζεται στην λογική των μη-κυριαρχούμενων λύσεων (non-dominated sorting algorithms) σε επίπεδα κυριαρχίας ή αλλιώς σε διαφορετικά μέτωπα λύσεων. Ο αλγόριθμος NSGA-II αποτελεί μια βελτιωμένη εκδοχή του NSGA [26] που προτάθηκε από τους Srinivas και Deb. Ο NSGA-II έχει αρχείο Pareto, P και πληθυσμό Q για γενετική αναζήτηση. Οι απόγονοι προκύπτουν χρησιμοποιώντας δυαδικό τουρνουά επιλογής που βασίζεται σε τελεστή σύγκρισης του συνωστισμού. Ο αριθμός των ατόμων, N, στα δύο μεγέθη είναι ίδιος. Οι διαδικασίες της διασταύρωσης και της μετάλλαξης γίνονται στον πληθυσμό Q. Το αρχείο για την επόμενη γενιά P^{t+1} επιλέγεται από $P^t \cup Q^t$ και ο νέος πληθυσμός Q^{t+1} επιλέγεται από P^{t+1}. Η διαδικασία επιλογής βασίζεται σε δυο κυρίως μηχανισμούς , ‘μη-κυριαρχούμενη ταξινόμηση’ και ‘crowding distance sorting’.

Αρχικά, όπως και στους άλλους αλγορίθμους, δημιουργείται ένας τυχαίος αρχικός πληθυσμός μέσα στα προκαθορισμένα όρια του προβλήματός μας. Σε κάθε λύση αντιστοιχίζεται μια τιμή, η οποία έχει να κάνει με την κατάταξή της σε κάποιο μέτωπο, η καλύτερη κατάταξη που μπορεί να πάρει η λύση είναι η τιμή 1 και από την τιμή 2 και πέρα μειώνεται και η ποιότητα της λύσης. Στην αρχή χρησιμοποιούνται οι ήδη γνωστές μέθοδοι δημιουργίας του επόμενου πληθυσμού όπως είναι ο δυαδικός διαγωνισμός (binary tournaments), η μετάλλαξη κλπ. Μετά την αρχικοποίηση του πληθυσμού η διαδικασία διαφέρει, καθώς στον NSGA-II συγκρίνεται ο υπάρχων πληθυσμός με τις καλύτερες μη-κυριαρχούμενες λύσεις οι οποίες έχουν βρεθεί προηγουμένως. Στο Σχήμα 3.11 δίνεται παραστατικά όλη η διαδικασία που εκτελεί ο NSGA-II.



Σχήμα 3.10 Η διαδικασία που ακολουθείται κατά την εκτέλεση του αλγορίθμου NSGA-II

Ο αλγόριθμος NSGA-II έχει την ικανότητα μέσα σε λίγους μόνο κύκλους να δημιουργεί ένα μέτωπο Pareto το οποίο είναι πολύ κοντά στο επιθυμητό. Βέβαια, λόγω της πολυπλοκότητας που εισάγει μέσω των υπεράριθμων συγκρίσεων υστερεί ως προς την αποτελεσματικότητα συγκριτικά με άλλους αλγορίθμους .

3.2.9 Neighborhood Cultivation Genetic Algorithm (NCGA)

Στον Neighborhood Cultivation Genetic Algorithm κάθε αντικειμενική παράμετρος αντιμετωπίζεται ξεχωριστά. Ωστόσο σε αντίθεση με τους άλλους γενετικούς αλγόριθμους, όπου κατά την διασταύρωση οι γονείς επιλέγονται τυχαία από τον πληθυσμό, στην διασταύρωση γειτονίας οι γονείς είναι άτομα τα οποία έχουν κοντινές τιμές με έναν από τους στόχους της βελτιστοποίησης. Στο τέλος της βελτιστοποίησης δημιουργείται μέτωπο Pareto. Κάθε σχέδιο έχει τον καλύτερο συνδυασμό αντικειμενικών τιμών και βελτιώνοντας έναν στόχο είναι αδύνατο να μην θυσιαστούν ένας ή περισσότεροι στόχοι.

Η ιδέα για την διασταύρωση γειτονίας προέρχεται από το Island Model κάποιων γενετικών αλγόριθμων. Στον NCGA ο πληθυσμός ουσιαστικά χωρίζεται σε υποπληθυσμούς ανάλογα με τις τιμές που έχουν τα άτομα σε κάθε στόχο. Η διασταύρωση γίνεται μεταξύ των ατόμων του κάθε υποπληθυσμού. Σύμφωνα με πλήθος ερευνών έχει βρεθεί ότι, η διασταύρωση όμοιων ατόμων παράγει καλύτερα αποτελέσματα από την διασταύρωση εντελώς διαφορετικών ατόμων. Τα άτομα διαχωρίζονται μεταξύ τους ανάλογα με την τιμή μιας από τις αντικειμενικές συναρτήσεις. Η αντικειμενική συνάρτηση που χρησιμοποιείται για την ταξινόμηση, σε κάθε γενιά αλλάζει. Ωστόσο, με την πλήρη ταξινόμηση υπάρχει κίνδυνος να χαθεί η ποικιλομορφία στην λύση, δηλαδή ο αλγόριθμος να γίνει πολύ ελιτιστικός. Γι' αυτό τον λόγο, περίπου 5% διαταραχή εφαρμόζεται στα ταξινομημένα άτομα προτού αρχίσει η διαδικασία της διασταύρωσης.

Η διαφορά του αλγόριθμου NCGA με τον NSGA-II βρίσκεται στην μέθοδο επιλογής του πληθυσμού αναζήτησης $Q(t+1)$ από το αρχείο του Pareto $P(t+1)$. Στον NCGA η μέθοδος περιλαμβάνει την αντιγραφή του αρχείου $P(t+1)$, N ατόμων στο Q. Ο NSGA-II επιλέγει άτομα μέσω τουρνουά δυαδικής επιλογής με επανάληψη. Η μέθοδος αυτή, μπορεί να προκαλέσει έλλειψη ποικιλομορφίας και υπάρχει κίνδυνος εγκλωβισμού της βελτιστοποίησης σε τοπικό ακρότατο.

3.2.10 Evolutionary Optimization Algorithm (EVOL)

Ο εξελικτικός αλγόριθμος βελτιστοποίησης (Evol) είναι μια στρατηγική εξέλιξης βασισμένη στα έργα του Rechenberg και του Schwefel που μεταλλάσσει τους σχεδιασμούς προσθέτοντας μια κανονικά κατανεμημένη τυχαία τιμή σε κάθε μεταβλητή σχεδιασμού. Η ισχύς της μετάλλαξης, δηλαδή η τυπική απόκλιση της κανονικής κατανομής, είναι αυτοπροσαρμοσμένη και μεταβάλλεται κατά τη διάρκεια του βρόχου βελτιστοποίησης. Ο αλγόριθμος έχει ρυθμιστεί στο Isight για την αποτελεσματική επίλυση προβλημάτων σχεδίασης με χαμηλό αριθμό μεταβλητών και με κάποιο θόρυβο στον χώρο σχεδιασμού.

Ο αλγόριθμος EVOL έχει τα ακόλουθα χαρακτηριστικά:

- Διακριτοποίηση χώρου σχεδιασμού. Ο αλγόριθμος θεωρεί μόνο διακεκριμένα σημεία σχεδιασμού.
- Repeat calculation check. Ο αλγόριθμος διασφαλίζει ότι δεν υπάρχουν δύο ίδια σημεία σχεδιασμού που υποβάλλονται για αξιολόγηση.
- Διόγκωση τυπικής απόκλισης. Εάν εντοπιστούν μόνο επαναληπτικοί υπολογισμοί μετά την τυχαιοποίηση, ο αλγόριθμος αυξάνει την τυπική απόκλιση της τυχαίας κανονικής κατανομής.
- Διαδοχική αναζήτηση μεταβλητών. Ο αλγόριθμος μπορεί να μεταβάλλει είτε όλες τις μεταβλητές σχεδιασμού ταυτόχρονα είτε μία μεταβλητή τη φορά.

- Παράλληλη εκτέλεση. Ο αλγόριθμος έχει παραλληλισθεί ώστε να παράγει *n* παιδιά όταν υπάρχουν διαθέσιμες *n* παράλληλες πηγές και να χρησιμοποιήσει το καλύτερο από τα *n* παιδιά για να προωθήσει την επόμενη επανάληψη.

3.2.11 Archive based Micro Genetic Algorithm (AMGA)

Ο AMGA είναι ένας εξελικτικός αλγόριθμος βελτιστοποίησης και βασίζεται σε φορείς γενετικής διαφοροποίησης για τη δημιουργία νέων λύσεων. Η διαδικασία παραγωγής που αναπτύσσεται σε αυτόν τον αλγόριθμο μπορεί να ταξινομηθεί ως γενεσιουργική επειδή κατά τη διάρκεια μίας συγκεκριμένης επανάληψης, γενιάς, μόνο οι λύσεις που δημιουργήθηκαν πριν από αυτή την επανάληψη συμμετέχουν στη διαδικασία επιλογής. Ο αλγόριθμος, ωστόσο, παράγει ένα μικρό αριθμό νέων λύσεων, η συνιστώμενη τιμή είναι δύο λύσεις ανά επανάληψη. Ο αλγόριθμος λειτουργεί με ένα μικρό μέγεθος πληθυσμού, η συνιστώμενη τιμή είναι 4 και διατηρεί ένα εξωτερικό αρχείο με καλές λύσεις. Σε κάθε επανάληψη, δημιουργείται ένας μικρός αριθμός λύσεων χρησιμοποιώντας χειριστές γενετικών παραλλαγών. Ο αλγόριθμος αναφέρεται ως μικρογενετικός αλγόριθμος βασισμένος σε αρχείο (AMGA), λόγω του ότι λειτουργεί με πολύ μικρό μέγεθος πληθυσμού και χρησιμοποιεί ένα αρχείο για να διατηρεί το ιστορικό αναζήτησης. Συνιστάται η χρήση ενός μεγάλου μεγέθους για το αρχείο και τα καλύτερα αποτελέσματα επιτυγχάνονται εάν το μέγεθος του αρχείου είναι ίδιο με τον αριθμό των επιτρεπόμενων αξιολογήσεων λειτουργίας, στην ουσία ο αλγόριθμος να αποθηκεύει το πλήρες ιστορικό αναζήτησης του. Το μέγεθος του αρχείου καθορίζει την υπολογιστική πολυπλοκότητα του αλγορίθμου. Ωστόσο, για υπολογιστικά δαπανηρά προβλήματα βελτιστοποίησης, ο πραγματικός χρόνος που απαιτείται από τον αλγόριθμο είναι αμελητέος. Ο γονικός πληθυσμός ενημερώνεται χρησιμοποιώντας την αρχειοθέτηση και η επιλογή δυαδικών τουρνουά γίνεται στον πληθυσμό των γονέων, για τη δημιουργία του πληθυσμού που θα διασταυρωθεί.

Κάθε αντικειμενική παράμετρος αντιμετωπίζεται ξεχωριστά. Χρησιμοποιεί μηχανισμό καταλληλότητας τριών επιπέδων. Η πρώτη βαθμίδα καταλληλότητας καθορίζεται με βάση το επίπεδο κυριαρχίας μιας λύσης στον πληθυσμό. Η καταλληλότητα δεύτερης βαθμίδας βασίζεται στη συμβολή της λύσης στο ιστορικό αναζήτησης του αλγορίθμου και η καταλληλότητα τρίτης βαθμίδας λαμβάνει υπόψη την ποικιλομορφία της λύσης. Μέχρι το τέλος της διαδικασίας βελτιστοποίησης κατασκευάζεται ένα σύνολο παραστάσεων όπου κάθε σχεδιασμός έχει τον «καλύτερο» συνδυασμό αντικειμενικών τιμών και η βελτίωση ενός στόχου είναι αδύνατη χωρίς να θυσιάζεται ένας ή περισσότεροι από τους άλλους στόχους.

Οι Van der Velden et al. [27], σύγκριναν τους αλγόριθμους NCGA, NSGA-II και AMGA σε πέντε πολυκριτηριακά προβλήματα, το καθένα για ελαχιστοποίηση δύο στόχων. Τα αποτελέσματα έδειξαν ότι ο AMGA έχει συνολικά την καλύτερη απόδοση. Παρόλο που η πολυπλοκότητα του αλγόριθμου είναι μεγαλύτερη από τους άλλους δύο ο υπολογιστικός χρόνος δεν είχε μεγάλη διαφορά.

3.2.12 Sequential Quadratic Programming (NLPQL)

Η τεχνική υποθέτει ότι η αντικειμενική συνάρτηση και οι περιορισμοί είναι συνεχώς διαφορίσιμοι. Στην τεχνική NLPQL, η ιδέα είναι να δημιουργηθεί μια ακολουθία τετραγωνικών υποπρογραμμάτων, που λαμβάνονται από μια τετραγωνική προσέγγιση της συνάρτησης Lagrangian και μια γραμμικοποίηση των περιορισμών. Οι πληροφορίες δεύτερης τάξης ενημερώνονται από την quasi-Newton formula και η μέθοδος σταθεροποιείται με μια επιπλέον αναζήτηση γραμμής.

Ένα ιδιαίτερο πλεονέκτημα του NLPQL είναι η εύκολη χρήση του, καθώς ο χρήστης επιλέγει μόνο τον αριθμό των επαναλήψεων και την επιθυμητή ακρίβεια. Η χρήση του είναι διαδεδομένη για την επίλυση μη-γραμμικών προβλημάτων βελτιστοποίησης όπου η συνάρτηση κόστους και οι περιορισμοί είναι διαφορίσιμοι και οι παράγωγοι μπορούν να εκτιμηθούν με επαρκή ακρίβεια.

3.2.13 Multifunction Optimization System Tool (MOST)

Η τεχνική επιλύει, αρχικά, το δεδομένο πρόβλημα σχεδιασμού σαν να ήταν καθαρά συνεχές πρόβλημα, χρησιμοποιώντας τον διαδοχικό τετραγωνικό προγραμματισμό για να εντοπίσει μια αρχική κορυφή. Αν όλες οι μεταβλητές είναι πραγματικοί αριθμοί, η βελτιστοποίηση σταματάει εκεί. Εάν υπάρχουν ακέραιες (ή διακριτές) μεταβλητές, η MOST θα χρησιμοποιήσει την συνεχή λύση ως αρχικό σημείο για τον τροποποιημένο branch-and-bound αλγόριθμό του. Σε αυτό το στάδιο, οι ακέραιες μεταβλητές μειώνονται μια την φορά. Το μειωμένο συνεχές πρόβλημα βελτιστοποίησης επιλύεται για κάθε μεταβλητή που απορρίφθηκε, ρυθμίζοντας τις τιμές τους στα ακέραια επίπεδα επάνω και κάτω των βέλτιστων τιμών. Όσο η διαδικασία προχωρά η τεχνική εστιάζει στις τιμές διαδοχικών μη πραγματικών παραμέτρων, μέχρι όλα τα όρια να ικανοποιηθούν.

3.2.14 Pointer

Ο αλγόριθμος είναι μια αυτοματοποιημένη τεχνική η οποία ανάλογα με την μορφή του προβλήματος χρησιμοποιεί μια πληθώρα άλλων αλγορίθμων για την αντιμετώπισή του. Ο αλγόριθμος απευθύνεται σε μη ειδικούς χρήστες που δεν γνωρίζουν από βελτιστοποίηση. Ο Pointer περιέχει τους αλγόριθμος downhill simplex, SQP και διάφορους γενετικούς αλγόριθμους τους οποίους χρησιμοποιεί για την επίλυση του εκάστοτε προβλήματος. Ο αλγόριθμος ουσιαστικά αυτό που κάνει είναι η επιλογή του κατάλληλου αλγόριθμου βελτιστοποίησης ανάλογα με την δομή του προβλήματος με βάσει κάποιο κριτήριο το οποίο δεν είναι γνωστό. Γι’ αυτό τον λόγο και επειδή δεν υπάρχει γνώση πως ρυθμίζονται οι αλγόριθμοι που εφαρμόζει δεν θα γίνει η χρήση του.

3.2.15 Modified Method of Feasible Directions (MMFD)

Ο αλγόριθμος MMFD βασίζεται σε μια τροποποίηση της αρχικής μεθόδου του εφικτού αλγορίθμου κατευθύνσεων. Η τροποποίηση παρέχει μια κατεύθυνση αναζήτησης που ακολουθεί τα όρια ενεργών περιορισμών (αντί να μετακινηθεί μέσα στην εφικτή περιοχή) προς το βέλτιστο. Η τεχνική MMFD χρησιμοποιεί μία από τις ακόλουθες μεθόδους για να βρει την κατεύθυνση αναζήτησης σε κάθε επανάληψη *q*:

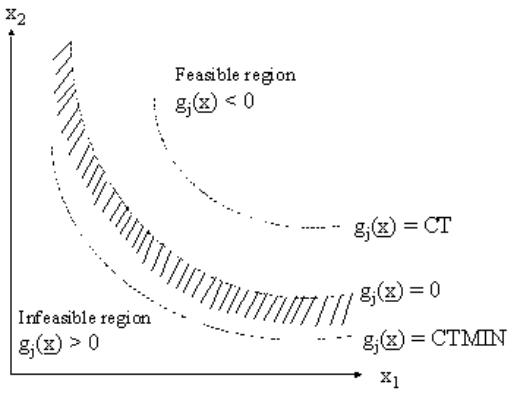
- Εάν δεν υπάρχουν ενεργοί ή παραβιασμένοι περιορισμοί, χρησιμοποιείται η μέθοδος χωρίς περιορισμούς Conjugate Gradient.
- Αν τυχόν περιορισμοί είναι ενεργοί και κανένας δεν παραβιάζεται, η τεχνική MMFD ελαχιστοποιείται.

Η τεχνική αποκτά γρήγορα ένα βέλτιστο σχεδιασμό και χειρίζεται άνισους και ίσους περιορισμούς. Χαρακτηριστικό αυτής της μεθόδου είναι ότι ικανοποιεί τους περιορισμούς με μεγάλη ακρίβεια στο βέλτιστο σημείο.

Οι ενεργοί και παραβιασμένοι περιορισμοί προσδιορίζονται ως εξής:

$g_j(x)$ είναι ενεργός, αν $CT \leq g_j(x) \leq CTMIN$

$g_j(x)$ παραβιάζεται, αν $g_j(x) > CTMIN$



Σχήμα 3.11 Active and Violated Constraint Identification

3.3 Approximations

Το Isight δίνει την δυνατότητα χρησιμοποίησης προσεγγίσεων (approximations) δηλαδή curve fitting τεχνικές για την επίτευξη πληροφορίας σχετικά με των συσχετισμό των παραμέτρων εισόδου και εξόδου βάση των δεδομένων που συλλέγονται πειραματικά. Οι παραγόμενοι συσχετισμοί είναι χρήσιμοι για την δημιουργία μεταπροτύπων (surrogate models).

Ένα μεταπρότυπο συχνά χρησιμοποιείται σε ένα πρόβλημα για την απόκτηση πληροφοριών, υψηλής ποιότητας, σχετικά με το μοντέλο σε όλη την διαδικασία της βελτιστοποίησης χωρίς το υψηλό υπολογιστικό κόστος που συνδέεται με μια βελτιστοποίηση. Τα μεταπρότυπα είναι αναλυτικές εξισώσεις που προσεγγίζουν τις

μεταβλητές εξόδου των περίπλοκων συστημάτων που δημιουργούνται παρέχοντας ένα σετ υπολογιστικών φθηνών προσομοιώσεων. Οι προσεγγίσεις βελτιστοποίησης που βασίζονται σε μεταπρότυπα έχουν εφαρμοστεί από διάφορους ερευνητές για την αντιμετώπιση ακριβών υπολογιστικών προβλημάτων λόγω των πλεονεκτημάτων τους.

3.3.1 Kriging Model

Το Kriging μοντέλο έχει τις ρίζες του στην γεωστατική επιστήμη και είναι χρήσιμο στο να προβλέπει προσωρινά και χωρικά συσχετισμένα δεδομένα. Τα μεταπρότυπα Kriging είναι εξαιρετικά ευέλικτα καθώς δίνεται η δυνατότητα επιλογής από ένα μεγάλο εύρος συναρτήσεων συσχέτισης για την δημιουργία του μεταπροτύπου. Επιπλέον στο Isight, υπάρχει επιλογή το μεταπρότυπο 'να τιμήσει τα δεδομένα', παρέχοντας ακριβή παρεμβολή των δεδομένων, ή 'να εξομαλύνει τα δεδομένα', παρέχοντας μια ανακριβή παρεμβολή.

Οι συναρτήσεις συσχέτισης που είναι διαθέσιμες στο Isight είναι:

- Exponential
- Gaussian
- Matern Linear
- Matern Cubic

Το Kriging απαιτεί έναν συνδυασμό από ένα πολυωνυμικό μοντέλο της ακόλουθης μορφής: $y(x) = f(x) + Z(x)$

Όπου *y(x)* είναι η άγνωστη συνάρτηση κέρδους, ενώ *f(x)* είναι μια γνωστή πολυωνυμική συνάρτηση του *x* που ονομάζεται *τάση* και *Z(x)* είναι η πραγματοποίηση μιας στοχαστικής διαδικασίας με μέση τιμή μηδέν, διακύμανση σ^2 και μη μηδενική συνδιακύμανση. Ο όρος *f(x)* στην προηγούμενη εξίσωση είναι παρόμοιος με το πολυώνυμικό μοντέλο σε μια επιφάνεια απόκρισης, παρέχοντας ένα μοντέλο του χώρου σχεδιασμού.

Ενώ το *f(x)* "συνολικά" προσεγγίζει τον χώρο σχεδιασμού, το *Z(x)* δημιουργεί "εντοπισμένες" αποκλίσεις έτσι ώστε το μοντέλο Kriging να παρεμβάλλει τα n_s δειγματοληπτικά σημεία δεδομένων. Μόλις επιλεχθεί η συνάρτηση συσχέτισης και προσδιοριστεί η καλύτερη παράμετρος συσχέτισης, το Kriging μοντέλο μπορεί να χρησιμοποιηθεί για τον προσδιορισμό της απόκρισης *y(X)*.

Το Isight επιπλέον υποστηρίζει την δημιουργία ισοτροπικών προσεγγίσεων με το Kriging μοντέλο. Ισοτροπική προσέγγιση χρησιμοποιείται όταν όλες οι ανεξάρτητες μεταβλητές συμπεριφέρονται παρόμοια, έτσι το Isight υποθέτει ότι όλες οι τιμές των παραμέτρων συσχέτισης είναι ίδιες. Αφού υπάρχει μόνο μια τιμή της παραμέτρου συσχέτισης η ισοτροπική εφαρμογή είναι συνήθως γρηγορότερη.

Ανάλογα με τον αριθμό των παραμέτρων εισόδου, των σημείων σχεδιασμού και τον αριθμό των αποκρίσεων (outputs) του Kriging μοντέλου, η δημιουργία του μπορεί να είναι χρονοβόρα. Όσο ο αριθμός των παραγόμενων πινάκων αυξάνεται η υπολογιστική ισχύ για τον χειρισμό των πινάκων αυξάνεται εκθετικά.

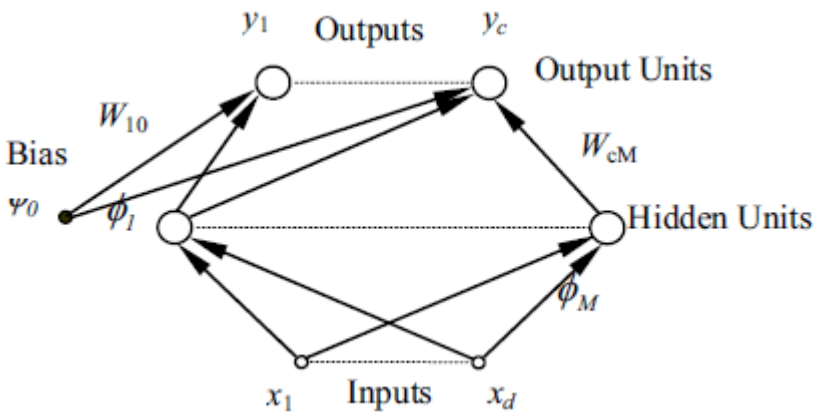
Η ποιότητα ενός μοντέλου εξαρτάται από τη θέση των σημείων δειγματοληψίας στον χώρο σχεδιασμού. Το μοντέλο Kriging έχει παρατηρηθεί ότι λειτουργεί καλύτερα με space-filling σχεδιασμούς όπου σημεία δειγματοληψίας τοποθετούνται πολύ μακριά. Όταν τα σημεία συγκεντρώνονται μαζί, οι πίνακες που χρησιμοποιούνται στην ενσωμάτωση (fitting) του μοντέλου Kriging γίνονται ill-conditioned, με αποτέλεσμα την κακή εφαρμογή τους. Για την αποφυγή του ill-conditioning δίνεται η δυνατότητα φιλτραρίσματος των σημείων του δείγματος με βάση την απόσταση. Όλα τα σημεία που είναι πιο κοντά από μια τιμή που ονομάζεται φίλτρο εξομάλυνσης (Smoothing Filter) αφαιρούνται από το σετ δειγμάτων.

Μόλις το μοντέλο έχει κατασκευαστεί, είναι έτοιμο να χρησιμοποιηθεί στην ανάλυση σχεδιασμού. Ο χρόνος που απαιτείται για την εφαρμογή ενός μοντέλου Kriging είναι αισθητά μεγαλύτερος από αυτόν που απαιτείται για άλλες τεχνικές παρεμβολής, όπως είναι οι Radial Basis Functions. Ωστόσο, οι χρόνοι πρόβλεψης είναι συγκρίσιμοι.

3.3.2 RBF and EBF Models

Η RBF προσέγγιση είναι ένας τύπος νευρωνικού δικτύου που χρησιμοποιεί ακτινικές μονάδες και χαρακτηρίζεται από γρήγορη εκπαίδευση και συμπαγή δίκτυα. Αποτελούνται από τρία επίπεδα, το επίπεδο εισόδου, το κρυφό επίπεδο και τους εξωτερικούς κόμβους. Κάθε συνιστώσα των μονάδων εισόδου x τροφοδοτεί προς τα εμπρός στις m κρυφές μονάδες, των οποίων οι έξοδοι συνδυάζονται γραμμικά με βάρη $\{w_j\}_{j-1}^{m}$ στις μονάδες εξόδου y του δικτύου, όπου κάθε κρυφή μονάδα αντιπροσωπεύει μια συνάρτηση ακτινικής βάσης φ. Ένα σύνολο RBF μπορεί να χρησιμεύσει ως βάση για την αναπαράσταση μιας ευρείας τάξης συναρτήσεων που μπορούν να εκφραστούν ως γραμμικοί συνδυασμοί των RBF: $y(x) = \sum_{j=1}^{M} w_j \varphi(\|x - \mu_j\|)$

.



Σχήμα 3.12 Τυπικό δίκτυο ακτινικής βάσης συνάρτησης

Τα κύρια χαρακτηριστικά των RBF [28] είναι ότι τα δίκτυα έχουν καλή τοπική κατασκευή και αποδοτικούς αλγόριθμους, επομένως έχουν χρησιμοποιηθεί ευρέως σε διάφορα πεδία. Το Isight χρησιμοποιεί την μέθοδο του Hardy [29] όπως περιγράφεται από τον Kansa [30]. Για ένα σετ δοσμένων κόμβων N, υπάρχουν ένα σετ από ακτινικές συναρτήσεις N. Η RBF παρεμβολή αποκτάται με την επίλυση $N + 1$ γραμμικών εξισώσεων για $N + 1$ άγνωστους συντελεστές επέκτασης.

Τα EBF είναι παρόμοια με τα RBF αλλά χρησιμοποιούν ελλειπτικές μονάδες αντί για ακτινικές. Σε αντίθεση με τα RBF όπου όλες οι είσοδοι αντιμετωπίζονται ίσα, τα EBF δίκτυα μεταχειρίζονται κάθε είσοδο ξεχωριστά χρησιμοποιώντας ξεχωριστά βάρη. Οι RBF προσεγγίσεις χαρακτηρίζονται από σχετικά γρήγορη εκπαίδευση. Από την άλλη μεριά, τα δίκτυα EBF χρειάζονται περισσότερες επαναλήψεις για να εκτιμηθούν τα ξεχωριστά βάρη των μεταβλητών εισόδου και είναι πιο ακριβή από τα RBF.

Ένα από τα κύρια πλεονεκτήματα της επιλογής του EBF έναντι του RBF είναι η ικανότητα του EBF να ταξινομεί τις μεταβλητές εισόδου με τη σειρά επιρροής στη μεταβλητή εξόδου.

3.3.3 Response Surface Model

Τα Μοντέλα Επιφάνειας Απόκρισης (Response Surface Models, RSM) στο Isight χρησιμοποιούν πολυώνυμα χαμηλής τάξης (από 1 έως 4) για την προσέγγιση της απόκρισης ενός πραγματικού κώδικα ανάλυσης. Για την κατασκευή ενός μοντέλου πρέπει να εκτελεστεί ένας αριθμός αναλύσεων ή μπορεί να χρησιμοποιηθεί ένα αρχείο δεδομένων με ένα σύνολο αναλυμένων σημείων σχεδιασμού. Τα μοντέλα αυτά μπορούν να χρησιμοποιηθούν σε μελέτες βελτιστοποίησης και ευαισθησίας με μικρό υπολογιστικό κόστος, επειδή η αξιολόγηση περιλαμβάνει μόνο τον υπολογισμό της τιμής ενός πολυωνύμου για ένα δεδομένο σύνολο τιμών εισόδου. Η ακρίβεια του μοντέλου εξαρτάται σε μεγάλο βαθμό από την ποσότητα των δεδομένων που χρησιμοποιούνται για την κατασκευή του, το σχήμα της ακριβούς συνάρτησης απόκρισης που προσεγγίζεται και τον όγκο του χώρου σχεδιασμού στον οποίο κατασκευάζεται το μοντέλο. Σε έναν αρκετά μικρό όγκο του χώρου σχεδιασμού, οποιαδήποτε ομαλή συνάρτηση μπορεί να προσεγγιστεί με ένα τετραγωνικό πολυώνυμο με καλή ακρίβεια. Για εξαιρετικά μη γραμμικές συναρτήσεις, μπορούν να χρησιμοποιηθούν πολυώνυμα τρίτης ή τέταρτης τάξης. Αν το μοντέλο χρησιμοποιείται εκτός του χώρου σχεδιασμού όπου κατασκευάστηκε, η ακρίβειά του είναι μειωμένη και το μοντέλο πρέπει να επανεκτιμηθεί.

3.4 Τεχνικές που προτείνονται για την βελτιστοποίηση του προβλήματος

Το πολυτομεακό πρόβλημα της βελτιστοποίησης της αποστολής του αεροσκάφους, όπως αναφέρθηκε και στα προηγούμενα κεφάλαια, είναι εξαιρετικά μη-γραμμικό και υψηλής τάξης με ασυνεχή χώρο σχεδιασμού. Δεν υπάρχει μια

συγκεκριμένη προσέγγιση για την αντιμετώπιση καθώς στο παρελθόν έχουν χρησιμοποιηθεί διαφορετικές τεχνικές [2]–[5], [7], [8], [12], [17], [20] .Στο πλαίσιο της παρούσας διπλωματικής εργασίας θα εφαρμοστεί η χρήση της DOE τεχνικής για την κατανόηση των αλληλοεπιδράσεων των παραμέτρων του προβλήματος. Στη συνέχεια θα χρησιμοποιηθούν διάφοροι αλγόριθμοι βελτιστοποίησης για την επίλυση του και θα συγκριθούν για την ακρίβεια και την ταχύτητά τους. Τέλος, θα δημιουργηθούν μεταπρότυπα του προβλήματος [4].

Όπως αναφέρθηκε προηγουμένως η Optimal Latin Hypercube είναι η ιδανικότερη τεχνική για την δημιουργία χώρου σχεδιασμού, λόγω των πλεονεκτημάτων της σε σχέση με την Latin Hypercube και Orthogonal Arrays. Οι παραγοντικοί σχεδιασμοί Full-Factorial και Fractional-Factorial είναι απαγορευτικοί καθώς όσο ο αριθμός των παραμέτρων αυξάνεται, το υπολογιστικό κόστος γίνεται απαγορευτικό. Το ίδιο συμβαίνει και με τους σχεδιασμούς Box-Behnken και Central Composite. Η παραμετρική μελέτη δεν παρέχει πληροφορίες για τις αλληλεπιδράσεις των παραμέτρων μεταξύ τους ωστόσο σε πιο περίπλοκα προβλήματα είναι ικανή να δείξει την τάση των μεταβλητών σχεδίασης. Παρόλα αυτά η αποτελεσματικότητά της είναι περιορισμένη.

Οι αλγόριθμοι βελτιστοποίησης που υπάρχουν στο Isight είναι διαφόρων τύπων. Λόγω της φύσης του προβλήματος απορρίπτονται όλες οι τεχνικές που βασίζονται στην κλίση (LSGRG, MISQP, MMFD, MOST, NLQPL) καθώς για ικανοποιητικά αποτελέσματα χρειάζονται συνεχή χώρο σχεδιασμού. Επομένως, θα χρησιμοποιηθούν τεχνικές αναζήτησης μοτίβου και γενετικοί αλγόριθμοι. Η τεχνική Downhill Simplex έχει χρησιμοποιηθεί λόγω της ταχύτητας της ωστόσο υπάρχει μεγάλη πιθανότητα να εγκλωβιστεί σε τοπικό ακρότατο. Γι’ αυτόν τον λόγο θα χρησιμοποιηθεί και η ASA ως μέτρο σύγκρισης των τεχνικών αναζήτησης μοτίβου καθώς μπορεί να είναι πιο χρονοβόρα αλλά είναι εξαιρετικά ακριβής μέθοδος και έχει χρησιμοποιηθεί και στο παρελθόν [9],[14] με καλά αποτελέσματα. Ο ένας γενετικός αλγόριθμος που θα χρησιμοποιηθεί είναι ο PSO, καθώς έχει χρησιμοποιηθεί αρκετές φορές για την αντιμετώπιση πολύπλευρων προβλημάτων [7]. Ένας άλλος γενετικός αλγόριθμος που θα χρησιμοποιηθεί είναι ο MIGA καθώς σε γενικές γραμμές οι γενετικοί αλγόριθμοι με υποπληθυσμούς παρουσιάζουν καλύτερες επιδόσεις όσον αφορά την σύγκλιση σε σύγκριση με την περίπτωση που χρησιμοποιείται ένας πληθυσμός [31].

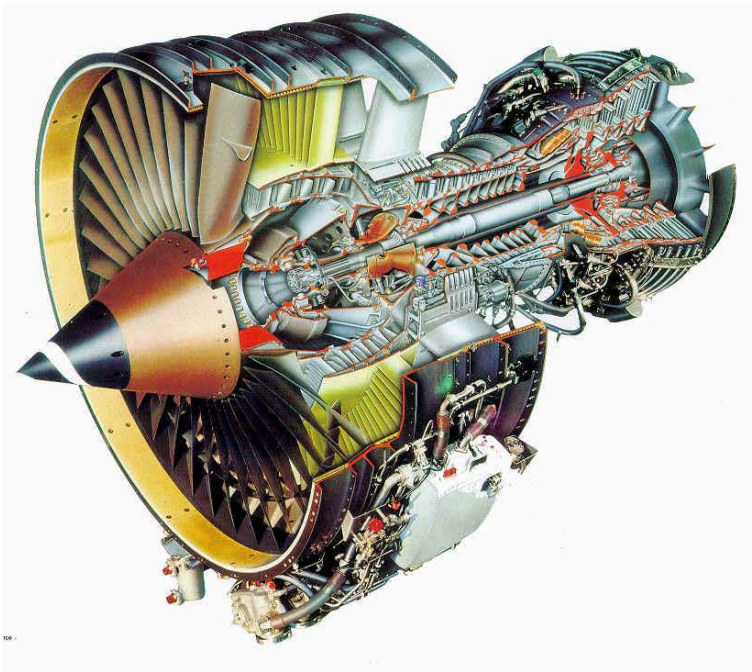
Για την δημιουργία μεταπροτύπου θα χρησιμοποιηθούν και θα συγκριθούν οι προσεγγίσεις Kriging και RBFs. Αφού αποτελέσματα έχουν δείξει [32] ότι η προσέγγιση Kriging είναι η καλύτερη, όταν υπάρχουν λίγα σημεία δειγματοληψίας τα οποία είναι ομοιόμορφα κατανεμημένα και μικρής κλίμακας μη-γραμμικά προβλήματα αλλά τα RBFs έχουν δείξει ότι είναι πιο αποτελεσματικά σε μεγάλης κλίμακας μη-γραμμικά προβλήματα [33]. Ωστόσο και οι δύο τεχνικές έχουν δείξει ότι είναι αρκετά αποτελεσματικές σε σύγκριση με άλλες τεχνικές προσεγγίσεις [34].

4 Εφαρμογή στον κινητήρα CFM56

Στο κεφάλαιο αυτό θα εφαρμοστούν οι τεχνικές βελτιστοποίησης που αναφέρθηκαν στο προηγούμενο κεφάλαιο, σε ένα μοντέλο προσομοίωσης ενός turbofan κινητήρα και συγκεκριμένα στον κινητήρα CFM56. Σκοπός της εφαρμογής είναι η πιστοποίηση της απόδοσης των τεχνικών βελτιστοποίησης που προτάθηκαν στο προηγούμενο κεφάλαιο, με την χρήση του λογισμικού Isight. Στη συνέχεια θα αναλυθούν τα αποτελέσματα και θα συγκριθούν οι τεχνικές μεταξύ τους

4.1 Ο αεροπορικός κινητήρας CFM56

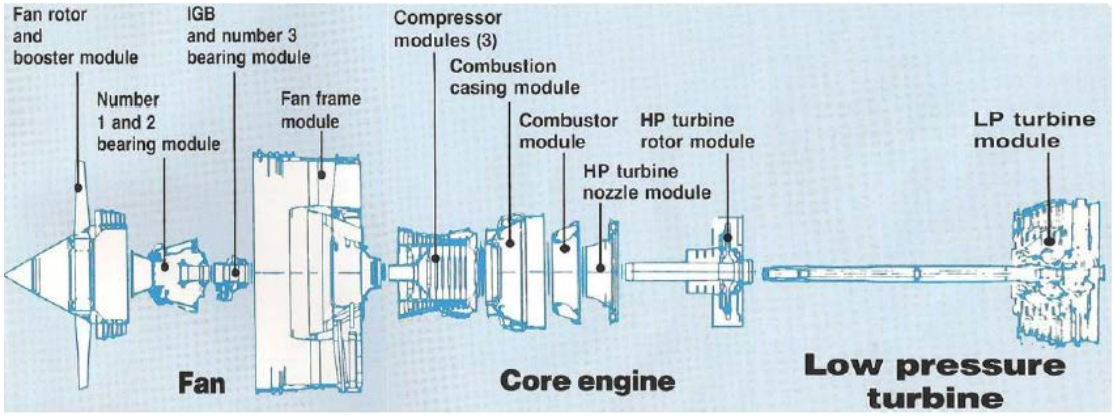
Ο κινητήρας CFM56 είναι turbofan κινητήρας διπλής ατράκτου, διπλού ρεύματος, μεγάλου λόγου παράκαμψης χωρίς ανάμιξη. Ο συγκεκριμένος κινητήρας χρησιμοποιείται σε κάποια από τα πιο γνωστά πολιτικά αεροσκάφη όπως τα Airbus A320/A340 και τα Boeing 737. Συγκεκριμένα η έκδοση του κινητήρα CFM56 που θα μελετηθεί θα είναι η 7B27.



Σχήμα 4.1 Ο κινητήρας CFM56-7B27

Ο κινητήρας αποτελείται από τις παρακάτω συνιστώσες: Τον αγωγό εισόδου, τον ανεμιστήρα (fan), τον συμπιεστή χαμηλής πίεσης (booster), τον συμπιεστή υψηλής πίεσης, τον θάλαμο καύσης, τον στρόβιλο υψηλής πίεσης, τον στρόβιλο χαμηλής πίεσης

και το ακροφύσιο κύριου ρεύματος και ρεύματος παράκαμψης. Στο Σχήμα 4.2 παρουσιάζεται η δομή του κινητήρα CFM56.



Σχήμα 4.2 Δομή του κινητήρα CFM56

Η μοντελοποίηση του κινητήρα έγινε στο λογισμικό προσομοίωσης κινητήρων Proosis από το ΕΘΣ. Δημιουργήθηκε ένα deck το οποίο μοντελοποιεί τις επιδόσεις του κινητήρα σε ένα σημείο λειτουργίας το cruise και σε άλλα 11 εκτός σχεδιασμού σημεία. Δίνεται η δυνατότητα στον χρήστη να τροποποιήσει την ποσοστιαία μεταβολή κάποιων παραμέτρων σε σχέση με τις τιμές αναφοράς που δίνονται από τον κατασκευαστή. Το σημείο λειτουργίας που θα βελτιστοποιηθεί, ως προς το SFC, είναι αυτό κατά την πτήση σε σταθερό ύψος ή κρουαζιέρα (cruise). Οι τιμές αναφοράς σε αυτό το σημείο φαίνονται στον Πίνακα 4.1.

Πίνακας 4.1

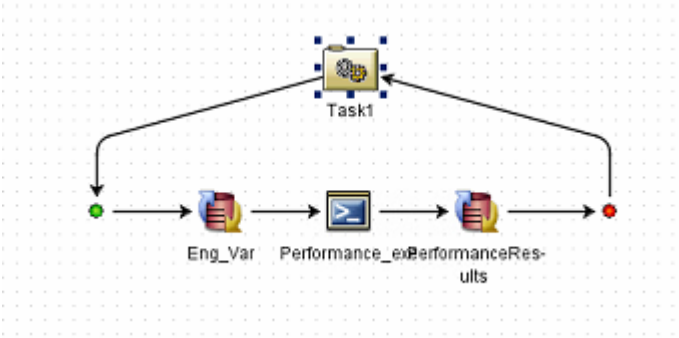
Point Name: Cruise					
Configuration: BASESR					
ALT:	10668.00	[m]	PWX4:	67.11	[kW]
M0:	0.78	[-]	N4:	13807.83	[rpm]
V0:	231.42	[m/s]	PW43:	8312.75	[kW]
DTisa:	0.00	[K]	T41m:	886.76	[K]
Ta:	218.81	[K]	T41c:	686.86	[K]
Pa:	23.84	[kPa]	EffC41:	0.63	[-]
PW2:	5953.64	[kW]	W41:	20.34	[kg/s]
NL:	0.90	[-]	W43:	22.07	[kg/s]
W2:	135.49	[kg/s]	P41:	918.76	[kPa]
W23:	22.13	[kg/s]	P43:	236.63	[kPa]
W13:	113.36	[kg/s]	T41:	1364.27	[K]
P2:	35.65	[kPa]	T43:	1001.97	[K]
P23:	60.59	[kPa]	P41Q43:	3.88	[-]

P13:	58.41	[kPa]	E43:	0.90	[-]
T2:	245.53	[K]	FAR41:	0.02	[-]
T23:	290.77	[K]	FAR43:	0.02	[-]
T13:	289.08	[K]	PW5:	6725.69	[kW]
P23Q2:	1.70	[-]	W46:	22.07	[kg/s]
P13Q2:	1.64	[-]	W5:	22.07	[kg/s]
E23:	0.89	[-]	P46:	236.63	[kPa]
E13:	0.86	[-]	P5:	58.73	[kPa]
FAR2:	0.00	[-]	T46:	1001.97	[K]
FAR23:	0.00	[-]	T5:	735.65	[K]
FAR13:	0.00	[-]	P46Q5:	4.03	[-]
BPR:	5.12	[-]	E5:	0.90	[-]
PW24:	704.78	[kW]	FAR46:	0.02	[-]
NI:	0.90	[-]	FAR5:	0.02	[-]
W24:	22.13	[kg/s]	V8:	493.92	[m/s]
W25:	22.13	[kg/s]	M8:	1.00	[-]
P24:	60.59	[kPa]	A8:	0.26	[m2]
P25:	82.46	[kPa]	W7:	22.07	[kg/s]
T24:	290.77	[K]	W8:	22.07	[kg/s]
T25:	322.48	[K]	P7:	58.73	[kPa]
P25Q24:	1.36	[-]	P8:	57.84	[kPa]
E25:	0.84	[-]	PS8:	30.93	[kPa]
FAR24:	0.00	[-]	T7:	735.65	[K]
FAR25:	0.00	[-]	T8:	735.65	[K]
PW26:	8162.51	[kW]	P7QN:	2.43	[-]
NH:	0.93	[-]	T7QN:	3.36	[-]
W26:	22.13	[kg/s]	FG8:	12.74	[kN]
WB27:	0.44	[kg/s]	V18:	311.23	[m/s]
W3:	21.69	[kg/s]	M18:	1.00	[-]
P26:	81.65	[kPa]	A18:	0.83	[m2]
P3:	977.14	[kPa]	W14:	113.36	[kg/s]
T26:	322.48	[K]	W18:	113.36	[kg/s]
T3:	686.86	[K]	P14:	57.21	[kPa]
P3Q26:	11.97	[-]	P18:	57.21	[kPa]
E3:	0.88	[-]	PS18:	30.21	[kPa]
FAR26:	0.00	[-]	T14:	289.08	[K]
FAR3:	0.00	[-]	T18:	289.08	[K]
W31:	17.35	[kg/s]	P14QN:	2.40	[-]
W4:	17.73	[kg/s]	T14QN:	1.32	[-]

P31:	967.33	[kPa]	FG18:	40.59	[kN]
P4:	918.76	[kPa]	OPR:	27.41	[-]
T31:	686.86	[K]	SFC:	17.50	[g/(kN*s)]
T4:	1455.00	[K]	SFN:	162.12	[N*s/kg]
WF:	0.38	[kg/s]	ETAth:	0.44	[-]
dP4Q31:	0.05	[-]	ETAprop:	0.69	[-]
E4:	1.00	[-]	ETAov:	0.31	[-]
FAR4:	0.02	[-]	FD:	31.36	[kN]
PWX46:	0.00	[kW]	FN:	21.97	[kN]
N46:	4543.25	[rpm]			

4.2 Ενσωμάτωση στο Isight

Το deck δομήθηκε έτσι ώστε να υπάρχουν τρία διαφορετικά αρχεία. Στο Isight ενσωματώθηκαν τα αρχεία Eng_Var.txt το οποίο περιέχει τις μεταβλητές σχεδιασμού, το εκτελέσιμο Performance.exe το οποίο πραγματοποιεί τους απαραίτητους υπολογισμούς και το PerformanceResults.txt στο οποίο βρίσκονται τα αποτελέσματα της μοντελοποίησης. Τα αρχεία .txt ενσωματώθηκαν με το component εφαρμογής Data Exchanger ενώ το εκτελέσιμο με το component OS Command. Στο Σχήμα 4.3 φαίνεται η προσομοίωση της ροής της διαδικασίας, στο γραφικό περιβάλλον του Isight.



Σχήμα 4.3 Προσομοίωση της ροής της διαδικασίας

Σαν μεταβλητές σχεδιασμού ορίστηκαν οι ακόλουθες παράμετροι: Ο λόγος πίεσης του Fan (FANOUT_PR) και οι παροχής αέρα στον αγωγό εισόδου (dW1) και στον συμπιεστή χαμηλής πίεσης (dW23). Επιλέχθηκαν αυτές οι τρεις μεταβλητές σχεδιασμού έτσι ώστε το πρόβλημα να παραμείνει σχετικά απλό, καθώς ο σκοπός της συγκεκριμένης εφαρμογής είναι να δείξει τις ικανότητες του λογισμικού. Επιπλέον οι συγκεκριμένες μεταβλητές επηρεάζουν σημαντικά την προωθητική απόδοση του κινητήρα. Με το πέρας των υπολογισμών από το εκτελέσιμο, δημιουργείται το

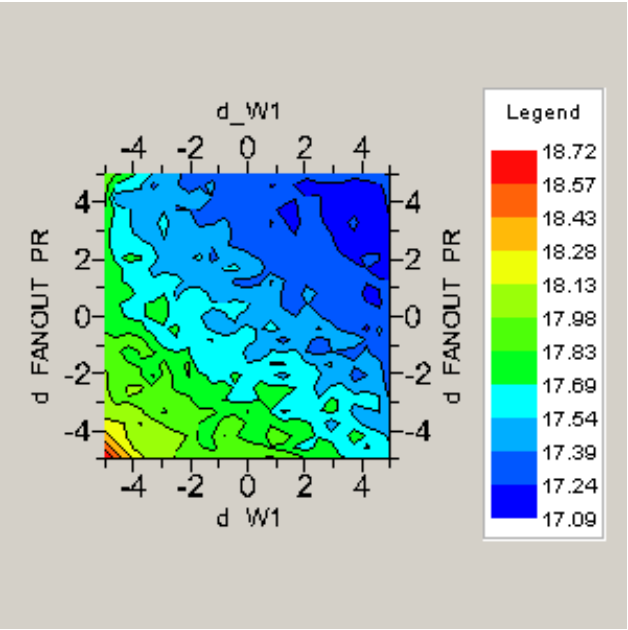
PerformanceResults.txt με τιμές διαφόρων παραμέτρων στο σημείο σχεδίασης (top-of climb) και στα 11 σημεία λειτουργίας εκτός σημείου σχεδίασης (cruise, take-off, 4 σημεία πιστοποίησης ICAO (100, 85, 30 & 7% of SLS Thrust), 3 σημεία πιστοποίησης Noise (Sideline, Cutback & Approach), top-of-climb ISA and SLS take-off Hot Day). Ως συνάρτηση κόστους ορίζεται η τιμή της ειδικής κατανάλωσης καυσίμου (SFC) κατά την κρουαζιέρα. Στο πρόβλημα τέθηκαν και δύο περιορισμοί. Ο ένας είναι η θερμοκρασία μετάλλου στα κινητά πτερύγια του στροβίλου κατά την απογείωση σε ζεστή μέρα (T41m_Hot), όπου δεν πρέπει να ξεπερνάει τους 1100 K. Ο δεύτερος ήταν η ελάχιστη ώση (FN) που απαιτείται για την απογείωση σε ζεστή μέρα όπου είναι 115 kN καθώς είναι η ελάχιστη που απαιτείται για την απογείωση του αεροσκάφους, σύμφωνα με τον κατασκευαστή.

4.3 Εφαρμογή των τεχνικών για την βελτιστοποίηση του προβλήματος

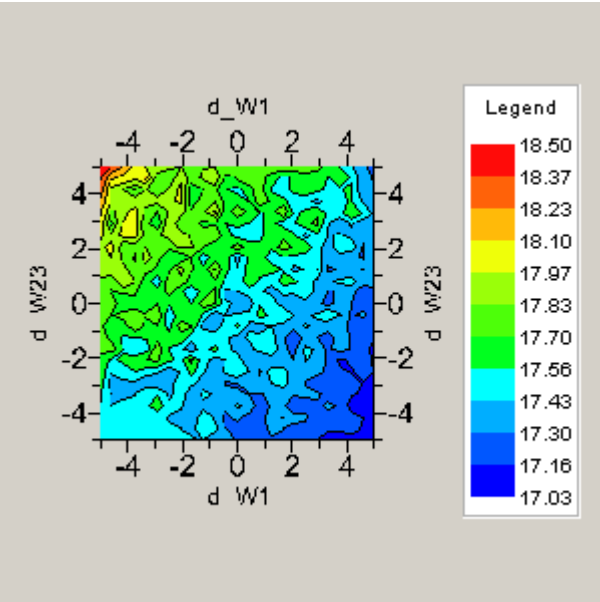
Για την βελτιστοποίηση του προβλήματος θα χρησιμοποιηθούν και θα συγκριθούν μεταξύ τους τέσσερις αλγόριθμοι βελτιστοποίησης. Επιπλέον θα γίνει χώρος σχεδιασμού των πειραμάτων, ο οποίος μπορεί να δείξει την τάση των μεταβλητών σχεδίασης σε σχέση με την τιμή αναφοράς τους. Τέλος θα κατασκευαστούν δύο μεταπρότυπα για την γρήγορη προσέγγιση της λύσης. Οι τιμές των μεταβλητών σχεδίασης θα κυμανθούν στο [-5%,5%] εύρος από την τιμή αναφοράς τους.

4.3.1 Δημιουργία DOE με την χρήση της Optimal Latin Hypercube τεχνικής

Για την δημιουργία του χώρου σχεδιασμού των πειραμάτων επιλέχθηκε η τεχνική Optimal Latin Hypercube η οποία υπερτερεί από τις άλλες όπως εξηγήθηκε στο προηγούμενο κεφάλαιο. Ωστόσο η απεικόνιση του χώρου σχεδιασμού είναι δύσκολη όταν μελετώνται παραπάνω από δύο παράγοντες. Παρόλα αυτά μπορεί να φανεί μια τάση ως προς τις τιμές των παραμέτρων. Στα Σχήματα 4.4, 4.5 και 4.6 φαίνεται η τάση των τιμών των μεταβλητών σχεδίασης σε σχέση με την αντικειμενική συνάρτηση. Η Optimal Latin Hypercube πραγματοποιήθηκε με 400 σημεία σχεδιασμού και ο χρόνος για την ισοκατανομή των σημείων διήρκησε περίπου 4 λεπτά. Η επιλογή του αριθμού των σημείων έγινε γιατί με κατάλληλη ανάλυση ευαισθησίας των αποτελεσμάτων αυξάνοντας σταδιακά τον αριθμό των σημείων. παραπάνω σημεία τα αποτελέσματα δεν θα διέφεραν και δεν δικαιολογείται το επιπλέον υπολογιστικό κόστος. Ενώ η χρησιμοποίηση λιγότερων σημείων, δεν βγάζει ικανοποιητικά αποτελέσματα καθώς δεν γίνεται καλή η απεικόνιση ενός τόσο κατακερματισμένου χώρου σχεδιασμού. Επομένως, όταν υπάρχει σχετικά μικρός αριθμός μεταβλητών σχεδιασμού 3-6, 400 σημεία σχεδίασης επαρκούν για την δημιουργία χώρου σχεδιασμού των πειραμάτων.



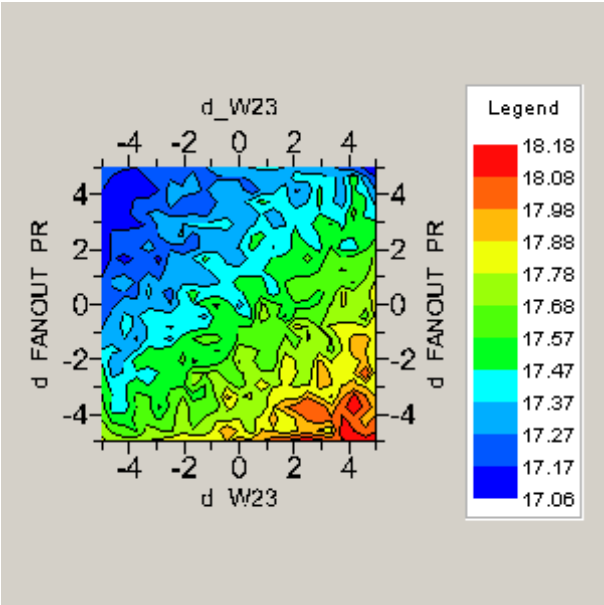
Σχήμα 4.4 Μεταβλητές σχεδιασμού dW1 και dFANOUT_PR σε σχέση με την SFC



Σχήμα 4.5 Μεταβλητές σχεδιασμού dW1 και dW23 σε σχέση με την SFC

Στο Σχήμα 4.4 διακρίνεται η τάση του λόγου πίεσης του ανεμιστήρα και της παροχής αέρα στον αγωγό εισόδου να αυξηθεί. Ουσιαστικά το W1 καθορίζει το μέγεθος του ανεμιστήρα, όσο αυξάνεται τόσο θα αυξάνεται και το μέγεθός του.

Στο Σχήμα 4.5 φαίνεται ότι η παροχή αέρα στον συμπιεστή χαμηλής πίεσης τείνει να μειωθεί ενώ παράλληλα να αυξηθεί η αντίστοιχη παροχή στον αγωγό εισόδου. Δηλαδή να αυξηθεί σε μέγεθος ο ανεμιστήρας και να μικρύνει ο πυρήνας της μηχανής, επομένως να αυξηθεί ο λόγος παράκαμψης του κινητήρα (BPR).



Σχήμα 4.6 Μεταβλητές σχεδιασμού dW23 και dFANOUT_PR σε σχέση με την SFC

Στο Σχήμα 4.6 φαίνονται ίδιες τάσεις στις μεταβλητές όπως στα Σχήματα 4.4 και 4.5.

4.3.2 Ο αλγόριθμος βελτιστοποίησης Adaptive Simulated Annealing

Ο πρώτος αλγόριθμος βελτιστοποίησης που χρησιμοποιήθηκε είναι ο ASA. Κάθε αλγόριθμος στο Isight έχει ξεχωριστό configuration. Σε κάθε περίπτωση οι τιμές των παραμέτρων του κάθε αλγόριθμου επιλέχθηκαν σύμφωνα με τις πληροφορίες που υπάρχουν στο εγχειρίδιο του λογισμικού και στην βιβλιογραφία. Οι τιμές των παραμέτρων του αλγορίθμου ASA για το συγκεκριμένο πρόβλημα φαίνονται στον Πίνακα 4.2. Στη συνέχεια επεξηγείται η επιλογή της τιμής της κάθε παραμέτρου.

Πίνακας 4.2

Optimization Technique:	ASA
Convergence Epsilon	0.000001
Failed Run Objective Value	1E+30
Failed Run Penalty Value	1E+30
Init Param Temperature	1
Max Failed Runs	-1
Max Num of Failed Designs	10
Max Num of Generated Designs	2000
Min Ratio of Accept Des for Reannealing	0.1
Num of Des for Conv Check	2
Num of Accept Des Before Reannealing	100
Num of Des Before Reannealing	1000

Penalty Base	0
Penalty Exponent	2
Penalty Multiplier	1000
Random seed value	-1
Reanneal Cost Function	TRUE
Reanneal Parameters	TRUE
Rel Rate of Cost Annealing	1
Rel Rate of Cost Quenching	1
Rel Rate of Param Annealing	1
Rel Rate of Param Quenching	1
Rel Grad Step for Reannealing	0.00001
Use fixed random seed	FALSE

Το Convergence Epsilon αποτελεί το κριτήριο σύγκλισης του αλγόριθμου, δηλαδή την μέγιστη επιτρεπόμενη διαφορά κάθε αποδεκτής λύσης με την καλύτερη μέχρι στιγμής. Οι παράμετροι Penalty Base, Penalty Exponent, Penalty Multiplier και Failed Run Penalty Value επηρεάζουν την συνάρτηση ποινής που στο συγκεκριμένο πρόβλημα δεν έχει επιλεχθεί να χρησιμοποιηθεί, καθώς οι περιορισμοί δεν επηρεάζουν τόσο την τιμή της αντικειμενικής συνάρτησης. Οπότε αυτές οι παράμετροι, αφού το Penalty Base ορίστηκε 0, ότι τιμή και να πάρουν δεν έχουν αντίκτυπο. Η μέγιστη τιμή της παραμέτρου Initial Parameter Temperature είναι 1. Αποδεκτές τιμές είναι (0,1]. Όσο μικραίνει η παράμετρος τόσο αυξάνεται ο χρόνος εκτέλεσης της βελτιστοποίησης, για την εύρεση καλύτερης τιμής, ωστόσο ο επιπλέον χρόνος υπολογισμού δεν συμφέρει για το συγκεκριμένο πρόβλημα. Η τιμή της παραμέτρου Max Failed Runs είναι ένας αρνητικός αριθμός και επιτρέπει στην διαδικασία να μην σταματήσει ανεξάρτητού αριθμού μη επιτρεπτών αξιολογήσεων. Η παράμετρος αυτή υπάρχει σε όλους τους αλγόριθμους βελτιστοποίησης του Isight και η τιμή θα είναι -1 σε όλες τις εκτελέσεις. Παρόμοια παράμετρος είναι και η Maximum Number of Failed Designs, μόνο που εδώ έχει να κάνει με αριθμό σχεδιασμών. Κύριο χαρακτηριστικό της ASA είναι η επανόπτηση (Reannealing) των παραμέτρων και της συνάρτησης κόστους. Όταν ο αλγόριθμος βρεθεί σε στασιμότητα τότε δίνεται η δυνατότητα επανεκκίνησης της διαδικασίας από τον καλύτερο σχεδιασμό μέχρι στιγμής. Η δυνατότητα αυτή χρησιμοποιήθηκε στην βελτιστοποίηση με την χρήση των επιλογών Reanneal Parameters και Reanneal Cost Function. Τα διάφορα κριτήρια για την επιλογή Reanneal Parameters και Cost Function είναι τα Number of Generated Designs Before Annealing, Number of Accepted Designs Before Reannealing ενώ το πιο αποτελεσματικό κριτήριο είναι το Minimum Ratio of Accepted Designs for Reannealing. Ο λόγος αυτός, Accepted Designs προς Number of Generated Designs, καθορίζει πότε θα γίνει η επανόπτηση. Οι τιμές των Relative Rate of Cost Quenching, Parameter Quenching, Cost Annealing και Parameter Annealing ορίστηκε ως η μέγιστη επιτρεπόμενη τους 1. Όσο μειώνεται η τιμή τους τόσο

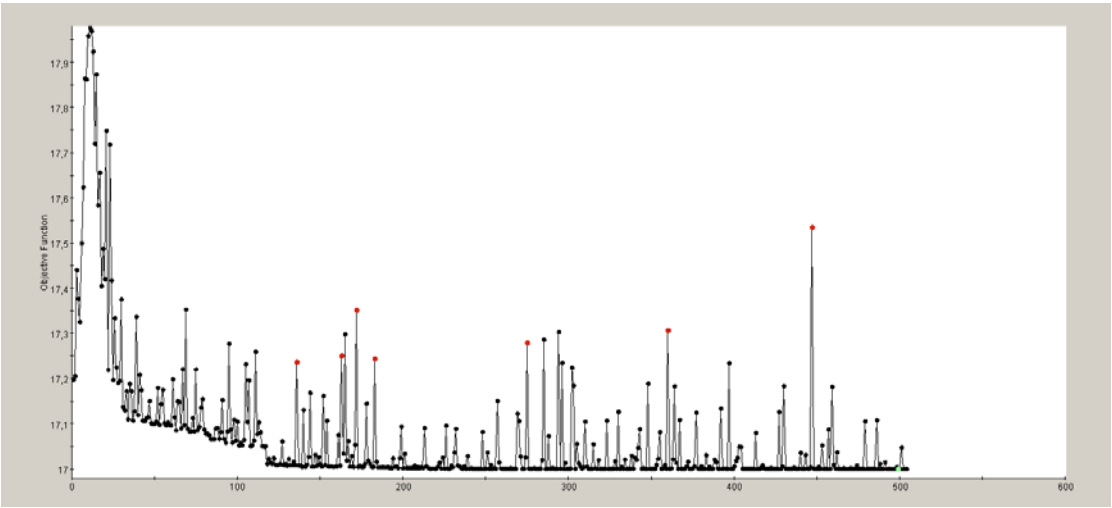
μεγαλώνει ο χρόνος μείωσης των 'θερμοκρασιών' και ως εκ τούτου η σύγκλιση του αλγόριθμου. Η τελευταία σημαντική παράμετρος της ASA είναι η Relative Gradient Step for Reannealing. Κατά την επανόπτηση, οι θερμοκρασίες των παραμέτρων αυξάνονται ανάλογα με την επίδραση τους στην συνάρτηση κόστους. Για τον καθορισμό της επιρροής κάθε μεταβλητής σχεδιασμού στην συνάρτηση κόστους, υπολογίζονται παράγωγοι με την χρήση πεπερασμένων διαφορών. Η τιμή του βήματος που επιλέξαμε είναι 10^{-5} .

Τα αποτελέσματα της βελτιστοποίησης παρουσιάζονται στον Πίνακα 4.3.

Πίνακας 4.3 Τιμές παραμέτρων στο βέλτιστο σημείο σχεδιασμού

Optimum design point:						
		Μεταβλητές σχεδιασμού			Περιορισμοί	
Run #	Objective/SFC cruise	dFANOUT_PR	dW1	dW23	FN_TakeOffHot	T41m_TakeOffHot
499	17.00033794	4.998703502	4.9989106	-4.999342807	116.7021245	1076.278991

Στο Σχήμα 4.7 φαίνεται η τιμή της αντικειμενικής συνάρτησης σε κάθε αξιολόγηση. Με κόκκινο είναι τα σημεία που δεν ικανοποιούν τους περιορισμούς ή δεν βγάζει αποτέλεσμα το εκτελέσιμο. Με πράσινο φαίνεται η βέλτιστη τιμή της αντικειμενικής συνάρτησης.



Σχήμα 4.7 History of objective function -ASA

Ο αλγόριθμος σύγκλινε μετά από 504 αξιολογήσεις και έτσι δεν πρόλαβε να γίνει η διαδικασία της επανόπτησης. Η επανόπτηση είναι χρήσιμη για πιο πολύπλοκα προβλήματα με περισσότερες μεταβλητές σχεδιασμού και ένα από αυτά θα μελετηθεί στην επόμενη εφαρμογή.

4.3.3 Ο αλγόριθμος βελτιστοποίησης Downhill Simplex

Ο επόμενος αλγόριθμος που χρησιμοποιήθηκε ήταν ο Downhill Simplex. Η ρύθμιση των παραμέτρων του αλγόριθμου φαίνεται στον Πίνακα 4.4.

Πίνακας 4.4

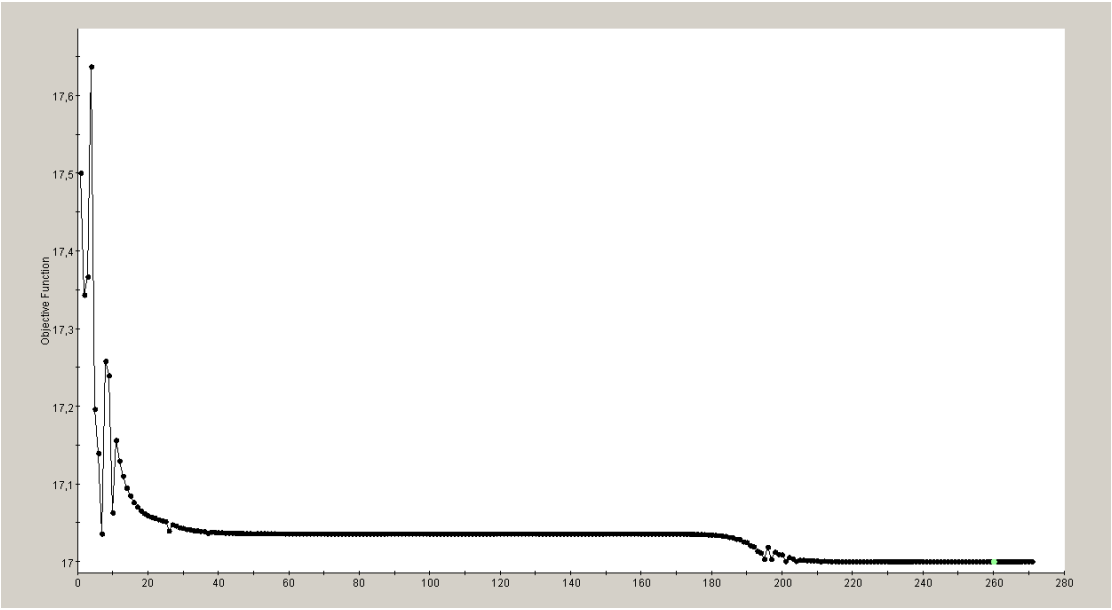
Optimization Technique:	DownhillSimplex
Failed Run Objective Value	1E+30
Failed Run Penalty Value	1E+30
Initial Simplex Size	0.3
Max Failed Runs	-1
Max Iterations	400

Το αρχικό μέγεθος του σχήματος Simplex ορίστηκε ως το 30% του μεγέθους του χώρου σχεδιασμού ενώ ο μέγιστος αριθμός αξιολογήσεων ήταν 400. Για το μέγεθος του Simplex θεωρήθηκε ότι 30% είναι ικανοποιητικό καθώς όσο μεγαλύτερο γίνεται, τόσο μεγαλύτερος κίνδυνος υπάρχει για τον εγκλωβισμό του αλγορίθμου σε κάποιο τοπικό ακρότατο. Τα αποτελέσματα της βελτιστοποίησης παρουσιάζονται στον Πίνακα 4.5.

Πίνακας 4.5

Optimum design point:						
	Μεταβλητές Σχεδιασμού				Περιορισμοί	
Run #	Objective/SFC Cruise	d_FANOUT_PR	d_W1	d_W23	FN_TakeOff_Hot	T41m_TakeOff_Hot
260	17.00031	5	5	-5	116.7013	1076.28

Όπως και στο Σχήμα 4.7 έτσι και στο Σχήμα 4.8 φαίνεται η τιμή της αντικειμενικής συνάρτησης σε κάθε αξιολόγηση. Ο αλγόριθμος σύγκλινε μετά από 271 αξιολογήσεις.



Σχήμα 4.8 History of objective function – Downhill Simplex

Οι επόμενοι αλγόριθμοι που χρησιμοποιήθηκαν είναι εξελικτικοί. Πρόκειται για τον PSO και τον MIGA.

4.3.4 Ο αλγόριθμος βελτιστοποίησης Multi-Island Genetic Algorithm

Οι τιμές των παραμέτρων του αλγόριθμου φαίνονται στον Πίνακα 4.6.

Πίνακας 4.6

Optimization Technique:	Multi-Island GA
Elite Size	1
Failed Run Objective Value	1E+30
Failed Run Penalty Value	1E+30
Interval of Migration	3
Max Failed Runs	-1
Number of Generations	15
Number of Islands	3
Penalty Base	0
Penalty Exponent	2
Penalty Multiplier	1000
Random seed value	-1
Rate of Crossover	0.8
Rate of Migration	0.1

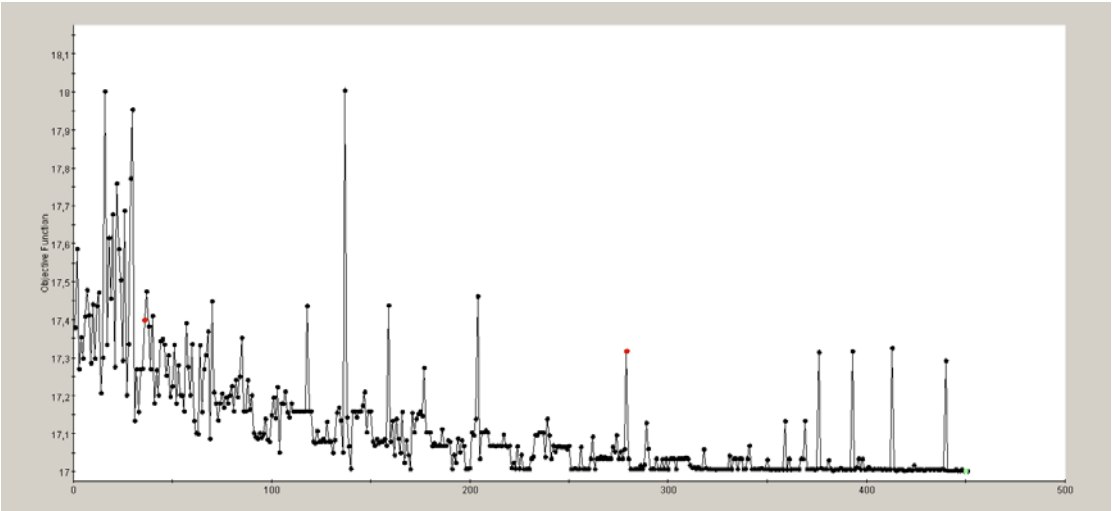
Rate of Mutation	0.01
Rel Tournament Size	0.35
Sub-Population Size	10
Use fixed random seed	FALSE

Ο αριθμός των ατόμων που θα μεταφερθούν σε κάθε επόμενη γενιά σε κάθε υποπληθυσμό είναι 1 κάνοντας τον αλγόριθμο ελιτιστικό και πιο γρήγορο με αντάλλαγμα την έλλειψη ποικιλομορφίας. Ο αριθμός των νησιών που ορίστηκαν είναι τρία καθώς παραπάνω νησιά θα αύξανε τον αριθμό των ατόμων κάνοντας την διαδικασία αρκετά αργή. Επιπλέον, με την χρήση μικρού αριθμού νησιών θα φανεί η διαφορά του αλγόριθμου σε σχέση με έναν εξελικτικό ενός πληθυσμού. Οι [35]–[37] είναι κάποιες benchmark μελέτες για την επίδραση των παραμέτρων των αλγορίθμων που χρησιμοποιούν το μοντέλο των νησιών. Ωστόσο επειδή κάθε αλγόριθμος έχει κωδικοποιηθεί διαφορετικά και δεν υπάρχει γνώση πως εφαρμόζεται ο MIGA στο Isight, τα αποτελέσματα των μελετών ερμηνεύτηκαν με βάση την εμπειρία που προέκυψε από προηγούμενα τρεξίματα. Παρόλα αυτά η αρχή κάθε αλγόριθμου, που χρησιμοποιεί αυτό το μοντέλο, είναι ίδια. Οι πολύ συχνές μεταναστεύσεις μειώνουν την ποικιλομορφία σε κάθε νησί επειδή τα νησιά μεταξύ τους ανταλλάσσουν άτομα τόσο γρήγορα, που όλα μετά περιέχουν τα ίδια άτομα. Οι σπάνιες μεταναστεύσεις δεν αφήνουν τα καλά άτομα από κάποια νησιά να μεταναστεύσουν στα άλλα. Στο συγκεκριμένο πρόβλημα η μετανάστευση γίνεται μετά από κάθε τρεις γενιές. Όσον αφορά το μέγεθος των ατόμων που θα μεταναστεύουν (Migration Size), 10% είναι αρκετά ικανοποιητικό [36], [37]. Το ποσοστό των σχεδιασμών στα οποία θα γίνει διασταύρωση ορίστηκε ως 80% ενώ η πιθανότητα μετάλλαξης ως 1% που είναι και η συνήθης τιμή στους γενετικούς αλγορίθμους. Το ποσοστό των σχεδιασμών που επιλέχθηκαν, κάθε υποπληθυσμού για την παραγωγή νέων απογόνων ορίστηκε ως 35%. Τα αποτελέσματα της βελτιστοποίησης φαίνονται στον Πίνακα 4.7.

Πίνακας 4.7

Optimum design point:						
		Μεταβλητές Σχεδιασμού			Περιορισμοί	
Run #	Objective/SFC cruise	d_FANOUT_PR	d_W1	d_W23	FN_TakeOff_Hot	T41m_TakeOff_Hot
450	17.002	4.691615	4.95285	-4.9852	116.8338	1075.991

Στο Σχήμα 4.9 φαίνεται η τιμή της αντικειμενικής συνάρτησης σε κάθε αξιολόγηση. Ο αλγόριθμος σύγκλινε μετά από 451 αξιολογήσεις.



Σχήμα 4.9 History of objective function -MIGA

4.3.5 Ο αλγόριθμος βελτιστοποίησης Particle Swarm Optimizer

Οι τιμές των παραμέτρων του αλγόριθμου παρουσιάζονται στον Πίνακα 4.8.

Πίνακας 4.8

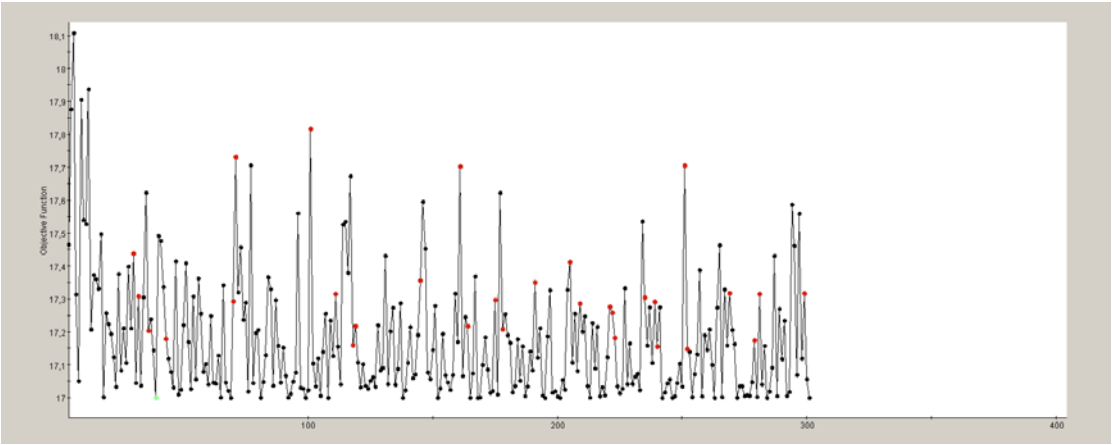
Optimization Technique:	Multi-Objective Particle Swarm
Failed Run Objective Value	1E+30
Failed Run Penalty Value	1E+30
Global Increment	1.655
Inertia	0.721
Max Failed Runs	-1
Maximum Iterations	20
Maximum Velocity	10
Number of Particles	15
Particle Increment	1.655
Random seed value	-1
Use fixed random seed	FALSE

Οι τιμές των συντελεστών Global Increment, Particle Increment και Inertia έχουν παρθεί από την [38] ύστερα από διάφορες συγκρίσεις με διάφορες τιμές της [39]. Ο ιδανικός αριθμός των σωματιδίων κυμαίνεται από [4.0,5.0] επί των αριθμών των μεταβλητών σχεδιασμού. Η μέγιστη ταχύτητα έχει να κάνει με την απόλυτη διαφορά που μπορεί να πάρει η αντικειμενική συνάρτηση από την αρχική της τιμή. Επειδή ήδη είναι γνωστή η ελάχιστη τιμή της αντικειμενικής, η τιμή της μέγιστης ταχύτητας θα ναι 10. Στον Πίνακα 4.8 φαίνονται τα αποτελέσματα της βελτιστοποίησης.

Πίνακας 4.8

Optimum design point:						
		Μεταβλητές Σχεδιασμού			Περιορισμοί	
Run #	Objective/SFC cruise	d_FANOUT_PR	d_W1	d_W23	FN_TakeOff_Hot	T41m_TakeOff_Hot
39	17.00031	5	5	-5	116.7013	1076.28

Στο Σχήμα 4.10 φαίνεται η τιμή της αντικειμενικής συνάρτησης σε κάθε αξιολόγηση. Ο αλγόριθμος σύγκλινε μετά από 301 αξιολογήσεις.



Σχήμα 4.10 History of objective function -PSO

4.3.6 Χρήση μεταπρότυπων Kriging και RBF

Η τελευταία τεχνική που χρησιμοποιήθηκε ήταν η κατασκευή μεταπροτύπων. Δύο είδη μεταπροτύπων επιλέχθηκαν τα Kriging και τα RBF για τους λόγους που αναφέρθηκαν στο προηγούμενο κεφάλαιο.

Kriging μεταπρότυπο

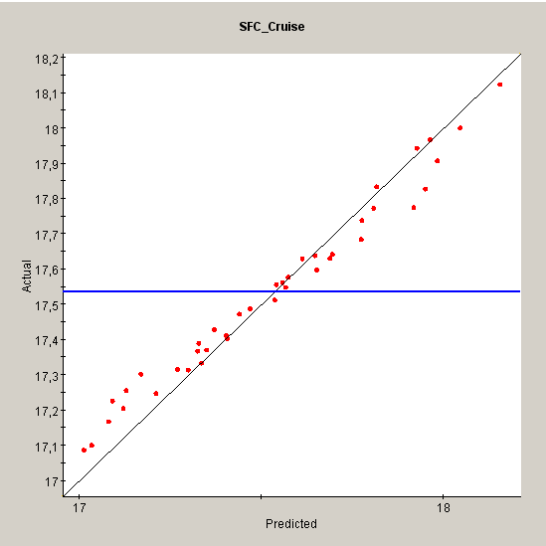
Τα Kriging μεταπρότυπα είναι μια αρκετά διαδεδομένη τεχνική προσέγγισης, που προσεγγίζει τον χώρο σχεδιασμού χρησιμοποιώντας μια συνάρτηση συσχέτισης και μια τοπική απόκλιση από το μοντέλο αναφοράς. Στην συγκεκριμένη εφαρμογή χρησιμοποιήθηκε η συνάρτηση συσχέτισης Matern Linear καθώς ήταν η μοναδική από τις υπόλοιπες επιλογές Gaussian, Exponential και Mattern Cubic που έδινε αποδεκτό fit. Επιπλέον χρησιμοποιήθηκε ανισοτροπική εφαρμογή (fit) η οποία εξετάζει την συμπεριφορά της κάθε μεταβλητής ανεξάρτητα. Στο λογισμικό δίνεται η δυνατότητα χρησιμοποίησης φίλτρου που αποκόπτει τα απομακρυσμένα σημεία, με βάση ένα κριτήριο απόστασης, για την καλή εφαρμογή του μεταπροτύπου. Η τιμή του ορίστηκε 0,05. Η αποτελεσματικότητα των μεταπροτύπων σχετίζεται με τον αριθμό των σημείων δειγματοληψίας. Όσο δυνατόν περισσότερα σημεία υπάρχουν για την εκπαίδευση του μεταπροτύπου τόσο πιο ακριβής θα είναι η προσέγγιση. Ωστόσο, ο υπολογιστικός χρόνος για την δημιουργία του DOE αυξάνεται σημαντικά με την ύπαρξη πολλών

σημείων. Οι Jin et al. [33] πρότειναν για μικρής κλίμακας προβλήματα με *n* αριθμό μεταβλητών σχεδίασης περίπου *10n* σημεία δειγματοληψίας. Το Isight προτείνει 32, ωστόσο επιλέχθηκαν 40 σημεία καθώς ο επιπλέον υπολογιστικός χρόνος είναι ελάχιστος και επειδή κάποια σημεία προκύπτουν infeasible, μερικές φορές προκύπτουν λιγότερα σημεία από όσα θέτει ο χρήστης. Η τεχνική δημιουργίας χώρου σχεδιασμού των πειραμάτων ήταν η Optimal Latin Hypercube όπως εξηγήθηκε στο προηγούμενο κεφάλαιο, λόγω των πλεονεκτημάτων της [32]. Ως είσοδοι, ορίστηκαν οι τρεις μεταβλητές σχεδιασμού ενώ ως έξοδοι η αντικειμενική συνάρτηση και οι περιορισμοί. Η μέθοδος για την ανάλυση του σφάλματος που χρησιμοποιήθηκε είναι η cross-validation και πιο συγκεκριμένα η leave-one-out cross-validation. Η μέθοδος αυτή αφαιρεί ένα σημείο από τα σημεία δειγματοληψίας, οι συντελεστές της προσέγγισης υπολογίζονται ξανά, ενώ οι ακριβείς και οι προσεγγιστικές τιμές των εξόδων συγκρίνονται μεταξύ τους. Το σημείο στη συνέχεια τοποθετείται πίσω και επιλέγεται το επόμενο σημείο. Επιλέχθηκαν όλα τα σημεία δειγματοληψίας για την ανάλυση σφάλματος. Τα αποτελέσματα της ανάλυσης σφάλματος φαίνονται στον Πίνακα 4.9

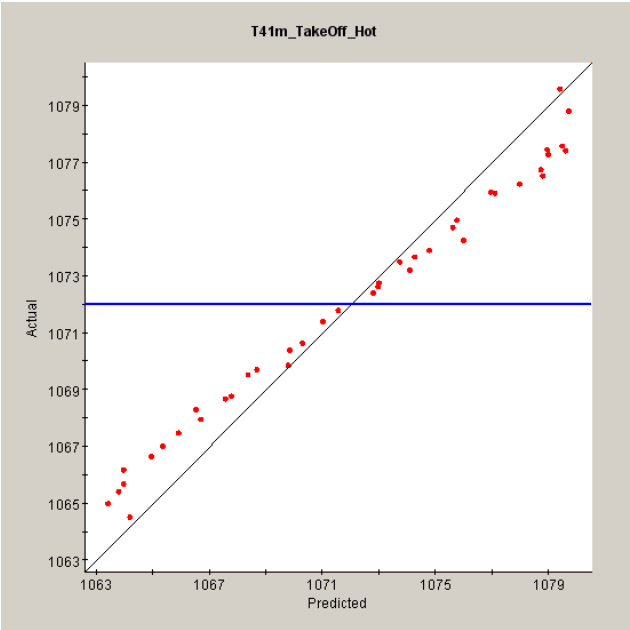
Πίνακας 4.9

Kriging	Error Type			
Outputs	Maximum	Average	Root Mean Square	R Square
Sfc_cruise	0.13691	0.0466	0.0605	0.94085
T41m_takeoff_hot	0.14864	0.07412	0.08543	0.90278
Fn_takeoff_Hot	0.14998	0.05951	0.07233	0.9127

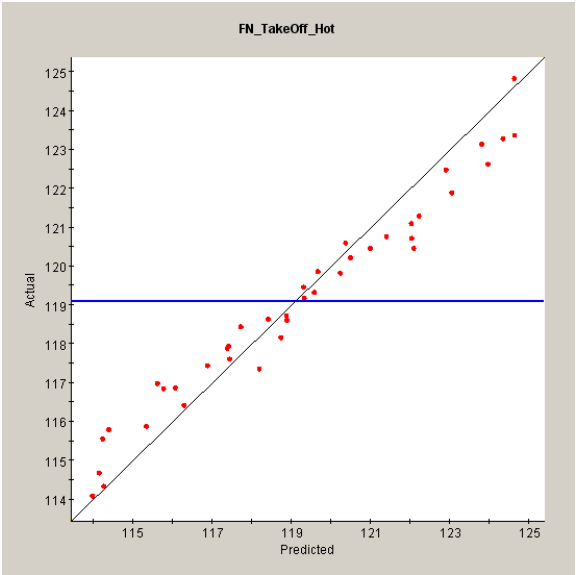
Στα Σχήματα 4.11, 4.12 και 4.13 φαίνεται η εφαρμογή των αποκρίσεων των παραμέτρων εξόδου. Η διαγώνια γραμμή αντιπροσωπεύει την τέλεια εφαρμογή. Η μπλε οριζόντια δείχνει την μέση τιμή της απόκρισης χρησιμοποιώντας δεδομένα από το πραγματικό δείγμα σφάλματος.



Σχήμα 4.11 Response fit of SFC on cruise

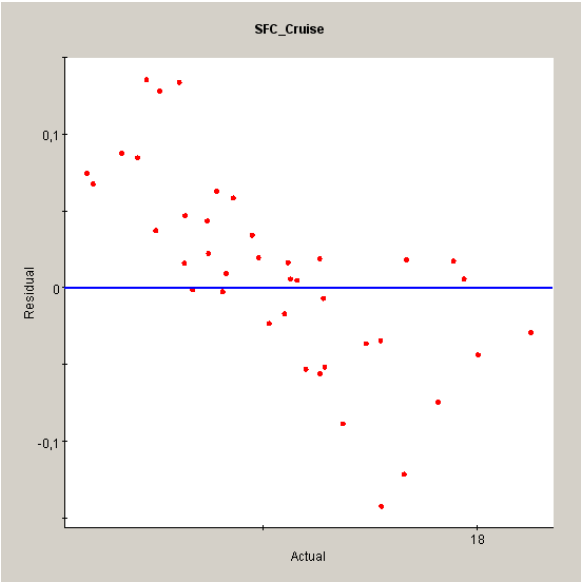


Σχήμα 4.12 Response fit of T41m

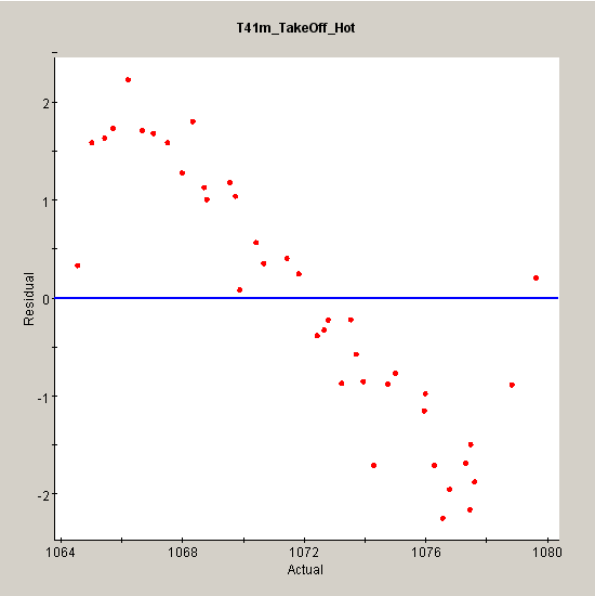


Σχήμα 4.13 Response fit of FN

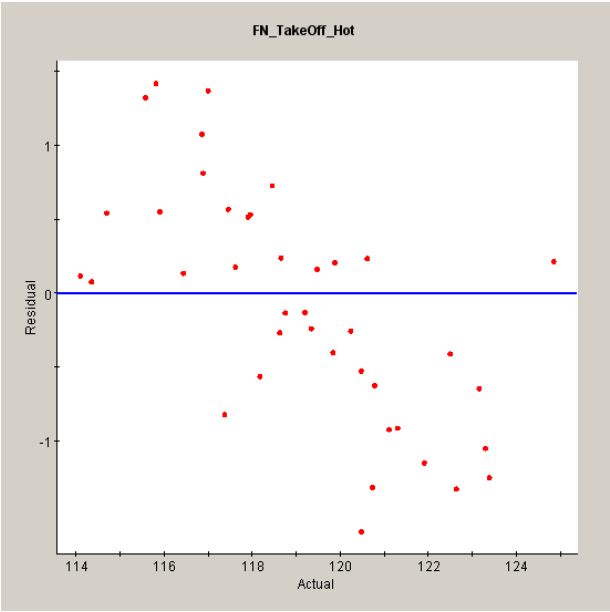
Στα Σχήματα 4.14, 4.15 και 4.16 φαίνονται τα υπόλοιπα, δηλαδή η διαφορά μεταξύ πραγματικών και προβλεπόμενων τιμών για όλα τα σημεία δειγματοληψίας και για κάθε απόκριση. Η οριζόντια μπλε γραμμή στο 0 αντιπροσωπεύει 0 υπόλοιπο ή σφάλμα πρόβλεψης 0. Αυτά τα σχήματα μπορούν να αναγνωρίσουν απόμακρα σημεία ή τάσεις στα υπόλοιπα. Ιδανικά η κατανομή των σημείων στα υπόλοιπα, θα πρέπει να είναι τυχαία.



Σχήμα 4.14 Residual graph of SFC on Cruise



Σχήμα 4.15 Residual Graph of T41m



Σχήμα 4.16 Residual Graph of FN

Η βελτιστοποίηση με χρήση των μεταπροτύπων kriging έγινε με την χρήση διαδοχικού τετραγωνικού προγραμματισμού (NLPQL) που έχει αναλυθεί στο κεφάλαιο 3. Ο αλγόριθμος χρησιμοποιεί εμπρός διαφορές για την εύρεση των παραγώγων. Οι παράμετροι του αλγόριθμου φαίνονται στον Πίνακα 4.10. Τα αποτελέσματα της βελτιστοποίησης φαίνονται στον Πίνακα 4.11. Επίσης στον ίδιο πίνακα παρουσιάζεται και η σύγκριση των προβλεπόμενων τιμών των εξόδων με τις πραγματικές, για τις ίδιες τιμές των μεταβλητών σχεδίασης.

Πίνακας 4.10

NLPQL	
Max Iterations	100
Termination Accuracy	1,0E-6
Rel Step Size	1,0E-6
Min Abs Step Size	1,0E-6
Use Central Differences	FALSE
Max Failed Runs	-1
Failed Run Penalty Value	1,0E30
Failed Run Objective Value	1,0E30

Πίνακας 4.11

Kriging	Αντικειμενική	Μεταβλητές Σχεδιασμού			Περιορισμοί	
	SFC cruise	d_FANOUT_PR	d_W1	d_W23	FN_TakeOff_Hot	T41m_TakeOff_Hot
Μεταπρότυπο	16.9805725	3.559397167	3.843557299	-3.44682216	118.7384083	1077.538271
Πραγματικό	17.29765084	3.559397167	3.843557299	-3.44682216	123.5987343	1077.032315
Απόκλιση %	-1.833071642				-3.932342901	0.046976881

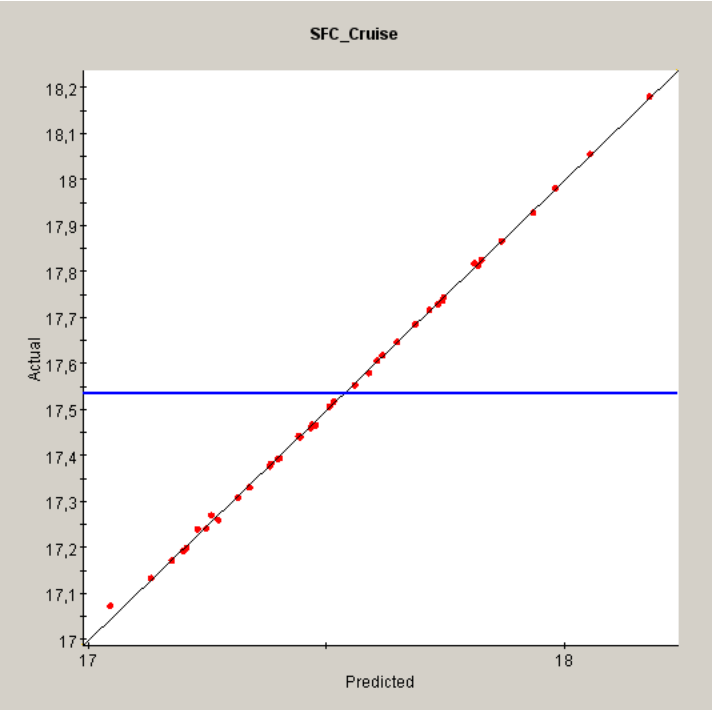
RBF μεταπρότυπα

Η RBF προσέγγιση είναι ένας τύπος νευρωνικού δικτύου που χρησιμοποιεί ακτινικές μονάδες. Οι μεταβλητές σχεδιασμού και οι έξοδοι που επιλέχθηκαν είναι οι ίδιες όπως και στην προηγούμενη περίπτωση. Για την κατασκευή του μεταπροτύπου χρησιμοποιήθηκε η επιλογή Smoothing Filter. Ουσιαστικά πρόκειται για ένα φίλτρο το οποίο αποκόπτει απομακρυσμένα σημεία. Το φίλτρο λειτουργεί στον Ευκλείδειο χώρο με κανονικοποιημένους τομείς στο [0,1]. Η μέγιστη επιτρεπόμενη τιμή για το φίλτρο είναι 0.1. Αυτό σημαίνει ότι υπάρχουν το πολύ 10 σημεία το ένα πολύ κοντά στο άλλο για κάθε είσοδο. Η τιμή που επιλέχθηκε για την προσέγγιση είναι 0.05, έτσι ώστε να μην είναι αρκετά aggressive. Ο αριθμός σημείων δειγματοληψίας και ο αριθμός των σημείων για cross-validation είναι ίδιος όπως και στην προηγούμενη περίπτωση. Τα αποτελέσματα της ανάλυσης σφάλματος φαίνονται στον Πίνακα 4.12.

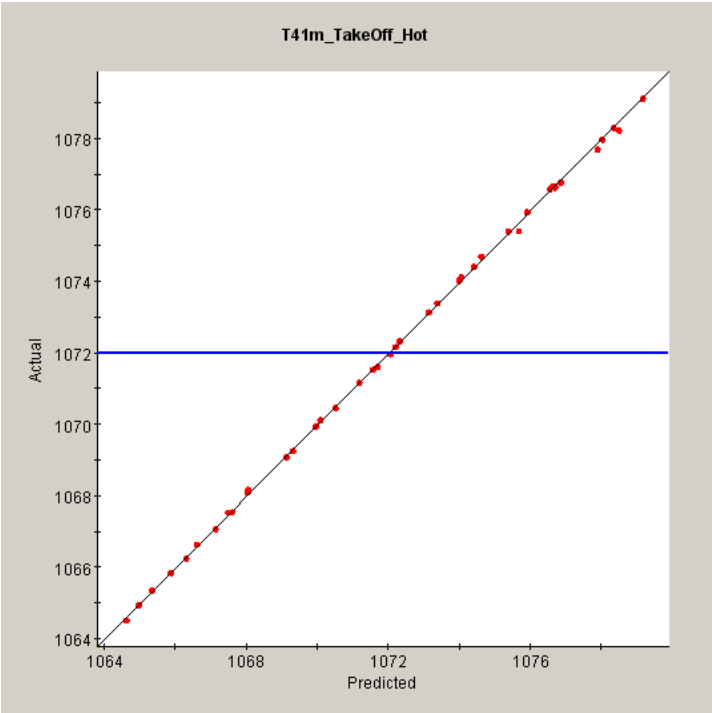
Πίνακας 4.12

RBF	Error Type			
Outputs	Maximum	Average	Root Mean Square	R Square
Sfc_cruise	0.0269	0.0044	0.00637	0.99931
T41m_takeoff_hot	0.01661	0.00393	0.00543	0.99965
Fn_takeoff_Hot	0.02577	0.00503	0.00699	0.99926

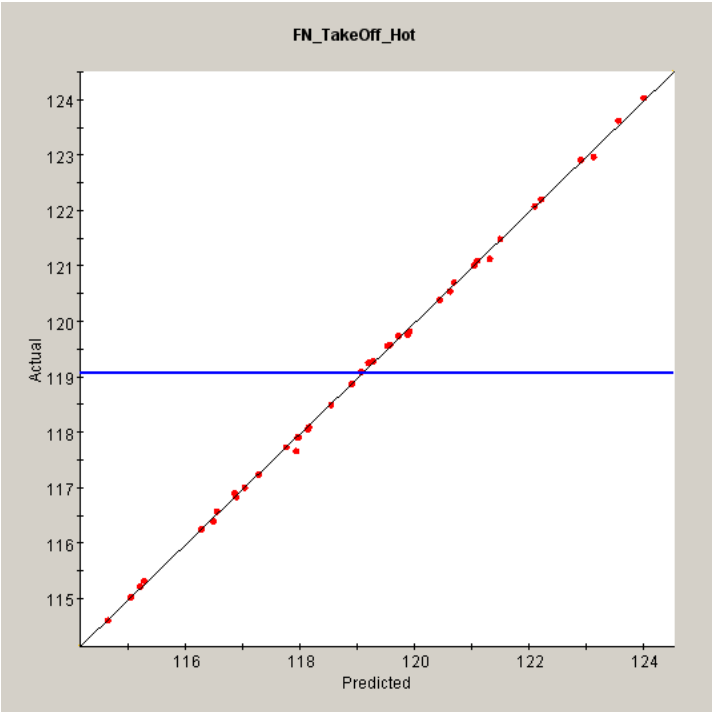
Στα σχήματα 4.17-4.22 παρουσιάζονται οι αποκρίσεις των αποκρίσεων των εξόδων και ατ αντίστοιχα σφάλματα. Επίσης στον πίνακα 4.13 παρουσιάζονται τα αποτελέσματα της βελτιστοποίησης με χρήση των μεταπροτύπων.



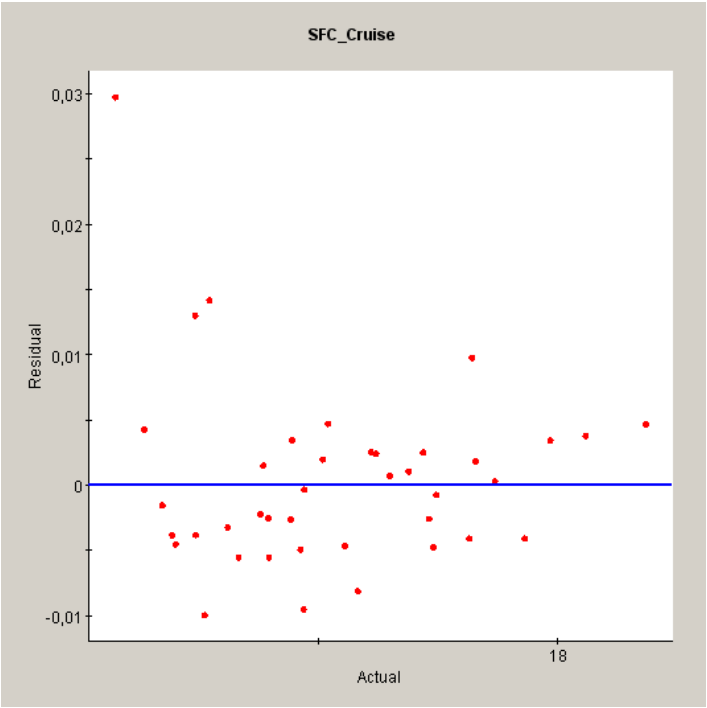
Σχήμα 4.17 Response fit of SFC



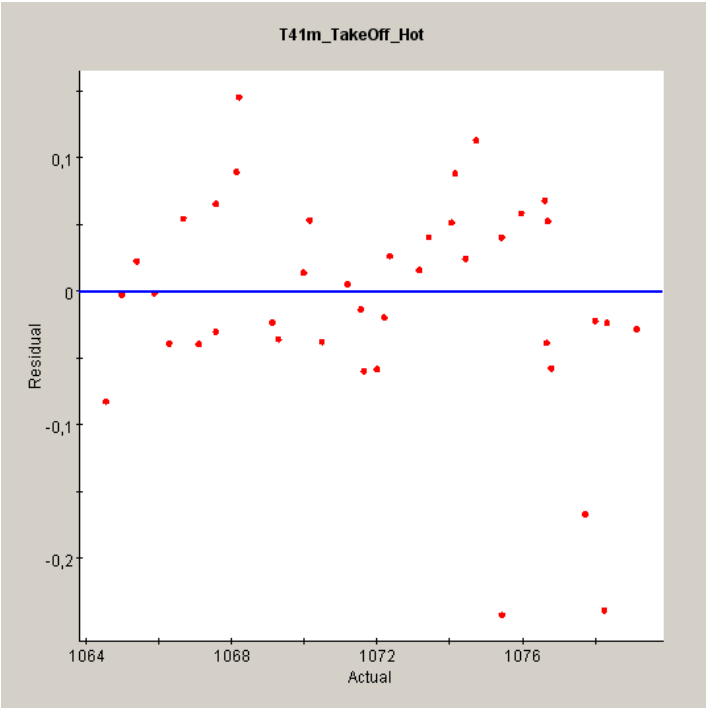
Σχήμα 4.18 Response fit of T41m



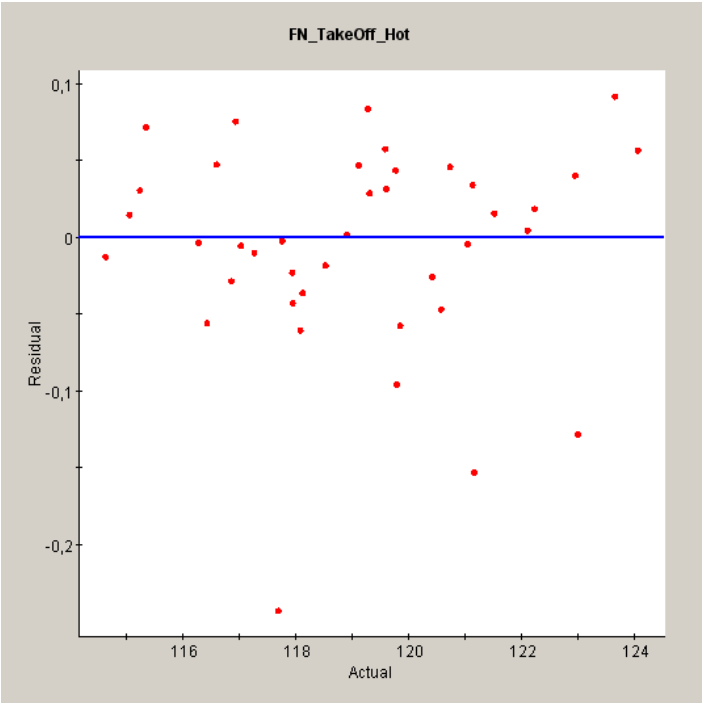
Σχήμα 4.19 Response fit of FN



Σχήμα 4.20 Residual Graph of SFC



Σχήμα 4.21 Residual Graph of T41m



Σχήμα 4.22 Residual Graph of FN

Πίνακας 4.13

RBF	Αντικειμενική	Μεταβλητές Σχεδιασμού			Περιορισμοί	
	SFC cruise	d_FANOUT_PR	d_W1	d_W23	FN_TakeOff_Hot	T41m_TakeOff_Hot
Μεταπρότυπο	16.89960501	5	5	-4.999995	117.369075	1077.043377
Πραγματικό	17.00030904	5	5	-4.999995	116.7012685	1076.279713
Απόκλιση %	-0.592365896				0.572235894	0.07095406

4.4 Σύγκριση και Ανάλυση των αποτελεσμάτων

Στον Πίνακα 4.14 γίνεται η σύγκριση των αλγορίθμων μεταξύ τους. Η βέλτιστη λύση φαίνεται να επιτυγχάνεται από τους αλγόριθμους PSO και Downhill Simplex. Ωστόσο, και οι υπόλοιποι αλγόριθμοι προσεγγίζουν με μικρή απόκλιση την βέλτιστη λύση.

Πίνακας 4.14

		Μταβήτές Σχεδιασμού			Αντικειμενική	Περιορισμοί	
Algorithm	Run	d_FANOUT_PR	d_W1	d_W23	SFC_Cruise	FN_TakeOff_Hot	T41m_TakeOff_Hot
PSO	39	5	5	-5	17.00030897	116.7012637	1076.279711
SIMPLEX	260	5	5	-5	17.00030897	116.7012637	1076.279711
ASA	499	4.998703502	4.9989106	-4.999342807	17.00033794	116.7021245	1076.278991
MIGA	450	4.691615167	4.952849622	-4.985198749	17.00199577	116.8338096	1075.990975
Reference		0	0	0	17.50022413	119.4788024	1072.342748

Στον Πίνακα 4.15 παρουσιάζονται οι επί τοις εκατό διαφορές στις εξόδους, για κάθε αλγόριθμο. Τα αποτελέσματα είναι αρκετά κοντά μεταξύ τους κάτι που ήταν αναμενόμενο καθώς το συγκεκριμένο πρόβλημα είναι αρκετά απλό, λόγω του μικρού αριθμού μεταβλητών σχεδίασης και περιορισμών καθώς και του εύρους των τιμών που γίνεται η αναζήτηση.

Πίνακας 4.15

Απόκλισεις %	Objective Function	Constraints	
	SFC_Cruise	FN_TakeOff_Hot	T41m_TakeOff_Hot
PSO	-2.856621494	-2.324712497	0.367136666
SIMPLEX	-2.856621494	-2.324712497	0.367136666
ASA	-2.856455945	-2.323992037	0.367069527
MIGA	-2.846982732	-2.213775765	0.340210891

Από τα σχήματα των αποκρίσεων των εξόδων σε κάθε μεταπρότυπο φαίνεται η καλύτερη εφαρμογή των RBF έναντι του Kriging. Επιπλέον το ιδανικό γράφημα υπολοίπου πρέπει να είναι τυχαίο. Στην συγκεκριμένη περίπτωση μεγαλύτερη έλλειψη μοτίβου υπάρχει και πάλι στο RBF μεταπρότυπο.

Στον Πίνακα 4.16 παρουσιάζονται τα αποτελέσματα των αναλύσεων σφαλμάτων των μεταπροτύπων Kriging και RBF μεταξύ τους.

Πίνακας 4.16

Error Analysis	Error Type							
	Kriging				*RBF*			
Outputs	Maximum	Average	RMS	R^2	Maximum	Average	RMS	R^2
Sfc_cruise	0.13691	0.0466	0.0605	0.94085	0.0269	0.0044	0.00637	0.99931
T41m_takeoff_hot	0.14864	0.07412	0.08543	0.90278	0.01661	0.00393	0.00543	0.99965
Fn_takeoff_Hot	0.14998	0.05951	0.07233	0.9127	0.02577	0.00503	0.00699	0.99926

Όπως παρατηρείται το μέγιστο και μέσο σφάλμα είναι μεγαλύτερο στην περίπτωση του Kriging σε σχέση με των RBF. Ακόμη το RMS του Kriging μεταπροτύπου είναι μια τάξη μεγέθους μεγαλύτερο κάτι που μπορεί να φανεί και από τα σχήματα των υπολοίπων. Τέλος το R^2 των RBF είναι μεγαλύτερο κάτι που δείχνει πόσο καλή εφαρμογή έχει κάνει το μεταπρότυπο. Στους Πίνακες 4.12 και 4.13 επισημαίνονται στην πράξη η διαφορά της ακρίβειας κάθε μεταπροτύπου. Τα RBF βρήκαν την καλύτερη λύση και με μικρότερη απόκλιση από το πραγματικό μοντέλο σε σχέση με το Kriging.

Η καλύτερη λύση κατά την βελτιστοποίηση, βρίσκεται στα άκρα των μεταβλητών σχεδίασης. Στον Πίνακα 4.17 φαίνεται η βέλτιστη λύση.

Πίνακας 4.17

Design Variables			Objective Function	Constraints	
d_FANOUT_PR	d_W1	d_W23	SFC_Cruise	FN_TakeOff_Hot	T41m_TakeOff_Hot
5	5	-5	17.00030897	116.7012637	1076.279711
Διαφορά επί % από την τιμή αναφοράς			-2.856621494	-2.324712497	0.367136666

Ο λόγος πίεσης του ανεμιστήρα και η παροχή αέρα του αγωγού εισόδου, δηλαδή το μέγεθος του ανεμιστήρα, αυξάνονται όσο είναι επιτρεπτό. Ενώ η παροχής του αέρα στην είσοδο του συμπιεστή χαμηλής πίεσης, δηλαδή το μέγεθος του πυρήνα μειώνεται όσο είναι επιτρεπτό. Η βελτίωση της SFC είναι περίπου 2.85% ενώ η ώση κατά την απογείωση μειώνεται. Η θερμοκρασία μετάλλου στα κινητά πτερύγια του στροβίλου παραμένει σχεδόν σταθερή. Στον Πίνακα 4.18 φαίνονται πως μεταβάλλονται διάφορα μεγέθη, για τις συγκεκριμένες τιμές των μεταβλητών σχεδιασμού, σε σχέση με τις συνθήκες αναφοράς.

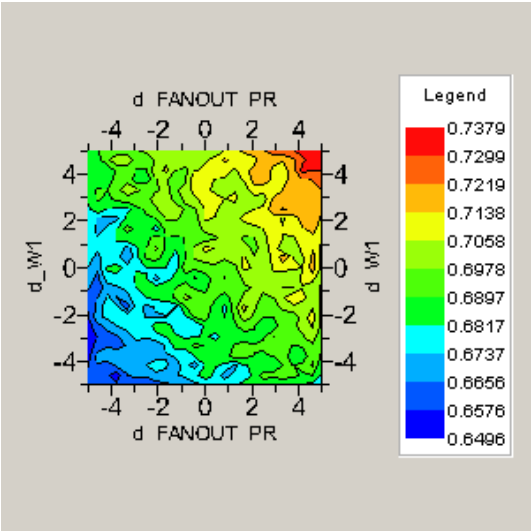
Πίνακας 4.18 Απόκλιση επί τοις εκατό από τις συνθήκες αναφοράς

ETAthermal	ETApropulsive	ETAoverall	Cruise_SFN	OPR_Cruise	FN_Cruise
-4.145518783	7.392604586	2.940623991	-7.917901348	4.657710592	-3.754545251

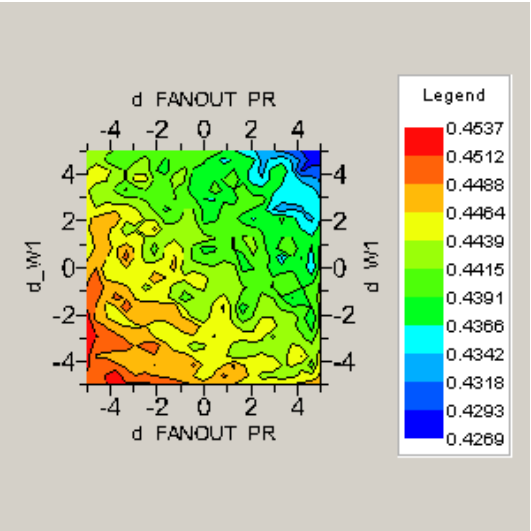
Όπως παρατηρείται ο θερμικός βαθμός απόδοσης του κινητήρα μειώνεται ενώ ο προωθητικός του αυξάνεται σημαντικά. Ο συνολικός βαθμός, ωστόσο αυξάνεται περίπου 3%.

Ο θερμικός βαθμός απόδοσης είναι ένα μέγεθος που σχετίζεται με την ποιότητα του θερμοδυναμικού κύκλου της μηχανής. Εκφράζει το πόσο αποδοτικά μετατρέπει ο κινητήρας τη θερμική ενέργεια του καυσίμου σε κινητική ενέργεια καυσαερίων για πρόωση. Η αύξηση της θερμικής απόδοσης μπορεί να επιτευχθεί με την αλλαγή του κύκλου της μηχανής, κυρίως με την αύξηση του συνολικού λόγου πίεσης και με την αύξηση της θερμοκρασίας εισόδου στον στρόβιλο. Μπορεί ακόμη να αυξηθεί με την χρήση προηγμένων τεχνολογιών όπως χρήση intercooler και recuperators.

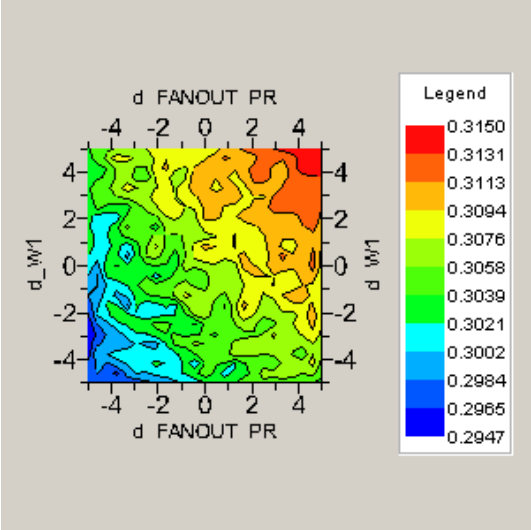
Ο προωθητικός βαθμός απόδοσης, στους αεροπορικούς κινητήρες, είναι ένα μέτρο αποτελεσματικότητας με το οποίο η διαθέσιμη ενέργεια του πυρήνα, δηλαδή η συνολική ώση και η αχρησιμοποίητη κινητική ενέργεια της δέσμης αέρα, χρησιμοποιείται για την προώθηση του αεροσκάφους. Η προωθητική απόδοση βελτιώνεται μειώνοντας την ειδική ώση (SFN), που αυτό γίνεται συνήθως αυξάνοντας τον λόγο παράκαμψης και μειώνοντας τον λόγο πίεσης του ανεμιστήρα. Ο συνδυασμός των μεταβλητών σχεδίασης W1 και W23 δίνει τον λόγο παράκαμψης. Στα σχήματα 4.23-4.25 παρουσιάζεται η επίδραση του FANAOUT_PR και του W1 στον προωθητικό, στο θερμικό και στον συνολικό βαθμό απόδοσης αντίστοιχα..



Σχήμα 4.23 Επίδραση των Fanaout_PR και W1 στον προωθητικό βαθμό απόδοσης (ETAprop)



Σχήμα 4.24 Επίδραση των Fanaout_PR και W1 στον θερμικό βαθμό απόδοσης (ETAth)



Σχήμα 4.25 Επίδραση των Fanaout_PR και W1 στον συνολικό βαθμό απόδοσης (ETAovr)

Στην συγκεκριμένη εφαρμογή με την αύξηση του προωθητικού βαθμού, μειώθηκε και η ειδική ώση (SFN) κάτι που ήταν και το αναμενόμενο. Η ειδική ώση, είναι η ώση ανά μονάδα παροχής αέρα, που είναι ένα μέτρο μέτρησης της ταχύτητας της δέσμης αέρα. Η πρόωση είναι πιο αποδοτική αν γίνεται με μικρή επιτάχυνση μεγάλων μαζών αέρα. Επίσης, ο ολικός λόγος πίεσης αυξάνεται κάτι το οποίο είναι λογικό, καθώς έτσι αυξάνεται η θερμική απόδοση του κινητήρα. Όπως παρατηρείται, για την μείωση της SFC αυξάνεται η διάμετρος του ανεμιστήρα και ο λόγος παράκαμψης, ουσιαστικά η μηχανή τείνει να μεγαλώσει. Ωστόσο, αυτό δημιουργεί άλλα προβλήματα όταν ενσωματωθεί ο κινητήρας σε ένα πραγματικό αεροσκάφος, Το μεγάλο μέγεθος της μηχανής αυξάνει την οπισθέλκουσα του αεροσκάφους που αυτό συνεπάγεται σε

μεγαλύτερη κατανάλωση καυσίμου κατά την αποστολή του αεροσκάφους. Επιπλέον, το μεγάλο μέγεθος αυξάνει το βάρος του αεροσκάφους και μπορεί ακόμη να μην γίνεται να ενσωματωθεί στο αεροσκάφος λόγω κατασκευαστικών περιορισμών. Γι' αυτό η ανάλυση μόνο των παραμέτρων της μηχανής και όχι συνολικά των διάφορων συνιστωσών που αφορούν την συνολική αποστολή του αεροσκάφους είναι μη επαρκής για την βελτιστοποίηση του κινητήρα.

5 Εφαρμογή κατά την αποστολή του αεροσκάφους

Στο κεφάλαιο αυτό γίνεται εφαρμογή των ίδιων τεχνικών όπως και στην προηγούμενη περίπτωση όμως σε ένα πρόβλημα αρκετά πιο περίπλοκο και σύνθετο. Στην περίπτωση αυτή εκτός από την συνιστώσα του κινητήρα γίνεται χρήση και άλλων συνιστωσών όπως του αεροσκάφους με στόχο την μείωση της συνολικής κατανάλωσης καυσίμου στα πλαίσια μιας αποστολής.

5.1 Το εργαλείο TERA και η διάταξη GTCRSR

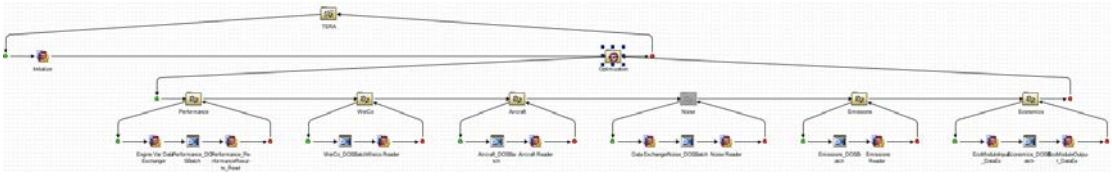
Για την ανάλυση των διαφόρων συνιστωσών του προβλήματος θα γίνει χρήση του λογισμικού TERA2020 [14]. Το TERA2020 αποτελεί ένα εργαλείο για την αξιολόγηση της απόδοσης καινοτόμων σχεδιασμών κινητήρων, στα πλαίσια της μείωσης της συνολικής κατανάλωσης καυσίμου της αποστολής του αεροσκάφους. Το εργαλείο αυτό ακολουθεί μια διαφορετική προσέγγιση σε σχέση με την τυπική προσέγγιση της βιομηχανίας. Στην βιομηχανία γίνεται πρώτα η βελτιστοποίηση του αεροσκάφους και στην συνέχεια η επιλογή της βέλτιστης μηχανής ανάλογα με τα χαρακτηριστικά του αεροσκάφους. Στην προκειμένη περίπτωση γίνεται η βελτιστοποίηση του κινητήρα σε συνδυασμό με το μοντέλο του αεροσκάφους για την εύρεση του κατάλληλου σχεδιασμού. Το εργαλείο χρησιμοποιεί ένα «rubberized» μοντέλο αεροσκάφους, δηλαδή η γεωμετρία και το βάρος του αεροσκάφους προκύπτει ανάλογα με την απόδοση του κινητήρα. Οι διαστάσεις της ατράκτου του αεροπλάνου, ωστόσο παραμένουν σταθερές και οι συνιστώσες που διαφοροποιούνται είναι τα φτερά.

Ο κινητήρας ο οποίος θα μελετηθεί στην συγκεκριμένη εφαρμογή είναι ο GTCRSR, για τον οποίο ήδη έχει γίνει βελτιστοποίηση από τον Alexiou [17]. Ωστόσο έχει χρησιμοποιηθεί μια μόνο τεχνική για την βελτιστοποίηση του προβλήματος και δεν γνωρίζουμε αν αυτή είναι η ιδανικότερη.. Η ύπαρξη ήδη της συγκεκριμένης βελτιστοποίησης δίνει την δυνατότητα να υπάρχει ένα σημείο αναφοράς ως προς το ποια είναι η βέλτιστη λύση και αν γίνεται να βρεθεί μια ακόμη καλύτερη.

Οι συνιστώσες που περιλαμβάνονται στο TERA δημιουργήθηκαν από διάφορα πανεπιστήμια μεταξύ αυτών και του ΕΘΣ/ΕΜΠ. Η πρώτη συνιστώσα είναι το Engine Performance, το οποίο αναπτύχθηκε από το ΕΘΣ και μοντελοποιεί τον κινητήρα. Η επόμενη συνιστώσα είναι το WEICO το οποίο μοντελοποιεί τις διαστάσεις και στην συνέχεια το βάρος του αεροπορικού κινητήρα. Η επόμενη είναι το AIRCRAFT το οποίο ανάλογα με τα αποτελέσματα του WEICO, υπολογίζει διάφορα δεδομένα για την αποστολή του αεροσκάφους. Στη συνέχεια το EMISSIONS υπολογίζει τους ρύπους που

προκύπτουν κατά την αποστολή του αεροσκάφους και το NOISE τον θόρυβο που δημιουργείται σε διάφορα σημεία κατά την αποστολή. Στην συγκεκριμένη εφαρμογή όμως, η συνιστώσα NOISE δεν χρησιμοποιείται. Τέλος υπάρχει το ECONOMICS το οποίο υπολογίζει το κόστος συντήρησης της μηχανής για μια περίοδο 30 ετών. Η διαδικασία υπολογισμού δηλαδή είναι η εξής: ENGINE PERFORMANCE => WEICO => AIRCRAFT =>EMISSIONS και ENVIRONMENTAL IMPACT => NOISE => ECONOMICS.

Οι παραπάνω συνιστώσες ενσωματώθηκαν στο Isight για την σύνθεση του προβλήματος όπως φαίνεται στο Σχήμα 5.1.



Σχήμα 5.1 Η προσωμοίωση της ροής της διαδικασίας του προβλήματος στο γραφικό περιβάλλον του Isight

Στόχος της βελτιστοποίησης είναι η ελαχιστοποίηση της κατανάλωσης καυσίμου κατά την αποστολή του αεροσκάφους, επομένως η αντικειμενική συνάρτηση αποτελεί το Aircraft Block Fuel. Οι μεταβλητές σχεδιασμού στην συγκεκριμένη εφαρμογή ήταν πέντε. Πρόκειται για κάποιες βασικές αεροθερμοθυναμικές παραμέτρους που επηρεάζουν την λειτουργία του κινητήρα κατά την άνοδο (Top-of-climb) του αεροσκάφους. Αυτές είναι: οι dW1, dW23 και dFANAOUT_PR οι οποίες είναι ίδιες όπως και στην προηγούμενη εφαρμογή με τον κινητήρα CFM56 και οι λόγοι πίεσης του booster και του συμπιεστή υψηλής πίεσης, οι dIPC_PR και dHPC_PR αντίστοιχα. Όπως, τονίστηκε και στην προηγούμενη εφαρμογή ο συνδυασμός των μεταβλητών dW1 και dW23 καθορίζουν τον λόγο παράκαμψης (BPR). Οι μεταβλητές dIPC_PR, dHPC_PR και dFANAOUT_PR τον συνολικό λόγο πίεσης του κινητήρα (OPR). Αναλυτικά οι μεταβλητές σχεδιασμού παρουσιάζονται στον Πίνακα 5.1.

Πίνακας 5.1

Μεταβλητές Σχεδιασμού	
dW1	Inlet mass flow
dW23	Core mass flow
dFanBPPR	Fan bypass pressure ratio
dHPCPR	HPC pressure ratio
dIPCPR	Booster pressure ratio

Οι μεταβλητές σχεδιασμού μεταβάλλονται επί τοις εκατό σε σχέση με μια τιμή αναφοράς τους. Οι τιμές αναφοράς για τις σημαντικότερες παραμέτρους του προβλήματος παρουσιάζονται στον Πίνακα 5.2.

Πίνακας 5.2

Τιμές Αναφοράς του Προβλήματος					
BPR_ToC	11.7	-	BladeCreepLife	345375392	[hr]
W1_ToC	224.7	[kg/s]	BladeFatigueLife	36582.95121	[hr]
FANOUT_PR_ToC	1.547	-	BPRCR	12.90212099	-
LPC_PR_ToC	2.38	-	EngDryWt	1913.268723	[kg]
HPC_PR_ToC	15.1	-	EngWt	3176.650463	[kg]
AcBlockFuel	2716.267716	[kg]	ETAovCR	0.383534329	-
AcClimbTimeDes	14.20022256	[min]	ETAovTO	0.21023172	-
AcTODDes	1.703330395	[min]	ETAovTOC	0.378592149	-
HpcLastStgBHeight	0.014155103	[m]	ETApropCR	0.804266704	-
T30TO	898.0576877	[K]	ETApropTO	0.476434895	-
T40Idle	1123.868626	[K]	ETApropTOC	0.768925079	-
T41mTO	1105.561286	[K]	ETAthCR	0.476874558	-
TBO	32667.8259	[hrs]	ETAthTO	0.441260122	-
AcDescentTime	20.52499489	[min]	ETAthTOC	0.492365459	-
AcFuelEff	2.337901704	-	FNCR	22.09734609	[kN]
AcFuelWtClimb	805.3334942	[kg]	FNICAOTO	133.0460253	[kN]
AcFuelWtCruise	1053.955527	[kg]	FNTO	100.5069252	[kN]
AcFuelWtDescent	321.837808	[kg]	FNTOC	28.22714096	[kN]
AcOEWt	41860.51724	[kg]	OPRCR	39.18684214	-
AcToWtmax	69233.61959	[kg]	SFCCR	13.99205974	[g/(kN*s]
AcWingA	115.2951706	[kg]	SFNCR	101.786888	[N*s/kg]
AcWingSpan	32.91111825	[m]			

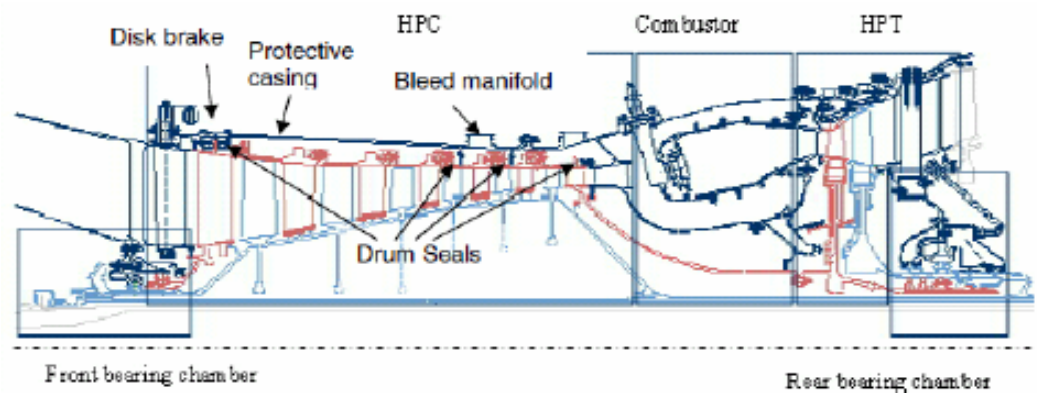
Όπως είναι αναμενόμενο η βελτιστοποίηση του προβλήματος δεν γίνεται χωρίς την ύπαρξη περιορισμών. Οι παράμετροι σχεδιασμού του Πίνακα 5.1 έχουν περιορισμένα όρια για διάφορους λόγους. Για παράδειγμα για δεδομένο μέγεθος πυρήνα, το FPR και BPR δεν μπορεί να ξεπεράσει κάποιο όριο γιατί ο ανεμιστήρας θα χρειαστεί περισσότερη ισχύ απ' ότι μπορεί να του προσδώσει ο booster. Ο συνολικός λόγος πίεσης ενός κινητήρα περιορίζεται από την μέγιστη θερμοκρασία που μπορούν να αντέξουν τα πτερύγια του στροβίλου υψηλής πίεσης και την μέγιστη θερμοκρασία που μπορεί να προσδώσει ο συμπιεστής υψηλής πίεσης. Επιπλέον η πυκνότητα του αέρα κατά την διαδρομή του αερίου αυξάνεται με αποτέλεσμα τα πτερύγια στον συμπιεστή τείνουν να μικρύνουν. Οι απώλειες που προκύπτουν από τα κενά στις κορυφές των πτερυγίων αυξάνονται σημαντικά, επομένως χρειάζεται ένας περιορισμός για το ελάχιστο ύψος των

πτερυγίων για να διατηρηθεί ο βαθμός απόδοσης του συμπιεστή σε ικανοποιητικά επίπεδα. Ακόμη δεν είναι δυνατή η κατασκευή πολύ μικρών πτερυγίων. Με την ύπαρξη υψηλών μέσων θερμοκρασιών κατά μήκος του κινητήρα, σε όλη την διάρκεια της αποστολής του αεροσκάφους μειώνεται η διάρκεια ζωής των στοιχείων του κινητήρα. Για αυτό τον λόγο έχει τεθεί ένας περιορισμός όσον αφορά τον χρόνο για κάθε γενική επιθεώρηση της μηχανής. Λιγότερος χρόνος που απαιτείται για κάθε γενική επιθεώρηση, μεταφράζεται σε υψηλότερο λειτουργικό κόστος. Η ώση του κινητήρα δεν είναι σταθερή, ενώ για την μείωση της κατανάλωσης καυσίμου η ώση τείνει να μειώνεται. Ωστόσο, με μικρή ώση δεν είναι δυνατόν το αεροσκάφος, που θα προκύψει να ικανοποιεί κάποιους κανονισμούς όσον αφορά την αποστολή του (Federal Aviation Regulations) σχετικά με το μήκος του διαδρόμου που απαιτείται για την απογείωση του και τον χρόνο που χρειάζεται για να φτάσει στο ύψος για την κρουαζιέρα. Επομένως έχουν τεθεί δύο περιορισμοί όσον αφορά το μέγιστο μήκος διαδρόμου απογείωσης και τον ελάχιστο χρόνο ανόδου στο ύψος που ξεκινάει την κρουαζιέρα. Τέλος έχει τεθεί ένας επιπλέον περιορισμός αυτός της θερμοκρασίας εισόδου στον στρόβιλο στο σημείο σχεδιασμού στο Idle για λόγους σύγκλισης του κώδικα Engine Performance. Οι περιορισμοί αναλυτικά φαίνονται στον Πίνακα 5.3

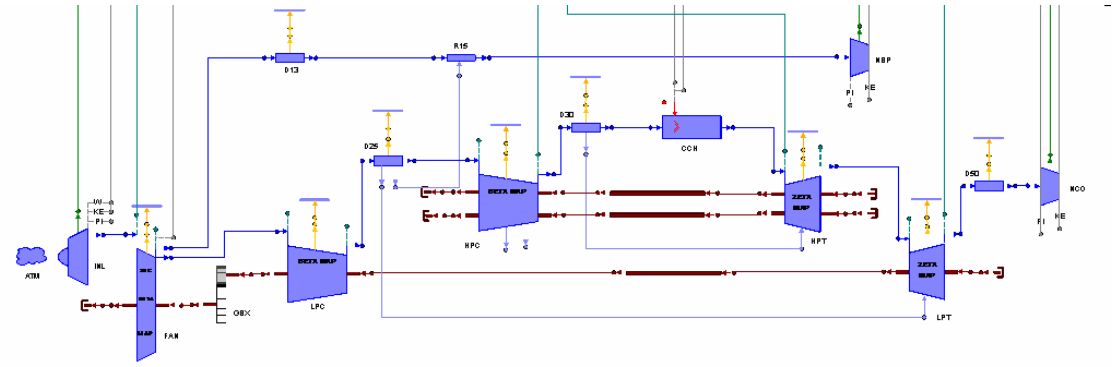
Πίνακας 5.3

Περιορισμοί-Constraints				
Parameters	Abbreviation	Lower Limit	Upper Limit	Units
FAR Take-off distance	AcTODDes	-	2,5	[km]
Climb to 35000 feet	AcClimbTimeDes	-	22,5	[min]
HPC delivery temperature	T30TO	-	970	[K]
Turbine blade mean metal temperature	T41mTO	-	1350	[K]
Turbine inlet temperature at Idle	T40Idle	-	1350	[K]
HPC last stage blade height	HpcLastStgBHeight	0,01	-	[m]
Time Between Overhaul	TBO	23000	-	[hrs]

Το μοντέλο του GTCRCSR δημιουργήθηκε στο Proosis από το ΕΘΣ και εξήχθη σε μορφή deck, όπως και στην προηγούμενη εφαρμογή με παρόμοια αρχεία εισόδου και εξόδου. Στο Σχήμα 5.2 φαίνεται η δομή του πυρήνα (CRC) του κινητήρα, ενώ στο Σχήμα 5.3 φαίνεται το σχηματικό του GTCRSR στο περιβάλλον του Proosis.



Σχήμα 5.2 Η δομή του πυρήνα (CRC) του κινητήρα GTCRSR (από [41])



Σχήμα 5.3 PROOSIS schematic of GTCRSR configuration (από [17])

5.2 Εφαρμογή των τεχνικών βελτιστοποίησης

Σε προηγούμενες βελτιστοποιήσεις με το λογισμικό αξιολόγησης TERA2020 [14], οι ερευνητές πρότειναν τον αλγόριθμο βελτιστοποίησης ASA , λόγω των σημαντικών πλεονεκτημάτων της όπως αναφέρθηκε προηγουμένως. Ωστόσο, εφαρμόστηκαν διαφορετικές προσεγγίσεις ανάλογα με την διάταξη του κινητήρα. Στις περισσότερες περιπτώσεις εφαρμόστηκε πρώτα μια μελέτη ευαισθησίας για την κατανόηση της επίδρασης των μεταβλητών σχεδίασης στην κατανάλωση καυσίμου και ύστερα έγινε η βελτιστοποίηση. Σε κάποιες άλλες, μεταξύ αυτών και στην περίπτωση του GTACSR έγινε βηματική βελτιστοποίηση. Οι ερευνητές χώρισαν τις μεταβλητές σχεδιασμού ανά ομάδες και έκανα βελτιστοποίηση ανά βήματα. Στο πρώτο βήμα βρήκαν το σετ βέλτιστων τιμών της πρώτης ομάδας και στην συνέχεια κρατώντας τις τιμές σταθερές έκαναν βελτιστοποίηση στην επόμενη ομάδα. Η βελτιστοποίηση συνεχίστηκε μέχρι να βελτιστοποιηθούν όλες οι μεταβλητές σχεδίασης. Αργότερα, εφάρμοσαν καθολική βελτιστοποίηση, επιτρέποντας όλες τις μεταβλητές να μεταβληθούν ταυτόχρονα. Τα αποτελέσματα της καθολικής βελτιστοποίησης ήταν καλύτερα από την «χειροκίνητη» μέθοδο. Αυτό οφείλεται στο ότι οι μεταβλητές σχεδιασμού επηρεάζουν η μια την άλλη δημιουργώντας το λεγόμενο «snowball effect».

Για την βελτιστοποίηση της GTCRCSR διάταξης πρώτα έγινε μια ανάλυση ευαισθησίας όπου οι μεταβλητές σχεδιασμού μεταβλήθηκαν [-2%,+2%] από τη τιμή αναφοράς τους, για να φανεί η τάση τους σχετικά με την συνολική κατανάλωση καυσίμου. Στη συνέχεια, έγινε η βελτιστοποίηση με τις μεταβλητές να έχουν περιορισμένα όρια ανάλογα με τα αποτελέσματα της ανάλυσης ευαισθησίας. Φυσικά, υπήρχαν και οι περιορισμοί που αναφέρθηκαν προηγουμένως. Ο αλγόριθμος βελτιστοποίησης ήταν ο ASA. Το βέλτιστο σημείο που προέκυψε οδήγησε σε μείωση της συνολικής κατανάλωσης καυσίμου κατά την διάρκεια της αποστολής του αεροσκάφους κατά 7.31%. Τέλος, για την επιβεβαίωση ότι η λύση είναι το καθολικό ελάχιστο, δημιουργήθηκε DOE όπου οι μεταβλητές σχεδιασμού μεταβλήθηκαν [-5%,+5%] από την βέλτιστη τιμή τους.

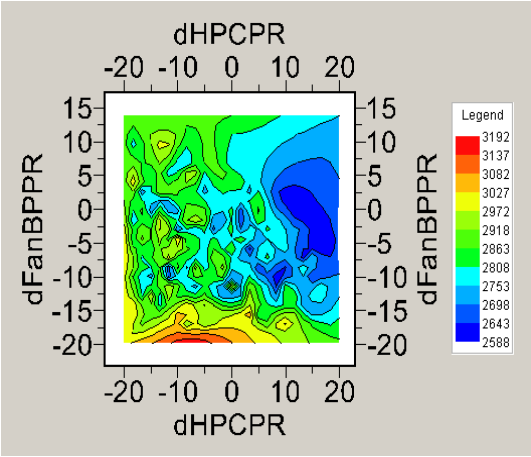
Στην παρούσα εργασία θα γίνει εφαρμογή των ίδιων τεχνικών όπως και στην περίπτωση του κινητήρα CFM56.

5.2.1 Optimal Latin Hypercube και Sensitivity Analysis

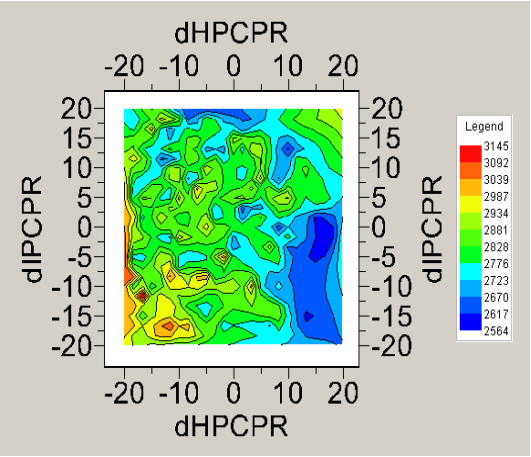
Για την δημιουργία του DOE χρησιμοποιήθηκε ξανά η Optimal Latin Hypercube για του λόγους που διατυπώθηκαν στην πρώτη εφαρμογή. Στην συγκεκριμένη περίπτωση λόγω της μεγάλης πολυπλοκότητας του προβλήματος επιλέχθηκε μεγαλύτερος αριθμός σημείων σχεδιασμού. Ο χρόνος που δόθηκε για βελτιστοποίηση του τυχαίου Latin Hypercube ήταν 10 min. Ο αριθμός των σημείων που επιλέχθηκε είναι 1500, ωστόσο επειδή οι περισσότεροι σχεδιασμοί αποτυγχάνουν κατά την εκτέλεση, δημιουργήθηκε χώρος σχεδιασμού των πειραμάτων μόνο με 318 σχεδιασμούς. Στα Σχήματα 5.4, 5.5 και 5.6 φαίνονται κάποια από τα αποτελέσματα της δημιουργίας του DOE.

Όπως είναι αναμενόμενο λόγω του μικρού αριθμού σχεδιασμών τα αποτελέσματα δεν είναι ικανοποιητικά, ενώ επίσης αναδεικνύεται μια αδυναμία κατανόησης της απεικόνισης των αποτελεσμάτων σε ένα μεγάλο εύρος των μεταβλητών σχεδίασης. Όπως φαίνεται από τα σχήματα ο χώρος σχεδιασμού του προβλήματος είναι ιδιαίτερα κατακερματισμένος και δεν διακρίνεται ποιο εύρος των τιμών να επιλεγεί για εξοικονόμιση υπολογιστικού χρόνου. Γι αυτό τον λόγο έγινε μια ανάλυση ευαισθησίας, με μεταβολή των τιμών των μεταβλητών σχεδιασμού μια την φορά διατηρώντας τις υπόλοιπες μεταβλητές σταθερές.

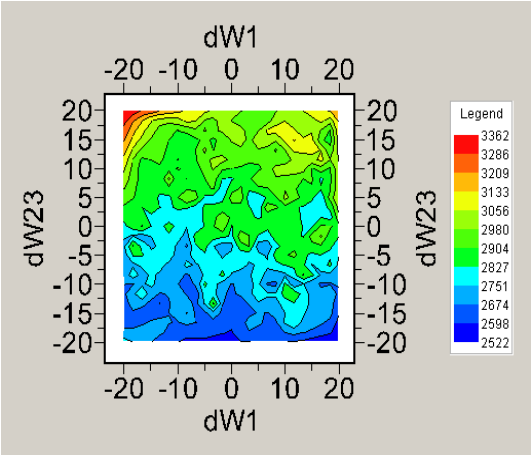
Για την ανάλυση ευαισθησίας επιλέχθηκαν 30 επίπεδα σχεδιασμού, ενώ το εύρος της κάθε μεταβλητής ήταν [+2%,-2%] από την τιμή αναφοράς της. Στα Σχήματα 5.7-5.11 φαίνονται τα αποτελέσματα για κάθε μεταβλητή σχεδιασμού σε σχέση με την αντικειμενική συνάρτηση, AcBlockFuel.



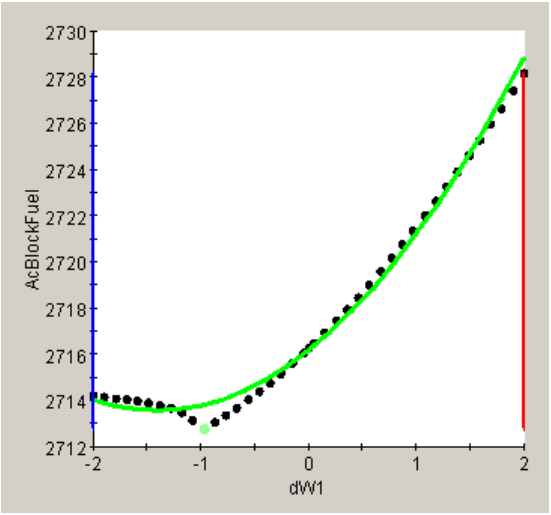
Σχήμα 5.4 DOE των μεταβλητών HPCPR-FPR



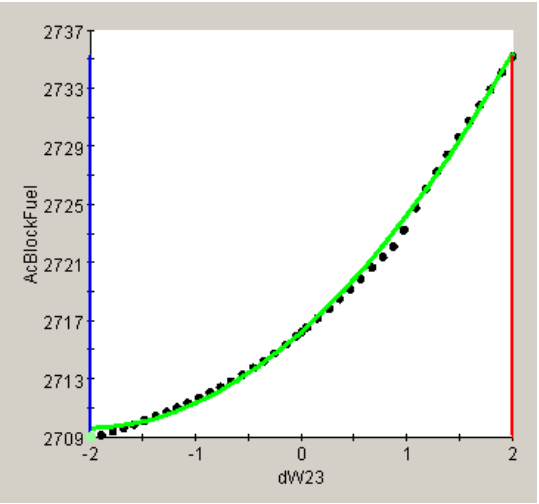
Σχήμα 5.5 DOE των μεταβλητών HPCPR-IPCPR



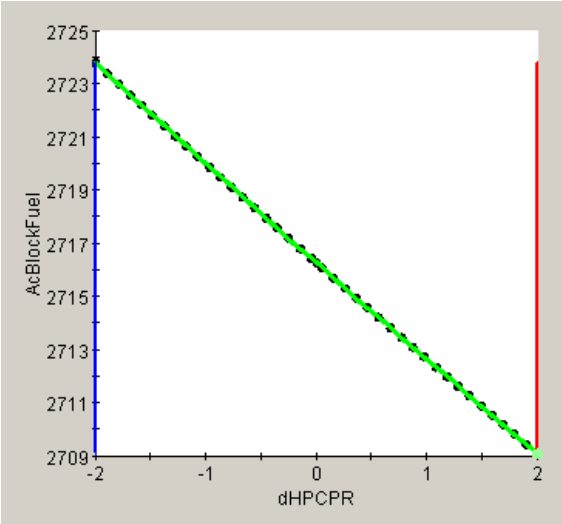
Σχήμα 5.6 DOE των μεταβλητών W1-W23



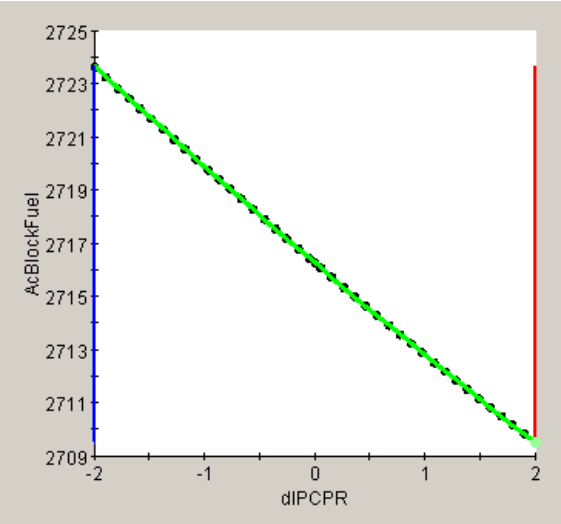
Σχήμα 5.7 Sensitivity analysis της W1



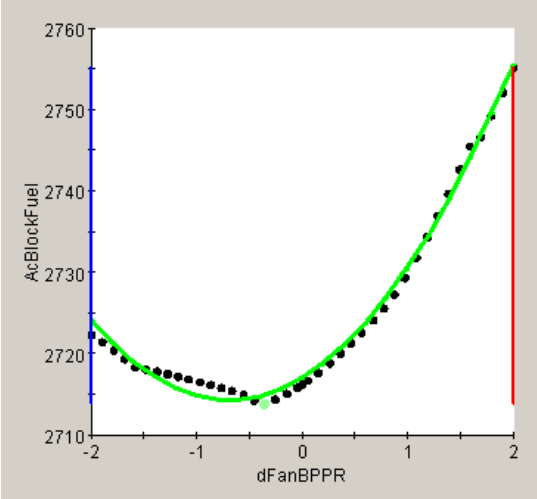
Σχήμα 5.8 Sensitivity analysis της W23



Σχήμα 5.9 Sensitivity analysis της HPCPR



Σχήμα 5.10 Sensitivity analysis της IPCPR



Σχήμα 5.11 Sensitivity analysis της FPR

Τα αποτελέσματα από την ανάλυση ευαισθησίας δείχνουν ότι οι μεταβλητές σχεδιασμού dW1, dW23 και dFanBPPR θέλουν να μειωθούν σε σχέση με την τιμή αναφοράς τους. Σε αντίθετη κατεύθυνση τείνουν οι λόγοι πίεσης του συμπιεστή υψηλής πίεσης και του booster dHCPR και dIPCPR. Τα αποτελέσματα θα χρησιμοποιηθούν για την μείωση του εύρους αναζήτησης στις μεταβλητές σχεδιασμού.

5.3 Αλγόριθμοι Βελτιστοποίησης

Οι αλγόριθμοι βελτιστοποίησης που θα εφαρμοστούν θα είναι οι ίδιοι όπως και στην εφαρμογή του CFM56 με την προσθήκη μόνο του inhouse λογισμικού βελτιστοποίησης EASY [40]. Στην συγκεκριμένη εφαρμογή θα εξεταστούν δύο εύρη αναζήτησης, ένα μικρό με βάση τα αποτελέσματα της ανάλυσης ευαισθησίας και της λύσης που έχει βρεθεί από την βελτιστοποίηση που έχει πραγματοποιηθεί στο παρελθόν και ένα διευρυμένο χωρίς γνώση της προηγούμενης βέλτιστης λύσης. Στον Πίνακα 5.4 φαίνονται τα όρια των δύο βελτιστοποιήσεων. Η αρχική τιμή της κάθε μεταβλητής βρίσκεται στο μέσο των ορίων της.

Πίνακας 5.4

	Μικρό εύρος			Μεγάλο εύρος		
Πεδίο αναζήτησης	Lower Limit	Uper Limit	Initial Value	Lower Limit	Uper Limit	Initial Value
dFanBPPR	-10	0	-5	-20	20	0
dHPCPR	0	20	10	-20	20	0
dIPCPR	0	20	10	-20	20	0
dW1	-20	0	-10	-20	20	0
dW23	-20	0	-10	-20	20	0

5.3.1 Adaptive Simulated Annealing

Ο πρώτος αλγόριθμος που χρησιμοποιήθηκε, όπως και στην προηγούμενη εφαρμογή είναι ο ASA. Οι ρυθμίσεις του αλγόριθμου στο λογισμικό είναι ίδιες όπως και στην εφαρμογή του CFM56, με μόνη διαφορά την παράμετρο Num. of Des. for Conv. Check όπου παίρνει την τιμή 3, δηλαδή τρεις σχεδιασμοί για έλεγχο σύγκλισης του αλγόριθμου. Οι ρυθμίσεις του αλγόριθμου παρουσιάζονται στον Πίνακα 5.5.

Πίνακας 5.5

Optimization Technique:	ASA
Convergence Epsilon	0.000001
Failed Run Objective Value	1E+30
Failed Run Penalty Value	1E+30
Init Param Temperature	1
Max Failed Runs	-1
Max Num of Failed Designs	10
Max Num of Generated Designs	2000
Min Ratio of Accept Des for Reannealing	0.1
Num of Des for Conv Check	3
Num of Accept Des Before Reannealing	100
Num of Des Before Reannealing	1000
Penalty Base	0
Penalty Exponent	2
Penalty Multiplier	1000
Random seed value	-1
Reanneal Cost Function	TRUE
Reanneal Parameters	TRUE
Rel Rate of Cost Annealing	1
Rel Rate of Cost Quenching	1
Rel Rate of Param Annealing	1
Rel Rate of Param Quenching	1
Rel Grad Step for Reannealing	0.000001
Use fixed random seed	FALSE

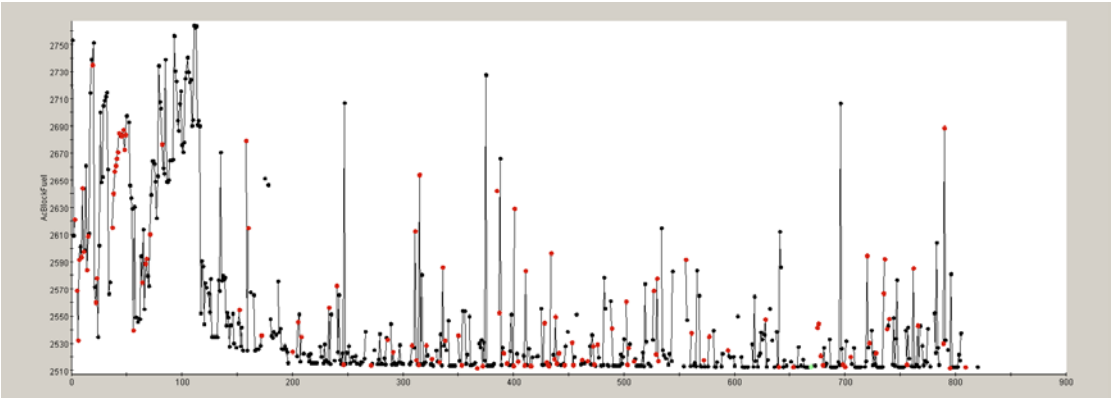
Τα αποτελέσματα της βελτιστοποίησης για μικρό και μεγάλο εύρος αναζήτησης φαίνονται στον Πίνακα 5.6.

Πίνακας 5.6

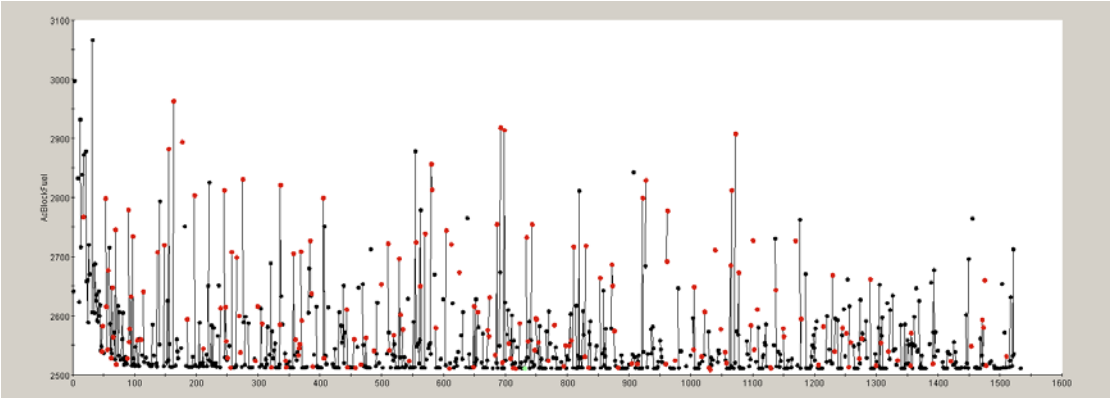
Optimum design point			
		ASA_R	ASA
	Run #	731	669
Μεταβλητές Σχεδιασμού	dFanBPPR	-4.74913077	-4.19774438
	dHPCPR	5.9615414	7.20293862
	dIPCPR	14.7943462	13.1070511
	dW1	-10.2067814	-11.2590043
	dW23	-15.6111873	-15.4602981
Objective	AcBlockFuel	2511.14052	2512.85734

Περιορισμοί	AcClimbTimeDes	20.5314311	20.4818196
	AcTODDes	1.93551934	1.93720275
	HpcLastStgBHeight	0.01047356	0.01062876
	T30TO	943.969824	944.25097
	T40Idle	1100.47439	1100.33774
	T41mTO	1139.96311	1140.17389
	TBO	23061.6479	23011.6119

Όπως φαίνεται από τα αποτελέσματα η βελτιστοποίηση με μειωμένο εύρος αναζήτησης οδηγεί σε πιο γρήγορα αποτελέσματα. Η βελτιστοποίηση με διευρυμένα όρια βρήκε ελάχιστα καλύτερη λύση κάτι που αποτελεί τυχαίο γεγονός αφού σε όλες τις υπόλοιπες βελτιστοποιήσεις που έγιναν, βρέθηκαν αντίθετα αποτελέσματα. Στα Σχήματα 5.12 και 5.13 φαίνεται η πορεία σύγκλισης του αλγόριθμου για τις δύο περιπτώσεις. Με κόκκινο παρουσιάζονται τα σημεία τα οποία δεν ικανοποιούν τους περιορισμούς και με πράσινο, η βέλτιστη λύση. Για μεγάλο εύρος ο αλγόριθμος συνέκλινε ύστερα από 1533 αξιολογήσεις ενώ για μειωμένο ύστερα από 820.



Σχήμα 5.12 History of objective function- με μειωμένα όρια (ASA)



Σχήμα 5.13 History of objective function- με διευρυμένα όρια (ASA_R)

5.3.2 Downhill Simplex

Ο επόμενος αλγόριθμος βελτιστοποίησης είναι ο Downhill Simplex. Στην βελτιστοποίηση με μεγάλο εύρος αναζήτησης ο αλγόριθμος, εγκλωβιζόταν εύκολα σε τοπικό ακρότατο. Μάλιστα οι λύσεις που έβρισκε δεν ήταν καθόλου ικανοποιητικές, οπότε δεν υπάρχει νόημα να παρουσιαστούν τα αποτελέσματα τους. Γι' αυτό τον λόγο θα παρουσιαστούν τα αποτελέσματα της βελτιστοποίησης με μειωμένο εύρος αναζήτησης. Στον Πίνακα 5.7 φαίνονται οι ρυθμίσεις του αλγόριθμου. Το μέγεθος του Simplex ορίστηκε ως το 40% του χώρου σχεδιασμού, ενώ ο μέγιστος αριθμός αξιολογήσεων είναι 400.

Πίνακας 5.7

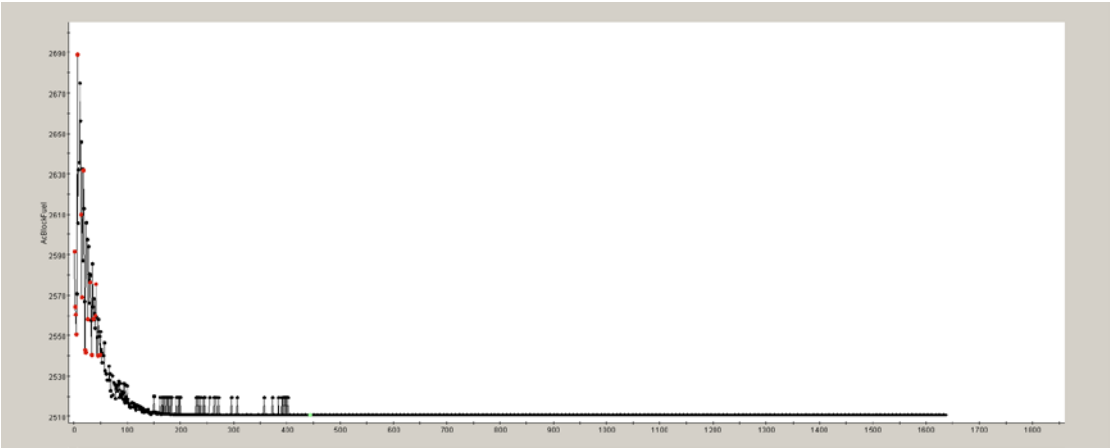
Optimization Technique:	Downhill Simplex
Failed Run Objective Value	1E+30
Failed Run Penalty Value	1E+30
Initial Simplex Size	0.4
Max Failed Runs	-1
Max Iterations	400

Τα αποτελέσματα της βελτιστοποίησης παρουσιάζονται στον Πίνακα 5.8.

Πίνακας 5.8

Optimum design point:		
Downhill Simplex		
	Run #	443
Μεταβλητές Σχεδιασμού	dFanBPPR	-5.3492589
	dHPCPR	2.85018301
	dIPCPR	17.17842484
	dW1	-9.320072174
	dW23	-16.0053215
Αντικειμενική	AcBlockFuel	2510.909973
Περιορισμοί	AcClimbTimeDes	20.62811684
	AcTODDes	1.936553264
	HpcLastStgBHeight	0.01039832
	T30TO	940.5972083
	T40Idle	1090.935508
	T41mTO	1137.434682
	TBO	23674.84088

Η πορεία σύγκλισης φαίνεται στο Σχήμα 5.14, ενώ ο αλγόριθμος συνέκλινε ύστερα από 1636 αξιολογήσεις.



Σχήμα 5.14 History of objective function -Downhill Simplex

5.3.3 Multi-Island Genetic Algorithm

Ο επόμενος αλγόριθμος είναι ο MIGA. Οι ρυθμίσεις του αλγόριθμου και για τις δύο βελτιστοποιήσεις φαίνονται στον Πίνακα 5.9.

Πίνακας 5.9

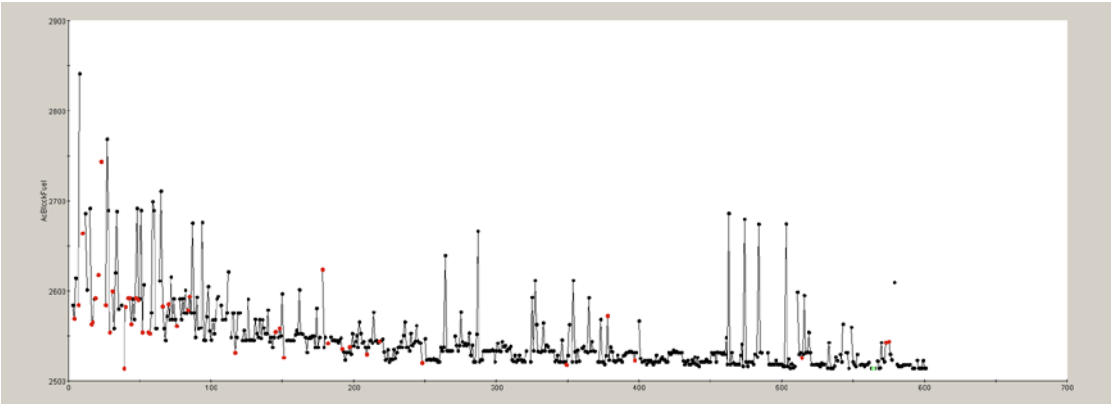
Optimization Technique:	Multi-Island GA
Elite Size	1
Failed Run Objective Value	1E+30
Failed Run Penalty Value	1E+30
Interval of Migration	3
Max Failed Runs	-1
Number of Generations	20
Number of Islands	3
Penalty Base	0
Penalty Exponent	2
Penalty Multiplier	1000
Random seed value	-1
Rate of Crossover	0.8
Rate of Migration	0.1
Rate of Mutation	0.01
Rel Tournament Size	0.35
Sub-Population Size	10
Use fixed random seed	FALSE

Η μόνη διαφορά με την πρώτη εφαρμογή είναι ο αριθμός των γενιών που τώρα είναι 20, μιας και το πρόβλημα είναι πιο περίπλοκο. Στον Πίνακα 5.10 φαίνονται τα αποτελέσματα των βελτιστοποιήσεων.

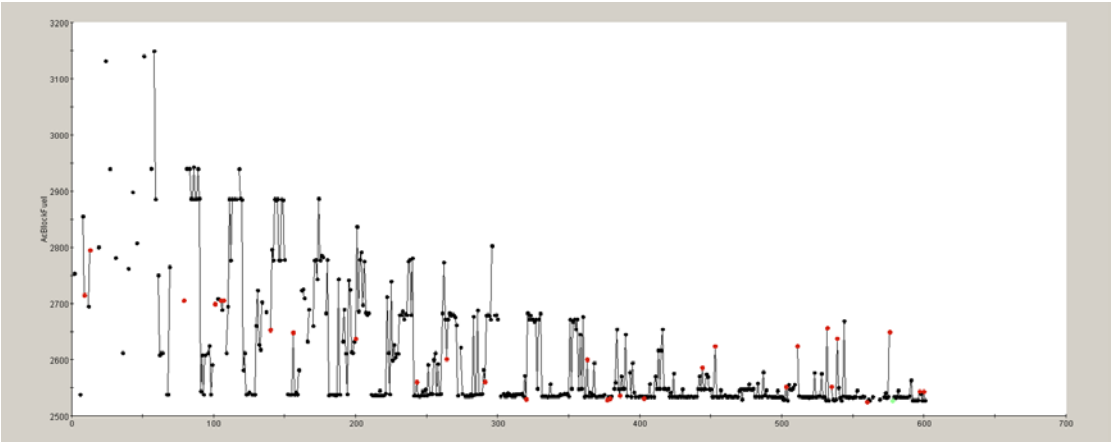
Πίνακας 5.10

Optimum design point			
		MIGA_R	MIGA
	Run #	578	564
Μεταβλητές Σχεδιασμού	dFanBPPR	-5.91165	-1.41329061
	dHPCPR	6.5424582	6.190585183
	dIPCPR	14.046845	10.39902342
	dW1	-8.123598	-17.1575494
	dW23	-15.74487	-14.8705272
Objective	AcBlockFuel	2514.0309	2527.027098
Περιορισμοί	AcClimbTimeDes	20.521498	20.25372646
	AcTODDes	1.9260348	1.952282554
	HpcLastStgBHeight	0.0103555	0.011589043
	T30TO	941.92458	941.7890375
	T40Idle	1082.3515	1082.926719
	T41mTO	1138.4298	1138.328148
	TBO	23426.594	23466.80713

Όπως φαίνεται η βελτιστοποίηση με μειωμένο εύρος έβγαλε καλύτερα και πιο γρήγορα αποτελέσματα. Στην συνέχεια παρουσιάζονται η πορεία σύγκλισης των βελτιστοποιήσεων. Ο αλγόριθμος και στις δύο περιπτώσεις συνέκλινε ύστερα από 601 αξιολογήσεις.



Σχήμα 5.15 History of objective function- με μειωμένα όρια (MIGA)



Σχήμα 5.16 History of objective function- με διευρυμένα όρια (MIGA_R)

5.3.4 Particle Swarm Optimizer

Ο τελευταίος αλγόριθμος είναι ο PSO, στον οποίο διατηρήθηκαν οι ίδιες τιμές των παραμέτρων του με εξαίρεση των αριθμό των σωματιδίων. Όπως αναφέρθηκε στην προηγούμενη εφαρμογή ο βέλτιστος αριθμός σωματιδίων, καθορίζεται από το πλήθος των μεταβλητών σχεδίασης. Επειδή σε αυτήν την εφαρμογή είναι 5, ο αριθμός των σωματιδίων τέθηκε ίσος με 25. Επίσης, η Vmax ορίστηκε 250 καθώς η απόκλιση της βέλτιστης λύσης από το σημείο αναφοράς είναι περίπου 200 μονάδες. Στον Πίνακα 5.11 φαίνονται οι τιμές των παραμέτρων του αλγόριθμου.

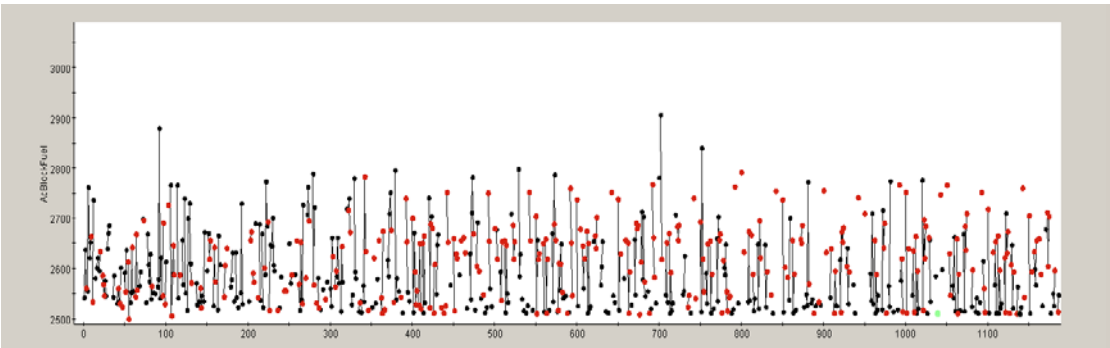
Πίνακας 5.11

Optimization Technique:	Multi-Objective Particle Swarm
Failed Run Objective Value	1E+30
Failed Run Penalty Value	1E+30
Global Increment	1.655
Inertia	0.721
Max Failed Runs	-1
Maximum Iterations	50
Maximum Velocity	250
Number of Particles	25
Particle Increment	1.655
Random seed value	-1
Use fixed random seed	FALSE

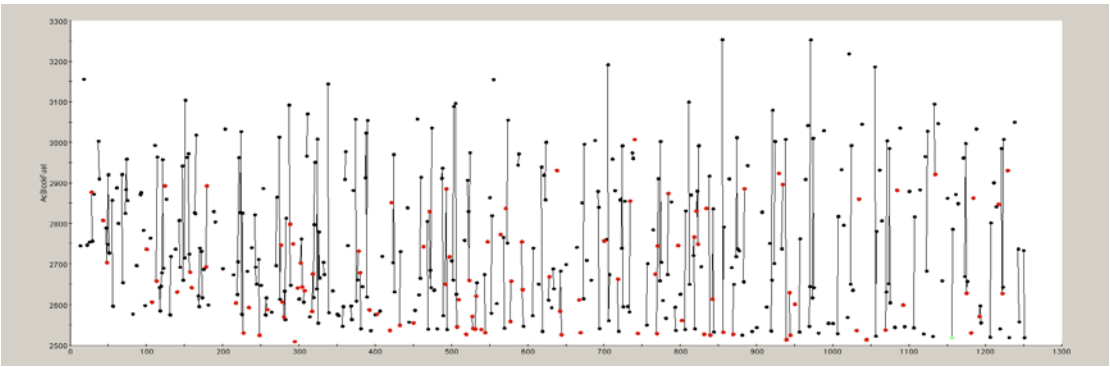
Τα αποτελέσματα των βελτιστοποιήσεων φαίνονται στον Πίνακα 5.12. Στη συνέχεια φαίνονται η πορεία σύγκλισης του αλγόριθμου για τις δύο περιπτώσεις.

Πίνακας 5.12

Optimum design point			
		PSO_R	PSO
	Run #	1156	1065
Μεταβλητές Σχεδιασμού	dFanBPPR	-4.49787378	-5.03396078
	dHPCPR	8.998790728	4.98570328
	dIPCPR	11.53758923	16.2606201
	dW1	-9.146435281	-9.72816236
	dW23	-14.92003628	-15.6963102
Objective	AcBlockFuel	2518.685903	2509.6353
Περιορισμοί	AcClimbTimeDes	20.14472406	20.5780118
	AcTODDes	1.921702955	1.93561267
	HpcLastStgBHeight	0.010532899	0.01037479
	T30TO	943.5429556	944.304883
	T40Idle	1133.119003	1100.63314
	T41mTO	1139.643081	1140.21431
	TBO	23132.55724	23002.1788



Σχήμα 5.17 History of objective function- με μειωμένα όρια (PSO)



Σχήμα 5.18 History of objective function- με διευρυμένα όρια (PSO_R)

5.4 EASY

Το εργαλείο βελτιστοποίησης EASY (Evolutionary Algorithms SYstem) [40] αναπτύχθηκε από τη Μονάδα Παράλληλης Υπολογιστικής Ρευστομηχανικής και Βελτιστοποίησης του ΕΘΣ [16] και υλοποιεί στοχαστικές και υβριδικές μεθόδους βελτιστοποίησης. Έχει χρησιμοποιηθεί και στο παρελθόν για την βελτιστοποίηση συνιστωσών του αεροπορικού κινητήρα [42]. Κύριο χαρακτηριστικό του είναι οι χρησιμοποίηση Εξελικτικών Αλγόριθμων και είναι κατάλληλο για μονοκριτηριακά και πολυκριτηριακά προβλήματα, με ή χωρίς περιορισμούς. Για την εξοικονόμηση χρόνου και του υπολογιστικού κόστους ο EASY διαθέτει διάφορα εργαλεία που μπορούν να χρησιμοποιηθούν μεμονωμένα ή συνδυαστικά μεταξύ τους.

Ένα σημαντικό εργαλείο του είναι ο χωρισμός της διαδικασίας της βελτιστοποίησης σε επίπεδα (multilevel evaluation) [43]. Ο EASY χρησιμοποιεί, στο πρώτο επίπεδο, λογισμικό με μικρό κόστος για να περιορίσει τον χώρο εξερεύνησης των λύσεων. Στην συνέχεια, χρησιμοποιείται ακριβές λογισμικό αξιολόγησης, διαφορετικό από το πρώτο, βελτιώνοντας ουσιαστικά τα αποτελέσματα του προηγούμενου επιπέδου. Συνηθίζεται η χρήση Εξελικτικών Αλγόριθμων για τον περιορισμό του χώρου αναζήτησης των λύσεων, στο πρώτο επίπεδο και αιτιοκρατικών μεθόδων στο υψηλότερο επίπεδο.

Επιπλέον παρέχει τη δυνατότητα για παράλληλη επεξεργασία [44] , που έχει ως αποτέλεσμα την μείωση του χρόνου υπολογισμού ιδιαίτερα κατά την εκτέλεση Εξελικτικών Αλγόριθμων. Στην συγκεκριμένη εφαρμογή δεν χρησιμοποιήθηκε αυτή η δυνατότητα γιατί το πρόβλημα δεν έχει πολλές μεταβλητές σχεδιασμού και ο υπολογιστικός χρόνος είναι σχετικά μικρός. Επίσης μπορεί να χρησιμοποιήσει κατανεμημένους εξελικτικούς αλγόριθμος (Distributed Evolutionary Algorithms, DEA), όπου τα άτομα χωρίζονται σε υποομάδες πληθυσμών σε κάθε γενιά, οι οποίοι επικοινωνούν μεταξύ τους [45]. Δίνεται έτσι η δυνατότητα κάθε υποπληθυσμός να έχει διαφορετικές παραμέτρους που αφορούν εξελικτικούς τελεστές. Αυξάνεται έτσι, η ποικιλομορφία των λύσεων με την πάροδο των γενεών.

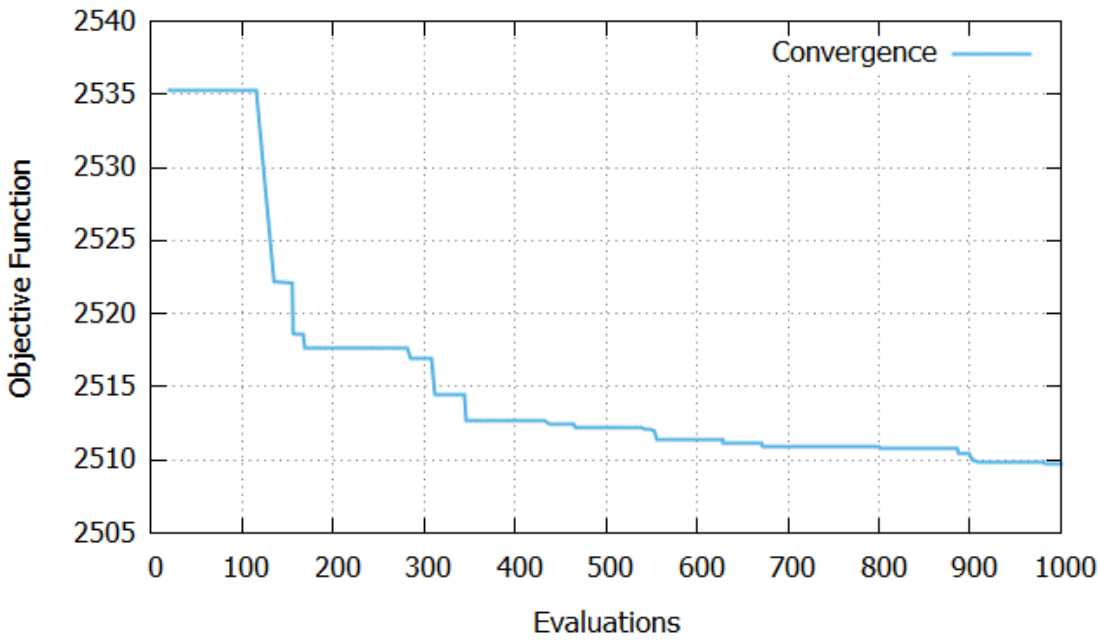
Τέλος δίνεται η δυνατότητα χρησιμοποίησης μεταπροτύπων για την αντιμετώπιση υψηλού κόστους επίλυσης προβλημάτων. Χρησιμοποιούνται εξελικτικοί αλγόριθμοι υποβοηθούμενοι με μεταπρότυπα (Metamodel-assisted EAs, MAEAs) [46]–[48]. Η διαφορά του EASY με τα μεταπρότυπα που χρησιμοποιεί το Isight είναι πως, ο EASY, χρησιμοποιεί μεταπρότυπα που εκπαιδεύονται online σε αντίθεση με το Isight που χρησιμοποιεί offline μεταπρότυπα. Στα offline μεταπρότυπα χρησιμοποιείται μία τεχνική δειγματοληψίας όπως η Latin Hypercube, ύστερα χρησιμοποιείται μια τεχνική εκπαίδευσης του μεταπροτύπου όπως τα RBF και στην συνέχεια με την χρήση ενός αλγόριθμου βελτιστοποίησης όπως SQP, βρίσκονται οι βέλτιστες λύσεις. Ανάλογα με το μέγεθος του σφάλματος το μεταπρότυπο ενημερώνεται και επαναλαμβάνεται η διαδικασία. Στα online μεταπρότυπα που χρησιμοποιούνται στον EASY [49], πρώτα αξιολογούνται τα άτομα των πρώτων γενεών χρησιμοποιώντας ακριβό λογισμικό

αξιολόγησης και στη συνέχεια χρησιμοποιούνται για την εκπαίδευση τοπικών μεταπροτύπων. Πιο συγκεκριμένα, κάθε νέο άτομο της επόμενης γενιάς αξιολογείται χρησιμοποιώντας ένα μεταπρότυπο εκπαιδευμένο με βάση τα γειτονικά άτομα, με την χρήση ενός κριτηρίου απόστασης. Σε αντίθεση με τα offline μεταπρότυπα τα online επανεκπαιδεύονται παράλληλα με την εξέλιξη. Ο EASY χρησιμοποιεί RBF δίκτυα ως μεταπρότυπα και οι MAEAs βασίζονται στην χαμηλού κόστους προαξιολόγηση [49], [50] με βάση την οποία ο χώρος των λύσεων εξερευνάται από τα μεταπρότυπα, εντοπίζονται οι καλύτερες λύσεις και μόνο αυτές αξιολογούνται από το ακριβές λογισμικό. Με αυτόν τον τρόπο, εξοικονομείται ο υπολογιστικός χρόνος που θα απαιτούνταν για την ακριβή αξιολόγηση όχι και τόσο σημαντικών λύσεων. Ωστόσο, στην συγκεκριμένη εφαρμογή δεν χρησιμοποιήθηκαν μεταπρότυπα καθώς το πρόβλημα δεν ήταν τόσο χρονοβόρο.

Στον Πίνακα 5.13 φαίνεται το αποτέλεσμα της βελτιστοποίησης με τον EASY ενώ στο σχήμα 5.19 φαίνεται η πορεία σύγκλισης του αλγόριθμου. Η βελτιστοποίηση έγινε μόνο με διευρυμένα όρια αναζήτησης.

Πίνακας 5.13

Optimum design point		
EASY		
	Run #	902
Μεταβλητές Σχεδιασμού	dFanBPPR	-5.142818917
	dHPCPR	1.558988522
	dIPCPR	19.99730968
	dW1	-9.938808124
	dW23	-15.64037639
Objective	AcBlockFuel	2509.768726
Περιορισμοί	AcClimbTimeDes	20.54511876
	AcTODDes	1.936102602
	HpcLastStgBHeight	0.010359939
	T30TO	943.966924
	T40Idle	1089.237164
	T41mTO	1139.960931
	TBO	23066.57097



Σχήμα 5.19 History of objective function-EASY

5.5 Χρήση των Kriging και RBF μεταπροτύπων

Όπως και στην προηγούμενη εφαρμογή δημιουργήθηκαν μεταπρότυπα Kriging και RBF για τον εντοπισμό της βέλτιστης λύσης. Ωστόσο, θέτοντας μεγάλο εύρος αναζήτησης στις μεταβλητές σχεδιασμού, δημιουργούνται μεταπρότυπα με τεράστιο σφάλμα. Για την μείωση του σφάλματος σε αποδεκτά όρια, περιορίστηκε το εύρος αναζήτησης στο [-2.5%,+2.5%] από την βέλτιστη λύση των προηγούμενων βελτιστοποιήσεων. Ουσιαστικά, η χρήση μεταπροτύπων στην περίπτωση αυτή θα επιβεβαιώσει ή όχι την βέλτιστη λύση που προέκυψε από την εφαρμογή των αλγόριθμων βελτιστοποίησης.

Kriging μεταπρότυπα

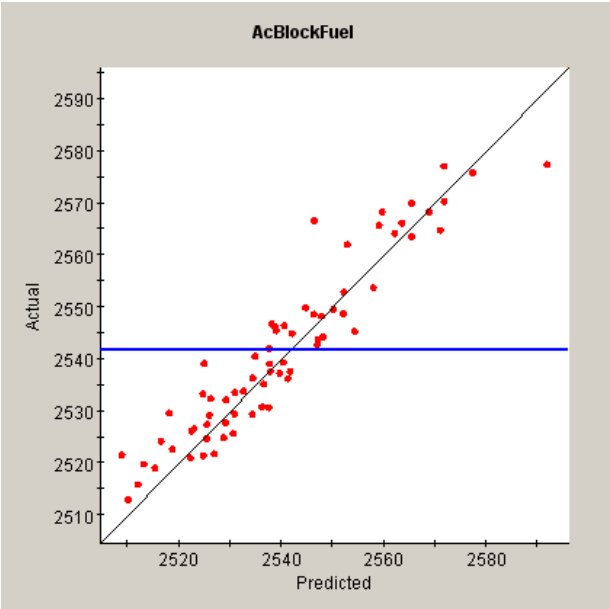
Για την δημιουργία του Kriging χρησιμοποιήθηκε, ως συνάρτηση συσχέτισης η Mattern Cubic καθώς αυτή μόνο έδινε ικανοποιητικά αποτελέσματα, ενώ το κριτήριο απόστασης του φίλτρου τέθηκε ίσο με 0.1, έτσι ώστε να γίνει καλύτερη εφαρμογή. Ο αριθμός των σημείων δειγματοληψίας, επιλέχθηκε με βάση την [33] όπως και στην προηγούμενη εφαρμογή. Το πρόβλημα δεν είναι ούτε μικρού μεγέθους αλλά ούτε και μεγάλου. Το Isight προτείνει 62 σημεία, ωστόσο για να επιτευχθεί αυτός ο αριθμός χρειάζονται περισσότερα σημεία καθώς κάποια σημεία προκύπτουν infeasible. Επομένως ο αριθμός των σημείων που επιλέχθηκε ήταν 100 από τα οποία προέκυψαν τελικά 69. Η τεχνική δειγματοληψίας που επιλέχθηκε είναι η Optimal Latin Hypercube, ενώ για την ανάλυση σφάλματος χρησιμοποιήθηκαν όλα τα σημεία και η μέθοδος leave-one-out cross-validation.

Τα αποτελέσματα της ανάλυσης σφάλματος παρουσιάζονται στον Πίνακα 5.14.

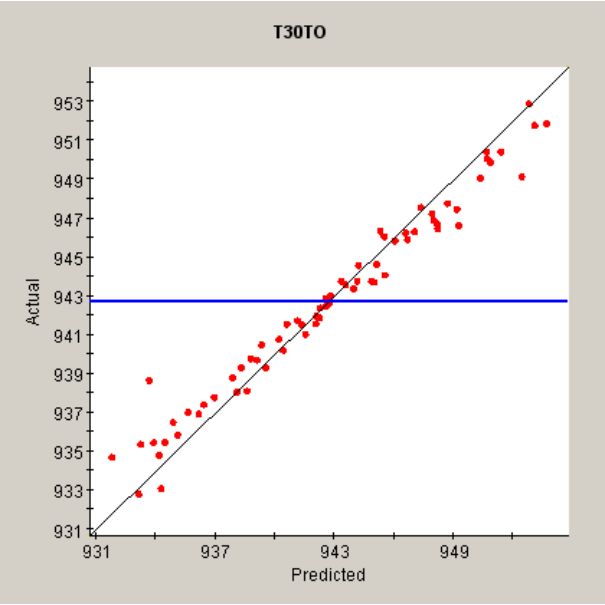
Πίνακας 5.14

Kriging	Error Type			
	Average	Maximum	RMS	R Squared
AcBlockFuel	0.07261	0.31409	0.09285	0.87001
AcClimbTimeDes	0.22571	0.51788	0.26389	0
AcTODDes	0.06899	0.24708	0.08674	0.85508
HpcLastStgBHeight	0.04818	0.24843	0.06368	0.93275
T30TO	0.04487	0.24882	0.06111	0.93641
T40Idle	0.05413	0.48243	0.08419	0.77903
T41mTO	0.044817	0.24881	0.06111	0.93642
TBO	0.04499	0.25227	0.06141	0.93603

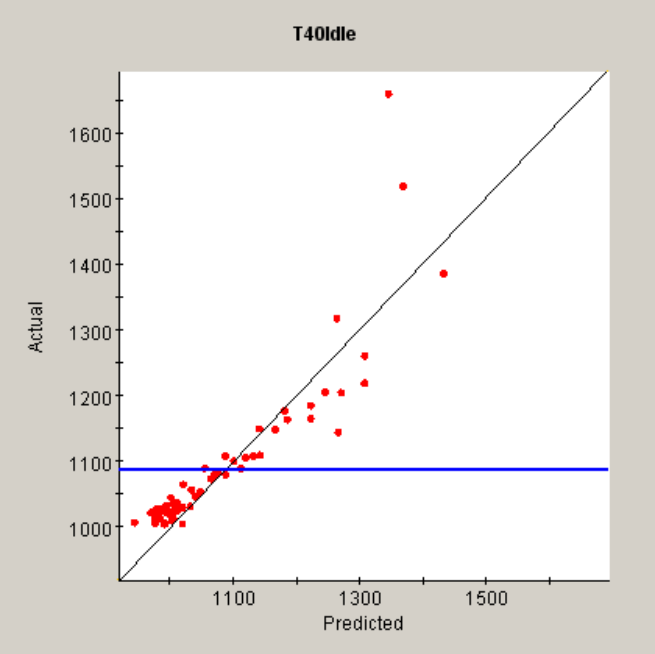
Όπως παρατηρείται ακόμη και με μειωμένο εύρος αναζήτησης υπάρχει μεγάλο σφάλμα στο μεταπρότυπο, ειδικά στην παράμετρο AcClimbTimeDes. Αυτό οφείλεται στην πολυπλοκότητα του προβλήματος καθώς επίσης και στο ότι κάποια σημεία δειγματοληψίας προκύπτουν infeasible, με αποτέλεσμα τα προκύπτοντα σημεία δειγματοληψίας να μην είναι ομοιόμορφα κατανεμημένα. Στη συνέχεια παρατίθενται τα σχήματα που δείχνουν την εφαρμογή των αποκρίσεων των εξόδων. Χαρακτηριστικά θα παρουσιαστούν τα σχήματα της εφαρμογής της απόκρισης της αντικειμενικής συνάρτησης και τριών περιορισμών εκ των οποίων, οι δύο που είχαν την χειρότερη εφαρμογή.



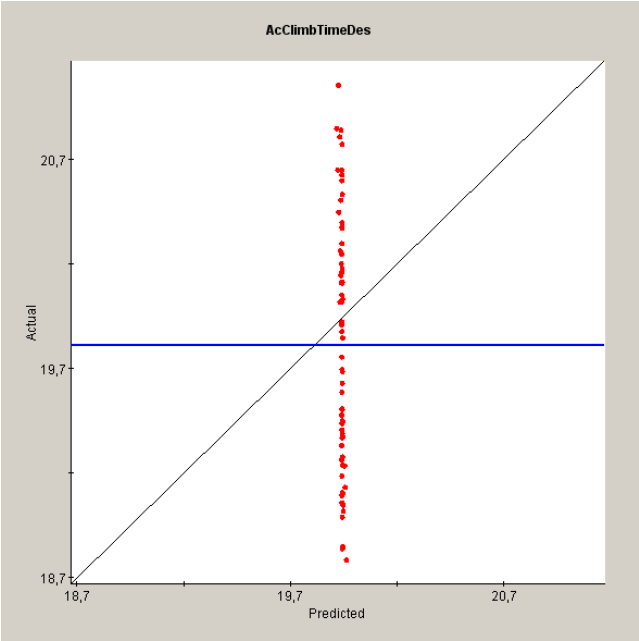
Σχήμα 5.20 Response fit of AcBlockFuel



Σχήμα 5.21 Response fit of T30TO

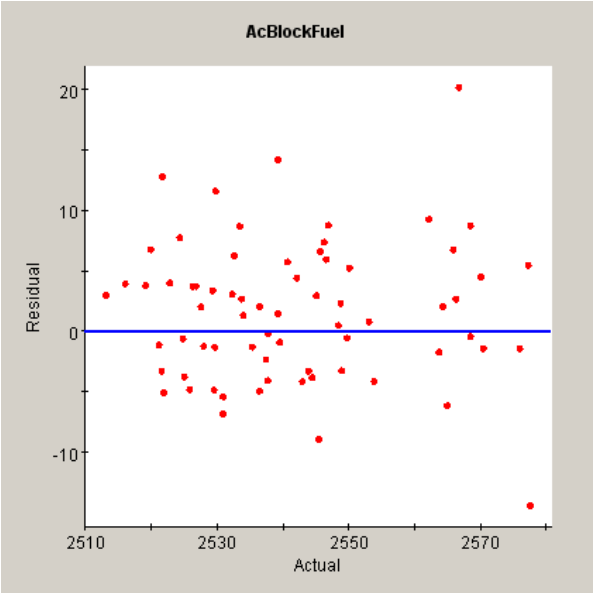


Σχήμα 5.22 Response fit of T40Idle

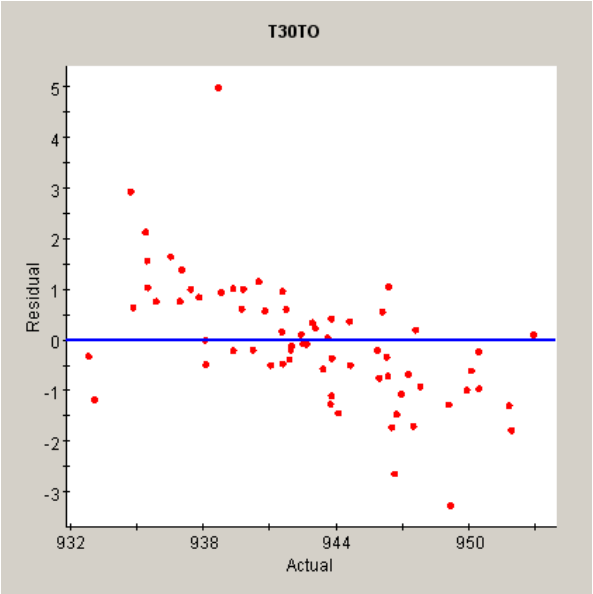


Σχήμα 5.23 Response fit of AcClimbTimeDes

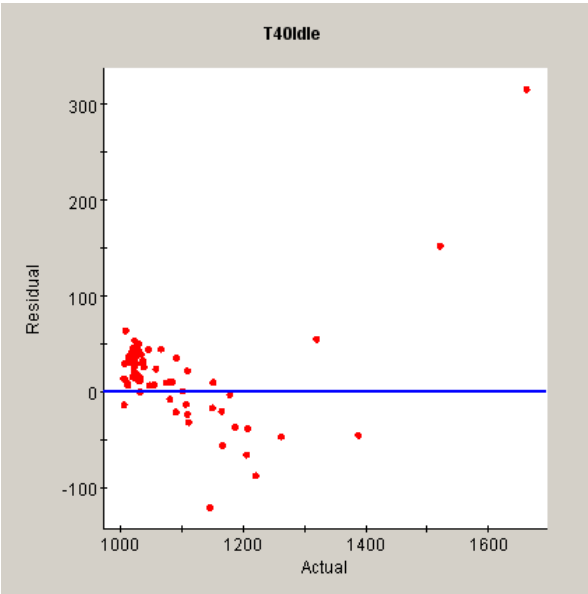
Όπως παρατηρείται στο Σχήμα 5.22 και 5.23 δεν υπάρχει σχεδόν καθόλου εφαρμογή, ακόμη και με το μειωμένο εύρος αναζήτησης. Τα αποτελέσματα του Πίνακα 5.12 επιβεβαιώνονται και από τα σχήματα που δείχνουν την εφαρμογή των αποκρίσεων. Στη συνέχεια παρουσιάζονται τα υπόλοιπα των εξόδων. Τα σχήματα αφορούν πάλι την αντικειμενική συνάρτηση και τους τρεις περιορισμούς.



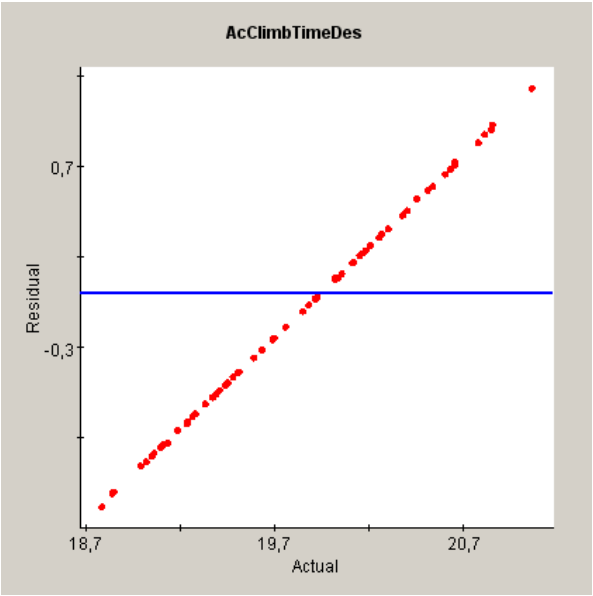
Σχήμα 5.24 Residual graph of AcBlockFuel



Σχήμα 5.25 Residual graph of T30TO



Σχήμα 5.26 Residual graph of T40Idle



Σχήμα 5.27 Residual graph of AcClimbTimeDes

Τα σχήματα των υπολοίπων πάλι επιβεβαιώνουν το σφάλμα που υπάρχει στις δύο παραμέτρους την AcClimbTimeDes και στην T40Idle.

Ο Πίνακας 5.15 δείχνει τα αποτελέσματα της βελτιστοποίησης που έγινε με την χρήση του αλγόριθμου NLPQL. Οι τιμές των παραμέτρων του αλγόριθμου είναι οι ίδιες με την προηγούμενη εφαρμογή.

Πίνακας 5.15

Kriging			Πραγματικό	Απόκλιση %
Αντικειμενική	AcBlockFuel	2502.668956	2514.33327	-0.464
Μεταβλητές Σχεδιασμού	dFanBPPR	-5.876270391	-5.876270391	
	dHPCPR	5.5443005	5.5443005	
	dIPCPR	15.77696336	15.77696336	
	dW1	-8.860225615	-8.860225615	
	dW23	-15.5732225	-15.5732225	
Περιορισμοί	AcClimbTimeDes	21.01722663	20.51511836	2.448
	AcTODDes	1.937054397	1.926971703	0.523
	HpcLastStgBHeight	0.01020054	0.010374053	-1.673
	T30TO	944.2762748	943.8943221	0.040
	T40Idle	1020.316278	1066.857349	-4.362
	T41mTO	1140.192757	1139.906489	0.025
	TBO	22999.99688	23074.08826	-0.321

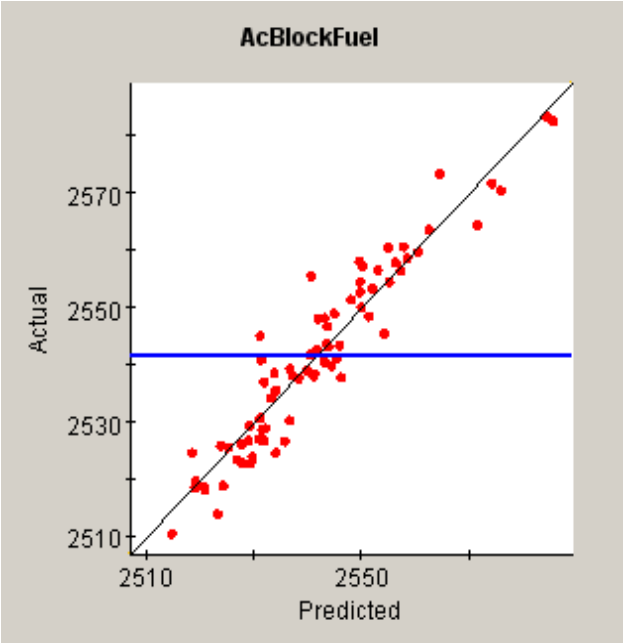
RBF μεταπρότυπα

Στον Πίνακα 5.16 φαίνονται τα αποτελέσματα από την ανάλυση του σφάλματος στην περίπτωση των μεταπροτύπων RBF.

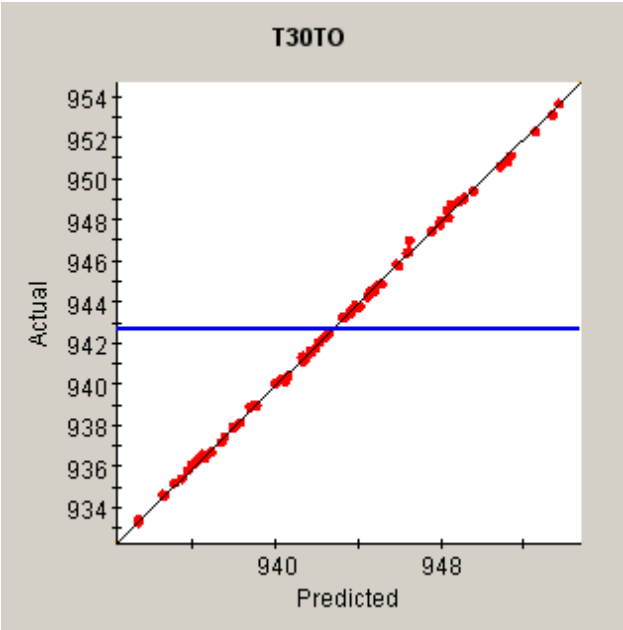
Πίνακας 5.16

RBF	Error Type			
	Average	Maximum	RMS	R Squared
AcBlockFuel	0.0522	0.20617	0.06801	0.90791
AcClimbTimeDes	0.02287	0.07835	0.02792	0.98794
AcTODDes	0.03092	0.11005	0.0376	0.97557
HpcLastStgBHeight	0.00582	0.02257	0.00759	0.99894
T30TO	0.00532	0.02873	0.00702	0.99919
T40Idle	0.0304	0.25481	0.04563	0.94599
T41mTO	0.00532	0.02872	0.00702	0.99919
TBO	0.0053	0.0291	0.00698	0.9992

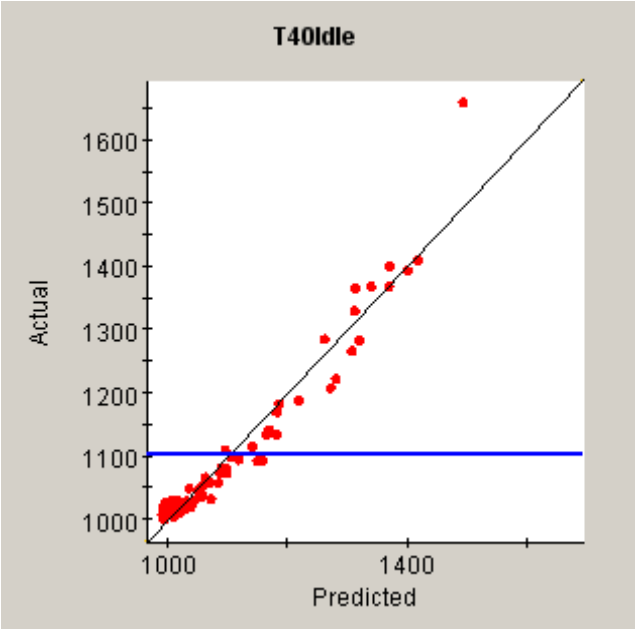
Τα αποτελέσματα από την ανάλυση είναι σαφώς καλύτερα σε σχέση με το Kriging. Τα Σχήματα 5.28-5.35 παρουσιάζουν τις αντίστοιχες αποκρίσεις των εξόδων και τα σφάλματά τους.



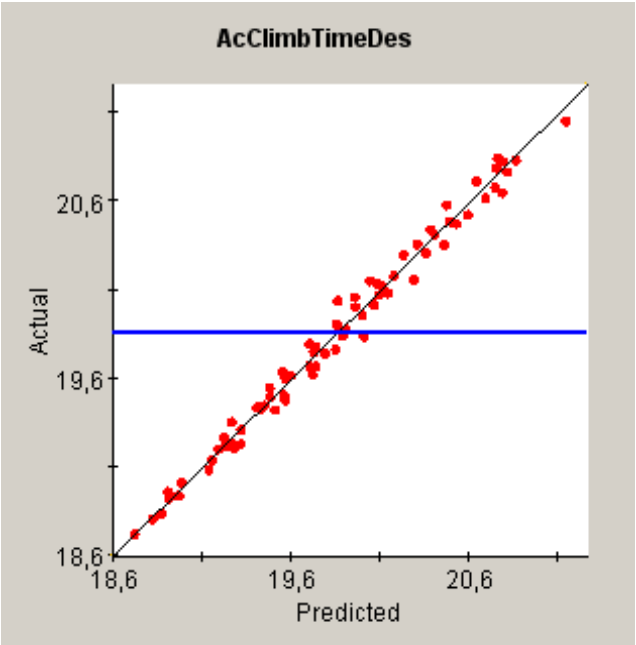
Σχήμα 5.28 Response fit of AcBlockFuel



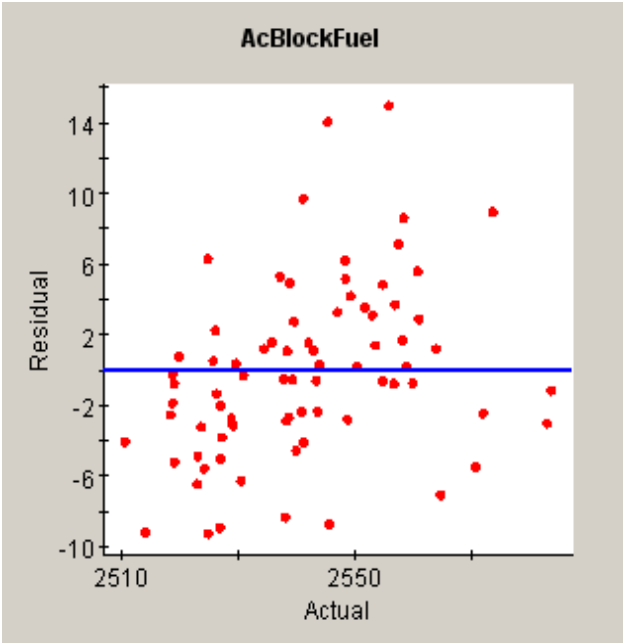
Σχήμα 5.29 Response fit of T30TO



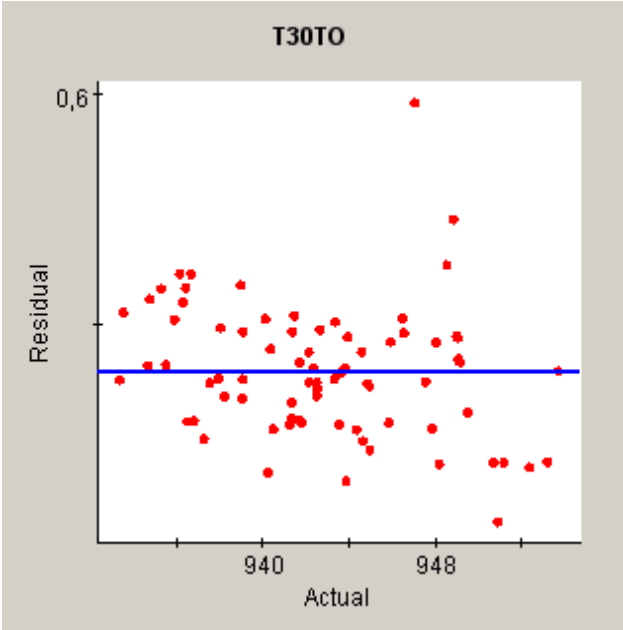
Σχήμα 5.30 Response fit of T40Idle



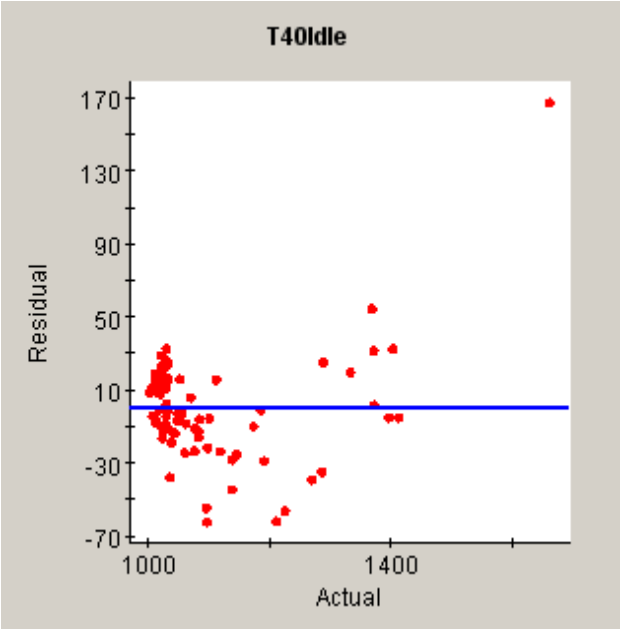
Σχήμα 5.31 Response fit of AcClimdTimeDes



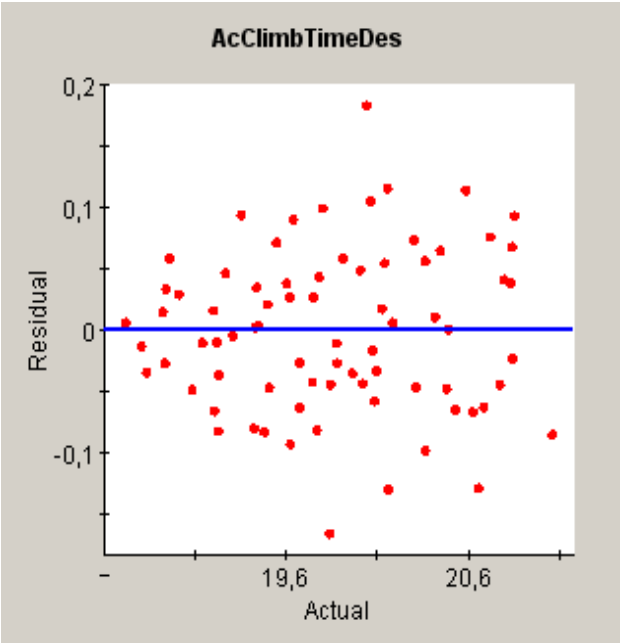
Σχήμα 5.32 Residual graph of AcBlockFuel



Σχήμα 5.33 Residual graph of T30TO



Σχήμα 5.34 Residual graph of T40Idle



Σχήμα 5.35 Residual graph of AcClimbTimeDes

Στον Πίνακα 5.17 παρουσιάζονται τα αποτελέσματα της βελτιστοποίησης με την χρήση του αλγόριθμου NLPQL.

Πίνακας 5.17

RBF			Πραγματικό	Απόκλιση %
Αντικειμενική	AcBlockFuel	2507.957096	2509.6353	-0.067
Μεταβλητές Σχεδιασμού	dFanBPPR	-5.66781515	-5.6678152	
	dHPCPR	6.166251018	6.16625102	
	dIPCPR	15.35439762	15.3543976	
	dW1	-9.252119605	-9.2521196	
	dW23	-16.444428	-16.444428	
Περιορισμοί	AcClimbTimeDes	21.04260586	20.5780118	2.258
	AcTODDes	1.94686583	1.93561267	0.581
	HpcLastStgBHeight	0.010243911	0.01023826	0.055
	T30TO	944.3177367	944.304883	0.001
	T40Idle	1101.88131	1100.63314	0.113
	T41mTO	1140.22396	1140.21431	0.001
	TBO	22999.99879	22998.3658	0.007

5.6 Σύγκριση και Ανάλυση των αποτελεσμάτων

Στον Πίνακα 5.18 παρουσιάζονται τα συγκεντρωτικά αποτελέσματα, από όλους τους αλγόριθμους βελτιστοποίησης, ενώ στον Πίνακα 5.19 φαίνονται οι επί τοις εκατό αποκλίσεις τους από τις τιμές αναφοράς. Τα αποτελέσματα αυτά, είναι τα καλύτερα κάθε τεχνικής βελτιστοποίησης από το πλήθος των εκτελέσεων που πραγματοποιήθηκαν.

Πίνακας 5.18

Optimum design point									
		PSO_R	PSO	ASA	ASA_R	MIGA	MIGA_R	Simplex	EASY
	Run #	1156	1065	420	828	564	578	443	902
Μεταβλητές Σχεδιασμού	dFanBPPR	-4.498	-5.034	-6.157	-6.406	-5.912	-1.413	-5.349	-5.142
	dHPCPR	8.999	4.986	5.566	17.810	6.542	6.191	2.850	1.558
	dIPCPR	11.538	16.261	15.977	4.192	14.047	10.399	17.178	19.997
	dW1	-9.146	-9.728	-8.516	-6.023	-8.124	-17.158	-9.320	-9.938
	dW23	-14.920	-15.696	-15.986	-15.278	-15.745	-14.871	-16.005	-15.640
Objective	AcBlockFuel	2518.686	2509.635	2510.819	2517.838	2514.031	2527.027	2510.910	2509.768
Περιορισμοί	AcClimbTimeDes	20.145	20.578	20.730	20.278	20.521	20.254	20.628	20.545
	AcTODDes	1.922	1.936	1.932	1.906	1.926	1.952	1.937	1.936
	HpcLastStgBHeight	0.011	0.010	0.010	0.010	0.010	0.012	0.010	0.010
	T30TO	943.543	944.305	943.850	943.559	941.925	941.789	940.597	943.966
	T40Idle	1133.119	1100.633	1069.532	1075.442	1082.351	1082.927	1090.936	1089.237
	T41mTO	1139.643	1140.214	1139.873	1139.655	1138.430	1138.328	1137.435	1139.961
	TBO	23132.557	23002.179	23081.397	23114.436	23426.594	23466.807	23674.841	23066.571

Πίνακας 5.19

Αποκλίσεις %		PSO_R	PSO	ASA	ASA_R	MIGA	MIGA_R	Simplex	EASY
Objective	AcBlockFuel	-7.274	-7.607	-7.564	-7.305	-7.445	-6.967	-7.56	-7.602
Περιορισμοί	AcClimbTimeDes	41.862	44.913	45.986	42.799	44.515	42.63	45.266	44.682
	AcTODDes	12.82	13.637	13.432	11.872	13.075	14.616	13.692	13.666
	HpcLastStgBHeight	-25.589	-26.706	-27.431	-26.643	-26.842	-18.128	-26.54	-26.811
	T30TO	5.065	5.15	5.099	5.067	4.885	4.87	4.737	5.112
	T40Idle	0.823	-2.067	-4.835	-4.309	-3.694	-3.643	-2.93	-3.081
	T41mTO	3.083	3.134	3.104	3.084	2.973	2.964	2.883	3.112
	TBO	-29.189	-29.588	-29.345	-29.244	-28.288	-28.165	-27.529	-29.391

Από τα αποτελέσματα των βελτιστοποιήσεων ο PSO και ο EASY βρήκαν την καλύτερη λύση, ωστόσο και οι υπόλοιποι αλγόριθμοι προσέγγισαν αρκετά την βέλτιστη λύση. Επίσης ο αλγόριθμος Downhill Simplex είναι ιδιαίτερα γρήγορος και σχετικά ακριβής. Στην πλειοψηφία των εκτελέσεων όμως, ο αλγόριθμος απέτυχε να βρει κάποια

λύση κοντά στο -7% από την τιμή αναφοράς. Η εύρεση της βέλτιστης λύσης από τον αλγόριθμο είναι σπάνια και οφείλεται στην τυχαιότητα της θέσης εκκίνησης του αλγόριθμου. Οι αλγόριθμοι PSO, ASA και MIGA έβγαζαν συνεχώς καλές λύσεις, γι' αυτόν το λόγο προτείνεται η χρήση τους για τέτοιου είδους προβλήματα. Αυτό που φαίνεται ξεκάθαρα είναι η εύρεση της βέλτιστης λύσης, όταν υπάρχει μειωμένο εύρος αναζήτησης. Αυτό μάλιστα δείχνει την ανωτερότητα του EASY καθώς μπορεί να περιορίσει το εύρος αναζήτησης αυτόματα. Στο σημείο αυτό πρέπει να τονίσουμε ότι τα συγκεκριμένα συμπεράσματα αφορούν την συγκεκριμένη εφαρμογή με τις συγκεκριμένες ρυθμίσεις κάθε αλγόριθμου.

Σύγκριση μεταπροτύπων

Στον Πίνακα 5.20 φαίνονται συγκριτικά η ανάλυση σφάλματος μεταξύ των μεταπροτύπων RBF και Kriging.

Πίνακας 5.20

Error Analysis	Error Type							
	Kriging				RBF			
	Average	Maximum	RMS	R^2	Average	Maximum	RMS	R^2
AcBlockFuel	0.07261	0.31409	0.09285	0.87001	0.0522	0.20617	0.06801	0.90791
AcClimbTimeDes	0.22571	0.51788	0.26389	0	0.02287	0.07835	0.02792	0.98794
AcTODDes	0.06899	0.24708	0.08674	0.85508	0.03092	0.11005	0.0376	0.97557
HpcLastStgBHeight	0.04818	0.24843	0.06368	0.93275	0.00582	0.02257	0.00759	0.99894
T30TO	0.04487	0.24882	0.06111	0.93641	0.00532	0.02873	0.00702	0.99919
T40Idle	0.05413	0.48243	0.08419	0.77903	0.0304	0.25481	0.04563	0.94599
T41mTO	0.044817	0.24881	0.06111	0.93642	0.00532	0.02872	0.00702	0.99919
TBO	0.04499	0.25227	0.06141	0.93603	0.0053	0.0291	0.00698	0.9992

Το RBF έχει μικρότερο μέσο, μέγιστο και RMS σφάλμα σε όλες τις αποκρίσεις ενώ μάλιστα το R^2 είναι ικανοποιητικό αναλόγως της φύσης του προβλήματος. Μάλιστα το Kriging έχει μεγάλη ανακρίβεια στις παραμέτρους AcClimbTimeDes και T40Idle κάτι το οποίο φαίνεται και από την απόκλισή τους κατά την βελτιστοποίηση.

Στον Πίνακα 5.21 παρουσιάζονται συγκριτικά οι αποκλίσεις των βελτιστοποιήσεων με χρήση των δύο μεταπροτύπων.

Πίνακας 5.21

Αποκλίσεις % από το πραγματικό		RBF	Kriging
Αντικειμενική	AcBlockFuel	-0.066870318	-0.463912824
Περιορισμοί	AcClimbTimeDes	2.257720843	2.447503659
	AcTODDes	0.581374757	0.523240408
	HpcLastStgBHeight	0.055200597	-1.672571196
	T30TO	0.001361164	0.040465615
	T40Idle	0.11340502	-4.362445563
	T41mTO	0.000846439	0.025113207
	TBO	0.007100314	-0.321102093

Και από τους δύο πίνακες παρατηρούμε ότι το RBF υπερτερεί έναντι του Kriging στην συγκεκριμένη εφαρμογή.

Η καλύτερη λύση που βρέθηκε κατά την βελτιστοποίηση του προβλήματος προέκυψε από τον αλγόριθμο PSO, μετά την εκτέλεση με μειωμένο εύρος αναζήτησης. Η βέλτιστη λύση και οι ποσοστιαίες διαφορές των παραμέτρων από την τιμή αναφοράς φαίνονται στον Πίνακα 5.22.

Πίνακας 5.22

Βέλτιστη Λύση				
		PSO	Τιμές Αναφοράς	Διαφορά επί τοις %
Μεταβλητές Σχεδιασμού	dFanBPPR	-5.034	0	
	dHPCPR	4.986	0	
	dIPCPR	16.261	0	
	dW1	-9.728	0	
	dW23	-15.696	0	
Objective	AcBlockFuel	2509.635	2716.267	-7.607
Περιορισμοί	AcClimbTimeDes	20.578	14.200	44.913
	AcTODDes	1.936	1.703	13.637
	HpcLastStgBHeight	0.0104	0.01415	-26.706
	T30TO	944.305	898.057	5.150
	T40Idle	1100.633	1123.868	-2.067
	T41mTO	1140.214	1105.561	3.134
	TBO	23002.179	32667.825	-29.588

Όπως παρατηρείται έχουμε βελτίωση περίπου 7.6% από την τιμή αναφοράς και περίπου 0.3 από την [17]. Φαίνεται ότι αυξάνεται ο χρόνος που χρειάζεται για να φτάσει το αεροσκάφος στο επιθυμητό ύψος καθώς το AcClimbTimeDes αυξάνεται κατά περίπου 45%, όπως και το μήκος του διαδρόμου που χρειάζεται για απογείωση το αεροσκάφος, αφού το AcTODDes αυξάνεται κατά 13.6 %. Αυτό υποδηλώνει

μικρότερη ώση που προκύπτει από μικρότερη μηχανή. Κάτι που συμβαίνει σε αυτήν την περίπτωση αφού οι μεταβλητές FANBPPR και W1 καθορίζουν το μέγεθος του ανεμιστήρα, ενώ η W23 το μέγεθος του πυρήνα. Εφόσον οι μεταβλητές αυτές μειώνονται, μειώνεται και το μέγεθος της μηχανής. Ωστόσο οι δύο περιορισμοί δεν φτάνουν στα όριά τους. Αυτό οφείλεται γιατί δύο άλλοι περιορισμοί αγγίζουν τα όρια τους το HpcLastStgBHeight και το TBO. Με μικρότερη μηχανή μειώνεται το ύψος των πτερυγίων της τελευταίας βαθμίδας του συμπιεστή υψηλής πίεσης, μέχρι να φτάσει το όριο που του έχει οριστεί. Με μικρότερη μηχανή και πτερύγια αυξάνεται η μέση θερμική καταπόνηση που δέχεται συνολικά η μηχανή με αποτέλεσμα ο χρόνος ζωής της να μειώνεται, με αποτέλεσμα μεγαλύτερο κόστος συντήρησης της, λόγω της συχνής ανάγκης για γενική επιθεώρηση. Οι θερμοκρασίες όπως φαίνεται αυξάνονται, κάτι που είναι λογικό καθώς έτσι αυξάνεται ο θερμικός βαθμός απόδοσης της μηχανής αλλά καμία δεν αγγίζει τα όρια που έχουν τεθεί. Στον Πίνακα 5.23 φαίνονται οι ποσοστιαίες διαφορές κάποιων βασικών παραμέτρων του προβλήματος σε σχέση με τις τιμές αναφοράς τους.

Πίνακας 5.23

Διαφορά επί % από τις τιμές αναφοράς	
ETAovCR	0.438
ETApropCR	1.080
ETAthCR	-0.635
FN_CR	-18.893
OPR_CR	17.525
SFC_CR	-0.436
SFN_CR	-10.474
Engine Weight	-14.048
BPR_CR	8.123
BladeCreepLife	-49.435
BladeFatigueLife	-29.588
Aircraft Wing Area	-2.936
Aircraft WingSpan	-1.479
Aircraft operating empty weight:	-3.166
Aircraft maximum ramp weight:	-2.919
Aircraft maximum take-off weight:	-2.936
Aircraft maximum landing weight:	-2.587
Aircraft maximum zero fuel weight:	-2.370
Aircraft maximum fuel weight:	-2.770
Aircraft ramp weight:	-2.966
Aircraft take-off weight:	-2.936
Aircraft landing weight:	-2.587
Aircraft zero fuel weight:	-2.370
Total fuel weight:	-5.451

Παρατηρούμε, ότι η ώση (FN) μειώνεται ως αποτέλεσμα της μικρότερης μηχανής και επομένως μειώνεται και το βάρος της (Engine Weight). Η μείωση του μεγέθους της μηχανής έχει ως αποτέλεσμα όπως εξηγήθηκε νωρίτερα την μείωση του χρόνου ζωής των εξαρτημάτων κάτι που φαίνεται και από την τεράστια μείωση των παραμέτρων BladeCreepLife και BladeFatigueLife μειώνοντας σημαντικά το TBO.

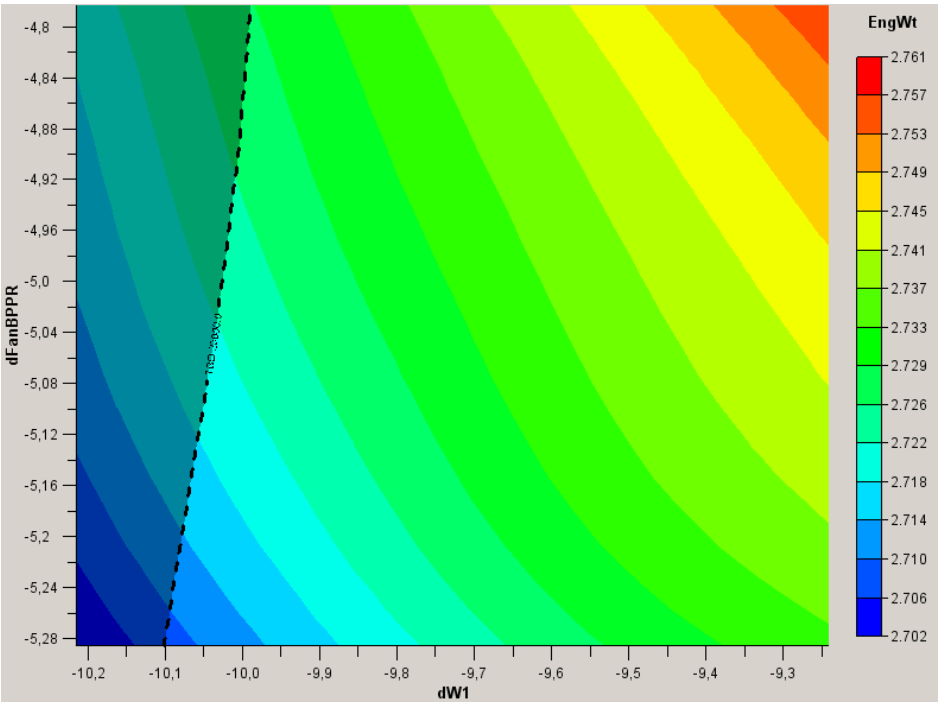
Ο θερμικός βαθμός απόδοσης μειώνεται ελάχιστα παρόλο που υπήρχε σημαντική αύξηση του συνολικού λόγου πίεσης (OPR) και της θερμοκρασίας εισόδου στον στρόβιλο, λόγω αύξησης του λόγου παράκαμψης. Από την άλλη, ο προωθητικός βαθμός αυξάνεται με την αύξηση του λόγου παράκαμψης και μάλιστα λίγο περισσότερο από την αντίστοιχη μείωση του θερμικού βαθμού απόδοσης. Επομένως ο συνολικός βαθμός απόδοσης, που αποτελεί συνδυασμό του προωθητικού και θερμικού βαθμού απόδοσης βελτιώνεται με επακόλουθη βελτίωση του SFC.

Το βάρος του κινητήρα και συνολικά του αεροσκάφους μειώνεται . Η μείωση αυτή μαζί με την βελτίωση του SFC έχει ως αποτέλεσμα τελικά, την συνολική μείωση της κατανάλωσης καυσίμου στην αποστολή του αεροσκάφους. Σημαντικότερη βέβαια επίδραση έχει η μείωση του συνολικού βάρους.

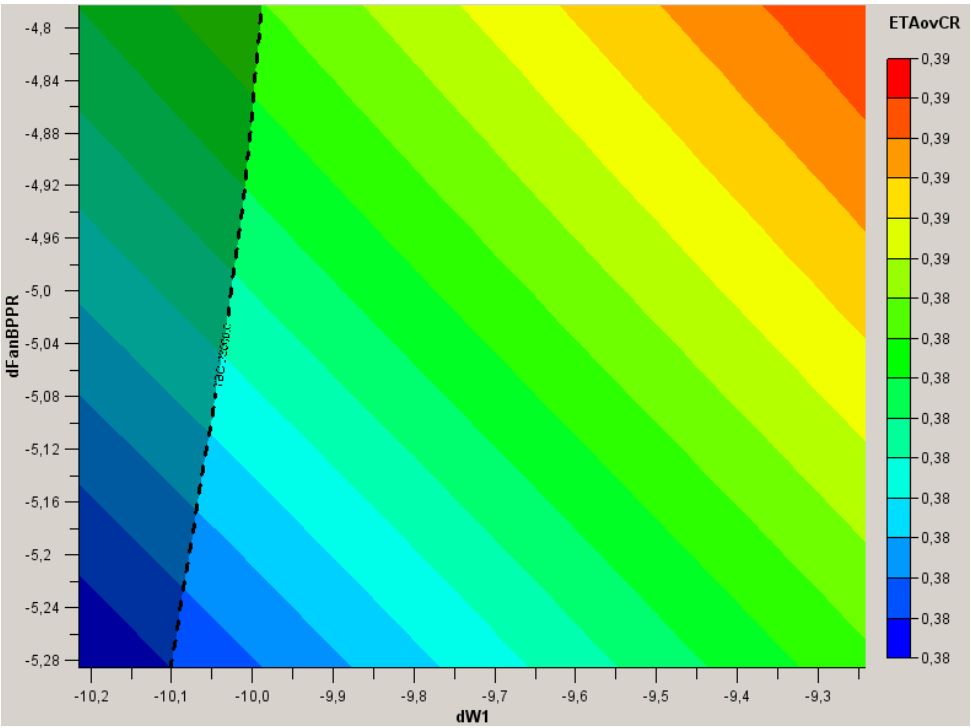
Έχοντας βρει το βέλτιστο σημείο σχεδιασμού, δημιουργείται ένας χώρος σχεδιασμού με την χρήση RBF μεταπροτύπων με την μέθοδο που χρησιμοποιήθηκε προηγουμένως. Οι μεταβλητές σχεδίασης μεταβλήθηκαν [-5%,+5%] από την τιμή τους, η οποία αποτελεί την καλύτερη λύση που βρέθηκε από την βελτιστοποίηση. Στην συνέχεια παρουσιάζεται η επίδραση των μεταβλητών σχεδίασης στις σημαντικότερες παραμέτρους του Πίνακα 5.23 με επικάλυψη περιορισμών, για την καλύτερη κατανόηση των αποτελεσμάτων.

Στα Σχήματα 5.36 έως 5.42 παρουσιάζονται οι μεταβλητές FanBPPR και W1 οι οποίες καθορίζουν τον λόγο πίεσης του Fan και την παροχή στον αγωγό εισόδου, με επικάλυψη των περιορισμών. Στην συγκεκριμένη περίπτωση ο περιορισμός που αγγίζει τα όρια πρώτος είναι ο TBO. Στα σχήματα φαίνεται η τάση αύξησης του FPR και της παροχής W1 για την βελτίωση της απόδοσης. Το σχήμα 5.42 δείχνει ότι η αύξηση της παροχής W1 θα μείωνε περαιτέρω την ειδική ώση SFN, ωστόσο αυτό θα είχε ως αποτέλεσμα την μείωση της ώσης FN άρα αδυναμία εκπλήρωσης των περιορισμών του FAR. Επίσης φαίνεται να υπάρχει περιθώριο βελτίωσης στο συνολικό καύσιμο AcBlockFuel, ωστόσο τα σχήματα προκύπτουν από extrapolation του δημιουργούμενου μεταπροτύπου και η απεικόνισή τους δεν είναι ακριβής.

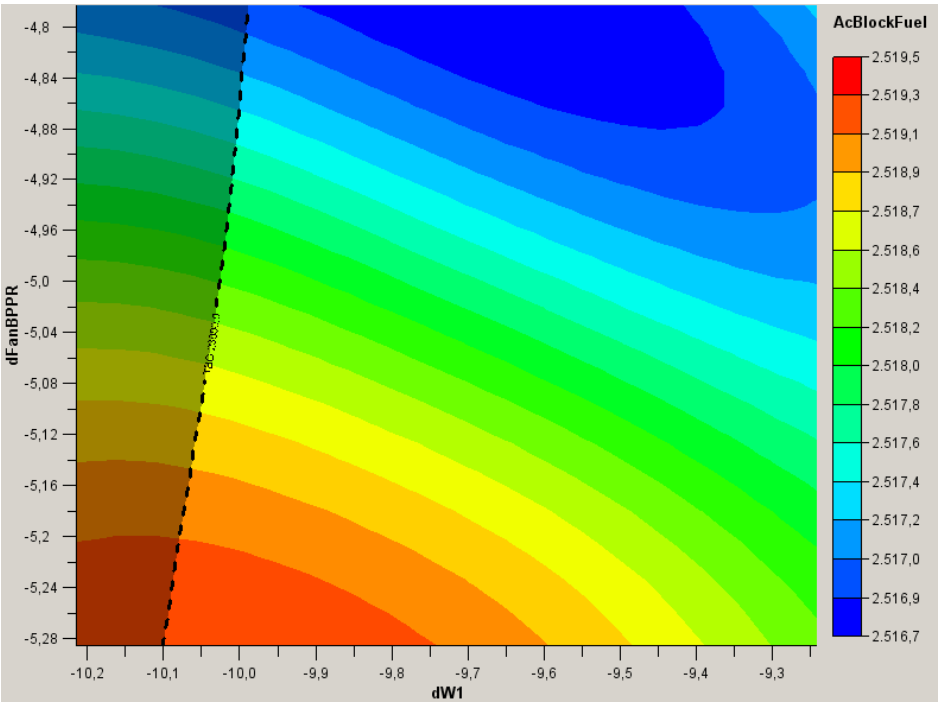
Στα Σχήματα 5.43-5.49 παρουσιάζεται η επίδραση των λόγων πίεσης των συμπιεστών στις ίδιες παραμέτρους. Παρατηρούμε ότι η βελτίωση των λόγων πίεσης των συμπιεστών έχει φτάσει στο όριο τους. Περαιτέρω αύξηση των λόγων πίεσης εμποδίζεται από τον περιορισμό του TBO. Αυτό φαίνεται και από το Σχήμα 5.45 όπου δεν υπάρχουν περιθώρια για την βελτίωση του συνολικού βαθμού απόδοσης. Επομένως ο πιο καθοριστικός περιορισμός στην συγκεκριμένη περίπτωση, προκύπτει να είναι το λειτουργικό κόστος της μηχανής.



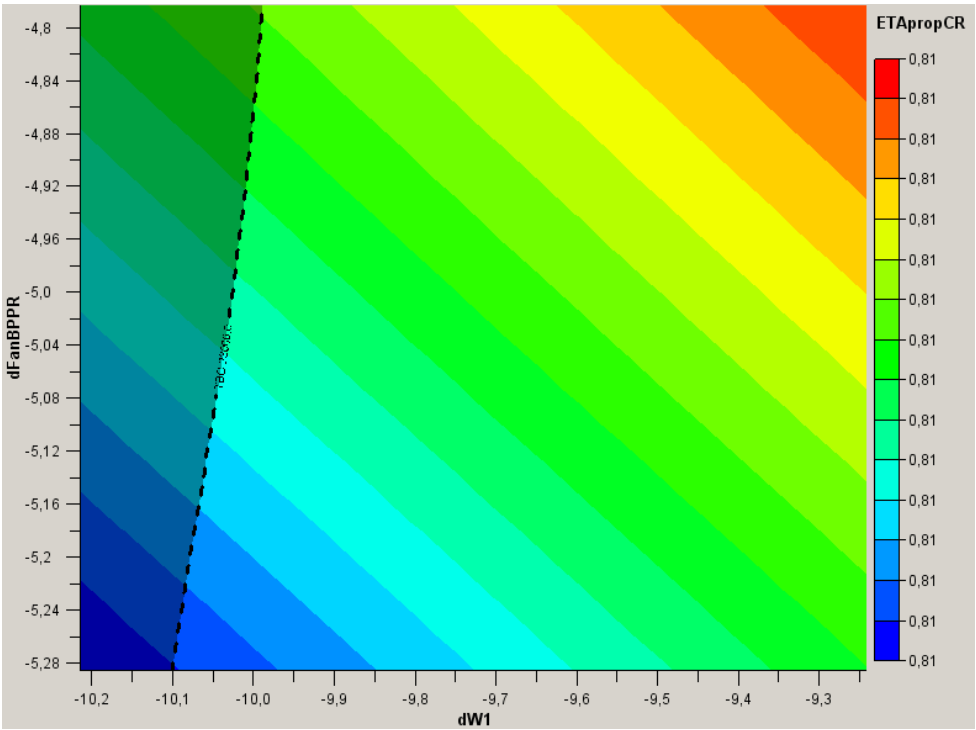
Σχήμα 5.36 Μεταβλητές σχεδιασμού FPR και W1 σε σχέση με το EngineWeight



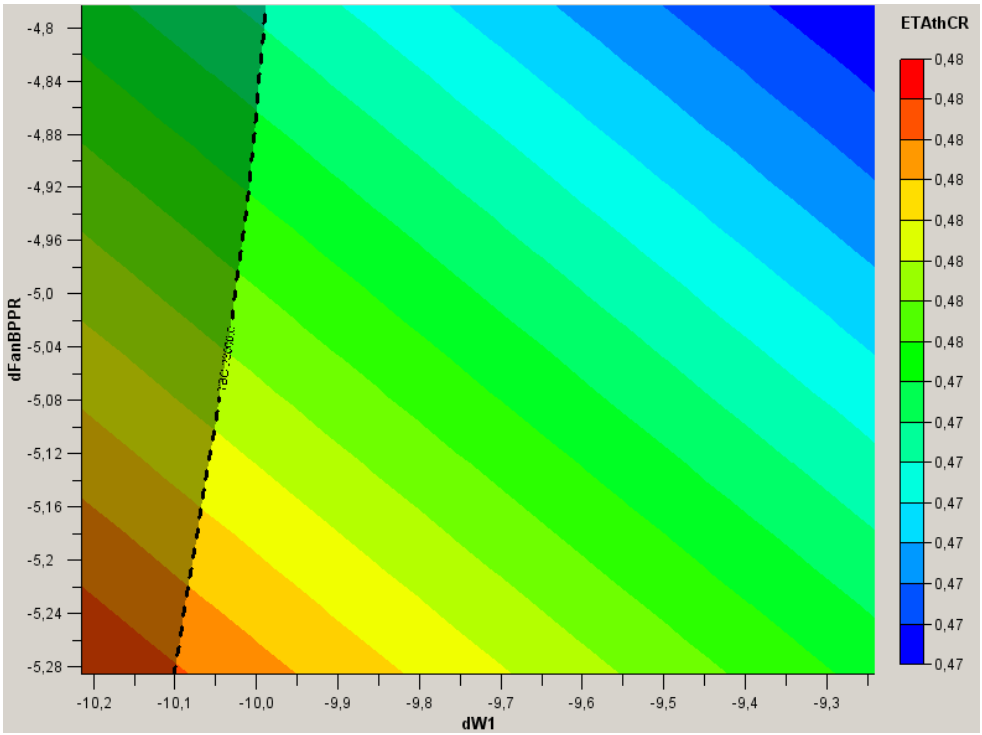
Σχήμα 5.37 Μεταβλητές σχεδιασμού FPR και W1 σε σχέση με τον ETAovCR



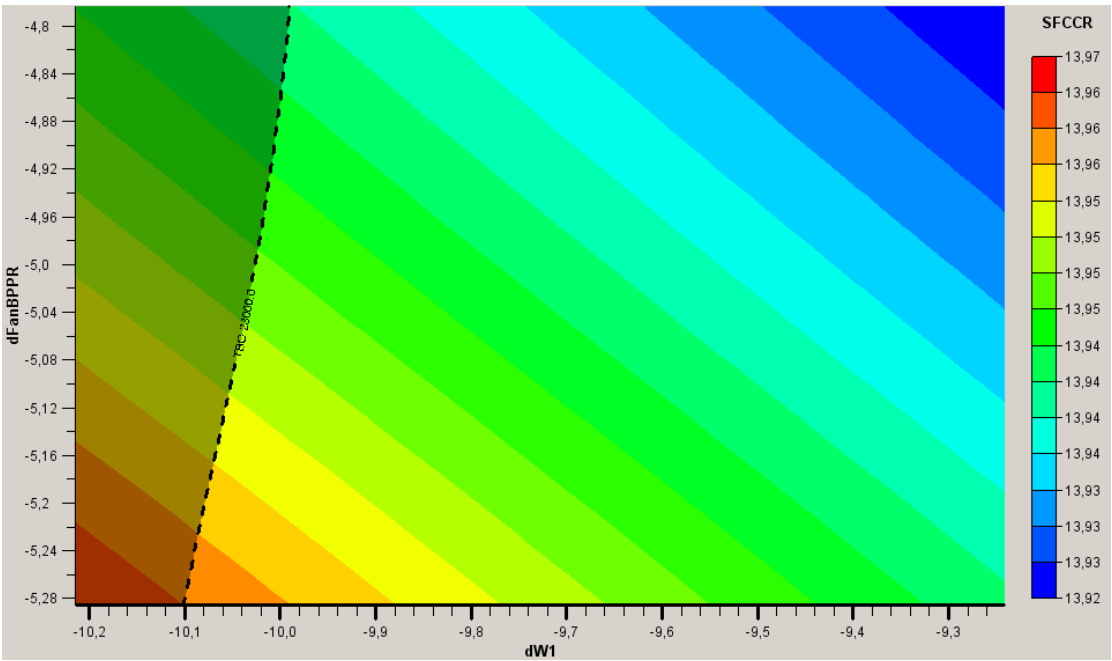
Σχήμα 5.38 Μεταβλητές σχεδιασμού FPR και W1 σε σχέση με την αντικειμενική συνάρτηση AcBlockFuel



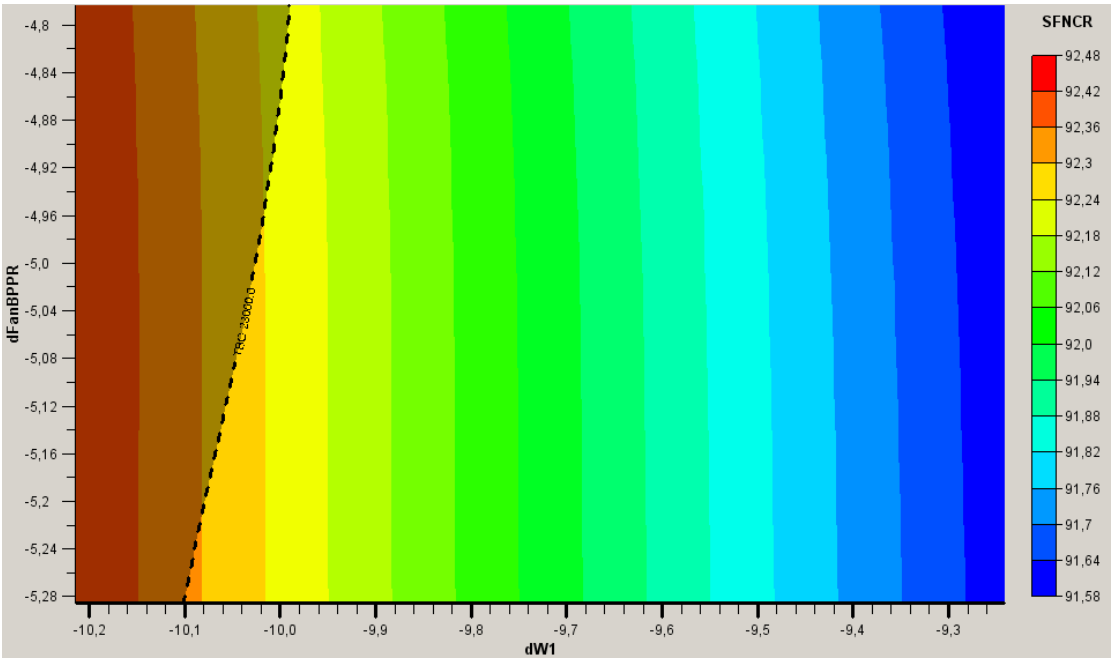
Σχήμα 5.39 Μεταβλητές σχεδιασμού FPR και W1 σε σχέση με τον ETApropCR



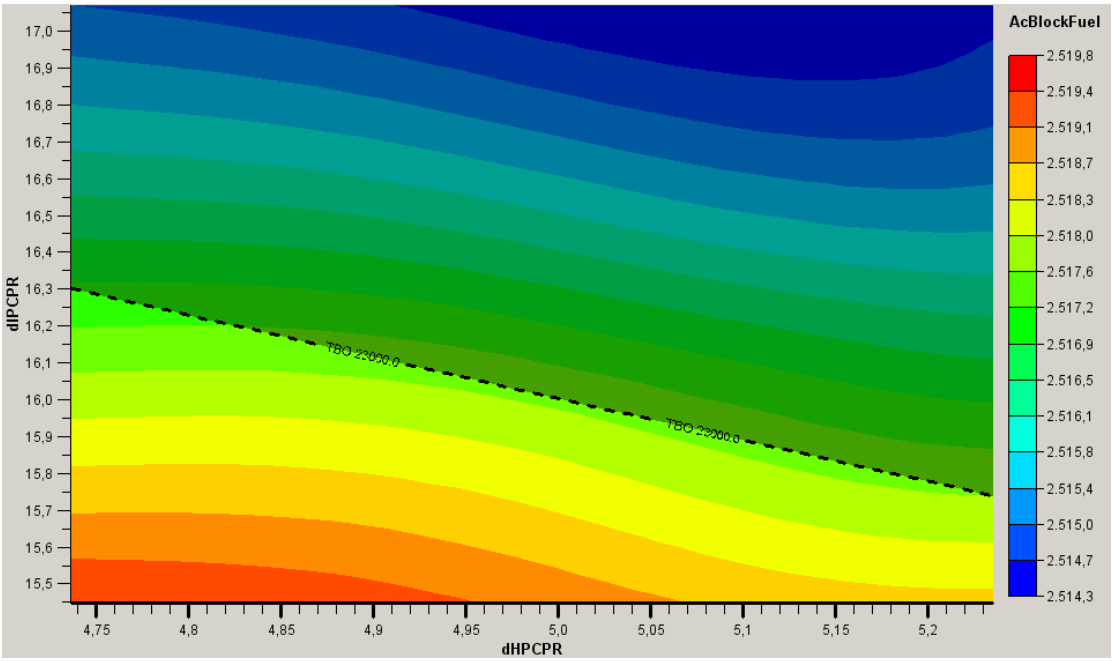
Σχήμα 5.40 Μεταβλητές σχεδιασμού FPR και W1 σε σχέση με τον ETAthCR



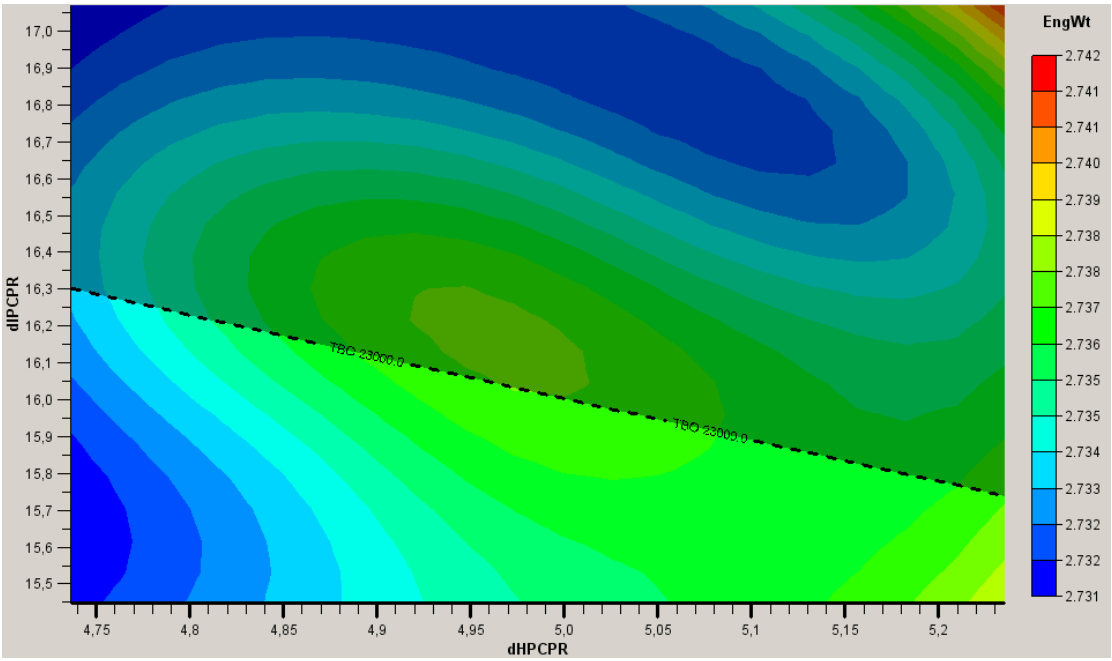
Σχήμα 5.41 Μεταβλητές σχεδιασμού FPR και W1 σε σχέση με τη SFC_CR



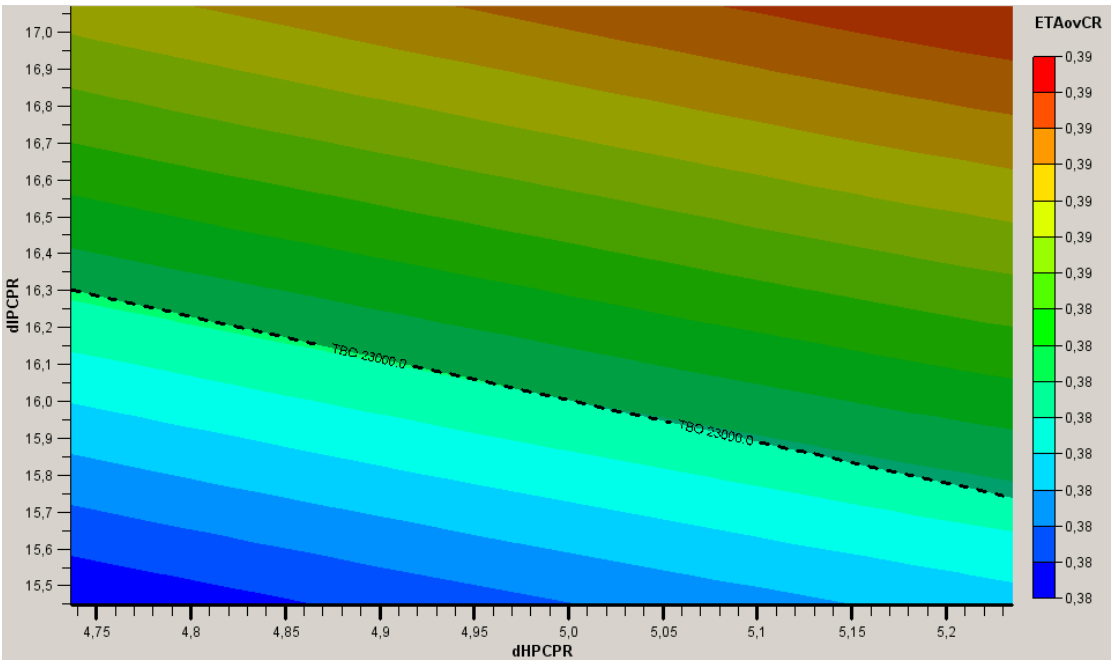
Σχήμα 5.42 Μεταβλητές σχεδιασμού FPR και W1 σε σχέση με τη SFN_CR



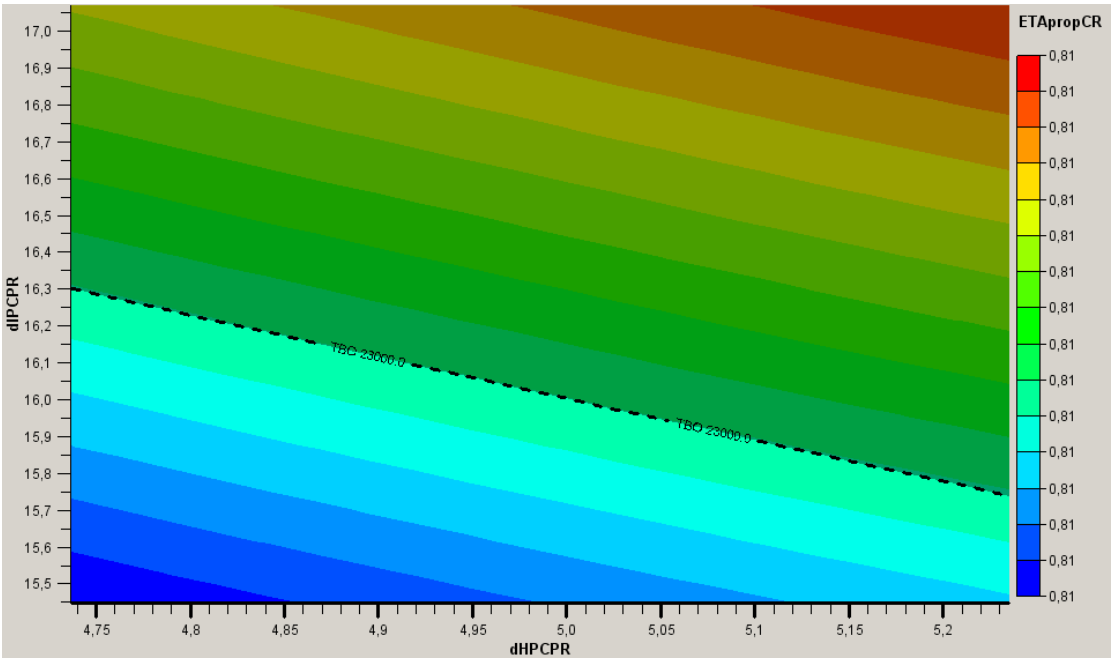
Σχήμα 5.43 Μεταβλητές σχεδιασμού IPCPR και HPCPR σε σχέση με το AcBlocFuel



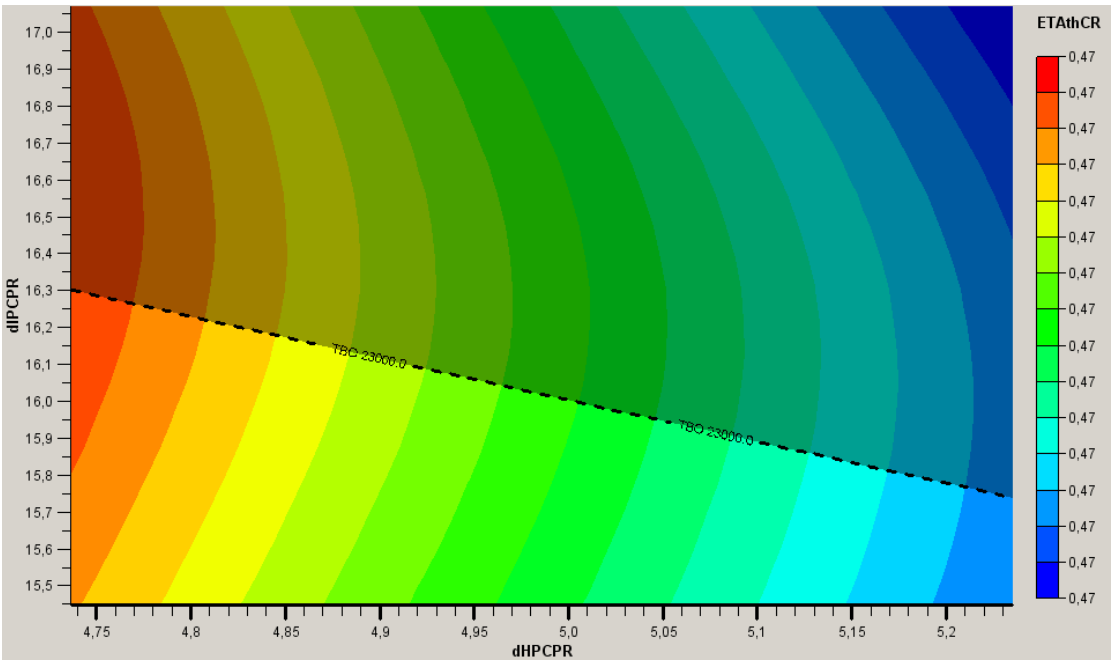
Σχήμα 5.44 Μεταβλητές σχεδιασμού IPCPR και HPCPR σε σχέση με το EngineWeight



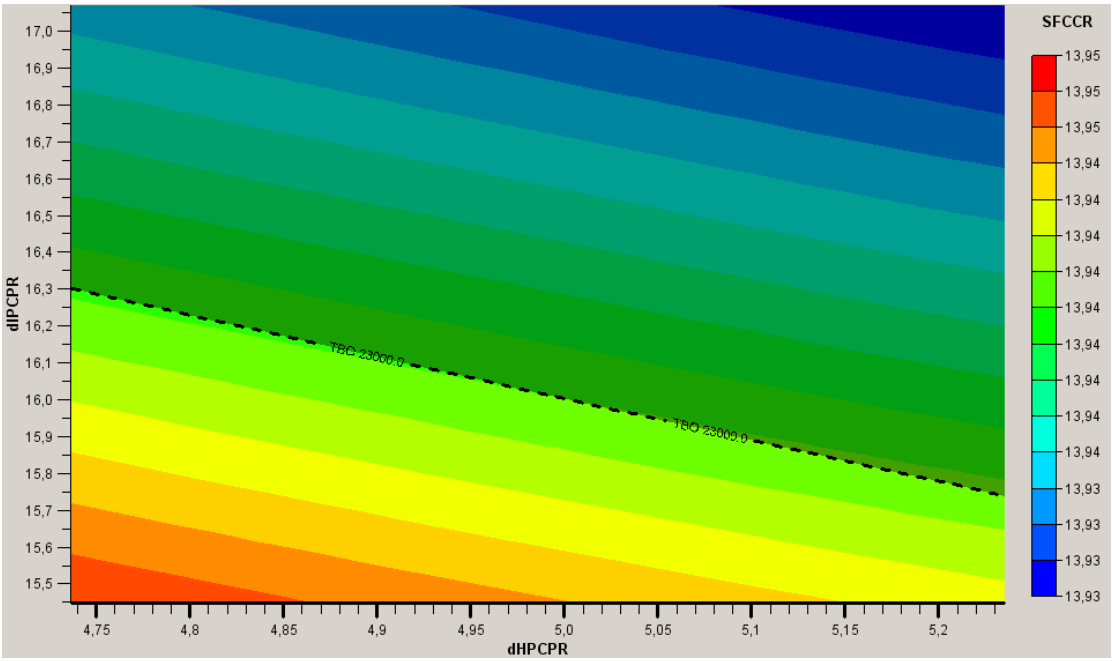
Σχήμα 5.45 Μεταβλητές σχεδιασμού IPCPR και HPCPR σε σχέση με το ETAovCR



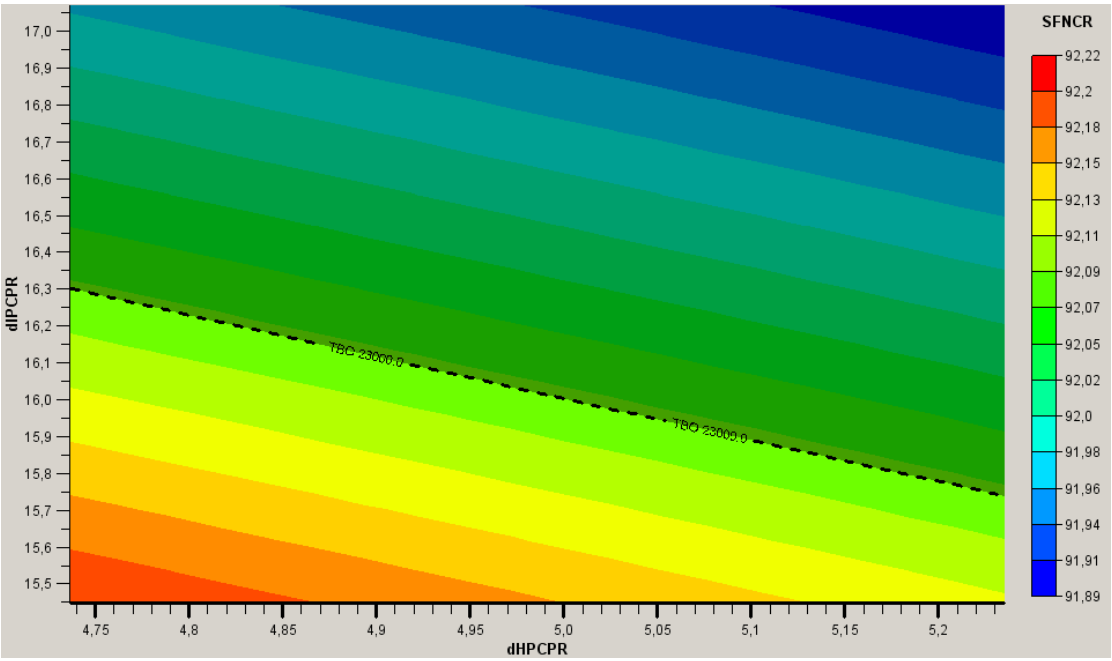
Σχήμα 5.46 Μεταβλητές σχεδιασμού IPCPR και HPCPR σε σχέση με το ETApropCR



Σχήμα 5.47 Μεταβλητές σχεδιασμού IPCPR και HPCPR σε σχέση με το ETAthCR



Σχήμα 5.48 Μεταβλητές σχεδιασμού IPCPR και HPCPR σε σχέση με τη SFC_CR



Σχήμα 5.49 Μεταβλητές σχεδιασμού IPCPR και HPCPR σε σχέση με τη SFN_CR

6 Ανακεφαλαίωση-Συμπεράσματα-Προτάσεις

6.1 Ανακεφαλαίωση

Στα πλαίσια της παρούσας διπλωματικής εργασίας πραγματοποιήθηκε η βελτιστοποίηση του προωστήριου συστήματος ενός αεροσκάφους με στόχο την μείωση της κατανάλωσης καυσίμου του. Για την βελτιστοποίηση χρησιμοποιήθηκε το εμπορικό λογισμικό Isight.

Αρχικά έγινε μια περιγραφή του λογισμικού Isight. Έπειτα μελετήθηκαν οι αλγόριθμοι και οι τεχνικές που περιλαμβάνονται σε αυτό και στη συνέχεια μετά από εκτενή βιβλιογραφική μελέτη επιλέχθηκαν οι πιο κατάλληλες για το συγκεκριμένο πρόβλημα που μελετάται. Επίσης η επιλογή των ρυθμίσεων και παραμέτρων κάθε τεχνικής έγινε με βάση την βιβλιογραφία και τις προκαθορισμένες τιμές του λογισμικού. Οι τεχνικές που επιλέχθηκαν και χρησιμοποιήθηκαν ήταν δύο από την κατηγορία Design of Experiments, η Optimal Latin Hypercube και η Parameter Study, τέσσερις αλγόριθμοι βελτιστοποίησης, ο ASA, ο Downhill Simplex, ο PSO και ο MIGA, ενώ χρησιμοποιήθηκαν και δύο τεχνικές δημιουργίας μεταπροτύπων η Kriging και τα RBF.

Στην συνέχεια γίνεται εφαρμογή των παραπάνω μεθόδων και σύγκριση των αποτελεσμάτων τους σε μια απλή περίπτωση για την βελτιστοποίηση του κινητήρα CFM56, που έχει μοντελοποιηθεί στο PROOSIS από το ΕΘΣ, με στόχο την μείωση της ειδικής κατανάλωσης καυσίμου του κατά την κρουαζιέρα. Σκοπός της εφαρμογής αυτής ήταν η εξοικείωση με το λογισμικό και η προσπάθεια ερμηνείας των αποτελεσμάτων των μεθόδων σε μια απλή περίπτωση.

Στη συνέχεια γίνεται εφαρμογή των ίδιων μεθόδων σε μια πιο πολύπλοκη περίπτωση για την βελτιστοποίηση του προωστήριου συστήματος ενός αεροσκάφους με στόχο την μείωση της κατανάλωσης καυσίμου σε ολόκληρη την αποστολή. Για το σκοπό αυτό χρησιμοποιείται το λογισμικό αξιολόγησης TERA που περιλαμβάνει μοντέλο του κινητήρα, μοντέλο αεροσκάφους, υπολογισμό βάρους αεροσκάφους - κινητήρα, υπολογισμό ήχου και ρύπων καθώς και υπολογισμό χρόνου μεταξύ διαστημάτων συντήρησης κινητήρα (TBO). Εκτός από τις παραπάνω μεθόδους, στην περίπτωση αυτή, έγινε επίσης χρήση του λογισμικού βελτιστοποίησης EASY που έχει αναπτυχθεί από το ΕΘΣ. Τέλος αναλύονται τα αποτελέσματα της βελτιστοποίησης και συγκρίνονται οι τεχνικές με βάση τα αποτελέσματα τους και το χρόνο σύγκλισής τους.

6.2 Συμπεράσματα

Με βάση τα αποτελέσματα που παρουσιάστηκαν, μπορούν να εξαχθούν τα εξής συμπεράσματα, όσον αφορά τις τεχνικές βελτιστοποίησης:

- Με την χρήση της τεχνικής Design of Experiments δεν γίνεται επαρκής οπτικοποίηση του χώρου σχεδιασμού σε δύο διαστάσεις, όταν υπάρχει μεγάλος αριθμός μεταβλητών σχεδιασμού. Προτείνεται η χρήση της μόνο σε προβλήματα με μικρό αριθμό μεταβλητών σχεδιασμού.
- Οι πιο κατάλληλοι αλγόριθμοι για την αντιμετώπιση των συγκεκριμένων προβλημάτων βρέθηκαν να είναι ο PSO και ο ASA, καθώς επίσης και το εργαλείο βελτιστοποίησης EASY.
- Ωστόσο, όλοι οι αλγόριθμοι που προτάθηκαν, βγάζουν ικανοποιητικά αποτελέσματα με τις κατάλληλες ρυθμίσεις αλλά υπάρχει ευαισθησία στην κατάλληλη ρύθμιση των παραμέτρων τους.
- Ο αλγόριθμος D. Simplex μπορεί να χρησιμοποιηθεί μόνο αν γίνεται συνεχής επανεκκίνηση του, από διαφορετικό αρχικό σημείο σχεδιασμού αλλά και πάλι η πιθανότητα να εγκλωβιστεί σε τοπικό ακρότατο είναι μεγάλη.
- Υπάρχει αύξηση της πιθανότητας εύρεσης της καλύτερης λύσης με μείωση των ορίων αναζήτησης, κάτι που φαίνεται και στα αποτελέσματα της δεύτερης εφαρμογής.
- Η χρήση offline μεταπροτύπων είναι κατάλληλη για απλά προβλήματα με λίγες μεταβλητές σχεδιασμού, ωστόσο σε πιο πολύπλοκα προβλήματα ακόμα και με περιορισμένο εύρος αναζήτησης παρουσιάζουν σημαντικό σφάλμα.
- Για τα προβλήματα που μελετήθηκαν προτείνονται τα μεταπρότυπα RBF έναντι του Kriging λόγω καλύτερης συμπεριφοράς.
- Προτείνεται η χρήση online μεταπροτύπων όπως εφαρμόζονται από το λογισμικό βελτιστοποίησης EASY.

Όσον αφορά τις συγκεκριμένες εφαρμογές:

- Η βελτιστοποίηση μόνο του αεροπορικού κινητήρα χωρίς να ληφθούν υπόψη άλλες συνιστώσες ή κριτήρια οδηγεί σε λάθος συμπεράσματα.
- Η αποστολή του αεροσκάφους μπορεί να επιτευχθεί με μικρότερη μηχανή με βάση τους περιορισμούς που έχουμε επιλέξει.
- Ο TBO περιορισμός είναι τελικά ο καθοριστικός παράγοντας για την βελτιστοποίηση της συνολικής κατανάλωσης καυσίμου.

Προτάσεις για μελλοντική έρευνα:

- Μελέτη της 2ης εφαρμογής με περισσότερες μεταβλητές σχεδιασμού και περισσότερους στόχους ή και κριτήρια.
- Συστηματική μελέτη της επίδρασης των παραμέτρων κάθε αλγορίθμου, στα αποτελέσματα του.
- Μελέτη χρησιμοποίησης συνδυαστικά δύο αλγορίθμων ενός φθηνού στην αρχή για περιορισμό των ορίων αναζήτησης και στην συνέχεια χρήση ενός ακριβού για την εύρεση της βέλτιστης λύσης.

7 Βιβλιογραφία

[1] L. Europe ACARE %J Publications of the European Communities, “European Aeronautics: a vision for 2020. Meeting society’s needs and winning global leadership,” 2001.

[2] C. Celis, R. Long, V. Sethi, and D. Zammit-Mangion, “On Trajectory Optimization for Reducing the Impact of Commercial Aircraft Operations on the Environment,” *XIX Int. Symp. Air Breath. Engines*, pp. 7–11, 2009.

[3] I. Goulos, V. Pachidis, C. Celis, R. D’Ippolito, and J. Stevens, “Simulation framework development for aircraft mission analysis,” *ASME Turbo Expo 2010 Power Land Sea Air*, pp. 341–351, 2010.

[4] B. Kestner, J. S. Schutte, J. C. Tai, C. A. Perullo, and D. N. Mavris, “Surrogate modeling for simultaneous engine cycle and technology optimization for next generation subsonic aircraft,” *ASME Turbo Expo 2012 Turbine Tech. Conf. Expo.*, pp. 133–145, 2012.

[5] R. d’Ippolito, J. Stevens, V. Pachidis, A. Berta, I. Goulos, and C. Rizzi, “A Multidisciplinary Simulation Framework for Optimization of Rotorcraft Operations and Environmental Impact,” *2nd Int. Conf. Eng. Optim. EngOpt 2010 Lisbon Port. Sept.*, pp. 6–9, 2010.

[6] J. Stevens, R. D???Ippolito, V. Pachidis, M. Mohseni, and I. Goulos, “Simulation framework development for helicopter mission analysis,” *ASME Turbo Expo 2010 Power Land Sea Air*, pp. 843–852, 2010.

[7] F. Ali, K. Tzanidakis, I. Goulos, V. Pachidis, R. %J J. of E. for G. T. d’Ippolito, and Power, “Multi-objective optimization of conceptual rotorcraft powerplants: trade-off between rotorcraft fuel efficiency and environmental impact,” vol. 137, no. 7, p. 071201, 2015.

[8] K. G. Kyprianidis, A. M. Rolt, T. %J J. of E. for G. T. Gr??nstedt, and Power, “Multidisciplinary analysis of a geared fan intercooled core aero-engine,” vol. 136, no. 1, p. 011203, 2014.

[9] K. G. Kyprianidis and A. M. Rolt, “On the optimisation of a geared fan intercooled core engine design,” *ASME Turbo Expo 2014 Turbine Tech. Conf. Expo.*, p. V03AT07A018-V03AT07A018, 2014.

[10] K. G. Kyprianidis, R. F. C. Quintero, D. S. Pascovici, S. O. Ogaji, P. Pilidis, and A. I. Kalfas, “EVA: a tool for environmental assessment of novel propulsion cycles,” *ASME Turbo Expo 2008 Power Land Sea Air*, pp. 547–556, 2008.

[11] K. G. Kyprianidis, “Multi-disciplinary conceptual design of future jet engine systems,” 2010.

[12] L. Larsson, T. Gro??nstedt, and K. G. Kyprianidis, “Conceptual design and mission analysis for a geared turbofan and an open rotor configuration,” *ASME 2011 Turbo Expo Turbine Tech. Conf. Expo.*, pp. 359–370, 2011.

[13] “Isight & SIMULIA™ Execution Engine - Dassault Systèmes®.” [Online]. Available: https://www.3ds.com/products-services/simulia/products/isight-simulia-execution-engine/. [Accessed: 21-Feb-2019].

[14] “NEWAC Deliverable D1.3.2C TERA2020 Optimisation for NEWAC Powerplants.” 05-Jan-2006.

[15] E. A. Internacional, “PROOSIS, Propulsion Object Oriented Simulation Software,” *EcosimPro | PROOSIS*. [Online]. Available: https://www.ecosimpro.com/products/proosis/. [Accessed: 21-Feb-2019].

[16] “National Technical University of Athens, School of Mechanical Engineering,Parallel CFD and Optimization Unit.” .

[17] A. Alexiou, “NEWAC TERA2020 GTCRSR Optimisation Results - R1.2,” Mar. 2011.

[18] X. Deng, F. Guo, Y. Liu, and P. Han, “Aero-Mechanical Optimization Design of a Transonic Fan Blade,” *ASME Turbo Expo 2013 Turbine Tech. Conf. Expo.*, p. V06BT43A014-V06BT43A014, 2013.

[19] S. Naz, “Multidisciplinary Design Optimization (MDO) of Transonic Fan Blade,” 2014.

[20] L. Larsson, R. Avell??n, and T. Gr??nstedt, “Mission optimization of the geared turbofan engine,” *Int. Soc. Airbreathing Engines ISABE Gothenbg. 2011ISABE-2011-1314*, 2011.

[21] R. Jin, W. Chen, and A. Sudjianto, “An efficient algorithm for constructing optimal design of computer experiments,” *ASME 2003 Int. Des. Eng. Tech. Conf. Comput. Inf. Eng. Conf.*, pp. 545–554, 2003.

[22] M. E. Johnson, L. M. Moore, D. %J J. of statistical planning Ylvisaker, and inference, “Minimax and maximin distance designs,” vol. 26, no. 2, pp. 131–148, 1990.

[23] M. D. Morris, T. J. %J J. of statistical planning Mitchell, and inference, “Exploratory designs for computational experiments,” vol. 43, no. 3, pp. 381–402, 1995.

[24] A. Antonakis, T. Nikolaidis, and P. %J A. S. Pilidis, “Multi-Objective Climb Path Optimization for Aircraft/Engine Integration Using Particle Swarm Optimization,” vol. 7, no. 5, p. 469, 2017.

[25] X. Hu, X. Chen, Y. Zhao, and W. %J A. in S. R. Yao, “Optimization design of satellite separation systems based on multi-island genetic algorithm,” vol. 53, no. 5, pp. 870–876, 2014.

[26] N. Srinivas and K. %J E. computation Deb, “Muiltiobjective optimization using nondominated sorting in genetic algorithms,” vol. 2, no. 3, pp. 221–248, 1994.

[27] A. Van der Velden and P. %J A. H. Koch, “Isight design optimization methodologies,” vol. 22, p. 79, 2010.

[28] J. Ghosh and A. Nag, “An overview of radial basis function networks,” *Radial Basis Funct. Netw. 2*, pp. 1–36, 2001.

[29] R. L. %J J. of geophysical research Hardy, “Multiquadric equations of topography and other irregular surfaces,” vol. 76, no. 8, pp. 1905–1915, 1971.

[30] E. J. %J R. Kansa, “Motivation for using radial basis functions to solve PDEs,” vol. 64, no. 1, p. 1, 1999.

[31] Z. Skolicki and K. De Jong, “Improving evolutionary algorithms with multi-representation island models,” *Int. Conf. Parallel Probl. Solving Nat.*, pp. 420–429, 2004.

[32] S. Devanathan and P. N. Koch, “Comparison of meta-modeling approaches for optimization,” *ASME 2011 Int. Mech. Eng. Congr. Expo.*, pp. 827–835, 2011.

[33] R. Jin, W. Chen, T. W. %J S. Simpson, and multidisciplinary optimization, “Comparative studies of metamodelling techniques under multiple modelling criteria,” vol. 23, no. 1, pp. 1–13, 2001.

[34] L. Chen, H. Wang, F. Ye, W. %J S. Hu, and M. Optimization, “Comparative study of HDMRs and other popular metamodeling techniques for high dimensional problems,” pp. 1–22, 2018.

[35] Z. Skolicki and K. De Jong, “The influence of migration sizes and intervals on island models,” *Proc. 7th Annu. Conf. Genet. Evol. Comput.*, pp. 1295–1302, 2005.

[36] D. Whitley, S. Rana, R. B. %J J. of computing Heckendorn, and information technology, “The island model genetic algorithm: On separability, population size and convergence,” vol. 7, no. 1, pp. 33–47, 1999.

[37] N. Holtschulte, “Optimal Population Size in Island Model Genetic Algorithms,” *8th Stud. Conf.*, p. 94, 2012.

[38] M. Clerc, “Stagnation analysis in particle swarm optimisation or what happens when nothing happens,” 2006.

[39] A. Serani *et al.*, “On the use of synchronous and asynchronous single-objective deterministic particle swarm optimization in ship design problems,” *1st Int. Conf. Eng. Appl. Sci. Optim. Kos*, 2014.

[40] National Technical University of Athens, School of Mechanical Engineering, and Parallel CFD and Optimization Unit, *The Evolutionary Algorithm SYstem, User’ s Manual*, 2.0. 2008.

[41] “NEWAC Deliverable D2.3.1B -Final Evaluation for Future Innovative Core Architectures, NEWAC-VAC-DEL-2.3.1B-R0.2,” Feb. 2011.

[42] P. I. Giannatselis, “Μέθοδος και Λογισμικό Παραμετροποίησης της Μορφής Πτερυγώσεων Στροβιλομηχανών,” B.S. thesis, 2014.

[43] M. A. El-Beltagy and A. J. Keane, “A comparison of various optimization algorithms on a multilevel problem,” *Eng. Appl. Artif. Intell.*, vol. 12, no. 5, pp. 639–654, 1999.

[44] T. Bäck, “Parallel optimization of evolutionary algorithms,” in *International Conference on Parallel Problem Solving from Nature*, 1994, pp. 418–427.

[45] Y.-J. Gong *et al.*, “Distributed evolutionary algorithms and their models: A survey of the state-of-the-art,” *Appl. Soft Comput.*, vol. 34, pp. 286–300, 2015.

[46] Y. Jin, “Surrogate-assisted evolutionary computation: Recent advances and future challenges,” *Swarm Evol. Comput.*, vol. 1, no. 2, pp. 61–70, 2011.

[47] Y. S. Ong, P. B. Nair, and A. J. Keane, “Evolutionary optimization of computationally expensive problems via surrogate modeling,” *AIAA J.*, vol. 41, no. 4, pp. 687–696, 2003.

[48] M. K. Karakasis and K. C. Giannakoglou, “On the use of metamodel-assisted, multi-objective evolutionary algorithms,” *Eng. Optim.*, vol. 38, no. 8, pp. 941–957, 2006.

[49] K. C. Giannakoglou, A. P. Giotis, and M. K. Karakasis, “Low-cost genetic optimization based on inexact pre-evaluations and the sensitivity analysis of design parameters,” *Inverse Probl. Eng.*, vol. 9, no. 4, pp. 389–412, 2001.

[50] L. Bull, “On model-based evolutionary computation,” *Soft Comput.*, vol. 3, no. 2, pp. 76–82, 1999.