

ΠΟΛΥΤΕΧΝΕΙΟ ΚΡΗΤΗΣ

*Εφαρμογή του αλγόριθμου
Διαφορικής Εξέλιξης στο
πρόβλημα του βέλτιστου
σχεδιασμού γραμμής
προϊόντων*

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ ΚΑΙ
ΔΙΟΙΚΗΣΗΣ

ΛΕΥΤΕΡΗΣ ΧΑΡΑΛΑΜΠΟΥΣ
2019

Ευχαριστίες

Με την εκπόνηση της παρούσας διπλωματικής εργασίας κλείνει ένας μεγάλος κύκλος για εμένα στο Πολυτεχνείο Κρήτης , στο τμήμα Μηχανικών Παραγωγής και Διοίκησης. Σε αυτό το σημείο θα ήθελα να καταρχάς να ευχαριστήσω τον υπεύθυνο καθηγητή μου , κύριο Τσαφαράκη Στέλιο για την εναπόθεση της συγκεκριμένης εργασίας και το πολύτιμο υλικό που μου παρείχε και τον υποψήφιο Διδάκτωρ , κύριο Ζερβουδάκη Κωνσταντίνο για την καθοδήγηση του καθ' όλη της διάρκειας εκπόνησης της εργασίας. Επίσης θα ήθελα να ευχαριστήσω την οικογένεια μου για την στήριξη της όλο αυτό το διάστημα. Τέλος θα ήθελα να αφιερώσω αυτή την εργασία στον παππού μου Λευτέρη Στυλιανίδη.

Περιεχόμενα

Ευχαριστίες	1
Περίληψη	6
1. Εισαγωγή	7
1.1. Δομή της Εργασίας	7
1.2. Το Πρόβλημα Βέλτιστου Σχεδιασμού γραμμής προϊόντων	8
2. Έρευνα αγοράς.....	10
2.1 Μέτρηση καταναλωτικών προτιμήσεων.....	10
2.1.1 Ανάλυση Συζυγιών (Conjoint Analysis).....	10
2.1.3 Η συλλογή των δεδομένων	13
2.2 Μοντελοποίηση της Απόφασης	14
2.3 Κριτήριο Βελτιστοποίησης	16
2.4 Μοντελοποίηση του προβλήματος	17
2.5 Μοντελοποίηση του υπό μελέτη προβλήματος	18
3. Προσεγγιστικοί Αλγόριθμοι	20
3.1 Ευρετικοί και Μεθευρετικοί Αλγόριθμοι	20
3.2 Αλγόριθμοι που έχουν εφαρμοστεί στο πρόβλημα του βέλτιστου σχεδιασμού γραμμής προϊόντων.....	23
3.2.1 Greedy Heuristic	23
3.2.2 Divide-and-Conquer Heuristic.....	24
3.2.3 Product-Swapping Heuristic	24
3.2.4 Dynamic Programming Heuristic	24
3.2.5 Beam Search Heuristic.....	25
3.2.6 Nested Partitions Heuristic	25
3.2.7 Coordinate Ascent.....	26

3.2.8 Genetic Algorithm	26
3.2.9 Simulated Annealing.....	27
3.2.10 Particle Swarm Optimization.....	28
4. Ο αλγόριθμος διαφορικής εξέλιξης	29
4.1 Βασική μορφή διαφορικής εξέλιξης	29
Ψευδοκώδικας Αλγορίθμου Διαφορικής Εξέλιξης.....	34
4.2 Διαφοροποιήσεις στην εξίσωση μετάλλαξης.....	35
4.3 Βασικές Εφαρμογές Διαφορικής Εξέλιξης	36
4.4 Ερευνητικά ερωτήματα.....	38
5. Εφαρμογή Αλγορίθμου Διαφορικής Εξέλιξης στο πρόβλημα βελτιστοποίησης γραμμής προϊόντων.....	40
5.1 Περιγραφή προτεινόμενου μοντέλου.....	40
5.2 Προσαρμογή αλγορίθμου στο πρόβλημα	43
5.2.1 Αναπαράσταση λύσεων	43
5.3 Αποτελέσματα.....	44
Τοπική Αναζήτηση (Local Search).....	49
5.3 Δοκιμασία Ευρωστίας – Robustness Testing	50
5.4 Γενίκευση του Μοντέλου.....	52
6. Επίλογος.....	56
6.1 Παρατηρήσεις	56
6.2 Συμπέρασμα	57
Βιβλιογραφία	58

Περιεχόμενα Εικόνων

Εικόνα 1: Παρουσιάζοντας ένα απλό σχέδιο μετάλλαξης DE σε παραμετρικό χώρο 2-D	31
Εικόνα 2: Παράδειγμα διωνυμικής διασταύρωσης.....	32
Εικόνα 3: Παράδειγμα εκθετικής διασταύρωσης	33
Εικόνα 4: Διάφοροι πιθανοί φορείς δοκιμής σχηματίστηκαν λόγω ομοιομορφίας διωνυμικής διασταύρωσης μεταξύ του στόχου και των μεταλλαγμένων φορέων σε χώρο αναζήτησης 2-D	34
Εικόνα 5: Βήματα Αλγόριθμου Διαφορικής Εξέλιξης	35

Περιεχόμενα Πινάκων

Πίνακας 1: DE/rand/1/bin.....	45
Πίνακας 2:DE/best/1/bin.....	45
Πίνακας 3:DE/rand-to-best/1/bin.....	46
Πίνακας 4:DE/rand/2/bin.....	47
Πίνακας 5:DE/best/2/bin.....	48
Πίνακας 6:Εφαρμογή τοπικής αναζήτησης (local search) στα καλύτερα αποτελέσματα του DE/rand-to-best/1/bin	49
Πίνακας 7: Αποτελέσματα αλγορίθμων.....	50
Πίνακας 8: Αποτελέσματα αλγορίθμων.....	52
Πίνακας 9: Μεγέθη προβλήματος.....	53
Πίνακας 10: Αποτελέσματα Αλγορίθμων.....	54

Περίληψη

Στην παρούσα εργασία γίνεται εφαρμογή του αλγόριθμου της Διαφορικής Εξέλιξης όπου προτάθηκε από τους Storn και Price το 1995, στο πρόβλημα του βέλτιστου σχεδιασμού γραμμής προϊόντων. Σκοπός είναι ο σχεδιασμός μιας γραμμής προϊόντων των οποίων ο συνδυασμός τους θα ικανοποιεί τους στόχους που έχουν τεθεί από την συγκεκριμένη εταιρεία.

Κύριος στόχος είναι η βελτιστοποίηση του κέρδους της εταιρείας εισάγοντας τη γραμμή αυτή σε μια δεδομένη ανταγωνιστική αγορά. Τα προϊόντα αναλύονται με βάση τα επίπεδα των χαρακτηριστικών τους. Το παρών πρόβλημα κατατάσσεται στα NP-hard προβλήματα καθώς οι συνδυασμοί των χαρακτηριστικών που μπορεί να προκύψουν είναι τόσο πολλοί, ώστε κανείς αλγόριθμος να μην μπορεί να εγγυηθεί σε πολυωνυμικό χρόνο πως το βέλτιστο που προσδιορίζει είναι το ολικό βέλτιστο του προβλήματος.

Το πρόβλημα έχει ήδη επιλυθεί με διάφορες ευρετικές και μεθευρετικές μεθόδους στις οποίες θα γίνει αναφορά. Σκοπός μας είναι να ελέγξουμε την απόδοση του αλγόριθμου Διαφορικής Εξέλιξης στο παρών πρόβλημα.

1. Εισαγωγή

Οι επιχειρήσεις και οι διαδικασίες με τις οποίες λειτουργούν στις μέρες μας συνεχώς προσπαθούν να εξελίσσονται έτσι ώστε να προσαρμόζονται και να επιβιώνουν στη σύγχρονη αγορά. Οι πληθώρα καινοτόμων ιδεών και οι αυξανόμενες ανάγκες των καταναλωτών έχουν ως αποτέλεσμα στην συνεχή δημιουργία νέων προϊόντων που συμβάλλουν στον ανταγωνισμό μεταξύ των επιχειρήσεων, με τον καταναλωτή να ερευνά πριν να επιλέξει ένα προϊόν που θα αγοράσει έτσι ώστε να είναι σίγουρος ότι του ταιριάζει απόλυτα. Έτσι και η τεχνολογία αλληλένδετα έχει εισχωρήσει βαθιά μέσα στις επιχειρήσεις με υπολογιστικά συστήματα που τις βοηθάνε να πάρουνε σημαντικές αποφάσεις στο να ανταπεξέλθουν σ' αυτό το περιβάλλον. Οι επιχειρήσεις που καθυστερούν να ανταποκριθούν ή λαμβάνουν λάθος αποφάσεις με την εναπόθεση κεφαλαίων στην ανάπτυξη προϊόντων που στο τέλος δεν θα αποφέρουν τα επιθυμητά κέρδη ,λόγω του υψηλού ανταγωνισμού κινδυνεύουν να μην επιβιώσουν. Οι μάνατζερ επιχειρήσεων και τα στελέχη Μάρκετινγκ καλούνται να κάνουν αναλύσεις με το τι διαμορφώνει την καταναλωτική προτίμηση των πελατών τους έτσι ώστε να εκτιμήσουν στο πρώιμο στάδιο σχεδιασμού του προϊόντος πως θα αντιδράσει η αγορά με το νέο αυτό προϊόν και αν θα αποφέρει τελικά τα επιθυμητά κέρδη. Αυτό δεν είναι εύκολο λόγω των πολλών περιορισμών που δημιουργούνται οι οποίοι μπορεί να είναι οικονομικοί, χωρικοί , χρονικοί , νομοθετικοί κ.α. Επίσης, δημιουργούνται πολύπλοκα συστήματα με αμέτρητες υποψήφιες λύσεις που ο ανθρώπινος νους δεν μπορεί να σκεφτεί σε ποια λύση θα καταλήξει τελικά η οποία πρέπει να είναι βέλτιστη. Προκειμένου λοιπόν να αναλυθούν και να επιλυθούν τέτοια συστήματα κρίνεται αναγκαία η ανάπτυξη ευρετικών και μεθευρετικών μεθόδων που θα «εξοπλίσουν» τα σύγχρονα πληροφοριακά συστήματα στο «κυνήγι» των βέλτιστων λύσεων.

1.1. Δομή της Εργασίας

Η παρούσα εργασία προσπαθεί να εφαρμόσει ένα εναλλακτικό τρόπο επίλυσης στο πρόβλημα βέλτιστου σχεδιασμού γραμμής προϊόντων. Αρχικά, περιγράφεται η διαδικασία έρευνας των καταναλωτικών προτιμήσεων καθώς και το συγκεκριμένο πρόβλημα. Στη συνέχεια ιδιαίτερα σημαντικά είναι τα κεφάλαια 3 και 4 όπου εισάγουμε το θεωρητικό υπόβαθρο όσο αφορά τις ευρετικές και μεθευρετικές μεθόδους και ειδικότερα του αλγορίθμου βελτιστοποίησης Διαφορικής Εξέλιξης. Θα γίνει περιγραφή των διαφόρων μεθόδων που έχουν δοκιμαστεί για να επιλύσουν το παρόν πρόβλημα . Τέλος στο κεφάλαιο

5 γίνεται ολοκληρωμένη παρουσίαση του προτεινόμενου μοντέλου για την επίλυση του εν λόγω προβλήματος καθώς και τα αποτελέσματα του αλγορίθμου μας. Επίσης θα γίνει σύγκριση αποτελεσμάτων με τις υφιστάμενες μεθόδους που έχουν δοκιμαστεί ήδη.

1.2. Το Πρόβλημα Βέλτιστου Σχεδιασμού γραμμής προϊόντων

Το πρόβλημα βέλτιστου σχεδιασμού γραμμής προϊόντος αποτελεί αντικείμενο μελέτης εδώ και πάρα πολλά χρόνια. Πρώτη φορά παρουσιάστηκε το πρόβλημα βέλτιστου σχεδιασμού ενός προϊόντος από τον **Zufryden** το 1977. Αργότερα το 1985 οι **Green and Krieger** παρουσίασαν το πρόβλημα βέλτιστου σχεδιασμού γραμμής προϊόντος όπου είναι αντικείμενο μελέτης μέχρι και σήμερα.

Το πρόβλημα βέλτιστου σχεδιασμού γραμμής νέων προϊόντων κατατάσσεται στην κατηγορία των NP – hard προβλημάτων, καθώς κανένας αλγόριθμος δεν μπορεί να πιστοποιήσει σε πολυωνυμικό χρόνο, ότι το βέλτιστο που προσδιορίζει είναι το ολικό βέλτιστο του προβλήματος. Πληθώρα μεθόδων βελτιστοποίησης έχουν εφαρμοστεί, με σκοπό τον προσδιορισμό όσο γίνεται καλύτερων βέλτιστων σε διαχειρίσιμο χρόνο. Μερικές μέθοδοι που αναπτύχθηκαν ενσωματώθηκαν σε ευφυή συστήματα μάρκετινγκ ώστε να αποτελέσουν εργαλείο για τους αναλυτές που αντιμετώπιζαν το πρόβλημα.

Για την επίλυση του εν λόγω προβλήματος είναι απαραίτητη η μέτρηση των προτιμήσεων των καταναλωτών έτσι ώστε η επιχείρηση να γνωρίζει τι προϊόν απαιτείται από την αγορά αποφεύγοντας εν μέρει την αποτυχία του μετά την διείσδυσή του στην αγορά. Για να γίνει η μέτρηση των καταναλωτικών αναγκών διεξάγεται έρευνα αγοράς όπου γίνεται μοιράζοντας ερωτηματολόγια σε μελλοντικούς πελάτες ή με προσωπικές συνεντεύξεις. Κάθε προϊόν αποτελείται από χαρακτηριστικά (attributes) και το κάθε χαρακτηριστικό αποτελείται από επίπεδα(levels). Έτσι κάθε καταναλωτής, καλείται να αξιολογήσει ένα αριθμό προϊόντων, όπου θα δηλώνει την προτίμηση που έχει για κάθε επίπεδο του κάθε χαρακτηριστικού του προϊόντος (partworths). Στη συνέχεια με εφαρμογή της μεθόδου της συζυγούς ανάλυσης (conjointanalysis) αντιστοιχίζεται σε κάθε επίπεδο κάθε χαρακτηριστικού η μερική αξία (partworth) που προσδίδει στον κάθε καταναλωτή. Τέλος αθροίζοντας τις μερικές αξίες βρίσκουμε την συνολική αξία (utility) κάθε προϊόντος για κάθε καταναλωτή.

Με αυτό τον τρόπο οι επιχειρήσεις, χρησιμοποιώντας ως δεδομένο τις προτιμήσεις των καταναλωτών προχωρούν στην σχεδίαση γραμμής προϊόντων η οποία θα ανταποκρίνεται στις ανάγκες των πελατών. Οι επιχειρήσεις τότε εκπληρώνουν τους στόχους τους οι οποίοι είναι η μεγιστοποίηση του μεριδίου αγοράς (market share maximization) ή η μεγιστοποίηση των κερδών της (profit maximization).

2. Έρευνα αγοράς

Οι επιστήμονες στον χώρο του μάρκετινγκ αντιμετωπίζουν εδώ και πολλά χρόνια να καταλάβουν με ποιο τρόπο σκέφτεται ένας καταναλωτής και πως διαμορφώνεται η τελική του επιλογή. Γι' αυτό το πρόβλημα δεν υπάρχει κάποια σαφής απάντηση αφού κάθε καταναλωτής δρα με τον δικό του τρόπο και παίρνει αποφάσεις που επηρεάζονται από την λογική του και τα συναισθήματά του. Γι' αυτό το λόγο έχουν αναπτυχθεί μοντέλα που παρουσιάζουν πιο ρεαλιστικά τη διαδικασία λήψης αποφάσεων των καταναλωτών.

2.1 Μέτρηση καταναλωτικών προτιμήσεων

Οι περισσότερες εταιρείες δρουν με γνώμονα τις προτιμήσεις των καταναλωτών κάνοντας δοκιμές στην αγορά. Το κοινό που απευθύνονται έρχεται αντιμέτωπο με ερωτήματα σχετικά με το προσωπικό τους γούστο και καινούρια προϊόντα που προορίζονται να βγούν στην αγορά. Η παραπάνω ενέργεια δεν συμφέρει τις εταιρείες σε χρόνο και κόστος καθώς εμβαθύνει αρκετά σε κάθε προϊόν ξεχωριστά και εξετάζει όλες τις πιθανές παραλλαγές που μπορεί να υποστεί. Ακόμη οι δοκιμές αυτές υστερούν από αξιοπιστία μιας και βασίζονται εξ' ολοκλήρου σε προσεγγίσεις του κοινού (π.χ κατάταξη προϊόντος με βάση την μάρκα) που δεν κάνουν κάποια αναφορά στο κόστος και δεν αποφέρουν αρκετά στοιχεία για την ελαχιστοποίηση αυτού.

Ωστόσο η ανάλυση συζυγιών (Conjoint Analysis) χρησιμοποιεί τα καλύτερα στοιχεία από τις παραπάνω προσεγγίσεις σε μια έρευνα που θέτει ως στόχο την μείωση του κόστους. Αναλυτικότερα η Conjoint Analysis επεξεργάζεται τα δεδομένα που έχουν συλλεχθεί και με την χρήση μεθόδων παλινδρόμησης αναλογεί σε όλα τα επίπεδα για κάθε χαρακτηριστικό μια τιμή η οποία αναγράφεται ως η μερική αξία του (part worth).

2.1.1 Ανάλυση Συζυγιών (Conjoint Analysis)

Η ανάλυση συζυγιών (Conjoint Analysis) αναπτύχθηκε από την αρχική έρευνα του ψυχολόγου Luce και του στατιστικολόγου Tukey (1964) και η ίδια αποτελεί τη προέλευση των τεχνικών των Green και Srinivasan (1978) και των Carroll και Green (1995). Σειρά ερευνητών μελέτησε το θεωρητικό υπόβαθρο των παραπάνω μελετών και ανέπτυξαν μια ποικιλία μη μετρικών μοντέλων για τον υπολογισμό της μερικής αξίας από τις προτιμήσεις

των ερωτηθέντων σε όλα τα πολλαπλά ερωτήματα που τους τέθηκαν, όπως για παράδειγμα οι περιγραφές των προϊόντων και των υπηρεσιών. Σύμφωνα με έρευνες από τους Wittink και Cattin (1989) και τους Wittink, Vriens και Burhenne (1994) η Conjoint Analysis χρησιμοποιείται καθολικά στην έρευνα μάρκετινγκ για την ανάλυση των συμβιβασμών (trade off) που πραγματοποιεί το καταναλωτικό κοινό.

Η ανάλυση συζυγιών (Conjoint Analysis) είναι στατιστική από-συνθετική μέθοδος, η οποία περιέχει πολλές μεταβλητές και στοχεύει στην μοντελοποίηση της ανθρώπινης συμπεριφοράς, πιο συγκεκριμένα στη αγοραστική τους συμπεριφορά με σκοπό τον υπολογισμό των καταναλωτικών τους προτιμήσεων. Επιπλέον, είναι εύλογο να αναφερθούμε στον τρόπο με τον οποίο η συγκεκριμένη μέθοδος είναι η πιο αναπτυσσόμενη και μία από τις πιο ευρέως χρησιμοποιούμενες μεθόδους σήμερα. Ένα αίτιο που καταφέρνει τα παραπάνω επιτεύγματα είναι ότι παρέχει χρήσιμα συμπεράσματα, τα οποία αν παρουσιαστούν με τον κατάλληλο τρόπο τα διευθυντικά στελέχη μπορούν εύκολα να τα κατανοήσουν και να δράσουν αναλόγως. Αναλυτικότερα, αυτή καθίσταται αρκετά δημοφιλής, δεδομένου ότι υπεύθυνοι του τομέα μάρκετινγκ, ερευνητές αλλά και αναλυτές τη χρησιμοποιούν για να καθορίσουν τα χαρακτηριστικά που θα πρέπει να έχει ένα νέο προϊόν ή ένα υφιστάμενο που χρήζει βελτίωσης και πως αυτό θα πρέπει να κοστολογηθεί.

Βασικές υποθέσεις και παραδοχές της ανάλυσης συζυγιών (Conjoint Analysis):

- **Η πρώτη βασική** υπόθεση που γίνεται από την εν λόγω τεχνική ανάλυση είναι ότι το εκάστοτε προϊόν ή η εκάστοτε υπηρεσία μπορεί να αναλυθεί σε ένα σύνολο χαρακτηριστικών, με κάθε χαρακτηριστικό να λαμβάνει διάφορες τιμές, οι οποίες ονομάζονται επίπεδα.
- **Η δεύτερη υπόθεση** είναι ότι οι καταναλωτές επιλέγουν το προϊόν, που τους μεγιστοποιεί τη χρησιμότητά τους, δηλαδή την αντιλαμβανόμενη αξία ή αλλιώς την υποκειμενική προτιμησιακή κρίση του ατόμου που εκφράζει τη συνολική αξία, η οποία προκύπτει από τα χαρακτηριστικά του προϊόντος και από την κατανάλωσή του με σκοπό την ικανοποίηση συγκεκριμένης ανάγκης.
- **Επιπρόσθετα**, η ανάλυση συζυγιών (Conjoint Analysis) υποθέτει πως οι καταναλωτές αξιολογούν τη χρησιμότητα ενός προϊόντος ή μιας υπηρεσίας συνδυάζοντας τις επιμέρους αξίες (part worths), οι οποίες προκύπτουν από τις τιμές των επιμέρους χαρακτηριστικών του προϊόντος. Σε αυτό το σημείο είναι σωστό να

τονισθεί πως η σημαντικότητα του εκάστοτε χαρακτηριστικού , πηγάζει από αυτές τις μερικές αξίες.

Είναι εύκολο να αντιληφθεί κανείς ότι η εν λόγω μέθοδος στηρίζεται στην κρίση των καταναλωτών, και πιο συγκεκριμένα στην υπόθεση ότι αυτοί αξιολογούν την «αξία», ενός προϊόντος, συνδυάζοντας την μερική αξία (part worth) που έχει κάθε ιδιότητα του συγκεκριμένου αντικειμένου. Μια υψηλή αξιολόγηση ενός αντικειμένου από τους ερωτώμενους, συνεπάγει ότι αυτό έχει και μια υψηλή καταναλωτική αξία ή «χρησιμότητα»(όρος που θα εξηγηθεί παρακάτω). Η ανάλυση συζυγιών (Conjoint Analysis), επιτρέπει τον προσδιορισμό της σχετικής αξίας / σημασίας κάθε ιδιότητας, υπολογίζοντας την χρησιμότητα της μερικής αξίας (part worth utility), ενός ατόμου για το κάθε επίπεδο των ιδιοτήτων που εξετάζονται. Έχοντας προσδιορίσει τις μερικές αξίες στη συνέχεια μπορεί να υπολογιστεί η συνολική αξία (total utility), που αποδίδουν οι ερωτώμενοι, σε κάθε συνδυασμό ιδιοτήτων. Η «χρησιμότητα» (utility), αποτελεί την εννοιολογική βάση για τον προσδιορισμό της αξίας που έχει ένα αντικείμενο, κάνοντας χρήση της ανάλυσης συζυγιών (Conjoint Analysis). Οι βαθμολογίες ή οι κατατάξεις, που συλλέγονται από τους ερωτηθέντες, χρησιμοποιούνται ως εξαρτημένες μεταβλητές, οι ιδιότητες του αντικειμένου ως ανεξάρτητες μεταβλητές και οι μερικές χρησιμότητες προσδιορίζονται στατιστικά.

2.1.2 Διαδικασία της Ανάλυσης Συζυγιών (Conjoint Analysis)

Τα τρία στάδια από τα οποία αποτελείται η ανάλυση συζυγιών είναι:

- i. Το αρχικό στάδιο είναι ο **σχεδιασμός της έρευνας**, όπου αποτελείται από την επιλογή των χαρακτηριστικών ανάλογα με την κατηγορία των προϊόντων, την επιλογή των επιπέδων για κάθε χαρακτηριστικό και την δημιουργία των προφίλ των προϊόντων προς αξιολόγηση.
- ii. Το δεύτερο στάδιο είναι η **συλλογή των δεδομένων από τους καταναλωτές**, όπου περιλαμβάνει τον σχεδιασμό της διαδικασίας συλλογής των δεδομένων και την επιλογή των μεθόδων αξιολόγησης των προφίλ.
- iii. Το τελευταίο στάδιο είναι η **αξιοποίηση των αποτελεσμάτων** που έχουν συλλεχθεί από τα δύο προηγούμενα στάδια. Αυτό το στάδιο αποτελείται από την τμηματοποίηση των καταναλωτών με βάση τους βαθμούς αξίας που έχουν

καταναίμει στα διάφορα χαρακτηριστικά, την προσομοίωση της αγοράς και τη βελτιστοποίηση του προϊόντος.

2.1.3 Η συλλογή των δεδομένων

Στο πρόβλημα βέλτιστου σχεδιασμού γραμμής προϊόντων καλύπτονται όλα τα παραπάνω κριτήρια για την εφαρμογή της ανάλυσης συζυγίων και την εξαγωγή των part-worths των εν δυνάμει αγοραστών. Τα σακίδια αναλύονται σε 10 χαρακτηριστικά, καθένα από αυτά έχει δύο διακριτά επίπεδα πέραν του χαρακτηριστικού τιμή που έχει 7 διακριτά επίπεδα. Μία μεταβολή βελτιώνει κάποιο χαρακτηριστικό χωρίς όμως να παρεμβαίνει στη θέση κάποιου άλλου. Το παραπάνω αποτελεί ιδιότητα χαρακτηριστικών που είναι αποτελεσματικά κατά Pareto. Η επίδραση των χαρακτηριστικών φαίνεται μόνο στη συνολική ποιότητα του προϊόντος, δηλαδή δεν υπάρχουν χαρακτηριστικά που δρουν ομαδικά ή που η ύπαρξη του ενός αποκλείει την ύπαρξη κάποιου άλλου. Κάτι που είναι λογικό μιας και τα προϊόντα δεν έχουν κυκλοφορήσει στην αγορά, οι καταναλωτές δεν έχουν κάποια ανάλογη εμπειρία χρήσης ώστε να επηρεαστεί η γνώμη τους.

Το δείγμα καταναλωτών υπόκειται να εκπληρώσει τις παρακάτω εργασίες :

Self –Explicated Questions δηλαδή **ερωτήσεις αυτοπροσδιορισμού** που ο ερωτώμενος καλείται να προσδιορίσει σε κλίμακα 1 έως 4 πόσο σημαντικό είναι για αυτόν κάθε ένα από τα χαρακτηριστικά.

Paired –Comparison Questions δηλαδή τη **σύγκριση κατά ζεύγη** σε 16 ανεξάρτητα σακίδια που επιλέγονται με τυχαίο τρόπο, και είναι απαντήσεις ανεξάρτητες με τις προηγούμενες. Ο ερωτώμενος καλείται να δηλώσει την πιθανότητα να επιλέξει ένα από τα σακίδια.

Purchase Intention ή **πρόθεση αγοράς**. Οι ερωτώμενοι καλούνται να προσδιορίσουν πόσο πιθανό είναι να αγοράσουν κάθε ένα από τα 6 συγκεκριμένα σακίδια.

Filler task που ο πιθανός πελάτης υπόκειται στην αξιολόγηση των ερωτήσεων τις έρευνας και κατά πόσο αυτές τον ικανοποιούν. Στόχος του σταδίου αυτού είναι να αποφορτιστεί η μνήμη του σχετικά με τις έως τώρα απαντήσεις του και να περάσει στο τελικό στάδιο επιλογής.

Τέλος το **Final Bag Selection** αφορά την **τελική επιλογή τσάντας**. Μοιράζονται σε κάθε υποψήφιο 100\$ με την προϋπόθεση ότι θα εισπράξουν τη διαφορά της τιμής αν επιλέξουν ένα φθινό σακίδιο. Πρέπει να επιλέξουν μεταξύ 5 σακιδίων που επιλέγονται με τυχαίο τρόπο από ένα συγκεκριμένο σχεδιασμό 16 αντιπροσωπευτικών σακιδίων. Ισχύει η ανεξαρτησία δεδομένων και σ' αυτό το στάδιο δηλαδή τα 5 αυτά σακίδια δεν σχετίζονται με τις προηγούμενες απαντήσεις. Τέλος ο ερωτούμενος καλείται να κατατάξει και τις επόμενες του επιλογές σε περίπτωση που κάποια από τα σακίδια δεν είναι διαθέσιμο.

Το δείγμα πιθανών πελατών που συμμετείχε στην έρευνα αποτελούνταν από πρωτοετής φοιτητές και δεν δίνονταν αρκετές πληροφορίες όσον αφορά τον σκοπό της έρευνας. Η έρευνα πραγματοποιήθηκε σε μια web εφαρμογή με χρήσιμες επεξηγήσεις για την εύκολη ολοκλήρωσή της, συλλέχθηκαν 330 ολοκληρωμένες απαντήσεις και επεξεργάστηκαν σε εξειδικευμένο λογισμικό. Αφού υποβλήθηκε το ερωτηματολόγιο ειδικοί εξασφάλισαν την αλληλουχία των ερωτήσεων με τρόπο τέτοιο ώστε να μην επηρεάζεται το τελικό αποτέλεσμα. Επίσης έλεγξαν ότι οι ερωτήσεις είναι σαφείς και απόλυτα κατανοητές.

2.2 Μοντελοποίηση της Απόφασης

Η μοντελοποίηση της απόφασης ορίζεται ως η διαδικασία κατά την οποία προσομοιώνεται η συμπεριφορά του καταναλωτή, ο οποίος καλείται να διαλέξει ανάμεσα από διάφορες εναλλακτικές λύσεις. Η διαδικασία πραγματοποιείται χρησιμοποιώντας ένα μοντέλο επιλογής (choice model). Ο καταναλωτής καλείται να επιλέξει ένα προϊόν, μέσα από μια σειρά ανταγωνιστικών προϊόντων ενσωματώνοντας παράλληλα πληροφορίες που θα τον βοηθήσουν στην επιλογή του.

Το μοντέλο επιλογής είναι ένα μαθηματικό μοντέλο που αναπαριστά το προφίλ του καταναλωτή δημιουργώντας μια σχέση μεταξύ προτίμησης και απόφασης. Στην ουσία μετατρέπει τις χρησιμότητες των προϊόντων που ένας καταναλωτής ορίζει έχοντας εξετάσει τις εναλλακτικές, σε πιθανότητες επιλογής για κάθε εναλλακτική. Οι κανόνες επιλογής διακρίνονται σε ντετερμινιστικούς (deterministic choice models) είτε πιθανολογικούς (probabilistic choice models).

Ο κανόνας πρώτης επιλογής (first choice) ή μέγιστης χρησιμότητας (maximum utility) αποτελεί ντετερμινιστικό μοντέλο και υποθέτει ότι η επιλογή προϊόντος του καταναλωτή θα είναι πάντα αυτό με τη υψηλότερη χρησιμότητα. Στην εναλλακτική με τη υψηλότερη

χρησιμότητα δίνεται πιθανότητα επιλογής 1, ενώ οι υπόλοιπες εναλλακτικές έχουν πιθανότητα 0. Το μοντέλο πρώτης επιλογής υστερεί σημαντικά στο ότι μας πληροφορεί μόνο για το προϊόν με τη υψηλότερη αξία χωρίς να παραθέτει πληροφορίες για τις σχετικές αξίες που δόθηκαν από τους πιθανούς πελάτες στα εναλλακτικά προϊόντα. Αυτό φέρει ως συνέπεια την αλλοίωση του τελικού αποτελέσματος.

Είναι κοινώς γνωστό, ότι ο καταναλωτής επιλέγει ένα προϊόν με μια σύνθετη διαδικασία που αποτελείται από τυχαίες και μη προβλέψιμες παραμέτρους. Η επιλογή του καταναλωτή μπορεί να επηρεαστεί από αστάθμητους παράγοντες όπως για παράδειγμα τη σύγχυση, το μεγάλο κόστος αναζήτησης, το εύρος επικογών κ.α. Όλοι αυτοί είναι παράγοντες που είναι δυνατόν να οδηγήσουν τον καταναλωτή στην λήψη λάθους απόφασης.

Λύση στο παραπάνω πρόβλημα έφεραν τα πιθανολογικά μοντέλα (στοχαστικά) που παρουσιάζουν ρεαλιστικά τη διαδικασία με την οποία ο καταναλωτής αποφασίζει και αντί να προσδίδουν ολόκληρη την πιθανότητα επιλογής σε ένα προϊόν, συσχετίζουν την τιμή της πιθανότητας του προϊόντος με την τιμή της χρησιμότητάς του (utility). Τα πιθανολογικά (στοχαστικά) μοντέλα επιλογής συνδυάζουν όλες τις αξίες και προσδίδουν μια πιθανότητα επιλογής ακόμη και στο προϊόν με τη χαμηλότερη αξία. Τα πιθανολογικά μοντέλα επιλογής χωρίζονται στα μοντέλα σταθερής αξίας(constant utility models) και στα μοντέλα τυχαίας αξίας(random utility models) .

Τα μοντέλα σταθερής αξίας, θεωρούν ότι όλες οι αξίες των προϊόντων είναι σταθερές και προσομοιώνουν τη στοχαστική φύση της ανθρώπινης συμπεριφοράς, εισάγοντας ένα επίπεδο αβεβαιότητας στον κανόνα της απόφασης. Το δημοφιλέστερο πιθανολογικό μοντέλο σταθερής αξίας είναι το **BTL** το οποίο υπολογίζεται ως :

$$p_{ij} = \frac{U_{ij}}{\sum_{j=1}^n U_{ij}} \quad (2.1)$$

Όπου p_{ij} , είναι η πιθανότητα ο καταναλωτής i να επιλέξει το προϊόν j , U_{ij} είναι η συνολική αξία που δίνει ο καταναλωτής στο προϊόν και n είναι ο αριθμός των ανταγωνιστικών προϊόντων.

Ο **Pessemier** (1971) έχει αναπτύξει περαιτέρω το BTL δημιουργώντας ένα καινούργιο μοντέλο. Οι αξίες των προϊόντων μετασχηματίζονται με έναν εκθέτη ο οποίος ελέγχει τις τιμές των πιθανοτήτων επιλογής, διατηρώντας την αυθεντική κατάταξη των προτιμήσεων. Έτσι το μοντέλο γίνεται:

$$p_{ij} = \frac{U_{ij}^{\alpha}}{\sum_{j=1}^n U_{ij}^{\alpha}} \quad (2.2)$$

Όπου α , προσδιορίζεται από τον ερευνητή και μπορεί να πάρει διάφορες τιμές. Για μικρές τιμές του εκθέτη α , υπερεκτιμώνται οι πιθανότητες επιλογής για προϊόντα με χαμηλή αξία. Αντίστοιχα όσο το α πλησιάζει στο άπειρο, το μοντέλο τείνει να μετατραπεί σε μοντέλο πρώτης επιλογής.

Τα μοντέλα τυχαίας αξίας σε αντίθεση με τα μοντέλα σταθερής αξίας, θεωρούν ότι ο καταναλωτής πάντα επιλέγει την εναλλακτική με τη μεγαλύτερη αξία (U) η οποία προσδιορίζεται από το ντετερμινιστικό και το στοχαστικό μέρος.

Το ντετερμινιστικό κομμάτι (V) που προσδιορίζεται από τη συνάρτηση που υπολογίζει τις υποκειμενικές αξίες και το στοχαστικό κομμάτι (e) που αναπαριστά τη μη υπολογίσιμη απόκλιση στις αξίες (Baltas and Doyle, 2001): $U = V + e$.

Το μοντέλο MNL θεωρεί ανεξάρτητα πανομοιότυπα σφάλματα (στοχαστικό κομμάτι) στον πληθυσμό των καταναλωτών σύμφωνα με τη διπλή εκθετική κατανομή. Με βάση αυτή τη θεώρηση και την αρχή μεγιστοποίησης της αξίας, η πιθανότητα ο πελάτης i να επιλέξει το προϊόν j δίδεται από τον τύπο:

$$p_{ij} = \frac{e_{ij}^U}{\sum_{j=1}^n e_{ij}^U} \quad (2.3)$$

2.3 Κριτήριο Βελτιστοποίησης

Οι Shoker and Srinivasan ανέφεραν το πρώτο κριτήριο βελτιστοποίησης αυτό της μεγιστοποίησης του μεριδίου αγοράς (share of market). Μετέπειτα εισήχθησαν ακόμα δύο

κριτήρια βελτιστοποίησης, το 'κέρδος του πωλητή'(seller's welfare) από τους Green και το 'κέρδος του αγοραστή'(buyer's welfare) το 1988 από τους Green and Krieger. Όσο αφορά το κριτήριο του κέρδους του αγοραστή υποθέτουμε ότι δεν υπάρχει καθόλου ανταγωνισμός και στόχος είναι η μεγιστοποίηση του αθροίσματος των αξιών που προσφέρουν τα υποψήφια προϊόντα στους καταναλωτές. Αυτό χρησιμοποιείται κυρίως για τη βελτιστοποίηση προϊόντων και υπηρεσιών που προσφέρονται από δημόσιους οργανισμούς. Το κριτήριο του κέρδους του πωλητή βελτιστοποιεί το κέρδος της εταιρείας. Το συγκεκριμένο κριτήριο πρέπει να λάβει υπόψη στην αντικειμενική συνάρτηση την οριακή απόδοση που προσδοκεί η εταιρεία από κάθε επίπεδο του χαρακτηριστικού.

2.4 Μοντελοποίηση του προβλήματος

Η υπό μελέτη έρευνα όπως έχουμε αναφέρει ανήκει στην κατηγορία NP-hard προβλημάτων των οποίων ο εντοπισμός του πραγματικού ολικού βέλτιστου σε πραγματικό χρόνο είναι αδύνατος. Όταν ο αριθμός χαρακτηριστικών και επιπέδων μεγαλώνει, σε πραγματικές συνθήκες, αυξάνεται απότομα και το πλήθος των διαφορετικών προϊόντων που είναι πιθανό να προκύψουν. Αυτό φέρει ως αποτέλεσμα να περιπλέκεται η επιλογή ενός κατάλληλου συνδυασμού χαρακτηριστικών και επιπέδων. Παραδείγματος χάρι ένα πρόβλημα με 5 χαρακτηριστικά που το καθένα αποτελείται από 8 επίπεδα ανέρχεται σε 32768 προϊόντα.

Κατά τους Green and Krieger(1988), για τη μοντελοποίηση του προβλήματος πρωτεύουσα σημασία έχει ο προσδιορισμός των πιθανών εναλλακτικών. Στη συνέχεια προσδιορίζονται οι παράμετροι που βελτιστοποιούν το πρόβλημα ανάλογα με την εκάστοτε μέθοδο που επιλέχθηκε για την λύση του. Δημιουργούνται ερωτήματα σχετικά με τον αριθμό των προϊόντων που συνθέτουν τις εναλλακτικές επιλογές και το τρόπο παράθεσης αυτών ώστε να περιλαμβάνονται όλες οι δυνητικά καλές λύσεις. Αυτό πλέον είναι δυνατό αφού στην εποχή που ζούμε χάρι στην ραγδαία ανάπτυξη υπολογιστών υψηλών ταχυτήτων και την ανάπτυξη και χρήση αλγόριθμων μεγαλύτερου επιπέδου, μπορούν σχεδιασθούν προϊόντα απευθείας από τις μερικές αξίες των επιπέδων τους.

Η μοντελοποίηση του προβλήματος εξαρτάται από το μοντέλο επιλογής που θα χρησιμοποιήσουμε καθώς και από το κριτήριο βελτιστοποίησης. Η πιο συνηθισμένη προσέγγιση είναι η επίλυση του προβλήματος με βελτιστοποίηση του μεριδίου αγοράς και χρήση του μοντέλου πρώτης επιλογής.

2.5 Μοντελοποίηση του υπό μελέτη προβλήματος

Στη συγκεκριμένη εργασία στην οποία χρησιμοποιήθηκαν υφιστάμενα δεδομένα από τους Belloni et al.(2008) , σκοπός είναι να επιλύσουμε ένα πραγματικό πρόβλημα τσαντών laptop. Αναλυτικότερα, καθιστούμε στο βέλτιστο σχεδιασμό γραμμής 5 τσαντών, έναντι μιας δεδομένης σταθερής ανταγωνιστικής γραμμής 3 τσαντών. Δεδομένο της έρευνας αποτελούν οι προτιμήσεις 324 καταναλωτών και ερχόμαστε αντιμέτωποι με την βελτιστοποίηση του συνολικού κέρδους που θα εισπράξει η εταιρεία. . Τα σακίδια αναλύονται σε 10 χαρακτηριστικά ,καθένα από αυτά έχει δύο διακριτά επίπεδα (χερούλι, εναλλακτικό χρώμα, θήκη PDA κτλ.) που έχουν ως απάντηση δυο τιμές (ναι / όχι) πέραν του χαρακτηριστικού τιμή που έχει 7 διακριτά επίπεδα (70\$ 75\$ 80\$ 85\$ 90\$ 95\$ 100\$) Χρήζει προσοχής ότι ο συνδυασμός των χαρακτηριστικών μπορεί να δημιουργήσει 3584 διαφορετικά προϊόντα από τα οποία προκύπτουν $4,9 \times 10^{15}$ διαφορετικές γραμμές από 5 προϊόντα. Για κάθε χαρακτηριστικό, για κάθε καταναλωτή, έχει υπολογισθεί από τους ερευνητές η οριακή μερική χρησιμότητα που θα προσθέσει ή θα αφαιρέσει η προσθήκη ή έλλειψη του χαρακτηριστικού καθώς και τα πιθανά κέρδη που έχει η εταιρεία από κάθε προϊόν. Ως συνέπεια αρκετοί από τους αλγορίθμους που έχουν εφαρμοστεί στο υπό μελέτη πρόβλημα εφαρμόζονται μόνο σε προβλήματα στα οποία μπορεί κανείς να απαριθμήσει γρήγορα όλα τα πιθανά προϊόντα. Για το χαρακτηριστικό τιμή, η οριακή μερική χρησιμότητα εκφράζει τη μεταβολή χρησιμότητας που θα επέλθει για κάθε διαδοχική αύξηση κατά 5\$ (συνήθως η αύξηση της τιμής προκαλεί δυσαρέσκεια άρα μείωση της μερικής χρησιμότητας). Για τα υπόλοιπα χαρακτηριστικά, έχουμε ως δεδομένο το επιπλέον κόστος που προσδίδει στην εταιρεία η πρόσθεση συγκεκριμένου χαρακτηριστικού. Επίσης μέσω δεδομένων συζυγούς ανάλυσης που έχουν εστιάσει στην τιμή ξέρουμε ότι το σταθερό κόστος κατασκευής του προϊόντος χωρίς κάποιο επιπλέον χαρακτηριστικό είναι 35\$.

Το συγκεκριμένο μοντέλο θεωρεί ότι η επιλογή του πελάτη είναι το προϊόν που του αποφέρει τη μεγαλύτερη χρησιμότητα , καθώς σκοπός της έρευνας είναι η μεγιστοποίηση του κέρδους της εταιρείας. Στην παρούσα εργασία έχει γίνει χρήση ντετερμινιστικών μοντέλων επιλογής με κριτήριο την βελτιστοποίηση του κέρδους του πωλητή. Τα δεδομένα αποτελούνται από την εκτίμηση των προτιμήσεων (part-worth) κάθε καταναλωτή ξεχωριστά για κάθε χαρακτηριστικό, και επίσης μια εκτίμηση του κόστους κατασκευής που επιφέρει κάθε χαρακτηριστικό στην εταιρεία.

Τα τρία ανταγωνιστικά προϊόντα που θεωρήθηκαν από τους ερευνητές έχουν σχεδιαστεί αυθαίρετα και περιλαμβάνουν : ένα προϊόν με όλα τα εννιά χαρακτηριστικά στην τιμή των 100\$, ένα προϊόν με πέντε χαρακτηριστικά στην τιμή των 85\$ και ένα προϊόν με κανένα από τα χαρακτηριστικά στην τιμή των 70\$. Αν κάποιο από τα προϊόντα της γραμμής που δημιουργείται δεν ξεπερνά σε χρησιμότητα ένα προϊόν του ανταγωνιστή, τότε ο καταναλωτής θεωρείται ότι διαλέγει το ανταγωνιστικό προϊόν.

3. Προσεγγιστικοί Αλγόριθμοι

Στην επιστήμη των υπολογιστών και στην επιχειρησιακή έρευνα, οι προσεγγιστικοί αλγόριθμοι είναι αποτελεσματικοί αλγόριθμοι που βρίσκουν προσεγγιστικές λύσεις στα NP-hard προβλήματα βελτιστοποίησης με αποδεδειγμένες εγγυήσεις για την απόσταση της επιστρεφόμενης λύσης ως προς τη βέλτιστη. Οι προσεγγιστικοί αλγόριθμοι προέκυψαν από τον τομέα της θεωρητικής επιστήμης των υπολογιστών ως συνέπεια της ευρέως θεωρητικής εικασίας $P \neq NP$. Υπό αυτή την υπόθεση, μια μεγάλη τάξη προβλημάτων βελτιστοποίησης δεν μπορεί να λυθεί ακριβώς σε πολυωνυμικό χρόνο. Ο τομέας των προσεγγιστικών αλγορίθμων, ως εκ τούτου, προσπαθεί να καταλάβει πόσο στενά είναι δυνατόν να προσεγγίσουν βέλτιστες λύσεις σε τέτοια προβλήματα σε πολυωνυμικό χρόνο. Σε μια συντριπτική πλειοψηφία των περιπτώσεων, η εγγύηση τέτοιων αλγορίθμων είναι πολλαπλασιαστική που εκφράζεται ως λόγος προσέγγισης ή συντελεστής προσέγγισης, δηλαδή η βέλτιστη λύση είναι εγγυημένη πάντα εντός ενός (προκαθορισμένου) πολλαπλασιαστικού παράγοντα της επιστρεφόμενης λύσης.

Σκοπός είναι η μεγιστοποίηση (ή ελαχιστοποίηση) ενός στόχου δηλαδή μιας αντικειμενικής συνάρτησης που εκφράζει για παράδειγμα το κέρδος (μεγιστοποίηση), το κόστος (ελαχιστοποίηση), το χώρο που θα καταλάβουν τα μηχανήματα (ελαχιστοποίηση), το χρόνο που θα χρειαστούν ορισμένες εργασίες (ελαχιστοποίηση), τη χρησιμότητα (μεγιστοποίηση) που θα λάβουμε από τα αντικείμενα που θα πάρουμε μέσα στο σακίδιο μας και άλλα. Η λύση αναζητείται κάτω από ορισμένους περιορισμούς όπως για παράδειγμα χωρικούς, χρονικούς, οικονομικούς και άλλους, καθώς και κάτω από τις συνθήκες του προβλήματος όπως πχ. ο χρόνος προετοιμασίας των μηχανημάτων, το κόστος μετακινήσεων οχημάτων, το κόστος ενοικίασης διαφημιστικού χώρου / χρόνου και άλλα.

3.1 Ευρετικοί και Μεθευρετικοί Αλγόριθμοι

Οι προσεγγιστικοί αλγόριθμοι χωρίζονται σε δύο κατηγορίες, τους ευρετικούς και τους μεθευρετικούς. Ο στόχος ενός **ευρετικού** είναι να παράγει μια λύση σε ένα εύλογο χρονικό πλαίσιο που να είναι αρκετά καλό για την επίλυση του προβλήματος στο χέρι. Αυτή η λύση μπορεί να μην είναι η καλύτερη από όλες τις λύσεις σε αυτό το πρόβλημα ή μπορεί απλά να προσεγγίσει την ακριβή λύση. Αλλά εξακολουθεί να είναι πολύτιμη, διότι η εύρεση του δεν απαιτεί υπερβολικά μεγάλο χρονικό διάστημα. Οι ευρετικοί αλγόριθμοι μπορούν να

παράγουν αποτελέσματα από μόνοι τους ή μπορούν να χρησιμοποιηθούν σε συνδυασμό με αλγόριθμους βελτιστοποίησης για τη βελτίωση της αποτελεσματικότητάς τους (π.χ., μπορούν να χρησιμοποιηθούν για την παραγωγή καλών τιμών σπόρων). Τα αποτελέσματα σχετικά με NP-hard προβλήματα στη θεωρητική επιστήμη των υπολογιστών καθιστούν τους ευρετικούς τη μόνη βιώσιμη επιλογή για μια ποικιλία σύνθετων προβλημάτων βελτιστοποίησης που πρέπει να λυθούν συνήθως σε πραγματικές εφαρμογές. Οι ερευνητικοί υπόκεινται σε ολόκληρο το πεδίο της Τεχνητής Νοημοσύνης και στην προσομοίωση της σκέψης του υπολογιστή, καθώς μπορούν να χρησιμοποιηθούν σε καταστάσεις όπου δεν υπάρχουν γνωστοί αλγόριθμοι

Ένα σημείο που πρέπει να διευκρινιστεί όσο αφορά τους ευρετικούς αλγόριθμους, είναι η ποιότητα της λύσης. Σε μερικά προβλήματα, όπως αναφέρθηκε, είναι σχεδόν αδύνατο να βρεθεί η βέλτιστη λύση για κάποιο πρόβλημα σε ικανοποιητικό χρόνο, και έτσι γεννιέται το ερώτημα: Πώς είναι δυνατό να είμαστε βέβαιοι για την ποιότητα της λύσης που προέκυψε από τον αλγόριθμο που αναπτύξαμε. Η απάντηση δεν είναι εύκολο να δοθεί. Ένας απλός αλλά και ο πιο συνηθισμένος τρόπος απόδειξης είναι να δημιουργήσουμε μικρότερα παραδείγματα από αυτό που θέλουμε να λύσουμε, τα οποία μπορούμε να τα λύσουμε με κάποια ακριβή μέθοδο, και να δούμε πόσο κοντά στο βέλτιστο είναι η λύση που παίρνουμε με τη χρήση του ευρετικού αλγορίθμου. Ένας άλλος τρόπος είναι η δημιουργία ενός φράγματος αποδεκτής λύσης με την επίλυση ενός χαλαρωμένου προβλήματος, για παράδειγμα με τη διαδικασία διακλάδωσης και οριοθέτησης (branch and bound), και έτσι όλες οι λύσεις που θα πάρουμε με τη χρήση του ευρετικού αλγορίθμου και που η τιμή τους δεν θα παραβιάζει την τιμή του φράγματος θα είναι ικανοποιητικές

Επειδή πολλοί ευρετικοί αλγόριθμοι παγιδεύονται σε κάποιο τοπικό ελάχιστο και δεν μπορεί να βελτιωθεί η λύση έχουν προταθεί άλλοι αλγόριθμοι που βοηθάνε στο να ξεφύγουμε από αυτό. Οι αλγόριθμοι αυτοί ονομάζονται μεθευρετικοί αλγόριθμοι μια βέλτιστη λύση για όλο τον πληθυσμό για κάποια τάξη προβλημάτων. Πολλοί μεθευρετικοί αλγόριθμοι εφαρμόζουν κάποια μορφή στοχαστικής βελτιστοποίησης, έτσι ώστε η λύση που έχει βρεθεί εξαρτάται από το σύνολο των τυχαίων μεταβλητών που δημιουργούνται. Στην συνδυαστική βελτιστοποίηση, με την αναζήτηση μιας μεγάλης σειράς εφικτών λύσεων, οι μεθευρετικοί μπορούν συχνά να βρουν καλές λύσεις με λιγότερη υπολογιστική προσπάθεια από τους αλγόριθμους βελτιστοποίησης, τις επαναληπτικές μεθόδους ή από απλούς ευρετικούς. Ως εκ τούτου, είναι χρήσιμες προσεγγίσεις για προβλήματα βελτιστοποίησης.

Υπάρχει πληθώρα μεθευρετικών αλγορίθμων γεγονός που καθιστά την προσέγγιση όλων δύσκολη. Για αυτόν τον λόγο θα παρουσιαστούν σε τέσσερις γενικές κατηγορίες:

1. **Μεθευρετικοί αλγόριθμοι βασισμένοι στη γειτονιά αναζήτησης:** Αλγόριθμοι οι οποίοι χρησιμοποιώντας τη μέθοδο της τοπικής αναζήτησης διερευνούν την περιοχή γύρω από ένα σημείο που έχει βρεθεί ως τοπικό ελάχιστο, δηλαδή πραγματοποιούν μικρές αλλαγές σε μία μόνο λύση. Οι αλγόριθμοι αυτοί λειτουργούν με διάφορες στρατηγικές και έχουν κατηγοριοποιηθεί σύμφωνα με τους Μαρινάκη et al (2011) ως εξής:
 - Επαναληπτικές διαδικασίες που αρχίζουν από διαφορετικές αρχικές λύσεις π.χ. αλγόριθμος επαναληπτικής τοπικής αναζήτησης (iterated local search).
 - Αλγόριθμοι που δέχονται γειτονικές κινήσεις που δεν βελτιώνουν τη λύση π.χ. αλγόριθμος προσομοιωμένης απόπτωσης (simulated annealing).
 - Αλγόριθμοι που αλλάζουν τη γειτονιά αναζήτησης π.χ. αλγόριθμος μεταβλητής γειτονιάς αναζήτησης (variable neighborhood search).
 - Αλγόριθμοι που αλλάζουν την αντικειμενική συνάρτηση ή κάποια από τα δεδομένα του προβλήματος π.χ. αλγόριθμος καθοδηγούμενης τοπικής αναζήτησης (guided local search)
2. **Κατασκευαστικοί μεθευρετικοί αλγόριθμοι:** Κατασκευάζουν λύσεις από δικά τους συστατικά στοιχεία αντί να βελτιώνουν υπάρχουσες λύσεις π.χ. η διαδικασία άπληστης τυχαιοποιημένης προσαρμοστικής αναζήτησης (greedy randomized adaptive search procedure - GRASP).
3. **Μεθευρετικοί αλγόριθμοι βασισμένοι στον πληθυσμό:** Αλγόριθμοι οι οποίοι βρίσκουν καλές λύσεις επιλέγοντας και συνδυάζοντας κατ' επανάληψη ήδη υπάρχουσες λύσεις από μια ομάδα που ονομάζεται πληθυσμός. Ξεκινώντας από έναν αρχικό πληθυσμό επανειλημμένα αντικαθιστούν την τρέχουσα με μία νέα γενιά πληθυσμού. Τα πιο σημαντικά μέλη αυτής της κατηγορίας είναι οι **εξελικτικοί**

αλγόριθμοι, οι οποίοι μέσω μιας επαναληπτικής διαδικασίας εξελίσσουν νέους πληθυσμούς ατόμων από τους παλιούς π.χ. γενετικοί αλγόριθμοι (genetic algorithms). Σε αυτήν την κατηγορία ανήκει και ο αλγόριθμος της διαφορικής εξέλιξης που εξετάζεται εκτενέστερα στην παρούσα εργασία.

4. **Υβριδικοί μεθευρετικοί αλγόριθμοι:** Αλγόριθμοι οι οποίοι ενσωματώνουν σε μεθευρετικούς αλγόριθμους κάποιες άλλες τεχνικές. Οι Blum και Roli (2008) διαχωρίζουν του υβριδικούς μεθευρετικούς αλγόριθμους σε δύο κατηγορίες:

- Αλγόριθμοι οι οποίοι συνδυάζουν στοιχεία διαφορετικών μεθευρετικών τεχνικών.
- Αλγόριθμοι οι οποίοι συνδυάζουν μεθευρετικές τεχνικές με τεχνικές άλλων πεδίων όπως της επιχειρησιακής έρευνας και της τεχνητής νοημοσύνης.

3.2 Αλγόριθμοι που έχουν εφαρμοστεί στο πρόβλημα του βέλτιστου σχεδιασμού γραμμής προϊόντων

3.2.1 Greedy Heuristic

Είναι ένας **ευρετικός αλγόριθμος** που αρχικά εισήχθη από τους Green and Krieger (1985) αλλά έπειτα χρησιμοποιήθηκε και σε άλλες εφαρμογές επίλυσης του συγκεκριμένου προβλήματος. Ο αλγόριθμος ξεκινά με το σχεδιασμό μιας γραμμής προϊόντων η οποία περιέχει ένα προϊόν ως το μοναδικό που μεγιστοποιεί τα κέρδη και στη συνέχεια προσθέτει κάθε φορά το προϊόν που μεγιστοποιεί την αντικειμενική συνάρτηση της ημιτελούς λύσης. Η επιλογή των προϊόντων γίνεται από ένα προεπιλεγμένο υποσύνολο προϊόντων (σύνολο αναφοράς). Η μέθοδος σταματά όταν φτάσουμε τον επιθυμητό αριθμό προϊόντων της γραμμής παραγωγής. Η μέθοδος δεν εξασφαλίζει ότι θα πάρουμε την βέλτιστη λύση αλλά ότι η λύση που θα βρει θα είναι ικανοποιητικής Αυτό συμβαίνει γιατί ότ η τελική λύση εξαρτάται από την επιλογή του πρώτου στοιχείου της λύσης, που μπορεί να μην είναι τοπικά βέλτιστο.

3.2.2 Divide-and-Conquer Heuristic

Η μέθοδος του «**διαίρει και βασίλευε**» εισήχθη από τους **Green and Krieger** (1988) και έπειτα (1993) προτάθηκε για την επίλυση του προβλήματος βέλτιστου σχεδιασμού γραμμής προϊόντων. Ο αλγόριθμος αυτός διαιρεί τη γραμμή προϊόντων σε ομάδες χαρακτηριστικών και διατηρώντας τις ομάδες σταθερές, ερευνά όλους τους πιθανούς συνδυασμούς μίας ομάδας, επιλέγοντας τη συνολικά καλύτερη λύση. Στο συγκεκριμένο πρόβλημα θεωρούμε κάθε προϊόν ως ένα δικό του σύνολο από ιδιότητες. Έτσι ξεκινάμε από μια τυχαία λύση γραμμής προϊόντων όπου δοκιμάζουμε κάθε φορά όλους τους πιθανούς συνδυασμούς για ένα προϊόν, βελτιστοποιώντας το κέρδος του και κρατώντας τα υπόλοιπα σταθερά. Η διαδικασία συνεχίζεται και με τα υπόλοιπα προϊόντα μέχρις ότου η λύση δεν βελτιώνεται περαιτέρω με τη βελτίωση ενός προϊόντος. Η διαδικασία δεν εγγυάται καθολική βέλτιστη λύση αφού μπορεί να εγκλωβιστεί σε κάποιο τοπικό βέλτιστο.

3.2.3 Product-Swapping Heuristic

Αυτή είναι μια ευρετική μέθοδος των **Green and Krieger** (1985) όπου χρησιμοποιεί την εναλλαγή προϊόντων για τη δημιουργία λύσεων. Όμοιά της είναι η *Interchange heuristic*. Η μέθοδος δημιουργεί μια τυχαία αρχική λύση γραμμής προϊόντων και βρίσκει τα κέρδη που αποφέρει. Στη συνέχεια εξετάζεται αν η εναλλαγή ενός προϊόντος με κάποιο άλλο «υποψήφιο» που δεν είναι μέλος της τρέχουσας λύσης, βελτιώνει την τιμή της αυξάνει τα κέρδη της συγκεκριμένης γραμμής προϊόντων. Αν τα αυξάνει τότε το τρέχον προϊόν αντικαθίσταται από το νέο. Η διαδικασία επαναλαμβάνεται και σταματά όταν η γραμμή δεν επιδέχεται περαιτέρω βελτίωση. Όπως και στην μέθοδο *Divide-and-Conquer Heuristic*, δε μπορούμε να εγγυηθούμε το ολικό βέλτιστο αλλά ενός τοπικού βέλτιστου καθώς η αλλαγή ενός μόνο προϊόντος μπορεί να μη βελτιώνει τη λύση.

3.2.4 Dynamic Programming Heuristic

Οι **Kohli and Krishnamusti** (1987), ανέπτυξαν το συγκεκριμένο μοντέλο ευρετικού δυναμικού προγραμματισμού, για την επίλυση του βέλτιστου σχεδιασμού ενός προϊόντος και αργότερα οι **Kohli and Sukumar** (1990) επεκτάθηκαν και το εφάρμοσαν για περισσότερα προϊόντα. Η διαδικασία ξεκινά κατασκευάζοντας μια γραμμή προϊόντων με ένα χαρακτηριστικό σε κάθε βήμα. Στο πρώτο στάδιο, στο πρώτο χαρακτηριστικό του πρώτου προϊόντος αξιολογείται το κάθε επίπεδο του όσο αφορά το οριακό κέρδος ανάλογα

με τους στόχους που έχουν τεθεί. Επιλέγεται, εκείνο το επίπεδο που ικανοποιεί καλύτερα το στόχο που έχουμε θέσει. Μετά, για το δεύτερο χαρακτηριστικό, μελετώνται όλα τα πιθανά επίπεδα που σε συνδυασμό με το πρώτο χαρακτηριστικό, δίνουν την καλύτερη ημιτελή λύση. Οι συνδυασμοί είναι τόσοι όσοι τα επίπεδα του δεύτερου χαρακτηριστικού. Η διαδικασία συνεχίζεται με τον ίδιο τρόπο μέχρι και το τελευταίο χαρακτηριστικό και έως ότου ο αριθμός των συνδυασμών των χαρακτηριστικών είναι ίσος με τον αριθμό των επιθυμητών προϊόντων στη γραμμή παραγωγής. Η τελική λύση εξαρτάται από το σημείο εκκίνησης, γι αυτό επιλέγουμε τυχαία κάθε φορά διαφορετικό χαρακτηριστικό να αξιολογήσουμε πρώτα.

3.2.5 Beam Search Heuristic

Ο ευρετικός αλγόριθμος δενδρικής αναζήτησης Beam Search αναπτύχθηκε αρχικά για προβλήματα τεχνητής νοημοσύνης στην αναγνώριση φωνής και εικόνας. Για το πρόβλημα βέλτιστου σχεδιασμού γραμμής προϊόντων εφαρμόστηκε για πρώτη φορά από τους Nair et al. (1995). Η διαδικασία είναι όμοια με το dynamic programming heuristic αλλά με δύο βασικές διαφορές. Αρχικά, αντί να εξετάζει ένα χαρακτηριστικό τη φορά, εξετάζει ταυτόχρονα διαφορετικά σύνολα χαρακτηριστικών. Επίσης, αντί να δημιουργεί ολόκληρη γραμμή παραγωγής, προσθέτει ένα προϊόν τη φορά με τρόπο ανάλογο του *Greedy Heuristic*. Έτσι η έρευνα πρέπει να ξεκινά κάθε φορά από μια τυχαίοποιημένη σειρά χαρακτηριστικών. Η συγκεκριμένη διαδικασία δημιουργεί ταυτόχρονα ένα σύνολο λύσεων γραμμής παραγωγής όπου κρατάμε την καλύτερη.

3.2.6 Nested Partitions Heuristic

Η συγκεκριμένη μέθοδος εφαρμόστηκε στο πρόβλημα της βέλτιστης γραμμής προϊόντων από τους Shi et al. (2001). Αρχικά διαιρεί το χώρο των λύσεων σε διαφορετικές περιοχές όπου τις αξιολογεί και επιλέγει τον χώρο όπου θα φέρει τα καλύτερα αποτελέσματα. Έπειτα, συνεχίζει να διαιρεί και να αξιολογεί τις υποσχόμενες περιοχές σε μικρότερες και μικρότερες έως ότου καταλήξει σε μια γραμμή παραγωγής. Για την αξιολόγηση των υπό μελέτη περιοχών μπορούν να χρησιμοποιηθούν άλλοι ευρετικοί αλγόριθμοι όπως είναι η one-opt-coordinate ascent method όπου κατά την εφαρμογή τους στην περιοχή ξεχωρίζουμε τις καλύτερες λύσεις και δημιουργούμε ένα δείκτη ως μέτρο της ποιότητας των λύσεων της περιοχής. Αν διαπιστωθεί ότι γειτονικές περιοχές περικλείουν την υπό

μελέτη περιοχή με καλύτερης ποιότητας λύσεις μπορούμε να εφαρμόσουμε ένα «άλμα» προς τα πίσω (backtrack) και να αξιολογήσουμε ξανά την περιοχή.

3.2.7 Coordinate Ascent

Η συγκεκριμένη ευρετική μέθοδος εφαρμόστηκε στο υπό μελέτη πρόβλημα από τους *Green et al.*(1989). Ξεκινά επιλέγοντας και αξιολογώντας μια τυχαία γραμμή προϊόντων. Στη συνέχεια επιλέγει τυχαία κάθε πιθανό επίπεδο του κάθε χαρακτηριστικού του προϊόντος και εξετάζει αλλαγές του επιπέδου, παρατηρώντας αν βελτιώνει τη λύση. Σε περίπτωση που αυτό συμβαίνει αποδέχεται την αλλαγή. Η εξέταση συνεχίζεται έως ότου δεν παρατηρηθεί κάποια σημαντική μεταβολή στα κέρδη. Υπάρχουν οι εκδοχές αλγόριθμου one-opt, two-opt, three-opt που ελέγχουν την μεταβολή ενός, δύο και τριών χαρακτηριστικών αντίστοιχα. Η εύρεση καθολικού βέλτιστου δεν είναι εγγυημένη καθώς η τρέχουσα λύση μπορεί να αποτελεί τοπικό βέλτιστο και η υπό μελέτη αλλαγές να μη βελτιώνουν το κέρδος της λύσης, αντίθετα άλλες λύσεις που διαφέρουν σε περισσότερα χαρακτηριστικά να οδηγούν σε μεγαλύτερα κέρδη.

3.2.8 Genetic Algorithm

Ο γενετικός αλγόριθμος (GA) είναι μεθευρετικός που εμπνέεται από τη διαδικασία φυσικής επιλογής που ανήκει στη μεγαλύτερη τάξη εξελικτικών αλγορίθμων (EA). Οι γενετικοί αλγόριθμοι χρησιμοποιούνται συνήθως για τη δημιουργία λύσεων υψηλής ποιότητας, βασιζόμενα σε φορείς που εμπνέονται από βιολογικά στοιχεία, όπως η μετάλλαξη, η διασταύρωση και η επιλογή. Ο John Holland εισήγαγε αρχικά τον γενετικό αλγόριθμο (1960) με βάση την έννοια της θεωρίας της εξέλιξης του Δαρβίνου. Στη συνέχεια, ο σπουδαστής του Goldberg τον επέκτεινε (1989). Στο συγκεκριμένο πρόβλημα εφαρμόστηκαν από τους Balakrishnan and Jacob, Alexouda and Paparrizos, Steiner and Hruschka, Balakrishnan et al. . Η διαδικασία ξεκινά με ένα πληθυσμό από τυχαίες λύσεις του οποίου τα «δυνατότερα» μέλη επικρατούν και επιβιώνουν και παράγουν την νέα γενιά λύσεων. Η «Αναπαραγωγή» (*Reproduction*) περιλαμβάνει τη «Διασταύρωση»(crossover) των γονιίων για να παράγουν τα παιδιά και στη συνέχεια μικρές «Μεταλλάξεις» (*mutation*). Η καταλληλότητα ενός οργανισμού – χρωμοσώματος αξιολογείται ανάλογα με τη δυνατότητα του να αναπαραχθεί πριν πεθάνει, στην προκειμένη περίπτωση είναι η τιμή της αντικειμενικής συνάρτησης. Αυτό σημαίνει ότι όσο μεγαλύτερη τιμή έχει ένα χρωμόσωμα, τόσο μεγαλύτερη πιθανότητα έχει να επιλεγεί για την επόμενη γενιά. Εάν δύο

χρωμοσώματα ανταλλάζουν τα καλά τους γονίδια, τότε υπάρχει υψηλή πιθανότητα να παράγουν καλούς απογόνους οι οποίοι με τη σειρά τους θα ανταλλάξουν τα καλά τους γονίδια. Για λόγους διαφοροποίησης συχνά συμβαίνει η μετάλλαξη (mutation). Στη διαδικασία της μετάλλαξης ένα από τα γονίδια του χρωμοσώματος που έχει επιλεγεί, επιλέγεται τυχαία και η τιμή του αντικαθίσταται από μία άλλη που παράγεται τυχαία. Η μετάλλαξη παράγει νέα χρωμοσώματα που δεν θα είχαν τη δυνατότητα να παραχθούν με τη διαδικασία της διασταύρωσης. Με τον τρόπο αυτό σε κάθε γενιά παράγονται νέα χρωμοσώματα επιτρέποντας στον αλγόριθμο να αναζητήσει λύσεις σε νέα μονοπάτια, αποφεύγοντας τον εγκλωβισμό σε τοπικά ελάχιστα. Επίσης, η διασταύρωση δύο χρωμοσωμάτων μπορεί να γίνεται σε παραπάνω από ένα σημεία διαχωρισμού για επίτευξη μεγαλύτερης διαφοροποίησης. Συμπερασματικά, ενώ η αναπαραγωγή μειώνει την ποικιλομορφία του πληθυσμού, η μετάλλαξη την ενισχύει με αποτέλεσμα να αποφεύγεται η πρόωρη σύγκλιση του αλγορίθμου σε τοπικά βέλτιστα. Η συγκεκριμένη διαδικασία σταματά όταν ικανοποιηθεί το κριτήριο τερματισμού.

3.2.9 Simulated Annealing

Η συγκεκριμένη μέθοδος της προσομοιωμένης απόπτωσης είναι ένας δημοφιλής "αλγόριθμος τελευταίας λύσης" για δύσκολα διακριτά προβλήματα βελτιστοποίησης. Το όνομα της μεθόδου προέρχεται από τη φυσική διαδικασία της απόπτωσης, στην οποία ένα υγρό ψύχεται βραδέως σε ένα θερμαινόμενο λουτρό για να σχηματίσει ένα στερεό σε κατάσταση χαμηλής ενέργειας. Μια λεπτομερής εισαγωγή στην μέθοδο παρουσιάστηκε στο Aarts, από Korst και van Laarhoven (1997). Ο αλγόριθμος εφαρμόστηκε στο υπό μελέτη πρόβλημα βέλτιστης γραμμής προϊόντων από τους Alexandre Belloni , Robert M. Freund, Matthew Selove , Duncan Simester. Ο simulated annealing είναι παρόμοιος με τον coordinate ascent κατά το ότι ξεκινά με τυχαία επιλεγμένη λύση και προχωρά στη δοκιμή αλλαγών τυχαίων χαρακτηριστικών στην τρέχουσα λύση. Η διαφορά είναι ότι ο συγκεκριμένος αλγόριθμος μερικές φορές δέχεται αλλαγές χαρακτηριστικών που μειώνουν τα κέρδη. Η πιθανότητα αποδοχής μιας τέτοιας αρνητικής αλλαγής εξαρτάται από το μέγεθος της πτώσης των κερδών και επίσης μειώνεται με την πάροδο του χρόνου καθώς ο αλγόριθμος εξελίσσεται μέσω ενός προκαθορισμένου "προγράμματος ψύξης". Όπως ο coordinate ascent, και ο simulated annealing εγγυάται ένα τοπικό βέλτιστο αλλά όχι το καθολικό βέλτιστο. Επειδή η διαδικασία μερικές φορές δέχεται αλλαγές χαρακτηριστικών που μειώνουν τα κέρδη, έχει τη δυνατότητα να ξεφύγει από μια τοπικά βέλτιστη λύση με την ελπίδα να βρεθεί κάποια καλύτερη λύση. Για το λόγο αυτό, η μέθοδος μπορεί να αναμένεται να ξεπεράσει την απόδοση του coordinate ascent.

3.2.10 Particle Swarm Optimization

Η ιδέα εφαρμόζεται στον τομέα της τεχνητής νοημοσύνης και ο όρος εισήχθη από τους Gerardo Beni και Jing Wang το 1989. Η νοημοσύνη σμήνους είναι η συλλογική συμπεριφορά μη κατανεμημένων αυτοοργανωμένων φυσικών ή τεχνητών συστημάτων. Δρουν μαζί ως μια μάζα χωρίς να ελέγχονται από κάποιο συγκεκριμένο οργανισμό, πετυχαίνοντας στόχους που δεν θα μπορούσαν τα τους πετύχουν ατομικά. Ο αλγόριθμος βελτιστοποίησης σμήνους σωματιδίων (Particle Swarm Optimization). Προτάθηκε για πρώτη φορά από τους Kennedy και Eberhat και εφαρμόστηκε στο υπό μελέτη πρόβλημα το 2010 στην εργασία «Particle swarm optimization for optimal product line design» των Tsafarakis Stelios, Marinakis Ioannis και Matsatsinis Nikolaos. Ο αλγόριθμος χρησιμοποιεί τη φυσική κίνηση των ατόμων του πληθυσμού στο σμήνος και χρησιμοποιεί ευέλικτους μηχανισμούς για να προσαρμόζεται στις τοπικές και ολικές ικανότητες εξερεύνησης των ατόμων του σμήνους. Συγκεκριμένα, κάθε σωματίδιο έχει μια θέση που αντιστοιχεί στο διάνυσμα της θέσης του. Τα σωματίδια συνεργάζονται για την εύρεση καλύτερων λύσεων όπου και συνεχίζουν να ακολουθούν την καλύτερη. Ο αλγόριθμος χρησιμοποιεί μηχανισμούς και περιοριστικούς παράγοντες που εξυπηρετούν άλλοτε την σύγκλιση σε κάποια ποιοτική λύση και άλλοτε τον απεγκλωβισμό από τοπικά ελάχιστα.

4. Ο αλγόριθμος διαφορικής εξέλιξης

Η **Διαφορική Εξέλιξη (Differential Evolution (DE))** είναι ένας στοχαστικός, βασιζόμενος σε πληθυσμό, αλγόριθμος. Προτάθηκε από τους Storn και Price το 1995. Τα αμέσως επόμενα χρόνια η επιτυχία του ήταν εμφανής καθώς το 1996 ξεχώρισε στον πρώτο διεθνή διαγωνισμό για την εξελικτική βελτιστοποίηση και στη συνέχεια δημοσιεύθηκαν αρκετά βιβλία με θέμα τον συγκεκριμένο αλγόριθμο που τον αναγνώριζαν πλέον ως ένα από τους πιο ανταγωνιστικούς της κατηγορίας του. Η Διαφορική Εξέλιξη έχει τα βασικά χαρακτηριστικά των εξελικτικών αλγορίθμων αλλά σε σύγκριση με του υπόλοιπους εξελικτικούς αλγορίθμους είναι πιο απλός αφού έχει απλούστερη εφαρμογή. Η κύρια του μορφή χρειάζεται μερικές γραμμές κώδικα και δύο παραμέτρους, το τελεστή μετάλλαξης και τον τελεστή διασταύρωσης. Η μέθοδος του αλγόριθμου εστιάζει στην απόσταση μεταξύ των μελών του πληθυσμού και στις διαφορετικές κατευθύνσεις που μπορεί να κινηθεί κάποιο μέλος του πληθυσμού. Για να εφαρμοστεί η διαφορική εξέλιξη θα πρέπει να έχουμε κωδικοποιήσει τις λύσεις με αναπαράσταση πραγματικού αριθμού ώστε να μπορεί να εφαρμοστεί ο τελεστής μετάλλαξης. Αρχικά από τη στιγμή που η επιτυχία του αλγόριθμου της διαφορικής εξέλιξης βασίζεται στις αποστάσεις μεταξύ των ατόμων της ομάδας πρέπει να υπάρχει σωστή κατανομή των ατόμων στο χώρο λύσεων του προβλήματος με σχετική απόσταση μεταξύ δύο ατόμων. Καθώς θα εξελίσσονται οι επαναλήψεις οι αποστάσεις ανάμεσα στα άτομα γίνονται μικρότερες μέχρι που στο τέλος των επαναλήψεων, θεωρητικά, θα πρέπει να συγκλίνουν σε μία λύση ή σε λύσεις γύρω από ένα βέλτιστο σημείο. Οι αποστάσεις μεταξύ των ατόμων του πληθυσμού είναι ένας πολύ καλός δείκτης της διασποράς των λύσεων του πληθυσμού. Εάν οι αποστάσεις ανάμεσα στα άτομα είναι μεγάλες τότε ο αλγόριθμος θα πραγματοποιεί μεγάλα βήματα κατά τη διάρκεια των επαναλήψεων και έτσι θα μπορεί να εξερευνήσει όσο το δυνατόν μεγαλύτερα σημεία του χώρου αναζήτησης. Από την άλλη αν οι αποστάσεις ανάμεσα στα άτομα είναι μικρές τότε τα βήματα που θα πραγματοποιούνται θα είναι επίσης μικρά γεγονός που θα αυξάνει την ικανότητα αναζήτησης σε περιορισμένα σημεία του χώρου λύσεων, για παράδειγμα γύρω από ένα τοπικό ελάχιστο.

4.1 Βασική μορφή διαφορικής εξέλιξης

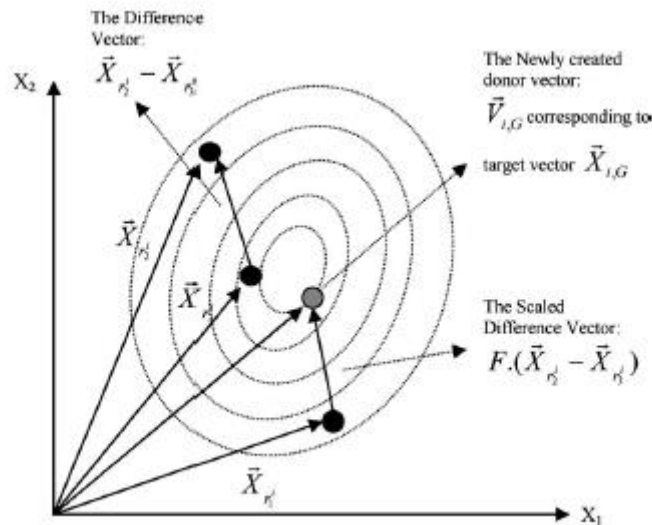
Η **αρχικοποίηση (initialization)** του αλγόριθμου γίνεται φτιάχνοντας ένα τυχαίο πληθυσμό υποψηφίων λύσεων όπου ο αρχικός πληθυσμός πρέπει να καλύπτει όσο το

δυνατόν περισσότερα ομοιόμορφα τυχαιοποιημένα άτομα X_i εντός του χώρου αναζήτησης μεταξύ των καθορισμένων ορίων.

Η **μετάλλαξη (mutation)** σημαίνει ξαφνική αλλαγή στα γονιδιακά χαρακτηριστικά ενός χρωμοσώματος. Στα πλαίσια του εξελικτικού υπολογιστικού παραδείγματος η μετάλλαξη θεωρείται επίσης ως αλλαγή ή διαταραχή με τυχαίο στοιχείο. Ο τελεστής μετάλλαξης παράγει ένα δοκιμαστικό διάνυσμα για κάθε μέλος του πληθυσμού με μετάλλαξη ενός διανύσματος στόχου στο οποίο προστίθεται η διαφορά ανάμεσα στις τιμές των γονιδίων δύο ή περισσότερων ατόμων του πληθυσμού πολλαπλασιασμένων με κάποιο βάρος. Για κάθε γονέα $x_i(t)$, το δοκιμαστικό διάνυσμα, $u_i(t)$, δημιουργείται ως εξής :ένα διάνυσμα στόχου (κάποιο άλλο μέλος του πληθυσμού), $x_{i1}(t)$, επιλέγεται από τον πληθυσμό, έτσι ώστε $i \neq i1$. Επίσης, δύο άλλα μέλη του πληθυσμού, x_{i2} και x_{i3} , επιλέγονται τυχαία από τον πληθυσμό τέτοια ώστε $i \neq i1 \neq i2 \neq i3$. Χρησιμοποιώντας τα επιλεγμένα διανύσματα, το δοκιμαστικό διάνυσμα υπολογίζεται με αντιμετάθεση των στοιχείων του διανύσματος στόχου ως ακολούθως:

$$u_i(t) = x_{i1}(t) + F(x_{i2}(t) - x_{i3}(t)) \quad (4.1)$$

όπου το $F \in (0, \infty)$ είναι ένας παράγοντας κανονικοποίησης. Το πάνω όριο του F είναι συνήθως η τιμή 1 γιατί πειραματικά έχει αποδειχθεί ότι εάν το $F > 1$ δεν υπάρχει μεγάλη έως καθόλου βελτίωση των λύσεων. Γενικά όσο μικρότερο είναι το F , τόσο μικρότερο είναι και το βήμα κατά τη διάρκεια της μετάλλαξης και τόσο περισσότερο χρόνο θα πάρει στον αλγόριθμο να συγκλίνει. Από την άλλη μεριά αν το F είναι πολύ μεγάλο τότε το βήμα θα είναι μεγάλο, γεγονός που μπορεί να οδηγήσει σε υπερπήδηση κάποιου τοπικού βέλτιστου. Έτσι, οι τιμές του F πρέπει να έχουν επιλεγθεί με τέτοιο τρόπο ώστε να μπορούν να βρίσκουν τα τοπικά βέλτιστα και ταυτόχρονα να μπορούν να εξερευνούν αποτελεσματικά ένα τοπικό βέλτιστο. Έτσι, η πιο συνηθισμένη τιμή που χρησιμοποιείται είναι το 0.5 .

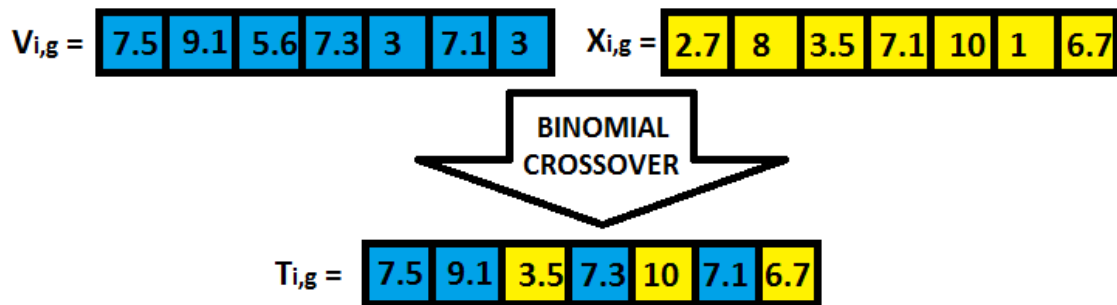


Εικόνα 1: Παρουσιάζοντας ένα απλό σχέδιο μετάλλαξης DE σε παραμετρικό χώρο 2-D

Μετά την ολοκλήρωση της φάσης της μετάλλαξης εφαρμόζεται η λειτουργία Διασταύρωσης (**crossover**) προκειμένου να ενισχυθεί η δυναμική πολυμορφία του πληθυσμού. Στους εξελικτικούς αλγόριθμους ο διαχειριστής διασταύρωσης συνήθως συνδυάζει χαρακτηριστικά από διάφορους γονείς. Στην περίπτωση της διαφορικής εξέλιξης αλγόριθμοι, αφού ο χειριστής μετάλλαξης βασίζεται ήδη σε ανασυνδυασμός ατόμων, ο ρόλος του crossover είναι κάπως διαφορετικός. Απλώς επιτρέπει την κατασκευή ενός απογόνου (που ονομάζεται δοκιμαστικό διάνυσμα), z_i , με ανάμειξη των στοιχείων του τρέχοντος διανύσματος, x_i , και των στοιχείων τα οποία έχουν παραχθεί την από μετάλλαξη, u_i . Υπάρχουν δύο κύριες παραλλαγές διασταύρωσης για ΔΕ: η διωνυμική και η εκθετική. Στην διωνυμική διασταύρωση τα γονίδια επιλέγονται τυχαία από το δοκιμαστικό διάνυσμα και από τον γονέα. Αρχικά επιλέγεται μια παράμετρος για τον τελεστή διασταύρωσης (CR) η οποία ελέγχει την αναλογία των γονιδίων που θα επιλεγούν από το δοκιμαστικό διάνυσμα. Η τιμή του CR συγκρίνεται με ένα τυχαίο αριθμό στο διάστημα $randi(0, 1)$. Εάν ο τυχαίος αριθμός είναι μικρότερος ή ίσος με το CR , η τιμή του γονιδίου του απογόνου κληρονομείται από το δοκιμαστικό διάνυσμα, αλλιώς επιλέγεται από το γονέα :

$$Z_i(t) = \begin{cases} u_i(t), & \text{εάν } randi(0,1) \leq Cr \\ x_i(t), & \text{αλλιώς.} \end{cases} \quad (4.2)$$

Συνεπώς, η επιλογή της παραμέτρου CR είναι πολύ σημαντική γιατί αν η τιμή είναι κοντά ή ίση με το 1, τότε, τα περισσότερα (ή όλα στην περίπτωση που είναι ίση με το 1) τα γονίδια του απογόνου έχουν κληρονομηθεί από το δοκιμαστικό διάνυσμα αλλιώς αν είναι κοντά στο μηδέν τότε όλα τα γονίδια κληρονομούνται από το γονέα.

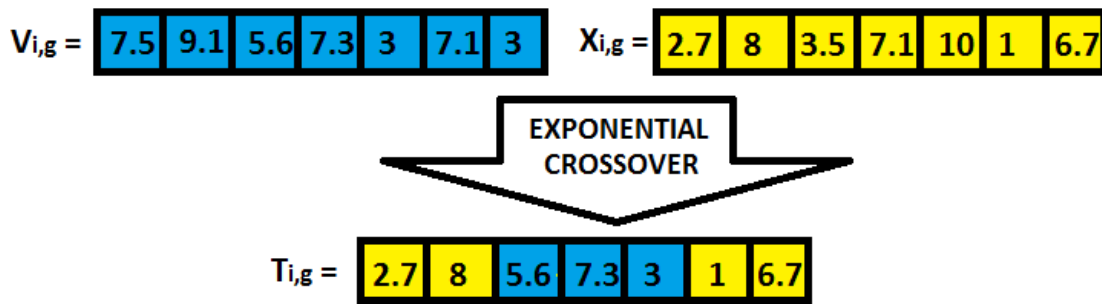


Εικόνα 2: Παράδειγμα διωνυμικής διασταύρωσης

Το εκθετικό crossover σχεδιάστηκε για να είναι παρόμοιο με το one-opt και two-opt crossover παραλλαγές που χρησιμοποιούνται σε γενετικούς αλγορίθμους. Επομένως ο δοκιμαστικός φορέας περιέχει μια αλληλουχία διαδοχικών (κυκλικά) που λαμβάνονται από τον μεταλλαγμένο φορέα. Αρχικά διαλέγουμε ένα τυχαίο αριθμό n στο διάστημα $[1,D]$. Ο ακέραιος αυτός λειτουργεί ως σημείο εκκίνησης στο διάνυσμα στόχο, από όπου ξεκινάει η ανταλλαγή σωματιδίων με τον δότη φορέα. Επίσης επιλέγουμε έναν άλλο ακέραιο L από το διάστημα $[1,D]$ που υποδηλώνει τον αριθμό σωματιδίων που θα περιέχει

το διάνυσμα στόχου από τον δότη φορέα. Η δομή του δοκιμαστικού φορέα μπορεί να περιγραφεί από :

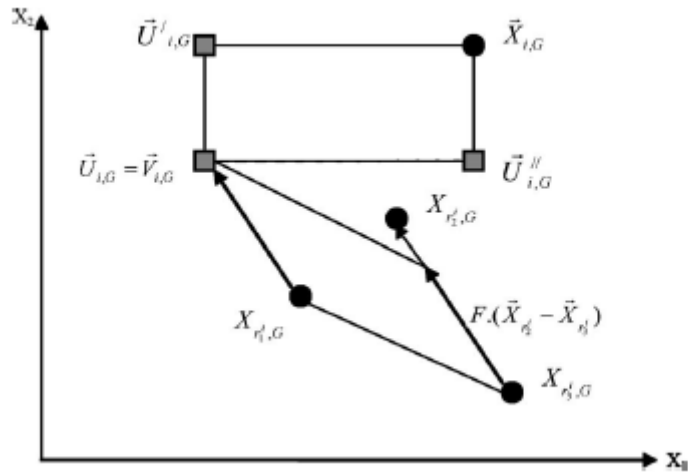
$$Z_{j,i}(t) = \begin{cases} u_{j,i}(t) & \text{για } j = \langle n \rangle_D, \langle n + 1 \rangle_D, \dots, \langle n + L - 1 \rangle_D \\ x_{j,i}(t), & \text{για τα υπόλοιπα } j \in [1, D]. \end{cases} \quad (4.3)$$



Εικόνα 3: Παράδειγμα εκθετικής διασταύρωσης

Μετά τον τελεστή διασταύρωσης, το τελικό στάδιο είναι η **επιλογή (selection)** όπου η συνάρτηση καταλληλότητας του απογόνου $z_i(t)$ υπολογίζεται και εάν είναι καλύτερο από την συνάρτηση ακαταλληλότητας του γονέα τότε επιλέγεται για την επόμενη γενιά αλλιώς ο γονέας επιβιώνει για μία ακόμα γενιά.

$$X_i(t) = \begin{cases} z_i(t), & \text{εάν } f(x_i(t)) \leq f(z_i(t)) \\ x_i(t), & \text{αλλιώς.} \end{cases} \quad (4.4)$$



Εικόνα 4: Διάφοροι πιθανοί φορείς δοκιμής σχηματίστηκαν λόγω ομοιομορφίας διωνυμικής διασταύρωσης μεταξύ του στόχου και των μεταλλαγμένων φορέων σε χώρο αναζήτησης 2-D

Ψευδοκώδικας Αλγορίθμου Διαφορικής Εξέλιξης

Αρχικοποίηση

Αρχικοποίηση των παραμέτρων ελέγχου F και CR

Επιλογή του τελεστή μετάλλαξης

Δημιουργία του αρχικού πληθυσμού

Υπολογισμός της συνάρτησης ποιότητας για κάθε μέλος του πληθυσμού

Κύρια Φάση

Do while δεν έχουν ικανοποιηθεί τα κριτήρια τερματισμού

Επέλεξε το διάνυσμα του γονέα $\mathbf{x}_i(\mathbf{t})$

Δημιούργησε το δοκιμαστικό διάνυσμα $\mathbf{u}_i(\mathbf{t})$ εφαρμόζοντας τον τελεστή μετάλλαξης

Δημιούργησε τον απόγονο $\mathbf{z}_i(\mathbf{t})$ εφαρμόζοντας τον τελεστή διασταύρωσης

Υπολογισμός της συνάρτησης ποιότητας (fitness) για τους απογόνους

If fitness ($\mathbf{z}_i(\mathbf{t})$) \leq fitness($\mathbf{x}_i(\mathbf{t})$) **then**

Αντικατέστησε το γονέα με τον απόγονο στην επόμενη γενιά

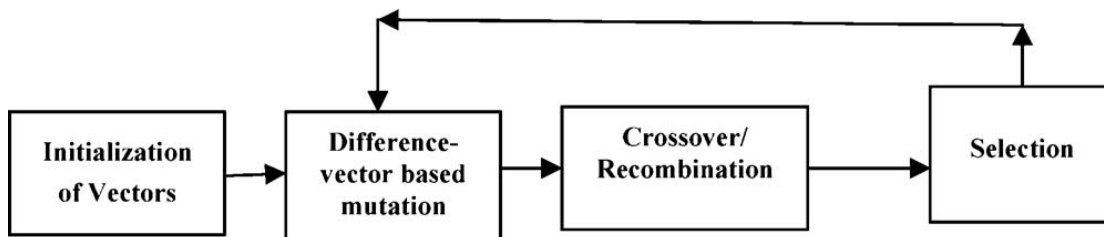
else

Πρόσθεσε το γονέα στην νέα γενιά

endif

Enddo

Επέστρεψε το καλύτερο μέλος του πληθυσμού (βέλτιστη λύση).



Εικόνα 5: Βήματα Αλγόριθμου Διαφορικής Εξέλιξης

4.2 Διαφοροποιήσεις στην εξίσωση μετάλλαξης

Διαφοροποιήσεις πάνω στην βασική μορφή του αλγορίθμου βλέπουμε κυρίως στο σχήμα μετάλλαξης. Εκτός από τον βασικό σχήμα μετάλλαξης όπου αναφέραμε στην προηγούμενη ενότητα, έχουν προταθεί από τους Storn and Pierce ακόμη τέσσερα σχήματα μετάλλαξης τα οποία περιγράφονται στη συνέχεια. Για να μπορεί κάποιος να ξεχωρίζει αυτές τις παραλλαγές στο χώρο της Διαφορικής Εξέλιξης έχει δοθεί ένας συμβολισμός που βοηθάει στο να ξεχωρίσει κάποιος τον τελεστή μετάλλαξης που χρησιμοποιείται. Έτσι, η κάθε παραλλαγή συμβολίζεται με τέσσερα στοιχεία της μορφής DE/x/y/z. Όπου, το DE σημαίνει ότι χρησιμοποιούμε τη μέθοδο της διαφορικής εξέλιξης, το x αναφέρεται στον τρόπο επιλογής του διανύσματος στόχου, το y αναφέρεται στο πλήθος των εξισώσεων διαφορών που χρησιμοποιούνται και το z αναφέρεται στον τελεστή διασταύρωσης που

χρησιμοποιείται. Επομένως, η στρατηγική που περιγράψαμε παραπάνω συμβολίζεται ως $DE/rand/1/z$.

Υπάρχουν οι ακόλουθες παραλλαγές :

- **DE/best/1/z:**

$$u_i(t) = X_{best}(t) + F(X_{i1}(t) - X_{i2}(t)) \quad (4.5)$$

- **DE/best/2/z:**

$$u_i(t) = X_{best}(t) + F(X_{i1}(t) - X_{i2}(t)) + F(X_{i3}(t) - X_{i4}(t)) \quad (4.6)$$

- **DE/rand-to-best/1/z:**

$$u_i(t) = X_i(t) + F(X_{best}(t) - X_i(t)) + F(X_{i1}(t) - X_{i2}(t)) \quad (4.7)$$

- **DE/rand/2/z:**

$$u_i(t) = X_{i1}(t) + F(X_{i2}(t) - X_{i3}(t)) + F(X_{i4}(t) - X_{i5}(t)) \quad (4.8)$$

Όπου: οι δείκτες X_{i1} , X_{i2} , X_{i3} , X_{i4} , X_{i5} είναι ακέραιοι αριθμοί που επιλέχθηκαν από τον τυχαίο πληθυσμό, και όλοι είναι διαφορετικοί από τον δείκτη βάσης i . Αυτοί οι δείκτες δημιουργούνται τυχαία μία φορά για κάθε φορέα δότη. Ο παράγοντας μετάλλαξης F είναι μια θετική παράμετρος ελέγχου για την κλιμάκωση των διαφορών διανυσμάτων. Το $X_{best}(t)$ είναι ο καλύτερος μεμονωμένος φορέας με την καλύτερη φυσική ικανότητα (δηλ. ψηλότερη αντικειμενική τιμή συνάρτησης για ένα πρόβλημα βελτιστοποίησης) στον πληθυσμό της γενιάς t . Παρατηρούμε ότι ορισμένες από τις στρατηγικές για τη δημιουργία του φορέα δότη μπορεί να είναι μεταλλαγμένοι ανασυνδυασμοί, για παράδειγμα, το μοντέλο **DE/rand-to-best/1/z** που αναφέρθηκε παραπάνω μεταλλάσσει βασικά ένα διπλό διάνυσμα ανασυνδυασμένο. Οι Storn και Price πρότειναν συνολικά δέκα διαφορετικές στρατηγικές για την ΔΕ και ορισμένες κατευθυντήριες γραμμές για την εφαρμογή αυτών των στρατηγικών σε οποιοδήποτε πρόβλημα. Αυτές οι στρατηγικές προέρχονται από τα πέντε διαφορετικά σχήματα μετάλλαξης διαφορικής εξέλιξης που περιγράφονται πιο πάνω. Κάθε στρατηγική μετάλλαξης συνδυάστηκε είτε με το "εκθετικό" (exponential) crossover ή το "διωνυμικό" (binomial) crossover. Αυτό απέφερε συνολικά $5 \times 2 = 10$ στρατηγικές ΔΕ.

4.3 Βασικές Εφαρμογές Διαφορικής Εξέλιξης

Ο αλγόριθμος ΔΕ έχει εφαρμοστεί στον ευρύτερο χώρο της επιχειρησιακής έρευνας αλλά και σε άλλους επιστημονικούς τομείς.

Το **πρόβλημα δρομολόγησης στόλου** (Vehicle Routing Problem, πολλά οχήματα / πολλοί πωλητές) είναι γνωστό NP-hard πρόβλημα βελτιστοποίησης που έχει εφαρμοστεί σε ευρεία διαδεδομένες και καθημερινές εφαρμογές και σε ποικιλία διαφορετικών εκδόσεων. Λόγω της πολυπλοκότητας, μεγάλες περιπτώσεις VRP είναι δύσκολο να επιλυθούν χρησιμοποιώντας ακριβείς μεθόδους. Αντίθετα, διαφορετικοί ευρετικοί και μεθευρετικοί αλγόριθμοι χρησιμοποιήθηκαν για την εύρεση εφικτών λύσεων VRP. Το VRP ορίστηκε αρχικά από τον Dantzig. Είναι ένα πρόβλημα σχεδιασμού ενός ελάχιστου κόστους διαδρομής που εξυπηρετεί μια συλλογή πελατών με στόλο οχημάτων, γνωστή ως πρόβλημα δρομολόγησης οχημάτων και αποτελεί σημαντική πρόκληση στον τομέα της εφοδιαστικής, της διανομής και της μεταφοράς, επειδή εν μέρει η μεταφορά και η διανομή συμβάλλουν περίπου 20% του συνολικού κόστους ενός προϊόντος. Σε συγκεκριμένη έρευνα των Pavel Kromer, Vaclav Snasel, Ajith Abraham, Eshetie Berhan για την επιχείρηση εξυπηρέτησης λεωφορείων της πόλης Anbessa (ACBSE), Αντίς Αμπέμπα, Αιθιοπία, αναπαριστούν ένα σύνολο από διαδρομές ως μεταλλαγή των θεωρούμενων τοποθεσιών. Ο DE χρησιμοποιεί μεταστοιχειώσεις στις αναπαραστάσεις που αυτόματα βελτιστοποιεί τον αριθμό των οχημάτων όταν δίνεται ένα ανώτερο όριο και αποφεύγει το δημιουργία παράνομων υποψηφίων λύσεων. Στο συγκεκριμένο πρόβλημα ο DE σε σύγκριση με παραδοσιακό αλγόριθμο αποθήκευσης παρουσιάζει καλύτερα αποτελέσματα σε όσο αφορά την αντικειμενική του συνάρτηση.

Το **πρόβλημα προγραμματισμού ροής παραγωγής** (Scheduling flow shops) είναι ένα πρόβλημα σχεδιασμού παραγωγής στο οποίο πρέπει να επεξεργαστούν n εργασίες στην ίδια σειρά σε μηχανές m . Οι υποθέσεις είναι ότι δεν υπάρχουν καταστροφές μηχανών και ότι όλες οι θέσεις εργασίας είναι προληπτικές. Αυτό συμβαίνει συνήθως σε πολλά συστήματα παραγωγής όπου οι εργασίες μεταφέρονται από το μηχάνημα σε μηχάνημα με κάποιο είδος αυτόματου συστήματος χειρισμού υλικών. Η παραγωγή μπορεί να διαμορφωθεί γενικά με την αλληλουχία των εργασιών n σε μηχανές m υπό την προϋπόθεση προτεραιότητας, με χαρακτηριστικές αντικειμενικές λειτουργίες να είναι η ελαχιστοποίηση του μέσου όρου χρόνου ροής, ελαχιστοποιώντας τον απαιτούμενο χρόνο για να ολοκληρώσετε όλες τις εργασίες ελαχιστοποιώντας τη μέγιστη καθυστέρηση, και την ελαχιστοποίηση του αριθμού των καθυστερημένων εργασιών. Εάν ο αριθμός των εργασιών είναι σχετικά μικρός, τότε το πρόβλημα μπορεί να λυθεί χωρίς να χρησιμοποιηθεί κανένας γενικός αλγόριθμος βελτιστοποίησης. Όμως, πιο συχνά, η επεξεργασία του αριθμού εργασιών είναι μεγάλη κατά συνέπεια, κάποιο είδος αλγόριθμου να είναι απαραίτητο σε αυτό είδος προβλήματος. Σε συγκεκριμένη εργασία **πρόβλημα**

προγραμματισμού ροής παραγωγής (Scheduling flow shops) using differentialevolution algorithm από τους Godfrey Onwubolu , Donald Davendra το 2004 παρατηρείται από τα αποτελέσματα ότι η προσέγγιση διαφορικής εξέλιξης παρέχει ανταγωνιστικό makespan δηλαδή ελαχιστοποίηση στον χρόνο εργασιών, μέσο χρόνο ροής και σύνολο καθυστέρηση σε σύγκριση με τον γενετικό αλγόριθμο. Η επιτυχής προσέγγιση της διαφορικής εξέλιξης καταδεικνύεται με την αντιμετώπιση του προβλήματος συνόλου δεδομένων. Οι περιπτώσεις δεδομένων δημιουργήθηκαν τυχαία για το χρόνο επεξεργασίας των εργασιών στις μηχανές. Από τα αποτελέσματα, ο αλγόριθμος διαφορικής εξέλιξης έχει βρεθεί να εφαρμόζεται καλύτερα από τον γενετικό αλγόριθμο για μικρού μεγέθους προβλήματα και ανταγωνίζεται αισθητά με γενετικό αλγόριθμο για προβλήματα μέσου έως μεγάλου μεγέθους.

4.4 Ερευνητικά ερωτήματα

Παρακάτω θέτουμε μερικά ερωτήματα τα οποία θα επιχειρήσουμε να απαντήσουμε μετά την εφαρμογή και ανάλυση του αλγόριθμου Διαφορικής Εξέλιξης στο υπό μελέτη πρόβλημα.

- Είναι κατάλληλος ο αλγόριθμος Διαφορικής Εξέλιξης για την επίλυση του προβλήματος του βέλτιστου σχεδιασμού γραμμής προϊόντων;
- Με ποιον τρόπο γίνεται αναπαράσταση των λύσεων;
- Ποιες τροποποιήσεις πρέπει να γίνουν ώστε να μπορεί ο αλγόριθμος διαφορικής εξέλιξης να λύσει ένα διακριτό πρόβλημα όπως αυτό του βέλτιστου σχεδιασμού γραμμής προϊόντων;
- Είναι αρκετά ανταγωνιστικός ο αλγόριθμος ώστε να συγκριθεί με τις υπόλοιπες μεθόδους που έχουν ήδη εφαρμοστεί στο συγκεκριμένο πρόβλημα
- Ποια στρατηγική Διαφορικής Εξέλιξης ικανοποιεί το σκοπό του υπό μελέτη προβλήματος;

- Ποιο είναι το κατάλληλο αρχικό μέγεθος πληθυσμού που πρέπει να οριστεί έτσι ώστε ο αλγόριθμος να είναι αξιόπιστος και να βρίσκει μια ‘καλή λύση’ ;
- Ποια είναι η κατάλληλη τιμή της παραμέτρου παράγοντα κανονικοποίησης της εξίσωσης μετάλλαξης F και ποια για τον τελεστή διασταύρωσης C_r ;
- Είναι αναγκαία η εφαρμογή τοπικής αναζήτησης (local search) για την εύρεση καλύτερου αποτελέσματος;

5. Εφαρμογή Αλγορίθμου Διαφορικής Εξέλιξης στο πρόβλημα βελτιστοποίησης γραμμής προϊόντων

Σε αυτή την ενότητα περιγράφετε η προσαρμογή του αλγορίθμου της Διαφορικής Εξέλιξης, όπως το περιγράψαμε παραπάνω, ούτως ώστε να επιλυθεί το πρόβλημα βέλτιστου σχεδιασμού γραμμής προϊόντων. Θα δοκιμάσουμε όλες τις εξισώσεις μετάλλαξης με διαφορετικές παραμέτρους ούτως ώστε να καταλήξουμε στην καλύτερη μορφή του κώδικα. Μετά από τις πειραματικές δοκιμές ο αλγόριθμος θα προσαρμοστεί στα δεδομένα και στους περιορισμούς του προβλήματος για εύρεση καλής ποιότητας λύσεων σε λογικό χρόνο.

5.1 Περιγραφή προτεινόμενου μοντέλου

Σκοπός της έρευνας όπως αναφέραμε, είναι η βελτιστοποίηση του κέρδους της εταιρείας εισάγοντας στην αγορά μια γραμμή πέντε σακιδίων, έχοντας ως ανταγωνισμό 3 ήδη γνωστά σακίδια. Μετά από μελέτη των ερευνητών έχουμε δεδομένο τις προτιμήσεις των καταναλωτών πάνω στα δέκα χαρακτηριστικά από τα οποία αποτελείται το κάθε σακίδιο, καθώς και το κόστος του κάθε χαρακτηριστικού. Τα σακίδια αναλύονται σε 10 χαρακτηριστικά ,καθένα από αυτά έχει δύο διακριτά επίπεδα (χερούλι, εναλλακτικό χρώμα, θήκη PDA κτλ.) που έχουν ως απάντηση δυο τιμές (ναι / όχι) πέραν του χαρακτηριστικού τιμή που έχει 7 διακριτά επίπεδα (70\$ 75\$ 80\$ 85\$ 90\$ 95\$ 100\$) .Χρήζει προσοχής ότι ο συνδυασμός των χαρακτηριστικών μπορεί να δημιουργήσει 3584 διαφορετικά προϊόντα από τα οποία προκύπτουν $4,9 \times 10^{15}$ διαφορετικές γραμμές από 5 προϊόντα.

Το καθαρό κέρδος από την πώληση μια τσάντας προέρχεται από την τιμή πώλησης του, μείον το κόστος των χαρακτηριστικών που και το σταθερό κόστος κατασκευής που ισούται με 35\$. Επιπλέον για κάθε, για κάθε χαρακτηριστικό έχουμε την δυνατότητα υπολογίσουμε τη συνολική χρησιμότητα για όλους τους ερωτώμενους που συμμετείχαν στην έρευνα. Επίσης για κάθε ερωτώμενο μπορούμε να εντοπίσουμε την τσάντα που προτιμά από την δική μας γραμμή προϊόντων και από την ανταγωνιστική. Στη συνέχεια συγκρίνουμε τις δύο αντίστοιχες χρησιμότητες και μπορούμε να καταλήξουμε στην τελική επιλογή του ερωτώμενου (μοντέλο πρώτης επιλογής / maximum utility). Σε τελικό στάδιο , η αξιολόγηση της λύσης συνεπάγεται από το συνολικό κέρδος της γραμμής προϊόντων

.Αυτό το βρίσκουμε αθροίζοντας το κέρδος από την κάθε πώληση της τσάντας που απαρτίζει την γραμμή.

Η αντικειμενική συνάρτηση για την επίλυση του προβλήματος περιλαμβάνει έξυπνες μεθόδους που προ υπολογίζουν και αποθηκεύουν σε πίνακες ορισμένα μεγέθη ώστε να μην χρειάζονται πολύπλοκοι συνεχόμενοι υπολογισμοί την ώρα της επίλυσης του προβλήματος. Παρακάτω αναφέρονται οι μέθοδοι που αποθηκεύουν τα απαραίτητα δεδομένα που χρειάζονται για την υλοποίηση του προβλήματος:

- Η μέθοδος **load_data** αποθηκεύει δεδομένα στους 3 παρακάτω πίνακες .
- Στον πίνακα **profits** , αποθηκεύει το οριακό κόστος από την προσθήκη του κάθε χαρακτηριστικού όπως έχουμε ήδη αναφέρει παραπάνω. Λογικό είναι πως μόνο η τιμή έχει θετική τιμή profit αφού μόνο αυτή αποφέρει έσοδα.
- Στον πίνακα **worths** αποθηκεύει την μερική αξία του κάθε χαρακτηριστικού του προϊόντος για τον κάθε ερωτώμενο από τους 324, όπως έχουμε ήδη αναφέρει .
- Στον πίνακα **bags** με διαστάσεις 1*3584 στον οποίο δημιουργούνται όλες οι πιθανές τσάντες που μπορούν να σχηματιστούν από τα χαρακτηριστικά ,και κάθε μια αναπαριστάται σαν ένα διάνυσμα 10 χαρακτηριστικών. Με αυτό τον τρόπο η θέση κάθε τσάντας εντοπίζεται στον συγκεκριμένο πίνακα .
- Στον πίνακα **bag_profits** υπολογίζεται και αποθηκεύεται το οριακό κέρδος κάθε τσάντας.
- Στον πίνακα **coop_bags** είναι αποθηκευμένα τα ανταγωνιστικά προϊόντα (σαν δείκτες του πίνακα bags).
- Επίσης υπολογίζεται ο πίνακας **bag_worths** που αποθηκεύει την χρησιμότητα κάθε προϊόντος για κάθε έναν από τους ερωτώμενους με διαστάσεις 324*3584,
- ο πίνακας **base_util** στον οποίο διατηρείται η χρησιμότητα του προτιμότερου προϊόντος από την ανταγωνιστική γραμμή για τον ερωτώμενο. Ο συγκεκριμένος πίνακας θα υπολογιστεί μια φορά μιας και η ανταγωνιστική γραμμή είναι σταθερή .
- Επιπλέον υπολογίζεται ο πίνακας **util** στον οποίο διατηρείται η χρησιμότητα του προτιμότερου προϊόντος από τη δική μας γραμμή για κάθε ερωτώμενο.

- Τελευταίος πίνακας είναι ο **getone** στον οποίο καταγράφονται η χρησιμότητες των προϊόντων ανάλογα με το ποια γραμμή προτιμά ο ερωτώμενος. Αυτό υπολογίζεται ελέγχοντας την συνθήκη **util>base_util**.
- Τέλος υπολογίζεται το κέρδος της γραμμής προϊόντων.

5.2 Προσαρμογή αλγορίθμου στο πρόβλημα

Ο αλγόριθμος Διαφορικής Εξέλιξης που εφαρμόσαμε στο συγκεκριμένο πρόβλημα πρώτα αρχικοποιεί τους πληθυσμούς που θα χρησιμοποιηθούν : τον γονέα , τον πληθυσμό μετάλλαξης και τον πληθυσμό που θα δημιουργηθεί μετά την διασταύρωση του γονέα και αυτού της μετάλλαξης. Επίσης αρχικοποιεί το μέγεθος του πληθυσμού μας και τις τιμές του παράγοντα μετάλλαξης και του τελεστή διασταύρωσης. Όπως θα δούμε στη συνέχεια οι τιμές που χρησιμοποιήθηκαν συχνότερα και παρουσίασαν τα καλύτερα αποτελέσματα μετά από πολλές δοκιμές ήταν για $FE \in [0.3, 0.7]$ και για $CR = 0.2$ και για $CR \in [0.8, 0.9]$.

Στη συνέχεια δημιουργεί ένα τυχαίο πληθυσμό με αριθμούς από 1 μέχρι 3584 όπου είναι όλα τα προϊόντα που μπορούν να δημιουργηθούν από τους ανασυνδυασμούς όλων των χαρακτηριστικών, όπου η κάθε γραμμή του πληθυσμού αντικατοπτρίζει μια σειρά 5 προϊόντων. Μετέπειτα θα τρέξουμε την αντικειμενική συνάρτηση για να πάρουμε τον γονέα με τα κέρδη που αποφέρει κάθε μια σειρά προϊόντων. Μετά θα ταξινομήσουμε τις σειρές προϊόντων βάση των κερδών από την μικρότερη στη μεγαλύτερη.

Στην επόμενη φάση μπαίνουμε στην κυρίως επανάληψη που θα τρέξει για όσες φορές θα δηλώσουμε . Με την εξίσωση μετάλλαξης (mutation) θα δημιουργήσουμε ένα καινούργιο πληθυσμό σωματιδίων .Αργότερα εφαρμόζεται η λειτουργία διασταύρωσης (crossover) προκειμένου να ενισχυθεί η δυναμική πολυμορφία του πληθυσμού. Ο απόγονος που θα φτιαχτεί θα είναι μια ανάμειξη από σειρές προϊόντων του γονέα και του πληθυσμού μετάλλαξης με τον τρόπο που αναφέραμε στο προηγούμενο κεφάλαιο. Μετά θα τρέξουμε την αντικειμενική συνάρτηση για τον απόγονο. Τέλος , θα ενώσουμε τους δύο πληθυσμούς και με ταξινόμηση των σειρών βάση των κερδών θα πάρουμε σειρές προϊόντων με τα μεγαλύτερα κέρδη όσες είναι ο πληθυσμός που είχαμε αρχικοποιήσει . Οι επαναλήψεις θα συνεχίζονται κάθε φορά με τα καλύτερα σωματίδια ώστε να καταλήξουμε σε μια βέλτιστη λύση όταν ο αλγόριθμος θα τερματίσει.

5.2.1 Αναπαράσταση λύσεων

Ο αλγόριθμος Διαφορικής Εξέλιξης είναι ένας αλγόριθμος βελτιστοποίησης για συνεχή προβλήματα σε αντίθεση το πρόβλημα του βέλτιστου σχεδιασμού γραμμής προϊόντων που είναι διακριτό. Κάθε φορά που ο αλγόριθμος μας πρέπει να αξιολογήσει την ποιότητα μίας

λύσης, θα πρέπει πρώτα να γίνεται μετατροπή των τιμών από συνεχείς σε διακριτές. Αυτό το πετυχαίνουμε με την εντολή `round` της Matlab.

Για παράδειγμα:

Συνεχείς τιμές λύσης	Μετατροπή σε Διακριτές τιμές
2094.32	2094
512.023	512
3366.4	3366
195.533	196
1860.88	1861

Το πεδίο ορισμού της αντικειμενικής συνάρτησης είναι τιμές από το 1 μέχρι το 3584 (όλες οι πιθανές τσάντες). Εάν κάποιο σωματίδιο ξεπεράσει αυτά τα άκρα, τότε ως τιμή του θεωρούνται αυτές οι τιμές. Π.χ εάν κάποιο σωματίδιο πάρει την τιμή 3600, τότε θα γίνει αυτόματα 3584. Αντίστοιχα για τιμές μικρότερες του 1, θα γίνει 1.

5.3 Αποτελέσματα

Προχωράμε σε πειραματικές δοκιμές των παραλλαγών του αλγορίθμου. Θα δοκιμάσουμε διάφορες τιμές των παραμέτρων, των επαναλήψεων και του πληθυσμού. Καταγράφουμε κάθε φορά τον μέσο όρο (average), τον μεγαλύτερο (max), τον διάμεσο (median), την τυπική απόκλιση του πληθυσμού (stddev) μετά από 50 δοκιμές κάθε τροποποιημένης μορφής του αλγορίθμου.

Η παράμετρος F για κάθε στρατηγική παίρνει τυχαίες τιμές σε κάθε επανάληψη στο διάστημα $[0.3, 0.7]$ όπου βάση της βιβλιογραφίας και από πειραματικές δοκιμές που εφαρμόσαμε στο συγκεκριμένο πρόβλημα δίνουν τις περισσότερο βέλτιστες λύσεις. Ξέρουμε από το άρθρο του Belloni et al. (2008) ότι το ολικό βέλτιστο είναι 12226.

Πίνακας 1: DE/rand/1/bin

Iterations	Population Size	CR	Max	Median	Stdev	Average earnings	Average time
5000	50	0.8	10875.5	10382	224.9	10421.3	55 s
5000	50	0.2	11714	10681	494	10690	57 s
2000	75	0.8	10953	10287.5	262.1	10319.4	30 s
2000	75	0.9	11752.5	10468.2	468.7	10552.6	30 s
2000	75	0.2	11638	10523	346.6	10609	33 s
1000	100	0.8	11076	10174	244.3	10244.9	42 s
1000	150	0.8	10954	10367	290	10352	50 s
1000	500	0.2	11616.5	10942.2	267.6	10991	42 s

Πίνακας 2: DE/best/1/bin

Iterations	Population Size	CR	Max	Median	Stdev	Average earnings	Average time
3000	75	0.9	11078	10036	508.873	10170.9	40 s

2000	100	0.9	11078	10053	501.8	10105.8	45 s
2000	75	0.8	11116.5	10166.7	514.13	10198.5	37 s
1000	100	0.8	11167	9870	548.75	9911.9	35 s
1000	150	0.8	11503.5	10140.5	559.41	10276.2	55 s
1000	500	0.2	11471	10850	423.8	10767.6	68 s

Πίνακας 3:DE/rand-to-best/1/bin

Iterations	Population Size	CR	Max	Median	Stdev	Average earnings	Average time
2000	75	0.8	12224	11347.5	405.268	11304.34	25 s
2000	75	0.9	12035	11402.5	467.734	11359.9	25 s
2000	75	0.6	11931	11278	352.6605	11300.06	25 s
2000	75	0.2	11767	11026.4	547.7	11024	26 s
3000	75	0.2	11857	11112	474	10973	33 s
1000	100	0.8	11578	10754	529.6	10723	37 s
2000	100	0.2	11763	11083	350	11133	36 s
1000	150	0.8	11887	11369	331.02	11345.4	24 s

1000	150	0.2	12084	11460.7	390	11376	26 s
2000	250	0.2	11865	11569	233.8	11586	35 s
1500	300	0.2	12053	11669.7	182	11651.3	29 s
1500	300	0.4	12084	11598.2	312.6	11534	29 s
2000	300	0.2	12002	11666.2	165.3	11656	38 s
1500	350	0.2	11967	11729.2	175	11672.5	33 s
1000	450	0.2	11949	11717.7	179	11696	28 s
1000	500	0.2	11910	11730	253.1	11678	37 s
1000	500	0.9	11769	11177	601.9	11010	33 s
2000	500	0.2	12096	11775	187.5	11756	68 s
1000	750	0.2	11990	11829	133.5	11805	58 s
500	1000	0.2	11867	11225.5	273.8	11251.75	58 s

Πίνακας 4: DE/rand/2/bin

Iterations	Population Size	CR	Max	Median	Stdev	Average earnings	Average time
------------	-----------------	----	-----	--------	-------	------------------	--------------

2000	75	0.9	10596	9784	315	9807.39	40 s
3000	50	0.8	10419	9906.5	262	9983	50 s
2000	100	0.8	10903	9953	296	10060	53 s
2000	100	0.9	11068.5	10099	369	10150	55 s
1000	150	0.8	11068.5	1078	314	10077	46 s

Πίνακας 5:DE/best/2/bin

Iterations	Population Size	CR	Max	Median	Stdev	Average earnings	Average time
3000	75	0.9	10695	10726	353	10695	56 s
2000	75	0.9	11342.5	10800.5	328.4	10719	40 s
2000	75	0.8	1112.5	10679	247.5	10679	37 s
2000	100	0.9	10558.7	10659	218	10559	45 s
1000	100	0.9	11342.5	10781.25	343	10680	35 s

Τοπική Αναζήτηση (Local Search)

Στην προσπάθεια βελτίωσης των αποτελεσμάτων μας, μετά το πέρας κάθε επανάληψης του DE καταλήγουμε στην καλύτερη γραμμή προϊόντων που έχει εντοπιστεί μέχρι στιγμής από τον αλγόριθμο. Τότε εφαρμόζουμε μέθοδο τοπικής αναζήτησης όπου αλλάζουμε τυχαία ένα προϊόν της γραμμής με ένα άλλο που βρίσκεται 512 θέσεις μετά από αυτό. Με αυτόν τον τρόπο γίνεται αλλαγή ενός μόνο χαρακτηριστικού του συγκεκριμένου προϊόντος. Αν η γραμμή προϊόντων με το καινούργιο προϊόν αποφέρει περισσότερα κέρδη τότε κρατάμε αυτή . Αν όχι τότε συνεχίζει με την ήδη υπάρχουσα βέλτιστη λύση που έχει εντοπιστεί.

Πίνακας 6: Εφαρμογή τοπικής αναζήτησης (local search) στα καλύτερα αποτελέσματα του DE/rand-to-best/1/bin

Iterations	Population Size	CR	Max	Median	Stdev	Average earnings	Average time
1000	750	0.2	12172.5	11882.5	128.25	11878.37	46 s
1500	500	0.2	12043.5	11800	141.5	11803,88	47 s

Εκτελούμε όλες τις μεθόδους οι οποίες προτάθηκαν από τους Belloni et al.(2008) 50 φορές και βλέπουμε το μέσο όρο των κερδών, το μέγιστο κέρδος που έχει καταγραφεί καθώς και το μέσο χρόνο εκτέλεσης , όπως φαίνεται στον πίνακα 7. Η παραμετροποίηση των αλγορίθμων παρέμεινε όπως είχε προταθεί από αυτούς:

Πίνακας 7: Αποτελέσματα αλγορίθμων

Method		Max	Average earnings	Average time
Coordinate Ascent (1-Opt)	As-	12175	11451.9	0.2 s
Coordinate Ascent (2-Opt)	As-	12226	11950.8	2 s
Coordinate Ascent (3-Opt)	As-	12226	12044.6	165 s
Genetic		12226	12125.8	10 s
Simulated Annealing	An-	12226	12221.8	62 s
Divide and Conquer		12226	12182	4 s
Beam Search		12132	11293.98	1 s
DP Heuristic		12016	11556.7	3 s
Nested Partitions		12034.5	11846.24	4 s
Greedy Heuristic		12034.5	1234.5	2 s
Product Swapping	Swap-	12226	12212.13	6 s
DE(local search)		12172.5	11878.37	46 s

5.3 Δοκιμασία Ευρωστίας – Robustness Testing

Η φερεγγυότητα των αποτελεσμάτων εκτός από την μέθοδο που χρησιμοποιείται για την επίλυση του συγκεκριμένου προβλήματος, συνδέεται και με άλλους παράγοντες. Από τους πιο σημαντικούς είναι η ακρίβεια ή η εσφαλμένη, σύμφωνα με την ανάλυση συζυγιών

(conjoint analysis) , εκτίμηση για τις προτιμήσεις των καταναλωτών και η εκτίμηση των μερικών αξιών που αποδίδουν στο κάθε χαρακτηριστικό . Μολονότι υπάρχουν στατιστικές τεχνικές υπεύθυνες για την μέτρηση της απόκλισης των πραγματικών προτιμήσεων των καταναλωτών για την βελτίωση του δείγματος,ο στόχος αυτής της ενότητας είναι ο εντοπισμός της ευαισθησίας των μεθόδων επίλυσης η πιο συγκεκριμένα το κατά πόσο επηρεάζονται από την ύπαρξη σφαλμάτων κατά τον προσδιορισμό των μερικών αξιών (part – worths).

Για τον σκοπό αυτό κατασκευάσαμε ένα δείγμα σύγχυσης (perturbed dataset) , το οποίο διαταράσσει το δείγμα των καταναλωτικών προτιμήσεων που αναφέρθηκε προηγουμένως Το προγενέστερο δείγμα θεωρείται ότι είναι η εκτίμηση που κάναμε με ένα σφάλμα εκτίμησης ίσο με ϵ .

$$u'_{i,j} = u_{i,j} + \epsilon_{i,j} \quad (5.1)$$

Όπου, $u_{i,j}$ είναι η πραγματική προτίμηση του καταναλωτή i για το χαρακτηριστικό j , $\epsilon_{i,j}$ το σφάλμα μέτρησης όπου είναι μια ανεξάρτητη , όχι σταθερή μεταβλητή , που ακολουθεί την κανονική κατανομή με μέσο όρο το 0 και $u'_{i,j}$ η διαταραγμένη προτίμηση του καταναλωτή i για το χαρακτηριστικό j .

Η τυπική απόκλιση των διαταραχών έγινε με τη χρήση του τυπικού σφάλματος για το αντίστοιχο $u_{i,j}$ κατά την εκτίμηση με την μέθοδο των ελαχίστων τετραγώνων.

Τα τυπικά σφάλματα είχαν μέσο όρο το 55% της απόλυτης τιμής της μερικής αξίας (part-worth).

Θα εκτελέσουμε το πείραμα 100 φορές , με 100 διαφορετικά δείγματα σύγχυσης. Καταγράφουμε μέσο όρο των κερδών, το μέγιστο κέρδος που έχει καταγραφεί καθώς και το μέσο χρόνο εκτέλεσης ,όπως φαίνεται στον παρακάτω πίνακα :

Πίνακας 8: Αποτελέσματα αλγορίθμων

Method		Max	Average earnings	Average time
Coordinate Ascent (1-Opt)	As-	12042	10807	0.2 s
Coordinate Ascent (2-Opt)	As-	11956	11423.4	2 s
Coordinate Ascent (3-Opt)	As-	12096.5	11655.6	56 s
Genetic		12077	11635	7 s
Simulated Annealing	An-	12189	11783.8	55 s
Divide and Conquer		12091	11765.6	4 s
Beam Search		12013	10984.5	1 s
DP Heuristic		12054	11406.61	3 s
Nested Partitions		12057.5	11371.3	4 s
Greedy Heuristic		12057.5	11659.4	2 s
Product Swapping	Swap-	12189	11772.5	5 s
DE (local search)		11950	11405	56 s

5.4 Γενίκευση του Μοντέλου

Θα αξιολογήσουμε την γενίκευση της μεθόδου που αναπτύξαμε για την επίλυση του προβλήματος βέλτιστου σχεδιασμού γραμμής προϊόντων. Αυτό που θα κάνουμε είναι ότι θα αλλάζουμε το μέγεθος του προβλήματος και θα δοκιμάζουμε τον αλγόριθμο μετρώντας τις επιδόσεις του κάθε φορά. Συγκεκριμένα θα τροποποιήσουμε τον πιθανό αριθμό προϊόντων της γραμμής, τον πιθανό αριθμό των χαρακτηριστικών που μπορεί να πάρει το

καθένα και τα επίπεδα του κάθε χαρακτηριστικού. Επίσης θα αλλάξει και ο αριθμός καταναλωτών. Με την αλλαγή μεγέθους του προβλήματος αλλάζουν επίσης όλα τα πιθανά προϊόντα και κατ'επέκταση όλες οι πιθανές γραμμές προϊόντων, ο αριθμός των οποίων καθορίζει την πολυπλοκότητα του προβλήματος.

Θα δημιουργηθούν 12 μεγέθη προβλήματος όπου το μέγεθος γραμμής θα είναι 3 ή 4 προϊόντα, τα χαρακτηριστικά των προϊόντων 3, 5 ή 7, τα επίπεδα των χαρακτηριστικών 2,3,5 ή 8 και αριθμός καταναλωτών 50 ή 100

Πίνακας 9: Μεγέθη προβλήματος

ΠΕΛΑΤΕΣ	ΧΑΡΑΚΤΗΡΙΣΤΗΚΑ	ΕΠΙΠΕΔΑ	ΠΡΟΙΟΝΤΑ
50	3	5	4
100	3	5	4
50	5	3	4
100	5	3	4
50	7	2	4
100	7	2	4
50	3	8	3
100	3	8	3
50	5	5	3
100	5	5	3
50	7	3	3
100	7	3	3

Βάση της μελέτης των ερευνητών Alexandre Beloni, Robert Freund, Matthew Selove, Duncan Simester το ολικό βέλτιστο για κάθε μέγεθος του προβλήματος έχει βρεθεί με την μέθοδο Langarian Relaxation with Brand and Bound. Ο Simulated Annealing για κάθε παραλλαγή του προβλήματος είναι απόλυτα ακριβής αφού πετυχαίνει ποσοστό 100% .

Για να μετρήσουμε την ακρίβεια του αλγόριθμου Διαφορικής Εξέλιξης για το συγκεκριμένο πρόβλημα εκτελούμε τον Simulated Annealing ο οποίος βρίσκει το ολικό βέλτιστο και στη συνέχεια συγκρίνουμε τα αποτελέσματα των δύο μεταξύ τους. Επίσης

εκτελούμε και τις υπόλοιπες μεθόδους για να συγκρίνουμε τον αλγόριθμο μας. Το ποσοστό ακρίβειας καθώς και ο χρόνος εκτέλεσης κάθε μεθόδου καταγράφεται στον παρακάτω πίνακα:

Πίνακας 10: Αποτελέσματα Αλγορίθμων

Method	Average performance	Average time
Simulated Annealing	100,0%	23 s
Genetic	99,9%	3 s
Beam Search	98,7%	0.1 s
Divide and Conquer	98,6%	0.2 s
Product Swapping	98,0%	0.8 s
DE (local search)	97,8%	34 s
Greedy Heuristic	97,2%	0.2 s
Coordinate Ascent (3-Opt)	96,4%	10 s
DP Heuristic	95,6%	0.7 s
Coordinate Ascent (2-Opt)	95,5%	1 s
Coordinate Ascent (1-Opt)	94,3%	0.1 s
Nested Partitions	93,9%	1.3 s

Ο αλγόριθμος Διαφορικής Εξέλιξης έχει φτάσει στο 97,8% την ακρίβεια του Simulated Annealing με μεγαλύτερο χρόνο εκτέλεσης. Βρίσκει καλύτερο ποσοστό από τους Coordinate Ascent (1-Opt), Coordinate Ascent (2-Opt), Coordinate Ascent (3-Opt), DP Heuristic,

Nested Partitions και Greedy Heuristic αλλά πάλι χρειάζεται περισσότερο χρόνο ώστε να το καταφέρει αυτό.

6. Επίλογος

6.1 Παρατηρήσεις

- Η μόνη στρατηγική που πλησιάζει το ολικό βέλτιστο και βρίσκει ‘καλές λύσεις’ είναι η *DE/rand-to-best/1/bin* όπου έγινε και περαιτέρω ανάλυση.
- Το μεγαλύτερο κέρδος 1224 έχει καταγραφεί για πληθυσμό 75, $FE [0.3, 0.7]$, $Cr = 0.8$ και επαναλήψεις 2000. Παρατηρούμε όμως για τις 50 φορές που έτρεξε ο συγκεκριμένος αλγόριθμος ο μέσος όρος είναι μικρός σε σχέση με το βέλτιστο που βρήκε. Αυτό μας δείχνει ότι δεν είναι τόσο αξιόπιστος ο αλγόριθμος με αυτές τις τιμές των παραμέτρων αφού δεν εγγυάται ‘καλές λύσεις’ όποτε εφαρμοστεί.
- Έχει παρατηρηθεί όμως για την στρατηγική *DE/rand-to-best/1/bin* , πως για μεγάλους πληθυσμούς, από 250 μέχρι 1000 και μικρότερο $Cr=0.2$, βρίσκει μεγαλύτερο μέσο όρο έχοντας κοντέψει αρκετά πάλι στο βέλτιστο. Αυτό δείχνει μεγαλύτερη αξιοπιστία του αλγορίθμου με τις συγκεκριμένες τιμές των παραμέτρων.
- Οι καλύτερες επιδόσεις που παρατηρήθηκαν και καταγράφηκαν για μικρούς αρχικούς πληθυσμούς από 50 μέχρι 100 ήταν για Cr από 0.8 και 0.9.
- Οι καλύτερες επιδόσεις του αλγορίθμου έχουν καταγραφεί με τυχαίες τιμές $FE [0.3, 0.7]$ για κάθε επανάληψη.
- Η εφαρμογή τοπικής αναζήτησης(local search) στα καλύτερα αποτελέσματα που βρεθήκανε βοήθησε σε μικρό βαθμό την αύξησή τους.

6.2 Συμπέρασμα

Η παρούσα εργασία παρέχει μια ολοκληρωμένη λύση σε ένα συγκεκριμένο πρόβλημα του Μάρκετινγκ. Ο αλγόριθμος Διαφορικής Εξέλιξης (DE) δεν καταφέρνει να προσαρμοστεί επαρκώς στο πρόβλημα του βέλτιστου σχεδιασμού γραμμής προϊόντων όσο οι άλλοι μέθοδοι οι οποίοι παρουσιάστηκαν. Σε γενικές γραμμές όμως παρουσιάζει μια καλή εικόνα με ικανοποιητικά αποτελέσματα.

Ο αλγόριθμος προσαρμόζεται στις ιδιαιτερότητες του προβλήματος όπως στο να μετατρέπει τις συνεχείς τιμές σε διακριτές ώστε να καταφέρει να αξιολογήσει την ποιότητα λύσης. Βασικό ρόλο στην υλοποίηση και στην καταμέτρηση των αποτελεσμάτων παίζουν οι τιμές των παραμέτρων οι οποίες θα θέσουμε για το συγκεκριμένο πρόβλημα. Ο αλγόριθμος Διαφορικής Εξέλιξης (DE) δεν είναι καθολικός. Ανάλογα με το πρόβλημα είναι απαραίτητο να αλλάζουν και οι παράμετροι του αλγορίθμου έτσι ώστε να καθίσταται αποτελεσματικός. Με λάθος τιμές, μπορεί να καταλήξει σε ανεπιθύμητα αποτελέσματα. Έχοντας κάνει πολλές πειραματικές δοκιμές, καταφέραμε να καταλήξουμε στις παραμέτρους που ικανοποιούν πλήρως το συγκεκριμένο πρόβλημα.

Με την παρούσα έρευνα καταλήξαμε ότι ο αλγόριθμος Διαφορικής Εξέλιξης (DE) είναι αξιόπιστος για αρχικούς πληθυσμούς με οριακές τιμές από 400 μέχρι 750, τελεστή διασταύρωσης $Cr=0.2$ και τυχαίες τιμές του παράγοντα ικανοποίησης $F \in [0.3, 0.7]$ σε κάθε επανάληψη. Ο χρόνος που χρειάζεται ο αλγόριθμος για εύρεση μιας καλής ποιότητας λύσης κυμαίνεται χρονικά από 40 μέχρι 55 δευτερόλεπτα. Η εφαρμογή του τοπικής αναζήτησης (local search) βοήθησε στη βελτίωση των αποτελεσμάτων που είχαν βρεθεί.

Συγκριτικά με τους αλγορίθμους Beam Search, Nested Partitions, Coordinate Ascent (one-opt) και Dynamic Programming Heuristic που έχουν ήδη παρουσιαστεί, συμπεραίνουμε ότι η Διαφορική Εξέλιξη (DE) αποφέρει μεν υψηλότερο κέρδος αλλά με την συνύπαρξη μεγαλύτερου χρόνου εκτέλεσης.

Βιβλιογραφία

- Belloni, A., Freund, R., Selove, M. and Simester, D. (2008). Optimizing Product Line Designs: Efficient Methods and Comparisons. *Management Science*, 54(9), pp.1544-1552.
- Das, S. and Suganthan, P. (2011). Differential Evolution: A Survey of the State-of-the-Art. *IEEE Transactions on Evolutionary Computation*, 15(1), pp.4-31.
- En.wikipedia.org. (2019). *Approximation algorithm*. [online] Available at: https://en.wikipedia.org/wiki/Approximation_algorithm [Accessed 17 Jan. 2019].
- Green, P. and Krieger, A. (1985). Models and Heuristics for Product Line Selection. *Marketing Science*, 4(1), pp.1-19.
- Green, P. and Srinivasan, V. (1978). Conjoint Analysis in Consumer Research: Issues and Outlook. *Journal of Consumer Research*, 5(2), p.103.
- Kromer, P., Snasel, V., Berhan, E. and Abraham, A. (2013). On the Differential Evolution for Vehicle Routing Problem.
- Leon, M. (2016). *ENHANCING DIFFERENTIAL EVOLUTION ALGORITHM FOR SOLVING CONTINUOUS OPTIMIZATION PROBLEMS*.
- Onwubolu, G. and Davendra, D. (2006). Scheduling flow shops using differential evolution algorithm. *European Journal of Operational Research*, 171(2), pp.674-692.

- Tsafarakis, S., Marinakis, Y. and Matsatsinis, N. (2011). Particle swarm optimization for optimal product line design. *International Journal of Research in Marketing*, 28(1), pp.13-22.
- Zaharie, D. (2009). Influence of crossover on the behavior of Differential Evolution Algorithms. *Applied Soft Computing*, 9(3), pp.1126-1138.
- Zufryden, F. (1983). Course Evaluation and Design Optimization: A Conjoint Analysis-Based Application. *Interfaces*, 13(2), pp.87-94.
- ΔΕΠΟΥΝΤΗ, Ω. (2012). Υβριδικοί αλγόριθμοι διαφορικής εξέλιξης στο πρόβλημα προγραμματισμός συστημάτων παραγωγής συνεχούς ροής. Εφαρμογή με πραγματικά δεδομένα εταιρίας με ευέλικτο σύστημα παραγωγής συνεχούς ροής. Διπλωματική Διατριβή, Σχολή Μηχανικών Παραγωγής και Διοίκησης, Πολυτεχνείο Κρήτης, Χανιά, Ελλάς
- Μαρινάκης, Ι., Μαρινάκη, Μ., Φ. Ματσατσίνης, Ν. and Ζοπουνίδης, Κ. (2011). Μεθευρετικοί και Εξελικτικοί Αλγορίθμοι σε Προβλήματα Διοικητικής Επιστήμης.
- Μπαλαράς, Γ. (2016). «Εφαρμογή του αλγορίθμου Tabu Search για την επίλυση του προβλήματος βέλτιστου σχεδιασμού γραμμής προϊόντων». Μεταπτυχιακή Διατριβή, Σχολή Μηχανικών Παραγωγής και Διοίκησης, Πολυτεχνείο Κρήτης, Χανιά, Ελλάς
- Νταμαδάκη, Β. (2015). Εφαρμογή αλγορίθμων τεχνητού ανοσοποιητικού συστήματος για την επίλυση του προβλήματος του βέλτιστου σχεδιασμού γραμμής προϊόντων. Μεταπτυχιακή Διατριβή, Σχολή Μηχανικών Παραγωγής και Διοίκησης, Πολυτεχνείο Κρήτης, Χανιά, Ελλάς

- Πελώνης Χαράλαμπος (2017). Διπλωματική Εργασία Ανάλυση καταναλωτικών προτιμήσεων σχετικά με τρισδιάστατους (3D) εκτυπωτές. Διπλωματική Διατριβή, Σχολή Μηχανικών Παραγωγής και Διοίκησης, Πολυτεχνείο Κρήτης, Χανιά, Ελλάδα