Αριστοτέλειο Πανεπιστήμιο Θεσσαλονίκης
Πολυτεχνική Σχολή
Τμήμα Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών

Τεχνικές Βελτιστοποίησης

Αξιμιώτης Δημήτρης 10622

Δεκέμβριος 2024

1 Μαθηματική περιγραφή προβλήματος

Σε αυτήν την ενότητα θα μελετήσουμε την εφαρμογή ενός γενετικού αλγορίθμου σε ένα πρόβλημα ελαχιστοποίησης. Στόχος μας είναι να ελαχιστοποιήσουμε τον συνολικό χρόνο διάσχισης αυτοκινήτων σε ένα οδικό δίκτυο.Για να επιλύσουμε το πρόβλημα με γενετικό αλγόριθμο αρχικά χρειαζόμαστε μια συνάρτηση προσαρμογής που σε αυτην την περίπτωση ορίζεται ως το συνολικό άθροισμα των χρόνων ανά δρόμο, δηλαδή

$$g = \sum_{i=1}^{17} T_i(x_i)$$

$$g = \sum_{i=1}^{17} t_i + \sum_{i=1}^{17} a_i \frac{x_i}{1 - \frac{x_i}{c_i}}$$

, καθώς τα t_i είναι σταθερά δεν θα επηρασουν το τελικό χρόνο για διάφορες τιμές του x_i 'άρα θα επιλέξουμε ως τελική συνάρτηση ελαχιστοποίησης την

$$f = \sum_{i=1}^{17} a_i \frac{x_i}{1 - \frac{x_i}{c_i}}$$

Η παραπάνω συνάρτηση θα την ορίσουμε ως fitness function για το πρόβλημα ελαχιστοποίησης.Η διαδικασία που ακολουθούμε μέσω γενετικού αλγορίθμου είναι η δημιουργία χρωμοσωμμάτων ,δηλαδή πιθανών λύσεων που αποτελούν πιθανή λύση για το πρόβλημα ελαχιστοποίησης.Κάθε γενιά (δηλαδή κάθε επανάληψη στον αλγόριθμο) δημιουργεί ένα σετ x_i πλήθους το οποίο το ορίζουμε εμείς,μέσω μετάλλαξης,διασταύρωσης , τυχαίας επιλογής και επιλογής βάση ικανότητας.Ανά γενιά δημιουργούνται καινούριες λύσεις οι οποίες σε βάθος χρόνου ελαχιστοποιούν την fitness function.Η επιλογή ωστόσο αυτών χρωμοσωμμάτων θα πρέπει κάθε φορά να γίνεται μέσω περιορισμών που προκύπτουν από το πρόβλημα.

2 Περιορισμοί

Όπως φαίνετα από το σχήμα του προβλήματος θέλουμε για κάθε κόμβο του οδικού δικτύου , ο ρυθμός των εισερχόρμενων οχημάτων να ισούται με τόν αριθμό που εξέρχονται.Επομένων έχουμε

$$x_1 + x_2 + x_3 + x_4 = 100$$
$$x_4 = x_9 + x_{10}$$
$$x_9 + x_3 + x_8 = x_{11} + x_{12} + x_{13}$$
$$x_2 = x_8 + x_7$$
$$x_1 = x_5 + x_6$$
$$x_{11} + x_{10} = x_{17}$$
$$x_{13} + x_7 + x_6 = x_{15} + x_{14}$$
$$x_5 + x_{14} = x_{16}$$
$$x_{17} + x_{12} + x_{15} + x_{16} = 100$$

Κάθε χρωμόσωμα που προκύπτει θα πρέπει να βρίσκεται εντός περιορισμών ώστε να μπορεί να θεωρηθεί αποδεκτή λύση.Καθώς στο matlab δεν μπορούμε να βάλουμε ισοτικούς περιορισμούς τους μετατρέπουμε σε ανισοτικούς και θέτουμε σαν σφάλμα για την κάθε εξίσωση ένα $e = 0.001$.Δηλαδή θέλουμε $|x_1 + x_2 + x_3 + x_4 - 100| < e$, και παρόμοια και για τις άλλες εξισώσεις.

3 Πράξη μετάλλαξης

Προκείμενου να επιλέξουμε ένα χρωμόσωμα και να δημιουργήσουμε μια νέα γενιά θα πρέπει να εκτελέσουμε μία σειρά πράξεων που έχουν αποδειχτεί ότι μπορούν να οδηγήσουν το πρόβλημα σε βέλτιστη λύση.Μία από αυτές τις πράξεις είναι η πράξη μετάλλαξης (mutation) στην οποία επιλέγουμε ένα δείγμα από την αρχική γενιά να μεταβάλλουμε τα χαρακτηριστικά του να δούμε αν μπορεί να δώσει καλύτερο αποτέλεσμα.Σε αυτό το πρόβλημα δημιουργούμε έναν τυχαίο αριθμό από κανονική κατανομή με διακύμανση σ=0.1.Έχουμε επιλέξει μεσσαία διακύμανση ώστε να αποφεύγουμε την σύγκλιση σε τοπικά βέλτιστα αλλά παράλληλα δεν είναι αρκετά μεγάλη ώστε να αποκλίνουμε πολύ από βέλτιστες λύσεις.Με επιλογή μεγαλύτερης διακύμανσης θα δημιουργούσαμε μεγάλη τυχαιότητα και μπορεί να αποκλίναμε σημαντικά από το ελάχιστο, ενώ με μικρότετη τιμή δικύμανσης δεν θα προκαλούσαμε ισχυρές μεταβολές ώστε να μπορέσουμε να αποφύγουμε τοπικά ελάχιστα.Με αυτό το τρόπο μπορούμε να εξερευνήσουμε ένα μεγαλύτερο χώρο λύσεων οι οποίες μπορούν να οδηγήσουν το σύστημα σε καλύτερη λύση.

4 Πράξη Διασταύρωσης

Ένας δεύτερος τρόπος να δημιουργήσουμε νέα χρωμοσώμματα είναι με την μέθοδο διασταύρωσης (crossover) , με την οποία επιλέγουμε 2 γονείς (2 δείγματα την προηγούμενης γενιάς) .Μπορούμε να πάρουμε είτε την μέση τιμή τους είτε εναλλάξ χαρακτηριστικά ώστε να προκύψει απόγονος με που έχει χαρακτηριστικά και από τους 2.Σε αυτό το πρόβλημα έχουμε επιλέξει την μέση τιμή από 2 υποψήφια δείγματα που προκύπτουν από την πράξη ρουλέτας.Με αυτο το τρόπο δημιουργούνται πάλι νέα δείγματα που προκύπτουν από ικανές λύσεις ώστε να προκύψει το καλύτερο δυνατό αποτέλεσμα.Ορίζουμε νέο απόγονο με την μέση τιμή και όχι με ανάμιξη χαρακτηριστικών καθώς κάτι τέτοιο θα καθυστερήσει τον αλγόριθμο και είναι πιθανό να μην μπορέσει να δημιουργήσει λύσεις που προκύπτουν εντός των περιορισμών.Επέλεξα να διαλέγει γονείς με τυχαίο τρόπο και όχι με ρουλέτα καθώς φαίνεται ότι οδήγησε σε καλύτερες λύσεις.

5 Πράξη Ρουλέτας

Παραπάνω έχουμε ορίσει την fitness function την οποία θέλουμε να ελαχιστοποιήσουμε.Ωστόσο κάθε λύση έχει διαφορετικό αντίκτυπο καθώς δίνει διαφορετική τιμή για την συνάρτηση ελαχιστοποίησης.Αυτό καθιστά κάποιες λύσεις πιο ικανές από κάποιες άλλες.Μπορούμε λοιπόν να ορίσουμε ως συνάρτηση ικανότητας την $1/f$ στην οποία όσο μικρότερη είναι η f τόσο μεγαλύτερη είναι η τιμή της ικανότητας.Μπορούμε λοιπόν να επιλέγουμε γονείς από crossover , δείγματα προς mutation αλλά και πιθανά χρωμοσώμματα για την επόμενη γενιά μέσω της ρουλέτας.Στην ουσία ταξινομούμε όλα τα χρωμοσώμματα βάση της ικανότητας τους και επιλέγουμε τυχαία ένα απο αυτά (προφανώς αυτά με την μεγαλύτερη ικανότητα έχουν μεγαλύτερη πιθνότητα να βρεθπύν στην επόμενη γενιά).

6 Επιλογή επόμενης γενιάς

Προκειμένου να οδηγηθούμε σε μια καλύτερη λύση θα πρέπει να επιλέξουμε επόμενη γενιά με τρόπο τέτοιο ώστε να ελαχιστοποιούμε την συνάρτηση προσαρμογής.Στον αλγόριθμο που έχω υλοποιήσει επιλέγουμε 100 χρωμοσώμματα ανά γενιά απο τα οποία:

-75 προκύπτουν μέσω της πράξης διασταύρωσης

-5 προκύπτουν από την πράξη μετάλλαξης

-10 προκύπτουν από τυχαία επιλογή μέσω ρουλέτας

-10 προκύπτουν επιλέγοντας κάθες φορά τις 10 βέλτιστες λύσεις βάση της συνάρτησης ικανότητας.

Με αυτή την επιλογή μπορούμε να επιλέγουμε βέλτιστες λύσεις για την επόμενη γενιά αλλά και τυχαίες λύσεις που μπορούν να οδηγήσουν σε μιά λύση που αποφεύγει τοπικά ελάχιστα και επιτυγχάνει ολική βελτιστοποίηση σε ένα σύνθετο πρόβλημα με πολλούς περιορισμούς.Πρέπει ωστόσο κάθε γενιά που προκύπτει να πληρεί τους περιορισμούς που έχουμε αναφέρει παραπάνω.Το πλήθος γενεών είναι 10000.

7 Διαγράμματα και παρατηρήσεις

7.1 V=100

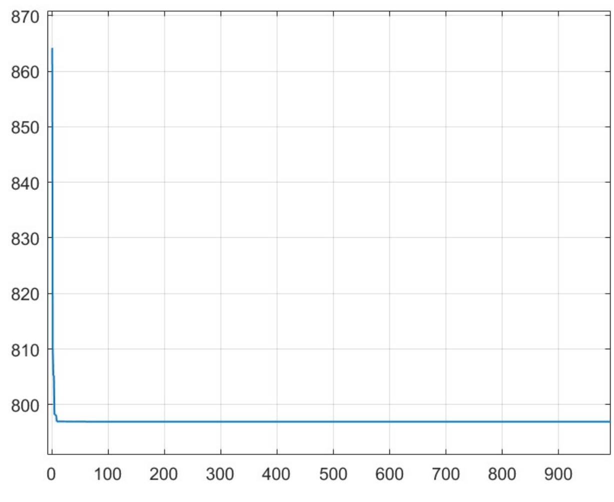


Figure 1: Γραφική παράσταση fitness function f ανα γεννεά με τελική τιμή f=796.927

Βλέπουμε ότι ο αλγόριθμος με το πέρας το γεννεών ελαχιστοποεί την τιμή της αντικειμενικής συνάρτησης φτάνοντας σε τελική τιμή f_{min}=796.927 και διάνυσμα

$$\mathbf{X}=[32.8261 \\ 13.8433 \\ 23.0774 \\ 30.2514 \\ 20.8623 \\ 11.9634 \\ 8.2021 \\ 5.6409 \\ 12.7545 \\ 17.4965 \\ 9.2550 \\ 17.8908 \\ 14.3271 \\ 8.5413 \\ 25.9517 \\ 29.4039 \\ 26.7518]$$

ως βέλτιστη λύση.

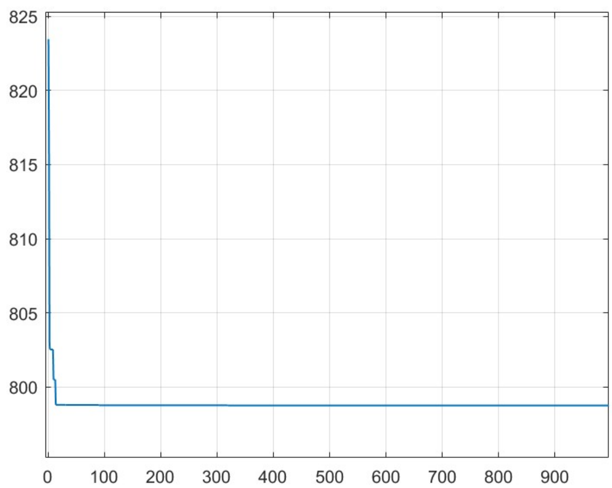


Figure 2: Γραφική παράσταση fitness function f ανα γεννεά με τελική τιμή f=798.7584

Αξίζει να τονίσουμε πως και στις 2 προσομοιώσεις όλα τα χρωμοσώμματα τείνουν να έχουν τα ίδια χαρακτηριστικά,δηλαδή πλησιάζουμε στην βέλτιση λύση οπότε είναι κάτι που το αναμέναμε.Ο λόγος που κάθε φορά δεν προσεγγίζουμε το ίδιο ελάχιστο οφείλεται και στην τυχαίοτητα που εισάγει ο συγκεκριμένος αλγόριθμος.Πιθανόν έχουμε ξεκίνησει αρχική γενεά που δεν μπορεί να οδηγήσει σε βέλτιστη λύση ή λόγω των πράξεων που έχουμε αναφέρει προηγουμένως να χάνουμε υποψήφιες λύσεις που μπορούν να ελαχιστοποιήσουν περαιτέρω την συνάρτηση.Ίσως χρειζόμασταν μεγαλύτερο αριθμό δειγμάτων ώστε να μπορέσουμε να απεγκλωβιστούμε από τοπικές λύσεις.

7.2 Μεταβαλλόμενο V

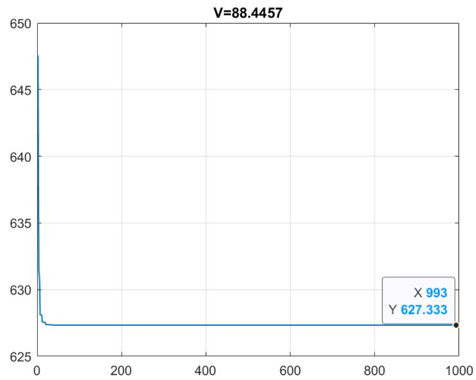


Figure 3

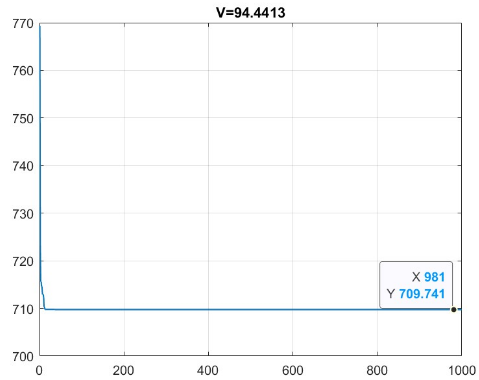


Figure 4

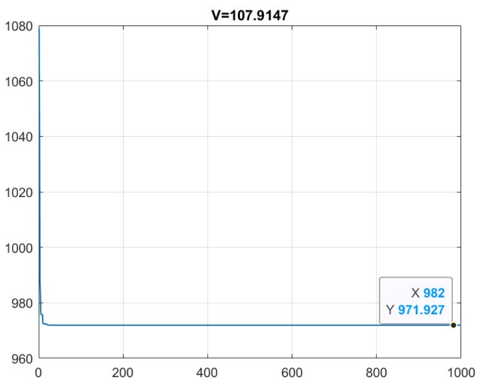


Figure 5

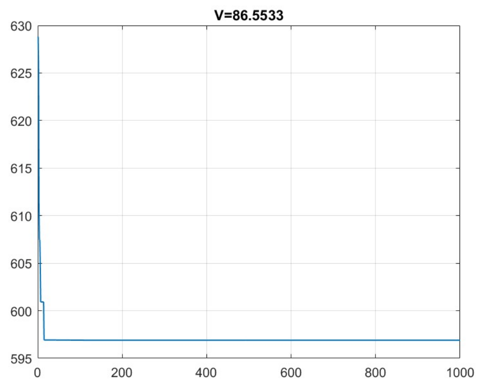


Figure 6

Τα αποτελέσματα είναι αναμενόμενα καθώς όπως παρατηρούμε για αύξηση της κίνησης η βέλτιστη λύση που προκύπτει δίνει αυξημένο ελάχιστο χρόνο διάσχισης που το περιμένουμε αφού ο αριθμός των εισερχόμενων οχημάτων αυξάνεται.Επίσης για μείωση της εισερχόμενης ροής όπως παρατηρούμε μειώνεται και ο συνολικός χρόνος διάσχισης.

7.3 Μέση Βέλτιστη λύση ανά γεννεά

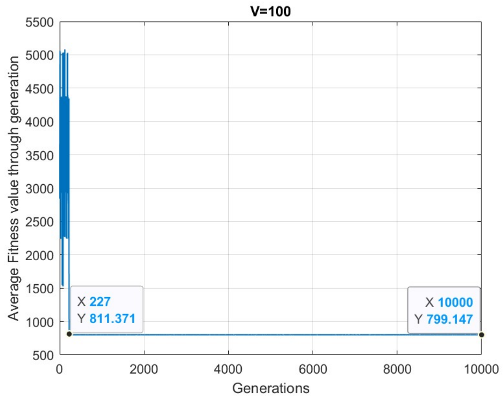


Figure 7

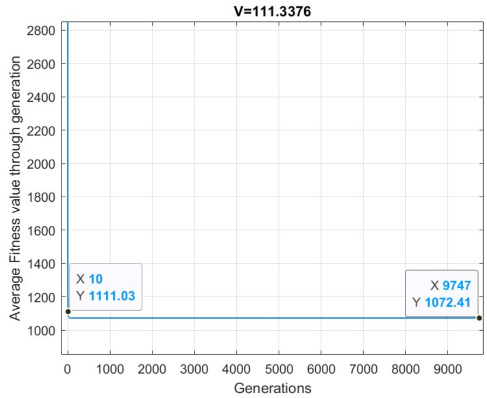


Figure 8

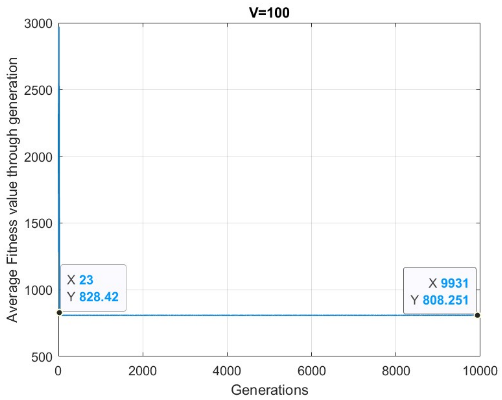


Figure 9

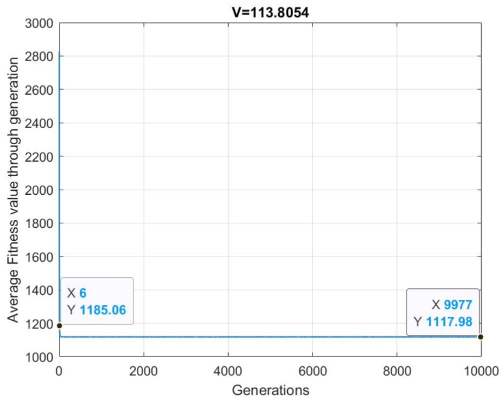
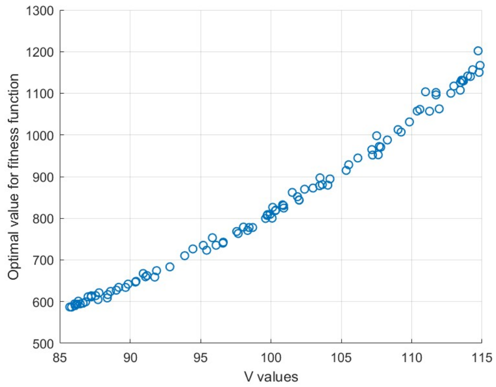


Figure 10

Παρατηρούμε ότι με το πέρας των γεννεών η μέση βέλτιστη λύση τείνει να είναι ίδια με την βέλτιστη λύση του πληθυσμού δηλαδή καταλήγουμε μέσω crossover και mutation σε λύσεις που είναι αρκετά όμοιες μεταξύ τους δηλαδή υπάρχει μια ομογένεια.Αυτό είναι εφικτό ωστόσο εάν επιλέξουμε μεγάλο πλήθος γεννεών.Όπως φαίνεται στο figure 7 για 227 γενιές οι τελικές λύσεις δεν μοιάζουν αρκετά μεταξύ τους επομένως και φαίνεται ότι η μέση τιμή της Objective function για όλο τον πληθυσμό διαφέρει αρκετά από την βέλτιστη λύση.

7.4 Εξάρτηση V με βέλτιστη λύση

Όπως φαίνεται και από τις γραφικές παραστάσεις καθώς το V μεγαλώνει η βέλτιστη τιμή της αντικειμενικής συνάρτησης αυξάνεται.Συνεπώς υπάρχει μια συσχέτιση μεταξύ V και τελικού χρόνου.



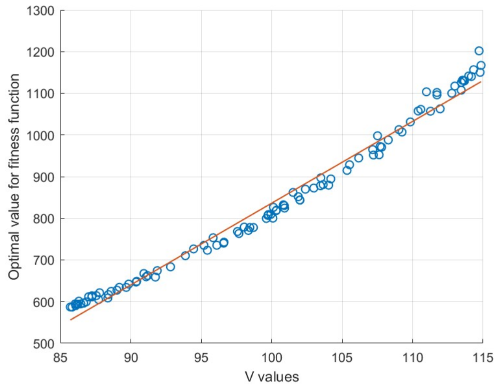


Figure 11: Γραφική παράσταση βέλτιστης τιμής fitness function f ανά είσοδο V

Figure 12: Γραφική παράσταση βέλτιστης τιμής fitness function f ανά είσοδο V

Η κόκκινη ευθεία που παρουσιάζεται στο δεξιό διάγραμμα είναι μια ευθεία γραμμικής παλινδρόμηση μέσω ελαχίστων τετραγώνων που φαίνεται να προσαρμόζει καλά στα δεδομένα με προσαρμοσμένο συντελεστή προσδιορισμού:$adjR^2 = 0.9830$,το οποίο είναι αρκετά κοντά στην μονάδα οπότε φαίνεται πως προσαρμόζει αρκετά καλά. Ωστόσο απτήν μορφή των δεδομένων φαίνεται πως η συσχέτιση είναι πολυωνυμική οπότε θα δοκιμάσουμε ένα μοντέλο 2ου βαθμού με συντελεστές που προκύπτουν μέσω ελαχίστων τετραγώνων.

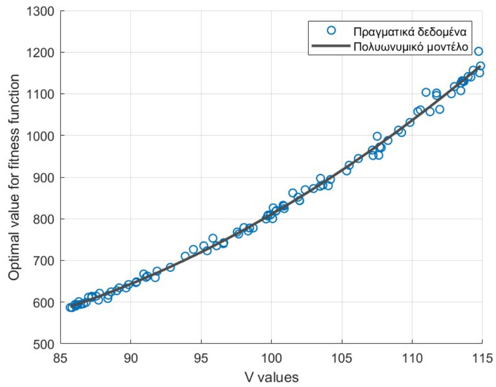


Figure 13: Γραφική παράσταση βέλτιστης τιμής fitness function f ανά είσοδο V

Για πολύωνυμο μορφής f=$aV^2 + bV + c$

c=1753.21 b=-38.50 a=0.29081 , και ο συντελεστής προσδιορισμού είναι $adjR^2 = 0.9967$,προσαρμόζει τέλεια συναρτήσει της εισόδου. Θα επαναλάβουμε την παραπάνω διαδικασία για 100 δείγματα V εύρους 85-115 και θα εξετάσουμε την επάρκεια του μοντέλου που αναπτύξαμε:

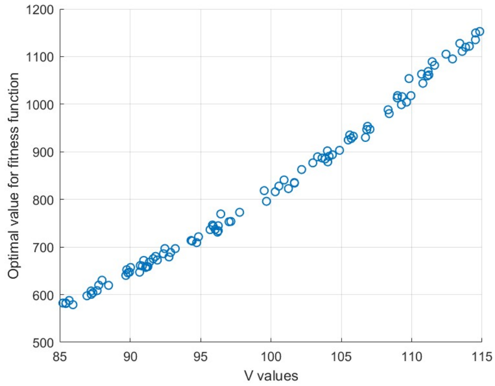


Figure 14: Γραφική παράσταση βέλτιστης τιμής fitness function f ανά είσοδο V

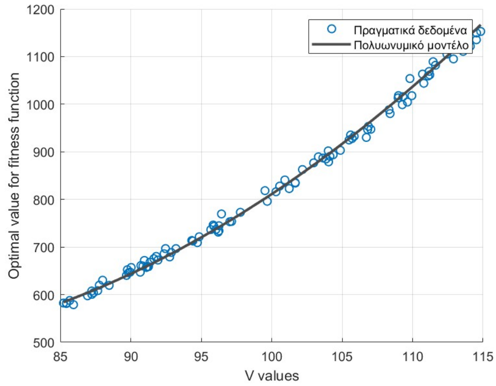


Figure 15: Γραφική παράσταση βέλτιστης τιμής fitness function f ανά είσοδο V

Στα νέα δεδομένα ο συντελεστής προσδιορισμού έχει τιμή $adjR^2 = 0.9963$ συνεπώς φαίνεται πως η βέλτιστη τιμή της αντικειμενικής συνάρτησης του γενετικού αλγορίθμου εξαρτάται πολυωνυμικά με τον ρυθμό ροής οχημάτων στην είσοδο του δικτύο με πολυώνυμο δευτέρου βαθμού.

8 Σχόλια

Τα αρχεία που αφορούν την υλοποίηση του γενετικού αλγορίθμου είναι όλα εκτός απ το script.m.Τρέχοντας το RunMain.m παράγονται τα διαγράμματα που αφορούν τα αποτελέσματα ανά γεννεά.Το script.m αφορά τα αποτελέσματα που παίρνω απ την RunMain.m και κάνω την ανάλυση δεδομένων που περιγράφεται στο section 7.4.Δεν χρησιμοποιείται σε κανένα άλλο αρχείο οπότε το RunMain.m αρκεί για να τρέξετε τον αλγόριθμο.