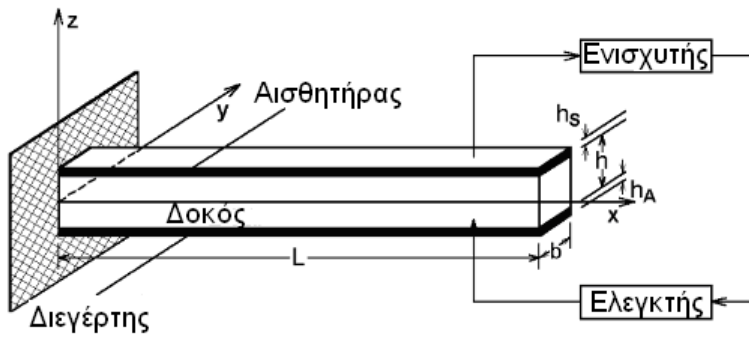


διαφορετικές θέσεις αισθητήρα-διεγέρτη



ΟΝΟΜΑ ΦΟΙΤΗΤΗ: ΣΕΛΛΗΣ ΗΛΙΑΣ

ΑΡΙΘΜΟΣ ΜΗΤΡΩΟΥ: 2004010054

ΤΜΗΜΑ: ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ ΚΑΙ ΔΙΟΙΚΗΣΗΣ

ΕΠΙΒΛΕΠΩΝ ΚΑΘΗΓΗΤΗΣ : ΣΤΑΥΡΟΥΛΑΚΗΣ ΓΕΩΡΓΙΟΣ

ΠΕΡΙΕΧΟΜΕΝΑ

1. ΕΙΣΑΓΩΓΗ

2. ΘΕΩΡΙΑ

3. ΠΕΡΙΓΡΑΦΗ ΠΡΟΒΛΗΜΑΤΟΣ ΚΑΙ ΚΩΔΙΚΑ

4. 1^ο ΜΕΡΟΣ

5. 2^ο ΜΕΡΟΣ

6. ΣΥΜΠΕΡΑΣΜΑΤΑ

7. ΒΙΒΛΙΟΓΡΑΦΙΑ

1. ΕΙΣΑΓΩΓΗ

Η διπλωματική έγινε με την επίβλεψη και την βοήθεια του κύριου Γεώργιου Σταυρουλάκη καθηγητή του τμήματος Μηχανικών Παραγωγής και Διοίκησης Πολυτεχνείου Κρήτης.

Αφορά την μελέτη ασαφούς ελεγκτή για διάφορες θέσης αισθητήρα . Το σύστημα πάνω στο οποίο δοκιμάζουμε τον ελεγκτή έχει σχεδιαστεί και έχει χρησιμοποιηθεί από τον κύριο Γεώργιο Ταιρίδη στα πλαίσια της μεταπτυχιακής του διατριβής με τίτλο ' προσομοίωση σύνθετων ευφυών μηχανικών συστημάτων με αισθητήρες και διεγέρτες από πιεζοηλεκτρικά υλικά ' Χανιά Μάιος 2009.

Το σύστημα που εξετάζουμε αποτελείται από μια πακτωμένη ράβδο (πρόβολο δοκού) η οποία εκτίθεται σε αρμονικά ταλάντωση .Προσομοιώνουμε την μηχανική συμπεριφορά σε MATLAB. Στόχος μας είναι να ελέγξουμε όσο το δυνατό γίνεται την ταλάντωση ως προς το εύρος της και την ευστάθεια στο χρόνο, να βρούμε την βέλτιστη θέση του αισθητήρα και να ελέγξουμε πως αντιδρά σε ένα τυχαίο εύρος τιμών.

2. ΘΕΩΡΙΑ

Ι) Ασαφής λογική

Η ασαφής λογική αφορά τα εντος-γραμμής έμπειρα συστήματα που έχουν εφαρμοστεί για τον έλεγχο βιομηχανικών διαδικασιών σε πραγματικό χρόνο. Ακόμα κατέχει παράλληλη θέση με τη θεωρία πιθανοτήτων , την οποία όμως δεν την αντικαθιστά, και στοχεύει να αναπαράγει τον ασαφή συμπερασμό για αβέβαιες και σύνθετες διαδικασίες.

Η θεωρία της ασαφής λογική βασίζεται στην προϋπόθεση ότι ο περιβάλλον χώρος απαρτίζεται από στοιχεία που ανήκουν σε σύνολα με διαφορετικούς βαθμούς συμμετοχής. Η ασάφεια δημιουργεί μια πλειότιμη έννοια στο χώρο της αβεβαιότητας, παραδείγματα της οποίας είναι η αλήθεια το ψεύδος και οι ενδιάμεσες έννοιες ασάφεια μπορεί να εισαχθεί στη θεωρία των συνόλων , αν γενικευτεί η χαρακτηριστική συνάρτηση για να λαμβάνει άπειρο αριθμό τιμών στο διάστημα $[0,1]$.

Όταν ένα στοιχείο ανήκει στο υπό εξέταση σύνολο τότε λαμβάνει την τιμή 1 αλλιώς την τιμή 0 και δεν ανήκει στο σύνολο ,αυτό το στοιχείο ανήκει στο κλασικό σύνολο .Σε αντίθεση στα ασαφή σύνολα δεν ισχύει αυτό ,δηλαδή κάθε στοιχείο μπορεί να λάβει τιμή από ένα εύρος τιμών και να ανήκει στο υπό εξέταση σύνολο με αυτήν την τιμή .Όσο μεγαλύτερη είναι αυτή η τιμή τόσο μεγαλύτερη είναι και η συμμετοχή του στοιχείου στο σύνολο. Αυτή η συνάρτηση συνάρτηση συμμετοχής.

Συνεπώς στα ασαφή σύνολα το δίπολο τιμών 0 ή 1 αντικαθιστάται από ένα διάστημα τιμών $[0 , 1]$. Η ασαφοποίηση είναι πλέον σημαντική διαδικασία για την ασαφή θεωρία. Συγκεκριμένα πρόκειται για την διαδικασία μετατροπής μιας ακριβούς αριθμητικής ποσότητας σε ασαφή η αναπαριστάται με τις συναρτήσεις συμμετοχής.

Στις συναρτήσεις συμμετοχής οι τιμές ορίζονται είτε διαισθητικά είτε με χρήση αλγόριθμων και λογικών διαδικασιών όπως: διαίσθηση , συνεπαγωγή, κατάξη , γωνιακά ασαφή σύνολα, νευρωνικά δίκτυα , γενετικοί αλγόριθμοι και επαγωγική προσαρμογή.

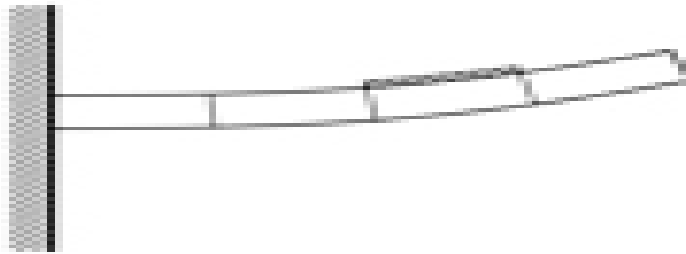
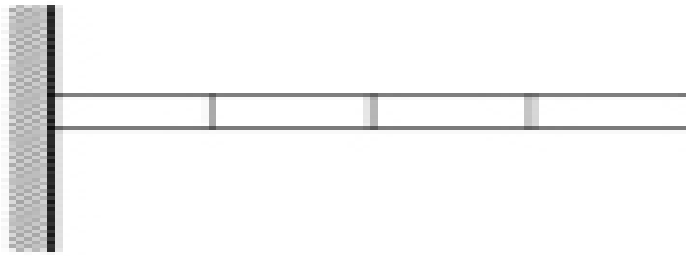
Γι 'αυτή την εργασία και συγκεκριμένα για το πρόβλημα της δοκού βασιστήκαμε στη <<διαίσθηση του μηχανικού>> για να δώσουμε τις τιμές στις συναρτήσεις συμμετοχής τις οποίες πήρα με 'έτοιμες από τον κύριο Ταϊριδη.

Η αποασαφοποίηση ορίζεται η διαδικασία μετατροπής ασαφών εξόδων σε ακριβείς , η διαδικασία αυτή είναι απαραίτητη γιατί οι τιμές των εξόδων πρέπει να είναι ακριβείς ,ειδικά όταν το σύστημα χρησιμοποιείται σαν ελεγκτής .Οι μέθοδοι που χρησιμοποιούνται αποασαφοποίηση είναι: Αρχή μέγιστου , κεντροειδής μέθοδος μέθοδος σταθμισμένου μέσου , μέθοδος μέσου- μέγιστου , κέντρο βάρους αθροισμάτων , κέντρο βάρους μέγιστης επιφάνειας και πρώτη ή τελευταία τιμή μέγιστου. Στην δικιά μας περίπτωση χρησιμοποιήσαμε την μέθοδο μέσου-μέγιστου για την αποασαφοποίηση της εξόδου του ασαφούς ελεγκτή.

3. ΠΕΡΙΓΡΑΦΗ ΠΡΟΒΛΗΜΑΤΟΣ ΚΑΙ ΚΩΔΙΚΑ

Έχουμε μια δοκό την οποία την έχουμε χωρίσει σε 7 μέρη.

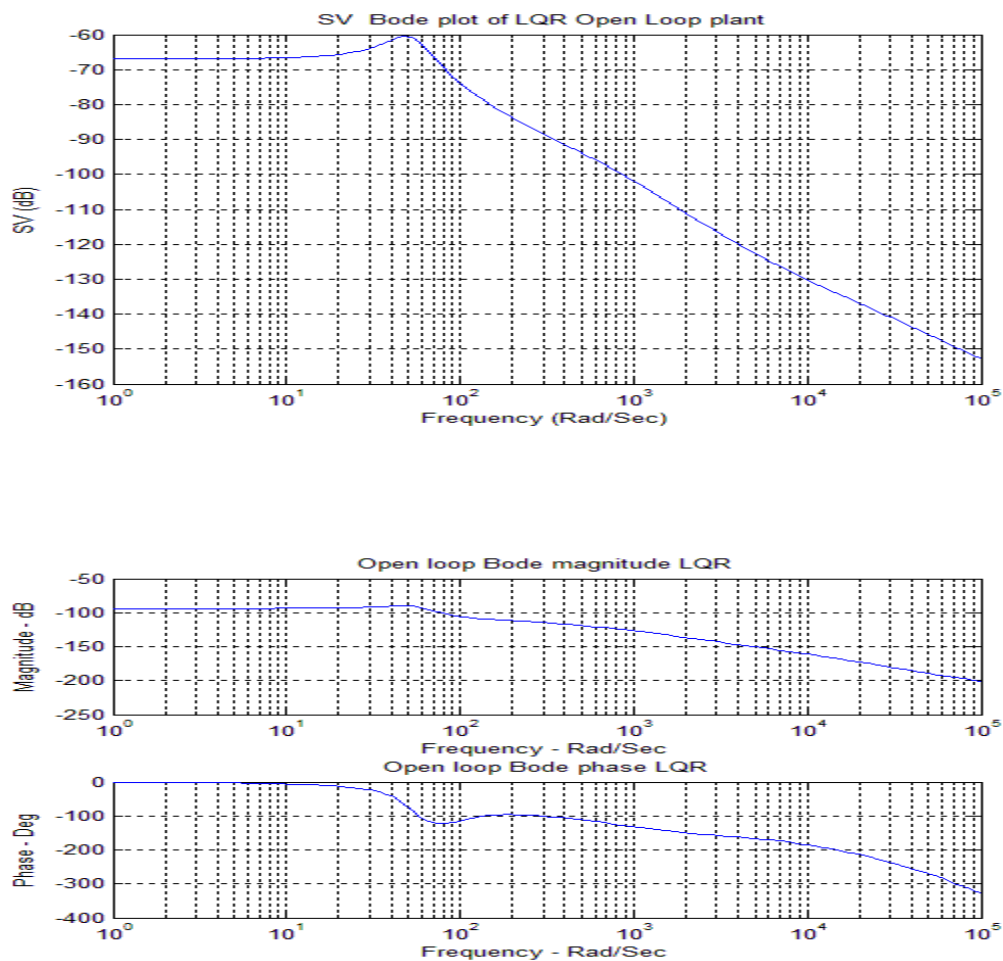
Στο καθένα υπάρχει και ένας αισθητήρας που μα δίνει πληροφορίες για την ταχύτητα την επιτάχυνση και την μετατόπιση.

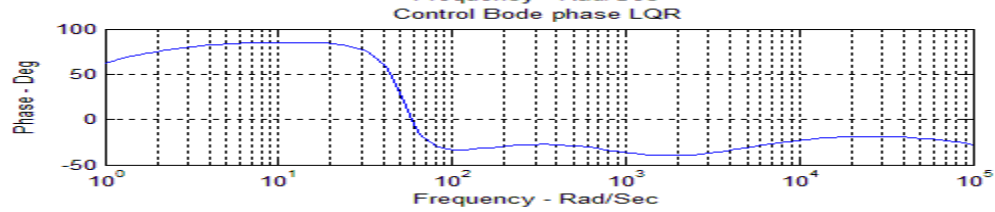
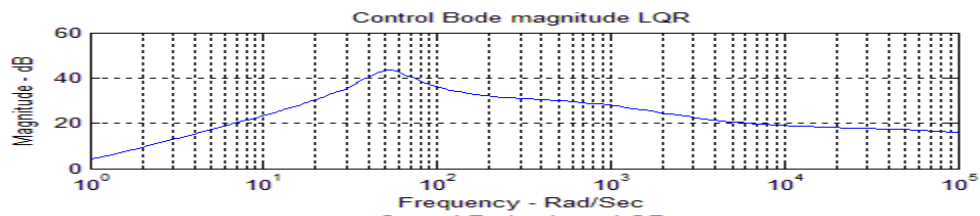
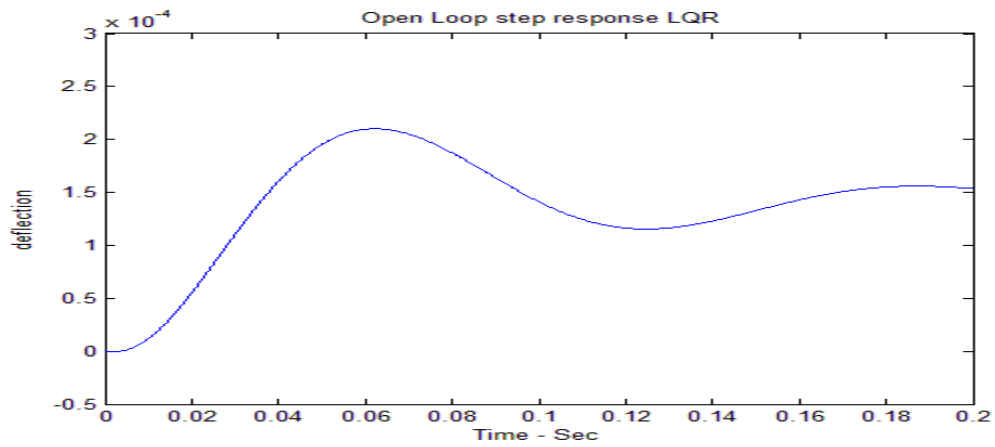
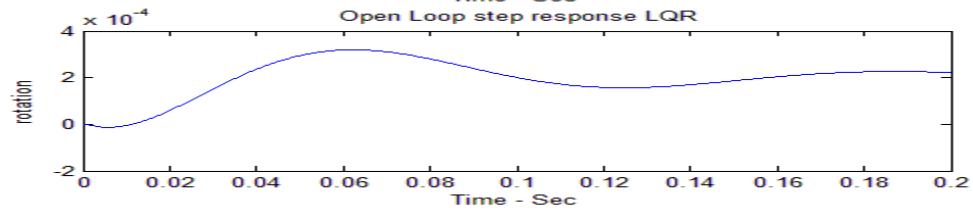
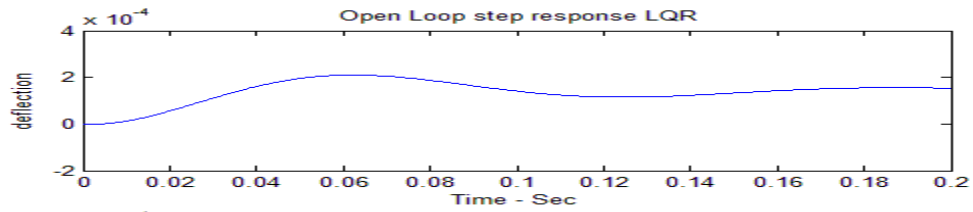


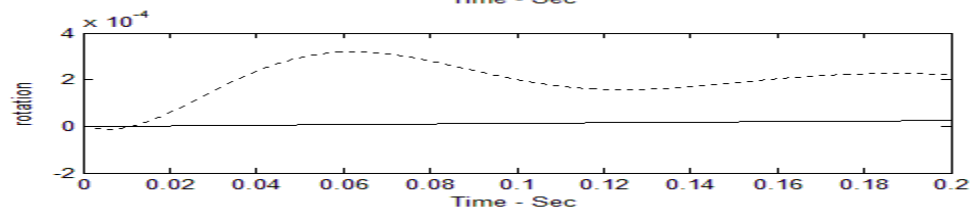
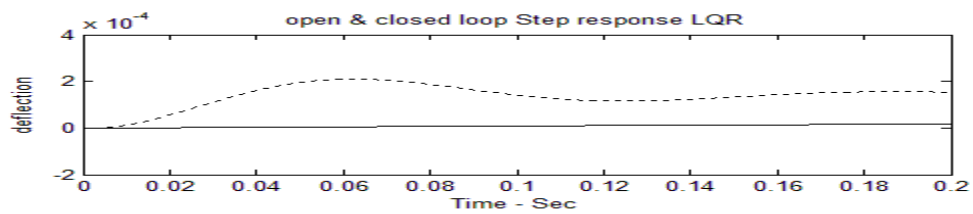
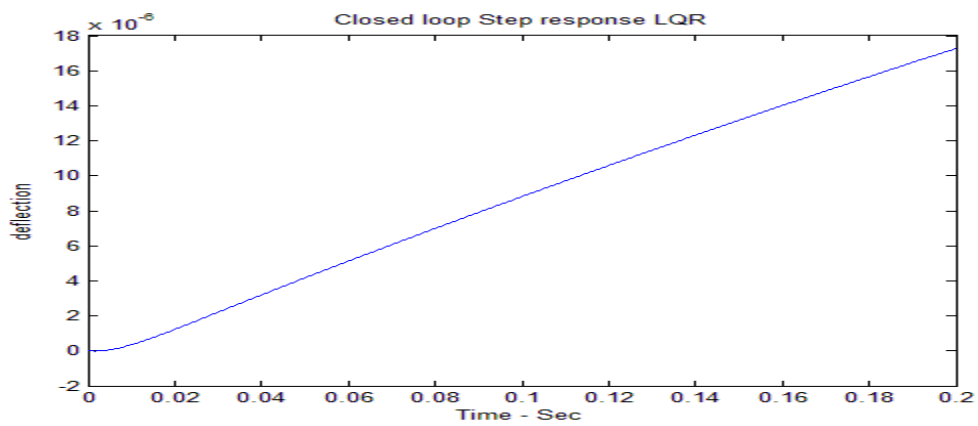
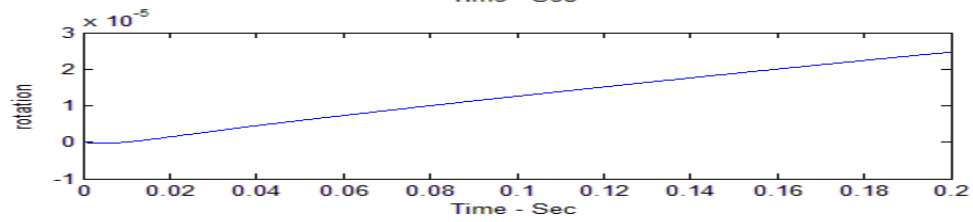
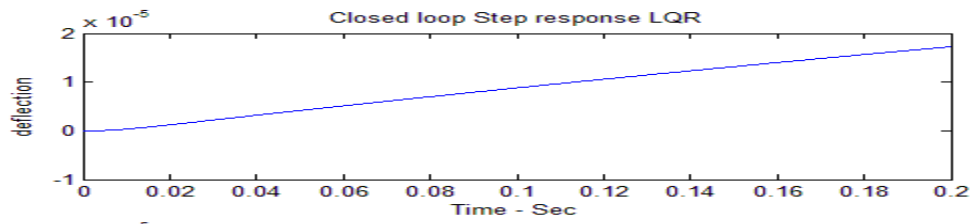
Το πρόγραμμα Beam_lqr_ges υπολογίζει την ταλάντωση και τον κλασικό έλεγχο δοκού.

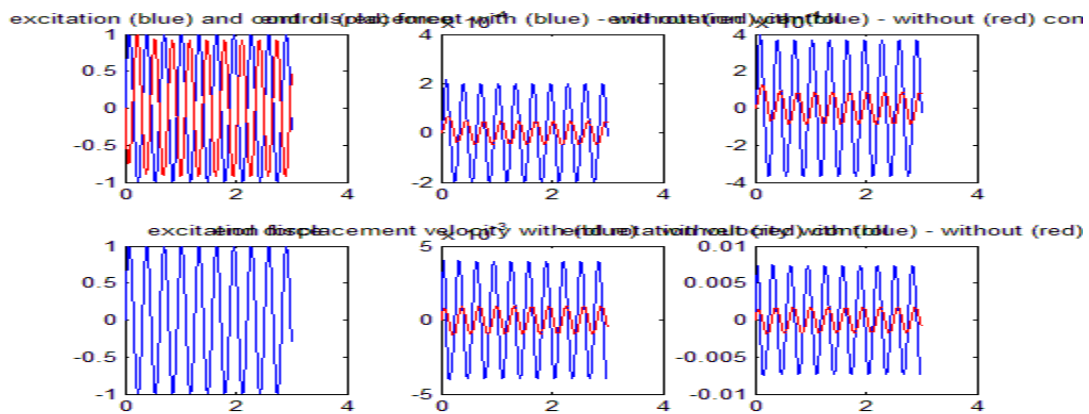
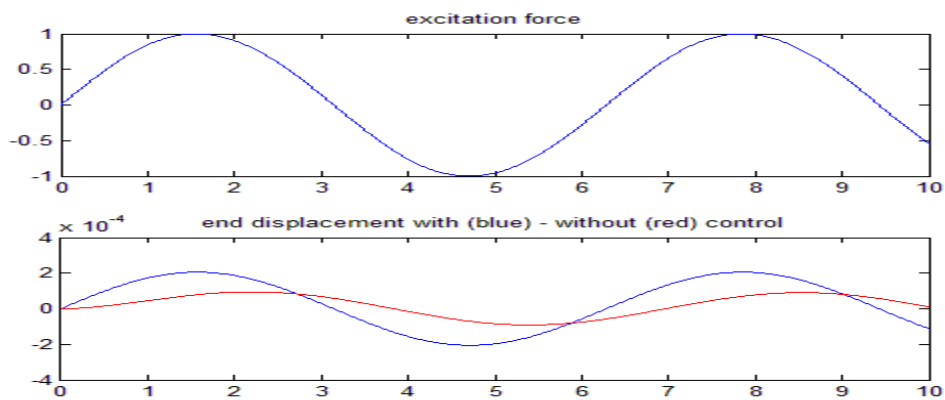
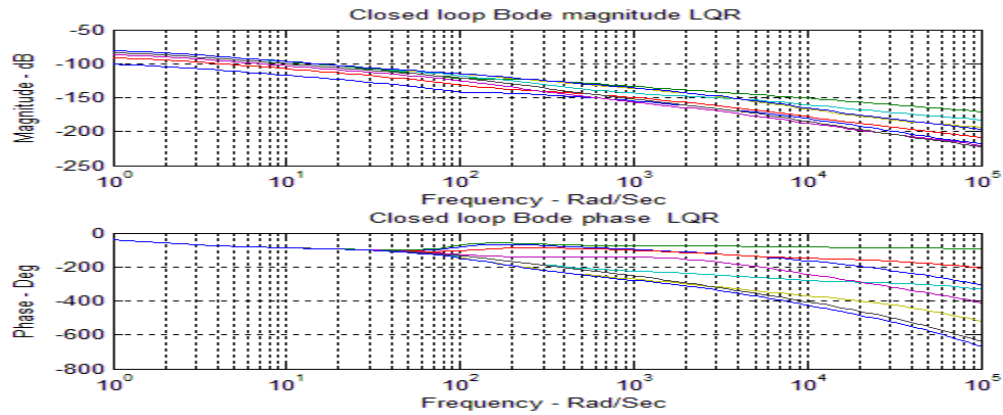
Αρχικά τρέχουμε τον αλγόριθμο `beam_lqr_ges` το οποίο είναι έτοιμο από τον κύριο Ταϊρίδη Γεώργιο. Αυτός ο αλγόριθμος αρχικά υπολογίζει την ταλάντωση και τον κλασσικό έλεγχο της δοκού. Δημιουργεί τα δεδομένα της δοκού τα οποία αποθηκεύονται στο `databeam` για μελλοντική χρήση σε άλλους αλγόριθμους.

Τα διαγράμματα που παίρνουμε από την εκτέλεση αυτού του αλγόριθμου είναι:









Το πρόγραμμα Integration_matrix ενσωματώνει μέσα στην ολοκλήρωση στο χρόνο (μεθ. Χούμβολτ) τον έλεγχο με την χρήση ασαφούς λογικής.

Τώρα θα πούμε λίγα λόγια για το πρόβλημα της προβόλου δοκού ,καθώς και τον τρόπο που μοντελοποιούνται .Υπάρχουν πολλές γνωστές μέθοδοι ολοκλήρωσης αλλά εμείς χρησιμοποιούμε μέθοδο Χούμ볼τ ,η οποία είναι μια από τις πιο παραδοσιακές μεθόδους αριθμητικής ολοκλήρωσης και η πιο σταθερή σε τέτοιου είδους προβλήματα .Αυτή χρησιμοποιεί και ο κύριος Ταϊρίδης στην μοντελοποίηση του προβλήματος την οποία ακολουθούμε και εμείς .

$$M\ddot{u} + C\dot{u} + Ku = P + Z$$

όπου: M, C, K είναι τα μητρώα μάζας, απόσβεσης και δυσκαμψίας

$u, \dot{u}, \ddot{u}$ είναι τα πεδία μετακινήσεων, ταχυτήτων και επιταχύνσεων και τέλος

P, Z είναι οι εξωτερικές δυνάμεις και οι δυνάμεις ελέγχου αντίστοιχα.

Η μέθοδος αυτή χρησιμοποιεί δυο σταθερές β και γ (σταθερές Houbolt) .Για το πρόβλημα μας αυτές οι δυο σταθερές λαμβάνουν αντίστοιχα τις τιμές 0,25 και 0,50.

Ο συνολικός χρόνος της διαδικασίας είναι 3 δευτερόλεπτα βήμα που επιλέχθηκε ,μετά από δοκιμές (Ταϊρίδης εργασία) είναι 0,001 δευτερόλεπτα ,δημιουργώντας έτσι 3000 επαναλήψεις.

Ο σταθερές ολοκλήρωσης (c_i) δίνονται από τις ακόλουθες εξισώσεις:

$$c_1 = \frac{1}{\beta(\Delta t)^2}, c_2 = \frac{1}{\beta\Delta t}, c_3 = \frac{1}{2\beta}, c_4 = \frac{\gamma}{\beta\Delta t}, c_5 = \frac{\gamma}{\beta}, c_6 = \Delta t \left(\frac{\gamma}{2\beta} - 1 \right)$$

Ο αλγόριθμος Houbolt, όπως υλοποιήθηκε με τον ασαφή έλεγχο, συνοψίζεται στα ακόλουθα βήματα.

Βήμα 1: Αρχικοποίηση μεταβλητών

$$X, \dot{X}, \ddot{X}, F_m, M, \Lambda, K, \beta, \gamma, c_1, c_2, c_3, c_4, c_5, c_6$$

Βήμα 2: Υπολογισμός ενδιάμεσης ποσότητας F^*

$$K^* = K + c_1 M + c_4 \Lambda$$

$$\text{Αντιστροφή του } K^*: F^* = \left(K^* \right)^{-1}$$

Έναρξη βρόχου for από t_0 έως t_f

Βήμα 3: Υπολογισμός ενδιάμεσης ποσότητας P^*

$$\text{Υπολογισμός μεταβολής φορτίσεων: } dF_m = F_m(t+1) - F_m(t)$$

$$\text{Υπολογισμός μεταβολής δύναμης ελέγχου } u$$

$$\text{Προσθήκη στην ποσότητα } dF_m: dF_m = dF_m + u$$

Υπολογισμός της ποσότητας P^* με χρήση των μητρώων μάζας (M) και απόσβεσης (Λ) του συστήματος:

$$P^* = dF_m + M(c_2\dot{X}(t) + c_3\ddot{X}(t)) + \Lambda(c_5\dot{X}(t) + c_6\ddot{X}(t))$$

Βήμα 4: Υπολογισμός του βήματος απόκρισης dX

$$dX = F^* P^*$$

Βήμα 5: Επίλυση του επόμενου χρονικού βήματος $(t + \Delta t)$

Calculate acceleration: $\ddot{X}(t+1) = \ddot{X}(t) + c_1 dX - c_2 \dot{X}(t) - c_3 \ddot{X}(t)$

Calculate velocity: $\dot{X}(t+1) = \dot{X}(t) + c_4 dX - c_5 \dot{X}(t) - c_6 \ddot{X}(t)$

Calculate displacement: $X(t+1) = X(t) + dX$

Τερματισμός του βρόχου for

Τέλος

Στον κώδικα `integration_matrix` αφού ορίσαμε τις μεταβλητές μας και δημιουργήσαμε αυτό που αναφέραμε προηγουμένως δηλαδή την διαδικασία Houbolt δημιουργήσαμε και μια συνάρτηση την `fuzzy_initial` η

οποία παίρνει ορίσματα τα δυο εύρη που βρίσκουμε απο την παραπάνω διαδικασία και έτσι δημιουργεί τους κανόνες για να γίνει ο fuzzy έλεγχος.

Οι κανόνες (τους οποίους πήραμε έτοιμους από τον κύριο Ταίριδη είναι οι εξής:

1. If (Displacement is Far_L) and (Velocity is Left) then (Control_Force is Max) (1)
2. If (Displacement is Far_L) and (Velocity is Null) then (Control_Force is Med+) (1)
3. If (Displacement is Far_L) and (Velocity is Right) then (Control_Force is Low+) (1)
4. If (Displacement is Close_L) and (Velocity is Left) then (Control_Force is Med+) (1)
5. If (Displacement is Close_L) and (Velocity is Null) then (Control_Force is Low+) (1)
6. If (Displacement is Close_L) and (Velocity is Right) then (Control_Force is Null) (1)
7. If (Displacement is Equilibrium) and (Velocity is Left) then (Control_Force is Low+) (1)
8. If (Displacement is Equilibrium) and (Velocity is Null) then (Control_Force is Null) (1)
9. If (Displacement is Equilibrium) and (Velocity is Right) then (Control_Force is Low-) (1)
10. If (Displacement is Close_R) and (Velocity is Left) then (Control_Force is Null) (1)

11. If (Displacement is Close_R) and (Velocity is Null) then (Control_Force is Low-) (1)
12. If (Displacement is Close_R) and (Velocity is Right) then (Control_Force is Med-) (1)
13. If (Displacement is Far_R) and (Velocity is Left) then (Control_Force is Low-) (1)
14. If (Displacement is Far_R) and (Velocity is Null) then (Control_Force is Med-) (1)
15. If (Displacement is Far_R) and (Velocity is Right) then (Control_Force is Min) (1)

Θα διερευνήσουμε αρχικά την βέλτιστη θέση του αισθητήρα ελέγχοντας και τις 7 πιθανές θέσεις που μπορεί να πάρει. Η θέση του διεγέρτη δηλαδή της δύναμης που ασκείται δεν μας ενδιαφέρει γιατί το εύρος που χρειαζόμαστε έτσι ώστε να φτάσουμε στο αποτέλεσμα που θέλουμε, εξαρτάται από την θέση του αισθητήρα(dofcontrol) και όχι από το που ασκείται η δύναμη (dofload).Βλέπουμε και στον κώδικα αυτό που αναφέραμε πιο πάνω:

- **ΕΥΡΟΣ ΜΕΤΑΤΟΠΙΣΗΣ ΤΑΧΥΤΗΤΑΣ & ΕΠΙΤΑΧΥΝΣΗΣ ΧΩΡΙΣ FUZZY ΕΛΕΓΧΟ**

```
umax(dofcontrol,1) = max(u(dofcontrol,:)) -  
min(u(dofcontrol,:));
```

```
udmax(dofcontrol,1) = max(ud(dofcontrol,:)) -  
min(ud(dofcontrol,:));
```

```
uddmax(dofcontrol,1) = max(udd(dofcontrol,:)) -  
min(udd(dofcontrol,:));
```

- **ΕΥΡΟΣ ΜΕΤΑΤΟΠΙΣΗΣ ΤΑΧΥΤΗΤΑΣ & ΕΠΙΤΑΧΥΝΣΗΣ FUZZY ΕΛΕΓΧΟ**

```
u_f_euros(dofcontrol,1)=max(u_f(dofcontrol,:))-  
min(u_f(dofcontrol,:));
```

```
ud_f_euros(dofcontrol,1)=max(ud_f(dofcontrol,:)-  
min(ud_f(dofcontrol,:));
```

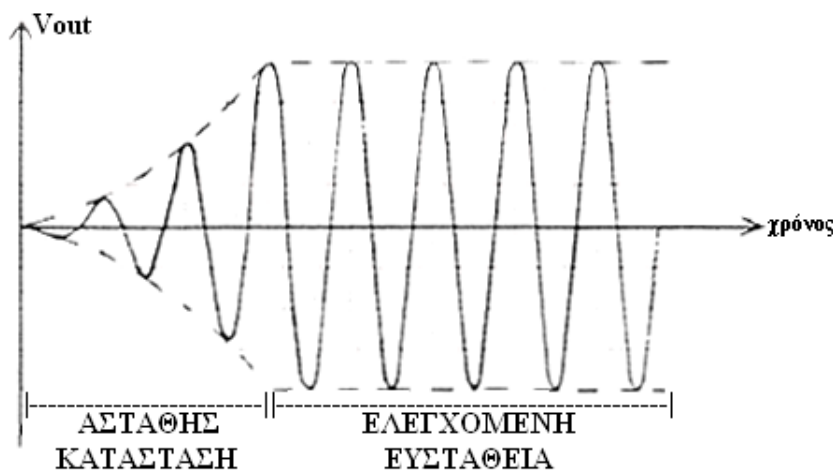
```
udd_f_euros(dofcontrol,1)=max(udd_f(dofcontrol,:)-  
min(udd_f(dofcontrol,:));
```

Γι' αυτό τον λόγο όταν τρέχω τον κώδικα το dofcontrol παίρνει τιμές από το 1 έως το 7 ενώ για dofload βάζω τυχαίες αφού δεν επηρεάζει τα αποτελέσματά μου. Για κάθε θέση του αισθητήρα θα ελέγχω τα 3 μεγέθη ταχύτητα(velocity) μετατόπιση(displacement) και επιτάχυνση (rotation).

Σκοπός μας είναι να βρούμε που βελτιστοποιούνται αυτά τα μεγέθη.

Τα κριτήρια είναι τα εξής:

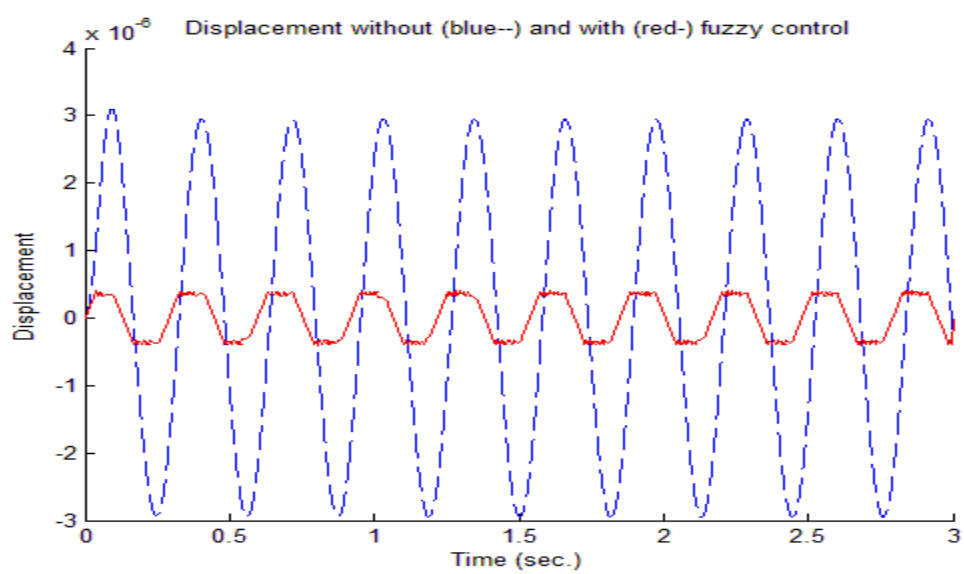
1. Η ταλάντωση να έχει ελεγχόμενη ευστάθεια στο χρόνο
2. Το εύρος της ταλάντωσης να είναι μικρό



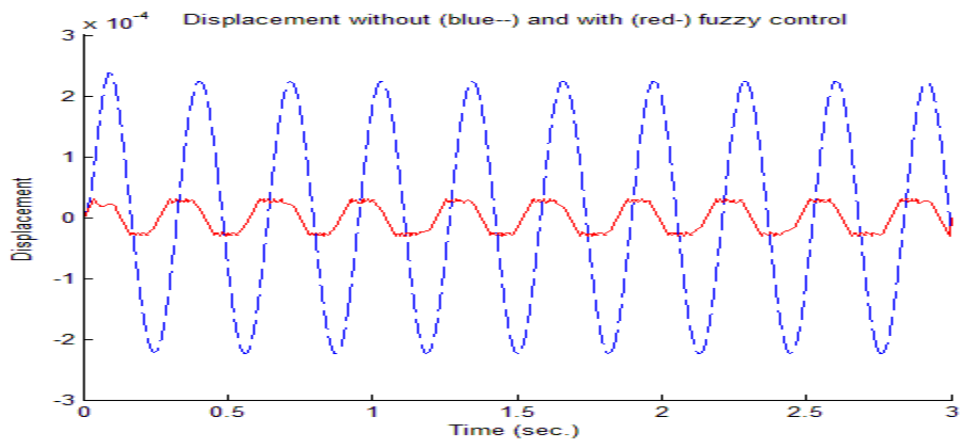
4. 1^ο ΜΕΡΟΣ

✚ Ξεκινάμε με την μετατόπιση για θέση αισθητήρα 1εως 7(dofcontrol=1:7)

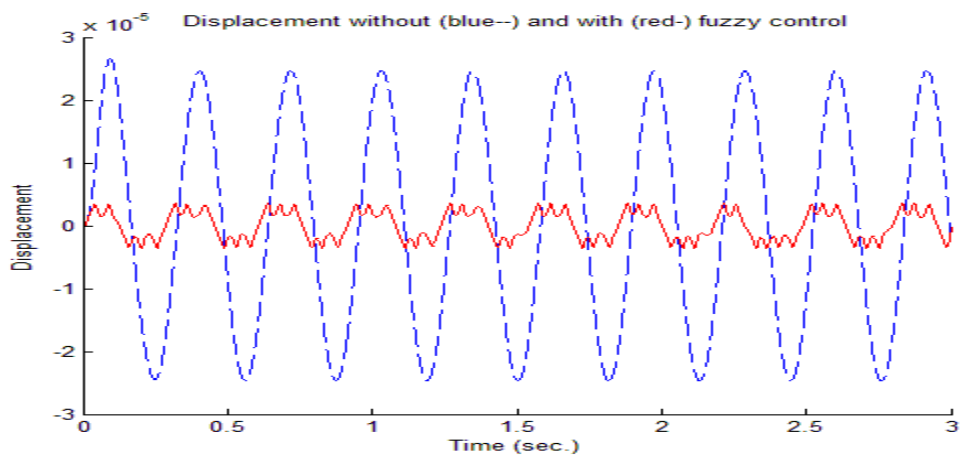
1.



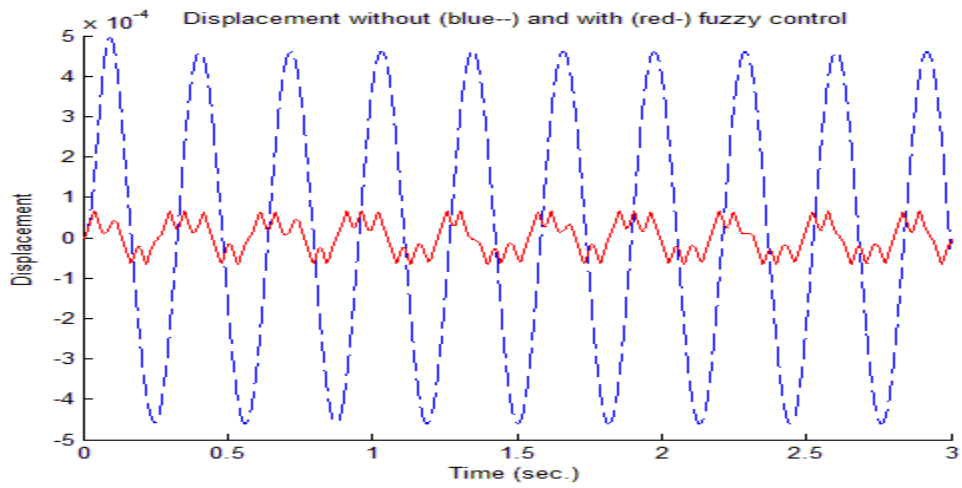
2.



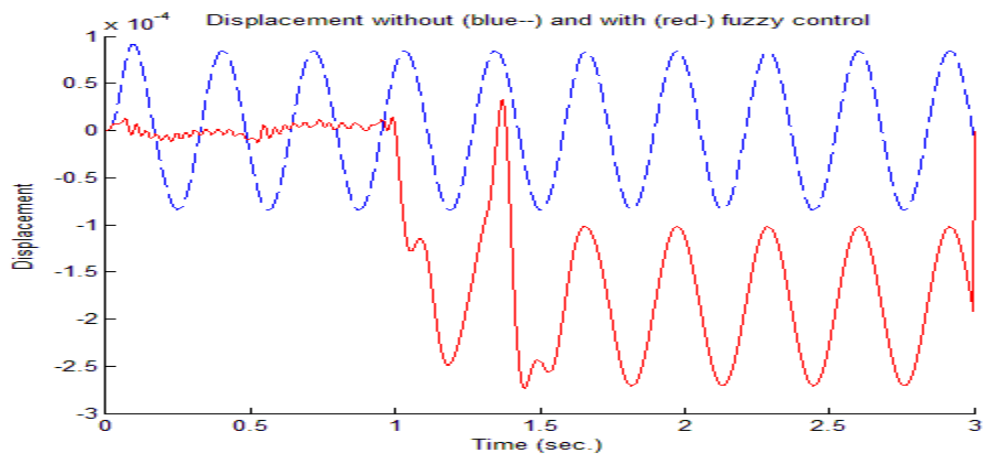
3.



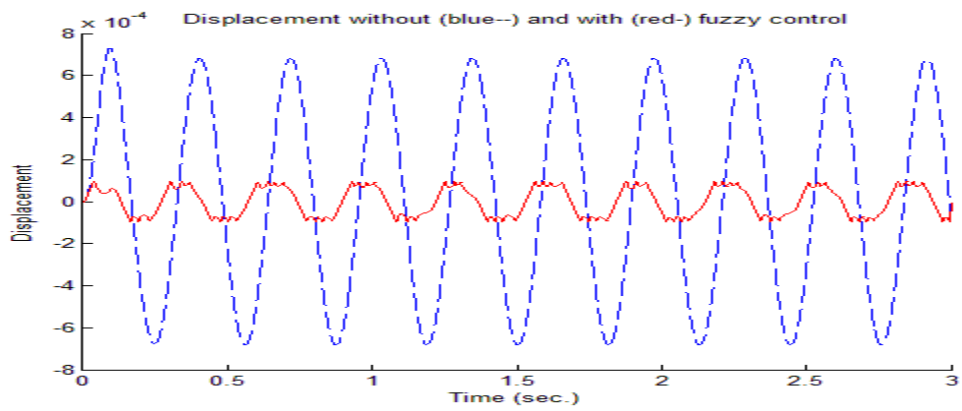
4.



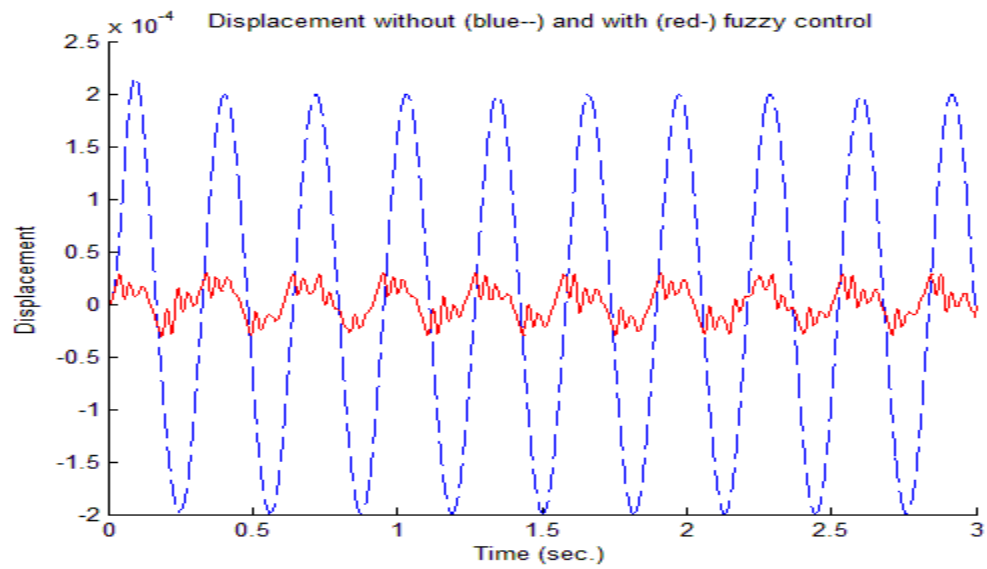
5.



6.

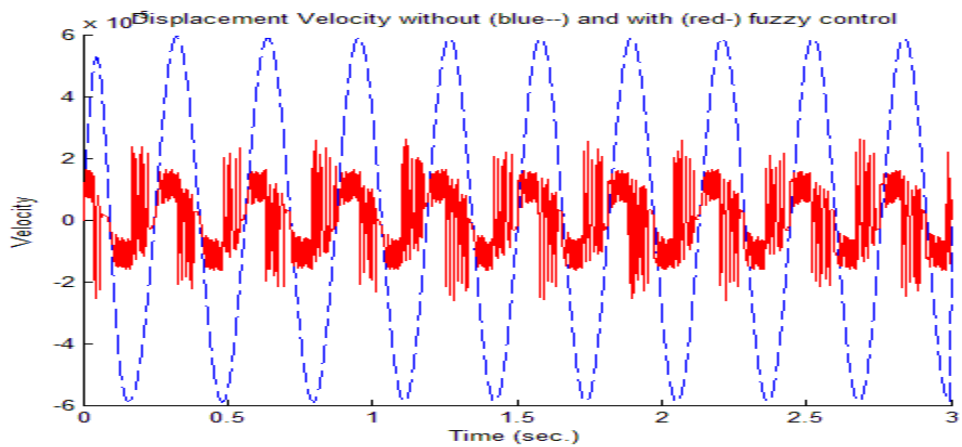


7.

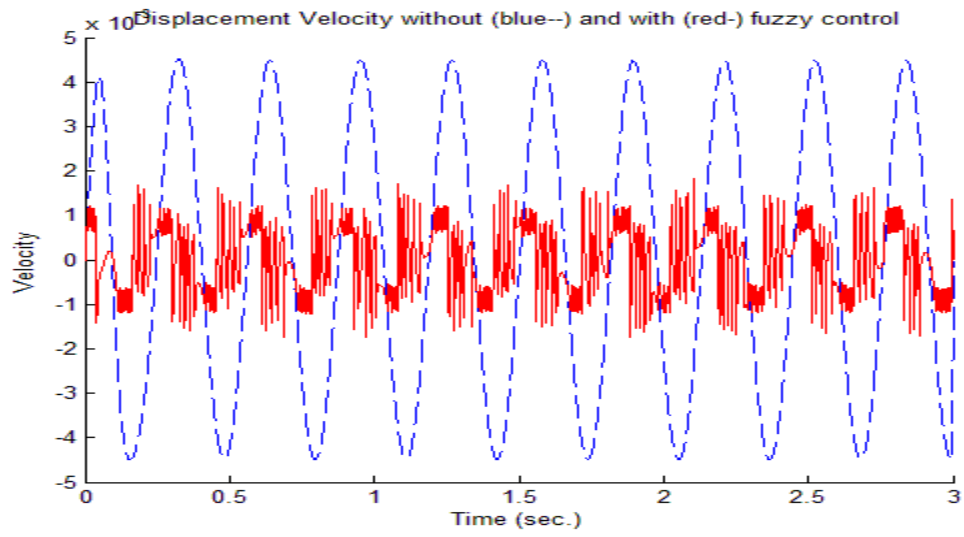


✚ Συνεχίζουμε με την ταχύτητα για θέση αισθητήρα 1εως 7(dofcontrol=1:7)

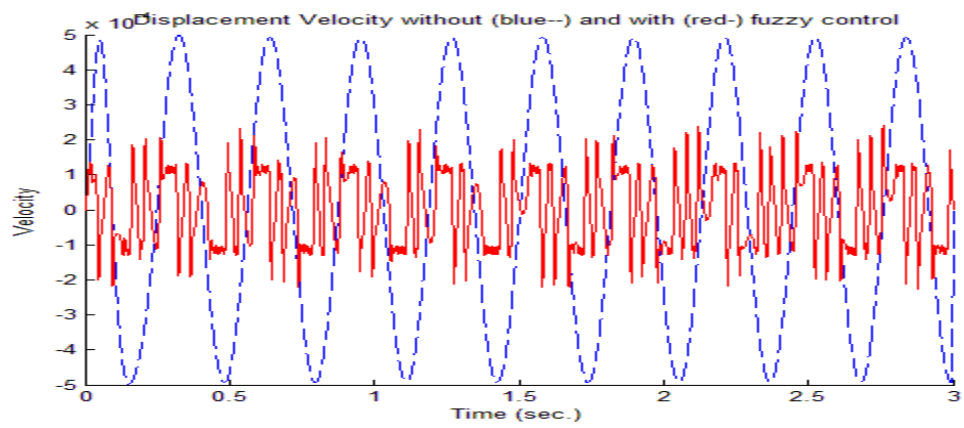
1.



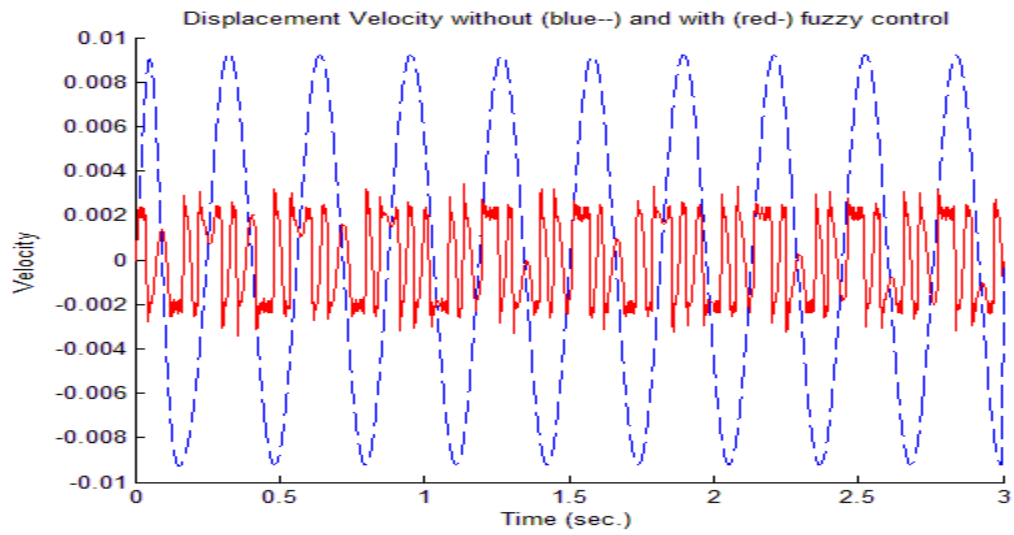
2.



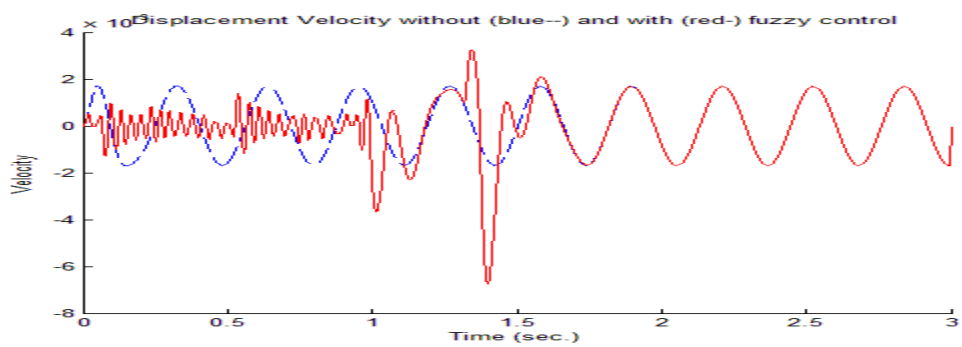
3.



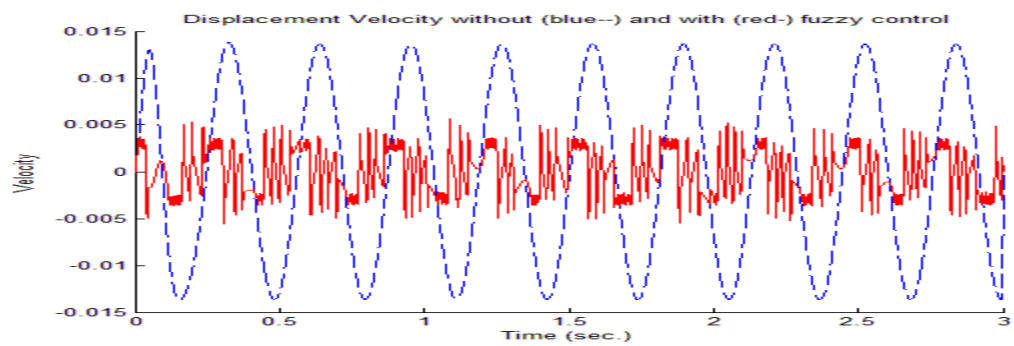
4.



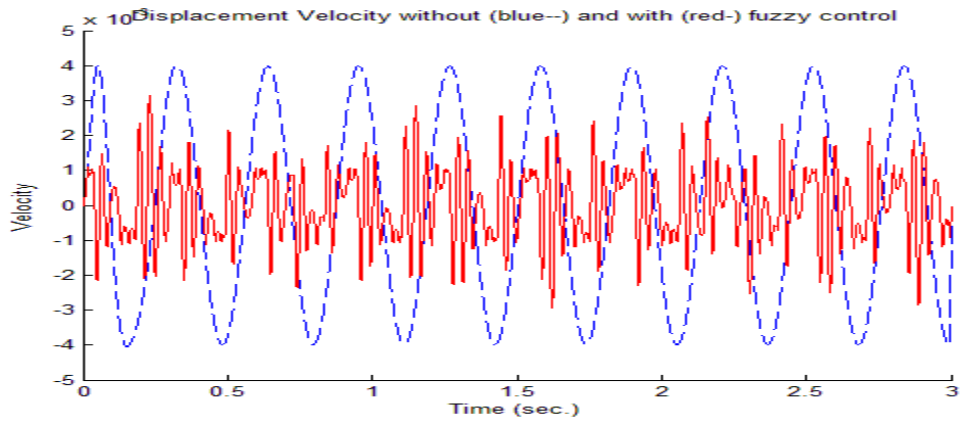
5.



6.

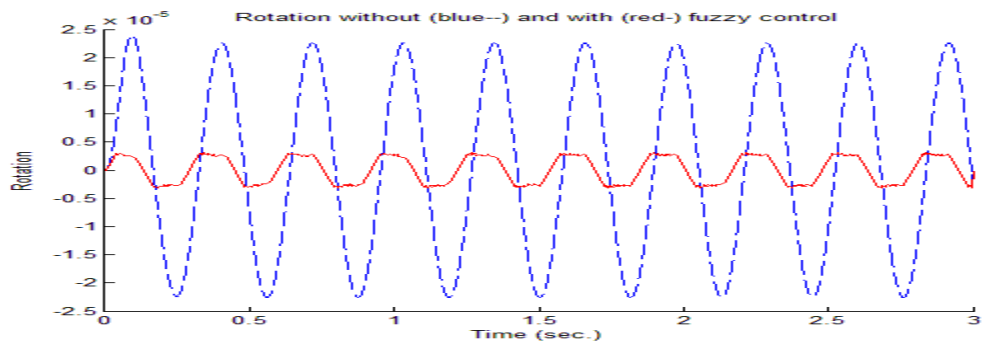


7.

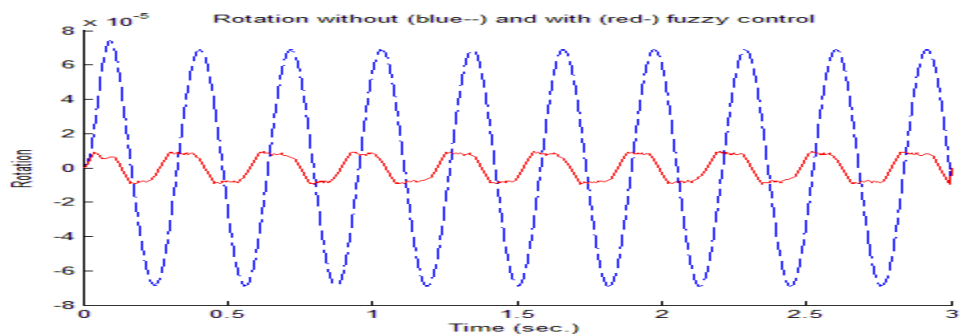


✚ Τώρα ελέγχουμε την επιτάχυνση για θέση αισθητήρα 1εως 7(dofcontrol=1:7)

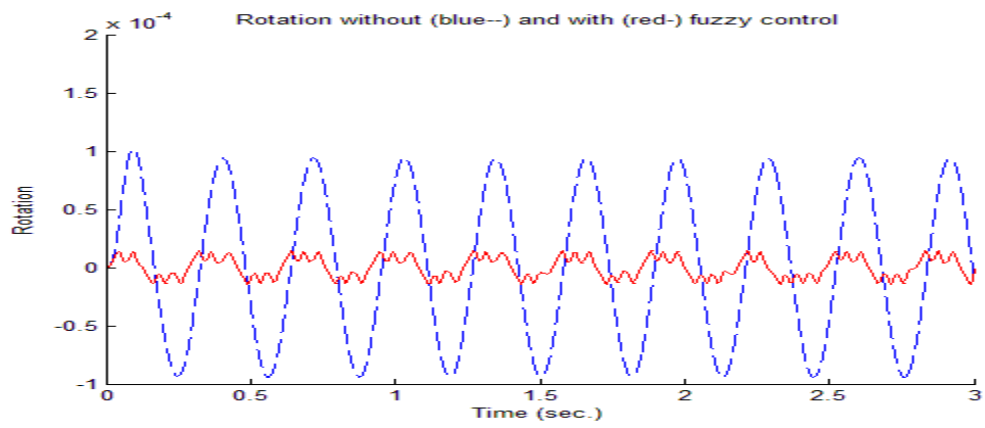
1.



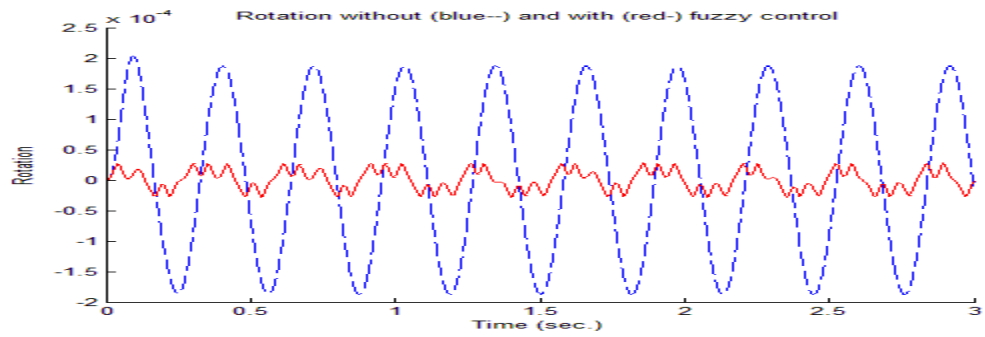
2.



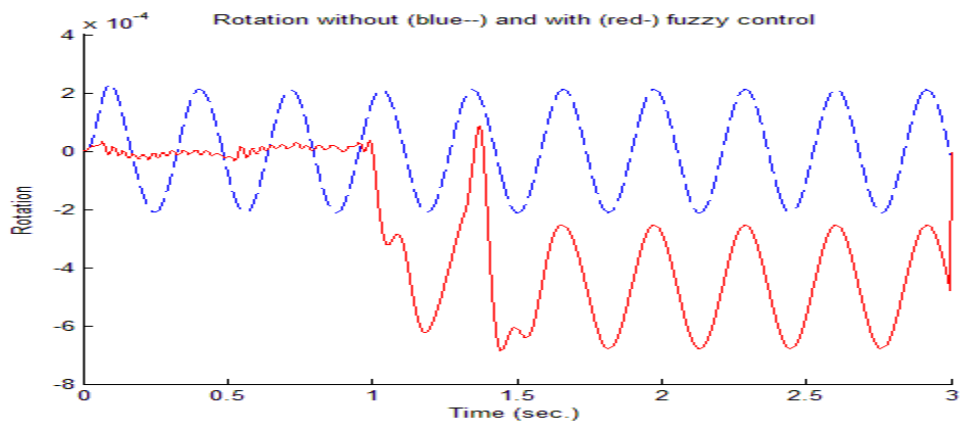
3.



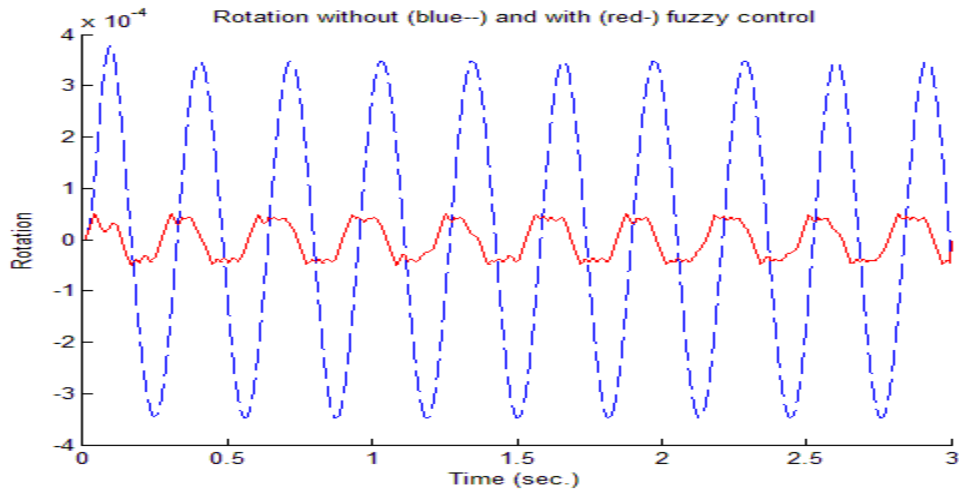
4.



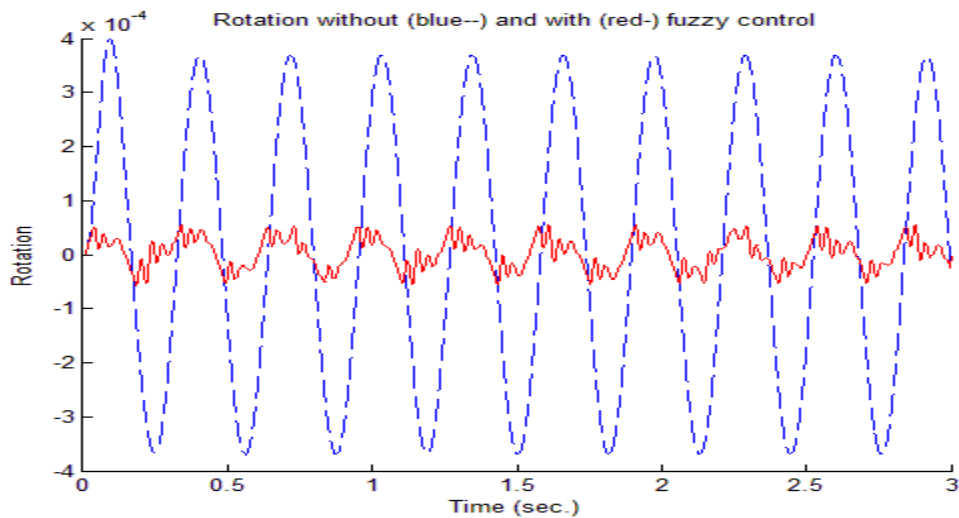
5.



6.



7.



Τα αποτελέσματα που παίρνουμε από τα παραπάνω στα διαγράμματα είναι τα εξής:\

✓ Τα ακόλουθα αποτελέσματα αφορούν μετρήσεις χωρίς fuzzy έλεγχο.

Για την θέση του αισθητήρα στη θέση 1 έχουμε εύρος για :

μετατόπιση (χ)= 0.605×10^{-5}

ταχύτητα (u)= 0.1183×10^{-3}

επιτάχυνση(a)= 0.0041

Για την θέση του αισθητήρα στη θέση 2 έχουμε εύρος για :

$$\text{μετατόπιση } (\chi) = 4.62 \cdot 10^{-4}$$

$$\text{ταχύτητα } (u) = 0.0090$$

$$\text{επιτάχυνση } (\alpha) = 0.3266$$

Για την θέση του αισθητήρα στη θέση 3 έχουμε εύρος για :

$$\text{μετατόπιση } (\chi) = 5.12 \cdot 10^{-5}$$

$$\text{ταχύτητα } (u) = 9.95 \cdot 10^{-4}$$

$$\text{επιτάχυνση } (\alpha) = 0.0305$$

Για την θέση του αισθητήρα στη θέση 4 έχουμε εύρος για :

$$\text{μετατόπιση } (\chi) = 9.57 \cdot 10^{-4}$$

$$\text{ταχύτητα } (u) = 0.0186$$

$$\text{επιτάχυνση } (\alpha) = 0.6065$$

Για την θέση του αισθητήρα στη θέση 5 έχουμε εύρος για :

$$\text{μετατόπιση } (\chi) = 1.7622 \cdot 10^{-4}$$

$$\text{ταχύτητα } (u) = 0.0034$$

$$\text{επιτάχυνση } (\alpha) = 0.1103$$

Για την θέση του αισθητήρα στη θέση 6 έχουμε εύρος για :

$$\text{μετατόπιση } (\chi) = 0.0014$$

$$\text{ταχύτητα } (u) = 0.0275$$

$$\text{επιτάχυνση } (\alpha) = 0.8362$$

Για την θέση του αισθητήρα στη θέση 7 έχουμε εύρος για :

$$\text{μετατόπιση } (\chi) = 4.15 \cdot 10^{-4}$$

$$\text{ταχύτητα } (u) = 0.0081$$

$$\text{επιτάχυνση } (\alpha) = 0.2572$$

✓ Τώρα έχουμε τα αποτελέσματα με fuzzy έλεγχο

Για την θέση του αισθητήρα στη θέση 1 έχουμε εύρος για :

$$\begin{aligned}\text{μετατόπιση } (\chi) &= 0.8045 * 10^{-6} \\ \text{ταχύτητα } (u) &= 5.2260 * 10^{-5} \\ \text{επιτάχυνση } (a) &= 0.0645\end{aligned}$$

Για την θέση του αισθητήρα στη θέση 2 έχουμε εύρος για :

$$\begin{aligned}\text{μετατόπιση } (\chi) &= 0.622 * 10^{-4} \\ \text{ταχύτητα } (u) &= 0.0036 \\ \text{επιτάχυνση } (a) &= 6.043\end{aligned}$$

Για την θέση του αισθητήρα στη θέση 3 έχουμε εύρος για :

$$\begin{aligned}\text{μετατόπιση } (\chi) &= 0.726 * 10^{-5} \\ \text{ταχύτητα } (u) &= 4.65 * 10^{-4} \\ \text{επιτάχυνση } (a) &= 0.1212\end{aligned}$$

Για την θέση του αισθητήρα στη θέση 4 έχουμε εύρος για :

$$\begin{aligned}\text{μετατόπιση } (\chi) &= 1.32 * 10^{-4} \\ \text{ταχύτητα } (u) &= 0.0069 \\ \text{επιτάχυνση } (a) &= 6.4055\end{aligned}$$

Για την θέση του αισθητήρα στη θέση 5 έχουμε εύρος για :

$$\begin{aligned}\text{μετατόπιση } (\chi) &= 3.056 * 10^{-4} \\ \text{ταχύτητα } (u) &= 0.0100 \\ \text{επιτάχυνση } (a) &= 0.6346\end{aligned}$$

Για την θέση του αισθητήρα στη θέση 6 έχουμε εύρος για :

$$\text{μετατόπιση } (\chi) = 1.932 \cdot 10^{-4}$$

$$\text{ταχύτητα } (u) = 0.0112$$

$$\text{επιτάχυνση } (a) = 7.429$$

Για την θέση του αισθητήρα στη θέση 7 έχουμε εύρος για :

$$\text{μετατόπιση } (\chi) = 0.604 \cdot 10^{-4}$$

$$\text{ταχύτητα } (u) = 0.0061$$

$$\text{επιτάχυνση } (a) = 1.1053$$

Παρατηρούμε σύμφωνα και από τα αποτελέσματα που παίρνουμε από τον κώδικα ότι η πιο βέλτιστη επιλογή είναι η θέση 1 του αισθητήρα. Αυτό συμβαίνει καθώς η ταλάντωση έχει ελεγχόμενη ευστάθεια στο χρόνο, πράγμα που συμβαίνει σε σχεδόν όλα τα διαγράμματα της μετατόπισης εκτός της περίπτωσης του αισθητήρα που βρίσκεται στην θέση 5 που παρατηρούμε μεγάλη αστάθεια στην περίπτωση που έχουμε fuzzy έλεγχο. Ακόμα από τα αποτελέσματα που παίρνουμε τον κώδικα το μικρότερο εύρος το έχουμε στην θέση 1.

Τα παραπάνω αποτελέσματα τα τρέξαμε ένα προς ένα για να γίνει πιο εύκολη η διερεύνηση δηλαδή για συντομία θα μπορούσαμε να δημιουργήσουμε ένα for dofcontrol 1:7 αλλά ήταν πολύ δύσκολο να ελεγχθούν όλα τα γραφήματα μαζί.

5. 2^ο ΜΕΡΟΣ

Στο προηγούμενο βήμα υπολογίζαμε την θέση του αισθητήρα βρίσκοντας το εύρος της μετατόπισης, ταχύτητας και επιτάχυνσης με τον τρόπο που είχε βρει ο κύριος Ταϊρίδης, ενώ τώρα θα προσπαθήσουμε να δούμε τι αποτέλεσμα θα έχουμε αν τα εύρη που αναφέραμε πριν τα επιλέγουμε

τώρα μέσα από ένα τυχαίο εύρος τιμών. Όλα αυτά τα ελέγχουμε για τον έλεγχο με fuzzy καθώς ς τα εύρη αυτά μπαίνουν ορίσματα στην fuzzy_initial συνάρτηση .

Δημιουργήσαμε στον κώδικα 2 for που δίνουν τιμές στα umax & udmax .

```
for umax=0.001:0.005:0.016  
  
    auks1=auks1+1;  
    auks2=0;  
  
    for udmax=0.0001:0.005:0.0151
```

```
        auks2=auks2+1;
```

και για να βρίσκουμε το εύρος του fuzzy που είναι και το ζητούμενο χρησιμοποιούμε αυτές τις τρεις γραμμές κώδικα και για να τα αποθηκεύω δημιούργησα τρεις πίνακες 4*4.

```
u_f_euros (dofcontrol,1)=max(u_f(dofcontrol,:))-  
min(u_f(dofcontrol,: ));  
  
ud_f_euros(dofcontrol,1)=max(ud_f(dofcontrol,:))-  
min(ud_f(dofcontrol,: ));  
  
udd_f_euros(dofcontrol,1)=max(udd_f(dofcontrol,:))-  
min(udd_f(dofcontrol,: ));  
  
matrix_u_f_euros(auks1,auks2)=u_f_euros(dofcontrol,  
1);  
  
matrix_ud_f_euros(auks1,auks2)=ud_f_euros(dofcontro  
l,1);
```

```
matrix_udd_f_euros(auks1,auks2)=udd_f_euros(dofcontrol,1);
```

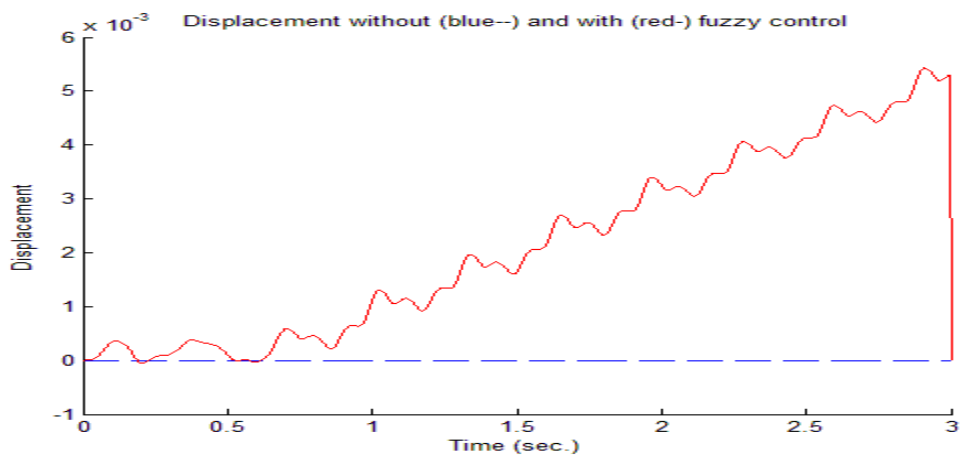
Τα ζευγάρια που μας δίνουν οι δύο for είναι 16, από τα οποία το καθένα μας δίνει από τρεις γραφικές για τα τρία μεγέθη που ελέγχουμε και πριν .

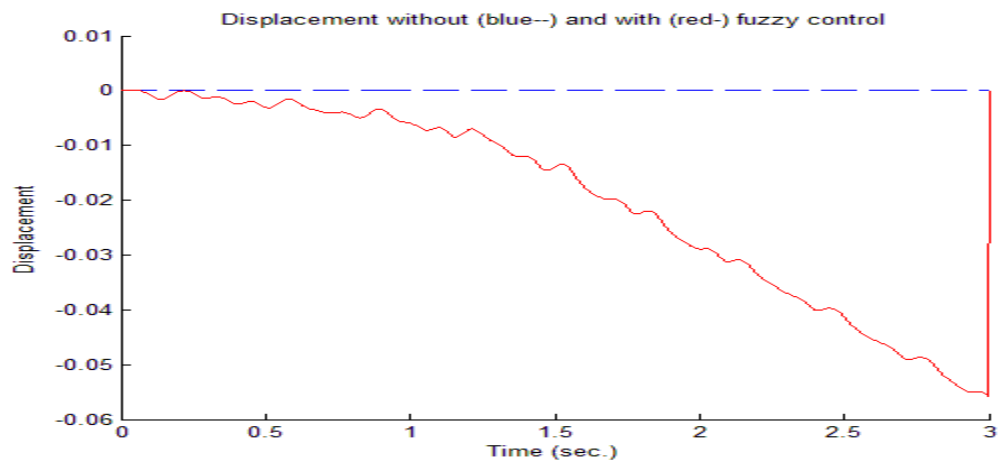
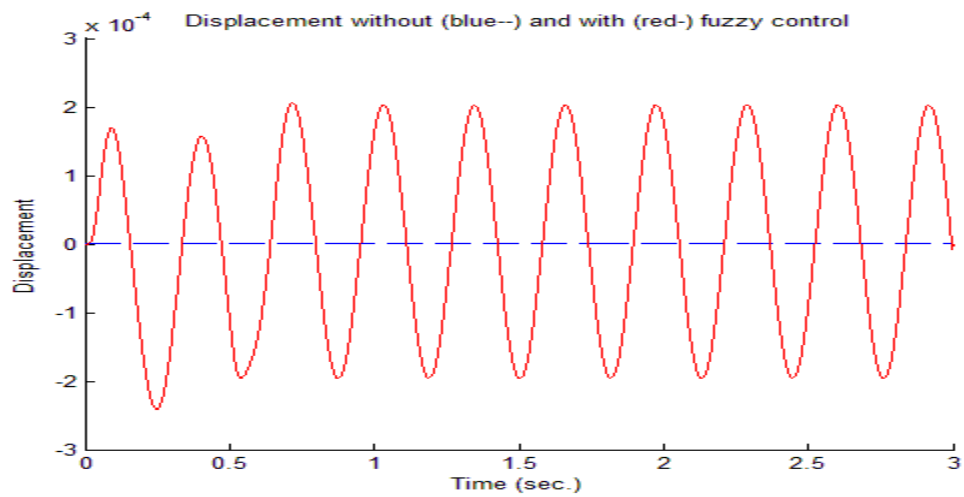
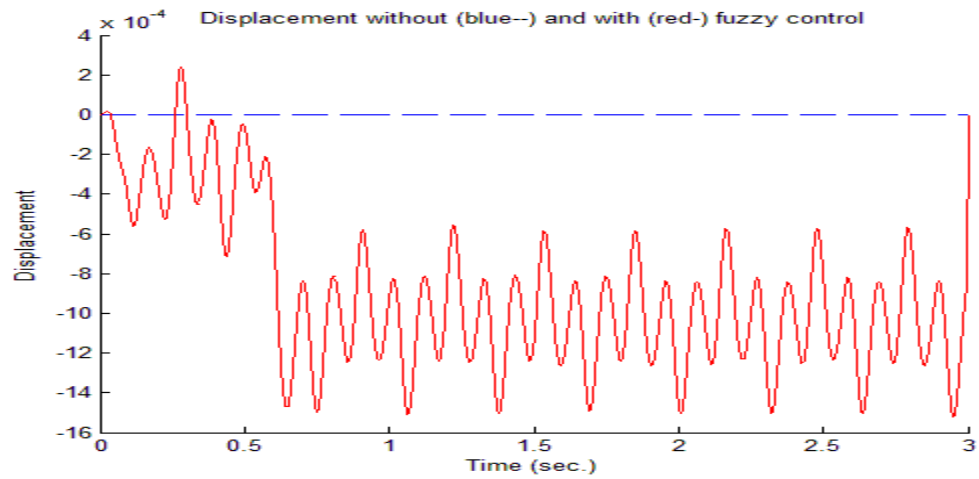
Τα ζευγάρια που παίρνουμε για την μετατόπιση είναι τα εξής:

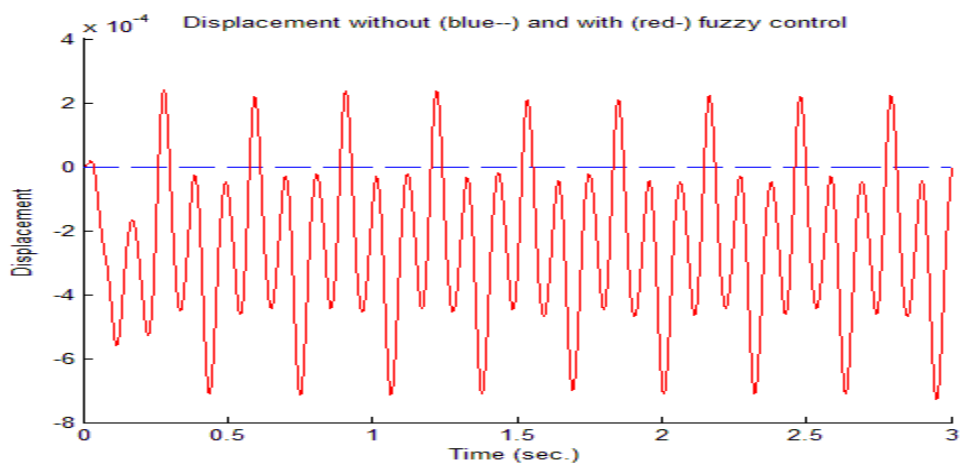
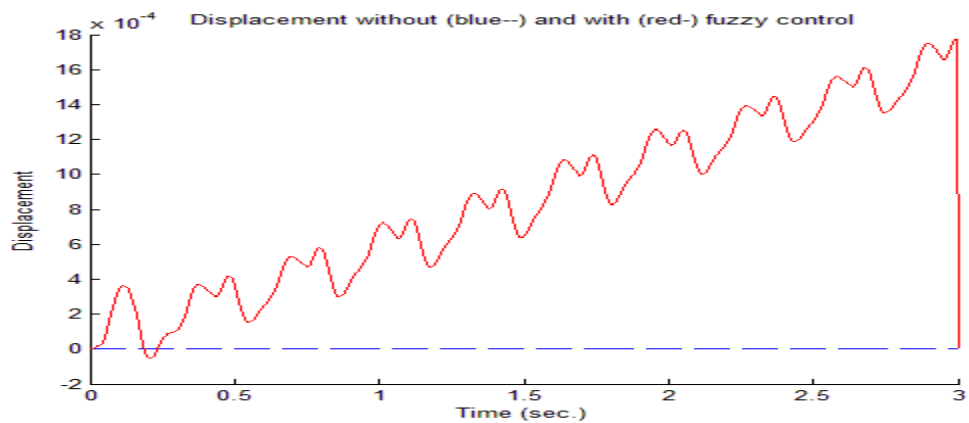
επαναλήψεις	U _{max}	u _{dmax}		ευρος
1η	0.001	0.0001	αστάθεια	$0,41587 \cdot 10^{-3}$
2η	0.001	0.0051	αστάθεια	$0,955 \cdot 10^{-3}$
3η	0.001	0.0101	ευστάθεια	0,082
4η	0.001	0.0151	αστάθεια	0,0058
5η	0.006	0.0001	αστάθεια	0,0027
6η	0.006	0.0051	αστάθεια	0,0032
7η	0.006	0.0101	ευστάθεια	0,095
8η	0.006	0.0151	αστάθεια	0,0149
9η	0.011	0.0101	αστάθεια	0,0605
10η	0.011	0.0101	αστάθεια	$0,411 \cdot 10^{-3}$
11η	0.011	0.0101	αστάθεια	$0,97 \cdot 10^{-3}$
12η	0.011	0.0101	αστάθεια	0,0018

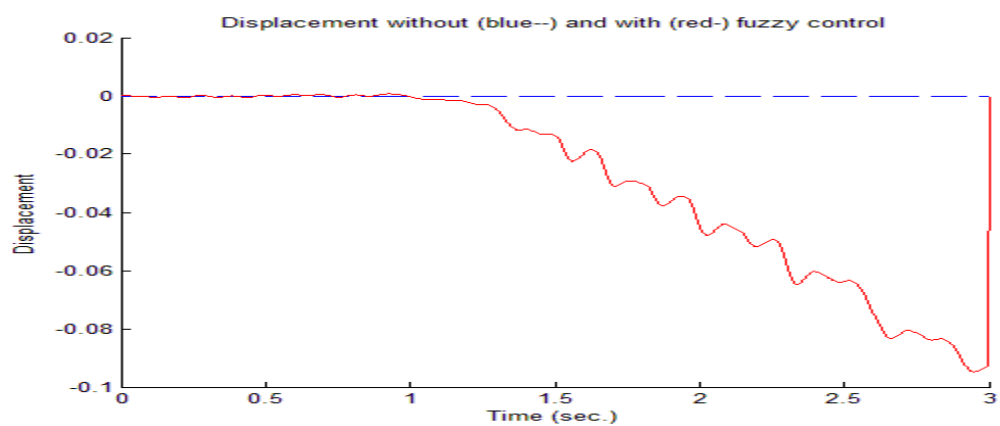
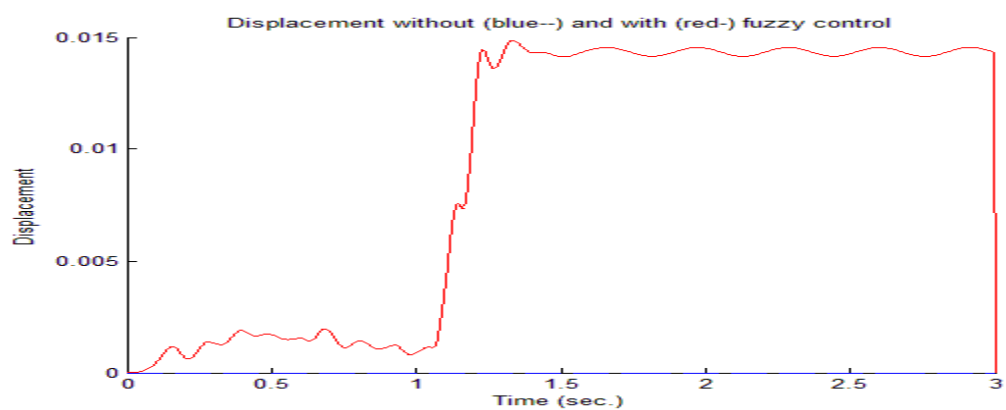
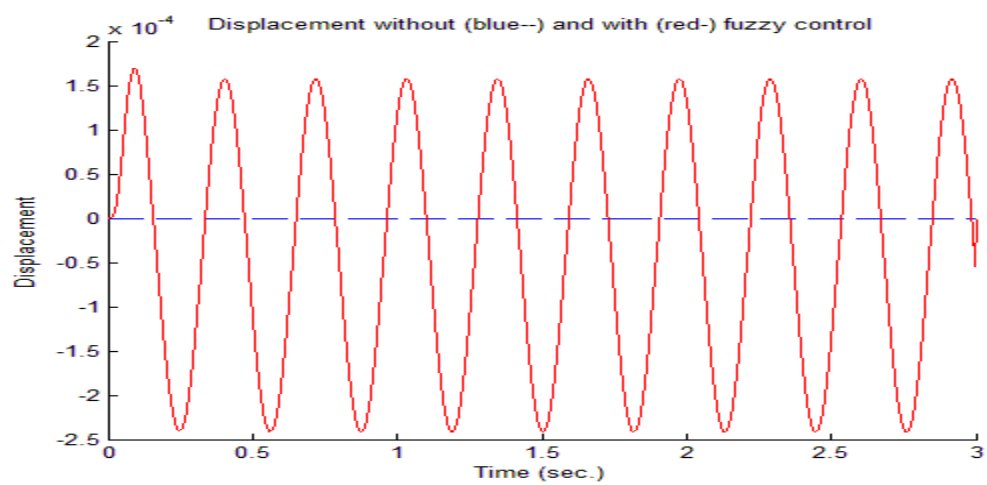
13η	0.016	0.0151	αστάθεια	0,056
14η	0.016	0.0151	αστάθεια	$0,445 \cdot 10^{-3}$
15η	0.016	0.0151	αστάθεια	0,0018
16η	0.016	0.0151	ευστάθεια	0,0055

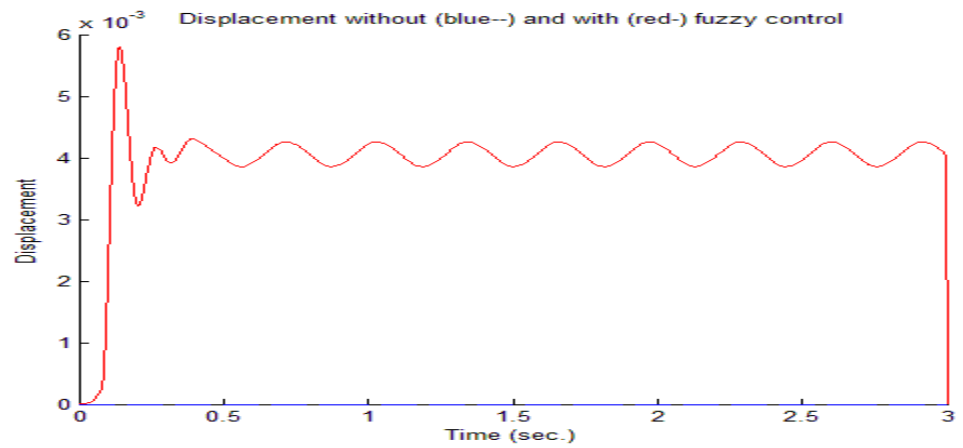
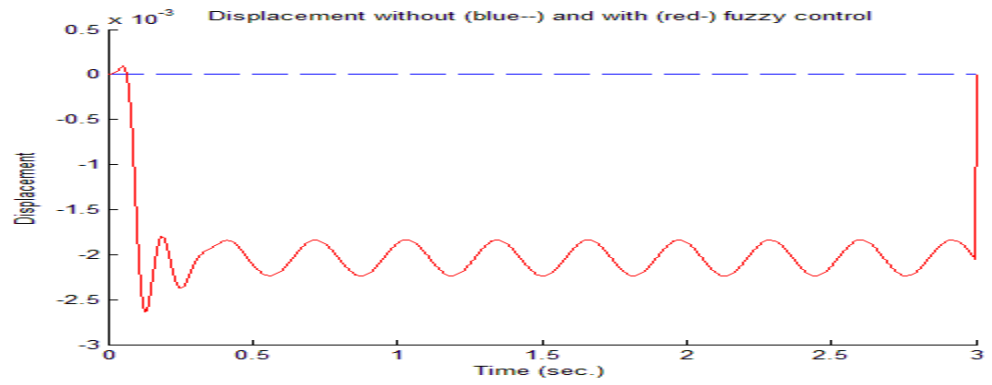
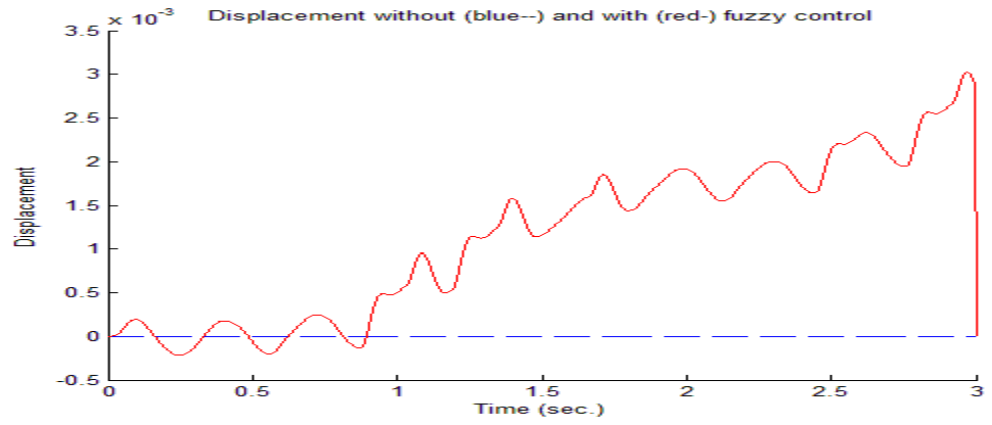
Παρατηρούμε ότι η 3^η, 7^η και 16^η γραφική παράσταση έχουν την μεγαλύτερη ευστάθεια από τις άλλες

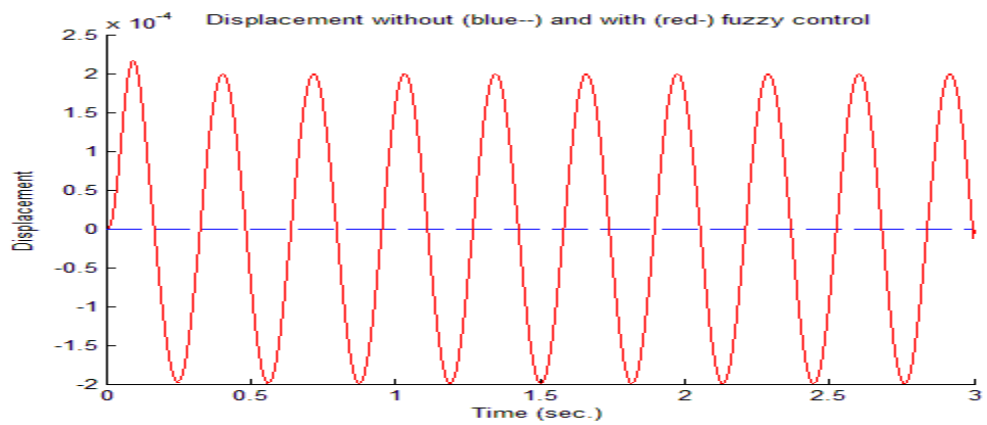
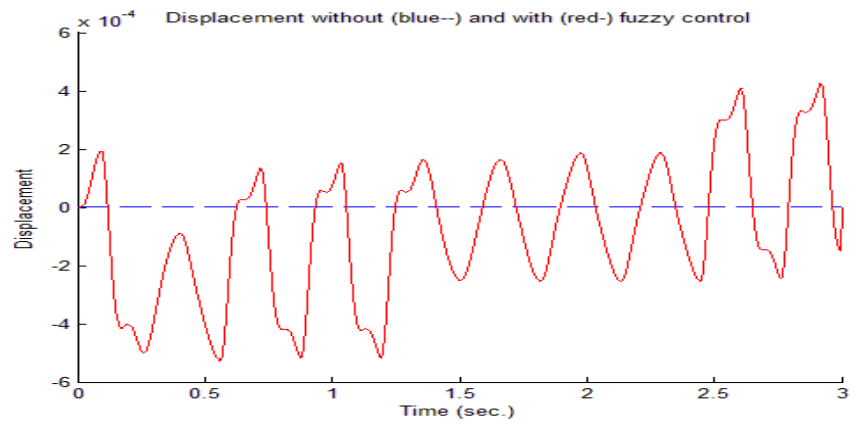
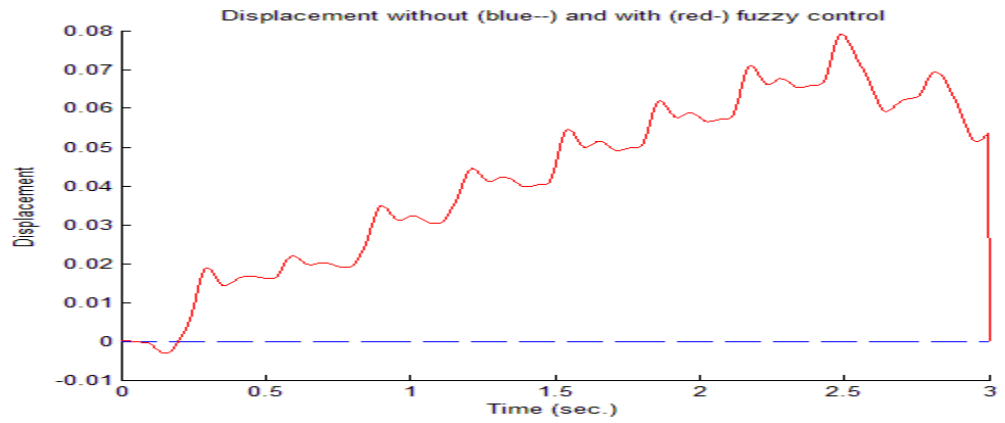








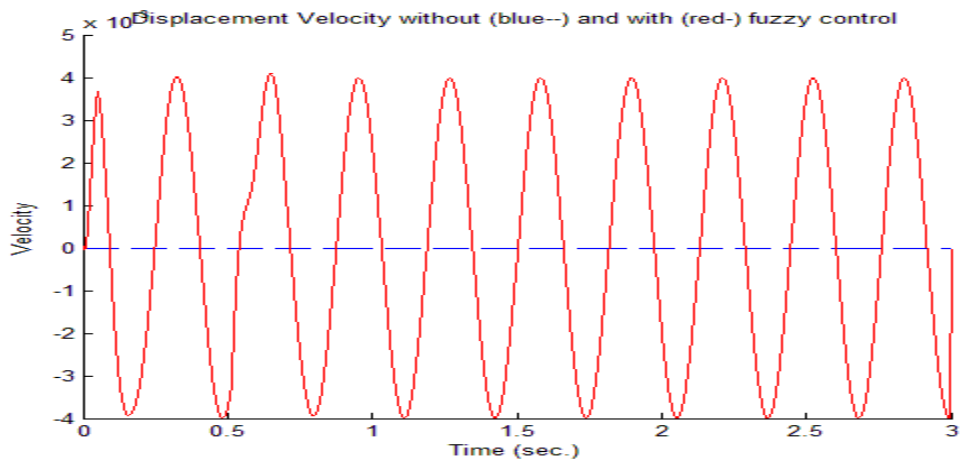
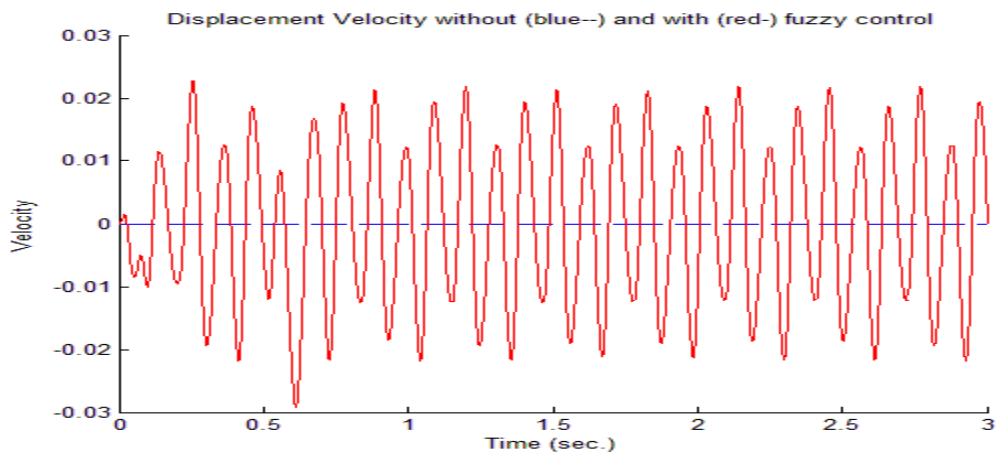
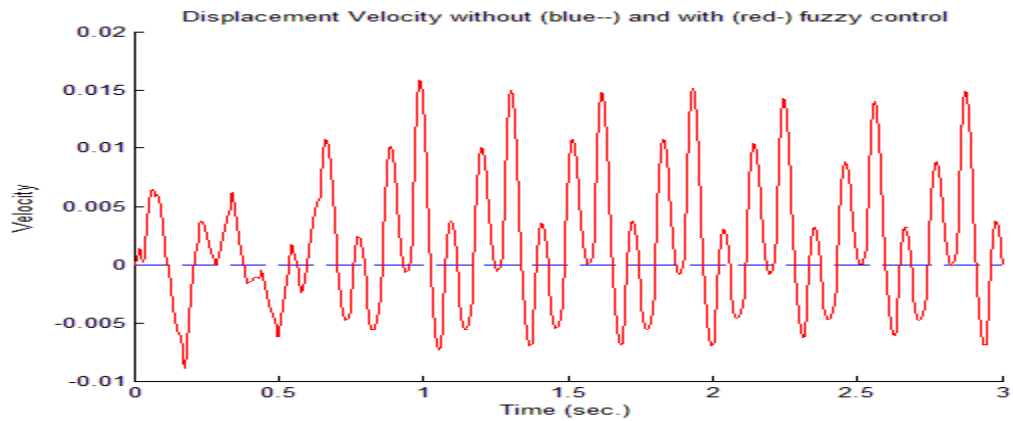


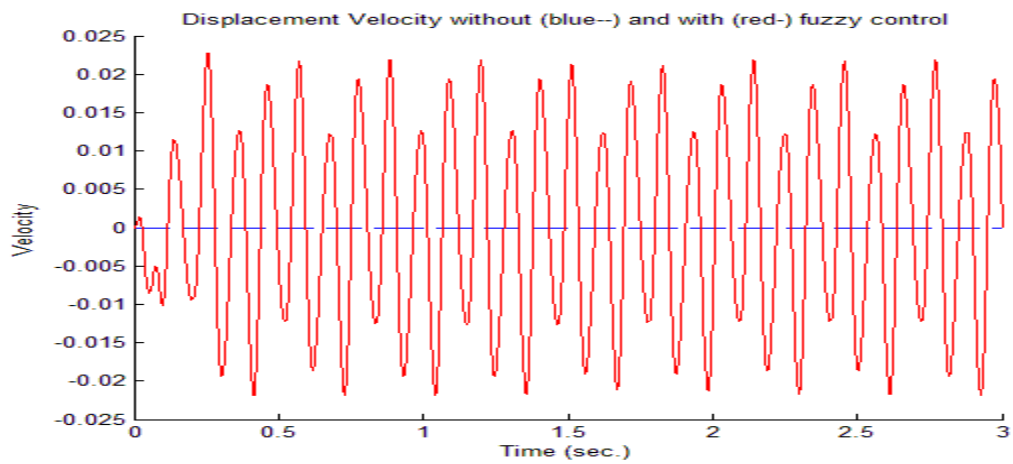
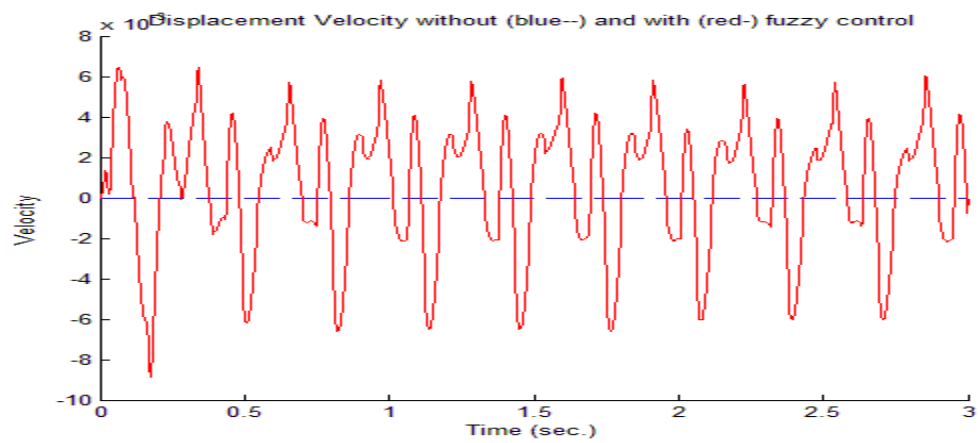
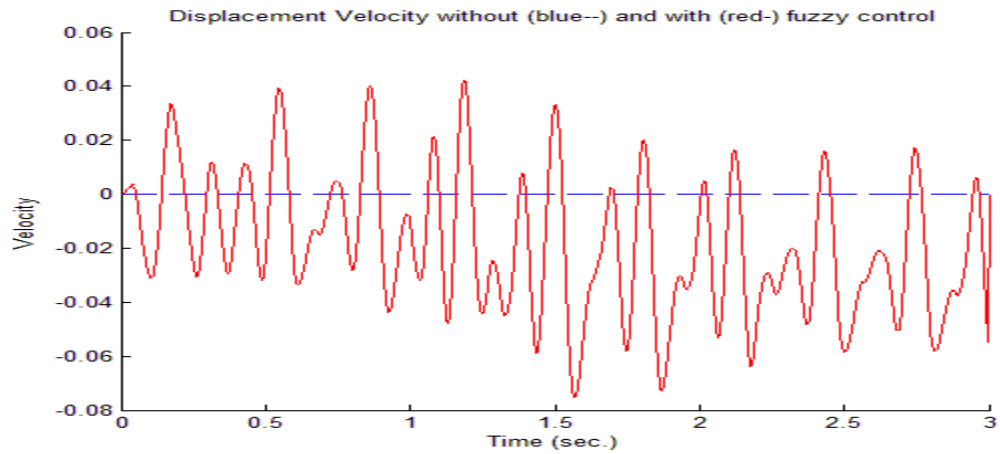


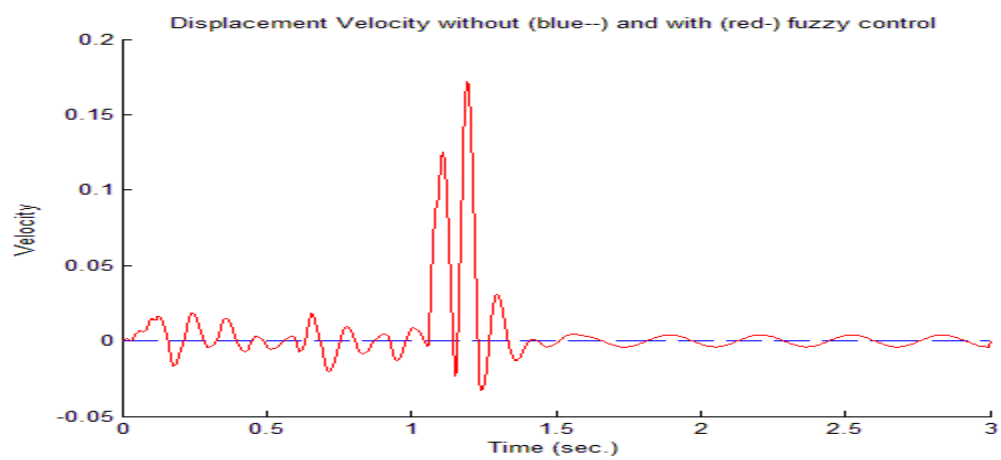
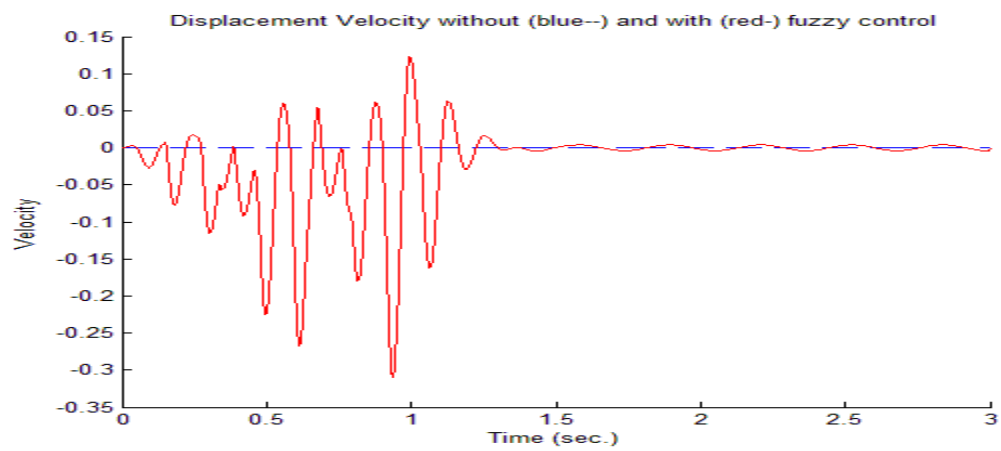
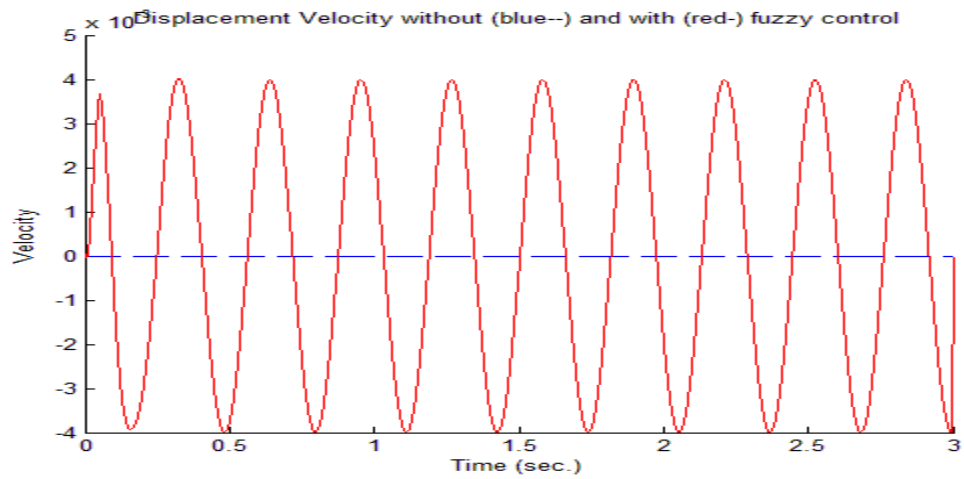
Τα ζευγάρια που παίρνουμε για την ταχύτητα είναι τα εξής:

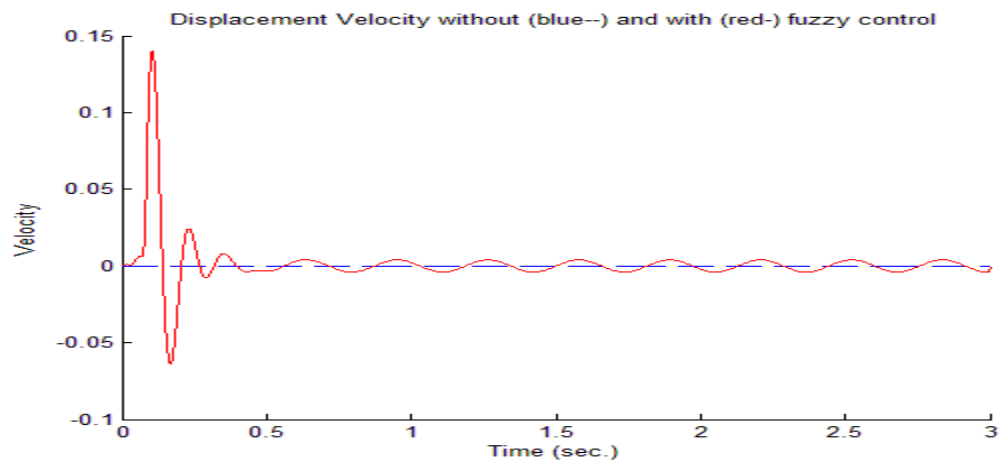
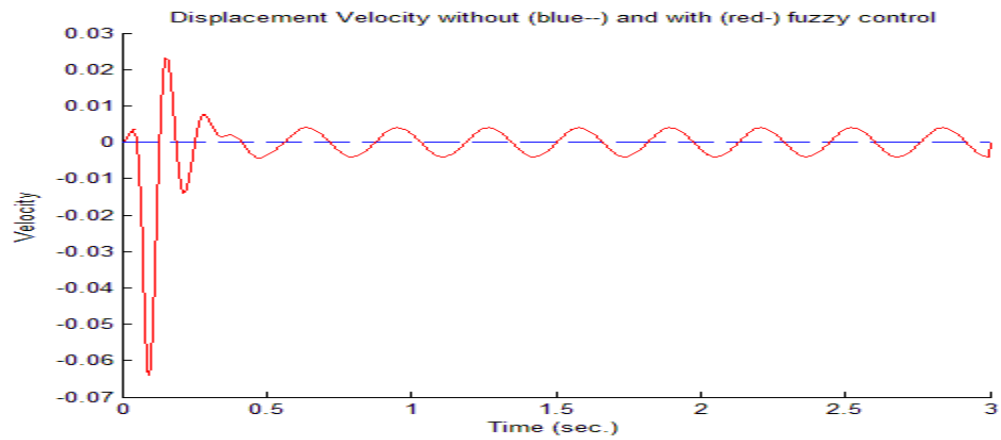
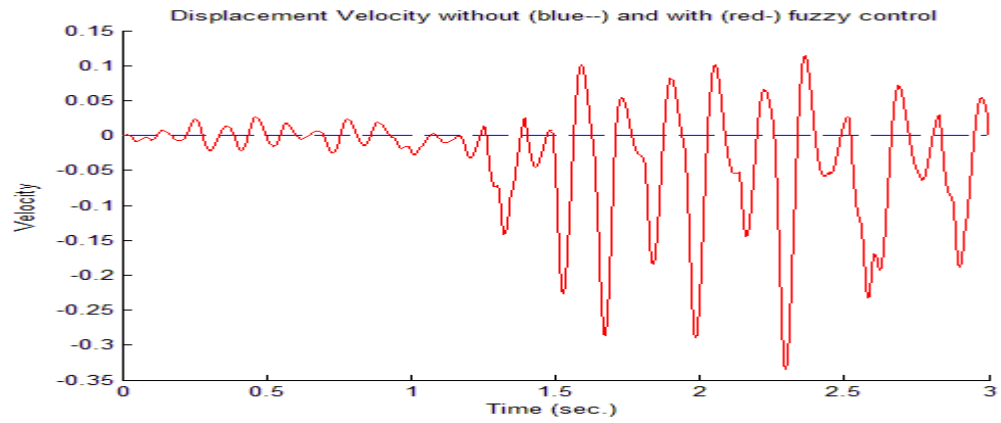
επαναλήψεις	u _{max}	u _{dmax}		ευρος
1η	0.001	0.0001	αστάθεια	0,0081
2η	0.001	0.0051	ευστάθεια	0,0256
3η	0.001	0.0101	ευστάθεια	0,5099
4η	0.001	0.0151	αστάθεια	0,2039
5η	0.006	0.0001	αστάθεια	0,0873
6η	0.006	0.0051	ευστάθεια	0,0231
7η	0.006	0.0101	ευστάθεια	0,4488
8η	0.006	0.0151	αστάθεια	0,2054
9η	0.011	0.0101	αστάθεια	0,4335
10η	0.011	0.0101	αστάθεια	0,0080
11η	0.011	0.0101	αστάθεια	0,0447
12η	0.011	0.0101	αστάθεια	0,0153
13η	0.016	0.0151	αστάθεια	0,1176
14η	0.016	0.0151	αστάθεια	0,0081
15η	0.016	0.0151	αστάθεια	0,052
16η	0.016	0.0151	ευστάθεια	0,0247

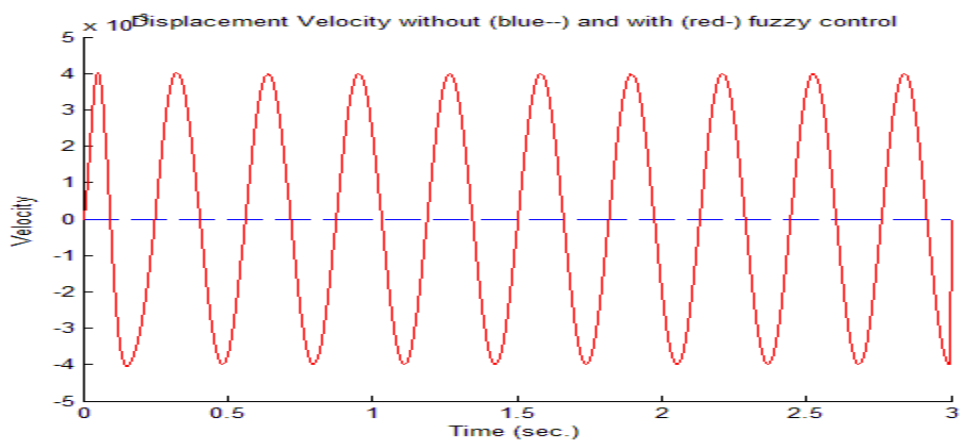
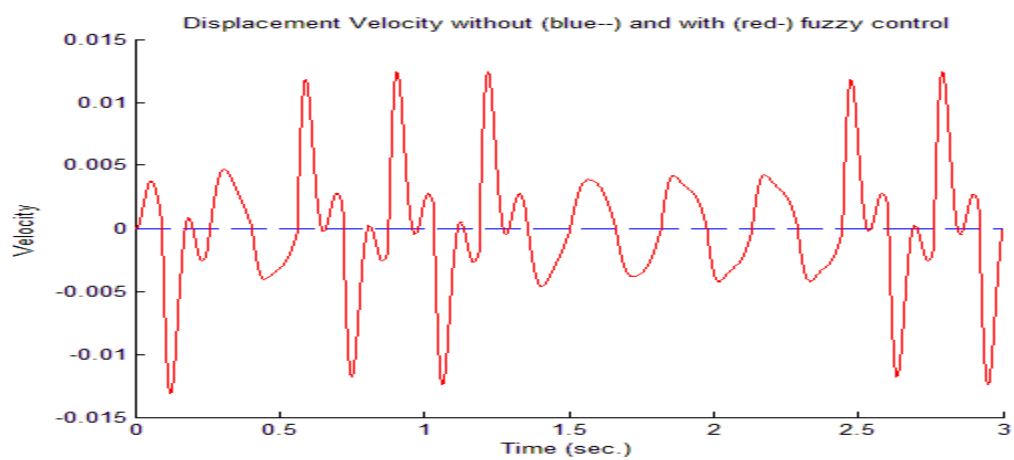
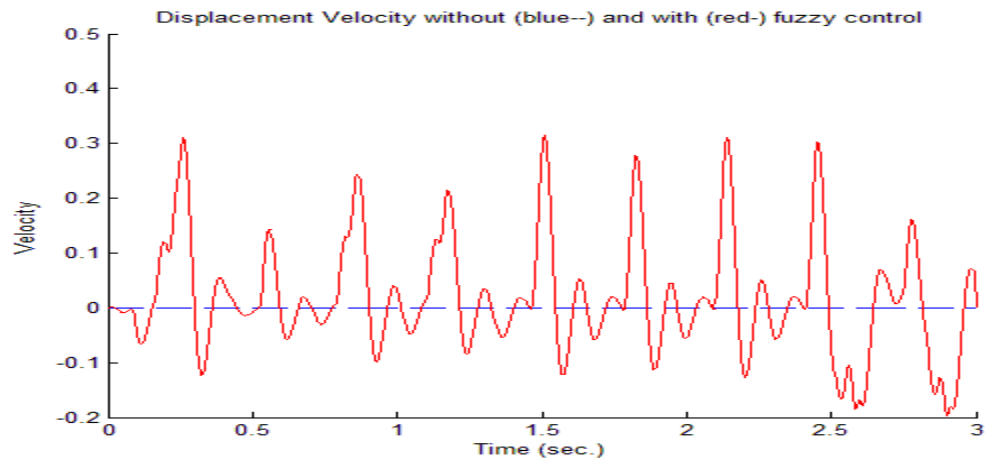
Παρατηρούμε ότι για την ταχύτητα παίρνουμε καλύτερα αποτελέσματα από πριν δηλαδή περισσότερες γραφικές παραστάσεις έχουν ευστάθεια (2^{η} , 3^{η} , 6^{η} , 7^{η} , 16^{η})







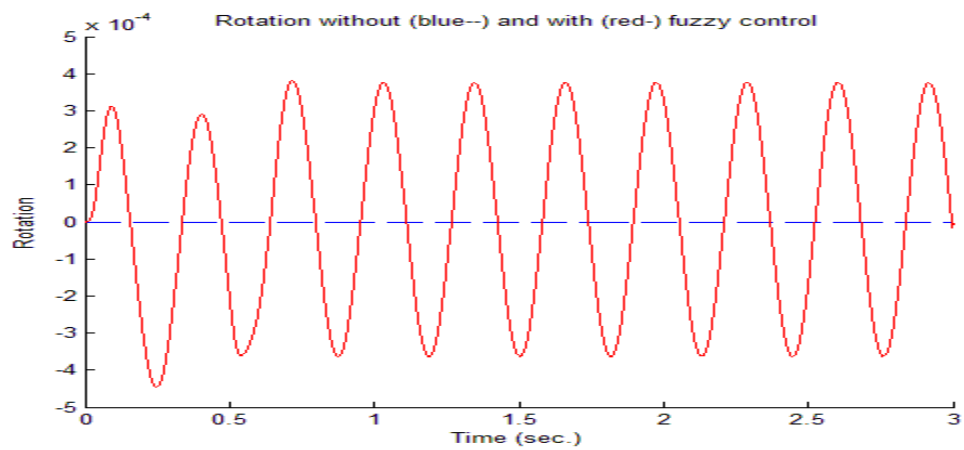
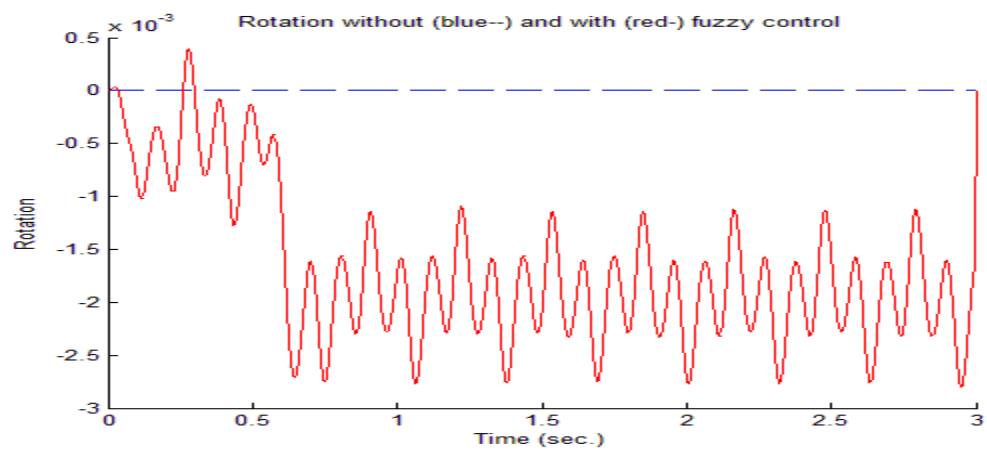
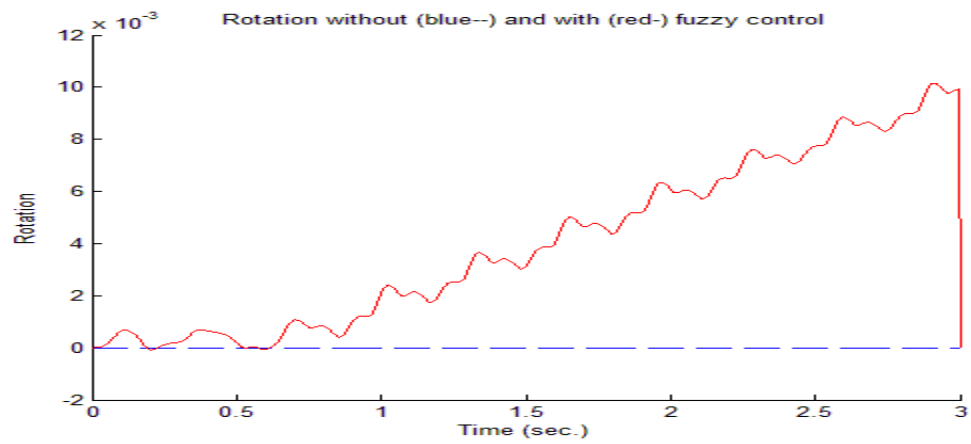


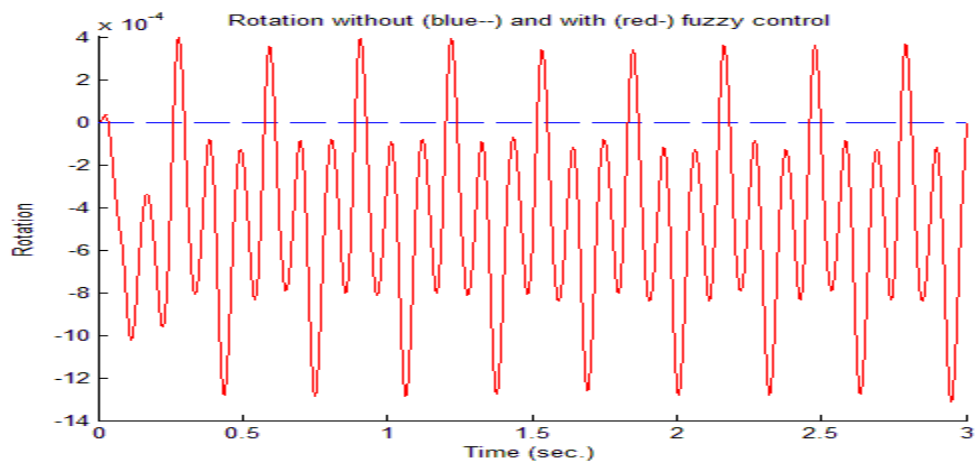
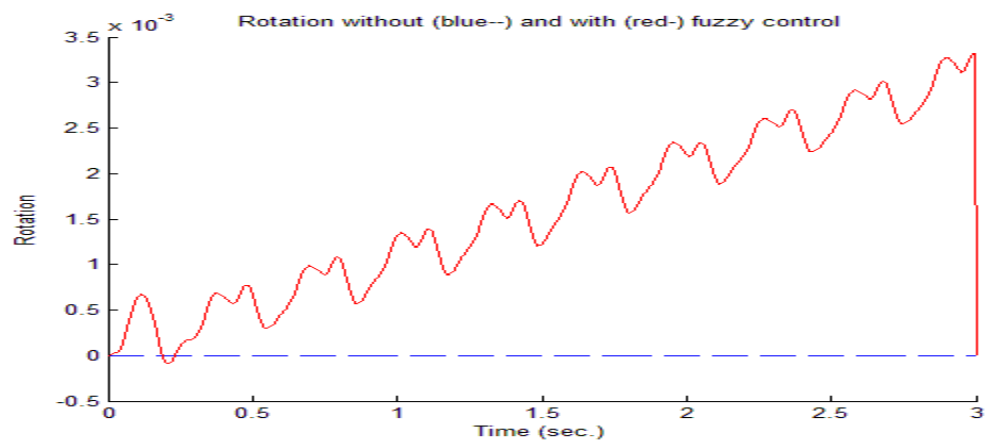
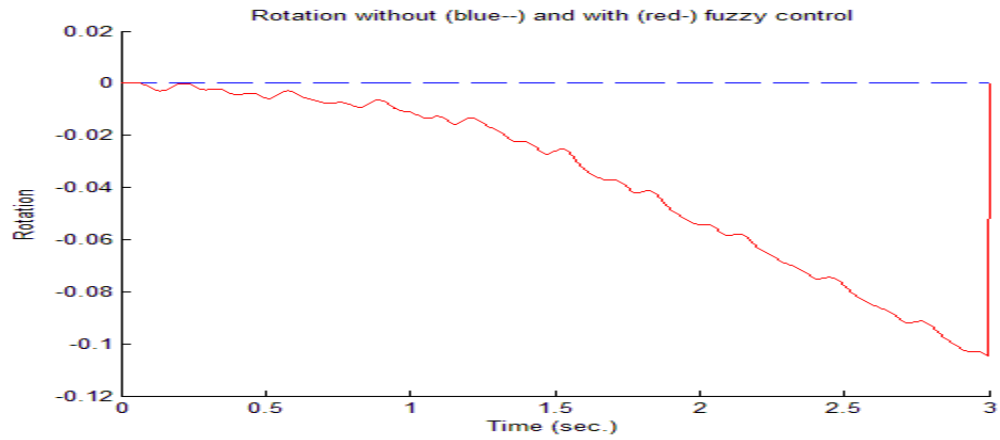


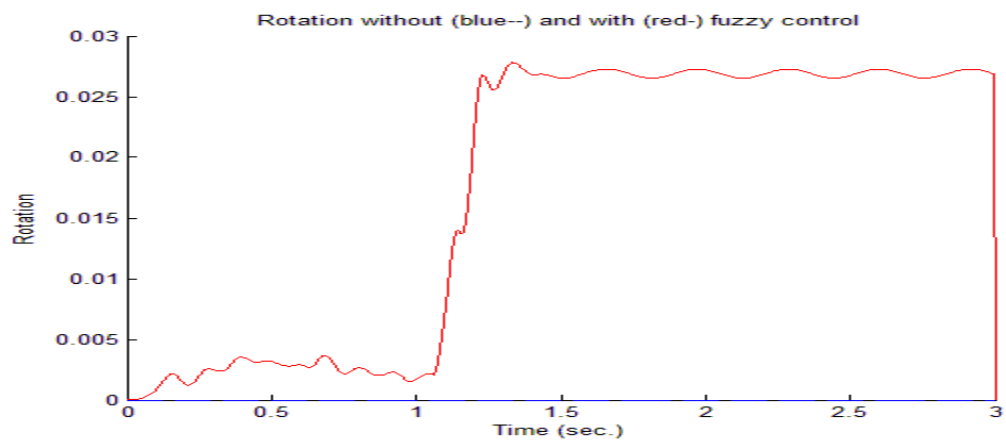
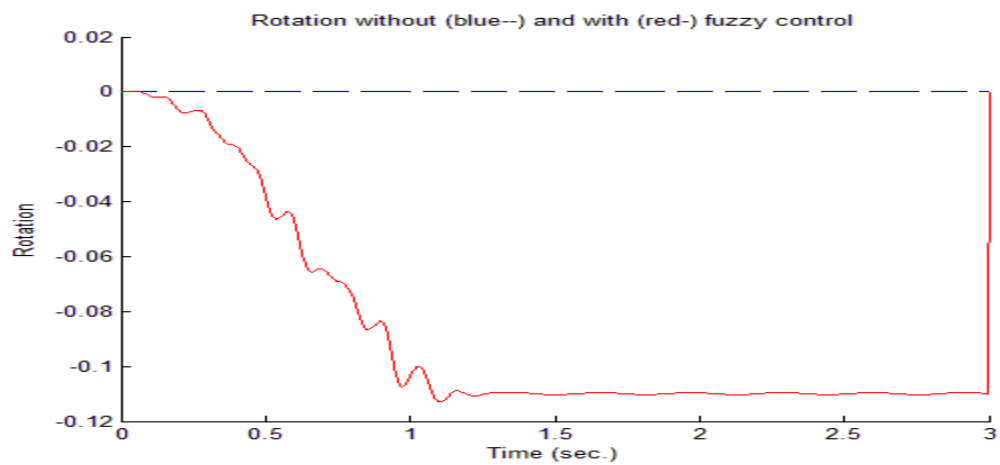
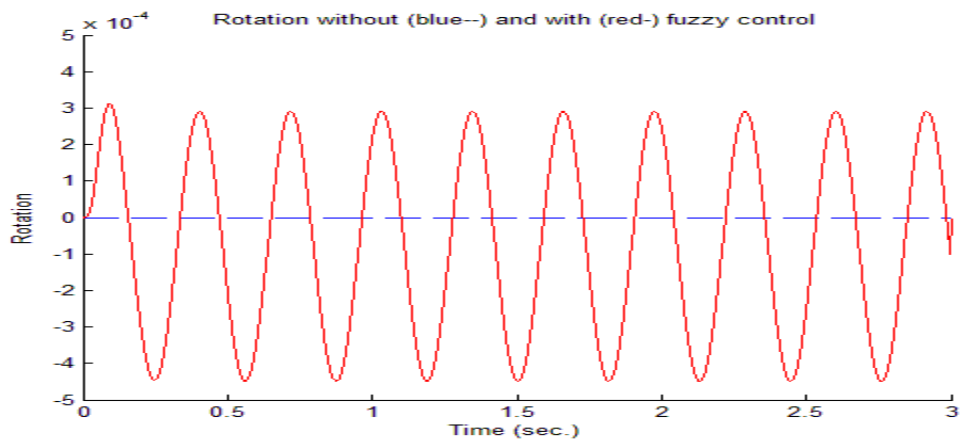
Παρατηρούμε ότι ευστάθεια έχουν η 3^η 7^η και 16^η προσπάθεια.

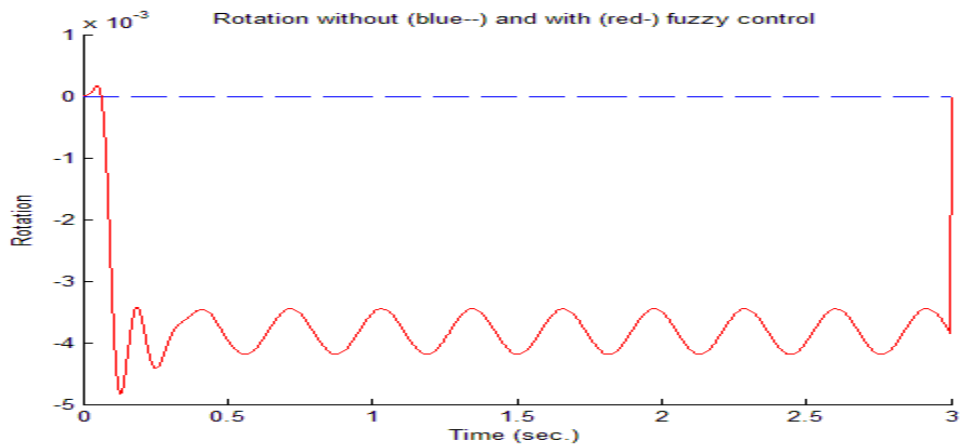
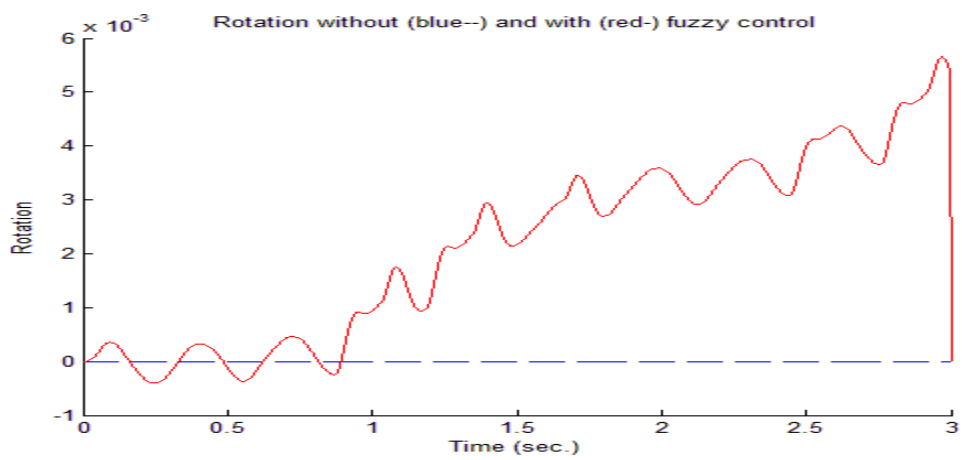
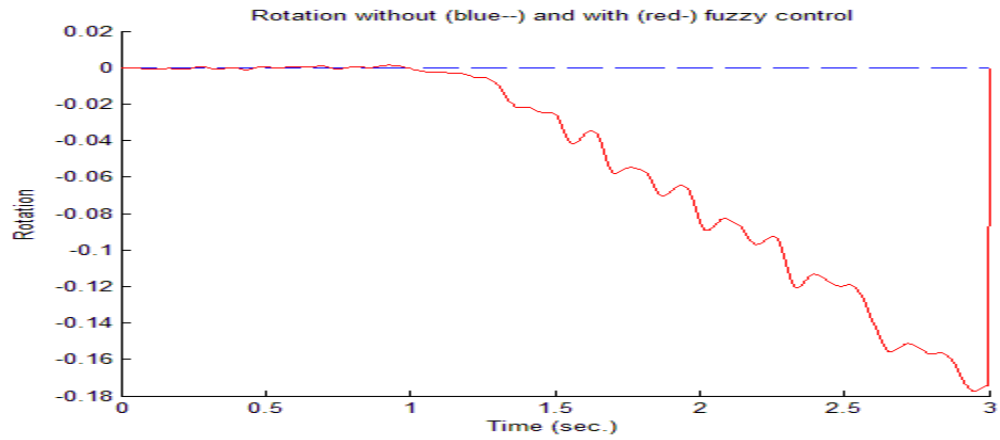
Τα αποτελέσματα που παίρνουμε για την επιτάχυνση είναι τα εξής :

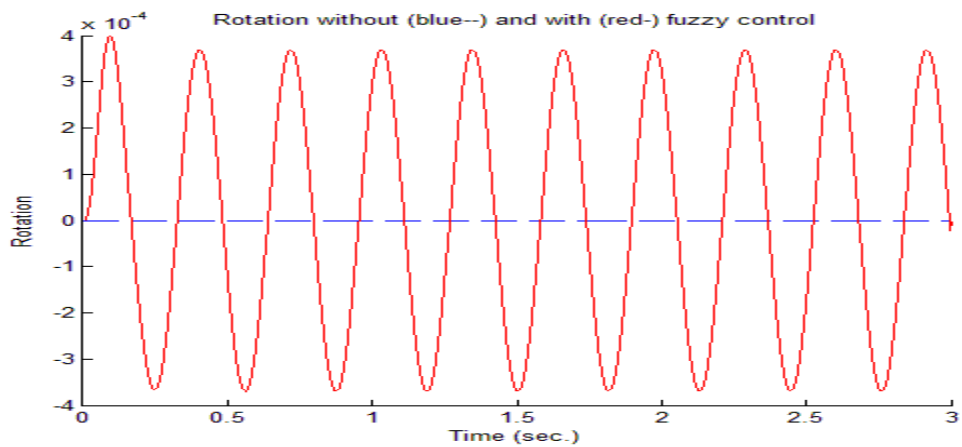
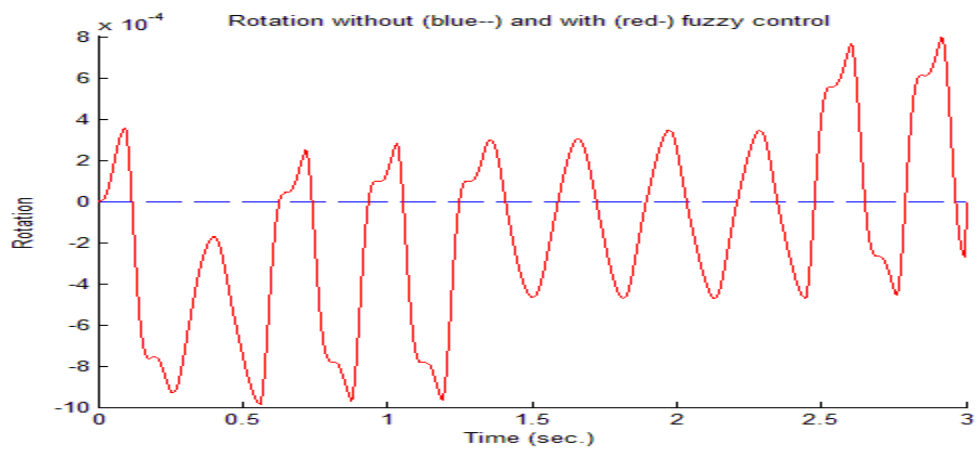
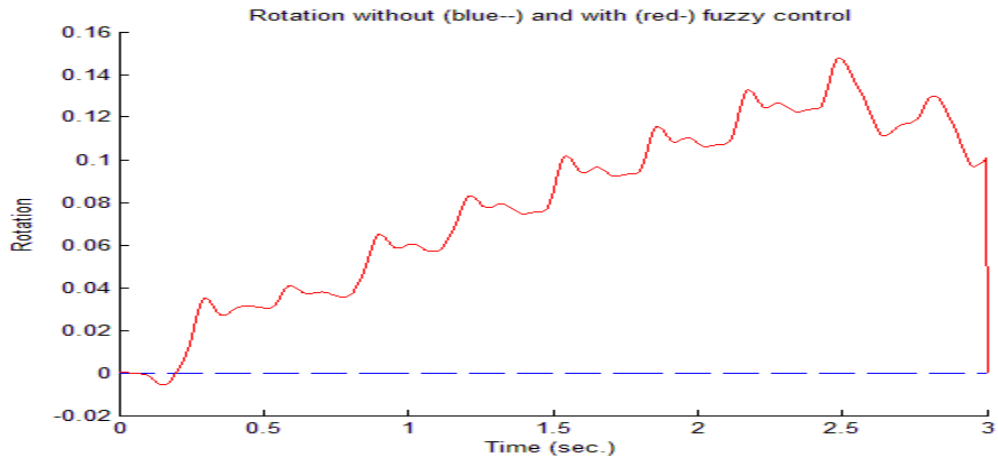
επαναλήψεις	u _{max}	u _{dmax}		ευρος
1η	0.001	0.0001	αστάθεια	0,2572
2η	0.001	0.0051	αστάθεια	3,05
3η	0.001	0.0101	ευστάθεια	21,8
4η	0.001	0.0151	αστάθεια	13,5
5η	0.006	0.0001	αστάθεια	6,2756
6η	0.006	0.0051	αστάθεια	2,13
7η	0.006	0.0101	ευστάθεια	22,6
8η	0.006	0.0151	αστάθεια	14,2
9η	0.011	0.0101	αστάθεια	22,8
10η	0.011	0.0101	αστάθεια	0,236
11η	0.011	0.0101	αστάθεια	2,96
12η	0.011	0.0101	αστάθεια	1,05
13η	0.016	0.0151	αστάθεια	5,12
14η	0.016	0.0151	αστάθεια	0,3987
15η	0.016	0.0151	αστάθεια	2,96
16η	0.016	0.0151	ευστάθεια	1,67











Συμπεραίνουμε ότι το πρώτο κριτήριο της ευστάθειας μονό 3^n , 7^n και 16^n προσπάθεια το επαληθεύουν απο τις τρεις όμως προσπάθειες η πιο βέλτιστη είναι η 16^n η οποία έχει και στα τρία μεγέθη πάντα το μικρότερο εύρος.

Τελικό συμπέρασμα από αυτό τον έλεγχο που κάναμε είναι ότι επιλέγουμε βέλτιστο ζευγάρι το (0,016 , 0,0151).

6. ΓΕΝΙΚΑ ΣΥΜΠΕΡΑΣΜΑΤΑ

1) Παρατηρήσαμε ότι στο σύστημα που έχουμε να διερευνήσουμε αισθητήρα –διεγέρτη βασικό ρόλλο παίρνει η θέση του αισθητήρα. Η οποία μας βοηθάει να βρούμε το u_{max} & ud_{max} τα οποία μπαίνουν σαν ορίσματα στη fuzzy initial. Μέσα στη ρουτίνα $a = \text{fuzzy_initial}(u_{max}, ud_{max}, pm)$. Εκεί γίνεται ο ορισμός του συστήματος ασαφής λογικής.

2) Είδαμε ότι καθώς τρέξαμε τον αλγόριθμο για τις 7 διαφορετικές θέσεις του αισθητήρα πήραμε αποτελέσματα τα οποία μας οδήγησαν μέσα από κάποια κριτήρια σε μια βέλτιστη θέση του αισθητήρα και συγκεκριμένα στην θέση 1.

3) Στην 2^η διερεύνηση που κάναμε , σταματήσαμε να βρίσκουμε το εύρος της μετατόπισης και της ταχύτητας χωρίς ασαφή έλεγχο μέσα από την διαδικασία που έκανε ο κύριος Ταϊρίδης αλλά δημιουργήσαμε ένα εύρος τιμών που θα ορίζονται αυτά τα δυο μεγέθη ,έτσι ώστε να ορίζονται τυχαία ζευγάρια. Από την διερεύνηση αυτή οδηγηθήκαμε σε κάποια αποτελέσματα τα οποία ήταν 1) $u_{max}=0.016$

2) $ud_{max}=0,0151$

3) $u_f_euros=0,0055$

4) $ud_f_euros=0,0247$

$$5) u_{dd_f_euros} = 1,67$$

Τα αποτελέσματα που βρήκαμε στην αρχική διερεύνηση που ψάχναμε την βέλτιστη θέση του αισθητήρα ήταν:

$$1) u_{max} = 0,0605 * 10^{-6}$$

$$2) u_{dmax} = 1,18 * 10^{-4}$$

$$3) u_{f_euros} = 0,8045 * 10^{-6}$$

$$4) u_{d_f_euros} = 0,52260 * 10^{-5}$$

$$5) u_{dd_f_euros} = 0,0645$$

Παρατηρούμε ότι τα αποτελέσματα που είχαμε βρει αρχικά ήταν καλύτερα από αυτά που βρήκαμε τώρα με τις τυχαίες τιμές που ορίσαμε το εύρος τιμών χωρίς ασαφή έλεγχο. Πράγμα που δεν σημαίνει ότι αυτός ο τρόπος των τυχαίων τιμών είναι λάθος γιατί θα υπάρχει κάποιο εύρος τιμών που θα δίνει καλύτερα αποτελέσματα.

Τέλος ελέγχουμε τα αποτελέσματα με αυτά τα κύριου Ταίριδη που είχε επιλέξει θέση 7 αισθητήρα.

$$1) u_{max} = 0,415 * 10^{-4}$$

$$2) u_{dmax} = 0,0081$$

$$3) u_{f_euros} = 0,22315 * 10^{-3}$$

$$4) u_{d_f_euros} = 0,0168$$

5)udd_f_euros=2,430

Παρατηρούμε ότι και εδώ έχουμε επιτύχει μικρότερο εύρος.

6. ΒΙΒΛΙΟΓΡΑΦΙΑ

**1. ΡΟΒΕΡΤΟΣ- Ε ΚΙΝΓΚ –‘ΕΥΦΥΗΣ ΕΛΕΓΧΟΣ’ –ΕΚΔΟΣΕΙΣ
:ΤΖΙΟΛΑ , 2004**

2. Γ. ΤΑΙΡΙΔΗΣ-΄Προσομοίωση σύνθετων ευφυών μηχανικών συστημάτων με αισθητήρες και διεγέρτες από πιεζοηλεκτρικά υλικά ΄Διατριβή για απόκτηση μεταπτυχιακού διπλώματος ειδίκευσης΄

3. Χ. Κιλιμπέρης ΄Παραμετρική διερεύνηση συστήματος ασαφούς ελέγχου με εφαρμογή σε πιεζοηλεκτρική ευφυή δοκό΄ διπλωματική εργασία.

4. M. Marinaki-Y. Marinakis- G. Stauroulakis-΄Fuzzy control optimized by a multi objective particle swarm optimization algorithm for vibration suppression of smart structure΄ research paper 12 August 2010