Αριστοτέλειο Πανεπιστήμιο Θεσσαλονίκης Τμήμα Φυσικής

Πρόγραμμα Μεταπτυχιακών Σπουδών Ειδίκευσης : Υπολογιστική Φυσική

Μελέτη Συστημάτων Lotcka - Volterra



Συγγραφέας : Παναγιωτακόπουλος Χρήστος Επιβλέπων : Μελετλίδου Ευθυμία

Θεσσαλονίκη 2008

II

ΠΕΡΙΕΧΟΜΕΝΑ

Α ΜΕΡΟΣ: Εισαγωγή	1
Κεφάλαιο 1 Εισαγωγή στα δυναμικά συστήματα	1
1.1 Αυτόμομα και μη αυτόμομα, συστόματα	1
1.1 Αυτονομά και μη αυτονομά ουστηματά	····1 2
1.3 Γραμμικοποίηση στην γειτονιά των σημείων ισοροσπίας	2
1.4 Γραμμικό σύστημα με σταθερούς συντελεστές	4
1.5 Ευστάθεια	5
Κεφάλαιο 2	
Εισαγωγή στη θεωρία διακλαδώσεων	9
1 Διακλάδωση Σάγματος -Κόμβου	10
 Υποκοίσιμη διακλάδωση (transcritical). 	10
 Διακλάδωση διγάλας (pitchfork) 	11
4. Διακλάδωση Hopf	12
Κεφάλαιο 3	
Μη γραμμικά συστήματα και χάος	13
3.1 Βασικές έννοιες	13
3.2 Εκθέτες Lyapunov	15
3.3 Φάσμα Fourier	16
Κεφάλαιο 4	
Εισαγωγή στη δυναμική πληθυσμών	18
	10
Ι. Μοντελο γεννησεων θανατων	18
Π. Μοντέλο ενοοειοικου ανταγωνισμου	19
Π. Μοντέλο διαειοικού ανταγωνισμού	19
	17
Β ΜΕΡΟΣ : Σύστημα Loteka - Volterra	20
1. Σημεία Ισορροπίας -Ιδιοτιμές -Διακλαδώσεις	20
2. Διακλαδώσεις	30
3. Δυναμική Μελέτη-Εμφάνιση Χαοτικής συμπεριφοράς-Θεώρημα του Silnikov.	32
	40
Δυμπερασματα	48
Παραρτημα Α	52
Παραρτημα Β (κωοικαζ)	34
Βιβλιογραφία	93

IV

Α ΜΕΡΟΣ : ΕΙΣΑΓΩΓΗ

Κεφάλαιο 1 Εισαγωγή στα δυναμικά συστήματα

Στο πρώτο κεφάλαιο θα ασχοληθούμε με τη λύση συστημάτων διαφορικών εξισώσεων (ΔE) πρώτης τάξης. Θα αναφέρουμε μερικούς βασικούς ορισμούς, θα μιλήσουμε για τα σημεία ισορροπίας (ΣΙ) και θα δούμε πώς η γραμμικοποίηση γύρω από τα ΣΙ, όταν ισχύει το θεώρημα Hartman-Grobman (H-G) και ευσταθούς πολλαπλότητας, μας δίνει πολύτιμες πληροφορίες για τη λύση του συστήματος.

1.1 Αυτόνομα και μη αυτόνομα συστήματα

Έστω ένα σύστημα ΔΕ πρώτης τάξης [2] της μορφής

$$\dot{x} = f_i(x_i, t)$$
, $i, j = 1, 2, 3, ...n.$ (1.1)

όπου τα x_i είναι συντεταγμένες πάνω σε μια n-διάστατη επιφάνεια M (η M μπορεί να είναι και ένας ευκλείδειος χώρος \mathbb{R}^n). Αν στο σύστημα (1.1) οι f_i δεν είναι άμεσα εξαρτημένες από το χρόνο t τότε το σύστημα (1.1) παίρνει τη μορφή

$$\dot{\mathbf{x}} = f_i(\mathbf{x}_i) \tag{1.2}$$

Το σύστημα (1.2) ονομάζεται αυτόνομο ενώ το σύστημα (1.1) μη αυτόνομο.

Οι λύσεις του μη αυτόνομου συστήματος εξαρτώνται από τις αρχικές x_{j0} συνθήκες και από τον αρχικό t_0 . Επομένως η λύση του συστήματος (1.1) θα είναι της μορφής

$$x_i(t) = x_i(t; x_{i0}, t_0)$$
(1.3)

όπου το t είναι παράμετρος του συστήματος και η λύση (1.3) αντιστοιχεί σε μια n+1 παραμετρική οικογένεια καμπυλών. Ο χώρος καταστάσεων του συστήματος είναι ο χώρος $M \times \mathbb{R}$ διάστασης n+1 (το \mathbb{R} προκύπτει από το ότι ο αρχικός χρόνος επηρεάζει την εξέλιξη του συστήματος).

Η μορφή της λύσης του συστήματος μπορεί να αναπαρασταθεί γραφικά στο χώρο καταστάσεων. Αποδεικνύεται πως αν οι $f_i(x_j,t)$ είναι παραγωγίσιμες συναρτήσεις των x_j,t τότε από κάθε σημείο του χώρου καταστάσεων διέρχεται μόνο μια λύση (1.3). Δηλαδή με την προϋπόθεση πως οι f_i είναι παραγωγίσιμες, η εξέλιξη της λύσης του συστήματος (1.1) είναι μονοσήμαντα ορισμένη. Οι διαδοχικές θέσεις της λύσης στο χώρο καταστάσεων αναπαρίστανται από μια καμπύλη που την ονομάζουμε τροχιά.

Η λύση του αυτόνομου συστήματος (1.2) είναι της μορφής

$$x_i(t) = x_i(t - t_0; x_{i0})$$
(1.4)

Όπως φαίνεται ο χώρος καταστάσεων είναι ο Μ. Η προϋπόθεση για να έχουμε μία και μόνο μία λύση από ένα σημείο του χώρου καταστάσεων σε συστήματα της μορφής (1.2) είναι τα δεξιά μέλη της (1.2) $f_i(x_j)$ να είναι παραγωγίσιμες συναρτήσεις των x_j και να υπάρχει ένα τουλάχιστον $i \in \{1, 2, ...n\} : f_i(x_j) \neq 0$.

(Σημ. Στην περίπτωση που $f_i(x_j) = 0$ για όλα τα i είναι δυνατόν από το ίδιο σημείο του χώρου καταστάσεων να διέρχονται περισσότερες από μία τροχιές).

Επειδή στη λύση του αυτόνομου συστήματος (1.4) ο αρχικός χρόνος t_0 εμφανίζεται μέσω του συμπλέγματος $t - t_0$, για τις ίδιες αρχικές x_{j0} και για διαφορετικό t_0 θα έχουμε την ίδια τροχιά, μόνο που η λύση του συστήματος θα διέρχεται από τα διαφορετικά σημεία της σε διαφορετικούς χρόνους (έτσι ώστε $t - t_0 = t' - t'_0$). Από τα προηγούμενα προκύπτει πως στα αυτόνομα συστήματα η μορφή της λύσης εξαρτάται μόνο από τις αρχικές συνθήκες x_j και είναι ανεξάρτητη από τον αρχικό χρόνο t_0 .

Στη συνέχεια θα ασχοληθούμε μόνο με αυτόνομα συστήματα της μορφής (1.2).

1.2 Σημεία Ισορροπίας

Σημεία ισορροπίας (ΣΙ) ενός συστήματος ΔΕ ονομάζονται εκείνα τα σημεία του χώρου καταστάσεων, τα οποία αν θεωρηθούν αρχικές συνθήκες για το σύστημα η τροχιά που προκύπτει παραμένει στο ίδιο το σημείο. Τέτοια σημεία είναι αυτά για τα οποία $\dot{x}_i = 0, \forall t$ και επομένως $f_i(x_i) = 0$ (1.5) $\forall i, i = 1, ...n$.

Επειδή στα μη γραμμικά συστήματα ΔΕ συνήθως δεν είναι εφικτό να βρεθεί η αναλυτική λύση τους, γίνεται ποιοτική μελέτη των ιδιοτήτων των λύσεων γύρω από τα σημεία ισορροπίας. Αυτό επιτυγχάνεται με τη γραμμικοποίηση του συστήματος στην περιοχή των σημείων ισορροπίας, και από τη λύση του γραμμικοποιημένου συστήματος με τη χρήση του θεωρήματος Hartman-Grobman προκύπτουν χρήσιμα συμπεράσματα για τις τοπολογικές ιδιότητες του αρχικού μη γραμμικού συστήματος.

1.3 Γραμμικοποίηση στην γειτονιά των σημείων ισορροπίας.

Έστω το αυτόνομο σύστημα [2],[7]

$$\dot{x}_i = f(x_i) \tag{1.2}$$

Όπως αναφέρθηκε τα σημεία ισορροπίας προκύπτουν από τις λύσεις του αλγεβρικού συστήματος

$$f_i(x_i) = 0 \quad \forall i = 1, 2, ...n$$
 (1.5)

Έστω πως μια λύση του συστήματος (1.5) είναι η $x^* = \{x_1^*, x_2^*, ..., x_n^*\}$ δηλαδή το x^* είναι ένα σημείο ισορροπίας. Τότε ένα γειτονικό σημείο στο x^* είναι το

$$x = x^* + \xi$$
 $\dot{\eta}$ $x_i = x_i^* + \xi_i$ (1.6)

Αφού το x ανήκει στην γειτονιά του x^* το ξ γίνεται οσοδήποτε μικρό θέλουμε. Σε αυτήν την περίπτωση μιλάμε για μια μικρή διαταραχή γύρω από το σημείο ισορροπίας.

Στη συνέχεια θα γίνει χρήση της θεωρίας διαταραχών.

Το σύστημα (1.2) για την μικρή διαταραχή (1.6) γίνεται

$$\dot{x}_{i}^{*} + \dot{\xi}_{i} = f_{i}(x_{i}^{*} + \xi_{i})$$

και αναπτύσσοντας σε σειρά Taylor προκύπτει

$$\dot{x}_{i}^{*} + \dot{\xi}_{i} = f(x_{i}^{*}) + \frac{\partial f_{i}}{\partial \xi_{i}} |_{x^{*}} * \xi_{i} + O(\xi_{i}^{2})$$
(1.7)

Στη σχέση (1.7) καθώς το ξ γίνεται πολύ μικρό οι όροι $O(\xi_i^2)$ δεύτερης τάξης γίνονται πολύ μικροί και ο εφαπτόμενος χώρος που ορίζεται από την (1.7) τείνει να συμπέσει στον M.

Κρατώντας τους γραμμικούς όρους της (1.7) και εξαιτίας της (1.5) προκύπτει

$$\dot{\xi}_{i} = \frac{\partial f_{i}}{\partial \xi_{j}} |_{x^{*}} \cdot \xi_{j} \qquad \text{h se morgh pinkawn} \qquad \dot{\xi} = \mathbf{A}\xi \qquad (1.8)$$

όπου τα στοιχεία του πίνακα Α είναι $A_{ij} = \frac{\partial f_i}{\partial x_j} \Big|_{x^*}$.

Είναι προφανές πως ο πίνακας Α είναι σταθερός γιατί είναι υπολογισμένος στο σημείο ισορροπίας x^{*}.

Το σύστημα (1.8) ονομάζεται προσαρτημένο γραμμικό σύστημα ή σύστημα εξισώσεων μεταβολών στο σημείο ισορροπίας x^* .

Πριν τη λύση του γραμμικού συστήματος (1.8) με σταθερούς συντελεστές είναι σκόπιμο να διατυπωθούν μερικοί ορισμοί και να αναφερθούν μερικά χρήσιμα θεωρήματα.

Για ένα μη γραμμικό σύστημα:

Αν x_0 είναι ένα σημείο ισορροπίας ενός συστήματος, x είναι ένα σημείο του χώρου καταστάσεων και ϕ_t η ροή του συστήματος τότε :

Τοπική ευσταθής πολλαπλότητα S:

 $S = \{ x \in V / \phi_t(x) \to x_0 \quad \kappa \alpha \theta \omega \varsigma \quad t \to +\infty \quad \phi_t(x) \in V \quad \gamma \iota \alpha \quad t \ge 0 \}$

Τοπική ασταθής πολλαπλότητα U:

 $U = \{x \in V / \phi_t(x) \to x_0 \quad \kappa \alpha \theta \omega \varsigma \quad t \to -\infty \quad \phi_t(x) \in V \quad \gamma \iota \alpha \quad t \le 0\}$ όπου V ανοιχτή περιοχή του x

δηλαδή τοπική ευσταθής πολλαπλότητα είναι ο γεωμετρικός τόπος των σημείων, τα οποία κάτω από τη ροή ϕ_i καταλήγουν στο \mathbf{x}_0 καθώς $t \to +\infty$, ενώ τοπική ασταθής πολλαπλότητα είναι ο γεωμετρικός τόπος των σημείων, τα οποία κάτω από τη ροή ϕ_i

katalúgoun sto \mathbf{x}_{0} , kabác $t \to -\infty$.

Σε ένα γραμμικό σύστημα η ευσταθής και ασταθής πολλαπλότητα βρίσκονται κατά μήκος των ιδιοδιανυσμάτων ή είναι γραμμικός συνδυασμός των ιδιοδιανυσμάτων (άρα είναι ευθείες ή γενικευμένα επίπεδα, που κάτω από τη ροή απεικονίζονται στον εαυτό τους) και ονομάζονται αναλλοίωτοι υποχώροι. Ένας αναλλοίωτος υποχώρος είναι ευσταθής E^s, αν οι ιδιοτιμές που αντιστοιχούν στα ιδιοδιανύσματα που αποτελούν τη βάση του έχουν πραγματικό μέρος αρνητικό και ασταθής E^u αν οι ιδιοτιμές που αντιστοιχούν στα ιδιοδιανύσματα που έχει πραγματικό μέρος θετικό.

Αποδεικνύεται ότι κοντά στο x_0 , η ευσταθής S και ασταθής U πολλαπλότητα του μη γραμμικού συστήματος, εφάπτονται αντίστοιχα στον E^s και E^u αναλλοίωτο υποχώρο του αντίστοιχου γραμμικού συστήματος.

Σύμφωνα με το θεώρημα Hartman-Grobman όταν δεν υπάρχουν ιδιοτιμές του γραμμικού συστήματος με μηδενικό πραγματικό μέρος, η τοπολογία του χώρου των φάσεων διατηρείται κατά τη μετάβαση από το γραμμικό στο μη γραμμικό σύστημα και συνεπώς διατηρείται η ευστάθεια ή η αστάθεια του σημείου ισορροπίας.

Σύμφωνα με το θεώρημα ευσταθούς πολλαπλότητας όταν δεν υπάρχουν ιδιοτιμές του γραμμικού συστήματος με πραγματικό μέρος μηδέν, διατηρούνται ο ευσταθής και ο ασταθής υποχώρος του συστήματος κατά τη μετάβαση από το γραμμικό στο μη γραμμικό σύστημα

1.4 Γραμμικό σύστημα με σταθερούς συντελεστές

Σε αυτή την παράγραφο θα αναφερθούμε στις ιδιότητες ενός γραμμικού συστήματος [2] με σταθερούς συντελεστές, της μορφής

$$\dot{x} = Ax \tag{1.9}$$

όπου Α σταθερός πίνακας και $x \in \mathbb{R}^n$.

Όπως στην περίπτωση της εξίσωσης $\dot{y} = ay$ αναζητούνται λύσεις της μορφής $y = e^{at}y_0$ έτσι και στην περίπτωση του συστήματος (1.9) αναζητούνται λύσεις της

μορφής $x(t) = e^{At} x_0$ με $x_0 = (x_{10}, x_{20}, ..., x_{n0})$, $x(0) = x_0$ για (t=0), όπου η εκθετική μορφή πίνακα ορίζεται ως εξής :

$$e^{At} = I + A + \frac{1}{2!}A^2 + \dots = \sum_{k=0}^{\infty} \frac{1}{k!}A^k$$
 $\mu\epsilon \quad A^0 = I$ (1.10)

$$\begin{cases} e^{PAP^{-1}} = PAP^{-1} \\ e^{A+B} = e^{A} \cdot e^{B} \Leftrightarrow AB = BA \\ \frac{d}{dt} e^{At} = Ae^{At} \end{cases}$$

με ιδιότητες :

Από την τρίτη ιδιότητα προκύπτει η λύση του συστήματος $x(t) = e^{At}x_0$ (1.10).

Επειδή όμως δεν είναι γνωστή η αναλυτική έκφραση συνάρτησης πίνακα χρησιμοποιούνται οι ιδιοτιμές του και τα ιδιοδιανύσματά του, και αυτό γιατί κατά μήκος των ιδιοδιανυσμάτων ενός πίνακα το αποτέλεσμα της δράσης του πίνακα στα ιδιοδιανύσματα είναι ισοδύναμο με τον πολλαπλασιασμό των ιδιοτιμών με το ιδιοδιάνυσμα.

Δηλαδή $Ax = \lambda x$, $\lambda \in \mathbb{C}$ από όπου συνεπάγεται $(A - \lambda I)x = O$ (1.11)

Επομένως για να βρεθούν οι ιδιοτιμές, αρκεί το ομογενές σύστημα (1.11) να έχει λύση $x \neq 0$, άρα πρέπει

$$\det(A - \lambda I) = 0 \tag{1.12}$$

Το *n* βαθμού σύστημα (1.12) έχει *n* πραγματικές ή μιγαδικές ρίζες (οι μιγαδικές εμφανίζονται κατά ζεύγη συζυγών).

Ανάλογα με τη μορφή των ιδιοτιμών και των ιδιοδιανυσμάτων η λύση του γραμμικού συστήματος κατηγοριοποιείται ως εξής :

Αν οι ιδιοτιμές είναι πραγματικές τότε

$$x(t) = P \begin{pmatrix} e^{\lambda_{1}t} & 0 \\ 0 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & e^{\lambda_{n}t} \end{pmatrix} p^{-1}x_{0} ,$$

όπου Ρ πίνακας με στήλες τα ιδιοδιανύσματα του Α.

Αν οι ιδιοτιμές είναι συζυγείς μιγαδικές τότε

$$x(t) = P \begin{pmatrix} e^{at}(\cos(\beta t)) & e^{at}(-\sin(\beta t)) & 0...0 \\ e^{at}(\sin(\beta t)) & e^{at}(\cos(\beta t)) & 0...0 \\ 0 & \ddots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & 0...0 \end{pmatrix} P^{-1}x_0$$

1.5 Ευστάθεια

Ευσταθές χαρακτηρίζεται το σημείο ισορροπίας x₀ για το οποίο, για κάθε γειτονιά του $N_{\varepsilon}(x_0)$ υπάρχει γειτονιά $N_{\delta}(x_0)$, μέσα από την οποία αν επιλέξουμε αρχικές συνθήκες, μετά από οποιονδήποτε πεπερασμένο χρόνο $t \ge 0$, $\phi(N_{\delta}) \in N_{\varepsilon}$. Ασταθές είναι το σημείο ισορροπίας x₀ το οποίο δεν είναι ευσταθές [7], [11].

Ασυμπτωτικά ευσταθές χαρακτηρίζεται ένα σημείο ισορροπίας, αν υπάρχει γειτονιά του, από την οποία αν επιλέξουμε αρχικές συνθήκες η τροχιά τείνει να πέσει πάνω στο σημείο ισορροπίας, για $t \to +\infty$. Περιοδική τροχιά είναι η τροχιά που επαναλαμβάνεται σε κάποιο χρόνο. Δηλαδή μετά από πεπερασμένο χρόνο η τροχιά θα επανέλθει στο σημείο από το οποίο ξεκίνησε x(t+T) = x(t).

Υπερβολικό είναι ένα σημείο ισορροπίας του οποίου η ορίζουσα δεν έχει καμιά ιδιοτιμή με πραγματικό μέρος μηδέν.

Όπως αναφέρθηκε, από το θεώρημα Hartman-Grobman προκύπτει πως αν ένα σημείο ισορροπίας x_0 είναι υπερβολικό, τότε το σημείο ισορροπίας διατηρεί την ευστάθεια του κατά τη μετάβαση από το γραμμικό στο μη γραμμικό σύστημα.

Συνέπεια των θεωρημάτων Hartman-Grobman και ευσταθούς πολλαπλότητας είναι το ότι, αν για ένα σημείο ισορροπίας ο πίνακας Α έχει έστω και μια ιδιοτιμή με πραγματικό μέρος θετικό ($\operatorname{Re}(\lambda_i) \ge 0$) τότε το σημείο ισορροπίας είναι ασταθές, ενώ αν για όλες τις ιδιοτιμές το πραγματικό μέρος είναι αρνητικό ($\operatorname{Re}(\lambda_i) < 0, \forall i = 1, 2, ..., n$), τότε το σημείο ισορροπίας είναι ασυμπτωτικά ευσταθές.

Στον \mathbb{R}^2 υπάρχουν τέσσερα είδη σημείων ισορροπίας. Ανάλογα με τις ιδιοτιμές του πίνακα Α προκύπτουν και διαφορετικές τοπολογικά ισοδύναμες τροχιές στο χώρο των φάσεων, που αντιστοιχούν σε διαφορετικό φασικό πορτρέτο στη γειτονιά του σημείου ισορροπίας.

<u>Όταν ο πίνακας του συστήματος είναι A(2x2).</u>

Πραγματικές ιδιοτιμές

1. An
$$\begin{cases} \lambda_i < 0, \forall i = 1, 2\\ \dot{\eta}\\ \lambda_i > 0, \forall i = 1, 2 \end{cases}$$
 tóte to shield eínal kómbog (node).

- 2. Αν υπάρχουν ετερόσημες ιδιοτιμές τότε το σημείο είναι σάγμα (Saddle).
- Μιγαδικές ιδιοτιμές
 - 1. Αν οι ιδιοτιμές είναι συζυγείς μιγαδικές με $\operatorname{Re}(\lambda_i) \neq 0$ τότε το σημείο είναι εστία (focus).
 - 2. Av $\operatorname{Re}(\lambda_i) = 0$ τότε το σημείο είναι κέντρο (center).

Όταν ο πίνακας του συστήματος είναι Α(3x3).

Οι ιδιοτιμές είναι τρεις και η θεωρία για τα πορτρέτα φάσης και το χαρακτηρισμό των σημείων, ως προς το είδος τους και την ευστάθεια που παρουσιάζουν,

επεκτείνεται. Σε αυτήν την περίπτωση είναι πολύ χρήσιμη η αναπαράσταση των ιδιοτιμών στο μιγαδικό επίπεδο. Οι δυνατές περιπτώσεις που μπορούν να προκύψουν παρουσιάζονται παρακάτω [13].



Εικόνα 1

Σε αυτές τις περιπτώσεις μπορούν να προστεθούν και οι συμμετρικές ως προς τον άξονα των φανταστικών ((a'),(b'),(c'),(d') και (e')).

- Πραγματικές ιδιοτιμές (Άμεση επέκταση της προηγούμενης περίπτωσης.)
 - 1. Περίπτωση (a) , $\lambda_i < 0, \forall i = 1, 2, 3$ τότε το σημείο είναι κόμβος.
 - Περίπτωση (b), ετερόσημες ιδιοτιμές τότε το σημείο είναι σάγμα με δυδιαστατό ευσταθή υποχώρο και μονοδιάστατο ασταθή. (Στην περίπτωση συμμετρικών περιπτώσεων (a'),(b') ο κόμβος είναι ασταθής και η διάσταση των υποχώρων αντίστροφη).

- Μιγαδικές ιδιοτιμές
 - Στις περιπτώσεις (c), (d) το σημείο ισορροπίας είναι συνδυασμός των περιπτώσεων κόμβου και εστίας. Στον υποχώρο που ορίζεται από τα ιδιοδιανύσματα των συζυγών ιδιοτιμών το σημείο ισορροπίας παρουσιάζει συμπεριφορά ευσταθούς εστίας (ασταθούς στις περιπτώσεις (c'), (d')) και στη διεύθυνση που ορίζεται από το ιδιοδιάνυσμα που αντιστοιχεί στην πραγματική ιδιοτιμή το σημείο ισορροπίας παρουσιάζει ευστάθεια ίδια με την ευστάθεια του υποχώρου που αντιστοιχεί στις μιγαδικές ιδιοτιμές . Η περίπτωση c (Re(λ₁₂)<λ₃<0) παρουσιάζεται στο (i) και η περίπτωση d (λ₃<Re(λ₁₂)<0) παρουσιάζεται στο (ii).



Εικόνα 2

2. Στην περίπτωση ε σημείο ισορροπίας συμπεριφέρεται σα συνδυασμός ευσταθούς εστίας (ασταθούς στην ε') και σάγματος. Η συμπεριφορά της εστίας παρατηρείται στον υποχώρο που ορίζουν τα ιδιοδιανύσματα των μιγαδικών ιδιοτιμών, ενώ στη διεύθυνση που ορίζεται από το ιδιοδιάνυσμα που αντιστοιχεί στην πραγματική ιδιοτιμή το σημείο ισορροπίας παρουσιάζει ευστάθεια αντίθετη από την ευστάθεια του υποχώρου που αντιστοιχεί στις μιγαδικές ιδιοτιμές. Αυτά τα σημεία ισορροπίας αναφέρονται ως σάγμα-εστία (Saddle-focus). Το πορτρέτο φάσης είναι το παρακάτω.



Εικόνα 3

Κεφάλαιο 2 Εισαγωγή στη θεωρία διακλαδώσεων

Επειδή σε πολλά φυσικά συστήματα η τιμή μιας σταθεράς μεταβάλλεται ανάλογα με τις συνθήκες (π.χ. ο ρυθμός γεννήσεων σε ένα πληθυσμό), είναι χρήσιμο να μελετηθεί πως μεταβάλλονται τα χαρακτηριστικά των συστημάτων συναρτήσει της παραμέτρου[2],[7].

Έστω μια οικογένεια δυναμικών συστημάτων με παράμετρο το μ

$$\dot{x} = f_i(x_i, \mu), \ i, j = 1, 2, ..., n$$
 (1.13)

Και έστω ένα σημείο ισορροπίας για κάποια τιμή της παραμέτρου $\dot{x} = f_i(x_i^*, \mu_0) = 0$.

Παρακάτω θα μελετήσουμε αν διατηρείται το σημείο ισορροπίας του συστήματος και σε άλλα μέλη της οικογένειας δυναμικών συστημάτων (1.13) καθώς μεταβάλλεται η τιμή της παραμέτρου μ.

• Av to shift isoppoptias $f_i(x_j^*, \mu_0) = 0$ είναι υπερβολικό (δηλ. αν όλες οι $\partial f_i = \frac{\partial f_i}{\partial t_i}$

ιδιοτιμές του πίνακα $\frac{\partial f_i}{\partial x_j}\Big|_{x^*}$ έχουν πραγματικό μέρος διάφορο του μηδενός),

τότε το θεώρημα πεπλεγμένων συναρτήσεων (implicit function theorem) μας βεβαιώνει πως το σημείο ισορροπίας συνεχίζεται με μοναδικό και συνεχή τρόπο και στα υπόλοιπα μέλη της οικογένειας διαφορικών εξισώσεων (1.13).

Κατά συνέπεια τα υπερβολικά σημεία ισορροπίας, για συνεχή μεταβολή της παραμέτρου διατηρούν την ευστάθεια τους.

- Αν το σημείο ισορροπίας είναι μη υπερβολικό με λ_i = 0 (δηλ. Re(λ_i) = 0 και Im(λ_i) = 0, ∀i = 1, 2, ...n), τότε δεν συνεχίζεται με μοναδικό τρόπο και μπορεί να εμφανίζεται αλλαγή ευστάθειας και πιθανώς αλλαγή του αριθμού των σημείων ισορροπίας.
- Αν το σημείο ισορροπίας είναι μη υπερβολικό με Re(λ_i) = 0 και Im(λ_i) ≠ 0, τότε δεν παρατηρείται αλλαγή του αριθμού των σημείων ισορροπίας αλλά ίσως εμφανιστεί οριακός κύκλος. Οριακός κύκλος είναι μια περιοδική τροχιά στην οποία τείνουν να καταλήξουν τροχιές από την γειτονιά της. Όταν αυτό συμβαίνει για $t \rightarrow +\infty$ ο οριακός κύκλος είναι ευσταθής, ενώ όταν αυτό συμβαίνει για $t \rightarrow -\infty$ ο οριακός κύκλος είναι ασταθής.

Τα μη υπερβολικά σημεία ισορροπίας ονομάζονται σημεία διακλάδωσης. Στα σημεία διακλάδωσης για μικρές μεταβολές της παραμέτρου μ παρουσιάζονται ουσιώδεις μεταβολές των τροχιών στο χώρο καταστάσεων. Η εποπτική αναπαράσταση αυτής της ουσιωδώς διαφορετικής τοπολογίας στο χώρο καταστάσεων γίνεται με τα διαγράμματα διακλάδωσης.

Διάγραμμα διακλάδωσης είναι τα διαγράμματα πάνω στα οποία εμφανίζονται οι μεταβολές στη θέση, στην ευστάθεια και στον αριθμό των σημείων ισορροπίας συναρτήσει της παραμέτρου μ.

Σε όλα τα παρακάτω μ_0 είναι η τιμή της παραμέτρου για την οποία παρουσιάζεται διακλάδωση.

1. Διακλάδωση Σάγματος -Κόμβου

Σε αυτήν την διακλάδωση για $\mu < \mu_0$ δεν υπάρχει κανένα σημείο ισορροπίας, για $\mu = \mu_0$ εμφανίζεται ένα και για $\mu > \mu_0$ παρουσιάζονται δύο, το ένα είναι ευσταθής κόμβος και το άλλο σάγμα. Στο διάγραμμα 1 φαίνεται το διάγραμμα διακλάδωσης για την αντιπροσωπευτική οικογένεια συστημάτων $\begin{cases} \dot{x} = \mu - x^2 \\ \dot{y} = -y \end{cases}$. Η διακλάδωση εμφανίζεται στο $\mu_0 = 0$.



2. Υποκρίσιμη διακλάδωση (transcritical)

Αντιπρόσωπος της οικογένειας δυναμικών συστημάτων στα οποία εμφανίζεται υποκρίσιμη διακλάδωση είναι το $\begin{cases} \dot{x} = \mu x - x^2 \\ \dot{y} = -y \end{cases}$ Η διακλάδωση εμφανίζεται στο $\mu_0 = 0, \ \gamma \text{i} \alpha \quad \mu < 0 \quad \text{υπάρχουν δύο σημεία ισορροπίας, ένα σάγμα και ένας ευσταθής κόμβος, για μ = 0 τα δύο σημεία ταυτίζονται και για μ > 0 εμφανίζονται εκ' νέου δύο σημεία, ένα σάγμα και ένας ευσταθής κόμβος τα οποία έχουν αλλάξει μεταξύ τους την ευστάθειά τους.$



3. Διακλάδωση διχάλας (pitchfork)

Χαρακτηριστικός αντιπρόσωπος αυτής της διακλάδωσης είναι το σύστημα $\begin{cases} \dot{x} = \mu x - x^3 \\ \dot{y} = -y \end{cases}$. Για $\mu < 0$ υπάρχει ένα σημείο ισορροπίας, το οποίο είναι ευσταθής κόμβος, για $\mu = 0$ εμφανίζεται διακλάδωση και για $\mu > 0$ ο ευσταθής κόμβος μετασχηματίζεται σε σάγμα ενώ παράλληλα εμφανίζονται δύο νέα σημεία ισορροπίας, τα οποία είναι ευσταθείς κόμβοι.



4. <u>Διακλάδωση Hopf</u>

Antiposows autής της κατηγορίας διακλαδώσεων είναι το σύστημα $\begin{cases} \dot{x} = -y + x(\mu - x^2 - y^2) \\ \dot{y} = x + y(\mu - x^2 - y^2) \end{cases}$. Διακλάδωση έχουμε στο $\mu = 0$ όπου και στο

γραμμικοποιημένο σύστημα υπάρχει κέντρο (δηλ. $\operatorname{Re}(\lambda_i) = 0$ και $\operatorname{Im}(\lambda_i) \neq 0$). Για $\mu < 0$ υπάρχει μια ευσταθής εστία, για $\mu > 0$ η εστία είναι ασταθής, ενώ κατά τη μετάβαση από την ευσταθή εστία στην ασταθή μέσω κέντρου, εμφανίζεται οριακός κύκλος. Παρακάτω φαίνεται το φασικό πορτρέτο για τις δύο περιπτώσεις $\mu < 0$ και $\mu > 0$.



Διάγραμμα ΙV

Στο παρακάτω διάγραμμα είναι εμφανής ο οριακός κύκλος που δημιουργείται καθώς οι τροχιές συγκλίνουν σε αυτόν.



Διάγραμμα V

Κεφάλαιο 3 Μη γραμμικά συστήματα και χάος

3.1 Βασικές έννοιες

Απεικόνιση Poincare: Ένα βασικό εργαλείο μελέτης των μη γραμμικών συστημάτων είναι η απεικόνιση Poincare. Αν το σύστημα είναι δύο διαστάσεων επιλέγοντας κάποιες αρχικές συνθήκες, είναι εύκολο να αναπαρασταθούν οι τροχιές σε ένα χώρο δύο διαστάσεων και επομένως υπάρχει εποπτική εικόνα της συμπεριφοράς του συστήματος (π.χ. σε ποιες περιοχές είναι περατωμένες οι τροχιές του συστήματος, αν υπάρχουν περιοδικές τροχιές κ.λ.π.). Αν όμως το σύστημα είναι περισσοτέρων διαστάσεων δεν είναι εύκολο με αυτόν τον τρόπο να αποκτηθεί εποπτική εικόνα της συμπεριφοράς του συστήματος. Για να μειωθεί η πολυπλοκότητα στην απεικόνιση του συστήματος χρησιμοποιείται η τομή Poincare. Η μέθοδος συνίσταται στο να παίρνονται τομές στο χώρο καταστάσεων, δηλαδή από μια τροχιά που εξελίσσεται, ζωγραφίζονται μόνο εκείνα τα σημεία της τροχιάς που αντιστοιχούν σε τομή αυτής με ένα προκαθορισμένο επίπεδο (π.γ. το επίπεδο που αντιστοιχεί σε σταθερή τιμή μιας μεταβλητής). Με αυτόν τον τρόπο υποβιβάζεται η διάσταση του συστήματος και ταυτόχρονα είναι εύκολο να ελεγχθούν κάποια ποιοτικά χαρακτηριστικά του (π.χ. μια περιοδική τροχιά εμφανίζεται σαν ένα σημείο πάνω στην τομή ή μια περιοδική τροχιά περιόδου δύο σαν δύο σημεία κ. λ . π .).

Μια από τις πιο θεμελιώδεις ιδιότητες των μη γραμμικών συστημάτων είναι το χάος. Στη συνέχεια θα δοθεί ο ορισμός του χάους και θα αναλυθούν κάποια από τα εργαλεία μελέτης του.

Ένα σύστημα έχει ένα χαοτικό αναλλοίωτο σύνολο [7] όταν το σύστημα σε αυτό το σύνολο έχει τις παρακάτω ιδιότητες:

- 1. Έχει πυκνό σύνολο περιοδικών τροχιών.
- 2. Είναι τοπολογικά μεταβατικό.
- 3. Έχει ευαίσθητη εξάρτηση από τις αρχικές συνθήκες.

(Σημ. Η τελευταία ιδιότητα αποδεικνύεται από τις άλλες δύο.)

- Πυκνό σύνολο περιοδικών τροχιών ουσιαστικά σημαίνει πως οσοδήποτε κοντά σε κάθε σημείο του αναλλοίωτου συνόλου υπάρχει περιοδική τροχιά.
- Τοπολογικά μεταβατικό είναι ένα σύνολο όταν με αρχικές συνθήκες από μια περιοχή του χώρου καταστάσεων κάτω από τη δράση της ροής του συστήματος είναι δυνατή η μετάβαση σε οποιαδήποτε άλλη περιοχή του χώρου καταστάσεων.
- Ευαίσθητη εξάρτηση από τις αρχικές συνθήκες υπάρχει όταν γειτονικές αρχικές συνθήκες μετά τη δράση του συστήματος, απομακρύνονται τόσο ώστε μετά από πεπερασμένο χρόνο να "φαίνονται" ασυσχέτιστες μεταξύ τους. Ουσιαστικά μετά από κάποιο χρόνο υπάρχει αδυναμία πρόβλεψης της εξέλιξης μιας τροχιάς αν δεν είναι ορισμένες οι αρχικές συνθήκες με άπειρη ακρίβεια.

(Σχόλιο. Παρά την αδυναμία πρόβλεψης της εξέλιξης του συστήματος το σύστημα είναι απόλυτα ντετερμινιστικό. Το χάος δηλαδή δεν σημαίνει άρση της αιτιότητας, γιατί αν οριστούν με άπειρη ακρίβεια οι αρχικές συνθήκες θα είναι απόλυτα γνωστή η εξέλιξη του συστήματος.)

Αποδεικνύεται πως το Bernouli shift έχει τις παραπάνω ιδιότητες και εμφανίζει χάος. Ένα άλλο σύστημα για να είναι χαοτικό αρκεί να έχει τις παραπάνω ιδιότητες ή αρκεί να είναι τοπολογικά συζυγές με το Bernouli shift.

Επειδή στις περισσότερες περιπτώσεις είναι πολύ δύσκολο να αποδειχθεί άμεσα πότε ένα σύστημα εμφανίζει χάος, η μελέτη του χάους και των χαρακτηριστικών του γίνεται με άλλες μεθόδους που περιλαμβάνουν κυρίως μεγέθη που παραμένον αναλλοίωτα κατά την εξέλιξη του συστήματος όπως είναι οι εκθέτες Lyapunov, η φασματική ανάλυση, η μορφοκλασματική διάσταση του παράξενου ή χαοτικού ελκυστή, η εντροπία και η πυκνότητα των τροχιών [3].

Ελκυστής σε ένα δυναμικό σύστημα είναι ο γεωμετρικός τόπος των σημείων που έλκει τροχιές από τη γειτονιά του. Ελκυστής μπορεί να είναι ένα σημείο, μια περιοδική τροχιά ή ένας παράξενος ελκυστής.

Παράξενος ελκυστής είναι μια τροχιά του συστήματος με μη ακέραια διάσταση πάνω στην οποία τείνουν να πέσουν άλλες τροχιές. Όταν αυτές οι τροχιές παρουσιάζουν χάος ο ελκύστης είναι χαοτικός. Ένας ελκυστής χαρακτηρίζεται από την ευκλείδεια διάσταση του ευκλείδειου χώρου που περιέχει τον ελκυστή στην περίπτωση του συστήματος που θα μελετήσουμε αυτή είναι 3.

Ένα χαρακτηριστικό των χαοτικών ελκυστών είναι η αυτό-ομοιότητα, δηλαδή η ύπαρξη όμοιων (χωρικά) δομών σε οποιαδήποτε κλίμακα και αν παρατηρήσουμε τον ελκυστή. Για να μελετηθεί η αυτό-ομοιότητα εισάγεται η έννοια της μορφοκλασματικής διάστασης. Η διάσταση αυτή είναι συνεπής με τη συνήθη έννοια της διάστασης, δηλαδή σύνολα σημείων έχουν διάσταση 0, γραμμές έχουν διάσταση 1 κλπ, και επιπλέον για αντικείμενα με την ιδιότητα της αυτό-ομοιότητας όπως οι παράξενοι ελκυστές, η μορφοκλασματική διάσταση είναι μη ακέραια. Μέτρα της μορφοκλασματικής διάστασης είναι η διάσταση συσχέτισης v (correlation) και η διάσταση χωρητικότητας (Box Counting).

Διάσταση χωρητικότητας: Η διάσταση χωρητικότητας είναι ένα μέτρο του πόσο "εξαπλωμένο" είναι ένα σύνολο σημείων μέσα στον ευκλείδειο χώρο στον οποίο είναι βυθισμένο. Ο υπολογισμός της διάστασης χωρητικότητας γίνεται αν χωριστεί ο χώρος σε κουτιά πλευράς l (τετράγωνα αν η διάσταση εμβύθισης είναι 2) και αθροιστεί ο αριθμός των κουτιών που περιέχουν έστω και ένα τμήμα του συνόλου[8]. Όταν το μέγεθος των κουτιών τείνει στο 0 τότε η διάσταση χωρητικότητας θα είναι $D_c = \lim_{l/L \to 0} (\frac{logN(l/L)}{log(L/l)})$, όπου L είναι το μήκος της

πλευράς του τετραγώνου και N(l/L) είναι τα κουτιά που περιέχουν έστω και ένα σημείο του συνόλου.

Διάσταση συσχέτισης: Έχουμε N σημεία ενός ελκυστή [3] (πρέπει το N να είναι μεγάλο) και ορίζουμε την πιθανότητα $P(||x_i - x_j|| < r)$ η οποία είναι η απόσταση δύο σημείων του ελκυστή να είναι μικρότερη από κάποια απόσταση r, όπού ||x|| είναι το μήκος κάποιου διανύσματος x σε κάποια μετρική π.χ. η ευκλείδεια. Αν μ_i είναι ο αριθμός των σημείων που βρίσκονται μέσα σε σφαίρα με ακτίνα r και κέντρο x_i , τότε η μέση τιμή ως προς όλα τα x_i προσεγγίζει την παραπάνω πιθανότητα. Ο νόμος κλιμάκωσης είναι $\langle \mu_i \rangle \sim r^{\nu}$ όταν $r \to 0$, δηλαδή για μικρές ακτίνες r η πιθανότητα η

κάποια δύναμη v της απόστασης, (v : σταθερό) η οποία εκφράζει το βαθμό αυτόομοιότητας και ονομάζεται διάσταση συσχέτισης. Αν ο ελκυστής είναι μια γραμμή ή ένα επίπεδο, τότε οι τιμές της διάστασης συσχέτισης είναι v=1ή v=2 ενώ αν είναι παράξενος η διάσταση συσχέτισης είναι μη ακέραια. Ο εκθέτης v υπολογίζεται από

τον τύπο $v = \frac{d \log C(r)}{d \log r}$, όπου το C(r) είναι το άθροισμα συσχέτισης

$$C(r) = \frac{2}{N(N-1)} \sum_{i=1}^{N} \sum_{j=i+1}^{N} \Theta(r - ||x_i - x_j||), \quad \text{kal} \quad \Theta(x) \quad \eta \quad \text{suvapthermal} \quad \text{Heaviside}$$
$$\Theta(x) = \begin{cases} 0 & , \alpha v \quad x < 0 \\ 1 & , \alpha v \quad x \ge 0 \end{cases}.$$

Όπως προαναφέρθηκε βασική ιδιότητα του χάους είναι η ευαίσθητη εξάρτηση από τις αρχικές συνθήκες. Αυτό σημαίνει πως δύο γειτονικές τροχιές απομακρύνονται μεταξύ τους πολύ γρήγορα και μάλιστα εκθετικά. Ο έλεγχος για την ταχύτητα απόκλισης γίνεται μέσω των εκθετών Lyapunov.

3.2 Εκθέτες Lyapunov

Av έχουμε ένα δυναμικό σύστημα $\dot{x} = f(x), x \in \mathbb{R}^n$ τότε για κάθε σημείο μιας τροχιάς του συστήματος μπορεί να βρεθεί η εξίσωση μεταβολών του συστήματος πάνω στον εφαπτόμενο χώρο $\dot{\xi} = Df(x(t)) \cdot \xi$, $\xi \in \mathbb{R}^n$. Πάνω σε αυτόν τον εφαπτόμενο χώρο ένα τυχαίο διάνυσμα ξ_i (συνήθως μοναδιαίο) κάτω από τη δράση του συστήματος μεταβολών, οδηγείται στο ξ_{i+1} . Αν l_t είναι το μέτρο του καινούργιου διανύσματος δια το μέτρο του αρχικού $l_t = \frac{\xi_{i+1}}{\xi_i}$ για κάθε χρονική στιγμή, τότε για $t \to \infty$ ο εκθέτης Lyapunov προκύπτει από το λογάριθμο l_t για κάθε χρονική στιγμή δια το χρόνο που έχει εξελιχθεί η τροχιά $\lambda = \overline{\lim_{t\to\infty} \frac{1}{t}} \log(l_t)$ [14].

Οι εκθέτες Lyapunov είναι τόσοι όση είναι η διάσταση του συστήματος και αυτό γιατί προς κάθε διαφορετική (γραμμικά ανεξάρτητη διεύθυνση) υπάρχει διαφορετική ταχύτητα απόκλισης των τροχιών. Παρακάτω γίνεται αναφορά στον αριθμητικό υπολογισμό του μέγιστου εκθέτη Lyapunov.

Ο μέγιστος εκθέτης Lyapunov [3],[4] υπολογίζεται από το μέσο όρο της εκθετικής απόκλισης δύο γειτονικών τροχιών όταν αυτές εξελιχθούν για θεωρητικά άπειρο χρόνο. Επειδή οι αρχικά γειτονικές τροχιές (έστω x_i και x'_i)με την πάροδο του χρόνου παύουν να είναι γειτονικές πρέπει σε κάθε βήμα του υπολογισμού οι τροχιές να ξαναγίνονται γειτονικές, άρα σε κάθε βήμα ξαναορίζονται αρχικές συνθήκες με τέτοιο τρόπο ώστε οι καινούργιες τροχιές να είναι συνέχεια των προηγουμένων. Αυτό επιτυγχάνεται με το να οριστούν έτσι ώστε οι καινούργιες αρχικές συνθήκες να βρίσκονται στη διεύθυνση που ορίζεται από το ευθύγραμμο τμήμα μεταξύ των περάτων των τροχιών (x_{i+t} και x'_{i+t}), όπως φαίνεται στην εικόνα 4.





Αν η απόσταση των αρχικών συνθηκών είναι δ_{0i} και μετά από χρόνο t η απόσταση των τροχιών είναι δ_{ii} τότε ο μέγιστος εκθέτης Lyapunov είναι

$$\lambda = \frac{1}{Nt} \sum_{i=1}^{N} \log \frac{\delta_{ii}}{\delta_{0i}}$$

Αποδεικνύεται πως :

- αν λ>0 τότε η τροχιά είναι χαοτική
- αν λ<0 τότε η τροχιά είναι τακτική

3.3 Φάσμα Fourier

Η ανάλυση Fourier είναι ένα πολύ χρήσιμο εργαλείο μελέτης συναρτήσεων, χρονοσειρών, κυματομορφών ή τροχιών ενός δυναμικού συστήματος. Με την ανάλυση Fourier αναλύεται μια χρονοσειρά σε επαλληλία ημιτονοειδών συναρτήσεων. Ο ευθύς μετασχηματισμός μεταφέρει από το πεδίο του χρόνου στο πεδίο των συχνοτήτων και ο αντίστροφος από το πεδίο των συχνοτήτων στο πεδίο του χρόνου. Αν f(t) είναι μια συνάρτηση του χρόνου [4] τότε ο ευθύς μετασχηματισμός είναι $F(\omega) = \int_{-\infty}^{+\infty} f(t)e^{-i\omega t}d\omega$ (IA) όπου το $F(\omega)$ γενικά είναι μιγαδικός, και ο αντίστροφος μετασχηματισμός είναι $f(t) = \frac{1}{2\pi} \int_{-\infty}^{+\infty} F(\omega)e^{i\omega t}d\omega$ (IB). Η ποσότητα $P(\omega) = |F(\omega)|^2$ ονομάζεται φάσμα ισχύος και δίνει τη συνεισφορά κάθε συχνότητας στη συνάρτηση. Αν η f(t) είναι πραγματική τότε $P(\omega) = P(-\omega)$ και $F(-\omega) = F(\omega)^*$. Επειδή στα πραγματικά προβλήματα η συνάρτηση f(t) δεν υπάρχει σε κλειστή μορφή παρά μόνο οι τιμές της σε διακριτό χρόνο $f_n \equiv f(\Delta t \cdot n)$, n = 0, 1, 2, 3, ..., N - 1, αντί για τους μετασχηματισμούς (IA, IB) χρησιμοποιούνται οι διακριτοί μετασχηματισμοί Fourier $F_k = \sum_{n=0}^{N-1} f_n e^{i\frac{2\pi}{N}kn}$ και $f_n = \frac{1}{N} \sum_{k=0}^{N-1} F_k e^{i\frac{2\pi}{N}kn}$.

Αν μια τροχιά είναι περιοδική τότε στο φάσμα ισχύος θα εμφανίζεται μια θεμελιώδης συχνότητα f_0 και ένας πεπερασμένος αριθμός συχνοτήτων ανάλογες με την θεμελιώδη $f_k \sim k \cdot f_0$. Αν μια τροχιά είναι ημιπεριοδική τότε θα έχουμε δύο θεμελιώδεις (ή περισσότερες) συχνότητες f_1 , f_2 και στο φάσμα θα εμφανίζονται κορυφές σε θέσεις που αποτελούν γραμμικό συνδυασμό δύο ή περισσοτέρων θεμελιωδών συχνοτήτων $f = \kappa \cdot f_1 + \lambda \cdot f_2$, $\kappa, \lambda \in \mathbb{Z}$. Τέλος αν μια τροχιά είναι συνεχές.

Κεφάλαιο 4 Εισαγωγή στη δυναμική πληθυσμών

Πληθυσμός είναι το σύνολο των ατόμων ενός είδους που διαβιούν σε ένα χώρο για ορισμένο χρονικό διάστημα [5], [6], [9], [10], [12].

Δυναμική πληθυσμών είναι ο κλάδος της οικολογίας που μελετά τις μεταβολές των πληθυσμών στο χρόνο.

Στη δυναμική πληθυσμών προσομοιώνουμε την εξέλιξη των πληθυσμών με μαθηματικά μοντέλα που σκοπό έχουν να μιμηθούν τη συμπεριφορά των βιολογικών συστημάτων. Υπάρχουν δύο είδη μαθηματικών μοντέλων για την περιγραφή ενός βιολογικού συστήματος:

- Συστήματα διαφορικών εξισώσεων όταν οι μεταβολές είναι συνεχείς.
- Συστήματα εξισώσεων διαφορών όταν το φαινόμενο μεταβάλλεται ασυνεχώς.

Παρότι στη δυναμική πληθυσμών οι μεταβολές είναι ασυνεχείς (γεννιέται και πεθαίνει ακέραιος αριθμός ατόμων), εφόσον οι πληθυσμοί και ο ρυθμός των γεννήσεων και των θανάτων είναι πολύ μεγάλοι θα θεωρήσουμε συνεχείς μεταβολές. Θα ασχοληθούμε λοιπόν με συστήματα διαφορικών εξισώσεων.

Οι διαφορικές εξισώσεις είναι πρώτης τάξης της μορφής
$$\frac{dx_i}{dt} = f(x_i)$$
, όπου x_i είναι ο

πληθυσμός ενός είδους και f είναι η συνάρτηση μέσω της οποίας μεταβάλλεται ο πληθυσμός του είδους. Στην f ενσωματώνονται οι φυσικές ιδιότητες του είδους, οι αλληλεπιδράσεις του με το περιβάλλον και με άλλα είδη που διαβιούν στον ίδιο χώρο.

Στη συνέχεια θα δούμε μερικά πληθυσμιακά μοντέλα ξεκινώντας από τα πιο απλά, στα οποία δεν υπάρχουν περιορισμοί και θα συνεχίσουμε με μοντέλα ενδοειδικού και διαειδικού ανταγωνισμού.

Ι. Μοντέλο γεννήσεων- θανάτων

Σε αυτό το μοντέλο ο ρυθμός μεταβολής του πληθυσμού δεν έχει φυσικούς περιορισμούς (π.χ. έλλειψη αποθεμάτων τροφής). Συνεπώς αν στη μονάδα του χρόνου, από κάθε άτομο του πληθυσμού γεννιούνται b απόγονοι και για κάθε άτομο του πληθυσμού γεννιούνται b απόγονοι και για κάθε άτομο του πληθυσμού του πληθυσμού του πληθυσμού συμβαίνουν d θάνατοι, τότε ο ρυθμός μεταβολής του πληθυσμού θα δίνεται από τον τύπο

$$\frac{dx}{dt} = bx - dx \qquad b > 0 \text{ kat } d > 0 \qquad (2.1a)$$

όπου bx οι συνολικές γεννήσεις (είσοδος στον πληθυσμό) και dx οι συνολικοί θάνατοι (απόσυρση από τον πληθυσμό). Η εξίσωση (2.1α) μπορεί να γραφτεί και στη μορερή $\frac{dx}{dx} = (b - d)x$ και αν τεθεί x = b - d τελικά ποριγίπτει

μορφή $\frac{dx}{dt} = (b-d)x$ και αν τεθεί r = b-d τελικά προκύπτει

$$\frac{dx}{dt} = rx \tag{2.1}$$

Η λύση της (2.1) είναι η $x = x_0 e^{rt}$. Αν b > d ο πληθυσμός συνεχώς αυξάνεται, ενώ αν b < d ο πληθυσμός θα εξαφανισθεί.

Παρότι το μοντέλο δεν είναι ρεαλιστικό είναι το πρώτο βήμα από το οποίο προκύπτουν πολλά άλλα πολύ πιο σύνθετα.

ΙΙ. Μοντέλο ενδοειδικού ανταγωνισμού

Σε αυτό το μοντέλο υπάρχουν περιορισμένα αποθέματα τροφής, επομένως αύξηση του πληθυσμού οδηγεί σε μείωση της πιθανότητας επιβίωσης. Κατά συνέπεια η σταθερά d πρέπει να εξαρτάται από τον υπάρχοντα πληθυσμό x και μάλιστα γραμμικά, δηλαδή όσο αυξάνει ο πληθυσμός αυξάνει το d επομένως d = ax. Άρα το μοντέλο γίνεται

$$\frac{dx}{dt} = bx - ax \cdot x \qquad \acute{\eta} \qquad \frac{dx}{dt} = x(b - ax) \qquad (2.2)$$

ΙΙΙ. Μοντέλο διαειδικού ανταγωνισμού

Σε αυτό το μοντέλο τα περιορισμένα αποθέματα τροφής τα μοιράζονται δύο είδη. Επομένως η πιθανότητα επιβίωσης μειώνεται και εξαιτίας της αύξησης του πληθυσμού του δεύτερου είδους, ισχύει $d = ax_1 + cx_2$. Οπότε το μοντέλο θα είναι

$$\left(\frac{dx_1}{dt} = b_1 x_1 - a_1 x_1 x_1 - c_1 x_1 x_2\right)$$
(2.3*a*)

$$\begin{cases} \frac{dx_2}{dt} = b_2 x_2 - a_2 x_2 x_2 - c_2 x_2 x_1 \\ (2.3\beta) \end{cases}$$

ΙV. Μοντέλο θηρευτή – θηράματος

Εκτός του έμμεσου ανταγωνισμού (ταυτόχρονη διεκδίκηση κοινών αποθεμάτων τροφής) υπάρχει και ο άμεσος ανταγωνισμός. Αυτό συμβαίνει όταν το ένα είδος αποτελεί τροφή για το άλλο.

Έστω x ο θηρευτής και y το θήραμα. Για την κατασκευή του μοντέλου υποθέτουμε πως απουσιάζει ο μηχανισμός θήρευσης και ο μηχανισμός θανάτου για το θήραμα, τότε ο πληθυσμός του θηράματος θα ελεγχόταν από την εξίσωση $\frac{dy}{dt} = ay$ και ο πληθυσμός του θηρευτή (μόνη πηγή τροφής είναι το θήραμα) θα δινόταν από την $\frac{dx}{dt} = -bx$, δηλαδή θα αυξανόταν ο πληθυσμός του μη θηρευόμενου είδους και θα μειωνόταν ο πληθυσμός του μη θηρεύοντος.

Αν εισαχθεί στο σύστημα και η θήρευση τότε ο πληθυσμός του θηράματος θα μειώνεται ανάλογα με τον πληθυσμό του θηρευτή και ο πληθυσμός του θηρευτή θα αυξάνεται ανάλογα με τον πληθυσμό του θηράματος, επομένως το μοντέλο θα γίνει

$$\begin{cases} \frac{dy}{dt} = ay - pxy\\ \frac{dx}{dt} = -bx + qxy \end{cases}$$
(2.4)

Αυτές οι εξισώσεις είναι γνωστές ως εξισώσεις Loteka-Volterra.

(Σημ. Σε όλα τα προηγούμενα οι τιμές των παραμέτρων εκφράζουν την ικανότητα του είδους για επιβίωση και τον τρόπο αλληλεξάρτησής του από το περιβάλλον.)

Β ΜΕΡΟΣ : Συστήματα Lotcka - Volterra

Στην παρούσα εργασία θα μελετήσουμε συστήματα της μορφής $\partial_t N_i = N_i \sum_{j=1}^3 a_{ij} (1 - N_j)$ τα οποία ανήκουν στην κατηγορία Volterra [15],[10],[12]. Επιλέγοντας τιμές για τις παραμέτρους του πίνακα a_{ij} και διατηρώντας ως παράμετρο

μία από αυτές παίρνουμε τον πίνακα $(a_{ij})_{\mu} = \begin{pmatrix} 0.5 & 0.5 & 0.1 \\ -0.5 & -0.1 & 0.1 \\ \mu & 0.1 & 0.1 \end{pmatrix}$. Επομένως το

δυναμικό σύστημα που έχουμε να μελετήσουμε είναι το :

$$\begin{cases} \frac{\partial x}{\partial t} = x(0.5(1-x) + 0.5(1-y) + 0.1(1-z)) \\ \frac{\partial y}{\partial t} = y(-0.5(1-x) - 0.1(1-y) + 0.1(1-z)) \\ \frac{\partial z}{\partial t} = z(\mu(1-x) + 0.1(1-y) + 0.1(1-z)) \end{cases}$$
(2.5)

Η φυσική σημασία του συστήματος φαίνεται όταν το γράψουμε στη μορφή :

$$\begin{cases} \frac{dx}{dt} = 1.1x - 0.5xx - 0.5xy - 0.1xz \\ \frac{dy}{dt} = -0.5y + 0.5yx + 0.1yy - 0.1yz \\ \frac{dz}{dt} = (\mu + 0.2)z - \mu zx - 0.1yz - 0.1zz \end{cases}$$
(2.6)

Από τη μορφή του συστήματος (2.6) διαπιστώνουμε πως το πρώτο είδος (x) υπόκειται στους περιορισμούς του ενδοειδικού, ενώ ταυτόχρονα αποτελεί θήραμα για το δεύτερο είδος (y). Το είδος y εκτός του ότι θηρεύει το είδος x και ότι η αλληλεπίδρασή του με μέλη του είδους του αυξάνουν τις πιθανότητες επιβίωσης (π.χ. αγελαίο είδος) υπόκειται και σε διαειδικό ανταγωνισμό με το τρίτο είδος (z). Ενώ το z για $\mu < 0$ είναι θηρευτής του x και για $\mu > 0$ διαειδικός ανταγωνιστής του (διεκδίκηση κοινών αποθεμάτων τροφής).

1. Σημεία Ισορροπίας -Ιδιοτιμές -Διακλαδώσεις

Στο σύστημα (2.5) υπάρχουν τα εξής σημεία ισορροπίας A1(0,0,0), B2 (0,5,0),
Γ3(0.7,1.5,0), Δ4(2.2,0,0), E5 (
$$\frac{0.09-0.1\mu}{0.05-0.1\mu}$$
, 0, $\frac{0.-0.6\mu}{0.05-0.1\mu}$),
Z6($0.7 + \frac{0.0015+0.009\mu}{0.005+0.03\mu}$, $1.5 - \frac{0.0025+0.015\mu}{0.005+0.03\mu}$, $-\frac{0.005+0.03\mu}{0.005+0.03\mu}$), H7(0,3.5+5μ,-
1.5+5μ) και Θ8 (0,0,2+10μ).

1. Το γραμμικοποιημένο σύστημα για το σημείο ισορροπίας (0,0,0) έχει και τις τρεις ιδιοτιμές πραγματικές

$$\begin{cases} \lambda_1 = 1.1 \\ \lambda_2 = -0.5 \\ \lambda_3 = 0.2 + \mu \end{cases}$$

Το σημείο είναι σάγμα. Για $\mu < -0.2$ έχει ένα μονοδιάστατο ευσταθή υποχώρο που αντιστοιχεί στην αρνητική ιδιοτιμή και ένα δισδιάστατο ασταθή που αντιστοιχεί στις θετικές ιδιοτιμές. Για την τιμή $\mu = -0.2$ το σημείο είναι μη υπερβολικό, και για $\mu > 0.2$ ο ευσταθής υποχώρος γίνεται δισδιάστατος και ο ασταθής μονοδιάστατος.

Στην τιμή $\mu = -0.2$ παρατηρείται η διακλάδωση (I) του (0,0,0) με το (0,0, 2+10μ).



(Σημ. Στα διαγράμματα των ιδιοτιμών το κόκκινο χρώμα αντιστοιχεί στις πραγματικές ιδιοτιμές, το μπλε στο πραγματικό μέρος των μιγαδικών ιδιοτιμών και το πράσινο στο φανταστικό μέρος. Στο διάγραμμα της ευστάθειας του σημείου η τιμή -1 αντιστοιχεί σε ασταθές σημείο και η τιμή +1 σε ευσταθές.)

 Το γραμμικοποιημένο σύστημα για το σημείο ισορροπίας (0, 5, 0) είναι σάγμα με τρεις πραγματικές ιδιοτιμές

$$\begin{cases} \lambda_1 = -1.4 \\ \lambda_2 = 0.5 \\ \lambda_3 = -0.3 + \mu \end{cases}$$

Για $\mu < 0.3$ το σημείο έχει ένα δισδιάστατο ευσταθή υποχώρο που αντιστοιχεί στις αρνητικές ιδιοτιμές και ένα μονοδιάστατο ασταθή που αντιστοιχεί στην θετική ιδιοτιμή. Για $\mu = 0.3$ το σημείο είναι μη υπερβολικό, ενώ για $\mu > 0.3$ ο ευσταθής υποχώρος γίνεται μονοδιάστατος και ο ασταθής δισδιάστατος. Στην τιμή $\mu = 0.3$ παρατηρείται η διακλάδωση (III) του (0,5,0) με το (0, 3.5+5μ, -1.5+5μ)



 Το γραμμικοποιημένο σύστημα για το σημείο ισορροπίας (0.7, 1.5, 0) έχει δύο συζυγείς μιγαδικές ιδιοτιμές και μια πραγματική.

$$\begin{cases} \lambda_1 = 0.05 + 0.3\,\mu\\ \lambda_2 = -0.1 - i0.447214\\ \lambda_3 = -0.1 + i0.447214 \end{cases}$$

Για $\mu < -1/6$ το σημείο έχει ένα ευσταθές επίπεδο που αντιστοιχεί στις μιγαδικές ιδιοτιμές και ένα ευσταθή μονοδιάστατο υποχώρο που αντιστοιχεί στην πραγματική ιδιοτιμή. Στην τιμή $\mu = -1/6$ το σημείο είναι μη υπερβολικό και διακλαδώνεται στη διακλάδωση (VII) (βλέπε παρακάτω). Για $\mu > -1/6$ ο ευσταθής μονοδιάστατος υποχώρος που αντιστοιχεί στην πραγματική ιδιοτιμή γίνεται ασταθής, οπότε το σημείο γίνεται σάγμα– εστία.



4. Το γραμμικοποιημένο σύστημα για το σημείο ισορροπίας (2.2,0,0) έχει πραγματικές ιδιοτιμές

$$\begin{cases} \lambda_1 = 0.6 \\ \lambda_2 = 0.5(-0.9 - 1.2\mu - 1.2\sqrt{1.17361 - 2.16667\mu + \mu^2}) \\ \lambda_3 = 0.5(-0.9 - 1.2\mu + 1.2\sqrt{1.17361 - 2.16667\mu + \mu^2}) \end{cases}$$

Για $\mu < 1/6$ το σημείο είναι σάγμα με δισδιάστατο ασταθή υποχώρο που αντιστοιχεί στις θετικές ιδιοτιμές και μονοδιάστατο ευσταθή που αντιστοιχεί στην αρνητική ιδιοτιμή. Όταν $\mu = 1/6$ το σημείο είναι μη υπερβολικό, ενώ για $\mu > 1/6$ ο ασταθής υποχώρος γίνεται μονοδιάστατος και ο ευσταθής δισδιάστατος. Στην τιμή $\mu = 1/6$ παρατηρείται η διακλάδωση (II) του (2.2,0,0) με το ($\frac{0.09 - 0.1\mu}{0.05 - 0.1\mu}$, 0, $\frac{0.1 - 0.6\mu}{0.05 - 0.1\mu}$).



{ SHMEIO ISOPPONIAS (2.2,0,0) }





5. Το γραμμικοποιημένο σύστημα για το σημείο ισορροπίας $(\frac{0.09-0.1\mu}{0.05-0.1\mu}, 0,$

$$\begin{split} \frac{0.-0.6\mu}{0.05-0.1\mu}) & \text{éxel idiotiµés}:\\ \begin{cases} \lambda_1 &= \frac{-0.1-0.6\mu}{-0.5+\mu} \\ \lambda_2 &= \frac{0.275-0.55\mu}{-0.5+\mu} + \frac{(0.25-0.5\mu)\sqrt{\frac{0.1225+0.43\mu-2.55\mu^2+2.4\mu^3}{0.25-\mu+\mu^2}}}{-0.5+\mu} \\ \lambda_3 &= \frac{0.275-0.55\mu}{-0.5+\mu} - \frac{(0.25-0.5\mu)\sqrt{\frac{0.1225+0.43\mu-2.55\mu^2+2.4\mu^3}{0.25-\mu+\mu^2}}}{-0.5+\mu} \end{split}$$

Για την τιμή $\mu = -1/6$ και $\mu = 0.9$ το σημείο είναι μη υπερβολικό. Για $\mu = -1/6$ παρατηρείται η διακλάδωση (VII), και για $\mu = 0.9$ παρατηρείται η διακλάδωση (V) όπου το $(\frac{0.09 - 0.1\mu}{0.05 - 0.1\mu}, 0, \frac{0.1 - 0.6\mu}{0.05 - 0.1\mu})$ διακλαδώνεται με το

- $(0, 0, 2+10\mu).$
- a. Για $\mu \in (-\infty, -1/6)$ το σημείο έχει ένα ευσταθές επίπεδο που αντιστοιχεί στις μιγαδικές ιδιοτιμές (πάνω στο επίπεδο παρατηρείται συμπεριφορά εστίας) και έναν ευσταθή υποχώρο που αντιστοιχεί στην πραγματική ιδιοτιμή, έτσι ώστε όλος ο χώρος είναι ευσταθής.
- b. Για μ∈ (-1/6,-0.144408) το σημείο έχει ένα ευσταθές επίπεδο που αντιστοιχεί στις μιγαδικές ιδιοτιμές (πάνω στο επίπεδο παρατηρείται συμπεριφορά εστίας) και έναν ασταθή υποχώρο που αντιστοιχεί στην θετική πραγματική ιδιοτιμή, άρα το σημείο είναι σάγμα-εστία.

- c. Στο $\mu = -0.144408$ το σημείο μετατρέπεται σε σάγμα (το $\mu = -0.144408$ δεν είναι σημείο διακλάδωσης) και παραμένει σάγμα έως την τιμή $\mu = 1/6$ με μονοδιάστατο ασταθή και δισδιάστατο ευσταθή υποχώρο.
- d. Για $\mu \in (1/6, 0.5)$ το σημείο είναι σάγμα με δισδιάστατο ασταθή και μονοδιάστατο ευσταθή υποχώρο.
- e. Για την τιμή $\mu = 0.5$ το σημείο τείνει στο άπειρο.
- f. Καθώς προσεγγίζεται η τιμή $\mu = 0.5$ από αριστερά (όλες οι ιδιοτιμές είναι πραγματικές) οι ιδιοτιμές λ_1 και λ_3 απειρίζονται θετικά και η ιδιοτιμή λ_2 αρνητικά και καθώς προσεγγίζεται η τιμή $\mu = 0.5$ από δεξιά η ιδιοτιμή λ_1 απειρίζεται αρνητικά (οι λ_2 και λ_3 πλέον είναι μιγαδικές).
- g. Για $\mu \in (0.5, 0.706908)$ το σημείο έχει ένα ευσταθές επίπεδο που αντιστοιχεί στις μιγαδικές ιδιοτιμές (πάνω στο επίπεδο παρατηρείται συμπεριφορά εστίας) και έναν ευσταθή υποχώρο που αντιστοιχεί στην αρνητική πραγματική ιδιοτιμή, έτσι ώστε όλος ο χώρος είναι ευσταθής.
- h. Στο $\mu = 0.706908$ το σημείο μετατρέπεται σε ευσταθή κόμβο (το $\mu = 0.706908$ δεν είναι σημείο διακλάδωσης) και παραμένει ευσταθής κόμβος έως την τιμή $\mu = 0.9$.
- i. Για $\mu \in (0.9, +\infty)$ το σημείο είναι σάγμα με μονοδιάστατο ασταθή και δισδιάστατο ευσταθή υποχώρο.



6. Το γραμμικοποιημένο σύστημα για το σημείο ισορροπίας

$$(0.7 + \frac{0.0015 + 0.009\mu}{0.005 + 0.03\mu}, 1.5 - \frac{0.0025 + 0.015\mu}{0.005 + 0.03\mu}, -\frac{0.005 + 0.03\mu}{0.005 + 0.03\mu})$$

που για $\mu \neq -1/6$ είναι το (1, 1, 1) έχει ιδιοτιμές που η μορφή τους απεικονίζεται στο Διάγραμμα XI



{ $\Sigma HMEIO I\Sigma OPPO\PiIA\Sigma (1,1,1)$ }

Για τις τιμές $\mu = 0.9545454$ και $\mu = -1/6$ το σημείο είναι μη υπερβολικό. Για $\mu = -1/6$ παρατηρείται η διακλάδωση (VII), και για $\mu = 0.9545454$ έχουμε διακλάδωση Hopf. Από τα διαγράμματα των ιδιοτιμών προκύπτει πως το σημείο είναι :

- a. Για $\mu \in (-\infty, -1/6)$ το σημείο έχει ένα ευσταθές επίπεδο που αντιστοιχεί στις μιγαδικές ιδιοτιμές (πάνω στο επίπεδο παρατηρείται συμπεριφορά εστίας) και έναν ασταθή υποχώρο που αντιστοιχεί στη θετική πραγματική ιδιοτιμή, άρα το σημείο είναι σάγμα-εστία.
- b. Για $\mu \in (-1/6, 0.954)$ το σημείο έχει ένα ευσταθές επίπεδο που αντιστοιχεί στις μιγαδικές ιδιοτιμές (πάνω στο επίπεδο παρατηρείται συμπεριφορά εστίας) και έναν ευσταθή υποχώρο που αντιστοιχεί στην αρνητική πραγματική ιδιοτιμή, έτσι ώστε όλος ο χώρος είναι ευσταθής.
- c. Για $\mu \in (0.954, +\infty)$ το σημείο έχει ένα ασταθές επίπεδο που αντιστοιχεί στις μιγαδικές ιδιοτιμές (πάνω στο επίπεδο παρατηρείται συμπεριφορά ασταθούς εστίας) και έναν ευσταθή υποχώρο που αντιστοιχεί στην πραγματική ιδιοτιμή, άρα το σημείο είναι σάγμα-εστία.
- Το γραμμικοποιημένο σύστημα για το σημείο ισορροπίας (0,3.5+5μ, -1.5+5μ) έχει ιδιοτιμές που δίνονται από τις σχέσεις

$$\begin{cases} \lambda_1 = 0.25 - 0.707107\sqrt{(-0.153553 + \mu)(0.553553 + \mu)} \\ \lambda_2 = 0.25 + 0.707107\sqrt{(-0.153553 + \mu)(0.553553 + \mu)} \\ \lambda_3 = -0.5 - 3\mu \end{cases}$$

Για τις τιμές $\mu = -1/6$, $\mu = -0.7$ και $\mu = 0.3$ το σημείο είναι μη υπερβολικό. Για $\mu = -1/6$ παρατηρείται η διακλάδωση (VII), για $\mu = -0.7$ τα σημεία (0,0,2+10μ) και (0, 3.5+5μ, -1.5+5μ) διακλαδώνονται στη διακλάδωση (IV) και για $\mu = 0.3$ τα σημεία (0, 3.5+5μ, -1.5+5μ) και (0, 5,0) διακλαδώνονται στην διακλάδωση (III).

Το σημείο είναι :

- a. σάγμα για $\mu \in (-\infty, -0.7)$ με μονοδιάστατο ευσταθή υποχώρο που αντιστοιχεί στην αρνητική πραγματική ιδιοτιμή και δισδιάστατο ασταθή υποχώρο που αντιστοιχεί στις θετικές πραγματικές ιδιοτιμές.
- b. ασταθής κόμβος για $\mu \in (-0.7, -0.55)$, όλες οι ιδιοτιμές είναι πραγματικές και θετικές.
- c. Για $\mu \in (-0.55, -1/6)$ το σημείο έχει ένα ασταθές επίπεδο που αντιστοιχεί στις μιγαδικές ιδιοτιμές (πάνω στο επίπεδο παρατηρείται συμπεριφορά ασταθούς εστίας) και έναν ασταθή υποχώρο που αντιστοιχεί στην πραγματική ιδιοτιμή, άρα όλος ο χώρος είναι ασταθής.
- d. Για $\mu \in (-1/6, 0.153)$ το σημείο έχει ένα ασταθές επίπεδο που αντιστοιχεί στις μιγαδικές ιδιοτιμές (πάνω στο επίπεδο παρατηρείται συμπεριφορά ασταθούς εστίας) και έναν ευσταθή υποχώρο που αντιστοιχεί στην αρνητική πραγματική ιδιοτιμή, άρα το σημείο είναι σάγμα-εστία.
- e. Το μ=0.153 δεν είναι σημείο διακλάδωσης.
- f. Για $\mu \in (0.153, 0.3)$ το σημείο είναι σάγμα με μονοδιάστατο ευσταθή υποχώρο που αντιστοιχεί στην αρνητική πραγματική ιδιοτιμή και έναν δισδιάστατο ασταθή υποχώρο που αντιστοιχεί στις θετικές πραγματικές ιδιοτιμές.
- *g*. Για $\mu \in (0.3, +\infty)$ το σημείο είναι σάγμα με δισδιάστατο ευσταθή υποχώρο και μονοδιάστατο ασταθή.

{ ΣΗΜΕΙΟ ΙΣΟΡΡΟΠΙΑΣ (0,3.5+5m,-1.5+5m) }



8. Το γραμμικοποιημένο σύστημα για το σημείο ισορροπίας (0,0,2+10μ) έχει ιδιοτιμές που δίνονται από τις σχέσεις

$$\begin{cases} \lambda_1 = -0.7 - \mu \\ \lambda_2 = 0.9 - \mu \\ \lambda_3 = -0.2 - \mu \end{cases}$$

Για τις τιμές $\mu = -0.7$, $\mu = -0.2$ και $\mu = 0.9$ το σημείο είναι μη υπερβολικό. Για $\mu = -0.7$ τα σημεία (0, 0, 2+10μ) και (0, 3.5+5μ, -1.5+5μ) διακλαδώνονται στη διακλάδωση (IV), για $\mu = -0.2$ παρατηρείται η διακλάδωση (I) του (0,0,0) με το (0,0, 2+10μ), και για $\mu = 0.9$ παρατηρείται η διακλάδωση (V), όπου το $(\frac{0.09-0.1\mu}{0.05-0.1\mu}, 0, \frac{0.1-0.6\mu}{0.05-0.1\mu})$ διακλαδώνεται με το (0, 0, 2+10μ).

- a. Fia $\mu \in (-\infty, -0.7)$ to shipe evaluated astaby kómbog .
- b. Για $\mu \in (-0.7, -0.2)$ το σημείο είναι σάγμα με μονοδιάστατο ευσταθή υποχώρο και δισδιάστατο ασταθή.
- c. Για $\mu \in (-0.2, 0.9)$ το σημείο είναι σάγμα με δισδιάστατο ευσταθή υποχώρο και μονοδιάστατο ασταθή.
- d. Fia $\mu \in (0.9, \infty)$ to shift einal eustabhic kómboc.

{ SHMEIO ISOPPONIAS ($0, 0, 2+10\mu$) }



Στη συνέχεια συγκεντρώνονται όλα τα αποτελέσματα στο διάγραμμα XIV (που παρήχθησαν με το πρόγραμμα voltisoropia_ola_22new.nb). Στον κατακόρυφο άξονα αντιστοιχεί ο αύξοντας αριθμός του σημείου και στον οριζόντιο η τιμή της παραμέτρου μ. Το κόκκινο σημαίνει ασταθές σημείο και το μπλε ευσταθές. Το μέγεθος αντιστοιχεί στο είδος του σημείου Σάγμα:το μικρό, "Εστία":το μεσαίο, Κόμβος:το μεγάλο. Όταν το σημείο είναι Σάγμα-Εστία αυτό απεικονίζεται με δύο χρώματα





Στο διάγραμμα φαίνεται πως τρία σημεία είναι συνεχώς σάγματα το A1(0,0,0) το B2 (0, 5, 0) και το Δ4(2.2, 0,0) τα οποία είναι συνεχώς ακίνητα. Επίσης παρατηρούμε ότι για $\mu > 0.954$ υπάρχει μόνο ένα ευσταθές σημείο ισορροπίας το (0, 0, 2+10μ).

2. Διακλαδώσεις

- Ι. Για $\mu = -0.2$ τα σημεία (0,0,0) και (0,0, 2+10μ) συναντιούνται και διακλαδώνονται. Για $\mu < -0.2$ το σημείο (0,0,0) είναι σάγμα με μονοδιάστατο ασταθή και δισδιάστατο ευσταθή υποχώρο, και το σημείο (0,0, 2+10μ) είναι σάγμα με δισδιάστατο ασταθή και μονοδιάστατο ευσταθή υποχώρο. Για $\mu > -0.2$ τα σημεία εξακολουθούν να είναι σάγματα, αλλά το (0,0,0) έχει δισδιάστατο ασταθή και μονοδιάστατο ευσταθή υποχώρους, και το σημείο (0,0,2+10μ) έχει μονοδιάστατο ασταθή και δισδιάστατο ευσταθή υποχώρο.
- II. Για $\mu = 1/6$ τα σημεία (2.2,0,0) και $(\frac{0.09 0.1\mu}{0.05 0.1\mu}, 0, \frac{0.1 0.6\mu}{0.05 0.1\mu})$ συναντιούνται και διακλαδώνονται. Για $\mu < 1/6$ το σημείο (2.2,0,0) είναι σάγμα με δισδιάστατο ασταθή και μονοδιάστατο ευσταθή υποχώρο, και το σημείο $(\frac{0.09 - 0.1\mu}{0.05 - 0.1\mu}, 0, \frac{0.1 - 0.6\mu}{0.05 - 0.1\mu})$ είναι σάγμα με μονοδιάστατο ασταθή και δισδιάστατο ευσταθή υποχώρο. Για $\mu > 1/6$ τα σημεία εξακολουθούν να είναι σάγματα, αλλά το (2.2,0,0) έχει μονοδιάστατο ασταθή και δισδιάστατο ευσταθή υποχώρο, και το σημείο $(\frac{0.09 - 0.1\mu}{0.05 - 0.1\mu})$
 - ,0 , $\frac{0.1 0.6\mu}{0.05 0.1\mu}$) έχει δισδιάστατο ασταθή και μονοδιάστατο ευσταθή υποχώρο.
- III. Για $\mu = 0.3$ τα σημεία (0, 3.5+5 μ , -1.5+5 μ) και (0, 5,0) συναντιούνται και διακλαδώνονται. Για $\mu < 0.3$ το σημείο (0, 3.5+5 μ , -1.5+5 μ) είναι σάγμα με δισδιάστατο ασταθή και μονοδιάστατο ευσταθή υποχώρο, και το σημείο (0, 5, 0) είναι σάγμα με μονοδιάστατο ασταθή και δισδιάστατο ευσταθή υποχώρο. Για $\mu > 0.3$ τα σημεία εξακολουθούν να είναι σάγματα, αλλά το (0, 3.5+5 μ , -1.5+5 μ) έχει μονοδιάστατο ασταθή και δισδιάστατο ευσταθή υποχώρο, και το σημείο (0, 5, 0) είναι σάγματα.
- IV. Για $\mu = -0.7$ τα σημεία (0, 0, 2+10μ) και (0, 3.5+5μ, -1.5+5μ) συναντιούνται και διακλαδώνονται. Για $\mu < -0.7$ το σημείο (0, 0, 2+10μ) είναι ασταθής κόμβος, το σημείο (0, 3.5+5μ, -1.5+5μ) είναι σάγμα με δισδιάστατο ασταθή και μονοδιάστατο ευσταθή υποχώρο. Για $\mu > -0.7$, το σημείο (0, 3.5+5μ, -1.5+5μ) παραμένει σάγμα και το (0, 0, 2+10μ) γίνεται σάγμα με δισδιάστατο ασταθή και μονοδιάστατο ευσταθή υποχώρο. Επομένως η διακλάδωση είναι υποκρίσιμη.

V. Fix $\mu = 0.9$ to shift (0, 0, 2+10 μ) kai to $(\frac{0.09 - 0.1\mu}{0.05 - 0.1\mu}, 0)$

 $,\frac{0.1-0.6\mu}{0.05-0.1\mu})$ συναντιούνται και διακλαδώνονται. Για $\mu < 0.9$ το σημείο $(\frac{0.09-0.1\mu}{0.05-0.1\mu})$, 0, $\frac{0.1-0.6\mu}{0.05-0.1\mu}$) είναι ευσταθής κόμβος και το σημείο $(0,0,2+10\mu)$ είναι σάγμα με μονοδιάστατο ασταθή και δισδιάστατο ευσταθή υποχώρο. Για $\mu > 0.9$ το σημείο $(\frac{0.09-0.1\mu}{0.05-0.1\mu})$, 0, $\frac{0.1-0.6\mu}{0.05-0.1\mu})$

γίνεται σάγμα με μονοδιάστατο ασταθή και δισδιάστατο ευσταθή υποχώρο και το σημείο (0, 0, 2+10μ) γίνεται ευσταθής κόμβος. Επομένως η διακλάδωση είναι υποκρίσιμη.

VI. Το σημείο $(1,1,1)^*$ γύρω από το $\mu=0.9545454$ είναι σάγμα-εστία, έχει έναν ασταθή υποχώρο, και ένα επίπεδο που αντιστοιχεί στις μιγαδικές ιδιοτιμές, πάνω στο οποίο το σημείο συμπεριφέρεται σαν ευσταθή εστία για $\mu<0.9545454$ και σαν ασταθή εστία για $\mu>0.9545454$ άρα σε αυτό το επίπεδο (επειδή αναφερόμαστε στο γραμμικοποιημένο σύστημα καθώς αλλάζει το μ το επίπεδο αλλάζει κλίση) περνάει από ευσταθή εστία σε ασταθή εστία μέσω κέντρου, ενώ παράλληλα δημιουργείται ένας ευσταθής οριακός κύκλος**. Επομένως έχουμε διακλάδωση Hopf.

(* στην πραγματικότητα το σημείο (1, 1, 1) είναι το

$$(0.7 + \frac{0.0015 + 0.009\mu}{0.005 + 0.03\mu}, 1.5 - \frac{0.0025 + 0.015\mu}{0.005 + 0.03\mu}, -\frac{0.005 + 0.03\mu}{0.005 + 0.03\mu})$$

που για μ ≠ -1/6 γίνεται το (1,1,1))

(** Ο οριακός κύκλος και η λεκάνη έλξης του υπολογίζονται παρακάτω.)

VII. Για την τιμή $\mu = -1/6$ τα σημεία ισορροπίας (0.7, 1.5, 0),

$$\left(\frac{0.09 - 0.1\mu}{0.05 - 0.1\mu} , 0, \frac{0.1 - 0.6\mu}{0.05 - 0.1\mu}\right) , \qquad (0, \quad 3.5 + 5\mu , \quad -1.5 + 5\mu) \quad \kappa\alpha\alpha ,$$
$$\left(0.7 + \frac{0.0015 + 0.009\mu}{0.005 + 0.03\mu} , \quad 1.5 - \frac{0.0025 + 0.015\mu}{0.005 + 0.03\mu} , \quad -\frac{0.005 + 0.03\mu}{0.005 + 0.03\mu} \right)$$

γίνονται μη υπερβολικά .

Για $\mu = -1/6$ τα τρία πρώτα γίνονται

$$\begin{cases} (0.7, 1.5, 0) \\ (1.6, 0, 3) \\ (0, 8/3, -4/3) \end{cases} (A_1)$$

ενώ το τέταρτο γίνεται (0.7+0.3z,1.5-0.5z,z) (A₂). Στο διάγραμμα XV φαίνεται η θέση των παραπάνω σημείων καθώς και η

ευθεία A_2 από την οποία διέρχονται για $\mu = -1/6$.



Διάγραμμα ΧV

Ουσιαστικά το τέταρτο σημείο για $\mu = -1/6$ μετατράπηκε σε μια απειρία σημείων, τα οποία διέρχονται από την ευθεία (A₂), οι συντεταγμένες των σημείων (A₁) ικανοποιούν την (A₂) πράγμα που σημαίνει πως αυτά τα τέσσερα σημεία διακλαδώνονται "με αυτόν τον ιδιότυπο τρόπο". Έτσι για $\mu < -1/6$ το σημείο (0.7,1.5,0) πάνω στο επίπεδο, που αντιστοιχούν οι μιγαδικές ιδιοτιμές, εμφανίζεται με συμπεριφορά ευσταθούς εστίας και στην διεύθυνση που αντιστοιχεί στην πραγματική ιδιοτιμή έχει έναν ευσταθή υπόχωρο, το σημείο (1.6,0,3) είναι σάγμα-εστία, το σημείο (0,8/3,-4/3) πάνω στο επίπεδο που αντιστοιχούν οι μιγαδικές ιδιοτιμές εμφανίζεται με συμπεριφορά ασταθούς εστίας και στη διεύθυνση που αντιστοιχεί στην πραγματική ιδιοτιμή έχει έναν ασταθή υπόχωρο και το σημείο (0.7+0.3z,1.5-0.5z,z) είναι σάγμα-εστία. Για $\mu > -1/6$ σε όλα τα σημεία αλλάζει η ευστάθεια του μονοδιάστατου υποχώρου από ευσταθή σε ασταθή (ή από ασταθή σε ευσταθή), ενώ παραμένει ανέπαφο το επίπεδο που αντιστοιχεί στις μιγαδικές ιδιοτιμές.

3. Δυναμική Μελέτη – Εμφάνιση Χαοτικής συμπεριφοράς

Θεώρημα του Silnikov [13],[15]

Στη συνέχεια θα διατυπωθεί το θεώρημα του Silnikov και θα διαπιστώσουμε αριθμητικά την εμφάνιση του χάους.

Το θεώρημα του Silnikov προϋποθέτει ότι όταν σε ένα τρισδιάστατο σύστημα διαφορικών εξισώσεων έχουμε ένα σημείο ισορροπίας που είναι Σάγμα-εστία (Saddle-focus) και υπάρχει μια ομοκλινική τροχιά Γ_0 τότε αυτή περιέχει ένα αριθμήσιμο πλήθος περιοδικών τροχιών σαγματικού τύπου. Η ύπαρξη απείρων περιοδικών τροχιών είναι ένα σημαντικό βήμα για την εμφάνιση χάους σε ένα δυναμικό σύστημα.

Ένα σενάριο με βάση το οποίο μπορεί να εμφανισθεί αυτή η ομοκλινική τροχιά, όταν έχουμε μια μονοπαραμετρική οικογένεια συστήματος διαφορικών εξισώσεων είναι:

- a. Για μια τιμή της παραμέτρου μ_0 να υπάρχει ένα Σάγμα Εστία.
- b. Όταν η τιμή της παραμέτρου γίνει μεγαλύτερη από μια τιμή $\mu_{\rm H} > \mu_0$ δημιουργείται μια διακλάδωση Hopf. Σε αυτήν την περίπτωση η
ασταθής πολλαπλότητα του σημείου (Σάγμα – Εστία) μπορεί να τείνει στον οριακό κύκλο.

c. Υπάρχει περίπτωση καθώς η τιμή του μ αυξάνει ολοένα και περισσότερο η ασταθής πολλαπλότητα του Σάγματος-Εστία να ξεφύγει από την περιοχή έλξης του οριακού κύκλου και να αναγκασθεί να αναδιπλωθεί. Τότε δημιουργείται η ομοκλινική τροχιά, που είναι η προϋπόθεση του θεωρήματος του Silnikov, και έχουμε την εμφάνιση αριθμήσιμου πλήθους περιοδικών τροχιών, που είναι σαγματικού τύπου.

Παρακάτω θα μελετήσουμε ως προς τα χαρακτηριστικά τους κάποιες αντιπροσωπευτικές καταστάσεις του συστήματος. Τα προγράμματα που χρησιμοποιήθηκαν για την επίλυση του συστήματος είναι το πρόγραμμα volterra, για τους εκθέτες Lyapunov τα προγράμματα liap_volter2gramikos και liap_volter, για τις τομές Poincare το poincare_volt.

Για τιμές της παραμέτρου $\mu < 0.96$, δηλαδή πριν την διακλάδωση Hopf, όλες οι τροχιές είναι τακτικές, δηλαδή δεν έχει εμφανιστεί χάος με αποτέλεσμα για οποιεσδήποτε αρχικές συνθήκες η εξέλιξη του συστήματος να οδηγεί είτε σε κάποιο από τα σημεία ισορροπίας (κυρίως τα ευσταθή) είτε σε απειρισμό κάποιων από τους πληθυσμούς. Το τελευταίο αποτέλεσμα προφανώς είναι λανθασμένο και οφείλεται στο ότι το μοντέλο είναι πολύ απλό και δεν έχουν συμπεριληφθεί σε αυτό όλοι οι φυσικοί περιορισμοί.

Το ενδιαφέρον ως προς τη μελέτη του συστήματος ξεκινάει κυρίως για τιμές της παραμέτρου μ μεγαλύτερες από $\mu \cong 0.96$ και αυτό γιατί στο $\mu = 0.9545454$ γίνεται διακλάδωση Hopf και εμφανίζεται οριακός κύκλος (σχήμα1), ο οποίος διευρύνει το μέγεθός του έως την τιμή $\mu \cong 1.3$. Στη συνέχεια μέσω του φαινομένου διπλασιασμού περιόδου σχήμα 2 καταλήγει σε παράξενο ελκυστή κοντά στην τιμή $\mu \cong 1.4$.



Σχήμα 1

(Σημ. Τα σχήματα γίνονται με χρήση του Matlab.)

Στο σχήμα 1 με τα διαφορετικά χρώματα αποτυπώνεται η αύξηση του εύρους του οριακού κύκλου ενώ στο σχήμα 2 αναπαρίσταται το φαινόμενο διπλασιασμού περιόδου που πολύ σύντομα οδηγεί το σύστημα στη δημιουργία παράξενου ελκυστή.



Σχήμα 2

Για μ=1.20 με αρχικές συνθήκες (0.558437,1.28227,1) (οι οποίες αναζητήθηκαν με το πρόγραμμα periodicOrbit) έχουμε τον οριακό κύκλο που απεικονίζεται στο Σχήμα 3.



Για αυτήν την τροχιά υπολογίζουμε την εξέλιξη του εκθέτη Lyapunov και την απεικονίζουμε σε λογαριθμικό διάγραμμα, ο χρόνος που εξελίχθηκε η τροχιά είναι t=5000000 χρονικές μονάδες και η τιμή του εκθέτη Lyapunov είναι λ=0.0. Επίσης

στο σχήμα 5 φαίνεται το φάσμα ισχύος. Από τα δύο σχήματα αποδεικνύεται πως η τροχιά είναι τακτική .



Σχήμα 4



Σχήμα 5

(Σημ. Τα φάσματα παράγονται στο Origin με τον αλγόριθμο FFT για 2^{17} ή 2^{18} σημεία.)

Στο φάσμα ισχύος οι κορυφές είναι στις θέσεις f_0 =0.05226, f_1 =0.10448, f_2 =0.15671, f_3 =0.20893, f_4 =0.26113, f_5 =0.31342 και f_6 =0.36564. Όπως διαπιστώνεται τα μέγιστα

βρίσκονται σε συχνότητες που είναι ακέραια πολλαπλάσια της βασικής συχνότητας, επομένως η τροχιά είναι περιοδική με περίοδο $T = \frac{1}{f_0} = 19,135094$.

Στη συνέχεια στο σχήμα 6 φαίνεται η λεκάνη έλξης του οριακού κύκλου, η οποία δημιουργήθηκε με το πρόγραμμα perioxi_elxis. Στο σχήμα διαπιστώνουμε πως οι προβολές της περιοχής έλξης στα επίπεδα x-y και x-z είναι περατωμένες, ενώ με την αύξηση του z φαίνεται να διευρύνεται αυτή η περιοχή. Υπολογιστικοί περιορισμοί δεν επιτρέπουν να διαπιστωθεί αν με την αύξηση του z η λεκάνη έλξης είναι περατωμένη και κατά τον άξονα z (να σημειωθεί πως για μ=1.20 το σύστημα έχει μόνο ένα ευσταθές σημείο ισορροπίας, το οποίο βρίσκεται μακριά από τον οριακό κύκλο στην θέση (0,0,2+10μ)).



Σχήμα 6

Για μ=1.32 έχουμε περιοδική τροχιά περιόδου 2 και για μ=1.38 έχουμε περιοδική τροχιά περιόδου 4.



Σχήμα 7



Σχήμα 8

Στο παρακάτω σχήμα (σχήμα 9) δείχνουμε την εξέλιξη του εκθέτη Lyapunov της τροχιάς περιόδου 4. Ο χρόνος που αφήσαμε την τροχιά να εξελιχθεί είναι t=1000000 χρονικές μονάδες και η τιμή του εκθέτη Lyapunov που βρήκαμε είναι λ =-1.0·10⁻⁶. Στο σχήμα 10 φαίνεται σε λογαριθμικό διαγραμμα το φάσμα ισχύος. Και σε αυτήν την περίπτωση τόσο από τον εκθέτη Lyapunov όσο και από το φάσμα ισχύος φαίνεται πως η τροχιά είναι τακτική.



Σχήμα 9



Σχήμα 10

Στη συνέχεια για μ=1.43 δημιουργείται παράξενος ελκυστής. Στο σχήμα 11 ζωγραφίζουμε τον ελκυστή ενώ στο σχήμα 12 σε λογαριθμικό διάγραμμα ζωγραφίζουμε την εξέλιξη του εκθέτη Lyapunov για χρόνο t=5000000 χρονικές μονάδες και βρίσκουμε τιμή λ=0.01853.Επίσης στο σχήμα 13 αναπαρίσταται για την ίδια τροχιά το φάσμα ισχύος



Σχήμα 11



Σχήμα 12





Από τον εκθέτη Lyapunov και το φάσμα ισχύος διαπιστώνεται πως η τροχιά είναι χαοτική αφού στο φάσμα ισχύος φαίνεται να υπάρχουν όλες οι συχνότητες και ο εκθέτης Lyapunov τείνει στη θετική τιμή 0.01893. Στη συνέχεια ζωγραφίζουμε (σχήματα 14,15) την τομή Poincare του ελκυστή στο επίπεδο z=1 και μια μεγέθυνσή της. Από το σχήμα 15 διαπιστώνουμε την fractal δομή που δημιουργείται.



Σχήμα 14



Σχήμα 15

Παρακάτω στο σχήμα 16 ,δείχνουμε τη μορφή του ελκυστή για μ=1.55 καθώς και την εξέλιξη του εκθέτη Lyapunov σε λογαριθμικό διάγραμμα (σχήμα 16) και το φάσμα ισχύος (σχήμα 17).



Σχήμα 16



Σχήμα 17



Σχήμα 18

Ο εκθέτης Lyapunov για t=1000000 χρονικές μονάδες παίρνει την τιμή λ =0.03222. Σε συνδυασμό με το φάσμα ισχύος επιβεβαιώνεται πως ο ελκυστής είναι χαοτικός. Στα σχήματα 19,20 φαίνεται η απεικόνιση Poincare του προηγούμενου ελκυστή καθώς και μεγέθυνσή της, για περισσότερο υπολογιστικό χρόνο. Στην απεικόνιση Poincare φαίνεται η αυτό-ομοιότητα που υπάρχει στη δομή του ελκυστή. Με το πρόγραμμα box_coun2 προσδιορίζεται η διάσταση χωρητικότητας της απεικόνισης Dc=1.071716, και με το πρόγραμμα boxcoun_3d προσδιορίζεται η διάσταση χωρητικότητας του ελκυστή Dc=2.299796.



Σχήμα 19



Σχήμα 20

Όπως αναφέρθηκε ένα μέτρο της αυτό-ομοιότητας είναι η διάσταση συσχέτισης του ελκυστή. Με το πρόγραμμα correlfirst βρίσκουμε τη διάσταση συσχέτισης περίπου ίση με $v \approx 1.58$. Ο υπολογισμός γίνεται για τις μικρές τιμές του r στην περιοχή του πλατό όπως φαίνεται στο σχήμα 21.



Σχήμα 21

Στη συνέχεια στο σχήμα 22 ζωγραφίζουμε την τροχιά του σημείου (0.67, 0.3, 5.5) για μ=1.65, την εξέλιξη του εκθέτη Lyapunov σε λογαριθμικό διάγραμμα (για t=1000000 χρονικές μονάδες λ =0.01571) και το φάσμα ισχύος.



Σχήμα 22



Σχήμα 23





Από τα προηγούμενα αποτελέσματα διαπιστώνουμε πως ο ελκυστής εξακολουθεί να είναι χαοτικός άλλα άρχισε να συμπτύσσεται πράγμα που συνεχίζεται για περαιτέρω αύξηση της παραμέτρου. Έτσι όταν μ=1.708 ο εκθέτης Lyapunov παίρνει τιμή λ=1.1*10⁻⁵ πράγμα που σημαίνει πως πλέον δεν υπάρχει χάος.



Σχήμα 25



Σχήμα 26



Σχήμα 27



Σχήμα 28

Συμπεράσματα

Συγκεντρώνοντας τα αποτελέσματα μπορούμε να χωρίσουμε τη συμπεριφορά του συστήματος ως προς την εξέλιξη των πληθυσμών σε τρεις κατηγορίες:

 Πριν την εμφάνιση οριακού κύκλου μ<0.9545454 η εξέλιξη των τροχιών είναι τέτοια που μετά από λίγες ταλαντώσεις οι τιμές των πληθυσμών να καταλήγουν σε ένα από τα σημεία ισορροπίας π.χ. για μ=-0.44, με αρχικές συνθήκες (2.75, 1, 1.2) σχήμα 22 ή κάποιος από τους πληθυσμούς να απειρίζεται π.χ. απειρισμό έχουμε για μ=-0.45 με αρχικές συνθήκες (1,7,1).



Σχήμα 29

Για μ=0.55 και μ=0.9 με αρχικούς πληθυσμούς (1 , 0.4 , 1) σχήμα 23, 24 οι ταλαντώσεις των πληθυσμών καταλήγουν στο σημείο (1,1,1).



Σχήμα 30



Σχήμα 31

2. Όταν εμφανίζεται ο οριακός κύκλος, αν οι αρχικοί πληθυσμοί ανήκουν στη λεκάνη έλξης του οριακού κύκλου, τότε αφού η εξέλιξη των τροχιών οδηγηθεί στον οριακό κύκλο από κάποιο χρόνο και μετά οι πληθυσμοί θα εκτελούν επαναλαμβανόμενες ταλαντώσεις, π.χ. για μ=1.2 το σημείο με αρχικές συνθήκες (1.1, 2, 1), όπως φαίνεται στο σχήμα 25.



Σχήμα 32

Όταν ο οριακός κύκλος είναι μεγαλύτερης περιόδου, π.χ. για μ=1.35 ή μ=1.38 (0.5 , 0.3 , 1), τότε οι ταλαντώσεις θα είναι ποιο σύνθετες, όπως φαίνεται στα σχήματα 26,27.



Σχήμα 33



Σχήμα 34

3. Όταν εμφανιστεί ο παράξενος ελκυστής αν οι αρχικές τιμές των πληθυσμών βρίσκονται στη λεκάνη έλξης του ελκυστή θα εκτελούν χαοτική κίνηση και οι τιμές των πληθυσμών μπορούν να υπάρξουν σε οποιονδήποτε συνδυασμό μέσα από τη λεκάνη έλξης του.



Σχήμα 35

4. Τέλος όταν η τιμή της παραμέτρου γίνει μ=1.708 οι τροχιές του συστήματος έχουν ξαναγίνει τακτικές π.χ.για αρχικές συνθήκες (0.5, 1, 0.1) και έχουμε την παρακάτω εξέλιξη των πληθυσμών.



Σχήμα 36

ΠΑΡΑΡΤΗΜΑ Α

Στην παραπάνω ανάλυση του συστήματος ήταν γνωστές οι διαφορικές εξισώσεις που αντιστοιχούσαν στο μοντέλο, που αντιπροσωπεύει το σύστημα και για να μελετήσουμε τη συμπεριφορά του δεν είχαμε παρά να λύσουμε το σύστημα των διαφορικών εξισώσεων.

Όμως στα περισσότερα πραγματικά προβλήματα δεν είναι γνωστές οι διαφορικές εξισώσεις που ελέγχουν ένα φαινόμενο. Σε τέτοιες περιπτώσεις στη διάθεσή μας έχουμε μόνο μια χρονοσειρά με τις τιμές ενός μεγέθους π.χ. οι τιμές ενός πληθυσμού σε τακτά χρονικά διαστήματα. Σε αυτήν την περίπτωση υπάρχουν μέθοδοι, με τη χρήση τηων οποίων μπορούν να εξαχθούν συμπεράσματα για το σύστημα που παράγει αυτήν τη χρονοσειρά. Η μέθοδος που θα χρησιμοποιήσουμε εμείς είναι η μέθοδος των υστερήσεων.

Στο παράρτημα αυτό κάνουμε την αντίστροφη διαδικασία, δηλαδή από τις τιμές του ενός πληθυσμού σε τακτά χρονικά διαστήματα, ανακατασκευάζουμε τον χώρο καταστάσεων με τέτοιο τρόπο ώστε στην τροχιά που προέκυψε να αποτυπώνονται οι κυριότερες ιδιότητες του συστήματος.

Κατά την ανακατασκευή τα σημεία της τροχιάς έχουν συντεταγμένες $x_i = [x_i, x_{i-\tau}, ..., x_{i-(m-1)\tau}]'$, όπου η διάσταση εμβύθισης *m* είναι ο αριθμός των παρατηρήσεων που γίνονται συνιστώσες του ανακατασκευασμένου σημείου και η υστέρηση τ ορίζει το χρονικό βήμα υστέρησης.

Η υστέρηση επιλέγεται από τη συνάρτηση αυτοσυσχέτισης (π.χ. όταν η αυτοσυσχέτιση γίνει για πρώτη φορά μηδέν), ενώ η διάσταση εμβύθισης επιλέγεται με τη μέθοδο των ψευδών κοντινότερων γειτόνων. Η μέθοδος των ψευδών κοντινότερων γειτόνων συνίσταται στο να διαπιστώσουμε αν δύο γειτονικά σημεία μιας ανακατασκευής για δεδομένα μ, τ είναι πραγματικά γειτονικά ή ψευδή (λόγω κακής επιλογής του μ). Για αυτό βρίσκουμε το κοντινότερο σημείο στο υπό εξέταση σημείο και ελέγχουμε πόσο θα μεγαλώσει η απόστασή τους αν προσθέσουμε άλλη μία συνιστώσα στα σημεία (δηλαδή εξετάζουμε αν παραμένουν γειτονικά αυξάνοντας τη διάσταση εμβύθισης κατά ένα). Αν η απόσταση (εδώ χρησιμοποιούμε την ευκλείδεια απόσταση) μεγαλώσει δραματικά τότε τα σημεία δεν είναι πραγματικά γειτονικά. Αν το ποσοστό των ψευδών γειτονικών σημείων είναι μεγάλο, αυξάνουμε τη διάσταση εμβύθισης κατά ένα. Το πρόγραμμα μέσω του οποίου ανακατασκευάζουμε τον χώρο καταστάσεων είναι το diastash.cpp.

Μια εφαρμογή του τί πληροφορίες μπορούμε να πάρουμε με την παραπάνω μέθοδο φαίνεται στο πρόγραμμα correlationdimension.cpp, στο οποίο βρίσκουμε τη διάσταση συσχέτισης για διάφορες ανακατασκευές. Τα αποτελέσματα φαίνονται στο παρακάτω σχήμα 37 στο οποίο παρατηρούμε πως η εκτίμηση της διάστασης συσχέτισης καθώς αυξάνει η διάσταση εμβύθισης συγκλίνουν σε μια τιμή κοντά στο $ν \approx 1.7$, αποτέλεσμα που είναι πολύ κοντά με τον απευθείας υπολογισμό που έγινε στο σχήμα 21 παρά το γεγονός πως για λόγους υπολογιστικής ισχύος δεν χρησιμοποιήσαμε μεγάλες χρονοσειρές.



Σχήμα 37

ПАРАРТНМА В

Τα προγράμματα που χρησιμοποιήθηκαν είναι τα εξής

voltisoropia_ola_22new.nb

Με το πρόγραμμα αυτό υπολογίζονται τα σημεία ισορροπίας, η ευστάθεια τους η θέση τους στο χώρο για διάφορες τιμές της παραμέτρου και στη συνέχεια ζωγραφίζεται η εξέλιξη της θέσης συνάρτησει της παραμέτρου. Τα αποτελέσματα που παράγει είναι της μορφής:



Το κόκκινο σημαίνει ασταθές σημείο και το μπλε ευσταθές. Το μέγεθος αντιστοιχεί στο είδος του σημείου Σάγμα:το μικρό, "Εστία":το μεσαίο, Κόμβος:το μεγάλο.

```
ClearAll["Global`*"];
<<Graphics `Graphics
<<Graphics`Graphics3D`
a11=0.5;a12=0.5;a13=0.1;a21=-0.5;a22=-
0.1;a23=0.1;a31=m;a32=0.1;a33=0.1;
f[1] = x^{*} (a11^{*} (1-x) + a12^{*} (1-y) + a13^{*} (1-z));
f[2]=y^{*}(a21^{*}(1-x)+a22^{*}(1-y)+a23^{*}(1-z));
f[3]=z*(a31*(1-x)+a32*(1-y)+a33*(1-z));
sol=Solve[{f[1]==0, f[2]==0, f[3]==0}, {x, y, z}];
l=Length[sol];
xx=Table[0,{i,1,1},{j,1,3}];
jacobian=Table[{D[f[i],x],D[f[i],y],D[f[i],z]},{i,1,3}];
xx1=Table[0,{i,1,1}];
xx2=Table[0,{i,1,1}];
xx3=Table[0,{i,1,1}];
For[i=0,i<1,</pre>
    xx[[i,1]]=x/.sol[[i]];
    xx1[[i]]=xx[[i,1]];
    xx[[i,2]]=y/.sol[[i]];
    xx2[[i]]=xx[[i,2]];
    xx[[i,3]]=z/.sol[[i]];
    xx3[[i]]=xx[[i,3]]
```

```
;
i++];
dm=0.004;
mmin=-1.1;
mmax=1.2;
aaaa={ };
point=0;
bagr=1;
For[i=0,i<8,
    listblue[i][1]={};
    listblue[i][2]={};
    listblue[i][3]={};
    listblue[i][4]={};
    listred[i][1]={};
    listred[i][2]={};
    listred[i][3]={};
    listred[i][4]={};
    listidiotim[i][1]={};
    listidiotim[i][2]={};
    listidiotim[i][3]={};
    listidiotimfocusimagine[i][a]={};
    listidiotimfocusimagine[i][b]={};
    listidiotimfocusSigred[i]={};
    listidiotimfocusSigblew[i]={};
    listidiotimfocusreal[i]={};
    inplotfoull[i]={};
    indexbefore[i]=0;
    inble[i]={};
    inred[i]={};
    pointsize[i]={};
    pointsizefoc[i]={};
    index11[i]={};
    index[i]={};
    diktis[i]={}
    ,i++];
For[m=mmin,m<mmax,</pre>
    For[i=0,i<Length[sol],</pre>
       jacobian1[i]=jacobian/.{x 	o xx[[i,1]], y 	o xx[[i,2]], z 	o xx[[i,3]]};
      ll[i]=Eigenvalues[jacobian1[i]];
      vec[i]=Eigenvectors[jacobian1[i]];
      For[j=0,j<3,</pre>
idiot[i][j]=l1[i][[j]];reidiot[i][j]=Re[idiot[i][j]];imidiot[i][j]=Im
[idiot[i][j]];
        vectors[i][j]=vec[i][[j]],
        j++];
      If[i==6,
```

```
AppendTo[aaaa,ll[i]];
        ];
      sig=0;
      For[j=0,j<3,</pre>
        If[reidiot[i][j]<0, sig=sig, sig=sig+1],</pre>
        i++1;
      If[sig==0,
        index[i]=1;index11[i]=Stable;inble[i]=1;inred[i]=0; (*Print["
Το ",i," ",m," σημείο ισορροπίας είναι Ευσταθες
                                                      "]*),
        index[i]=-
1; index11[i]=Unstable; inble[i]=0; inred[i]=1; (*Print[" To ",i,"
",m," σημείο ισοροπίας είναι Ασταθες
                                          "]*)
        1;
      (*If[indexbefore[i]*index[i]<0,Print[" i= ",i," Για την τιμή
του m = ", m, "εχουμε σημείο διακλάδωσης με αλλαγή ευστάθειας "]];*)
      diktis[i]=0;
      If[imidiot[i][1]==0 && imidiot[i][2]==0 && imidiot[i][3]==0,
        If[ (reidiot[i][1] < 0 && reidiot[i][2] < 0 && reidiot[i][3]</pre>
< 0) || (reidiot[i][1] > 0 && reidiot[i][2] > 0 && reidiot[i][3] > 0)
          ,(*Print["i= ",i," m= ",m," κομβος ",index[i],"
",index11[i]];*)diktis[i]=1;pointsize[i]=0.05;pointsizefoc[i]=0
          ,(*Print["i= ",i," m= ",m," σαγμα ",index[i],"
",index11[i]];*)diktis[i]=2;pointsize[i]=0.026;pointsizefoc[i]=0
          ],If[ ( reidiot[i][1] == 0 && reidiot[i][2]== 0
||reidiot[i][1] == 0&& reidiot[i][3] == 0||reidiot[i][2] == 0&&
reidiot[i][3] == 0)
,AppendTo[listidiotim[i][1]];AppendTo[listidiotim[i][2]];AppendTo[lis
tidiotim[i][3]];(*Print["i= ",i," m= ",m," κέντρο ",index[i],"
", index11[i]]; *) diktis[i]=3; pointsize[i]=0.1; pointsizefoc[i]=0
          ,(*Print["i= ",i," m= ",m," Εστία ",index[i],"
",index11[i]];*)diktis[i]=4;pointsize[i]=0.035;pointsizefoc[i]=0.0035
;
          1
        ];
      (*
        If[(diktisbefore[i]!=diktis[i]), Print["i= ",i," Για την τιμη
m = ",m, "έχουμε αλλαγή του είδους ισορροπίας"]];*)
      diktisbefore[i]=diktis[i];
      indexbefore[i]=index[i];
      AppendTo[listidiotimfocusSigred[i], {m,-10}];
      AppendTo[listidiotimfocusSigblew[i], {m, -10}];
      If[diktis[i] == 4,
        listidiotimfocusSigred[i]=Drop[listidiotimfocusSigred[i],-1];
        listidiotimfocusSigblew[i]=Drop[listidiotimfocusSigblew[i],-
1];
        For[j=0,j<3,</pre>
          If[Im[ll[i][[j]]]!=0,
              If[Re[l1[i][[j]]]<0,</pre>
```

```
AppendTo[listidiotimfocusSigblew[i], {m,i}];AppendTo[listidiotimfocusS
igred[i], {m, -10}],
AppendTo[listidiotimfocusSigred[i], {m, i}]; AppendTo[listidiotimfocusSi
gblew[i], {m, -10}];
                1;
              Break[];
              ];
          ,j++]
        ];
      If[diktis[i]==4,
        For[j=0,j<3,
If[Im[ll[i][[j]]]==0&&j==1,AppendTo[listidiotimfocusreal[i], {m,Re[ll[
i][[1]]]};AppendTo[listidiotimfocusimagine[i][a], {m,Re[ll[i][[2]]]}]
;AppendTo[listidiotimfocusimagine[i][b], {m, Im[ll[i][2]]}};Break[]];
If[Im[ll[i][[j]]]==0&&j==2,AppendTo[listidiotimfocusreal[i], {m,Re[ll[i
[[2]]}];AppendTo[listidiotimfocusimagine[i][a], {m, Re[ll[i][[1]]]}];
AppendTo[listidiotimfocusimagine[i][b], {m, Im[ll[i][[1]]]};Break[]];
If[Im[ll[i][[j]]]==0&&j==3,AppendTo[listidiotimfocusreal[i], {m,Re[ll[i
[[3]]}];AppendTo[listidiotimfocusimagine[i][a], {m, Re[ll[i][2]]}];
AppendTo[listidiotimfocusimagine[i][b], {m, Im[ll[i][[2]]]}];Break[]],j
++]];
If[index[i]==1,AppendTo[listblue[i][diktis[i]], {m,i}];AppendTo[listre
d[i][diktis[i]], {m,2}], AppendTo[listblue[i][diktis[i]], {m,-
2}];AppendTo[listred[i][diktis[i]], {m, i}]];
      i++1;
inplot[1]={};inplot[2]={};inplot[3]={};inplot[4]={};inplot[5]={};inpl
ot[6]={};inplot[7]={};inplot[8]={};
    AppendTo[inplot[1], {xx1[[1]], xx2[[1]], xx3[[1]]};
AppendTo[inplot[2], {xx1[[2]], xx2[[2]], xx3[[2]]}]; AppendTo[inplot[3], {
xx1[[3]],xx2[[3]],xx3[[3]]}];
    AppendTo[inplot[4], {xx1[[4]], xx2[[4]], xx3[[4]]};
    AppendTo[inplot[5], {xx1[[5]], xx2[[5]], xx3[[5]]};
AppendTo[inplot[6], {xx1[[6]], xx2[[6]], xx3[[6]]}]; AppendTo[inplot[7], {
xx1[[7]],xx2[[7]],xx3[[7]]};
    AppendTo[inplot[8], {xx1[[8]], xx2[[8]], xx3[[8]]}];
        AppendTo[inplotfoull[1], {xx1[[1]], xx2[[1]], xx3[[1]]}];
AppendTo[inplotfoull[2], {xx1[[2]], xx2[[2]], xx3[[2]]}]; AppendTo[inplot
foul1[3], {xx1[[3]], xx2[[3]], xx3[[3]]};
    AppendTo[inplotfoull[4], {xx1[[4]], xx2[[4]], xx3[[4]]}];
    AppendTo[inplotfoul1[5], {xx1[[5]], xx2[[5]], xx3[[5]]};
AppendTo[inplotfoull[6], {xx1[[6]], xx2[[6]], xx3[[6]]}; AppendTo[inplot
foull[7], {xx1[[7]], xx2[[7]], xx3[[7]]}];
AppendTo[inplotfoull[8], {xx1[[8]], xx2[[8]], xx3[[8]]}];
    ss=7;
    rr=8;
```

```
fq78=(xx[[ss,1]]-xx[[rr,1]])^2+(xx[[ss,2]]-
xx[[rr,2]])^2+(xx[[ss,3]]-xx[[rr,3]])^2;
    ss=1;
    rr=8;
    fg18=(xx[[ss,1]]-xx[[rr,1]])^2+(xx[[ss,2]]-
xx[[rr,2]])^2+(xx[[ss,3]]-xx[[rr,3]])^2;
    ss=2;
    rr=7;
    fg27=(xx[[ss,1]]-xx[[rr,1]])^2+(xx[[ss,2]]-
xx[[rr,2]])^2+(xx[[ss,3]]-xx[[rr,3]])^2;
    ss=4;
    rr=5;
    fq45=(xx[[ss,1]]-xx[[rr,1]])^2+(xx[[ss,2]]-
xx[[rr,2]])^2+(xx[[ss,3]]-xx[[rr,3]])^2;
    ss=5;
    rr=8;
    fq58=(xx[[ss,1]]-xx[[rr,1]])^2+(xx[[ss,2]]-
xx[[rr,2]])^2+(xx[[ss,3]]-xx[[rr,3]])^2;
If[fq18<0.001||fq78<0.001||fq27<0.001||fq45<0.001||fq58<0.001,bagr=0.
9;point++,bagr=1];
    If[fq18<0.1,simio=fg18;f="1-8",simio=∞;f="0-0"];</pre>
    If[fq78<0.1, simio=fq78; f="7-8", simio=∞; f="0-0"];</pre>
    If[fg27<0.1, simio=fg27; f="2-7", simio=\omega; f="0-0"];</pre>
    If[fq45<0.1, simio=fq45; f="4-5", simio=∞; f="0-0"];</pre>
    If[fq58<0.1,simio=fq58;f="5-8",simio=∞;f="0-0"];</pre>
s1=ScatterPlot3D[inplot[1],PlotStyle>{PointSize[pointsize[1]],RGBColo
r[inred[1],0,inble[1]]},DisplayFunction→Identity];
s2=ScatterPlot3D[inplot[2],PlotStyle>{PointSize[pointsize[2]],RGBColo
r[inred[2],0,inble[2]]},DisplayFunction→Identity];
s3=ScatterPlot3D[inplot[3],PlotStyle→{PointSize[pointsize[3]],RGBColo
r[inred[3],0,inble[3]]},DisplayFunction→Identity];
s4=ScatterPlot3D[inplot[4],PlotStyle→{PointSize[pointsize[4]],RGBColo
r[inred[4],0,inble[4]]},DisplayFunction→Identity];s5=ScatterPlot3D[in
plot[5],PlotStyle→{PointSize[pointsize[5]],RGBColor[inred[5],0,inble[
5]]},DisplayFunction→Identity];s6=ScatterPlot3D[inplot[6],PlotStyle→
{PointSize[pointsize[6]],RGBColor[inred[6],0,inble[6]]},DisplayFuncti
on→Identity];s7=ScatterPlot3D[inplot[7],PlotStyle→{PointSize[pointsi
ze[7]],RGBColor[inred[7],0,inble[7]]},DisplayFunction→Identity];s8=Sc
atterPlot3D[inplot[8],PlotStyle→{PointSize[pointsize[8]],RGBColor[inr
ed[8],0,inble[8]]},DisplayFunction→Identity];
    Print["m = ",m," ","fg78 = ",fg78," fg18 = ",fg18," fg27 =
",fg27," fg45 = ",fg45," fg58 = ",fg58];
    q11=Show[{s1,s2,s3,s4,s5,s6,s7,s8},PlotRange→{{-10,10}, {-8,8}, {-
15,15}}, ViewPoint→{0,-5,1}, DisplayFunction→Identity];
    g22=Show[{s1,s2,s3,s4,s5,s6,s7,s8},PlotRange→{{-10,10},{-8,8},{-
20,20},ViewPoint\rightarrow{4,0,1},DisplayFunction\rightarrowIdentity];g33=Show[{s1,s2,
s3, s4, s5, s6, s7, s8, PlotRange \rightarrow { {-10, 10 }, {-8, 8 }, {-
15,15}}, ViewPoint \rightarrow {4, -
2,0},AxesLabel→TraditionalForm/@{{"m=",m},{"dist=",simio},{f}},TextSt
yle→{FontFamily→"Times", FontSize→10}, DisplayFunction→Identity];
    Show[GraphicsArray[{g11,g22,g33}],Background→GrayLevel[bagr]
```

```
, DisplayFunction→$DisplayFunction];
```

```
m=m+dm];
```

,

ggg61=ListPlot[listidiotimfocusSigred[6],PlotRange→{{mmin,mmax},{0,10 }},PlotStyle→{RGBColor[1,0,0],PointSize[0.0045]},DisplayFunction→Ide ntity]; ggg62=ListPlot[listidiotimfocusSigblew[6],PlotRange→{{mmin,mmax},{0,1 0}, PlotStyle >{RGBColor[0,0,1], PointSize[0.0045]}, DisplayFunction >Id entity];qqq31=ListPlot[listidiotimfocusSigred[3],PlotRange→{{mmin,mma x}, {0,10}}, PlotStyle \rightarrow {RGBColor[1,0,0], PointSize[0.0045]}, DisplayFunct $ion \rightarrow Identity];$ $qqg32=ListPlot[listidiotimfocusSigblew[3],PlotRange \rightarrow \{\{mmin,mmax\}, \{0,1\}\}$ 0}},PlotStyle→{RGBColor[0,0,1],PointSize[0.0045]},DisplayFunction→Id entity];gqq51=ListPlot[listidiotimfocusSigred[5],PlotRange→{{mmin,mma x},{0,10}},PlotStyle→{RGBColor[1,0,0],PointSize[0.0045]},DisplayFunct ion→Identity]; ggg52=ListPlot[listidiotimfocusSigblew[5],PlotRange→{{mmin,mmax},{0,1 0},PlotStyle>{RGBColor[0,0,1],PointSize[0.0045]},DisplayFunction>Id entity];ggg71=ListPlot[listidiotimfocusSigred[7],PlotRange→{{mmin,mma x}, {0,10}}, PlotStyle > {RGBColor[1,0,0], PointSize[0.0045]}, DisplayFunct $ion \rightarrow Identity];$ $qqq72=ListPlot[listidiotimfocusSigblew[7],PlotRange \rightarrow \{\{mmin,mmax\}, \{0,1\}\}$ 0}, PlotStyle→{RGBColor[0,0,1], PointSize[0.0045]}, DisplayFunction→Id entity];

```
For[i=0,i<Length[sol],
    For[j=0,j<4,</pre>
```

If[Length[listblue[i][j]]>0,If[j=1,grblue[i][j]=ListPlot[listblue[i] [j],PlotRange→{{mmin,mmax},{0,10}},PlotStyle→{RGBColor[0,0,1],PointS ize[0.012]},DisplayFunction→Identity]];If[j=2,grblue[i][j]=ListPlot[listblue[i][j],PlotRange→{{mmin,mmax},{0,10}},PlotStyle→{RGBColor[0, 0,1],PointSize[0.0015]},DisplayFunction→Identity]];

If[j=3,grblue[i][j]=ListPlot[listblue[i][j],PlotRange→{{mmin,mmax},{
0,10}},PlotStyle→{RGBColor[0,0,1],PointSize[0.1]},DisplayFunction→Id
entity]];If[j=4,grblue[i][j]=ListPlot[listblue[i][j],PlotRange→{{mmi
n,mmax},{0,10}},PlotStyle→{RGBColor[0,0,1],PointSize[0.006]},DisplayF
unction→Identity]]

```
];
```

If[Length[listred[i][j]]>0,

If[j==1,grred[i][j]=ListPlot[listred[i][j],PlotRange→{{mmin,mmax},{0, 10}},PlotStyle→{RGBColor[1,0,0],PointSize[0.012]},DisplayFunction→Id entity]];If[j==2,grred[i][j]=ListPlot[listred[i][j],PlotRange→{{mmin, mmax},{0,10}},PlotStyle→{RGBColor[1,0,0],PointSize[0.0015]},DisplayFu nction→Identity]];If[j==3,grred[i][j]=ListPlot[listred[i][j],PlotRang e→{{mmin,mmax},{0,10}},PlotStyle→{RGBColor[1,0,0],PointSize[0.1]},Di splayFunction→Identity]];

```
If[j==4,grred[i][j]=ListPlot[listred[i][j],PlotRange→{{mmin,mmax},{0,
10}},PlotStyle→{RGBColor[1,0,0],PointSize[0.006]},DisplayFunction→Id
entity]]
]
```

```
j++]
    ,
    i++
    ];
gr={ };
For[i=0,i<Length[sol],</pre>
 For[j=0,j<4,
If[Length[listred[i][j]]>0,AppendTo[gr,grred[i][j]];If[Length[listbl
ue[i][j]]>0,AppendTo[gr,grblue[i][j]]],
    j++],
  i++]
Show[{gr,ggg61,ggg62,ggg51,ggg52,ggg31,ggg32,ggg32,ggg72,ggg72},Displ
ayFunction→$DisplayFunction,AxesLabel→{ "m"," Point
"},Frame→True,AxesOrigin→{0,1}];
Show[gr,DisplayFunction→$DisplayFunction,AxesLabel→{" m "," Point
"},Frame→True,AxesOrigin→{0,1}];
```

volt_i1.nb,volt_i2.nb,volt_i3.nb,volt_i4.nb,volt_i5.nb,volt_i6.nb,volt_i7.nb,volt_i8 .nb //Με αυτά το πρόγραμμα βρίσκω τις ιδιοτιμές για τα σημεία ισορροπίας για κάθε τιμή της παραμέτρου

```
ClearAll["Global`*"];
<<Graphics `Graphics
<<Graphics`Graphics3D`
a11=0.5;a12=0.5;a13=0.1;a21=-0.5;a22=-
0.1;a23=0.1;a31=m;a32=0.1;a33=0.1;
f[1] = x^{*} (a11^{*} (1-x) + a12^{*} (1-y) + a13^{*} (1-z));
f[2]=y*(a21*(1-x)+a22*(1-y)+a23*(1-z));
f[3]=z*(a31*(1-x)+a32*(1-y)+a33*(1-z));
sol=Solve[{f[1]==0, f[2]==0, f[3]==0}, {x, y, z}];
l=Length[sol];
xx=Table[0,{i,1,1},{j,1,3}];
jacobian=Table[{D[f[i],x],D[f[i],y],D[f[i],z]},{i,1,3}];
xx1=Table[0, {i,1,1}];
xx2=Table[0,{i,1,1}];
xx3=Table[0,{i,1,1}];
For[i=0,i<1,</pre>
    xx[[i,1]]=x/.sol[[i]];
    xx1[[i]]=xx[[i,1]];
    xx[[i,2]]=y/.sol[[i]];
    xx2[[i]]=xx[[i,2]];
```

```
xx[[i,3]]=z/.sol[[i]];
           xx3[[i]]=xx[[i,3]];
           Print[ "x ",i," =",xx[[i,1]]," y ",i," =",xx[[i,2]], " z ",i,"
=", xx[[i,3]]]
           i++];
jacobian//MatrixForm;
Clear[L];
i=1;
Clear[m];
jacobian1[i]=Evaluate[jacobian//N]/.{x→xx[[i,1]],y→xx[[i,2]],z→xx[[i
 ,3]]};
ll[i]=Eigenvalues[jacobian1[i]];
lx1=ll[i][[1]]
ly2=ll[i][[2]]
lz3=ll[i][[3]]
Eigenvectors[jacobian1[i]][[1]]//MatrixForm
Eigenvectors[jacobian1[i]][[2]]//MatrixForm
Eigenvectors[jacobian1[i]][[3]]//MatrixForm
list6x={};
list6y={};
list6z={};
hh1={};
dm1=0.001;
mmin=-2;
mmax=2;
For[m1=mmin,m1<mmax,</pre>
     idx=lx1/.m→m1;
     idy=ly2/.m→m1;
     idz=lz3/.m→m1;
     AppendTo[hh1, {m1, idx, idy, idz}];
     If[Im[idx] == 0, AppendTo[list6x, {m1, idx}],
           AppendTo[list6x, {m1, -10}]];
     If[Im[idy] == 0, AppendTo[list6y, {m1, idy}],
           AppendTo[list6y, {m1, -10}]];
     If[Im[idz]==0,AppendTo[list6z,{m1,idz}],
           AppendTo[list6z, {m1, -10}]];
     ,m1=m1+dm1]
rrx6=ListPlot[list6x,PlotStyle→{RGBColor[1,0,0],PointSize[0.005]},Dis
playFunction \rightarrow Identity]; rry6=ListPlot[list6y, PlotStyle \rightarrow {RGBColor[1,0, Interval and Interv
0], PointSize[0.005]}, DisplayFunction→Identity];
rrz6=ListPlot[list6z,PlotStyle→{RGBColor[1,0,0],PointSize[0.005]},Dis
playFunction→Identity];
list6xRe={};
list6xIm={};
list6yRe={};
list6yIm={};
list6zRe={};
```

```
list6zIm={};
list={};
dm1=0.001;
For[m1=mmin,m1<mmax,</pre>
  idx=lx1/.m→m1;
  idy=ly2/.m→m1;
  idz=lz3/.m→m1;
If[Re[idx]<=0&&Re[idy]<=0&&Re[idz]<=0,AppendTo[list,{m1,1}],AppendTo[</pre>
list, {m1, -1}]];
  If [Im[idx] \neq 0,
    AppendTo[list6xIm, {m1, Im[idx]}];
    AppendTo[list6xRe, {m1, Re[idx]}],
    AppendTo[list6xIm, {m1, -10}];
    AppendTo[list6xRe, {m1, -10}]
    1;
  If [Im[idy] \neq 0,
    AppendTo[list6yIm, {m1, Im[idy]}];
    AppendTo[list6yRe, {m1, Re[idy]}],
    AppendTo[list6yIm, {m1, -10}];
    AppendTo[list6yRe, {m1, -10}]
    1;
  If [Im[idz] \neq 0,
    AppendTo[list6zIm, {m1, Im[idz]}];
    AppendTo[list6zRe, {m1, Re[idz]}],
    AppendTo[list6zIm, {m1, -10}];
    AppendTo[list6zRe, {m1, -10}]
    ];
  ,m1=m1+dm1]
iix6=ListPlot[list6xIm,PlotStyle→{RGBColor[0,1,0],PointSize[0.0005]},
DisplayFunction→Identity];
irx6=ListPlot[list6xRe,PlotStyle→{RGBColor[0,0,1],PointSize[0.0001]},
DisplayFunction→Identity];
iiy6=ListPlot[list6yIm,PlotStyle→{RGBColor[0,1,0],PointSize[0.0005]},
DisplayFunction→Identity];
iry6=ListPlot[list6yRe,PlotStyle>{RGBColor[0,0,1],PointSize[0.0001]},
iiz6=ListPlot[list6zIm,PlotStyle→{RGBColor[0,1,0],PointSize[0.0005]},
DisplayFunction→Identity];
irz6=ListPlot[list6zRe,PlotStyle→{RGBColor[0,0,1],PointSize[0.0001]},
g1=Show[{irx6, rrx6, iix6}, AxesLabel\rightarrow{"\mu", "\lambda1"}, Frame\rightarrowTrue, DisplayFunc
tion→$DisplayFunction,PlotRange→{{mmin,mmax},{-
2,2}},AxesOrigin\rightarrow{0,0}];
```

```
g2=Show[\{iry6, iiy6, rry6\}, AxesLabel \rightarrow \{"\mu", "\lambda2"\}, Frame \rightarrow True, DisplayFunction \rightarrow DisplayFunction, PlotRange \rightarrow \{\{mmin, mmax\}, \{-2,2\}\}, AxesOrigin \rightarrow \{0,0\}];
g3=Show[\{irz6, rrz6, iiz6\}, AxesLabel \rightarrow \{"\mu", "\lambda3"\}, DisplayFunction \rightarrow DisplayFunction, PlotRange \rightarrow \{\{mmin, mmax\}, \{-2,2\}\}, AxesOrigin \rightarrow \{0,0\}, Frame \rightarrow True];
g4=ListPlot[list, PlotStyle \rightarrow RGBColor[1,0,0], AxesLabel \rightarrow \{"\mu", "EYSTA \\ PlotRange \rightarrow \{\{mmin, mmax\}, \{-2,2\}\}\}
Frame \rightarrow True, PlotRange \rightarrow \{\{mmin, mmax\}, \{-2,2\}\}]
Show[GraphicsArray[\{\{g1,g2\}, \{g3,g4\}\}], PlotLabel \rightarrow \{"EMEIOINE \\ ISOPPOHIAS (0,0,0)"\}]
```

Το πρόγραμμα με το οποίο λύνω αριθμητικά το σύστημα των διαφορικών εξισώσεων είναι το παρακάτω. Η μέθοδος που χρησιμοποιείται είναι η Runke Kutta Fehlberg

```
volterra.cpp
```

}

```
#include "stdafx.h"
#include "initial.h"
#include "runge kutta fehberg.h"
#include <fstream>
#include "math.h"
using namespace std;
extern double m[13];
void main(void) {
      double m1=1.708;
      ofstream file graf("graf1.dat");
      initial(0.5,0.5,0.1,-0.5,-0.1,0.1,m1,0.1,0.1,0.0,0.0,0.0);
    int n=18;
    double x1[4];
      double tmax, h=0.02;
      tmax=h*pow(2,n)-h;
      //tmax=10000;
      x1[0]=0;
      x1[1]=0.668675;
      x1[2]=0.298598;
      x1[3]=5.4985;
      runge kutta fehlberg(x1,tmax,h);
      //file graf<<x1[0]<<"\t"<<x1[1]<<"\t"<<x1[2]<<"\t"<<x1[3]<<endl
;
      cout<<"gia xrono tmax= "<<x1[0]<<" exoume ";</pre>
      cout<<x1[0]<<"\t"<<x1[1]<<"\t"<<x1[2]<<"\t"<<x1[3]<<endl;
```

runge_kutta_fehberg.cpp //αριθμητική ολοκλήρωση με τη Runge Kutta Fehlberg

```
#include "stdafx.h"
#include "runge kutta fehberg.h"
#include "math.h"
#include "dynsystem.h"
using namespace std;
extern double m[13];
FILE *fill=fopen("graf.dat", "wt");
FILE *filz=fopen("grafz.dat","wt");
void runge kutta fehlberg(double xin[4], double t1, double dx) {
      double t=0.0;
      double h;
      double xbar[4];
      double xout[4];
      double x0,x1,x2,x3;
      double k1,k2,k3,k4,k5,k6,l1,l2,l3,l4,l5,l6,r1,r2,r3,r4,r5,r6;
      double tmax=t1;
      int ii=0,ij=0;
      while (xin[0]<tmax) {</pre>
            h=dx;
            ii=ii+1;
            ij=ij+1;
      double jj=0;
      do {
    t=xin[0];
      x0=xin[0];x1=xin[1];x2=xin[2];x3=xin[3];//x0,x1,x2,x3
      k1=h*f1(x0,x1,x2,x3);
      l1=h*f2(x0,x1,x2,x3);
      r1=h*f3(x0,x1,x2,x3);
      k2=h*f1(x0+0.25*h,x1+0.25*k1,x2+0.25*l1,x3+0.25*r1);
      l2=h*f2(x0+0.25*h,x1+0.25*k1,x2+0.25*l1,x3+0.25*r1);
      r2=h*f3(x0+0.25*h,x1+0.25*k1,x2+0.25*l1,x3+0.25*r1);
      k3=h*f1(x0+3.0*h/8.0,x1+3.0*k1/32.0+9.0*k2/32.0,x2+3.0*11/32.0+
9.0*12/32.0,x3+3.0*r1/32.0+9.0*r2/32.0);
      13=h*f2(x0+3.0*h/8.0,x1+3.0*k1/32.0+9.0*k2/32.0,x2+3.0*11/32.0+
9.0*12/32.0,x3+3.0*r1/32.0+9.0*r2/32.0);
      r3=h*f3(x0+3.0*h/8.0,x1+3.0*k1/32.0+9.0*k2/32.0,x2+3.0*l1/32.0+
9.0*12/32.0,x3+3.0*r1/32.0+9.0*r2/32.0);
```

```
k4=h*f1(x0+12.0*h/13.0,x1+1932.0/2197.0*k1-
7200.0/2197.0*k2+7296.0/2197.0*k3,x2+1932.0/2197.0*11-
7200.0/2197.0*12+7296.0/2197.0*13,x3+1932.0/2197.0*r1-
7200.0/2197.0*r2+7296.0/2197.0*r3);
      14=h*f2(x0+12.0*h/13.0,x1+1932.0/2197.0*k1-
7200.0/2197.0*k2+7296.0/2197.0*k3,x2+1932.0/2197.0*l1-
7200.0/2197.0*12+7296.0/2197.0*13,x3+1932.0/2197.0*r1-
7200.0/2197.0*r2+7296.0/2197.0*r3);
      r4=h*f3(x0+12.0*h/13.0,x1+1932.0/2197.0*k1-
7200.0/2197.0*k2+7296.0/2197.0*k3,x2+1932.0/2197.0*11-
7200.0/2197.0*12+7296.0/2197.0*13,x3+1932.0/2197.0*r1-
7200.0/2197.0*r2+7296.0/2197.0*r3);
      k5=h*f1(x0+h,x1+493.0/216.0*k1-8.0*k2+3680.0/513.0*k3-
845.0/4104.0*k4,x2+493.0/216.0*11-8.0*12+3680.0/513.0*13-
845.0/4104.0*14,x3+493.0/216.0*r1-8.0*r2+3680.0/513.0*r3-
845.0/4104.0*r4);
      15=h*f2(x0+h,x1+493.0/216.0*k1-8.0*k2+3680.0/513.0*k3-
845.0/4104.0*k4,x2+493.0/216.0*11-8.0*12+3680.0/513.0*13-
845.0/4104.0*14,x3+493.0/216.0*r1-8.0*r2+3680.0/513.0*r3-
845.0/4104.0*r4);
      r5=h*f3(x0+h,x1+493.0/216.0*k1-8.0*k2+3680.0/513.0*k3-
845.0/4104.0*k4,x2+493.0/216.0*11-8.0*12+3680.0/513.0*13-
845.0/4104.0*14,x3+493.0/216.0*r1-8.0*r2+3680.0/513.0*r3-
845.0/4104.0*r4);
      k6=h*f1(x0+0.5*h,x1-8.0/27.0*k1+2.0*k2-
3544.0/2565.0*k3+1859.0/4104.0*k4-11.0/40.0*k5,x2-8.0/27.0*11+2.0*12-
3544.0/2565.0*13+1859.0/4104.0*14-11.0/40.0*15,x3-8.0/27.0*r1+2.0*r2-
3544.0/2565.0*r3+1859.0/4104.0*r4-11.0/40.0*r5);
      16=h*f2(x0+0.5*h,x1-8.0/27.0*k1+2.0*k2-
3544.0/2565.0*k3+1859.0/4104.0*k4-11.0/40.0*k5,x2-8.0/27.0*l1+2.0*l2-
3544.0/2565.0*13+1859.0/4104.0*14-11.0/40.0*15,x3-8.0/27.0*r1+2.0*r2-
3544.0/2565.0*r3+1859.0/4104.0*r4-11.0/40.0*r5);
      r6=h*f3(x0+0.5*h,x1-8.0/27.0*k1+2.0*k2-
3544.0/2565.0*k3+1859.0/4104.0*k4-11.0/40.0*k5,x2-8.0/27.0*11+2.0*12-
3544.0/2565.0*13+1859.0/4104.0*14-11.0/40.0*15,x3-8.0/27.0*r1+2.0*r2-
3544.0/2565.0*r3+1859.0/4104.0*r4-11.0/40.0*r5);
      xbar[1]=x1+(25.0/216.0*k1+1408.0/2565.0*k3+2197.0/4104.0*k4-
0.2*k5);
      xbar[2]=x2+(25.0/216.0*11+1408.0/2565.0*13+2197.0/4104.0*14-
0.2*15);
      xbar[3]=x3+(25.0/216.0*r1+1408.0/2565.0*r3+2197.0/4104.0*r4-
0.2*r5);
      xout[1]=x1+(16.0/135.0*k1+6656.0/12825.0*k3+28561.0/56430*k4-
9.0/50.0*k5+2.0/55.0*k6);
      xout[2]=x2+(16.0/135.0*11+6656.0/12825.0*13+28561.0/56430*14-
9.0/50.0*15+2.0/55.0*16);
      xout[3]=x3+(16.0/135.0*r1+6656.0/12825.0*r3+28561.0/56430*r4-
9.0/50.0*r5+2.0/55.0*r6);
```



```
if(fabs(xbar[1]-xout[1])>(1.0/(2.0))*h)
{
```

```
h=0.5*h;jj=jj+1;
                                               }
      }
while (fabs(xbar[1]-xout[1])>1.0/2.0*h);
      t=t+h;
     xout[0]=t;
     for(int i=0;i<5;i++) {</pre>
           xin[i]=xout[i];
      }
     if (ii==5)
      {
           ii=0;
fprintf(fil1,"% f % f % f % f \n",xin[0],xin[1],xin[2],xin[3]);
fprintf(filz,"% f \n",xin[2]);
      }
      }
     fclose(fil1); fclose(filz);}
dynsystem.cpp //Το δυναμικό σύστημα
#include "stdafx.h"
#include "dynsystem.h"
#include "math.h"
#include "initial.h"
extern double m[13];
double f1(double xx0, double xx1, double xx2, double xx3)
{
     return ( xx1* (m[1]*(1-xx1)+m[2]*(1-xx2)+m[3]*(1-xx3)));
}
double f2(double xx0, double xx1, double xx2, double xx3)
{
     return( xx2*(m[4]*(1-xx1)+m[5]*(1-xx2)+m[6]*(1-xx3)));
}
double f3(double xx0, double xx1, double xx2, double xx3)
{
     return ( xx3* (m[7]*(1-xx1)+m[8]*(1-xx2)+m[9]*(1-xx3)));
}
```

initial.cpp //αρχικοποίηση παραμέτρων

```
#include "stdafx.h"
#include "initial.h"
double m[13];
int initial(double m11,double m22,double m33,double m44,double
m55,double m66,double m77,double m88,double m99,double m1010,double
m1111,double m1212)
{
m[1]=m11;m[2]=m22;m[3]=m33;m[4]=m44;m[5]=m55;m[6]=m66;m[7]=m77;
m[8]=m88;m[9]=m99;m[10]=m1010;m[11]=m1111;m[12]=m1212;
```

```
return(1);
```

}

poincare_volt.cpp //Toµή Poincare

```
#include "stdafx.h"
#include <cmath>
#include "initial.h"
#include "section_poincare.h"
using namespace std;
extern double m[13];
void main(void) {
      FILE *fill=fopen("graf.dat","wt");
      double m1=1.708;
        initial(0.5,0.5,0.1,-0.5,-0.10,0.1,m1,0.1,0.1,0.0,0.0,0.0);
      double x1[4];
      double h0=0.1;
      double tmax=2000000;
      //initial condition
      x1[0]=0;
      x1[1]=0.16501;
      x1[2]=1.531;
      x1[3]=1.0;
      int i=1;
      while (x1[0]<tmax) {</pre>
            i=i+1;
            sectionp(x1,h0);
            fprintf(fil1,"% f % f \n",x1[1],x1[2]);
            }
            fclose(fil1);
   }
```

section_poincare.cpp

```
#include "stdafx.h"
#include "section_poincare.h"
#include "runge_kutta_fehberg.h"
#include "math.h"
double x0,x1,x2,x3,x3before,h;
void sectionp(double xinp[4],double dx){
    double xi[5];
    xi[0]=xinp[0];
    xi[1]=xinp[1];
    xi[2]=xinp[2];
    xi[3]=xinp[3];
    xi[4]=dx;
```

```
do{
   xi[4]=0.5*xi[4];
   h=xi[4];
   xinp[0]=xi[0];
   xinp[1]=xi[1];
   xinp[2]=xi[2];
   xinp[3]=xi[3];
   do{
      xi[0]=xinp[0];
      xi[1]=xinp[1];
      xi[2]=xinp[2];
      xi[3]=xinp[3];
      h=xi[4];
        runge kutta fehlberg(xinp,h);
      if (xinp[3]==1.0) break;
      }while ( !(( xinp[3]>=1.0 ) && ( xi[3]<=1.0 )) );</pre>
      if (xinp[3]==1.0) break;
}while(fabs(xinp[3]-1.0)>0.000001);
```

```
}
```

liap_volter.cpp //Εκθέτης Lyapunov με χρήση του γραμμικού συστήματος

```
#include "stdafx.h"
#include "math.h"
#include "initial.h"
#include "runge kutta fehberg.h"
#include "rungegramiko.h"
#include <fstream>
using namespace std;
extern double m[13];
void main(void)
{
      double m1=1.708;
      FILE *fill=fopen("graf1.dat", "wt");
      FILE *fil2=fopen("graf2.dat","wt");
      initial(0.5,0.5,0.1,-0.5,-0.10,0.1,m1,0.1,0.1,0.0,0.0,0.0);
      double x1[4], x2[4], dx1[4];
      double liap=0,jn,an=0;
      double tmax=1000000, h=0.01;
      x1[0]=0; //initial condition
      x1[1]=0.668675;
      x1[2]=0.298598;
```
```
x1[3]=5.4985;
      dx1[0]=0;
      dx1[1]=1;//
      dx1[2]=0;
      dx1[3]=0;
      int i=0;
      double n1=0;
      while (x1[0]<tmax) {</pre>
            n1=n1+1;
            i=i+1;
             x2[0]=x1[0];
             x2[1]=x1[1];
             x2[2]=x1[2];
             x2[3]=x1[3];
            runge kutta fehlberg(x1,h);
            rungegram(dx1,x1,h);
jn=sqrt( (dx1[1])*(dx1[1])+(dx1[2])*(dx1[2])+(dx1[3])*(dx1[3]) );
            liap=liap+log( jn );
            if(i==5000){
                   cout<<x1[0]<<endl;</pre>
                   i=0;
fprintf(fil1,"% f % f % f % f \n",x1[0],x1[1],x1[2],x1[3]);
      fprintf(fil2,"% f % f \n",x1[0],liap/x1[0]);
        }
            dx1[1]=dx1[1]/jn;
            dx1[2]=dx1[2]/jn;
            dx1[3]=dx1[3]/jn;
      }
      fclose(fil1);
      fclose(fil2);
}
rungegramiko.cpp
#include "stdafx.h"
#include "math.h"
#include "rungegramiko.h"
#include "sustimagram.h"
using namespace std;
extern double m[13];
void rungegram(double dxin[4], double xin[4], double dx) {
      double t=0.0;
      double h=dx;
      double xbar[4];
      double xout[4];
      double x0,x1,x2,x3,y10,y20,y30;
      double k1, k2, k3, k4, k5, k6, l1, l2, l3, l4, l5, l6, r1, r2, r3, r4, r5, r6;
```

double jj=0; do { t=dxin[0]; x0=dxin[0];x1=dxin[1];x2=dxin[2];x3=dxin[3];//z0,z1,z2,z3 y10=xin[1];y20=xin[2];y30=xin[3];//x1,x2,x3 k1=h*f11(x0, y10,y20,y30, x1,x2,x3); l1=h*f22(x0, y10,y20,y30, x1,x2,x3); r1=h*f33(x0, y10,y20,y30, x1,x2,x3); k2=h*f11(x0+0.25*h, y10,y20,y30, x1+0.25*k1,x2+0.25*11,x3+0.25*r1); 12=h*f22(x0+0.25*h, y10,y20,y30, x1+0.25*k1,x2+0.25*l1,x3+0.25*r1); r2=h*f33(x0+0.25*h, y10,y20,y30, x1+0.25*k1,x2+0.25*11,x3+0.25*r1); k3=h*f11(x0+3.0*h/8.0, y10,y20,y30, x1+3.0*k1/32.0+9.0*k2/32.0,x2+3.0*l1/32.0+9.0*l2/32.0,x3+3.0*r1/32.0+ 9.0*r2/32.0); 13=h*f22(x0+3.0*h/8.0, y10,y20,y30, x1+3.0*k1/32.0+9.0*k2/32.0,x2+3.0*l1/32.0+9.0*l2/32.0,x3+3.0*r1/32.0+ 9.0*r2/32.0); r3=h*f33(x0+3.0*h/8.0, y10,y20,y30, x1+3.0*k1/32.0+9.0*k2/32.0,x2+3.0*11/32.0+9.0*12/32.0,x3+3.0*r1/32.0+ 9.0*r2/32.0); k4=h*f11(x0+12.0*h/13.0, y10,y20,y30, x1+1932.0/2197.0*k1-7200.0/2197.0*k2+7296.0/2197.0*k3,x2+1932.0/2197.0*l1-7200.0/2197.0*12+7296.0/2197.0*13,x3+1932.0/2197.0*r1-7200.0/2197.0*r2+7296.0/2197.0*r3); 14=h*f22(x0+12.0*h/13.0, y10,y20,y30, x1+1932.0/2197.0*k1-7200.0/2197.0*k2+7296.0/2197.0*k3,x2+1932.0/2197.0*11-7200.0/2197.0*12+7296.0/2197.0*13,x3+1932.0/2197.0*r1-7200.0/2197.0*r2+7296.0/2197.0*r3); r4=h*f33(x0+12.0*h/13.0, y10,y20,y30, x1+1932.0/2197.0*k1-7200.0/2197.0*k2+7296.0/2197.0*k3,x2+1932.0/2197.0*l1-7200.0/2197.0*12+7296.0/2197.0*13,x3+1932.0/2197.0*r1-7200.0/2197.0*r2+7296.0/2197.0*r3); k5=h*f11(x0+h, y10,y20,y30, x1+493.0/216.0*k1-8.0*k2+3680.0/513.0*k3-845.0/4104.0*k4,x2+493.0/216.0*11-8.0*12+3680.0/513.0*13-845.0/4104.0*14,x3+493.0/216.0*r1-8.0*r2+3680.0/513.0*r3-845.0/4104.0*r4); 15=h*f22(x0+h, y10,y20,y30, x1+493.0/216.0*k1-8.0*k2+3680.0/513.0*k3-845.0/4104.0*k4,x2+493.0/216.0*11-8.0*12+3680.0/513.0*13-845.0/4104.0*14,x3+493.0/216.0*r1-8.0*r2+3680.0/513.0*r3-845.0/4104.0*r4); r5=h*f33(x0+h, y10,y20,y30, x1+493.0/216.0*k1-8.0*k2+3680.0/513.0*k3-845.0/4104.0*k4,x2+493.0/216.0*11-8.0*12+3680.0/513.0*13-845.0/4104.0*14,x3+493.0/216.0*r1-8.0*r2+3680.0/513.0*r3-845.0/4104.0*r4);

```
k6=h*f11(x0+0.5*h, y10,y20,y30, x1-8.0/27.0*k1+2.0*k2-
3544.0/2565.0*k3+1859.0/4104.0*k4-11.0/40.0*k5,x2-8.0/27.0*11+2.0*12-
3544.0/2565.0*13+1859.0/4104.0*14-11.0/40.0*15,x3-8.0/27.0*r1+2.0*r2-
3544.0/2565.0*r3+1859.0/4104.0*r4-11.0/40.0*r5);
      16=h*f22(x0+0.5*h, y10,y20,y30, x1-8.0/27.0*k1+2.0*k2-
3544.0/2565.0*k3+1859.0/4104.0*k4-11.0/40.0*k5,x2-8.0/27.0*11+2.0*12-
3544.0/2565.0*13+1859.0/4104.0*14-11.0/40.0*15,x3-8.0/27.0*r1+2.0*r2-
3544.0/2565.0*r3+1859.0/4104.0*r4-11.0/40.0*r5);
      r6=h*f33(x0+0.5*h,
                          y10,y20,y30, x1-8.0/27.0*k1+2.0*k2-
3544.0/2565.0*k3+1859.0/4104.0*k4-11.0/40.0*k5,x2-8.0/27.0*11+2.0*12-
3544.0/2565.0*13+1859.0/4104.0*14-11.0/40.0*15,x3-8.0/27.0*r1+2.0*r2-
3544.0/2565.0*r3+1859.0/4104.0*r4-11.0/40.0*r5);
      xbar[1]=x1+(25.0/216.0*k1+1408.0/2565.0*k3+2197.0/4104.0*k4-
0.2*k5);
      xbar[2]=x2+(25.0/216.0*11+1408.0/2565.0*13+2197.0/4104.0*14-
0.2*15);
      xbar[3]=x3+(25.0/216.0*r1+1408.0/2565.0*r3+2197.0/4104.0*r4-
0.2*r5);
      xout[1]=x1+(16.0/135.0*k1+6656.0/12825.0*k3+28561.0/56430*k4-
9.0/50.0*k5+2.0/55.0*k6);
      xout[2]=x2+(16.0/135.0*11+6656.0/12825.0*13+28561.0/56430*14-
9.0/50.0*15+2.0/55.0*16);
      xout[3]=x3+(16.0/135.0*r1+6656.0/12825.0*r3+28561.0/56430*r4-
9.0/50.0*r5+2.0/55.0*r6);
            }
      if(fabs(xbar[1]-xout[1])>(1.0/(8.0))*h)
      {
            h=0.5*h;jj=jj+1;
            cout<<h<<"\t"<<jj<<endl;</pre>
      }
}while (fabs(xbar[1]-xout[1])>1.0/8.0*h);
      t=t+h;
      xout[0]=t;
      for(int i=0;i<5;i++) {</pre>
            dxin[i]=xout[i];
} }
runge kutta fehberg.cpp
#include "stdafx.h"
#include "math.h"
#include "runge kutta_fehberg.h"
#include "dynsystem.h"
using namespace std;
extern double m[13];
void runge kutta fehlberg(double xin[4], double dx) {
```

```
double t=0.0;
double h=dx;
double xbar[4];
```

```
double xout[4];
double x0,x1,x2,x3;
double k1,k2,k3,k4,k5,k6,l1,l2,l3,l4,l5,l6,r1,r2,r3,r4,r5,r6;
```

// do{

//αυτός ο βρόχος χρειάζεται στην αναζήτηση του εκθέτη Lyapunov για να αποφευχθεί ή πιθανότητα να έχει αλλάξει το βήμα στην ολοκλήρωση της πρώτης τροχιάς, να μην αλλάξει στη δεύτερη ολοκλήρωση και να συγκριθούν αποτελέσματα που αντιστοιχούν σε διαφορετικούς χρόνους

```
do{
  {
    {
      t=xin[0];
      x0=xin[0];x1=xin[1];x2=xin[2];x3=xin[3];//x0,x1,x2,x3
      k1=h*f1(x0,x1,x2,x3);
      l1_h+f2(x0,x1,x2,x3);
       l1_h+f2(x0,x1,x2,x3);
      l1_h+f2(x0,x1,x2,x3);
      l1_h+f2(x0,x1,x2,x3);
      l1_h+f2(x0,x1,x2,x3);
      l1_h+f2(x0,x1,x2,x3);
      l1_h+f2(x0,x1,x2,x3);
      l1_h+f2(x0,x1,x2,x3);
      l1_h+f2(x0,x1,x2,x3);
      l1_h+f2(x0,x1,x2,x3);
      l1_h+f2(x0,x1,x2,x3);
      l1_h+f2(x0,x1,x2,x3);
      l1_h+f2(x0,x1,x2,x3);
      l1_h+f2(x0,x1,x2,x3);
      l1_h+f2(x0,x1,x2,x3);
      l1_h+f2(x0,x1,x2,x3);
      l1_h+f2(x0,x1,x2,x3);
      l1_h+f2(x0,x1,x2,x3);
      l1_h+f2(x0,x1,x2,x3);
      l1_h+f2(x0,x1,x2,x3);
      l1_h+f2(x0,x1,x2,x3);
      l1_h+f2(x0,x1,x2,x3);
      l1_h+f2(x0,x1,x2,x3);
      l1_h+f2(x0,x1,x2,x3);
      l1_h+f2(x0,x1,x2,x3);
      l1_h+f2(x0,x1,x2,x3);
      l1_h+f2(x0,x1,x2,x3);
      l1_h+f2(x0,x1,x2,x3);
      l1_h+f2(x0,x1,x2,x3);
      l1_h+f2(x0,x1,x2,x3);
      l1_h+f2(x0,x1,x2,x3);
      l1_h+f2(x0,x1,x2,x3);
      l1_h+f2(x0,x1,x2,x3);
      l1_h+f2(x0,x1,x2,x3);
      l1_h+f2(x0,x1,x2,x3);
      l1_h+f2(x0,x1,x2,x3);
      l1_h+f2(x0,x1,x2,x3);
      l1_h+f2(x0,x
```

l1=h*f2(x0,x1,x2,x3); r1=h*f3(x0,x1,x2,x3);

k2=h*f1(x0+0.25*h,x1+0.25*k1,x2+0.25*l1,x3+0.25*r1); l2=h*f2(x0+0.25*h,x1+0.25*k1,x2+0.25*l1,x3+0.25*r1); r2=h*f3(x0+0.25*h,x1+0.25*k1,x2+0.25*l1,x3+0.25*r1);

k3=h*f1(x0+3.0*h/8.0,x1+3.0*k1/32.0+9.0*k2/32.0,x2+3.0*11/32.0+ 9.0*12/32.0,x3+3.0*r1/32.0+9.0*r2/32.0); 13=h*f2(x0+3.0*h/8.0,x1+3.0*k1/32.0+9.0*k2/32.0,x2+3.0*11/32.0+ 9.0*12/32.0,x3+3.0*r1/32.0+9.0*r2/32.0);

```
r3=h*f3(x0+3.0*h/8.0,x1+3.0*k1/32.0+9.0*k2/32.0,x2+3.0*l1/32.0+
9.0*l2/32.0,x3+3.0*r1/32.0+9.0*r2/32.0);
```

```
k4=h*f1(x0+12.0*h/13.0,x1+1932.0/2197.0*k1-
7200.0/2197.0*k2+7296.0/2197.0*k3,x2+1932.0/2197.0*11-
7200.0/2197.0*12+7296.0/2197.0*13,x3+1932.0/2197.0*r1-
7200.0/2197.0*r2+7296.0/2197.0*r3);
14=h*f2(x0+12.0*h/13.0,x1+1932.0/2197.0*k1-
7200.0/2197.0*k2+7296.0/2197.0*k3,x2+1932.0/2197.0*11-
7200.0/2197.0*12+7296.0/2197.0*13,x3+1932.0/2197.0*r1-
7200.0/2197.0*r2+7296.0/2197.0*r3);
r4=h*f3(x0+12.0*h/13.0,x1+1932.0/2197.0*k1-
7200.0/2197.0*k2+7296.0/2197.0*k3,x2+1932.0/2197.0*k1-
7200.0/2197.0*k2+7296.0/2197.0*k3,x2+1932.0/2197.0*k1-
7200.0/2197.0*k2+7296.0/2197.0*k3,x2+1932.0/2197.0*11-
7200.0/2197.0*k2+7296.0/2197.0*k3,x2+1932.0/2197.0*11-
7200.0/2197.0*12+7296.0/2197.0*k3,x2+1932.0/2197.0*11-
```

```
7200.0/2197.0*r2+7296.0/2197.0*r3);
```

k5=h*f1(x0+h,x1+493.0/216.0*k1-8.0*k2+3680.0/513.0*k3-845.0/4104.0*k4,x2+493.0/216.0*l1-8.0*l2+3680.0/513.0*l3-845.0/4104.0*l4,x3+493.0/216.0*r1-8.0*r2+3680.0/513.0*r3-845.0/4104.0*r4); 15=h*f2(x0+h,x1+493.0/216.0*k1-8.0*k2+3680.0/513.0*k3-845.0/4104.0*k4,x2+493.0/216.0*l1-8.0*l2+3680.0/513.0*l3-

845.0/4104.0*14,x3+493.0/216.0*r1-8.0*r2+3680.0/513.0*r3-845.0/4104.0*r4); r5=h*f3(x0+h,x1+493.0/216.0*k1-8.0*k2+3680.0/513.0*k3-845.0/4104.0*k4,x2+493.0/216.0*11-8.0*12+3680.0/513.0*13-845.0/4104.0*14,x3+493.0/216.0*r1-8.0*r2+3680.0/513.0*r3-845.0/4104.0*r4);

```
k6=h*f1(x0+0.5*h,x1-8.0/27.0*k1+2.0*k2-
3544.0/2565.0*k3+1859.0/4104.0*k4-11.0/40.0*k5,x2-8.0/27.0*11+2.0*12-
3544.0/2565.0*13+1859.0/4104.0*14-11.0/40.0*15,x3-8.0/27.0*r1+2.0*r2-
3544.0/2565.0*r3+1859.0/4104.0*r4-11.0/40.0*r5);
      16=h*f2(x0+0.5*h,x1-8.0/27.0*k1+2.0*k2-
3544.0/2565.0*k3+1859.0/4104.0*k4-11.0/40.0*k5,x2-8.0/27.0*l1+2.0*l2-
3544.0/2565.0*13+1859.0/4104.0*14-11.0/40.0*15,x3-8.0/27.0*r1+2.0*r2-
3544.0/2565.0*r3+1859.0/4104.0*r4-11.0/40.0*r5);
      r6=h*f3(x0+0.5*h,x1-8.0/27.0*k1+2.0*k2-
3544.0/2565.0*k3+1859.0/4104.0*k4-11.0/40.0*k5,x2-8.0/27.0*11+2.0*12-
3544.0/2565.0*13+1859.0/4104.0*14-11.0/40.0*15,x3-8.0/27.0*r1+2.0*r2-
3544.0/2565.0*r3+1859.0/4104.0*r4-11.0/40.0*r5);
      xbar[1]=x1+(25.0/216.0*k1+1408.0/2565.0*k3+2197.0/4104.0*k4-
0.2*k5);
      xbar[2]=x2+(25.0/216.0*11+1408.0/2565.0*13+2197.0/4104.0*14-
0.2*15);
      xbar[3]=x3+(25.0/216.0*r1+1408.0/2565.0*r3+2197.0/4104.0*r4-
0.2*r5);
      xout[1]=x1+(16.0/135.0*k1+6656.0/12825.0*k3+28561.0/56430*k4-
9.0/50.0*k5+2.0/55.0*k6);
      xout[2]=x2+(16.0/135.0*11+6656.0/12825.0*13+28561.0/56430*14-
9.0/50.0*15+2.0/55.0*16);
      xout[3]=x3+(16.0/135.0*r1+6656.0/12825.0*r3+28561.0/56430*r4-
9.0/50.0*r5+2.0/55.0*r6);
            }
      double jj=0;
      if (fabs (xbar[1]-xout[1])>(1.0/(128.0))*h)
      {
            h=0.5*h;jj=jj+1;
            cout<<h<<"\t"<<jj<<endl;</pre>
      }while (fabs(xbar[1]-xout[1])>1.0/128.0*h);
            t=t+h;
            xout[0]=t;
      //}while ( t-dx < 0 );
      for(int i=0;i<5;i++) {</pre>
            xin[i]=xout[i];
      }
}
```

```
sustimagram.cpp
```

```
#include "stdafx.h"
#include "sustimagram.h"
#include "math.h"
#include "initial.h"

extern double m[13];
//gramiko systima efaptomeniki apoikonisi
double f11(double z0,double xx1,double xx2,double xx3,double
z1,double z2,double z3)
{
    return( (m[1]*(1.0-2.0*xx1)+m[2]*(1.0-xx2)+m[3]*(1.0-xx3))*z1-
m[2]*xx1*z2-m[3]*xx1*z3 );
}
```

```
double f22(double z0,double xx1,double xx2,double xx3,double
z1,double z2,double z3)
{
    return( -m[4]*xx2*z1+(m[4]*(1.0-xx1)+m[5]*(1.0-
2.0*xx2)+m[6]*(1.0-xx3))*z2-m[6]*xx2*z3);
}
double f33(double z0,double xx1,double xx2,double xx3,double
z1,double z2,double z3)
{
    return( -m[7]*xx3*z1-m[8]*xx3*z2+(m[7]*(1.0-xx1)+m[8]*(1.0-
xx2)+m[9]*(1.0-2.0*xx3))*z3);
}
```

```
liap volter.cpp//Εκθέτης Lyapunov με ολοκλήρωση
γειτονικών τροχιών
#include "stdafx.h"
#include "math.h"
#include "initial.h"
#include "runge kutta fehberg.h"
using namespace std;
extern double m[13];
void main(void)
{
      double m1=1.2;
      FILE *fill=fopen("graf.dat","wt");
      initial(0.5,0.5,0.1,-0.5,-0.10,0.1,m1,0.1,0.1,0.0,0.0,0.0);
      double x1[4], dx1[4];
      double liap=0, jn, jnbefore, an=0, xbefore;
      double tmax=100000, h=0.01;
      x1[0]=0; //initial condition
      x1[1]=0.452796;
      x1[2]=0.199;
      x1[3]=1.2;
      double dxdx1=sqrt(3.0);
      dx1[0]=0;
      dx1[1]=x1[1]+h/dxdx1;//ετσι ώστε η απόσταση μεταξύ x1 και dx1
να είναι h*h
      dx1[2]=x1[2]+h/dxdx1;
      dx1[3]=x1[3]+h/dxdx1;
      int i=0;
      double n1=0;
      while (x1[0]<tmax) {</pre>
            n1=n1+1;
           i=i+1;
jnbefore=sqrt((x1[1]-dx1[1])*(x1[1]-dx1[1])+(x1[2]-dx1[2])*(x1[2]-
dx1[2])+(x1[3]-dx1[3])*(x1[3]-dx1[3]));
            runge kutta fehlberg(x1,h);
            runge_kutta_fehlberg(dx1,h);
```

```
jn=sqrt((x1[1]-dx1[1])*(x1[1]-dx1[1])+(x1[2]-dx1[2])*(x1[2]-
dx1[2])+(x1[3]-dx1[3])*(x1[3]-dx1[3]));
                                                        liap=liap+log(jn/jnbefore);
                                                        if(i==50){
                                                                                    i=0;
                                                                                    fprintf(fil1,"% f % f \n",x1[0],liap/(x1[0]));
                                                         }
                                                       xbefore=dx1[1];
                   dx1[1]=x1[1]+h*h/sqrt(1.0+((dx1[2]-x1[2])*(dx1[2]-x1[2])+(dx1[3]-x1[2])+(dx1[3]-x1[2])+(dx1[3]-x1[2])+(dx1[3]-x1[2])+(dx1[3]-x1[3])+(dx1[3]-x1[3])+(dx1[3]-x1[3])+(dx1[3]-x1[3])+(dx1[3]-x1[3])+(dx1[3]-x1[3])+(dx1[3]-x1[3])+(dx1[3]-x1[3])+(dx1[3]-x1[3])+(dx1[3]-x1[3])+(dx1[3]-x1[3])+(dx1[3]-x1[3])+(dx1[3]-x1[3])+(dx1[3]-x1[3])+(dx1[3]-x1[3])+(dx1[3]-x1[3])+(dx1[3]-x1[3])+(dx1[3]-x1[3])+(dx1[3]-x1[3])+(dx1[3]-x1[3])+(dx1[3]-x1[3])+(dx1[3]-x1[3])+(dx1[3]-x1[3])+(dx1[3]-x1[3])+(dx1[3]-x1[3])+(dx1[3]-x1[3])+(dx1[3]-x1[3])+(dx1[3]-x1[3])+(dx1[3]-x1[3])+(dx1[3]-x1[3])+(dx1[3]-x1[3])+(dx1[3]-x1[3])+(dx1[3]-x1[3])+(dx1[3]-x1[3])+(dx1[3]-x1[3])+(dx1[3]-x1[3])+(dx1[3]-x1[3])+(dx1[3]-x1[3])+(dx1[3]-x1[3])+(dx1[3]-x1[3])+(dx1[3]-x1[3])+(dx1[3]-x1[3])+(dx1[3]-x1[3])+(dx1[3]-x1[3])+(dx1[3]-x1[3])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(dx1[x])+(
                                                             x1[3])*(dx1[3]-x1[3]))/((dx1[1]-x1[1])*(dx1[1]-x1[1])));
                   dx1[2]=x1[2]+(dx1[2]-x1[2])*(dx1[1]-x1[1])/((xbefore-x1[1]));
                                    dx1[3]=x1[3]+(dx1[3]-x1[3])*(dx1[1]-x1[1])/((xbefore-x1[1]));
                            }
                            cout<<"gia xrono tmax= "<<x1[0]<<" exoume liap=";</pre>
                            cout<<liap/(n1*h)<<endl;</pre>
                            fclose(fil1);
 }
```

box coun1.cpp//Διάσταση χωρητικότητας της τομής Poincare

```
#include "stdafx.h"
#include "math.h"
#include <iostream>
#include <fstream>
using namespace std;
void main(void)
{
      ifstream input file("graf1.dat");
      double Dc=0;
      int N=90300,N1;
      double L=1.0;
      double l=L;
      double xstart=0.0, xend=xstart+L;
      double ystart=0.5, yend=ystart+L;
      double*x;x=new double[N];
      double*y;y=new double[N];
      for(int i=0;i<N;i++) {</pre>
            input file>>x[i];
            input file>>y[i];
            if((x[i]>100000)||(x[i]<-1000000))
            {
                   N1=i-1;
                  break;
            }
      }
      cout<<x[N1]<<" "<<y[N1]<<endl;
      int n=1;
```

```
int counter=0;
      int diktis=0;
      do {
      for(int jx=1;jx<=n;jx++) {</pre>
            for(int jy=1;jy<=n;jy++) {</pre>
                   for(int i=0;i<=N1;i++)</pre>
                   {
                          if((x[i]<xstart+(jx)*l)&&(x[i]>=xstart+(jx-
1) *1) && (y[i] <ystart+(jy) *1) && (y[i] >=ystart+(jy-1) *1) )
                          {
                                counter=counter+1;
                                diktis=1;
                                break;
                          }
                          else diktis=0;
                  }
             }
      }
      Dc=log(double(counter))/log(L/l);
      cout<<n<<" "<<1/L<<" "<<counter<<" Dc= "<<Dc<<endl;
      1=1*0.5;
      n=2*n;
      counter=0;
      }while(1/L>0.0005);
      delete
                   []x;
      delete
                   []y;
}
box coun2.cpp //Διάσταση χωρητικότητας της τομής Poincare
#include "stdafx.h"
#include "math.h"
#include <iostream>
#include <fstream>
using namespace std;
void main(void)
{
      FILE *fill=fopen("graf.dat","wt");
      ifstream input_file("grafm155.dat");
      double Dc=0;
      int N=90300, M=9000, N1;
      double L=1.0;
      double l=L/4.0;
      double xstart=0.0, xend=xstart+L;
      double ystart=0.5, yend=ystart+L;
      double mindok,min1=sqrt(2*(L)*(L));
      double*x;
      double*y;
      x=new double[N];
      y=new double[N];
      int*xx;
```

```
xx=new int[M*M];
      for(int i=0;i<N;i++) {</pre>
             input file>>x[i]>>y[i];
             if((x[i]>100000)||(x[i]<-1000000))
             {
                   N1=i-1;
                   break;
             }
      }
cout<<" min= "<<min1<<endl;</pre>
      int n=4, iix, iiy;
      int counter=0;
      int diktis=0;
      do {
             for(int i=0;i<M;i++) {</pre>
                   for(int j=0;j<M;j++) {</pre>
                          xx[i*M+j]=0;
                    }
             }
                   for(int i=0;i<=N1;i++)</pre>
                    {
                          iix=int((x[i]-xstart)/l);
                          iiy=int((y[i]-ystart)/l);
                          xx[iix*n+iiy]=xx[iix*n+iiy]+1;
                    }
      for(int i=0;i<M;i++) {</pre>
             for(int j=0;j<M;j++) {</pre>
      if((xx[i*M+j]>0)&&(!(xx[i*M+j]>100000)||(xx[i*M+j]<-1000000)))
counter=counter+1;
             }
      }
      Dc=log(double(counter))/log(L/l);
      cout<<n<<" "<<l/L<<" "<<counter<<" Dc= "<<Dc<<endl;</pre>
      fprintf(fil1,"% i % f \n",n,Dc);
      1=1*0.5;
      n=2*n;
      counter=0;
      }while(1/L>0.0001);
      delete
                   []x;
      delete
                   []y;
      delete
                  []xx;
      fclose(fil1);
}
```

```
box coun3d.cpp // Διάσταση χωρητικότητας του ελκυστή
#include "stdafx.h"
#include "math.h"
#include <iostream>
#include <fstream>
using namespace std;
void main(void)
{
      FILE *fill=fopen("graf1.dat", "wt");
      ifstream input file("graf.dat");
      double Dc=0;
      int N=90300, M=1000, N1;
      double*tt1;
      double*x;
      double*y;
      double*z;
      ttl=new double[N];
      x=new double[N];
      y=new double[N];
      z=new double[N];
      bool*xx;
      xx=new bool[M*M*M];
      for(int i=0;i<N;i++) {</pre>
             input file>>tt1[i];
             input file>>x[i]>>y[i]>>z[i];
             if((x[i]>100000)||(x[i]<-1000000))
             {
                   N1=i-1;
                   break;
             }
             else N1=N-1;
      }
      double xmin=x[0], xmax=x[0];
      double ymin=y[0],ymax=y[0];
      double zmin=z[0], zmax=z[0];
      for(int i=0;i<N1;i++) {</pre>
             if (x[i]<xmin) xmin=x[i];</pre>
             if (x[i]>xmax) xmax=x[i];
          if (y[i]<ymin) ymin=y[i];</pre>
             if (y[i]>ymax) ymax=y[i];
             if (z[i]<zmin) zmin=z[i];</pre>
             if (z[i]>zmax) zmax=z[i];
      }
      double xstart=xmin-0.001, xend=xstart+xmax+0.0011;
```

```
double ystart=ymin-0.001, yend=ystart+ymax+0.0011;
      double zstart=zmin-0.001,zend=zstart+zmax+0.0011;
      double L=max(xend-xstart, yend-ystart);
      L=max(L, zend-zstart);
      double l=L;
      cout<<xend-xstart<<" "<<yend-ystart<<" "<<zend-zstart<<"</pre>
"<<L<<endl;
      double mindok,min1=sqrt(3*(L)*(L));
      int n=1, iix, iiy, iiz;
      int counter=0;
      int diktis=0;
      1=1*0.5;
      n=2*n;
      do {
             for(int i=0;i<M;i++) {</pre>
                   for(int j=0;j<M;j++) {</pre>
                          for(int k=0; k<M; k++) {</pre>
                                 xx[i*M+j*M+k]=0;
                          }
                   }
             }
             for(int i=0;i<=N1;i++)</pre>
             {
                   iix=int((x[i]-xstart)/l);
                   iiy=int((y[i]-ystart)/l);
                   iiz=int((z[i]-zstart)/l);
                   xx[iix*n+iiy*n+iiz]=1;
             }
             for(int i=0;i<M;i++) {</pre>
                    for(int j=0;j<M;j++) {</pre>
                          for (int k=0; k<M; k++) {</pre>
      if((xx[i*M+j*M+k]==0)&&(!(xx[i*M+j*M+k]>100000)||(xx[i*M+j*M+k]
<-1000000))) counter=counter+1;
                          }
                    }
             }
             Dc=log(double(counter))/log(L/l);
                fprintf(fil1,"% i % f \n",n,Dc);
             1=1*0.5;
             n=2*n;
             counter=0;
      }while(1/L>0.0001);
```

```
delete []tt1;
delete []x;
delete []y;
delete []z;
delete []xx;
fclose(fill);
}
```

diastash.cpp //Διάσταση εμβύθισης

```
#include "stdafx.h"
#include "math.h"
#include <iostream>
#include <fstream>
using namespace std;
void main(void)
{
      ifstream input file("grafz.dat");
      ofstream file graf("aytosis.dat");
      ofstream file_grafanakataskeyh3("anakataskeyh3.dat");
      int N=4000000,N1,Nmax,M,MMax=5;
      N=10000;
      N1=N-1;
      double 1;
      double xstart, xend;
      double ystart=0.0,yend=ystart;
      double*x;
      x=new double[N];
      double*xhelp;
      xhelp=new double[N];
      int kk=1000;
      double*rk;
      rk=new double[kk];
      for(int i=0;i<N;i++) {</pre>
             input file>>x[i];
             if((x[i]>100000)||(x[i]<-100000))
             {
                   N1=i-1;
                   break;
             }
      }
      double minxi=x[0], maxxi=x[0];
      for(int i=1;i<=N1;i++) {</pre>
             if(x[i] < minxi) minxi=x[i];</pre>
            if(x[i] > maxxi) maxxi=x[i];
      }
      xstart=minxi-0.00001;
      xend=maxxi+0.00001;
```

```
l=abs(xend-xstart);
//αυτοσυσχέτιση
  double mesos oros=0;
  for(int i=0;i<=N1;i++) {</pre>
        mesos oros+=x[i];
  }
  mesos oros=mesos oros/double(N1);
  double diaspora, sindiaspora;
  for(int k=0; k<kk; k++) {</pre>
        diaspora=0;
        sindiaspora=0;
        for(int i=1;i<=N1;i++) {</pre>
               diaspora+=(x[i]-mesos_oros)*(x[i]-mesos_oros);
         }
        for(int i=1;i<=N1-k;i++) {</pre>
               sindiaspora+=(x[i]-mesos oros)*(x[i+k]-mesos oros);
         }
        rk[k]=sindiaspora/diaspora;
        file graf<<k<<"\t"<<rk[k]<<endl;</pre>
  }
  int tt;
  for(int i=1;i<kk;i++) {</pre>
        if( (rk[i]<=0) && (rk[i-1]>0) )
         {
               tt=i;//υστεριση τ
               break;
        }
  }
  for(int i=0;i<=N1;i++)</pre>
  {
        xhelp[N1-i]=x[i];
  }
        cout<<" N1= "<<N1<<endl;</pre>
  //διάσταση εμβύθισης
  double*xx;
  int*xxn;
  double*NearestNeighbors;
  NearestNeighbors=new double[MMax+1];
  double sumd, distanse=1, mindistanse, distanseplus, sumdnear;
  double count=0;
  int nearpoint;
  xx=new double[N1];
  xxn=new int[N1];
  for(int m=1;m<=MMax;m++) {</pre>
```

```
M=m;
             int iii=0;
             for(int i=0;i<=N1;i++) {</pre>
                   for(int j=0;j<M;j++) {</pre>
                          xx[i*M+j]=xhelp[iii];
                          xxn[i*M+j]=0;
                          iii=iii+1;
                          if(iii==N1) break;
                   }
                   if(iii==N1)
                   {
                          Nmax=i;
                          break;
                   }
             1
             Nmax=N1-(M-1)*tt;
             count=0;
             for(int i=0;i<=Nmax;i++) {</pre>
                   mindistanse=1;
                          for(int ii1=0;ii1<=Nmax;ii1++) {</pre>
                                sumd=0;
                                for(int jj1=0;jj1<M;jj1++) {</pre>
      sumd=sumd+(xx[i+jj1*tt]-xx[ii1+jj1*tt])*(xx[i+jj1*tt]-
xx[ii1+jj1*tt]);
                                 }
                                distanse=sqrt(sumd);
                                if( (distanse<mindistanse) && ( i!=ii1</pre>
)) {
                                       mindistanse=distanse;
                                       nearpoint=ii1;
                                       sumdnear=sumd;
                                }
                          }
xxn[nearpoint]=i;
sumd=sumdnear+(xx[i+(M)*tt]-xx[nearpoint+(M)*tt])*(xx[i+(M)*tt]-
xx[nearpoint+(M)*tt]);
                                //sumdnear=0;
distanseplus=sqrt(sumd);
if( ( distanseplus/mindistanse )>100.0*double(M)/double(M+1) )
count=count+1;
}
             NearestNeighbors[m]=count;
             if(NearestNeighbors[m]==0)
             {
```

```
cout<<endl;</pre>
                   cout<<"
                               break;
                                            "<<endl;
                   break;
             }
      for(int i=0;i<=Nmax-3*tt;i++) {</pre>
      file grafanakataskeyh3<<xx[i]<<"\t"<<xx[i+tt]<<"\t"<<xx[i+2*tt]</pre>
<<endl;
      }
      delete
                  []x;
      delete
                   []xhelp;
      delete
                   []xx;
      delete
                   []xxn;
      delete
                   []rk;
      delete
                   []NearestNeighbors;
```

```
}
```

perioxi elxis.cpp //Λεκάνη έλξης της περιοδικής τροχιάς

```
#include "stdafx.h"
#include "math.h"
#include "initial.h"
#include "runge kutta fehberg.h"
#include "run2.h"
#include "time.h"
#include <fstream>
using namespace std;
extern double m[13];
extern int *dik;
void main(void) {
      time t system time;
      ofstream file_graf("graf1.dat");
      ofstream file_grafelxi("grafelxi.dat");
      ofstream file_grafelxi11("grafelxi11.dat");
      ofstream file_grafelxi22("grafelxi22.dat");
      ofstream file grafelxiolo("grafelxiolo.dat");
      ofstream file grafelxiexoterika("grafelxiexoterika.dat");
       initial(0.5,0.5,0.1,-0.5,-0.1,0.1,1.2,0.1,0.1,0.0,0.0,0.0);
      double x1[4],x2[4];
      double tmax=4000.5, h=0.05;
      x1[0]=0; //initial condition
      x1[1]=0.652518;
      x1[2]=1.33328;
      x1[3]=1.0;
      double dd, dxp=0.8, dyp=0.015, dminxy=-5.0, dmaxy=5, dmaxx=50.0;
      double a0,ax,ay,az;
      int dik=0, c1=0, c2=0;
```

```
system time=time(NULL);
      x2[0]=0;
      x2[1]=dminxy;
      x2[2]=dminxy;
      x2[3]=1.0;
      x1[0]=0;
      x1[1]=x2[1];
      x1[2]=x2[2];
      x1[3]=x2[3];
      for(double iz=-0.3;iz<65;iz=iz+1.26) {</pre>
            x2[1]=dminxy;
            for(int i=1;i<(dmaxx-dminxy)/dxp;i++) {</pre>
                 system time=time(NULL);
                 x2[1]=x2[1]+dxp;
                 x1[1]=x2[1];
                 x2[2]=dminxy;
                 x1[2]=x2[2];
                   for(int j=1;j<(dmaxy-dminxy)/dyp;j++) {</pre>
                       x2[0]=0;
                       x1[0]=x2[0];
                       x1[1]=x2[1];
                       x2[2]=x2[2]+dyp;
                       x1[2]=x2[2];
                       x2[3]=iz;
                       x1[3]=x2[3];
                       dd=0;
                   runge_kutta_fehlberg(x1,20000.0*h,h);
                         a0=x1[0];
                         ax=x1[1];
                         ay=x1[2];
                         az=x1[3];
                          x1[0]=0;
                   run2(x1,20000.0*h,h,&dik);
dd=sqrt( (1.0-x1[1])*(1.0-x1[1])+(1.0-x1[2])*(1.0-x1[2])+(1.0-
x1[3])*(1.0-x1[3]) );
                     if( dik==1 ) {
      file grafelxi<<dd<<"\t"<<x2[1]<<"\t"<<x2[2]<<"\t"<<x2[3]<<endl;
                       }
                   }
               }
        }
   }
```

```
run2.cpp
```

```
#include "stdafx.h"
#include "run2.h"
#include "math.h"
#include "dynsystem.h"
using namespace std;
extern double m[13];
void run2(double xin[4],double t1,double dx,int *dik){
      double t=0.0;
      double h;
      double xbar[4];
      double xout[4];
      double dd=0;
      double x0, x1, x2, x3;
      double k1, k2, k3, k4, k5, k6, l1, l2, l3, l4, l5, l6, r1, r2, r3, r4, r5, r6;
      double tmax=t1;
      int ii=0,ij=0;
      double dmin=10.0, dmax=0.0;
      while (xin[0]<tmax) {</pre>
            h=dx;
            ii=ii+1;
            ij=ij+1;
      double jj=0;
      do {
            t=xin[0];
           x0=xin[0];x1=xin[1];x2=xin[2];x3=xin[3];//x0,x1,x2,x3
           k1=h*f1(x0,x1,x2,x3);
           l1=h*f2(x0,x1,x2,x3);
           r1=h*f3(x0,x1,x2,x3);
           k2=h*f1(x0+0.25*h,x1+0.25*k1,x2+0.25*11,x3+0.25*r1);
           l2=h*f2(x0+0.25*h,x1+0.25*k1,x2+0.25*l1,x3+0.25*r1);
           r2=h*f3(x0+0.25*h,x1+0.25*k1,x2+0.25*l1,x3+0.25*r1);
k3=h*f1(x0+3.0*h/8.0,x1+3.0*k1/32.0+9.0*k2/32.0,x2+3.0*l1/32.0
                 +9.0*12/32.0,x3+3.0*r1/32.0+9.0*r2/32.0);
13=h*f2(x0+3.0*h/8.0,x1+3.0*k1/32.0+9.0*k2/32.0,x2+3.0*11/32.0
                 +9.0*12/32.0,x3+3.0*r1/32.0+9.0*r2/32.0);
r3=h*f3(x0+3.0*h/8.0,x1+3.0*k1/32.0+9.0*k2/32.0,x2+3.0*11/32.0
                 +9.0*12/32.0,x3+3.0*r1/32.0+9.0*r2/32.0);
k4=h*f1(x0+12.0*h/13.0,x1+1932.0/2197.0*k17200.0/2197.0*k2
+7296.0/2197.0*k3,x2+1932.0/2197.0*117200.0/2197.0*12+7296.0/2197.0*1
3,x3
+1932.0/2197.0*r1-7200.0/2197.0*r2+7296.0/2197.0*r3);
            14=h*f2(x0+12.0*h/13.0,x1+1932.0/2197.0*k1-
7200.0/2197.0*k2
+7296.0/2197.0*k3,x2+1932.0/2197.0*117200.0/2197.0*12+7296.0/2197.0*1
3,x3
+1932.0/2197.0*r1-7200.0/2197.0*r2+7296.0/2197.0*r3);
            r4=h*f3(x0+12.0*h/13.0,x1+1932.0/2197.0*k1-
7200.0/2197.0*k2
```

```
+7296.0/2197.0*k3,x2+1932.0/2197.0*117200.0/2197.0*12+7296.0/2197.0*1
3,x3
+1932.0/2197.0*r1-7200.0/2197.0*r2+7296.0/2197.0*r3);
            k5=h*f1(x0+h,x1+493.0/216.0*k1-8.0*k2+3680.0/513.0*k3-
845.0/4104.0*k4,
x2+493.0/216.0*11-8.0*12+3680.0/513.0*13-
845.0/4104.0*14,x3+493.0/216.0*r1-8.0*r2+3680.0/513.0*r3-
845.0/4104.0*r4);
            15=h*f2(x0+h,x1+493.0/216.0*k1-8.0*k2+3680.0/513.0*k3-
845.0/4104.0*k4,
x2+493.0/216.0*11-8.0*12+3680.0/513.0*13-
845.0/4104.0*14,x3+493.0/216.0*r1-8.0*r2+3680.0/513.0*r3-
845.0/4104.0*r4);
            r5=h*f3(x0+h,x1+493.0/216.0*k1-8.0*k2+3680.0/513.0*k3-
845.0/4104.0*k4,
x2+493.0/216.0*11-8.0*12+3680.0/513.0*13-
845.0/4104.0*14,x3+493.0/216.0*r1-8.0*r2+3680.0/513.0*r3-
845.0/4104.0*r4);
      k6=h*f1(x0+0.5*h,x1-8.0/27.0*k1+2.0*k2-
3544.0/2565.0*k3+1859.0/4104.0*k4-11.0/40.0*k5,x2-8.0/27.0*11+2.0*12-
3544.0/2565.0*13+1859.0/4104.0*14-11.0/40.0*15,x3-8.0/27.0*r1+2.0*r2-
3544.0/2565.0*r3+1859.0/4104.0*r4-11.0/40.0*r5);
      16=h*f2(x0+0.5*h,x1-8.0/27.0*k1+2.0*k2-
3544.0/2565.0*k3+1859.0/4104.0*k4-11.0/40.0*k5,x2-8.0/27.0*11+2.0*12-
3544.0/2565.0*13+1859.0/4104.0*14-11.0/40.0*15,x3-8.0/27.0*r1+2.0*r2-
3544.0/2565.0*r3+1859.0/4104.0*r4-11.0/40.0*r5);
      r6=h*f3(x0+0.5*h,x1-8.0/27.0*k1+2.0*k2-
3544.0/2565.0*k3+1859.0/4104.0*k4-11.0/40.0*k5,x2-8.0/27.0*l1+2.0*l2-
3544.0/2565.0*13+1859.0/4104.0*14-11.0/40.0*15,x3-8.0/27.0*r1+2.0*r2-
3544.0/2565.0*r3+1859.0/4104.0*r4-11.0/40.0*r5);
      xbar[1]=x1+(25.0/216.0*k1+1408.0/2565.0*k3+2197.0/4104.0*k4-
0.2*k5);
      xbar[2]=x2+(25.0/216.0*11+1408.0/2565.0*13+2197.0/4104.0*14-
0.2*15);
      xbar[3]=x3+(25.0/216.0*r1+1408.0/2565.0*r3+2197.0/4104.0*r4-
0.2*r5);
      xout[1]=x1+(16.0/135.0*k1+6656.0/12825.0*k3+28561.0/56430*k4-
9.0/50.0*k5+2.0/55.0*k6);
      xout[2]=x2+(16.0/135.0*11+6656.0/12825.0*13+28561.0/56430*14-
9.0/50.0*15+2.0/55.0*16;
      xout[3]=x3+(16.0/135.0*r1+6656.0/12825.0*r3+28561.0/56430*r4-
9.0/50.0*r5+2.0/55.0*r6);
      if(fabs(xbar[1]-xout[1])>(1.0/(2.0))*h)
              h=0.5*h;jj=jj+1;
      {
            if (jj>10) break;
        1
      if(jj>2) cout<<h<<"\t"<<jj<<endl;</pre>
      dd=sqrt( (1.0-xout[1])*(1.0-xout[1])+(1.0-xout[2])*(1.0-
xout[2])+(1.0-xout[3])*(1.0-xout[3]) );
      if ( abs(xout[3]-14)<0.1 ) {
            *dik=0;
```

```
break;
      }
     if( (abs(xout[3]-0)<0.01) ) {
           *dik=0;
           break;
      }
     if( ( xout[3] < 1.1 )&&( xout[3] > 0.99)&&( xout[1] <
0.55846+0.1 )&&( xout[1] > 0.55846-0.1 )&&( xout[2] < 1.28234+0.1
)&&( xout[2] > 1.28234-0.1 ) )
{
           *dik=1;
           break;
      }
     if(dd>50){
           *dik=0;
           break;
      }
     }while (fabs(xbar[1]-xout[1])>h);
     t=t+h;
     xout[0]=t;
     for(int i=0;i<5;i++) {</pre>
           xin[i]=xout[i];
      }
     if( ( xout[3] < 1.1 )&&( xout[3] > 0.99)&&( xout[1] <
0.55846+0.1 )&&( xout[1] > 0.55846-0.1 )&&( xout[2] < 1.28234+0.1
)&&( xout[2] > 1.28234-0.1 ) ) {
           *dik=1;
           break;
     }
     if(dd>50){
           *dik=0;
           break;
      }
     if( abs(xout[3]-14)<0.1 ) {
           *dik=0;
           break;
      }
     if( (abs(xout[3]-0)<0.01)) {
           *dik=0;
           break;
      }
}
}
correlationdimension.cpp //Διάσταση συσχέτισης
```

```
#include "stdafx.h"
#include "math.h"
#include <iostream>
#include <fstream>
```

```
using namespace std;
void main(void)
{
      ifstream input file("grafz.dat");
      ofstream file graf1("graf1.dat");
      ofstream file graf2("graf2.dat");
      int N=4000000,N1,Nmax,M,Rmax=100,Mmax=10;
      N=10000;
      N1=N-1;
      double 1;
      double xstart, xend;
      double ystart=0.0, yend=ystart;
      double*x;
      x=new double[N];
      double*xhelp;
      xhelp=new double[N];
      int kk=1000;
      double*rk;
      rk=new double[kk];
      for(int i=0;i<N;i++) {</pre>
             input file>>x[i];
             if((x[i]>100000)||(x[i]<-100000))
             {
                   N1=i-1;
                   break;
             }
      }
      cout<<x[N1]<<" All point = "<<N1<<endl;</pre>
      double minxi=x[0],maxxi=x[0];
      for(int i=1;i<=N1;i++) {</pre>
             if(x[i] < minxi) minxi=x[i];</pre>
             if(x[i] > maxxi) maxxi=x[i];
      }
      xstart=minxi-0.00001;
      xend=maxxi+0.00001;
      l=abs(xend-xstart);
    //αυτοσυσχέτιση
      double mesos_oros=0;
      for(int i=0;i<=N1;i++) {</pre>
             mesos oros+=x[i];
      }
      mesos oros=mesos oros/double(N1);
       double diaspora, sindiaspora;
      for(int k=0; k<kk; k++) {</pre>
```

```
diaspora=0;
             sindiaspora=0;
             for(int i=1;i<=N1;i++) {</pre>
                    diaspora+=(x[i]-mesos oros)*(x[i]-mesos oros);
             for(int i=1;i<=N1-k;i++) {</pre>
                    sindiaspora+=(x[i]-mesos oros)*(x[i+k]-mesos oros);
             }
             rk[k]=sindiaspora/diaspora;
       } int tt;
      for(int i=1;i<kk;i++) {</pre>
             if( (rk[i]<=0) && (rk[i-1]>0) )
             {
                    tt=i;//υστεριση τ
                    break;
             }
       }
      for(int i=0;i<=N1;i++)</pre>
       {
             xhelp[N1-i]=x[i];
       }
             double*xx;
             xx=new double[N1+1];
             for(int i=0;i<=N1;i++) {</pre>
                           xx[i]=xhelp[i];
             }
             Nmax=N1;
      double*C;
      C=new double[(Rmax+1)*Mmax];
      double sumd, distanse;
      double count=0;
      double r=0.0001, rbef=0.0001, dr=0.025;
      for(int m=1;m<=Mmax;m++) {</pre>
             M=m;
             count=0;rbef=0; r=0;
             for(int R=1; R<=Rmax; R++) {</pre>
                    r=r+dr;
                    count=0;
                    for(int i=0;i<=Nmax;i++) {</pre>
                           for(int j=i+1;j<=Nmax;j++) {</pre>
                                  sumd=0;
                                  for(int jj1=1;jj1<=M;jj1++) {</pre>
             sumd=sumd+(xx[i+jj1*tt]-xx[j+jj1*tt])*(xx[i+jj1*tt]-
xx[j+jj1*tt]);
                                  }
                                  distanse=sqrt(sumd);
                                  if( r-distanse>0 ) count=count+1;
                           }
                    }
```

```
C[(m-1)*Rmax+R]=2.0/(double(Nmax)*(double(Nmax)-
1.0)) *double(count);
                   if(R>1)
                   {
file graf1<<log(r)<<"\t"<<(log(C[(m-1)*Rmax+R])-log(C[(m-1)*Rmax+R-</pre>
1]))/(log(r)- log(rbef))<<"\t"<<m<<endl;
file graf2<<log(r)<<"\t"<<(log(C[(m-1)*Rmax+R]))<<endl;</pre>
                   }
             rbef=r;
             }
      }
    cout<<" telos "<<endl;</pre>
      delete
                   []x;
      delete
                   []xhelp;
      delete
                   []rk;
      delete
                   []C;
      delete
                   []xx;
}
```

periodic Orbit.cpp //Αναζήτηση περιοδικών τροχιών

```
#include "stdafx.h"
#include <cmath>
#include "initial.h"
#include "section_poincare.h"
#include "DxDy.h"
using namespace std;
extern double m[13];
void main(void) {
      FILE *fill=fopen("graf.dat", "wt");
      double m1=1.38;
        initial(0.5,0.5,0.1,-0.5,-0.10,0.1,m1,0.1,0.1,0.0,0.0,0.0);
      double x1[4], xb[4], xxp[4]; //, xbefore[13], xb[1]
      double h0=0.01, dx0=h0*0.00000001, dy0=h0*0.00000001;
      double tmax=30;
      double dx=0,dy=0;
      x1[0]=0; //initial condition
      x1[1]=0.6;
      x1[2]=1.11;
      x1[3]=1;
      int i=1;
            do{
                  i=i+1;
                  x1[0]=x1[0];
                  x1[1]=x1[1]+dx;
                  x1[2]=x1[2]+dy;
```

x1[3]=x1[3]; xb[0]=x1[0]; xb[1]=x1[1]; xb[2]=x1[2]; xb[3]=x1[3]; sectionp(x1,h0); fprintf(fil1,"% f % f \n",x1[1],x1[2]); xxp[0]=x1[0]; xxp[1]=x1[1]; xxp[1]=x1[1]; xxp[2]=x1[2]; xxp[3]=x1[3]; DxDy(xb,xxp,h0,dx0,dy0,&dx,&dy); }while (abs(x1[1]-xb[1])+abs(x1[2]-xb[2])+abs(x1[3]-

xb[3])>0.000001);

fclose(fil1);

}

DxDy.cpp // επιστρέφει τις παραγώγους

```
#include "stdafx.h"
#include "DxDy.h"
using namespace std;
#include "section poincare.h"
double xxx[4];
double xxx1, xxx2, dxdx0, dxdy0;
double yyy1, yyy2, dydy0, dydx0;
double a;
void DxDy(double x0[4], double xout[4], double h0, double dx0, double
dy0, double *dx, double *dy) {
      {
            xxx[0]=x0[0];
            xxx[1]=x0[1]-dx0;
            xxx[2]=x0[2];
            xxx[3]=x0[3];
            sectionp(xxx,h0);
            xxx1=xxx[1];
            yyy1=xxx[2];
            xxx[0]=x0[0];
            xxx[1] = x0[1] + dx0;
            xxx[2]=x0[2];
            xxx[3]=x0[3];
            sectionp(xxx,h0);
```

```
xxx2=xxx[1];
      yyy2=xxx[2];
      dxdx0=(xxx2-xxx1)/(2*dx0);
      dydx0=(yyy2-yyy1)/(2*dx0);
}
{
      xxx[0]=x0[0];
      xxx[1]=x0[1];
      xxx[2] = x0[2] - dy0;
      xxx[3]=x0[3];
      sectionp(xxx,h0);
      xxx1=xxx[1];
      yyy1=xxx[2];
      xxx[0]=x0[0];
      xxx[1]=x0[1];
       xxx[2]=x0[2]+dy0;
       xxx[3]=x0[3];
       sectionp(xxx,h0);
         xxx2=xxx[1];
       yyy2=xxx[2];
       dxdy0=(xxx2-xxx1)/(2*dy0);
       dydy0=(yyy2-yyy1)/(2*dy0);
}
a = (dxdx0-1) * (dydy0-1) - (dxdy0*dydx0);
*dx=((dydy0-1)*(x0[1]-xout[1]) +(-dxdy0)*(x0[2]-xout[2]))/a;
*dy= ((-dydx0) * (x0[1]-xout[1]) + (dxdx0-1) * (x0[2]-xout[2]))/a; }
```

ΒΙΒΛΙΟΓΡΑΦΙΑ

[1] Κ. Κόκκοτας, Σημειώσεις Αριθμητικής Ανάλυσης ,Α.Π.Θ. τμήμα εκδόσεων ,2006

[2] Σημειώσεις Μελετλίδου Ευθυμία .Μη Γραμμική δυναμική.

[3] Σημειώσεις Προσομοίωση χαοτικών χρονοσειρών - Μη γραμμική ανάλυση χρονοσειρών, Κουγιουμτζής Δημήτριος.

[4] Σημειώσεις Προσομοίωση χαοτικών συστημάτων Βουγιατζής Γεώργιος

[5] Στάμου-Σγαρδέλης-Παντής, Πληθυσμιακή οικολογία. Α.Π.Θ. τμήμα εκδόσεων Πανεπιστημιακό τυπογραφείο 2001-2002

[6] Σ.Π.Σγαρδέλης Μαθηματικά μοντέλα στη βιολογία Α.Π.Θ., τμήμα εκδόσεων Πανεπιστημιακό τυπογραφείο 2004

[7] Σ.Ιχτιάρογλου-Ι.Χατζηδημητρίου , Δυναμικά συστήματα και χάος Α.Π.Θ. τμήμα εκδόσεων Πανεπιστημιακό τυπογραφείο 2000

[8] Τ.Μπούντης, Ο θαυμαστός κόσμος των FRACTAL ,Leader Books 2004.

[9] Σ. Τραχανάς, Mathematica και εφαρμογές, Πανεπιστημιακές Εκδόσεις Κρήτης 2001.

[10] Mark Kot, Elements of Mathematical Ecology, Cambridge University Press 2003.

[11] Lawrence Perco , Differential Equations and Dynamical Systems, Springer-verlag. 1991

[12] J.D.Maray, Mathematical Biology I. An Introduction Third Edition, Springer 2001.

[13] G.Nicolis, Introduction To NonLinear Science, Cambridge Univercity Press 1995.

[14] S.Wiggins ,Introduction to Applied Nonlinear Dynamical Systems and Chaos , Springer 1996.

Άρθρα

[15] A.Arneodo-P.Coullet-C.Tresser "occurrence of strange attractors in threedimensional volterra equations"