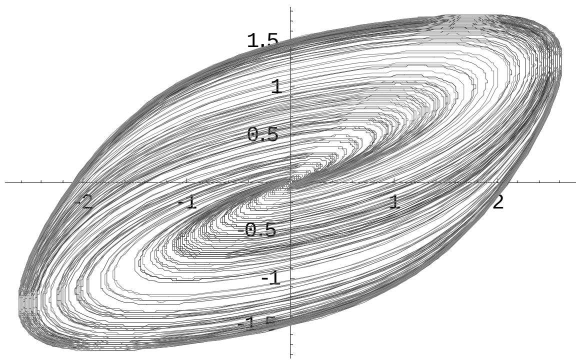
ΔΙΑΦΟΡΙΚΕΣ ΕΞΙΣΩΣΕΙΣ

στο MATHEMATICA 4.1

(Σημειώσεις)



Γ. Βουγιατζής
Λέκτορας του τμήματος Φυσικής, Α.Π.Θ.

Με την Ενίσχυση του Προγράμματος
ΕΠΕΑΕΚ ΙΙ : «Εισαγωγή Δεξιοτήτων Πληροφορικής και Νέων Τεχνολογιών»
Αναμόρφωση του Π.Π.Σ. του τμήματος Φυσικής του Α.Π.Θ.

ΠΡΟΛΟΓΟΣ

Ο «*Συμβολικός Προγραμματισμός*» αποτελεί λογισμικό με στόχο την αναλυτική διαχείριση μαθηματικών προβλημάτων. Παρουσιάστηκε για πρώτη φορά την δεκαετία του ‘80. Σήμερα, οι δυνατότητές του βρίσκονται σε τέτοιο βαθμό που μπορεί να εξυπηρετήσει, όχι μόνο εκπαιδευτικά, αλλά και ερευνητικά θέματα, στα πλαίσια της μαθηματικής ανάλυσης και των εφαρμογών της σε διάφορες επιστήμες. Αποτελεί ένα σημαντικό εργαλείο, που μας απαλλάσσει σε μεγάλο βαθμό από την πολυπλοκότητα των μαθηματικών πράξεων, που παρουσιάζονται σε πολλά, έστω και απλά, προβλήματα. Λογισμικά πακέτα συμβολικού προγραμματισμού αποτελούν τα Mathematica, Maple, Derive κ.α. Στις σημειώσεις αυτές γίνεται χρήση του πακέτου Mathematica για το οποίο ο αναγνώστης μπορεί να αναφερθεί, για περισσότερες πληροφορίες, στη διεύθυνση http://www.wolfram.com.

Στόχος των σημειώσεων αυτών, είναι να αποτελέσουν ένα τεχνικό εγχειρίδιο σχετικά με την επίλυση και διαχείριση των Διαφορικών εξισώσεων στο υπολογιστικό περιβάλλον του MATHEMATICA. Δίνεται ιδιαίτερη προσοχή στις περιπτώσεις όπου τα αποτελέσματα μπορεί να μην παρουσιάζονται στην αναμενόμενη μορφή τους. Μέσω παραδειγμάτων, εξετάζονται οι δυνατότητες του MATHEMATICA καθώς και οι περιορισμοί του για το συγκεκριμένο αντικείμενο. Αρκετά παραδείγματα και ασκήσεις αναφέρονται και αναλύονται στο βιβλίο του καθ. Γ. Μπόζη “Διαφορικές εξισώσεις και εφαρμογές” [1]. Έτσι μπορεί να γίνει μια άμεση σύγκριση μεταξύ των αναλυτικών μεθόδων της θεωρίας των διαφορικών εξισώσεων και των υπολογιστικών μεθόδων του MATHEMATICA. Ο αναγνώστης, θα πρέπει να διαθέτει κάποια βασική εξοικείωση με τον συμβολικό προγραμματισμό, τον τρόπο σύνταξής του, το περιβάλλον του προγράμματος καθώς και κάποιες γνώσεις σχετικά με βασικές εντολές ανάλυσης και σχεδίασης. Το βιβλίο «Εισαγωγή στη Mathematica» του Καθ. Γ.Θεοδώρου [2] μπορούν να καλύψουν τον αναγνώστη σε προαπαιτούμενες γνώσεις.

Τα προγράμματα, που παρουσιάζονται, γράφτηκαν με χρήση, όσο το δυνατό, λίγων εντολών και επιλογών και εκτελέστηκαν στο ολοκληρωμένο περιβάλλον του Mathematica 4.1 για PC-Windows. Ευχαριστίες εκφράζονται στον υποψήφιο διδάκτορα Β.Κουκουλογιάννη για την βοήθειά του. Οι παρούσες σημειώσεις γράφτηκαν στα πλαίσια της αναβάθμισης των σπουδών για τους φοιτητές του Φυσικού τμήματος του Α.Π.Θ.

Περιεχόμενα

ΜΕΡΟΣ 1^{o} : ΑΝΑΛΥΤΙΚΗ ΜΕΘΟΔΟΣ

§1. Σύνταξη Συνήθων Διαφορικών Εξισώσεων στο MATHEMATICA. Η εντολή DSolve.

Μια συνήθης διαφορική εξίσωση (ΣΔΕ) n-τάξης αναφέρεται σε μια άγνωστη συνάρτηση $y = y(x) \in R$ μιας ανεξάρτητης μεταβλητής $x \in R$ και είναι μια εξίσωση που περιέχει τουλάχιστον μια παράγωγο n-τάξης της συνάρτησης ως προς την ανεξάρτητη μεταβλητή της

$$F(x, y, \frac{dy}{dx}, \frac{d^2y}{dy^2}, ..., \frac{d^ny}{dy^n}) = 0 \qquad (1)$$

Για τις παραγώγους , είναι εύχρηστο να χρησιμοποιείται ο συμβολισμός

$$\frac{dy}{dx} = y', \frac{d^2y}{dx^2} = y'', \frac{d^3y}{dx^3} = y''', \frac{d^4y}{dx^4} = y^{(4)}, ..., \frac{d^ny}{dx^n} = y^{(n)} \qquad (2)$$

Οι ΣΔΕ στο MATHEMATICA γράφονται σε αντιστοιχία με τα παραπάνω, λαμβάνοντας υπόψη μας ότι οι συναρτήσεις και οι παράγωγες συντάσσονται ως εξής

$$y(x) \to y[x], \quad dy/dx \to D[y[x], x] \to y'[x], \quad d^2y/dx^2 \to D[y[x], \{x, 2\}] \to y''[x] \quad \text{κ.ο.κ.}$$

Γενικά ο τελεστής παραγώγισης **D[,x]** του MATHEMATICA ισοδυναμεί με μερική παραγώγιση ως προς την μεταβλητή **x** δηλαδή

$$D[f, x] \equiv \partial f / \partial x$$

Όμως για τις ΣΔΕ, όπου εμπλέκεται μια ανεξάρτητη μεταβλητή, η μερική και ολική παραγώγιση είναι συμβολικά ισοδύναμες.

Παραδείγματα

α. $\frac{d^2y}{dx^2} + \frac{1}{x}\frac{dy}{dx} + xy = 0 \quad (\eta' \quad y'' + \frac{1}{x}y' + xy = 0)$

```
eq1 = D[y[x], {x, 2}] + D[y[x], x] / x + x*y[x] == 0
```

ή

```
eq1 = y''[x] + y'[x] / x + x*y[x] == 0
```

β. $y''' + xy' + z = 0, \quad z \in R$

```
eq2 = y'''[x] + x y'[x] + z == 0
```

γ. $s'' + s = 0$

```
eq3 = s''[r] + s[r] == 0
```

δ. $y' + x^2 = x + (y')^2$

```
eq4 = y'[x] + x^2 == x + y'[x]^2
```

Το διπλό ίσον "==" αναφέρεται στην ταυτοτική ισότητα του αριστερού μέλους της εξίσωσης με το δεξιό και είναι ένας λογικός τελεστής με αποτέλεσμα True ή False. Τα σύμβολα[1] eq1, eq2 κλπ αντιπροσωπεύουν την εξίσωση που τους έχει ανατεθεί με το σύμβολο ανάθεσης "=". Έτσι, σε επόμενες γραμμές εντολών μπορούμε να αναφερθούμε σε μια ΣΔΕ με το σύμβολό της αντί να την ξαναγράφουμε. Η άγνωστη συνάρτηση πρέπει να δηλώνεται πάντοτε με το όρισμά της. Στην περίπτωση της εξίσωσης (β) δεν είναι σαφές αν η άγνωστη συνάρτηση y είναι συνάρτηση της μεταβλητής x ή της z. Στην σύνταξη του MATHEMATICA, αυτό γίνεται σαφές γράφοντας την άγνωστη συνάρτηση ως *y[x]*. Έτσι στην eq2 το z αντιμετωπίζεται σαν μια σταθερή παράμετρος της εξίσωσης. Στην εξίσωση (γ) δεν περιέχεται η ανεξάρτητη μεταβλητή και έχουμε την ελευθερία (και την υποχρέωση) να επιλέξουμε κάποιο έγκυρο σύμβολο ως μεταβλητή (πχ το r). Επίσης στην περίπτωση

[1] ή, αλλιώς μεταβλητές ή ποσότητες ή ετικέτες

(δ) γίνεται σαφές ότι το δεύτερο μέλος της εξίσωσης δεν είναι απαραίτητα το μηδέν και οι όροι της ΣΔΕ μπορούν να διαχωριστούν αριστερά και δεξιά του τελεστή ισότητας “==”. Σε κάθε περίπτωση θα πρέπει να χρησιμοποιούνται *έγκυρα* σύμβολα για τη συνάρτηση και την ανεξάρτητη μεταβλητή, δηλαδή, σύμβολα στα οποία δεν τους έχει ανατεθεί προηγουμένως κάποια τιμή με τον τελεστή “=”.

Κάθε συνάρτηση $y=y(x)$ η οποία ικανοποιεί τη ΣΔΕ (1) ονομάζεται **λύση** της (1) και με τον όρο επίλυση της (1) εννοούμε την εύρεση αυτών των συναρτήσεων $y=y(x)$ οι οποίες μπορεί να δίνονται και σε πλεγμένη μορφή $F(x,y)=0$. Δηλαδή αν αντικαταστήσουμε στη διαφορική εξίσωση την $y=y(x)$ και εκτελέσουμε τις απαιτούμενες παραγωγίσεις καταλήγουμε σε ταυτοποίηση του 1ου και 2ου μέλους της ΣΔΕ. Πχ. μια λύση της $y'+xy''=1$ είναι η $y_1=x+log(x)$:

```
In[1]:= y1 = x + Log[x]; eq1 = D[y1, x] + x D[y1, {x, 2}] == 1

Out[1]= True
```

Βέβαια και η συνάρτηση $y_2=x+2\ log(x)$ αποτελεί λύση καθώς και η συνάρτηση $y_3=x+c\cdot log(x),\ \forall c\in R$ (δηλαδή η παράμετρος c εκφράζει μια αυθαίρετη σταθερά). Η λύση y_3 αποτελεί μια μονοπαραμετρική οικογένεια συναρτήσεων, με παράμετρο το c, και φυσικά εμπεριέχει και τις λύσεις y_1 και y_2 για c=1 και c=2 αντίστοιχα. Ως **γενική λύση** μιας ΣΔΕ, n τάξης, εννοούμε μια n-παραμετρική οικογένεια συναρτήσεων, δηλαδή μια συνάρτηση του x και n αυθαίρετων σταθερών :

$$(\alpha)\quad y = y(x;c_1,c_2,...,c_n)\quad \eta'\quad (\beta)\quad F(x,y;c_1,c_2,...,c_n)=0 \qquad (3)$$

Για παράδειγμα, η γενική λύση της $y'+xy''=1$, που είναι δεύτερης τάξης, θα πρέπει να περιέχει δύο αυθαίρετες σταθερές και είναι η $y=x+c_1log(x)+c_2$.

Η επίλυση μιας ΣΔΕ με το MATHEMATICA γίνεται με την εντολή **DSolve** η οποία συντάσσεται με τρία βασικά ορίσματα, ως

DSolve[εξίσωση , αγνωστη συνάρτηση, ανεξάρτητη μεταβλητή]

DSolve[*equation*, y[x],x]

• Η λειτουργία της παραπάνω εντολής στηρίζεται στη διαμόρφωση (αν είναι απαραίτητο) της διαφορικής εξίσωσης σε κάποια γενική μορφή για την οποία έχουν ήδη αναπτυχθεί συγκεκριμένοι αλγόριθμοι επίλυσης. Το MATHEMATICA, λοιπόν, *λύνει διαφορικές εξισώσεις που ήδη έχουν λυθεί* στα πλαίσια των μαθηματικών. Δεν πρέπει να συγχέεται η μη δυνατότητα εύρεσης της λύσης από το MATHEMATICA με τη μη ύπαρξη αναλυτικής λύσης για τη διαφορική εξίσωση[2]. Γενικά η DSolve, όπου είναι δυνατό, παράγει λύσεις της μορφής (3α) (λυμένες μορφές – explicit forms).

Παράδειγμα 1

```
In[2]:= DSolve[y'[x] + x y''[x] == 1, y[x], x]

Out[2]= {{y[x] → x + C[2] + C[1] Log[x]}}
```

Παρατηρούμε ότι, η DSolve μας δίνει την αναμενόμενη λύση με αυθαίρετες σταθερές τις C[1] και C[2]. Η ονομασία –συμβολισμός των αυθαίρετων σταθερών μπορεί να αλλάξει με την επιλογή DSolveConstants η οποία εισάγεται σαν τέταρτο όρισμα στη DSolve πχ

```
In[9]:= DSolve[y'[x] + x y''[x] == 1, y[x], x, DSolveConstants → k]

Out[9]= {{y[x] → x + k[2] + k[1] Log[x]}}
```

Παράδειγμα 2

```
In[2]:= DSolve[y'[x] - y[x] == Sin[x], y[x], x]
```

Out[2]= $\left\{\left\{y[x] \to e^{x} C[1] + e^{x}\left(-\frac{1}{2} e^{-x} Cos[x] - \frac{1}{2} e^{-x} Sin[x]\right)\right\}\right\}$

[2] Πληροφορίες σχετικά με τις περιπτώσεις οι οποίες αντιμετωπίζονται επιτυχώς δίνονται στο HELP του MATHEMATICA για την εντολή DSolve.

Αν και είναι προφανής η απλοποίηση, που θα έπρεπε να γίνει στο αποτέλεσμα `Out[2]`, αυτό δεν αποτελεί αρμοδιότητα της `DSolve`. Έτσι η εφαρμογή της εντολής `Simplify` (ή `FullSimplify`) στο αποτέλεσμα της `DSolve`, γενικά, ενδείκνυται.

```
In[3]:= Simplify[%]
```

Out[3]= $\left\{\left\{y[x] \to e^{x} C[1] - \frac{Cos[x]}{2} - \frac{Sin[x]}{2}\right\}\right\}$

Παράδειγμα 3

In[4]:= $\text{DSolve}\left[y'[x] == \frac{x + y[x]^2}{2 x y[x]}, y[x], x\right]$

Out[4]= $\left\{\left\{y[x] \to -\sqrt{x C[1] + x Log[x]}\right\}, \left\{y[x] \to \sqrt{x C[1] + x Log[x]}\right\}\right\}$

Φυσικά η παραπάνω εξίσωση δεν έχει δύο γενικές λύσεις, απλά η γενική λύση εκφράζεται σε δύο κλάδους, ξεχωριστά για $y<0$ και για $y>0$. Μια πιο συμπαγής μορφή λύσης θα ήταν η $y^2 = xc_1 + x\log x$, όμως η `DSolve` δίνει λύσεις της λυμένης μορφής (3α).

§2. Διαχείριση των λύσεων και των αυθαίρετων σταθερών

Όπως φαίνεται από τα παραδείγματα της προηγούμενης παραγράφου, οι λύσεις δίνονται με τα παρακάτω χαρακτηριστικά :

α) Δίνονται σαν αντικείμενα μιας λίστας (δηλαδή η έξοδος της `DSolve` είναι μια λίστα και όχι μια συνάρτηση.

β) Δίνονται μέσω του συμβόλου "→" και δεν αναθέτονται στη μεταβλητή *y* η οποία ήταν η άγνωστη συνάρτηση. Με το τρόπο αυτό δεν δεσμεύεται το σύμβολο *y* και έτσι δεν αλλοιώνεται η διαφορική εξίσωση.

Η λίστα των λύσεων μπορεί να ανατεθεί σε ένα σύμβολο. Επίσης είναι εύχρηστο, για μια περαιτέρω επεξεργασία της λύσης, να ανατεθεί η λύση σε μια μεταβλητή ή μια συνάρτηση επιλογής του χρήστη.

Παράδειγμα 1

```
In[1]:= eq1 = y'[x] + x y''[x] == 1;  solution = DSolve[eq1, y[x], x]

Out[1]= {{y[x] → x + C[2] + C[1] Log[x]}}
```

Η λίστα `solution` περιέχει ένα στοιχείο (λύση) το οποίο μπορεί να αποδοθεί στο σύμβολο yy

```
In[2]:= yy = y[x] /. solution[[1]]  (*ή ισοδύναμα  yy=solution[[1,1,2]]*)

Out[2]= x + C[2] + C[1] Log[x]
```

Η απόδοση της λύσης δεν γίνεται στο σύμβολο `y` (αλλά στο `yy`) γιατί η ΣΔΕ `eq1` θα αλλοιωθεί και δεν μπορεί να ξαναχρησιμοποιηθεί παρά μόνο αν διαγραφεί ο πυρήνας της MATHEMATICA (*Quit Kernel*). Επίσης θα μπορούσαμε να χρησιμοποιήσουμε αντί της `yy` την `yy[x]` οπότε ορίζουμε ρητά την λύση σαν συνάρτηση του *x*.

Αν στο δεύτερο όρισμα της `DSolve` αντί του `y[x]` χρησιμοποιηθεί το όρισμα `y` τότε παίρνουμε λύσεις σε μορφή *καθαρής* συνάρτησης (pure or anonymous function)

```
In[1]:= eq1 = y'[x] + x y''[x] == 1;  solution = DSolve[eq1, y, x]

Out[1]= {{y → Function[{x}, x + C[2] + C[1] Log[x]]}}
```

Η παραπάνω μορφή συνάρτησης δεν δεσμεύει σύμβολα για το όνομά της καθώς και για την ανεξάρτητη μεταβλητή και παρουσιάζει αρκετά πλεονεκτήματα σε θέματα χρήσης και αντικατάστασης (βλ. HELP του MATHEMATICA και [2],[3]) παράδειγμα η επαλήθευση της λύσης μπορεί να γίνει άμεσα:

In[3]:= **eq1 /. solution[[1]]**

Out[3]= True

. Η απόδοση της λύσης σε ένα σύμβολο γίνεται όπως δείξαμε παραπάνω. Οι αυθαίρετες σταθερές C[*n*], που περιλαμβάνονται στη λύση, αποτελούν προστατευόμενες ποσότητες για το MATHEMATICA και δεν μπορούν να τις αποδοθούν άμεσα τιμές. Έτσι για παράδειγμα η έκφραση C[1]=1; θα μας δώσει το μήνυμα λάθους "Tag C in C[1] is protected". Έτσι είναι εύχρηστο να γίνεται αντικατάστασή τους με σύμβολα του χρήστη πχ για την λύση yy γράφουμε

In[4]:= **yy /. {C[1] → c1, C[2] → c2}**

Out[4]= c2 + x + c1 Log[x]

Θέτοντας στη γενική λύση της ΣΔΕ συγκεκριμένες τιμές για τις *n* αυθαίρετες παραμέτρους c_i παίρνουμε συγκεκριμένες λύσεις που ονομάζονται "**μερικές λύσεις**" της ΣΔΕ.

Για διαφορικές εξισώσεις 1ης τάξης, η γενική λύση παρουσιάζεται σαν μια μονοπαραμετρική οικογένεια μερικών λύσεων *y=y(x;c)*, όπου η κάθε μια (για κάθε τιμή του *c*) εκφράζει μια καμπύλη στο χώρο *x-y*. Συνήθως η γενική λύση είναι μια συνεχής συνάρτηση ως προς την παράμετρο, και η σχεδίαση μερικών καμπυλών μας δίνει την αίσθηση της ποιοτικής συμπεριφορά όλων των άλλων λύσεων

Παράδειγμα 2.

In[2]:= **solution = DSolve**$\left[y'[x] == \frac{y[x] - (x+1)\,e^{-x}}{x}, y[x], x\right]$

Out[2]= {{y[x] → e^{-x} + x C[1]}}

Από την παραπάνω μερική λύση παίρνουμε μια μερική λύση αντικαθιστώντας την σταθερά C[1] με μια συγκεκριμένη τιμή.

In[3]:= **PartialSolution = solution /. C[1] → 3**

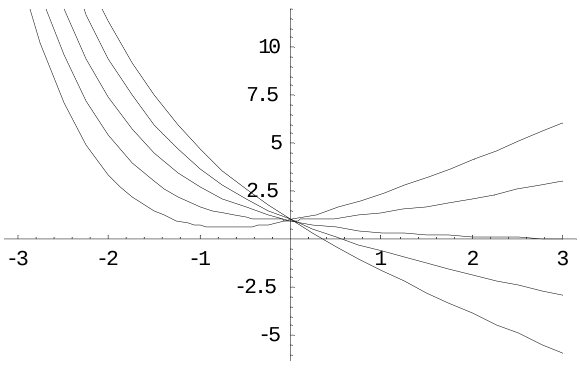
Out[3]= {{y[x] → e^{-x} + 3 x}}

Η σχεδίαση της παραπάνω λύσης σε ένα διάστημα [x_{min},x_{max}] μπορεί να γίνει με την εντολή Plot[y[x]\.PartialSolution, {x,xmin,xmax}] ή Plot[PartialSolution[[1,1,2]], {x,xmin,xmax}]. Για περισσότερες μερικές λύσεις μπορούμε να γράψουμε

```
In[4]:= Plot[{solution[[1, 1, 2]] /. C[1] → -2,
          solution[[1, 1, 2]] /. C[1] → -1,
          solution[[1, 1, 2]] /. C[1] → 0,
          solution[[1, 1, 2]] /. C[1] → 1,
          solution[[1, 1, 2]] /. C[1] → 2}, {x, -3, 3}]
```

ή, ισοδύναμα

```
In[7]:= partialsolutions =
         Table[solution[[1, 1, 2]] /. C[1] → i, {i, -2, 2}];
        Plot[Evaluate[partialsolutions], {x, -3, 3}]
```

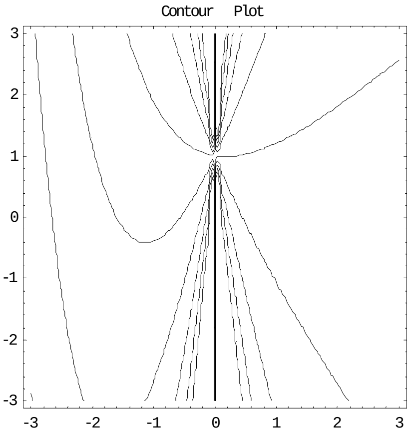


Ένας εναλλακτικός τρόπος για την σχεδίαση των μερικών λύσεων μιας ΣΔΕ 1ης τάξης είναι να φέρουμε τη λύση σε πλεγμένη μορφή F[x,y]=c και να σχεδιάσουμε τις ισοκλινείς της καμπύλες

In[7]:= **Implicitsolution = Solve[yy == solution[[1, 1, 2]], C[1]]**

Out[7]= $\left\{\left\{C[1] \to -\frac{e^{-x} - yy}{x}\right\}\right\}$

```
In[8]:= ContourPlot[Implicitsolution[[1, 1, 2]], {x, -3, 3}, {yy, -3, 3},
    ContourShading → False, PlotPoints → 100]
```



Χρησιμοποιήσαμε το σύμβολο yy και όχι το y για τους λόγους που αναφέραμε παραπάνω. Επίσης σε μερικές περιπτώσεις μπορεί να φανεί χρήσιμη η χρήση της ImplicitPlot[].

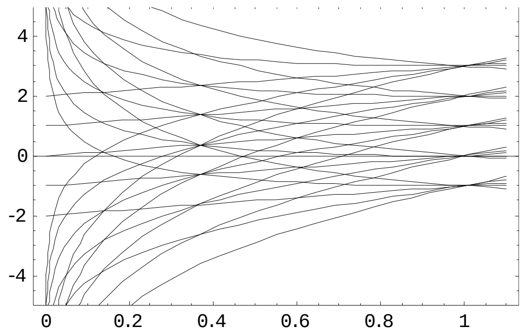
Ένας ακόμη τρόπος για την διαχείριση των μερικών λύσεων, βάσει των αυθαίρετων σταθερών, είναι να ορίσουμε τις λύσεις σαν μια συνάρτηση με όρισμα την αυθαίρετη σταθερά, πχ

```
[  ] = y[x] /. solution[[1]] /. C[1] → c;
Plot[{yy[-2], yy[-1], yy[0],  .. κλπ ..}, {x, -3, 3}]
```

Ο παραπάνω τρόπος μπορεί να επεκταθεί και για ΣΔΕ μεγαλύτερης τάξης από ένα.

Παράδειγμα 3. Σχεδιάζουμε μερικές λύσεις της εξίσωσης του παραδείγματος 1.

```
In[1]:= eq1 = y'[x] + x y''[x] == 1;
    solution = DSolve[eq1, y, x]
    yy[c1_, c2_] =
     y[x] /. solution[[1]] /. {C[1] → c1, C[2] → c2}
    psols =
     Table[yy[c1, c2], {c1, -2, 2}, {c2, -2, 2}]
    Plot[Evaluate[psols], {x, 0, 1.1},
       PlotRange → {-5, 5}, Frame → True]
```



§3. Αρχικές και Συνοριακές συνθήκες

Στις εφαρμογές των ΣΔΕ, συνήθως αντιμετωπίζονται *Προβλήματα αρχικών ή συνοριακών τιμών*, δηλαδή αναζητούμε μερικές λύσεις (μέσα από το σύνολο λύσεων της γενικής λύσης) οι οποίες πληρούν συγκεκριμένες (ή συμπληρωματικές) συνθήκες . Η εύρεση μιας τέτοιας μερικής λύσης σημαίνει τον προσδιορισμό κατάλληλων τιμών για τις αυθαίρετες σταθερές. Μπορούμε να κατατάξουμε τις απαιτούμενες συνθήκες σε δύο βασικές κατηγορίες

α) Αρχικές συνθήκες. Αναφέρονται στις συνθήκες, οι οποίες πρέπει να πληρούνται από τη λύση, σε μια συγκεκριμένη τιμή της ανεξάρτητης μεταβλητής. Γενικά για κάποιο $x=x_0$, ζητούμε συγκεκριμένες τιμές για τη τιμή του y ή/και των παραγώγων της (για $x=x_0$ είναι $y(x_0)=y_0$, $y'(x_0)=a$, $y''(x_0)=b$ κλπ)

β) Συνοριακές συνθήκες. Αναφέρονται στις συνθήκες, οι οποίες πρέπει να πληρούνται από τη λύση, σε διάφορες τιμές της μεταβλητής x (πχ για $x=x_1$ είναι $y(x_1)=y_1$, για $x=x_2$ είναι $y(x_2)=y_2$ κλπ)

Έστω, ότι για το παράδειγμα 2§2 ζητούμε τη λύση που περνάει από το σημείο (x_0,y_0)= (1,1), δηλαδή για $x=x_0=1$ είναι $y=y_0=1$. Πρέπει λοιπόν να πληρείται η εξίσωση y(1)=1, όπου $y=cx+e^{-x}$ η γενική λύση, δηλαδή για την τιμή της αυθαίρετης σταθεράς

```
In[1]:= Solve[c x + Exp[-x] == 1, c] /. {x → 1}
```

Out[1]= $\left\{\left\{c \to 1 - \frac{1}{e}\right\}\right\}$

Οι συνθήκες, που θέτουμε σε ένα *Πρόβλημα αρχικών τιμών*, μπορεί να μην ικανοποιούνται από καμία λύση (πχ στο παράδειγμα 2§2 δεν υπάρχει λύση που περνάει από το σημείο x=0,y=2). Επίσης, για n-παραμετρικές λύσεις, με $n>1$, μπορεί να έχουμε μια ολόκληρη οικογένεια λύσεων που πληρούν τις συνθήκες (πχ όταν δεν αρκούν οι αρχικές συνθήκες για να εξαλείψουμε όλες τις αυθαίρετες σταθερές) όπως στο παραπάνω πρόβλημα όπου η συνθήκη *y(0)=1* πληρείται από όλες τις λύσεις. Για μια *n*-τάξης ΣΔΕ, ένα πλήρες σύστημα αρχικών συνθηκών είναι το

$$y(x_0)=y_o\ ,\ y'(x_0)=a_1\ ,\ y''(x_0)=a_2\ ,.....,\ y^{(n-1)}(x_0)=a_{n-1}\qquad a_i:\text{σταθ.}\qquad (4)$$

Η ύπαρξη και η μοναδικότητα λύσεων, που πληρούν τις συνθήκες (4), εξασφαλίζονται από συγκεκριμένα θεωρήματα[3]. Χρειάζεται ιδιαίτερη προσοχή για να ερμηνεύονται σωστά τα αποτελέσματα της MATHEMATICA όταν δεν πληρούνται αυτά τα θεωρήματα.

Η εύρεση λύσεων που ικανοποιούν αρχικές ή συνοριακές συνθήκες γίνεται άμεσα από την εντολή `DSolve` η οποία μπορεί να συνταχθεί και ως εξής

DSolve[{ΔΕ, συνθήκη1,συνθήκη2,...**},** Άγνωστη συνάρτηση**,** Ανεξάρτητη μεταβλητή**]**

DSolve[{equation,y[x₀]==a,y’[x₀]=b,...},y[x],x]

Παράδειγμα 1.

In[1]:= **eq1 = y'[x] == $\frac{y[x]-(x+1)\,e^{-x}}{x}$; DSolve[{eq1, y[1] == 1}, y[x], x] // Simplify**

Out[1]= $\left\{\left\{y[x]\to e^{-x}+x-\frac{x}{e}\right\}\right\}$

Αν θεωρήσουμε τώρα τις περιπτώσεις των ιδιόμορφων σημείων (0,1) και (0,2), που αναφέρθηκαν παραπάνω, θα έχουμε αντίστοιχα

In[2]:= **DSolve[{eq1, y[0] == 1}, y[x], x]**

```
DSolve::bvnr : For some branches of th  general solution, the given boundary
   conditions does not restrict the existing freedom in the general solution.
```

Out[2]= $\{\{y[x]\to e^{-x}(1+e^{x}\,x\,C[1])\}\}$

δηλαδή παίρνουμε όλες τις λύσεις αφού η αρχική συνθήκη δεν δεσμεύει τις λύσεις, και

In[3]:= **DSolve[{eq1, y[0] == 2}, y[x], x]**

```
DSolve::bvnul : For some bran hes of the general
   solution, the given boundary conditions lead to an empty solution.
```

δηλαδή δεν βρέθηκε καμία λύση.

Γενικά μπορούμε να κάνουμε χρήση «αυθαίρετων» αρχικών συνθηκών με την έννοια ότι δεν αναφέρονται σε συγκεκριμένες τιμές. Γενικά, για το πρόβλημα αρχικών συνθηκών (4), η γενική λύση μπορεί να γραφτεί ως $y=y(x;\ x_0,y_0,y'_0,...)$ και η οποία μπορεί να δοθεί από το MATHEMATICA όπως παρακάτω

Παράδειγμα 2

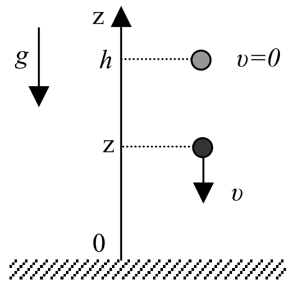
```
In[1]:= solution = DSolve[{y'[x] + x y''[x] == 1, y[x0] == y0, y'[x0] == p0}, y[x], x];
        yy[x_, x_0, p_0] = solution[[1, 1, 2]] // Simplify
```

```
Out[2]= x - x0 + y0 + (-1 + p0) x0 Log[x] + (x0 - p0 x0) Log[x0]
```

Παράδειγμα 3. Εξισώσεις με ελεύθερες σταθερές παραμέτρους

Θεωρούμε σώμα μάζας *m* το οποίο αφήνεται να πέσει από ύψος *h* μέσα στο ομογενές πεδίο βαρύτητας της Γης. Αν η αντίσταση του αέρα είναι ανάλογη της ταχύτητας, με συντελεστή αναλογίας *k*>0, να βρεθεί η ταχύτητα πτώσης υ_π του σώματος στην επιφάνεια σαν συνάρτηση του ύψους.

[3] Για τα θέματα αυτά βλ. [1] και [8]



Η διαφορική εξίσωση που περιγράφει την ευθύγραμμη πτώση του σώματος θα είναι η εξίσωση του Νεύτωνα

$$m\frac{d^2z}{dt^2}=F_B+F_{\alpha\nu\tau}=-mg-k\frac{dz}{dt}$$

η οποία γράφεται, μαζί με τις αρχικές συνθήκες, ως

$$\begin{aligned} &z''=-g-a\,z' \quad , \quad a=k/m>0 \\ &z(0)=h, \quad z'(0)=0 \end{aligned} \qquad (5)$$

Η λύση *z=z(t)* του προβλήματος αρχικών τιμών (5) δίνεται από τη `DSolve` :

```
In[1]:= solution = DSolve[{z''[t] == -g - a z'[t], z[0] == h, z'[0] == 0}, z, t];
        zz = z[t] /. solution[[1]] // Simplify
        υ = z'[t] /. solution[[1]] // Simplify
```

με αποτέλεσμα

```
In[4]:= Print["ύψος zz=", zz, "          ταχύτητα υ=", υ]
```

ύψος $zz=\dfrac{g-e^{-at}g+a^2h-agt}{a^2}$ ταχύτητα $\upsilon=\dfrac{(-1+e^{-at})\,g}{a}$

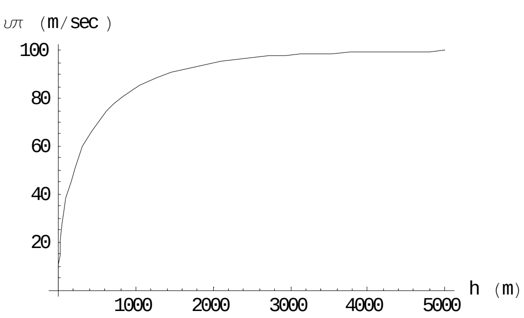
Η λύση δεν περιέχει αυθαίρετες σταθερές, αφού προήλθε από ένα πρόβλημα αρχικών τιμών. Όμως αποτελεί ένα σύνολο λύσεων αφού η αρχική συνθήκη *z=h* δεν είναι προκαθορισμένη (το *h* είναι παράμετρος του προβλήματος). Με τη συγκεκριμένη συνθήκη *υ(0)=z'(0)=0* απαλείφθηκε μια αυθαίρετη σταθερά.

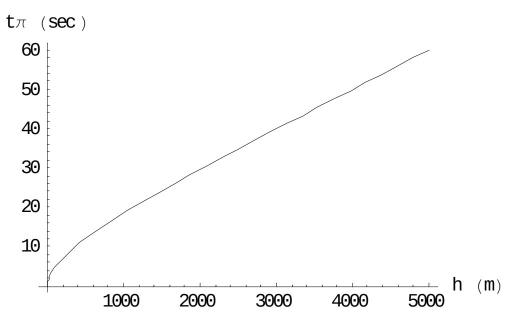
Για να βρούμε τη σχέση που μας ζητείται $\upsilon_\pi=f(h)$ πρέπει να επιλύσουμε τις εξισώσεις *z(t)=0* και $\upsilon(t)=\upsilon_\pi$ το οποίο αποτελεί ένα σύστημα 2 εξισώσεων με δύο αγνώστους, τον χρόνο πτώσης και την ταχύτητα πτώσης. Η Solve δεν μπορεί να επιλύσει το παραπάνω σύστημα και έτσι επιλύουμε πρώτα την εξίσωση *z(t)=0* ως προς το χρόνο και αντικαθιστούμε στην $\upsilon(t)=\upsilon_\pi$.

Χρόνος Πτώσης $=\left\{\dfrac{g+a^2h+g\,\text{ProductLog}\left[-e^{-1-\frac{a^2h}{g}}\right]}{a\,g}\right\}$ ταχύτητα Πτώσης $\left\{\dfrac{\left(-1+e^{-\frac{g+a^2h+g\,\text{ProductLog}\left[-e^{-1-\frac{a^2h}{g}}\right]}{g}}\right)g}{a}\right\}$

Τα αποτελέσματα δίνονται από την ειδική συνάρτηση[4] *ProductLog*. Θέτοντας τώρα συγκεκριμένες τιμές για τις παραμέτρους *g,m* και *k* μπορούμε να σχεδιάσουμε γραφικά τις σχέσεις (h,υ_π) και (h,t_π).

```
In[11]:= g = 10; m = 1; k = 0.1; a = k/m;
         Plot[Abs[υπ], {h, 0, 5000}, AxesLabel → {"h (m)", "υπ (m/sec)"}];
         Plot[Abs[tπ], {h, 0, 5000}, AxesLabel → {"h (m)", "tπ (sec)"}];
         Show[GraphicsArray{graph1, graph2}];
```





4 βλ και §4.

§4. Γενικές Παρατηρήσεις και Ειδικές Περιπτώσεις

Σε πολλές περιπτώσεις η λύσεις που δίνονται από το MATHEMATICA δεν είναι κομψές και απαιτείται κάποια ιδιαίτερη προσοχή για την κατανόησή τους

Παράδειγμα 1.

Θεωρούμε την διαχωρίσιμη ΣΔΕ $y'=1-y^2$

```
In[3]:= DSolve[y'[x] == 1 - y[x]^2, y[x], x] // Simplify
```

Out[3]= $\left\{\left\{y[x] \to \frac{e^{2x}+e^{2C[1]}}{e^{2x}-e^{2C[1]}}\right\}\right\}$

Παρατηρούμε ότι η αυθαίρετη σταθερά παρουσιάζεται στη μορφή e^{2c}. Αν θέσουμε $C\equiv e^{2c}$ τότε θα πρέπει να δηλώσουμε με σαφήνεια ότι $C>0$. Με τον τρόπο αυτό υποδηλώνεται, από το MATHEMATICA , ο θετικός χαρακτήρας των αυθαίρετων όρων που εισάγονται στη γενική λύση. Αν απλοποιήσουμε τη λύση χρησιμοποιώντας την εντολή `FullSimplify`, παίρνουμε μια, φαινομενικά, διαφορετική λύση στη μορφή

```
Out[4]= {{y[x] → Coth[x - C[1]]}}
```

όπου τώρα η αυθαίρετη σταθερά μπορεί να είναι είτε αρνητική είτε θετική. Φυσικά, οι δύο παραπάνω μορφές λύσης είναι ισοδύναμες εξ' ορισμού.

Παράδειγμα 2

Γενικά οι n-τάξης μη γραμμικές διαφορικές εξισώσεις με $n>1$ **ΔΕΝ** έχουν αναλυτικές λύσεις

```
In[1]:= eq = y''[x] + y[x] - y[x]^3 + x == 0;
        DSolve[eq, y[x], x]
```

Out[2]= DSolve$[x+y[x]-y[x]^3+y''[x]==0,\ y[x],\ x]$

Οι γραμμικές ΔΕ 2ης τάξης (με μη σταθερούς συντελεστές) χωρίζονται σε διάφορες κατηγορίες , οι οποίες, γενικά, επιδέχονται αναλυτικές λύσεις μέσω **ειδικών συναρτήσεων** (βλ. [9][10]), πχ

In[1]:= **DSolve**$[x^2 y''[x] + x\, y'[x] + x^2 y[x] == 0,\ y[x],\ x]$

```
Out[1]= {{y[x] → BesselJ[0, x] C[1] + BesselY[0, x] C[2]}}
```

ή λύσεις που είναι αρκετά πολύπλοκες

In[2]:= **DSolve**$[y''[x] + y[x] - y[x]^3 == 0,\ y[x],\ x]$

```
Solve::ifun : Inverse functions are being used by Solve, so some solutions may not be found.
```

$$\Bigg\{\Bigg\{y[x]\to \frac{1}{2}\Bigg(2\,i\sqrt{2}\sqrt{-\frac{1}{2-\sqrt{4-8C[1]}}}\ \text{JacobiSN}\Big[-\frac{1}{2}\sqrt{\left(2x^2+x^2\sqrt{4-8C[1]}+4x\,C[2]+2x\sqrt{4-8C[1]}\,C[2]+2C[2]^2+\sqrt{4-8C[1]}\,C[2]^2\right)},\ \frac{2-\sqrt{4-8C[1]}}{2+\sqrt{4-8C[1]}}\Big]-i\sqrt{2}\sqrt{-\frac{1}{2-\sqrt{4-8C[1]}}}\sqrt{4-8C[1]}\ \text{JacobiSN}\Big[-\frac{1}{2}\sqrt{\left(2x^2+x^2\sqrt{4-8C[1]}+4x\,C[2]+2x\sqrt{4-8C[1]}\,C[2]+2C[2]^2+\sqrt{4-8C[1]}\,C[2]^2\right)},\ \frac{2-\sqrt{4-8C[1]}}{2+\sqrt{4-8C[1]}}\Big]\Bigg)\Bigg\},$$

(κλπ ...)

Το μήνυμα που παρουσιάζεται επισημαίνει την προσοχή μας στο ενδεχόμενο οι κλάδοι της γενικής λύσης που βρέθηκαν να μην είναι επαρκείς για όλο το χώρο των αρχικών συνθηκών.

Παράδειγμα 3

Θεωρούμε την 1ης τάξης ΣΔΕ Bernoulli $y' + xy - x^3 y^3 = 0$. Υπενθυμίζουμε ότι οι εξισώσεις Bernoulli μετατρέπονται σε γραμμικές με το μετασχηματισμό $y = z^{1/r}$ (για την συγκεκριμένη περίπτωση $y = 1/\sqrt{z}$, z>0). Έτσι προκύπτει η λύση

$$y = 1/\sqrt{c\,e^{x^2} + x^2 + 1} \quad , \quad c\,e^{x^2} + x^2 + 1 > 0 \qquad (6)$$

Χρησιμοποιώντας το MATHEMATICA παίρνουμε

In[1]:= **DSolve**[y'[x] + x y[x] - x^3 y[x]^3 == 0, y[x], x] // **FullSimplify**

$$\text{Out[1]=}\ \left\{\left\{y[x] \to -\frac{i}{\sqrt{-1 - x^2 - e^{x^2}\,C[1]}}\right\}, \left\{y[x] \to \frac{i}{\sqrt{-1 - x^2 - e^{x^2}\,C[1]}}\right\}\right\}$$

Η λύση εκφράζεται σε μιγαδική μορφή και δίνεται σε δύο κλάδους. Αν όμως αλλάξουμε τα πρόσημα μέσα στη τετραγωνική ρίζα παίρνουμε

$$y = \pm 1/\sqrt{c\,e^{x^2} + x^2 + 1} \quad , \quad c\,e^{x^2} + x^2 + 1 > 0 \qquad (7)$$

Η (7) ορίζεται λοιπόν και για *y<0* οπότε και είναι γενικότερη της (6). Αυτός ο επιπλέον κλάδος προέκυψε γιατί το MATHEMATICA <u>δεν προδικάζει ότι *y∈R*</u> (δεν έλαβε υπόψη του τη συνθήκη *z>0*). Επίσης, σε πολλές περιπτώσεις πραγματικών λύσεων, το MATHEMATICA τις παρουσιάζει σαν μιγαδικές, δηλαδή εμφανίζεται η μιγαδική μονάδα **i** και η απλοποίησή της ίσως να μην είναι προφανής.

Παράδειγμα 4

Θεωρούμε την εξίσωση τύπου Clairaut $y = xy' + 2y'^2$. Χρησιμοποιώντας το MATHEMATICA παίρνουμε τη γενική λύση στην παρακάτω μορφή

In[1]:= **DSolve**[x y'[x] + 2 y'[x]^2 == y[x], y[x], x]

$$\text{Out[1]=}\ \Big\{\Big\{y[x] \to \frac{1}{2}\left(16 + 8\,e^{C[1]} - 2\sqrt{2}\,e^{\frac{[\ ]}{2}}\,(-8 + x) - 4x\right),$$
$$\Big\{y[x] \to \frac{1}{2}\left(16 + 8\,e^{C[1]} + 2\sqrt{2}\,e^{\frac{C[1]}{2}}\,(-8 + x) - 4x\right)\Big\},$$
$$\Big\{y[x] \to \frac{1}{4}\left(32 - e^{C[1]} - 8x - \sqrt{2}\,\sqrt{-64\,e^{C[1]} + 16\,e^{C[1]}\,x - e^{C[1]}\,x^2}\right)\Big\},$$
$$\Big\{y[x] \to \frac{1}{4}\left(32 - e^{C[1]} - 8x + \sqrt{2}\,\sqrt{-64\,e^{C[1]} + 16\,e^{C[1]}\,x - e^{C[1]}\,x^2}\right)\Big\}\Big\}$$

Έχουμε 4 κλάδους στους οποίους οι αυθαίρετοι όροι είναι θετικοί ($e^{c[1]}$). Δεν είναι καθόλου προφανές ότι η παραπάνω λύση έχει την απλή μορφή

$$y = c\,x + 2c^2 \qquad (8)$$

Η (8) μπορεί να προκύψει από τον 1ο και 2ο κλάδο της λύσης του MATHEMATICA θέτοντας $c \equiv 2\log[-(c+2)/\sqrt{2}]$ και $c \equiv 2\log[(c+2)/\sqrt{2}]$. Ο 3ος και 4ος κλάδος αποτελούν μιγαδικές λύσεις. Επίσης υπάρχει και η ιδιάζουσα λύση[5] *y=-x²/8* η οποία όμως <u>**δεν**</u> εντοπίζεται από το MATHEMATICA.

Παράδειγμα 5

Η λύση μιας ΔΕ μπορεί να εκφράζεται σε πεπλεγμένη μορφή *Φ(x,y)=0* η οποία δεν μπορεί να λυθεί ως προς *y*, πχ για την διαχωρίσιμη ΣΔΕ $y' + xy^2(y+1) = 0$ παίρνουμε το παρακάτω αποτέλεσμα :

---

[5] Ιδιάζουσα είναι μια λύση της ΔΕ που δεν εντάσσεται στη γενική λύση.

```
In[2]:= DSolve[ y'[x] + x y[x]^3 == -x y[x]^2, y[x], x]

Solve::tdep : The equations appear to involve
   the variables to be solved for in an essentially non-algebraic way.
```

Out[2]= $\left\{\left\{y[x] \to \text{InverseFunction}\left[-\text{Log}[\#1] + \text{Log}[1+\#1] - \frac{1}{\#1} \,\&\right]\left[-\frac{x^2}{2} + C[1]\right]\right\}\right\}$

Δηλαδή $-\log y + \log(1+y) - 1/y = x^2/2 + c$. Αν η λύση δεν μπορεί να διαχωριστεί όπως προηγουμένως με αλγεβρικό τρόπο τότε, αντί της `InverseFunction` παίρνουμε ως αποτέλεσμα την `Solve[F==0, y[x]]`, όπου F=$F(x,y;c_i)$ η λύση σε πλεγμένη μορφή.

Παράδειγμα 5

Αν στο παράδειγμα 3§3 θεωρήσουμε ότι η αντίσταση του αέρα είναι ανάλογη του τετραγώνου της ταχύτητας τότε έχουμε τη ΣΔΕ $z'' = -g - az'^2$, με γενική λύση

```
In[1]:= eq = z''[x] == -g - a z'[x]^2;
        DSolve[eq, z[x], x]
```

Out[2]= $\left\{\left\{z[x] \to C[2] + \frac{\text{Log}\left[\text{Cos}\left[\sqrt{a}\,\sqrt{g}\,x - \sqrt{a}\,\sqrt{g}\,C[1]\right]\right]}{a}\right\}\right\}$

Αν όμως προσπαθήσουμε να επιλύσουμε το πρόβλημα αρχικών συνθηκών $z(0)=h$, $z'(0)=0$ τότε δεν είναι δυνατή η εύρεση της μερικής αυτής λύσης (!)

```
In[3]:= eq = z''[x] == -g - a z'[x]^2;
        DSolve[{eq, z[0] == h, z'[0] == 0}, z[x], x]
```

----- Επεξηγήσεις -----

§5. Συστήματα Διαφορικών εξισώσεων

Δύο ή περισσότερες διαφορικές εξισώσεις, οι οποίες περιέχουν περισσότερες από μία άγνωστες συναρτήσεις μιας ανεξάρτητης μεταβλητής, σχηματίζουν ένα σύστημα διαφορικών εξισώσεων. Με τον όρο λύση του συστήματος εννοούμε την εύρεση όλων των άγνωστων συναρτήσεων οι οποίες ικανοποιούν όλες τις εξισώσεις του συστήματος.

Ως, παράδειγμα, θεωρούμε το παρακάτω σύστημα :

$$a)\quad \frac{d^2 y_1}{dx^2} - y_2 + x = 0 \qquad b)\quad \frac{dy_2}{dx} + y_1 = 0 \tag{9}$$

που αποτελείται από δύο ΣΔΕ με άγνωστες συναρτήσεις τις y_1 και y_2 της ανεξάρτητης μεταβλητής x. Το MATHEMATICA χρησιμοποιεί και πάλι την εντολή `DSolve` για την επίλυση συστημάτων ΣΔΕ και η οποία συντάσσεται ως εξής

`DSolve`*[{Εξίσωση1, Εξίσωση2 κλπ},{Αγν.συναρτηση1, Αγν.συνάρτηση 2, κλπ}, Ανεξάρτ μεταβλητή]*

DSolve[{eq1,eq2,...},{y1[x],y2[x],..},x]

Παράδειγμα 1

Θεωρούμε και επιλύουμε το παρακάτω γραμμικό σύστημα που αποτελείται από δύο ΣΔΕ μιας $2^{ης}$ και μιας $1^{ης}$ τάξης:

```
In[1]:= eq1 = y1''[x] - y2[x] + x == 0; eq2 = y2'[x] + y1[x] == 0;
        solution = DSolve[{eq1, eq2}, {y1[x], y2[x]}, x] // FullSimplify
```

Out[2]=

$$\Big\{\Big\{y2[x] \to \frac{1}{6}\, e^{-\left(\frac{3}{2}+(-1)^{2/3}\right)x}\Big(2\, e^{\frac{1}{2} i\sqrt{3}\,x}\,(3\, e^{x} x - C[1] + C[2] + C[3]) + e^{3x/2}\left(C[1] + 2C[2] - C[3] - i\sqrt{3}\,(C[1]+C[3])\right) + e^{\frac{3x}{2}+i\sqrt{3}\,x}\left(C[1] + 2C[2] - C[3] + i\sqrt{3}\,(C[1]+C[3])\right)\Big),$$

$$y1[x] \to \frac{1}{6}\, e^{-\left(\frac{3}{2}+(-1)^{2/3}\right)x}\Big(-2\, e^{\frac{1}{2} i\sqrt{3}\,x}\,(3\, e^{x} + C[1] - C[2] - C[3]) + e^{\frac{3x}{2}+i\sqrt{3}\,x}\left(C[1] - C[2] - i\sqrt{3}\,(C[1]+C[2]) + 2C[3]\right) + e^{3x/2}\left(C[1] - C[2] + i\sqrt{3}\,(C[1]+C[2]) + 2C[3]\right)\Big)\Big\}\Big\}$$

Το αποτέλεσμα της DSolve είναι μία λίστα με ένα μόνο στοιχείο, αλλά με δύο σύμβολα (→) αντικατάστασης, ένα για την κάθε συνάρτηση-λύση. Έτσι, η μεταφορά των δύο λύσεων στις συναρτήσεις yy1 και yy2 του χρήστη γίνεται με τον παρακάτω τρόπο

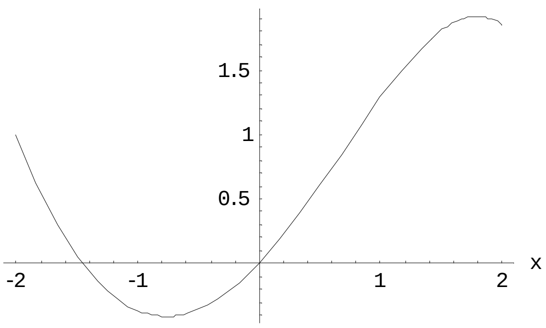
```
In[3]:= yy1 = y1[x] /. solution[[1]] /. {C[1] → c1, C[2] → c2, C[3] → c3}
        yy2 = y2[x] /. solution[[1]] /. {C[1] → c1, C[2] → c2, C[3] → c3}
```

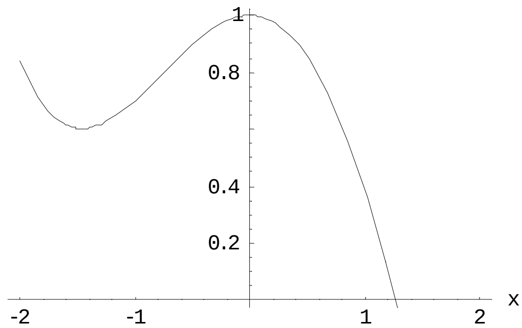
όπου c_1, c_2, c_3 οι τρεις αυθαίρετες σταθερές σύμφωνα με την παραπάνω αντικατάσταση. Αν στο παραπάνω σύστημα, εισαγάγουμε μια νέα άγνωστη συνάρτηση $y_3 = y_1'$ θα πάρουμε ένα σύστημα τριών εξισώσεων 1ης τάξης με τρεις άγνωστες συναρτήσεις

$$\frac{dy_1}{dx} = y_3 \qquad \frac{dy_2}{dx} = -y_1 \qquad \frac{dy_3}{dx} = y_2 - x \tag{10}$$

Επιλύνοντας το παραπάνω σύστημα, θα πάρουμε τις παραπάνω λύσεις για τις y_1 και y_2 ενώ για την y_3 θα έχουμε $y_3 \equiv dy_1/dx$, όπως την ορίσαμε. Τα αποτελέσματα της DSolve για το σύστημα (10) δίνονται σε διαφορετική μορφή (αλλά ισοδύναμη) με αυτά του Out[2]. Και στις δύο περιπτώσεις οι λύσεις μοιάζουν μιγαδικές ενώ, με κάποια προσοχή, παρατηρούμε ότι, είναι πραγματικές και μπορούμε να πάρουμε τη γραφική τους παράσταση για συγκεκριμένες τιμές των τριών αυθαίρετων σταθερών :

```
In[13]:= c1 = 1; c2 = 1; c3 = 1;
```





Αν και το MATHEMATICA δεν έχει περιορισμούς ως προς τον τρόπο γραφής του συστήματος (πχ το (9) ή το (10)) ο τρόπος γραφής (10) είναι πιο εύχρηστος και έτσι αναφέρεται στην βιβλιογραφία :

Ένα σύστημα ΣΔΕ (n-τάξης), με άγνωστες συναρτήσεις τις $y_1, y_2, \ldots, y_n$, περιγράφεται από τις σχέσεις

$$y_i' = \frac{dy_i}{dx} = f(y_1, y_2, \ldots, y_n, x) \qquad i = 1, \ldots, n \tag{11}$$

και η γενική του λύση είναι της μορφής

$$y_i(x) = y_i(x; c_1, c_2, ..., c_n) \qquad (12)$$

όπου $c_1, c_2, ..., c_n$ αυθαίρετες σταθερές.

Αν το σύστημα (11) είναι της μορφής

$$y_i'(x) = \mathbf{A}(x) y_i + \mathbf{B}(x) \qquad (13)$$

όπου **A** ένας $n \times n$ πίνακας με στοιχεία συναρτήσεις του x και **B** πίνακας στήλη με n-στοιχεία, επίσης συναρτήσεις του x, το σύστημα ονομάζεται **γραμμικό** και, *γενικά*, έχει αναλυτικές λύσεις[6]. Αν ένα σύστημα δεν είναι της μορφής (13) ονομάζεται **μη-γραμμικό** και, *γενικά*, δεν έχει αναλυτικές λύσεις.

Οι δυνατότητες του MATHEMATICA έχουν ως εξής[7]

α) Επιλύει κάθε τάξης γραμμικό σύστημα με σταθερούς συντελεστές (δηλαδή όταν οι πίνακες **A** και **B** της (12) δεν εξαρτώνται από την ανεξάρτητη μεταβλητή x)

β) Μπορεί να λύσει τις περισσότερες περιπτώσεις γραμμικών συστημάτων 2^{ης} τάξης (με συντελεστές συναρτήσεις του x)

γ) Μπορεί να λύσει κάποιες ιδικές μορφές μη-γραμμικών συστημάτων.

Όπως και στις ΣΔΕ, για συγκεκριμένες τιμές των αυθαίρετων σταθερών, παίρνουμε μερικές λύσεις που εντάσσονται στο σύνολο λύσεων (12). Παρόμοια, συγκεκριμένες τιμές για τις αυθαίρετες σταθερές προκύπτουν όταν το σύστημα (11) οφείλει να πληροί ένα σύνολο αρχικών (ή συνοριακών) συνθηκών. Ένα πλήρες σύνολο αρχικών συνθηκών θα είναι της μορφής

$$y_i(x_0) = a_i \qquad i = 1,..,n \qquad (14)$$

όπου a_i συγκεκριμένες τιμές των άγνωστων συναρτήσεων y_i για $x = x_0$.

Παράδειγμα 2. Γραμμικό σύστημα 2x2.

$$y_1' = y_2 \qquad y_2' = -y_1 + \sin(x) \qquad (15)$$

```
In[1]:= seq = {y1'[x] == y2[x], y2'[x] == -y1[x] + Sin[x]}

Out[1]= {y1'[x] == y2[x], y2'[x] == Sin[x] - y1[x]}
```

α) Γενική λύση

```
In[2]:= solution = DSolve[seq, {y1[x], y2[x]}, x] // FullSimplify
```

Out[2]= $\left\{\left\{\text{y1}[x] \to \left(-\frac{x}{2} + C[1]\right)\text{Cos}[x] + C[2]\,\text{Sin}[x],\ \text{y2}[x] \to \frac{1}{2}\left((-1 + 2C[2])\,\text{Cos}[x] + (x - 2C[1])\,\text{Sin}[x]\right)\right\}\right\}$

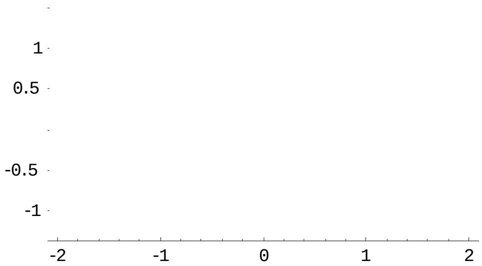
β) Μερική λύση με αρχικές συνθήκες $y_1(0)=1,\ y_2(0)=0\ \ (x_0=0)$

```
In[3]:= partialsol = DSolve[{seq, y1[0] == 1, y2[0] == 0}, {y1[x], y2[x]}, x] // FullSimplify
```

{{ [ ] 1/2 (-(-) [ ] [ ]) [ ] 1/2 (-) n[x]}}

γ) Γραφικές παραστάσεις $y_1(x)$ και $y_2(x)$

```
In[4]:= Plot[{partialsol[[1, 1, 2]], partialsol[[1, 2, 2]]},
        {x, -2, 2},
        PlotStyle→ {Dashing[{0.01}], Dashing[{0.02}]},
        Frame→ True]
```

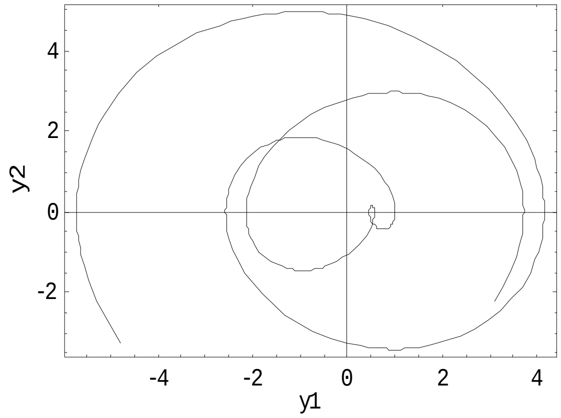


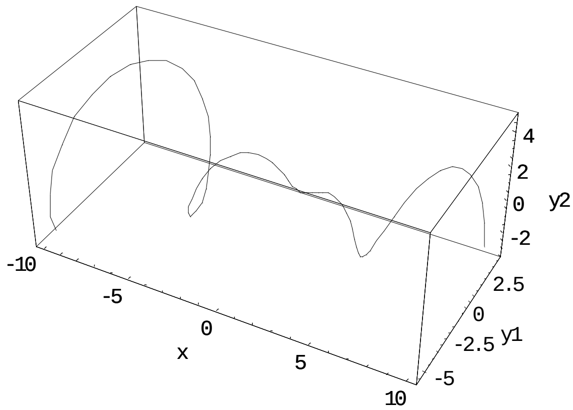
[6] Για την μεθοδολογία επίλυσης και τις ιδιότητες των λύσεων βλ. [1][8]
[7] Βλ Help του MATHEMATICA και [3].

δ) Παραμετρικές καμπύλες ($y_1(x)$, $y_2(x)$) και (x, $y_1(x)$, $y_2(x)$)

```
In[5]:= ParametricPlot[{partialsol[[1, 1, 2]], partialsol[[1, 2, 2]]}, {x, -10, 10},
         Frame → True, FrameLabel → {"y1", "y2"}]

In[6]:= ParametricPlot3D[{x, partialsol[[1, 1, 2]], partialsol[[1, 2, 2]]}, {x, -10, 10},
         AxesLabel → {"x", "y1", "y2"}]
```





Η παραμετρική καμπύλη (x,$y_1(x)$,$y_2(x)$) ονομάζεται **ολοκληρωτική καμπύλη** και ο χώρος xy_1y_2 "**επεκταμένος χώρος φάσεων**". Οι παραπάνω ορισμοί επεκτείνονται αντίστοιχα για συστήματα μεγαλύτερης τάξης. Έτσι το σύνολο αρχικών συνθηκών (14) αντιστοιχεί σε ένα σημείο του παραπάνω χώρου. Αν οι συνθήκες των θεωρημάτων ύπαρξης και μοναδικότητας των λύσεων (βλ.[8]) ικανοποιούνται τότε από κάθε σημείο του παραπάνω χώρου περνάει μία και μόνο μία ολοκληρωτική καμπύλη ή, αλλιώς, οι μερικές λύσεις δεν τέμνονται στον επεκταμένο χώρο φάσης.

Ιδιαίτερο ενδιαφέρον παρουσιάζουν τα **αυτόνομα συστήματα** για τα οποία οι συναρτήσεις f_i στον ορισμό (11) δεν εξαρτώνται από την ανεξάρτητη μεταβλητή x. Τότε ο χώρος $y_1y_2...y_n$ ονομάζεται **χώρος των φάσεων** και οι καμπύλες ($y_1(x)$,$y_2(x)$,...,$y_n(x)$) **φασικές τροχιές**. Υπό τις συνθήκες των θεωρημάτων ύπαρξης και μοναδικότητας των λύσεων, *οι φασικές τροχιές δεν πρέπει να τέμνονται*. Για συστήματα 2ης τάξης (2×2), οι φασικές καμπύλες, πάνω στον επίπεδο χώρο των φάσεων y_1y_2, ακολουθούν μια τοπολογία που χαρακτηρίζει το συγκεκριμένο σύστημα και αποκαλύπτει την γενικότερη συμπεριφορά του (βλ [1],[8]).

Παράδειγμα 3. Θεωρούμε το αυτόνομο σύστημα

$$y_1' = -y_1 - y_2 \qquad y_2' = y_1 - y_2 \tag{16}$$

Η λύση του συστήματος, θεωρώντας τις «αυθαίρετες» αρχικές συνθήκες $y_1(0)=y_{10}$ και $y_2(0)=y_{20}$ θα είναι

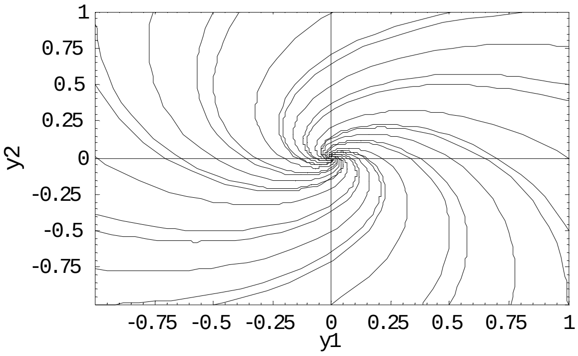
```
In[1]:= solution = DSolve[{y1'[x] == -y1[x] - y2[x], y2'[x] == y1[x] - y2[x], y1[0] == y10, y2[0] == y20}, {y1[x], y2[x]}, x];
        yy1 = y1[x] /. solution[[1]]
        yy2 = y2[x] /. solution[[1]]

Out[2]= e^-x (y10 Cos[x] - y20 Sin[x])

Out[3]= e^-x (y20 Cos[x] + y10 Sin[x])
```

Δημιουργούμε έναν πίνακα από φασικές καμπύλες (yy1,yy2) για διάφορες τιμές των (y10,y20) και σχηματίζουμε το φασικό διάγραμμα:

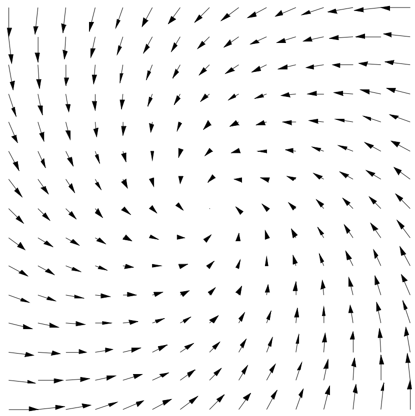
```
In[4]:= soltable =
         Table[{yy1 /. {y10 → i, y20 → j},
           yy2 /. {y10 → i, y20 → j}}, {i, -1, 1, 0.5},
          {j, -1, 1, 0.5}];
        soltable = Flatten[soltable, 1];
        ParametricPlot[Evaluate[soltable], {x, -2, 5},
         PlotRange → {{-1, 1}, {-1, 1}}, Frame → True,
         FrameLabel → {"y1", "y2"}]
```



Παρατηρούμε ότι όλες οι φασικές τροχιές έχουν μια σπειροειδή μορφή με κέντρο το (0,0). Όμως οι φασικές τροχιές έχουν και μια ορισμένη κατεύθυνση (δυναμική) που δείχνει την εξέλιξή τους καθώς η ανεξάρτητη μεταβλητή x αυξάνεται[8]. Για $x=0$ οι τροχιές ξεκινούν από τα σημεία των αρχικών συνθηκών τους και, καθώς το x αυξάνεται, τείνουν προς το σημείο (0,0). Αυτή η ασυμπτωτική συμπεριφορά των λύσεων είναι γενική όπως φαίνεται από την γενική λύση, για την οποία $(y_1(x),y_2(x))\to(0,0)$, καθώς $x\to\infty$, για κάθε ζεύγος τιμών των αυθαίρετων σταθερών. Η δυναμική ροή στο χώρο των φάσεων φαίνεται παραστατικά αν ζωγραφίσουμε το διανυσματικό πεδίο του συστήματος

$$\vec{V}=(V_1,V_2)=(y_1',y_2')=(f_1(y_1,y_2),f_2(y_1,y_2)) \qquad (17)$$

```
In[30]:= << Graphics`PlotField`
         V1 = -y1 - y2;
         V2 = y1 - y2;
         PlotVectorField[{V1, V2},
          {y1, -1, 1}, {y2, -1, 1}]
```



Οι φασικές τροχιές οφείλουν να εφάπτονται στα παραπάνω διανύσματα και να ακολουθούν την κατεύθυνση αυτών, καθώς το x αυξάνεται. Το σημείο (0,0) φαίνεται να είναι ένα κρίσιμο σημείο για το διανυσματικό πεδίο. Πράγματι οι y_1' και y_2' μηδενίζονται και η προφανής λύση $y_1=0,\ y_2=0$ αποτελεί μια λύση ισορροπίας (βλ. [1][8]). Η μορφή του φασικού πορτρέτου συνδέεται με το *είδος* (πραγματικές, μιγαδικές, μηδενικές κλπ) των ιδιοτιμών του πίνακα **A** του συστήματος. (βλ [8]).

Κλείνοντας την παράγραφο αυτή, θα τονίσουμε και πάλι ότι τα μη γραμμικά συστήματα *γενικά* <u>δεν έχουν</u> αναλυτικές λύσεις, πχ

```
In[1]:= eq1 = y1'[x] == y1[x] + y1[x] y2[x];
        eq2 = y2'[x] == y1[x] - y2[x];
        DSolve[{eq1, eq2}, {y1[x], y2[x]}, x]

Out[3]= DSolve[{y1′[x] == y1[x] + y1[x] y2[x], y2′[x] == y1[x] - y2[x]},
         {y1[x], y2[x]}, x]
```

Στις περιπτώσεις αυτές καταφεύγουμε στην αριθμητική επίλυση των εξισώσεων που παρουσιάζεται στο δεύτερο μέρος.

[8] Στη Φυσική και στη Δυναμική, γενικότερα, συνήθως, η ανεξάρτητη μεταβλητή είναι ο χρόνος.

§6. Διαφορικές Εξισώσεις με Μερικές Παραγώγους (ΔΕΜΠ)

Οι ΔΕΜΠ αποτελούν εξισώσεις στις οποίες η άγνωστη συνάρτηση (έστω z) εξαρτάται από *δύο* μεταβλητές[9] (έστω x και y) και οι εξίσωση περιέχει *μερικές παραγώγους* της συνάρτησης ως προς τις μεταβλητές αυτές:

$$f(x,y,z,\frac{\partial z}{\partial x},\frac{\partial z}{\partial y},\frac{\partial^2 z}{\partial x^2},\frac{\partial^2 z}{\partial y^2},\frac{\partial^2 z}{\partial x\partial y},....)=0 \tag{18}$$

Η μεγαλύτερη τάξη παραγώγου που παρουσιάζεται στην εξίσωση ονομάζεται και *τάξη* της εξίσωσης. Μια λύση για την (19) θα είναι της μορφής

$$(\alpha)\quad z=f(x,y) \qquad \eta' \qquad (\beta)\quad f(x,y,z)=0 \tag{19}$$

δηλαδή μια επιφάνεια στον χώρο xyz η οποία εκφράζεται ή με τη λυμένη μορφή (19)α ή την πεπλεγμένη μορφή (19)β. Η παραπάνω λύση (αν υπάρχει) εν γένει δεν είναι μοναδική. Όπως στις ΣΔΕ η γενική λύση περιέχει αυθαίρετες σταθερές, στις n-τάξης ΔΕΜΠ, η γενική λύση περιέχει n **αυθαίρετες συναρτήσεις**.

Η επίλυση ΔΕΜΠ είναι ένα πολύ δύσκολο πρόβλημα και σε ελάχιστες περιπτώσεις είναι δυνατή η εύρεση μιας γενικής λύσης. Σε κάποιες περιπτώσεις βρίσκεται μια απειρία λύσεων μέσω αυθαίρετων σταθερών, αλλά δεν αποτελούν τη γενική λύση. Επίσης, σε πολλές φυσικές εφαρμογές, θεωρούμε λύσεις συγκεκριμένης μορφής οπότε και το πρόβλημα κάπως απλουστεύεται. Στη παράγραφο αυτή θα παρουσιάσουμε μόνο μερικά βασικά στοιχεία για τον τρόπο που διαχειρίζεται το MATHEMATICA τις ΔΕΜΠ.

Έστω *z=z[x,y]* η άγνωστη συνάρτηση της εξίσωσης. Στην περίπτωση αυτή δεν μπορεί να χρησιμοποιηθεί ο συμβολισμός *z'* για την παράγωγο γιατί δεν δηλώνεται έτσι σε ποια από τις δύο μεταβλητές αναφέρεται. Για το λόγο αυτό καταφεύγουμε στον συμβολισμό με τον τελεστή **`D[,]`** που αναφέραμε στην παράγραφο §1. Η επίλυση μιας ΔΕΜΠ γίνεται και πάλι με την εντολή `DSolve` που συντάσσεται με τον ίδιο τρόπο όπως στις ΣΔΕ.

Παράδειγμα 1

Θεωρούμε την 1ης τάξης γραμμική ΔΕΜΠ $\frac{\partial z}{\partial x}-\frac{\partial z}{\partial y}=z \quad , \qquad z=z(x,y)$

```
In[1]:= Clear["Global`*"]
        eq = D[z[x, y], x] - D[z[x, y], y] == z[x, y]
        sol = DSolve[eq, z[x, y], {x, y}]

Out[2]= -z^(0,1)[x, y] + z^(1,0)[x, y] == z[x, y]

Out[3]= {{z[x, y] → e^x C[1][x + y]}}
```

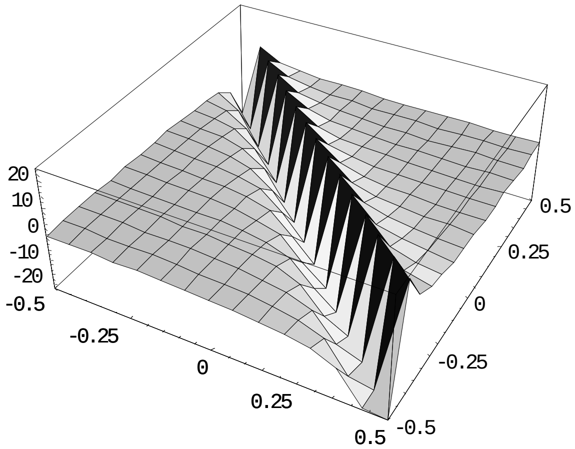
Στο `Out[2]` παρουσιάζεται η ΔΕΜΠ με διαφορετικό συμβολισμό, όπου οι δύο εκθέτες (μέσα στην παρένθεση) εκφράζουν την τάξη της παραγώγισης ως προς την 1η και 2η μεταβλητή αντίστοιχα. Το σύμβολο `C[1]`, που εμφανίζεται στην λύση αναφέρεται σε μια αυθαίρετη συνάρτηση της μεταβλητής που ακολουθεί, δηλαδή της *u=x+y*. Για μεγαλύτερης τάξης εξισώσεις οι επιπλέον αυθαίρετες συναρτήσεις συμβολίζονται σαν `C[2]`, `C[3]` κλπ, και πάντα ακολουθούνται από το όρισμά τους που είναι μια μεταβλητή *u=u(x,y)*. Η διαχείριση της λύσης μπορεί να γίνει όπως στο παρακάτω παράδειγμα:

[9] Αναφερόμαστε σε δύο ανεξάρτητες μεταβλητές για λόγους απλότητας. Φυσικά μπορούμε να θεωρήσουμε συναρτήσεις περισσοτέρων μεταβλητών.

```
In[4]:= (*Εισαγωγή της λύσης στην μεταβλητή z1 με ταυτόχρονη
        αντικατάσταση του προστατευόμενου συμβόλου C[1] με το f
        (αυθαίρετη συνάρτηση) *)
     z1 = z[x, y] /. sol[[1]] /. {C[1] → f};
     Print["Γενική λύση z=", z1]
      (*Ορίζουμε μια συγκεκριμένη συνάρτηση στη θέση της f για να
        πάρουμε μια μερική λύση*)
     f = 1/#&;   (* f(u)=1/u *)
     Print["Μια μερική λύση η z=", z1]
     Plot3D[z1, {x, -0.5, 0.5}, {y, -0.5, 0.5}]
```

Γενική λύση $z=e^{x} f[x+y]$

Μια μερική λύση η $z=\frac{e^{x}}{x+y}$



<u>Παρατήρηση</u>. Στην περίπτωση μιας ΔΕΜΠ, η DSolve δεν μπορεί να διαχειριστεί προβλήματα αρχικών-συνοριακών τιμών.

Ασκήσεις

1. Να βρεθεί η γενική λύση της $y'=y(x+1)$ και να σχεδιαστούν μερικές λύσεις για διάφορες τιμές της αυθαίρετης σταθεράς.

2. Να βρεθεί η γενική λύση των ΣΔΕ
α) $y'=\kappa\, y\, (\lambda-y)$, κ,λ σταθερές παράμετροι
β) $y'=(x+1)/y$
γ) $y'=(x^3+y^3)/(xy^2)$

3. Να βρεθεί η γενική λύση των ΣΔΕ
α) $y'+x^2y+x^2y^3=0$ (Bernoulli)
β) $y'=-k\, y^2+m\, (2x^2-1)y-(x^4-x^2-2x)$ για α) $k=1,m=1$ β) $k=1,\ m\neq 1$ γ) $k\neq 1,m=1$ (Riccati)
γ) $y'=(y+xy^2)/x$ (Πλήρης με πολλαπλασιαστή Euler)
δ) $y=xy'^2+y'^2$ (Lagrange)

4. Για τις παρακάτω γραμμικές εξισώσεις της μορφής $y'+g(x)y+h(x)=0$,
α) $y'+y-x=0$
β) $y'+y-\sin(x)=0$
γ) $y'+y/\sqrt{x}-3x=0$
δ) $y'+x\, y-1=0$
να βρεθούν οι γενικές λύσεις χρησιμοποιώντας α) τη DSolve και β) τον τύπο

$$y=\left(c-\int h(x)e^{\int g(x)dx}dx\right)e^{-\int g(x)dx}$$

και να συγκριθούν.

5. Να βρεθεί η γενική λύση της $y'+\sin(y')=x$ (βλ. [1,σελ.49])

6. Να λυθεί η ΣΔΕ $y'=|y|$ (βλ. [1,σελ 9])

7. Να βρεθεί η γενική λύση της

$$y'=\sqrt{x+y+1}-1$$

και να σχεδιαστούν οι λύσεις για c=1,0 και –1 (βλ. [1,σελ 23]

8. Να λυθεί η $y'=x^3\sqrt{y}$ και να σχεδιαστούν οι μερικές λύσεις που παιρνούν από τα σημεία α) (1,0) β) (1,1) γ) (1,-1). Η y=0 αποτελεί μια προφανή λύση (βλ [1, σελ 8]). Προκύπτει από τη DSolve;

9. Να λυθούν τα προβλήματα αρχικών τιμών και να σχεδιαστούν οι λύσεις
α) $y'^2-x(x+y)y'+x^3y=0,\ y(1)=1$
β) $y'=-(x-y)^2,\ y(0)=\sqrt{3}$

10. Για την ΣΔΕ $y''+\sqrt{x}=0$ α) να βρεθεί η γενική λύση, β) Να βρεθούν οι λύσεις για τις οποίες $y(2)=1$ και να σχεδιαστούν μερικές από αυτές γ) να βρεθεί η μερική λύση για τις συνθήκες $y(2)=1$, $y'(2)=0$.

11. Να βρεθεί με τη DSolve και να σχεδιαστεί η λύση της εξίσωσης

$$x^2y''+xy'+x^2y=0,\qquad y(0)=1$$

Αφού η παραπάνω εξίσωση είναι 2ης τάξης, γιατί αρκεί μόνο η αρχική συνθήκη $y(0)=1$ για να πάρουμε μία και μοναδική μερική λύση;

12. Θεωρούμε την εξίσωση του αρμονικού ταλαντωτή με αντίσταση και εξωτερική διέγερση

$$\ddot{x} = -\omega^2 x - \underbrace{b\dot{x}}_{\alpha\nu\tau\iota\sigma\tau\alpha\sigma\eta} + \underbrace{a\sin t}_{\varepsilon\xi\omega\tau\varepsilon\rho\iota\kappa\eta\,\delta\iota\varepsilon\gamma.}$$

Αφήνοντας τον ταλαντωτή με ταχύτητα 0 στη θέση x_0 να μελετήσετε τις ταλαντώσεις $x=x(t)$ για διάφορες τιμές των παραμέτρων b,a,ω και x_0.

13. Να βρεθεί (χωρίς/με τη χρήση της `FullSimplify`) η γενική λύση της ΣΔΕ Euler (βλ [1, σελ. 177])

$$x^3 y''' + kx^2 y'' - 2xy' = 2x - 4$$

για k=0, 1, 2, και για οποιοδήποτε k ως παράμετρο.

14. Για το γραμμικό σύστημα ΣΔΕ, με ανεξάρτητη μεταβλητή το χρόνο t,

$$y_1' = y_2 \quad , \quad y_2' = -y_1 + t$$

α) Να βρεθεί η γενική λύση β) Να βρεθεί η μερική λύση για αρχικές συνθήκες $y_1(0)=1$, $y_2(0)=0$ και να σχεδιαστούν τα $y_1=y_1(t)$, $y_2=y_2(t)$ καθώς και η ολοκληρωτική καμπύλη $(t,y_1(t),y_2(t))$.

15. Για τα αυτόνομα γραμμικά συστήματα με σταθερούς συντελεστές

$$\begin{pmatrix} y'_1 \\ y'_2 \end{pmatrix} = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \end{pmatrix}$$

Να βρεθεί η γενική λύση, να υπολογιστούν οι ιδιοτιμές λ_1,λ_2 του πίνακα (a_{ij}), να σχεδιαστεί το διανυσματικό πεδίο του συστήματος και να δοθούν ενδεικτικές φασικές τροχιές για τις παρακάτω τιμές παραμέτρων

a	b	c	d	λ_1	λ_2
0	1	1	1		
1	1	1	1		
-1	1	1	-1		
1	1	-2	1		
1	1	-3	-1		

16. Έστω το σύστημα ΣΔΕ

$$\dot{x} = -y + x(1 - x^2 - y^2) \quad , \qquad \dot{y} = x + y(1 - x^2 - y^2)$$

α) Να μετασχηματιστεί το παραπάνω σύστημα σε πολικές συντεταγμένες (r,θ) (με το mathematica) και στη συνέχεια να βρεθεί η γενική λύση $r=r(t)$, $\theta=\theta(t)$ η οποία να μετασχηματιστεί σε καρτεσιανές συντεταγμένες $x=x(t)$, $y=y(t)$.
β) να βρεθούν οι μερικές λύσεις για $r=1/2$, $\theta=0$ (ή $x=1/2$, $y=0$) και $r=2$, $\theta=0$ (ή $x=2$, $y=0$) και να σχεδιαστούν οι φασικές τροχιές. Παρατηρήστε ότι οι τροχιές αυτές (αλλά και οποιαδήποτε άλλη) καταλήγει σε έναν οριακό κύκλο (βλ. [1,σελ. 209][13, σελ. 79])

17. Έστω ένα καρτεσιανό σύστημα συντεταγμένων (x,y,z) σε ένα σημείο Ο της επιφάνειας της γης με τον άξονα z προς το ζενίθ, τον άξονα y προς την ανατολή και τον άξονα x προς το νότο. Θεωρούμε ομογενές το πεδίο βαρύτητας και αφήνουμε ένα σώμα να πέσει από ύψος h, δηλαδή από το σημείο $x=y=0$, $z=h$. Λόγω της περιστροφής της γης (με γωνιακή ταχύτητα ω), στο μη αδρανειακό σύστημα ενός παρατηρητή, που βρίσκεται στο σημείο Ο, η κίνηση του σώματος θα περιγράφεται από τις εξισώσεις (βλ [1,σελ. 181][13])

$$\ddot{x} = 2\omega\sin(\varphi)\dot{y} \quad , \quad \ddot{y} = -2\omega\sin(\varphi)\dot{x} - 2\omega\cos(\varphi)\dot{z} \quad , \quad \ddot{z} = -g + 2\omega\cos(\varphi)\dot{y}$$

όπου φ το γεωγραφικό πλάτος. Να βρεθεί η λύση του παραπάνω προβλήματος αρχικών τιμών και να σχεδιαστεί η τροχιά $(x(t),y(t),z(t))$ της πτώσης, όπως την αντιλαμβάνεται ο μη αδρανειακός παρατηρητής (θεωρήστε κάποιες ενδεικτικές τιμές για το ύψος h και για το γεωγραφικό πλάτος. Επίσης θεωρήστε μεγάλες τιμές για το ω ($\approx 1000\cdot\omega_{\Gamma\eta\varsigma}$)). Μπορεί να βρεθεί με το MATHEMATICA η γενική λύση του παραπάνω συστήματος;

ΜΕΡΟΣ 2^{ο} : ΑΡΙΘΜΗΤΙΚΗ ΜΕΘΟΔΟΣ

§1. Βασικές έννοιες της Αριθμητικής Ολοκλήρωσης

Έστω μια διαφορική εξίσωση 1^{ης} τάξης, της μορφής

$$\frac{dy}{dx} = f(x,y) \tag{1}$$

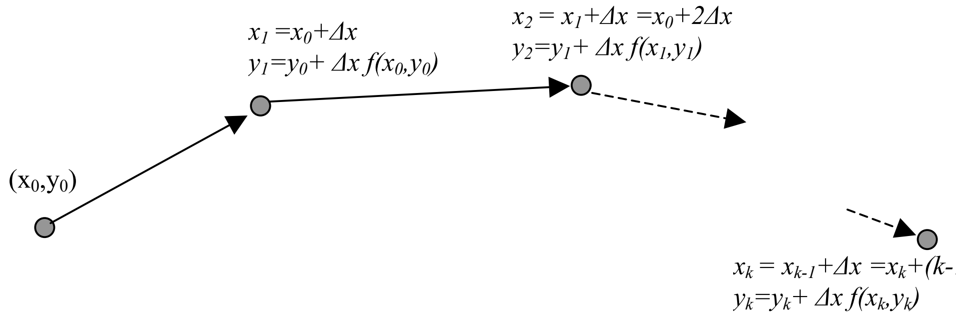
με άγνωστη συνάρτηση την $y=y(x)$. Η άμεση αριθμητική πληροφορία που μας παρέχει η (1) είναι η κλίση της συνάρτησης (δηλαδή η τιμή της παραγώγου dy/dx) σε κάθε σημείο του επιπέδου Oxy. Για το σημείο $x=x_0$, $y=y_0=y(x_0)$, από τον ορισμό της παραγώγου, έχουμε την ακόλουθη προσέγγιση (**Προσέγγιση Euler**) :

$$\left.\frac{dy}{dx}\right|_{x=x_0} = \lim_{\Delta x\to 0}\frac{y(x_0+\Delta x)-y(x_0)}{\Delta x} \simeq \frac{y(x_0+\Delta x)-y(x)}{\Delta x} \quad (\Delta x \ll 1) \tag{2}$$

ή

$$y(x_0+\Delta x) \simeq y_0 + \Delta x\, f(x_0,y_0) \tag{3}$$

Από την σχέση (3) προκύπτει ότι αν γνωρίζουμε την τιμή y_0 της άγνωστης συνάρτησης σε ένα σημείο $x=x_0$ τότε μπορούμε να υπολογίσουμε προσεγγιστικά, σε ένα γειτονικό σημείο $x_1 =x_0+\Delta x$, την τιμή της συνάρτησης $y_1=y(x_1)$. Ακολουθώντας, μια επαναληπτική διαδικασία, όπως φαίνεται στο παρακάτω σχήμα, παίρνουμε ένα σύνολο σημείων (x_k,y_k) τα οποία αποτελούν μια προσεγγιστική δειγματοληψία για την άγνωστη συνάρτηση $y=y(x)$.



Σχήμα 1 Επαναληπτική διαδικασία προσέγγισης της λύσης

Για την πραγματοποίηση της παραπάνω επαναληπτικής διαδικασίας, απαιτείται η "γνώση" ενός αρχικού σημείου. Όμως, από οποιοδήποτε σημείο (x_0,y_0) και αν ξεκινήσουμε, η παραπάνω επαναληπτική διαδικασία, και για όσο ορίζεται η συνάρτηση $f(x,y)$ στο πραγματικό χώρο, θα μας δώσει ένα σύνολο σημείων

$$A(x_0,y_0) = \{\,(x_k,y_k)\in R^2, k=0,1,...,N\} \tag{4}$$

με **βήμα** $\Delta x=x_k-x_{k-1}$ (σταθερό)

Σύμφωνα, με αυτά που αναφέραμε στο 1ο μέρος ,§3, κάθε σημείο (x_0,y_0) αποτελεί μια αρχική συνθήκη, και για κάθε αρχική συνθήκη αντιστοιχεί μια μοναδική μερική λύση της διαφορικής εξίσωσης[10]. Ουσιαστικά, δηλώνοντας το αρχικό σημείο (x_0,y_0), ορίζουμε πλέον ένα πρόβλημα αρχικών τιμών, που η λύση του $y=y(x)$ δεν περιέχει αυθαίρετες σταθερές.Το σύνολο των σημείων (4) αποτελεί μια προσεγγιστική αριθμητική παράσταση της μερικής λύσης του προβλήματος αρχικών τιμών και ονομάζεται **αριθμητική λύση** ενώ η επαναληπτική διαδικασία που ακολουθείται ονομάζεται **αριθμητική ολοκλήρωση**.

Παρατήρηση 1. Μια αριθμητική λύση αναφέρεται σε μια συγκεκριμένη μερική λύση της διαφορικής εξίσωσης. Ο όρος «αριθμητική λύση μιας διαφορικής εξίσωσης» δεν είναι ακριβής , αντίθετα θα πρέπει να αναφερόμαστε σε «αριθμητική λύση ενός προβλήματος αρχικών τιμών».

Παρατήρηση 2. Η αναλυτική λύση μιας διαφορικής εξίσωσης , η ο οποία, γενικά, περιγράφει κάποιο φυσικό σύστημα, μας αποκαλύπτει πλήρως τη συμπεριφορά του συστήματος, τα κρίσιμα και ουσιώδη σημεία του. Αντίθετα, ακολουθώντας μόνο την αριθμητική ολοκλήρωση, χωρίς κάποιες θεωρητικές ενδείξεις για την συμπεριφορά των λύσεων, γίνεται μια τυφλή εξερεύνηση του χώρου των αρχικών συνθηκών, η διάσταση του οποίου αυξάνεται με την τάξη της διαφορικής εξίσωσης ή το πλήθος των διαφορικών εξισώσεων του συστήματος.

Παρατήρηση 3. Σε παλαιότερα βιβλία, η αριθμητική ολοκλήρωση συνιστούνταν σε περιπτώσεις όπου η αναλυτική λύση μιας διαφορικής εξίσωσης ήταν αρκετά περίπλοκη (πιθανών και με ειδικές συναρτήσεις) οπότε και η σχεδίαση και η μελέτη κάποιων μερικών λύσεων ήταν μια κοπιώδης διαδικασία. Σήμερα δεν συντρέχει τέτοιος λόγος, δεδομένου ότι ο συμβολικός προγραμματισμός (πχ το MATHEMATICA) αντιμετωπίζει σχεδόν όλες αυτές τις περιπτώσεις . Όμως, σήμερα, και περισσότερο από κάθε άλλη φορά, η αριθμητική ολοκλήρωση είναι απαραίτητη στα περισσότερα προβλήματα αφού, όπως τονίσαμε στο πρώτο μέρος, γενικά οι διαφορικές εξισώσεις δεν έχουν αναλυτικές λύσεις. Οι ΔΕ (n>1) που λύνονται αναλυτικά αποτελούν εξαιρέσεις, οι οποίες αντιστοιχούν σε εκτενείς απλουστεύσεις κάποιου πολύπλοκου συστήματος (π.χ. ο αρμονικός ταλαντωτής).

Μια αριθμητική λύση αποτελεί προσέγγιση της πραγματικής λύσης. Η αξιοπιστία της προσέγγισης με τη μέθοδο Euler, παρουσιάζεται με τα παρακάτω παραδείγματα. Τα αποτελέσματα προκύπτουν από το πρόγραμμα **NEULER**, στο οποίο εισάγουμε την ΔΕ και τις απαραίτητες παραμέτρους όπως δηλώνονται από τα σχόλια του προγράμματος.

Παράδειγμα 1.

Θεωρούμε την γραμμική διαφορική εξίσωση 1ης τάξης $y'=y-x$. Επιλύουμε την εξίσωση αναλυτικά και αριθμητικά με το **NEULER** και παίρνουμε τα παρακάτω αποτελέσματα:

Πρόβλημα Αρχ.Τιμών : $y'=-x+y$, $x0=0$ $y(x0)=\frac{1}{2}$

Αναλυτική Λύση : $y=1-\frac{e^x}{2}+x$

Αριθμητική λύση , βήμα $\Delta x=0.1$ Σημεία N=30.

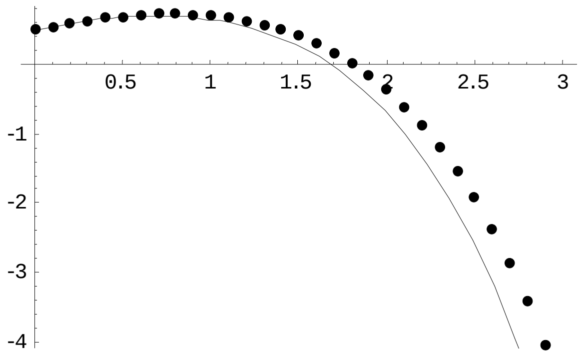
Διάστημα ολοκλήρωσης από x=0 εως3

$\left\{\left\{0, \frac{1}{2}\right\}, \{0.1, 0.55\}, \{0.2, 0.595\}, \ll 26 \gg, \{2.9, -4.03155\}, \{3., -4.7247\}\right\}$

Σχεδιάζοντας την αναλυτική λύση και τα σημεία της αριθμητικής λύσης, στο επίπεδο (x,y), παίρνουμε το σχήμα 2.

[10] Θεωρούμε ότι, οι συνθήκες των θεωρημάτων ύπαρξης και μοναδικότητας ικανοποιούνται.

```
(*NEuler : Αναλυτική και Αριθμιτική λύση Euler *)
(*Διαφορική εξίσωση y'=f(x,y) και αρχικές συνθήκες*)
f[x_, y_] := y - x; eq = y'[x] == y[x] - x;
x0 = 0; y0 = 1/2;
Print["Πρόβλημα Αρχ.Τιμών :  y'=", f[x, y], " ,    x0=", x0, "  y(x0)=", y0]
xmax = 3; (*integration and plot interval (0,xmax)*)
(*---- Αναλυτική Επίλυση ----- *)
DSolve[{eq, y[x0] == y0}, y[x], x] // Simplify;
yanalytic = y[x] /. %[[1]]; Print["Αναλυτική Λύση : y=", yanalytic]
gr1 = Plot[yanalytic, {x, 0, xmax}, DisplayFunction → Identity];
(*----- Αριθμητική προσέγγιση Euler -----*)
Clear[y]; dx = 0.05;  (*βήμα*)
dat = {{x0, y0}}; (*λίστα για την αποθήκευση των σημείων (xk,yk)*)
n = xmax / dx;              (*Αριθμός σημείων*)
(*Το (x,y) αντιπροσωπεύει το αρχικό σημείο και το (x1,y1) το επόμενο *)
x = x0; y = y0; (*Αρχή*)
For[i = 0, i < n, i++,          (*επαναληπτική διαδικασία n φορές*)
   x1 = x + dx;  y1 = y + dx * f[x, y]; (*τύπος Euler*)
   x = x1; y = y1; (*το νέο σημείο γίνεται αρχή για την επόμενη επανάληψη*)
   AppendTo[dat, {x, y}];]  (*Εισαγωγή του σημείου στη λίστα*)
gr2 = ListPlot[dat, PlotStyle → PointSize[0.01], DisplayFunction → Identity];
Print["Αριθμητική λύση , βήμα Δx=", dx, "  ", Short[dat, 3]] (*μερικά από τα σημεία*)
Show[{gr1, gr2}, DisplayFunction → $DisplayFunction]
Null
```



Σχήμα 2 Αναλυτική (συνεχής γραμμή) και αριθμητική λύση (σημεία)

Η αριθμητική λύση, ενώ αρχικά ακολουθεί σωστά την αναλυτική (πραγματική) λύση, στη συνέχεια παρουσιάζει μια αυξανόμενη απόκλιση. Η απόκλιση αυτή ονομάζεται **συστηματικό σφάλμα** της αριθμητικής μεθόδου και οφείλεται στο γεγονός ότι η παράγωγος $dy/dx=f(x,y)$ στο διάστημα (x_k,x_{k+1}), θεωρήθηκε σταθερή και ίση με $(y_{k+1}-y_k)/\Delta x$. Μια καλύτερη προσέγγιση θα μπορούσε να επιτευχθεί μειώνοντας το βήμα ολοκλήρωσης, γεγονός όμως που συνεπάγεται τις ακόλουθες επιπτώσεις:

- Απαιτείται μεγαλύτερος αριθμός επαναλήψεων για την κάλυψη της ίδιας περιοχής (*0,xmax*) και έτσι απαιτείται μεγαλύτερος υπολογιστικός χρόνος (CPU time).

- Οι διάφορες ποσότητες αντιπροσωπεύονται με πραγματικούς αριθμούς πεπερασμένης ακρίβειας. Το λάθος ή η στρογγυλοποίηση του τελευταίου δεκαδικού (*roundoff error*), μπορεί να προκαλεί αυξανόμενα σφάλματα μετά από κάθε επανάληψη. Έτσι όσο αυξάνεται ο αριθμός των επαναλήψεων, το παραπάνω λάθος μπορεί να γίνει πολύ σημαντικό.

Επίσης, οι αποκλίσεις των αριθμητικών λύσεων εξαρτώνται και από την συμπεριφορά των πραγματικών λύσεων και μπορεί να παρουσιάζουν απότομες μεταβολές όπως στο παρακάτω παράδειγμα.

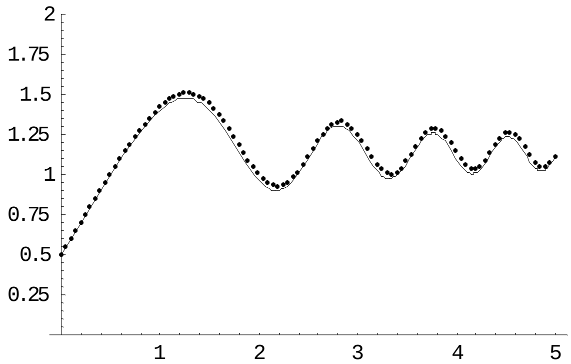
Παράδειγμα 2

Θεωρούμε τη ΣΔΕ $y' = \cos(x^2)$, η οποία λύνεται άμεσα με μία ολοκλήρωση[11], και τις αρχικές συνθήκες y(0)=1/2. Χρησιμοποιώντας το **NEULER** παίρνουμε τα παρακάτω αποτελέσματα

Πρόβλημα Αρχ.Τιμών : $y' = \mathrm{Cos}[x^2]$, $x0 = 0$ $y(x0) = \frac{1}{2}$

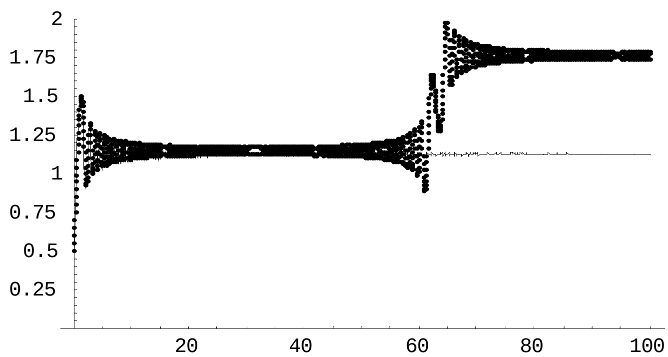
Αναλυτική Λύση : $y = \frac{1}{2}\left(1 + \sqrt{2\pi}\ \mathrm{FresnelC}\left[\sqrt{\frac{2}{\pi}}\ x\right]\right)$

Αριθμητική λύση , βήμα $\Delta x = 0.05$ Σημεία N=100.



Παρατηρούμε και πάλι μια μικρή απόκλιση από την αναλυτική λύση. Γενικά όμως, η αριθμητική λύση φαίνεται αρκετά ακριβής για το διάστημα ολοκλήρωσης $0<x<5$. Επισημαίνουμε ότι, η αναλυτική λύση εκτελεί ταλαντώσεις, με μειούμενο πλάτος, γύρω από την τιμή $y_L=1$ καθώς $x\to\infty$. Όμως η συμπεριφορά της αριθμητικής λύσης είναι πολύ διαφορετική για μεγαλύτερο διάστημα ολοκλήρωσης, όπως φαίνεται από το σχήμα 3. Τα σφάλματα οδηγούν σε αριθμητική λύση μακριά της πραγματικής.

Διάστημα ολοκλήρωσης από x=0 εως 100



Σχήμα 3 Αριθμητική λύση Euler για το παράδειγμα 2 και για μεγάλο διάστημα ολοκλήρωσης. Για $x>10$, τα σφάλματα είναι ουσιώδη.

Η μέθοδος Euler αποτελεί την πιο απλή μέθοδο αριθμητικής ολοκλήρωσης. Η ανάπτυξη μεθόδων αριθμητικής επίλυσης διαφορικών εξισώσεων αποτελεί αντικείμενο της **Αριθμητικής Ανάλυσης**. Βασικός στόχος, της αριθμητικής ανάλυσης, στο πεδίο αυτό, είναι η κατασκευή μεθόδων ολοκλήρωσης με βασικά χαρακτηριστικά α) Μικρότερα και ελεγχόμενα σφάλματα β) Μικρότερη πολυπλοκότητα, δηλαδή λιγότερες επαναλήψεις και, γενικά, λιγότερες αριθμητικές πράξεις γ) Εύκολη και αποδοτική προσαρμογή της μεθόδου για όσο το δυνατό περισσότερες κατηγορίες ΣΔΕ.

Οι πιο γνωστές, σήμερα μέθοδοι, είναι οι Runge-Kutta, Adams, Bulirsch-Stoer, Μέθοδος σειρών κ.α. (βλ. [11][12])

§2. Επίλυση ΣΔΕ και συστημάτων ΣΔΕ με την NDSolve

Όπως τονίστηκε στην προηγούμενη παράγραφο, μια αριθμητική λύση αναφέρεται στη λύση ενός προβλήματος αρχικών τιμών και για ένα πεπερασμένο διάστημα της ανεξάρτητης μεταβλητής (*διάστημα ολοκλήρωσης*). Η αριθμητική επίλυση στο MATHEMATICA γίνεται με την εντολή `NDSolve`, η

[11] Το ολοκλήρωμα δίνεται από την ειδική συνάρτηση τύπου Fresnel, βλ [9][10]

οποία συντάσσεται όπως και η `DSolve` (για ένα πρόβλημα αρχικών τιμών) αλλά μαζί με την δήλωση της ανεξάρτητης μεταβλητής δηλώνεται και το διάστημα ολοκλήρωσης

`NDSolve`[*Πρόβλημα Αρχικών/συνοριακών τιμών*, *Άγνωστη συνάρτηση*, *Ανεξ. Μεταβλητή&Διάστημα Ολοκλήρωσης*]

`NDSolve[{`Διαφορική εξ., Συμπληρωματικές συνθήκες**`}, y,{x,xmin,xmax}]`**

Το αποτέλεσμα της `NDSolve` δεν δίνεται σαν ένα σύνολο δεδομένων (*x,y*)[12], αλλά σαν μια «**συνάρτηση παρεμβολής**» (*Interpolating function*) του MATHEMATICA (βλ [2][3]), μέσω της οποίας μπορούμε να πάρουμε τα παραπάνω ζεύγη τιμών για οποιοδήποτε x στο διάστημα (*xmin,xmax*)[13].

<u>Παρατήρηση</u>: Στη θέση της άγνωστης συνάρτησης δηλώσαμε μια “καθαρή” (pure) συνάρτηση `y` και όχι την *y[x]*. Αν και μπορούμε να χρησιμοποιήσουμε μια συνάρτηση `y[x]`, στη συνέχεια δεν έχουμε την άμεση δυνατότητα να πάρουμε τις παραγώγους της λύσης, δηλαδή δεν μπορούμε να χρησιμοποιήσουμε τον τελεστή `D[]`, όπως στην αναλυτική περίπτωση. Για το λόγο αυτό, η έξοδος της `NDSolve` συνιστάται να δίνεται από μία “καθαρή” συνάρτηση.

Παράδειγμα 1.

Θεωρούμε τον ταλαντωτή με απόσβεση, ο οποίος περιγράφεται από την παρακάτω ΣΔΕ (βλ. [1, σελ. 166], με t την ανεξάρτητη μεταβλητή (χρόνος) και $y=y(t)$ την άγνωστη θέση του σώματος,

$$\ddot{y} + b\dot{y} + \omega^2 y = 0 \qquad (5)$$

όπου b ο συντελεστής απόσβεσης και ω η κυκλική συχνότητα του ταλαντωτή. Η γενική λύση της (5) είναι η

```
[   '[t] + b y'[t] + w^2y[t] == 0, y[t], t]
```

Out[1]= $\left\{\left\{y[t] \to e^{\frac{1}{2} t\left(-b-\sqrt{b^2-4w^2}\right)} C[1] + e^{\frac{1}{2} t\left(-b+\sqrt{b^2-4w^2}\right)} C[2]\right\}\right\}$

ενώ για τις αρχικές συνθήκες y*(0)=1,y'(0)=1* έχουμε

```
In[2]:= DSolve[{y''[t] + b y'[t] + w^2y[t] == 0, y[0] == 1, y'[0] == 0}, y[t], t]
```

Out[2]= $\left\{\left\{y[t] \to \frac{1}{2\sqrt{b^2-4w^2}}\left(-b\, e^{\frac{1}{2} t\left(-b-\sqrt{b^2-4w^2}\right)} + b\, e^{\frac{1}{2} t\left(-b+\sqrt{b^2-4w^2}\right)} + e^{\frac{1}{2} t\left(-b-\sqrt{b^2-4w^2}\right)}\sqrt{b^2-4w^2} + e^{\frac{1}{2} t\left(-b+\sqrt{b^2-4w^2}\right)}\sqrt{b^2-4w^2}\right)\right\}\right\}$

Η γενική λύση που βρίσκουμε παραπάνω περιγράφει πλήρως όλη τη συμπεριφορά του συστήματος. Η μερική λύση αναφέρεται στη λύση που ξεκινάει από το σημείο του χώρου αρχικών συνθηκών (*t,y,y'*)=(0,1,0) αλλά, αναφέρεται επίσης για κάθε τιμή των παραμέτρων b και ω. Για την εύρεση αριθμητικής λύσης, πρέπει να καθοριστούν συγκεκριμένες αριθμητικές τιμές για τις παραμέτρους αυτές, π.χ.

```
In[3]:= w = 1; b = 1/2;
        NDSolve[{y''[t] + by'[t] + w^2y[t] == 0, y[0] == 1, y'[0] == 0}, y, {t, 0, 10}]

Out[4]= {{y → InterpolatingFunction[{{0., 10.}}, <>]}}
```

Την συνάρτηση παρεμβολής, που παίρνουμε ως έξοδο, την αναθέτουμε σε ένα σύμβολο για να μπορούμε να αναφερόμαστε στη λύση αυτή σε επόμενες γραμμές του προγράμματος. Η συνάρτηση

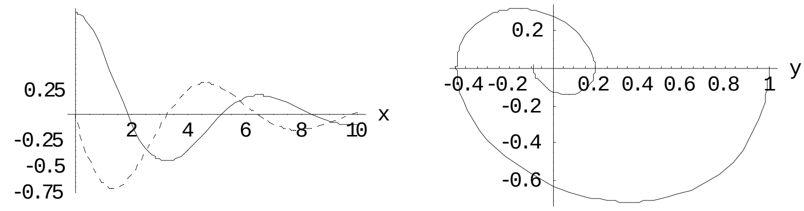
[12] Στην περίπτωση συστήματος έχουμε τις (ν+1)-άδες (x,y_1,y_2...).

[13] Μπορούμε να αναφερθούμε και σε τιμές έξω από το διάστημα ολοκλήρωσης αλλά, γενικά, οι τιμές που θα πάρουμε μπορεί να απέχουν πολύ από την πραγματικότητα.

αυτή μπορεί να μας δώσει τις τιμές αυτής, καθώς και των παραγώγων της, για κάθε *t* στο διάστημα ολοκλήρωσης και να σχεδιαστεί άμεσα.

```
y = y /. Out[4][[1]];
Print["-------Αποτελέσματα----------"]
t = 0; Print["t=", t, "  y=", y[t], "  y'=", y'[t]]
t = 1; Print["t=", t, "  y=", y[t], "  y'=", y'[t]]
t = 2.7; Print["t=", t, " y=", y[t], "  y'=", y'[t], "  κ.λ.π."]
gr1 = Plot[{y[t], y'[t]}, {t, 0, 10}, PlotStyle → {{}, Dashing[{0.02}]},
  AxesLabel → {"x", "y(t),y'(t)"}, DisplayFunction → Identity]
gr2 = ParametricPlot[{y[t], y'[t]}, {t, 0, 10}, AxesLabel → {"y", "y'"}, DisplayFunction → Identity]
Show[GraphicsArray[{gr1, gr2}], DisplayFunction → $DisplayFunction]
```

```
t=2.7 y=-0.373832  y'=-0.264626  κ.λ.π.
```



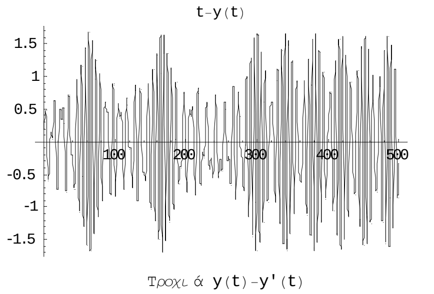
Παραπάνω συμβολίσαμε την συνάρτησή μας σαν *y* (όπως αναφέρεται στη ΣΔΕ) και, επίσης, χρησιμοποιήσαμε το σύμβολο *t* για να του δώσουμε κάποιες τιμές. Έτσι, αν εκτελέσουμε ξανά τις γραμμές εντολών `In[3]` θα πάρουμε μήνυμα λάθους, αφού τα σύμβολα *y* και *t* δεν είναι πλέον έγκυρα. Είναι καλύτερα λοιπόν να συμβολίζουμε τη λύση και την ανεξάρτητη μεταβλητή με διαφορετικά σύμβολα. Επίσης χρησιμοποιήσαμε στην `NDSolve` το όρισμα `y` και όχι το `y[t]`. Στη δεύτερη περίπτωση δεν θα μπορούσαμε να αναπαράγουμε τις τιμές για το `y'[t].` Αν συγκρίνουμε την παραπάνω λύση με την αντίστοιχη αναλυτική, θα δούμε ότι συμπίπτουν με πάρα πολύ καλή ακρίβεια (βλέπε σχετικά, στην §3) και στα παραπάνω διαγράμματα δεν μπορούν να εντοπιστούν διαφορές.

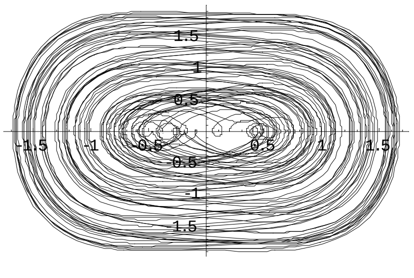
Παράδειγμα 2. Θεωρούμε τον μη γραμμικό ταλαντωτή (τύπου Duffing βλ [7])

$$\frac{d^2y}{dt^2}+2\gamma\frac{dy}{dt}+\alpha y+\beta y^3=f\cos(\omega t) \tag{6}$$

Η εξίσωση αυτή δεν επιδέχεται αναλυτική λύση[14] και χρησιμοποιούμε το παρακάτω πρόγραμμα για την αριθμητική της επίλυση

```
ω = 1; α = 0; β = 1; γ = 0.0; f = 0.2; (*Παράμετροι*)
y0 = 0.2; yd0 = 0; (*Αρχικές συνθήκες*)
tmax = 500; (*Διάστημα Ολοκλήρωσης από 0 έως tmax*)
eq = y''[t] + α y[t] + 2 γ y'[t] + β y[t]^3 == f Cos[ω t];
sol = NDSolve[{eq, y[0] == y0, y'[0] == yd0}, y, {t, 0, tmax},
  MaxSteps → ∞]
(*Σχεδίαση λύσης και ολοκληρωτικής καμπύλης*)
Plot[y[t] /. sol[[1]], {t, 0, tmax}, PlotLabel → "t-y(t)",
 PlotPoints → 1000]
ParametricPlot[Evaluate[{y[t] /. sol[[1]], y'[t] /. sol[[1]]}],
 {t, 0, tmax}, PlotLabel → "Τροχιά y(t)-y'(t)",
 PlotPoints → 1000]
```





[14] πλην συγκεκριμένων τιμών των παραμέτρων.

Παρατηρούμε μια μη συνηθισμένη συμπεριφορά που μοιάζει με θόρυβο. Μια τέτοια συμπεριφορά μοιάζει με αυτή του παραδείγματος 2§2 η οποία οφείλονταν στα αριθμητικά σφάλματα. Στη περίπτωση αυτή η παραπάνω συμπεριφορά είναι ένα χαρακτηριστικό των λύσεων της εξίσωσης (6) και ονομάζεται **χαοτική συμπεριφορά**. Για την επίλυση χρησιμοποιήθηκε η επιλογή `MaxSteps` η οποία αναλύεται στην επόμενη παράγραφο.

Ανάλογα με τη σύνταξη της `DSolve` για την επίλυση συστημάτων, συντάσσεται και η `NDSolve`.

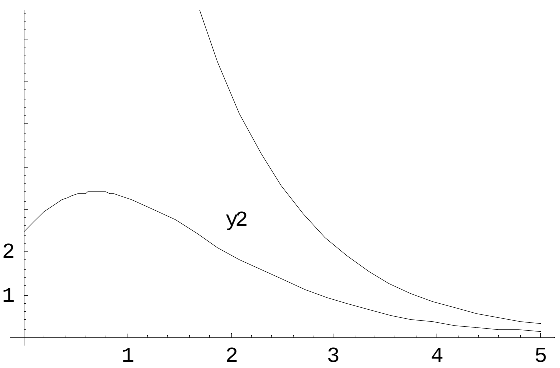
Παράδειγμα 3

Έστω το μη γραμμικό αυτόνομο σύστημα 2×2

$$y_1' = -y_1 + y_2^2/2 \qquad y_2' = -y_2 + y_1 y_2/2 \qquad (7)$$

με αρχικές συνθήκες $y_1(0)=4$ και $y_2(0)=1/4$. Θεωρούμε ανεξάρτητη μεταβλητή την x και το διάστημα ολοκλήρωσης (0,5) :

```
In[1]:= eq1 = y1'[x] == -y1[x] + y2[x]^2/2;
        eq2 = y2'[x] == -y2[x] + y1[x]*y2[x]/2;
        sol = NDSolve[{eq1, eq2, y1[0] == 4, y2[0] == 1/4}, {y1, y2}, {x, 0, 5}]
        Plot[{y1[x] /. sol[[1]], y2[x] /. sol[[1]]}, {x, 0, 5},
         Epilog -> Text["y1\n\n\n\n y2", {2, 0.4}]]
```



Η έξοδος της `NDSolve` είναι και πάλι μία λίστα με <u>ένα</u> στοιχείο το οποίο περιέχει τη λύση και για τις <u>δύο</u> άγνωστες συναρτήσεις. Για την παράσταση αυτών των συναρτήσεων μπορεί να χρησιμοποιηθούν σύμβολα του χρήστη πχ `q1=y1[x]/.sol[[1]]` και `q2=y2[x]/.sol[[1]]` ή συναρτήσεις του χρήστη `q1[x]=y1[x]/.sol[[1]]` και `q2[x]=y2[x]/.sol[[1]]` (βλ. 1ο μέρος, §2).

§3 – Επιλογές και έλεγχος της Αριθμητικής Ολοκλήρωσης

Η `NDSolve` χρησιμοποιεί κλασσικές μεθόδους αριθμητικής ολοκλήρωσης και συγκεκριμένα τις Runge-Kutta (τάξης 4.5) , Adam (τάξη 1-12) και Gear (τάξη 1-5)[15] . Οι παραπάνω μέθοδοι χρησιμοποιούν παραμέτρους οι οποίες καθορίζονται αυτόματα από την `NDSolve`. Υπάρχουν όμως και μερικές βασικές παράμετροι οι οποίες, αν και έχουν κάποιες προκαθορισμένες τιμές, μπορούν να αλλαχθούν από το χρήστη με βάση τις απαιτήσεις του συγκεκριμένου προβλήματος. Έτσι, συνολικά, η NDSolve συντάσσεται ως εξής

NDSolve[{equations}, {y_1,y_2,...},{x,xmin,xmax}, *Επιλογή1→ Νέα Τιμή*, *Επιλογή2→ Νέα Τιμή* κ.λ.π., **]**

[15] Για λεπτομέρειες σχετικά με τις παραπάνω μεθόδους βλ. βιβλιογραφία σχετική με θέματα αριθμητικής ανάλυσης πχ [11][12]

Οι διαθέσιμες Επιλογές και η χρήση τους είναι οι ακόλουθες

Method (Προεπιλεγμένη τιμή Automatic)

Επιλέγουμε ποια αριθμητική μέθοδο θα χρησιμοποιήσει η NDSolve. Οι δυνατές τιμές είναι

Method→RungeKutta, Method→Adams, Method→Gear, Method→Automatic.

Τα πλεονεκτήματα και μειονεκτήματα των παραπάνω μεθόδων έχουν να κάνουν με το συγκεκριμένο σύστημα προς επίλυση. Με την επιλογή Automatic το MATHEMATICA εναλλάσσει τις μεθόδους *Adams* και *Gear*, ανάλογα με τη συμπεριφορά της λύσης, για την επίτευξη μεγαλύτερης ακρίβειας. Για συνηθισμένες εφαρμογές στα πλαίσια του MATHEMATICA, η μέθοδος Automatic είναι πιο ακριβής αλλά η Runge-Kutta είναι πιο γρήγορη.

WorkingPrecision (Προεπιλεγμένη τιμή $MachinePrecision)

Είναι ο αριθμός των σημαντικών ψηφίων που χρησιμοποιούνται κατά τους εσωτερικούς υπολογισμούς. Ο αριθμός των σημαντικών ψηφίων που χρησιμοποιεί ο υπολογιστής εξαρτάται από το μήκος λέξης του επεξεργαστή του. Οι σημερινοί προσωπικοί υπολογιστές εκτελούν επεξεργασία στα 32-bit η οποία παρέχει τους πραγματικούς αριθμούς με ακρίβεια 16 σημαντικών ψηφίων. Η ακρίβεια αυτή αναγνωρίζεται από το MATHEMATICA με την τιμή $MachinePrecision. Αν θέσουμε WorkingPrecision<$MachinePrecision τότε δεν λαμβάνεται υπόψη η νέα τιμή. Αν θέσουμε WorkingPrecision>$MachinePrecision, τότε τα επιπλέον σημαντικά ψηφία παράγονται με ειδικό κώδικα του MATHEMATICA ο οποίος όμως έχει πολύ σημαντικό κόστος στον χρόνο των υπολογισμών. Με την αύξηση του WorkingPrecision μειώνονται τα σφάλματα στρογγυλοποίησης (*roundoff errors*).

AccuracyGoal (AG) και **PrecisionGoal** (PG)

(Προεπιλεγμένη τιμή= WorkingPrecision-10)

Καθορίζουν το απόλυτο και το σχετικό σφάλμα της λύσης για το κάθε βήμα ολοκλήρωσης και εκφράζονται με τον αριθμό των σημαντικών ψηφίων που επιθυμούμε. Έστω ότι η λύση y σε ένα σημείο x προσεγγίζεται, εσωτερικά, με μια σειρά συγκλινουσών επαναληπτικών διαδικασιών

1^{o} εσωτερικό βήμα $y^{(1)}$=15.123456789

2^{o} εσωτερικό βήμα $y^{(2)}$=15.256789452

3^{o} εσωτερικό βήμα $y^{(3)}$=15.234567890

4^{o} εσωτερικό βήμα $y^{(4)}$=15.234800445 κ.λ.π

Αν θέσουμε AccuracyGoal→3, τότε η παραπάνω διαδικασία θα σταματήσει στο 3^{o} βήμα αφού η $y^{(3)}$ προκύπτει, σε τρία (=AG) σημαντικά ψηφία, ίδια με την $y^{(2)}$. Αν θέσουμε PrecisionGoal→3 η διαδικασία θα πρέπει να σταματάει στο βήμα k για το οποίο ισχύει $|((y^{(k)}-y^{(k-1)})/y(k)|<10^{-PG}$, δηλαδή, για το παραπάνω παράδειγμα, στο βήμα 4. <u>Η NDSolve σταματάει τις εσωτερικές επαναλήψεις αν επιτευχθεί **έστω η μία** από τις παραπάνω συνθήκες.</u> Γι'αυτό πρέπει να δίνονται κατάλληλες τιμές **και για τις δύο** αυτές επιλογές. Συνήθως μας ενδιαφέρει η PrecisionGoal. Ετσι μπορούμε να ρυθμίζουμε μόνο αυτή τη παράμετρο αλλά θέτοντας ταυτόχρονα AccuracyGoal→∞ (επιλογή που δεν πληρείται ποτέ!)

Οι τιμές AG και PG πρέπει υποχρεωτικά να είναι μικρότερες του WorkingPrecision. Όσο οι τιμές AG και PG πλησιάζουν τη τιμή του WorkingPrecision η ακρίβεια της αριθμητικής ολοκλήρωσης αυξάνεται αλλά απαιτούνται περισσότερες εσωτερικές επαναλήψεις και το κόστος σε χρόνο υπολογισμών αυξάνεται σημαντικά.

<u>Προσοχή</u> : Οι τιμές των AC ή PG δεν εκφράζουν το πραγματικό σφάλμα μετά το τέλος της ολοκλήρωσης, αλλά σχετίζονται με το σφάλμα για το κάθε βήμα. Τα σφάλματα αυξάνουν σε κάθε βήμα ανάλογα με το συγκεκριμένο σύστημα και τη μορφή της συγκεκριμένης λύσης. Όμως, ένα μικρό σφάλμα στην αρχή προδικάζει και μικρότερα σφάλματα στο μέλλον.

MaxSteps (Προεπιλεγμένη τιμή=1000[16])

Κατά την αριθμητική ολοκλήρωση με την NDSolve, αποθηκεύονται στη μνήμη οι τιμές της λύσης ανά διαστήματα με βήμα Δx της ανεξάρτητης μεταβλητής και κατασκευάζεται από αυτά η συνάρτηση παρεμβολής που θα αποτελέσει τη τελική μορφή της λύσης. Ο αριθμός των βημάτων εξαρτάται τόσο από το πόσο μεγάλο είναι το διάστημα ολοκλήρωσης (*xmin,xmax*) αλλά και από τη μορφή της λύσης και τη σύγκλιση της επαναληπτικής διαδικασίας στην απαιτούμενη ακρίβεια (δηλαδή το *Δx* στις μεθόδους της NDSolve είναι μεταβλητό). Η τιμή MaxSteps καθορίζει το μέγιστο αριθμό βημάτων (ή δεδομένων) που μπορεί να εκτελέσει η NDSolve. Θα πρέπει να επισημάνουμε ότι, αύξηση του MaxStep σημαίνει παραγωγή περισσότερων σημείων και συνεπώς μεγαλύτερες απαιτήσεις σε μνήμη. Η επιλογή MaxSteps→∞, επιτρέπει στο MATHEMATICA να χρησιμοποιήσει όσα βήματα χρειάζονται για να επιτευχθεί η επιθυμητή ακρίβεια.

StartingStepSize (Προεπιλεγμένη τιμή=Automatic) και
MaxStepSize (Προεπιλεγμένη τιμή=∞):

Το StartingStepSize είναι το βήμα *Δx* με το οποίο ξεκινάει η ολοκλήρωση. Αν στο σημείο εκκίνησης δεν υπάρχει κοντά κάποιο ιδιόμορφο σημείο του προβλήματος, τότε η επιλογή Automatic είναι η βέλτιστη. Στην περιοχή μιας ιδιομορφίας, από όπου διέρχονται περισσότερες λύσεις, μπορεί να απαιτείται ένα πολύ μικρό βήμα εκκίνησης. Επειδή το βήμα *Δx* είναι μεταβλητό, μπορεί να αυξηθεί αρκετά σημαντικά, μειώνοντας έτσι το χρόνο υπολογισμών ενώ τα κριτήρια AG και PG ικανοποιούνται. Με την επιλογή MaxStepSize δηλώνουμε έναν περιορισμό για την μέγιστη τιμή την οποία μπορεί να πάρει το βήμα ολοκλήρωσης. Ο λόγος ύπαρξης αυτής της επιλογής είναι το γεγονός ότι, πολύ μεγάλα βήματα μπορεί να δημιουργήσουν σημαντικά σφάλματα τα οποία δεν γίνονται αντιληπτά από την αριθμητική μέθοδο.

InterpolationPrecision (Προεπιλεγμένη τιμή =WorkingPrecision)

Αναφέρεται στην ακρίβεια των ψηφίων που επιστρέφει η συνάρτηση παρεμβολής που εκφράζει τη λύση. Δεν σχετίζεται με την ακρίβεια της αριθμητικής ολοκλήρωσης.

Compiled (Προεπιλεγμένη τιμή=True) : Με την τιμή True, οι εξισώσεις και η μέθοδος αριθμητικής επίλυσης μετατρέπονται σε κώδικα μηχανής (μέσω compiler της C) και στη συνέχεια γίνεται η εκτέλεση της αριθμητικής επίλυσης. Με το τρόπο αυτό επιτυγχάνεται μεγάλη ταχύτητα. Γενικά η τιμή False πρέπει να επιλέγεται μόνο στην περίπτωση κάποιου προβλήματος κωδικοποίησης.

Η ακρίβεια της NDSolve καθώς και η χρήση των παραπάνω επιλογών εξετάζονται με τη βοήθεια του παρακάτω παραδείγματος.

Παράδειγμα 1

Εξετάζουμε την επίπεδη κίνηση (*x,y*) ενός πλανήτη γύρω από τον ήλιο ο οποίος βρίσκεται στο σημείο (0,0) (ηλιοκεντρικό σύστημα). Η μελέτη μας θα στηριχθεί στο **ΠΡΟΓΡΑΜΜΑ-4**. Αν *m, M* είναι η μάζα του πλανήτη και του Ήλιου αντίστοιχα, η κίνηση του πλανήτη θα περιγράφεται από την ΣΔΕ σε διανυσματική μορφή

$$m\frac{d^2\vec{r}}{dt}=\vec{F}=-G\frac{m\,M}{\vec{r}^{\,2}} \qquad (8)$$

όπου $\vec{r}=(x,y)$ η θέση του σώματος. Τα *x,y* (και φυσικά το *r*) είναι συναρτήσεις του χρόνου και μεταβάλλονται με βάση τη διαφορική εξίσωση (8). Για απλούστευση, χρησιμοποιούμε το σύστημα μονάδων για το οποίο $G{\cdot}M$=1 και *m*=1. Έτσι έχουμε[17]

[16] Για ΔΕΜΠ η προεπιλεγόμενη τιμή είναι 100

[17] Για την θεωρητική ανάλυση του προβλήματος και τους τύπους που χρησιμοποιούνται βλ. [??]

$$\vec{F} = -\frac{1}{r^2}\vec{e}_r\,, \qquad V = -\int \vec{F} d\vec{r} = -\int F dr = -\frac{1}{r} = -\frac{1}{\sqrt{x^2 + y^2}}$$

όπου V το δυναμικό, και οι διαφορικές εξισώσεις παίρνουν τη μορφή

$$\frac{d^2\vec{r}}{dt^2} = -\vec{\nabla} V = -\left(\frac{\partial V}{\partial x}\vec{i} + \frac{\partial V}{\partial y}\vec{j}\right) \Rightarrow \begin{array}{l} d^2 x/dt = \dfrac{-x}{(x^2+y^2)^{3/2}} \\ d^2 y/dt = \dfrac{-y}{(x^2+y^2)^{3/2}} \end{array} \tag{9}$$

Η κατασκευή των εξισώσεων (9) στο MATHEMATICA, γίνεται στον τομέα Α του Π4.

Είναι γνωστό, ότι το παραπάνω σύστημα ΣΔΕ έχει ως λύσεις τις «κωνικές τομές» [13] οι οποίες όμως δεν δίνονται στην κλειστή μορφή *x=x(t)* και *y=y(t)*. Έτσι η χρήση της `DSolve` δεν δίνει αποτελέσματα αλλά αντιμετωπίζει το σύστημα (9) σαν μη επιλύσιμο. Στο παράδειγμά μας θα ασχοληθούμε με κλειστές τροχιές (τροχιές που δεν φεύγουν στο άπειρο) και, ως γνωστό είναι ελλείψεις με μια εστία το (0,0) (Ήλιος).

Ποσοτικά στοιχεία, που περιγράφουν την επίπεδη ελλειπτική τροχιά του πλανήτη, είναι η ενέργεια και η στροφορμή ή η εκκεντρότητα, ο μεγάλος ημιάξονας, και η περίοδος [13]. Οι συναρτήσεις που δίνουν τα παραπάνω στοιχεία αναπτύσσονται στον τομέα Β του Π4.

Θα επιλύσουμε το σύστημα (9) αριθμητικά για συγκεκριμένες αρχικές συνθήκες οι οποίες εκφράζουν την αρχική θέση (`x0,y0`) και αρχική ταχύτητα (`vx0,vy0`). Η ολοκλήρωση γίνεται στο χρονικό διάστημα (`0,tmax`). Οι παραπάνω αρχικές συνθήκες εισάγονται στον τομέα Γ του προγράμματος. Στο τομέα Δ του προγράμματος υπολογίζονται και εκτυπώνονται τα στοιχεία της τροχιάς με βάση την αρχική θέση και ταχύτητα, που αντιστοιχούν σε `t`=0. Αφού γίνει η αριθμητική ολοκλήρωση, στον τομέα Ε, έχουμε τις συνιστώσες της θέσης και ταχύτητας σαν συναρτήσεις του χρόνου

```
x=x[t] , y=y[t],  vx=x'[t] , vy=y'[t]
```

και εκτυπώνουμε και πάλι τα στοιχεία της τροχιάς για `t=tmax`. Επίσης σχεδιάζουμε την τροχιά για την πρώτη περιστροφή της και για την τελευταία.

- Είναι γνωστό ότι τα παραπάνω στοιχεία της ελλειπτικής τροχιάς (Ενέργεια, στροφορμή κ.λ.π) **δεν μεταβάλλονται** με το χρόνο. Άρα η αρχική και τελική κατάσταση, όπως προκύπτει από το πρόγραμμα, θα πρέπει να είναι ακριβώς οι ίδιες.

Με βάση το παραπάνω χαρακτηριστικό της κίνησης θα εξετάσουμε τα αποτελέσματα της `NDSolve` ως προς την ακρίβειά της και πως αυτή διαμορφώνεται μεταβάλλοντας τις τιμές των επιλογών της.

Εκτελώντας το **ΠΡΟΓΡΑΜΜΑ-4** (όπως ακριβώς δίνεται), παίρνουμε τα αποτελέσματα του πίνακα **Ι**.

```
(*ΠΡΟΓΡΑΜΜΑ 4 *)
(*ΚΕΝΤΡΙΚΟ ΠΕΔΙΟ ΔΥΝΑΜΕΩΝ F = -1/r^2 *)
Clear["Global`*"]
(* (A) ---------ΣΥΣΤΗΜΑ------------------------------------------------ *)
F = -1/r^2;    (*Δύναμη*)
V = -∫ F ⅆr; (*Δυναμικό*)
r = √(x[t]^2 + y[t]^2); (*ακτίνα σε καρτεσιανές συντεταγμένες (ΚΣ) *)
eqx = x''[t] == -D[V, x[t]]; (*ΔΕ σε ΚΣ, 2 x2 2ης τάξης*)
eqy = y''[t] == -D[V, y[t]];

(* (B) ---Χρήσιμες συναρτήσεις για τα στοιχεία της τροχιάς------------*)
Energy[x_, y_, vx_, vy_] := 1/2 (vx^2 + vy^2) - 1/√(x^2 + y^2); (*Ενέργεια*)
AngMom[x_, y_, vx_, vy_] := x * vy - vx * y;              (*Στροφορμή*)
eccentricity[angmom_, energy_] := √(1 + 2 energy angmom^2);  (*εκκεντρότητα*)
SemimajorAx[angmom_, eccen_] := angmom^2/(1 - eccen^2);        (*Μεγάλος ημιάξονας*)
Period[semimajorax_] := 2 π √(semimajorax^3);          (*Περίοδος περιστροφής*)

(* (Γ) ---- Αρχικές συνθήκες -------------------------------------------*)
x0 = 1.0; y0 = 0; vx0 = 0; vy0 = 0.8;
tmax = 5;  (*Διάστημα Ολοκλήρωσης (0,tmax) *)

(* (Δ) ---- Αρχικά στοιχεία της τροχιάς -----------------------------*)
Print["Αρχικά τροχιακά στοιχεία για t=0 "]
ener = Energy[x0, y0, vx0, vy0]; angmom = AngMom[x0, y0, vx0, vy0];
ecc = eccentricity[angmom, ener];
semimajorax = SemimajorAx[angmom, ecc];
per = Period[semimajorax];
Print["Energy=", InputForm[ener]]
Print["Angular Momentum=", InputForm[angmom]]
Print["Eccentricity=", InputForm[ecc]];
Print["SemimajorAxis=", InputForm[semimajorax]]
Print["Period=", InputForm[per]];
(* (E) ---- Πρόβλημα αρχικών τιμών ΠΑΤ και αριθμητική επίλυση --------*)
PAT = {eqx, eqy, x[0] == x0, y[0] == y0, x'[0] == vx0, y'[0] == vy0};
sol = NDSolve[PAT, {x, y}, {t, 0, tmax}];
x = x /. sol[[1]];   (*ανάθεση αποτελέσματος-λύσης στα σύμβολα x,y*)
y = y /. sol[[1]];
(* (ΣΤ) -------------------- ΑΠΟΤΕΛΕΣΜΑΤΑ ΟΛΟΚΛΗΡΩΣΗΣ -----------------*)
Print["Τροχιά για την πρώτη περιστροφή  0<t<period"]
ParametricPlot[{x[t], y[t]}, {t, 0, per}, AxesLabel → {"x", "y"},
 PlotPoints → 1000]
Print["Τελικά Τροχιακά στοιχεία για t=tmax=", tmax, " ---"]
ener = Energy[x[tmax], y[tmax], x'[tmax], y'[tmax]];
angmom = AngMom[x[tmax], y[tmax], x'[tmax], y'[tmax]];
ecc = eccentricity[angmom, ener];
semimajorax = SemimajorAx[angmom, ecc];
per = Period[semimajorax];
Print["Energy=", InputForm[ener]]
Print["Angular Momentum=", InputForm[angmom]]
Print["Eccentricity=", InputForm[ecc]];
Print["SemimajorAxis=", InputForm[semimajorax]]
Print["Period=", InputForm[per]];
Print["Τροχιά για την τελευταία περιστροφή  (tmax-period)<t<tmax"]
t1 = tmax - per;
ParametricPlot[{x[t], y[t]}, {t, t1, tmax}, AxesLabel → {"x", "y"},
 PlotPoints → 1000]
```

Null

ΠΙΝΑΚΑΣ Ι (13 έτη)

Αρχικά τροχιακά στοιχεία για t=0

Energy=-0.6799999999999999

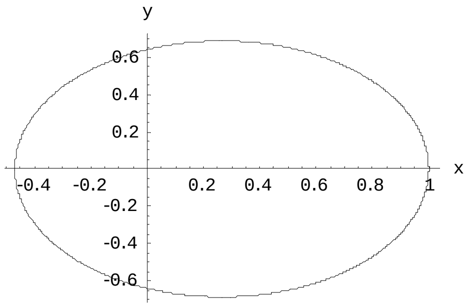
Angular Momentum=0.8

Eccentricity=0.35999999999999993

SemimajorAxis=0.735294117647059

Period=3.9616080528290407

Τροχιά για την πρώτη περιστροφή 0<t<period



Τελικά Τροχιακά στοιχεία για t=tmax=50 ---

Energy=-0.6799196830898645

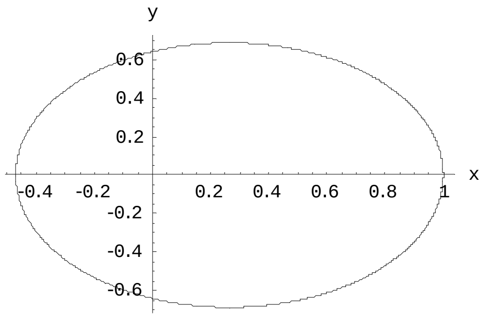
Angular Momentum=0.8000240577269115

Eccentricity=0.36007007849493583

SemimajorAxis=0.7353809757760982

Period=3.9623100332001884

Τροχιά για την τελευταία περιστροφή



Παρατηρώντας τα αρχικά τροχιακά στοιχεία στον πίνακα **I**, διαπιστώνουμε ότι η ενέργεια και η εκκεντρότητα, που θα έπρεπε να έχουν τις ακριβείς τιμές –0.68 και 0.36 αντίστοιχα, παρουσιάζονται με σφάλμα «στρογγυλοποίησης» το οποίο προέρχεται από τους υπολογισμούς που έγιναν με βάση τις τιμές των αρχικών συνθηκών. Τα σφάλματα αυτά αυξάνονται στην επαναληπτική διαδικασία της αριθμητικής ολοκλήρωσης αλλά, γενικά, μπορεί να γίνουν σημαντικά μόνο για μεγάλους χρόνους ολοκλήρωσης. Τα κυρίαρχα σφάλματα είναι τα *συστηματικά* σφάλματα που προέρχονται από τη μέθοδο ολοκλήρωσης.

Στο διάστημα (0,tmax) έχουμε tmax/Period≈13 περιστροφές (**έτη**). Η αρχική και τελική περιστροφή φαίνονται, πράγματι, να είναι οι ίδιες. Αν όμως παρατηρήσουμε τα τροχιακά στοιχεία, βλέπουμε μια σημαντική απόκλιση της τάξης των 4 με 5 σημαντικών ψηφίων σε σχέση με τα αρχικά. Αν μελετήσουμε την εξέλιξη της τροχιάς μετά από περίπου 100 έτη (δηλαδή θέσουμε *tmax=100*Period≈400*) θα πάρουμε από την NDSolve το παρακάτω μήνυμα[18]

```
NDSolve::mxst :
  Maximum number of 1000 steps reached at the point t == 61.54722333025309`.
```

που σημαίνει ότι η προεπιλεγμένη τιμή MaxSteps=1000 για τον αριθμό των βημάτων, δεν ήταν αρκετή για να καλύψει όλο το διάστημα ολοκλήρωσης (0,400) αλλά καλύφθηκε μόνο το διάστημα (0,61.5...). Απαιτείται λοιπόν μια αύξηση στην τιμή της επιλογής MaxSteps, η οποία πρέπει να είναι περίπου 6500[19]. Έτσι, συμπληρώνουμε την NDSolve με την επιλογή

NDSolve[*βασικά ορίσματα* , MaxSteps→ 6500**]**

Μπορούμε, επίσης, να θέσουμε MaxSteps→Infinity, για να μελετήσουμε τις τροχιές για πολύ μεγάλους χρόνους. Τα αποτελέσματα όμως μπορεί να καταλαμβάνουν μεγάλο αριθμό δεδομένων με συνέπεια τη πιθανή δημιουργία προβλημάτων στο υπολογιστικό σύστημα (πχ να επιβραδύνουν τη

[18] Τα υπόλοιπα μηνύματα προκύπτουν εξαιτίας του πρώτου που προκύπτει από την NDSolve

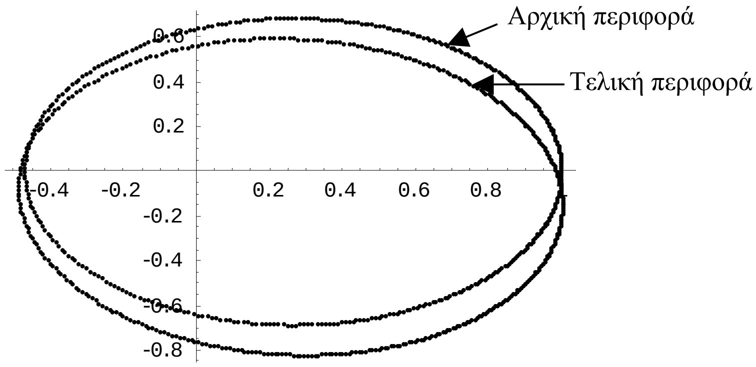
[19] Πρόχειρος υπολογισμός με απλή μέθοδο των τριών

λειτουργία του ή να σταματήσει η ανταπόκριση του MATHEMATICA). Αν εκτελέσουμε τώρα το πρόγραμμα για tmax=400 θα πάρουμε και πάλι τις ίδιες ελλείψεις αλλά το σφάλμα στα τροχιακά στοιχεία θα αυξηθεί.

Στο πίνακα **II** παρουσιάζονται τα αποτελέσματα της εκτέλεσης για tmax=100000 (ή 25000 έτη περίπου)[20]. Τα σφάλματα στα τροχιακά στοιχεία είναι πιο σημαντικά και παρατηρούνται και στην ελλειπτική τροχιά, η απόκλιση της οποίας, από την αρχική μορφή, είναι εμφανής (σχήμα 5)

ΠΙΝΑΚΑΣ **II** (25000 έτη)

Αρχικά τροχιακά στοιχεία για t=0	Τελικά Τροχιακά στοιχεία για t=tmax=100000 ---
Energy=-0.6799999999999999	Energy=-0.6638437291439974
Angular Momentum=0.8	Angular Momentum=0.8054608630650693
Eccentricity=0.3599999999999993	Eccentricity=0.3723438231655967
SemimajorAxis=0.735294117647059	SemimajorAxis=0.7531893095453837
Period=3.9616080528290407	Period=4.1071076783824125



Σχήμα 4 Οι περιφορές του πλανήτη κατά το 1ο και το 25000ο έτος, όπως προκύπτουν από την NDSolve.

Στις παραπάνω εκτελέσεις του προγράμματος, αναφερθήκαμε σε αρχικές συνθήκες που αντιστοιχούν σε ομαλές ελλειπτικές τροχιές μακριά από το κρίσιμο σημείο (0,0) των διαφορικών εξισώσεων. Γενικότερα, τα σφάλματα αυξάνουν σημαντικά όταν οι τροχιές είναι αρκετά περίπλοκες ή πλησιάζουν σε κρίσιμα σημεία. Για το σύστημα (9), μπορούμε να πλησιάσουμε στο σημείο (0,0) (τροχιά κοντά στον ήλιο) αν μειώσουμε, π.χ., την αρχική συνιστώσα της ταχύτητας vy0 στις αρχικές συνθήκες. Τότε, ουσιαστικά, αυξάνουμε την εκκεντρότητα της τροχιάς η οποία γίνεται μια στενή έλλειψη και πλησιάζει πολύ κοντά στο κέντρο του πεδίου δυνάμεων. Θεωρούμε λοιπόν, την εκτέλεση του προγράμματος (όπως στην αρχή) για tmax=50 αλλά θέτουμε vy0=0.02. Τα αποτελέσματα παρουσιάζονται στον πίνακα **III** και δείχνουν πολύ σημαντικά σφάλματα από την πρώτη κιόλας περιφορά (προσέξτε την πολύ στενή κλίμακα του άξονα *y*). Αντί της τελευταίας περιφοράς, παρουσιάζεται η εξέλιξη σε όλο το χρονικό διάστημα.

[20] Για την παραπάνω ολοκλήρωση τα δεδομένα καλύπτουν 300Mb μνήμης περίπου και ο υπολογισμός επιτυγχάνεται σε χρόνο 1min σε υπολογιστή Athlon2GHz, 512MbRam.

ΠΙΝΑΚΑΣ **III** (13 έτη, vy0=0.02)

Αρχικά τροχιακά στοιχεία για t=0

Energy=-0.9998

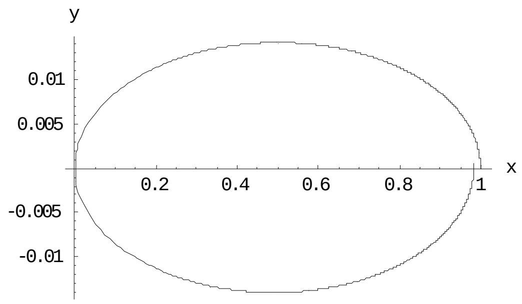
Angular Momentum=0.02

Eccentricity=0.9996

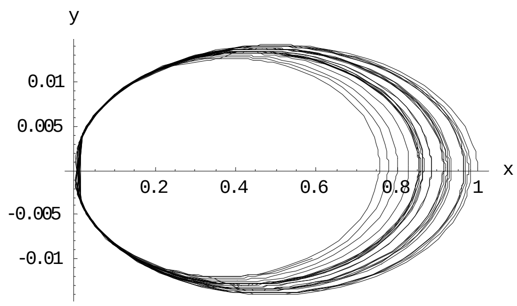
SemimajorAxis=0.5001000200040727

Period=2.22210806816738

Τροχιά για την πρώτη περιστροφή 0<t<period



Τροχιά για όλο το διάστημα ολοκλήρωσης



Τελικά Τροχιακά στοιχεία για t=tmax=50 ---

Energy=-1.3114539113158963

Angular Momentum=0.019941468158731584

Eccentricity=0.9994783483547306

SemimajorAxis=0.3812562497893873

Period=1.4791265267975986

Στην παραπάνω εκτέλεση, η NDSolve χρησιμοποιεί τις επιλογές ακρίβειας PG=AG=16-10=6. Για να βελτιώσουμε, λοιπόν την ακρίβεια της ολοκλήρωσης θα πρέπει να αυξήσουμε την παραπάνω τιμή, πχ :

NDSolve[*βασικά ορίσματα* , MaxSteps→∞, AccuracyGoal→10, PrecisionGoal→10**]**

Με τις παραπάνω τιμές επιτυγχάνουμε παρόμοια ακρίβεια με αυτή του πίνακα **I**.

Αν θέσουμε tmax=2000, παίρνουμε το μήνυμα

Δηλαδή, εξαιτίας της ολίσθησης της τροχιάς από τα αριθμητικά σφάλματα, η λύση πλησιάζει πολύ κοντά στο ιδιόμορφο σημείο (0,0) τη στιγμή t=1840, το βήμα ολοκλήρωσης γίνεται σχεδόν μηδέν, και η ολοκλήρωση τερματίζεται. Γενικά για να ξεπεραστούν τυχών ιδιομορφίες, απαιτείται πολύ μεγάλη ακρίβεια υπολογισμών, η οποία επιτυγχάνεται με το MATHEMATICA αν αυξήσουμε την τιμή της επιλογής WorkingPrecision. Έτσι, με τις επιλογές

NDSolve[*βασικά ορίσματα* , MaxSteps→∞, , WorkingPrecision→32,
AccuracyGoal→10, PrecisionGoal→10**]**

ο σκόπελος της παραπάνω ιδιομορφίας μπορεί να ξεπεραστεί αλλά, ίσως, μόνο προσωρινά.

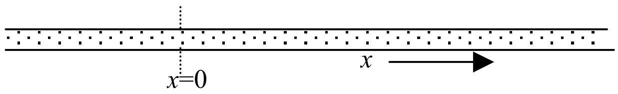
Συμπερασματικά, λοιπόν, οι δυνατότητες του MATHEMATICA (αλλά και των άλλων πακέτων συμβολικού προγραμματισμού) είναι κάπως περιορισμένες αλλά αυξάνονται με κάθε νέα έκδοση του προγράμματος.. Για πολύπλοκα συστήματα που απαιτούν μεγάλους χρόνους ολοκλήρωσης, χρησιμοποιούνται συνήθως οι κλασσικές γλώσσες προγραμματισμού με τις οποίες δίνεται πλήρης ελευθερία στην ανάπτυξη ειδικών αλγορίθμων, στην διαχείριση ειδικών περιπτώσεων και στην κατ' επιλογήν έξοδο των δεδομένων. Εξάλλου και κάθε πακέτο συμβολικού προγραμματισμού είναι γραμμένο σε μια τέτοια γλώσσα[21]. Γενικά, δεν υπάρχει δυνατότητα επαλήθευσης μιας αριθμητικής λύσης και, για το λόγο αυτόν, απαιτείται ιδιαίτερη προσοχή.

§4. Αριθμητικές λύσεις εξισώσεων με Μερικές Παραγώγους[22].

Το MATHEMATICA, προς το παρόν, έχει ενσωματωμένους αλγορίθμους που μπορούν να επιλύσουν λίγες κατηγορίες ΔΕΜΠ εφοδιασμένες με κατάλληλες αρχικές και συνοριακές συνθήκες. Η επίλυσή τους γίνεται και πάλι με την εντολή `NDSolve`, η σύνταξη της οποίας ακολουθεί παρόμοιους κανόνες με αυτούς που έχουν προαναφερθεί στις παραγράφους §6(1^{o} μέρος) και §2, §3 (2^{o} μέρος). Θα παρουσιάσουμε τη λειτουργία της `NDSolve` με το παρακάτω παράδειγμα

Παράδειγμα 1.

Θεωρούμε την ροή ενός ρευστού σε έναν ευθύγραμμο σωλήνα κατά μήκος του άξονα *x*.



Η συνάρτηση κατανομής της πυκνότητας $\rho=\rho(x,y,z,t)$ του ρευστού οφείλει να ικανοποιεί την εξίσωση συνέχειας

$$\frac{\partial\rho}{\partial t}+\nabla(\rho\cdot\vec{v})=0 \qquad \text{η'σε μια διασταση} \quad \frac{\partial\rho}{\partial t}+v\frac{\partial\rho}{\partial x}+\rho\frac{\partial v}{\partial x}=0 \tag{10}$$

όπου $\vec{v}$ το πεδίο ταχυτήτων του ρευστού, και για το μονοδιάστατο πρόβλημα είναι $\rho=\rho(t,x)$ και $v=v(x,t)$. Με δεδομένο το πεδίο ταχυτήτων, η εξίσωση (10) αποτελεί μια ΔΕΜΠ $1^{ης}$ τάξης με άγνωστη συνάρτηση την κατανομή πυκνότητας $\rho=\rho(t,x)$

Η αριθμητική επίλυση της (10) θα πρέπει να αναφερθεί σε μια μερική λύση της η οποία οφείλει να ικανοποιεί ένα κατάλληλο σύνολο αρχικών και συνοριακών συνθηκών. Θα θεωρήσουμε τα ακόλουθα συμπληρωματικά στοιχεία στο πρόβλημα

Πεδίο ταχυτήτων $\vec{v}=\frac{1}{4}\cos(x)\vec{i}$ (ταχύτητες ανεξάρτητες του χρόνου)

Αρχικές και συνοριακές συνθήκες $\rho(0,\,x)\ =\ 1,\quad \rho(t,\ 0)\ =\ 1$

Διάστημα ολοκλήρωσης $0\le t\le 2\pi,\ 0\le x\le 2\pi$

Τα σωματίδια του ρευστού, ανάλογα με τη θέση *x* στην οποία βρίσκονται και για οποιαδήποτε χρονική στιγμή, οφείλουν να έχουν ταχύτητα ίση με *cos(x)/4*. Επίσης τη χρονική στιγμή t=0 το ρευστό είναι ομογενές, δηλαδή σε κάθε σημείο του σωλήνα η πυκνότητα είναι σταθερή και ίση με 1. Τέλος, κατά τη χρονική εξέλιξη του συστήματος, η πυκνότητα δεν μεταβάλλεται στο σημείο x=0.

Η λύση που θα προκύψει, για το παραπάνω πρόβλημα, είναι ένα σύνολο τιμών για την πυκνότητα *p* σε συγκεκριμένα σημεία ενός πλέγματος του επιπέδου Ο*xy*. Οι τιμές αυτές συνιστούν στο

[21] Γενικά, ο συμβολικός προγραμματισμός στηρίζεται στις δυνατότητες και οργάνωση που προσφέρει ο αντικειμενοστραφής προγραμματισμός. Παράδειγμα τέτοιων αντικειμένων είναι οι πραγματικοί αριθμοί με μεγαλύτερο αριθμό δεκαδικών ψηφίων από αυτόν που παρέχει ο επεξεργαστής. Στην περίπτωση αυτή, ακόμα και οι απλές αριθμητικές πράξεις, εκτελούνται μέσω ειδικών διαδικασιών που αναφέρονται στο συγκεκριμένο αντικείμενο (βλ. [14]),

[22] Λόγω της πολυπλοκότητας του προβλήματος δεν θα αναφερθούμε σε λεπτομερή ζητήματα. Κάθε κατηγορία ΔΕΜΠ παρουσιάζει τα εξειδικευμένα χαρακτηριστικά και, γενικά στην βιβλιογραφία, αντιμετωπίζονται ξεχωριστά. Οι δυσκολίες αυτές δεν επιτρέπουν στο MATHEMATICA να αντιμετωπίσει πολλά προβλήματα, αλλά οι κάποιες περιπτώσεις που μπορεί να διαχειριστεί είναι πολύ σημαντικές για την Φυσική (βλ. Help της MATHEMATICA).

MATHEMATICA μια συνάρτηση παρεμβολής δύο διαστάσεων (στο παρακάτω πρόγραμμα θέσαμε στη θέση του ρ το σύμβολο p).

```
In[1]:= v = Cos[x] / 4;
        eq = D[p[t, x], t] + p[t, x] D[v, x] + D[p[t, x], x] v == 0
        sol = NDSolve[{eq, p[0, x] == 1, p[t, 0] == 1}, p, {t, 0, 2 π}, {x, 0, 2 π},
          MaxSteps → 2000, StartingStepSize → 0.01]
```

Out[2]= $-\frac{1}{4}$ p[t, x] Sin[x] + $\frac{1}{4}$ Cos[x] $p^{(0,1)}$[t, x] + $p^{(1,0)}$[t, x] == 0

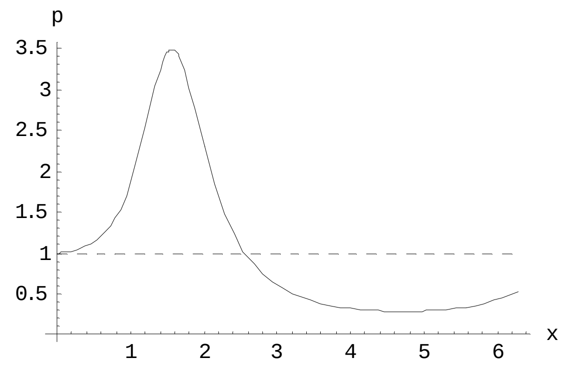
```
Out[3]= {{p → InterpolatingFunction[{{0., 6.28319}, {0., 6.28319}}, <>]}}
```

Μπορούμε να χειριστούμε τη λύση `sol`, που παρουσιάζεται στο `Out[3]`, αναθέτοντάς την σε μία συνηθισμένη συνάρτηση (π.χ. την `pp`)

```
In[4]:= pp[t_, x_] = p[t, x] /. sol[[1]]
        Print["Πυκνότητα για t=1.5 στο x=2, p=", pp[1.5, 2]]
        Plot[{pp[0, x], pp[5, x]}, {x, 0, 2 Pi}, AxesLabel → {"x", "p"},
         PlotStyle → {Dashing[{0.02}], {}}]
```

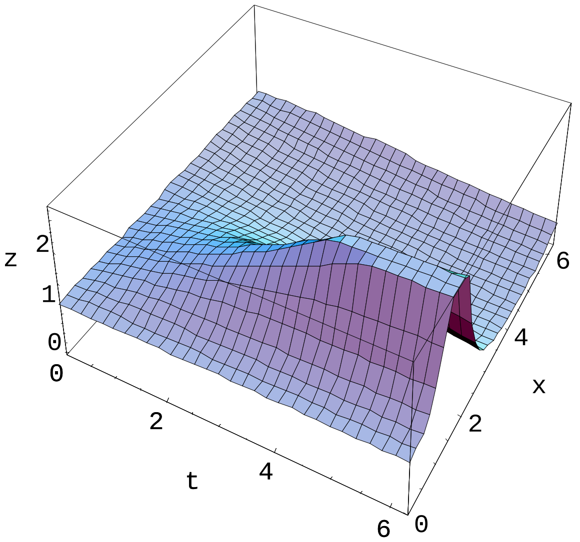
Με τον παραπάνω κώδικα φαίνεται πως μπορούμε να πάρουμε την τιμή της λύσης σε ένα συγκεκριμένο σημείο του διαστήματος ολοκλήρωσης. Επίσης, μπορούμε να πάρουμε μια συνήθη μονοδιάστατη σειρά δεδομένων και να σχεδιάσουμε τη λύση με την `Plot`. Έτσι με τον παραπάνω κώδικα σχεδιάζεται η χωρική κατανομή της πυκνότητας σε συγκεκριμένες χρονικές στιγμές (πχ η αρχική κατανομή για t=0 (διακεκομμένη γραμμή) και για μετά από χρόνο t=5 (συνεχής))

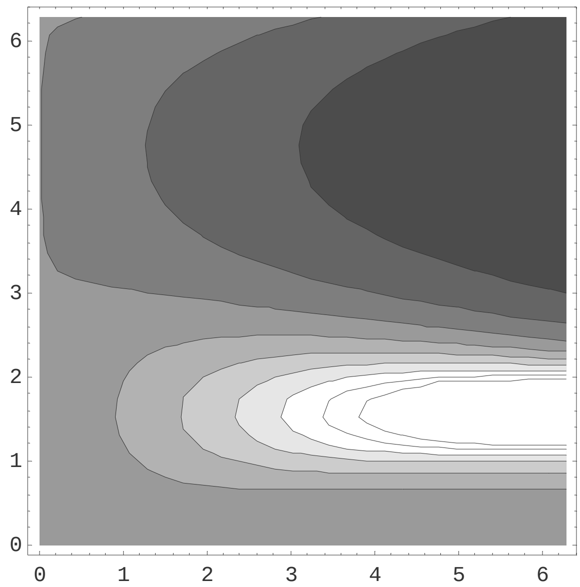
```
Πυκνότητα για t=1.5 στο x=2, p=1.38484
```



Η λύση $\rho=\rho(t,x)$ μπορεί να σχεδιαστεί στο χώρο και στο χρόνο σαν μια συνάρτηση δύο διαστάσεων, δηλαδή με ένα τρισδιάστατο γράφημα (`Plot3D`) ή με διάγραμμα ισοϋψών καμπύλων (`ContourPlot`) όπου οι σκοτεινές και φωτεινές περιοχές αντιστοιχούν σε μικρές και μεγάλες τιμές της συνάρτησης $\rho(t,x)$, αντίστοιχα.

```
      [  [    ]  {          }  {          }
 AxesLabel → {"t", "x", "z"}]
ContourPlot[pp[t, x], {t, 0, 2 π}, {x, 0, 2 π}, PlotPoints → 30,
 AxesLabel → {"t", "x"}]
```





- Στην εντολή `NDSolve` χρησιμοποιήθηκε η επιλογή

$$\texttt{StartingStepSize} \to \texttt{0.01}$$

Αν δεν γίνει η παραπάνω επιλογή, τότε, η προεπιλεγμένη τιμή Automatic δίνει στη μέθοδο αρχικό βήμα $\Delta t \approx 0.2$ το οποίο προκαλεί μεγάλα σφάλματα. Επίσης πρέπει να επισημάνουμε ότι, αν θέσουμε μέγιστο αριθμό βημάτων ίσο με N (με την επιλογή `MaxSteps`), τότε οι τριάδες των σημείων που αποθηκεύονται μπορεί να φτάσουν τον αριθμό N^2.

Ασκήσεις

1. Για την ΣΔΕ $y''=-y-xy'$ και για $y(0)=0$, $y'(0)=1$, να βρεθεί η λύση αναλυτικά και αριθμητικά, και να γίνει η γραφική παράσταση των $y(x)$ και $y'(x)$ για $x\in(-5,5)$ και για τις δύο περιπτώσεις..

2. Να λυθεί αριθμητικά το πρόβλημα αρχικών τιμών

$$y_1'=-y_1-xy_2' \quad , \quad y_1(0)=0$$
$$y_2'=y_1-y_2 \quad , \quad y_2(0)=1$$

Να σχεδιαστούν α) οι $y_1(x)$ και $y_2(x)$ με την `Plot`, β) Οι φασική τροχιά ($y_1(x)$, $y_2(x)$) με την `ParametricPlot` γ) η ολοκληρωτική καμπύλη ($x, y_1(x)$, $y_2(x)$) με την `ParametricPlot3D`.

3. Ο αρμονικός ταλαντωτής

$$\ddot{x}=-x$$

έχει ως λύση την $x=A\cos(t)+B\sin(t)$, όπου A,B σταθερές που εξαρτώνται από τις αρχικές συνθήκες. α) Να λυθεί η παραπάνω εξίσωση αριθμητικά, για $x(0)=1$, $x'(0)=0$, για χρόνο 100 χ.μ. και 10000 χ.μ. με τιμές, για τις επιλογές PrecisionGoal και AccuracyGoal, από 1 έως 8. Να σχεδιαστεί η αριθμητική και αναλυτική λύση. Προκύπτουν διαφορές; β) Η ενέργεια του συστήματος $E=(\dot{x}^2+x^2)/2$ θα πρέπει να διατηρείται σταθερή και ίση με 0.5 για τις παραπάνω αρχικές συνθήκες. Μελετήστε το σφάλμα της αριθμητικής ολοκλήρωσης μέσω της χρονικής μεταβολής της ενέργειας, για τις παραπάνω τιμές των επιλογών, σχεδιάζοντας την τιμή $\Delta E=E-0.5$ σαν συνάρτηση του χρόνου.

4. Προσθέτουμε έναν μη γραμμικό όρο $a\sin(y_2-y_1)$ στο σύστημα του παραδείγματος (1§5,1^o μέρος) και παίρνουμε το σύστημα

$$\dot{y}_1=-y_1-y_2+a\sin(y_2-y_1)$$
$$\dot{y}_2=y_1-y_2$$

Θεωρώντας ένα τυχαίο σύνολο αρχικών συνθηκών στο τετράγωνο γύρω από το (0,0) του φασικού χώρου, να επιλυθεί το σύστημα αριθμητικά, για το διάστημα ολοκλήρωσης $-4<t<4$ και για τις παραπάνω αρχικές συνθήκες, και να σχεδιαστεί το φασικό του πορτρέτο για τιμές τις παραμέτρου a στο διάστημα (-5,5).

5. Για την εξίσωση (6) του Duffing (βλ. παράδειγμα 2§2), και αρχικές συνθήκες $y(0)=y'(0)=0$, να μελετηθεί η συμπεριφορά των ταλαντώσεων και των φασικών τροχιών του συστήματος για $\omega=\alpha=\beta=1$ και για διάφορες τιμές των παραμέτρων γ και F.

6. Όπως στο παράδειγμα 6, μελετήστε τις τροχιές ενός σώματος μέσα σε κεντρικό πεδίο δυνάμεων $F=-1/r^\alpha$ με $\alpha=2+\varepsilon$, $|\varepsilon|<1$. Συνεχίζουν οι κλειστές τροχιές να είναι ελλείψεις;

Βιβλιογραφία

[1] Γ. Μπόζης , *Διαφορικές Εξισώσεις και Εφαρμογές*, Α.Π.Θ. , 1982

[2] Γ. Θεοδώρου, *Εισαγωγή στη Mathematica*, Εκδόσεις Τζιόλα, 2003

[3] S. Wolfram, *Mathematica Reference Guide*, Addison-Wesley Publ., 1992

[4] Σ. Τραχανάς, *Mathematica και εφαρμογές*, Πανεπ. Εκδόσεις Κρήτης, 2002

[5] R. Gaylord, S.Kamin, P. Wellin, *Introduction to programming with Mathematica*, Springer, 1993

[6] T. Wickham-Jones, *Mathematica Graphics*, Springer, 1994

[7] R. Enns and G. McGuire, *Nonlinear Physics with Mathematica*, Birkhauser, 2001

[8] Α. Μπούντης, *Μη γραμμικές Συνήθεις Διαφορικές εξισώσεις*, Εκδόσεις Γ.Α. Πνευματικός, 1997

[9] M. Spiegel, *Ανώτερα Μαθηματικά*, SHAUM'S series, (Ελληνική έκδοση : ΕΣΠΙ 1978)

[10] M.Abramowitz and I. Stegun, *Handbook of mathematical functions*, Dover, 1972

[11] Κ. Κόκκοτας, *Σημειώσεις στην Αριθμητική Ανάλυση*, Τμ. Φυσικής, ΑΠΘ.

[12] W.H.Press, S.A.Teukolsky, W.T.Vetterling, B.P.Flannery, *Numerical Recipes*, Cambridge Univ. Press, 1988-1996.

[13] Ι. Χατζηδημητρίου, *Θεωρητική Μηχανική*, 1^{ος} Τόμος, Εκδ. Γιαχούδη-Γιαπούλη, 2000.

[14] O.Aberth, *Precise Numerical methods using C++*, Academic Press, 1998

[15] M.Abell, J.Braselton, *Differential Equations with Mathematica,* 2nd ed., Academic Press, 1997.