Κούρσα εξοπλισμών

Κώστας Κούδας

Στην παρούσα μελέτη θα μελετήσουμε την πορεία εξέλιξης των εξοπλιστικών προγραμμάτων κάποιων ανταγωνιστικών κρατών. Φυσικά, δεν αξιωνόμαστε ότι με αυτήν μπορούμε να κάνουμε αυστηρές προβλέψεις. Η ποσοτική ανάλυση εν προκειμένω είναι απλά ένα εργαλείο για να βγάλουμε ποιοτικά συμπεράσματα.

Δύο ανταγωνιστικά κράτη

Μοντελοποίηση προβλήματος

Ας υποθέσουμε ότι έχουμε δύο εχθρικά μεταξύ τους κράτη. Λόγω αυτής ακριβώς της εχθρότητας μπαίνουν σε ένα ανταγωνισμό εξοπλισμών. Ας ορίσουμε ως `x[t]` την ποσότητα εξοπλισμών που έχει τη χρονική στιγμή `t` το πρώτο κράτος και `y[t]` το δεύτερο.

Είναι λογικό να θεωρήσουμε ότι υπάρχει ένα άνω φράγμα στο πόσο μεγάλους εξοπλισμούς μπορεί να αποκτήσει ένα κράτος. Όσο και να θέλει η Ελλάδα να αποκτήσει π.χ. 5000000 πυραύλους, δεν μπορεί αυτό να συμβεί, είτε για οικονομικούς λόγους, είτε για λόγους υπάρχοντος προσωπικού κ.τ.λ. Έτσι, δεν είναι παράλογο να υποθέσουμε ότι όσο οι διαθέσιμοι εξοπλισμοί πλησιάζουν αυτό το άνω φράγμα, τόσο μικραίνει ο ρυθμός με τον οποίο αυτοί αυξάνονται. Σε ένα απλουστευτικό μοντέλο θα μπορούσαμε να θεωρήσουμε ότι ο ρυθμός μεταβολής του `x` (ήτοι το `x'[t]`) είναι ανάλογος της διαφοράς `SX-x[t]`, όπου `SY` το άνω φράγμα των εξοπλισμών της πρώτης χώρας. Ομοίως το `y'[t]` είναι ανάλογο του `SY-y[t]`.

Από την άλλη, όσο μεγαλώνει η σχετική διαφορά των δύο εξοπλισμών, τόσο αυτό πιέζει την πιο αδύναμη χώρα να προβεί σε εξοπλισμούς. Και αντιστρόφως, φυσικά! Οι Η.Π.Α. δεν θα είχαν ανάγκη να εξοπλιστούν παραπάνω αν, υποθετικά μιλώντας, είχαν να ανταγωνιστούν μόνο την Αλβανία. Η σχετική διαφορά της πρώτης χώρας σε σχέση με τη δεύτερη είναι η `(x-y)/y`. Θεωρούμε επίσης ότι κάθε χώρα θέλει να έχει ένα «μαξιλαράκι» απόστασης από την άλλη (`px`, `py` αντίστοιχα) και ότι όσο περισσότερο απέχει από αυτό, τόσο πιο έντονη είναι η ανάγκη για εξοπλισμούς. Έτσι, μπορούμε να υποθέσουμε ότι το `x'[t]` είναι επίσης ανάλογο του `px-(x[t]-y[t])/y[t]` και ομοίως ότι το `y'[t]` είναι επίσης ανάλογο του `py-(y[t]-x[t])/x[t]`.

Έχουμε, λοιπόν, το κάτωθι μοντέλο.

```
In[]:= Clear["Global`*"]
dEqX := x'[t] == ax * (SX - x[t]) (px - (x[t] - y[t]) / y[t])
dEqY := y'[t] == ay * (SY - y[t]) (py - (y[t] - x[t]) / x[t])
dEqX
dEqY
```

Out[]=

$$x'[t] == ax\,(SX - x[t])\left(px - \frac{x[t] - y[t]}{y[t]}\right)$$

Out[]=

$$y'[t] == ay\,(SY - y[t])\left(py - \frac{-x[t] + y[t]}{x[t]}\right)$$

Ας του εκχωρήσουμε και τις κάτωθι αρχικές συνθήκες.

```
In[]:= initX := x[0] == x0
initY := y[0] == y0
initX
initY
```

Out[]=

x[0] == x0

Out[]=

y[0] == y0

Σημεία ισορροπίας

Πάμε πρώτα να βρούμε τα σημεία ισορροπίας του συστήματος αυτού.

```
In[]:= vecX := ax * (SX - x) (px - (x - y) / y)
vecY := ay * (SY - y) (py - (y - x) / x)
sol0 := Solve[{vecX == 0, vecY == 0}, {x, y}]
Print[
 "(x,y)=(" <> ToString[x /. sol0[[1]][[1]]] <> ", " <> ToString[y /. sol0[[1]][[2]]] <> ")"]
Print[
 "(x,y)=(" <> ToString[x /. sol0[[2]][[1]]] <> ", " <> ToString[y /. sol0[[2]][[2]]] <> ")"]
Print["(x,y)=(" <> ToString[x /. sol0[[3]][[1]]] <> ", " <> ToString[y /. sol0[[3]][[2]]] <> ")"]
```

(x,y)=(SX, SX + py SX)

(x,y)=(SX, SY)

(x,y)=(SY + px SY, SY)

Αυτά τώρα θα τα αξιολογήσουμε ως προς την ευστάθεια. Προς τούτο θα βρούμε πρώτα τον Ιακωβιανό πίνακα.

In[•]:=
```
jacobianMatix := {{D[vecX, x], D[vecX, y]}, {D[vecY, x], D[vecY, y]}}
MatrixForm[jacobianMatix]
```

Out[•]//MatrixForm=

$$\begin{pmatrix} -\left(ax\left(px-\frac{x-y}{y}\right)\right)-\frac{ax\,(SX-x)}{y} & ax\,(SX-x)\left(\frac{x-y}{y^2}+\frac{1}{y}\right) \\ ay\,(SY-y)\left(\frac{1}{x}+\frac{-x+y}{x^2}\right) & -\frac{ay\,(SY-y)}{x}-ay\left(py-\frac{-x+y}{x}\right) \end{pmatrix}$$

Για τα σημεία ισορροπίας αυτός γίνεται ως ακολούθως.

In[•]:=
```
Print[
 "(x,y)=(" <> ToString[x /. sol0⟦1⟧⟦1⟧] <> ", " <> ToString[y /. sol0⟦1⟧⟦2⟧] <> ")"]
jacobianMatix /. sol0⟦1⟧ // MatrixForm
Print[
 "(x,y)=(" <> ToString[x /. sol0⟦2⟧⟦1⟧] <> ", " <> ToString[y /. sol0⟦2⟧⟦2⟧] <> ")"]
jacobianMatix /. sol0⟦2⟧ // MatrixForm
Print[
 "(x,y)=(" <> ToString[x /. sol0⟦3⟧⟦1⟧] <> ", " <> ToString[y /. sol0⟦3⟧⟦2⟧] <> ")"]
jacobianMatix /. sol0⟦3⟧ // MatrixForm
```

(x,y)=(SX, SX + py SX)

Out[•]//MatrixForm=

$$\begin{pmatrix} -\left(ax\left(px+\frac{py\,SX}{SX+py\,SX}\right)\right) & 0 \\ ay\left(\frac{1}{SX}+\frac{py}{SX}\right)(-SX-py\,SX+SY) & -\frac{ay\,(-SX-py\,SX+SY)}{SX} \end{pmatrix}$$

(x,y)=(SX, SY)

Out[•]//MatrixForm=

$$\begin{pmatrix} -\left(ax\left(px-\frac{SX-SY}{SY}\right)\right) & 0 \\ 0 & -\left(ay\left(py-\frac{-SX+SY}{SX}\right)\right) \end{pmatrix}$$

(x,y)=(SY + px SY, SY)

Out[•]//MatrixForm=

$$\begin{pmatrix} -\frac{ax\,(SX-SY-px\,SY)}{SY} & ax\left(\frac{1}{SY}+\frac{px}{SY}\right)(SX-SY-px\,SY) \\ 0 & -\left(ay\left(py+\frac{px\,SY}{SY+px\,SY}\right)\right) \end{pmatrix}$$

Ακολούθως θα βρούμε τις ιδιοτιμές του στα σημεία ισορροπίας.

In[ ]:=
```
Print[
 "(x,y)=(" <> ToString[x /. sol0[[1]][[1]]] <> ", " <> ToString[y /. sol0[[1]][[2]]] <> ")"]
Eigenvalues[jacobianMatix /. sol0[[1]]] // Simplify
Print[
 "(x,y)=(" <> ToString[x /. sol0[[2]][[1]]] <> ", " <> ToString[y /. sol0[[2]][[2]]] <> ")"]
Eigenvalues[jacobianMatix /. sol0[[2]]] // Simplify
Print[
 "(x,y)=(" <> ToString[x /. sol0[[3]][[1]]] <> ", " <> ToString[y /. sol0[[3]][[2]]] <> ")"]
Eigenvalues[jacobianMatix /. sol0[[3]]] // Simplify
```

(x,y) = (SX, SX + py SX)

Out[ ]=

$$\left\{-\frac{ax\ (px+py+px\ py)}{1+py},\ ay\left(1+py-\frac{SY}{SX}\right)\right\}$$

(x,y) = (SX, SY)

Out[ ]=

$$\left\{-\frac{ay\ (SX+py\ SX-SY)}{SX},\ \frac{ax\ (SX-(1+px)\ SY)}{SY}\right\}$$

(x,y) = (SY + px SY, SY)

Out[ ]=

$$\left\{-\frac{ay\ (px+py+px\ py)}{1+px},\ ax\left(1+px-\frac{SX}{SY}\right)\right\}$$

Αφού η πρώτη ιδιοτιμή είναι στην 1η και την 3η περίπτωση αρνητική, αυτό που μάς ενδιαφέρει είναι πότε η δεύτερη ιδιοτιμή είναι αρνητική. Το πρόσημο της δεύτερης ιδιοτιμής είναι:

- το πρόσημο της παράστασης `1+py-SY/SX` στην 1η περίπτωση,
- το πρόσημο της παράστασης `1+px-SX/SY` στην 3η περίπτωση.

Όσον αφορά την 2η περίπτωση, τα πρόσημα των ιδιοτιμών της είναι:

- το πρόσημο της παράστασης `SY/SX-(1+py)` της πρώτης ιδιοτιμής
- το πρόσημο της παράστασης `SX/SY-(1+px)` της δεύτερης ιδιοτιμής.

Συγκεκριμένα παραδείγματα

Ας θεωρήσουμε τώρα ότι έχουμε μία ισχυρή χώρα και μία αδύναμη. Θα υποθέσουμε παραδειγματικά ότι SY=6 και SX=4. Είναι λογικό η αδύναμη χώρα να συμβιβάζεται με μια μικρότερη εξοπλιστική υπεροχή έναντι της ισχυρής, όταν την επιτυγχάνει, οπότε θα διαλέξουμε `px=0.01` και `py=0.02`. Τέλος, δεν είναι παράλογο να θεωρήσουμε ότι η αδύναμη χώρα θα δέχεται μεγαλύτερες πιέσεις από την ισχυρή στο να ακολουθήσει τον εξοπλιστικό ανταγωνισμό, οπότε θα θέσουμε ax=3 και ay=2.

```
ax = 3;
ay = 2;
SX = 4;
SY = 6;
px = 0.01;
py = 0.02;
dEqX
dEqY
initX
initY
```

Out[ ]=

$$x'[t] == 3\ (4 - x[t])\ \left(0.01 - \frac{x[t] - y[t]}{y[t]}\right)$$

Out[ ]=

$$y'[t] == 2\ (6 - y[t])\ \left(0.02 - \frac{-x[t] + y[t]}{x[t]}\right)$$

Out[ ]=

$$x[0] == x0$$

Out[ ]=

$$y[0] == y0$$

Πάμε τώρα να λύσουμε το σύστημα διαφορικών εξισώσεων για τα δεδομένα που θέσαμε.

```
sol := NDSolve[{dEqX, dEqY, initX, initY}, {x, y}, {t, 0, 30}];
```

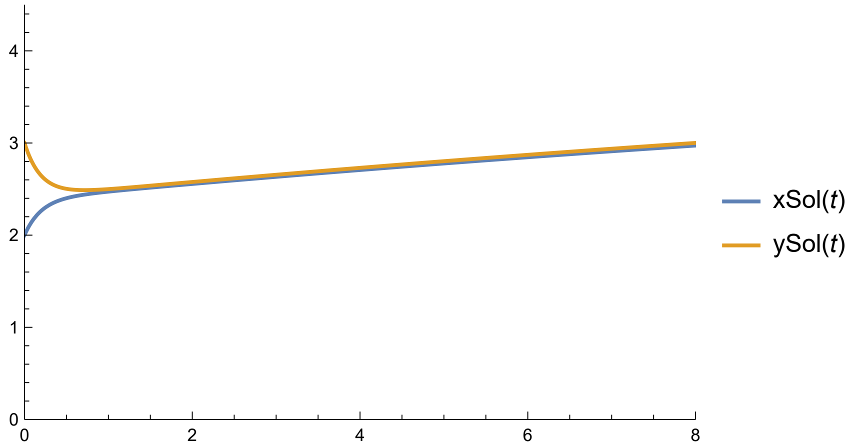
Δίνουμε ένα εύλογο προβάδισμα στην ισχυρή χώρα.

```
x0 = 2;
y0 = 3;
xSol[t_] := Evaluate[x[t] /. sol⟦1⟧⟦1⟧]
ySol[t_] := Evaluate[y[t] /. sol⟦1⟧⟦2⟧]
```

Και απεικονίζουμε τώρα τις λύσεις μας.

```
Plot[{xSol[t], ySol[t]}, {t, 0, 30},
 PlotLegends → "Expressions", PlotRange → {{0, 8}, {0, 0.75 Max[SX, SY]}}]
```

Out[ ]=

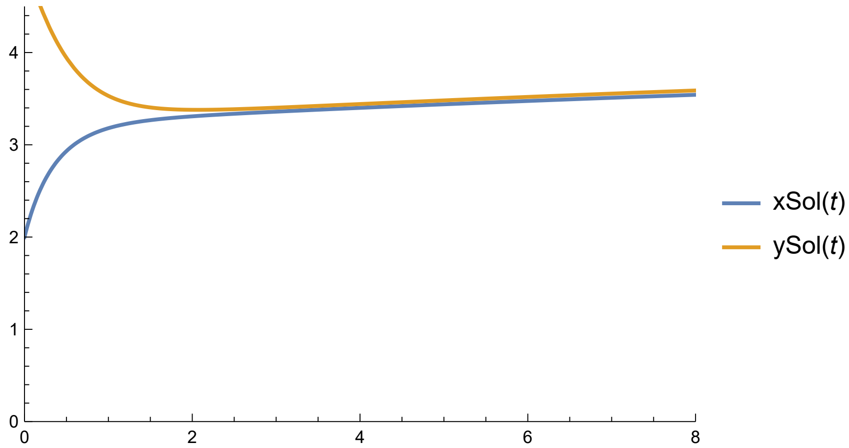


Πάμε ν' αλλάξουμε την αρχική τιμή της δεύτερης χώρας, να δούμε πώς θ' αλλάξει η συμπεριφορά του ανταγωνισμού.

```
Clear[ySol]
y0 = 5;
xSol[t_] := Evaluate[x[t] /. sol⟦1⟧⟦1⟧]
ySol[t_] := Evaluate[y[t] /. sol⟦1⟧⟦2⟧]
```

```
Plot[{xSol[t], ySol[t]}, {t, 0, 30},
 PlotLegends → "Expressions", PlotRange → {{0, 8}, {0, 0.75 Max[SX, SY]}}]
```

Out[∘]=

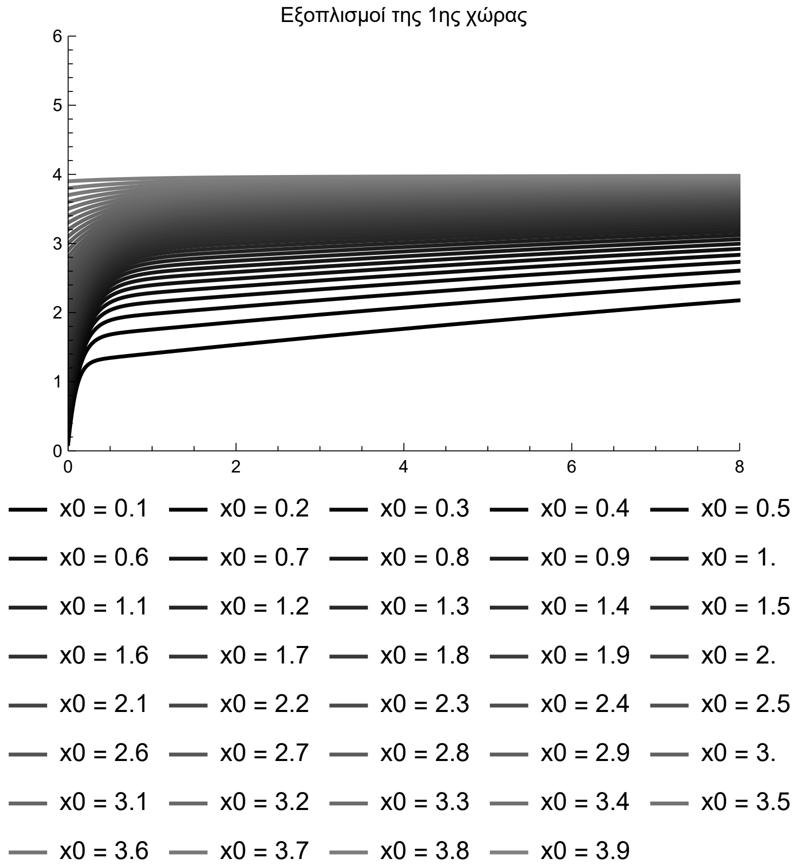


Βλέπουμε ότι αλλάζοντας την αρχική τιμή έστω της μίας χώρας, επηρρεάζεται και η άλλη.

```
xSolutions = Table[x[t] /. sol⟦1⟧⟦1⟧, {x0, 0.1, SX - 0.1, 0.1}];
xSols = Evaluate[xSolutions];
```

```
numOfa = (SX - 0.1 - 0.1) / 0.1;
grayList = Table[GrayLevel[n], {n, 0, 0.5, 0.5 / numOfa}];
names = Table["x0 = " <> ToString[n], {n, 0.1, SX - 0.1, 0.1}];
Plot[xSols, {t, 0, 30}, PlotRange → {{0, 8}, {0, Max[SX, SY]}}, PlotStyle → grayList,
 PlotLabel → "Εξοπλισμοί της 1ης χώρας", PlotLegends → Placed[names, Below]]
```

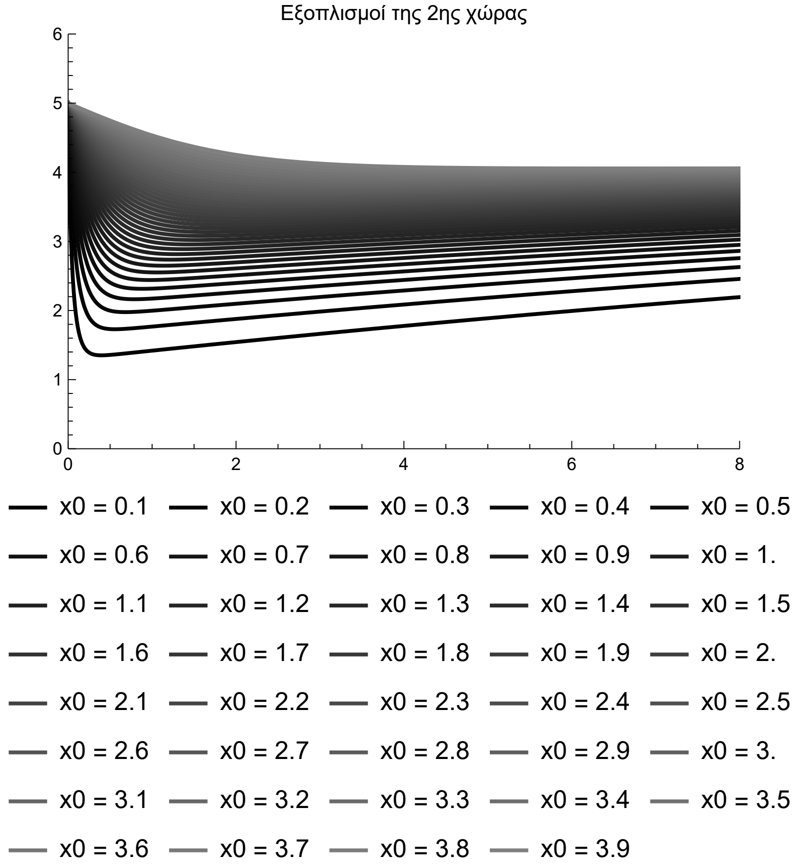
Out[∘]=




```
ySolutions = Table[y[t] /. sol[[1]][[2]], {x0, 0.1, SX - 0.1, 0.1}];
ySols = Evaluate[ySolutions];
```

```
Plot[ySols, {t, 0, 30}, PlotRange → {{0, 8}, {0, Max[SX, SY]}}, PlotStyle → grayList,
 PlotLabel → "Εξοπλισμοί της 2ης χώρας", PlotLegends → Placed[names, Below]]
```

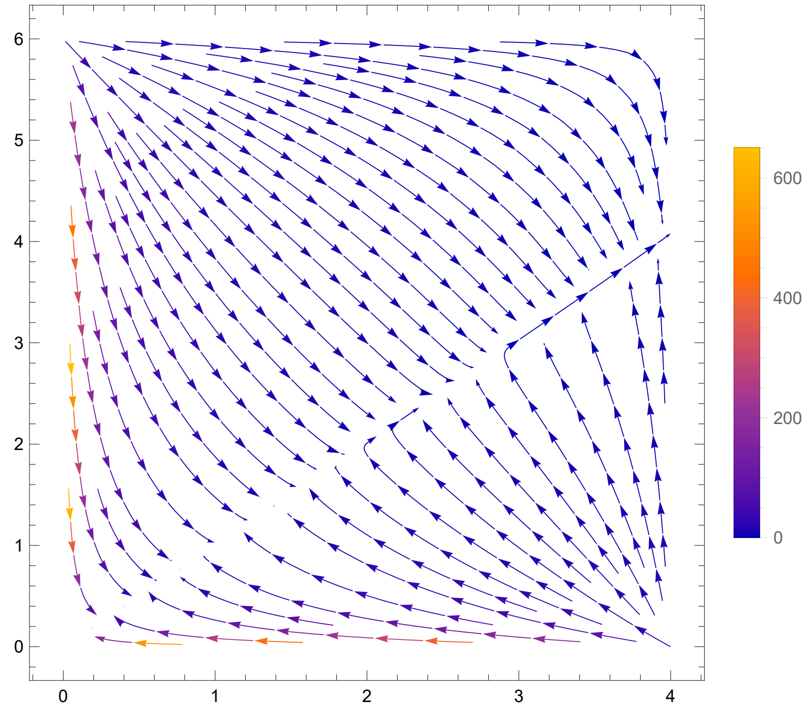
Out[•]=



Ας δούμε πώς συνεξελίσσονται οι εξοπλισμοί των δύο αυτών χωρών!

```
StreamPlot[{vecX, vecY}, {x, 0, SX}, {y, 0, SY}, PlotLegends → Automatic]
```

Out[•]=



Τρία ανταγωνιστικά κράτη

Ένα κράτος και συμμαχία δύο κρατών

Μία συμμαχία δύο κρατών ενάντια σε μια άλλη συμμαχία δύο κρατών