Progetto per l’esame di Sistemi Dinamici

Gabriele Granzotto, Sara Trabucco

Conigli contro pecore

Si consideri il sistema dinamico:

$$\begin{aligned} \dot{x} &= x(3 - ax - by) \\ \dot{y} &= y(2 - x - y) \end{aligned}$$

che rappresenta la competizione per le stesse risorse di una popolazione di conigli x e di pecore y.

$a > 0$ è l’inverso della capacità portante dell’ambiente per i conigli e $b > 0$ il coefficiente di competizione dei conigli con le pecore.

In particolare si consideri il caso $a = 2$ e $b = 1$.

• **Determinare i punti fissi e studiarne la stabilità**.

• **Tracciare le isocline**.

• **Studiare numericamente il campo vettoriale**.

• **Linearizzare il flusso intorno ai punti fissi e tracciare numericamente alcune orbite intorno ai punti fissi**.

• **Individuare le varietà lineari stabile e instabile vicino ai punti fissi**.

• **Studiare numericamente l’estensione nonlineare delle varietà stabili e instabili**.

Usando il codice numerico, si studi cosa succede al variare di a e di b nell’intervallo $1 < a < 3, 1 < b < 3$.

Per cominciare, si vogliono determinare i punti fissi e le relative stabilità

Fissati $a = 2, b = 1$, il sistema dinamico sarà:

$$\begin{aligned} \dot{x} &= x(3 - 2x - y) \\ \dot{y} &= y(2 - x - y) \end{aligned}$$

I punti fissi del sistema sono $(0,0)$, $(0,2)$, $\left(0, \frac{2}{3}\right)$, $(1,1)$; per studiarne la stabilità, bisogna considerare la jacobiana del sistema…

$$J = \begin{bmatrix} 3 - 4x - y & -x \\ -y & 2 - 2y - x \end{bmatrix}$$

…per poi considerare il flusso linearizzato in ogni punto fisso:

- $J_{(0,0)} = \begin{bmatrix} 3 & 0 \\ 0 & 2 \end{bmatrix}$

$$\tau = tr(J) = 3 + 2 = 5 > 0$$

$$\Delta = det(J) = 3 \cdot 2 = 6 > 0$$

Il punto potrebbe essere un *nodo instabile* oppure una *spirale espandente*; per verificarlo, bisogna calcolare gli autovalori. Essendo la matrice diagonale, gli autovalori saranno gli elementi diagonali, quindi $\lambda_1 = 3,\ \lambda_2 = 2$

Essendo entrambi numeri reali puri, il punto fisso $A(0,0)$ è un **nodo instabile**.

- $J_{(0,2)} = \begin{bmatrix} 1 & 0 \\ -2 & -2 \end{bmatrix}$

$$\tau = tr(J) = 1 + (-2) = -1 < 0$$

$$\Delta = det(J) = 1 \cdot (-2) = -2 < 0$$

Il punto fisso $B(0,2)$ è un **punto di sella** e gli autovalori della matrice, essendo essa triangolare, sono $\lambda_1 = 1,\ \lambda_2 = -2$.

- $J_{\left(\frac{3}{2},0\right)} = \begin{bmatrix} -3 & -\frac{3}{2} \\ 0 & \frac{1}{2} \end{bmatrix}$

$$\tau = tr(J) = -3 + \frac{1}{2} = -\frac{5}{2} < 0$$

$$\Delta = det(J) = -3 \cdot \frac{1}{2} = -\frac{3}{2} < 0$$

Il punto fisso $C\left(\frac{3}{2}, 0\right)$ è un **punto di sella** e gli autovalori sono $\lambda_1 = -3,\ \lambda_2 = \frac{1}{2}$.

- $J_{(1,1)} = \begin{bmatrix} -2 & 1 \\ 1 & 1 \end{bmatrix}$

$$\tau = tr(J) = -2 + (-1) = -3 < 0$$

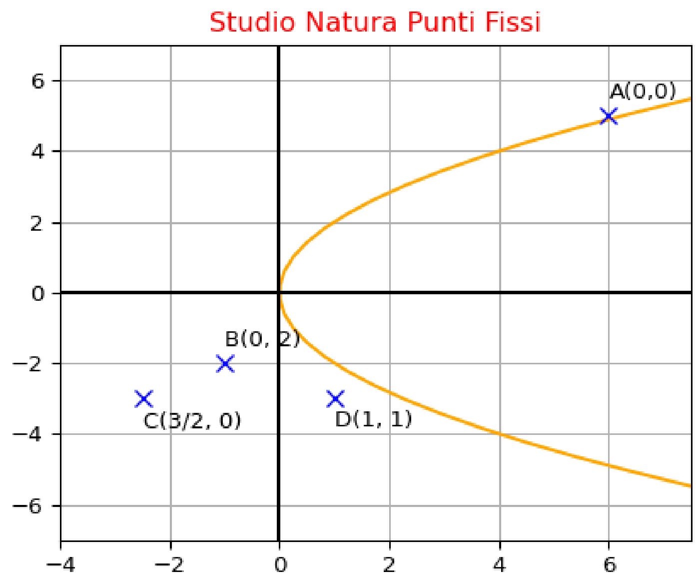
$$\Delta = det(J) = -2 \cdot (-1) - (-1) \cdot (-1) = 1 > 0$$

Dato che $\tau < 0$ e $\Delta > 0$, il punto fisso potrebbe essere un *nodo stabile* oppure una *spirale contraente*. Bisogna calcolare allora gli autovalori:

$$\lambda_{1,2} = \frac{\tau \pm \sqrt{\tau^2 - 4\Delta}}{2} = \frac{3 \pm \sqrt{3^2 - 4 \cdot 1}}{2}$$

$$\Rightarrow \lambda_1 = \frac{-3+\sqrt{5}}{2}, \quad \lambda_2 = \frac{-3-\sqrt{5}}{2}$$

Dato che sono entrambi numeri reali puri, il punto fisso $D(1,1)$ è un **nodo stabile**.



I punti B e C si trovano nella regione dei **punti di sella**, il punto D si trova nella regione dei **nodi stabili** o **contraenti** mentre il punto A, sebbene si trovi quasi sulla parabola (ed in effetti sembra avere un comportamento simile ad una **stella**), in realtà risulta essere un nodo **instabile** o **espandente**.

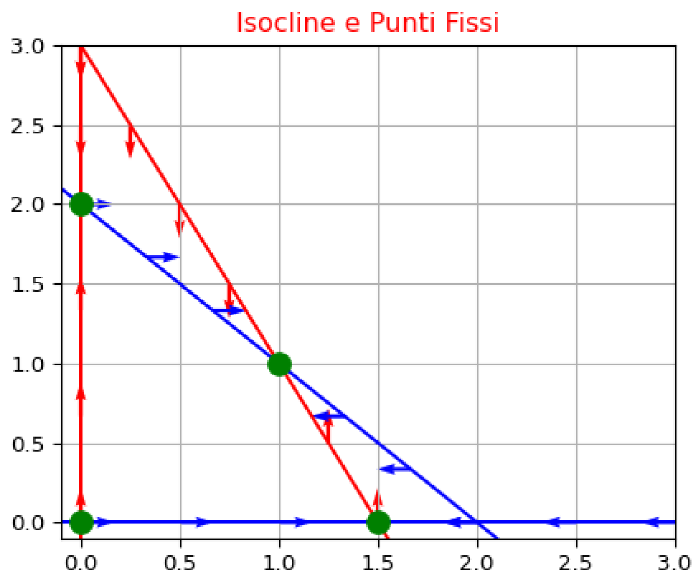
Per valutare il comportamento del campo vettoriale, vale la pena considerare le **isocline.** Le equazioni delle isocline saranno:

$y = 0, \ \ y = -2x + 3$, che hanno la componente x nulla

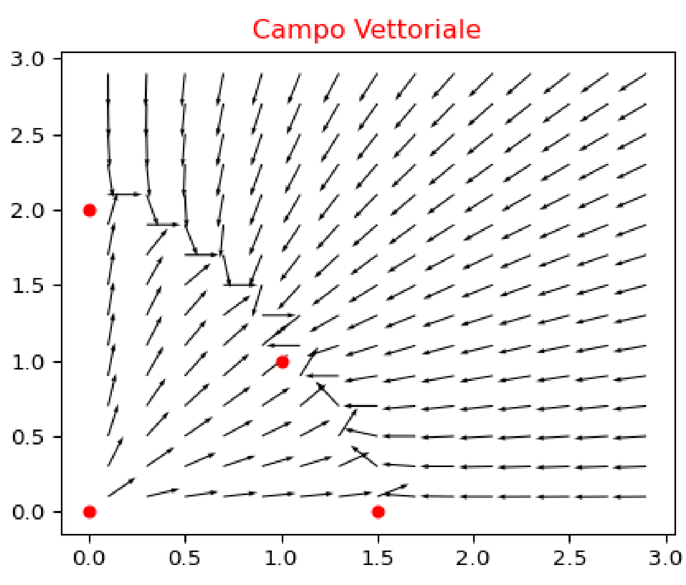
$x = 0, \ \ x = -y + 2$, che hanno la componente y nulla

Perché sono importanti le isocline?

Lungo le isocline la direzione del campo vettoriale del sistema sarà sempre la stessa (ossia le curve integrali o traiettorie intersecano le isocline sempre con la stessa inclinazione) ed il loro punto di intersezione è un punto fisso, infatti:



Per studiare in modo numerico la funzione è sempre meglio partire con una rappresentazione sul campo vettoriale in modo da darci una rappresentazione grezza del comportamento delle equazioni sul piano:



Cosa succede invece alle traiettorie intorno ai punti fissi?

Dall'analisi del sistema sono stati trovati quattro punti fissi, di cui un nodo instabile in $A(0,0)$, uno stabile in $D(1,1)$ e due punti di sella in $B(0,2)$ e $C\left(\frac{3}{2},0\right)$.

Per una visione più generale di cosa succede al campo vettoriale in un loro intorno, si considerano i rispettivi manifold stabili e instabili:

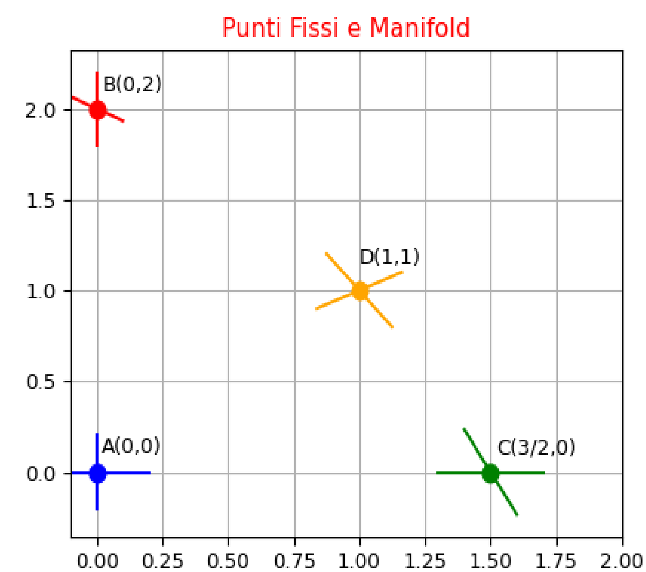
- $A(0,0)$ è un **nodo instabile**, quindi gli autovalori sono entrambi reali positivi e si hanno due **manifold instabili**; le autodirezioni saranno $(1,0)$ e $(0,1)$ e saranno entrambe *espandenti*;

- $B(0,2)$ è un **punto di sella** ed ha un autovalore positivo ed uno negativo; le autodirezioni sono $(0,1)$ e $(3,-2)$, rispettivamente *contraente* ed *espandente*;

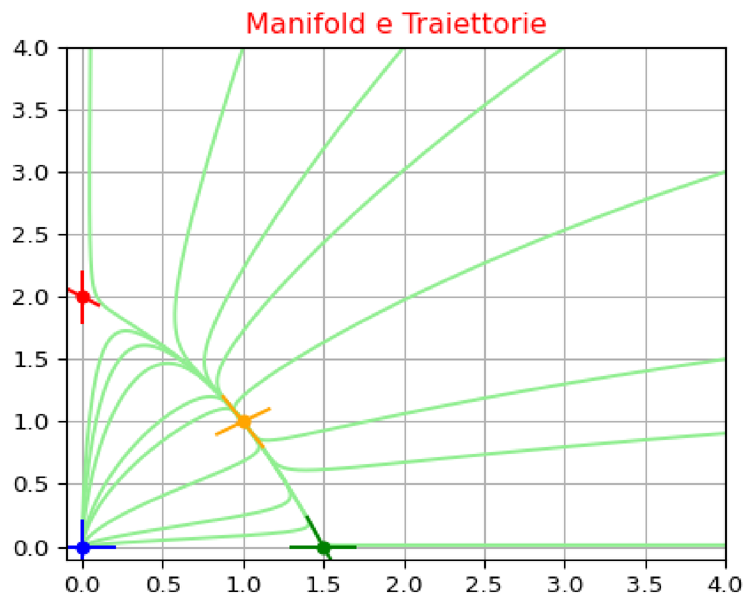
- $C\left(\frac{3}{2},0\right)$ è a sua volta un **punto di sella**, anche qui gli autovalori sono reali di segno opposto; le autodirezioni saranno $\left(-\frac{3}{7},1\right)$ e $(1,0)$, la prima *contraente* e la seconda *espandente*;

- $D(1,1)$ è un **nodo stabile**, con i rispettivi autovalori entrambi reali negativi, si avranno quindi due manifold entrambi *contraenti*; le autodirezioni sono $\left(\frac{1+\sqrt{5}}{2},1\right)$ e $\left(\frac{1-\sqrt{5}}{2},1\right)$.

Si può osservare inoltre che il manifold riferito all'autovalore $\lambda_2 = \frac{-3-\sqrt{5}}{2}$ contrae più velocemente rispetto all'altro.

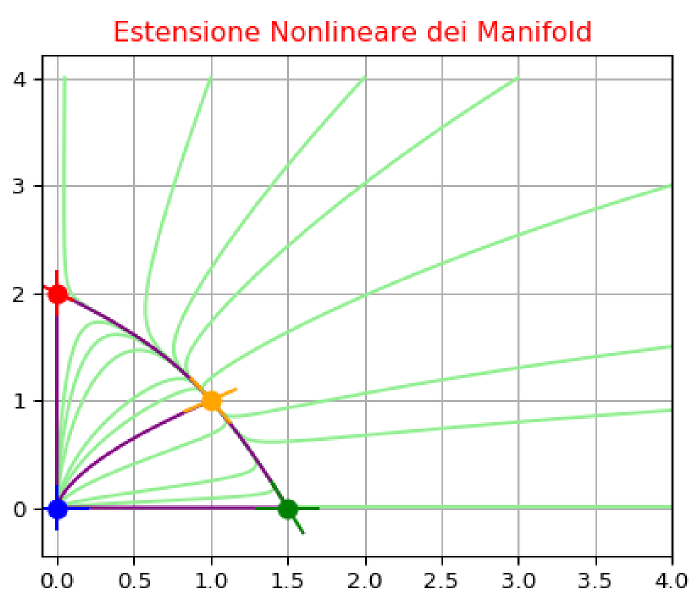


Graficamente, alcune traiettorie intorno ai punti fissi saranno:



Ma cosa succede estendendo i manifold dei due punti di sella, del nodo instabile e di quello stabile?

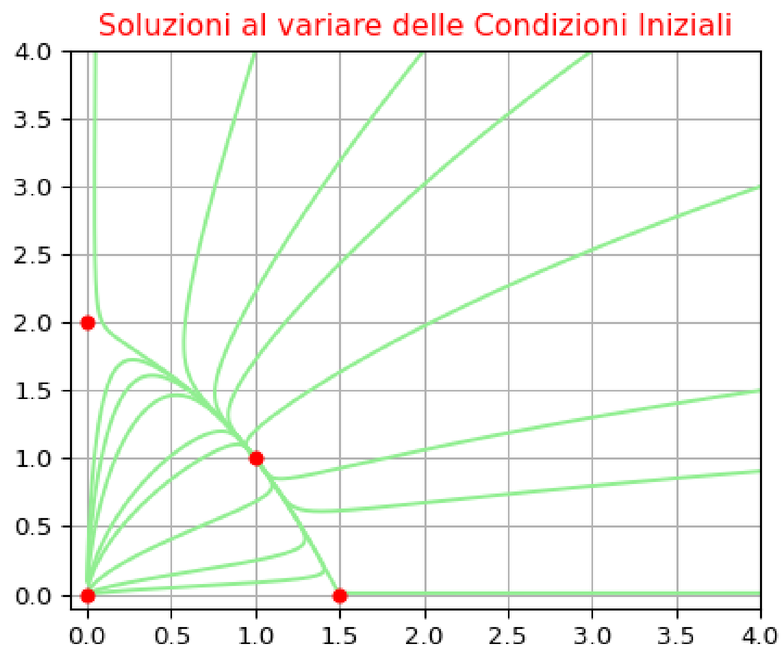
La linea viola rappresenta l'estensione non lineare:



I punti $A(0,0)$, $B(0,2)$, $C\left(\frac{3}{2},0\right)$ e $D(1,1)$ risultano essere **eteroclini**: i manifold instabili che partono dai punti A, B e C si congiungono ai manifold stabili del punto D.

Traiettorie al variare delle condizioni iniziali

Le soluzioni al variare dei valori iniziali ed il campo vettoriale di questo sistema saranno:



Analizzando il sistema dal punto di vista dei conigli e delle pecore, sembra che, per qualsiasi condizione iniziale che non abbia una delle componenti x o y nulle, si arrivi ad uno stato di equilibrio tra le due specie (cioé il nodo stabile (1,1)).

Cosa succede al variare di a,b nell'intervallo (1,3)?

Per valutare come cambi la natura dei punti fissi, si può fare prima di tutto una piccola analisi analitica, considerando le jacobiane nei punti fissi del sistema in cui a e b sono due valori generici:

$$\begin{aligned} \dot{x} &= x(3 - ax - by) \\ \dot{y} &= y(2 - x - y) \end{aligned}$$

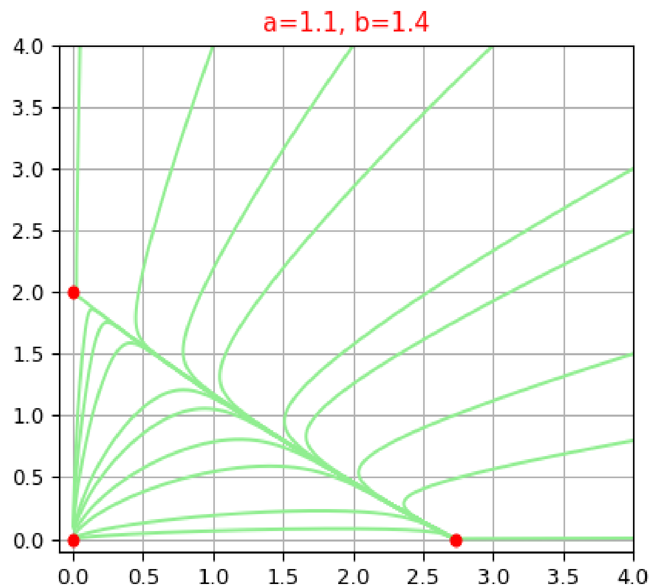
$$J = \begin{bmatrix} 3 - 2ax - y & -bx \\ -y & 2 - 2y - x \end{bmatrix}$$

I punti fissi di questo sistema sono quattro, $A(0{,}0)$, $B(0{,}2)$, $C\left(\frac{3}{a}, 0\right)$ e $D\left(\frac{2b-3}{b-a}, \frac{3-2a}{b-a}\right)$; considerando le rispettive jacobiane si possono fare un paio di osservazioni preliminari:

- Il punto B dipende dai valori che assume la costante b e risulta essere:
  - un punto di sella se $b < \frac{3}{2}$
  - se $b = \frac{3}{2}$ si ha un caso di neutralità (uno degli autovalori è nullo)
  - negli altri casi è un nodo stabile (come nel caso di *rabbits vs sheeps* del libro di Strogatz)
- Il punto C dipende invece dal valore che assume a e risulta essere:
  - un punto di sella se $a > \frac{3}{2}$
  - se $a = \frac{3}{2}$ si ha un caso di neutralità (uno degli autovalori è nullo)
  - negli altri casi è un nodo stabile (come nel caso di *rabbits vs sheeps* del libro di Strogatz)
- Il punto D dipende dai valori assunti sia da a che da b, quindi è più difficile studiarne la natura analiticamente. Tuttavia si può osservare che se $a = b$, il nodo non compare (il denominatore si annullerebbe!) ma nel caso $a = b = \frac{3}{2}$, la jacobiana è la matrice nulla: questo fa sospettare il caso di un nodo degenere.

Dopo quest'analisi sulle jacobiane, si considerano alcuni valori particolari di a e b:

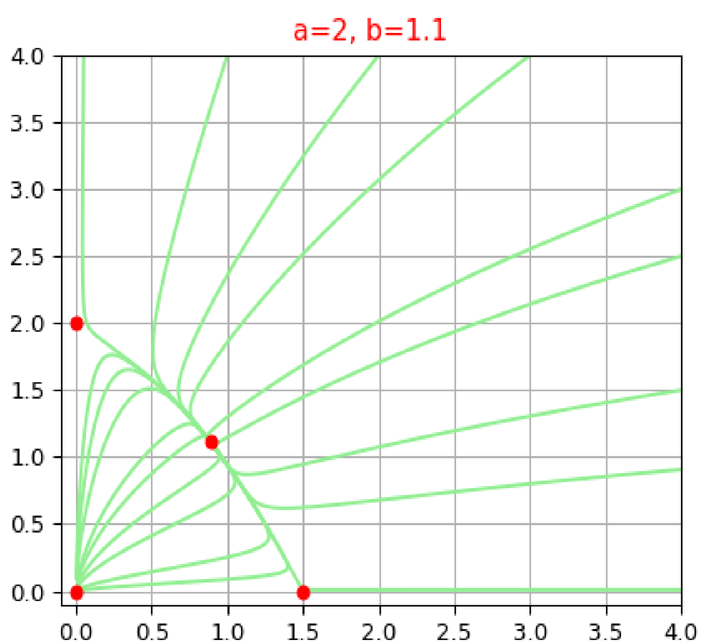
- $1 < a < \frac{3}{2}$ e $1 < b < \frac{3}{2}$:



Il punto B è un punto di sella, il nodo C è un nodo stabile. Sembra che il nodo D sia "collassato" sul punto C.

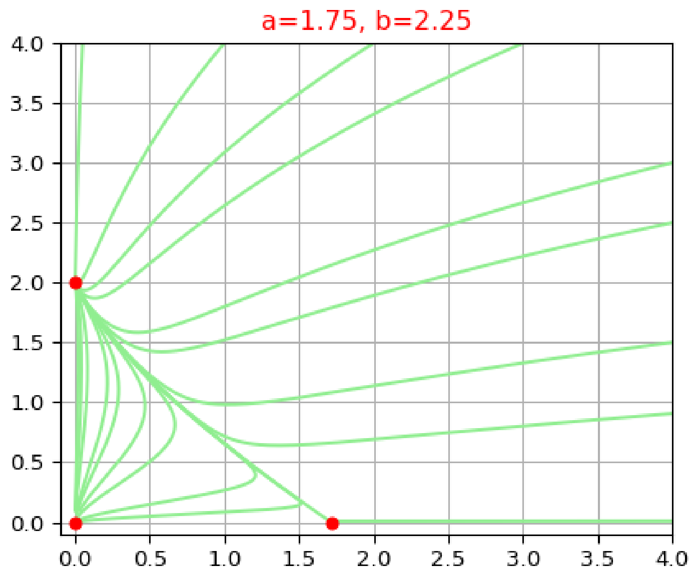
Dal punto di vista dei conigli e delle pecore, sembra che, a meno che la componente x non sia nulla, si finisca sempre nella situazione in cui prevalgono i conigli.

- $\frac{3}{2} < a < 3$ e $1 < b < \frac{3}{2}$ (che è simile al caso considerato)



I punti B e C sono due punti di sella, mentre il punto D è il nodo stabile a cui le orbite convergono.

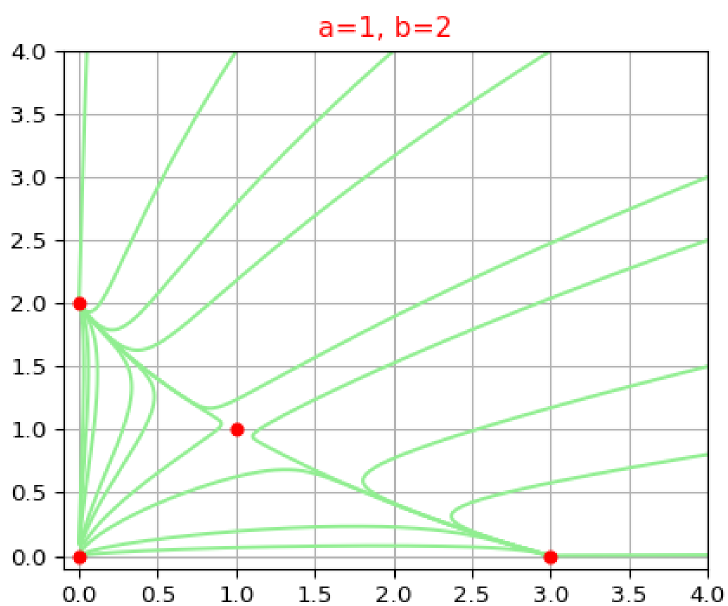
- $\frac{3}{2} < a < 3$ e $\frac{3}{2} < b < 3$



Il punto B è un nodo stabile mentre il punto C è un punto di sella. Sembra che il punto D sia “collassato” sul punto fisso B.

Dal punto di vista di conigli e pecore, sembra che, per qualsiasi condizione iniziale che non abbia componente y nulla, prevalgano le pecore.

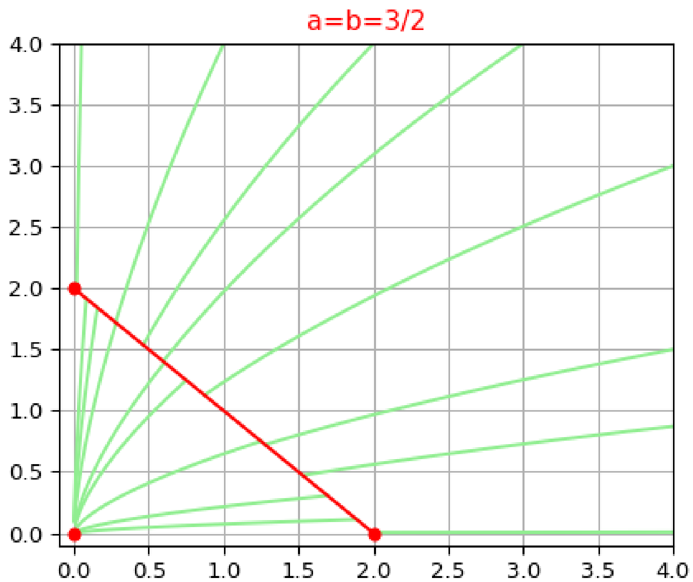
- $1 < a < \frac{3}{2}$ e $\frac{3}{2} < b < 3$ (che è il caso *rabbits vs sheeps* del libro di Strogatz)



Si può osservare che i punti B e C sono nodi stabili, mentre il punto D è un punto di sella.

Dal punto di vista delle due popolazioni, si può osservare come le soluzioni si dividano in due zone divise dai manifold del punto di sella; in alcuni casi possono prevalere i conigli, in altri le pecore.

- $a = b = \frac{3}{2}$



In B e C si ha una situazione di neutralità (una direzione di ciascuna è neutra) mentre il punto fisso D non è più visibile.

Per quanto riguarda il codice utilizzato…

L'analisi numerica è stata possibile grazie a Python ed in particolare alle librerie **numpy** per l'ottimizzazione computazionale, **matplotlib.pyplot** per i plot dei grafici.

Per la simulazione numerica sono stati usati l'algoritmo di Eulero, Eulero modificato e Runge-Kutta riportati qua sotto:

```
def euler_method_2d(initial_time, final_time, initial_value, delta_time, modified=False):
    time = initial_time
    result1 = initial_value[0]
    result2 = initial_value[1]
    elements = round((final_time - initial_time)/delta_time)

    time_vector = np.empty(elements)
    result1_vector = np.empty(elements)
    result2_vector = np.empty(elements)


    for i in range(elements):
        time_vector[i] = time
        result1_vector[i] = result1
        result2_vector[i] = result2

        time += delta_time

        if modified:
            fake_result1 = chosen_function(result1, result2, time)[0]*delta_time + result1
            result1 += (chosen_function(result1, result2, time)[0]+chosen_function(fake_result1, 
result2, time)[0])*delta_time/2

            fake_result2 = chosen_function(result1, result2, time)[1]*delta_time + result2
            result2 += (chosen_function(result1, result2, time)[1]+chosen_function(result1, 
fake_result2, time)[1])*delta_time/2
        else:
            result1 += chosen_function(result1, result2, time)[0]*delta_time
            result2 += chosen_function(result1, result2, time)[1]*delta_time


    return time_vector, result1_vector, result2_vector

def runge_kutta_method(initial_time, final_time, initial_value, delta_time):
    time = initial_time
    result = initial_value
    elements = round((final_time - initial_time)/delta_time)

    time_vector = np.empty(elements)
    result_vector = np.empty(elements)

    for i in range(elements):
        time_vector[i] = time
        result_vector[i] = result
        time += delta_time

        fake_results_1 = chosen_function(result, time)*delta_time
        fake_results_2 = chosen_function(result+0.5*fake_results_1, time)*delta_time
        fake_results_3 = chosen_function(result+0.5*fake_results_2, time)*delta_time
        fake_results_4 = chosen_function(result+fake_results_3, time)*delta_time
        result += (fake_results_1+ 2*fake_results_2+ 2*fake_results_3+ fake_results_4)/6

    return time_vector, result_vector
```