Physik der sozio-ökonomischen Systeme *mit dem Computer*

JOHANN WOLFGANG GOETHE UNIVERSITÄT
27.11.2020

MATTHIAS HANAUSKE

FRANKFURT INSTITUTE FOR ADVANCED STUDIES
JOHANN WOLFGANG GOETHE UNIVERSITÄT
INSTITUT FÜR THEORETISCHE PHYSIK
ARBEITSGRUPPE RELATIVISTISCHE ASTROPHYSIK
D-60438 FRANKFURT AM MAIN
GERMANY

Aufgrund der Corona Krise findet die Vorlesung und die freiwilligen Übungstermine in diesem Semester nur Online statt.

4. Vorlesung

Plan für die heutige Vorlesung

- Kurze Wiederholung der Inhalte der 3. Vorlesung

- Einführung in die Evolutionäre Spieltheorie
  - Die Differentialgleichung eines evolutionären, symmetrischen (2x2)-Spiels
    - Dominante Spiele
    - Koordinationsspiele
    - Anti-Koordinationsspiele

  - Das System von Differentialgleichung eines evolutionären, unsymmetrischen (2x2)-Spiels (Bi-Matrix Spiele)
    - Eckenspiele (Corner Class Games)
    - Sattelpunktspiele (Saddle Class Games)
    - Zentrumspiele (Center Class Games)

Klassen symmetrischer (2x2)-Spiele

	Spieler B Strategie 1 y=1	Spieler B Strategie 1 y=0
Spieler A Strategie 1 x=1	(a , a)	(b , c)
Spieler A Strategie 2 x=0	(c , b)	(d , d)

Symmetrisches (2x2)-Spiel

Symmetrische (2x2)-Spiele lassen sich in drei unterschiedliche Klassen gliedern:

1. Dominante Spiele
2. Koordinationspiele
3. Anti-Koordinationsspiele

Die Klasse der dominanten Spiele ($a > c$ und $b > d$ bzw. $a < c$ und $b < d$)

Bei dieser Spielklasse dominiert eine Strategie die andere. Es existiert nur ein reines Nash-Gleichgewicht welches die dominante Strategie des Spiels darstellt. Dieser Fall tritt ein, falls:

$a > c$ und $b > d$: Strategie 1 dominiert Strategie 2; dominante Strategie bei $(x,y)=(1,1)$.

$a < c$ und $b < d$: Strategie 2 dominiert Strategie 1; dominante Strategie bei $(x,y)=(0,0)$.

Koordinationsspiele ($a > c$ und $b < d$)

Ein Koordinationsspiel existiert, falls die Parameter a, b, c und d der Auszahlungmatrix die folgenden Bedingungen erfüllen: $a > c$ und $b < d$. Bei dieser Spielklasse existieren drei Nash-Gleichgewichte, ein gemischtes Nash-Gleichgewicht und zwei reine, symmetrische Nash-Gleichgewicht bei $(x,y)=(0,0)$ und $(x,y)=(1,1)$.

Anti-Koordinationsspiele ($a < c$ und $b > d$)

Ein Anti-Koordinationsspiel existiert, falls die Parameter a, b, c und d der Auszahlungmatrix die folgenden Bedingungen erfüllen: $a < c$ und $b > d$. Bei dieser Spielklasse existieren drei Nash-Gleichgewichte, ein gemischtes Nash-Gleichgewicht und zwei reine, unsymmetrische Nash-Gleichgewicht bei $(x,y)=(0,1)$ und $(x,y)=(0,1)$.

Ursprünge der evolutionären Spieltheorie

- Der von Maynard Smith im Jahre 1972 veröffentlichte Artikel (*J. Maynard Smith* **Game theory and the evolution of fighting**, *In "On Evolution"*, Seiten 8-28. Edingburgh University Press, Edingburgh, 1972) gilt allgemein als der erste spieltheoretische Ansatz der ***Evolutionären Spieltheorie***. Smith beschreibt in dem Artikel, wie man die biologische, evolutionäre Entwicklung von Organismen aus den Nash-Gleichgewichten von symmetrischen (2x2)-Spielen ablesen kann. Er zeigt, wie die dynamische Entwicklung der Häufigkeitsverteilung der Organismen in einem stabilen Zustand endet – der sogenannten *evolutionär stabilen Strategie*.

$$\frac{dx_i^A(t)}{dt} = \left[\underbrace{\sum_{l=1}^{m_B} \$_{il}^A \, x_l^B(t)}_{\text{Fitness der Strategie i}} - \underbrace{\sum_{l=1}^{m_B} \sum_{k=1}^{m_A} \$_{kl}^A \, x_k^A(t) \, x_l^B(t)}_{\text{Durchschn. Fitness der Population A}} \right] x_i^A(t) \qquad (1)$$

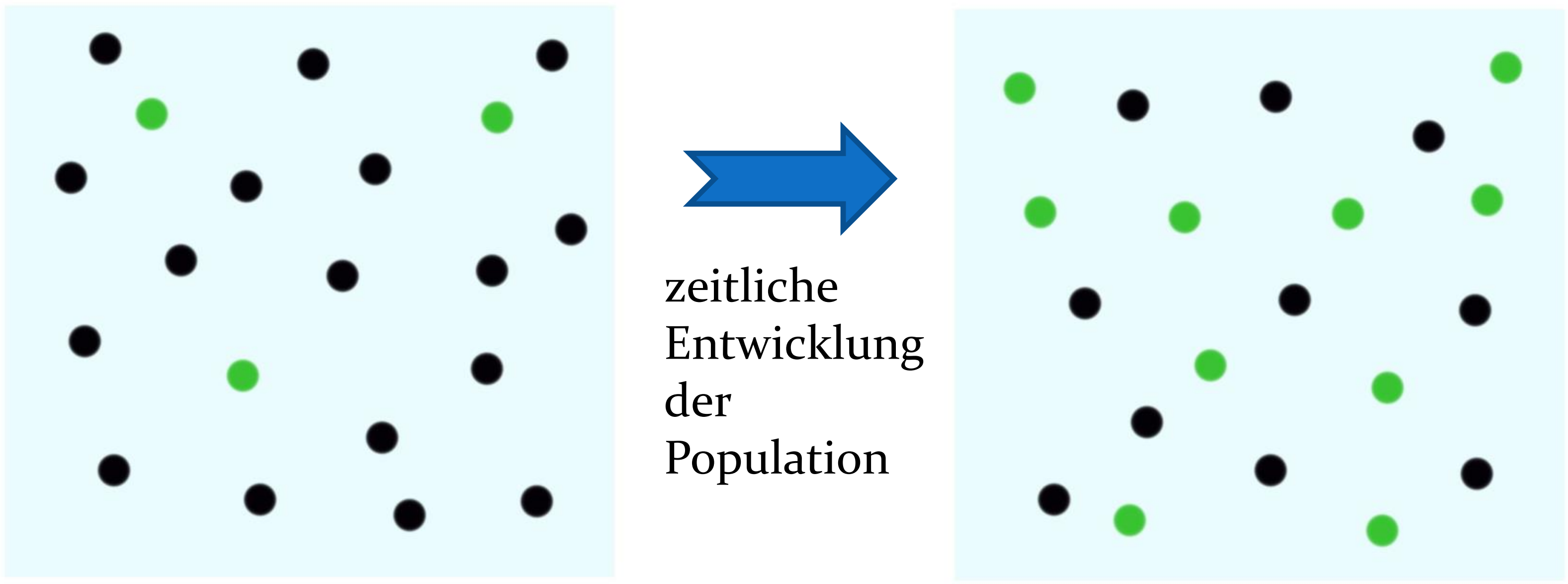
$$\frac{dx_j^B(t)}{dt} = \left[\underbrace{\sum_{l=1}^{m_A} \$_{lj}^B \, x_l^A(t)}_{\text{Fitness der Strategie j}} - \underbrace{\sum_{l=1}^{m_A} \sum_{k=1}^{m_B} \$_{lk}^B \, x_l^A(t) \, x_k^B(t)}_{\text{Durchschn. Fitness der Population B}} \right] x_j^B(t) \quad ,$$

wobei $x_i^A(t)$, $i = 1, 2, \ldots, m_A$ und $x_j^B(t)$, $j = 1, 2, \ldots, m_B$ die Anteile der in den Spielergruppen A und B zur Zeit t gewählten Strategien widerspiegeln und in der Soziobiologie den Frequenzen der *Quasispezies* entsprechen.

Evolutionäre Spieltheorie

Symmetrische (2x2)-Spiele einer Population

Die evolutionäre Spieltheorie betrachtet die zeitliche Entwicklung des strategischen Verhaltens einer gesamten Spielerpopulation.



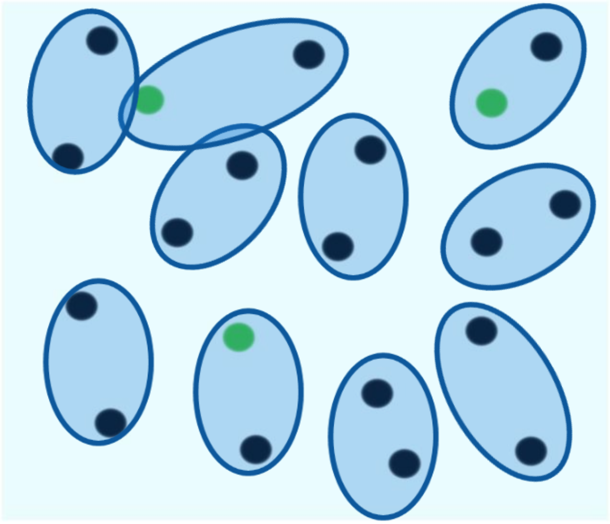
x(0)=0.15 x(10)=0.5

Mögliche Strategien: (grün , schwarz), Parameter t stellt die „Zeit“ dar.
x(t) : Anteil der Spieler, die im Zeitpunkt t die Strategie „grün“ spielen.

Evolutionäre Spieltheorie

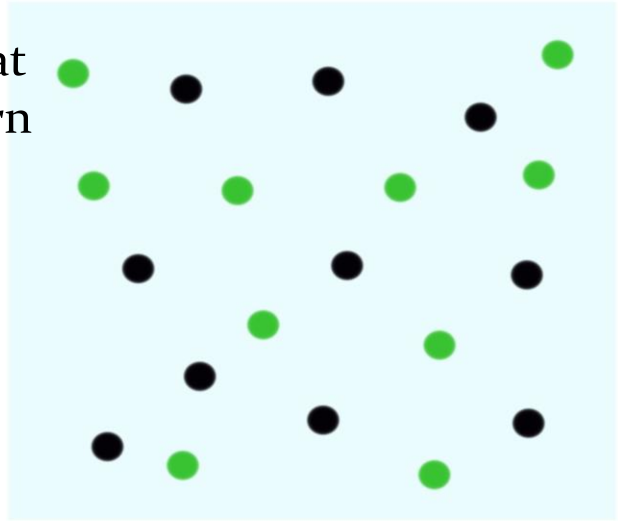
Symmetrische (2x2)-Spiele einer Population

Die einzelnen Akteure innerhalb der betrachteten Population spielen ein andauernd sich wiederholendes Spiel miteinander, wobei sich jeweils zwei Spieler zufällig treffen, das Spiel spielen und danach zu dem nächsten Spielpartner wechseln .



$x(0)=0.15$

Die Anfangspopulation von Spielern spielt zum Zeitpunkt t=0 das erste Mal das Spiel. Die Spieler wählen im Mittel zu 15% die grüne Strategie.



$x(10)=0.5$

Das evolutionäre Spiel schreitet voran und die grüne Strategie wird für die Spieler zunehmend attraktiver. Zum Zeitpunkt t=10 spielen schon 50% grün.

Nimmt man zusätzlich ein symmetrisches Spiel an ($\hat{\$} := \hat{\$}^A = \left(\hat{\$}^B\right)^T$), in welchem die Auszahlungswerte (Fitness-Werte) der Populationsgruppen gleich sind, so kann man die beiden Gruppen von ihrer mathematischen Struktur her als ununterscheidbare Spielergruppen mit identischen Populationsvektoren $x(t) = y(t)$ annehmen. Die Differentialgleichung schreibt sich dann wie folgt:

$$\frac{dx(t)}{dt} = \left[(\$_{11} - \$_{21})(x - x^2) + (\$_{12} - \$_{22})(1 - 2x + x^2)\right] \; x(t) \; =: \; g(x) \qquad (3)$$

Verallgemeinert man diese Differentialgleichung wieder auf mehr als zwei Strategien, so kann man abkürzend die folgende Formulierung schreiben:

$$\frac{d\vec{x}}{dt} = \hat{\mathbf{x}} \left(\hat{\$}\, \vec{x}\right) - \left(\left(\hat{\$}\, \vec{x}\right)^T \vec{x}\right) \vec{x}$$

Evolutionär stabile Strategien (ESS)

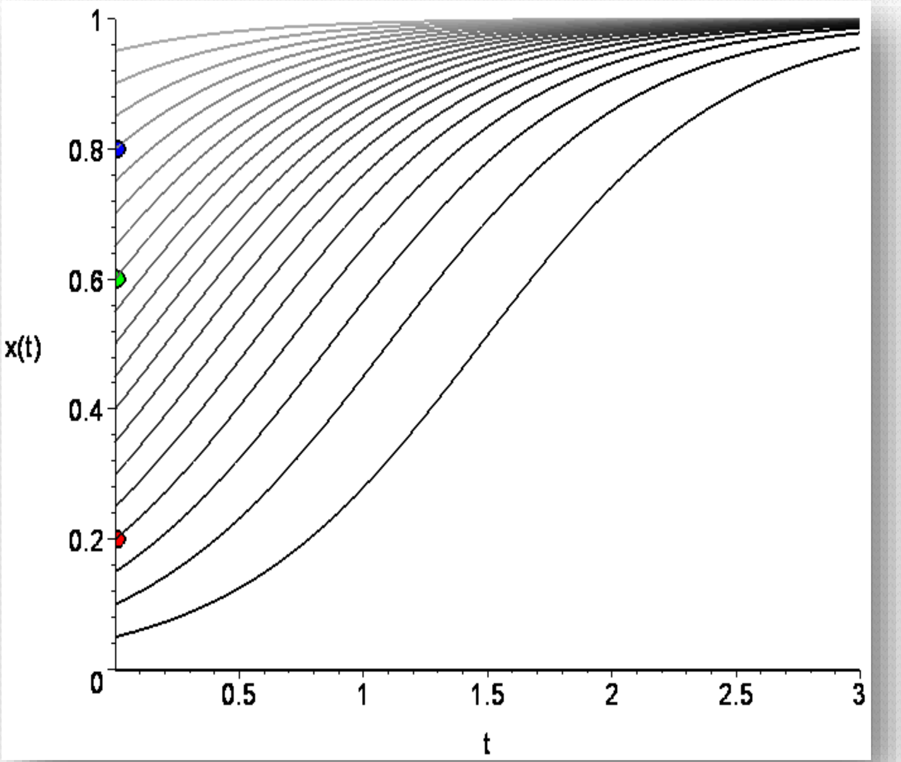
- Evolutionär stabile Strategien sind die stabilen Endzustände der Häufigkeitsverteilung x(t):

$$\underset{t\to\infty}{Limes}\left(x(t)\right)$$

Eine notwendige (aber nicht hinreichende) Bedingung für die Existenz einer ESS ist, dass diese ein Nash – Gleichgewicht des zugrundeliegenden Spiels ist.

Beispiel:
Gefangenendilemma ähnliche Spiele
Für die ESS des evolutionären Gefangenendilemma – Spiels ergibt sich:

$$\underset{t\to\infty}{Limes}\left(x(t)\right)=1 \quad \Rightarrow \quad \text{alle "gestehen"}$$



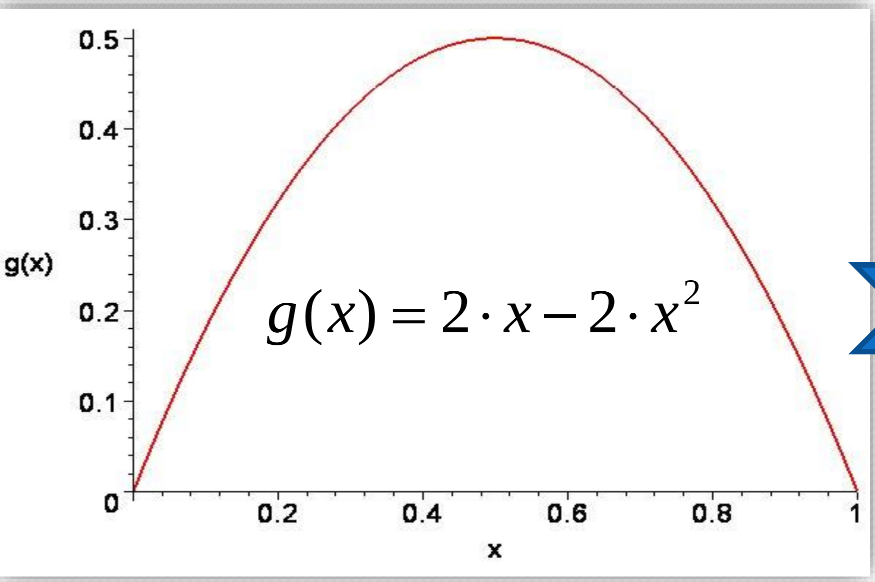
Replikatordynamik
(für das Gefangenendilemma)

	Ge	Sc
Ge	(-7 , -7)	(-1 , -9)
Sc	(-9 , -1)	(-3 , -3)

Die Differentialgleichung der Replikatordynamik für das Gefangenendilemma lautet:

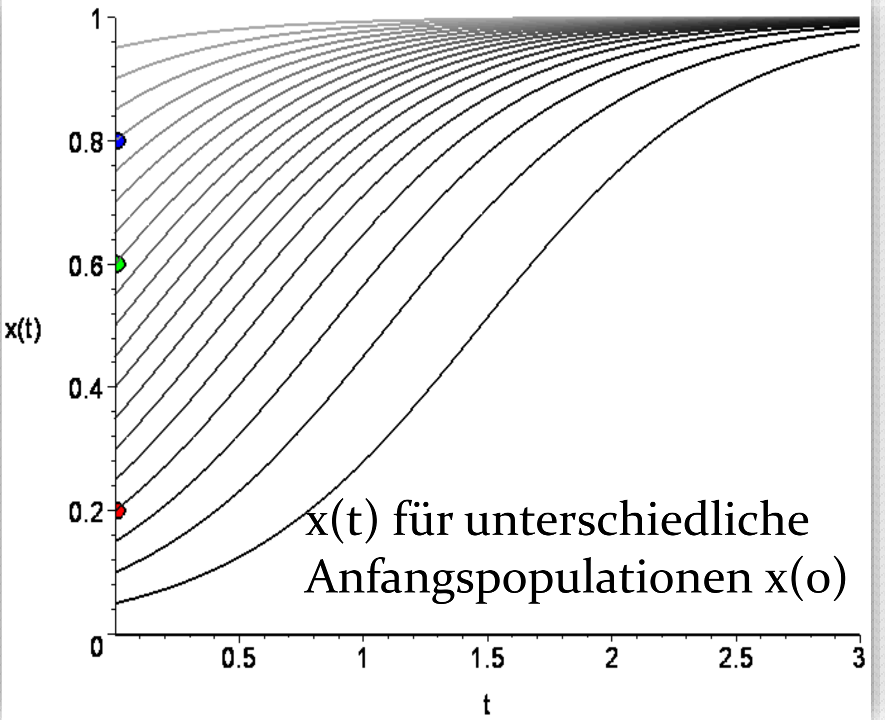
$$\dot{x}(t) := \frac{dx(t)}{dt} = 2 \cdot x(t) - 2 \cdot \left(x(t)\right)^2 = g(x(t))$$

$$g(x(t)) := x(t) \cdot \left((-7-(-9)) \cdot (x(t) - x(t)^2) + (-3-(-1)) \cdot (2 \cdot x(t) - 1 - x(t)^2)\right)$$



$$g(x) = 2 \cdot x - 2 \cdot x^2$$

Beispiel: Gefangenendilemma
g(x)=g(x(t)) im Bereich [0,1] dargestellt



x(t) für unterschiedliche Anfangspopulationen x(0)

Replikatordynamik

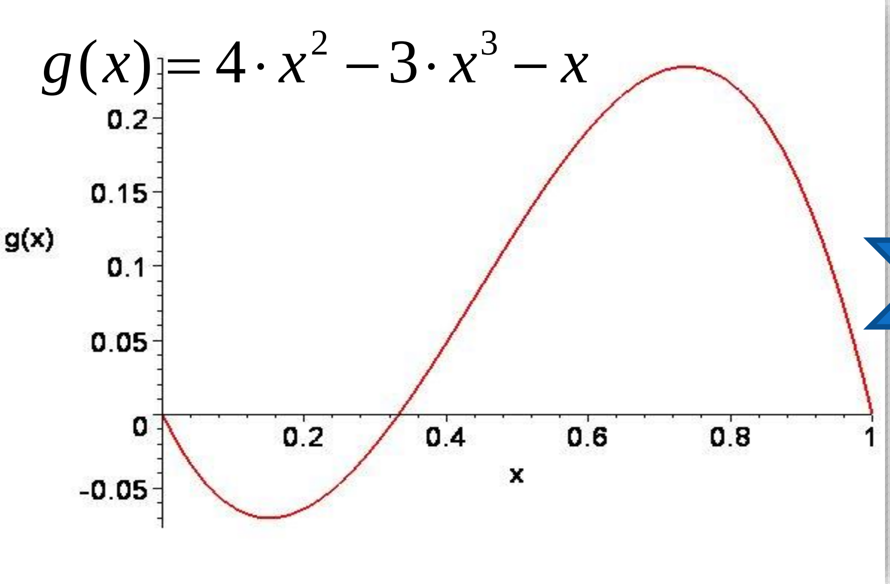
(für das Hirschjagt-Spiel)

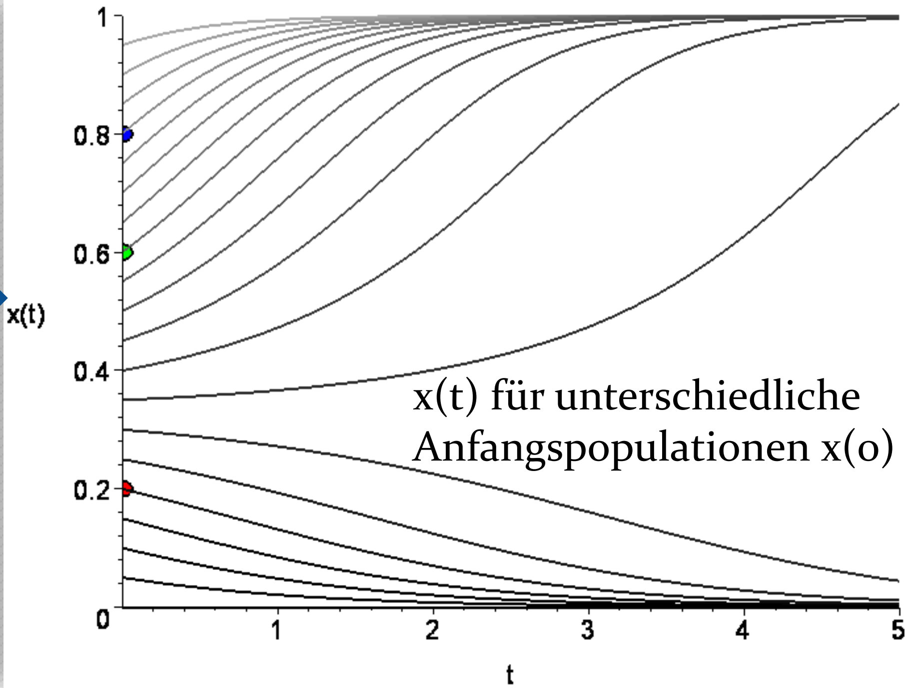
	Hasen	Hirsch
Hasen	(2 , 2)	(4 , 0)
Hirsch	(0 , 4)	(5 , 5)

Die Differentialgleichung der Replikatordynamik
für das Hirschjagt-Spiel lautet:

$$\dot{x}(t) := \frac{dx(t)}{dt} = 4 \cdot (x(t))^2 - 3 \cdot (x(t))^3 - x(t) - = g(x(t))$$

$$g(x(t)) := x(t) \cdot \left((2-0) \cdot (x(t) - x(t)^2) + (5-4) \cdot (2 \cdot x(t) - 1 - x(t)^2)\right)$$





Beispiel: Hirschjagt-Spiel
g(x)=g(x(t)) im Bereich [0,1] dargestellt

Replikatordynamik

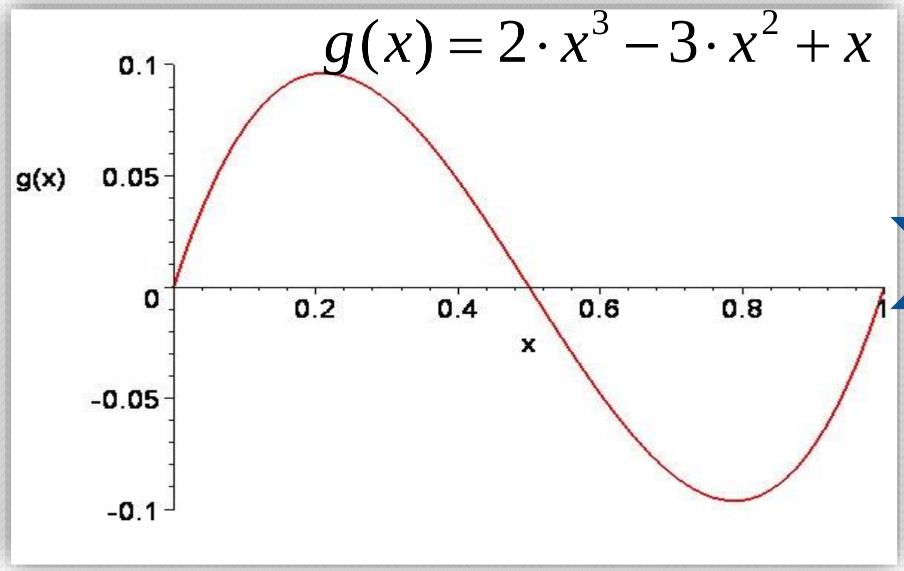
(für das Angsthasen-Spiel)

	Springe nicht	Springe
Springe nicht	(-1 , -1)	(2 , 0)
Springe	(0 , 2)	(1 , 1)

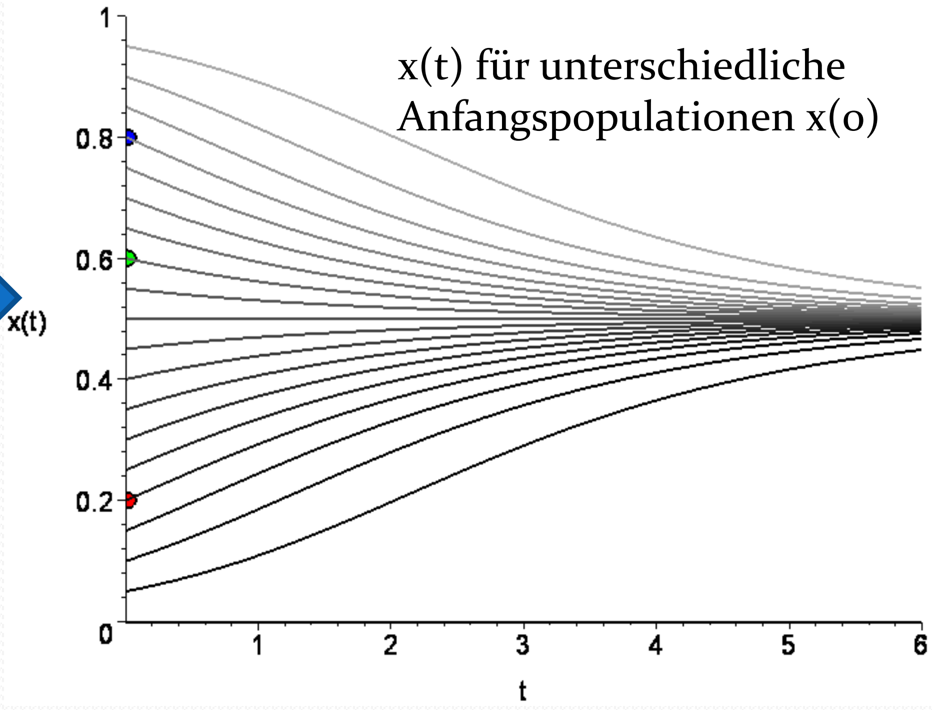
Die Differentialgleichung der Replikatordynamik für das Angsthasen-Spiel lautet:

$$\dot{x}(t) := \frac{dx(t)}{dt} = 2 \cdot (x(t))^3 - 3 \cdot (x(t))^2 + x(t) = g(x(t))$$

$$g(x(t)) := x(t) \cdot \left((-1-0) \cdot (x(t) - x(t)^2) + (1-2) \cdot (2 \cdot x(t) - 1 - x(t)^2) \right)$$



Beispiel: Angsthasen-Spiel
g(x)=g(x(t)) im Bereich [0,1] dargestellt



Evolutionär Stabile Strategien

- Betrachten Sie die folgenden Beispiele:

Beispiel 1:

	Kugel	Keine Kugel
Kugel	(0 , 0)	(2 , -1)
Keine Kugel	(-1 , 2)	(1 , 1)

Beispiel 2:

	Kugel	Keine Kugel
Kugel	(-1 , -1)	(3 , 0)
Keine Kugel	(0 , 3)	(1 , 1)

Beispiel 3:

	Kugel	Keine Kugel
Kugel	(0 , 0)	(2 , -2)
Keine Kugel	(-2 , 2)	(3 , 3)

1. Geben Sie mögliche dominante Strategien und Nash-Gleichgewichte der Spiele an.
2. Bestimmen und zeichnen Sie die Funktion g(x) für alle drei Spiele?
3. Berechnen Sie die Nullstellen der Funktion g(x) (g(x)=0).
4. Geben Sie die evolutionär stabilen Strategien der Spiele an?

Dominante Strategien und Nash-Gleichgewichte

Beispiel 1	Kugel	Keine Kugel
Kugel	(0 , 0)	(2 , -1)
Keine Kugel	(-1 , 2)	(1 , 1)

Das erste Spiel besitzt nur ein Nash-Gleichgewicht das gleichzeitig die dominante Strategie des Spiels ist (Kugel, Kugel). Das Spiel gehört zur Klasse der dominanten Spiele.

Beispiel 2	Kugel	Keine Kugel
Kugel	(-1 , -1)	(3 , 0)
Keine Kugel	(0 , 3)	(1 , 1)

Das zweite Spiel besitzt keine dominante Strategie, aber zwei unsymmetrische Nash-Gleichgewicht in reinen Strategien ((K,KK) und (KK,K)) und ein gemischtes Nash-Gleichgewicht (0.67 K , 0.33 KK). Das Spiel gehört zur Klasse der Anti-Koordinationsspiele.

Beispiel 3	Kugel	Keine Kugel
Kugel	(0 , 0)	(2 , -2)
Keine Kugel	(-2 , 2)	(3 , 3)

Das dritte Spiel besitzt ebenfalls keine dominante Strategie, aber zwei symmetrische Nash-Gleichgewicht in reinen Strategien ((K,K) und (KK,KK)) und ein gemischtes Nash-Gleichgewicht (0.33 K , 0.67 KK). Das Spiel gehört zur Klasse der Koordinationsspiele.

Nullstellen der Funktion g(x)

Beispiel 1:

$$g(x) = x - x^2$$

$$x - x^2 = 0$$

$$x \cdot (1 - x) = 0 \quad \Rightarrow \quad x_1 = 0$$

$$1 - x = 0 \quad \Rightarrow \quad x_2 = 1$$

Die Funktion hat zwei Nullstellen:

$$x_1 = 0 \;\text{ und }\; x_2 = 1$$

Beispiel 3:
Die Funktion hat drei Nullstellen:

$$g(x) = -3 \cdot x^3 + 4 \cdot x^2 - x$$

$$x_1 = 0 \;,\; x_2 = 1 \,\text{und}\; x_3 = \frac{1}{3}$$

Beispiel 2:

$$g(x) = 3 \cdot x^3 - 5 \cdot x^2 + 2 \cdot x$$

$$x \cdot \left(3 \cdot x^2 - 5 \cdot x + 2\right) = 0 \quad \Rightarrow \quad x_1 = 0$$

$$3 \cdot x^2 - 5 \cdot x + 2 = 0 \;\Big|\; /3$$

$$x^2 - \frac{5}{3} \cdot x + \frac{2}{3} = 0 \;\Big|\; \text{p - q Formel}$$

$$x_{2/3} = \frac{5}{6} \pm \sqrt{\left(\frac{5}{6}\right)^2 - \frac{2}{3}}$$

$$x_{2/3} = \frac{5}{6} \pm \sqrt{\frac{25 - 24}{36}}$$

$$x_{2/3} = \frac{5}{6} \pm \frac{1}{6} \quad \Rightarrow \quad x_2 = 1,\; x_3 = \frac{2}{3}$$

Die Funktion hat drei Nullstellen:

$$x_1 = 0 \;,\; x_2 = 1 \,\text{und}\; x_3 = \frac{2}{3}$$

	Kugel	Keine Kugel
Kugel	(0 , 0)	(2 , -1)
Keine Kugel	(-1 , 2)	(1 , 1)

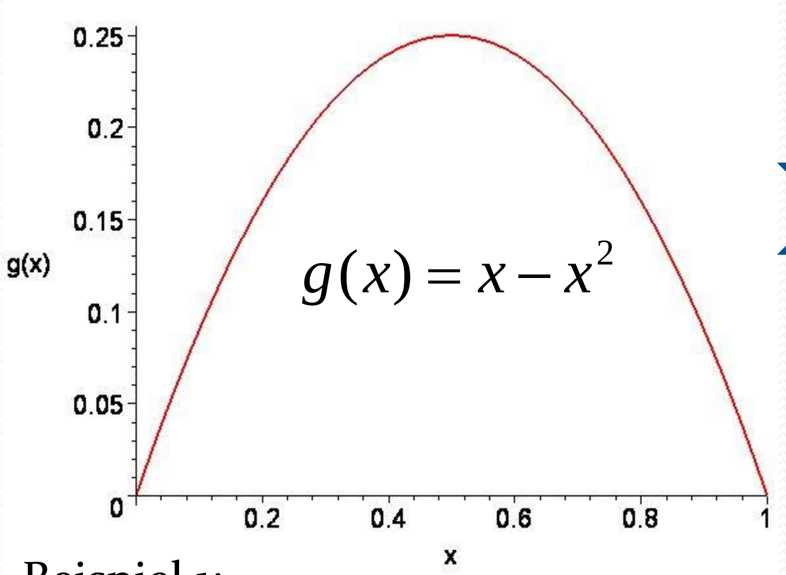
Evolutionäre Strategien (Beispiel 1)

Die Differentialgleichung der Replikatordynamik für das erste Beispiel lautet:

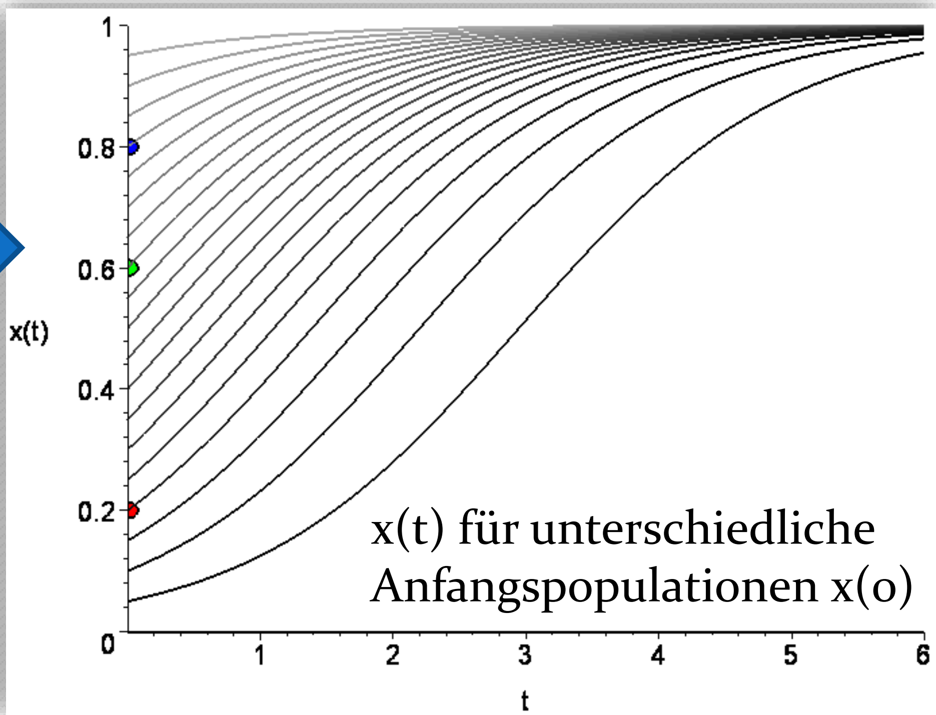
$$\frac{dx(t)}{dt} = g(x(t)) = x(t) - (x(t))^2$$

Da es sich bei diesem Beispiel um ein dominantes, symmetrisches (2x2)-Spiel handelt und die Funktion g(x) im relevanten Bereich (x=[0,1]) immer größer-gleich Null ist, strebt der Populationsanteil der Kugel-Spieler unabhängig vom Anfangswert immer gegen 1.

$\boxed{\text{Eine ESS bei x=1}}$



Beispiel 1:
g(x)=g(x(t)) im Bereich [0,1] dargestellt



x(t) für unterschiedliche Anfangspopulationen x(0)

Evolutionäre Strategien (Beispiel 2)

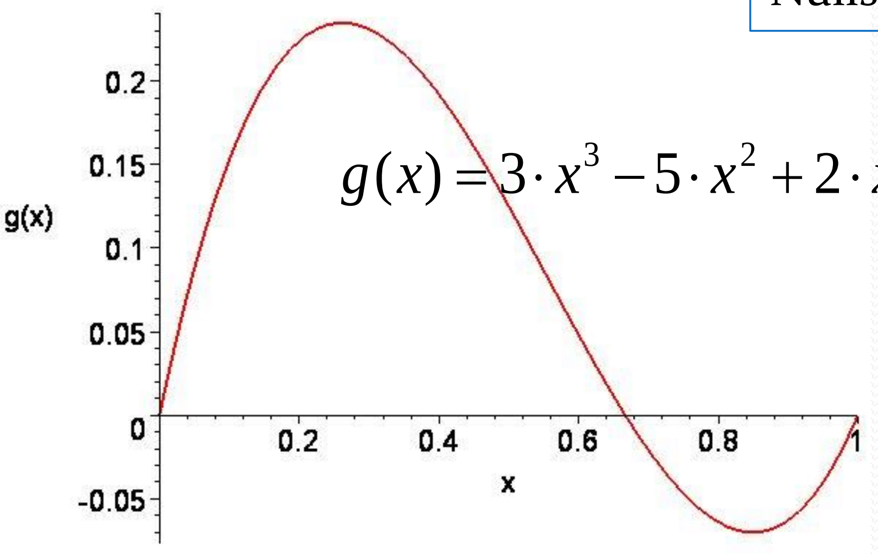
	Kugel	Keine Kugel
Kugel	(-1 , -1)	(3 , 0)
Keine Kugel	(0, 3)	(1 , 1)

Die Differentialgleichung der Replikatordynamik für das zweite Beispiel lautet:

$$\frac{dx(t)}{dt} = g(x(t)) = 3\cdot(x(t))^3 - 5\cdot(x(t))^2 + 2\cdot x(t)$$

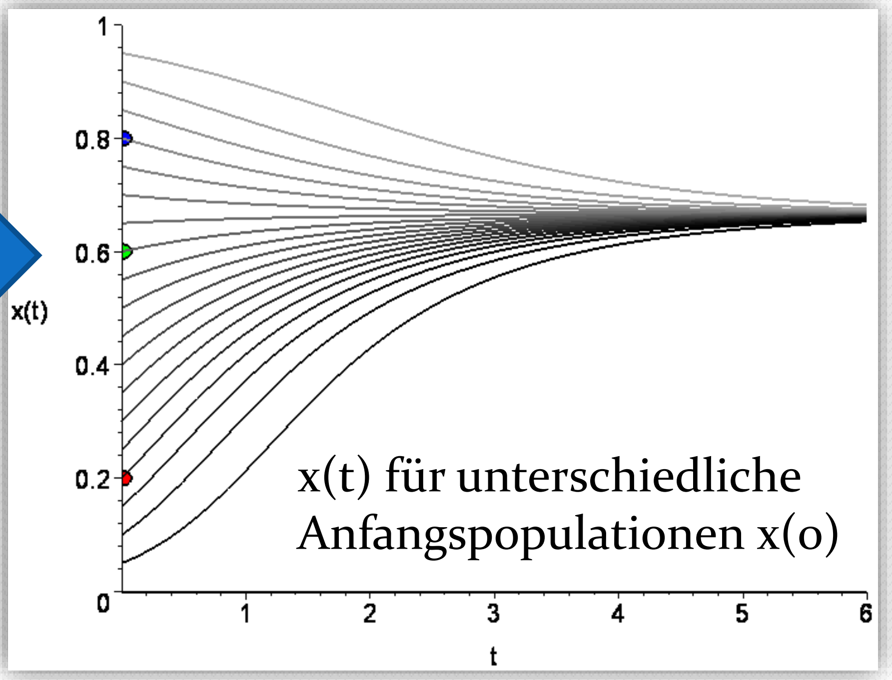
Da es sich bei diesem Beispiel um ein symmetrisches Anti-Koordinationsspiel handelt, strebt der Populationsanteil der Kugel-Spieler unabhängig vom Anfangswert immer zu dem gemischten Nash-Gleichgewicht, was identisch mit der mittleren Nullstelle der Funktion g(x) ist (x=0.67).

$\boxed{\text{Eine ESS bei x=0.67}}$



$$g(x) = 3\cdot x^3 - 5\cdot x^2 + 2\cdot x$$

Beispiel 2:
g(x)=g(x(t)) im Bereich [0,1] dargestellt



Evolutionäre Strategien (Beispiel 3)

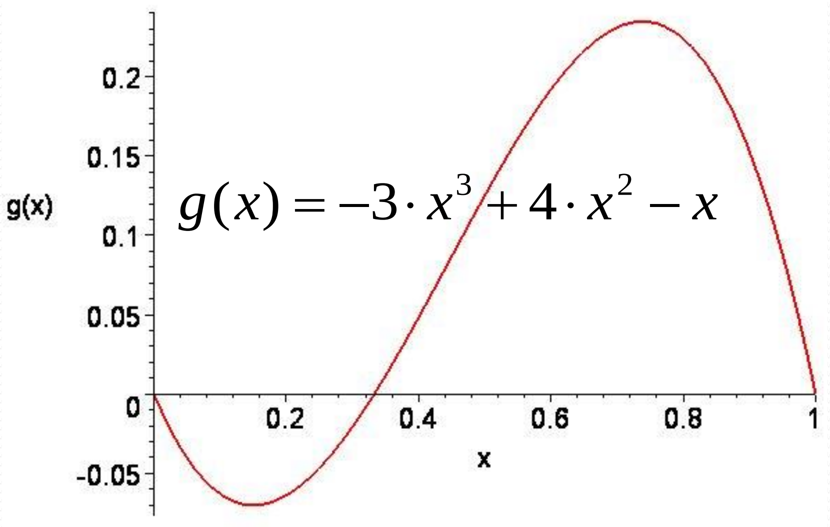
	Kugel	Keine Kugel
Kugel	(0, 0)	(2 , -2)
Keine Kugel	(-2, 2)	(3 , 3)

Die Differentialgleichung der Replikatordynamik für das zweite Beispiel lautet:

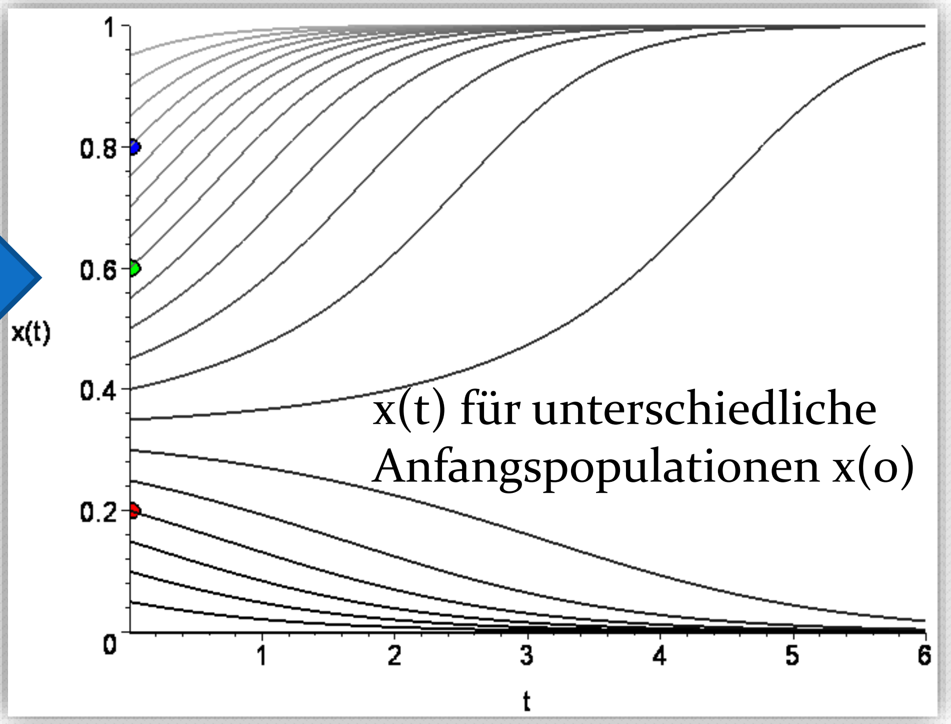
$$\frac{dx(t)}{dt} = g(x(t)) = -3\cdot(x(t))^3 + 4\cdot(x(t))^2 - x(t)$$

$$\boxed{\text{Zwei ESSs : (x=1 und x=0)}}$$

Da es sich bei diesem Beispiel um ein symmetrisches Koordinationsspiel handelt, strebt der Populationsanteil der Kugel-Spieler abhängig vom Anfangswert zu einem der beiden reinen Nash-Gleichgewichte (x=1 oder x=0).



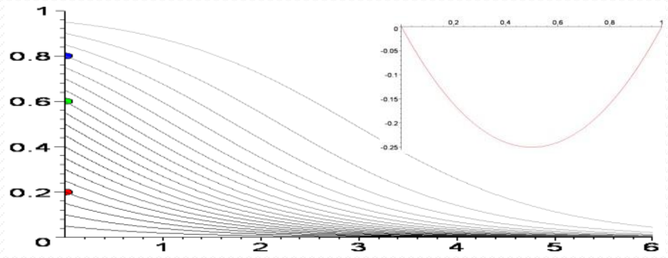
Beispiel 3:
g(x)=g(x(t)) im Bereich [0,1] dargestellt



Klassifizierung von evolutionären, symmetrischen (2x2)-Spielen

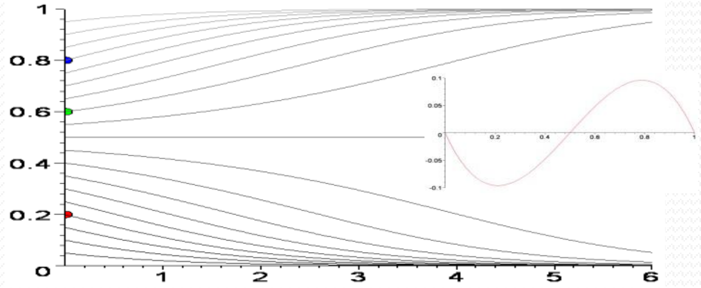
- **Dominante Spiele**
  (2. Strategie dominiert 1.Strategie)

  Es existiert ein Nash - Gleichgewicht, welches die anderen Strategien dominiert. ESS bei x=0.



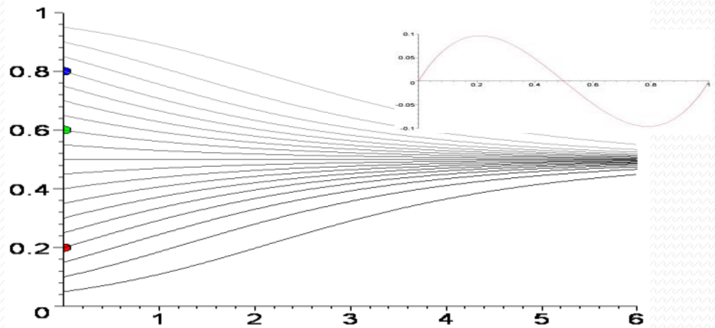
- **Koordinationsspiele**

  Es existieren drei Nash – Gleichgewichte und zwei reine ESS, die abhängig von der Anfangsbedingung realisiert werden.



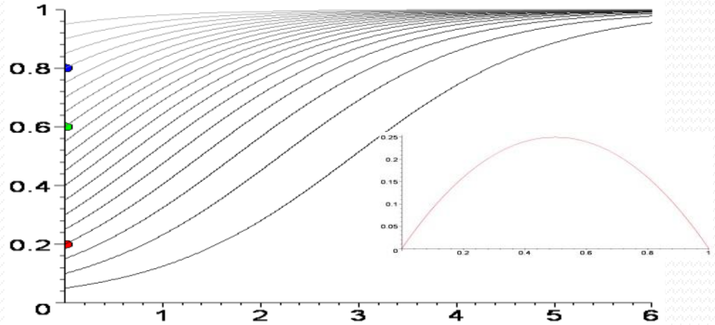
- **Anti – Koordinationsspiele**

  Es existieren drei Nash – Gleichgewichte aber nur eine gemischte ESS, die unabhängig von der Anfangsbedingung realisiert wird.



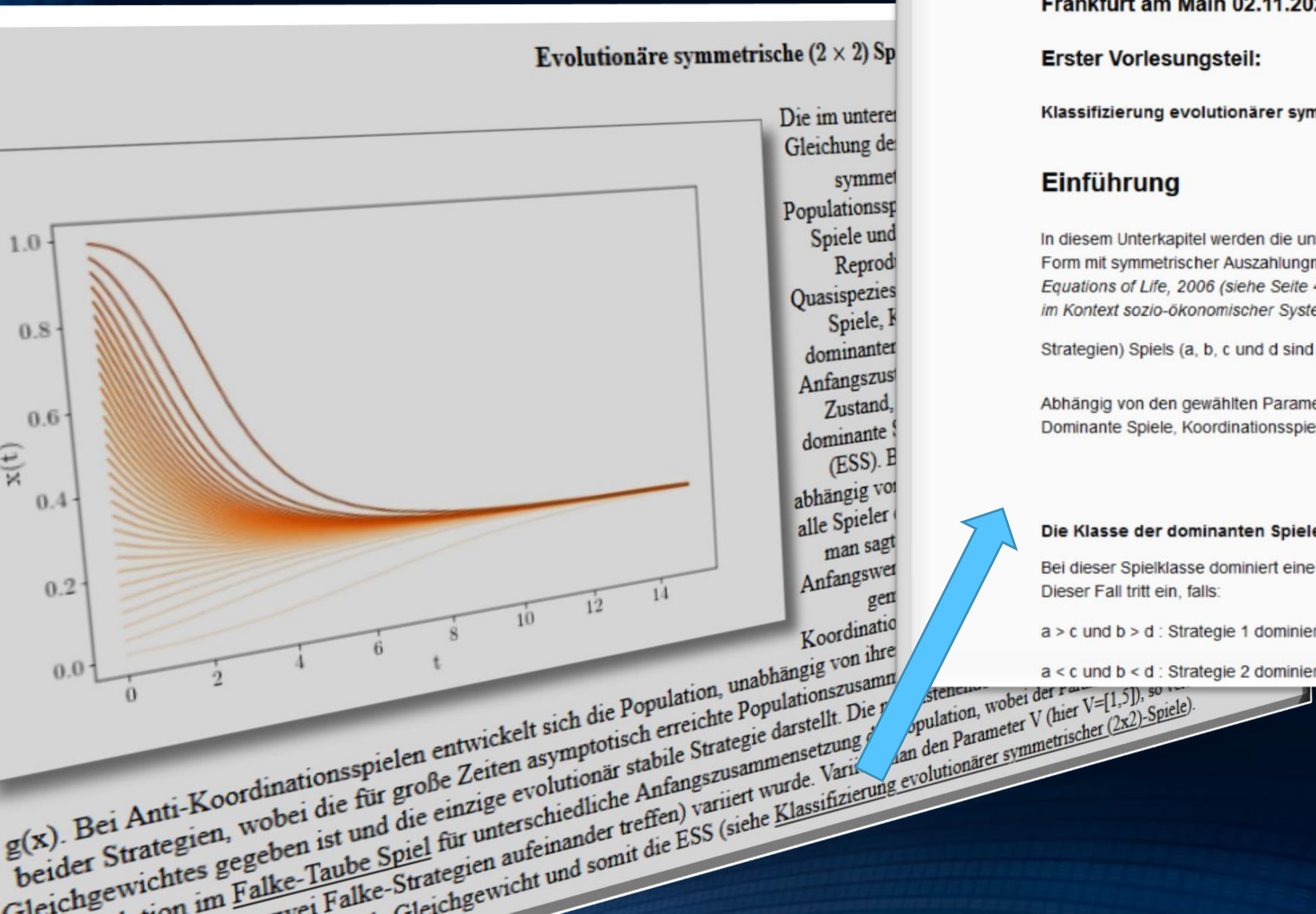
- **Dominante Spiele**
  (1. Strategie dominiert 2.Strategie)

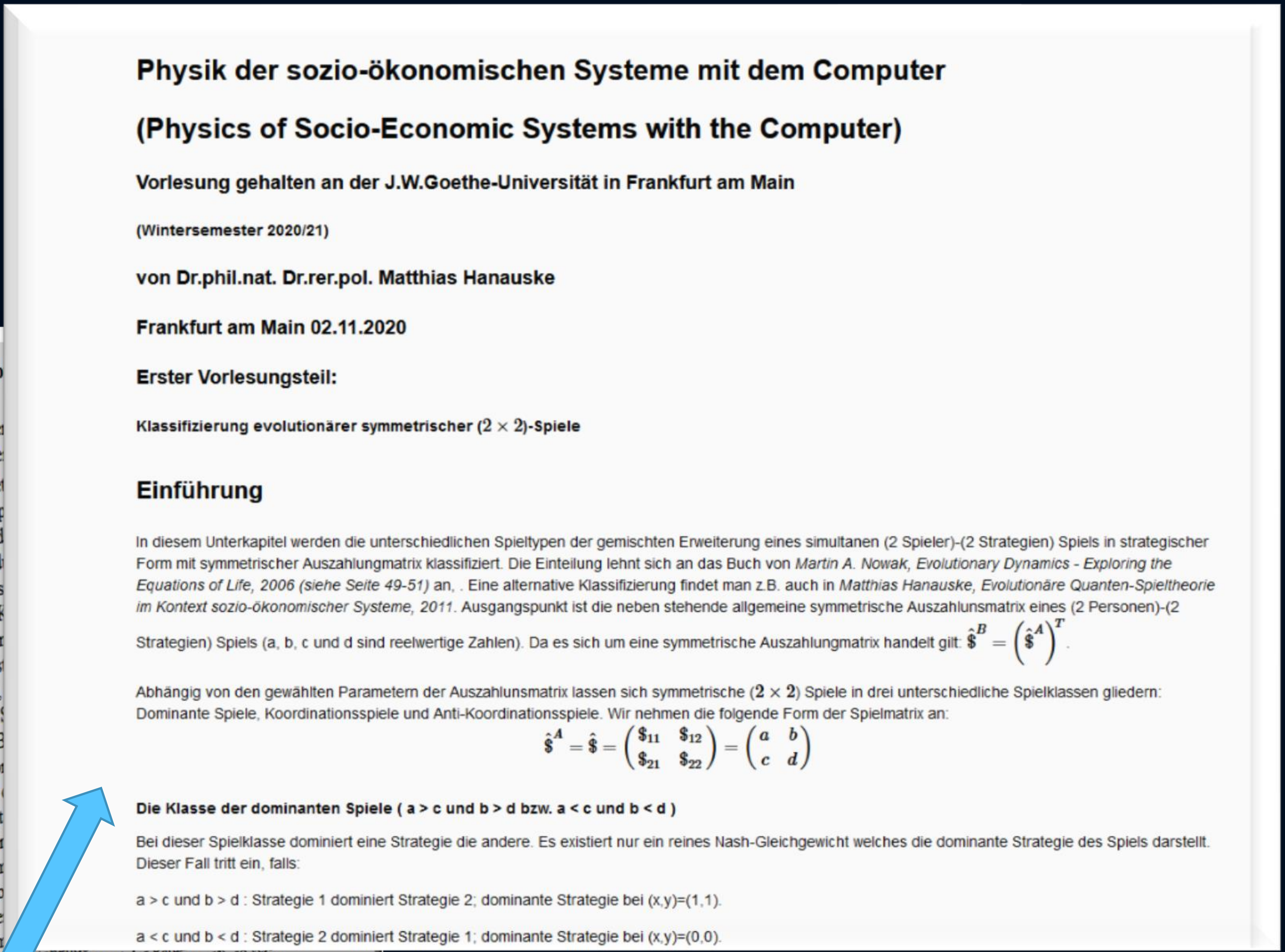
  Es existiert ein Nash - Gleichgewicht, welches die anderen Strategien dominiert. ESS bei x=1.



Jupyter Notebook

Auf der Internetseite der Vorlesung





Physik der sozio-ökonomischen Systeme mit dem Computer

(Physics of Socio-Economic Systems with the Computer)

Vorlesung gehalten an der J.W.Goethe-Universität in Frankfurt am Main

(Wintersemester 2020/21)

von Dr.phil.nat. Dr.rer.pol. Matthias Hanauske

Frankfurt am Main 02.11.2020

Erster Vorlesungsteil:

Klassifizierung evolutionärer symmetrischer (2×2)-Spiele

Einführung

In diesem Unterkapitel werden die unterschiedlichen Spieltypen der gemischten Erweiterung eines simultanen (2 Spieler)-(2 Strategien) Spiels in strategischer Form mit symmetrischer Auszahlungmatrix klassifiziert. Die Einteilung lehnt sich an das Buch von *Martin A. Nowak, Evolutionary Dynamics - Exploring the Equations of Life, 2006 (siehe Seite 49-51)* an, . Eine alternative Klassifizierung findet man z.B. auch in *Matthias Hanauske, Evolutionäre Quanten-Spieltheorie im Kontext sozio-ökonomischer Systeme, 2011*. Ausgangspunkt ist die neben stehende allgemeine symmetrische Auszahlunsmatrix eines (2 Personen)-(2 Strategien) Spiels (a, b, c und d sind reelwertige Zahlen). Da es sich um eine symmetrische Auszahlungmatrix handelt gilt: $\hat{\$}^B = \left(\hat{\$}^A\right)^T$.

Abhängig von den gewählten Parametern der Auszahlunsmatrix lassen sich symmetrische $(\mathbf{2} \times \mathbf{2})$ Spiele in drei unterschiedliche Spielklassen gliedern: Dominante Spiele, Koordinationsspiele und Anti-Koordinationsspiele. Wir nehmen die folgende Form der Spielmatrix an:

$$\hat{\$}^A = \hat{\$} = \begin{pmatrix} \$_{11} & \$_{12} \\ \$_{21} & \$_{22} \end{pmatrix} = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$$

Die Klasse der dominanten Spiele (a > c und b > d bzw. a < c und b < d)

Bei dieser Spielklasse dominiert eine Strategie die andere. Es existiert nur ein reines Nash-Gleichgewicht welches die dominante Strategie des Spiels darstellt. Dieser Fall tritt ein, falls:

a > c und b > d : Strategie 1 dominiert Strategie 2; dominante Strategie bei (x,y)=(1,1).

a < c und b < d : Strategie 2 dominiert Strategie 1; dominante Strategie bei (x,y)=(0,0).

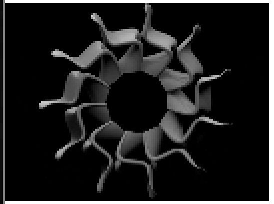
Zusätzlich befindet sich das Jupyter Notebook auch auf der OLAT Seite des Kurses

Maple Worksheet

Auf der Internetseite der Vorlesung

Download
Maple Worksheet

Weiterführende Links



- Folien der 4.Vorlesung
- View Jupyter Notebook: Evolutionäre Spieltheorie symmetrischer (2x2)-Spiele
- Download Jupyter Notebook: Evolutionäre Spieltheorie symmetrischer (2x2)-Spiele
- Download Python Programm: Evolutionäre Spieltheorie symmetrischer (2x2)-Spiele
- View Jupyter Notebook: Evolutionäre Spieltheorie unsymmetrischer (2x2)-Spiele
- Download Jupyter Notebook: Evolutionäre Spieltheorie unsymmetrischer (2x2)-Spiele
- Download Python Programm: Evolutionäre Spieltheorie unsymmetrischer (2x2)-Spiele
- Maple Worksheet: Klassifizierung von symmetrischen evolutionären (2x2)-Spielen
- Maple Worksheet: Evolutionäre Bi-Matrix Spiele

Evolution1.mw

Physik der sozio-ökonomischen Systeme mit dem Computer

Physics of Socio-Economic Systems with the Computer

Vorlesung gehalten an der J.W.Goethe-Universität in Frankfurt am Main (Wintersemester 2017/18)

von Dr.phil.nat. Dr.rer.pol. Matthias Hanauske

Frankfurt am Main 22.08.2017

Erster Vorlesungsteil:

Klassifizierung von symmetrischen evolutionären (2 x 2)-Spielen

Einführung

Dieses Maple-Worksheet basiert auf dem Maple-Worksheet Spielklassen1.mw und erweitert dieses in einen evolutionären, zeitabhängigen Kontext.Ausgangspunkt ist wiederum die folgende allgemeine symmetrische Auszahlunsmatrix eines (2 Personen)-(2 Strategien) Spiels (a, b, c und d sind reelwertige Zahlen):

```
> restart:
  with(plots):
  with(plottools):
```

Lösen des evolutionären Spiels mit Maple

(Vorlage2.mw)

```
> restart:
  with(LinearAlgebra):
  with(plots):
```

Definition der Auszahlunsmatrix für Spieler A (USA) im Spiel des Wettrüstens:

```
> D_A11:=1:
  D_A12:=4:
  D_A21:=0:
  D_A22:=2:
  D_A:=Matrix(2,2,[D_A11,D_A12,D_A21,D_A22]);
```

Da sich das Spiel um ein symmetrisches (2 Personen)-(2 Strategien) Spiel handelt, erhält man die Auszahlunsmatrix für Spieler B (Nord Korea) durch die transponierte Matrix des Spielers A:

```
> D_B:=Transpose(D_A);
```

Wir betrachten im folgenden die evolutionäre Erweiterung des Dilemma des Wettrüstens. Die Population bestehe aus den Entscheidungsträgern von vielen Ländern und zu jedem Zeitpunkt treffen sie erneut die Entscheidung "Aufrüsten" oder "Abrüsten". Die Differentialgleichung, die die zeitliche Entwicklung des Populationsvektors x(t) (Anteil der Länder die "Aufrüsten") beschreibt lautet:

```
> DGL:=diff(x(t),t)=simplify(((D_A11-D_A21)*(x(t)-x(t)^2) + (D_A12-D_A22)*(1-2*x(t)+x
  (t)^2))*x(t));
  g:=simplify(((D_A11-D_A21)*(x-x^2) + (D_A12-D_A22)*(1-2*x+x^2))*x);
```

Die die zeitliche Entwicklung bestimmende Funktion g(x) besitzt das folgende Aussehen:

```
> plot(g, x=0..1);
```

Lösen des evolutionären Spiels mit Maple

(Vorlage2.mw)

x(t), der Anteil der Spieler (Länder) die zum Zeitpunkt t die Strategie 1 ("Aufrüsten") spielen , hängt neben der Funktion g(x) von dem Anfangswert x(t=0) ab.Setzt man z.B. x(t=0)=0.1 (entspricht einer Anfangspopulation von 10% der Länder rüstet auf und 90% rüstet ab) so kann man die Differentialgleichung analytisch Lösen (nicht immer möglich):

```
> LoesDGL:=simplify(dsolve({DGL,x(0)=0.1}));
```

Darstellen der Lösung:

```
> plot(rhs(LoesDGL),t=0..4);
```

Ausgabe des Wertes zu einer festen Zeit (z.B. t=2)

```
> evalf(subs({t=2},rhs(LoesDGL)));
```

Falls eine analytische Lösung nicht möglich ist kann man wie folgt die DGL numerisch lösen und grafisch darstellen:

```
> LoesDGL:=dsolve({DGL,x(0)=0.1},x(t),type=numeric):
  odeplot(LoesDGL,[t,x(t)],0..4);
```

Ausgabe des Wertes zu einer festen Zeit (z.B. t=2)

```
> LoesDGL(2);
```

Python Programm

Auf der Internetseite der Vorlesung

```
import numpy as np
import matplotlib.pyplot as plt
import matplotlib

# Definition der Funktion g
def g(x,a,b,c,d):
  g=((a-c)*(x-x*x) + (b-d)*(1-2*x+x*x))*x
  return g

# Loesen der DGL
def solve_dgl(numpoints,tend,x0,a,b,c,d):
  sol=np.empty([numpoints,2])
  t=np.linspace(0,tend,numpoints)
  dt=t[1]-t[0]
  sol[0].flat[0] = t[0]
  sol[0].flat[1] = x0
  i = 1
  while i < len(sol):
    sol[i].flat[0] = t[i]
    dx=g(sol[i-1,1],a,b,c,d)*dt
    sol[i].flat[1] = sol[i-1,1] + dx
    i = i + 1
  return sol

# Groessenfestlegung der Labels usw. im Bild
params = {
    'text.usetex'       : True,
    'axes.titlesize' : 22,
    'axes.labelsize' : 20,
    'xtick.labelsize' : 20 ,
    'ytick.labelsize' : 20
}
matplotlib.rcParams.update(params)

#Festlegung der Auszahlungsmatrix des symmetrischen (2x2)-Spiels
# Dominant Game
#a=-7
#b=-1
#c=-9
```

Weiterführende Links

- Folien der 4.Vorlesung
- View Jupyter Notebook: Evolutionäre Spieltheorie symmetrischer (2x2)-Spiele
- Download Jupyter Notebook: Evolutionäre Spieltheorie symmetrischer (2x2)-Spiele
- Download Python Programm: Evolutionäre Spieltheorie symmetrischer (2x2)-Spiele
- View Jupyter Notebook: Evolutionäre Spieltheorie unsymmetrischer (2x2)-Spiele
- Download Jupyter Notebook: Evolutionäre Spieltheorie unsymmetrischer (2x2)-Spiele
- Download Python Programm: Evolutionäre Spieltheorie unsymmetrischer (2x2)-Spiele
- Maple Worksheet: Klassifizierung von symmetrischen evolutionären (2x2)-Spielen
- Maple Worksheet: Evolutionäre Bi-Matrix Spiele

Lösen des evolutionären Spiels mit Python

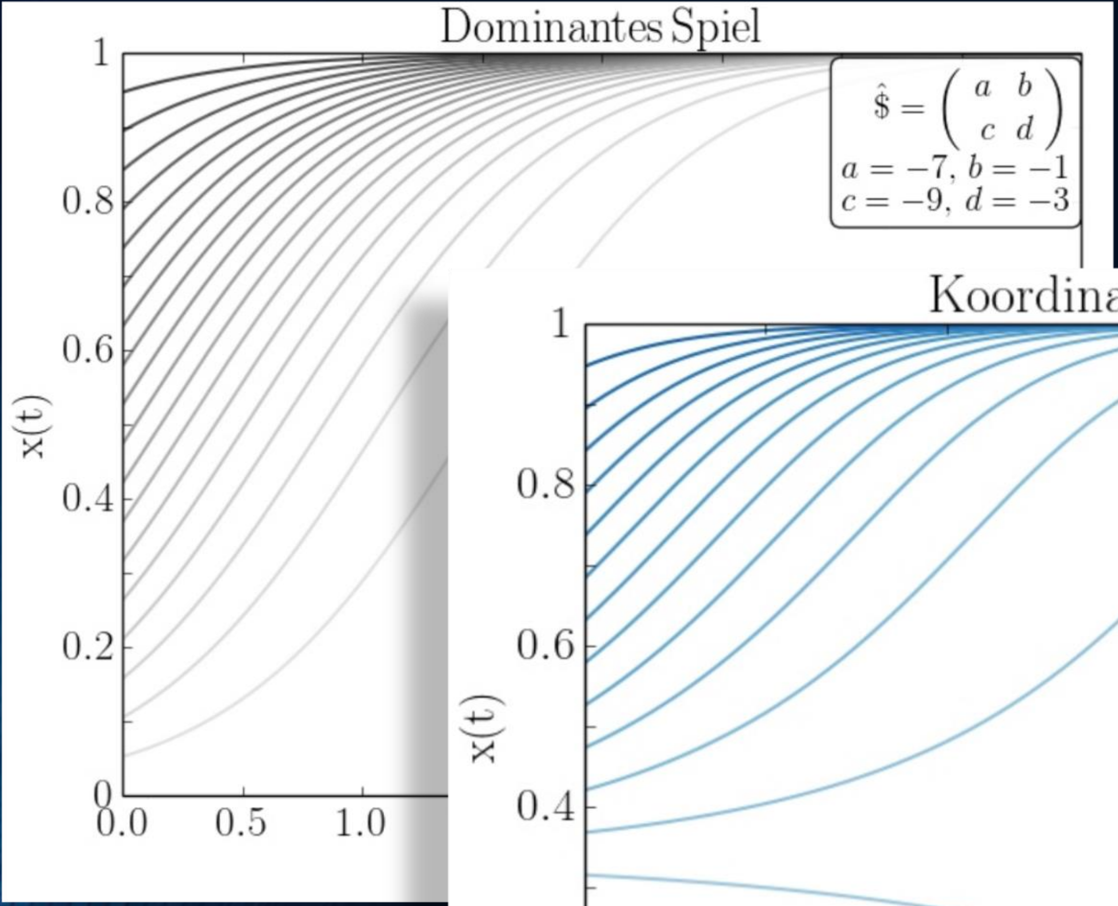
```python
import numpy as np
import matplotlib.pyplot as plt
import matplotlib

# Groessenfestlegung der Labels usw. im Bild
params = {
    'text.usetex'       : True,
    'axes.labelsize'   : 20,
    'xtick.labelsize' : 20,
    'ytick.labelsize' : 20
}
matplotlib.rcParams.update(params)

# Definition der Funktion g
def g(x,a,b,c,d):
  g=((a-c)*(x-x*x) + (b-d)*(1-2*x+x*x))*x
  return g

#Festlegung der Auszahlungsmatrix des symmetrischen (2x2)-Spiels
a=2
b=4
c=0
d=5
#End(zeit)punkt, Anzahl der Zeitschritte
tend=6
numpoints=500
#Anfangswert der Population
x0=0.4
```

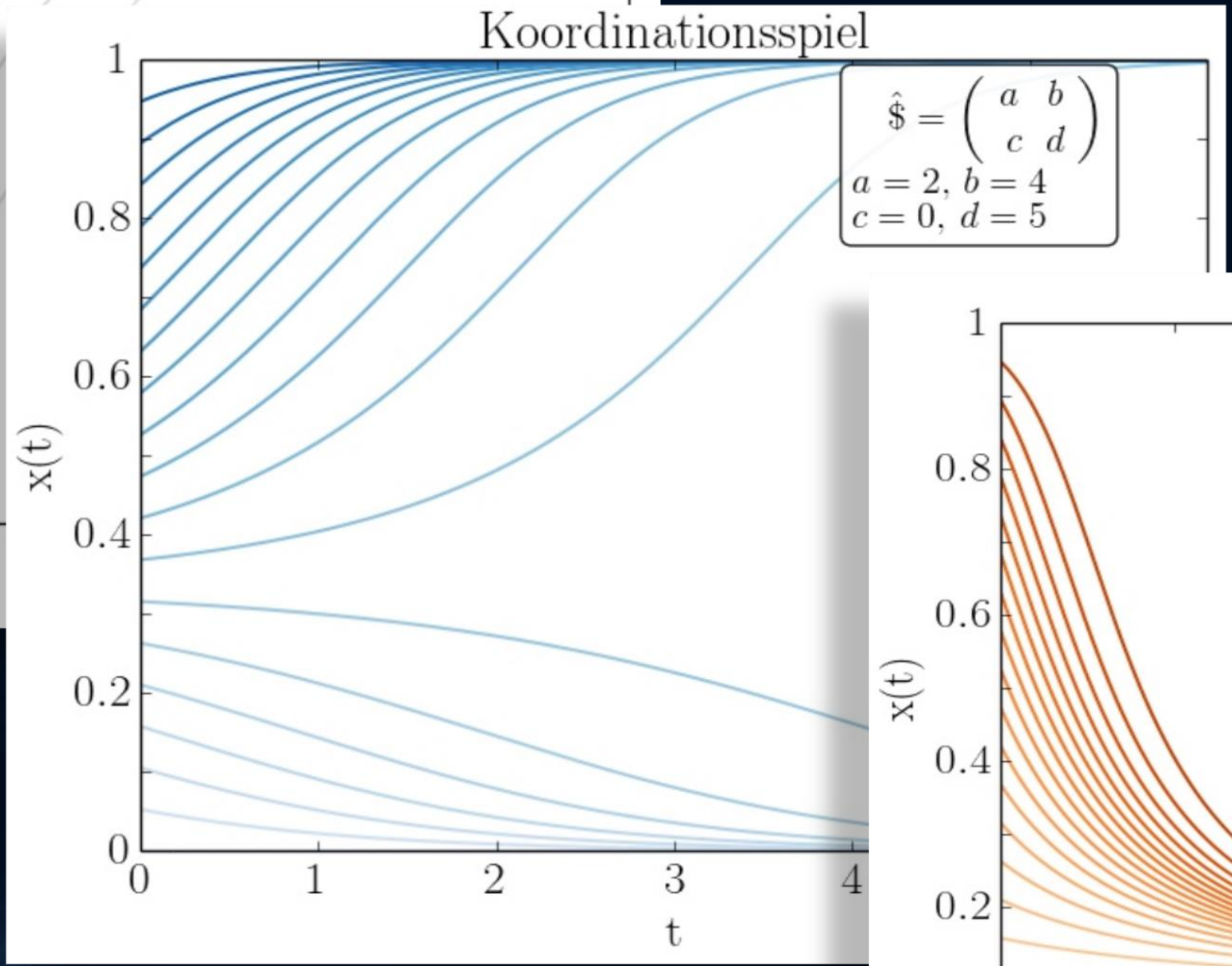
Lösen des evolutionären Spiels mit Python

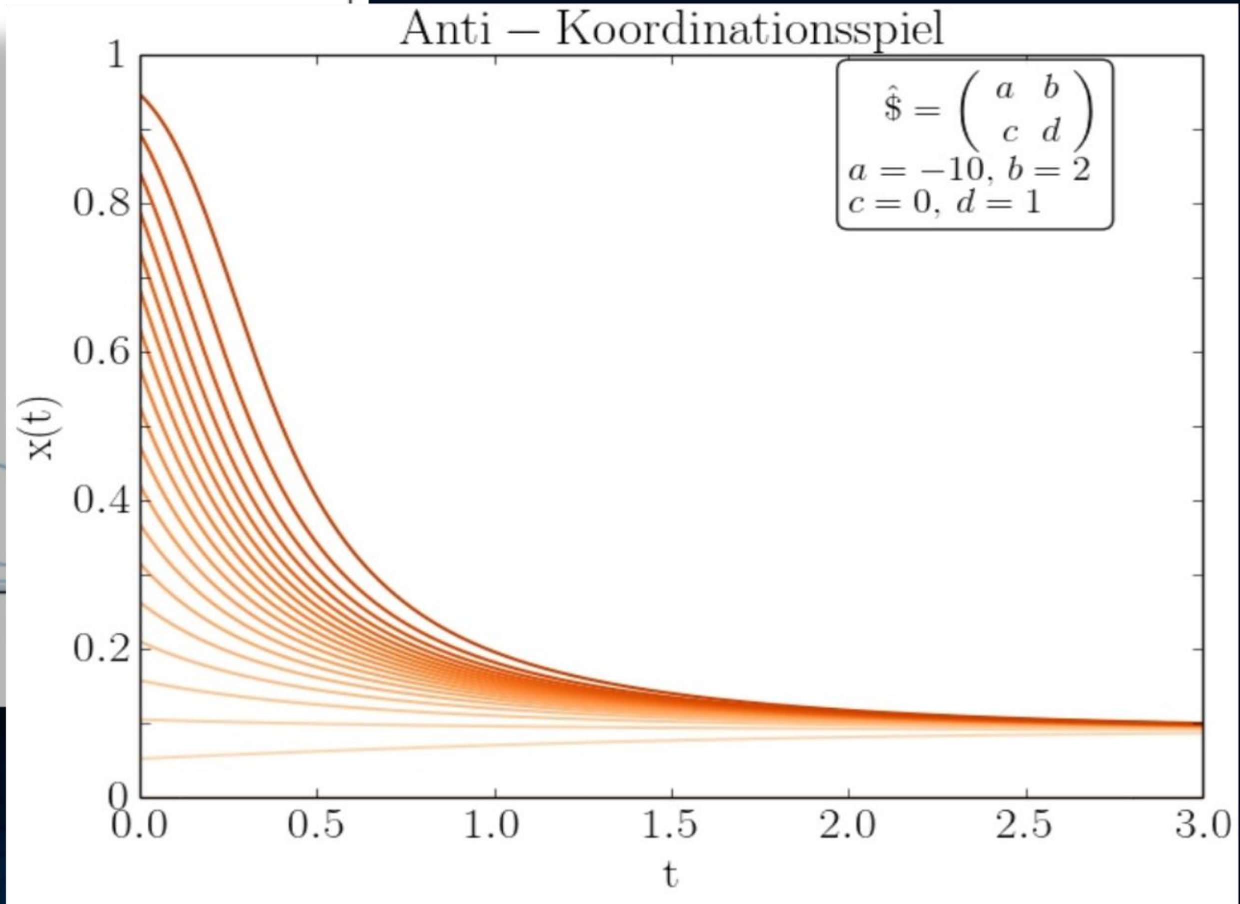
```
#Loesen der DGL
Loes=np.empty([numpoints,2])
t=np.linspace(0,tend,numpoints)
dt=t[1]-t[0]
Loes[0].flat[0] = t[0]
Loes[0].flat[1] = x0
i = 1
while i < len(Loes):
    Loes[i].flat[0] = t[i]
    dx=g(Loes[i-1,1],a,b,c,d)*dt
    Loes[i].flat[1] = Loes[i-1,1] + dx
    i = i + 1

#Plotten des Bildes
plt.plot(Loes[:,0],Loes[:,1],c="black", linewidth=1.5, linestyle='-')

# Achsenbeschriftungen usw.
plt.ylim(0,1)
plt.ylabel(r"$\rm x(t)$")
plt.xlabel(r"$\rm t$")

#Speichern des Bildes als .jpg--Datei
saveFig="./evol.jpg"
plt.savefig(saveFig, dpi=100,bbox_inches="tight",pad_inches=0.05,format="jpg")
plt.show()
```

Euler-Verfahren

Lösen des evolutionären Spiels mit Python Version evol1.py



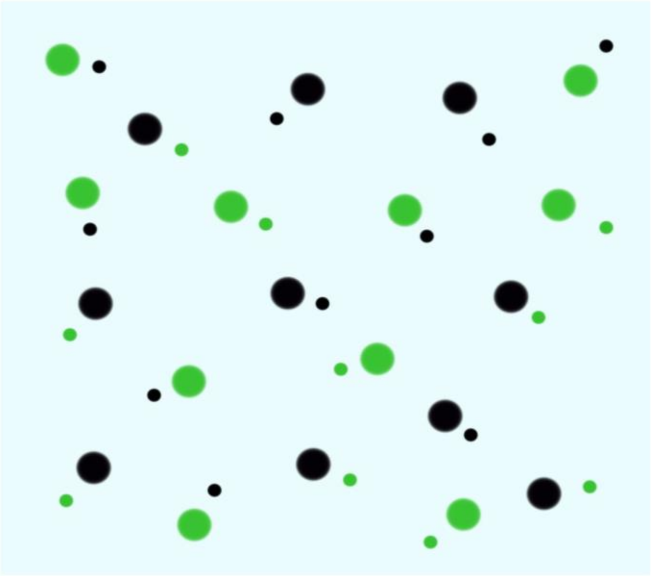




Evolutionäre Spieltheorie

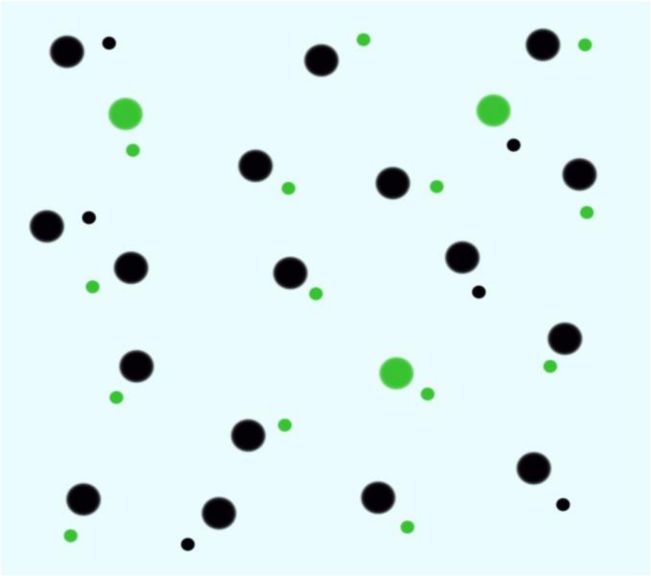
Unsymmetrische (2x2)-Spiele (Bimatrixspiele) zweier Populationen

Bei unsymmetrischen (2x2)-Spielen besteht die zugrundeliegende Population aus zwei Gruppen (hier große und kleine Kreise). Aufgrund der unterschiedlichen Auszahlungsmatrizen können die Populationsgruppen sich in ihren Strategieentscheidungen (grün , schwarz) unterschiedlich entwickeln.



$x(0)=0.5$, $y(0)=0.5$

zeitliche Entwicklung der Population



$x(10)=0.15$, $y(10)=0.7$

Mögliche Strategien: (grün , schwarz), Parameter t stellt die „Zeit“ dar.

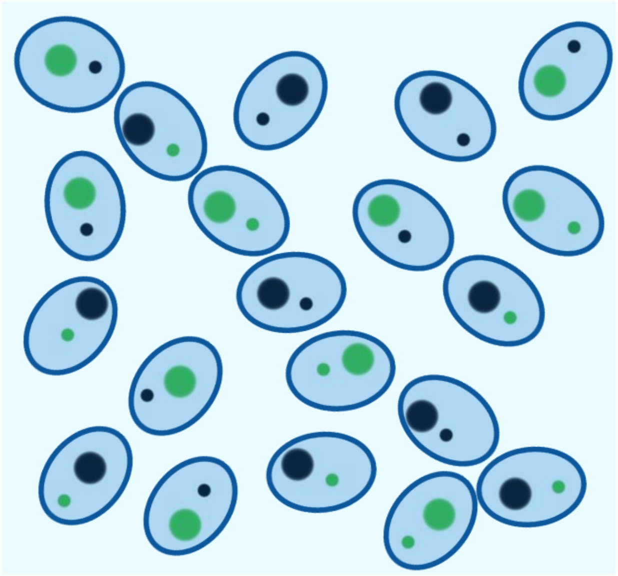
$x(t)$: Anteil der großen Spieler, die im Zeitpunkt t die Strategie „grün“ spielen.

$y(t)$: Anteil der kleinen Spieler, die im Zeitpunkt t die Strategie „grün“ spielen.

Evolutionäre Spieltheorie

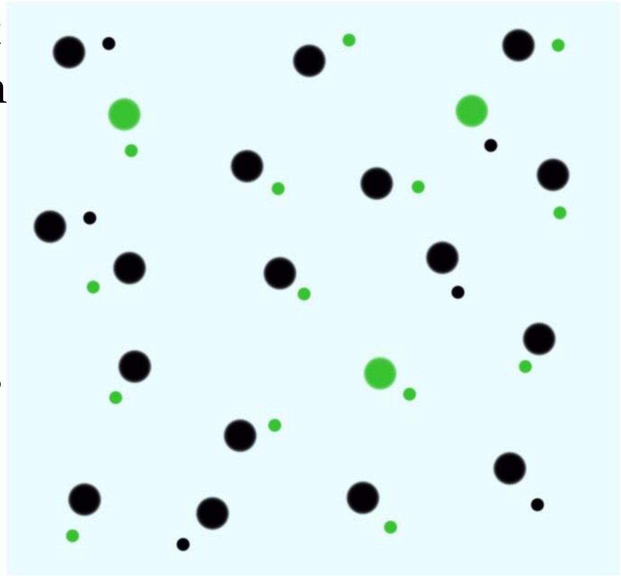
Unsymmetrische (2x2)-Spiele (Bimatrixspiele)

Die einzelnen Akteure innerhalb der betrachteten gesamten Population spielen ein andauernd sich wiederholendes Spiel miteinander, wobei sich jeweils zwei Spieler mit unterschiedlichen Gruppenzugehörigkeiten zufällig treffen, das Spiel spielen und danach zu dem nächsten Spielpartner der anderen Gruppe wechseln .



x(10)=0.5 , y(10)=0.5

Die Anfangspopulation von Spielern spielt zum Zeitpunkt t=0 das erste Mal das Spiel. Es bilden sich stets Zweier-Gruppen aus großen und kleinen Kreisen.



x(10)=0.15 , y(10)=0.7

Das evolutionäre Spiel schreitet voran und die grüne Strategie wird für die kleinen Spieler zunehmend attraktiver (y(10)=0.7), wohingegen sie für die großen Spieler zunehmend weniger attraktiv wird (x(10)=0.15).

Wir beschränken uns im folgenden auf den 2-Strategien Fall ($m_A = m_B = 2$), lassen jedoch weiter eine Unsymmetrie der Auszahlungsmatrix zu. Die beiden Komponenten der zweidimensionalen gruppenspezifischen Populationsvektoren lassen sich dann, aufgrund ihrer Normalisierungsbedingung, auf eine Komponente reduzieren ($x_2^A = 1 - x_1^A$ und $x_2^B = 1 - x_1^B$). Das zeitliche Verhalten der Komponenten der Populationsvektoren (Gruppe A: $x(t) := x_1^A(t)$ und Gruppe B: $y(t) := x_1^B(t)$) wird in der Reproduktionsdynamik mittels des folgenden Systems von Differentialgleichungen beschrieben (siehe z.B. [4], S:116 oder [3], S:69):

$$\frac{dx(t)}{dt} = \left[\left(\$_{11}^A + \$_{22}^A - \$_{12}^A - \$_{21}^A \right) \, y(t) + \left(\$_{12}^A - \$_{22}^A \right) \right] \left(x(t) - (x(t))^2 \right) \;=:\; g_A(x,y) \qquad (2)$$

$$\frac{dy(t)}{dt} = \left[\left(\$_{11}^B + \$_{22}^B - \$_{12}^B - \$_{21}^B \right) \, x(t) + \left(\$_{12}^B - \$_{22}^B \right) \right] \left(y(t) - (y(t))^2 \right) \;=:\; g_B(x,y)$$

Beispiel 1:

Kampf der Geschlechter als evolutionäres Spiel

	Kino	Disko
Kino	(1 , 3)	(0 , 0)
Disko	(0 , 0)	(3 , 1)

Das gekoppelte System von Differentialgleichung lautet:

$$\frac{dx(t)}{dt} = x(t)\cdot\left(4\cdot y(t) - 4\cdot x(t)\cdot y(t) + 3\cdot x(t) - 3\right)$$

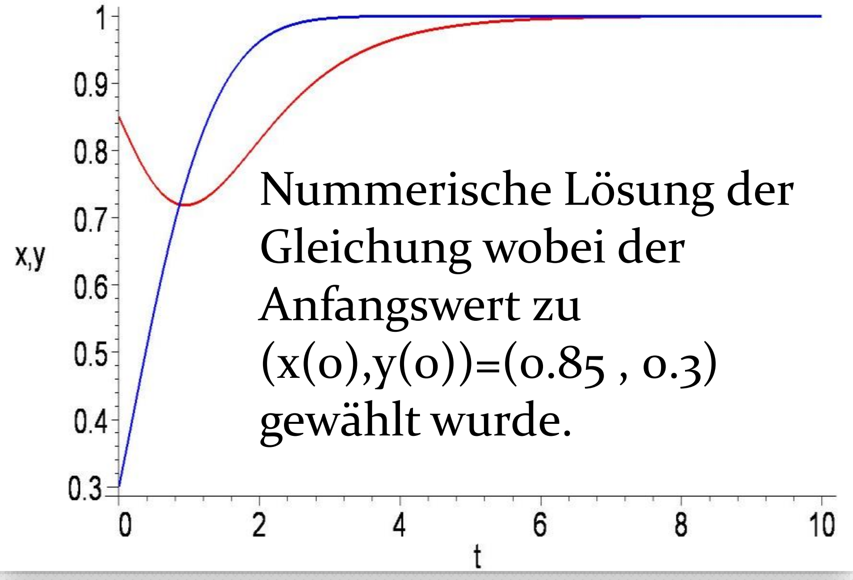
$$\frac{dy(t)}{dt} = y(t)\cdot\left(4\cdot x(t) - 4\cdot x(t)\cdot y(t) + y(t) - 1\right)$$

Die Lösung der Gleichung erfolgt wiederum durch Integration bzw. kann mithilfe des Computers nummerisch berechnet werden. Es muss in beiden Fällen ein fester Anfangswert (x(0),y(0)) gewählt werden.

Die beiden Gruppen der Population sind Männer (A, x(t))und Frauen (B, y(t)). Da nur zwei Strategien wählbar sind, lassen sich die jeweiligen Populationsanteile durch lediglich eine Größe ausdrücken (x(t) und y(t)).

$$x(t) := x_1^A(t) \;\;,\;\; x_2^A(t) = 1 - x(t)$$

$$y(t) := x_1^B(t) \;\;,\;\; x_2^B(t) = 1 - y(t)$$



Nummerische Lösung der Gleichung wobei der Anfangswert zu (x(0),y(0))=(0.85 , 0.3) gewählt wurde.

Bi-Matrix Spiele
Gemischte Nash-Gleichgewichte

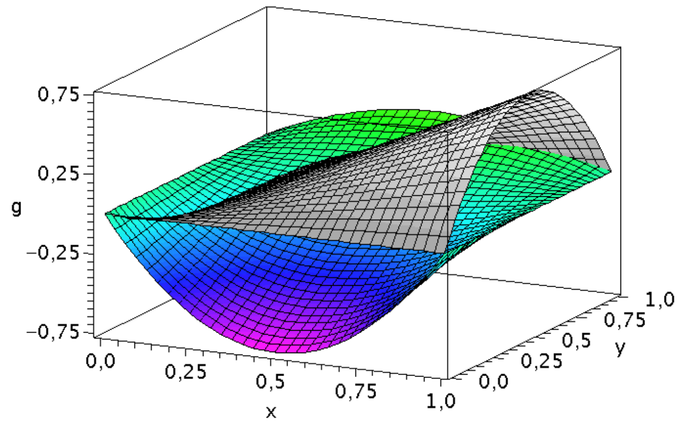
Nash-Gleichgewicht in gemischten Strategien:

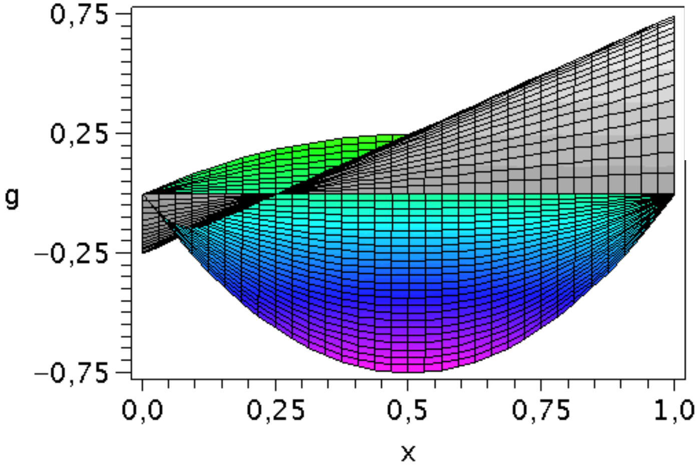
$$\left.\frac{\partial \tilde{\$}^{A}(\tilde{s}^{A},\tilde{s}^{B})}{\partial \tilde{s}^{A}}\right|_{\tilde{s}^{B}=\tilde{s}^{B\star}} = 0 \quad \forall \;\; \tilde{s}^{A} \in [0,1] \;\; , \;\; \tilde{s}^{B\star} \in]0,1[$$

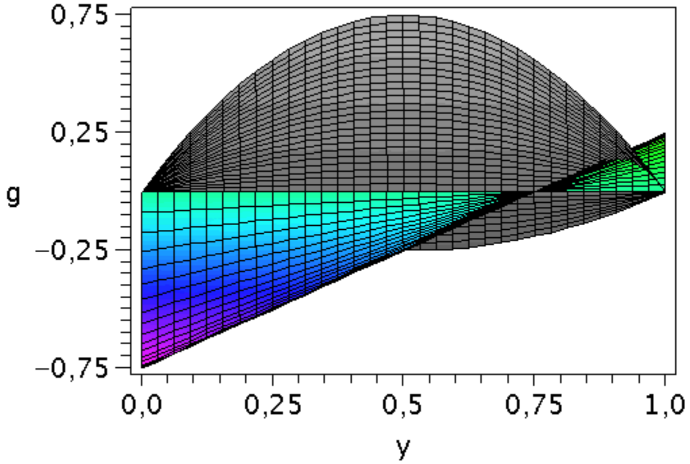
$$\left.\frac{\partial \tilde{\$}^{B}(\tilde{s}^{A},\tilde{s}^{B})}{\partial \tilde{s}^{B}}\right|_{\tilde{s}^{A}=\tilde{s}^{A\star}} = 0 \quad \forall \;\; \tilde{s}^{B} \in [0,1] \;\; , \;\; \tilde{s}^{A\star} \in]0,1[$$

Eigenschaften der Funktionen $g_A(x,y)$ und $g_B(x,y)$

Die das Bimatrix Spiel bestimmenden Funktionen $g_A(x,y)$ (farbige Fläche) und $g_B(x,y)$ (graue Fläche) sind in den unteren Abbildungen veranschaulicht. Beide Teilgruppenspiele gehören der Klasse der Koordinationsspiele an.







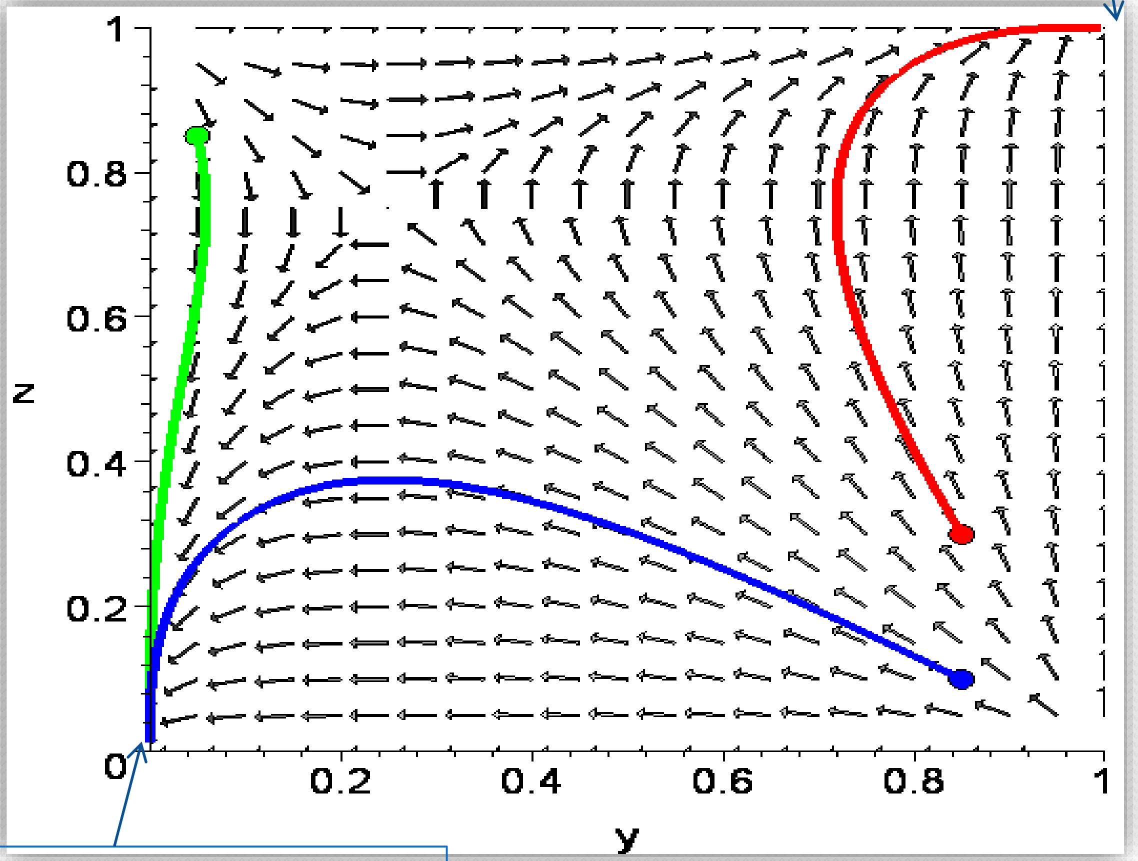
Beispiel 1:

Kampf der Geschlechter als evolutionäres Spiel

	Kino	Disko
Kino	(1 , 3)	(0 , 0)
Disko	(0 , 0)	(3 , 1)

Das „Kampf der Geschlechter"- Spiel gehört der Klasse der „Sattelpunktsspiele" (Saddle Class) an. Das Phasenportrait des Spiels besitzt das folgende Aussehen:

(1,1) entspricht (Kino,Kino)



(0,0) entspricht (Disko,Disko)

Die rechte Abbildung zeigt das zeitliche Verhalten von drei nummerischen Lösungen mit unterschiedlichen Anfangsbedingungen. Die evolutionäre Erweiterung des Spiels besitzt zwei evolutionär stabile Strategien ((0,0) und (1,1)). Die blaue und grüne Populationsentwicklung enden bei (0,0) während die Anfangsbedingung der roten Population bei (1,1) endet.

Beispiel 2:

Klasse der Zentrumsspiele (Center Class)

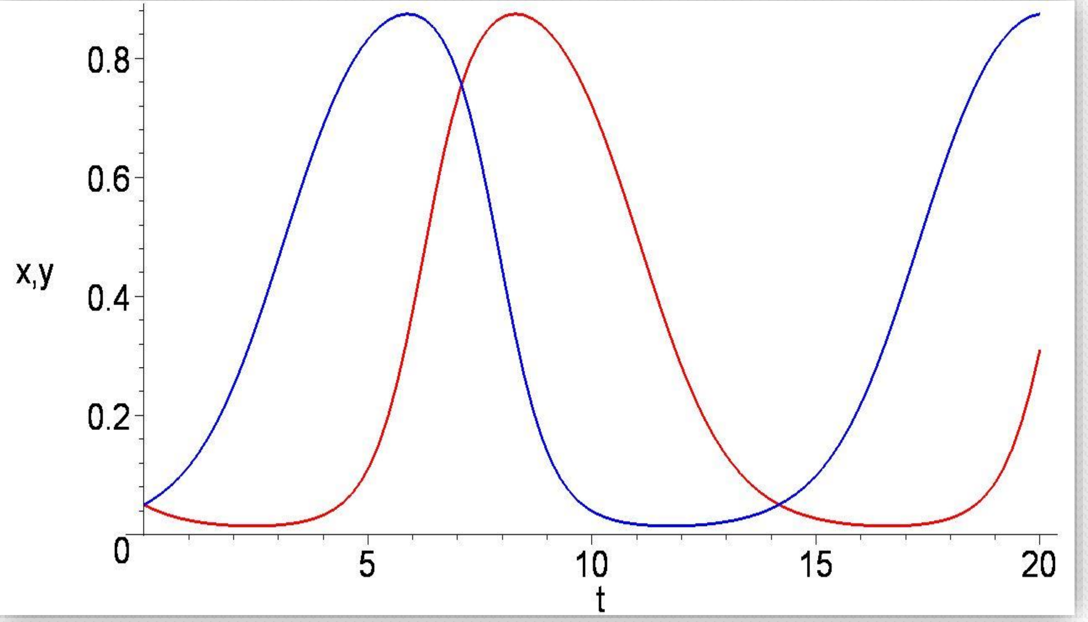
	Strat. 1	Strat. 2
Strat. 1	(2 , -2)	(0 , 0)
Strat. 2	(0 , 0)	(1 , -1)

Das gekoppelte System von Differentialgleichung für dieses Beispiel lautet:

$$\frac{dx(t)}{dt} = x(t)\cdot\left(3\cdot y(t) - 3\cdot x(t)\cdot y(t) + x(t) - 1\right)$$
$$\frac{dy(t)}{dt} = y(t)\cdot\left(-3\cdot x(t) + 3\cdot x(t)\cdot y(t) - y(t) + 1\right)$$

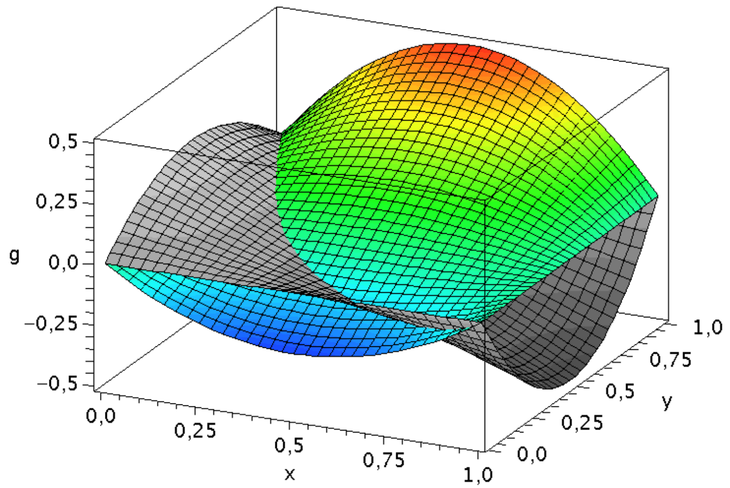
Die Lösung der Gleichung erfolgt wiederum durch Integration bzw. kann mithilfe des Computers nummerisch berechnet werden. Es wurde der Anfangswert zu (x(0),y(0)) =(0.05 , 0.05) gewählt.

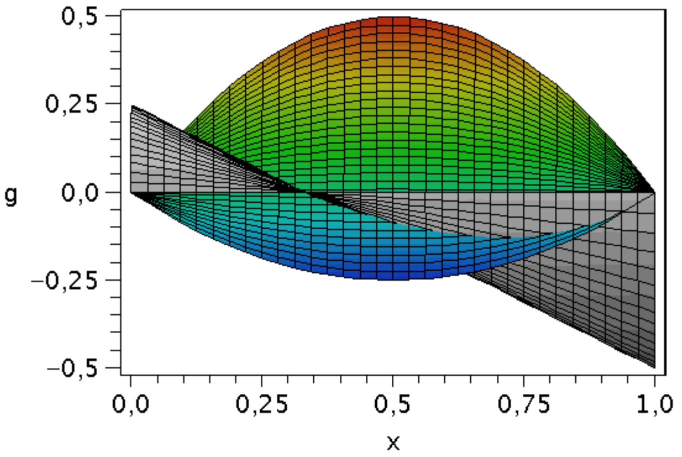
Die rechte Abbildung zeigt eine nummerische Lösung der obigen Gleichung, wobei der Anfangswert zu (x(0),y(0)) =(0.05 , 0.05) gewählt wurde. Im Gegensatz zu allen anderen möglichen Klassen von Bimatrixspielen, treten bei der Klasse der Zentrumsspiele periodische Verläufe der Populationsanteile x(t) und y(t) auf - die Populationsanteile enden nicht in einer evolutionär stabilen Strategie.

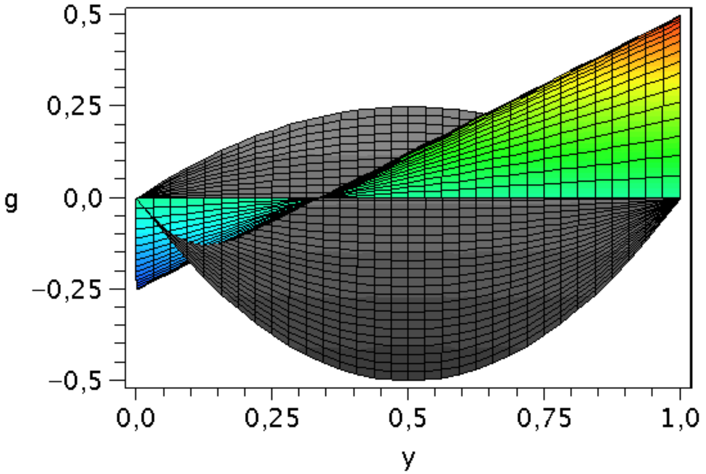


Eigenschaften der Funktionen $g_A(x,y)$ und $g_B(x,y)$

Die das Bimatrix Spiel bestimmenden Funktionen $g_A(x,y)$ (farbige Fläche) und $g_B(x,y)$ (graue Fläche) sind in den unteren Abbildungen veranschaulicht. Das Spiel der Gruppe A gehört der Klasse der Koordinationsspiele an, das der Gruppe B der Klasse der Anti-Koordinationsspiele.







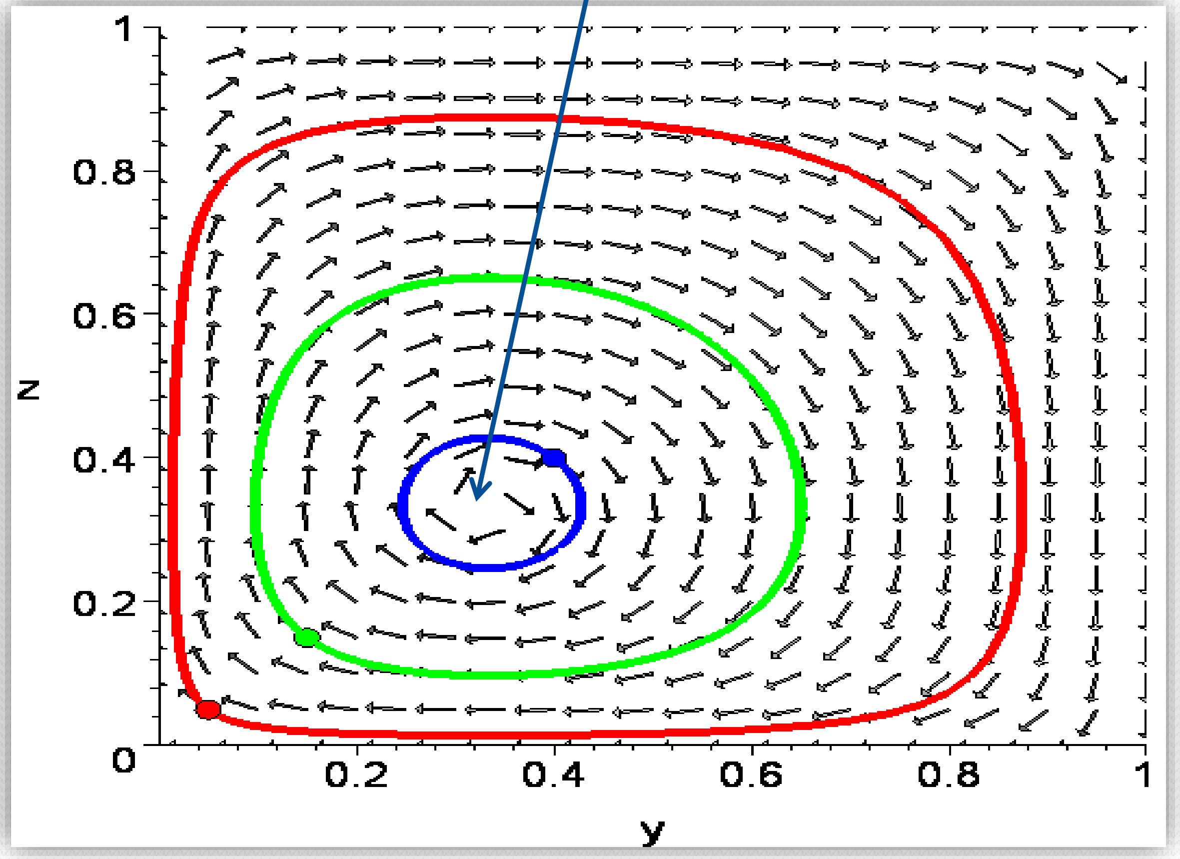
Beispiel 2:

Klasse der Zentrumsspiele (Center Class)

	Strat. 1	Strat. 2
Strat. 1	(2 , -2)	(0 , 0)
Strat. 2	(0 , 0)	(1 , -1)

Das Phasenportrait des zweiten Beispiels besitzt das folgende Aussehen:

Zentrum: Gemischtes Nash-Gleichgewicht des Spiels



Die rechte Abbildung zeigt das zeitliche Verhalten von drei nummerischen Lösungen mit unterschiedlichen Anfangsbedingungen. Dieses Bimatrixspiel besitzt keine evolutionär stabile Strategie, da die einzelnen Phasenraum-Trajektorien sich keinem Punkt annähern, sondern auf einer geschlossenen, zyklischen Bahn um das gemischte Nash-Gleichgewicht kreisen.

Beispiel 3:

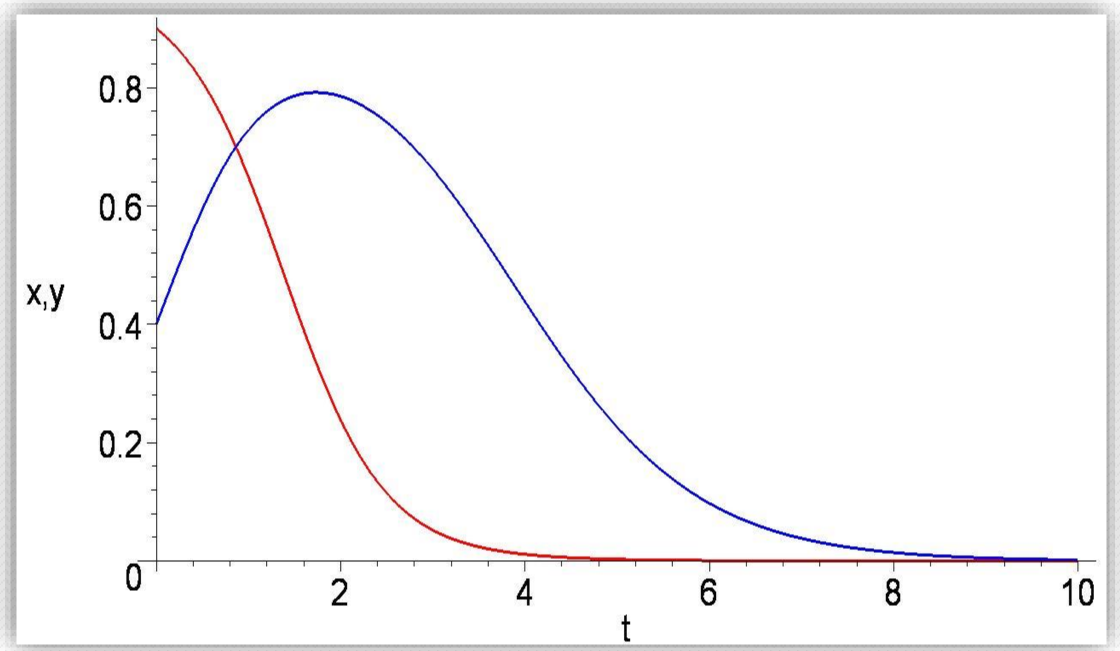
Klasse der Eckenspiele (Corner Class)

	Strat. 1	Strat. 2
Strat. 1	(-2 , 2)	(0 , 0)
Strat. 2	(0 , 0)	(1 , 1)

Das gekoppelte System von Differentialgleichung für dieses Beispiel lautet:

$$\frac{dx(t)}{dt} = x(t)\cdot\left(-y(t) + x(t)\cdot y(t) + x(t) - 1\right)$$
$$\frac{dy(t)}{dt} = y(t)\cdot\left(3\cdot x(t) - 3\cdot x(t)\cdot y(t) + y(t) - 1\right)$$

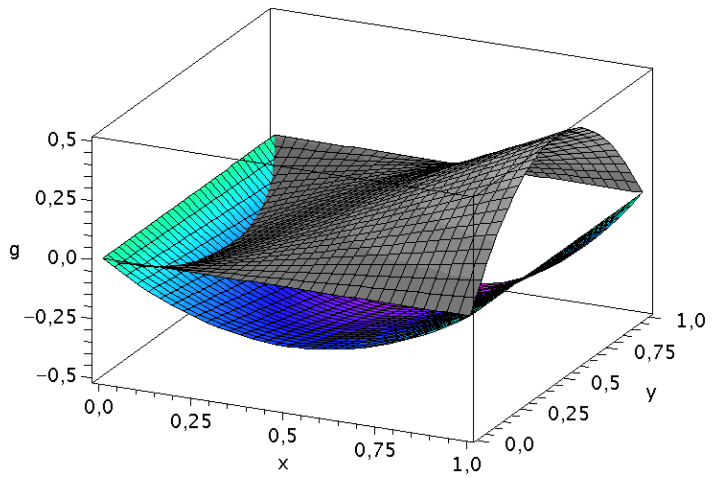
Die Lösung der Gleichung erfolgt wiederum durch Integration bzw. kann mithilfe des Computers nummerisch berechnet werden. Es wurde der Anfangswert zu
(x(0),y(0)) =(0.9 , 0.4) gewählt.

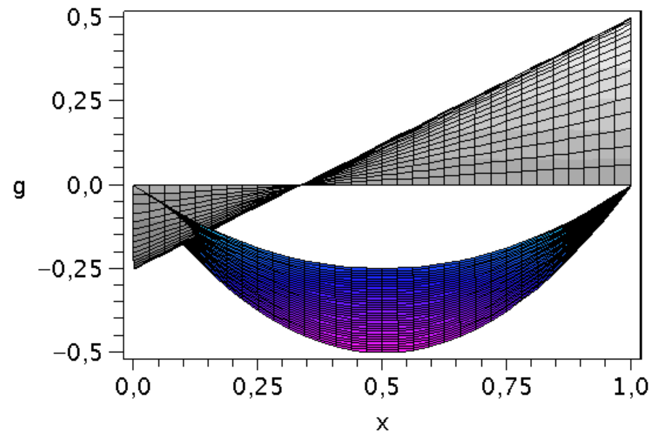


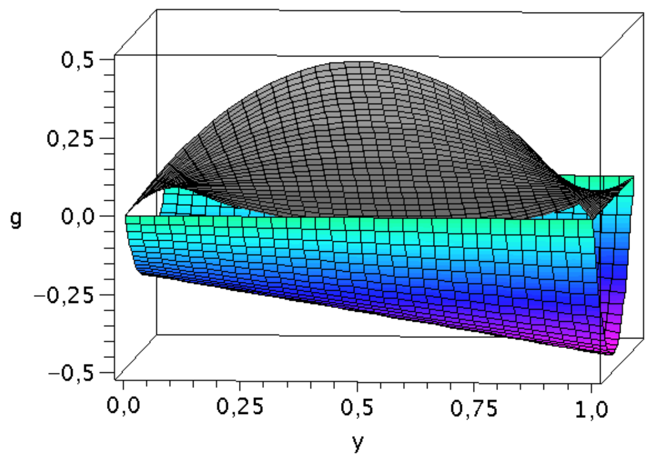
Die rechte Abbildung zeigt eine nummerische Lösung der obigen Gleichung, wobei der Anfangswert zu (x(0),y(0)) =(0.9 , 0.4) gewählt wurde. Bei der Klasse der „Eckspiele“ gibt es eine evolutionär stabile Strategie, die unabhängig von der gewählten Anfangsbedingung stets von der Population angestrebt wird.

Eigenschaften der Funktionen $g_A(x,y)$ und $g_B(x,y)$

Die das Bimatrix Spiel bestimmenden Funktionen $g_A(x,y)$ (farbige Fläche) und $g_B(x,y)$ (graue Fläche) sind in den unteren Abbildungen veranschaulicht. Das Spiel der Gruppe A gehört der Klasse der dominanten Spiele an, das der Gruppe B der Klasse der Koordinationsspiele.







Beispiel 3:

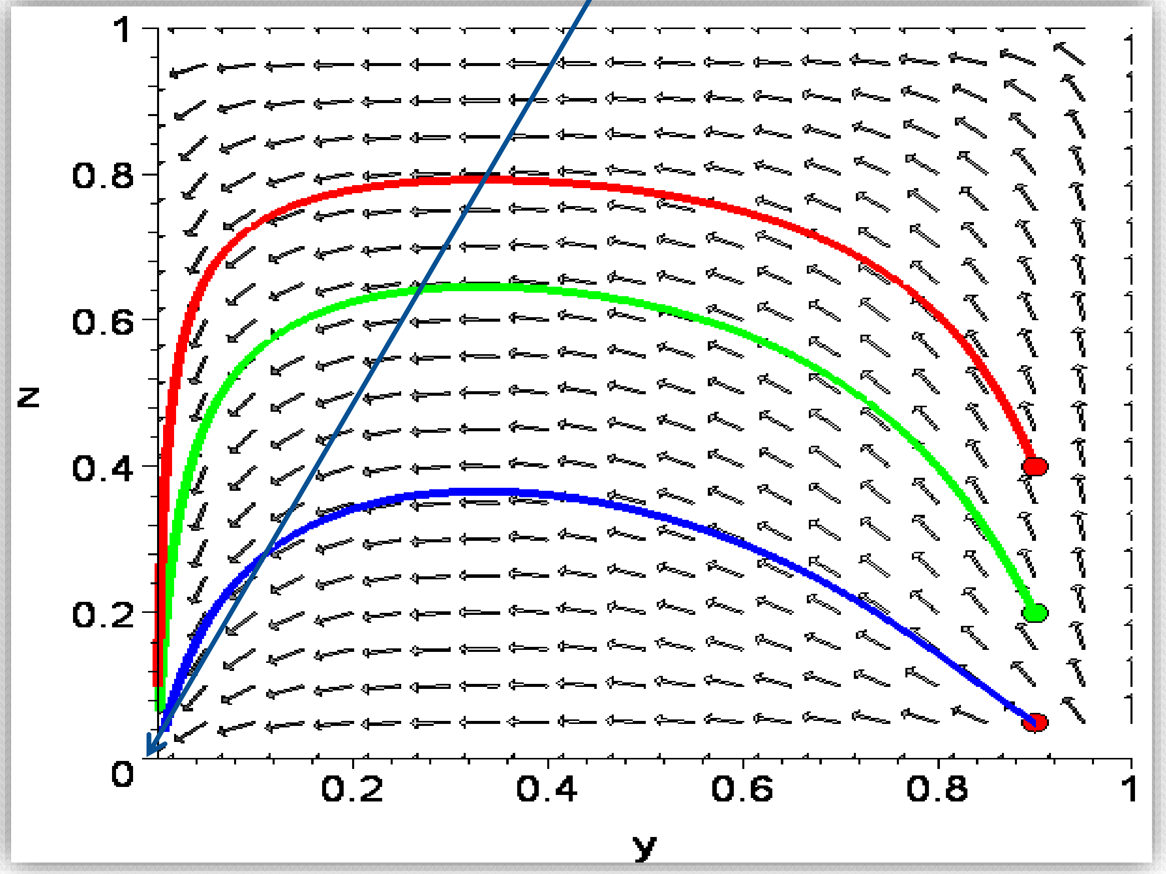
Klasse der Eckspiele (Corner Class)

	Strat. 1	Strat. 2
Strat. 1	(-2 , 2)	(0 , 0)
Strat. 2	(0 , 0)	(1 , 1)

Das Phasenportrait des dritten Beispiels besitzt das folgende Aussehen:

Die rechte Abbildung zeigt das zeitliche Verhalten von drei nummerischen Lösungen mit unterschiedlichen Anfangsbedingungen. Dieses Bimatrixspiel besitzt eine evolutionär stabile Strategie, da es nur ein gemeinsames symmetrisches Nash-Gleichgewicht gibt ((x,y)=(0,0)). Der rote Spieler besitzt sogar bei (0,0) eine dominante Strategie.

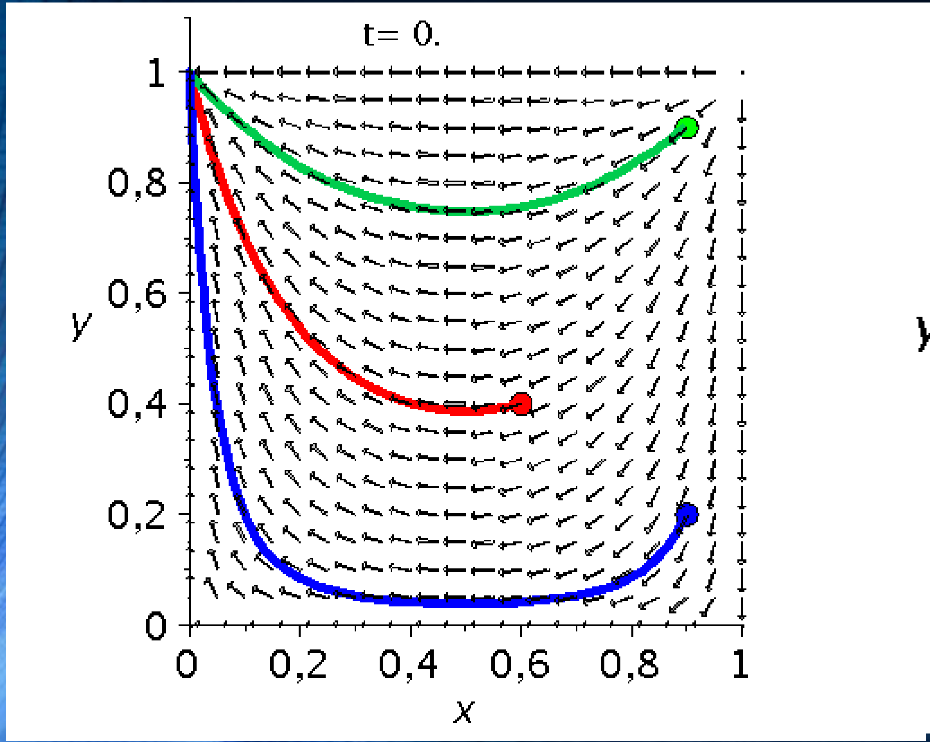
Einziges gemeinsames Nash-Gleichgewicht des Spiels



Klassifizierung von Bi-Matrix Spielen

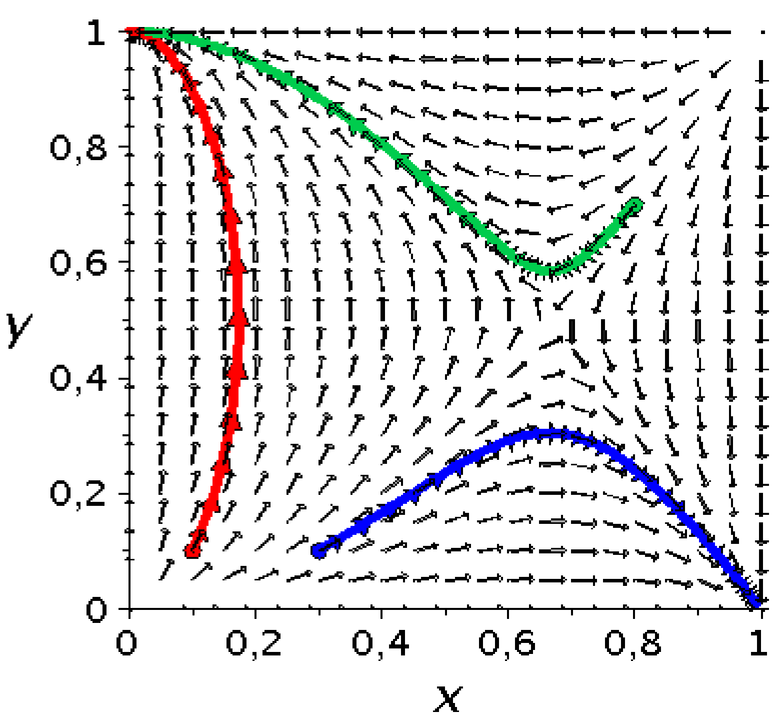
Eckspiele

Die Spielklasse der Gruppe A oder der Gruppe B ist ein *Dominantes Spiel*



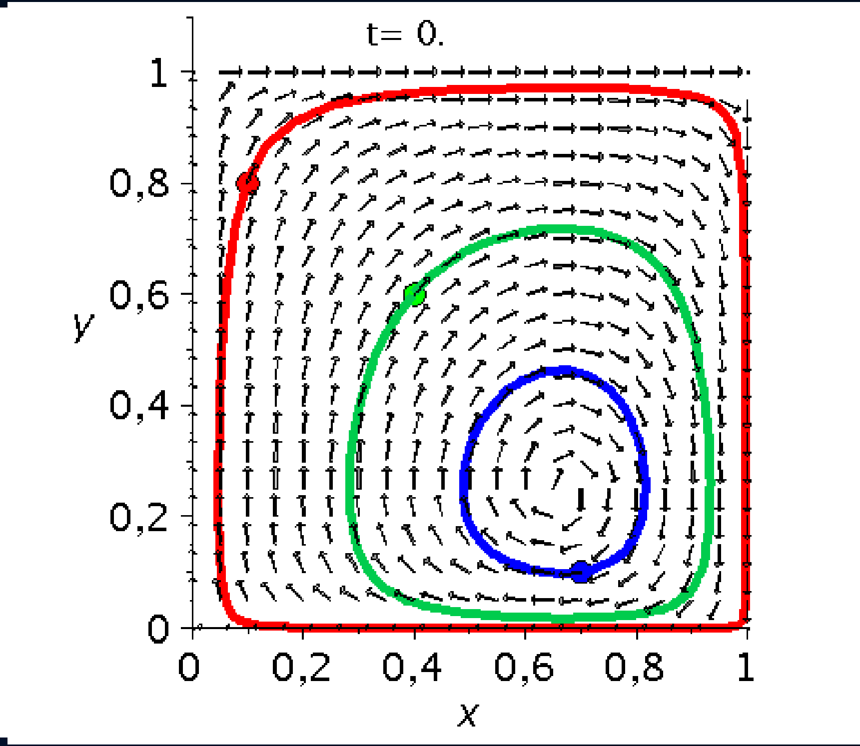
Sattelspiele

Spiel A: *Koordinationspiel*
Spiel B: *Koordinationspiel*
oder
Spiel A: *Anti-Koordinationspiel*
Spiel B: *Anti-Koordinationspiel*



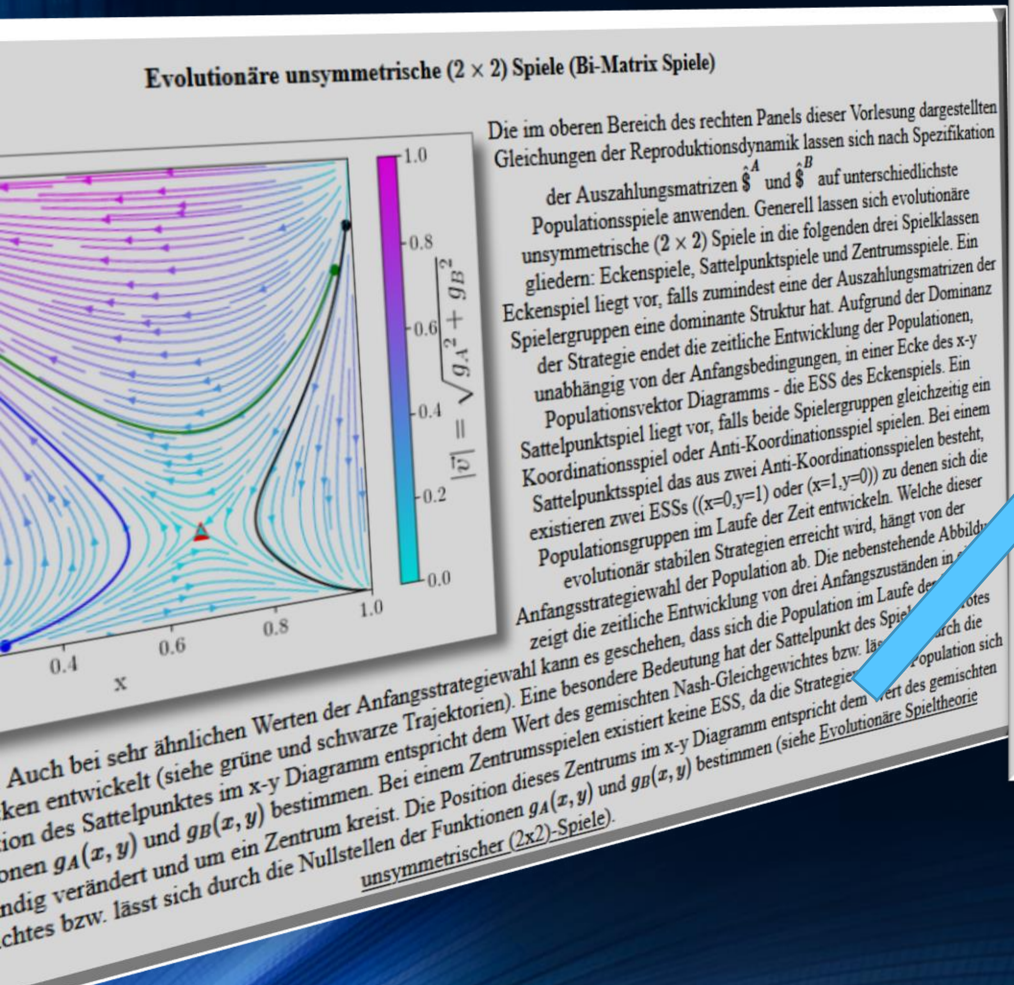
Zentrumsspiele

Spiel A: *Koordinationspiel*
Spiel B: *Anti-Koordinationspiel*
oder
Spiel A: *Anti-Koordinationspiel*
Spiel B: *Koordinationspiel*



Jupyter Notebook

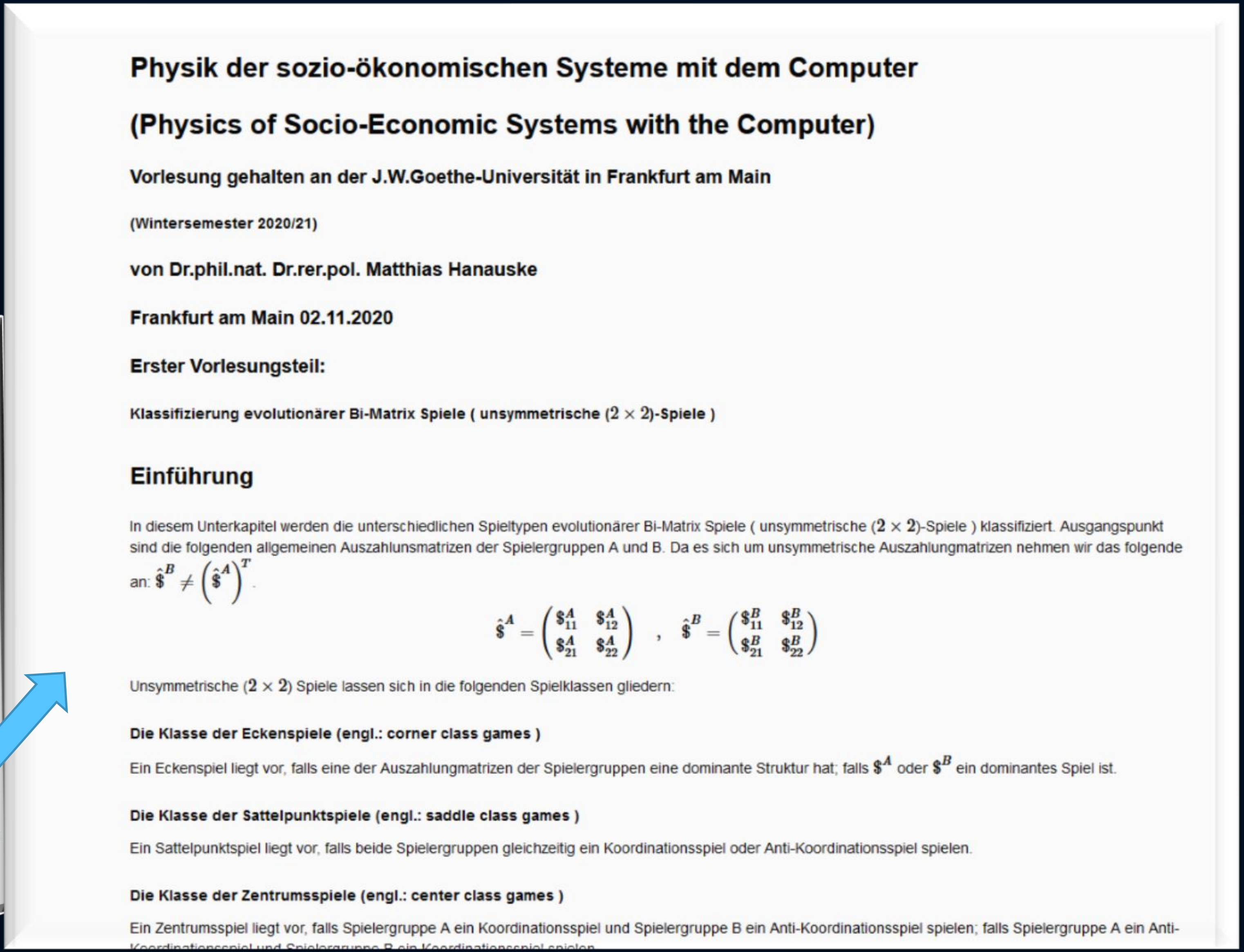
Auf der Internetseite der Vorlesung



Evolutionäre unsymmetrische (2 × 2) Spiele (Bi-Matrix Spiele)

Die im oberen Bereich des rechten Panels dieser Vorlesung dargestellten Gleichungen der Reproduktionsdynamik lassen sich nach Spezifikation der Auszahlungsmatrizen $\hat{\$}^A$ und $\hat{\$}^B$ auf unterschiedlichste Populationsspiele anwenden. Generell lassen sich evolutionäre unsymmetrische (2 × 2) Spiele in die folgenden drei Spielklassen gliedern: Eckenspiele, Sattelpunktspiele und Zentrumsspiele. Ein Eckenspiel liegt vor, falls zumindest eine der Auszahlungsmatrizen der Spielergruppen eine dominante Struktur hat. Aufgrund der Dominanz der Strategie endet die zeitliche Entwicklung der Populationen, unabhängig von der Anfangsbedingungen, in einer Ecke des x-y Populationsvektor Diagramms - die ESS des Eckenspiels. Ein Sattelpunktspiel liegt vor, falls beide Spielergruppen gleichzeitig ein Koordinationsspiel oder Anti-Koordinationsspiel spielen. Bei einem Sattelpunktsspiel das aus zwei Anti-Koordinationsspielen besteht, existieren zwei ESSs ((x=0,y=1) oder (x=1,y=0)) zu denen sich die Populationsgruppen im Laufe der Zeit entwickeln. Welche dieser evolutionär stabilen Strategien erreicht wird, hängt von der Anfangsstrategiewahl der Population ab. Die nebenstehende Abbildung zeigt die zeitliche Entwicklung von drei Anfangszuständen im Laufe der ...

... Auch bei sehr ähnlichen Werten der Anfangsstrategiewahl kann es geschehen, dass sich die Population im Laufe des ... entwickelt (siehe grüne und schwarze Trajektorien). Eine besondere Bedeutung hat der Sattelpunkt des Spiels ... des Sattelpunktes im x-y Diagramm entspricht dem Wert des gemischten Nash-Gleichgewichtes bzw. ... Population sich ... $g_A(x,y)$ und $g_B(x,y)$ bestimmen. Bei einem Zentrumsspielen existiert keine ESS, da die Strategie... ... verändert und um ein Zentrum kreist. Die Position dieses Zentrums im x-y Diagramm entspricht dem Wert des gemischten ... lässt sich durch die Nullstellen der Funktionen $g_A(x,y)$ und $g_B(x,y)$ bestimmen (siehe Evolutionäre Spieltheorie unsymmetrischer (2x2)-Spiele).



Physik der sozio-ökonomischen Systeme mit dem Computer

(Physics of Socio-Economic Systems with the Computer)

Vorlesung gehalten an der J.W.Goethe-Universität in Frankfurt am Main

(Wintersemester 2020/21)

von Dr.phil.nat. Dr.rer.pol. Matthias Hanauske

Frankfurt am Main 02.11.2020

Erster Vorlesungsteil:

Klassifizierung evolutionärer Bi-Matrix Spiele (unsymmetrische (2×2)-Spiele)

Einführung

In diesem Unterkapitel werden die unterschiedlichen Spieltypen evolutionärer Bi-Matrix Spiele (unsymmetrische (2×2)-Spiele) klassifiziert. Ausgangspunkt sind die folgenden allgemeinen Auszahlunsmatrizen der Spielergruppen A und B. Da es sich um unsymmetrische Auszahlungmatrizen nehmen wir das folgende an: $\hat{\$}^B \neq \left(\hat{\$}^A\right)^T$.

$$\hat{\$}^A = \begin{pmatrix} \$^A_{11} & \$^A_{12} \\ \$^A_{21} & \$^A_{22} \end{pmatrix} \quad , \quad \hat{\$}^B = \begin{pmatrix} \$^B_{11} & \$^B_{12} \\ \$^B_{21} & \$^B_{22} \end{pmatrix}$$

Unsymmetrische (2×2) Spiele lassen sich in die folgenden Spielklassen gliedern:

Die Klasse der Eckenspiele (engl.: corner class games)

Ein Eckenspiel liegt vor, falls eine der Auszahlungmatrizen der Spielergruppen eine dominante Struktur hat; falls $\$^A$ oder $\$^B$ ein dominantes Spiel ist.

Die Klasse der Sattelpunktspiele (engl.: saddle class games)

Ein Sattelpunktspiel liegt vor, falls beide Spielergruppen gleichzeitig ein Koordinationsspiel oder Anti-Koordinationsspiel spielen.

Die Klasse der Zentrumsspiele (engl.: center class games)

Ein Zentrumsspiel liegt vor, falls Spielergruppe A ein Koordinationsspiel und Spielergruppe B ein Anti-Koordinationsspiel spielen; falls Spielergruppe A ein Anti-

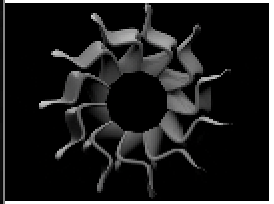
Zusätzlich befindet sich das Jupyter Notebook auch auf der OLAT Seite des Kurses

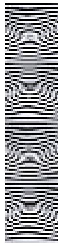
Maple Worksheet

Auf der Internetseite der Vorlesung

Download
Maple Worksheet

Weiterführende Links





- Folien der 4.Vorlesung
- View Jupyter Notebook: Evolutionäre Spieltheorie symmetrischer (2x2)-Spiele
- Download Jupyter Notebook: Evolutionäre Spieltheorie symmetrischer (2x2)-Spiele
- Download Python Programm: Evolutionäre Spieltheorie symmetrischer (2x2)-Spiele
- View Jupyter Notebook: Evolutionäre Spieltheorie unsymmetrischer (2x2)-Spiele
- Download Jupyter Notebook: Evolutionäre Spieltheorie unsymmetrischer (2x2)-Spiele
- Download Python Programm: Evolutionäre Spieltheorie unsymmetrischer (2x2)-Spiele
- Maple Worksheet: Klassifizierung von symmetrischen evolutionären (2x2)-Spielen
- Maple Worksheet: Evolutionäre Bi-Matrix Spiele

BiMatrix1.mw

>

Physik der sozio-ökonomischen Systeme mit dem Computer

Physics of Socio-Economic Systems with the Computer

Vorlesung gehalten an der J.W.Goethe-Universität in Frankfurt am Main (Wintersemester 2017/18)

von Dr.phil.nat. Dr.rer.pol. Matthias Hanauske

Frankfurt am Main 22.08.2017

Erster Vorlesungsteil:

Bi-Matrix Spiele

Einführung

Dieses Maple-Worksheet befasst sich mit Bi-Matrix Spielen und berechnet deren Lösungen in einem evolutionären, zeitabhängigen Kontext.Ausgangspunkt ist eine allgemeine unsymmetrische Auszahlunsmatrix eines (2 Personen)-(2 Strategien) Spiels.

```python
import numpy as np
import matplotlib.pyplot as plt
import matplotlib

# Definition der Funktionen g_A und g_B
def gA(x,y,a,b,c,d):
  g=((a+d-c-b)*y + (b-d))*(x-x*x)
  return g
def gB(x,y,a,b,c,d):
  g=((a+d-c-b)*x + (b-d))*(y-y*y)
  return g

# Loesen der DGL
def solve_dgl(numpoints,tend,x0,y0,Aa,Ab,Ac,Ad,Ba,Bb,Bc,Bd):
  sol=np.empty([numpoints,3])
  t=np.linspace(0,tend,numpoints)
  dt=t[1]-t[0]
  sol[0].flat[0] = t[0]
  sol[0].flat[1] = x0
  sol[0].flat[2] = y0
  i = 1
  while i < len(sol):
    sol[i].flat[0] = t[i]
    dx=gA(sol[i-1,1],sol[i-1,2],Aa,Ab,Ac,Ad)*dt
    dy=gB(sol[i-1,1],sol[i-1,2],Ba,Bb,Bc,Bd)*dt
    sol[i].flat[1] = sol[i-1,1] + dx
    sol[i].flat[2] = sol[i-1,2] + dy
    i = i + 1
  return sol

# Groessenfestlegung der Labels usw. im Bild
params = {
    'text.usetex'       : True,
    'axes.titlesize' : 22,
    'axes.labelsize' : 20,
    'xtick.labelsize' : 20 ,
    'ytick.labelsize' : 20
}
matplotlib.rcParams.update(params)
```

Weiterführende Links

- Folien der 4.Vorlesung
- View Jupyter Notebook: Evolutionäre Spieltheorie symmetrischer (2x2)-Spiele
- Download Jupyter Notebook: Evolutionäre Spieltheorie symmetrischer (2x2)-Spiele
- Download Python Programm: Evolutionäre Spieltheorie symmetrischer (2x2)-Spiele
- View Jupyter Notebook: Evolutionäre Spieltheorie unsymmetrischer (2x2)-Spiele
- Download Jupyter Notebook: Evolutionäre Spieltheorie unsymmetrischer (2x2)-Spiele
- Download Python Programm: Evolutionäre Spieltheorie unsymmetrischer (2x2)-Spiele
- Maple Worksheet: Klassifizierung von symmetrischen evolutionären (2x2)-Spielen
- Maple Worksheet: Evolutionäre Bi-Matrix Spiele

Python Programm

Auf der Internetseite der Vorlesung

Bi-Matrix Spiele mit Python V1

```python
import numpy as np
import matplotlib.pyplot as plt
import matplotlib

# Definition der Funktionen g_A und g_B
def gA(x,y,a,b,c,d):
  g=((a+d-c-b)*y + (b-d))*(x-x*x)
  return g
def gB(x,y,a,b,c,d):
  g=((a+d-c-b)*x + (b-d))*(y-y*y)
  return g

# Loesen der DGL
def solve_dgl(numpoints,tend,x0,y0,Aa,Ab,Ac,Ad,Ba,Bb,Bc,Bd):
  sol=np.empty([numpoints,3])
  t=np.linspace(0,tend,numpoints)
  dt=t[1]-t[0]
  sol[0].flat[0] = t[0]
  sol[0].flat[1] = x0
  sol[0].flat[2] = y0
  i = 1
  while i < len(sol):
    sol[i].flat[0] = t[i]
    dx=gA(sol[i-1,1],sol[i-1,2],Aa,Ab,Ac,Ad)*dt
    dy=gB(sol[i-1,1],sol[i-1,2],Ba,Bb,Bc,Bd)*dt
    sol[i].flat[1] = sol[i-1,1] + dx
    sol[i].flat[2] = sol[i-1,2] + dy
    i = i + 1
  return sol

# Groessenfestlegung der Labels usw. im Bild
params = {
    'text.usetex'       : True,
    'axes.titlesize' : 22,
    'axes.labelsize' : 20,
    'xtick.labelsize' : 20 ,
    'ytick.labelsize' : 20
}
matplotlib.rcParams.update(params)
```

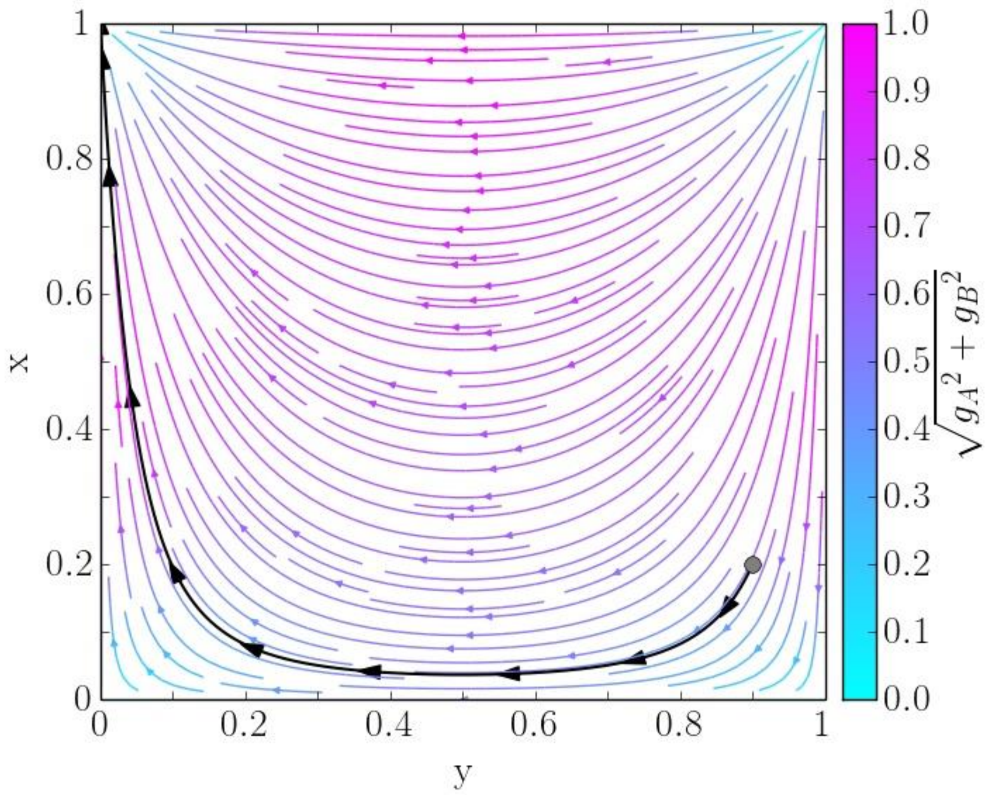
```python
#Festlegung der Auszahlungsmatrix des unsymmetrischen (2x2)-Spiels, Anfangspopulation, Endzeit
Aa=10
Ab=4
Ac=9
Ad=5
Ba=10
Bb=4
Bc=7
Bd=5
tend=12
x0=0.6
y0=0.1

# Weitere Festlegungen
numpoints=1000

# Loesung der DGL fuer eine Anfangspopulation (x0,y0)
Loes=solve_dgl(numpoints,tend,x0,y0,Aa,Ab,Ac,Ad,Ba,Bb,Bc,Bd)

# Plotten des Bildes
plt.plot(Loes[:,1],Loes[:,2],c="black", linewidth=1.5, linestyle='-')
plt.plot(Loes[0,1],Loes[0,2], marker='o', color='grey', markersize=8)

# Achsenbeschriftungen usw.
plt.xlim(0,1)
plt.ylim(0,1)
plt.ylabel(r"$\rm y$")
plt.xlabel(r"$\rm x$")

#Ausgabe des Wertes des Populationsvektors zur Endzeit im Terminal
print "Zur Zeit t=", Loes[numpoints-1,0]
print "Der Wert von x ist: x=", Loes[numpoints-1,1]
print "Der Wert von y ist: y=", Loes[numpoints-1,2]


#Speichern der Bilder als .jpg- und .pdf-Datei
saveFig="./bimatrix.jpg"
plt.savefig(saveFig, dpi=100,bbox_inches="tight",pad_inches=0.05,format="jpg")
plt.show()
```

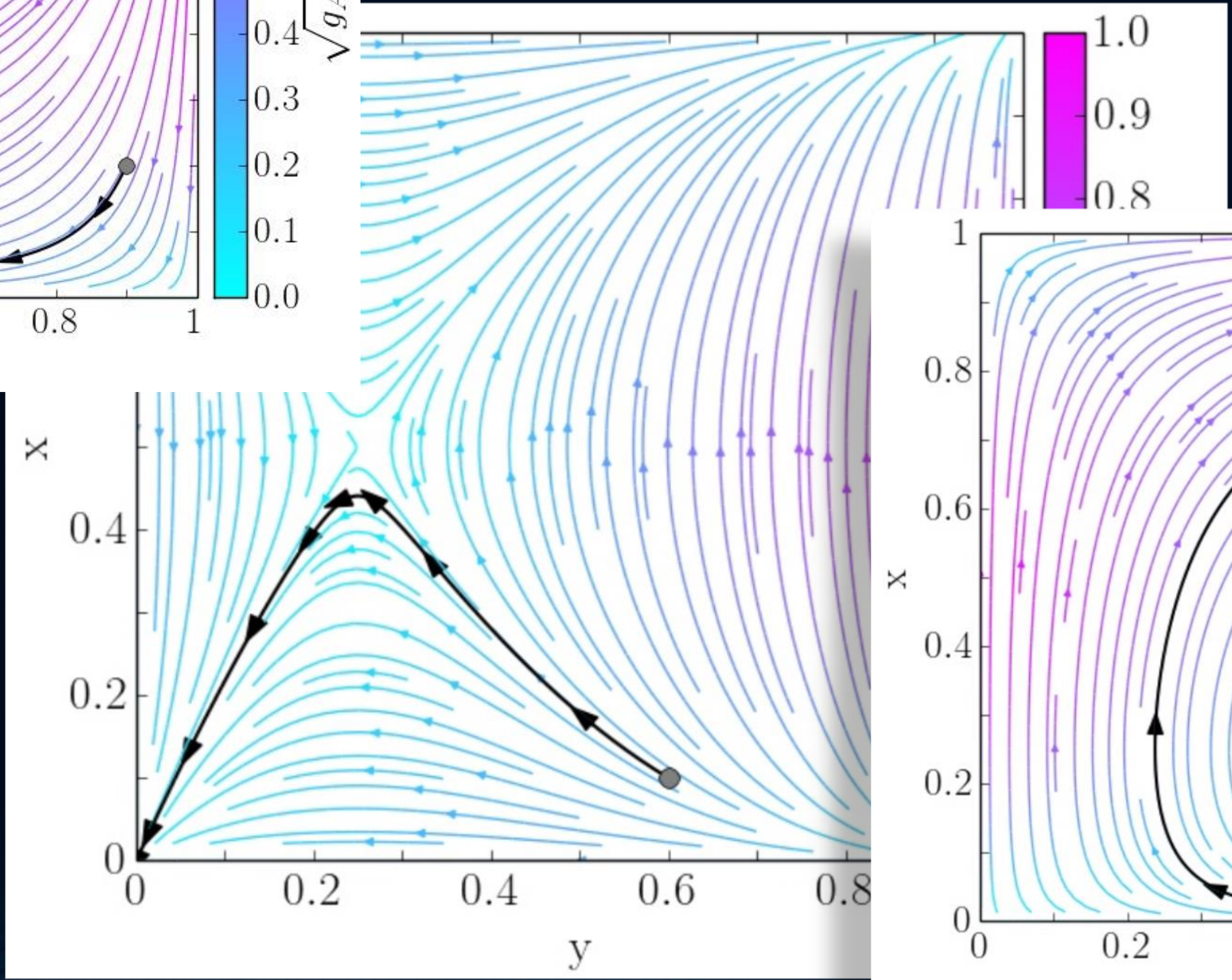
Bi-Matrix Spiele mit Python V2 (Feldliniendiagramm)

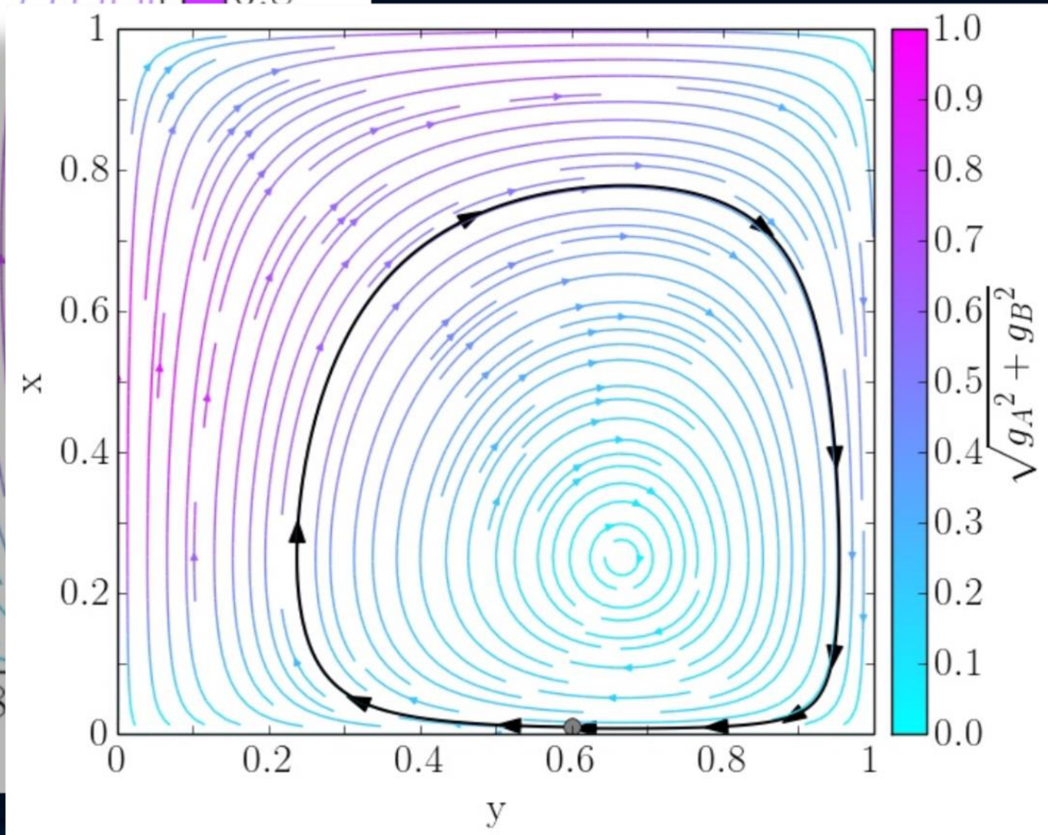
```
import numpy as np
import matplotlib.pyplot as plt
import matplotlib

# Definition der Funktionen g_A und g_B
def gA(x,y,a,b,c,d):
  g=((a+d-c-b)*y + (b-d))*(x-x*x)
  return g
def gB(x,y,a,b,c,d):
  g=((a+d-c-b)*x + (b-d))*(y-y*y)
  return g

# Groessenfestlegung der Labels usw. im Bild
params = {
    'text.usetex'       : True,
    'axes.titlesize' : 22,
    'axes.labelsize' : 20,
    'xtick.labelsize' : 20 ,
    'ytick.labelsize' : 20
}
matplotlib.rcParams.update(params)

#Festlegung der Auszahlungsmatrix des
Aa=10
Ab=4
Ac=9
Ad=5
Ba=10
Bb=4
Bc=7
Bd=5
tend=12
x0=0.6
y0=0.1
```

```
# Fuer die Darstellung des Feldliniendiagramms streamplot
SY,SX=np.mgrid[0:1:100j,0:1:100j]
SgA=gA(SX,SY,Aa,Ab,Ac,Ad)
SgB=gB(SX,SY,Ba,Bb,Bc,Bd)
# Die Farbe wird die Geschwindigkeit der Aenderung des Populationsvektors anzeigen
speed=np.sqrt(SgA*SgA + SgB*SgB)
colorspeed = speed/speed.max()

# Plotten des Bildes
strm = plt.streamplot(SX,SY,SgA,SgB,density=[2, 2], linewidth=1,color=colorspeed, cmap=plt.cm.cool)
# Erzeugung der nebenstehenden Farblegende colorbar
cbar=plt.colorbar(strm.lines,pad=0.02)
cbar.set_label(r'$\sqrt{{g_A}^2 + {g_B}^2}$',size=20)

# Achsenbeschriftungen usw.
plt.xlim(0,1)
plt.ylim(0,1)
plt.ylabel(r"$\rm y$")
plt.xlabel(r"$\rm x$")

#Speichern der Bilder als .jpg- und .pdf-Datei
saveFig="./bimatrix.jpg"
plt.savefig(saveFig, dpi=100,bbox_inches="tight",pad_inches=0.05,format="jpg")
plt.show()
```

Bi-Matrix Spiele mit Python
Version bimatrix1.py







Aufgaben auf Lon-Cappa

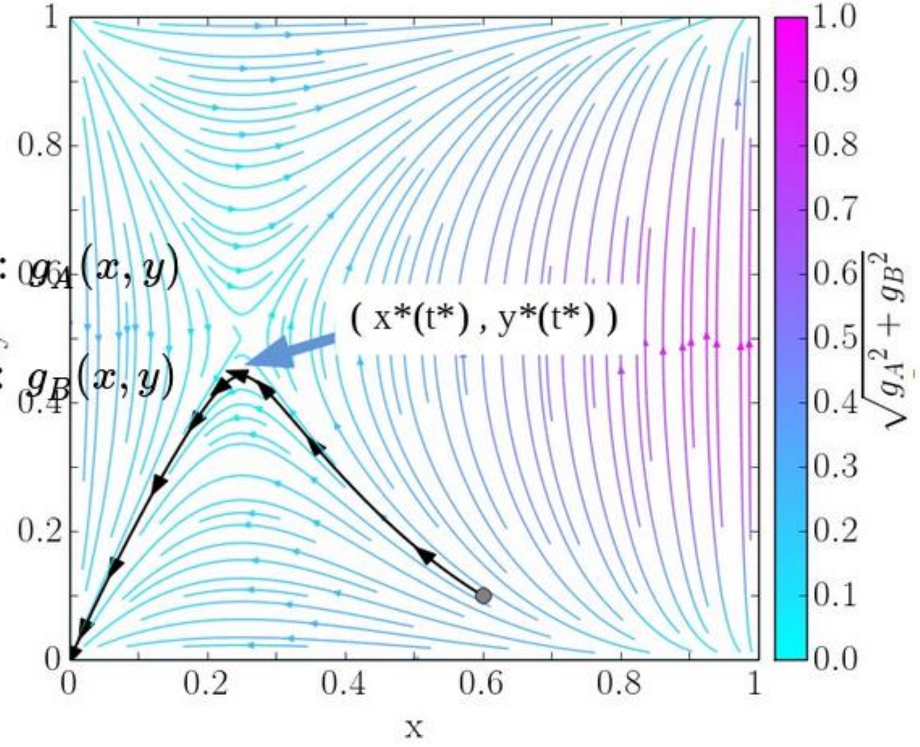
Das zeitliche Verhalten der Komponenten der Populationsvektoren (Gruppe A: $x(t) := x_1^A(t)$ und Gruppe B: $y(t) := x_1^B(t)$) wird in der Reproduktionsdynamik mittels des folgenden Systems von Differentialgleichungen beschrieben:

$$\frac{dx(t)}{dt} = \left[\left(\$_{11}^A + \$_{22}^A - \$_{12}^A - \$_{21}^A\right) y(t) + \left(\$_{12}^A - \$_{22}^A\right)\right]\left(x(t) - (x(t))^2\right) =: g_A(x,y)$$

$$\frac{dy(t)}{dt} = \left[\left(\$_{11}^B + \$_{22}^B - \$_{12}^B - \$_{21}^B\right) x(t) + \left(\$_{12}^B - \$_{22}^B\right)\right]\left(y(t) - (y(t))^2\right) =: g_B(x,y)$$

Das durch die folgende Auszahlungstabelle definierte Bimatrix Spiel gehört der Klasse der Sattelpunktsspiele an.

A/B	s_1^B	s_2^B
s_1^A	(10 , 10)	(4 , 7)
s_2^A	(9 , 4)	(5 , 5)



t^* = ____ , y^* = ____ (bitte geben Sie den numerischen Wert auf mindestens 4 Nachkommastellen an; z.B. 0.4135)

Antwort einreichen Versuche 0/5

Der Populationsvektor zur Zeit t=0 sei (x(0)=0.6 , y(0)=0.0565). Der Anteil der Spieler in der Gruppe B die die Strategie s_1^B spielen nimmt zunächst zu, erreicht dann ein Maximum und nimmt dannach wieder ab (siehe nebenstehende Abbildung). Berechnen Sie den Zeitpunkt t^* an dem der maximale Wert y^* erreicht wird.