

Mathematische Modellierung SS20

Lösungen der Beispiele auf Blatt 8

Beispiele und Skriptum

Inhaltsverzeichnis

• Beispiel 1: Strömungsmechanik	2
○ Aufgabe	2
○ Lösung	2
* Teil (a) Quadratisches Gebiet, Beschränkter Reibung	2
* Teil (b) Herleitung	3
* Teil (c) Allgemein Quadratisches Gebiet	4
* Teil (d) Allgemeines Gebiet, Numerische Rechnungen	13
* Teil (e) Allgemeines Gebiet, Variationelle Rechnungen	21
• Beispiel 2: Fischfangflotte	25
○ Aufgabe	25
○ Lösung	25
* Teil (a) Zweistufige Steuerung	25
* Teil (b) Allgemeine Steuerung, Numerische Rechnungen	33
* Teil (c) Allgemeine Steuerung, Variationelle Rechnungen	37
• Beispiel 3: Kooperation – Brunnenteilen	46
○ Aufgabe	46
○ Lösung	46
* Teil (a) Strafe	46
* Teil (b) Selbstständiger Verlauf	47
* Teil (c) Strategien für Tagefolgen	47
* Teil (d) Simulation	49
• Beispiel 4: Nash Verhandlung - Ressourcenteilen	53
○ Aufgabe	53
○ Lösung	53
• Beispiel 5: Gleichgewicht - Wahlsystem	54
○ Aufgabe	54
○ Lösung	55
* Teil (a) Implementierung	55
* Teil (b) Simplex Gleichseitig	58
* Teil (c) Satz von Arrow	58
* Teil (d) Strategischen Schritt	60
* Teil (d) Stabilen Fixpunkt	61
* Teil (e) Voraussagbar	62
• Beispiel 6: Gleichgewicht - Prüfungssystem	62
○ Aufgabe	62
○ Lösung	63

- **Beispiel 1: Strömungsmechanik**

- **Aufgabe**

Der Durchfluss in einem Rohr soll bezüglich der Geometrie eines Querschnitts Ω mit einem festgelegten Flächeninhalt maximiert werden.

- (a) Das Profil des Rohrs sei rechteckig. Die Seiten a und b eines Querschnitts erfüllen $0 \leq a, b \leq 2$ und $ab = 1$. Die zwei senkrechten Seiten werden so bearbeitet, dass es keine Reibung für die Flüssigkeit an diesen Seiten gibt. Es gibt die üblichen Reibungseffekte an den waagerechten Seiten. Bestimmen Sie die Durchfluss maximierende Geometrie der Leitung unter diesen Bedingungen.
- (b) Leiten Sie die Bedingung für die Strömungsgeschwindigkeit im Inneren des Gebiets E von dem Navier-Stokes Gleichungssystem her.
- (c) Unter der Bedingung, dass es Reibung für die Flüssigkeit an allen vier Seiten des rechteckigen Rohrs gibt, zeigen Sie, dass der Durchfluss bei $a = b$ maximiert wird.
- (d) Sei die Geometrie des Rohrprofils beliebig, außer der Durchschnittsflächeninhalt bei 1 bleiben muss. Weiters gibt es Reibungseffekte entlang des ganzen abgeschlossenen Rands. Zeigen Sie durch Rechnungen mit einer Diskretisierung des Problems, dass der Durchfluss bei einem kreisförmigen Profil maximiert wird.
- (e) Sei die Geometrie des Rohrprofils beliebig, außer der Durchschnittsflächeninhalt bei 1 bleiben muss. Weiters gibt es Reibungseffekte entlang des ganzen abgeschlossenen Rands. Zeigen Sie theoretisch durch variationelle Rechnungen, dass der Durchfluss bei einem kreisförmigen Profil maximiert wird.

- **Lösung**

- * **Teil (a) Quadratisches Gebiet, Beschränkter Reibung**

Ein Querschnitt E der rechteckigen Leitung sei durch $0 \leq x \leq b$, $0 \leq y \leq a$ definiert. Seien $\partial E_x = \partial E \cap \{x = 0, b\}$ und $\partial E_y = \partial E \cap \{y = 0, a\}$. Sei L die Länge der Leitung. Der Druck am Flussaufwärtsende der Leitung ist p_2 und der Druck am Flussabwärtsende ist p_1 . Setze $\Delta p = p_2 - p_1$. Sei $w(x, y)$ die Geschwindigkeit in E . Wegen der Reibungsbedingungen ist w_y die einzige nicht triviale partielle Ableitung der Geschwindigkeit. Die im Skriptum hergeleitete PDG für w ist

$$-\Delta p = L \nabla \cdot (\mu \nabla w) = L \mu w_{yy} \quad \text{in } E, \quad w = 0 \quad \text{am } \partial E_y, \quad w_x = 0 \quad \text{am } \partial E_x$$

wobei die Randbedingung am ∂E_x sowieso für eine Funktion $w = w(y)$ erfüllt wird. Die Lösung dieses Randwertproblems ist

$$w(y) = \frac{y(a-y)\Delta p}{2L\mu}$$

Der Fluss durch die Leitung ist

$$Q = \int_E w dx dy = b \int_0^a w(y) dy = \frac{b \Delta p}{2L\mu} \frac{a^3}{6} \stackrel{ab=1}{=} \frac{\Delta p}{2L\mu} \frac{a^2}{6}.$$

Wegen der Einschränkung $a \in [0, 2]$ wird der Fluss bei $a = 2$ und (wegen der Einschränkung $ab = 1$) $b = 1/2$ maximiert.

* Teil (b) Herleitung

Die Bedingung im Inneren der Leitung

$$-\Delta p = L \nabla \cdot (\mu \nabla w) = L \mu w_{yy} \quad \text{in } E$$

wird vom Navier-Stokes Gleichungssystem (von oben nach unten) folgendermaßen hergeleitet. Details und Definitionen der Parameter befinden sich [hier](#). Das System von partiellen Differentialgleichungen besteht aus Massenerhaltung,

$$\partial_t \rho + \nabla \cdot (\rho \mathbf{w}) = 0$$

Impulserhaltung (vgl. Seite 47 im Skriptum mit Viskosität μ)

$$\partial_t (\rho \mathbf{w}) + \nabla \cdot (\rho \mathbf{w} \mathbf{w}^\top) = -\nabla p + \nabla \cdot S + \mathbf{f}$$

wobei $S = \mu [\nabla \mathbf{w} + \nabla \mathbf{w}^\top] + \lambda (\nabla \cdot \mathbf{w}) I$

und Energieerhaltung

$$\partial_t [\rho e + \frac{1}{2} \rho \mathbf{w}^\top \mathbf{w}] + \nabla \cdot [\rho e \mathbf{w} + \frac{1}{2} \rho \mathbf{w}^\top \mathbf{w} + p] = \nabla \cdot (S \cdot \mathbf{w}) + \nabla \cdot (\kappa \nabla T) + \rho \mathbf{w} \cdot \mathbf{f}$$

wobei $p = p(\rho, e)$ und $e = e(T)$

und die letzten zwei algebraischen (thermodynamischen) Bedingungen von der Flüssigkeit abhängen. Diese Form des Gleichungssystems gilt für *kompressible Fluide*, aber dies wird vereinfacht. Die vollständige Liste der zutreffenden Annahmen für die Aufgabe ist:

- Die Flüssigkeit ist inkompressibel, d.h. $\rho = \text{Konstante}$.
- Es gibt keine äußere Kräfte, d.h. $\mathbf{f} = 0$.
- Die Geschwindigkeit hat nur eine z -Komponente, d.h. $\mathbf{w} = (u, v, w)$, $u = v = 0$.
- Die Strömung ist im Fließgleichgewicht und insbesondere gelten $w_t = 0$, $w_z = 0$.
- Weiters ist die Druckverteilung linear mit $p = (1 - z/L)p_2 + (z/L)p_1$.
- Die Parameter μ , λ und κ sind fixierte Konstante.

Mit $\partial_t \rho = 0$ und $\nabla \rho = 0$ folgt aus Massenerhaltung

$$0 = \underbrace{\partial_t \rho}_{=0} + \underbrace{\nabla \rho \cdot \mathbf{w}}_{=0} + \rho \nabla \cdot \mathbf{w} = \rho \nabla \cdot \mathbf{w}$$

die Bedingung der Inkompressibilität,

$$\nabla \cdot \mathbf{w} = 0.$$

Weiters mit

$$\nabla \cdot (\mathbf{w} \mathbf{w}^\top) = \nabla \cdot \begin{bmatrix} u\mathbf{w} \\ v\mathbf{w} \\ w\mathbf{w} \end{bmatrix} = \begin{bmatrix} u(\nabla \cdot \mathbf{w}) + \nabla u \cdot \mathbf{w} \\ v(\nabla \cdot \mathbf{w}) + \nabla v \cdot \mathbf{w} \\ w(\nabla \cdot \mathbf{w}) + \nabla w \cdot \mathbf{w} \end{bmatrix} = \mathbf{w}(\nabla \cdot \mathbf{w}) + \mathbf{w} \cdot \nabla \mathbf{w}$$

$\rho_t = 0$, $\nabla \rho = 0$ und $\nabla \cdot \mathbf{w} = 0$ wird die linke Seite der Impulserhaltung

$$\underbrace{\partial_t (\rho \mathbf{w})}_{=0} + \nabla \cdot (\rho \mathbf{w} \mathbf{w}^\top) = \underbrace{(\nabla \rho)}_{=0} \mathbf{w} \mathbf{w}^\top + \rho \mathbf{w} \underbrace{(\nabla \cdot \mathbf{w})}_{=0} + \rho \mathbf{w} \cdot \nabla \mathbf{w} = \rho \begin{bmatrix} 0 \\ 0 \\ \nabla w \cdot \mathbf{w} \end{bmatrix} = 0$$

wobei die letzte Gleichung mit $\mathbf{w} = (0, 0, w)$ und $w_z = 0$ folgt. Der Spannungstensor erfüllt

$$S = \mu[\nabla\mathbf{w} + \nabla\mathbf{w}^\top] + \lambda(\nabla \cdot \mathbf{w})I = \mu[\nabla\mathbf{w} + \nabla\mathbf{w}^\top]$$

und daher mit $\mathbf{w} = (0, 0, w)$ und $w_z = 0$,

$$\nabla \cdot S = \nabla \cdot \left\{ \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ w_x & w_y & 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 & w_x \\ 0 & 0 & w_y \\ 0 & 0 & 0 \end{bmatrix} \right\} = \begin{bmatrix} w_{xz} \\ w_{yz} \\ w_{xx} + w_{yy} \end{bmatrix} = \hat{\mathbf{e}}_z \nabla^2 w$$

Der Druck $p = (1 - z/L)p_2 + (z/L)p_1$ erfüllt

$$-\nabla p = - \begin{bmatrix} 0 \\ 0 \\ (p_1 - p_2)/L \end{bmatrix} = \hat{\mathbf{e}}_z(p_2 - p_1)/L$$

Also mit $\mathbf{f} = 0$ wird die rechte Seite der Impulserhaltung

$$-\nabla p + \nabla \cdot S + \mathbf{f} = \hat{\mathbf{e}}_z[(p_2 - p_1)/L + \mu \nabla^2 w]$$

Da die linke Seite der Impulserhaltung Null ist, ergibt sich die gezielte Poissonsche Gleichung

$$\mu \nabla^2 w = -(p_2 - p_1)/L.$$

Nachdem die Geschwindigkeit w so gegeben ist, lässt sich die (innere) Energie e (und implizit die Temperatur $e = e(T)$) aus Energieerhaltung bestimmen.

* Teil (c) Allgemein Quadratisches Gebiet

Für $E = (0, a) \times (0, b)$ ist die exakte Lösung des Randwertproblems

$$\mu \nabla^2 w = -(p_2 - p_1)/L \text{ in } E, \quad w = 0 \text{ auf } \partial E$$

gegeben durch

$$w(x, y) = \sum_{m=0}^{\infty} \sum_{n=0}^{\infty} W_{n,m} \sin\left(\frac{\pi mx}{a}\right) \sin\left(\frac{\pi ny}{b}\right)$$

wobei

$$W_{n,m} = \frac{p_2 - p_1}{L\mu} \cdot \frac{4}{\pi^4} \cdot \frac{[1 - (-1)^m][1 - (-1)^n]}{[(m/a)^2 + (n/b)^2]mn}.$$

Der Durchfluss ist gegeben durch

$$\begin{aligned} Q(a, b) &= \int_0^b \int_0^a w(x, y) dx dy = \sum_{m=0}^{\infty} \sum_{n=0}^{\infty} W_{n,m} \int_0^b \int_0^a \sin\left(\frac{\pi mx}{a}\right) \sin\left(\frac{\pi ny}{b}\right) dx dy \\ &= \sum_{m=0}^{\infty} \sum_{n=0}^{\infty} W_{n,m} \left\{ ab \frac{[1 - (-1)^m][1 - (-1)^n]}{mn\pi^2} \right\} \\ &= \frac{p_2 - p_1}{L\mu} \cdot \frac{4}{\pi^6} \cdot \sum_{m=0}^{\infty} \sum_{n=0}^{\infty} \left(\frac{[1 - (-1)^m][1 - (-1)^n]}{mn} \right)^2 \frac{ab}{(m/a)^2 + (n/b)^2} \end{aligned}$$

Unter der Einschränkung $ab = 1$ gilt

$$Q(a, 1/a) = \frac{p_2 - p_1}{L} \cdot \frac{64}{\pi^6} \cdot \sum_{m=1}^{\infty} \sum_{n=1}^{\infty} \frac{1}{[(m/a)^2 + (an)^2]m^2n^2}$$

Die Geschwindigkeit und der Durchfluss werden mit dieser Formulierung durch den folgenden Matlab Code berechnet.

```

h1 = figure(1); close(h1); h1 = figure(1);
set(h1,'Position',[10 10 1500 500]);

p2 = 2;                                % pressures with
p1 = 1;                                %      Dp = 1
Ln = p2-p1;                            % length*viscosity so that f0=1
f0 = (p2-p1)/Ln;                        % wxx + wyy = -f0

N  = 20;                                % N*N = number of summands
h  = 1/N;

smin = 0.5;                            % side lengths between
smax = 2.0;                            %      smin and smax
Ns = 25;
s1 = 1 + (smin-1)*linspace(0,1,Ns).^2;
s2 = 1 + (smax-1)*linspace(0,1,Ns).^2;
s = unique([s1,s2]);                  % concentrated near s=1
Ns = length(s);
a = s;                                % constraint is a*b=1
b = 1./s;

Q = 0;                                  % compute slow output
for n=1:N
    for m=1:N
        d = (2*m-1)^2./a.^2 + (2*n-1)^2./b.^2;
        d = d*(2*m-1)^2*(2*n-1)^2;
        Q = Q + 1./d;
    end
end
Q = 64*Q/pi^6;
Q = f0*Q;
i = find(Q == max(Q));                % max flow

subplot(1,3,1)                         % plot side length vs flow
plot(s,Q,'b',s(i),Q(i),'r*')
legend(sprintf('Q mit N=%0.0f',N),sprintf('amax=%0.1e',s(i)))
xlabel('a')
ylabel('Q')
title(sprintf('Durchfluss, Qmax=%0.2e',Q(i)))

a = s(i);                            % optimal side lengths
b = 1/a;
Nx = round(a/h);
Ny = round(b/h);
x = linspace(0,a,Nx+1);
y = linspace(0,b,Ny+1);
xx = kron(x',ones(1,Ny+1));        % grid with these side lengths
yy = kron(ones(Nx+1,1),y);

```

```

w = zeros(Nx+1,Ny+1); % compute velocity
Lw = zeros(Nx+1,Ny+1); % and its Laplacian
for n=1:N
    for m=1:N
        d = (2*m-1)^2./a.^2 + (2*n-1)^2./b.^2;
        d = d*(2*m-1)*(2*n-1);
        xc = pi*(2*m-1)/a;
        yc = pi*(2*n-1)/b;
        w = w + sin(xc*xx).*sin(yc*yy)/d;
        c = xc^2 + yc^2;
        Lw = Lw - c*sin(xc*xx).*sin(yc*yy)/d;
    end
end
w = (16/pi^4)*w;
Lw = (16/pi^4)*Lw;
w = f0*w;
Lw = -f0*Lw;

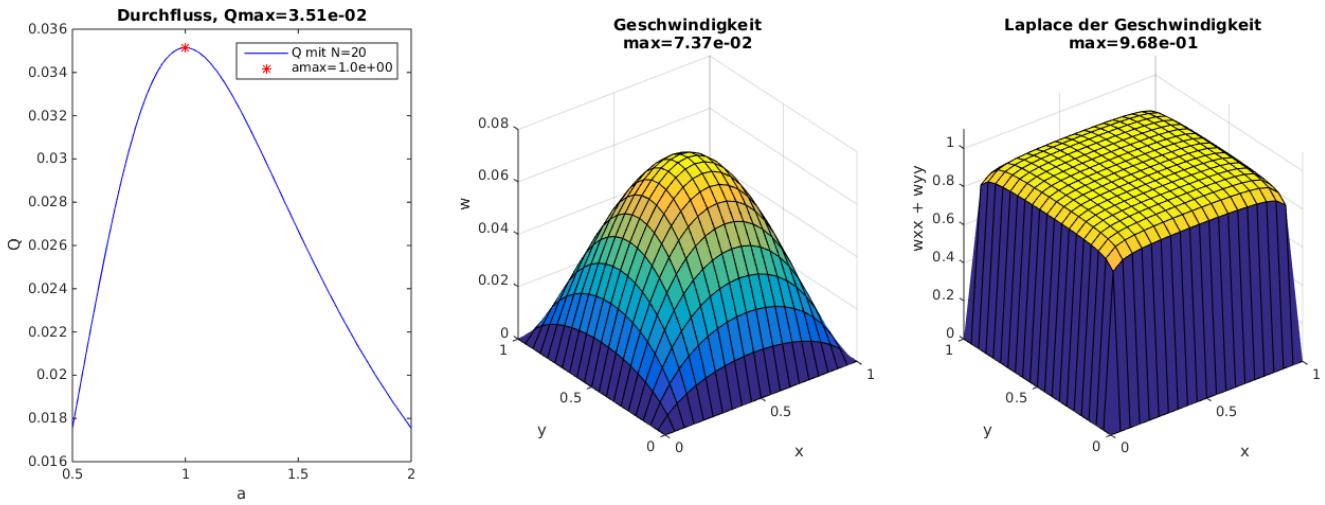
subplot(1,3,2) % graph the velocity
surf(xx,yy,w)
axis([0 a 0 b 0 0.08]);
pbaspect([a b 1])
xlabel('x')
ylabel('y')
zlabel('w')
title(sprintf('Geschwindigkeit\nmax=%0.2e',max(w(:)))) 

subplot(1,3,3) % graph the Laplacian of the velocity
surf(xx,yy,Lw)
axis([0 a 0 b 0 1.1]);
pbaspect([a b 1])
xlabel('x')
ylabel('y')
zlabel('wxx + wyy')
title(sprintf('Laplace der Geschwindigkeit\nmax=%0.2e',max(Lw(:)))) 

disp(sprintf('Qmax=%0.2e, amax=%0.2e',Q(i),s(i)))

```

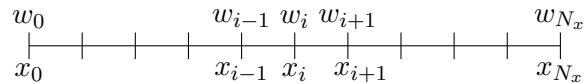
Die Ergebnisse werden folgendermaßen grafisch dargestellt. Klar wird der Durchfluss bei $a = b = 1$ maximiert.



Anstatt einer solchen spektralen Darstellung der Geschwindigkeit kann sie leichter mit finiten Differenzen berechnet werden. Beispielsweise wird die Lösung des Randwertproblems

$$\begin{cases} w_{xx} = f, & x \in (0, a) \\ w = 0, & x \in \{0, a\} \end{cases}$$

auf dem ein-dimensionalen Gitter



mit $x_{\min} = 0$, $x_{\max} = a$, $h_x = a/N_x$, (N_x = Anzahl der Zellen)

$$x_i = x_{\min} + (x_{\max} - x_{\min})ih_x, \quad i = 1, \dots, N_x$$

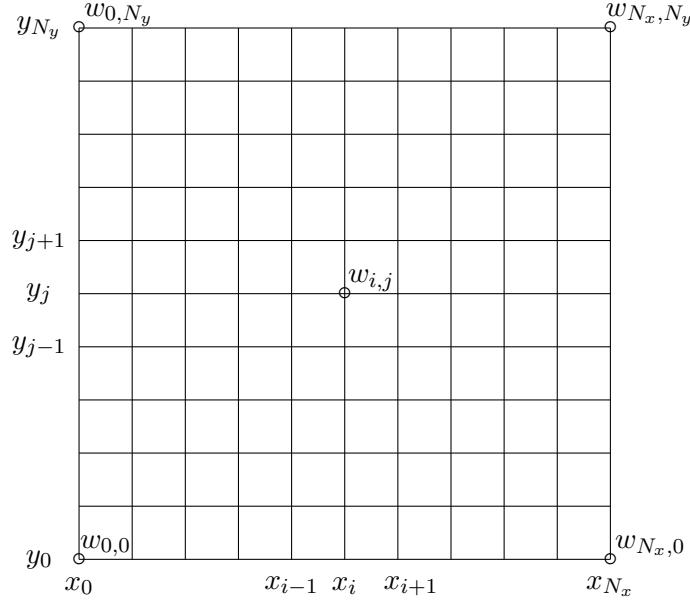
$w_i \approx w(x_i)$, $f_i \approx f(x_i)$ und den Differenzenquotienten approximiert,

$$\frac{w_{i+1} - 2w_i + w_{i-1}}{h_x^2} = f_i, \quad i = 2, \dots, N_x - 1, \quad w_0 = w_{N_x} = 0.$$

Analog wird die Lösung des Randwertproblems

$$\begin{cases} w_{xx} + w_{yy} = f, & x \in (0, a) \times (0, b) \\ w = 0, & x \in \{0, a\} \times \{0, b\} \end{cases}$$

auf dem zwei-dimensionalen Gitter



mit $x_{\min} = 0$, $x_{\max} = a$, $y_{\min} = 0$, $y_{\max} = b$, $h_x = a/N_x$, $h_y = a/N_y$,

$$\begin{aligned} x_i &= x_{\min} + (x_{\max} - x_{\min})ih_x, \quad i = 1, \dots, N_x \\ y_j &= y_{\min} + (y_{\max} - y_{\min})jh_y, \quad j = 1, \dots, N_y \end{aligned}$$

$w_{i,j} \approx w(x_i, y_j)$, $f_{i,j} \approx f(x_i, y_j)$ und den Differenzenquotienten approximiert,

$$\begin{aligned} \frac{w_{i+1,j} - 2w_{i,j} + w_{i-1,j}}{h_x^2} + \frac{w_{i,j+1} - 2w_{i,j} + w_{i,j-1}}{h_y^2} &= f_{i,j}, \\ i &= 2, \dots, N_x - 1, \quad j = 2, \dots, N_y - 1, \\ w_{0,j} &= w_{N_x,j} = 0, \quad w_{i,0} = w_{i,N_y} = 0. \end{aligned}$$

Um dieses Gleichungssystem in Matrixform umzuschreiben seien definiert,

$$\begin{aligned} \mathbf{dx} &= \frac{1}{hx} \begin{bmatrix} -1 & +1 & & 0 \\ & \ddots & \ddots & \\ 0 & & -1 & +1 \end{bmatrix} \in \mathbb{R}^{N_x \times (N_x+1)}, \quad \mathbf{iy} = \begin{bmatrix} +1 & & 0 \\ & \ddots & \\ 0 & & +1 \end{bmatrix} \in \mathbb{R}^{N_x \times (N_y+1) \times (N_y+1)} \\ \mathbf{Dx} &= \mathbf{iy} \otimes \mathbf{dx} = \begin{bmatrix} \mathbf{dx} & & 0 \\ & \ddots & \\ 0 & & \mathbf{dx} \end{bmatrix} \quad (= \text{kron}(\mathbf{iy}, \mathbf{dx})) \\ \mathbf{dy} &= \frac{1}{hy} \begin{bmatrix} -1 & +1 & & 0 \\ & \ddots & \ddots & \\ 0 & & -1 & +1 \end{bmatrix} \in \mathbb{R}^{N_y \times (N_y+1)}, \quad \mathbf{ix} = \begin{bmatrix} +1 & & 0 \\ & \ddots & \\ 0 & & +1 \end{bmatrix} \in \mathbb{R}^{N_x \times (N_x+1) \times (N_x+1)} \\ \mathbf{Dy} &= \mathbf{dy} \otimes \mathbf{ix} = \frac{1}{hy} \begin{bmatrix} -\mathbf{ix} & \mathbf{ix} & & 0 \\ & \ddots & \ddots & \\ 0 & & -\mathbf{ix} & \mathbf{ix} \end{bmatrix} \quad (= \text{kron}(\mathbf{dy}, \mathbf{ix})) \\ \mathbf{z} &= \{\mathbf{z}_{i,j}\}, \quad \mathbf{z}_{i,j} = \begin{cases} 0, & i = 0, N_x, j = 0, N_y \\ 1, & \text{sonst} \end{cases} \quad \mathbf{Z} = \text{diag}(\mathbf{z}) \quad \mathbf{I} = \text{diag}(1) \end{aligned}$$

$$L = Dx^\top Dx + Dy^\top Dy, \quad f = \{f_{i,j}\}$$

und $w = \{w_{i,j}\}$ ist gegeben durch die Lösung von

$$[ZL + (I - Z)]w = Zf$$

Die Geschwindigkeit und der Durchfluss werden mit dieser Formulierung durch den folgenden Matlab Code berechnet.

```

h1 = figure(1); close(h1); h1 = figure(1);
set(h1,'Position',[10 10 1500 500]);

p2 = 2;                                % pressures with
p1 = 1;                                %      Dp = 1
Ln = p2-p1;                            % length*viscosity so that f0=1
f0 = (p2-p1)/Ln;                        % wxx + wyy = -f0

N  = 20;                                % Nx*Ny ~= N*N
h  = 1/N;                               % 1/hx + 1/hy ~= 2/h

smin = 0.5;                            % side lengths between
smax = 2.0;                            %      smin and smax
Ns = 25;
s1 = 1 + (smin-1)*linspace(0,1,Ns).^2;
s2 = 1 + (smax-1)*linspace(0,1,Ns).^2;
s = unique([s1,s2]);                  % concentrated near s=1
Ns = length(s);

sv = [] ;                                % save for plots
Qv = [] ;

for k=2:Ns                                % compute flow for various
    a = s(k);                            %      side lengths
    b = 1/a;

    Nx = round(a/h);
    Ny = round(b/h);
    hx = a/Nx;
    hy = b/Ny;

    x = linspace(0,a,Nx+1);
    y = linspace(0,b,Ny+1);
    xx = kron(x',ones(1,Ny+1));    % grid on (0,a) X (0,b)
    yy = kron(ones(Nx+1,1),y);

    z = ones(Nx+1,Ny+1);            % boundary indicator
    z(1,:) = 0;z(Nx+1,:) = 0;z(:,1) = 0;z(:,Ny+1) = 0;
    Z = spdiags(z(:,0,(Nx+1)*(Ny+1),(Nx+1)*(Ny+1));
    I = speye((Nx+1)*(Ny+1));      % identity

```

```

dx = spdiags(ones(Nx,1),1,Nx,Nx+1) ...
- spdiags(ones(Nx,1),0,Nx,Nx+1);
dx = dx/hx;
iy = speye(Ny+1);
Dx = kron(iy,dx); % d/dx

dy = spdiags(ones(Ny,1),1,Ny,Ny+1) ...
- spdiags(ones(Ny,1),0,Ny,Ny+1);
dy = dy/hy;
ix = speye(Nx+1);
Dy = kron(dy,ix); % d/dy

L = Dx'*Dx + Dy'*Dy; % L*w ~= -(wxx + wyy)
L = Z*L + (I-Z); % w=0 on boundary

f = f0*ones(Nx+1,Ny+1); % -(wxx + wyy) = -f0
f = Z*f(:); % w=0 on boundary

% solution
w = L\f(:);
w = reshape(w,Nx+1,Ny+1);
Q = sum(w(:))*hx*hy; % flow output

Qv = [Qv,Q];
sv = [sv,a];
i = find(Qv == max(Qv)); % max flow

% plot the results
subplot(1,3,1) % plot side length vs flow
plot(sv,Qv,'b',sv(i),Qv(i),'r*')
legend(sprintf('Q mit 1/h=%0.0f',1/h),sprintf('amax=%0.1e',s(i)))
xlabel('a')
ylabel('Q')
title(sprintf('Durchfluss, Qmax=%0.2e',Qv(i)))

subplot(1,3,2) % plot the velocity
surf(xx,yy,w)
axis([0 a 0 b 0 0.08]);
pbaspect([a b 1])
xlabel('x');
ylabel('y');
title(sprintf('Geschwindigkeit\nmax=%0.2e',max(w(:))));

Lw = reshape(L*w(:),Nx+1,Ny+1);
subplot(1,3,3) % plot the Laplacian of the velocity
surf(xx,yy,Lw)
axis([0 a 0 b 0 1.1]);

```

```

pbaspect([a b 1])
xlabel('x');
ylabel('y');
title(sprintf('Laplace der Geschwindigkeit\nmax=%0.2e',max(Lw(:))))
drawnow;
end

i = find(Qv == max(Qv)); % max flow
Q = Qv(i);
a = sv(i); % optimal side lengths
b = 1/a;
Nx = round(a/h);
Ny = round(b/h);
hx = a/Nx;
hy = b/Ny;
x = linspace(0,a,Nx+1);
y = linspace(0,b,Ny+1);
xx = kron(x',ones(1,Ny+1));
yy = kron(ones(Nx+1,1),y); % grid on (0,a) X (0,b)
z = ones(Nx+1,Ny+1); % boundary indicator
z(1,:) = 0;z(Nx+1,:) = 0;z(:,1) = 0;z(:,Ny+1) = 0;
Z = spdiags(z(:,0,(Nx+1)*(Ny+1),(Nx+1)*(Ny+1)));
I = speye((Nx+1)*(Ny+1)); % identity
dx = spdiags(ones(Nx,1),1,Nx,Nx+1) ...
- spdiags(ones(Nx,1),0,Nx,Nx+1);
dx = dx/hx;
iy = speye(Ny+1);
Dx = kron(iy,dx); % d/dx
dy = spdiags(ones(Ny,1),1,Ny,Ny+1) ...
- spdiags(ones(Ny,1),0,Ny,Ny+1);
dy = dy/hy;
ix = speye(Nx+1);
Dy = kron(dy,ix); % d/dy
L = Dx'*Dx + Dy'*Dy; % L*w ~= -(wxx + wyy)
L = Z*L + (I-Z); % w=0 on boundary
f = f0*ones(Nx+1,Ny+1); % -(wxx + wyy) = -f0
f = Z*f(:); % w=0 on boundary
w = L\f;
w = reshape(w,Nx+1,Ny+1);

subplot(1,3,2) % plot the velocity
surf(xx,yy,w)
axis([0 a 0 b 0 0.08]);
pbaspect([a b 1])
xlabel('x')
ylabel('y')
zlabel('w')

```

```

title(sprintf('Geschwindigkeit\nmax=%0.2e',max(w(:)))) 

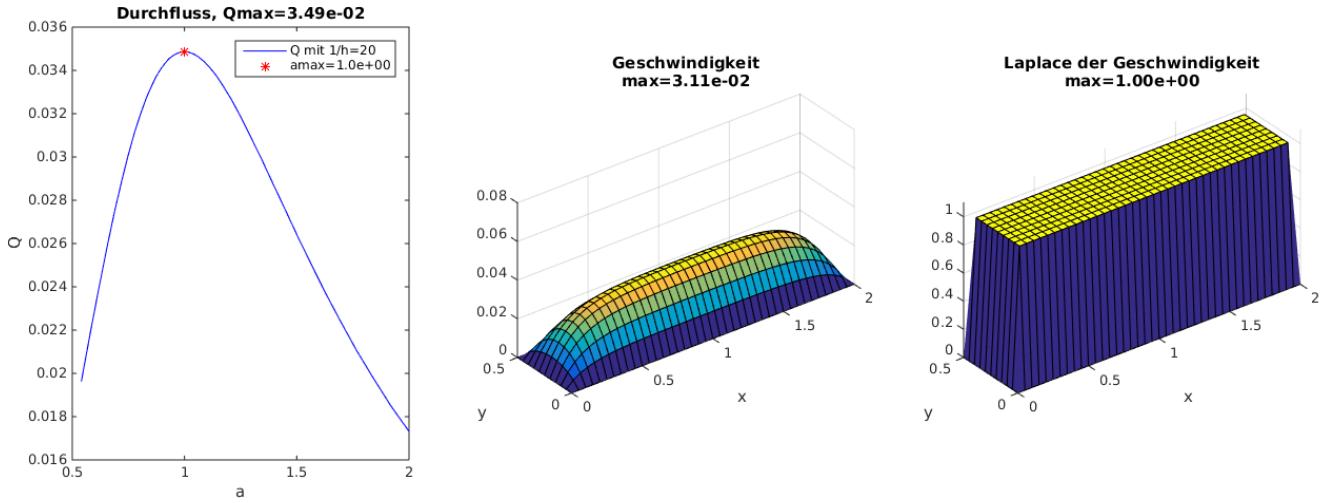
Lw = reshape(L*w(:,Nx+1,Ny+1); 
subplot(1,3,3) % plot the Laplacian of the velocity
surf(xx,yy,Lw)
axis([0 a 0 b 0 1.1]);
pbaspect([a b 1])
xlabel('x');
ylabel('y');
title(sprintf('Laplace der Geschwindigkeit\nmax=%0.2e',max(Lw(:)))) 

disp(sprintf('Qmax=%0.2e, amax=%0.2e',Q,a))

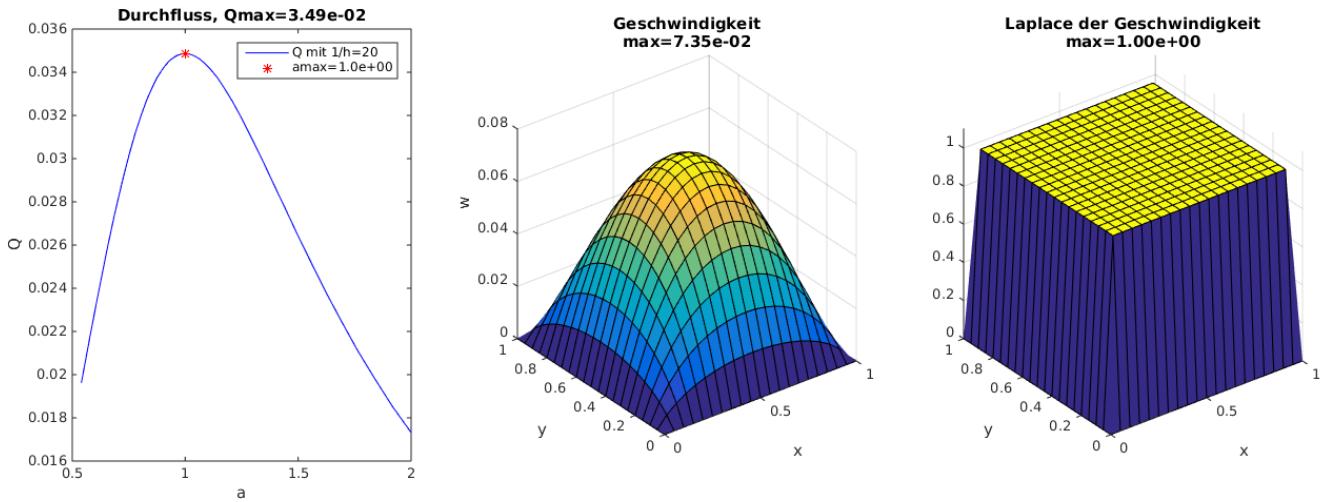
```

Die Ergebnisse werden folgendermaßen grafisch dargestellt. Zuerst wird ein Fall gezeigt, in dem das Gebiet $(0, a) \times (0, b)$, $b = 1/a$, nicht quadratisch ist, und zwar $a = 2$ mit dem Durchfluss unten rechts

im linken Graphen.



Die Stelle mit dem roten Sternchen entspricht $a = 1$, und dieser Fall wird hier gezeigt:



Klar wird der Durchfluss bei $a = b = 1$ maximiert.

* Teil (d) Allgemeines Gebiet, Numerische Rechnungen

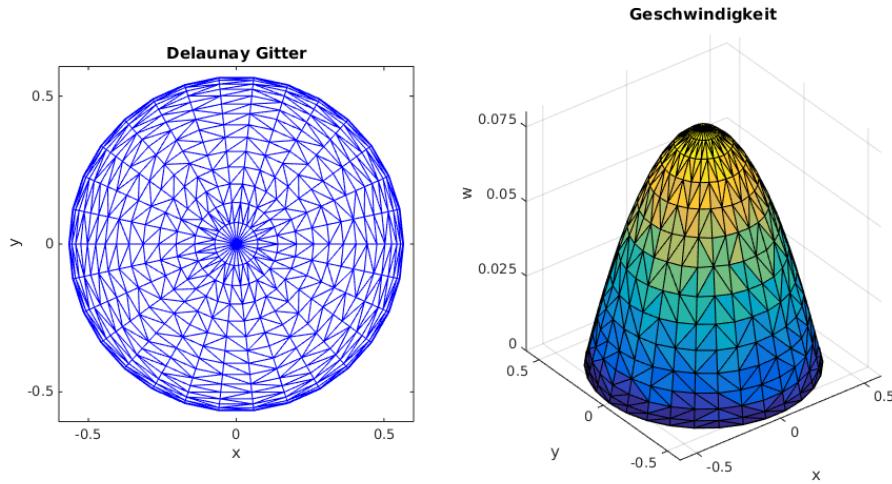
Um eine allgemeine Geometrie zu ermöglichen, wird die Lösung w des Randwertproblems

$$w_{xx} + w_{yy} = f, \quad (x, y) \in \Omega, \quad w = 0, \quad (x, y) \in \partial\Omega$$

mit

$$f = -(p_2 - p_1)/(L\mu) \quad \Omega = B(0, 1/\sqrt{\pi})$$

nun mit finiten Elementen berechnet, wobei der Flächeninhalt $|\Omega| = 1$ erfüllt. Weitere Details dieser Methode befinden sich in diesem [Skriptum](#) besonders ab Seite 226. Diese Ergebnisse



werden mit dem folgenden Matlab Code berechnet, wobei die Schritte ausführlich mit Kommentare erklärt werden. Die numerische Lösung wird auch mit der exakten Lösung im Code verglichen.

```
% set up figure
h1 = figure(1); close(h1); h1 = figure(1);
set(h1,'Position',[100 10 1000 500]);

p2 = 2; % pressures with
p1 = 1; % Dp = 1
Lm = p2-p1; % length*viscosity so that f0=1
f0 = (p2-p1)/Lm; % wxx + wyy = -f0

% spatial parameters
Nr = 15; % number of radial points
Nt = 30; % number of polar points

% make elements
r = linspace(0,1,Nr+1); % make ideal elements on convex domain
r = (1 - (1 - r).^2); % radial points 0 to 1
r = r(2:(Nr+1));
t = 2*pi*linspace(0,1,Nt+1); % polar points 0 to 2*pi
t = t(1:Nt);
x = kron(r',cos(t)); % x = r*cos(t)
y = kron(r',sin(t)); % y = r*sin(t)
b = zeros(Nr,Nt); % boundary indicator
b(Nr,:) = 1; % include origin
x = [0;x(:)];
y = [0;y(:)];
nodes = [x,y]; % list of nodes
```

```

n_nodes = size(nodes,1);
elements = delaunay(x,y);           % elements by delaunay
n_elements = size(elements,1);
bndry_nodes = find(b == 1);         % boundary nodes

% constrain area
u = ones(Nt,1);                   % this particular start gives area = pi
t = 2*pi*linspace(0,1,Nt+1);       % polar points
a = 0;
for j=1:Nt                         % vertices of triangle, origin to boundary
    x1 = u(j)*cos(t(j)); x2 = u(j)*cos(t(j+1)); x3 = 0; % 1st, 2nd and 3rd
    y1 = u(j)*sin(t(j)); y2 = u(j)*sin(t(j+1)); y3 = 0; % vertices
    A = [x2-x1,x3-x1;y2-y1,y3-y1];
    detA = det(A);           % detA/2 = area of current triangle
    a = a + 0.5*detA;        % sum up all triangle areas
end
% current area is
%     a = int_0^(2*pi) int_0^u(t) r dr dt
%     = int_0^(2*pi) u(t)^2/2 dt
% constrained to be 1,
%     1 = (1/a)a = int_0^(2*pi) (1/a)u(t)^2/2 dt
%     = int_0^(2*pi) [u(t)/sqrt(a)]^2/2 dt
% so set u = u/sqrt(a)
u = u/sqrt(a);                   % scale u to have area = 1
r = linspace(0,1,Nr+1);           % radial points 0 to 1
r = (1 - (1 - r).^2);
r = r(2:(Nr+1));
t = 2*pi*linspace(0,1,Nt+1);       % polar points 0 to 2*pi
t = t(1:Nt);
x = kron(r',cos(t))*diag(u);     % x(i,j) = u(j)*r(i)*cos(t(j))
y = kron(r',sin(t))*diag(u);     % y(i,j) = u(j)*r(i)*sin(t(j))
x = [0;x(:)];                   % include origin
y = [0;y(:)];
nodes = [x,y];                   % new nodes but same elements in
                                  % case domain loses convexity
subplot(1,2,1)
triplot(elements,x,y)           % graph Delaunay triangulation
xlabel('x')
ylabel('y')
title('Delaunay Gitter')
axis([-0.6 +0.6 -0.6 +0.6])
set(gca,'Xtick',[-0.5 0 +0.5],'Ytick',[-0.5 0 +0.5])
pbaspect([1 1 1])
drawnow;

% force
f = f0*ones(n_nodes,1);

```

```

% stiffness (L) and mass (G) matrices
I = speye(n_nodes);
L = zeros(n_nodes,n_nodes);
G = zeros(n_nodes,n_nodes);
for e=1:n_elements
    E = elements(e,:); % current element
    X1 = nodes(E(1,:)); x1=X1(1); y1=X1(2); % X1, X2 and X3 are
    X2 = nodes(E(2,:)); x2=X2(1); y2=X2(2); % vertices of the
    X3 = nodes(E(3,:)); x3=X3(1); y3=X3(2); % current element
    A = [x2-x1,x3-x1;y2-y1,y3-y1];
    A1 = inv(A);
    detA = det(A); % detA/2 = area of current element
    p = [1,1,1]/3; % ref basis evaluation in centroid
    pp = p'*p;
    Dp = [-1,1,0;-1,0,1]; % ref basis gradient is constant
    DpDp = (A1'*Dp)*(A1'*Dp); % transformed to local basis
    L(E,E) = L(E,E) + 0.5*detA*DpDp; % assembly of stiffness matrix
    G(E,E) = G(E,E) + 0.5*detA*pp; % assembly of mass matrix
end
F = G*f; % force vector
Z = I; % boundary indicator
for e=1:length(bndry_nodes)
    E = bndry_nodes(e);
    Z(E,:) = 0;
end
L = Z*L + (I-Z); % stiffness but identity on boundary
F = Z*F; % force but zero on boundary

w = L\F; % solution at nodes

wz = (1 - pi*(x.^2 + y.^2))/(4*pi); % exact solution
disp(sprintf('max|w-w*| = %0.2e',max(abs(w(:)-wz(:)))));

subplot(1,2,2)
trisurf(elements,x,y,w) % plot the solution
xlabel('x')
ylabel('y')
zlabel('w')
title('Geschwindigkeit')
axis([-0.6 +0.6 -0.6 +0.6 0 0.08])
set(gca,'Xtick',[ -0.5 0 +0.5], ...
    'Ytick',[ -0.5 0 +0.5], ...
    'Ztick',[ 0 +0.025 +0.05 +0.075])
drawnow;

Q = sum(G*w); % flow output, integral of w

```

Nun wird der Durchfluss bezüglich der Geometrie unter der Einschränkung $a = |\Omega| = 1$ maximiert. Teile des oben stehenden Codes werden im folgenden Matlab Code aufgerufen, um mit der Matlab Funktion `fminunc` die Maximierung durchzuführen. Wie bei dem letzten Code werden die Schritte ausführlich mit Kommentare erklärt.

```

function B08Bsp1d

    global Nr Nt f0 n_nodes n_elements nodes bndry_nodes elements Jv

    h1 = figure(1);
    close(h1);
    h1 = figure(1);
    set(h1,'Position',[10 10 1500 500]);

    p2 = 2;                                % pressures with
    p1 = 1;                                %      Dp = 1
    Lm = p2-p1;                            % length*viscosity so that f0=1
    f0 = (p2-p1)/Lm;                        % wxx + wyy = -f0

    % spatial parameters
    Nr    = 15;                            % number of radial points
    Nt    = 30;                            % number of polar points

    % make elements
    r = linspace(0,1,Nr+1);                % make ideal elements on convex domain
    r = (1 - (1 - r).^2);                  % radial points 0 to 1
    r = r(2:(Nr+1));
    t = 2*pi*linspace(0,1,Nt+1);          % polar points 0 to 2*pi
    t = t(1:Nt);
    x = kron(r',cos(t));                  % x = r*cos(t)
    y = kron(r',sin(t));                  % y = r*sin(t)
    b = zeros(Nr,Nt);                     % boundary indicator
    b(Nr,:) = 1;
    x = [0;x(:)];                        % include origin
    y = [0;y(:)];                        % list of nodes
    nodes = [x,y];
    n_nodes = size(nodes,1);
    elements = delaunay(x,y);            % elements by delaunay
    n_elements = size(elements,1);
    bndry_nodes = find(b == 1);           % boundary nodes

    Jv = [];                                % initialize cost function evaluations
    u0 = ones(Nt,1)/sqrt(pi) ...            % initialize u with a perturbation
        + 0.1*randn(Nt,1);                  %      of the known exact solution

    u  = fminunc(@flow,u0);                  % perform optimization

end

```

```

function J = flow(u)

    global Nr Nt f0 n_nodes n_elements nodes bndry_nodes elements Jv

% constrain area
    t = 2*pi*linspace(0,1,Nt+1);           % polar points
    a = 0;
    for j=1:Nt                           % vertices of triangle, origin to boundary
        x1 = u(j)*cos(t(j)); x2 = u(j)*cos(t(j+1)); x3 = 0; % 1st, 2nd and 3rd
        y1 = u(j)*sin(t(j)); y2 = u(j)*sin(t(j+1)); y3 = 0; % vertices
        A = [x2-x1,x3-x1;y2-y1,y3-y1];
        detA = det(A);                      % detA/2 = area of current triangle
        a = a + 0.5*detA;                  % sum up all triangle areas
    end
% current area is
%     a = int_0^(2*pi) int_0^u(t) r dr dt
%         = int_0^(2*pi) u(t)^2/2 dt
% constrained to be 1,
%     1 = (1/a)a = int_0^(2*pi) (1/a)u(t)^2/2 dt
%         = int_0^(2*pi) [u(t)/sqrt(a)]^2/2 dt
% so set u = u/sqrt(a)
    u = u/sqrt(a);                      % scale u to have area = 1
    r = linspace(0,1,Nr+1);             % radial points 0 to 1
    r = (1 - (1 - r).^2);
    r = r(2:(Nr+1));
    t = 2*pi*linspace(0,1,Nt+1);       % polar points 0 to 2*pi
    t = t(1:Nt);
    x = kron(r',cos(t))*diag(u);      % x(i,j) = u(j)*r(i)*cos(t(j))
    y = kron(r',sin(t))*diag(u);      % y(i,j) = u(j)*r(i)*sin(t(j))
    x = [0;x(:)];                   % include origin
    y = [0;y(:)];
    nodes = [x,y];                  % new nodes but same elements in
                                    % case domain loses convexity
    subplot(1,3,1)
    triplot(elements,x,y)           % graph Delaunay triangularization
    xlabel('x')
    ylabel('y')
    title('Delaunay Gitter')
    gr = max(0.6,1.05*max(abs(nodes(:)))); % set plot scale
    axis([-gr +gr -gr +gr])
    set(gca,'Xtick',[-0.5 0 +0.5],'Ytick',[-0.5 0 +0.5])
    pbaspect([1 1 1])
    drawnow;

% force
    f = f0*ones(n_nodes,1);

```

```

% stiffness (L) and mass (G) matrices
I = speye(n_nodes);
L = zeros(n_nodes,n_nodes);
G = zeros(n_nodes,n_nodes);
for e=1:n_elements
    E = elements(e,:); % current element
    X1 = nodes(E(1,:)); x1=X1(1); y1=X1(2); % X1, X2 and X3 are
    X2 = nodes(E(2,:)); x2=X2(1); y2=X2(2); % vertices of the
    X3 = nodes(E(3,:)); x3=X3(1); y3=X3(2); % current element
    A = [x2-x1,x3-x1;y2-y1,y3-y1];
    A1 = inv(A);
    detA = det(A); % detA/2 = area of current element
    p = [1,1,1]/3; % ref basis evaluation in centroid
    pp = p'*p;
    Dp = [-1,1,0;-1,0,1]; % ref basis gradient is constant
    DpDp = (A1'*Dp)*(A1*Dp); % transformed to local basis
    L(E,E) = L(E,E) + 0.5*detA*DpDp; % assembly of stiffness matrix
    G(E,E) = G(E,E) + 0.5*detA*pp; % assembly of mass matrix
end
F = G*f; % force vector
Z = I; % boundary indicator
for e=1:length(bndry_nodes)
    E = bndry_nodes(e);
    Z(E,:) = 0;
end
L = Z*L + (I-Z); % stiffness but identity on boundary
F = Z*F; % force but zero on boundary

w = L\F; % solution at nodes

subplot(1,3,2)
trisurf(elements,x,y,w) % plot the solution
xlabel('x')
ylabel('y')
zlabel('w')
title('Geschwindigkeit')
axis([-gr +gr -gr +gr 0 0.08])
set(gca,'Xtick',[-0.5 0 +0.5], ...
    'Ytick',[-0.5 0 +0.5], ...
    'Ztick',[0 +0.025 +0.05 +0.075])
drawnow;

Q = sum(G*w); % flow output, integral of w
J = -Q + (a-1)^2; % min(-q) and min(area - 1)^2

Jv = [Jv,J];
subplot(1,3,3)
plot(Jv) % plot cost functional evaluations

```

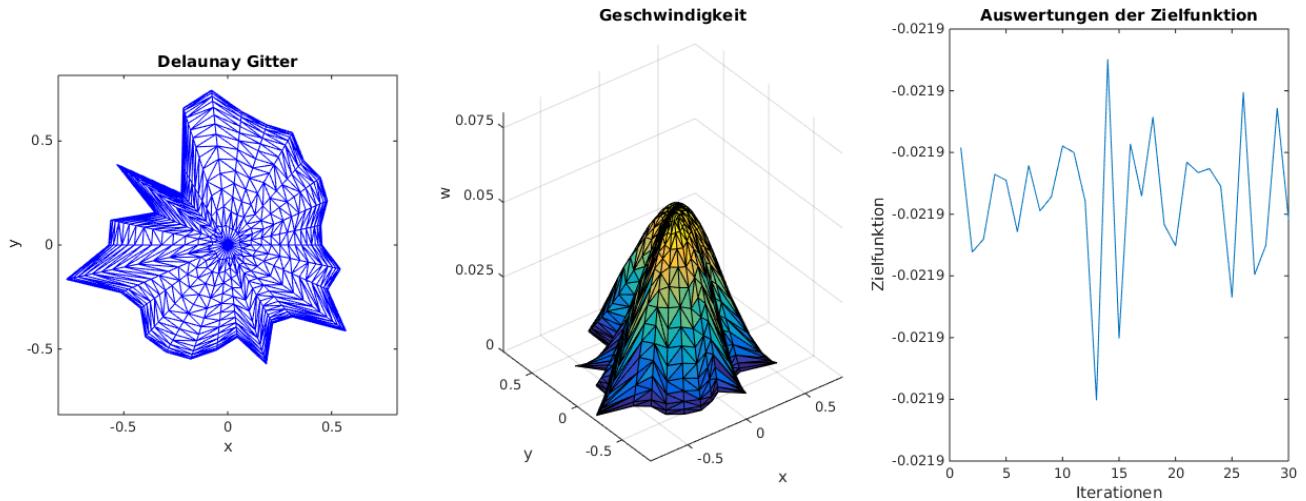
```

xlabel('Iterationen')
ylabel('Zielfunktion')
title('Auswertungen der Zielfunktion')
drawnow;

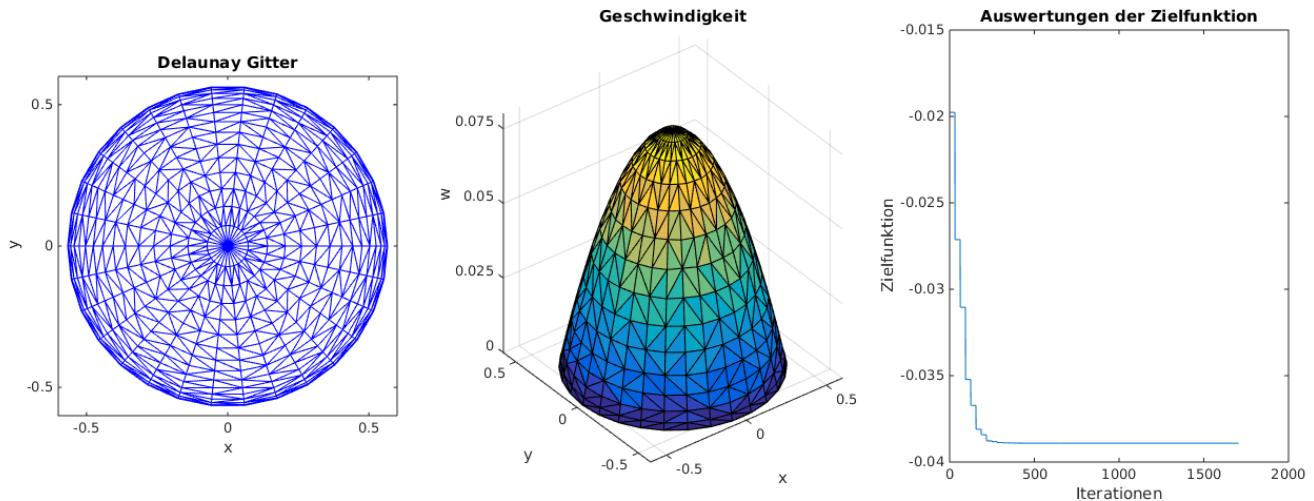
end

```

Mit der Einschränkung $|\Omega| = 1$ ist die bekannte Lösung, dass $\Omega = B(0, 1/\sqrt{\pi})$ gelten soll. Diese Lösung wird anfänglich gestört, wie man in der folgenden Grafik von den ersten Iterationen sieht.



Nachdem die Iterationen für die Maximierung konvergiert, sehen die Ergebnisse folgendermaßen aus.



Diese numerische Lösung stimmt mit der bekannten exakten Lösung überein.

* Teil (e) Allgemeines Gebiet, Variationelle Rechnungen

Die Flussgeschwindigkeit $w(x, y)$ des Flusses sei allgemein in einem sternförmigen Rohrprofil Ω definiert,

$$\Omega = \{(x, y) : \sqrt{x^2 + y^2} < u(\theta), y = x \tan(\theta)\}$$

wobei $u(\theta)$ eine zu bestimmende Funktion vom Polarwinkel θ ist. Diese Funktion $u(\theta)$ sei durch den Flächeninhalt eingeschränkt,

$$1 = \int_0^{2\pi} \int_0^{u(\theta)} r dr d\theta = \int_0^{2\pi} \frac{1}{2} u(\theta)^2 d\theta.$$

Die Flussgeschwindigkeit ist durch die Poisson-Gleichung eingeschränkt, $w_{xx} + w_{yy} = -\kappa$ in Ω , wobei die Konstante $\kappa = (p_2 - p_1)\ell/\mu > 0$ fixiert ist. Hier ist $p_2 - p_1$ die Druckdifferenz entlang der Länge ℓ des Rohrs. Wegen der Viskosität μ gibt es Reibung am Rand des Rohrs, und deshalb gilt die Randbedingung $w = 0$ auf $\partial\Omega$. Der Fluss durch das Rohr mit Profil Ω ist gegeben durch

$$F = \int_{\Omega} w(x, y) dx dy.$$

Die zutreffende Lagrange-Funktion, die stationär in der gesuchten Funktion u sein soll, ist

$$L(u, \lambda) = \int_{\Omega} w(x, y) dx dy - \lambda \left[1 - \int_0^{2\pi} \frac{1}{2} u(\theta)^2 d\theta \right]$$

unter der Nebenbedingung

$$w_{xx} + w_{yy} = -\kappa \text{ in } \Omega, \quad w = 0 \text{ auf } \partial\Omega$$

wobei w implizit von u abhängt. Um die variationellen Ableitungen von L zu erleichtern wird das Gebiet Ω in den Einheitskreis abgebildet,

$$K = \{(\rho, \theta) : 0 \leq \rho \leq 1, 0 \leq \theta \leq 2\pi\}$$

und zwar mit der Parameterisierung,

$$\begin{aligned} \rho &= r/u(\theta), \quad r = \sqrt{x^2 + y^2}, \quad y = x \tan(\theta) \\ x &= \rho u(\theta) \cos(\theta), \quad y = \rho u(\theta) \sin(\theta). \end{aligned}$$

Die partiellen Ableitungen für die Poisson-Gleichung sind

$$\begin{aligned} w_x &= w_{\rho} \rho_x + w_{\theta} \theta_x \\ w_y &= w_{\rho} \rho_y + w_{\theta} \theta_y \\ w_{xx} &= w_{\rho} \rho \rho_x^2 + w_{\rho} \theta \rho_x \theta_x + w_{\rho} \rho_{xx} + w_{\theta} \theta \rho_x^2 + w_{\theta} \rho \theta_x \rho_x + w_{\theta} \theta_{xx} \\ w_{yy} &= w_{\rho} \rho \rho_y^2 + w_{\rho} \theta \rho_y \theta_y + w_{\rho} \rho_{yy} + w_{\theta} \theta \rho_y^2 + w_{\theta} \rho \theta_y \rho_y + w_{\theta} \theta_{yy} \end{aligned}$$

wobei

$$\rho_x = \partial_x \left[\frac{r}{u(\theta)} \right] = \dots = \frac{u(\theta) \cos(\theta) + u'(\theta) \sin(\theta)}{u^2(\theta)}$$

$$\rho_y = \partial_y \left[\frac{r}{u(\theta)} \right] = \dots = \frac{u(\theta) \sin(\theta) - u'(\theta) \cos(\theta)}{u^2(\theta)}$$

$$\theta_x = \partial_x \tan^{-1} \left(\frac{y}{x} \right) = \dots = -\frac{\sin(\theta)}{\rho u(\theta)}$$

$$\theta_y = \partial_y \tan^{-1} \left(\frac{y}{x} \right) = \dots = \frac{\cos(\theta)}{\rho u(\theta)}$$

und

$$\rho_{xx} = \partial_x^2 \left[\frac{r}{u(\theta)} \right] = \dots = \frac{\sin^2(\theta)[u^2(\theta) + 2u'^2(\theta) - u(\theta)u''(\theta)]}{\rho u^4(\theta)}$$

$$\rho_{yy} = \partial_y^2 \left[\frac{r}{u(\theta)} \right] = \dots = \frac{\cos^2(\theta)[u^2(\theta) + 2u'^2(\theta) - u(\theta)u''(\theta)]}{\rho u^4(\theta)}$$

$$\theta_{xx} = \partial_x^2 \tan^{-1} \left(\frac{y}{x} \right) = \dots = \frac{\sin(2\theta)}{\rho^2 u^2(\theta)}$$

$$\theta_{yy} = \partial_y^2 \tan^{-1} \left(\frac{y}{x} \right) = \dots = -\frac{\sin(2\theta)}{\rho^2 u^2(\theta)}$$

und für die Integrale gibt es schließlich

$$1/\det \left[\frac{\partial(\rho, \theta)}{\partial(x, y)} \right] = \dots = \rho u(\theta).$$

Dann mit $w = w(\rho, \theta)$ und

$$w_{xx} + w_y =$$

$$w_{\rho\rho} \rho_x^2 + w_{\rho\theta} \rho_x \theta_x + w_{\rho} \rho_{xx} + w_{\theta\theta} \theta_x^2 + w_{\theta\rho} \rho_x \theta_x + w_{\theta} \theta_{xx} +$$

$$w_{\rho\rho} \rho_y^2 + w_{\rho\theta} \rho_y \theta_y + w_{\rho} \rho_{yy} + w_{\theta\theta} \theta_y^2 + w_{\theta\rho} \rho_y \theta_y + w_{\theta} \theta_{yy} =$$

$$w_{\rho\rho} \left[\frac{u \cos(\theta) + u' \sin(\theta)}{u^2} \right] + w_{\rho} \left[\frac{\sin^2(\theta)[u^2 + 2u'^2 - uu'']}{\rho u^4} \right] + 2w_{\rho\theta} \left[\frac{u \cos(\theta) + u' \sin(\theta)}{u^2} \right] \left[-\frac{\sin(\theta)}{\rho u} \right] + w_{\theta\theta} \left[-\frac{\sin(\theta)}{\rho u} \right]^2 + w_{\theta} \left[\frac{\sin(2\theta)}{\rho^2 u^2} \right]$$

$$+ w_{\rho\rho} \left[\frac{u \sin(\theta) - u' \cos(\theta)}{u^2} \right] + w_{\rho} \left[\frac{\cos^2(\theta)[u^2 + 2u'^2 - uu'']}{\rho u^4} \right] + 2w_{\rho\theta} \left[\frac{u \sin(\theta) - u' \cos(\theta)}{u^2} \right] \left[\frac{\cos(\theta)}{\rho u} \right] + w_{\theta\theta} \left[\frac{\cos(\theta)}{\rho u} \right]^2 + w_{\theta} \left[-\frac{\sin(2\theta)}{\rho^2 u^2} \right]$$

$$= w_{\rho\rho} \left[\frac{u^2 + u'^2}{u^4} \right] + w_{\rho} \left[\frac{u^2 + 2u'^2 - uu''}{\rho u^4} \right] + 2w_{\rho\theta} \left[-\frac{u'}{\rho u^3} \right] + w_{\theta\theta} \left[\frac{1}{\rho^2 u^2} \right]$$

gilt

$$-\kappa = [w_{xx} + w_{yy}] = w_{\rho\rho} \frac{u^2 + u'^2}{u^4} + w_{\rho} \frac{u^2 + 2u'^2 - uu''}{\rho u^4} - w_{\rho\theta} \frac{2u'}{\rho u^3} + w_{\theta\theta} \frac{1}{\rho^2 u^2}.$$

Die Lagrange-Funktion nimmt die Form

$$L(u, \lambda) = \int_0^{2\pi} \int_0^1 w u \rho d\rho d\theta + \lambda \left[\int_0^{2\pi} \frac{1}{2} u^2 d\theta - 1 \right].$$

Die variationelle Ableitung von L nach u an der Stelle u^* mit Störung v ist definiert durch

$$\frac{\delta L}{\delta u}(u^*, \lambda; v) = \lim_{\epsilon \rightarrow 0} \frac{L(u^* + \epsilon v, \lambda) - L(u^*, \lambda)}{\epsilon}$$

Zu zeigen ist, mit $u^*(\theta) = 1/\sqrt{\pi}$ gilt

$$\frac{\delta L}{\delta u}(u^*, \lambda; v) = 0, \quad \forall v \in \mathcal{C}^\infty([0, 2\pi]) \text{ periodisch.}$$

Die variationelle Ableitung ist

$$\frac{\delta L}{\delta u}(u, \lambda; v) = \int_0^{2\pi} \int_0^1 \left[\frac{\delta w}{\delta u} u + wv \right] \rho d\rho d\theta + \lambda \int_0^{2\pi} u v d\theta$$

wobei die variationelle Ableitung von w nach u mit Störung v gegeben ist durch

$$\frac{\delta}{\delta u} \left\{ w_{\rho\rho} \frac{u^2 + u'^2}{u^3} + w_\rho \frac{u^2 + 2u'^2 - uu''}{\rho u^3} - w_{\rho\theta} \frac{2u'}{\rho u^2} + w_{\theta\theta} \frac{1}{\rho^2 u} + \kappa u \right\} (u; v) = 0$$

d.h. $(\delta w / \delta u)(u; v)$ erfüllt die Poisson-Gleichung,

$$\begin{aligned} & \left\{ \left(\frac{\delta w}{\delta u} \right)_{\rho\rho} \left[\frac{u^2 + u'^2}{u^3} \right] + \right. \\ & \quad \left(\frac{\delta w}{\delta u} \right)_\rho \left[\frac{u^2 + 2u'^2 - uu''}{\rho u^3} \right] \\ & \quad \left. - \left(\frac{\delta w}{\delta u} \right)_{\rho\theta} \left[\frac{2u'}{\rho u^2} \right] + \right. \\ & \quad \left. \left(\frac{\delta w}{\delta u} \right)_{\theta\theta} \left[\frac{1}{\rho^2 u} \right] \right\} = \\ & - \left\{ w_{\rho\rho} \left[\frac{(2uv + 2u'v')u^3 - (u^2 + u'^2)(3u^2v)}{u^6} \right] + \right. \\ & \quad w_\rho \left[\frac{(2uv + 4u'v' - vu'' - uv'')u^3 - (u^2 + 2u'^2 - uu'')(3u^2v)}{\rho u^6} \right] \\ & \quad \left. - w_{\rho\theta} \left[\frac{(2v')u^2 - 2u'(2uv)}{\rho u^4} \right] + \right. \\ & \quad \left. w_{\theta\theta} \left[\frac{-v}{\rho^2 u^2} \right] + \kappa v \right\} \text{ in } K \\ & \quad \frac{\delta w}{\delta u}(u; v) = 0, \text{ auf } \partial K \end{aligned}$$

Mit $u^* = 1/\sqrt{\pi}$ ist w gegeben durch

$$w(\rho, \theta) = \frac{\kappa}{4\pi} (1 - \pi \rho^2)$$

und mit $w_\theta = 0$, $w_\rho = -\kappa \rho / 2$, $w_{\rho\rho} = -\kappa / 2$ erfüllt $(\delta w / \delta u)(u; v)$

$$\begin{cases} \left[\left(\frac{\delta w}{\delta u} \right)_{\rho\rho} + \frac{1}{\rho} \left(\frac{\delta w}{\delta u} \right)_\rho + \frac{1}{\rho^2} \left(\frac{\delta w}{\delta u} \right)_{\theta\theta} \right] (u^*; v) = -\kappa \left[\frac{1 + \pi}{\sqrt{\pi}} v + \frac{\sqrt{\pi}}{2} v'' \right] \text{ in } K \\ \frac{\delta w}{\delta u}(u; v) = 0 \text{ auf } \partial K. \end{cases}$$

Laut einem Greenschen Satz gilt

$$\int_{\Omega} \left[w \Delta \left(\frac{\delta w}{\delta u} \right) - \left(\frac{\delta w}{\delta u} \right) \Delta w \right] dx dy = \int_{\partial\Omega} \left[\underbrace{w}_{=0} \frac{\partial}{\partial n} \left(\frac{\delta w}{\delta u} \right) - \underbrace{\left(\frac{\delta w}{\delta u} \right)}_{=0} \frac{\partial w}{\partial n} \right] ds \quad (ds^2 = dx^2 + dy^2).$$

Mit $u^* = 1/\sqrt{\pi}$ und den gewonnenen Charakterisierungen für w und $\delta w/\delta u = 0$ folgen

$$\int_{\Omega} \frac{\delta w}{\delta u}(u^*; v) \Delta w dx dy = \int_0^{2\pi} \int_0^1 \frac{\delta w}{\delta u}(u^*; v) (-\kappa)(u^* \rho d\rho d\theta) = -\frac{\kappa}{\sqrt{\pi}} \int_0^{2\pi} \int_0^1 \frac{\delta w}{\delta u}(u^*; v) \rho d\rho d\theta$$

und

$$\begin{aligned} \int_{\Omega} \frac{\delta w}{\delta u}(u^*; v) \Delta w dx dy &= \int_{\Omega} w \Delta \left(\frac{\delta w}{\delta u}(u^*; v) \right) dx dy \\ &= \frac{\kappa}{4\pi} \int_0^{2\pi} \int_0^1 (1 - \pi \rho^2) \Delta \left(\frac{\delta w}{\delta u}(u^*; v) \right) (u^* \rho d\rho d\theta) \\ &= \frac{\kappa}{4\pi} \int_0^{2\pi} \int_0^1 (1 - \pi \rho^2) (-\kappa) \left[\frac{1 + \pi}{\sqrt{\pi}} + \frac{\sqrt{\pi}}{2} v'' \right] \frac{\rho d\rho d\theta}{\sqrt{\pi}} \\ &= -\frac{\kappa^2 (1 + \pi)}{4\pi^2} \left[\int_0^1 (1 - \pi \rho^2) \rho d\rho \right] \left[\int_0^{2\pi} v(\theta) d\theta \right] - \frac{\kappa^2}{8\pi} \left[\int_0^1 (1 - \pi \rho^2) \rho d\rho \right] \left[\int_0^{2\pi} v''(\theta) d\theta \right] \\ &= -\frac{\kappa^2 (1 + \pi)}{4\pi^2} \left[\frac{1}{2} - \frac{\pi}{4} \right] \left[\int_0^{2\pi} v(\theta) d\theta \right] - \frac{\kappa^2}{8\pi} \left[\frac{1}{2} - \frac{\pi}{4} \right] \underbrace{\left[v'(\theta) \Big|_0^{2\pi} \right]}_{=0} \\ &= \frac{\kappa^2}{16\pi^2} (1 + \pi)(\pi - 2) \int_0^{2\pi} v(\theta) d\theta \end{aligned}$$

oder

$$\frac{1}{\sqrt{\pi}} \int_0^{2\pi} \int_0^1 \frac{\delta w}{\delta u}(u^*; v) \rho d\rho d\theta = -\frac{\kappa}{16\pi^2} (1 + \pi)(\pi - 2) \int_0^{2\pi} v(\theta) d\theta.$$

Mit $u^* = 1/\sqrt{\pi}$ und $w = \kappa(1 - \pi \rho^2)/(4\pi)$ gilt

$$\begin{aligned} \frac{\delta L}{\delta u}(u^*, \lambda; v) &= \int_0^{2\pi} \int_0^1 \left[\frac{\delta w}{\delta u}(u^*; v) \frac{1}{\sqrt{\pi}} + \frac{\kappa}{4\pi} (1 - \pi \rho^2) v \right] \rho d\rho d\theta + \lambda \left[\int_0^{2\pi} \frac{v}{\sqrt{\pi}} d\theta \right] \\ &= \frac{1}{\sqrt{\pi}} \int_0^{2\pi} \int_0^1 \frac{\delta w}{\delta u}(u^*; v) \rho d\rho d\theta - \frac{\kappa}{8\pi} (\pi - 2) \int_0^{2\pi} v d\theta + \frac{\lambda}{\sqrt{\pi}} \int_0^{2\pi} v d\theta \\ &= \left[-\frac{\kappa}{16\pi^2} (1 + \pi)(\pi - 2) - \frac{\kappa}{8\pi} (\pi - 2) + \frac{\lambda}{\sqrt{\pi}} \right] \int_0^{2\pi} v(\theta) d\theta. \end{aligned}$$

Also mit

$$u^* = \frac{1}{\sqrt{\pi}} \quad \text{und} \quad \lambda^* = \frac{\kappa(\pi - 2)}{\sqrt{\pi}} \left[\frac{1 + \pi}{16\pi} + \frac{1}{8} \right]$$

gelten

$$\frac{\delta L}{\delta u}(u^*, \lambda^*; v) = 0, \quad \forall v \in \mathcal{C}^\infty([0, 2\pi]) \text{ periodisch}$$

und

$$\frac{\partial L}{\partial \lambda}(u^*, \lambda^*) = 0.$$

- **Beispiel 2: Fischfangflotte**

- **Aufgabe**

Zur Maximierung des langfristigen Profits $P(t) = ph(t) - cu(t)$ einer Fischerei, soll die Zielfunktion

$$J(u) = \int_0^\infty e^{-\delta t} [ph(t) - cu(t)] dt$$

mit dem Raubeffekt für die Fische

$$h(t) = qu(t)x(t), \quad x'(t) = Rx(t)(1 - x(t)/K) - h(t), \quad x(0) = K/N$$

maximiert werden.

(a) Bestimmen Sie die optimale Steuerung der folgenden Form,

$$u(t) = \begin{cases} U_1, & 0 \leq t < s \\ U_2, & s \leq t < \infty \end{cases} \quad \text{wobei } x(t) = x(s), \quad \forall t > s.$$

(b) Wiederholen Sie die Aufgabe mit einer bliebigen Anzahl N von Konstanten Zuständen $u(t) = U_i$, $s_{i-1} \leq t < s_i$, $i = 1, \dots, N$, $U_i \in [0, U_{\max}]$, $U_{\max} = (R/q)(1 - c/(pqK))$, $s_N = \infty$, $x(t) = x(s_{N-1})$, $t > s_{N-1}$, um das diskretisierte optimale Steuerungsproblem allgemein zu lösen.

(c) Zeigen Sie durch variationelle Rechnungen, dass die im Skriptum berichtete Steuerung

$$u(t) = \begin{cases} U_1, & t \in [0, s) \\ U_2, & t \in [s, \infty) \end{cases}, \quad U_1 = \begin{cases} 0, & N > (1 - qU_2/R)^{-1} \\ U_{\max}, & N < (1 - qU_2/R)^{-1} \end{cases},$$

$$U_2 = \frac{R}{4q} \left[3 - \frac{c}{pqK} + \frac{\delta}{R} - \sqrt{\left(1 + \frac{c}{pqK} - \frac{\delta}{R}\right)^2 + \frac{8\delta}{R} \frac{c}{pqK}} \right],$$

$$s = s(U_1, U_2) = \frac{1}{R\rho(U_1)} \log \left[\frac{(N\rho(U_1) - 1)\rho(U_2)}{\rho(U_1) - \rho(U_2)} \right]$$

tatsächlich die allgemeinen Stationaritätsbedingungen zur eingeschränkten Maximierung des Kostenfunktionalen J erfüllt.

- **Lösung**

- * **Teil (a) Zweistufige Steuerung**

Der Zustand x in der Nebenbedingung ist gegeben durch

$$x(t) = \begin{cases} \frac{K\rho(U_1)}{1 + (N\rho(U_1) - 1)\exp(-\rho(U_1)Rt)} & 0 < t \leq s \\ K\rho(U_2), & s \leq t. \end{cases}$$

wobei

$$\rho(U) = 1 - \frac{qU}{R}.$$

Da der langfristige Profit

$$\lim_{t \rightarrow +\infty} P(t) = (pqK(R - qU)/R - c)U$$

positiv sein soll, werden U_1 und U_2 so eingeschränkt,

$$0 \leq U_1, U_2 \leq U_{\max} = \frac{R}{q} \left[1 - \frac{c}{pqK} \right], \quad \text{d.h.} \quad \rho(U_1), \rho(U_2) > 0.$$

Für die Stetigkeit von x an der Stelle $t = s$ muss die Schaltzeit die Gleichung erfüllen,

$$s(U_1, U_2) = \frac{1}{R\rho(U_1)} \log \left[\frac{(N\rho(U_1) - 1)\rho(U_2)}{\rho(U_1) - \rho(U_2)} \right].$$

Damit die Schaltzeit positiv bleibt, muss zusammen mit $\rho(U_1) > 0$ und $\rho(U_2) > 0$ auch die folgende Ungleichung erfüllt werden,

$$Q(U_1, U_2) = \frac{(N\rho(U_1) - 1)\rho(U_2)}{\rho(U_1) - \rho(U_2)} > 1.$$

Falls diese Bedingung nicht erfüllt werden sollte, muss die Schaltzeit zu $+\infty$ gesetzt werden, da der Zustand x in der Phase mit Steuerung U_1 zum Zustand mit Steuerung U_2 nicht stetig geführt werden kann.

Mit diesen Definitionen nimmt das Kostenfunktional die folgende Form.

$$J(u) = U_1 \int_0^{s(U_1, U_2)} e^{-\delta t} \left[\frac{pqK\rho(U_1)}{1 + (N\rho(U_1) - 1) \exp(-\rho(U_1)Rt)} - c \right] dt + U_2 \int_{s(U_1, U_2)}^{\infty} e^{-\delta t} [pqK\rho(U_2) - c] dt$$

Mit dem folgenden Matlab Code wird J maximiert.

```

h1 = figure(1); close(h1); h1 = figure(1);
set(h1,'Position',[10 10 1000 500]);

c = 1; % costs per boat
p = 1; % price per fish
q = 1; % fish catching rate per boat
R = 4; % 1 / time scale for fish logistic growth
K = 4; % fish capacity
d = 1; % continuous interest rate
N = 2; % initially K/N fish
nU = 100; % number of U points for profit evaluation
tmax = 10; % time horizon
nT = 1000; % number of time points for integration

Rq = R/q; % stored parameters
pq = p*q;
pqK = pq*K;
cpqK = c/pqK;
dR = d/R;

Umax = Rq*(1 - cpqK); % 0 < U1, U2 < Umax

```

```

Uopt = (Rq/4)*(3 - cpqK + dR - sqrt((1 + cpqK - dR)^2 + 8*dR*cpqK));

% expected optimum:
% (U1,U2) = ( 0,Uopt) if N(1 - q*Uopt/R) > 1
% (U1,U2) = (Umax,Uopt) if N(1 - q*Uopt/R) < 1

smax = 0;                                % get the maximal switching time smax
for i=1:nU
    u1 = (i-1)*Umax/(nU-1);
    for j=1:nU
        u2 = (j-1)*Umax/(nU-1);

        r1 = (u1 ~= Rq).* (1 - u1/Rq);
        r2 = (u2 ~= Rq).* (1 - u2/Rq);

        if (r1*(N*r2-1)*(r1-r2) > 0)
            Q = (N*r1-1)*r2/(r1-r2);
            s = log(Q)/(R*r1);      % switching time
            smax = max(smax,s);
        end
    end
end
tmax = max(2*smax,tmax);                  % time horizon large enough

f = zeros(nU,nU);                         % evaluate the cost function
for i=1:nU
    u1 = (i-1)*Umax/(nU-1);            % u1 between 0 and Umax
    for j=1:nU
        u2 = (j-1)*Umax/(nU-1);    % u2 between 0 and Umax

                % determine switching time for cases
        r1 = (u1 ~= Rq).* (1 - u1/Rq);
        r2 = (u2 ~= Rq).* (1 - u2/Rq);
        if ((N*r1-1)*r2 == (r1-r2))
            s = 0;
        elseif (r1*(N*r2-1)*(r1-r2) > 0)
            Q = (N*r1-1)*r2/(r1-r2);
            s = log(Q)/(R*r1);
        else
            s = tmax;
        end

        if (s == 0)                  % if s=0 then only 2nd integral F2
            nT2 = nT;
            t2 = linspace(0,tmax,nT2);
            dt2 = t2(2)-t2(1);
            x2 = K*r2*ones(size(t2));
            F2 = u2*exp(-d*t2).* (pq*x2-c);
        end
    end
end

```

```

f(i,j) = sum((F2(2:nT2)+F2(1:nT2-1)))*dt2/2;
end

if (s == tmax) % if s=tmax then only 1st integral F1
    nT1 = nT;
    t1 = linspace(0,tmax,nT1);
    dt1 = t1(2)-t1(1);
    x1 = K*r1./(1 + (N*r1-1)*exp(-r1*R*t1));
    F1 = u1*exp(-d*t1).*(pq*x1-c);
    f(i,j) = sum((F1(2:nT1)+F1(1:nT1-1)))*dt1/2;
end

% if 0 < s < tmax then both integrals
if ((s > 0) && (s < tmax))
    nT1 = max(2,round(nT*s/tmax));
    t1 = linspace(0,s,nT1);
    dt1 = t1(2)-t1(1);
    x1 = K*r1./(1 + (N*r1-1)*exp(-r1*R*t1));
    F1 = u1*exp(-d*t1).*(pq*x1-c);
    nT2 = nT - nT1;
    t2 = linspace(s,tmax,nT2);
    dt2 = t2(2)-t2(1);
    x2 = K*r2*ones(size(t2));
    F2 = u2*exp(-d*t2).*(pq*x2-c);
    f(i,j) = sum((F1(2:nT1)+F1(1:nT1-1)))*dt1/2 ...
        + sum((F2(2:nT2)+F2(1:nT2-1)))*dt2/2;
end

end
end

fmax = max(f(:)); % store the optimum
fmin = min(f(:));
i = find(f(:) == fmax);
u1 = linspace(0,Umax,nU); u1 = kron(u1',ones(1,nU));
u2 = linspace(0,Umax,nU); u2 = kron(ones(nU,1),u2);

subplot(1,2,1) % plot the cost function
surf(u1,u2,f)
axis tight
hold on;
plot3(u1(i),u2(i),fmax*ones(size(i)), 'r*')
plot3(u1(i),u2(i),fmin*ones(size(i)), 'r*')
hold off;
df = fmax-fmin;
text(u1(i(1)),u2(i(1)),fmax + 0.1*df, ...
    sprintf('f=%0.1e',fmax), 'Color', 'red')
text(u1(i(1)),u2(i(1)),fmin + 0.1*df, ...
    sprintf('U1=%0.1e\nU2=%0.1e',u1(i(1)),u2(i(1))), 'Color', 'red')

```

```

xlabel('U1')
ylabel('U2')
zlabel('f(U1,U2)')
title('Langfristiger Profit')

disp(sprintf(' '))
% summarize results
if (N*(1-Uopt/Rq)>1)
    disp(sprintf(' N*(1 - q*Uopt/R) = %0.0e > 1',N*(1-Uopt/Rq)))
    disp(sprintf(' expected optimal: U1 = %0.3e, U2* = %0.3e',0,Uopt))
else
    disp(sprintf(' N*(1 - q*Uopt/R) = %0.0e < 1',N*(1-Uopt/Rq)))
    disp(sprintf(' expected optimal: U1 = %0.3f, U2* = %0.3e',Umax,Uopt))
end
disp(sprintf('f=fmax=%0.3e\nin U1 = %0.3e, U2 = %0.3e', ...
    fmax,u1(i(1)),u2(i(1))))
disp(sprintf(' '))

u1 = u1(i(1)); % determine optimal switching time
u2 = u2(i(1));
r1 = (u1 ~= Rq).* (1 - u1/Rq);
r2 = (u2 ~= Rq).* (1 - u2/Rq);
if ((N*r1-1)*r2 == (r1-r2))
    s = 0;
elseif (r1*(N*r2-1)*(r1-r2) > 0)
    Q = (N*r1-1)*r2/(r1-r2);
    s = log(Q)/(R*r1);
else
    s = tmax;
end

if (s == 0) % if s=0 then x = equilibrium
    t1 = [];
    x1 = [];
    nT2 = nT;
    t2 = linspace(0,tmax,nT2);
    dt2 = t2(2)-t2(1);
    x2 = K*r2*ones(size(t2));
end

if (s == tmax) % if s=tmax then x=logistic
    nT1 = nT;
    t1 = linspace(0,tmax,nT1);
    dt1 = t1(2)-t1(1);
    x1 = K*r1./(1 + (N*r1-1)*exp(-r1*R*t1));
    t2 = [];
    x2 = [];
end

```

```

if ((s > 0) && (s < tmax))           % if 0 < s < tmax then switch
    nT1 = max(2,round(nT*s/tmax));
    t1 = linspace(0,s,nT1);
    dt1 = t1(2)-t1(1);
    x1 = K*r1./(1 + (N*r1-1)*exp(-r1*R*t1));
    nT2 = nT - nT1;
    t2 = linspace(s,tmax,nT2);
    dt2 = t2(2)-t2(1);
    x2 = K*r2*ones(size(t2));
end

t = [t1,t2];                         % fish population without control
x = K./(1 + (N-1)*exp(-R*t));

subplot(1,2,2)                         % plot fish population
if (isempty(x1))
    xmax = 1.1*max([x2(:);x(:)]);
    plot(t,x,'r:',t2,x2,'b')
    legend('ohne','nachher','Location','best')
elseif (isempty(x2))
    xmax = 1.1*max([x1(:);x(:)]);
    plot(t,x,'r:',t1,x1,'g')
    legend('ohne','vorher','Location','best')
else
    xmax = 1.1*max([x1(:);x2(:);x(:)]);
    plot(t,x,'r:',t1,x1,'g',t2,x2,'b')
    legend('ohne','vorher','nachher','Location','best')
end
axis([0 tmax 0 xmax])
xlabel('t')
ylabel('x')
title('Fisch Population')

```

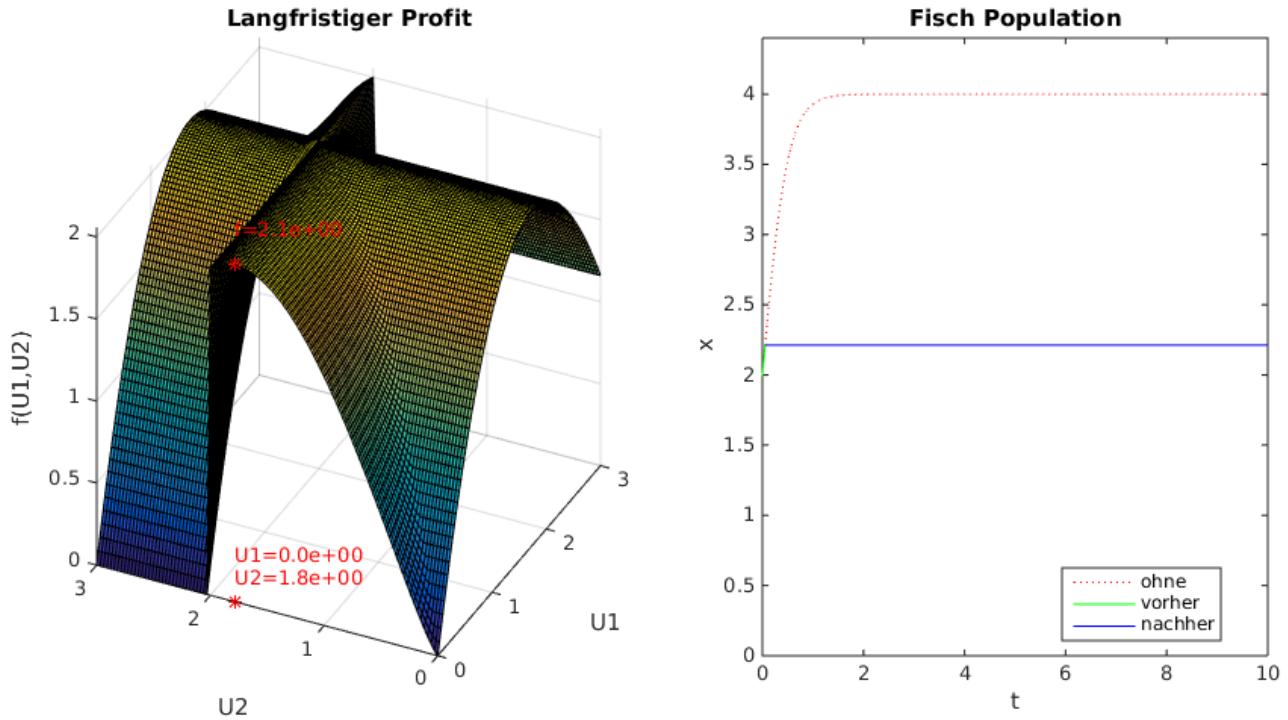
Die Ergebnisse werden folgendmaßen grafisch dargestellt.

Im ersten Fall gelten

```

N = 2;
N*(1 - q*U2opt/R) = 1e+00 > 1
expected optimal: U1 = 0.000e+00, U2* = 1.775e+00
f = fmax = 2.055e+00
in U1 = 0.000e+00, U2 = 1.788e+00

```



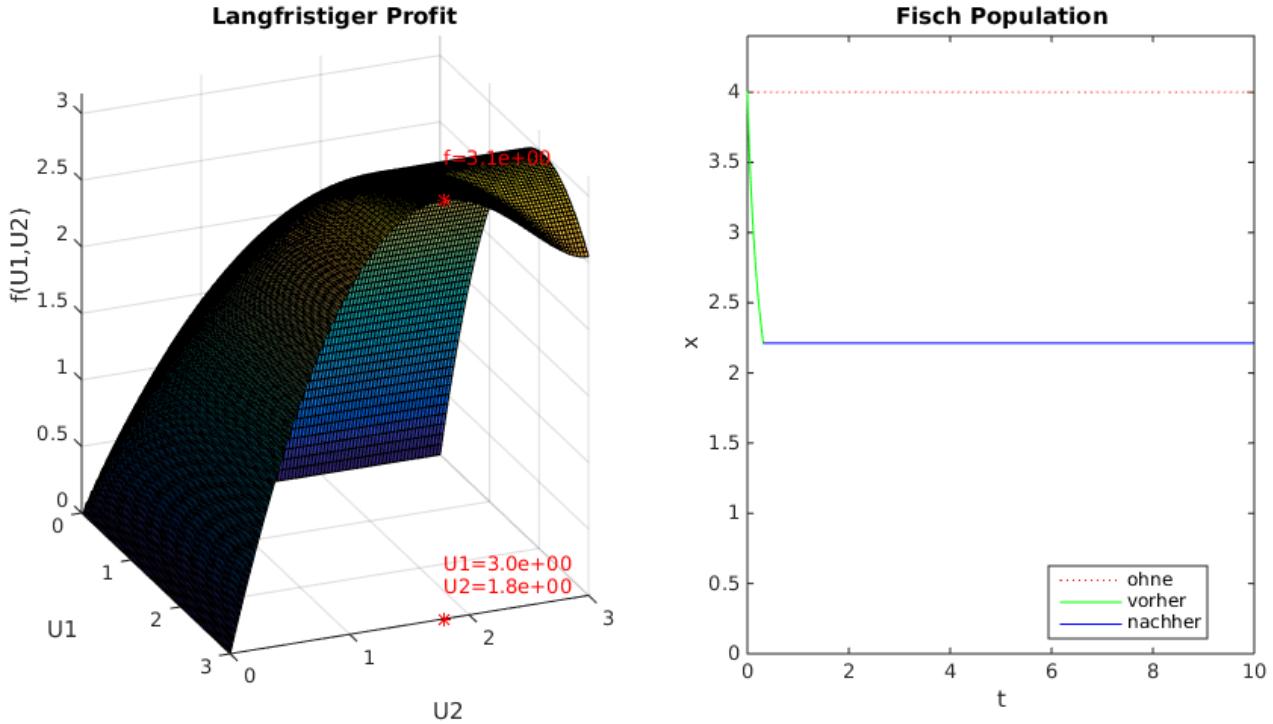
und das berechnete Optimum stimmt mit dem theoretischen überein, wie man an der rot markierten Stelle links sieht. Für $N > 1$ ist die Fischpopulation anfänglich unterhalb der Kapazität, wie man mit der roten Kurve (ohne Steuerung) rechts sieht. Die Fischpopulation steigt logistisch bis zu der Schaltzeit $s = 0.0532$, und dann bleibt sie dem Zustand $x = 2.2121$, während die optimale Steuerung $U2 = 1.788 \approx U_2^*$ eingesetzt wird.

Im nächsten Fall gelten

```

N = 1
N*(1 - q*U2opt/R) = 6e-01 < 1
expected optimal: U1 = 3.000, U2* = 1.775e+00
f = fmax = 3.145e+00
in U1 = 3.000e+00, U2 = 1.788e+00

```



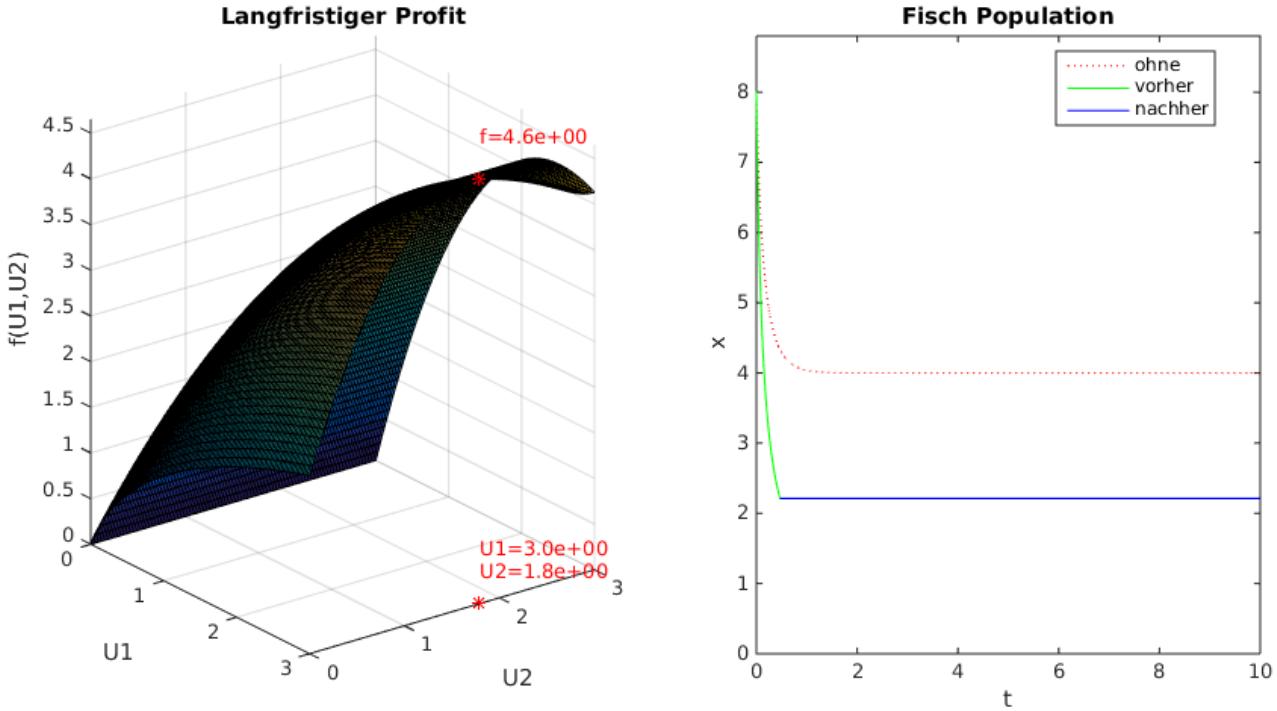
und das berechnete Optimum stimmt mit dem theoretischen überein, wie man an der rot markierten Stelle links sieht. Für $N = 1$ ist die Fischpopulation anfänglich bei der Kapazität, wie man mit der roten Kurve (ohne Steuerung) rechts sieht. Die Fischpopulation fällt von diesem Zustand, während die Steuerung $U_1 = U_{1\max}$ eingesetzt wird, und zu der Schaltzeit $s = 0.3139$ bleibt sie beim Zustand $x = 2.2121$, während die optimale Steuerung $U_2 = 1.788 \approx U_2^*$ eingesetzt wird.

Im letzten Fall gelten

```

N = 0.5;
N*(1 - q*U2opt/R) = 3e-01 < 1
expected optimal: U1 = 3.000, U2* = 1.775e+00
f = fmax = 4.645e+00
in U1 = 3.000e+00, U2 = 1.788e+00

```



und das berechnete Optimum stimmt mit dem theoretischen überein, wie man an der rot markierten Stelle links sieht. Für $N < 1$ ist die Fischpopulation anfänglich oberhalb der Kapazität, wie man mit der roten Kurve (ohne Steuerung) rechts sieht. Die Fischpopulation fällt von diesem Zustand, während die Steuerung $U_1 = U_{1\max}$ eingesetzt wird, und zu der Schaltzeit $s = 0.4680$ bleibt sie beim Zustand $x = 2.2121$, während die optimale Steuerung $U_2 = 1.788 \approx U_2^*$ eingesetzt wird.

* Teil (b) Allgemeine Steuerung, Numerische Rechnungen

Das Kostenfunktional wird allgemeiner für eine beliebige Steuerungsfunktion u mit dem folgenden Matlab Code maximiert.

```
function B08Bsp2b

global c d K N p q R si tmax ti nU nT sopt Uopt Us Umax xopt Jv

% min int_0^T exp(-d*t) [p q x(t) - c] u(t) dt

h1 = figure(1); close(h1); h1 = figure(1);
set(h1,'Position',[10 10 1600 400]);

c = 1; % costs per boat
p = 1; % price per fish
q = 1; % fish catching rate per boat
R = 4; % 1 / time scale for fish logistic growth
K = 4; % fish capacity
d = 1; % continuous interest rate
```

```

N = 10; % initially K/N fish

Rq = R/q; % stored parameters
pq = p*q;
pqK = pq*K;
cpqK = c/pqK;
dR = d/R;

Umax = Rq*(1 - cpqK); % 0 < U < Umax
Uopt = (Rq/4)*(3 - cpqK + dR - sqrt((1 + cpqK - dR)^2 + 8*dR*cpqK));

% expected optimum:
% (U1,U2) = (0,Uopt) if N(1 - q*Uopt/R) > 1
% (U1,U2) = (Umax,Uopt) if N(1 - q*Uopt/R) < 1

if (N*(1-Uopt/Rq)>1)
    U1 = 0;
    U2 = Uopt; % optimal control in this case
else
    U1 = Umax;
    U2 = Uopt; % optimal control in this case
end
r1 = 1 - U1/Rq;
r2 = 1 - U2/Rq;
Q = (N*r1-1)*r2/(r1-r2);
sopt = log(Q)/(R*r1); % optimal switching time

tmax = 10; % time horizon
si = [0,sopt/2,linspace(sopt,tmax,3)]; % switching times
nU = length(si)-1;
si(nU+1) = 1.01*si(nU+1); % si(nU+1)>tmax easier for u-evaluation
nT = 1000;
ti = linspace(0,tmax,nT+1)'; % evaluation times for profit

Us = U1*((si(1:nU)+si(2:(nU+1)))/2 < sopt)' ...
    + U2*((si(1:nU)+si(2:(nU+1)))/2 >= sopt)';
xopt = fish(Us); % fish for the known optimal control
Jv = [];
Jv = -profit(Us); % cost for the known optimal control

U0 = Umax*ones(nU,1); % worse case start
lb = zeros(nU,1); % lower bound
ub = Umax*ones(nU,1); % upper bound
opts = optimoptions('fmincon', ...
    'MaxIter',500,'TolFun',1.0e-15,'TolX',1.0e-15);
U = fmincon(@profit,U0,[],[],[],[],lb,ub);

end

```

```

function J = profit(U)

global c d K N p q R si tmax ti nU nT sopt Uopt Us Umax xopt Jv

xi = fish(U); % population xi and profit Pexp
Pexp = exp(-d*ti).* (p*q*xi-c).*u(ti,U);

subplot(1,4,1) % plot profit
plot(ti,Pexp,'b',ti,zeros(nT+1,1),'r:')
xlabel('t')
ylabel('exp(-td) P(t)')
title('Profit')
drawnow;

dt = ti(2)-ti(1); % integral of Pexp bei midpoint rule
J = sum((Pexp(1:nT)+Pexp(2:(nT+1)))*dt/2);
Jv = [Jv,J];
J = -J; % max(J) = min(-J)

subplot(1,4,2) % plot history of J-evaluations
Jn = length(Jv); it = 1:Jn;
plot(it,Jv,it,Jv(1)*ones(1,Jn),'g:')
title('Zielfunktion')
xlabel('Iterationen')
ylabel('J(u)')
drawnow;

subplot(1,4,3) % plot fish population
plot(ti,xopt,'b',ti,xi,'r',ti,K*ones(nT+1,1),'g:')
xlabel('t')
ylabel('x(t)')
title('Fish Population')
axis([0 tmax 0 1.1*K])
drawnow;

subplot(1,4,4) % plot controls
plot(ti,u(ti,Us),'b',ti,u(ti,U),'r',ti,Umax*ones(nT+1,1),'g:')
xlabel('t')
ylabel('u(t)')
title('Boote')
axis([0 tmax 0 1.1*Umax])
drawnow;

end

function xi = fish(U)

```

```

global c d K N p q R si tmax ti nU nT sopt Uopt Us Umax xopt Jv

% solve x' = R x(1 - x/K) - q u x
opts = odeset('RelTol',1.0e-10);
[t,x] = ode45(@(t,x) R*x.* (1 - x/K) - q*u(t,U).*x,[0,tmax],K/N,opts);
xi = interp1(t,x,ti);

end

function boats = u(t,U)

global c d K N p q R si tmax ti nU nT sopt Uopt Us Umax xopt Jv

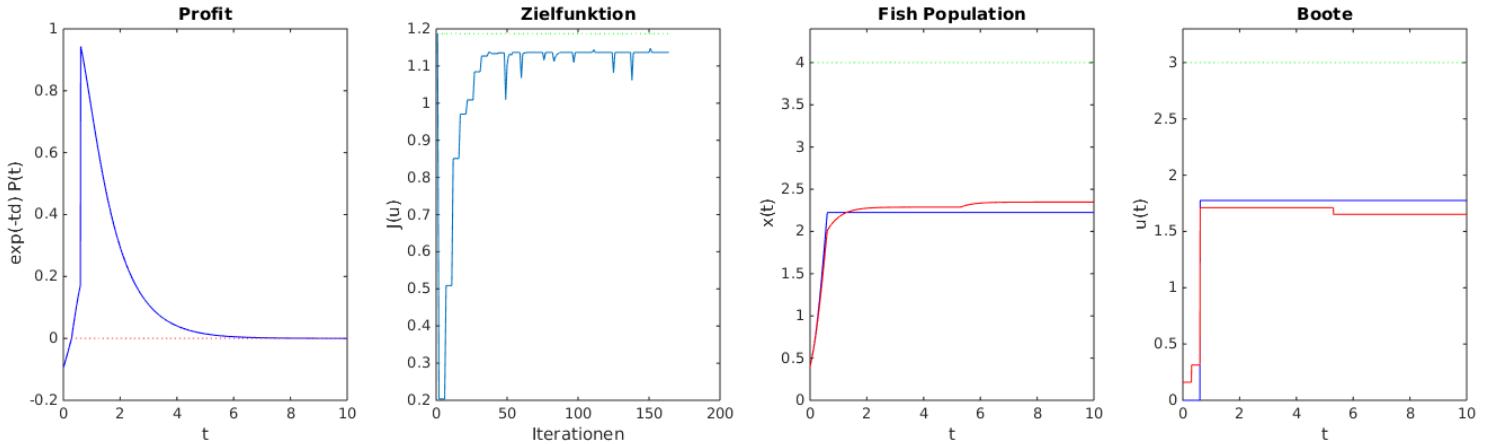
% evaluate sum_i U(i) chi_i(t), chi_i(t) = 1, s(i) < t < s(i+1)

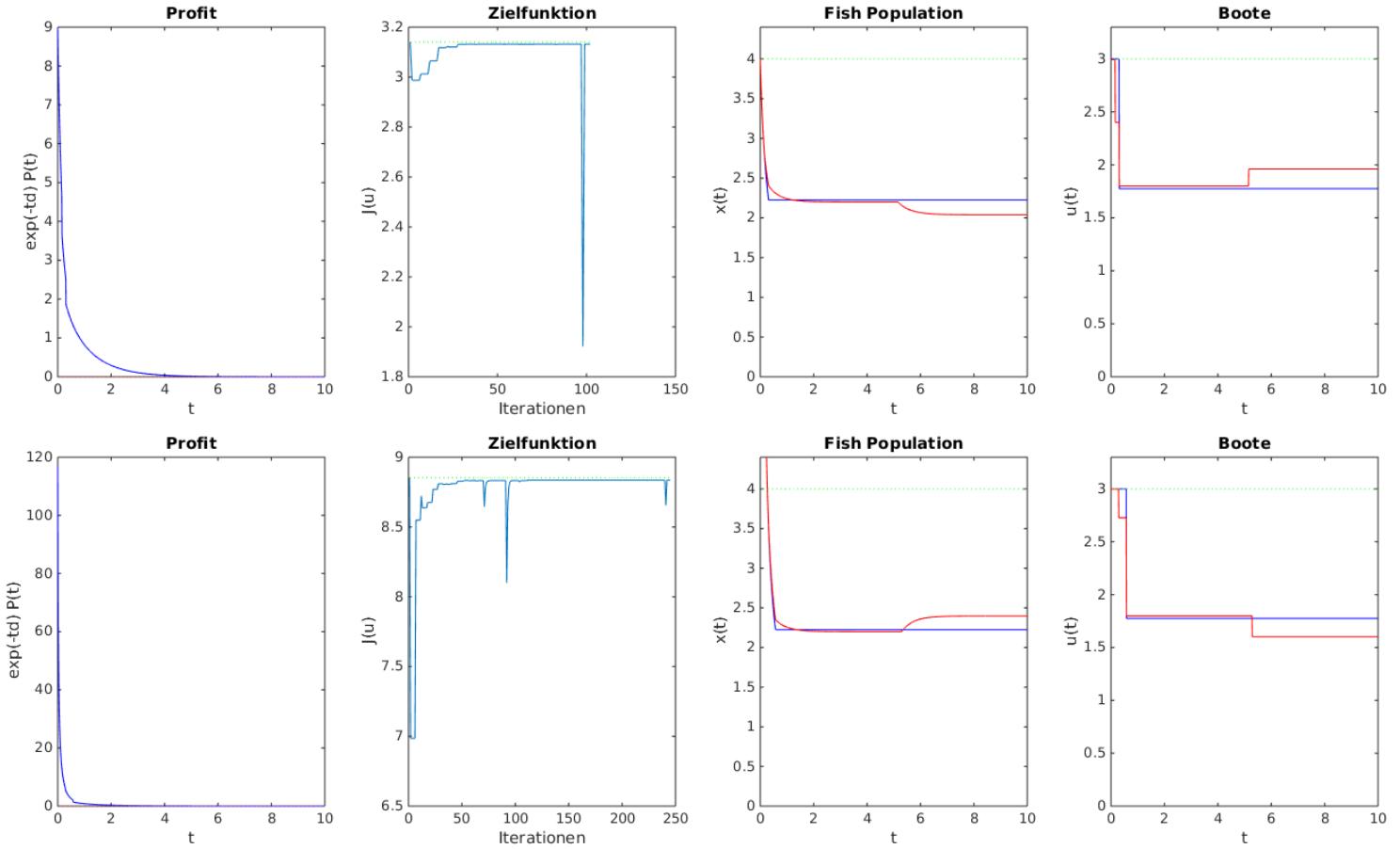
n = length(t);
A1 = kron(ones(n,1),si(1:nU));
A2 = kron(ones(n,1),si(2:(nU+1)));
At = kron(t,ones(1,nU));
A = (A1 <= At) & (At < A2);
boats = A*abs(U);

end

```

Die Ergebnisse für die Fälle $N = 10$, $N = 1$ bzw. $N = 1/10$ werden folgendermaßen grafisch dargestellt. Gegenüber ihren Gegenstücken der vorherigen Rechnungen sehen diese ähnlich aus.





* Teil (c) Allgemeine Steuerung, Variationelle Rechnungen

Man sucht einen stationären Zustand des Lagrangeschen Funktionalss,

$$L(u, \lambda, \mu) = \int_0^\infty e^{-\delta t} [p q x(t) - c u(t)] dt - \int_0^\infty \lambda(t) u(t) dt - \int_0^\infty \mu(t) [u_{\max} - u(t)] dt$$

unter den Komplementaritätsbedingungen,

$$\lambda(t) u(t) = 0, \quad \mu(t) [u_{\max} - u(t)] = 0, \quad \forall t > 0.$$

Die variationelle Ableitung nach u an der Stelle u mit Störung v ist,

$$\frac{\partial L}{\partial u}(u, \lambda, \mu; v) = \lim_{\epsilon \rightarrow 0} D_\epsilon L(u + \epsilon v, \lambda, \mu) = \int_0^\infty \left\{ e^{-\delta t} \left[p q \frac{\partial x}{\partial u}(t, u; v) - c v(t) \right] - v(t) (\lambda(t) - \mu(t)) \right\} dt.$$

Die variationelle Ableitung $\partial x / \partial u(t, u; v)$ ist gegeben durch

$$\begin{aligned} 0 &= \lim_{\epsilon \rightarrow 0} D_\epsilon [-D_t x(t, u + \epsilon v) + R x(t, u + \epsilon v) [1 - x(t, u + \epsilon v)/K] - q[u(t) + \epsilon v(t)] x(t, u + \epsilon v)] \\ &= -D_t \frac{\partial x}{\partial u}(t, u; v) + R \frac{\partial x}{\partial u}(t, u; v) - 2 \frac{R}{K} \frac{\partial x}{\partial u}(t, u; v) x(t, u) - q x(t, u) v(t) - q u(t) \frac{\partial x}{\partial u}(t, u; v). \end{aligned}$$

Da die Anfangsbedingung $x(0) = K/N$ von u nicht abhängt, wird $\partial x / \partial u(t, u; v)$ durch die Lösung des Anfangswertproblems bestimmt,

$$\begin{cases} D_t \frac{\partial x}{\partial u}(t, u; v) = \frac{\partial x}{\partial u}(t, u; v) \left[R - 2 \frac{R}{K} x(t, u) - qu(t) \right] - qx(t, u)v(t), & t > 0 \\ \frac{\partial x}{\partial u}(0, u; v) = 0. \end{cases}$$

Die optimale Steuerung, die unter den Bedingungen

$$x(0) \ll K, \quad \lim_{t \rightarrow \infty} P(t) \lim_{t \rightarrow \infty} [pqx(t, u(t)) - c] > 0$$

im Lehrbuch berichtet wird, ist

$$u^*(t) = \begin{cases} 0, & 0 \leq t < s^* \\ U^*, & s^* \leq t < \infty \end{cases}$$

wobei

$$U^* = \frac{R}{4q} \left[3 - \frac{c}{pqK} + \frac{\delta}{R} - \sqrt{\left(1 + \frac{c}{pqK} - \frac{\delta}{R}\right)^2 + \frac{8\delta}{R} \frac{c}{pqK}} \right]$$

$$s^* = \frac{1}{R} \ln \left[(N-1) \left(\frac{R}{qU^*} - 1 \right) \right]$$

und

$$x(t) = \begin{cases} K/[1 + (N-1)e^{-Rt}], & 0 \leq t \leq s^* \\ K(1 - qU^*/R), & s^* \leq t < \infty. \end{cases}$$

Für die Lösung des Anfangswertproblems für $\partial x / \partial u$ seien die folgenden Funktionen definiert,

$$-F(t) = R - 2 \frac{R}{K} x(t, u^*) - qu^*(t) = R \frac{(N-1)e^{-Rt} - 1}{(N-1)e^{-Rt} + 1}, \quad 0 \leq t \leq s^*$$

und

$$-G(t) = R - 2 \frac{R}{K} x(t, u^*) - qu^*(t) = qU^* - R, \quad s^* \leq t < \infty$$

mit

$$\int_s^t F(s) ds = -R(t-s) + \ln \left(\frac{N-1+e^{Rt}}{N-1+e^{Rs}} \right)^2, \quad 0 \leq s, t \leq s^*$$

und

$$\int_s^t G(s) ds = (t-s)(R - qU^*), \quad s^* \leq s, t < \infty.$$

Die Ableitung $\partial x / \partial u$ erfüllt

$$\begin{cases} D_t \frac{\partial x}{\partial u}(t, u^*, v) = -\frac{\partial x}{\partial u}(t, u^*, v) F(t) - \frac{qK}{2R} (F(t) + R)v(t) & 0 \leq t \leq s^* \\ D_t \frac{\partial x}{\partial u}(t, u^*, v) = -\frac{\partial x}{\partial u}(t, u^*, v) G(t) - \frac{qK}{R} G(t)v(t) & s^* \leq t < \infty \\ D_t \frac{\partial x}{\partial u}(0, u^*, v) = 0 \end{cases}$$

Zuerst wird der Teil des Anfangswertproblems für $0 \leq t \leq s^*$ bearbeitet. Es gelten

$$D_s \left[\frac{\partial x}{\partial u}(s, u^*, v) \exp \left(\int_0^s F(r) dr \right) \right] = -\frac{qK}{2R} (F(s) + R) v(s) \exp \left(\int_0^s F(r) dr \right), \quad 0 \leq s \leq s^*$$

und

$$\frac{\partial x}{\partial u}(t, u^*, v) = -\frac{qK}{2R} \int_0^t (F(s) + R) v(s) \exp \left(-\int_s^t F(r) dr \right) ds, \quad 0 \leq t \leq s^*$$

Mit $v(t) = \delta_\tau(t) = \delta(t - \tau)$, $0 < \tau < s^*$, gilt

$$\begin{aligned} \frac{\partial x}{\partial u}(t, u^*, \delta_\tau) &= \begin{cases} 0, & 0 \leq t < \tau \\ -\frac{qK}{2R} (F(\tau) + R) \exp \left(-\int_\tau^t F(r) dr \right), & \tau \leq t \leq s^* \\ \exp \left(-\int_{s^*}^t G(s) ds \right) \frac{\partial x}{\partial u}(s^*, u^*, \delta_\tau), & s^* \leq t < \infty \end{cases} \\ &= \begin{cases} 0, & 0 \leq t < \tau \\ -qK e^{-R(t-\tau)} \frac{1 + (N-1)e^{-R\tau}}{(1 + (N-1)e^{-Rt})^2}, & \tau \leq t \leq s^* \\ -e^{-(R-qU^*)(t-s^*)} qK e^{-R(s^*-\tau)} \frac{1 + (N-1)e^{-R\tau}}{(1 + (N-1)e^{-Rs^*})^2}, & s^* \leq t < \infty \end{cases} \end{aligned}$$

Da $u^*(t) = 0$ für $0 \leq t < s^*$ gilt, müssen $\lambda(t) > 0$ und $\mu(t) = 0$ für $0 \leq t < s^*$ gelten. Das Lagrangesche Funktional erfüllt

$$\begin{aligned} \frac{\partial L}{\partial u}(u^*, \lambda, 0; \delta_\tau) &= \int_0^\infty \left\{ e^{-\delta t} \left[pq \frac{\partial x}{\partial u}(t, u; \delta_\tau) - c\delta(t - \tau) \right] - \delta(t - \tau) \lambda(\tau) \right\} dt \\ &= pq \int_\tau^\infty e^{-\delta t} \frac{\partial x}{\partial u}(t, u; \delta_\tau) dt - ce^{-\delta\tau} - \lambda(\tau). \end{aligned}$$

Daher mit der Stationaritätsbedingungen folgt

$$\begin{aligned} \lambda(\tau) &= pq^2 K (N-1) (1 + (N-1)e^{-R\tau}) \int_\tau^{s^*} \frac{e^{-(R+\delta)t}}{(1 + (N-1)e^{-Rt})^2} dt \\ &+ \frac{pq^2 K (N-1) (1 + (N-1)e^{-R\tau})}{(1 + (N-1)e^{-Rs^*})^2} \frac{e^{-(R+\delta)s^*}}{R - qU^* + \delta} - ce^{-\delta\tau}, \quad 0 \leq \tau < s^*. \end{aligned}$$

Nun wird der Teil des Anfangswertproblems für $s^* \leq t < \infty$ bearbeitet. Es gelten

$$D_s \left[\frac{\partial x}{\partial u}(s, u^*, v) \exp \left(\int_{s^*}^s G(r) dr \right) \right] = -\frac{qK}{R} G(s) v(s) \exp \left(\int_{s^*}^s G(r) dr \right), \quad s^* \leq s < \infty$$

und

$$\begin{aligned} \frac{\partial x}{\partial u}(t, u^*, v) &= \exp \left(-\int_{s^*}^t G(s) ds \right) \frac{\partial x}{\partial u}(s^*, u^*, v) \\ &- \frac{qK}{R} \int_{s^*}^t G(s) v(s) \exp \left(-\int_s^t G(r) dr \right) ds, \quad s^* \leq t < \infty. \end{aligned}$$

Mit $v(t) = \delta_\tau(t) = \delta(t - \tau)$, $s^* \leq \tau$, gilt

$$\begin{aligned} \frac{\partial x}{\partial u}(t, u^*, \delta_\tau) &= \begin{cases} 0, & 0 \leq t < \tau \\ -\frac{qK}{R} G(\tau) \exp\left(-\int_\tau^t G(r) dr\right), & \tau \leq t < \infty \end{cases} \\ &= \begin{cases} 0, & 0 \leq t < \tau \\ -\frac{qK}{R} (R - qU^*) e^{-(R - qU^*)(t - \tau)}, & \tau \leq t < \infty \end{cases} \end{aligned}$$

Da $0 < u^*(t) < u_{\max}$ für $0 \leq t < s^*$ gilt, müssen $\lambda(t) = 0$ und $\mu(t) = 0$ für $0 \leq t < s^*$ gelten. Das Lagrangesche Funktional erfüllt

$$\begin{aligned} \frac{\partial L}{\partial u}(u^*, 0, 0; \delta_\tau) &= \int_0^\infty e^{-\delta t} \left[pq \frac{\partial x}{\partial u}(t, u; \delta_\tau) - c\delta(t - \tau) \right] dt \\ &= -\frac{pq^2 K}{R} (R - qU^*) e^{(R - qU^*)\tau} \int_\tau^\infty e^{-(R - qU^* + \delta)t} dt - ce^{-\delta\tau} \\ &= -\left[\frac{pq^2 K}{R} \frac{R - qU^*}{R - qU^* + \delta} + c \right] e^{-\delta\tau} \end{aligned}$$

aber dies ist nicht Null...?

$$\begin{cases} x'(t) &= Rx(t)[1 - x(t)/K] - qu(t)x(t) & t > 0 \\ x(0) &= K/N \end{cases}$$

$$\begin{cases} \frac{x'(t)}{x(t)^2} &= \frac{R}{x(t)} - \frac{R}{K} - q \frac{u(t)}{x(t)} & t > 0 \\ \frac{1}{x(0)} &= \frac{N}{K} \end{cases}$$

$z(t) = 1/x(t)$,

$$\begin{cases} z'(t) &= (qu(t) - R)z(t) + R/K & t > 0 \\ z(0) &= N/K \end{cases}$$

$$D_s \left[z(s) \exp \left(\int_0^s [R - qu(r)] dr \right) \right] = \frac{R}{K} \exp \left(\int_0^s [R - qu(r)] dr \right)$$

$$z(t) \exp \left(\int_0^t [R - qu(r)] dr \right) - z(0) = \frac{R}{K} \int_0^t \exp \left(\int_0^s [R - qu(r)] dr \right) ds$$

$$z(t) = \frac{N}{K} \exp \left(- \int_0^t [R - qu(r)] dr \right) + \frac{R}{K} \int_0^t \exp \left(- \int_s^t [R - qu(r)] dr \right) ds$$

$$z(t) = \frac{N}{K} e^{-Rt} \exp \left(q \int_0^t u(r) dr \right) + \frac{R}{K} \int_0^t e^{-R(t-s)} \exp \left(q \int_s^t u(r) dr \right) ds$$

$$U(t) = \int_0^t u(r) dr, \quad V(t) = \int_0^t v(r) dr$$

$$z(t) = \frac{e^{-Rt+qU(t)}}{K} \left[N + R \int_0^t e^{Rs-qU(s)} ds \right]$$

$$x(t, u) = Ke^{Rt-qU(t)} / \left[N + R \int_0^t e^{Rs-qU(s)} ds \right]$$

Probe: Mit $u(t) = u^*(t)$, $U(t) = U^*(t - s^*)\theta(t - s^*) = U^*(t - s^*)_+$,

$$\begin{aligned}
x(t, u^*) &= \frac{Ke^{Rt-qU^*(t-s^*)_+}}{N + R \int_0^t e^{Rs-qU^*(s-s^*)_+} ds} = \frac{K \left[\theta(s^* - t) e^{Rt} + \theta(t - s^*) e^{Rt-qU^*(t-s^*)} \right]}{N + R \int_0^t \left[\theta(s^* - s) e^{Rs} + \theta(s - s^*) e^{Rs-qU^*(s-s^*)} \right] ds} \\
&= \frac{K \left[\theta(s^* - t) e^{Rt} + \theta(t - s^*) e^{Rt-qU^*(t-s^*)} \right]}{N + R\theta(s^* - t) \left[\int_0^t e^{Rs} ds \right] + R\theta(t - s^*) \left[\int_0^{s^*} e^{Rs} + \int_{s^*}^t e^{Rs-qU^*(s-s^*)} ds \right]} \\
&= \frac{K \left[\theta(s^* - t) e^{Rt} + \theta(t - s^*) e^{Rt-qU^*(t-s^*)} \right]}{N + \theta(s^* - t)(e^{Rt} - 1) + \theta(t - s^*) \left[(e^{Rs^*} - 1) + \frac{R}{R - qU^*} (e^{Rt-qU^*(t-s^*)} - e^{Rs^*}) \right]} \\
&= \frac{K \left[\theta(s^* - t) e^{Rt} + \theta(t - s^*) e^{Rt-qU^*(t-s^*)} \right]}{N + \theta(s^* - t)(e^{Rt} - 1) + \theta(t - s^*) \left[(e^{Rs^*} - 1) + (1 + (N - 1)e^{-Rs^*}) (e^{Rt-qU^*(t-s^*)} - e^{Rs^*}) \right]} \\
&= \frac{K \left[\theta(s^* - t) e^{Rt} + \theta(t - s^*) e^{Rt-qU^*(t-s^*)} \right]}{N + \theta(s^* - t)(e^{Rt} - 1) + \theta(t - s^*) \left[(N - 1 + e^{Rs^*}) e^{(R - qU^*)(t-s^*)} - N \right]} \\
&= \begin{cases} \frac{Ke^{Rt}}{N - 1 + e^{Rt}}, & 0 \leq t < s^* \\ \frac{K}{1 + (N - 1)e^{-Rt}}, & s^* \leq t < \infty \\ K(1 - qU^*/R), & s^* \leq t < \infty \end{cases}
\end{aligned}$$

Die variationelle Ableitung:

$$\begin{aligned}
\frac{\partial x}{\partial u}(t, u; v) &= \lim_{\epsilon \rightarrow 0} D_\epsilon x(t, u + \epsilon v) = \\
\lim_{\epsilon \rightarrow 0} D_\epsilon \frac{Ke^{Rt-q(U(t)+\epsilon V(t))}}{N + R \int_0^t e^{Rs-q(U(s)-\epsilon V(s))} ds} &= \lim_{\epsilon \rightarrow 0} \left\{ \frac{-qKV(t)e^{Rt-q(U(t)+\epsilon V(s))}}{N + R \int_0^t e^{Rs-q(U(s)+\epsilon V(s))} ds} \right. \\
&\quad \left. - \frac{Ke^{Rt-q(U(t)+\epsilon V(t))}}{\left[N + R \int_0^t e^{Rs-q(U(s)+\epsilon V(s))} ds \right]^2} R \int_0^t e^{Rs-q(U(s)+\epsilon V(s))} (-qV(s)) ds \right\} \\
&= \left\{ \frac{-qKV(t)e^{Rt-qU(t)}}{N + R \int_0^t e^{Rs-qU(s)} ds} + qRK e^{Rt-qU(t)} \frac{\int_0^t e^{Rs-qU(s)} V(s) ds}{\left[N + R \int_0^t e^{Rs-qU(s)} ds \right]^2} \right\}
\end{aligned}$$

$$= qx(t, u) \left\{ -V(t) + R \frac{\int_0^t e^{Rs-qU(s)} V(s) ds}{N + R \int_0^t e^{Rs-qU(s)} ds} \right\}$$

Mit $v(t) = \delta_\tau(t) = \delta(t - \tau)$, $\tau \geq 0$,

$$V(t) = \theta(t - \tau) = \begin{cases} 0, & t < \tau \\ 1, & \tau \leq t \end{cases}$$

und

$$\frac{\partial x}{\partial u}(t, u, \delta_\tau) = qx(t, u)\theta(t - \tau) \left\{ R \frac{\int_\tau^t e^{Rs-qU(s)} ds}{N + R \int_0^t e^{Rs-qU(s)} ds} - 1 \right\}$$

Mit $u(t) = u^*(t)$ und $0 \leq \tau, t \leq s^*$,

$$\begin{aligned} \frac{\partial x}{\partial u}(t, u^*, \delta_\tau) &= qx(t, u^*)\theta(t - \tau) \left\{ \frac{e^{Rt} - e^{R\tau}}{N + e^{Rt} - 1} - 1 \right\}, \quad 0 \leq \tau, t \leq s^* \\ &= \frac{qK}{1 + (N - 1)e^{-Rt}} \left\{ \frac{1 - e^{-R(t-\tau)} - (1 + (N - 1)e^{-Rt})}{1 + (N - 1)e^{-Rt}} \right\} \theta(t - \tau), \quad 0 \leq \tau, t \leq s^* \\ &= -qKe^{-R(t-\tau)} \frac{1 + (N - 1)e^{-R\tau}}{(1 + (N - 1)e^{-Rt})^2} \theta(t - \tau), \quad 0 \leq \tau, t \leq s^* \end{aligned}$$

Mit $u(t) = u^*(t)$ und $0 \leq \tau \leq s^* < t$,

$$\begin{aligned} \frac{\partial x}{\partial u}(t, u^*, \delta_\tau) &= qK(1 - qU^*/R) \left\{ \frac{R \int_\tau^{s^*} e^{Rs} ds + R \int_{s^*}^t e^{(R-qU^*)s} ds}{N + R \int_0^{s^*} e^{Rs} ds + R \int_{s^*}^t e^{(R-qU^*)s} ds} - 1 \right\}, \quad 0 \leq \tau \leq s^* < t \\ &= qK(1 - qU^*/R) \left\{ \frac{R \int_\tau^{s^*} e^{Rs} ds + R \int_{s^*}^t e^{(R-qU^*)s} ds - N - R \int_0^{s^*} e^{Rs} ds - R \int_{s^*}^t e^{(R-qU^*)s} ds}{N + R \int_0^{s^*} e^{Rs} ds + R \int_{s^*}^t e^{(R-qU^*)s} ds} \right\}, \quad 0 \leq \tau \leq s^* < t \\ &= -qK(1 - qU^*/R) \left\{ \frac{N + R \int_0^\tau e^{Rs} ds}{N + R \int_0^{s^*} e^{Rs} ds + R \int_{s^*}^t e^{(R-qU^*)s} ds} \right\}, \quad 0 \leq \tau \leq s^* < t \\ &= -qK(1 - qU^*/R) \left\{ \frac{N + e^{R\tau} - 1}{N + (e^{Rs^*} - 1) + \frac{1}{1 - qU^*/R} (e^{(R-qU^*)t} - e^{(R-qU^*)s^*})} \right\}, \quad 0 \leq \tau \leq s^* < t \end{aligned}$$

$$= -qK(1-qU^*/R) \left\{ \frac{N + e^{R\tau} - 1}{e^{Rs^*}(1 + (N-1)e^{-Rs^*}) + \frac{1}{1-qU^*/R} (e^{(R-qU^*)t} - e^{(R-qU^*)s^*})} \right\}, \quad 0 \leq \tau \leq s^* < t$$

$$\begin{aligned} &= -qK(1-qU^*/R) \left\{ \frac{N + e^{R\tau} - 1}{e^{Rs^*} \frac{1}{1-qU^*/R} + \frac{1}{1-qU^*/R} (e^{(R-qU^*)t} - e^{(R-qU^*)s^*})} \right\}, \quad 0 \leq \tau \leq s^* < t \\ &= -qK(1-qU^*/R) \left\{ \frac{(N + e^{R\tau} - 1)(1-qU^*/R)}{e^{Rs^*} + e^{(R-qU^*)t} - e^{(R-qU^*)s^*}} \right\}, \quad 0 \leq \tau \leq s^* < t \\ &= -qKe^{-R(s^*-\tau)} \left\{ \frac{(1 + (N-1)e^{-R\tau})(1-qU^*/R)^2}{1 + e^{-Rs^*}(e^{(R-qU^*)(t-s^*)} - 1)e^{(R-qU^*)s^*}} \right\}, \quad 0 \leq \tau \leq s^* < t \\ &= -qKe^{-R(s^*-\tau)} \left\{ \frac{(1 + (N-1)e^{-R\tau})(1-qU^*/R)^2}{1 + e^{-qU^*s^*}(e^{(R-qU^*)(t-s^*)} - 1)} \right\}, \quad 0 \leq \tau \leq s^* < t \end{aligned}$$

$$x(t, u) = Ke^{Rt-qU(t)} / \left[N + R \int_0^t e^{Rs-qU(s)} ds \right]$$

Die variationelle Ableitung:

$$\begin{aligned} \frac{\partial x}{\partial u}(t, u; v) &= \lim_{\epsilon \rightarrow 0} D_\epsilon x(t, u + \epsilon v) = \\ &\lim_{\epsilon \rightarrow 0} D_\epsilon \frac{Ke^{Rt-q(U(t)+\epsilon V(t))}}{N + R \int_0^t e^{Rs-q(U(s)-\epsilon V(s))} ds} = \lim_{\epsilon \rightarrow 0} \left\{ \frac{-qKV(t)e^{Rt-q(U(t)+\epsilon V(s))}}{N + R \int_0^t e^{Rs-q(U(s)+\epsilon V(s))} ds} \right. \\ &\quad \left. - \frac{Ke^{Rt-q(U(t)+\epsilon V(t))}}{\left[N + R \int_0^t e^{Rs-q(U(s)+\epsilon V(s))} ds \right]^2} R \int_0^t e^{Rs-q(U(s)+\epsilon V(s))} (-qV(s)) ds \right\} \\ &= \left\{ \frac{-qKV(t)e^{Rt-qU(t)}}{N + R \int_0^t e^{Rs-qU(s)} ds} + qRKe^{Rt-qU(t)} \frac{\int_0^t e^{Rs-qU(s)} V(s) ds}{\left[N + R \int_0^t e^{Rs-qU(s)} ds \right]^2} \right\} \\ &= qx(t, u) \left\{ -V(t) + R \frac{\int_0^t e^{Rs-qU(s)} V(s) ds}{N + R \int_0^t e^{Rs-qU(s)} ds} \right\} \\ &= qx(t, u) \left\{ -V(t) + \frac{Rx(t, u)}{Ke^{Rt-qU(t)}} \int_0^t e^{Rs-qU(s)} V(s) ds \right\} \end{aligned}$$

Mit $v(t) = \delta_\tau(t) = \delta(t - \tau)$, $\tau \geq 0$, folgt

$$V(t) = \theta(t - \tau) = \begin{cases} 0, & t < \tau \\ 1, & \tau \leq t \end{cases}$$

und

$$\frac{\partial x}{\partial u}(t, u, \delta_\tau) = qx(t, u)\theta(t - \tau) \left\{ \frac{Rx(t, u^*)}{Ke^{Rt-qU(t)}} \int_\tau^t e^{Rs-qU(s)} ds - 1 \right\}$$

Mit $u(t) = u^*(t)$ und $0 \leq \tau, t \leq s^*$, folgt

$$\begin{aligned} \frac{\partial x}{\partial u}(t, u, \delta_\tau) &= qx(t, u^*)\theta(t - \tau) \left\{ \frac{Rx(t, u^*)}{Ke^{Rt}} (e^{Rt} - e^{R\tau}) - 1 \right\}, \quad 0 \leq \tau, t \leq s^* \\ &= \frac{qK}{1 + (N-1)e^{-Rt}} \left\{ \frac{K(e^{Rt} - e^{R\tau})}{(1 + (N-1)e^{-Rt})Ke^{Rt}} - 1 \right\} \theta(t - \tau), \quad 0 \leq \tau, t \leq s^* \\ &= \frac{qK}{1 + (N-1)e^{-Rt}} \left\{ \frac{(1 - e^{-R(t-\tau)}) - (1 + (N-1)e^{-Rt})}{1 + (N-1)e^{-Rt}} \right\} \theta(t - \tau), \quad 0 \leq \tau, t \leq s^* \\ &= -qKe^{-R(t-\tau)} \frac{1 + (N-1)e^{-R\tau}}{(1 + (N-1)e^{-Rt})^2} \theta(t - \tau), \quad 0 \leq \tau, t \leq s^* \end{aligned}$$

Mit $u(t) = u^*(t)$ und $0 \leq \tau \leq s^* = t$,

$$\begin{aligned} \frac{\partial x}{\partial u}(s^*, u, \delta_\tau) &= qx(s^*, u)\theta(s^* - \tau) \left\{ \frac{Rx(s^*, u^*)}{Ke^{Rs^*-qU(s^*)}} \int_\tau^{s^*} e^{Rs-qU(s)} ds - 1 \right\} \\ &= qK(1 - qU^*/R) \left\{ \frac{RK(1 - qU^*/R)}{Ke^{(R-qU^*)s^*}} \int_\tau^{s^*} e^{Rs} ds - 1 \right\} \\ &= qK(1 - qU^*/R) \left\{ \frac{(1 - qU^*/R)}{e^{(R-qU^*)s^*}} (e^{Rs^*} - e^{R\tau}) - 1 \right\} \\ &= \frac{qK(1 - qU^*/R)^2}{e^{(R-qU^*)s^*}} \left\{ (e^{Rs^*} - e^{R\tau}) - \frac{e^{(R-qU^*)s^*}}{(1 - qU^*/R)} \right\} \\ &= \frac{-qKe^{-(R-qU^*)(s^*-\tau)}}{(1 + (N-1)e^{-Rs^*})^2} \left\{ \frac{e^{(R-qU^*)(s^*-\tau)}}{(1 - qU^*/R)} - e^{-(R-qU^*)\tau} (e^{Rs^*} - e^{R\tau}) \right\} \end{aligned}$$

Mit $u(t) = u^*(t)$ und $0 \leq \tau \leq s^* < t$,

$$\begin{aligned} \frac{\partial x}{\partial u}(t, u, \delta_\tau) &= qx(t, u)\theta(t - \tau) \left\{ \frac{Rx(t, u^*)}{Ke^{Rt-qU(t)}} \int_\tau^t e^{Rs-qU(s)} ds - 1 \right\} \\ &= qK(1 - qU^*/R) \left\{ \frac{RK(1 - qU^*/R)}{Ke^{(R-qU^*)t}} \left[\int_\tau^{s^*} e^{Rs} ds + \int_{s^*}^t e^{(R-qU^*)s} ds \right] - 1 \right\} \\ &= qK(1 - qU^*/R) \left\{ \frac{R(1 - qU^*/R)}{e^{(R-qU^*)t}} \left[\frac{e^{Rs^*} - e^{R\tau}}{R} + \frac{e^{(R-qU^*)t} - e^{(R-qU^*)s^*}}{R - qU^*} \right] - 1 \right\} \end{aligned}$$

$$\begin{aligned}
&= \frac{qK(1 - qU^*/R)^2}{e^{(R-qU^*)t}} \left\{ (e^{Rs^*} - e^{R\tau}) + \frac{e^{(R-qU^*)t} - e^{(R-qU^*)s^*}}{1 - qU^*/R} - \frac{e^{(R-qU^*)t}}{1 - qU^*/R} \right\} \\
&= \frac{qK(1 - qU^*/R)^2}{e^{(R-qU^*)t}} \left\{ (e^{Rs^*} - e^{R\tau}) - \frac{e^{(R-qU^*)s^*}}{1 - qU^*/R} \right\} \\
&= \frac{qK(1 - qU^*/R)^2}{e^{(R-qU^*)t}} \left\{ (e^{Rs^*} - e^{R\tau}) - e^{(R-qU^*)s^*} (1 + (N-1)e^{-Rs^*}) \right\} \\
&= \frac{-qKe^{-(R-qU^*)(t-s^*)}}{(1 + (N-1)e^{-Rs^*})^2} \left\{ (1 + (N-1)e^{-Rs^*}) - e^{-(R-qU^*)s^*} (e^{Rs^*} - e^{R\tau}) \right\}
\end{aligned}$$

Mit $u(t) = u^*(t)$ und $s^* < \tau < t$,

$$\begin{aligned}
\frac{\partial x}{\partial u}(t, u; v) &= qx(t, u) \left\{ -V(t) + \frac{Rx(t, u)}{Ke^{Rt-qU(t)}} \int_0^t e^{Rs-qU(s)} V(s) ds \right\} \\
&= qK(1 - qU^*/R) \left\{ \frac{RK(1 - qU^*/R)}{Ke^{(R-qU^*)t}} \left[\int_{\tau}^t e^{(R-qU^*)s} ds \right] - 1 \right\} \theta(t - \tau) \\
&= qK(1 - qU^*/R) \left\{ \frac{R(1 - qU^*/R)}{e^{(R-qU^*)t}} \left[\frac{e^{(R-qU^*)t} - e^{(R-qU^*)\tau}}{R - qU^*} \right] - 1 \right\} \theta(t - \tau) \\
&= qK(1 - qU^*/R) \left\{ e^{-(R-qU^*)t} \left[e^{(R-qU^*)t} - e^{(R-qU^*)\tau} \right] - 1 \right\} \theta(t - \tau) \\
&= qK(1 - qU^*/R) \left\{ 1 - e^{-(R-qU^*)(t-\tau)} - 1 \right\} \theta(t - \tau) \\
&= -qK(1 - qU^*/R) e^{-(R-qU^*)(t-\tau)} \theta(t - \tau)
\end{aligned}$$

$$\begin{aligned}
\frac{\partial L}{\partial u}(u^*, \lambda, \mu; \delta_{\tau}) &= \int_0^{\infty} \left\{ e^{-\delta t} \left[pq \frac{\partial x}{\partial u}(t, u^*; \delta_{\tau}) - c\delta_{\tau}(t) \right] - \delta_{\tau}(t)(\lambda(t) - \mu(t)) \right\} dt. \\
&= pq \int_0^{\infty} e^{-\delta t} \frac{\partial x}{\partial u}(t, u^*, \delta_{\tau}) dt - ce^{-\delta \tau}, \quad s^* < \tau \\
&= -pq^2 K(1 - qU^*/R) \int_{\tau}^{\infty} e^{-\delta t} e^{-(R-qU^*)(t-\tau)} dt - ce^{-\delta \tau}, \quad s^* < \tau \\
&= -pq^2 K(1 - qU^*/R) e^{(R-qU^*)\tau} \int_{\tau}^{\infty} e^{-(R-qU^*+\delta)t} dt - ce^{-\delta \tau}, \quad s^* < \tau \\
&\quad - pq^2 K(1 - qU^*/R) e^{(R-qU^*)\tau} \left[\frac{0 - e^{-(R-qU^*+\delta)\tau}}{-(R - qU^* + \delta)} \right] - ce^{-\delta \tau}, \quad s^* < \tau \\
&= - \left\{ \frac{pq^2 K(1 - qU^*/R)}{R - qU^* + \delta} + c \right\} e^{-\delta \tau}
\end{aligned}$$

- **Beispiel 3: Kooperation – Brunnenteilen**

- **Aufgabe**

Zwei Nachbarn müssen normalerweise einem Wasserversorger €25 pro Tag für den täglichen Verbrauch bezahlen. Um diese Kosten zu reduzieren, möchten sie einen Brunnen teilen. Wenn die Nachbarn nur den fairen Anteil in Anspruch nehmen, reduzieren sie beide ihre Kosten beim Wasserversorger auf €10 pro Tag. Erfahrung zeigt, wegen Wetterbedingungen und der Wirkung auf Grundwasser gibt für einen beliebigen Tag eine 25% Wahrscheinlichkeit, dass der Brunnen nicht verwendet werden kann, und Verwendbarkeit an einem Tag ist unabhängig von Verwendbarkeit an einem anderen Tag.

- (a) Angenommen ist es möglich, dass genau ein Nachbar den ganzen täglichen Bedarf in Anspruch nimmt. Dann erspart er sich alle Kosten beim Wasserversorger, und die üblichen Kosten für den anderen Nachbar werden nicht reduziert. Wenn dies an einem bestimmten Tag passiert, gibt es ab dem nächsten Tag eine tägliche Strafe von der Gemeinde für den überlaufenden Nachbarn in der Höhe von ϵs . Bestimmen Sie die Strafe s von der Gemeinde, mit der sich eine Kooperation zwischen den Nachbarn auszahlt.
- (b) Eines Tages will die Gemeinde keine Strafe mehr von den Nachbarn kassieren. Sie sorgt dafür, dass der Brunnen immer ausreichend gefüllt ist und zuverlässig verwendet werden kann. Zusätzlich wird ein Mechanismus eingerichtet, wobei ein Nachbar den fairen Anteil vom Brunnen bekommt, aber nur wenn der andere Nachbar vor Ort ein Pumpensystem für ihn betätigt. Für seine Mühe bekommt er von dem so begünstigten Nachbar €1 abgebucht. Bestimmen Sie das Gleichgewicht für dieses Spiel, auch wenn es für eine bekannte Anzahl von Tagen wiederholt wird.
- (c) Für das Spiel im Teil (b) sei für die beiden Nachbarn die Anzahl der fortlaufenden Tage unbekannt, an denen das Handeln stattfindet. Die Anzahl wird aber mit einer geometrischen Verteilung abgeschätzt. Bestimmen Sie, ob es sich für die Nachbarn auszahlt, dass sie kooperieren. Entwickeln Sie Strategien für eine Folge von Tagen, anstatt für einen einzigen beliebigen zufälligen Tag, um den eigenen Gewinn eines Nachbars für eine Tagefolge zu erhöhen.
- (d) Schreiben Sie einen Matlab Code, um das Verhalten dieser zwei Nachbarn zu simulieren. In der Simulation soll der eine Nachbar eine Strategie vom Teil (c) verwenden, und der andere Nachbar soll die sogenannte *tit-for-tat* (Axelrod) Strategie verwenden, wobei man am ersten Tag kooperiert, und nachher verhält man sich, wie der Gegner sich am vorigen Tag verhalten hat.

- **Lösung**

- * **Teil (a) Strafe**

Das Handeln beginnt nicht notwendigerweise an einem Tag, an dem die Verwendbarkeit des Brunnens gewiss ist, also ist die Wahrscheinlichkeit der Verwendbarkeit am ersten Tag gegeben durch $p = 1 - 0.25 = 0.75$. Da die Verwendbarkeit des Brunnens an einem Tag unabhängig ist von der Verwendbarkeit an einem anderen Tag, ist die Wahrscheinlichkeit der Verwendbarkeit am m ten Tag gegeben durch die Potenz p^m , $m \in \mathbb{N}$. Angenommen kooperieren die beiden Nachbarn über $m - 1$ fortlaufende Tage, und dann am m ten Tag überläuft ein Nachbar. Vor dem Tag bezahlt er $R = €10$ täglich für Wasser, an dem Tag des Überlaufens bezahlt er $T = €0$, und nach dem Tag bezahlt er $U = €(10 + s)$. Der

Erwartungswert seiner gesamten Kosten ist

$$\begin{aligned}
& R \sum_{n=1}^{m-1} p^n + Tp^m + U \sum_{n=m+1}^{\infty} p^n = R \frac{p - p^m}{1 - p} + Tp^m + U \frac{p^{m+1}}{1 - p} \\
& = 10 \frac{3/4 - p^m}{1 - 3/4} + 0 + (10 + s) \frac{(3/4)p^m}{1 - 3/4} = 10 \frac{3/4}{1 - 3/4} + [(10 + s)(3/4) - 10] \frac{p^m}{1 - 3/4} = 30 + \frac{3s - 10}{4} p^m.
\end{aligned}$$

Wenn er ununterbrochen kooperiert, ist der Erwartungswert seiner gesamten Kosten

$$R \sum_{n=1}^{\infty} p^n = R \frac{p}{1 - p} = 30.$$

Das Kooperieren zahlt sich aus, wenn

$$30 < 30 + \frac{3s - 10}{4} p^m \quad \text{oder} \quad s > \frac{10}{3}.$$

* Teil (b) Selbstständiger Verlauf

Mit der neuen Regelung von der Gemeinde, wobei die Nachbarn selbstständig handeln, lassen sich die Gewinne für ihre Strategien relativ zu den Default-Kosten in der folgenden Tabelle zusammenfassen.

		Nachbar 2	
		Gewinne (1, 2)	nachbar 2
Nachbar 1	pumpen	(14, 14)	(-1, 15)
	nicht	(15, -1)	(0, 0)

Sei M die Anzahl der fortlaufenden Tage an denen das Handeln zwischen den Nachbarn stattfindet. Weiters sei M den beiden Nachbarn bekannt. Angenommen kooperieren die beiden Nachbarn an den ersten $m - 1$ Tagen, und am m ten Tag überläuft Nachbar N_1 . Dann ab dem $(m + 1)$ ten Tag reagieren die beiden Nachbarn N_1 und N_2 auf das verletzte Vertrauen und kooperieren nicht mehr. Der gesamte Gewinn für N_1 ist

$$G_1 = 14(m - 1) + 15 + 0(M - m - 1) = \begin{cases} 14m + 1, & 1 \leq m \leq M \\ 14M, & m > M \text{ (immer kooperieren)} \end{cases}$$

d.h. N_1 maximiert seinen Gewinn $G_1 \rightarrow 14M + 1$, wenn er am M ten Tag überläuft. Dies zur Kenntnis nehmend überläuft N_2 am M ten Tag, und so wird der maximale Gewinn von N_1 auf $G_1 \rightarrow 14(M - 1)$ reduziert. Dies zur Kenntnis nehmend überläuft N_1 am $(M - 1)$ ten Tag, um seinen Gewinn mit $G_1 \rightarrow 14(M - 1) + 1$ unter den neuen Umständen zu maximieren. So läuft das Argument rückwärts weiter bis zum ersten Tag. Das Nach Gleichgewicht für eine fixierte, endliche, allen bekannte Anzahl von Tagen ist klar: nicht pumpen und nichts kriegen.

* Teil (c) Strategien für Tagefolgen

Sei nun die Anzahl M der Tage den beiden Nachbarn unbekannt, aber sie schätzen ab, dass die Anzahl geometrisch verteilt ist, d.h. $P(M = m) = (1 - 1/\mu)^{m-1}/\mu$, $m \in \mathbb{N}$ mit $\mu \gg 1$. Es gelten

$$P(M \geq m) = \sum_{n=m}^{\infty} P(M = n) = \sum_{n=m}^{\infty} (1 - 1/\mu)^{m-1}/\mu = (1 - 1/\mu)^{m-1}$$

und

$$\mathbb{E}[M] = \sum_{m=1}^{\infty} m(1 - 1/\mu)^{m-1} / \mu = \mu.$$

Laut dieser Rechnungen ist μ eine Abschätzung von den Nachbarn, wie viele Tage das Handeln tatsächlich läuft. Mit $\mu = 1$ ist man beim letzten Abschnitt mit $M = 1$. Diese Verteilung wird nun verwendet, um Erwartungswerte zu bestimmen. Wenn der Nachbar N_1 am m ten Tag überläuft sollte, während Nachbar N_2 bis dorthin kooperieren würde, ist der erwartete Gewinn von N_1 gegeben durch

$$\begin{aligned} \mathbb{E}[G_1] &= 14 \sum_{n=1}^{m-1} P(M \geq n) + 15P(M \geq m) + 0 \sum_{n=m+1}^{\infty} P(M \geq n) \\ &= \frac{14}{\mu} \sum_{n=1}^{m-1} (1 - 1/\mu)^{n-1} + \frac{15}{\mu} (1 - 1/\mu)^{m-1} + 0 \sum_{n=m+1}^{\infty} (1 - 1/\mu)^{n-1} / \mu \\ &= \frac{14}{\mu} \frac{1 - (1 - 1/\mu)^{m-1}}{1 - (1 - 1/\mu)} + \frac{15}{\mu} (1 - 1/\mu)^{m-1} = 14 - (14 - 15/\mu)(1 - 1/\mu)^{m-1} \end{aligned}$$

aber wenn er immer kooperieren sollte,

$$\mathbb{E}[G_1] = 14 \sum_{n=1}^{\infty} P(M \geq n) = \frac{14}{\mu} \sum_{n=1}^{\infty} (1 - 1/\mu)^{n-1} = \frac{14}{\mu} \frac{1}{1 - (1 - 1/\mu)} = 14$$

d.h. für $\mu > 15/14$ zahlt es sich aus, dass er ununterbrochen kooperiert, wenn Nachbar N_2 ununterbrochen kooperiert. Jedoch wenn der Nachbar N_2 am m ten Tag überläuft, während N_1 bis dorthin kooperiert, ist der erwartete Gewinn für N_1

$$\begin{aligned} \mathbb{E}[G_1] &= 14 \sum_{n=1}^{m-1} P(M \geq n) + 0P(M \geq m) + 0 \sum_{n=m+1}^{\infty} P(M \geq n) \\ &= \frac{14}{\mu} \sum_{n=1}^{m-1} (1 - 1/\mu)^{n-1} + 0(1 - 1/\mu)^{m-1} / \mu + 0 \sum_{n=m+1}^{\infty} (1 - 1/\mu)^{n-1} / \mu \\ &= \frac{14}{\mu} \frac{1 - (1 - 1/\mu)^{m-1}}{1 - (1 - 1/\mu)} = 14 - 14(1 - 1/\mu)^{m-1} \end{aligned}$$

d.h. weniger als wenn er selber überläuft hätte. Die Botschaft ist, dass die Nachbarn mit Vorsicht kooperieren sollten.

Sei die Anzahl M der Tage den Nachbarn weiterhin unbekannt. Nun überlegen sie eine Strategie für eine Folge von Tagen anstatt für einen einzigen beliebigen zufälligen Tag. Wenn der Gegner immer überläuft, dann zahlt es sich nicht aus, dass ein Nachbar kooperiert. Auf der anderen Seite ist der Anreiz zu kooperieren entsprechend höher, wenn der Gegner in gewissem Ausmaß kooperiert. In dieser Hinsicht gibt es die folgende Strategie.

Strategie 1: Man beginnt mit Vorsicht und kooperiert am ersten Tag nicht. Nachher basiert die Entscheidung über das eigene Verhalten auf das Verhalten des Gegners. Der Prozentsatz der bisherigen Tage, an denen der Gegner kooperiert hat, sei durch $100p\%$ gegeben. Dann am bevorstehenden Tag kooperiert man mit Wahrscheinlichkeit p .

Die folgende Strategie basiert auf die Bayesche Regel, um den eigenen Gewinn zu maximieren. Man kooperiert mit einer Wahrscheinlichkeit, die durch die bedingte Wahrscheinlichkeit gegeben ist, dass der Gegner kooperiert, nachdem man selber kooperiert hat.

Strategie 2: Man beginnt mit Vorsicht und kooperiert am ersten Tag nicht. Nachher für die bisherigen Tage, an denen der Gegner kooperiert hat, sei $100q\%$ der Prozentsatz dieser, wobei am Tag vorher man selber kooperiert hat. Dann am bevorstehenden Tag kooperiert man mit Wahrscheinlichkeit q . Bei einer gierigen Variante überläuft man am bevorstehenden Tag mit einer kleinen Wahrscheinlichkeit, auch wenn die Kooperationsschwelle q erreicht worden ist.

Im nächsten Abschnitt werden diese zwei Strategien mit der folgenden *quid-pro-quo* Strategie verglichen, die für ihre Robustheit bekannt ist.

Strategie 3: Man beginnt mit Vertrauen und kooperiert am ersten Tag. Nachher an jedem bevorstehenden Tag verhält man sich, wie der Gegner sich am vorigen Tag verhalten hat. Eine Variante dieser Strategie ist *quid-pro-quo mit Vergeben*, wobei man mit einer kleinen Wahrscheinlichkeit am bevorstehenden Tag kooperiert, auch wenn der Gegner am vorigen Tag nicht kooperiert hat.

* Teil (d) Simulation

Das Handeln der zwei Nachbarn wird mit dem folgenden Matlab Code simuliert.

```

h1 = figure(1); close(h1); h1 = figure(1);
set(h1,'Position',[10 50 1500 500]);

M = 100; % number of days of dealing

G1 = 0; G1v = [];
G2 = 0; G2v = [];

n1 = 0; % number of 1-coops
n2 = 0; % number of 2-coops

k1 = 0; % 1-coops initially not
k2 = 1; % 1-coops initially

subplot(1,3,1)
disp('Strategie 1 gegen Strategie 3')
for m=1:M
    % gains
    G1 = G1 + (k2 == 1)*15 - (k1 == 1); G1v = [G1v,G1/m];
    G2 = G2 + (k1 == 1)*15 - (k2 == 1); G2v = [G2v,G2/m];

    n1 = n1+k1; % total 1-coops
    n2 = n2+k2; % total 2-coops
    p1 = n1/m; % freq of 1-coops
    p2 = n2/m; % freq of 2-coops

    l1 = k1; % last 1-decision
    l2 = k2; % last 2-decision

    k1 = 1.0*(rand(1) < p2); % 1 coops next with prob p2
    k2 = 1.0*(l1 == 1); % 2 does next what 1 did last

```

```

plot(1:m,G1v,'r',1:m,G2v,'b',[1,m],[14,14],'k:');
xlabel('Tage') % plot current state
ylabel('Durchschnittlicher Gewinn')
legend('G_1 mit S_1','G_2 mit S_3')
title('Strategie 1 gegen Strategie 3')
axis([1-eps m+eps -5 20])
drawnow;
end
disp(sprintf('Hufigkeit der Kooperationen vom Nachbar 1: %0.1e',p1))
disp(sprintf('Hufigkeit der Kooperationen vom Nachbar 2: %0.1e',p2))

M = 100; % number of days of dealing

G1 = 0; G1v = []; % 1-gain with strategy 2
G2 = 0; G2v = []; % 2-gain with strategy 3

n1 = 0; % number of 1-coops
n2 = 0; % number of 2-coops

m1 = 0; % number of 1-coops after 2-coops
m2 = 0; % number of 2-coops after 1-coops
q1 = 1; % freq of 1-coops after 2-coops
q2 = 1; % freq of 2-coops after 1-coops

k1 = 0; % 1-coops initially not
k2 = 1; % 2-coops initially

subplot(1,3,2)
disp('Strategie 2~ gegen Strategie 3~')
for m=1:M
    % gains
    G1 = G1 + (k2 == 1)*15 - (k1 == 1); G1v = [G1v,G1/m];
    G2 = G2 + (k1 == 1)*15 - (k2 == 1); G2v = [G2v,G2/m];

    n1 = n1+k1; % total 1-coops
    n2 = n2+k2; % total 2-coops
    p1 = n1/m; % freq of 1-coops
    p2 = n2/m; % freq of 2-coops

    l1 = k1; % last decision for 1
    l2 = k2; % last decision for 2

    k1 = 1.0*(rand(1) < q2); % 1 coops next with prob q2
    k1 = k1*(rand(1) < 0.95); % but with greed
    k2 = 1.0*(l1 == 1); % 2 does next what 1 did last
    k2 = 1.0*(k2 == 1) + ... % but with forgiveness
    1.0*(k2 == 0) ...
    *(rand(1) < 0.05);

```

```

m1 = m1 + ((l2 == 1) && (k1 == 1)); % number of 1-coops after 2-coops
m2 = m2 + ((l1 == 1) && (k2 == 1)); % number of 2-coops after 1-coops
q1 = m1/n1; % prop of 1-coops after 2-coops
q2 = m2/n2; % prop of 2-coops after 1-coops

plot(1:m,G1v,'r',1:m,G2v,'b',[1,m],[14,14],'k:')
xlabel('Tage') % plot current state
ylabel('Durchschnittlicher Gewinn')
legend('G_1 mit S_2~','G_2 mit S_3~')
title('Strategie 2~ gegen Strategie 3~')
axis([1-eps m+eps -5 20])
drawnow;
end
disp(sprintf('Hufigkeit der Kooperationen vom Nachbar 1: %0.1e',p1))
disp(sprintf('Hufigkeit der Kooperationen vom Nachbar 2: %0.1e',p2))

M = 100; % number of days of dealing

G1 = 0; G1v = []; % 1-gain with strategy 2
G2 = 0; G2v = []; % 2-gain 2 with tit-for-tag

n1 = 0; % number of 1-coops
n2 = 0; % number of 2-coops

m1 = 0; % number of 2-coops after 1-coops
m2 = 0; % number of 1-coops after 2-coops
q1 = 1; % freq of 2-coops after 1-coops
q2 = 1; % freq of 1-coops after 2-coops

k1 = 0; % 1-coops initially not
k2 = 0; % 2-coops initially not

subplot(1,3,3)
disp('Strategie 1 gegen Strategie 2~')
for m=1:M
    % gains
    G1 = G1 + (k2 == 1)*15 - (k1 == 1); G1v = [G1v,G1/m];
    G2 = G2 + (k1 == 1)*15 - (k2 == 1); G2v = [G2v,G2/m];

    n1 = n1+k1; % total 1-coops
    n2 = n2+k2; % total 2-coops
    p1 = n1/m; % freq of 1-coops
    p2 = n2/m; % freq of 2-coops

    l1 = k1; % last decision for 1
    l2 = k2; % last decision for 2

```

```

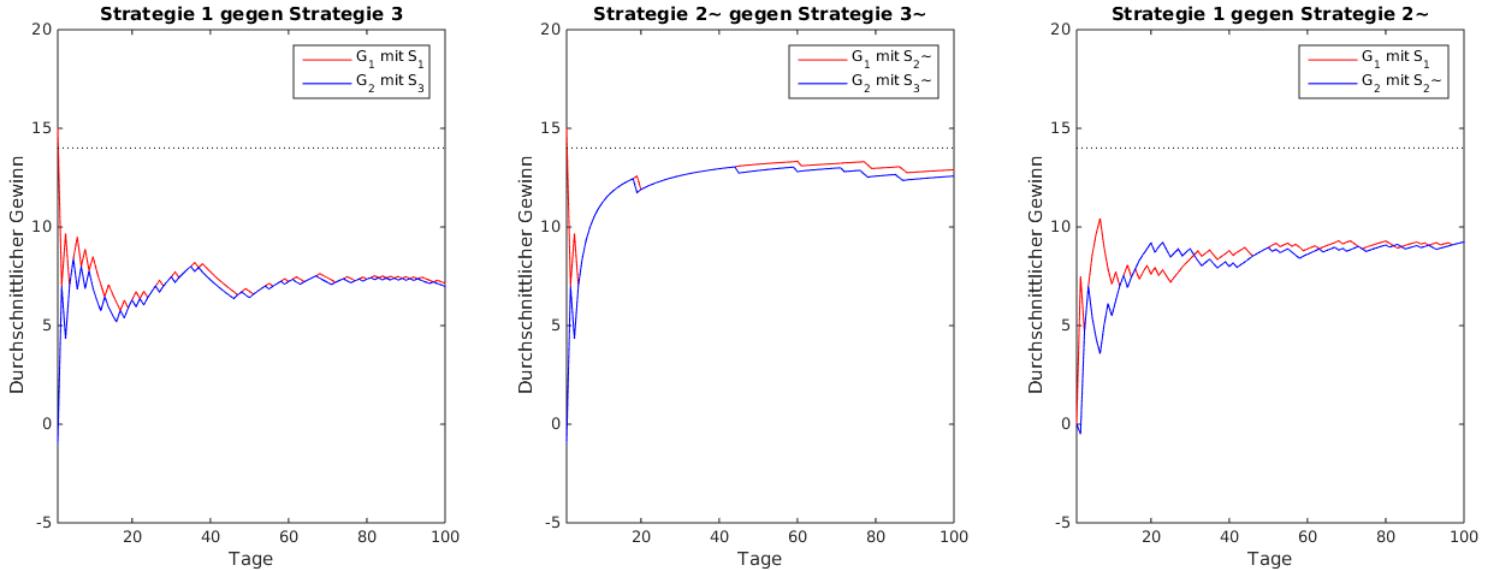
k1 = 1.0*(rand(1) < p2); % 1 coops next with prob p2
k2 = 1.0*(rand(1) < q1); % 2 coops next with prob q1
k2 = k2*(rand(1) < 0.95); % but with greed

m1 = m1 + ((l2 == 1) && (k1 == 1)); % number of 1-coops after 2-coops
m2 = m2 + ((l1 == 1) && (k2 == 1)); % number of 2-coops after 1-coops
q1 = m1/n1; % prop of 1-coops after 2-coops
q2 = m2/n2; % prop of 2-coops after 1-coops

plot(1:m,G1v,'r',1:m,G2v,'b',[1,m],[14,14],'k:')
xlabel('Tage') % plot current state
ylabel('Durchschnittlicher Gewinn')
legend('G_1 mit S_1','G_2 mit S_2~')
title('Strategie 1 gegen Strategie 2~')
axis([1-eps m+eps -5 20])
drawnow;
end
disp(sprintf('Hufigkeit der Kooperationen vom Nachbar 1: %0.1e',p1))
disp(sprintf('Hufigkeit der Kooperationen vom Nachbar 2: %0.1e',p2))

```

Die Ergebnisse werden folgendermaßen grafisch dargestellt. Aus den ersten zwei Grafiken ist die Robustheit der *quid-pro-quo* Strategie ersichtlich.



In der linken Grafik ist ein Vergleich der durchschnittlichen Gewinne *G₁ mit Strategie 1* und *G₂ mit Strategie 3*. Strategie 3 ist knapp niedriger als Strategie 1. Jedoch bleiben die beiden Nachbarn beim durchschnittlichen Gewinn 14, wenn Strategie 1 mit Vertrauen beginnt.

In der mittleren Grafik ist ein Vergleich der durchschnittlichen Gewinne *G₁ mit Strategie 2 mit Gier* und *G₂ mit Strategie 3 mit Vergeben*. Strategie 3 ist knapp niedriger als Strategie 2. Jedoch bleiben die beiden Nachbarn beim durchschnittlichen Gewinn 14, wenn Strategie 2 nicht gierig ist.

In der rechten Grafik ist ein Vergleich der durchschnittlichen Gewinne *G₁ mit Strategie 1* und *G₂ mit Strategie 2 mit Gier*. Hier gibt es Kreuzung zwischen den Kurven, und jede dieser Strategien kann

zum Schluss mit einem höheren durchschnittlichen Gewinn liegen. Daher können die Leistungen von Strategie 1 und Strategie 2 vergleichbar gesehen. Obwohl die Leistung von Strategie 3 hier konsistent knapp niedriger liegt, bleibt sie robust vergleichbar mit dem Gegner, ohne auffällig dabei zu versagen.

• Beispiel 4: Nash Verhandlung - Ressourceneinteilen

o Aufgabe

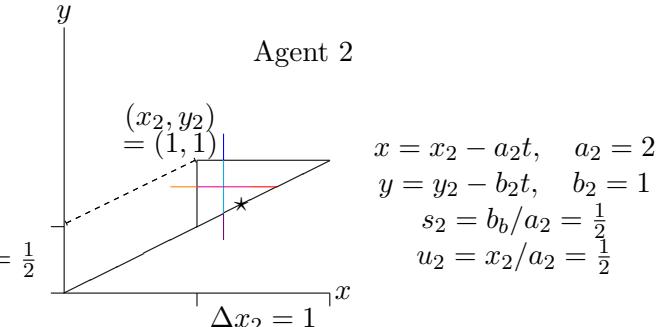
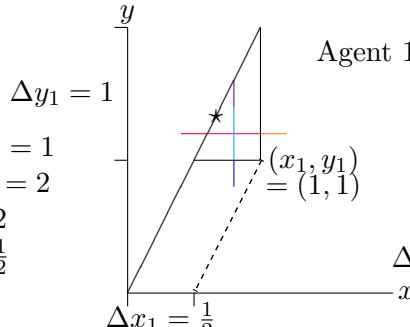
Zwei Personen treffen sich in der Wüste und überlegen einen Tausch von zwei Ressourcen x und y , damit ihre Lebenszeiten u_1 und u_2 verlängert werden können. Für die Ressourcen x und y hat die erste Person die Vorräte x_1 und y_1 und die entsprechenden Verbrauchsrationen a_1 und b_1 , und die zweite Person hat die Vorräte x_2 und y_2 und die entsprechenden Verbrauchsrationen a_2 und b_2 . Eine Person stirbt genau dann, wenn eine eigene Ressource verbraucht worden ist. Die Ressourcen und die Verbrauchsrationen sind: $x_i = y_i = 1$, $a_1 = b_2 = 1$, $a_2 = b_1 = 2$. Finden Sie ein Gleichgewicht bei diesem Tausch. Berechnen Sie den optimalen Tausch bei der Nash Verhandlungsstrategie, wenn das *Status Quo* bei dem Tausch ist: nichts tauschen.

o Lösung

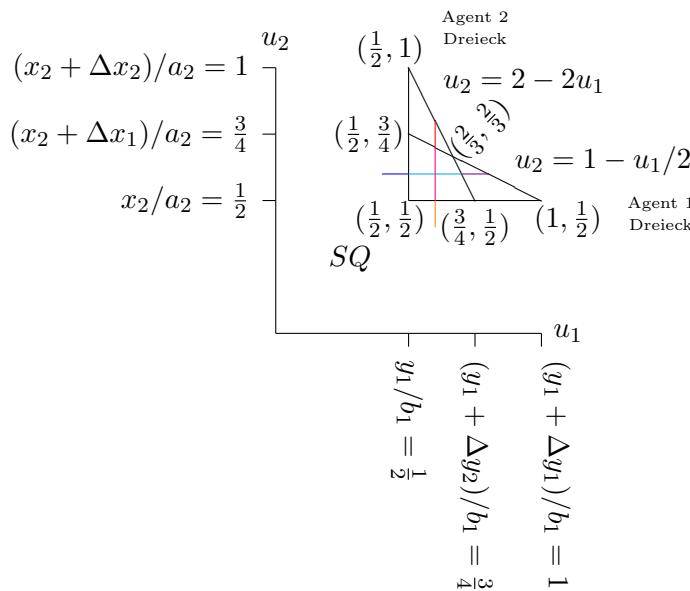
Die Werte entsprechen den Diagrammen im Skriptum.

$$\begin{aligned} & \Delta y_1 = 1 \\ & x = x_1 - a_1 t, \quad a_1 = 1 \\ & y = y_1 - b_1 t, \quad b_1 = 2 \\ & s_1 = b_1/a_1 = 2 \\ & u_1 = y_1/b_1 = \frac{1}{2} \\ & \Delta x_1 = \frac{1}{2} \end{aligned}$$

$$\begin{aligned} (y|_0 - y_1) = s_1(x|_{\Delta x_1} - x_1) \Rightarrow & \Delta x_1 = x_1 - y_1/s_1 = \frac{1}{2} \\ y|_{y_1 + \Delta y_1} = s_1 x|_{x_1} \Rightarrow & \Delta y_1 = s_1 x_1 - y_1 = 1 \end{aligned}$$



$$\begin{aligned} \Delta x_2 = y_2/s_2 - x_2 = 1 & \Leftarrow (y|_{y_2 + \Delta y_2} - y_2) = s_2(x|_{\Delta x_2} - x_2) \\ \Delta y_2 = y_2 - s_2 x_2 = \frac{1}{2} & \Leftarrow (y|_{\Delta y_2} - y_2) = s_2(x|_0 - x_2) \end{aligned}$$



Eine genauere Betrachtung der Utilitäten entlang der bunten Geradenstrecken führt zur Erkenntnis, dass das Spiel innerhalb des Schnitts der Dreiecke stattfindet. Auf der **magenta** Geradenstrecke bleibt die Lebenszeit des 1. Agenten gleich, aber auf der **roten** Verlängerung fällt seine Lebenszeit, und auf der **orangen** Verlängerung fällt die Lebenszeit des 2. Agenten. Auf der **cyan** Geradenstrecke bleibt die Lebenszeit des 2. Agenten gleich, aber auf der **violetten** Verlängerung fällt seine Lebenszeit, und auf der **blauen** Verlängerung fällt die Lebenszeit des 1. Agenten.

Sei ein beliebiger Punkt im Spielraum ausgewählt. An dieser Stelle überlegt ein Agent, ob er seinen Beitrag innerhalb des verfügbaren Spielraums ändern möchte, und zwar unter der Annahme, dass der andere Agent bei seinem Beitrag bleibt. Agent 1 würde sich aber nur waagerecht innerhalb seines Dreiecks bewegen und seine Lebenszeit nicht erhöhen. Analog würde Agent 2 sich nur senkrecht innerhalb seines Dreiecks bewegen und seine Lebenszeit nicht erhöhen. Da kein Spieler einen Anreiz hat, anders zu spielen, wenn der Gegner bei seiner Strategie bleibt, ist jede Stelle innerhalb des Spielraums ein Gleichgewicht.

Für die Nash Verhandlungsstrategie ist $p(u_1, u_2) = (u_1 - \frac{1}{2})(u_2 - \frac{1}{2})$ in dem Schnitt der zwei Dreiecke zu maximieren. Entlang der Gerade $u_2 = 1 - u_1/2$,

$$p(u_1, 1 - \frac{1}{2}u_1) = (u_1 - \frac{1}{2})(1 - \frac{1}{2}u_1 - \frac{1}{2}) = \max \quad \text{in} \quad u_1 = \frac{3}{4} \quad \text{mit} \quad p(\frac{3}{4}, \frac{5}{8}) = \frac{1}{32}$$

aber der Punkt $(\frac{3}{4}, \frac{5}{8})$ (Mittelpunkt der Strecke) befindet sich im Schnitt nicht. Entlang der Gerade $u_2 = 2 - 2u_1$,

$$p(u_1, 2 - 2u_1) = (u_1 - \frac{1}{2})(2 - 2u_1 - \frac{1}{2}) = \max \quad \text{in} \quad u_1 = \frac{5}{8} \quad \text{mit} \quad p(\frac{5}{8}, \frac{3}{4}) = \frac{1}{32}$$

aber der Punkt $(\frac{5}{8}, \frac{3}{4})$ (Mittelpunkt der Strecke) befindet sich im Schnitt nicht. Die Kreuzung der Geraden $u_2 = 1 - u_1/2$ und $u_2 = 2 - 2u_1$ liegt in $(\frac{2}{3}, \frac{2}{3})$, wo der maximale Wert $p(\frac{2}{3}, \frac{2}{3}) = \frac{1}{36}$ angenommen wird. Daher stellt der Punkt $(\frac{2}{3}, \frac{2}{3})$ den optimalen Tausch dar. Seien $\lambda \Delta y_2$ und $\mu \Delta x_1$ die Beträge von Ressourcen y_2 bzw. x_1 , die die Agenten einander überreichen. Mit

$$\frac{1 + \lambda/2}{2} = \frac{y_1 + \lambda \Delta y_2}{b_1} = \frac{2}{3}, \quad \frac{1 + \mu/2}{2} = \frac{x_2 + \mu \Delta x_1}{a_2} = \frac{2}{3}$$

folgen $\lambda = \mu = 2/3$, d.h. Agenten 1 und 2 geben $1/3$ von x_1 bzw. $1/3$ von y_2 her, und diese Stellen sind mit Sternen in den Diagrammen für die einzelnen Agenten markiert.

• Beispiel 5: Gleichgewicht - Wahlsystem

◦ Aufgabe

Um ein gemeinsames Reiseziel auszuwählen, verwenden die Personen einer Gruppe das folgende Wahlsystem, das natürlich für andere Entscheidungen verwendet werden kann. Es seien M Reiseziele und N Personen. Seien die Reiseziele mit Eckpunkten $\{\mathbf{r}_i\}_{i=1}^M \subset \mathbb{R}^{M-1}$ eines M -Simplex $\mathcal{S}_M \subset \mathbb{R}^{M-1}$ dargestellt, und $R = \{\mathbf{r}_1, \dots, \mathbf{r}_M\} \in \mathbb{R}^{(M-1) \times M}$. Die j te Person gibt eine *verteilte Wahlstimme* $\mathbf{w}_j^* \in \mathbb{R}^M$ ab, wobei $R\mathbf{w}_j^*$ eine konvexe Kombination der Reiseziele in \mathcal{S}_M ist, die die eigenen Präferenzen darstellt. Wenn der Entscheidungsprozess in einer einzigen Runde abgeschlossen werden soll, sind das verteilte Wahlergebnis \mathbf{r}^* und das reine Wahlergebnis $\hat{\mathbf{r}}$ gegeben durch

$$\hat{\mathbf{r}} = \underset{1 \leq i \leq M}{\operatorname{argmin}} \|\mathbf{r}_i - \mathbf{r}^*\|, \quad \mathbf{r}^* = \frac{1}{N} \sum_{j=1}^N R\mathbf{w}_j^*.$$

Mit $\mathbf{w}_j^{(1)} = \mathbf{w}_j^*$ sei nun $\mathbf{w}_j^{(l)}$ die verteilte Wahlstimme der j ten Person in einer l ten Runde, wobei das verteilte Ergebnis

$$\mathbf{r}^{(l)} = \frac{1}{N} \sum_{j=1}^N R\mathbf{w}_j^{(l)}, \quad l = 0, 1, 2, \dots$$

am Ende der l ten Runde bekanntgegeben wird. Angenommen wird $\mathbf{w}_j^{(l+1)}$ so ausgewählt, dass die eigene Utilität maximiert wird,

$$\mathbf{w}_j^{(l+1)} = \underset{\mathbf{w} \in \mathcal{S}_M}{\operatorname{argmin}} \|R(\mathbf{w} - \mathbf{w}_j^{(l)}) - \alpha_j^{(l)}(R\mathbf{w}_j^* - \mathbf{r}^{(l)})\|$$

wobei der Wert $\alpha_j^{(l)} = 2$ vorläufig gemeint ist, aber das Wählen kann durch $\alpha_j^{(l)} \in (0, 2)$ gebremst werden. Das reine Wahlergebnis nach der l ten Runde ist

$$\hat{\mathbf{r}}^{(l)} = \underset{1 \leq i \leq M}{\operatorname{argmin}} \|\mathbf{r}_i - \mathbf{r}^{(l)}\|.$$

- (a) Mit zufällig ausgewählten Präferenzen $\{\mathbf{w}_j^*\}_{j=1}^N$ der Wähler, simulieren Sie das Wahlsystem mit einem Matlab Code.
- (b) Zeigen Sie anhand der Wahlergebnisse nach einer Runde, dass das Simplex gleichseitig sein soll, da sonst eine nicht geeignete Korrespondenz zwischen dem verteilten und dem reinen Ergebnis sich ergeben kann.
- (c) Zeigen Sie, dass das Ergebnis nach der ersten Runde erfüllt die Bedingungen für eine *faire* Gruppenentscheidung im [Satz von Arrow](#) (über die Nichtexistenz einer fairen Gruppenentscheidung), nämlich
 - (i) die Gruppenentscheidung soll nicht von einer Person bestimmt werden,
 - (ii) wenn alle Personen \mathbf{r}_i vor \mathbf{r}_j bevorzugen, dann soll in der Gruppenentscheidung \mathbf{r}_i vor \mathbf{r}_j gereiht werden, und
 - (iii) wenn alle Personen \mathbf{r}_i vor \mathbf{r}_j bevorzugen und bei dieser Reihung bleiben, dann in der Gruppenentscheidung bleibt \mathbf{r}_i vor \mathbf{r}_j vorgereiht, unabhängig von Änderungen in der Einreihung von anderen Optionen \mathbf{r}_k , $k \neq i, j$.
- (d) Bestimmen Sie strategisch geeignete Werte $\{\alpha_j^{(l)}\}$.
- (e) Bestimmen Sie, ob die Iterierten $\{\mathbf{r}_k^{(l)}\}_{l=1}^\infty$ immer einen Fixpunkt haben,
- (f) und zwar welcher anhand des Ergebnisses der ersten Runde vorausgesagt werden kann oder nicht.

Bei Simulationen und anfänglichen Überlegungen nehmen Sie zur Vereinfachung an, dass $M = 3$ und $N = 2$ gelten, damit das Wahlsystem veranschaulicher und leicht grafisch dargestellt wird.

- o **Lösung**

- * **Teil (a) Implementierung**

Mit der Strategie $\alpha_j^{(l)} = 2$ wird das Wahlsystem mit dem folgenden Matlab Code simuliert.

```

h1 = figure(1); close(h1); h1 = figure(1); % setup figure
set(h1,'Position',[10 10 700 700]);

N = 2; % number of persons
M = 3; % number of options

K = 1000; % max number of iterations
tol = 1.0e-6; % termination criterion
al = 2; % weight on steps

w0 = rand(M,N); % true preferences
w0 = w0*diag(1./sum(w0,1));

I = eye(M); % true weighted choice
r0 = mean(w0,2);
dr = sum((kron(r0,ones(1,M)) - I).^2,1);
is = find(min(dr) == dr);
r0s = I(:,is); % true choice

if (M==3) % plot if visualizable
    plot3([1 0 0 1],[0 1 0 0],[0 0 1 0],'k')
    pbaspect([1 1 1]); view([1,1,1]); axis off;
    hold on;
    plot3([1 0],[0 0.5],[0 0.5],'k:')
    plot3([0 0.5],[1 0],[0 0.5],'k:')
    plot3([0 0.5],[0 0.5],[1 0],'k:')
    plot3(w0(1,:),w0(2,:),w0(3,:), ...
        'Color',[0,0,1],'Marker','*','LineStyle','none')
    plot3(r0(1),r0(2),r0(3), ...
        'Color',[0,0,1],'Marker','o')
    plot3(r0s(1),r0s(2),r0s(3), ...
        'Color',[0,0,1],'Marker','s','MarkerSize',10)
    drawnow;
end

% constraints: -w <= 0, 1'*w = 1
A = -eye(M); % A*w <= b
b = zeros(M,1);
Aeq = ones(1,M); % Aeq*w = beq
beq = 1;

% perform this prior iteration to get the number of iterations
% used for the graphical representation below
r1 = r0;
w1 = w0;
w2 = zeros(M,N);
for k=0:K
    for i=1:N % update ith voter

```

```

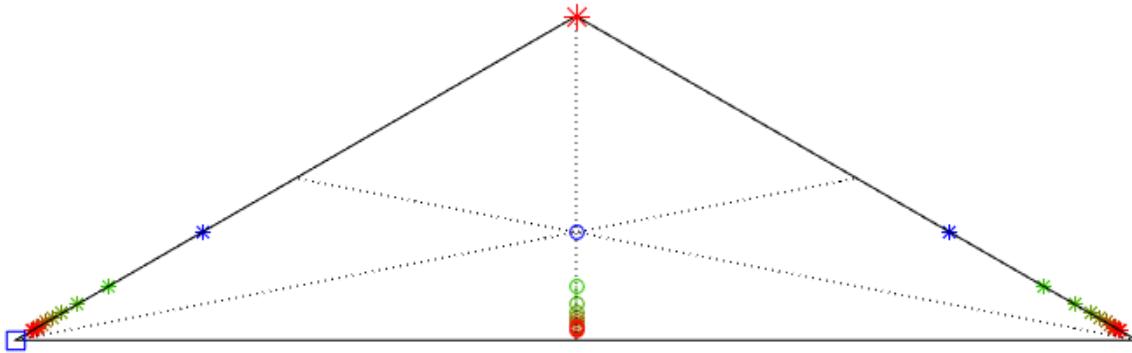
f = @(w) norm(w - w1(:,i) - al*(w0(:,i) - r1));
w2(:,i) = fmincon(f,w2(:,i),A,b,Aeq,beq);
end
r2 = mean(w2,2); % update group vote
if (norm(w1-w2) < tol)
    break;
end
w1 = w2;
r1 = r2;
end
if (k == K)
    input('no convergence with current parameters! ')
end
K = k;

% perform the iteration again just for the graphical representation
if (M == 3) % if visualizable
    r1 = r0;
    w1 = w0;
    w2 = zeros(M,N);
    for k=0:K
        for i=1:N % update ith voter
            f = @(w) norm(w - w1(:,i) - al*(w0(:,i) - r1));
            w2(:,i) = fmincon(f,w2(:,i),A,b,Aeq,beq);
        end
        r2 = mean(w2,2); % update group vote
        for l=1:N
            plot3([w1(1,1),w2(1,1)], [w1(2,1),w2(2,1)], [w1(3,1),w2(3,1)], ...
                'Color',[k/K,1-k/K,0], 'Marker','*')
        end
        plot3([r1(1),r2(1)], [r1(2),r2(2)], [r1(3),r2(3)], ...
            'Color',[k/K,1-k/K,0], 'Marker','o')
        plot3(r0(1),r0(2),r0(3), ...
            'Color',[0,0,1], 'Marker','o')
        plot3(w0(1,:),w0(2,:),w0(3,:), ...
            'Color',[0,0,1], 'Marker','*', 'LineStyle','none')
        drawnow;
        w1 = w2;
        r1 = r2;
    end
    dr = sum((kron(r2,ones(1,M)) - I).^2,1);
    is = find(min(dr) == dr);
    r2s = I(:,is); % final choice
    plot3(r2s(1),r2s(2),r2s(3), ...
        'Color',[1,0,0], 'Marker','*', 'MarkerSize',10)
    hold off;
end

```

* Teil (b) Simplex Gleichseitig

Der obige Code ist für das Beispiel in diesem Abschnitt nicht verwendet worden, und das Beispiel hat nur das Ziel, die Wirkung der Form des Simplex zu zeigen. Falls das Simplex nicht gleichseitig wäre, könnte z.B. die folgende Situation entstehen.



Das Format der Abbildungen ist Folgendes. Die **blauen Sternchen** sind die ursprünglichen verteilten Wahlstimmen. Der **blaue Kreis** ist das verteilte Wahlergebnis der ersten Runde. Das große **blaue Quadrat** ist das reine Wahlergebnis der ersten Runde. Mit den **grünen Stellen** laufen die Runden weiter, und mit den **roten Stellen** sind die Runden zu einem Gleichgewicht gekommen. Das große **rote Sternchen** ist das endgültige reine Wahlergebnis.

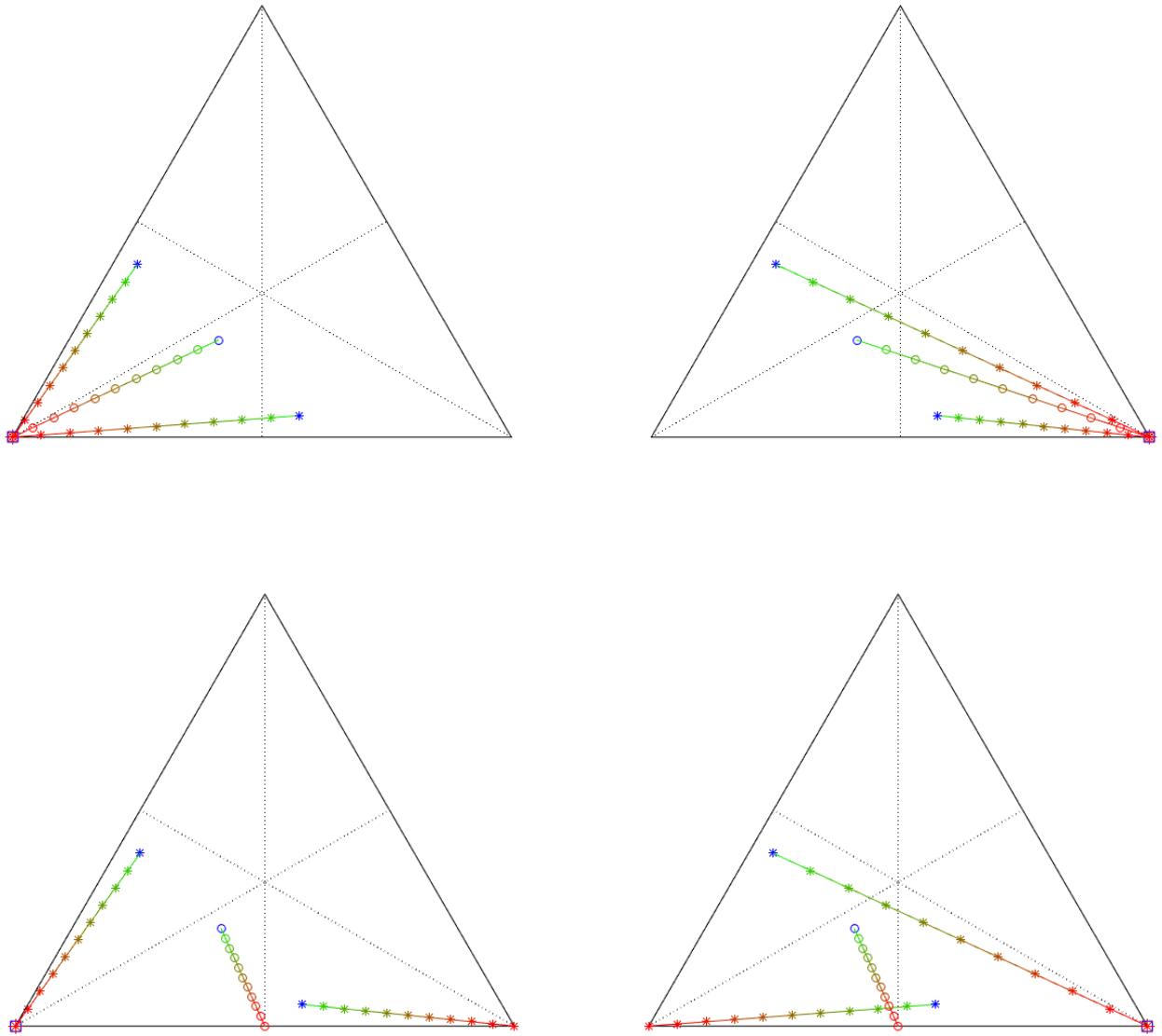
Im obigen Beispiel bevorzugen 2 Personen den obigen Eckpunkt *nicht*, sondern einen Eckpunkt unten links bzw. unten rechts. Im **Lauf** der Runden versuchen die Wähler, das verteilte Wahlergebnis gegen den gewünschten Eckpunkt zu ziehen. Die verteilten Wahlergebnisse **laufen** aber zum **Mittelpunkt** der unteren Seite, und geeignet wäre, wenn das reine Endergebnis im **großen roten Sternchen** mit dem ursprünglichen reinen Ergebnis unten links im **großen blauen Quadrat** übereinstimmen würde. Da hier das Simplex nicht gleichseitig ist, liegt das **rote Sternchen** im oberen Eckpunkt am nächsten zum Mittelpunkt der unteren Seite, und dieser wird nicht geeigneterweise das reine Endergebnis.

Zur Bequemlichkeit sei das $(M - 1)$ -dimensionale Simplex in \mathbb{R}^M mit $(r_i)_j = \delta_{i,j}$, $i, j = 1, \dots, M$, eingebettet, wie im Matlab Code zu sehen ist.

* Teil (c) Satz von Arrow

Der obige Algorithmus ist für die Beispiele in diesem Abschnitt nicht verwendet worden, und die eigene Utilitäten der Wähler werden nur mit dem Ziel künstlich bestimmt, Aspekte der Bedingungen des Arrow Satzes bildlich darzustellen. Die Bedingungen werden bezüglich dieser Beispiele mit nur zwei Wählern diskutiert, aber die Ergebnisse sind analog für eine höhere Anzahl von Wählern. In allen Beispielen fangen zwei Wähler mit den selben ursprünglichen **Präferenzen** an, aber im **Lauf** der Runden laufen sie gegen die **Eckpunkte**, wie in den Abbildungen zu sehen ist. Die Präferenzen sind so, dass die beiden Wähler untere Eckpunkte vor dem oberen Eckpunkt bevorzugen. Der Schluss ist, dass dieses Wahlsystem die Bedingungen vom Satz von Arrow erfüllen, aber aus dem folgenden Grund wird der Satz nicht widersprochen. Laut dem Satz von Arrow gibt es anhand von *Rangordnungen* der Wähler keine Gruppenentscheidung, die die drei Bedingungen erfüllen kann. Bei dem hier untersuchten

Wahlsystem wird mit einer *Verteilung* von Präferenzen mehr Information abgegeben.



Bedingung (i)

Man betrachtet hier nur die **blauen Markierungen**. Das verteilte Wahlergebnis ist der blaue **Kreis**, der von den *beiden Wählern* bestimmt wird. Der Kreis liegt hier näher zum Eckpunkt unten links, aber wenn der blaue Sternchen unten rechts weiter nach rechts liegen sollte, würde der blaue Kreis näher zum Eckpunkt unten rechts liegen. Wenn die beiden Wähler ihre jeweiligen Präferenzen in den gegenüberliegenden unteren Eckpunkten auswählen würden, gäbe es eine Kollisionsregel, um entweder den Eckpunkt links oder den Eckpunkt rechts auszuwählen. Klar wird das Wahlergebnis nicht nur von einem Wähler bestimmt.

Bedingung (ii)

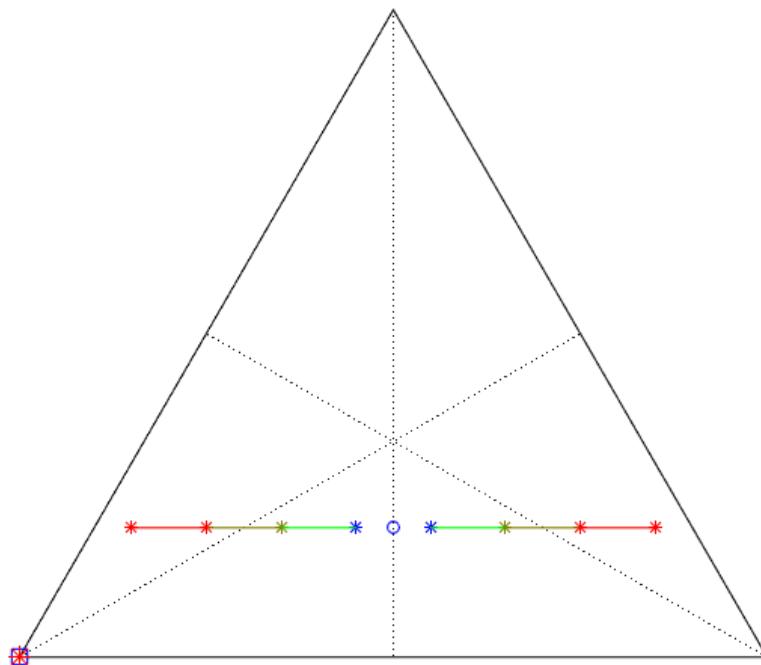
Man betrachtet hier wieder nur die blauen **Markierungen**. Mit den **blauen Sternchen** bevorzugen die beiden Wähler den Eckpunkt unten links vor dem oberen Eckpunkt. Dies erkennt man dadurch, dass die beiden Sternchen in einem Teildreieck $\hat{\mathcal{S}} \subset \mathcal{S}_M$ liegen, das das Simplex halbiert und den oberen Eckpunkt ausschließt. Das verteilte Wahlergebnis ist der blaue **Kreis**, der ebenfalls den Eckpunkt unten links vor dem oberen Eckpunkt bevorzugt. Diese Eigenschaft ergibt sich, weil $\hat{\mathcal{S}}$ konvex ist. Klar bleiben solche paarweise Präferenzen in der Gruppenentscheidung erhalten.

Bedingung (iii)

Hier werden alle Markierungen betrachtet. Für einen beliebigen unteren Eckpunkt r_i , entweder $i = 1$ oder $i = 2$, bevorzugen die beiden Wähler r_i vor dem oberen Eckpunkt r_3 , egal welche Züge in die Richtung des verbliebenen unteren Eckpunktes r_j , $j \neq i, 3$, unternommen werden. Klar bleiben solche paarweise Präferenzen in der Gruppenentscheidung erhalten, unabhängig von Änderungen in der Einreihung von anderen Eckpunkten.

* Teil (d) Strategischen Schritt

Der obige Code ist für die Beispiele in diesem Abschnitt nicht verwendet worden, und die Beispiele haben nur das Ziel, die Auswahl an $\alpha_j^{(l)}$ grafisch abzubilden. Angenommen weiß kein Wähler, genau wie viele andere Wähler es tatsächlich gibt, aber von außen werden hier nur zwei Wähler betrachtet. Der Wert $\alpha_j^{(l)} = 2$ entspricht der Strategie, die in der folgenden Abbildung gezeigt wird.



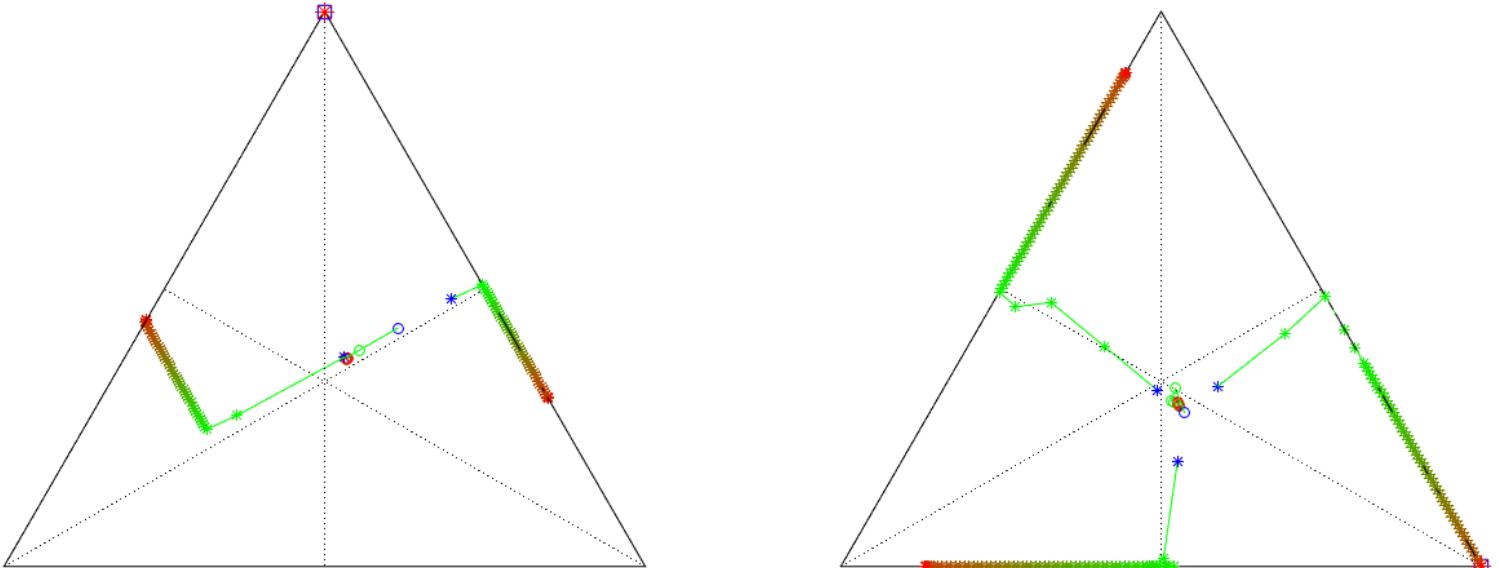
Hier formuliert man ein Modell der Wirkung aller anderen Wähler, das anhand der eigenen Wahlstimme zu dem bekanntgegebenen verteilten Wahlergebnis führen würde. Beispielsweise wird von der

Perspektive des Wählers mit dem linken blauen Sternchen $w_1^{(1)} = w_1^*$ betrachtet. Am Ende der ersten Runde wird der blaue Kreis bekannt gegeben. Egal wie viele Wähler es gegeben hat, ist der Kreis der Mittelpunkt zwischen den zwei blauen Sternchen, als ob es nur zwei Wähler gegeben hätte. Wenn das rechte blaue Sternchen $w_2^{(1)} = w_2^*$ stehen bleibt, dann soll der betrachtete Wähler in der nächsten Runde das linke (dunkel) grüne Sternchen $w_1^{(2)}$ auswählen, damit der Wunsch, d.h. das linke blaue Sternchen w_1^* , der Mittelpunkt zwischen $w_2^{(1)}$ und $w_1^{(2)}$ ist. Ebenfalls ist der linke (dunkel) grüne Sternchen $w_1^{(2)}$ der Mittelpunkt zwischen $w_2^{(2)}$ und dem linken (hell) roten Sternchen $w_1^{(3)}$. Diese Folge entspricht der Auswahl $\alpha_j^{(l)} = 2$. Der Parameter α kann aber kleiner ausgewählt werden, um die Dynamik zu bremsen, sogar einen Differentialprozess zu erzeugen.

* Teil (d) Stabilen Fixpunkt

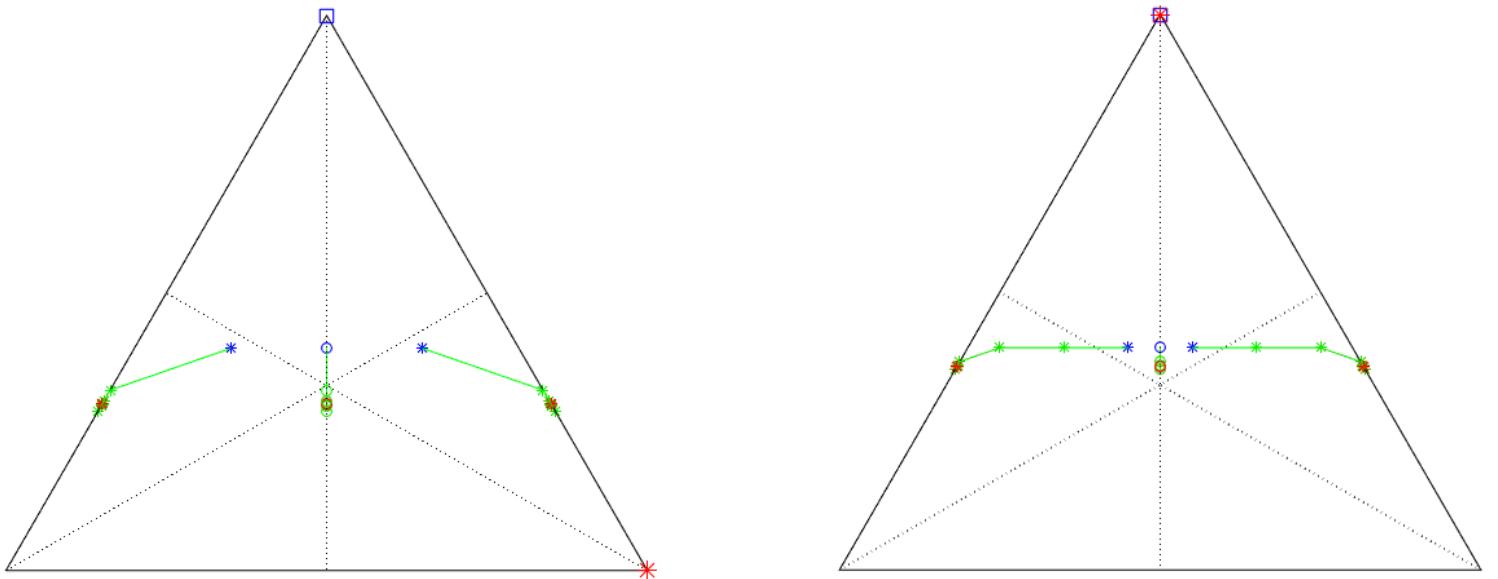
Die Wahl-Strategie des letzten Abschnitts ist im obigen Matlab Code implementiert worden, und der Code ist für alle Beispiele in diesem und im nächsten Abschnitt verwendet worden. Keine zufällig ausgewählten Anfangswerte sind gefunden worden, bei denen die Iterierten nicht konvergieren. Diese Beobachtung ist natürlich kein Beweis der allgemeinen Existenz eines eindeutig bestimmten Fixpunktes.

In den zwei nächsten Beispielen sind ganz viele Iterationen notwendig, damit die Folge konvergiert. In der linken Grafik wählen zwei Personen, und in der rechten Grafik wählen drei Personen.



- * Teil (e) Voraussagbar

In den nächsten zwei Beispielen sieht man, dass das verteilte Ergebnis der ersten Runde nicht voraussagen kann, was das Endergebnis sein wird. Im ersten Fall liegt der anfängliche blaue Kreis näher zum oberen Eckpunkt, obwohl der endgültige rote Kreis nicht, und das Endergebnis liegt unten.



Im zweiten Fall liegt der anfängliche blaue Kreis näher zum oberen Eckpunkt, und der endgültige rote Kreis auch, also ist das Endergebnis oben. In den beiden Fällen stellt das Endergebnis die Wünsche der Wähler gerecht dar, da im ersten Fall der obere Kreis nicht bevorzugt war, und im zweiten Fall schon.

- **Beispiel 6: Gleichgewicht - Prüfungssystem**

- **Aufgabe**

Um einen Kurs positiv abzuschließen müssen die Teilnehmer eine Prüfung mit *multiple response* Fragen bestehen. Dafür gibt es für jede Frage vier Antwortmöglichkeiten, wobei mindestens eine richtig ist und mindestens eine falsch ist. Alle richtigen Antworten werden gleich positiv gewichtet und alle falschen Antworten gleich negativ. Die Summe der positiven Gewichte ist $+1$, und die Summe der negativen Gewichte ist -1 . Der Kurs wird positiv abgeschlossen, wenn die durchschnittliche Leistung bei mindestens 0.5 liegt. Die Teilnehmer überlegen eine optimale Strategie der Prüfungskunst, und der Kursleiter überlegt eine optimale Strategie, die eine Motivation zum Lernen anbietet. Bestimmen Sie die besten jeweiligen Strategien.

- **Lösung**

Unter der Bedingung, dass es mindestens eine richtige Antwort und mindestens eine falsche Antwort gibt, gibt es 14 verschiedene Möglichkeiten für die Positionen von richtigen bzw. falschen Antworten:

$$\begin{aligned}\alpha_1 : 4 &= \binom{4}{1} \text{ Folgen mit genau einer richtigen Antwort an einer der vier Stellen} \\ \alpha_2 : 6 &= \binom{4}{2} \text{ Folgen mit genau zwei richtigen Antworten an zwei der vier Stellen} \\ \alpha_3 : 4 &= \binom{4}{3} \text{ Folgen mit genau drei richtigen Antworten an drei der vier Stellen.}\end{aligned}$$

Seien alle Möglichkeiten gleich wahrscheinlich. Diese 3 Kategorien entsprechen auch den möglichen Strategien des Kursleiters, die auch mit α_i bezeichnet werden. Sei L eine Zufallsvariable für seine ausgewählte Strategie. Da alle Möglichkeiten gleich wahrscheinlich sind, folgen

$$P(L = \alpha_1) = 4/14, \quad P(L = \alpha_2) = 6/14, \quad P(L = \alpha_3) = 4/14.$$

Die verfügbaren Strategien für einen Teilnehmer werden bezeichnet mit

$$\sigma_i = \text{genau } i \text{ Antwortmöglichkeiten auswählen, } i = 1, 2, 3.$$

Die Gewichtungen der Antwortmöglichkeiten für die Kursleiter-Strategien α_i , $i = 1, 2, 3$, sind

$$\begin{aligned}\alpha_1 : & +1 \text{ für die richtige Antwort, } -1/3 \text{ für die falschen Antworten,} \\ \alpha_2 : & +1/2 \text{ für die richtigen Antworten, } -1/2 \text{ für die falschen Antworten,} \\ \alpha_3 : & +1/3 \text{ für die richtigen Antworten, } -1 \text{ für die falsche Antwort.}\end{aligned}$$

Erwartungswerte für die zufällige Leistung eines Teilnehmers mit der Strategie σ_i , $i = 1, 2, 3$, werden bestimmt. Es werden aber verschiedene Vorbereitungsstufen betrachtet. Sei X_i eine Zufallsvariable für einen Teilnehmer mit der Prüfungsstrategie σ_i , der sich nicht vorbereitet hat. Sei Y_i eine Zufallsvariable für einen Teilnehmer mit der Prüfungsstrategie σ_i , der ausreichend gelernt hat, um entweder genau eine richtige oder genau eine falsche Antwort zuverlässig bestimmen zu können, je nachdem was zu einer besseren Leistung führt. Sei Z_i eine Zufallsvariable für einen Teilnehmer mit der Prüfungsstrategie σ_i , der ausreichend gelernt hat, um genau eine richtige und genau eine falsche Antwort zuverlässig bestimmen zu können.

Die Möglichkeiten für Y_i , $i = 1, 2, 3$, mit $L = \alpha_j$, $j = 1, 2, 3$, werden in den folgenden Tabellen explizit aufgelistet. Ein grünes \checkmark bedeutet, dass der Teilnehmer diese Antwortmöglichkeit als richtig erkennen kann. Ein grünes \otimes bedeutet, dass der Teilnehmer diese Antwortmöglichkeit als falsch erkennen kann. Eine Zeile mit grünen * bedeutet, diese wäre eine mögliche Reaktion des Teilnehmers. Eine Zeile mit roten * bedeutet, diese wäre keine mögliche Reaktion des Teilnehmers.

$L = \alpha_1$				$L = \alpha_2$				$L = \alpha_3$			
✓	\otimes	\otimes	\otimes	✓	\checkmark	\otimes	\otimes	✓	\checkmark	\checkmark	\otimes
*				*				*			
	*				*				*		
		*				*				*	
			*				*				*
$P(Y_1 = 1) = 1$				$P(Y_1 = \frac{1}{2}) = 1$				$P(Y_1 = \frac{1}{3}) = 1$			

$L = \alpha_1$	\checkmark	\otimes	\otimes	\otimes
	*	*		
	*		*	
	*			*
		*	*	
		*		*
		*		
			*	*

$$P(Y_2 = \frac{2}{3}) = 1$$

$L = \alpha_2$	\checkmark	\checkmark	\otimes	\otimes
	*	*		
	*		*	
	*			*
	*			*
		*	*	
		*		*
		*		
			*	*

$$P(Y_2 = 0) = \frac{2}{3}$$

$$P(Y_2 = 1) = \frac{1}{3}$$

$L = \alpha_3$	\checkmark	\checkmark	\checkmark	\otimes
	*	*		
	*		*	
	*			*
	*			*
		*	*	
		*		*
		*		
			*	*

$$P(Y_2 = \frac{2}{3}) = 1$$

$L = \alpha_1$	\checkmark	\otimes	\otimes	\otimes
	*	*	*	
	*		*	
	*			*
	*		*	*
		*	*	*
		*		*
		*		
			*	*

$$P(Y_3 = +\frac{1}{3}) = 1$$

$L = \alpha_2$	\checkmark	\checkmark	\otimes	\otimes
	*	*	*	
	*		*	
	*			*
	*		*	*
		*	*	*
		*		*
		*		
			*	*

$$P(Y_3 = \frac{1}{2}) = 1$$

$L = \alpha_3$	\checkmark	\checkmark	\checkmark	\otimes
	*	*	*	
	*		*	
	*			*
	*		*	*
		*	*	*
		*		*
		*		
			*	*

$$P(Y_3 = 1) = 1$$

Die Erwartungswerte für die Zufallsvariablen Y_i , $i = 1, 2, 3$, werden durch die Bayesche Regel gegeben:

$$\begin{aligned} \mathbb{E}(Y_1) &= \sum_{y \in \{+1\}} y \cdot \underbrace{P(Y_1 = y \mid L = \alpha_1)}_{1, y=+1} \cdot \underbrace{P(L = \alpha_1)}_{4/14} \\ &+ \sum_{y \in \{+\frac{1}{2}\}} y \cdot \underbrace{P(Y_1 = y \mid L = \alpha_2)}_{1, y=+\frac{1}{2}} \cdot \underbrace{P(L = \alpha_2)}_{6/14} \\ &+ \sum_{y \in \{+\frac{1}{3}\}} y \cdot \underbrace{P(Y_1 = y \mid L = \alpha_3)}_{1, y=+\frac{1}{3}} \cdot \underbrace{P(L = \alpha_3)}_{4/14} \\ &= 25/42 \approx 0.60 \end{aligned}$$

$$\begin{aligned} \mathbb{E}(Y_2) &= \sum_{y \in \{+\frac{2}{3}\}} y \cdot \underbrace{P(Y_2 = y \mid L = \alpha_1)}_{1, y=+\frac{2}{3}} \cdot \underbrace{P(L = \alpha_1)}_{4/14} \\ &+ \sum_{y \in \{0, +1\}} y \cdot \underbrace{P(Y_2 = y \mid L = \alpha_2)}_{\frac{2}{3}, y=0; \frac{1}{3}, y=+1} \cdot \underbrace{P(L = \alpha_2)}_{6/14} \\ &+ \sum_{y \in \{+\frac{2}{3}\}} y \cdot \underbrace{P(Y_2 = y \mid L = \alpha_3)}_{1, y=+\frac{2}{3}} \cdot \underbrace{P(L = \alpha_3)}_{4/14} \\ &= 22/42 \approx 0.52 \end{aligned}$$

$$\begin{aligned} \mathbb{E}(Y_3) &= \sum_{y \in \{+\frac{1}{3}\}} y \cdot \underbrace{P(Y_3 = y \mid L = \alpha_1)}_{1, y=+\frac{1}{3}} \cdot \underbrace{P(L = \alpha_1)}_{4/14} \\ &+ \sum_{y \in \{+\frac{1}{2}\}} y \cdot \underbrace{P(Y_3 = y \mid L = \alpha_2)}_{1, y=+\frac{1}{2}} \cdot \underbrace{P(L = \alpha_2)}_{6/14} \\ &+ \sum_{y \in \{+1\}} y \cdot \underbrace{P(Y_3 = y \mid L = \alpha_3)}_{1, y=+1} \cdot \underbrace{P(L = \alpha_3)}_{4/14} \\ &= 25/42 \approx 0.60 \end{aligned}$$

Mit dem selben Format wie oben werden die Möglichkeiten für Z_i , $i = 1, 2, 3$, mit $L = \alpha_j$, $j = 1, 2, 3$, in den folgenden Tabellen explizit aufgelistet.

$L = \alpha_1$			
✓	\otimes	\otimes	\otimes
*			
	*		
		*	
			*

$$P(Z_1 = 1) = 1$$

$L = \alpha_2$			
✓	✓	\otimes	\otimes
*			
	*		
		*	
			*

$$P(Z_1 = +\frac{1}{2}) = 1$$

$L = \alpha_3$			
✓	✓	✓	\otimes
*			
	*		
		*	
			*

$$P(Z_1 = +\frac{1}{3}) = 1$$

$L = \alpha_1$			
✓	\otimes	\otimes	\otimes
*	*		
*		*	
*			*
*			
	*	*	
	*		*
		*	*

$$P(Z_2 = +\frac{2}{3}) = 1$$

$L = \alpha_2$			
✓	✓	\otimes	\otimes
*	*		
*		*	
*			*
*			
	*	*	
	*		*
		*	*

$$P(Z_2 = 0) = \frac{1}{2}$$

$$P(Z_2 = 1) = \frac{1}{2}$$

$L = \alpha_3$			
✓	✓	✓	\otimes
*	*		
*		*	
*			*
*			
	*	*	
	*		*
		*	*

$$P(Z_2 = +\frac{2}{3}) = 1$$

$$P(Z_2 = 1) = \frac{1}{2}$$

$L = \alpha_1$			
✓	\otimes	\otimes	\otimes
*	*	*	
*	*		*
*		*	*
*		*	*
	*	*	*
	*		*
		*	*
			*

$$P(Z_3 = +\frac{1}{3}) = 1$$

$L = \alpha_2$			
✓	✓	\otimes	\otimes
*	*	*	
*		*	
*			*
*			
	*	*	
	*		*
		*	*

$$P(Z_3 = +\frac{1}{2}) = 1$$

$L = \alpha_3$			
✓	✓	✓	\otimes
*	*	*	
*		*	
*			*
*			
	*	*	
	*		*
		*	*

$$P(Z_3 = 1) = 1$$

Die Erwartungswerte für die Zufallsvariablen Z_i , $i = 1, 2, 3$, sind durch die Bayesche Regel gegeben:

$$\begin{aligned} \mathbb{E}(Z_1) &= \sum_{z \in \{+1\}} z \cdot \underbrace{P(Z_1 = z \mid L = \alpha_1)}_{1, z=+1} \cdot \underbrace{P(L = \alpha_1)}_{4/14} \\ &+ \sum_{z \in \{\frac{1}{2}\}} z \cdot \underbrace{P(Z_1 = z \mid L = \alpha_2)}_{1, z=+\frac{1}{2}} \cdot \underbrace{P(L = \alpha_2)}_{6/14} \\ &+ \sum_{z \in \{\frac{1}{3}\}} z \cdot \underbrace{P(Z_1 = z \mid L = \alpha_3)}_{1, z=+\frac{1}{3}} \cdot \underbrace{P(L = \alpha_3)}_{4/14} \\ &= 25/42 \approx 0.60 \end{aligned}$$

$$\begin{aligned}
\mathbb{E}(Z_2) &= \sum_{z \in \{+\frac{2}{3}\}} z \cdot \underbrace{P(Z_2 = z \mid L = \alpha_1)}_{1, z=+\frac{2}{3}} \cdot \underbrace{P(L = \alpha_1)}_{4/14} \\
&+ \sum_{z \in \{0,+1\}} z \cdot \underbrace{P(Z_2 = z \mid L = \alpha_2)}_{\frac{1}{2}, z=0; \frac{1}{2}, z=+1} \cdot \underbrace{P(L = \alpha_2)}_{6/14} \\
&+ \sum_{z \in \{+\frac{2}{3}\}} z \cdot \underbrace{P(Z_2 = z \mid L = \alpha_3)}_{1, z=+\frac{2}{3}} \cdot \underbrace{P(L = \alpha_3)}_{4/14} \\
&= 25/42 \approx 0.60
\end{aligned}$$

$$\begin{aligned}
\mathbb{E}(Z_3) &= \sum_{z \in \{+\frac{1}{3}\}} z \cdot \underbrace{P(Z_3 = z \mid L = \alpha_1)}_{1, z=+\frac{1}{3}} \cdot \underbrace{P(L = \alpha_1)}_{4/14} \\
&+ \sum_{z \in \{+\frac{1}{2}\}} z \cdot \underbrace{P(Z_3 = z \mid L = \alpha_2)}_{1, z=+\frac{1}{2}} \cdot \underbrace{P(L = \alpha_2)}_{6/14} \\
&+ \sum_{z \in \{+1\}} z \cdot \underbrace{P(Z_3 = z \mid L = \alpha_3)}_{1, z=+1} \cdot \underbrace{P(L = \alpha_3)}_{4/14} \\
&= 25/42 \approx 0.60
\end{aligned}$$

Ähnliche Rechnungen zeigen,

$$\mathbb{E}(X_1) = \mathbb{E}(X_2) = \mathbb{E}(X_3) = 0.$$

Anhand der Ergebnisse

$$\mathbb{E}(Y_1) = 0.60 = \mathbb{E}(Y_3)$$

kann eine Mischung der Kursleiter-Strategien α_i , $i = 1, 2, 3$, ausgenutzt werden, und daher ist es für den Kursleiter ratsam, eine reine Strategie zu überlegen.

Für Y_i , $i = 1, 2, 3$, ergeben sich die folgenden Erwartungswerte, wenn es genau eine richtige Antwort geben sollte,

$$\begin{aligned}
\mathbb{E}(Y_1 \mid L = \alpha_1) &= \sum_{y \in \{+1\}} y \cdot \underbrace{P(Y_1 = y \mid L = \alpha_1)}_{1, y=+1} \cdot \underbrace{P(L = \alpha_1)}_1 = 1 \\
\mathbb{E}(Y_2 \mid L = \alpha_1) &= \sum_{y \in \{\frac{2}{3}\}} y \cdot \underbrace{P(Y_2 = y \mid L = \alpha_1)}_{1, y=+\frac{2}{3}} \cdot \underbrace{P(L = \alpha_1)}_1 = 2/3 \\
\mathbb{E}(Y_3 \mid L = \alpha_1) &= \sum_{y \in \{-\frac{1}{3}\}} y \cdot \underbrace{P(Y_3 = y \mid L = \alpha_1)}_{1, y=-\frac{1}{3}} \cdot \underbrace{P(L = \alpha_1)}_1 = 1/3
\end{aligned}$$

oder genau zwei richtige Antworten,

$$\begin{aligned}
\mathbb{E}(Y_1 \mid L = \alpha_2) &= \sum_{y \in \{+\frac{1}{2}\}} y \cdot \underbrace{P(Y_1 = y \mid L = \alpha_2)}_{1, y=+\frac{1}{2}} \cdot \underbrace{P(L = \alpha_2)}_1 = 1/2 \\
\mathbb{E}(Y_2 \mid L = \alpha_2) &= \sum_{y \in \{0,+1\}} y \cdot \underbrace{P(Y_2 = y \mid L = \alpha_2)}_{\frac{2}{3}, y=0; \frac{1}{3}, y=+1} \cdot \underbrace{P(L = \alpha_2)}_1 = 1/3 \\
\mathbb{E}(Y_3 \mid L = \alpha_2) &= \sum_{y \in \{-\frac{1}{2}, +\frac{1}{2}\}} y \cdot \underbrace{P(Y_3 = y \mid L = \alpha_2)}_{\frac{1}{6}, y=-\frac{1}{2}; \frac{5}{6}, y=+\frac{1}{2}} \cdot \underbrace{P(L = \alpha_2)}_1 = 1/2
\end{aligned}$$

oder schließlich genau drei richtige Antworten,

$$\begin{aligned}\mathbb{E}(Y_1 \mid L = \alpha_3) &= \sum_{y \in \{+\frac{1}{3}\}} y \cdot \underbrace{P(Y_1 = y \mid L = \alpha_3)}_{1, y = +\frac{1}{3}} \cdot \underbrace{P(L = \alpha_3)}_1 = 1/3 \\ \mathbb{E}(Y_2 \mid L = \alpha_3) &= \sum_{y \in \{-\frac{2}{3}, +\frac{2}{3}\}} y \cdot \underbrace{P(Y_2 = y \mid L = \alpha_3)}_{\frac{1}{6}, y = -\frac{2}{3}; \frac{5}{6}, y = +\frac{2}{3}} \cdot \underbrace{P(L = \alpha_3)}_1 = 2/3 \\ \mathbb{E}(Y_3 \mid L = \alpha_3) &= \sum_{y \in \{+\frac{1}{3}, +1\}} y \cdot \underbrace{P(Y_3 = y \mid L = \alpha_3)}_{\frac{1}{2}, y = +\frac{1}{3}; \frac{2}{3}, y = +1} \cdot \underbrace{P(L = \alpha_3)}_1 = 1.\end{aligned}$$

Für Z_i , $i = 1, 2, 3$, ergeben sich die folgenden Erwartungswerte, wenn es genau eine richtige Antwort geben sollte,

$$\begin{aligned}\mathbb{E}(Z_1 \mid L = \alpha_1) &= \sum_{z \in \{+1\}} z \cdot \underbrace{P(Z_1 = z \mid L = \alpha_1)}_{1, z = +1} \cdot \underbrace{P(L = \alpha_1)}_1 = 1 \\ \mathbb{E}(Z_2 \mid L = \alpha_1) &= \sum_{z \in \{\frac{2}{3}\}} z \cdot \underbrace{P(Z_2 = z \mid L = \alpha_1)}_{1, z = \frac{2}{3}} \cdot \underbrace{P(L = \alpha_1)}_1 = 2/3 \\ \mathbb{E}(Z_3 \mid L = \alpha_1) &= \sum_{z \in \{+\frac{1}{3}\}} z \cdot \underbrace{P(Z_3 = z \mid L = \alpha_1)}_{1, z = +\frac{1}{3}} \cdot \underbrace{P(L = \alpha_1)}_1 = 1/3\end{aligned}$$

oder genau zwei richtige Antworten,

$$\begin{aligned}\mathbb{E}(Z_1 \mid L = \alpha_2) &= \sum_{z \in \{+\frac{1}{2}\}} z \cdot \underbrace{P(Z_1 = z \mid L = \alpha_2)}_{1, z = +\frac{1}{2}} \cdot \underbrace{P(L = \alpha_2)}_1 = 1/2 \\ \mathbb{E}(Z_2 \mid L = \alpha_2) &= \sum_{z \in \{0, +1\}} z \cdot \underbrace{P(Z_2 = z \mid L = \alpha_2)}_{\frac{1}{2}, z = 0; \frac{1}{2}, z = +1} \cdot \underbrace{P(L = \alpha_2)}_1 = 1/2 \\ \mathbb{E}(Z_3 \mid L = \alpha_2) &= \sum_{z \in \{+\frac{1}{2}\}} z \cdot \underbrace{P(Z_3 = z \mid L = \alpha_2)}_{1, z = +\frac{1}{2}} \cdot \underbrace{P(L = \alpha_2)}_1 = 1/2\end{aligned}$$

oder schließlich genau drei richtige Antworten,

$$\begin{aligned}\mathbb{E}(Z_1 \mid L = \alpha_3) &= \sum_{z \in \{+\frac{1}{3}\}} z \cdot \underbrace{P(Z_1 = z \mid L = \alpha_3)}_{1, z = +\frac{1}{3}} \cdot \underbrace{P(L = \alpha_3)}_1 = 1/3 \\ \mathbb{E}(Z_2 \mid L = \alpha_3) &= \sum_{z \in \{+\frac{2}{3}\}} z \cdot \underbrace{P(Z_2 = z \mid L = \alpha_3)}_{1, z = +\frac{2}{3}} \cdot \underbrace{P(L = \alpha_3)}_1 = 2/3 \\ \mathbb{E}(Z_3 \mid L = \alpha_3) &= \sum_{z \in \{+1\}} z \cdot \underbrace{P(Z_3 = z \mid L = \alpha_3)}_{1, z = +1} \cdot \underbrace{P(L = \alpha_3)}_1 = 1.\end{aligned}$$

Anhand der Ergebnisse

$$\begin{aligned}\mathbb{E}(Y_1 \mid L = \alpha_1) &= 1, & \mathbb{E}(Z_1 \mid L = \alpha_1) &= 1, \\ \mathbb{E}(Y_2 \mid L = \alpha_1) &= 2/3, & \mathbb{E}(Z_2 \mid L = \alpha_1) &= 2/3, \\ \mathbb{E}(Y_2 \mid L = \alpha_3) &= 2/3, & \mathbb{E}(Z_2 \mid L = \alpha_3) &= 2/3, \\ \mathbb{E}(Y_3 \mid L = \alpha_3) &= 1, & \mathbb{E}(Z_3 \mid L = \alpha_3) &= 1,\end{aligned}$$

können die reinen Kursleiter-Strategien α_1 oder α_3 ausgenutzt werden, und daher ist Strategie α_2 für den Kursleiter ratsam. Anhand der Ergebnisse

$$\begin{aligned}\mathbb{E}(Y_1 \mid L = \alpha_2) &= 1/2, & \mathbb{E}(Z_1 \mid L = \alpha_2) &= 1/2, \\ \mathbb{E}(Y_2 \mid L = \alpha_2) &= 1/3, & \mathbb{E}(Z_2 \mid L = \alpha_2) &= 1/2, \\ \mathbb{E}(Y_3 \mid L = \alpha_2) &= 1/2, & \mathbb{E}(Z_3 \mid L = \alpha_2) &= 1/2,\end{aligned}$$

ist es für einen Teilnehmer ratsam, entweder eine oder drei Antwortmöglichkeiten auszuwählen, außer absolut Sicherheit dafür herrscht, welche 2 Antwortmöglichkeiten richtig sind. So können die Paaren (α_2, σ_1) und (α_2, σ_3) als Gleichgewichte im Spiel gesehen werden.