Computerphysik WS 2018/2019

Prof. Dr. Petra Imhof, FU Berlin

Hausarbeit

18. Januar 2018

Abgabe bis **09.02.2019** 23:55 Uhr . Geben Sie bitte zusätzlich zum jupyter notebook auch ein pdf oder html-Export an.

Hinweis: Zur Lösung von linearen Gleichungssystemen können Sie `numpy.linalg.solve` verwenden. Andere `numpy` und `scipy`, die Sie verwenden dürfen, sind in der entsprechenden Aufgabe angegeben.

Für Zufallszahlen verwenden Sie `numpy.random`.

Erlaubt sind außerdem die üblichen "simplen" numpy Funktionen`numpy.log(), numpy.sin(), numpy.cos(), math.sqrt(), math.factorial(), numpy.arange(), numpy.exp(),...` *und natürlich* `matplotlib.pyplot`

1 Bewegung im einer Energielandschaft (25 Punkte)

Wir betrachten eine Energielandschaft mit dem Potential

$V(x, y) = 1.2x^4 + 0.1x^3 - 1.2x^2 - 0.4x + y^4 + 0.2y^3 - 1.1y^2 - 0.1y + 0.2x^4y^2 + 0.1$

a) Charakterisierung der Potantiallandschaft

(i) (1 Punkt) Plotten Sie das Potential als Funktion von x und y im Bereich $x \in [-1., 1.]$ und $y \in [-1., 1.]$

(ii) (4 Punkte) Erzeugen Sie ein zweidimensionales Gitter mit Abstand 0.1 in $x \in [-1., 1.]$ und $y \in [-1., 1.]$. Starten Sie von jedem der Gitterpunkte (x,y) auf dem Gitter eine lokale Optimierung und finden Sie den zugehörigen stationären Punkt. Implementieren Sie eventuell dafür benötigte Ableitungsfunktionen direkt. Runden Sie die erhaltenen (x,y)-Werte der Optima auf 3 dezimale Ziffern. Wieviele verschiedene (gerundete) Optima erhalten sie?

Stellen Sie die Zielpunkte der Optimierung für jeden Startpunkt farbig dar, indem Sie die Zellen eines Gitters nach den Zielpunkten einfärben. Zum plotten können sie z.B. `plt.imshow` verwenden.

(iii) (2 Punkte) Bestimmen Sie, ob es sich bei den Endpunkten der Optimierung um Minima, Maxima oder Sattelpunkte handelt. Stellen Sie, wieder in einem gefärbten Gitter, zu jedem Startpunkt die Art des erreichten Zielpunkts dar.

(iv) (1 Punkt) Ordnen Sie nun jeden Punkt, der *nicht* zu einem Minimum konvergiert, dem nächstgelegenen Minimum zu. Plotten Sie wiederum ein Gitter, dass nun nach den zugeprdeneten Minima gefärbt ist.

b) Betrachten sie nun die Bewegung eines Teilchens der Masse $m = 1kg$ durch das zweidimensionale Potential.

(i) (1 Punkt) Implementieren Sie hierzu einen Velocity-Verlet Integrator, welcher Positionen q und Geschwindigkeiten v nach

$$q(t + \Delta t) = q(t) + \Delta t v(t) + \frac{1}{2}\Delta t^2 a(t) \tag{1}$$

$$v(t + \Delta t) = v(t) + \frac{1}{2}\Delta t \left(a(t) + a(t + \Delta t)\right) \tag{2}$$

mit Zeitschritt Δt propagiert, a ist die Beschleunigung.

(ii) (4 Punkte) Simulieren Sie nun die Bewegung durch die Potentiallandschaft für 100000 Schritte mit einem Zeitschritt von $\Delta t = 0.01$s. Nutzen Sie folgende Startwerte (Positionen in m, Geschwigkeiten in m/s): (qx,qy)=(0.6,-0.2); (vx,vy)=(0.6,-0.8) und): (qx,qy)=(-0.8,0.6); (vx,vy)=(0.1,0.2)

Um das Teilchen im Bereich ([-1.,1],[-1,1]) zu halten, verwenden Sie periodische Randbedingungen. (Was man darunter versteht finden Sie in der Aufgabe zu Fuchs und Hase).

Plotten Sie die Zeitreihen der erhaltenen Positionen in (x,y). Was beobachten Sie? Diskutieren Sie Ihre Beobachtung.

(iii) (4 Punkte) Implementieren Sie nun eine Bewegung durch das Potential mittels Matropolis Monte-Carlo. Dazu gehen Sie folgendermassen vor: In jedem Schritt wählen Sie zufällig, ob eine Vorwärts- oder Rückwärtsbewegung ausgeführt werden soll, wobei beide mit gleicher Wahrscheinlichkeit eintreten sollen. Ermmitteln Sie für eine Bewegung mit Schrittlänge dq=0.01 in jeder Komponente die neue Position und die dazugehörige potentielle Energie E_{neu}. Ist diese Energie niedriger als die vorherige E_{alt}, akzeptieren Sie die neue Position. Ist Sie höher, akzeptieren Sie die neue Position mit der Wahrscheinlichkeit $\exp\left(-\beta(E_{neu} - E_{alt})\right)$. Setzen Sie β=0.6 . Verwenden Sie als Startposition (qx,qy)=(-0.8,0.6) und berechnen Sie 100000 Schritte. Stellen Sie die Trajektorie wieder als ein x,y-Bild dar.

Ordnen Sie die in der Trajektorie besuchten Positionen dem Gitter und dem entsprechenden zugehörigen Minimum aus Teilaufgabe a) zu. Das zugehörige Minimum soll im Folgenden einen Zustand charakterisieren. *Hinweis: Sollten Sie aus der vorherigen Teilaufgabe keine Einteilung des Gitters in zugehörige Minima haben, teilen Sie das Gitter in vier Quadranten und ordenen die Positionen der Trajektorie entsprechend diesen zu "Zuständen".*

Berechnen Sie die Wahrscheinlichkeit der Zustände in der Trajektorie vorzukommen und stellen Sie diese als ein Histogramm dar.

c) Diskrete Zustände und Übergangswahrscheinlichkeiten

(i) (4 Punkte) Aus der Zuordnung der Positionen zu Zuständen erhalten Sie eine Zeitreihe von Zuständen. Berechnen Sie aus dieser eine Matrix $\mathbf{T}$ der Übergangswahrscheinlihckeiten von jedem Zustand mit einem Monte-Carlo-Schritt in jeden anderen (inkl. desselben) zu gelangen. Die Matrix ist also $n \times n$ wenn n die Anzahl der Zustände ist. Die Summe der Wahrscheinlichkeiten einer Zeile muss 1 sein, da alle Möglichkeiten, des durch diese Zeile beschriebenen Zustands darin enthalten sind.

Berechnen Sie Eigenwerte und zugehörige rechte und linke Eigenvektoren. Hierzu können Sie `scipy.linalg.eig` verwenden. (Es können hierbei komplexe Eigenwerte/-vektoren auftreten. Betrachten Sie in solchen Fällen nur den Realteil.) Vergleichen sie den linken Eigenvektor zum größten Eigenwert mit der Wahrscheinlichkeitsverteilung der Zustände. Was fällt Ihnen auf?

(ii) (3 Punkte) Bestimmen Sie nun Übergangsmatrizen $\mathbf{T}(\tau)$ mit sog. Lag-time τ. Für z.B. eine Lag-time τ zählen Sie die Wahrscheinlichkeiten nach genau 2 Schritten von einen Zustand i zu einem Zustand j gelangt zu sein. Ermitteln Sie so Übergangsmatrozen für die Lag-times τ=1,100,200,...,1000. Berechnen Sie für jede dieser Matrizen die Eigenwerte. Plotten Sie für alle Eigenwerte ew außer dem ersten (Warum?) Eigenwert $-\frac{\tau}{log(ew)}$ gegen τ. Was beobachten Sie? Diskutieren Sie Ihre Beobachtung.

(iii) (1 Punkt) Wenden Sie die Übergangsmatrix, die Sie für $\tau = 1000$ erhalten haben $\mathbf{T}(\tau = 1000)$, 100mal von rechts auf einen Zeilenvektor $\vec{l}$ der Länge n mit Einträgen $1/n$ an, wobei n wieder die Anzahl der Zustände ist. Was fällt Ihnen auf?

2 Push it to the Limit (20 Punkte)

Gegeben sei die folgende Differentialgleichung

$$\ddot{x} = -k(x^2 - a)\dot{x} - b^2 x. \tag{3}$$

a) (6 Punkte) Zunächst wollen wir die Bewegungsgleichung lösen. Berechnen Sie die Trajektorie ausreichend lang. Nutzen Sie für die Parameter $k = b^2 = 1$ und variieren Sie a und Ihren Startvektor (starten Sie unter anderem einmal im Koordinatenursprung). Stellen Sie Ihre Ergebnisse in Plots $\dot{x}$ gegenüber x dar.

b) (3 Punkte) Disktuieren Sie Ihre Beobachtungen.

c) Nun fixieren wir einen Startvektor und zusätzlich noch $a = 2$.

(i) (2 Punkte) Schreiben Sie eine Funktion, die Ihnen den letzten vollständigen Orbit aus Ihrer Trajektorie extrahiert.

(ii) (3 Punkte) Berechnen Sie anschließend den Flächeninhalt des Orbits. Vergleichen Sie Ihr Ergebnis mit dem Ergebnis, das Sie durch Spline-Interpolation und Integration mit der Bibliothek *from scipy.interpolate import InterpolatedUnivariateSpline* erhalten.

(iii) (2 Punkte) Plotten Sie den Orbit inkl. benötigter Start- und Endpunkte. Heben Sie die tatsächlich berechnete Fläche farblich hervor.

(iv) (1 Punkt) Wie groß ist die Periodendauer T?

(v) (1 Punkt) Bestimmen Sie mit einer geeigneten Methode aus der Vorlesung dasjenige a, welches zu einer Periodendauer von T=10 führt.

d) (2 Punkte) Nun ergänzen wir die Differentialgleichung durch eine Inhomogenität:

$$\ddot{x} = -k(x^2 - a)\dot{x} - b^2 x - A\sin(\omega * t) \tag{4}$$

mit $\omega = \pi/2$ und $a = 2$. Bestimmen Sie bis zu welcher Amplitude A die Eigenfrequenz des Orbits erhalten bleibt und ab welcher Amplitude A die Schwingung mit der Anregerfrequenz ω stattfindet. Ihre Auswertung soll auf der numerischen Ableitung $\frac{dT}{dA}$ basieren.

3 Einmal um den Mond (15 Punkte)

Fouriertransformationen funktionieren nicht nur in einer Dimension, sondern auch in zwei:

$$\hat{a}_{k,l} = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} a_{m,n} \cdot \exp\left(-2\pi \mathrm{i} \left(\frac{mk}{M} + \frac{nl}{N}\right)\right) \tag{5}$$

für $k = 0, \ldots, M-1$ und $l = 0, \ldots, N-1$. Die Fouriertransformation wird also einmal auf alle Koordinatenrichtungen angewendet. Die Rücktransformation sieht entsprechend wie folgt aus:

$$a_{m,n} = \frac{1}{MN} \sum_{k=0}^{M-1} \sum_{l=0}^{N-1} \hat{a}_{k,l} \cdot \exp\left(2\pi \mathrm{i} \left(\frac{mk}{M} + \frac{nl}{N}\right)\right) \tag{6}$$

für $m = 0, \ldots, M-1$ und $n = 0, \ldots, N-1$.

a) (5 Punkte) Implementieren Sie die zweidimensionale Fouriertransformation und Rücktransformation als FFT. Da das Ergebnis der Fouriertransformation ein komplexes Signal ist, betrachten wir Betrag, Real- und Imaginärteil der komplexen Zahl separat. Für ein augenfreundliches Frequenzbild mit nach außen hin größer werdenden Frequenzen müssen die Sektoren im Frequenzraum geshiftet werden.
Hinzweis: Führen Sie die dazu die zweidimensional Fouriertransfromation auf ein Produkt aus

$$\begin{array}{|c|c|}\hline 2 & 1 \\ \hline 3 & 4 \\ \hline \end{array} \longrightarrow \begin{array}{|c|c|}\hline 4 & 3 \\ \hline 1 & 2 \\ \hline \end{array}$$

eindimensionalen Fourieretransformationen zurück.

b) (5 Punkte) Anwendung auf Standardmuster Erzeugen Sie 128×128 große Bilder (Arrays) und beschreiben Sie für folgende Muster den Einfluss auf den Frequenzraum:

- Quadrat im Zentrum in Abhängigkeit der Kantenlänge.
- Quadrat in Abhängigkeit der Position.
- Kreis vs. Quadrat.
- Horizontale periodische Linien in Abhängigkeit des Linienabstands.
- Vertikale vs. horizintale Linien. Was erwarten Sie für schiefe Linien?

c) (5 Punkte) Nun sollen Sie ein Bild bereinigen, also von Noise und anderen periodischen Störungen befreien, das am 23.12.1968 in der Apollo-8-Mission aufgenommen wurde. Lesen Sie die Datei Bild.png mit der Funktion `imread` aus dem Paket `matplotlib.image` ein und visualisieren Sie es mit `plt.imshow()`. Nutzen Sie nun Ihre zweidimensionale Fouriertransformationsfunktion, um das originale Bild wiederherzustellen. Dafür betrachten Sie nur den Betrag im Freuquenzraum. Um mehr sehen zu können, sollten Sie den Frequenzraum logarithmisch darstellen. Multiplizieren Sie zur "BereinigungÏhr Bild im Frequenzraum mit geeigneten Filtern, das heißt Arrays aus Nullen und Einsen. Den Effekt dieser Filter können Sie durch Rücktransformation betrachten. Zeigen Sie Ihr gefiltertes Bild im Orts- und Freuquenzraum und erläutern Sie Ihr Vorgehen.

4 Nichtlineares Pendel (15 Punkte)

Das mathematische Pendel stellt eine Idealisierung eines realen Pendels dar. Dabei wird eine Masse am Ende eines masselosen Stabes befestigt, welcher sich um einen Aufhängepunkt drehen kann. Alle Reibungseffekte werden vernachlässigt. Die Bewegung des Pendels kann dann durch seine Auslenkung aus der vertikalen Ruhelage beschrieben werden. Für kleine Auslenkungswinkel vereinfachen sich die Bewegungsgleichungen dann zu denen eines gedämpften harmonischen Oszillators, bei großen Auslenkungen jedoch wird die Bewegung durch eine nichtlineare Differentialgleichung beschrieben,

$$\ddot{\varphi} = -\sin\varphi - \gamma\dot{\varphi} + f_0 \cos\omega_f t. \tag{7}$$

Hierbei sei γ die (dimensionslose) Reibungskonstante und $f_0 \cos\omega t$ sei die externe treibende Kraft. Die Anfangsbedingung seien $\varphi(t=0) = \varphi_0$ und $\dot{\varphi}(t=0) = 0$.

Unter der Bewegungsgleichung (7) kann das nichtlineare Pendel entsprechend der vorgegebenen Randbedingungen eine reiche Fülle möglicher Trajektorien durchlaufen.

a) (4 Punkte) Berechnen Sie $\varphi(t)$ und $\dot{\varphi}(t)$ zunächst für $f_0 = 0$, $\gamma = 0$. Plotten Sie die φ über t (Amplitudenverlauf) und $\dot{\varphi}$ über φ (Phasenporträt) für $\varphi_0 \in 0, 0.1, 1, 3$. Begründen Sie Ihre Vorgehensweise und erläutern Sie Ihre Beobachtungen.

b) (3 Punkte) Variieren Sie nun $\gamma \in 0, 10^{-3}$, $f_0 \in 0, 10^{-2}$ und $\varphi_0 \in 0, 3$ für $\omega_f = 3$ und $\dot{\varphi}_0 = 0$. Plotten und erläutern Sie die Phasenporträts. Berücksichtigen Sie hierzu auch Zeiten $t \geq 5 \cdot 10^4$. *Hinweis: Nicht alle Parameterkombinationen sind sinnvoll.*

c) (6 Punkte) Bestimmen Sie das Leistungsspektrum des Pendels,

$$S(\omega) = |\tilde{\varphi}(\omega)|^2, \tag{8}$$

für das ungedämpfte und nicht getriebene Pendel ($\gamma = \omega_f = 0$) für $\varphi_0 \in [0.2, 3]$ (Schrittweite 0.2). Plotten Sie drei aussagekräftige Spektra. Bestimmen Sie die Schwingungsdauern aus den Spektren und stellen Sie diese in Abhängigkeit des Auslenkungswinkels dar. Wie verändert sich das Leistungsspektrum mit zunehmender Auslenkung? Plotten Sie auch die Spektra aus vorherigen Teilaufgabe. Was beobachten Sie hier?

Hinweis: Für diese Teilaufgabe dürfen Sie die gesamte `numpy`-Bibliothek benutzen.

d) (2 Punkte) Nun widmen wir uns dem Fall $\varphi_0 = 0$, $f_0 = 0.01$, $\omega_f = 3$ und $\gamma = 0$ im Zeitintervall $[0, 40]$ in Zeitschritten von $\Delta t = 0.05$. Bestimmen Sie die zwei Schwingungsfrequenzen, indem Sie die Datenpunkte mit zwei überlagerten Sinusschwingungen fitten.

5 Eine Geschichte über Hasen und Füchse (25 Punkte)

Hasen und Füchse jagen sich schon seit Ewigkeiten durch den Wald. Das wollen wir nun simulieren.

a) (11 Punkte) Approximieren Sie den Wald als ein 10×10 grosses Gitter, mit einem Gitterabstand $a = 1$. Sowohl Füchse als auch Hasen bekommen erst einmal jeweils ihr eigenes Array (beide sind in ihrem jeweiligen Gitter zu Beginn zufällig positioniert). Anschliessend müssen Sie deren Bewegungen im Wald simulieren (Zeitschritt Δt=1). Es handelt sich als um zwei partielle Differentialgleichungen, welche jeweils die "Tierdiffusion- Änderung der Anzahl Tiere pro Zeit und Ort beschreiben.

   (i) (1 Punkt) Implementieren Sie einen einfachen Eulerintegrator für partielle Differentialgleichungen in zwei Dimensionen.

(ii) (3 Punkte) Implementieren Sie eine Funktion, welche Füchse und Hasen sich unabhängig voneinander bewegen lässt. Sowohl Hase als auch Fuchs dürfen sich pro Zeitschritt nur ein Feld bewegen (diagonale Felder zählen auch). Um zu wissen, welche Bewegung jedes Tier macht, verwenden Sie einen einfachen Eulerschritt, um eine Matrix mit Aufenthaltswahrscheinlichkeiten zum neuen Zeitpunkt zu erhalten. Anhand von diesen Aufenthalswahrscheinlichkeiten losen Sie aus, auf welches Feld das jeweilige Tier im aktuellen Zeitschritt dann tatsächlich zieht. Nachdem sich alle Tiere bewegt haben, setzen Sie alles auf Null, bis auf die Positionen auf denen sich die Tiere aktuell befinden. Falls mehrere Hasen oder Füchse sich gleichzeitig auf einem Feld befinden, so gehen sie einfach verloren (Sie widmen sich später einer besseren Lösung). *Hinweis: Die Summe der Aufenthaltswahrscheinlichkeiten Zellen um und auf welchem das jeweilige Tier steht, ergeben nicht zwangsläufig* 1. *Finden Sie einen Weg den Übergang zu einer anderen Zelle dennoch "fair" zu gestalten!*

(iii) (1 Punkt) Erweitern Sie ihren Eulerintegrator und die Arrays derart, dass Sie den Wald mit absorbierenden Randbedingungen simulieren können

(iv) (2 Punkte) Simulieren (mit absorbierende Randbedingungen) & animieren (Plots von ein paar Zeitpunkten sind gleich viel Wert wie eine Animation) Sie 10 Hasen mit einer Bewegungswahrscheinlichkeit von einem Feld zu einem Nachbarfeld $p = 0.5$ und 4 Füchse mit einer Bewegungswahrscheinlichkeit $p = 0.75$ für jeweils mindestens 10 Schritte. Beobachten und beschreiben Sie Sie die Randbedingungen in Aktion! *Hinweis #1: Aufenthaltswahrscheinlichkeit ist nicht das Gleiche wie Bewegungswahrscheinlichkeit! Hinweis #2: Anstatt Hasen und Füchse anfangs zufällig zu positionieren, dürfen Sie diese hier geschickt manuell platzieren, um die Randbedingugen zu beobachten. Hinweis #3: Beachten Sie die Randbedingungen wenn Sie den Wald der Hasen und den Wald der Füchse initialisieren.*

(v) (2 Punkte) Neben absorbierenden Randbedingungen gibt es auch noch periodische Randbedingugen. Im Grunde bedeutet dies, dass wenn man aus dem Wald auf einer Seite raus geht, dann kommt man auf der gegenüberliegenden Seite wieder herein. Nehmen wir ein Beispiel: Fuchs ist am linken Rand des Waldes auf der Höhe h. Wenn dieser noch einen Schritt nach links machen würde, dann kommt er am rechten Rand auf der Höhe h wieder rein. (Entsprechendes gilt auch für rechts raus und links rein und für oben und unten) Erweitern Sie Ihre Funktionen derart, dass Sie den Wald nun auch mit periodischen Randbedingungen simulieren können.

(vi) (2 Punkte) Simulieren (periodische Randbedingungen) & animieren (Plots von ein paar Zeitpunkten sind gleich viel Wert wie eine Animation) Sie 10 Hasen und 4 Füchse mit den gleichen Bewegungswahrscheinlichkeiten für jeweils mindestens 10 Schritte. Beobachten und beschreiben Sie die Randbedingungen in Aktion! *Hinweis: Anstatt Hasen und Füchse anfangs zufällig zu positionieren, dürfen Sie diese hier geschickt manuell platzieren, um die Randbedingugen zu beobachten.*

b) (6 Punkte) Nun gestalten Sie die Simulation der Natur etwas realistischer. Dazu bauen Sie folgende zusätzliche Funktionen ein:

(i) (1 Punkt) Kommen wir noch Mal auf mehrere Tiere in einer Zelle zurück. Die Regel bleibt bestehen, dass am Ende eines jeden Zeitschrittes nur noch 1 oder 0 in den "Waldgitterzellen" stehen dürfen. Allerdings setzen Sie nun nicht einfach die Zelle gleich 1, sondern verteilen die Hasen und Füchse um. Das bedeutet: wenn mehrere Tiere auf dem gleichen Feld sind, so verteilen Sie diese auf erreichbare benachbarte freie Felder (zufällig). Sind keine Felder mehr frei, so verhungen alle weiteren Tiere in der zu vollen Zelle.

(ii) (1 Punkt) Sowohl Füchse als auch Hasen brauchen etwas zu Essen. Der Wald ist voller Leckerbissen für die Hasen. Tummeln sich allerdings 3 oder mehr Hasen über einen Zeitschritt auf einem Feld, so verhungern alle ((Zuviele Hasen zum Zeitpunkt t, dann sinf zum Zeitpunkt $t + dt$ alle verhungert). Gleiches gilt für die Füchse mit einer Ausnahme: Befindet sich gleichzeitig noch ein Hase auf dem Feld, so überleben alle.

(iii) (1 Punkt) Sowohl Hasen als auch Füchse haben das Bedürfniss sich zu reproduzieren. Verbringen nun 2 Hasen* oder 2 Füchse* einen Zeitschritt auf dem gleichen Feld, so gibt es Nachwuchs. Bei Hasen kann der Wurf 1 bis 5 neue Hasen zur Folge haben (Zufall), bei den Füchsen ist es immer nur 1 neuer Fuchs. Aber Vorsicht, die Natur regelt Überpopulationen nicht nur mit Hungersnot! Wir regulieren diese mit einer maximalen Gesamt-Popoulation von Füchsen $n_{\text{max fox}} = \begin{cases} \text{int}\left(\frac{n_{\text{rabbit}}}{2}\right) & n_{\text{rabbit}} > 0 \\ 1 & \text{sonst} \end{cases}$ und Hasen $n_{\text{max rabbit}}$.

Hierbei steht n_{rabbit} für die aktuelle Hasenpopulation. Auch haben Schwangerschaften (zumindest in Ihrer Simulation) etwas mit dem Zufall zu tun. So ist die Wahrscheinlichkeit, dass zwei Hasen zusammen Nachwuchs bekommen folgendermassen definiert

$$p_{\text{preg rabbit}} = 1 - \frac{n_{\text{rabbit}}}{n_{\text{max rabbit}}}$$

Ähnlich sieht es bei den Füchsen aus:

$$p_{\text{preg fox}} = 1 - \frac{n_{\text{fox}}}{n_{\text{rabbit}}}$$

Hier steht n_{fox} für die aktuelle Fuchspopulation.

(iv) (1 Punkt) Wenn ein Hase einen Fuchs auf einem Feld stehen sieht, wird dieser bestimmt nicht dort hin hoppeln! Das bedeutet, dass die Aufenthaltsswahrscheinlichkeit des Hasen auf dem Feld des Fuchses (und damit per Los dahin zu hoppeln) 0 ist.

(v) (1 Punkt) Der Fuchs wiederum möchte unbedingt zum Hasen. Daher ist die Aufenthatswahrscheinlichkeit auf dem Feld des Hasen zu sein 1. *Hinweis: Das bedeutet allerdings nicht, dass die Aufenthaltswahrscheinlichkeit für alle anderen Felder um den Fuchs herum* 0 *werden! Hat ein Fuchs mehrere Nachbarfelder mit Hasen zur Auswahl, so wird der schmackhafteste zufällig bestimmt.*

(vi) (1 Punkt) Treffen 1 odere mehrere Füchse auf 1 oder mehrere Hasen auf dem selben Feld und verweilen dort einen Zeitschritt, so werden alle Hasen gefressen.

c) (6 Punkte) Nun sollen Sie die Hasen & Füchse Simulation (Version 2.0) auch noch ausprobieren! Dazu nehmen Sie an der Wald ist 64×64 gross. Die Hasenpopulation ist auf 2000 Exemplare beschränkt. Unser Wald beinhaltet am Anfang ca. 2000 Hasen und 25 Füchse. Dieses Mal simulieren Sie 500 Zeitschritte. *Hinweis: Beachten Sie die Randbedingungen wenn sie den Wald der Hasen und den Wald Füchse initialisieren.*

(i) (3 Punkte) Absorbierende Randbedingungen

i. (2 Punkte) Simulieren & animieren (Plots von ein paar Zeitpunkten sind gleich viel Wert wie eine Animation) Sie den Wald mit allen Störtermen!

ii. (1 Punkt) Stellen Sie die Hasen- und Fuchspopulation über die Zeit dar.

(ii) (3 Punkte) Periodische Randbedingungen

(i) (2 Punkte) Simulieren & animieren (Plots von ein paar Zeitpunkten sind gleich viel Wert wie eine Animation) Sie den Wald mit allen Störtermen!

(ii) (1 Punkt) Stellen Sie die Hasen- und Fuchspopulation über die Zeit dar.

d) (2 Punkte) Diskutieren Sie Ihre Ergebnisse! Gehen Sie hierbei auch auf die Unterschiede zwischen den absorbierenden und periodischen Randbedingungen ein! Wie könnte man das Modell für die Simulation des Waldes verbessern, um realistischere Ergebnisse zu erhalten?

* Zur besseren Lesbarkeit und zur Vereinfachung der Simulation wird hier geschlechtsunabhängig immer nur von Fuchs oder Hase gesprochen, bzw. damit gerechnet.