
SIMULATION DES WATOR RÄUBER-BEUTE MODELLS UND ERWEITERUNGEN

31. Mai 2017

Sebastian von der Thannen
Thomas Heitzinger

TECHNISCHE UNIVERSITÄT WIEN

Somewhere, in a direction that can only be called recreational at a distance limited only by one's programming prowess, the planet WATOR swims among the stars. It is shaped like a torus, or doughnut, and is entirely covered with water. The two dominant denizens of Wa-Tor are sharks and fish, so called because these are the terrestrial creatures they most closely resemble. The sharks of Wa-Tor eat the fish and the fish of Wa-Tor seem always to be in plentiful supply.

Alexander Keewatin Dewdney 1984



1 Das klassische WATOR Programm

Das WATOR Programm (abgeleitet von Water-Torus) beschreibt ein einfaches Räuber-Beute Modell, die Beute - in unserem Fall Fische haben immer ausreichend Futter und keine natürliche Sterberate. Einzig die Räuber - in unserem Fall Haie können ihnen gefährlich werden, diese können ohne Fische als Nahrungsquelle nicht überleben und würden nach kurzer Zeit aussterben. Wäre das auch schon das ganze Modell, so könnte man diese Zusammenhänge durch die klassischen Lotka-Volterra Gleichungen beschreiben

$$\begin{aligned}x'_B(t) &= x_B(t) (\alpha - \beta x_R(t)) \\x'_R(t) &= x_R(t) (\gamma x_B(t) - \delta)\end{aligned}\tag{1.1}$$

Das Wachstum der Beute $b(t)$ ist abhängig von der aktuellen Beutepopulation, sowie auf positive Weise von der Reproduktionsrate $\alpha > 0$, und negativ durch die von den Räuber verursachte Sterberate $\beta > 0$. Ganz ähnlich ist das zeitliche Verhalten der Räuberpopulation proportional zur Fressrate pro Beutelebewesen $\gamma > 0$ und zur natürlichen Sterberate $\delta > 0$ wenn keine Beute vorhanden ist.

Wir wollen die Sache jedoch ein wenig genauer wissen. Anstatt idealisierte stetige Populationsgrößen x_B, x_R anzunehmen, wollen wir tatsächlich eine diskrete Anzahl von einzelne Lebewesen simulieren, und deren Erfolg (oder Scheitern) von ihrem Aufenthaltsort auf WATOR, sowie anderen Lebewesen in unmittelbarer Nähe abhängig machen. Das WATOR Programm enthält einige einfache Regeln, die das Verhalten der Fische und Haie bestimmen. Der Ozean, in dem sie sich tummeln, besteht aus einem rechteckigen Gitter, dessen gegenüberliegende Seiten verbunden werden. Das heißt einfach, dass ein Fisch oder Hai der sich etwa in einer Zelle am rechten Rand befindet und nach rechts schwimmt, in der entsprechenden Zelle am linken Rand wieder auftaucht. Die Zeit vergeht in diskreten Zeitabschnitten die Chronen genannt werden, und jeder Fisch oder Hai darf sich pro Chronen um eine Zelle entweder nach Norden, Osten,

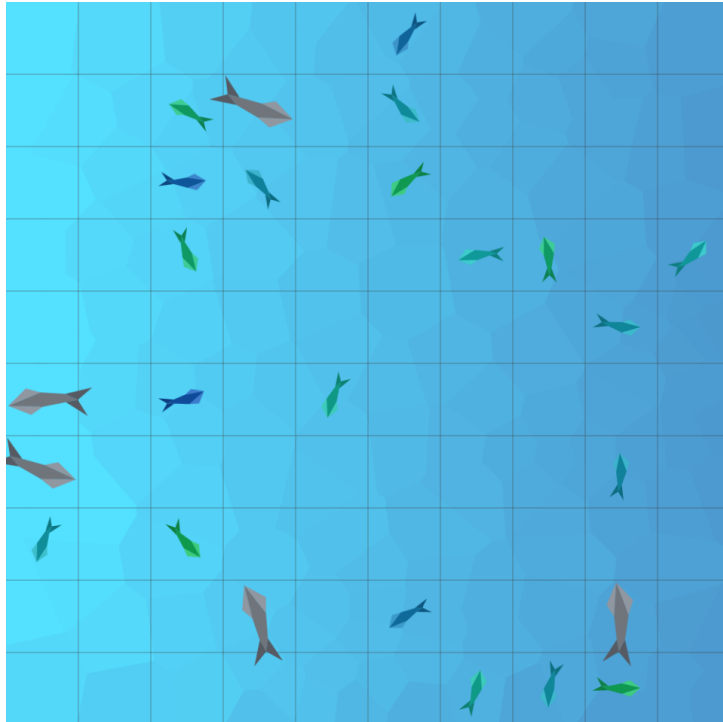
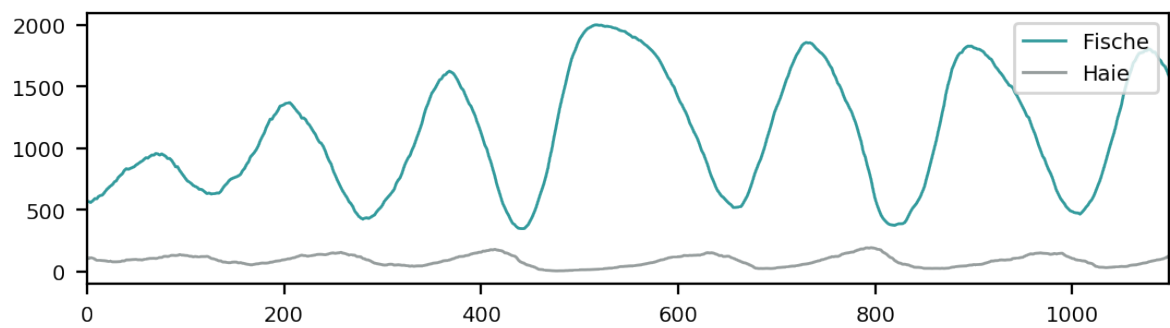
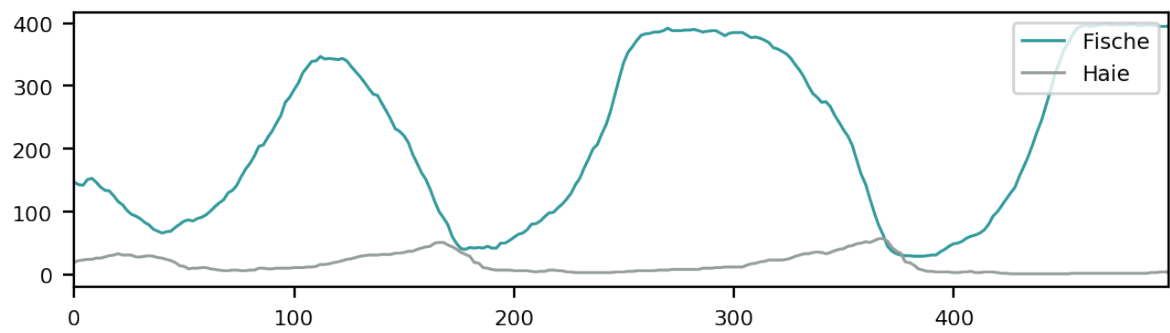
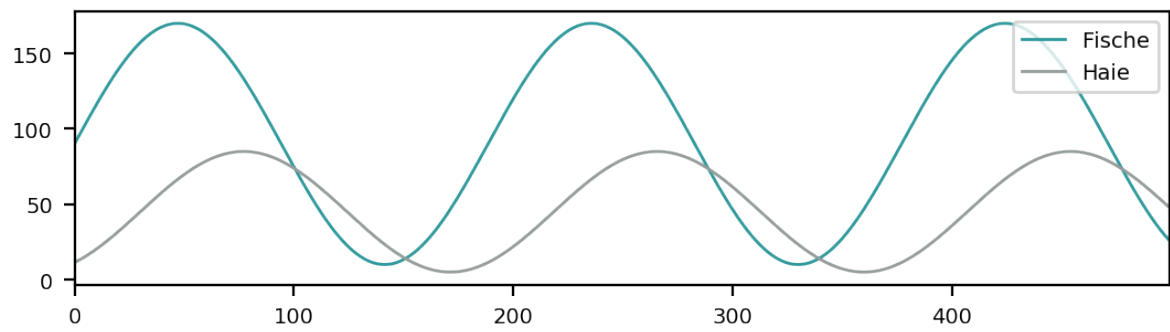


Abbildung 1: Ein typischer Tag auf WATOR

Süden oder Westen bewegen. Falls jedoch bereits alle benachbarten Zellen durch die eigene Spezies besetzt sind, so entfällt diese Regeln. Die Wahl der Bewegungsrichtung ist für Fisch ganz einfach: Wähle zufällig. Haie haben aufgrund ihrer Natur als Jäger jedoch ein wenig mehr zu beachten. Es gilt: Befindet sich in einer benachbarte Zellen ein oder mehrere Fische, so wähle eine dieser Zellen und bewege dich dort hin – der Fisch wird dabei gefressen. Falls diese Regel nicht anwendbar ist, so verhalte dich wie ein Fisch und wähle zufällig.

Zusätzlich zu diesen Bewegungsregeln dürfen sich unsere WATOR Bewohner unter bestimmten Voraussetzungen vermehren. Für unsere Fische führen wir dafür einen zusätzlichen Parameter **fbreed** ein, welcher die Zeit – also die Anzahl der Chronen – angibt die die Fische am Leben sein müssen bevor sie sich vermehren dürfen. Ganz analog verwenden wir für die Haie den Wert **hbreed** und noch einen zusätzlich Parameter **starve** der angibt wie viele Chronen ein Hai überleben kann ohne gefressen zu haben. Für eine erfolgreiche Simulation ist es schlussendlich noch notwendig initiale Parameter für den Anbeginn der Zeit, also $t = 0$ zu wählen. Das sind genau die Anzahl der Fische und Haie und deren Position auf WATOR. Die Positionen werden wir in unserer Simulation ganz einfach zufällig wählen, und für die initiale Anzahl von Fischen und Haien verwenden wir die Parameter **nfish** und **nshark**.

Von den idealisierten Lotka-Volterra Gleichungen würden wir uns Sinusähnliche Populationsverläufe erwarten. In der Realität sieht das ganze jedoch naturgemäß komplizierter aus, insbesondere wenn wir nur einen kleinen Planeten simulieren spielt der Zufall eine große Rolle.



2 Das stetige WATOR Programm

2.1 Schwarmverhalten

3 Animationen

Abbildungsverzeichnis