Lotka-Volterra-Model mit ressourcenabhängigem Produzenten

Gleichungen

$\frac{dS}{dt} = s_{in}(t) - b \cdot S \cdot P + g \cdot K$	Resource (Substrat)
$\frac{dP}{dt} = c \cdot S \cdot P - d \cdot K \cdot P$	Prey (Produzent)
$\frac{dK}{dt} = e \cdot P \cdot K - f \cdot K$	Predator (Konsument)
wobei:	
$s_{in}(t) = s_{in} + f_{pulse}(t, s_{start}, s_{duration}, s_{pulse})$	Substratimport

Hierbei kontrolliert c die Wachstumsrate des Produzenten, d und e beschreiben die Interaktion zwischen Produzenten und Konsumenten (Gefressenwerden bzw. Fraß) und f die Mortalität des Konsumenten.

Die zusätzliche Gleichung dS/dt beschreibt eine für das Wachstum des Produzenten notwendige Ressource (Substrat). Der Parameter b steht für den Verbrauch des Substrats durch Wachstum des Produzenten, der Parameter g ermöglicht die Simulation eines Substratrecyclings.

Außerdem beschreibt die zeitabhängige Funktion $s_{in}(t)$, eine impulsförmige Zugabe des Substrates als Belastungsimpuls, wobei Höhe (`s_pulse`), Zeitpunkt (`s_start`) und Zeitdauer (`s_duration`) verändert werden können.

Aufgaben

Ziel ist es, wesentliche Eigenschaften des Modells kennenzulernen. Hierzu könnte man das Modell mathematisch analysieren oder das Modell rein spielerisch kennen lernen. Der folgende Ansatz versucht letzteres und gibt ein paar Anregungen.

Wichtig: Protokollieren Sie genau, welche Auswirkungen die Parameteränderungen auf Amplitude, Periodendauer und Form der Zeitreihen haben. Plotten Sie zusätzlich zum Zeitreihenverlauf auch Zustandsdiagramme.

Gleichgewichte

Gegeben sind zunächst die Standardparameter und Startwerte wie folgt: $b = 0, g = 0, c = d = e = f = 0.1, S_0 = 1, P_0 = 1, K_0 = 0.5, S_{in} = 0$.

- Schreiben Sie die Gleichungen des Modells so auf, dass alle "unnötigen" Terme wegfallen.
- Verändern Sie den Startwert für den Konsumenten (K_0) auf 2 bzw. auf 1. Was passiert?
- Setzen Sie nochmals $K_0 = 2$. Versuchen Sie ein c (Wachstumsrate des Produzenten) zu finden, bei dem wieder ein Gleichgewicht auftritt.

Periodendauer und Amplitude

- Stellen Sie das Standardszenario wieder her! Wie müssen die Raten c, d, e, f verändert werden (mit $c = d = e = f$), damit sich die Schwingungsfrequenz:
  - erhöht: $c = d = e = f =?$
  - erniedrigt: $c = d = e = f =?$
  - Warum wirken die Parameter in die jeweils beobachtete Richtung?
- Setzen Sie wieder die Standardparameterwerte ein.
  - Verdoppeln Sie die Wachstumsrate c auf 0.2. Wie ist das Ergebnis zu erklären?
  - Verdoppeln Sie die Verlustrate von K auf 0.2. Wie ist das Ergebnis zu erklären?

- Welche Folgerungen lassen sich zu **Umsatz** und **Populationsdichte** in realen Systemen treffen?

- Verdoppeln und halbieren Sie den Umsatz zwischen P und K. Verwenden Sie dazu die Parameter: $d = e = 0.05 \ldots 0.2$.

Ressourcenabhängiges Wachstum

Stellen Sie zunächst wieder die Standardparameterwerte ein.

- Setzen Sie den Parameter für Ressourcenverbrauch $b = 0.01$
  - Schreiben Sie die nun gültigen Zustandsgleichungen auf!
  - Was passiert hinsichtlich:
    - Amplitude?
    - Schwingungsfrequenz?
  - Warum wirken die Parameter in diese Richtung?
- Erhöhen Sie nun die Verbrauchsrate b und protokollieren Sie die Änderungen.
- Warum kommt es nicht zum Aussterben des **Produzenten**? Wie müsste das Gleichungssystem aussehen, damit ein realistischeres Verhalten erzeugt wird?

Regenerierbare Ressource

- Setzen Sie wieder die Standardparameterwerte ein und setzen Sie $b = 0.1$.
- Setzen Sie nun auch die Importrate des Substrates auf einen konstanten Wert $S_{in} = 0.1$. Was passiert?
- Erhöhen Sie nun den Import S_{in} auf 0.2. Gibt es eine Analogie in natürlichen Systemen?

Belastungsimpuls

- Stellen Sie ein gedämpftes System auf (Standardszenario, jedoch $S_{in} = 0.2, b = 0.2$). Simulieren Sie nun einen Belastungsstoß zum Zeitpunkt $s_{start} = 100$ mit der Stärke $s_{pulse} = 0.5$ und $s_{duration} = 1$ und beurteilen Sie das Verhalten von Produzenten und Konsumenten.
- Simulieren Sie folgendes Szenario: $S_{in} = b = 0.01, c = 0.1, d = e = f = 0.2$.
- Wie wirkt sich ein Belastungsstoß auf ein System im Gleichgewicht aus?

Zusatzaufgabe

- Gibt es eine Möglichkeit, ein schwingendes System durch einen externen Impuls in einen Zustand nahe dem Gleichgewichtszustand zu bringen? Experimentieren Sie mit der Höhe und dem Zeitpunkt des Impulses.
- Von welchen Faktoren hängt die Wirksamkeit bzw. Wirkungsrichtung eines Störimpulses ab?

Lösungen

Die Lösung benutzt die Programmiersprache R und das Add-On-Paket `deSolve` (Soetaert, Petzoldt and Setzer, 2010). Alternativ kann auch ein anderes Programmpaket oder eine Webanwendung genutzt werden.

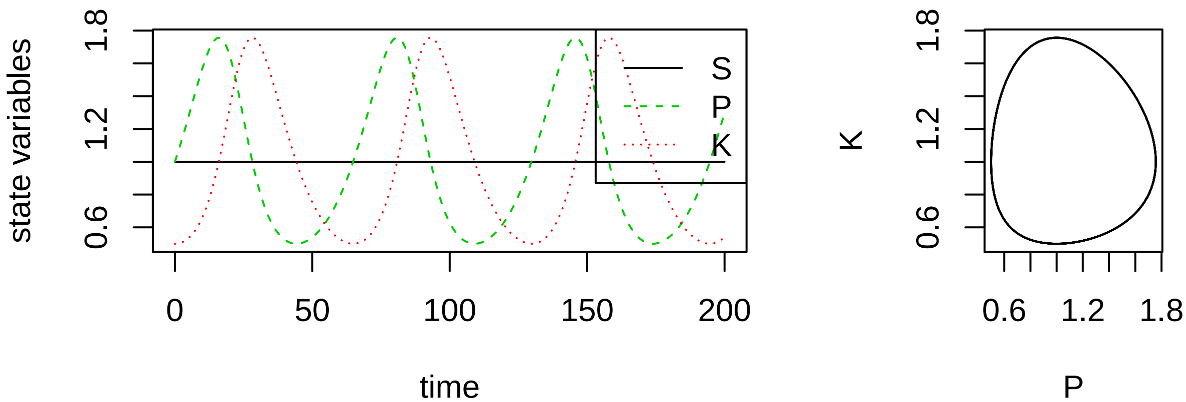
Das Modell besteht aus einer Funktion `lv` mit den Differentialgleichungen und einer Hilfsfunktion für einen Ressourcenimpuls sowie den vordefinierten Parametern für das Standardszenario.

```r
library(deSolve)

lv <- function (t, x, parms) {
  with(as.list(c(x, parms)),{
    ds <- s_in - b * s * p + g * k + pulse(t, s_start, s_duration, s_pulse)
    dp <- c * s * p - d * k * p
    dk <- e * p * k - f * k
    list(c(ds, dp, dk))
  })
}

pulse <- function(t, s_start, s_duration, h_pulse) {
  if ((s_start <= t) & (t < (s_start + s_duration))) h_pulse else 0
}

times  <- seq(0, 200, by=0.1)
parms  <- c(s_in=0, b=0, c=0.1, d=0.1, e=0.1, f=0.1, g=0, s_start=0, s_duration=0, s_pulse=0)
init   <- c(s=1, p=1, k=0.5)
```

```r
runSimulation <- function() {
  res <- ode(init, times, lv, parms, method="adams")
  layout(matrix(c(1,2), nrow=1, byrow=TRUE), widths=c(2, 1))
  matplot(res[,1], res[,2:4], type="l", col=c(1,3,2), xlab="time", ylab="state variables")
  legend("topright", legend=c("S", "P", "K"), col=c(1, 3, 2), lty=1:3)
  plot(res[,3], res[,4], type="l", xlab="P", ylab="K")
}
```

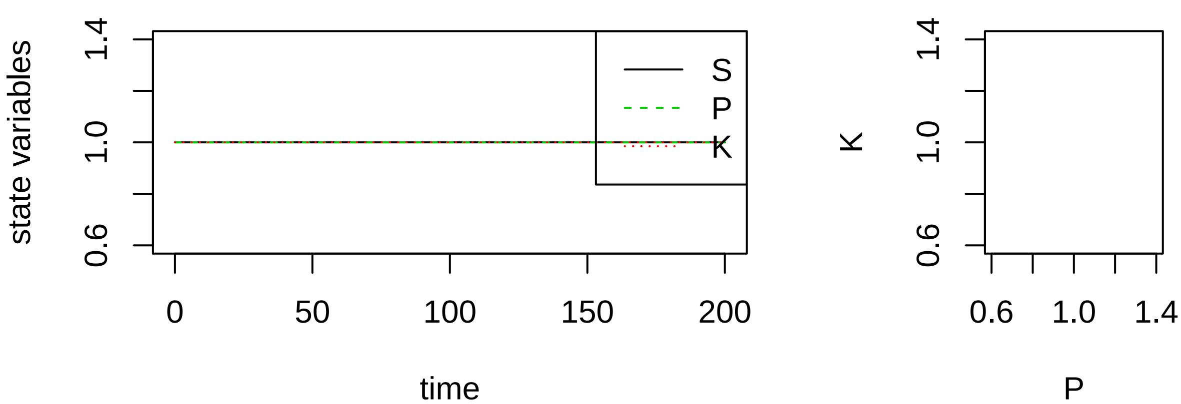
Standardszenario

```
parms  <- c(s_in=0, b=0, c=0.1, d=0.1, e=0.1, f=0.1, g=0, s_start=0, s_duration=0, s_pulse=0)
init   <- c(s=1, p=1, k=0.5)
runSimulation()
```

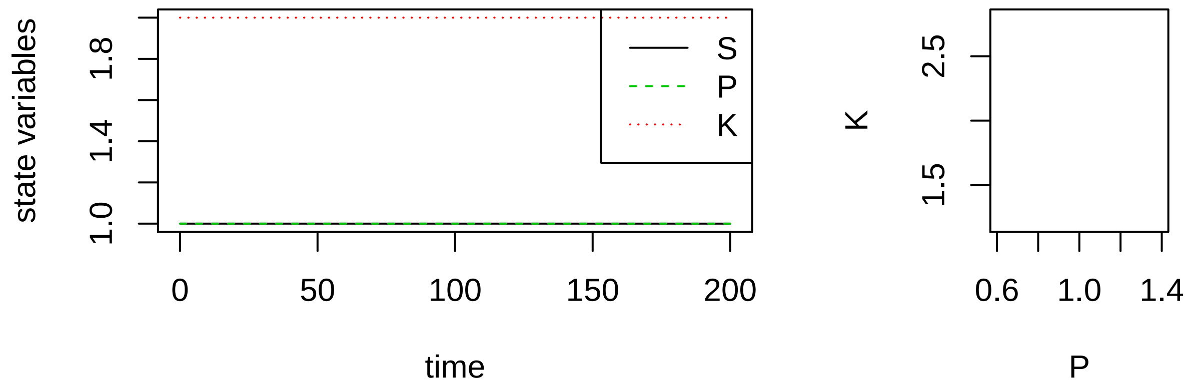


Gleichgewichte

```
parms  <- c(s_in=0, b=0, c=0.1, d=0.1, e=0.1, f=0.1, g=0, s_start=0, s_duration=0, s_pulse=0)
init   <- c(s=1, p=1, k=1)
runSimulation()
```

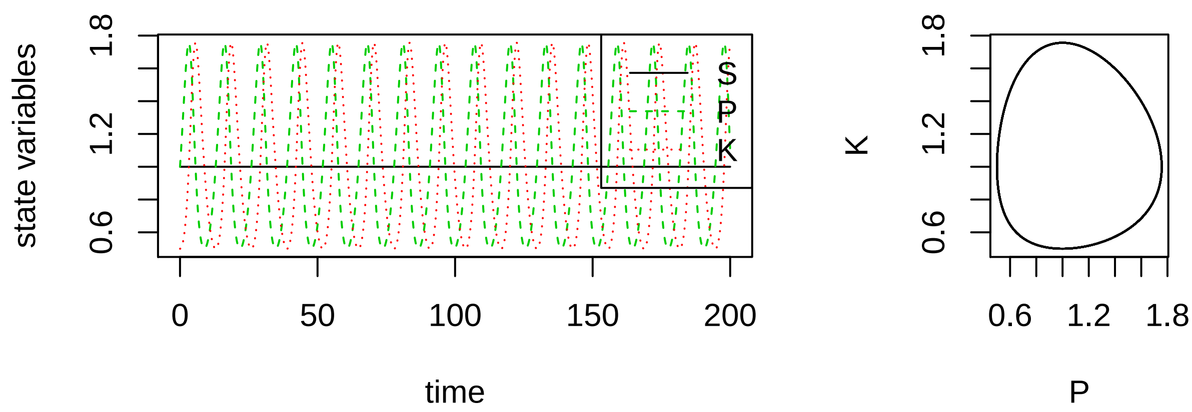



```
parms  <- c(s_in=0, b=0, c=0.2, d=0.1, e=0.1, f=0.1, g=0, s_start=0, s_duration=0, s_pulse=0)
init   <- c(s=1, p=1, k=2)
runSimulation()
```



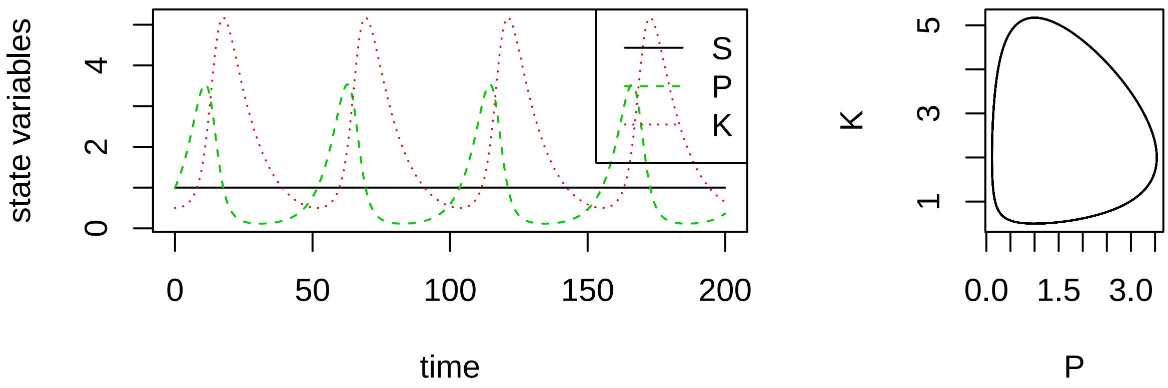
Verkürzte Periodendauer

```
parms  <- c(s_in=0, b=0, c=0.5, d=0.5, e=0.5, f=0.5, g=0, s_start=0, s_duration=0, s_pulse=0)
init   <- c(s=1, p=1, k=0.5)
runSimulation()
```



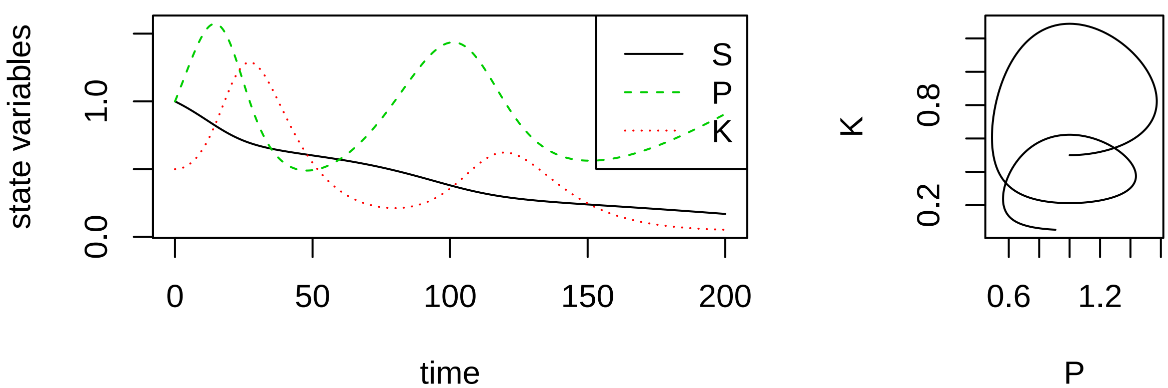
Auf den Umsatz kommt es an

```
parms  <- c(s_in=0, b=0, c=0.2, d=0.1, e=0.1, f=0.1, g=0, s_start=0, s_duration=0, s_pulse=0)
init   <- c(s=1, p=1, k=0.5)
runSimulation()
```



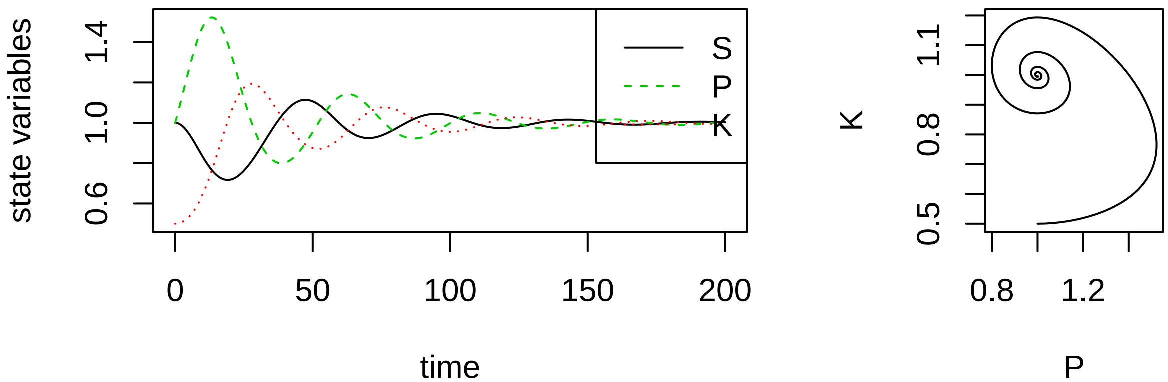
Ressourcenverbrauch

```
parms  <- c(s_in=0, b=0.01, c=0.1, d=0.1, e=0.1, f=0.1, g=0, s_start=0, s_duration=0, s_pulse=0)
init   <- c(s=1, p=1, k=0.5)
runSimulation()
```



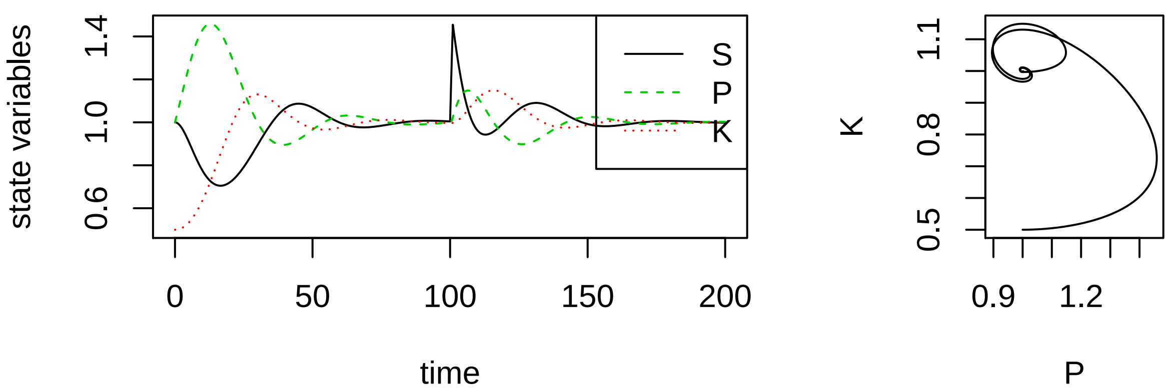
Regenerierbare Ressource

```
parms  <- c(s_in=0.1, b=0.1, c=0.1, d=0.1, e=0.1, f=0.1, g=0, s_start=0, s_duration=0, s_pulse=0)
init   <- c(s=1, p=1, k=0.5)
runSimulation()
```



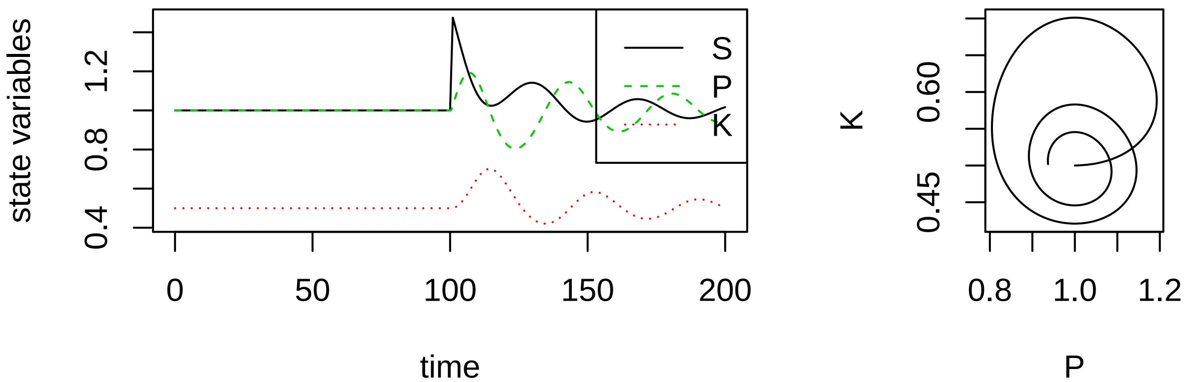
Belastungsimpuls

```
parms  <- c(s_in=0.2, b=0.2, c=0.1, d=0.1, e=0.1, f=0.1, g=0, s_start=100, s_duration=1, s_pulse=.5)
init   <- c(s=1, p=1, k=0.5)
runSimulation()
```



Gestörtes Gleichgewicht

```
parms  <- c(s_in=0.1, b=0.1, c=0.1, d=0.2, e=0.2, f=0.2, g=0, s_start=100, s_duration=1, s_pulse=.5)
init   <- c(s=1, p=1, k=0.5)
runSimulation()
```



Literatur

Hindmarsh, Alan C. (1983) ODEPACK, A Systematized Collection of ODE Solvers. In: Stepleman, R.W. et al. (ed.) Scientific Computing, p.55-64. North-Holland, Amsterdam.

Soetaert, Karline; Petzoldt, Thomas and R. Woodrow Setzer (2010) Solving Differential Equations in R: Package deSolve. Journal of Statistical Software, 33(9), 1-25. http://www.jstatsoft.org/v33/i09/

Volterra, V. (1926) Fluctuations in the Abundance of a Species considered Mathematically. Nature 118, 558-560.

ThPe