

# Anleitung für die Simulationssoftware

## **Grundsätzliches**

Die von mir entwickelte Simulationssoftware ermöglicht Modellbildung und Simulation dynamischer Systeme nach dem Konzept der „**System Dynamics**“.

## **Programmiersprache und Grafiktool**

Da der erste Anstoß zu der Entwicklung von einem Einsatz im Physikunterricht und Programmierung mit Python ausging, entschied ich mich damals, nicht das von mir sonst bevorzugte Grafiktool wxPython, sondern das bei Standardinstallationen von Python mit gelieferte Tkinter zu verwenden. Das hatte gravierende Folgen, was die Klarheit eines objektorientierten Ansatzes anging und die tag- (nicht der Tag, sondern das englische *tag*) und ID-Orientierung ist sehr gewöhnungsbedürftig und die Dokumentation von Tkinter unbefriedigend. Allerdings stellte sich die Orientierung an tag und ID in der Umsetzung aber auch als durchaus passend heraus.

**Zusammengefasst: Sie benötigen nur eine aktuelle Standardinstallation von Python.**

## **Zur Verfügung stehende Komponenten**

Neben der eigentlichen Modellierungs- und Simulationssoftware stehen weitere Komponenten zur Verfügung, die alle auf der selben Basis entwickelt sind:

- Eine Software zum Erstellen von Wirkungsdiagrammen
- Eine Software zum (interaktiven!) Konvertieren der Wirkungsdiagrammdateien in Simulationsdateien (Was sind jeweils Bestandsgrößen, Flüsse, Parameter oder Zwischengrößen)
- Software zur Demonstration des Simulationsverlaufs bei Einsatz von Euler-Cauchy und Runge-Kutta<sup>1</sup>.

## **Historie**

System Dynamics, Simulation dynamischer Systeme haben internationale Aufmerksamkeit in der Öffentlichkeit durch die Veröffentlichungen des „*Club of Rome*“ (1968) mit dem deutschen Titel „*Die Grenzen des Wachstums*“ (1972) erzielt. Die zu Grunde liegenden Verfahren sind lange bekannt, sind aber erst mit der Entwicklung großer Computeranlagen ab der Zeit umfassend eingesetzt worden. Sie werden inzwischen weit verbreitet eingesetzt, wenn es darum geht, die zeitliche Entwicklung von Systemen zu modellieren, um sie zu verstehen und fundierte Aussagen über zukünftige Entwicklungen zu machen.

---

<sup>1</sup> Mathematisch: Numerische Lösung von Differentialgleichungssystemen mit Anfangswerten.

## Das Simulationstool

Sie starten das Simulationsprogramm über die Datei **SimulationApp.py**. Sie startet zwei Fenster:

- das Modellentwicklungsfenster mit dem Titel „*Entwicklung Simulation*“ und
- das Simulationsfenster mit dem Titel „*Grafikfenster*“.

## Das Fenster für die Modellentwicklung

Die Entwicklung läuft weitgehend über die Menüs, allerdings gibt es neben einigen Mausaktionen zu vielen Aktionen Kurztasten. Siehe dazu im Menü „*Hilfe zu Kurztasten*“.

### Datei-Menü

Keine besonderen Aktionen, allerdings hier nur ein Hinweis, dass für die Modelldateien (Dateinamensergänzung `.simdia`) der Ordner Modelle vorhanden sein sollte.

### Menü Modellentwicklung

Über dieses Menü können die verschiedenen Elemente der Modellierung erzeugt werden.

Vielleicht nicht ganz konsequent verwende ich parallel die Bezeichnungen „Größe“ und „Faktor“. Bestand, Bestandsgröße und **Bestandsfaktor** sind also synonym.

Bei der Entwicklung muss bereits festgelegt werden, wenn eine Größe ein **Fluss** (Flussfaktor) ist.

**Parameter** (Inputfaktoren) sind Größen, die Eingabewerte liefern. Sie können konstant sein, aber auch veränderliche Werte sind möglich.

In **Zwischengrößen** werden mehrere Eingabewerte verrechnet, um sie an andere Größen weiter zu geben.

**Wirkungen** werden dargestellt, müssen aber klar von Flüssen unterschieden werden. Ihre Ausgangswerte müssen in den **Termen** (Formeln) der Zielwerte auftreten.

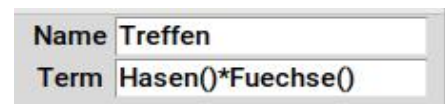
Beim Einbau einer Wirkung mit Hilfe der Maus müssen Sie darauf achten, die linke obere Ecke der Ausgangsgröße anzuklicken und dann zur linken oberen Ecke des Ziels zu ziehen. Achten Sie darauf, dass Sie vorher **Wirkung mit Maus** angeklickt haben, da Sie sonst die Ausgangsgröße verschieben.

### Dialogfenster

Zu den Menüpunkten vieler Menüs werden Dialogfenster geöffnet. Beim Erstellen einer neuen Bestandsgröße muss beispielsweise zunächst der Name und dann ein Anfangswert angegeben werden.

Achten Sie bei der Vergabe von **Namen** darauf, mit Python konforme Namen zu verwenden, da zu den Namen entsprechende Pythonobjekte erzeugt werden.

Weiterhin müssen in Termen die Eingangsgrößen von Flüssen und Zwischengrößen immer hinter ihrem Namen ein **Paar von öffnender und schließender Klammer** aufweisen, da nur dann der Term von Python richtig



ausgewertet wird. Im Beispiel wird mit **Hasen( ) \* Fuechse( )** jeweils der aktuelle Wert der Größe **Hasen** mit dem aktuellen Wert der Größe **Fuechse** multipliziert.

Wenn Sie in die Klammer eine Zahl eingeben, wird der um diese Zahl verzögerte Simulationswert der Größe verwendet. Beispiel **Hasen(3)**.

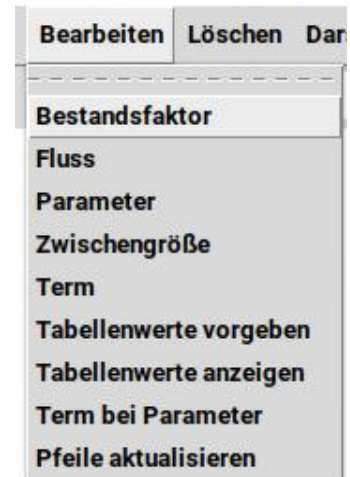
### Menü Bearbeiten

Bearbeiten von Bestandsfaktor, Fluss, Parameter  
Zwischengröße, Term und Term bei Parameter sollte selbst erklärend sein.

Bei Parametern können die Werte mit **Tabelle vorgeben** definiert werden. Dazu müssen in mehreren Schritten die Paare von Zeitpunkt und zugehörigem Wert eingegeben werden.

Der Zustand kann mit **Tabelle anzeigen** gezeigt werden.

Nach Verschiebungen im Diagramm sind Wirkungspfeile teilweise nicht mehr aktuell. Sie können mit **Pfeile aktualisieren** auf den neuen Stand gebracht werden.



### Menü Löschen

Die Punkte in diesem Menü sollten auch selbst erklärend sein.

### Menü Darstellung

Der Menüpunkt **Faktoren positionieren** ermöglicht es, die Faktoren im Fenster auf eine exakte Position durch Angabe der Koordinaten zu verschieben. Die aktuellen Positionen werden im Auswahlfenster angezeigt.

**Modelldaten zeigen** stellt die Werte in einem Textfenster dar.

**Modelldiagramm speichern** ermöglicht das Abspeichern des Diagramms in eine postscript – Datei (Dateiendung .ps). Um diese in ein anderes Dateiformat zu konvertieren verwenden Sie bitte ein externes Programm. Unter Windows kann beispielsweise das kostenlose Programm IrfanView (extensions installieren!) dies leisten.

### Menü Hilfe

Dieses Menü bietet Hilfen zur Modellentwicklung, Hilfe zu den Dateien und zu den Kurztasten.

### Mausaktionen

Mausaktionen auf der Simulationsoberfläche sind an die oben beschriebenen *tags* gekoppelt, mit denen eine Zuordnung für die Ereignisbehandlung möglich wird.

Mit der **linken Maustaste** kann man die Symbole ziehen.

Die **rechte Maustaste** über der linken oberen Ecke eines Symbols öffnet jeweils ein Konfigurationsfenster.

Eine weitere Aktion, Klick mit der **rechten Maustaste**, ermöglicht es, die Bogenorientierung einer Wirkung zu wechseln.

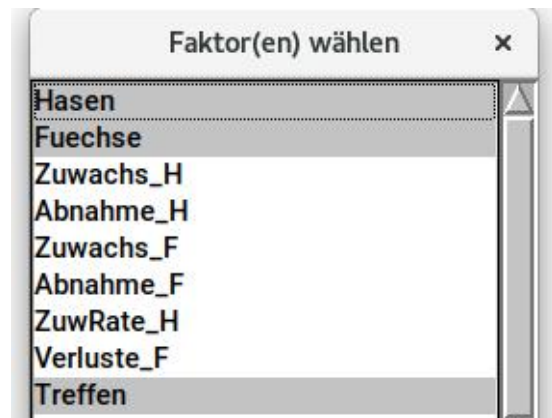
## Das Fenster für die Simulationsdurchführung

### Menü Simulation

Um die Entwicklung der Größen darstellen zu können, müssen Sie zunächst auswählen, welche der Größen (Faktoren) verwendet werden sollen. Halten Sie dazu die Taste Strg (Ctrl) gedrückt und wählen mit der linken Maustaste die gewünschten Größen aus.

Im Beispiel wurden so die Bestandsgrößen **Hasen**, **Fuechse** und die Zwischengröße **Treffen** ausgewählt.

Wenn Sie nun auf den Button **Berechnen** klicken, wird die Simulation mit den im Fenster gezeigten Werten und der vorgegebenen Simulationsmethode „erweitert Euler“ (entspricht Runge-Kutta-2) verwendet. Im Beispiel erhält man die abgebildete Darstellung.

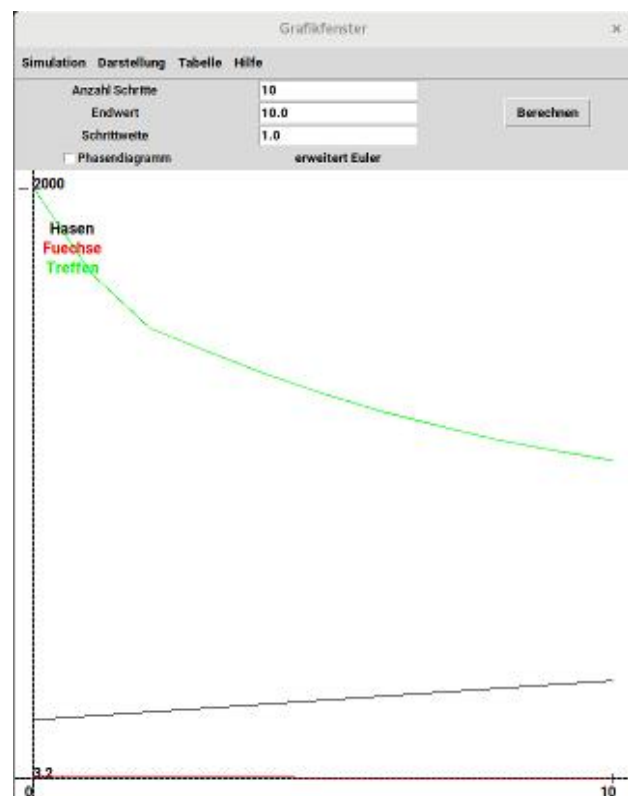


### Weiter im Menü Simulation

Sie können die gewünschte Simulationsmethode wählen, also **Euler**, **erweitert Euler** oder **Runge-Kutta (4)**. Bei diskreter Simulation (keine Zwischenwerte sinnvoll) ist das einfache Euler-Verfahren zwingend. Runge-Kutta gelingt es in der Regel am besten auch schwierige Verläufe gut numerisch zu generieren.

### Menü Darstellung

Sie können die **Fenstergröße ändern**, die **Werte für die y-Achse setzen** und entsprechend dem Modelldiagramm das **Verlaufdiagramm speichern**.



### MenüTabelle

**Tabellenwerte anzeigen** stellt die aktuellen Werte zum Simulationsfenster in einem Tabellenfenster dar, während

**Tabelle in csv-Datei** die Möglichkeit bietet, die Daten in eine Datei zu schreiben. In dieser Form können sie von Tabellenkalkulationsprogrammen angezeigt werden.

### Menü Hilfe

Selbst erklärend

### Textfelder

Auch die Textfelder sollten selbst erklärend sein. **Anzahl Schritte**, **Endwert** und **Schrittweite** sind von einander abhängende Werte, so dass sich Veränderungen bei den jeweils anderen ergeben.

Ändert man als beispielsweise allein die Anzahl Schritte, ergibt sich durch die unveränderte Schrittweite ein anderer Endwert.

Wenn andererseits die Schrittweite geändert wird und der Endwert bleibt, muss sich die Anzahl der Schritte ändern.

Beachten Sie, dass die Darstellung erst nach dem Anklicken von **Berechnen** erneuert wird.

### **Projekt zum Erstellen von Wirkungsdiagrammen**

Starten Sie die Datei **wirkungsdiagrammApp.py**.

Wegen der fehlenden Ausführung gibt es hier nur ein einzelnes Fenster. Der Menüaufbau ähnelt dem Simulationsprogramm.

#### **Datei-Menü**

Keine besonderen Aktionen, Modelldateien (Dateinamensergänzung **.wirkdia**) im Ordner Modelle.

#### **Menü Neu**

Über **Neues Modell** kann zusätzlich der Modellname festgelegt werden. Eingefügt können nur **Faktoren** (Größen) und **Wirkungen** zwischen ihnen, die ebenfalls wieder mit der Maus (linke obere Ecke des Faktors jeweils und mit der Maus ziehen) eingebaut werden können.

#### **Menü Bearbeiten**

**Faktor bearbeiten** zum Ändern des Namens. **Wirkung bearbeiten** wechselt zwischen verstärkender (gleichsinniger) bzw dämpfender (gegensinniger) Wirkung, die mit + bzw – im Wirkungspfeil gekennzeichnet werden. Den selben Effekt erzielen Sie mit **Linksklick** auf das Pfeilsymbol, während **Rechtsklick die Bogenorientierung** wechselt.

#### **Menü Löschen**

Sollte selbst erklärend sein.

#### **Menü Darstellung**

**Modelldaten zeigen** stellt die Werte in einem Textfenster dar.

**Modelldiagramm speichern** ermöglicht das Abspeichern des Diagramms in eine postscript – Datei (Dateiendung **.ps** ; dazu siehe S.3).

#### **Menü Hilfe**

Sollte selbst erklärend sein.

### **Projekt zum Exportieren von Wirkungsdiagrammen**

Starten Sie die Datei `export_WirkungsdiagrammApp.py`. Auch hier wird nur ein Fenster benötigt.

#### **Begründung**

Wirkungsdiagramme lassen sich nicht direkt verwandeln, da das Simulationsprogramm zu den einzelnen Größen (Faktoren) ihre Einordnung als Bestand, Fluss, Parameter oder Zwischengröße benötigt.

#### **Menü Datei**

Das Programm erfüllt allein die Aufgabe, Wirkungsdiagramme zu geladenen Dateien (Dateiendung `.wirkdia`) so zu bearbeiten, dass die entsprechenden Simulationsdateien (Dateiendung `.simdia`) gespeichert werden können. Die Darstellung bleibt dabei im Format des Wirkungsdiagramms.

#### **Menü Bearbeiten**

Dazu stellt das Menü die Punkte

- **Faktor ist Bestandsfaktor**
- **Faktor ist Flussfaktor**
- **Faktor ist Parameter**
- **Faktor ist Zwischengröße**

zur Verfügung.

Zusätzlich werden **Faktornamen ändern** und **Wirkung bearbeiten** (s.o.) angeboten

#### **Menü Neu**

Das Menü ermöglicht das Hinzufügen von Faktoren und Wirkungen.

#### **Menü Löschen**

Entsprechend das Löschen von Faktoren und Wirkungen.

#### **Menü Darstellung**

**Modelldaten zeigen** stellt den bearbeiteten Zustand in einem Textfenster dar. Das ist daher wichtig, weil die grafische Darstellung nicht die Einordnung der Faktoren darstellt.

#### **Menü Hilfe**

Sollte selbst erklärend sein.