Vorwort

(= persönliches)

In der Forschung aber auch zu Unterrichtszwecken werden viele Theorien vermittelt und Simuliert. Solche Simulationen basieren meistens auf einem strikt mathematischen Hintergrund, meinst Integrale erster Ordnung. Forschende Personen im Bereich Physik / Biologie verwenden dabei bevorzugt eine Modellierungssoftware, da es nicht ihr Fachgebiet ist selbst zu programmieren. Andreas Butti hat sich bei der Verwendung von Simulationstools über deren Plattformabhängigkeit und Benutzerunfreundlichkeit gestört. Als Informatiker kommt man da schnell in Versuchung selbst etwas besserer zu schreiben. Nach einem Gespräch mit dem Physikdozenten, Herr Scheidegger, wurde daraus dann diese BA, die jedoch nicht nur ein Benutzerfreundliches Simulationswertzeug sein soll, sondern auch bisher nicht vorhandene Möglichkeiten für die Simulationen von Biologischen Abläufen, wie das innere einer Zelle, darstellen soll. TODO: Danksagung

2. Methode (Graph.)

→ Darstellung der Modelle

→ Numerik

→ Programmiersprache und Umgebung

Problemstellung aus der Natur (Biologie, Medizin) Technik

↓

Aufgabe: Modelleditor

↓

Was existiert?

↓

Aufgabenstellung: Ziele bzw. was isch nöti

Ziel ist (somit) die Entwicklung eines ...

Mit unserer Simulation ist es zusätzlich möglich in einem XY Modell mehrere Meso Kompartmente abzubilden, ein Meso Kompartment ist das vorhin genannte Flussmodel. Diese Meso Kompartmente können sich wären der Simulation im XY-Raum bewegen. Es können Dichten angegeben werden, die über den XY Raum verteilt sind, und die Meso Kompartmente können an Ihrer aktuellen Position von der dichte Konsumieren oder dichte Produzieren, somit kann die Umgebung beeinflusst werden. Mit diesen Fähigkeiten ist es möglich das innenleben einer Zelle oder andere Biologische Prozesse einfach, grafisch abzubilden. ~~Die Idee und Vorgabe dieser Simulationsmethode stammt von Herr Scheidegger, und wurde zusammen mit Herr Füchslin und uns ausgearbeitet.~~

Methode

2 Numerische Lösungsverfahren

Um Biologische Prozesse zu verstehen, nachzubilden und zu entwickeln, werden mathematische Systeme modelliert und danach mit rechnerunterstützten Solvern Näherungslösungen berechnet. Als Beispiel kann sich die Population eines Algenvolkes in einem See mit der Zeit ändern, je nachdem welche äussere Einflüsse auf die Algen einwirken. Die Geburten- und Sterberate bilden die Änderung der Population über die Zeit. Dabei werden oftmals Differentialgleichung verwendet.[Sch11]

Darstellung Graphisch

In diesem Abschnitt wird eine Einführung in die numerischen Lösungsverfahren beschrieben, die ein Modelleditor wie unser Simulations-Tool ausführt um eine approximierte Lösung zu erhalten. In unserer Simulation haben wir verschiedene numerische Verfahren implementiert, die ~~wir nachfolgen kurz behandeln werden.~~ (im Folgenden kurz vorgestellt werden)

2.1 Analytisch oder Numerisch

[Kos94]

Umsetzung Numerik

2.2 Gewöhnliche Differentialgleichungen

Eine Gewöhnliche Differentialgleichung (engl. Ordinary Differential Equation ODE) ist eine mathematische Gleichung, die Ableitungen, die Funktion selbst sowie die unabhängige Variable enthalten kann. Ableitungen und die Funktion selbst treten nach genau einer unabhängigen Variable auf. Lösung $y(t)$ einer Differentialgleichung $y^{(n)}$ ist eine Funktion, die mit ihren Ableitungen deckungsgleich mit der Differentialgleichung selbst ist.

Programmiertechn.

$$y^{(n)} = f\left(t,\, y(t),\, y'(t),\, \ldots,\, y^{(n-1)}(t)\right) \tag{2.1}$$

Auf analytischem Weg kann eine Differentialgleichung durch Integration erfolgen.

$$\int y^{(n)} \cdot dt = y^{(n-1)} + C \tag{2.2}$$

Die Fallbeschleinigung in Abbildung ??? soll in Gleichung 2.3 als Beispiel gezeigt werden. Wir kennen die Beschleunigungs-Funktion $a(t)$, suchen die Weg-Funktion $s(t)$, finden durch Integration eine Lösung und kommen durch abermaliges Differenzieren wieder auf die Ursprüngliche Beschleunigungs-Funktion zurück.

scst: schlechtes Beispiel -> welches dann?

typisch biol. Beispiel $\frac{dN}{dt} = \alpha(c) \cdot N - \beta N^2 = f(N, c)$

$\frac{dc}{dt} = f(N, t)$

oder so

$\dot{y} \to \frac{\partial y}{\partial t}$

Δt

$$
\begin{aligned}
y' &= f\left(t,\, y\left(t\right)\right) \\
y_{n+1} &= y_n + h \cdot f\left(t_n,\, y_n\right)
\end{aligned}
\tag{2.6}
$$

In unserem Beispiel mit der Fallbeschleunigung müssen wir das Verfahren zwei Mal anwenden pro Zeitschritt, da wir eine Differentialgleichung 2. Ordnung lösen möchten. Dabei müsen die Anfangswerte v_0 und s_0 sowie die Schrittweite h bekannt sein.

scst: kürzen -> warum?

Grundidee $dN \to \Delta N \to$ Umformung $\to$ Euler

$\to$ ...

$\to$ ...

$$
\begin{aligned}
h &= 0.1 \\
v_0 &= 15 \\
s_0 &= 0 \\
v_1 &= 15 + 0.1 \cdot (-9.81) = 14.019 \\
s_1 &= 0 + 0.1 \cdot [15 + 0.1 \cdot (-9.81)] = 1.4019
\end{aligned}
\tag{2.7}
$$

Bei Differentialgleichungen 1. Ordnung und kleiner Schrittweite h erhält man ausreichende Genauigkeit. Unser Beispiel mit 2. Ordnung büst bei jedem Schritt an Genauigkeit ein. Es gibt bessere Verfahrung, die auch Ordnungen höheren Grades mit weniger Genauigkeitsverlust zulassen.

Butcher Tableau

Um weitere Verfahren besser zu Verstehen, wird zuerst eine Tabelle eingeführt. Die Tabelle wurde in den 1960er Jahren von **John Charles Butcher** entwickelt für den besseren Umgang mit dem nachfolgenden Runge-Kutta-Verfahren und nennt sich *Butcher tableau* in Gleichung 2.8. Die Tabelle beinhaltet einen Vektor c_i, eine Matrix $A = (a_{ij})$ und einen Vektor b_i wobei $i, = 1, ..., s,\ j = 1, ..., s$ die Zeilen- und Spalten-Indizes und s die Dimension ist. Verwendet wird nur ein Teil der Matrix A, nähmlich ein Dreieck von a_{21} schräg herunter zu $a_{s(s-1)}$.

Warum, was macht A und b_i

a_{ij}

$$
\begin{array}{c|c} c & A \\ \hline & b^T \end{array}
=
\begin{array}{c|cccc}
c_1 & a_{11} & a_{12} & \dots & a_{1s} \\
c_2 & a_{21} & a_{22} & \dots & a_{2s} \\
\vdots & \vdots & \vdots & & \vdots \\
c_s & a_{s1} & a_{s2} & \dots & a_{ss} \\
\hline
 & b_1 & b_2 & \dots & b_s
\end{array}
=
\begin{array}{c|cccc}
0 & & & & \\
c_2 & a_{21} & & & \\
\vdots & \vdots & \ddots & & \\
c_s & a_{s1} & \dots & a_{s(s-1)} & \\
\hline
 & b_1 & b_2 & \dots & b_s
\end{array}
\tag{2.8}
$$

Reihenfolge

Wie beim Euler-Verfahren in Gleichung 2.6 ergibt sich ein neuer Wert y_{n+1} aus dem alten Wert y_n addiert mit einer festen Schrittweite h multiplitiert mit der Ableitung. Doch beim Runge-Kutta-Verfahren werden statt einer Ableitung verschieden gewichtete Ableitungen $b_i k_i$ aufsummiert, wobei das Gewicht b_i und die Ableitung k_i ist.

$$
y_{n+1} = y_n + h \sum_{i=1}^{s} b_i k_i \tag{2.9}
$$

Eine Ableitung, auch Zwischenschritt genannt, ist in Gleichung 2.10 erläutert. Je grösser die Dimension s ist, desto mehr Zwischenschritte werden berechnet. Zu Beginn jedes Schrittes wird der Vektor k_i zurückgesetzt.

2.5 Gradienten-Verfahren

3 Methode

Text: Für die graphische Darstellung...

Diffusionsgleichung, Meso-> stoff konsumieren / ausscheiden

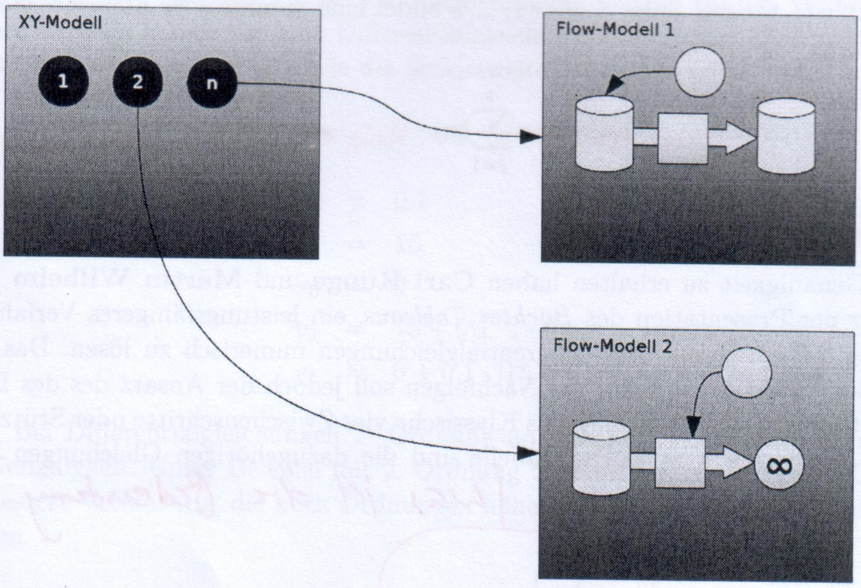

Abbildung 3.1: bli

~~3.1 Legende~~

das ist kein Abschnitt

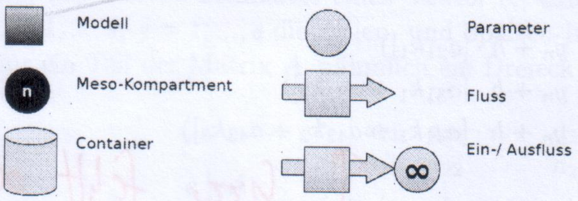

Abbildung 3.2: bla

Es kann entweder ein Herkömmliches «Flow-Modell» erstellt werden, das bereits bekannt ist da es vom Konzept her identisch bereits von vielen Simulationstools angeboten wird. Oder es kann ein «XY-Modell» erstellt werden, das dann Meso Kopartments beinhaltet, diese befinden sich an einer Position (X / Y) und verweisen auf ein Modell («Flow-Model-X»). Im «XY-Modell» könne sich «Dichten» befinden, das sind Stoffe die eine gewisse Konzentration an einer gewissen Stelle aufweisen. Ein Meso Kompartment kann sich wären der Simulation im «XY-Modell» bewegen, es kann Dichten konsumieren oder produzieren.

4 Werkzeuge und Hilfsmittel

Wie ist Lösung und warum diese Lösung?

Methode
- Numerik
- graphische Umsetzung
- Programmiersprache & Entwicklungsumgebung