

Übung 5: Plotten mit Matplotlib

Zielstellung

Im technischen Bereich ist die Visualisierung von Daten ein sehr hilfreiches Werkzeug. Oft werden in Berechnungen Zwischenergebnisse graphisch dargestellt, um diese auf Plausibilität zu überprüfen oder Fehler im Rechengang zu finden. Darüber hinaus steht die Dokumentation wissenschaftlicher Arbeiten und Erkenntnisse aus Experimenten oder Berechnungen im Vordergrund - eine der grundlegenden Aufgaben von Wissenschaftlern und Ingenieuren.

In der Lehrveranstaltung werden Kenntnisse zur 2D-Visualisierung von numerischen Daten mit dem Paket matplotlib vermittelt. Im Ergebnis sind die Teilnehmer in der Lage, Ergebnisse einer Simulationsrechnung in publikationstauglicher Qualität zu plotten und zu speichern (z.B. für die Weiterverwendung in LaTeX oder Word).

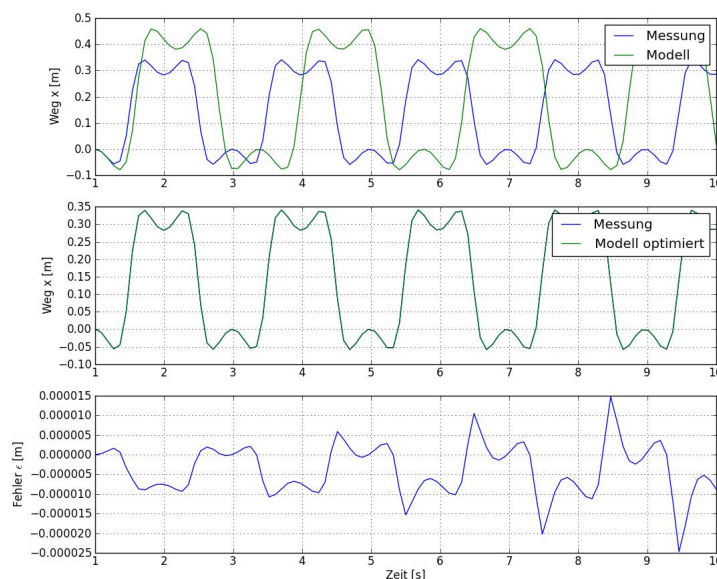
Aufgabe 1

Ergebnisvisualisierung der Identifikationsaufgabe (plotten.py)

Hinweis: es werden jeweils nur die Verläufe der Ergebnisgröße x (Weg der Laufkatze) verwendet.

1. Legen sie eine Liste zum Speichern der Zwischenergebnisse der Optimierung an (in oberster Ebene des Skriptes) und modifizieren Sie die Funktion `'min_target'` so, dass `'res'` in jedem Schritt an diese Liste angehängt wird.
2. Erzeugen Sie eine neue figure mit einem 3x1 Subplot-Layout.
3. Plotten Sie die Daten aus der Ergebnisdatei und das Ergebnis des ersten Optimierungsschrittes in Subplot 1.
4. Plotten Sie die Daten der Ergebnisdatei und das Ergebnis des letzten Optimierungsschrittes in Subplot 2.
5. Plotten Sie den Fehler zwischen Ergebnisdatei und optimiertem Simulationsmodell in Subplot 3
6. Kompletieren sie das Diagramm (Achsenbeschriftungen, Legende, Hilfslinien) und speichern sie es im pdf- und im png-Format ab.

Ergebnis:



Was ist an der Darstellung ungünstig / könnte noch verbessert werden?

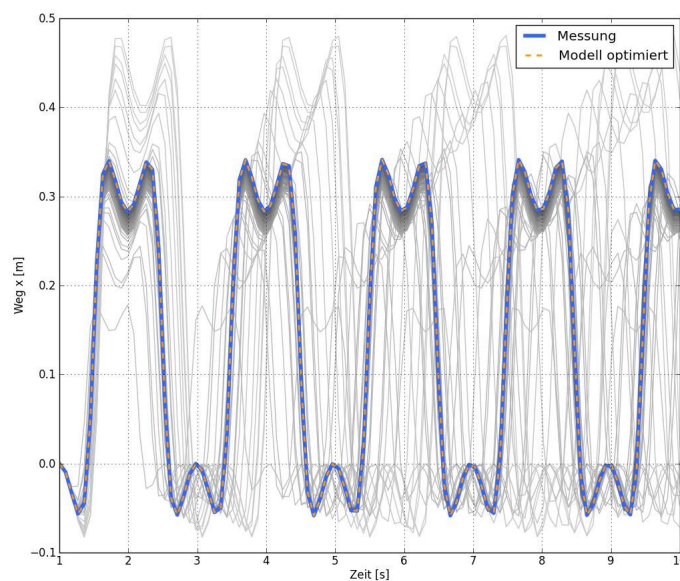
Aufgabe 2

Visualisierung des Verlaufes der Optimierung (plotten.py)

Nun sollen alle Zwischenergebnisse der Optimierung in einem Diagramm dargestellt werden. Um die Lesbarkeit zu gewährleisten, wird ein Farbgradient entsprechend des Optimierungsschrittes vorgesehen.

1. Legen Sie eine neue Figure samt Subplot an (1x1 Layout)
2. Erzeugen Sie einen Plot für jedes Zwischenergebnis bis auf das Endergebnis und weisen Sie einen Grauton zwischen 0 und 1 zu (linear skaliert zur Länge der Liste)
3. Plotten Sie die Messung farbig
4. Plotten Sie das Ergebnis der Optimierung farbig
5. Kompletieren Sie das Diagramm (s. Aufgabe 1)

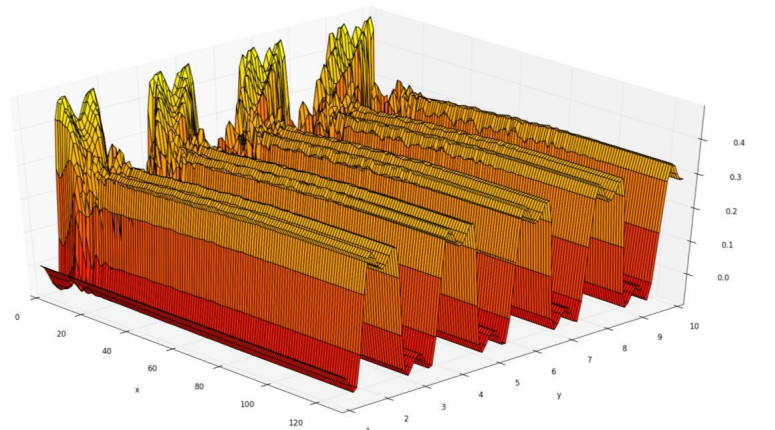
Ergebnis:



Zusatz: 3D Plot (siehe auch Skript plotten.py)

1. Generieren Sie ein Meshgrid für die alle Zwischenplots
2. Erzeugen Sie einen 3D-Plot mit `plot_surface()`
3. Welchen Einfluss haben die Parameter `rstride` und `cstride`?
4. Kompletieren Sie den Plot, probieren Sie verschiedene Colormaps (`matplotlib.cm`)

Ergebnis:



Aufgabe 3

Darstellung der Energie zur Modellkontrolle (Selbststudium)

1. Erstellen Sie eine Funktion zur Berechnung der Energie im System (aus symbolischer Lösung)
2. Speichern sie die Energie als Funktion der Zeit innerhalb der Zustandsableitung.
3. Simulieren Sie das System 100s lang mit 0.01s Solverschrittweite
4. Plotten sie die Energie dieser Simulationsrechnung
5. Variieren Sie die Schrittweite des Solvers und plotten sie die Ergebnisse in ein gemeinsames Diagramm - welcher Effekt ist zu erkennen?