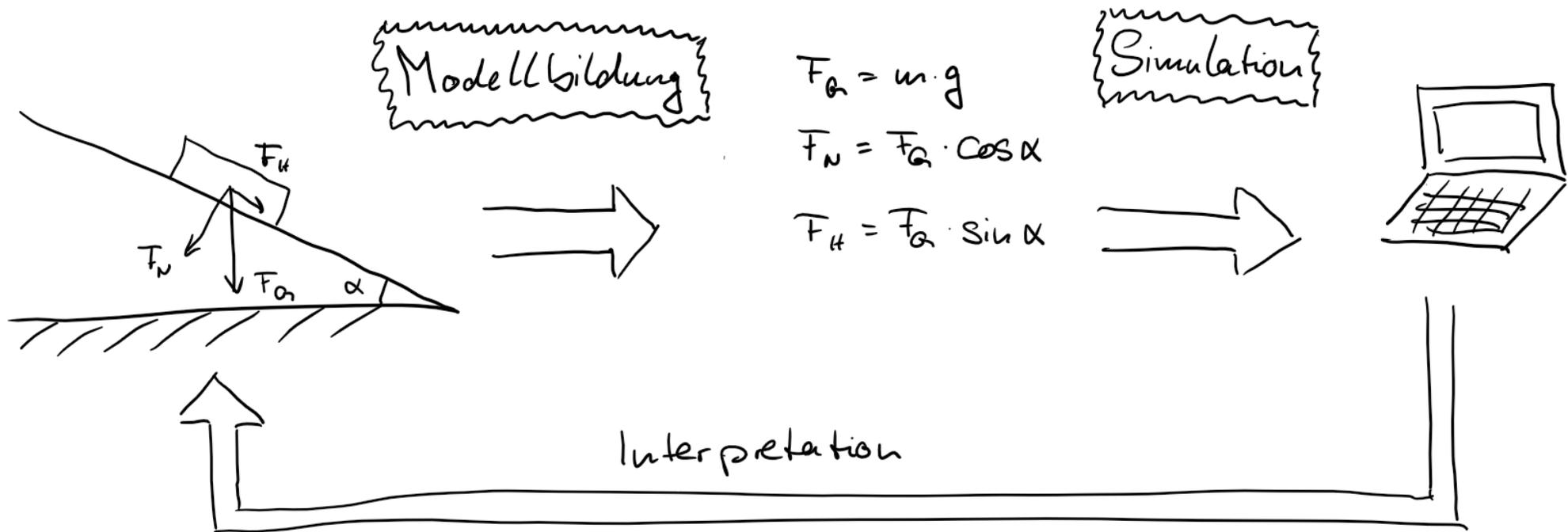


Ziel der Vorlesung

Modellbildung und Simulation



1. Schritt: Modellbildung

- Naturwissenschaften
- Mathematische Beschreibung physikalischer Regeln und Zusammenhänge
- Physikalische Systeme in dieser Vorlesung
 - Passive elektrische Netzwerke
 - Mechanische Feder-Masse-Systeme
- Mathematische Beschreibung
 - Lineare Algebra
 - Differentialgleichungen

2. Schritt: Simulation

- Beschreibung mathematischer Zusammenhänge auf einem Computer
- Berechnung und Visualisierung der Ergebnisse
- Analyse verschiedener Fragestellungen
- Simulationstools
 - General purpose programming language
 - Low-code Plattformen
 - Spezialisierte Simulationsumgebung

General purpose programming languages

- Beispiele: C/C++, Python, Matlab, Julia, Fortran, etc.
- Vorteile:
 - Hohe Flexibilität bei der Realisierung
 - Gute Ausführungsperformance
 - Hohe Skalierbarkeit
 - Schnittstellen zu anderen IT-Systemen
- Nachteile:
 - Meist eingeschränkte Visualisierungsmöglichkeiten
 - Hoher Implementierungsaufwand

Low-code Plattformen

- Beispiele: Simulink, Modelica, etc.
- Vorteile:
 - Schnelle Entwicklung
 - Wenig IT- bzw. Programmierkenntnisse notwendig
 - Mittlerweile akzeptable Ausführungsperformance
- Nachteile:
 - Häufig Lizenzkosten
 - Eingeschränkte Schnittstellen zu anderen Simulationssystemen
 - Vendor-lock-in: Gefangen im Eco-System des Herstellers

Spezialisierte Simulationsumgebung

- Beispiele: PSpice, LTSpice, Pscad, etc.
- Vorteile:
 - Keine IT- bzw. Programmierkenntnisse notwendig
 - Sehr spezialisierte Abbildung der physikalischen Realität
 - Umfangreiche Modelle
- Nachteile:
 - Häufig Lizenzkosten
 - Einhalten des vorgegebenen Workflows
 - Wenig Erweiterungsmöglichkeiten

Simulationstool in dieser Vorlesung: Matlab/Simulink

– Warum:

- Quasi Industrie-Standard im Bereich Simulation
- Viele Erweiterungen für unterschiedliche Disziplinen (sog. *Toolboxen*)
- Leicht zu lernende Programmiersprache Matlab
- Intuitive Bedienung der low-code Erweiterung Simulink

– Nachteile:

- Lizenzkosten
- Geschlossenes Eco-System (geringe Flexibilität)

Alternativen zu Matlab/Simulink

– Alternativen zu Matlab:

- Octave
- Python (mit Scipy und Numpy)
- Julia
- Scilab (freier Matlab-Klon)

– Alternativen zu Simulink:

- XCos (Erweiterung von Scilab)
- OpenModelica (freie Implementierung von Modelica)
- Simplorer
- Plecs (für die Domäne Leistungselektronik)