

Allgemeine Relativitätstheorie mit dem Computer

*PC-POOL RAUM 01.120
JOHANN WOLFGANG GOETHE UNIVERSITÄT
05. JULI, 2019*

MATTHIAS HANAUSKE

*FRANKFURT INSTITUTE FOR ADVANCED STUDIES
JOHANN WOLFGANG GOETHE UNIVERSITÄT
INSTITUT FÜR THEORETISCHE PHYSIK
ARBEITSGRUPPE RELATIVISTISCHE ASTROPHYSIK
D-60438 FRANKFURT AM MAIN
GERMANY*

11. Vorlesung

Plan für die heutige Vorlesung

- Das Einstein Toolkit (ET)
 - Visualisieren der Ergebnisse einer Simulation mit Python
- Arbeit an den Vorlesungsprojekte (einzeln oder in Gruppen)

In den letzten beiden Vorlesungstermine werden werden die bearbeiteten Projekte in einem Kurzvortrag vorgestellt

Mündlichen Prüfungen:

Termin 1: 27. und 28. August 2019

Termin 2: 8. Oktober 2019

Studenten, die eine mündliche Prüfung machen möchten schreiben Sie mir bitte eine E-Mail (hanauske@th.physik.uni-frankfurt.de, Termin und präferierte Uhrzeit angeben).

3) Ausführung einer Simulation

Bevor man eine Simulation mit dem Einstein-Toolkit ausführen kann, muss man dem Programm beschreiben was man eigentlich ausrechnen will - dies wird in einer sogenannten Parameter-Datei gemacht. In diesem Unterpunkt wird zunächst die Struktur einer solchen Parameter-Datei beschrieben und dannach die einzelnen Schritte erläutert, wie man eine Simulation mit dem Einstein-Toolkit startet. Die beschriebene Parameter-Datei legt eine Einstein-Toolkit Installation ET_2015_05 zugrunde (für eine aktuelle Version (2017) siehe: [Aktuelle .par-files](#))

4) Kollaps eines Neutronensterns zum schwarzen Loch

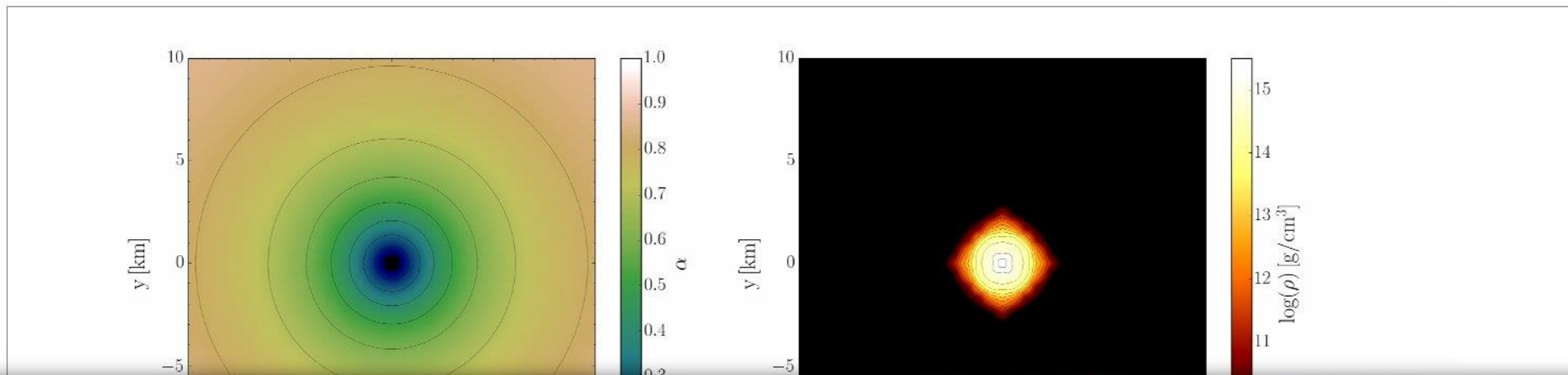
In diesem Unterpunkt werden die Simulationsergebnisse eines Kollapses eines Neutronensterns zu einem schwarzen Loch vorgestellt (näheres siehe [NS_collapse.pdf](#)). Einige der Ausgabedateien der Simulation kann man sich unter dem folgenden Link herunterladen: [output.zip](#)

Visualisierung der Simulationsergebnisse mit Mathematica

Um die Simulationsergebnisse mit Mathematica zu visualisieren ist es vorteilhaft sich auf der Internetseite <http://simulationtools.org/> die folgenden Zusatzpakete herunterzuladen: SimulationTools-1.0.0.tar.gz und h5mma-1.2.0.tar.gz. Diese muss man dann in den entsprechenden Mathematica Folder kopieren (cp *.gz .Mathematica/Applications/) und dort entpacken (tar -xvf ...).

Visualisierung der Simulationsergebnisse mit Python (matplotlib)

Oft wird auch die Programmiersprache Python benutzt um die Simulationsergebnisse des Einstein-Toolkits zu visualisieren. Es ist hierbei vorteilhaft sich auf der Bitbucket-Seite <https://bitbucket.org/dradice/scidata> Zusatzpakete herunterzuladen. Die mittels des Python-Scripts erzeugten einzelnen Bilder kann man z.B. wie folgt in einen Film umwandeln: `avconv -framerate 5 -i './img-%01d.jpg' -s 2000x1200 alphasrho.mp4`. Weiterführende Materialien findet man z.B. unter <http://matplotlib.org/>.

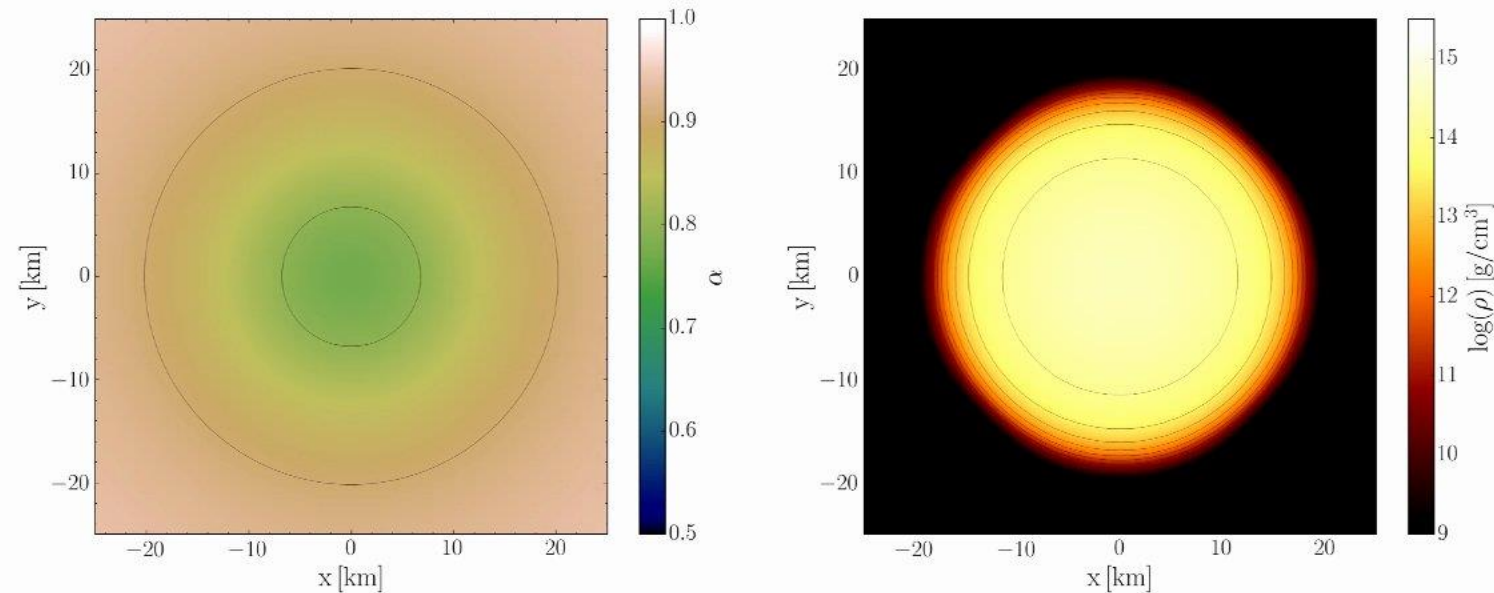


5) Migration eines instabilen Neutronensterns

In diesem Unterpunkt werden die Simulationsergebnisse einer Migration eines instabilen Neutronensterns zum stabilen Zweig vorgestellt (näheres siehe [Task 3](#)). Einige der Ausgabedateien der Simulation kann man sich unter dem folgenden Link herunterladen: [output.zip](#)

Visualisierung der Simulationsergebnisse mit Mathematica

Visualisierung der Simulationsergebnisse mit Python (matplotlib)



Migration eines instabilen Neutronensterns. Links die Lapse Funktion alpha und rechts der Logarithmus des Dichteprofiles. [Link zum Python-Script](#)

[Simulierte Beispiel-Simulationen](#)

Mögliche Vorlesungsprojekte

- Teil I: Simulationen und Berechnungen in Maple
 - Weiterführende Themen der Kerr-Metrik
 - Kosmologie und die Robertson-Walker Metrik
 - Alternative Gravitationstheorien
- Teil II: C++ oder Python
 - Die Masse-Radius Beziehung von Zwillingsternen
 - Geodätengleichung mit C++ oder Python lösen (nichtrotierendes schwarzes Loch)
 - Geodätengleichung mit C++ oder Python lösen (rotierendes schwarzes Loch)