Allgemeine Relativitätstheorie mit dem Computer

PC-POOL RAUM 01.120
JOHANN WOLFGANG GOETHE UNIVERSITÄT
21. JUNI, 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

9.+10. Vorlesung

Plan für die heutige Vorlesung

- Einführung in Teil III
- Der (3+1)-Split der Einsteingleichung
- Das Einstein Toolkit (ET)
  - Installation des ET mit Simfactory (Laptop, Fuchs-Cluster)
  - Die ".par"-Datei einer Simulation
  - "create" and "submit" einer Testsimulation auf dem Fuchs-Cluster
  - Beispiel-Simulation: Radiale Oszillation eines Neutronensterns mit dem ET
  - Visualisieren der Ergebnisse einer Simulation mit Gnuplot, Mathematica und Python
- Vorlesungsprojekte (einzeln oder in Gruppen)

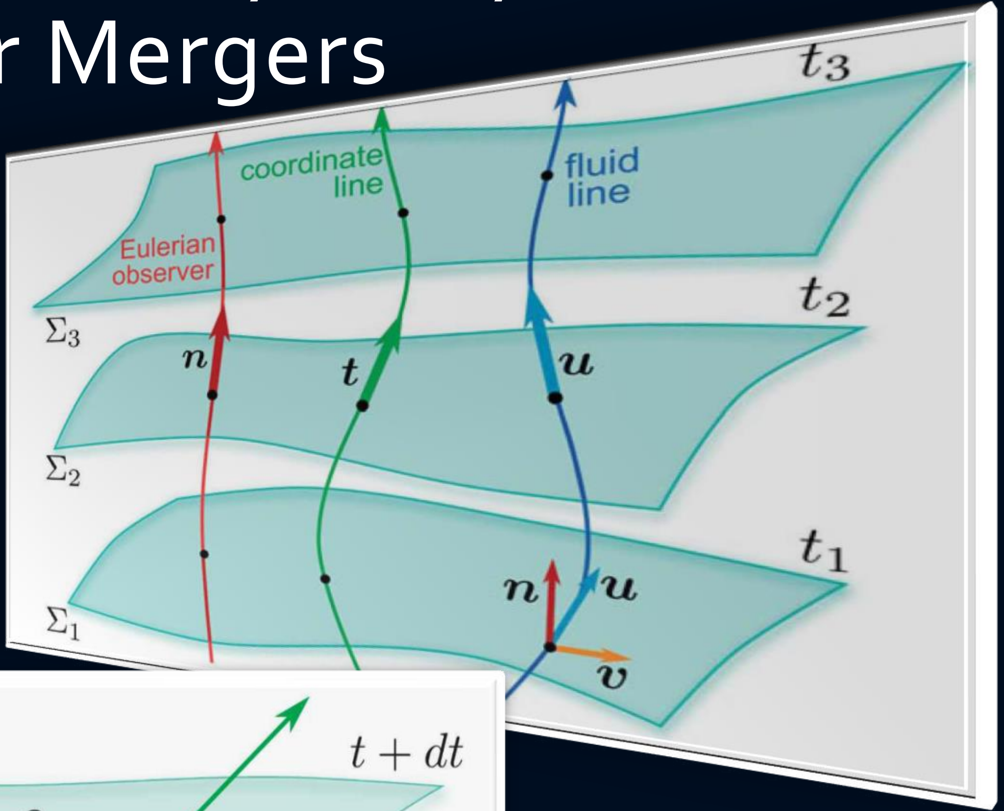
Die letzten beiden Vorlesungstermine werden vertreten
(bin auf der GR22 Konferenz in Valencia)

Numerical Relativity and Relativistic Hydrodynamics of Binary Neutron Star Mergers

A realistic numerical simulation of a twin star collapse, a merger of two compact stars or a collapse to a black hole needs to go beyond a static, spherically symmetric TOV-solution of the Einstein- and hydrodynamical equations.

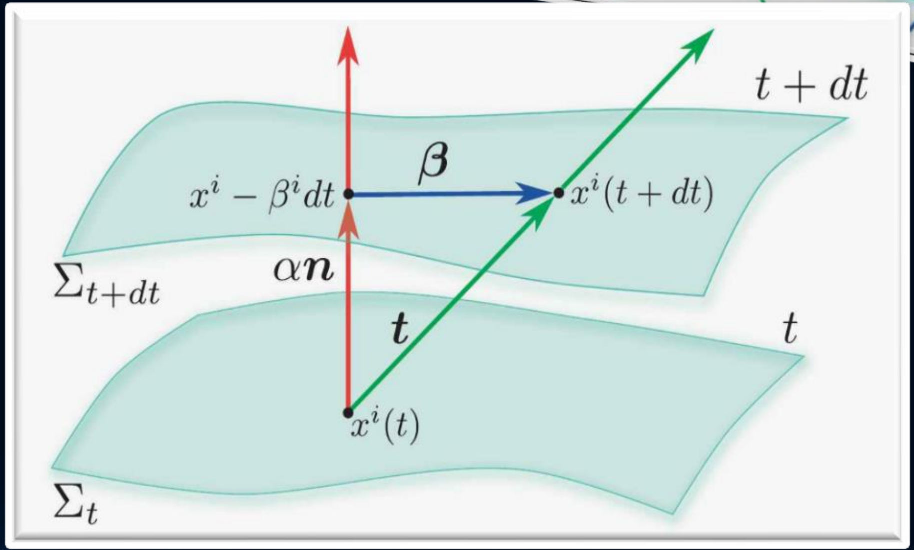
$$R_{\mu\nu} - \frac{1}{2} g_{\mu\nu} R = 8\pi T_{\mu\nu}$$

$$\nabla_\mu (\rho u^\mu) = 0\,, \quad \nabla_\nu T^{\mu\nu} = 0\,.$$



(3+1) decomposition of spacetime

$$g_{\mu\nu} = \begin{pmatrix} -\alpha^2 + \beta_i \beta^i & \beta_i \\ \beta_i & \gamma_{ij} \end{pmatrix}$$



$$d\tau^2 = \alpha^2(t, x^j) dt^2 \qquad x^i_{t+dt} = x^i_t - \beta^i(t, x^j) dt$$

All figures and equations from: Luciano Rezzolla, Olindo Zanotti: Relativistic Hydrodynamics, Oxford Univ. Press, Oxford (2013)

The ADM equations

The ADM (Arnowitt, Deser, Misner) equations come from a reformulation of the Einstein equation using the (3+1) decomposition of spacetime.

$$\partial_t \gamma_{ij} = -2\alpha K_{ij} + \mathscr{L}_{\boldsymbol{\beta}} \gamma_{ij} = -2\alpha K_{ij} + D_i \beta_j + D_j \beta_i$$

$$\partial_t K_{ij} = -D_i D_j \alpha + \beta^k \partial_k K_{ij} + K_{ik} \partial_j \beta^k + K_{kj} \partial_i \beta^k + \alpha \left({}^{(3)}R_{ij} + K K_{ij} - 2 K_{ik} K^k{}_j \right) + 4\pi\alpha \left[\gamma_{ij} (S - E) - 2 S_{ij} \right]$$

Time evolving part of ADM

$$D_j (K^{ij} - \gamma^{ij} K) = 8\pi S^i$$

$${}^{(3)}R + K^2 - K_{ij} K^{ij} = 16\pi E$$

Constraints on each hypersurface

Three dimensional covariant derivative

$$D_\nu := \gamma^\mu{}_\nu \nabla_\mu = (\delta^\mu_\nu + n_\nu n^\mu) \nabla_\mu$$

Three dimensional Riemann tensor

$${}^{(3)}R^\mu{}_{\nu\kappa\sigma} = \partial_\kappa {}^{(3)}\Gamma^\mu_{\nu\sigma} - \partial_\sigma {}^{(3)}\Gamma^\mu_{\nu\kappa} + {}^{(3)}\Gamma^\mu_{\lambda\kappa} {}^{(3)}\Gamma^\lambda_{\nu\sigma} - {}^{(3)}\Gamma^\mu_{\lambda\sigma} {}^{(3)}\Gamma^\lambda_{\nu\kappa}$$

$${}^{(3)}\Gamma^\alpha_{\beta\gamma} = \frac{1}{2} \gamma^{\alpha\delta} \left(\partial_\beta \gamma_{\gamma\delta} + \partial_\gamma \gamma_{\delta\beta} - \partial_\delta \gamma_{\beta\gamma} \right)$$

Spatial and normal projections of the energy-momentum tensor:

$$S_{\mu\nu} := \gamma^\alpha{}_\mu \gamma^\beta{}_\nu T_{\alpha\beta} ,$$

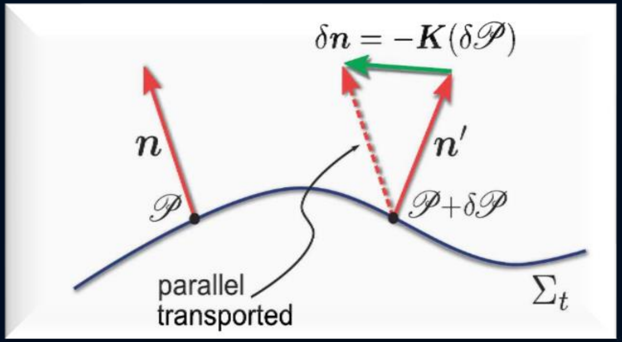
$$S_\mu := -\gamma^\alpha{}_\mu n^\beta T_{\alpha\beta} ,$$

$$S := S^\mu{}_\mu ,$$

$$E := n^\alpha n^\beta T_{\alpha\beta} ,$$

Extrinsic Curvature:

$$K_{\mu\nu} := -\gamma^\lambda{}_\mu \nabla_\lambda n_\nu$$



All figures and equations from: Luciano Rezzolla, Olindo Zanotti: Relativistic Hydrodynamics, Oxford Univ. Press, Oxford (2013)

From ADM to BSSNOK

Unfortunately the ADM equations are only weakly hyperbolic (mixed derivatives in the three dimensional Ricci tensor) and therefore not "well posed". It can be shown that by using a conformal traceless transformation, the ADM equations can be written in a hyperbolic form. This reformulation of the ADM equations is known as the BSSNOK (Baumgarte, Shapiro, Shibata, Nakamuro, Oohara, Kojima) formulation of the Einstein equation. Most of the numerical codes use this (or even better the CCZ4) formulation.

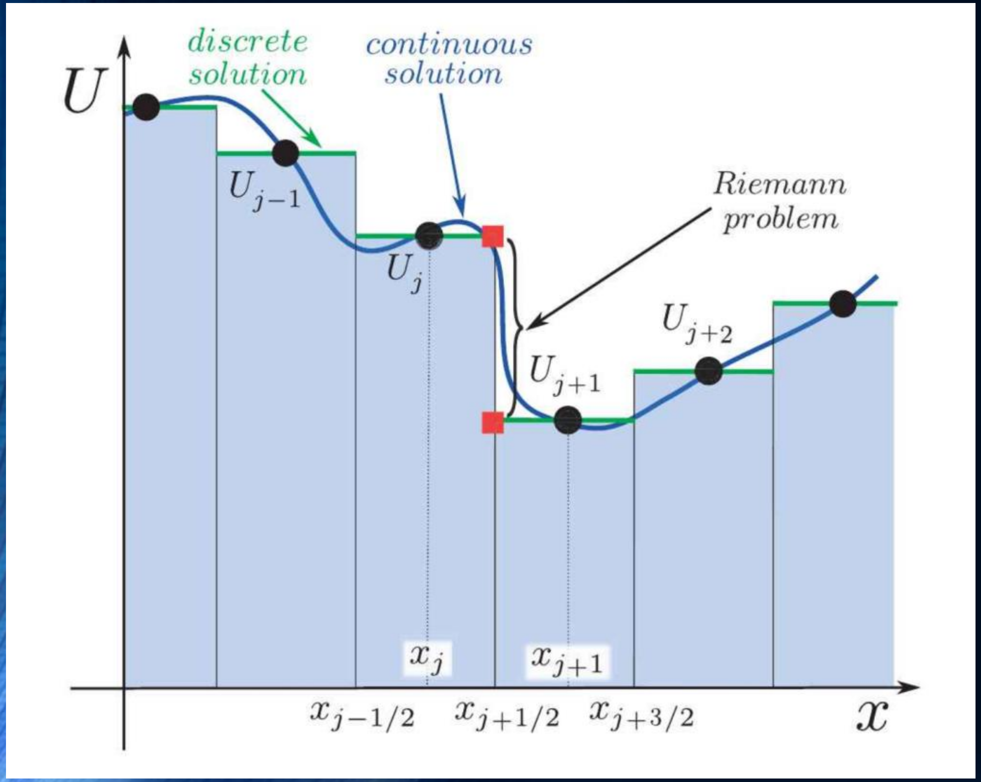
The 3+1 Valencia Formulation of the Relativistic Hydrodynamic Equations

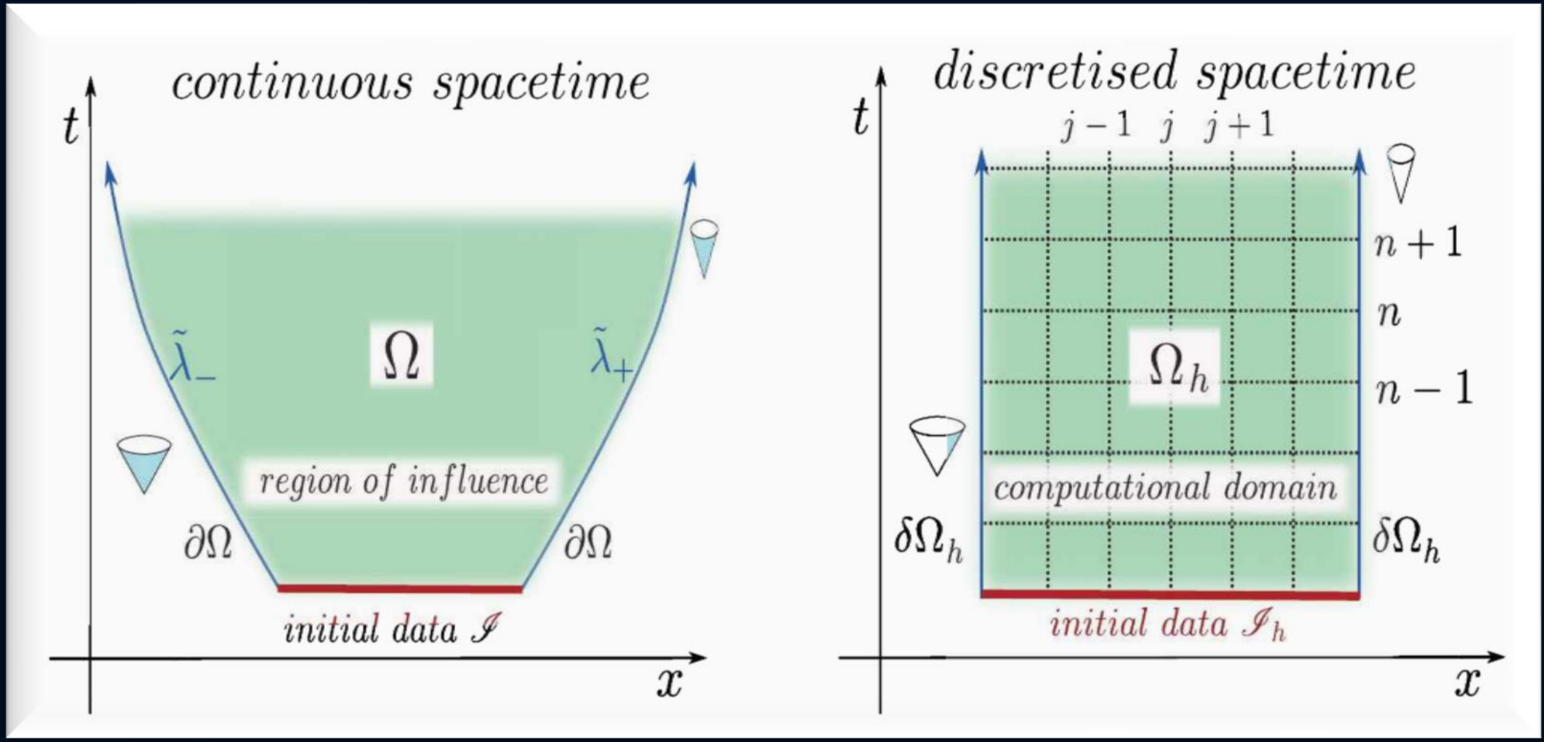
$$\nabla_\mu(\rho u^\mu) = 0\,,$$
$$\nabla_\nu T^{\mu\nu} = 0\,.$$

To guarantee that the numerical solution of the hydrodynamical equations (the conservation of rest mass and energy-momentum) converge to the right solution, they need to be reformulated into a conservative formulation. Most of the numerical “hydro codes” use here the 3+1 Valencia formulation.

Finite difference methods

Discretisation of a hyperbolic initial value boundary problem.





High resolution shock capturing methods (HRSC methods) are needed, when Riemann problems of discontinuous properties and shocks needs to be evolved accurately.

All figures from: Luciano Rezzolla, Olindo Zanotti: Relativistic Hydrodynamics, Oxford Univ. Press, Oxford (2013)

Gauge Conditions

On each spatial hypersurface, four additional degrees of freedom need to be specified: A slicing condition for the lapse function and a spatial shift condition for the shift vector need to be formulated to close the system. In an optimal gauge condition, singularities should be avoided and numerical calculations should be less time consuming.

Bona-Massó family of slicing conditions: $\partial_t \alpha - \beta^k \partial_k \alpha = -f(\alpha)\,\alpha^2 (K - K_0)$

"1+log" slicing condition: $f = 2/\alpha$ where $f(\alpha) > 0$ and $K_0 := K(t = 0)$

"Gamma-Driver" shift condition:

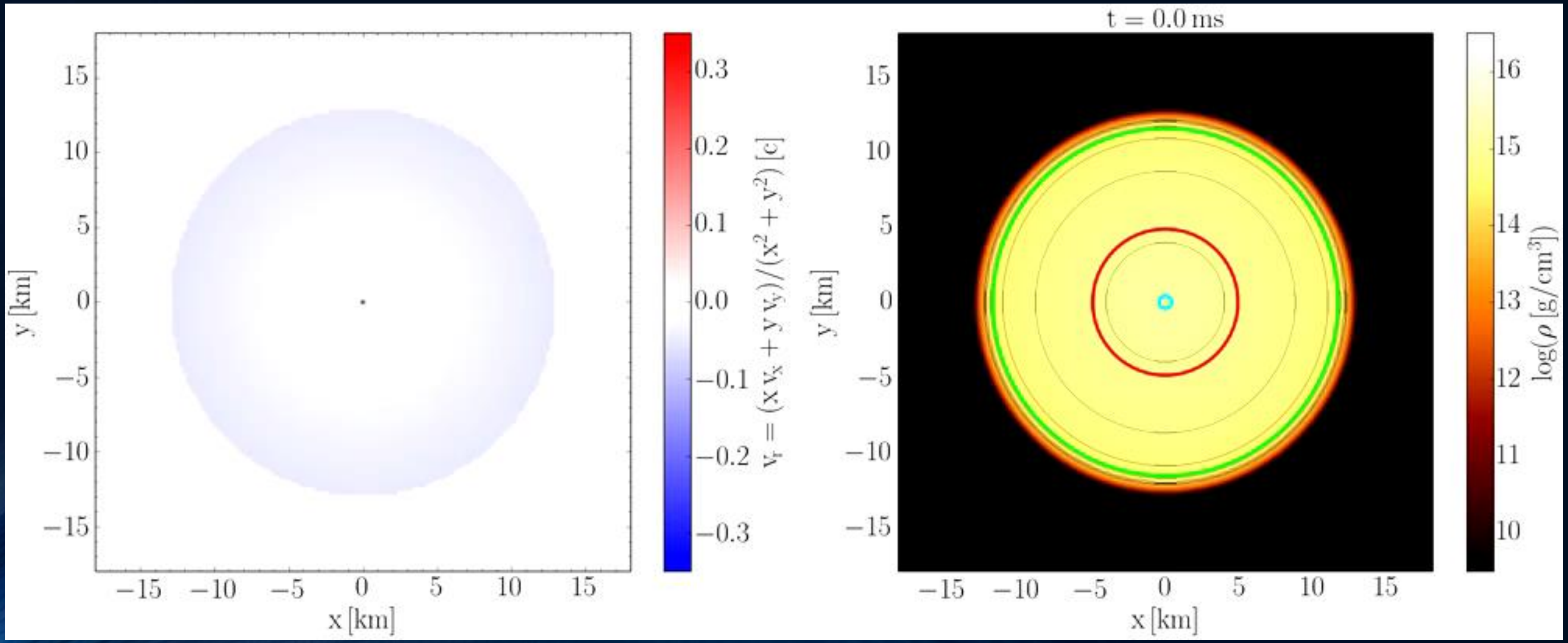
$$\partial_t \beta^i - \beta^j \partial_j \beta^i = \frac{3}{4} B^i,$$

$$\partial_t B^i - \beta^j \partial_j B^i = \partial_t \tilde{\Gamma}^i - \beta^j \partial_j \tilde{\Gamma}^i - \eta B^i$$

Teil III

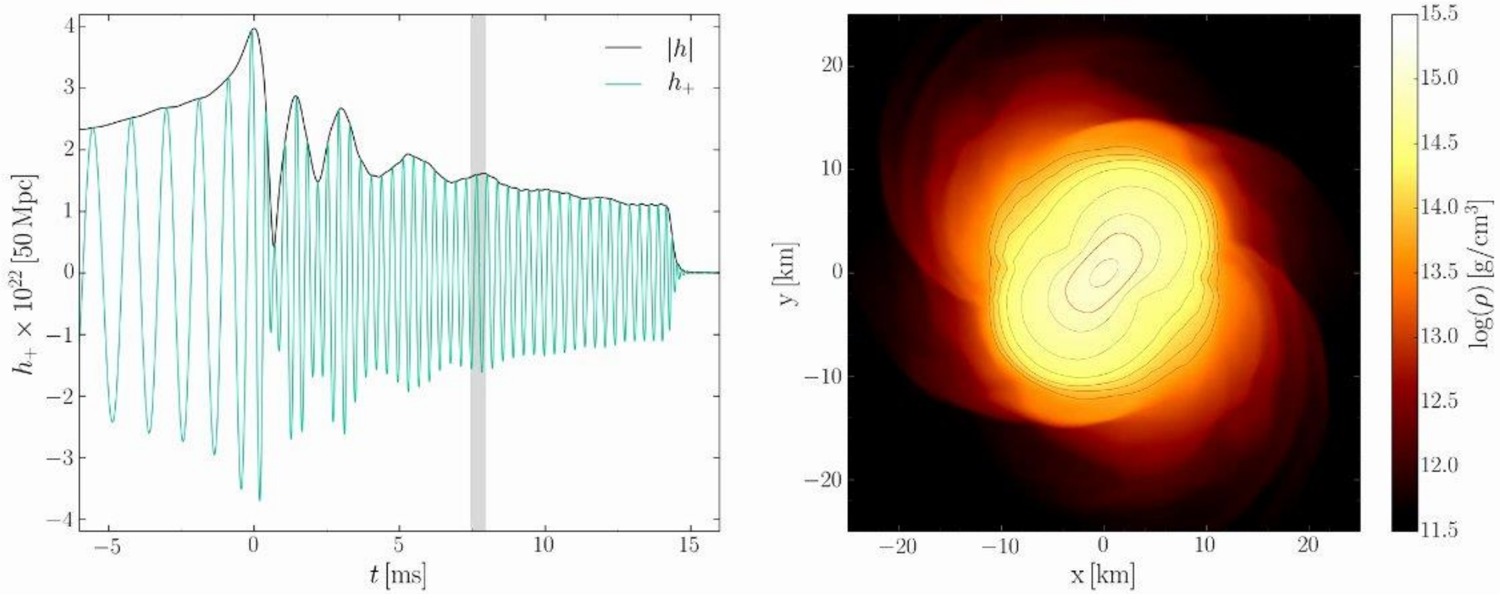
<u>Inhalte des Teil III:</u>

- How to download and build (compile) the Einstein Toolkit
- How to run a test simulation
- Run and visualize (Mathematica or Python) one of the following problems
  - Migration of an unstable neutron star to a stable configuration
  - Collapse of an unstable neutron star to a black hole
  - Binary neutron star mergers
  - Collapse of a neutron star to a quark star (twin star collapse)



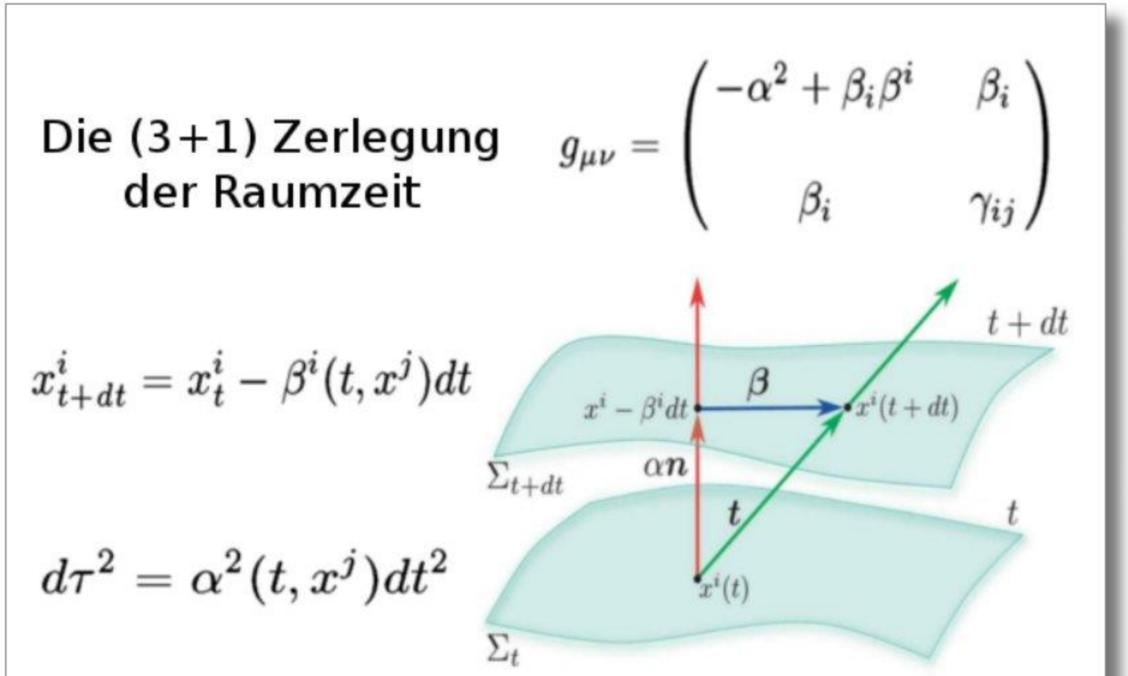
Teil III: Computersimulationen mit dem Einstein-Toolkit

In diesem dritten Teil der Vorlesung soll ein Einblick in die allgemeinrelativistische Simulation auf Supercomputern gegeben werden. Unter zuhilfenahme des sogenannten Einstein-Toolkit, einer frei zugänglichen Software zur Berechnung allgemeinrelativistischer Probleme, werden unterschiedliche, realistische Systeme betrachtet. Die folgende Animation wurde mit dem Einstein-Toolkit simuliert und zeigt die emittierten Gravitationswellen und das Dichteprofil einer Neutronenstern Kollision, wobei die Animation kurz vor der Kollision startet und kurz nach dem Kollaps zum Kerr schwarzen Loch endet. Das in dieser Simulation ca. 14 Millisekunden existierende Zwischenstadium bezeichnet man als einen Hypermassiven Neutronenstern. Nachdem in den letzten Jahren mehrere Gravitationswellen zweier kollidierender schwarzer Löcher direkt nachgewiesen wurden (siehe GW150914, GW151226 und GW170104), hofft man bei der nächsten LIGO Aufzeichnung auch Gravitationswellen zweier kollidierender Neutronensterne zu finden.



Links: Amplitude der emittierte Gravitationswellen im Abstand 50 Mpc von der Kollision. Rechts: Logarithmus des Dichteprofils in der äquatorialen Ebene des Hypermassiven Neutronensterns; die rote Kontourlinie markiert den Anfang der Quarkphase im inneren Bereich des Sterns.

1) Die Einsteingleichung im (3+1)-Split



Credit: L. Rezzolla, O. Zanotti: Relativistic Hydrodynamics, Oxford Univ. Press (2013)

Um die zeitliche Entwicklung von komplizierten, allgemeinrelativistischen Systemen auf dem Computer zu simulieren muss die Einsteingleichung zunächst umformuliert werden. In der sogenannten (3+1)-Zerlegung der Raumzeit wird die vierdimensionale Mannigfaltigkeit der Raumzeit in dreidimensionale, raumartige Hyperflächen Σ_t zerlegt. Die Metrik der Raumzeit $g_{\mu\nu}$ (siehe nebenstehende Abbildung) besteht in dieser Zerlegung aus einer Lapse-Funktion α, aus einem Shift-Vektor β^i und aus einer rein raumartigen Metrik γ_{ij} ($\mu\nu$=0,..3 und i,j=1..3). Die Lapse-Funktion α beschreibt den Unterschied zwischen der Koordinatenzeit t und der Eigenzeit τ und der Shift-Vektor β^i beschreibt wie stark ein Probekörper in der Raumrichtung "i" von der Struktur der Raumzeit mitgezogen wird, wenn er sich um einen infinitesimalen Zeitschritt dt bewegt. Setzt man diesen Ansatz in die Einsteingleichung ein, so gelangt man zu den sogenannten ADM-Gleichungen (nach Richard Arnowitt, Stanley Deser und Charles W. Misner), die ein System von Differentialgleichungen erster Ordnung darstellen. Um die Konvergenzeigenschaften numerischer Lösungen sicherzustellen wird im Einstein-Toolkit zusätzlich noch eine konforme, spurlose Transformation der Metrik durchgeführt und das System von Differentialgleichungen in eine hyperbolische Form gebracht. Diese Gleichungen werden dann, zusammen mit den hydrodynamischen Gleichungen, im Einstein-Toolkit numerisch gelöst.

2) Download und Kompilierung des Einstein-Toolkit

In diesem Unterpunkt werden die einzelnen Schritte beschrieben, wie man das frei erhältliche Simulationsprogramm Einstein-Toolkit installiert (für eine ausführliche Beschreibung siehe Einstein-Toolkit).

Einstein Toolkit

"The Einstein Toolkit Consortium is developing and supporting open software for relativistic astrophysics. Our aim is to provide the core computational tools that can enable new science, broaden our community, facilitate interdisciplinary research and take advantage of emerging petascale computers and advanced cyberinfrastructure."

* Consortium: 94 members, 49 sites, 14 countries
* Sustainable community model:
  * 9 Maintainers from 6 sites: oversee technical developments, quality control, verification and validation, distributions and releases
  * Whole consortium engaged in directions, support, development
  * Open development meetings
  * Governance model: still being discussed (looking at CIG, iPlant)

HTTP://WWW.EINSTEINTOOLKIT.ORG

Das Einstein Toolkit: Weitere Informationen

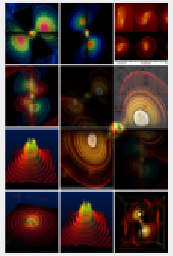
einstein toolkit

- Welcome
- About the Toolkit
  - Members
  - Maintainers
  - Governance
  - Capabilities
  - Gallery
  - Releases
  - Tools
- Download
- Community Services
  - Wiki
  - Blog
  - Support
  - Seminars
  - Issue Tracker
- Documentation
  - Tutorial for New Users
  - Citing

Welcome

The Einstein Toolkit Consortium is developing and supporting open software for relativistic astrophysics. Our aim is to provide the core computational tools that can enable new science, broaden our community, facilitate interdisciplinary research and take advantage of emerging petascale computers and advanced cyberinfrastructure.

Please read our pages about the Einstein Toolkit, its governance, and how to get started with the toolkit for more information.



EinsteinToolkit@Flickr

Download

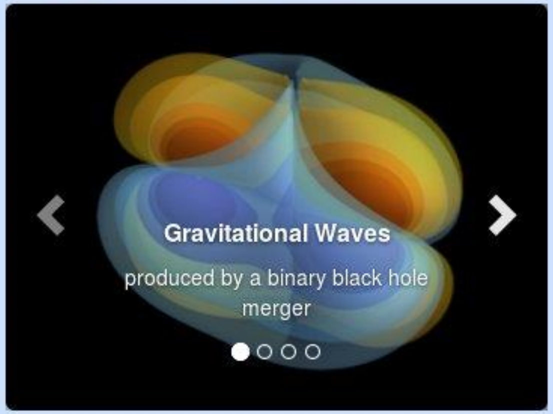
November 2014: We are pleased to announce the tenth release (code name "Herschel") of the Einstein Toolkit, an open, community developed software infrastructure for relativistic astrophysics.

https://www.youtube.com/watch?v=EO4d32ch6OI
https://www.youtube.com/watch?v=p5bq2iUO3DE
https://www.youtube.com/watch?v=MNpyd_o0MT4
https://www.youtube.com/watch?v=Qg6PwRI2uS8
https://www.youtube.com/watch?v=ZW3aV7U-aik

Das Einstein Toolkit

einstein toolkit Home About Download Documentation Help! Contribute Gallery

The Einstein Toolkit



Gallery

Einstein Toolkit School and Workshop

Join us at the North American Einstein Toolkit School and Workshop at NCSA, at the University of Illinois at Urbana-Champaign from July 31 to August 4 2017.

This meeting is open to anyone interested in numerical relativity and computational astrophysics and cosmology and in particular to Einstein toolkit users.

The first three days will be dedicated to a school useful for new users of the Einstein Toolkit followed by a two day long workshop open to developers interested in the Einstein Toolkit.

Registration closes *July 17, 2017*.

More information

About

The Einstein Toolkit is a community-driven software platform of core computational tools to advance and support research in relativistic astrophysics and gravitational physics.

About

Documentation

A lot of the documentation within the Einstein Toolkit is generated from comments in the source code, and more can be found on the Einstein Toolkit Wiki or other documents. We provide links to guides, tutorials and references.

Documentation

Download

We provide a convenient method to get all of the Einstein Toolkit with just a few commands, and explain the whole process.

Download

Contribute

The Einstein Toolkit would not exist without numerious contributions from its community. It is easy to learn how you can contribute as well.

Contribute

The Einstein Toolkit has been supported by NSF 1550551/1550461/1550436/1550514, NSF 1212401/1212426/1212433/1212460, NSF 0903973/0903782/0904015 (CIGR), 0701566/0855892 (XiRel), 0721915 (Alpaca), 0905046/0941653(PetaCactus/PRAC). Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

Das Einstein Toolkit: Download

einstein toolkit Home About Download Documentation Help! Contribute Gallery

Download & Requirements

The Einstein Toolkit is hosted on many different machines around the world. We provide a script called GetComponents to simplify downloading the toolkit. This page just describes how to download the toolkit - you may also be interested in the Tutorial for New Users which leads you through these steps and more on the Queen Bee supercomputer, or in a simpler tutorial for setup on a typical Linux box.

Users of the Einstein Toolkit are encouraged to register which also signs up for the users mailing list.

Main Toolkit

Citations

The development of production level scientific software, such as the components of the Einstein Toolkit, represents the academic output of researchers. These scientific contributions should be acknowledged and respected on par with those solely based in theory or experiment. Please review our Citation Policy.

Current release: Payne-Gaposchkin (released on December 16th, 2016)

This is the recommended version of the toolkit for most users. See the release notes for more information.

Note: OSX users cannot use the 'subversion' client shipped by Apple. In that case install subversion either from homebrew or macports.

Enter the directory on your machine in which you would like to download the ET (for example, your home directory), and type the commands listed below. This will create a directory called Cactus in which the components of the Einstein Toolkit are downloaded.

```
curl -kLO https://raw.githubusercontent.com/gridaphobe/CRL/ET_2016_11/GetComponents
chmod a+x GetComponents
./GetComponents --parallel https://bitbucket.org/einsteintoolkit/manifest/raw/ET_2016_11/einsteintoolkit.th
```

A tarball of the release is also available here, but using GetComponents is the preferred method to obtain the code. Use the tarball only if there is no way to use GetComponents (which should almost never be the case).

ET-Download auf dem Fuchs-Cluster

```
             into: Cactus/arrangements
------------------------------------------------------------------
  Checking out module: PITTNullCode/SphericalHarmonicDecomp
      from repository: https://bitbucket.org/einsteintoolkit/pittnull
```

```
[prakti1@login02.csc ~]$ cd ET-2016-11/
[prakti1@login02.csc ET-2016-11]$ curl -kLO https://raw.githubusercontent.com/gridaphobe/CRL/ET_2016_11/GetComponents
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100 99330  100 99330    0     0   486k      0 --:--:-- --:--:-- --:--:-- 30.9M
[prakti1@login02.csc ET-2016-11]$ chmod a+x GetComponents
[prakti1@login02.csc ET-2016-11]$ ./GetComponents --parallel https://bitbucket.org/einsteintoolkit/manifest/raw/ET_2016_11/einsteintoolkit.th
-----------------------------------------------------------------
  Checking out module: par
      from repository: https://bitbucket.org/einsteintoolkit/einsteinexamples.git
                 into: Cactus
```

```
                 into: Cactus
                   as: flesh
-----------------------------------------------------------------
  Checking out module: COPYRIGHT
      from repository: https://bitbucket.org/cactuscode/cactus.git
                 into: Cactus
                   as: flesh
-----------------------------------------------------------------
  Checking out module: doc
      from repository: https://bitbucket.org/cactuscode/cactus.git
                 into: Cactus
                   as: flesh
-----------------------------------------------------------------
  Checking out module: lib
      from repository: https://bitbucket.org/cactuscode/cactus.git
                 into: Cactus
                   as: flesh
-----------------------------------------------------------------
  Checking out module: ./utils
      from repository: https://bitbucket.org/cactuscode/utilities.git
                 into: Cactus
                   as: utils
-----------------------------------------------------------------
  Checking out module: Makefile
```

```
      from repository: https://bitbucket.org/einsteintoolkit/pittnull
                 into: Cactus/arrangements
------------------------------------------------------------------
  Checking out module: EinsteinInitialData/IDConstraintViolate
      from repository: https://bitbucket.org/einsteintoolkit/einsteini
                 into: Cactus/arrangements
------------------------------------------------------------------
  Checking out module: ./CoreDoc
      from repository: https://bitbucket.org/cactuscode/coredoc.git
                 into: Cactus/arrangements/CactusDoc
                   as: CoreDoc
------------------------------------------------------------------
  268 components checked out successfully.
  0 components updated successfully.

  Time Elapsed: 18 minutes, 5 seconds

[prakti1@login02.csc ET-2016-11]$
```

Das Einstein Toolkit: Setup mit SimFactory

```
[prakti1@login02.csc Cactus]$ ./simfactory/bin/sim setup --machine fuchs


Here we will define some necessary Simulation Factory defaults.


Determining local machine name: login02.cm.cluster
Creating machine login02.cm.cluster from generic: machine login02.cm.cluster [/home/agmisc/prakti1/ET-2016-11/Cactus/repos/simfactory2/mdb/machin
enter value for key user [prakti1]:
enter value for key email [prakti1]:
enter value for key allocation []:
enter value for key sourcebasedir (the parent directory containing the Cactus sourcetree) [/home/agmisc/prakti1/ET-2016-11]:
enter value for key basedir (the location of simfactory simulations) [/home/agmisc/prakti1/simulations]:


would you like to enter key/value pairs for a specific machine? [Y/N*]:

--------------------SUMMARY--------------------:


[default]
user            = prakti1
email           = prakti1
allocation      =
sourcebasedir   = /home/agmisc/prakti1/ET-2016-11
basedir         = /home/agmisc/prakti1/simulations


------------------END SUMMARY------------------:

Save contents [Y*/N]:
Contents successfully written to /home/agmisc/prakti1/ET-2016-11/Cactus/repos/simfactory2/etc/defs.local.ini
[prakti1@login02.csc Cactus]$ █
```

Das Einstein Toolkit: Kompilierung

```
[praktil@login02.csc Cactus]$ ./simfactory/bin/sim build et --thornlist ./manifest/einsteintoolkit.th --machine fuchs
Using configuration: et
Reconfiguring et
Writing configuration to: /home/agmisc/praktil/ET-2016-11/Cactus/configs/et/OptionList
Cactus - version: 4.2.3
Reconfiguring et.
Using configuration options from configure line
  Setting fds to '4,5 -j --'
End of options from configure line
Adding configuration options from '/home/agmisc/praktil/ET-2016-11/Cactus/configs/et/OptionList'...
  Setting VERSION to '2015-05-16'
  Setting CPP to 'cpp'
  Setting FPP to 'cpp'
  Setting CC to '/cm/shared/apps/intel/composer_xe/2013_sp1.3.174/composer_xe_2013_sp1.3.174/bin/intel64/icc'
  Setting CXX to '/cm/shared/apps/intel/composer_xe/2013_sp1.3.174/composer_xe_2013_sp1.3.174/bin/intel64/icpc'
  Setting F77 to '/cm/shared/apps/intel/composer_xe/2013_sp1.3.174/composer_xe_2013_sp1.3.174/bin/intel64/ifort'
  Setting F90 to '/cm/shared/apps/intel/composer_xe/2013_sp1.3.174/composer_xe_2013_sp1.3.174/bin/intel64/ifort'
  Setting CPPFLAGS to '-DCCTK_DISABLE_OMP_COLLAPSE -DCCTK_DISABLE_RESTRICT'
  Setting FPPFLAGS to '-DCCTK_DISABLE_OMP_COLLAPSE -traditional -DCCTK_DISABLE_RESTRICT'
  Setting CFLAGS to '-g -traceback -msse3 -align -std=c99 -U__STRICT_ANSI__'
  Setting CXXFLAGS to '-g -traceback -msse3 -align -std=c++11 -D__builtin_fmaxf=fmaxf -D__builtin_fmaxl=fmaxl -D__builtin_fm
  Setting F77FLAGS to '-g -traceback -msse3 -align -pad -safe-cray-ptr'
  Setting F90FLAGS to '-g -traceback -msse3 -align -pad -safe-cray-ptr'
  Setting C_LINE_DIRECTIVES to 'yes'
  Setting F_LINE_DIRECTIVES to 'yes'
  Setting LDFLAGS to '-Wl,--export-dynamic -Wl,-rpath,/cm/shared/apps/intel/composer_xe/2013_sp1.3.174/composer_xe_2013_sp1.3
th,/cm/shared/apps/intel/composer_xe/2013_sp1.3.174/composer_xe_2013_sp1.3.174/ipp/lib/intel64 -Wl,-rpath,/cm/shared/apps/int
_xe_2013_sp1.3.174/tbb/lib/intel64/gcc4.4'
  Setting BEGIN_WHOLE_ARCHIVE_FLAGS to '-Wl,--whole-archive'
```

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

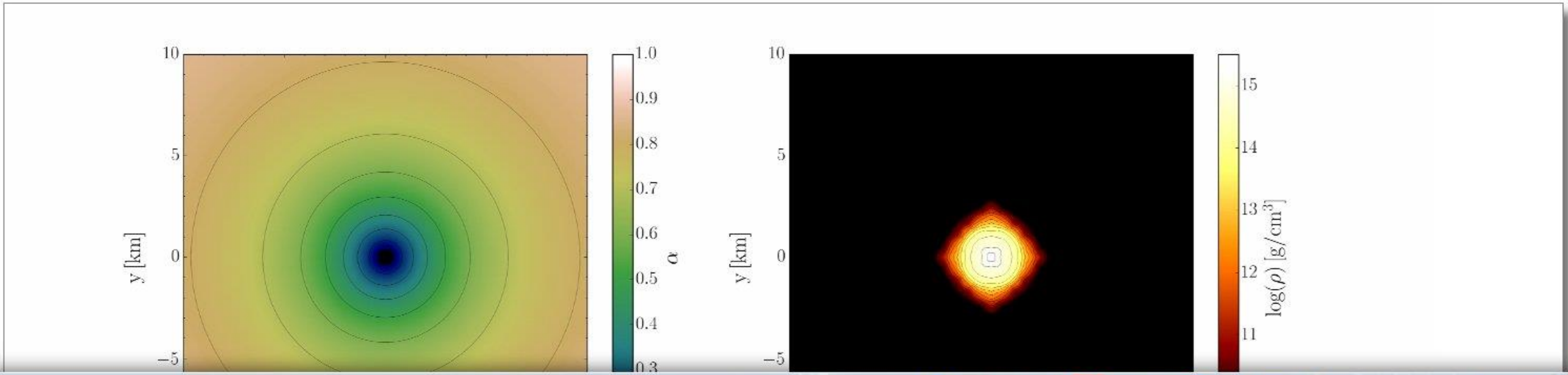
In diesem Unterpunkt werden die Simulationsergebnisse eines Kollapses eines Neutronensterns zu einem schwarzen Loch vorgestellt (näheres siehe NS_collapse.pdf). Einige der Ausgabedateien der Simulation kann man sich unter dem folgenden Link herunterladen: output.zip

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 alpharho.mp4 .Weiterführende Materialien findet man z.B. unter http://matplotlib.org/.

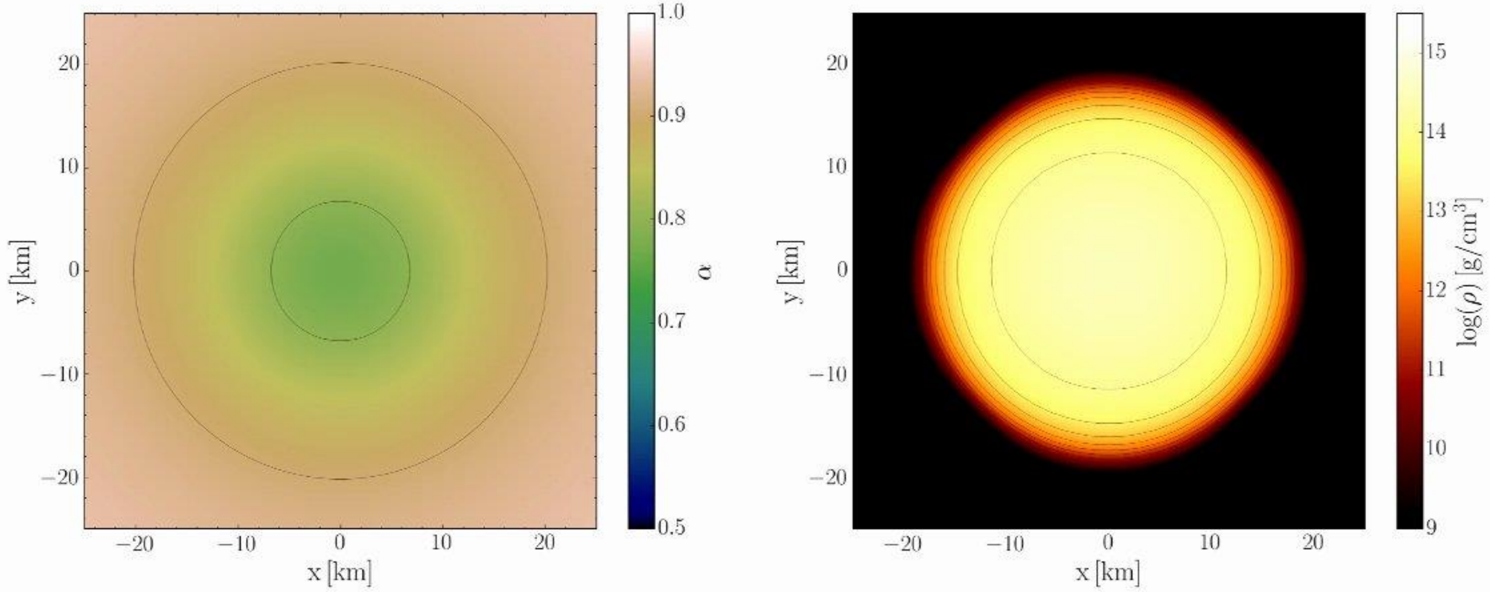


5) Migration eines instabilen Neutronensterns

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

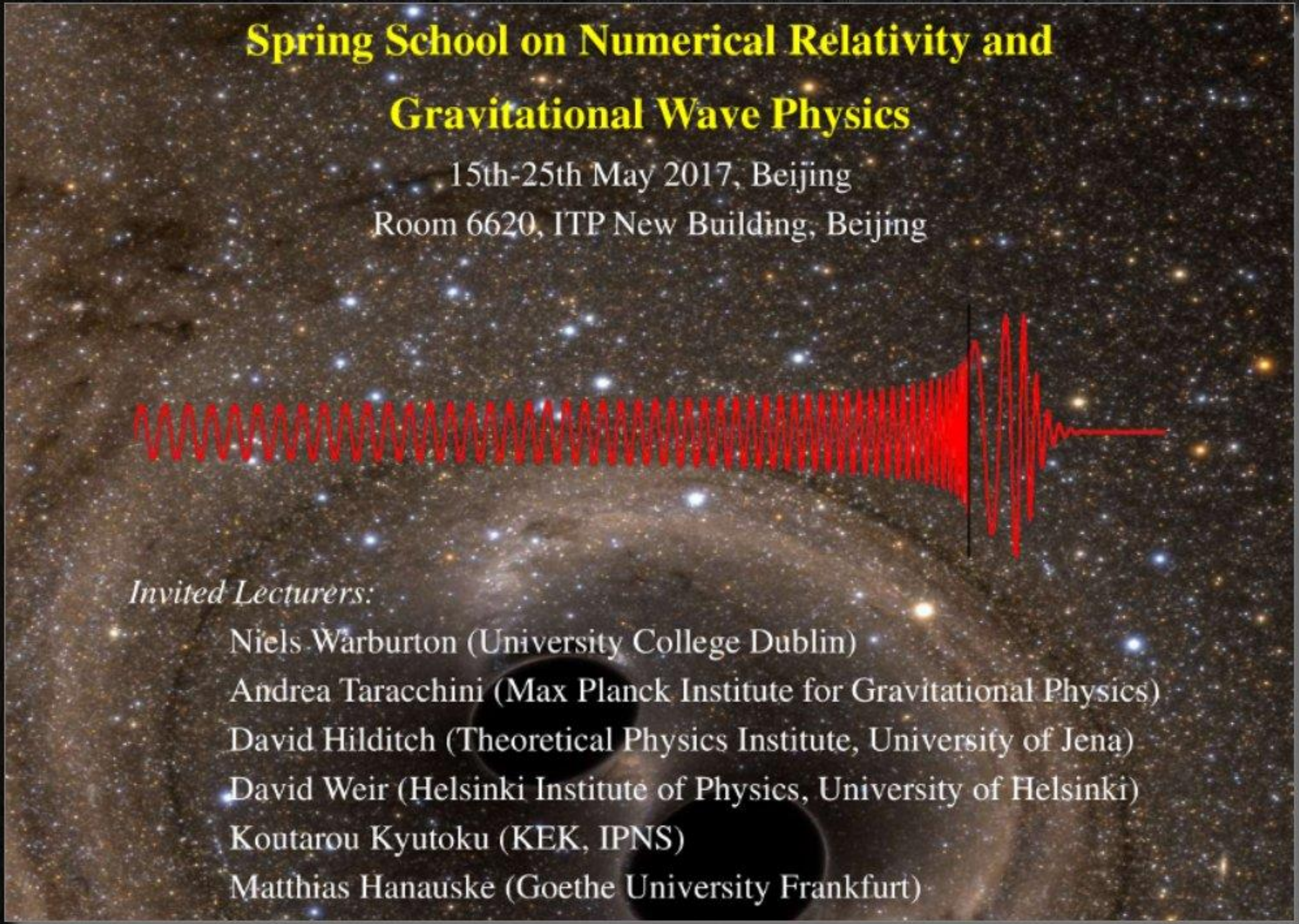
Visualisierung der Simulationsergebnisse mit Mathematica

Visualisierung der Simulationsergebnisse mit Python (matplotlib)



Migration eines instabilen Neutronensterns. Links die Lapse Funktion alpha und rechts der Logarithmus des Dichteprofils. Link zum Python-Script

Simulierte Beispiel-Simulationen



Die Peking Frühlingsschule 2017
Numerische Relativitätstheorie und die Physik der Gravitationswellen
(Spring School on Numerical Relativity and Gravitational Wave Physics)

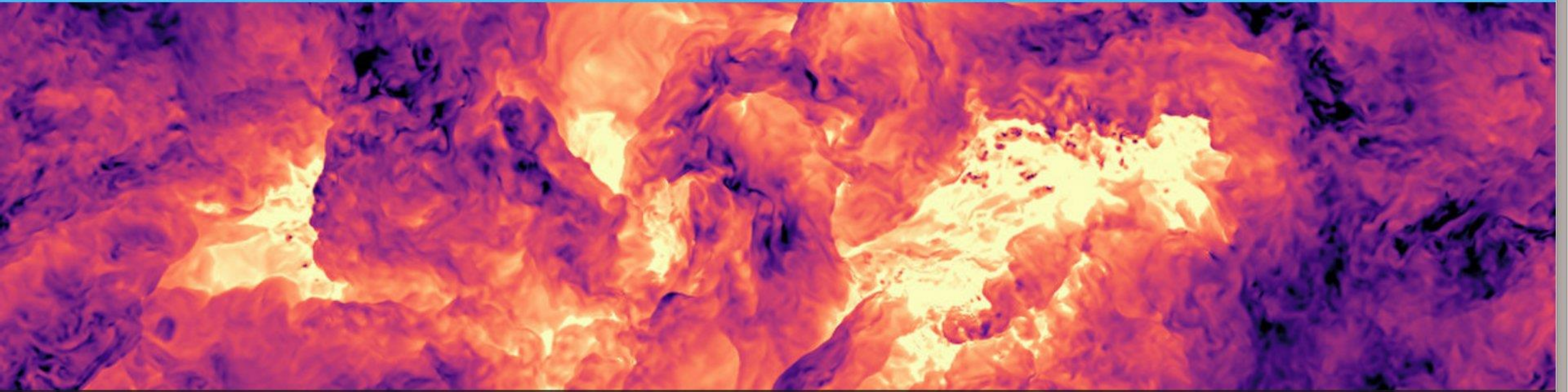
Vorlesungsreihe (6 Vorlesungen) über
Gravitationswellen von kollidierenden kompakten Sternen und die Eigenschaften seltsamer Materie
(Gravitational waves from colliding compact star binaries in the context of strange/exotic matter)
致密星碰撞引起的引力波和奇异物质的性质
Beijing, China, 15.-25. May 2017

Die im Jahre 2017 gehaltene Vorlesungsreihe führt einerseits in die Allgemeine Relativitätstheorie ein, andererseits fokussiert sie sich auf den speziellen Teilaspekt der relativistischen Astrophysik kollidierender hybrider Neutronensterne, in deren innerem Bereich es zur Bildung seltsamer und exotischer Materie kommen kann. Kollabiert ein instabiler Neutronenstern zu einem schwarzen Loch oder zu einem Quark Stern? Wie kann man anhand des ausgesandten Gravitationswellen-Signals zweier kollidierender kompakter Sterne die Eigenschaften der Nuklearen- und Quark-Materie entschlüsseln?

(The series of lectures held in 2017. Topics: theory of general relativity theory, relativistic astrophysics of colliding hybrid neutron stars, strange and exotic matter in the interior of compact stars. Questions: Does an unstable neutron star collapse to a black hole or quark star? How can we extract the strange properties of high density nuclear and quark matter by means of the emitted gravitational wave signal of two colliding compact stars?)

在2017年开设的课程，一方面介绍广义相对论理论，另一方面聚焦于相对论天体物理中的一个特殊部分：混合致密星碰撞,以及在其内部可能生成的奇异和异常物质。

David Radice - Homepage



WhiskyTHC: the General-Relativistic Templated Hydrodynamics Code **Download**

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 Zwillingssternen
  - Geodätengleichung mit C++ oder Python lösen (nichtrotierendes schwarzes Loch)
  - Geodätengleichung mit C++ oder Python lösen (rotierendes schwarzes Loch)