Master

Martin Kjøllesdal Johnsrud
Supervisor: Jens Oluf Andersen

January 14, 2022

Todo list

Ha appendix på functional derivatives . . . 11

Utlede? . . . 11

Forklar . . . 13

Contents

1 General relativity and the TOV-equation **5**

1.1 Differential geometry . . . 5

1.2 General relativity . . . 11

1.3 The TOV equation . . . 13

A Code **15**

Chapter 1

General relativity and the TOV-equation

General relativity describes how the presence of matter and energy bends the fabric of space and time. It was first written down by Einstein more than a century ago and is to this day the most accurate model of gravitational effects. It makes accurate and counterintuitive predictions, which experiments have borne out. In this chapter, we will survey the basics of general relativity as well as some mathematical prerequisites. We will then use this to derive the Tolman-Oppenheimer-Volkoff (TOV) equation. This differential equation models massive stellar objects, such as stars. This chapter is based on [1, 2].

1.1 Differential geometry

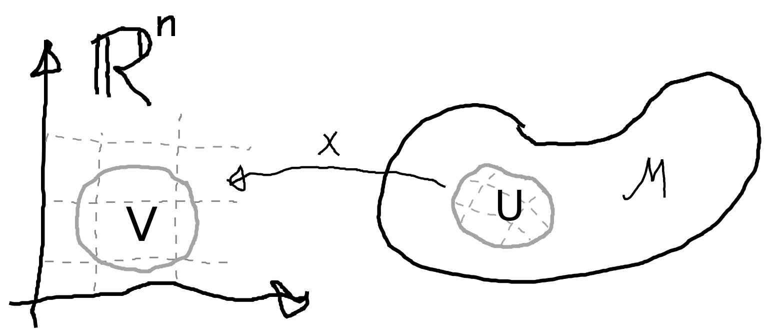

Figure 1.1: (Kladd) The coordinate funciton x maps a neighborhood U in the manifold $\mathcal{M}$ to a negiborhood V in $\mathbb{R}^n$.

General relativity is formulated in the language of *differential geometry*, which generalizes multivariable calculus to more general spaces than $\mathbb{R}^n$. These spaces are *smooth manifolds*. A manifold is a set of points, $\mathcal{M}$, that are locally homeomorphic to $\mathbb{R}^n$. That is, for all points $p \in \mathcal{M}$, there exists a neighborhood U around p, together with a corresponding set of continuous, bijective functions,

$$x : U \subseteq \mathcal{M} \longmapsto V \subseteq \mathbb{R}^n, \tag{1.1}$$
$$p \longmapsto x^\mu(p). \tag{1.2}$$

We call $x(p) = (x^0(p), \ldots, x^{n-1}(p)) = x^\mu(p)$ a coordinate function of $\mathcal{M}$. The inverse of x, x^{-1}, obeys $x^{-1}(x(p)) = p$, for all $p \in U$. A smooth manifold is one in which the coordinate functions are infinitely differentiable. Differentiability is defined by considering two different coordinate functions, x and x'. The corresponding domains, U and U', may or may not be overlapping. We then define the transition function,

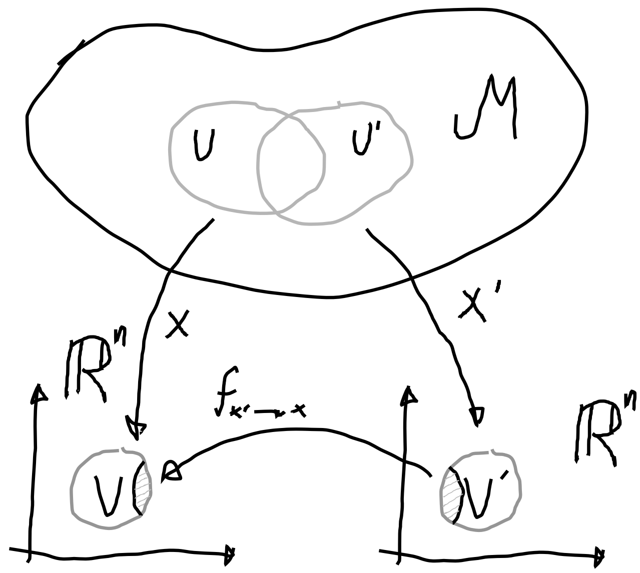

Figure 1.2: (Kladd) The transition map $f_{x' \to x}$ between two coordinate functions, x and x', maps between the images of these function, "through" the manifold $\mathcal{M}$. The domain and image of this funciton as to be restricted to a (possibly empty) subset of the images of x and x', as illustrated by the shaded regions in V and V'.

a function between subsets of $\mathbb{R}^n$ by mapping via $\mathcal{M}$, as

$$f_{x' \to x} = x' \circ x^{-1} : \mathbb{R}^n \mapsto \mathbb{R}^n. \tag{1.3}$$

The map is illustrated in Figure 1.2. [1] A set of coordinate functions $\mathcal{A} = \{x_i\}$ whose domain cover $\mathcal{M}$ is called an *atlas* of $\mathcal{M}$. If the transition function between all pairings of coordinate functions in the atlas is smooth, that is infinitely differentiable, we call the atlas smooth. We then define a smooth manifold as the topological manifold $\mathcal{M}$ together with a *maximal* smooth atlas $\mathcal{A}$. A smooth atlas is maximal if no coordinate function can be added while the atlas remains smooth.[2]

Consider two m and n dimensional smooth manifolds $\mathcal{M}$ and $\mathcal{N}$. Let x denote the coordinates on $\mathcal{M}$, while y denotes the coordinates on $\mathcal{N}$. We can define smooth functions between these manifolds similarly to how we define smooth coordinates. Consider the function

$$F : \mathcal{M} \longmapsto \mathcal{N}. \tag{1.4}$$

It is said to be smooth if, for all points $p \in M$, there is a set of local coordinates x around p and y around $F(p)$ such that the map $\tilde{F} = y \circ F \circ x^{-1}$ is smooth. This map is defined by the diagram

$$\begin{tikzcd} \mathcal{M} \arrow[r, "F"] \arrow[d, "x"] & \mathcal{N} \arrow[d, "y"] \\ \mathbb{R}^m \arrow[r, "\tilde{F}"] & \mathbb{R}^n \end{tikzcd} \tag{1.5}$$

We will not be careful with the distinction between F, the function between the abstract manifolds, and $\tilde{F}$, the function of their coordinates, but rather denote both by $F(x)$. We may take the partial derivative of such a function with respect to the coordinates x, $\partial F / \partial x^\mu$. However, this is obviously dependent on our choice of coordinates, as a set of local coordinates can always be scaled. Any physical theory must be independent of our choice of coordinates, so our next task is to define the properties of a smooth manifold in a coordinate independent way.

[1] To be rigorous, one has to restrict the domains and image of the coordinate function when combining them.

[2] The maximal condition is to ensure that two equivalent atlases correspond to the same differentiable manifold. A single manifold can be combined with different maximal atlases, also called differentiable structures.

Vectors and tensors

A curve γ through $\mathcal{M}$ is a function from $\mathbb{R}$ to $\mathcal{M}$,

$$\gamma : \mathbb{R} \longmapsto \mathcal{M} \tag{1.6}$$
$$\lambda \longmapsto \gamma(\lambda). \tag{1.7}$$

Such curves are often denote only by their coordinates and the parameter λ, $x^\mu(\lambda) = (x^\mu \circ \gamma)(\lambda)$. With this curve we, can take the directional derivative of a real-valued function on the manifold, $f : \mathcal{M} \mapsto \mathbb{R}$. Assume $\gamma(\lambda = 0) = p$. As we are always taking the derivative of functions between $\mathbb{R}^n$, for different n, we can use the chaine rule. The directional derivative of f at p, given by this curve γ, is then

$$\frac{\mathrm{d}}{\mathrm{d}\lambda} f(x(\lambda))\bigg|_p = \frac{\mathrm{d}x^\mu}{\mathrm{d}\lambda}\bigg|_{\lambda=0} \frac{\partial}{\partial x^\mu} f(x)\bigg|_p . \tag{1.8}$$

The set of all such directional derivatives, $\mathrm{d}/\mathrm{d}\lambda$ at p, form a vector space, $T_p\mathcal{M}$, called the *tangent space*. The coordinates x^μ induce a basis of this vector space,

$$e_\mu = \frac{\partial}{\partial x^\mu} = \partial_\mu, \tag{1.9}$$

so any element $v \in T_p\mathcal{M}$ can be written

$$v = v^\mu \partial_\mu = \frac{\mathrm{d}x^\mu}{\mathrm{d}\lambda}\bigg|_{\lambda=0} \frac{\partial}{\partial x^\mu}\bigg|_p . \tag{1.10}$$

Here, λ is the parameter of the curve corresponding to the directional derivative v.[3] We assume $\lambda = 0$ corresponds to p. The evaluation at $\lambda = 0$ and p will often be implicit, for ease of notation. This directional derivative acts on funcitons $f : \mathcal{M} \mapsto \mathbb{R}$ as

$$v(f) = v^\mu \partial_\mu f. \tag{1.11}$$

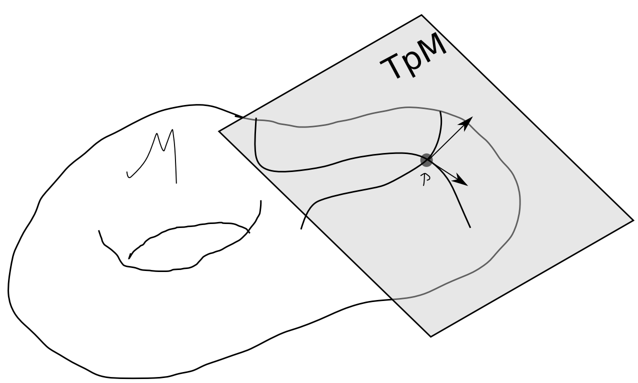

Figure 1.3: (Kladd) The tangent space $T_p\mathcal{M}$, shaded, is the sett of all directional derivatives at p. A directional derivative is defined in terms of a curve which passes through p.

A map F between two manifolds $\mathcal{M}$ and $\mathcal{N}$ also induces a map between the tangent spaces of these manifolds. This is the *differential* of F at p,

$$\mathrm{d}F_p : T_p\mathcal{M} \longmapsto T_p\mathcal{N}, \tag{1.12}$$
$$v \longmapsto \mathrm{d}F_p(v). \tag{1.13}$$

As $\mathrm{d}F_p(v)$ is an element of $T_p\mathcal{N}$, it is a directional derivative on $\mathcal{N}$, defined as

$$\mathrm{d}F_p(v)(g) = v(g \circ F), \tag{1.14}$$

[3] There is not only one curve corresponding to any directional derivative, but rather an equivalence class.

for functions $g : \mathcal{N} \mapsto \mathbb{R}$. It thus acts on functions on $\mathcal{N}$ by "extending" the derivative v. This is a linear map between vector spaces and may be written on component form by considering the differentials of the coordinate functions. Denote the coordinates of $\mathcal{N}$ by y^μ, and $y^\mu \circ F = F^\mu$. Then,

$$\mathrm{d}F_p(\partial_\mu)(g) = \partial_\mu(g \circ F)|_p = \left.\frac{\partial F^\nu}{\partial x^\mu}\right|_p \left.\frac{\partial g}{\partial y^\nu}\right|_{F(p)}, \tag{1.15}$$

or more suggestivley

$$\mathrm{d}F\left(\frac{\partial}{\partial x_\mu}\right) = \frac{\partial F^\mu}{\partial x^\nu}\frac{\partial}{\partial y^\mu}. \tag{1.16}$$

The differential is thus a generalization of the Jacobian of a function. In the case of a real valued funciton, $f : \mathcal{M} \mapsto \mathbb{R}$, and $g : \mathbb{R} \mapsto \mathbb{R}$, we get

$$\mathrm{d}f(v)(g) = v(g \circ f) = (v^\mu \partial_\mu f)\,\frac{\mathrm{d}g}{\mathrm{d}y}, \tag{1.17}$$

Gives us a natural map from vectors v to real numbers,

$$\mathrm{d}f(v) = \mathrm{d}f(v)(y) = v^\mu \partial_\mu f. \tag{1.18}$$

The set of all such maps form the *dual space* of $T_p\mathcal{M}$, denoted $T_p^*\mathcal{M}$. This is a vector space with the same dimensionality as $T_p\mathcal{M}$. We can regard each of the coordinate functions as real-valued functions, with a corresponding differential. This differential obeys

$$\mathrm{d}x^\mu(\partial_\nu) = \frac{\partial x^\mu}{\partial x_\nu} = \delta^\mu_\nu. \tag{1.19}$$

The differentials of the coordinate functions thus form a basis for $T_p^*\mathcal{M}$, called the dual basis. Using this, we can assume $\mathrm{d}f = \omega_\mu \mathrm{d}x^\nu$ for some components ω_μ, and find that $\omega_\mu = \partial_\mu f$. Or, in other words, we recover a rigorous justification for the classical expression

$$\mathrm{d}f = \frac{\partial f}{\partial x^\mu}\mathrm{d}x^\mu, \tag{1.20}$$

however we now interpret it as a covector-field instead of an "infinitesimal displacement".

Linear map from vectors to real numbers is generalized by *tensors*. Given a vector space V, general (n, m) tensor T is a multilinear map, which associates n elements from V and m from its dual V^* to the real numbers, i.e.,

$$T : V \times V \times \cdots \times V^* \times \ldots \longmapsto \mathbb{R}, \tag{1.21}$$

$$(v, u \ldots; \omega, \ldots) \longmapsto T(v, u, \ldots; \omega, \ldots). \tag{1.22}$$

Multilinear means that T is linear in each argument. The set of all such maps is the tensor product space $V \otimes V \otimes \cdots \otimes V^* \otimes \ldots$, a $\dim(V)^{n+m}$-dimensional vector space. If $\{e_\mu\}$ and $\{e^\mu\}$ are the basis for V and V^*, then we can write the basis of this of the tensor product space as $\{e_\mu \otimes \cdots \otimes e^\nu \otimes \ldots\}$. The tensor can thus be written

$$T = T^{\mu\nu\ldots}{}_{\rho\ldots}\, e_\mu \otimes e_\nu \otimes \ldots e^\rho \otimes \ldots, \tag{1.23}$$

where

$$T^{\mu\nu\ldots}{}_{\rho\ldots} = T(e^\mu e^\nu, \ldots; e_\rho, \ldots). \tag{1.24}$$

Geometries and the metric

The metric is a symmetric, non-degenerate $(0, 2)$ tensor

$$\mathrm{d}s^2 = g_{\mu\nu}\,\mathrm{d}x^\mu \otimes \mathrm{d}x^\nu. \tag{1.25}$$

It defines the geometry of the manifold $\mathcal{M}$, and is the main object of study in general relativity. As it is invertible, we can define $g^{\mu\nu} = (g^{-1})_{\mu\nu}$, which is the components of a $(2, 0)$ tensor. We use this to raise and lower indices, as is done with the Minkowski metric $\eta_{\mu\nu}$ in special relativity.

Up until now, we have only considered the tangent space $T_p\mathcal{M}$ at a point p. We are, however, more interested in fields of vectors, covectors, and tensors. A tensor field T takes the a value $T(p)$ in the tensor product space corresponding to the tangent space at $T_p\mathcal{M}$. We will use a vector field to illustrate. This vector field can be written as

$$v(p) = v^\mu(p)\partial_\mu|_p. \tag{1.26}$$

We will mostly be working with the components v^μ, which are functions of $\mathcal{M}$. For ease of notation, we write the vector as a function of the coordinates x. The vector field $v(x)$ is unchanged by a coordinate-transformation $x^\mu \to x'^\mu$; the coordinate is only for our convenience. However, with a new set of coordinates, we get a new set of basis vectors, ∂'_μ:

$$v = v^\mu\partial_\mu = v^\mu\frac{\partial x'^\nu}{\partial x^\mu}\partial'_\nu = v'^\mu\partial'_\mu, \tag{1.27}$$

This gives us the transformation rules for the components of vectors,

$$v'^\mu = \frac{\partial x'^\mu}{\partial x^\nu}v^\nu. \tag{1.28}$$

Tangent vectors are also called *contravariant* vectors, as their components transform contra to to basis vectors. For covectors, it is

$$\omega'_\mu = \frac{\partial x^\nu}{\partial x'^\mu}\omega_\nu, \tag{1.29}$$

which is why covectors also are called *covariant* vectors.

The gradient of a scalar function f, $\mathrm{d}f = \partial_\mu f \mathrm{d}x^\mu$, is a coordinate-independent derivative, as $\partial_\mu f$ follows the transformation law for covectors. We define the covariant derivative, ∇, as a map from (n, m) tensor fields to $(n, m+1)$ tensor fields. When considering a scalar as a $(0, 0)$ tensor, we see that this generalizes the scalar derivative. Of the covariant derivative, we assume

- Linearity: $\nabla(T + S) = \nabla T + \nabla S$.
- The product rule: $\nabla(T \otimes S) = (\nabla T) \otimes S + T \otimes (\nabla S)$.
- Reduces to partial derivative for scalars: $\nabla_\mu f = \partial_\mu f$.
- Krönecker delta gives zero: $\nabla_\mu \delta^\rho_\nu = 0$.

With this, we can, in general, write the covariant derivative as [1]

$$\nabla_\mu v^\nu = \partial_\mu v^\mu + \Gamma^\mu_{\nu\rho}v^\rho, \tag{1.30}$$

$$\nabla_\mu \omega_\nu = \partial_\mu \omega_\nu - \Gamma^\rho_{\mu\nu}\omega_\rho, \tag{1.31}$$

for vectors and covectors. $\Gamma^\mu_{\nu\rho}$ are called *Christoffel symbols*. The generalization for higher-order tensors is straight forward,

$$\nabla_\mu T^{\nu\dots}{}_{\rho\dots} = \partial_\mu T^{\nu\dots}{}_{\rho\dots} + \Gamma^\mu_{\nu\lambda}T^{\lambda\dots}{}_{\rho\dots} + \dots - \Gamma^\lambda_{\mu\rho}T^{\mu\dots}{}_{\lambda\dots} - \dots \tag{1.32}$$

This is still not enough to uniquely determine the covariant derivative. We will furhtermore assume $\Gamma^\lambda_{\mu\nu} = \Gamma^\lambda_{\nu\mu}$ and $\nabla_\mu g_{\nu\rho} = 0$. With these, we can find an explicit formula of the Christoffel symbols in terms of the metric,

$$\Gamma^\rho_{\mu\nu} = \frac{1}{2}g^{\rho\sigma}(\partial_\mu g_{\nu\sigma} - \partial_\sigma g_{\mu\nu} + \partial_\nu g_{\sigma\mu}). \tag{1.33}$$

The curvature of the manifold $\mathcal{M}$, with the metic $g_{\mu\nu}$, is encoded in the Riemann tensor. It is defined by

$$[\nabla_\mu, \nabla_\nu]v^\rho = R^\rho{}_{\sigma\mu\nu}v^\sigma, \tag{1.34}$$

which in our case gives the explicit formula

$$R^\rho{}_{\sigma\mu\nu} = \partial_\mu\Gamma^\rho_{\nu\sigma} - \partial_\nu\Gamma^\rho_{\mu\sigma} + \Gamma^\rho_{\mu\lambda}\Gamma^\lambda_{\nu\sigma} - \Gamma^\rho_{\nu\lambda}\Gamma^\lambda_{\mu\sigma} \tag{1.35}$$

Although the Christoffel symbols are not tensors, this quantity is a well-defined tensor due to its definition using covariant derivatives. We can therefore contract some of its indices to get other tensor quantities. We define the Ricci tensor and Ricci scalar as

$$R_{\mu\nu} = R^{\rho}{}_{\mu\rho\nu}, \tag{1.36}$$
$$R = R^{\mu}{}_{\mu} = g^{\mu\nu} R_{\mu\nu}. \tag{1.37}$$

These are the quantities we need to start working with general relativity.

Integration on manifolds

The integral of a scalar function on a manifold is not a coordinate-independent notion, and we must introduce the notion of n-forms. A n-form is a antisymmetric $(0, n)$ tensor. To ease notation, we introduce the symmetrization of a tensor T,

$$T_{(\mu_1 \cdots \mu_n)} = \frac{1}{n!} \sum_{\sigma \in S_n} T_{\mu_{\sigma(1)} \cdots \mu_{\sigma(n)}}, \tag{1.38}$$

where S_n is the set of all permutations of n objects. The antisymmetrization of a tensor is defined as

$$T_{[\mu_1 \cdots \mu_n]} = \frac{1}{n!} \sum_{\sigma \in S_n} \operatorname{sgn}(\sigma) T_{\mu_{\sigma(1)} \cdots \mu_{\sigma(n)}}. \tag{1.39}$$

The function $\sigma = \pm 1$, depending on if σ is a even or odd permutation.

We are interested in a coordinated independent quantity that we can integrate over. To that end, we define

$$\mathrm{d}^n x := \mathrm{d}x^0 \wedge \cdots \wedge \mathrm{d}x^{n-1}. \tag{1.40}$$

Here, $\wedge$ is the wedge product, defined as

$$(A \wedge B)_{\mu_1 \cdots \mu_{n+m}} = \frac{(n+m)!}{n!m!} A_{[\mu_1 \cdots \mu_n} B_{\mu_{n+1} \cdots \mu_{n+m}]}, \tag{1.41}$$

and $\mathrm{d}x^{\mu}$ is the one-form corresponding to the x^{μ}-coordinate function. Given a different set of coordinates, x'^{μ}, these are related by

$$\mathrm{d}^n x = \det\left(\frac{\partial x}{\partial x'}\right) \mathrm{d}^n x', \tag{1.42}$$

by the properties of the wedge product. We define $|g| = |\det(g_{\mu\nu})|$, which, by the transformation properties of tensors, transforms as

$$\sqrt{|g'|} = \left|\det\left(\frac{\partial x'}{\partial x}\right)\right| \sqrt{|g'|}, \tag{1.43}$$

This means that we can use this to compensate for the transformation of $\mathrm{d}^n x$, and get a volume form with a coordinate independent expression,

$$\mathrm{d}V = \sqrt{|g|}\, \mathrm{d}^n x = \sqrt{|g'|}\, \mathrm{d}^n x'. \tag{1.44}$$

With this, we can integrate scalars in a well-defined way, by mapping them to a corresponding n-form, $f \to f\mathrm{d}V$. We define the integral of a scalar function f on a manifold $\mathcal{M}$ with a metric g as

$$I = \int_{\mathcal{M}} \mathrm{d}V\, f = \int_{\mathcal{M}} \mathrm{d}^n x\, \sqrt{|g(x)|}\, f(x). \tag{1.45}$$

Stoke's thorem generalizes the fundamental theorem of calculus, as well as the divergence theorem, to manifolds. The most general statement of the theoem uses the exterior derivative, a maps from n-forms to $n+1$-forms, defined by

$$(\mathrm{d}T)_{\mu_1 \cdots \mu_{n+1}} = (n+1) \partial_{[\mu_1} T_{\mu_2 \cdots \mu_{n+1}]}. \tag{1.46}$$

Let $\mathcal{M}$ be a differential manifold of dimension n, with the boundary $\partial\mathcal{M}$. Stoeks theorem is then that, for a $n-1$-form ω,

$$\int_{\mathcal{M}} \mathrm{d}\omega = \int_{\partial\mathcal{M}} \omega. \tag{1.47}$$

Stoke's theorem then implies a generalized divergence theorem. The boundary of $\mathcal{M}$ is then $n-1$ dimensional, and a metric g on $\mathcal{M}$ will induce a new metric γ on $\partial\mathcal{M}$. Furthermore, there will be a vector field n^μ of normalized vectors orthogonal to all elements of $T\partial\mathcal{M}$. This theorem states that, for a vector field V^μ on $\mathcal{M}$,

$$\int_{\mathcal{M}} \mathrm{d}^n x\,\sqrt{|g|}\,\nabla_\mu V^\mu = \int_{\partial\mathcal{M}} \mathrm{d}^{n-1}y\,\sqrt{|\gamma|}\,n_\mu V^\mu. \tag{1.48}$$

1.2 General relativity

General relativity describes spacetime as a smooth manifold $\mathcal{M}$, with a (sudo-Rimanian) metric, $g_{\mu\nu}$. This metric is treated as a dynamical field, which is affected by the presence of matter and energy. The matter and energy contents of spacetime are encoded in the stress-energy tensor $T_{\mu\nu}$, while the behavior of $g^{\mu\nu}$ is encoded in a scalar Lagrangian density. The most obvious—and correct—choice is to use the Ricci scalar, which results in the Einstein-Hilbert action,

$$S_{\mathrm{EH}} = k\int_{\mathcal{M}} \mathrm{d}^n x\,\sqrt{|g|}\,R, \tag{1.49}$$

where k is a constant. This constant can then be related to Newton's constant of gravitation by

$$k = \frac{1}{16\pi G}. \tag{1.50}$$

The total action will include contributions from other fields with an action S_{m}, so that the total action is

$$S = S_{\mathrm{EH}} + S_{\mathrm{m}}. \tag{1.51}$$

The equations of motion of the dynamical field, which in this case is the metric, are given by Hamilton's principle of stationary action. Using functional derivatives, as defined in (REF:APPENDIX), this is stated as

Ha appendix på functional derivatives

$$\frac{1}{\sqrt{|g|}}\frac{\delta S}{\delta g^{\mu\nu}} = 0, \tag{1.52}$$

We define

$$T_{\mu\nu} = -\frac{2}{\sqrt{|g|}}\frac{\delta S_{\mathrm{m}}}{\delta g^{\mu\nu}}. \tag{1.53}$$

This results in the equations of motion for the metric, the Einstein field equations

Utlede?

$$R_{\mu\nu} - \frac{1}{2}Rg_{\mu\nu} = \kappa T_{\mu\nu}, \tag{1.54}$$

where $\kappa = 8\pi G$. The left-hand side of the Einstein field equations is called the Einstein tensor,

$$G_{\mu\nu} = R_{\mu\nu} - \frac{1}{2}Rg_{\mu\nu}. \tag{1.55}$$

This tensor obeys the identity

$$\nabla^\mu G_{\mu\nu} = 0, \tag{1.56}$$

as a consequence of the more general Bianchi identity.

Spherically symmetric spacetime

To model stars, we will assume that the metric is spherically symmetric and time-independent. In this case, the most general metric can be written, at least locally, as [1]

$$\mathrm{d}s^2 = e^{2\alpha(r)}\mathrm{d}t^2 - e^{2\beta(r)}\mathrm{d}r^2 - r^2(\mathrm{d}\theta^2 + \sin^2\theta\,\mathrm{d}\varphi^2), \tag{1.57}$$

where α and β are general functions of the radial coordinate r. In matrix form, this corresponds to

$$g_{\mu\nu} = \begin{pmatrix} e^{2\alpha(r)} & 0 & 0 & 0 \\ 0 & -e^{2\beta(r)} & 0 & 0 \\ 0 & 0 & -r^2 & 0 \\ 0 & 0 & 0 & -r^2\sin^2(\theta) \end{pmatrix}. \tag{1.58}$$

Using Eq. (1.33), we can now compute the Christoffel symbols in terms of the unknown funcitons. These computations in this subsection are done using computer code, which is shown in Appendix A. The results are

$$\Gamma^t_{\mu\nu} = \begin{pmatrix} 0 & \frac{d}{dr}\alpha(r) & 0 & 0 \\ \frac{d}{dr}\alpha(r) & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}, \tag{1.59}$$

$$\Gamma^r_{\mu\nu} = \begin{pmatrix} e^{2\alpha(r)}e^{-2\beta(r)}\frac{d}{dr}\alpha(r) & 0 & 0 & 0 \\ 0 & \frac{d}{dr}\beta(r) & 0 & 0 \\ 0 & 0 & -re^{-2\beta(r)} & 0 \\ 0 & 0 & 0 & -re^{-2\beta(r)}\sin^2(\theta) \end{pmatrix}, \tag{1.60}$$

$$\Gamma^\theta_{\mu\nu} = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{r} & 0 \\ 0 & \frac{1}{r} & 0 & 0 \\ 0 & 0 & 0 & -\sin(\theta)\cos(\theta) \end{pmatrix}, \tag{1.61}$$

$$\Gamma^\phi_{\mu\nu} = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{r} \\ 0 & 0 & 0 & \frac{\cos(\theta)}{\sin(\theta)} \\ 0 & \frac{1}{r} & \frac{\cos(\theta)}{\sin(\theta)} & 0 \end{pmatrix}. \tag{1.62}$$

With these result into Eq. (1.35) gives the Riemann tensor curvature tensor. We can then obtain the Ricci tensor by taking the trace, as shown in Eq. (1.36). The results are

$$R_{tt} = \left(r\left(\frac{d}{dr}\alpha(r)\right)^2 - r\frac{d}{dr}\alpha(r)\frac{d}{dr}\beta(r) + r\frac{d^2}{dr^2}\alpha(r) + 2\frac{d}{dr}\alpha(r)\right)\frac{e^{2\alpha(r)}e^{-2\beta(r)}}{r}, \tag{1.63}$$

$$R_{rr} = -\frac{1}{r}\left(r\left(\frac{d}{dr}\alpha(r)\right)^2 - r\frac{d}{dr}\alpha(r)\frac{d}{dr}\beta(r) + r\frac{d^2}{dr^2}\alpha(r) - 2\frac{d}{dr}\beta(r)\right), \tag{1.64}$$

$$R_{\theta\theta} = -\left(r\frac{d}{dr}\alpha(r) - r\frac{d}{dr}\beta(r) - e^{2\beta(r)} + 1\right)e^{-2\beta(r)}, \tag{1.65}$$

$$R_{\varphi\varphi} = R_{\theta\theta}\sin^2(\theta). \tag{1.66}$$

All other components are zero. Finally, the trace of the Ricci tensor gives the Ricci scalar,

$$R = \frac{2e^{-2\beta(r)}}{r^2}\left[r^2\left(\frac{d}{dr}\alpha(r)\right)^2 - r^2\frac{d}{dr}\alpha(r)\frac{d}{dr}\beta(r) + r^2\frac{d^2}{dr^2}\alpha(r) + 2r\frac{d}{dr}\alpha(r) - 2r\frac{d}{dr}\beta(r) - e^{2\beta(r)} + 1\right]. \tag{1.67}$$

The unknown functions α and β are now determined by the matter and energy content of the universe, which is encoded in $T_{\mu\nu}$, through Einstein's field equation, Eq. (1.54).

1.3 The TOV equation

We will model a star as being made up of a *perfect fulid*, with energy density ρ and pressure p. The relationship between the pressure and energy density of a substance is called the *equation of state*, or EOS, and has the form

$$f(p, \rho, \{\xi_i\}) = 0, \tag{1.68}$$

where $\{\xi_i\}$ are possible other thermodynamic variables. We will be working at zero temperature, in which case there are no other free thermodynamic variables. This allows us to, at least locally, express the pressure as a function of the energy density, $p = p(\rho)$. The stress-energy tensor of a perfect fluid is

Forklar

$$T_{\mu\nu} = (\rho + p)u_\mu u_\nu - p g_{\mu\nu}, \tag{1.69}$$

where u_μ is the 4-velocity of the fluid. In the rest frame of the fluid, we may write

$$u_\mu = (u_0, 0, 0, 0)\,. \tag{1.70}$$

This, together with the normalization condition of 4-velocities, $u_\mu u^\mu = 1$, allows us to calculate that

$$u_\mu u^\mu = g^{\mu\nu} u_\mu u_\nu = g^{00}(u_0)^2 = 1. \tag{1.71}$$

Using Eq. (1.58), we see that

$$u_0 = e^{\alpha(r)}. \tag{1.72}$$

This gives us the stress-energy tensor of the perfect fluid in its rest frame,

$$T_{\mu\nu} = \begin{pmatrix} \rho(r)e^{2\alpha(r)} & 0 & 0 & 0 \\ 0 & p(r)e^{2\beta(r)} & 0 & 0 \\ 0 & 0 & p(r)r^2 & 0 \\ 0 & 0 & 0 & p(r)r^2\sin^2(\theta) \end{pmatrix}. \tag{1.73}$$

We will use the tt and rr components of the Einstein field equations, which are

$$8\pi G r^2 \rho(r) e^{2\beta(r)} = 2r\frac{d}{dr}\beta(r) + e^{2\beta(r)} - 1 \tag{1.74}$$

$$8\pi G r^2 p(r) e^{2\beta(r)} = 2r\frac{d}{dr}\alpha(r) - e^{2\beta(r)} + 1. \tag{1.75}$$

In analogy with the Schwartzhild metmric, we define the function $m(r)$ by

$$e^{2\beta(r)} = \left(1 - \frac{2Gm(r)}{r}\right)^{-1}. \tag{1.76}$$

Substituting this into Eq. (1.74) yields

$$\frac{\mathrm{d}m(r)}{\mathrm{d}r} = 4\pi r^2\,\rho(r). \tag{1.77}$$

The solution is simply

$$m(r) = 4\pi \int_0^r \mathrm{d}r'\, {r'}^2 \rho(r'). \tag{1.78}$$

We see that $m(r)$ is the matter content contained within a radius r. If $\rho = 0$ for $r > R$ and $m(r > R) = M$, then the metric on a constant-time surface, i.e. $\mathrm{d}t = 0$, is

$$\mathrm{d}s^2 = \left(1 - \frac{2GM}{r^2}\right)^{-1} \mathrm{d}r^2 + r^2(\mathrm{d}\theta^2 + \sin^2\theta\,\mathrm{d}\varphi^2), \tag{1.79}$$

which is the same as the for the Schwartzhild solution.

Using the Bianchi identity, Eq. (1.56), together with EInstein's equation, we find that

$$\nabla^\mu G_{\mu\nu} = \nabla^\mu T_{\mu\nu} = 0. \tag{1.80}$$

The r-component of this equation is

$$\nabla_\mu T^{\mu r} = \partial_r T^{rr} + \Gamma^\mu_{\mu\nu} T^{\nu r} + \Gamma^r_{\mu\nu} T^{\mu\nu}, \tag{1.81}$$

where we have used the particular form of $T_{\mu\nu}$ and the Christoffel symbols to eliminate vanishing terms. We calculate

$$\begin{aligned}\nabla_\mu T^{\mu r} &= \partial_r \left(p e^{-2\beta}\right) + (2\Gamma^r_{rr} + \Gamma^t_{tr}) T^{rr} + \Gamma^r_{tt} T^{tt} \\ &= e^{-2\beta} \left(\partial_r p + p \partial_r \alpha + \rho \partial_r \alpha\right) = 0.\end{aligned}$$

This allows us to relate α to p and ρ, via

$$\partial_r \alpha = -\frac{\partial_r p}{p + \rho} \tag{1.82}$$

When we substitute this, together with the definition of $m(r)$, into Eq. (1.75), we obtain

$$\frac{\mathrm{d}p}{\mathrm{d}r} = -\frac{G\left(4\pi r^3 p + m\right)(\rho + p)}{r\,(r - 2Gm)}, \tag{1.83}$$

the Tolman-Oppenheimer-Volkoff (TOV) equation.

To summarize, we have three functions describeing the star, ρ, p, and m, as well as thre equations, Eq. (1.68), Eq. (1.78) and Eq. (1.83). Together with inital conditions, such as $p(0) = p_0$, this is enough to determin the unknown funcions. For completness, the equations are

$$\begin{aligned} f(p, \rho) &= 0, \\ m(r) &= 4\pi \int_0^r \mathrm{d}r\, {r'}^2 \rho(r'), \\ \frac{\mathrm{d}p}{\mathrm{d}r} &= -\frac{G\left(4\pi r^3 p + m\right)(\rho + p)}{r\,(r - 2Gm)}. \end{aligned}$$

Appendix A

Code

All code is available at: `https://github.com/martkjoh/master`.

The code for calculations in chapter 1 is included below. The code is written in Python in a Jupyter notebook, and the `.ipynb` file with executable code is available in the repository linked above.

```
[17]: from sympy import MatrixSymbol, Matrix, Array, pprint
      from sympy import symbols, diff, exp, log, cos, sin, simplify, Rational
      from sympy.core.symbol import Symbol
      from sympy import pi

      import numpy as np
      import sympy as sp
      from IPython.display import display, Latex
```

Tensor operations

```
[18]: def INDX(i, place, num_indx):
          """
          Acceses an index at 'place' for 'num_indx' order tensor
          T_(a0 ... âp ... an-1) = T[INDX(i, place=p, num_indx=n)] = T[:,...<-p-> ,␣
       ↪i, :,...<-(n-p-1)->]
          """
          indx = []
          assert place<num_indx
          for j in range(num_indx):
              if place==j: indx.append(i)
              else: indx.append(slice(None))
          return tuple(indx)
```

```
[19]: def contract(T, g=None, g_inv=None, num_indx=2, upper=1, indx=(0, 1)):
          """
          contracts indecies indx=(a_p, a_q) on tensor T with 'num_indx',
          'upper' of whom are upper. If upper=0, all indecies are assumed lower.
          With indx=(a_k, a_l), upper=n, num_indx=n+m, this gives
          T^(a_0...a_n-1)_(a_n...a_n+m-1) -> T^(a_0...a_k=a...a_n-1)_(a_n...a_k...
       ↪a_n+m-1),
          with the necesarry metric. If wrong metric is given, this wil throw error.
          """
          assert indx[0] < indx[1]  # we have to know if the index to the left␣
       ↪dissapears
          dim = np.shape(T)[0]
          a = (indx[0] < upper) + (indx[1] < upper) # number of upper indecies to be␣
       ↪contracted
          if a==2: g0 = g # two upper
          elif a==0: g0 = g_inv # two lower
          else: g0 = np.identity(dim, dtype=Rational)

          Tc = Rational(0) * np.ones((T.shape)[:-2], dtype=Rational)
          for i in range(dim):
              for j in range(dim):
                  Tc += g0[i, j] * (T[INDX(i, indx[0], num_indx)])[INDX(j, indx[1] -␣
       ↪1, num_indx - 1)]
```

```python
    return Tc

def raise_indx(T, g_inv, indx, num_indx):
    """
    Raise index 'indx' of a tensor T with 'num_indx' indices.
    """
    dim = np.shape(T)[0]
    Tu = np.zeros_like(T)
    for i in range(dim):
        I = INDX(i, indx, num_indx)
        for j in range(dim):
            J = INDX(j, indx, num_indx)
            Tu[I] += g_inv[i, j] * T[J]
    return Tu

def lower_indx(T, g, indx, num_indx):
    return raise_indx(T, g, indx, num_indx)

def get_g_inv(g):
    return np.array(Matrix(g)**(-1))
```

Calculate Christoffel symbols and Riemann curvature tensor

[20]:
```python
def Christoffel(g, g_inv, var):
    """
    Work out the christoffel symbols, given a metric an its variables
    Γ^i_jk = C[i, j, k]
    """
    dim = len(var)
    C = np.zeros((dim, dim, dim), dtype=Symbol)
    for i in range(dim):
        for j in range(dim):
            for k in range(dim):
                for m in range(dim):
                    C[i, j, k] += Rational(1, 2) * (g_inv)[i, m] * (
                        diff(g[m, k], var[j])
                        + diff(g[m, j], var[k])
                        - diff(g[k, j], var[m])
                    )

    return C
```

[21]:
```python
def Riemann_tensor(C, var):
    """
    Riemann_tensor(Christoffel_symbols, (x_1, ...)) = R[i, j, k, l] = R^i_jkl
    Compute the Riemann tensor from the Christoffel symbols
```

```
    """
    dim = len(var)
    R = np.zeros([dim] * 4, dtype=Symbol)
    indx = [(i, j, k, l)
        for i in range(dim)
        for j in range(dim)
        for k in range(dim)
        for l in range(dim)
    ]

    for (a, b, r, s) in indx:
        R[a, b, r, s] += diff(C[a, b, s], var[r]) - diff(C[a, b, r], var[s])
        for k in range(dim):
            R[a, b, r, s] += C[a, k, r] * C[k, b, s] - C[a, k, s] * C[k, b, r]
    return R
```

Printing functions

```
[22]: print_latex = False

def print_christoffel(C, var):
    """ A function for dsiplaying christoffels symbols """
    output = []
    for i in range(len(var)):
        txt = "$$"
        txt += "\\Gamma^" + sp.latex(var[i]) + "_{\\mu \\nu} ="
        txt += sp.latex(Matrix(C[i]))
        txt += "$$"
        print(txt) if print_latex else print()
        output.append(display(Latex(txt)))

    return output

def print_matrix(T):
    txt = "$$" + sp.latex(Matrix(T)) +"$$"
    print(txt) if print_latex else print()
    return display(Latex(txt))

def print_scalar(T):
    txt = "$$" + sp.latex(T) +"$$"
    print(txt) if print_latex else print()
    return display(Latex(txt))

def print_eq(eq):
    txt = "$$" + sp.latex(eq) +"=0" + "$$"
    print(txt) if print_latex else print()
    return display(Latex(txt))
```

Metric $g_{\mu\nu}$ for spherically symmetric spacetime

[23]:
```
t, r, th, ph = symbols("t, r, \\theta, \\phi")
x1 = r * cos(ph) * sin(th)
x2 = r * sin(ph) * sin(th)
x3 = r * cos(th)

one = Rational(1)
eta = sp.diag(one, -one, -one, -one)
var = (t, r, th, ph)
J = Matrix([t, x1, x2, x3]).jacobian(var)
g = np.array(simplify(J.T *eta* J))

a = sp.Function("\\alpha", real=True)(r)
b = sp.Function("\\beta", real=True)(r)
g[0, 0] *= exp(2 * a)
g[1, 1] *= exp(2 * b)
g_inv = get_g_inv(g)

print_matrix(g)
print_matrix(g_inv)
```

$$\begin{bmatrix} e^{2\alpha(r)} & 0 & 0 & 0 \\ 0 & -e^{2\beta(r)} & 0 & 0 \\ 0 & 0 & -r^2 & 0 \\ 0 & 0 & 0 & -r^2\sin^2(\theta) \end{bmatrix}$$

$$\begin{bmatrix} e^{-2\alpha(r)} & 0 & 0 & 0 \\ 0 & -e^{-2\beta(r)} & 0 & 0 \\ 0 & 0 & -\frac{1}{r^2} & 0 \\ 0 & 0 & 0 & -\frac{1}{r^2\sin^2(\theta)} \end{bmatrix}$$

[24]:
```
C = Christoffel(g, g_inv, var)
c = print_christoffel(C, var)
```

$$\Gamma^t_{\mu\nu} = \begin{bmatrix} 0 & \frac{d}{dr}\alpha(r) & 0 & 0 \\ \frac{d}{dr}\alpha(r) & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

$$\Gamma^{r}_{\mu\nu} = \begin{bmatrix} e^{2\alpha(r)}e^{-2\beta(r)}\frac{d}{dr}\alpha(r) & 0 & 0 & 0 \\ 0 & \frac{d}{dr}\beta(r) & 0 & 0 \\ 0 & 0 & -re^{-2\beta(r)} & 0 \\ 0 & 0 & 0 & -re^{-2\beta(r)}\sin^2(\theta) \end{bmatrix}$$

$$\Gamma^{\theta}_{\mu\nu} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{r} & 0 \\ 0 & \frac{1}{r} & 0 & 0 \\ 0 & 0 & 0 & -\sin(\theta)\cos(\theta) \end{bmatrix}$$

$$\Gamma^{\phi}_{\mu\nu} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{r} \\ 0 & 0 & 0 & \frac{\cos(\theta)}{\sin(\theta)} \\ 0 & \frac{1}{r} & \frac{\cos(\theta)}{\sin(\theta)} & 0 \end{bmatrix}$$

[25]:
```
Rie = Riemann_tensor(C, var)
Ricci = contract(Rie, num_indx=4, upper=1, indx=(0, 2))

for i in range(4):
    print_scalar(Ricci[i, i].factor())
```

$$\frac{\left(r\left(\frac{d}{dr}\alpha(r)\right)^2 - r\frac{d}{dr}\alpha(r)\frac{d}{dr}\beta(r) + r\frac{d^2}{dr^2}\alpha(r) + 2\frac{d}{dr}\alpha(r)\right)e^{2\alpha(r)}e^{-2\beta(r)}}{r}$$

$$-\frac{r\left(\frac{d}{dr}\alpha(r)\right)^2 - r\frac{d}{dr}\alpha(r)\frac{d}{dr}\beta(r) + r\frac{d^2}{dr^2}\alpha(r) - 2\frac{d}{dr}\beta(r)}{r}$$

$$-\left(r\frac{d}{dr}\alpha(r) - r\frac{d}{dr}\beta(r) - e^{2\beta(r)} + 1\right)e^{-2\beta(r)}$$

$$-\left(r\frac{d}{dr}\alpha(r) - r\frac{d}{dr}\beta(r) - e^{2\beta(r)} + 1\right)e^{-2\beta(r)}\sin^2(\theta)$$

[26]:
```
R = contract(Ricci, g_inv=g_inv, upper=0).simplify()
print_scalar(R)
```

$$\frac{2\left(r^2\left(\frac{d}{dr}\alpha(r)\right)^2 - r^2\frac{d}{dr}\alpha(r)\frac{d}{dr}\beta(r) + r^2\frac{d^2}{dr^2}\alpha(r) + 2r\frac{d}{dr}\alpha(r) - 2r\frac{d}{dr}\beta(r) - e^{2\beta(r)} + 1\right)e^{-2\beta(r)}}{r^2}$$

[27]:
```
G = Ricci - Rational(1, 2) * R * g
for i in range(4):
    G[i, i] = G[i, i].simplify().factor()
    print_scalar(G[i, i])
```

$$\frac{\left(2r\frac{d}{dr}\beta(r) + e^{2\beta(r)} - 1\right)e^{2\alpha(r)}e^{-2\beta(r)}}{r^2}$$

$$\frac{2r\frac{d}{dr}\alpha(r) - e^{2\beta(r)} + 1}{r^2}$$

$$r\left(r\left(\frac{d}{dr}\alpha(r)\right)^2 - r\frac{d}{dr}\alpha(r)\frac{d}{dr}\beta(r) + r\frac{d^2}{dr^2}\alpha(r) + \frac{d}{dr}\alpha(r) - \frac{d}{dr}\beta(r)\right)e^{-2\beta(r)}$$

$$r\left(r\left(\frac{d}{dr}\alpha(r)\right)^2 - r\frac{d}{dr}\alpha(r)\frac{d}{dr}\beta(r) + r\frac{d^2}{dr^2}\alpha(r) + \frac{d}{dr}\alpha(r) - \frac{d}{dr}\beta(r)\right)e^{-2\beta(r)}\sin^2(\theta)$$

0.0.1 Stress-energy tensor $T_{\mu\nu}$ for perfect fluid

[28]:
```
p = sp.Function("p")(r)
rho = sp.Function("\\rho")(r)

UU = np.zeros((4, 4), dtype=sp.Rational)
UU[0, 0] = exp(2 * a)

T = (p + rho) * UU - p * g
for i in range(4):
    T[i, i] = T[i, i].simplify()
print_matrix(T)
```

$$\begin{bmatrix} \rho(r)e^{2\alpha(r)} & 0 & 0 & 0 \\ 0 & p(r)e^{2\beta(r)} & 0 & 0 \\ 0 & 0 & r^2p(r) & 0 \\ 0 & 0 & 0 & r^2p(r)\sin^2(\theta) \end{bmatrix}$$

0.0.2 Einstin equation

$R_{\mu\nu} - \frac{1}{2}Rg_{\mu\nu} = 8\pi GT_{\mu\nu}$

[29]:
```
G_newton = sp.Symbol("G")

eq = []
for i in range(len(G)):
    eq.append((G[i, i] - 8 * pi * G_newton * T[i, i]).simplify())

# Some manual simplification
Rtt = sp.Symbol("R_{\\theta \\theta}")
eq[0] = eq[0] * r**2 / exp(2 * a)/exp(-2*b ) * (-1 )
eq[1] = eq[1] * r**2 * (-1)
eq[2] = eq[2] / r / exp(-2*b)
eq[3] = eq[3].subs(eq[2], Rtt)
for i in range(len(G)):
    print_eq(eq[i].simplify())
```

$$8\pi Gr^2\rho(r)e^{2\beta(r)} - 2r\frac{d}{dr}\beta(r) - e^{2\beta(r)} + 1 = 0$$

$$8\pi Gr^2p(r)e^{2\beta(r)} - 2r\frac{d}{dr}\alpha(r) + e^{2\beta(r)} - 1 = 0$$

$$-8\pi Grp(r)e^{2\beta(r)} + r\left(\frac{d}{dr}\alpha(r)\right)^2 - r\frac{d}{dr}\alpha(r)\frac{d}{dr}\beta(r) + r\frac{d^2}{dr^2}\alpha(r) + \frac{d}{dr}\alpha(r) - \frac{d}{dr}\beta(r) = 0$$

$$R_{\theta\theta}re^{-2\beta(r)}\sin^2(\theta) = 0$$

Define $e^{2\beta} = [1 - 2Gm(r)/r]^{-1}$

[30]:
```
m = sp.Function("m", Real=True)(r)
f = (1 - 2 * G_newton * m / r)**(-1)
eq1 = (eq[0] * exp(- 2 *a)).simplify().subs(b, Rational(1, 2) * log(f)).
 ↪simplify().expand().simplify()
s = sp.solve(eq1, m.diff(r))
eq1 = m.diff(r) - s[0]
```

Use $\nabla_\mu T^{\mu r} = 0 \implies (p+\rho)\partial_r \alpha = -\partial_r p$.

[31]:
```
eq2 = (eq[1] * r**2).subs(exp(2 * b), f).simplify()
s = sp.solve(eq2, a.diff(r))
eq2 = a.diff(r) - s[0]
eq2 = ((a.diff(r) - s[0]).subs(a.diff(r), - p.diff(r) / (p + rho))*(p + rho)).
 ↪simplify()
s = sp.solve(eq2, p.diff())
eq2 = p.diff(r) - s[0].factor()
```

The TOV-equation and equation for $m(r)$, both expressions are equal to 0.

[32]:
```
print_eq(eq1)
print_eq(eq2)
```

$$-4\pi r^2 \rho(r) + \frac{d}{dr} m(r) = 0$$

$$\frac{G\left(4\pi r^3 p(r) + m(r)\right)\left(\rho(r) + p(r)\right)}{r\left(-2Gm(r) + r\right)} + \frac{d}{dr} p(r) = 0$$

Bibliography

[1] S. M. Carroll. *Spacetime and Geometry: An Introduction to General Relativity*. First. Cambridge University Press, Aug. 2019. ISBN: 978-1-108-48839-6 978-1-108-77038-5. DOI: 10.1017/9781108770385.

[2] J. M. Lee. "Smooth Manifolds". In: *Introduction to Smooth Manifolds*. Ed. by J. M. Lee. Graduate Texts in Mathematics. New York, NY: Springer, 2012, pp. 1–31. ISBN: 978-1-4419-9982-5. DOI: 10.1007/978-1-4419-9982-5_1.