

```

> restart:

assume(r>0);
assume(M>0);

libname:="H:\\Desktop\\grtensor-master\\lib\\",
libname:grOptionMetricPath:="H:\\Desktop\\grtensor-
master\\metrics\\" :with(grtensor):
"GRTensor III v2.1.11 Apr 26, 2018"
"Copyright 2018, Peter Musgrave, Denis Pollney, Kayll Lake"
"Latest version is at http://github.com/grtensor/grtensor"
"For help ?grtensor"
"Support/contact grtensor3@gmail.com"

```

(1)

```

> with(orthopoly):
with(plots):

```

Start with definitions of the spherical harmonics. First are the vectors, then tensors and Omega is the metric dependent component. Following Martel and Poisson Y=even and X=odd parity while we also drop the 'l' and 'm' with 2=theta and 3=phi

```

> Y__2:=diff(Y(theta,phi), theta); Y__3:=I*m*Y(theta,phi);
X__2:=-I*m*Y(theta,phi)/sin(theta); X__3:=sin(theta)*(diff(Y
(theta,phi), theta));

```

$$\begin{aligned}
 Y_2 &:= \frac{\partial}{\partial \theta} Y(\theta, \phi) \\
 Y_3 &:= I m Y(\theta, \phi) \\
 X_2 &:= \frac{-I m Y(\theta, \phi)}{\sin(\theta)} \\
 X_3 &:= \sin(\theta) \left( \frac{\partial}{\partial \theta} Y(\theta, \phi) \right)
 \end{aligned}
 \tag{2}$$

Next tensor spherical harmonics. This time there are three: even, odd and the one proportional the two-metric which I call Omega (again MP notation). Again Y=even and X=odd parity

```

> Y__22:=((1/2)*lambda+m^2*(1/sin(theta)^2))*Y(theta,phi)-cos
(theta)*(diff(Y(theta,phi), theta))*(1/sin(theta));
Y__23:=expand((I*m*(diff(Y(theta,phi), theta))*sin(theta)-I*cos
(theta)*m*Y(theta,phi))*(1/sin(theta)));
Y__33 := (-m^2+(1/2)*(-lambda*(sin(theta)^2)))*Y(theta,phi)+sin
(theta)*cos(theta)*(diff(Y(theta,phi), theta));

```

$$\begin{aligned}
 Y_{22} &:= \left( \frac{\lambda}{2} + \frac{m^2}{\sin(\theta)^2} \right) Y(\theta, \phi) - \frac{\cos(\theta) \left( \frac{\partial}{\partial \theta} Y(\theta, \phi) \right)}{\sin(\theta)} \\
 Y_{23} &:= I m \left( \frac{\partial}{\partial \theta} Y(\theta, \phi) \right) - \frac{I \cos(\theta) m Y(\theta, \phi)}{\sin(\theta)}
 \end{aligned}$$

$$Y_{33} := \left( -m^2 - \frac{\lambda \sin(\theta)^2}{2} \right) Y(\theta, \phi) + \sin(\theta) \cos(\theta) \left( \frac{\partial}{\partial \theta} Y(\theta, \phi) \right) \quad (3)$$

Remember that  $\lambda = -l(l+1)$ . Here we add in the tensor components

```
> X__22 := -I*m*(diff(Y(theta,phi), theta))*(1/sin(theta))+I*cos
(theta)*m*Y(theta,phi)*(1/sin(theta)^2);
X__23 := (((1/2)*(sin(theta)*lambda))+m^2*(1/sin(theta)))*Y
(theta,phi)-cos(theta)*(diff(Y(theta,phi), theta));
X__33 := I*m*(diff(Y(theta,phi), theta))*sin(theta)-I*cos(theta)*
m*Y(theta,phi);
```

$$X_{22} := -\frac{Im \left( \frac{\partial}{\partial \theta} Y(\theta, \phi) \right)}{\sin(\theta)} + \frac{I \cos(\theta) m Y(\theta, \phi)}{\sin(\theta)^2}$$

$$X_{23} := \left( \frac{\sin(\theta) \lambda}{2} + \frac{m^2}{\sin(\theta)} \right) Y(\theta, \phi) - \cos(\theta) \left( \frac{\partial}{\partial \theta} Y(\theta, \phi) \right)$$

$$X_{33} := Im \left( \frac{\partial}{\partial \theta} Y(\theta, \phi) \right) \sin(\theta) - I \cos(\theta) m Y(\theta, \phi) \quad (4)$$

Final tensor harmonics

```
> Omega__22 := Y(theta,phi);
Omega__23 := 0;
Omega__33 := Y(theta,phi)*sin(theta)^2;
      Omega_22 := Y(theta,phi)
      Omega_23 := 0
      Omega_33 := Y(theta,phi)*sin(theta)^2 \quad (5)
```

'Linearize' linearizes in terms of epsilon to first order

```
> Linearize := proc (T)
    local T1;
    T1 := convert(series(T, epsilon, 2), polynom);
    simplify(coeff(T1, epsilon, 0)+epsilon*simplify
(coeff(T1, epsilon, 1)));
end proc;
```

Apply a fix to the cosine bias in the simplify command for maple

```
> FixSin := proc (T)
    local T1;
    T1 := expand(subs(cos(theta)^2=1-sin(theta)^2, expand(T)));
end proc;
```

Substitute in the Spherical Harmonics into the metric perturbations.

**This is expressed in Regge-Wheeler gauge. \*\*\*\*\*This needs to be done in general as well\*  
\*\*\*\*\*eventually**

**Gauge transformations are inputted in the last section. This section has to be revisited for the matter introductin. or does it? pretty sure it does lolz**

```
> hExpand:=proc(T)
  local T1,T2,T3,T4,T5,T6;
  T1:=subs(H__scal(r,theta,phi)=h__rr(r)*Y(theta,phi),T):
  T2:=subs(H__V1(r,theta,phi)=h__Veven(r)*Y__2+h__Vodd(r)*X__2,
  T1):
  T3:=subs(H__V2(r,theta,phi)=h__Veven(r)*Y__3+h__Vodd(r)*X__3,
  T2):
  T4:=subs(H__T11(r,theta,phi)=h__Teven(r)*Y__22+h__Todd(r)*
  X__22+h__trace(r)*Omega__22,T3):
  T5:=subs(H__T12(r,theta,phi)=h__Teven(r)*Y__23+h__Todd(r)*
  X__23+h__trace(r)*Omega__23,T4):
  T6:=subs(H__T22(r,theta,phi)=h__Teven(r)*Y__33+h__Todd(r)*
  X__33+h__trace(r)*Omega__33,T5):
  expand(eval(T6));
end proc:
```

Added a macro for my expression for Extrinsic Curvature. The grdef command would only accept these definitions k1 and not k[0] or k\_[0]

```
> kExpand:=proc(T)
  local T1,T2,T3,T4,T5,T6;
  T1:=subs(k__scal(r,theta,phi)=k__rr(r)*Y(theta,phi),T):
  T2:=subs(k__V1(r,theta,phi)=k__Veven(r)*Y__2+k__Vodd(r)*X__2,
  T1):
  T3:=subs(k__V2(r,theta,phi)=k__Veven(r)*Y__3+k__Vodd(r)*X__3,
  T2):
  T4:=subs(k__T11(r,theta,phi)=k__Teven(r)*Y__22+k__Todd(r)*
  X__22+k__trace(r)*Omega__22,T3):
  T5:=subs(k__T12(r,theta,phi)=k__Teven(r)*Y__23+k__Todd(r)*
  X__23+k__trace(r)*Omega__23,T4):
  T6:=subs(k__T22(r,theta,phi)=k__Teven(r)*Y__33+k__Todd(r)*
  X__33+k__trace(r)*Omega__33,T5):
  expand(eval(T6));
end proc:
```

ApplyId applies the scalar spherical harmonic eigenvalue equation to get rid of double theta derivatives

```
> ApplyId:=proc(T)
  local T1;
  T1:=expand(subs(diff(Y(theta,phi),theta,theta)=m^2*Y(theta,
  phi)/sin(theta)^2-cot(theta)*diff(Y(theta,phi),theta)+lambda*Y
  (theta,phi),T)):
end proc:
```

RemPhi replaces phi derivatives with I\*m\*Y.

```

> RemPhi:=proc(T)
    local T1;
    T1:=expand(subs(diff(Y(theta,phi),phi)=I*m*Y(theta,phi),T)):
end proc:

```

CollectY groups together theta derivatives and powers of m

```

> CollectY:=proc(T)
    local T1,T2,T3;
    T1:=simplify(coeff(T,diff(Y(theta,phi),theta))*diff(Y(theta,phi),theta):
    T2:=factor(FixSin(simplify(T-T1))):
    T3:=T1+T2;
end proc:

```

The next two functions take the theta and phi components of a two-vector and use them to identity the even/odd decomposition. They should be redundant but I include both so I can check for consistency.

```

> IdV2:=proc(V2)
    local Ycoeff,Xcoeff;
    Ycoeff:=expand(coeff(V2,diff(Y(theta,phi),theta)));
    Xcoeff:=expand(simplify(sin(theta)/m/Y(theta,phi)*I*(V2-Ycoeff*diff(Y(theta,phi),theta))));
    return(expand(Ycoeff/epsilon),expand(Xcoeff/epsilon));
end proc:

> IdV3:=proc(V3)
    local Ycoeff,Xcoeff;
    Xcoeff:=expand(coeff(V3,diff(Y(theta,phi),theta)));
    Ycoeff:=expand(simplify(sin(theta)/m/Y(theta,phi)*I*(V3-Xcoeff*diff(Y(theta,phi),theta))));
    return(expand(-Ycoeff/epsilon/sin(theta)),expand(Xcoeff/epsilon/sin(theta)));
end proc:

```

Next decomposing two-tensors. It's easiest to start with the theta-phi component to get the Y\_AB and X\_AB components and get the Omega\_AB component from the theta-theta. Finally we can check it all for consistency against phi-phi.

```

> IdT23:=proc(T23)
    local Ycoeff, Xcoeff;
    Ycoeff:=expand(simplify(coeff(T23,m,1)/Y__23));
    Xcoeff:=expand(simplify((T23-m*Ycoeff*Y__23)/X__23));
    return(expand(FixSin(expand(m*Ycoeff/epsilon))),expand(FixSin(expand(Xcoeff/epsilon))));
end proc:

> IdT22:=proc(T22,Ycoeff,Xcoeff)
    expand(simplify((T22-epsilon*Ycoeff*Y__22-epsilon*Xcoeff*X__22)/Omega__22/epsilon));
end proc:

> IdT33:=proc(T33,Ycoeff,Xcoeff)
    expand(simplify((T33-epsilon*Ycoeff*Y__33-epsilon*Xcoeff*X__33)/Omega__33/epsilon));
end proc:

```

## Load the Metric

```
> qload(lpschwNewFixed);
grdisplay(g(dn, dn));
Error, (in H[T11]) wrong number of arguments
Error, (in H[T11]) wrong number of arguments
Calculated ds for lpschwNewFixed (0.000000 sec.)
Default spacetime = lpschwNewFixed
For the lpschwNewFixed spacetime:
Coordinates
x(up)

$$x^a = \begin{bmatrix} r & \theta & \phi \end{bmatrix}$$

For the lpschwNewFixed spacetime:
constraints
constraint = [ $\epsilon^2=0, \epsilon^3=0, \epsilon^4=0, \epsilon^5=0, \epsilon^6=0, \epsilon^7=0, \epsilon^8=0, \epsilon^9=0, \epsilon^{10}=0$ ]
For the lpschwNewFixed spacetime:
Covariant metric tensor
g(dn, dn)
```

$$g_{ab} = \begin{bmatrix} 1 + \epsilon H_{scal}(r, \theta, \phi) & \epsilon H_{VI}(r, \theta, \phi) & \epsilon H_{V2}(r, \theta, \phi) \\ \epsilon H_{VI}(r, \theta, \phi) & r^2 + \epsilon H_{T11}(r, \theta, \phi) & \epsilon H_{T12}(r, \theta, \phi) \\ \epsilon H_{V2}(r, \theta, \phi) & \epsilon H_{T12}(r, \theta, \phi) & r^2 \sin(\theta)^2 + \epsilon H_{T22}(r, \theta, \phi) \end{bmatrix} \quad (6)$$

Calculate the inverse metric

```
> grcalc(g(up, up));
grmap(g(up, up), Linearize, 'x');
grdisplay(_);
Calculated detg for lpschwNewFixed (0.016000 sec.)
Calculated g(up,up) for lpschwNewFixed (0.032000 sec.)
CPU Time = 0.032
Procedure name:LinearizeApplying routine Linearize to g(up,up)
For the lpschwNewFixed spacetime:
Contravariant metric tensor
g(up, up)
```

$$g^{ab} = \begin{bmatrix} 1 - \epsilon H_{scal}(r, \theta, \phi) & -\frac{H_{VI}(r, \theta, \phi) \epsilon}{r^2} & -\frac{H_{V2}(r, \theta, \phi) \epsilon}{r^2 \sin(\theta)^2} \\ -\frac{H_{VI}(r, \theta, \phi) \epsilon}{r^2} & \frac{1}{r^2} - \frac{\epsilon H_{T11}(r, \theta, \phi)}{r^4} & -\frac{H_{T12}(r, \theta, \phi) \epsilon}{\sin(\theta)^2 r^4} \\ -\frac{H_{V2}(r, \theta, \phi) \epsilon}{r^2 \sin(\theta)^2} & -\frac{H_{T12}(r, \theta, \phi) \epsilon}{\sin(\theta)^2 r^4} & \frac{1}{r^2 \sin(\theta)^2} - \frac{\epsilon H_{T22}(r, \theta, \phi)}{r^4 \sin(\theta)^4} \end{bmatrix} \quad (7)$$

Need to force the definition for the perturbed extrinsic curvature

```
> grdef(`er{a}:=[1, 0, 0]`);
grdef(`et{a}:=[0, 1, 0]`);
grdef(`ep{a}:=[0, 0, 1]`);
grcalc(er(dn), et(dn), ep(dn));

grdisplay(_);

grdef(`KK{a b}:=(sqrt(2)/2*sqrt(M)/r^(3/2)+epsilon*k__scal(r,
theta,phi))*er{a}*er{b}+(epsilon*k__V1(r,theta,phi))*er{a}*et{b}+
(epsilon*k__V2(r,theta,phi))*er{a}*ep{b}+(epsilon*k__V1(r,theta,
phi))*er{b}*et{a}+(epsilon*k__V2(r,theta,phi))*er{b}*ep{a}+(-sqrt
(2)*sqrt(M)*sqrt(r)+epsilon*k__T11(r,theta,phi))*et{a}*et{b}+
(epsilon*k__T12(r,theta,phi))*et{a}*ep{b}+(epsilon*k__T12(r,
theta,phi))*et{b}*ep{a}+(-sqrt(2)*sqrt(M)*sqrt(r)*sin(theta)^2+
epsilon*k__T22(r,theta,phi))*ep{a}*ep{b}`);
grcalc(KK(dn, dn));
grdisplay(KK(dn, dn));
```

Components assigned for metric: lpschwNewFixed

Created definition for er(dn)

Components assigned for metric: lpschwNewFixed

Created definition for et(dn)

Components assigned for metric: lpschwNewFixed

Created definition for ep(dn)

CPU Time =0.

*For the lpschwNewFixed spacetime:*

$er(dn)$

$er(dn)$

$$er_a = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}$$

$ep(dn)$

$ep(dn)$

$$ep_a = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}$$

Created definition for KK(dn,dn)

Calculated KK(dn,dn) for lpschwNewFixed (0.000000 sec.)

CPU Time =0.

*For the lpschwNewFixed spacetime:*

$KK(dn,dn)$

$KK(dn, dn)$

$$KK_{ab} = \left[ \left[ \frac{2 \epsilon k_{scal}(r, \theta, \phi) r^{3/2} + \sqrt{2} \sqrt{M}}{2 r^{3/2}}, \epsilon k_{V1}(r, \theta, \phi), \epsilon k_{V2}(r, \theta, \phi) \right], \right. \\ \left[ \epsilon k_{V1}(r, \theta, \phi), -\sqrt{2} \sqrt{M} \sqrt{r} + \epsilon k_{T11}(r, \theta, \phi), \epsilon k_{T12}(r, \theta, \phi) \right], \\ \left. \left[ \epsilon k_{V2}(r, \theta, \phi), \epsilon k_{T12}(r, \theta, \phi), -\sqrt{2} \sqrt{M} \sqrt{r} \sin(\theta)^2 + \epsilon k_{T22}(r, \theta, \phi) \right] \right] \quad (8)$$

Now to form the constraint equation.

First get the extrinsic curvature squared

```
> grdef(`K2:=KK{^a ^b}*KK{a b}`);
grcalc(K2);
grmap(K2, Linearize, 'x');
grmap(K2, FixSin, 'x');
grdisplay(K2);
Created definition for KK(up,up)
Created definition for K2
Calculated KK(up,up) for lpschwNewFixed (0.016000 sec.)
Calculated K2 for lpschwNewFixed (0.000000 sec.)
CPU Time =0.016
Procedure name:LinearizeApplying routine Linearize to K2
Procedure name:FixSinApplying routine FixSin to K2
For the lpschwNewFixed spacetime:
```

$$K2 = -\frac{MH_{scal}(r, \theta, \phi) \epsilon}{r^3} + \frac{9M}{2r^3} + \frac{\sqrt{2} k_{scal}(r, \theta, \phi) \epsilon \sqrt{M}}{r^{3/2}} - \frac{2\sqrt{2} k_{T11}(r, \theta, \phi) \epsilon \sqrt{M}}{r^{7/2}} \quad (9)$$

$$- \frac{2\sqrt{2} k_{T22}(r, \theta, \phi) \epsilon \sqrt{M}}{r^{7/2} \sin(\theta)^2} - \frac{4MH_{T11}(r, \theta, \phi) \epsilon}{r^5} - \frac{4MH_{T22}(r, \theta, \phi) \epsilon}{r^5 \sin(\theta)^2}$$

Next formulate the trace of the extrinsic curvature

```
> grdef(`KTr:=g{^a ^b}*KK{a b}`);
grcalc(KTr);
grmap(KTr, Linearize, 'x');
grmap(KTr, FixSin, 'x');
grdisplay(_);
Created definition for KTr
Calculated KTr for lpschwNewFixed (0.000000 sec.)
CPU Time =0.
Procedure name:LinearizeApplying routine Linearize to KTr
Procedure name:FixSinApplying routine FixSin to KTr
For the lpschwNewFixed spacetime:
```

$$KTr = \frac{k_{T11}(r, \theta, \phi) \epsilon}{r^2} + \frac{k_{T22}(r, \theta, \phi) \epsilon}{r^2 \sin(\theta)^2} + k_{scal}(r, \theta, \phi) \epsilon - \frac{H_{scal}(r, \theta, \phi) \sqrt{2} \sqrt{M} \epsilon}{2r^{3/2}} \quad (10)$$

$$+ \frac{H_{T11}(r, \theta, \phi) \sqrt{2} \sqrt{M} \epsilon}{r^{7/2}} - \frac{3\sqrt{2} \sqrt{M}}{2r^{3/2}} + \frac{\sqrt{2} \sqrt{M} H_{T22}(r, \theta, \phi) \epsilon}{r^{7/2} \sin(\theta)^2}$$

Start the Hamilton constraint equation

```
> grdef(`Cons:=KTr^2-K2`);
grcalc(Cons);
```

```

    grmap(Cons, Linearize, 'x');
    grmap(Cons, FixSin, 'x');
Created definition for Cons
Calculated Cons for lpschwNewFixed (0.000000 sec.)
      CPU Time =0.
Procedure name:LinearizeApplying routine Linearize to Cons
Procedure name:FixSinApplying routine FixSin to Cons

```

Insert the harmonics for the metric and the Extrinsic curvature

```

> grmap(Cons, hExpand, 'x');
    grmap(Cons, kExpand, 'x');
    grdisplay(_);
Procedure name:hExpandApplying routine hExpand to Cons
Procedure name:kExpandApplying routine kExpand to Cons
      For the lpschwNewFixed spacetime:

```

$$\begin{aligned}
 & \text{Cons} = - \frac{2\sqrt{2} \epsilon \sqrt{M} k_{\text{trace}}(r) Y(\theta, \phi)}{r^{7/2}} - \frac{4\sqrt{2} k_{rr}(r) Y(\theta, \phi) \epsilon \sqrt{M}}{r^{3/2}} \\
 & + \frac{4 M h_{rr}(r) Y(\theta, \phi) \epsilon}{r^3} - \frac{4 M \epsilon h_{\text{trace}}(r) Y(\theta, \phi)}{r^5}
 \end{aligned} \tag{11}$$

Next we need the Ricci Scalar

```

> grcalc(Ricciscalar); grdisplay(_);
Calculated g(dn,dn,pdn) for lpschwNewFixed (0.000000 sec.)
Calculated Chr(dn,dn,dn) for lpschwNewFixed (0.000000 sec.)
Calculated Chr(dn,dn,up) for lpschwNewFixed (0.000000 sec.)
Calculated R(dn,dn) for lpschwNewFixed (0.016000 sec.)
Calculated Ricciscalar for lpschwNewFixed (0.031000 sec.)
      CPU Time =0.047

```

For the lpschwNewFixed spacetime:

Ricci scalar

$$R = 149090 \text{ words. Exceeds } grOptionDisplayLimit \tag{12}$$

Run some macros on R before finishing the constraint equation

```

> grmap(Ricciscalar, Linearize, 'x');
    grmap(Ricciscalar, FixSin, 'x');
    grmap(Ricciscalar, hExpand, 'x');
    grmap(Ricciscalar, ApplyId, 'x');
    grmap(Ricciscalar, RemPhi, 'x');
    grmap(Ricciscalar, RemPhi, 'x');
    grmap(Ricciscalar, FixSin, 'x');
    grdisplay(Ricciscalar);
Procedure name:LinearizeApplying routine Linearize to Ricciscalar
Procedure name:FixSinApplying routine FixSin to Ricciscalar
Procedure name:hExpandApplying routine hExpand to Ricciscalar
Procedure name:ApplyIdApplying routine ApplyId to Ricciscalar

```



Procedure name:RemPhiApplying routine RemPhi to Ricciscalar  
 Procedure name:RemPhiApplying routine RemPhi to Ricciscalar  
 Procedure name:FixSinApplying routine FixSin to Ricciscalar  
*For the lpschwNewFixed spacetime:*

*Ricci scalar*

$R = 15892$  words. Exceeds grOptionDisplayLimit

(13)

Now we can form final expression for the constraint equation for the Initial-Value Problem

```
> grdef(`Con:=Ricciscalar+Cons`);
grcalc(Con);
grmap(Con, hExpand, 'x');
grmap(Con, ApplyId, 'x');
gralter(Con, simplify);
gralter(Con, expand);

collect(collect(collect(grcomponent(Con), h__trace(r)), h__Teven
(r)), h__rr(r)):
```

Created definition for Con

Calculated Con for lpschwNewFixed (0.000000 sec.)

*CPU Time =0.*

Procedure name:hExpandApplying routine hExpand to Con

Procedure name:ApplyIdApplying routine ApplyId to Con

Component simplification of a GRTensorIII object:

Applying routine simplify to object Con

*CPU Time =0.046*

Component simplification of a GRTensorIII object:

Applying routine expand to object Con

*CPU Time =0.*

(14)

```
> grdef(`Con0:=Con/Y(theta,phi)/epsilon*r/2`);
grcalc(Con0);
gralter(Con, simplify);
gralter(Con0, expand);
H0X:=collect(collect(collect(grcomponent(Con0), h__trace(r)),
h__Teven(r)), h__rr(r));
```

Created definition for Con0

Calculated Con0 for lpschwNewFixed (0.000000 sec.)

*CPU Time =0.*

Component simplification of a GRTensorIII object:

Applying routine simplify to object Con

*CPU Time =0.016*

Component simplification of a GRTensorIII object:

Applying routine expand to object Con0

*CPU Time =0.*

$$H0X := \left( \frac{2M}{r^2} - \frac{\lambda}{2r} + \frac{1}{r} \right) h_{rr}(r) + \left( \frac{\lambda}{2r^3} + \frac{\lambda^2}{4r^3} \right) h_{Teven}(r) + \left( -\frac{2M}{r^4} - \frac{\lambda}{2r^3} \right)$$

(15)

$$\begin{aligned}
& -\frac{1}{r^3} \Big) h_{trace}(r) - \frac{2\sqrt{2} k_{rr}(r) \sqrt{M}}{\sqrt{r}} - \frac{\sqrt{2} \sqrt{M} k_{trace}(r)}{r^{5/2}} + \frac{\left( \frac{d}{dr} h_{Even}(r) \right) \lambda}{r} \\
& + \frac{h_{Even}(r) \lambda}{r^2} + \frac{\frac{d}{dr} h_{trace}(r)}{r^2} - \frac{\frac{d^2}{dr^2} h_{trace}(r)}{r} + \frac{d}{dr} h_{rr}(r)
\end{aligned}$$

Now we want to formulate the momentum constraint

```
> grdef(`JJ{^a ^b}:=KK{^a ^b}-g{^a ^b}*KTr`);
grcalc(JJ(up, up));
```

```
grdef(`J{b}:=JJ{^a b ;a}`);
```

```
grcalc(J(dn));
```

```
grmap(J(dn), Linearize, 'x');
grmap(J(dn), FixSin, 'x');
grmap(J(dn), hExpand, 'x');
grmap(J(dn), kExpand, 'x');
grmap(J(dn), RemPhi, 'x');
grmap(J(dn), ApplyId, 'x');
grmap(J(dn), CollectY, 'x');
```

```
grcalc(J(dn));
```

```
gralter(J(dn), simplify);
gralter(J(dn), expand);
```

Created definition for JJ(up,up)

Calculated JJ(up,up) for lpschwNewFixed (0.016000 sec.)

CPU Time =0.016

Created definition for JJ(up,dn)

Created a definition for JJ(up,dn,cdn)

Created definition for J(dn)

Calculated JJ(up,dn) for lpschwNewFixed (0.000000 sec.)

Calculated JJ(up,dn,cdn) for lpschwNewFixed (0.672000 sec.)

Calculated J(dn) for lpschwNewFixed (0.172000 sec.)

CPU Time =0.844

Procedure name:LinearizeApplying routine Linearize to J(dn)

Procedure name:FixSinApplying routine FixSin to J(dn)

Procedure name:hExpandApplying routine hExpand to J(dn)

Procedure name:kExpandApplying routine kExpand to J(dn)

Procedure name:RemPhiApplying routine RemPhi to J(dn)

Procedure name:ApplyIdApplying routine ApplyId to J(dn)

Procedure name:CollectYApplying routine CollectY to J(dn)

CPU Time =0.

Component simplification of a GRTensorIII object:

Applying routine simplify to object J(dn)

CPU Time =0.110

Component simplification of a GRTensorIII object:

Applying routine expand to object J(dn)

*CPU Time = 0.*

(16)

Below are the consistency checks. The r component is fine but the theta and phi are not matching Ivan's. trying many different things.

r component

```
> H1:=-epsilon*Y(theta, phi)*(-4*k__rr(r)*r^(13/2)-2*k__Veven(r)*
lambda*r^(11/2)+4*(diff(k__trace(r), r))*r^(11/2)-4*k__trace(r)*
r^(9/2)+sqrt(M)*sqrt(2)*h__Veven(r)*lambda*r^4+2*sqrt(M)*sqrt(2)*
h__rr(r)*r^5+sqrt(M)*sqrt(2)*(diff(h__trace(r), r))*r^4-4*sqrt(M)
*sqrt(2)*h__trace(r)*r^3)/(2*r^(15/2));
```

```
Test0:=simplify(H1-grcomponent(J(dn), [r]));
```

*Test0 := 0*

(17)

Theta check

```
> H2:=-( ((- (1/2)*lambda-1)*k__Teven(r)+k__trace(r))*(r^(3/2))+
(k__rr(r)-(diff(k__Veven(r), r)))*(r^(7/2))-2*k__Veven(r)*(r^
(5/2)))+(1/4)*(sqrt(M)*(-4*(diff(h__Veven(r), r))*(r^2)+(-2*
lambda-4)*h__Teven(r)+h__rr(r)*(r^2)-2*h__Veven(r)*r+4*h__trace
(r))*sqrt(2))) *epsilon*(diff(Y(theta, phi), theta))*(1/r^(7/2))-
I*epsilon*Y(theta, phi)*m*(2*(diff(k__Vodd(r), r))*(r^(11/2))+4*
k__Vodd(r)*(r^(9/2))+k__Todd(r)*lambda*(r^(7/2))+2*k__Todd(r)*(r^
(7/2))+2*sqrt(M)*sqrt(2)*(diff(h__Vodd(r), r))*(r^4)+sqrt(M)*sqrt
(2)*h__Todd(r)*lambda*(r^2)+sqrt(M)*sqrt(2)*h__Vodd(r)*(r^3)+2*
sqrt(M)*sqrt(2)*h__Todd(r)*(r^2))*(1/(2*r^(11/2)*sin(theta)))):
```

```
Test2:=simplify(expand(H2-grcomponent(J(dn), [theta])));
```

*Test2 := 0*

(18)

Phi check

```
> H3:=2*sin(theta)*epsilon*((1/4)*(k__Todd(r)*(lambda+2)*(r^(3/2))
+k__Vodd(r)*(r^(5/2)))+(1/2)*((diff(k__Vodd(r), r))*(r^(7/2)))+(
1/4)*(sqrt(M)*(2*(diff(h__Vodd(r), r))*(r^2)+(lambda+2)*h__Todd
(r)+h__Vodd(r)*r)*sqrt(2)))*(diff(Y(theta, phi), theta))*(1/r^
(7/2))+I*epsilon*m*Y(theta, phi)*(4*(diff(k__Veven(r), r))*(r^
(11/2))-4*k__rr(r)*(r^(11/2))+8*k__Veven(r)*(r^(9/2))+2*k__Teven
(r)*lambda*(r^(7/2))+4*k__Teven(r)*(r^(7/2))-4*k__trace(r)*(r^
(7/2))+4*sqrt(M)*sqrt(2)*(diff(h__Veven(r), r))*(r^4)-sqrt(2)*
sqrt(M)*h__rr(r)*(r^4)+2*sqrt(M)*sqrt(2)*h__Veven(r)*(r^3)+2*sqrt
(M)*sqrt(2)*h__Teven(r)*lambda*(r^2)+4*sqrt(M)*sqrt(2)*h__Teven
(r)*(r^2)-4*sqrt(M)*sqrt(2)*h__trace(r)*(r^2))*(1/(4*r^(11/2)))):
```

```
Test3:=simplify(H3-grcomponent(J(dn), [phi]));
```

*Test3 := 0*

(19)

Start with the decomposition of the vectors from the IDV2 commands

```
> H2__even, H2__odd:=IdV2(grcomponent(J(dn), [theta])):
H3__even, H3__odd:=IdV3(grcomponent(J(dn), [phi])):
```

Check that the components are the same for each mode type

```
> simplify(FixSin(H2__even-H3__even));
simplify(FixSin(H2__odd-H3__odd));
```

0  
0

(20)

Now we can setup the constraint equations in vacuum: Following Baumgarte and Shapiro pg 124

Hamiltonian Constraint

```
> H0__vac:=collect(collect(collect(expand(subs(H0X)), h__rr
(r)), h__trace(r)), h__Teven(r)), m);
```

$$H0_{vac} := \left( \frac{2M}{r^2} - \frac{\lambda}{2r} + \frac{1}{r} \right) h_{rr}(r) + \left( \frac{\lambda}{2r^3} + \frac{\lambda^2}{4r^3} \right) h_{Teven}(r) + \left( -\frac{2M}{r^4} - \frac{\lambda}{2r^3} - \frac{1}{r^3} \right) h_{trace}(r) - \frac{2\sqrt{2} k_{rr}(r) \sqrt{M}}{\sqrt{r}} - \frac{\sqrt{2} \sqrt{M} k_{trace}(r)}{r^{5/2}} + \frac{\left( \frac{d}{dr} h_{Teven}(r) \right) \lambda}{r} + \frac{h_{Teven}(r) \lambda}{r^2} + \frac{\frac{d}{dr} h_{trace}(r)}{r^2} - \frac{\frac{d^2}{dr^2} h_{trace}(r)}{r} + \frac{d}{dr} h_{rr}(r) \quad (21)$$

```
> H0__vacX:=collect(collect(collect(expand(subs(h__rr(r)=h__rr(r)
/r, h__Teven(r)=h__Teven(r)/r, r*H0__vac)), h__rr(r)), h__trace(r)),
h__Teven(r));;
```

$$H0_{vacX} := \left( \frac{\lambda}{2r^2} + \frac{\lambda^2}{4r^2} \right) h_{Teven}(r) + \left( -\frac{2M}{r^3} - \frac{\lambda}{2r^2} - \frac{1}{r^2} \right) h_{trace}(r) + \left( \frac{2M}{r^2} - \frac{\lambda}{2r} \right) h_{rr}(r) - 2\sqrt{r} \sqrt{2} k_{rr}(r) \sqrt{M} - \frac{\sqrt{2} \sqrt{M} k_{trace}(r)}{r^{3/2}} + \frac{\left( \frac{d}{dr} h_{Teven}(r) \right) \lambda}{r} + \frac{\frac{d}{dr} h_{trace}(r)}{r} - \left( \frac{d^2}{dr^2} h_{trace}(r) \right) + \frac{d}{dr} h_{rr}(r) \quad (22)$$

Momentum Constraint

```
> H1__vac:=collect(expand(-subs(subs(k__trace(r)=r*k__trace(r),
expand(r/2*grcomponent(J(dn), [r])/epsilon/Y(theta,phi))))), m);
```

$$H1_{vac} := -\frac{k_{Teven}(r) \lambda}{2r} + \frac{d}{dr} k_{trace}(r) - k_{rr}(r) + \frac{\sqrt{2} \sqrt{M} h_{rr}(r)}{2r^{3/2}} + \frac{\sqrt{2} \sqrt{M} h_{Teven}(r) \lambda}{4r^{5/2}} + \frac{\sqrt{2} \sqrt{M} \left( \frac{d}{dr} h_{trace}(r) \right)}{4r^{5/2}} - \frac{\sqrt{2} \sqrt{M} h_{trace}(r)}{r^{7/2}} \quad (23)$$

```
> expand(subs(h__trace(r)=r^4*h__trace(r), expand((sqrt(2)*h__Teven
(r)*lambda*(1/(4*r^(5/2)))+sqrt(2)*h__rr(r)*(1/(2*r^(3/2))))+sqrt
```

```
(2)*(diff(h__trace(r), r))*(1/(4*r^(5/2)))-sqrt(2)*h__trace(r)*
(1/r^(7/2)))/sqrt(2)*r^(3/2)*2));
```

$$\frac{h_{Veven}(r) \lambda}{2 r} + h_{rr}(r) + \frac{r^3 \left( \frac{d}{dr} h_{trace}(r) \right)}{2} \quad (24)$$

```
> Heven__vac:=collect(collect(collect(collect(expand(subs(k__Veven
(r)=k__Veven(r)/r^2,h__Veven(r)=h__Veven(r)/sqrt(r),expand(r^2*
H2__even))),k__Teven(r)),k__trace(r)),h__Teven(r)),M);
```

$$Heven_{vac} := \left( \left( \frac{\sqrt{2} \lambda}{2 r^{3/2}} + \frac{\sqrt{2}}{r^{3/2}} \right) h_{Teven}(r) + \sqrt{2} \left( \frac{d}{dr} h_{Veven}(r) \right) - \frac{\sqrt{r} \sqrt{2} h_{rr}(r)}{4} \right. \\ \left. - \frac{\sqrt{2} h_{trace}(r)}{r^{3/2}} \right) \sqrt{M} + \left( \frac{\lambda}{2} + 1 \right) k_{Teven}(r) - k_{trace}(r) - k_{rr}(r) r^2 + \frac{d}{dr} k_{Veven}(r) \quad (25)$$

```
> collect(expand(op(1,Heven__vac)/sqrt(2*M)*sqrt(r)),h__Teven);
```

$$\left( \frac{\lambda}{2 r} + \frac{1}{r} \right) h_{Teven}(r) + \sqrt{r} \left( \frac{d}{dr} h_{Veven}(r) \right) - \frac{r h_{rr}(r)}{4} - \frac{h_{trace}(r)}{r} \quad (26)$$

```
> Hodd__vac:=collect(collect(collect(expand(subs(h__Vodd(r)=h__Vodd
(r)/sqrt(r),k__Vodd(r)=k__Vodd(r)/r^2,r^2*H2__odd))),k__Todd(r)),
h__Todd(r)), M);
```

$$Hodd_{vac} := \left( \left( \frac{\sqrt{2} \lambda}{2 r^{3/2}} + \frac{\sqrt{2}}{r^{3/2}} \right) h_{Todd}(r) + \sqrt{2} \left( \frac{d}{dr} h_{Vodd}(r) \right) \right) \sqrt{M} + \left( \frac{\lambda}{2} + 1 \right) k_{Todd}(r) \\ + \frac{d}{dr} k_{Vodd}(r) \quad (27)$$

## Initial Data Formulation

# Time Evolution

## Metric

The unperturbed values are from PG and the lapse and shift are left free to insert the gauge at a later time

```
> NX:=1+epsilon*n__L(r)*Y(theta, phi);
```

$$NX := 1 + \epsilon n_L(r) Y(\theta, \phi) \quad (28)$$

```
> grdef(`VX{a}:=[sqrt(2*M/r)+epsilon*v__scalar(r)*Y(theta,phi),
epsilon*v__even(r)*Y__2+epsilon*v__odd(r)*X__2,epsilon*v__even(r)
*Y__3+epsilon*v__odd(r)*X__3]`);
grcalc(VX(dn));
```

```
grmap(VX(dn), Linearize, 'x');
grcalc(VX(up));
grdisplay(VX(dn));
```

Components assigned for metric: lpschwNewFixed  
Created definition for VX(dn)

CPU Time =0.

Procedure name:LinearizeApplying routine Linearize to VX(dn)

Created definition for VX(up)  
 Calculated VX(up) for lpschwNewFixed (0.000000 sec.)  
 CPU Time =0.

*For the lpschwNewFixed spacetime:*

$$\begin{aligned}
 & VX(dn) \\
 & VX(dn) \\
 & VX_r = \frac{\epsilon v_{scalar}(r) Y(\theta, \phi) r + \sqrt{2} \sqrt{M} \sqrt{r}}{r} \\
 & VX_\theta = - \frac{\epsilon \left( I v_{odd}(r) m Y(\theta, \phi) - v_{even}(r) \left( \frac{\partial}{\partial \theta} Y(\theta, \phi) \right) \sin(\theta) \right)}{\sin(\theta)} \\
 & VX_\phi = \left( I v_{even}(r) m Y(\theta, \phi) + v_{odd}(r) \sin(\theta) \left( \frac{\partial}{\partial \theta} Y(\theta, \phi) \right) \right) \epsilon
 \end{aligned} \tag{29}$$

```
> grdef(`DV{a b}:=VX{b ;a}`);
grcalc(DV(dn, dn));
grdisplay(DV(dn, dn));
```

Created a definition for VX(dn,cdn)  
 Created definition for DV(dn,dn)  
 Calculated VX(dn,cdn) for lpschwNewFixed (0.000000 sec.)  
 Calculated DV(dn,dn) for lpschwNewFixed (0.000000 sec.)  
 CPU Time =0.

*For the lpschwNewFixed spacetime:*

$$\begin{aligned}
 & DV(dn,dn) \\
 & DV(dn, dn) \\
 & DV_{rr} = 5642 \text{ words. Exceeds grOptionDisplayLimit} \\
 & DV_{\theta r} = 6129 \text{ words. Exceeds grOptionDisplayLimit} \\
 & DV_{\phi r} = 6143 \text{ words. Exceeds grOptionDisplayLimit} \\
 & DV_{r\theta} = 6220 \text{ words. Exceeds grOptionDisplayLimit} \\
 & DV_{\theta\theta} = 6349 \text{ words. Exceeds grOptionDisplayLimit} \\
 & DV_{\phi\theta} = 6330 \text{ words. Exceeds grOptionDisplayLimit} \\
 & DV_{r\phi} = 6236 \text{ words. Exceeds grOptionDisplayLimit} \\
 & DV_{\theta\phi} = 6244 \text{ words. Exceeds grOptionDisplayLimit} \\
 & DV_{\phi\phi} = 7136 \text{ words. Exceeds grOptionDisplayLimit}
 \end{aligned} \tag{30}$$

Coordinate time derivative of the metric

```
> grdef(`hdot{a b}:=2*NX*KK{a b}+DV{a b}+DV{b a}`);
grcalc(hdot(dn, dn));

grmap(hdot(dn, dn), Linearize, 'x');
grmap(hdot(dn, dn), FixSin, 'x');
grmap(hdot(dn, dn), hExpand, 'x');
grmap(hdot(dn, dn), kExpand, 'x');
```

```

grmap(hdot(dn, dn), RemPhi, 'x');
grmap(hdot(dn, dn), ApplyId, 'x');
Created definition for hdot(dn, dn)
Calculated hdot(dn, dn) for lpschwNewFixed (0.000000 sec.)
CPU Time = 0.
Procedure name:LinearizeApplying routine Linearize to hdot(dn,
dn)
Procedure name:FixSinApplying routine FixSin to hdot(dn, dn)
Procedure name:hExpandApplying routine hExpand to hdot(dn, dn)
Procedure name:kExpandApplying routine kExpand to hdot(dn, dn)
Procedure name:RemPhiApplying routine RemPhi to hdot(dn, dn)
Procedure name:ApplyIdApplying routine ApplyId to hdot(dn, dn)

```

Now we need to treat each of the components. First the rr component

```

> hdot__C11:=collect(expand(grcomponent(hdot(dn, dn), [1, 1])
/epsilon/Y(theta, phi)), sqrt(M)):
hdot__C11:=factor(op(1, hdot__C11))+op(2, hdot__C11)+op(3,
hdot__C11);

```

$$hdot_{C11} := \frac{\sqrt{2} \left( -r \left( \frac{d}{dr} h_{rr}(r) \right) + n_L(r) \right) \sqrt{M}}{r^{3/2}} + 2 \left( \frac{d}{dr} v_{scalar}(r) \right) + 2 k_{rr}(r) \quad (31)$$

Check the hdot\_\_C11 matches Ivan's

```

> hdot__C11Ivan:=sqrt(2)*(-r*(diff(h__rr(r), r))+n__L(r))*sqrt(M)
/r^(3/2)+2*(diff(v__scalar(r), r))+2*k__rr(r);
simplify(hdot__C11-hdot__C11Ivan);

```

$$hdot_{C11Ivan} := \frac{\sqrt{2} \left( -r \left( \frac{d}{dr} h_{rr}(r) \right) + n_L(r) \right) \sqrt{M}}{r^{3/2}} + 2 \left( \frac{d}{dr} v_{scalar}(r) \right) + 2 k_{rr}(r) \quad (32)$$

0

Next we move to the vector components and check that they are the same and then look at the components

```

> IdV2(grcomponent(hdot(dn, dn), [1, 2]))-IdV3(grcomponent(hdot(dn,
dn), [1, 3]));
0

```

(33)

Seperate into the even and odd components

```

> hdot__CEven, hdot__CVodd:=IdV2(grcomponent(hdot(dn, dn), [1, 2])
):
hdot__CEven;
hdot__CVodd;

```

$$-\frac{\sqrt{2} h_{rr}(r) \sqrt{M}}{\sqrt{r}} + \frac{2 \sqrt{2} \sqrt{M} h_{even}(r)}{r^{3/2}} + \frac{d}{dr} v_{even}(r) + v_{scalar}(r) - \frac{2 v_{even}(r)}{r}$$

$$+ 2 k_{\text{Even}}(r)$$

$$2 k_{\text{Odd}}(r) + \frac{d}{dr} v_{\text{odd}}(r) + \frac{2 \sqrt{2} h_{\text{Odd}}(r) \sqrt{M}}{r^3} - \frac{2 v_{\text{odd}}(r)}{r} \quad (34)$$

Lastly, we attend to the tensor components

The [2, 3] component matches Ivan's

```
> hdot__Teven, hdot__Todd:=IdT23(grcomponent(hdot(dn, dn), [2, 3]))
:
```

```
hdot__Teven;
hdot__Todd;
```

$$\begin{aligned} & -\frac{2 \sqrt{2} h_{\text{Even}}(r) \sqrt{M}}{\sqrt{r}} + \frac{\sqrt{2} \sqrt{M} \left( \frac{d}{dr} h_{\text{Teven}}(r) \right)}{\sqrt{r}} + 2 k_{\text{Teven}}(r) + 2 v_{\text{even}}(r) \\ & -\frac{2 \sqrt{2} h_{\text{Odd}}(r) \sqrt{M}}{\sqrt{r}} + \frac{\sqrt{2} \sqrt{M} \left( \frac{d}{dr} h_{\text{Todd}}(r) \right)}{\sqrt{r}} + 2 k_{\text{Todd}}(r) + 2 v_{\text{odd}}(r) \end{aligned} \quad (35)$$

```
> hdot__TOmega:=collect(IdT22(grcomponent(hdot(dn, dn), [2, 2]),
hdot__Teven, hdot__Todd), sqrt(M));
hdot__T33:=collect(IdT33(grcomponent(hdot(dn, dn), [3, 3]),
hdot__Teven, hdot__Todd), sqrt(M));
```

$$\begin{aligned} hdot_{\text{TOmega}} &:= \left( \frac{\sqrt{2} \left( \frac{d}{dr} h_{\text{trace}}(r) \right)}{\sqrt{r}} - \frac{\sqrt{2} h_{\text{Even}}(r) \lambda}{\sqrt{r}} - 2 \sqrt{r} \sqrt{2} h_{rr}(r) \right. \\ & \quad \left. - 2 \sqrt{r} n_L(r) \sqrt{2} \right) \sqrt{M} + v_{\text{even}}(r) \lambda + 2 v_{\text{scalar}}(r) r + 2 k_{\text{trace}}(r) \\ hdot_{\text{T33}} &:= \left( \frac{\sqrt{2} \left( \frac{d}{dr} h_{\text{trace}}(r) \right)}{\sqrt{r}} - \frac{\sqrt{2} h_{\text{Even}}(r) \lambda}{\sqrt{r}} - 2 \sqrt{r} \sqrt{2} h_{rr}(r) \right. \\ & \quad \left. - 2 \sqrt{r} n_L(r) \sqrt{2} \right) \sqrt{M} + v_{\text{even}}(r) \lambda + 2 v_{\text{scalar}}(r) r + 2 k_{\text{trace}}(r) \end{aligned} \quad (36)$$

There to be an issue here where the check should be against the [3, 3] component but instead the code is the [2, 2] component against itself.

\*\*\*\*\* As can be seen above the [2, 2] and the [3, 3] components are indeed the same as expected.

```
> hdot__TOmega__Check:=collect(expand(simplify(grcomponent(hdot(dn,
dn), [2, 2])*sin(theta)^2+expand(epsilon*hdot__Todd*X__33+
epsilon*hdot__Teven*Y__33))/sin(theta)^2/epsilon/Y(theta, phi)),
```



**sqrt(M) ;**

$$h_{dot_{T\Omega Check}} := \left( \frac{\sqrt{2} \left( \frac{d}{dr} h_{trace}(r) \right)}{\sqrt{r}} - \frac{\sqrt{2} h_{V_{even}}(r) \lambda}{\sqrt{r}} - 2 \sqrt{r} \sqrt{2} h_{rr}(r) \right. \\ \left. - 2 \sqrt{r} n_L(r) \sqrt{2} \right) \sqrt{M} + v_{even}(r) \lambda + 2 v_{scalar}(r) r + 2 k_{trace}(r) \quad (37)$$

**> expand(hdot\_\_TOmega-hdot\_\_TOmega\_\_Check) ;**  
0 (38)

**> expand(op(1, hdot\_\_TOmega)/sqrt(M)/sqrt(2)/sqrt(r)) ;**

$$\frac{\frac{d}{dr} h_{trace}(r)}{r} - \frac{h_{V_{even}}(r) \lambda}{r} - 2 h_{rr}(r) - 2 n_L(r) \quad (39)$$

## K\_ij Evolution Equations

Start with the Ricci term - from Ivan's notes Ricci is trivial and N0 is 1 so we include the lapse N in case of changes later on

```
> grcalc(R(dn, dn));
grmap(R(dn, dn), Linearize, 'x');
grmap(R(dn, dn), FixSin, 'x');
grmap(R(dn, dn), hExpand, 'x');
grmap(R(dn, dn), kExpand, 'x');
grmap(R(dn, dn), RemPhi, 'x');
grmap(R(dn, dn), ApplyId, 'x');
grmap(R(dn, dn), ApplyId, 'x');
grmap(R(dn, dn), RemPhi, 'x');
gralter(R(dn, dn), simplify);
grmap(R(dn, dn), RemPhi, 'x');
gralter(R(dn, dn), simplify);
```

*CPU Time = 0.*

Procedure name:LinearizeApplying routine Linearize to R(dn,dn)

Procedure name:FixSinApplying routine FixSin to R(dn,dn)

Procedure name:hExpandApplying routine hExpand to R(dn,dn)

Procedure name:kExpandApplying routine kExpand to R(dn,dn)

Procedure name:RemPhiApplying routine RemPhi to R(dn,dn)

Procedure name:ApplyIdApplying routine ApplyId to R(dn,dn)

Procedure name:ApplyIdApplying routine ApplyId to R(dn,dn)

Procedure name:RemPhiApplying routine RemPhi to R(dn,dn)

Component simplification of a GRTensorIII object:

Applying routine simplify to object R(dn,dn)

*CPU Time = 0.282*

Procedure name:RemPhiApplying routine RemPhi to R(dn,dn)

Component simplification of a GRTensorIII object:

Applying routine simplify to object R(dn,dn)

*CPU Time = 0.062*

(40)

The scalar component is first

```
> NRc__rr:=Linearize(NX*grcomponent(R(dn, dn), [1, 1]));
```

$$NRc_{rr} := \frac{1}{r^4} \left( Y(\theta, \phi) \left( \left( \frac{d}{dr} h_{rr}(r) \right) r^3 + \left( \frac{d}{dr} h_{Veven}(r) \right) \lambda r^2 - \frac{h_{rr}(r) \lambda r^2}{2} - \left( \frac{d^2}{dr^2} h_{trace}(r) \right) r^2 + 2 \left( \frac{d}{dr} h_{trace}(r) \right) r - 2 h_{trace}(r) \right) \epsilon \right) \quad (41)$$

Next is the vector components

```
> NRc__12:=Linearize(NX*grcomponent(R(dn, dn), [1, 2]));
NRc__13:=Linearize(NX*grcomponent(R(dn, dn), [1, 3]));
```

Check against Ivan's

```
> NRc__12Ivan:=(epsilon/(4*sin(theta)*r^3))*(-(2*(-(1/2)*r*
(lambda+2)*(diff(h__Teven(r), r))+(diff(h__trace(r), r))*r+
(lambda+2)*h__Teven(r)-h__rr(r)*r^2+2*h__Veven(r)*r-2*h__trace(r)
))*sin(theta)*(diff(Y(theta, phi), theta))-I*((diff(h__Todd(r),
r))*r-2*h__Vodd(r)*r-2*h__Todd(r))*(lambda+2)*Y(theta, phi)*m):

simplify(expand(NRc__12-NRc__12Ivan));
0
```

(42)

```
> NRc__13Ivan:=(epsilon/(4*r^3))*(sin(theta)*(lambda+2)*((diff
(h__Todd(r), r))*r-2*h__Vodd(r)*r-2*h__Todd(r))*(diff(Y(theta,
phi), theta))-(2*I)*(-(1/2)*r*(lambda+2)*(diff(h__Teven(r), r))+
(diff(h__trace(r), r))*r+(lambda+2)*h__Teven(r)-h__rr(r)*r^2+2*
h__Veven(r)*r-2*h__trace(r))*Y(theta, phi)*m):

simplify(NRc__13-NRc__13Ivan);
0
```

(43)

Seperate the vector components into odd/even components

```
> NRc__V12even, NRc__V12odd:=IdV2(NRc__12):
NRc__V12even;
NRc__V12odd;
```

$$\begin{aligned} & \left( \frac{d}{dr} h_{Teven}(r) \right) \lambda \frac{1}{4r^2} + \frac{d}{dr} h_{Teven}(r) \frac{1}{2r^2} - \frac{d}{dr} h_{trace}(r) \frac{1}{2r^2} - \frac{h_{Teven}(r) \lambda}{2r^3} - \frac{h_{Teven}(r)}{r^3} + \frac{h_{rr}(r)}{2r} \\ & - \frac{h_{Veven}(r)}{r^2} + \frac{h_{trace}(r)}{r^3} \\ & - \frac{h_{Vodd}(r) \lambda}{2r^2} - \frac{h_{Vodd}(r)}{r^2} + \left( \frac{d}{dr} h_{Todd}(r) \right) \lambda \frac{1}{4r^2} + \frac{d}{dr} h_{Todd}(r) \frac{1}{2r^2} - \frac{h_{Todd}(r) \lambda}{2r^3} - \frac{h_{Todd}(r)}{r^3} \end{aligned} \quad (44)$$

```
> NRc__V13even, NRc__V13odd:=IdV3(NRc__13):
```

**NRC\_\_V13even;**  
**NRC\_\_V13odd;**

$$\begin{aligned} & \left( \frac{d}{dr} h_{Teven}(r) \right) \frac{\lambda}{4 r^2} + \frac{d}{dr} h_{Teven}(r) \frac{1}{2 r^2} - \frac{d}{dr} h_{trace}(r) \frac{1}{2 r^2} - \frac{h_{Teven}(r) \lambda}{2 r^3} - \frac{h_{Teven}(r)}{r^3} + \frac{h_{rr}(r)}{2 r} \\ & - \frac{h_{Veven}(r)}{r^2} + \frac{h_{trace}(r)}{r^3} \\ & - \frac{h_{Vodd}(r) \lambda}{2 r^2} - \frac{h_{Vodd}(r)}{r^2} + \left( \frac{d}{dr} h_{Todd}(r) \right) \frac{\lambda}{4 r^2} + \frac{d}{dr} h_{Todd}(r) \frac{1}{2 r^2} - \frac{h_{Todd}(r) \lambda}{2 r^3} - \frac{h_{Todd}(r)}{r^3} \end{aligned} \quad (45)$$

Finally the tensor components

Start oin the [2, 3] component

**> NRC\_\_23:=Linearize(NX\*grcomponent(R(dn, dn), [2, 3])):**  
**NRC\_\_T23even, NRC\_\_T23odd:=IdT23(NRC\_\_23):**  
**NRC\_\_T23even;**  
**NRC\_\_T23odd;**

$$\begin{aligned} & - \frac{h_{rr}(r)}{2} + \frac{d}{dr} h_{Veven}(r) - \frac{\left( \frac{d^2}{dr^2} h_{Teven}(r) \right)}{2} + \frac{d}{dr} h_{Teven}(r) \frac{1}{r} - \frac{h_{Teven}(r)}{r^2} \\ & \frac{d}{dr} h_{Vodd}(r) - \frac{\left( \frac{d^2}{dr^2} h_{Todd}(r) \right)}{2} + \frac{d}{dr} h_{Todd}(r) \frac{1}{r} - \frac{h_{Todd}(r)}{r^2} \end{aligned} \quad (46)$$

Check against Ivan

**> NRC\_\_T23Ivan:=- (1/2)\*(diff(h\_\_Teven(r), r, r))+diff(h\_\_Veven(r), r)-(1/2)\*h\_\_rr(r)+(diff(h\_\_Teven(r), r))/r-h\_\_Teven(r)/r^2, diff(h\_\_Vodd(r), r)-(1/2)\*(diff(h\_\_Todd(r), r, r))+diff(h\_\_Todd(r), r))/r-h\_\_Todd(r)/r^2:**

**simplify(NRC\_\_T23Ivan-IdT23(NRC\_\_23));**  
**0**

(47)

Next the [2, 2] component which will be checked against the [3, 3] component

**> NRC\_\_T22:=Linearize(NX\*grcomponent(R(dn, dn), [2, 2])):**  
**NRC\_\_TOmega:=simplify(ApplyId(FixSin(IdT22(NRC\_\_T22,**  
**NRC\_\_T23even, NRC\_\_T23odd))));**

$$\begin{aligned} NRC_{TOmega} := & \frac{1}{4 r^2} \left( -2 \left( \frac{d^2}{dr^2} h_{trace}(r) \right) r^2 + 2 \left( \frac{d}{dr} h_{Veven}(r) \right) \lambda r^2 + 2 \left( \frac{d}{dr} h_{rr}(r) \right) r^3 \right. \\ & \left. + (\lambda^2 + 2 \lambda) h_{Teven}(r) - r^2 (\lambda - 4) h_{rr}(r) + 4 \left( h_{Veven}(r) r - \frac{h_{trace}(r)}{2} \right) \lambda \right) \end{aligned} \quad (48)$$

Now the check against the [3, 3] component

```
> NRc__T33:=Linearize(NX*grcomponent(R(dn, dn), [3, 3])):
RcCheck:=simplify(ApplyId(expand(subs(NRc__T33-epsilon*
NRc__T23even*Y__33-epsilon*NRc__T23odd*X__33-epsilon*NRc__TOmega*
Omega__33)))));

RcCheck := 0
```

(49)

The Lapse term

We define the lapse as follows from Baumgarte and Shapiro (????double check definition) Seems like a standard definition from literature regarding the lapse function

```
> grdef(`DN:=1+epsilon*n__L(r)*Y(theta, phi)`);
grcalc(DN);
Created definition for DN
Calculated DN for lpschwNewFixed (0.000000 sec.)
CPU Time = 0.

> grdef(`DDN{i j}:=DN{;i ;j}`);
grcalc(DDN(dn, dn));
grmap(DDN(dn, dn), Linearize, 'x');
Created a definition for DN(cdn)
Created a definition for DN(cdn,cdn)
Created definition for DDN(dn,dn)
Calculated DN(cdn) for lpschwNewFixed (0.000000 sec.)
Calculated DN(cdn,cdn) for lpschwNewFixed (0.000000 sec.)
Calculated DDN(dn,dn) for lpschwNewFixed (0.000000 sec.)
CPU Time = 0.

Procedure name:LinearizeApplying routine Linearize to DDN(dn,dn)
```

(50)

With the second covariant derivatives of the Lapse we can now separate out all the components

```
> DDN__rr:=grcomponent(DDN(dn, dn), [1, 1]);
```

$$DDN_{rr} := \left( \frac{d^2}{dr^2} n_L(r) \right) Y(\theta, \phi) \in$$

(51)

```
> DDN__12:=grcomponent(DDN(dn, dn), [1, 2]):
DDN__13:=grcomponent(DDN(dn, dn), [1, 3]):

DDN__Veven, DDN__Vodd:=IdV2(DDN__12);
```

$$DDN_{Veven}, DDN_{Vodd} := \frac{d}{dr} n_L(r) - \frac{n_L(r)}{r}, 0$$

(52)

```
> DDN__22:=ApplyId(Linearize(grcomponent(DDN(dn, dn), [2,2]))):
DDN__23:=RemPhi(Linearize(grcomponent(DDN(dn, dn), [2,3]))):
DDN__33:=RemPhi(Linearize(grcomponent(DDN(dn, dn), [3,3]))):

DDN__Teven, DDN__Todd:=IdT23(DDN__23);

DDN__TOmega:=IdT22(DDN__22, DDN__Teven, DDN__Todd);
```

$$DDN_{Teven}, DDN_{Todd} := n_L(r), 0$$

$$DDN_{TOmega} := \frac{n_L(r) \lambda}{2} + \left( \frac{d}{dr} n_L(r) \right) r \quad (53)$$

The extrinsic curvature term is next for the evolution equation

```
> grdef(`NKK{i j}:=NX*(KTr*KK{i j}-2*g{^k ^l}*KK{i k}*KK{j l})`);
grcalc(NKK(dn, dn));

grmap(NKK(dn, dn), Linearize, 'x');
grmap(NKK(dn, dn), kExpand, 'x');
grmap(NKK(dn, dn), hExpand, 'x');
gralter(NKK(dn, dn), simplify);
grmap(NKK(dn, dn), FixSin, 'x');
grmap(NKK(dn, dn), Linearize, 'x');
```

Created definition for NKK(dn,dn)

Calculated NKK(dn,dn) for lpschwNewFixed (0.000000 sec.)

CPU Time =0.

Procedure name:LinearizeApplying routine Linearize to NKK(dn,dn)

Procedure name:kExpandApplying routine kExpand to NKK(dn,dn)

Procedure name:hExpandApplying routine hExpand to NKK(dn,dn)

Component simplification of a GRTensorIII object:

Applying routine simplify to object NKK(dn,dn)

CPU Time =0.203

Procedure name:FixSinApplying routine FixSin to NKK(dn,dn)

Procedure name:LinearizeApplying routine Linearize to NKK(dn,dn)

```
> NKK__rr:=expand(grcomponent(NKK(dn, dn), [1, 1]));
```

$$NKK_{rr} := \frac{\sqrt{2} \epsilon \sqrt{M} k_{trace}(r) Y(\theta, \phi)}{r^{7/2}} - \frac{3 \sqrt{2} k_{rr}(r) Y(\theta, \phi) \epsilon \sqrt{M}}{r^{3/2}} \quad (54)$$

$$+ \frac{M h_{rr}(r) Y(\theta, \phi) \epsilon}{2 r^3} + \frac{2 M \epsilon h_{trace}(r) Y(\theta, \phi)}{r^5} - \frac{5 M}{2 r^3} - \frac{5 M n_L(r) Y(\theta, \phi) \epsilon}{2 r^3}$$

```
> NKK__even, NKK__odd:=IdV2(grcomponent(NKK(dn, dn), [1, 2]));
IdV3(grcomponent(NKK(dn, dn), [1, 3]));
```

$$NKK_{even}, NKK_{odd} := -\frac{\sqrt{2} k_{Veven}(r) \sqrt{M}}{2 r^{3/2}} - \frac{2 M h_{Veven}(r)}{r^3}, -\frac{\sqrt{2} k_{Vodd}(r) \sqrt{M}}{2 r^{3/2}} \quad (55)$$

$$- \frac{2 M h_{Vodd}(r)}{r^3}$$

```
> NKK__23:=FixSin(expand(grcomponent(NKK(dn, dn), [2, 3])));
NKK__Teven, NKK__Todd:=IdT23(NKK__23);
```

$$NKK_{Teven}, NKK_{Todd} := \frac{5 \sqrt{2} k_{Teven}(r) \sqrt{M}}{2 r^{3/2}} + \frac{4 M h_{Teven}(r)}{r^3}, \frac{5 \sqrt{2} \sqrt{M} k_{Todd}(r)}{2 r^{3/2}} \quad (56)$$

$$+ \frac{4 M h_{Todd}(r)}{r^3}$$

```
> NKK__TOmega:=IdT22(grcomponent(NKK(dn, dn), [2, 2])+M/r,
NKK__Teven,NKK__Todd);
```

$$NKK_{TOmega} := -\sqrt{r} \sqrt{2} k_{rr}(r) \sqrt{M} + \frac{\sqrt{2} \sqrt{M} k_{trace}(r)}{2 r^{3/2}} - \frac{M n_L(r)}{r} + \frac{M h_{rr}(r)}{r} \quad (57)$$

Lastly we need to construct the shift terms for the extrinsic curvature equaiton before we can start into the time derivates. We already have the shift vector defined as VX[dn]

```
> grdef(`LVK{i j}:=VX{^k}*KK{i j ,k}+KK{i k}*VX{^k ,j}+KK{k j}*VX
{^k ,i}`);
grcalc(LVK(dn, dn));
```

```
grmap(LVK(dn, dn), Linearize, 'x');
grmap(LVK(dn, dn), hExpand, 'x');
grmap(LVK(dn, dn), kExpand, 'x');
grmap(LVK(dn, dn), ApplyId, 'x');
grmap(LVK(dn, dn), RemPhi, 'x');
gralter(LVK(dn, dn), simplify);
grmap(LVK(dn, dn), FixSin, 'x');
```

Created a definition for KK(dn,dn,pdn)

Created a definition for VX(up,pdn)

Created definition for LVK(dn,dn)

Calculated KK(dn,dn,pdn) for lpschwNewFixed (0.000000 sec.)

Calculated VX(up,pdn) for lpschwNewFixed (0.000000 sec.)

Calculated LVK(dn,dn) for lpschwNewFixed (0.031000 sec.)

CPU Time =0.031

Procedure name:LinearizeApplying routine Linearize to LVK(dn,dn)

Procedure name:hExpandApplying routine hExpand to LVK(dn,dn)

Procedure name:kExpandApplying routine kExpand to LVK(dn,dn)

Procedure name:ApplyIdApplying routine ApplyId to LVK(dn,dn)

Procedure name:RemPhiApplying routine RemPhi to LVK(dn,dn)

Component simplification of a GRTensorIII object:

Applying routine simplify to object LVK(dn,dn)

CPU Time =0.266

Procedure name:FixSinApplying routine FixSin to LVK(dn,dn)

```
> LVK__rr:=collect(grcomponent(LVK(dn, dn), [1, 1]), Y(theta, phi))
;
```

$$LVK_{rr} := \left( -\frac{2 M \left( \frac{d}{dr} h_{rr}(r) \right) \epsilon}{r^2} + \frac{\sqrt{2} \left( \frac{d}{dr} k_{rr}(r) \right) \epsilon \sqrt{M}}{\sqrt{r}} \right. \\ + \frac{\sqrt{2} \left( \frac{d}{dr} v_{scalar}(r) \right) \epsilon \sqrt{M}}{r^{3/2}} - \frac{3 \sqrt{2} v_{scalar}(r) \epsilon \sqrt{M}}{4 r^{5/2}} - \frac{\sqrt{2} k_{rr}(r) \epsilon \sqrt{M}}{r^{3/2}} \\ \left. + \frac{5 M h_{rr}(r) \epsilon}{2 r^3} \right) Y(\theta, \phi) - \frac{5 M}{2 r^3} \quad (58)$$

```
> LVK__Veven, LVK__Vodd:=IdV2(grcomponent(LVK(dn, dn), [1, 2])):
LVK__Veven3, LVK__Vodd3:=IdV3(grcomponent(LVK(dn, dn), [1, 3])):
```

```
simplify(LVK__Veven-LVK__Veven3);
simplify(LVK__Vodd-LVK__Vodd3);
0
0
```

(59)

```
> LVK__Teven, LVK__Todd:=IdT23(grcomponent(LVK(dn, dn), [2, 3]));
```

$$LVK_{Teven}, LVK_{Todd} := \frac{\sqrt{2} \left( \frac{d}{dr} k_{Teven}(r) \right) \sqrt{M}}{\sqrt{r}} - \frac{2\sqrt{2} \sqrt{M} v_{even}(r)}{r^{3/2}} + \frac{4 M h_{Veven}(r)}{r^2}, \quad (60)$$

$$\frac{\sqrt{2} \sqrt{M} \left( \frac{d}{dr} k_{Todd}(r) \right)}{\sqrt{r}} - \frac{2\sqrt{2} \sqrt{M} v_{odd}(r)}{r^{3/2}} + \frac{4 M h_{Vodd}(r)}{r^2}$$

```
> LVK__TOmega0:=coeff(grcomponent(LVK(dn, dn), [2, 2]), epsilon, 0);
```

```
LVK__TOmega:=(expand(IdT22(epsilon*coeff(grcomponent(LVK(dn, dn), [2, 2]), epsilon, 1), LVK__Teven, LVK__Todd)));
```

$$LVK_{TOmega0} := -\frac{M}{r}$$

$$LVK_{TOmega} := \frac{M h_{rr}(r)}{r} - \frac{\sqrt{2} \sqrt{M} v_{even}(r) \lambda}{r^{3/2}} - \frac{\sqrt{2} \sqrt{M} v_{scalar}(r)}{2\sqrt{r}} \quad (61)$$

$$+ \frac{\sqrt{2} \sqrt{M} \left( \frac{d}{dr} k_{trace}(r) \right)}{\sqrt{r}} + \frac{2 M h_{Veven}(r) \lambda}{r^2}$$

```
> LVK__TOmegaIvan:=M*h__rr(r)/r-sqrt(2)*v__even(r)*lambda*sqrt(M)/r^(3/2)-sqrt(2)*v__scalar(r)*sqrt(M)/(2*sqrt(r))+sqrt(2)*(diff(k__trace(r), r))*sqrt(M)/sqrt(r)+2*M*h__Veven(r)*lambda/r^2:
```

```
simplify(LVK__TOmega-LVK__TOmegaIvan);
0
```

(62)

The time rate of changes are the final thing to do with this evolution equation

```
> Kdot__rr:=collect(collect(collect(expand((DDN__rr-NRc__rr-NKK__rr+LVK__rr)/epsilon/Y(theta,phi)), h__trace(r)), h__rr(r)), M):
```

```
Kdot__rrX:=collect(collect(collect(expand(subs(h__trace(r)=r*h__trace(r), k__rr(r)=k__rr(r)/r^2, r*Kdot__rr)), h__trace(r)), h__rr(r)), M);
```

$$Kdot_{rrX} := \left( \frac{2 h_{rr}(r)}{r^2} - \frac{2 h_{trace}(r)}{r^3} + \frac{5 n_L(r)}{2 r^2} - \frac{2 \left( \frac{d}{dr} h_{rr}(r) \right)}{r} \right) M + \left( -\frac{\sqrt{2} k_{trace}(r)}{r^{5/2}} \right. \quad (63)$$

$$\left. + \frac{\sqrt{2} \left( \frac{d}{dr} k_{rr}(r) \right)}{r^{3/2}} + \frac{\sqrt{2} \left( \frac{d}{dr} v_{scalar}(r) \right)}{\sqrt{r}} - \frac{3\sqrt{2} v_{scalar}(r)}{4 r^{3/2}} \right) \sqrt{M} + \frac{h_{rr}(r) \lambda}{2 r}$$

$$+ r \left( \frac{d^2}{dr^2} n_L(r) \right) - \left( \frac{d}{dr} h_{rr}(r) \right) - \frac{\left( \frac{d}{dr} h_{Veven}(r) \right) \lambda}{r} + \frac{d^2}{dr^2} h_{trace}(r)$$

> Kdot\_Veven:=collect(collect(collect(expand(subs(DDN\_Veven-NRc\_V12even-NKK\_even+LVK\_Veven)), h\_Veven(r)), h\_Teven(r)), M);

$$Kdot_{Veven} := \left( -\frac{3 h_{Veven}(r)}{r^3} + \frac{2 \left( \frac{d}{dr} h_{Veven}(r) \right)}{r^2} - \frac{h_{rr}(r)}{r^2} \right) M + \left( \frac{\sqrt{2} \left( \frac{d}{dr} k_{Veven}(r) \right)}{\sqrt{r}} \right. \\ \left. + \frac{\sqrt{2} v_{scalar}(r)}{2 r^{3/2}} - \frac{\sqrt{2} \left( \frac{d}{dr} v_{even}(r) \right)}{r^{3/2}} + \frac{2 \sqrt{2} v_{even}(r)}{r^{5/2}} \right) \sqrt{M} + \left( \frac{\lambda}{2 r^3} \right. \\ \left. + \frac{1}{r^3} \right) h_{Teven}(r) + \frac{h_{Veven}(r)}{r^2} + \frac{d}{dr} n_L(r) - \frac{n_L(r)}{r} - \frac{\left( \frac{d}{dr} h_{Teven}(r) \right) \lambda}{4 r^2} \\ - \frac{\frac{d}{dr} h_{Teven}(r)}{2 r^2} + \frac{\frac{d}{dr} h_{trace}(r)}{2 r^2} - \frac{h_{rr}(r)}{2 r} - \frac{h_{trace}(r)}{r^3}$$

> Kdot\_Vodd:=collect(collect(collect(expand(DDN\_Vodd-NRc\_V12odd-NKK\_odd+LVK\_Vodd), h\_Vodd(r)), h\_Todd(r)), M);

Kdot\_VoddIvan:=(-3\*h\_Vodd(r)/r^3+2\*(diff(h\_Vodd(r), r))/r^2)\*M+((diff(k\_Vodd(r), r))\*sqrt(2)/sqrt(r)-sqrt(2)\*(diff(v\_odd(r), r))/r^(3/2)+2\*sqrt(2)\*v\_odd(r)/r^(5/2))\*sqrt(M)+(lambda/(2\*r^3)+1/r^3)\*h\_Todd(r)+(lambda/(2\*r^2)+1/r^2)\*h\_Vodd(r)-lambda\*(diff(h\_Todd(r), r))/(4\*r^2)-(diff(h\_Todd(r), r))/(2\*r^2):

simplify(Kdot\_Vodd-Kdot\_VoddIvan);

$$Kdot_{Vodd} := \left( -\frac{3 h_{Vodd}(r)}{r^3} + \frac{2 \left( \frac{d}{dr} h_{Vodd}(r) \right)}{r^2} \right) M + \left( \frac{\sqrt{2} \left( \frac{d}{dr} k_{Vodd}(r) \right)}{\sqrt{r}} \right. \\ \left. - \frac{\sqrt{2} \left( \frac{d}{dr} v_{odd}(r) \right)}{r^{3/2}} + \frac{2 \sqrt{2} v_{odd}(r)}{r^{5/2}} \right) \sqrt{M} + \left( \frac{\lambda}{2 r^3} + \frac{1}{r^3} \right) h_{Todd}(r) + \left( \frac{\lambda}{2 r^2} \right. \\ \left. + \frac{1}{r^2} \right) h_{Vodd}(r) - \frac{\left( \frac{d}{dr} h_{Todd}(r) \right) \lambda}{4 r^2} - \frac{\frac{d}{dr} h_{Todd}(r)}{2 r^2}$$

0

(65)

> Kdot\_Veven:=collect(collect(collect(collect(expand(subs(h\_Veven(r)=r^(3/2)\*h\_Veven(r), v\_even(r)=r^2\*h\_even(r), h\_Teven(r)=r^2\*h\_Teven(r), h\_trace(r)=r^2\*h\_trace(r), n\_L(r)=r\*n\_L(r), DDN\_Veven-NRc\_V12even-NKK\_even+LVK\_Veven)), h\_Veven(r)), h\_Teven(r)), diff(h\_Teven(r), r)), M):

Kdot\_VevenIvan:=(2\*(diff(h\_Veven(r), r))/sqrt(r)-h\_rr(r)/r^2)\*M+(sqrt(2)\*(diff(k\_Veven(r), r))/sqrt(r)+sqrt(2)\*v\_scalar(r)/(2\*r^(3/2))-sqrt(r)\*sqrt(2)\*(diff(v\_even(r), r)))\*sqrt(M)+(-



$(1/4)*\lambda-1/2)*(diff(h\_Teven(r), r))+r*(diff(n\_L(r), r))+$   
 $(1/2)*(diff(h\_trace(r), r))-h\_rr(r)/(2*r)+h\_Veven(r)/sqrt(r):$

**simplify(Kdot\_\_Veven-Kdot\_\_VevenIvan);**

$$-\sqrt{2} \sqrt{M} \sqrt{r} \left( \frac{d}{dr} h_{even}(r) - \left( \frac{d}{dr} v_{even}(r) \right) \right) \quad (66)$$

**> simplify(Kdot\_\_Veven-Kdot\_\_Vodd);**

$$\begin{aligned} & -\frac{1}{r^{7/2}} \left( \left( \left( -\frac{\lambda}{4} - \frac{1}{2} \right) \left( \frac{d}{dr} h_{Todd}(r) \right) + 2M \left( \frac{d}{dr} h_{Vodd}(r) \right) + \left( \frac{\lambda}{2} + 1 \right) h_{Vodd}(r) \right. \right. \\ & \quad \left. \left. + M h_{rr}(r) \right) r^{3/2} + \left( \left( \frac{\lambda}{4} + \frac{1}{2} \right) \left( \frac{d}{dr} h_{Teven}(r) \right) - \frac{\left( \frac{d}{dr} h_{trace}(r) \right)}{2} \right) r^{7/2} \right. \\ & \quad \left. + \frac{h_{rr}(r) r^{5/2}}{2} - \left( \frac{d}{dr} n_L(r) \right) r^{9/2} + 2\sqrt{2} r \left( \frac{r^3 \left( \frac{d}{dr} h_{even}(r) \right)}{2} \right. \right. \\ & \quad \left. \left. + \frac{\left( \frac{d}{dr} k_{Vodd}(r) \right) r^2}{2} - \frac{\left( \frac{d}{dr} k_{Veven}(r) \right) r^2}{2} - \frac{v_{scalar}(r) r}{4} - \frac{\left( \frac{d}{dr} v_{odd}(r) \right) r}{2} \right. \right. \\ & \quad \left. \left. + v_{odd}(r) \right) \sqrt{M} + \left( -3M h_{Vodd}(r) + \frac{(\lambda+2) h_{Todd}(r)}{2} \right) \sqrt{r} - 2 \left( M \left( \frac{d}{dr} h_{Veven}(r) \right) \right. \right. \\ & \quad \left. \left. + \frac{h_{Veven}(r)}{2} \right) r^3 \right) \end{aligned} \quad (67)$$

**> expand(simplify((coeff(Kdot\_\_Veven,sqrt(M))/sqrt(2/r))) assuming r>0;**

$$-r \left( \frac{d}{dr} h_{even}(r) \right) + \frac{d}{dr} k_{Veven}(r) + \frac{v_{scalar}(r)}{2r} \quad (68)$$

Mismatch here.....???

**> Kdot\_\_VoddX:=collect(collect(collect(collect(expand(subs(h\_\_Vodd(r)=r^(3/2)\*H\_\_Vodd(r),v\_\_odd(r)=r^2\*v\_\_odd(r),h\_\_Todd(r)=r^2\*H\_\_Todd(r),DDN\_\_Vodd-NRc\_\_V12odd-NKK\_\_odd+LVK\_\_Vodd),H\_\_Vodd(r),H\_\_Todd(r)),diff(H\_\_Todd(r),r),M);**

$$\begin{aligned} Kdot_{VoddX} := & \frac{2M \left( \frac{d}{dr} H_{Vodd}(r) \right)}{\sqrt{r}} + \left( \frac{\sqrt{2} \left( \frac{d}{dr} k_{Vodd}(r) \right)}{\sqrt{r}} - \sqrt{2} \sqrt{r} \left( \frac{d}{dr} \right. \right. \\ & \left. \left. V_{odd}(r) \right) \right) \sqrt{M} + \left( -\frac{\lambda}{4} - \frac{1}{2} \right) \left( \frac{d}{dr} H_{Todd}(r) \right) + \left( \frac{\lambda}{2\sqrt{r}} + \frac{1}{\sqrt{r}} \right) H_{Vodd}(r) \end{aligned} \quad (69)$$

**> factor(op(1,Kdot\_\_VoddX)+op(4,Kdot\_\_VoddX));**

(70)

$$\frac{4 M \left( \frac{d}{dr} H_{Vodd}(r) \right) + H_{Vodd}(r) \lambda + 2 H_{Vodd}(r)}{2 \sqrt{r}} \quad (70)$$

> simplify(op(3,Kdot\_\_VoddX)/sqrt(2\*M/r)) assuming M>0,r>0;

$$-\frac{(\lambda + 2) \left( \frac{d}{dr} H_{Todd}(r) \right) \sqrt{2} \sqrt{r}}{8 \sqrt{M}} \quad (71)$$

> Kdot\_\_Teven:=collect(collect(expand(DDN\_\_Teven-NRc\_\_T23even-NKK\_\_Teven+LVK\_\_Teven),h\_\_Teven(r)),M);

$$\begin{aligned} Kdot_{Teven} := & \left( -\frac{4 h_{Teven}(r)}{r^3} + \frac{4 h_{Veven}(r)}{r^2} \right) M + \left( -\frac{5 \sqrt{2} k_{Teven}(r)}{2 r^{3/2}} \right. \\ & + \frac{\sqrt{2} \left( \frac{d}{dr} k_{Teven}(r) \right)}{\sqrt{r}} - \frac{2 \sqrt{2} v_{even}(r)}{r^{3/2}} \left. \right) \sqrt{M} + \frac{h_{Teven}(r)}{r^2} + n_L(r) + \frac{h_{rr}(r)}{2} \\ & - \left( \frac{d}{dr} h_{Veven}(r) \right) + \frac{\left( \frac{d^2}{dr^2} h_{Teven}(r) \right)}{2} - \frac{\frac{d}{dr} h_{Teven}(r)}{r} \end{aligned} \quad (72)$$

> Kdot\_\_TevenX:=collect(collect(expand(subs(h\_\_Teven(r)=r\*H\_\_Teven(r),k\_\_Teven(r)=K\_\_Teven(r)\*r^(5/2),Kdot\_\_Teven)),h\_\_Teven(r)),M);

$$\begin{aligned} Kdot_{TevenX} := & \left( -\frac{4 H_{Teven}(r)}{r^2} + \frac{4 h_{Veven}(r)}{r^2} \right) M + \left( r^2 \sqrt{2} \left( \frac{d}{dr} K_{Teven}(r) \right) \right. \\ & - \frac{2 \sqrt{2} v_{even}(r)}{r^{3/2}} \left. \right) \sqrt{M} + n_L(r) + \frac{h_{rr}(r)}{2} - \left( \frac{d}{dr} h_{Veven}(r) \right) + \frac{r \left( \frac{d^2}{dr^2} H_{Teven}(r) \right)}{2} \end{aligned} \quad (73)$$

> factor(factor(op(1,Kdot\_\_TevenX))/M\*r^2);  
simplify(expand(factor(op(2,Kdot\_\_TevenX))/sqrt(2\*M/r^3)))  
assuming r>0,M>0;

$$\begin{aligned} & -4 H_{Teven}(r) + 4 h_{Veven}(r) \\ & \left( \frac{d}{dr} K_{Teven}(r) \right) r^{7/2} - 2 v_{even}(r) \end{aligned} \quad (74)$$

> Kdot\_\_Todd:=collect(expand((DDN\_\_Todd-NRc\_\_T23odd-NKK\_\_Todd+LVK\_\_Todd)),M);

$$\begin{aligned} Kdot_{Todd} := & \left( -\frac{4 h_{Todd}(r)}{r^3} + \frac{4 h_{Vodd}(r)}{r^2} \right) M + \left( -\frac{5 \sqrt{2} k_{Todd}(r)}{2 r^{3/2}} + \frac{\sqrt{2} \left( \frac{d}{dr} k_{Todd}(r) \right)}{\sqrt{r}} \right. \\ & - \frac{2 \sqrt{2} v_{odd}(r)}{r^{3/2}} \left. \right) \sqrt{M} - \left( \frac{d}{dr} h_{Vodd}(r) \right) + \frac{\left( \frac{d^2}{dr^2} h_{Todd}(r) \right)}{2} - \frac{\frac{d}{dr} h_{Todd}(r)}{r} \\ & + \frac{h_{Todd}(r)}{r^2} \end{aligned} \quad (75)$$

> Kdot\_\_ToddX:=collect(collect(expand(subs(h\_\_Todd(r)=r\*H\_\_Todd(r),  
k\_\_Todd(r)=K\_\_Todd(r)\*r^(5/2),Kdot\_\_Todd)),h\_\_Todd(r)),M);

$$Kdot_{ToddX} := \left( -\frac{4 H_{Todd}(r)}{r^2} + \frac{4 h_{Vodd}(r)}{r^2} \right) M + \left( r^2 \sqrt{2} \left( \frac{d}{dr} K_{Todd}(r) \right) - \frac{2 \sqrt{2} v_{odd}(r)}{r^{3/2}} \right) \sqrt{M} - \left( \frac{d}{dr} h_{Vodd}(r) \right) + \frac{r \left( \frac{d^2}{dr^2} H_{Todd}(r) \right)}{2} \quad (76)$$

> Kdot\_\_TOmega:=collect(collect(collect(collect(expand(DDN\_\_TOmega-NRc\_\_TOmega-NKK\_\_TOmega+LVK\_\_TOmega)),h\_\_Teven(r)),h\_\_rr(r)),h\_\_trace(r)),M);

$$Kdot_{TOmega} := \left( \frac{n_L(r)}{r} + \frac{2 h_{Veven}(r) \lambda}{r^2} \right) M + \left( \sqrt{r} \sqrt{2} k_{rr}(r) - \frac{\sqrt{2} k_{trace}(r)}{2 r^{3/2}} - \frac{\sqrt{2} v_{even}(r) \lambda}{r^{3/2}} - \frac{\sqrt{2} v_{scalar}(r)}{2 \sqrt{r}} + \frac{\sqrt{2} \left( \frac{d}{dr} k_{trace}(r) \right)}{\sqrt{r}} \right) \sqrt{M} + \left( \frac{\lambda}{4} - 1 \right) h_{rr}(r) + \left( -\frac{\lambda^2}{4 r^2} - \frac{\lambda}{2 r^2} \right) h_{Teven}(r) + \frac{n_L(r) \lambda}{2} + \left( \frac{d}{dr} n_L(r) \right) r - \frac{r \left( \frac{d}{dr} h_{rr}(r) \right)}{2} - \frac{\left( \frac{d}{dr} h_{Veven}(r) \right) \lambda}{2} + \frac{\left( \frac{d^2}{dr^2} h_{trace}(r) \right)}{2} - \frac{h_{Veven}(r) \lambda}{r} + \frac{h_{trace}(r) \lambda}{2 r^2} \quad (77)$$

> Kdot\_\_TOmegaX:=collect(collect(collect(collect(expand(subs(h\_\_rr(r)=H\_\_rr(r)/r^2,k\_\_trace(r)=sqrt(r)\*K\_\_trace(r),h\_\_Veven(r)=H\_\_Veven(r)/r^2,2\*Kdot\_\_TOmega)),h\_\_Teven(r)),h\_\_rr(r)),h\_\_trace(r)),M);

$$Kdot_{TOmegaX} := \left( \frac{2 n_L(r)}{r} + \frac{4 H_{Veven}(r) \lambda}{r^4} \right) M + \left( 2 \sqrt{r} \sqrt{2} k_{rr}(r) - \frac{2 \sqrt{2} v_{even}(r) \lambda}{r^{3/2}} - \frac{\sqrt{2} v_{scalar}(r)}{\sqrt{r}} + 2 \sqrt{2} \left( \frac{d}{dr} K_{trace}(r) \right) \right) \sqrt{M} + \left( -\frac{\lambda^2}{2 r^2} - \frac{\lambda}{r^2} \right) h_{Teven}(r) + \frac{H_{rr}(r) \lambda}{2 r^2} + n_L(r) \lambda + 2 \left( \frac{d}{dr} n_L(r) \right) r - \frac{\frac{d}{dr} H_{rr}(r)}{r} - \frac{\lambda \left( \frac{d}{dr} H_{Veven}(r) \right)}{r^2} + \frac{d^2}{dr^2} h_{trace}(r) + \frac{h_{trace}(r) \lambda}{r^2} \quad (78)$$

> expand(coeff(Kdot\_\_TOmegaX,sqrt(M))/sqrt(2/r)) assuming r>0;

$$2 r k_{rr}(r) - \frac{2 v_{even}(r) \lambda}{r} - v_{scalar}(r) + 2 \sqrt{r} \left( \frac{d}{dr} K_{trace}(r) \right) \quad (79)$$

Gauge Tranformations

```
> grdef(`xi{a}:=[epsilon*xi__r(r)*Y(theta, phi), epsilon*xi__even
(r)*Y__2+epsilon*xi__odd(r)*X__2, epsilon*xi__even(r)*Y__3+
epsilon*xi__odd(r)*X__3]`);
grcalc(xi(dn));
```

```
grmap(xi(dn), Linearize, 'x');
grmap(xi(dn), hExpand, 'x');
grmap(xi(dn), RemPhi, 'x');
grmap(xi(dn), ApplyId, 'x');
grmap(xi(dn), FixSin, 'x');
grmap(xi(dn), RemPhi, 'x');
grmap(xi(dn), RemPhi, 'x');
```

```
grdisplay(_);
```

Components assigned for metric: lpschwNewFixed  
Created definition for xi(dn)

*CPU Time = 0.*

Procedure name:LinearizeApplying routine Linearize to xi(dn)  
Procedure name:hExpandApplying routine hExpand to xi(dn)  
Procedure name:RemPhiApplying routine RemPhi to xi(dn)  
Procedure name:ApplyIdApplying routine ApplyId to xi(dn)  
Procedure name:FixSinApplying routine FixSin to xi(dn)  
Procedure name:RemPhiApplying routine RemPhi to xi(dn)  
Procedure name:RemPhiApplying routine RemPhi to xi(dn)

*For the lpschwNewFixed spacetime:*

$$xi(dn)$$

$$\xi(dn)$$

$$\xi_r = \epsilon \xi_r(r) Y(\theta, \phi)$$

$$\xi_\theta = \epsilon \xi_{even}(r) \left( \frac{\partial}{\partial \theta} Y(\theta, \phi) \right) - \frac{I \epsilon \xi_{odd}(r) m Y(\theta, \phi)}{\sin(\theta)}$$

$$\xi_\phi = I \epsilon \xi_{even}(r) m Y(\theta, \phi) + \epsilon \xi_{odd}(r) \sin(\theta) \left( \frac{\partial}{\partial \theta} Y(\theta, \phi) \right) \quad (80)$$

```
> grdef(`Lg__xi{a b}:=xi{a ;b}+xi{b ;a}`);
grcalc(Lg__xi(dn, dn));
```

```
grmap(Lg__xi(dn, dn), Linearize, 'x');
grmap(Lg__xi(dn, dn), hExpand, 'x');
grmap(Lg__xi(dn, dn), RemPhi, 'x');
grmap(Lg__xi(dn, dn), ApplyId, 'x');
grmap(Lg__xi(dn, dn), FixSin, 'x');
grmap(Lg__xi(dn, dn), RemPhi, 'x');
grmap(Lg__xi(dn, dn), RemPhi, 'x');
```

Created a definition for xi(dn,cdn)

Created definition for Lg\_\_xi(dn,dn)

Calculated xi(dn,cdn) for lpschwNewFixed (0.000000 sec.)

Calculated Lg\_\_xi(dn,dn) for lpschwNewFixed (0.016000 sec.)

*CPU Time = 0.016*

Procedure name:LinearizeApplying routine Linearize to Lg\_\_xi(dn,

```

dn)
Procedure name:hExpandApplying routine hExpand to Lg__xi(dn,dn)
Procedure name:RemPhiApplying routine RemPhi to Lg__xi(dn,dn)
Procedure name:ApplyIdApplying routine ApplyId to Lg__xi(dn,dn)
Procedure name:FixSinApplying routine FixSin to Lg__xi(dn,dn)
Procedure name:RemPhiApplying routine RemPhi to Lg__xi(dn,dn)
Procedure name:RemPhiApplying routine RemPhi to Lg__xi(dn,dn)

```

```

> grdef(`GGNew{a b}:=g{a b} + Lg__xi{a b}`);
grcalc(GGNew(dn, dn));

```

```

grmap(GGNew(dn,dn), Linearize, `x`);
grmap(GGNew(dn,dn), hExpand, 'x');
grmap(GGNew(dn,dn), RemPhi, 'x');
grmap(GGNew(dn,dn), ApplyId, 'x');
grmap(GGNew(dn,dn), FixSin, 'x');
grmap(GGNew(dn,dn), RemPhi, 'x');
grmap(GGNew(dn,dn), RemPhi, 'x');

```

Created definition for GGNew(dn,dn)  
 Calculated GGNew(dn,dn) for lpschwNewFixed (0.000000 sec.)  
 CPU Time =0.

```

Procedure name:LinearizeApplying routine Linearize to GGNew(dn,
dn)

```

```

Procedure name:hExpandApplying routine hExpand to GGNew(dn,dn)
Procedure name:RemPhiApplying routine RemPhi to GGNew(dn,dn)
Procedure name:ApplyIdApplying routine ApplyId to GGNew(dn,dn)
Procedure name:FixSinApplying routine FixSin to GGNew(dn,dn)
Procedure name:RemPhiApplying routine RemPhi to GGNew(dn,dn)
Procedure name:RemPhiApplying routine RemPhi to GGNew(dn,dn)

```

```

> LGG__xi__r:=grcomponent(GGNew(dn, dn), [r, r])/epsilon/Y(theta,
phi);
LGG__xi__Veven, LGG__xi__Vodd:=IdV2(grcomponent(GGNew(dn, dn),
[r, theta]));
LGG__xi__Teven, LGG__xi__Todd:=IdT23(grcomponent(GGNew(dn, dn),
[theta, phi]));
LGG__xi__Trace:=IdT22(grcomponent(GGNew(dn, dn), [theta, theta]),
LGG__xi__Teven, LGG__xi__Todd);

```

$$LGG_{\xi_r} := \frac{1 + \epsilon h_{rr}(r) Y(\theta, \phi) + 2 \left( \frac{d}{dr} \xi_r(r) \right) Y(\theta, \phi) \epsilon}{\epsilon Y(\theta, \phi)}$$

$$LGG_{\xi_{Veven}}, LGG_{\xi_{Vodd}} := \frac{d}{dr} \xi_{even}(r) + \xi_r(r) - \frac{2 \xi_{even}(r)}{r} + h_{Veven}(r), \frac{d}{dr} \xi_{odd}(r) + h_{Vodd}(r) - \frac{2 \xi_{odd}(r)}{r}$$

$$LGG_{\xi_{Teven}}, LGG_{\xi_{Todd}} := 2 \xi_{even}(r) + h_{Teven}(r), 2 \xi_{odd}(r) + h_{Todd}(r)$$

$$LGG_{\xi_{Trace}} := \xi_{even}(r) \lambda + 2 r \xi_r(r) + h_{trace}(r) + \frac{r^2}{Y(\theta, \phi) \epsilon}$$

(81)

```

> Lg__xi__r:=grcomponent(Lg__xi(dn, dn), [r, r])/epsilon/Y(theta,
phi);
Lg__xi__Veven, Lg__xi__Vodd:=IdV2(grcomponent(Lg__xi(dn, dn), [r,

```

```
theta])));
Lg__xi__Teven, Lg__xi__Todd:=IdT23(grcomponent(Lg__xi(dn, dn),
[theta, phi]));
```

$$Lg_{\xi_r} := 2 \left( \frac{d}{dr} \xi_r(r) \right)$$

$$Lg_{\xi_{Teven}}, Lg_{\xi_{Todd}} := \xi_r(r) + \frac{d}{dr} \xi_{even}(r) - \frac{2 \xi_{even}(r)}{r}, \frac{d}{dr} \xi_{odd}(r) - \frac{2 \xi_{odd}(r)}{r}$$

$$Lg_{\xi_{Teven}}, Lg_{\xi_{Todd}} := 2 \xi_{even}(r), 2 \xi_{odd}(r) \quad (82)$$

```
> grcalc(xi(up));
grmap(xi(up), Linearize, 'x');
Created definition for xi(up)
Calculated xi(up) for lpschwNewFixed (0.000000 sec.)
CPU Time =0.
Procedure name:LinearizeApplying routine Linearize to xi(up)
> Lg__xi__Trace:=IdT22(grcomponent(Lg__xi(dn, dn), [theta, theta]),
Lg__xi__Teven, Lg__xi__Todd);
Lg_{\xi_{Trace}} := \xi_{even}(r) \lambda + 2 r \xi_r(r) \quad (83)
```

```
> grdef(`LgG__xi{a b}:=xi{^l}*g{a b ,l} + g{a p}*xi{^p ,b} + g{b q}
*x{i{^q ,a}`);
grcalc(LgG__xi(dn,dn));
```

```
grmap(LgG__xi(dn,dn), Linearize, `x`);
grmap(LgG__xi(dn,dn), hExpand, 'x');
grmap(LgG__xi(dn,dn), RemPhi, 'x');
grmap(LgG__xi(dn,dn), ApplyId, 'x');
grmap(LgG__xi(dn,dn), FixSin, 'x');
grmap(LgG__xi(dn,dn), RemPhi, 'x');
grmap(LgG__xi(dn,dn), RemPhi, 'x');
```

```
grdisplay(LgG__xi(dn,dn));
```

```
Created a definition for xi(up,pdn)
Created definition for LgG__xi(dn,dn)
Calculated xi(up,pdn) for lpschwNewFixed (0.000000 sec.)
Calculated LgG__xi(dn,dn) for lpschwNewFixed (0.000000 sec.)
CPU Time =0.
Procedure name:LinearizeApplying routine Linearize to LgG__xi
(dn,dn)
Procedure name:hExpandApplying routine hExpand to LgG__xi(dn,dn)
Procedure name:RemPhiApplying routine RemPhi to LgG__xi(dn,dn)
Procedure name:ApplyIdApplying routine ApplyId to LgG__xi(dn,dn)
Procedure name:FixSinApplying routine FixSin to LgG__xi(dn,dn)
Procedure name:RemPhiApplying routine RemPhi to LgG__xi(dn,dn)
Procedure name:RemPhiApplying routine RemPhi to LgG__xi(dn,dn)
```

*For the lpschwNewFixed spacetime:*

$$LgG_{xi(dn,dn)}$$

$$LgG_{\xi}(dn, dn)$$

$$LgG_{\xi_r} = 2 \left( \frac{d}{dr} \xi_r(r) \right) Y(\theta, \phi) \in$$

$$LgG_{\xi_{\theta} r} = - \frac{I \in \left( \frac{d}{dr} \xi_{odd}(r) \right) Y(\theta, \phi) m}{\sin(\theta)} + \frac{2 I \in \xi_{odd}(r) Y(\theta, \phi) m}{r \sin(\theta)} + \epsilon \xi_r(r) \left( \frac{\partial}{\partial \theta} Y(\theta, \phi) \right) \\ + \epsilon \left( \frac{d}{dr} \xi_{even}(r) \right) \left( \frac{\partial}{\partial \theta} Y(\theta, \phi) \right) - \frac{2 \epsilon \xi_{even}(r) \left( \frac{\partial}{\partial \theta} Y(\theta, \phi) \right)}{r}$$

$$LgG_{\xi_{\phi} r} = \epsilon \left( \frac{d}{dr} \xi_{odd}(r) \right) \left( \frac{\partial}{\partial \theta} Y(\theta, \phi) \right) \sin(\theta) - \frac{2 \epsilon \xi_{odd}(r) \sin(\theta) \left( \frac{\partial}{\partial \theta} Y(\theta, \phi) \right)}{r} \\ + I \in \xi_r(r) m Y(\theta, \phi) + I \in \left( \frac{d}{dr} \xi_{even}(r) \right) Y(\theta, \phi) m - \frac{2 I \in \xi_{even}(r) m Y(\theta, \phi)}{r}$$

$$LgG_{\xi_r \theta} = - \frac{I \in \left( \frac{d}{dr} \xi_{odd}(r) \right) Y(\theta, \phi) m}{\sin(\theta)} + \frac{2 I \in \xi_{odd}(r) Y(\theta, \phi) m}{r \sin(\theta)} + \epsilon \xi_r(r) \left( \frac{\partial}{\partial \theta} Y(\theta, \phi) \right) \\ + \epsilon \left( \frac{d}{dr} \xi_{even}(r) \right) \left( \frac{\partial}{\partial \theta} Y(\theta, \phi) \right) - \frac{2 \epsilon \xi_{even}(r) \left( \frac{\partial}{\partial \theta} Y(\theta, \phi) \right)}{r}$$

$$LgG_{\xi_{\theta} \theta} = - \frac{2 I \in \xi_{odd}(r) \left( \frac{\partial}{\partial \theta} Y(\theta, \phi) \right) m}{\sin(\theta)} + \frac{2 I \in \xi_{odd}(r) Y(\theta, \phi) \cos(\theta) m}{\sin(\theta)^2} \\ + 2 \epsilon \xi_r(r) Y(\theta, \phi) r + \frac{2 \epsilon \xi_{even}(r) m^2 Y(\theta, \phi)}{\sin(\theta)^2} - 2 \epsilon \xi_{even}(r) \cot(\theta) \left( \frac{\partial}{\partial \theta} Y(\theta, \phi) \right) \\ + 2 \epsilon \xi_{even}(r) \lambda Y(\theta, \phi)$$

$$LgG_{\xi_{\phi} \theta} = 2 I \in \xi_{even}(r) \left( \frac{\partial}{\partial \theta} Y(\theta, \phi) \right) m - \frac{2 I \in \xi_{even}(r) Y(\theta, \phi) \cos(\theta) m}{\sin(\theta)} \\ + \frac{2 \epsilon \xi_{odd}(r) m^2 Y(\theta, \phi)}{\sin(\theta)} - \epsilon \xi_{odd}(r) \left( \frac{\partial}{\partial \theta} Y(\theta, \phi) \right) \cos(\theta) \\ - \epsilon \sin(\theta) \xi_{odd}(r) \cot(\theta) \left( \frac{\partial}{\partial \theta} Y(\theta, \phi) \right) + \epsilon \sin(\theta) \xi_{odd}(r) \lambda Y(\theta, \phi)$$

$$LgG_{\xi_r \phi} = \epsilon \left( \frac{d}{dr} \xi_{odd}(r) \right) \left( \frac{\partial}{\partial \theta} Y(\theta, \phi) \right) \sin(\theta) - \frac{2 \epsilon \xi_{odd}(r) \sin(\theta) \left( \frac{\partial}{\partial \theta} Y(\theta, \phi) \right)}{r} \\ + I \in \xi_r(r) m Y(\theta, \phi) + I \in \left( \frac{d}{dr} \xi_{even}(r) \right) Y(\theta, \phi) m - \frac{2 I \in \xi_{even}(r) m Y(\theta, \phi)}{r}$$

$$LgG_{\xi_{\theta} \phi} = 2 I \in \xi_{even}(r) \left( \frac{\partial}{\partial \theta} Y(\theta, \phi) \right) m - \frac{2 I \in \xi_{even}(r) Y(\theta, \phi) \cos(\theta) m}{\sin(\theta)}$$

$$\begin{aligned}
& + \frac{2 \epsilon \xi_{odd}(r) m^2 Y(\theta, \phi)}{\sin(\theta)} - \epsilon \xi_{odd}(r) \left( \frac{\partial}{\partial \theta} Y(\theta, \phi) \right) \cos(\theta) \\
& - \epsilon \sin(\theta) \xi_{odd}(r) \cot(\theta) \left( \frac{\partial}{\partial \theta} Y(\theta, \phi) \right) + \epsilon \sin(\theta) \xi_{odd}(r) \lambda Y(\theta, \phi)
\end{aligned}$$

$$\begin{aligned}
LgG_{\xi_\phi} = & -2 I \epsilon \xi_{odd}(r) Y(\theta, \phi) \cos(\theta) m + 2 \epsilon \xi_r(r) Y(\theta, \phi) r \sin(\theta)^2 - 2 \epsilon \xi_{even}(r) m^2 Y(\theta, \\
& \phi) + 2 \epsilon \xi_{even}(r) \left( \frac{\partial}{\partial \theta} Y(\theta, \phi) \right) \cos(\theta) \sin(\theta) + 2 I \epsilon \xi_{odd}(r) \sin(\theta) m \left( \frac{\partial}{\partial \theta} Y(\theta, \phi) \right)
\end{aligned} \quad (84)$$

Lie derivative of Extrinsic Curvature

```

> grdef(`LK__xi{a b}:=xi{^l}*KK{a b ,l}+ KK{a p}*xi{^p ,b}+KK{b q}*
xi{^q ,a}^`);
grcalc(LK__xi(dn, dn));

```

```

grmap(LK__xi(dn, dn), Linearize, 'x');
grmap(LK__xi(dn, dn), FixSin, 'x');
grmap(LK__xi(dn, dn), RemPhi, 'x');
grmap(LK__xi(dn, dn), ApplyId, 'x');
grmap(LK__xi(dn, dn), FixSin, 'x');
grmap(LK__xi(dn, dn), RemPhi, 'x');

```

Created definition for LK\_\_xi(dn,dn)

Calculated LK\_\_xi(dn,dn) for lpschwNewFixed (0.000000 sec.)

CPU Time =0.

Procedure name:LinearizeApplying routine Linearize to LK\_\_xi(dn, dn)

Procedure name:FixSinApplying routine FixSin to LK\_\_xi(dn,dn)

Procedure name:RemPhiApplying routine RemPhi to LK\_\_xi(dn,dn)

Procedure name:ApplyIdApplying routine ApplyId to LK\_\_xi(dn,dn)

Procedure name:FixSinApplying routine FixSin to LK\_\_xi(dn,dn)

Procedure name:RemPhiApplying routine RemPhi to LK\_\_xi(dn,dn)

```

> LK__xi__r:=simplify(grcomponent(LK__xi(dn, dn), [r, r]));
LK__xi__12:=simplify(grcomponent(LK__xi(dn, dn), [r, theta]));
LK__xi__13:=simplify(grcomponent(LK__xi(dn, dn), [r, phi]));

```

$$LK_{\xi_r} := \frac{\epsilon \sqrt{2} \sqrt{M} Y(\theta, \phi) \left( 4 \left( \frac{d}{dr} \xi_r(r) \right) r - 3 \xi_r(r) \right)}{4 r^{5/2}}$$

$$\begin{aligned}
LK_{\xi_{12}} := & - \frac{1}{r^{5/2} \sin(\theta)} \left( 2 \sqrt{M} \left( \right. \right. \\
& \sin(\theta) \left( r \xi_r(r) - 2 \left( \frac{d}{dr} \xi_{even}(r) \right) r + 4 \xi_{even}(r) \right) \left( \frac{\partial}{\partial \theta} Y(\theta, \phi) \right) \\
& - \frac{4}{4} + I Y(\theta, \phi) \left( \right. \\
& \left. \left. - \frac{\left( \frac{d}{dr} \xi_{odd}(r) \right) r}{2} + \xi_{odd}(r) \right) m \right) \sqrt{2} \epsilon
\end{aligned}$$



$$LK_{\xi_{13}} := \frac{1}{2 r^{5/2}} \left( \sqrt{M} \left( 4 \sin(\theta) \left( -\frac{\left( \frac{d}{dr} \xi_{odd}(r) \right) r}{2} + \xi_{odd}(r) \right) \left( \frac{\partial}{\partial \theta} Y(\theta, \phi) \right) \right. \right. \\ \left. \left. + I \left( r \xi_r(r) - 2 \left( \frac{d}{dr} \xi_{even}(r) \right) r + 4 \xi_{even}(r) \right) Y(\theta, \phi) m \right) \sqrt{2} \epsilon \right) \quad (85)$$

Consistency check against Ivan's

```
> LK_xi_rIvan:=epsilon*sqrt(M)*sqrt(2)*Y(theta, phi)*(4*r*(diff
(xi__r(r), r))-3*xi__r(r))/(4*r^(5/2)):
```

```
LK_xi_12Ivan:=-2*epsilon*(-sin(theta)*(-(1/2)*(diff(xi__even
(r), r))*r+(1/4)*r*xi__r(r)+xi__even(r))*(diff(Y(theta, phi),
theta))+I*m*(-(1/2)*(diff(xi__odd(r), r))*r+xi__odd(r))*Y(theta,
phi))*sqrt(M)*sqrt(2)/(r^(5/2)*sin(theta)):
```

```
LK_xi_13Ivan:=-epsilon*sqrt(M)*sqrt(2)*sin(theta)*(diff(Y
(theta, phi), theta))*(diff(xi__odd(r), r))/r^(3/2)+2*epsilon*
sqrt(M)*sqrt(2)*xi__odd(r)*sin(theta)*(diff(Y(theta, phi), theta)
)/r^(5/2)+I*epsilon*sqrt(M)*sqrt(2)*m*Y(theta, phi)*xi__r(r)/(2*
r^(3/2))-I*epsilon*sqrt(M)*sqrt(2)*(diff(xi__even(r), r))*Y
(theta, phi)*m/r^(3/2)+(2*I)*epsilon*sqrt(M)*sqrt(2)*xi__even(r)*
Y(theta, phi)*m/r^(5/2):
```

```
simplify(LK_xi_r-LK_xi_rIvan);
simplify(LK_xi_12-LK_xi_12Ivan);
simplify(LK_xi_13-LK_xi_13Ivan);
```

0

0

0

(86)

```
> LK_xi_Veven, LK_xi_Vodd:=IdV2(LK_xi_12);
IdV3(LK_xi_13);
```

$$LK_{\xi_{Veven}}, LK_{\xi_{Vodd}} := \frac{\sqrt{M} \sqrt{2} \xi_r(r)}{2 r^{3/2}} - \frac{\sqrt{M} \sqrt{2} \left( \frac{d}{dr} \xi_{even}(r) \right)}{r^{3/2}} + \frac{2 \sqrt{M} \sqrt{2} \xi_{even}(r)}{r^{5/2}},$$

$$- \frac{\sqrt{M} \sqrt{2} \left( \frac{d}{dr} \xi_{odd}(r) \right)}{r^{3/2}} + \frac{2 \sqrt{M} \sqrt{2} \xi_{odd}(r)}{r^{5/2}}$$

$$\frac{\sqrt{M} \sqrt{2} \xi_r(r)}{2 r^{3/2}} - \frac{\sqrt{M} \sqrt{2} \left( \frac{d}{dr} \xi_{even}(r) \right)}{r^{3/2}} + \frac{2 \sqrt{M} \sqrt{2} \xi_{even}(r)}{r^{5/2}}, \quad (87)$$

$$- \frac{\sqrt{M} \sqrt{2} \left( \frac{d}{dr} \xi_{odd}(r) \right)}{r^{3/2}} + \frac{2 \sqrt{M} \sqrt{2} \xi_{odd}(r)}{r^{5/2}}$$

```
> LK_xi_22:=grcomponent(LK_xi(dn, dn), [theta, theta]);
LK_xi_23:=grcomponent(LK_xi(dn, dn), [theta, phi]);
LK_xi_33:=grcomponent(LK_xi(dn, dn), [phi, phi]);
```

$$\begin{aligned}
LK_{\xi_{22}} &:= - \frac{2 \epsilon \sqrt{M} \sqrt{2} \xi_{\text{even}}(r) m^2 Y(\theta, \phi)}{r^{3/2} \sin(\theta)^2} + \frac{2 \epsilon \sqrt{M} \sqrt{2} \xi_{\text{even}}(r) \cot(\theta) \left( \frac{\partial}{\partial \theta} Y(\theta, \phi) \right)}{r^{3/2}} \\
&- \frac{2 \epsilon \sqrt{M} \sqrt{2} \xi_{\text{even}}(r) \lambda Y(\theta, \phi)}{r^{3/2}} + \frac{2 I \epsilon \sqrt{M} \sqrt{2} \xi_{\text{odd}}(r) \left( \frac{\partial}{\partial \theta} Y(\theta, \phi) \right) m}{r^{3/2} \sin(\theta)} \\
&- \frac{2 I \epsilon \sqrt{M} \sqrt{2} \xi_{\text{odd}}(r) Y(\theta, \phi) \cos(\theta) m}{r^{3/2} \sin(\theta)^2} - \frac{\epsilon \sqrt{M} \sqrt{2} \xi_r(r) Y(\theta, \phi)}{2 \sqrt{r}} \\
LK_{\xi_{23}} &:= - \frac{2 I \epsilon \sqrt{M} \sqrt{2} \xi_{\text{even}}(r) \left( \frac{\partial}{\partial \theta} Y(\theta, \phi) \right) m}{r^{3/2}} \\
&+ \frac{2 I \epsilon \sqrt{M} \sqrt{2} \xi_{\text{even}}(r) Y(\theta, \phi) \cos(\theta) m}{r^{3/2} \sin(\theta)} - \frac{2 \epsilon \sqrt{M} \sqrt{2} \xi_{\text{odd}}(r) m^2 Y(\theta, \phi)}{r^{3/2} \sin(\theta)} \\
&+ \frac{\epsilon \sqrt{M} \sqrt{2} \xi_{\text{odd}}(r) \left( \frac{\partial}{\partial \theta} Y(\theta, \phi) \right) \cos(\theta)}{r^{3/2}} \\
&+ \frac{\epsilon \sqrt{M} \sqrt{2} \sin(\theta) \xi_{\text{odd}}(r) \cot(\theta) \left( \frac{\partial}{\partial \theta} Y(\theta, \phi) \right)}{r^{3/2}} \\
&- \frac{\epsilon \sqrt{M} \sqrt{2} \sin(\theta) \xi_{\text{odd}}(r) \lambda Y(\theta, \phi)}{r^{3/2}} \\
LK_{\xi_{33}} &:= \frac{2 I \epsilon \sqrt{M} \sqrt{2} \xi_{\text{odd}}(r) Y(\theta, \phi) \cos(\theta) m}{r^{3/2}} - \frac{\epsilon \sqrt{M} \sqrt{2} \xi_r(r) Y(\theta, \phi) \sin(\theta)^2}{2 \sqrt{r}} \\
&+ \frac{2 \epsilon \sqrt{M} \sqrt{2} \xi_{\text{even}}(r) m^2 Y(\theta, \phi)}{r^{3/2}} \\
&- \frac{2 \epsilon \sqrt{M} \sqrt{2} \xi_{\text{even}}(r) \left( \frac{\partial}{\partial \theta} Y(\theta, \phi) \right) \cos(\theta) \sin(\theta)}{r^{3/2}} \\
&- \frac{2 I \epsilon \sqrt{M} \sqrt{2} \xi_{\text{odd}}(r) \sin(\theta) m \left( \frac{\partial}{\partial \theta} Y(\theta, \phi) \right)}{r^{3/2}}
\end{aligned} \tag{88}$$

Consistency check against Ivan's

```

> LK_xi_22Ivan:=(2*I)*epsilon*sqrt(M)*sqrt(2)*xi__odd(r)*(diff(Y
(theta, phi), theta))*m/(r^(3/2)*sin(theta))+2*sqrt(2)*epsilon*
sqrt(M)*cos(theta)*xi__even(r)*(diff(Y(theta, phi), theta))/(r^
(3/2)*sin(theta))-epsilon*sqrt(M)*sqrt(2)*Y(theta, phi)*xi__r(r)/

```

```

(2*sqrt(r))-2*epsilon*sqrt(M)*sqrt(2)*xi__even(r)*lambda*Y(theta,
phi)/r^(3/2)-(2*I)*epsilon*sqrt(M)*sqrt(2)*cos(theta)*Y(theta,
phi)*xi__odd(r)*m/(r^(3/2)*sin(theta)^2)-2*sqrt(2)*epsilon*sqrt
(M)*xi__even(r)*m^2*Y(theta, phi)/(r^(3/2)*sin(theta)^2):
LK_xi__23Ivan:=(2*I)*epsilon*sqrt(2)*sqrt(M*r)*xi__even(r)*
(diff(Y(theta, phi), theta))*m/r^2+2*epsilon*sqrt(2)*sqrt(M*r)*
cos(theta)*xi__odd(r)*(diff(Y(theta, phi), theta))/r^2-epsilon*
sqrt(2)*sqrt(M*r)*sin(theta)*xi__odd(r)*lambda*Y(theta, phi)/r^2
-2*epsilon*sqrt(2)*sqrt(M*r)*xi__odd(r)*m^2*Y(theta, phi)/(sin
(theta)*r^2)+(2*I)*epsilon*sqrt(2)*sqrt(M*r)*cos(theta)*xi__even
(r)*Y(theta, phi)*m/(sin(theta)*r^2):
LK_xi__33Ivan:=(2*I)*epsilon*sqrt(M)*sqrt(2)*xi__odd(r)*sin
(theta)*(diff(Y(theta, phi), theta))*m/r^(3/2)-2*epsilon*sqrt(M)*
sqrt(2)*cos(theta)*xi__even(r)*sin(theta)*(diff(Y(theta, phi),
theta))/r^(3/2)+2*epsilon*sqrt(M)*sqrt(2)*m^2*xi__even(r)*Y
(theta, phi)/r^(3/2)+(2*I)*epsilon*sqrt(M)*sqrt(2)*cos(theta)*Y
(theta, phi)*xi__odd(r)*m/r^(3/2)+epsilon*sqrt(M)*sqrt(2)*cos
(theta)^2*Y(theta, phi)*xi__r(r)/(2*sqrt(r))-epsilon*sqrt(M)*sqrt
(2)*Y(theta, phi)*xi__r(r)/(2*sqrt(r)):

simplify(LK_xi__22-LK_xi__22Ivan);
simplify(LK_xi__23-LK_xi__23Ivan);
simplify(LK_xi__33-LK_xi__33Ivan);
0
0
0

```

(89)

```

> LK_xi__Teven,LK_xi__Todd:=IdT23(LK_xi__23);

```

$$LK_{\xi_{Teven}}, LK_{\xi_{Todd}} := -\frac{2\sqrt{2}\sqrt{M}\xi_{even}(r)}{r^{3/2}}, -\frac{2\sqrt{M}\sqrt{2}\xi_{odd}(r)}{r^{3/2}}$$

(90)

```

> LK_xi__Trace:=factor(IdT22(LK_xi__22,LK_xi__Teven,
LK_xi__Todd));

```

$$LK_{\xi_{Trace}} := -\frac{\sqrt{M}\sqrt{2}\left(2\xi_{even}(r)\lambda + r\xi_r(r)\right)}{2r^{3/2}}$$

(91)

Time

(Metric part is easy.....according to Ivan lol)

```

> xi__N:=epsilon*xi__0(r)*Y(theta, phi);

```

$$\xi_N := \epsilon \xi_0(r) Y(\theta, \phi)$$

(92)

```

> xidotg__11:=Linearize(xi__N*grcomponent(KK(dn, dn), [r, r]));

```

$$xidotg_{11} := \frac{\xi_0(r) Y(\theta, \phi) \sqrt{2} \sqrt{M} \epsilon}{2r^{3/2}}$$

(93)

```

> xidotg__12:=Linearize(xi__N*grcomponent(KK(dn, dn), [r, theta]));
xidotg__13:=Linearize(xi__N*grcomponent(KK(dn, dn), [r, phi]));

```

$$xidotg_{12} := 0$$

$$xidotg_{13} := 0 \quad (94)$$

```
> xidotg__22:=Linearize(xi__N*grcomponent(KK(dn, dn), [theta,
theta]));
xidotg__23:=Linearize(xi__N*grcomponent(KK(dn, dn), [theta, phi])
);
xidotg__33:=Linearize(xi__N*grcomponent(KK(dn, dn), [phi, phi]));
```

$$xidotg_{22} := -\xi_0(r) Y(\theta, \phi) \sqrt{2} \sqrt{M} \sqrt{r} \epsilon$$

$$xidotg_{23} := 0$$

$$xidotg_{33} := -\xi_0(r) Y(\theta, \phi) \sqrt{2} \sqrt{M} \sqrt{r} \sin(\theta)^2 \epsilon \quad (95)$$

```
> xidotg__r:=xidotg__11/epsilon/Y(theta, phi);
```

$$xidotg_r := \frac{\xi_0(r) \sqrt{2} \sqrt{M}}{2 r^{3/2}} \quad (96)$$

```
> xidotg__Veven, xidotg__Vodd:=IdV2(xidotg__12);
```

$$xidotg_{Veven}, xidotg_{Vodd} := 0, 0 \quad (97)$$

```
> xidotg__Teven, xidotg__Todd:=IdT23(xidotg__23);
```

$$xidotg_{Teven}, xidotg_{Todd} := 0, 0 \quad (98)$$

```
> xidotg__Trace:=IdT22(xidotg__22, xidotg__Teven, xidotg__Todd);
```

$$xidotg_{Trace} := -\xi_0(r) \sqrt{2} \sqrt{M} \sqrt{r} \quad (99)$$

## Extrinsic Curvature Deformation

There needed to be some additional setup for the first term in the deformation which is the double covariant derivative of the Lapse function

```
> grdef(`xi__NTest:=epsilon*xi__0(r)*Y(theta, phi)`);
grcalc(xi__NTest);
grcalc(xi__NTest(cdn, cdn));
```

```
grmap(xi__NTest(cdn, cdn), Linearize, 'x');
```

```
grdef(`xi__NLast{a b}:=xi__NTest{;a ;b}`);
```

Created definition for xi\_\_NTest

Calculated xi\_\_NTest for lpschwNewFixed (0.000000 sec.)

CPU Time =0.

Created a definition for xi\_\_NTest(cdn)

Created a definition for xi\_\_NTest(cdn,cdn)

Calculated xi\_\_NTest(cdn) for lpschwNewFixed (0.000000 sec.)

Calculated xi\_\_NTest(cdn,cdn) for lpschwNewFixed (0.000000 sec.)

CPU Time =0.031

Procedure name:LinearizeApplying routine Linearize to xi\_\_NTest(cdn,cdn)

Created definition for xi\_\_NLast(dn, dn)

```
> grdef(`DDxi__N{a b}:=xi__NLast{a b}`);
grcalc(DDxi__N(dn, dn));
```

Created definition for DDxi\_\_N(dn,dn)  
 Calculated xi\_\_NLast(dn,dn) for lpschwNewFixed (0.000000 sec.)  
 Calculated DDxi\_\_N(dn,dn) for lpschwNewFixed (0.000000 sec.)  
 CPU Time = 0.

(100)

```
> grdef(`xiKK{a b}:=xi__N*(KTr*KK{a b}-2*g{^k ^l}*KK{a k}*KK{b l})
`);
grcalc(xiKK(dn, dn));
```

```
grmap(xiKK(dn, dn), Linearize, 'x');
grmap(xiKK(dn, dn), RemPhi, 'x');
grmap(xiKK(dn, dn), FixSin, 'x');
grmap(xiKK(dn, dn), ApplyId, 'x');
grmap(xiKK(dn, dn), RemPhi, 'x');
grmap(xiKK(dn, dn), FixSin, 'x');
```

Created definition for xiKK(dn,dn)  
 Calculated xiKK(dn,dn) for lpschwNewFixed (0.000000 sec.)  
 CPU Time = 0.

Procedure name:LinearizeApplying routine Linearize to xiKK(dn, dn)

Procedure name:RemPhiApplying routine RemPhi to xiKK(dn,dn)

Procedure name:FixSinApplying routine FixSin to xiKK(dn,dn)

Procedure name:ApplyIdApplying routine ApplyId to xiKK(dn,dn)

Procedure name:RemPhiApplying routine RemPhi to xiKK(dn,dn)

Procedure name:FixSinApplying routine FixSin to xiKK(dn,dn)

```
> xidotK__11:=simplify(grcomponent(DDxi__N(dn, dn), [r, r])-
grcomponent(xiKK(dn, dn), [r, r]));
xidotK__12:=simplify(grcomponent(DDxi__N(dn, dn), [r, theta])-
grcomponent(xiKK(dn, dn), [r, theta]));
xidotK__13:=simplify(grcomponent(DDxi__N(dn, dn), [r, phi])-
grcomponent(xiKK(dn, dn), [r, phi]));
xidotK__22:=ApplyId(RemPhi(simplify(grcomponent(DDxi__N(dn, dn),
[theta, theta])-grcomponent(xiKK(dn, dn), [theta, theta]))));
xidotK__23:=ApplyId(RemPhi(simplify(grcomponent(DDxi__N(dn, dn),
[theta, phi])-grcomponent(xiKK(dn, dn), [theta, phi]))));
xidotK__33:=ApplyId(RemPhi(simplify(grcomponent(DDxi__N(dn, dn),
[phi, phi])-grcomponent(xiKK(dn, dn), [phi, phi]))));
```

$$xidotK_{11} := \frac{Y(\theta, \phi) \in \left( 2 \left( \frac{d^2}{dr^2} \xi_0(r) \right) r^3 + 5 \xi_0(r) M \right)}{2 r^3}$$

$$xidotK_{12} := \frac{\in \left( \frac{\partial}{\partial \theta} Y(\theta, \phi) \right) \left( \left( \frac{d}{dr} \xi_0(r) \right) r - \xi_0(r) \right)}{r}$$

$$xidotK_{13} := \frac{\in \left( \frac{\partial}{\partial \phi} Y(\theta, \phi) \right) \left( \left( \frac{d}{dr} \xi_0(r) \right) r - \xi_0(r) \right)}{r}$$

$$xidotK_{22} := \in r \left( \frac{d}{dr} \xi_0(r) \right) Y(\theta, \phi) + \frac{\xi_0(r) Y(\theta, \phi) M \in}{r} + \frac{\in \xi_0(r) m^2 Y(\theta, \phi)}{\sin(\theta)^2}$$

$$- \in \xi_0(r) \cot(\theta) \left( \frac{\partial}{\partial \theta} Y(\theta, \phi) \right) + \in \xi_0(r) \lambda Y(\theta, \phi)$$

$$\begin{aligned}
xidotK_{23} &:= I \in \xi_0(r) \, m \left( \frac{\partial}{\partial \theta} Y(\theta, \phi) \right) - \frac{I \in \xi_0(r) \cos(\theta) \, m Y(\theta, \phi)}{\sin(\theta)} \\
xidotK_{33} &:= I \in \xi_0(r) \, m \left( \frac{\partial}{\partial \phi} Y(\theta, \phi) \right) + \epsilon \xi_0(r) \left( \frac{\partial}{\partial \theta} Y(\theta, \phi) \right) \cos(\theta) \sin(\theta) - \epsilon r \left( \frac{d}{dr} \right.
\end{aligned} \tag{101}$$

$$\begin{aligned}
&\xi_0(r) \left. \right) Y(\theta, \phi) \cos(\theta)^2 - \frac{\epsilon M \xi_0(r) Y(\theta, \phi) \cos(\theta)^2}{r} + \epsilon r \left( \frac{d}{dr} \xi_0(r) \right) Y(\theta, \phi) \\
&+ \frac{\xi_0(r) Y(\theta, \phi) M \epsilon}{r}
\end{aligned}$$

**> xidotK\_\_r:=xidotK\_\_11;**

$$xidotK_r := \frac{Y(\theta, \phi) \in \left( 2 \left( \frac{d^2}{dr^2} \xi_0(r) \right) r^3 + 5 \xi_0(r) M \right)}{2 r^3} \tag{102}$$

**> xidotK\_\_Even, xidotK\_\_Vodd:=IdV2(xidotK\_\_12);**

$$xidotK_{Even}, xidotK_{Vodd} := \frac{d}{dr} \xi_0(r) - \frac{\xi_0(r)}{r}, 0 \tag{103}$$

**> xidotK\_\_Teven, xidotK\_\_Todd:=IdT23(xidotK\_\_23);**

$$xidotK_{Teven}, xidotK_{Todd} := \xi_0(r), 0 \tag{104}$$

**> xidotK\_\_Trace:=IdT22(xidotK\_\_22, xidotK\_\_Teven, xidotK\_\_Todd);**

$$xidotK_{Trace} := \left( \frac{d}{dr} \xi_0(r) \right) r + \frac{\xi_0(r) \lambda}{2} + \frac{\xi_0(r) M}{r} \tag{105}$$

Now the big combo

**> delta\_\_r:=Lg\_\_xi\_\_r + xidotg\_\_r;  
deltag\_\_Even:=Lg\_\_xi\_\_Even + xidotg\_\_Even;  
deltag\_\_Vodd:=Lg\_\_xi\_\_Vodd + xidotg\_\_Vodd;  
deltag\_\_Teven:=Lg\_\_xi\_\_Teven + xidotg\_\_Teven;  
deltag\_\_Todd:=Lg\_\_xi\_\_Todd + xidotg\_\_Todd;  
deltag\_\_Trace:=Lg\_\_xi\_\_Trace + xidotg\_\_Trace;**

$$\delta_r := 2 \left( \frac{d}{dr} \xi_r(r) \right) + \frac{\xi_0(r) \sqrt{2} \sqrt{M}}{2 r^{3/2}}$$

$$deltag_{Even} := \xi_r(r) + \frac{d}{dr} \xi_{even}(r) - \frac{2 \xi_{even}(r)}{r}$$

$$deltag_{Vodd} := \frac{d}{dr} \xi_{odd}(r) - \frac{2 \xi_{odd}(r)}{r}$$

$$deltag_{Teven} := 2 \xi_{even}(r)$$

$$deltag_{Todd} := 2 \xi_{odd}(r)$$

$$deltag_{Trace} := \xi_{even}(r) \lambda + 2 r \xi_r(r) - \xi_0(r) \sqrt{2} \sqrt{M} \sqrt{r} \tag{106}$$

**> deltaK\_\_r:=collect((LK\_\_xi\_\_r + xidotK\_\_r)/epsilon/Y(theta, phi),**

```

M);
deltaK_Veven:=collect((LK_xi_Veven + xidotK_Veven), M);
deltaK_Vodd:=collect((LK_xi_Vodd + xidotK_Vodd), M);
deltaK_Teven:=LK_xi_Teven + xidotK_Teven;
deltaK_Todd:=LK_xi_Todd + xidotK_Todd;
deltaK_Trace:=collect(LK_xi_Trace + xidotK_Trace, M);

```

$$\text{deltaK}_r := \frac{5 \xi_0(r) M}{2 r^3} + \frac{\sqrt{2} \left( 4 \left( \frac{d}{dr} \xi_r(r) \right) r - 3 \xi_r(r) \right) \sqrt{M}}{4 r^{5/2}} + \frac{d^2}{dr^2} \xi_0(r)$$

$$\text{deltaK}_{V\text{even}} := \left( \frac{\sqrt{2} \xi_r(r)}{2 r^{3/2}} - \frac{\sqrt{2} \left( \frac{d}{dr} \xi_{\text{even}}(r) \right)}{r^{3/2}} + \frac{2 \sqrt{2} \xi_{\text{even}}(r)}{r^{5/2}} \right) \sqrt{M} + \frac{d}{dr} \xi_0(r) - \frac{\xi_0(r)}{r}$$

$$\text{deltaK}_{V\text{odd}} := \left( - \frac{\sqrt{2} \left( \frac{d}{dr} \xi_{\text{odd}}(r) \right)}{r^{3/2}} + \frac{2 \sqrt{2} \xi_{\text{odd}}(r)}{r^{5/2}} \right) \sqrt{M}$$

$$\text{deltaK}_{T\text{even}} := - \frac{2 \sqrt{2} \sqrt{M} \xi_{\text{even}}(r)}{r^{3/2}} + \xi_0(r)$$

$$\text{deltaK}_{T\text{odd}} := - \frac{2 \sqrt{M} \sqrt{2} \xi_{\text{odd}}(r)}{r^{3/2}}$$

$$\text{deltaK}_{\text{Trace}} := - \frac{\sqrt{M} \sqrt{2} (2 \xi_{\text{even}}(r) \lambda + r \xi_r(r))}{2 r^{3/2}} + \left( \frac{d}{dr} \xi_0(r) \right) r + \frac{\xi_0(r) \lambda}{2} + \frac{\xi_0(r) M}{r} \quad (107)$$

```

> expand(subs(xi__r(r)=Xi(r)*r^(3/4), op(2, deltaK__r)));

```

$$\frac{\sqrt{2} \sqrt{M} \left( \frac{d}{dr} \Xi(r) \right)}{r^{3/4}} \quad (108)$$

```

> factor(expand(subs(xi__even(r)=r^2*Xi(r), op(1, deltaK__Veven))))
;

```

$$\frac{\sqrt{M} \sqrt{2} \left( -2 r^2 \left( \frac{d}{dr} \Xi(r) \right) + \xi_r(r) \right)}{2 r^{3/2}} \quad (109)$$

```

> deltaK__Trace;

```

$$- \frac{\sqrt{M} \sqrt{2} (2 \xi_{\text{even}}(r) \lambda + r \xi_r(r))}{2 r^{3/2}} + \left( \frac{d}{dr} \xi_0(r) \right) r + \frac{\xi_0(r) \lambda}{2} + \frac{\xi_0(r) M}{r} \quad (110)$$

**Now in the RW gauge.**

Constraints

Hamiltonian Constraint

```

> HamilCon:=eval(subs(h__Teven(r)=0,h__Todd(r)=0,h__Veven(r)=0,

```

H0\_\_vac) ) ;

$$\begin{aligned} \text{HamilCon} := & \left( \frac{2M}{r^2} - \frac{\lambda}{2r} + \frac{1}{r} \right) h_{rr}(r) + \left( -\frac{2M}{r^4} - \frac{\lambda}{2r^3} - \frac{1}{r^3} \right) h_{trace}(r) \\ & - \frac{2\sqrt{2} k_{rr}(r) \sqrt{M}}{\sqrt{r}} - \frac{\sqrt{2} \sqrt{M} k_{trace}(r)}{r^{5/2}} + \frac{\frac{d}{dr} h_{trace}(r)}{r^2} - \frac{\frac{d^2}{dr^2} h_{trace}(r)}{r} + \frac{d}{dr} h_{rr}(r) \end{aligned} \quad (111)$$

Momentum Constraint

> MomentumConR:=eval(subs(h\_\_Teven(r)=0,h\_\_Todd(r)=0,h\_\_Veven(r)=0,H1\_\_vac)) ;

$$\begin{aligned} \text{MomentumConR} := & -\frac{k_{Veven}(r) \lambda}{2r} + \frac{d}{dr} k_{trace}(r) - k_{rr}(r) + \frac{\sqrt{2} \sqrt{M} h_{rr}(r)}{2r^{3/2}} \\ & + \frac{\sqrt{2} \sqrt{M} \left( \frac{d}{dr} h_{trace}(r) \right)}{4r^{5/2}} - \frac{\sqrt{2} \sqrt{M} h_{trace}(r)}{r^{7/2}} \end{aligned} \quad (112)$$

> MomentumConVEven:=eval(subs(h\_\_Teven(r)=0,h\_\_Todd(r)=0,h\_\_Veven(r)=0,k\_\_Teven(r)=0,H2\_\_even)) ;

MomentumConVOdd:=eval(subs(h\_\_Teven(r)=0,h\_\_Todd(r)=0,h\_\_Veven(r)=0,k\_\_Teven(r)=0,H2\_\_odd)) ;

$$\begin{aligned} \text{MomentumConVEven} := & -\frac{k_{trace}(r)}{r^2} - k_{rr}(r) + \frac{d}{dr} k_{Veven}(r) + \frac{2k_{Veven}(r)}{r} \\ & - \frac{\sqrt{2} \sqrt{M} h_{rr}(r)}{4r^{3/2}} - \frac{\sqrt{2} \sqrt{M} h_{trace}(r)}{r^{7/2}} \\ \text{MomentumConVOdd} := & \frac{k_{Todd}(r) \lambda}{2r^2} + \frac{k_{Todd}(r)}{r^2} + \frac{2k_{VOdd}(r)}{r} + \frac{d}{dr} k_{VOdd}(r) \\ & + \frac{\sqrt{2} \left( \frac{d}{dr} h_{VOdd}(r) \right) \sqrt{M}}{r^{3/2}} + \frac{\sqrt{2} h_{VOdd}(r) \sqrt{M}}{2r^{5/2}} \end{aligned} \quad (113)$$

Time derivative of h

> HdotR:=subs(n\_\_L(r)=-h\_\_rr(r)/2,expand(hdot\_\_C11)) ;

$$\text{HdotR} := -\frac{\sqrt{2} \sqrt{M} \left( \frac{d}{dr} h_{rr}(r) \right)}{\sqrt{r}} - \frac{\sqrt{2} \sqrt{M} h_{rr}(r)}{2r^{3/2}} + 2 \left( \frac{d}{dr} v_{scalar}(r) \right) + 2k_{rr}(r) \quad (114)$$

> HdotVEven:=eval(subs(h\_\_Teven(r)=0,h\_\_Todd(r)=0,h\_\_Veven(r)=0,k\_\_Teven(r)=0,v\_\_even(r)=0,v\_\_odd(r)=-k\_\_Todd(r)+sqrt(M)\*h\_\_Vodd(r)\*sqrt(2)\*(1/sqrt(r)),hdot\_\_CEven)) ;

HdotVOdd:=eval(subs(h\_\_Teven(r)=0,h\_\_Todd(r)=0,h\_\_Veven(r)=0,k\_\_Teven(r)=0,v\_\_even(r)=0,v\_\_odd(r)=-k\_\_Todd(r)+sqrt(M)\*h\_\_Vodd(r)\*sqrt(2)\*(1/sqrt(r)),hdot\_\_CVOdd)) ;

$$\text{HdotVEven} := -\frac{\sqrt{2} h_{rr}(r) \sqrt{M}}{\sqrt{r}} + v_{scalar}(r) + 2k_{Veven}(r)$$



$$HdotVOdd := 2 k_{Vodd}(r) - \left( \frac{d}{dr} k_{Todd}(r) \right) + \frac{3 \sqrt{2} h_{Vodd}(r) \sqrt{M}}{2 r^{3/2}} \quad (115)$$

$$+ \frac{\sqrt{2} \left( \frac{d}{dr} h_{Vodd}(r) \right) \sqrt{M}}{\sqrt{r}} - \frac{2 \left( -k_{Todd}(r) + \frac{\sqrt{2} h_{Vodd}(r) \sqrt{M}}{\sqrt{r}} \right)}{r}$$

```
> HdotTEven:=eval(subs(h__Teven(r)=0,h__Todd(r)=0,h__Veven(r)=0,
k__Teven(r)=0,v__even(r)=0,n__L(r)=-h__rr(r)/2,hdot__Teven));
HdotTOdd:=eval(subs(h__Teven(r)=0,h__Todd(r)=0,h__Veven(r)=0,
k__Teven(r)=0,v__even(r)=0,v__odd(r)=-k__Todd(r)+sqrt(M)*h__Vodd
(r)*sqrt(2)*(1/sqrt(r)),n__L(r)=-h__rr(r)/2,hdot__TOdd));
HdotTTrace:=eval(subs(h__Teven(r)=0,h__Todd(r)=0,h__Veven(r)=0,
k__Teven(r)=0,v__even(r)=0,n__L(r)=-h__rr(r)/2,hdot__TOmega));
HdotTEven := 0
HdotTOdd := 0
```

$$HdotTTrace := \left( \frac{\sqrt{2} \left( \frac{d}{dr} h_{trace}(r) \right)}{\sqrt{r}} - \sqrt{r} \sqrt{2} h_{rr}(r) \right) \sqrt{M} + 2 v_{scalar}(r) r + 2 k_{trace}(r) \quad (116)$$

Time derivative of K

```
> KdotR:=eval(subs(h__Teven(r)=0,h__Todd(r)=0,h__Veven(r)=0,
k__Teven(r)=0,v__even(r)=0,v__odd(r)=-k__Todd(r)+sqrt(M)*h__Vodd
(r)*sqrt(2)*(1/sqrt(r)),n__L(r)=-h__rr(r)/2,Kdot__rr));
```

$$KdotR := \left( \frac{3 h_{rr}(r)}{4 r^3} - \frac{2 h_{trace}(r)}{r^5} - \frac{2 \left( \frac{d}{dr} h_{rr}(r) \right)}{r^2} \right) M + \left( - \frac{\sqrt{2} k_{trace}(r)}{r^{7/2}} \right. \\ \left. + \frac{2 \sqrt{2} k_{rr}(r)}{r^{3/2}} + \frac{\sqrt{2} \left( \frac{d}{dr} k_{rr}(r) \right)}{\sqrt{r}} + \frac{\sqrt{2} \left( \frac{d}{dr} v_{scalar}(r) \right)}{r^{3/2}} - \frac{3 \sqrt{2} v_{scalar}(r)}{4 r^{5/2}} \right) \\ \sqrt{M} + \frac{h_{rr}(r) \lambda}{2 r^2} + \frac{2 h_{trace}(r)}{r^4} - \frac{\left( \frac{d^2}{dr^2} h_{rr}(r) \right)}{2} - \frac{\frac{d}{dr} h_{rr}(r)}{r} + \frac{\frac{d^2}{dr^2} h_{trace}(r)}{r^2} \\ - \frac{2 \left( \frac{d}{dr} h_{trace}(r) \right)}{r^3} \quad (117)$$

```
> KdotVEven:=eval(subs(h__Teven(r)=0,h__Todd(r)=0,h__Veven(r)=0,
k__Teven(r)=0,v__even(r)=0,v__odd(r)=-k__Todd(r)+sqrt(M)*h__Vodd
(r)*sqrt(2)*(1/sqrt(r)),n__L(r)=-h__rr(r)/2,Kdot__Veven));
```

$$KdotVEven := - \frac{M h_{rr}(r)}{r^2} + \left( \frac{\sqrt{2} \left( \frac{d}{dr} k_{Veven}(r) \right)}{\sqrt{r}} + \frac{\sqrt{2} v_{scalar}(r)}{2 r^{3/2}} - \sqrt{2} \sqrt{r} \left( \frac{d}{dr} \right. \right. \\ \left. \left. h_{even}(r) \right) \right) \sqrt{M} - \frac{r \left( \frac{d}{dr} h_{rr}(r) \right)}{2} + \frac{\left( \frac{d}{dr} h_{trace}(r) \right)}{2} - \frac{h_{rr}(r)}{2 r} \quad (118)$$

```
> KdotVOdd:=collect(eval(subs(h__Teven(r)=0,h__Todd(r)=0,h__Veven
```

```
(r)=0,k_Teven(r)=0,v_even(r)=0,v_odd(r)=-k_Todd(r)+sqrt(M)*
h_Vodd(r)*sqrt(2)*(1/sqrt(r)), n_L(r)=-h_rr(r)/2,Kdot_Vodd)),
diff);
```

$$KdotVOdd := \frac{\sqrt{2} \sqrt{M} \left( \frac{d}{dr} k_{Todd}(r) \right)}{r^{3/2}} + \frac{\sqrt{2} \sqrt{M} \left( \frac{d}{dr} k_{Vodd}(r) \right)}{\sqrt{r}} - \frac{3 M h_{Vodd}(r)}{r^3} \quad (119)$$

$$+ \left( \frac{h_{Vodd}(r) \sqrt{M}}{r^3} + \frac{2 \sqrt{2} \left( -k_{Todd}(r) + \frac{\sqrt{2} h_{Vodd}(r) \sqrt{M}}{\sqrt{r}} \right)}{r^{5/2}} \right) \sqrt{M} + \left( \frac{\lambda}{2 r^2} + \frac{1}{r^2} \right) h_{Vodd}(r)$$

```
> KdotTEven:=eval(subs(h_Teven(r)=0,h_Todd(r)=0,h_Veven(r)=0,
k_Teven(r)=0,v_even(r)=0,v_odd(r)=-k_Todd(r)+sqrt(M)*h_Vodd
(r)*sqrt(2)*(1/sqrt(r)), n_L(r)=-h_rr(r)/2,Kdot_Teven));
```

$$KdotTEven := 0 \quad (120)$$

```
> KdotTOdd:=expand(eval(subs(h_Teven(r)=0,h_Todd(r)=0,h_Veven(r)
=0,k_Teven(r)=0,v_even(r)=0,v_odd(r)=-k_Todd(r)+sqrt(M)*
h_Vodd(r)*sqrt(2)*(1/sqrt(r)), n_L(r)=-h_rr(r)/2,Kdot_Todd)))
```

$$KdotTOdd := -\frac{\sqrt{2} \sqrt{M} k_{Todd}(r)}{2 r^{3/2}} + \frac{\sqrt{2} \sqrt{M} \left( \frac{d}{dr} k_{Todd}(r) \right)}{\sqrt{r}} - \left( \frac{d}{dr} h_{Vodd}(r) \right) \quad (121)$$

```
> KdotTTrace:=eval(subs(h_Teven(r)=0,h_Todd(r)=0,h_Veven(r)=0,
k_Teven(r)=0,v_even(r)=0,v_odd(r)=-k_Todd(r)+sqrt(M)*h_Vodd
(r)*sqrt(2)*(1/sqrt(r)), n_L(r)=-h_rr(r)/2,Kdot_TOmega));
```

$$KdotTTrace := -\frac{M h_{rr}(r)}{2 r} + \left( \sqrt{r} \sqrt{2} k_{rr}(r) - \frac{\sqrt{2} k_{trace}(r)}{2 r^{3/2}} - \frac{\sqrt{2} v_{scalar}(r)}{2 \sqrt{r}} \right) \quad (122)$$

$$+ \frac{\sqrt{2} \left( \frac{d}{dr} k_{trace}(r) \right)}{\sqrt{r}} \sqrt{M} + \left( \frac{\lambda}{4} - 1 \right) h_{rr}(r) - \frac{h_{rr}(r) \lambda}{4} - r \left( \frac{d}{dr} h_{rr}(r) \right)$$

$$+ \frac{\left( \frac{d^2}{dr^2} h_{trace}(r) \right)}{2} + \frac{h_{trace}(r) \lambda}{2 r^2}$$

## Brill-Lindquist

```
> spacetime(Br-Lind, coord=[r, theta, phi], ds=PSI^4*(d[r]^2/(1-2*
M/r)+r^2*(d[theta]^2+sin(theta)^2*d[phi]^2));
```

Calculated ds for Br-Lind (0.000000 sec.)

CPU Time = 0.

For the  $\|(Br - Lind)\|$  spacetime:

Line element

$$ds^2 = \frac{PSI^4}{1 - \frac{2M}{r}} d r^2 + PSI^4 r^2 d \theta^2 + PSI^4 r^2 \sin(\theta)^2 d \phi^2 \quad (123)$$

```
> BLH__rr:=grcomponent(g(dn,dn),[r,r]);
BLH__tt:=grcomponent(g(dn,dn),[theta,theta]);
BLH__pp:=grcomponent(g(dn,dn),[phi,phi]);
```

$$\begin{aligned} BLH_{rr} &:= \frac{PSI^4}{1 - \frac{2M}{r}} \\ BLH_{tt} &:= PSI^4 r^2 \\ BLH_{pp} &:= PSI^4 r^2 \sin(\theta)^2 \end{aligned} \quad (124)$$

Constraints

```
> BLMomentumConR:=eval(subs(h__rr(r)=BLH__rr,h__trace(r)=BLH__pp,
MomentumConR));
```

$$\begin{aligned} BLMomentumConR &:= -\frac{k_{Veven}(r) \lambda}{2r} + \frac{d}{dr} k_{trace}(r) - k_{rr}(r) + \frac{\sqrt{2} \sqrt{M} PSI^4}{2 r^{3/2} \left(1 - \frac{2M}{r}\right)} \\ &\quad - \frac{\sqrt{2} \sqrt{M} PSI^4 \sin(\theta)^2}{2 r^{3/2}} \end{aligned} \quad (125)$$

```
> BLMomentumConVEven:=eval(subs(h__Veven(r)=0,h__Todd(r)=0,h__rr(r)
=BLH__rr,h__trace(r)=BLH__pp,MomentumConVEven));
BLMomentumConVOdd:=eval(subs(h__Vodd(r)=0,MomentumConVOdd));
```

$$\begin{aligned} BLMomentumConVEven &:= -\frac{k_{trace}(r)}{r^2} - k_{rr}(r) + \frac{d}{dr} k_{Veven}(r) + \frac{2 k_{Veven}(r)}{r} \\ &\quad - \frac{\sqrt{2} \sqrt{M} PSI^4}{4 r^{3/2} \left(1 - \frac{2M}{r}\right)} - \frac{\sqrt{2} \sqrt{M} PSI^4 \sin(\theta)^2}{r^{3/2}} \\ BLMomentumConVOdd &:= \frac{k_{Todd}(r) \lambda}{2 r^2} + \frac{k_{Todd}(r)}{r^2} + \frac{2 k_{Vodd}(r)}{r} + \frac{d}{dr} k_{Vodd}(r) \end{aligned} \quad (126)$$

Time derivative

```
> hdot__r:=expand(subs(h__rr(r)=BLH__rr,HdotR));
```

$$hdot_r := \frac{2 \sqrt{2} M^{3/2} PSI^4}{r^{5/2} \left(1 - \frac{2M}{r}\right)^2} - \frac{\sqrt{2} \sqrt{M} PSI^4}{2 r^{3/2} \left(1 - \frac{2M}{r}\right)} + 2 \left( \frac{d}{dr} v_{scalar}(r) \right) + 2 k_{rr}(r) \quad (127)$$

```
> hdotVEven:=subs(h__rr(r)=BLH__rr,HdotVEven);
hdotVOdd:=subs(h__Vodd(r)=0,HdotVOdd);
```

$$hdotVEven := -\frac{\sqrt{2} PSI^4 \sqrt{M}}{\sqrt{r} \left(1 - \frac{2M}{r}\right)} + v_{scalar}(r) + 2 k_{Veven}(r)$$

$$h\dot{V}Odd := 2 k_{Vodd}(r) - \left( \frac{d}{dr} k_{Todd}(r) \right) + \frac{\sqrt{2} \left( \frac{d}{dr} 0 \right) \sqrt{M}}{\sqrt{r}} + \frac{2 k_{Todd}(r)}{r} \quad (128)$$

```
> HdotTEven:=eval(subs(h__Teven(r)=0,h__Todd(r)=0,h__Veven(r)=0,
k__Teven(r)=0,v__even(r)=0,hdot__Teven));
HdotTOdd:=eval(subs(h__Teven(r)=0,h__Todd(r)=0,h__Veven(r)=0,
k__Teven(r)=0,v__even(r)=0,v__odd(r)=-k__Todd(r)+sqrt(M)*h__Vodd
(r)*sqrt(2)*(1/sqrt(r)),hdot__Todd));
hdotTTrace:=simplify(subs(h__rr(r)=BLH__rr, h__trace(r)=BLH__pp,
HdotTTrace));
```

$$HdotTEven := 0$$

$$HdotTOdd := 0$$

$$h\dot{T}Trace := \frac{1}{\sqrt{r} (-r + 2M)} \left( r^2 \sqrt{2} \sqrt{M} PSI^4 + 4 v_{scalar}(r) r^{3/2} M + 2 M^{3/2} \sqrt{2} \left( \frac{\partial}{\partial r} \right. \right. \quad (129)$$

$$\left. \left. (PSI^4 r^2 \sin(\theta)^2) \right) - 2 v_{scalar}(r) r^{5/2} - \sqrt{2} \sqrt{M} \left( \frac{\partial}{\partial r} (PSI^4 r^2 \sin(\theta)^2) \right) r \right.$$

$$\left. + 4 k_{trace}(r) \sqrt{r} M - 2 k_{trace}(r) r^{3/2} \right)$$

Time derivative of K

```
> BLKdotR:=eval(subs(h__rr(r)=BLH__rr, h__trace(r)=BLH__pp,
H__trace(r)=BLH__pp,KdotR));
```

$$BLKdotR := \left( \frac{3 PSI^4}{4 r^3 \left( 1 - \frac{2M}{r} \right)} - \frac{2 PSI^4 \sin(\theta)^2}{r^3} + \frac{4 PSI^4 M}{r^4 \left( 1 - \frac{2M}{r} \right)^2} \right) M + \left( \quad (130)$$

$$- \frac{\sqrt{2} k_{trace}(r)}{r^{7/2}} + \frac{2 \sqrt{2} k_{rr}(r)}{r^{3/2}} + \frac{\sqrt{2} \left( \frac{d}{dr} k_{rr}(r) \right)}{\sqrt{r}} + \frac{\sqrt{2} \left( \frac{d}{dr} v_{scalar}(r) \right)}{r^{3/2}} \right.$$

$$\left. - \frac{3 \sqrt{2} v_{scalar}(r)}{4 r^{5/2}} \right) \sqrt{M} + \frac{PSI^4 \lambda}{2 r^2 \left( 1 - \frac{2M}{r} \right)} - \frac{4 PSI^4 M^2}{\left( 1 - \frac{2M}{r} \right)^3 r^4}$$

```
> BLKdotVEven:=eval(subs(h__rr(r)=BLH__rr, h__trace(r)=BLH__pp,
H__trace(r)=BLH__pp, H__Teven=0, H__Veven=0,KdotVEven));
```

$$BLKdotVEven := - \frac{M PSI^4}{r^2 \left( 1 - \frac{2M}{r} \right)} + \left( \frac{\sqrt{2} \left( \frac{d}{dr} k_{Veven}(r) \right)}{\sqrt{r}} + \frac{\sqrt{2} v_{scalar}(r)}{2 r^{3/2}} \quad (131)$$

$$- \sqrt{2} \sqrt{r} \left( \frac{d}{dr} h_{even}(r) \right) \right) \sqrt{M} + \frac{PSI^4 M}{r \left( 1 - \frac{2M}{r} \right)^2} + PSI^4 r \sin(\theta)^2$$

$$- \frac{PSI^4}{2 r \left( 1 - \frac{2M}{r} \right)}$$

> BLKdotVOdd:=eval(subs(h\_\_rr(r)=BLH\_\_rr, h\_\_trace(r)=BLH\_\_pp,  
H\_\_trace(r)=BLH\_\_pp, H\_\_Teven=0, H\_\_Todd=0, H\_\_Veven=0, H\_\_Vodd=0,  
KdotVOdd));

$$BLKdotVOdd := \frac{\sqrt{2} \sqrt{M} \left( \frac{d}{dr} k_{Todd}(r) \right)}{r^{3/2}} + \frac{\sqrt{2} \sqrt{M} \left( \frac{d}{dr} k_{Vodd}(r) \right)}{\sqrt{r}} - \frac{3 M h_{Vodd}(r)}{r^3} \quad (132)$$

$$+ \left( \frac{h_{Vodd}(r) \sqrt{M}}{r^3} + \frac{2 \sqrt{2} \left( -k_{Todd}(r) + \frac{\sqrt{2} h_{Vodd}(r) \sqrt{M}}{\sqrt{r}} \right)}{r^{5/2}} \right) \sqrt{M} + \left( \frac{\lambda}{2 r^2} + \frac{1}{r^2} \right) h_{Vodd}(r)$$

> BLKdotTEven:=eval(subs(h\_\_rr(r)=BLH\_\_rr, h\_\_trace(r)=BLH\_\_pp,  
H\_\_trace(r)=BLH\_\_pp, H\_\_Teven=0, H\_\_Veven=0, KdotTEven));  
BLKdotTEven := 0

(133)

> BLKdotTOdd:=expand(eval(subs(h\_\_rr(r)=BLH\_\_rr, h\_\_trace(r)=  
BLH\_\_pp, H\_\_trace(r)=BLH\_\_pp, H\_\_Teven=0, H\_\_Veven=0, KdotTOdd)));

$$BLKdotTOdd := -\frac{\sqrt{2} \sqrt{M} k_{Todd}(r)}{2 r^{3/2}} + \frac{\sqrt{2} \sqrt{M} \left( \frac{d}{dr} k_{Todd}(r) \right)}{\sqrt{r}} - \left( \frac{d}{dr} h_{Vodd}(r) \right) \quad (134)$$

> BLKdotTTrace:=eval(subs(h\_\_rr(r)=BLH\_\_rr, h\_\_trace(r)=BLH\_\_pp,  
H\_\_trace(r)=BLH\_\_pp, H\_\_Teven=0, H\_\_Veven=0, KdotTTrace));

$$BLKdotTTrace := -\frac{M P S I^A}{2 r \left( 1 - \frac{2 M}{r} \right)} + \left( \sqrt{r} \sqrt{2} k_{rr}(r) - \frac{\sqrt{2} k_{trace}(r)}{2 r^{3/2}} - \frac{\sqrt{2} v_{scalar}(r)}{2 \sqrt{r}} \right) \quad (135)$$

$$+ \frac{\sqrt{2} \left( \frac{d}{dr} k_{trace}(r) \right)}{\sqrt{r}} \sqrt{M} + \frac{\left( \frac{\lambda}{4} - 1 \right) P S I^A}{1 - \frac{2 M}{r}} - \frac{P S I^A \lambda}{4 \left( 1 - \frac{2 M}{r} \right)} + \frac{2 P S I^A M}{r \left( 1 - \frac{2 M}{r} \right)^2}$$

$$+ P S I^A \sin(\theta)^2 + \frac{P S I^A \sin(\theta)^2 \lambda}{2}$$