example

June 18, 2021

0.1 GRTC Example Code

[1]:
```
from GRTC import *
from sympy import symbols, sin, init_printing

init_printing()

# Defining the symbols in the coordinate system
t, r, theta, phi = symbols('t, r, theta, phi')

# Defining some extra symbols
M, alpha = symbols('M, alpha')

#Defining the coordinate system
coord_sys = [t, r, theta, phi]

# Defining the diagonal components of the metric tensor
diag_comp = [-1, 1/alpha, M*alpha**2, M*sin(theta)**2]
```

1 Metric Tensor

[2]:
```
# Obtaining the metric tensor
mt = MetricTensor(diag_comp, coord_sys)
metric_tensor = mt.get_metrictensor()
metric_tensor
```

[2]:
$$\begin{bmatrix} -1 & 0 & 0 & 0 \\ 0 & \frac{1}{\alpha} & 0 & 0 \\ 0 & 0 & M\alpha^2 & 0 \\ 0 & 0 & 0 & M\sin^2(\theta) \end{bmatrix}$$

[3]:
```
# Default type of the metric tensor
mt.get_metrictensor_type()
```

[3]:
```
'dd'
```

[4]:
```
# Varying type 'dd' metric tensor to 'ud'
mt.vary_metrictensor_type(metric_tensor, 'ud')
```

[4]:
$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

[5]:
```
mt.get_metrictensor_type()
```

[5]:
```
'ud'
```

[6]:
```
# Varying type 'dd' metric tensor to 'uu'
mt.vary_metrictensor_type(metric_tensor, 'uu')
```

[6]:
$$\begin{bmatrix} -1 & 0 & 0 & 0 \\ 0 & \alpha & 0 & 0 \\ 0 & 0 & \frac{1}{M\alpha^2} & 0 \\ 0 & 0 & 0 & \frac{1}{M\sin^2(\theta)} \end{bmatrix}$$

[7]:
```
mt.get_metrictensor_type()
```

[7]:
```
'uu'
```

[8]:
```
# Obtaining the inverse of the metric tensor directly
mt.get_inverse()
```

[8]:
$$\begin{bmatrix} -1 & 0 & 0 & 0 \\ 0 & \alpha & 0 & 0 \\ 0 & 0 & \frac{1}{M\alpha^2} & 0 \\ 0 & 0 & 0 & \frac{1}{M\sin^2(\theta)} \end{bmatrix}$$

2 Christoffel Symbol

[9]:
```
# Obtaining the Christoffel Symbol
cs = ChristoffelSymbol(diag_comp, coord_sys)
chris_symbol = cs.get_christoffelsymbol()
chris_symbol
```

[9]:
$$\begin{bmatrix} \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} & \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} & \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -\frac{\sin(\theta)\cos(\theta)}{\alpha^2} \end{bmatrix} & \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{\cos(\theta)}{\sin(\theta)} \\ 0 & 0 & \frac{\cos(\theta)}{\sin(\theta)} & 0 \end{bmatrix} \end{bmatrix}$$

[10]:
```
# Default type of the Christoffel Symbol
cs.get_christoffelsymbol_type()
```

[10]:
```
'udd'
```

[11]:
```
# Varying type 'udd' Christoffel Symbol to 'ddd'
chris_symbol03 = cs.vary_christoffelsymbol_type(chris_symbol, 'ddd')
```

```
chris_symbol03
```

[11]:
$$\left[\begin{bmatrix}0&0&0&0\\0&0&0&0\\0&0&0&0\\0&0&0&0\end{bmatrix}\ \begin{bmatrix}0&0&0&0\\0&0&0&0\\0&0&0&0\\0&0&0&0\end{bmatrix}\ \begin{bmatrix}0&0&0&0\\0&0&0&0\\0&0&0&0\\0&0&0&-M\sin(\theta)\cos(\theta)\end{bmatrix}\ \begin{bmatrix}0&0&0&0\\0&0&0&0\\0&0&0&M\sin(\theta)\cos(\theta)\\0&0&M\sin(\theta)\cos(\theta)&0\end{bmatrix}\right]$$

[12]:
```
cs.get_christoffelsymbol_type()
```

[12]:
```
'ddd'
```

[13]:
```
# Obtaining the non-zero components of the given Christoffel Symbol for type ddd
cs.nonzero_christoffelsymbol(chris_symbol03)
```

$$\Gamma_{\theta\phi\phi} = -M\sin(\theta)\cos(\theta)$$

$$\Gamma_{\phi\theta\phi} = M\sin(\theta)\cos(\theta)$$

$$\Gamma_{\phi\phi\theta} = M\sin(\theta)\cos(\theta)$$

[14]:
```
# Varying type 'udd' Christoffel Symbol to 'uud'
chris_symbol21 = cs.vary_christoffelsymbol_type(chris_symbol, 'uud')
chris_symbol21
```

[14]:
$$\left[\begin{bmatrix}0&0&0&0\\0&0&0&0\\0&0&0&0\\0&0&0&0\end{bmatrix}\ \begin{bmatrix}0&0&0&0\\0&0&0&0\\0&0&0&0\\0&0&0&0\end{bmatrix}\ \begin{bmatrix}0&0&0&0\\0&0&0&0\\0&0&0&0\\0&0&0&-\frac{\cos(\theta)}{M\alpha^2\sin(\theta)}\end{bmatrix}\ \begin{bmatrix}0&0&0&0\\0&0&0&0\\0&0&0&\frac{\cos(\theta)}{M\alpha^2\sin(\theta)}\\0&0&\frac{\cos(\theta)}{M\sin^3(\theta)}&0\end{bmatrix}\right]$$

[15]:
```
cs.get_christoffelsymbol_type()
```

[15]:
```
'uud'
```

[16]:
```
# Varying type 'udd' Christoffel Symbol to 'uuu'
chris_symbol30 = cs.vary_christoffelsymbol_type(chris_symbol, 'uuu')
chris_symbol30
```

[16]:
$$\left[\begin{bmatrix}0&0&0&0\\0&0&0&0\\0&0&0&0\\0&0&0&0\end{bmatrix}\ \begin{bmatrix}0&0&0&0\\0&0&0&0\\0&0&0&0\\0&0&0&0\end{bmatrix}\ \begin{bmatrix}0&0&0&0\\0&0&0&0\\0&0&0&0\\0&0&0&-\frac{\cos(\theta)}{M^2\alpha^2\sin^3(\theta)}\end{bmatrix}\ \begin{bmatrix}0&0&0&0\\0&0&0&0\\0&0&0&\frac{\cos(\theta)}{M^2\alpha^2\sin^3(\theta)}\\0&0&\frac{\cos(\theta)}{M^2\alpha^2\sin^3(\theta)}&0\end{bmatrix}\right]$$

[17]:
```
cs.get_christoffelsymbol_type()
```

[17]:
```
'uuu'
```

[18]:
```
cs.nonzero_christoffelsymbol(chris_symbol30)
```

$$\Gamma^{\theta\phi\phi} = -\frac{\cos(\theta)}{M^2\alpha^2\sin^3(\theta)}$$

$$\Gamma^{\phi\theta\phi} = \frac{\cos(\theta)}{M^2\alpha^2\sin^3(\theta)}$$

$$\Gamma^{\phi\phi\theta} = \frac{\cos(\theta)}{M^2\alpha^2\sin^3(\theta)}$$

3 Riemann Tensor

[19]:
```
# Obtaining the Riemann Tensor
rt = RiemannTensor(diag_comp, coord_sys)
riemann_tensor = rt.get_riemanntensor()
riemann_tensor
```

[19]:
$$\begin{bmatrix}
\begin{bmatrix}0&0&0&0\\0&0&0&0\\0&0&0&0\\0&0&0&0\end{bmatrix} & \begin{bmatrix}0&0&0&0\\0&0&0&0\\0&0&0&0\\0&0&0&0\end{bmatrix} & \begin{bmatrix}0&0&0&0\\0&0&0&0\\0&0&0&0\\0&0&0&0\end{bmatrix} & \begin{bmatrix}0&0&0&0\\0&0&0&0\\0&0&0&0\\0&0&0&0\end{bmatrix} \\
\begin{bmatrix}0&0&0&0\\0&0&0&0\\0&0&0&0\\0&0&0&0\end{bmatrix} & \begin{bmatrix}0&0&0&0\\0&0&0&0\\0&0&0&0\\0&0&0&0\end{bmatrix} & \begin{bmatrix}0&0&0&0\\0&0&0&0\\0&0&0&0\\0&0&0&0\end{bmatrix} & \begin{bmatrix}0&0&0&0\\0&0&0&0\\0&0&0&0\\0&0&0&0\end{bmatrix} \\
\begin{bmatrix}0&0&0&0\\0&0&0&0\\0&0&0&0\\0&0&0&0\end{bmatrix} & \begin{bmatrix}0&0&0&0\\0&0&0&0\\0&0&0&0\\0&0&0&0\end{bmatrix} & \begin{bmatrix}0&0&0&0\\0&0&0&0\\0&0&0&0\\0&0&0&-\frac{\sin^2(\theta)}{\alpha^2}\end{bmatrix} & \begin{bmatrix}0&0&0&0\\0&0&0&0\\0&0&0&\frac{\sin^2(\theta)}{\alpha^2}\\0&0&0&0\end{bmatrix} \\
\begin{bmatrix}0&0&0&0\\0&0&0&0\\0&0&0&0\\0&0&0&0\end{bmatrix} & \begin{bmatrix}0&0&0&0\\0&0&0&0\\0&0&0&0\\0&0&0&0\end{bmatrix} & \begin{bmatrix}0&0&0&0\\0&0&0&0\\0&0&0&0\\0&0&1&0\end{bmatrix} & \begin{bmatrix}0&0&0&0\\0&0&0&0\\0&0&-1&0\\0&0&0&0\end{bmatrix}
\end{bmatrix}$$

[20]:
```
# Default type of the Riemann Tensor
rt.get_riemanntensor_type()
```

[20]:
```
'uddd'
```

[21]:
```
# Varying type 'uddd' Riemann Tensor to 'dddd'
riemann_tensor04 = rt.vary_riemanntensor_type(riemann_tensor, 'dddd')
riemann_tensor04
```

[21]:

$$\left[\begin{bmatrix}0&0&0&0\\0&0&0&0\\0&0&0&0\\0&0&0&0\end{bmatrix}\quad\begin{bmatrix}0&0&0&0\\0&0&0&0\\0&0&0&0\\0&0&0&0\end{bmatrix}\quad\begin{bmatrix}0&0&0&0\\0&0&0&0\\0&0&0&0\\0&0&0&0\end{bmatrix}\quad\begin{bmatrix}0&0&0&0\\0&0&0&0\\0&0&0&0\\0&0&0&0\end{bmatrix}\right.$$
$$\begin{bmatrix}0&0&0&0\\0&0&0&0\\0&0&0&0\\0&0&0&0\end{bmatrix}\quad\begin{bmatrix}0&0&0&0\\0&0&0&0\\0&0&0&0\\0&0&0&0\end{bmatrix}\quad\begin{bmatrix}0&0&0&0\\0&0&0&0\\0&0&0&0\\0&0&0&0\end{bmatrix}\quad\begin{bmatrix}0&0&0&0\\0&0&0&0\\0&0&0&0\\0&0&0&0\end{bmatrix}$$
$$\begin{bmatrix}0&0&0&0\\0&0&0&0\\0&0&0&0\\0&0&0&0\end{bmatrix}\quad\begin{bmatrix}0&0&0&0\\0&0&0&0\\0&0&0&0\\0&0&0&0\end{bmatrix}\quad\begin{bmatrix}0&0&0&0\\0&0&0&0\\0&0&0&0\\0&0&0&-M\sin^2(\theta)\end{bmatrix}\quad\begin{bmatrix}0&0&0&0\\0&0&0&0\\0&0&0&M\sin^2(\theta)\\0&0&0&0\end{bmatrix}$$
$$\left.\begin{bmatrix}0&0&0&0\\0&0&0&0\\0&0&0&0\\0&0&0&0\end{bmatrix}\quad\begin{bmatrix}0&0&0&0\\0&0&0&0\\0&0&0&0\\0&0&0&0\end{bmatrix}\quad\begin{bmatrix}0&0&0&0\\0&0&0&0\\0&0&0&0\\0&0&M\sin^2(\theta)&0\end{bmatrix}\quad\begin{bmatrix}0&0&0&0\\0&0&0&0\\0&0&-M\sin^2(\theta)&0\\0&0&0&0\end{bmatrix}\right]$$

[22]:
```
rt.get_riemanntensor_type()
```

[22]:
```
'dddd'
```

[23]:
```
rt.nonzero_riemanntensor(riemann_tensor04)
```

$R_{\theta\theta\phi\phi} = -M\sin^2(\theta)$

$R_{\theta\phi\theta\phi} = M\sin^2(\theta)$

$R_{\phi\theta\phi\theta} = M\sin^2(\theta)$

$R_{\phi\phi\theta\theta} = -M\sin^2(\theta)$

4 Ricci Tensor

[24]:
```
# Obtaining the Ricci Tensor
rit = RicciTensor(diag_comp, coord_sys)
ricci_tensor = rit.get_riccitensor()
ricci_tensor
```

[24]:
$$\begin{bmatrix}0&0&0&0\\0&0&0&0\\0&0&1&0\\0&0&0&\frac{\sin^2(\theta)}{\alpha^2}\end{bmatrix}$$

[25]:
```
# Default type of the Ricci Tensor
rit.get_riccitensor_type()
```

[25]:
```
'dd'
```

[26]:
```
# Varying type 'dd' Ricci Tensor to 'uu'
rit.vary_riccitensor_type(ricci_tensor, 'uu')
```

[26]:
$$\begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{M^2\alpha^4} & 0 \\ 0 & 0 & 0 & \frac{1}{M^2\alpha^2\sin^2(\theta)} \end{bmatrix}$$

[27]:
```
rit.get_riccitensor_type()
```

[27]:
```
'uu'
```

5 Ricci Scalar

[28]:
```
# Obtaining the Ricci Scalar
rs = RicciScalar(diag_comp, coord_sys)
ricci_scalar = rs.get_ricciscalar()
ricci_scalar
```

[28]:
$$\frac{2}{M\alpha^2}$$

6 Traceless Ricci Tensor

[29]:
```
# Obtaining the Traceless Ricci Tensor
trt = TracelessRicciTensor(diag_comp, coord_sys)
traceless_ricci_tensor = trt.get_trclss_riccitensor()
traceless_ricci_tensor
```

[29]:
$$\begin{bmatrix} \frac{1}{2M\alpha^2} & 0 & 0 & 0 \\ 0 & -\frac{1}{2M\alpha^3} & 0 & 0 \\ 0 & 0 & \frac{1}{2} & 0 \\ 0 & 0 & 0 & \frac{\sin^2(\theta)}{2\alpha^2} \end{bmatrix}$$

[30]:
```
# Default type of the Traceless Ricci Tensor
trt.get_trclss_riccitensor_type()
```

[30]:
```
'dd'
```

[31]:
```
# Varying type 'dd' Traceless Ricci Tensor to 'uu'
trt.vary_trclss_riccitensor_type(traceless_ricci_tensor, 'uu')
```

[31]:
$$\begin{bmatrix} \frac{1}{2M\alpha^2} & 0 & 0 & 0 \\ 0 & -\frac{1}{2M\alpha} & 0 & 0 \\ 0 & 0 & \frac{1}{2M^2\alpha^4} & 0 \\ 0 & 0 & 0 & \frac{1}{2M^2\alpha^2\sin^2(\theta)} \end{bmatrix}$$

[32]:
```
trt.get_trclss_riccitensor_type()
```

[32]:
```
'uu'
```

7 Weyl Tensor

[33]:
```
# Obtaining the Weyl Tensor
wyl = WeylTensor(diag_comp, coord_sys)
weyl_tensor = wyl.get_weyltensor()
weyl_tensor
```

[33]:
$$\left[\begin{bmatrix}0&0&0&0\\0&0&0&0\\0&0&0&0\\0&0&0&0\end{bmatrix}\quad\begin{bmatrix}0&-\frac{1}{3M\alpha^3}&0&0\\\frac{1}{3M\alpha^3}&0&0&0\\0&0&0&0\\0&0&0&0\end{bmatrix}\quad\begin{bmatrix}0&0&\frac{1}{6}&0\\0&0&0&0\\-\frac{1}{6}&0&0&0\\0&0&0&0\end{bmatrix}\quad\begin{bmatrix}0&0&0&\frac{\sin^2(\theta)}{6\alpha^2}\\0&0&0&0\\0&0&0&0\\-\frac{\sin^2(\theta)}{6\alpha^2}&0&0&0\end{bmatrix}\right.$$
$$\left[\begin{bmatrix}0&\frac{1}{3M\alpha^3}&0&0\\-\frac{1}{3M\alpha^3}&0&0&0\\0&0&0&0\\0&0&0&0\end{bmatrix}\quad\begin{bmatrix}0&0&0&0\\0&0&0&0\\0&0&0&0\\0&0&0&0\end{bmatrix}\quad\begin{bmatrix}0&0&0&0\\0&0&-\frac{1}{6\alpha}&0\\0&\frac{1}{6\alpha}&0&0\\0&0&0&0\end{bmatrix}\quad\begin{bmatrix}0&0&0&0\\0&0&0&-\frac{\sin^2(\theta)}{6\alpha^3}\\0&0&0&0\\0&\frac{\sin^2(\theta)}{6\alpha^3}&0&0\end{bmatrix}\right.$$
$$\left[\begin{bmatrix}0&0&-\frac{1}{6}&0\\0&0&0&0\\\frac{1}{6}&0&0&0\\0&0&0&0\end{bmatrix}\quad\begin{bmatrix}0&0&0&0\\0&0&\frac{1}{6\alpha}&0\\0&-\frac{1}{6\alpha}&0&0\\0&0&0&0\end{bmatrix}\quad\begin{bmatrix}0&0&0&0\\0&0&0&0\\0&0&0&0\\0&0&0&-M\sin^2(\theta)\end{bmatrix}\quad\begin{bmatrix}0&0&0&0\\0&0&0&0\\0&0&0&\frac{M\sin^2}{3}\\0&0&\frac{2M\sin^2(\theta)}{3}&0\end{bmatrix}\right.$$
$$\left[\begin{bmatrix}0&0&0&-\frac{\sin^2(\theta)}{6\alpha^2}\\0&0&0&0\\0&0&0&0\\\frac{\sin^2(\theta)}{6\alpha^2}&0&0&0\end{bmatrix}\quad\begin{bmatrix}0&0&0&0\\0&0&0&\frac{\sin^2(\theta)}{6\alpha^3}\\0&0&0&0\\0&-\frac{\sin^2(\theta)}{6\alpha^3}&0&0\end{bmatrix}\quad\begin{bmatrix}0&0&0&0\\0&0&0&0\\0&0&0&\frac{2M\sin^2(\theta)}{3}\\0&0&\frac{M\sin^2(\theta)}{3}&0\end{bmatrix}\quad\begin{bmatrix}0&0&0&0\\0&0&0&0\\0&0&-M\sin^2(\theta)&0\\0&0&0&0\end{bmatrix}\right.$$

[34]:
```
# Default type of the Weyl Tensor
wyl.get_weyltensor_type()
```

[34]:
```
'dddd'
```

[35]:
```
# Varying type 'dddd' Weyl Tensor to 'uuuu'
weyl_tensor40 = wyl.vary_weyltensor_type(weyl_tensor, 'uuuu')
weyl_tensor40
```

[35]:

$$\left[\begin{bmatrix}0&0&0&0\\0&0&0&0\\0&0&0&0\\0&0&0&0\end{bmatrix} \quad \begin{bmatrix}0&-\frac{1}{3M\alpha}&0&0\\\frac{1}{3M\alpha}&0&0&0\\0&0&0&0\\0&0&0&0\end{bmatrix} \quad \begin{bmatrix}0&0&\frac{1}{6M^2\alpha^4}&0\\0&0&0&0\\-\frac{1}{6M^2\alpha^4}&0&0&0\\0&0&0&0\end{bmatrix}\right.$$

$$\begin{bmatrix}0&\frac{1}{3M\alpha}&0&0\\-\frac{1}{3M\alpha}&0&0&0\\0&0&0&0\\0&0&0&0\end{bmatrix} \quad \begin{bmatrix}0&0&0&0\\0&0&0&0\\0&0&0&0\\0&0&0&0\end{bmatrix} \quad \begin{bmatrix}0&0&0&0\\0&0&-\frac{1}{6M^2\alpha^3}&0\\0&\frac{1}{6M^2\alpha^3}&0&0\\0&0&0&0\end{bmatrix}$$

$$\begin{bmatrix}0&0&-\frac{1}{6M^2\alpha^4}&0\\0&0&0&0\\\frac{1}{6M^2\alpha^4}&0&0&0\\0&0&0&0\end{bmatrix} \quad \begin{bmatrix}0&0&0&0\\0&0&\frac{1}{6M^2\alpha^3}&0\\0&-\frac{1}{6M^2\alpha^3}&0&0\\0&0&0&0\end{bmatrix} \quad \begin{bmatrix}0&0&0&0\\0&0&0&0\\0&0&0&0\\0&0&0&-\frac{1}{M^3\alpha^4\sin^2(\theta)}\end{bmatrix}$$

$$\left.\begin{bmatrix}0&0&0&-\frac{1}{6M^2\alpha^2\sin^2(\theta)}\\0&0&0&0\\0&0&0&0\\\frac{1}{6M^2\alpha^2\sin^2(\theta)}&0&0&0\end{bmatrix} \quad \begin{bmatrix}0&0&0&0\\0&0&0&\frac{1}{6M^2\alpha\sin^2(\theta)}\\0&0&0&0\\0&-\frac{1}{6M^2\alpha\sin^2(\theta)}&0&0\end{bmatrix} \quad \begin{bmatrix}0&0&0&0\\0&0&0&0\\0&0&0&\frac{2}{3M^3\alpha^4\sin^2(\theta)}\\0&0&\frac{1}{3M^3\alpha^4\sin^2(\theta)}&0\end{bmatrix}\right]$$

[36]:
```
wyl.get_weyltensor_type()
```

[36]: 'uuuu'

[37]:
```
wyl.nonzero_weyltensor(weyl_tensor40)
```

$$C^{trtr} = -\frac{1}{3M\alpha}$$

$$C^{trrt} = \frac{1}{3M\alpha}$$

$$C^{t\theta t\theta} = \frac{1}{6M^2\alpha^4}$$

$$C^{t\theta\theta t} = -\frac{1}{6M^2\alpha^4}$$

$$C^{t\phi t\phi} = \frac{1}{6M^2\alpha^2\sin^2(\theta)}$$

$$C^{t\phi\phi t} = -\frac{1}{6M^2\alpha^2\sin^2(\theta)}$$

$$C^{rttr} = \frac{1}{3M\alpha}$$

$$C^{rtrt} = -\frac{1}{3M\alpha}$$

$$C^{r\theta r\theta} = -\frac{1}{6M^2\alpha^3}$$

$$C^{r\theta\theta r} = \frac{1}{6M^2\alpha^3}$$

$$C^{r\phi r\phi} = -\frac{1}{6M^2\alpha\sin^2(\theta)}$$

$$C^{r\phi\phi r} = \frac{1}{6M^2\alpha\sin^2(\theta)}$$

$$C^{\theta tt\theta} = -\frac{1}{6M^2\alpha^4}$$

$$C^{\theta t\theta t} = \frac{1}{6M^2\alpha^4}$$

$$C^{\theta rr\theta} = \frac{1}{6M^2\alpha^3}$$

$$C^{\theta r \theta r} = -\frac{1}{6M^2\alpha^3}$$

$$C^{\theta\theta\phi\phi} = -\frac{1}{M^3\alpha^4\sin^2(\theta)}$$

$$C^{\theta\phi\theta\phi} = \frac{1}{3M^3\alpha^4\sin^2(\theta)}$$

$$C^{\theta\phi\phi\theta} = \frac{2}{3M^3\alpha^4\sin^2(\theta)}$$

$$C^{\phi t t\phi} = -\frac{1}{6M^2\alpha^2\sin^2(\theta)}$$

$$C^{\phi t\phi t} = \frac{1}{6M^2\alpha^2\sin^2(\theta)}$$

$$C^{\phi r r\phi} = \frac{1}{6M^2\alpha\sin^2(\theta)}$$

$$C^{\phi r\phi r} = -\frac{1}{6M^2\alpha\sin^2(\theta)}$$

$$C^{\phi\theta\theta\phi} = \frac{2}{3M^3\alpha^4\sin^2(\theta)}$$

$$C^{\phi\theta\phi\theta} = \frac{1}{3M^3\alpha^4\sin^2(\theta)}$$

$$C^{\phi\phi\theta\theta} = -\frac{1}{M^3\alpha^4\sin^2(\theta)}$$

8 Einstein Tensor

[38]:
```
# Obtaining the Einstein Tensor
et = EinsteinTensor(diag_comp, coord_sys)
einstein_tensor = et.get_einsteintensor()
einstein_tensor
```

[38]:
$$\begin{bmatrix} \frac{1}{M\alpha^2} & 0 & 0 & 0 \\ 0 & -\frac{1}{M\alpha^3} & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

[39]:
```
# Default type of the Einstein Tensor
et.get_einsteintensor_type()
```

[39]:
```
'dd'
```

[40]:
```
# Varying type 'dd' Einstein Tensor to 'uu'
et.vary_einsteintensor_type(einstein_tensor, 'uu')
```

[40]:
$$\begin{bmatrix} \frac{1}{M\alpha^2} & 0 & 0 & 0 \\ 0 & -\frac{1}{M\alpha} & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

[41]:
```
et.get_einsteintensor_type()
```

[41]:
```
'uu'
```

9 Kretschmann Scalar

[42]:
```
ks = KretschmannScalar(diag_comp, coord_sys)
kret_scalar = ks.get_kretschmannscalar()
kret_scalar
```

[42]:
$$\frac{4}{M^2\alpha^4}$$