# On Algebraic Computations of Electric and Magnetic Parts of the Weyl Tensor

A.H. Hasmani, B.N. Patel and Ravi Panchal

***Abstract*—Electromagnetic field theory is an established theory and the characteristic of electric and magnetic fields are well understood. Motivated by electromagnetic field theory, the Weyl tensor which represents pure gravitational field in general relativity, is decomposed into electric and magnetic part. The analysis of electric and magnetic parts of the Weyl tensor becomes important to understand gravitational field it represents. The computations in general relativity are known to be complicated as the tensors involved are up to rank-4; and they are obtained using second order partial derivatives of metric tensor and their combinations. This task can be handled efficiently by computers. The present paper describes a Mathematica program written for the computations of electric and magnetic parts of the Weyl Tensor.**

***Index Terms*—General relativity, electric and magnetic parts of the Weyl tensor, algebraic computations.**

***MSC 2010 Codes* – 83-04, 83-08, 83C99**

## I. Introduction

THE analogies between electromagnetic theory and gravitational theory are rich. Some of these analogies are developed, while some are still uncovered. One of the analogies is the Maxwell-like form of the gravitational field tensor (the Weyl tensor). Thus, in order to understand gravitational field we may carry out analysis of electric and magnetic parts of the Weyl tensor. However, these computations contain second order partial derivative of metric tensor, contractions and their combinations. This makes these computations difficult and there is a high risk of error. Computer aided computations ensure error free results and time consumed is very less when compared to manual computations. In view of these facts, Dautcourt and Jann [1] developed a REDUCE program for the algebraic computations of Christoffel symbols, Riemann tensor, Ricci tensor, Ricci scalar, etc. Campbell and Wainwright [2] have used symbolic manipulation language CAMAL to compute Weyl tensor and Ricci tensor relative to a complex null tetrad. d'Inverno [3] developed a computer system called ALAM for programming language LISP, which was specifically designed for carrying out computations in General Relativity. Mathematica is general computer algebra system (CAS) widely used for symbolic as well as numeric computations by researchers in many fields of science and engineering. Mathematica was used for the computation of Ricci tensor by Hasmani and Rathva [4]. Hasmani [5] has used Mathematica to compute complex scalars representing Weyl tensor in Newman-Penrose formalism. Hasmani and Andharia [6] have used Mathematica to carry out algebraic computation of spin coefficients in Newman-Penrose formalism. More recently, Hasmani and Panchal [7], [8] have developed two Mathematica programs, one for the algebraic computation of various general observer quantities and the other for the algebraic computation of tetrad components of Ricci tensor. It is worth noting that in Mathematica several general packages (e.g. xAct, GRTensor, etc.) are available for doing such computations. To use such packages one needs to learn about that package besides learning Mathematica. Our attempt is to give simple Mathematica program.

A.H. Hasmani, Department of Mathematics, Sardar Patel University, Vallabh Vidyanagar, India. (E-mail: ah_hasmani@spuvvn.edu)

B.N. Patel, Department of Applied Science and Humanities, G.H. Patel College of Engineering and Technology, Vallabh Vidyanagar, India. (E-mail: bhavyata_patel2007@yahoo.co.in).

Ravi Panchal, Department of Mathematics, Sardar Patel University, Vallabh Vidyanagar, India. (E-mail: ravipanchal1712@gmail.com)

The present paper describes a program written in the computer algebra system Mathematica, using classical tensor analysis techniques to compute electric and magnetic parts of the Weyl tensor. The program uses coordinate system, metric tensor and unit time-like velocity vector as input. It is worth noting that computation of tensorial components is very complicated, but the program discussed here can handle the situation smoothly.

Section II contains expressions of electric and magnetic parts of the Weyl tensor and their properties; the terminology and notations are standard and found in the relevant literature. In section III, Mathematica program for the computation of electric and magnetic parts of the Weyl tensor is discussed. Input for the Szekeres metric [9] has been given in Subsection III-A. In Subsection III-B, we have given explanation of necessary steps of the logic used in the program. Subsection III-C contains verification of vanishing properties of Weyl tensor, electric part and magnetic part of the Weyl tensor; this supports the correctness of the results obtained. Subsection (III-D) explains how to get output of the program and Subsection III-E shows output for the Szekeres metric.

Section IV contains examples of some space-times of interest to researchers. We have listed the metric expression, chosen unit time-like vector, output of Electric and Magnetic parts of the Weyl tensor generated using our program and time taken by the program for getting output.

## II. Electric and Magnetic parts of the Weyl Tensor

Weyl tensor $C_{ijkl}$ is the gravitational field tensor [10], defined by

$$C^{ij}{}_{kl} = R^{ij}{}_{kl} + \frac{1}{3}R\delta^i_{[k}\delta^j_{l]} - 2\delta^i_{[k}R^j_{l]},$$

where $R_{ijkl}$ - Riemann tensor, $R_{ij}$ - Ricci tensor, $R$ - Ricci scalar and $\delta_{ij}$ - Kronecker delta.

Now, electric and magnetic parts of the Weyl tensor with respect to a unit time-like vector $u^i$ (physically, a timelike vector field $u^i$ is often taken to be the 4-velocity of fluid) are defined by

$$E_{ik} = C_{ijkl} u^j u^l,$$
$$H_{ik} =^* C_{ijkl} u^j u^l,$$

where $^*C_{ijkl}$ represents dual of the Weyl tensor defined using completely antisymmetric Levi-Civita pseudo-tensor $\epsilon_{ijkl}$ given by

$$^*C_{ijkl} = \frac{1}{2}\eta_{ijmn} C^{mn}{}_{kl},$$

in which $\eta_{ijkl} = \sqrt{-g}\epsilon_{ijkl}$ and $g = \det g_{ij}$.

Both $E_{ik}$ and $H_{ik}$ are symmetric, orthogonal to unit time-like vector and traceless, i.e.

$$E_{ik} = E_{ki}, E_{ik}u^k = 0, E_{ik}g^{ik} = E^i{}_i = 0,$$
$$H_{ik} = H_{ki}, H_{ik}u^k = 0, H_{ik}g^{ik} = H^i{}_i = 0.$$

It is known from the classical electromagnetic theory that, electromagnetic field tensor consists of components of electric and magnetic field. In analogous way, in general relativity, the gravitational field tensor (Weyl tensor) can be decomposed into its electric and magnetic parts [11] as,

$$C_{ijkl} =(\eta_{ijpq}\eta_{klrs} + g_{ijpq}g_{klrs})u^p u^r E^{qs}$$
$$- (\eta_{ijpq}g_{klrs} + g_{ijpq}\eta_{klrs})u^p u^r H^{qs},$$

where $g_{ijkl} = g_{ik}g_{jl} - g_{il}g_{jk}$. The notations and terminologies used here are standard and can be found in any relevant literature.

## III. Mathematica Program

Mathematica is a general purpose computer algebra system; we have exploited its excellent capacity of handling matrices and differentials. This section shows how Mathematica can be used in computing electric and magnetic parts of the Weyl tensor. The present work is based on various commands of Mathematica and programming techniques described in [12]. Mathematica is a general CAS; and one learns only relevant commands and features. The program is presented in the following subsections in parts, this provides better understanding of the present program on one hand and provides clue for writing similar programs on the other hand.

In Section (III-A), we consider input for the Szekeres metric [9], which describes axially symmetric space-time

$$ds^2 = -e^{\lambda}dr^2 - e^{2\beta}(dy^2 + dz^2) + dt^2,$$

where $\lambda$ and $\beta$ are functions of all four space time coordinates $r$, $y$, $z$ and $t$. We choose unit time-like vector $u^i = \delta^i_4$.

### A. Input

The current program uses coordinate system, metric tensor and unit time-like vector as input and it should be changed according to space-time under consideration. The corresponding Mathematica input are described below.

The following will assign coordinate system to `coor`,

$$\texttt{coor} = \{\texttt{r},\texttt{y},\texttt{z},\texttt{t}\} \quad (1)$$

The following sets covariant components of metric tensor $g_{ij}$ to `metric`,

$$\begin{aligned}\texttt{metric} =&\{\{-\texttt{Exp}[\lambda[\texttt{r},\texttt{y},\texttt{z},\texttt{t}]],0,0,0\}\\ &,\{0,-\texttt{Exp}[2\beta[\texttt{r},\texttt{y},\texttt{z},\texttt{t}]],0,0\}\\ &,\{0,0,-\texttt{Exp}[2\beta[\texttt{r},\texttt{y},\texttt{z},\texttt{t}]],0\}\\ &,\{0,0,0,1\}\}\end{aligned}$$

Now, contravariant unit time-like vector $u^i$ is assigned to `npu` as follows,

$$\texttt{npu} =\{0,0,0,1\} \quad (2)$$

### B. Logic of the Program

The covariant components of the unit time-like vector $u_i$ (`npulow`) are computed below,

```
npulow=npu.metric;
```

The following will compute contravariant components of the metric tensor $g^{ij}$ (`gup`),

```
gup=Inverse[metric];
```

Now, Christoffel symbols of first kind $\Gamma_{ij,k}$ (`gama`) and second kind $\Gamma^h{}_{ij}$ (`gamaup`) [4]–[7] are computed as follows,

```
gama=Table[FullSimplify[
 (1/2)(D[metric[[i,k]],coor[[j]]]
 +D[metric[[j,k]],coor[[i]]]
 -D[metric[[i,j]],coor[[k]]])]
,{i,4},{j,4},{k,4}];
gamaup=Table[FullSimplify[
 Sum[gup[[h,k]]gama[[i,j,k]],{k,4}]]
,{h,4},{i,4},{j,4}];
```

The following gives covariant components of Riemann tensor $R_{hijk}$ (`riemannlowhijk`) [8],

```
riemannlowhijk=Table[(1/2)
(D[metric[[h,k]],coor[[i]],coor[[j]]]
+D[metric[[i,j]],coor[[h]],coor[[k]]]
-D[metric[[i,k]],coor[[h]],coor[[j]]]
-D[metric[[h,j]],coor[[i]],coor[[k]]])
+Sum[Sum[metric[[a,b]]
(gamaup[[a,i,j]]gamaup[[b,h,k]]
-gamaup[[a,i,k]]gamaup[[b,h,j]])
,{a,4}],{b,4}]
,{h,4},{i,4},{j,4},{k,4}];
```

Contraction of Riemann tensor with metric tensor gives the Ricci tensor $R_{ij}$ (`rij`) [4] as follows,

```
rij=Table[
 Sum[Sum[
  gup[[h1,j1]]
  riemannlowhijk[[h1,i,j1,k]]
 ,{h1,4}],{j1,4}]
,{i,4},{k,4}];
```

One more contraction of Ricci tensor with metric tensor will give rise to Ricci scalar $R$ and it is assigned to `ricci` [4],

```
ricci= Sum[Sum[
 gup[[i1,j1]]rij[[i1,j1]]
,{i1,4}],{j1,4}];
```

Using quantities computed above Weyl tensor $C_{hijk}$ (weyl) [5] is defined as follows,

```
weyl=Table[riemannlowhijk[[h,i,j,k]]
     -(1/2)(metric[[h,j]]rij[[i,k]]
            +metric[[i,k]]rij[[h,j]]
            -metric[[h,k]]rij[[i,j]]
            -metric[[i,j]]rij[[h,k]])
 +(ricci/6)(metric[[h,j]]metric[[i,k]]
 -metric[[h,k]]metric[[i,j]])
,{h,4},{i,4},{j,4},{k,4}];
```

Levi-Civita tensor $\eta_{ijkl}$ (levilow) [7] is now computed using a built in command in Mathematica,

```
eijkl=-Normal[LeviCivitaTensor[4]];
levilow=Sqrt[-Det[metric]] eijkl;
```

Dual of Weyl tensor $*C_{hijk}$ (cstarhijk) is computed now,

```
cstarhijk=Table[
Sum[Sum[Sum[Sum[
 (1/2)levilow[[h,i,r1,s1]]gup[[a1,r1]]
 gup[[b1,s1]]weyl[[a1,b1,j,k]]
,{r1,4}],{s1,4}],{a1,4}],{b1,4}]
,{h,4},{i,4},{j,4},{k,4}];
```

Following will compute electric part $E_{ik}$ (eik) and magnetic part $H_{ik}$ (hik) of the Weyl tensor,

```
eik=Table[FullSimplify[
 Sum[Sum[
  weyl[[i,j1,k,l1]]npu[[j1]]npu[[l1]]
 ,{j1,4}],{l1,4}]]
,{i,4},{k,4}];
hik=Table[FullSimplify[
 Sum[Sum[
  cstarhijk[[i,j1,k,l1]]
  npu[[j1]]npu[[l1]]
 ,{j1,4}],{l1,4}]]
,{i,4},{k,4}];
```

### C. Verifying Conditions

Many times the computations are not done correctly due to programming mistakes, so we will now use the trace-less property of Weyl tensor as verifying condition. If the trace of Weyl tensor $C^{j}{}_{ijk}$ (tr) does not vanish, the error message will be displayed and program will terminate.

```
tr=Table[FullSimplify[
 Sum[Sum[
    gup[[j1,h1]]weyl[[h1,i,j1,k]]
 ,{h1,4}],{j1,4}]]
,{i,4},{k,4}];
If[Boole[
tr===Table[0,{i,4},{j,4}]]===0
,Print["Trace of Weyl Tensor does
not Vanish"];Quit[]];
```

Now, vanishing properties of electric part and magnetic part namely, $E_{ik}u^k = 0$ and $H_{ik}u^k = 0$ respectively, are verified as follows. If they are not satisfied, the error message will be displayed and the program will terminate.

```
checkeik=Table[FullSimplify[
 Sum[eik[[i,k]]npu[[i]],{i,4}]]
,{k,4}];
checkhik=Table[FullSimplify[Sum[
 hik[[i,k]]npu[[i]],{i,4}]]
,{k,4}];
If[Boole[checkeik===checkhik
 ===Table[0,{i,4}]]===0
 ,Print["Vanishing Condition/s of
 Electric and/or Magnetic parts of
 the Weyl Tensor are not Satisfied"]
;Quit[]];
```

### D. Designing Output

The electric and magnetic parts of the Weyl tensor, being tensor of second rank, they have 16 components each, however, due to symmetry this number reduces to 10. The output is programmed so that only non-vanishing independent components are displayed.

```
If[Boole[
 eik===Table[0,{i,4},{j,4}]]===1
,Print["All Components of Electric
 Part Vanish"]
,For[i=1,i<=4,i++,For[k=i,k<=4,k++
,If[eik[[i,k]]=!=0
,Print["E",i,k,"=",eik[[i,k]]]]]]];
If[Boole[
 hik===Table[0,{i,4},{j,4}]]===1
,Print["All Components of Magnetic
 Part Vanish"]
,For[i=1,i<=4,i++,For[k=i,k<=4,k++
,If[hik[[i,k]]=!=0
,Print["H",i,k,"=",hik[[i,k]]]]]]];
```

### E. Output

Using input (1), (2) and (2) we get the following output, in which a prime denotes partial derivative with respect to $r$; an overhead dot denotes partial derivative with respect to $t$; the partial derivatives with respect to $y$ and $z$ are denoted by corresponding suffixes.

Non-vanishing independent components of electric part are:

$$
\begin{aligned}
E_{11} =& \frac{1}{24}e^{-2\beta}(e^{\lambda}(\lambda_z^2 - 8\beta_{zz} + 2\lambda_{zz} + \lambda_y^2 - 8\beta_{yy} + 2\lambda_{yy}) \\
&+ 2e^{2\beta}(e^{\lambda}(-2\dot{\beta}\dot{\lambda} + \dot{\lambda}^2 - 4\ddot{\beta} + 2\ddot{\lambda}) - 2\beta'\lambda' + 4\beta'')) \\
E_{12} =& \frac{1}{4}(-\lambda_y\beta' + 2\beta'_y) \\
E_{13} =& \frac{1}{4}(-\lambda_z\beta' + 2\beta'_z) \\
E_{22} =& \frac{1}{24}e^{-\lambda}(e^{\lambda}(6\beta_z\lambda_z - 2\lambda_z^2 + 4\beta_{zz} - 4\lambda_{zz} - 6\beta_y\lambda_y \\
&+ \lambda_y^2 + 4\beta_{yy} + 2\lambda_{yy}) \\
&+ e^{2\beta}(e^{\lambda}(2\dot{\beta}\dot{\lambda} - \dot{\lambda}^2 + 4\ddot{\beta} - 2\ddot{\lambda}) + 2\beta'\lambda' - 4\beta'')) \\
E_{23} =& \frac{1}{8}(-2\beta_z\lambda_y + \lambda_z(-2\beta_y + \lambda_y) + 2\lambda yz) \\
E_{33} =& \frac{1}{24}e^{-\lambda}(e^{\lambda}(-6\beta_z\lambda_z + \lambda_z^2 + 4\beta_{zz} + 2\lambda_{zz} + 6\beta_y\lambda_y \\
&- 2\lambda_y^2 + 4\beta_{yy} - 4\lambda_{yy}) \\
&+ e^{2\beta}(e^{\lambda}(2\dot{\beta}\dot{\lambda} - \dot{\lambda}^2 + 4\ddot{\beta} - 2\ddot{\lambda}) + 2\beta'\lambda' - 4\beta''))
\end{aligned}
$$

Non-vanishing independent components of magnetic part are:

$$
\begin{aligned}
H_{12} =& \frac{1}{8}e^{\lambda/2}((-2\dot{\beta} + \dot{\lambda})\lambda_z - 4\dot{\beta}_z + 2\dot{\lambda_z}) \\
H_{13} =& \frac{1}{8}e^{\lambda/2}((2\dot{\beta} - \dot{\lambda})\lambda_y + 4\dot{\beta}_y - 2\dot{\lambda_y})
\end{aligned}
$$

**Remark:**The output generated by Mathematica may not be user friendly as the output contains symbols defined by Mathematica in its own way; the output above is written in standard terminology.

## IV. Examples

Our program was tested for some useful metrics to compute electric and magnetic parts of the Weyl tensor. Here, we list time taken for the computation of electric and magnetic parts of the Weyl tensor and; our observation about the Weyl tensor for different metrics with chosen unit time-like vector. The following are some examples.

1) Gödel metric

$$ds^2 = dt^2 - dx^2 + 2e^{ax}dtdy + \frac{1}{2}e^{2ax}dy^2 - dz^2$$

Unit time-like vector: $u^i = \delta^i_4$
Non-vanishing independent components of electric part are:

$$E_{11} = -\frac{a^2}{6};\ E_{22} = -\frac{1}{12}e^{2ax}a^2;\ E_{33} = \frac{a^2}{3}$$

All Components of Magnetic Part Vanish.
Time taken: 0.45 seconds

2) Vaidya metric [14]

$$
\begin{aligned}
ds^2 =& - r^2d\theta^2 - r^2\sin^2\theta d\phi^2 + \left(1 - \frac{2m[u]}{r}\right)du^2 \\
&+ 2dudr
\end{aligned}
$$

Unit time-like vector: $u^i = \frac{1}{2\sqrt{2}}(1 + \frac{2m[u]}{r})\delta^i_1 + \frac{1}{\sqrt{2}}\delta^i_4$
Non-vanishing independent components of electric part are:

$$
\begin{aligned}
E_{11} =& \frac{m[u]}{r^3}; & E_{14} = -\frac{m[u](r + 2m[u])}{2r^4} \\
E_{22} =& -\frac{m[u]}{r}; & E_{33} = -\frac{m[u]\sin^2\theta}{r} \\
E_{44} =& \frac{m[u](r + 2m[u])^2}{4r^5}
\end{aligned}
$$

All Components of Magnetic Part Vanish.
Time taken: 0.64 seconds

3) Griffiths metric [13]

$$
\begin{aligned}
ds^2 =& 2dudv + 2a(2v - y)dudx + 2dudy \\
&+ 2a(2u - y)dvdx + 2dvdy \\
&+ \left(-\frac{3}{2} + 2a^2(2u - y)(2v - y) - 6a^2(u + v)^2\right)dx^2 \\
&- 2a(u + v + 2y)dxdy + \frac{1}{2}dy^2
\end{aligned}
$$

where a is a non-zero real constant
Unit time-like vector: $u^i = \frac{1}{\sqrt{2}}(\delta^i_1 + \delta^i_2)$
Non-vanishing independent components of electric part are:

$$
\begin{aligned}
E_{11} =& \frac{2}{9}(-27 + 2892u^2 + 6064uv + 3012v^2 \\
&- 524(u + v)y + 688y^2 \\
&- 72a(9u^2 - 2uv + 9v^2 - 8(u + v)y + y^2) \\
&+ 3a^2(-1 + 4(9u^2 - 2uv + 9v^2 \\
&- 8(u + v)y + y^2))) \\
E_{12} =& \frac{2}{9}(27 - 4(723u^2 + 1516uv + 753v^2) \\
&+ 524(u + v)y - 688y^2 \\
&+ 72a(9u^2 - 2uv + 9v^2 - 8(u + v)y \\
&+ y^2) + 3a^2(1 - 4(9u^2 - 2uv + 9v^2 \\
&- 8(u + v)y + y^2))) \\
E_{13} =& \frac{4}{9}(u - v)(-243 + a(6(9 + 644u^2) \\
&- 72a(2(9u^2 + 8uv + 9v^2) + (u + v)y + y^2) \\
&+ 3a^2(-1 + 72u^2 + 64uv + 72v^2 \\
&+ 4(u + v)y + 4y^2) + 8(v(1001u + 498v) \\
&+ 56(u + v)y + 86y^2))) \\
E_{14} =& 24(-3 + a)^2(u - v)(u + v + y) \\
E_{22} =& \frac{2}{9}(-27 + 2892u^2 + 6064uv + 3012v^2 \\
&- 524(u + v)y + 688y^2 \\
&- 72a(9u^2 - 2uv + 9v^2 - 8(u + v)y + y^2) \\
&+ 3a^2(-1 + 4(9u^2 - 2uv + 9v^2 \\
&- 8(u + v)y + y^2)))
\end{aligned}
$$

$$
\begin{aligned}
E_{23} =& -\frac{4}{9}(u-v)(-243+a(6(9+644u^2) \\
&-72a(2(9u^2+8uv+9v^2)+(u+v)y+y^2) \\
&+3a^2(-1+72u^2+64uv+72v^2+4(u+v)y \\
&+4y^2)+8(v(1001u+498v) \\
&+56(u+v)y+86y^2))) \\
E_{24} =& -24(-3+a)^2(u-v)(u+v+y) \\
E_{33} =& \frac{4}{9}(81-6(564u^2+1001uv+579v^2) \\
&-4a(3968u^4+27556u^3v+405v^2+4092v^4 \\
&+u^2(405+54796v^2)+u(270v+28052v^3)) \\
&-579(u+v)y-4a(u+v)(189+5546u^2 \\
&+15902uv+5732v^2)y-759y^2 \\
&-12a(-18+99u^2+802uv+99v^2)y^2 \\
&-5436a(u+v)y^3 \\
&-576a^3(u^2+uv+v^2)(9u^2-2u(v-5y) \\
&+(v+y)(9v+y))+24a^4(u^2+uv+v^2)(-1 \\
&+36u^2-8u(v-5y)+4(v+y)(9v+y)) \\
&+a^2(9+4(135u^2+2424u^4+72uv+2122u^3v \\
&+135v^2+1306u^2v^2+2062uv^3+2454v^4 \\
&+(u+v)(72+3685u^2+6700uv+3775v^2)y \\
&+(-9+616u^2+436uv+616v^2)y^2 \\
&+435(u+v)y^3))) \\
E_{34} =& \frac{4}{3}(8(-496+9(-3+a)^2a)u^3 \\
&-3(54-9a+a^3)v \\
&+12(-341+6(-3+a)^2a)v^3+27(-3+a)y \\
&+2(-1843+6a(202+13(-6+a)a))v^2y \\
&+2(-250+a(139+6(-6+a)a))vy^2 \\
&+2(-453+145a)y^3 \\
&+2u^2((-7826+60(-3+a)^2a)v \\
&+(-1781+6a(197+13(-6+a)a))y) \\
&+u(-2(81+7888v^2) \\
&+3a(9-a^2+40(-3+a)^2v^2) \\
&+4(-2718+a(1163+42(-6+a)a))vy \\
&+2(-250+a(139+6(-6+a)a))y^2)) \\
E_{44} =& \frac{2}{3}(-27-636u^2-2060uv+910(u+v)y \\
&-696v^2-182y^2-72a(3u^2+3v^2+10vy \\
&+y^2+10u(v+y))+3a^2(-1+4(3u^2+3v^2 \\
&+10vy+y^2+10u(v+y))))
\end{aligned}
$$

Non-vanishing independent components of magnetic part are:

$$
\begin{aligned}
H_{11} =& 24(-3+a)(u-v) \\
H_{12} =& -24(-3+a)(u-v) \\
H_{13} =& \frac{2}{9}(81(1-24u^2-24v^2+12(u+v)y) \\
&-4a(u+v)(81u+7936u^3+81v+31304u^2v \\
&+31552uv^2+8184v^3 \\
&+2(81+1578u^2+5030uv+1640v^2)y \\
&+1406(u+v)y^2+906y^3) \\
&+a^2(9+4(81u^2+1920u^4+54uv \\
&+10120u^3v+81v^2+16520u^2v^2 \\
&+10360uv^3+2040v^4 \\
&+(u+v)(27+20(65u^2+191uv+68v^2))y \\
&+750(u+v)^2y^2+290(u+v)y^3))) \\
H_{14} =& \frac{1}{9}(4(u+v)(-243+81a-7936u^2+1920au^2 \\
&+8(-2921+785a)uv+24(-341+85a)v^2) \\
&+16((-789+325a)u^2+5(-503+191a)uv \\
&+20(-41+17a)v^2)y \\
&+8(-703+375a)(u+v)y^2 \\
&+8(-453+145a)y^3) \\
H_{22} =& 24(-3+a)(u-v) \\
H_{23} =& \frac{2}{9}(81(-1+24u^2+24v^2-12(u+v)y) \\
&+4a(u+v)((u+v)(7936u^2+23368uv \\
&+81+8184v^2)+2(81+1578u^2+5030uv \\
&+1640v^2)y+1406(u+v)y^2+906y^3) \\
&+a^2(-9-4(1920u^4+10120u^3v \\
&+3v^2(27+680v^2)+2uv(27+5180v^2) \\
&+u^2(81+16520v^2)) \\
&-4(u+v)(27+20(65u^2+191uv+68v^2))y \\
&-3000(u+v)^2y^2-1160(u+v)y^3)) \\
H_{24} =& \frac{1}{9}(-4(u+v)(-243+81a-7936u^2+1920au^2 \\
&+8(-2921+785a)uv+24(-341+85a)v^2) \\
&-16((-789+325a)u^2+5(-503+191a)uv \\
&+20(-41+17a)v^2)y+8(453-145a)y^3 \\
&-8(-703+375a)(u+v)y^2) \\
H_{33} =& \frac{8}{3}(u-v)(81+a(3a^2+432a(u^2+uv+v^2) \\
&-2(888u^2+1433uv+903v^2) \\
&-85(u+v)y-145y^2)) \\
H_{34} =& \frac{8}{9}(u-v)((u+v)(-486+a(81-7936u^2 \\
&+3a(9+640u^2)+8(-2921+785a)uv \\
&+24(-341+85a)v^2)) \\
&+(-243+a(162-27a-3156u^2+1300au^2 \\
&+20(-503+191a)uv+80(-41+17a)v^2))y \\
&+2a(-703+375a)(u+v)y^2 \\
&+2a(-453+145a)y^3)
\end{aligned}
$$

Time taken: 883.21 seconds ($\approx$ 15 minutes)

## V. Conclusion

In General Relativity Weyl tensor represents gravitational field in vacuum. This tensor can be decomposed into two second rank tensors namely electric part and magnetic part of the Weyl tensor. The analysis of Weyl tensor is useful in understanding of gravitational field. The computing of Weyl tensor and its electric and magnetic parts are lengthy, time consuming and complicated.

In this paper, we have described a Mathematica program developed by us which computes electric and magnetic parts of the Weyl tensor for given metric and a unit time-like vector. As noted earlier there are many general packages for doing algebraic computations in general relativity. Use of such packages does not give user any information about the computations and for a user it may be difficult to do particular computations in the first attempt. We have discussed the parts of the program in detail which helps the reader in understanding how computations are done, accordingly one can prepare programs for individual use.

## Acknowledgment

The authors are thankful to the University Grants Commission, India for providing financial support under UGC-SAP-DRS(III) provided to the Department of Mathematics, Sardar Patel Universiry, Vallabh Vidyanagar, where the work was carried out. RP is also thankful to the UGC, India for providing financial support under UGC-BSR Fellowship(F.4-1/2006 (BSR)/7-159/2007(BSR)).

## References

[1] G. Dautcourt and K. P. Jann, “REDUCE programs for algebraic computation in general relativity”, *Astron. Nachr.*, vol. 304, no. 5, pp. 231-236, 1983. doi:10.1002/asna.2113040505

[2] S. J. Campbell and J. Wainwright, “Algebraic computing and the Newman-Penrose formalism in general relativity”, *General Relativity and Gravitation*, vol. 8, no. 12, pp. 987–1001, 1997.

[3] R. A. d’Inverno, “Algebraic computing in general relativity”, *General Relativity and Gravitation*, vol. 6, no. 6, pp. 567-593, 1975. doi:10.1007/BF00761964

[4] A. H. Hasmani,and G. A. Rathva, “Algebraic computations in general relativity using Mathematica”, *Prajna Jour. of Pure and Appl. Sci.*, vol. 15, pp. 111-117, 2007.

[5] A. H. Hasmani, “Algebraic computation of Newmann-Penrose scalars in general relativity using Mathematica”, *Jour. of Sci.*, vol. 1, pp. 82-83, 2010.

[6] A. H. Hasmani and P. I. Andharia, “Algebraic computation of spin co-efficients in Newmann-Penrose Formalism Using Mathematica”, *Jour. of Dyn. Sys. and Geom. Theor.*, vol. 9, no. 1, pp. 27-36, 2011. doi:10.1080/1726037X.2011.10698589

[7] A. H. Hasmani and R. R. Panchal, “Algebraic computations of general observer quantities using Mathematica”, *Astrophysics and Space Science*, vol. 359, no.1, pp. 1-5, 2015. doi:10.1007/s10509-015-24,65-6

[8] A. H. Hasmani and R. R. Panchal, “Algebraic computations of complex tetrad components of Ricci tensor”, *Proc. of Recent Trends in Mathematics and Statistics*, pp. 97-104, 2015.

[9] P. Szekeres, “A Class of Inhomogeneous Cosmological Models”, *Commun. Math. Phys.*, vol. 41, no.1, pp. 55-64, 1975. doi:10.1007/BF01608547

[10] Z. Ahsan, *Tensors: Mathematics of Differential Geometry and Relativity*, PHI Learning Pvt. Ltd. India, 2008.

[11] G. F. R. Ellis, “Republication of relativistic cosmology”, *Gen. Rel. and Cosmo.*, vol. 41, no. 3, pp. 581-660, 2009. doi:10.1007/s10714-009-0760-7

[12] S. Wolfram, *The Mathematica Book*, Fourth Edition, Camb. Uni. Press, Cambridge, 1999.

[13] J. B. Griffiths, “On the superposition of gravitational waves”, *Math. Proc. of the Camb. Phil. Soc.*, vol. 77, no. 3, pp. 559-565, 1975. doi:10.1017/s0305004100051380

[14] P. C. Vaidya, “The gravitational field of a radiating star”, *Proc. of the Indian Aca. of Sci.*, vol. 33, no. 5, pp. 264-276, 1951. doi:10.1007/bf03173260