Journal of Hyperstructures 4 (1) (2015), 57-68.
ISSN: 2322-1666 print/2251-8436 online

THE USE OF MATLAB PLATFORM TO COMPTE SOME GEOMETRIC QUANTITIES OF 3-MANIFOLD SOL_3

NEMAT ABAZARI AND MASOUD SAHRAIY

ABSTRACT. In this paper we use MATLAB platform [7, 8] to calculate some quantities of 3-manifold sol_3 like Christoffel symbols, curvatures, Einstein tensor and plot the geodesics of this space.

Key Words: 3-Manifold sol_3, Geometric quantities, Computer algebra.
2010 Mathematics Subject Classification: Primary: 13A15; Secondary: 13F30, 13G05.

1. Introduction

The space sol_3 is a simply connected homogeneous 3-dimensional manifold whose isometry group has dimension 3 and it is one of the eight models of geometry of Thurston [4, 5, 6]. The space sol_3 can be viewed as $\mathbb{R}^3$ with this metric

$$ds^2 = e^{2z}dx^2 + e^{-2z}dy^2 + dz^2.$$

In this paper we calculate some geometric quantities like Christoffel symbols, Riemannian curvature tensor, Ricci tensor, scalar curvature and Einstein tensor in sol_3 space by building MATLAB files [7, 8] that compute all of these quantities. For example in 2.5 we introduce MZSOL6 function in MATLAB. This function is able to calculate Einstein tensor in sol_3 space and each of G_{jk}'s can be obtain by calling this function. In 2.6 we construct the geodesic equations and solve these equations

Received: 12 July 2015, Accepted: 05 August 2015. Communicated by Ali Taghavi Jelodar;
∗Address correspondence to N. Abazari; E-mail: nematabazari@gmail.com

by MATLAB to plot the geodesics in sol_3 space. MATLAB files are included in appendix at the end of the paper.

2. Calculate some geometric quantities and plot the geodesics of 3-manifold sol_3

This metric in 3-manifold sol_3 is the form [5, 6]

$$ds^2 = e^{2z}dx^2 + e^{-2z}dy^2 + dz^2.$$

So the metric coefficients and the invers of matrix coefficients are as follows :

$$g_{ij} = \begin{bmatrix} e^{2z} & 0 & 0 \\ 0 & e^{-2z} & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad g^{ij} = \begin{bmatrix} e^{-2z} & 0 & 0 \\ 0 & e^{2z} & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

2. 1. Christoffel Symbols

The Christoffel symbols of first kind are defined by [3]:

$$\Gamma_{ij}^{m} = \frac{1}{2}g^{km}\left(-\frac{\partial g_{ij}}{\partial x_k} + \frac{\partial g_{ik}}{\partial x_j} + \frac{\partial g_{jk}}{\partial x_i}\right).$$

So in sol_3 space we design function file MZSOL1 to compute Christoffel symbols. The function file MZSOL1 has 3 arguments. The first two arguments are lower indexes in Γ_{ij}^{m} and the third argument is upper index in Γ_{ij}^{m} . for example to compute Γ_{11}^{3} it is enough to call MZSOL1 with (1,1,3) or type MZSOL1 (1,1,3) in MATLAB , so we have:

≫ MZsol1(1,1,3)

The christoffel symbol(gama _11^3) is:

ans =

-exp(2*z)

So we will have this equality: Γ_{ij}^{m}=MZSOL1(i,j,m)

By this process we have obtained Christoffel symbols. The nonzero components are the following:

$$\Gamma_{13}^{1} = 1, \Gamma_{31}^{1} = 1, \Gamma_{23}^{2} = -1$$

$$\Gamma_{32}^{2} = -1, \Gamma_{11}^{3} = -e^{2z}, \Gamma_{22}^{3} = e^{-2z}.$$

2. 2. Riemannian Curvature Tensor

The Riemannian curvature tensor is given by [3]:

$$R_{ijk}^{s} = \frac{\partial \Gamma_{ij}^{s}}{\partial x^k} - \frac{\partial \Gamma_{ik}^{s}}{\partial x^j} + \Gamma_{ij}^{r}\Gamma_{rk}^{s} - \Gamma_{ik}^{r}\Gamma_{rj}^{s}, R_{ijkl} = g_{si}R_{jkl}^{s}.$$

It is clear that to compute Riemannian curvature we should calculate Christoffel symbols and metric coefficient. We have metric coefficient and obtained Christoffel symbols in 2.1. To compute R^i_{jk} we construct MZSOL2 in sol_3 space. MZSOL2 is a function with 4 arguments .The first 3 arguments are lower indexes and the forth argument is upper indexes of R^s_{ijk} for example to compute R^1_{313} it is enough to call MZSOL2 in MATLAB or type $MZsol2(3,1,3,1)$ in MATLAB, we have:
$\gg$ MZsol2(3,1,3,1)
The curvature (R_313^1) is:
ans = *-1*

So we have:

$$R^s_{ijk} = MZsol2(i,j,k,s)$$

With this method we have obtained R^s_{ijk} in sol_3 space that nonzero components are follows:

$$R^2_{112} = -e^{2z}, \quad R^3_{113} = e^{2z}, \quad R^2_{121} = e^{2z},$$
$$R^3_{131} = -e^{2z}, \quad R^2_{212} = e^{-2z}, \quad R^1_{221} = -e^{-2z},$$
$$R^3_{223} = e^{-2z}, \quad R^2_{232} = -e^{-2z}, \quad R^1_{313} = -1,$$
$$R^2_{323} = -1, \quad R^1_{331} = 1, \quad R^2_{332} = 1.$$

Also to calculate R_{ijkl} in sol_3 space we construct MZSOL3 with 4 arguments. These arguments are R_{ijkl} indexes. For example to calculate R_{2332} it is enough to call MZSOL3 with (2, 3, 3, 2) or type $MZsol3(2,3,3,2)$ In MATLAB and we have:
$\gg$ MZsol3(2,3,3,2)
The curvature(R_2332) is:
ans =
*1/exp(2*z)*
So we will have this equality:$R_{ijkl} = MZSOL3\,(i,j,k,l)\,.$
By this process we have obtained R_{ijkl} in sol_3 space that nonzero components are the following:

$$R_{1212} = 1, \quad R_{1221} = -1, \quad R_{1313} = -e^{2z},$$
$$R_{1331} = e^{2z}, \quad R_{2112} = -1, \quad R_{2121} = 1,$$
$$R_{2323} = -e^{-2z}, \quad R_{2332} = e^{-2z}, \quad R_{3113} = e^{-2z},$$
$$R_{3131} = -e^{2z}, \quad R_{3223} = e^{-2z}, \quad R_{3232} = -e^{-2z}.$$

Also with this method we can see that these $R'_{ijkl}s$ satisfying the following equalities:

$$R_{ijkl} = -R_{jikl} = -R_{ijlk} = R_{klji}.$$

2. 3. Ricci Tensor

From [2] we know that the components of Ricci curvature tensor are defined by

$$R_{jk} = R^i_{jkl}.$$

To compute Ricci tensor in sol_3 space we construct function file MZSOL4 with two arguments and these arguments are R_{jk} indexes. For example to compute R_{33} it is enough to call MZSOL4 with arguments (3, 3) or type the $MZsol4$ (3, 3) and we have:

≫ MZsol4(3,3(

The Ricci tensor (R_33) is:

ans =

-2

So we will have:

$$R_{jk} = MZSOL4\,(j, k)\,,$$

Finally the Ricci tensor is the following:

$$R_{ij} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & -2 \end{bmatrix}.$$

2. 4. Scalar Curvature

The scalar curvature is defined by [1]:

$$S = R_{jk} g^{kj}.$$

To compute scalar curvature in sol_3 space we construct MATLAB script MZSOL5 and it is enough to run MZSOL5 to obtain scalar curvature. So we have:

≫ MZsol5

The scalar curvature is:

-2

We can see that the scalar curvature in sol_3 space is equal with -2.

2. 5. Einstein Tensor

The Einstein tensor is defined by [1]:

$$G_{jk} = R_{jk} - \frac{s}{2}\, g_{jk}.$$

In this equation the R_{jk}'s are Ricci tensor that we calculated in 2.4 and $_{jk}$'s are metric coefficients. To compute Einstein tensor in sol_3 space we construct MZSOL6 with two arguments and these arguments are $G'_{jk}{}^{s}$ Indexes. For example to compute G_{11} It is enough to call MZSOL6 with (1, 1) argument or type MZSOL6(1,1) in MATLAB and we have:

≫ MZsol6(1,1)

The Einstein tensor (G_ 11) is:
ans =
exp(2*z)
So we will have:

$$G_{jk} = MZSOL6(j,k)$$

By this process we have obtained $G_{jk}^{\prime s}$ that nonzero components are the following:

$$G_{11} = e^{2z}, \quad G_{22} = e^{-2z}, \quad G_{33} = -1.$$

So the Einstein tensor in sol_3 space is:

$$G_{jk} = \begin{bmatrix} e^{2z} & 0 & 0 \\ 0 & e^{-2z} & 0 \\ 0 & 0 & -1 \end{bmatrix}.$$

2. 6. Geodesic

Definition: A parameterized curve $\gamma : I \to M$ is a geodesic at $t_0 \in I$ if $\frac{D}{dt}\left(\frac{d\ \gamma}{dt}\right) = 0$ at the point t_0 . If γ is a geodesic at, for all $t \in I$, we say that γ is a geodesic.

Also we know that in local coordinates the curve $\gamma(t) = (x_1(t)\ ,\ \ldots\ ,\ x_n(t))$ is a geodesic if and only if it satisfies in geodesic equation:

$$\frac{d^2x_k}{dt^2} + \sum_{i,j} \Gamma_{ij}^k \frac{dx_i}{dt}\frac{dx_j}{dt} = 0.$$

So the geodesic equations of sol_3 space are:

$$\frac{d^2x(t)}{dt} + 2\frac{d\ x(t)}{dt}\frac{d\ z(t)}{dt} = 0,$$

$$\frac{d^2y(t)}{dt} - 2\frac{d\ y(t)}{dt}\frac{d\ z(t)}{dt} = 0,$$

$$\frac{d^2z(t)}{dt} - e^{2z(t)}\frac{dx(t)}{dt}\frac{dx(t)}{dt} + e^{-2z(t)}\frac{dy(t)}{dt}\frac{dy(t)}{dt} = 0.$$

To solve the geodesic equations we use numerical solution of ODEs in MATLAB. First we build the function file MZSOL7 containing the geodesic equations and then we build the MATLAB script MZSOL8 with ODE45 to solve the geodesic equations and plot the result.

These pictures are geodesics of sol_3 space. The first two pictures are side view of geodesics in 3-dimensional and the picture in second row is geodesic of sol_3 space in 2-dimensional XY.

3. Conclusion

In this paper we illustrated the possibility of the study of geodesic motion and geometric objects for sol_3 space using the graphical facilities of computer platform. This method by MATLAB can be used to compute many quantities in geometry and other sciences.

4. Appendix (M-Files)

MZsol1:

```
function gama_nm_k =MZsol1(n,m,k)
%Description: compute the christoffel symbol of sol3 space
if ((n>3) |(n<1)) |((m>3) |(m<1)) |((k>3) |(k<1))
disp(' error : notice that 1 <= n,m,k <= 3 because dimension is 3 .' )
elseif n<=3 && m<=3 && k<=3
syms x y z;
g=[exp(2*z) 0 0;0 exp(-2*z) 0;0 0 1];
G=inv(g);X=[x y z];
gama_nm_k=0;
for s=1:3
W=1/2*G(s,k)*(diff(g(m,s),X(n))+diff(g(s,n),X(m))-diff(g(n,m),X(s)));
gama_nm_k=W+gama_nm_k;
end
v1=' the christoffel symbol';
v2='gama _';v3=' is :';
disp([v1 '(' v2 num2str(n) num2str(m) '^' num2str(k) ')' v3]);
sum(gama_nm_k);
end
```

MZsol2:

```
function R_ABC_D =MZsol2(A,B,C,D)
%Description: compute the curvature(R_ikj^s) of sol3 space
x=sym('x','real');y=sym('y','real');
z=sym('z','real');X=[x y z];
L_1=diff(gama(A,C,D),X(B))-diff(gama(A,B,D),X(C));
L_3=0;R_ABC_D=0;
for S=1:3
L_2=((gama(A,C,S))*(gama(S,B,D)))-((gama(A,B,S))*(gama(S,C,D)));
L_3=L_3+L_2;
```

```
end
R_ABC_D=L_1+L_3+R_ABC_D;
function gama_nm_k = gama(n,m,k)
%compute christoffel symbols
g=[exp(2*z) 0 0;0 exp(-2*z) 0;0 0 1];
G=inv(g);X=[x y z];
gama_nm_k=0;
for s=1:3
q=1/2*G(s,k)*(diff(g(m,s),X(n))+diff(g(s,n),X(m))-diff(g(n,m),X(s)));
gama_nm_k=q+gama_nm_k;
end
sum(gama_nm_k);
end
w1=’ the curvature’;w2=’ R_’;w3=’ is : ’;
disp([w1 ’(’ w2 num2str(A) num2str(B) num2str(C) ’ˆ’ num2str(D) ’)’
w3]);
sum(R_ABC_D);
end
```

MZsol3:

```
function R_aABC =MZsol3(a,A,B,C)
%Description: compute the curvature (R_ikjs) of sol3 space
x=sym(’x’,’real’);y=sym(’y’,’real’);z=sym(’z’,’real’);
g=[exp(2*z) 0 0;0 exp(-2*z) 0;0 0 1];
R_aABC=0;
for w=1:3
F=g(a,w).*(masoud102(A,B,C,w));
R_aABC= R_aABC+F;
end
function R_ABC_D = masoud102(A,B,C,D)
%compute R_ijkˆs of sol3 space.
X=[x y z];
L_1=diff(gama(A,C,D),X(B))-diff(gama(A,B,D),X(C));
L_3=0;R_ABC_D=0;
for S=1:3
L_2=((gama(A,C,S))*(gama(S,B,D)))-((gama(A,B,S))*(gama(S,C,D)));
L_3=L_3+L_2;
end
```

```
R_ABC_D=L_1+L_3+R_ABC_D;
function gama_nm_k = gama(n,m,k)
%compute christoffel symbols.
g=[exp(2*z) 0 0;0 exp(-2*z) 0;0 0 1];
G=inv(g);X=[x y z];gama_nm_k=0;
for s=1:3
q=1/2*G(s,k)*(diff(g(m,s),X(n))+diff(g(s,n),X(m))-diff(g(n,m),X(s)));
gama_nm_k=q+gama_nm_k;
end
sum(gama_nm_k);
end
sum(R_ABC_D);
end
p1=' the curvature';p2=' R_';p3=' is : ';
disp([p1 '(' p2 num2str(a) num2str(A) num2str(B) num2str(C) ')' p3]);
sum( R_aABC);
end
```

MZsol4:

```
function ricci =MZsol4(u,v)
%Description: compute the ricci tensor of sol3 space
x=sym('x','real');y=sym('y','real');z=sym('z','real');
R_uv=0;
for o=1:3
t=sum(masoud102(u,o,v,o));
R_uv=t+R_uv;
end
function R_ABC_D = masoud102(A,B,C,D)
%compute of R_ikj^s of sol3 space.
X=[x y z];
L_1=diff(gama(A,C,D),X(B))-diff(gama(A,B,D),X(C));
L_3=0;R_ABC_D=0;
for S=1:3
L_2=((gama(A,C,S))*(gama(S,B,D)))-((gama(A,B,S))*(gama(S,C,D)));
L_3=L_3+L_2;
end
R_ABC_D=L_1+L_3+R_ABC_D;
function gama_nm_k = gama(n,m,k)
```

```
%compute christoffel symbols.
g=[exp(2*z) 0 0;0 exp(-2*z) 0;0 0 1];
G=inv(g);X=[x y z];
gama_nm_k=0;
for s=1:3
E=1/2*G(s,k)*(diff(g(m,s),X(n))+diff(g(s,n),X(m))-diff(g(n,m),X(s)));
gama_nm_k=E+gama_nm_k;
end
sum(gama_nm_k);
end
end
h1=' the ricci tensor';h2=' R_';h3=' is :';
disp([h1 '(' h2 num2str(u) num2str(v) ')' h3]);
sum(R_uv)
end
```

MZsol5:

```
%Description: compute scalar curvature of sol3 space
clear all;clc;
x=sym('x','real');y=sym('y','real');z=sym('z','real');
g=[exp(2*z) 0 0;0 exp(-2*z) 0;0 0 1];
G=inv(g);X=[x y z];
R=[0 0 0;0 0 0;0 0 -2];
S=0;
for j=1:3
for k=1:3
S1=(R(j,k))*(G(k,j));
S=S+S1;
end
end
t1=' the scalar curvature is : ';
disp(t1)
disp(S)
```

MZsol6:

```
function G_ab =MZsol6(j,k)
%Description: compute the Einstein tensor of sol3 space
```

```
S=-2;G_ab=0;
Q=masricc(j,k)-((-2/2)*g(j,k));
G_ab=Q+G_ab;
function R_uv=masricc(u,v )
% compute of ricci tensor (R_jk)
x=sym('x','real');y=sym('y','real');z=sym('z','real');
R_uv=0;
for o=1:3
t=sum(masoud102(u,o,v,o));
R_uv=t+R_uv;
end
function R_ABC_D = masoud102(A,B,C,D)
%compute of R_ikj^s of sol3 space.
X=[x y z];
L_1=diff(gama(A,C,D),X(B))-diff(gama(A,B,D),X(C));
L_3=0;R_ABC_D=0;
for p=1:3
L_2=((gama(A,C,p))*(gama(p,B,D)))-((gama(A,B,p))*(gama(p,C,D)));
L_3=L_3+L_2;
end
R_ABC_D=L_1+L_3+R_ABC_D;
function gama_nm_k = gama(n,m,k)
%compute christoffel symbols.
g=[exp(2*z) 0 0;0 exp(-2*z) 0;0 0 1];
G=inv(g);X=[x y z];
gama_nm_k=0;
for s=1:3
E=1/2*G(s,k)*(diff(g(m,s),X(n))+diff(g(s,n),X(m))-diff(g(n,m),X(s)));
gama_nm_k=E+gama_nm_k;
end
sum(gama_nm_k);
end
end
sum(R_uv);
end
w1=' the einstein tensor ';w2=' G_ ';w3=' is :';
disp([w1 '(' w2 num2str(j) num2str(k) ')' w3]);
end
```

MZsol7:

```
function dy = MZsol7(t,y)
%input arg of ODE45 - for geodesic differential equations of sol3 space
dy=zeros(6,1);
dy(1)=y(2);
dy(2)=-2*(y(2))*(y(6));
dy(3)=y(4);
dy(4)=2*(y(4))*(y(6));
dy(5)=y(6);
dy(6)=(exp(2*y(5)))*(y(2)^2)-(exp(-2*y(5)))*(y(4)^2);
end
```

```
%Description: plot geodesic of sol3 space
clear all;clc;
tspan = [0 3];
y0 = [0;1;0;1;0;1];
[t, y] = ode45(@MZsol7, tspan, y0)
plot3(y(:,1),y(:,3),y(:,5),'linewidth',3,'color','b');
axis square;
xlabel('X');ylabel('Y');zlabel('Z');
grid on;box on;
```

Acknowledgments

The authors would like to thank the University of Mohaghegh Ardabili for support.

References

[1] M. Do Carmo, *Riemannian Geometry*, Birkhauser, Boston, (1993).
[2] B. D. Hahn and D. T. Valentine, *Essential MATLAB for Engineers and Scientists*, Elsevier.
[3] W. Kuhnel, *Differential Geometry: Curves – Surfaces – Manifolds*, AMS, (2006).
[4] R. Lopez and M. I. Munteanu, *Translation minimal surfaces in* Sol_3, J. Math. Soc. Japan, in press, (2010).
[5] S. R. Otto, J. P. Denier, *An Introduction to Programming and Numerical Methods in MATLAB*, Springer, (2005).

[6] P. Petersen, *Riemannian geometry*, Second Edition, volume 171 of Graduate Texts in Math-ematics. Springer-Verlag, New York, (2006).
[7] P. Scott, *The geometries of 3-manifolds*, Bull. London Math. Soc., (1983).
[8] W. Thurston, Three-dimensional geometry and topology, Princeton Math. Ser. 35, Princenton Univ. Press, Princeton, NJ. (1997).

Nemat Abazari
Department of Mathematics and Applications, Faculty of Mathematical Sciences, University of Mohaghegh Ardabili, 56199-11367, Ardabil, Iran.
Email: nematabazari@gmail.com

Masoud Sahraiy
Department of Mathematics and Applications, Faculty of Mathematical Sciences, University of Mohaghegh Ardabili, 56199-11367, Ardabil, Iran.
Email: masoud.sahrai@gmail.com