

```

<!DOCTYPE html>
<html><head><title>...Jacobiano+ tensore metrico...</title>
<style type="text/css">
h2 { color:blue; background:yellow }
i { background:#bbffbb; }
b { color:red }
</style></head>
<body><blockquote>
<h2>Jacobiano e nuovo tensore metrico in coordinate sferiche</h2>
<p><a href="http://www.elegio.it/omnia/mc/"
>http://www.elegio.it/omnia/mc/</a> argomenti di
meccanica classica...</p>
Dopo avere installato sul proprio PC il
gratutito e trasportabilissimo, su chiavetta USB, wxMaxima...
<p><a href="http://www.elegio.it/omnia/mc/mc20140203.mac.html"
>http://www.elegio.it/omnia/mc/mc20140203.mac.html</a></p>
<p><a href="http://www.elegio.it/omnia/mc/mc20140203.wxm.html"
>http://www.elegio.it/omnia/mc/mc20140203.wxm.html</a></p>
<pre>

```

```

<i>Pro memoria di come visualizzare numeri in virgola
mobile e con alta precisione</i>

```

```

(%i1) fpprec:36$
print("256 elevato a 1250 vale quasi ",
" 2 moltiplicato per 10 alla 3010")$
print(bfloat(256^1250))$
print("Un numeretto non molto piccolino...")$

```

```

256 elevato a 1250 vale quasi 2 moltiplicato per 10 alla 3010
1.99506311688075838488374216268358508b3010
Un numeretto non molto piccolino...

```

```

<i>Ecco le due funzioni fondamentali definite qui per lavorare
in spazi CLASSICI ossia in TRE DIMENSIONI</i>

```

```

(%i5) fajacobiano(X,Y,Z):=block([mj],
mj:matrix([diff(X,x),diff(X,y),diff(X,z)],
[diff(Y,x),diff(Y,y),diff(Y,z)],
[diff(Z,x),diff(Z,y),diff(Z,z)]),
return(ratsimp(mj)))$
print("definisco la funzione: fajacobiano(X,Y,Z)")$
/*<hr/>*/
nuovogm(GM,Xv,Yv,Zv):=block([gmp,gm,jacobiano],
gmp:trigsimp(ev(GM,
X=Xv,Y=Yv,Z=Zv)),
jacobiano:matrix([diff(Xv,x),diff(Xv,y),diff(Xv,z)],
[diff(Yv,x),diff(Yv,y),diff(Yv,z)],
[diff(Zv,x),diff(Zv,y),diff(Zv,z)]),
gm:ratsimp(transpose(jacobiano).gmp.jacobiano),
return (trigsimp(ev(gm,x=X,y=Y,z=Z)))
)$
print("definisco la funzione: nuovogm(GM,Xv,Yv,Zv)")$

```

```

definisco la funzione: fajacobiano(X,Y,Z)
definisco la funzione: nuovogm(GM,Xv,Yv,Zv)

```

Ora scrivo le coordinate cartesiane espresse come funzioni delle coordinate sferiche tradizionali...

```
(%i9) X1 : x*sin(y)*cos(z)$
      Y1 : x*sin(y)*sin(z)$
      Z1 : x*cos(y)$
      print("Cartesiane classiche espresse da sferiche classiche ")$
      print("X1 vale: ",X1)$
      print("Y1 vale: ",Y1)$
      print("Z1 vale: ",Z1)$
```

Cartesiane classiche espresse da sferiche classiche

X1 vale: x sin(y) cos(z)

Y1 vale: x sin(y) sin(z)

Z1 vale: x cos(y)

Ecco come viene lo jacobiano usato per ottenere il tensore metrico in coordinate sferiche classiche

```
(%i16) fajacobiano(X1,Y1,Z1);
(%o16) [ sin(y) cos(z)  x cos(y) cos(z)  -x sin(y) sin(z)
        sin(y) sin(z)  x cos(y) sin(z)   x sin(y) cos(z)
        cos(y)         -x sin(y)         0 ]
```

Ora scrivo le coordinate sferiche classiche espresse come coordinate cartesiane classiche:

```
(%i17) X2:sqrt(x^2+y^2+z^2)$
      Y2:acos(z/sqrt(x^2+y^2+z^2))$
      Z2:asin(y/sqrt(x^2+y^2))$
      print("Sferiche classiche espresse tramite le cartesiane")$
      print("X2 vale : ",X2)$
      print("Y2 vale : ",Y2)$
      print("Z2 vale : ",Z2)$
```

Sferiche classiche espresse tramite le cartesiane

X2 vale : $\sqrt{z^2+y^2+x^2}$

Y2 vale : $\text{acos}\left(\frac{z}{\sqrt{z^2+y^2+x^2}}\right)$

Z2 vale : $\text{asin}\left(\frac{y}{\sqrt{y^2+x^2}}\right)$

Il tensore metrico cartesiano in tre dimensioni e' semplicissimo ossia e' la matrice identita'

```
(%i24) gmcartesiana:diag_matrix(1,1,1)$
print("Il tensore metrico cartesiano classico !")$
print("Semplicemente una matrice identita' ",
      "e il covariante coincide col controvariante")$
print("gmcartesiana vale: ",gmcartesiana)$
```

*Il tensore metrico cartesiano classico !
Semplicemente una matrice identita'
e il covariante coincide col controvariante*

$$gmcartesiana \text{ vale: } \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

<i>Calcolo dal cartesiano classico e dalle formule per ottenere il cartesiano note le coordinate sferiche classiche:</i>

```
(%i28) gmsferico:nuovogm(gmcartesiana,X1,Y1,Z1)$
print("calcolo nuovogm(gmcartesiana,X1,Y1,Z1) ")$
print("X1 : ",X1)$
print("Y1 : ",Y1)$
print("Z1 : ",Z1)$
print("Ed ecco il tensore metrico covariante sferico classico ")$
print("gmsferico vale ",gmsferico)$
```

calcolo nuovogm(gmcartesiana,X1,Y1,Z1)

X1 : x sin(y) cos(z)

Y1 : x sin(y) sin(z)

Z1 : x cos(y)

Ed ecco il tensore metrico covariante sferico classico

$$gmsferico \text{ vale } \begin{bmatrix} 1 & 0 & 0 \\ 0 & x^2 & 0 \\ 0 & 0 & x^2 \sin^2(y) \end{bmatrix}$$

<i>Per curiosita' vediamo cosa e' la matrice jacobiana quando si vuole riottenere il tensore metrico cartesiano ossia la semplicissima matrice identita'</i>

```
(%i35) print(" La matrice jacobiana delle coordinate sferiche ",
            "esprese tramite le coordinate cartesiane")$
print("fajacobiano(X2,Y2,Z2) : ", fajacobiano(X2,Y2,Z2))$
```

*La matrice jacobiana delle coordinate sferiche
esprese tramite le coordinate cartesiane*

$$fajacobiano(X2,Y2,Z2) : \begin{bmatrix} \frac{x}{\sqrt{z^2+y^2+x^2}} & \frac{y}{\sqrt{z^2+y^2+x^2}} & \frac{z}{\sqrt{z^2+y^2+x^2}} \\ \frac{xz}{\sqrt{y^2+x^2}(z^2+y^2+x^2)} & \frac{yz}{\sqrt{y^2+x^2}(z^2+y^2+x^2)} & \frac{\sqrt{y^2+x^2}}{z^2+y^2+x^2} \\ \frac{xy}{|x|y^2+x^2|x|} & \frac{x^2}{|x|y^2+x^2|x|} & 0 \end{bmatrix}$$

Se il tensore metrico sferico covariante e' giusto, facendo l'operazione inversa debbo riottenere il classicissimo tensore metrico cartesiano ossia la matrice identita' di ordine 3.

```
(%i37) gmcartesianaricalcolata:nuovogm(gmsferico,X2,Y2,Z2)$
print("Lecito chiedersi se gmsferico e' giusto")$
print("Verifico facendo l'operazione inversa")$
print("Uso il tensore metrico sferico",
      " per riottenere il cartesiano")$
print("calcolo nuovogm(gmsferico,X2,Y2,Z2) ")$
print("X2 : ",X2)$
print("Y2 : ",Y2)$
print("Z2 : ",Z2)$
print("Ed ecco il tensore metrico covariante cartesiano ")$
print("gmcartesianaricalcolata vale ",
      gmcartesianaricalcolata)$
```

Lecito chiedersi se gmsferico e' giusto

Verifico facendo l'operazione inversa

Uso il tensore metrico sferico per riottenere il cartesiano

calcolo nuovogm(gmsferico,X2,Y2,Z2)

$$X2 : \sqrt{z^2 + y^2 + x^2}$$

$$Y2 : \arccos\left(\frac{z}{\sqrt{z^2 + y^2 + x^2}}\right)$$

$$Z2 : \arcsin\left(\frac{y}{\sqrt{y^2 + x^2}}\right)$$

Ed ecco il tensore metrico covariante cartesiano

$$gmcartesianaricalcolata \text{ vale } \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

```
(%i47) determinant(gmsferico);
```

```
(%o47) X^4 sin(Y)^2
```

<hr/>

Qui iniziano le formule non ovvie...

```
(%i48) X3 : x*sqrt(1-y^2)*cos(z)$
Y3 : x*sqrt(1-y^2)*sin(z)$
Z3 : x*y$
print("Cartesiane classiche espresse da sferiche NON classiche ")$
print("X3 vale: ",X3)$
print("Y3 vale: ",Y3)$
print("Z3 vale: ",Z3)$
```

Cartesiane classiche espresse da sferiche NON classiche

$$X3 \text{ vale: } x\sqrt{1-y^2} \cos(z)$$

$$Y3 \text{ vale: } x\sqrt{1-y^2} \sin(z)$$

$$Z3 \text{ vale: } x y$$

<i>Per calcolare le sferiche NON classiche usando le solite coordinate cartesiane : </i>

```
(%i55) X4:sqrt(x^2+y^2+z^2)$
      Y4:z/sqrt(x^2+y^2+z^2)$
      Z4:asin(y/sqrt(x^2+y^2))$
      print("Sferiche NON classiche espresse tramite le cartesiane")$
      print("X4 vale : ",X4)$
      print("Y4 vale : ",Y4)$
      print("Z4 vale : ",Z4)$
```

Sferiche NON classiche espresse tramite le cartesiane

$$X4 \text{ vale : } \sqrt{z^2 + y^2 + x^2}$$

$$Y4 \text{ vale : } \frac{z}{\sqrt{z^2 + y^2 + x^2}}$$

$$Z4 \text{ vale : } \operatorname{asin}\left(\frac{y}{\sqrt{y^2 + x^2}}\right)$$

<i>Ecco come viene bene il tensore metrico usando queste coordinate sferiche NON classiche ossia quando uso per theta che sarebbe 90-latitudine ossia vale 0 al polo Nord e 180 al polo sud o meglio, in radianti %pi...</i>

```
(%i62) gmsferico2:nuovogm(gmcartesiana,X3,Y3,Z3)$
      print("calcolo nuovogm(gmcartesiana,X3,Y3,Z3) ")$
      print("X3 : ",X3)$
      print("Y3 : ",Y3)$
      print("Z3 : ",Z3)$
      print("Ed ecco il tensore metrico covariante sferico NON classico ")$
      print("gmsferico2 vale ",gmsferico2)$
```

calcolo nuovogm(gmcartesiana,X3,Y3,Z3)

$$X3 : x\sqrt{1-y^2} \cos(z)$$

$$Y3 : x\sqrt{1-y^2} \sin(z)$$

$$Z3 : xy$$

Ed ecco il tensore metrico covariante sferico NON classico

$$gmsferico2 \text{ vale } \begin{bmatrix} 1 & 0 & 0 \\ 0 & -\frac{x^2}{y^2-1} & 0 \\ 0 & 0 & x^2 - x^2 y^2 \end{bmatrix}$$

<i>A questo punto controllo se, noto il tensore metrico sferico NON classico riesco a riottenere quello solito cartesiano...</i>

```
(%i69) gmcartesianaricalcolata2:nuovogm(gmsferico2,X4,Y4,Z4)$
print("Lecito chiedersi se gmsferico2 e' giusto")$
print("Verifico facendo l'operazione inversa")$
print("Uso il tensore metrico sferico NON classico",
      " per riottenere il cartesiano")$
print("calcolo nuovogm(gmsferico2,X4,Y4,Z4) ")$
print("X4 : ",X4)$
print("Y4 : ",Y4)$
print("Z4 : ",Z4)$
print("Ed ecco il tensore metrico covariante cartesiano ")$
print("gmcartesianaricalcolata2 vale ",
      gmcartesianaricalcolata2)$
```

Lecito chiedersi se gmsferico2 e' giusto
Verifico facendo l'operazione inversa
Uso il tensore metrico sferico NON classico
per riottenere il cartesiano
calcolo nuovogm(gmsferico2,X4,Y4,Z4)

$$X4 : \sqrt{z^2 + y^2 + x^2}$$

$$Y4 : \frac{z}{\sqrt{z^2 + y^2 + x^2}}$$

$$Z4 : \operatorname{asin}\left(\frac{y}{\sqrt{y^2 + x^2}}\right)$$

Ed ecco il tensore metrico covariante cartesiano

$$\text{gmcartesianaricalcolata2 vale } \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

<i>Pero' c'e' questo difettuccio ossia il determinante della metrica non vale 1 ossia non e' indipendente dal valore locale delle coordinate...</i>

```
(%i79) ratsimp(determinant(gmsferico2));
(%o79) X^4
```

<i>Ma se voglio NON far dipendere il determinante del tensore metrico dalle coordinate, come posso fare ?</i>

<i>ECCO COME SI FA !</i>

```
(%i80) X5 : (3*x)^(1/3)*sqrt(1-y^2)*cos(z)$
      Y5 : (3*x)^(1/3)*sqrt(1-y^2)*sin(z)$
      Z5 : (3*x)^(1/3)*y$
      print("Cartesiane classiche espresse da sferiche BISONN classiche ")$
      print("X5 vale: ",X5)$
      print("Y5 vale: ",Y5)$
      print("Z5 vale: ",Z5)$
```

Cartesiane classiche espresse da sferiche BISONN classiche

$X5 \text{ vale: } 3^{1/3} x^{1/3} \sqrt{1-y^2} \cos(z)$

$Y5 \text{ vale: } 3^{1/3} x^{1/3} \sqrt{1-y^2} \sin(z)$

$Z5 \text{ vale: } 3^{1/3} x^{1/3} y$

<i>Da cui mi vien fuori questo tensore metrico</i>

```
(%i87) gmsferico3:nuovogm(gmcartesiana,X5,Y5,Z5)$
      print("calcolo nuovogm(gmcartesiana,X5,Y5,Z5) ")$
      print("X5 : ",X5)$
      print("Y5 : ",Y5)$
      print("Z5 : ",Z5)$
      print("Ed ecco il tensore metrico ",
            "covariante sferico BISONN classico ")$
      print("gmsferico3 vale ",gmsferico3)$
```

calcolo nuovogm(gmcartesiana,X5,Y5,Z5)

$X5 : 3^{1/3} x^{1/3} \sqrt{1-y^2} \cos(z)$

$Y5 : 3^{1/3} x^{1/3} \sqrt{1-y^2} \sin(z)$

$Z5 : 3^{1/3} x^{1/3} y$

Ed ecco il tensore metrico covariante sferico BISONN classico

gmsferico3 vale

$$\begin{bmatrix} \frac{1}{3^{4/3} x^{4/3}} & 0 & 0 \\ 0 & \frac{3^{2/3} x^{2/3}}{y^2-1} & 0 \\ 0 & 0 & x^{2/3} (3^{2/3} - 3^{2/3} y^2) \end{bmatrix}$$

<i>Dunque con questa metrica sferica BISONN classica il determinante del tensore metrico vale 1 come nel caso della metrica cartesiana e questo fatto semplifica la scrittura di certi operatori differenziali e varie altre cose...</i>

```
(%i94) ratsimp(determinant(gmsferico3));
(%o94) 1
```

<i>Lo scrivo perche' e' TROOOOPPPPO BELLO !</i>

```
</pre><h3 style="color:red">gmbello:diag_matrix(1/(3*X)^(4/3),
(3*X)^(2/3)/(1-Y^2), (3*X)^(2/3)*(1-Y^2))</h3></pre>
```

```
(%i95) gmbello:diag_matrix(1/(3*X)^(4/3),
(3*X)^(2/3)/(1-Y^2), (3*X)^(2/3)*(1-Y^2))$
print("gmbello ossia gmsferico3 vale:",gmbello)$
```

$$gmbello \text{ ossia } gmsferico3 \text{ vale: } \begin{bmatrix} \frac{1}{3^{4/3} X^{4/3}} & 0 & 0 \\ 0 & \frac{3^{2/3} X^{2/3}}{1-Y^2} & 0 \\ 0 & 0 & 3^{2/3} X^{2/3} (1-Y^2) \end{bmatrix}$$

Faccio molti errorini ... controllo di non avere sbagliato...

```
(%i97) print("Verifico di non avere fatto errori : ")$
print("ratsimp(gmbello-gmsferico3) vale :",
ratsimp(gmbello-gmsferico3))$
```

Verifico di non avere fatto errori :

$$ratsimp(gmbello-gmsferico3) \text{ vale : } \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

Certo in gmbello mi compaiono un po' di radici cubiche ma in fondo non e' un dramma...

Se ho le coordinate cartesiane e voglio esprimerle come BISNON classiche devo usare queste formule che mi servono anche per verificare la correttezza del tensore metrico covariante espresso nelle coordinate BISNON classiche...

```
(%i99) X6:(sqrt(x^2+y^2+z^2)^3)/3$
Y6:z/sqrt(x^2+y^2+z^2)$
Z6:asin(y/sqrt(x^2+y^2))$
print("Sferiche BISNON classiche",
" espresse tramite le cartesiane :")$
print("X6 vale : ",X6)$
print("Y6 vale : ",Y6)$
print("Z6 vale : ",Z6)$
```

Sferiche BISNON classiche espresse tramite le cartesiane :

$$X6 \text{ vale : } \frac{(z^2+y^2+x^2)^{3/2}}{3}$$

$$Y6 \text{ vale : } \frac{z}{\sqrt{z^2+y^2+x^2}}$$

$$Z6 \text{ vale : } \operatorname{asin}\left(\frac{y}{\sqrt{y^2+x^2}}\right)$$

In sostanza la coordinata radiale diventa PROPORZIONALE AL VOLUME DELLA SFERA COL CENTRO ALL'ORIGINE DELLE COORDINATE

<i>Infatti facendo la verifica riottengo il tensore metrico cartesiano ossia la solita matrice identita'</i>

```
(%i106) gmcartesianaricalcolata3:nuovogm(gmsferico3,X6,Y6,Z6)$
print("Lecito chiedersi se gmsferico3 e' giusto")$
print("Verifico facendo l'operazione inversa")$
print("Uso il tensore metrico sferico NON classico",
" per riottenere il cartesiano")$
print("calcolo nuovogm(gmsferico3,X6,Y6,Z6) ")$
print("X6 : ",X6)$
print("Y6 : ",Y6)$
print("Z6 : ",Z6)$
print("Ed ecco il tensore metrico covariante cartesiano ")$
print("gmcartesianaricalcolata3 vale ",
gmcartesianaricalcolata3)$
```

*Lecito chiedersi se gmsferico3 e' giusto
Verifico facendo l'operazione inversa
Uso il tensore metrico sferico NON classico
per riottenere il cartesiano
calcolo nuovogm(gmsferico3,X6,Y6,Z6)*

$$X6 : \frac{(z^2 + y^2 + x^2)^{3/2}}{3}$$

$$Y6 : \frac{z}{\sqrt{z^2 + y^2 + x^2}}$$

$$Z6 : \operatorname{asin}\left(\frac{y}{\sqrt{y^2 + x^2}}\right)$$

Ed ecco il tensore metrico covariante cartesiano

$$\text{gmcartesianaricalcolata3 vale } \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

`</pre>`

`<p>Per usare questo file con http://maxima.sourceforge.net/ cancellare l'estensione .html e lasciargli l'estensione .mac se e' stato creato con questa estensione da wxMaxima.oppure se e' un file che aveva estensione .wxm a cui io ho aggiunto, per camuffarlo da file di internet, l'estensione .html bisogna cancellare questa e ridargli la estensione che aveva all'origine ossia .wxm</p>`

`<p>Di solito Windows NASCONDE LE ESTENSIONI per evitare che un utente pasticcione ed ignorante che non sa che servono per identificare il tipo del file le cancelli volendo scrivere nomi di file senza punti</p>`

`<p>Pero' questa precauzione intralcia quando si vuole mascherare una estensione per CAMUFFARE il file da file .html</p>`

<p>Per aggirare questa difficolta' si puo' aprire il file usando NOTEPAD.EXE che e' un editor standard presente in qualunque calcolatore che usa Windows. Lo si apre e lo si salva mettendogli l'estensione giusta ossia, per wxMaxima, a seconda dei casi, .mac o .wxm </p>

<h2>Amen e BUON DIVERTIMENTO con gli Jacobiani !</h2>
</blockquote></body>
</html>