



Coordinate systems

Mauro Mongiardo¹

¹ Department of Engineering, University of Perugia, Perugia, Italy.

Table of contents

1. Coordinate systems
2. Coordinates transformations
3. Unit vectors
4. Coordinate transformations for vector components
5. Pauli matrices in Cylindrical coordinates
6. Pauli matrices in Spherical coordinates

Coordinate systems

Coordinate systems

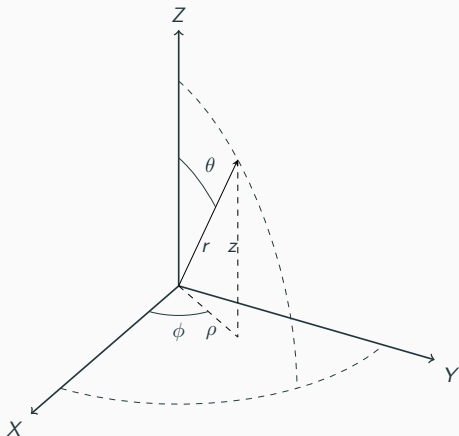


Figure 1: Convention for coordinate orientation.

We will use cartesian, circular cylindrical and spherical coordinates systems.

Coordinates transformations

Cylindrical Coordinates

When passing from rectangular to cylindrical coordinates the following transformations apply:

$$\begin{aligned}\rho &= \sqrt{x^2 + y^2} = r \sin \theta \\ \phi &= \tan^{-1} \left(\frac{y}{x} \right) \\ z &= r \cos \theta\end{aligned}\tag{1}$$

Cylindrical Coordinates

When passing from rectangular to cylindrical coordinates the following transformations apply:

$$\begin{aligned}\rho &= \sqrt{x^2 + y^2} = r \sin \theta \\ \phi &= \tan^{-1} \left(\frac{y}{x} \right) \\ z &= r \cos \theta\end{aligned}\tag{1}$$

From cylindrical to rectangular:

$$\begin{aligned}x &= \rho \cos \phi \\ y &= \rho \sin \phi \\ z &= z\end{aligned}\tag{2}$$

Cylindrical Coordinates

When passing from rectangular to cylindrical coordinates the following transformations apply:

$$\begin{aligned}\rho &= \sqrt{x^2 + y^2} = r \sin \theta \\ \phi &= \tan^{-1} \left(\frac{y}{x} \right) \\ z &= r \cos \theta\end{aligned}\tag{1}$$

From cylindrical to rectangular:

$$\begin{aligned}x &= \rho \cos \phi \\ y &= \rho \sin \phi \\ z &= z\end{aligned}\tag{2}$$

The volume element is

$$dV = \rho d\rho d\phi dz\tag{3}$$

Spherical Coordinates

Rectangular to spherical

$$\begin{aligned}r &= \sqrt{x^2 + y^2 + z^2} = \sqrt{\rho^2 + z^2} \\ \theta &= \tan^{-1} \left(\frac{\sqrt{x^2 + y^2}}{z} \right) = \tan^{-1} \left(\frac{\rho}{z} \right) \\ \phi &= \tan^{-1} \left(\frac{y}{x} \right) .\end{aligned}\tag{4}$$

Spherical Coordinates

Rectangular to spherical

$$\begin{aligned}r &= \sqrt{x^2 + y^2 + z^2} = \sqrt{\rho^2 + z^2} \\ \theta &= \tan^{-1} \left(\frac{\sqrt{x^2 + y^2}}{z} \right) = \tan^{-1} \left(\frac{\rho}{z} \right) \\ \phi &= \tan^{-1} \left(\frac{y}{x} \right) .\end{aligned}\tag{4}$$

and for the spherical to rectangular

$$\begin{aligned}x &= r \sin \theta \cos \phi \\ y &= r \sin \theta \sin \phi \\ z &= r \cos \theta .\end{aligned}\tag{5}$$

Spherical Coordinates

Rectangular to spherical

$$\begin{aligned}r &= \sqrt{x^2 + y^2 + z^2} = \sqrt{\rho^2 + z^2} \\ \theta &= \tan^{-1} \left(\frac{\sqrt{x^2 + y^2}}{z} \right) = \tan^{-1} \left(\frac{\rho}{z} \right) \\ \phi &= \tan^{-1} \left(\frac{y}{x} \right) .\end{aligned}\tag{4}$$

and for the spherical to rectangular

$$\begin{aligned}x &= r \sin \theta \cos \phi \\ y &= r \sin \theta \sin \phi \\ z &= r \cos \theta .\end{aligned}\tag{5}$$

The volume element is

$$dV = r^2 \sin \theta \, dr \, d\theta \, d\phi\tag{6}$$

Examples

Before proceeding further complete examples 3.4, 3.5, 3.6

Unit vectors

Unit vectors: transformations from cylindrical/spherical to rectangular

The coordinate unit vectors are defined as vectors of unit length pointing along coordinate lines in the direction of increasing coordinate variables.

Unit vectors: transformations from cylindrical/spherical to rectangular

The coordinate unit vectors are defined as vectors of unit length pointing along coordinate lines in the direction of increasing coordinate variables.

It is left as an exercise to draw them in the three coordinate systems.

Unit vectors: transformations from cylindrical/spherical to rectangular

The coordinate unit vectors are defined as vectors of unit length pointing along coordinate lines in the direction of increasing coordinate variables.

It is left as an exercise to draw them in the three coordinate systems.

The following rules are applicable to transformations among unit coordinate vectors.

Transformations from cylindrical/spherical to rectangular:

$$\begin{aligned}\mathbf{u}_x &= \mathbf{u}_\rho \cos \phi - \mathbf{u}_\phi \sin \phi \\ &= \mathbf{u}_r \sin \theta \cos \phi + \mathbf{u}_\theta \cos \theta \cos \phi - \mathbf{u}_\phi \sin \phi \\ \mathbf{u}_y &= \mathbf{u}_\rho \sin \phi + \mathbf{u}_\phi \cos \phi \\ &= \mathbf{u}_r \sin \theta \sin \phi + \mathbf{u}_\theta \cos \theta \sin \phi + \mathbf{u}_\phi \cos \phi \\ \mathbf{u}_z &= \mathbf{u}_r \cos \theta - \mathbf{u}_\theta \sin \theta\end{aligned}\tag{7}$$

Unit vectors: transformations from cylindrical/spherical to rectangular

The coordinate unit vectors are defined as vectors of unit length pointing along coordinate lines in the direction of increasing coordinate variables.

It is left as an exercise to draw them in the three coordinate systems.

The following rules are applicable to transformations among unit coordinate vectors.

Transformations from cylindrical/spherical to rectangular:

$$\begin{aligned}\mathbf{u}_x &= \mathbf{u}_\rho \cos \phi - \mathbf{u}_\phi \sin \phi \\ &= \mathbf{u}_r \sin \theta \cos \phi + \mathbf{u}_\theta \cos \theta \cos \phi - \mathbf{u}_\phi \sin \phi \\ \mathbf{u}_y &= \mathbf{u}_\rho \sin \phi + \mathbf{u}_\phi \cos \phi \\ &= \mathbf{u}_r \sin \theta \sin \phi + \mathbf{u}_\theta \cos \theta \sin \phi + \mathbf{u}_\phi \cos \phi \\ \mathbf{u}_z &= \mathbf{u}_r \cos \theta - \mathbf{u}_\theta \sin \theta\end{aligned}\tag{7}$$

Transformations from rectangular/spherical to cylindrical

Transformations from rectangular/spherical to cylindrical

$$\begin{aligned}\mathbf{u}_\rho &= \mathbf{u}_x \cos \phi + \mathbf{u}_y \sin \phi = \mathbf{u}_r \sin \theta + \mathbf{u}_\theta \cos \theta \\ \mathbf{u}_\phi &= -\mathbf{u}_x \sin \phi + \mathbf{u}_y \cos \phi \\ \mathbf{u}_z &= \mathbf{u}_r \cos \theta - \mathbf{u}_\theta \sin \theta\end{aligned}\tag{8}$$

Transformations from rectangular/cylindrical to spherical

Transformations from rectangular/cylindrical to spherical:

$$\begin{aligned}\mathbf{u}_r &= \mathbf{u}_x \sin \theta \cos \phi + \mathbf{u}_y \sin \theta \sin \phi + \mathbf{u}_z \cos \theta \\ &= \mathbf{u}_\rho \sin \theta + \mathbf{u}_z \cos \theta \\ \mathbf{u}_\theta &= \mathbf{u}_x \cos \theta \cos \phi + \mathbf{u}_y \cos \theta \sin \phi - \mathbf{u}_z \sin \theta \\ &= \mathbf{u}_\rho \cos \theta - \mathbf{u}_z \sin \theta \\ \mathbf{u}_\phi &= -\mathbf{u}_x \sin \phi + \mathbf{u}_y \cos \phi\end{aligned}\tag{9}$$

Coordinate transformations for vector components

Coordinate transformations for vector components

Rectangular to cylindrical components

$$\begin{aligned}F_{\rho} &= F_x \cos \phi + F_y \sin \phi \\F_{\phi} &= -F_x \sin \phi + F_y \cos \phi \\F_z &= F_z\end{aligned}\tag{10}$$

Coordinate transformations for vector components

Rectangular to cylindrical components

$$\begin{aligned}F_{\rho} &= F_x \cos \phi + F_y \sin \phi \\F_{\phi} &= -F_x \sin \phi + F_y \cos \phi \\F_z &= F_z\end{aligned}\tag{10}$$

Rectangular to spherical components

$$\begin{aligned}F_r &= F_x \sin \theta \cos \phi + F_y \sin \theta \sin \phi + F_z \cos \theta \\F_{\theta} &= F_x \cos \theta \cos \phi + F_y \cos \theta \sin \phi - F_z \sin \theta \\F_{\phi} &= -F_x \sin \phi + F_y \cos \phi\end{aligned}\tag{11}$$

Cylindrical to rectangular components

$$\begin{aligned}F_x &= F_\rho \cos \phi - F_\phi \sin \phi \\F_y &= F_\rho \sin \phi + F_\phi \cos \phi \\F_z &= F_z\end{aligned}\tag{12}$$

Cylindrical to rectangular components

$$\begin{aligned}F_x &= F_\rho \cos \phi - F_\phi \sin \phi \\F_y &= F_\rho \sin \phi + F_\phi \cos \phi \\F_z &= F_z\end{aligned}\tag{12}$$

Cylindrical to spherical components

$$\begin{aligned}F_r &= F_\rho \sin \theta + F_z \cos \theta \\F_\theta &= F_\rho \cos \theta - F_z \sin \theta \\F_\phi &= F_\phi\end{aligned}\tag{13}$$

Spherical to rectangular components

$$\begin{aligned}F_x &= F_r \sin \theta \cos \phi + F_\theta \cos \theta \cos \phi - F_\phi \sin \phi \\F_y &= F_r \sin \theta \sin \phi + F_\theta \cos \theta \sin \phi + F_\phi \cos \phi \\F_z &= F_r \cos \theta - F_\theta \sin \phi\end{aligned}\tag{14}$$

Coordinate systems

Spherical to rectangular components

$$\begin{aligned}F_x &= F_r \sin \theta \cos \phi + F_\theta \cos \theta \cos \phi - F_\phi \sin \phi \\F_y &= F_r \sin \theta \sin \phi + F_\theta \cos \theta \sin \phi + F_\phi \cos \phi \\F_z &= F_r \cos \theta - F_\theta \sin \phi\end{aligned}\tag{14}$$

Spherical to cylindrical components

$$\begin{aligned}F_\rho &= F_r \sin \theta + F_\theta \cos \theta \\F_\phi &= F_\phi \\F_z &= F_r \cos \theta - F_\theta \sin \theta\end{aligned}\tag{15}$$

Summary of coordinate transformations:r2c

```
/* [wxMaxima batch file version 1] [ DO NOT EDIT BY HAND! ]*/
/* [ Created with wxMaxima version 11.08.0 ] */

/* [wxMaxima: input    start ] */
kill(all)$
load(vect)$

/* numerical
fpprintprec:5$
ratprint : false$
x :5.0 ;
y :2.0 ;
z :4.6 ;
*/

/* rectangular to cylindrical */
%rho : sqrt(x^2+y^2)$
phi : atan2(y,x)$
z :z$
print("rectangular to cylindrical")$
print('%rho, " = ",%rho)$
print('phi, " = ",phi)$
print('z, " = ",z)$

print("bye")$
/* [wxMaxima: input    end    ] */

/* Maxima can't load/batch files which end with a comment! */
"Created with wxMaxima"$
```

Summary of coordinate transformations:c2r

```
/* [wxMaxima batch file version 1] [ DO NOT EDIT BY HAND! ]*/
/* [ Created with wxMaxima version 11.08.0 ] */

/* [wxMaxima: input    start ] */
kill(all)$
load(vect)$
/* c2r.wxm */

/* numerical
fpprintprec:5$
ratprint : false$
%rho :5.0 ;
phi : 3.14/4.0 ;
z :4.6 ;
*/

/* cylindrical to rectangular */
x : %rho * cos(phi)$
y : %rho * sin(phi)$
z : z$

print("cylindrical to rectangular")$
print('x, " = ",x)$
print('y, " = ",y)$
print('z, " = ",z)$

print("bye")$
/* [wxMaxima: input    end  ] */

/* Maxima can't load/batch files which end with a comment! */
"Created with wxMaxima"$
```

Summary of coordinate transformations: r2s

```
/* [wxMaxima batch file version 1] [ DO NOT EDIT BY HAND! ]*/
/* [ Created with wxMaxima version 11.08.0 ] */

/* [wxMaxima: input    start ] */
kill(all)$
load(vect)$
/* r2s.wxm */

/* numerical
fpprintprec:5$
ratprint : false$
x :5.0 ;
y :2.0 ;
z :4.6 ;
*/

r : sqrt(x^2+y^2+z^2)$
theta : atan2(sqrt(x^2 +y^2),z)$
phi : atan2(y,x)$

print("rectangular to spherical")$
print('r, " = ",r)$
print('theta, " = ",theta)$
print('phi, " = ",phi)$

print("bye")$
/* [wxMaxima: input    end    ] */

/* Maxima can't load/batch files which end with a comment! */
"Created with wxMaxima"$
```

Summary of coordinate transformations: s2r

```
/* [wxMaxima batch file version 1] [ DO NOT EDIT BY HAND! ]*/
/* [ Created with wxMaxima version 11.08.0 ] */

/* [wxMaxima: input    start ] */
kill(all)$
load(vect)$
/* s2r.wxm */

/* numerical
fpprintprec:5$
ratprint : false$
r :5.0 ;
theta :float(%pi)/4 ;
phi :4.6 ;
*/

x : r * sin(theta) * cos(phi)$
y : r * sin(theta) * sin(phi)$
z : r *cos(theta)$

print("spherical to rectangular")$
print('x, " = ",x)$
print('y, " = ",y)$
print('z, " = ",z)$

print("bye")$
/* [wxMaxima: input    end    ] */

/* Maxima can't load/batch files which end with a comment! */
"Created with wxMaxima"$
```

Summary of coordinate transformations: c2s

```
/* [wxMaxima batch file version 1] [ DO NOT EDIT BY HAND! ]*/
/* [ Created with wxMaxima version 11.08.0 ] */

/* [wxMaxima: input    start ] */
kill(all)$
load(vect)$
/* c2s.wxm */

/* numerical
fpprintprec:5$
ratprint : false$
%rho :5.0 ;
phi :4.6 ;
z :2.0;
*/

/* cylindrical to spherical */
r : sqrt(%rho^2 + z^2)$
theta : atan2(%rho,z) $
phi : phi$

print("cylindrical to spherical")$
print('r, " = ",r)$
print('theta, " = ",theta)$
print('phi, " = ",phi)$

print("bye")$
/* [wxMaxima: input    end  ] */

/* Maxima can't load/batch files which end with a comment! */
"Created with wxMaxima"$
```

Summary of coordinate transformations: s2c

```
/* [wxMaxima batch file version 1] [ DO NOT EDIT BY HAND! ]*/
/* [ Created with wxMaxima version 11.08.0 ] */

/* [wxMaxima: input    start ] */
kill(all)$
load(vect)$
/* s2c.wxm */

/* numerical */
fpprintprec:5$
ratprint : false$
r :5.0 ;
theta :float(%pi)/4 ;
phi :0.6 ;

%rho : r * sin(theta) $
phi : phi$
z : r *cos(theta)$

print("spherical to cylindrical")$
print('%rho, " = ",%rho)$
print('phi, " = ",phi)$
print('z, " = ",z)$

print("bye")$
/* [wxMaxima: input    end    ] */

/* Maxima can't load/batch files which end with a comment! */
"Created with wxMaxima"$
```


Summary of vector transformations: v r2c

```
/* [wxMaxima batch file version 1] [ DO NOT EDIT BY HAND! ]*/
/* [ Created with wxMaxima version 11.08.0 ] */

/* [wxMaxima: input start ] */
kill(all)$
load(vect)$
/* V.r2c.wxm */

print(" from the cartesian expression of a vector pass to the cylindrical representation")$
/* numerical
fpprintprec:5$
ratprint : false$
A[x] :5.0 ;
A[y] : 3.14/4.0 ;
A[z] :4.6 ;
phi : atan2(A[y],A[x]);
*/

/* for unit vectors the following holds:
A[x] : x[0];
A[y] : y[0];
A[z] : z[0];
*/

print(" Transformation matrix r2cmat")$
r2cmat : matrix([cos(phi),sin(phi),0],[-sin(phi),cos(phi),0],[0,0,1]);

print(" vector A in rectangular coordinates")$
Avect : [A[x],A[y],A[z]];

print(" rectangular to cylindrical")$
Aspher : r2cmat . Avect ;

print("bye")$
/* [wxMaxima: input end ] */

/* Maxima can't load/batch files which end with a comment! */
"Created with wxMaxima"$
```

Summary of vector transformations: v c2r

```
/* [wxMaxima batch file version 1] [ DO NOT EDIT BY HAND! ]*/
/* [ Created with wxMaxima version 11.08.0 ] */

/* [wxMaxima: input start ] */
kill(all)$
load(vect)$
/* V_r2c.wxm */

print(" from the cylindrical expression of a vector pass to the cartesian representation")$
/* numerical */
fpprintprec:5$
ratprint : false$
Arho :1.0 ;
Aphi : 3.14/4.0 ;
Az : 1.0 ;
print("in order to evaluate phi we need to know where the vector is to be evaluated!!!")$
x :1.0; y :1.0 ; phi : float(atan2(y,x));

/* for unit vectors the following holds:
A%rho : %rho[0];
Aphi : phi[0];
Az : z[0];
*/

print(" Transformation matrix r2cmat")$
r2cmat : matrix([cos(phi),sin(phi),0],[-sin(phi),cos(phi),0],[0,0,1])$
c2rmat:trigsimp(invert(r2cmat));
c2rmat:float(c2rmat);

print("vector A in cylindrical coordinates")$
Acyl : [Arho,Aphi,Az];

print("cylindrical to rectangular")$
Arect : c2rmat . Acyl ;

print("bye")$
/* [wxMaxima: input end ] */

/* Maxima can't load/batch files which end with a comment! */
"Created with wxMaxima"$
```

Summary of vector transformations: v r2s

```
/* [wxMaxima batch file version 1] [ DO NOT EDIT BY HAND! ]*/
/* [ Created with wxMaxima version 11.08.0 ] */

/* [wxMaxima: input  start ] */
kill(all)$
load(vect)$
/* V_c2s.wxm */

print(" from the cartesian expression of a vector express the spherical representation")$
/* numerical
fpprintprec:5$
ratprint : false$
Ax :1.0 ;
Ay : 0.0 ;
Az : 0.0 ;
print(" coordinates where the vector is evaluated")$
Px: 0.0;
Py: 1.0;
Pz: 0.0;
phi : float(atan2(Py,Px));
theta : float(atan2(sqrt(Px^2+Py^2),Pz));
*/

/* for unit vectors the following holds:
Ax : x0;
Ay : y0;
Az : z0;
*/

print(" Transformation matrix s2rmat")$
r2smat : matrix([sin(theta)* cos(phi), sin(theta)*sin(phi),cos(theta)],
               [cos(theta)*cos(phi), cos(theta)*sin(phi),-sin(theta)],
               [-sin(phi),cos(phi),0]);

print(" vector A in rectangular coordinates")$
Arect : [Ax,Ay,Az];

print(" rectangular to spherical")$
Aspher : r2smat . Arect ;

print("bye")$
/* [wxMaxima: input  end ] */

/* Maxima can't load/batch files which end with a comment! */
"Created with wxMaxima"$
```

Summary of vector transformations: v s2r

```
/* wxMaxima batch file version 1 [ DO NOT EDIT BY HAND! ] */
/* [ Created with wxMaxima version 11.08.0 ] */

/* wxMaxima: input  start ] */
kill(all)$
load(vect)$
/* V.s2r.wom */

print("From the spherical expression of a vector express the cartesian representation")$
/* numerical
fpprintprec:5$
ratprint : false$
Ar :1.0 ;
Atheta : 0.0 ;
Aphi : 0.0 ;
print("coordinates where the vector is evaluated")$
Px: 0.0;
Py: 1.0;
Pz: 0.0;
phi : float(atan2(Py,Px));
theta : float(atan2(sqrt(Px^2+Py^2),Pz));
*/

print("Transformation matrix s2rmat")$
r2smat : matrix([sin(theta)* cos(phi), sin(theta)*sin(phi), cos(theta)],
               [cos(theta)*cos(phi), cos(theta)*sin(phi), -sin(theta)],
               [-sin(phi), cos(phi), 0])$
s2rmat : trigsimp(invert(r2smat));

print("vector A in spherical coordinates")$
Aspher : [Ar,Atheta,Aphi];

print("spherical to rectangular")$
Arect : s2rmat . Aspher ;

print("bye")$
/* wxMaxima: input  end ] */

/* Maxima can't load/batch files which end with a comment! */
"Created with wxMaxima"$
```

Summary of vector transformations: v c2s

```
/* [wxMaxima batch file version 1] [ DO NOT EDIT BY HAND! ]*/
/* [ Created with wxMaxima version 11.08.0 ] */

/* [wxMaxima: input start ] */
kill(all)$
load(vect)$
/* V_c2s.wxm */

print("From the cylindrical representation of a vector express the spherical one")$
/* numerical
fpprintprec:5$
ratprint : false$
Arho :0.0 ;
Aphi : 0.0 ;
Az : 1.0 ;
print(" coordinates where the vector is evaluated")$
Px: 0.0;
Py: 1.0;
Pz: 0.0;
phi : float(atan2(Py,Px));
theta : float(atan2(sqrt(Px^2+Py^2),Pz));
*/

print(" Transformation matrix c2smat")$
c2smat : matrix([sin(theta),0,cos(theta)],
               [cos(theta),0,-sin(theta)],
               [0,1,0]);

print(" vector A in cyl coordinates")$
Acyl : [Arho,Aphi,Az];

print(" cylindrical to spherical")$
Aspher : c2smat . Acyl ;

print("bye")$
/* [wxMaxima: input end ] */

/* Maxima can't load/batch files which end with a comment! */
"Created with wxMaxima"$
```

Summary of vector transformations: v s2c

```
/* [wxMaxima batch file version 1] [ DO NOT EDIT BY HAND! ]*/
/* [ Created with wxMaxima version 11.08.0 ] */

/* [wxMaxima: input  start ] */
kill(all)$
load(vect)$
/* V_c2s.wom */

print("From the spherical representation of a vector express the cylindrical one")$
/* numerical */
fpprintprec:5$
ratprint : false$
Ar :0.0 ;
Atheta : -1.0 ;
Aphi : 1.0 ;
print("coordinates where the vector is evaluated")$
Px: 0.0;
Py: 1.0;
Pz: 0.0;
phi : float(atan2(Py,Px));
theta : float(atan2(sqrt(Px^2+Py^2),Pz));

print(" Transformation matrix c2smat")$
c2smat : matrix([sin(theta),0,cos(theta)],
               [cos(theta),0,-sin(theta)],
               [0,1,0])$
s2cmat : trigsimp(invert(c2smat));

print("vector A in spher coordinates")$
Aspher : [Ar,Atheta,Aphi];

print("spherical to cylindrical")$
Acyl : s2cmat . Aspher ;
/* */
print("bye")$
/* [wxMaxima: input  end ] */

/* Maxima can't load/batch files which end with a comment! */
"Created with wxMaxima"$
```

Pauli matrices in Cylindrical coordinates

Pauli matrices in Cylindrical coordinates

We have already introduced the Pauli matrices for the rectangular coordinate system σ_i , with $i = 0, \dots, 3$. When expressing a vector can identify σ_1 with \mathbf{u}_x , σ_2 with \mathbf{u}_y and σ_3 with \mathbf{u}_z . By using (8) we can write:

$$\begin{aligned}\mathbf{u}_\rho &= \mathbf{u}_x \cos \phi + \mathbf{u}_y \sin \phi \\ \sigma_\rho &= \cos \phi \sigma_1 + \sin \phi \sigma_2 \\ &= \begin{pmatrix} 0 & e^{-i\phi} \\ e^{i\phi} & 0 \end{pmatrix} .\end{aligned}\tag{16}$$

Pauli matrices in Cylindrical coordinates

We have already introduced the Pauli matrices for the rectangular coordinate system σ_i , with $i = 0, \dots, 3$. When expressing a vector can identify σ_1 with \mathbf{u}_x , σ_2 with \mathbf{u}_y and σ_3 with \mathbf{u}_z . By using (8) we can write:

$$\begin{aligned}\mathbf{u}_\rho &= \mathbf{u}_x \cos \phi + \mathbf{u}_y \sin \phi \\ \sigma_\rho &= \cos \phi \sigma_1 + \sin \phi \sigma_2 \\ &= \begin{pmatrix} 0 & e^{-i\phi} \\ e^{i\phi} & 0 \end{pmatrix}.\end{aligned}\tag{16}$$

Similarly, for the unit vector along ϕ we can write

$$\begin{aligned}\mathbf{u}_\phi &= -\mathbf{u}_x \sin \phi + \mathbf{u}_y \cos \phi \\ \sigma_\phi &= -\sin \phi \sigma_1 + \cos \phi \sigma_2 \\ &= \begin{pmatrix} 0 & -i e^{-i\phi} \\ i e^{i\phi} & 0 \end{pmatrix}.\end{aligned}\tag{17}$$

Pauli matrices in Cylindrical coordinates

Naturally, for the z component we have that $\sigma_z = \sigma_3$. Summarizing, we have the following three matrices in cylindrical coordinates

$$\begin{aligned}\sigma_\rho &= \begin{pmatrix} 0 & e^{-i\phi} \\ e^{i\phi} & 0 \end{pmatrix} \\ \sigma_\phi &= \begin{pmatrix} 0 & -ie^{-i\phi} \\ ie^{i\phi} & 0 \end{pmatrix} \\ \sigma_z &= \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} .\end{aligned}\tag{18}$$

Pauli matrices in Cylindrical coordinates

It is readily proved that the matrices $\sigma_\rho, \sigma_\phi, \sigma_z$ when multiplied by themselves give the identity matrix σ_0 , their trace is null, their determinant is always -1 and their dot product is zero if they are not the same.

Pauli matrices in Cylindrical coordinates

It is readily proved that the matrices $\sigma_\rho, \sigma_\phi, \sigma_z$ when multiplied by themselves give the identity matrix σ_0 , their trace is null, their determinant is always -1 and their dot product is zero if they are not the same. In addition we have

$$\begin{aligned}i \sigma_z &= \sigma_\rho \sigma_\phi \\i \sigma_\rho &= \sigma_\phi \sigma_z \\i \sigma_\phi &= \sigma_z \sigma_\rho .\end{aligned}\tag{19}$$

Pauli matrices in Cylindrical coordinates

It is readily proved that the matrices $\sigma_\rho, \sigma_\phi, \sigma_z$ when multiplied by themselves give the identity matrix σ_0 , their trace is null, their determinant is always -1 and their dot product is zero if they are not the same. In addition we have

$$\begin{aligned}i \sigma_z &= \sigma_\rho \sigma_\phi \\i \sigma_\rho &= \sigma_\phi \sigma_z \\i \sigma_\phi &= \sigma_z \sigma_\rho .\end{aligned}\tag{19}$$

Therefore the generic vector \mathbf{A} can be expressed in cylindrical coordinates in terms of Pauli matrices as

$$\begin{aligned}\tilde{A} &= A_\rho \sigma_\rho + A_\phi \sigma_\phi + A_z \sigma_z \\&= \begin{pmatrix} A_z & e^{-i\phi} (A_\rho - i A_\phi) \\ e^{i\phi} (A_\rho + i A_\phi) & -A_z \end{pmatrix} .\end{aligned}\tag{20}$$

Pauli matrices in Cylindrical coordinates

It is noted that since the vector is the same when expressed in rectangular or cylindrical components we have the identity

$$A_\rho \sigma_\rho + A_\phi \sigma_\phi + A_z \sigma_z = A_x \sigma_1 + A_y \sigma_2 + A_z \sigma_3 . \quad (21)$$

By performing the dot product of the above expression with a selected sigma matrix we can recover the desired field component in terms of the other basis.

Pauli matrices in Cylindrical coordinates

As an example let us consider to retrieve the expression of A_ρ in terms of the rectangular components.

Pauli matrices in Cylindrical coordinates

As an example let us consider to retrieve the expression of A_ρ in terms of the rectangular components. It is sufficient to dot multiply both sides of (27) times σ_ρ to retrieve

$$\begin{aligned} A_\rho &= A_x \sigma_1 \cdot \sigma_\rho + A_y \sigma_2 \cdot \sigma_\rho + A_z \sigma_3 \cdot \sigma_\rho \\ &= A_x \cos \phi + A_y \sin \phi \end{aligned} \tag{22}$$

where we remind that performing the dot product is equivalent to perform the matrix product and take half of the trace. It is left as an exercise for the reader to recover the results in the previous section by using Pauli matrices multiplication.

Pauli matrices in Cylindrical coordinates

[illegible]

Pauli matrices in Spherical coordinates

Pauli matrices in Spherical coordinates

So far we have introduced the Pauli matrices for the rectangular and cylindrical coordinate systems. We can proceed in the same way that we have followed for the cylindrical coordinate system.

Pauli matrices in Spherical coordinates

So far we have introduced the Pauli matrices for the rectangular and cylindrical coordinate systems. We can proceed in the same way that we have followed for the cylindrical coordinate system. By using (9) we can write:

$$\begin{aligned}\mathbf{u}_r &= \mathbf{u}_x \sin \theta \cos \phi + \mathbf{u}_y \sin \theta \sin \phi + \mathbf{u}_z \cos \theta \\ \sigma_r &= \sin \theta \cos \phi \sigma_1 + \sin \theta \sin \phi \sigma_2 + \cos \theta \sigma_3 \\ &= \begin{pmatrix} \cos \theta & e^{-i\phi} \sin \theta \\ e^{i\phi} \sin \theta & -\cos \theta \end{pmatrix} .\end{aligned}\tag{23}$$

Pauli matrices in Spherical coordinates

Similarly, for the unit vector along θ we can write:

$$\begin{aligned}\mathbf{u}_\theta &= \mathbf{u}_x \cos \theta \cos \phi + \mathbf{u}_y \cos \theta \sin \phi - \mathbf{u}_z \sin \theta \\ \sigma_\theta &= \cos \theta \cos \phi \sigma_1 + \cos \theta \sin \phi \sigma_2 - \sin \theta \sigma_3 \\ &= \begin{pmatrix} -\sin \theta & e^{-i\phi} \cos \theta \\ e^{i\phi} \cos \theta & \sin \theta \end{pmatrix} .\end{aligned}\tag{24}$$

Pauli matrices in Spherical coordinates

Similarly, for the unit vector along θ we can write:

$$\begin{aligned}\mathbf{u}_\theta &= \mathbf{u}_x \cos \theta \cos \phi + \mathbf{u}_y \cos \theta \sin \phi - \mathbf{u}_z \sin \theta \\ \sigma_\theta &= \cos \theta \cos \phi \sigma_1 + \cos \theta \sin \phi \sigma_2 - \sin \theta \sigma_3 \\ &= \begin{pmatrix} -\sin \theta & e^{-i\phi} \cos \theta \\ e^{i\phi} \cos \theta & \sin \theta \end{pmatrix} .\end{aligned}\tag{24}$$

Not surprisingly for the ϕ coordinate the result is the same as in cylindrical coordinates.

Pauli matrices in Spherical coordinates

Summarizing, we have the following three matrices in spherical coordinates

$$\begin{aligned}\sigma_r &= \begin{pmatrix} \cos \theta & e^{-i\phi} \sin \theta \\ e^{i\phi} \sin \theta & -\cos \theta \end{pmatrix} \\ \sigma_\theta &= \begin{pmatrix} -\sin \theta & e^{-i\phi} \cos \theta \\ e^{i\phi} \cos \theta & \sin \theta \end{pmatrix} \\ \sigma_\phi &= \begin{pmatrix} 0 & -i e^{-i\phi} \\ i e^{i\phi} & 0 \end{pmatrix} .\end{aligned}\tag{25}$$

Pauli matrices in Spherical coordinates

It is readily proved that the matrices $\sigma_r, \sigma_\theta, \sigma_\phi$ when multiplied by themselves give the identity matrix σ_0 , their trace is null, their determinant is always -1 and their dot product is zero if they are not the same. In addition we have:

$$\begin{aligned}i \sigma_\phi &= \sigma_r \sigma_\theta \\i \sigma_r &= \sigma_\theta \sigma_\phi \\i \sigma_\theta &= \sigma_\phi \sigma_r .\end{aligned}\tag{26}$$

Pauli matrices in Spherical coordinates

Therefore, the generic vector \mathbf{A} can be expressed in spherical coordinates in terms of Pauli matrices as:

$$\begin{aligned}\tilde{A} &= A_r \sigma_r + A_\theta \sigma_\theta + A_\phi \sigma_\phi \\ &= \begin{pmatrix} A_r \cos \theta - A_\theta \sin \theta & e^{-i\phi} (A_r \sin \theta - i A_\phi + A_\theta \cos \theta) \\ e^{i\phi} (A_r \sin \theta + i A_\phi + A_\theta \cos \theta) & -A_r \cos \theta + A_\theta \sin \theta \end{pmatrix} \quad (27)\end{aligned}$$

Pauli matrices in Spherical coordinates

It is noted that since the vector is the same when expressed in rectangular, cylindrical or spherical components we have the identity

$$\begin{aligned}\tilde{A} &= A_x\sigma_1 + A_y\sigma_2 + A_z\sigma_3 \\ &= A_\rho\sigma_\rho + A_\phi\sigma_\phi + A_z\sigma_z \\ &= A_r\sigma_r + A_\theta\sigma_\theta + A_\phi\sigma_\phi.\end{aligned}\tag{28}$$

Pauli matrices in Spherical coordinates

It is noted that since the vector is the same when expressed in rectangular, cylindrical or spherical components we have the identity

$$\begin{aligned}\tilde{A} &= A_x\sigma_1 + A_y\sigma_2 + A_z\sigma_3 \\ &= A_\rho\sigma_\rho + A_\phi\sigma_\phi + A_z\sigma_z \\ &= A_r\sigma_r + A_\theta\sigma_\theta + A_\phi\sigma_\phi.\end{aligned}\tag{28}$$

By performing the dot product of the above expression with a selected sigma matrix we can recover the desired field component in terms of the other basis.

Thus all the transformation between vectors in different coordinate systems can be simply obtained by matrix multiplication and trace operation.

Pauli matrices in Spherical coordinates

[illegible]

Pauli matrices in Spherical coordinates

An example of application is provided with the following theorem

[illegible]