



Nabla operator with Pauli matrices

Mauro Mongiardo¹

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

Table of contents

1. Vectors
2. Gradient
3. Nabla definition
4. Nabla operator in rectangular coordinates
5. Nabla operator in circular cylindrical coordinates using Pauli matrices
6. Nabla operator in spherical coordinates using Pauli matrices
7. Relating vector algebra to conventional vector analysis

Vectors

Vectors are generally represented in the various coordinate system as

$$\begin{aligned}\mathbf{A} &= A_x \mathbf{x}_0 + A_y \mathbf{y}_0 + A_z \mathbf{z}_0 \\ &= A_\rho \boldsymbol{\rho}_0 + A_\phi \boldsymbol{\phi}_0 + A_z \mathbf{z}_0 \\ &= A_r \mathbf{r}_0 + A_\theta \boldsymbol{\theta}_0 + A_\phi \boldsymbol{\phi}_0\end{aligned}\tag{1}$$

Let us now represent them with the following form

$$\mathbf{A} = A_1 \mathbf{e}_1 + A_2 \mathbf{e}_2 + A_3 \mathbf{e}_3\tag{2}$$

and let us impose that for $i = 1, 2, 3$

$$(\mathbf{e}_i)^2 = 1\tag{3}$$

and that for $i \neq j$

$$\mathbf{e}_i \mathbf{e}_j = -\mathbf{e}_j \mathbf{e}_i\tag{4}$$

These two conditions are sufficient to establish a Clifford algebra $Cl(3)$.

Disadvantages of the vector representation with versors

- Vectors cannot be multiplied
- it is not possible to find the inverse
- given two vectors **a** and **b** it is not possible to find the transformation which leads from **a** to **b**
- the cross product is misleading; instead, a bivector which is an oriented surface should be used.
- no volume elements are present.
- As we have seen the representation of a vector with Pauli matrices is the same in all coordinate systems, while with versors we only have the components.

Relation with Pauli matrices

By using the correspondence in Table 9 the conditions (3) and (4) are realized.

Table 1: Basis vector in rectangular, cylindrical and spherical coordinate systems. The \mathbf{e}_i are a Clifford basis and correspond to the appropriate Pauli matrices.

	\mathbf{e}_1	\mathbf{e}_2	\mathbf{e}_3
rectangular	σ_1	σ_2	σ_3
cylindrical	σ_ρ	σ_ϕ	σ_3
spherical	σ_r	σ_θ	σ_ϕ

Table 2: Vector \mathbf{A} equivalence for rectangular, cylindrical and spherical coordinate systems.

	A_1	A_2	A_3
rectangular	A_x	A_y	A_z
cylindrical	A_ρ	A_ϕ	A_z
spherical	A_r	A_θ	A_ϕ

Pauli matrices in Rectangular coordinates

$$\begin{aligned}\sigma_1 &= \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \\ \sigma_2 &= \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix} \\ \sigma_3 &= \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} .\end{aligned}\tag{5}$$

Pauli matrices in Cylindrical coordinates

$$\begin{aligned}\sigma_\rho &= \begin{pmatrix} 0 & e^{-i\phi} \\ e^{i\phi} & 0 \end{pmatrix} \\ \sigma_\phi &= \begin{pmatrix} 0 & -ie^{-i\phi} \\ ie^{i\phi} & 0 \end{pmatrix} \\ \sigma_3 &= \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} .\end{aligned}\tag{6}$$

Attention! The Pauli matrices in Cylindrical coordinates depend on ϕ !

Pauli matrices in Spherical coordinates

We have the following three matrices in spherical coordinates

$$\begin{aligned}\sigma_r &= \begin{pmatrix} \cos \theta & e^{-i\phi} \sin \theta \\ e^{i\phi} \sin \theta & -\cos \theta \end{pmatrix} \\ \sigma_\theta &= \begin{pmatrix} -\sin \theta & e^{-i\phi} \cos \theta \\ e^{i\phi} \cos \theta & \sin \theta \end{pmatrix} \\ \sigma_\phi &= \begin{pmatrix} 0 & -i e^{-i\phi} \\ i e^{i\phi} & 0 \end{pmatrix} .\end{aligned}\tag{7}$$

Attention! The Pauli matrices in spherical coordinates depend on θ, ϕ !

Summary of Pauli matrices

Table 3: Basis vector in rectangular, cylindrical and spherical coordinate systems. The \mathbf{e}_i are a Clifford basis and correspond to the appropriate Pauli matrices.

	\mathbf{e}_1	\mathbf{e}_2	\mathbf{e}_3
rectangular	$\sigma_1 = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$	$\sigma_2 = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}$	$\sigma_3 = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$
cylindrical	$\sigma_\rho = \begin{pmatrix} 0 & e^{-i\phi} \\ e^{i\phi} & 0 \end{pmatrix}$	$\sigma_\phi = \begin{pmatrix} 0 & -i e^{-i\phi} \\ i e^{i\phi} & 0 \end{pmatrix}$	$\sigma_3 = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$
spherical	$\sigma_r = \begin{pmatrix} \cos\theta & e^{-i\phi} \sin\theta \\ e^{i\phi} \sin\theta & -\cos\theta \end{pmatrix}$	$\sigma_\theta = \begin{pmatrix} -\sin\theta & e^{-i\phi} \cos\theta \\ e^{i\phi} \cos\theta & \sin\theta \end{pmatrix}$	$\sigma_\phi = \begin{pmatrix} 0 & -i e^{-i\phi} \\ i e^{i\phi} & 0 \end{pmatrix}$

Gradient

Gradient

Let us consider a scalar function $\phi(\mathbf{r})$, which is single valued and continuous in a volume V . Physically, the function ϕ may represent an electric potential, a temperature, etc.

At a point P the function will take the value ϕ_P . Now suppose to draw a sphere of radius Δs centered in P . We can check the values of $\phi(\mathbf{r})$ on this sphere and, in general, there will be a point Q on which the variation $\Delta\phi = \phi_P - \phi_Q$ is maximum.

This defines also the direction from P to Q . This direction is taken as the direction of a new vector which is called the **gradient**. The magnitude of the gradient is defined as the value $\Delta\phi/\Delta s$ in this preferred direction.

Thus the gradient of a scalar may be defined as

$$\text{grad}\phi = \nabla\phi = \mathbf{a}_{\max} \lim_{\Delta s \rightarrow 0} \left(\frac{\Delta\phi}{\Delta s} \right)_{\max} \quad (8)$$

where \mathbf{a}_{\max} is a unit vector pointing in the direction of maximum $\frac{\Delta\phi}{\Delta s}$.

Gradient expression

It is noted that the definition (8) is not dependent on a particular coordinate system.

In **rectangular coordinates**, (8) reduces to

$$\nabla\phi = \mathbf{a}_x \frac{\partial\phi}{\partial x} + \mathbf{a}_y \frac{\partial\phi}{\partial y} + \mathbf{a}_z \frac{\partial\phi}{\partial z} . \quad (9)$$

In **circular-cylinder coordinates**, Δs in the angular direction ψ is not equal to $\Delta\psi$ but $\delta s = \rho\Delta\psi$. Thus we have

$$\nabla\phi = \mathbf{a}_\rho \frac{\partial\phi}{\partial\rho} + \frac{\mathbf{a}_\psi}{\rho} \frac{\partial\phi}{\partial\psi} + \mathbf{a}_z \frac{\partial\phi}{\partial z} . \quad (10)$$

Similarly, in **spherical coordinates**,

$$\nabla\phi = \mathbf{a}_r \frac{\partial\phi}{\partial r} + \frac{\mathbf{a}_\psi}{r \sin\theta} \frac{\partial\phi}{\partial\psi} + \frac{\mathbf{a}_\theta}{r} \frac{\partial\phi}{\partial\theta} . \quad (11)$$

Nabla definition

Nabla definition

The nabla operator is defined as

$$\nabla = \sum_{i=1}^3 c_i \mathbf{e}_i \partial_i \quad (12)$$

where we have introduced appropriate scaling coefficients c_i , a Clifford vector base composed by \mathbf{e}_i , and the partial derivatives ∂_i .

Note that the \mathbf{e}_i are a *Clifford basis* and they satisfy the properties of a Clifford basis, i.e. the $\mathbf{e}_i \mathbf{e}_i = 1$ and $\mathbf{e}_i \mathbf{e}_j = -\mathbf{e}_j \mathbf{e}_i$ for $i \neq j$. Depending on the coordinate system we will identify the basis vector as reported in Table 4.

Table 4: Basis vector in rectangular, cylindrical and spherical coordinate systems. The \mathbf{e}_i are a Clifford basis and correspond to the appropriate Pauli matrices.

	\mathbf{e}_1	\mathbf{e}_2	\mathbf{e}_3
rectangular	σ_1	σ_2	σ_3
cylindrical	σ_ρ	σ_ϕ	σ_3
spherical	σ_r	σ_θ	σ_ϕ

Nabla definition

The scaling coefficients are reported in Table 5. The partial derivatives symbols ∂_i assume the meaning reported in Table 6

Table 5: Scaling coefficients in rectangular, cylindrical and spherical coordinate systems.

	c_1	c_2	c_3
rectangular	1	1	1
cylindrical	1	$1/\rho$	1
spherical	1	$1/r$	$1/(r \sin(\theta))$

Table 6: Partial derivatives for rectangular, cylindrical and spherical coordinate systems.

	∂_1	∂_2	∂_3
rectangular	∂_x	∂_y	∂_z
cylindrical	∂_ρ	∂_ϕ	∂_z
spherical	∂_r	∂_θ	∂_ϕ

Fundamental identity (the geometric product)

From the fundamental identity (the geometric product) we have:

$$\nabla \mathbf{F} = \nabla \cdot \mathbf{F} + \nabla \wedge \mathbf{F}. \quad (13)$$

The term \mathbf{F} can be a multivector composed, in general, by a scalar part (grade 0), a vector part (grade 1), a bivector part (grade 2) and a trivector or pseudo scalar (grade 3). In Table 7 it is reported the application of the nabla operator to a scalar function.

Table 7: For a scalar function ϕ application of the nabla operator gives a vector (i.e. the gradient), here assumed equal to the external product of ∇ and ϕ . Note that in the geometric product expansion of ϕ also a term of the type $\nabla \cdot \phi$ is present, but this term is equal to zero. Further application of the nabla operator provides the Laplacian (which is of 0 grade) and the term $\nabla \wedge \nabla \wedge \phi = 0$.

Grade	0	1	2	3
quantity	ϕ			
$\nabla \phi$	0	$\nabla \wedge \phi = \nabla \phi$		
$\nabla \nabla \phi = \nabla^2 \phi$	$\nabla \cdot \nabla \phi$		$\nabla \wedge \nabla \wedge \phi = 0$	

Grade structure

In Table 8 the nabla operator has been applied to a vector function.

Table 8: For a vector function \mathbf{A} application of the nabla operator gives a scalar (i.e. the divergence), and a bivector. From $\nabla \cdot \mathbf{A}$ a further application of the nabla operator gives the vector term $\nabla \nabla \cdot \mathbf{A}$, while the nabla operator applied to $\nabla \wedge \mathbf{A}$ gives the term $\nabla \cdot \nabla \wedge \mathbf{A}$ and the term $\nabla \wedge \nabla \wedge \mathbf{A}$ which is equal to zero.

Grade	0	1	2	3
quantity		\mathbf{A}		
$\nabla \mathbf{A}$	$\nabla \cdot \mathbf{A}$		$\nabla \wedge \mathbf{A}$	
$\nabla \nabla \mathbf{A} = \nabla^2 \mathbf{A}$		$\nabla \nabla \cdot \mathbf{A} + \nabla \cdot \nabla \wedge \mathbf{A}$		$\nabla \wedge \nabla \wedge \mathbf{A} = 0$

It is noted that by substituting the vector \mathbf{A} with $i\mathbf{B}$ we have the corresponding table for nabla operating on a bivector. Similarly, by substituting ϕ with $i\psi$ we have nabla operating on a pseudoscalar.

Nabla operator in rectangular coordinates

Nabla operator in rectangular coordinates

By using Pauli matrices a field vector \mathbf{F} may be written as

$$\begin{aligned}\tilde{\mathbf{F}} &= \sigma_1 F_x + \sigma_2 F_y + \sigma_3 F_z \\ &= \begin{pmatrix} F_z & F_x - i F_y \\ i F_y + F_x & -F_z \end{pmatrix}\end{aligned}\tag{14}$$

Similarly, the Pauli matrix representation of the ∇ operator takes the form

$$\begin{aligned}\tilde{\nabla} &= \sigma_1 \partial_x + \sigma_2 \partial_y + \sigma_3 \partial_z \\ &= \begin{pmatrix} \partial_z & \partial_x - i \partial_y \\ i \partial_y + \partial_x & -\partial_z \end{pmatrix}.\end{aligned}\tag{15}$$

From the fundamental identity (the geometric product) we have:

$$\nabla \mathbf{F} = \nabla \cdot \mathbf{F} + \nabla \wedge \mathbf{F}.\tag{16}$$

Matrix product evaluation

When evaluating $\nabla \mathbf{F}$ via Pauli matrices we simply need to perform the following matrix product:

$$\begin{aligned}\tilde{\nabla} \tilde{\mathbf{F}} &= \begin{pmatrix} \partial_z & \partial_x - i \partial_y \\ i \partial_y + \partial_x & -\partial_z \end{pmatrix} \begin{pmatrix} F_z & F_x - i F_y \\ i F_y + F_x & -F_z \end{pmatrix} \\ &= \begin{pmatrix} \partial_z F_z + \partial_y F_y + \partial_x F_x & 0 \\ 0 & \partial_z F_z + \partial_y F_y + \partial_x F_x \end{pmatrix} + \\ &+ \begin{pmatrix} i (\partial_x F_y - \partial_y F_x) & i (\partial_y F_z) - \partial_x F_z - i (\partial_z F_y) + \partial_z F_x \\ i (\partial_y F_z) + \partial_x F_z - i (\partial_z F_y) - \partial_z F_x & -i (\partial_x F_y - \partial_y F_x) \end{pmatrix}. \quad (17)\end{aligned}$$

In (17) we have separated the scalar part corresponding to $\nabla \cdot \mathbf{F}$ (diagonal matrix) from the external product $\nabla \wedge \mathbf{F}$.

In the following code it is shown how to compute, for a vector \mathbf{E} , the two geometric products $\nabla \mathbf{E}$ and $\nabla \nabla \mathbf{E}$.

```

/* [Mathematica kernel file version 0] [ DO NOT EDIT BY HAND! ] v/
/* Created with notebook version 11.00.0 ] v/

/* [Mathematica: input start ] v/
k[[1]]={1}];

dependencies[{{a11,a21,a22},{x,y,z}}];
Ap = matrix[{{a11,a21},{a11,a22}}];

/*
The block Mathematica perform the table operation
in rectangular coordinates on a Pauli matrix Ap
*/

Mathematica[Ap] :=
block[{}];

print["*****"];
print["*****"];
print["Math: in="];

M0a11 = d100[Ap[1,1]]-x+d100[Ap[2,1]]-y;
M0a12 = d100[Ap[1,2]]-x+d100[Ap[2,2]]-y;
M0a21 = -d100[Ap[2,1]]-x+d100[Ap[1,1]]-y;
M0a22 = -d100[Ap[2,2]]-x+d100[Ap[1,2]]-y;
M0a23 = -d100[Ap[2,2]]-x+d100[Ap[1,2]]-y;
M0a24 = -d100[Ap[2,2]]-x+d100[Ap[1,2]]-y;

M0p = factor[matrix[{{M0a11,M0a12},{M0a21,M0a22}}]]
];

print["*****"];
print["Consider a vector R with components Rx, Ry, Rz"];
dependencies[{{Rx,Ry,Rz},{x,y,z}}];

/*
print["Consider an Element Field of the form"]R
Rx = "a x" + b x + c] R
Ry = R0
Rz = R0
*/
{{Rx,Ry,Rz}};

N0a0a[0] = matrix[{{1,0},{0,1}}];
N0a0a[1] = matrix[{{0,1},{1,0}}];
N0a0a[2] = matrix[{{0,1},{1,0}}];
N0a0a[3] = matrix[{{0,1},{1,0}}];

print["Math: as Pauli matrix in rectangular coordinates"]R
Np = do + N0a0a[1] + dy + N0a0a[2] + dz + N0a0a[3];

print["vector R as Pauli matrix"]R
Rp = Rx + N0a0a[1] + Ry + N0a0a[2] + Rz + N0a0a[3];

print["table R in"]R
M0p = Mathematica[Rp];

print["table table R in"]R
M0Mp = Mathematica[M0p];

Gradway[0] :=
block[{}];

print["*****"];
print["separate the components"];
print["*****"];
M0 = realpart[M0];
M0 = imagpart[M0];
/*
print["real part"]M0;
print["imaginary part"]M0;
*/
print["Grade 0"];
a0b = rational[1/2*(M0[1,1]+M0[2,2])];
print["a0b = ",a0b];
print["Grade 1"];
a1b = rational[1/2*(M0[1,2]+M0[2,1])];
print["a1b = ",a1b];
a2b = rational[1/2*(M0[2,1]-M0[1,2])];
print["a2b = ",a2b];
a3b = rational[1/2*(M0[2,2]-M0[1,1])];
print["a3b = ",a3b];

print["Grade 2"];
R1a = rational[1/2*(M0[1,2]+M0[2,1])];
print["R1a = ",R1a];
R2a = rational[1/2*(M0[1,2]-M0[2,1])];
print["R2a = ",R2a];
R3a = rational[1/2*(M0[1,2]-M0[2,1])];
print["R3a = ",R3a];
a4b = rational[1/2*(M0[1,1]+M0[2,2])];
print["a4b = ",a4b];

print["Grade 3"];
];

print["Finding the grade elements of the following Pauli matrix"]R
M = M0p;
print["My calling Gradway[0] on how"]R
Gradway[0];

print["Finding the grade elements of the following Pauli matrix"]R
M = M0Mp;
print["My calling Gradway[0] on how"]R
Gradway[0];

print["end"]R
/* [Mathematica: input end ] v/
/* Mathematica can't load/kash files which end with a comment */
"Created with notebook 0

```

Instead of programming several times the nabla operation it is more convenient to use the block function.

We have one block called Nablarect that performs the nabla operator on a given Pauli matrix.

Then we have another block, called Gradesep, which for a given Pauli matrix separate the different grades.

Second order expressions

In several instances it is necessary to form the second order expressions, e.g. $\nabla\nabla\mathbf{F}$. Computation of this quantity via Pauli matrices is embarrassing simple, since only matrix multiplication is required

$$\begin{aligned}\tilde{\nabla}\tilde{\nabla}\tilde{\mathbf{F}} &= \begin{pmatrix} \partial_z & \partial_x - i\partial_y \\ i\partial_y + \partial_x & -\partial_z \end{pmatrix} \begin{pmatrix} \partial_z & \partial_x - i\partial_y \\ i\partial_y + \partial_x & -\partial_z \end{pmatrix} \begin{pmatrix} F_z & F_x - iF_y \\ iF_y + F_x & -F_z \end{pmatrix} \\ &= \begin{pmatrix} \partial_z^2 + (\partial_x - i\partial_y)(i\partial_y + \partial_x) & 0 \\ 0 & \partial_z^2 + (\partial_x - i\partial_y)(i\partial_y + \partial_x) \end{pmatrix} \begin{pmatrix} F_z & F_x - iF_y \\ iF_y + F_x & -F_z \end{pmatrix} \\ &= \begin{pmatrix} \Delta F_z & \Delta F_x - i\Delta F_y \\ i\Delta F_y + \Delta F_x & \Delta F_z \end{pmatrix} \\ &= \Delta\tilde{\mathbf{F}}\end{aligned}\tag{18}$$

where we have introduced the laplacian Δ defined as

$$\Delta = \partial_z^2 + \partial_y^2 + \partial_x^2\tag{19}$$

Nabla operator in circular cylindrical coordinates using Pauli matrices

Nabla operator in circular cylindrical coordinates

The gradient operator of a scalar function w in circular cylindrical coordinates has the following form:

$$\nabla w = \mathbf{u}_\rho \frac{\partial w}{\partial \rho} + \mathbf{u}_\phi \frac{1}{\rho} \frac{\partial w}{\partial \phi} + \mathbf{u}_z \frac{\partial w}{\partial z}, \quad (20)$$

from which we can infer the Pauli matrix representation of the nabla operator $\tilde{\nabla}$ in cylindrical coordinates.

By substituting the unit versors with the matrices in Table 4, and using the abbreviated notation e.g. $\partial_\rho = \partial/\partial\rho$, we obtain

$$\begin{aligned} \tilde{\nabla} &= \sigma_\rho \partial_\rho + \frac{1}{\rho} \sigma_\phi \partial_\phi + \sigma_z \partial_z \\ &= \begin{pmatrix} \partial_z & \frac{e^{-i\phi}}{\rho} (\rho \partial_\rho - i \partial_\phi) \\ \frac{e^{i\phi}}{\rho} (\rho \partial_\rho + i \partial_\phi) & -\partial_z \end{pmatrix} \\ &= \begin{pmatrix} \partial_z & \nabla_t^* \\ \nabla_t & -\partial_z \end{pmatrix} \end{aligned} \quad (21)$$

Nabla operator in circular cylindrical

where we have introduced the operator ∇_t and its complex conjugate ∇_t^* defined as

$$\begin{aligned}\nabla_t &= \frac{e^{i\phi}}{\rho} (\rho \partial_\rho + i \partial_\phi) \\ \nabla_t^* &= \frac{e^{-i\phi}}{\rho} (\rho \partial_\rho - i \partial_\phi) .\end{aligned}\tag{22}$$

Let us now consider a vector \mathbf{A} expressed in terms of cylindrical Pauli matrices as:

$$\tilde{A} = \begin{pmatrix} A_z & e^{-i\phi} (A_\rho - i A_\phi) \\ e^{i\phi} (A_\rho + i A_\phi) & -A_z \end{pmatrix}\tag{23}$$

and let us recall that, due to the fundamental identity of geometric algebra, we also have:

$$\nabla \mathbf{A} = \nabla \cdot \mathbf{A} + \nabla \wedge \mathbf{A} .\tag{24}$$

vector \mathbf{A} expressed in terms of cylindrical Pauli matrices

By performing the matrix multiplication of (21) with (23), and by denoting with $(\tilde{\nabla} \tilde{A})_{ij}$ the ij element of the matrix, we obtain:

$$\begin{aligned}(\tilde{\nabla} \tilde{A})_{11} &= \nabla \cdot \mathbf{A} + i \left[\frac{A_\phi}{\rho} + \frac{\partial A_\phi}{\partial \rho} - \frac{1}{\rho} \frac{\partial A_\rho}{\partial \phi} \right] \\(\tilde{\nabla} \tilde{A})_{12} &= \left[- \left(\frac{\partial A_z}{\partial \rho} - \frac{\partial A_\rho}{\partial z} \right) + i \left(\frac{1}{\rho} \frac{\partial A_z}{\partial \phi} - \frac{\partial A_\phi}{\partial z} \right) \right] e^{-i\phi} \\(\tilde{\nabla} \tilde{A})_{21} &= \left[\left(\frac{\partial A_z}{\partial \rho} - \frac{\partial A_\rho}{\partial z} \right) + i \left(\frac{1}{\rho} \frac{\partial A_z}{\partial \phi} - \frac{\partial A_\phi}{\partial z} \right) \right] e^{i\phi} \\(\tilde{\nabla} \tilde{A})_{22} &= \nabla \cdot \mathbf{A} - i \left[\frac{A_\phi}{\rho} + \frac{\partial A_\phi}{\partial \rho} - \frac{1}{\rho} \frac{\partial A_\rho}{\partial \phi} \right].\end{aligned}\tag{25}$$

In (25), the divergence term

$$\nabla \cdot \mathbf{A} = \frac{\partial A_\rho}{\partial \rho} + \frac{A_\rho}{\rho} + \frac{1}{\rho} \frac{\partial A_\phi}{\partial \phi} + \frac{\partial A_z}{\partial z}\tag{26}$$

has been singled out.

The matrix $\tilde{\nabla} \tilde{A}$

The matrix $\tilde{\nabla} \tilde{A}$ contains therefore the divergence term, which can be obtained by dividing by two the matrix trace, and contains also all the terms relative to the external product of $\tilde{\nabla}$ and \tilde{A} .

The specific component of the part corresponding to the external product, i.e. to the bivector, is obtained in the following manner.

First the matrix $\tilde{\nabla} \wedge \tilde{A}$ is obtained as

$$\tilde{\nabla} \wedge \tilde{A} = \tilde{\nabla} \tilde{A} - (\nabla \cdot \mathbf{A}) \sigma_0. \quad (27)$$

Then the component are retrieved by dot multiplication for the appropriate Pauli matrix. The dot multiplication is obtained from the multiplication of the matrices and then by taking half of the trace.

Curl cylindrical coordinates

As an example for the ρ component we have

$$\begin{aligned}(\nabla \times \mathbf{A})_{\rho} &= \frac{1}{2i} \text{trace} \left\{ \left(\tilde{\nabla} \wedge \tilde{A} \right) \sigma_{\rho} \right\} \\ &= \frac{1}{\rho} \frac{\partial A_z}{\partial \phi} - \frac{\partial A_{\phi}}{\partial z}\end{aligned}\tag{28}$$

while for the ϕ component one has

$$\begin{aligned}(\nabla \times \mathbf{A})_{\phi} &= \frac{1}{2i} \text{trace} \left\{ \left(\tilde{\nabla} \wedge \tilde{A} \right) \sigma_{\phi} \right\} \\ &= \frac{\partial A_{\rho}}{\partial z} - \frac{\partial A_z}{\partial \rho}\end{aligned}\tag{29}$$

and, finally, for the z component we obtain

$$\begin{aligned}(\nabla \times \mathbf{A})_z &= \frac{1}{2i} \text{trace} \left\{ \left(\tilde{\nabla} \wedge \tilde{A} \right) \sigma_z \right\} \\ &= \frac{A_{\phi}}{\rho} + \frac{\partial A_{\phi}}{\partial \rho} + \frac{1}{\rho} \frac{\partial A_{\rho}}{\partial \phi}.\end{aligned}\tag{30}$$

Second order operators

Further application of the nabla operator allows to obtain

$$\tilde{\nabla} \tilde{\nabla} \tilde{A} = \tilde{\nabla}^2 \tilde{A} \quad (31)$$

which, in terms of Pauli matrices, can be computed by matrix multiplication. In vector terms the second order operator gives the vector Laplacian $\nabla^2 \mathbf{A}$:

$$\nabla (\nabla \mathbf{A}) = \nabla^2 \mathbf{A} = \nabla \cdot \nabla \wedge \mathbf{A} + \nabla \wedge \nabla \cdot \mathbf{A} \quad (32)$$

which shows that the final result is a vector.

It is convenient to introduce the scalar Laplacian operator ∇_s^2 defined as:

$$\nabla_s^2 = \frac{\partial^2}{\partial \rho^2} + \frac{1}{\rho^2} \frac{\partial^2}{\partial \phi^2} + \frac{\partial^2}{\partial z^2} + \frac{1}{\rho} \frac{\partial}{\partial \rho} \quad (33)$$

Differences between the scalar Laplacian (33) and the vector Laplacian (32)

As a result of the matrix multiplication, and by identifying the various components, the following results are obtained:

$$\begin{aligned}\left(\tilde{\nabla}^2 \tilde{A}\right)_\rho &= \nabla_s^2 A_\rho - \frac{A_\rho}{\rho^2} - \frac{2}{\rho^2} \frac{\partial A_\phi}{\partial \phi} \\ \left(\tilde{\nabla}^2 \tilde{A}\right)_\phi &= \nabla_s^2 A_\phi - \frac{A_\phi}{\rho^2} + \frac{2}{\rho^2} \frac{\partial A_\rho}{\partial \phi} \\ \left(\tilde{\nabla}^2 \tilde{A}\right)_\rho &= \nabla_s^2 A_z.\end{aligned}\tag{34}$$

It is important to note the differences between the scalar Laplacian (33) and the vector Laplacian (32).

In conventional vector algebra the vector Laplacian is expressed as

$$\nabla^2 \mathbf{A} = \nabla \nabla \cdot \mathbf{A} - \nabla \times \nabla \times \mathbf{A}.\tag{35}$$

Code for Nabla operations and grade retrieval in cylindrical coordinates

A code illustrating the nabla operation and the grade extraction is shown next.

The block `Nablacyl` performs the nabla operation on a Pauli matrix.

The block `Gradecyl` perform the grade extraction on a Pauli matrix in cylindrical coordinates.

Optionally these operations are used to find the gradient of a line charge distribution and the magnetic field on an infinite current line.

In the following code the relevant quantities are computed first in terms of conventional operators.

Then they are computed in terms of Pauli matrices.

When feasible a test has been introduced showing that the results are coincident.

It is noted that so far we have considered all the components with all the spatial dependencies.

In several cases noticeable simplifications can occur.

Consider e.g. the case of infinite filament of constant current, placed in the z direction.

The potential A will be of the form $A_z(\rho)$ and therefore the problem is considerably simpler.

Nabla operator in spherical coordinates using Pauli matrices

Nabla operator in spherical coordinates using Pauli matrices

The gradient operator of a scalar function w in spherical coordinates has the following form:

$$\nabla w = \mathbf{u}_r \frac{\partial w}{\partial r} + \mathbf{u}_\theta \frac{1}{r} \frac{\partial w}{\partial \theta} + \frac{1}{r \sin \theta} \mathbf{u}_\phi \frac{\partial w}{\partial \phi}, \quad (36)$$

from which we can infer the Pauli matrix representation of the nabla operator $\tilde{\nabla}$ in spherical coordinates.

By substituting the unit versors with the matrices in Table 4, and using the abbreviated notation e.g. $\partial_r = \partial/\partial r$, we obtain

$$\begin{aligned} \tilde{\nabla} &= \sigma_r \partial_r + \frac{1}{r} \sigma_\theta \partial_\theta + \frac{1}{r \sin \theta} \sigma_\phi \partial_\phi \\ &= \begin{pmatrix} \cos \theta \partial_r - \sin \theta \partial_\theta & e^{-i\phi} (\sin \theta \partial_r + \cos \theta \partial_\theta - i \partial_\phi) \\ e^{i\phi} (\sin \theta \partial_r + \cos \theta \partial_\theta + i \partial_\phi) & -(\cos \theta \partial_r - \sin \theta \partial_\theta) \end{pmatrix} \end{aligned} \quad (37)$$

vector \mathbf{A}

Let us now consider a vector \mathbf{A} expressed in terms of spherical Pauli matrices as:

$$\tilde{A} = \begin{pmatrix} A_r \cos \theta - A_\theta \sin \theta & e^{-i\phi} (A_r \sin \theta - i A_\phi + A_\theta \cos \theta) \\ e^{i\phi} (A_r \sin \theta + i A_\phi + A_\theta \cos \theta) & -A_r \cos \theta + A_\theta \sin \theta \end{pmatrix} \quad (38)$$

and let us recall that, due to the fundamental identity of geometric algebra, we also have:

$$\nabla \mathbf{A} = \nabla \cdot \mathbf{A} + \nabla \wedge \mathbf{A}. \quad (39)$$

By performing the matrix multiplication of (37) with (38), we obtain

$$\tilde{\nabla} \tilde{A} = \nabla \cdot \mathbf{A} \sigma_0 + n_r \sigma_r + n_\theta \sigma_\theta + n_\phi \sigma_\phi, \quad (40)$$

where the following symbols have been used:

$$\begin{aligned}
 \nabla \cdot \mathbf{A} &= \partial_r A_r + \frac{2 A_r}{r} + \frac{\partial_\theta A_\theta}{r} + \frac{\cos \theta A_\theta}{r \sin \theta} + \frac{\partial_\phi A_\phi}{r \sin \theta} \\
 n_r &= \frac{i}{r \sin \theta} (-\partial_\phi A_\theta + \sin \theta \partial_\theta A_\phi + \cos \theta A_\phi) \\
 n_\theta &= i \left(\frac{\partial_\phi A_r}{r \sin \theta} - \frac{A_\phi}{r} - \partial_r A_\phi \right) \\
 n_\phi &= i \left(\frac{\partial_\theta A_r}{r} - \frac{A_\theta}{r} + \partial_r A_\theta \right).
 \end{aligned} \tag{41}$$

In (41) the term $\nabla \cdot \mathbf{A}$ is the conventional divergence.

Since $\nabla \wedge \mathbf{A} = i \nabla \times \mathbf{A}$ the other terms n_r, n_θ, n_ϕ are simply the components along r, θ and ϕ of the curl operator multiplied by i .

It is noted that while in conventional algebra there is no single operator providing both the divergence and the curl, by using Pauli matrices a single operator (37) exists for the nabla representation.

Second order operators

Further application of the nabla operator allows to obtain

$$\tilde{\nabla} \tilde{\nabla} \tilde{A} = \tilde{\nabla}^2 \tilde{A} \quad (42)$$

which, in terms of Pauli matrices, can be computed by matrix multiplication.

In vector terms the second order operator gives the vector Laplacian $\nabla^2 \mathbf{A}$:

$$\nabla (\nabla \mathbf{A}) = \nabla^2 \mathbf{A} = \nabla \cdot \nabla \wedge \mathbf{A} + \nabla \wedge \nabla \cdot \mathbf{A} \quad (43)$$

which shows that the final result is a vector.

It is convenient to introduce the scalar Laplacian operator for spherical coordinates ∇_s^2 defined as:

$$\nabla_s^2 w = \frac{1}{r^2} \frac{\partial}{\partial r} \left(r^2 \frac{\partial w}{\partial r} \right) + \frac{1}{r^2 \sin \theta} \frac{\partial}{\partial \theta} \left(\sin \theta \frac{\partial w}{\partial \theta} \right) + \frac{1}{r^2 \sin^2 \theta} \frac{\partial^2 w}{\partial \phi^2} \quad (44)$$

Vector Laplacian

and the coefficients:

$$\begin{aligned}N_r &= \nabla_s^2 A_r - \frac{2}{r^2 \sin \theta} (A_r \sin \theta + A_\theta \cos \theta + \sin \theta \partial_\theta A_\theta + \partial_\phi A_\phi) \\N_\theta &= \nabla_s^2 A_\theta + \frac{1}{r^2 \sin^2 \theta} (2 \sin^2 \theta \partial_\theta A_r - A_\theta - 2 \cos \theta \partial_\phi A_\phi) \\N_\phi &= \nabla_s^2 A_\phi + \frac{1}{r^2 \sin^2 \theta} (2 \partial_\phi A_r + 2 \cos \theta \partial_\phi A_\theta - A_\phi) .\end{aligned}\quad (45)$$

The second order nabla operator can finally be expressed as:

$$\tilde{\nabla}^2 \tilde{A} = N_r \sigma_r + N_\theta \sigma_\theta + N_\phi \sigma_\phi ,\quad (46)$$

Differences between the scalar Laplacian (33) and the vector Laplacian (43)

It is important to note the differences between the scalar Laplacian (33) and the vector Laplacian (43). In conventional vector algebra the vector Laplacian is expressed as

$$\nabla^2 \mathbf{A} = \nabla \nabla \cdot \mathbf{A} - \nabla \times \nabla \times \mathbf{A}. \quad (47)$$

In the following code the relevant quantities are computed first in terms of conventional operators. Then they are computed in terms of Pauli matrices. When feasible a test has been introduced showing that the results are coincident.

```

1 1. The first step is to create a new project in the IDE.
2 2. The second step is to create a new class named 'Main'.
3 3. The third step is to create a new method named 'main'.
4 4. The fourth step is to create a new variable named 'x'.
5 5. The fifth step is to create a new variable named 'y'.
6 6. The sixth step is to create a new variable named 'z'.
7 7. The seventh step is to create a new variable named 'w'.
8 8. The eighth step is to create a new variable named 'v'.
9 9. The ninth step is to create a new variable named 'u'.
10 10. The tenth step is to create a new variable named 't'.
11 11. The eleventh step is to create a new variable named 's'.
12 12. The twelfth step is to create a new variable named 'r'.
13 13. The thirteenth step is to create a new variable named 'q'.
14 14. The fourteenth step is to create a new variable named 'p'.
15 15. The fifteenth step is to create a new variable named 'o'.
16 16. The sixteenth step is to create a new variable named 'n'.
17 17. The seventeenth step is to create a new variable named 'm'.
18 18. The eighteenth step is to create a new variable named 'l'.
19 19. The nineteenth step is to create a new variable named 'k'.
20 20. The twentieth step is to create a new variable named 'j'.
21 21. The twenty-first step is to create a new variable named 'i'.
22 22. The twenty-second step is to create a new variable named 'h'.
23 23. The twenty-third step is to create a new variable named 'g'.
24 24. The twenty-fourth step is to create a new variable named 'f'.
25 25. The twenty-fifth step is to create a new variable named 'e'.
26 26. The twenty-sixth step is to create a new variable named 'd'.
27 27. The twenty-seventh step is to create a new variable named 'c'.
28 28. The twenty-eighth step is to create a new variable named 'b'.
29 29. The twenty-ninth step is to create a new variable named 'a'.
30 30. The thirtieth step is to create a new variable named 'z'.
31 31. The thirty-first step is to create a new variable named 'y'.
32 32. The thirty-second step is to create a new variable named 'x'.
33 33. The thirty-third step is to create a new variable named 'w'.
34 34. The thirty-fourth step is to create a new variable named 'v'.
35 35. The thirty-fifth step is to create a new variable named 'u'.
36 36. The thirty-sixth step is to create a new variable named 't'.
37 37. The thirty-seventh step is to create a new variable named 's'.
38 38. The thirty-eighth step is to create a new variable named 'r'.
39 39. The thirty-ninth step is to create a new variable named 'q'.
40 40. The fortieth step is to create a new variable named 'p'.
41 41. The forty-first step is to create a new variable named 'o'.
42 42. The forty-second step is to create a new variable named 'n'.
43 43. The forty-third step is to create a new variable named 'm'.
44 44. The forty-fourth step is to create a new variable named 'l'.
45 45. The forty-fifth step is to create a new variable named 'k'.
46 46. The forty-sixth step is to create a new variable named 'j'.
47 47. The forty-seventh step is to create a new variable named 'i'.
48 48. The forty-eighth step is to create a new variable named 'h'.
49 49. The forty-ninth step is to create a new variable named 'g'.
50 50. The fiftieth step is to create a new variable named 'f'.
51 51. The fifty-first step is to create a new variable named 'e'.
52 52. The fifty-second step is to create a new variable named 'd'.
53 53. The fifty-third step is to create a new variable named 'c'.
54 54. The fifty-fourth step is to create a new variable named 'b'.
55 55. The fifty-fifth step is to create a new variable named 'a'.
56 56. The fifty-sixth step is to create a new variable named 'z'.
57 57. The fifty-seventh step is to create a new variable named 'y'.
58 58. The fifty-eighth step is to create a new variable named 'x'.
59 59. The fifty-ninth step is to create a new variable named 'w'.
60 60. The sixtieth step is to create a new variable named 'v'.
61 61. The sixty-first step is to create a new variable named 'u'.
62 62. The sixty-second step is to create a new variable named 't'.
63 63. The sixty-third step is to create a new variable named 's'.
64 64. The sixty-fourth step is to create a new variable named 'r'.
65 65. The sixty-fifth step is to create a new variable named 'q'.
66 66. The sixty-sixth step is to create a new variable named 'p'.
67 67. The sixty-seventh step is to create a new variable named 'o'.
68 68. The sixty-eighth step is to create a new variable named 'n'.
69 69. The sixty-ninth step is to create a new variable named 'm'.
70 70. The seventieth step is to create a new variable named 'l'.
71 71. The seventy-first step is to create a new variable named 'k'.
72 72. The seventy-second step is to create a new variable named 'j'.
73 73. The seventy-third step is to create a new variable named 'i'.
74 74. The seventy-fourth step is to create a new variable named 'h'.
75 75. The seventy-fifth step is to create a new variable named 'g'.
76 76. The seventy-sixth step is to create a new variable named 'f'.
77 77. The seventy-seventh step is to create a new variable named 'e'.
78 78. The seventy-eighth step is to create a new variable named 'd'.
79 79. The seventy-ninth step is to create a new variable named 'c'.
80 80. The eightieth step is to create a new variable named 'b'.
81 81. The eighty-first step is to create a new variable named 'a'.
82 82. The eighty-second step is to create a new variable named 'z'.
83 83. The eighty-third step is to create a new variable named 'y'.
84 84. The eighty-fourth step is to create a new variable named 'x'.
85 85. The eighty-fifth step is to create a new variable named 'w'.
86 86. The eighty-sixth step is to create a new variable named 'v'.
87 87. The eighty-seventh step is to create a new variable named 'u'.
88 88. The eighty-eighth step is to create a new variable named 't'.
89 89. The eighty-ninth step is to create a new variable named 's'.
90 90. The ninetieth step is to create a new variable named 'r'.
91 91. The ninety-first step is to create a new variable named 'q'.
92 92. The ninety-second step is to create a new variable named 'p'.
93 93. The ninety-third step is to create a new variable named 'o'.
94 94. The ninety-fourth step is to create a new variable named 'n'.
95 95. The ninety-fifth step is to create a new variable named 'm'.
96 96. The ninety-sixth step is to create a new variable named 'l'.
97 97. The ninety-seventh step is to create a new variable named 'k'.
98 98. The ninety-eighth step is to create a new variable named 'j'.
99 99. The ninety-ninth step is to create a new variable named 'i'.
100 100. The hundredth step is to create a new variable named 'h'.

```

Relating vector algebra to conventional vector analysis

Triple products

We have previously shown that the following identities between vectors hold:

$$\mathbf{a} \wedge \mathbf{b} = i \mathbf{a} \times \mathbf{b} \quad (48)$$

$$\mathbf{a} \cdot \mathbf{b} \wedge \mathbf{c} = -\mathbf{a} \times \mathbf{b} \times \mathbf{c} \quad (49)$$

$$\mathbf{a} \wedge \mathbf{b} \wedge \mathbf{c} = i \mathbf{a} \cdot (\mathbf{b} \times \mathbf{c}). \quad (50)$$

These equations allows to translate into conventional vector analysis (VA) the identities obtained in geometric algebra (GA) and viceversa. As an example, consider the triple geometric product $\mathbf{a} \mathbf{b} \mathbf{c}$ expressed as:

$$\begin{aligned} \mathbf{a} \mathbf{b} \mathbf{c} &= \mathbf{a} (\mathbf{b} \cdot \mathbf{c} + \mathbf{b} \wedge \mathbf{c}) \\ &= \mathbf{a} \cdot (\mathbf{b} \cdot \mathbf{c} + \mathbf{b} \wedge \mathbf{c}) + \mathbf{a} \wedge (\mathbf{b} \cdot \mathbf{c} + \mathbf{b} \wedge \mathbf{c}) \\ &= \mathbf{a} \cdot \mathbf{b} \cdot \mathbf{c} + \mathbf{a} \cdot (\mathbf{b} \wedge \mathbf{c}) + \mathbf{a} \wedge (\mathbf{b} \cdot \mathbf{c}) + \mathbf{a} \wedge \mathbf{b} \wedge \mathbf{c} \\ &= -\mathbf{a} \times \mathbf{b} \times \mathbf{c} + \mathbf{a}(\mathbf{b} \cdot \mathbf{c}) + i \mathbf{a} \cdot (\mathbf{b} \times \mathbf{c}) \end{aligned} \quad (51)$$

where the term $\mathbf{a} \cdot \mathbf{b} \cdot \mathbf{c}$ evidently is equal to zero.

Triple products: grade interpretation

Computationally, by using the Pauli matrices ($\tilde{a}, \tilde{b}, \tilde{c}$) the triple product is trivial and is simply a matrix product.

However, from direct interpretation of the result in (51) one may also recognize:

- the term $\mathbf{a} \cdot (\mathbf{b} \wedge \mathbf{c}) = -\mathbf{a} \times \mathbf{b} \times \mathbf{c}$ is the dot product of a vector and a bivector giving as a result a vector;
- the term $\mathbf{a} \wedge (\mathbf{b} \cdot \mathbf{c}) = \mathbf{a}(\mathbf{b} \cdot \mathbf{c})$ corresponds also to a vector;
- the last term $\mathbf{a} \wedge \mathbf{b} \wedge \mathbf{c} = i \mathbf{a} \cdot (\mathbf{b} \times \mathbf{c})$ is a trivector (or pseudoscalar).

The above identities are also useful in order to find some relationships concerning the nabla operator.

The nabla operator in geometric algebra

It is instructive to start from (48)–(50) and to formally substitute \mathbf{a} with the nabla operator ∇ , thus obtaining:

$$\nabla \wedge \mathbf{b} = i \nabla \times \mathbf{b} \quad (52)$$

$$\nabla \cdot (\mathbf{b} \wedge \mathbf{c}) = -\nabla \times (\mathbf{b} \times \mathbf{c}) \quad (53)$$

$$\nabla \wedge (\mathbf{b} \wedge \mathbf{c}) = i \nabla \cdot (\mathbf{b} \times \mathbf{c}). \quad (54)$$

Equation (52) simply relates the external product of ∇ with a vector \mathbf{b} with the curl operation.

The other two equations (53), (54) are interesting, since they refer to the divergence and the external product of a *bivector*.

Equation (53) states that the divergence of a bivector $\mathbf{b} \wedge \mathbf{c}$ is equal to minus the curl of $\mathbf{b} \times \mathbf{c}$, i.e. to a vector.

Equation (54) tells us that the external product of ∇ with a bivector $\mathbf{b} \wedge \mathbf{c}$ is equal to the divergence of $\mathbf{b} \times \mathbf{c}$ multiplied by i .

Proof of $\nabla \cdot (\mathbf{b} \wedge \mathbf{c}) = -\nabla \times (\mathbf{b} \times \mathbf{c})$

An alternative way to prove (53) is the following. Let us assume

$$\nabla = \mathbf{e}_1 \partial_1 + \mathbf{e}_2 \partial_2 + \mathbf{e}_3 \partial_3 \quad (55)$$

and consider a bivector $\hat{\mathbf{B}}$ given by

$$\hat{\mathbf{B}} = i\mathbf{B} = i(\mathbf{e}_1 B_1 + \mathbf{e}_2 B_2 + \mathbf{e}_3 B_3) = \mathbf{e}_{23} B_1 + \mathbf{e}_{31} B_2 + \mathbf{e}_{12} B_3. \quad (56)$$

Dot multiplication provides:

$$\begin{aligned} \nabla \cdot \hat{\mathbf{B}} &= (\mathbf{e}_1 \partial_1 + \mathbf{e}_2 \partial_2 + \mathbf{e}_3 \partial_3) \cdot (\mathbf{e}_{23} B_1 + \mathbf{e}_{31} B_2 + \mathbf{e}_{12} B_3) \\ &= -\mathbf{e}_3 \partial_1 B_2 + \mathbf{e}_2 \partial_1 B_3 + \mathbf{e}_3 \partial_2 B_1 - \mathbf{e}_1 \partial_2 B_3 - \mathbf{e}_2 \partial_3 B_1 + \mathbf{e}_1 \partial_3 B_2 \\ &= -[\mathbf{e}_1 (\partial_2 B_3 - \partial_3 B_2) + \mathbf{e}_2 (\partial_3 B_1 - \partial_1 B_3) + \mathbf{e}_3 (\partial_1 B_2 - \partial_2 B_1)] \\ &= -\nabla \times \mathbf{B} \end{aligned} \quad (57)$$

which proves the relation (52).

We also have $\nabla \cdot (\nabla \wedge \mathbf{c}) = -\nabla \times (\nabla \times \mathbf{c})$

Proof of $\nabla \wedge (\mathbf{b} \wedge \mathbf{c}) = i \nabla \cdot (\mathbf{b} \times \mathbf{c})$

Equation (54) can be verified by considering:

$$\nabla = \mathbf{e}_1 \partial_1 + \mathbf{e}_2 \partial_2 + \mathbf{e}_3 \partial_3 \quad (58)$$

$$\hat{\mathbf{B}} = i\mathbf{B} = i(\mathbf{e}_1 B_1 + \mathbf{e}_2 B_2 + \mathbf{e}_3 B_3) = \mathbf{e}_{23} B_1 + \mathbf{e}_{31} B_2 + \mathbf{e}_{12} B_3. \quad (59)$$

$$\begin{aligned} \nabla \wedge \hat{\mathbf{B}} &= (\mathbf{e}_1 \partial_1 + \mathbf{e}_2 \partial_2 + \mathbf{e}_3 \partial_3) \wedge (\mathbf{e}_{23} B_1 + \mathbf{e}_{31} B_2 + \mathbf{e}_{12} B_3) \\ &= \mathbf{e}_{123} \partial_1 B_1 + \mathbf{e}_{231} \partial_2 B_2 + \mathbf{e}_{312} \partial_3 B_3 \\ &= i \nabla \cdot \mathbf{B} \end{aligned} \quad (60)$$

since

$$i = \mathbf{e}_{123} = \mathbf{e}_{231} = \mathbf{e}_{312}. \quad (61)$$

We also have $\nabla \wedge (\nabla \wedge \mathbf{c}) = i \nabla \cdot (\nabla \times \mathbf{c}) = 0$

Geometric product of $\nabla(\mathbf{b} \mathbf{c})$

By formally substituting \mathbf{a} with ∇ and considering the latter acting on $\mathbf{b} \mathbf{c}$ one derives from (51):

$$\begin{aligned}\nabla(\mathbf{b} \mathbf{c}) &= \nabla \cdot (\mathbf{b} \wedge \mathbf{c}) + \nabla \wedge (\mathbf{b} \cdot \mathbf{c}) + \nabla \wedge (\mathbf{b} \wedge \mathbf{c}) \\ &= -\nabla \times (\mathbf{b} \times \mathbf{c}) + \nabla(\mathbf{b} \cdot \mathbf{c}) + i \nabla \cdot (\mathbf{b} \times \mathbf{c}).\end{aligned}\quad (62)$$

Naturally $\mathbf{b} \mathbf{c}$ is a product of two vectors and, as such, is a scalar plus a bivector.

The operation $\nabla \wedge (\mathbf{b} \wedge \mathbf{c})$ provides a trivector, while the other two operations, $\nabla \cdot (\mathbf{b} \wedge \mathbf{c})$ and $\nabla \wedge (\mathbf{b} \cdot \mathbf{c})$, return a vector.

Note that we have made use of the fact (cite Jancewicz) that, for a scalar function ϕ , we may interpret $\nabla\phi = \nabla \wedge \phi$ i.e. the gradient is obtained as an external product of a vector with a scalar.

Second order derivatives

Always referring to (51) it is also possible to make the additional substitution of \mathbf{b} with ∇ . It is noted that in GA we have (the external part of a vector to itself is null)

$$\nabla \wedge \nabla \wedge \mathbf{c} = 0, \quad (63)$$

and therefore we get the identity:

$$\nabla \nabla \mathbf{c} = \nabla \cdot (\nabla \wedge \mathbf{c}) + \nabla \wedge (\nabla \cdot \mathbf{c}) \quad (64)$$

$$= \nabla^2 \mathbf{c} = \Delta \mathbf{c} \quad (65)$$

where Δ is the *Laplacian*.

It is straightforward to prove that the above identity also holds for multivectors.

Noticeable cases

From (63) applied to a scalar function ϕ and taking into account (48) one gets

$$\nabla \wedge \nabla \wedge \phi = 0 = i [\nabla \times (\nabla \phi)] , \quad (66)$$

i.e. the classical $\text{curl}(\text{grad } \phi = 0)$.

Moreover, from (63) and (50) we get:

$$\nabla \wedge \nabla \wedge \mathbf{c} = 0 = i [\nabla \cdot (\nabla \times \mathbf{c})] , \quad (67)$$

i.e. $\text{div}(\text{curl } \mathbf{c} = 0)$.

Consequences

The above identities suggest that if we have e.g. a vector field \mathbf{E} , which has to satisfy $\nabla \wedge \mathbf{E} = 0$, such a field can be expressed as $\mathbf{E} = \nabla \phi$ and (66) will be automatically satisfied.

Similarly, if we have a bivector field $\hat{\mathbf{H}} = i \mathbf{H}$, which has to satisfy

$$\nabla \wedge \hat{\mathbf{H}} = \nabla \cdot (i \mathbf{H}) = 0,$$

the bivector field $\hat{\mathbf{H}}$ can be expressed as $\hat{\mathbf{H}} = \nabla \wedge \mathbf{A}$, with (67) satisfied.

Note that in (67) we can also give another interpretation by looking at the r.h.s.. In fact, we can say that if a vector \mathbf{H} has to have zero divergence than it can be expressed as the curl of a vector \mathbf{A} .

Summary of Pauli matrices

Table 9: Basis vector in rectangular, cylindrical and spherical coordinate systems. The \mathbf{e}_i are a Clifford basis and correspond to the appropriate Pauli matrices.

	\mathbf{e}_1	\mathbf{e}_2	\mathbf{e}_3
rectangular	$\sigma_1 = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$	$\sigma_2 = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}$	$\sigma_3 = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$
cylindrical	$\sigma_\rho = \begin{pmatrix} 0 & e^{-i\phi} \\ e^{i\phi} & 0 \end{pmatrix}$	$\sigma_\phi = \begin{pmatrix} 0 & -i e^{-i\phi} \\ i e^{i\phi} & 0 \end{pmatrix}$	$\sigma_3 = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$
spherical	$\sigma_r = \begin{pmatrix} \cos\theta & e^{-i\phi} \sin\theta \\ e^{i\phi} \sin\theta & -\cos\theta \end{pmatrix}$	$\sigma_\theta = \begin{pmatrix} -\sin\theta & e^{-i\phi} \cos\theta \\ e^{i\phi} \cos\theta & \sin\theta \end{pmatrix}$	$\sigma_\phi = \begin{pmatrix} 0 & -i e^{-i\phi} \\ i e^{i\phi} & 0 \end{pmatrix}$

Summary of vector expressions

$$\begin{aligned}\tilde{A} &= \sigma_1 A_x + \sigma_2 A_y + \sigma_3 A_z \\ &= \begin{pmatrix} A_z & A_x - i A_y \\ i A_y + A_x & -A_z \end{pmatrix}\end{aligned}$$

$$\begin{aligned}\tilde{A} &= \sigma_\rho A_\rho + \sigma_\phi A_\phi + \sigma_3 A_z \\ &= \begin{pmatrix} A_z & e^{-i\phi} (A_\rho - i A_\phi) \\ e^{i\phi} (A_\rho + i A_\phi) & -A_z \end{pmatrix}\end{aligned}$$

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

Summary of Nabla expressions

Rectangular:

$$\begin{aligned}\tilde{\nabla} &= \sigma_1 \partial_x + \sigma_2 \partial_y + \sigma_3 \partial_z \\ &= \begin{pmatrix} \partial_z & \partial_x - i \partial_y \\ i \partial_y + \partial_x & -\partial_z \end{pmatrix}.\end{aligned}$$

Cylindrical:

$$\begin{aligned}\tilde{\nabla} &= \sigma_\rho \partial_\rho + \frac{1}{\rho} \sigma_\phi \partial_\phi + \sigma_z \partial_z \\ &= \begin{pmatrix} \partial_z & \frac{e^{-i\phi}}{\rho} (\rho \partial_\rho - i \partial_\phi) \\ \frac{e^{i\phi}}{\rho} (\rho \partial_\rho + i \partial_\phi) & -\partial_z \end{pmatrix}\end{aligned}$$

Spherical:

$$\begin{aligned}\tilde{\nabla} &= \sigma_r \partial_r + \frac{1}{r} \sigma_\theta \partial_\theta + \frac{1}{r \sin \theta} \sigma_\phi \partial_\phi \\ &= \begin{pmatrix} \cos \theta \partial_r - \sin \theta \partial_\theta & e^{-i\phi} (\sin \theta \partial_r + \cos \theta \partial_\theta - i \partial_\phi) \\ e^{i\phi} (\sin \theta \partial_r + \cos \theta \partial_\theta + i \partial_\phi) & -(\cos \theta \partial_r - \sin \theta \partial_\theta) \end{pmatrix}\end{aligned}$$