5.2 Curl, Line Integrals, and Circulation

5.2.1 Line Integrals

Consider a curve C in a Euclidean vector space $\mathbb{R}^3$. Let $\boldsymbol{r}(t) = [x(t), y(t), z(t)]$ be a parametric representation of C. Furthermore, let $\boldsymbol{F} : \mathbb{R}^3 \to \mathbb{R}^3$ be a vector-valued function of $\boldsymbol{r}$ and let $\boldsymbol{r}'(t)$ be the tangent vector. The **line integral** is

$$\int_C \boldsymbol{F}(\boldsymbol{r}(t)) \cdot \boldsymbol{r}'(t)\,\mathrm{d}t \tag{5.8}$$

which integrates $\boldsymbol{F}$ along the curve. For more on computing line integrals, see (Schey 2005; pp. 63-74) and (Kreyszig 2011; § 10.1 and 10.2).

If $\boldsymbol{F}$ is a **force** being applied to an object moving along the curve C, the line integral is the **work** done by the force. More generally, the line integral integrates $\boldsymbol{F}$ along the tangent of C.

5.2.2 Circulation

Consider the line integral over a closed curve C, denoted by

$$\oint_C \boldsymbol{F}(\boldsymbol{r}(t)) \cdot \boldsymbol{r}'(t)\,\mathrm{d}t. \tag{5.9}$$

We call this quantity the **circulation** of $\boldsymbol{F}$ around C.

For certain vector-valued functions $\boldsymbol{F}$, the circulation is zero for every curve. In these cases (static electric fields, for instance), this is sometimes called the **the law of circulation**.

5.2.3 Curl

Consider the division of the circulation around a curve in $\mathbb{R}^3$ by the surface area it encloses ΔS,

$$\frac{1}{\Delta S}\oint_C \boldsymbol{F}(\boldsymbol{r}(t)) \cdot \boldsymbol{r}'(t)\,\mathrm{d}t. \tag{5.10}$$

In a manner analogous to the operation that gaves us the divergence, let's consider shrinking this curve to a point and the surface area to zero,

$$\lim_{\Delta S \to 0} \frac{1}{\Delta S}\oint_C \boldsymbol{F}(\boldsymbol{r}(t)) \cdot \boldsymbol{r}'(t)\,\mathrm{d}t. \tag{5.11}$$

We call this quantity the "scalar" **curl** of $\boldsymbol{F}$ at each point in $\mathbb{R}^3$ *in the direction normal to* ΔS as it shrinks to zero. Taking three (or n for $\mathbb{R}^n$) "scalar" curls in indepedent normal directions (enough to span the vector space), we obtain the **curl** proper, which is a vector-valued function $\text{curl} : \mathbb{R}^3 \to \mathbb{R}^3$.

The curl is coordinate-independent. In cartesian coordinates, it can be shown to be equivalent to the following.

Equation 5.12 curl: differential form, cartesian coordinates

$$\text{curl}\, \boldsymbol{F} = \begin{bmatrix} \partial_y F_z - \partial_z F_y & \partial_z F_x - \partial_x F_z & \partial_x F_y - \partial_y F_x \end{bmatrix}^\top$$

But what does the curl of $\boldsymbol{F}$ represent? It quantifies the local rotation of $\boldsymbol{F}$ about each point. If $\boldsymbol{F}$ represents a fluid's velocity, $\text{curl}\, \boldsymbol{F}$ is the local rotation of the fluid about each point and it is called the **vorticity**.

5.2.4 Zero Curl, Circulation, and Path Independence

5.2.4.1 Circulation It can be shown that if the circulation of $\boldsymbol{F}$ on all curves is zero, then in each direction $\boldsymbol{n}$ and at every point $\text{curl}\, \boldsymbol{F} = 0$ (i.e. $\boldsymbol{n} \cdot \text{curl}\, \boldsymbol{F} = 0$). Conversely, for $\text{curl}\, \boldsymbol{F} = 0$ *in a simply connected region*[2], $\boldsymbol{F}$ has *zero circulation*.

Succinctly, informally, and without the requisite qualifiers above,

$$\text{zero circulation} \Rightarrow \text{zero curl} \tag{5.13}$$

$$\text{zero curl } + \text{simply connected region} \Rightarrow \text{zero circulation.} \tag{5.14}$$

5.2.4.2 Path Independence It can be shown that if the path integral of $\boldsymbol{F}$ on all curves between any two points is **path-independent**, then in each direction $\boldsymbol{n}$ and at every point $\text{curl}\, \boldsymbol{F} = 0$ (i.e. $\boldsymbol{n} \cdot \text{curl}\, \boldsymbol{F} = 0$). Conversely, for $\text{curl}\, \boldsymbol{F} = 0$ in a simply connected region, all line integrals are independent of path.

Succinctly, informally, and without the requisite qualifiers above,

$$\text{path independence} \Rightarrow \text{zero curl} \tag{5.15}$$

$$\text{zero curl } + \text{simply connected region} \Rightarrow \text{path independence.} \tag{5.16}$$

5.2.4.3 And How They Relate It is also true that

$$\text{path independence} \Leftrightarrow \text{zero circulation.} \tag{5.17}$$

So, putting it all together, we get figure 5.5.

2. A region is simply connected if every curve in it can shrink to a point without leaving the region. An example of a region that is not simply connected is the surface of a toroid.

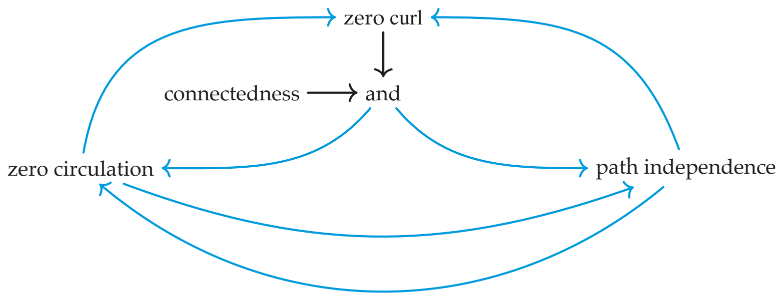

Figure 5.5. An implication graph relating zero curl, zero circulation, path independence, and connectedness. Blue edges represent implication (a implies b) and black edges represent logical conjunctions.

5.2.5 Exploring Curl

Curl is perhaps best explored by considering it for a vector field in $\mathbb{R}^2$. Such a field in cartesian coordinates $\boldsymbol{F} = F_x\hat{\boldsymbol{\imath}} + F_y\hat{\boldsymbol{\jmath}}$ has curl

$$\begin{aligned}\operatorname{curl}\boldsymbol{F} &= \begin{bmatrix}\partial_y 0 - \partial_z F_y & \partial_z F_x - \partial_x 0 & \partial_x F_y - \partial_y F_x\end{bmatrix}^\top \\ &= \begin{bmatrix}0-0 & 0-0 & \partial_x F_y - \partial_y F_x\end{bmatrix}^\top \\ &= \begin{bmatrix}0 & 0 & \partial_x F_y - \partial_y F_x\end{bmatrix}^\top. \end{aligned} \tag{5.18}$$

That is, $\operatorname{curl}\boldsymbol{F} = (\partial_x F_y - \partial_y F_x)\hat{\boldsymbol{k}}$ and the only nonzero component is normal to the xy-plane. If we overlay a quiver plot of $\boldsymbol{F}$ over a "color density" plot representing the $\hat{\boldsymbol{k}}$-component of $\operatorname{curl}\boldsymbol{F}$, we can increase our intuition about the curl. First, load some Python packages.

```
import numpy as np
import sympy as sp
import matplotlib.pyplot as plt
from matplotlib.ticker import LogLocator
from matplotlib.colors import *
```

Now we define some symbolic variables and functions.

```
x = sp.Symbol('x', real=True)
y = sp.Symbol('y', real=True)
F_x = sp.Function('F_x')(x, y)
F_y = sp.Function('F_y')(x, y)
```

We use a variation of the `quiver_plotter_2D()` from above to make several of these plots.

```python
def quiver_plotter_2D(
  field={F_x: x*y, F_y: x*y},
  grid_width=3,
  grid_decimate_x=8,
  grid_decimate_y=8,
  norm=Normalize(),
  density_operation='div',
  print_density=True,
):
  # Define symbolics
  x, y = sp.symbols('x y', real=True)
  F_x = sp.Function('F_x')(x, y)
  F_y = sp.Function('F_y')(x, y)
  field_sub = field
  # Compute density
  if density_operation == 'div':
    den = F_x.diff(x) + F_y.diff(y)
  elif density_operation == 'curl':
    den = F_y.diff(x) - F_x.diff(y)  # in the k direction
  else:
    raise ValueError('div and curl are the only density operators')
  den_simp = den.subs(field_sub).doit().simplify()
  if den_simp.is_constant():
    print('Warning: density operator is constant (no density plot)')
  if print_density:
    print(f'The {density_operation} is: {den_simp}')
  # Lambdify for numerics
  F_x_sub = F_x.subs(field_sub)
  F_y_sub = F_y.subs(field_sub)
  F_x_fun = sp.lambdify((x, y),F_x.subs(field_sub), 'numpy')
  F_y_fun = sp.lambdify((x, y), F_y.subs(field_sub), 'numpy')
  if F_x_sub.is_constant:
    F_x_fun1 = F_x_fun # Dummy
    F_x_fun = lambda x, y: F_x_fun1(x, y)*np.ones(x.shape)
  if F_y_sub.is_constant:
    F_y_fun1 = F_y_fun # Dummy
    F_y_fun = lambda x, y: F_y_fun1(x, y)*np.ones(x.shape)
  if not den_simp.is_constant():
    den_fun = sp.lambdify((x, y), den_simp,'numpy')
  # Create grid
  w = grid_width
  Y, X = np.mgrid[-w:w:100j, -w:w:100j]
  # Evaluate numerically
  F_x_num = F_x_fun(X, Y)
  F_y_num = F_y_fun(X, Y)
  if not den_simp.is_constant():
    den_num = den_fun(X, Y)
```

```
# Plot
fig, ax = plt.subplots()
if not den_simp.is_constant():
  cmap = plt.get_cmap('coolwarm')
  im = plt.pcolormesh(X, Y, den_num, cmap=cmap, norm=norm)
  plt.colorbar()
dx = grid_decimate_y
dy = grid_decimate_x
plt.quiver(
  X[::dx,::dy],Y[::dx,::dy],
  F_x_num[::dx,::dy], F_y_num[::dx,::dy],
  units='xy', scale=10)
plt.title(fr'$F(x, y) = \left[ {sp.latex(F_x.subs(field_sub))},' + \
          fr'{sp.latex(F_y.subs(field_sub))} \right]$')
return fig, ax
```

Let's inspect several cases.

```
fig, ax = quiver_plotter_2D(
  field={F_x: 0, F_y: sp.cos(2*sp.pi*x)}, density_operation='curl',
  grid_decimate_x=2, grid_decimate_y=10, grid_width=1
)
plt.draw()
```

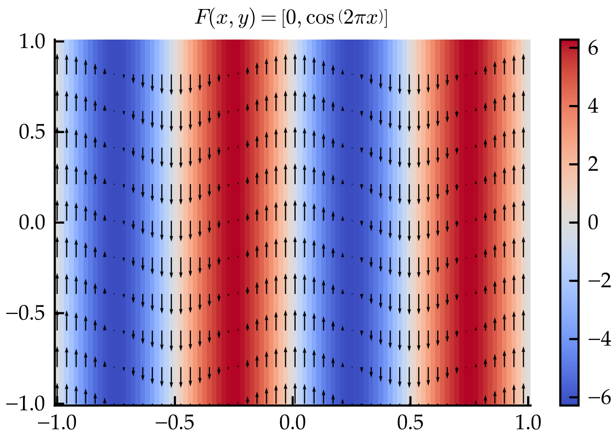

Figure 5.6. Quiver plot of $F(x, y) = [0, \cos(2\pi x)]$

```
fig, ax = quiver_plotter_2D(
  field={F_x: 0, F_y: x**2}, density_operation='curl',
  grid_decimate_x=2, grid_decimate_y=20,
)
plt.draw()
```

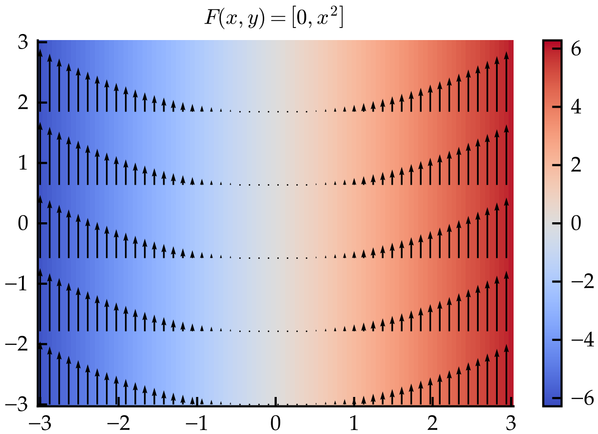

Figure 5.7. Quiver plot of $F(x, y) = [0, x^2]$

```
fig, ax = quiver_plotter_2D(
  field={F_x: y**2, F_y: x**2}, density_operation='curl',
  grid_decimate_x=2, grid_decimate_y=20,
)
plt.draw()
```

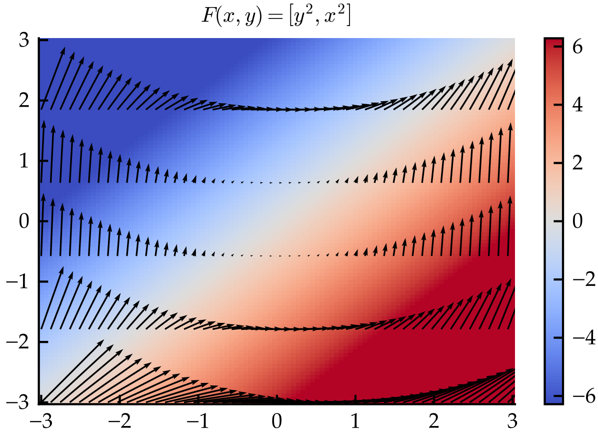

Figure 5.8. Quiver plot of $F(x, y) = [y^2, x^2]$

```
fig, ax = quiver_plotter_2D(
  field={F_x: -y, F_y: x}, density_operation='curl',
  grid_decimate_x=6, grid_decimate_y=6,
)
plt.show()
```

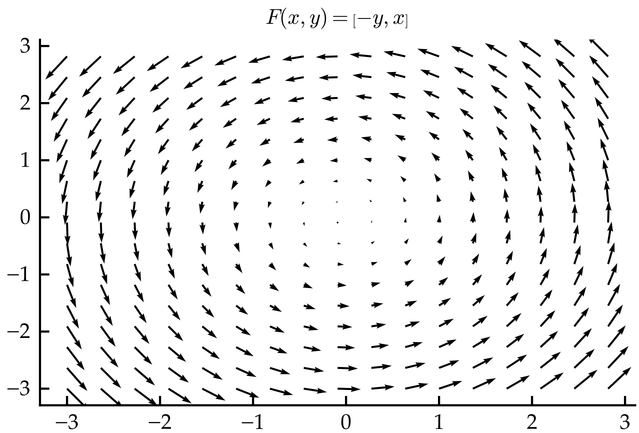

Figure 5.9. Quiver plot of $F(x, y) = [-y, x]$

5.3 Gradient

5.3.1 Gradient

The **gradient** grad of a scalar-valued function $f : \mathbb{R}^3 \to \mathbb{R}$ is a vector field $\boldsymbol{F} : \mathbb{R}^3 \to \mathbb{R}^3$; that is, grad f is a vector-valued function on $\mathbb{R}^3$. The gradient's local **direction** and **magnitude** are those of the local maximum rate of increase of f. This makes it useful in optimization (e.g., in the method of gradient descent).

This principle tells us that nature's laws quite frequently seem to be derivable by assuming a certain quantity—called *action*—is minimized. Considering, then, that the gradient supplies us with a tool for optimizing functions, it is unsurprising that the gradient enters into the equations of motion of many physical quantities.

The gradient is coordinate-independent, but its coordinate-free definitions don't add much to our intuition.

Equation 5.19 gradient: cartesian coordinates

$$\text{grad}\, f = \begin{bmatrix} \partial_x f & \partial_y f & \partial_z f \end{bmatrix}^\top$$

5.3.2 Vector Fields from Gradients Are Special

Although all gradients are vector fields, not all vector fields are gradients. That is, given a vector field $\boldsymbol{F}$, it may or may not be equal to the gradient of any scalar-valued function f. Let's say of a vector field that is a gradient that it has **gradience**.[3] Those vector fields that *are* gradients have special properties. Surprisingly, those properties are connected to path independence and curl. It can be shown that iff a field is a gradient, line integrals of the field are path independent. That is, for a vector field,

$$\text{gradience} \Leftrightarrow \text{path independence.} \tag{5.20}$$

Considering what we know from section 5.2 about path independence we can expand figure 5.5 to obtain figure 5.10.

3. This is nonstandard terminology, but we're bold.