

composition_ondes

August 4, 2019

1 Composition d'ondes harmoniques planes

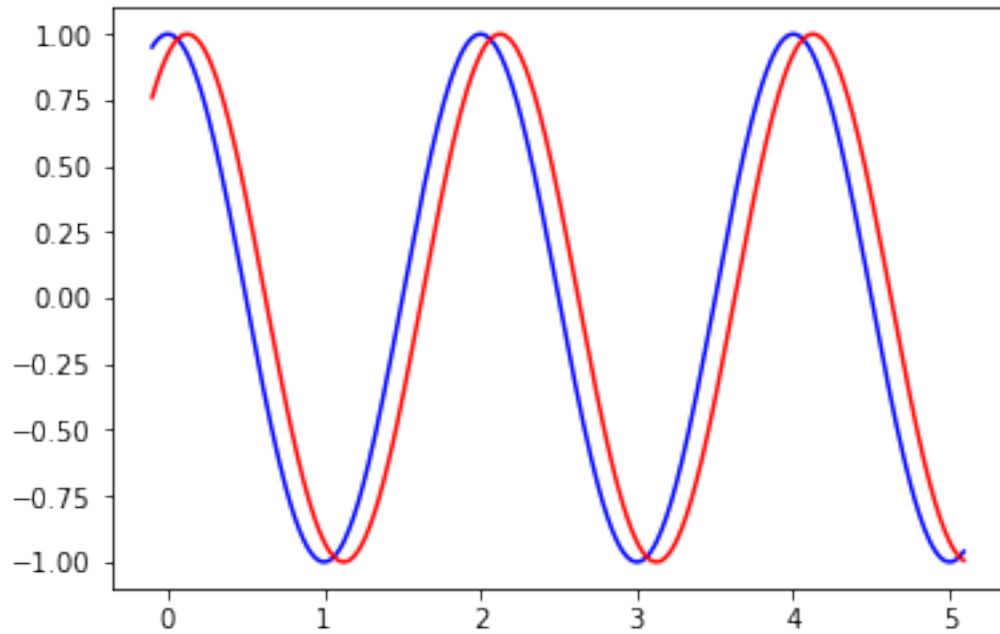
1.0.1 Définition de deux ondes semblables: mêmes amplitude et longueur

```
In [18]: import matplotlib.pyplot as plt
import numpy as np

phi0 = 1.0
f = 3.0      # Hz
lamb = 2.0   # m
c = lamb * f
omega = 2.0 * np.pi * f
k = omega / c
Phi = np.pi / 8

x = np.arange(-0.1, 5.1, 0.01)
t = 0.0
y1 = 1.0*np.cos(omega * t - k * x)
y2 = 1.0*np.cos(omega * t - k * x + Phi)

plt.plot(x, y1, 'b')
plt.plot(x, y2, 'r')
plt.show()
```

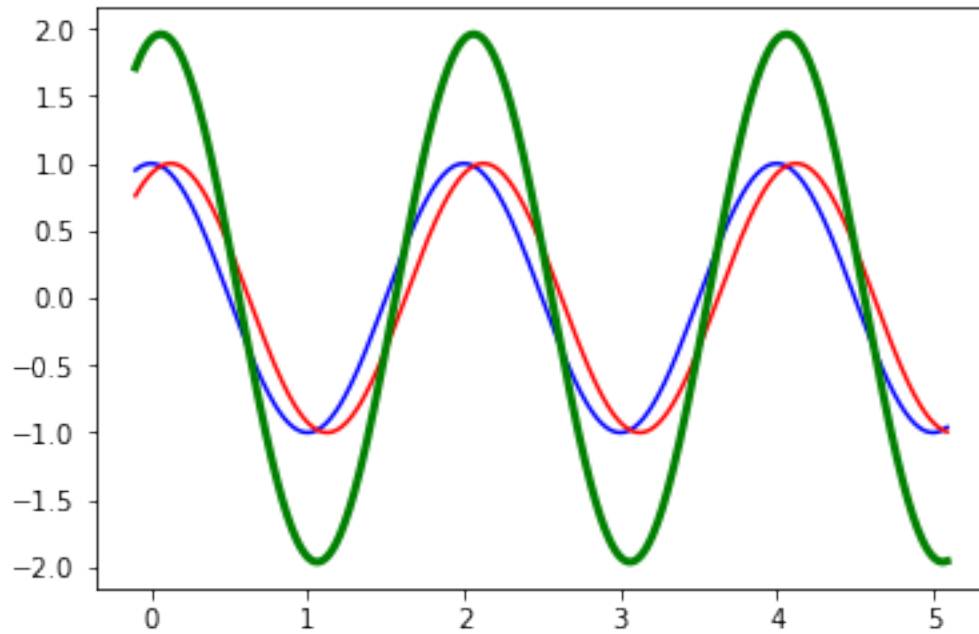


1.0.2 Composition des deux ondes, avec un déphasage très petit

Les deux ondes se renforcent, et le résultat est visiblement reconnaissable comme une onde sinusoïdale, de même fréquence mais d'amplitude plus grande. *Les deux ondes se sont renforcées.*

```
In [19]: y1 = 1.0*np.cos(omega * t - k * x)
          y2 = 1.0*np.cos(omega * t - k * x + Phi)
          y = y1 + y2

          plt.plot(x, y1, 'b-')
          plt.plot(x, y2, 'r-')
          plt.plot(x, y, 'g', linewidth=3)
          plt.show()
```



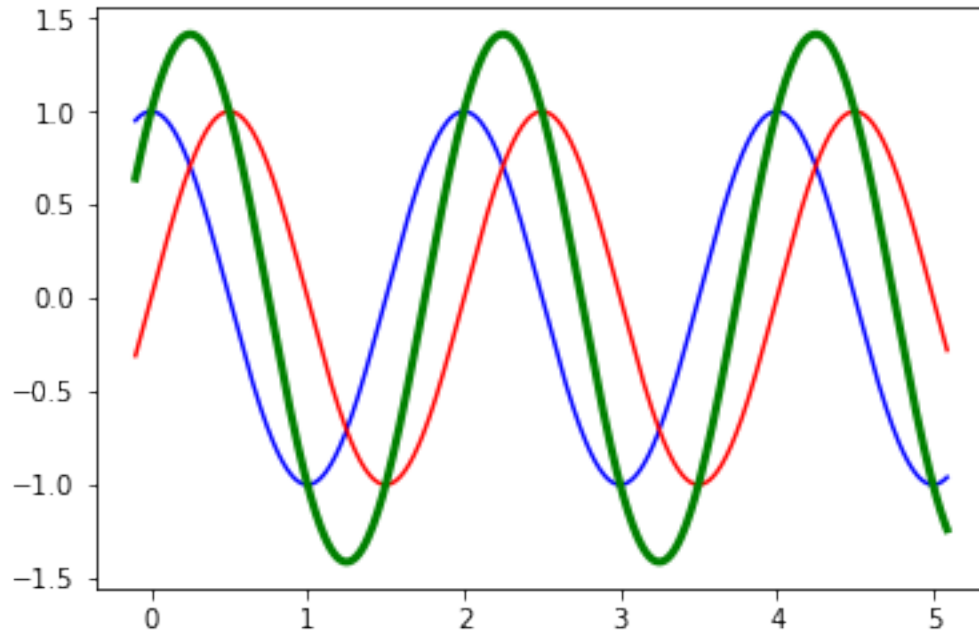
1.0.3 Composition des deux ondes, avec un déphasage plus grand

Les deux ondes continuent à se renforcer, mais chaque fois moins à mesure que le déphasage entre elles augmente.

```
In [20]: Phi = np.pi / 2

y1 = 1.0*np.cos(omega * t - k * x)
y2 = 1.0*np.cos(omega * t - k * x + Phi)
y = y1 + y2

plt.plot(x, y1, 'b-')
plt.plot(x, y2, 'r-')
plt.plot(x, y, 'g', linewidth=3)
plt.show()
```



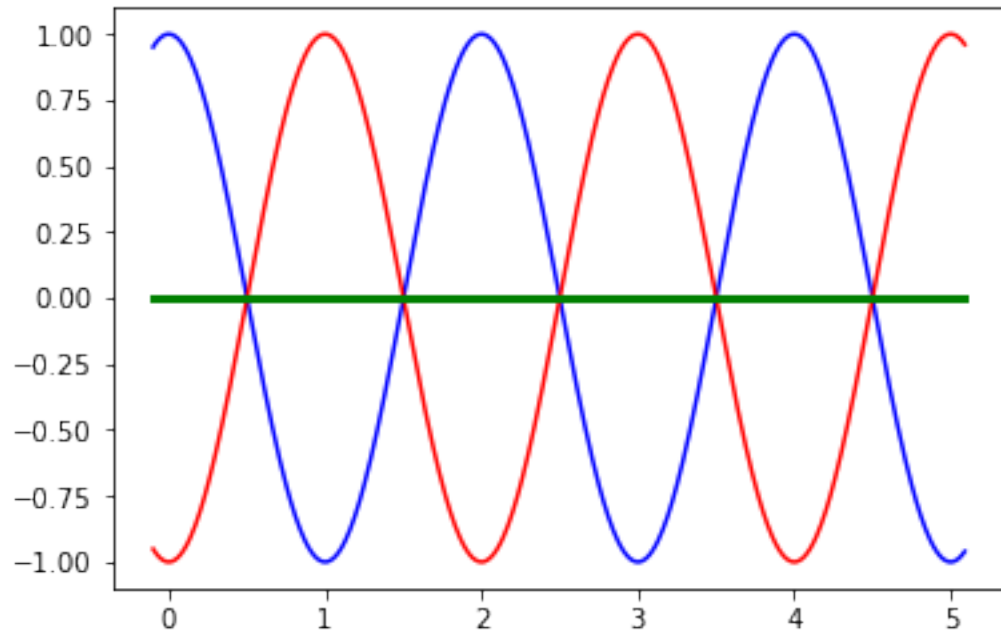
1.0.4 Composition de deux ondes en opposition de phase

Quand le déphasage entre les deux ondes es de $\pi/2$, elles sont dites en *opposition de phase*. Alors, leurs évolutions sont exactement inversées, et elles "s'anulent" l'une l'autre.

```
In [21]: Phi = np.pi
```

```
y1 = 1.0*np.cos(omega * t - k * x)
y2 = 1.0*np.cos(omega * t - k * x + Phi)
y = y1 + y2
```

```
plt.plot(x, y1, 'b-')
plt.plot(x, y2, 'r-')
plt.plot(x, y, 'g', linewidth=3)
plt.show()
```

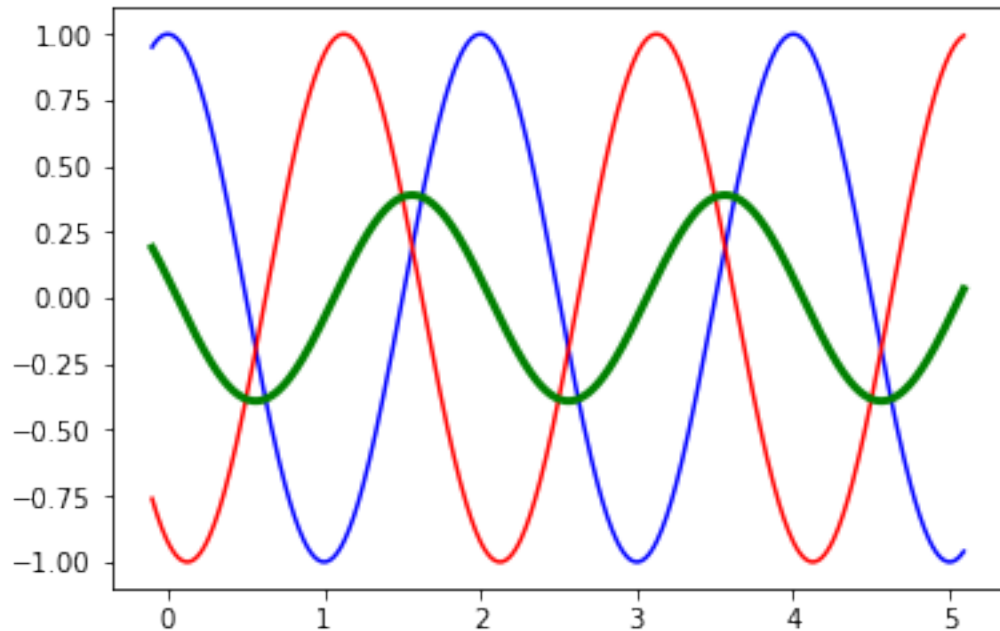


Mais si on introduit de nouveau un petit déphasage supplémentaire à $\pi/2$, alors l'onde résultante croît une fois de plus.

```
In [22]: Phi = np.pi + np.pi / 8

y1 = 1.0*np.cos(omega * t - k * x)
y2 = 1.0*np.cos(omega * t - k * x + Phi)
y = y1 + y2

plt.plot(x, y1, 'b-')
plt.plot(x, y2, 'r-')
plt.plot(x, y, 'g', linewidth=3)
plt.show()
```



1.0.5 Composition d'ondes avec des longueurs d'onde différentes

On peut aussi trouver dans la nature des ondes superposées avec des amplitudes, des fréquences ou même des vitesses de propagation (et donc des longueurs d'onde) différentes. Leurs compositions peuvent alors donner des résultats surprenants.

```
In [24]: y = 1.0*np.cos(omega * t - k * x) \
          - 0.333*np.cos(omega * t - k * 3 * x) \
          + 0.2*np.cos(omega * t - k * 5 * x) \
          - 0.143*np.cos(omega * t - k * 7 * x)

plt.plot(x, y, 'g')
plt.show()
```

