In [1]:
```python
from IPython.display import IFrame
import os
import numpy as np
import matplotlib.pyplot as plt
import matplotlib.ticker as ticker
```

3D Stokes parameters

Sea $\vec{E}(\vec{r},z)$ un campo cualquiera, se pueden considerar su parte real e imaginaria por separado

$$\vec{E}(\vec{r},z) = \vec{A}(\vec{r},z) + i\vec{B}(\vec{r},z) \tag{1}$$

Se puede demostrar que existen dos vectores $\vec{P}(\vec{r},z)$ y $\vec{Q}(\vec{r},z)$ ortogonales entre sí que vienen dados por

$$\vec{P}(\vec{r},z) = \cos[\alpha(\vec{r},z)]\cdot\vec{A}(\vec{r},z) + \sin[\alpha(\vec{r},z)]\cdot\vec{B}(\vec{r},z) \tag{2}$$

$$\vec{Q}(\vec{r},z) = \cos[\alpha(\vec{r},z)]\cdot\vec{B}(\vec{r},z) - \sin[\alpha(\vec{r},z)]\cdot\vec{A}(\vec{r},z) \tag{3}$$

donde

$$\tan[\alpha(\vec{r},z)] = \frac{2\,\vec{A}\cdot\vec{B}}{|\vec{A}|^2 - |\vec{B}|^2} \tag{4}$$

Entonces $\vec{P}$ i $\vec{Q}$ y $\vec{N} = \vec{P}\times\vec{Q}$ son un conjunto de tres vectores ortogonales en cada punto del haz. El campo puede escrivirse como

$$E_P = \frac{\vec{E}\cdot\vec{P}}{|\vec{P}|} = |\vec{P}|\,e^{i\alpha} \tag{5}$$

$$E_Q = \frac{\vec{E}\cdot\vec{P}}{|\vec{P}|} = |\vec{P}|\,e^{i\alpha} \tag{6}$$

$$E_N = 0 \quad ; \quad \forall\,(\vec{r},z) \tag{7}$$

Es decir, el campo está contenido en el plano PQ y $\vec{N}$ es la normal a dicho plano. Entonces, el campo será, por lo general, elipticamente polarizado referido a los ejes $\vec{P}$ i $\vec{Q}$. Notese que $\vec{A}\times\vec{B} = \vec{P}\times\vec{Q}$

Se puede demostrar que los parametros de Stokes del campo $\vec{E}$ en cada punto, y referidos a dicho plano PQ pueden calcularse como

$$S_0(\vec{r},z) = |\vec{E}(\vec{r},z)|^2 = |\vec{A}(\vec{r},z)|^2 + |\vec{B}(\vec{r},z)|^2 \tag{8}$$

$$S_1(\vec{r},z) = \frac{|\vec{A}(\vec{r},z)|^2 - |\vec{B}(\vec{r},z)|^2}{\cos[2\,\alpha(\vec{r},z)]} \tag{9}$$

$$S_2(\vec{r},z) = 0 \tag{10}$$

$$S_3(\vec{r},z) = 2\left|\vec{A}(\vec{r},z)\times\vec{B}(\vec{r},z)\right| \tag{11}$$

Campo incidente circularmente polarizado

Se ensaya con el siguiente campo incidente circularmente polarizado a derechas

$$\vec{E}_S^{circ}(\theta,\varphi) = \frac{1}{\sqrt{2}}g(\theta)\,(1,i) \tag{12}$$

donde

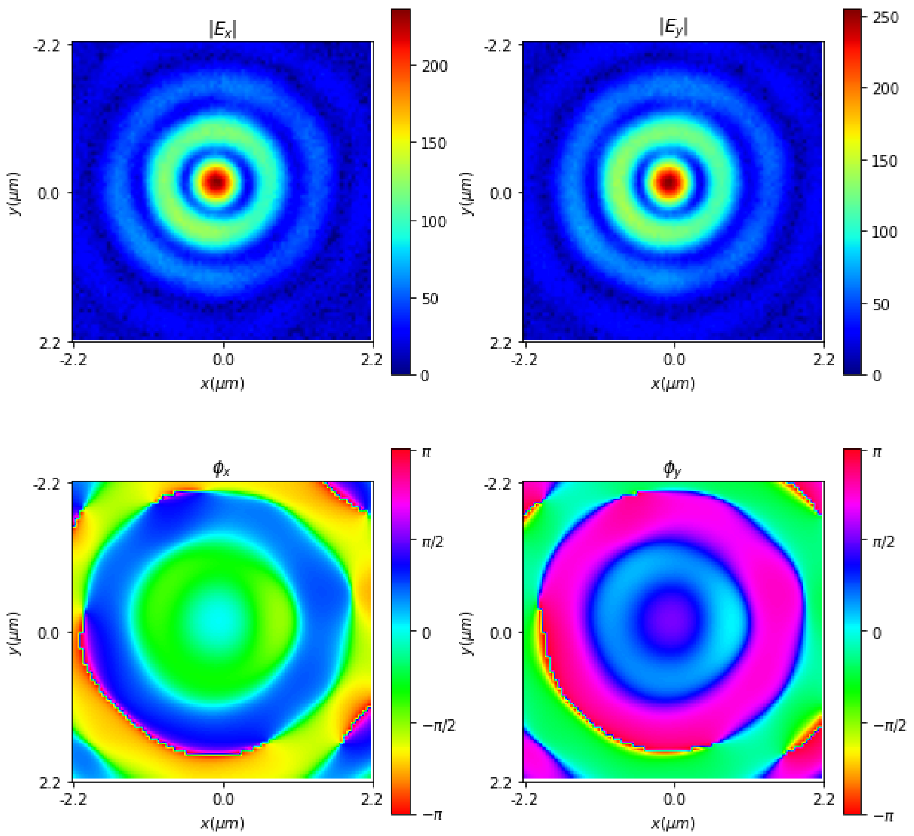
$$g(\theta) = \frac{\exp\left\{-\frac{\sigma(\cos\theta-\bar{\alpha})}{2\,(1-\alpha_0)^2}\right\}}{\pi\sqrt{\cos\theta}\,(1+\cos\theta)} \tag{13}$$

In [2]:
```python
os.chdir(r"C:\Users\dmaluenda\OneDrive - Universitat de Barcelona\Research\WorkInProgress\3Dpolarization\experimental

# print(list(np.load("Dextro/amplitudes.npz").keys()))
with np.load("Dextro/amplitudes.npz") as file:
    dextro_Ax = file['Ax']
    dextro_Ay = file['Ay']
    dextro_p = file['p']

# print(list(np.load("Dextro/phases.npz").keys()))
```

```python
with np.load("Dextro/phases.npz") as file:
    dextro_phx = file['phi_x']
    dextro_phy = file['phi_y']
    dextro_ros = file['ros']
```

In [3]:
```python
def plot_transversal_field(Ax, Ay, phx, phy, trim=None):

    scale = 58.5  # nm/px
    fig, axs = plt.subplots(2,2, figsize=(10,10))
    for idx, ax in enumerate(axs.flatten()):
        ax.set_aspect('equal')
        ax.set_xlabel(r'$x (\mu m)$')
        ax.set_ylabel(r'$y (\mu m)$')

        if idx == 0:
            im = ax.imshow(Ax[trim:-trim,trim:-trim], cmap='jet')
            ax.set_title(r'$|E_x|$')
            lims = Ax[trim:-trim,trim:-trim].shape
        elif idx == 1:
            im = ax.imshow(Ay[trim:-trim,trim:-trim], cmap='jet')
            ax.set_title(r'$|E_y|$')
        elif idx == 2:
            im = ax.imshow(np.angle(phx)[trim:-trim,trim:-trim], cmap='hsv')
            ax.set_title(r'$\phi_x$')
        elif idx == 3:
            im = ax.imshow(np.angle(phy)[trim:-trim,trim:-trim], cmap='hsv')
            ax.set_title(r'$\phi_y$')

        ax.set_xticks([0,lims[0]//2,lims[0]])
        ax.set_yticks([0,lims[1]//2,lims[1]])
        ticks = [-lims[0]/2*scale/1000, 0, lims[0]/2*scale/1000]
        format_ticks = ticker.FormatStrFormatter('%.1f').format_ticks(ticks)
        ax.set_xticklabels(format_ticks)
        ax.set_yticklabels(format_ticks)
        if idx in [2,3]:
            cbar = fig.colorbar(im, ax=ax, orientation='vertical',
                                ticks=[-np.pi+0.01,-np.pi/2, 0, np.pi/2, np.pi-0.05])
            cbar.ax.set_yticklabels([r'$-\pi$', r'$-\pi/2$', r'$0$', r'$\pi/2$', r'$\pi$'])
        else:
            cbar = fig.colorbar(im, ax=ax, orientation='vertical')

    # plt.subplots_adjust(left=0.1, bottom=0.1, right=0.9, top=0.9,
    #                     wspace=0.2, hspace=0.2)
    plt.show()
    return lims
```

In [4]:
```python
_ =plot_transversal_field(dextro_Ax, dextro_Ay, dextro_phx, dextro_phy, trim=int(2**6.5))
```



La figura anterior muestra el campo transversal en el plano focal. Mientras que la siguiente muestra el campo longitudinal en el plano focal calculado a través de [cita al SciRep]

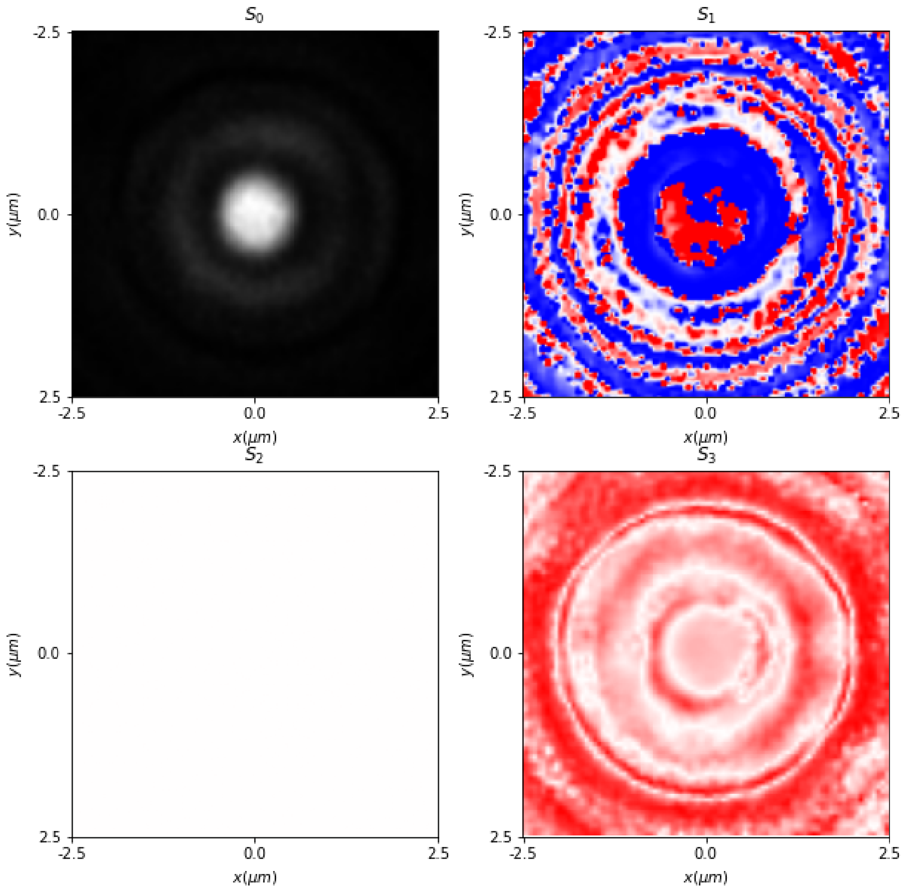
$$\mathcal{F}\left\{E_z(\vec{k}_\perp; z=0)\right\} = -\frac{\vec{k}_\perp \cdot \mathcal{F}\left\{\vec{E}_\perp(\vec{k}_\perp; z=0)\right\}}{k_z} \tag{14}$$

circular_longitudinal

Calculamos los parametros de Stokes según las ecuaciones de (8) a (11) y mostramos los resultados en la siguiente figura

In [5]:
```
def plot_3D_stokes(*stokes_fn):
    fig, axs = plt.subplots(2,2, figsize=(10,10))
    axs[1,1].imshow(np.ones((10,10)), cmap='seismic')
    for idx, ax in enumerate(axs.flatten()):
        ax.set_aspect('equal')
        ax.set_xlabel(r'$x (\mu m)$')
        ax.set_ylabel(r'$y (\mu m)$')

        s_i = plt.imread(stokes_fn[idx])
        im = ax.imshow(s_i)
        ax.set_title(fr'$S_{idx}$')

        lims = s_i.shape
        ax.set_xticks([0,lims[0]//2,lims[0]])
        ax.set_yticks([0,lims[1]//2,lims[1]])
        ticks = [-2.5, 0, 2.5]
        format_ticks = ticker.FormatStrFormatter('%.1f').format_ticks(ticks)
        ax.set_xticklabels(format_ticks)
        ax.set_yticklabels(format_ticks)

#     fig.colorbar(im, ax=axs.ravel().tolist(), cmap='seismic')

    plt.show()
    return lims
```

In [6]:
```
_ = plot_3D_stokes(*[f'Dextro/s{i}_pDex.png' for i in range(4)])
```

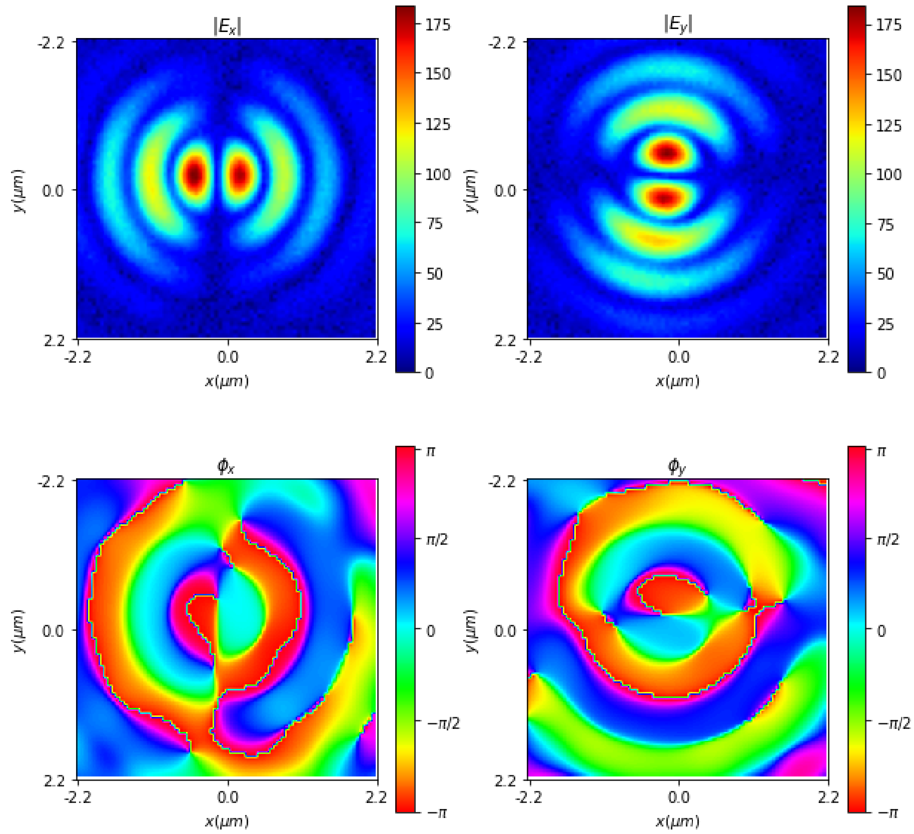


Campo incidente radialmente polarizado (sin singularidad central)

$$\vec{E}_S^{rad}(\theta, \varphi) = \frac{1}{\sqrt{2}} g(\theta) \left(\cos\varphi, \sin\varphi\right) \tag{15}$$

A continuación mostramos el campo transversal en el plano focal para el caso de campo incidente radialmente polarizado

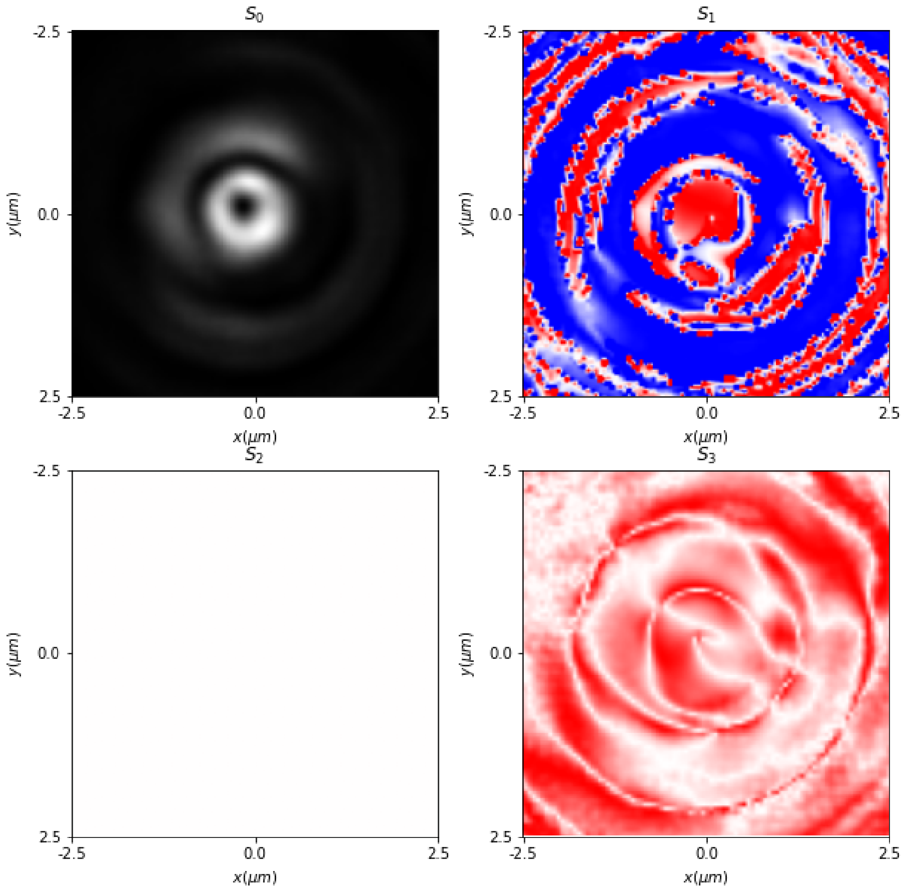
In [7]:
```
os.chdir(r"C:\Users\dmaluenda\OneDrive - Universitat de Barcelona\Research\WorkInProgress\3Dpolarization\experimental
with np.load("Radial/amplitudes.npz") as file:
    radial_Ax = file['Ax']
    radial_Ay = file['Ay']
    radial_p = file['p']

with np.load("Radial/phases.npz") as file:
    radial_phx = file['phi_x']
    radial_phy = file['phi_y']
    radial_ros = file['ros']

_ = plot_transversal_field(radial_Ax, radial_Ay, radial_phx, radial_phy, trim=int(2**6.5))
```



Se calcula la componente longitudinal del campo en el plano focal segun (14) radial_longitudinal

Calculamos los parametros de Stokes según las ecuacones de (8) a (11) y mostramos los resultados en la siguiente figura

In [8]:
```
_ = plot_3D_stokes(*[f'Radial/s{i}_pRad.png' for i in range(4)])
```




```
In [ ]:
```