# Assume nx, ny same as kappa shape

nx, ny = kappa.shape

# Define image plane grid (theta\_x, theta\_y)

theta\_x, theta\_y = np.meshgrid(np.arange(nx), np.arange(ny))

# Deflection angles already computed: alpha\_x, alpha\_y

# Compute source plane coordinates (beta\_x, beta\_y)

beta\_x = theta\_x - alpha\_x

beta\_y = theta\_y - alpha\_y

# For visualization, plot how a grid would map to the source plane

plt.figure(figsize=(10,10))

plt.scatter(beta\_x[::50, ::50], beta\_y[::50, ::50], s=1, color='blue')

plt.title("Mapped Source Plane Coordinates (Sampled Every 50 Pixels)")

plt.xlabel("Beta X")

plt.ylabel("Beta Y")

plt.gca().invert\_yaxis()

plt.show()