



BENEMÉRITA UNIVERSIDAD AUTÓNOMA DE PUEBLA

FACULTAD DE CIENCIAS FÍSICO MATEMÁTICAS

MODULACIÓN DE AMPLITUD VECTORIAL FUERA DE  
FASE

T E S I S

PRESENTADA PARA OBTENER EL TÍTULO DE:

**Licenciado en Física Aplicada**

PRESENTA:

**Carlos Augusto Flores Meneses**

DIRECTOR:

Dr. Cruz Meneses Fabián



Julio, 2018







# Resumen

---

Se muestra un estudio sobre el caso vectorial del método de modulación de amplitud fuera de fase, conocido también como modulación de amplitud en cuadratura y no-cuadratura. Este método consiste en sumar dos campos polarizados arbitrariamente fuera de fase y variar sus amplitudes sin modificar sus estados de polarización. Se demuestra que el campo resultante puede ser modulado en amplitud y frente de onda de manera independiente, mientras que se encuentra una dependencia del estado de polarización resultante con el frente de onda. De esta manera el presente método promete modular un campo en solo amplitud (AM), solo fase con una modulación de polarización intrínseca (PoPM) y amplitud, fase y polarización intrínseca (PoPAM).

**Palabras Clave:** *polarizacion, NQAM, elipse, superposicion, campo optico, luz polarizada.*



# Índice general

---

Índice de figuras	VII
Índice de tablas	IX
<b>1. Introducción</b>	<b>1</b>
<b>2. Antecedentes</b>	<b>3</b>
2.1. Descripción matemática de los estados de polarización . . . . .	3
2.1.1. Polarización lineal . . . . .	3
2.1.2. Polarización circular . . . . .	4
2.1.3. Polarización elíptica . . . . .	6
2.1.4. Los Parámetros de Stokes . . . . .	8
2.1.5. Esfera de Poincaré . . . . .	12
<b>3. Modulación de amplitud vectorial</b>	<b>13</b>
3.1. Modulación de amplitud fuera de fase: caso escalar . . . . .	13
3.1.1. Tipos de modulación . . . . .	14
3.2. Modulación de amplitud vectorial fuera de fase . . . . .	17
3.2.1. Modulación de amplitud vectorial: tipos de modulación . . . . .	21
<b>4. Simulación de estados de polarización</b>	<b>27</b>
4.1. Simulación de polarización lineal . . . . .	27
4.2. Simulación de polarización circular . . . . .	33
4.3. Simulación de polarización elíptica . . . . .	38
4.4. Campo Vectorial . . . . .	43
<b>5. Conclusiones</b>	<b>45</b>
5.1. Trabajo a futuro . . . . .	46
<b>A. Simulación de la superposición de dos estados de polarización</b>	<b>47</b>
<b>B. Animación de la superposición de dos estados de polarización</b>	<b>53</b>
<b>C. Simulación de la superposición de dos frentes de onda</b>	<b>59</b>

## ÍNDICE GENERAL

---

<b>D. Simulación de los parámetros de Stokes mapeados a la esfera de Poincare</b>	<b>61</b>
<b>E. Simulación de las relaciones inversas del estado resultante de dos estados de polarización</b>	<b>65</b>
<b>F. Modulación de amplitud mediante NQAM</b>	<b>69</b>
<b>G. Modulación de amplitud vectorial mediante NQAM</b>	<b>75</b>
<b>Bibliografía</b>	<b>79</b>

# Índice de figuras

---

2.1. Polarización lineal, viajando en dirección $z$ . . . . .	4
2.2. Polarización circular viajando en dirección $z$ . . . . .	5
2.3. Luz elíptica . . . . .	7
2.4. Polarización elíptica . . . . .	8
2.5. Esfera de Poincaré . . . . .	12
3.1. Modulación de amplitud en no-cuadratura en modulación de solo amplitud.	15
3.2. Modulación de amplitud no-cuadratura en modulación de solo con fase.	16
3.3. Modulación de amplitud en no-cuadratura en modulación de fase y amplitud . . . . .	17
3.4. Dos estados de polarización elípticos. El primer estado $\vec{E}_1$ tiene como valores $\alpha_{1x} = \pi/5$ , $A_{1x} = 3.0$ , $\alpha_{1y} = \pi/5$ y $A_{1y} = 3$ . El segundo estado $\vec{E}_2$ tiene como valores $\alpha_{2x} = \pi/2$ , $A_{2x} = 3$ , $\alpha_{2y} = \pi/7$ y $A_{2y} = 3$ y el resultado de su superposición $\vec{E}$ . . . . .	18
3.5. Modulación de solo amplitud vectorial (AM) . . . . .	22
3.6. Modulación de amplitud vectorial mapeada a la esfera de Poincaré . . . . .	23
3.7. Modulación de polarización y fase vectorial (PoPM) . . . . .	24
3.8. Modulación de polarización y fase mapeada a la esfera de Poincaré . . . . .	24
3.9. Modulación de polarización, fase y amplitud vectorial (POPAM) . . . . .	25
3.10. Modulación de polarización, fase y amplitud vectorial mapeada a la esfera de Poincaré . . . . .	26
4.1. Superposición de dos estados ortogonales con polarización lineal . . . . .	27
4.2. Amplitud resultante $A$ , variando $A_1$ y $A_2$ en el intervalo de $(-5, 5)$ manteniendo $\Delta\alpha_x = \pi/3$ . . . . .	28
4.3. $\sigma$ , variando $A_1$ y $A_2$ en el intervalo de $(-5, 5)$ manteniendo $\Delta\alpha_x = \pi/3$ . . . . .	28
4.4. $\alpha$ , variando $A_1$ y $A_2$ en el intervalo de $(-5, 5)$ manteniendo $\Delta\alpha_x = \pi/3$ . . . . .	29
4.5. $\alpha_x$ , variando $A_1$ y $A_2$ en el intervalo de $(-5, 5)$ manteniendo $\Delta\alpha_x = \pi/3$ . . . . .	29
4.6. Esfera de Poincaré variando $A_1$ y $A_2$ en el intervalo de $(-5, 5)$ manteniendo $\Delta\alpha_x = \pi/3$ . . . . .	30
4.7. $A$ , $\sigma$ , $\alpha$ , y $\alpha_x$ cuando $\Delta\alpha_x$ varía en el intervalo $(-\pi, \pi)$ y con $A_1 = A_2 = 3$ . . . . .	31
4.8. Esfera de Poincaré cuando $\Delta\alpha_x$ varía en el intervalo $(-\pi, \pi)$ y con $A_1 = A_2 = 3$ . . . . .	32

## ÍNDICE DE FIGURAS

---

4.9. Esfera de Poincaré, con $A_1, A_2$ variando en $(-5, 5)$ y $\Delta\alpha_x$ variando en $(-\pi, \pi)$ . . . . .	33
4.10. Superposición de dos estados de polarización circular a derechas y a izquierdas respectivamente . . . . .	34
4.11. Amplitud resultante $A$ , variando $A_1$ y $A_2$ en el intervalo de $(-5, 5)$ . . . . .	34
4.12. $\sigma$ variando $A_1$ y $A_2$ en el intervalo de $(-5, 5)$ . . . . .	35
4.13. $\alpha$ variando $A_1$ y $A_2$ en el intervalo de $(-5, 5)$ . . . . .	35
4.14. $\alpha_x$ variando $A_1$ y $A_2$ en el intervalo de $(-5, 5)$ . . . . .	36
4.15. $A, \sigma, \alpha$ , y $\alpha_x$ cuando $\Delta\alpha_x$ varia en $(-\pi, \pi)$ y con $A_1 = A_2 = 3$ . . . . .	36
4.16. Esfera de Poincaré cuando $\Delta\alpha_x$ varia en $(-\pi, \pi)$ y con $A_1 = A_2 = 3$ . . . . .	37
4.17. Esfera de Poincaré cuando $\Delta\alpha_x$ varia en $(-\pi, \pi)$ y con $A_1$ y $A_2$ varian en $(-5, 5)$ . . . . .	38
4.18. Superposición de dos estados elípticos. . . . .	39
4.19. $A$ con $A_1$ y $A_2$ variables en el intervalo de $(-5, 5)$ . . . . .	39
4.20. $\sigma$ con $A_1$ y $A_2$ variables en el intervalo de $(-5, 5)$ . . . . .	40
4.21. $\alpha$ con $A_1$ y $A_2$ variables en el intervalo de $(-5, 5)$ . . . . .	40
4.22. $\alpha_x$ con $A_1$ y $A_2$ variables en el intervalo de $(-5, 5)$ . . . . .	41
4.23. Esfera de Poincaré con $A_1$ y $A_2$ variables en el intervalo de $(-5, 5)$ . . . . .	41
4.24. $A, \sigma, \alpha$ , y $\alpha_x$ cuando $A_1 = 4$ y $A_2 = 2$ y $\Delta\alpha_x$ variable en $(-\pi, \pi)$ : . . . . .	42
4.25. Esfera de Poincaré con $A_1 = 4, A_2 = 2$ y $\Delta\alpha_x$ variable en $(-\pi, \pi)$ . . . . .	42
4.26. Esfera de Poincaré con $A_1, A_2$ variando en $(-5, 5)$ y $\Delta\alpha_x$ variable en $(-\pi, \pi)$ . . . . .	43
4.27. Superposición de dos frentes de onda, la primera figura representa un frente onda con los parámetros descritos, la siguiente otro frente de onda y la tercera la superposición de estos . . . . .	44

# Índice de tablas

---

2.1. Vectores de Stokes para algunos estados . . . . .	11
--	----



---

## Capítulo 1

# Introducción

---

La polarización de la luz es uno de los fenómenos más notables en la naturaleza y ha llevado a numerosos descubrimientos y aplicaciones. Hoy continúa desempeñando un papel vital en la óptica. Tener conocimientos en el campo de la polarización ayuda a comprender mejor la teoría de la luz en general, al tiempo que contribuye a corregir la exagerada importancia que se concede al tratamiento geométrico de la luz. Los fenómenos relacionados con la polarización no sólo son fáciles de demostrar. Además de su valor académico, el estudio de la polarización clarifica muchas aplicaciones prácticas que se están desarrollando para el uso de la luz polarizada. (1)

Comprender y manipular la polarización de la luz es crucial para muchas aplicaciones ópticas. El diseño óptico frecuentemente se enfoca en la longitud de onda y la intensidad de la luz, mientras descuida su polarización. La polarización, sin embargo, es una propiedad importante de la luz que afecta incluso a aquellos sistemas ópticos que no la miden explícitamente. La polarización de la luz afecta el enfoque de los rayos láser, influye en las longitudes de onda de corte de los filtros y puede ser importante para evitar reflejos secundarios no deseados. Es esencial para muchas aplicaciones de metrología, como análisis de estrés en vidrio o plástico, análisis de ingredientes farmacéuticos y microscopía biológica. Las diferentes polarizaciones de la luz también pueden ser absorbidas en diferentes grados por los materiales, una propiedad esencial para pantallas LCD, películas en 3D y gafas de sol que reducen el deslumbramiento.

A fin de explicar el comportamiento de la luz, hay una gran variedad de conceptos útiles, aunque sean diferentes. Para algunos fenómenos, es suficiente, y a veces incluso necesario, que se piense en la luz como una corriente de partículas (teoría corpuscular); para otros, es mejor imaginar la luz similar a las ondas de sonido (ondas longitudinales). La teoría probablemente más sofisticada y en general más aplicable requiere que se trate la luz como un complicado movimiento de ondas transversales. La polarización de la luz puede entenderse aplicando este último concepto.

Una onda transversal es una onda en movimiento que consiste en oscilaciones que

## 1. INTRODUCCIÓN

---

ocurren perpendiculares (en ángulo recto) a la dirección de propagación de la onda. Si una onda transversal se mueve en la dirección  $x$  positiva, sus oscilaciones están en direcciones ascendente y descendente que se encuentran en el plano  $y-z$ . La luz es un ejemplo de una onda transversal, mientras que el sonido es una onda longitudinal. Una ondulación en un estanque y una ola en una cuerda se visualizan fácilmente como ondas transversales. (2)

## Antecedentes

---

### 2.1. Descripción matemática de los estados de polarización

#### 2.1.1. Polarización lineal

Para la descripción de la polarización lineal en primer lugar consideremos dos perturbaciones ópticas ortogonales, en la forma:

$$\vec{E}_x(z, t) = \hat{i}E_{0x} \cos(kz - \omega t), \quad (2.1)$$

y

$$\vec{E}_y(z, t) = \hat{j}E_{0y} \cos(kz - \omega t + \alpha), \quad (2.2)$$

Aquí  $\alpha$  representa la diferencia de fase entre las ondas ambas viajando en la dirección de  $z$ . Ya que la fase tiene la forma  $kz - \omega t$  el sumar un  $\alpha$  *positivo* tiene como significado que la función coseno de (2.1) no podrá alcanzar el mismo valor que el coseno de (2.2) si no hasta que tome el valor de  $\alpha/\omega$ . Por tanto,  $E_y$  queda detrás de  $E_x$  en  $\alpha > 0$ . Ahora, si  $\alpha$  es una cantidad negativa,  $E_y$  precede a  $E_x$  en  $\alpha < 0$ . La perturbación óptica resultante es la suma vectorial de estas dos ondas perpendiculares:

$$\vec{E}(z, t) = \vec{E}_x(z, t) + \vec{E}_y(z, t). \quad (2.3)$$

Si  $\alpha$  es cero o un múltiplo entero de  $\pm 2\pi$  se dice que las ondas están en fase, en este caso la ecuación (2.3) se escribe como:

$$\vec{E} = (\hat{i}E_{0x} + \hat{j}E_{0y}) \cos(kz - \omega t). \quad (2.4)$$

La onda resultante tiene por consiguiente una amplitud fija igual a  $(\hat{i}E_{0x} + \hat{j}E_{0y})$ , es decir esta **linealmente polarizada** (3).

## 2. ANTECEDENTES

---

Las ondas avanzan hacia un plano de observación cuyos campos tendrán que medirse. Ahí se podrá ver una onda resultante  $\vec{E}$  que oscila a lo largo de una línea inclinada, según una cosinusoidde en el tiempo. Este proceso se puede llevar igualmente a cabo a la inversa es decir, podemos resolver cualquier onda linealmente polarizada en dos componentes ortogonales.

Supongamos ahora que  $\alpha$  es un múltiplo entero impar de  $\pm\pi$ . Las dos ondas están desfasadas  $180^\circ$ :

$$\vec{E} = (\hat{i}E_{0x} + \hat{j}E_{0y}) \cos(kz - \omega t), \quad (2.5)$$

esta onda esta de nuevo linealmente polarizada pero se ha girado el plano de vibración.

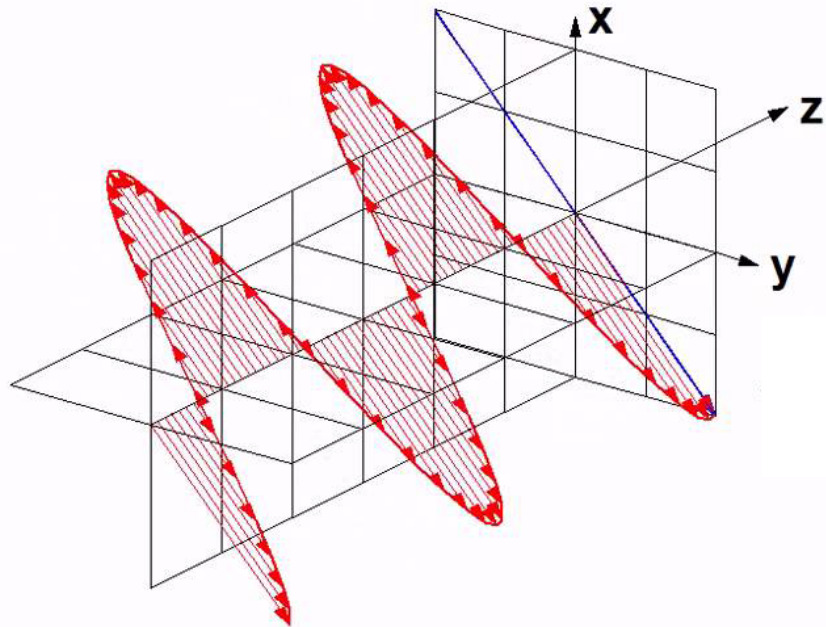


Figura 2.1: Polarización lineal, viajando en dirección z

### 2.1.2. Polarización circular

Consideremos ahora que ambas ondas tienen igual amplitud,  $E_{0x} = E_{0y} = E_0$  y además su diferencia de fase relativa es  $\alpha = -\pi/2 + 2m\pi$  donde  $m = 0, \pm 1, \pm 2, \dots$ . Por tanto:

$$\vec{E}_x(z, t) = \hat{i}E_0 \cos(kz - \omega t), \quad (2.6)$$

y

$$\vec{E}_y(z, t) = \hat{j}E_0 \sin(kz - \omega t), \quad (2.7)$$

la onda resultante es entonces:

$$\vec{E} = E_0[\hat{i} \cos(kz - \omega t) + \hat{j} \sin(kz - \omega t)]. \quad (2.8)$$

Note que ahora la amplitud escalar de  $\vec{E}$  es  $(\vec{E} \cdot \vec{E})^{1/2} = E_0$  es una constante. Pero la dirección de  $\vec{E}$  es variable con el tiempo y no está restringida, como antes, a un solo plano. La polarización circular puede clasificarse en dos, **polarización circular a izquierdas** y **polarización circular a derechas**. La polarización a derechas ocurre en este caso tomando a  $\alpha = -\pi/2 + 2m\pi$  donde  $m = 0, \pm 1, \pm 2, \dots$ . En comparación si  $\alpha = \pi/2, 5\pi/2, 9\pi/2$ , etc, es decir  $\alpha = \pi/2 \pm 2m\pi$  donde  $m = 0, \pm 1, \pm 2, \dots$  entonces:

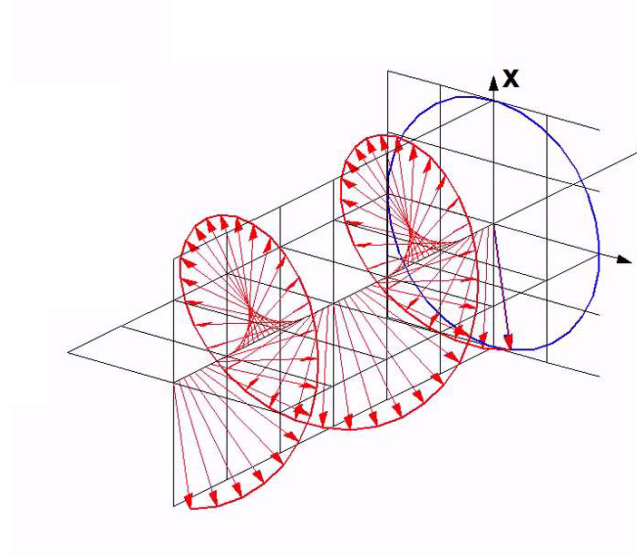
$$\vec{E} = E_0[\hat{i} \cos(kz - \omega t) - \hat{j} \sin(kz - \omega t)]. \quad (2.9)$$

La amplitud no se ve afectada pero  $\vec{E}$  gira a izquierdas, y la onda tiene polarización circular a izquierdas.

Una onda linealmente polarizada se puede sintetizar partiendo de dos ondas con polarización circular opuesta de igual amplitud. En concreto, si sumamos la onda circular a derechas y la onda circular izquierdas obtenemos:

$$\vec{E} = 2E_0\hat{i} \cos(kz - \omega t), \quad (2.10)$$

que tiene un vector de amplitud constante  $2E_0\hat{i}$  siendo por consiguiente linealmente polarizado.



**Figura 2.2:** Polarización circular viajando en dirección z

### 2.1.3. Polarización elíptica

Por lo que a la descripción matemática se refiere, tanto la luz lineal como la circular se pueden considerar como casos especiales de la luz elípticamente polarizada o simplemente luz elíptica. Esto significa que en general, el vector de campo eléctrico resultante  $\vec{E}$  girará cambiando también su magnitud. En tales casos el extremo de  $\vec{E}$  trazara una elipse, en un plano fijo perpendicular a  $\vec{k}$ , cuando la onda avanza. Podemos verlo escribiendo una expresión para la curva trazada por la punta de  $\vec{E}$ . Con este fin recordemos que

$$E_x = E_{0x} \cos(kz - \omega t), \quad (2.11)$$

y

$$E_y = E_{0y} \cos(kz + \omega t + \alpha). \quad (2.12)$$

La ecuación de la curva buscada no debe ser función ni de la posición ni del tiempo, es decir, debemos poder librarnos de la dependencia de  $(kz - \omega t)$ . Desarrollemos la expresión para  $E_y$  en:

$$\frac{E_y}{E_{0y}} = \cos(kz - \omega t) \cos \alpha - \sin(kz - \omega t) \sin \alpha,$$

y combinémosla con  $E/E_0$  y obtenemos:

$$\frac{E_y}{E_{0y}} - \frac{E_x}{E_{0x}} \cos \alpha = -\sin(kz - \omega t) \sin \alpha, \quad (2.13)$$

de la ecuación (2.11) se deduce que:

$$\sin(kz - \omega t) = [1 - (E_x/E_{0x})^2]^{1/2},$$

y la ecuación (2.13) nos lleva a

$$\left( \frac{E_y}{E_{0y}} - \frac{E_x}{E_{0x}} \cos \alpha \right)^2 = \left[ 1 - \left( \frac{E_x}{E_{0x}} \right)^2 \right] \sin^2 \alpha.$$

Finalmente ordenando los términos tenemos que

$$\left( \frac{E_y}{E_{0y}} \right)^2 + \left( \frac{E_x}{E_{0x}} \right)^2 - 2 \left( \frac{E_x}{E_{0x}} \right) \left( \frac{E_y}{E_{0y}} \right) \cos \alpha = \sin^2 \alpha. \quad (2.14)$$

Esta es la ecuación de una elipse que forma un ángulo  $\psi$  con el sistema coordenado  $(E_x, E_y)$  tal que:

$$\tan 2\psi = \frac{2E_{0x}E_{0y} \cos \alpha}{E_{0x}^2 - E_{0y}^2}, \quad (2.15)$$

donde  $\psi \in (-\pi/2, \pi/2)$  y  $\alpha \in (-\pi, \pi)$ , definimos:

$$\tan \sigma = \frac{E_{0y}}{E_{0x}}, \quad (2.16)$$

donde  $\sigma \in (0, \pi/2)$  es conocido como el ángulo auxiliar, de forma trigonométrica:

$$\tan 2\psi = \tan(2\sigma) \cos \alpha. \quad (2.17)$$

Para calcular el ángulo de elipticidad depende del eje mayor  $E'_{0y}$  y el eje menor  $E'_{0x}$ :

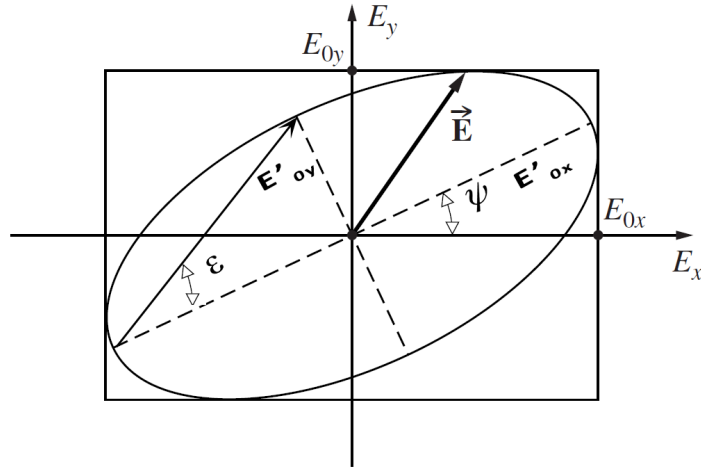
$$\tan(\varepsilon) = \frac{E'_{0y}}{E'_{0x}}, \quad (2.18)$$

donde  $\varepsilon \in (-\pi/4, \pi/4)$ ,

$$\sin(2\varepsilon) = \sin(2\sigma) \sin \alpha, \quad (2.19)$$

la elipticidad es determinada por

$$e = \tan \varepsilon. \quad (2.20)$$



**Figura 2.3:** Luz elíptica

Es posible probar estas dos ecuaciones donde  $\psi$  y  $\varepsilon$  están relacionadas:

$$\cos(2\sigma) = \cos(2\psi) \cos(2\varepsilon) \quad (2.21)$$

$$\tan \alpha = \csc(2\psi) \tan(2\varepsilon) \quad (2.22)$$

La ecuación (2.14) podría reconocerse un poco más si los ejes principales de la elipse estuvieran alineados con los ejes coordenados, es decir  $\psi = 0$  o de manera equivalente,  $\alpha = \pm\pi/2, \pm3\pi/2, \dots$  en cuyo caso tenemos la forma canónica

$$\frac{E_y^2}{E_{0y}^2} + \frac{E_x^2}{E_{0x}^2} = 1. \quad (2.23)$$

## 2. ANTECEDENTES

---

Ademas, si  $E_{0y} = E_{0x} = E_0$  esto se reduce a

$$E_y^2 + E_x^2 = E_0^2, \quad (2.24)$$

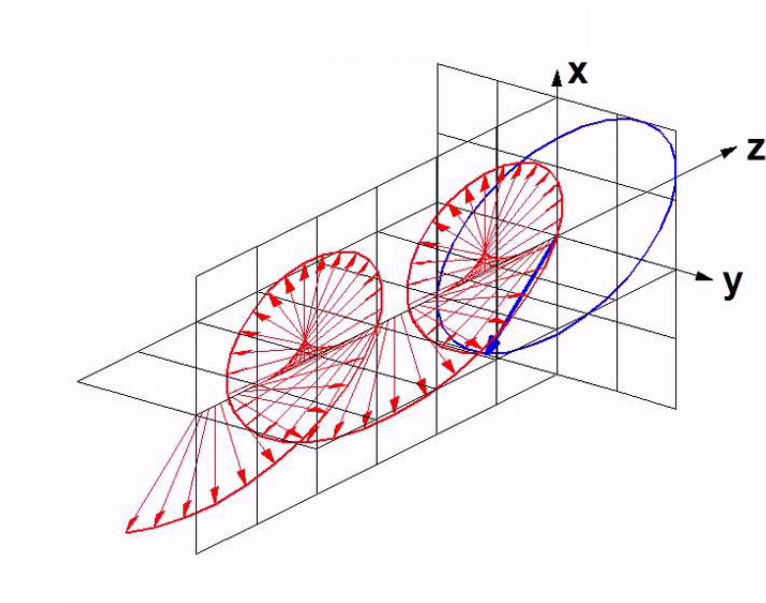
lo cual de acuerdo a los primeros resultados, es un círculo. Si  $\alpha$  es un múltiplo de  $\pi$  la ecuación (2.14) resulta en:

$$E_y = \frac{E_{0y}}{E_{0x}} E_x, \quad (2.25)$$

de manera similar para múltiplos impares de  $\pi$

$$E_y = -\frac{E_{0y}}{E_{0x}} E_x. \quad (2.26)$$

Ambas son líneas rectas con pendiente  $\pm E_{0y}/E_{0x}$  es decir se tiene luz linealmente polarizada.



**Figura 2.4:** Polarización elíptica

### 2.1.4. Los Parámetros de Stokes

La representación moderna de la luz polarizada en realidad se remota al año 1852 al trabajo de G.G. Stokes (3) quien presentó cuatro cantidades que son funciones solamente de las observables de la onda electromagnética y que se denominan ahora parámetros de Stokes. El estado de polarización de un haz de luz se puede describir en términos de estas cantidades.

Imaginemos una serie de cuatro filtros cada uno de los cuales bajo iluminación natural, transmitirá la mitad de la luz incidente descartando la otra mitad. La elección no es única sino que existe un buen número de posibilidades. Supongamos entonces que el primer filtro sea simplemente isótropo, pasado por todos los estados de igual manera, mientras que el segundo y tercero son polarizadores lineales cuyos ejes de transmisión son horizontales y a  $+45^\circ$  (diagonal a lo largo del primer cuadrante) respectivamente. El último filtro es un polarizador circular opaco a los estados a izquierdas. Cada uno de estos cuatro filtros se coloca a solas en la trayectoria del haz que está siendo investigado, midiendo las irradiancias transmitidas  $I_0, I_1, I_2, I_3$  con un medidor insensible a la polarización. La definición operacional de los parámetros de Stokes deduce las relaciones

$$S_0 = 2I_0 \quad (2.27a)$$

$$S_1 = 2I_1 - 2I_0 \quad (2.27b)$$

$$S_2 = 2I_2 - 2I_0 \quad (2.27c)$$

$$S_3 = 2I_3 - 2I_0 \quad (2.27d)$$

Obsérvese que  $S_0$  es simplemente la irradiancia incidente mientras que  $S_1, S_2$  y  $S_3$  especifican el estado de polarización. Entonces  $S_1$  refleja una tendencia de la polarización para asemejarse a un estado lineal horizontal (después de lo cual  $S_1 > 0$ ) o a uno vertical (en cuyo caso  $S_1 < 0$ ) (4). Cuando el haz no muestra ninguna orientación preferencial con respecto a estos ejes ( $S_1 = 0$ ) puede ser elíptico a  $\pm 45^\circ$ , o circular o no polarizado. De manera parecida,  $S_2$  implica la tendencia de la luz a asemejarse a un estado lineal orientado bien sea en la dirección  $+45^\circ$  (cuando  $S_2 > 0$ ) o en la dirección  $-45^\circ$  (cuando  $S_2 < 0$ ) o en ninguna de las dos ( $S_2 = 0$ ) de la misma forma,  $S_3$  revela una tendencia del haz a tener sentido a derechas ( $S_3 > 0$ ), sentido a izquierdas ( $S_3 < 0$ ) o ninguno de los dos ( $S_3 = 0$ ).

Consideremos las siguientes perturbaciones

$$\vec{E}_x(t) = \hat{i}E_{0x}(t) \cos[(kz - wt)] + \alpha_x(t) \quad (2.28a)$$

$$\vec{E}_y(t) = \hat{j}E_{0y}(t) \cos[(kz - wt)] + \alpha_y(t) \quad (2.28b)$$

donde  $\vec{E} = \vec{E}_x(t) + \vec{E}_y(t)$ . Utilizando estos parámetros es posible remodelar (5) como

$$S_0 = \langle E_{0x}^2 \rangle_T + \langle E_{0y}^2 \rangle_T \quad (2.29a)$$

$$S_1 = \langle E_{0x}^2 \rangle_T - \langle E_{0y}^2 \rangle_T \quad (2.29b)$$

$$S_2 = \langle 2E_{0x}E_{0y} \cos \alpha \rangle_T \quad (2.29c)$$

$$S_3 = \langle 2E_{0x}E_{0y} \sin \alpha \rangle_T \quad (2.29d)$$

## 2. ANTECEDENTES

---

Aquí  $\alpha = \alpha_y - \alpha_x$  y hemos eliminado la constante  $\epsilon_0 c/2$  de tal manera que los parámetros son ahora proporcionales a la irradiancias. Observese que para la luz completamente polarizada, se deduce de (2.29) que

$$S_0^2 = S_1^2 + S_2^2 + S_3^2 \quad (2.30)$$

Con esta demostración de la conexión entre los parámetros de Stokes y la elipse de polarización, los parámetros de Stokes pueden escribirse en términos de los parámetros de la elipse de polarización:

$$S_1 = \cos(2\sigma) \cos(2\alpha) \quad (2.31a)$$

$$S_2 = \cos(2\sigma) \sin(2\alpha) \quad (2.31b)$$

$$S_3 = \sin(2\sigma) \quad (2.31c)$$

Vectores de Stokes	Estado de polarización
$\begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \end{bmatrix}$	Polarización lineal horizontal
$\begin{bmatrix} 1 \\ -1 \\ 0 \\ 0 \end{bmatrix}$	Polarización lineal vertical
$\begin{bmatrix} 1 \\ 0 \\ 1 \\ 0 \end{bmatrix}$	Polarización lineal a $+45^\circ$
$\begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \end{bmatrix}$	Polarización lineal a $-45^\circ$
$\begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \end{bmatrix}$	Polarización circular a derechas
$\begin{bmatrix} 1 \\ 0 \\ 0 \\ -1 \end{bmatrix}$	Polarización circular a izquierdas

**Tabla 2.1:** Vectores de Stokes para algunos estados

### 2.1.5. Esfera de Poincaré

Por sí misma, la elipse de polarización es una excelente manera de visualizar la luz polarizada. Sin embargo, es prácticamente imposible determinar los ángulos de orientación y elipticidad que miran la elipse de polarización (6). Además, los cálculos necesarios para determinar los nuevos ángulos de un haz polarizado que se propaga a través de uno o más elementos de polarización son difíciles y tediosos.

Para superar estas dificultades, Poincaré (1892) sugirió usar una esfera ahora conocida como la esfera de Poincaré para representar la luz polarizada. La siguiente figura muestra la esfera de Poincaré y sus coordenadas esféricas y cartesianas. (7)

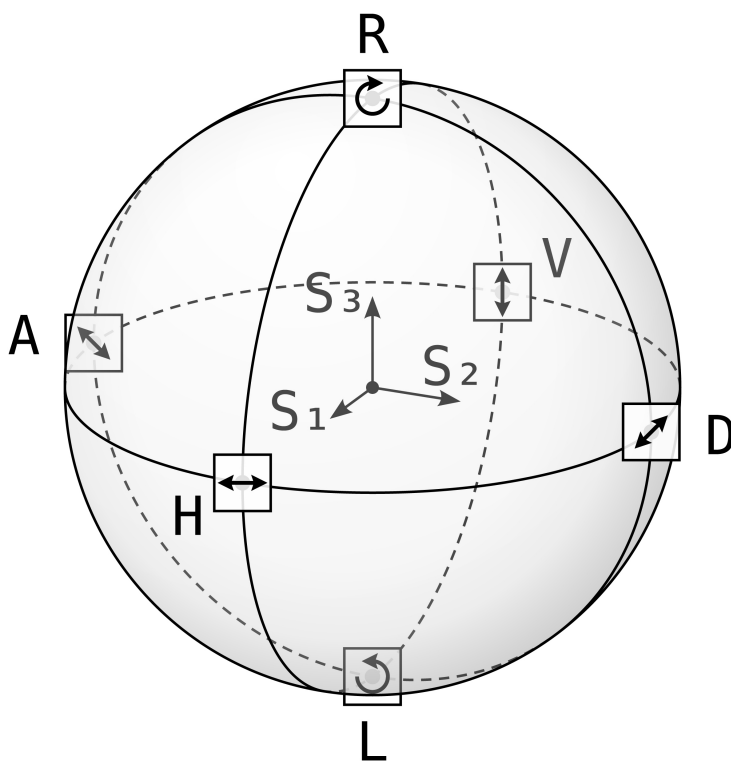


Figura 2.5: Esfera de Poincaré

Los puntos sobre la superficie de la esfera son estados de polarización y están mapeados por los parámetros de Stokes en las ecuaciones (2.31).

## Modulación de amplitud vectorial

---

### 3.1. Modulación de amplitud fuera de fase: caso escalar

Consideremos dos campos viajando en la dirección  $z$ , por simplicidad consideremos que ambos tienen fase y amplitud constante en el espacio y además están polarizados linealmente en el mismo plano (8):

$$E_1(x, y) = A_1 \cos(kz - \omega t + \alpha_1) \quad (3.1)$$

$$E_2(x, y) = A_2 \cos(kz - \omega t + \alpha_2), \quad (3.2)$$

La superposición de estos dos campos da un nuevo campo el cual tiene la forma:

$$E_1(x, y) + E_2(x, y) = E(x, y) = A \cos(kz - \omega t + \phi), \quad (3.3)$$

donde  $A$  es la amplitud resultante y  $\phi$  la fase resultante de la superposición, respectivamente. Utilizando las leyes trigonométricas de senos y cosenos y un poco de álgebra fundamental es posible obtener las expresiones para estas cantidades, están dadas por las siguientes ecuaciones:

$$A^2 = A_1^2 + A_2^2 + 2A_1A_2 \cos(\alpha_2 - \alpha_1) \quad (3.4a)$$

$$\tan(\alpha) = \frac{A_1 \sin \alpha_1 + A_2 \sin \alpha_2}{A_1 \cos \alpha_1 + A_2 \cos \alpha_2}. \quad (3.4b)$$

Notemos que si  $\alpha_1 = 0$  y  $\alpha_2 = \pi/2$  obtenemos expresiones de particular interés para  $A$  y  $\alpha$ , tienen la forma:

$$A^2 = A_1^2 + A_2^2 \quad (3.5a)$$

$$\tan \phi = \frac{A_2}{A_1}. \quad (3.5b)$$

Se puede ver que en las ecuaciones (3.4) tanto  $A$  como  $\phi$  dependen de  $A_1, A_2, \alpha_1$  y  $\alpha_2$ , y considerando estas últimas fijas, por tanto se puede cambiar ya sea la amplitud o

### 3. MODULACIÓN DE AMPLITUD VECTORIAL

---

la fase resultante de la superposición de dos campos variando unicamente las amplitudes de los campos que lo conforman. A esta técnica se le conoce como modulación de amplitud en cuadratura (QAM, de sus siglas en ingles Quadrature Amplitude Modulation).

De manera general, las relaciones inversas se pueden obtener de forma relativamente sencilla a partir de las expresiones (3.4):

$$A_1 = \frac{A}{\sin \Delta\alpha} \sin(\Delta\alpha + \alpha_1 - \phi), \quad (3.6a)$$

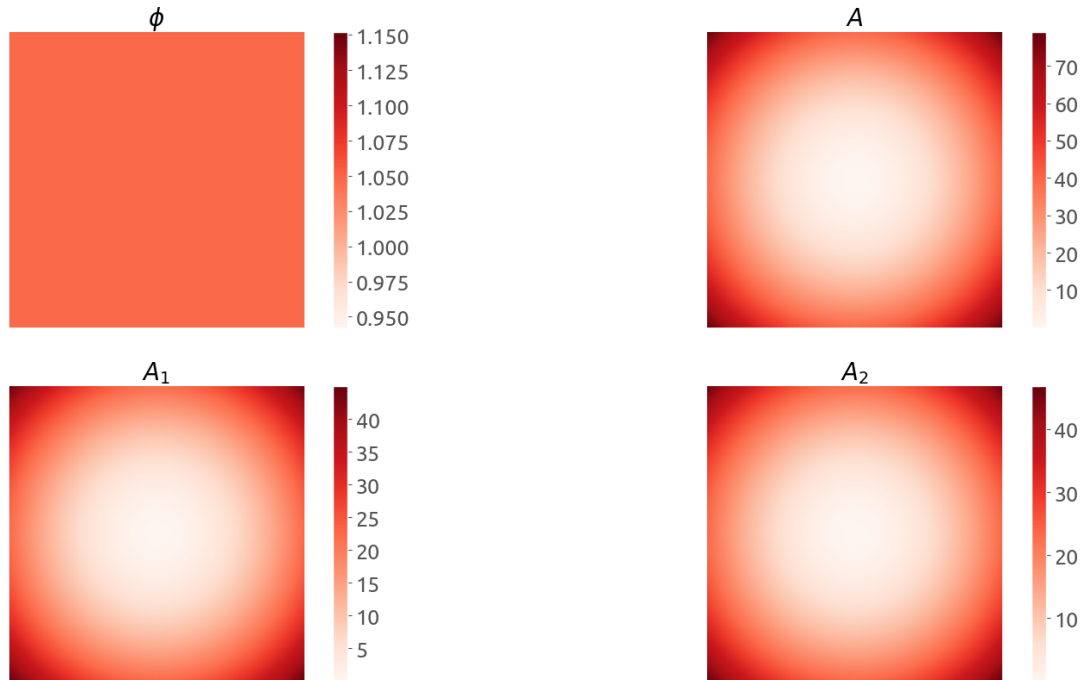
$$A_2 = \frac{A}{\sin \Delta\alpha} \sin(\phi - \alpha_1), \quad (3.6b)$$

donde  $\Delta\alpha = \alpha_2 - \alpha_1$ .

#### 3.1.1. Tipos de modulación

Al variar  $A_1$  y  $A_2$  en las ecuaciones (3.6) se obtienen tres casos: modulación de amplitud (AM), modulación de fase (PM) y modulación de amplitud y fase (PAM).

En la figura 3.1 se muestra el caso cuando en las ecuaciones (3.6)  $\phi$  es constante y  $A$  tiene la forma  $x^2 + y^2$ , este caso corresponde al de amplitud modulada, AM. Note que bajo estas consideraciones  $A_1$  y  $A_2$  resultan ser proporcionales a  $A$ , como bien se puede apreciar en la figura.

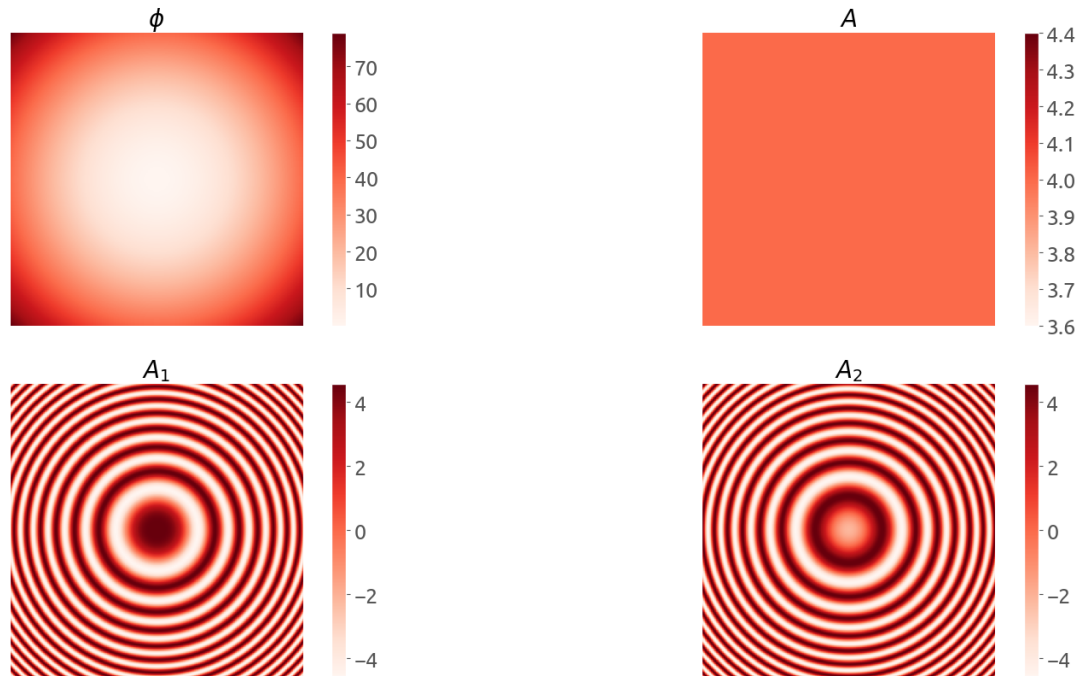


**Figura 3.1:** Modulación de amplitud en no-cuadratura en modulación de solo amplitud.

En la figura 3.2 se muestra el caso cuando la amplitud  $A$  es constante y la fase  $\phi$  es variable que corresponde al caso de modulación de fase, PM. Notemos que en este caso al ser  $A$  constante los puntos  $(A_1, A_2)$  se moverán al rededor de una elipse.

### 3. MODULACIÓN DE AMPLITUD VECTORIAL

---



**Figura 3.2:** Modulación de amplitud no-cuadratura en modulación de solo con fase.

Y finalmente se muestra el caso donde  $A$  y  $\phi$  tienen la forma  $x^2 + y^2$ , se tiene el caso de modulación de amplitud y fase, PAM.

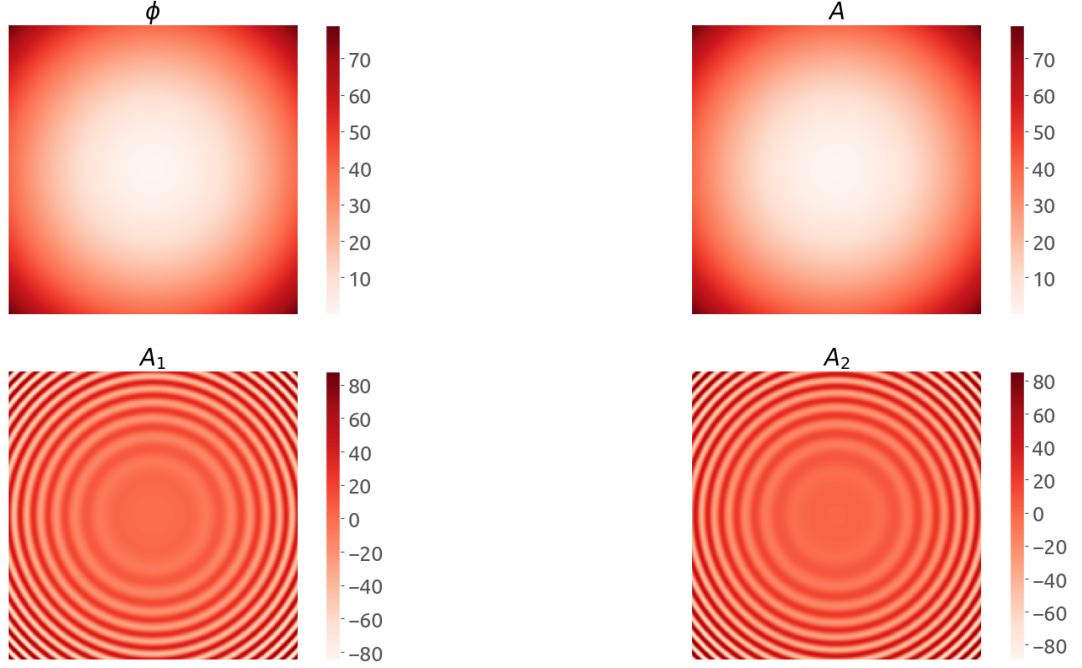


Figura 3.3: Modulación de amplitud en no-cuadratura en modulación de fase y amplitud

### 3.2. Modulación de amplitud vectorial fuera de fase

Podemos dar un tratamiento más general que el visto en la sección (3.1), si consideremos ahora dos campos con fase constante viajando en la dirección  $z$ , con la forma:

$$\vec{E}_1(\vec{r}, t) = \hat{i}A_{1x}(x, y) \cos(kz - \omega t + \alpha_{1x}) + \hat{j}A_{1y}(x, y) \cos(kz - \omega t + \alpha_{1y}) \quad (3.7)$$

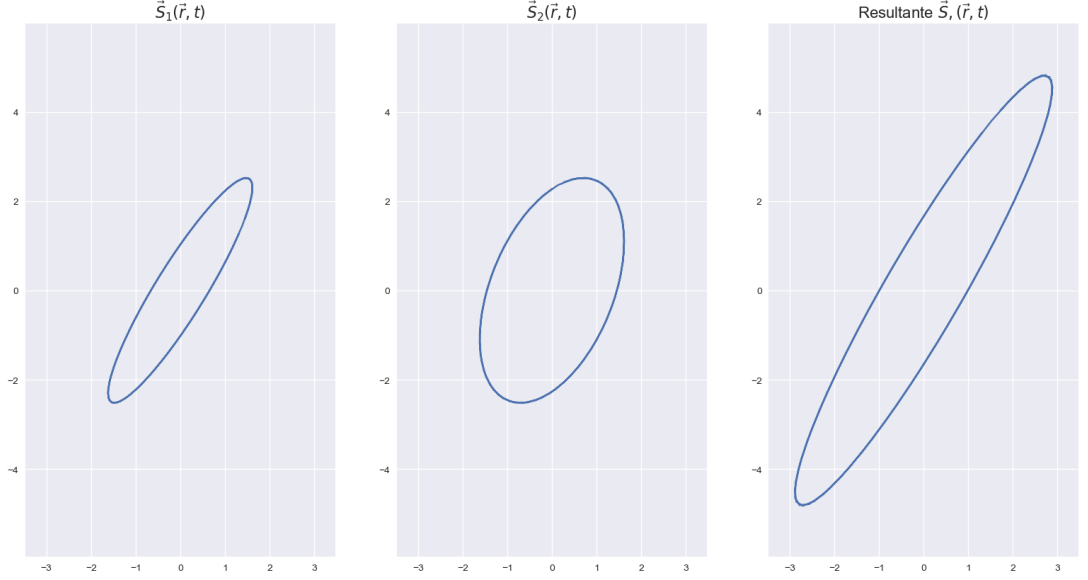
$$\vec{E}_2(\vec{r}, t) = \hat{i}A_{2x}(x, y) \cos(kz - \omega t + \alpha_{2x}) + \hat{j}A_{2y}(x, y) \cos(kz - \omega t + \alpha_{2y}). \quad (3.8)$$

Las cuales describen estados de polarización elípticos. Notemos además que sus estado de polarización no se modifica al variar la amplitud. Si superponemos ahora estos dos campos obtenemos un tercer campo que esta dado por la suma vectorial de las ecuaciones (3.7) y (3.8), en forma vectorial tenemos que:

$$\begin{aligned} \vec{E}(\vec{r}, t) &= \vec{E}_1(\vec{r}, t) + \vec{E}_2(\vec{r}, t) = \\ & \hat{i}[A_{1x}(x, y) \cos(kz - \omega t + \alpha_{1x}) + A_{2x}(x, y) \cos(kz - \omega t + \alpha_{2x})] + \\ & \hat{j}[A_{1y}(x, y) \cos(kz - \omega t + \alpha_{1y}) + A_{2y}(x, y) \cos(kz - \omega t + \alpha_{2y})] = \\ & \hat{i}A_x(x, y) \cos(kz - \omega t + \alpha_x) + \hat{j}A_y(x, y) \cos(kz - \omega t + \alpha_y), \end{aligned} \quad (3.9)$$

la cual describe un nuevo estado de polarización elíptico.

### 3. MODULACIÓN DE AMPLITUD VECTORIAL



**Figura 3.4:** Dos estados de polarización elípticos. El primer estado  $\vec{E}_1$  tiene como valores  $\alpha_{1x} = \pi/5$ ,  $A_{1x} = 3.0$ ,  $\alpha_{1y} = \pi/5$  y  $A_{1y} = 3$ . El segundo estado  $\vec{E}_2$  tiene como valores  $\alpha_{2x} = \pi/2$ ,  $A_{2x} = 3$ ,  $\alpha_{2y} = \pi/7$  y  $A_{2y} = 3$  y el resultado de su superposición  $\vec{E}$ .

En la figura (3.4) se puede ver un ejemplo de la superposición de estos dos campos dados por las ecuaciones (3.7) y (3.8) y el campo resultante (3.9).

Notemos que es posible escribir a  $A_{1x}$ ,  $A_{1y}$ ,  $A_{2x}$  y  $A_{2y}$  como:

$$A_{1x}(x, y) = A_1 \cos \sigma_1 \quad (3.10a)$$

$$A_{1y}(x, y) = A_1 \sin \sigma_1 \quad (3.10b)$$

$$A_{2x}(x, y) = A_2 \cos \sigma_2 \quad (3.10c)$$

$$A_{2y}(x, y) = A_2 \sin \sigma_2 \quad (3.10d)$$

$$A_x(x, y) = A \cos \sigma \quad (3.10e)$$

$$A_y(x, y) = A \sin \sigma. \quad (3.10f)$$

Así, reescribimos a  $\vec{E}(\vec{r}, t)$  como:

$$\vec{E}(\vec{r}, t) = \hat{i}A \cos \sigma \cos(kz - \omega t + \alpha_x) + \hat{j}A \sin \sigma \cos(kz - \omega t + \alpha_y) = A (\hat{i} \cos \sigma \cos(kz - \omega t + \alpha_x) + \hat{j} \sin \sigma \cos(kz - \omega t + \alpha_y)). \quad (3.11)$$

Si separamos las componentes de la ecuación (3.11) podemos tratar estas expresiones de manera similar al visto en la sección 3.1. La componente  $\hat{i}$  y  $\hat{j}$  de la ecuación 3.11 son respectivamente:

$$A \cos \sigma \cos(kz - \omega t + \alpha_x) = A_1 \cos \sigma_1 \cos(kz - \omega t + \alpha_{1x}) + A_2 \cos \sigma_2 \cos(kz - \omega t + \alpha_{2x}) \quad (3.12)$$

$$A \sin \sigma \cos(kz - \omega t + \alpha_y) = A_1 \sin \sigma_1 \cos(kz - \omega t + \alpha_{1y}) + A_2 \sin \sigma_2 \cos(kz - \omega t + \alpha_{2y}). \quad (3.13)$$

Notemos que las expresiones (3.12) y (3.13) tienen la forma ya estudiada en la sección 3.1 dada por la ecuación (3.3). Por tanto podemos aplicar los resultados vistos en las expresiones (3.4):

$$(A \cos \sigma)^2 = (A_1 \cos \sigma_1)^2 + (A_2 \cos \sigma_2)^2 + 2A_1 \cos \sigma_1 A_2 \cos \sigma_2 \cos \Delta\alpha_x \quad (3.14)$$

$$(A \sin \sigma)^2 = (A_1 \sin \sigma_1)^2 + (A_2 \sin \sigma_2)^2 + 2A_1 \sin \sigma_1 A_2 \sin \sigma_2 \cos \Delta\alpha_y \quad (3.15)$$

$$\tan(\alpha_x) = \frac{A_1 \cos \sigma_1 \sin \alpha_{1x} + A_2 \cos \sigma_2 \sin \alpha_{2x}}{A_1 \cos \sigma_1 \cos \alpha_{1x} + A_2 \cos \sigma_2 \cos \alpha_{2x}} \quad (3.16)$$

$$\tan(\alpha_y) = \frac{A_1 \sin \sigma_1 \sin \alpha_{1y} + A_2 \sin \sigma_2 \sin \alpha_{2y}}{A_1 \sin \sigma_1 \cos \alpha_{1y} + A_2 \sin \sigma_2 \cos \alpha_{2y}}, \quad (3.17)$$

donde  $\Delta\alpha_x = \alpha_{2x} - \alpha_{1x}$  y  $\Delta\alpha_y = \alpha_{2y} - \alpha_{1y}$  son las diferencias de fase.

Para estudiar los estados de polarización requerimos los parámetros  $A$ ,  $\sigma$ ,  $\Delta\alpha = \alpha_y - \alpha_x$ . Para obtener  $A$  se suman las expresiones (3.14) y (3.15) y realizando un poco de álgebra se obtiene la expresión:

$$A^2 = A_1^2 + A_2^2 + 2A_1 A_2 (\cos \sigma_1 \cos \sigma_2 \cos \Delta\alpha_x + \sin \sigma_1 \sin \sigma_2 \sin \Delta\alpha_y). \quad (3.18)$$

Para obtener la expresión para  $\sigma$  simplemente realizamos el cociente de las expresiones (3.15) y (3.14), así obtenemos de manera directa la ecuación:

$$\tan(\sigma)^2 = \frac{(A_1 \sin \sigma_1)^2 + (A_2 \sin \sigma_2)^2 + 2A_1 \sin \sigma_1 A_2 \sin \sigma_2 \cos \Delta\alpha_y}{(A_1 \cos \sigma_1)^2 + (A_2 \cos \sigma_2)^2 + 2A_1 \cos \sigma_1 A_2 \cos \sigma_2 \cos \Delta\alpha_x}. \quad (3.19)$$

Y finalmente para obtener la expresión para  $\Delta\alpha$  aplicamos arctan a las expresiones (3.16) y (3.17):

### 3. MODULACIÓN DE AMPLITUD VECTORIAL

---

$$\alpha_x = \arctan \left[ \frac{A_1 \cos \sigma_1 \sin \alpha_{1x} + A_2 \cos \sigma_2 \sin \alpha_{2x}}{A_1 \cos \sigma_1 \cos \alpha_{1x} + A_2 \cos \sigma_2 \cos \alpha_{2x}} \right] \quad (3.20)$$

$$\alpha_y = \arctan \left[ \frac{A_1 \sin \sigma_1 \sin \alpha_{1y} + A_2 \sin \sigma_2 \sin \alpha_{2y}}{A_1 \sin \sigma_1 \cos \alpha_{1y} + A_2 \sin \sigma_2 \cos \alpha_{2y}} \right]. \quad (3.21)$$

En esta forma utilizamos la identidad trigonométrica:

$$\arctan x \pm \arctan y = \arctan \left[ \frac{x \pm y}{1 \mp xy} \right], \quad (3.22)$$

donde  $x$  y  $y$  están claramente dados por las ecuaciones (3.20) y (3.21).

De manera análoga, las relaciones inversas pueden obtenerse a partir de (3.6) aplicándolas a (3.14) y (3.15):

$$A_1 = \frac{A \cos \sigma}{\cos \sigma_1 \sin \Delta \alpha_x} \sin(\alpha_{2x} - \alpha_x) \quad (3.23)$$

$$A_2 = \frac{A \cos \sigma}{\cos \sigma_2 \sin \Delta \alpha_x} \sin(\alpha_x - \alpha_{1x}) \quad (3.24)$$

$$A_1 = \frac{A \cos \sigma}{\cos \sigma_1 \sin \Delta \alpha_y} \sin(\alpha_{2y} - \alpha_y) \quad (3.25)$$

$$A_2 = \frac{A \cos \sigma}{\cos \sigma_2 \sin \Delta \alpha_y} \sin(\alpha_y - \alpha_{1y}). \quad (3.26)$$

Notemos que tenemos dos ecuaciones para cada valor de  $A_1$  y  $A_2$  respectivamente dadas por las expresiones (3.23) - (3.26). Veamos que los valores para las relaciones inversas (3.23) y (3.25) parecieran ser independientes, la primera en principio no dependería del  $\alpha_y$  y la segunda, de la misma manera, no dependería del parámetro  $\alpha_x$ , sin embargo si existe una relación directa entre ellas, al dar un valor de  $\alpha_x$  estamos confinados a obtener uno de  $\alpha_y$ . Se puede ver esto si sustituimos la ecuación (3.23) en (3.13), para realizar los cálculos correspondientes, consideremos la ecuación (3.13). Realizando esta sustitución obtenemos:

$$\tan \sigma_1 \frac{\sin(\alpha_{2x} - \alpha_x)}{\sin \Delta \alpha_x} \cos \alpha_{1y} + \tan \sigma_2 \frac{\sin(\alpha_x - \alpha_{1x})}{\sin \Delta \alpha_x} \cos \alpha_{2y} = \tan \sigma \cos \alpha_y \quad (3.27a)$$

$$\tan \sigma_1 \frac{\sin(\alpha_{2x} - \alpha_x)}{\sin \Delta \alpha_x} \sin \alpha_{1y} + \tan \sigma_2 \frac{\sin(\alpha_x - \alpha_{1x})}{\sin \Delta \alpha_x} \sin \alpha_{2y} = \tan \sigma \sin \alpha_y. \quad (3.27b)$$

Ahora podemos tratar esta ecuación de manera análoga que en (3.4). Así obtenemos dos ecuaciones, una para  $\alpha = \alpha_y - \alpha_x$  que notemos esta en función de  $\alpha_x$  y otra para  $\sigma$  que igualmente se encuentra en función de  $\alpha_x$ , tiene la forma:

$$\tan(\alpha) = -\frac{\gamma_1 - \gamma_2}{\gamma_3 - \gamma_4}, \quad (3.28)$$

donde

$$\begin{aligned}\gamma_1 &= \tan \sigma_1 [\cos(\alpha_{2x} + \alpha_{1x} - 2\alpha_x) - \cos(\alpha_{1y} - \alpha_{2x})] \\ \gamma_2 &= \tan \sigma_2 [\cos(\alpha_{1x} + \alpha_{2y} - 2\alpha_x) - \cos(\alpha_{2y} - \alpha_{1x})] \\ \gamma_3 &= \tan \sigma_1 [\sin(\alpha_{2x} + \alpha_{1x} - 2\alpha_x) - \sin(\alpha_{1y} - \alpha_{2x})] \\ \gamma_4 &= \tan \sigma_2 [\sin(\alpha_{1x} + \alpha_{2y} - 2\alpha_x) - \sin(\alpha_{2y} - \alpha_{1x})],\end{aligned}$$

y para  $\sigma$

$$\tan(\sigma) = (\psi_1)^2 + (\psi_2)^2 + 2\psi_1\psi_2 \cos(\alpha_{2y} - \alpha_{1y}), \quad (3.29)$$

donde:

$$\begin{aligned}\psi_1 &= \tan \sigma_1 \frac{\sin(\alpha_{2x} - \alpha_x)}{\sin \Delta\alpha_x} \\ \psi_2 &= \tan \sigma_2 \frac{\sin(\alpha_x - \alpha_{1x})}{\sin \Delta\alpha_x}.\end{aligned}$$

Así podemos ver la dependencia que hay entre estas dos cantidades para poder estudiarla a detalle.

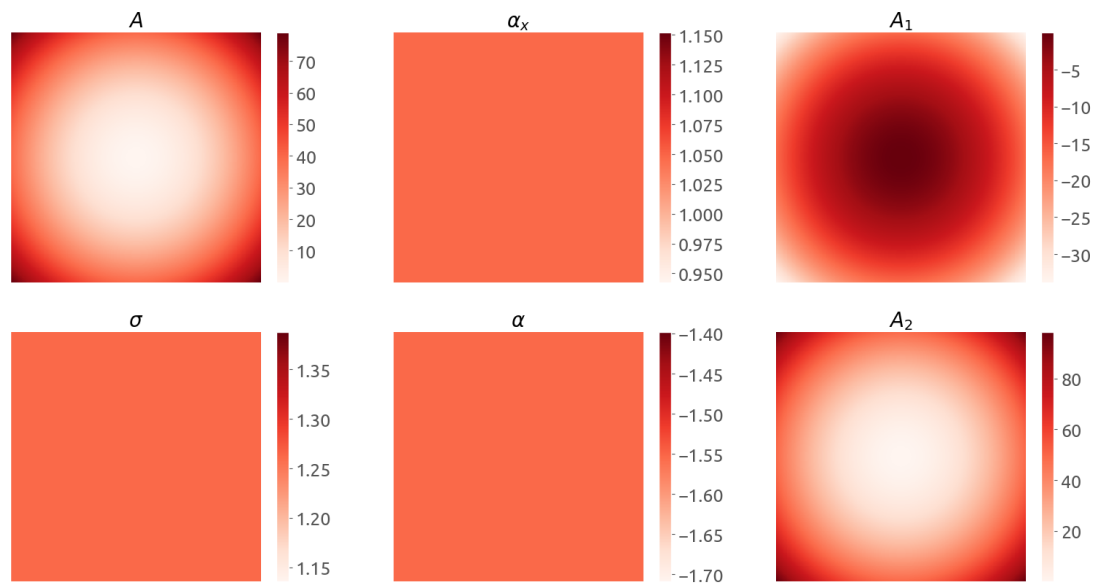
### 3.2.1. Modulación de amplitud vectorial: tipos de modulación

Como los casos vistos en la sección 3.1.1, al variar las amplitudes (3.23) y (3.24) o bien usando las expresiones (3.25) y (3.26) también obtenemos diferentes casos de modulación. Con algunas restricciones dadas por las ecuaciones (3.29) y (3.28), notemos que ambas ecuaciones están en función de  $\alpha_x$ , por esta razón tenemos la polarización directamente relacionada a nuestros resultados para la modulación en los casos de modulación de solo fase y modulación de amplitud y fase, denotaremos a estos últimos como modulación de polarización y fase (PoPM), y modulación de polarización, fase y amplitud (PoPAM).

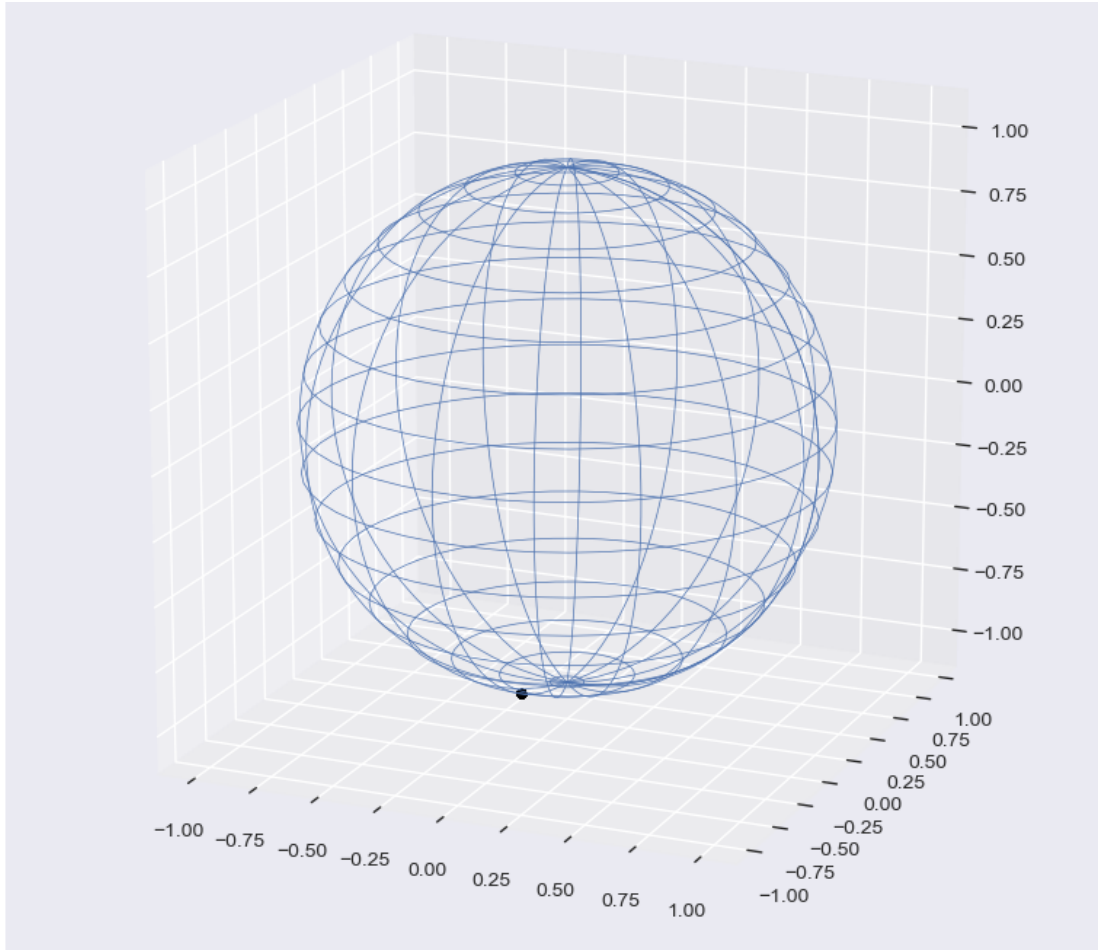
Primero consideremos el caso de modulación de amplitud (AM). Como en la sección 3.1.1, consideremos a  $\alpha_x$  como constante y variaremos  $A$ , se muestra a continuación los resultados:

### 3. MODULACIÓN DE AMPLITUD VECTORIAL

---



**Figura 3.5:** Modulación de solo amplitud vectorial (AM)



**Figura 3.6:** Modulación de amplitud vectorial mapeada a la esfera de Poincaré

Ahora se considera a  $A$  como constante y se varia la fase  $\alpha_x$  la cual se encuentra relaciona con  $\alpha$  y  $\sigma$ , obtenemos PoPM:

### 3. MODULACIÓN DE AMPLITUD VECTORIAL

---

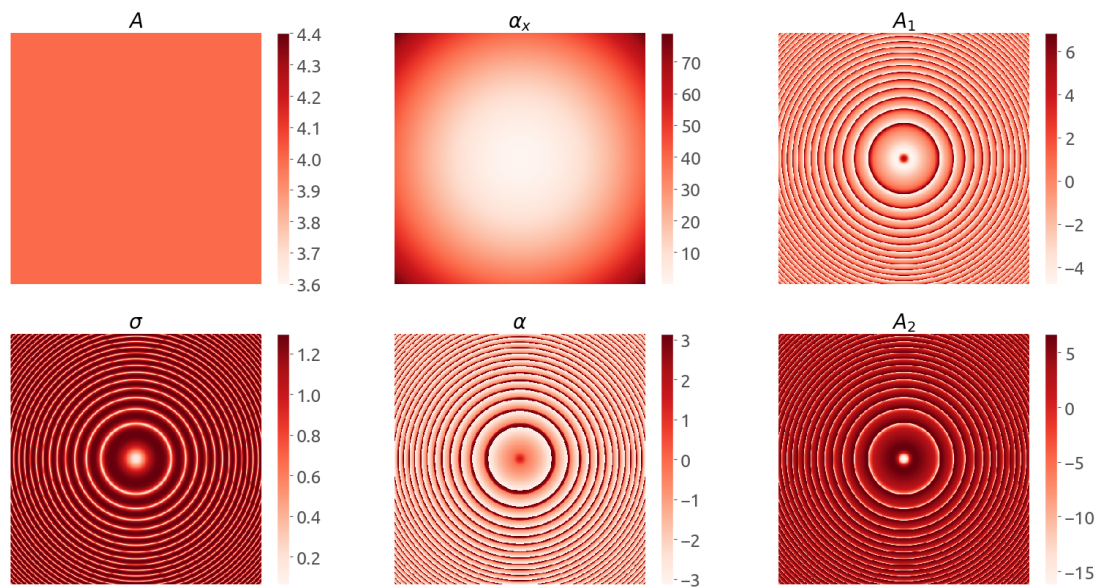


Figura 3.7: Modulación de polarización y fase vectorial (PoPM)

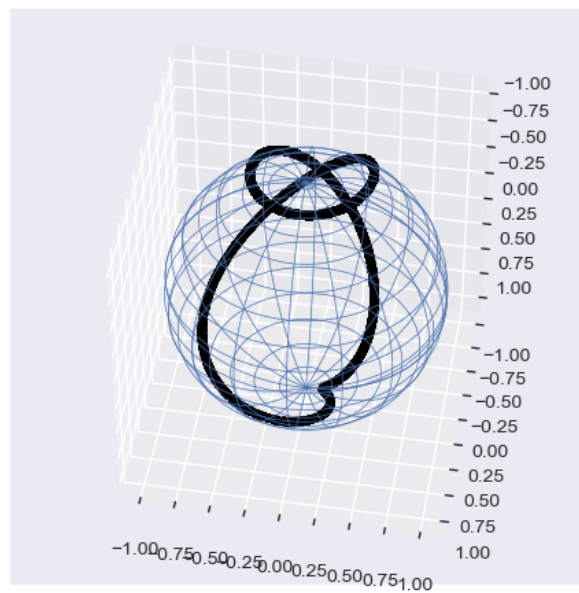
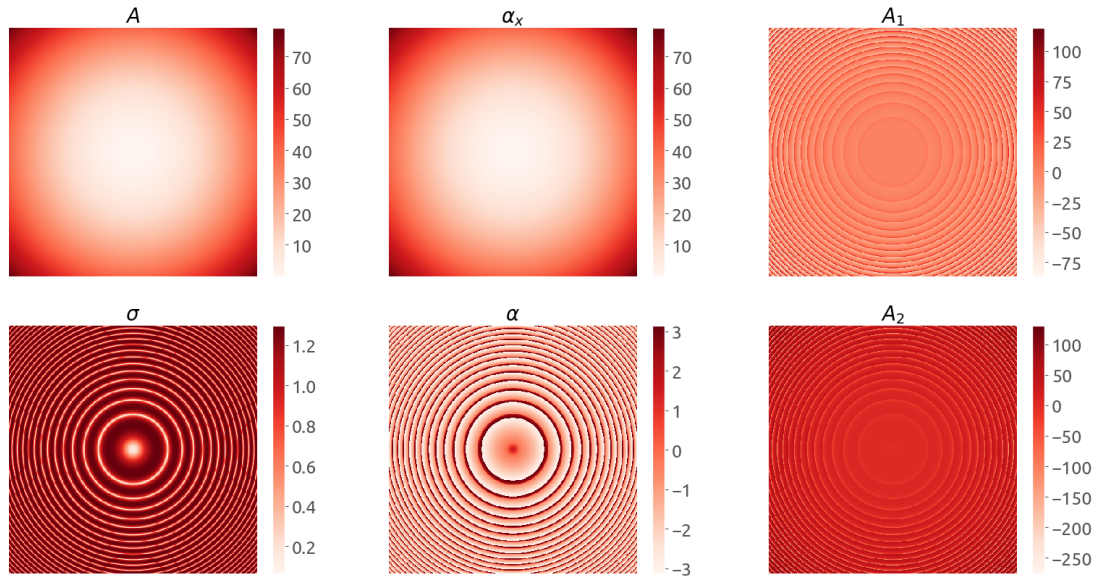


Figura 3.8: Modulación de polarización y fase mapeada a la esfera de Poincaré

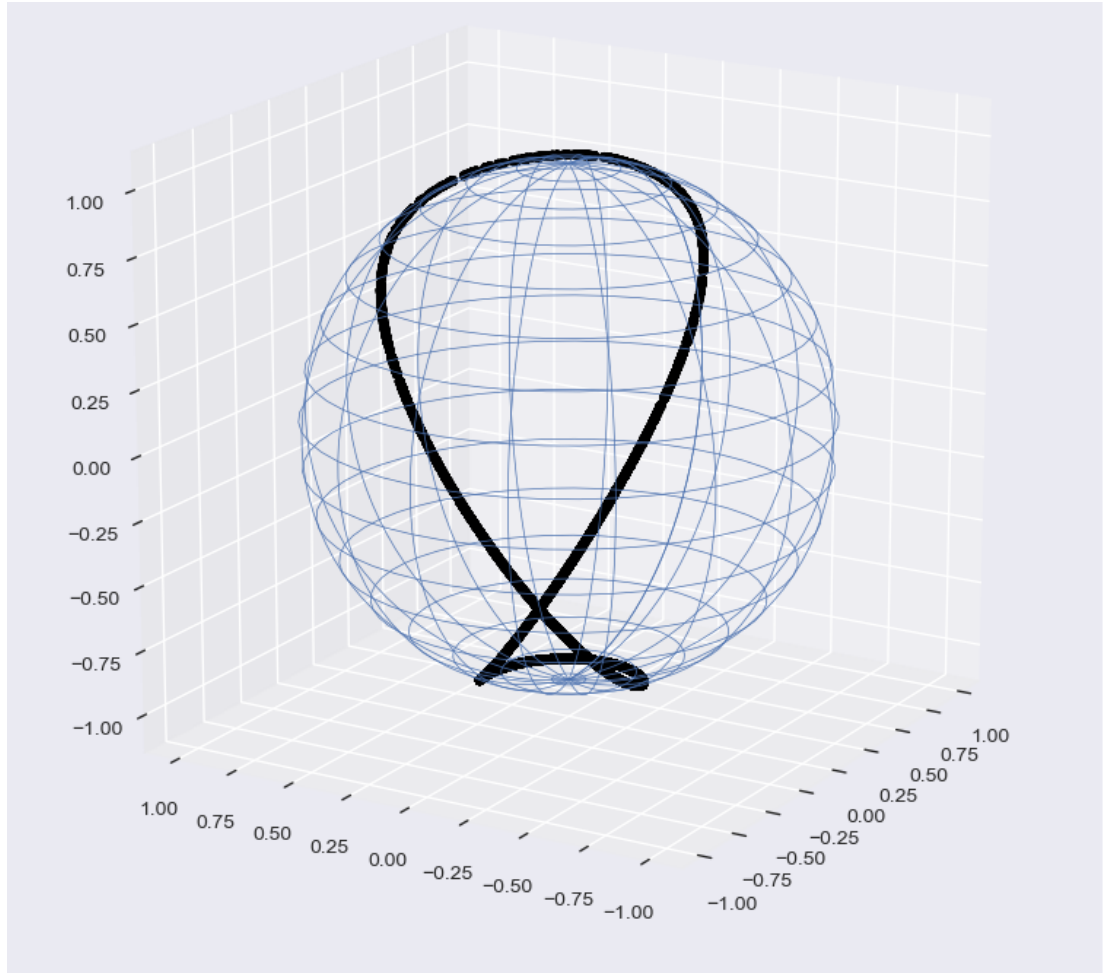
Finalmente variamos ambos parámetros,  $\alpha_x$ ,  $A$  y consecuentemente  $\alpha$  y  $\sigma$ , en este caso obtenemos PoPAM :



**Figura 3.9:** Modulación de polarización, fase y amplitud vectorial (POPAM)

### 3. MODULACIÓN DE AMPLITUD VECTORIAL

---



**Figura 3.10:** Modulación de polarización, fase y amplitud vectorial mapeada a la esfera de Poincaré

---

## Capítulo 4

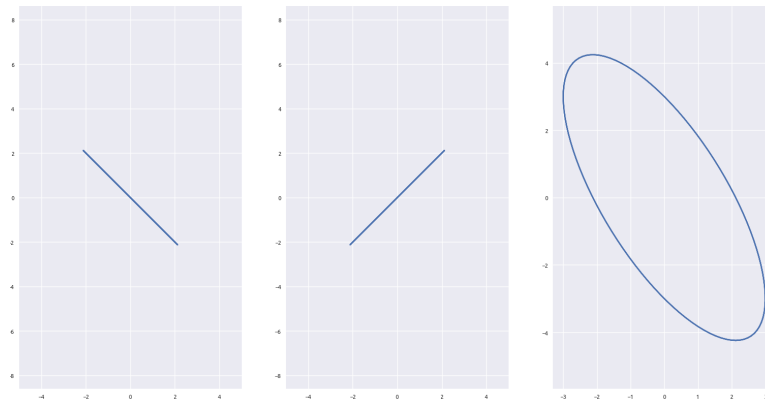
# Simulación de estados de polarización

---

Se realizaron simulaciones numéricas utilizando como lenguaje *python* (9) (10). Los códigos de estas simulaciones se encuentran en los apéndices.

### 4.1. Simulación de polarización lineal

Se presenta la polarización de dos estados ortogonales lineales a  $-45^\circ$  y  $45^\circ$  respectivamente, el cual se simuló utilizando el método descrito en la sección 3.2, los parámetros utilizados para la simulación son los siguientes:  $A_1 = 3$ ,  $A_2 = 3$ ,  $\Delta\alpha_x = \pi/3$ ,  $\alpha_{1x} = 0$ ,  $\alpha_{2x} = \alpha_{1x} + \Delta\alpha_x$ ,  $\alpha_1 = \pi$ ,  $\alpha_2 = 0$ ,  $\alpha_{1y} = \alpha_{1x} + \alpha_1$ ,  $\alpha_{2y} = \alpha_{1x} + \Delta\alpha_x + \alpha_2$ ,  $\sigma_1 = \pi/4$  y  $\sigma_2 = \pi/2 - \sigma_1$



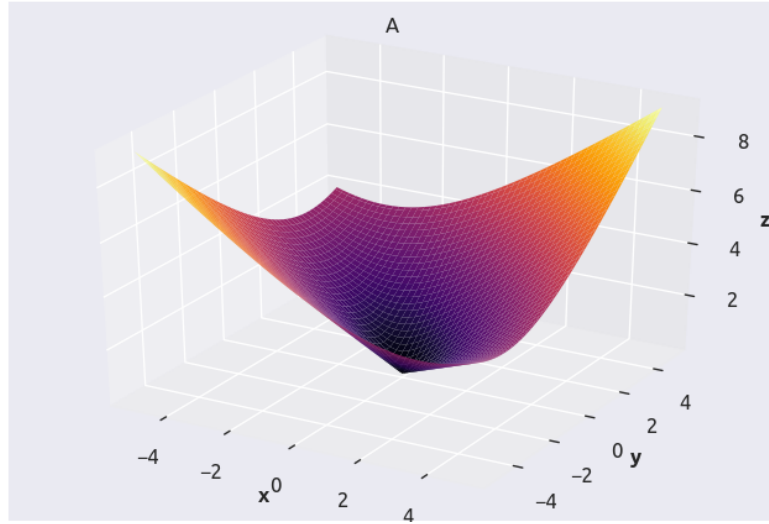
**Figura 4.1:** Superposición de dos estados ortogonales con polarización lineal

Utilizando estos mismos valores, ahora variamos  $A_1$  y  $A_2$  en intervalo de  $(-5, 5)$  manteniendo a  $\Delta\alpha_x = \pi/3$ . Se tiene que los valores para  $A$ ,  $\sigma$ ,  $\alpha$ , y  $\alpha_x$  están en función de  $A_1$  y  $A_2$ , esto es  $A(A_1, A_2)$ ,  $\sigma(A_1, A_2)$ ,  $\alpha(A_1, A_2)$  y  $\alpha_x(A_1, A_2)$ . Presentamos los

#### 4. SIMULACIÓN DE ESTADOS DE POLARIZACIÓN

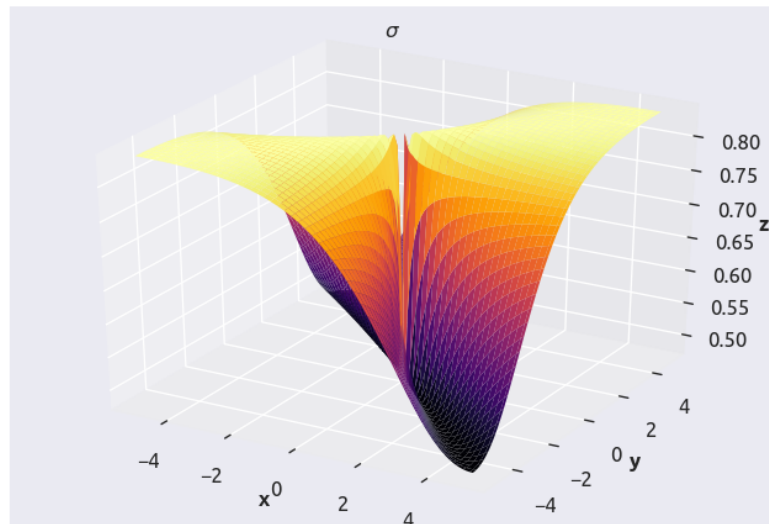
---

resultados de esta simulación. En la figura 4.2 se muestra a  $A$ , la amplitud resultante de la superposición de los estados lineales mostrados en la la figura 4.1



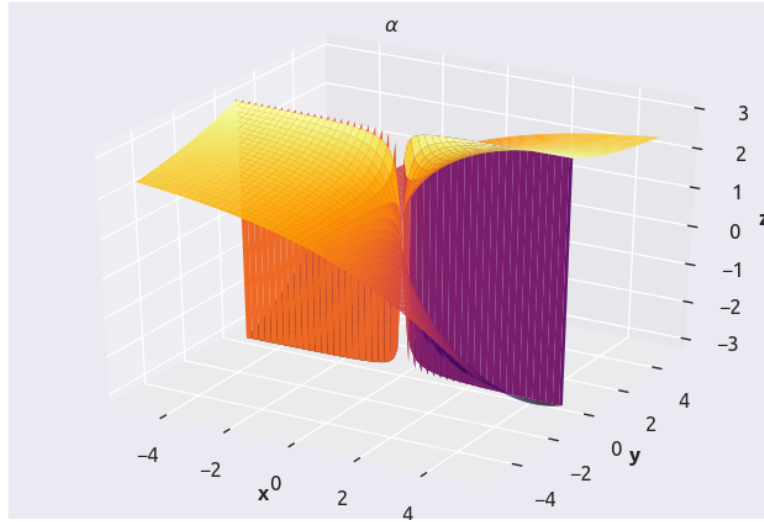
**Figura 4.2:** Amplitud resultante  $A$ , variando  $A_1$  y  $A_2$  en el intervalo de  $(-5, 5)$  manteniendo  $\Delta\alpha_x = \pi/3$ .

En la figura 4.3 se muestra a  $\sigma$  para este mismo caso:

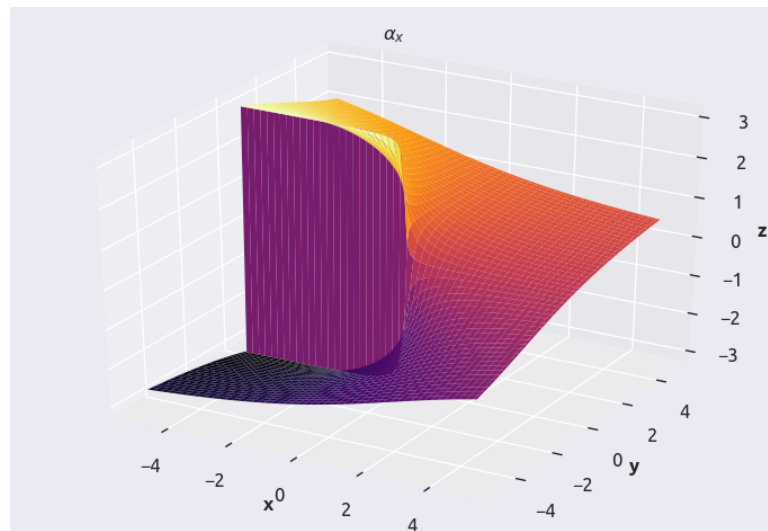


**Figura 4.3:**  $\sigma$ , variando  $A_1$  y  $A_2$  en el intervalo de  $(-5, 5)$  manteniendo  $\Delta\alpha_x = \pi/3$ .

Y finalmente mostramos a  $\alpha$  y  $\alpha_x$ :

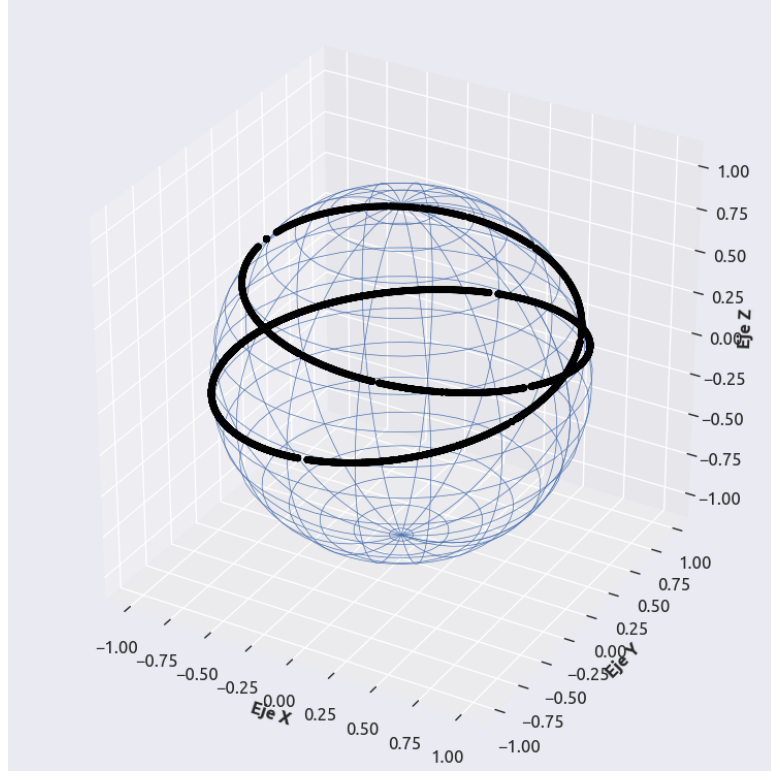


**Figura 4.4:**  $\alpha$ , variando  $A_1$  y  $A_2$  en el intervalo de  $(-5, 5)$  manteniendo  $\Delta\alpha_x = \pi/3$ .



**Figura 4.5:**  $\alpha_x$ , variando  $A_1$  y  $A_2$  en el intervalo de  $(-5, 5)$  manteniendo  $\Delta\alpha_x = \pi/3$ .

En la figura 4.6 mapeamos los resultados anteriores a la esfera de Poincaré



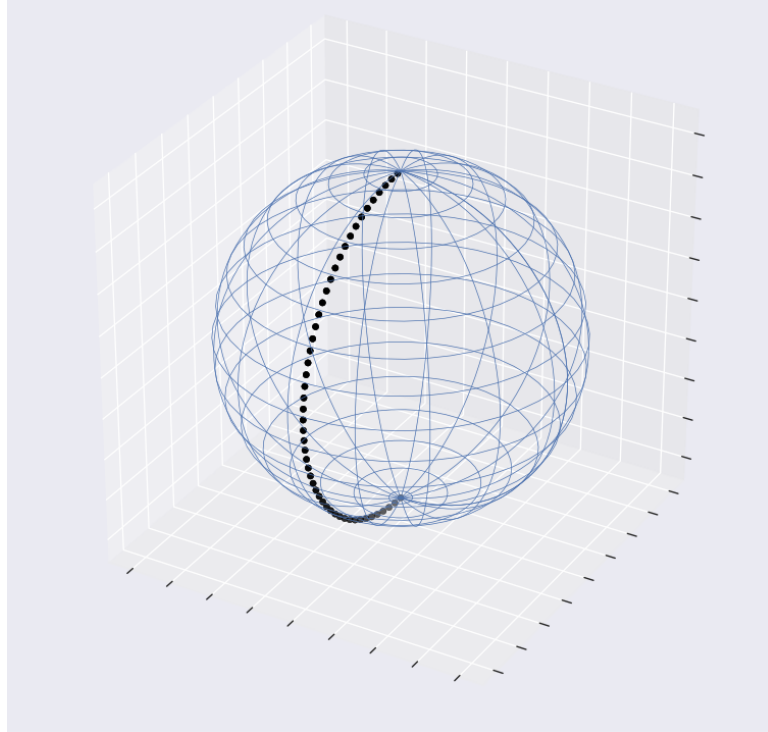
**Figura 4.6:** Esfera de Poincare variando  $A_1$  y  $A_2$  en el intervalo de  $(-5, 5)$  manteniendo  $\Delta\alpha_x = \pi/3$ .

Ahora se considera el caso en el que  $A_1$  y  $A_2$  con valores constantes, y variamos a  $\Delta\alpha_x$  en el rango de  $(-\pi, \pi)$ . Ahora tenemos a  $A$ ,  $\sigma$ ,  $\alpha$ , y  $\alpha_x$  en función de  $\Delta\alpha_x$ . En la figura 4.7 se muestra este caso:



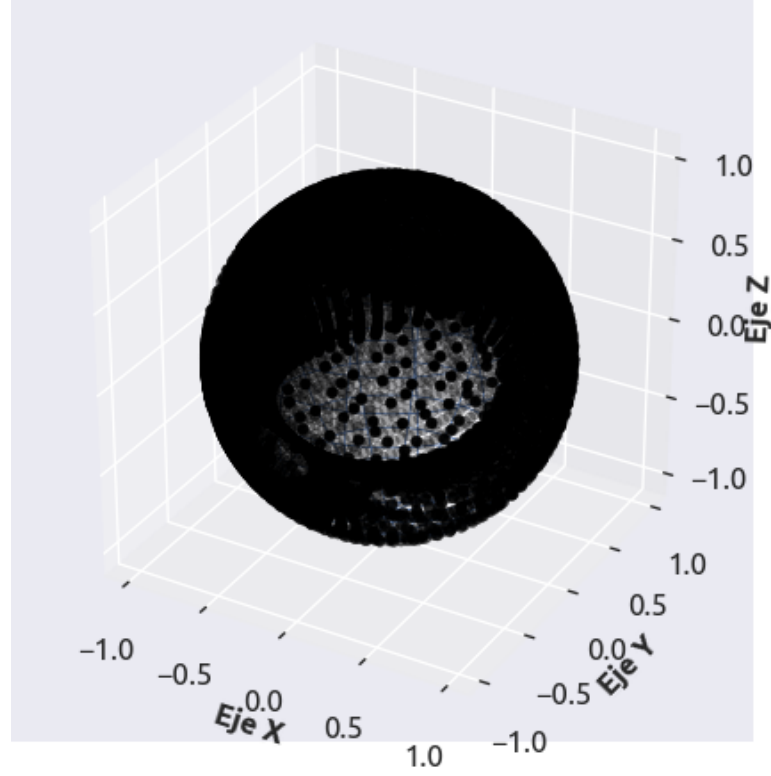
**Figura 4.7:**  $A$ ,  $\sigma$ ,  $\alpha$ , y  $\alpha_x$  cuando  $\Delta\alpha_x$  varía en el intervalo  $(-\pi, \pi)$  y con  $A_1 = A_2 = 3$ .

De igual manera mapeamos los valores a la esfera de Poincaré:



**Figura 4.8:** Esfera de Poincaré cuando  $\Delta\alpha_x$  varía en el intervalo  $(-\pi, \pi)$  y con  $A_1 = A_2 = 3$ .

Finalmente, se considera ahora que tanto  $A_1$  y  $A_2$  como  $\Delta\alpha_x$  son variables en los mismos intervalos utilizados anteriormente. Ahora al tener funciones de tres variables, únicamente mapeamos los valores a la esfera de Poincaré. En la figura 4.9 mostramos este caso:



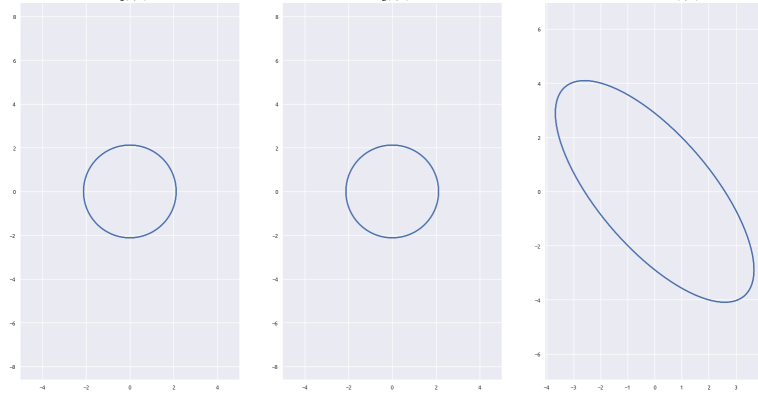
**Figura 4.9:** Esfera de Poincaré, con  $A_1, A_2$  variando en  $(-5, 5)$  y  $\Delta\alpha_x$  variando en  $(-\pi, \pi)$ .

## 4.2. Simulación de polarización circular

Aquí se muestra la superposición de dos estados de polarización circular a izquierdas y derechas, simulado con las consideraciones estudiadas en la sección 2.1.2. Los parámetros de simulación son los siguientes  $A_1 = 3, A_2 = 3, \Delta\alpha_x = \pi/3, \alpha_{1x} = \pi, \alpha_{2x} = \alpha_{1x} + \Delta\alpha_x, \alpha_1 = -\pi/2, \alpha_2 = 0, \alpha_{1y} = \alpha_{1x} + \alpha_1, \alpha_{2y} = \alpha_{1x} + \Delta\alpha_x + \alpha_2, \sigma_1 = \pi/4$  y  $\sigma_2 = \pi/2 - \sigma_1$

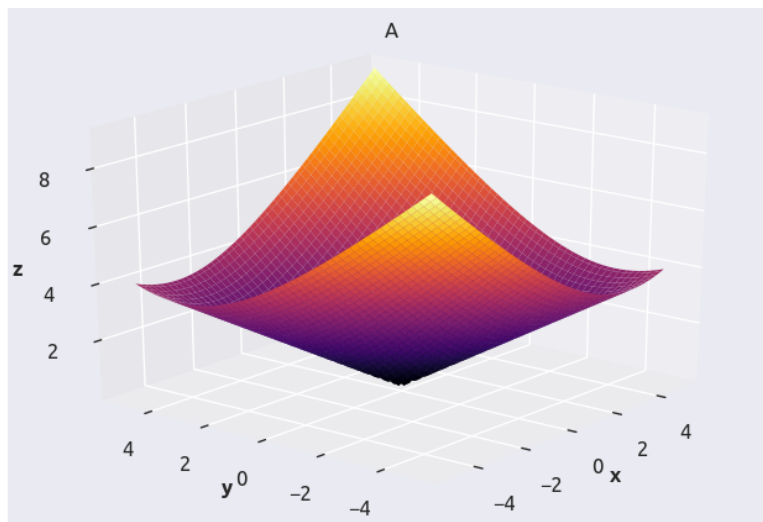
#### 4. SIMULACIÓN DE ESTADOS DE POLARIZACIÓN

---



**Figura 4.10:** Superposición de dos estados de polarización circular a derechas y a izquierdas respectivamente

Nuevamente como en la sección 4.1 se considera el caso cuando  $A_1$  y  $A_2$  varían en el mismo intervalo  $(-5, 5)$ . En las siguientes figuras, 4.11, 4.12, 4.13 y 4.14 se muestra a  $A$ ,  $\sigma$ ,  $\alpha$  y  $\alpha_x$  respectivamente, para este caso



**Figura 4.11:** Amplitud resultante  $A$ , variando  $A_1$  y  $A_2$  en el intervalo de  $(-5, 5)$ .

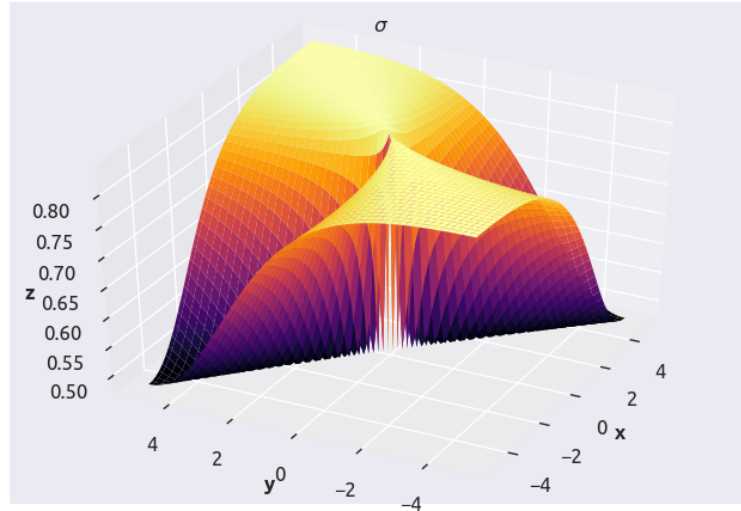


Figura 4.12:  $\sigma$  variando  $A_1$  y  $A_2$  en el intervalo de  $(-5, 5)$ .

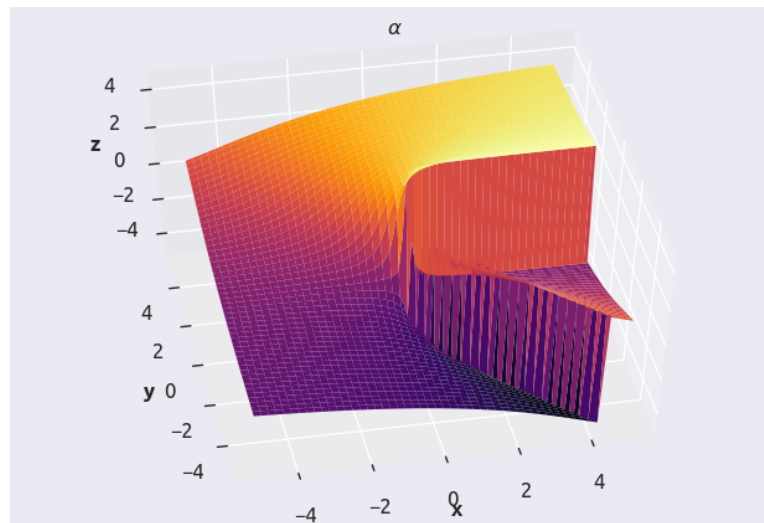
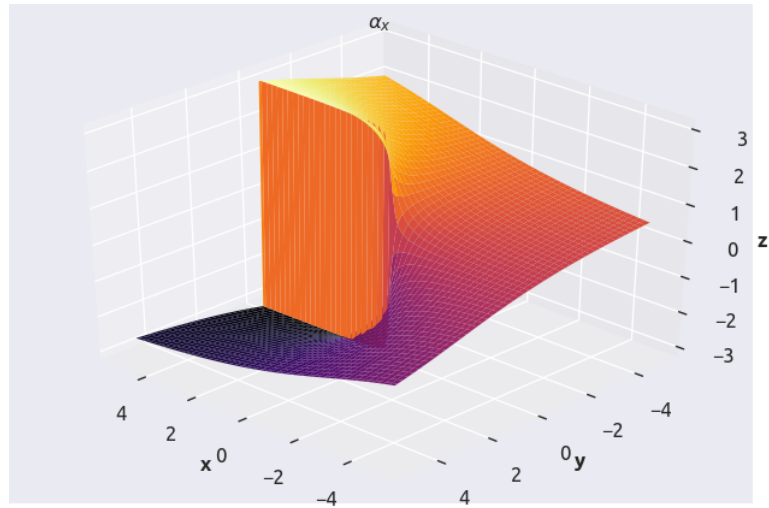


Figura 4.13:  $\alpha$  variando  $A_1$  y  $A_2$  en el intervalo de  $(-5, 5)$ .

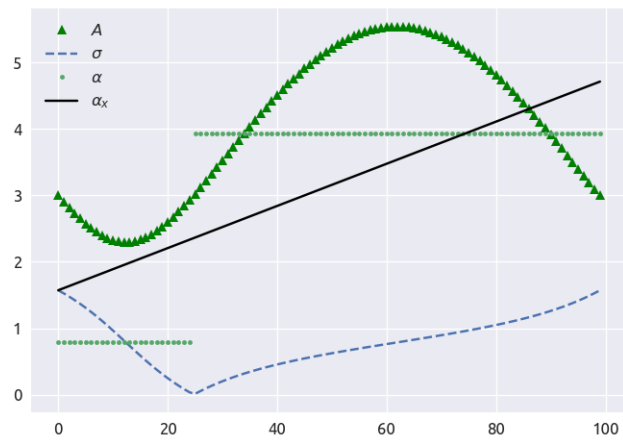
#### 4. SIMULACIÓN DE ESTADOS DE POLARIZACIÓN

---

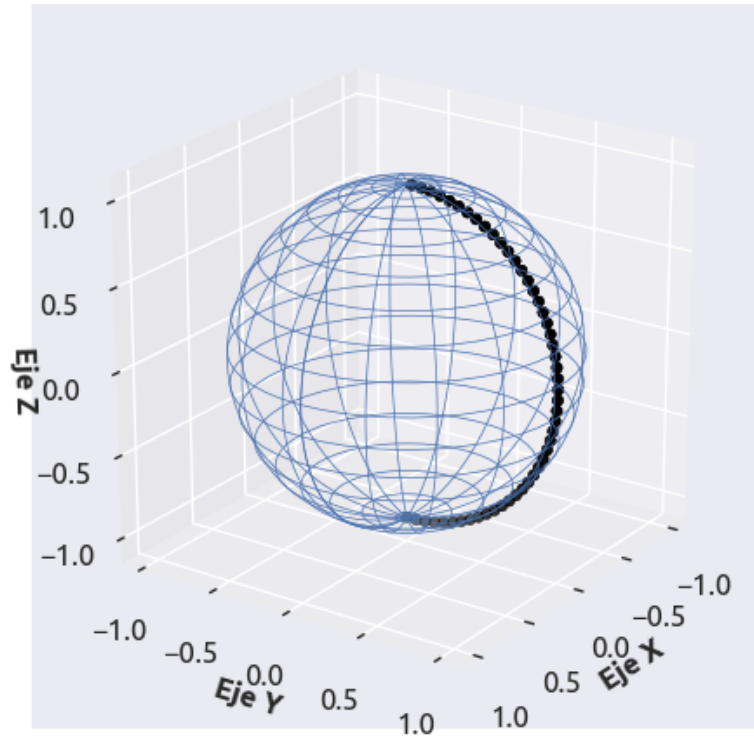


**Figura 4.14:**  $\alpha_x$  variando  $A_1$  y  $A_2$  en el intervalo de  $(-5, 5)$ .

Ahora se considera nuevamente el caso de mantener  $A_1$  y  $A_2$  como constantes y variamos nuevamente  $\Delta\alpha_x$  en el intervalo de  $(-\pi, \pi)$ . Se presentan los resultados:

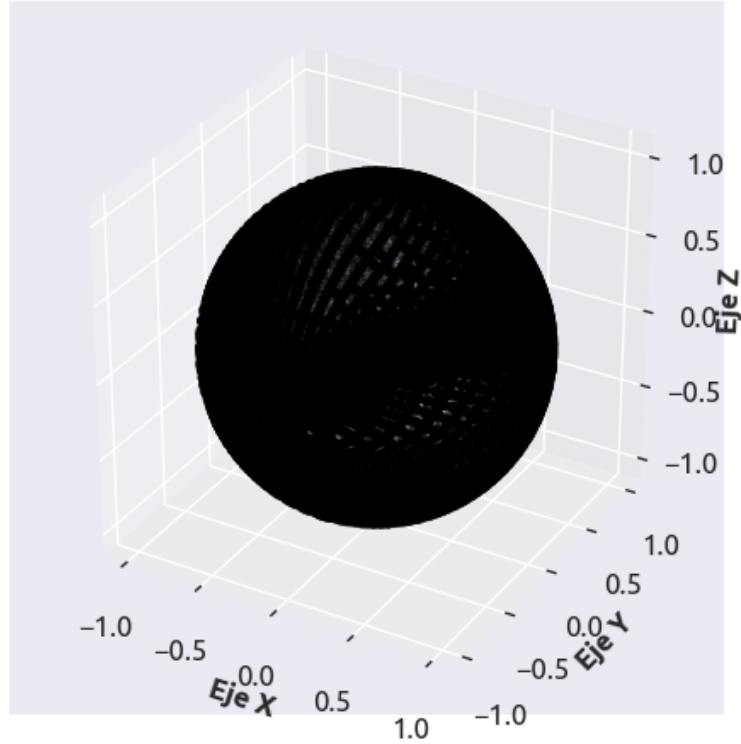


**Figura 4.15:**  $A$ ,  $\sigma$ ,  $\alpha$ , y  $\alpha_x$  cuando  $\Delta\alpha_x$  varia en  $(-\pi, \pi)$  y con  $A_1 = A_2 = 3$ .



**Figura 4.16:** Esfera de Poincaré cuando  $\Delta\alpha_x$  varía en  $(-\pi, \pi)$  y con  $A_1 = A_2 = 3$ .

Finalmente el caso con  $A_1$ ,  $A_2$  y  $\Delta\alpha_x$  variables:

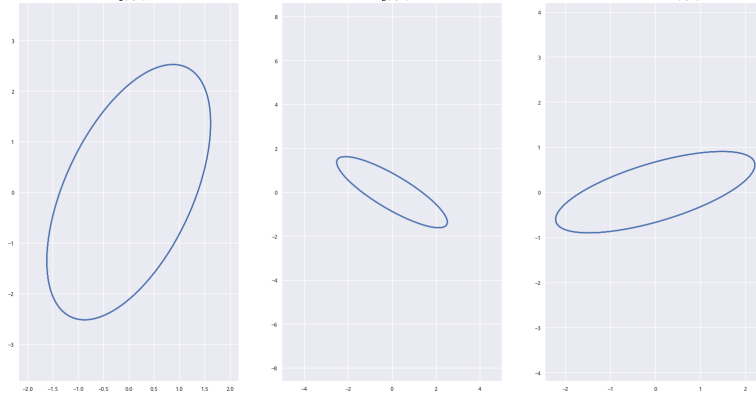


**Figura 4.17:** Esfera de Poincaré cuando  $\Delta\alpha_x$  varia en  $(-\pi, \pi)$  y con  $A_1$  y  $A_2$  varian en  $(-5, 5)$ .

### 4.3. Simulación de polarización elíptica

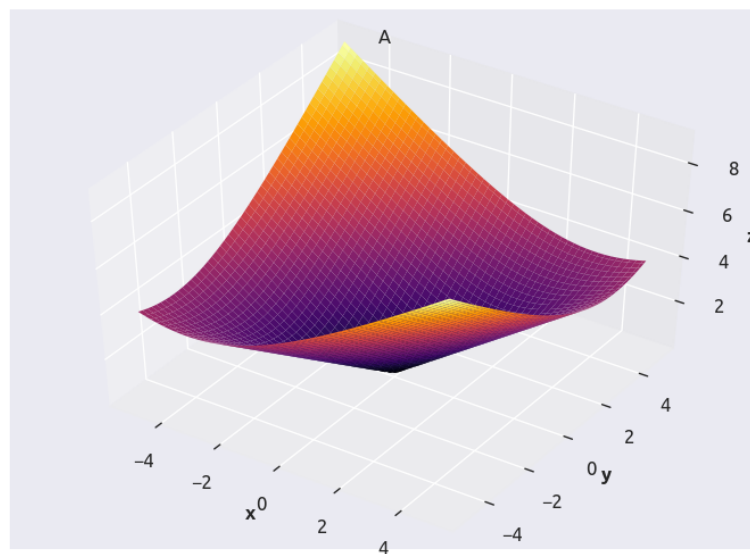
Se creo un estado de polarización elíptico a derechas conformado por dos estados de polarización ortogonales entre si, uno a izquierdas y otro a derechas respectivamente, variando la amplitud del segundo, los parámetros de simulación son:  $A_1 = 4$ ,  $A_2 = 2$ ,  $\Delta\alpha_x = \pi/3$ ,  $\alpha_{1x} = 1$ ,  $\alpha_{2x} = \alpha_{1x} + \Delta\alpha_x$ ,  $\alpha_1 = 1$ ,  $\alpha_2 = 1 + \pi - \pi/3$ ,  $\alpha_{1y} = \alpha_{1x} + \alpha_1$ ,  $\alpha_{2y} = \alpha_{1x} + \Delta\alpha_x + \alpha_2$ ,  $\sigma_1 = 1$  y  $\sigma_2 = \pi/2 - \sigma_1$

La elipticidad de las dos elipses es  $\epsilon_1 = 0.5$ ,  $\epsilon_2 = -0.5$  y el de la resultante es  $\epsilon = -0.16$ .

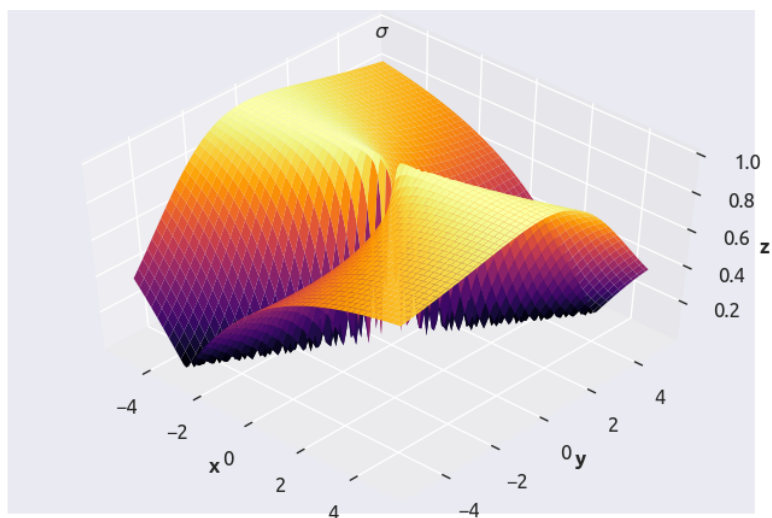


**Figura 4.18:** Superposición de dos estados elípticos.

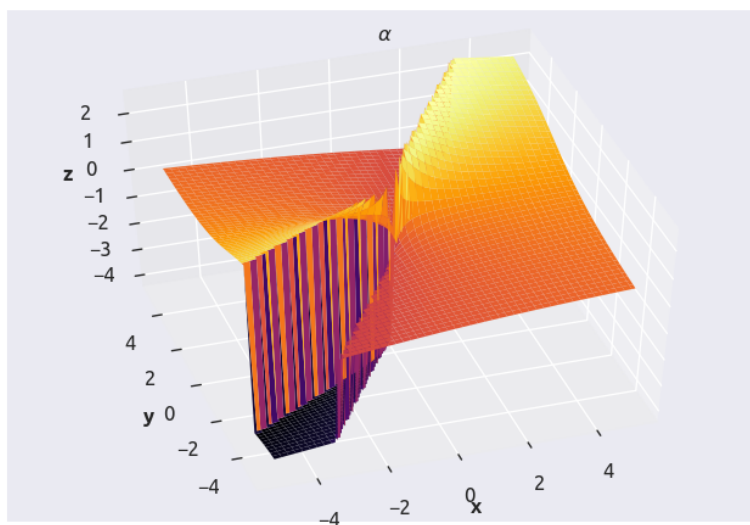
Se consideran los mismos casos que en las secciones 4.1 y 4.2. El primer caso con  $A_1$  y  $A_2$  variables:



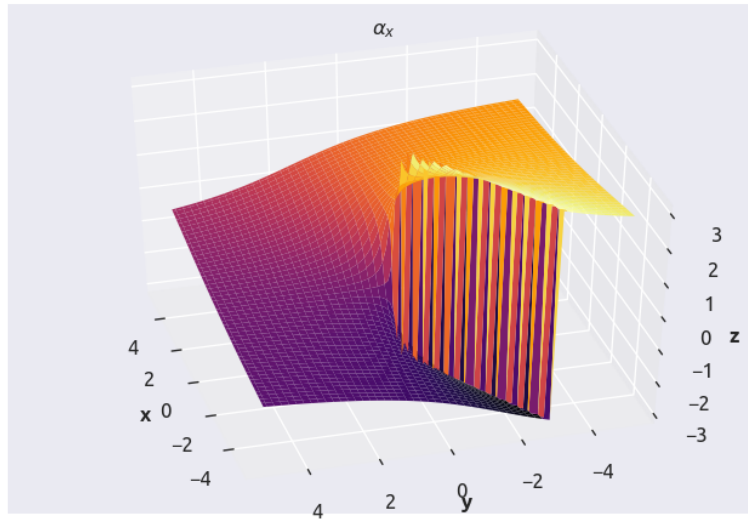
**Figura 4.19:**  $A$  con  $A_1$  y  $A_2$  variables en el intervalo de  $(-5, 5)$ .



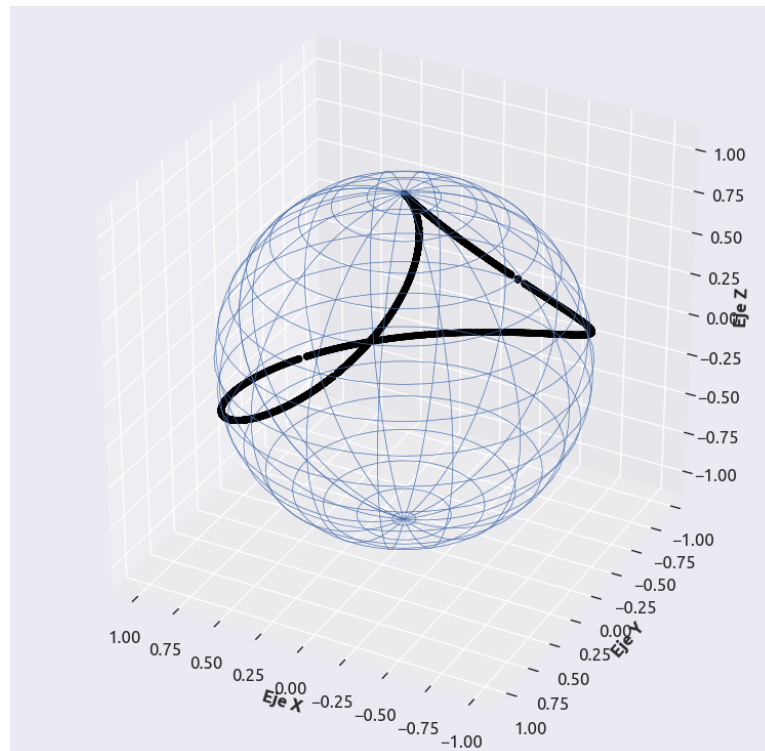
**Figura 4.20:**  $\sigma$  con  $A_1$  y  $A_2$  variables en el intervalo de  $(-5, 5)$ .



**Figura 4.21:**  $\alpha$  con  $A_1$  y  $A_2$  variables en el intervalo de  $(-5, 5)$ .



**Figura 4.22:**  $\alpha_x$  con  $A_1$  y  $A_2$  variables en el intervalo de  $(-5, 5)$ .

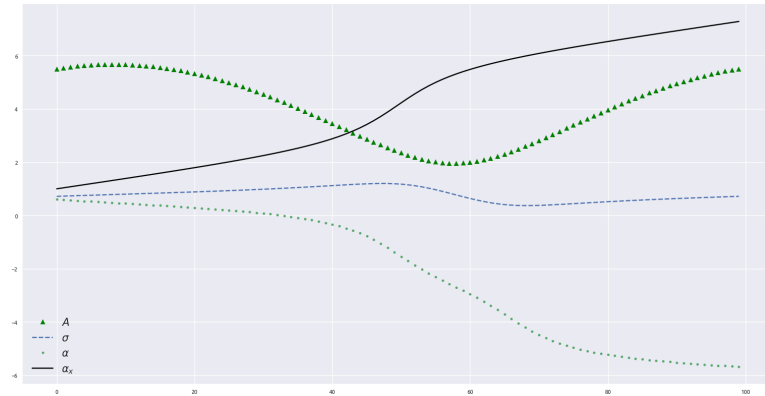


**Figura 4.23:** Esfera de Poincaré con  $A_1$  y  $A_2$  variables en el intervalo de  $(-5, 5)$ .

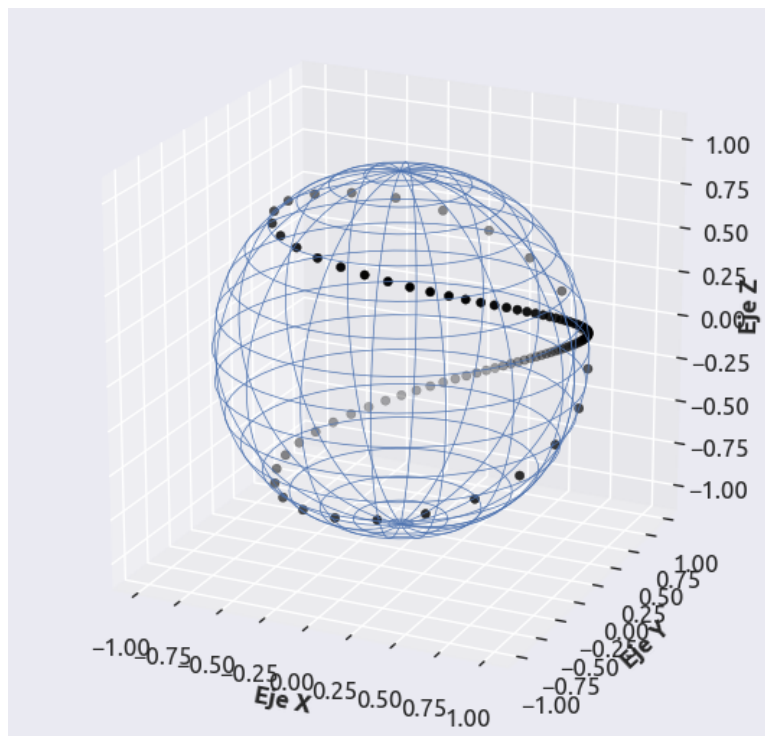
El segundo caso con  $A_1$  y  $A_2$  constantes y  $\Delta\alpha_x$  variable en  $(-\pi, \pi)$ :

#### 4. SIMULACIÓN DE ESTADOS DE POLARIZACIÓN

---

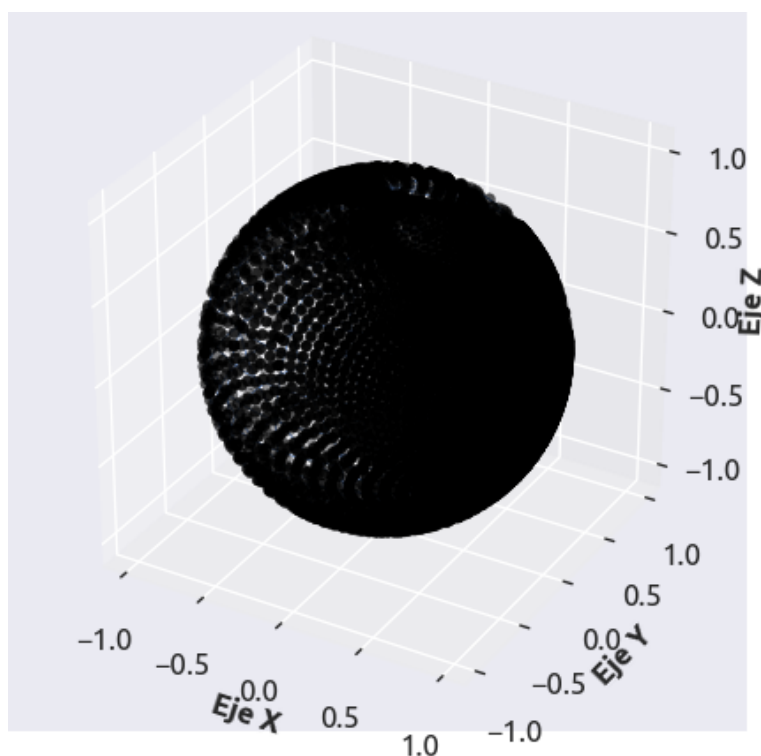


**Figura 4.24:**  $A$ ,  $\sigma$ ,  $\alpha$ , y  $\alpha_x$  cuando  $A_1 = 4$  y  $A_2 = 2$  y  $\Delta\alpha_x$  variable en  $(-\pi, \pi)$ :



**Figura 4.25:** Esfera de Poincaré con  $A_1 = 4$ ,  $A_2 = 2$  y  $\Delta\alpha_x$  variable en  $(-\pi, \pi)$ .

Y el ultimo caso, con  $A_1$ ,  $A_2$  y  $\Delta\alpha_x$  variables:



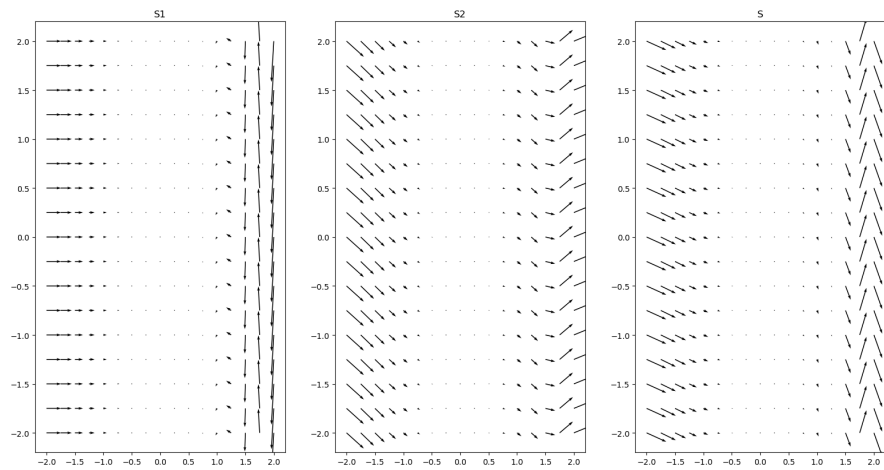
**Figura 4.26:** Esfera de Poincare con  $A_1, A_2$  variando en  $(-5, 5)$  y  $\Delta\alpha_x$  variable en  $(-\pi, \pi)$ .

#### 4.4. Campo Vectorial

Se simuló un frente de onda con la suma de los estados de polarización estudiados en la sección 3.2 haciendo variar los parámetros como funciones exponenciales, gaussianas y polinomios con valores arbitrarios, con el fin de visualizar su comportamiento.

#### 4. SIMULACIÓN DE ESTADOS DE POLARIZACIÓN

---



**Figura 4.27:** Superposición de dos frentes de onda, la primera figura representa un frente onda con los parámetros descritos, la siguiente otro frente de onda y la tercera la superposición de estos

---

## Capítulo 5

# Conclusiones

---

A lo largo del capítulo 2 se ha estudiado a detalle las características de los estados de polarización es las formas en las que se presentan; lineal, circular y elíptico, siendo este último el caso mas general, es decir que a partir del estado elíptico podemos llegar a los estados circular y lineal. También se estudio una forma práctica de describir los estados de polarización a través de los parámetros de Stokes los cuales terminaron siendo una alternativa matemática conveniente para describir los estados de polarización en términos de la intensidad. Estos parámetros pueden verse en la esfera Poincare, el cual resultó ser un recurso bastante didáctico para mostrar los estados de polarizaron. La esfera de Poincaré representa a los tres últimos parámetros de Stokes en coordenadas esféricas.

En el capítulo 3 se estudio matemáticamente la superposición de dos campos de la cual resultaba un tercer campo con otro estado de polarización y las relaciones inversas que conformaban este último. Se mostró que podemos obtener dos expresiones para las amplitudes  $A_1$  y  $A_2$  que en principio parecían que no guardaban relación con las fases de los campos que los conformaban y que sin embargo están, en efecto relacionadas entre si.

Con las simulaciones mostradas en el 4 podemos obtener una visualización practica y didáctica de lo que ocurre con la superposición de dos diferentes estados de polarización. Ademas también fue realizada una animación en la cual pueden modularse las amplitudes los estados para visualizar el comportamiento de su superposición, el código de dicho programa esta en el apéndice B. También se presento los parámetros de Stokes mapeados a la esfera de Poincaré, los cuales con los parámetros de simulación pudimos cubrir un porcentaje significativo en la esfera, utilizando otros parámetros debería ser posible cubrir toda la esfera.

### 5.1. Trabajo a futuro

- Explorar casos útiles de la modulación de amplitud en no cuadratura para modular la polarización del campo resultante.
- Aplicar esta técnica a interferometría de corrimiento de fase.

---

## Apéndice A

# Simulación de la superposición de dos estados de polarización

---

```
1 import matplotlib.pyplot as plt
2 import numpy as np
3 from mpl_toolkits import mplot3d
4 from mpl_toolkits.mplot3d import Axes3D
5 import math
6 from matplotlib.widgets import Slider
7 from numpy import exp, arange
8 from pylab import meshgrid, cm, imshow, contour, clabel,
9 colorbar, axis, title, show
10 from mpl_toolkits.mplot3d import Axes3D
11 from matplotlib import cm
12 from matplotlib.ticker import LinearLocator, FormatStrFormatter
13 import matplotlib.pyplot as plt
14 from matplotlib import style
15 from matplotlib import rcParams
16
17 plt.rcParams['font.family'] = 'serif'
18 plt.rcParams['font.serif'] = 'Ubuntu'
19 plt.rcParams['font.monospace'] = 'Ubuntu Mono'
20 plt.rcParams['font.size'] = 10
21 plt.rcParams['axes.labelsize'] = 10
22 plt.rcParams['axes.labelweight'] = 'bold'
23 plt.rcParams['xtick.labelsize'] = 8
24 plt.rcParams['ytick.labelsize'] = 8
25 plt.rcParams['legend.fontsize'] = 15
26 plt.rcParams['figure.titlesize'] = 12
27
```

## A. SIMULACIÓN DE LA SUPERPOSICIÓN DE DOS ESTADOS DE POLARIZACIÓN

---

```
28 style.use('seaborn')
29
30 lim = np.pi; step = 0.1; w = 2.0; t = np.arange(-lim, lim, step)
31
32 alpha_1x = np.pi/5
33 alpha_1y = np.pi/3
34 alpha_2x = np.pi/2
35 alpha_2y = np.pi/7
36 d_alphax = alpha_2x - alpha_1x
37 d_alphay = alpha_2y - alpha_1y
38 sigma1 = 1
39 sigma2 = 1
40 S01 = 3
41 S02 = 3
42
43 X01 = S01 * np.cos(sigma1)
44 Y01 = S01 * np.sin(sigma1)
45 X02 = S02 * np.cos(sigma2)
46 Y02 = S02 * np.sin(sigma2)
47
48 S0 = np.sqrt(X01 ** 2 + X02 ** 2 + 2 * X01 * X02 * np.cos(d_alphax)
49 + Y01 ** 2 + Y02 ** 2 + 2 * Y01 * Y02 * np.cos(d_alphay))
50
51 sigma0 = np.arctan2(np.sqrt(Y01 ** 2 + Y02 ** 2
52 + 2 * Y01 * Y02 * np.cos(d_alphay)),
53 np.sqrt(X01 ** 2 + X02 ** 2 + 2 * X01 * X02 * np.cos(d_alphax)))
54
55 y1 = X01 * np.sin(alpha_1x) + X02 * np.sin(alpha_2x)
56 x1 = X01 * np.cos(alpha_1x) + X02 * np.cos(alpha_2x)
57 y2 = Y01 * np.sin(alpha_1y) + Y02 * np.sin(alpha_2y)
58 x2 = Y01 * np.cos(alpha_1y) + Y02 * np.cos(alpha_2y)
59 x = y1 / x1
60 y = y2 / x1
61
62 alphax = np.arctan2(y1, x1)
63 alphay = np.arctan2(y2, x2)
64
65 k1 = S0 * np.cos(sigma0) * np.cos(w * t + alphax)
66 k2 = S0 * np.sin(sigma0) * np.cos(w * t + alphay)
67
68 S1x = X01 * np.cos(w * t + alpha_1x)
69 S1y = Y01 * np.cos(w * t + alpha_1y)
70 S2x = X02 * np.cos(w * t + alpha_2x)
71 S2y = Y02 * np.cos(w * t + alpha_2y)
```

---

```

72
73 Sx = S1x + S2x
74 Sy = S1y + S2y
75
76 def Elipticidad(sigma, alpha):
77     elip = 0.5*np.arcsin(np.sin(2*sigma)*np.sin(alpha))
78     return elip
79
80 print('Elipticidad E1=',Elipticidad(sigma1,alpha_1y-alpha_1x))
81 print('Elipticidad E2=',Elipticidad(sigma2,alpha_2y-alpha_2x))
82 print('Elipticidad E=', Elipticidad(sigma0, alphay - alphax))
83
84
85
86 x = 6
87 y = 8
88
89 plt.subplot(131)
90 plt.plot(S1x, S1y)
91 #plt.grid()
92 plt.axis('equal')
93 plt.axis([-x, x, -y, y])
94 #plt.axis('normal')
95 plt.title(r'$\vec{S}_{1}$ (\vec{r},t)$', fontsize=16)
96
97 plt.subplot(132)
98 plt.plot(S2x, S2y)
99 plt.axis('equal')
100 plt.axis([-x, x, -y, y])
101 #plt.axis('normal')
102 plt.title(r'$\vec{S}_{2}$ (\vec{r},t)$', fontsize=16)
103 #plt.grid()
104
105 plt.subplot(133)
106 plt.axis('equal')
107 plt.axis([-x, x, -y, y])
108 plt.plot(k1, k2)
109 #plt.axis('normal')
110 plt.title('Resultante' r'$\ \vec{S}, (\ \vec{r},t)$', fontsize=16)
111
112 #plt.grid()
113
114 plt.show()
115 '''

```

---

## A. SIMULACIÓN DE LA SUPERPOSICIÓN DE DOS ESTADOS DE POLARIZACIÓN

---

```
116 plt.subplot(121)
117 plt.plot(Sx, Sy)
118 plt.grid()
119 plt.subplot(122)
120 plt.title('Directas')
121 plt.plot(k1, k2)
122 plt.grid()
123 plt.show()
124 '''
125 def S0_(S01, S02):
126     S0 = np.sqrt(S01 * np.cos(sigma1)** 2 + S02 * np.cos(sigma2)** 2 +
127     2 * S01 * np.cos(sigma1) * S02 * np.cos(sigma2) * np.cos(d_alphax) +
128     S01 * np.sin(sigma1)** 2 + S02 * np.sin(sigma2)** 2 +
129     2 * S01 * np.sin(sigma1) * S02 * np.sin(sigma2) * np.cos(d_alphay))
130     return S0
131
132 def sigma0_(S01, S02):
133     sigma0 = np.arctan2(np.sqrt(S01 * np.sin(sigma1)** 2
134     + S02 * np.sin(sigma2)** 2 +
135     2 * S01 * np.sin(sigma1) * S02 * np.sin(sigma2) * np.cos(d_alphay)),
136     np.sqrt(S01 * np.cos(sigma1)** 2 + S02 * np.cos(sigma2)** 2
137     + 2 * S01 * np.cos(sigma1) * S02 * np.cos(sigma2) * np.cos(d_alphax)))
138     return sigma0
139
140
141 def alphax_(S01, S02):
142     X01 = S01 * np.cos(sigma1)
143     X02 = S02 * np.cos(sigma2)
144     y1 = X01 * np.sin(alpha_1x) + X02 * np.sin(alpha_2x)
145     x1 = X01 * np.cos(alpha_1x) + X02 * np.cos(alpha_2x)
146     alphax = np.arctan2(y1, x1)
147
148     return alphax
149
150 def alphay_(S01, S02):
151     Y01 = S01 * np.sin(sigma1)
152     Y02 = S02 * np.sin(sigma2)
153     y2 = Y01 * np.sin(alpha_1y) + Y02 * np.sin(alpha_2y)
154     x2 = Y01 * np.cos(alpha_1y) + Y02 * np.cos(alpha_2y)
155     alphay = np.arctan2(y2, x2)
156
157     return alphay
158
159 '''
```

---

```

160 fig = plt.figure(figsize=plt.figaspect(1.2))
161
162 ax = fig.add_subplot(1, 2, 1, projection='3d')
163 X = np.arange(0, lim, step)
164 Y = np.arange(0, lim, step)
165 X, Y = np.meshgrid(X, Y)
166 Z = SO_(X, Y)
167 surf=ax.plot_surface(X, Y, Z, rstride=1, cstride=1,
168 cmap='coolwarm',
169 linewidth=0, antialiased=False)
170 fig.colorbar(surf, shrink=0.5, aspect=10)
171 ax.set_xlabel('SO1')
172 ax.set_ylabel('SO2')
173 ax.set_zlabel('SO')
174 ax.set_title('SO')
175
176 ax = fig.add_subplot(1, 2, 2, projection='3d')
177 X = np.arange(0, lim, step)
178 Y = np.arange(0, lim, step)
179 X, Y = np.meshgrid(X, Y)
180 Z = sigma0_(X, Y)
181 surf = ax.plot_surface(X, Y, Z, rstride=1, cstride=1, cmap='coolwarm',
182 linewidth=0, antialiased=False)
183 fig.colorbar(surf, shrink=0.5, aspect=10)
184 ax.set_xlabel('SO1')
185 ax.set_ylabel('SO2')
186 ax.set_zlabel('Sigma0')
187 ax.set_title('Sigma0')
188 plt.show()
189
190 fig = plt.figure()
191 ax = fig.gca(projection='3d')
192
193 X = np.arange(0, lim, step)
194 Y = np.arange(0, lim, step)
195 X, Y = np.meshgrid(X, Y)
196 Z = d_alpha_(X, Y)
197 Z1 = alpha_x_(X, Y)
198 Z2 = alphay_(X, Y)
199
200 surf = ax.plot_surface(X, Y, Z, rstride=1, cstride=1, cmap='inferno',
201 linewidth=0, antialiased=False)
202 surf1 = ax.plot_surface(X, Y, Z1, rstride=1, cstride=1, cmap='viridis',
203 linewidth=0, antialiased=False)

```

## A. SIMULACIÓN DE LA SUPERPOSICIÓN DE DOS ESTADOS DE POLARIZACIÓN

---

```
204 surf2 = ax.plot_surface(X, Y, Z2, rstride=1, cstride=1, cmap='spring',
205 linewidth=0, antialiased=False)
206
207 fig.colorbar(surf, shrink=0.5, aspect=10, label='d_alpha')
208 fig.colorbar(surf1, shrink=0.5, aspect=10, label='alpha_x')
209 fig.colorbar(surf2, shrink=0.5, aspect=10, label='alpha_y')
210
211 ax.set_xlabel('S01')
212 ax.set_ylabel('S02')
213 ax.set_zlabel('D_alpha')
214 ax.set_title('Diferencia de Fase')
215
216 plt.show()
217 '''
```

---

---

## Apéndice B

# Animación de la superposición de dos estados de polarización

---

```
1 import matplotlib.pyplot as plt
2 import numpy as np
3 from mpl_toolkits import mplot3d
4 from mpl_toolkits.mplot3d import Axes3D
5 import math
6 from matplotlib.widgets import Slider
7 from numpy import exp,arange
8 from pylab import meshgrid, cm, imshow, contour,
9 clabel, colorbar, axis, title, show
10 from mpl_toolkits.mplot3d import Axes3D
11 from matplotlib import cm
12 from matplotlib.ticker import LinearLocator, FormatStrFormatter
13 import matplotlib.pyplot as plt
14 from matplotlib import style
15 style.use('seaborn')
16 plt.rcParams['font.family'] = 'serif'
17 plt.rcParams['font.serif'] = 'Ubuntu'
18 plt.rcParams['font.monospace'] = 'Ubuntu Mono'
19 plt.rcParams['font.size'] = 10
20 plt.rcParams['axes.labelsize'] = 10
21 plt.rcParams['axes.labelweight'] = 'bold'
22 plt.rcParams['xtick.labelsize'] = 8
23 plt.rcParams['ytick.labelsize'] = 8
24 plt.rcParams['legend.fontsize'] = 15
25 plt.rcParams['figure.titlesize'] = 12
26
27 lim = 10.0; n = 1000; w=2.0; t= np.linspace(-lim,lim,n)
```

## B. ANIMACIÓN DE LA SUPERPOSICIÓN DE DOS ESTADOS DE POLARIZACIÓN

---

```
28 #FASES INICIALES
29 alpha_1x = 1
30 alpha_1y = 2
31 alpha_2x = 2
32 alpha_2y = 7
33 sigma1 = 4
34 sigma2 = np.pi / 4
35 d_alphax = alpha_2x - alpha_1x
36 d_alphay = alpha_2y - alpha_1y
37
38 S01 = 3.0
39 S02 = 10.0
40
41 X01 = S01 * np.cos(sigma1)
42 Y01 = S01 * np.sin(sigma1)
43 X02 = S02 * np.cos(sigma2)
44 Y02 = S02 * np.sin(sigma2)
45
46 #CALCULO DE PARAMETROS
47 # Amplitud de campo
48 S0 = np.sqrt(X01 ** 2 + X02 ** 2 + 2 * X01 * X02 * np.cos(d_alphax)
49 + Y01 ** 2 + Y02 ** 2 + 2 * Y01 * Y02 * np.cos(d_alphay))
50
51 # Estado de polarizacion sigma0,d_alpha
52 sigma0 = np.arctan2(np.sqrt(Y01 ** 2 + Y02 ** 2
53 + 2 * Y01 * Y02 * np.cos(d_alphay)),
54 np.sqrt(X01 ** 2 + X02 ** 2 + 2 * X01 * X02 * np.cos(d_alphax)))
55
56 y1 = X01 * np.sin(alpha_1x) + X02 * np.sin(alpha_2x)
57 x1 = X01 * np.cos(alpha_1x) + X02 * np.cos(alpha_2x)
58 y2 = Y01 * np.sin(alpha_1y) + Y02 * np.sin(alpha_2y)
59 x2 = Y01 * np.cos(alpha_1y) + Y02 * np.cos(alpha_2y)
60 x = y1 / x1
61 y = y2 / x1
62
63 alphax = np.arctan2(y1, x1)
64 alphay = np.arctan2(y2, x2)
65
66 #Elipse 1
67 def S1x(S01):
68     S1x = S01 * np.cos(sigma1) * np.cos(w * t + alpha_1x)
69     return S1x
70 def S1y(S01):
71     S1y = S01 * np.sin(sigma1) * np.cos(w * t + alpha_1y)
```

---

```

72 return S1y
73 #Elipse 2
74 def S2x(S02):
75     S2x = S02 * np.cos(sigma2) * np.cos(w * t + alpha_2x)
76     return S2x
77 def S2y(S02):
78     S2y = S02 * np.sin(sigma2) * np.cos(w * t + alpha_2y)
79     return S2y
80
81 # ELIPSE RESULTANTE
82 def Sx(S01, S02):
83     X01 = S01 * np.cos(sigma1)
84     Y01 = S01 * np.sin(sigma1)
85     X02 = S02 * np.cos(sigma2)
86     Y02 = S02 * np.sin(sigma2)
87
88 #CALCULO DE PARAMETROS
89 S0 = np.sqrt(X01 ** 2 + X02 ** 2 + 2 * X01 * X02 * np.cos(d_alphax)
90 + Y01 ** 2 + Y02 ** 2 + 2 * Y01 * Y02 * np.cos(d_alphay))
91
92 sigma0 = np.arctan2(np.sqrt(Y01 ** 2 + Y02 ** 2
93 + 2 * Y01 * Y02 * np.cos(d_alphay)),
94 np.sqrt(X01 ** 2 + X02 ** 2 + 2 * X01 * X02 * np.cos(d_alphax)))
95
96 y1 = X01 * np.sin(alpha_1x) + X02 * np.sin(alpha_2x)
97 x1 = X01 * np.cos(alpha_1x) + X02 * np.cos(alpha_2x)
98 alphax = np.arctan2(y1, x1)
99 Sx = S0 * np.cos(sigma0) * np.cos(w * t + alphax)
100 return Sx
101 def Sy(S01, S02):
102     X01 = S01 * np.cos(sigma1)
103     Y01 = S01 * np.sin(sigma1)
104     X02 = S02 * np.cos(sigma2)
105     Y02 = S02 * np.sin(sigma2)
106
107 #CALCULO DE PARAMETROS
108 S0 = np.sqrt(X01 ** 2 + X02 ** 2 + 2 * X01 * X02 * np.cos(d_alphax)
109 + Y01 ** 2 + Y02 ** 2 + 2 * Y01 * Y02 * np.cos(d_alphay))
110
111 sigma0 = np.arctan2(np.sqrt(Y01 ** 2 + Y02 ** 2
112 + 2 * Y01 * Y02 * np.cos(d_alphay)),
113 np.sqrt(X01 ** 2 + X02 ** 2 + 2 * X01 * X02 * np.cos(d_alphax)))
114
115 y2 = Y01 * np.sin(alpha_1y) + Y02 * np.sin(alpha_2y)

```

---

## B. ANIMACIÓN DE LA SUPERPOSICIÓN DE DOS ESTADOS DE POLARIZACIÓN

---

```
116 x2 = Y01 * np.cos(alpha_1y) + Y02 * np.cos(alpha_2y)
117 alphay = np.arctan2(y2, x2)
118 Sy = S0 * np.sin(sigma0) * np.cos(w * t + alphay)
119 return Sy
120
121 fig = plt.figure(figsize=plt.figaspect(1.2))
122
123 # Primera Elipse
124 ax=fig.add_subplot(1, 3, 1)
125 p0=0.1
126 plt.subplots_adjust(left=0.25, bottom=0.25)
127 l, = plt.plot(S1x(p0), S1y(p0), lw=1, color='red')
128 plt.title('ELIPSE 1',fontsize=20)
129 plt.grid()
130 plt.axis([-20, 20, -20, 20])
131
132 axcolor = 'lightgoldenrodyellow'
133 S01 = plt.axes([0.25, 0.16, 0.65, 0.03], facecolor=axcolor)
134 S_S01 = Slider(S01, r'$S_{01}$', 0.001, 15.0, valinit=0.1)
135
136 def update(val):
137     S01 = S_S01.val
138     l.set_xdata(S1x(S01))
139     l.set_ydata(S1y(S01))
140     fig.canvas.draw_idle()
141     S_S01.on_changed(update)
142
143 #Segunda Elipse
144 ax = fig.add_subplot(1, 3, 2)
145
146 plt.subplots_adjust(left=0.25, bottom=0.25)
147
148 k, = plt.plot(S2x(p0), S2y(p0), lw=1, color='b')
149 plt.title('ELIPSE 2', fontsize=20)
150 plt.grid()
151 plt.axis([-20, 20, -20, 20])
152
153 axcolor = 'lightgoldenrodyellow'
154 S02= plt.axes([0.25, 0.06, 0.65, 0.03], facecolor=axcolor)
155 S_S02 = Slider(S02, r'$S_{02}$', 0.001, 15, valinit=0.1)
156
157 def update2(val):
158     S02 = S_S02.val
159     k.set_xdata(S2x(S02))
```

---

```

160 k.set_ydata(S2y(S02))
161 fig.canvas.draw_idle()
162
163 S_S02.on_changed(update2)
164
165
166 # Resultante
167 ax = fig.add_subplot(1, 3, 3)
168
169 plt.subplots_adjust(left=0.25, bottom=0.25)
170 h, = plt.plot(Sx(p0,p0), Sy(p0,p0), lw=1, color='c')
171 plt.title('RESULTANTE', fontsize=20)
172 plt.grid()
173 plt.axis([-20, 20, -20, 20])
174
175 axcolor = 'lightgoldenrodyellow'
176 S01 = plt.axes([0.25, 0.16, 0.65, 0.03], facecolor=axcolor)
177 #Sigma1 = plt.axes([0.25, 0.11, 0.65, 0.03], facecolor=axcolor)
178 S02 = plt.axes([0.25, 0.06, 0.65, 0.03], facecolor=axcolor)
179 #Sigma2 = plt.axes([0.25, 0.01, 0.65, 0.03], facecolor=axcolor)
180
181 def update3(val):
182     S01 = S_S01.val
183     # Sigma1 = S_Sigma1.val
184     S02 = S_S02.val
185     # Sigma2 = S_Sigma2.val
186     #L1(S01,sigma1,S02,sigma2)
187     h.set_xdata(Sx(S01,S02))
188     h.set_ydata(Sy(S01,S02))
189     fig.canvas.draw_idle()
190
191 S_S01.on_changed(update3)
192 #S_Sigma1.on_changed(update3)
193 S_S02.on_changed(update3)
194 #S_Sigma2.on_changed(update3)
195
196 plt.show()

```

---



---

## Apéndice C

# Simulación de la superposición de dos frentes de onda

---

```
1 import matplotlib.pyplot as plt
2 import numpy as np
3 from numpy import ma
4
5
6 class Functions(object):
7     def __init__(self, lim, n):
8         self._lim = lim
9         self._n = n
10        self._X, self._Y = np.meshgrid(np.linspace(-lim,lim,n),
11        np.linspace(-lim,lim,n))
12        def polinomio(self, n):
13            x = self._X ** n
14            return x
15        def exponencial(self, n):
16            x = np.exp(n * self._X)
17            return x
18        def gaussiana(self, a, b, c):
19            x = a * (np.exp(-(self._X - b)** 2)) / (2 * c ** 2)
20            return x
21        def sin(self,n):
22            x = np.sin(self._X ** n)
23            return x
24
25        lim = 2
26        n = 10
27
```

## C. SIMULACIÓN DE LA SUPERPOSICIÓN DE DOS FRENTES DE ONDA

---

```
28 S01 = Functions(lim, n)
29 S02 = Functions(lim, n)
30 sigma1 = Functions(lim, n)
31 sigma2 = Functions(lim, n)
32 alpha_1x = Functions(lim, n)
33 alpha_1y = Functions(lim, n)
34 alpha_2x = Functions(lim, n)
35 alpha_2y = Functions(lim, n)
36
37
38 S1x = S01.polinomio(2) * np.cos(sigma1.exponencial(1)) *
39     np.cos(alpha_1x.exponencial(1))
40 S1y = S01.polinomio(4) * np.sin(sigma1.exponencial(5)) *
41     np.cos(alpha_1y.gausiana(1,0,1))
42 S2x = S02.polinomio(2) * np.cos(sigma2.gausiana(1,2,3)) *
43     np.cos(alpha_2x.gausiana(1,1,1))
44 S2y = S02.polinomio(2) * np.sin(sigma2.exponencial(1)) *
45     np.cos(alpha_2y.exponencial(1))
46
47 Sx = S1x + S2x
48 Sy = S1y + S2y
49
50
51 plt.figure()
52 plt.title('Campo Vectorial S1+S2')
53 Q = plt.quiver(S01._X, S01._Y, Sx, Sy, units='width')
54 qk = plt.quiverkey(Q, 0.9, 0.9, 2, r'$2 \frac{m}{s}$', labelpos='E',
55     coordinates='figure')
56 plt.show()
```

---

## Apéndice D

# Simulación de los parámetros de Stokes mapeados a la esfera de Poincare

---

```
1 import matplotlib.pyplot as plt
2 import numpy as np
3 from mpl_toolkits import mplot3d
4 from mpl_toolkits.mplot3d import Axes3D
5 import math
6 from matplotlib.widgets import Slider
7 from matplotlib import style
8 style.use('seaborn')
9
10 lim = np.pi; n = 100; w = 1.0; t = np.linspace(-lim, lim, n)
11
12 '''
13 alpha_1x = 0
14 alpha_2x = np.pi/2
15 alpha_1y = np.pi/2
16 alpha_2y = 0
17 sigma1 = np.pi/4
18 sigma2 = np.pi/4
19 d_alphax = alpha_2x - alpha_1x
20 d_alphay = alpha_2y - alpha_1y
21 '''
22
23 def alpha(alphax):
24 alpha = -np.arctan2(
25 np.tan(sigma1) * np.cos(alpha_2x + alpha_1x - 2 * alphax)
26 - np.tan(sigma2) * np.cos(alpha_1x + alpha_2y - 2 * alphax)
27 - np.tan(sigma1) * np.cos(alpha_1y - alpha_2x)
```

## D. SIMULACIÓN DE LOS PARÁMETROS DE STOKES MAPEADOS A LA ESFERA DE POINCARÉ

---

```
28 + np.tan(sigma2) * np.cos(alpha_2y - alpha_1x)
29 ,np.tan(sigma1) * np.sin(alpha_2x + alpha_1x - 2 * alphax)
30 - np.tan(sigma2) * np.sin(alpha_1x + alpha_2y - 2 * alphax)
31 - np.tan(sigma1) * np.sin(alpha_1y - alpha_2x)
32 + np.tan(sigma2) * np.sin(alpha_2y - alpha_1x)
33 )
34 return alpha
35
36 def sigma0_(alphax):
37 sigma0 = np.arctan2(np.sqrt((np.tan(sigma1)
38 * np.sin(alpha_2x - alphax) / np.sin(d_alphax))** 2
39 + (np.tan(sigma2) * np.sin(alphax - alpha_1x) / np.sin(d_alphax))** 2
40 + 2 * (np.tan(sigma1) * np.sin(alpha_2x - alphax) / np.sin(d_alphax))
41 * (np.tan(sigma2) * np.sin(alphax - alpha_1x) / np.sin(d_alphax))
42 * np.cos(alpha_2y - alpha_1y)), 1)
43 return sigma0
44
45
46 fig = plt.figure()
47 ax = fig.add_subplot(111, projection='3d')
48 u = np.linspace(0, 2 * np.pi, 50)
49 v = np.linspace(0, np.pi,50)
50 x = 1 * np.outer(np.cos(u), np.sin(v))
51 y = 1 * np.outer(np.sin(u), np.sin(v))
52 z = 1 * np.outer(np.ones(np.size(u)), np.cos(v))
53
54 ax.plot_wireframe(x, y, z, rstride = 3, cstride = 3, linewidth = 0.3)
55
56 def frange(start, end=None, inc=None):
57 "A range function, that does accept float increments..."
58
59 if end == None:
60 end = start + 0.0
61 start = 0.0
62
63 if inc == None:
64 inc = 1.0
65
66 L = []
67 while 1:
68 next = start + len(L) * inc
69 if inc > 0 and next >= end:
70 break
71 elif inc < 0 and next <= end:
```

---

```

72 break
73 L.append(next)
74 return L
75
76
77 for i in frange(-2*np.pi, 2*np.pi, 0.1):
78     alpha_1x=1
79     d_alphax=i
80     alpha_2x=alpha_1x + d_alphax
81     alpha1=np.pi / 2
82     alpha2= -np.pi / 2
83     alpha_1y=alpha_1x + alpha1
84     alpha_2y=alpha_1x + d_alphax + alpha2
85     sigma1=np.pi / 4
86     sigma2=np.pi / 4
87
88     S3 = np.cos(2 * sigma0_(t))
89     S2 = np.sin(2 * sigma0_(t)) * np.cos(2*np.unwrap(alpha(t)))
90     S1 = np.sin(2 * sigma0_(t)) * np.sin(2*np.unwrap(alpha(t)))
91     ax.scatter3D(S1, S2, S3, S3, c='black', label='sigma');
92
93
94     ax.set_xlabel('Eje X')
95     ax.set_ylabel('Eje Y')
96     ax.set_zlabel('Eje Z')
97     plt.show()

```

---



---

## Apéndice E

# Simulación de las relaciones inversas del estado resultante de dos estados de polarización

---

```
1 import matplotlib.pyplot as plt
2 import numpy as np
3 from mpl_toolkits import mplot3d
4 from mpl_toolkits.mplot3d import Axes3D
5 import math
6 from matplotlib.widgets import Slider
7 from matplotlib import style
8 style.use('seaborn-dark')
9
10 lim = np.pi; n=100; w=1.0; t= np.linspace(-lim,lim,n)
11 #FASES INICIALES
12 alpha_1x = 1.8
13 alpha_1y = 10
14 alpha_2x = 1.1
15 alpha_2y = 3.6
16 d_alphax = alpha_2x - alpha_1x
17 d_alphay = alpha_2y - alpha_1y
18
19 #####
20 ##### ECUACIONES DIRECTAS#####
21 #####
22
23 # Parametros iniciales
```

## E. SIMULACIÓN DE LAS RELACIONES INVERSAS DEL ESTADO RESULTANTE DE DOS ESTADOS DE POLARIZACIÓN

---

```
24 sigma1 = 4.3
25 sigma2 = 1.0
26 S01 = 3
27 S02 = 4
28
29 X01 = S01 * np.cos(sigma1)
30 Y01 = S01 * np.sin(sigma1)
31 X02 = S02 * np.cos(sigma2)
32 Y02 = S02 * np.sin(sigma2)
33
34
35 #CALCULO DE PARAMETROS
36 S0 = np.sqrt(X01 ** 2 + X02 ** 2 + 2 * X01 * X02 * np.cos(d_alphax)
37 + Y01 ** 2 + Y02 ** 2 + 2 * Y01 * Y02 * np.cos(d_alphay))
38
39 sigma0 = np.arctan2(np.sqrt(Y01 ** 2 + Y02 ** 2
40 + 2 * Y01 * Y02 * np.cos(d_alphay)),
41 np.sqrt(X01 ** 2 + X02 ** 2 + 2 * X01 * X02 * np.cos(d_alphax)))
42
43 alphax=np.arctan2(X01 * np.sin(alpha_1x) + X02 * np.sin(alpha_2x),
44 X01 * np.cos(alpha_1x) + X02 * np.cos(alpha_2x))
45
46 alphay=np.arctan2(Y01 * np.sin(alpha_1y) + Y02 * np.sin(alpha_2y),
47 Y01 * np.cos(alpha_1y) + Y02 * np.cos(alpha_2y))
48 print(alphax, alphay)
49 '''
50 y2 = X01 * np.sin(alpha_1x) + X02 * np.sin(alpha_2x)
51 x2 = X01 * np.cos(alpha_1x) + X02 * np.cos(alpha_2x)
52 y1 = Y01 * np.sin(alpha_1y) + Y02 * np.sin(alpha_2y)
53 x1 = Y01 * np.cos(alpha_1y) + Y02 * np.cos(alpha_2y)
54 x = y1 / x1
55 y = y2 / x1
56 '''
57 '''
58 alphax = np.arctan2(y2, x2)
59 alphay = np.arctan2(y1, x1)
60
61 d_alpha = np.arctan2(y1*x2-y2*x1,x1*x2+y1*y2)
62 '''
63 '''
64 print('-----')
65 print('S0=', S0)
66 print('sigma0=', sigma0)
67 print('dalpha=', alphay - alphax)
```

---

```

68 print('dalpha=', d_alpha)
69 '''
70
71
72 #ECUACIONES CON LOS PARAMETROS CALCULADOS
73 S1x_directa = S0 * np.cos(sigma0) * np.cos(w * t + alphax)
74 S1y_directa = S0 * np.sin(sigma0) * np.cos(w * t + alphay)
75
76 #####
77 ##### ECUACIONES Inversas#####
78 #####
79
80 #Parametros iniciales
81
82 Cx = S0 * np.cos(sigma0)
83 Cy = S0 * np.sin(sigma0)
84
85 S01x = (Cx / np.sin(alpha_2x - alpha_1x)) * (np.sin(alpha_2x - alphax) /
86 np.cos(sigma1))
87 S02x = (Cx / np.sin(alpha_2x - alpha_1x)) * (np.sin(alphax - alpha_1x) /
88 np.cos(sigma2))
89
90 S01y = (Cy / np.sin(alpha_2y - alpha_1y)) * (np.sin(alpha_2y - alphay) /
91 np.sin(sigma1))
92 S02y = (Cy / np.sin(alpha_2y - alpha_1y)) * (np.sin(alphay - alpha_1y) /
93 np.sin(sigma2))
94
95 S01 = S01x
96 S02 = S02x
97
98
99 '''
100 A1 = S0*((np.cos(sigma0)) * np.sin(sigma2) * np.cos(alphay - alpha_2y) -
101 np.sin(sigma0) * np.cos(sigma2) * np.cos(alphax - alpha_2x)) / (
102 np.cos(sigma1) * np.cos(alphax - alpha_1x)
103 * np.sin(sigma2) * np.cos(alphay - alpha_2y)
104 -np.sin(sigma1) * np.cos(alphay - alpha_1y)
105 * np.cos(sigma2) * np.cos(alphax - alpha_2x)
106 )
107
108 A2 = S0*(np.cos(sigma0) * np.sin(sigma1) * np.cos(alphay - alpha_1y) -
109 np.sin(sigma0) * np.cos(sigma1) * np.cos(alphax - alpha_1x)) / (
110 np.cos(sigma2) * np.cos(alphax - alpha_2x)
111 * np.sin(sigma1) * np.cos(alphay - alpha_1y)

```

---

## E. SIMULACIÓN DE LAS RELACIONES INVERSAS DEL ESTADO RESULTANTE DE DOS ESTADOS DE POLARIZACIÓN

---

```
112     -np.sin(sigma2) * np.cos(alpha2y - alpha_2y)
113     * np.cos(sigma1) * np.cos(alpha1x - alpha_1x)
114     )
115     '''
116
117     S1x_inversa = (S01 * np.cos(sigma1) * np.cos(w * t + alpha_1x)
118 + S02 * np.cos(sigma2) * np.cos(w * t + alpha_2x))
119
120     S1y_inversa = ( S01 * np.sin(sigma1) * np.cos(w * t + alpha_1y)
121 + S02 * np.sin(sigma2) * np.cos(w * t + alpha_2y))
122
123
124
125     plt.subplot(121)
126     plt.title('Directas')
127     plt.plot(S1x_directa, S1y_directa)
128     plt.grid()
129     plt.subplot(122)
130     plt.title('Inversas')
131     plt.plot(S1x_inversa, S1y_inversa)
132     plt.grid()
133     plt.show()
```

---

## Apéndice F

# Modulación de amplitud mediante

# NQAM

---

```
1 import matplotlib.pyplot as plt
2 import numpy as np
3 from mpl_toolkits import mplot3d
4 from matplotlib import style
5 from matplotlib import rcParams
6 plt.rcParams['font.family'] = 'serif'
7 plt.rcParams['font.serif'] = 'Ubuntu'
8 plt.rcParams['font.monospace'] = 'Ubuntu Mono'
9 plt.rcParams['font.size'] = 10
10 plt.rcParams['axes.labelsize'] = 10
11 plt.rcParams['axes.labelweight'] = 'bold'
12 plt.rcParams['xtick.labelsize'] = 8
13 plt.rcParams['ytick.labelsize'] = 8
14 plt.rcParams['legend.fontsize'] = 15
15 plt.rcParams['figure.titlesize'] = 12
16 style.use('ggplot')
17 alpha_1 = 0.5
18 alpha_2 = np.pi / 2
19 d_alpha = alpha_2 - alpha_1
20 lim=2*np.pi
21 n=2000
22 t=np.linspace(-lim,lim,n)
23 A=10.0
24 Alpha = 10.0
25
26
27 #Matriz para modulacion de fase
```

## F. MODULACIÓN DE AMPLITUD MEDIANTE NQAM

---

```
28 def matrix(x):
29     t=np.linspace(-np.pi/4,np.pi/4,n)
30     x=t**2
31     y=t**2
32     X,Y = np.meshgrid(x,x)
33     return X+Y
34     #Matriz para modulacion de amplitud
35 def matrix2(x):
36     t=np.linspace(-lim,lim,n)
37     x=t**2
38     y=t**2
39     X,Y = np.meshgrid(x,y)
40     return X + Y
41
42 def matrix3(x):
43     t=np.linspace(-lim,lim,n)
44     x=np.sin(t)**2
45     y=t**2
46     X,Y = np.meshgrid(x,y)
47     return X
48
49 Amplitud_coconstante = np.zeros((n, n)) + 4
50 phase_constante = np.zeros((n, n)) + np.pi/3
51
52 gamma = matrix(Alpha)
53
54 A1_amplitudconstante = -Amplitud_coconstante * np.sin(matrix(gamma)
55 - alpha_1-d_alpha) / np.sin(d_alpha)
56 A2_amplitudconstante = Amplitud_coconstante * np.sin(matrix(gamma)
57 - alpha_1) / np.sin(d_alpha)
58
59
60 A1_phaseconst = -matrix2(A)*np.sin(phase_constante - alpha_1
61 - d_alpha)/np.sin(d_alpha)
62 A2_phaseconst = matrix2(A) * np.sin(phase_constante - alpha_1)
63 / np.sin(d_alpha)
64
65 A1_ = -matrix2(A) * np.sin(matrix3(gamma)-alpha_1-d_alpha)
66 / np.sin(d_alpha)
67 A2_ = matrix2(A) * np.sin(matrix3(gamma) -alpha_1)
68 / np.sin(d_alpha)
69
70 '''
71 plt.matshow(matrix(1))
```

---

```

72 plt.title('Matrix')
73 plt.colorbar()
74 plt.show()
75 ax = plt.axes(projection='3d')
76 ax.plot_surface(X, Y, matrix(1), rstride=1, cstride=1,
77 cmap='viridis', edgecolor='none')
78 ax.set_title('Matrix')
79 ax.set_xlabel('x')
80 ax.set_ylabel('y')
81 ax.set_zlabel('z');
82 plt.show() '''
83
84 plt.subplot(221)
85 plt.imshow(matrix(gamma), cmap='Reds')
86 plt.title(r'$\alpha$')
87 plt.suptitle('Modulación de fase con amplitud constante', fontsize=16)
88 plt.gca().invert_yaxis()
89 plt.colorbar()
90 plt.grid()
91
92 plt.subplot(222)
93 plt.imshow(Amplitud_coconstante, cmap='Reds')
94 plt.title(r'$A$')
95 plt.colorbar()
96 plt.gca().invert_yaxis()
97 plt.grid()
98
99 plt.subplot(223)
100 plt.imshow(A1_amplitdconstante, cmap='Reds')
101 plt.title(r'$A_1$')
102 plt.colorbar()
103 plt.gca().invert_yaxis()
104 plt.grid()
105
106 plt.subplot(224)
107 plt.imshow(A2_amplitudconstnate, cmap='Reds')
108 plt.title(r'$A_2$')
109 plt.grid()
110 plt.colorbar()
111 plt.gca().invert_yaxis()
112 plt.show()
113
114 plt.subplot(221)
115 plt.imshow(phase_constante, cmap='Reds')

```

```

116 plt.title(r'\alpha$')
117 plt.suptitle('Modulación de amplitud con fase constante', fontsize=16)
118 plt.grid()
119 plt.colorbar()
120 plt.gca().invert_yaxis()
121
122 plt.subplot(222)
123 plt.imshow(matrix2(A), cmap='Reds')
124 plt.title(r'$A$')
125 plt.grid()
126 plt.colorbar()
127 plt.gca().invert_yaxis()
128
129 plt.subplot(223)
130 plt.imshow(A1_phaseconst, cmap='Reds')
131 plt.title(r'$A_1$')
132 plt.grid()
133 plt.colorbar()
134 plt.gca().invert_yaxis()
135
136 plt.subplot(224)
137 plt.imshow(A2_phaseconst, cmap='Reds')
138 plt.title(r'$A_2$')
139 plt.grid()
140 plt.colorbar()
141 plt.gca().invert_yaxis()
142
143 plt.show()
144
145 plt.subplot(221)
146 plt.imshow(matrix(gamma), cmap='Reds')
147 plt.title(r'\alpha$')
148 plt.suptitle('Modulación de amplitud y fase', fontsize=16)
149 plt.grid()
150 plt.colorbar()
151 plt.gca().invert_yaxis()
152
153 plt.subplot(222)
154 plt.imshow(matrix2(A), cmap='Reds')
155 plt.title(r'$A$')
156 plt.grid()
157 plt.colorbar()
158 plt.gca().invert_yaxis()
159

```

---

```
160 plt.subplot(223)
161 plt.imshow(A2_, cmap='Reds')
162 plt.title(r'$A_1$')
163 plt.grid()
164 plt.colorbar()
165 plt.gca().invert_yaxis()
166
167 plt.subplot(224)
168 plt.imshow(A1_, cmap='Reds')
169 plt.title(r'$A_2$')
170 plt.grid()
171 plt.colorbar()
172 plt.gca().invert_yaxis()
173
174 plt.show()
```

---



---

## Apéndice G

# Modulación de amplitud vectorial mediante NQAM

---

```
1 import matplotlib.pyplot as plt
2 import numpy as np
3 from matplotlib import style
4 from matplotlib import rcParams
5 plt.rcParams['font.family'] = 'serif'
6 plt.rcParams['font.serif'] = 'Ubuntu'
7 plt.rcParams['font.monospace'] = 'Ubuntu Mono'
8 plt.rcParams['font.size'] = 10
9 plt.rcParams['axes.labelsize'] = 10
10 plt.rcParams['axes.labelweight'] = 'bold'
11 plt.rcParams['xtick.labelsize'] = 8
12 plt.rcParams['ytick.labelsize'] = 8
13 plt.rcParams['legend.fontsize'] = 15
14 plt.rcParams['figure.titlesize'] = 12
15 style.use('seaborn')
16
17 lim = np.pi; n=1000; w=1.0; t= np.linspace(-lim,lim,n)
18
19
20 class Modulacion(object):
21     def Modulacion_1(self, a):
22         lim=5
23         t= np.linspace(-lim,lim,n)
24         x=t**a
25         y=t**a
26         X,Y = np.meshgrid(x,y)
27         return X + Y
```

```

28 def Modulacion_2(self, a):
29     lim = 2
30     t = np.linspace(-lim, lim, n)
31     x = np.cos(t)**a
32     y = np.sin(t)**a
33     X,Y = np.meshgrid(x,y)
34     return X + Y
35 def Modulacion_phase(self, a):
36     t= np.linspace(-np.pi/2,np.pi/2,n)
37     x=np.cos(t)**a
38     y=np.sin(t)**a
39     X,Y = np.meshgrid(x,y)
40     return X + Y
41 def Constante(self, cons):
42     valor = np.zeros((n, n)) + cons
43     return valor
44
45 value = Modulacion()
46
47
48
49 alpha_1x = np.pi/10
50 alpha_1y = np.pi/9
51 alpha_2x = np.pi/6
52 alpha_2y = np.pi/8
53 d_alpha_1 = alpha_1y - alpha_1x
54 d_alpha_2 = alpha_2y - alpha_2x
55 d_alphax = alpha_2x - alpha_1x
56 d_alphay = alpha_2y - alpha_1y
57 sigma1 = 3
58 sigma2 = 2
59
60 alphax = value.Constante(1.2)
61 S0 = value.Modulacion_2(2)
62 #CALCULO DE PARAMETROS
63 '''
64 X01 = S01 * np.cos(sigma1)
65 Y01 = S01 * np.sin(sigma1)
66 X02 = S02 * np.cos(sigma2)
67 Y02 = S02 * np.sin(sigma2)
68
69 S0 = np.sqrt(X01 ** 2 + X02 ** 2 + 2 * X01 * X02 * np.cos(d_alphax)
70 + Y01 ** 2 + Y02 ** 2 + 2 * Y01 * Y02 * np.cos(d_alphay))
71

```

---

```

72  sigma0 = np.arctan2(np.sqrt(Y01 ** 2 + Y02 ** 2 + 2 * Y01 * Y02 * np.cos(d_alphay)),
73  np.sqrt(X01 ** 2 + X02 ** 2 + 2 * X01 * X02 * np.cos(d_alphax)))
74
75  alphax=np.arctan2(X01 * np.sin(alpha_1x) + X02 * np.sin(alpha_2x),
76  X01 * np.cos(alpha_1x) + X02 * np.cos(alpha_2x))
77
78  alphay=np.arctan2(Y01 * np.sin(alpha_1y) + Y02 * np.sin(alpha_2y),
79  Y01 * np.cos(alpha_1y) + Y02 * np.cos(alpha_2y))
80  print(alphax,alphay)
81  '''
82
83  def alpha_(alphax):
84  alpha = -np.arctan2(
85  np.tan(sigma1) * np.cos(alpha_2x + alpha_1x - 2 * alphax)
86  - np.tan(sigma2) * np.cos(alpha_1x + alpha_2y - 2 * alphax)
87  - np.tan(sigma1) * np.cos(alpha_1y - alpha_2x)
88  + np.tan(sigma2) * np.cos(alpha_2y - alpha_1x)
89  ,np.tan(sigma1) * np.sin(alpha_2x + alpha_1x - 2 * alphax)
90  - np.tan(sigma2) * np.sin(alpha_1x + alpha_2y - 2 * alphax)
91  - np.tan(sigma1) * np.sin(alpha_1y - alpha_2x)
92  + np.tan(sigma2) * np.sin(alpha_2y - alpha_1x)
93  )
94  return alpha
95
96
97  def sigma0_(alphax):
98  sigma0 = np.arctan2(np.sqrt((np.tan(sigma1) * np.sin(alpha_2x - alphax) / np.sin(d_alphax))** 2
99  + (np.tan(sigma2) * np.sin(alphax - alpha_1x) / np.sin(d_alphax))** 2
100  + 2 * (np.tan(sigma1) * np.sin(alpha_2x - alphax) / np.sin(d_alphax))
101  * (np.tan(sigma2) * np.sin(alphax - alpha_1x) / np.sin(d_alphax))
102  * np.cos(alpha_2y - alpha_1y)), 1)
103  return sigma0
104
105
106  Cx = S0 * np.cos(sigma0_(alphax))
107  Cy = S0 * np.sin(sigma0_(alphax))
108
109  S01x = (Cx / np.sin(alpha_2x - alpha_1x)) * (np.sin(alpha_2x - alpha_(alphax)) / np.cos(sigma1))
110  S02x = (Cx / np.sin(alpha_2x - alpha_1x)) * (np.sin(alpha_(alphax) - alpha_1x) / np.cos(sigma2))
111  print(alpha_(alphax).shape)
112
113  '''
114  S01y = (Cy / np.sin(alpha_2y - alpha_1y)) * (np.sin(alpha_2y - alphay) / np.sin(sigma1))
115  S02y = (Cy / np.sin(alpha_2y - alpha_1y)) * (np.sin(alphay - alpha_1y) / np.sin(sigma2))

```

---

```
116     '''
117
118     plt.subplot(231)
119     plt.imshow(S0, cmap='Reds')
120     plt.title(r'$S_{0}$')
121     plt.grid()
122     plt.colorbar()
123     plt.gca().invert_yaxis()
124
125     plt.subplot(232)
126     plt.imshow(alpha_(alphax), cmap='Reds')
127     plt.title(r'$\alpha$')
128     plt.grid()
129     plt.colorbar()
130     plt.gca().invert_yaxis()
131
132     plt.subplot(233)
133     plt.imshow(sigma0_(alphax), cmap='Reds')
134     plt.title(r'$\sigma_0$')
135     plt.grid()
136     plt.colorbar()
137     plt.gca().invert_yaxis()
138
139     plt.subplot(234)
140     plt.imshow(S01x, cmap='Reds')
141     plt.title(r'$S_{01x}$')
142     plt.grid()
143     plt.colorbar()
144     plt.gca().invert_yaxis()
145
146     plt.subplot(235)
147     plt.imshow(S02x, cmap='Reds')
148     plt.title(r'$S_{02x}$')
149     plt.grid()
150     plt.colorbar()
151     plt.gca().invert_yaxis()
152
153     plt.show()
```

# Bibliografía

---

- [1] et al. (eds.) Bass M. *OSA Handbook of Optics*, volume Volume 1. MGH, 1994. [1](#)
- [2] G. Landsberg. *Optica I*. Mir, 1976. [2](#)
- [3] Eugene Hecht. *Optics*. Pearson Education, 5; global edition, 2016. [3](#), [8](#)
- [4] Emil Wolf. *Introduction to the Theory of Coherence and Polarization of Light*. Cambridge University Press, 2007. [9](#)
- [5] Edward Collet. A description of polarization in classical physics. *American Journal of Physics*, 36, 1968. [9](#)
- [6] Edward Collett. *Field guide to polarization*. Spie Press, New York, 1991. [12](#)
- [7] Sharma. *Optics. Principles and Applications*. Elsevier, 2006. [12](#)
- [8] Gildardo Pablo Lemus Alonso. *Luz Polarizada mediante modulación de amplitud en no-cuadratura*. PhD thesis, Benemérita Universidad Autónoma de Puebla, Facultad de ciencias físico matemáticas, 1994. [13](#)
- [9] Hans Petter Langtangen. *Python Scripting For Computational Science*. Texts in Computational Science and Engineering. Springer Berlin Heidelberg, 2009. [27](#)
- [10] Fabio Nelli (auth.). *Python Data Analytics: Data Analysis and Science Using Pandas, matplotlib, and the Python Programming Language*. Apress, 2015. [27](#)