

**Centro de Investigación Científica y de
Educación Superior de Ensenada**



**TECNICA PARA LA MEDICION DEL INDICE
DE REFRACCION NO LINEAL DE UN MATERIAL
CON UN SOLO PULSO**

**TESIS
MAESTRIA EN CIENCIAS**

ALEXANDRO RUIZ DE LA CRUZ

Ensenada, B. Cfa. Mexico.

Febrero de 2000.

**CENTRO DE INVESTIGACIÓN CIENTÍFICA Y DE
EDUCACIÓN SUPERIOR DE ENSENADA**



CICESE

**DIVISIÓN DE FÍSICA APLICADA
DEPARTAMENTO DE ÓPTICA**

**TÉCNICA PARA LA MEDICIÓN DEL ÍNDICE DE
REFRACCIÓN NO LINEAL DE UN MATERIAL CON UN
SOLO PULSO**

TESIS

que para cubrir parcialmente los requisitos necesarios para obtener el grado de
Maestro en Ciencias presenta:

Alexandro Ruiz de la Cruz

Ensenada, Baja California, México, Febrero del 2000

Resumen de la tesis de **ALEXANDRO RUIZ DE LA CRUZ**, presentada como requisito parcial para la obtención del grado de **MAESTRO EN CIENCIAS EN ÓPTICA CON ESPECIALIDAD EN OPTOELECTRÓNICA**. Ensenada Baja California, México
Noviembre de 1999.

TÉCNICA PARA MEDIR EL ÍNDICE DE REFRACCIÓN NO LINEAL DE UN MATERIAL CON UN SOLO PULSO.

Resumen aprobado por:

Mehrdad Mohebi

Dr. Mehrdad Mohebi
Director de tesis

Se desarrolló una nueva técnica para medir el índice de refracción no lineal de un material que utiliza un solo pulso. Para esta técnica se proponen dos arreglos experimentales que basan su funcionamiento en la técnica conocida como z-scan. El primero de estos arreglos utiliza dos muestras no lineales, donde la función de una de ellas es la de hacer un barrido de la cintura del haz por la segunda muestra, haciendo uso del autoenfocamiento provocado por la intensidad del haz en la primer muestra. Obteniéndose un cambio en la transmitancia de una abertura colocada frente a la segunda muestra.

El segundo arreglo que se propuso solo utiliza una sola muestra no lineal y básicamente es igual al arreglo del z-scan. La diferencia está en que para nuestro arreglo utilizamos detectores rápidos que pueden resolver el pulso que se propaga a través del arreglo. La muestra no lineal se mantiene en una sola posición, mientras el perfil de intensidad del pulso de entrada causa diferentes condiciones de autoenfocamiento en la muestra, provocando cambios en la transmitancia de una abertura colocada frente a la muestra.

También se desarrolló un programa para la simulación numérica de los resultados que se obtendrían en el laboratorio. Para esta simulación se utilizó teoría de propagación gaussiana, de difracción y de óptica no lineal de tercer orden. En la simulación se observó muy buena sensibilidad por parte de los arreglos propuestos.

En el laboratorio se implementó el arreglo que solo utiliza una muestra no lineal. Como muestra se utilizó una celda llena con CS_2 cuyo índice de refracción no lineal es bien conocido en la literatura ($n_2 = 1.2E - 11$ esu).

Finalmente se muestran los resultados obtenidos en el laboratorio y el ajuste teórico obtenido para el valor correspondiente de n_2 del material. Los resultados obtenidos mostraron estar en buen acuerdo con la teoría.

Palabras clave: autoenfocamiento, índice de refracción no lineal, transmitancia.

A SINGLE-SHOT TECHNIQUE FOR MEASUREMENT OF THE NONLINEAR REFRACTIVE INDEX OF MATERIALS.

ABSTRACT

A technique to measure the nonlinear refractive index of optical materials using a single pulse was developed. Two different experimental setups were proposed for this technique. The first setup we proposed uses two nonlinear samples. The function of one of these samples is to scan the other nonlinear sample with the beam focus using the self-focusing of the beam in the first sample. This self-focusing follows the input pulse intensity profile. We obtain a variation in the transmission of an aperture located in the far field in front of the second sample.

The second setup only uses one nonlinear sample. Basically it is the same z-scan setup but we use fast detectors so that we can resolve the input pulse. The nonlinear sample is in a fixed position while the pulse profile causes the self-focusing condition in the sample to change with time. With this we obtain a time dependent variation in the transmission of the aperture or disc located in the far field in front of the detector.

A numerical simulation of both setups was developed to compare theoretical and experimental results. Here we used gaussian propagation theory, diffraction theory and nonlinear optics theory.

The setup with one nonlinear sample was implemented in the laboratory. The sample was a cell filled with CS_2 with a nonlinear refractive index $n_2 = 1.2E - 11 \text{ esu}$. Finally we show the results obtained in the laboratory and its theoretical fit with for different energies of the input pulse.

Keywords: Nonlinear refractive index, self-focusing, single pulse.

A mis padres

AGRADECIMIENTOS

A toda mi familia, abuelas y abuelos, tias y tios, primas y primos, sobrinos no porque todavía no tengo, les agradezco el apoyo brindado a lo largo de todo este tiempo.

A mi novia Viole, muchisimas gracias por esperar.

A Miguel Ángel técnico del laboratorio por su valiosa ayuda con el programa.

A todos mis cuates que no los nombro porque seguro se me olvida alguno.

A todos los miembros del comite por su apreciable ayuda.

Al CONACYT por la beca. Numero de becario 118438.

TABLA DE CONTENIDO

CAPITULO I	1
INTRODUCCIÓN	1
I.1 TÉCNICAS PARA MEDIR n_2	1
I.2 OBJETIVO	4
I.3 DESCRIPCIÓN GENERAL	4
I.4 ARREGLO CON UNA SOLA MUESTRA NO LINEAL.	6
CAPITULO II	8
TEORÍA	8
II.1 DESCRIPCIÓN DEL FUNCIONAMIENTO DEL ARREGLO PROPUESTO	8
II.2 CÁLCULO DE LA PROPAGACIÓN DEL HAZ	10
II.2.1 TEORÍA DE DIFRACCIÓN	11
II.2.1.1 Aproximación de Fresnel	11
II.2.1.2 Aproximación de Rayleigh-Sommerfeld	13
II.2.2 ÓPTICA NO LINEAL DE TERCER ORDEN.	15
II.2.2.1 Muestras delgadas y muestras gruesas	16
II.3 RESULTADOS DE LA SIMULACIÓN NUMÉRICA DEL F-SCAN	17

II.4 ARREGLO ALTERNATIVO CON UNA SOLA MUESTRA NO LINEAL	21
II.4.1 FUNCIONAMIENTO DEL ARREGLO ALTERNATIVO	22
II.4.2 SIMULACIÓN DEL ARREGLO CON UNA SOLA MUESTRA NO LINEAL	23
CAPITULO III	26
RESULTADOS EXPERIMENTALES	26
III.1 MONTAJE EXPERIMENTAL	26
III.2 PERFIL DEL HAZ	27
III.3 RESULTADOS EXPERIMENTALES	30
III.3.1 MEDICIÓN CON BAJA ENERGÍA	30
III.3.2 MEDICIÓN CON ALTA ENERGÍA	32
III.3.3 PROCESAMIENTO DE LOS DATOS	33
III.3.4 AJUSTE TEÓRICO	37
CAPITULO IV	41
CONCLUSIONES	41
LITERATURA CITADA	45
APÉNDICE A (TEORÍA DE ÓPTICA NO LINEAL DE TERCER ORDEN)	46

<u>APÉNDICE B (TEORÍA DE DIFRACCIÓN).</u>	49
APROXIMACIÓN DE FRESNEL	50
APROXIMACIÓN DE RAYLEIGH-SOMMERFELD	51
<u>APÉNDICE C (PROPAGACIÓN GAUSIANA)</u>	52
<u>APÉNDICE D (ALGORITMO DE LA TRANSFORMADA RÁPIDA DE HANKEL)</u>	56
<u>APÉNDICE E (DESCRIPCIÓN DEL PROGRAMA)</u>	58

LISTA DE FIGURAS

Figura	Título	Página
1	Arreglo experimental para el z-scan	3
2	Arreglo propuesto	5
3	Arreglo experimental del arreglo con una sola muestra no lineal	6
4	Desplazamiento de la cintura del haz después de L2	17
5	Resultado de la simulación del F-SCAN	19
6	Gráficas de la transmitancia pico como función de algunos parámetros del arreglo del F-SCAN. a) como función de la distancia entre L1 y L2, b) como función de la longitud focal de L2 y c) como función de la posición de MNL1	20
7	Esquema del arreglo alternativo al F-SCAN	22
8	Simulación del arreglo alternativo al F-SCAN	24
9	Montaje experimental	26
10	Perfil del haz	28
11	Ajuste de los datos de la propagación gaussiana	29
12	Medición de referencia	31
13	Perfil del pulso en D1 y D2 con alta energía	33
14	Procesamiento de los datos: a) cociente de los detectores D1 y D2 para baja energía, b) cociente de los detectores D1 y D2 para alta energía, c) cociente de los incisos a y b	34
15	Segundo formato para presentar los datos	36
16	Ajuste de los datos experimentales	38
17	Datos experimentales y ajuste con los datos de la Tabla IV	39
18	Datos experimentales y ajuste con los datos de la Tabla V	40

LISTA DE TABLAS

Tabla	Título	Página
I	Parámetros de la simulación del F-SCAN	18
II	Condiciones para la simulación	24
III	Condiciones para el ajuste	37
IV	Datos del ajuste de la Figura 17	39
V	Datos para el ajuste de la Figura 18	40

Técnica para la medición del índice de refracción no lineal de un material con un solo pulso.

CAPITULO I

Introducción

Anteriormente se pensaba que el índice de refracción de un material era un valor constante que dependía solamente de la longitud de onda de la luz incidente en el material. A partir del desarrollo del láser, la intensidad de la luz incidente se incrementó enormemente. Con esto se pudieron observar una gran variedad de fenómenos no lineales en los materiales. Uno de estos efectos no lineales es que el índice de refracción para una longitud de onda dada varía con la intensidad del haz incidente. Para este caso el índice de refracción del material está dado por $n = n_0 + n_2 I$ [Boyd, 1992]. Donde n_0 es el índice de refracción lineal, y n_2 es el índice de refracción no lineal.

Esta propiedad es muy importante, ya que muchos dispositivos optoelectrónicos basan su funcionamiento en esta propiedad. Entre las aplicaciones mas importantes se encuentran, por ejemplo, moduladores para comunicaciones por fibra óptica, conmutación óptica, limitación óptica, conjugación de fase, etc. Por esto, uno de los parámetros mas importantes al caracterizar un material es el índice de refracción no lineal.

1.1 Técnicas para medir n_2

Para medir el parámetro n_2 se han ideado una gran variedad de técnicas. Entre ellas están, la interferometría no lineal, el mezclado degenerado de cuatro ondas, el “z-scan” (barrido en z), y sus variantes (ez-scan e iez-scan), etc. Enseguida se describen brevemente algunas de estas técnicas.

Mezclado degenerado de cuatro ondas (conjugación de fase): En esta técnica se genera un patrón de interferencia sobre la muestra provocando una modulación del índice de refracción del material, que sigue las variaciones en la intensidad del patrón de interferencia. Esta modulación del índice de refracción se puede ver también como una rejilla de difracción, de esta manera lo que se mide es la eficiencia con que se difracta otro haz al pasar por esta rejilla. Con esta técnica se puede obtener información sobre la magnitud de n_2 pero no de su signo. Para este método se necesita un arreglo muy complicado, donde se deben hacer coincidir espacial y temporalmente tres haces, haciéndolo muy difícil de implementar en el laboratorio, dando lugar a errores experimentales grandes. Además, con esta técnica solo podemos obtener información acerca de la magnitud de $\chi^{(3)}$. Tiene la ventaja de tener una gran sensibilidad.

Interferometría no lineal: Para esta técnica se utiliza un interferómetro, en el cual se coloca en uno de sus brazos un material no lineal. Se observa el patrón de interferencia formado y se miden los corrimientos provocados por la propagación del haz a través de la muestra no lineal en el patrón de interferencia al aplicar haces de diferente intensidad. De estos corrimientos se deduce la magnitud y el signo de n_2 después una serie de elaborados cálculos.

Z-SCAN: Esta técnica fue desarrollada en 1989 por Sheik-Bahae *et al.* y se ha convertido en una de las más utilizadas. El arreglo de este método se muestra en la Figura 1. Aquí un haz es enfocado por la lente L1, y el material no lineal se hace pasar por el foco de L1. A continuación se mide en D2 la transmitancia de la abertura como función de la posición de la muestra respecto al foco de L1. En la Figura 1 también se muestra la gráfica de transmitancia contra posición de la muestra, que se obtendría en un típico barrido en z. De esta técnica se puede obtener la magnitud y el signo de n_2 , así como el coeficiente de absorción no lineal β . Para el caso en el que la absorción no lineal del material es despreciable, podemos obtener el valor de n_2 simplemente de la diferencia entre el pico y el valle en la gráfica de la transmitancia contra posición de la muestra. En el caso en el que existe absorción no lineal, se debe obtener primero una gráfica sin utilizar abertura. Ofrece una gran simplicidad ya que solo se utiliza un haz haciéndola experimentalmente muy sencilla.

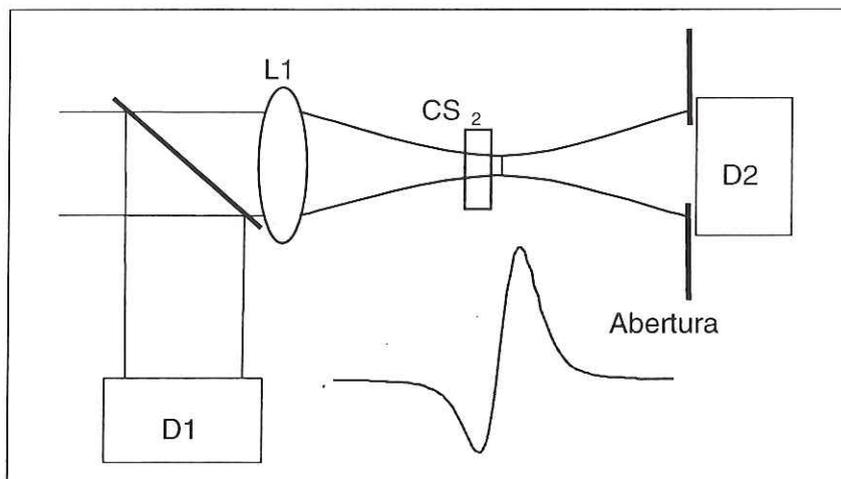


Figura 1. Arreglo experimental para el z-scan

EZ-SCAN: Esta variante del z-scan presenta mayor sensibilidad. En esta la diferencia con el z-scan es que se utiliza un disco que bloquea gran parte del haz en vez de una abertura. Esto provoca que la señal de fondo sea muy pequeña. De esta manera cualquier pequeña variación en la transmitancia del disco podrá ser detectado, ya que bloqueando casi todo el haz excepto sus extremos, la cantidad de energía que pasaría al detector será menor, de esta manera cualquier incremento o decremento en la energía será más contrastante. Esta técnica aún conserva la sencillez del z-scan y los cálculos numéricos son los mismos. La manera de obtener el valor de n_2 y de β es similar al z-scan.

1.2 Objetivo

El objetivo de esta tesis es desarrollar una técnica capaz de medir el índice de refracción no lineal de un material utilizando un solo pulso.

El desarrollo de la técnica comprende proponer un arreglo experimental, desarrollar la teoría para recrear los resultados experimentales y llevar a cabo algunas mediciones de n_2 para mostrar la funcionalidad del método.

1.3 Descripción general

El arreglo experimental que se propone se muestra en la Figura 2. En esta sección se hará una breve descripción del mismo. En el capítulo II se explica más detalladamente su funcionamiento.

En este arreglo se utilizan dos materiales no lineales, el primero es un material ya caracterizado al cual se le conoce el valor de n_2 , el segundo es el material que se desea caracterizar.

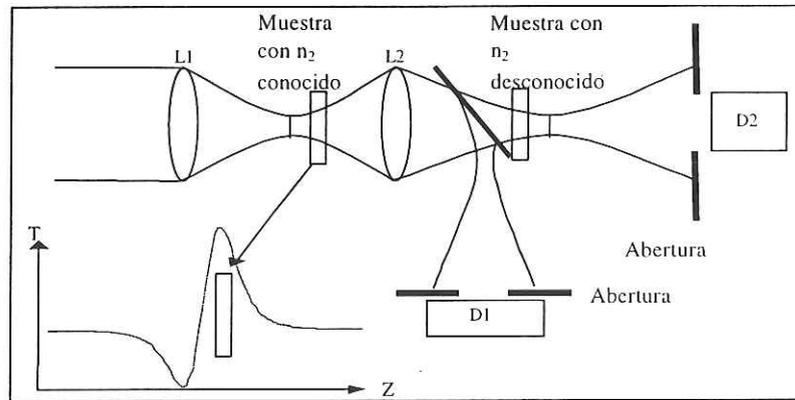


Figura 2. Arreglo propuesto

La manera en que funciona este arreglo es la siguiente: un pulso se propaga a través del arreglo, al principio del pulso, la intensidad del haz en la primera muestra es muy pequeña, y el haz se propagará sin ninguna distorsión en su propagación hasta la abertura colocada frente a D2. Siguiendo el perfil del pulso, la intensidad en la primera muestra irá aumentando, y llegará un punto en el que la intensidad será suficiente para modificar el índice de refracción de esta. Si suponemos que el índice de refracción no lineal de la primera muestra es positivo, y que el perfil, del haz es gaussiano, entonces el haz tenderá a enfocarse (autoenfocamiento). Este autoenfocamiento cambiará la posición del plano focal de la segunda lente, acercándose a esta. De aquí en adelante la posición de la cintura se acercará a la segunda lente y luego volverá a su lugar original, siguiendo el perfil del pulso. De esta manera se logra hacer un barrido del foco sobre la segunda muestra sin moverla. De igual forma que en el z-scan, se mide la variación de la transmitancia de la abertura frente a

D2, obteniendo una gráfica de transmitancia contra tiempo. Por esto el nombre más apropiado para esta técnica es el de F-SCAN, ya que lo que se mueve es la posición del foco a través de la muestra.

Para poder obtener el valor de n_2 , se hace la simulación del arreglo introduciendo todas las condiciones experimentales, teniendo como única variable n_2 .

Las ventajas que presenta esta técnica son por ejemplo, se minimizan los errores por variaciones en la intensidad entre pulso y pulso, ya que solo se utiliza un pulso para la medición, el arreglo es bastante sencillo y que la sensibilidad mostrada en las simulaciones numéricas mostró ser muy buena, entre otras.

1.4 Arreglo con una sola muestra no lineal.

También se propone otro arreglo que solo utiliza una sola muestra no lineal (Figura 3). Este arreglo al requerir de menos elementos, resulta mas sencillo de simular e implementar en el laboratorio. Debido a estas ventajas, será este arreglo con el que se obtendrán los resultados experimentales presentados en este trabajo. A continuación se explica brevemente su funcionamiento.

De la figura 3, vemos que el arreglo es básicamente el mismo que se utiliza para el Z-SCAN (figura 1). La diferencia está en que los detectores utilizados aquí son detectores rápidos. El arreglo está formado por un láser pulsado, un divisor de haz, un detector de referencia (D1), una lente (L1) para enfocar el haz, la muestra no lineal, un disco o abertura para obstruir parte del haz y el detector que registrará los cambios en la transmitancia del disco o abertura (D2).

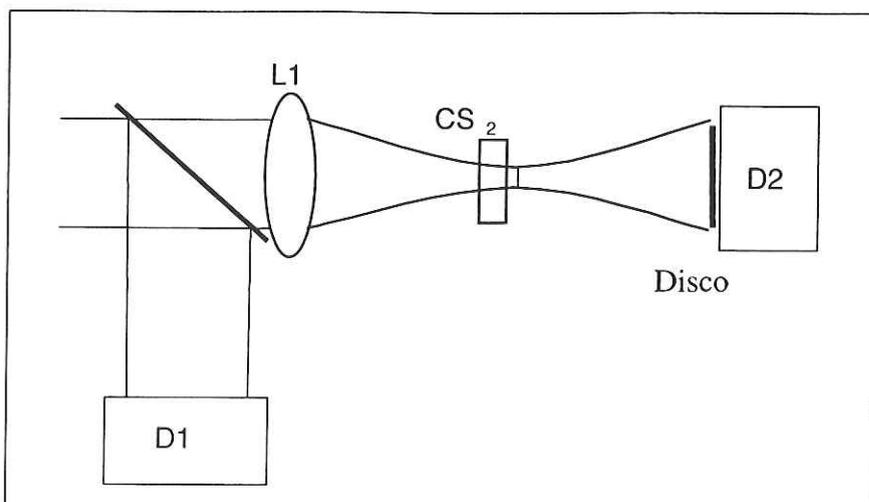


Figura 3 . Arreglo experimental del arreglo con una sola muestra no lineal

De acuerdo a la figura 3, el pulso se propaga a través de la lente y se enfoca muy cerca de la muestra no lineal. Consideramos que la muestra se encuentra antes de la cintura del haz y que su índice de refracción no lineal es positivo. Al principio del pulso la intensidad será mínima y no causará ningún efecto no lineal en la muestra. De esta manera el haz se propagará hasta la abertura o disco de forma gaussiana. Cuando la intensidad del pulso en la muestra sea suficientemente alta, se producirá un autoenfocamiento del haz. Esto provocará mayor divergencia en la propagación del haz después de la cintura. Al llegar al disco frente a D2, el tamaño del haz será mayor, aumentando de esta manera la transmitancia del disco. Así la transmitancia del disco cambiará siguiendo al perfil de intensidad del pulso, debido al autoenfocamiento del haz en la muestra no lineal.

CAPITULO II

Teoría

En este capítulo se describe la teoría utilizada para realizar la simulación numérica del arreglo aquí propuesto (figura 2). La teoría necesaria para calcular la propagación del pulso a través del arreglo es muy variada. Esta teoría comprende la propagación gaussiana, la difracción utilizando las aproximaciones de Fresnel y de Rayleigh-Sommerfeld y la teoría de óptica no lineal de tercer orden.

También se propone un segundo arreglo experimental que sólo utiliza una muestra no lineal. Finalmente se presentan los resultados teóricos que se obtuvieron para ambos arreglos.

II.1 Descripción del funcionamiento del arreglo propuesto

El arreglo propuesto en este trabajo ya se presentó en la Figura 2 del capítulo anterior. A continuación se da una explicación detallada de la propagación de un pulso a través de este arreglo.

Un pulso se propaga a través del arreglo hasta la abertura frente al detector D2. Suponemos que el haz tiene perfil gaussiano y que la primera muestra no lineal (MNL1) tiene $n_2 > 0$, y ni la magnitud ni el signo de n_2 en la segunda muestra no lineal (MNL2) son conocidos. La posición de MNL1 es como se muestra en la figura 2 después del foco de

L1. También se muestra la posición de MNL1 respecto al foco en lo que sería la traza normal de un barrido en z típico.

Suponiendo un pulso con perfil temporal gaussiano, al principio del pulso la intensidad del haz en la primera muestra será muy pequeña y no afectará al índice de refracción del material. Sin variaciones en el índice de refracción de la muestra, el haz seguirá su propagación hasta la abertura frente a D2, siguiendo un comportamiento gaussiano. De acuerdo al perfil del pulso, la intensidad del haz en la primer muestra aumentará, y, en algún punto, será suficiente para cambiar el índice de refracción del material significativamente. Como el índice de refracción del material está dado por

$$n = n_0 + n_2 I , \quad (1)$$

y de acuerdo al perfil gaussiano del haz, el índice de refracción en el centro será mayor e irá decreciendo hacia los extremos del haz. De esta manera, podemos ver a la muestra como una lente positiva, y el haz tenderá a enfocarse. Esto es lo que se conoce como autoenfocamiento.

Cuando el haz llegue a L2 su divergencia será menor que antes, debido al autoenfocamiento en la MNL1. Así la posición del foco de L2 cambiará, quedando un poco mas cerca de L2. Desde este punto en el pulso y hasta el pico del mismo, la intensidad del haz en MNL1 irá en aumento. Siguiendo con la propagación de la manera en que se ha explicado, la posición del foco de L2 seguirá acercándose pasando por MNL2, haciendo de esta manera un barrido de MNL2.

Lo que sucede en MNL2 es lo mismo que sucede en MNL1, pero el cambio en la intensidad del haz en MNL2 es más drástico porque así como el pulso incrementa su intensidad, el foco se acerca a la muestra y por el menor tamaño del haz la intensidad también aumenta.

D1 es el detector de referencia, y registra los cambios en la propagación del haz producidos por MNL1. La distancia de D1 a L2 es la misma que la de L2 a D2. Las aberturas también son iguales, de esta manera podremos utilizar D1 como referencia.

En este arreglo también podemos aplicar las variaciones del z-scan, como el ez-scan y el iez-scan, incrementándose con ello la sensibilidad. De hecho, en la simulación numérica calculamos los resultados para un disco y para una abertura frente a los detectores, observándose mayor sensibilidad cuando se utiliza un disco.

II.2 Cálculo de la propagación del haz

A continuación se presentan las distintas aproximaciones utilizadas en este trabajo para la propagación del haz a través del arreglo. Todas las expresiones que aquí se muestran provienen de distintas aproximaciones al principio de Huygens-Fresnel y de manipulaciones de las ecuaciones de Maxwell. Estas aproximaciones se muestran en detalle en los apéndices al final de este trabajo.

Aunque la teoría que se utiliza para la simulación numérica es muy general y permite la propagación de cualquier tipo de haz, nos limitamos a propagar haces gaussianos con simetría radial para simplificar algunos cálculos.

II.2.1 Teoría de difracción

La teoría de difracción nos permite calcular la propagación de un campo en cualquier medio lineal. En este trabajo la utilizaremos para calcular la propagación del haz en el espacio libre entre los distintos elementos del arreglo, y, bajo ciertas condiciones, para la propagación del haz dentro del material no lineal. Para este propósito, utilizaremos las aproximaciones de Fresnel y de Rayleigh-Sommerfeld (ver apéndice B).

II.2.1.1 Aproximación de Fresnel

La ecuación que se obtiene bajo la aproximación de Fresnel, que representa el campo $U(\xi, \eta)$ propagado una distancia z es la siguiente

$$U(x, y) = \frac{e^{jkz}}{j\lambda z} e^{\frac{jk}{2z}(x^2+y^2)} \int \int_{-\infty}^{\infty} \left[U(\xi, \eta) e^{\frac{jk}{2z}(\xi^2+\eta^2)} \right] e^{-\frac{j2\pi}{\lambda z}(x\xi+y\eta)} d\xi d\eta, \quad (3)$$

donde $U(x, y)$ es el campo después de propagarlo, k es el número de onda ($\frac{2\pi}{\lambda}$), λ es la longitud de onda, z es la distancia a propagar y $U(\xi, \eta)$ es el campo que deseamos propagar.

El desarrollo necesario para llegar a esta ecuación se muestra en el apéndice B. Esta ecuación también la podemos ver como la transformada exacta de Fourier, con algunos factores multiplicativos, del campo que queremos propagar $U(\xi, \eta)$, multiplicado por el factor de fase exponencial $e^{\frac{jk}{2z}(\xi^2+\eta^2)}$.

Para poder utilizar esta aproximación se debe cumplir la siguiente condición [Goodman, 1988]

$$z^3 \gg \frac{\pi}{4\lambda} \left[(x-\xi)^2 + (y-\eta)^2 \right]_{max}, \quad (2)$$

donde z es la distancia a propagar, λ la longitud de onda, (x, y) son las coordenadas en el plano del campo ya propagado y (ξ, η) son las coordenadas en el plano del campo inicial. En el arreglo que se propone, esta condición se cumple sin ningún problema para la distancia entre los elementos del arreglo.

Si asumimos que el haz tiene simetría radial podemos simplificar aún más el cálculo de la ecuación (3), transformándola de una ecuación en dos dimensiones a una ecuación en una sola dimensión. Esto se hace de la siguiente manera ver [Gaskill, 1978]:

El campo que queremos propagar debe ser una función circularmente simétrica, es decir

$$U(\xi, \eta) = U(\sqrt{\xi^2 + \eta^2}), \quad (4)$$

sustituyendo esto en la ecuación (3) obtenemos

$$U(x, y) = \frac{e^{jkz}}{j\lambda z} e^{\frac{jk}{2z}(x^2+y^2)} \int \int_{-\infty}^{\infty} \left[U(\sqrt{\xi^2 + \eta^2}) e^{\frac{jk}{2z}(\xi^2+\eta^2)} \right] e^{-\frac{j2\pi}{\lambda z}(x\xi+y\eta)} d\xi d\eta. \quad (5)$$

Si ahora escribimos esta última ecuación en coordenadas circulares donde

$$x = r \cos \theta, \quad r = \sqrt{x^2 + y^2},$$

$$y = r \sen \theta, \quad \theta = \tan^{-1} \left(\frac{x}{y} \right),$$

$$\xi = \rho \cos \phi, \quad \rho = \sqrt{\xi^2 + \eta^2},$$

$$\eta = \rho \sen \phi, \quad \phi = \tan^{-1} \left(\frac{\eta}{\xi} \right).$$

De esta manera la expresión para $U(x, y)$ queda:

$$U(r \cos \theta, r \sin \theta) = \frac{e^{jkz}}{j\lambda z} e^{\frac{jk}{2z}(r^2)} \int_0^\infty \int_0^{2\pi} \left[U(\rho) e^{\frac{jk}{2z}(\rho^2)} \right] e^{-\frac{j2\pi}{\lambda z} r \rho \cos(\theta - \phi)} \rho d\rho d\phi, \quad (6)$$

donde se utilizó la identidad $\cos \theta \cos \phi + \sin \theta \sin \phi = \cos(\theta - \phi)$. Usando una de las identidades para las funciones Bessel obtenemos

$$U(r \cos \theta, r \sin \theta) = 2\pi \frac{e^{jkz}}{j\lambda z} e^{\frac{jk}{2z}(r^2)} \int_0^\infty \left[U(\rho) e^{\frac{jk}{2z}(\rho^2)} \right] J_0\left(\frac{2\pi}{\lambda z} r \rho\right) \rho d\rho. \quad (7)$$

En esta ecuación podemos notar que el lado derecho es independiente de θ , por lo que la podemos igualar a una función que solo dependa de r quedando entonces como

$$U(r) = U(r \cos \theta, r \sin \theta) = 2\pi \frac{e^{jkz}}{j\lambda z} e^{\frac{jk}{2z}(r^2)} \int_0^\infty \left[U(\rho) e^{\frac{jk}{2z}(\rho^2)} \right] J_0\left(\frac{2\pi}{\lambda z} r \rho\right) \rho d\rho. \quad (8)$$

Esto es lo que se conoce como una transformada de Hankel, que reduce nuestro problema a una sola dimensión. En el apéndice D se describe un algoritmo para calcular numéricamente la transformada de Hankel, conocido como transformada rápida de Hankel.

II.2.1.2 Aproximación de Rayleigh-Sommerfeld

Esta aproximación nos permitirá propagar un campo distancias muy pequeñas. La condición que se debe cumplir para utilizar esta aproximación es que la distancia a propagar debe ser mucho mayor que la longitud de onda [Gaskill, 1978]. Esta aproximación la utilizaremos en caso de que no podamos despreciar los cambios en el ancho del haz al propagarse dentro de la muestra no lineal. Este tema será explicado más adelante en la sección II.2.2.1.

La expresión que se obtiene en el apéndice B bajo la aproximación de Rayleigh-Sommerfeld queda de la siguiente forma

$$U(x, y) = \int \int_{-\infty}^{\infty} U(\xi, \eta) \frac{1}{j\lambda z} \frac{e^{jkz \left[1 + \left(\frac{x-\xi}{z} \right)^2 + \left(\frac{y-\eta}{z} \right)^2 \right]^{1/2}}}{\left[1 + \left(\frac{x-\xi}{z} \right)^2 + \left(\frac{y-\eta}{z} \right)^2 \right]} d\xi d\eta, \quad (9)$$

donde las variables son las mismas que ya se definieron para la aproximación de Fresnel. Esta expresión también la podemos simplificar si la vemos como una convolución del campo que vamos a propagar, $U(\xi, \eta)$ con la función de respuesta al impulso

$$h(x, y, \xi, \eta) = \frac{1}{j\lambda z} \frac{e^{jkz \left[1 + \left(\frac{x-\xi}{z} \right)^2 + \left(\frac{y-\eta}{z} \right)^2 \right]^{1/2}}}{\left[1 + \left(\frac{x-\xi}{z} \right)^2 + \left(\frac{y-\eta}{z} \right)^2 \right]}. \quad (10)$$

La expresión del campo ya propagado queda

$$U(x, y) = \int \int_{-\infty}^{\infty} U(\xi, \eta) h(x, y, \xi, \eta) d\xi d\eta. \quad (11)$$

Si le aplicamos la transformada de Fourier, y de acuerdo a la teoría que nos dice que la transformada de una convolución es la multiplicación de las transformadas de cada una de las funciones que se están convolucionando, entonces

$$\mathfrak{F}\{U(x, y)\} = \mathfrak{F}\{U(\xi, \eta)\} \mathfrak{F}\{h(\xi, \eta)\}. \quad (12)$$

El campo inicial $U(\xi, \eta)$ ya lo conocemos, y podemos utilizar la transformada de Hankel para simplificar el problema. La transformada de la función de respuesta al impulso se puede resolver analíticamente dando como resultado

$$\mathfrak{F}\{h(\xi, \eta)\} = e^{jkz \sqrt{1 - \lambda^2 (x^2 + y^2)}}. \quad (13)$$

Representando esta ecuación en coordenadas cilíndricas para simplificar obtenemos

$$H(r) = e^{jkz\sqrt{1-\lambda^2}(r^2)}. \quad (14)$$

Finalmente para encontrar el campo difractado, aplicamos la transformada de Fourier inversa.

II.2.2 Óptica no lineal de tercer orden.

Las dos aproximaciones presentadas pueden describir la propagación en medios lineales, homogéneos, isotrópicos y no magnéticos. Estas aproximaciones las podemos aplicar a la propagación en el espacio libre. Pero para la propagación en medios no lineales necesitamos otro tratamiento.

Este nuevo tratamiento de la propagación del haz, se muestra en el apéndice A. Principiando con las ecuaciones de Maxwell, se obtiene la ecuación de onda. Resolviéndose esta ecuación para una polarización no lineal de tercer orden, llegamos finalmente a una expresión para un cambio $\Delta\phi$ en la fase del campo, que depende de la distancia de propagación dentro del medio no lineal y de la intensidad del haz

$$\Delta\phi(r, L) = k_0 n_2 I(r) L. \quad (15)$$

En esta ecuación k_0 es el número de onda, n_2 es el índice de refracción no lineal del material, $I(r)$ es el perfil de intensidad del haz y L la longitud de la muestra.

Con esta expresión podemos calcular el cambio de fase producido por la intensidad del haz al propagarse este a través de la muestra no lineal. Para esto se añade este el cambio de fase de la ecuación 15 a la fase del campo propagado hasta la posición de la muestra.

II.2.2.1 Muestras delgadas y muestras gruesas

Esta es una consideración que se hace dependiendo de la longitud de la muestra tomando como referencia la distancia de Rayleigh del haz (z_0). La distancia de Rayleigh, es la longitud a partir de la cintura del haz en la cual el ancho del haz aumenta por un factor de $\sqrt{2}$. Si la longitud de la muestra es menor que este rango la podemos considerar delgada, en cambio si la longitud es mayor la muestra se considera gruesa.

Considerar que una muestra es delgada equivale a despreciar la variación del ancho del haz durante su propagación a través de esta, así como los cambios producidos por las variaciones en el índice de refracción del material. Para propagar el haz en una muestra delgada, solo tenemos que añadir al campo el cambio de fase provocado por la intensidad del haz y la longitud de la muestra (ecuación 15).

Cuando la muestra se considera gruesa, ya no podemos despreciar las variaciones del ancho del haz en la muestra debidos a la difracción natural del haz. En este caso el procedimiento para propagar el haz en la muestra es el siguiente. Se divide la longitud total de la muestra en secciones de un tamaño menor a la distancia de Rayleigh del haz. Se toma una sección y se propaga el haz la longitud de ésta utilizando la aproximación de Rayleigh-Sommerfeld explicada anteriormente. A continuación se utiliza la ecuación (15) para añadir los cambios de fase causados por el cambio en el índice de refracción del material que a su vez fueron causados por la intensidad del haz al propagarse la longitud de esa sección del material. De la misma manera se procede con todas las demás secciones del material, tomando como campo inicial el resultante de la propagación en sección anterior.

II.3 Resultados de la simulación numérica del F-SCAN

Ya que el funcionamiento de este arreglo depende de que la posición del foco después de L2 varíe con la intensidad del haz, a continuación se presentan los resultados teóricos para la posición de la cintura del haz, después de L2 y para diferentes intensidades del haz.

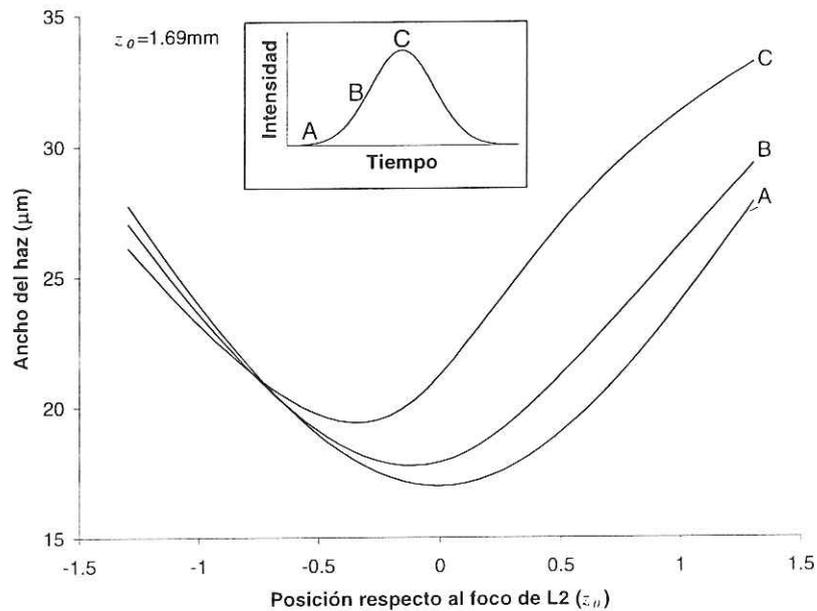


Figura 4. Desplazamiento de la cintura del haz después de L2.

En la Figura 4 se muestra el tamaño del haz desde $-1.5z_0$ antes del plano focal hasta $1.5z_0$ después del plano focal de L₂ del arreglo de la Figura 2. Cada una de las curvas representa el perfil de la propagación del haz para diferentes puntos en el tiempo de un pulso, es decir para diferentes intensidades. La curva centrada en 0 corresponde al perfil de propagación en el punto de mínima intensidad del pulso. Conforme aumenta la intensidad la posición del mínimo del ancho del haz (cintura) se desplaza hacia la lente L₂. Este

desplazamiento es el que nos va a permitir hacer un barrido de la segunda muestra. El desplazamiento es de algunos cientos de micras, lo cual es suficiente porque la intensidad del haz varía más en MNL2. Esto se debe a que además de la variación temporal de la intensidad del pulso, el ancho del haz también cambia con el tiempo como vemos en la Figura 4.

Tabla I Parámetros de la simulación del F-SCAN

Distancia de la cintura del haz a L1	20cm	Índice de refracción no lineal de MNL1	1.2×10^{-11} esu
Tamaño de la cintura del haz	0.2cm	Índice de refracción no lineal de MNL2	1.2×10^{-11} esu -1.2×10^{-11} esu
Longitud focal de L1 y L2	10 cm	Ancho del pulso FW1/eM I	4ns
Distancia entre L1 y L2	25 cm	Energía del pulso	50 μ J
Distancia entre MNL1 y L1	10cm	Distancia del disco/abertura a L2	40cm
Distancia entre MNL2 y L2	29.7cm	Diámetro del disco	0.2cm
Longitud de la muestra	0.1cm		

Ahora se mostrará el resultado de una simulación del F-SCAN para el caso donde se utiliza una abertura y para el caso de un disco. En la Tabla I están las condiciones de la simulación. Como material no lineal utilizamos CS₂ el valor de n_2 se obtuvo de [Sheik-Bahae, 1989].

En la figura 5, se presenta la transmitancia normalizada como función del tiempo, para una abertura y para un disco frente a D2 bajo las condiciones de la Tabla I. Cabe recordar que estos resultados se obtuvieron de la propagación de un solo pulso a través del arreglo.

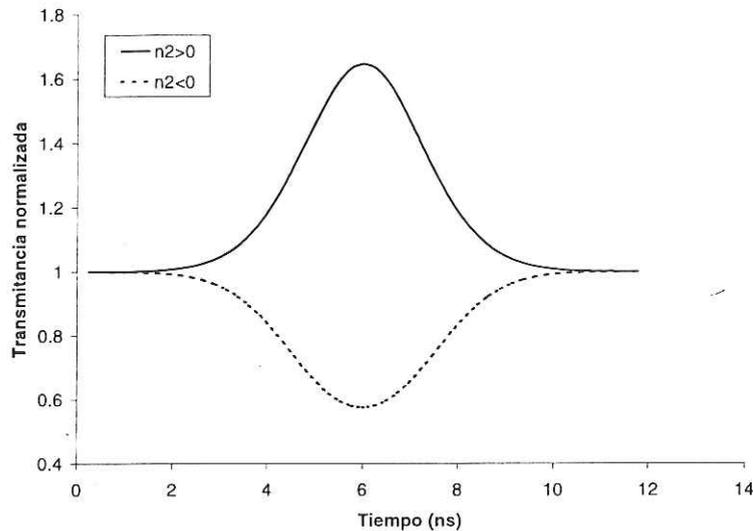


Figura 5 Resultado de la simulación del F-SCAN

La sensibilidad mostrada por este arreglo es similar a la mostrada por el z-scan y a la del ez-scan para la abertura y para el disco respectivamente. Pero esto puede optimizarse variando los diferentes parámetros del arreglo. Estos parámetros pueden ser la distancia focal de las lentes, la separación entre las lentes, la posición de las muestras, el tamaño de la abertura o el disco, etc.

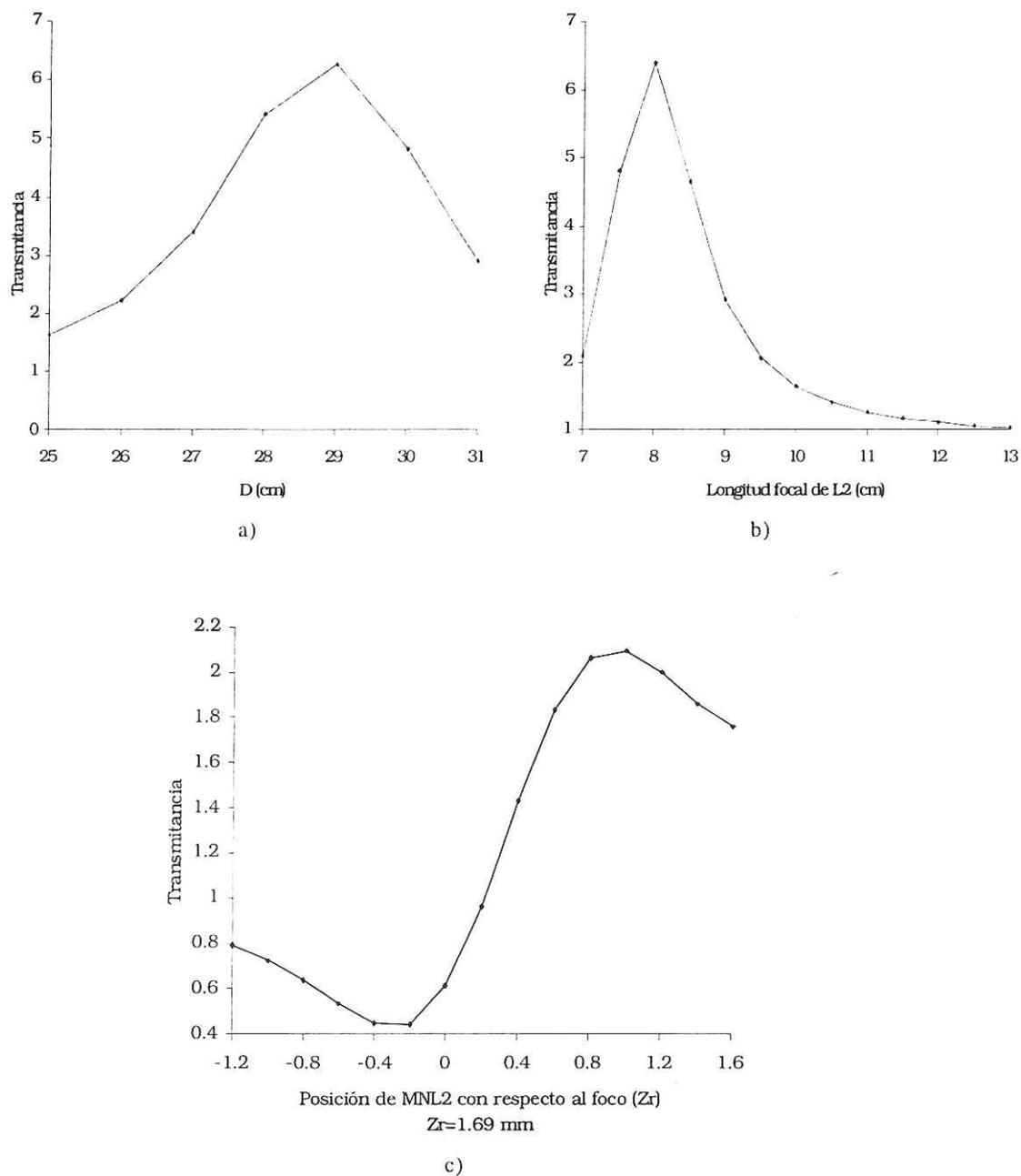


Figura 6 Gráficas de la transmitancia pico como función de algunos parámetros del arreglo del F-SCAN. a) como función de la distancia entre L1 y L2, b) como función de la longitud focal de L2 y c) como función de la posición de MNL1.

Estas pruebas para optimizar el arreglo se hicieron en la simulación y se presentan en la Figura 6. Allí se muestran las gráficas de la transmitancia pico contra los siguientes parámetros: separación entre L1 y L2 (distancia D), distancia focal de L2 y posición de MNL2. En la obtención de todos estos resultados se utilizó un disco frente a D2.

La transmitancia pico se toma como el valor donde está el máximo diferencial en la transmitancia obtenido de la simulación del arreglo. Por ejemplo, de la figura 5 el valor de la transmitancia pico para el disco es 1.55. Los valores de las gráficas de la Figura 6 se obtuvieron utilizando los parámetros de la Tabla I y variando el parámetro que se indica en cada gráfica. Para a) la variación fue en la distancia entre L1 y L2, para b) la variación fue en la distancia focal de L2 y finalmente para c) la variación fue la posición de MNL2 respecto a la cintura del haz después de L2.

II.4 Arreglo alternativo con una sola muestra no lineal

En el arreglo del F-SCAN utilizamos una muestra no lineal para mover la posición de la cintura del haz y de esta manera hacer un barrido de la muestra bajo estudio con un sólo pulso. Entonces también podemos con una sola muestra no lineal causar una variación en la transmitancia de una abertura o disco. Esto es posible porque como ya vimos, la propagación del haz se ve afectada de acuerdo al perfil de intensidad del pulso. En esta sección se presentará el esquema de este arreglo así como su funcionamiento y los resultados de la simulación numérica del mismo.

II.4.1 Funcionamiento del arreglo alternativo

La Figura 7 muestra el esquema del arreglo alternativo, donde sólo necesitamos una muestra no lineal. De igual manera que en el otro arreglo propagamos un pulso a través del arreglo. Este pulso al principio, no tendrá suficiente intensidad para afectar el índice de refracción del material y el haz se propagará hasta la abertura siguiendo un comportamiento gaussiano.

Siguiendo el perfil de intensidad del pulso de entrada, el haz incrementará su intensidad en la muestra. Cuando la intensidad sea suficiente para causar una variación significativa en el índice de refracción del material, la propagación del haz se verá afectada. La manera en que se propagará el haz después de pasar por la muestra no lineal, dependerá del signo del índice de refracción del material. Si el signo de n_2 es positivo el haz se autoenfocará, fenómeno que ya se explicó en la sección II.1.

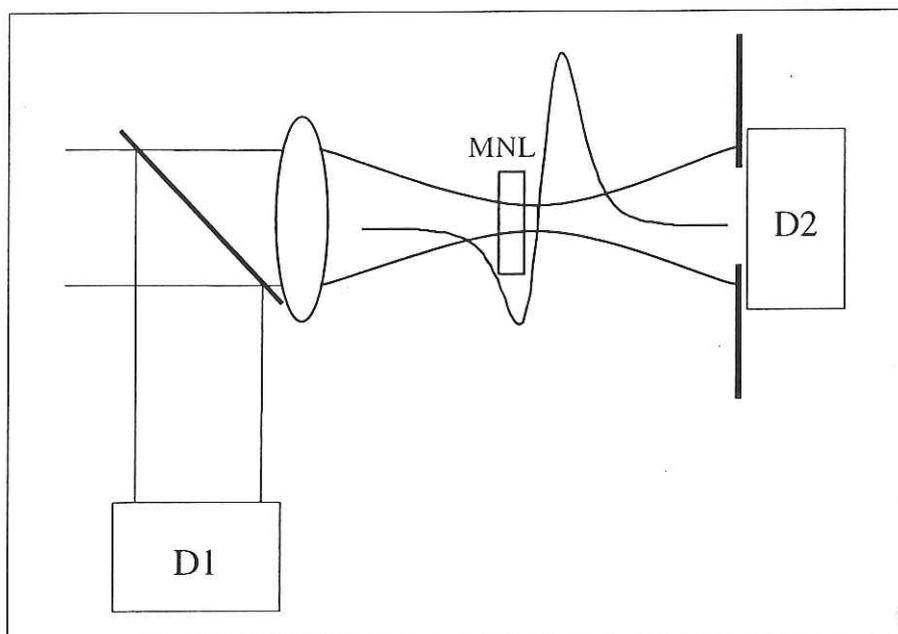


Figura 7 Esquema del arreglo alternativo al F-SCAN

Por la posición de la muestra no lineal antes de la cintura como se muestra en la Figura 7 y el autoenfocamiento, el haz se enfocará mas fuertemente. Al propagarse el haz después de la cintura, su divergencia será mayor, es decir el haz aumentará su diámetro más rápidamente conforme se propaga. Cuando el haz llegue al objeto colocado frente a D2, su transmitancia disminuirá o aumentará dependiendo de si se trata de una abertura o de un disco respectivamente. La intensidad del haz continuará aumentando y el haz seguirá autoenfocándose cada vez más fuertemente siguiendo el perfil de intensidad del pulso. Finalmente, estas variaciones en la transmitancia serán registradas en D2.

Este arreglo ofrece mayor simplicidad al propuesto anteriormente. La teoría involucrada es la misma, pero la simulación del arreglo se simplifica. Esto se debe a que disminuye el número de elementos por los que se tiene que propagar el haz, minimizando así la propagación de errores numéricos debido a las aproximaciones utilizadas.

II.4.2 Simulación del arreglo con una sola muestra no lineal

En la simulación de este arreglo se utilizaron las mismas rutinas ya desarrolladas para el F-SCAN. La Figura 5 muestra los resultados obtenidos para este arreglo utilizando un disco bajo las condiciones de la Tabla II.

En la Figura 8 está graficada la transmitancia normalizada del disco frente a D2, contra el tiempo. Estos datos se obtuvieron de la propagación de un solo pulso a través del arreglo.

Tabla II Condiciones para la simulación

Energía del pulso	25 μ J	Longitud de la muestra	1mm
Ancho temporal del pulso	7ns	Posición de la muestra respecto a la cintura	-0.85z ₀
Longitud de onda	532nm	Índice de refracción no lineal de la muestra	1.2 $\times 10^{-11}$ esu
Posición de la cintura respecto a L1	273.6cm	Longitud focal de L1	12.5cm
Tamaño de la cintura	164 μ m	Transmitancia del disco	5%

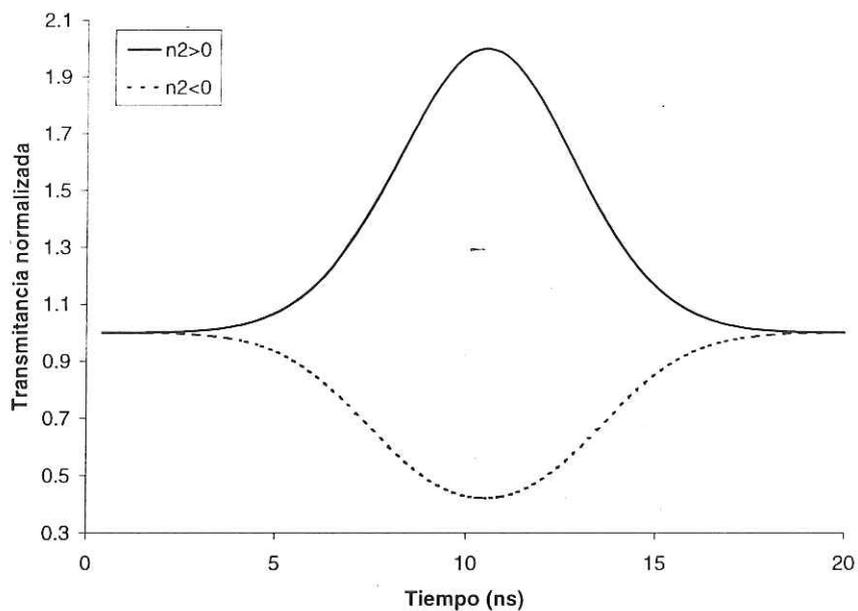


Figura 8 Simulación del arreglo alternativo al F-SCAN

La sensibilidad presentada por este arreglo es mayor que la mostrada por el F-SCAN y el ez-scan. Además el montaje experimental es más sencillo y la simulación no requiere de tantos cálculos como en el F-SCAN por el menor número de elementos en el arreglo. Debido a estas ventajas este arreglo será en el que nos enfocaremos en este trabajo y con el que se trabajará experimentalmente.

CAPITULO III

Resultados experimentales

En este capítulo se describe detalladamente el arreglo experimental de la Figura 9. En el capítulo anterior se explicaron las razones por las que se decidió trabajar con este arreglo.

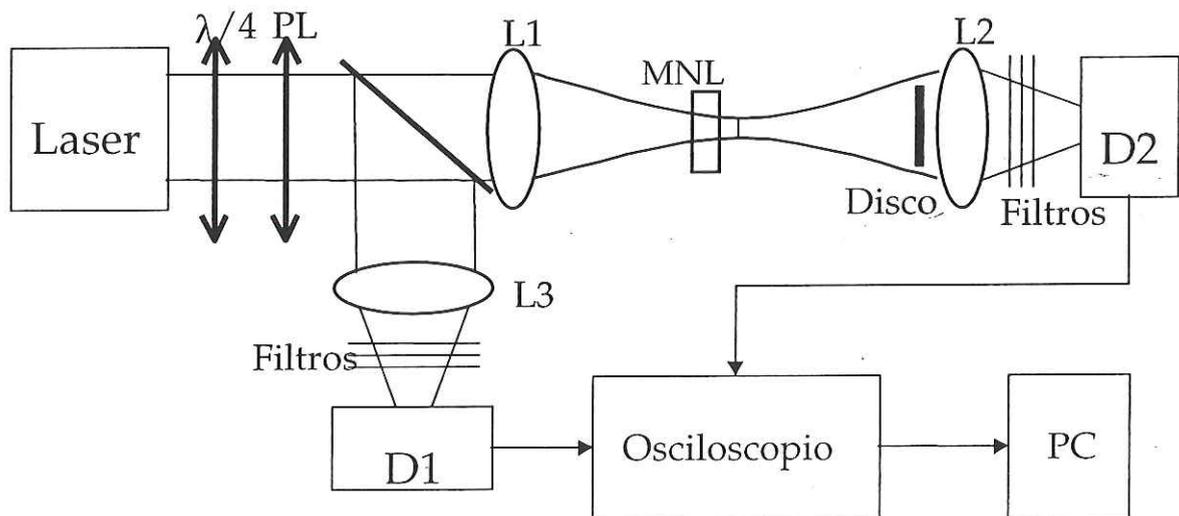


Figura 9. Montaje experimental

También se explica la manera en que se obtuvieron los datos experimentales, su procesamiento y finalmente como se hizo el ajuste teórico de estos resultados para obtener el valor de n_2 del material bajo estudio.

III.1 Montaje experimental

Para armar en el laboratorio el arreglo de la Figura 9, se utilizó un láser de Nd-YAG de frecuencia doblada a 532nm que emite pulsos de 7 ns. Como material no lineal (MNL) se

utilizó una celda de cuarzo de 1mm de longitud llena con CS_2 ($n_2 = 1.2 \times 10^{-11} \text{ esu}$) y se usó una lente (L1) con longitud focal de 12.5cm para enfocar el haz. También se usaron dos detectores rápidos de silicio (D1 y D2) con tiempo de subida y de bajada de 1ns. Debido a que el área activa de estos detectores es muy pequeña, hubo la necesidad de colocar lentes (L2 y L3) frente a estos detectores para enfocar el haz en ella. Para controlar la energía de los pulsos del láser se necesitó construir un atenuador. En este se utilizó una placa retardadora de $\lambda/4$ y un polarizador lineal (PL). La placa retardadora cambia la polarización del haz al girarla, y, el polarizador que está fijo selecciona solo una componente de polarización del campo que varía al girar la placa retardadora. Para no dañar a los detectores se utilizaron filtros de densidad neutrales atenuando de esta forma el haz.

También se usó un osciloscopio digital Tektronix modelo TDS 520A, con una velocidad máxima de adquisición de datos de 500MHz (toma un dato cada 2ns máximo), donde se observan los pulsos detectados en D1 y D2. Estos datos se graban en un archivo mediante una computadora conectada al osciloscopio.

En el laboratorio se utilizaron los dos canales del osciloscopio para tomar los datos de D1 y D2, lo que bajó la velocidad de toma de datos de este a la mitad (un dato cada 4ns).

III.2 Perfil del haz

La teoría utilizada en este trabajo es bastante general. La simulación numérica nos permite propagar cualquier tipo de perfil de haz, teniendo como única limitante que el haz tenga simetría radial.

Ya que la mayoría de los láseres tienen como salida un haz con perfil gaussiano, y además para simplificar los cálculos, en la simulación se supuso que el haz tenía perfil gaussiano. En el laboratorio se verificó que la salida del láser cumpliera con esto. Para esto se realizaron algunos barridos del perfil del haz en diferentes posiciones respecto a la salida del láser.

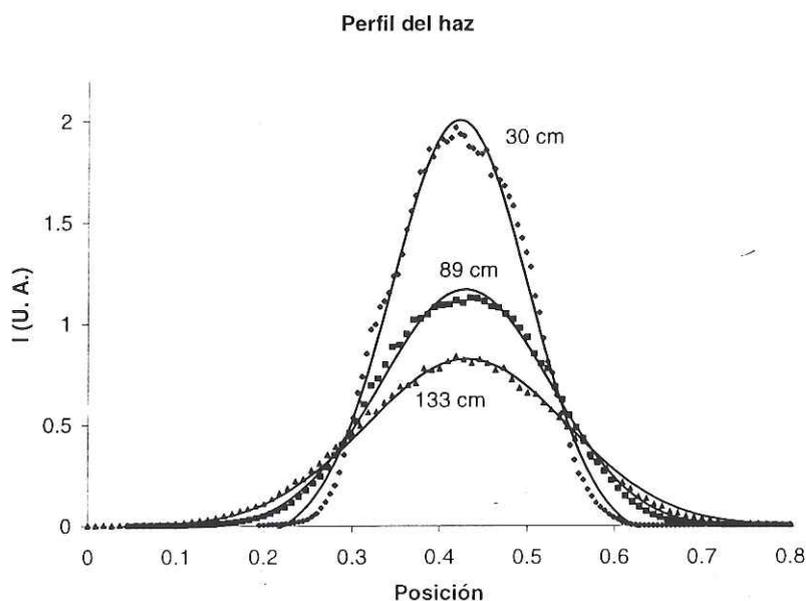


Figura 10 Perfil del haz

Los resultados de los barridos del haz con sus respectivos ajustes a una gaussiana, y la posición en que estos se hicieron respecto a la salida del láser se muestran en la Figura 10. Aquí vemos que el perfil permanece gaussiano hasta 133cm después de la salida del láser. Ahora tenemos que comprobar que la propagación del haz hasta este punto obedece a las fórmulas de propagación gaussiana. Aquí cabe recalcar que los valores del ancho del haz que se obtienen directamente de la Figura 10, son valores referentes al perfil de intensidad y no al campo. Las fórmulas de propagación gaussiana utilizan los valores del campo. Para

obtener estos valores a partir de la intensidad solo tenemos que multiplicar por $\sqrt{2}$. Otra aclaración pertinente es que el valor de la cintura del haz del campo se denotará como w_0 y para la intensidad será r_0 .

De los perfiles del haz de la Figura 10 tomamos el ancho del haz a $\frac{1}{e}$ del máximo.

Con estos valores y las ecuaciones de propagación gaussiana, podemos resolver un sistema de ecuaciones para encontrar la posición y el tamaño de w_0 . Este proceso se muestra en detalle en el apéndice C, allí se muestra el sistema de ecuaciones resultantes y su solución simbólica. En esta solución se introducen los valores obtenidos en el laboratorio y aquí se muestran los resultados.

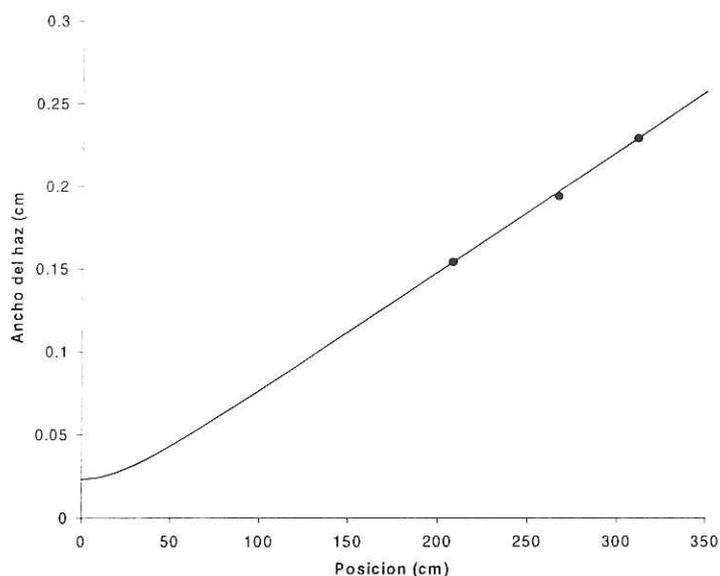


Figura 11 Ajuste de los datos de la propagación gaussiana

Para una longitud de onda de 532nm, y tomando el ancho del haz (w) a 30 y 133cm del láser (1.54 y 2.288mm respectivamente) el ancho de la cintura y su posición con

respecto a la salida del láser son: $w_0 = 231.4\mu m$ y la posición de la cintura es $z_0 = -1.78m$, es decir, la cintura se encuentra antes de la salida del láser. Conociendo estos valores podemos caracterizar completamente al haz. En la Figura 11 se presenta una gráfica del ancho del haz y su posición respecto a la cintura del haz utilizando los valores obtenidos anteriormente en la ecuación para la propagación gaussiana. También se muestran los valores obtenidos con el barrido del haz.

III.3 Resultados experimentales

En esta sección se explica paso a paso el proceso seguido en el laboratorio para obtener los datos experimentales que se utilizarán para encontrar el índice de refracción no lineal del material utilizado en el arreglo experimental.

III.3.1 Medición con baja energía

En esta técnica el dato que nos interesa medir en el laboratorio es el valor de la transmitancia del disco frente a D2. En la simulación obtenemos valores absolutos y normalizados de este parámetro, pero en la práctica necesitamos tener una referencia para poder obtener valores como estos y poder comparar los resultados experimentales con los teóricos. En la práctica esta referencia será el comportamiento del arreglo cuando no se presenta el fenómeno no lineal. Para lograr esto, hacemos que el pulso que se propaga a través del arreglo tenga muy baja energía, para que no cambie el índice de refracción del material. Se utilizan filtros de densidad colocados después del atenuador para disminuir la

energía del pulso. Frente a los detectores D1 y D2 también se colocan filtros para balancear la respuesta de estos, procurando que sea la misma en los dos detectores. En la Figura 12 se muestra el pulso de referencia en D1 y D2. En la obtención de estas gráficas se utilizó una lente de 12.5cm de longitud focal y un disco bloqueando aproximadamente el 97% de la luz.

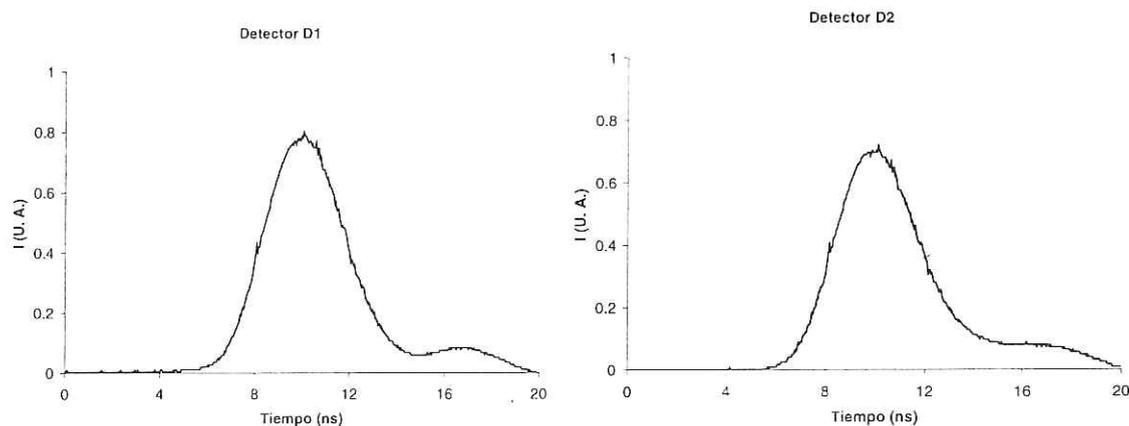


Figura 12. Medición de referencia

El número de puntos en cada gráfica de la Figura 12 es de 500, tomados en un tiempo de 20ns, de acuerdo a la escala. Esto no corresponde ni con la velocidad del osciloscopio ni con la de los detectores. Para obtener tal número de datos en un pulso se utilizó un modo especial de adquisición de datos del osciloscopio llamado "extended time". Se utilizó este modo para incrementar la resolución de los pulsos. Básicamente en este modo de adquisición de datos el osciloscopio toma datos de varios pulsos para reconstruir un solo pulso y rellena los espacios entre los datos. Todos los datos tomados de aquí en adelante se obtuvieron utilizando esta función.

También se podría utilizar un osciloscopio mas veloz para incrementar la resolución de los pulsos. Con un osciloscopio suficientemente rápido se podría realizar la medición utilizando realmente un solo pulso.

III.3.2 Medición con alta energía

El siguiente paso es retirar los filtros colocados después del atenuador. La misma cantidad de filtros que se han retirado se añaden a los filtros frente a los detectores. Así la respuesta de los detectores permanecerá igual, al menos en D1. De esta manera será más fácil hacer el procesamiento de los datos para obtener la transmitancia ya normalizada.

Enseguida se coloca la muestra en una posición cercana a la cintura del haz, después de L1, procurando obtener en esta posición la mayor sensibilidad posible del arreglo. Esto se logra simplemente moviendo la muestra a través del foco y observando en el osciloscopio la respuesta de D2 hasta que obtengamos un máximo.

Ya que el arreglo experimental es básicamente el mismo que se utiliza en el z-scan o ez-scan, habrán dos posiciones donde tendremos máxima sensibilidad para una muestra delgada sin absorción no lineal. Estas corresponden a las posiciones donde se encuentran el valle y el pico de un z-scan o ez-scan normal.

Según [Sheik-Bahae, 1989], la distancia entre la posición del pico y el valle es de aproximadamente $1.7z_0$. Donde $z_0 = \frac{\pi w_0^2}{\lambda}$ (rango de Rayleigh). Si consideramos que la gráfica del z-scan es simétrica con respecto a la cintura del haz, entonces la distancia de la cintura del haz a la posición de la muestra es de $0.85z_0$. En nuestro caso en donde

utilizamos un disco, la posición de la muestra es $-0.85z_0$ respecto a la cintura. Este dato es muy importante porque debemos saber la posición de la muestra, que introduciremos en el programa para hacer el ajuste teórico a los datos experimentales. Una vez colocada la muestra en esta posición, se toman los datos registrados por los detectores D1 y D2 que se muestran a continuación en la Figura 13.

En esta figura vemos que la respuesta del detector D1 es la misma que la mostrada por éste detector cuando se hizo la medición con baja energía (Figura 12). En cambio, la respuesta del detector D2 aumentó respecto su respuesta en la medición de baja energía (Figura 12).

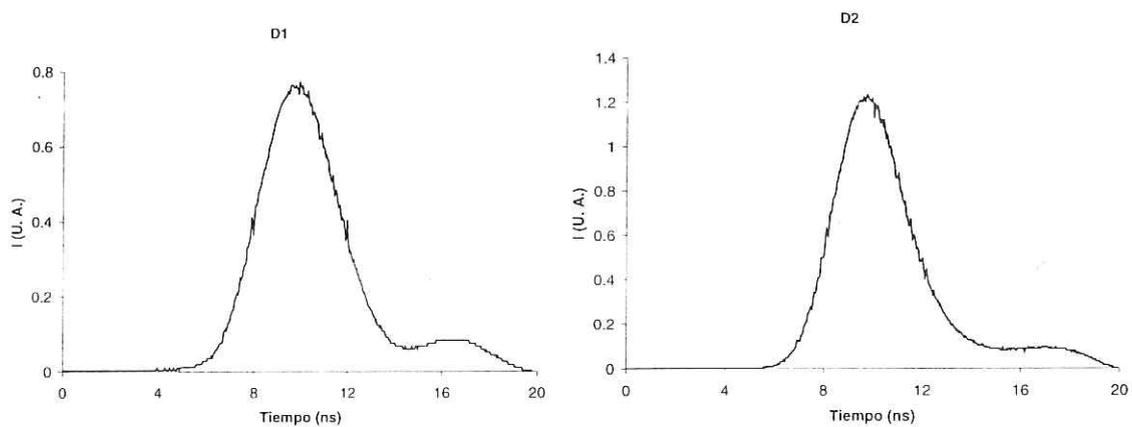


Figura 13. Perfil del pulso en D1 y D2 con alta energía

III.3.3 Procesamiento de los datos

En este momento ya tenemos todos los datos necesarios para obtener el índice de refracción no lineal de la muestra. Los datos obtenidos se tienen que normalizar y posteriormente se hace un ajuste teórico de los resultados experimentales para encontrar n_2 .

Para normalizar, tomamos los datos de la medición de baja energía y dividimos D1 sobre D2. Se hace lo mismo con los datos de alta energía. Finalmente se divide el cociente de los datos de alta energía sobre el cociente de los datos de baja energía. Para tener una idea mas clara de estos pasos, a continuación en la Figura 14 se muestran las gráficas de los cocientes obtenidos junto con un pulso de referencia para observar que pasa en cada punto del perfil del pulso.

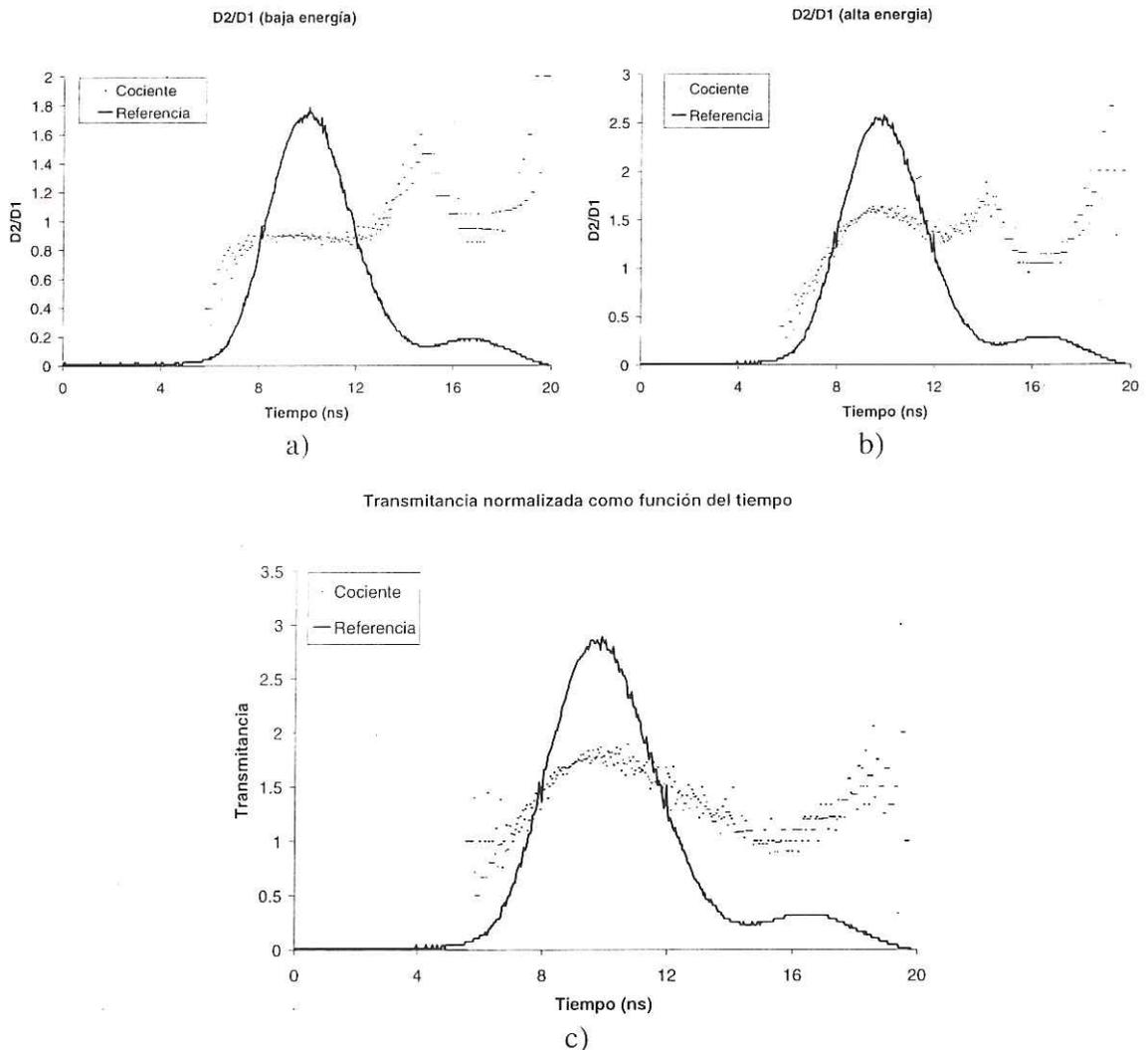


Figura 14 Procesamiento de los datos: a) cociente de los detectores D1 y D2 para baja energía, b) cociente de los detectores D1 y D2 para alta energía, c) cociente de los incisos a y b.

En la Figura 14 vemos que el nivel de ruido durante todo el pulso es el mismo. Pero al realizar el cociente $D2/D1$, al principio y al final del pulso el valor de este cociente puede adquirir valores muy grandes o muy pequeños. Esto se debe a que la relación señal a ruido en los extremos del pulso es muy pequeña. La mayor parte de la información relevante sobre el efecto no lineal en la muestra está en la parte central del pulso, donde la relación señal a ruido es grande. Así podemos decir que las variaciones que se observan en el cociente $D2/D1$ en los extremos del pulso son artificiales y no corresponden al fenómeno que aquí se trata.

Para evitar estas variaciones en la presentación de los datos, graficamos $D2$ vs. $D1$ en la Figura 15. En esta figura se ve claramente que para bajas intensidades no se presentan variaciones. La mayor parte de las variaciones debidas a efectos no lineales se presentan en la forma de una curvatura. En la Figura 15 c) los datos obtenidos con alta energía presentan una curvatura mientras que los obtenidos con baja energía se observan como una línea recta.

Para graficar $D2$ vs $D1$, primero se deben normalizar las respuestas de los detectores obtenidas en la medición de alta energía. Para esto se toma el máximo de la respuesta de $D1$ y $D2$ de la medición de baja energía y se divide sobre la respuesta de éste mismo detector en la medición de alta energía. Finalmente se grafica $D2$ vs $D1$. El resultado es el mostrado en Figura 15.

Aquí se debe hacer otra aclaración. En los datos obtenidos en las mediciones de baja y alta energía, los picos de los pulsos no coinciden, por lo que se debió hacer un ligero corrimiento para que coincidieran. El máximo corrimiento que se hizo corresponde a 200ps,

que es una cantidad que el osciloscopio no puede resolver. El corrimiento puede deberse a una diferencia en la distancia recorrida por el haz hasta los detectores, por ejemplo una diferencia de 5cm equivale a 166ps. O que la lectura de los datos no se hace al mismo tiempo por el osciloscopio.

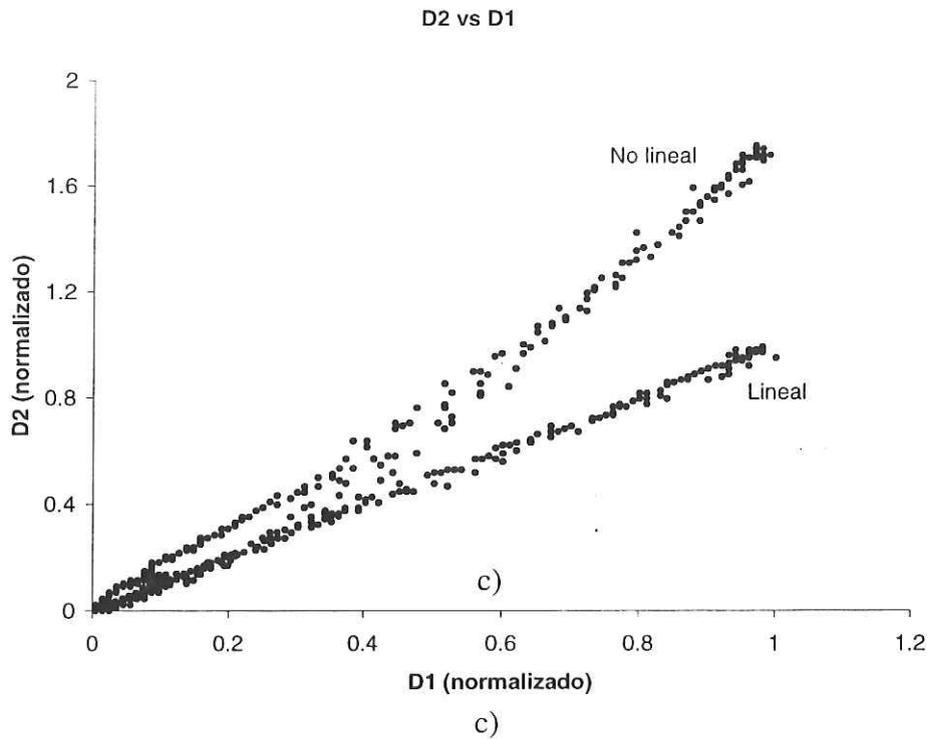
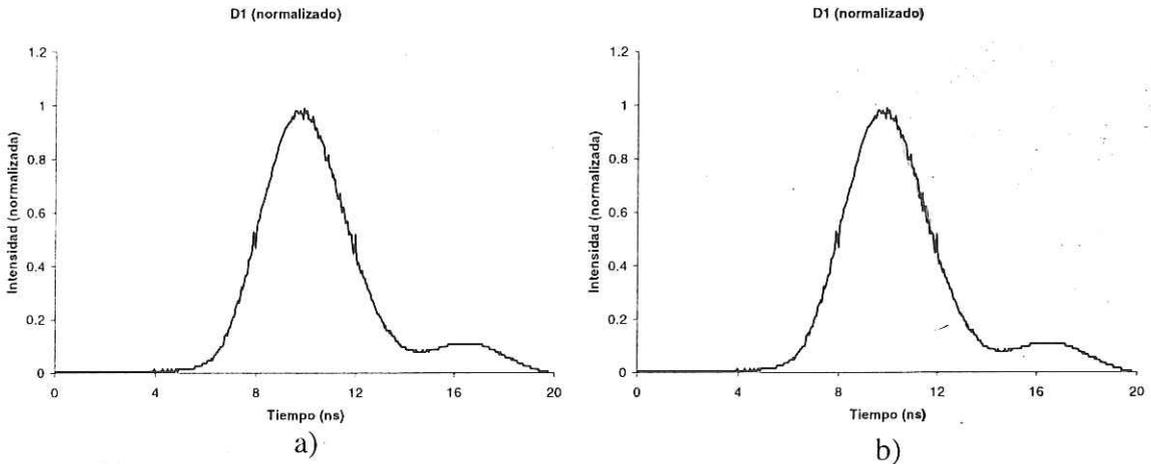


Figura 15 Segundo formato para presentar los datos

III.3.4 Ajuste teórico

Una vez procesados los datos, el siguiente paso consiste en utilizar el programa desarrollado con la teoría del capítulo II. Con este programa se hará el ajuste teórico a los datos experimentales. Se introducen todas las condiciones del arreglo en el programa y se propone un índice de refracción no lineal para el material bajo estudio. Basándose en el resultado de esta simulación se modificará el valor del índice de refracción no lineal hasta lograr el mejor ajuste a los datos experimentales.1

Para nuestro caso particular, el valor de la energía del pulso resultó muy incierto, ya que no contábamos con detectores calibrados para medir con precisión los valores tan pequeños de energía que se utilizaron. Ya que el valor de n_2 para el CS_2 es muy conocido ($1.2E-11 \text{esu}$ [Sheik-Bahae, 1990]). El parámetro que se utilizó para ajustar la teoría a la práctica fue la energía del pulso y no el valor de n_2 . El valor de la energía del pulso obtenido en el ajuste teórico fue de $25\mu\text{J}$ como se muestra en la Tabla III. Ahí se muestran los parámetros de la simulación.

Tabla III Condiciones para el ajuste

Energía del pulso	20 μJ	Longitud focal de L1	12.5cm
Ancho temporal del pulso	7ns	Longitud de la muestra	0.1cm
Longitud de onda	532nm	Valor de n_2 de la muestra	$1.2 \times 10^{-11} \text{esu}$
Posición de la cintura del haz antes de L1	273cm	Transmitancia del disco	5%
Ancho de la cintura	160 μm		

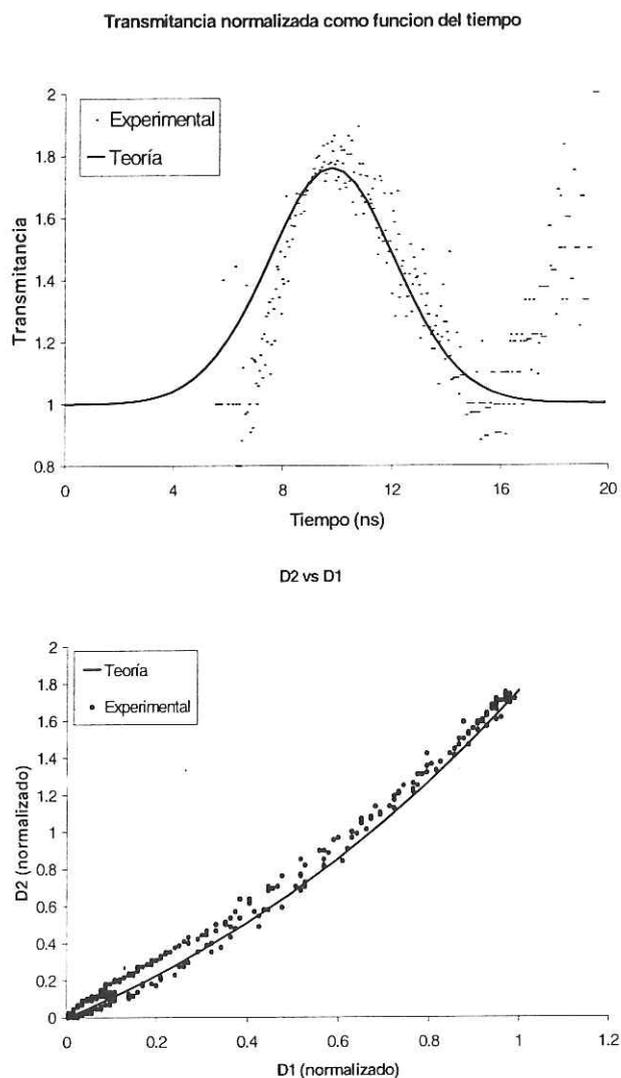


Figura 16 Ajuste de los datos experimentales

El ajuste que se obtuvo con los datos de la Tabla III se muestra en la Figura 16 en los dos formatos propuestos en este trabajo para presentar los datos.

Otros datos experimentales y sus ajustes se muestran a continuación en el formato D2 vs. D1. En la tabla IV están las condiciones de ajuste para la figura 16 y en la tabla V los datos de ajuste de la figura 17.

Tabla IV Datos del ajuste de la Figura 17

Energía del pulso	17 μ J	Longitud focal de L1	12.5cm
Ancho temporal del pulso	7ns	Longitud de la muestra	0.1cm
Longitud de onda	532nm	Valor de n_2 de la muestra	1.2×10^{-11} esu
Posición de la cintura del haz antes de L1	273cm	Transmitancia del disco	5%
Ancho de la cintura	160 μ m		

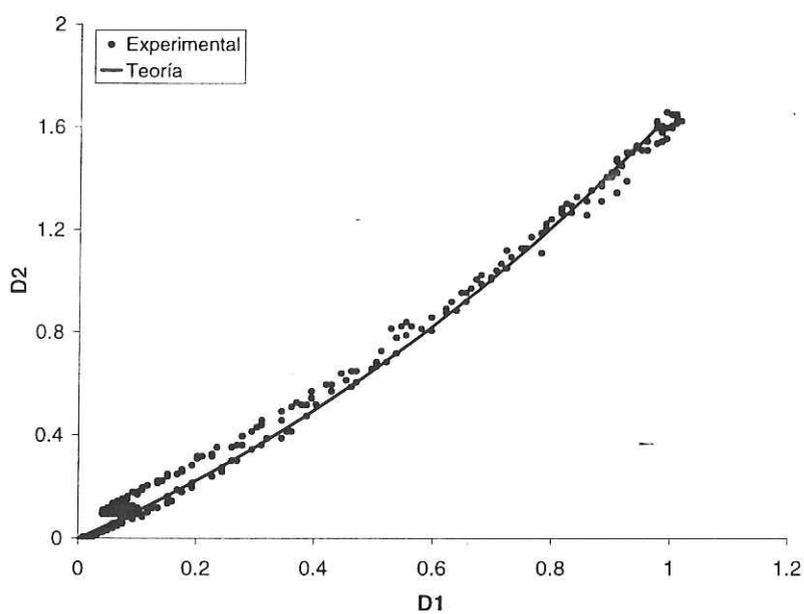


Figura 17 Datos experimentales y ajuste con los datos de la Tabla IV

Tabla V Datos para el ajuste de la Figura 18

Energía del pulso	6 μ J	Longitud focal de L1	12.5cm
Ancho temporal del pulso	7ns	Longitud de la muestra	0.1cm
Longitud de onda	532nm	Valor de n_2 de la muestra	1.2×10^{-11} esu
Posición de la cintura del haz antes de L1	273cm	Transmitancia del disco	5%
Ancho de la cintura	160 μ m		

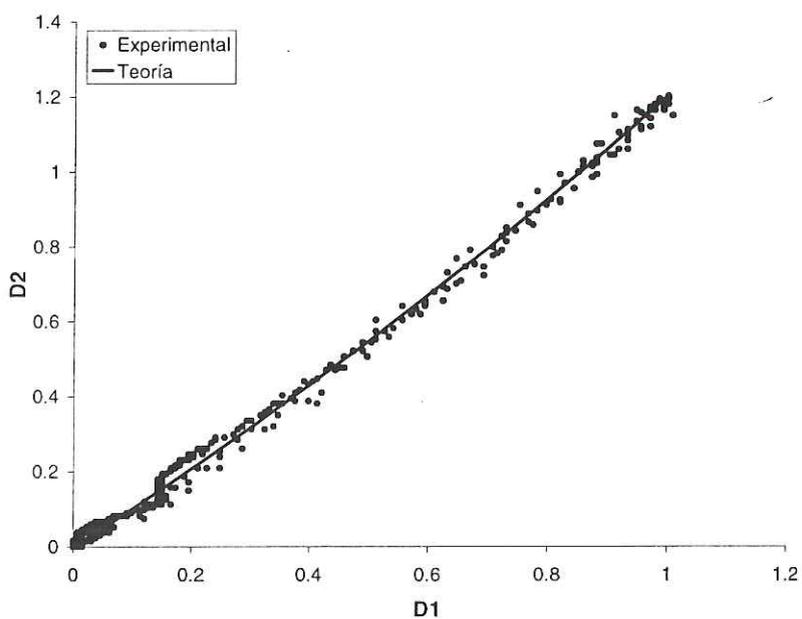


Figura 18 Datos experimentales y ajuste con los datos de la Tabla V

CAPITULO IV

Conclusiones

El objetivo de este trabajo fue desarrollar una técnica capaz de medir el índice de refracción no lineal de un material utilizando para esto un solo pulso. Anteriormente ya se han desarrollado técnicas para este propósito, estas se describieron en la sección I.1. La idea principal de presentar otra técnica más para medir este parámetro es que esta sea sencilla, sensible y rápida. Se propuso un arreglo (figura 2), cuyo funcionamiento se basaba en el z-scan que es una de las técnicas más utilizadas por su sencillez y sensibilidad. El arreglo que propusimos cuenta con dos muestras no lineales, donde una de las muestras básicamente hace un barrido de la cintura del haz en la otra muestra, siguiendo el perfil de intensidad de un pulso de entrada explicado en la sección II. 1.

Para simular este arreglo se desarrolló un programa donde se utilizaron teorías de propagación gaussiana, difracción y óptica no lineal de tercer orden. De los resultados de esta simulación (sección II.3) observamos una sensibilidad bastante buena. Esto es de esperarse dado que la técnica se basa en el z-scan, resultando en una sensibilidad bastante similar.

En esta simulación se incluyó una nueva rutina para la propagación del haz en la muestra no lineal. Esta rutina permite considerar muestras gruesas, donde los cambios del ancho del haz debidos a la difracción natural del haz en la muestra no se pueden despreciar. En esta rutina se utilizó la teoría de difracción bajo la aproximación de Rayleigh-Sommerfeld que nos permitió propagar el haz pequeñas distancias dentro de la muestra.

Durante el desarrollo del modelo teórico, asumimos una respuesta instantánea del material no lineal. Esto simplificó mucho los cálculos que si consideráramos el caso donde la respuesta del material es lenta. Esta técnica también se puede utilizar para este caso, pero los cálculos en el modelo teórico sería más complejos. Se necesitaría resolver un conjunto de ecuaciones de la respuesta del material en junto con las ecuaciones de propagación. Esta aproximación de la respuesta instantánea del material, en el caso de los resultados experimentales obtenidos es válida porque la no linealidad reorientacional del CS_2 tiene un tiempo de respuesta de 2ps. Esto es bastante más rápido que los pulsos de 7ns del láser de NdYAG que se utilizó en el laboratorio.

Otra consideración del programa es que no incluimos la absorción no lineal del material. Esta aproximación también es válida para el CS_2 a una longitud de onda de 532nm, donde la absorción no lineal es muy pequeña para tener algún efecto sobre las mediciones.

Otra aproximación fue suponer que el haz tiene simetría radial. Esto nos permitió utilizar la transformada de Hankel, que simplificó el problema de una transformada de Fourier en dos dimensiones a una sola dimensión, haciendo más rápidos los cálculos de la propagación del haz.

De este primer arreglo propuesto vimos que una sola muestra es suficiente para cambiar la propagación del haz como función del tiempo, siguiendo el perfil de intensidad de un pulso. Así que se propuso un segundo arreglo que solo utiliza una muestra no lineal (sección II.4). El modelo teórico de este arreglo es básicamente el mismo de la primera propuesta, pero aquí el número de elementos en el arreglo es menor y hace más rápida la ejecución del programa.

Este último arreglo con una sola muestra se implementó en el laboratorio porque presentaba mayor sencillez para su implementación, y, además de acuerdo a los resultados teóricos, mostraba una sensibilidad bastante buena. Los datos obtenidos con este arreglo requirieron de cierto procesamiento que básicamente consiste en normalizarlos utilizando los datos obtenidos previamente en una medición de referencia. Los datos ya procesados se presentaron como la transmitancia normalizada del detector D2 de la figura 2. Pero debido a que este formato presentaba algo de ruido, se optó por un segundo formato, donde se graficó el detector D2 vs. D1. Aquí pudimos observar que los efectos no lineales de la muestra se presentaban como una curvatura en la gráfica. Cuando no había ningún efecto no lineal la gráfica era una recta.

La comparación de los datos experimentales con los teóricos fue bastante buena. Para este ajuste se introdujeron en el programa todas las condiciones experimentales y el valor muy conocido y reportado en la literatura del índice de refracción no lineal del CS_2 . El valor de la energía que utilizamos en las mediciones era incierto, ya que no contábamos con un detector calibrado para medir valores tan pequeños de energía. Siendo así, para lograr los ajustes teóricos el valor que cambiamos en el programa para hacer los ajustes fue la energía del pulso.

La técnica ha demostrado ser sencilla para implementar en el laboratorio, al menos tan sencilla como el z-scan y sus variantes. La sensibilidad es tan buena e incluso mayor que el z-scan. Esta mayor sensibilidad se debe a que en el z-scan se utilizan detectores lentos que integran el pulso en el tiempo y dan como respuesta el área bajo la curva del pulso. En nuestra técnica tomamos la respuesta instantánea del material. En el z-scan al hacer la comparación de las áreas de los pulsos, obtenemos un resultado menor

que si comparáramos los valores de los picos de los pulsos. Los errores experimentales que se podría tener por las variaciones en el ancho temporal y la intensidad pico entre los pulsos al utilizar el z-scan, no se aplican a nuestra técnica, ya que solo utilizamos un solo pulso.

Como trabajo futuro se puede proponer la extensión de la técnica para medir también la absorción no lineal (saturable o inducida). También como trabajo futuro se propone trabajar con otros tipos de perfil de haz, por ejemplo haces tipo "Top Hat" (sombrero de copa), para incrementar la sensibilidad de la técnica.

Literatura citada

- Boyd, Robert. 1992. "Nonlinear Optics". Academic Press Inc. Primera edición. Londres. 439pp.
- Gaskill, J. 1978. "Linear Systems, Fourier Transforms and Optics". Wiley Interscience. Primera edición. New York. 554 pp.
- Goodman, J. 1988. "Introduction to fourier Optics". McGraw Hill. Segunda Edición. New York. 287pp.
- Magni, Vittorio, Glulio Cerulo y Sandro de Silvestri. 1992. "High-accuracy Fast Hankel Transform for Optical Beam Propagation". J. Opt. Soc. of Am. 9(10): 2031-2033 p.
- Paras N. Prasad, David J. Williams. 1991. "Introduction to Nonlinear Optical Effects in Molecules and Polymers". John Wiley & Sons, Inc. Primera edición. New York. 307pp.
- Reitz, J. R., Milford F. J., Christy R. W. 1979. "Foundations of electromagnetic theory". Addison Wesley. Tercera edición. Massachusetts. 534 pp.
- Sheik-Bahae, M. A. A. Said y E. W. Van Stryland. 1989. "High-Sensitivity Single Beam n_2 Measurements". Opt. Lett. 14(17):955-957 p.
- Sheik-Bahae, M. A. A. Said, Tai-Huei Wei, David J. Hagan, W. Van Stryland. 1990. "Sensitive Measurement of Optical Nonlinearities Using a Single Beam". IEEE J. of Quantum Elec. 26(4):760-769 p.
- Verdeyen, Joseph T. 1996. "Laser Electronics". Prentice Hall. Tercera edición. Amsterdam. 785 pp.

Apéndice A (Teoría de óptica no lineal de tercer orden)

Partimos de la ecuación de onda para medios dieléctricos dada por [Verdeyen, 1996]

$$\nabla^2 \mathbf{E} - \frac{n_0^2}{c^2} \frac{\partial^2 \mathbf{E}}{\partial t^2} = \mu_0 \frac{\partial^2 \mathbf{P}}{\partial t^2}. \quad \text{A-1}$$

Resolviendo la ecuación anterior para la polarización no lineal de tercer orden

$$\mathbf{P} = \epsilon_0 \chi^{(3)} \mathbf{E} \mathbf{E} \mathbf{E} \quad \text{A-2}$$

obtenemos

$$\nabla^2 \mathbf{E}(z, t) - \frac{n_0^2}{c^2} \frac{\partial^2 \mathbf{E}(z, t)}{\partial t^2} = \mu_0 \epsilon_0 \chi^{(3)} \frac{\partial^2}{\partial t^2} \langle \mathbf{E}^2 \rangle \mathbf{E}(z, t), \quad \text{A-3}$$

donde $\langle \mathbf{E}^2 \rangle = \frac{1}{2} |\mathbf{E}^2|$ [Reitz et al., 1979] es el promedio temporal de la amplitud. De (A-3)

se obtienen soluciones de la forma [Boyd, 1992]

$$\mathbf{E}(z, t) = \text{Re} \left\{ \mathbf{E}(z) e^{-i(\omega t - kz)} \right\}. \quad \text{A-4}$$

Sustituyendo (A-4) en (A-3) y bajo la aproximación de variaciones lentas [Boyd, 1991]

obtenemos

$$\left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} \right) \mathbf{E}(z, t) + 2in_0 k_0 \frac{\partial \mathbf{E}(z, t)}{\partial z} = -\frac{\omega^2}{2c^2} \chi^{(3)} |\mathbf{E}|^2 \mathbf{E}(z, t). \quad \text{A-5}$$

Para una onda plana las variaciones radiales en x y en y son nulas, entonces la ecuación

(A-5) queda

$$\frac{\partial \mathbf{E}(z, t)}{\partial t} = \frac{i\omega}{4n_0 c} \chi^{(3)} |\mathbf{E}|^2 \mathbf{E}(z, t). \quad \text{A-6}$$

Tomando la ecuación (A-4) con la amplitud compleja siendo

$$\mathbf{E}(z) = A(z)e^{i\phi(z)}, \quad \text{A-7}$$

sustituyendo en la ecuación A-6 y considerando la parte real y la parte imaginaria de $\chi^{(3)}$ obtenemos respectivamente

$$\frac{\partial \phi}{\partial z} = \frac{\omega}{4n_0 c} \chi_r^{(3)} A^2 \quad \text{A-8a}$$

y

$$\frac{\partial A}{\partial z} = \frac{\omega}{4n_0 c} \chi_i^{(3)} A^3. \quad \text{A-8b}$$

De la ecuación A-8a observamos que $\chi_r^{(3)}$ se relaciona con los cambios de fase, por lo tanto con el índice de refracción no lineal. Mientras que en la ecuación (A-8b) $\chi_i^{(3)}$ se relaciona con los cambios en la amplitud, por lo tanto con la absorción no lineal que no se trata en este trabajo. El caso de interés de este trabajo es el índice de refracción no lineal. Resolviendo (A-8a) obtenemos

$$\phi(z) = \frac{\omega}{4n_0 c} \chi_r^{(3)} A^2 z. \quad \text{A-9}$$

Tomando la fase total del campo de la ecuación (A-4) y sustituyendo (A-9) en (A-7) nos da

$$\Phi(z, t) = \omega t - k_0 z \left(n_0 + \frac{\chi_r^{(3)} A^2}{4n_0} \right), \quad \text{A-10}$$

donde consideramos que $k = k_0 n_0 = \frac{n_0 \omega}{c}$. De (A-10) podemos tomar

$$n = n_0 + \frac{\chi_r^{(3)} A^2}{4n_0} = n_0 + n_2 I, \quad \text{A-11}$$

$$\text{con } n_2 = \frac{c \mu_0 \chi_r^{(3)}}{n_0^2}.$$

Retomando la ecuación (A-9) y haciendo $\frac{\omega}{c} = k_0$ esta ecuación queda

$$\phi(z) = \frac{k_0}{4n_0} \chi_r^{(3)} A^2 z. \quad \text{A-12}$$

La distancia a propagar z la podemos sustituir por L , de acuerdo a la ecuación (A-11)

donde $n_2 I = \frac{\chi_r^{(3)} A^2}{4n_0}$ y tomando en cuenta que el haz es gaussiano y su intensidad varía en r ,

la expresión final para la variación de la fase es

$$\phi(r, L) = k_0 n_2 I(r) L, \quad \text{A-13}$$

donde

$$I = \frac{n_0 \epsilon_0 c}{2} |E|^2. \quad \text{A-14}$$

Apéndice B (Teoría de difracción).

Consideremos el principio de Huygens-Fresnel, básico para la teoría de difracción. Este principio lo podemos formular matemáticamente de la siguiente manera [Goodman, 1996]:

$$U(P_0) = \frac{1}{j\lambda} \iint_{\Sigma} U(P_1) \frac{e^{jk r_{01}}}{r_{01}} \cos\theta ds, \quad \text{B-1}$$

donde $\cos\theta = \frac{z}{r_{01}}$. De acuerdo a la figura B1 esta ecuación representa el campo óptico en P_0

como la superposición de las ondas esféricas divergentes $\frac{e^{jk r_{01}}}{r_{01}}$ originadas en las fuentes secundarias localizadas en las posiciones P_1 dentro de la abertura difractora Σ .

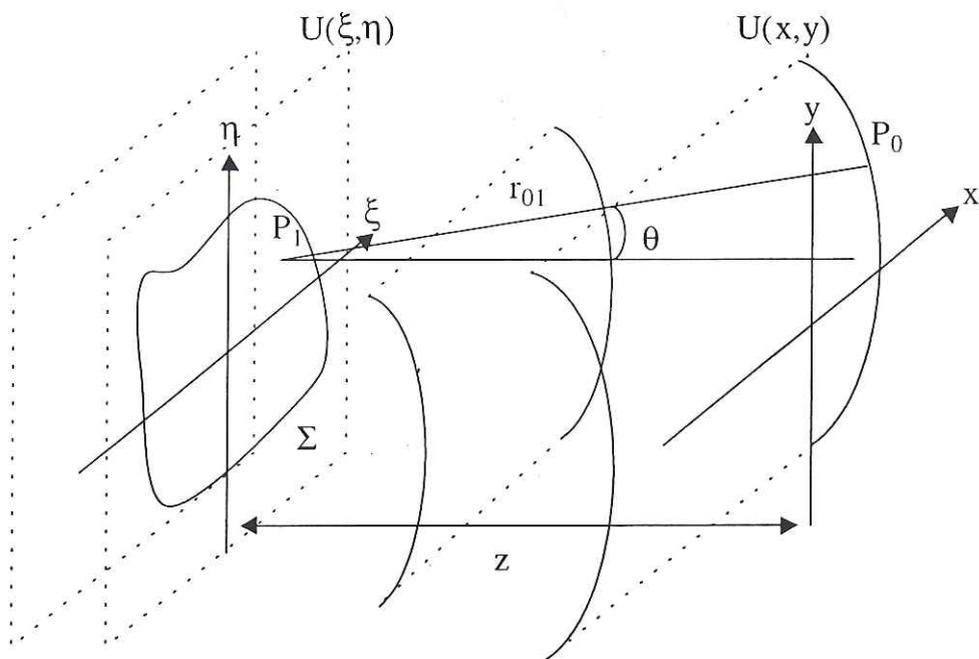


Figura B1 Principio de Huygens-Fresnel

Aproximación de Fresnel

Para resolver la ecuación (B-1), introducimos dos aproximaciones para la distancia r_{01} . La primera aproximación es para r_{01} en el factor exponencial, que se basa en la aproximación

binomial de la raíz cuadrada ($\sqrt{1+b} = 1 + \frac{b}{2} - \frac{b^2}{8} + \dots$) aplicada a:

$$r_{01} = \sqrt{z^2 + (x-\xi)^2 + (y-\eta)^2} \quad \text{B-2}$$

que después de sencillas manipulaciones algebraicas queda:

$$r_{01} = z \sqrt{1 + \left(\frac{x-\xi}{z}\right)^2 + \left(\frac{y-\eta}{z}\right)^2} \quad \text{B-3}$$

y aplicando la aproximación binomial hasta el segundo término nos da:

$$r_{01} \cong z \left[1 + \frac{1}{2} \left(\frac{x-\xi}{z}\right)^2 + \frac{1}{2} \left(\frac{y-\eta}{z}\right)^2 \right] \quad \text{B-4}$$

Para esta aproximación podemos asegurar que el valor del tercer término será menor que un radian si [Gaskill, 1978]

$$z^3 \gg \frac{\pi}{4\lambda} \left[(x-\xi)^2 + (y-\eta)^2 \right]_{max} \quad \text{B-5}$$

La segunda aproximación la haremos en el denominador de (B-1). Según [Gaskill, 1978] podemos hacer $r_{01} \cong z$ si cumplimos con la condición

$$z \gg \left[(x-\xi)^2 + (y-\eta)^2 \right]_{max} \quad \text{B-6}$$

Bajo estas dos aproximaciones la ecuación para el campo difractado queda:

$$U(x, y) = \frac{e^{-jkz}}{j\lambda z} e^{\frac{jk}{2z}(x^2+y^2)} \int \int_{-\infty}^{\infty} U(\xi, \eta) e^{\frac{jk}{2z}(\xi^2+\eta^2)} \left[e^{-\frac{j2\pi}{\lambda z}(x\xi+y\eta)} \right] d\xi d\eta \quad \text{B-7}$$

Aproximación de Rayleigh-Sommerfeld

Para este caso haremos la misma expresión para r_{01} , tanto en el denominador como en el factor exponencial de (B-1). La expresión es la ecuación (B-3), que sustituyéndola en (B-1) nos da la ecuación de difracción de Rayleigh-Sommerfeld.

$$U(x, y) = \int \int_{-\infty}^{\infty} U(\xi, \eta) \frac{1}{j\lambda z} \frac{e^{jkz \left[1 + \left(\frac{x-\xi}{z} \right)^2 + \left(\frac{y-\eta}{z} \right)^2 \right]^{1/2}}}{\left[1 + \left(\frac{x-\xi}{z} \right)^2 + \left(\frac{y-\eta}{z} \right)^2 \right]} d\xi d\eta \quad \text{B-7}$$

Para hacer más sencilla la solución a esta última ecuación, la podemos ver como una convolución del campo que queremos propagar $U(\xi, \eta)$ y la función de respuesta al impulso

$$h(x, y, \xi, \eta) = \frac{1}{j\lambda z} \frac{e^{jkz \left[1 + \left(\frac{x-\xi}{z} \right)^2 + \left(\frac{y-\eta}{z} \right)^2 \right]^{1/2}}}{\left[1 + \left(\frac{x-\xi}{z} \right)^2 + \left(\frac{y-\eta}{z} \right)^2 \right]} \quad \text{B-8}$$

Ahora tomamos la transformada de Fourier de la ecuación (B-7). De acuerdo a las propiedades de la transformada de Fourier de una convolución es la multiplicación de las transformadas de las dos funciones de la convolución, es decir

$$U(x, y) = U(\xi, \eta) * h(\xi, \eta) \quad \text{B-9}$$

$$\mathfrak{F}\{U(x, y)\} = \mathfrak{F}\{U(\xi, \eta)\} \mathfrak{F}\{h(\xi, \eta)\} \quad \text{B-10}$$

Y el último paso después de realizar el producto de las transformadas es obtener la transformada inversa para obtener finalmente el campo ya difractado.

Apéndice C (propagación gaussiana)

Las fórmulas de la propagación gaussiana son bien conocidas, y, a continuación se presenta la ecuación general para un haz gaussiano [Yariv, 1991]

$$\mathbf{E}(r) = \mathbf{E} \frac{w_0}{w(z)} e^{-i[kz - \eta(z)] - r^2 \left(\frac{1}{w^2(z)} + \frac{ik}{2R(z)} \right)}, \quad \text{C-1}$$

donde $k = \frac{2\pi}{\lambda}$ es el número de onda, $\eta(z) = \text{tg}^{-1} \left(\frac{z}{z_R} \right)$ es el fasor, $z_R = \frac{\pi w_0^2}{\lambda}$ es el rango de

Rayleigh, $R(z) = z \left[1 + \left(\frac{z_R}{z} \right)^2 \right]$ es el radio de curvatura, $w(z) = w_0 \sqrt{1 + \left(\frac{z}{z_R} \right)^2}$ es el ancho

del haz en z y w_0 es el ancho de la cintura del haz.

En la sección III.2 se caracterizó el haz haciendo algunos barridos del perfil del haz en distintas posiciones después de la salida del láser. De estas mediciones se pudo obtener el ancho del haz y la distancia relativa entre ellos. Lo que nos interesa conocer es el ancho de la cintura y su posición para poder caracterizar completamente al haz de entrada al arreglo. Para hacer esto solo necesitamos conocer el ancho del haz en dos posiciones y la distancia entre estas dos posiciones. Se introducen estos valores en la ecuación para el ancho del haz y se resuelve el sistema de ecuaciones. A continuación se muestran las dos ecuaciones obtenidas y su solución simbólica obtenida en Mathcad.

$$w_1 = w_0 \sqrt{1 + \left[\frac{z_1}{\left(\frac{\pi w_0^2}{\lambda} \right)} \right]^2} \quad \text{C-2a}$$

$$w_2 = w_0 \sqrt{1 + \left[\frac{z_1 + d}{\left(\frac{\pi w_0^2}{\lambda} \right)} \right]^2} \quad \text{C-2b}$$

Donde w_1 es el ancho del haz en la posición z_1 , d es la distancia entre las dos mediciones de w_1 y w_2 , w_2 es el ancho de la cintura en la posición (z_1+d) y λ es la longitud de onda.

Los parámetros que queremos encontrar son z_1 que es la distancia de w_1 a la cintura del haz y w_0 que es el ancho de la cintura del haz. Resolviendo C-2a y C-2b para w_0 y z_1 obtenemos un conjunto de cuatro ecuaciones (C-3a, C-3b, C-3c y C-3d).

De la figura 9 en la sección III.2 tomamos los valores para el ancho del haz para 30 y 133cm después de la salida del láser (w_1 y w_2 respectivamente). Introducimos los siguientes valores en el conjunto de ecuaciones C-3

$$w_1 = 1.54 \times 10^{-3} m$$

$$w_2 = 2.28 \times 10^{-3} m$$

$$d = 1.03 m$$

$$\lambda = 532 \times 10^{-9} m$$

Los valores obtenidos para el primer conjunto de soluciones w_{01} y z_1 ((C-3a) y (C-3b) respectivamente) es

$$w_{01} = 231 \mu m$$

$$z_1 = 2.08 m$$

Para el segundo conjunto de ecuaciones w_{02} y z_{12} ((C-3c) y (C-3d) respectivamente) obtenemos

$$w_{02} = 45\mu m$$

$$z_{12} = -0.41m$$

Con estos resultados podemos decir que los valores que buscamos son los de la primera solución. Los resultados de la segunda solución no concuerdan con los resultados observados en el laboratorio. La distancia z_{12} indica que la cintura del haz está fuera del láser, lo cual no es correcto. De esta manera asumimos que los resultados de la primera solución son los correctos. Para comprobarlos, introducimos los valores obtenidos de la primera solución en la ecuación para el ancho del haz y calculamos este valor para 0.89m después de la salida del láser.

$$z = z_1 + 0.59m$$

$$z_R = \frac{\pi(230 \times 10^{-6} m)^2}{532 \times 10^{-9} m} = 0.31m$$

$$w_0 = w_{01}$$

$$w(z) = 0.00197m$$

El valor obtenido para esta posición es de $w = 0.00193m$ que demuestra que la primera solución es la correcta y que además la propagación del haz es gaussiana.

$$w01 := \frac{1}{2} \frac{\sqrt{2} \sqrt{4d^2 \lambda^2 \pi^2 + w1^4 \pi - 2w1^2 \pi \cdot w2^4 \pi}}{2 \sqrt{2 \lambda^2 d^2 w2^2 \pi^2 + 2 \lambda^2 d^2 w1^2 \pi^2 - 4 \sqrt{\lambda^4 d^4 \pi^2} \cdot (-w1^2 \pi \cdot w2^2 + \lambda^2 d^2)}} \quad \text{C-3a}$$

$$z1 := \frac{-1}{2} \frac{\left[\frac{\lambda^2 d^2 + 1}{2} \frac{\pi^2}{(4d^2 \lambda^2 \pi^2 + w1^4 \pi - 2w1^2 \pi \cdot w2^4 \pi)} \left[\frac{2 \lambda^2 d^3 w2^2 \pi^2 + 2 \lambda^2 d^3 w1^2 \pi^2 + 4 \sqrt{\lambda^4 d^4 \pi^2} \cdot (-w1^2 \pi \cdot w2^2 + \lambda^2 d^2)}{\pi^2} \right] w1^2 - 1 \right]}{2 \left(4d^2 \lambda^2 \pi^2 + w1^4 \pi - 2w1^2 \pi \cdot w2^4 \pi \right)} \frac{\pi^2}{(\lambda^2 d^2)} \left[\frac{2 \lambda^2 d^3 w2^2 \pi^2 + 2 \lambda^2 d^3 w1^2 \pi^2 + 4 \sqrt{\lambda^4 d^4 \pi^2} \cdot (-w1^2 \pi \cdot w2^2 + \lambda^2 d^2)}{\pi^2} \right] w2^2$$

C-3b

$$w02 := \frac{1}{2} \frac{\sqrt{2} \sqrt{2 \lambda^2 d^2 w2^2 \pi^2 + 2 \lambda^2 d^2 w1^2 \pi^2 - 4 \sqrt{\lambda^4 d^4 \pi^2} \cdot (-w1^2 \pi \cdot w2^2 + \lambda^2 d^2)}}{2 \sqrt{4d^2 \lambda^2 \pi^2 + w1^4 \pi - 2w1^2 \pi \cdot w2^4 \pi}} \quad \text{C-3c}$$

$$z12 := \frac{-1}{2} \frac{\left[\frac{\lambda^2 d^2 + 1}{2} \frac{\pi^2}{(4d^2 \lambda^2 \pi^2 + w1^4 \pi - 2w1^2 \pi \cdot w2^4 \pi)} \left[\frac{2 \lambda^2 d^3 w2^2 \pi^2 + 2 \lambda^2 d^3 w1^2 \pi^2 - 4 \sqrt{\lambda^4 d^4 \pi^2} \cdot (-w1^2 \pi \cdot w2^2 + \lambda^2 d^2)}{\pi^2} \right] w1^2 - 1 \right]}{2 \left(4d^2 \lambda^2 \pi^2 + w1^4 \pi - 2w1^2 \pi \cdot w2^4 \pi \right)} \frac{\pi^2}{(\lambda^2 d^2)} \left[\frac{2 \lambda^2 d^3 w2^2 \pi^2 + 2 \lambda^2 d^3 w1^2 \pi^2 - 4 \sqrt{\lambda^4 d^4 \pi^2} \cdot (-w1^2 \pi \cdot w2^2 + \lambda^2 d^2)}{\pi^2} \right] w2^2$$

C-3d

Apéndice D (Algoritmo de la transformada rápida de Hankel)

El algoritmo para la transformada rápida de Hankel se expresa como [Magni et al., 1992]

$$\hat{U}(y_m) = \frac{1}{N_f y_m} FFT[FFT(\varphi_n) \times IFFT(J_{1n})]. \quad D-1$$

Para definir las variables de esta ecuación debemos retomar la ecuación 8 en la sección II.2.1.1. De la ecuación 8 se han normalizado los valores de las variables ρ y r de la siguiente forma

$$x = \frac{\rho}{\rho_m} \quad D-2a$$

$$y = \frac{r}{r_m}, \quad D-2b$$

donde r_m es el valor del radio del haz y ρ_m es la máxima frecuencia en el espacio de Hankel-Fourier. Con esto ya podemos definir $N_f = r_m \rho_m$. Las demás variables de (D-1) son

$$J_{1n} = J_1(2\pi N_f x_0 e^{\alpha(n+1-N)}), \quad n = 0, 1, 2, \dots, 2N-1 \quad D-3$$

$$\varphi_n = \left\{ \begin{array}{ll} k_0 [f(x_0) - f(x_1)] e^{\alpha(1-N)}, & n = 0 \\ [f(x_n) - f(x_{n+1})] e^{\alpha(n+1-N)}, & n = 1, 2, \dots, N-1 \\ 0, & n = N, N+1, \dots, 2N-1 \end{array} \right\}, \quad D-4$$

donde

$$k_0 = \frac{2e^\alpha + e^{2\alpha}}{(1+e^\alpha)(1-e^{-2\alpha})} \quad D-5$$

$$\alpha = -\frac{\ln(1-e^\alpha)}{(N-1)} \quad D-6$$

$$x_0 = \frac{(1 + e^\alpha e^{-\alpha N})}{2} \quad \text{D-7}$$

$$x_n = y_n = x_0 e^{\alpha n}, \quad n = 1, 2, \dots, N-1 \quad \alpha > 0, \quad \text{D-8}$$

siendo $f(x_n)$ la función que vamos a hacer un muestreo.

Un punto muy importante para poder llevar a cabo este algoritmo es el muestreo de $f(x_n)$.

El muestreo se hace dividiendo el intervalo $0 \leq x \leq 1$ en N subintervalos utilizando para

ello $N+1$ puntos ξ_n ($n = 0, 1, 2, \dots, N$), es decir $0 = \xi_0 < \xi_1 < \dots < \xi_N = 1$, donde

$$\xi_0 = 0 \quad \text{D-9a}$$

$$\xi_n = e^{\alpha(n-N)}. \quad \text{D-9b}$$

Apéndice E (Descripción del programa)

A continuación se describen de manera breve algunas de las rutinas que utiliza el programa que se utilizó para simular los arreglos experimentales que se propusieron en este trabajo. Una discusión mas extensa y detallada de este programa se hace en el trabajo previo de donde se tomaron las rutinas que aquí se describen [Morales Saavedra, 1997].

Initial.- Aquí se propaga el haz hasta antes de la muestra utilizando las fórmulas para la propagación gaussiana del apéndice C utilizando los valores para el ancho de la cintura (w_0) y su posición antes de la lente de la figura 6 (z_1). La distancia de la lente a la muestra está dada por (z_1). En este punto se construye el campo y se guarda en el arreglo $U(N)$.

Sample.- En esta rutina se propaga el campo a través de una muestra no lineal de longitud ($sample$) con un índice de refracción no lineal (n_2) asumiendo una muestra delgada. La propagación se hace añadiendo al campo la contribución a la fase por la muestra dada por la ecuación (15) de la sección II.2.2.

Sampleray.- Esta rutina también propaga el campo a través de una muestra no lineal, solo que no la consideramos como una muestra delgada. Para propagar el campo se utiliza el procedimiento de la sección II.2.2.1 para una muestra gruesa.

Frenel.- Propaga el campo en el espacio libre entre los elementos del arreglo. Para esto utiliza la teoría de difracción bajo la aproximación de Fresnel de la sección II.2.1.1.

Raylay.- Se utiliza en la rutina **sampleray** para propagar el campo distancias muy pequeñas dentro de la muestra no lineal. Así sumamos las contribuciones de fase debidas a la difracción natural del haz al propagarse dentro de la muestra. La teoría involucrada se muestra en la sección II.2.1.2 Índice

Algunas de las pruebas que se realizaron con el programa para comprobar su correcto funcionamiento son las siguientes.

Se calculó la energía del pulso a través del arreglo y después de cada elemento hasta la abertura para asegurarnos de la conservación de la energía.

También se modificó el número de puntos en el perfil del haz para encontrar y en el perfil del pulso para encontrar los valores que hacían más eficiente la simulación conservando la exactitud y precisión de los resultados.

Para la rutina donde la muestra se considera gruesa se varió el número de secciones en que la muestra se dividía. Así se pudo encontrar el mínimo número de secciones en que se podía dividir la muestra sin aumentara el error en la propagación de manera significativa. Se realizó la propagación de pulsos de baja energía procurando que las muestras no afectaran la propagación del haz, es decir una propagación lineal. Esta propagación es decir el ancho del haz se comparó con el que se obtendría utilizando solo las fórmulas de propagación gaussiana.

Finalmente se realizaron con las mismas rutinas algunos z-scan para compara con los resultados mostrados en la literatura. Todas estas pruebas se realizaron a pesar que estas rutinas ya se habían probado en un trabajo anterior [Morales Saavedra, 1997].

```

////////////////////////////////////
//  P R O G R A M A    Z S C A N . C P P          //
////////////////////////////////////

int DIM, cal; nu, ng;
char tecla;
int wait=1, nocal=0;
double ener0, ener01, zstart, rhoms[10], tindex0, rayran;
int  nums[20],nr,nz,niez, nbeam, np2,NS,NE, dirprop;
double pums[20], alpha, x0, outs, sdisk ;
double huge *xx, *T1, *T2, *P10, *P11, *P12, *P20, *P21, *P22, *TT, *FF;
double engy, sampl,pulsew,hwem,deltaT,wavlen,spot,n2,n21,focall,zel,dskd,
        focal2,apd,z1,z2,z3,z30,cof,pi, w0, w30, aw1, aw2, aw3,s2,
        localw0,
localz0, inpute, konst, kconst, lwl, lz1, refwl, refz1, refw2, refz2;
double spdl=2.99792458e10 ,rm, rhom, rhomx, rmUL;
double Range,wavnum,peak,W30THB,IoTHB,IoGAUS;
//int OPT;          //////////////// Graphics ready ////////////////
unsigned long int libre;

#include <dos.h>
#include <stdio.h>
#include <alloc.h>
#include <STDLIB.H>
#include <conio.h>
#include <MATH.H>
#include <COMPLEX.H>
#include <string.h>

FILE *arch, *datosJI, *datosFY , *datosU,
      *datosUT,*datosUTF, *datos,*datosSR;
complex huge *J1, huge *phase;
complex huge *U, huge *FY, huge *tmpu, huge *frac;

#define SQR(a) ((a)*(a))
#define GRAPH 1
#define GAUSSIAN 0
#define THB 1
#define HOLE 0
#define DISK 1
#define NO 0
#define YES 1
//#define ng 10

void computeJI(void);
void initial(void); // fills the E-field at apperture
void assignmemo(void); // assigns memory for arrays
void fhankel(complex huge *U,complex huge *J1,int tam); // calculates
FHT
void Frenel (complex huge *U, double z,double par);// Frenel prop.
void Raylay(complex huge *U,double z,double parl,double par2); //Raylay
prop.
void lens(complex huge *U, double F); // propagates through a thin lens
void scanning(void);// begins the Z or IEZ-Scan

```

```

void beamsize (complex huge *U, double W, double par); // evaluates the
beamsize
double omega(complex huge *U, double rm);
void sampleray(complex huge *U,double z, double nl, int
steps);//propagates through a NLS by steps
void sample(complex huge *U, double nl); // propagates through a NLS
void sndout(complex huge *U);// plots the phase profiles of the beam
double energy(complex huge *arr, double rs, double re, double mrad);
double rangebinf(double zr, double guess);
double energy_circle(complex huge *U,double mrad,double center);
double alfa(int tam);
void guslms(double w10, double z10, double fl, double Z2,
            double *W2, double *z20, double *w20);
void guspro(double w10, double zcom, double zx, double *wx);
void rcalib(void);
/*****/
#include "setup.h"
#include "plott.h"
#include "engine.h"
//#include "ranges.h"

void main()
{
    assignmemo();
    //OPT = 1;
    wait=0;
    rcalib();
    cal=0;
    //wait=0;
    scanning();
    farfree(FY); farfree(xx); farfree(J1);
    farfree(tmpu); farfree(U); farfree(frac);
    farfree(T1); farfree(T2); farfree(FF); farfree(TT); farfree(P10);
    farfree(P11); farfree(P12); farfree(P20); farfree(P21); farfree(P22);
    exit(0);
}

double getdisk(double tindex)
// called by calibonce
// Assumption is that there is no nonlinearity.
{
    int ii,it;
    double ener, dr, r,sin,oldsin,T;
// kconst=0;
// for (ciclo=0; ciclo<ng; ciclo++){
//     tim=ciclo*deltaT - 3.0* hwem; // time.
//     factor = exp(-SQR(tim/hwem));
//     kconst=kconst+factor;
//     }
//     kconst=2.0*kconst*deltaT;
// kconst=sqrt(M_PI)*hwem; exactly
dr=xx[0];
ener=kconst*kconst * SQR(abs(U[0])) * xx[0] * rm * dr * 0.5;
ii=0;

```

```

do
{
    dr=(xx[ii+1]-xx[ii])*rm;

ener=ener+kconst*kconst*(((SQR(abs(U[ii])))*xx[ii]+SQR(abs(U[ii+1])))*xx[ii+
1])/2.0)*rm*dr);
    ii++;
} while (((ener/ener01) < tindex) && (ii<DIM-2));
r=rm*xx[ii];
ener=kconst*energy(U,0,r,rm);
T=ener/ener01;
if (T<tindex) sin=1;
                                     else sin=-1;

it=0;
do{
    r = r + dr*sin;
    oldsin=sin;
    T=kconst*energy(U,0,r,rm)/ener01;
    if (T<tindex) sin=1;
                                     else sin=-1;
    if (sin!=oldsin) dr=dr/2.0;
    it=it+1;
}while(((fabs(T-tindex)>0.00000001) && (it < 30)));

return 2*r;
}

int calibonce(void)
{
////////////////////////////////////
// Calibrates Once. Called by rcalib ( Repeat Calibration )
// Se evalua la propagacion del haz, //
// pero sin MNL a fin de evaluar solamente su //
// comportamiento. Si todo esta bien se podra //
// proseguir con la MNL. //
////////////////////////////////////
    FILE *trans,*ome;
    double radius, zsample, zdisk;
    double enerin, enerout, enerin1, enerout1; // energias totales,
interna y externa
    double tim, factor, rmfrac; // time offset and factor
    double rmm,enerextra,tiempo,ta;
    int cont, ciclo, waitant, car;
    radius=0.05;
    getpar();
    menu();
    waitant=wait;
    wait=0; // wait for plot inside of calibration...
    replace();
    savepar();
    if (nocal==1) { return 1; } // scan, don't calibrate
    initial();
    //dskd=1.99969539*aw2;
    modografico();
}

```

```

//graf(U,DIM,220,400, "beam before the first sample" );
//leetecla(0);
inpute=energy(U,0,rm,rm);
//enerextra=inpute*2.0*deltaT;
kconst=0;

for (ciclo=0; ciclo<ng; ciclo++){
    tim=(ng-ciclo)*deltaT - 3.0* hwem; // time.
    TT[ciclo]=tim;
    factor = exp(-SQR(tim/hwem));
    kconst=kconst+factor;
}
kconst=2.0*kconst*deltaT; //exactly it should be
en=sqrt(M_PI)*hwem
if (ng==1) kconst=sqrt(M_PI)*hwem;
inpute=inpute*kconst;

// guardar U y rm...
rmfrac=rm;
for(cont=0;cont<DIM;cont++) { frac[cont]=U[cont]; } // se guarda U
ener0=0.0; enerout=0.0; enerin=0.0;ener0l=0.0; enerinl=0.0;
eneroutl=0.0; // accumulators

ome=fopen("omega.dat","w");
for (ciclo=0; ciclo<ng; ciclo++)
{
    tim=(ng-ciclo)*deltaT - 3.0 * hwem; // time.
    factor = exp(-0.5*SQR(tim/hwem));
    for(cont=0;cont<DIM;cont++) { U[cont]=frac[cont]*factor; } //
new U
    rm=rmfrac;
    FF[ciclo]=energy(U,0,rm,rm);
    graf(U,DIM,220,400, "before the first sample");
    lw1=refw1;
    lz1=refz1;
    //////////////////////////////////////
    for (cont=0; cont<DIM; cont++){tmpu[cont]=U[cont];}
    rmm=rm;
    //////////////////////////////////////
    //for (cont=0; cont<DIM; cont++){U[cont]=sqrt(0.96)*U[cont];}

    sampleray(U, sampl,n2,10);
    graf(U,DIM,220,400, "after the first sample");

    rhom=range(z2-0.1,1,nbeam);
    Frenel(U,z2-0.1,rhom); // from sample 1 to disk
    graf(U,DIM,220,400, "At the disk");
}

```

```

////////////////////////////////////
P10[ciclo]=energy(U,0,rm,rm);
P11[ciclo]=energy(U,0,dskd/2.0,rm);
P12[ciclo]=energy(U,(dskd/2.0),rm,rm);
////////////////////////////////////

ener0=ener0+P10[ciclo]; // energia total original
enerin=enerin+P11[ciclo];
enerout=enerout+P12[ciclo];

////////////////////////////////////
for (cont=0; cont<DIM; cont++){U[cont]=tmpu[cont];}
rm=rmm;

rhom=range(z2,1,nbeam);
Frenel(U,z2,rhom);
graf(U,DIM,20,450,"at the reference...");
P20[ciclo]=energy(U,0,rm,rm);
P21[ciclo]=energy(U,0,dskd/2.0,rm);
P22[ciclo]=energy(U,(dskd/2.0),rm,rm);
ener01=ener01+P20[ciclo]; // energia en la referencia
enerin1=enerin1+P21[ciclo];
enerout1=enerout1+P22[ciclo];
T1[ciclo]=P11[ciclo]/P21[ciclo];
T2[ciclo]=P12[ciclo]/P22[ciclo];
////////////////////////////////////
}
fclose(ome);

////////////////////////////////////
trans=fopen("trans.dat","w");
for (cont=0; cont < 14; cont++)
{
    fprintf(trans, " %14.16f\n", pums[cont] );
}
for (cont=0; cont < 4; cont++)
{
    fprintf(trans, "%d\n", nums[cont] );
}
fprintf(trans, " %14.16f\n", outs );
for (ciclo=0; ciclo<ng; ciclo++)
{
    fprintf(trans,"%g,%f,%f,%f,%f,%f,%f,%f,%f,%f\n",TT[ng-
1-ciclo]+.00000001065,FF[ng-1-ciclo],P10[ng-1-ciclo],P11[ng-1-
ciclo],P12[ng-1-ciclo],P20[ng-1-ciclo],P21[ng-1-ciclo],P22[ng-1-
ciclo],T1[ng-1-ciclo]/T1[ng-1],T2[ng-1-ciclo]/T2[ng-1]);
}
for (ciclo=0;ciclo<ng;ciclo++)
{
    fprintf(trans,"%g,%f,%f,%f,%f,%f,%f,%f,%f,%f\n",-
TT[ciclo]+.00000001065,FF[ciclo],P10[ciclo],P11[ciclo],P12[ciclo],P20[cic
lo],P21[ciclo],P22[ciclo],T1[ciclo]/T1[ng-1],T2[ciclo]/T2[ng-1]);
}
fclose(trans);

```

```

////////////////////////////////////

        if (ng==1){
            ener0=ener0*kconst;
            enerin=enerin*kconst;
            enerout=enerout*kconst;}
        else{
            ener01=2.0*ener01*deltaT;
            enerin1=2.0*enerin1*deltaT;
            enerout1=2.0*enerout1*deltaT;}
    if (niez==HOLE) { tindex0=enerin1/ener01; }
    if (niez==DISK) { tindex0=enerout1/ener01; }
    closegraph();
    wait=waitant; //restores wait status...
    //// recalculating beam shape on the disk at the peak power once
again
    //// only to be used by getdisk function (assuming no nonlinearity
    for(cont=0;cont<DIM;cont++) { U[cont]=frac[cont]; } // U
before sample again
    rm=rmfrac;
    //lwl=refw2; //rm
before sample again
    //lz1=refz2;
    //sampleray(U,sampl,n2,10);
// going through sample again
    //sample(U,n2);
    //zdisk=z3-zsample;
    rhom=range(z2,1,nbeam);
    Frenel(U,z2,rhom); // from focus to disk...
    if (niez==HOLE) { sdisk=getdisk(1-outs); }
    if (niez==DISK) { sdisk=getdisk(outs); }
    ///////////////////////////////////
    printf("Inside: %e Outside: %e Total energy:
    %e\n",enerin1,enerout1,ener01);
    printf(" T: %12.8lf S: %12.8lf \n",tindex0,1-tindex0);
    printf(" Disk/hole diameter should be: %16.12lf to get S: %e
    \n",sdisk,out);

    return 0; // give option to repeat...
} //calibonce

void rcalib(void)
{
    ///////////////////////////////////
    // repeats Calibrate Once ( calibonce ) until Escape or Scan
    ///////////////////////////////////
    do
    {
        if ( calibonce() ) { return; }
        printf("\n Press - C - to Calibrate, - S - to Scan, - Esc
- to Exit..");
        tecla=leetecla(0);

```

```

        } while ((tecla == 'c') || (tecla == 'C'));
        if (tecla == 27) {exit(0);}
} //rcalib

void scanning(void)
{
// Devuelve:
// Las transmitancias en el regimen no lineal
// Los parametro S y la transmitancia normalizada
// en SCANR[i]
// El valor pico-valle Dpv
// El valor del cambio de fase en el foco
// La irradiancia en el foco
double dz, zs, zpro, zdisk ;
//double ET, EN, DSKT, TO, calib;
double Min, Max, Mtem, Dpv, aux1, aux2;
double enert, enerin, enerout, ltindex ; //ener. total, int y ext
double tim, factor, rmfrac; // time offset and factor
//FILE *ues, *err, *oprhoms;
complex *SCANR;
int i, j, cont, ciclo;
//char mens1[80], arch[13];
getpar();
replace();
SCANR=(complex *)farmalloc(nz * sizeof(complex));
if (SCANR==NULL) {printf("no hay memoria..");leetecla(1);exit(1);}
initial();
modografico();
graf(U, DIM, 220, 400, "beam at the aperture...");
//from aperture to lens
rhom=range(z2, 1, nbeam);
Frenel(U, z2, rhom);
graf(U, DIM, 220, 400, "Before the Lens ");
lens(U, focal2);

for(i=0; i<DIM; i++) { tmpu[i]=U[i]; } rmUL=rm; // se guardan U y
rm
////////// NOW THE SCAN BEGINS //////////////////////////////////////
if (nz < 1) { nz=1; }
if (nz==1) { dz=zstart; } else { dz = -2.0*zstart/(nz-1); }
for (i=0; i<nz; i++)
{
    for(j=0; j<DIM; j++) { U[j]=tmpu[j]; } // se recupera U de tmpu
    rm=rmUL;
    //graf(U, DIM, 20, 400, "after the lens...");
    zs=zstart+dz*i;
    zpro=localz0+zs;
    //printf("STEP %d AT Z = %f ", i, zs);
    rhom=range(zpro, 2, nbeam);
    Frenel(U, zpro, rhom); // from lens to the sample....
    //graf(U, DIM, 120, 400, "Before sample");
    // guardar U...
    rmfrac=rm;
}

```

```

guarda U      for(cont=0;cont<DIM;cont++) { frac[cont]=U[cont]; } // se
enerout=0.0; enerout=0.0; enerin=0.0; // accumulators
for (ciclo=0; ciclo<ng; ciclo++)
{
    tim=(ng-ciclo)*deltaT-3.0* hwem; // time.
    factor = exp(-0.5*SQR(tim/hwem));
    for(j=0;j<DIM;j++) { U[j]=frac[j]*factor; } // new U
    rm=rmfrac;
    ////////// sample //////////////////////
    // calcular la nueva U de acuerdo a la fraccion y

calcular      // el tiempo (ancho de la fraccion)
              sample(U,n2);
              //graf(U,DIM,120,400, "after sample");
              ////////// sample to disk //////////////////
              zdisk=z3-zpro;
              rhom=range(zdisk, 3, nbeam);
              Frenel(U, zdisk, rhom); // from sample to the disk...
              enerout=enerout + energy(U, 0, rm, rm);
              enerin=enerin + energy(U, 0, dskd/2.0, rm);
              enerout=enerout + energy(U, dskd/2.0, rm, rm);
} // fraction loop

if (ng==1){
    enerout=enerout*kconst;
    enerin=enerin*kconst;
    enerout=enerout*kconst;}
else{
    enerout=2.0*enerout*deltaT;
    enerin=2.0*enerin*deltaT;
    enerout=2.0*enerout*deltaT;}
if (niez==HOLE) { ltindex=enerin/enerout; }
if (niez==DISK) { ltindex=enerout/enerout; }
SCANR[i]=complex(ltindex/tindex0,0.0);
}
////////////////////// END SCANNING
//////////////////////
clearkey();
graf3(SCANR,nz,20,400, " z-scan...");
leetecla(1);
////////////////////// FIND MAX. && MIN.
//////////////////////
Max = 1.0;
Min = 1.0;
for (i=0;i<nz;i++)
{
    Mtem = real(SCANR[i]);
    if (Mtem >= Max) {Max = Mtem;}
    if (Mtem <= Min) {Min = Mtem;}
}
Dpv = (Max-Min);
if (nbeam==0) {///aux1 = 2.0*engy/(M_PI*w30*w30);
                //aux1 = aux1/(hwem * sqrt(M_PI));

```

```

        aux1 = IoGAUS;
        aux2 = (n2*wavnum*aux1*sampl);
    }
    if (nbeam==1) { //aux1 = 2.0*engy/(M_PI*W30THB* W30THB);
        //aux1 = aux1/(hwem * sqrt(M_PI));
        aux1 = IoTHB;
        aux2 = (n2*wavnum*aux1*sampl);
    }

    printf("Peak Irradiance at focus Io [W/cm2]:  %14.4f\n", aux1);
    printf("Change in Phase at r = 0  D(Qo) :  %6.18f\n", aux2);
    printf("Peak-Valley distance is:  %14.11f", Dpv); leetecla(1);
    closegraph();
    ////////////////////////////////// S A V I N G   D A T A
    //////////////////////////////////
    datosSR=fopen("scan.dat", "w");
    //fprintf(datosSR, " The results of the scan are:\n\n");
    for (i=0; i<nz; i++)
    {
        zs=zstart+dz*(1.0*i);
        // writedata2(datosSR, SCANR[i], i, zs);
        fprintf(datosSR, "  %14.11f,  %14.11f\n", zs,
real(SCANR[i]));
    }
    fclose(datosSR);
    //farfree(tmpu);
    ////////////////////////////////// E N D
    //////////////////////////////////
}

```

```

////////////////////////////////// LIBRERIA GETPARAM ////////////////////////////////////

char leetecla(int tipo)
{
// Espera para que se toque una tecla, si tipo==1 entonces suena.
  int tecla ;
  if (tipo==1)
  {
      sound(1000); delay(100); nosound();
  }
  tecla=getch();
  if (tecla==0 ) { tecla=getch()+200; }
  return tecla;
}

void clearkey(void)
{
  while ( kbhit() ) leetecla(0); }

void assignmemo(void)
{
//////////////////////////////////
// Se reserva memoria para los arreglos //
// globales usados en este programa //
//////////////////////////////////
  unsigned long int espacio;
  espacio = farcoreleft();
  if (espacio > 32768L + 15000L) { DIM=256; }
  if (espacio > 262144L + 15000L) { DIM=2048; }
  if (espacio > 524288L + 15000L) { DIM=4096; }
  printf(" m ximo num. de puntos: %d \n", DIM);
  espacio= 1L * DIM * sizeof(complex);
  U = (complex *)farmalloc(espacio);
  //phase=(complex *)farmalloc(espacio);
  //UL = (complex *)farmalloc(espacio);
  tmpu = (complex *)farmalloc(espacio);
  espacio= 2L * DIM * sizeof(complex);
  J1 = (complex *)farmalloc(espacio);
  espacio= 2L * DIM * sizeof(double);
  xx = (double *)farmalloc(espacio);
  espacio= 2L * DIM* sizeof(complex);
  FY=(complex *)farmalloc(espacio);
  frac=(complex *)farmalloc(1L * DIM * sizeof(complex));
  T1=(double *)farmalloc(1L * 100 * sizeof(double));
  T2=(double *)farmalloc(1L * 100 * sizeof(double));
  FF=(double *)farmalloc(1L * 100 * sizeof(double));
  TT=(double *)farmalloc(1L * 100 * sizeof(double));
  P10=(double *)farmalloc(1L * 100 * sizeof(double));
  P11=(double *)farmalloc(1L * 100 * sizeof(double));
  P12=(double *)farmalloc(1L * 100 * sizeof(double));
  P20=(double *)farmalloc(1L * 100 * sizeof(double));
  P21=(double *)farmalloc(1L * 100 * sizeof(double));
  P22=(double *)farmalloc(1L * 100 * sizeof(double));
}

```

```

        if ((U==NULL)|| (J1==NULL) || (J1==NULL)
|| (T1==NULL) || (T2==NULL) || (P10==NULL) || (P11==NULL) ||
        (P12==NULL) || (P20==NULL) || (xx==NULL) || (FY==NULL) ||
(frac==NULL) ||
        (P21==NULL) || (P22==NULL))
        { printf("\n THERE IS NOT ENOUGH MEMORY "); exit(1); }
}

```

```

void fillxx (void)
{
// Se llena el arreglo xx[i] usado en fhankel() //
int cont;
x0=(1.0+exp(alpha)) * exp(-alpha*DIM)/2.0;
for (cont=0; cont < DIM; cont++)
{ xx[cont]=x0*exp(alpha*cont); }
for (cont=DIM; cont < 2*DIM; cont++)
{ xx[cont]=0.0 ; }
}

```

```

double alfa(int tam)
{
//////////////////////////////////////
// Esta funcion da los valores del parametro alfa //
// en funcion del numero de puntos de muestreo DIM//
//////////////////////////////////////
switch(tam)
{
case 2:
return 0.6931471805599453;
case 4:
return 0.3822450858399894;
case 8:
return 0.2274724648943759;
case 16:
return 0.1370225782622967;
case 32:
return 0.0819942832400628;
case 64:
return 0.04843797463793869;
case 128:
return 0.02820679286586392;
case 256:
return 0.01619947222642963;
case 512:
return 0.00918699809627926;
case 1024:
return 0.005152369214907255;
case 2048:
return 0.002861645741496277;
case 4096:
return 0.001575985871035693;
case 8192:
return 0.000861576965519784;
}
}

```

```

        default:
        { printf("\n argument for alfa must be power of 2!");
          exit(1);
        }
    }
    return 0.0;
}

```

```

void getpar(void)
{
    ////////////////////////////////////////////////////////////////////
    // Aqui se leen los parámetros del archivo "params"//
    // generado en la ultima corrida del programa //
    ////////////////////////////////////////////////////////////////////

    FILE *params; // *Rhoms;
    int cont; char cadena[40];
    params = fopen("param.DAT", "rt");
    for (cont=0; cont < 14; cont++)
    {
        fscanf(params, "%s", cadena );
        pums[cont] = atof(cadena);
    }
    for (cont=0; cont < 4; cont++)
    {
        fscanf(params, "%s", cadena );
        nums[cont]= atoi(cadena);
    }
    fscanf(params, "%s", cadena );
    outs = atof(cadena);
    /* if (nums[3]==1) {

        pums[11]=(pums[10]*pums[9])/(pums[9]-pums[10]);
        }*/
    fclose(params);
}

```

```

void savepar(void)
{
    ////////////////////////////////////////////////////////////////////
    // Aqui se salvan en "params" las modificaciones hechas//
    // en la presente aplicación //
    ////////////////////////////////////////////////////////////////////

    FILE *params;
    int cont;
    params = fopen("param.DAT", "w");
    for (cont=0; cont < 14; cont++)
    {
        fprintf(params, " %14.16f\n", pums[cont] );
    }
    for (cont=0; cont < 4; cont++)
    {

```

```

        fprintf(params, "%d\n", nums[cont] );
    }
    fprintf(params, " %14.16f\n", outs );
    fclose(params);
}

```

```

void openfiles (void)
{
    ////////////////////////////////////////////////////////////////////
    // Se abren todos los archivos de datos para escribir//
    // nueva informacion                                     //
    ////////////////////////////////////////////////////////////////////
    datosJI=fopen("DATOSJI","w");
    datosFY=fopen("DATOSFY","w");
    datosU=fopen("DATOSU","w");
    datosUT=fopen("DATOSUT","w");
    datosUTF=fopen("DATOSUTF","w");
}

```

```

void closefiles(void)
{
    ////////////////////////////////////////////////////////////////////
    // Se cierran todos los archivos de datos con//
    // la nueva informacion                               //
    ////////////////////////////////////////////////////////////////////
    fclose(datosJI);
    fclose(datosFY);
    fclose(datosU);
    fclose(datosUT);
    fclose(datosUTF);
}

```

```

void showpar(void)
{
    ////////////////////////////////////////////////////////////////////
    // Se muestra en pantalla los parámetros//
    // actuales para el barrido           //
    ////////////////////////////////////////////////////////////////////
    clrscr();
    printf("          THE CURRENT PHYSICAL CONDITIONS ARE: \n");
    printf("1. Pulse Energy (E)      (mj)=%f      ", pums[0]);
    printf("2. Sample Length (L) (cm)=%6.5f \n", pums[1]);
    printf("3. Pulse fwl/em of I(T)=%f      ", pums[2]);
    printf("4. Wavelen (l)      (æm)=%f\n" , pums[3]);
    printf("-----\n");
    printf("-----\n");
    printf("5. Spot hwl/e of I (W0)      (cm)=%f      ", pums[4]);
    printf("6. lens1 fl (cm)=%10.2f\n", pums[5]);
    printf("7. focus pos./lens1 (Z0) (cm)=%7.3f      ", pums[6]);
    printf("8. Distance between lenses (cm)=%5.3f\n", pums[7]);
    printf("9. Position of S1 (Zr)=%6.3f      ", pums[8]);
}

```

```

printf("10. Position of S2 (Zr) =%5.3f\n", pums[9]);
printf("-----\n");
--");
printf("-----\n");
printf("11. second lens f2      (cm)=%6.3f ", pums[10]);
printf("12. Disk dist./lens2 (Z3)(cm)=%6.3f \n", pums[11]);
printf("13. Disk/app. diam. (D) (cm)=%10.8f ", pums[12]);
printf("14. samp2 n2 (esu)= %5.3e \n", pums[13]);
printf("-----\n");
--");
printf("-----\n");
printf("          THE CURRENT NUMERICAL CONDITIONS ARE: \n");
printf("15. (nt) Num of steps in t = %d ", nums[0]);
printf("16. (nr) Num of steps in r = %d \n", nums[1]);
printf("17. (Z, EZ or IEZ)-Scan (0,1,1) = %d ", nums[2]);
printf("18. (Gaussian=0, Top Hat Beam=1) = %d\n", nums[3]);
printf("----- Program Control -----\n");
-");
printf("-----\n");
printf("20. Exit!                21. Scan now, no Calibration\n" );
printf("22. Direct Prop. & Exit: ");
if (dirprop) { printf("YES"); } else { printf("NO "); }
printf("          23. S: %6.4f \n", outs);
printf("24. Wait after plot : ");
if (wait) { printf("YES\n"); } else { printf("NO \n"); }
printf("          T: %16.12lf S: %16.12lf \n ", tindex0, 1-tindex0);
printf("Adjusted disk diam: %16.12lf\n", sdisk);
printf("Input energy to the system is: %8.6f", inpute*1000);
printf(" mj \n");
printf("output energy of the system is: %8.6f", ener0*1000);
printf(" mj \n");
printf("Apperture transmission is: %8.6f", inpute*1000/pums[0]);
printf(" mj \n");
}

void menu(void)
{
// Se despliega "showpar" y se da la opcion a cambios //
int opcion; char sopcion[80];
do
{
showpar();
printf("\n ENTER NUMBER OF ITEM TO BE CHANGED, 0 TO RUN, 20
to Exit...");
do
{gets(sopcion); }
while ( strlen(sopcion) < 1 );
opcion=atoi(sopcion);
if (opcion==20) { savepar(); exit(0); }
//if (opcion==21) { nocal=1; return; }
if
(opcion==22) {if(dirprop==YES) {dirprop=NO;} else {dirprop=YES;}}
if (opcion==23)
{

```

```

        printf(" current value: %f , enter new: ", outs);
        scanf("%lf",&outs);
    }
    if (opcion==24) { if (wait) { wait=NO;} else { wait=YES; }}
    if ((opcion>0)&&(opcion<19))
    {
        if (opcion<15)
        {
            printf(" current value: %f , enter new: ",pums[opcion-
1]);
            scanf("%lf",&pums[opcion-1]);
        }
        if (opcion>14)
        {
            printf(" current value: %d , enter new: ",nums[opcion-
15]);
            scanf("%d",&nums[opcion-15]);
        }
    }
    //if(nums[3]==1){pums[11]=(pums[10]*pums[9])/(pums[9]-
pums[10]);}
    } while (opcion != 0) ;
}

void replace(void)
{
    //////////////////////////////////////
    // Se reemplazan los cambios hechos en los parámetros//
    // para la aplicación //
    //////////////////////////////////////
    engy = pums[0] * 1.0e-3 ;
    sampl = pums[1];
    pulsew = pums[2] * 1.0e-9;
    hwem = 0.5 * pulsew;
    wavlen = pums[3] * 1.0e-4;
    spot = pums[4];
    focall = pums[5];
    zel = pums[6];
    apd = pums[7];
    z1 = pums[8];
    s2 = pums[9];
    focal2 = pums[10];
    z3 = pums[11];
    dskd = pums[12];
    n2 = 1.0e6*pums[13]*40.0*M_PI/(spdl*1.6) ;
    n21 = 1.0e6*1.2e-11*40.0*M_PI/(spdl*1.6);
    ng = nums[0];
    nr = nums[1];
    DIM = nums[1];
    niez = nums[2];
    nbeam = nums[3];
    cof = sqrt(8.0*M_PI/spdl);
    konst = 2.0*M_PI/(cof*cof);
    deltaT = 3.0*hwem/ng;
}

```

```

    np2    = log(1.0*nr)/log(2.0)+0.5;
    nz     = nums[0];
}

void writedata( FILE *arch , complex huge *arrc, int tam, double Rm)
{
// Rutina para guardar datos //
  int cont;
  for (cont=0; cont < tam; cont++)
  {
    fprintf(arch, "%5d  %14.11f  %14.11f  %14.11f  %14.11f \n",
    cont,Rm*xx[cont], real(arrc[cont]),imag(arrc[cont]),abs(arrc[cont]));
  }
}

void sndout(complex huge *U)
{
////////////////////////////////////
// Se evalua la fase de U[i]           //
// Recibe:      U[i]                   //
// Devuelve: La fase del haz           //
//              en el arreglo phase[i] //
////////////////////////////////////

  int i,na,nb,isul,isur,isui;
  double ur,ui,det,ereal,url;
  if (phase==NULL)
    { printf(" no hay memoria para phase..."); leetecla(1);
return; }
  det = 0.0;
  url=real(U[0]);
  for(i=0;i<DIM;i++)
  {
    ur = real(U[i]);
    ui = imag(U[i]);
    isul=1;
    if (url<0.0) {isul=-1;}
    isur=1;
    if (ur<0.0) {isur=-1;}
    isui=1;
    if (ui<0) {isui=-1;}
    if (ui==0) {isui=0;}
    na=isul*isur;
    nb=isul*isui;
    if (na<0) det=det+M_PI*nb;
    if (ur==0) { ereal=0.0; }
    else { ereal = atan(ui/ur)+det; }
    phase[i] = complex(ereal, 0.0);
    url=ur;
  }
  //graf(phase,DIM,20,450,"the Phase");
}

```

```

void writedata2( FILE *arch , complex arrc,int cont,double Rm)
{
// Rutina para guardar datos //
    fprintf(arch, "%5d  %14.11f  %14.11f  %14.11f\n",
        cont,Rm, real(arrc), imag(arrc));
}

void initial(void)
{
////////////////////////////////////////////////////////////////////
// Se generan tanto los haces Gaussianos como //
// los THB, se evalua la potencia total del //
// haz y se da la energia de transmision en el//
// regimen lineal a travez de A1,A2 y D segun //
// sea el caso //
////////////////////////////////////////////////////////////////////
/*
    z0: rango de rayleigh
    w10: spot en el foco
*/
    double z10, w10,w1, eamp, ereal, eimag, temp;
    double spot1,z0,z,peakf; //pow;
    double invcrv, fixfaze, rr, intensr, faze;
    int cont, Tint;

    alpha=alfa(DIM); //finds value of alpha.....
    fillxx(); //fills xx array.....

    w0 = spot* sqrt(2.0);
    guslms(w0,zel,focall,z1, &w1, &z10, &w10);

    z1=z1*((M_PI*w10*w10)/wavlen)+z10; //distancia de L1 a la muestra
1
    w1 = w10 * sqrt(1 +SQR((z1- z10)/((M_PI*w10*w10)/wavlen)));
    zstart=z10-z1;
    z2=10+zstart; //distancia de la muestra 1 al disco.
    aw1 = w1;
    //printf("ancho de la cintura en la primera muestra %f\n ",aw1);
    //leetecla(0);
    localw0 =w10;
    refw1=localw0;
    localz0 = z1-z10; // Distancia de la muestra 1 al foco de L1
    refz1=localz0;
    spot1 = w1/sqrt(2.0);
    z0 = M_PI*SQR(w10)/wavlen;
    z = z1 - z10;
    //pow = engy/hwem;
    peakf = engy/(M_PI*spot1*spot1);
    peak = peakf/(hwem * sqrt(M_PI));
    if (z == 0.0) {invcrv=0.0; }
    else {invcrv = 1.0/(z+z0*z0/z); }
    wavnum = 2.0*M_PI/wavlen;
}

```

```

fixfaze = wavnum*z-atan(z/z0);
if (nbeam==GAUSSIAN) { rm=3.0*awl; }
                    else { rm=0.5*apd; }
for (cont=0; cont<DIM; cont++)
{
    rr=rm*xx[cont];
    if ( rr< apd*0.5 )
    {
        intensr=peak*exp(-SQR(rr/spot1));
        faze=fixfaze+wavnum*0.5*rr*rr*invcrv;
        if (faze > 2*M_PI)
        {
            temp=(faze/2*M_PI);
            Tint=temp;
            faze=faze-Tint*2*M_PI;
        }
        eamp=cof*sqrt(intensr);
        ereal=eamp*cos(faze);
        eimag=eamp*sin(faze);
        U[cont]=complex(ereal,eimag);
    }
    else { U[cont]=complex(0.0,0.0); }
}
U[DIM-1]=complex(0.0,0.0); // siempre!
guspro(localw0,0,10,&aw2);
//printf("tamano del haz en la segunda lente %f\n", aw2);
//leetecla(0);
localz0 = localz0 + z2;
guslms(localw0,localz0,focal2,z3,&aw3,&z30,&w30);
//printf("tamano del haz en la segunda muestra %f\n", aw3);
//leetecla(0);
localw0=w30;
refw2=localw0;
localz0=z30;
rayran=M_PI*SQR(localw0)/wavlen;

refz2=-zstart;
aw3=w30*sqrt(1+(10/rayran)*(10/rayran));
//if (nbeam==GAUSSIAN) {zstart=-5.0*rayran;}else {zstart=-
250.0*rayran;}
//if (nbeam == GAUSSIAN)
//{
    if(apd < 6.0*awl)
    {
        printf("WARNING: the first apperture is too small for
");
        printf("the gaussian beam to pass\n");leetecla(1);
    }
    printf("Peak input intensity [w/cm2] = %f\n",peak);
    printf("Peak input fluence [j/cm2] = %f\n",peakf);
//}
} //initial

```

```

//////////////////////////////// LIBRERIA ENGINE //////////////////////////////////
void raylay(complex huge *U,double z,double par1, double par2)
{
// Recibe: U[i] y la distancia z por propagar //
// Devuelve: El haz propagado por medio de la //
// FFT y IFFT de U[i] en el mismo U[i] //
////////////////////////////////
    int i;
    double c,cc,RR,ereall,eimagl;
    rhom=par1;
    fhankel(U,J1,DIM);
    c=2*M_PI*z/wavlen;
    for(i=0;i<DIM;i++)
    {
        RR=rhom*xx[i];
        cc=c*sqrt(1-SQR(wavlen*RR));
        ereall=cos(cc);
        eimagl=sin(cc);
        U[i]=U[i]*complex(ereall,eimagl);
    }
    rm=rhom;
    rhom=par2;
    fhankel(U,J1,DIM);
    rm=rhom;
} //raylay

float bessj1(float x)
{
////////////////////////////////
// Aqui se generan los valores de la funcion de Bessel //
// de orden 1, requeridos en ComputeJ1() //
////////////////////////////////
    float ax,z;
    double xx,y,ans,ans1,ans2;

    if ((ax=fabs(x)) < 8.0)
    {
        y=x*x;
        ans1=x*(72362614232.0+y*(-7895059235.0+y*(242396853.1
            +y*(-2972611.439+y*(15704.48260+y*(-30.16036606))))));
        ans2=144725228442.0+y*(2300535178.0+y*(18583304.74
            +y*(99447.43394+y*(376.9991397+y*1.0))));
        ans=ans1/ans2;
    }
    else
    {
        z=8.0/ax;
        y=z*z;
        xx=ax-2.356194491;
        ans1=1.0+y*(0.183105e-2+y*(-0.3516396496e-4
            +y*(0.2457520174e-5+y*(-0.240337019e-6))));
        ans2=0.04687499995+y*(-0.2002690873e-3
            +y*(0.8449199096e-5+y*(-0.88228987e-6
            +y*0.105787412e-6)));
    }
}

```

```

        ans=sqrt(0.636619772/ax)*(cos(xx)*ans1-z*sin(xx)*ans2);
        if (x < 0.0) ans = -ans;
    }
    return ans;
} // bessj1

void FFT (complex huge *com,int nn,int isign)
{
    ////////////////////////////////////////////////////////////////////
    // Aqui se calcula la FFT de cualquier arreglo *com que llegue //
    // con dimension nn. Si isign=1 (FFT) y si isign=-1 (IFFT) //
    ////////////////////////////////////////////////////////////////////
    unsigned long n,mmax,m,j,istep,i;
    double wtemp,wr,wpr,wpi,wi,theta;
    double tempr,tempi, temp ;
    double huge *data;

    data = (double *) com; //shares same memory as the complex
    data = data-1; // data begins on [1], not [0]
    n=nn << 1;
    j=1;
    for (i=1;i<n;i+=2)
    {
        if (j > i)
        {
            temp=data[j];
            data[j]=data[i];
            data[i]=temp;
            temp=data[j+1];
            data[j+1]=data[i+1];
            data[i+1]=temp;
        }
        m=n >> 1;
        while (m >= 2 && j > m)
        {
            j -= m;
            m >>= 1;
        }
        j += m;
    }
    mmax=2;
    while (n > mmax)
    {
        istep=mmax << 1;
        theta=isign*(2.0*M_PI/mmax);
        wtemp=sin(0.5*theta);
        wpr = -2.0*wtemp*wtemp;
        wpi=sin(theta);
        wr=1.0;
        wi=0.0;
        for (m=1;m<n;m+=2)
        {
            for (i=m;i<n;i+=istep)
            {

```

```

        j=i+mmax;
        tempr=wr*data[j]-wi*data[j+1];
        tempi=wr*data[j+1]+wi*data[j];
        data[j]=data[i]-tempr;
        data[j+1]=data[i+1]-tempi;
        data[i] += tempr;
        data[i+1] += tempi;
    }

    wr=(wtemp=wr)*wpr-wi*wpi+wr;
    wi=wi*wpr+wtemp*wpi+wi;
}
mmax=istep;
}
if (isign== -1)
{
    temp = nn*1.0;
    for(i=0;i<2*nn+1;i++) { data[i]=data[i]/temp; }
}
} // FFT ends

double omega(complex huge *U, double rm)
// given a gaussian beam and rm, finds the radius where
// the field is this fraction of the maximum (center).
{
    double wf, wr;
    int cont = -1;
    //wf=exp(1.0);
    wf=abs(U[0])/ 2.718281828;
    do
    {
        cont++;
    } while ( abs(U[cont]) > wf );
    // sacar proporción entre 2 puntos sucesivos
    wr= xx[cont]-(wf-abs(U[cont])) *(xx[cont]-xx[cont-1]) /
        (abs(U[cont-1])-abs(U[cont])) ;
    return wr*rm;
} // omega

void beamsize (complex huge *U, double W, double par)
{
    ////////////////////////////////////////////////////////////////////
    // Se evalúan los tamaños de las cinturas //
    // medias del haz U[i] y se comparan con //
    // los valores teóricos dados por las //
    // fórmulas de propagación Gaussiana //
    // Recibe: U[i] y el valor teórico de la //
    // cintura W //
    // Devuelve: Los tamaños de las cinturas //
    ////////////////////////////////////////////////////////////////////

    int i,j,k;
    double M,Mlow,Mup,HWe2,HW2,Me,Me2,M2,temp,Enx;

```

```

if (nbeam == GAUSSIAN)
{
    HWe2=sqrt(2.0)*W;
    HW2=sqrt(0.69314718)*W;
    printf("Parameters are [cm]: rm = %f, rhom =
%f\n\n",rm*par,rhom);
    printf("The Gaussian propagation theory gives:\n");
    printf("HW1/eM is: %f, HW1/e2M is: %f, HW1/2M is:
%f\n\n",W,HWe2,HW2);
    M=abs(U[0]);
    Me=M/(2.718281828);
    Me2=Me/(2.718281828);
    M2=M/2.0;

    for (j=0;j<3;j++)
    {
        i=0;
        if(j==0) {temp=Me;}
        if(j==1) {temp=Me2;}
        if(j==2) {temp=M2;}

        do
        {
            Mup=abs(U[i]);
            Mlow=abs(U[i+1]);
            i++;
        } while ((Mup>=temp));
        k=i-2;
        Mup=abs(U[k]);
        Mlow=abs(U[k+1]);
        printf("The element i of the array corresponding to the
next ");
        printf("averaged half-width (XX[i]) is: %d\n", k);

        if ((Mup==temp) || fabs(Mup-temp)<0.00005)
            {Enx=rm*par*xx[k];}
        else if ((Mlow==temp) || fabs(temp-Mlow)<0.00005)
            {Enx=rm*par*xx[k+1];}
        else
            {Enx=rm*par*(xx[k]+(xx[k+1]-xx[k])/2.0);}

        if(j==0)
            {printf("The value by our program for HW1/eM is
[cm]: %f\n", Enx);}
        if(j==1)
            {printf("The value by our program for HW1/e2M is
[cm]: %f\n", Enx);}
        if(j==2)
            {printf("The value by our program for HW1/2M is
[cm]: %f\n", Enx);}
    }
}
if (nbeam == THB)
{

```

```

                printf("Parameters are [cm]: rm = %f, rhom =
%f\n\n",rm*par,rhom);
            }
} //beamsize

void lens(complex huge *U, double F)
{
////////////////////////////////////////////////////////////////////
// Se evalua la propagacion de U[i] a travez //
// de una lente delgada //
// Recibe: U[i] y la distancia focal F de la //
// lente //
// Devuelve: El haz propagado en la lente //
// en el mismo arreglo U[i] //
////////////////////////////////////////////////////////////////////

    int i;
    double c,rr,ereal,eimag;

    c=(M_PI*1.0)/(wavlen*F);
    for (i=0;i<DIM;i++)
    {
        rr=rm*xx[i];
        ereal=cos(c*rr*rr);
        eimag=sin(c*rr*rr);
        U[i]=U[i]*complex(ereal,-eimag);
    }
}

void sample(complex huge *U,double nl)
{
////////////////////////////////////////////////////////////////////
// Se evalua la propagacion de U[i] a travez //
// de la muestra no lineal de espesor delgado //
// Recibe: U[i] y el indice de refraccion no //
// lineal n2- //
// Devuelve: El haz propagado en la MNL //
// en el mismo arreglo U[i] //
////////////////////////////////////////////////////////////////////
    double cc , temp, Tr, Ti;
    int i;

    cc = 2.0*M_PI*sampl*nl/(wavlen*cof*cof);
    for (i=0; i<DIM; i++)
    {
        temp=SQR(abs(U[i]));
        Tr=cos(cc*temp);
        Ti=sin(cc*temp);
        U[i]=U[i]*complex(Tr,Ti);
    }
}

void sampleray(complex huge *U, double z,double nl, int steps /*, double
rhomx*/ )

```



```

double alpha, x0, k0, temp1, temp2;
int cont ;
unsigned long int espacio;
//complex huge *FY ;
complex tempc;
computeJI();
espacio= farcoreleft();
espacio= 2L * DIM* sizeof(complex);
//FY=(complex huge *)farmalloc(espacio);
if (FY==NULL)
{
    printf("\n %ld bytes de memoria para FY en fhankel \n",
espacio);
    printf(" Hay solamente %ld ....\n", farcoreleft() );
    leetecla(1);
    closegraph();
    exit(1);
}
alpha=alfa(tam);
x0=(1.0+exp(alpha)) * exp(-alpha*DIM)/2.0;
temp1=2.0 * exp(alpha) + exp(2.0*alpha);
temp2=(1.0+exp(alpha))*(1.0+exp(alpha))*(1.0-exp(-2.0*alpha));
k0=temp1/temp2;
// calculates
FY.....
FY[0] = k0 * (U[0]-U[1]) * exp(alpha*(1-tam));
temp1 = exp(alpha*(1-tam))/x0;
for ( cont=1; cont < tam-1; cont++)
    {FY[cont]=(U[cont]-U[cont+1])*xx[cont]*temp1;}
FY[tam-1]=U[tam-1];
for(cont=tam; cont < 2*tam; cont++)
    {FY[cont]=complex(0.0,0.0);}
FFT(FY, 2*tam, 1);
// multiplies FY by J1 term by term and stores in FY
.....
for( cont=0; cont < 2*tam -10; cont++)
{
    tempc= FY[cont] * J1[cont];
    FY[cont]=tempc;
}
for( cont=2*tam-10; cont < 2*tam ; cont++)
{
    tempc= FY[cont] * J1[cont];
    FY[cont]=tempc;
}
//FFT of the product .....
FFT(FY, 2*tam, 1);
temp1=rm/rhom;
for( cont=0; cont < tam-1; cont++)
{
    U[cont]=FY[cont]*temp1/xx[cont];
}
} //fhankel

```

```

double range(double zr, int prop, int type)
{
    double rhomx, rmx, zs;
    if (type==GAUSSIAN)
    {
        if (prop==1)
            { rmx=3.5*aw2; }
        else if (prop==2)
            { zs=localz0-zr;
            rmx=5.0*localw0*sqrt(1.0+(zs*zs)/(rayran*rayran)); }
        else if (prop==3)
            { rmx=4*aw3; }
    }
    else // THB
    {
        if (prop==1)
            { rmx=2*pow((4*wavlen*zr*zr*zr)/(10*M_PI),0.25)-apd/2.0; }
        else if (prop==2)
            { rmx=19.52*wavlen*focal2/apd; }
        else if (prop==3)
            { rmx=apd*z3/z2; }
    }
    rhomx=rmx/(wavlen*zr);
    return rhomx;
}

```

```

void Frenel (complex huge *U, double z,double par)
{
    int i;
    double rr, ereall, ereal2, eimag1, eimag2, c, cc;
    complex den,temp, temp1, temp2;
    rhom=par;
    //rhom=1.0/(2.0*xx[0]*rm);
    //computeJI(); //calculates J1 array.....
    c =(M_PI*1.0)/(z*wavlen);
    for(i=0;i<DIM;i++)
    {
        rr=rm*xx[i];
        ereall=cos(c*rr*rr);
        eimag1=sin(c*rr*rr);
        U[i]=U[i]*complex(ereall,eimag1);
    }
    fhankel(U,J1,DIM);
    cc=(2.0*M_PI*z)/wavlen;
    den=complex(0.0,wavlen*z);
    rm=wavlen*z*rhom;
    for(i=0;i<DIM;i++)
    {
        rr=xx[i]*rm;
        ereall=cos(c*rr*rr);
        eimag1=sin(c*rr*rr);
    }
}

```

```

    ereal2=cos(cc);
    eimag2=sin(cc);
    temp1=complex(ereall1,eimag1);
    temp2=complex(ereal2,eimag2);
    temp=temp1*temp2/den;
    U[i]=U[i]*temp;
}
} //frenel

double energy(complex huge *U, double rs, double re, double mrad)
{
// Se calcula la energia de cualquier
// haz de simetria radial, se reciben
// los limites de integracion ne y ns
// y el campo optico U[i]
// Se da el valor de la energia "energy"
complex Ue,Us;
double I , Energy,r,dr,dre,drs,cors,core;
int i,ns,ne;
if (rs<=0.0) {
    ns=0;
    dr=xx[0];
    cors=konst * SQR(abs(U[0])) * xx[0] * mrad
* dr * 0.5;
}
else {
    i=-1;
    do{
        i=i+1;
        r=mrad*xx[i];
    }while(r<=rs);
    ns=i;
    drs=xx[ns]*mrad-rs;
    dr=(xx[ns]-xx[ns-1])*mrad;
    Us=U[ns]+drs*(U[ns-1]-U[ns])/dr;
cors=konst*((SQR(abs(Us))*rs+SQR(abs(U[ns]))*xx[ns]*mrad)/2.0)*drs;
}
if (re>=mrad) {
    ne=DIM-1;
    core=0;
}
else {
    i=-1;
    do{
        i=i+1;
        r=mrad*xx[i];
    }while(r<=re);
    ne=i;
    dre=xx[ne]*mrad-re;
    dr=(xx[ne]-xx[ne-1])*mrad;
}
}

```



```

    double z2, z0;
    z2= zx+zcom;
    z0=(M_PI*w10*w10)/wavlen;
    *wx= w10*sqrt(1+SQR(z2/z0));
}

double energy_circle(complex huge *U,double mrad,double center)
{
    double cors, arg, M, dTETA, dr, I, r, r1, r2, E, TETA, radius, centro, i1, i2;
    int B, TN, i, f1, f2;
    centro=center;
    radius=0.05;
    E=0.0;
    I=0.0;
    TETA=0.0;
    TN=100;
    if(radius>=centro)
    {
        dTETA=M_PI/TN;
    }
    else
    {
        dTETA=((asin(radius/centro))/TN);
    }
    M=SQR(radius);
    for(B=0;B<TN;B++)
    {
        f1=1;
        f2=1;
        TETA=dTETA*B;
        arg=M-SQR(centro*sin(TETA));;
        r2=centro*cos(TETA)+(sqrt(arg));
        r1=centro*cos(TETA)-(sqrt(arg));
        for(i=0;i<DIM;i++)
        {
            r=mrad*xx[i];
            if((r>r1) & (f1))
            {
                i1=i; f1=0;
            }
            if((r>r2) & (f2))
            {
                i2=i; f2=0;
            }
            if(f2)
            {
                i2=DIM-1;
            }
        }
        for (i=i1;i<i2;i++)
        {
            dr=(xx[i+1]-xx[i])*mrad;

```

```
        I =I +
((SQR(abs(U[i]))*xx[i]+SQR(abs(U[i+1]))*xx[i+1])/2.0)*mrad*dr*dTETA;
    }
    }
    if(radius>centro){cors=konst * SQR(abs(U[0])) * xx[0] * mrad * dr *
0.5;}
    if(radius==centro){cors=konst * SQR(abs(U[0])) * xx[0] * mrad * dr *
0.25;}
    if(radius<centro){cors=0.0;}
    E=2.0*I/(cof*cof)+cors;
    return E;
}
//energy_circle
```

```

////////////////////////////////// LIBRERIA PLOTT ////////////////////////////////////
//extern long unsigned int libre;

#ifndef GRAPH
    // Juego de funciones vacias, solo para poder compilar.
    void modografico(){}
    void graf(complex huge *arrc, int tam, int xx, int yy, char
*mens){}
    void graf2(complex huge *arrc, int tam, int xx, int yy, char
*mens){}
    void graf3(complex huge *arrc, int tam, int xx, int yy, char
*mens){}
    void closegraph(){}
    void outtextxy(int x, int y, char far *textstring){}
    void grafcomp2(complex huge *arrc,int tam){}
    void constrnum(char *cadi, char *mens, double num){}

#else

    #include <graphics.h>

void constrnum(char *cadi, char *mens, double num1, double ener)
{
    char cnum[30]; //double ener;
    strcpy(cadi, mens);
    gcvt(num1, 8, cnum);
    strcat(cadi,cnum);
    strcat(cadi," energy= " );
    //ener=energy(U,0,DIM-1);
    gcvt(ener, 8, cnum);
    strcat(cadi,cnum);
    //return mens;
}

void modografico()
{
// SE INICIALIZA EL MODO GRAFICO DE C++ //
    int gdriver = DETECT, gmode;
    initgraph(&gdriver, &gmode, "");
    //libre=farcoreleft();
}

void grafcomp(complex huge *arrc,int tam)
{
//////////////////////////////////
// Se grafica el modulo de cualquier arreglo *arrc de //
// numeros complejos que representan una funcion discreta//
// de tamaño "tam" //
//////////////////////////////////
    int cont, y, maxx, maxy;
    complex tempc; char cadena[30];
    float x, maxv, minv, yspan, vabs;
    cleardevice();

```

```

setcolor(WHITE);
rectangle(0,0,getmaxx(),getmaxy());
// encuentra los valores maximos y minimos
tempc=*arrc;
vabs=abs(tempc);
maxv=vabs; minv=vabs;
for (cont=1; cont<tam; cont++)
{
    tempc=arrc[cont];
    vabs=abs(tempc);
    if (vabs > maxv) { maxv=vabs; }
    if (vabs < minv) { minv=vabs; }
}
gcvt(maxv,10, cadena);
outtextxy(getmaxx()/2, 20, cadena);
gcvt(minv,10, cadena);
outtextxy(getmaxx()/2, getmaxy()-20, cadena);
if ( minv < 0.0 ) { minv=0.0; }
maxx=getmaxx()-4; maxy=getmaxy()-4;
tempc= *arrc;
vabs=abs(tempc);
yspan=maxv-minv;
if (yspan==0.0) { maxv=2.0*maxv; yspan=maxv-minv; }
//x= 2;
x=(xx[0]*maxx)+2;
//y= maxy-maxy*(vabs-minv)/(yspan)+2;
y= maxy- maxy*(abs(arrc[0])-minv)/(yspan)+2;
setcolor(YELLOW);
moveto(x,y);
for (cont=1; cont<tam; cont++)
{
    x=(xx[cont]*maxx)+2;
    tempc=arrc[cont];
    y= maxy- maxy*(abs(tempc)-minv)/(yspan)+2;
    lineto(x,y);
}
}

void grafcomp3(complex huge *arrc,int tam)
{
    //////////////////////////////////////
    // Se grafica el modulo de cualquier arreglo *arrc de //
    // numeros complejos que representan una funcion discreta//
    // de tamaño "tam", pero sin evitar deformacion //
    //////////////////////////////////////.

    int cont, y, maxx, maxy;
    complex tempc; char cadena[30];
    float x, maxv, minv, yspan, vabs, maxxf;
    cleardevice();
    setcolor(WHITE);
    rectangle(0,0,getmaxx(),getmaxy());
    // encuentra los valores maximos y minimos

```

```

tempc=*arcc;
vabs=abs(tempc);
maxv=vabs; minv=vabs;
for (cont=1; cont<tam; cont++)
{
    tempc=arcc[cont];
    vabs=abs(tempc);
    if (vabs > maxv) { maxv=vabs; }
    if (vabs < minv) { minv=vabs; }
}
gcvt(maxv,10, cadena);
outtextxy(getmaxx()/2, 20, cadena);
gcvt(minv,10, cadena);
outtextxy(getmaxx()/2, getmaxy()-20, cadena);
if ( minv < 0.0 ) { minv=0.0; }
maxx=getmaxx()-4; maxy=getmaxy()-4;
maxxf=maxx;
tempc= *arcc;
vabs=abs(tempc);
yspan=maxv-minv;
if (yspan==0.0) { maxv=2.0*maxv; yspan=maxv-minv; }
x= 2;
y= maxy-maxy*(vabs-minv)/(yspan)+2;
setcolor(YELLOW);
moveto(x,y);
for (cont=1; cont<tam; cont++)
{
    x=(cont*maxxf)/tam+2;
    tempc=arcc[cont];
    y= maxy- maxy*(abs(tempc)-minv)/(yspan)+2;
    lineto(x,y);
}
} // grafcomp3

void grafcomp2(complex huge *arcc,int tam)
{
//////////
// Se grafican por separado tanto la parte Real como//
// la parte Imaginaria de cualquier arreglo *arcc //
// de numeros complejos que representan una funcion //
// discreta de tamaño "tam" //
//////////

int cont, x, y, maxx, maxy;
char cadena[30];
float maxv, minv, yspan, val;
cleardevice();
setcolor(WHITE);
rectangle(0,0,getmaxx(),getmaxy());
// encuentra los valores maximos y minimos
maxv= 0.0; minv=0.0;
for (cont=0; cont<tam; cont++)
{
    val=real(arcc[cont]);

```

```

        if (val > maxv) { maxv=val; }
        if (val < minv) { minv=val; }
    }
    for (cont=0; cont<tam; cont++)
    {
        val=imag(arrc[cont]);
        if (val > maxv) { maxv=val; }
        if (val < minv) { minv=val; }
    }
    gcvt(maxv,10, cadena);
    outtextxy(getmaxx()/2, 20, cadena);
    gcvt(minv,10, cadena);
    outtextxy(getmaxx()/2, getmaxy()-20, cadena);
    if ( minv > 0.0 ) { minv=0.0; }
    maxx=getmaxx()-4; maxy=getmaxy()-4;
    yspan=maxv-minv;
    if (yspan==0.0) { maxv=2.0*maxv; yspan=maxv-minv; }
    setcolor(YELLOW);
    x= 2;
    y= maxy-maxy*(real(arrc[0])-minv)/(yspan)+2;
    moveto(x,y);
    for (cont=1; cont<tam; cont++)
    {
        x=(xx[cont]*maxx)+2;
        y= maxy- maxy*(real(arrc[cont])-minv)/(yspan)+2;
        lineto(x,y);
    }
    setcolor(GREEN);
    x= 2;
    y= maxy-maxy*(imag(arrc[0])-minv)/(yspan)+2;
    moveto(x,y);
    for (cont=1; cont<tam; cont++)
    {
        x=(xx[cont]*maxx)+2;
        y= maxy- maxy*(imag(arrc[cont])-minv)/(yspan)+2;
        lineto(x,y);
    }
}

```

```

void graf(complex huge *arrc, int tam, int xx, int yy, char *mens)
{

```

```

    int tecla;
    grafcomp(arrc,tam);
    outtextxy(xx, yy, mens);
    if (wait)
    {
        tecla=leetecla(1);
        if (tecla==27) { closegraph(); exit(2); }
    }
}

```

```

void graf3(complex huge *arrc, int tam, int xx, int yy, char *mens)
{

```

```

    int tecla;

```