

# Curso de SolTrace

Isaías Moreno Cruz

9 de septiembre de 2018

## 1. Introducción

Una forma de calcular la irradiancia solar concentrada es la simulación de *trazado de rayos*, el cual se puede llevar a cabo mediante dos métodos:

- El método de *Monte Carlo*, basado en rayos al azar (método estadístico)
- El método de *Convolución*, basado en la convolución de distribuciones.

El método de Monte Carlo usa rayos al azar que vienen de una superficie, y luego determina cuales de ellos llegan a una segunda superficie. La irradiancia sobre una superficie elemental es proporcional al número de impactos de rayos. En el caso de un concentrador con una reflexión, este algoritmo es usado dos veces, primero entre el Sol y la superficie reflectiva con una distribución energética correspondiente de la forma del sol, después entre la faceta del heliostato y el receptor con las leyes estadísticas del error de distribución relacionado con los defectos de las facetas. La precisión y el tiempo de cálculo aumentan con el número de rayos y la complejidad de la geometría. En el método de convolución (o cono óptico), los rayos reflejados de los espejos elementales son considerados con un cono de error calculado por la convolución de la distribución normal Gaussiana correspondiente a cada error (forma del Sol y los errores del heliostato). Una comparación sistemática muestra que se obtienen resultados muy similares con ambos métodos de trazo de rayos, aunque el Método de Monte Carlo es más versátiles [3].

## 2. SolTrace

**SolTrace** es un software que utiliza el trazado de rayos por el método de Monte Carlo, desarrollado por el NREL, que puede usarse para analizar

y diseñar sistemas ópticos de potencia de energía solar. El cual puede ser adaptado a las principales tecnologías de conversión de energía térmica solar a electricidad y también a hornos solares. El sistema óptico esta organizado en etapas en un sistema de coordenadas globales: las etapas son secciones de la geometría óptica que exitosamente es golpeada por los rayos en su trayectoria del sol a el receptor final. Las etapas pueden ser ópticas (interacción física con los rayos) o virtuales (usadas para determinar la posición de los rayos o el mapa de flujo en cualquier parte del sistema sin interacción con él). Estas etapas estan compuestas de elementos con su propia apertura, forma, y propiedades ópticas (errores ópticos, error de especularidad, reflectividad, transmitividad, y refracción). Cada elemento esta descrito en un sistema coordinado asociado con su etapa. A diferencia de las otras herramientas, atenuación atmosférica y errores de seguimiento no se incluyen directamente pero pueden ser tomadas en cuenta, respectivamente, en la reflectividad del espejo o en su error de pendiente [6]. Los resultados pueden ser mostrados y guardados como gráficas, mapas de flujo y gráficas de desempeño, además de tener una interface amigable con el usuario.

La Figura 1(a) muestra las etapas necesarias para relizar una simulación en SolTrace, es decir:

- *Sun Shape*
- *Optical Properties*
- *System Stages*
- *Trace Options*
- *Results*

## 2.1. Forma Solar

En el apartado de la forma solar (*sun shape*) podemos indicar la **dirección del vector solar** mediante *coordenadas globales*, que identificaremos como  $\hat{s}$ , o al ingresar los parámetros de *latitud, día y hora*.

En el caso de que nosotros ingresemos el vector solar  $\hat{s}$ , nos permite definir nuestro sistema de referencia, es decir, que  $x$ ,  $y$  y  $z$  estarán referenciadas según nuestro propio interés. En este caso, y para todo el presente manual, los sistemas de referencia serán  $x$  hacia el este,  $y$  hacia el norte y  $z$  hacia el cenit. Cuando ingresamos la latitud, el día y la hora entonces el sistema de referencia estará dado por el sistema global de SolTrace que considera  $x$  hacia el oeste,  $z$  hacia el norte y  $y$  hacia el cenit.

Una vez que se haya indicado la dirección del vector solar hay que indicar la forma solar. La forma solar es la distribución de intensidad de los rayos solares que llegan a la superficie de la Tierra como función de su ángulo con respecto al centro del disco solar. Esta forma no es constante en el tiempo y tampoco es independiente de la localización geográfica, ya que se ve afectada por las condiciones climáticas. Existen diferentes formas solares: el sol uniforme (*pillbox*), el sol Gaussiano, y el sol real [1].

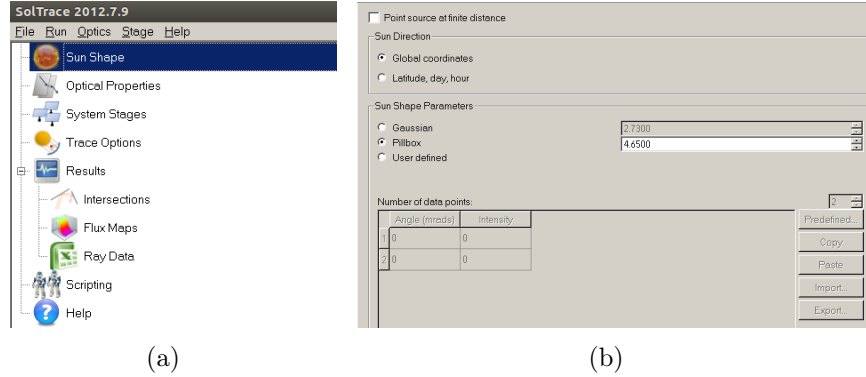


Figura 1: Del lado izquierdo se muestra el índice del programa de SolTrace y del izquierdo las opciones para definir la forma solar.

### 3. Propiedades ópticas

En un concentrador solar, varios factores estadísticos independientes contribuyen al error óptico: errores de pendiente<sup>1</sup>, ausencia de la especularidad ideal, errores de seguimiento, deformaciones y desplazamientos del receptor.

Se considera generalmente que los errores pueden ser representados adecuadamente por distribuciones de probabilidad Gaussiana [4]. El error global es una combinación de los diferentes errores y su dispersión estándar es una combinación en cuadratura de los errores individuales, Ec.1.

$$\sigma_{\text{óptica}}^2 = \sigma_{\text{especular}}^2 + 4\sigma_{\text{pendiente}}^2 + \sigma_{\text{seguimiento}}^2 \quad (1)$$

$\sigma_{\text{pendiente}}$  es multiplicado por 2 debido a la ley de Snell; en reflectores Fresnel  $\sigma_{\text{seguimiento}}$  debe ser también multiplicada por 2.

<sup>1</sup>El error de pendiente engloba los errores macroscópicos.

El ancho total del rayo  $\sigma_{\text{total}}$  es obtenida al agregar el semi-ángulo del disco solar.

$$\sigma_{\text{total}}^2 = \sigma_{\text{optica}}^2 + \sigma_{\text{sol}}^2 \quad (2)$$

En la sección de propiedades ópticas (*Optical Properties*) del índice principal del programa podemos definir las propiedades ópticas de nuestros materiales a utilizar.

- Reflectancia
- Transmitancia
- Error de pendiente
- Error especular
- Tipo de error

Figura 2: Propiedades ópticas.

Las propiedades ópticas dependerán del material del concentrador y del receptor que estemos utilizando.

## 4. Etapas del trazado de rayos

Es en esta sección donde podemos definir las características de nuestro concentrador y del receptor. Es posible definir más de una etapa, pero estas etapas deben de ser subsecuentes. En la sección de etapas podemos distinguir tres apartados. El primero de ellos es *Stage Properties* en donde podemos

asignar un nombre a la etapa y en donde esta seleccionada la etapa *Multiple hits per ray* que nos indica que en esa etapa el rayo puede incidir varias veces; la opción *Virtual stage* es utilizada cuando se desea saber que esta ocurriendo en una intervalo ente el concentrador y el receptor pero sin afectar la trayectoria de los rayos. Otra sección es *Global Coordinates*, puede ser utilizada para generar sistemas de referencia locales. En sección de *Element Editing* podemos definir los elementos que estrán interactuando en el trazado de rayos, condentradores o receptores.

- *Insert*. Puedes incentar elementos necesarios para la simulación
- *Append*. Te permite agregar elementos
- *Delete*. Como su nombre lo indica, puedes eliminar elementos

Una vez que agregas los elementos deseados pudes ir modificando las propiedades de cada uno de ellos. Indicando su posición  $(x, y, z)$ , su *Aimpoint* (ver 4.1), la apertura, el tipo de superficie, la interacción del elemento y sus propiedades ópticas.

En	X-Coord	Y-Coord	Z-Coord	X-AimPt	Y-AimPt	Z-AimPt	Z-Rot	Aperture	Surface	Interaction	Optics
<input checked="" type="checkbox"/>	0	0	0	0	0	1	0	r-0,0,0,0,0,0,0	f-0,0,0,0,0,0,0	Reflection	
<input checked="" type="checkbox"/>	0	0	0	0	0	1	0	r-0,0,0,0,0,0,0	f-0,0,0,0,0,0,0	Reflection	

Figura 3: Característica de las etapas.

Los tipos de aperturas y superficies que pueden ser utilizadas, se muestran en la Figura 4. A las cuales podemos ingresar al seleccionar el botón de *Aperture* o *Surface*, respectivamente.

#### 4.1. Aimpoint

El *aimpoint* es un vector que define SolTraces y nos indica la orientación de una superficie. Considerese la Figura 5 en donde se muestra una superficie plana en donde incide el vetor solar  $\hat{s}$  el cual es reflejado con una dirección  $\hat{r}$  que incide en el objetivo deseado; se definen adicionalmente los vectores  $\vec{R}_F$  y  $\vec{R}_T$  que son los vectores que van desde el origen de las coordenadas

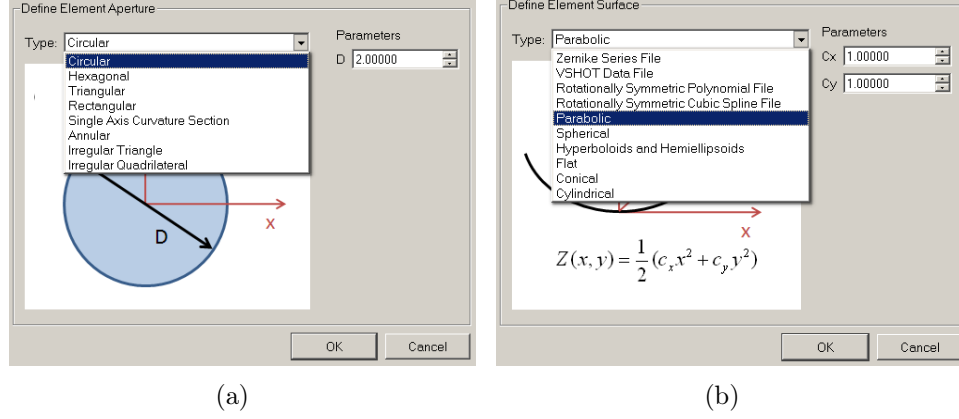


Figura 4: Tipos de apertura, a la izquierda, y de superficie, a la derecha.

globales a la superficies y al objetivo, respectivamente. La orientación de dicha superficie esta definida por su *aimpoint* que es un vector que va desde el origen a cualquier punto del vector normal  $\hat{n}$  de la superficie. El *aimpoint* es la suma de los vectores  $\vec{R}_F$  y  $\vec{n}$ . Hay que hacer notar que si nuestra superficie esta en el origen el *aimpoint* coincidirá con la dirección del vector normal, dado que  $\vec{R}_F$  es cero. Pero para las demás superficies que se encuentran fuera del origen el  $\vec{R}_F$  es diferente de cero. Como lo muestra la Figura 5 el *aimpoint* de una superficie fuera del origen esta dado por la Ec. 4

$$\hat{n} = \frac{\hat{s} + \hat{r}}{\|\hat{s} + \hat{r}\|} \quad (3)$$

$$\vec{a} = \vec{R}_F + \hat{n} \quad (4)$$

## 5. Trazado de rayos

Una vez que se han definido las diferentes secciones de los elementos se procede a realizar la simulación de trazado de rayos, para lo cual es necesario indicar el número de rayos para la interacción, hay que recordar que SolTrace utiliza el método de Monte Carlo, y dado que se trata de un método estadístico, entre más rayos se utilicen el resultado obtenido será más preciso. Usualmente se selecciona un número de rayos pequeño, e.g., un millón es un buen número para iniciar, y corregir tus posibles errores, sin embargo

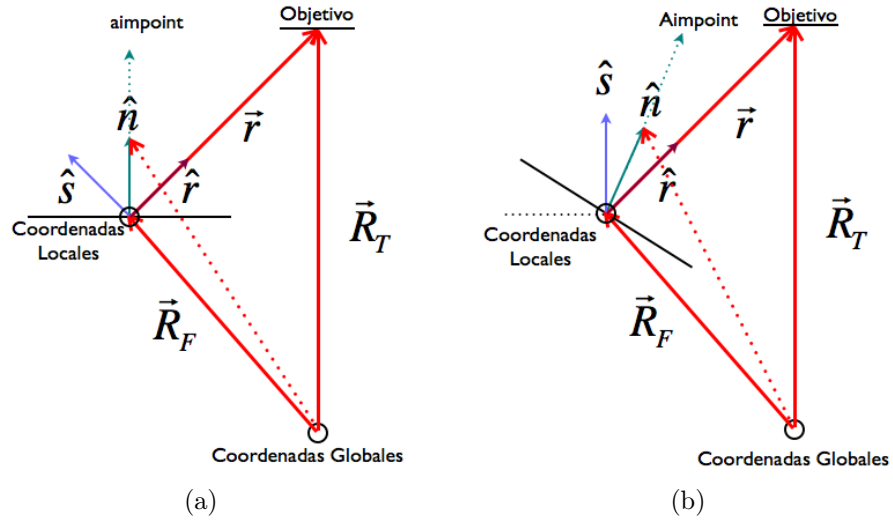


Figura 5: Definición de *aimpoint*.

es necesario realizar un análisis de independencia de malla para conocer el número óptimo de rayos necesarios para nuestra simulación.

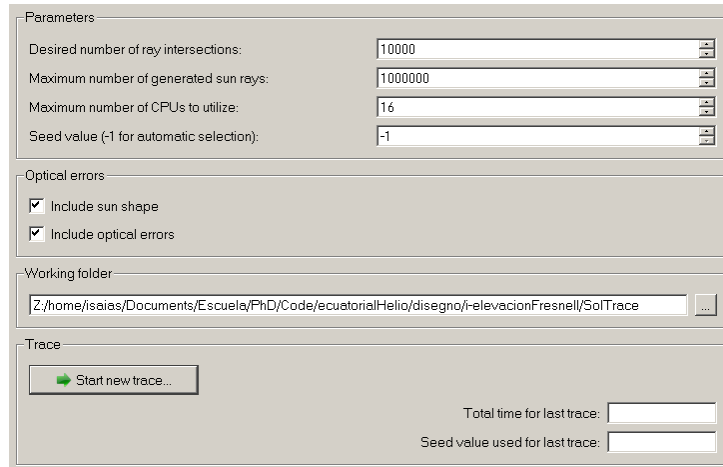


Figura 6: Opciones de trazado de rayos.

## 6. Plato parabólico

**Problema 6.1:** Simule un plato parabólico en SolTrace que tenga un ángulo de borde  $\psi_b = \pi/2$  y una apertura de dos metros.

**Solución:**Primero, considere la parábola de la Figura 7 generada con la con la ecuación (5) definida por el foco  $f$ .

$$y = \frac{x^2}{4f} - f \quad (5)$$

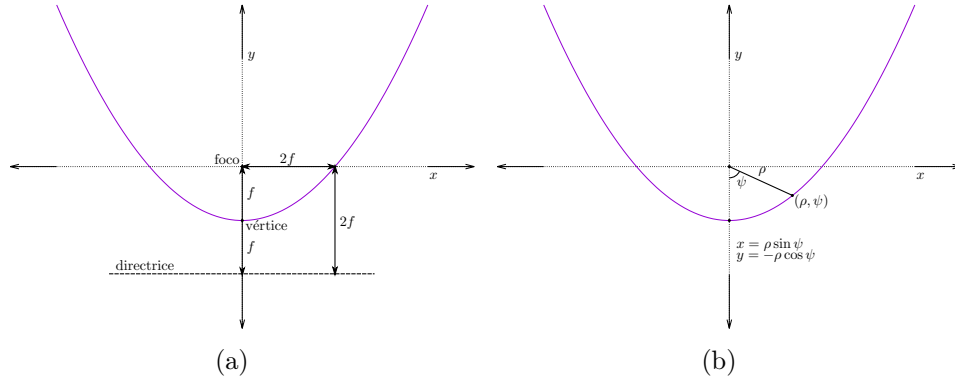


Figura 7: Parábola con el foco en el origen.

Es decir, que si el ángulo de borde es de  $\pi/2$ , implica que  $x_b = 1$  m ya que la apertura es de dos metros y entonces  $f$  es de 0.5 m.

Sabemos[5] que para un receptor cilíndrico el diámetro del receptor está dado por la ecuación (6). Si consideramos que el semi-ángulo de aceptación  $\Delta_r$  es del tamaño del cono solar, entonces  $\Delta = 4.65$  mrad. Luego entonces podemos calcular el diámetro del receptor ideal y es igual 9.3 mm.

$$D = \frac{4\Delta_r f}{1 + \cos \psi_b} \quad (6)$$

**SolTrace:**Para simular este disco parabólico consideraremos un vector solar paralelo al eje  $z$ , es decir  $\hat{s} = (0, 0, 1)$  y una forma Gaussiana. Respecto a los parámetros ópticos, para el concentrador consideraremos una reflectividad casi especular de 0.96, una transmitancia de prácticamente cero, sin errores de pendiente y especularidad. La Tabla 1 muestra los parámetros men-



cionados. Para el receptor generamos otro elemento en donde la reflectancia será nula.

Reflectivity	Transmissivity	Slope error	Specularity error	Error type
0.96	0.0001	0.0001	0.0001	Gaussian

Cuadro 1: Propiedades ópticas

En la sección de etapas vamos a generar nuevamente dos elementos, una para el concentrador y una para el receptor. La Tabla 5 muestra los parámetros del concentrador, para lo cual es necesario primero, mediante el botón *Insert*, insertar un elemento. En el cual ingresaremos la posición del paraboloide, que será el origen, en donde la su normal coincide con el *aimpoint* es decir  $(0, 0, 1)$ . La apertura es circular con un diámetro de 2 m y el tipo de superficie es una parábola con un foco de  $f = 0.5$  m. El tipo de interacción es *Reflection* y en el elemento *Optics* seleccionamos el de *concentrador* que previamente habíamos definido.

X-C	Y-C	Z-C	X-AimP	Y-AimP	Z-AimP	Z-Rot	Aperture	Surface
0	0	0	0	0	1	0	c-2,0,0,0,0,0,0,0	p-1,1,0,0,0,0,0,0

Cuadro 2: Etapa del concentrador.

En la segunda etapa generamos el receptor circular plano posicionado en el foco, es decir, ubicado en  $(0, 0, 0.5)$ , y con un *aimpoint* de  $(0, 0, -1)$  sigue siendo paralelo al eje  $z$  pero en sentido contrario. Seguimos teniendo una apertura circular con el radio calculado de 9.3 mm, pero ahora tenemos una superficie plana. Ahora las propiedades ópticas serán las definidas como *concentrador*.

X-C	Y-C	Z-C	X-AimP	Y-AimP	Z-AimP	Z-Rot	Aperture	Surface
0	0	0.5	0	0	-1	0	c-0.0093,0,0,0,0,0,0,0	f-0,0,0,0,0,0,0,0

Cuadro 3: Etapa del receptor

Ya que tenemos de finido el sistema podemos proceder a realizar la simulación, para lo cual seleccionaremos 1,000,000 de rayos. Una vez indicado el número de rayos procedemos a realizar la simulación mediante el botón *Start new trace*.

La sección de **Results** tiene tres secciones: *Intersections*, *Flux Maps* y *Ray Data*. En la sección de *Intersections* podemos visualizar los distintos elementos simulados, en donde solo observaremos la interacción de los rayos con las superficies si existen, también tenemos la opción de imprimir la trayectoria de los rayos deseados.

La Figura 8 muestra la sección de *Intersections* en ella podemos seleccionar la etapa de trazados de rayos y los elementos que contienen para visualizarlos. Una vez seleccionados se visualizan en la interacción de los rayos con las superficies que interactúan, además si seleccionamos *Plot path of ray #s* podemos indicar el intervalo de rayos que deseamos visualizar. Los rayos que se muestran en color rojos son aquellos que no inciden en el receptor. En esta sección se indica cuál es la irradiancia considerada para la simulación, en este caso es de  $1,000 \text{ W/m}^2$ . Por último se muestra una sección con un resumen de la información del trazado de rayos.

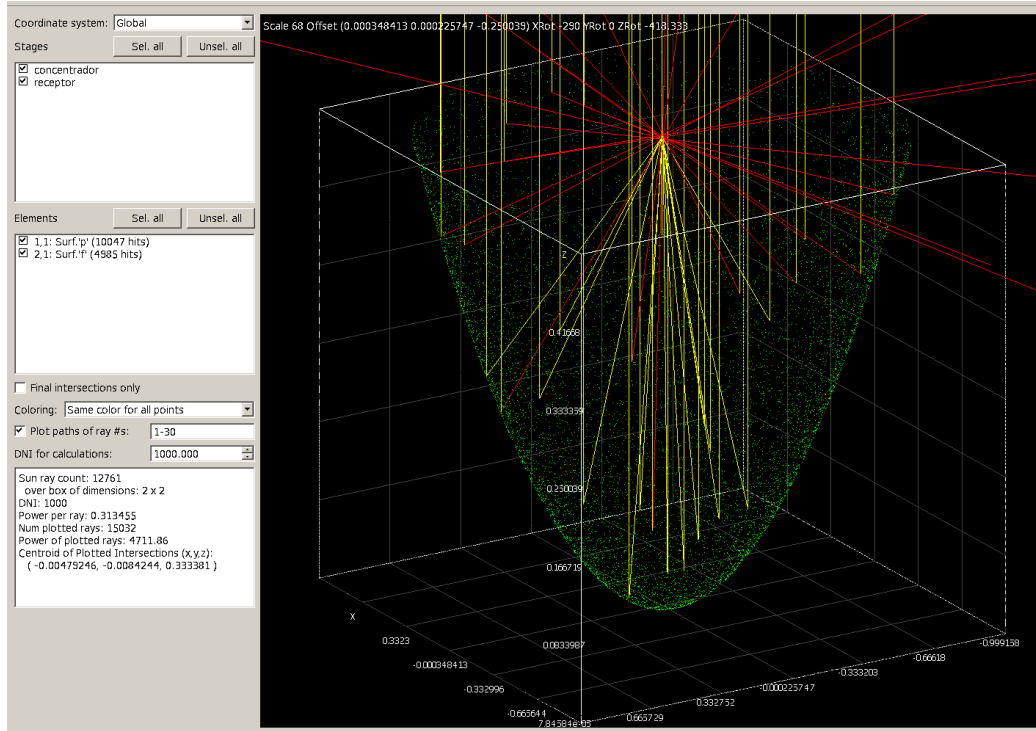


Figura 8: Sección de *Intersections* de resultados en SolTrace.

En la sección de *Flux Maps* se puede seleccionar la sección de interés a

visualizar. En este caso únicamente tenemos un elemento. En esta sección podemos indicar el tamaño de la malla, que por default es de  $20 \times 20$ . Existe también una sección para exportar los datos que nos genera dos archivos uno de con la extensión de “.tec” y “.flx”, el primero es un archivo de datos que contiene la coordenadas y el valor de intensidad que le corresponde. El segundo archivo es la misma información pero en un arreglo matricial. Y por último, existe una sección en donde se muestran un resumen de la simulación. La Figura 9 muestra la pestaña de *Contour Plot* en donde se observa la distribución de flujo, la pestaña de *Surface Plot* nos muestra la misma información en 3D.

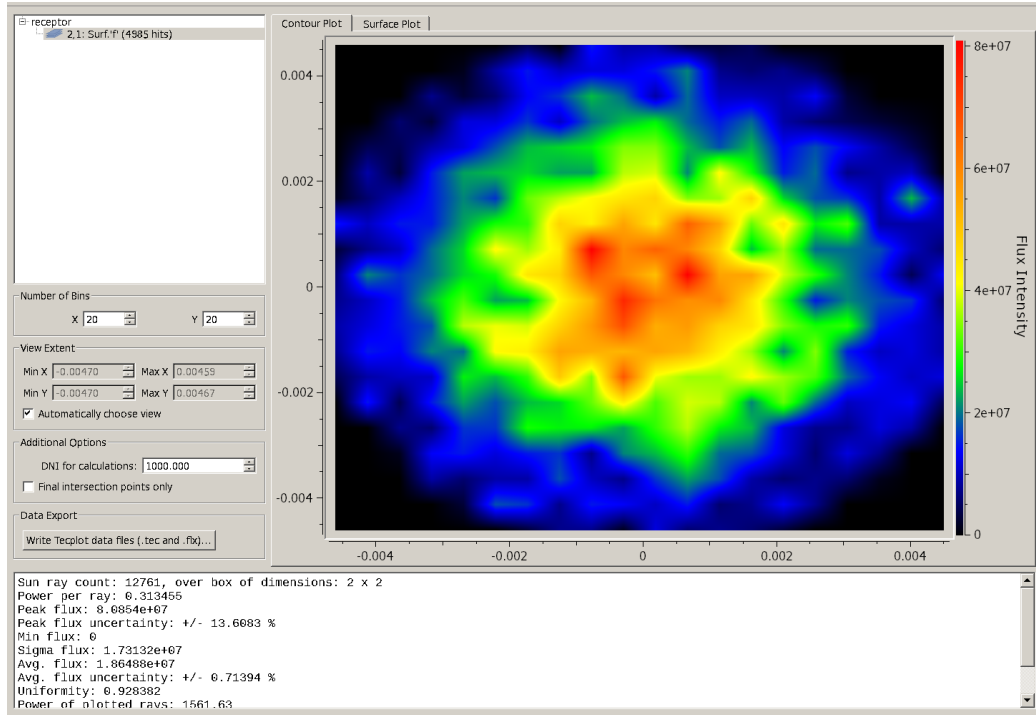


Figura 9: Distribución de flujo en el receptor.

La sección de *Ray Data* nos permite explotar la información del trazado de rayos en un archivo “.csv”.

Por último realizaremos un trazado de rayos con 7 millones de rayos, mismo que exportaremos con una malla de  $35 \times 35$ .

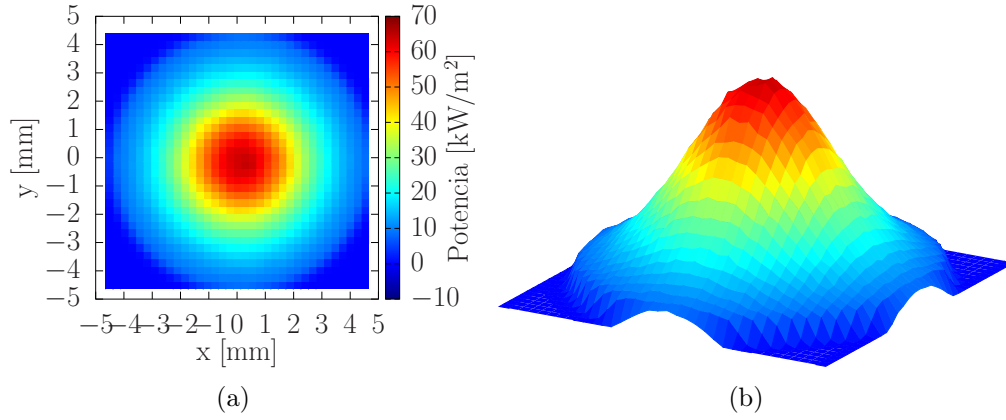


Figura 10: Flujo en el receptor.

## 7. Canal Parabólico

**Problema 7.1:** Realice una el trazados de rayos de un canal parabólico que tiene un ángulo de borde  $\psi = 80^\circ$ , y un ancho de apertura de 1.5 m.

**Solución:** Sabemos que de la ecuación 7 podemos calcular el foco  $f$  de la parábola, dado que conocemos el ángulo de borde  $\psi = 80^\circ$  y sabemos que  $x_b = \rho \sin \psi_b$  en donde  $x_b = 0.75$ , debido a que la apertura es de 1.5 m. Calculamos  $\rho$  que es igual a 0.76 m. Considerese la Figura 12.

$$\rho = \frac{2f}{1 + \cos \psi_b} \quad (7)$$

despejando  $f$  y sustituyendo los valores correspondientes  $f = 0.4469$  m.

Por otro lado, sabemos que el área de apertura es  $A_p = (2x_b)L$ , por lo que podemos considerar una  $L = 3$  m. Para el calculo del receptor utilizamos la ecuación 8, nuevamente, en donde consideraremos el semi-ángulo de aceptación  $\Delta = 4.65 \text{ mrad}$  igual al tamaño del cono solar.

$$D = \frac{4\Delta_r f}{1 + \cos \psi_b} \quad (8)$$

sustituyendo los valores podemos calcular que el diámetro ideal del receptor que es igual a 7.08 mm.

**SolTrace:** Para la simulación de trazado de rayos utilizamos utilizamos un vector solar  $\hat{s} = (0, 0, 1)$ , una función *Pillbox* forma solar. Las propiedades ópticas del concentrador se muestran en la Tabla 4

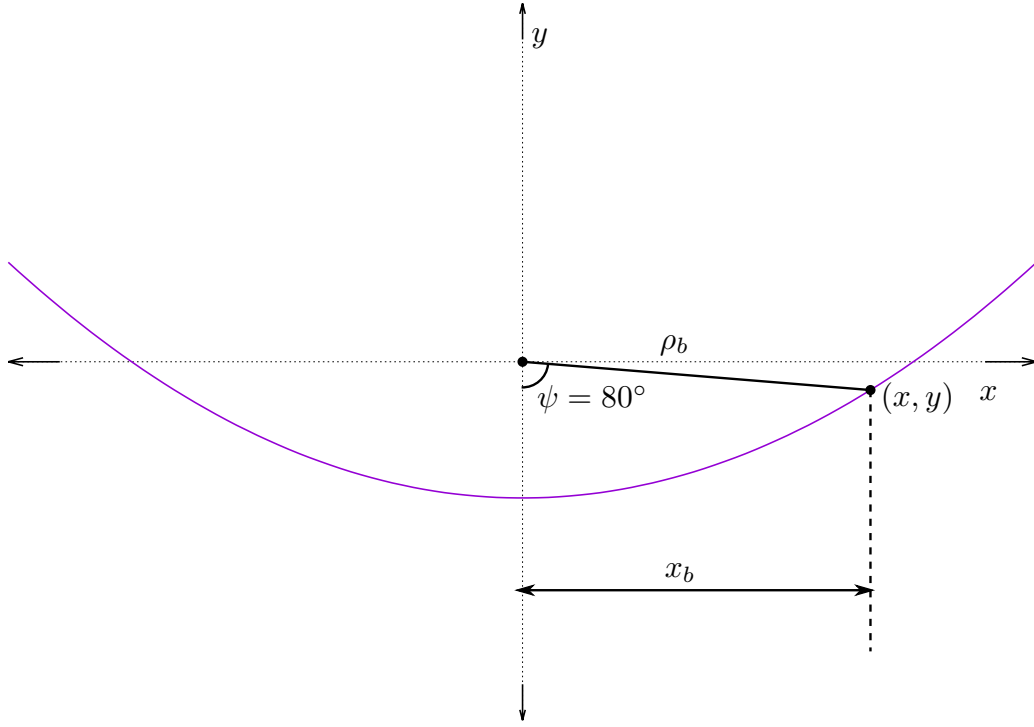


Figura 11: Parabola del concentrador de canal parabólico.

Reflectivity	Transmissivity	Slope error	Specularity error	Error type
0.96	0.0001	0.0001	0.0001	Gaussian

Cuadro 4: Propiedades ópticas

Se generan dos etapas una para el concentrador y otra para el receptor. Para simular un canal parabólico se selecciona la apertura *Rectangular* y se considera una longitud de 3 m. Se utiliza una superficie parabólica con el foco calculado. La interacción es de reflexión y los parámetros ópticos los definidos con anterioridad.

Las propiedades del receptor se muestran en la Tabla 6, donde el receptor lo colocamos en el foco. La apertura también es del tipo *Single Axis Curvature Section* con una longitud de 3 m. La superficie es de tipo cilíndrico con el diámetro calculado.

Realizando una simulación para con un millón de rayos se obtienen los

X-C	Y-C	Z-C	X-AimP	Y-AimP	Z-AimP	Z-Rot	Aperture	Surface
0	0	0	0	0	1	0	r-0.75,3,0,0,0,0,0	p-1.1213,0,0,0,0,0,0

Cuadro 5: Etapa del concentrador.

X-C	Y-C	Z-C	X-AimP	Y-AimP	Z-AimP	Z-Rot	Aperture	Surface
0	0	0.4469	0	0	-1	0	l-0,0,3,0,0,0,0	t-282.486,0,0,0,0,0,0

Cuadro 6: Etapa del receptor

resultados mostrados en la Figura 12.

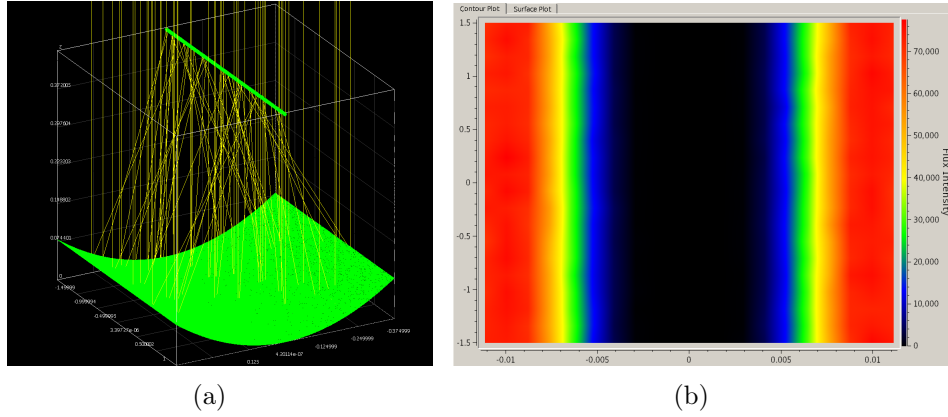


Figura 12: Trazado de rayos del un canal parabólico.

## 8. Helióstatos

**Problema 8.1:** Realizar el trazado de rayos de un helióstato y un receptor colocado en lo alto de una torre, ver Figura 13. Consideré que el helióstato tiene unas dimensiones de  $1 \times 1$  m, y se encuentra a 50 m de la torre, hacia el norte. El receptor esta a una altura de 32 m. Considere que el vector solar es igual a  $\hat{s} = (0, 0, 1)$ .

**Solución:** Sabemos que por la ley de reflexión podemos calcular la dirección de la normal  $\hat{n}$  del helióstato con la ecuación 9, de donde conocemos  $\hat{s}$  y el vector de reflexión es  $\vec{R} = T - H$  y su valor unitario es de  $\hat{r} = (0.00, -0.8422, 0.5390)$ . Por lo que el valor de  $\hat{n} = (0.00, -0.480, 0.877)$ .

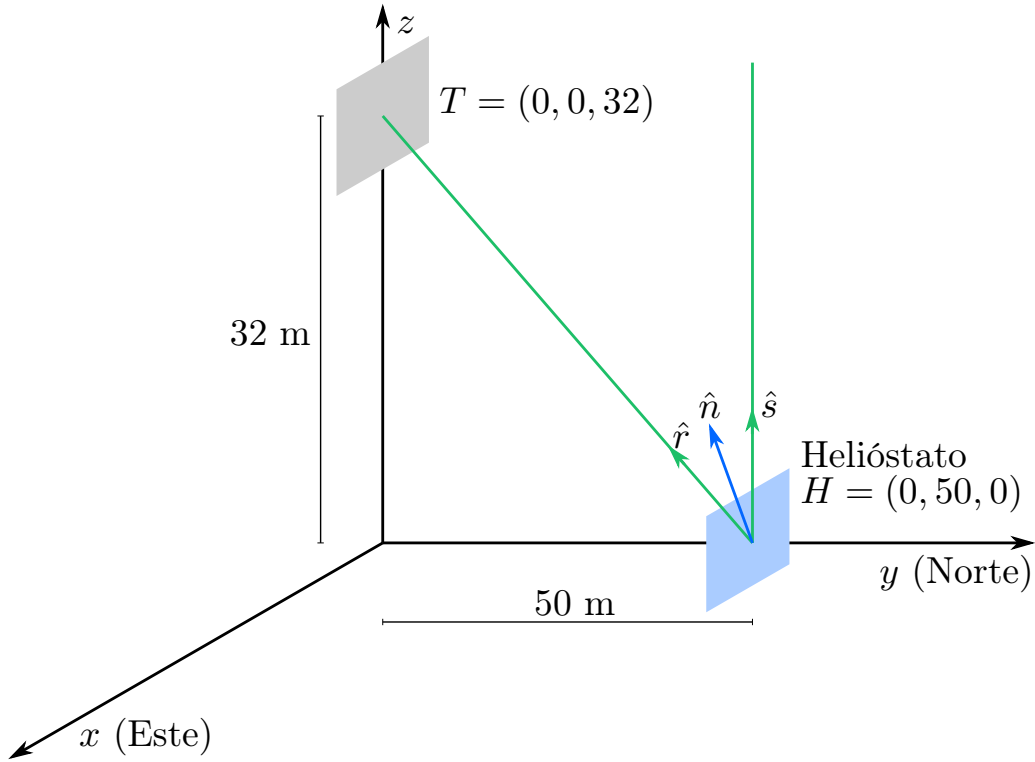


Figura 13: Heliostato

$$\hat{n} = \frac{\hat{s} + \hat{r}}{\|\hat{s} + \hat{r}\|} \quad (9)$$

El valor del *aimpoint*  $\vec{A} = \vec{R}_F + \hat{n}$ , donde  $vecR_F$  es un vector que va del origen a la posición del heliostato, en este caso  $\vec{R}_F = (0, 50, 0)$ , y  $\vec{A} = (0, 49.52, 0.877)$ .

**SolTrace::**

Considerese las propiedades ópticas de la Tabla 7 para el heliostato.

Reflectivity	Transmissivity	Slope error	Specularity error	Error type
0.95	0.0001	2.500	0.2000	Gaussian

Cuadro 7: Propiedades ópticas

Nuevamente se requieren dos etapas, una para el helióstato y otra para el receptor. La posición del helióstato es  $H = (0, 50, 0)$ , y su *aimpoint* es el calculado. El Helióstatto tiene una apertura de  $1 \times 1$  m y en este caso se trata de una superficie plana. La Tabla 8 indica los parámetros indicados.

X-C	Y-C	Z-C	X-AimP	Y-AimP	Z-AimP	Z-Rot	Aperture	Surface
0	50	0	49.52	0.877	0	0	r-1,1,0,0,0,0,0,0	p-0,0,0,0,0,0,0,0

Cuadro 8: Etapa del concentrador.

El receptor, en este caso, se trata de una superficie plana orientada hacia el norte. Dado que no indican el tamaño utilizaremos un receptor de  $3 \times 3$  m, como lo indica la Tabla 9.

X-C	Y-C	Z-C	X-AimP	Y-AimP	Z-AimP	Z-Rot	Aperture	Surface
0	0	32	0	1	32	0	r-3,3,0,0,0,0,0,0	f-0,0,0,0,0,0,0,0

Cuadro 9: Etapa del receptor

Posteriormente se realiza una simulación considerando a millón de rayos. La Figura 14 muestra la simulación.

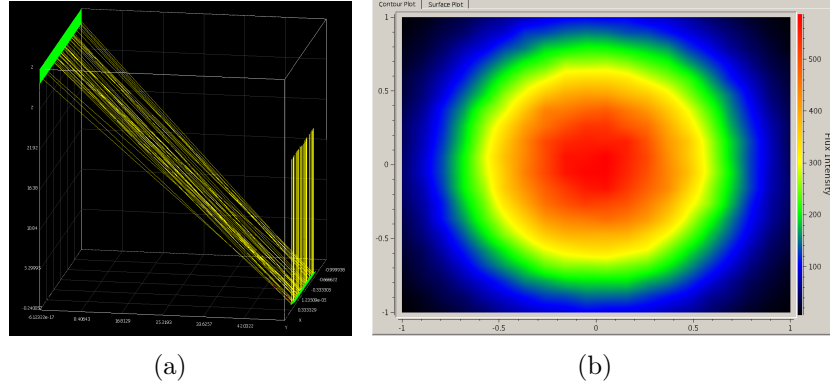
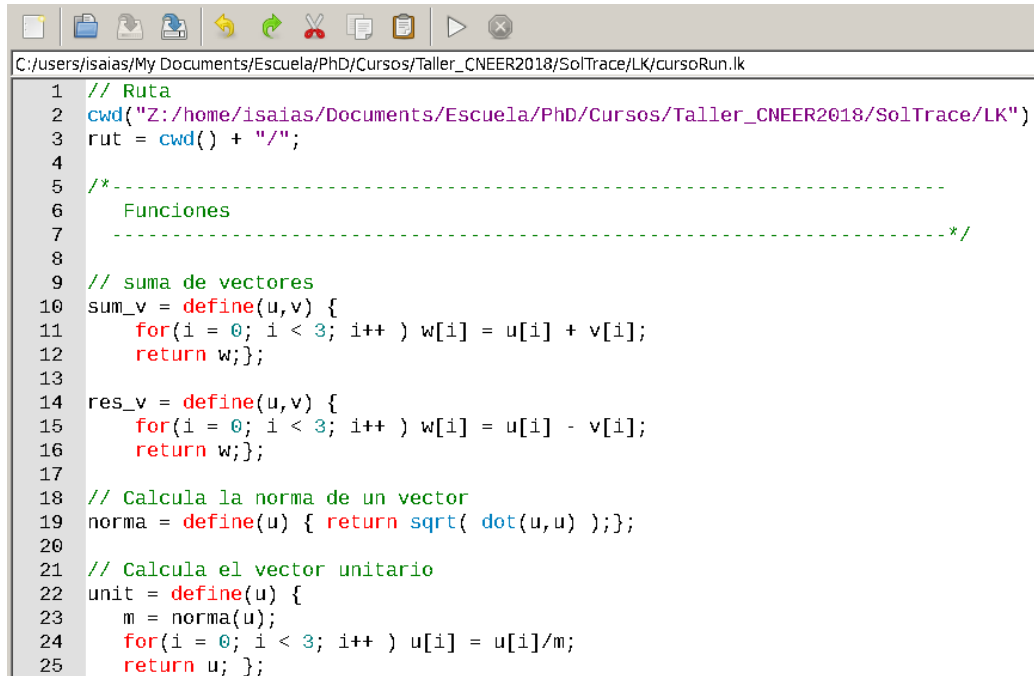


Figura 14: Simulación de trazado de rayos de un helióstato.



## 9. Scripting

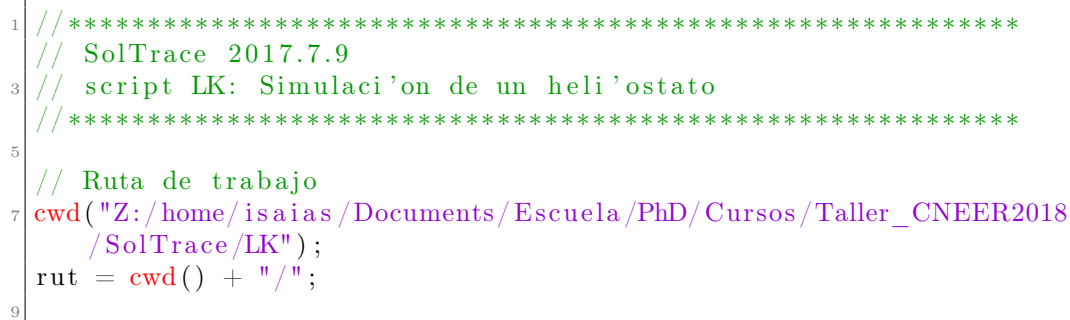
En SolTrace se puede utilizar scripts, ver Figura 15, para programar el trazado de rayos mediante LK. De esta manera es posible hacer simulaciones en lo largo del tiempo.



```
C:/users/isaia/My Documents/Escuela/PhD/Cursos/Taller_CNEER2018/SolTrace/LK/cursorun.lk
1 // Ruta
2 cwd("Z:/home/isaia/Documents/Escuela/PhD/Cursos/Taller_CNEER2018/SolTrace/LK");
3 rut = cwd() + "/";
4
5 /*-----
6     Funciones
7     -----*/
8
9 // suma de vectores
10 sum_v = define(u,v) {
11     for(i = 0; i < 3; i++ ) w[i] = u[i] + v[i];
12     return w;};
13
14 res_v = define(u,v) {
15     for(i = 0; i < 3; i++ ) w[i] = u[i] - v[i];
16     return w;};
17
18 // Calcula la norma de un vector
19 norma = define(u) { return sqrt( dot(u,u) );};
20
21 // Calcula el vector unitario
22 unit = define(u) {
23     m = norma(u);
24     for(i = 0; i < 3; i++ ) u[i] = u[i]/m;
25     return u; };
```

Figura 15: Script LK para la SolTrace.

El script LK es una modificación del lenguaje C, puedes para conocer las características básicas revisar el documento Guide LK. A continuación se muestra un script para simular el helióstato del problema de la Sección 8.



```
1 // *****
2 // SolTrace 2017.7.9
3 // script LK: Simulaci'on de un heli'ostato
4 // *****
5
6 // Ruta de trabajo
7 cwd("Z:/home/isaia/Documents/Escuela/PhD/Cursos/Taller_CNEER2018
8 /SolTrace/LK");
9 rut = cwd() + "/";
```

```

11 // *****
12 // FUNCIONES DE INTER'ES B'ASICAS
13 // *****
14
15 // suma de vectores
16 sum_v = define(u,v) {
17     for(i = 0; i < 3; i++ ) w[i] = u[i] + v[i];
18     return w;};
19
20 res_v = define(u,v) {
21     for(i = 0; i < 3; i++ ) w[i] = u[i] - v[i];
22     return w;};
23
24 // Calcula la norma de un vector
25 norma = define(u) { return sqrt( dot(u,u) );};
26
27 // Calcula el vector unitario
28 unit = define(u) {
29     m = norma(u);
30     for(i = 0; i < 3; i++ ) u[i] = u[i]/m;
31     return u; };
32
33 /* Calcula la normal del heliostato
34    s - vector solar
35    T - posico'on de la torre
36    H - posici'on del heli'ostato
37 */
38 normal = define(s, T, H){
39     s = unit(s);
40     t = res_v(T, H);
41     t = unit(t);
42     n = sum_v(s,t);
43     return unit(n);
44 };
45
46 // Calcula el aimpoint del heli'ostato
47 aimpoint = define(s, T, H){
48     n = normal(s,T,H);
49     ap = sum_v(H,n);
50     return ap;
51 };
52
53 // Irradiacia
54 AM = define(ha, theta){
55     d = cos(theta) + 0.5057*(96.080 - theta)^(-1.634);

```

```

55     return exp(-0.0001184*ha)/d;
56 };
57
58 G = define(G0, tau, am){
59     x = am^0.678;
60     Gh = G0*tau^x;
61     return Gh;
62 };
63
64 // *****
65 // DATOS B'ASICOS DE ENTRADA
66 // *****
67
68 // Vector solar unitario
69 s[0] = 0.428720;
70 s[1] = -0.897006;
71 s[2] = 0.107607;
72
73 // Posici'on del receptor
74 T[0] = 0.0;
75 T[1] = 0.0;
76 T[2] = 32.0;
77
78 // Dimensiones del receptor
79 wr = 3.0;
80 hr = 3.0;
81
82 // Aimpoint del receptor
83 apT[0] = 0.0;
84 apT[1] = 1.0;
85 apT[2] = T[2];
86
87 //Heli'ostato
88 H[0] = 0.0;
89 H[1] = 50.0;
90 H[2] = 0.0;
91
92 //Dimensiones del heli'ostato
93 w = 1.0;
94 h = 1.0;
95
96 // Calculo del aimpoint
97 ap = aimpoint(s, T, H);
98
99 // Calculo de irradiancia

```

```

101 G0 = 1367; // Constante solar [G0] = W/m2
tau = 0.7; // Transmitancia de cielo claro
ha = 1313; // Altitud del lugar [ha] = m

103 theta = acos(s[2]);
105 I = G(G0, tau, AM(ha, theta));
outln("Irradiancia = " + I);
107

109 // *****
// DATOS DE SOLTRACE
111 // *****

113 // Forma solar
Sun.shape = 'g';
115 Sun.sigma = 2.73;
Sun.x = s[0];
117 Sun.y = s[1];
Sun.z = s[2];
119 sunopt(Sun);

121 // Propiedades 'opticas
clearoptics();
123

addoptic('espejo');
125 opticopt('espejo', 1, {'refl' = 1, 'trans' = 0.0001, 'errslope'
= 2.64, 'errspec' = 0.0001});

127 // Etapas
clearstages();
129

// Heliostato
131 addstage('heliostato');
activestage('heliostato');
133

addelement(); // adds element to current stage
135 elementopt( 0, {'en'=true, 'x' = H[0], 'y' = H[1], 'z' = H[2],
'ax' = ap[0], 'ay' = ap[1], 'az' = ap[2],
137 'zrot'=0, 'aper'=['r',w,h,0,0,0,0,0,0],
'surf'=['f', 0,0,0,0,0,0,0,0],
139 'optic'='espejo' });

141 // Receptor
addstage('receptor');
143 activestage('receptor');

```

```

145 addelement(); // adds element to current stage
146 elementopt( 0, { 'en' = true, 'x' = T[0], 'y' = T[1], 'z' = T[2] ,
147   'ax' = apT[0], 'ay' = apT[1], 'az' = apT[2],
148   'zrot' = 0, 'aper' = [ 'r', wr, hr, 0, 0, 0, 0, 0, 0 ],
149   'surf' = [ 'f', 0, 0, 0, 0, 0, 0, 0, 0 ],
150   'optic' = 'espejo' } );
151
152 // Simulaci'on
153 // *****
154 n = 2000000; // Rayos de simulaci'on
155 traceopt({ "rays" = n } );
156 trace();
157
158 // Elemento a exportar
159 stage = 1;
160 element = 0;
161 nBinX = 40;
162 nBinY = 40;
163 DNI = I;
164
165 // Exportar matriz de flujo
166 fs = rut + "matriz.dat";
167 file_out = open(fs, "w");
168
169 // Estad'istica
170 stat = elementstats( stage, element, nBinX, nBinY, DNI, false );
171 da = stat.bin_size_x * stat.bin_size_y;
172
173 write_line(file_out, "## Muestra el numero de rayos que golpean
174   cada bin");
175 flux = "";
176 for (r = 0 ; r < #stat.flux ; r++){
177   for (c = 0 ; c < #stat.flux[r] ; c++){
178     aux = stat.flux[c][r] * stat.power_per_ray/da;
179     flux = flux + aux + "\t";
180   }
181   write_line(file_out, flux);
182   flux = "";
183 }
184 close(file_out);
185
186 // Exportar archivo de estad'isticas
187 // *****
188 fs = rut + "stats.dat";

```

```

189 file_out = open(fs , "w");
191 stat.flux = null;
191 stat.xvalues = null;
191 stat.yvalues = null;
193 keys = @stat;
195 write_line(file_out , "## Todas las estadisticas:");
195 for (i = 0; i < #keys; i++)
197     write_line( file_out , "# " + keys[i] + " = " + stat{keys[i]} );
197 write_line(file_out , "# ray hits on target: " + rayhits('1', '0')
197     );
199 close(file_out);
199 //Guarda el archivo de trabajo
201 //*****
203 name = "simulacion";
203 save_project(rut + name + ".stinput");

```

./../LK/cursorun.lk

## 10. Cálculo de la radiación solar

Dado que el valor de la irradiancia varia con el tiempo, es necesario calcularla en el tiempo.

La radiación directa puede ser calculada, para diferentes horas y días, con un modelo de cielo claro [2], utilizando:

$$G_b = G_0 \tau^{AM^{0.678}} \quad (10)$$

donde  $G_0$  es la constante solar de  $1,367 \text{ W/m}^2$ , y  $\tau$  es la transmitancia atmosférica, que para condiciones de cielo claro tiene un valor de 0.7, y la masa de aire es obtenida como función de la altitud del lugar.

$$AM = \frac{\exp(-0.0001184 h_{\text{altitud}})}{\cos \theta_z + 0.5057(96.080 - \theta_z)^{-1.634}} \quad (11)$$

$\theta_z$  es el ángulo cenital del vector solar, y  $h_{\text{altitud}}$  la altura sobre el nivel del mar.

## Referencias

- [1] D Buie, AG Monger, and CJ Dey. Sunshape distributions for terrestrial solar simulations. *Solar Energy*, 74(2):113–122, 2003.
- [2] MT Dunham, R Kasetty, A Mathur, and W Lipiński. Optical analysis of a heliostat array with linked tracking. *Journal of Solar Energy Engineering*, 135(3):034501, 2013.
- [3] Pierre Garcia, Alain Ferriere, and Jean-Jacques Bezian. Codes for solar flux calculation dedicated to central receiver system applications: A comparative review. *Solar Energy*, 82(3):189–197, March 2008.
- [4] Richard B Pettit. Characterization of the reflected beam profile of solar mirror materials. *Solar Energy*, 19(6):733–741, 1977.
- [5] Ari Rabl. *Active solar collectors and their applications*. Oxford University Press on Demand, 1985.
- [6] Tim Wendelin, Aron Dobos, and Allan Lewandowski. Soltrace: A ray-tracing code for complex solar optical systems. *Contract*, 303:275–3000, 2013.