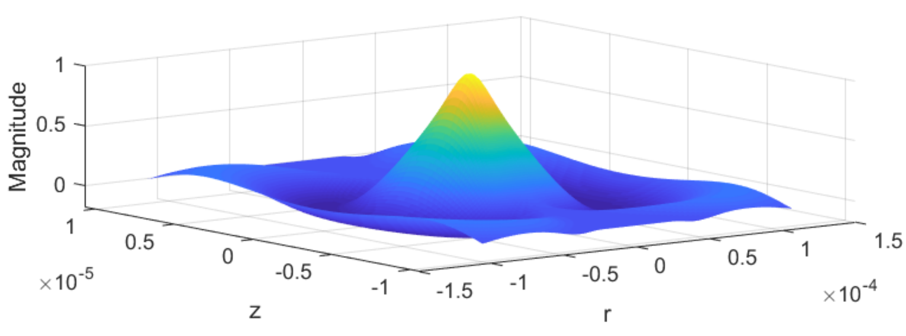
GaussCAD - Gaussian beam field simulator.

Schultze, Alexander

v1.0 / January 2021

GaussCAD is a matlab based software that combines two separate tools for evaluating simple paraxial optics: a ray transfer optics tracer and Gaussian beams fields evaluator.
A ray transfer analysis (for ABCD matrix) tracer functionality allows allows to propagate simple beams, as well as gaussian beams, through optical elements by simulating a optical bench.
The Gaussian beam field bench allows visual representation of intensity/phase fields as described by Gaussian optics. It allows intensity and phase propagation as well as generating interference intensities and interferogram when combining two beams on a screen. It allows phase evaluation based on numerical evaluation of the interferograms.

The Repository is available at: github.com/alexschultze/GaussCAD.



1 Gaussian Optics Beam (Class @gaussian.gauss-beam)

Gaussian Beam The Gaussian beam is described as a transversal electrical mode (TEM) is description of monochromatic electromagnetic radiation. The fundamental (TEM_{00}) can be described according to the following definition[1]:

$$\mathbf{E}(r,z) = E_0\,\hat{\mathbf{x}}\,\frac{w_0}{w(z)}\exp\left(\frac{-r^2}{w(z)^2}\right)\exp\left(-i\left(kz + k\frac{r^2}{2R(z)} - \psi(z)\right)\right) \tag{1}$$

r .. is the radial distance from the center axis of the beam,
z .. is the axial distance from the beam's focus (or waist),
i .. is the imaginary unit,
$k = 2\pi n/\lambda$ .. is the wave number (in radians per meter) for a free-space wavelength λ, and n is the index of refraction of the medium in which the beam propagates,
E_0 = E(0, 0) .. the electric field amplitude (and phase) at the origin at time 0,
$w(z)$ .. is the radius at which the field amplitudes fall to 1/e of their axial values (i.e., where the intensity values fall to 1/e2 of their axial values), at the plane z along the beam,
$w0 = w(0)$ .. is the waist radius,
$R(z)$ .. is the radius of curvature of the beam's wavefronts at z, and
$\psi(z)$ .. is the Gouy phase at z, an extra phase term beyond that attributable to the phase velocity of light.

GaussCAD The software can be used to describe and evaluate a beam in the beam reference system (r, z). The beam can be transformed in cartesian coordinates (x, y, z). In cartesian definition the vector $\overrightarrow{p}$ describes the location of the beam waist and the vector $\overrightarrow{n}$ describes the axial direction vector along along which the axial distance z propagates.

2 Gaussian Screen (Class @gaussian.field-beam)

Field screen represents a two dimensional pixel sensor array. It accepts two input beams and evaluates their field at each pixel. The screen is described by its parameters of position r, normal vector n, its physical dimension (2D) and pixel count (2D).

[1] From https://en.wikipedia.org/wiki/Gaussian_beam , retrieved 09.12.2020

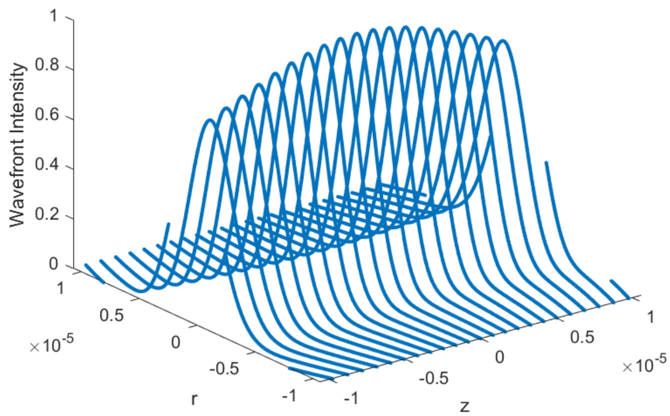


Figure 1: Wavefront progression with intensity for a Gaussian beam in beam coordinates r and z.

When rendered, the current field value for each beam is calculated and saved for each pixel.
The class provides additional functions for masking and segmentation to represent the shape of actual quadrant photodiode with gap.
The pixels irradiance (intensity) is equal to the magnitude of sum of the two phasors squared. The AC, DC is determined by the maximum and minimum values of the phasors magnitude and their difference. The phase information (interferogram) for each location pixel is the difference between the two beam phasor angles.

$$AC(x,y,z) = min(mag(E1(x,y,z)), mag(E2(x,y,z))) \tag{2}$$

$$DC(x,y,z) = max(mag(E1(x,y,z)), mag(E2(x,y,z))) - AC; \tag{3}$$

$$Phase(x,y,z)\phi = angle(E1(x,y,z)) - angle(E2(x,y,z)) \tag{4}$$

$$Int(x,y,z) = \sum_{n=1}^{2} E_n(x,y,z) \tag{5}$$

The interferogram may contain phase jumps which is in the next setup removed by 2D unwrapping based on total-variation (TV) denoising. [1]
Because each pixel has a different impact on the overall phase information recovered, the phase is weighed by the AC amplitude.

$$\phi_w(x,y,z) = AC(x,y,z) \cdot e \exp -i \cdot \phi \tag{6}$$

The actual phase when combining several pixel arrays (full screen, masked screen, quadrants) is calculated as the resulting phase of the sum of all pixel elements.

$$\Phi = angle(\sum \phi_w(x, y, z)) \tag{7}$$

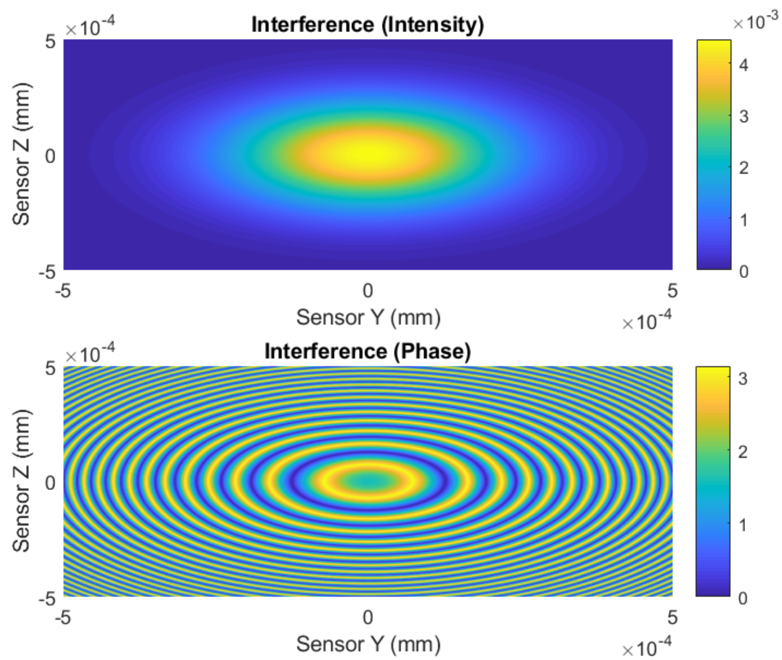


Figure 2: Exemple interferogram of two beams on a Screen class object.

Photodiode Masks / Quadrant Photodiodes The field screen class supports masking to represent round photodetector surfaces. A gap between the segments can be added.

Differential Wave-front Sensing DWS Differential Wave-front sensing is a technique used for evaluating tilts by evaluating phase differences between

Figure 3: Two gaussian beam interference showing interfering (AC) and non-interfering (DC) components on a sensor.

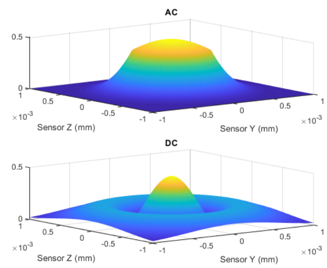
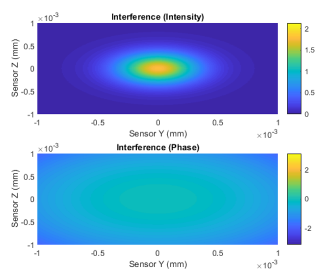


Figure 4: Showing the (phase dependent) current irradiance for a given phase and the resulting interferogram.



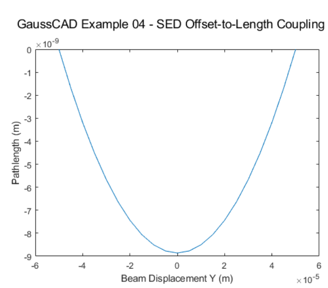


Figure 5: Study of impact of beam displacement into path length with one beam being offset on the sensor.

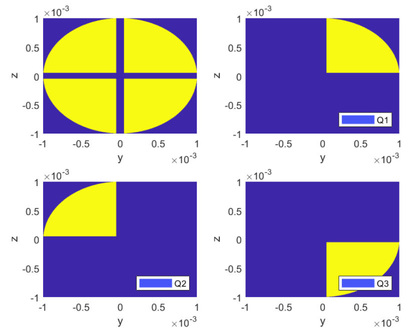


Figure 6: Definition of Quadrant Segments

different quadrants of a photodiode [2] [3].

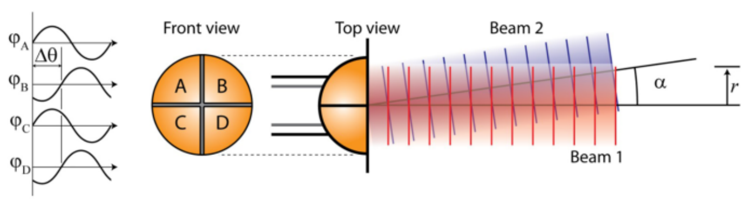


Figure 7: Differential wavefront sensing technique [4].

The class implements Differential Wave-front Sensing evaluation by masking quadrants either orthogonal or diagonally. The definition for calculating the phase difference between quadrants is is given in the following. Q1..Q4 denote the 4 quadrants.

$$DWS_y = \frac{1}{2} \cdot \frac{\Phi_{Q2} + \Phi_{Q3}}{\Phi_{Q1} + \Phi_{Q4}}; DWS_z = \frac{1}{2} \cdot \frac{\Phi_{Q3} + \Phi_{Q4}}{\Phi_{Q1} + \Phi_{Q2}} \tag{8}$$

$$DWS_{diag1} = \frac{\Phi_{Q1}}{\Phi_{Q3}}; DWS_{diag2} = \frac{\Phi_{Q2}}{\Phi_{Q4}} \tag{9}$$

3 ABCD matrix analysis

Ray transfer analysis, or ABCD matrix analysis, describes the propagation of a beam along an central optical axis.

The paraxial bench class (@paraxial.bench) can be used to perform operations of ray transfer matrix analysis. It can be used to visualise and calculate the interaction of optical elements for geometrical rays, as well as propagation of Gaussian beam parameters through optical elements. Optical elements are defined by the class (@paraxial.element). The constructor supports various default types (translation, lens,lensthick, refraction, mirror, mirror.curved, sphericalrefraction) or direct input of an ABCD matrix. Each element is added to a bench at a given location along the optical axis. The translation between optical elements is automatically considered.

3.1 Ray transfer with ABCD matrix

The ray is described by $\vec{r} = \begin{pmatrix} r \\ \alpha \end{pmatrix}$, where r is the axial distance and α the angle between ray and optical axis. An optical element can be described by an ABCD matrix. The relation between input and output ray is given as:

$$\begin{bmatrix} A & B \\ C & D \end{bmatrix} \begin{pmatrix} r_1 \\ \alpha_1 \end{pmatrix} = \begin{pmatrix} r_2 \\ \alpha_2 \end{pmatrix} \tag{10}$$

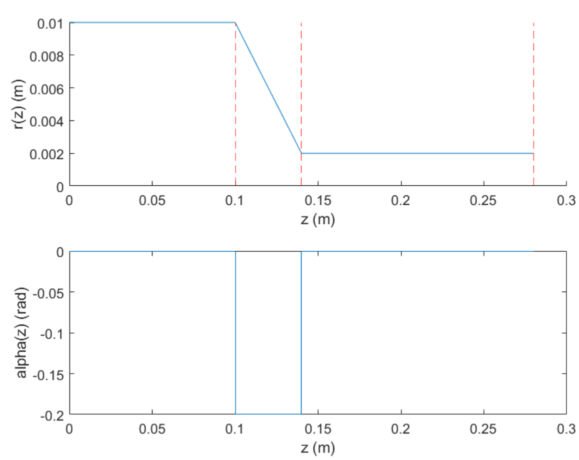


Figure 8: GaussCAD Example of paraxial matrix optics bench beam propagation for a simple example of a galileo telescope with two lenses and magnification of 5:1.

Listing 1: example 01:matrixoptics.m

```
%% Matrix Optics (ABCD) example 01
% to calculate simple ray transfer optics example.
% A. Schultze 01/10/2020 (GaussCAD toolbox)

%Define a bench .
this_bench= paraxial.bench_abcd();
%Add (1) lens with f=0.05 @ 0.1m
this_bench.add(0.1, paraxial.element('lens',0.05));
%Add (2) curved mirror with r=0.02 @ 0.14m
this_bench.add(0.14, paraxial.element('mirror_curved',-0.02));
%Add (3) screen for terminating the bench
this_bench.add(0.28, paraxial.element('screen',0.02));
% Trace a  initial ray of 0.01m and alpha=0 rad
this_bench.plot([0.01; 0]);
```

3.2 Gaussian beam propagation with ABCD matrix

Similar to paraxial optics a complex Gaussian beam parameter can be propagated through the ray transfer optical bench. The change of Gaussian beam parameters $q(z) = z + iz_R$, where z is the distance from the beam waist and $z_r = z_0$ the Rayleigh length, is calculated according to the following formula. The multiplication is performed on each optical element individually, for Gaussian propagation the matrices of optical elements and can not be combined by multiplication (as possible for paraxial optics).

Listing 2: example 02:matrixoptics gauss.m

```
%% Matrix Optics (ABCD) example 02
% to calculate new Gaussian complex beam parameter (q) with ray optics
% A. Schultze 01/10/2020 (GaussCAD toolbox)
this_bench= paraxial.bench_abcd();
this_bench.add(0.1, paraxial.element('lens',0.05));
this_bench.add(0.14, paraxial.element('lens',-0.05));
this_bench.add(0.28, paraxial.element('screen',0.02));
q=1+0.4i; % initial beam with complex beam parameter z=1 and z_r=0.4
this_bench.plot_gauss(q); % draw it
```

$$q_1(z) = \frac{Aq_0 + B}{Cq_0 + D} \tag{11}$$

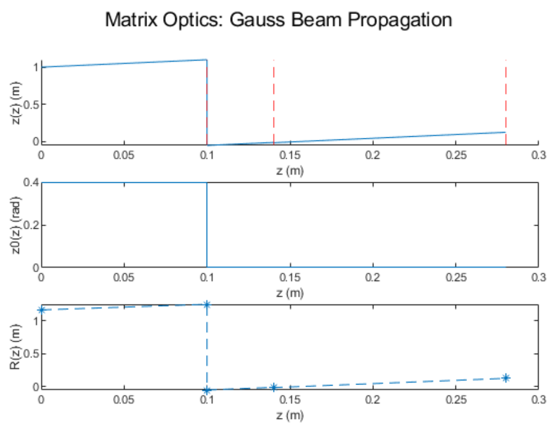


Figure 9: Example of ray transfer matrix optics used for Gaussian beam propagation.

References

[1] Howard Y. H. Huang, L. Tian, Z. Zhang, Y. Liu, Z. Chen, and G. Barbastathis. Path-independent phase unwrapping using phase gradient and total-variation (tv) denoising. *Opt. Express*, 20(13):14075–14089, Jun 2012.

[2] E. Morrison, B. J. Meers, D. I. Robertson, and H. Ward. Automatic alignment of optical interferometers. *Applied optics*, 33(22):5041–5049, 1994.

[3] E. Morrison, B. J. Meers, D. I. Robertson, and H. Ward. Experimental demonstration of an automatic alignment system for optical interferometers. *Applied optics*, 33(22):5037–5040, 1994.

[4] B. S. Sheard, G. Heinzel, K. Danzmann, D. A. Shaddock, W. M. Klipstein, and W. M. Folkner. Intersatellite laser ranging instrument for the grace follow-on mission. *Journal of Geodesy*, 86(12):1083–1095, 2012.

[5] Dieter Meschede. *Optik, Licht und Laser*. Teubner Studienbücher Physik. Vieweg+Teubner Verlag, Wiesbaden and s.l., 2., überarbeitete und erweiterte auflage edition, 2005.