GLAD Users Guide

GLAD has been developed by Applied Optics Research (AOR) to model almost any type of laser or physical optics system with a complete end-to-end analysis, including full diffraction propagation, detailed treatment of laser gain, and many other laser effects.

GLAD is the only commercially available program which is designed to be a comprehensive physical optics analysis tool and is by far the most widely used tool for optical and laser analysis. It is used in several hundred industry and national laboratories, worldwide.

To stop automatic display of Introduction to GLAD, delete demoinfo.txt.
You may access guide.pdf later with Adobe Acrobat Reader.

TABLE OF CONTENTS

Introduction to GLAD 1
Overview and features 3
    Features 5
    Technical support and warranty 14
How GLAD Works 16
How to run GLAD 28
Walk through of a simple resonator model 43
Examples of GLAD 59
Installation, Demo Modes, Distributors 96

Overview and features

GLAD (General Laser Analysis and Design) is the leading program in physical optics and laser analysis. Everyone who works with coherent (or partially coherent) light can benefit from the program.

GLAD has been applied to a wide variety of the most advanced physical optics modeling applications including laser fusion, isotope separation, high energy lasers, free-electron lasers, excimer/Raman and other nonlinear optical systems, stable and unstable resonators, photolithography, single and multiple mode waveguides.

This code may be used to analyze a large variety of optical and laser systems. GLAD performs a complete diffraction analysis of all aspects of a system. Optical beams are represented by complex amplitude distribution. This method gives a much more powerful capability for analysis than is possible with ray tracing programs. The physical optics methods used in GLAD are

essential for analyzing and designing laser systems and for many non-laser systems where diffraction plays an important role. The physical optics description used in GLAD and the way GLAD is organized provide great generality and flexibility so that a wide diversity of systems may be modeled. Simple programs which only model idealized two-mirror bare-cavity resonators provide only very limited and/or inefficient diffraction analysis capability.

GLAD includes most types of optical components including lenses, mirrors, apertures, binary optics, many types of gratings, beam splitters, beam combiners, Fresnel transmission and reflections losses, polarization effects, binary optics, etc. GLAD includes a large variety of nonlinear effects such as laser gain, Raman conversion, four-wave mixing, frequency doubling, thermal blooming, etc. GLAD can model almost any type of optical resonator, including any number of elements, rate equation gain, nonlinear components, branching, etc.

Features

With GLAD, the user is able to model both simple optical systems and highly complicated, multiple laser configurations. The code is designed to analyze all types of beam trains and laser devices including the effects of diffraction, active media, apertures, lenses and mirrors, and aberration.

In GLAD, optical beams are represented by complex amplitude using rectangular computer arrays. This is the most general and powerful approach. Old methods such as geometrical ray tracing, gaussian ray propagation, ABCD methods, and rotationally symmetric propagation methods can not compare in power and versatility.

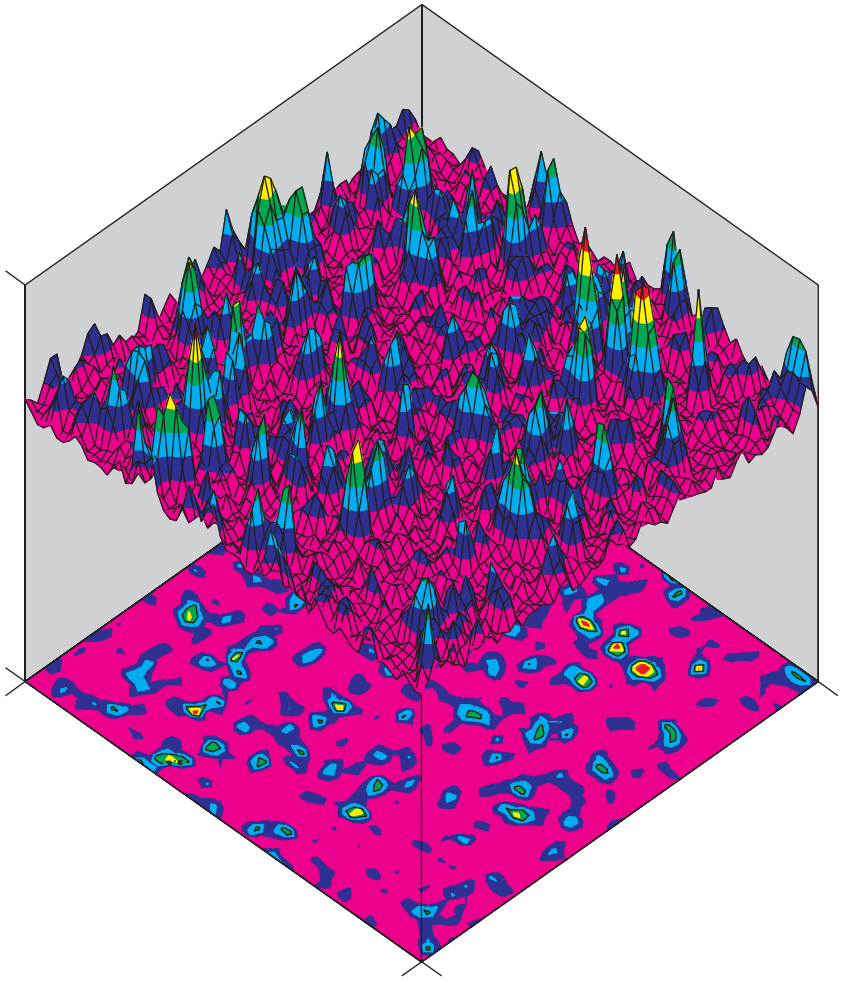
New features

- GLAD now completely converted to a native windows program (WIN32)
- GladEdit now incorporates editing and run features

- Traceback window displays program progress in real-time during execution, macro levels and iteration count, and line numbers
- HTML output: tables, glossary of output variables, built-in HTML viewer
- Separate output controls for writing to screen, text disk file, and HTML
- Watch now supports 20 windows
- Faster generation of plot files: between 4 to 10 times faster
- IDE input window has enhanced text character controls: up-arrow to access old lines, overwrite and insert modes
- Capabilities of GladEdit enhanced: single step, Save/Run from editor



- Install GLAD now to any folder
- Run GLAD now from any folder
- Input files (*.inp) and plot files (*.plt) are now registered extensions. Click on *.inp files to activate GLAD or *.plt files to start Watch.
- GLAD remembers desktop configuration and other user preferences
- Enhanced resolution for bitmap-style graphics
- Thermally-induced stress birefringence
- gradient index lens



- GLAD now allows 128 beam arrays
- 3D arrays may be defined
- New examples:
  - Michelson white light interferometer
  - Point diffracting interferometer
  - Moire interferometry for measuring spatial coherence of excimer laser
  - Thermally-induced stress birefringence
  - New reflecting wall waveguides models: resonators with internal waveguides and tapered, curved, cylindrical, pentagonal waveguides
  - Resolved and non-resolved gratings
  - 3D array capabilities.

System analysis features

- Simple conventional systems or complex multiple laser beam trains
- Coherent and incoherent interactions
- Nonlinear gain models
- Lenses and mirrors: spherical, toroidal or cylindrical

- General aperture shapes
- Near- and far-field diffraction propagation
- Stable and unstable resonator modeling
- Aberration effects including: Seidel, Zernike, phase grating, and smoothed random wavefronts
- Lens and mirror arrays
- Variable size arrays to 4096 x 4096 and beyond
- dual CPU capability under Windows NT
- Rectangular arrays and separable diffraction theory
- Propagation of multiple, independent laser beam trains
- Automatic propagation technique control (may be overridden)
- Special features for resonator design
- Gain sheets, rate equation kinetics
- Global coordinate system and geometrical aberrations
- Arbitrary mirror locations and rotations
- Raman, four-wave mixing, frequency doubling

- High Fresnel numbers
- Kolmogorov model of atmospheric aberration
- Thermal blooming
- Self-focusing effects
- Zonal adaptive optics model
- Phase conjugation
- Polarization modeling
- Partially coherent modeling
- Variable index of refraction modeling
- ABCD propagators
- GRIN elements
- Fiber optics and 3-D waveguides
- Binary optics and gratings
- Vector diffraction for high NA objective lenses
- M^2 characterization
- phase retrieval method

- finite-element thermal modeling

Optimization

- Least squares optimization of any configuration
- User-defined merit functions
- All system parameters may be used as optimizing variables
- Phase retrieval and simulated annealing

Geometrical optics

- Lens groups may be defined and analyzed using conventional geometrical optics methods

User Interface

- Integrated design environment (IDE)
- Interactive command structure
- Graphical displays: isometrics, profiles, polarization, contour plots
- Enhanced graphics: bitmaps, combined isometric and contour plots, Windows printing and metafiles

- Utilities for conversion of graphics to Postscript, Windows metafiles (*.wmf), and Common Graphic Metafile (*.cgm).
- Macros of commands
- Algebraic expressions and user-defined variables in commands
- Interface with user programs for pre- and post-processing
- More than 100 examples of all types of systems

Command language

GLAD has a simple but powerful command language so that problems can be set up rapidly and conveniently. To facilitate learning the command language, numerous examples are provided in the Examples Manual. To aid in modeling complex systems, a sophisticated macro capability is provided.

Automatic algorithm selection

GLAD makes diffraction calculations easy by handling the details needed for accurate numerical analysis. Diffraction calculations employ different algorithms for near- and far-field calculations and for points in between. GLAD selects the appropriate algorithm (or combination of algorithms) automatically to achieve best numerical accuracy.

Easy to use and learn

GLAD has been engineered to be easy to use. Commands are mnemonic. The program may be run interactively or in conjunction with files of commands.

GLAD is surprisingly easy to learn. You can begin working immediately from the command files of any of the more than 90 examples which are supplied with GLAD.

Instructional course on GLAD

The thousands of GLAD customers have found the code is readily learned from the numerous examples, comprehensive documentation, and technical support. To provide additional opportunities for learning, a three-day course on GLAD will be taught at various times. The schedule of courses will be posted on the AOR web page: www.aor.com. The course will provide a thorough grounding in the principles of physical optics and laser modeling and extensive hands-on experience in solving problems.

Technical support and warranty

AOR provides free technical support and warrant for one full year from the date of purchase or until the date of release of the next upgrade, whichever is longer. For international customers e-mail and fax allow quick and convenient support because of GLAD’s text-based command format.

Technical support for GLAD on all computer platforms is provided by AOR. If you have questions about a GLAD input command file, you may email that file to AOR to receive technical support. Information about product upgrades will be posted on the AOR web site. Updates are available for licensed users from ftp://ftp.aor.com/pub.

AOR warrants the program for one full year warranty from the date of purchase or until the date of release of the next upgrade, whichever is longer. Any operational defects will be repaired at no cost during the period of warranty.

How GLAD Works

As the complexity and variety of laser systems has expanded over the years, the need for powerful analytical methods has become increasingly important. Optical engineers and scientists need to be able to accurately calculate performance in order to optimize designs and to determine system tolerances.

Numerical analysis is quicker and less expensive than laboratory experiments and also serves as an educational tool. The optical engineer or scientist can determine the end-to-end performance of a complex device based on the characteristics of the lenses and mirrors, propagation distances, apertures, aberrations, laser gain, and other effects. The definition of the system components can be very detailed, including exact aperture shapes and the accurate aberration determined from interferometry or other means of measurement. A relatively complete description of the laser beam can be deter-

mined by the intensity and phase profiles. This information can be used to find the total power, the peak power, wavefront quality, wavefront variance, Strehl ratio, and properties of the focused beam.

Prior to the advent of lasers, optical analysis consisted largely of geometrical ray tracing for the design of photographic systems. Diffraction analysis was applied to various types of apertures from the turn of the century but generally not to the analysis of systems.

The laser was the stimulus for physical optics calculations. In geometrical analysis, the light is represented by a set of rays which are normal to the wavefront as shown in Fig 2.1a. For short propagation lengths such as are encountered in a common photographic lens, the diffraction effects are small and lo-

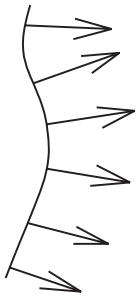


Fig. 2.1a. Representation of an optical beam by rays. The rays convey optical path difference errors and slight differences in ray direction indicate ray aberrations. Rudimentary energy density calculations may be done except near the focus.

calized to the edge of the beam. For this type of problem, rays do a good job of determining the aberrations of the system and a reasonably good job of determining the intensity variations.

For a conventional optical system the rays enable us to calculate the aberrations. These aberrations may be used to determine the pupil function and a simple far-field diffraction analysis may be made or, if the system is not diffraction-limited, the rays may be traced to the image plane and the geometric image size may be used as a measure of performance. Associating equal energy with each ray, we can get a rough estimate of the energy density but this method breaks down the region of the focus.

Consider a simple spatial filter which is common in many laser systems. Ray optics can approximately calculate the image at the focal point where the pinhole filter is placed. Ray optics is unable to predict removal of the phase aberrations by the spatial filtering and smoothing of the intensity distribution. Figure 2.1b. shows two lenses and a pinhole aperture at the inter-

mediate focus which act as a spatial filter. This simple device is used in many laser systems to remove the aberrations and to smooth out intensity variations. Using geometrical analysis, we may be able to approximately determine the image size but we cannot determine the reduction of aberrations and the change in the intensity distribution to be found in the expanded beam after the spatial filter. The spatial filter, like many other common components, can not be analyzed by geometrical analysis.

Physical optics analysis describes the optical beam by a complex amplitude function, describing the transverse beam distribution. The complex amplitude includes both the intensity and phase information of the beam at one axial position. This information can be modeled in the computer by a complex two-dimensional array where each point of the array corresponds to

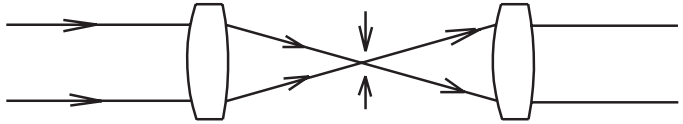


Fig. 2.1b. A simple spatial filter is commonly used in many laser systems. Rays optics can approximately calculate the image at the focal point where the pinhole filter is placed. Ray optic is unable to predict removal of the phase aberrations by the spatial filter and smoothing of the intensity distribution.

a point on the beam. We may use a single two-dimensional array if polarization analysis is not needed or separate arrays for two orthogonal polarization states, as shown in Fig. 2.2

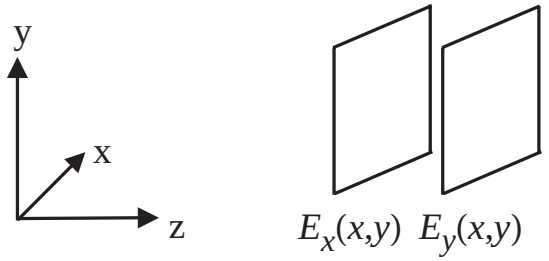


Fig. 2.2 The intensity and phase are determined by two-dimensional arrays of complex amplitude. Separate arrays for E_x and E_y. are used for orthogonal polarization states. Here x and y are the transverse directions and propagation is along the z-direction.

The earliest work in resonator analysis codes was done for optical communications in the 1960's. The military interest in high energy lasers stimulated intense development of physical optics modeling codes in the mid 1970's. The work by Siegman and Sziklas in 1974 and 1975 studied gas dynamic lasers including diffraction, the active gain medium, apertures, and aberration. The first paper by Siegman used an Hermite-Gaussian expansion for propagation. The second paper used a fast Fourier transform (FFT) method for propagation. A third

method based on finite-difference propagation a direct solution to the differential equation of diffraction was used by Rench and Chester in 1974.

Over time, the FFT method that is the primary method used in GLAD has become the mainstay of optical propagation codes, as much for its well-understood insensitivity to error as for its computational efficiency.

Review of physical optics modeling

GLAD is designed to calculate the performance of laser systems and other optical systems which have a well defined direction of propagation. GLAD represents the optical beam by the complex amplitude of the optical wavefront. This is distinct from geometrical optics codes which represent the optical beam by rays. Geometrical codes are very useful when near-field diffraction and gain are not important and where the transverse intensity distribution of the beam is either constant or some simple function, but can not model general cases.

Typical types of analysis

To illustrate application of the code, consider the schematic shown in Fig. 2.3. The configuration does not represent any particular system, although it has some resemblance to a Raman amplifier. Many of the important features of laser systems are present. GLAD allows many different ways of defining the starting distribution.

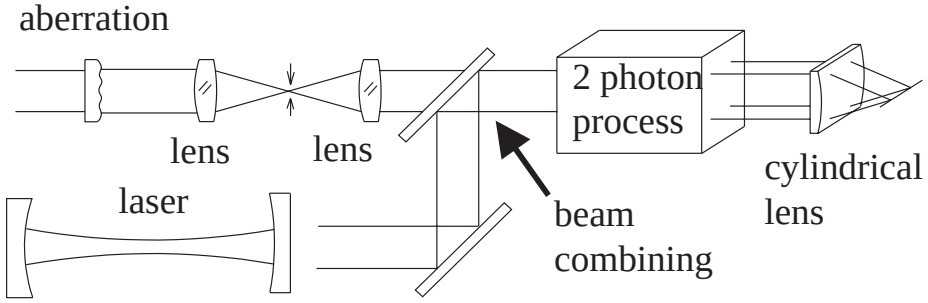


Fig. 2.3 Schematic of a “typical” beam train, showing aberration, lenses and mirrors, beam combiners, laser resonators, beam combiners and splitters, nonlinear optics. Physical optics modeling is needed to design and analyze such a system.

GLAD assumes no particular symmetry to the optical distribution. The distribution may be decentered and unsymmetrical in other ways.

The top beam of Fig. 2.3 is shown with an aberrated element. A large variety of types of aberration may be used in GLAD. The lens in the top beam brings the light to a focus. GLAD may be used to calculate the distribution at any point in the collimated, converging, or diverging part of the beam. An aperture at the focus of the lens acts to spatially filter the distribution to remove some of the aberration. The second lens recollimates the beam.

The lower beam is assumed to be generated by a laser and then combined with the upper beam by a beam combiner. The combined beams interact in a second medium. This might be a two-photon process such as Raman scattering cell.

After the two-photon process, a cylindrical lens is shown. Glad may be used to model spherical, cylindrical, or toroidal optical elements.

GLAD is designed to be very modular. With basic modules such as diffraction steps, lenses, mirrors, apertures, beam splitters, beam combiners, and gain media; a large variety of optical systems may be analyzed.

Discussion of three-dimensional modeling

GLAD was developed as a three-dimensional code modeling two transverse dimensions by the two-dimensional computer arrays and the axial dimension by successive calculations, as indicated in Fig. 2.2. In general, a four-dimensional solution may be required because of temporal dependence of the optical beam. Many systems may be approximated by steady-state solution, because the temporal pulse width is longer than the time constants of all processes in the system. GLAD is well suited to analyze this type of problem. Other systems have short temporal pulse widths. If the pulse width is shorter than the time constants of all the processes in the system, then the exact waveform of the pulse does not play a role: only the inte-

grated effect need be used in modeling. The beam may be represented by fluence in terms of joules per square centimeter.

With the increasing availability of very fast workstations, time-dependent, three-dimensional problems may be solved in reasonable time. Example 27 illustrates examples of jitter including the pulse-to-pulse variation and time-integration of the energy. Example 56 illustrates the spectral performance of a Fabry-Perot cavity.

The most difficult problems are ones where the temporal pulse shape plays an important role. GLAD has been applied to a variety of transient, three-dimensional problems. Example 79 illustrates transient Raman analysis and Example 80 describes a time-dependent analysis of a Q-switched laser. Partial coherent effects are shown in Example 83.

Three-dimensional waveguides may be modeled, as illustrated in Example 86. The three-dimensional model consists of two transverse dimensions (x,y) and the propagation direction z. This allows modeling of very general

waveguides including straight guides such as fiber optics, bent guides, splitters, combiners, coupled guides, and fiber lasers. Figs. 2.4 and 2.5 show an s-bend waveguide, where the higher index region is bend in the shape of an “s”. This index distribution is actually a circular distribution in the transverse plane, but only profiles are shown in Fig. 2.4. The guided mode shown in Fig. 2.5 shows the transient behavior of the mode as it travels through the s-bend. Example 86 presents several waveguide applications.

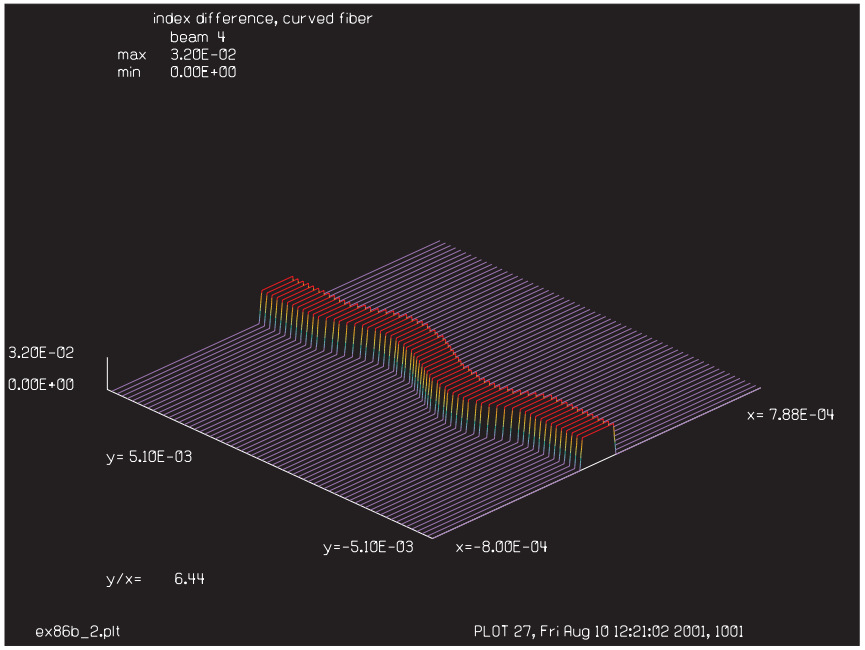


Fig. 2.4 Waveguide consisting of a high index core curved in an “S” shape.

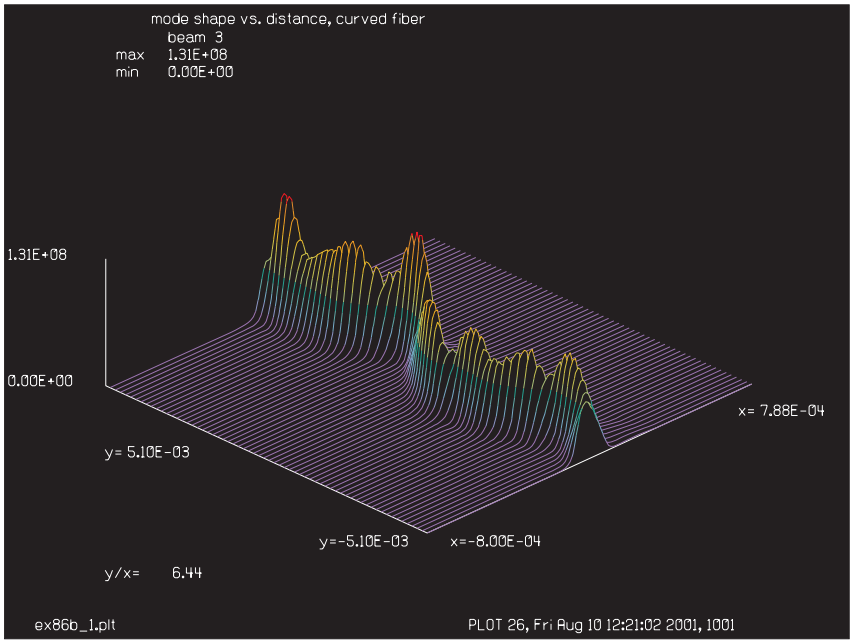
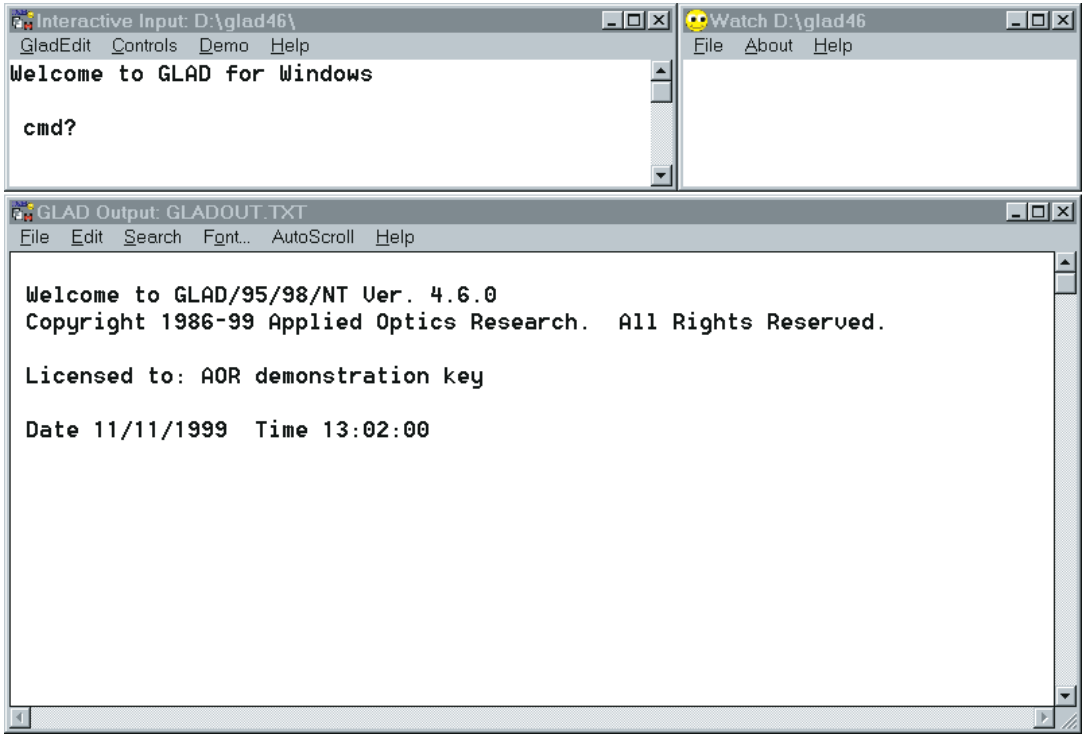


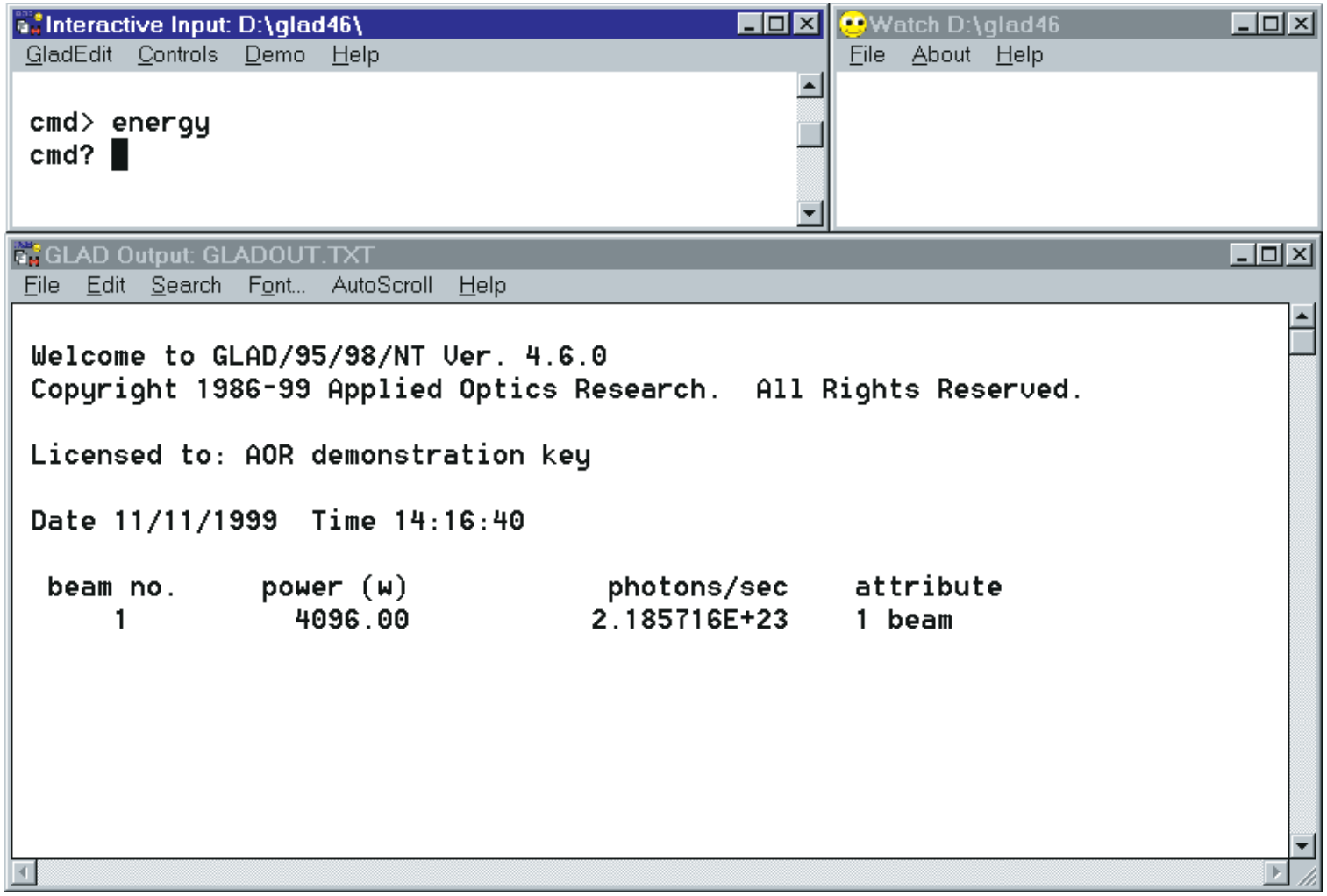
Fig. 2.5 Optical mode in the s-shaped waveguide.

How to run GLAD

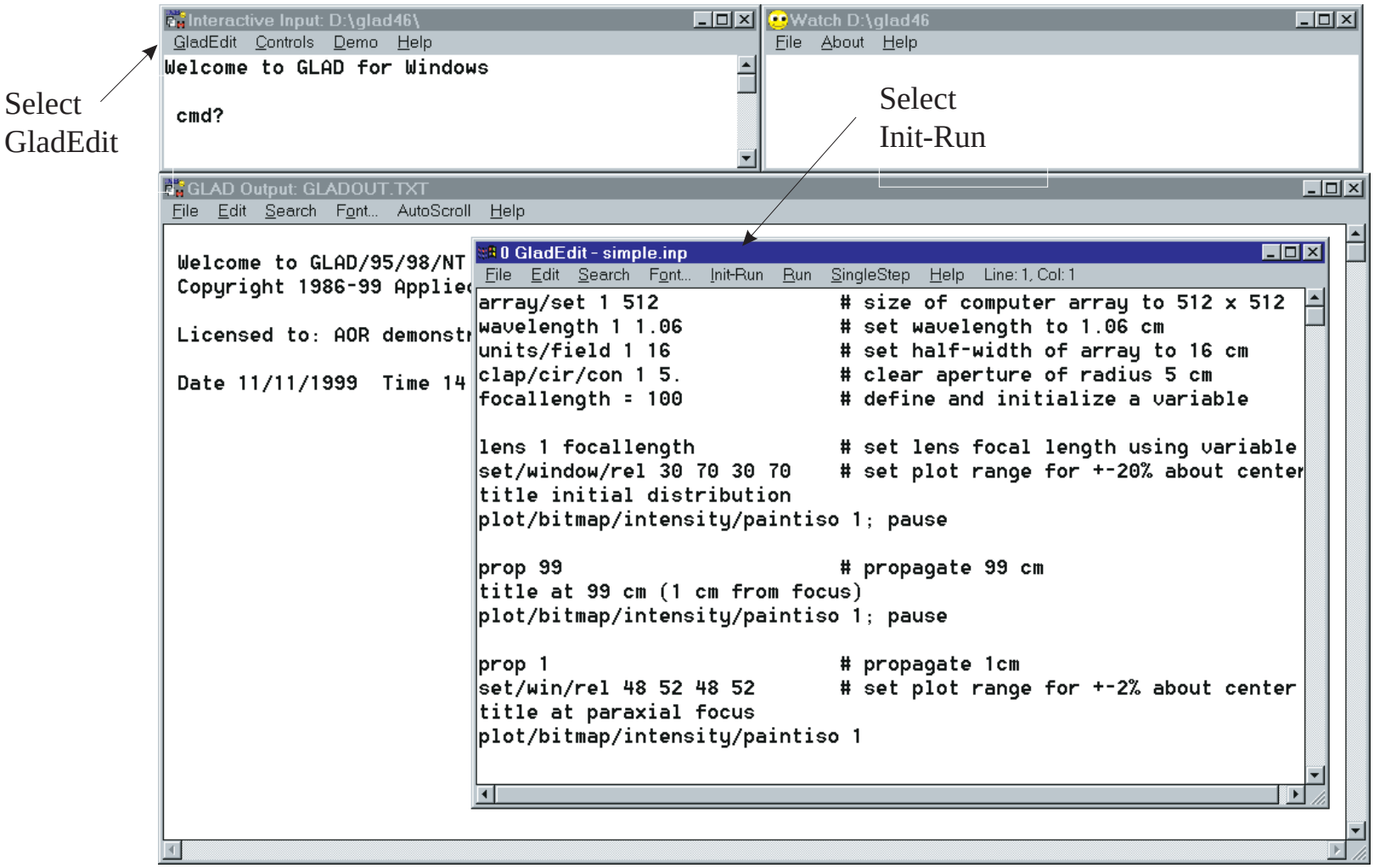
From Windows: Start, Programs, GLAD 4.6, GLAD IDE to run GLAD.



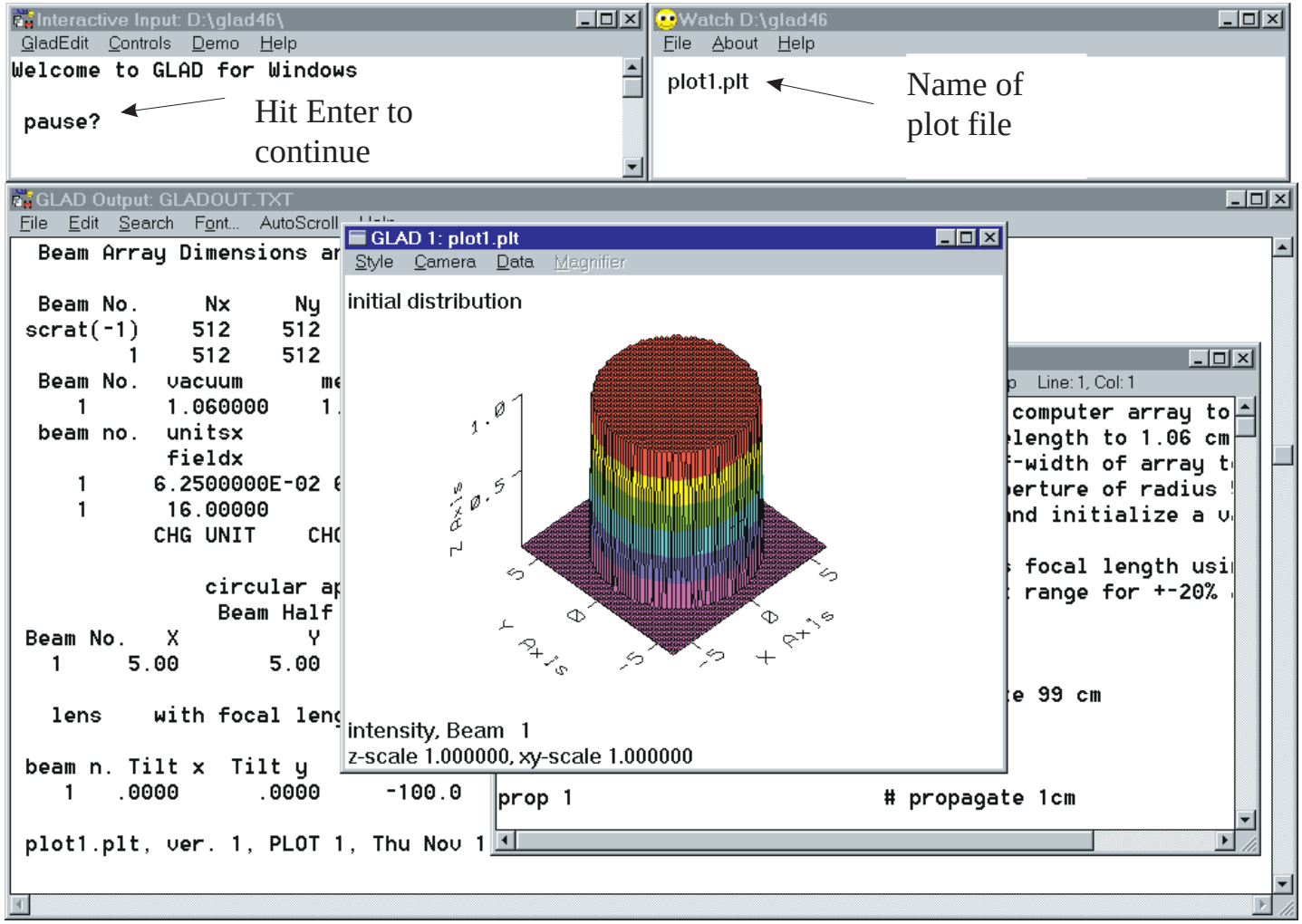
Typical starting configuration after clicking on GLAD IDE. The interactive window is on the upper left. Enter interactive commands into this window. Text output appears in the GLAD Out window below. The Watch window displays and controls graphic files as they are created by GLAD.



Entering the command “energy” into the interactive window causes GLAD to list the current integral over the array.



To create a command file or open an existing file, select GladEdit from the menu bar of the Interactive Input window. A version of GladEdit will be selected. Use File to open an existing command file. In this case simple.inp was opened. This file may be edited. Use Init-Run to reinitialize and run GLAD.

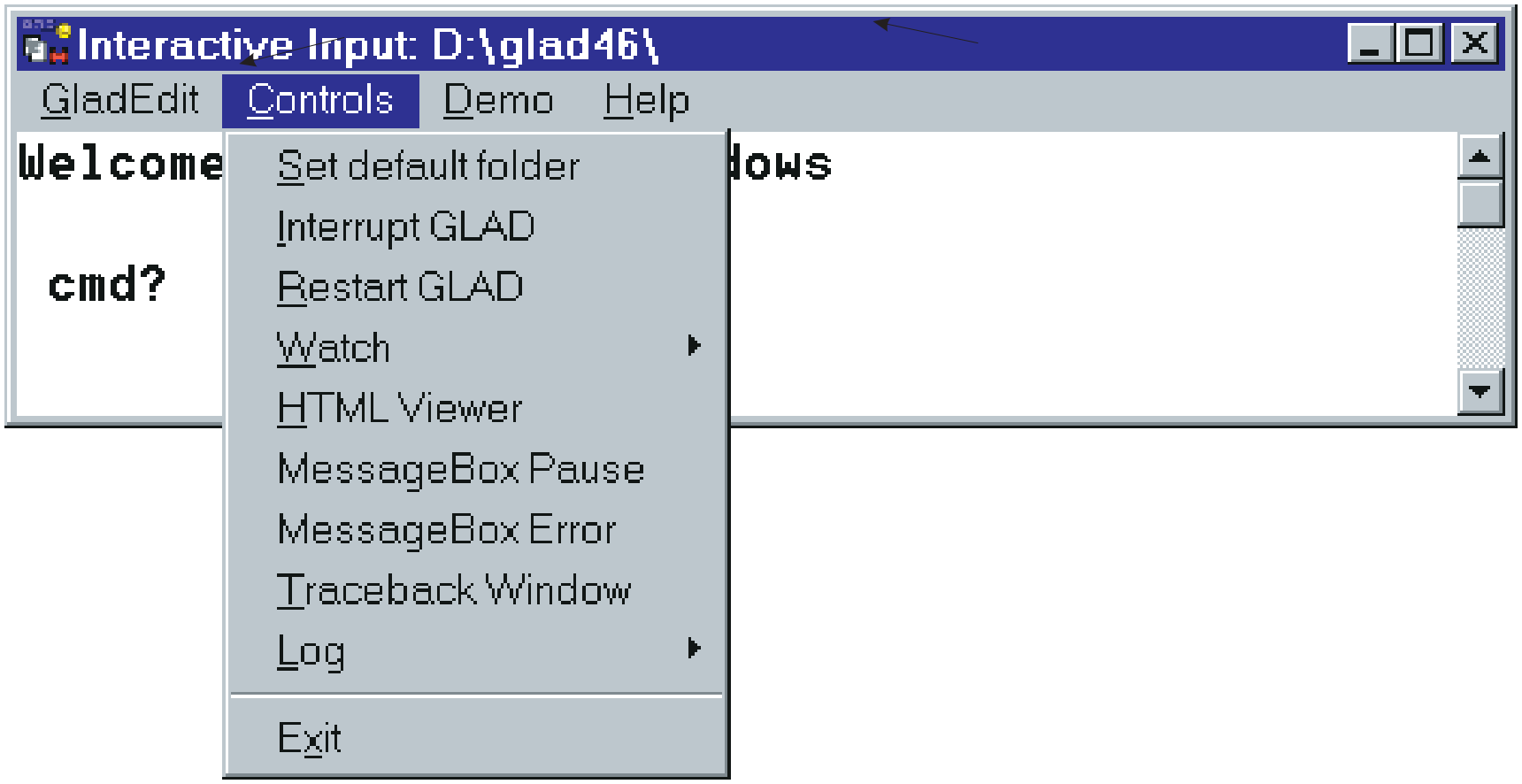


Running simple.inp by selecting Init-Run, we have a graphic file displayed and “pause ?” in the interactive input window. This pause command was written into simple.inp to require a response from the user before proceeding. Put the cursor into the Interactive Input window hit Enter to continue. Note the plot name “plot1.plt” in the Watch window.

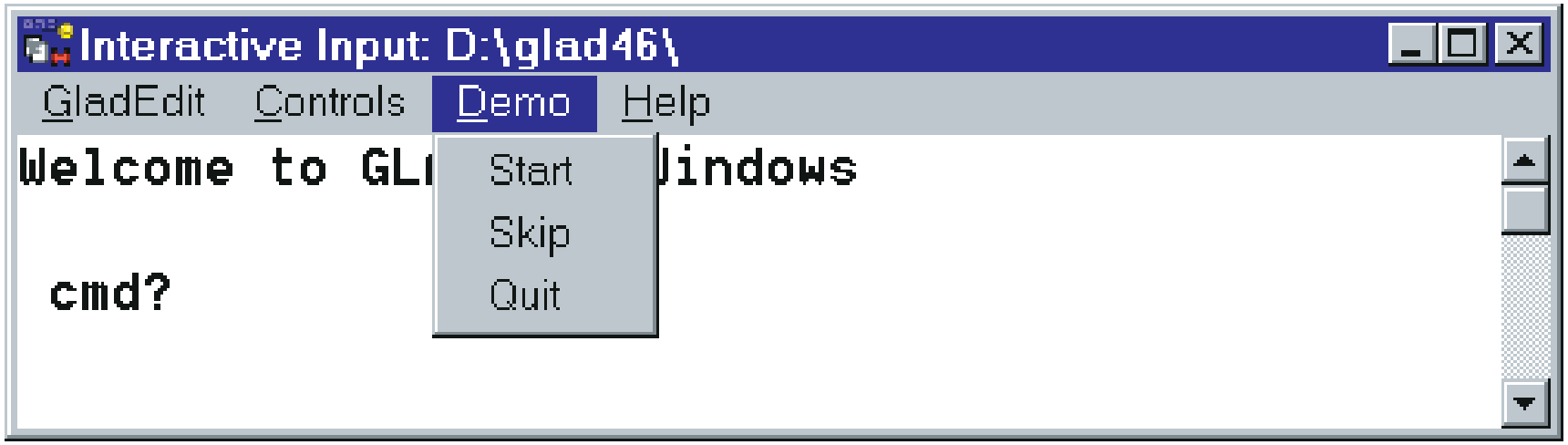


Command files may be entered directly in the Interactive Input window. “read/disk simple.inp” will read simple.inp and execute the commands as they are read.

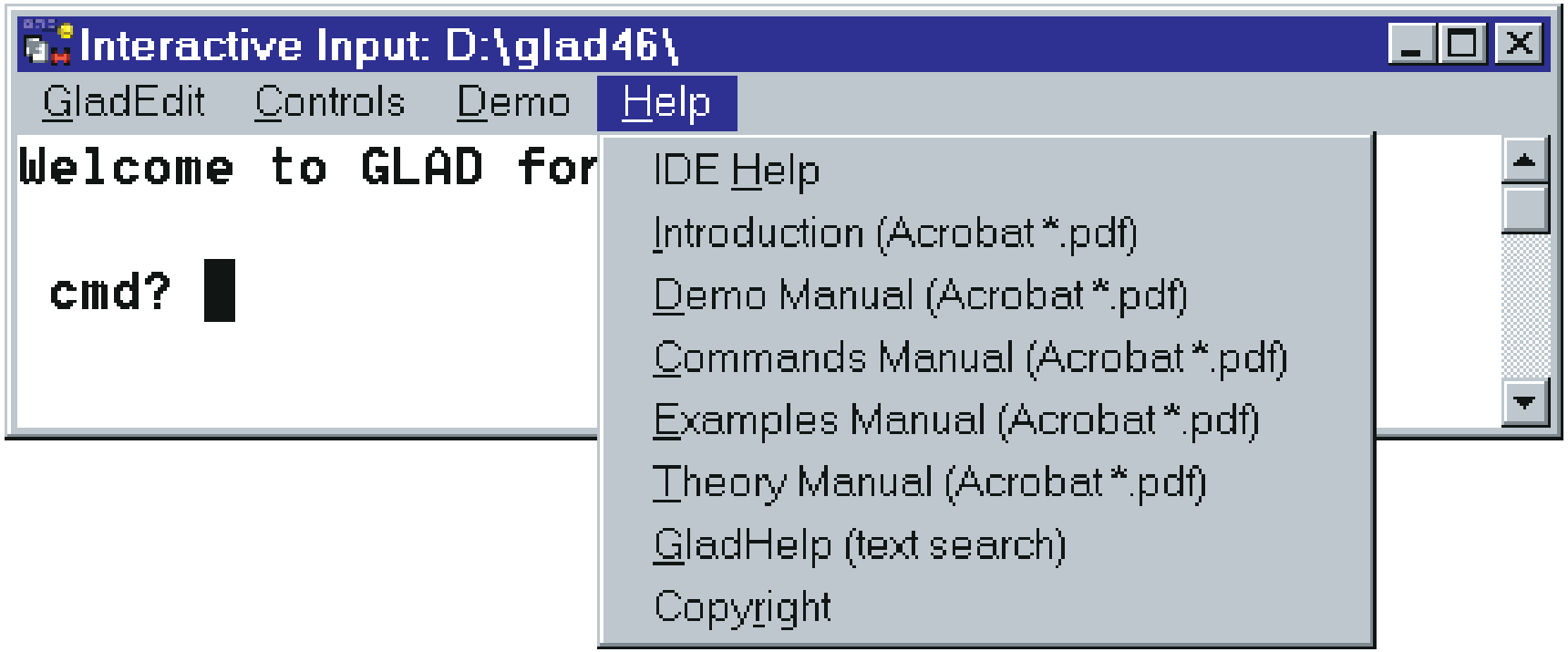
One can read command files directly from the command prompt in the Interactive Input window. The default directory is set under Controls (see next page).



The Controls menu item allows selection of a number of operations. See Help for a detailed explanation. Use “Set default folder” to select the folder for GLAD to work from.



The Demo menu item runs nine preselected examples. Select Start to begin the demo, Skip to skip to the next example, and Quit to end the demonstration. See Demo.pdf in the installation folder for a description of the examples.



IDE Help gives specific information about operating GLAD IDE. Details about the commands, examples, and theory are in the respective PDF files, viewed with the Adobe Acrobat Reader.

GLAD is designed to handle virtually any physical optics analysis problem from the simplest to the most complex. The GLAD command language provides a convenient and comprehensive capability to describe physical optics systems and operations. Since optical systems may be very complicated, it is important to be able to develop a comprehensive model in simple steps.

Online manuals

GLAD is comprehensively documented in three volumes (more than 1,800 pages). The complete set of manuals are on the CD ROM and may be viewed by the Adobe Acrobat Reader that is provided on the CD ROM. These files will be found in the main GLAD folder (\glad47 by default):

- GLAD Commands Manual commands.pdf
- GLAD Theoretical Description theory.pdf
- GLAD Examples Manual examples.pdf, examples_print.pdf
- GLAD Demonstration Examples demo.pdf
- GLAD Users Guide guide.pdf

The easiest way to use GLAD is to start by working some of the simpler examples. You will want to refer frequently to the online documentation on the CD ROM, particularly the Commands Manual and Examples Manual. You will find an explanation of a simple example in Chapter 4 of this Users Guide.

Often the process of system description consists of building a command file description by including the the part of the system that is already understood and then switching to interactive operation to examine the current state of the optical beams by graphical displays and text information and to test commands and operations. The command file may then be extended to include the new information derived from the interactive session. A simple command sequence is:

```
array/set 1 128    # set size of computer array to 128 x 128
wavelength 1 1.06  # set wavelength to 1.06 cm
units 1 .1         # set spacing between matrix points to be
                   # 0.1 cm
```

```
clap/cir/con 1 5.  # clear aperture of radius 5 cm
lens 1 100         # lens of focal length 100 cm
prop 100           # propagate 100 cm to focus
```

The simple command sequence above sets the size of the computer array of Beam 1 to be 128 x 128. Anything after “#” is a comment. The spacing between matrix units is set to 0.1 cm so the whole array spans 12.8 cm. A circular clear aperture of radius 5 cm is defined. An idealized

lens of focal length 100 cm is defined to cause the light to become converging. The beam is then propagated 100 cm to the paraxial focal point.

The command file makes it very easy to do either simple or complex problems. We can easily include variables, macros (similar to subroutines or functions), looping and branching, and many other aspects of a programming language.

The command language format makes it very easy for AOR to provide technical support. A customer may send a command file to AOR by email and we will see exactly the same performance from GLAD. We can then make corrections or comments and send back a modified email file which the customer may easily use.

Variables may be included:

```
focallength = 100     # define and initialize a variable
lens 1 focallength    # set lens focal length using variable
```

Equations may also be included:

```
x=2.
y=10.
lens 1 [y^x]   # sets focal length to 100 cm
```

The command language incorporates a powerful macro capability, similar to user-defined subroutines or functions. The macro capability allows almost any type of optical resonator to be defined no matter how complex.

The macro capability also is the basis for GLAD's system optimization capability with which and combination of system features may be optimized to achieve user-defined performance targets. The command language includes a powerful capability to define variables and use mathematical expressions that greatly facilitates use of the program. Here is an example of a simple macro

```
variable/dec/int pass           # declare "pass" as integer
macro/define simple_macro       # start definition of macro
   pass = pass + 1              # increment pass counter
   prop [2.*pass]               # propagate
   plot/watch plot@pass.plt     # define a plot file name
   plot/liso/int                # plot intensity
macro/end
macro/run simple_macro/3        # run macro 3 times
```

This macro propagates distances 2, 4, and 6 cm and creates plot files plot1.plt, plot2.plt, and plot3.plt.

Macros may be called from within macros allowing very complex problems to be organized into separate functions.

HTML Output

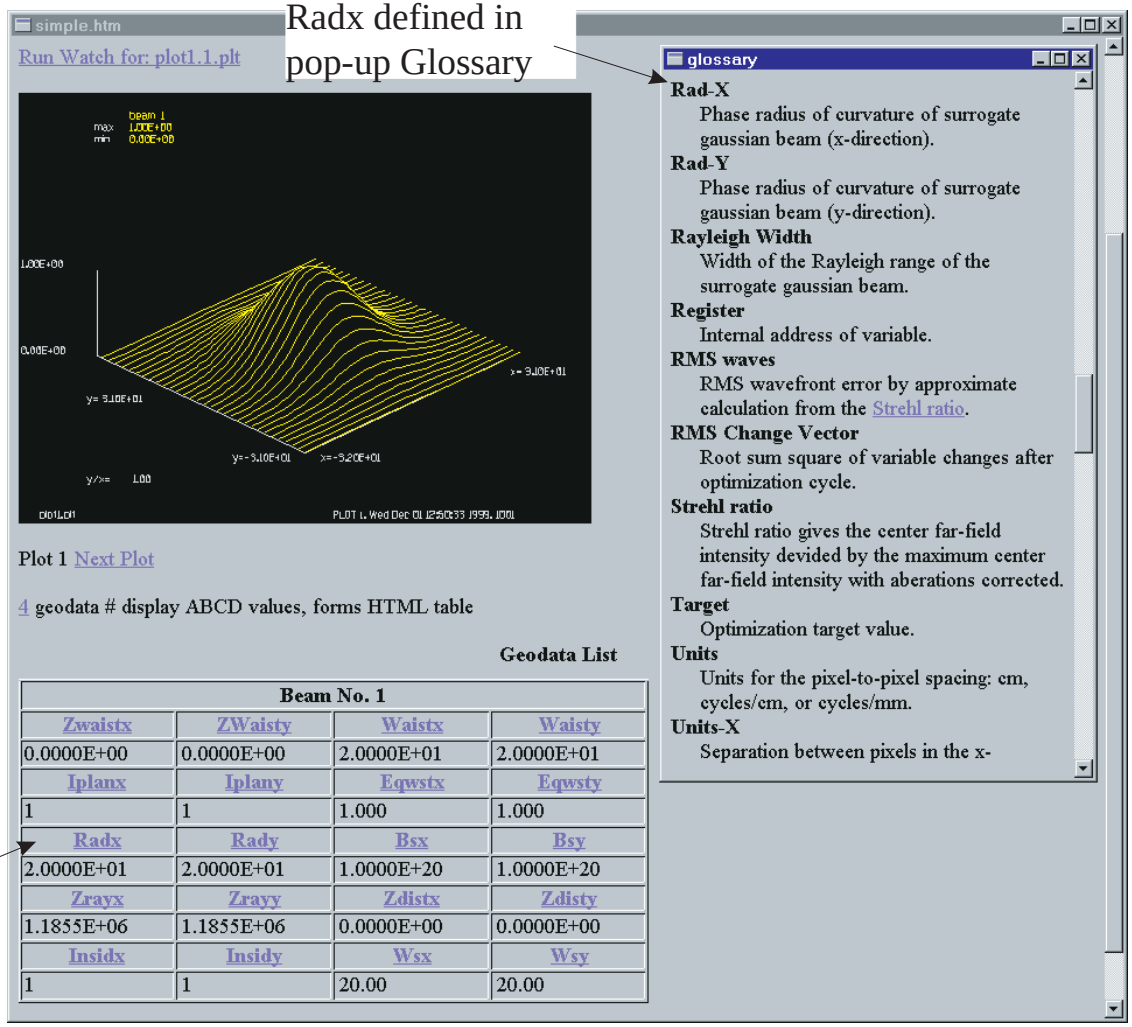
GLAD now provides HTML output and a built-in HTML viewer (or use Internet Explorer 5.0 or higher). Most of the important table-style output, for example `geodata`, is presented as an HTML table for easier viewing. The items in the table are defined in an associated glossary. One simply clicks on the item to start the pop-up glossary at the showing the definition of the item. Graphic files are displayed in the proper sequence among commands and tables. See html.inp below.

```
gauss/c/c 1 1 20            # make a gaussian beam
html/write/on simple.htm    # start writing to html file: simple.htm
html/wmf/on                 # start output of plots as WMF files
html/viewer/start           # start GLAD HTML viewer
plot/l                      # make a plot geodata
geodata                     # display GEODATA values, forms a table
```

Graphics are placed in sequence

Table output

Item Radx



HTML output genetaed by html.inp. Graphics are displayed in the proper sequence with commands and table output. Tables are provided for most of the important commands. Here `geodata` is used. A pop-up glossary is shown defining the item “X-rad”.

Walk through of a simple resonator model

A simple example will show how one uses the GLAD program. A simple resonator problem has been selected (resonator.inp). Even if the reader is not interested in resonators, this example will show some important parts of solving problems with GLAD:

- initializing the computer arrays and the units
- selecting the wavelength

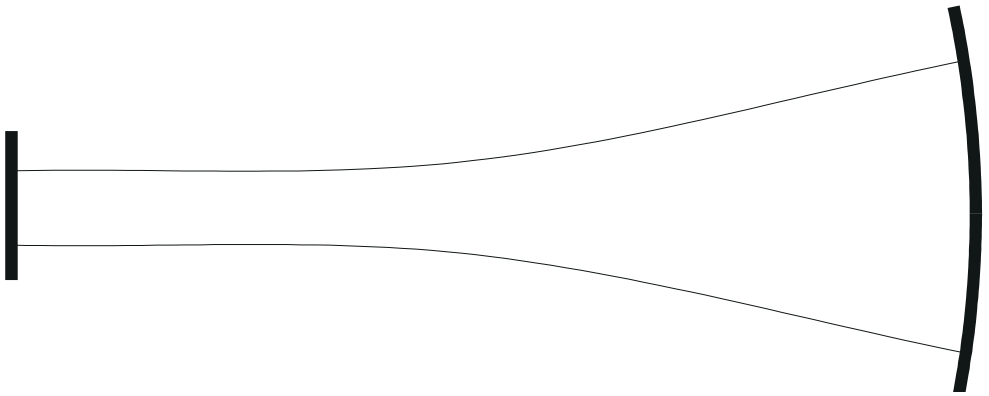


Fig. 4.1. Stable resonator configuration. The waist will form on the flat mirror and the phase radius will match the radius of the concave mirror for the ideal mode.

- defining the starting distributions
- use of macros for repetitive operations
- generating data to show the progress of a calculation

Table 4.1. Parameters	
length	45 cm.
mirror radius	50 cm.
wavelength	1.064 microns
Rayleigh range	15 cm.
waist radius	.02254
aperture radius of spherical mirror	.14 cm.

The resonator to be analyzed is half-symmetric consisting of a spherical mirror of radius 50 cm and a flat mirror. The length of the resonator is 46 cm. The output will be taken from the flat mirror. Table 4.1 summarizes the system.

To simplify the discussion we will neglect the gain and perform what is called a bare-cavity analysis. We start the analysis by preparing a command file as shown below.

Description of the command file (resonator.inp)

```
variab/dec/int pass
macro/def reson/o
   pass = pass + 1                  # increment pass counter
   prop 45                          # propagate 45 cm.
   mirror/sph 1 -50                 # mirror of 50 cm. radius
   clap/c/n 1 .14                   # .14 cm. radius aperture
   prop 45                          # propagate 45 cm. along beam
   mirror/flat 1                    # flat mirror
   variab/set Energy 1 energy       # set variable to energy value
   Energy = Energy - 1              # calculate energy difference
   udata/set pass pass Energy       # store energy differences
   energy/norm 1 1                  # renormalize energy
   plot/l 1 xrad=.15                # make a plot at each pass
macro/end
array/set 1 64                      # set array size
wavelength 0 1.064                  # set wavelengths
units 1 .005                        # set .005 cm sample spacing
resonator/name reson                # set name of resonator macro
resonator/eigen/test 1              # find resonator properties
resonator/eigen/set 1               # set surrogate beam to eigen mode
clear 1 0                           # clear the array
noise 1 1                           # start from noise
```

```
energy/norm 1 1                    # normalize energy
pass = 0                           # initialize pass counter
reson/run 100                      # run resonator 100 times
title Energy loss per pass
plot/watch plot1.plt               # set plot name
plot/udata min=-.05 max=.0         # plot summary of eigenvalues
title diffraction mode shape
set/density 32                     # set plot grid to 32 x 32
set/window/abs -.05 .05 -.05 .05 # set plot window
plot/watch plot2.plt               # set plot name
plot/iso 1                         # make an isometric plot
```

Let us consider each line in turn:

```
variab/dec/int pass
```

This line define an integer variable called `pass`. We will use `pass` to store information in a summary table. Variables that are not explicitly defined as integers will be established as real variables.

```
macro/def reson/o
```

This line starts the definition of a macro, which is like a subroutine or function. All lines between `macro/def` and `macro/end` are part of the definition of the macro. These lines are not executed at this time. The lines are placed into a file called MACLIB for later use. The lines in the body of the macro do not have to be indented. Indenting is done to make the macro easier to read.

```
pass = pass + 1                 # increment pass counter
```

This line increments the variable `pass`. It is a simple equation. We are using `pass` to count the number of times we have gone through the macro. The characters after “#” are comments. It is a very good idea to use many comments to make it easy to understand the command file.

```
prop 45                          # propagate 45 cm.
```

This line implements diffraction propagation of 45 cm. The diffraction propagation is usually the most time consuming step. However, with modern PC’s an array of 64 x 64, such as used in this example, will take very little time in diffraction propagation.

This propagation of 45 cm takes the beam from the flat mirror on the left, as shown in Fig. 4.1, to the curved mirror on the right.

```
mirror/sph 1 -50                 # mirror of 50 cm. radius
```

This command implements a spherical mirror of radius 50 cm. The “1" specifies that the mirror will be applied to Beam 1. There can be up to 40 beams, but only one is used in this analysis. The minus sign, in this context, makes the mirror concave. The concave mirror makes the light converge and reverses the direction of the light.

```
clap/c/n 1 .14                  # .14 cm. radius aperture
```

This command makes a circular clear aperture of 0.14 cm for Beam 1. Apertures are very important in resonators as they scrape off the widely scattered light. Over time, the apertures will clean up the beam leaving only the lowest loss mode.

```
prop 45                         # propagate 45 cm. along beam
```

This is the second propagation step and takes the beam from the spherical mirror on the right to the flat mirror on the left.

```
mirror/flat 1                   # flat mirror
```

This command represents the flat mirror on the left for Beam 1. For bare-cavity resonator analysis, the beam is simply directed back to the right. In a real laser, this mirror would be made partially transmitting so that some of the beam exits.

```
variab/set Energy 1 energy
```

The variable `Energy` is set to the value of the total energy (really power) in Beam 1. We do not have to declare real variables as they will automatically be registered by GLAD when they are first used in a defining statement.

```
Energy = Energy - 1           # calculate energy difference
```

This simple equation subtracts 1 from `Energy` to determine the loss-per-pass.

```
udata/set pass pass Energy    # store energy differences
```

This line stores the value of `Energy` in an array using the `udata` command. The first use of `pass` indicates the point in the data array to be set. The second use of `pass` sets the abscissa, see Fig. 4.2.

```
energy/norm 1 1                 # renormalize energy
```

This line renormalizes the energy in the resonator to unit. In a real laser, the energy lost by apertures and other effects, is balanced by the energy gain due to the amplifying medium, under steady-state conditions. In bare-cavity analysis, such as is being done here, we simulate steady-state gain simply by renormalizing the gain at the end of each pass.

```
plot/l 1 xrad=.14               # make a plot at each pass
```

Plot the cavity distribution with an isometric plot at each pass to show mode formation vs. time.

```
macro/end
```

This line ends the definition of the macro.

```
array/set 1 64                    # set array size
```

This command is the first in the main body of the calculation. The arrays size is being defined for Beam 1 as 64 x 64. An array of any size may be defined. For a simple stable-cavity resonator of this type, a small array is sufficient for good accuracy because only low order modes will be important.

```
wavelength 0 1.064                # set wavelengths
```

This line defines the wavelength for Beam 1 to be 1.06 microns.

```
units 1 .005                      # set array size
```

This line specifies the spacing between array points to be 0.005 cm so that the total array of 64 x 64 size will span a region of .32 cm.

```
resonator/name reson              # set name of resonator macro
```

This line identifies the macro named “reson” as constituting the resonator to be analyzed.

```
resonator/eigen/test 1          # find resonator properties
```

This line tests the resonator to determine its elementary properties. GLAD uses this information to determine the numerical algorithms to be use. It is very important that the numerical algorithms be performed in exactly the same fashion on each pass. The intensity and phase of the beam will, of course, be varying on each pass, but the algorithms must be performed identically to achieve correct results.

```
resonator/eigen/set 1           # initialize surrogate beam
```

This line sets the beam initially to be the lowest loss mode as determined by `resonator/eigen/test`. This ensures that a surrogate gaussian beam that is used to control the numerical algorithms is set up properly. We can change the data in the beam as is done in the next two lines.

```
clear 1 0                          # clear the array
noise 1 1                          # start from noise
```

The first of these lines sets the entire beam array to zero. The second line puts random noise into the array to simulate the random noise associated with spontaneous emission. Most lasers start from spontaneous emission, so this approach gives a more realistic starting condition than a simple plane wave. Of course, the steady-state solution should be the same no matter what condition we start with.

```
energy/norm 1 1                    # normalize energy
```

This line adjusts the intensity in the beam, without changing its shape, so that the total energy (or power) is unity. We will measure the energy at each pass and subtract 1 so that the difference represents the energy loss.

```
pass = 0                           # initialize variable
```

The pass counter is set to 0 at the start of the calculation.

```
reson/run 100
```

This line causes GLAD to run through the resonator macro 100 times. We may need more or fewer than 100 passes to get close to steady-state performance.

```
title Energy loss per pass
```

This line defines a title to be used in the subsequent plots.

```
plot/watch plot1.plt              # set plot name
```

This line defines the name of a plot file. The subsequent plots will be made to this plot file. The Watch program will automatically display this plot information and will automatically update the data when new plots are made to the same plot file. Watch will make a new plot window (up to a total of 9) for each plot file name that is generated. This allows many plots to observed simultaneously.

```
plot/udata min=-.05 max=.0
```

This line plots the summary data defined in the macro with `udata/set`. Minimum and maximum values are defined to show the small losses more effectively.

```
title diffraction mode shape
```

This line defines the title for the next plot.

```
set/density 32                  # set plot grid to 32 x 32
set/window/abs -.05 .05 -.05 .05
```

The first line sets the density of the plotting grid to be 32 x 32. The second line sets the width of the plot to be 0.05 x 0.05 cm. This provides a plot window just big enough to show the main part of the beam.

```
plot/watch plot2.plt            # set plot name
```

This line defines a new plot file for the subsequent plot. Watch will make a new plot window for the new plot.

```
plot/iso 1
```

This line makes an isometric plot to show show the shape of the resonator mode after the 100 passes. Because we started from random noise, the resonator is not completely converged after 100 passes and shows several low order Hermite-gaussian modes. If we run more passes, the calculation will ultimately stabilize to the expected gaussian mode.

Running the command file

We could, of course, enter each command line interactively, but this would be quite tedious. It is easier to enter the lines in to a file. Let us assume the name of the file is RESONATOR.INP. We would then enter the single command line:

```
read/disk resonator.inp
```

and all calculations will take place to generate the plots shown below.

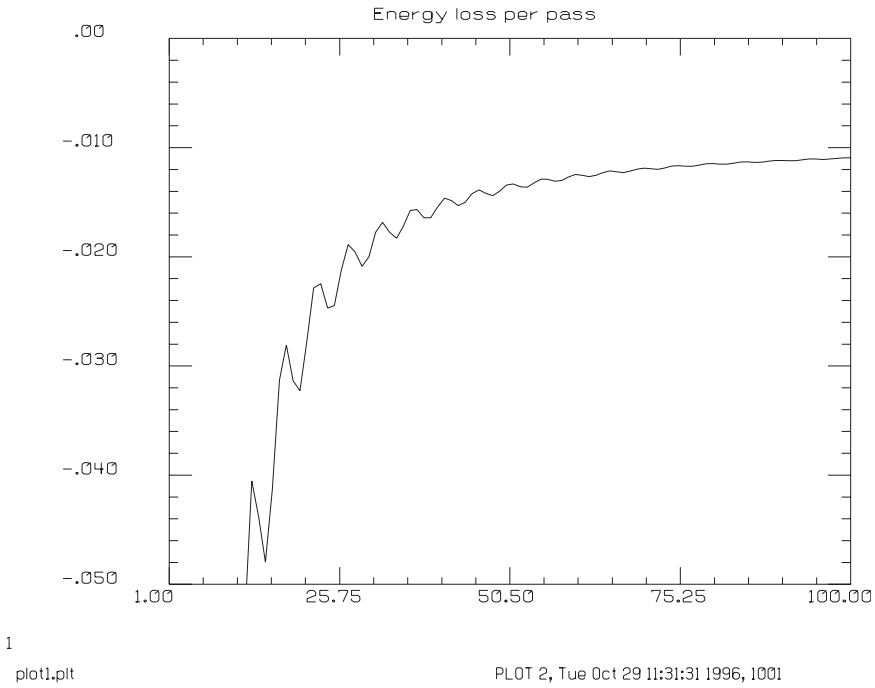


Fig. 4.2. Summary of energy loss per pass showing gradual convergence to steady-state condition from a random noise start.

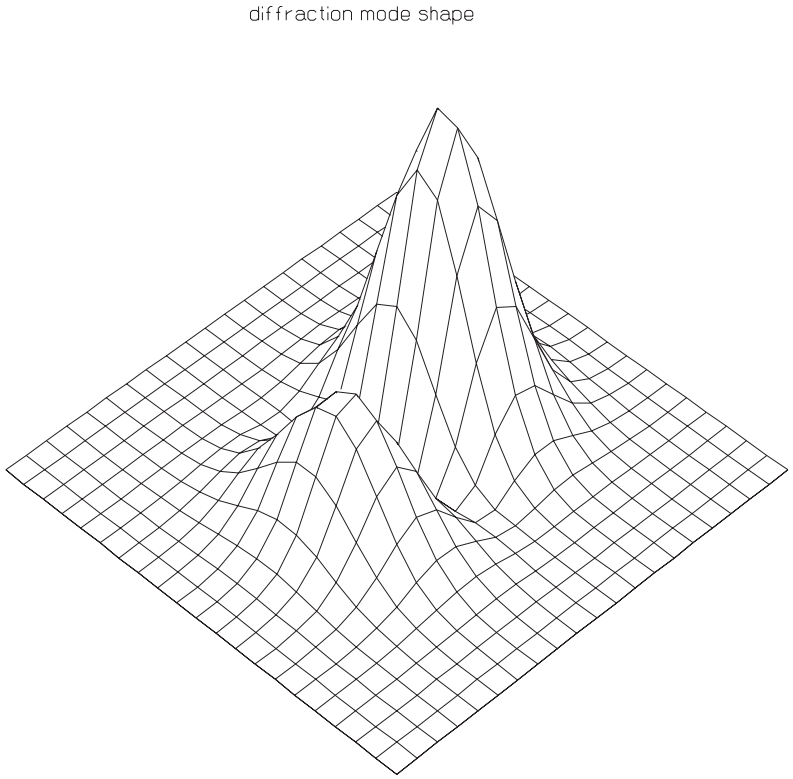


Fig. 4.3. The cavity mode after 100 passes from a random noise start. We still see a mixture of modes — primarily TEM(0,0) and TEM(1,0) .

Examples of GLAD

In this chapter we present a few examples. See the online manual: GLAD Examples Manual, for the full suite of examples.

SAMPLES

General apertures and obscurations 60
Phase aberrations 64
Misaligned unstable resonator 68
Stable resonator 73
Effect of spatial filter on polarization 76
Transient response of YAG laser 80
Through-focus aberration 84
Binary optics 86
Waveguide with s-bend 91

General apertures and obscurations

Example of unusual aperture and some of the plotting capabilities. In addition to the standard aperture shapes (circular, elliptical, square, and rectangular) GLAD can make apertures of arbitrary shape with `clap/gen` and `obs/gen`. This example illustrates a sample aperture consisting of the letters AOR.

A large array size of 512 x 512 is chosen to provide high sampling. The effects of diffraction are illustrated by propagating the beam after the aperture. Since there are so many points, it is difficult to resolve all of the data if the whole beam is displayed.

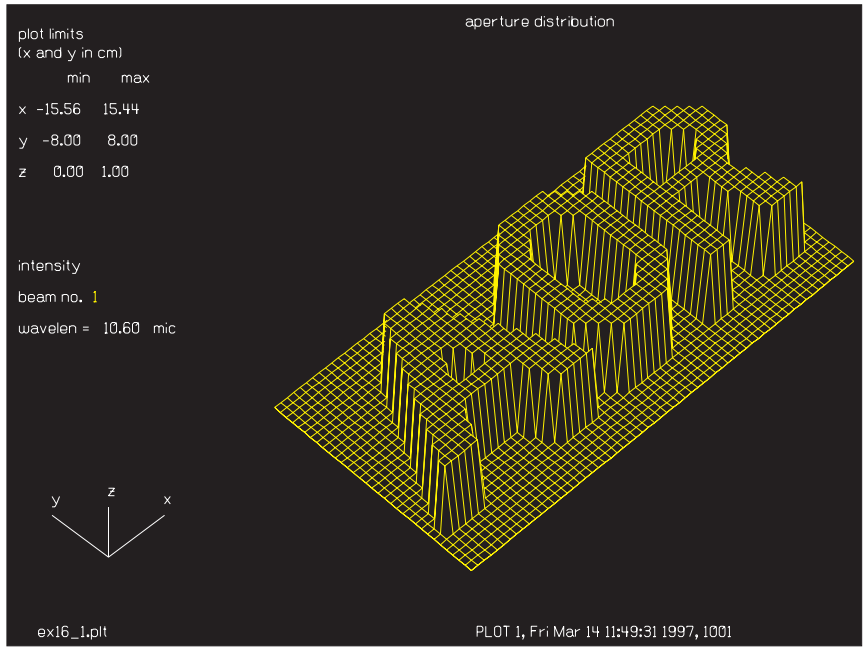


Fig. 5.1. Representation of the letters AOR with apertures and obscurations.

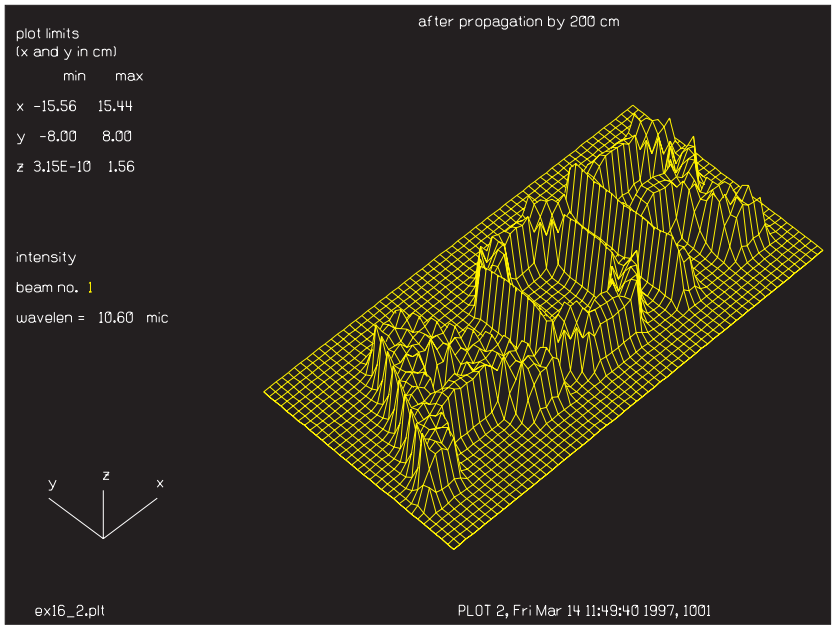


Fig. 5.2. Aperture function after propagation by 200 cm. The detail in the array is far greater than may be readily displayed when viewing the full array.

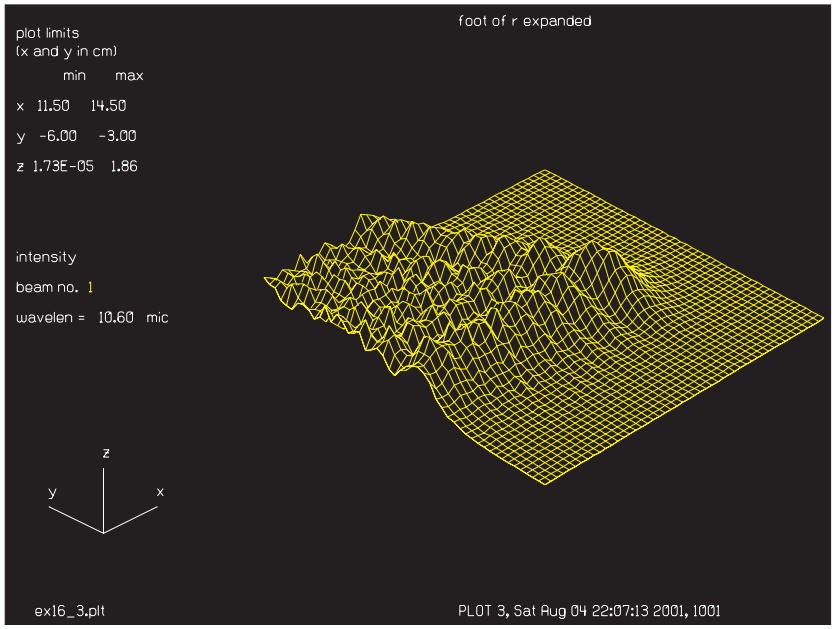


Fig. 5.3. A detailed view of the foot of the R is shown to illustrate the level of detail in the 512 x 512 array.

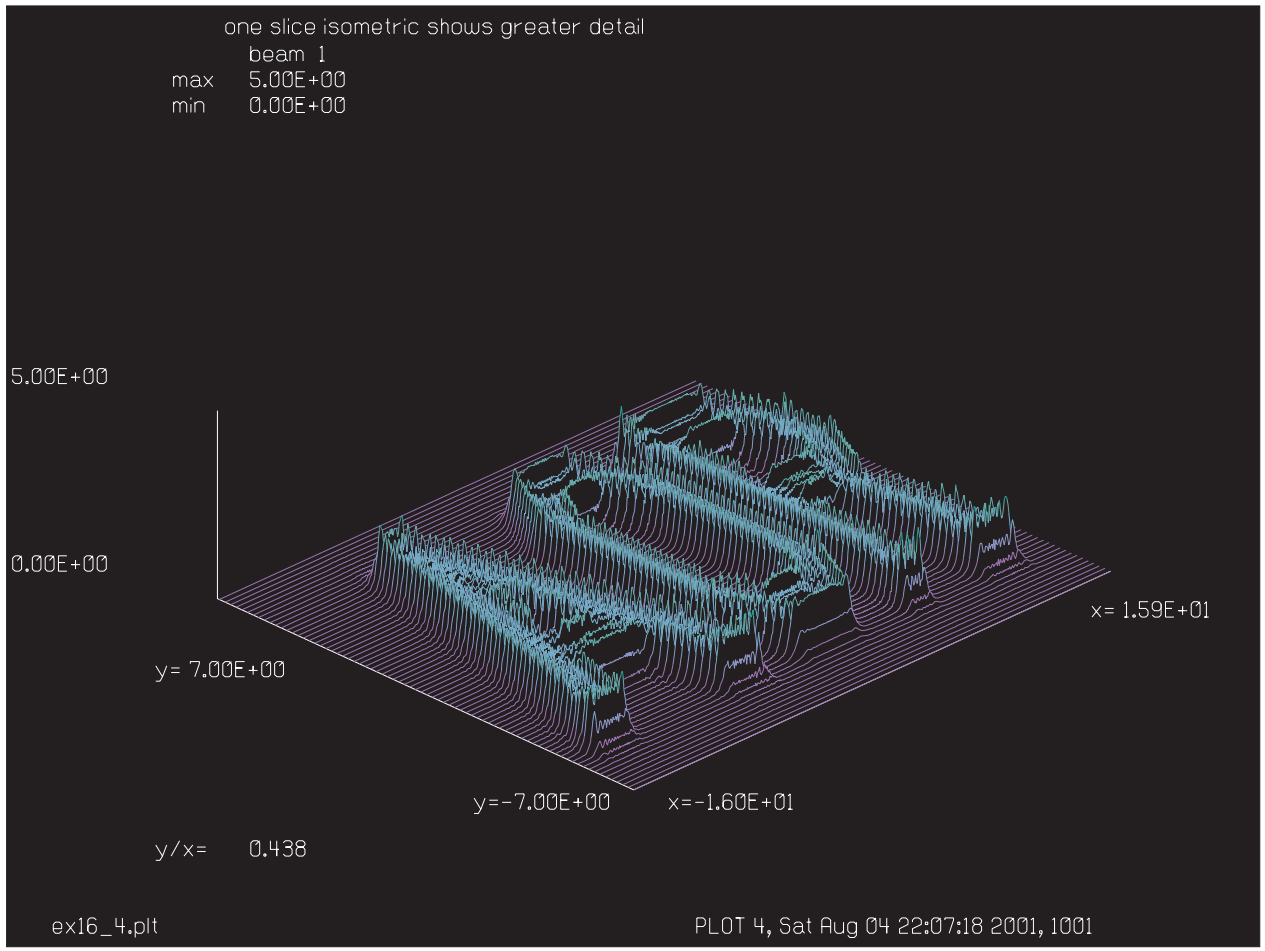


Fig. 5.4. The `plot/liso` command shows greater detail in the horizontal slices than `plot/iso`.

The `clap/gen` and `obs/gen` commands allow general clear apertures and obscurations to be defined in terms of a series of polynomials, as shown below.

```
clap/gen 1     # outline all letters with general aperture
4
-15 -5
14 -5
14 5
-15 5
obs/gen 1      # cut out left side of A
3
-15 -5
-10 5
-15 5
```

Phase aberrations

GLAD can model almost any type of aberration. The opposite page shows examples of a combination of spherical and astigmatism, a mixture of higher order Zernike aberrations (as isometric and contour plots), a phase grating, and random aberration with large autocorrelation and narrow autocorrelation widths (see below).

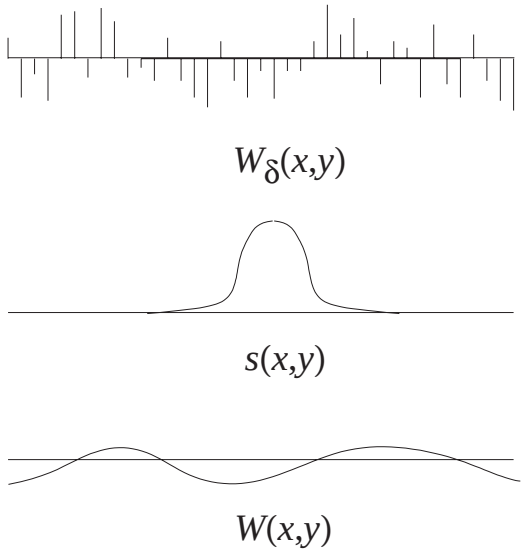


Fig. 5.5. The random wavefront is constructed from a delta-correlated wavefront generated from a sequence of random numbers. The delta-correlated wavefront is smoothed to create a wavefront with the desired statistics.

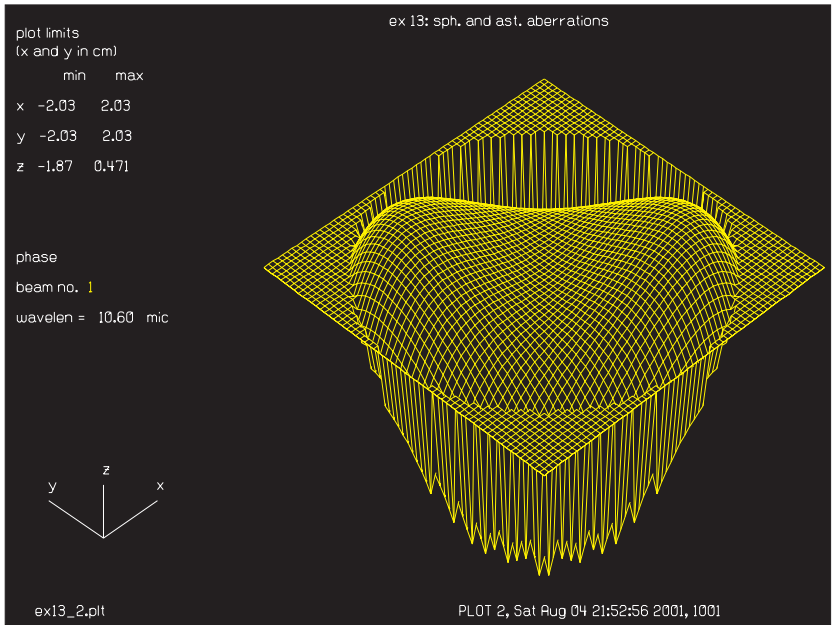


Fig. 5.6. Spherical aberration and astigmatism.

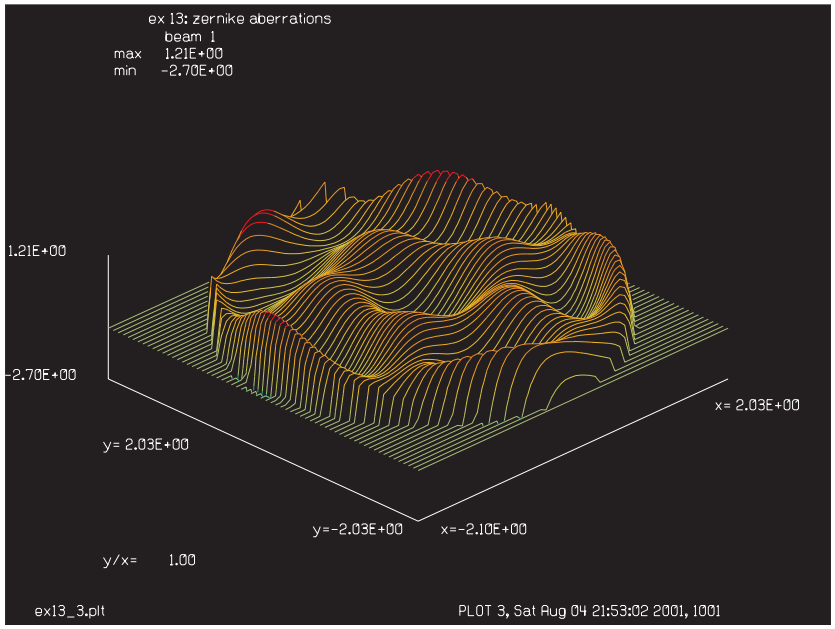


Fig. 5.7. Higher order Zernike aberrations.

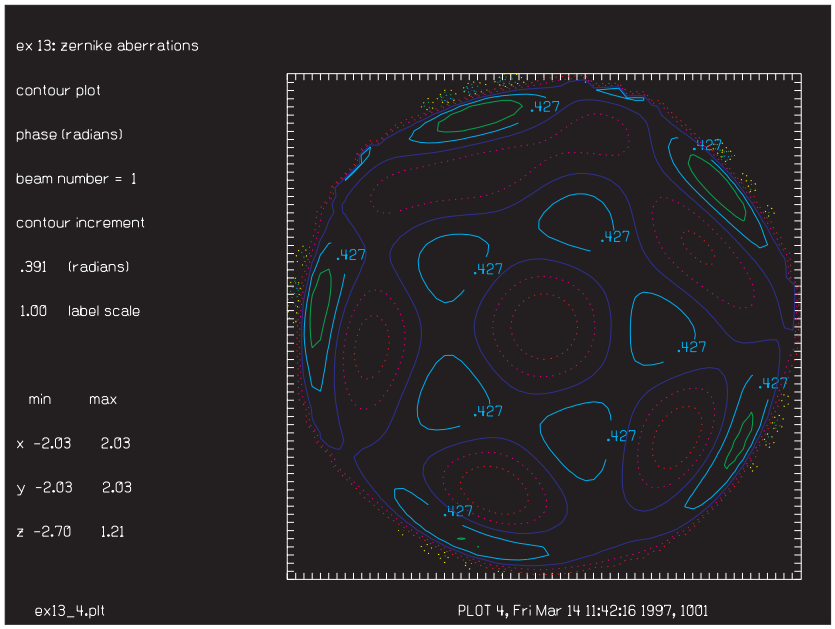


Fig. 5.8. Zernike aberrations, contour phase plot.

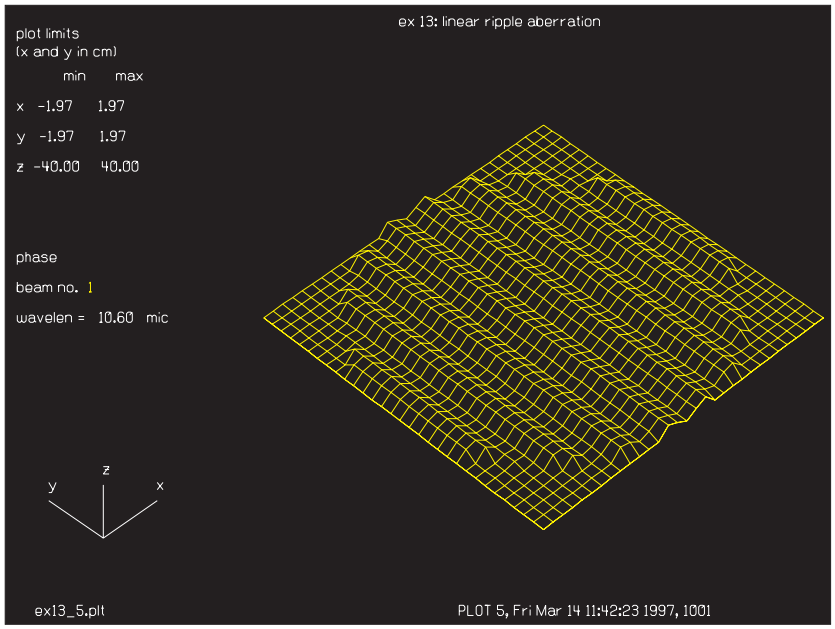


Fig. 5.9. Linear phase ripple for phase grating.

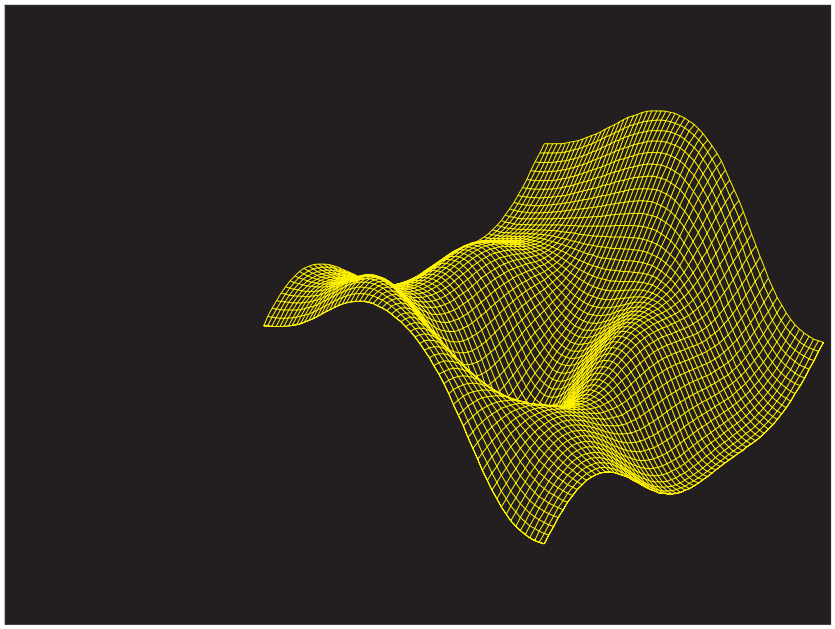


Fig. 5.10. Random aberration formed with a relatively wide autocorrelation width.

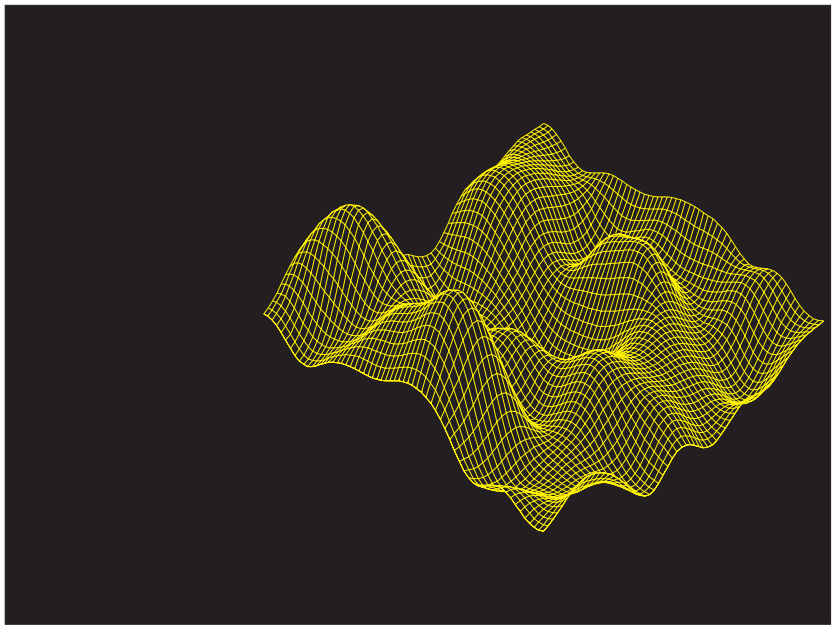


Fig. 5.11. Random aberration formed with a narrow autocorrelation width. The aberration was formed from the same set of random numbers as Fig. 5.10, but was given less smoothing.

Misaligned unstable resonator

Example 11a models a confocal unstable resonator with circular mirrors. The resonator collimated and equivalent Fresnel numbers are

$$N_c = \frac{Ma^2}{L\lambda}, \quad N_{eq} = \frac{M^2 - 1}{2M}\frac{a^2}{L\lambda}$$

where a is the aperture radius, L is the resonator length, λ is the wavelength, and M is the magnification.

The parameters that are used are

$L = 90$ cm, $a = .3$ cm, $M = 2$,

$\lambda = 10\ \mu$

This results in $N_c = 2$ and $N_{eq} = 0.75$.

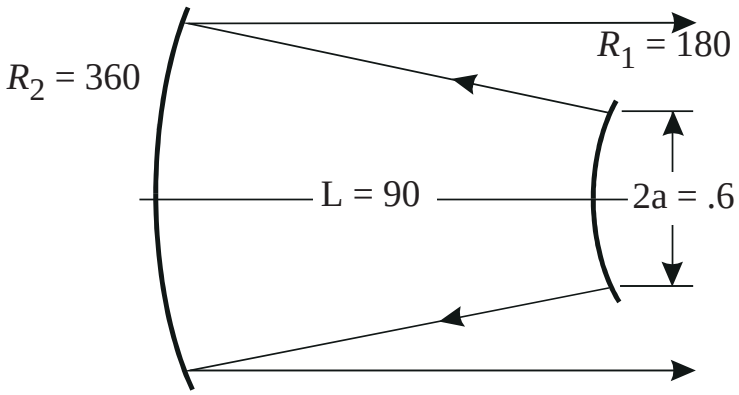


Figure 5.12. Unstable confocal resonator configuration with tilt.

After one round trip the units of the distribution are twice those of the starting distribution. To start the next round trip the distribution must be rescaled to the original units. The loss per cycle is 44%.

This example is misaligned by 0.1 wavelength. The resonator takes significantly longer to converge and the loss increases to about 55%.

The cavity mode starts from spontaneous emission. Fig. 5.13 shows a very noisy distribution after one pass, but Fig. 5.15 shows significant smoothing by the second pass. Burn patterns of the cavity mode, as shown in Figs. 5.14 and 5.16. The device is well converged after about 30 passes.

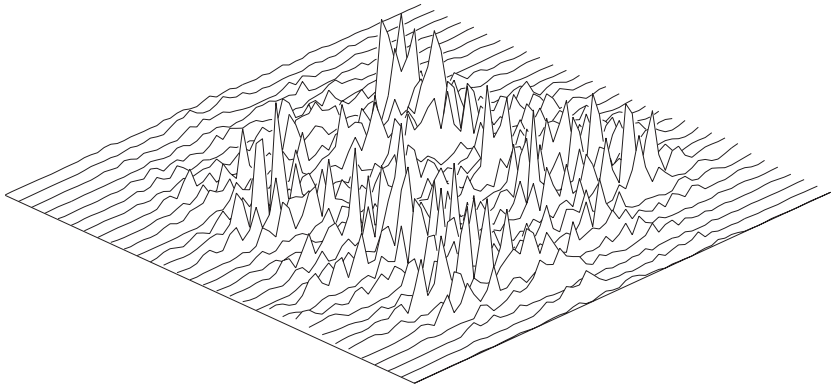


Fig. 5.13. Cavity mode at one pass after starting from spontaneous emission.

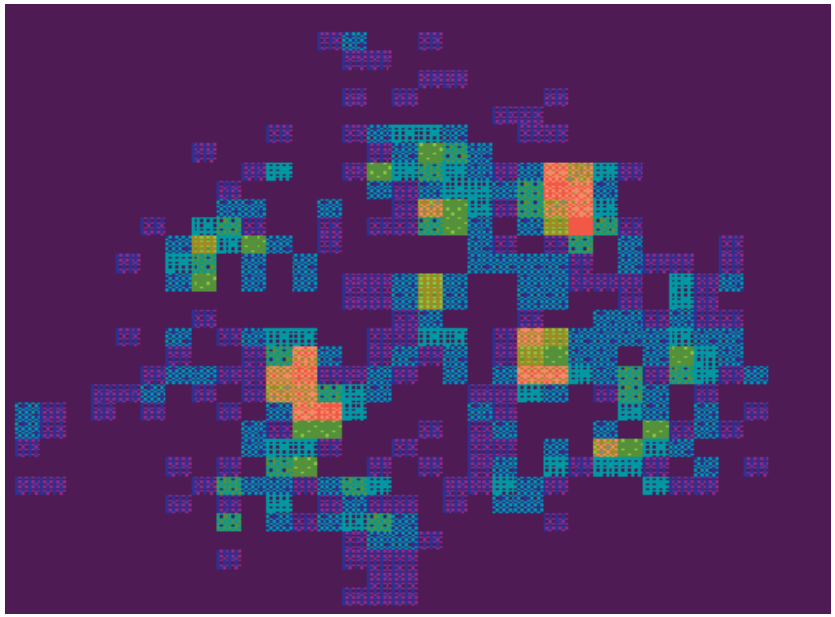


Fig. 5.14. Burn pattern of cavity mode after one pass.

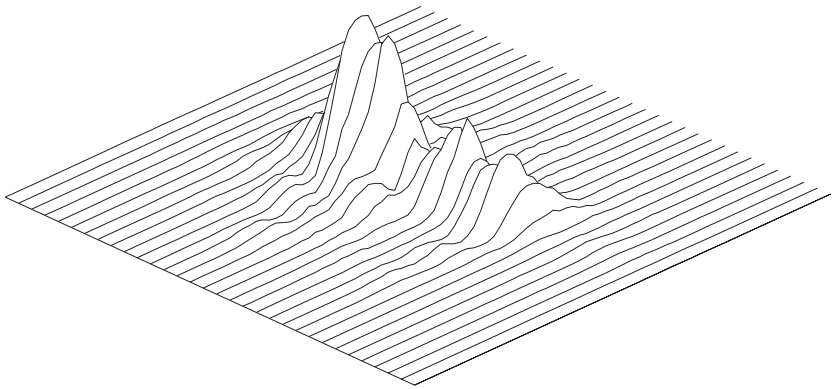


Fig. 5.15. Cavity mode at two passes after starting from spontaneous emission.

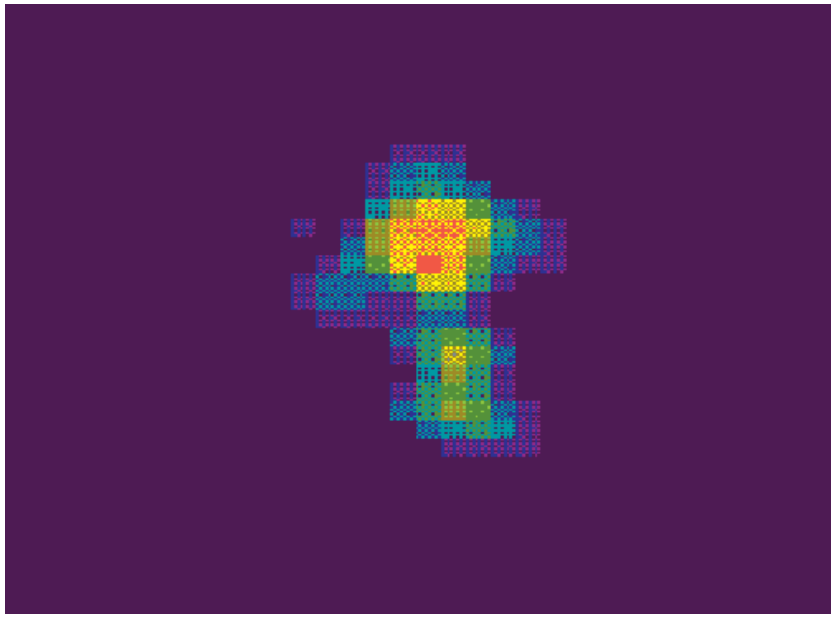


Fig. 5.16. Burn pattern of cavity mode after one pass.

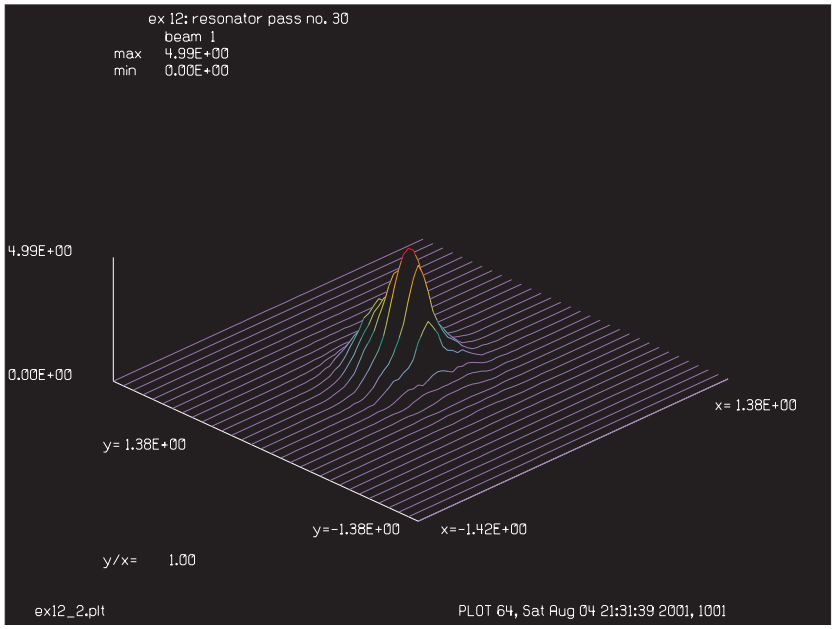


Fig. 5.17. Converged transverse mode before scraper mirror.

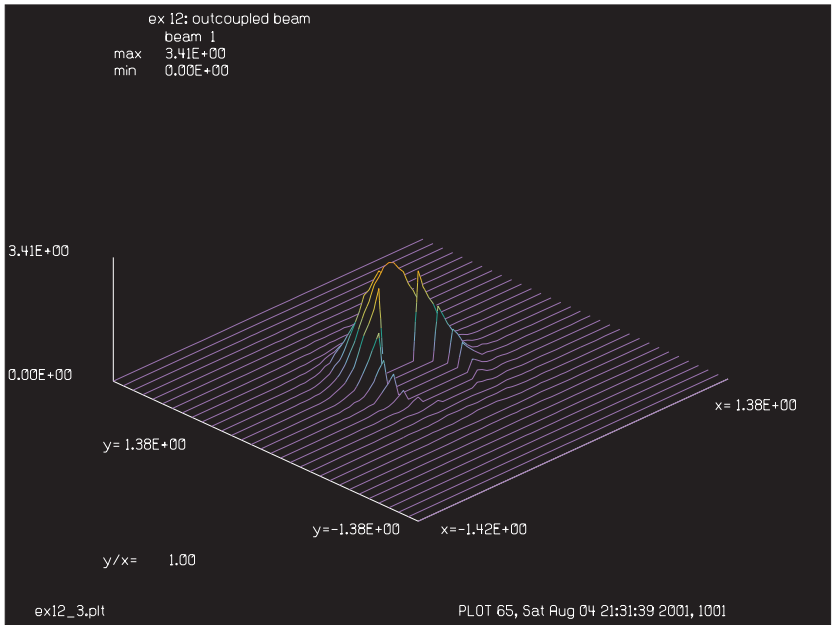


Fig. 5.18. Converged transverse mode after scraper mirror.

Stable resonator

This example is a stable resonator with spherical mirrors. The configuration consists of a flat mirror and a concave spherical mirror of radius 50 cm. The mirrors are separated by 45 cm. The parameters are summarized below.

Beam 1 is initialized as the ideal eigenmode. Beam 2 is reinitialized to be a flat-top function to observe convergence from a non-optimum starting point. Beam 1 converges almost immediately to .09 percent loss per pass for the aperture radius of .14 cm. Beam 2 takes about 90 passes to converge to within .1 percent.

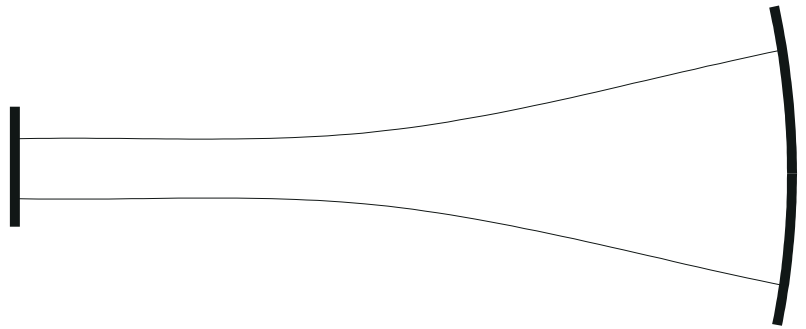


Fig. 5.19. Stable resonator configuration. The waist will form on the flat mirror and the phase radius will match the radius of the concave mirror in the ideal geometric mode.

Table 5.1. Parameters	
length	45 cm
mirror radius	50 cm
wavelength	1.064
Rayleigh range	15 cm
waist radius	.02253936 cm
aperture radius of spherical mirror	.14 cm

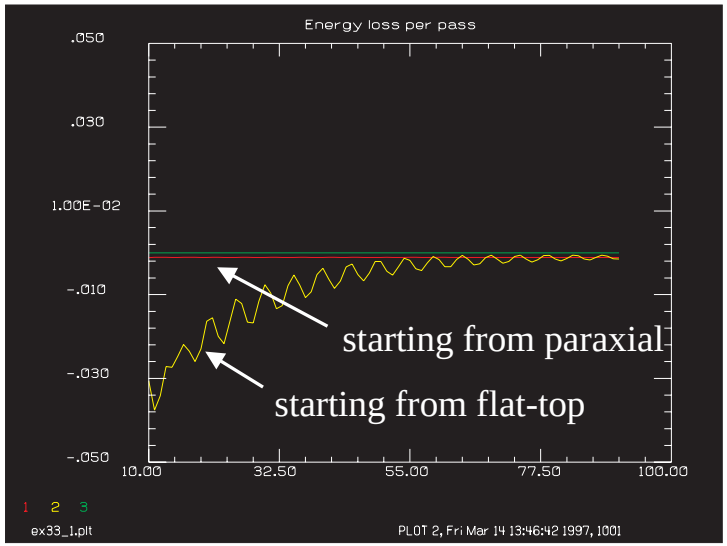


Fig. 5.20. Plot of energy loss per pass as a function of the pass number. The plot is from Pass 10 to Pass 91. The lower horizontal line is actually the the loss for Beam 1 which has already converged from the geometric mode with in 10 passes. The oscillating curve shows the convergence of Beam 2

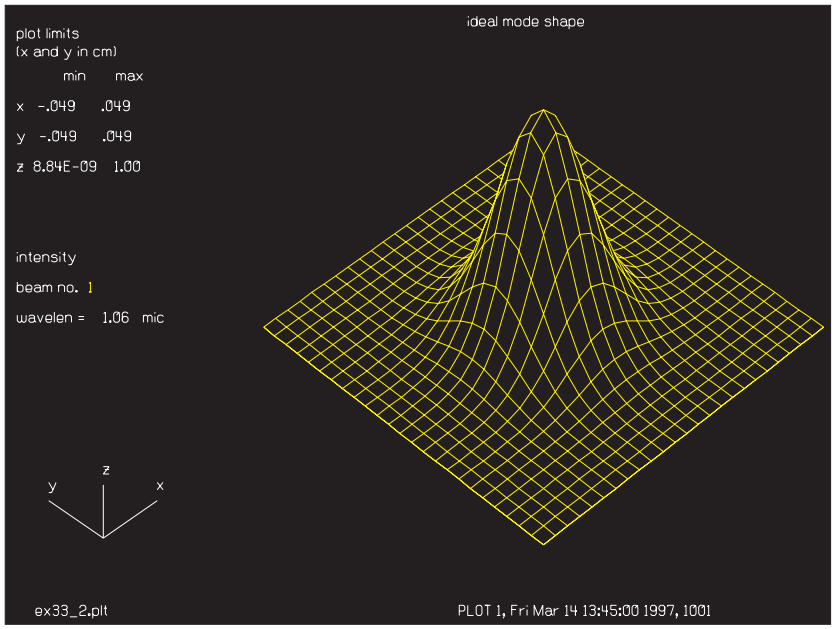


Fig. 5.21. Ideal resonator mode.

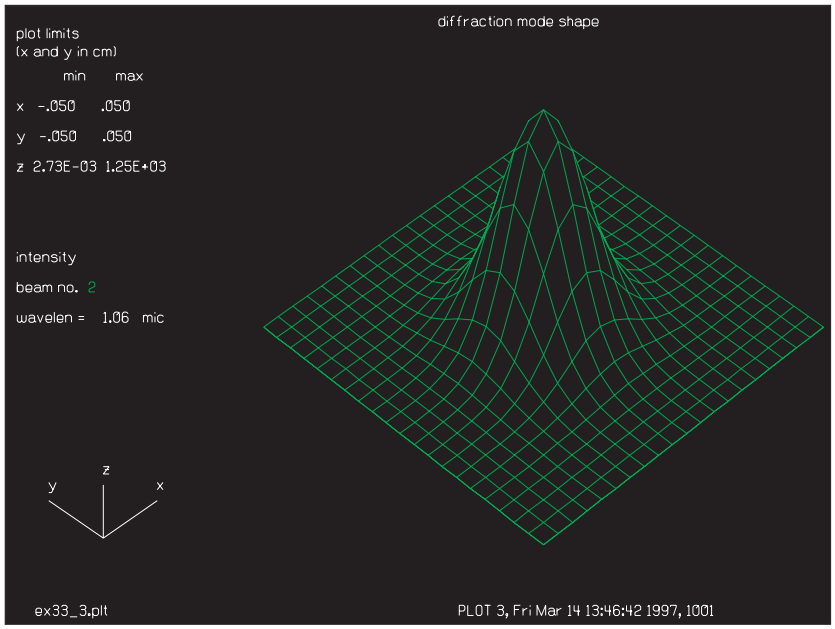


Fig. 5.22. Converged diffractive mode.

Effect of spatial filter on polarization

This example illustrates the effect of a spatial filter on polarization variation in the pupil. An arbitrary distribution of polarization is established in the pupil. The polarization is linear along the x- and y-axes and becomes circular along the 45 degree diagonals. The polarization variation reduces the Strehl ratio to about 0.5. The spatial filter smoothes out pupil variations including polarization. Generalized polarization filters can be used. The beam is separated into parts which are parallel and perpendicular to the input distribution with the command `jones/orthog`.

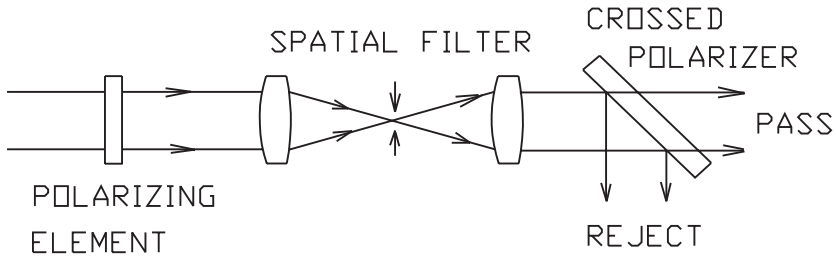


Fig. 5.23. Configuration for the example. A polarizing element is inserted in the beam. The spatial filter smoothes out the polarization in the pupil. A polarizing filter which exactly matches the input is used to select the part of the output common to the input and the part orthogonal to the input.

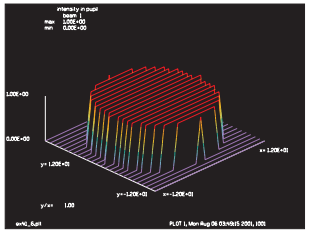


Fig. 5.24. Initial irradiance distribution.

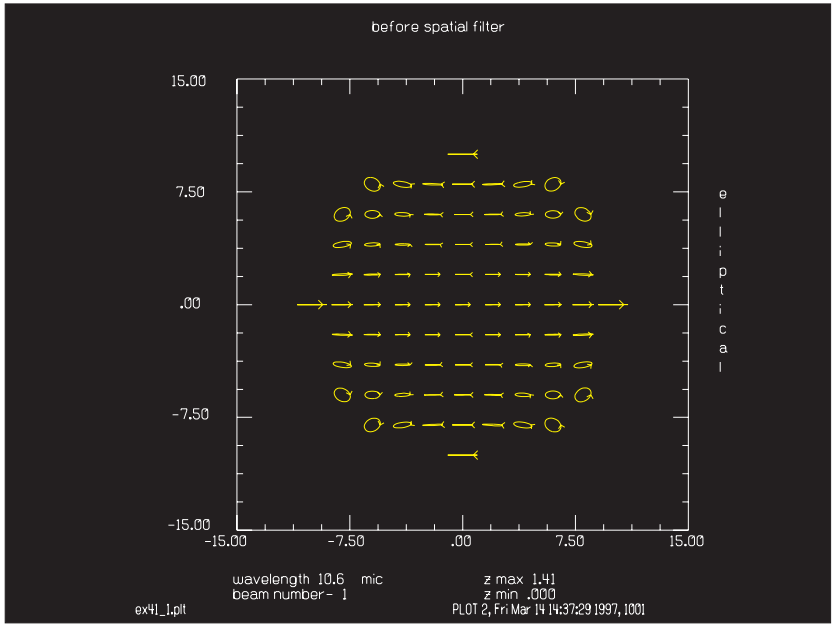


Fig. 5.25. Initial pupil polarization.

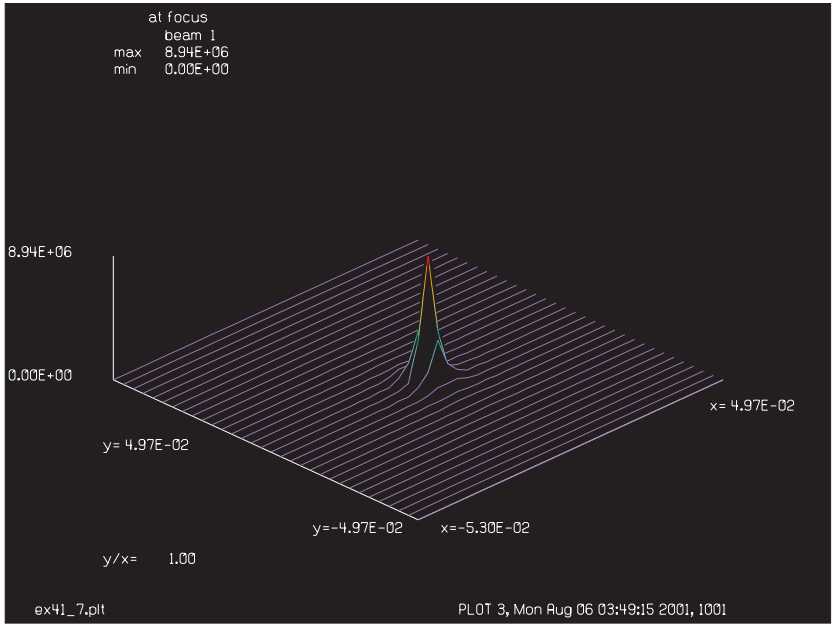


Fig. 5.26. Image irradiance.

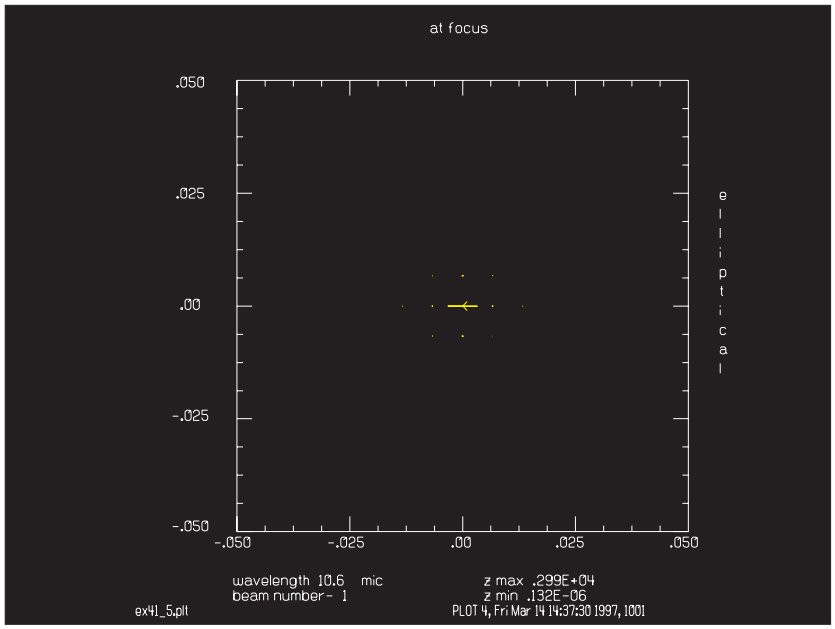


Fig. 5.27. Polarization in the far-field.

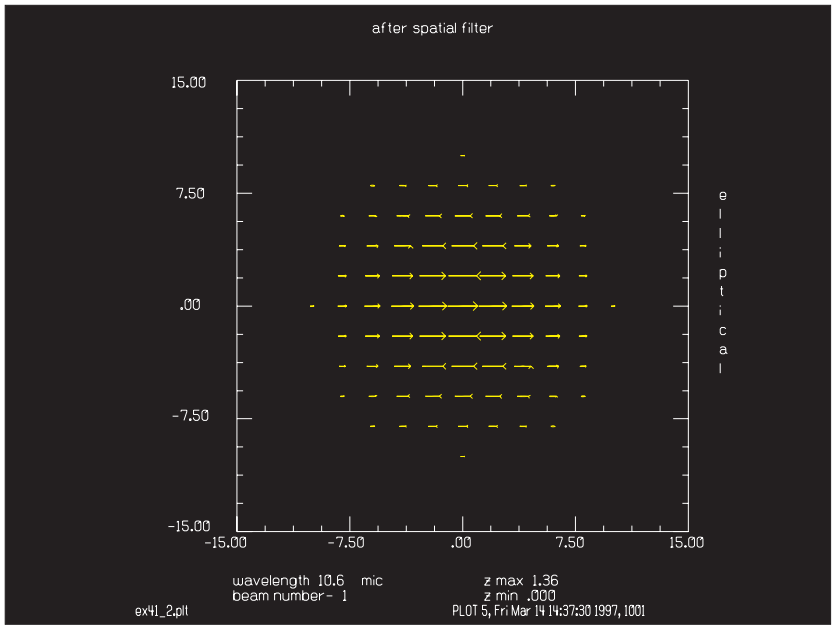


Fig. 5.28. Polarization after spatial filter.

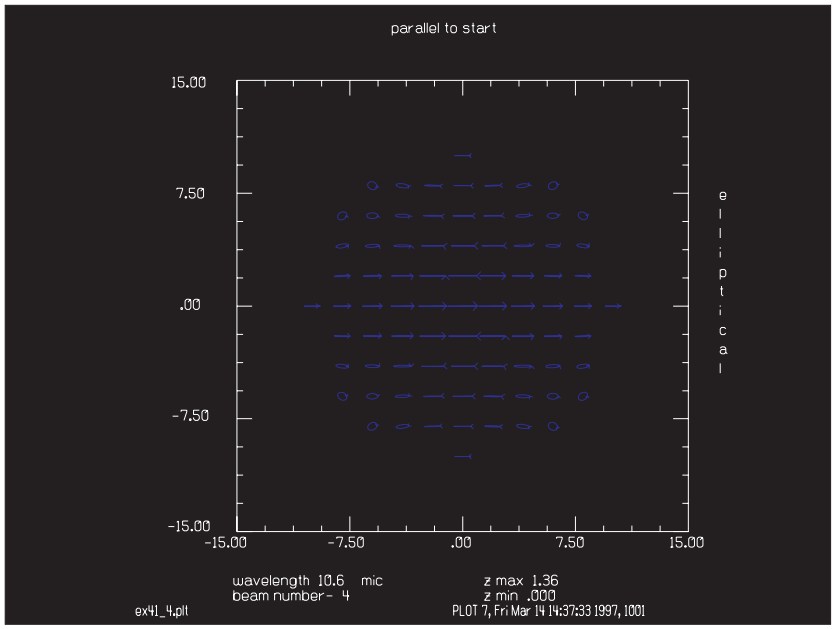


Fig. 5.29. Part of output parallel to input polarization.

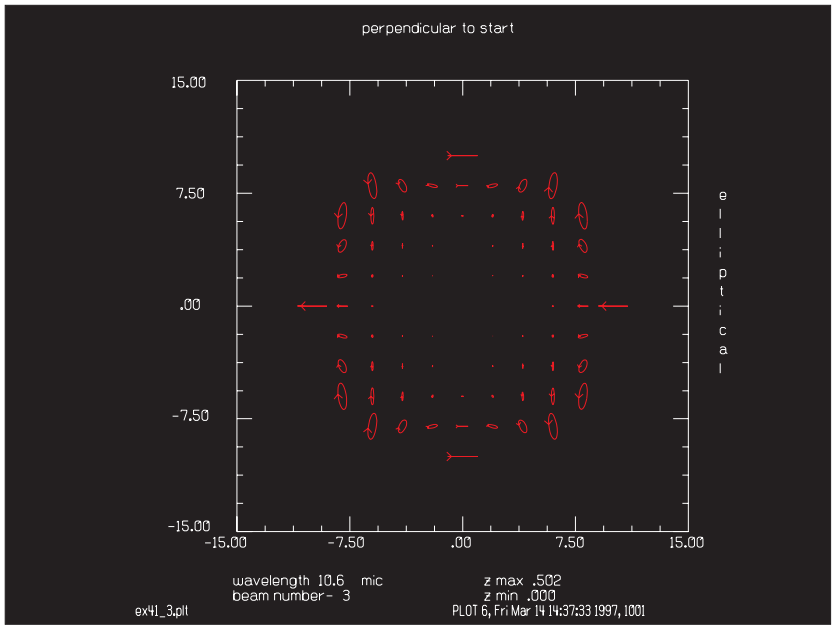


Fig. 5.30. Part of output orthogonal to input polarization.

Transient response of YAG laser

The objective of this example is to investigate and model the a Q-switch laser to determine the time-dependent output power, beam divergence, and the intensity profile. The beam quality is initially very poor because the device starts from noise but improves with each pass as high angle light is scattered out of the resonator.

The laser is a Q-switched YAG laser with about a 60 cm round-trip path and approximately a 10 ns pulse. The device has the following components

1) electro-optic Q-switch

2) 6 mm diameter YAG rod

3) polarizing output coupler

4) crossed roof mirrors as end reflectors

5) an off-axis cube corner reflector to fold the system

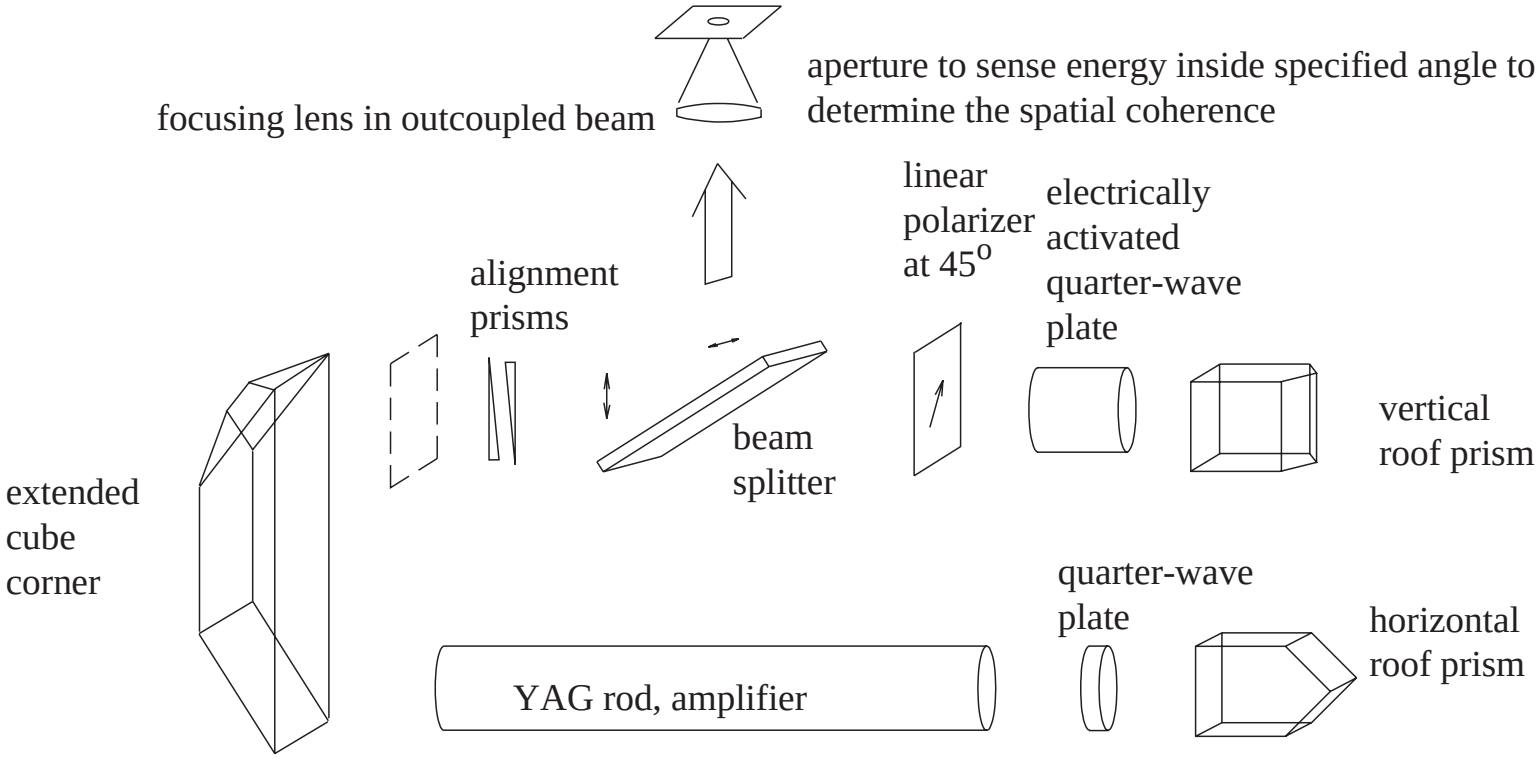


Fig. 5.31. Schematic of the Q-switch laser. The resonator is defined by the two crossed roof prism end reflectors, which assure that the optical axis is effectively perfectly aligned. The system is folded by an extended corner cube. A “stronglink” shutter consisting of a mirror rhomb provides a positive “off” condition. Alignment prisms are provided for rough centering of the beam with respect to the limiting apertures. A beam divider outcouples a percentage of the s-polarization but passes all of the p-polarization. The Q-switch consists of a 45^{o} linear polarizer and an electrically activated quarter-wave plate. When activated the combination of quarter-wave plate, in double pass, and roof mirror transmits the polarization component. When deactivated the combination blocks all light. The outcoupler is assumed to be rotated 45^{o} to line up with the linear polarizer.

6) alignment wedges

7) a periscope-type, stronglink “on-off” switch

8) a quarter-wave plate

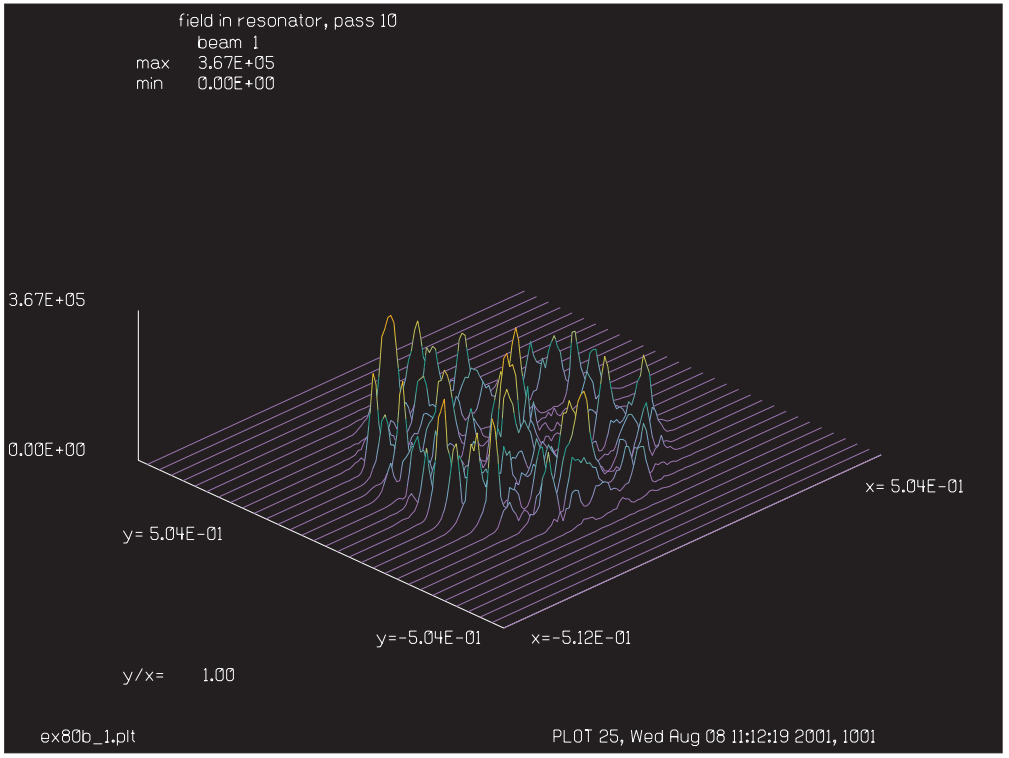


Fig. 5.32. Outcoupled beam distribution at pass 10 of ex80b.inp corresponding to 20 nanoseconds after the Q-switch. This beam is approximately 20 times diffraction limited.

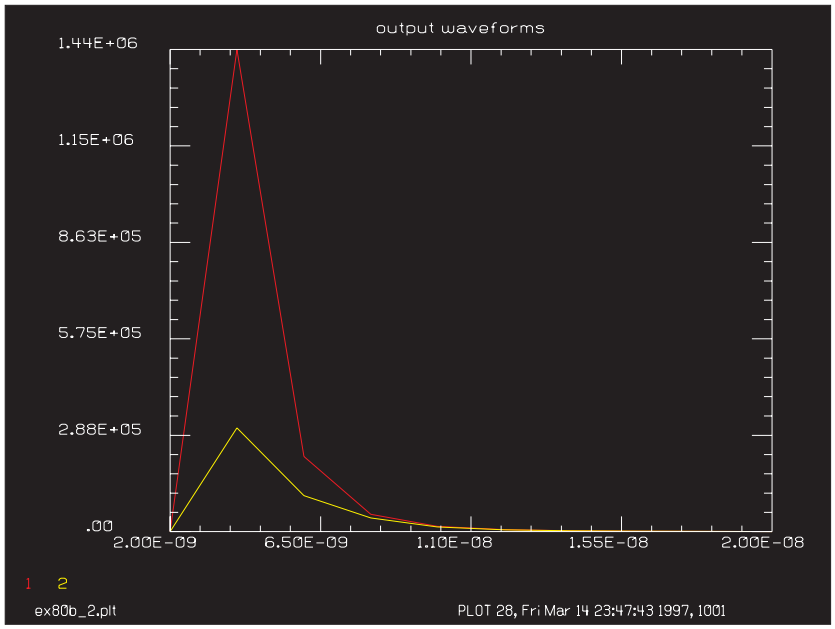


Fig. 5.33. Output power of Q-switch versus time and the output power falling within 5 milliradians.

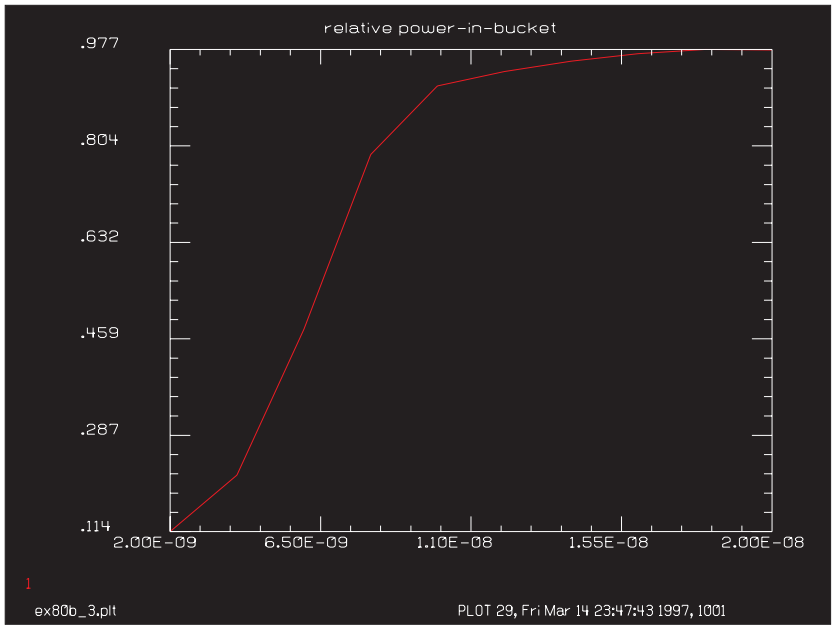


Fig. 5.34. Relative output power falling within 5 milliradians, showing the increase in spatial correlation.

Through-focus aberration

This example illustrates the nature of the speckle pattern in the far-field. A smoothed random wavefront is focused to create the speckle pattern. A series of scans across the image are taken at different axial points. A total of 64 axial scans are taken. Each axial scan is displayed in Fig. 5.37, which displays the contorted, snake-like speckle pattern. The scale in the transverse direction has been expanded to more clearly show the speckle structure.

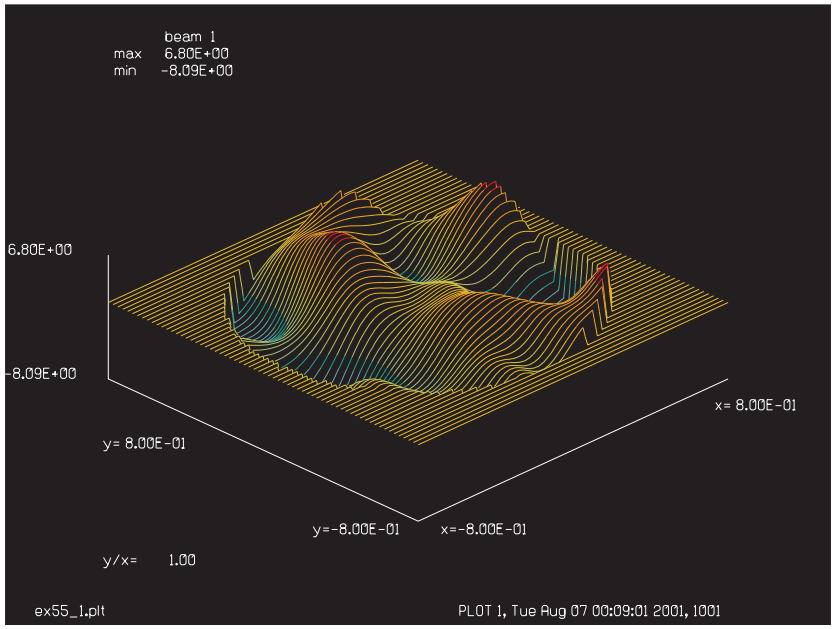


Fig. 5.35. Pupil aberration exhibiting smoothed random aberration of moderately high spatial frequency.

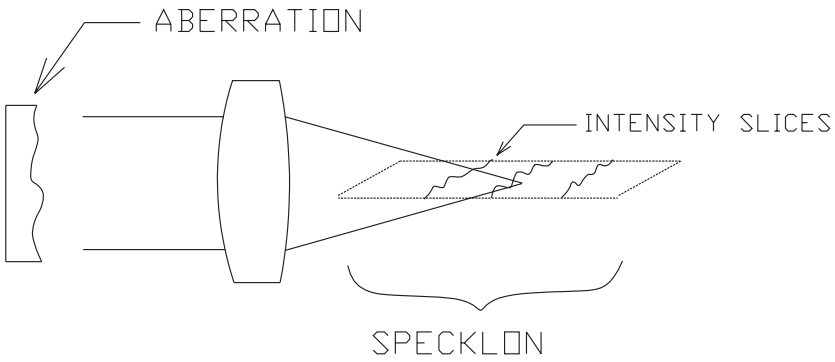


Fig. 5.36. A series of scans across the image are taken at 64 axial positions.

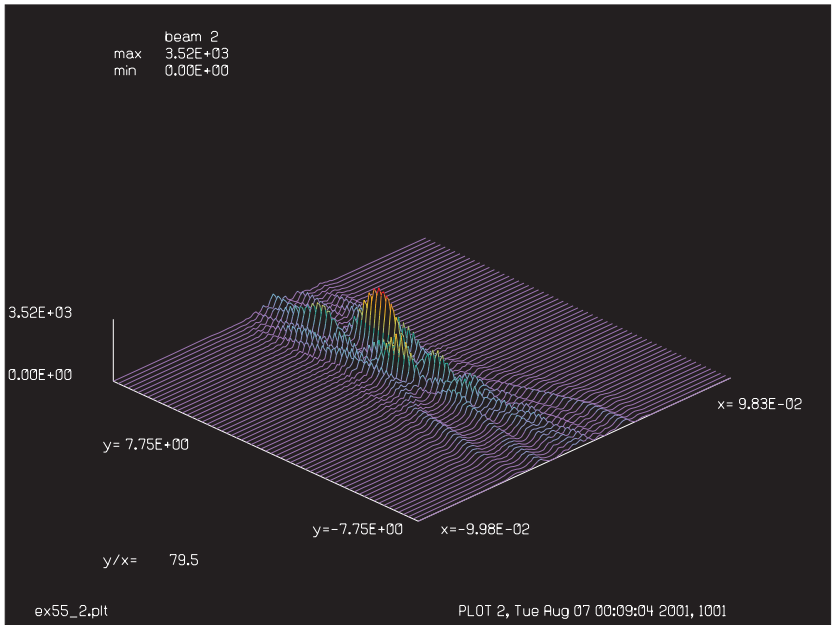


Fig. 5.37. The speckle pattern shows twisted structure in a snake-like pattern. The scale in the transverse dimension is expanded to more clearly illustrate the structure. The light is traveling from left to right. The plot shows a series of profiles across the center of the beam at successive axial

Binary optics

Binary optics consist of discrete layers of identical thickness grouped into major levels of thickness $\lambda/n-1$ and sublevels. Figure 5.38 shows a continuous surface after being divided into major levels and sublevels with 8 sublevels per major level. Since the major levels contribute exact multiples of 2π to the phase at the nominal wavelength, these layers may be subtracted from the surface. The lower figure of Fig. 5.19 shows the surface with major levels subtracted. In binary optics, the number sublevels per major level is generally set to be a power of 2, e.g., 2, 4, 8, etc. The maximum height of the binary optic is $\frac{N-1}{N}\frac{\lambda}{n-1}$, where N is the number of sublevels.

For $N = 2$, phase steps of 0 and π are created.

Figures 5.20-5.26 show a grating, positive lens, and a negative lens with two-level and four-level binary form. The higher the number of levels the more efficiently the component operates.

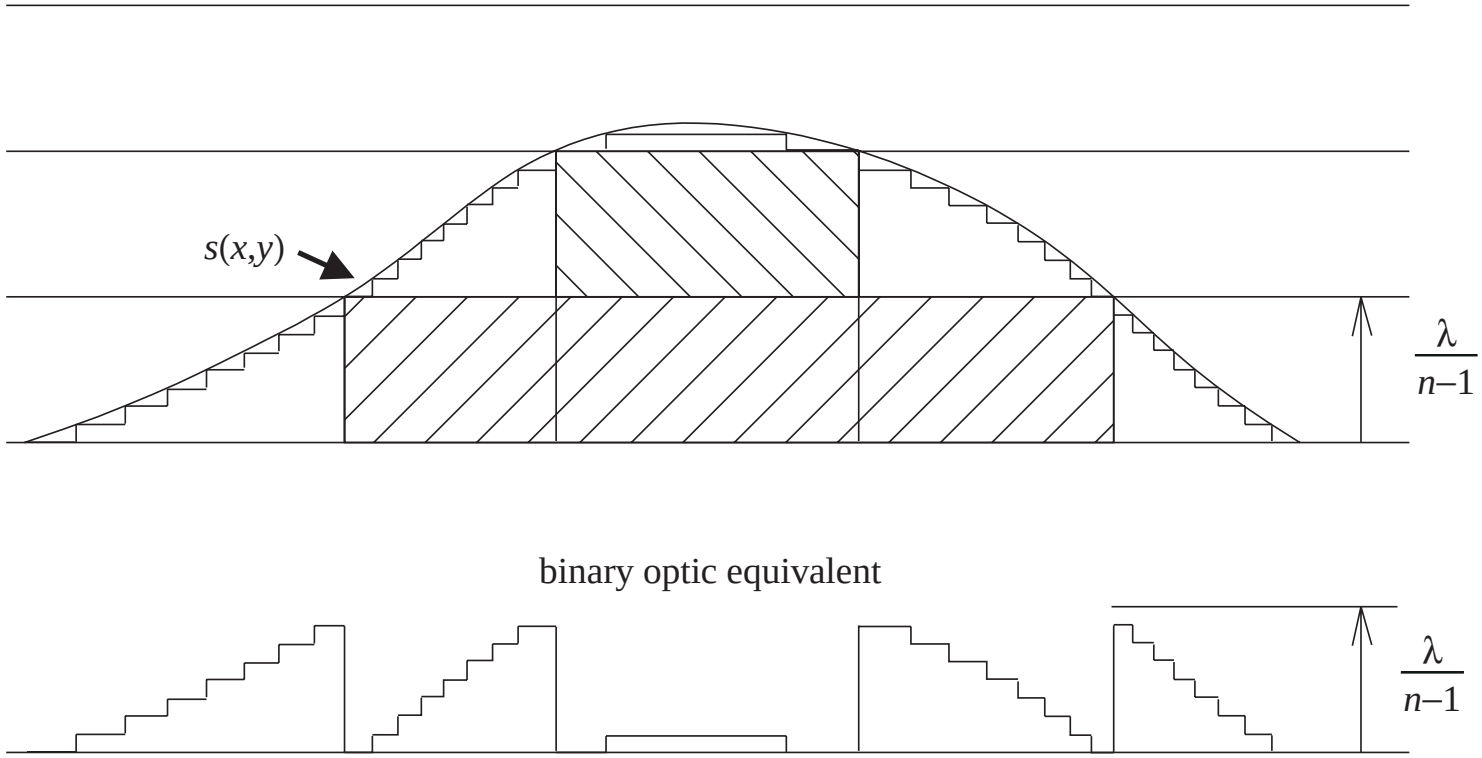


Fig. 5.38. A surface divided into major levels of thickness $\lambda/n-1$ and into 8 sub levels to create a binary optic. The major levels may be subtracted without optical effect at wavelength λ, i.e., by removing the hatched areas. The sublevels produce some diffractive loss into high angle scattering. Note that the maximum height of the binary optic is $\frac{N-1}{N}\frac{\lambda}{n-1}$, where N is the number of sublevels.

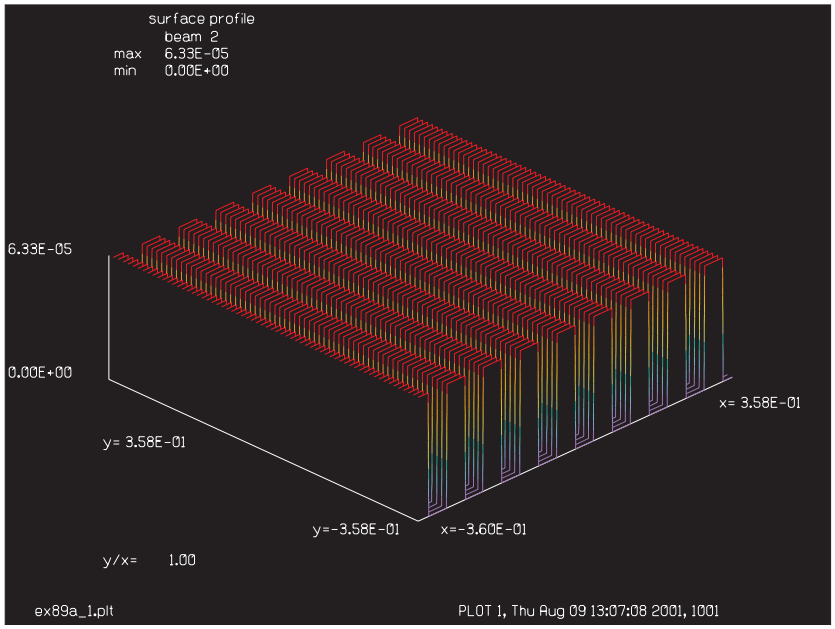


Fig. 5.39. Grating, two-level.

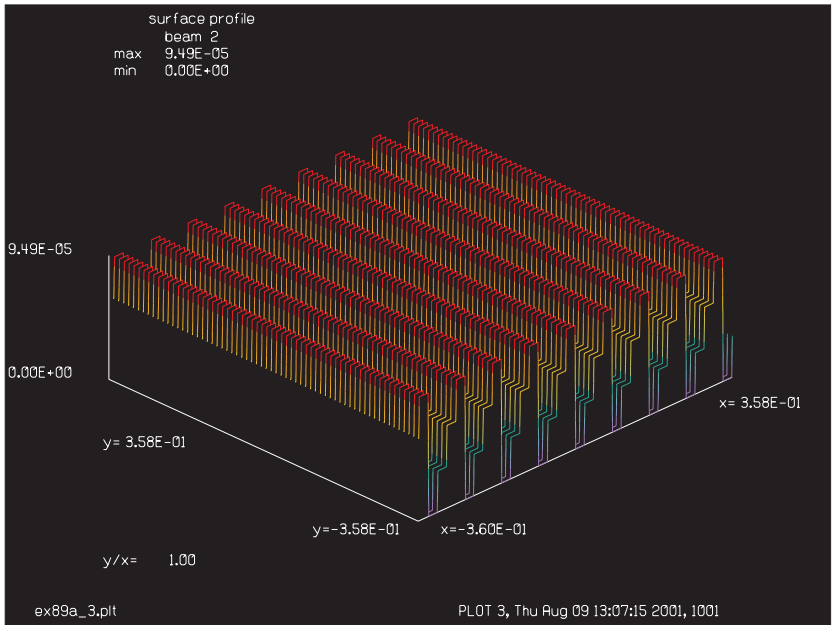


Fig. 5.40. Grating, four-level.

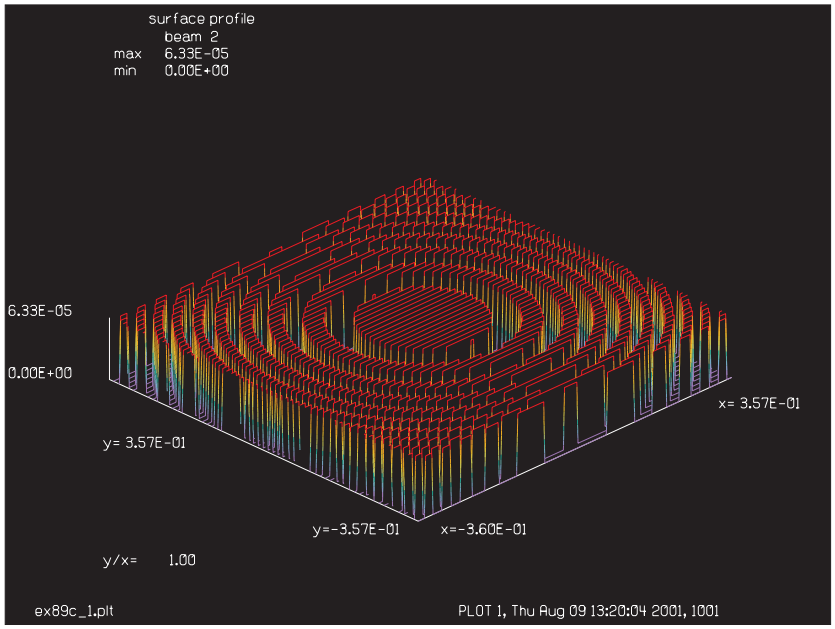


Fig. 5.42. Positive lens, two-level.

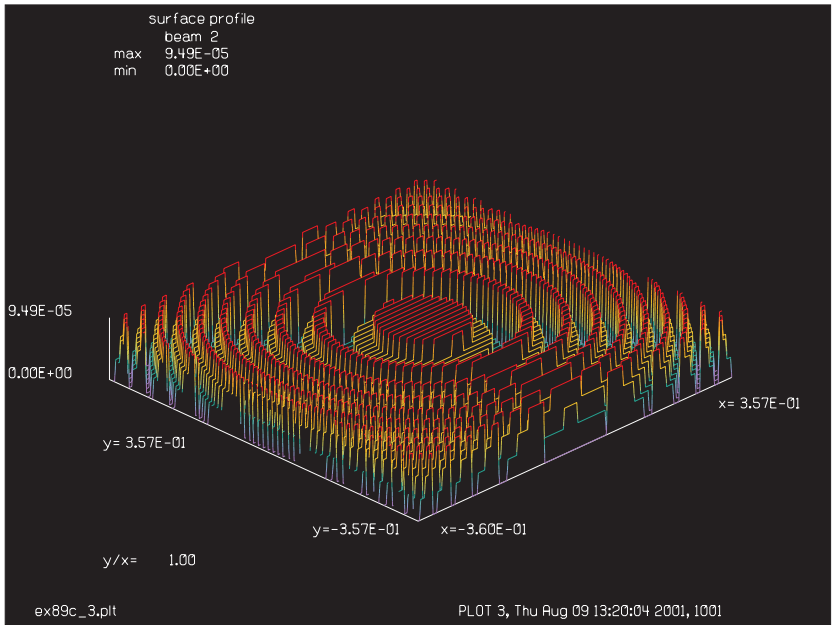


Fig. 5.43. Positive lens, four-level.

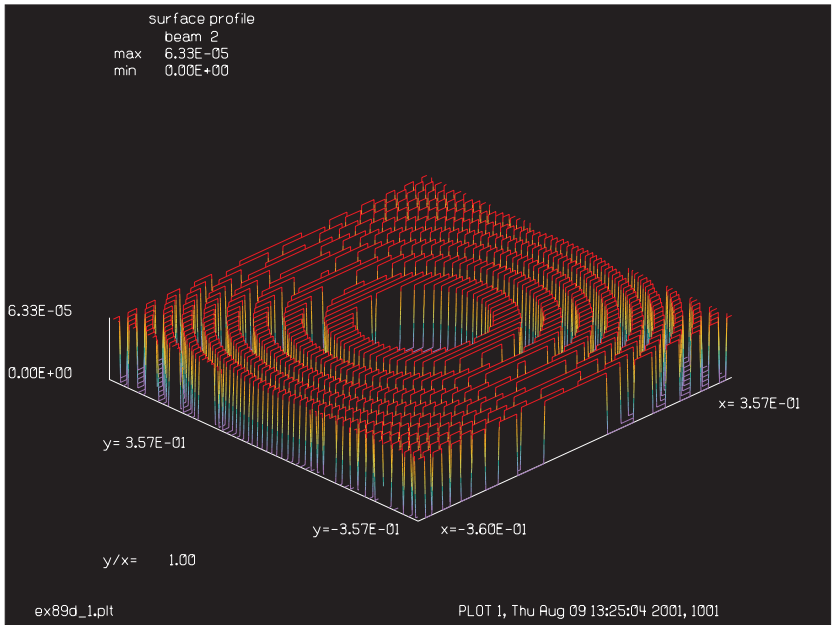


Fig. 5.44. Negative lens, two-level.

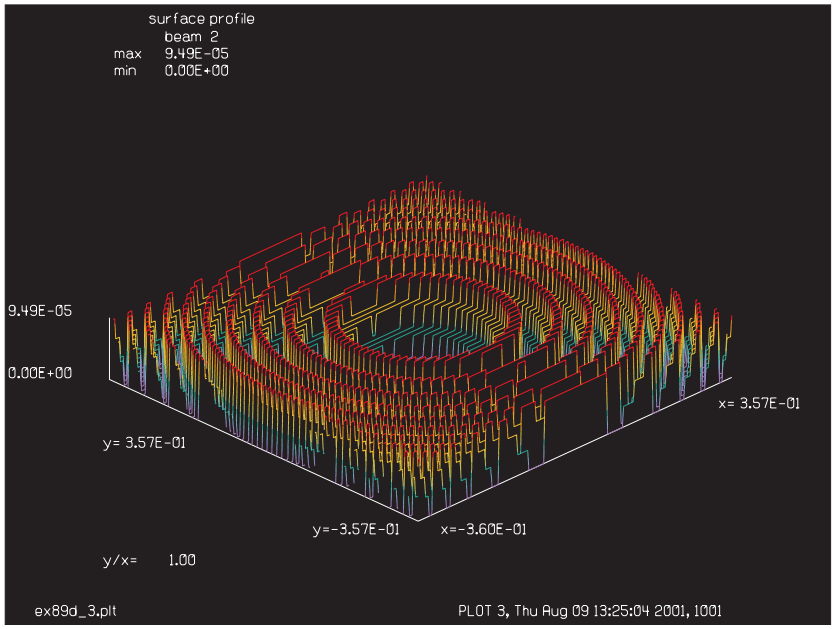


Fig. 5.45. Negative lens, four-level.

Waveguide with s-bend

In GLAD waveguides may be modeled in 3D, the two transverse dimensions (x,y) and the propagation direction z. The treatment is extremely general allowing bends, tapers, junctions, splitters, and multiple high index cores which exchange energy. This example illustrates the some simple cases of circular fiber optics. The optical fiber consists of a core of index 1.532 immersed in a cladding of index 1.5. The higher index core, in these examples, is 1.6 microns in diameter. The wavelength is .6328 microns. Without the high index core, an optical beam would simply diverge with propagation. The high index core acts as a positive lens. In this example, a gaussian beam is started into the fiber. If the gaussian beam has a narrow waist then, initially, the beam will diverge. If the starting gaussian is larger, then the beam converges initially. In the transient regime, the beam tends to go through cycles of blooming and pinching. The cyclical behavior damps out in the about the first 20 microns, for this example, as the higher index modes diffract out of the waveguide and single mode operation takes over.

In numerical modeling, it is necessary to provide some absorption at the edges of the array to emulate the loss of light out of the sides of the cladding. The absorption is represented as a supergaussian distribution with radius of 6 microns. The half-width of the array is set to 8 microns. The radius of the core region of .8 microns keeps the optical mode well confined within the 6 micron absorbing region. However the absorbing region, although it is well outside the optical mode, represents the only source of loss. This treatment is useful for determining the mode shape, but a larger array may be required to accurately determine the loss per unit length of the lowest loss mode.

The first example, Ex86a, is a simple straight fiber. In Ex. 86b the fiber has an s-curve with beginning and ending straight sections. The mode stabilizes in the initial straight section of 20 microns. The mode is disrupted by the next 62 microns of s-bend but restabalizes in the last 20 microns of straight fiber. There is a loss of a few percent in this relatively sharp bend. Note that

the mode follows the fiber through the s-curve but there is blooming and pinching because of multiple mode propagation in the curved region.

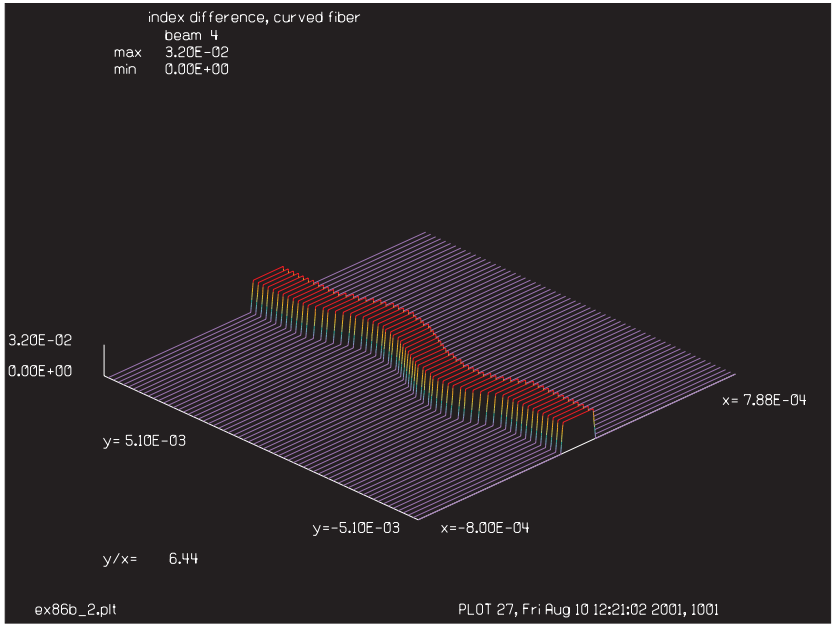


Fig. 5.46. Profile of s-bend in high-index core

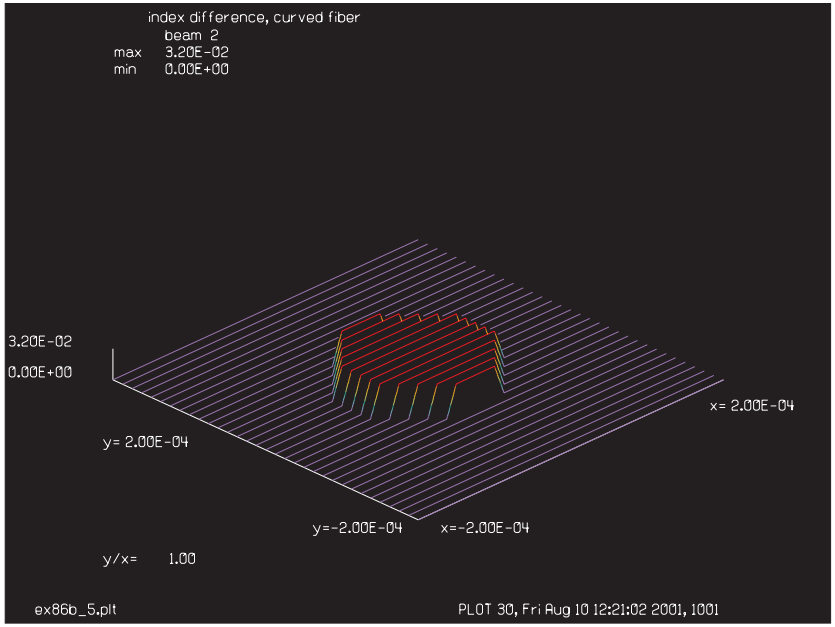


Fig. 5.47. Cross section of 3D high-index core

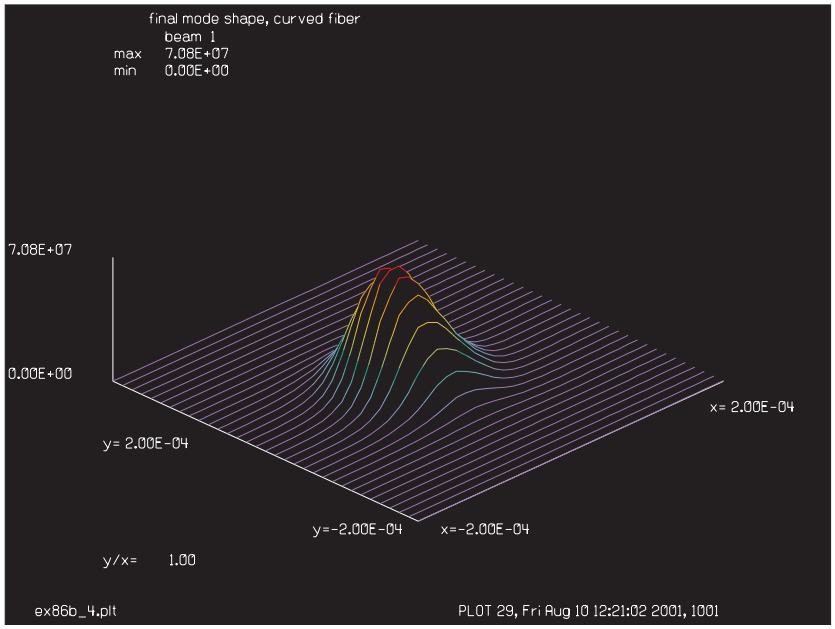


Fig. 5.48. Waveguide mode at the end of the S-bend. Note asymmetry.

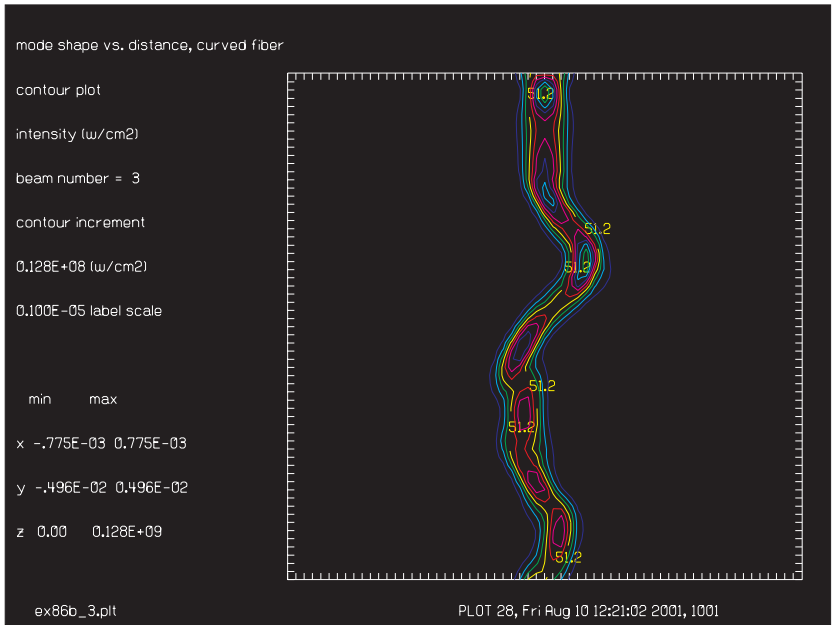


Fig. 5.49. Contour plot of mode as it propagates through the s-bend.

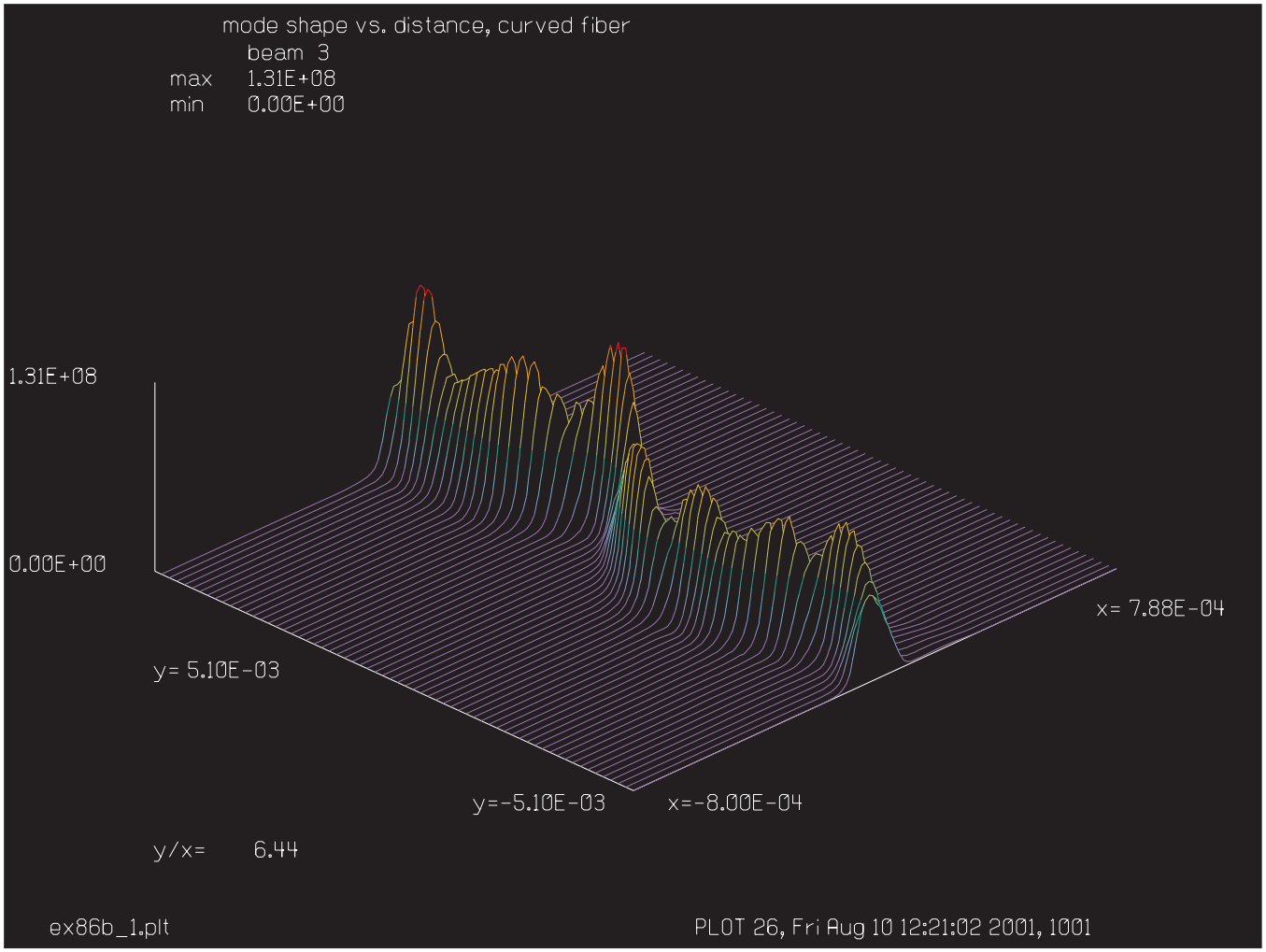


Fig. 5.50. A series of profiles as the beam travels from left to right showing the mode disruption due to s-bend.

Installation, Demo Modes, Distributors

The distribtion files contains the GLAD program and a complete set of on-line manuals. With purchase you receive a permanent key (dongle).

Demo Modes

Without a security key, GLAD will operate in one of two demo modes:

- Super Demo. Full-function operation for a limited period of time. GLAD reverts to regular Demo mode after Super Demo expires.
- Demo (regular). Limited functionality, does not expire. Examples can be run “as is” but not changed. Up to 20 interactive lines per session may be entered.

Installation

To install GLAD from the CD ROM, read details in \glad\readme.txt and install by running \glad\setup.exe.

See the secton “How to Run GLAD” in this document and the GLAD Commands Manual (commands.pdf) for operational details.

First Look

For a quick look at some popular examples, select “Demo” from the main menu.

System Requirements

GLAD requires Microsoft Windows 95 or Windows NT 3.5 (or higher), 8 Mbytes of disk space for GLAD, 32 Mbytes of disk space for the online documentation (optional, may be read from the CD), 12 Mbytes of memory for Win 95, and 16 Mbytes for Win NT.

Distributors

A permanent license for GLAD 95/NT may be purchased from Applied Optics Research:

Applied Optics Research
1087 Lewis River Rd. #217, Woodland WA, 98674 USA
tel: 360 225 9718, fax: 360 225 0347
email: glad@aor.com, http://www.aor.com

In Europe contact:

Optima Research, Ickleton Road, Duxford
CB2 4XQ, United Kingdom
Tel: 44 (0 in UK) 1223 837830, Fax: 44 (0 in UK) 1223 837831
Email: more-info@optima-research.com, http://www.optima-research.com

In Japan contact:

Leadintex International, Inc., 1-3 Nihonbashi-Ohdenmacho, Chuo-Ku,
Tokyo 103, Japan
tel: 8103-3661-5041, fax: 8103-3661-4005