QUADOA

optical systems

QUADOA - OPTICAL CAD

SOFTWARE MANUAL V. 24.08

Contents

1 The philosophy behind Quadoa

Quadoa Optical CAD is an optical design, simulation and analysis software with a unique architecture. The main difference compared to other software in this area is, that in Quadoa Optical CAD the optical system is not described as a list of surfaces that the light passes through, but as a hierarchical structure of high level elements like lenses or beamsplitters that can be arranged in assemblies. Furthermore, the optical system description is strictly separated from the raytracing. This means that the model of the optical system describes all the physical objects like lenses, mirrors, prisms, beam splitters etc., but not the properties of the light we want to trace through the system. The description and the setup of the light properties is done separately when setting up sequences to trace. By this unique approach of the separation of the optical system geometry and light sequence, Quadoa Optical CAD gains multiple benefits:

- **Multiple sequences in one model:** Contrary to other optical design software, it is easily possible to have more than one sequence in a single model. This is extremely useful when performing ghost analysis, or for designing and simulating optical systems with more than one optical path of interest, like e.g. interferometers.
- **Use Non-Sequential Raytracing only where it is really needed:** In general non-sequential raytracing, it takes more time for the computation, than sequential raytracing. This is because the time complexity of the non-sequential algorithm is exponential with the number of surfaces $\mathcal{O}(2^n)$. Therefore, for large numbers of surface interactions, non-sequential raytracing gets extremely slow. However many problems, like e.g. ghost analysis, can easily be performed through sequential raytracing by analyzing all the optical paths that may occur up to a certain number of reflection at optical surfaces. In Quadoa, this task can be performed within a single model by auto generated ghost sequences saving lots of computation time.
- **Easy setup of double path systems:** Double- and in general multi-path systems can easily be designed, by specifying the optical system in the Optical System Editor and afterwards by specifying the exact path the rays should travel through the system. With this approach, tracing rays through a surface is possible as many times as desired, without the need of copying the surface inside the list containing the surface properties.

- **Mechanics Integration:** An optical system can never exist without the mounting mechanics that hold the lenses and mirrors in place. A high-end system can never reach its design performance without a well designed mechanical mounting. Therefore, the collaboration of the optical- and mechanical designers is of great importance. To minimize the risk of errors that may get introduced during the exchange between the optical and mechanical design process, Quadoa allows to directly include mechanical components into the optical design model. The optical system can be as well exported to CAD data in order to import the data in mechanical CAD software. Through these data exchange features, errors like vignetting at mechanical parts or incorrect dimensions can be identified and fixed quickly. This saves a lot of time and effort due to less troubleshooting.

2 Getting Started

A main focus of the development of Quadoa Optical CAD is to make the usage of the software as well as the user interface as intuitive as possible. Nevertheless, not every feature can be completely self explaining. To keep the learning curve as shallow as possible, we provide a wide range of learning materials that come with your Quadoa Optical CAD installation.

2.1 Example Folder

With every Quadoa Optical CAD installation an example model folder is installed. Models from the folder can be found under the "File" tab on the "Open Examples" button. Here you will find a wide range of examples, reaching from models that showcase basic features, available surface properties, different options for coordinate systems or multi-sequential-raytracing. Furthermore, real word examples of optical systems are included, reaching from imaging systems like camera lenses, zoom lenses, telescopes and microscope objectives to complex systems like interferometers, spectrometers, free-from optics and more.

2.2 Video Tutorials

Video tutorials are a great way to learn and improve skills, especially when it comes to the usage of a new software. Quadoa Optical CAD provides video tutorials on all basic features of the software. The video tutorials can be found under the "Help" tab on the "Video Tutorial" button.

2.3 Manual

The Quadoa Optical CAD manual - which you are reading right now - is the best source for information concerning the details on how objects are defined or what functions and algorithms do.

The manual contains an overview on the parameters of each object or surface-property-item that is available. Under "Help" you will find the "Open Manual" button.

2.4 Online Forum

The online user forum is available at https://www.forum.quadoa.com In the forum, users have the possibility to interact with other users, sharing optical designs, posting questions regarding the use of Quadoa Optical CAD, general questions about optical design, simulation and more, as well as answering questions of other users. To be able to use the forum a user registration is required. The forum is available to all users, also those, who currently do not have an active support contract.

3 User Interface

The Graphical User Interface (GUI) consists out of different main areas as shown in figure 3.1.

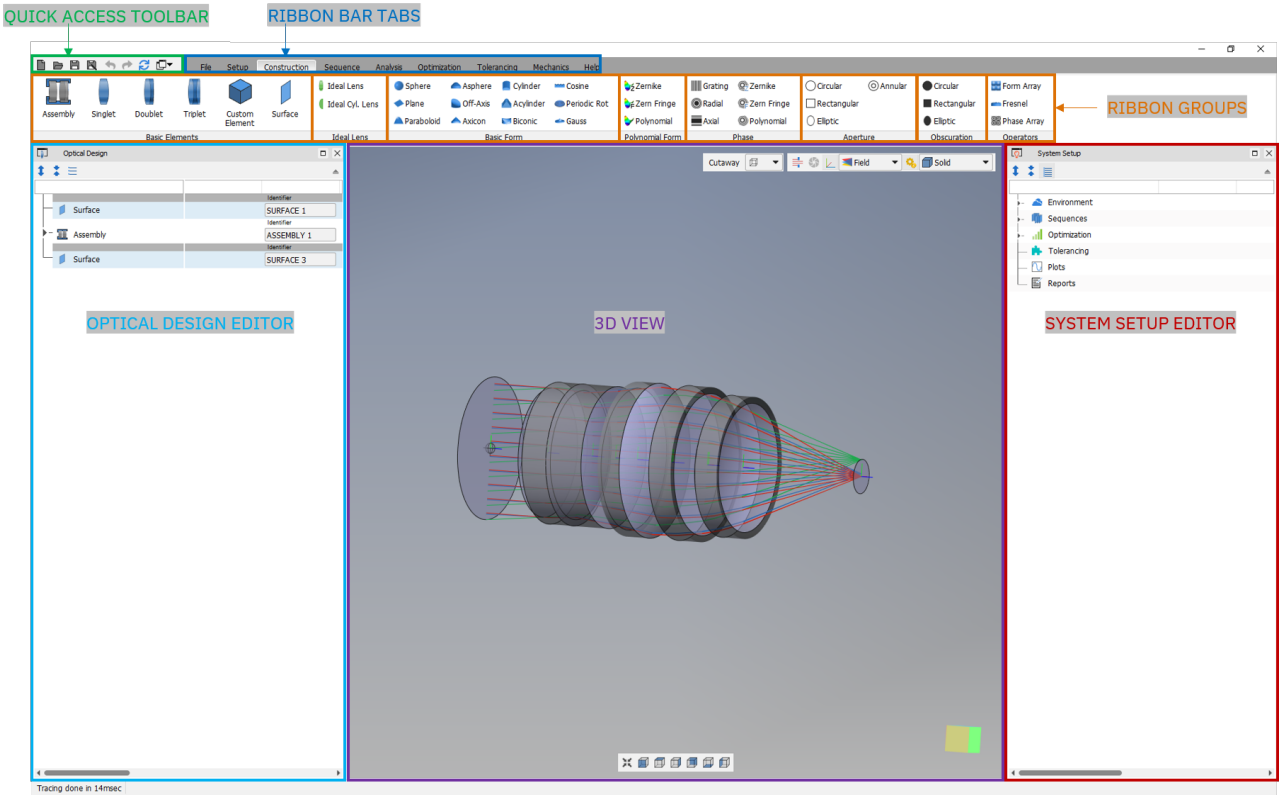


Figure 3.1: Graphical User Interface (GUI) description.

3.1 Quick Access Toolbar

The *Quick Access Toolbar* provides a quick way to commonly used functions and is always visible. Following buttons are contained:

- **New** (Ctrl + N): Generate a new empty model file.
- **Open** (Ctrl + O): Open a model file.

-  **Open** (Ctrl + O): Open a recently closed model file.
-  **Save** (Ctrl + S): Save model file. If the model was opened from an already existing file, the model will be saved in the same file. *Caution: the old model will be overwritten.*
-  **Save As** (Ctrl + Shift + S): Save model file with specific name and path.
-  **Undo** (Ctrl + Z): Previous action is undone.
-  **Redo** (Ctrl + Shift + Z): Previous Undo is undone.
-  **Update** (F5): Refresh current model.
-  **Auto Update**: Auto Refresh current model.
-  **Windows**: Open Editor Windows (*Optical Design Editor*, *System Setup Editor*, *Slider Interface*, *Multiconfig Lookup*).

3.2 3D-View

The *3D-View* is the central part of the Quado user interface. In the *3D-View* the optical system is shown as defined in the *Optical Design Editor* and the *System Setup Editor*. The view can be rotated, translated and zoomed. Furthermore, different drawing styles and cutaways are available. Besides the path of the rays and the polarization state of each ray segment can be visualized. Rays are drawn as solid lines when they are real rays and as dashed lines when they are virtual (see chapter 5.4 on the general definition of virtual rays). The right click tool menu inside the *3D-View* can be used for directly editing lenses of the optical system.

The background color as well as the ray colors can be changed in the *Global Settings Dialog*.

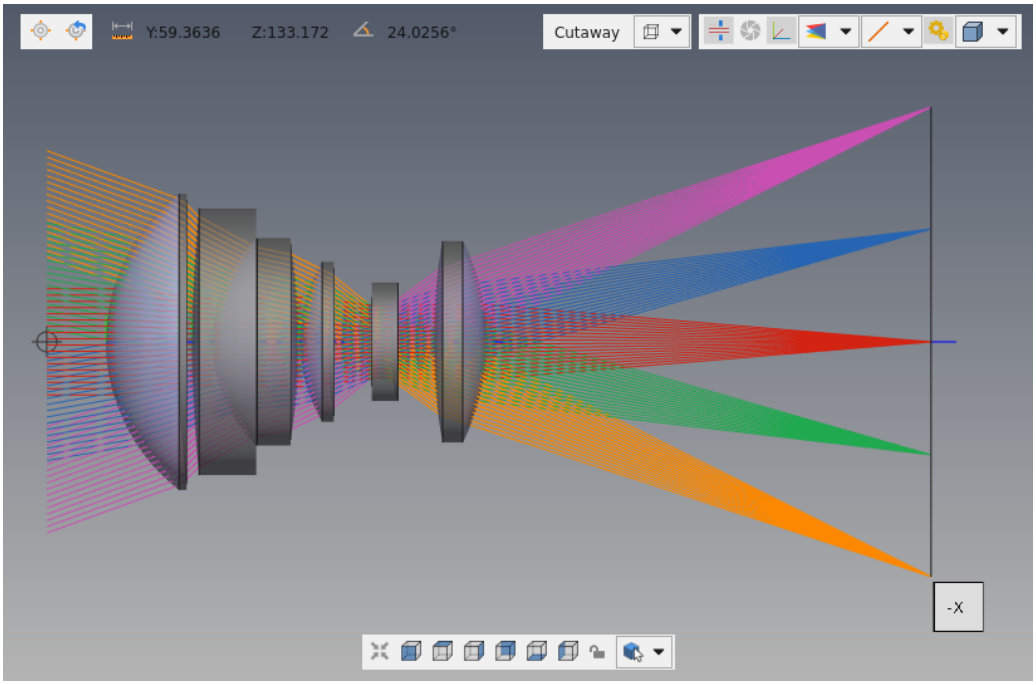


Figure 3.2: 3D View

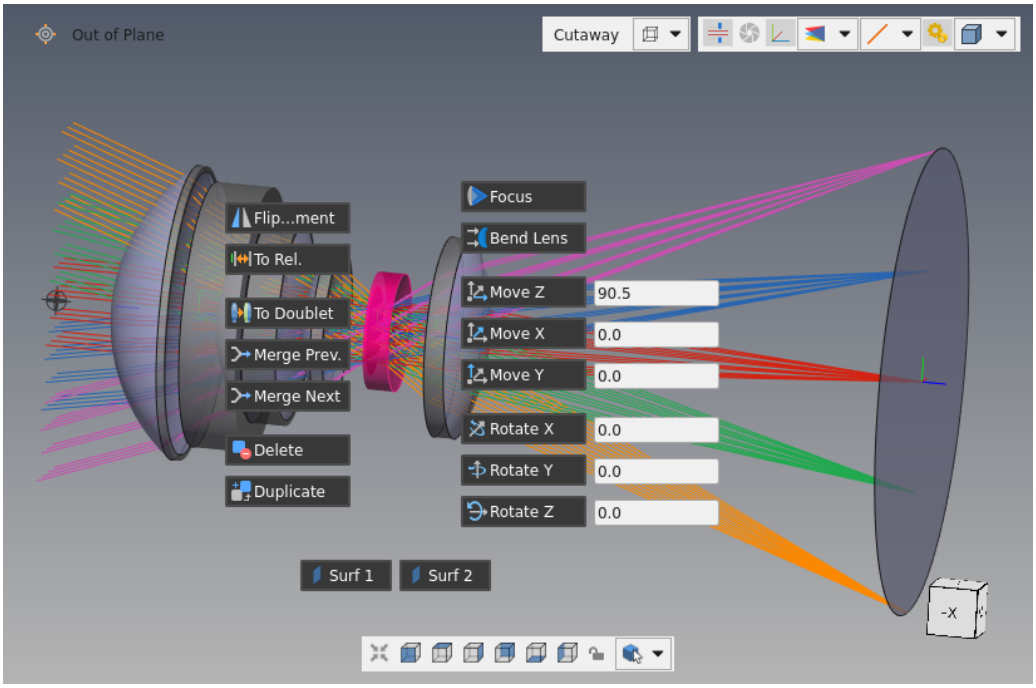


Figure 3.3: Interactive lens editing tool menu inside the 3D View

The control of the *3D-View* works as follows:

- **Left Mouse Button Move:** Turn the 3D-View of the Optical System with pressed left mouse button. Or select a lens or surface by clicking on it.
- **Right Mouse Button Move:** Move the 3D-View of the Optical System with pressed right mouse button or open the right click menu for a lens or surface by right clicking on it.
- **Right Mouse Button Click:** When clicked on a lens the edit tool menu for the lens will appear. Clicking on a blank are will close the tool.
- **Scrolling Wheel:** Scroll with the mouse up and down to zoom in and out of the 3D-View.
- **Reset Zoom:** Reset the zoom to default setting.
- **Set View:** Switch between different preset plane-views.
- **Select Element:** Select a whole element (e.g. Lens) when hovering over an element with the mouse pointer.
- **Select Surface:** Select only a surface when hovering over an element with the mouse pointer.
- **Select Assembly:** Select the whole associated assembly when hovering over an element.

- **Select Aperture:** Select only the aperture when hovering over an element.
- **Coordinate Reference Point:** Set a new coordinate reference point for setup of a new zero point. Default coordinate reference point is at zero point (0,0,0).
- **Tape Measure:** Shows the distance in y- and z-direction of the cursor-position to the coordinate reference point. The tape measure is only enabled, if view is set to YZ-Plane.
- **Angular Measure:** Shows the angle of the cursor-position to the coordinate reference point. The angular measure is only enabled, if view is set to YZ-Plane.
- **Cutaway:** Select different planes to switch between different Cutaways.
- **Lock View:** Lock/Unlock the rotation of the view.
- **Cutaway Slider:** Move the handle of the slider to increase or decrease the extent of the cutaway.
- **Show/Hide Multi configuration:** Show only the current configuration or an overlay of all configurations. If all configurations are shown, it is further possible to offset each configuration in Y so they are drawn side by side.
- **Show/Hide Vignetted Rays:** Show or hide vignetted rays in the 3D-View.
- **Show/Hide Exit Pupil:** Show or hide the exit pupil in the 3D-View.
- **Show/Hide Coordinate Systems:** Show or hide the coordinate systems in the 3D-View.
- **Select Ray Coloring:** Color the rays by Sequence, Field Wavelength or Spectrum. Spectrum will assign a color to each ray that tries to match the color of how a certain visible wavelength is perceived. The coloring scheme for the other options can be selected in the global settings.
- **Show/Hide Mechanical Parts:** Show or hide mechanical parts in the 3D-View.
- **Set Surface Draw Style:** Style the surface of the elements in the 3D-View (Solid, Wire Frame, Line Cut, Line Cut 2D).

3.3 Ribbon Bar Tabs

The *Ribbon Bar* provides one tab per category to work with Quadoa. Most functions can be found there. Often the functions are also available in the right click menu of the items in the views.

3.3.1 File Tab

The file tab contains any file related functions as loading and saving of models, import and export of third party file formats as well as the export of technical drawings. It is divided into the following categories:

3.3.1.1 File

The *File* category contains any functions to load or save Quadoa model files:

- **Model** (Ctrl + N): Generate a new empty model file.
- **Open Model** (Ctrl + O): Open a model file (proprietary Quadoa File Format *.optx).
- **Open Example**: Opens the example folder to open an example model (proprietary Quadoa File Format *.optx).
- **Save** (Ctrl + S): Save model file. If the model was opened from an already existing file, the model will be saved in the same file. **Caution: the old model will be overwritten**.
- *Save As*: Save model file with specific name and path.

3.3.1.2 Assembly

The *Assembly* category contains functions to import or export Assemblies or Lenses in between different Quadoa models. This can especially be used to e.g. export an objective lens as assembly file ('*.asbl') to insert it within another model that contains the lens as a sub assembly.

- **Import Assembly**: Import an element or assembly to selected assembly (proprietary Quadoa File Format *.asbl).
- **Export Assembly**: Export selected element or assembly to file (proprietary Quadoa File Format *.asbl).

3.3.1.3 Optical Design Files

The *Optical Design Files* category allows the import and export of third party optical design files used with other optical design software.

- **Import Optical Design**: Import third party optical design model (e.g. *.csv, *.seq or *.zmx) .
- **Import Assembly**: Import third party assembly (e.g. *.csv, *.seq or *.zmx) to current model.
- **Export Optical Design**: Export third party optical design model (e.g. *.csv, *.seq or *.zmx) .

3.3.1.4 Mechanical CAD

The *Mechanical CAD* category allows to export the model as CAD file as well as surface shapes in form of 3D point clouds, that can be used for e.g. diamond turning control of complex free-form surfaces.

- **Export to CAD**: Export whole model, single elements or single assemblies as CAD file format (e.g. *.STL, *.STEP, *.IGES) to exchange optical design with mechanical CAD.
- **Export Pointcloud**: Export surface form or phase to XYZ-file.

3.3.1.5 Technical Drawings

The *Technical Drawings* category is used to export drawings.

- **Assembly Drawing**: For creating a Technical Lens Drawing of a specific Assembly, click on the icon.
- **ISO 10110 Lens Drawing**: For creating an ISO 10110 Technical Lens Drawing of a specific element or lens, click on the icon. The Lens Drawing Wizard will open. For further information, see section 3.6.1.4 Technical Drawings ISO 10110:.
- **Export 3D View Image**: Export the current 3D-View Scene as an image.

3.3.2 Setup Tab

The *Setup Tab* contains the main project settings and manages the editors as well as catalogs and the scripting wizards.

3.3.2.1 Settings

The *Settings* category contains any global or project related settings of the user interface.

- **Global**: Open the Global Quadoa Settings to edit global settings for all projects as follows:
  - **Application Style**:
    Application Style: Editable Options: Select a Quadoa Theme Style (Light, DarkGray or DarkBlack).
    Plot and Ray Style: Editable Options: Ray Coloring Scheme, Plot Line Thickness, Plot Font Size, Default Plot Color, Default 3D Surface Style, Default 3D Image Style, Default ColorBar.
    3D View: Editable Options: Surface Resolution, Background Color, Ray Line Thickness, Default Mechanics Color, Lens Transparency, Mechanics Transparency.
  - **Navigation**:
    Tree View Navigation: Select mode for Keyboard Navigation in Tree View.
    3D View Mouse Navigation: Select Navigation Style for 3D View (Rotation, Translation).
  - **Units and Precision**:
    Decimal Points in Editor: Editable Options: General Editors, Tolerance Reports, Zernike Reports.
  - **ISO10110**:
    Decimal Points in Editor: Settings for Technical Drawing Report.
  - **File and Save**:
    User Data Folder: Folder for storing user defined catalogs like coatings and materials.
    Autosave Files: Editable Options: On Change, On Timer, Disabled, Time.
  - **Shortcuts**:
    General Shortcuts: Editable Options: Undo, Redo, Raytrace Update.
    File Shortcuts: Editable Options: New Model, Open Model, Save Model, Save As, Import Assembly, Export Assembly, Import Optical Design, Import 3rd Party Assembly, Export Optical Design, Export CAD, ISO10110 Drawing.

Setup Shortcuts: Editable Options: Optical Design Editor, System Editor, Multiconfig Editor, Slider Editor, Material Catalog, Lens Catalog, Python Wizard, Run Last Python Script, Run Python Script, Matlab Wizard.
Construction Shortcuts: Editable Options: Add Assembly, Add Singlet, Add Doublet, Add Triplet, Add Custom Element, Add Surface.
Plot Shortcuts: Editable Options: All Analysis Plots.
Optimization Shortcuts: Editable Options: Local Optimization, Extended Optimization, Global Optimization, Move to Focus, Substitute Materials, Remove Variables, Make all Radii variable.
Multiconfig Shortcuts: Editable Options: Previous Config, Next Config.

- **Python**:
  Specify Paths for Python and Editor: Editable Options: Path to Python 3, Path to Python Editor.

- **Project**: Settings that are specific to a single model file.

3.3.2.2 Editors

The *Editors* category allow to open the main editors. Another option to open them is via the *Quick Access Toolbar*.

- **Optical Design**: Opens the Optical Design Editor.
- **System Setup**: Opens the System Setup Editor.
- **Multi Config**: Opens the Multi Config Lookup Table.
- **Slider Interface**: Opens the Slider Interface.
- **Reset Editor**: Reset all Editor Positions to factory default settings.

3.3.2.3 Catalogs

The *Catalogs* category holds any catalogs that are shipped with Quadao or have been added by the user.

-  **Material Catalog**: The material catalogs hold Lens Materials as well as materials for Thin Film Stack Coatings. For searching for materials, click on the icon to open the Material Catalog Wizard. For further information, see section 3.6.2.1 Material Catalog:.
-  **Lens Catalog**: For searching for lenses, click on the icon to open the Lens Catalog Wizard. For further information, see section 3.6.2.2 Lens Catalog:.
-  **Reload Catalogs**: Reload catalog files and scan for new catalogs in the catalogs folder.

3.3.2.4 Script Interface

The *Script Interface* category contains the Scripting wizards as well as the possibility to run scripts directly from within the GUI.

-  **Python Wizard**: Shows Python functions help for QuadoaCore.
-  **Matlab Wizard**: Shows Matlab functions help for QuadoaCore.
-  **Run Last**: Run last script.
-  **Run**: Run script from local model folder.

3.3.3 Construction Tab

In the *Construction Tab*, the Optical Elements of the optical system can be added and modified. Any Optical Element can be added to the Optical Design Editor by clicking the icon in the ribbon menu. If an element in the Optical Design Editor is selected, the new item will - if possible - be added as child of the selected element. Otherwise it will be added besides the selected element. Surface Properties, e.g. Surface Shapes or Phase Functions can only exist as a child of a Surface and therefore can only be added when a Surface in the Optical Design Editor is selected.

3.3.3.1 Basic Elements

The *Basic Elements* category contains any buttons to add any Optical Elements to the optical system.

-  **Assembly**:
  By clicking with the left mouse button on the Assembly-Icon, a new Assembly will be inserted to the *Optical Design Editor*. If previously no Assembly was selected, the Assembly will be

inserted to the Global Coordinate System. If an Assembly was selected, the new Assembly will be inserted to the Local Coordinate System of the Assembly as child-element. For further information about the properties of an Assembly please read chapter 7.4.

-  **Singlet**:
  By clicking with the left mouse button on the Singlet-Icon, the Singlet Lens Wizard will open for easy setup of a Singlet Lens. (See section 3.6.3.1). If an Assembly is selected, the Singlet Lens will be inserted to the Local Coordinate System of the Assembly as a child-element. For further information about the properties of an Singlet Lens please read chapter 7.2.1.

-  **Doublet**:
  By clicking with the left mouse button on the Doublet-Icon, the Doublet Lens Wizard will open for easy setup of a Doublet Lens. (See section 3.6.3.2). If an Assembly is selected, the Doublet Lens will be inserted to the Local Coordinate System of the Assembly as a child-element. For further information about the properties of an Doublet Lens please read chapter 7.2.2.

-  **Triplet**:
  By clicking with the left mouse button on the Triplet-Icon, the Triplet Lens Wizard will open for easy setup of a Triplet Lens. (See section 3.6.3.3). If an Assembly is selected, the Triplet Lens will be inserted to the Local Coordinate System of the Assembly as a child-element. For further information about the properties of an Triplet Lens please read chapter 7.2.3.

-  **Custom Element**:
  By clicking with the left mouse button on the Custom Element-Icon, a Custom Element will be inserted to the Global Coordinate System. If an Assembly is selected, the Custom Element will be inserted to the Local Coordinate System of the Assembly as a child-element. For further information about the properties of a Custom Element please read chapter 7.3.

-  **Surface**:
  By clicking with the left mouse button on the Surface-Icon, a Surface will be inserted to the Global Coordinate System in the Optical Design Editor. If an Assembly or an Custom Element is selected, the Surface will be inserted to the Local Coordinate System of the selected element as a child-element. For further information about the properties of a Surface please read chapter 7.1.

3.3.3.2 Ideal Lens

The *Ideal Lens* category is used to add idealized elements like Ideal Lenses without aberrations to the optical system. If an Assembly is selected, the Ideal Element will be inserted to the Local Coordinate System of the Assembly as a child-element. See chapter 7.6 on Ideal Elements.

3.3.3.3 Basic Form

Any Surface in Quadoa has as default a spherical or planar shape. For any other surface types as aspheric surfaces, fresnel lenses or free-from surfaces, the Surface Form Property Items are used. To add a Surface Form Property Item to a Surface, the Surface has to be selected and then the button to add the desired Form has to be clicked. Another option is to use the right click menu on the Surface within the Optical Design Editor. The *Basics Form* holds a wide range of common optical surfaces descriptions. Beside the basic form types, polynomial form types are available (see *Polynomial Form*).

3.3.3.4 Polynomial Form

Any Surface in Quadoa has as default a spherical or planar shape. For any other surface types as aspheric surfaces, fresnel lenses or free-from surfaces, the Surface Form Property Items are used. To add a Surface Form Property Item to a Surface, the Surface has to be selected and then the button to add the desired Form has to be clicked. Another option is to use the right click menu on the Surface within the Optical Design Editor. The *Polynomial Form* holds the most common polynomials used for optical surfaces descriptions. (For basic form types see *Basic Form*).

3.3.3.5 Phase

Any Surface, no matter of its shape, can have a Phase Function assigned to it. To add a Surface Phase Property Item to a Surface, the Surface has to be selected and then the button to add the desired Phase Object has to be clicked. Another option is to use the right click menu on the Surface within the Optical Design Editor.

3.3.3.6 Aperture

Any Surface in Quadoa has a circular aperture as default. More advanced Apertures can be realized by adding one or more of the available Aperture Property Items to a Surface. To add a Aperture

Property Item to a Surface, the Surface has to be selected and then the button to add the desired Aperture has to be clicked. Another option is to use the right click menu on the Surface within the Optical Design Editor.

To lock or unlock an Aperture click on the following click on the lock symbol besides the Apertures in the Optical Design Editor. To lock or unlock all Apertures, click on following icons in the Construction Tab

- **Lock all Apertures**: Lock all floating apertures (see chapter 7.1).
- **Unlock all Apertures**: Set all apertures to floating, if no apertures via Aperture Property Items are defined (see chapter 7.1).

3.3.3.7 Obscuration

Obscurations are used to block the rays withing an area on the Surface. To add a Obscuration Property Item to a Surface, the Surface has to be selected and then the button to add the desired Obscuration has to be clicked. Another option is to use the right click menu on the Surface within the Optical Design Editor.

3.3.3.8 GRIN

GRIN objects are used to add a gradient index profile to Materials. This can be used to model gradient index lenses, but also e.g. for the modeling of material inhomogeneities. Only one grin object may be defined per Material. To add a GRIN Profile to a Material, the Material has to be selected and then the button to add the desired GRIN Profile has to be clicked. Another option is to use the right click menu on the Material within the Optical Design Editor.

3.3.3.9 Operators

Operators are used to modify Surface Form Property Items and Phase Property Items. To add a Phase Operator or Form Operator to a Surface, the Surface has to be selected and then the button to add the desired operator has to be clicked. Another option is to use the right click menu on the Surface within the Optical Design Editor. The Form or Phase which sould apply to the operator, has to be added as a child item to the operator. To do so, select the operator first and then - similar to adding a property item to a surface - the item can be added as child-element by clicking the icon in the Construction Tab. Alternatively select the operator first and via the right click menu inside the Optical Design Editor select the desired Form or Phase to add it to the operator.

3.3.4 Coating Tab

The Coating Tab contains a wide range of Coatings as well as Polarization Elements that can be assigned to a Surface.

3.3.4.1 Coatings

The *Coatings* category contains Coatings, that can be added to a Surface. To coat a Surface, the Surface has to be selected first, and afterwards the button to add the desired coating has to be clicked. Another option is to use the right click menu on the Surface within the Optical Design Editor. See chapter 12 for available Coatings.

Note that a Surface can only have one Coating at a time. If a coating is added to a Surface that is already coated, the old Coating will be replaced.

3.3.4.2 Polarization

The *Polarization* category contains Polarization Elements, that can be added to a Surface. To add a Polarization Elements to a Surface, the Surface has to be selected first and afterwards the button to add the desired Polarization Elements has to be clicked. Another option is to use the right click menu on the Surface within the Optical Design Editor. See chapter 13 for available Polarization Elements.

3.3.5 Sequence Tab

The *Sequence Tab* is used to define Sequential Raytraces within the optical system, as well as to automatically generate Sequence for the purpose of ghost reflection analysis.

3.3.5.1 Sequence Setup

The *Sequence Setup* category is used to define a basic Sequential Raytrace. (See chapter 14.1 on ray tracing methods)

- **New Auto Sequence**: The button adds a new Auto Sequence to the model. The Auto Sequence will appear inside the System Setup Editor.

-  **New User Sequence**: The button adds a new User Defined Sequence to the model. The User Sequence will appear inside the System Setup Editor.
-  **Add Surface**: The button adds a new Surface to a User Sequence. To add the Surface, the User Sequence, to which the Surface should be added, has to be selected first in the *System Setup Editor*. If no User Sequence is selected, an error message will appear. Another option to add a Surface to a Sequential Ray Trace is to select the Surface List of the Sequence within the *System Setup Editor* and use the right click menu, or press the Insert key.

  Note that a Surface can only be added to an User Sequence. The Auto Sequence will automatically generate the surface list.
-  **Add Field**: The button adds a new Field to the Source. To add the Field in case more than one Sequences exist, the Sequence or the Source, to which the Field should be added, has to be selected first. If no Sequence or Source is selected, an error message will appear. Another option is to use the right click menu on the Field.
-  **Add Wavelength**: The button adds a new Wavelength to the Source. To add the Wavelength in case more than one Sequences exist, the Sequence or the Source, to which the Wavelength should be added, has to be selected first. If no Sequence or Source is selected, an error message will appear. Another option is to use the right click menu on the Wavelength. A wide range of frequently used Wavelengths is also predefined there.

3.3.5.2 Convert

-  **To Auto Sequence**: The button will convert a User Sequence into a Auto Sequence. The User Sequence which should be converted into an Auto Sequence, has to be selected first in the System Setup Editor. If no User Sequence is selected, an error message will appear.

  Note that the user defined order of surfaces to be traced will be lost during this action.
-  **To User Sequence**: The button will convert an Auto Sequence into an User Sequence. The Auto Sequence which should be converted into an User Sequence, has to be selected first in the System Setup Editor. If no Auto Sequence is selected, an error message will appear. The selected Auto Sequence in the System Setup Editor is now converted to an User Sequence.

3.3.5.3 Vignetting Factors

Vignetting factors can be automatically generated or reset for a Sequence. See chapter 14.1.11.1 on the definition and use of Vignetting Factors.

- **Compute**: Compute Vignetting Factors for a sequence.
- **Reset**: Reset the Vignetting Factors for a sequence.

3.3.5.4 Ghost Analysis

- **Generate Ghosts**: The button is used to generate Ghost Reflexes from a selected Sequence. The Ghost Sequence Wizard will open. For further information, see section 3.6.4.1 Ghost Wizard:.
- **Diffractive Ghosts**: The button is used to generate Diffractive Ghost Reflexes on Surfaces with a Phase Function from a selected Sequence. The Diffractive Ghost Sequence Wizard will open. For further information, see section 3.6.4.2 Diffractive Ghost Wizard:.
- **To User Sequence**: The button is used to convert a Ghost Sequence into an User Sequence. The Ghost Sequence which should be converted into an User Sequence, has to be selected first in the System Setup Editor. If no Ghost Sequence is selected, an error message will appear. The selected Ghost Sequence in the System Setup Editor is now converted to an User Sequence.
- **To Ghost Sequence**: The button is used to convert an User Sequence into a Ghost Sequence. The User Sequence which should be converted into a Ghost Sequence, has to be selected first in the System Setup Editor. If no User Sequence is selected, an error message will appear. The selected User Sequence in the System Setup Editor is now converted to a Ghost Sequence and will be also included in the ghost analysis as other Ghost Sequences.
- **Remove all Ghosts**:The button is used to remove all Ghost Sequences from the Sequence List in the System Setup Editor.
- **Sort by Illuminance**: The button is used to sort the Ghost Sequences by maximum illuminance on the last surface (imaging surface). The Ghost Sequence List in the System Setup Editor is now sorted, beginning at the top with the Ghost Sequence, which has the most impact (regarding to illuminance) on the last surface (imaging surface) and ending with the Ghost Sequence, which has least impact (regarding to the illuminance) on the last surface (imaging surface).

- **Sort by Total Flux**: The button is used to sort the Ghost Sequences by maximum total flux on the last surface (imaging surface). The Ghost Sequence List in the System Setup Editor is now sorted, beginning at the top with the Ghost Sequence, which has the most impact (regarding to total flux) on the last surface (imaging surface) and ending with the Ghost Sequence, which has least impact (regarding to total flux) on the last surface (imaging surface).
- **Hide All Ghosts**: This button is used to hide all Ghost Sequences. The Ghost Reflexes will be hided in the 3D View and grayed out in the System Setup Editor. To hide or show single Ghost Sequences, the right click menu on the corresponding Sequence can be used.
- **Show All Ghosts**: This button is used to show all Ghost Sequences. The Ghost Reflexes will be shown in the 3D View and displayed in color again in the System Setup Editor. To hide or show single Ghost Sequences, the right click menu on the corresponding Sequence can be used.

3.3.6 Analysis Tab

The *Analysis* tab gives access to any analysis plots and reports available, that can be used for the evaluation of an optical system. On details about the available analysis features see chapter 17.

3.3.7 Optimization Tab

The *Optimization* tab holds any optimization related features.

3.3.7.1 Solver

The *Solver* category gives access to the available solver dialogs.

- **Local Optimization**: Run local optimization (see chapter 18.7.1).
- **Extended Optimization**: Run extended optimization (see chapter 18.7.2).
- **Global Optimization**: Run global optimization (see chapter 18.7.3).
- **Move to Focus**: Moves the last Surface in a Sequence to focus. As the criterion either the spot size or the RMS wavefront error can be chosen. The direction to move the Surface can be chosen ("X", "Y", "Z"), where "Z" is the default.

- **Collimate**: Moves an Element of a Sequence that can be defined by the user, so that the wavefront at the last surface is collimated. The direction to move the Element can be chosen ("X", "Y", "Z"), where "Z" is the default.
- **Replace Model**: Replaces all model materials with the closest match that is provided by the Substitute Material List. (Replacing a single material by the closest match can be achieved via the right click menu of in the 3D-View).

3.3.7.2 Optimization Settings

- **Substitute Materials**: Edit the Substitute Material List for a model. Materials and whole Catalogs can be added or removed from the list. (See chapter 18.6).
- **All Rad Var**: Set any radius parameter in the model to be variable.
- **All Air Var**: Set any Z-Position of a Lens in the model to be variable.
- **All Thick Var**: Set any Z-Position of a Lens Surface that is not the first one to be variable.
- **Remove all Variables**: Disables the activation state of any Variable in the model.
- **All Subst**: Enables the substitution for any Material in the model, that is specified via a catalog, by a material from the Substitute Material List .
- **Rem. Subst.**: Disables the substitution for any Material in the model.

3.3.7.3 Merit Function

The *Merit Function* section is used to define and edit merit functions.

- **Merit Function**: Generates a new Merit Function (see chapter 18.4).
- **Add Ray Trace**: Adds a Sequential Ray Trace to the selected Merit Function (see chapter 18.4.2).
- **Aberrations**: Adds an Aberration Type Optimization Goal to the selected Merit Function. To add the Optimization Goal as Constraint, the Constraint Container inside a Ray Trace of the Merit Function has to be selected before adding the Goal.

- **Optical Properties**: Adds an Optical Property Type Optimization Goal to the selected Merit Function. To add the Optimization Goal as Constraint, the Constraint Container inside a Ray Trace of the Merit Function has to be selected before adding the Goal.
- **Ray Data**: Adds an Ray Data Type Optimization Goal to the selected Merit Function. To add the Optimization Goal as Constraint, the Constraint Container inside a Ray Trace of the Merit Function has to be selected before adding the Goal.
- **Dimensional Properties**: Adds an Dimensional Property Type Optimization Goal to the selected Merit Function. To add the Optimization Goal as Constraint, the Global Constraint Container inside the Merit Function has to be selected before adding the Goal.

3.3.8 Tolerancing Tab

The tolerancing workbench is used to specify the tolerances for any elements or assemblies within the optical system. Further tolerance analysis can be carried out.

3.3.8.1 Wizards

- **Default Tolerances**: The button will open the *Default Tolerance Wizard*, which is used to generate the global tolerances that apply to all elements if no other tolerances have explicitly been specified for those. See chapter 3.6.6.1 as well as chapter 19 on tolerance analysis.

3.3.8.2 Specify ID

The *Specify ID* category is used to define tolerances for specific objects that differ from the global tolerances defined by the wizard. (See chapter 19.1.1).

- **Element**: Adds a Tolerance Container for a specific Element. (See chapter 19.1.1).
- **Surface**: Adds a Tolerance Container for a specific Surface. (See chapter 19.1.1).
- **Assembly**: Adds a Tolerance Container for a specific Assembly. (See chapter 19.1.1).

3.3.8.3 Element

The *Element* category is used to define Element tolerances as global tolerances or specific objects. To add the Tolerance Parameter, the Element Tolerance Container where the tolerance should be added to, has to be selected.

- **Position**: Adds a tolerance for the Elements position to the selected Element Tolerance Container
- **Angle**: Adds a tolerance for the Elements angle to the selected Element Tolerance Container
- **Thickness**: Adds a tolerance for the Elements thickness to the selected Element Tolerance Container
- **Index**: Adds a tolerance for the Elements refractive index to the selected Element Tolerance Container
- **Index**: Adds a tolerance for the Elements Abbe number to the selected Element Tolerance Container

3.3.8.4 Surface

The *Surface* category is used to define Surface tolerances as global tolerances or specific objects. To add the Tolerance Parameter, the Surface Tolerance Container object the tolerance should be added to has to be selected.

- **Decenter**: Adds a tolerance for the Surfaces position to the selected Surface Tolerance Container
- **Tilt**: Adds a tolerance for the Surfaces tilt to the selected Surface Tolerance Container
- **Radius**: Adds a tolerance for the Surfaces radius to the selected Surface Tolerance Container
- **Zernike**: Adds a tolerance for the Surfaces Zernike form error to the selected Surface Tolerance Container
- **Position Z**: Adds a tolerance for the Surfaces Z-Position to the selected Surface Tolerance Container. This is only available for single Surfaces that are not part of an Element. For surfaces inside Elements, the Element Thickness Tolerance or the Element Position Tolerance is used.

- **General**: Adds a general tolerance to the selected Surface Tolerance Container. The General Tolerance allows to specify a tolerance for any Parameter of a Surface feature like a Asphere Coefficient or Zernike Phase term.

3.3.8.5 Assembly

The *Assembly* category is used to define Assembly tolerances as global tolerances or specific objects. To add the Tolerance Parameter, the Assembly Tolerance Container object the tolerance should be added to has to be selected.

- **Position**: Adds a tolerance for the Assemblies position to the selected Assembly Tolerance Container
- **Angle**: Adds a tolerance for the Assemblies angle to the selected Assembly Tolerance Container

3.3.8.6 Multiconfig

- **Add**: Adds a toleranc for a multiconfig parameter.

3.3.8.7 Compensators

- **Add**: Adds a compensator to the tolerancing. (See chapter 19.3 on compensators).

3.3.8.8 Simulations

The *Simulations* category is used to run the tolerance analysis, after the tolerances for the model have been specified.

- **Monte-Carlo Simulation**: Runs a Monte Carlo Tolerance Analysis. (See chapter 19.4).
- **Sensitivity Analysis**: Runs a Sensitivity Tolerance Analysis. (See chapter 19.5).

3.3.9 Mechanics Tab

The *Mechanics* workbench is used to include mechanical parts like lens mounts, enclosures or any other non optical components into the optical system. (See chapter 20)

3.3.9.1 Primitives

The *Primitives* category allows to add any available Primitive Mechanical Part to the optical system.

3.3.9.2 CAD

The *CAD* category allows to add CAD Parts that have been created with third party mechanical CAD software to the optical system. (See chapter 20.0.2 on supported file formats)

3.3.9.3 Analysis

- **Check Ray Intersections**: The Ray Intersection check is used to check if there are intersections between the optical rays and the mechanical parts in the optical system. A popup message with the information about the number of optical rays which are intersecting with the mechanical part will appear. The mechanical parts, which do not intersect with the optical rays, are highlited in green. The mechanical parts, which do intersect with the optical rays, are highlighted in red.
- **Clear Highlight**: The button clears the highlights (green/red), which appeared from the Ray Intersection Analysis and restores the default colors of the mechanical parts.

3.3.10 Help Tab

- **Open Manual**: Opens the Quadoa Optical CAD user manual as *.PDF file format.
- **Video Tutorials**: Opens the website with the Quadoa Optical CAD video tutorials.
- **Python Help**: Opens the Python Help to show how to install and work with Quadoa in Python and to show all Core and DataObject Classes. (See chapter 24.1 for basic usage).
- **Matlab Help**: Opens the Matlab Help to show how to install and work with Quadoa in Matlab and to show all Core and DataObject Classes. (See chapter 24.2 for basic usage).
- **Go To Website**: Opens the official website of Quadoa Optical Systems.
- **About Quadoa**: Opens the About Quadoa Window that contains information about Quadoa Optical CAD as well as information on Open Source projects that have been used for Quadoa.

- **Show License**: Opens the Software Licensing Conditions.

3.4 Editors

3.4.1 Optical Design Editor

In the *Optical Design Editor*, the optical design is defined. This includes the position, thickness and materials of Lenses, the Surface Shapes and so on. The definition of the sequential ray trace including the Fields and Wavelengths is set up via the *System Setup Editor*. The elements are displayed in a tree-like structure. Any object that is added as child element of an overlying element is part of that element. E.g. a Surface can be part of a Lens and a Lens can be part of an Assembly. Besides that, Surfaces can also have child elements. These property items define the properties of the Surfaces, e.g. the Shape, the Aperture, the Phase Function and so on. Each column of the view holds one parameter of the object, e.g. the *Position* of the Lens or the *Radius* of a Surface. For the navigation and editing inside of the tree view, see Editor Operations in chapter 3.4.3. The Variable state of a parameter to be used for the Optimization, can be toggled by clicking the V icon or by pressing the 'Space' key. The icon will turn green V when the parameter is set to variable. The current Merit Function or Compensators for which the Variables should be edited, can be selected via the Combobox above the tree view of the Optical Design Editor.

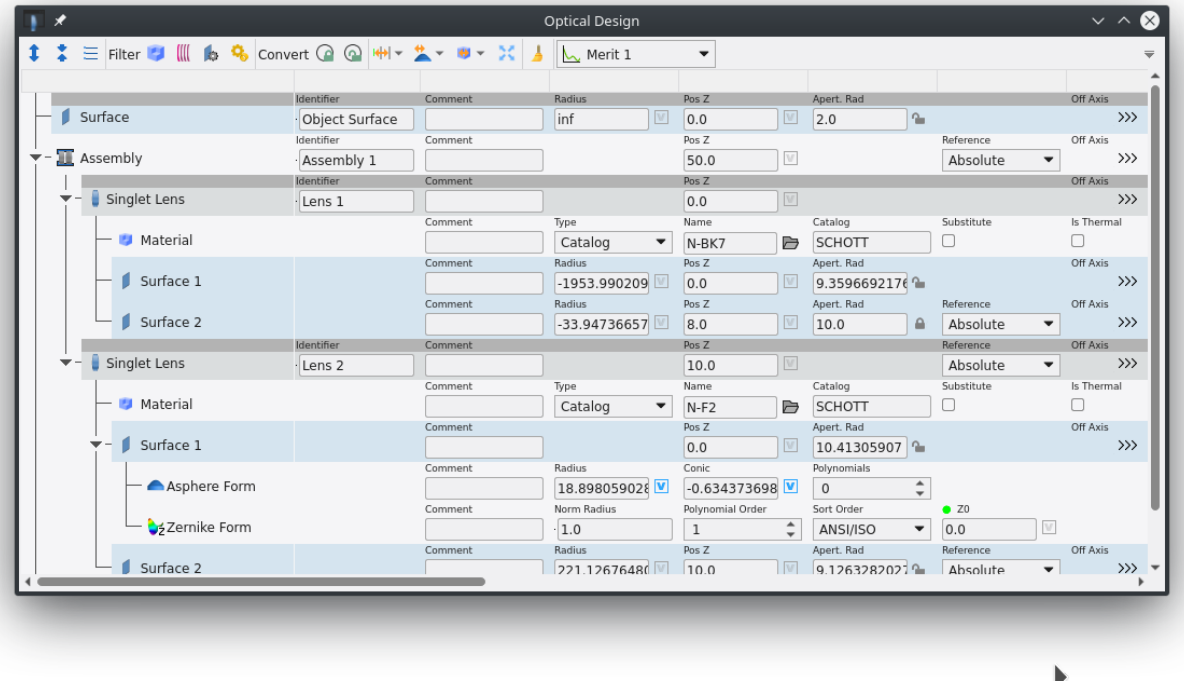


Figure 3.4: Optical Design Editor.

To unfold the tree structure of the element in order to unveil the child-elements, click with the left mouse button on the ▶ arrow. To retract the tree-structure, click with the left mouse button on the ▼ arrow. Click with the left mouse button on the ↕ icon, in order to unfold the complete tree structure of all elements in the tree view. Click with the left mouse button on the ↕ icon, in order to retract the complete tree structure of all elements in the tree view. Click with the left mouse button on the ☰ respectively ☰ icon, in order to switch between the compact- and the enlarged tree view. In the enlarged tree view, the Header-Information of each cell is written directly above the cell. In the compact tree view, the Header-Information of each cell will be displayed by clicking into the row, at the top of the tree view. The Header-Information describes the type of value for each cell.

The top menu of the *Optical Design Editor* provides filter options for the tree view as well as functions to be performed on the whole model. Following filter options are available:

- **Filter Material**: Filters out any Material out of the tree view.
- **Filter Coating**: Filters out any Coating out of the tree view.
- **Filter Surface Property**: Filters out any surface property such as Form, Phase, Aperture, Coating or Polarization out of the tree view.

-  **Filter Mech**: Filters out any Mech Part out of the tree view.

Following conversion methods are available:

-  **Lock all Apertures**: Lock all floating apertures (see chapter 7.1).
-  **Unlock all Apertures**: Set all apertures to floating, if no apertures via Aperture Property Items are defined (see chapter 7.1).
-  **Convert Position**: Globally convert in between position definitions e.g. convert all positions to be absolute or relative. (See chapter 6.4)
-  **Convert or Import Form**: Convert in between surface form definitions - e.g. convert in between different types of asphere definitions or polynomials - or fit a surface form to imported data.
-  **Convert Material**: Convert in between different material definitions or replace materials models with the best-fit catalog materials.
-  **Scale System**: Scale the whole system by a factor.
-  **Clean System**: Remove unnecessary surface properties from the model.

3.4.2 System Setup Editor

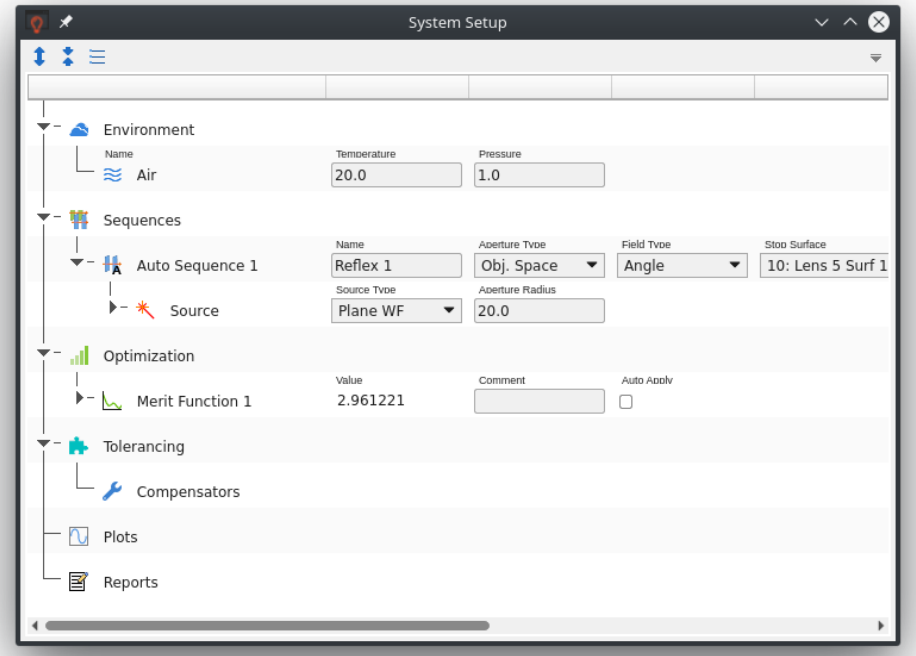


Figure 3.5: System Setup Editor.

The *System Setup Editor* holds several categories (Environment, Sequences, Optimization, Tolerancing, Plots, Reports) of items that are grouped in the top level container items. For the navigation and editing in the tree view see Editor operations (Mouse, Keyboard) see chapter 3.4.3.

To unfold the tree structure of the element in order to unveil the child-elements, click with the left mouse button on the ▶ arrow. To retract the tree-structure, click with the left mouse button on the ▼ arrow. Click with the left mouse button on the ↕ icon, in order to unfold the complete tree structure of all elements in the tree view. Click with the left mouse button on the ↕ icon, in order to retract the complete tree structure of all elements in the tree view. Click with the left mouse button on the ≣ respectively ☰ icon, in order to switch between the compact- and the enlarged tree view. In the enlarged tree view, the Header-Information of each cell is written directly above the cell. In the compact tree view, the Header-Information of each cell will be displayed by clicking into the row, at the top of the tree view. The Header-Information describes the type of value for each cell.

The *System Setup Editor* holds the following categories:

3.4.2.1 Environment

The Environment defines the following environmental conditions:

- *Temperature*: Set the temperature of the environment in degree celsius.
- *Pressure*: Set the pressure of the environment in bar.

The environment *Temperature* and *Pressure* will be applied to any lens in the optical system if it is not marked as thermal and has an own temperature (see chapter 8). When a Sequence is defined over a range of Surfaces the Material in between any of two following Surfaces, that do not belong to the same Element or Lens is assumed to be Air at the temperature and pressured defined by the environment. This is also true for Surfaces that are located at the exact same position. To model two elements that are kitted without an air-gap the Doublet or any other n-let lens has to be used.

3.4.2.2 Sequences

The Sequences Container object holds any Sequential Ray Trace defined for the model. As default, an Auto Sequence is added in every new file. To add new Sequences, click with the right mouse button on Sequences and select "New Auto Sequence" or "New User Sequence" or use the *Sequence Tab* to add or edit a Sequence. It is possible to add an arbitrary number of Sequences. For further information, see chapter 14.1. Adding additional Fields, Wavelengths or Surfaces to the Sequential Ray Trace can also either be done via the right click menu on the container items in the Sequence or via the *Sequence Tab*. To insert / delete Fields, Wavelengths or Surfaces also the 'Insert' / 'Delete' keys can be used when the container is selected. For adding common Wavelengths, the right click menu on the Wavelengths Container provides a set of common used Wavelengths as well as groups of Wavelengths.

3.4.2.3 Optimization

The Optimization Container contains all Merit Functions that have been defined by the user. To add a new Merit Function right click on the Optimization container and select "Add Merit Function". Another option for adding or editing Merit Functions is via the *Optimization Tab* in the main ribbon bar. Additional Ray Traces for a Merit Function Ray Trace Target as well as Global Targets can also be added either via the right click menu on the Optimization Goal Container or Constraint Container, and by using the *Optimization Tab*, by selecting the container where to add the goal or constraint to. Depending on the Container added to, the target will either behave as a Optimization Goal or as a Constraint (see chapter 18.2). To add a target as Constraint, the Constraint Container

needs to be selected first and then the target can be added. Another option is to directly use the right click menu on the Container. If the Constraint Container is not selected, the default action is to add the target as Goal.

Note that Ray Trace Targets can only be added to the Containers within a Ray Trace, whereas Global Targets can only be added to the Global Containers. In the category of Ray Trace Targets any Targets are placed, that are computed from the ray data like e.g. wavefront errors. Whereas in the category of global Global Targets any Targets are placed, that are computed - without ray data - from the optical system like e.g. lens thicknesses.

3.4.2.4  Tolerancing

The Tolerancing Container stores any Tolerances defined for the optical system as well as the Compensators. New Tolerances can be added by the right click menu on the Tolerance Containers or via the *Tolerancing Tab* in the main ribbon bar.

3.4.2.5  Plots

The Plots Container shows any Plots that are opened for a model. New Plots can be added via the right click menu or via the *Analysis Tab*. By double clicking a Plot listed in the container, it can be toggled in between visible and not visible. If a Plot is invisible it will be grayed out in the *System Setup Editor* and it will not use any CPU. The settings for the Plot will be saved, no matter if it is invisible.

3.4.2.6  Reports

The Reports Container shows any Reports that are opened for a model. New Reports can be added via the right click menu or via the *Analysis Tab*. By double clicking a Report listed in the container, it can be toggled in between visible and not visible. If a Report is invisible it will be grayed out in the *System Setup Editor* and it will not use any CPU. The settings for the Report will be saved, no matter if it is invisible.

3.4.3 Editor operations (Mouse, Keyboard)

Mouse Pointer:

- **Left Mouse Button:** By clicking with the left mouse button..:
  ..into an Editor-Cell, the cursor in that cell starts blinking and is ready for editing and entering data.
  ..on the arrow of the tree view, the tree view will unfold/close.
  ..on a checkbox, the checkbox will be checked or unchecked.
  ..on a drop-down menu, the menu will open.
  ..on the edge of a column, the icon will appear. Resize the column by draging with pressed left mouse button.
- **Right Mouse Button:** By clicking with the right mouse button on an element, a drop-down menu will open, and all the available actions for the specific element are shown.
- **Drag & Drop:** With pressed left mouse button, drag Basic Elements and drop them to other positions or drag them into other elements to create a child-element.

Keyboard:

- F2 **F2:** If a cell is focussed, press F2, the cursor in that cell starts blinking and is ready for editing and entering data.
- **Up:** Press the Arrow Up Button on the keyboard, to jump on the same level of the tree view one item up.
- **Down:** Press the Arrow Down Button on the keyboard, to jump on the same level of the tree view one item down.
- **Right:** Press the Arrorw Right Button on the keyboard, to jump on the same level of the tree view one item to the right.
- **Left:** Press the Left Key on the keyboard, to jump on the same level of the tree view one item to the left.
- Page up **Page Up:** Press the Page Up Button on the keyboard, to jump one level in the tree view up.

- Page down **Page Down:** Press the Page Down Button on the keyboard, to jump one level in the tree view down.
- **Tab:** Press the Tab Button on the keyboard, to jump on the same level of the tree view one item to the right.

3.5 Lookup Tables

The Lookup Tables in Quadoa provide the capability of realizing zoom systems, parametric models or any other systems that have more than one state or are not static. Two basic types of lookups exist: The Multi Config Lookup Table for defining multiple discrete states, as well as the Slider Interface for continuous ranges. Which one to use for which problem is discussed in chapter 16. The basic lookup mechanism is shown in fig. 3.6.

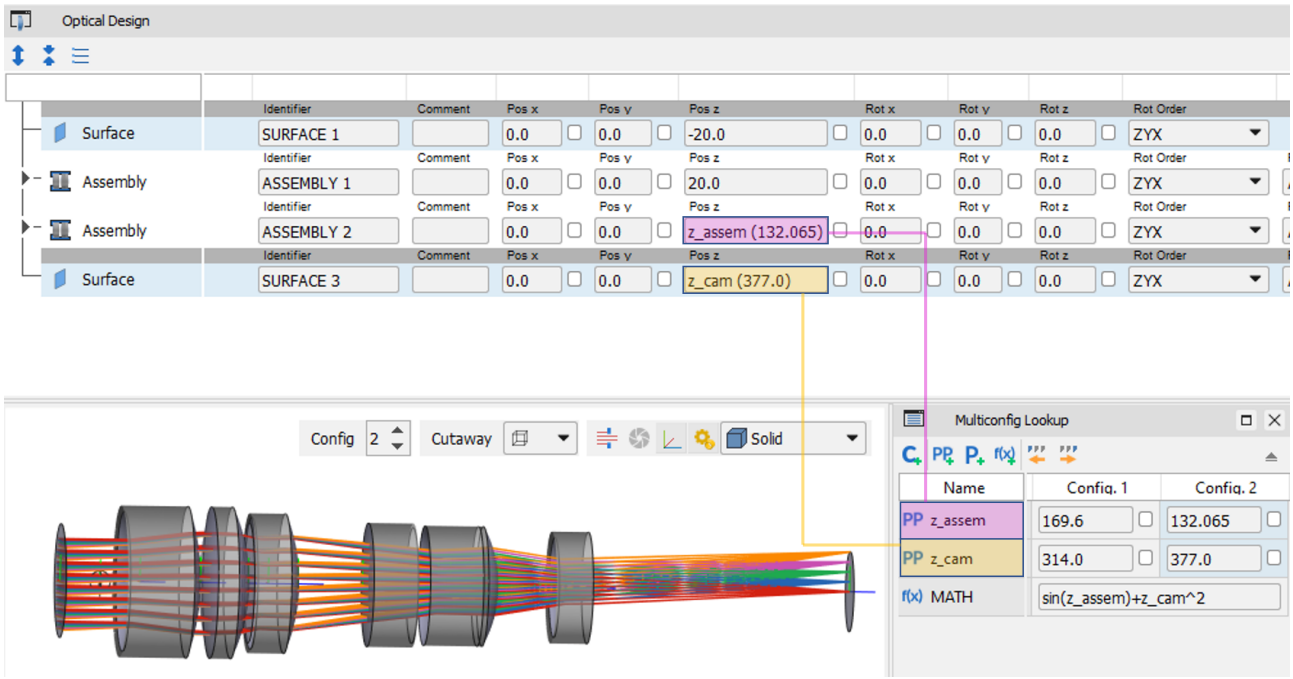


Figure 3.6: Example of a Zoom Lens System with Multiconfiguration. Quadoa Example Filepath: Quadoa/examples/Objectives/Zoom_Lens.optx

Each parameter in the lookup table holds a *Name* that is specified in the first column. To use the lookup parameter in e.g. in the *Optical Design Editor*, the value of the parameter is simply replaced by the *Name* of the lookup parameter. If the specified name is present in one of the lookup tables, the numeric value will be looked up from there. If it is not defined, an error message will appear and the item that holds the parameter will be highlighted in red.

3.5.1 Multiconfig Lookup Table

The Multi Config Lookup Table is used to supply a number of discrete states. The most common use for this feature is for the design of zoom lenses or scanning mirror systems. (see chapter 16.1 for the concept of the Multi Config Lookup Table). The current state of the configuration will be directly applied to the model that is visible in the main 3D-View. To show more than one configurations at once the 3D-View Window is used. If a parameter is set as lookup parameter, the Variable icon is no longer available in the *Optical Design Editor*. In this case the activation state for the Variable can be set independently for each configuration via the icon inside the Multi Config Lookup Table.

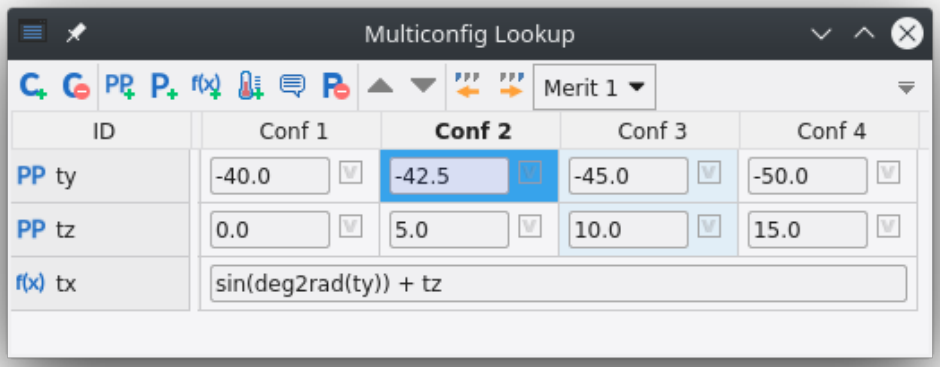


Figure 3.7: Multiconf

The main functionality can be accessed by the menu bar on top of the *Multiconfig Lookup Table*.

- **Add Config:** Click on the icon, to add a new Configuration to the Multi Config Lookup Table.
- **Delete Config:** Click on the icon, to delete the selected config from the table.
- **Add Multiconfig Parameter:** Click on the icon, to add a new Multiconfig Parameter to the Multi Config Lookup Table.
- **Add Parameter:** Click on the icon, to add a new Parameter to the Multi Config Lookup Table.
- **Add Math Expression:** Click on the icon, to add a new Math Expression to the Multi Config Lookup Table.

-  **Add Thermal Expansion Lenght:** Click on the icon, to add a new Thermal Expansion Lenght to the Multi Config Lookup Table.
-  **Add Comment:** Adds a free-text comment row to the Multi Config Lookup Table.
-  **Delete Multiconfig Parameter:** Click on the icon, to delete the selected multiconfig parameter.
-  **Move Up:** Moves the selected row up by one row.
-  **Move Down:** Moves the selected row down by one row.
-  **Switch to next Config:** Click on the icon, to switch to the next Config.
-  **Switch to previous Config:** Click on the icon, to switch to the previous Config.

Switching the configuration is also possible by clicking the header of the column to switch to or via the spinbox in the right corner of the 3D View. The combobox on top is used to specify for which Merit Function the Variable activation state is edited. The right click menu can be used as well to insert or delete configurations and parameters.

Note that any Analysis Plots will show the plot for the current configuration as default. However the behavior can be changed by selecting a specific configuration instead of "Current" in the plot settings.

3.5.2 Slider Interface

The *Slider Interface* provides the possibility to continuously change one or several parameters within the optical system using sliders. Furthermore, analytic expressions can be derived from the values of the sliders.

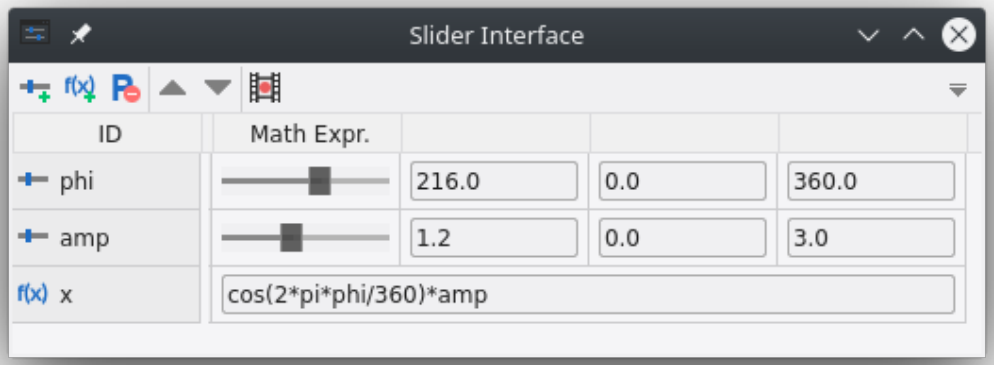


Figure 3.8: Slider Interface

The main functionality can be accessed by the menu bar on top of the *Slider Lookup Table*.

- **Add Slider:** Click on the icon, to add a new Slider to the Slider Interface.
- **Add Math Expression:** Click on the icon, to add a new Math Expression to the Slider Interface.
- **Delete Multiconfig Parameter:** Click on the icon, to delete the selected Slider or Parameter.
- **Move Up:** Moves the selected row up by one row.
- **Move Down:** Moves the selected row down by one row.
- **Record Video:** Record a video of the GUI or a single plot while moving the slider and/or moving/rotation the model in the *3D-View*.
- **Move Slider:** Move the handle of the slider, to slide between the minimum and the maximum value of the parameter.

The *Slider Interface* is only intended to be used when working from the Quadoa GUI. It is not possible to control the sliders over the Scripting Interfaces. The values can be changed by simply setting new values to the parameters.

3.6 Wizards & Dialogs

3.6.1 File

File dialogs are used for the import and export of various types of data.

3.6.1.1 Export Optical Design:

The *Export Optical Design Dialog* is used to export a Sequence to a third party optical design file. It can also be exported to a CSV file, which can be read with a text editor or any spread sheet software. Since most optical design software only supports one sequence per model, not the whole model can be exported at once, but only a single sequence per file. To export a whole multi sequential Quadoa model therefore multiple exports are necessary.

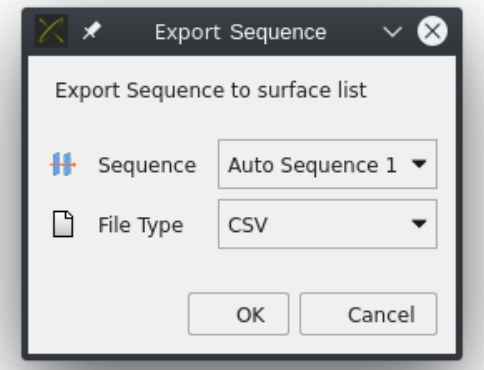


Figure 3.9: Export Optical Design

Note that the export only supports basic surface types and only on-axis systems. For off-axis systems it is recommended to export each on-axis sub assembly independently.

3.6.1.2 Export CAD:

The *Export CAD Dialog* is used to export the whole optical system, sub-assemblies or single lenses to common CAD formats that can be imported in mechanical CAD software. The opposite way (from mechanical CAD to Quadao is described in chapter 20). The resolution parameters are only used for meshed formats like STL files. For analytic CAD formats like STEP, spherical surfaces are exported directly. Rotationally symmetric surfaces like aspheres, are exported as rotations of spline curves with the *Radial Resolution* being the number of control points for the spline. Freeform surfaces without any symmetry are exported as 2D-spline. When exporting the whole model, for filetype STEP and IGES it is further possible to export the rays to the CAD file. Several options are available including ray-fans, grids or solids generated from a hull around the rays.

Note that for systems with strong caustic regions the solid beam export may generate some artifacts around the caustic region. Here it is recommended to export the rays directly.

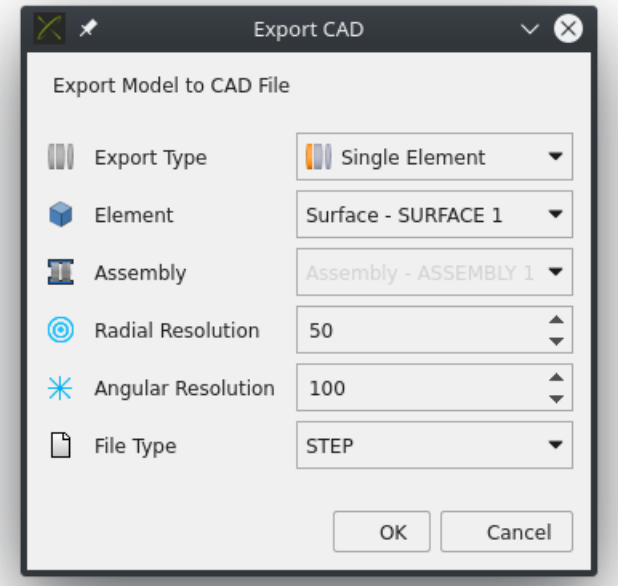


Figure 3.10: Export CAD

3.6.1.3 Export Pointcloud:

The *Export Pointcloud Dialog* is used to export the Sag or Phase Function of a single Surface to a point cloud. The point cloud is saved in a CSV-style text file where the first column is the x-

coordinate, the second column is the y-coordinate and the third column is the z-coordinate in length units. In case of phase exports, the unit is waves. The *Resolution* parameter specifies the number of grid points in each direction. If *Remove Nan Values* is checked, any values that lie outside the Surfaces Aperture or do not have a value, are removed from the list.

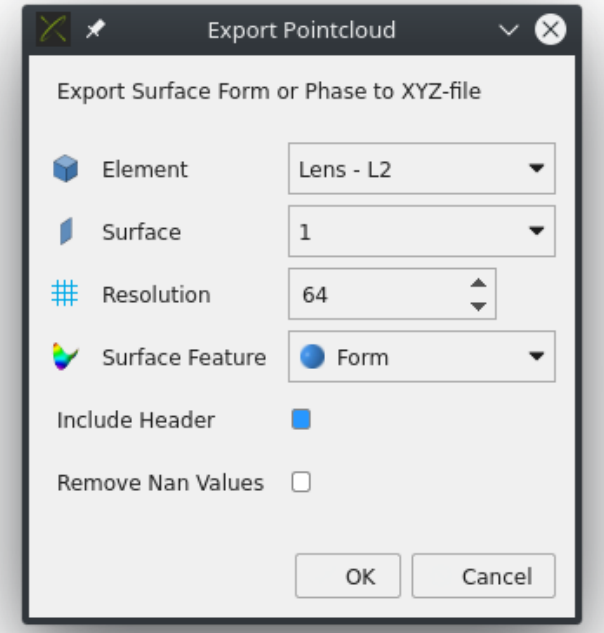


Figure 3.11: Export Pointcloud

3.6.1.4 Technical Drawings ISO 10110:

The *Technical Drawings ISO 10110 Dialog* is used to generate technical drawings of lenses according to ISO 10110, that can be saved as PDF files. To select the Element to generate the drawing for, click on the icon in the menu bar. The table specifying the lens properties and tolerances can be edited by clicking the cells. The content of the whole drawing will be saved together with the model. To save the drawing the icon is used. To reset the Tolerances for the technical drawing click . The drawing can be exported to a PDF file by clicking the icon, or to a SVG-File by clicking the icon.
The tolerances can be saved as default by clicking the icon. Applying default tolerances that have been saved previously is achieved by clicking the icon. To store tolerances in a file click on the icon. To load tolerances from a file click on the icon.

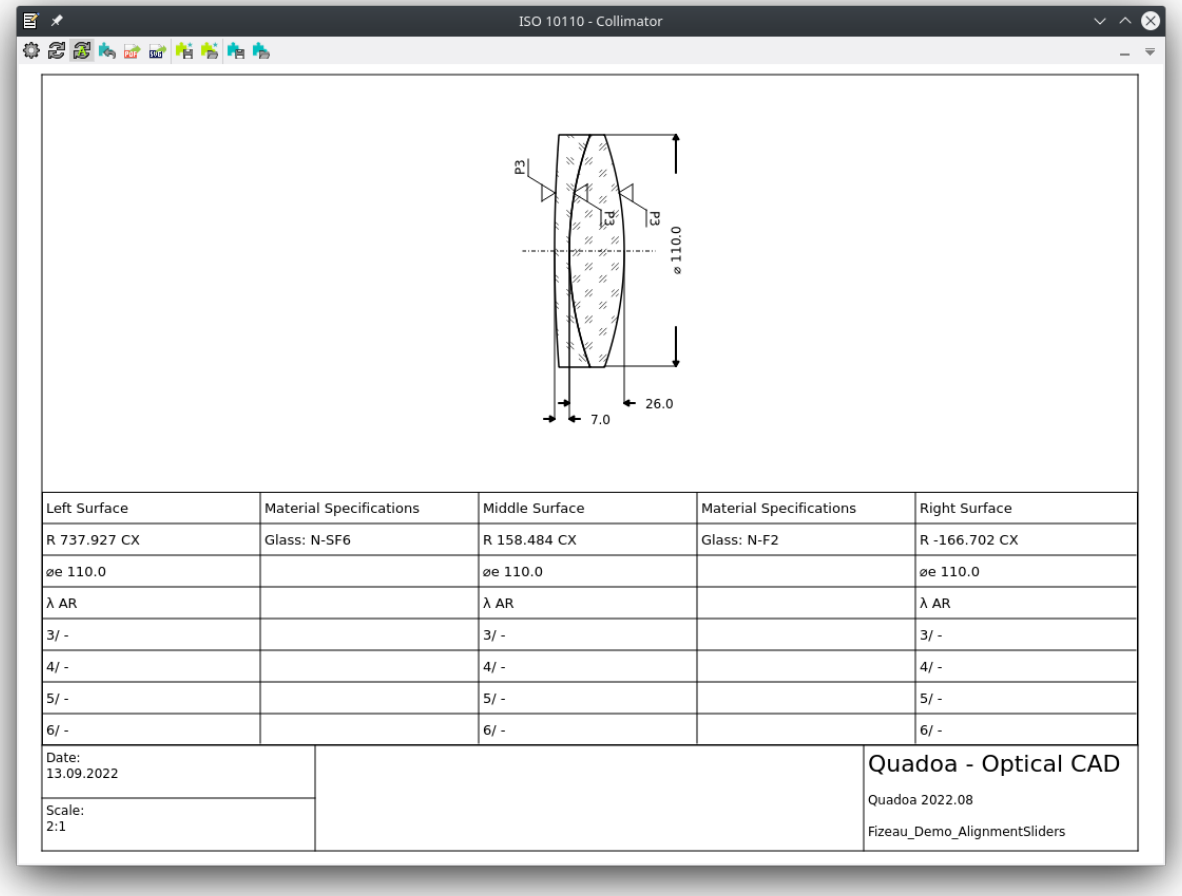


Figure 3.12: ISO 10110 Technical Lens Drawing Wizard.

3.6.2 Setup

3.6.2.1 Material Catalog:

The *Material Catalog Dialog* is used to browse materials. It further provides the feature to filter materials by different criteria like refractive index, Abbe number or name. When the *Material Catalog Dialog* is opened from the material catalog icon of a lens material inside the *Optical Design Editor* or from within a *Lens Wizard*, the selected material can directly be applied via the "Insert Material" button.

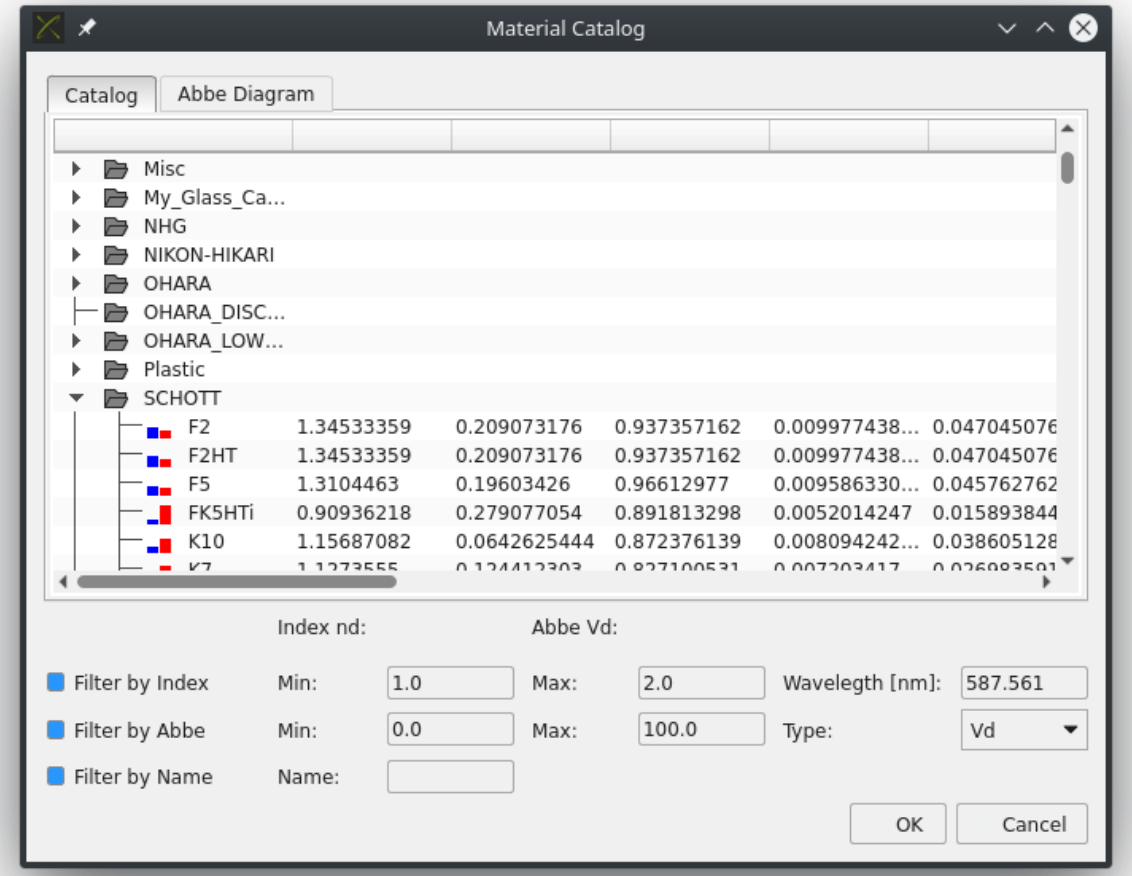


Figure 3.13: Materials Catalog - Catalog View

The material catalog can either be displayed as a tree view, where the refractive index and Abbe number are indicated in a small bar chart directly in the view. Another visualization is via the Abbe diagram. To view the parameters of a material in the Abbe diagram, the chart can be clicked.

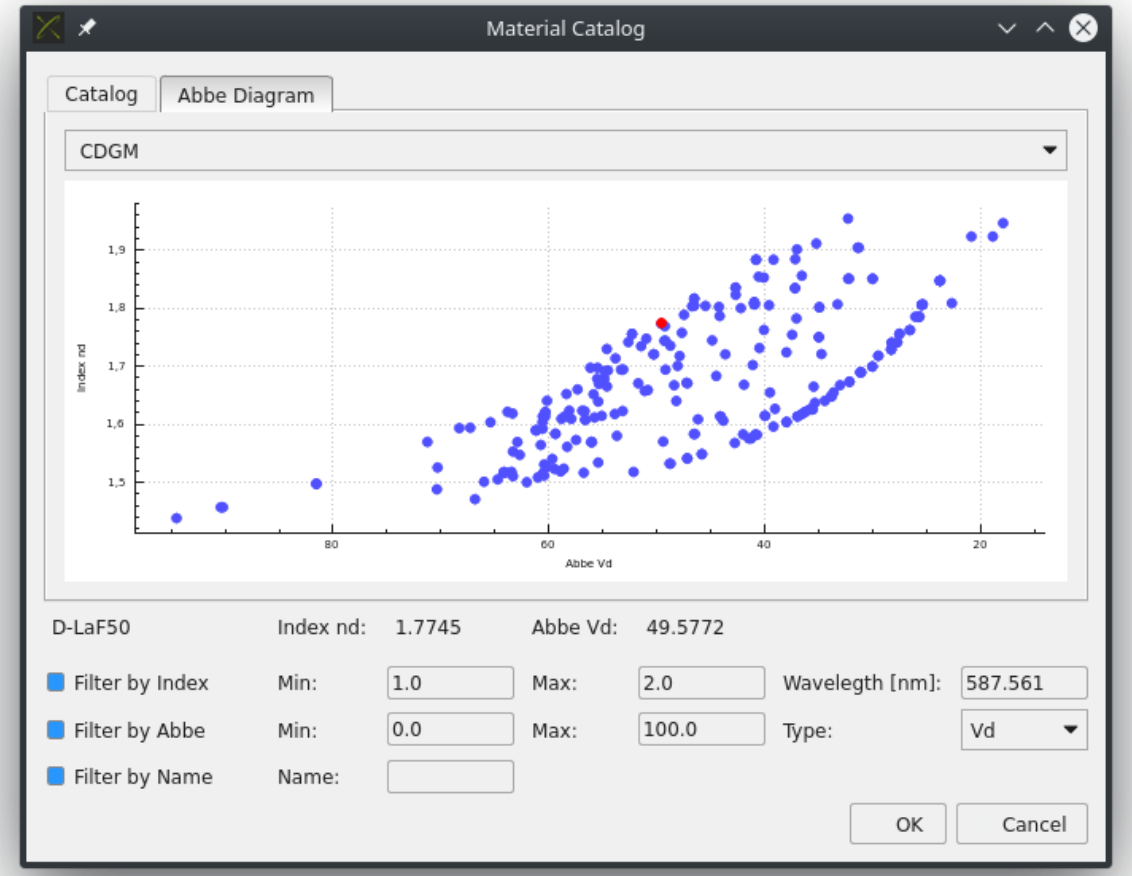


Figure 3.14: Materials Catalog - Abbe Diagram

The *Coating Material Catalog Dialog* is used to browse the materials that are used with Thin Film Coatings

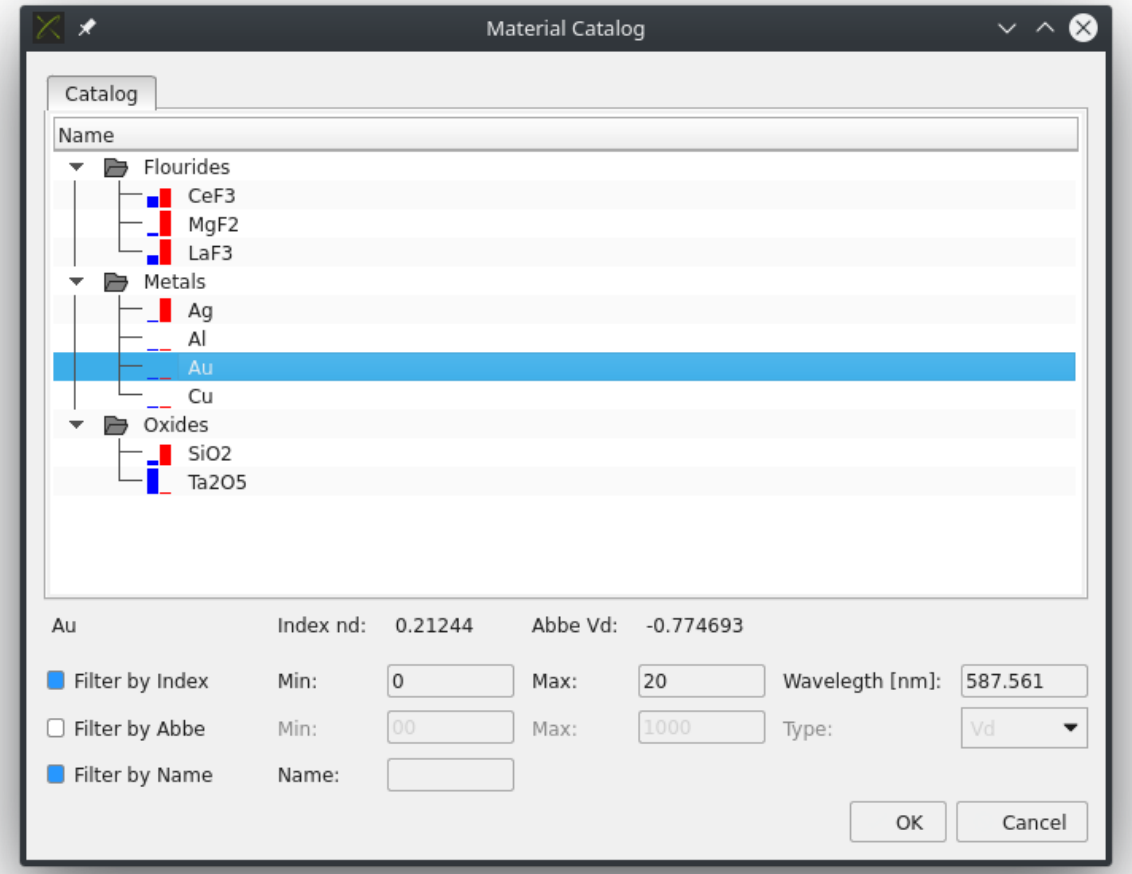


Figure 3.15: Materials Catalog.

3.6.2.2 Lens Catalog:

The Quadoa Installation comes with lens catalogs, that contain a wide range of stock lenses from different manufacturers. The catalogs can be browsed in a tree view. For easier discovery of lenses that are suitable for a specific application, the catalog can be filtered by different categories like *Focal Length*, *Aperture Radius* as well as different categories like signets / kitted, sphere / asphere or basic type. If a Lens is selected, it will be shown in the 3D-View of the catalog. By clicking the "Insert Lens" button, the lens will be inserted to the Optical Design Editor. If an Assembly is currently selected, it will be inserted inside the Local Coordinate System of the Assembly, otherwise it will be added in the Global Coordinate System.

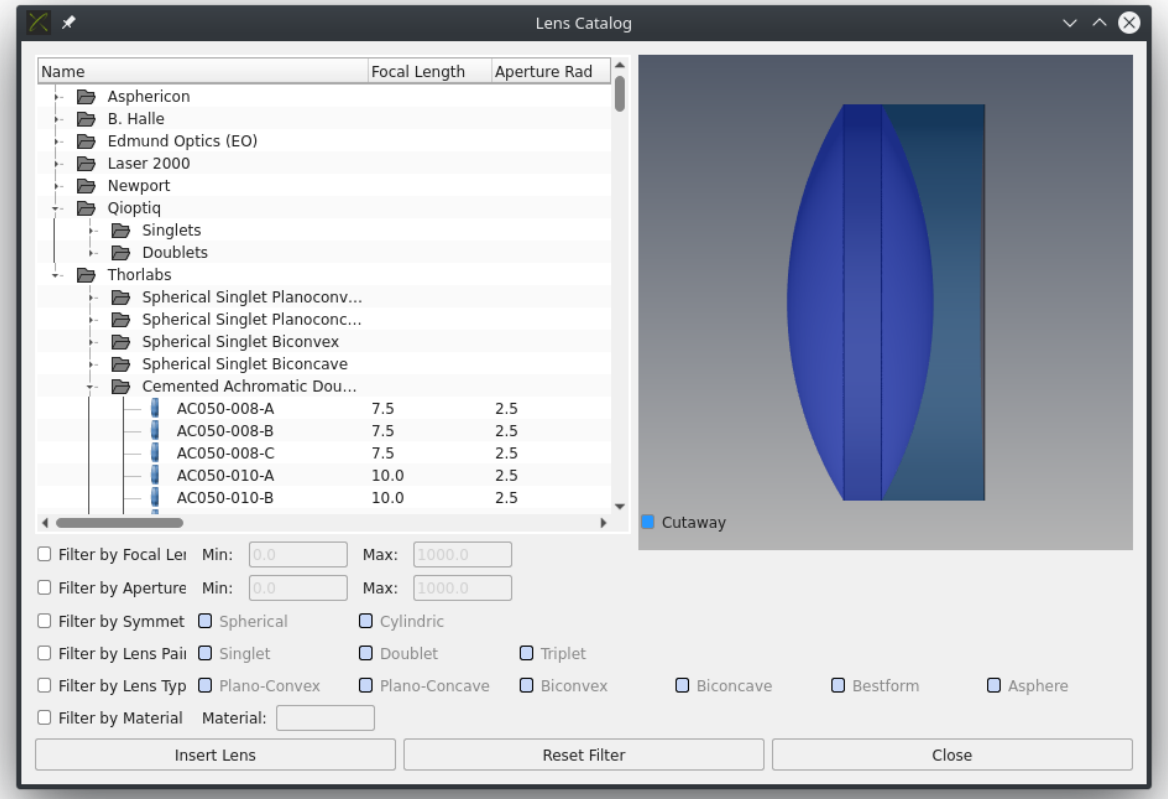


Figure 3.16: Lenses Catalog.

3.6.2.3 Python Script Wizard:

The *Python Script Wizard* is used to create Python scripts for a model from predefined templates. See chapter 24.1.3.

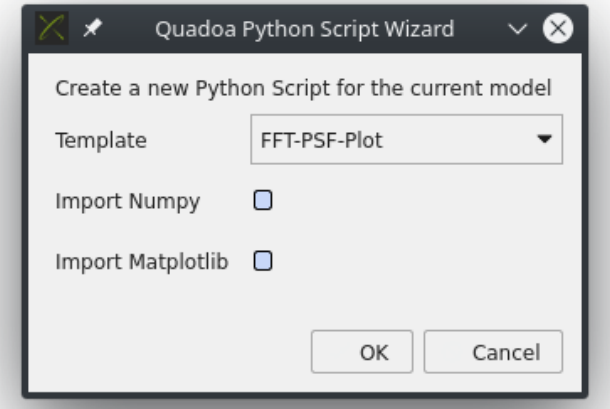


Figure 3.17: Python Script Wizard

3.6.2.4 Matlab Script Wizard:

The *Matlab Script Wizard* is used to create Matlab scripts for a model from predefined templates. See chapter 24.2.3.

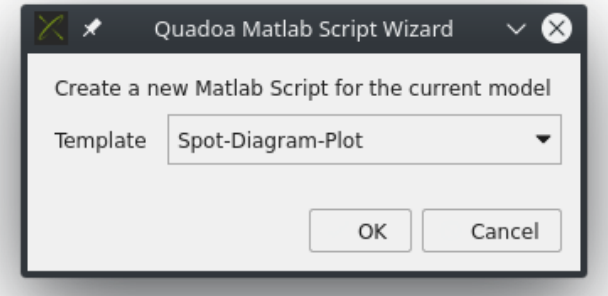


Figure 3.18: Matlab Script Wizard

3.6.3 Construction

3.6.3.1 Singlet Lens Wizard:

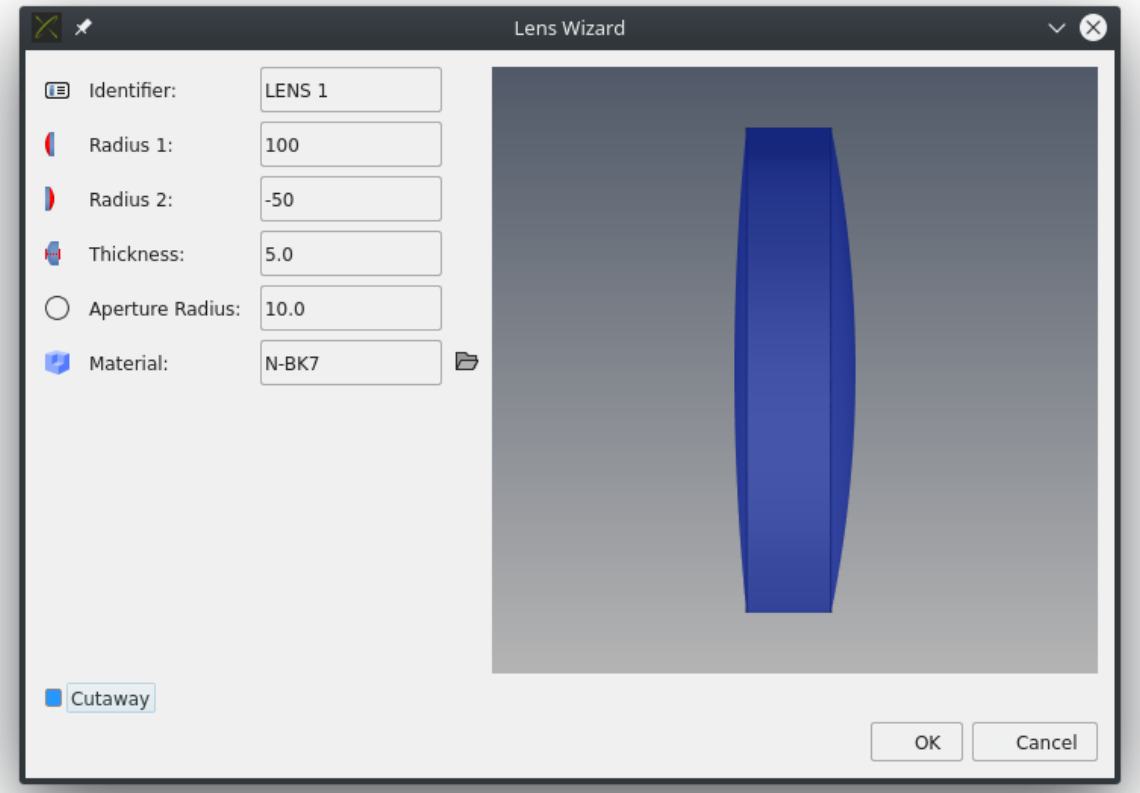


Figure 3.19: Singlet Lens Lens Wizard to create standard lenses.

The *Singlet Lens Wizard* creates a new Singlet Lens. The parameters for the lens are defined as follows:

- **Identifier**: Insert a user defined name.
- **Radius 1**: Radius of the first (left) surface of the Singlet Lens.
- **Radius 2**: Radius of the second (right) surface of the Singlet Lens.
- **Thickness**: Thickness of the lens.

- **Aperture Radius**: Aperture Radius of the lens.
- **Material 1**: Material of the lens. Click on the Catalog Icon to open the Materials Catalog.

Click 'OK' to insert the lens in the Optical Design Editor. If an Assembly is currently selected it will be inserted inside the local coordinate system of the Assembly, otherwise it will be added in the Global Coordinate System.

3.6.3.2 Doublet Lens Wizard:

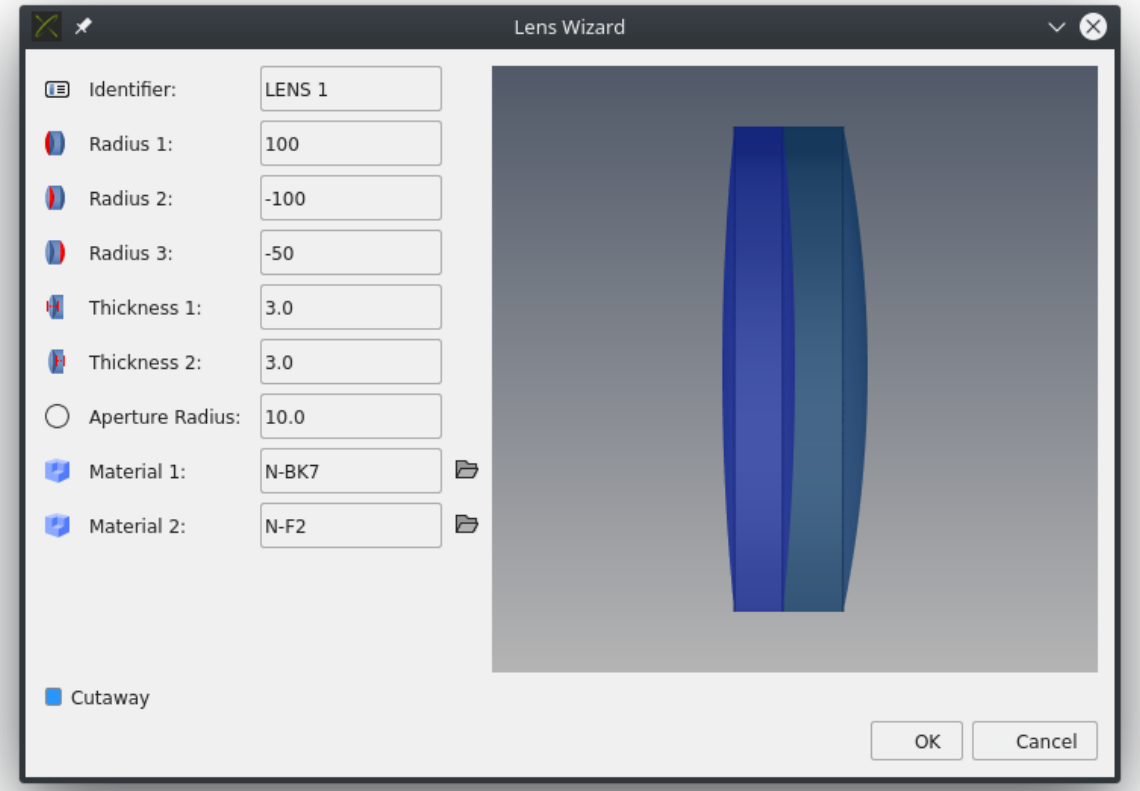


Figure 3.20: Doublet Lens Lens Wizard to create standard lenses.

The *Doublet Lens Wizard* creates a new Doublet Lens. The parameters for the lens are defined as follows:

- **Identifier**: Insert a user defined name.

- **Radius 1**: Radius of the first (left) surface of the Doublet Lens.
- **Radius 2**: Radius of the second (middle) surface of the Doublet Lens.
- **Radius 3**: Radius of the third (right) surface of the Doublet Lens.
- **Thickness 1**: Thickness of the first material of the lens.
- **Thickness 2**: Thickness of the second material of the lens.
- **Aperture Radius**: Aperture Radius of the lens.
- **Material**: Material of the first part of the lens. Click on the 🗁 Catalog Icon to open the Materials Catalog.
- **Material 2**: Material of the second part of the lens. Click on the 🗁 Catalog Icon to open the Materials Catalog.

Click "OK" to insert the lens in the Optical Design Editor. If an Assembly is currently selected, it will be inserted inside the Local Coordinate System of the Assembly, otherwise it will be added in the Global Coordinate System.

3.6.3.3 Triplet Lens Wizard:

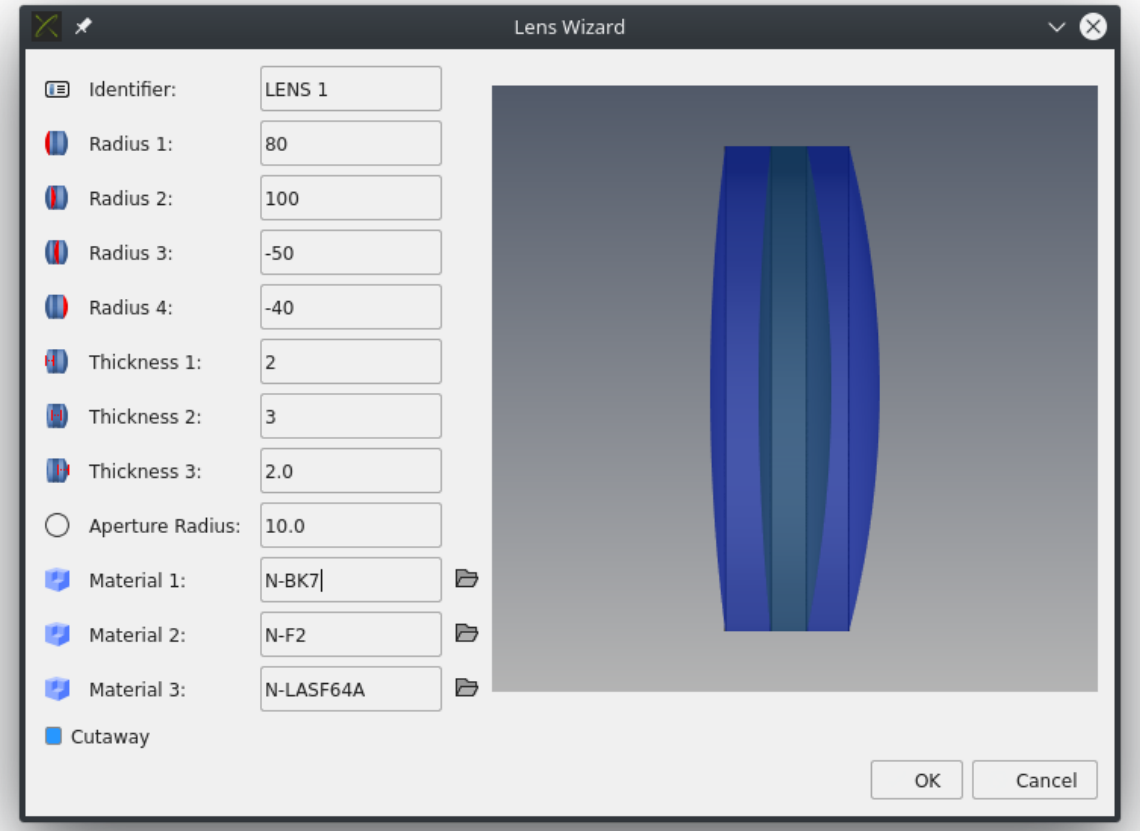


Figure 3.21: Triplet Lens Lens Wizard to create standard lenses.

The *Triplet Lens Wizard* creates a new Triplet Lens. The parameters for the lens are defined as follows:

- **Identifier**: Insert a user defined name.
- **Radius 1**: Radius of the first (left) surface of the Triplet Lens.
- **Radius 2**: Radius of the second surface of the Triplet Lens.
- **Radius 3**: Radius of the third surface of the Triplet Lens.
- **Radius 4**: Radius of the fourth (right) surface of the Triplet Lens.

- **Thickness 1**: Thickness of the second material of the lens.
- **Thickness 2**: Thickness of the second material of the lens.
- **Thickness 3**: Thickness of the third material of the lens.
- **Aperture Radius**: Aperture Radius of the lens.
- **Material 1**: Material of the first part of the lens. Click on the Catalog Icon to open the Materials Catalog.
- **Material 2**: Material of the second part of the lens. Click on the Catalog Icon to open the Materials Catalog.
- **Material 3**: Material of the third part of the lens. Click on the Catalog Icon to open the Materials Catalog.

Click "OK" to insert the lens in the Optical Design Editor. If an Assembly is currently selected it will be inserted inside the Local Coordinate System of the Assembly, otherwise it will be added in the Global Coordinate System.

3.6.4 Sequence

3.6.4.1 Ghost Wizard:

The *Ghost Wizard Dialog* is used to generate a set of Sequences - One for each ghost reflection path in the optical system.

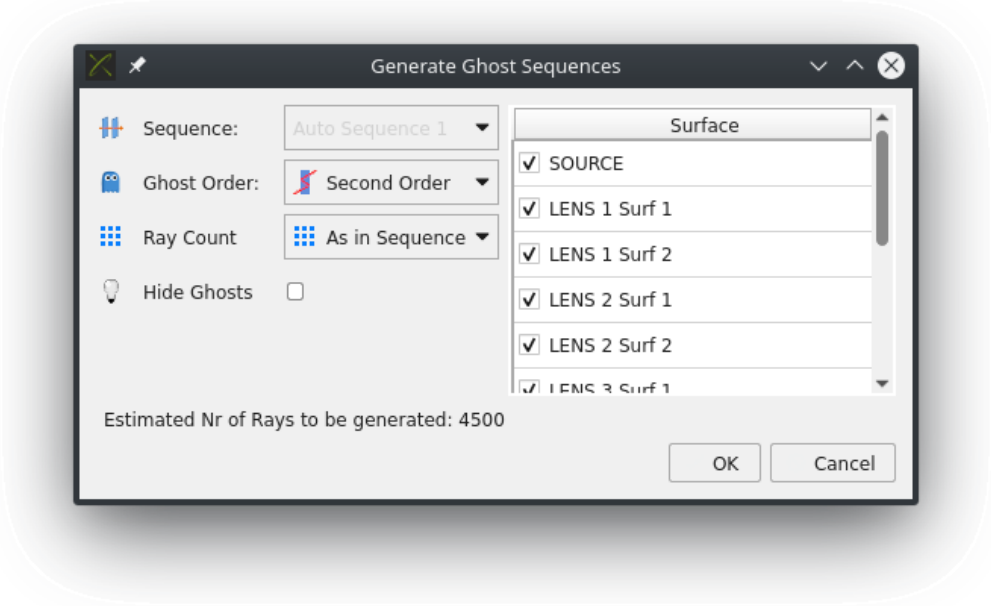


Figure 3.22: Ghost Sequence Wizard to generate Ghost Reflexes.

To generate ghost sequences, in the *Ghost Wizard Dialog*, select the Sequence for which to generate Ghost Reflexes from the drop-down menu. From the *Ghost Order* drop-down menu select the order of ghosts. You can choose between "First Order" and "Second Order". "First Order" will generate ghosts with a single-reflection on one of the Sequence surfaces. "Second Order" will generate ghosts with two reflections on surfaces of the Sequences. Selecting the *Hide Ghost* checkbox, will generate the ghosts without showing them in the *3D-View*. This is recommended when generating a large number of ghost (see chapter 4 on GUI Performance). Via the right click menu or the *Sequence Tab*, the visibility can be toggled afterwards with the Show and Hide buttons. Another option to keep the ray count low is to select "Minimal" on the *Ray Count* parameter. This will generate ghosts with a minimal XY-fan Distribution.

Note that the Ray Count as for any Sequence only affects the rays shown in the 3D-View. The Distribution and number of rays used for ghost analysis plots or any other analysis feature is chosen independently in the particular feature.

Click "OK" to generate the ghost reflex Sequences. For every single ghost reflex, a single ghost Sequence will be generated in the Sequences List in the System Setup Editor. The ghost rays - if not hidden - will be shown in the *3D-View*.

3.6.4.2 Diffractive Ghost Wizard:

The *Diffractive Ghost Wizard Dialog* is used to generate a set of Sequences for systems, that have at least one diffractive Surface with at least one Phase Property Item assigned to it.

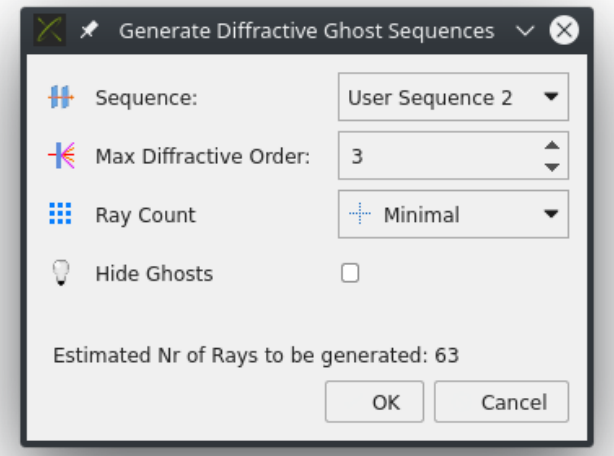


Figure 3.23: Diffractive Ghost Sequence Wizard to generate Diffractive Ghost Reflexes.

The ghosts are defined as the diffraction orders of the diffractive element, that are not the design order. To generate diffractive ghosts, the Sequence for which to generate them is chosen in the wizard. The *Maximum Diffractive Order* defines the maximum order of diffraction. This means, that if e.g. +1 is the design order for the diffractive element following orders will be generated: -3, -2, -1, 0 , +2, +3. Note that +1 is not included, since it is the design order. If the diffractive element is double pass - like e.g. when testing aspheres interferometrically with a CGH -, or if there are more than one diffractive element in the Sequence, any combination of diffractive orders that is not the design order will be generated.

Selecting the *Hide Ghost* checkbox, will generate the ghosts without showing them in the *3D-View*. This is recommended when generating a large number of ghost (see chapter 4 on GUI performance). Via the right click menu or the *Sequence Tab* Show and Hide buttons, the visibility can be toggled afterwards. Another option to keep the ray count low is to select "Minimal" on the *Ray Count* parameter. This will generate ghosts with a minimal XY-fan Distribution.

Note that the Ray Count as for any Sequence only affects the rays shown in the 3D-View. The Distribution and number of rays used for ghost analysis plots or any other analysis feature is chosen independently in the particular feature.

Click "OK" to generate the diffractive ghost Sequences. For every single diffractive ghost, a single ghost Sequence will be generated in the Sequences List in the System Setup Editor. The ghost rays - if not hidden - will be shown in the *3D-View*.

3.6.5 Optimization

3.6.5.1 Optimization Dialog:

The *Optimizer Dialog* is used in combination with any of the available Optimizer Types (see chapter 18.7).

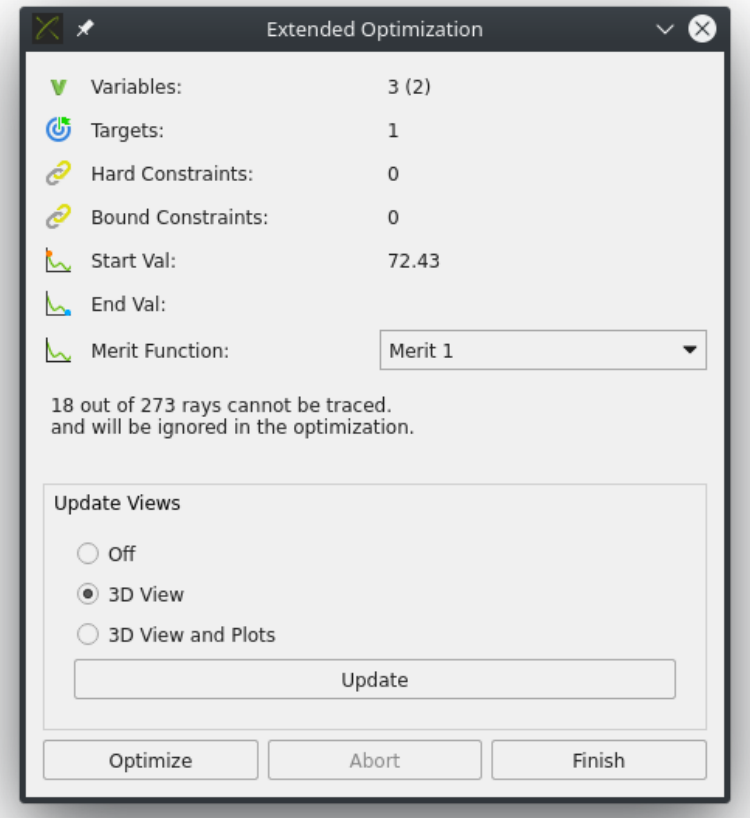


Figure 3.24: Optimizer Dialog

To start the optimization, the Merit Function is chosen via the *Merit Function* parameter. The optimization will either stop when finished (Local Optimizer), or run until the user stops it. The update behavior can be chosen via the settings in the *Update Views* field.

3.6.5.2 Move to Focus Dialog:

The *Move to Focus Dialog* allows the fast adjustment of the position of the last Surface in a Sequence in order to move it in the focus. If the optical axis is not the Z-Axis, the axis to move the Surface along can be chosen from by the *Direction* parameter. The movement will always be applied in the local coordinate system of the last Surface. As criterion for the focus position either the spot radius or the wavefront error can be chosen.

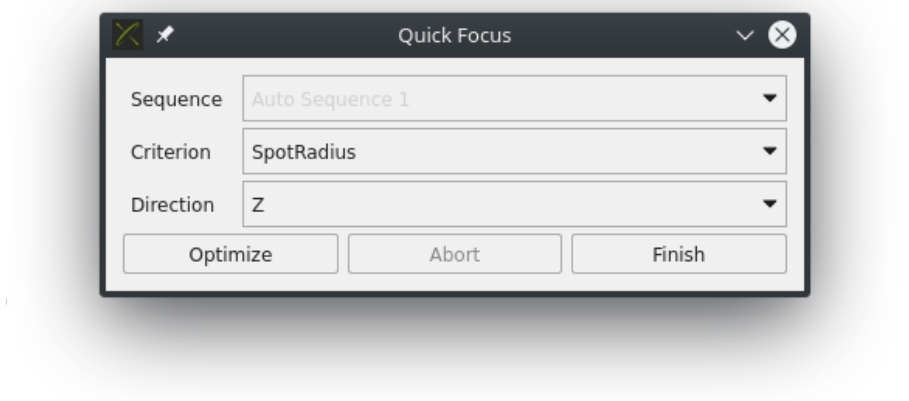


Figure 3.25: Move to Focus Dialog

3.6.6 Tolerancing

3.6.6.1 Default Tolerances Wizard:

The *Default Tolerances Wizard* is used for generating the tolerances for a model, that will be applied as Global Tolerances.

Figure 3.26: Tolerances

The "Save" and "Open" buttons allow to store common used tolerance settings in a file. The "Save as Default" button will save the settings of the dialog for a Quadoa installation, in order to load immediately the saved settings on startup. "Reset" will reset the settings to the default values if they have been changed. "Apply" will generate the Global Tolerance Parameters. If any Global Tolerances have been defined before, a dialog will appear if those should be overwritten. More specific tolerances for e.g. individual decenter parameters in x and y direction or for Specific Elements can be defined via the *Tolerancing Workbench*.

3.6.6.2 Sensitivity Analysis Dialog:

The *Sensitivity Analysis Dialog* is used to generate a Sensitivity Analysis Report. In the report, the effect on any tolerance defined for the system on the criterion defined for tolerancing will be shown. See chapter 19.5 on Sensitivity Analysis.

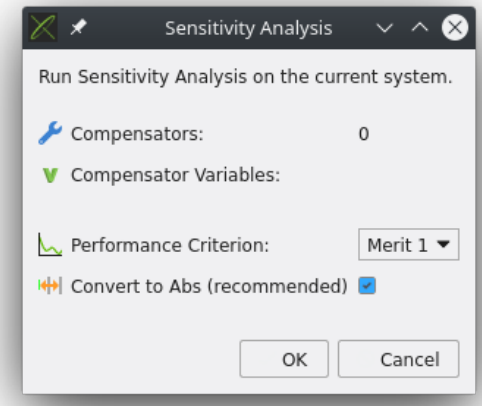


Figure 3.27: Sensitivity Analysis Dialog

3.6.6.3 Monte Carlo Analysis Dialog:

The *Monte Carlo Analysis Dialog* is used to run a Monte Carlo Analysis. It will generate a report about the overall behavior of the generated systems. Furthermore, it can be used to save a large amount of perturbed systems for further analysis. See chapter 19.4 on Monte Carlo Analysis.

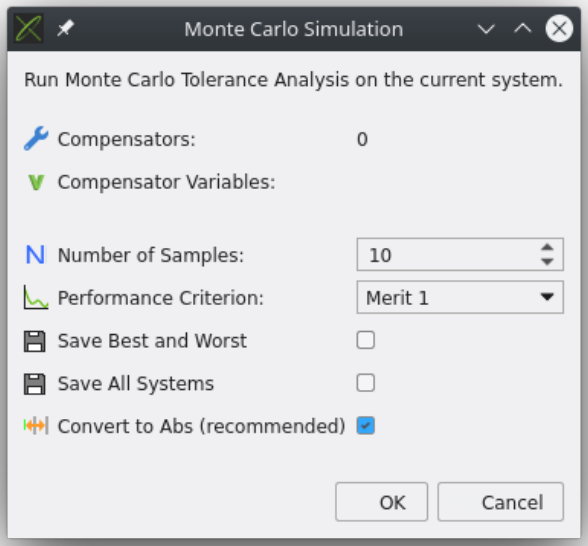


Figure 3.28: Monte Carlo Analysis Dialog

4 GUI Performance

As soon as the user applies a change to the optical system, by editing a parameter or adding an element, Quadoa will trace the rays that are visible in the 3D View and will update all the plots that are currently open. This allows the user to directly see the effects of the changes on the optical system and gives an intuitive feedback on the changes. However, in some cases this direct update may not be desired. When there are many Sequence with a large number of rays visible in the 3D View, e.g. when performing a ghost analysis, the tracing of the rays may lead to significant delays after each change is applied. Also some plots are computationally intensive and may lead to delays. To avoid unnecessary delays, Quadoa provides several options that can be used to trigger the large computation only when really desired and to keep the computational costs low.

- **Global Auto Update:** The Auto Update button in the Quick-Access-Toolbar allows to globally disable automatic updating of any plots. When the auto update is disabled, plots still can be manually updated by clicking the Update button on each plot. Furthermore, all open plots can be manually updated at once by clicking the Update button in the Quick-Access-Toolbar. Note that plots that have the auto update disabled on plot level will not be updated by this global update.
- **Plot Auto Update:** The Auto Update button on each plot can be disabled to prevent the automatic update of a specific plot. This is especially useful when the user wants to disable the update of computational intensive plots, but keep all other plots up to date. If the auto update of a plot is disabled, the user can trigger a manual update of the plot by clicking the update button in the plots toolbar.
- **Number of Rays in 3D View:** Large numbers of rays in the 3D View may also slow down the interface performance. Since the number of rays for any analysis plot or optimization is totally independent from the number of rays in the 3D View, it is typically not necessary to work with large amounts of rays there. Therefore, it is recommended to keep the number of rays to be displayed in the 3D View low. The distribution of the rays, as well as the total amount, can be set in the System Setup Editor independently for each Sequence. When generating ghost sequences for large systems with many surfaces, the number of rays for each ghost sequence can be reduced directly when generating the ghosts, which is recommended.

- **Hiding Rays:** Any Sequence can be hidden and made visible again via the right click menu in the System Setup Editor. When a Sequence is hidden, the rays will not be traced on the update. The hiding of the Sequence does not affect the analysis plots and plotting data of a hidden Sequence it totally fine. Besides the whole Sequence, individual Fields and Wavelengths can be hidden as well. It is recommended to only set the Sequences, Fields and Wavelengths to visible that you are currently working with. For ghost analysis, all ghost Sequences can be set to hidden (and back to visible again) via the Hide - and Show all Ghosts button in the Sequence Tab. Furthermore, the ghost dialog directly allows to generate the Sequences as hidden, which is recommend for large systems with a large number of surfaces.

5 Definitions

5.1 Style Conventions

For this reference manual the following style conventions are used to highlight keywords in the text.

- Objects: Any Objects that can be added to any of the tree data structures like the Optical Design Editor or the System Setup Editor are colored. By clicking it the hyperlink to the chapter in the manual is followed.
- *Parameter:* Any parameter is printed in italic.
- "Parameter Values" Any parameter value is enquoted.
- *GUI Elements:* Any *GUI Element* or *Dialog:* is colored. By clicking it the hyperlink to the chapter in the manual is followed.

5.2 Coordinate Systems

Every coordinate system in Quadoa is a right handed coordinate system. The Z-axis, which is typically the optical axis, is pointing to the right. The X-Axis is pointing away from the user into the monitor. The Y-axis is pointing upwards in vertical direction. The default coordinate system for plots that plot data over the Surface, plot the data in a coordinate system as when looked at the Surface from +Z direction. The direction of rotation can be determined by the right hand rule. When the thumb is pointing along the positive direction of the axis, then the fingers show the positive direction of rotation.

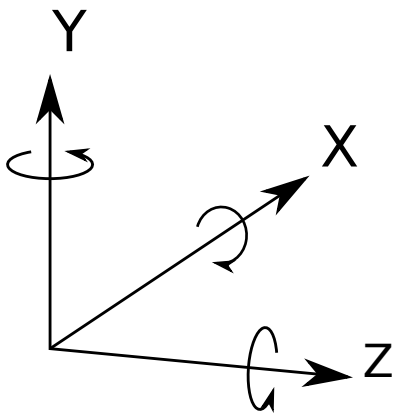


Figure 5.1: Coortdinate System - Arrows show in positive directions

5.3 Sign Conventions

For the radius of curvature a negative radius is, when the surface is curved to the left (in -z direction). A positive radius is, when the surface is curved to the right (in +z direction). In general, positive values for Form objects are in +Z direction.

5.4 Real and Virtual Rays

A ray becomes virtual (or real if it has been virtual before) as soon as its path from one surface to the next shows in the negative direction of the direction it propagated before (sign flip). The initial real/positive direction is in the direction of the positive z-coordinate of the object surface. This means that rays starting in negative z direction will start as virtual rays while rays starting in positive direction will start as real rays. The state of a ray (real or virtual) does not have any influence on the ray-tracing as long as the virtual rays are not banned by the Sequence (see parameter *Ban Virtual*) except that the OPL of the rays will have a negative sign.

6 Coordinate Systems

6.1 Hierarchical Tree Structure

The main data structure, that describes the setup of the optical system in Quadoa, is arranged as a tree like hierarchical structure. In general, each element can have zero to several child items. The Basic Elements are Surfaces, Lenses or Custom Elements as well as Ideal Elements and Assemblies. Furthermore, property items can be used to define the properties of the parent item. For elements and lenses, the material item defines the material the element is made of. In case of kitted lenses, more than one material is used to define the element. For Surfaces a wide range of property items exist. These are used to define the Form (sag), Apertures, Phase function, Coatings and Polarization properties or in general any property of the Surfaces they are added to. Many of these property items can be combined to describe more complex surface properties.

6.2 Position Parameters

Items that share position properties are

- Surfaces
- Lenses (Singlet, Doublet, Triplet, N-let)
- Custom Elements
- Ideal Elements
- Assemblies

Any position item has - as any other item that can be placed in the optical design editor - the following parameters in common:

Column	0	1	2
Name	Object Type	Identifier (optional)	Comment
ID	-	id	comment
Editable	no	yes	yes

- *Object Type* is a non editable parameter that describes the basic type of the item.
- *Identifier* is a unique string ID that is used to refer to the object. The identifier may be set to any string that starts with a lower or upper case character. Surfaces only have an ID of their own, if they are not part of a Lens or any other optical element. If they are part of an parental Element, the ID of the Element in combination with the surface number is used to refer to the Surface. See chapter 7.7 for further information about the *Identifier*.
- *Comment* is a free string parameter that can be used for comments and documentation.

The coordinate system is a right handed coordiante system with the Z-axis pointing along the optical axis, the X-axis into the monitor and the Y-axis to vertically upwards (see chapter 5.2).
The position related parameters can be found in the following tables:

Column	4	6	7
Name	Pos Z	Reference	Off Axis
ID	z	refe	off_axis
Editable	yes	yes	yes

Column	4	8	9	10	11	12
Name	Pos Z	Pos X	Pos Y	Rot X	Rot Y	Rot Z
ID	z	x	y	rx	ry	rz
Editable	yes	yes	yes	yes	yes	yes

Column	13	14
Name	Rot Order	Pos/Rot Order
ID	order	posrot
Editable	yes	yes

Column	15	16	17
Name	Pivot X	Pivot Y	Pivot Z
ID	rot_c_x	rot_c_y	rot_c_z
Editable	yes	yes	yes

- *Pos Z* defines the position of the object in Z-direction in length units (typically along the optical axis) inside the local coordinate system.

- *Reference* defines the position type: Available options are "Absolute" and "Relative". See chapter 6.4
- *Off Axis* This parameter is used to show/hide all position parameters, except *Pos Z*. If any of the position parameters has a value different to zero, *Off Axis* is automatically enabled, and the parameters cannot be hidden.
- *Pos X* defines the position of the object in X-direction in length units (typically a vertical decentering) inside the local coordinate system.
- *Pos Y* defines the position of the object in Y-direction in length units (typically a horizontal decentering) inside the local coordinate system.
- *Rot X* defines the rotation of the object around the X-axis of the local coordinate system in angular units.
- *Rot Y* defines the rotation of the object around the Y-axis of the local coordinate system in angular units.
- *Rot Z* defines the rotation of the object around the Z-axis of the local coordinate system in angular units.
- *Rot Order* defines the order the rotations are applied. Available options are: “XYZ”, “XZY”, “YXZ”, “YZX”, “ZXY”, “ZYX”. See chapter 6.5
- *Pos/Rot Order* defines the order of position (translation) and rotation. Available options are "Pos first", "Rot first" and "Z Pos XY". See chapter 6.6
- *Rot Center X* defines the center of rotation in x-direction relative to the objects origin. See chapter 6.7
- *Rot Center Y* defines the center of rotation in y-direction relative to the objects origin. See chapter 6.7
- *Rot Center Z* defines the center of rotation in z-direction relative to the objects origin. See chapter 6.7

6.3 Global and Local Coordinate Systems

Any object in the tree, that represents an physical object like a Lens, an Optical Element, a Surface or a group of lenses, that are arranged in an Assembly, has its own Local Coordinate System. The

position of any object is always defined in the Local Coordinate System defined by its parent object. If an element has no parent, its position is defined in the Global Coordinate System.

6.4 Absolute and Relative Positions

The position of any object can either be defined in absolute or relative coordinates via the *Reference* parameter. Following options are available:

- "Absolute": If the coordinates are defined as "Absolute", the item is positioned inside the Local Coordinate System of its parent item at the coordinates that are set. If the item is not a child of any other item, this means it is positioned in global coordinate space.
- "Relative To Preceding": If the position is set to "Relative To Preceding", the position of the item is relative to the preceding item in the tree (typically the last surface of the preceding lens). The preceding item is the item, that is next when going up in the Optical Design Editors tree view and does not have to be part of the same Element or Assembly. The first Surface of an Element can not be set to relative. This settings is similar to the relative positioning of surfaces in a classical spreadsheet based optical design software.
- "Relative To Sibling": If the position is set to "Relative To Sibling" then the position it relative to the preceding sibling inside the same Assembly.
- "Absolute (Relative Z)": This settings main purpose is for tolerancing of lenses that are mounted in a fill/stack mount, where the lenses are separated by tuning rings. For this mounting type the thickness and z-position (thickness of the tuning rings) of a lens in the stack influences the position of the next lens in the mount. The position type works as follows: The x and y position as well as the rotation of the item are defined in absolute coordinates as if the reference was set to "Absolute". However the z-position results from the sum of all z-positions of the preceding items inside the same assembly. This ensures that thickness tolerances influence the z-positions of the next lenses without introducing unwanted rotations or lateral translations as it would be the case with relative positions.

6.5 Order of Rotations

The order of rotation is defined by "Rot Order" and can be set in any possible order via the six available options "XYZ", "XZY", "YXZ", "YZX", "ZXY", "ZYX" . The default order is "ZYX".

6.6 Order of Rotation and Translation

The order of rotation and translation is defined by "Pos/Rot Order". It can be set to “Pos first”, “Rot first” and “Z Pos XY”. In the case of “Pos first” the object is first translated to the coordinates given by *Pos X*, *Pos Y* and *Pos Z* and then rotated around the three local axis according to the rotation order defined by "Rot Order" and the angles defined by *Rot X*, *Rot Y* and *Rot Z*. In the case of “Rot first” the object is first rotated. Afterwards the translation is applied in the new rotated local coordinate system. If the parameter is set to “Z Rot XY” the object will first be translated in Z direction (typically along the optical axis) then the object will be rotated in the order defined by *Rot Order*. Afterwards the object will be translated in X- and Y-direction (typically decentered) in the new rotated local coordinate system.

6.7 Center of Rotation

The center of rotation can be defined by the parameters *Rot Center X*, *Rot Center Y* and *Rot Center Z*. The default is (0, 0, 0) which is the origin of the objects own coordinate system. If *Pos/Rot Order* is set to “Rot First”, the objects origin is identical to the origin of the local or reference coordinate system. If the order of rotation is set to “Pos First” this means that the object is first translated and then rotated around the new translated origin of the object if the center is (0, 0, 0), or in general around the point that is at the position defined by the *Rot Center X*, *Rot Center Y* and *Rot Center Z* parameters relative to the objects coordinate origin. If *Pos/Rot Order* is set to “Z Rot XY” the object is first translated in Z direction. Afterwards it is rotated around the center of rotation which is defined relative to the new (in Z translated) origin.

6.8 Position Actions

- **Scale** The scale action allows to scale a Surface, Element or Assembly by a factor.
- **Convert Position** The definition of the position in 3D space defined by a Position can be converted to a different definition type. The actions available in this menu will convert the definition of the position while keeping the elements position. If the definition of the position relies on a lookup parameter from the Multiconfig Lookuptable or the Slider Interface, conversion in not allowed. Following conversions are available:
  - *Rotation Order* Changes the *Rotation Order* while keeping the position. See chapter 6.5

- *Pos Rot Order* Changes the *Pos Rot Order* while keeping the position. See chapter 6.6
- *Convert Pivot* Changes the pivot point while keeping the position. See chapter 6.7

Note: If any of the parameters that define the position of an item dependant on any type of lookup parameter like a slider or a multi configuration parameter conversion is not possible.

7 Basic Elements

Basic Elements are objects like Surfaces, Lenses and so on, that can be placed in the optical design editor. In addition to the Basic Elements, property items can be added to the Basic Elements to specify the shape, phase or in general any optical or physical properties of them. To specify the position of the Basic Elements inside the optical system, every Basic Element has all Position Parameters available. See chapter 6 for the options on how to position Basic Elements.

7.1 Surface

Surfaces are the basic objects any optical system is constructed of. The default Surface has - if no additional properties are assigned - a spherical or planar shape as well as a circular aperture. Furthermore, it is not coated. Surfaces exist in two types. As a Surface of a parental Optical Element or as a single independent Surface. If the Surface does not belong to a parent Element, it has its own "Identifier". If it is part of a parent Element (e.g. part of a Lens) the "Identifier" of the Lens in combination with the Surface Nr. is used to refer to the Surface. In that case the "Identifier" parameter is not available (see chapter 7.7 for further information about the *Identifier*). Besides the common parameters of any basic element the Surface has the following parameters:

Column	3	5
Name	Radius	Apert. Rad
ID	radius	float_ap
Editable	yes	yes

- *Radius* defines the radius of curvature of the Surface as long as the Surface has no form property items assigned to it. If the Surface has any form assigned, the "Radius" parameter is no longer available. In this case the sag of the Surface is define by the Form property items.
- *Apert. Rad* defines the physical and clear aperture of the Surface if no additional Apertures are defined. The default Aperture of the Surface is always circular and centered at (0, 0). It can be defined by the user by entering the radius of the aperture in length units. If the Aperture is unlocked, the radius will be automatically computed so that all rays of any sequence that pass

the Surface lie inside the aperture. If the Surface is not part of any Sequence, the Aperture will be set to zero and the Surface will not be visible in the Optical Design Editor. If one or more Aperture property items are added to the Surface, the parameter is no longer available. In that case the physical and clean apertures are defined by the items.

For Polarization Raytracing, the reflectivity and transmission of a Surface without a Coating are calculated according to the Fresnel equations for the glass/air interface. If the Surface does not belong to an element, it will transmit and reflect all rays without any loss. If this is not what is intended, the reflectivity and transmission can be changed by applying a Coating.

7.1.1 Surface Actions

- **Flipping a Surface** A Surface can be flipped in which case its sag will be mirrored on the xy-plane. The flipped Surface will have the same sag as the original Surface, however the value of the z-coordinate will change the sign. Besides the sag, also the Phase Function of the Surface will change the sign.

7.2 Lenses

Lenses are used to model optical elements that are made of an refractive material like lenses, but can also be used for objects with planar Surfaces like prisms. Mirrors can be modeled as lenses as well if the volumetric substrate should be included e.g. to check the mounting or collisions with other objects. Its parameters are the same as for any Basic Element (Basic parameters and position parameters, see chapter 6.2).

7.2.1  Singlet Lens

The Singlet Lens is a lens that has one Material and two Surfaces. The basic setup can be seen in figure 7.1. The material of the Singlet Lens is specified by the Material property item (See chapter 8 on how to specify the material). The thickness of the Singlet Lens is specified by the Z-position of the second Surface. The diameter of the Singlet Lens is specified over the Apertures of the two optical Surfaces, which may have two different values or set to unlocked, in which case they will be computed automatically. The radius of curvature of the Singlet Lens Surfaces is specified with the "Radius" parameter of each Surface. See chapter 7.1 on how to model lens Surfaces with non spherical shape or non circular Apertures as well as advanced properties.

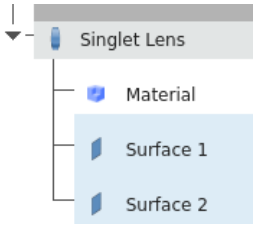


Figure 7.1: Basic architecture of the Singlet Lens

7.2.2 Doublet Lens

The Doublet Lens is a lens that has two Materials and three Surfaces. The basic setup can be seen in figure 7.2. The Materials of each kitted element of the Doublet Lens is specified by the two Material property items (See chapter 8 on how to specify the material). The first Material item specifies the material between Surface 1 and Surface 2. The second Material item specifies the material between Surface 2 and Surface 3. The thickness of the kitted parts of the Doublet Lens is specified by the Z-position of the second surface for the first part and the Z-position of the third surfaces for the second part. The diameter of the Doublet Lens is specified over the Apertures of the three optical Surfaces, which may have different values or set to unlocked, in which case they will be computed automatically. The radius of curvature of the lens Surface is specified with the "Radius" parameter of each Surface. See chapter 7.1 on how to model lens Surfaces with non spherical shape or non circular Apertures as well as advanced properties.

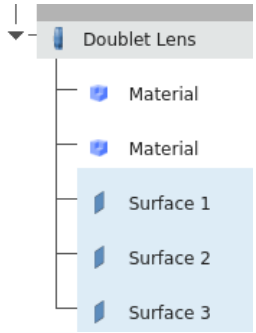


Figure 7.2: Basic architecture of the Doublet Lens

7.2.3 Triplet Lens

The Triplet Lens is a lens that has three Materials and four Surfaces. The basic setup can be seen in figure 7.3. The Materials of each kitted element of the Triplet Lens is specified by the three Material property items (See chapter 8 on how to specify the material). The first Material item specifies

the material between Surface 1 and Surface 2. The second Material item specifies the material between Surface 2 and Surface 3. The third Material item specifies the material between Surface 3 and Surface 4. The thickness of the kitted parts of the Triplet Lens is specified by the Z-position of the second Surface for the first part , the Z-position of the third Surface for the second part and the Z-position of the fourth Surface for the third part. The diameter of the Triplet Lens is specified over the Apertures of the four optical Surfaces, which may have different values or set to unlocked, in which case they will be computed automatically. The radius of curvature of the lens Surfaces is specified with the "Radius" parameter of each Surfaces. See chapter 7.1 on how to model lens Surfaces with non spherical shape or non circular Apertures as well as advanced properties.

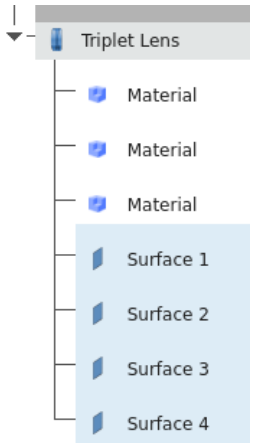


Figure 7.3: Basic architecture of the Triplet Lens

7.2.4 N-let Lens

The N-let Lens is a lens that has N kitted elements. It is the generalized version of the other kitted Lens types. It has N Materials as well as N+1 Surfaces. The basic setup can be seen in figure 7.4. The first Material specifies the material between Surface 1 and Surface 2, the second between Surface 2 and Surface 3 and so on. The thickness of each kitted part is specified by the Z-position of the second Surface of the part. The diameter of the N-let Lens is specified over the Apertures of the optical Surfaces, which may have different values or set to unlocked, in which case they will be computed automatically. The radius of curvature of the lens Surfaces is specified with the "Radius" parameter of each Surface. See chapter 7.1 on how to model lens Surfaces with non spherical shape or non circular Apertures as well as advanced properties.

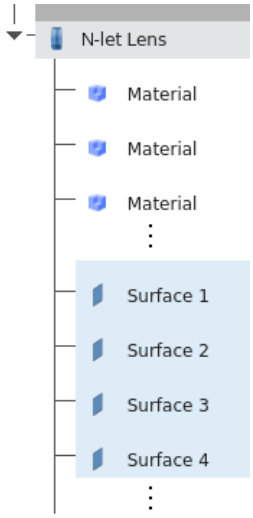


Figure 7.4: Basic architecture of the general N-let Lens

7.2.5 Lens Action

- **Flipping a Lens** A Lens can be flipped which will mirror the lens at the xy-plane. The flipping will revert the order of the Surfaces and - in case it has more than one Material - also the order of the Materials. In addition each Surface will be flipped (See chapter 7.1.1). The result is a complete mirrored version of the original lens. *Note: If any position of a surface inside the lens is dependant on any type of lookup parameter like a slider or a multi configuration parameter flipping is not possible.*
- **Convert to (N+1)-let** A Lens can be converted to a (N+1)-let. E.g. a Singlet lens can be converted to a Doublet, a Doublet to a Triplet and in general a N-let to a (N+1)-let. The conversion will add a additional Surface in between the last two Surfaces of the lens with the same Material as the Material of the last part of the lens. This means, the conversion itself will not change the optical properties of the lens until you change the Material of the added part or change the form of the Surfaces.
- **Convert to (N-1)-let** A Lens can be converted to a (N-1)-let. E.g. a Doublet lens can be converted to a Singlet, a Triplet to a Doublet and in general a N-let to a (N-1)-let. The conversion will remove the second last Surface as well as the last Material. This means, if the Materials of the two last parts of the lens were identical before the conversion, the optical properties of the lens will not change.
- **Split Lens Front** To convert a kitted lens like an achromatic Doublet into two single Lenses or an air-spaced Doublet, lenses can be split. In case the lens is a Doublet only one splitting option (the central Surface) is available. For Triplets or in general for any lens with more

kitted parts, two options are available. Splitting the lens front will remove the first Surface and the first Material from the lens, leaving the other parts as the new lens. The first Surface as well as a copy of the second Surface will be added to a new lens that is created in front of the original lens. The Material will be copied from the original lens part. Unless the user changes the split Surface of one of the lens parts, the optical properties of the two independent lenses will be the same as those of the previous kitted lens, except that now the environment material (typically air) is assumed to be in between the Surface. This may lead to total internal reflection of rays, that could previously be traced through the kitted lens.

- **Split Lens Back** To convert a kitted lens like an achromatic Doublet into two single Lenses or a air-spaced Doublet, lenses can be split. In case the lens is a Doublet only one splitting option (the central Surface) is available. For Triplets, or in general for any lens with more kitted parts, two options are available. Splitting the lens back will remove the last Surface and the last Material from the lens, leaving the other parts as the new lens. A copy of the second last Surface and the last Surface will be added to a new lens that is created below of the original lens. The Material will be copied from the original lens part. Unless the user changes the split Surface of one of the lens parts, the optical properties of the two independent lenses will be the same as those of the previous kitted lens, except that now the environment material (typically air) is assumed to be in between the Surface. This may lead to total internal reflection of rays, that could previously be traced through the kitted lens.
- **Merge with Previous** Merge with previous is the inverse action to splitting the front part of the lens. The merging will be performed by removing the first Surface of the lens and kitting it to the back of the previous one. In case the lens is not positioned directly behind the previous one, it will be moved to the position of the last Surface of the first lens prior kitting. The first Surface of the lens gets lost by this action.
- **Merge with Next** Merge with next is the inverse action to splitting the back part of the lens. The merging will be performed by removing the first Surface of the next lens and kitting it to the back of the lens. In case the next lens is not positioned directly behind the lens, it will be moved to the position of the last Surface of the lens prior kitting. The first Surface of the next lens gets lost by this action.
- **Changing the Power of a Lens** The power of the lens can be changed by either changing the curvature of the individual Surfaces or using the Focus action. The Focus action will change the radius of curvature of both, the first as well as the last Surface of the lens. If the lens has more than two Surfaces, the intermediate Surfaces will not change their curvature. The Focus action is only available for lenses that have an active curvature parameter on each of

the Surfaces (e.g. the default curvature parameter of the Surface, or an spherical or aspherical Form Object.)

- **Bending a Lens** The bending of the lens can be changed by either changing the curvature of the individual Surfaces or using the Bending action. The Bending action will change the radius of curvature of all Surfaces of the lens. The Bending action is only available for lenses that have an active curvature parameter on each of the Surfaces (e.g. the default curvature parameter of the Surface, or an spherical or aspherical Form Object.)

7.3 Custom Element

The Custom Element is an element, that is entirely made of one Material, but - contrary to a Singlet Lens - may have more than two Surfaces. The basic setup of the Custom Element can be seen in figure 7.5.

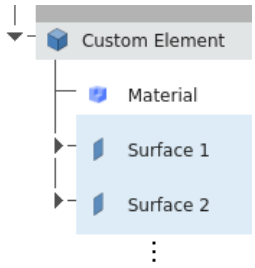


Figure 7.5: Basic architecture of the custom element

7.3.1 Modeling Prisms

To model a simple prism with two optical Surfaces, either a Singlet Lens or a Custom Element can be used. In both cases the two Surfaces typically have Rectangular Aperture and are planar. The angle of the Surfaces can either be applied by rotation the Surfaces via the Position parameters, or as an alternate option by assigning a Tilt Form to the Surface. In the case of prisms with more than two optical Surfaces, e.g. pentaprisms, using a Custom Element is the only available option. To model the prism, each optical Surface that is needed must be added to the element. The non optical Surfaces can - but do not have to - be added as well. When tracing rays through the prism the elements Material is assumed to be between any two of the Surfaces of the element.

7.3.2 Modeling Beam Splitter Cubes

To model a beam splitter cube, the four optical Surfaces, the intermediate splitting Surface and - optionally - the two non optical Surfaces need to be added to the Custom Element. Each Surface gets assigned a Rectangular Aperture. When tracing rays through the cube, the elements Material is assumed to be between any two of the Surfaces of the element. Depending on if only the path of the rays or also the intensity and polarization properties of the beam splitter are of interest, a Coating and/or a Polarization property item can be applied to the splitting Surface.

7.4 Assembly

Assemblies are used to group any Basic Elements where the position of the whole group is defined inside the Assemblies Local Coordinate System. Assemblies may be nested to hold other sub-assemblies. This allows the user to model the mechanical hierarchy according to the real world structure. The advantage when using Assemblies is, that e.g. lens groups can be positioned, decentered or tilted as a whole. This is especially of interest when it comes to Tolerancing (see chapter 19) or when complex optical systems with several parts are modeled.

7.5 Compound

The Compound Surface is a Surface that is made up of several individual Surfaces. It can be used for a wide range of application as segmented telescope mirrors, corner cubes, or other Surfaces where the sag description or other properties of the Surfaces vary between different zones. If the Apertures of the single Surfaces in the Compound Surface overlap (in the local Compound Surfaces coordinate system x, y-plane), always the one that is listed on top will be considered as active for the area. If used with an Auto Sequence the Compound will always be included as a single surface in the Surface List. For a User Defined Sequence it is possible to handle the Compound as a single surface but also to add individual Surfaces that are part of the Compound to the Surface List. The compound has one additional parameter:

Column	18
Name	Advanced Intersection
ID	adv_check
Editable	yes

- *Advanced Intersection*. The parameter should be set to true if the compounds surface sag cannot be expressed as a function $z(x, y)$. If set to true the intersection algorithm of the ray tracer will not only check if the ray is inside the aperture of a sub-surface of the compound, but also if the z-coordinate of the found intersection is on the surface. This can be necessary for surfaces like corner cubes, where the side surface is still inside (exactly at the edge) of the aperture of the other surface. For surfaces like faceted mirrors this check is not necessary and it is recommended to leave the parameter set to false, since it may slow down the ray tracing. Furthermore, the *Surface Form Plot* is not available for compounds with *Advanced Intersection* set to true.

Note that virtual rays cannot be traced through a Compound Surface. The reason for this behavior is, that in order to find the real intersection with the compound and avoid virtual intersections when more than one ray/surface intersection exists, the ray tracer checks if it is a real intersection and ignores the virtual one.

7.6 Ideal Elements

7.6.1  Ideal Lens

Ideal Lenses can be used to build a first draft of an optical system. There are two types of ideal lenses available that either use a paraxial approximation or obey the Abbe Sine Condition. Its parameters are stored in a additional Lens Properties item, that has the following parameters:

Column	3	4	5
Name	Focal Length	Corr. Obj. Dist.	Type
ID	f	u	ideal_type
Editable	yes	yes	yes

- *Focal Length* defines the focal length of the ideal lens
- *Corr. Obj. Dist.* is only available when *Type* is set to "Sine Condition". The parameter is only available when *Type* is set to "Sine Condition". The parameter allows to specify the object distance at which the lens creates an unaberrated image. The distance is specified in the coordinate system of the lens, meaning e.g. in a simple on axis system it would usually have a negative sign.
- *Type* defines the model that is used for the ideal lens. The available options are "Paraxial" and "Sine Condition"

Note that there is no reasonable definition for the optical path length of the "Paraxial" Ideal Lens that works for any scenario. Therefore any analysis that rely on the optical path length should not be used in combination with "Paraxial" Ideal Lenses

Note that the behavior for virtual rays or rays entering the ideal lens in the opposite direction (from right to left) is not well defined. Therefore depending on the combination of ray direction and the virtual state of a ray the focal length may be seem to have a flipped sign.

7.6.2 Ideal Cylindric Lens

Ideal Cylindric Lenses are paraxial lenses where the focal length in x- and y-direction can be specified independently. They focus rays to a perfect line without any aberrations, no matter what Wavelength or Field Angle. Its parameters are stored in a additional Lens Properties item, that has the following parameters:

Column	3	4
Name	Focal Length X	Focal Length Y
ID	fx	fy
Editable	yes	yes

Note that there is no reasonable definition for the optical path length of the Ideal Cylindric Lens that works for any scenario. Therefore any analysis that rely on the optical path length should not be used in combination with Ideal Cylindric Lens.

Note that the behavior for rays entering the ideal lens in the opposite direction (from right to left) is not well defined

7.7 Surface and Element IDs

Lenses, Elements as well as Assemblies and Surfaces have an *Identifier* as the first parameter. The *Identifier* is used to refer to the object. This is e.g. used in User Sequences to set up the List of Surface to be traced. Another application is when assigning Tolerances to specific elements (see chapter 19.1.1). Surfaces only have their own *Identifier* if they are not part of an parent Element. Else, the parent's Elements "Identifier" in combination with the *Surface Nr.* is used to refer to the Surfaces. The *Identifier* must start with a lower or capital letter and may - besides letters - also contain numbers as well as most special characters. The *Identifier* must be unique, therefore it is not possible to e.g. have a Surface as well as a Assembly that both share the same *Identifier*. If a

user accidentally enters the same *Identifier* twice, it will be automatically changed to some unique *Identifier*.

8 Material

The Material item is used to specify the material of a lens or in general of any optical element that is made of a material with a refractive index other than the Environment is made of. The Material has the following parameters:

Column	0	1	2	3
Name	Object Type	-	Comment	Type
ID	-	-	comment	model_type
Editable	no	no	yes	yes

- *Object Type* is a non editable parameter that describes the basic type of the item.
- *Identifier* is a unique string ID that is used to refer to the object. The identifier may be set to any string that starts with a lower or upper case character. Surfaces only have an ID of their own, if they are not part of a lens or any other optical element. If they are part of an Element, the ID of the Element in combination with the surface number is used to refer to the Surface.
- *Comment* is a free string parameter that can be used for comments and documentation.
- *Type* defines the material type: "Model" allows to model the Material via the refractive index and Abbe number. "Catalog" allows to specify a Material from any of the glass catalogs included in the Quadoa installation, as well as those added by the user.

8.1 Catalog Material

The catalog Material is used when *Type* is set to "Catalog". It has the following additional parameters:

Column	4	5	6	7	8
Name	Name	Catalog	Substitute	Is Thermal	Temperature
ID	name	ctlg	subst	is_thermal	temp
Editable	yes	yes	yes	yes	yes

- *Name* specifies the name of the Material in the glass catalog.
- *Catalog* specifies the glass catalog which contains the Material.
- *Substitute* is used to mark the Material as Substitute candidate. (See chapter 18.3)
- *Is Thermal* is used to enable the *Temperature* parameter for the Material. If the parameter is disabled, the Temperature of the Material is assumed to be the same as the temperature of the Environment.
- *Temperature* is used to set the temperature of the Material.

  Note that the temperature parameter only has an effect on the refractive index, if the temperature parameters are specified in the manufacturers glass catalog, which is not always the case (see chapter 25.1.2.1).

To specify the Material, click on the Catalog icon besides the Material name and browse the desired Catalog for the Material (see chapter 3.6.2.1 on how to use the Glass Catalog). Another option is to directly enter the Material name into the *Name* parameter field. In case only one Material with the specified name exists in any of the available Catalogs, the *Catalog* parameter will be automatically inserted. If a Material with the same name exists in more than one Glass Catalog, the user needs to specify the Catalog as well. As long as the Material is not specified, cannot be found in the specified Catalog or does not exist, the Material will be highlighted in red and an error message will be displayed in the *Status Info Bar*.

8.2 Model Material

The model Material is a simple material that directly lets the user specify the refractive index and - optionally - the Abbe number. It has the following additional parameters:

Column	4	5	6
Name	Abbe Type	Index	Abbe Nr
ID	model_sub_type	index	abbe_ve or abbe_vd
Editable	yes	yes	yes

The reference Environment for the The model Material is always assumed to be Air at $20^{c}irc$ and 1 bar of atmospheric pressure. The values for these parameters can be set as variable during optimization (see chapter 18). This is helpful to identify an useful parameter range when searching

for Materials e.g. for achromatization. Depending on the *Abbe Type* parameter following options are available.

8.2.1 Constant Index

When *Abbe Type* is set to "Const. Index", only the *Index* parameter is available. In this case the refractive index of the Material can directly be specified and is the same for any Wavelength of the light. This is typically only useful for monochromatic applications, where the dispersion of the Material can be ignored.

8.2.2 VD

When *Abbe Type* is set to "VD", the refractive index as well as the Abbe number V_D can be specified. V_D is defined by:

$$V_D = \frac{n_D - 1}{n_F - n_C} \tag{8.1}$$

Where n_D, n_F and n_C are the Wavelengths at the respective Fraunhofer lines.

To get the refractive index of a "Abbe Type" Material for a given Wavelength Cauchy's dispersion equation is used.

$$n(\lambda) = A\frac{B}{\lambda^2} \tag{8.2}$$

where

$$B = \frac{n_d - 1.0}{V_D(\lambda_F^{-2} - \lambda_C^{-2})} \tag{8.3}$$

and

$$A = n_e - \frac{B}{\lambda_d^2} \tag{8.4}$$

8.2.3 VE

When *Abbe Type* is set to "VE", the refractive index as well as the Abbe number V_E can be specified. V_E is defined by:

$$V_E = \frac{n_e - 1}{n_{F'} - n_{C'}} \tag{8.5}$$

Where n_e $n_{F'}$ and $n_{C'}$ are the Wavelengths at the respective Fraunhofer lines.

To get the refractive index of a "Abbe Type" Material for a given Wavelength Cauchy's dispersion equation is used.

$$n(\lambda) = A\frac{B}{\lambda^2} \tag{8.6}$$

where

$$B = \frac{n_e - 1.0}{V_E(\lambda_{F'}^{-2} - \lambda_{C'}^{-2})} \tag{8.7}$$

and

$$A = n_e - \frac{B}{\lambda_e^2} \tag{8.8}$$

8.3 Material Actions

- **Convert to Abbe VE** Converts the current Material to a Abbe VE Model Material by fitting the Abbe number and refractive index to match the current material.
- **Convert to Abbe VD** Converts the current Material to a Abbe VD Model Material by fitting the Abbe number and refractive index to match the current material.
- **Replace with best fit Catalog Material** Searches the Substitute Material Catalog for the material that is the closest match to the specified index and Abbe number and replaces the Model material with it.

8.4 GRIN

An GRIN objects can be used to add a gradient index profile to a Material. The index change defined by the grin profile is added to the base refractive index of the Material. The total refractive index evaluates to

$$n = n(\lambda, T) + n_{GRIN}(x, y, z), \tag{8.9}$$

where λ is the wavelength, T the Material temperature and x, y, z the position inside the local element coordinate system. To obtain the curved path of a ray through the inhomogeneous medium, the Eikonal equation is solved numerically. Each GRIN profiles therefore has a parameter *Z Step* that defines the step size after which the differential equation is solved. A smaller step size leads to a more accurate result however it also increases the computation time. To find the best value for

the step size it is recommended to start with a large value and reduce it until the result does not significantly change any more in the order of the desired accuracy.

If more than one GRIN profiles are defined for a Material only the first profile is considered.

8.4.1 Radial

The Radial GRIN profile defines a radial symmetric grin profile defined by a polynomial. It has the following parameters:

Column	3	4	5	6	...
Name	Z Step	Polynomial Order Y	A1	A2	An
ID	z_step	maxorder	a1	a2	an
Editable	yes	yes	yes	yes	yes

where *Polynomial Order* defines the maximum polynomial order, and *A* the coefficients of the polynomial. The profile is defined as

$$\Delta n = A_1 r + A_2 r^2 + A_3 r^3 + A_4 r^4 .... \tag{8.10}$$

with

$$r = \sqrt{x^2 + y^2} \tag{8.11}$$

being the distance from the material central axis.

8.4.2 Axial

The Axial GRIN profile defines a GRIN profile that is symmetric along the X- or Y-plane. It has the following parameters:

Column	3	4	5	6	7	...
Name	Z Step	Axis	Polynomial Order Y	A1	A2	An
ID	z_step	axis	maxorder	a1	a2	an
Editable	yes	yes	yes	yes	yes	yes

if *Axis* is set to "X" the profile is defined as

$$\Delta n = A_1 y + A_2 y^2 + A_3 y^3 + A_4 y^4 .... \tag{8.12}$$

else if *Axis* is set to "Y" the profile is defined as

$$\Delta n = A_1 x + A_2 x^2 + A_3 x^3 + A_4 x^4 .... \tag{8.13}$$

8.4.3 Spherical

The Spherical GRIN profile defines a grin profile that has a spherical symmetry. It has the following parameters:

Column	3	4	5	6	7	...
Name	Z Step	Z0	Polynomial Order	A1	A2	An
ID	z_step	z0	maxorder	a1	a2	an
Editable	yes	yes	yes	yes	yes	yes

The parameter *Z0* defined the center of the spherical symmetry along the local element Z-axis. The profile is defined as

$$\Delta n = A_1 r + A_2 r^2 + A_3 r^3 + A_4 r^4 .... \tag{8.14}$$

with

$$r = \sqrt{x^2 + y^2 + (z - Z_0)^2} \tag{8.15}$$

8.4.4 Luneburg

The Luneburg GRIN profile defines a GRIN profile to describe a Lüneburg lens. It has the following parameters:

Column	3	4	5
Name	Z Step	Radius	A
ID	z_step	rad	a
Editable	yes	yes	yes

The parameter *Radius* defined the center of the spherical symmetry along the local element Z-axis, which is also the radius of the Lüneburg lens. The profile is defined as

$$\Delta n = \sqrt{n_0^2(2.0 - \frac{r^2}{A^2})} \tag{8.16}$$

with

$$r = \sqrt{x^2 + y^2 + (z - Radius)^2} \tag{8.17}$$

8.4.5 Rochester

The Rochester GRIN profile defines a GRIN profile according to the University of Rochester GRIN equation. It has the following parameters:

Column	3	4	5	6	7	8	9	10	11
Name	Z Step	N_Z	N_Z2	N_Z3	N_Z4	N_R2	N_R4	N_R6	N_R8
ID	z_step	nz1	nz2	nz3	nz4	nr2	nr4	nr6	nr8
Editable	yes	yes	yes	yes	yes	yes	yes	yes	yes

its profile is defined by

$$\Delta n = N_Z z + N_{Z2} z^2 + N_{Z3} z^3 + N_{Z4} z^4 + N_{R2} r^2 + N_{R4} r^4 + N_{R6} r^6 + N_{R8} r^8 \tag{8.18}$$

8.4.6 3D Grid

The 3D Grid GRIN profile can define an arbitrary GRIN profile, that is defined by a 3D interpolation table. The main purpose of this GRIN profile is to describe material inhomogeneities or to import data from STOP analysis to e.g. model the effects of thermal lensing. It has the following parameters:

Column	3
Name	Z Step
ID	z_step
Editable	yes

9 Form / Surface Sag Description

Form Objects are used to describe the sag $z(x, y)$ of a surface in the surface Coordinate System. The value of x, y and the sag z is always in the length unit specified in the settings, typically mm. To add a Form Objects to a Surface, select the Surface in the *Optical Design Editor*. Then click the desired Form Objects which should be added. The object will appear as a child element of the selected Surface. If no Surface is selected, an error message will appear. Another option is to use the right-click-menu, which is available by right clicking on the Surface in the *Optical Design Editor*. If a Surface has no Form Objects, it will be handled as a default spherical Surface. If the Surface has a singe Form Object, the sag will be calculated by the sag formula $z(x, y)$ of the specific object. If a Surface has more than one Form Object, the sag will be calculated as a sum of the single Form Objects.

$$z(x, y) = \sum_{n=0}^{n_f - 1} z_n(x, y) \tag{9.1}$$

with n_f being the number of Form Objects and $z_n(x, y)$ the sag function of the single objects. Besides the Basic Form Objects for basic shapes, Polynomial Form Objects and Operator Objects exist. The operators can be used to e.g. generate arrays or Fresnel-type surfaces. Like the basic Form Objects, the Operator Objects can be stacked together with other objects as well.

All Form Objects have the first three parameters in common:

Column	0	1	2
Name	Object Type	-	Comment
ID	type	-	comment
Editable	no	no	yes

- *Object Type* is a non editable parameter that describes the basic type of the item.
- *Comment* is a free string parameter that can be used for comments and documentation.

9.1 Basic Form Objects

9.1.1 Sphere

The Sphere Surface Form Object describes a simple spherical surface. It has only one parameter which is the *Radius* of the sphere. The sag at the center of the surface coordinate system is always zero.
The Object has following additional parameters:

Column	3
Name	Radius
ID	radius
Editable	yes

9.1.2 Tilt

The Tilt Surface Form Object describes a tilted plane that is specified by the two tilt parameters.

Column	3	4
Name	Tilt X	Tilt Y
ID	t_x	t_y
Editable	yes	yes

The parameters *Tilt X* and *Tilt Y* describe the slope of the surface in x- and y-direction. This means that if e.g. the tilt in y-direction is 0.5, the surface will have a sag of +5.0 mm at x = 0.0, y = +10.0 mm and -5.0 mm at the position x = 0.0 y = -10.0mm.

9.1.3 Paraboloid

The Paraboloid Surface Form Objects describes a simple paraboloid surface. The Object has following additional parameters:

Column	3	4
Name	Offset	Quadratic
ID	offset	quadratic
Symbol	s	A^2
Editable	yes	yes

The *Quadratic* parameter is the coefficient for the parabola. *Offset* is the sag at the center.

$$z(x, y) = A^2 \cdot (x^2 + y^2) + s \tag{9.2}$$

9.1.4 Asphere

The Asphere Surface Form Object describes a standard aspheric surface. It has a variable amount of parameters depending on the value of the polynomials parameter. The sag at the center of the surface coordinate system is always zero.
The Asphere has following additional parameters, with i from 1 to the value of parameter polynomials:

Column	3	4	5	6	...	
Name	Radius	Conic	Polynomials	A2	...	A2i
ID	radius	conic	maxorder	a2	...	a2i
Symbol	r	c	-	A_2	...	A_{2i}
Editable	yes	yes	yes	yes	...	yes

With

$$h = \sqrt{x^2 + y^2} \tag{9.3}$$

being the distance from the vertex of the surface, the sag can be calculated to

$$z(h) = \frac{h^2/r}{1 + \sqrt{1 - (1 + c)(h/r)^2}} + \sum_{i=1}^{n} A_{2\mathrm{i}} \cdot h^{2i} \tag{9.4}$$

The number of polynomial coefficients can be set by parameter Polynomials.

9.1.5 Off-Axis-Asphere

The Off-Axis-Asphere Surface Form Object is equal to the Asphere surface form description. It has three additional parameters. *Decenter X* and *Decenter Y* define the decentering of the vertex in x and y direction. The parameter *Z Reference* defines the offset of the surface sag. If set to none, the vertex of the surface is at $z = 0$. If *Z Reference* is set to center, than $z(x = 0, y = 0) = 0$. The Off-Axis-Asphere has following additional parameters, where i goes from 1 to the value of parameter Polynomials:

Column	3	4	5	6	7	8	9	...	
Name	Radius	Conic	Dec. X	Dec. Y	Z Ref.	Polynomials	A2	...	A2i
ID	radius	conic	d_x	d_y	ref	maxorder	a2	...	a2i
Symbol	r	c	d_x	d_y	-	-	A_2	...	A_{2i}
Editable	yes	yes	yes	yes	yes	yes	yes	...	yes

With

$$h = \sqrt{(x - d_x)^2 + (y - d_y)^2} \tag{9.5}$$

being the distance from the vertex of the surface, the sag can be calculated to

$$z(h) = \frac{h^2/r}{1 + \sqrt{1 - (1 + c)(h/r)^2}} + \sum_{i=1}^{n} A_{2\mathrm{i}} \cdot h^{2i} \tag{9.6}$$

If Z Reference is set to center, the surface is shifted in z direction so $z(0, 0)$ is set to zero. The number of polynomial coefficients can be set by parameter Polynomials.

9.1.6 Asphere QCon

The Asphere QCon Surface Form Object describes an aspheric surface defined by th QCon polynomials introduced by G. Forbes. It has a variable amount of parameters depending on the value of the polynomials parameter. The sag at the center of the surface coordinate system is always zero. The Asphere QCon has following additional parameters, with n from 1 to the value of parameter polynomials:

Column	3	4	5	6	7	...	
Name	Radius	Conic	Norm Radius	Polynomials	Q0	...	Qn
ID	radius	conic	normrad	maxorder	Q0	...	Qn
Symbol	r	c	r_n	-	Q_0	...	Q_n
Editable	yes	yes	yes	yes	yes	...	yes

The number of polynomial coefficients can be set by parameter Polynomials. See [2] for the definition of the sag formula.

Note: the conic term is not included in the original definition of the polynomials and should not be used, since it has a negative effect on the numerical properties of the asphere, especially during optimization. To be compatible with other optical design software the conic is included, however it is strongly recommended to always set it to zero.

9.1.7 Asphere QBfs

The Asphere QBfs Surface Form Object describes an aspheric surface defined by th QBfs polynomials introduced by G. Forbes. It has a variable amount of parameters depending on the value of the polynomials parameter. The sag at the center of the surface coordinate system is always zero.
The Asphere QBfs has following additional parameters, with n from 1 to the value of parameter polynomials:

Column	3	4	5	6	7	...	
Name	Radius	Conic	Norm Radius	Polynomials	Q0	...	Qn
ID	radius	conic	normrad	maxorder	Q0	...	Qn
Symbol	r	c	r_n	-	Q_0	...	Q_n
Editable	yes	yes	yes	yes	yes	...	yes

The number of polynomial coefficients can be set by parameter Polynomials. See [2] for the definition of the sag formula.

Note: the conic term is not included in the original definition of the polynomials and should not be used, since it has a negative effect on the numerical properties of the asphere, especially during optimization. To be compatible with other optical design software the conic is included, however it is strongly recommended to always set it to zero.

9.1.8 Axicon

The Axicon Surface Form Object describes a cone shaped surface form, as it is often used in laser beam shaping to generate annular intensity distributions or Bessel beams. The sag at the center of the surface Coordinate System is always zero.
The Axicon has one additional parameters:

Column	4
Name	Angle
ID	angle
Symbol	α
Editable	yes

With

$$h = \sqrt{x^2 + y^2} \tag{9.7}$$

being the distance from the vertex of the surface, the sag can be calculated to

$$z(h) = h \cdot tan(\alpha) \tag{9.8}$$

9.1.9 Cylinder

The Cylinder Surface Form Object describes a cylindrical surface form. The sag at the center (as well as along the direction of the axis of the cylinder) of the surface Coordinate System is always zero.
The Cylinder has one additional parameters:

Column	3	4
Name	Radius	Axis
ID	radius	axis
Symbol	r	-
Editable	yes	yes

Depending on the value of the *Axis* parameter, the axis of the cylinder is oriented in x or y direction.

9.1.10 Acylinder

The Acylinder Surface Form Object describes an acylindrical surface, where the sag in one coordinate (eg. x) is constant and the sag in the other direction is described by a standard aspheric surface description. Acylinders are typically used to focus a beam to a line. It has a variable amount of parameters, depending on the value of the polynomials parameter. The sag at the center of the surface Coordinate System is always zero.
The Acylinder has following additional parameters, where i goes from 1 to the value of parameter Polynomials:

Column	3	4	5	6	7		
Name	Radius	Conic	Axis	Polynomials	A2	...	A2i
ID	radius	conic	axis	maxorder	a2	...	a2i
Symbol	r	c	-	-	A_2	...	A_{2i}
Editable	yes	yes	yes	yes	yes	...	yes

If *Axis* is set to y

$$z(x,y) = \frac{x^2/r}{1+\sqrt{1-(1+c)(x/r)^2}} + \sum_{i=1}^{n} A_{2\mathrm{i}} \cdot x^{2i} \tag{9.9}$$

else if *Axis* is set to x

$$z(x,y) = \frac{y^2/r}{1+\sqrt{1-(1+c)(y/r)^2}} + \sum_{i=1}^{n} A_{2i} \cdot y^{2i} \tag{9.10}$$

The number of polynomial coefficients can be set by parameter polynomials.

9.1.11 Acylinder QCon

The Acylinder QCon Surface Form Object describes an acylindrical surface, where the sag in one coordinate (eg. x) is constant and the sag in the other direction is described by a QCon-polynomial aspheric surface description. Acylinders are typically used to focus a beam to a line. It has a variable amount of parameters, depending on the value of the polynomials parameter. The sag at the center of the surface Coordinate System is always zero.
The Acylinder has following additional parameters, where i goes from 1 to the value of parameter Polynomials:

Column	3	4	5	6	7	8		
Name	Radius	Conic	Norm Rad	Axis	Polynomials	Q0	...	Qn
ID	radius	conic	normrad	axis	maxorder	Q0	...	Qn
Editable	yes	yes	yes	yes	yes	yes	...	yes

The number of polynomial coefficients can be set by parameter polynomials.

Note: the conic term is not included in the original definition of the polynomials and should not be used, since it has a negative effect on the numerical properties of the asphere, especially during optimization. To be compatible with other optical design software the conic is included, however it is strongly recommended to always set it to zero.

9.1.12 Acylinder QBfs

The Acylinder QBfs Surface Form Object describes an acylindrical surface, where the sag in one coordinate (eg. x) is constant and the sag in the other direction is described by a QBfs-polynomial aspheric surface description. Acylinders are typically used to focus a beam to a line. It has a variable amount of parameters, depending on the value of the polynomials parameter. The sag at the center of the surface Coordinate System is always zero.
The Acylinder has following additional parameters, where i goes from 1 to the value of parameter Polynomials:

Column	3	4	5	6	7	8		
Name	Radius	Conic	Norm Rad	Axis	Polynomials	Q0	...	Qn
ID	radius	conic	normrad	axis	maxorder	Q0	...	Qn
Editable	yes	yes	yes	yes	yes	yes	...	yes

The number of polynomial coefficients can be set by parameter polynomials.

Note: the conic term is not included in the original definition of the polynomials and should not be used, since it has a negative effect on the numerical properties of the asphere, especially during optimization. To be compatible with other optical design software the conic is included, however it is strongly recommended to always set it to zero.

9.1.13 Biconic

The Biconic Surface Form Object describes a surface that is based on a conic section in x and y direction. The parameters for each direction are independent from each other.

The Biconic object has following additional parameters:

Column	3	4	5	6
Name	Radius x	Radius y	Conic x	Conic y
ID	radius_x	radius_y	conic_x	conic_y
Symbol	r_x	r_y	c_x	c_y
Editable	yes	yes	yes	yes

$$z(x,y) = \frac{\frac{x^2}{r_x} + \frac{y^2}{r_y}}{1 + \sqrt{1 - (1+c_x)\frac{x^2}{r_x} - (1+c_y)\frac{y^2}{r_y}}} \tag{9.11}$$

9.1.14 Toroidal

The Toroidal Surface Form Object describes a surface that is based on a torus. The torus is generated by a circle where the radius of the circle is defied by *Radius of Circle* that gets extruded along a second circle where the radius is defined by *Radius of Revolution*.

The Toroidal object has following additional parameters:

Column	3	4	5
Name	Radius of Circle	Radius of Revolution	Axis of Revolution
ID	radius_small	radius_large	axis
Symbol	R_S	R_L	-
Editable	yes	yes	yes

If *Axis of Revolution* is set to "Y" the sag evaluated to

$$z(x,y) = R_L - \sqrt{(R_L - \frac{y^2}{R_S(\sqrt{1 - y^2/R_S^2} + 1)})^2 - x^2} \tag{9.12}$$

else if *Axis of Revolution* is set to "X" the sag evaluated to

$$z(x,y) = R_L - \sqrt{(R_L - \frac{x^2}{R_S(\sqrt{1 - x^2/R_S^2} + 1)})^2 - y^2} \tag{9.13}$$

9.1.15 Cosine

The Cosine Surface Form Object describes a periodic harmonic surface sag. The main purpose of this Form Object is to simulate the behavior of optical systems in regard to the spatial frequency of surface errors. The Cosine object has four additional parameters:

Column	3	4	5	6
Name	Periode Length	Amplitude	Phase Offset	Axis
ID	periode	amplitude	phase_offset	axis
Symbol	p	a	ϕ	-
Editable	yes	yes	yes	yes

If *Axis* is set to y

$$z(x,y) = a \cdot \cos(\frac{2\pi}{p} \cdot (x - \phi)) \tag{9.14}$$

else if *Axis* is set to x

$$z(x,y) = a \cdot \cos(\frac{2\pi}{p} \cdot (y - \phi)) \tag{9.15}$$

9.1.16 Periodic Rot

The Periodic Rot Surface Form Object describes a periodic harmonic surface sag, where the harmonic function is applied in the radial direction and then rotated around the z axis. The main purpose of

this Form Object is to simulate the behavior of optical systems in regard to the spatial frequency of surface errors. The Periodic Rot object has two additional parameters:

Column	3	4
Name	Periode Length	Amplitude
ID	periode	amplitude
Symbol	p	a
Editable	yes	yes

With

$$h = \sqrt{x^2 + y^2} \tag{9.16}$$

the sag is computed by

$$z(x, y) = a \cdot \cos(\frac{2\pi h}{p}) \tag{9.17}$$

9.1.17 Gauss

The Gauss Surface Form Objects describes a local surface sag bump that is based on a Gaussian function. This can be used to describe local surface defects in simulations and is not meant for design purposes but rather simulation of existing systems. By the combination of several (decentered) Gauss bumps, it is in principal possible to describe arbitrary surface defects.

The Gauss object has the following additional parameters:

Column	3	4	5	6	7
Name	Amplitude	Sigma	Dec. x	Dec. y	Norm
ID	amplitude	sigma	d_x	d_y	norm
Symbol	a	σ	d_x	d_y	–
Editable	yes	yes	yes	yes	yes

If *Norm* is set to true, the amplitude of the peak is normed. In this case the volume under the peak is always 1.0 and the sag is calculated by

$$z(x, y) = \frac{a}{2\pi\sigma^2} \cdot e^{-\frac{x^2+y^2}{2\sigma^2}} \tag{9.18}$$

If *Norm* is set to false, then the Amplitude directly affects the peaks height in length units. In this case the sag is calculated by

$$z(x, y) = a \cdot e^{-\frac{x^2+y^2}{2\sigma^2}} \tag{9.19}$$

9.1.18 Super Gauss

The Super Gauss Surface Form Objects describes a local surface sag bump that is based on a Super Gaussian function. This can be used to describe local surface defects in simulations and is not meant for design purposes but rather simulation of existing systems. By the combination of several (decentered) Gauss bumps, it is in principal possible to describe arbitrary surface defects.

The Super Gauss object has the following additional parameters:

Column	3	4	5	6	7
Name	Amplitude	Sigma	Power	Dec. x	Dec. y
ID	amplitude	sigma	pow	d_x	d_y
Symbol	a	σ	P	d_x	d_y
Editable	yes	yes	yes	yes	yes

The sag of the Super Gauss is defined by

$$z(x,y) = a \cdot e^{-(\frac{x^2+y^2}{2\sigma^2})^P} \tag{9.20}$$

9.1.19 Elliptic Gauss

The Elliptic Gauss Surface Form Objects describes a local surface sag bump that is based on a Elliptic Gaussian function. This can be used to describe local surface defects in simulations and is not meant for design purposes but rather simulation of existing systems. By the combination of several (decentered) Gauss bumps, it is in principal possible to describe arbitrary surface defects.

The Elliptic Gauss object has the following additional parameters:

Column	3	4	5	6	7
Name	Amplitude	Sigma X	Sigma Y	Dec. x	Dec. y
ID	amplitude	sigma_x	sigma_y	d_x	d_y
Symbol	a	σ_x	σ_y	d_x	d_y
Editable	yes	yes	yes	yes	yes

The sag of the Elliptic Gauss is defined by

$$z(x,y) = a \cdot e^{-(\frac{x^2}{2\sigma_x^2}+\frac{y^2}{2\sigma_y^2})} \tag{9.21}$$

9.1.20 Elliptic Super Gauss

The Elliptic Super Gauss Surface Form Objects describes a local surface sag bump that is based on a Elliptic Super Gaussian function. This can be used to describe local surface defects in simulations and is not meant for design purposes but rather simulation of existing systems. By the combination of several (decentered) Gauss bumps, it is in principal possible to describe arbitrary surface defects.

The Elliptic Super Gauss object has the following additional parameters:

Column	3	4	5	6	7	8
Name	Amplitude	Sigma X	Sigma Y	Exponent	Dec. x	Dec. y
ID	amplitude	sigma_x	sigma_y	pow	d_x	d_y
Symbol	a	σ_x	σ_y	P	d_x	d_y
Editable	yes	yes	yes	yes	yes	yes

The sag of the Elliptic Super Gauss is defined by

$$z(x,y) = a \cdot e^{-(\frac{x^2}{2\sigma_x^2}+\frac{y^2}{2\sigma_y^2})^P} \tag{9.22}$$

9.1.21 Rectangular Super Gauss

The Rectangular Super Gauss Surface Form Objects describes a local surface sag bump that is based on a Rectangular Super Gaussian function. This can be used to describe local surface defects in simulations and is not meant for design purposes but rather simulation of existing systems. By the combination of several (decentered) Gauss bumps, it is in principal possible to describe arbitrary surface defects.

The Rectangular Super Gauss object has the following additional parameters:

Column	3	4	5	6	7	8
Name	Amplitude	Sigma X	Sigma Y	Exponent	Dec. x	Dec. y
ID	amplitude	sigma_x	sigma_y	pow	d_x	d_y
Symbol	a	σ_x	σ_y	P	d_x	d_y
Editable	yes	yes	yes	yes	yes	yes

The sag of the Rectangular Super Gauss is defined by

$$z(x,y) = a \cdot e^{-((\frac{x^2}{2\sigma_x^2})^P+(\frac{y^2}{2\sigma_y^2})^P)} \tag{9.23}$$

9.1.22 Grid

The Grid Form Object is used to describe a sufrace sag where the sag value is defined by a 2D lookup table. For the table the interpolation can be set to "Nearest Neighbor", "Linear" or "Bicubic".

9.1.23 Python

The Python Surface Form Object is a general Form object that can define arbitrary surface sag shapes that are defined via a Python class. It has the following parameters

Column	3	4	5	6	7
Name	Class Name	Numeric Derivation	Param_1	...	Param_n
ID	cname	num_norm	-	-	-
Editable	yes	yes	yes	yes	yes

The *Class Name* parameter specifies the name of the Python class, that defines the shape. Depending on the parameters defined in the Python class additional parameters are available. To make a user defined Python form available in Quadoa, the Python class must be added in a file called python_forms.py which is located in the python folder inside the Quadoa User Data Folder. The default path is `C:\Users\UserName\Quadoa\python` on Windows or `/home/UserName/Quadoa/python` on Linux.

Note that models with Python surfaces are not supported from within the Quadoa Python module or Matlab® toolbox or & C++ SDK.

To define a Python Form class the following methods have to be implemented:

- **getParameters()** returns an array that contains the parameter names
- **getZ()** returns a double that is the sag at a given X-, Y-coordinate $z = z(x, y)$
- **getNormal()** returns three double values that define the surface normal at a given X-, Y-coordinate $nx, ny, nz = normal(x, y)$

if *Numeric Derivation* is set to true the getNormal() method is not necessary, however the evaluation will gerally be slower.

An example implementation of a Python Form could look like this:

```
# Eample 2 Bi-Paraboloid surface form with 2 parameters
# To refer to  this definition the Class Name Param of the Python Forms
# has to be set to "BiParaboloid"
class BiParaboloid():

    # Method that tells Quadoa what parameters the PyhtonSurface has
    # Here the parameters that will be available for the surface type
    # need to be listed
    def getParameters():
        return ["PowerX", "PowerY"]

    # Method to get z(x, y) of the surface (sag function)
    # Parameters declared in getParameters()
    # can be accesed by params["ParamName"]
    def getZ(x, y, params):
        return params["PowerX"]*x*x + params["PowerY"]*y*y

    # Method to get the normal vector for the surface at (x, y)
    # needs to return the normal vector nx, ny, nz
    # The vector does not have to be normed
    def getNormal(x, y, params):
        dx = -2*params["PowerX"]*x
        dy = -2*params["PowerY"]*y
        return dx, dy, 1.0
```

9.2 Polynomial Form Objects

9.2.1 Zernike

The Zernike Surface Form Object is a polynomial form object, that uses a Zernike polynomial (see Section 26.1 for details) to describe the sag function. It has the following additional parameters:

Column	3	4	5	6	...	
Name	Norm Radius	Polynomial Order	Sort Order	Z2	...	Zn
ID	normrad	maxorder	sort_order	z1	...	zn
Symbol	r_n	-	-	Z_1	...	Z_n
Editable	yes	yes	yes	yes	...	yes

9.2.2 Zernike Fringe

The Zernike Fringe Surface Form Object is a polynomial form object, that uses a Zernike Fringe polynomial (see Section 26.1.2 for details) to describe the sag function. It has the following additional parameters:

Column	3	4	...	
Name	Norm Radius	Z2	...	Zn
ID	normrad	z1	...	zn
Symbol	r_n	Z_1	...	Z_n
Editable	yes	yes	...	yes

9.2.3 XYZ Polynomial

The XYZ Polynomial Surface Form Object is a polynomial form object, that uses a XYZ Polynomial (see Section 26.2 for details) to describe the sag function. It has the following additional parameters:

Column	3	4	5	...	
Name	Norm Radius	Polynomial Order	X1Y0	...	XnYm
ID	normrad	maxorder	x1y0	...	xnym
Symbol	r_n	-	X_1Y_0	...	Y_nY_m
Editable	yes	yes	yes	...	yes

9.2.4 Form Operator Objects

9.2.4.1  Array Operator

The Form Array Operator is used to generate arrays of surface Form objects. This is typically used to generate micro lens arrays, with e.g. spherical, cylindrical or aspherical lenses. However, it can be used with any Form object as well as a Form stack. Like any other Form object it can be stacked.

It has the following additional parameters:

Column	3	4	5	6
Name	Pitch X	Pitch Y	Decenter X	Decenter Y
ID	pitch_x	pitch_y	dec_x	dec_y
Symbol	-	-	-	-
Editable	yes	yes	yes	yes

The *Pitch X* and *Pitch Y* values define the size of the single array cells in x and y direction, while the *Decenter X* and *Decenter Y* values specify the shift in x and y direction.

9.2.4.2  Hex Array Operator

The Form Hex Array Operator is used to generate hexagonal arrays of surface Form objects. This is typically used to generate micro lens arrays, with e.g. spherical, cylindrical or aspherical lenses. However, it can be used with any Form object as well as a Form stack. Like any other Form object it can be stacked.

It has the following additional parameters:

Column	3	4	5
Name	Size	Decenter X	Decenter Y
ID	size	dec_x	dec_y
Symbol	-	-	-
Editable	yes	yes	yes

The *Size* value define the size of the regular hexagons edge lenght of a single array cells, while the *Decenter X* and *Decenter Y* values specify the shift in x and y direction.

9.2.4.3 Transformation Operator

The Form Transformation Operator is used to generate decentered, rotate, enlarged or squeezed Forms. It will apply the transformation to any Form Objects that is added as child item.

It has the following additional parameters:

Column	3	4	5	6	7
Name	Decenter X	Decenter Y	Scale X	Scale Y	Rotate
ID	dec_x	dec_y	s_x	s_y	rot
Symbol	-	-	-	-	-
Editable	yes	yes	yes	yes	yes

The *Decenter* parameters move the X and Y coordinates inside the surface coordinate system. The *Scale* parameters applies a scaling operation along the X or Y axis. The *Rotation* parameter rotates the shape around the Z-axis. The order the operations are applied is as follows: first the shape is translated, then the rotation is applied and finally it is scaled along the axis of the surface coordinate system. To achieve other orders for the operations several Form Transformation Operators can be nested.

9.2.4.4 Fresnel Operator

The Fresnel Operator is used to generate Frensel type lenses out of surface Forms. It can be used with any Form object as well as a Form stack. Like any other Form object it can be stacked.

It has the following additional parameters:

Column	3
Name	Groove Height
ID	height
Symbol	-
Editable	yes

If the *Groove Height* is zero, than a Fresnel Operator Surface is calculated, for which the sag is always zero and the normal of the surface is equal to the normal direction of the underlying surface. This of course is just an idealized version of a Fresnel lens with zero thickness. If *Groove Height* is set to a value larger than zero, the surface sag is calculated by the underlying surface sag modulo the Groove Height.
Note that the vertical edged that occur in between the optical surface areas are not modeled.

10 Phase Description

Phase Objects are used to describe a phase function $\phi(x, y)$ on a Surface. Such phase function can be used to model diffractive optical elements (DOEs) or to computer generated holograms (CGHs), to introduce phase errors in case of simulation, or any other task that can be solved by manipulating the phase of a wavefront. The unit of any amplitude coefficient is - if not explicitly mentioned - always in waves (λ). The default diffraction order that will be used to calculate the rays at a surface that has a Phase Object is $+1$. If another diffraction order should be used, the order can be explicitly specified using a User Defined Sequence. Like Form Objects, the Phase Objects can be stacked to generate a combination of several basic or polynomial Phase Objects. By this approach, an arbitrary combination of Form- and Phase objects can be realized. Furthermore, Phase Operators can be used to modify basic Phase Objects to e.g. generate arrays. If a Phase Objects is assigned to a Surface that is the boundary surface of an Element where the phase function is defined in waves, than the phase function is assumed to be on the outside of the element which means the wavelength for the calculation of the diffraction is the one in the environment. If it is assigned to the intermediate surface of e.g. an double lens than the wavelength of the material with the lower refractive index is used for the calculation.

All Phase Objects have the first three parameters in common:

Column	0	1	2
Name	Object Type	-	Comment
ID	type	-	comment
Editable	no	no	yes

10.1 Basic Phase Objects

10.1.1  Grating

The Grating Phase Object is used to describe a linear grating with the lines either oriented in x or y direction. The parameter *Lines per mm* is used to specify the number of linepairs of the grating per mm, which is equivalent to a phase value in waves per mm.

Column	3	4
Name	Lines per mm	Direction
ID	lines_per_mm	dir
Symbol	–	-
Editable	yes	yes

10.1.2  Grating 2D

The 2D Grating Phase Object is used to describe a 2D linear grating that emerges from overlaying two 1D linear gratings resulting in a rectangular phase array. The parameter *Lines per mm X* is used to specify the number of linepairs of the grating per mm along the X axis and the parameter *Lines per mm Y* is used to specify the number of linepairs of the grating per mm along the Y axis. In contrast to other phase functions the 2D Grating supports the selection of two independent diffraction orders.

Column	3	4
Name	Lines per mm X	Lines per mm Y
ID	lines_per_mm_x	lines_per_mm_y
Symbol	–	-
Editable	yes	yes

10.1.3  Radial

The Radial Phase Object is used to describe an aspheric, rotationally symmetric, phase function. It can be used to model Fresnel-Zone-Plates or rotationally symmetric CGHs like used to measure aspheric surfaces in nulltest condition.

Column	3	4	5	...	
Name	Norm Radius	Polynomial Order	A2	...	A2i
ID	normrad	maxorder	a2	...	a2i
Symbol	r_n	-	a_2	...	a_{2i}
Editable	yes	yes	yes	...	yes

With

$$h = \frac{\sqrt{x^2 + y^2}}{r_n} \tag{10.1}$$

the phase can be calculated by

$$\phi(x, y) = \sum_{i=1}^{n} A_{2\text{i}} \cdot h^{2i} \tag{10.2}$$

The unit of the coefficients is wavelengths.

10.1.4  Axial

The Axial Phase Object is similar to the Radial Phase Object. However, the phase value is not dependent on the radius, but only on one coordinate x or y. It can be used to generate a line focussing diffractive element or to model a CGH for the measurement of acylindric surfaces in nulltest condition.

Column	3	4	5	6	...	
Name	Norm Radius	Axis	Polynomial Order	A2	...	A2i
ID	normrad	axis	maxorder	a2	...	a2i
Symbol	r_n	-	-	a_2	...	a_{2i}
Editable	yes	yes	yes	yes	...	yes

if Axis is set to x, the phase is calculated to

$$\phi(x, y) = \sum_{i=1}^{n} A_{2\text{i}} \cdot (\frac{y}{r_n})^{2i} \tag{10.3}$$

if Axis is set to y, the phase is calculated by

$$\phi(x, y) = \sum_{i=1}^{n} A_{2\text{i}} \cdot (\frac{x}{r_n})^{2i} \tag{10.4}$$

The unit of the coefficients is wavelengths.

10.1.5 Spiral

The Spiral Phase Object is used to describe a spiral phase function, where the phase value is linearly dependent on the angular coordinate Φ when viewed in a polar coordinate system. Depending on the sign of the parameter *Waves per Revolution* the handiness of the phase function can be flipped.

Column	3
Name	Waves per Revolution
ID	waves
Symbol	W
Editable	yes

$$\Phi(x, y) = atan2(x, y) \tag{10.5}$$

$$\phi(x, y) = \frac{W \cdot \Phi}{2 \cdot \pi} \tag{10.6}$$

10.1.6 Grid

The Grid Phase Object is used to describe a phase function where the phase value is defined by a 2D lookup table. For the table the interpolation can be set to "Nearest Neighbor", "Linear" or "Bicubic". The unit of the coefficients is wavelengths.

10.1.7 Python

The Python Surface Phase Object is a general Phase object that can define arbitrary surface phase functions that are defined via a Python class. It has the following parameters

Column	3	4	5	6	7
Name	Class Name	Numeric Derivation	Param_1	...	Param_n
ID	cname	num_norm	-	-	-
Editable	yes	yes	yes	yes	yes

The *Class Name* parameter specifies the name of the Python class, that defines the phase function. Depending on the parameters defined in the Python class additional parameters are available. To make a user defined Python form available in Quadoa, the Python class must be added in a file

called python_phase_functions.py which is located in the python folder inside the Quadoa User Data Folder. The default path is `C:\Users\UserName\Quadoa\python` on Windows or `/home/UserName/Quadoa/python` on Linux.

Note that models with Python surfaces are not supported from within the Quadoa Python module or Matlab® toolbox or & C++ SDK.

To define a Python Form class the following methods have to be implemented:

- **getParameters()** returns an array that contains the parameter names
- **getZPhase()** returns a double that is the phase at a given X-, Y-coordinate $\phi = \phi(x, y)$
- **getPhaseGradient()** returns two double values that define the phase gradient at a given X-, Y-coordinate $\Delta x, \Delta y = grad(x, y)$

if *Numeric Derivation* is set to true the getPhaseGradient() method is not necessary, however the evaluation will gerally be slower.

An example implementation of a Python Form could look like this:

```
# Eample of a simple acylindric parabola
# To refer to  this definition the Class Name Param of the Python Forms
# has to be set to "Paraoloid"
class Paraoloid1D():
    # Method that tells Quadoa what parameters the PyhtonSurface has
    # Here the parameters that will be available for the surface type
    # need to be listed
    def getParameters():
        return ["P1"

    # Method to get Phi(x, y) of the surface (phase function)
    # Parameters declared in getParameters()
    # can be accesed by params["ParamName"]
    def getZPhase(x, y, params):
        r_square = y*y
        return params["P1"]*r_square

    # Method to get the phase gradient vector grad(x, y)
    # needs to return the gradient in x and y direction
    def getPhaseGradient(x, y, params):
        dx = 0.0
        dy = -2*params["P1"]*y
        return dx, dy
```

10.2 Polynomial Phase Objects

10.2.1 Zernike

The Zernike Surface Phase Object is a polynomial form object, that uses a Zernike Polynomial (see Section 26.1 for details) to describe the phase function. It has the following additional parameters:

Column	3	4	5	6	...	
Name	Norm Radius	Polynomial Order	Sort Order	Z2	...	Zn
ID	normrad	maxorder	sort_order	z1	...	zn
Symbol	r_n	-	-	Z_1	...	Z_n
Editable	yes	yes	yes	yes	...	yes

10.2.2 Zernike Fringe

The Zernike Fringe Surface Phase Object is a polynomial form object, that uses a Zernike Fringe Polynomial (see Section 26.1.2 for details) to describe the phase function. It has the following additional parameters:

Column	3	4	...	
Name	Norm Radius	Z2	...	Zn
ID	normrad	z1	...	zn
Symbol	r_n	Z_1	...	Z_n
Editable	yes	yes	...	yes

10.2.3 Poly XYZ

The XYZ Polynomial Surface Phase Object is a polynomial form object, that uses a XYZ Polynomials (see Section 26.2 for details) to describe the phase function. It has the following additional parameters:

Column	3	4	5	...	
Name	Norm Radius	Polynomial Order	X1Y0	...	XnYm
ID	normrad	maxorder	x1y0	...	xnym
Symbol	r_n	-	X_1Y_0	...	Y_nY_m
Editable	yes	yes	yes	...	yes

10.2.4 Phase Operator Objects

10.2.4.1 Phase Rect Array Operator

The Phase Rect Array Operator is used to generate arrays of surface phase functions. This is typically used to generate diffractive micro lens arrays. It can be used with any Phase object as well as a Phase stack. It can be stacked like any other Phase Object.

It has the following additional parameters:

Column	3	4	5	6
Name	Pitch X	Pitch Y	Decenter X	Decenter Y
ID	pitch_x	pitch_y	dec_x	dec_y
Symbol	-	-	-	-
Editable	yes	yes	yes	yes

The pitch values define the size of the single array cells in x and y direction, while the decenter values specify the shift in x and y direction.

10.2.4.2 Phase Hex Array Operator

The Phase Hex Array Operator is used to generate hexagonal arrays of surface phase functions. This is typically used to generate diffractive micro lens arrays. It can be used with any Phase object as well as a Phase stack. It can be stacked like any other Phase Object.

It has the following additional parameters:

Column	3	4	5
Name	Size	Decenter X	Decenter Y
ID	size	dec_x	dec_y
Symbol	-	-	-
Editable	yes	yes	yes

The *Size* value define the size of the regular hexagons edge lenght of a single array cells, while the *Decenter X* and *Decenter Y* values specify the shift in x and y direction.

10.2.4.3 Transformation Operator

The Phase Transformation Operator is used to generate decentered, rotate, enlarged or squeezed Phase Functions. It will apply the transformation to any Phase Objects that is added as child item.

It has the following additional parameters:

Column	3	4	5	6	7
Name	Decenter X	Decenter Y	Scale X	Scale Y	Rotate
ID	dec_x	dec_y	s_x	s_y	rot
Symbol	-	-	-	-	-
Editable	yes	yes	yes	yes	yes

The *Decenter* parameters move the X and Y coordinates inside the surface coordinate system. The *Scale* parameters applies a scaling operation along the X or Y axis. The *Rotation* parameter rotates the Phase Function around the Z-axis. The order the operations are applied is as follows: first the Phase Functions is translated, then the rotation is applied and finally it is scaled along the axis of the surface coordinate system. To achieve other orders for the operations several Phase Transformation Operators can be nested.

11 Aperture and Obscuration

Aperture and Obscuration objects are used to specify the Aperture on a Surface. If a surface has no aperture as child object, the default circular aperture, which is a parameter in the Surface, is used. The default aperture radius can be locked or unlocked. When unlocked, it will automatically update its size to form the smallest possible circle around all rays.

Note, that if the Surface is not part of any Sequence and the aperture is not locked, the radius of the floating aperture will be zero and therefore the Surface will not be visible.

If there are no additional Aperture objects, the default aperture defines the physical as well as the clear aperture of the Surface. This means, that in the 3D-View or when exporting an Element to a CAD file, the aperture radius is the size that is used for the mechanical dimensions of the object. The first Aperture or Obscuration added to a Surface replaces the default aperture and specifies the physical size of the Surface. If more then one apertures are specified, the additional apertures will define the clear aperture and further constraint the area where rays may pass the Surface.

11.1 Aperture

An Aperture allows any rays that lie inside its area to pass. Any rays outside the Aperture will be vignetted. The default Aperture will never block any rays as long it is unlocked (see chapter 3.3.3 on how to lock or unlock apertures).
Following Apertures are available:

11.1.1 ◯ Circular

The Circular Aperture is the most common type of Aperture. It is also used for the floating Aperture. The Circular Aperture has the following parameters:

Column	3	4	5
Name	Radius	Center X	Center Y
ID	radius	c_x	c_y
Editable	yes	yes	yes

The *Radius* specifies the radius of the Aperture and the *Center X* / *Center Y* parameters specify the center of the circle in the local Surface coordinate system.

11.1.2 Rectangular

The Rectangular Aperture has the following parameters:

Column	3	4	5	6
Name	Half Size X	Half Size Y	Center X	Center Y
ID	s_x	s_y	c_x	c_y
Editable	yes	yes	yes	yes

The *Half Size X* and *Half Size Y* specify the half size of the sides of the rectangle. The *Center X* and *Center Y* parameters specify the position of the center of the rectangle in the local Surface Coordinate System.

11.1.3 Elliptic

The Elliptic Aperture has following parameters:

Column	3	4	5	6
Name	Half Axis X	Half Axis Y	Center X	Center Y
ID	ha_x	ha_y	c_x	c_y
Editable	yes	yes	yes	yes

The *Half Axis X* and *Half Axis Y* specify the two half axes of the ellipse. The *Center X* and *Center Y* parameters specify the position of the center of the ellipse in the local Surface Coordinate System.

11.1.4 Annular

The Annular Aperture is like an Circular Aperture that has a central Obscuration. It is typically used to describe primary telescope mirrors that have a hole in the center. The Annular Aperture has the following parameters:

Column	3	4	5	6
Name	Inner Rad	Outer Rad	Center X	Center Y
ID	rad_in	rad_out	c_x	c_y
Editable	yes	yes	yes	yes

The *Inner Radius* and *Outer Radius* specify the lower and upper bound of the radius in which rays may pass. The *Center X* and *Center Y* parameters specify the position of the center of the rectangle in the local Surface Coordinate System.

11.1.5 Hexagonal

The Hexagonal Aperture defines an Aperture that is described by a regular hexagonal shape. has the following parameters:

Column	3	4	5
Name	Size	Center X	Center Y
ID	size	c_x	c_y
Editable	yes	yes	yes

The *Size* specifies the length of each of the sides of the hexagon, which is equal to the radius of the circumscribed circle. The *Center X* / *Center Y* parameters specify the center of the hexagon in the local Surface coordinate system.

11.2 Obscuration

An Obscuration allows any rays that lie outside its area to pass. Any rays on the inside will be vignetted. Following Obscurations are available.

11.2.1 Circular

The Circular Obscuration has the following parameters:

Column	3	4	5
Name	Radius	Center X	Center Y
ID	radius	c_x	c_y
Editable	yes	yes	yes

The Radius specifies the radius of the Obscuration and the *Center X* / *Center Y* parameters specify the center of the circle in the local Surface Coordinate System.

11.2.2 Rectangular

The Rectangular Obscuration has the following parameters:

Column	3	4	5	6
Name	Half Size X	Half Size Y	Center X	Center Y
ID	s_x	s_y	c_x	c_y
Editable	yes	yes	yes	yes

The *Half Size X* and *Half Size Y* specify the half size of the sides of the rectangle. The *Center X* and *Center Y* parameters specify the position of the center of the rectangle in the local Surface Coordinate System.

11.2.3 Elliptic

The Elliptic Obscuration has following parameters:

Column	3	4	5	6
Name	Half Axis X	Half Axis Y	Center X	Center Y
ID	ha_x	ha_y	c_x	c_y
Editable	yes	yes	yes	yes

The *Half Axis X* and *Half Axis Y* specify the two half axes of the ellipse. The *Center X* and *Center Y* parameters specify the position of the center of the ellipse in the local Surface Coordinate System.

11.3 Aperture Operators

An Aperture Operators allow to apply operations to Apertures but also Obscurations.

11.3.1 Array

The Aperture Array Operator is used to generate an rectangular linear array of Aperture objects. It has the following parameters:

Column	3	4	5	6	7	8
Name	Count X	Count Y	Pitch X	Pitch Y	Decenter X	Decenter Y
ID	count_x	count_y	pitch_x	pitch_y	dec_x	dec_y
Editable	yes	yes	yes	yes	yes	yes

The *Count* parameters specify the number of rows and columns of the array. The *Pitch* parameter is used to define the pitch of the array in X- and Y-direction. The *Center* parameters allow to decenter the array inside the local surface coordinate system.

12 Coatings

Coating objects define the transmission and reflection of a Surface. They are assumed to have no thickness, and have no effects on the refraction of the rays (or in general the path a ray takes) at a Surface. Depending on the type of the Coating, the reflection and transmission may depend on the Wavelength, the incident angle of the ray as well as the Polarization of the ray. If a Surface has no Coating, the reflectance and trasmittance of a Surface are calculated according to the Fresnel equations for the glass air interface. In this case the reflectance is typically around 4% and depends on the incident angel as well as the Polarization of the ray. If no Polarization is defined, the ray is assumed to be unpolarized, resulting in a reflectance that is equal to the average of a horizontally and a vertically polarized ray. A Surface always can only have one Coating assigned to it.

All Coating Objects have the first three parameters in common:

Column	0	1	2
Name	Object Type	-	Comment
ID	type	-	comment
Editable	no	no	yes

- *Object Type* is a non editable parameter that describes the basic type of the item.
- *Comment* is a free string parameter that can be used for comments and documentation.

12.1 Simple Ideal Coatings

The simple coatings reflect or transmit a ray in an idealized way no matter what incident angle or polarization the ray has.

12.1.1 Ideal AR

The Ideal AR coating always transmits 100% of the rays and reflects 0%.

12.1.2 Ideal Mirror

The Ideal Mirror coating always transmits 0% of the rays and reflects 100%.

12.1.3 Beam Splitter

The Beam Splitter coating has the following parameter:

Column	3
Name	Reflection
ID	rfx
Symbol	α
Editable	yes

The *Reflection* parameter defines the reflectance of the coating. The transmittance therefore is $t = 1.0 - r$. Besides beam splitters, simple anti-reflection coatings that are not dependent of the Wavelength or the incident angle of the ray can be modeled with this coating type.

12.2 Interpolated Coatings

Interpolated Coatings use a linear or 2D-bilinear interpolation on the coating data provided. They can be used to model coatings, where only the transmittance / reflectance data is available from the manufacturer, but the thinfilm coating stack is not disclosed. The Interpolated Coatings do not have any additional parameters. The interpolation table can be entered by clicking on the + button on the first column in the *Optical Design Editor*.

12.2.1 Wavelength Dependent

The Wavelength Dependent coating reflects a certain amount of the light depending on the Wavelength of the incoming ray. The incident angle as well as the polarization do not influence the reflectance of this coating. The Wavelength at each data point is specified in nm and has to be sorted in ascending order. The spacing can, but does not have to lie on a equally spaced grid. The reflectance (r) can have a value between 0.0 and 1.0. The transmittance is calculated by $t = 1.0 - r$. If a Wavelength lies outside the specified data range available in the interpolation table, the intensity of the reflected as well as the transmitted ray will be set to NaN.

12.2.2 Incident Angle Dependent

The Incident Angle Dependent coating reflects a certain amount of the light depending on the incident angle of the incoming ray. The wavelength as well as the polarization do not influence the reflectance of this coating. The incident angle at each data point is specified in ° and has to be sorted in ascending order. The possible range is in between $0.0°$ and $90.0°$ The spacing can, but does not have to lie on a equally spaced grid. The reflectance (r) can have a value between 0.0 and 1.0. The transmittance is calculated by $t = 1.0 - r$. If a ray has an incident angle that lies outside the specified data range available in the interpolation table, the intensity of the reflected as well as the transmitted ray will be set to NaN.

12.2.3 Incident Angle and Wavelength Dependent

The Incident Angle and Wavelength Dependent coating reflects a certain amount of the light depending on the incident angle as well as the wavelength of the incoming ray. The polarization does not influence the reflectance of this coating. The incident angle at each data point is specified in ° and has to be sorted in ascending order. The possible range is in between $0.0°$ and $90.0°$. The Wavelength at each data point is specified in nm and has to be sorted in ascending order as well.

The spacing for both values can, but does not have to lie on a equally spaced grid. The reflectance (r) can have a value between 0.0 and 1.0. The transmittance is calculated by $t = 1.0 - r$. If a ray has an incident angle or a Wavelength, that lies outside the specified data range available in the interpolation table, the intensity of the reflected as well as the transmitted ray will be set to NaN.

12.2.4 Pupil Position Dependant

The Pupil Position Dependent coating reflects a certain amount of the light depending on the rays position on the surface. The wavelength, incidence angle and polarization do not influence the reflectance of this coating. The X and Y sample points for the interpolation table have to be specified in mm and in an ascending order. The reflectance (r) can have a value between 0.0 and 1.0. The transmittance is calculated by $t = 1.0 - r$ To generate periodic patters the pattern can either be directly entered in the interpolation table. However for patters with a large number of repetitions it is recommended to use the Pupil Position Dependent coating in combination with the Array Operator.

12.3 Thin Film Stack Coatings

The thin Film Stack Coatings are used to model coatings that are calculated via the transfer matrix method.

12.3.1 From Catalog

The From Catalog coating is used to refer to a coating that is defined in the *Coating Catalog*.

12.3.2 Thin Film

The Thin Film Coating allows to define a thin film stack directly inside the model. The advantage of defining the stack in the model is, that the layers can be modified e.g. for optimization or tolerancing of the coating stack. In most other cases it is recommended to define the coating in the *Coating Catalog* and refer to it using the From Catalog coating. The coating is defined by a set of layers, where for each layer the thickness and material is specified. For any interface - material to air or air to material - the first layer in the list is the interface layer that is interfacing to the lens material. The last layer is the top layer of the coating stack interfacing to the environment. If the coating is applied inside a kitted element, e.g. between the two parts of a doublet lens, the first layer is interfacing to the material of the first part of the lens and the last layer is the interface to the second part of the doublet. For the calculation of the optical path length of a system the coating is assumed to be infinitely thin and therefore does not contribute. Also for the calculation of the size of an element the coating is ignored.

12.3.3 Uniform Layer

The Uniform Layer is used to define a single layer of the Thin Film coating. It has the following parameters:

Column	3	4	5	6
Name	Thickness / nm	Type	Name	Catalog
ID	thic	model_type	name	ctlg
Editable	yes	yes	yes	yes

12.4 Attenuators

12.4.1  Constant

The Constant Attenuator attenuates a certain amount of light a and transmits $t = 1.0 - a$ It has the following additional parameter:

Column	3
Name	Attenuation
ID	att
Symbol	a
Editable	yes

12.4.2  Gaussian

The Gaussian Attenuator attenuates light depending on the position of the surface where the local attenuation coefficient is described by a Gaussian distribution. The typical application is to generate a flat-top intensity distribution out of a system that has a non uniform pupil apodization. It has the following additional parameters:

Column	3	4	5	6	7
Name	Sigma	Background Att.	Peak Att.	Decenter X	Decenter Y
ID	sigma	at_bg	at_peak	d_x	d_y
Symbol	σ	A_b	A_p	-	-
Editable	yes	yes	yes	yes	yes

The attenuation a is calculated by

$$a(x, y) = A_b + (A_p - A_b) \cdot e^{-\frac{(x^2+y^2)}{2\sigma^2}} \tag{12.1}$$

the transmits $t = 1.0 - a$.

12.4.3  Periodic

The Periodic Attenuator attenuates light depending on the position of the surface where the local attenuation coefficient is described by a cosine function. It has the following additional parameters:

Column	3	4	5	6	7
Name	Periode Length	Min Att.	Max Att.	Phase Offset	Axis
ID	periode	min	max	phase_offset	axis
Symbol	p	A_{min}	A_{max}	ϕ	-
Editable	yes	yes	yes	yes	yes

If *Axis* is set to y the attenuation a is calculated by

$$a(x,y) = A_{min} + (A_{max} - A_{min}) \cdot \cos(\frac{2\pi}{p} \cdot (y - \phi)) \tag{12.2}$$

else if *Axis* is set to x

$$a(x,y) = A_{min} + (A_{max} - A_{min}) \cdot \cos(\frac{2\pi}{p} \cdot (x - \phi)) \tag{12.3}$$

the transmits $t = 1.0 - a$.

12.5 Coating Operators

12.5.1 Array Operator

The Coating Array Operator is used to generate an rectangular linear array of Coating objects. It has the following parameters:

Column	3	4	5	6
Name	Pitch X	Pitch Y	Decenter X	Decenter Y
ID	pitch_x	pitch_y	dec_x	dec_y
Editable	yes	yes	yes	yes

The *Count* parameters specify the number of rows and columns of the array. The *Pitch* parameter is used to define the pitch of the array in X- and Y-direction. The *Center* parameters allow to decenter the array inside the local surface coordinate system. The Coating Array Operator only has an effect, if the coating transmittance / reflectance is dependent of the X-, Y-position of the ray. This is e.g. the case for the Pupil Position Dependent coating as well as the position dependent Attenuators.

13 Polarization Elements

Polarization Elements are used to manipulate the polarization state of a ray. They can be added to any Surface. See chapter 15 to learn how the polarization raytracing is computed.

All Polarization Elements have the first six parameters in common:

Column	0	1	2	3	4	5
Name	Object Type	-	Comment	Surface Action	Coo. Reference	Rz
ID	type	-	comment	action	ref	rz
Editable	no	no	yes	yes	yes	yes

- *Object Type* is a non editable parameter that describes the basic type of the item.
- *Comment* is a free string parameter that can be used for comments and documentation.
- *Surface Action* defines for which ray/surface interaction the Polarization Element is applied. It can be set to "All", "Transmission" or "Reflection". If the Polarization Element is explicitly defined in an User Defined Sequence this settings does not have any effect.
- *Coo. Reference* Defines the reference coordinate system in which the polarization matrix is applied. See chapter 15.4 for the definition of the possible settings.
- *Rz*: Polarization Element are generally defined with their main (fast) axis along the local surface x-axis. *Rz* defines the angle, the polarization matrix coordinate system is rotated around the local z-axis in respect to the local Surface coordinate system.

13.1 Static polarization elements

13.1.1  Linear Polarizer

The Linear Polarizer has the following additional parameters:

Column	6	7
Name	Tx	Ty
ID	tx	ty
Editable	yes	yes

It only passes the component of the electric field that lies parallel to the axis of the polarizer. To model an ideal polarizer, one of the two values for *Tx* or *Ty* is set to 1.0 and the other value is set to 0.0. To model a non-ideal polarizer, the loss can be modeled by setting the value to a number smaller than 1.0. Furthermore, the non perfect extinction of the light that passes the polarizer with a polarization that lies orthogonal to the axis of the polarizer can be modeled by setting the sectiond value to a number greater than 0.0.

The Jones matrix for the polarizer is

$$\begin{bmatrix} T_x & 0 \\ 0 & T_y \end{bmatrix} \tag{13.1}$$

13.1.2 Circular Polarizer

The Circular Polarizer has the following additional parameters:

Column	6
Name	Handedness
ID	hand
Editable	yes

If *Handedness* is set to "Left", the Jones matrix of the polarizer is defined as

$$\frac{1}{2}\begin{bmatrix} 1 & -i \\ i & 1 \end{bmatrix} \tag{13.2}$$

If *Handedness* is set to "Right", the Jones matrix of the polarizer is defined as

$$\frac{1}{2}\begin{bmatrix} 1 & i \\ -i & 1 \end{bmatrix} \tag{13.3}$$

13.1.3 Quarter-wave plate

The Quarter Wave Plate has no additional parameters. When not rotated the fast axis is aligned horizontally along the x-axis. The Jones matrix of the Quarter Wave Plate is defined as

$$e^{-\frac{i\pi}{4}} \begin{bmatrix} 1 & 0 \\ 0 & i \end{bmatrix} \tag{13.4}$$

13.1.4 Half-wave plate

The Half Wave Plate has no additional parameters. When not rotated, the fast axis is aligned horizontally along the x-axis. The Jones matrix of the Half Wave Plate is defined as

$$e^{-\frac{i\pi}{2}} \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \tag{13.5}$$

13.1.5 General Retarder

The General Retarder has the following additional parameter:

Column	6
Name	Retardation
ID	ret
Editable	yes

Where *Reatardation* defines the retardation in waves. The Jones matrix of the General Retarder is defined as

$$e^{-\frac{ir}{2}} \begin{bmatrix} 1 & 0 \\ 0 & e^{ir} \end{bmatrix} \tag{13.6}$$

13.1.6 General Jones Matrix

The General Jones Matrix has the following additional parameters:

Column	6	7	8	9	10	11	12	13
Name	J00r	J00i	J01r	J01i	J10r	J10i	J11r	J11i
ID	exx	ecxx	exy	ecxy	eyx	ecyx	eyy	ecyy
Editable	yes	yes	yes	yes	yes	yes	yes	yes

The Jones Matrix is directly composed from the parameters specifying the real and imaginary part for each matrix field and is defined as

$$\begin{bmatrix} J_{00r} + iJ_{00i} & J_{01r} + iJ_{01i} \\ J_{10r} + iJ_{10i} & J_{11r} + iJ_{11i} \end{bmatrix} \tag{13.7}$$

13.2 Interpolated Polarization Elements

Interpolated Polarization Elements use a linear or 2D-bilinear interpolation on the polarization data provided. The Interpolated Polarization Elements do not have any additional parameters. The interpolation table can be entered by clicking on the + button on the first column in the *Optical Design Editor*.

13.2.1 Wavelength Dependent Retarder

The Wavelength Dependent Retarder is used to model non-achromatic retarders, where the retardance typically depends on the Wavelength of the ray. The Wavelength at each data point is specified in nm and has to be sorted in ascending order. The spacing can, but does not have to lie on an equally spaced grid. The retardance is specified in waves. The interpolated value will be calculated for each incident angle and is used as input for the calculation of the Jones matrix which is the same as for the General Retarder

13.2.2 Incident Angle Dependent Retarder

The Incident Angle Dependent Retarder defines a retarder, where the retardance depends on the angle of incident of the ray. This is typically the case for any retarder, since the optical path length through the birefringent plan-parallel material - and therefore the retardance - depends on the angle of incident. The angle of incidence at each data point is specified in $circ$ and has to be sorted in ascending order. The spacing can, but does not have to lie on a equally spaced grid. The retardance is specified in waves. The interpolated value will be calculated for each wavelength and is used as input for the calculation of the Jones matrix, which is the same as for the General Retarder

13.2.3 Pupil Position Dependent Retarder

The Pupil Position Dependent Retarder defines a retarder, where the retardance depends on the position of the ray inside the Surface coordinate system. The position date for the x- and y-position is specified in mm and has to be sorted in ascending order. The spacing can, but does not have to lie on a equally spaced grid. The retardance is specified in waves. The interpolated value will be calculated for each position and is used as input for the calculation of the Jones matrix, which is the same as for the General Retarder

13.2.4 Wavelength and Incident Angle Dependent Retarder

The Wavelength and Incident Angle Dependent Retarder defines a retarder, where the retardance depends on the angle of incident as well as the Wavelength of the ray. The angle of incidence at each data point is specified in $circ$ and has to be sorted in ascending order. The Wavelength at each data point is specified in nm and has to be sorted in ascending order as well. The spacing can, but does not have to lie on a equally spaced grid. The retardance is specified in waves. The interpolated value will be calculated for each incident angle and wavelength and is used as input for the calculation of the Jones matrix, which is the same as for the General Retarder

13.2.5 Pupil Position and Wavelength Dependent Retarder

The Pupil Position and Wavelength Dependent Retarder defines a retarder, where the retardance depends on the position of the ray inside the Surface coordinate system as well as the Wavelength. The position date for the x- and y-position is specified in mm and has to be sorted in ascending order. The Wavelength at each data point is specified in nm and has to be sorted in ascending order as well. The spacing can, but does not have to lie on a equally spaced grid. The retardance is specified in waves. The interpolated value will be calculated for each position and wavelength combination and is used as input for the calculation of the Jones matrix, which is the same as for the General Retarder.

13.3 Mueller Matrix Element

The Mueller Matrix Element allows to directly input a Mueller matrix. It has the following additional parameters:

Column	6	7	8	9	10	11	12	13
Name	M00	M01	M02	M03	M10	M11	M12	M13
ID	M00	M01	M02	M03	M10	M11	M12	M13
Editable	yes	yes	yes	yes	yes	yes	yes	yes

Column	14	15	16	17	18	19	20	21
Name	M20	M21	M22	M23	M30	M31	M32	M33
ID	M20	M21	M22	M23	M30	M31	M32	M33
Editable	yes	yes	yes	yes	yes	yes	yes	yes

The Mueller Matrix is defined as

$$\begin{bmatrix} M_{00} & M_{01} & M_{02} & M_{03} \\ M_{10} & M_{11} & M_{12} & M_{13} \\ M_{20} & M_{21} & M_{22} & M_{23} \\ M_{30} & M_{31} & M_{32} & M_{33} \end{bmatrix} \tag{13.8}$$

Note: Mueller matrix elements are only applied if the Polarized Source is set to Stokes.

13.4 Scatter

Scatter objects can be used to simulate the effects of surface roughness or other effects that lead to scattering of rays. If an Surface has a scatter item assigned the ray-tracing calculation at the interface will first calculate the transmission, reflection or diffraction as usual and afterwards randomly change the ray direction according to the scatter property. In the *3D-View* the effect of the scatter on the rays is directly visualized. To see the effect in any of the analysis features as e.g. the *Spot Diagram* or the *Irradiance Plot* scattering has to explicitly be enabled in the plot settings. This option in only available in some plots. Scattering is not supported in Sequences with *Aperture Type* set to "From Stop". Most scatter models have a parameter *Scatter Probability*. This parameter allows to specify the probability a ray will scatter. Fig. 13.1 illustrates how exactly the directional change of the ray when scattered is defined. The red arrow show the specularly reflected (unscattered) ray path and the vector β_0 is the projection of the reflected ray onto the tangential plane to the surface. The orange arrow shows the scattered ray and the vector β is the projection of this vector to the tangential plane to the surface. Both vectors the specular and the scattered ray are unit vectors. Vector x is defined as $x = \beta - \beta_0$. The random length of this random vector x is what is defined by the probability distribution of the scatter objects.

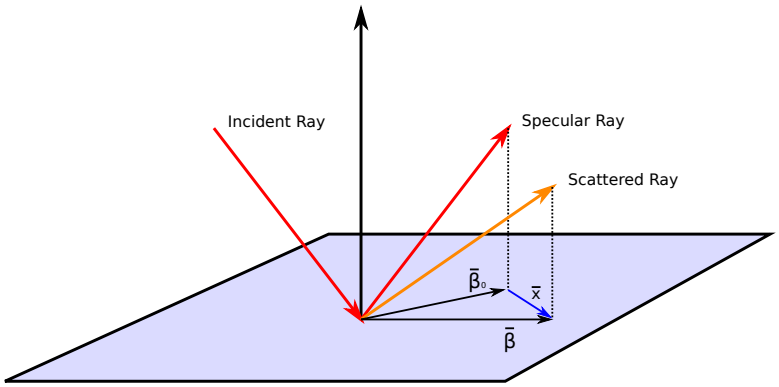


Figure 13.1: Definition of the vectors involved in the scattering calculation

13.4.1 Gaussian

The Gaussian Scatter defined a scatter distribution based in a Gaussian distribution.

Column	3	4
Name	Sigma	Scatter Probability
ID	sigma	prob
Editable	yes	yes

Parameter *Sigma* defines the standard deviation of the scatter distribution and parameter *Scatter Probability* defines the probability of a ray to scatter.

$$P(x,y) = \frac{a}{2\pi\sigma^2} \cdot e^{-\frac{x^2+y^2}{2\sigma^2}} \tag{13.9}$$

where x and y are the components of the $\vec{x}$.

13.4.2 Lambertian

The Lambertian Scatter defined a scatter distribution following the Lambertian cosine law. The probability for a scattered rays direction is therefore independent of the incident rays direction. Parameter *Scatter Probability* defines the probability of a ray to scatter.

Column	3
Name	Scatter Probability
ID	prob
Editable	yes

13.4.3 ABg

The ABg Scatter defined a scatter distribution according to the ABg formula. For this type of scatter the probability of a ray to scatter is always 1.0 since the distribution is defined in a way that already contains the secular rays.

Column	3	4	4
Name	A	B	g
ID	a	b	g
Editable	yes	yes	yes

$$P(x,y) = \frac{A}{B + |\vec{x}|^g} \tag{13.10}$$

Note: Depending on the values for A B and g the evaluation of the random distribution may be extremely costly leading to low ray tracing performance. Therefore, for scatter distributions where this is the case it is recommended to use Gaussian Scatter instead.

14 Sequential Raytracing

In Quadoa Optical CAD, the sequencial path to trace the light is strictly separated from the description of the optical System. To define a sequential raytrace, the Sequence Object is used. It allows the user to define an arbitrary number of sequences inside a single model, which is useful e.g. for ghost analysis, modeling systems with more than one optical path like e.g. interferometers, setting up and designing parts of complex systems step by step and many other fields.

14.1 Sequence

The Sequence is the basic object that is used to define a sequential ray trace through the system. It defines the sequential path for the rays traced through the system. Furthermore, the properties of the rays like Fields, Wavelengths the Aperture Type as well as any other ray properties are defined via the Source object of each Sequence. The Sequence has following parameters:

Column	1	2	3	4
Name	Name	Aperture Type	Field Type	Stop Surface
ID	name	aperturetype	fieldtype	stopsurface
Editable	yes	yes	yes	yes

Column	5	6	7	8
Name	OPD Reference	OPD Chief Reference	Image Scpace	Ban Virtual
ID	opd_ref	opd_chiefref	img_space_type	no_virt
Editable	yes	yes	yes	yes

- *Name*: This parameter is used as an human readable identifier for the Sequence and has no affect on the behavior at all.
- *Aperture Type*: The *Aperture Type* is used to define how the Aperture of the Sequence is defined. Following options are available:
  - "Object Space NA" will define the numerical aperture of the system in object space. The chiefray of each field will always go through the center of the Stop Surface - except

when the center for a Field is shifted via the vignetting factors (see chapter 14.1.11.1). This is the recommended Aperture Type to use if speed is important, since it is a lot faster compared to *Aperture Type* "From Stop". However for large field angles or when large aberrations are introduced in the system before the Stop Surface, the rays may be distorted at the Stop Surface leading to a non unitary sampling.

- "Entrance Pupil" will define the radius of the entrance pupil. This option is available for both, "Point" and "Plane Wave Front" sources. The chiefray of each field will always go through the center of the Stop Surface - except when the center for a Field is shifted via the vignetting factors (see chapter 14.1.11.1). This is the recommended Aperture Type to use if speed is important, since it is a lot faster compared to *Aperture Type* "From Stop". However for large field angles or when large aberrations are introduced in the system before the Stop Surface, the rays may be distorted at the Stop Surface leading to a non unitary sampling.
- "From Stop" Will generate the ray Distribution to perfectly cover the aperture of the stop surface. The numerical aperture or beam diameter in object space is automatically computed, so that the rays fill the whole aperture of the Stop Surface. Therefore it is not possible to have a Stop Surface without a fixed Aperture defined. As for *Aperture Type* "Object Space", the ray distribution on the Stop Surface can be shifted and scaled via the Vignetting Factors of each Field.

  Note, that the computation will take longer compared to Aperture Type "Object Space", since the rays need to be aimed to the coordinates on the Stop Surface. Therefore it is recommended to use Aperture Type "Object Space" whenever possible.
- "None" means, that the rays are traced totally independent from the stop surface. If selected, the Parameter *Stop Surface* is not available. Here the user can define all parameters like tilt and decenter of every Field totally free. The application of this *Aperture Type* is primary meant for illumination applications and not classical imaging optics.

  Note, that for this Aperture Type not all Analysis Plots or Optimization Goals are available, since some values like e.g. Distortion, Field Curvature, etc. do not have a meaning when the Stop Surface is not defined.

- *Field Type*: The *Field Type* is used to define how the field value is defined.
  - "Obj. Height" The field is defined by the position of the rays on the first Surface in the Sequence. This feature is only available for point like sources, since height does not have a meaning in case of a plane wavefront.

  - "Angle" The field is defined by the angle in angle units at the first Surface in the Sequence.
  - "Img. Heigth" The field is defined by the position of the chief ray on the last Surface of the Sequence. This feature only should be used for focal systems. The setting is not available for sequences with *Aperture Type* set to "None".
  - "Free" This setting is only available for systems with *Aperture Type* set to "None". In this case both, the fields tilt and angle, may be set by the user.
  - "None" For sequences where the *Source Type* is set to "From Surface" no Fields are available. In this case, the *Field Type* will be automatically set to None.

- *Stop Surface*: This parameter is used, to define which Surface in the Sequence should be the Stop Surface. The chief ray of each field will always go through the center of the Stop Surface, except when the central position for a Field is shifted via the Vignetting Factors. The parameter is not available if *Aperture Type* is set to "None", since this type of Source is independent of a Stop Surface definition. For User Sequences, the Stop Surface can also directly be specified in the Surface List. To Change the Stop Surface, right click on the surface you want to make Stop and select Make Stop.
- *OPD Reference*: The OPD Reference parameter specifies how the Optical Path Length (OPL) and Optical Path Difference (OPD) of a system is calculated. Following scenarios will help to understand what the parameter does:
  - "Exit Pupil:" The system is an imaging lens or some other kind of system where the rays are in focus on the image surface. In this case most probably the setting "Exit Pupil" is to be preferred. If "Exit Pupil" is selected, the rays are traced to the Exit Pupil and the OPD is plotted over the position in the pupil. In this case, the OPD are the deviation of the optical path from a perfect spherical wavefront in the exit pupil, which is the desired form for an aberration free focus.
  - "Afocal Exit Pupil:" The system is an afocal systems where the rays are collimated in image space. In this case most probably the setting "Afocal Exit Pupil" is to be preferred. If "Afocal Exit Pupil" is selected, the rays are traced to the Exit Pupil and the OPD is plotted over the position in the pupil. In this case, the OPD are the deviation of the optical path from a perfect plane wavefront in the exit pupil, where the tilt of the plane wavefront is equal to the tilt of the chief ray.
  - "Abs over Img:" This settings should be used when simulating e.g. wavefront-sensors, interference or other systems, where the central focus is to simulate the wavefront at the

last surface. In the plots, the lateral coordinates (x, y) represent the position of the ray on this surface and the OPD is the absolute OPL of the ray minus the OPL of the chief ray. If the system is focal or there is a caustic zone in the region of the last surface, this may lead to self-intersecting OPD Plots or Wave Front Plots, since in the presence of caustics the definition of a wavefront is physically no longer meaningful.

  - "Abs over EP:" In this case - as for the setting "Exit Pupil:" - the lateral x, y-position of the rays is analyzed in the exit pupil. However the OPD is calculated by the total absolute OPL of the ray, from the first to the last surface minus the OPL of the chief ray. This setting is probably not what is desired in most cases, however it has been implemented to be compatible with other optical design software.

- *OPD Chief Reference*: The *OPD Chief Reference* is used to define the reference ray, that is used to define the exit pupil location. This option is only available for Sequences with *OPD Reference* set to "Exit Pupil" or "Afocal Exit Pupil".

  - "Primary" If "Primary" is selected the chief ray of the primary wavelength is used to define the location of the exit pupil. This is the recommended setting for most imaging systems. Lateral chromatic aberration will be visible as tilt in OPD Plots or Wave Front Plots.

  - "Individual" If "Individual" is selected the chief ray of each wavelength is used to define the location of the exit pupil. For each wavelength a separate exit pupil will be computed. This setting can be used for systems like e.g. spectrometers, where the different wavelengths are observed independently. In the OPD Plots or Wave Front Plots the tilt of the wavefront will always be zero at the chief rays location.

- *Ban Virtual*: The parameter *Ban Virtual* allows to ban any virtual rays. The main purpose for this feature is to force the ray aiming towards the desired solution in the case of systems where two rays - one real, one virtual - provide a valid solution to the ray aiming search. (See chapter 5.4 on the definition of virtual rays)

  Note: during optimization it is recommended to disable this option. Since it vignettes any virtual ray the optimizer will be prevented from temporary finding a solution with intersecting lenses or any other configurations that contain virtual rays. However during optimization such a non physical system can still be useful, e.g. to get to a neighboring valley of the meritfunction, that contains a better physical solution and that could not be reached without crossing the region with non-physical, virtual rays

When a Sequence is defined over a range of Surfaces the Material in between any of two following Surfaces, that do not belong to the same Element or Lens is assumed to be Air at the temperature

and pressured defined by the environment. This is also true for Surfaces that are located at the exact same position. To model two elements that are kitted without an air-gap the Doublet or any other n-let lens has to be used. There are different kind of Sequence objects available:

14.1.1 Auto Sequence

The Auto Sequence is the default type of Sequence. For many classical systems - like objective lenses, imaging optics, beam shaping, etc. - where only a single optical path of the model is of primary interest and no beam splitters or double passes through lenses exist, the Auto Sequence is the Sequence of choice. The Auto Sequence will trace the rays starting at the first Surface in the *Optical Design Editor* from top to bottom ending at the last Surface. The action that will be performed at the Surface - if it has no Coating - is transmission. If the Coating has a reflectance over 50%, the default action is reflection. If the Surface has any Phase Property Objects, the first order of diffraction will be used. If more than one Polarization Elements are defined for a Surface, the selection of the element to be applied, is defined by the settings of the Polarization Elements (see chapter 15.5). To change the default behavior the Auto Sequence has to be converted to an User Sequence. This can by done by right-clicking on the Auto Sequence and select Convert to User Sequence or by selecting it and clicking the To User Sequence button in the Sequence tab.

14.1.2 User Sequence

The User Sequence is - as the Auto Sequence - used to define a sequential raytrace. In addition to the Auto Sequence, it allows to specify the order of surfaces for the rays to be traced through the Surface List. Besides specifying the order of the surfaces to be traced, the exact action at each Surface - e.g. if the rays are transmitted or reflected, what diffraction order or polarization element is applied, etc. - can be specified (see chapter 14.1.2.1). To convert an User Sequence to an Auto Sequence, right click on the Auto Sequence and select Convert to Auto Sequence or select the To Auto Sequence button in the Sequence tab.

Note: If converting an User Sequence to an Auto Sequence, the user defined specification, as the ray path through the Surface List, will be set to default settings and will be deleted.

14.1.2.1 Surface List

The Surface List in a User Sequence is used to specify the order of the Surfaces for the sequential raytrace. Furthermore, the action at each Surface - e.g. if the rays are transmitted or reflected, what diffraction order or polarization element is applied, etc. - can be specified. The rays are traced from top to the bottom of the Surface List. To add a Surface to the Surface List, following possibilities exist: Select the User Sequence and press the Add Surface button under the *Sequence Tab* or right click on the User Sequence or the Surface List and select Add Surface or press the Insert key. To change the order of the Surfaces, they can be moved by drag and drop. To remove a Surface, right click on it and select Delete, or select it and press delete on your keyboard. The Surface object in the Surface List has the following parameters:

Column	1	2	3	4	5	6
Name	Identifier	Surface Nr.	Surface Action	Diffraction Order	Jones Matrix	Reflect On
ID	is	surf	action	diff_order	jones	refl_mat
Editable	yes	yes	yes	yes	yes	yes

- *Identifier* is used to select the Surface that is defined inside the *Optical Design Editor* via its *Identifier* (see chapter 6.2). If a Surface belongs to a Lens or an Custom element, then the Surfaces is specified by the Identifier of the Element in combination with the number of the Surfaces inside the Element which is specified by the *Surface Nr.*. If the Surface is an element of its own, that does not belong to an higher level object (parent object), it has its own *Identifier* and can directly be selected.
- *Surface Nr.* is used to select the surface inside the Element specified by the *Identifier*. (1 is the first surface, 2 the second surface and so on, when counting the surfaces from top to bottom). This parameter is not available when the *Identifier* directly refers to a Surface that is not part of a higher level object (parent object).
- *Surface Action* specifies the action that is performed at a Surface. For the first and last Surface in the list, this parameter is not available. The *Surface Action* can be either "Transmission" or "Reflection".
- *Diffraction Order* specifies the diffraction order to be traced, in case the Surface has any Phase Properties defined. For Surfaces without Phase Properties, this parameter is not available.
- *Diffraction Order 2* specifies the second diffraction order to be traced, in case the Surface has any Phase Properties defined that allows to specify a second diffraction order (e.g. the 2D Grating. For Surfaces Phase Properties, or with phase objects that do not have a second diffraction order this parameter is not available.

- *Jones Matrix* can be used to select a specific Jones Matrix. How to work with independent Polarization Elements based on the *Surface Action* or in different Sequences see chapter 15.5. For Surfaces without Polarizers, this parameter is not available.
- *Reflect On* specifies the Material that is assumed to be "behind" the Surface when reflecting a ray. For Lenses this parameter will be set automatically. For Custom Elements the parameter has to be set by the user. This parameter is only available when *Surface Action* is set to "Reflection".

14.1.3 Source

Every Sequence contains a Source object as the first of its child items. The Source object is used to define the properties of the traced rays, including the Wavelengths, the Fields, the Polarization, the Apodization as well as the Distribution of the rays.

The Source has following parameters:

Column	1	2
Name	Source Type	Aperture Radius Numerical Aperture
ID	sourcetype	aperture_val
Editable	yes	yes

- *Source Type*: This parameter is used to specify the starting condition of the rays. There are following options:
  - "Point Source:" in this case all the rays start at a single point at the first Surface. This option is used for designing imaging systems, where the object is in a finite distance or to simulate a perfect point source.
  - "Plane WF" this setting can be used for imaging systems that image an object that is infinitely far away or in general to simulate an infinitely far away point source, which will cause the wavefront to be plane.
  - "WF from Surface" this setting can be used to simulate wavefront of an arbitrary shape, where the form of the wavefront can be specified by the Form of the first Surface in the Sequence. This setting is useful to simulate wavefronts, that have been measured e.g. by a Shack-Hartman-Sensor or an interferometer or to design null-optics for optical testing.

  - "Extended" allows to simulate an extended source. For the extended source, the Ray Distributions that are available, differ from those available for classical point-like or plane sources. Besides simple extended shapes, also models for Gaussian Beams and patterns like, e.g. grids are available. (see chapter 14.1.6)
  - "From Ray File" This setting allows to simulated light sources that are specified by a ray file. Ray files can be obtained for a wide range of LEDs, light bulbs, Lasers and other sources directly from the manufacturer. See chapter 25.3.1 on the supported Ray File Formats. Users can also generate their own Ray Files for more advanced simulations. See chapter 25.3.1.1 on how to set up a custom Ray File. To make a ray file available inside Quadoa the file has to be placed in the same folder the model is stored in. It then will be available via the *Ray File* parameters drop down menu in the Distribution.
  - "Gaussian Beam" This setting will trace a Gaussian beam using the skew ray method proposed by P. D. Colbourne[1]. For the Gaussian Beam source type several Distributions exist (See 14.1.7). The beam is visualized as a solid beam in the 3D-View and the beam properties can be analyzed using the *Gaussian Beam Report*.

    Note, that when "Gaussian Beam" is selected as Source Type most analysis plots will no longer be available.

- *Aperture Radius / Numerical Aperture:* If *Aperture Type* is set to "Object Space" or "None", this parameter allows the specification of the aperture of the rays. If *Aperture Type* is set to "From Stop", the aperture in object space is automatically computed directly from the diameter of the Stop Surface Aperture. If the *Source Type* is set to "Point Source" or "Extended", the Numerical Aperture (NA) in object space can be set. If *Aperture Type* is set to "Plane WF" or "From Surface", the radius of the aperture in object space can be specified. For *Source Type* "Ray File", the aperture is directly defined by the rays in the file and therefore the parameter is not available in that case.

14.1.4 Distribution

The distribution and amount of the traced rays is specified by the Distribution object. The Distribution is always the first child of the Source object in every Sequence. It has the following parameters:

Column	1	2	3	4
Name	Distribution	Ray Count 1	Ray Count 2 (optional)	Footprint Type (optional)
ID	disttype	rays	rays_2	footprinttype
Editable	yes	yes	yes	yes

Depending on the value of *Distribution*, the second and third parameters might vary or not be present. The parameter *Footprint Type* is only available for Sources with *Source Type* "Point Source" or "Plane WF" and simultaneously with the *Aperture Type* of the Sequence set to "Object Space NA" or "Entrance Pupil". It defines the way how Fields are applied. If the parameter is set to "On Plane", the rays will be distributed undistorted when looked at in a plane parallel to the object surface, that is intersecting the cone or cylinder that is formed by the rays. If the parameter is set to "On Sphere", the rays will be distributed without distortion when looked at on the surface of a sphere that intersects the cone or cylinder formed by the rays, where the center of the sphere lies at the origin of the rays.

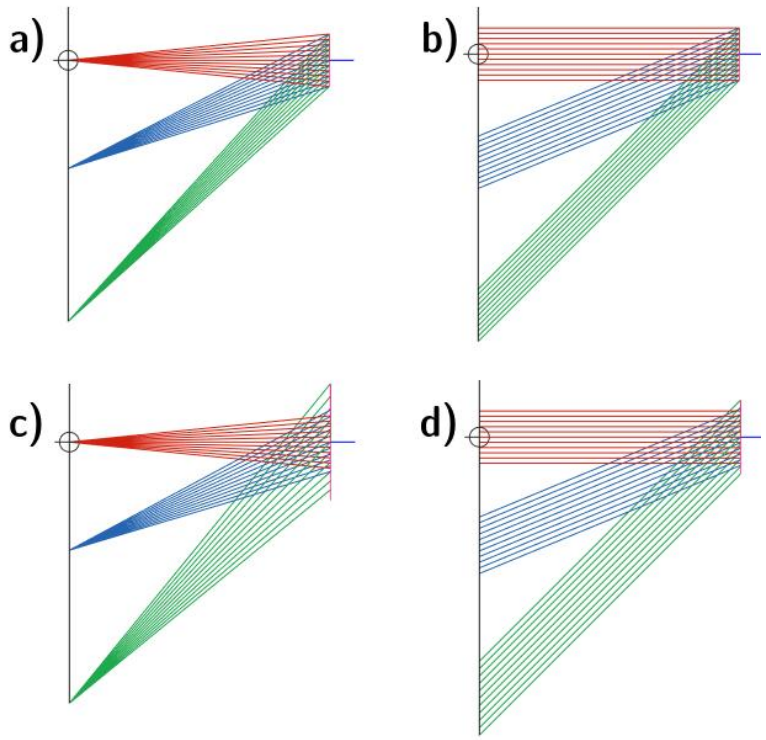


Figure 14.1: Footprint Reference "On Plane": For point source (a) and for plane wave front (b) the beam profile at the entrance pupil has a constant size for all fields. However the NA of the beam or in case of the plane wavefront the diameter is not constant any more. Footprint Reference "On Sphere": The NA (c) or the beam diameter (d) is constant for any field, however the beam profile is not constant any more when looked at the cross-section at the pupil plane.

The effect of this setting is only significant if large Fields are applied. As a rule of thumb it can be assumed, that for imaging systems "On Plane" is typically the right setting, since the entrance pupil lies in a plane that is typically parallel to the object surface. For illumination systems, where tilted light sources with a certain NA are simulated, "On Sphere" is typically the best choice. Furthermore, for ghost analysis it is also recommended to set the value to "On Sphere" which is the default setting

when using the *Ghost Wizard*. This way it can be assured, that the ray flux per area that enter the first aperture is equal for all fields of the ghost.

14.1.5 Available Ray Distributions

For simple - non extended - Sources the following distributions are available. (For extended Sources see chapter 14.1.6.

- "Single Ray" generates a single ray. The coordinate of the ray at the Stop Surface can be specified by the parameters *Pupil X* and *Pupil Y*.
- "Y-Fan" generates a fan of rays spread out in Y-direction. The number of rays can be specified by the parameter *Ray Nr.*
- "X-Fan" generates a fan of rays spread out in X-direction. The number of rays can be specified by the parameter *Ray Nr.*
- "XY-Fan" generates two fans of rays spread out in X- and Y-direction. The number of rays is counted as rays per fan and is specified by the parameter *Ray Nr.*
- "Ring" generates a single ring of rays that lie on a circle at the border of the aperture. The number of rays can be specified by the parameter *Ray Nr.*
- "Equal Rings" generates a set of concentric rings, where the spacing of the rays on each ring is equal to the distance of the rings. The number of rings can be specified by the parameter *Rings*.
- "Conc Rings" generates a set of concentric rings, where the amount of rays is equal for each ring resulting in an hub-like pattern. The parameter *Rings* allows to specify the number of rings. The parameter *Segments* allows to specify the number of rays on each ring.
- "Hex Grid" generates a hexagonal honeycomb pattern. The parameter *Grid Count* allows to specify the number of rays in horizontal direction of the pattern.
- "Opt Grid" generates a grid which is optimized to achieve a good sampling of the aperture, with a minimum amount of rays in case of rotationally symmetric systems. For more details see [3]
- "Rect Grid" generates an in X- and Y-direction equally spaced rectangular grid. The parameter *Grid Count* allows to specify the number of rays in each direction of the pattern.

- "Random" generates random rays while the position of rays is equally distributed over the whole aperture. The number of rays can be specified by the parameter *Ray Nr.*
- "Quasi Random" generates random rays with the density of rays being distributed following a quasi random distribution. For Monte Carlo sampling e.g. when computing the irradiance or ray density, this distribution is to be preferred over the "Random" one, since it typically results in less noise per number of rays. The number of rays can be specified by the parameter *Ray Nr.*. The real number of rays that are generated may slightly differ from the specified number.
- "Ray File" If the Source Type is set to "From Ray File", the distribution is automatically set to "Ray File". The maximum number of rays can be specified by the parameter "Ray Nr.". If the value of "Ray Nr." is larger than the number of rays available in the provided ray file, the number of traced rays will be reduced to the maximum available rays, no matter what the parameter is set to.
- "Gaussian Beam" If the Source Type is set to "Gaussian Beam", the distribution is automatically set to "Gaussian Beam". The parameter *Waist* specifies the size of the beam waist, *Waist Pos Z* specifies the position of the waist relative to the position of the first surface and *M2* specified the M^2 beam quality factor of the beam. The beam will be visualized as a solid beam in the 3D-View and the beam properties can be analyzed using the *Gaussian Beam Report*.

14.1.6 Ray Distributions for Extended Sources

If the Source Type is set to "Extended" a different set of distributions is available. The distribution of the rays inside the extended area, as well as inside the NA, is always random. The *Angular Distribution* can be set to "Lambertian", "Uniform", "Gaussian", "Super Gauss", "Gaussian 2D", "Super Gauss Elliptic" or "Super Gauss Rectangular", exept for the Gaussian Beam and Fiber type distribution where the angular distribution is specidied by other parameters. Following extended source types are available for the lateral distibution of the rays:

- "Disk" generates an extended disk-like area, where all rays lie within a circle and are equally distributed over the as well as in the angular range. The radius is defined by parameter *Radius*. The *Angular Distribution* can be set to "Lambertian", "Uniform", "Gaussian", "Super Gauss", "Gaussian 2D", "Super Gauss Elliptic" or "Super Gauss Rectangular".
- "Disk Gaussian" generates an extended disk-like area, where the ray density in the x, y. plane is defined by a Gaussian normal distribution. The *Angular Distribution* can be set to

"Lambertian", "Uniform", "Gaussian", "Super Gauss", "Gaussian 2D", "Super Gauss Elliptic" or "Super Gauss Rectangular".

- "Gaussian Beam" generates a Gaussian beam like distribution where the distribution of the rays is lateral and angular space is a Gaussian normal distribution. This i similar to the "Pos(Sigma) and NA(Sigma)" distribution, however the values for the standard deviation are taken from the *Waist* and *M2* parameters of the Gaussian Beam instead.
- "Gaussian Beam Elliptic" generates an astigmatic Gaussian beam like distribution. The basic principle is the same as for the "Gaussian Beam" distribution, however the waist (specified by *Waist Y* and *Waist X*) and M^2 values (specified by *M^2 Y* and *M^2 X*) can be specified independently in x and y direction. Furthermore, the position of the beam waist can be shifted along the optical axis independently for each direction via the parameters *Waist dz Y* and *Waist dz X*.
- "Fiber SM (NA)" generates a Gaussian beam like distribution as it exits from a single mode fiber, where the fiber is specified by the $\frac{1}{e^2}$ NA by the parameter *NA (1/e^2)*.
- "Fiber SM (NA) " generates a Gaussian beam like distribution as it exits from a single mode fiber, where the fiber is specified by the $\frac{1}{e^2}$ mode field diameter by the parameter *MFD (1/e^2)*. The unit of this parameter is μm.
- "Fiber MM" generates a extended distribution as it exits from a multi mode fiber, where the fiber is specified by the $\frac{1}{e^2}$ NA by the parameter *NA (1/e^2)* as well as the core diameter of the fiber by the parameter *Core Diameter*. The unit of this parameter is μm and intensity distribution inside the fiber core is assumed to follow a top-hat profile.
- "Rect" generates an extended rectangular area. The size of the rectangle is defined by parameter *Size X* and *Size Y*. The *Angular Distribution* can be set to "Lambertian", "Uniform", "Gaussian", "Super Gauss", "Gaussian 2D", "Super Gauss Elliptic" or "Super Gauss Rectangular".
- "Linepair H" generates two rectangular horizontal lines, that are separated by a void area. The size of the whole pattern is defined by parameter *Size X* and *Size Y*. The spacing of the lines is equal to the width of the lines and therefore is $\frac{1}{3}$ of *Size Y*. The *Angular Distribution* can be set to "Lambertian", "Uniform", "Gaussian", "Super Gauss", "Gaussian 2D", "Super Gauss Elliptic" or "Super Gauss Rectangular".
- "Linepair V" generates two rectangular vertical lines, that are separated by a void area. The size of the whole pattern is defined by parameter *Size X* and *Size Y*. The spacing of the lines is equal to the width of the lines and therefore is $\frac{1}{3}$ of *Size X*. The *Angular Distribution* can be

set to "Lambertian", "Uniform", "Gaussian", "Super Gauss", "Gaussian 2D", "Super Gauss Elliptic" or "Super Gauss Rectangular".

- "Linegrid H" generates 10 rectangular horizontal lines, that are separated by 9 void areas. The size of the whole pattern is defined by parameter *Size X* and *Size Y*. The spacing of the lines is equal to the width of the lines and therefore is $\frac{1}{19}$ of *Size Y*. The *Angular Distribution* can be set to "Lambertian", "Uniform", "Gaussian", "Super Gauss", "Gaussian 2D", "Super Gauss Elliptic" or "Super Gauss Rectangular".
- "Linegrid V" generates 10 rectangular vertical lines, that are separated by 9 void areas. The size of the whole pattern is defined by parameter *Size X* and *Size Y*. The spacing of the lines is equal to the width of the lines and therefore is $\frac{1}{19}$ of *Size X*. The *Angular Distribution* can be set to "Lambertian", "Uniform", "Gaussian", "Super Gauss", "Gaussian 2D", "Super Gauss Elliptic" or "Super Gauss Rectangular".
- "Rect Grid" generates 10x10 rectangular areas, that are separated by 9 void areas in each direction. The size of the whole pattern is defined by parameter *Size X* and *Size Y*. The spacing of the lines is equal to the width of the lines. The size of the single features therefore is $\frac{1}{19}$ of *Size X* in horizontal and $\frac{1}{19}$ of *Size Y* in vertical direction. The *Angular Distribution* can be set to "Lambertian", "Uniform", "Gaussian", "Super Gauss", "Gaussian 2D", "Super Gauss Elliptic" or "Super Gauss Rectangular".
- "Siemens Star" generates a Siemens Star pattern with 12 arms filled with rays and 12 void arms in between of the same size which cover an angular area of $15°$. The size of the Siemens Star is defined by parameter *Radius*. The *Angular Distribution* can be set to "Lambertian", "Uniform", "Gaussian", "Super Gauss", "Gaussian 2D", "Super Gauss Elliptic" or "Super Gauss Rectangular".

Following distributions are available for the angular density distribution of the rays:

- "Lambertian" Lambertian distribution of the rays within a cone. The maximum cone angle can be defined via *Max NA*.
- "Uniform" Uniform distribution of rays over the entrance pupil. In most cases this is not the correct setting for an extended source, since a typical black body emitter emits rays according to Lamberts cosine law. The maximum cone angle can be defined via *Max NA*.
- "Gauss" generates rays, where the ray density follows a Gaussian distribution over the NA according to:

$$F(x,y) = e^{-\frac{x^2+y^2}{2\sigma^2}} \tag{14.1}$$

- "Gauss 2D" generates rays, where the ray density follows a 2D elliptical Gaussian distribution over the NA according to:

$$F(x,y) = e^{-(\frac{x^2}{2\sigma_x^2}+\frac{y^2}{2\sigma_y^2})} \tag{14.2}$$

- "Super Gauss" generates rays, where the ray density follows a Super Gaussian distribution over the NA according to:

$$F(x,y) = e^{-(\frac{x^2+y^2}{2\sigma^2})^P} \tag{14.3}$$

- "Super Gauss Elliptic" generates rays, where the ray density follows a Elliptic Super Gaussian distribution over the NA according to:

$$F(x,y) = a \cdot e^{-(\frac{x^2}{2\sigma_x^2}+\frac{y^2}{2\sigma_y^2})^P} \tag{14.4}$$

- "Super Gauss Rectangular" generates rays, where the ray density follows a Rectangular Super Gaussian distribution over the NA according to:

$$F(x,y) = a \cdot e^{-((\frac{x^2}{2\sigma_x^2})^P+(\frac{y^2}{2\sigma_y^2})^P)} \tag{14.5}$$

14.1.7 Ray Distributions for Gaussian Beam Sources

If the Source Type is set to "Gaussian Beam" following distributions are available.

- "Circular" generates a rotationally symmetric Gaussian Beam. The parameter *Waist* defines the size of the waist. The parameter *M^2* defines the M^2 beam quality factor of the beam.
- "Elliptic" generates an elliptic Gaussian Beam where the beam waist and M^2 factor can be specified independently along the x and y axis. The parameter *Waist Y* and *Waist X* define the size of the waist in x and y direction. The parameters *M^2 Y* and *M^2 X* define the M^2 beam quality factor of the beam in each direction.
- "Fiber SM (NA)" generates a rotationally symmetric Gaussian Beam similar as with the "Circular" distribution, however the beam is specified via the $\frac{1}{e^2}$ NA of the fiber by the parameter *NA (1/e^2)*.

- "Fiber SM (MFD)" generates a rotationally symmetric Gaussian Beam similar as with the "Circular" distribution, however the beam is specified via the $\frac{1}{e^2}$ mode field diameter of the fiber by the parameter *MFD (1/e^2)*. The unit of this parameter is μm.

14.1.8 Apodization

The Apodization is used to specify the distribution of the intensity over the pupil, which is applied by changing the flux of the rays depending of their position on the pupil. It has an effect on any computation, where the flux of a ray is used, e.g. Irradiance Plots, Point Spread Functions or MTF-Analysis. The Apodization has the following parameters:

Column	1	2	3
Name	Type	Power	...
ID	apod_type	power	...
Editable	yes	yes	yes

- *Type* specifies the Apodization Type, meaning the type of the distribution of the flux over the pupil. The sum of all the rays flux per field and wavelength is equal to the power specified in the Source. Following Apodization Types are available:
  - "Constant" generates each ray with the same flux. The value of the flux of a single ray is the power of the Source divided by the number of rays.
  - "Gauss" generates rays, where the flux of each ray is calculated from a Gaussian distribution over the pupil. If *Norm Pupil Coordinates* is enabled, *Sigma* is defined in normed pupil coordinates, else *Sigma* is defined in mm in the entrance pupil plane.

$$F(x,y) = e^{-\frac{x^2+y^2}{2\sigma^2}} \tag{14.6}$$

  - "Gauss 2D" generates rays, where the flux of each ray is calculated from a elliptic 2D-Gaussian distribution over the pupil. If *Norm Pupil Coordinates* is enabled, the *Sigma* parameters are defined in normed pupil coordinates, else the *Sigma* parameters are defined in mm in the entrance pupil plane.

$$F(x,y) = e^{-(\frac{x^2}{2\sigma_x^2}+\frac{y^2}{2\sigma_y^2})} \tag{14.7}$$

  - "Super Gauss" generates rays, where the flux of each ray is calculated from a Super Gaussian distribution over the pupil. If *Norm Pupil Coordinates* is enabled, *Sigma* is defined in normed pupil coordinates, else *Sigma* is defined in mm in the entrance pupil plane.

$$F(x,y) = e^{-(\frac{x^2+y^2}{2\sigma^2})^P} \tag{14.8}$$

– "Super Gauss Elliptic" generates rays, where the flux of each ray is calculated from a elliptic 2D-Super Gaussian distribution over the pupil. If *Norm Pupil Coordinates* is enabled, the *Sigma* parameters are defined in normed pupil coordinates, else the *Sigma* parameters are defined in mm in the entrance pupil plane.

$$F(x,y) = a \cdot e^{-(\frac{x^2}{2\sigma_x^2}+\frac{y^2}{2\sigma_y^2})^P} \tag{14.9}$$

– "Super Gauss Rectangular" generates rays, where the flux of each ray is calculated from a rectangular 2D-Super Gaussian distribution over the pupil. If *Norm Pupil Coordinates* is enabled, the *Sigma* parameters are defined in normed pupil coordinates, else the *Sigma* parameters are defined in mm in the entrance pupil plane.

$$F(x,y) = a \cdot e^{-((\frac{x^2}{2\sigma_x^2})^P+(\frac{y^2}{2\sigma_y^2})^P)} \tag{14.10}$$

- *Power* specifies the total power (sum of ray fluxes) in Watts that will be set for each field / wavelength combination.
- *Norm Pupil Coordinates* switches between *Sigma* being defined in normed pupil coordinates or mm in the entracen pupil plane in the case of a non-constant apodizations.
- *Sigma*, *Sigma X*, *Sigma Y* specifies the standard deviation in the case of a Gaussian distributed apodizations.
- *Exponent*, *Exponent X*, *Exponent Y* specifies the exponent P in the case of a Super Gaussian distributed apodizations.

14.1.9 Polarization

The Polarization object is used to define the polarization properties of the rays that are used for Polarization Raytracing (see chapter 15). It has the following parameters

Column	1	2	3 - 6
Name	Type	Reference	depends on Type
ID	pol_type	pol_ref	...
Editable	yes	yes	yes

Following *Polarization Types* are available:

- "None" disables polarization raytracing. In this case only a simple raytrace will be performed, that doesn't take into account any polarization effects. The transmittance and reflectance of Coatings will be computed by assuming the light is unpolarized, which is equivalent to the average of a horizontally and a vertically polarized ray.
- "Linear" generates rays, that are liner polarized along an axis that can be specified by the *Angle* parameter in degrees. An *Angle* of $0.0°$ represents a horizontal polarization. For the resulting Jones vector See chapter 15.2.
- "Circular" generates rays, that are circular polarized. The *Rotation* parameter can be set to "Left" or Right to specify a left or right handed circular polarization. For the resulting Jones vector See chapter 15.2.
- "Jones Complex" generates a polarization state, that is defined by the Jones vector that can directly be specified by the user. This allows to generate an arbitrary - elliptical - polarization state. For the x-part of the vector, the parameters are *EX* and *EcX* for the complex part. For the y-part the parameters are *EY* and *EcY*.
- "Jones Phase" generates a polarization state, that is defined by a Jones vector that can directly be specified by the user. Contrary to the "Jones Complex" option, the state is not entered as real and complex numbers but as the amplitude and phase of each component. For the x-part, the parameters are of the vector are *Amplitude X* and *Phase X*. For the y-part, the parameters are *Amplitude Y* and *Phase Y*. The phase is specified in Waves.
- "Stokes" generates a polarization state, that is defined by the Stokes vector that can be directly be specified. For the Stokes vector the four parameters *I*, *O*, *U* and *V* are available. An alternate notation for the same values is *S0*, *S1*, *S2* and *s3* (Both Notations are displayed as the parameter name). See chapter 15.2.2 for how polarization with a Stokes source is computed.
- "Random" generates a random - in general elliptically - polarization for each ray. This will result in light that behaves - if sampled with a large enough number of rays - as unpolarized light. However, unlike with *Polarization Type* set to "None", polarization raytracing is not disabled. Therefore it is possible to start with unpolarized light and use e.g. a Linear Polarizer or any other Polarization Element to polarize it. Another option is to use *Polarization Type* "Stokes" which also allows the simulation of partially polarized light.

The parameter *Reference* specifies the Reference Coordinate System in which the polarization state is defined. See chapter 15.4 on how the reference is defined.

14.1.10 λ Wavelength

14.1.10.1 λ Wavelength Container

The Wavelength Container is used is the parent item, that contains the single wavelength definitions of a Source. The Wavelength Container has following parameters:

Column	1	2
Name	Primary	Reference
ID	prim	wave_ref
Editable	yes	yes

- *Primary* specifies which wavelength to use as the primary wavelength. The primary wavelength is used to get basic system parameters e.g. in the *System Report*, but also serves as reference for aberrations, e.g. for *Ray Fan Plots*. (See also the parameter *OPD Chief Reference* in Sequence. For polychromatic imaging systems, typically the central wavelength is a good choice as primary wavelength.
- *Reference* specifies if the wavelengths are considered to be measured in the Environment or in vacuum.

14.1.10.2 λ Wavelength

The Wavelength is used to specify the wavelengths of the rays. Each source can have one or more wavelength defined. The Wavelength has following parameters:

Column	1	2
Name	Lambda	Weight
ID	val	wght
Editable	yes	yes

- *Lambda* specifies the wavelenght in nm
- *Weight* specifies the weight of a wavelength. The *Weight* parameter is used to specify the relative weight of a wavelenght, when optimizing a system (see chapter 18.5). By right clicking on the Wavlenghts, a menu will appear that allows to select common wavelengths as well as adding wavelength combinations.

14.1.11 Field

The Field object is used to specify a field coordinate of the rays. A newly created Sequence already has one Field. To add additional fields, select the Sequence and click Add Field under the Sequence tab or right click on the Sequence or Source and select Add Field. Depending on the Field Type, different parameters are available. If the *Field Type* is set to "Obj. Heigth" or "Img. Heigt", the decenter in x- and y-direction can be specified in length units. If the *Field Type* is set to "Angle", the tilt of the incoming wavefront can be specified in °. If *Field Type* is set to "Free", both, the decenter and tilt, can be specified. The Field has following parameters:

Column	1	2	3	4	5
Name	Tilt X	Tilt Y	Decenter X	Decenter Y	Weight
ID	t_x	t_y	d_x	d_x	wght
Editable	yes	yes	yes	yes	yes

Column	6	7	8	9
Name	Vig. X	Vig. Y	Vig Dec. X	Vig Dec. Y
ID	vig_x	vig_y	vig_dec_x	vig_dec_y
Editable	yes	yes	yes	yes

The tilt parameters *Tilt X* and *Tilt Y* are used to specify the tilt of the Field in degrees in case the *Field Type* is "Angle" or "Free". For other *Field Types* they are not available.
The decenter parameters *Decenter X* and *Decenter Y* are used to specify the object height of the Field in case the *Field Type* is "Obj. Height", or the image height in case the *Field Type* is "Img. Height" in lenght units. For other *Field Types* they are not available.
The *Weight* parameter is used to specify the relative weight of a Field when optimizing a system (see chapter 18.5). Outside the optimization this parameter has no effect.

14.1.11.1 Vignetting Factors

The Vignetting Factors are used to tune the distribution of the rays independently for each Field, which is important for optimization. If a Field gets vignetted by some apertures in the system, only a fraction of the rays - if no Vignetting Factors are applied - can be traced. This leads to a bad sampling of the pupil and therefore to poor optimization results. The Vignetting Factors allow to compress and/or shift the distribution of the rays to achieve a better sampling of the vignetted pupil in the Field. Also most plots will compute their result based on the adjusted fields. The exception is the Vignetting over Field plot that shows the Vignetting of the unmodified rays. Furthermore, the PSF and MTF plots will also use the real system apertures (including the real vignetting at any

aperture) for their computation. For *Aperture Type* "None" Vignetting Factors are not available. Following parameters are available:

- Vig. X: Scaling of the ray distribution in X-direction. The default value is set to 1.0 and will have no effect on the ray distribution. This parameter will scale the X-coordinate of the ray in X-direction. For Sequences with *Aperture Type* "From Stop", the scaling will be applied on the Stop Surface. For Sequences with *Aperture Type* set to "Object Space", the scaling will be applied in Object Space scaling either the angular distribution or the position on the first Surface.
- Vig. Y: Scaling of the ray distribution in Y-direction. The default value is set to 1.0 and will have no effect on the ray distribution. This parameter will scale the Y-coordinate of the ray in Y-direction. For Sequences with *Aperture Type* "From Stop", the scaling will be applied on the Stop Surface. For Sequences with *Aperture Type* set to "Object Space", the scaling will be applied in Object Space scaling either the angular distribution or the position on the first Surface.
- Vig. Dec. X: This parameter will shift the chief ray on the stop surface in X-direction by the parameters value. The default is 0.0 and does not have any effect.
- Vig. Dec. Y: This parameter will shift the chief ray on the stop surface in Y-direction by the parameters value. The default is 0.0 and does not have any effect.

15 Polarization Raytracing

In addition to basic Sequential Raytracing, Quadoa is able to perform Polarization Raytracing. For Polarization Raytracing, the polarization state of the rays can be specified via the Polarization Parameters in the Source object for each Sequence. For the initial polarization state either a Jones Vector or a Stokes Vector can be used. If a polarized source is defined, all intensity data for analysis is calculated from the rays electrical field. Any ray-surface interactions will be computed considering the corresponding mechanisms. By default, the Fresnel-equations are applied. If a Coating or a Polarization Element is specified for the Surface instead of the Fresnel equations for the air/glass or glass/glass interface, the specified Polarization Element or the specified Coating defines the mechanism for the computation.

15.1 Polarization Coordinate systems

All Jones calculus operations are performed in the rays plane of polarization, perpendicular to the direction of propagation ($\vec{k}$). Since all Jones operations on a Sequence are defined per Surface (source, interaction or analysis surface), each rays local coordinate system($\vec{x}_j, \vec{y}_j, \vec{z}_j$) must be derived form the Surfaces coordinate system($\vec{x}, \vec{y}, \vec{z}$). Several options are available to do this:

- **Ideal:** The local Surface Coordinate System is rotated around $\vec{k} \times \vec{z}$ by the angle α between $\vec{k}$ and $\vec{z}$ by the rotation matrix $R(\alpha)$. This is equivalent to 'bending' the ray in its electrical field into a collimated state perpendicular to the surfaces x/y-plane, applying the Jones operation in that plane and 'bending' the ray back to its original state.

$$\vec{x}_j = R \cdot \vec{x}, \quad \vec{y}_j = R \cdot \vec{y}, \quad \vec{z}_j = R \cdot \vec{z} = \vec{k} \tag{15.1}$$

- **X-Axis Reference:** The local ray x-axis lies in the $\vec{x}/\vec{k}$ plane, perpendicular to $\vec{k}$

$$\vec{y}_j = \vec{k} \times \vec{x}, \quad \vec{x}_j = \vec{y}_j \times \vec{k}, \quad \vec{z}_j = \vec{k} \tag{15.2}$$

- **Y-Axis Reference:** The local ray y-axis lies in the $\vec{y}/\vec{k}$ plane, perpendicular to $\vec{k}$

$$\vec{x}_j = \vec{y} \times \vec{k}, \quad \vec{y}_j = \vec{k} \times \vec{x}_j, \quad \vec{z}_j = \vec{k} \tag{15.3}$$

- **S/P:** The local ray x-axis is perpendicular to the plane of incidence (defined by surface normal $\vec{n}$ and ray direction $\vec{k}$), the local ray y-axis lies within the plane of incidence

$$\vec{x}_j = \vec{k} \times \vec{n}, \quad \vec{y}_j = \vec{k} \times \vec{y}_j, \quad \vec{z}_j = \vec{k} \tag{15.4}$$

15.2 Source Definition

A polarized Source can be characterized in terms of a 2D Jones vector ($\vec{J}$) or via a 4D Stokes vector ($\vec{S}$). The Jones or Stokes vector is applied in a coordinate system perpendicular to a rays direction of propagation ($\vec{k}$). Each rays coordinate system is derived from the local Surface Coordinate System system($\vec{x}, \vec{y}, \vec{z}$). Quadoa Optical CAD offers several options to define these local ray coordinate systems (15.1).

15.2.1 Jones Vector

Several options to define the polarization of a source via Jones Vector are available.

Note, that the vector will always be normalized to $\|\vec{J}\| = 1$. The specification of the type and state of the polarization is set by the Polarization Parameters in the Source object for each Sequence.

- **Linear:** Linear polarization with orientation determined by angle α to the x-axis

$$\vec{J} = \begin{pmatrix} \cos\alpha \\ \sin\alpha \end{pmatrix} \tag{15.5}$$

- **Circular:** Circular polarization left- or right-handed

$$\vec{J} = \frac{1}{sqrt(2)} \begin{pmatrix} 1 \\ i \end{pmatrix} \tag{15.6}$$

$$\vec{J} = \frac{1}{sqrt(2)} \begin{pmatrix} 1 \\ -i \end{pmatrix} \tag{15.7}$$

- **Jones Complex:** Real and imaginary components of the Jones vector are entered directly

$$\vec{J} = \begin{pmatrix} E_x + i \cdot Ec_x \\ E_y + i \cdot Ec_y \end{pmatrix} \tag{15.8}$$

- **Jones Phase:** Jones vector is defined by amplitude and phase [waves] of the components

$$\vec{J} = \begin{pmatrix} A_x \cdot e^{i \cdot 2\pi p_x} \\ A_y \cdot e^{i \cdot 2\pi p_y} \end{pmatrix} \tag{15.9}$$

- **Random:** The polarization state for each ray is generated randomly. This results in a general elliptic polarization where the ellipticity, the orientation of the ellipse as well es the handedness are totally random for each ray. Each random state is normalized to $\|\vec{J}\| = 1$.

15.2.2 Stokes Vector

To allow for mixed polarization states a polarized source can be definied via Stokes Parameters.

$$\vec{S} = \begin{pmatrix} I \\ Q \\ U \\ V \end{pmatrix} = \begin{pmatrix} S_0 \\ S_1 \\ S_2 \\ S_3 \end{pmatrix} \tag{15.10}$$

Note, that the vector will always be normalized to $I = S_0 = 1$.

Note: When using the Stokes source definition, the phase of the electrical field is not retained even though Quadoa Optical CAD will still display phase information in some analysis tools.

15.2.3 Unpolarized

The polarization type "none" disables polarization ray tracing. To apply polarization ray tracing to an unpolarized source, the Stokes definition can be used

$$\vec{S} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} \tag{15.11}$$

15.3 Sequence Polarization Elements

Any Surface can have one or more Polarization Element. However only one Polarization Element will be applied during any Surface interaction within a Sequence (see chapter 15.5 o how to specify which Polarization Element will be used). Polarization elements are generally defined with their main (fast) axis along the local surface x-axis. A rotation of the Polarization Element coordinate system in regard to the local surface coordinate system around the local z-axis can be defined with *Rz*.

If the polarized source of a Sequence is defined via Stokes parameters, any Jones Matrix surface element will be internally converted to its corresponding Mueller Matrix surface element. For Jones sources any Mueller Matrix surface elements will be **ignored**.

15.4 Coordinate Reference

For general Polarization Elements, a choice of coordinate references is available (see 15.1). The Axis reference option refers to the Polarization Elements x-axis, determined by the local surface orientation and *Rz*.

Quadoa offers an addional option here for compatibility. This is not a rigorous application of the Jones or Mueller calculus and should be used with caution.

- **Projection:** The Jones or Mueller matrix is defined in the local Surface Coordinate Systems x/y-plane. This means that parts of the electrical field will be ignored depending on its angle to the local z-axis.

$$\begin{pmatrix} E'_x \\ E'_y \end{pmatrix} = M \cdot \begin{pmatrix} E_x \\ E_y \end{pmatrix} \tag{15.12}$$

  The remaining E_z field component is determined by the boundary condition:

$$\vec{E}' \cdot \vec{k} = 0 \tag{15.13}$$

  The magnitude is then renormalized to

$$\|\vec{E}'\| = \| \begin{pmatrix} E'_x \\ E'_y \\ E_z \end{pmatrix} \| \tag{15.14}$$

  to make sure the outgoing intensity is not amplified.

15.5 Working with more than one Polarization Element on a Surface

The default *Surface Action* ("Reflection" / "Transmission" / "All") for which a Polarization Element is used, is defined by the *Surface Action* parameter of the Polarization Element. If only one Polarization Element is provided for a Surface and the *Surface Action* specified in the Polarization Element is the same as the one defined in the Sequence, the default Fresnel equation surface interaction will be replaced by the polarization operation defined by the element - else the default Fresnel equations will be applied. If there are more than one Polarization Elements, following two possibilities exist:

15.5.1 Independent Polarization Elements for Transmission and Reflection

The following behavior applies, if the *Surface Action* inside the Surface List for the Surface is set to "Auto" and does not explicitly select another Polarization Element:

- "Transmission": By setting the Polarization Element *Surface Action* parameter to "Transmission", the polarization operation will always be applied whenever rays are transmitted at the Surface.
- "Reflection": By setting the Polarization Elements *Surface Action* parameter to "Reflection", the polarization operation will always be applied whenever rays are reflected at the Surface.
- "All": If the *Surface Action* is set to "All", the polarization operation will always be applied, no matter if the rays are transmitted or reflected.

15.5.2 User Selection of Polarization Elements

The *Surface Action* inside the Surface List for a Surface can be used to explicitly select a Polarization Element. If a specific Polarization Element is selected, the default *Surface Action* that is defined inside the element is ignored for this interaction.

16 Lookup Parameters and Lookup Tables

Lookup Tables allow to define parameters that can be used in the optical model. There exist two versions of lookup tables. The Multiconfig Lookup Table as well as the Slider Interface.

16.0.1 Lookup Parameters

The Multiconfig Lookup Table as well as the Slider Interface can hold different types of Lookup Parameters. The first column of all lookup parameters holds the parameter name. The parameter name is a unique name, that is used to reference the Lookup Parameter from parameters of the optical model, a Sequence, or from following Lookup Parameter inside the Lookup Table. The parameter name always has to begin with an upper or lower case letter and may contain besides letters as well numbers. To reference to a Lookup Parameter from another parameter, the value of the parameter has to be replaced with the name of the Lookup Parameter to reference to. If the Lookup Parameter can be found, the current value will be displayed in brackets besides the parameter name. If the Lookup Parameter does not exist, the item will be highlighted in red and an error message will appear.

16.1 Multiconfig

The Multiconfig Lookup Table allows to define optical systems, that have more than one discrete state, which means that some parameters in the model may have different values depending on the current configuration. This feature is typically used to model systems with moving assemblies like zoom lenses or scanning mirror systems. The advantage of having several discrete states side by side is, that the whole system can be optimized for all states at once. If just the simulation of a system with moving assemblies or elements or other non static parameters is desired, most probably the Slider Interface is the better alternative, since it allows a continuous variation of the parameters instead of having only certain discrete realizations. To define the parameters, the Multiconfig Lookup Table has to be opened under the *Windows Icon* in the *Quick Access Toolbar*.

16.1.0.1 PP Multiconfig Parameter

The Multiconfig Parameter has besides the parameter *Name* as many additional columns as configurations exist.

Column	0	1	...	n
Name	ID	Value for Config 1	...	Value for Config n
ID	id	val_0	...	val_n-1
Editable	yes	yes	yes	yes

Each column holds the lookup value for one configuration. The lookup parameter can be set as variable as well. The variable state can be enabled independently for each configuration. When the current configuration changes, or eg. an analysis window or the optimizer require a specific configuration, the values of any parameters that refer to a parameter in the Multiconfig Lookup Table will be updated to the current configuration.

16.1.0.2 P Simple Parameter

The Simple Parameter hold contrary to the Multiconfig Parameter only one single value that is independent of the current configuration. The main usage for the Simple Parameter are either models, where several model parameters should have the same value all the time (e.g. when setting up parametric models), or models to be referenced to from following lookup parameters within the table. Th Simple Parameter can be set as variable as well. It can also be used to allow the optimization of parameters within Math Expressions.

Column	0	1
Name	ID	Value
ID	id	val
Editable	yes	yes

Note: In some other optical design software there exists the concept of a "Pickup" where a surface can copy a parameter value from the parameter of a previous surface. In Quadoa the same functionality can be achieved by parametric modelling using the Simple Parameter with the advantage of having a quick overview of all the parameters.

16.1.0.3 f(x) Math Expression

The Math Expression can be used to compute simple mathematical expressions like $sqrt(\sin(myParam1)+\cos(anotherParam)) * 13.37$ where myParam1 and anotherParam are parameters defined in previous rows inside the Multiconfig Lookup Table. The expression to be computed can be set in the second row of the Math Expression Parameter and is independent from the current configuration. If a parameter that is used inside the Math Expression is a Multiconfig Parameter, the value of that parameter will be dependent on the configuration, and therefore also the value of the Math Expression. If there is an unknown parameter used in the expression, or the expression is not a valid expression, the row will be highlighted in red and an error message will be displayed.

The following table gives an overview of operators and function that are allowed inside mathematical expressions:

Table 16.1: Overview on Math Expression Operands

Numeric Values		eg. 42, 1337, 3.1415, or pi
Parameters		Parameters defined in previous rows of the table
Basic Operators	+	plus
	-	minus
	*	multiplication
	/	division
	^	x to the power of y
Brackets	(	brackets
	)	brackets
Common Functions	abs	Abs. value of x
	exp	e to the power of x
	log	Nat. logarithm of x
	log10	Lob base 10 of x
	log2	Log base 2 of x
	sqrt	Squareroot of x
	root	nth root of x, root(x, n)
	sgn	sign of x (+1 if x>0 -1 if x>0)
Trigonometric Functions	acos	Arc cosine of x
	acosh	Inverse hyperbolic cosine of x
	asin	Arc sine of x
	asinh	Inverse hyperbolic sine of x
	atan	Arc tangent of x
	atanh	Inverse hyperbolic tangent of x
	atan2	Arc tangent of (x1 / x2), atan(x1, x2)
	cos	Cosine of x
	cosh	Hyperbolic cosine of x
	cot	Cotangent
	sin	Sine of x
	sinc	Sinc
	sinh	Hyperbolic sine of x
	tan	Tangent of x
	tanh	Hyperbolic tangent of x
	rad2deg	Convert x from radiant to degree
	deg2rad	Convert x from degrees to radians

16.1.0.4 Thermal Length

The Thermal Length is used to setup a Multiconfig system, where each configuration represents a temperature state of the optical system to be simulated and the thermal expansion of lens mounts and/or the lenses themselves needs to be considered. This feature is typically used for the athermalization of a lens. The Thermal Length has the following parameters:

Column	0	1	2	3	4	5	6
Name	ID	Temp Param	Nom Length	Extra Length	TEC Type	dL/dT	Nom. Temp
ID	id	t_param	nom_len	extra_len	tec_type	tec	nom_temp
Symbol	-	-	L_{nom}	L_{ext}	-	TEC	T_{nom}
Editable	yes	yes	yes	yes	yes	yes	yes

- *Temp Param* This parameter holds the ID of another Multiconfig Parameter that needs to be defined previously to the obj[]Thermal Length parameter that is referring to it. The referred parameter represent the different temperature states of the system. (To make the temperature of the Environment also match the temperature of each configuration the ID of the temperature parameter needs to also be referred from the Environments *Temperature* parameter. If the temperature change only affects certain elements, it can be set via the *Temperature* parameter of the Materials of the affected elements).
- *Nom Length* This parameter specifies the nominal length of the thermal length, which is the length at the nominal temperature.
- *Extra Length* This parameter specifies the extra length of the thermal length. The extra length is added to the nominal length when calculating the expansion and allows to take into account that the length of the mechanical part that is expanding under thermal influence differs from the nominal length.
- *TEC Type* allows to select where the thermal expansion coefficient for the material is taken from, or if it is entered manually.
- *dL/dT* specifies the thermal expansion coefficient of the material.
- *Nom. Temp* specifies the nominal temperature at which the nominal length is defined.

The length at a temperature state T is calculated by

$$L = L_{nom} + (L_{nom} + L_{ext}) \cdot TEC \cdot (T - T_{nom}) \tag{16.1}$$

16.2 Slider Interface

The Slider Interface allows to define optical systems, where some parameter are not static like eg. zoom lenses, scanning mirror systems, optics mounted on mechanical stages or in general any non static parameters that can be intuitively tuned by the user using sliders. Furthermore, the interface can be used during the optical design process to get a rough feeling on which parameter in the optical system has which effect on the system properties by manually tweaking those parameter through the sliders. If the goal is to optimize a system for several discrete states of a non static parameter, probably the Multiconfig Lookup Table is the better alternative. To define the parameters, the Slider Interface has to be opened under the Windows Icon in the Quick Access Toolbar.

16.2.1 Slider

The Slider is used as any other Lookup Parameter. Furthermore, it allows the user an intuitive way to directly change the parameter value by dragging around the slider. The first column of the Slider, as for any Lookup Parameter, defines the parameter name. The second column holds the slider-widget to change the current value. Contrary to the Multiconfig Lookup Table, the value can be changed independently for each parameter. The third column shows the current value. The fourth and fifth column define the minimum- and maximum value of the parameter range. The current value of the parameter can be anything in between the minimum- and maximum value, depending on the current position of the slider.

Column	0	1	2	3	4
Name	ID	-Slider-	Value	Minimum Value	Maximum Value
ID	id	val	val_scaled	min	max
Editable	yes	yes	no	yes	yes

16.2.1.1 f(x) Math Expression

As in the Multiconfig Lookup Table, the Slider Interface can contain Math Expressions. The properties and features of the Math Expressions are the same as in the Multiconfig Lookup Table. In the Slider Interface, the Math Expression can have previous Sliders or other Math Expressions as parameters inside the expression statement. This allows to define arbitrary dependent parameters where the value of one parameter can be described as a function of the other parameters values. This is especially useful to describe zoom lenses, where the movement of the assemblies can be described analytically.

17 Analysis

17.1 Ray distribution

17.1.1 Spot Diagram

The *Spot Diagram Plot* plots the 2D ray distribution of rays on the image surface of a Sequence. If the Sequnece has set the parameter *Image Space* to "Afocal" instead of the ray position, the ray angle relative to the reference defiened by *Reference* is plotted.

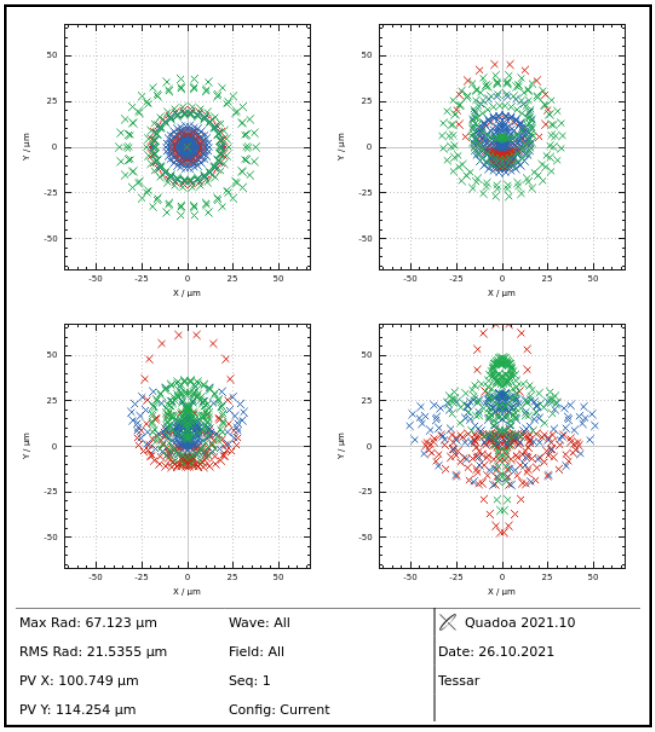


Figure 17.1: Example Spot Diagram

The plot has the following parameters:

- *Style*: Symbol to represent the coordinate of a ray in the plot

- *Sequence*: Sequence that defines the path of the light (order of Surfaces), Fields and Wavelengths for the raytrace.
- *Field*: Allows to select one specific or "All" Field. If "All" is selected one plot per Field is plotted.
- *Wavelength*: Allows to select one specific or "All" Wavelength.
- *Config*: Allows to select a specific state of the system defined by the Multiconfig Lookuptable. If "Current" is selected the plot will show the result for the config that is currently active.
- *Reference*: Allows to select weather the reference coordinate (0.0, 0.0) is taken from the "Chief Ray" or the "Centroid" of the rays
- *Distribution*: Allows to select the ray grid.
- *Param 1*: This parameter depends on the *Distribution* and is used to select the ray count.
- *Param 2*: This parameter depends on the *Distribution* and is used to select the ray count
- *Projection*: Plot the ray position in the "XY-Plane" or in a plane that is orthogonal to the chief ray ("Tang. Plane")
- *Color By*: Plot the rays in one color per "Field" or per "Wavelength".
- *Show Airy Disk*: If checked the Airy Disk (one per Wavelength) is included in the plot. In addition the Radius of the Airy disk of the primary Wavelength will be displayed in the *Plot Infobox*

17.1.2 Field vs. Wavelength Plot

The *Field vs. Wavelength Spot Diagram Plot* plots the 2D ray distribution of rays on the image surface of a Sequence in a grid, where each subplot contains the rays for one Field / Wavelength combination. The Wavelengths are arranged from left to right and the Fields are arranged from top to bottom according to the order in the Sequence. If the Sequnece has set the parameter *Image Space* to "Afocal" instead of the ray position, the ray angle relative to the reference defiened by *Reference* is plotted.

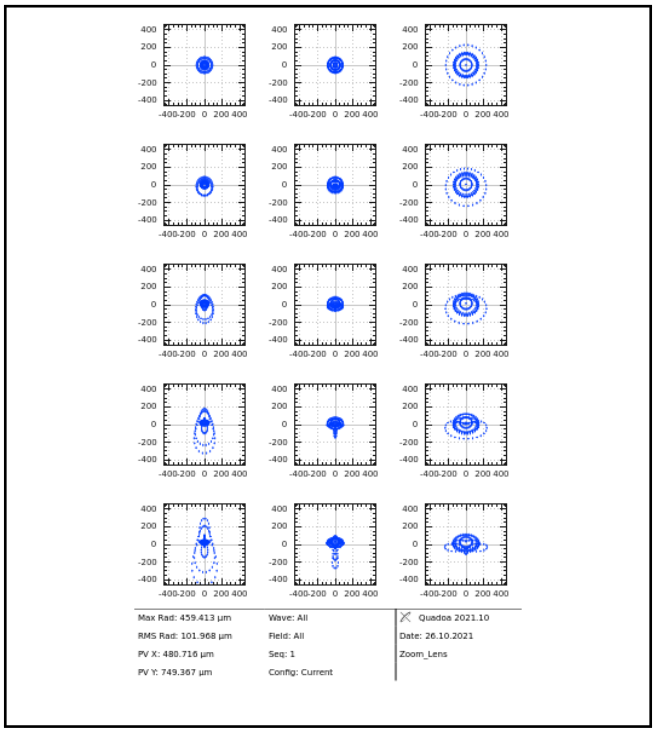


Figure 17.2: Example Field vs. Wavelength Spot Diagram

The plot has the following parameters:

- *Style*: Symbol to represent the coordinate of a ray in the plot
- *Sequence*: Sequence that defines the path of the light (order of Surfaces), Fields and Wavelengths for the raytrace.
- *Config*: Allows to select a specific state of the system defined by the Multiconfig Lookuptable. If "Current" is selected the plot will show the result for the config that is currently active.
- *Reference*: Allows to select weather the reference coordinate (0.0, 0.0) is taken from the "Chief Ray" or the "Centroid" of the rays
- *Distribution*: Allows to select the ray grid.
- *Param 1*: This parameter depends on the *Distribution* and is used to select the ray count.
- *Param 2*: This parameter depends on the *Distribution* and is used to select the ray count
- *Show Airy Disk*: If checked the Airy Disk (one per Wavelength) is included in the plot. In addition the Radius of the Airy disk of the primary Wavelength will be displayed in the *Plot Infobox*

17.1.3 Config vs. Field Plot

The *Config vs. Wavelength Spot Diagram Plot* plots the 2D ray distribution of rays on the image surface of a Sequence in a grid, where each subplot contains the rays for one Field / Configurations combination. The Configurations are arranged from left to right and the Fields are arranged from top to bottom according to the order in the Sequence. If the Sequnece has set the parameter *Image Space* to "Afocal" instead of the ray position, the ray angle relative to the reference defiened by *Reference* is plotted.

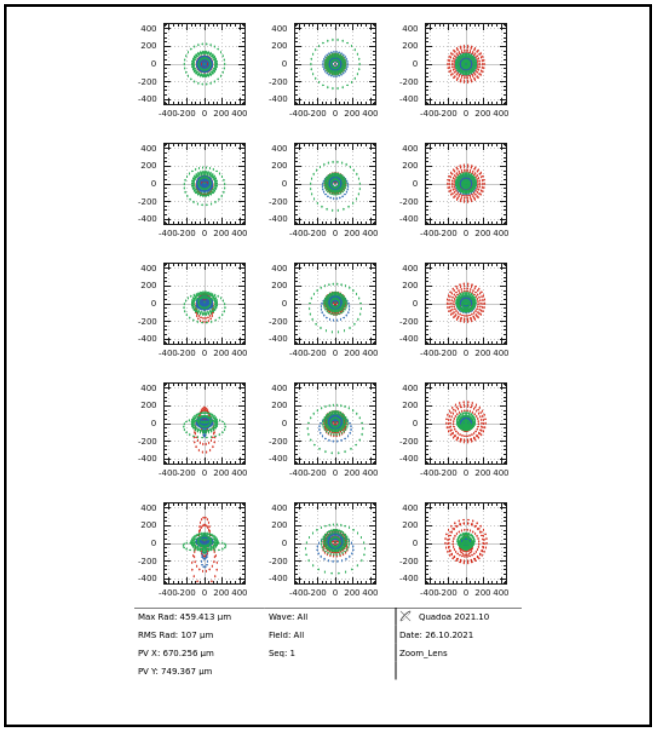


Figure 17.3: Example Config vs. Field Spot Diagram

The plot has the following parameters:

- *Style*: Symbol to represent the coordinate of a ray in the plot
- *Sequence*: Sequence that defines the path of the light (order of Surfaces), Fields and Wavelengths for the raytrace.
- *Wavelength*: Allows to select one specific or "All" Wavelength.
- *Reference*: Allows to select weather the reference coordinate (0.0, 0.0) is taken from the "Chief Ray" or the "Centroid" of the rays
- *Distribution*: Allows to select the ray grid.

- *Param 1*: This parameter depends on the *Distribution* and is used to select the ray count.
- *Param 2*: This parameter depends on the *Distribution* and is used to select the ray count
- *Show Airy Disk*: If checked the Airy Disk (one per Wavelength) is included in the plot. In addition the Radius of the Airy disk of the primary Wavelength will be displayed in the *Plot Infobox*

17.1.4 RMS Spotradius over Field

The *RMS Spotradius over Field* shows the RMS wavefront error plotted over the field. The plot has the following parameters:

- *Sequence*: Sequence that defines the path of the light (order of Surfaces), Fields and Wavelengths for the raytrace.
- *Wavelength*: Allows to select one specific or "All" Wavelength.
- *Config*: Allows to select a specific state of the system defined by the Multiconfig Lookuptable. If "Current" is selected the plot will show the result for the config that is currently active.
- *Reference*: Allows to select weather the reference coordinate (0.0, 0.0) is taken from the "Chief Ray" or the "Centroid" of the rays
- *Pupil Sampling*: Sampling of rays in the pupil
- *Field Sampling*: Sampling of spot diagrams along the field
- *Ray Weight*: Weight rays equally or by the "Apodization" or "Image Space Flux"
- *Enable Scattering*: Consider Scattering for the calculation

17.1.5 RMS Spot Radius over Field 2D

The *RMS Spotradius over Field 2D* shows the RMS wavefront error plotted over the X and Y field in two dimensions. The plot has the following parameters:

- *Sequence*: Sequence that defines the path of the light (order of Surfaces), Fields and Wavelengths for the raytrace.
- *Wavelength*: Allows to select one specific or "All" Wavelength.

- *Config*: Allows to select a specific state of the system defined by the Multiconfig Lookuptable. If "Current" is selected the plot will show the result for the config that is currently active.
- *Reference*: Allows to select weather the reference coordinate (0.0, 0.0) is taken from the "Chief Ray" or the "Centroid" of the rays
- *Pupil Sampling*: Weight rays equally or by the "Apodization" or "Image Space Flux"
- *Field Sampling*: Sampling of spot diagrams along the field
- *Enable Scattering*: Consider Scattering for the calculation

17.1.6 Footprint Diagram

The *Footprint Diagram Plot* shows the points at which the rays impinge a Surface. The x-, y-coordinates are plotted inside the local surface coordinate system. As a reference, the aperture of the surface is visualized as well.

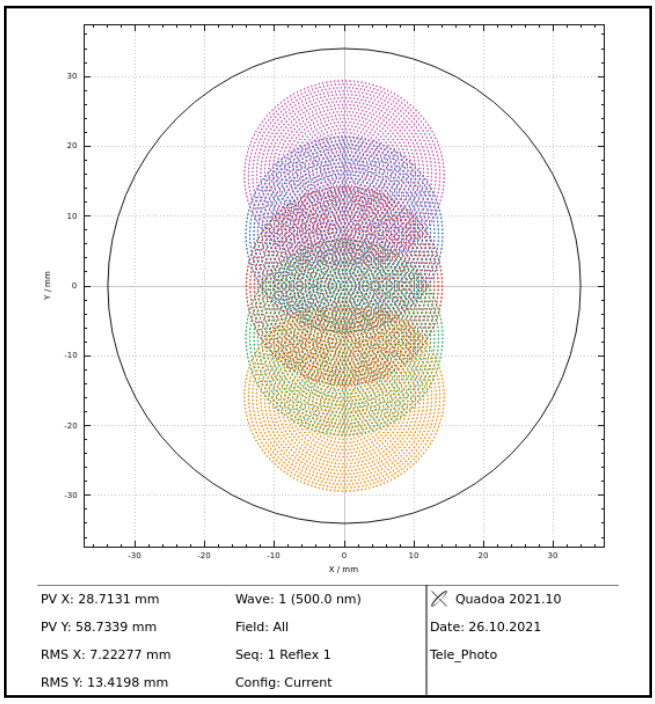


Figure 17.4: Example Footprint Diagram

The plot has the following parameters:

- *Style*: Symbol to represent the coordinate of a ray in the plot

- *Sequence*: Sequence that defines the path of the light (order of Surfaces), Fields and Wavelengths for the raytrace.
- *Field*: Allows to select one specific or "All" Field. If "All" is selected one plot per Field is plotted.
- *Wavelength*: Allows to select one specific or "All" Wavelength.
- *Config*: Allows to select a specific state of the system defined by the Multiconfig Lookuptable. If "Current" is selected the plot will show the result for the config that is currently active.
- *Surface*: Selection of the Surface on which the ray-coorindates are evaluated. The X, Y axis of the plot ar indentical to the local surface coordinate system x/y-axis
- *Color By*: Plot the rays in one color per "Field" or per "Wavelength".
- *Distribution*: Allows to select the ray grid.
- *Param 1*: This parameter depends on the *Distribution* and is used to select the ray count.
- *Param 2*: This parameter depends on the *Distribution* and is used to select the ray count.

17.1.7  Ghost Foot Print Diagram

The *Ghost Spot Diagram* plots the x-, y-coordinate of any Sequence that is marked as ghost sequence in the local coordinate system of the last surface of the ghost sequences. If different sets of ghost sequences for example one single and one double bounce are in the model, that end at different surfaces, the set of ghosts to be plotted can be selected by the surface to be analyzed via the *Surface* parameter. In the plot only the ghosts where the selected surface is the image surface of the sequence will be included. If the Sequnece has set the parameter *Image Space* to "Afocal" instead of the ray position, the ray angle relative to the reference defiened by *Reference* is plotted.

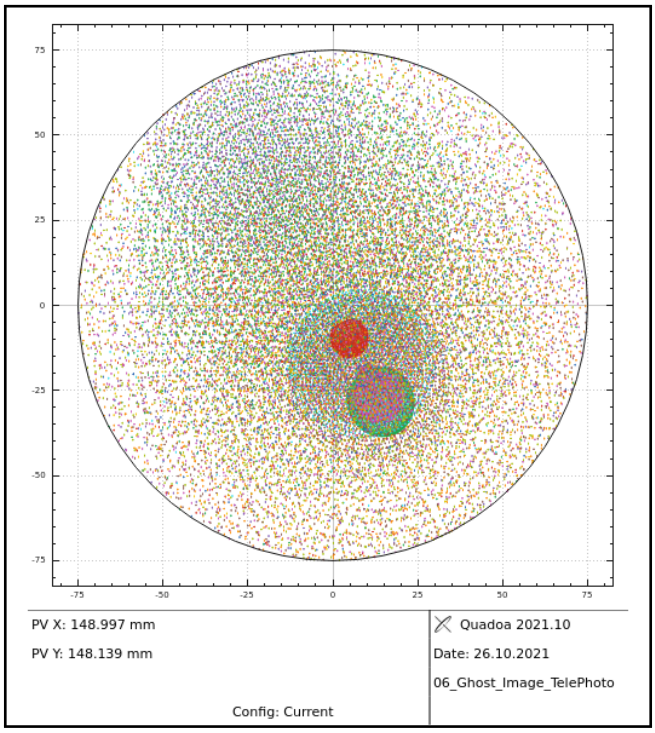


Figure 17.5: Example Ghost Spot Diagram

The plot has the following parameters:

- *Style*: Symbol to represent the coordinate of a ray in the plot
- *Surface*: Selects the surface on which to analyze the ghosts.
- *Config*: Allows to select a specific state of the system defined by the Multiconfig Lookuptable. If "Current" is selected the plot will show the result for the config that is currently active.
- *Distribution*: Allows to select the ray grid.
- *Param 1*: This parameter depends on the *Distribution* and is used to select the ray count.
- *Param 2*: This parameter depends on the *Distribution* and is used to select the ray count.

17.1.8 Vignetting Over Field Plot

The *Vignetting over Field* plot shows the relative amount of rays that reach the image surface. The Apodization of the rays is not taken into account in the calculation.

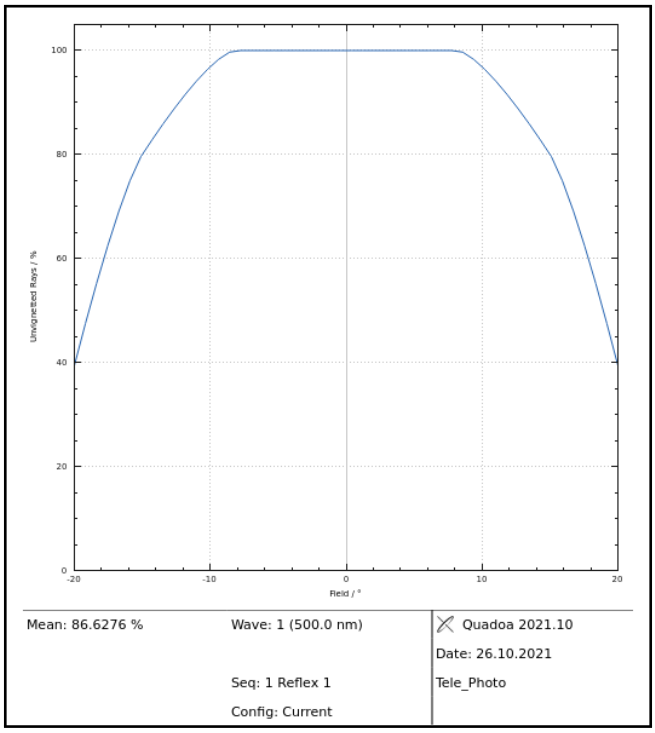


Figure 17.6: Example Vignetting over Field Diagram

The plot has the following parameters:

- *Sequence*: Sequence that defines the path of the light (order of Surfaces), Fields and Wavelengths for the raytrace.
- *Wavelength*: Defines which wavelength to use for the raytrace
- *Config*: Allows to select a specific state of the system defined by the Multiconfig Lookuptable. If "Current" is selected the plot will show the result for the config that is currently active.
- *Distribution*: Allows to select the ray grid.
- *Param 1*: This parameter depends on the *Distribution* and is used to select the ray count.
- *Param 2*: This parameter depends on the *Distribution* and is used to select the ray count.

17.2 Aberrations

17.2.1 OPD Fan Plot

The *OPD Fan Plot* plots the OPD along the x- and y-axis over the pupil of the sequence. Depending on the Sequence parameters *OPD Reference* and *OPD Chief Reference* the OPD may be corrected in different ways (See chapter 14.1).

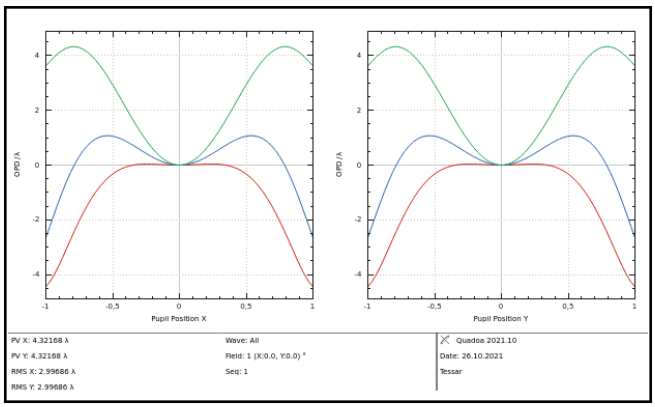


Figure 17.7: Example OPD Fan Plot

The plot has the following parameters:

- *Sequence*: Sequence that defines the path of the light (order of Surfaces), Fields and Wavelengths for the raytrace.
- *Field*: Allows to select one specific or "All" Field. If "All" is selected one plot per Field is plotted.
- *Wavelength*: Allows to select one specific or "All" Wavelength.
- *Config*: Allows to select a specific state of the system defined by the Multiconfig Lookuptable. If "Current" is selected the plot will show the result for the config that is currently active.
- *Sampling*: Number of points along the x- and y-axis for the sampling of the wavefront

17.2.2 Transversal Ray Fan Plot

The *Transversal Ray Fan Plot* plots the transversal aberration along the x- and y-axis over the pupil of the sequence. Depending on the Sequence parameters *OPD Refernence* and *OPD Chief Reference* the OPD may be corrected in different ways (See chapter 14.1).

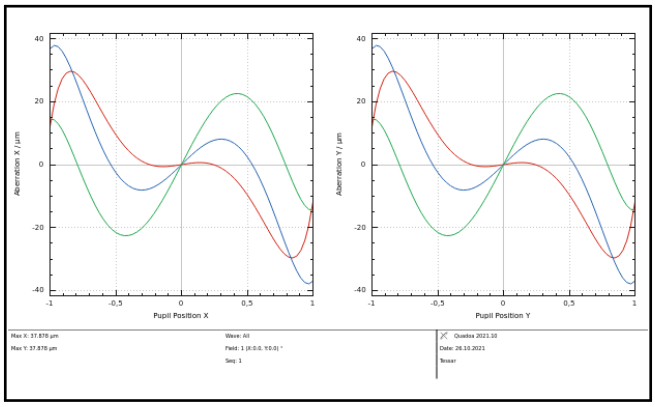


Figure 17.8: Example Transveral Aberration Fan Plot

The plot has the following parameters:

- *Sequence*: Sequence that defines the path of the light (order of Surfaces), Fields and Wavelengths for the raytrace.
- *Field*: Allows to select one specific or "All" Field. If "All" is selected one plot per Field is plotted.
- *Wavelength*: Allows to select one specific or "All" Wavelength.
- *Config*: Allows to select a specific state of the system defined by the Multiconfig Lookuptable. If "Current" is selected the plot will show the result for the config that is currently active.
- *Sampling*: Number of points along the x- and y-axis for the sampling of the wavefront

17.2.3 Seidel Bar Plot

The *Seidel Bar* plot shows the Seidel Aberration Surface contribution as well as the Seidel Sum for a given Sequence in form of a bar-chart. The bars are arranged in groups where each group represents the surface contribution of one surface. The Seidel sum is plotted in the last column.

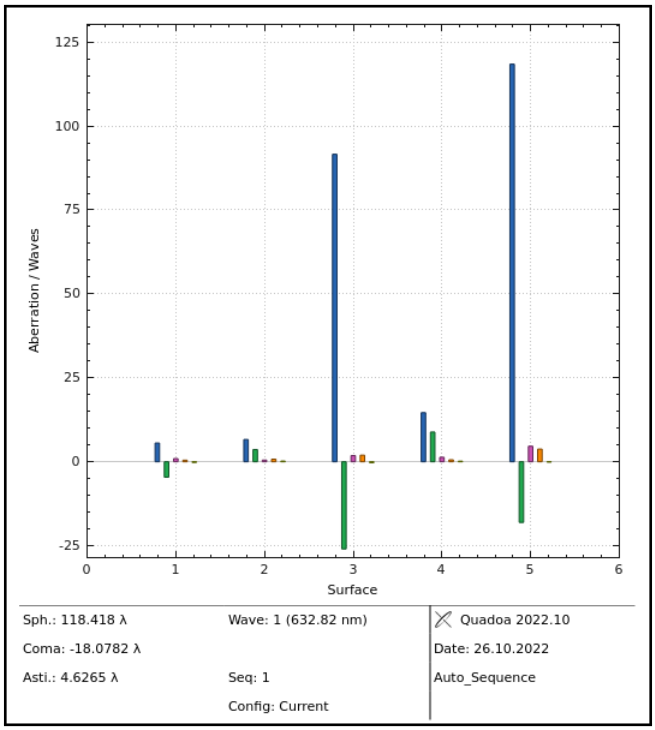


Figure 17.9: Example Seidel Bar Plot

The plot has the following parameters:

- *Sequence*: Sequence that defines the path of the light (order of Surfaces), Fields and Wavelengths for the raytrace.
- *Wavelength*: Selects the Wavelength for which to compute the Seidel Aberration.
- *Config*: Allows to select a specific state of the system defined by the Multiconfig Lookuptable. If "Current" is selected the plot will show the result for the config that is currently active.
- *Seidel Type*: Options are "S" and "W". "S" calculates the Seidel aberration such as S_I, S_{II}, S_{III}.... "W" calculates the wavefront aberrations such as W_{040}, W_{131}, W_{222}

Note: The Seidel aberrations only result in a meaningful value for classical systems using spherical lenses, without tilted or decentered elements. Cylindircal elements are supported as long as the curvature is along the y-axis.

17.2.4 Seidel Report

The *Seidel Report* shows the Seidel Aberration Surface contribution as well as the Seidel Sum for a given Sequence in form of a table. The report has the following parameters:

- *Sequence*: Sequence that defines the path of the light (order of Surfaces), Fields and Wavelengths for the raytrace.
- *Wavelength*: Selects the Wavelength for which to compute the Seidel Aberration.
- *Config*: Allows to select a specific state of the system defined by the Multiconfig Lookuptable. If "Current" is selected the plot will show the result for the config that is currently active.

Note: The Seidel aberrations only result in a meaningful value for classical systems using spherical lenses, without tilted or decentered elements. Cylindircal elements are supported as long as the curvature is along the y-axis.

17.2.5 Longitudinal Aberration Plot

The *Longitudinal Abberation Plot* shows the focal shift of a lens plotted over the wavelength. For the calculation the on axis Field with field coordinates (0, 0) is used. The range of the wavelength axis is defined by the minimum and maximum Wavelength in the Sequence.

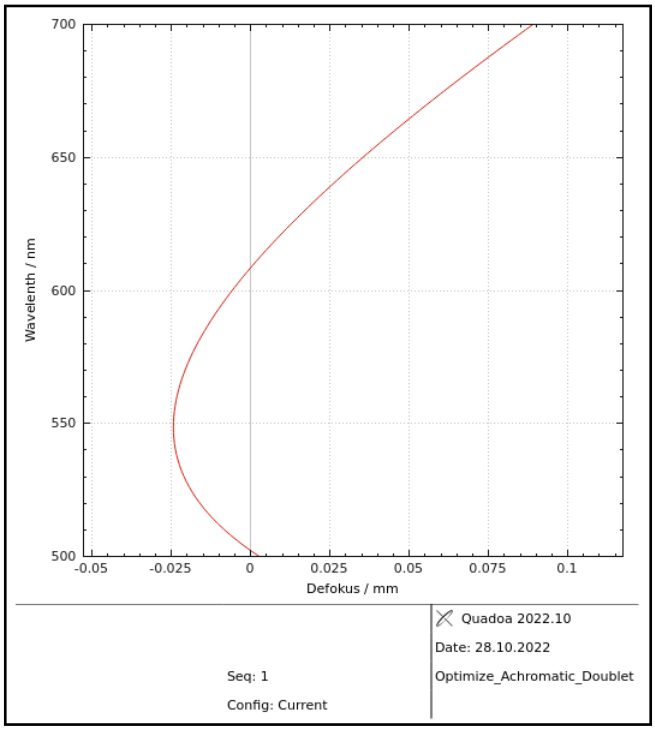


Figure 17.10: Example Longitudinal Aberration Plot

The plot has the following parameters:

- *Sequence*: Sequence that defines the path of the light (order of Surfaces), Fields and Wavelengths for the raytrace.
- *Config*: Allows to select a specific state of the system defined by the Multiconfig Lookuptable. If "Current" is selected the plot will show the result for the config that is currently active.
- *Calc Type*: Selects whether the paraxial or the marginal rays are used to calculate the defocus. Furthermore it can be chosen if the rays in $\pm x$ or $\pm y$ direction are used for the calculation.

17.2.6 Longitudinal Pupil Aberration Plot

The *Longitudinal Pupil Aberration Plot* shows the focal shift of a lens plotted over the pupil for each Wavelength in the Sequence. For the calculation the on axis Field with field coordinates (0, 0) is used.

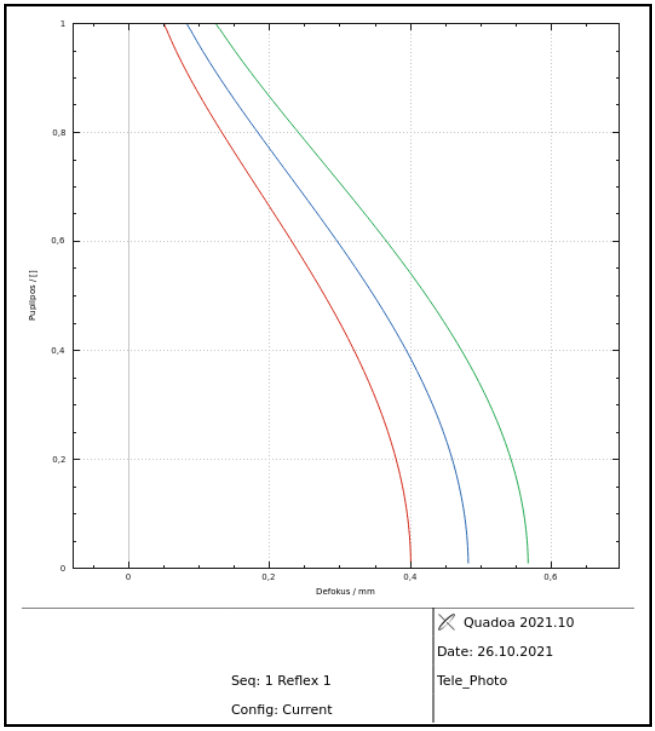


Figure 17.11: Example Longitudinal Pupil Aberration Plot

The plot has the following parameters:

- *Sequence*: Sequence that defines the path of the light (order of Surfaces), Fields and Wavelengths for the raytrace.

- *Config*: Allows to select a specific state of the system defined by the Multiconfig Lookuptable. If "Current" is selected the plot will show the result for the config that is currently active.
- *Fan Type*: Selects the direction of the fan that samples the pupil.

17.2.7 Lateral Chromatic Aberration Plot

The *Lateral Chromatic Aberration Plot* shows the lateral chromatic aberration of a lens plotted over the fiels for each Wavelength in the Sequence.

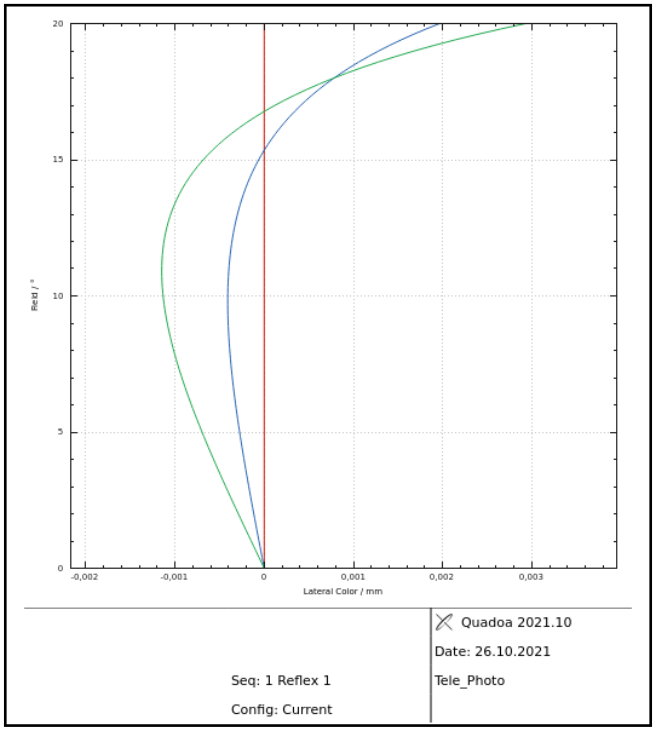


Figure 17.12: Example Lateral Chromatic Aberration Plot

The plot has the following parameters:

- *Sequence*: Sequence that defines the path of the light (order of Surfaces), Fields and Wavelengths for the raytrace.
- *Config*: Allows to select a specific state of the system defined by the Multiconfig Lookuptable. If "Current" is selected the plot will show the result for the config that is currently active.
- *Fan Type*: Selects the direction of the fan that samples the pupil.

17.2.8 Distortion 1D Plot

The *Distortion 1D Plot* shows the distortion of a lens along a fan of field points.

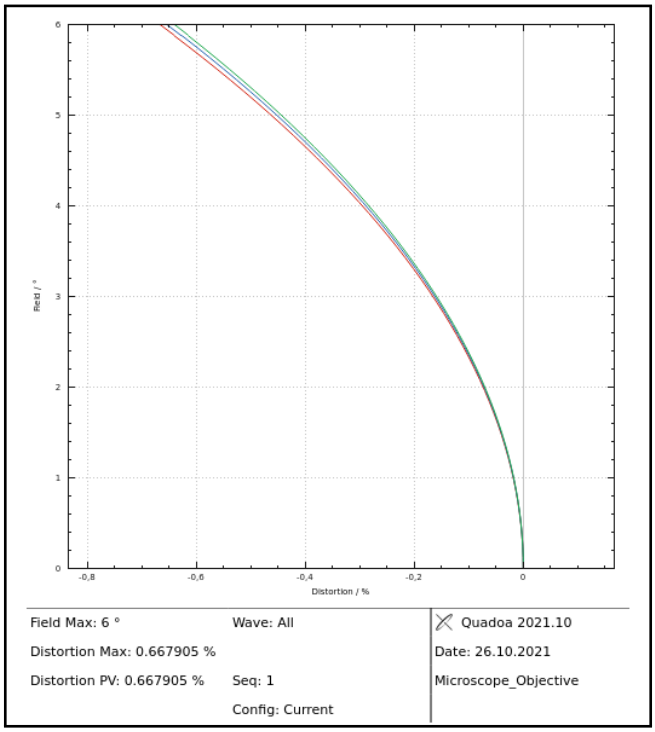


Figure 17.13: Example Distortion 1D Plot

The plot has the following parameters:

- *Sequence*: Sequence that defines the path of the light (order of Surfaces), Fields and Wavelengths for the raytrace.
- *Wavelength*: Selects the Wavelength for which to compute the distortion
- *Config*: Allows to select a specific state of the system defined by the Multiconfig Lookuptable. If "Current" is selected the plot will show the result for the config that is currently active.
- *Fan Type*: Selects the direction of the fan that samples the fields.
- *Field Referance*: This parameter can be either set to "F Tan(Theta" or "F Theta". If set to "F Tan(Theta)" a distortion free system is considered to be a system, where the image height linearly grows with the object height which is also the tangent of the chief rays field-angle in object space. If set to "F Theta" a distortion free system is considered to be a system, where the image height linearly grows with the field-angle of the chief ray in object space.

- *Sampling*: Number of field points to evaluate the distortion at.

17.2.9 Distortion 2D Plot

The *Distortion 2D Plot* shows the distortion of a lens over a grid along the field coordinates.

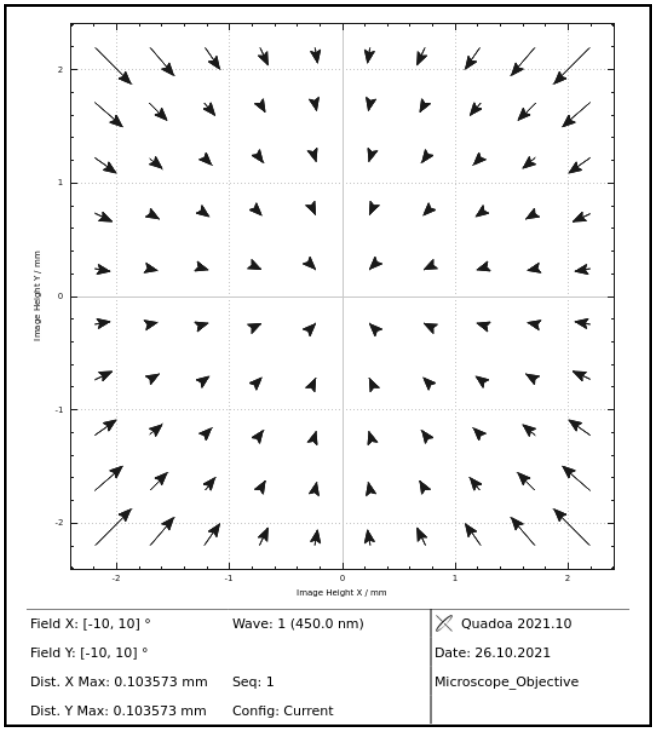


Figure 17.14: Example 2D-Distortion Plot

The plot has the following parameters:

- *Sequence*: Sequence that defines the path of the light (order of Surfaces), Fields and Wavelengths for the raytrace.
- *Wavelength*: Selects the Wavelength for which to compute the distortion
- *Config*: Allows to select a specific state of the system defined by the Multiconfig Lookuptable. If "Current" is selected the plot will show the result for the config that is currently active.
- *Field Referance*: This parameter can be either set to "F Tan(Theta" or "F Theta". If set to "F Tan(Theta)" a distortion free system is considered to be a system, where the image height linearly grows with the object height which is also the tangent of the chief rays field-angle in object space. If set to "F Theta" a distortion free system is considered to be a system, where the image height linearly grows with the field-angle of the chief ray in object space.

- *Sampling*: Number of field points in x, and y-direction to evaluate the distortion at.
- *Draw Type*: Draw the distortion as a vector field or as a distorted grid.
- *Scale Arrows*: This parameter can be used in combination with *Draw Type* set to "Vector" to exaggerate the distortion for a better visualization of small distortions.

17.2.10 Petzval Curvature Plot

The *Petzval Curvature Plot* shows the field curvature of a lens over the field coordinates, where the defocus is calculated by tracing a set of paraxial rays.

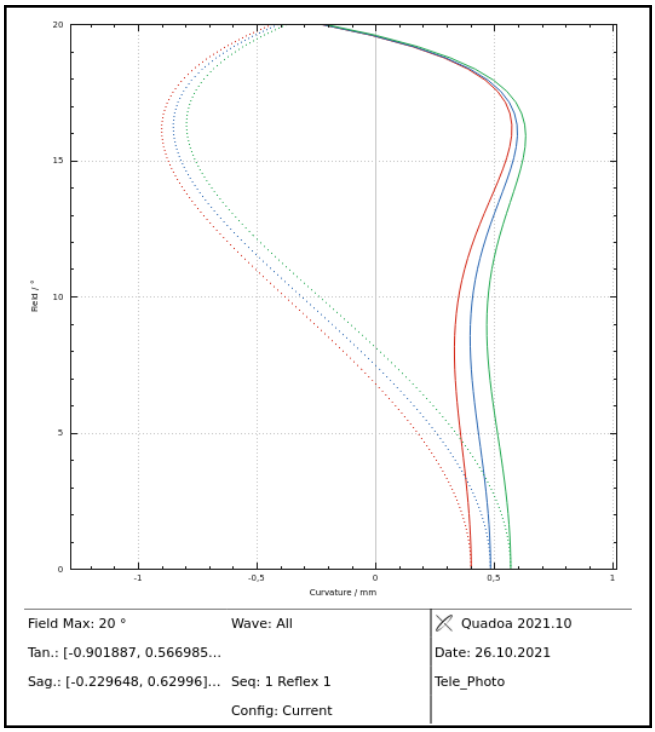


Figure 17.15: Example Petzval Curvature Plot

The plot has the following parameters:

- *Sequence*: Sequence that defines the path of the light (order of Surfaces), Fields and Wavelengths for the raytrace.
- *Wavelength*: Selects the Wavelength for which to compute the distortion
- *Config*: Allows to select a specific state of the system defined by the Multiconfig Lookuptable. If "Current" is selected the plot will show the result for the config that is currently active.

- *Fan Type*: Selects the direction of the fan that samples the fields.
- *Sampling*: Number of field points to evaluate the distortion at.

17.3 Wavefront

17.3.1 Wavefront Falsecolor

The *Wavefront Falsecolor Plot* shows the wavefront error of a Sequence in form of a 2D false color image, a 3D false color surface or a line cut. Depending on the Sequence parameters *OPD Reference* and *OPD Chief Reference* the OPD may be corrected in different ways (See chapter 14.1).

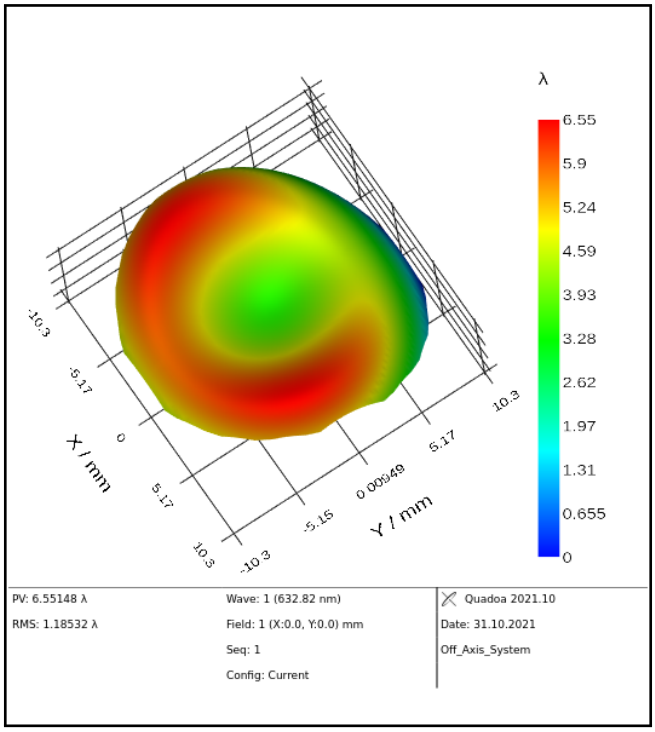


Figure 17.16: Example of Falsecolor Wavefront Plot

The plot has the following parameters:

- *Sequence*: Sequence that defines the path of the light (order of Surfaces), Fields and Wavelengths for the raytrace.
- *Field*: Allows to select one specific or "All" Field. If "All" is selected one plot per Field is plotted.

- *Wavelength*: Allows to select one specific or "All" Wavelength.
- *Config*: Allows to select a specific state of the system defined by the Multiconfig Lookuptable. If "Current" is selected the plot will show the result for the config that is currently active.
- *Pupil Sampling*: The sampling of the wavefront
- *Subtract Tilt*: If checked the tilt of the wavefront is subtracted
- *Subtract Defocus*: If checked the defocus of the wavefront is subtracted

17.3.2 Wavefront Fringes

The *Wavefront Fringes Plot* shows the wavefront error of a Sequence in form of a 2D image a 3D surface or a line cut. The color of the image is modulated from black to white according to the phase of the wavefront at each pixel. The result is comparable to the intensity modulation of the camera when the wavefront would interfere with a aberration free reference wavefront. Depending on the Sequence parameters *OPD Reference* and *OPD Chief Reference* the OPD may be corrected in different ways (See chapter 14.1).

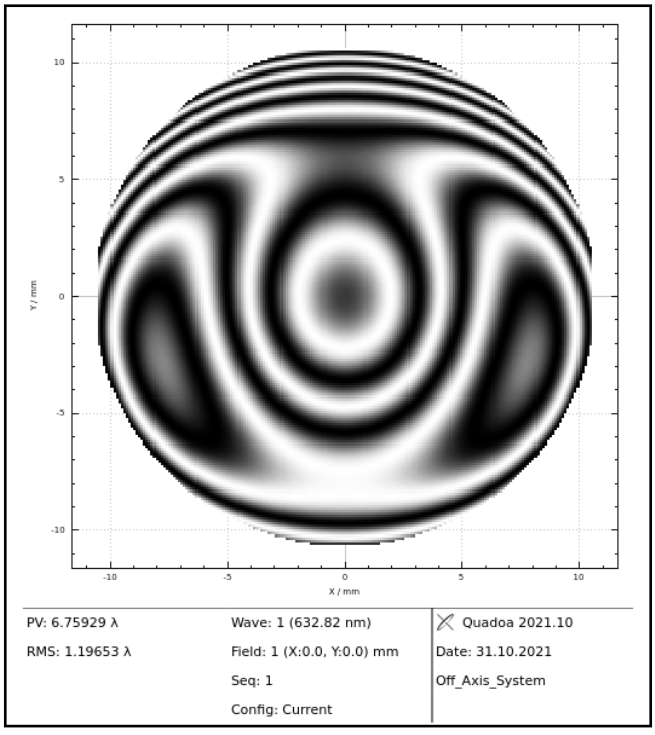


Figure 17.17: Example of Fringes Wavefront Plot

The plot has the following parameters:

- *Sequence*: Sequence that defines the path of the light (order of Surfaces), Fields and Wavelengths for the raytrace.
- *Field*: Allows to select one specific or "All" Field. If "All" is selected one plot per Field is plotted.
- *Wavelength*: Allows to select one specific or "All" Wavelength.
- *Config*: Allows to select a specific state of the system defined by the Multiconfig Lookuptable. If "Current" is selected the plot will show the result for the config that is currently active.
- *Pupil Sampling*: The sampling of the wavefront
- *Subtract Tilt*: If checked the tilt of the wavefront is subtracted
- *Subtract Defocus*: If checked the defocus of the wavefront is subtracted

17.3.3 Wavefront Gradient

The *Wavefront Gradient Plot* shows the gradient of the wavefront error of a Sequence in form of a 2D false color image, a 3D false color surface or a line cut. Depending on the Sequence parameters *OPD Reference* and *OPD Chief Reference* the OPD may be corrected in different ways (See chapter 14.1).

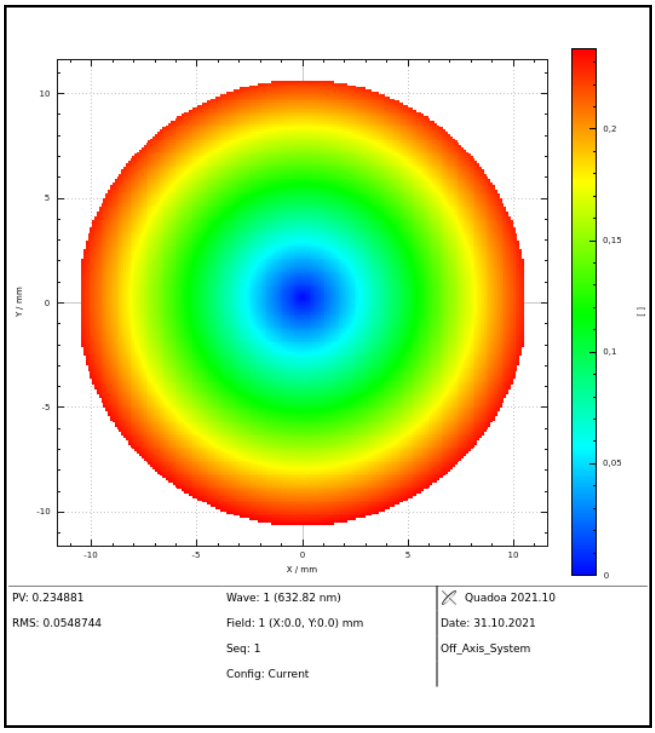


Figure 17.18: Example of Wavefront Gradient Plot

The plot has the following parameters:

- *Sequence*: Sequence that defines the path of the light (order of Surfaces), Fields and Wavelengths for the raytrace.
- *Field*: Allows to select one specific or "All" Field. If "All" is selected one plot per Field is plotted.
- *Wavelength*: Allows to select one specific or "All" Wavelength.
- *Config*: Allows to select a specific state of the system defined by the Multiconfig Lookuptable. If "Current" is selected the plot will show the result for the config that is currently active.
- *Pupil Sampling*: The sampling of the wavefront
- *Direction*: Selects how to compute the gradient. If "dX" is selected the gradient in x-direction is shown. If "dY" is selected the gradient in y-direction is shown. If "|| ||" is selected the absolute 2D gradient is shown

17.3.4 RMS RMS Wavefront over Field

The *RMS Wavefront over Field Plot* Plots the RMS Wavefront error over the Field.

The plot has the following parameters:

- *Sequence*: Sequence that defines the path of the light (order of Surfaces), Fields and Wavelengths for the raytrace.
- *Field*: Allows to select one specific or "All" Field. If "All" is selected one plot per Field is plotted.
- *Wavelength*: Allows to select one specific or "All" Wavelength.
- *Config*: Allows to select a specific state of the system defined by the Multiconfig Lookuptable. If "Current" is selected the plot will show the result for the config that is currently active.
- *Pupil Sampling*: The sampling of the wavefront
- *Field Sampling*: The sampling of wavefronts along the field
- *Subtract Tilt*: If checked the tilt of the wavefront is subtracted
- *Subtract Defocus*: If checked the defocus of the wavefront is subtracted
- *Ray Weight*: Weight rays equally or by "Apodization" or "Image Space Flux"

17.3.5 RMS RMS Wavefront over Field 2D

The *RMS Wavefront over Field 2D Plot* Plots the RMS Wavefront error over the X and Y-Field in two dimensions.

The plot has the following parameters:

- *Sequence*: Sequence that defines the path of the light (order of Surfaces), Fields and Wavelengths for the raytrace.
- *Field*: Allows to select one specific or "All" Field. If "All" is selected one plot per Field is plotted.
- *Wavelength*: Allows to select one specific or "All" Wavelength.
- *Config*: Allows to select a specific state of the system defined by the Multiconfig Lookuptable. If "Current" is selected the plot will show the result for the config that is currently active.

- *Pupil Sampling*: The sampling of the wavefront
- *Plot Sampling*: The sampling of wavefronts along the field
- *Subtract Tilt*: If checked the tilt of the wavefront is subtracted
- *Subtract Defocus*: If checked the defocus of the wavefront is subtracted
- *Ray Weight*: Weight rays equally or by "Apodization" or "Image Space Flux"

17.3.6  Zernike Report

The *Zernike Report* list the wavefront aberrations in form of coefficients of a Zernike polynomials fit to the wavefront. Depending on the Sequence parameters *OPD Reference* and *OPD Chief Reference* the OPD may be corrected in different ways prior to the fit (See chapter 14.1).

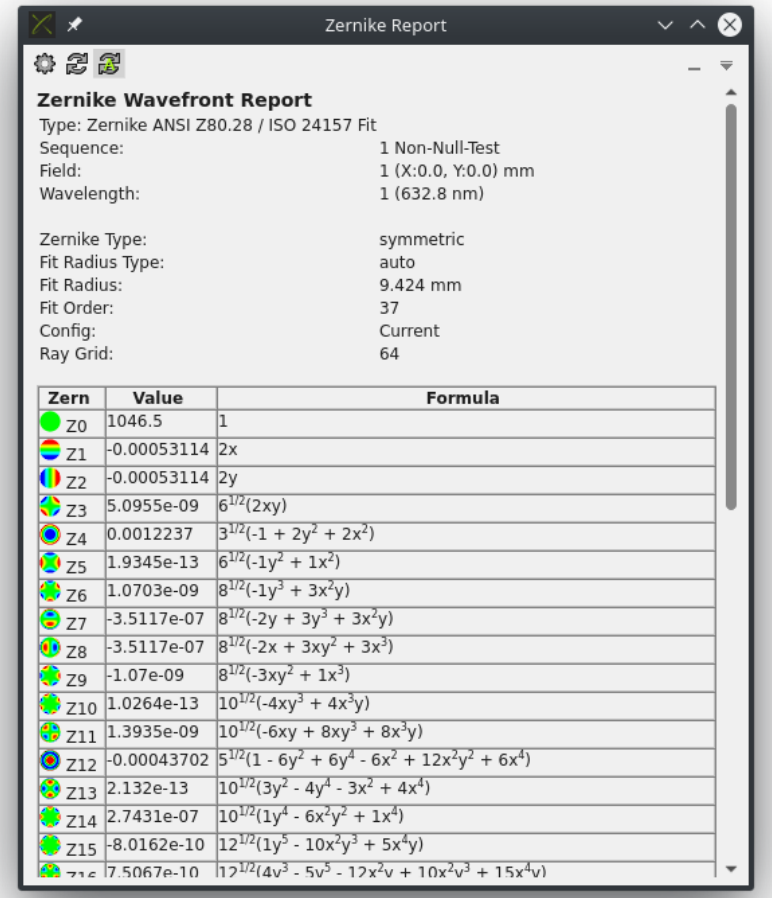


Figure 17.19: Example of Zernike Report

The plot has the following parameters:

- *Sequence*: Sequence that defines the path of the light (order of Surfaces), Fields and Wavelengths for the raytrace.
- *Field*: Allows to select one specific or "All" Field. If "All" is selected one plot per Field is plotted.
- *Wavelength*: Allows to select one specific or "All" Wavelength.
- *Config*: Allows to select a specific state of the system defined by the Multiconfig Lookuptable. If "Current" is selected the plot will show the result for the config that is currently active.
- *Sampling*: Pupil Sampling for the ray trace to fit the Zernike polynomial to

- *Fit Rad Mode*: Specifies how to define the fit radius of the Zernike polynomial. "Auto" will calculate the circumscribed circle around all rays in the OPD-Reference Surface and use this radius as the fit radius. The computed radius value will be listed at the beginning of the report. "From Aperture" will use the size of the Aperture of the OPD-Reference Surface as fit radius. "User" allows the user to enter the radius directly into the *Fit Radius* parameter.
- *Fit Radius*: Defines the fit radius for the Zernike fit. See *Fit Rad Mode*
- *Fit Order*: Defines the maximum polynomial order for the Zernike fit
- *Sort Order*: Defines the type of Zernike polynomial to fit. Options are "ANSI/ISO" and "Fringe" (See chapter 26.1)

17.4 Interferogram

17.4.1  Interferogram Fringes

The *Interferogram Fringes Plot* is used to visualize the difference in the optical path lengths of two ray traces in form of synthetic fringes. Other properties of the rays like the intensity, the polarization or coherence are not taken into account in this simulation. To simulate an interferogram as seen in an interferometer typically two Sequences need to be defined, where each Source shares a common wavelength and the *OPD Reference* parameter must be equal in both Sequences. In most cases the *OPD Reference* is set to "Abs over Image". In this case the interference on the last surface in the sequence will be computed. If set to any of the "Exit Pupil" options, the interference will be computed in the exit pupil surface. This can be used to visualize the wavefront difference between two ray traces, but does not correlate to a interferogram that could be observed in the real physical system.

Note: The Interferogram Plot is computed by first tracing the rays of the first Sequence. Then the rays of the second Sequence are traced. To get the overlapping area of the Sequences, where the light interferes a hull around the rays is computed and only the overlapping region is considered. In the next step the rays of the second Sequence are aimed on the exact coordinates of the rays of the first sequences on the OPD-Reference Surface. This way it can be avoided to interpolate the phase leading to a higher accuracy of the result. Since the ray aiming to the last Surface may be computational intense depending on the complexity of the Sequence it is always recommended to set the "simpler" Sequence as the second one.

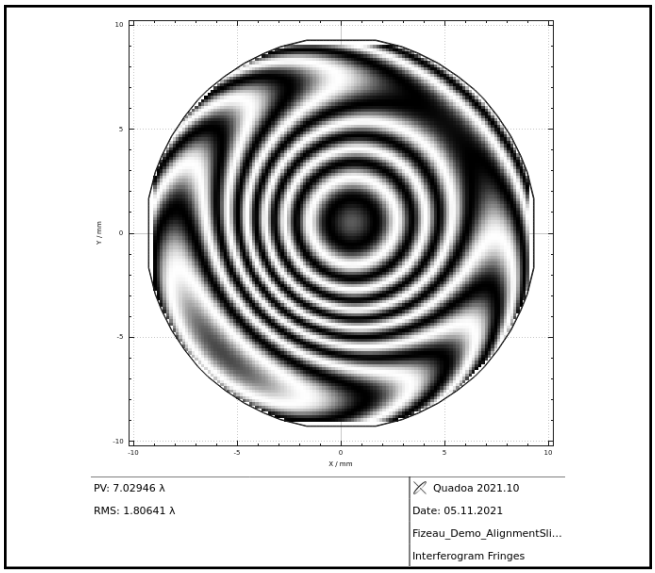


Figure 17.20: Example of Interferogram Fringes

The plot has the following parameters:

- *Sequence 1*: First Sequence that defines the path of the light for the ray trace of the test wavefront.
- *Sequence 2*: Second Sequence that defines the path of the light for the ray trace of the reference wavefront.
- *Field 1*: Field to use for the ray trace of the first Sequence.
- *Field 2*: Field to use for the ray trace of the second Sequence.
- *Wavelength 1*: Wavelength to use for the ray trace of the first Sequence.
- *Wavelength 2*: Wavelength to use for the ray trace of the second Sequence. The selected wavelength must match the wavelength selected by *Wavelength 1* otherwise an error will appear.
- *Config 1*: Selects a specific state of the system defined by the Multiconfig Lookuptable for the first ray trace. If "Current" is selected the plot will show the result for the config that is currently active.
- *Config 2*: Selects a specific state of the system defined by the Multiconfig Lookuptable for the first second trace. If "Current" is selected the plot will show the result for the config that is currently active.
- *Pixel Sampling*: Sampling of the wave front on the detector surface.

- *Substract Tilt*: Allows to remove the tilt from the computed interferogram
- *Subtract Defocus*: Allows to remove the defocus from the computed interferogram

17.4.2  Interferogram Unwrapped

The *Interferogram Fringes Plot* is used to visualize the difference in the optical path lengths of two ray traces in form of a unwrapped false-color phase map. Other properties of the rays like the intensity, the polarization or coherence are not taken into account in this simulation. To simulate an interferogram as seen in an interferometer typically two Sequences need to be defined, where each Source shares a common wavelength and the *OPD Reference* parameter must be equal in both Sequences. In most cases the *OPD Reference* is set to "Abs over Image". In this case the interference on the last surface in the sequence will be computed. If set to any of the "Exit Pupil" options, the interference will be computed in the exit pupil surface. This can be used to visualize the wavefront difference between two ray traces, but does not correlate to a interferogram that could be observed in the real physical system.

Note: The Interferogram Plot is computed by first tracing the rays of the first Sequence. Then the rays of the second Sequence are traced. To get the overlapping area of the Sequences, where the light interferes a hull around the rays is computed and only the overlapping region is considered. In the next step the rays of the second Sequence are aimed on the exact coordinates of the rays of the first sequences on the OPD-Reference Surface. This way it can be avoided to interpolate the phase leading to a higher accuracy of the result. Since the ray aiming to the last Surface may be computational intense depending on the complexity of the Sequence it is always recommended to set the "simpler" Sequence as the second one.

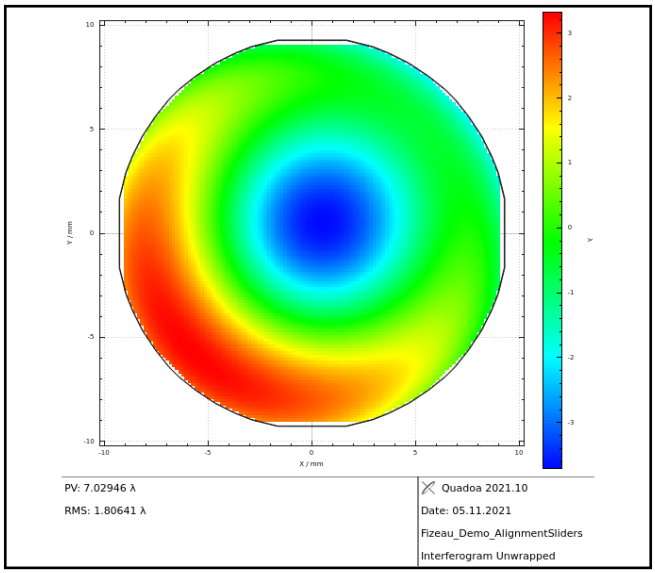


Figure 17.21: Example of Interferogram Unwrapped

The plot has the following parameters:

- *Sequence 1*: First Sequence that defines the path of the light for the ray trace of the test wavefront.
- *Sequence 2*: Second Sequence that defines the path of the light for the ray trace of the reference wavefront.
- *Field 1*: Field to use for the ray trace of the first Sequence.
- *Field 2*: Field to use for the ray trace of the second Sequence.
- *Wavelength 1*: Wavelength to use for the ray trace of the first Sequence.
- *Wavelength 2*: Wavelength to use for the ray trace of the second Sequence. The selected wavelength must match the wavelength selected by *Wavelength 1* otherwise an error will appear.
- *Config 1*: Selects a specific state of the system defined by the Multiconfig Lookuptable for the first ray trace. If "Current" is selected the plot will show the result for the config that is currently active.
- *Config 2*: Selects a specific state of the system defined by the Multiconfig Lookuptable for the first second trace. If "Current" is selected the plot will show the result for the config that is currently active.
- *Pixel Sampling*: Sampling of the wave front on the detector surface.

- *Substract Tilt*: Allows to remove the tilt from the computed interferogram
- *Subtract Defocus*: Allows to remove the defocus from the computed interferogram

17.5 Point Spread Function (PSF)

17.5.1 Fast Fourier Transform (FFT) PSF

The *Fast Fourier Transform (FFT) PSF Plot* shows the points spread function of a Sequence. The PSF is computed by aiming a set of rays to the exit pupil of the system. This leads to a map of the complex amplitude at the location of the exit pupil. From this map the PSF is computed by the method of Fourier optics [4].

Note that compared to the Huygens PSF Plot the FFT version is typically faster, however - especially for field that hit the image surface at a large angle - the Huygens PSF Plot is typically more accurate.

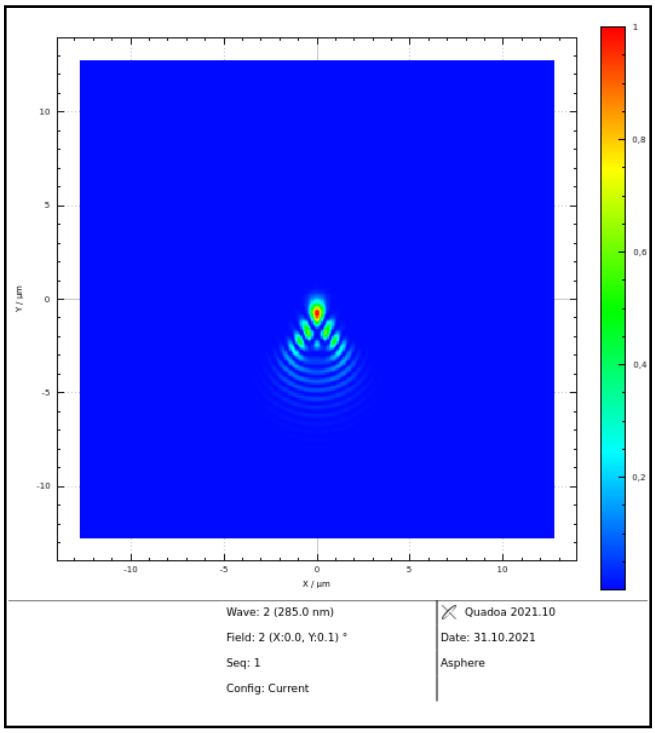


Figure 17.22: Example of FFT PSF Plot

The plot has the following parameters:

- *Sequence*: Sequence that defines the path of the light (order of Surfaces), Fields and Wavelengths for the raytrace.

- *Field*: Allows to select the Field
- *Wavelength*: Allows to select one specific or "Polychromatic". If "Polychromatic" is selected the PSF is computed for each wavelength of the Sequence and the result is the summed intensity of all wavelengths.
- *Config*: Allows to select a specific state of the system defined by the Multiconfig Lookuptable. If "Current" is selected the plot will show the result for the config that is currently active.
- *Pupil Sampling*: Sampling of rays in the pupil plane. The parameter defined the number of rays for the grid in one direction so the total amount of rays to be traced is the square of the number entered.
- *Image Sampling*: Sampling of pixels in the image plane. The parameter defined the number of pixels for the grid in one direction so the total amount of pixels is the square of the number entered.
- *Image Scale [µm]*: For the FFT method the size of the image area that is plotted directly results from the *Pupil Sampling* and *Image Sampling* and therefore cannot be chosen freely. This parameter however allows to crop the image if the area of interest is smaller. In general it is recommend to adjust the sampling parameters instead of cropping the image. The unit is μm.
- *PSF Type*: "Intensity" plots the intensity of the complex amplitude. "Phase" plots the phase of the complex amplitude. The "Phase" option is only available for single wavelengths, not for the polychromatic PSF.
- *Axis Type*: "Linear" or "Logarithmic"

17.5.2 Huygens PSF

The *Huygens PSF Plot* shows the points spread function of a Sequence. The PSF is computed by aiming a set of rays to the exit pupil of the system in the first step. In the second step from each of the rays in the pupil a 2D fan of rays (one ray per image space pixel) is aimed towards the image surface. This is equal to the launch of a grid of spherical wavefronts as derived from Huygens principal. On the image surface all the rays that arrive at a specific pixel interfere with each other by adding up the complex amplitude.

Note that compared to the FFT PSF Plot the Huygens version is typically slower, however - especially for field that hit the image surface at a large angle - it is typically more accurate.

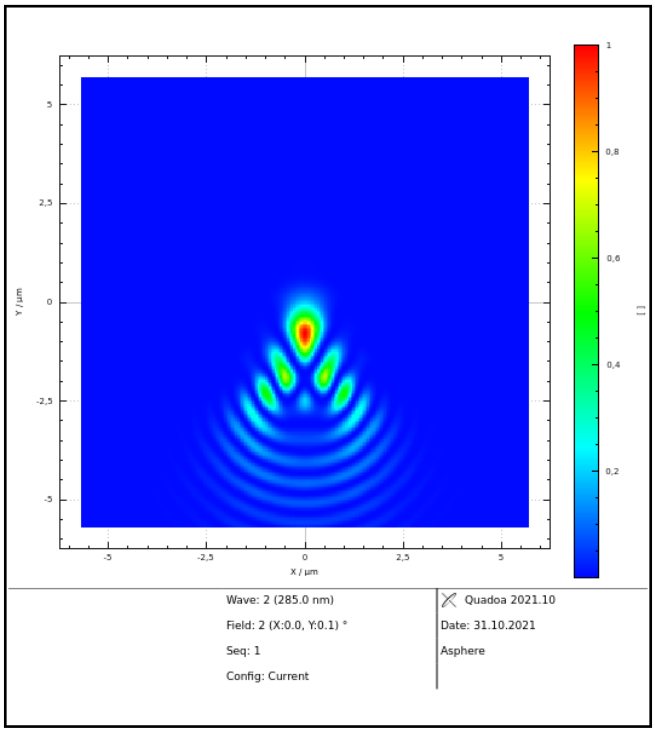


Figure 17.23: Example of Huygens PSF Plot

The plot has the following parameters:

- *Sequence*: Sequence that defines the path of the light (order of Surfaces), Fields and Wavelengths for the raytrace.
- *Field*: Allows to select the Field
- *Wavelength*: Allows to select one specific or "Polychromatic". If "Polychromatic" is selected the PSF is computed for each wavelength of the Sequence and the result is the summed intensity of all wavelengths.
- *Config*: Allows to select a specific state of the system defined by the Multiconfig Lookuptable. If "Current" is selected the plot will show the result for the config that is currently active.
- *Pupil Sampling*: Sampling of rays in the pupil plane. The parameter defined the number of rays for the grid in one direction so the total amount of rays to be traced is the square of the number entered.
- *Image Sampling*: Sampling of pixels in the image plane. The parameter defined the number of pixels for the grid in one direction so the total amount of pixels is the square of the number entered.

- *Image Scale [µm]*: This parameter defines the total half-size of the area on the image surface to be plotted in μm. If set to 0.0 the size will be estimated from the size of the Airy-disk.
- *PSF Type*: "Intensity" plots the intensity of the complex amplitude. "Phase" plots the phase of the complex amplitude. The "Phase" option is only available for single wavelengths, not for the polychromatic PSF.
- *Axis Type*: "Linear" or "Logarithmic"

17.5.3 Geometric PSF

The *Geometric PSF Plot* shows the density of the ray distribution of a ray trace on the image surface. It does not take any diffraction effects into account. Therefore the application of the *Geometric PSF Plot* is mainly for systems with large aberrations, where the *FFT PSF Plot* and *Huygens PSF Plot* break down and the aberrations dominate the diffraction effects.

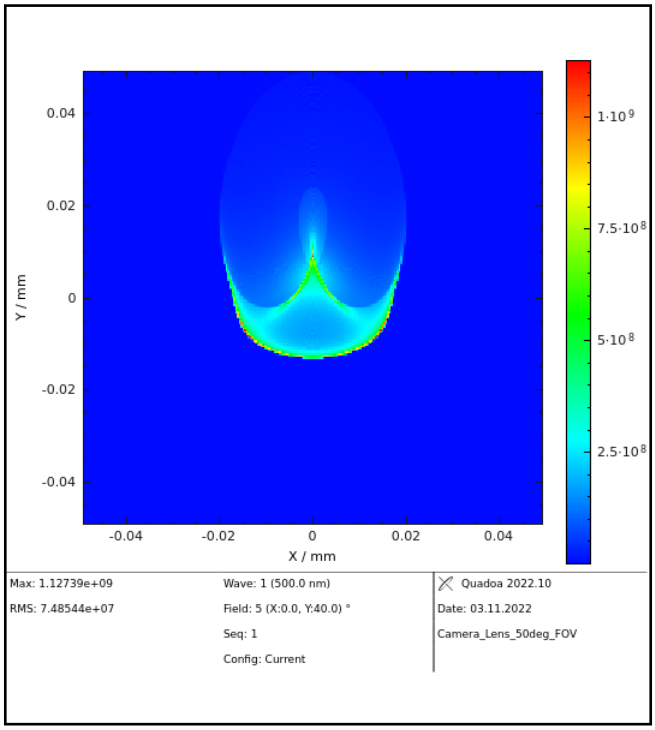


Figure 17.24: Example of Geometrical PSF Plot

The plot has the following parameters:

- *Sequence*: Sequence that defines the path of the light (order of Surfaces), Fields and Wavelengths for the raytrace.

- *Field*: Allows to select the Field
- *Wavelength*: Allows to select one specific or "Polychromatic". If "Polychromatic" is selected the PSF is computed for each wavelength of the Sequence and the result is the summed intensity of all wavelengths.
- *Config*: Allows to select a specific state of the system defined by the Multiconfig Lookuptable. If "Current" is selected the plot will show the result for the config that is currently active.
- *Pupil Sampling*: Number of rays in x, and y direction for the rasterization of the pupil plane that is used for PSF computation.
- *Image Sampling*: Rasterization of the pixels on the image surface

17.6 Modulation Transfer Function (MTF)

17.6.1  Fast Fouriert Transform (FFT) MTF

The *Fast Fourier Transform Modulation Transfer Function* or *FFT MTF Plot* shows the contrast of an image plotted over the spacial frequency of the image in the image plane, where for the computation of the PSF the method of the *FFT PSF* is used.

Note that depending on the sampling parameters the MTF may vary a lot, especially when the sampling is too low. To be sure that the sampling is sufficient for a given system with its aberrations it is therefore recommended to first check the PSF for any wavelengths and fields and only rely on the MTF result if all the PSF plots show a reasonable result. If the sampling works for the PSF the sampling parameters should be adapted 1:1 for the MTF plot.

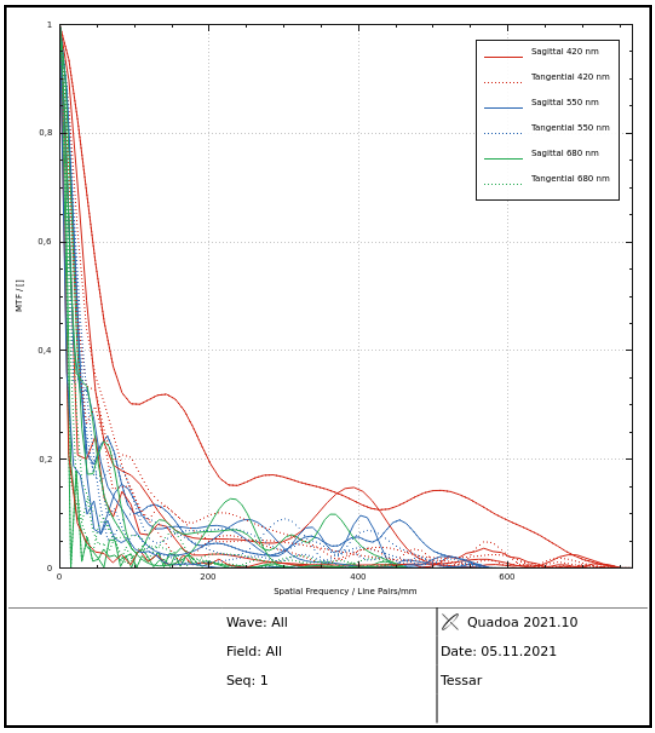


Figure 17.25: Example of FFT MTF Plot

The plot has the following parameters:

- *Sequence*: Sequence that defines the path of the light (order of Surfaces), Fields and Wavelengths for the raytrace.
- *Field*: Allows to select one specific or "All" Field. If "All" is selected one plot per Field is plotted.
- *Wavelength*: Allows to select one specific "All" or "Polychromatic". If "Polychromatic" is selected the PSF is computed for each wavelength of the Sequence and the result is the summed intensity of all wavelengths.
- *Config*: Allows to select a specific state of the system defined by the Multiconfig Lookuptable. If "Current" is selected the plot will show the result for the config that is currently active.
- *Pupil Sampling*: Sampling of rays in the pupil plane. The parameter defined the number of rays for the grid in one direction so the total amount of rays to be traced is the square of the number entered.
- *Image Sampling*: Sampling of pixels in the image plane. The parameter defined the number of pixels for the grid in one direction so the total amount of pixels is the square of the number entered.

- *Color by*: Color the individual plots by "Wavelength" or by "Field"

17.6.2 Huygens MTF

The *Huygens Modulation Transfer Function* shows the contrast of an image plotted over the spacial frequency of the image in the image plane, where for the computation of the PSF the method of the *Huygens PSF* is used.

Note that depending on the sampling parameters the MTF may vary a lot, especially when the sampling is too low. To be sure that the sampling is sufficient for a given system with its aberrations it is therefore recommended to first check the PSF for any wavelengths and fields and only rely on the MTF result if all the PSF plots show a reasonable result. If the sampling works for the PSF the sampling parameters should be adapted 1:1 for the MTF plot.

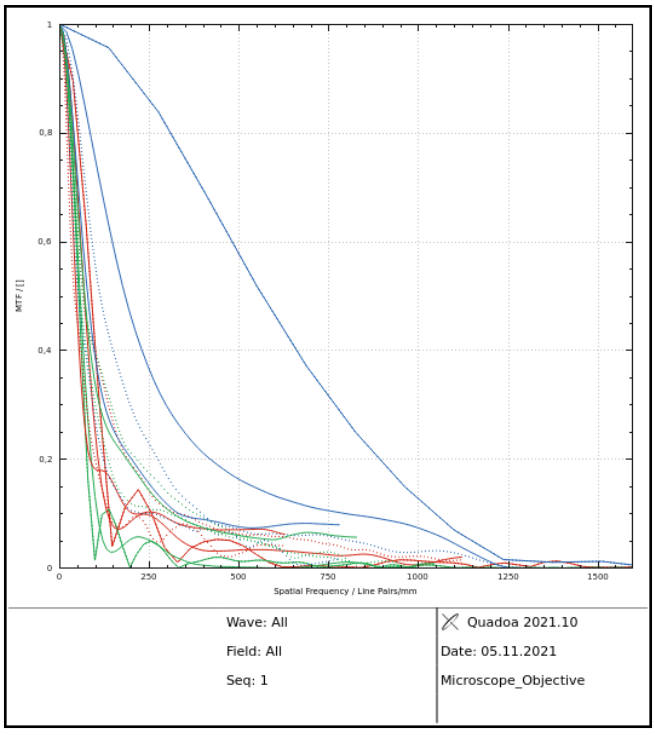


Figure 17.26: Example of Huygens MTF Plot

The plot has the following parameters:

- *Sequence*: Sequence that defines the path of the light (order of Surfaces), Fields and Wavelengths for the raytrace.

- *Field*: Allows to select one specific or "All" Field. If "All" is selected one plot per Field is plotted.
- *Wavelength*: Allows to select one specific "All" or "Polychromatic". If "Polychromatic" is selected the PSF is computed for each wavelength of the Sequence and the result is the summed intensity of all wavelengths.
- *Config*: Allows to select a specific state of the system defined by the Multiconfig Lookuptable. If "Current" is selected the plot will show the result for the config that is currently active.
- *Pupil Sampling*: Sampling of rays in the pupil plane. The parameter defined the number of rays for the grid in one direction so the total amount of rays to be traced is the square of the number entered.
- *Image Sampling*: Sampling of pixels in the image plane. The parameter defined the number of pixels for the grid in one direction so the total amount of pixels is the square of the number entered.
- *Image Scale [µm]*: This parameter defines the total half-size of the area on the image surface for the calculation of the PSF to be included in μm. If set to 0.0 the size will be estimated from the size of the Airy-disk. For aberrated systems this may lead to a cropping of the PSF. Therefore it is recommended to check the PSF before relying on the MTF plot.
- *Color by*: Color the individual plots by "Wavelength" or by "Field"

17.6.3  Geometric MTF

The *Geometric Modulation Transfer Function* shows the contrast of an image plotted over the spacial frequency of the image in the image plane, where for the computation of the PSF the method of the *Geometric PSF* is used.

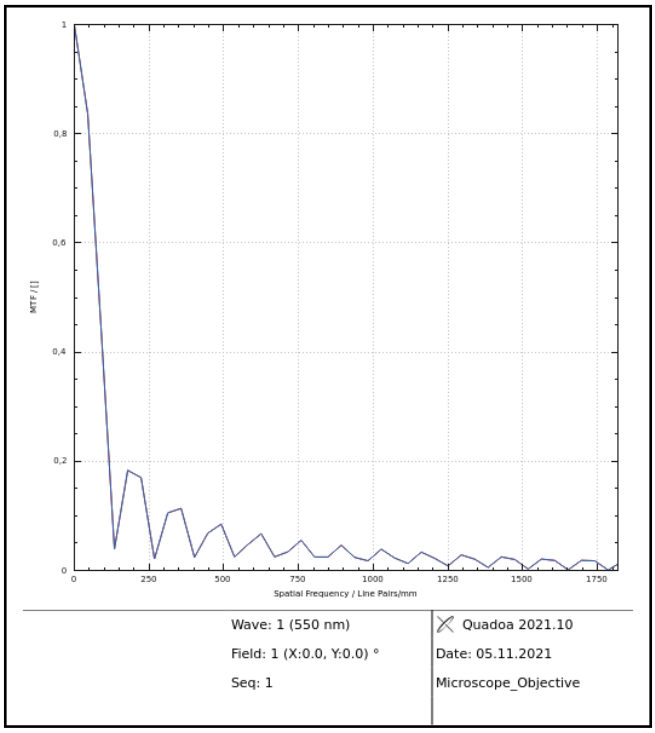


Figure 17.27: Example of Geometrical MTF Plot

The plot has the following parameters:

- *Sequence*: Sequence that defines the path of the light (order of Surfaces), Fields and Wavelengths for the raytrace.
- *Field*: Allows to select one specific or "All" Field. If "All" is selected one plot per Field is plotted.
- *Wavelength*: Allows to select one specific "All" or "Polychromatic". If "Polychromatic" is selected the PSF is computed for each wavelength of the Sequence and the result is the summed intensity of all wavelengths.
- *Config*: Allows to select a specific state of the system defined by the Multiconfig Lookuptable. If "Current" is selected the plot will show the result for the config that is currently active.
- *Pupil Sampling*: Number of rays in x, and y direction for the rasterization of the pupil plane that is used for PSF computation.
- *Image Sampling*: Rasterization of the pixels on the image surface
- *Color by*: Color the individual plots by "Wavelength" or by "Field"

17.6.4 Through Focus FFT MTF Scan

The *Through Focus FFT MTF Plot* shows the value of the MTF as computed by the *FFT MTF Plot* for a certain frequency while slightly defocussing the image over a range defined by the *Defocus* parameter in several steps defined by *Steps*.

Note that depending on the sampling parameters the MTF may vary a lot, especially when the sampling is too low. To be sure that the sampling is sufficient for a given system with its aberrations it is therefore recommended to first check the PSF for any wavelengths and fields and only rely on the MTF result if all the PSF plots show a reasonable result. If the sampling works for the PSF the sampling parameters should be adapted 1:1 for the MTF plot.

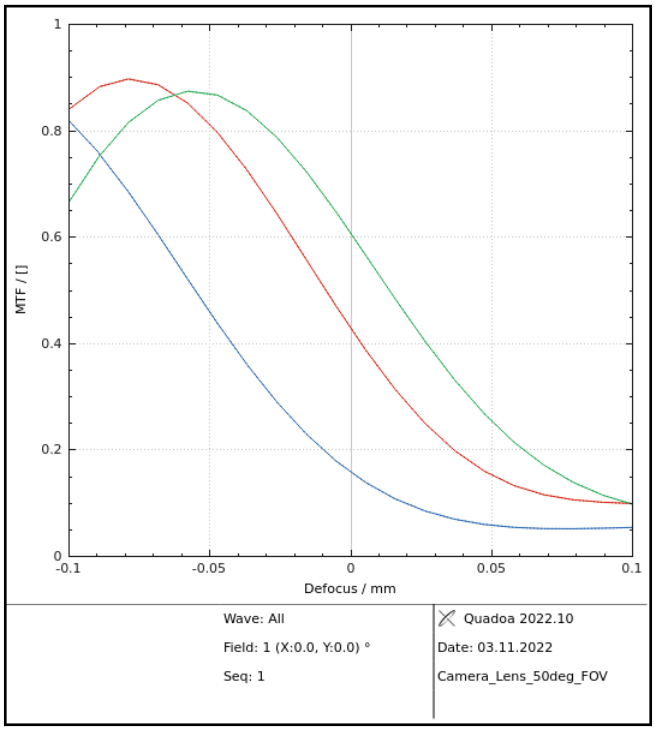


Figure 17.28: Example of Through Focus MTF Scan Plot

The plot has the following parameters:

- *Sequence*: Sequence that defines the path of the light (order of Surfaces), Fields and Wavelengths for the raytrace.
- *Field*: Allows to select one specific or "All" Field. If "All" is selected one plot per Field is plotted.

- *Wavelength*: Allows to select one specific "All" or "Polychromatic". If "Polychromatic" is selected the PSF is computed for each wavelength of the Sequence and the result is the summed intensity of all wavelengths.
- *Config*: Allows to select a specific state of the system defined by the Multiconfig Lookuptable. If "Current" is selected the plot will show the result for the config that is currently active.
- *Pupil Sampling*: Sampling of rays in the pupil plane. The parameter defined the number of rays for the grid in one direction so the total amount of rays to be traced is the square of the number entered.
- *Image Sampling*: Sampling of pixels in the image plane. The parameter defined the number of pixels for the grid in one direction so the total amount of pixels is the square of the number entered.
- *Color by*: Color the individual plots by "Wavelength" or by "Field"
- *Frequency*: Frequency at which to evaluate the MTF
- *Defocus*: Range of the defocus in positive and negative direction
- *Steps*: Number of steps for the sampling of the defocus range

17.6.5 Through Focus Huygens MTF Scan

The *Through Focus Huygens MTF Plot* shows the value of the MTF as computed by the *Huygens MTF Plot* for a certain frequency while slightly defocussing the image over a range defined by the *Defocus* parameter in several steps defined by *Steps*.

Note that depending on the sampling parameters the MTF may vary a lot, especially when the sampling is too low. To be sure that the sampling is sufficient for a given system with its aberrations it is therefore recommended to first check the PSF for any wavelengths and fields and only rely on the MTF result if all the PSF plots show a reasonable result. If the sampling works for the PSF the sampling parameters should be adapted 1:1 for the MTF plot.

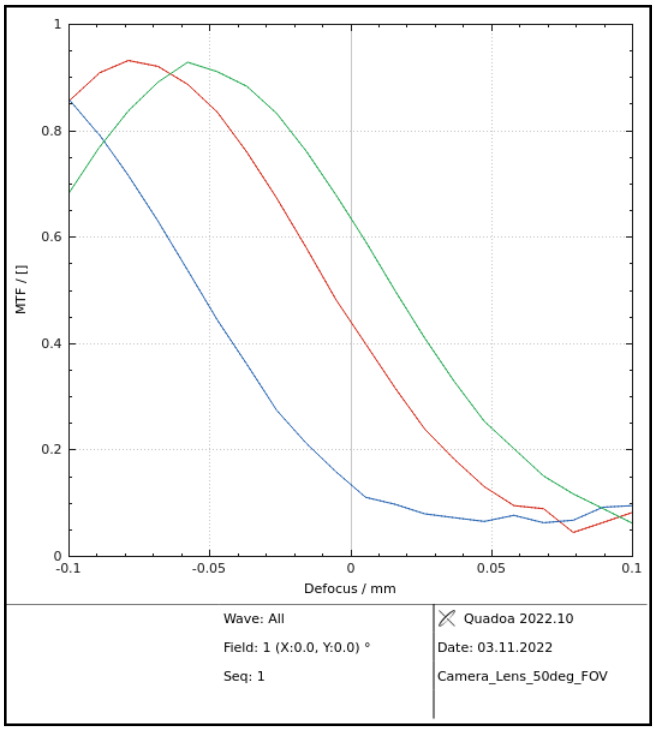


Figure 17.29: Example of Through Focus Huygens MTF Scan Plot

The plot has the following parameters:

- *Sequence*: Sequence that defines the path of the light (order of Surfaces), Fields and Wavelengths for the raytrace.
- *Field*: Allows to select one specific or "All" Field. If "All" is selected one plot per Field is plotted.
- *Wavelength*: Allows to select one specific "All" or "Polychromatic". If "Polychromatic" is selected the PSF is computed for each wavelength of the Sequence and the result is the summed intensity of all wavelengths.
- *Config*: Allows to select a specific state of the system defined by the Multiconfig Lookuptable. If "Current" is selected the plot will show the result for the config that is currently active.
- *Pupil Sampling*: Sampling of rays in the pupil plane. The parameter defined the number of rays for the grid in one direction so the total amount of rays to be traced is the square of the number entered.
- *Image Sampling*: Sampling of pixels in the image plane. The parameter defined the number of pixels for the grid in one direction so the total amount of pixels is the square of the number entered.

- *Image Scale [µm]*: This parameter defines the total half-size of the area on the image surface for the calculation of the PSF to be included in μm. If set to 0.0 the size will be estimated from the size of the Airy-disk. For aberrated systems this may lead to a cropping of the PSF. Therefore it is recommended to check the PSF before relying on the MTF plot.
- *Color by*: Color the individual plots by "Wavelength" or by "Field"
- *Frequency*: Frequency at which to evaluate the MTF
- *Defocus*: Range of the defocus in positive and negative direction
- *Steps*: Number of steps for the sampling of the defocus range

17.6.6 Through Focus Geometrical MTF Scan

The *Through Focus Geometrical MTF Plot* shows the value of the MTF as computed by the *Geometrical MTF Plot* for a certain frequency while slightly defocussing the image over a range defined by the *Defocus* parameter in several steps defined by *Steps*.

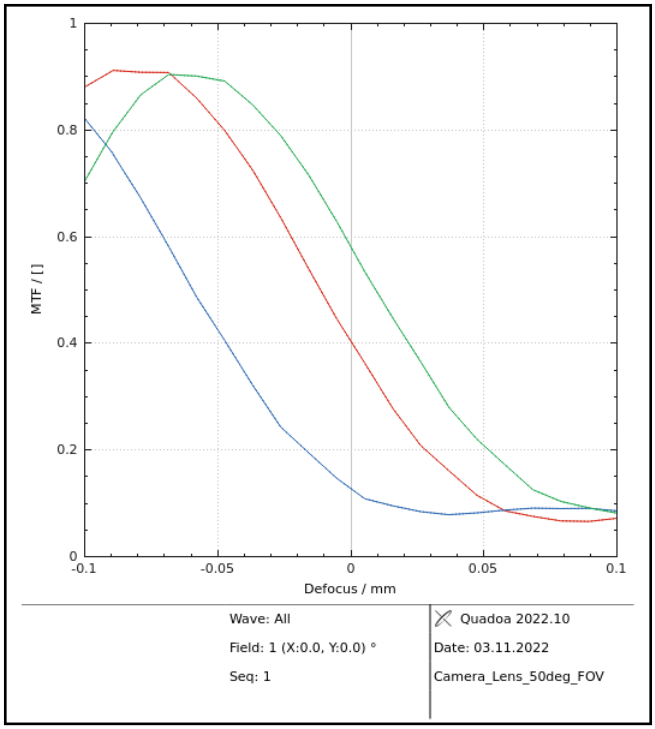


Figure 17.30: Example of Through Focus Geometrical MTF Scan Plot

The plot has the following parameters:

- *Sequence*: Sequence that defines the path of the light (order of Surfaces), Fields and Wavelengths for the raytrace.
- *Field*: Allows to select one specific or "All" Field. If "All" is selected one plot per Field is plotted.
- *Wavelength*: Allows to select one specific "All" or "Polychromatic". If "Polychromatic" is selected the PSF is computed for each wavelength of the Sequence and the result is the summed intensity of all wavelengths.
- *Config*: Allows to select a specific state of the system defined by the Multiconfig Lookuptable. If "Current" is selected the plot will show the result for the config that is currently active.
- *Pupil Sampling*: Number of rays in x, and y direction for the rasterization of the pupil plane that is used for PSF computation.
- *Image Sampling*: Rasterization of the pixels on the image surface
- *Color by*: Color the individual plots by "Wavelength" or by "Field"
- *Frequency*: Frequency at which to evaluate the MTF
- *Defocus*: Range of the defocus in positive and negative direction
- *Steps*: Number of steps for the sampling of the defocus range

17.6.7 FFT MTF vs. Field Plot

The *FFT MTF vs. Field Plot* shows the value of the MTF as computed by the *FFT MTF Plot* for a certain frequency over the field, in several steps defined by *Steps*.

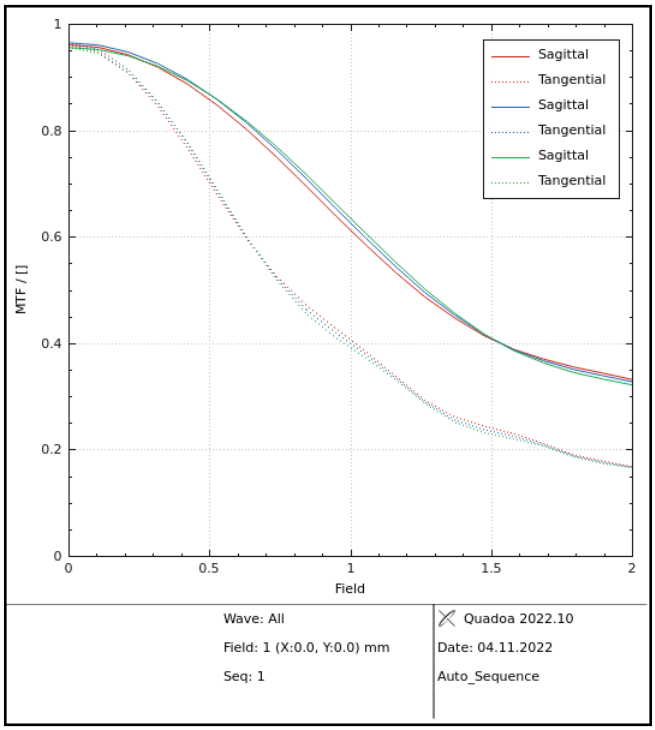


Figure 17.31: Example of FFT MTF vs. Field Plot

The plot has the following parameters:

- *Sequence*: Sequence that defines the path of the light (order of Surfaces), Fields and Wavelengths for the raytrace.
- *Wavelength*: Allows to select one specific "All" or "Polychromatic". If "Polychromatic" is selected the PSF is computed for each wavelength of the Sequence and the result is the summed intensity of all wavelengths.
- *Config*: Allows to select a specific state of the system defined by the Multiconfig Lookuptable. If "Current" is selected the plot will show the result for the config that is currently active.
- *Pupil Sampling*: Sampling of rays in the pupil plane. The parameter defined the number of rays for the grid in one direction so the total amount of rays to be traced is the square of the number entered.
- *Image Sampling*: Sampling of pixels in the image plane. The parameter defined the number of pixels for the grid in one direction so the total amount of pixels is the square of the number entered.
- *Color by*: Color the individual plots by "Wavelength" or by "Field"

- *Frequency*: Frequency at which to evaluate the MTF
- *Fan Type*: Direction of the field scan
- *Steps*: Number of steps for the sampling of the fields

17.6.8 Huygens MTF vs. Field Plot

The *Huygens MTF vs. Field Plot* shows the value of the MTF as computed by the *Huygens MTF Plot* for a certain frequency over the field, in several steps defined by *Steps*.

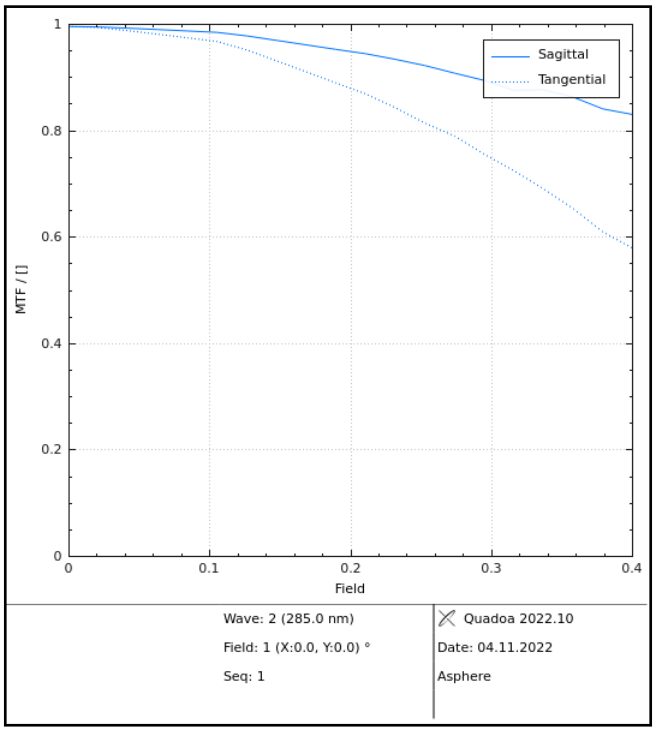


Figure 17.32: Example of Huygens MTF vs. Field Plot

The plot has the following parameters:

- *Sequence*: Sequence that defines the path of the light (order of Surfaces), Fields and Wavelengths for the raytrace.
- *Wavelength*: Allows to select one specific "All" or "Polychromatic". If "Polychromatic" is selected the PSF is computed for each wavelength of the Sequence and the result is the summed intensity of all wavelengths.

- *Config*: Allows to select a specific state of the system defined by the Multiconfig Lookuptable. If "Current" is selected the plot will show the result for the config that is currently active.
- *Pupil Sampling*: Sampling of rays in the pupil plane. The parameter defined the number of rays for the grid in one direction so the total amount of rays to be traced is the square of the number entered.
- *Image Sampling*: Sampling of pixels in the image plane. The parameter defined the number of pixels for the grid in one direction so the total amount of pixels is the square of the number entered.
- *[µm]*: This parameter defines the total half-size of the area on the image surface for the calculation of the PSF to be included in μm. If set to 0.0 the size will be estimated from the size of the Airy-disk. For aberrated systems this may lead to a cropping of the PSF. Therefore it is recommended to check the PSF before relying on the MTF plot.
- *Color by*: Color the individual plots by "Wavelength" or by "Field"
- *Frequency*: Frequency at which to evaluate the MTF
- *Fan Type*: Direction of the field scan
- *Steps*: Number of steps for the sampling of the fields

17.6.9  Geometrical MTF vs. Field Plot

The *Geometrical MTF vs. Field Plot* shows the value of the MTF as computed by the *Geometrical MTF Plot* for a certain frequency over the field, in several steps defined by *Steps*.

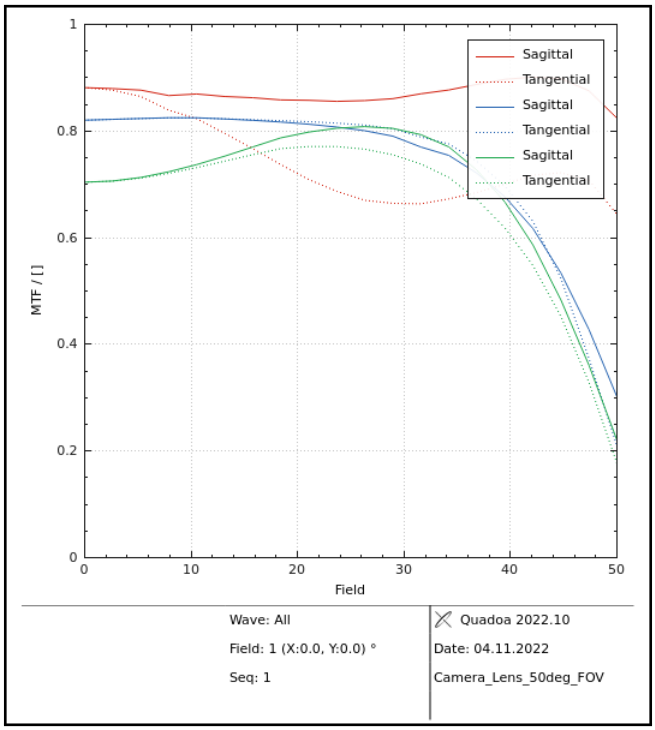


Figure 17.33: Example of Geometrical MTF vs. Field Plot

The plot has the following parameters:

- *Sequence*: Sequence that defines the path of the light (order of Surfaces), Fields and Wavelengths for the raytrace.
- *Wavelength*: Allows to select one specific "All" or "Polychromatic". If "Polychromatic" is selected the PSF is computed for each wavelength of the Sequence and the result is the summed intensity of all wavelengths.
- *Config*: Allows to select a specific state of the system defined by the Multiconfig Lookuptable. If "Current" is selected the plot will show the result for the config that is currently active.
- *Pupil Sampling*: Number of rays in x, and y direction for the rasterization of the pupil plane that is used for PSF computation.
- *Image Sampling*: Rasterization of the pixels on the image surface
- *Color by*: Color the individual plots by "Wavelength" or by "Field"
- *Frequency*: Frequency at which to evaluate the MTF
- *Fan Type*: Direction of the field scan

- *Steps*: Number of steps for the sampling of the fields

17.6.10 FFT Enclosed Energy Plot

The *FFT Enclosed Energy Plot* shows the energy that is contained within an certain area. This is achieved by integrating the PSF as computed by the *FFT PSF Plot* over a a varying region in several steps defined by *Steps* and plotting the relative energy that lies within that region.

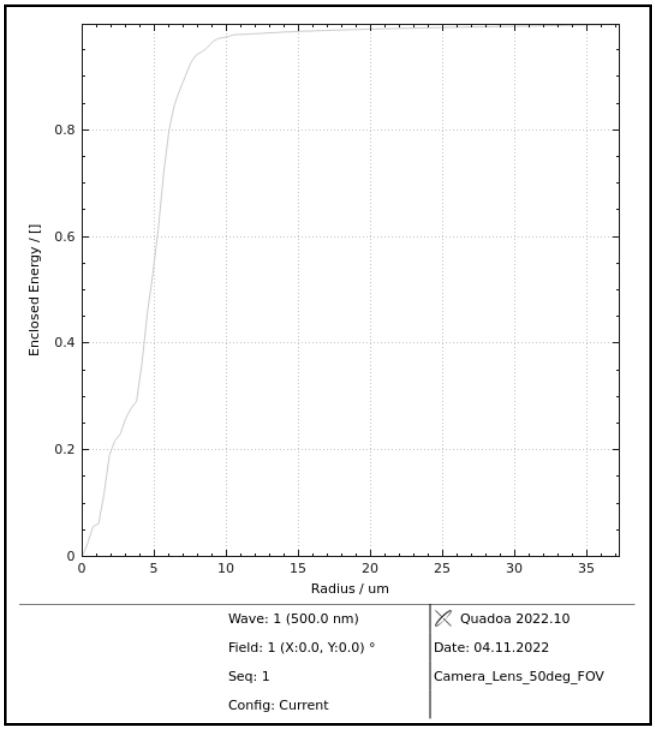


Figure 17.34: Example FFT Enclosed Energy Plot

The plot has the following parameters:

- *Sequence*: Sequence that defines the path of the light (order of Surfaces), Fields and Wavelengths for the raytrace.
- *Field*: Allows to select the Field
- *Wavelength*: Allows to select one specific or "Polychromatic". If "Polychromatic" is selected the PSF is computed for each wavelength of the Sequence and the result is the summed intensity of all wavelengths.
- *Config*: Allows to select a specific state of the system defined by the Multiconfig Lookuptable. If "Current" is selected the plot will show the result for the config that is currently active.

- *Pupil Sampling*: Sampling of rays in the pupil plane. The parameter defined the number of rays for the grid in one direction so the total amount of rays to be traced is the square of the number entered.
- *Image Sampling*: Sampling of pixels in the image plane. The parameter defined the number of pixels for the grid in one direction so the total amount of pixels is the square of the number entered.
- *Image Scale [µm]*: This parameter defines the total half-size of the area on the image surface to be plotted in μm. If set to 0.0 the size will be estimated from the size of the Airy-disk.
- *Enclose Type*: "Encircle" plots the encirceld energy over the radius of the circular area. "Ensquared" plots the ensquared energy inside a quadratic area ranging from $\pm r$ in x and y direction. "Range X" plots the enclosed energy inside an area that is infinite in y direction and reaches from $-r$ to $+r$ in x direction. "Range Y" plots the enclosed energy inside an area that is infinite in x direction and reaches from $-r$ to $+r$ in y direction.

17.6.11  Huygens Enclosed Energy Plot

The *Huygens Enclosed Energy Plot* shows the energy that is contained within an certain area. This is achieved by integrating the PSF as computed by the *Huygens PSF Plot* over a a varying region in several steps defined by *Steps* and plotting the relative energy that lies within that region.

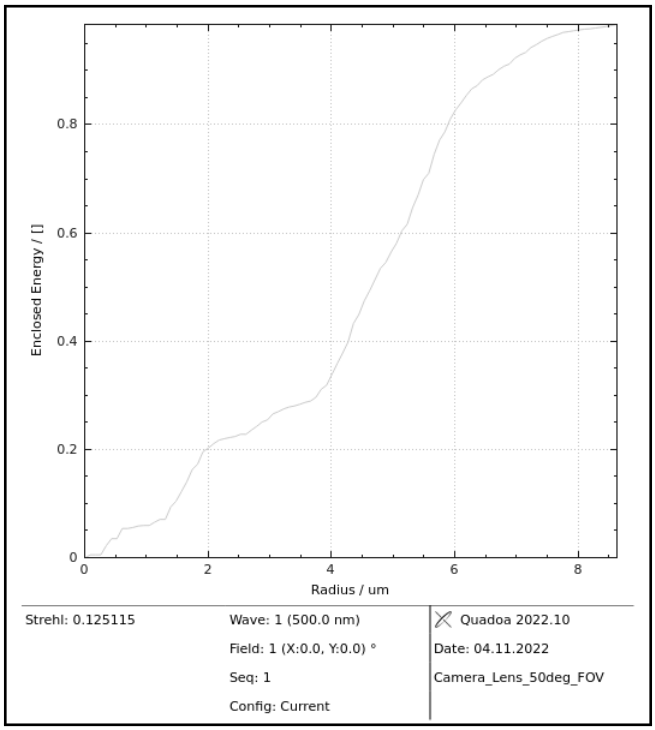


Figure 17.35: Example Huygens Enclosed Energy Plot

The plot has the following parameters:

- *Sequence*: Sequence that defines the path of the light (order of Surfaces), Fields and Wavelengths for the raytrace.
- *Field*: Allows to select the Field
- *Wavelength*: Allows to select one specific or "Polychromatic". If "Polychromatic" is selected the PSF is computed for each wavelength of the Sequence and the result is the summed intensity of all wavelengths.
- *Config*: Allows to select a specific state of the system defined by the Multiconfig Lookuptable. If "Current" is selected the plot will show the result for the config that is currently active.
- *Pupil Sampling*: Sampling of rays in the pupil plane. The parameter defined the number of rays for the grid in one direction so the total amount of rays to be traced is the square of the number entered.
- *Image Sampling*: Sampling of pixels in the image plane. The parameter defined the number of pixels for the grid in one direction so the total amount of pixels is the square of the number entered.

- *Image Scale [µm]*: This parameter defines the total half-size of the area on the image surface to be plotted in μm. If set to 0.0 the size will be estimated from the size of the Airy-disk.
- *Enclose Type*: "Encircle" plots the encirceld energy over the radius of the circular area. "Ensquared" plots the ensquared energy inside a quadratic area ranging from $\pm r$ in x and y direction. "Range X" plots the enclosed energy inside an area that is infinite in y direction and reaches from $-r$ to $+r$ in x direction. "Range Y" plots the enclosed energy inside an area that is infinite in x direction and reaches from $-r$ to $+r$ in y direction.

17.6.12 Geometrical Enclosed Energy Plot

The *Geometrical Enclosed Energy Plot* shows the energy that is contained within an certain area. This is achieved by integrating the PSF as computed by the *Geometrical PSF Plot* over a a varying region in several steps defined by *Steps* and plotting the relative energy that lies within that region.

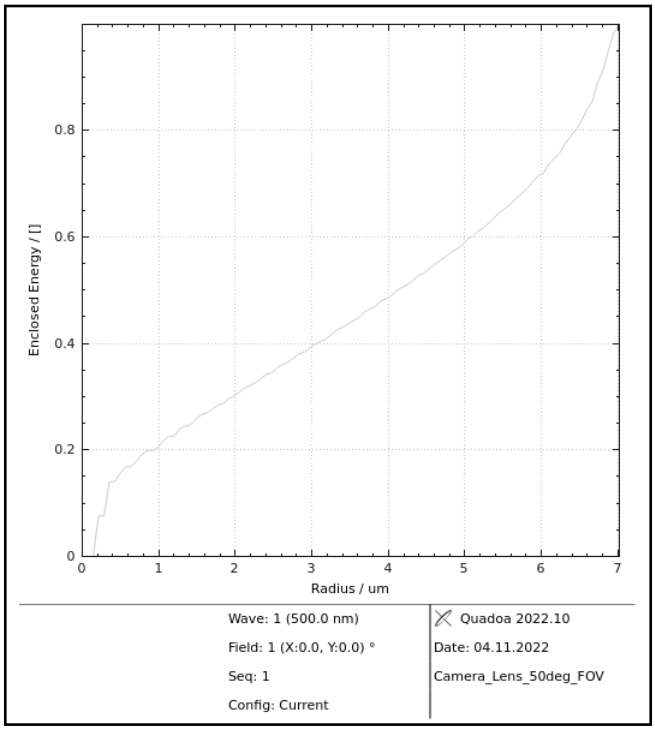


Figure 17.36: Example Geometrical Enclosed Energy Plot

The plot has the following parameters:

- *Sequence*: Sequence that defines the path of the light (order of Surfaces), Fields and Wavelengths for the raytrace.

- *Field*: Allows to select the Field
- *Wavelength*: Allows to select one specific or "Polychromatic". If "Polychromatic" is selected the PSF is computed for each wavelength of the Sequence and the result is the summed intensity of all wavelengths.
- *Config*: Allows to select a specific state of the system defined by the Multiconfig Lookuptable. If "Current" is selected the plot will show the result for the config that is currently active.
- *Pupil Sampling*: Number of rays in x, and y direction for the rasterization of the pupil plane that is used for PSF computation.
- *Image Sampling*: Rasterization of the pixels on the image surface
- *Enclose Type*: "Encircle" plots the encirceld energy over the radius of the circular area. "Ensquared" plots the ensquared energy inside a quadratic area ranging from $\pm r$ in x and y direction. "Range X" plots the enclosed energy inside an area that is infinite in y direction and reaches from $-r$ to $+r$ in x direction. "Range Y" plots the enclosed energy inside an area that is infinite in x direction and reaches from $-r$ to $+r$ in y direction.

17.7 Irradiance

17.7.1  Incoherent Irradiance Plot

The *Incoherent Irradiance Plot* shows the intensity distribution of rays of a Sequence on the image surface. For the evaluation the image surface is rasterized into single pixels, where the amount of pixels is defined by *Pixel Sampling*. For the tracing a large number of rays is traced at random positions in through the pupil, while the pupil apodization and total amount of power is defined by the Apodization of the Sequence. For each pixel the intensity of all rays that hit that pixel is summed up and the result is divided by the pixel area leading to a result in units of $\frac{W}{m^2}$. If the *Image Space* parameter of the Sequence is set to "Afocal" the pixelation will be in angular instead of lateral position coordinates. In this case the unit is $\frac{W}{degree^2}$.

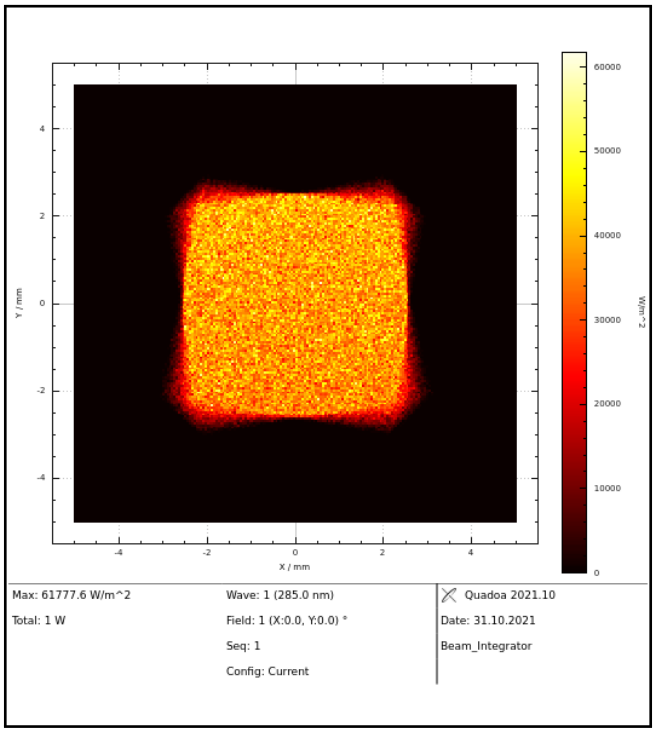


Figure 17.37: Example of Incoherent Irradiance Plot

The plot has the following parameters:

- *Sequence*: Sequence that defines the path of the light (order of Surfaces), Fields and Wavelengths for the raytrace.
- *Field*: Allows to select one specific or "All" Field. If "All" is selected one plot per Field is plotted.
- *Wavelength*: Allows to select one specific or "All" Wavelength.
- *Config*: Allows to select a specific state of the system defined by the Multiconfig Lookuptable. If "Current" is selected the plot will show the result for the config that is currently active.
- *Pixel Sampling*: Number of pixels in x, and y direction for the rasterization of the image plane.
- *Number Rays*: Number of rays to be traced
- *Distribution*: "Default" traces a large number of rays (defined by "Number Rays") at random positions in the pupil plane. "X-Fan" and "Y-Fan" will trace the rays through a random position in the pupil however the 2nd coordinate will always be 0.0.

17.7.2 Coherent Irradiance Plot

The *Coherent Irradiance Plot* shows the intensity distribution of rays of a Sequence on the image surface. For the evaluation the image surface is rasterized into single pixels, where the amount of pixels is defined by *Pixel Sampling*. For the tracing a large number of rays is traced at random positions in through the pupil, while the pupil apodization and total amount of power is defined by the Apodization of the Sequence. For each pixel the complex amplitude of all rays that hit that pixel is summed up. When the rays hit an individual pixel at an arbitrary position that is not at the center of the pixel under an angle a phase error is introduced. This error is corrected prior to summing the complex amplitude. The intensity I is obtained by $I = E^2$. The result is normed so that the brightest pixel has an intensity of 1.0. This plot is not available if the *Image Space* parameter of the Sequence is set to "Afocal".

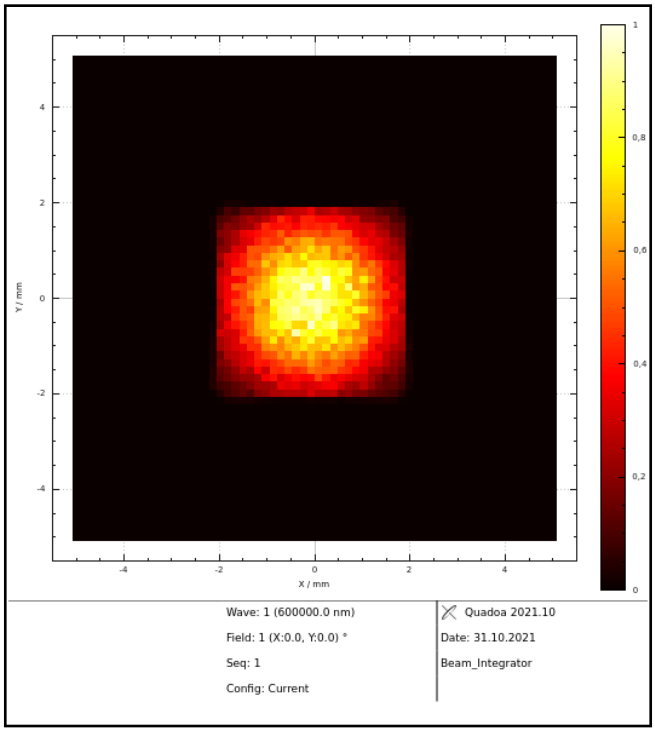


Figure 17.38: Example of Coherent Irradiance Plot

The plot has the following parameters:

- *Sequence*: Sequence that defines the path of the light (order of Surfaces), Fields and Wavelengths for the raytrace.
- *Field*: Allows to select one specific or "All" Field. If "All" is selected one plot per Field is plotted.

- *Wavelength*: Allows to select one specific or "All" Wavelength.
- *Config*: Allows to select a specific state of the system defined by the Multiconfig Lookuptable. If "Current" is selected the plot will show the result for the config that is currently active.
- *Pixel Sampling*: Number of pixels in x, and y direction for the rasterization of the image plane.
- *Number Rays*: Number of rays to be traced
- *Distribution*: "Default" traces a large number of rays (defined by "Number Rays") at random positions in the pupil plane. "X-Fan" and "Y-Fan" will trace the rays through a random position in the pupil however the 2nd coordinate will always be 0.0.

17.7.3 Ghost Irradiance Plot

The *Ghost Irradiance Plot* is used to visualize the impact of ghost reflections on the image surface. The computation is equal to the calculation of the Incoherent Image Analysis Plot. However the intensity on the image plane gets summed of for any Sequence that is marked as Ghost Sequence and that has the selected image surface (defined by *Surface*) in common.

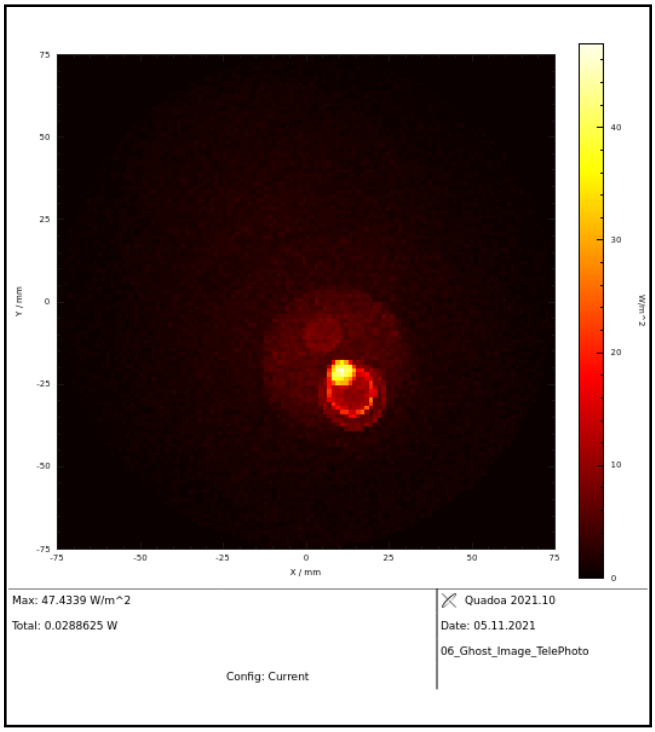


Figure 17.39: Example of Ghost Image Analysis Plot

The plot has the following parameters:

- *Surface*: Selects the image surface at which to evaluate the intensity. If Ghost Sequences with different image surfaces (e.g. a set of first order and a set of second order ghosts) are defined, only those that end at the selected surface will contribute to the plot.
- *Config*: Allows to select a specific state of the system defined by the Multiconfig Lookuptable. If "Current" is selected the plot will show the result for the config that is currently active.
- *Pixel Sampling*: Number of pixels in x, and y direction for the rasterization of the image plane.
- *Number Rays*: Number of rays to be traced
- *Angular Range*: If the *Image Space* parameter of the Sequence is set to "Afocal" this parameter defines the maximum angel to include in the plot.

17.8 Image Analysis

17.8.1 Geometric Image Analysis Plot

The *Geometric Image Analysis Plot* simulates the imaging quality of a lens by directly simulating the appearance of the image on the image plane. The plot covers effects like aberrations and chromatic aberrations of the lens, distortion, (chromatic) intensity changes due to coatings and or polarization. The plot works by computing a set of geometric point spread functions over the image field. Then the original image is convoluted with the point spread functions that are interpolated for each pixel and color channel in the image. For lenses that are close to diffraction limit see also the *Huygens Image Analysis Plot*

Figure 17.40: Example of Incoherent Irradiance Plot

The plot has the following parameters:

- *Sequence*: Sequence that defines the path of the light (order of Surfaces), Fields and Wavelengths for the raytrace.
- *Config*: Allows to select a specific state of the system defined by the Multiconfig Lookuptable. If "Current" is selected the plot will show the result for the config that is currently active.
- *Image*: Allows to select a test image for the simulation. The image file has to be stored in imagesimulation folder in the User Data Folder. The default path is `C:\Users\UserName\Quadoa\imagesimulation` on Windows or `/home/UserName/Quadoa/imagesimulation` on Linux.
- *Red Channel*: Wavelength in nm that is associated with the red pixels in the image, but doesn't necessarily have to be red
- *Green Channel*: Wavelength in nm that is associated with the green pixels in the image, but doesn't necessarily have to be green
- *Blue Channel*: Wavelength in nm that is associated with the blue pixels in the image, but doesn't necessarily have to be blue
- *Rotate Image 180*: If checked the image will be rotated by $180°$ after the computation (just for display)

- *PSF Pupil Sampling*: Number of rays in x, and y direction for the rasterization of the pupil plane that is used for PSF computation.
- *PSF Grid Sampling*: Number of PSFs to compute along the field axis of the image.
- *Object Half Height*: If set to auto the y-size of the image in object space is assumed to be the maximum value of the y-field. If a value is entered, the image is assumed to be this height in object space, where the unit in which the size is defined depends on the *Field Type* setting of the Sequence.

17.8.2 Huygens Image Analysis Plot

The *Huygens Image Analysis Plot* is comparable to the *Geometric Image Analysis Plot* however instead of a geometric PSF it uses the Huygens PSF for the convolution. If the system is not close to being diffraction limited it is recommended to use the *Geometric Image Analysis Plot* since it is faster and can deal with larger aberrations. Furthermore, it is recommended to check the PSFs by simulating a test-image with single pixel dots on it or directly the *Huygens PSF Plot* to be sure the result is reliable.

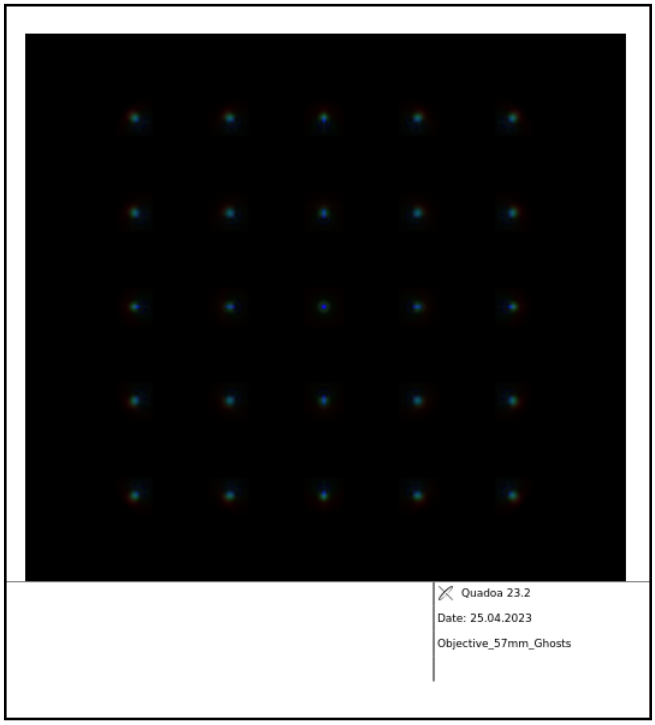


Figure 17.41: Example of Incoherent Irradiance Plot

The plot has the following parameters:

- *Sequence*: Sequence that defines the path of the light (order of Surfaces), Fields and Wavelengths for the raytrace.
- *Config*: Allows to select a specific state of the system defined by the Multiconfig Lookuptable. If "Current" is selected the plot will show the result for the config that is currently active.
- *Image*: Allows to select a test image for the simulation. The image file has to be stored in imagesimulation folder in the User Data Folder. The default path is `C:\Users\UserName\Quadoa\imagesimulation` on Windows or `/home/UserName/Quadoa/imagesimulation` on Linux.
- *Red Channel*: Wavelength in nm that is associated with the red pixels in the image, but doesn't necessarily have to be red
- *Green Channel*: Wavelength in nm that is associated with the green pixels in the image, but doesn't necessarily have to be green
- *Blue Channel*: Wavelength in nm that is associated with the blue pixels in the image, but doesn't necessarily have to be blue
- *Rotate Image 180*: If checked the image will be rotated by $180°$ after the computation (just for display)
- *PSF Image Sampling*: Number of pixels in x, and y direction for the rasterization of the image plane that is used for PSF computation.
- *PSF Pupil Sampling*: Number of rays in x, and y direction for the rasterization of the pupil plane that is used for PSF computation.
- *PSF Grid Sampling*: Number of PSFs to compute along the field axis of the image.
- *Object Half Height*: If set to auto the y-size of the image in object space is assumed to be the maximum value of the y-field. If a value is entered, the image is assumed to be this height in object space, where the unit in which the size is defined depends on the *Field Type* setting of the Sequence.

17.9 Wave Optics

In addition to ray tracing, where light is described as as set of rays that are traced from a source to the image plane, Quadoa provides the Wave Optics Analysis Toolbox, which uses a wavefront propagation algorithm to simulate the propagation of light through the optical system. The algorithm

approximates the wavefront by a set of beamlets that are propagated along their path through the system. In the image space the beamlets are added coherently to obtain the propagated wavefront. Since the propagation of one beamlet does not depend on the others, it is also possible to propagate beams along multiple Sequences and combine the result (see 17.9.5) to obtain the interference of the Sequences. Depending on the system and the effects to be simulated the wave optics approach can lead to more accurate results or show effects that could not be seen with ray tracing. The following list gives an overview on where it is recommended to use wave optics instead of pure ray tracing and where not. To get access to the wave optics analysis features the Quadoa Wave Optics Toolbox is required, which is not part of the basic Quadoa license.

- If the system has relatively large aberrations, which means that the geometric aberrations exceed the effects of diffraction, classical ray tracing typically is to be preferred since it is faster compared to the beamlet propagation algorithm and already covers the main effects on the PSF.
- If no long free space propagation parts are involved in the simulation and the PSF is close to diffraction limit without blocking apertures involved, the Huygens PSF is to be preferred. In contrary when the system is close to diffraction limit and long propagation distances or apertures that diffract the beam are involved, typically the beamlet propagation algorithm is to be preferred.
- If the light is not coherent beamlet propagation is not recommended since it assumes spatial coherence over the entire pupil as well as temporal coherence of all beamlets.
- For systems that do not focus the light to a point like e.g. the free space propagation of a laser beam or an axicon forming a Bessel beam typically beamlet propagation is to be preferred since the Huygens PSF does not work in these cases and cannot cover the effects from free space propagation.
- For systems with high spatial frequencies on optical surfaces it has to be made sure that the beamlets can sample those frequencies to obtain an accurate result.
- For very fast optics with high NA in image space beamlet propagation is not recommended.

17.9.1  Gaussian Beam PSF Plot

The *Gaussian Beam PSF Plot* shows the Point Spread Function of a Gaussian Beam. The Gaussian Beam is propagated by a method similar to the ABCD matrix formalism for paraxial optics. This means aberrations other than basic astigmatism are not taken into by this method. If the optical

system has non negligible aberrations or general astigmatism it it recommended to use the *Beamlet Propagation PSF Plot* instead.

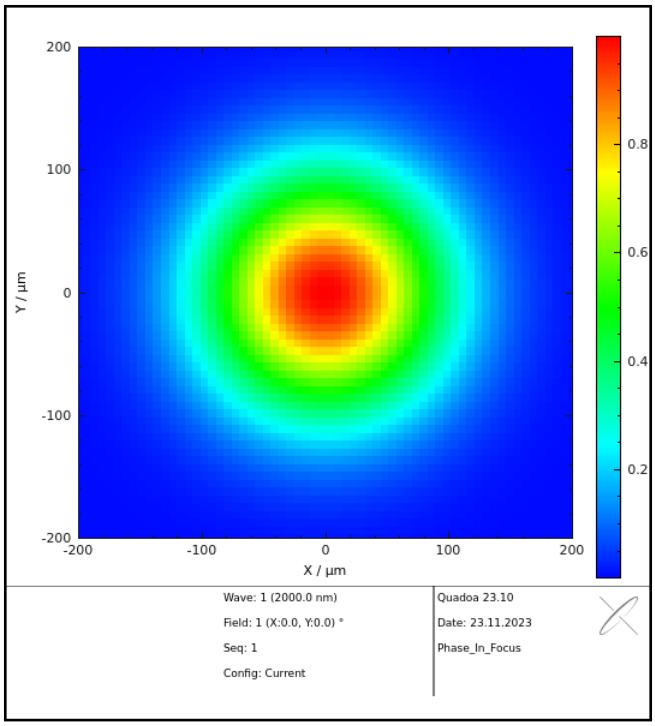


Figure 17.42: Example of Gaussian Beam PSF Plot

The plot has the following parameters:

- *Sequence*: Sequence that defines the path of the light (order of Surfaces), Fields and Wavelengths for the raytrace.
- *Field*: Allows to select the Field
- *Wavelength*: Allows to select the Wavelength
- *Config*: Allows to select a specific state of the system defined by the Multiconfig Lookuptable. If "Current" is selected the plot will show the result for the config that is currently active.
- *Image Sampling*: Sampling of pixels in the image plane. The parameter defined the number of pixels for the grid in one direction so the total amount of pixels is the square of the number entered.
- *Image Scale [µm]*: This parameter defines the total half-size of the area on the image surface for the calculation of the PSF to be included in μm.

- *PSF Type*: "Intensity" plots the intensity of the complex amplitude. "Phase" plots the phase of the complex amplitude. "Real" or "Imaginary" plot the real/imaginary component of the complex amplitude and the "Abs()" the absolute value of the component.
- *Axis Type*: "Linear" or "Logarithmic"

17.9.2 Beamlet Propagation PSF Plot

The *Beamlet Propagation PSF Plot* shows the Point Spread Function that is computed by the beamlet propagation in x, y coordinates of the image plane. Contrary to the *Huygens PSF* the *Beamlet Propagation PSF* can also be used for non focal systems, e.g. to simulate the propagation of a collimated beam or a Bessel beam.

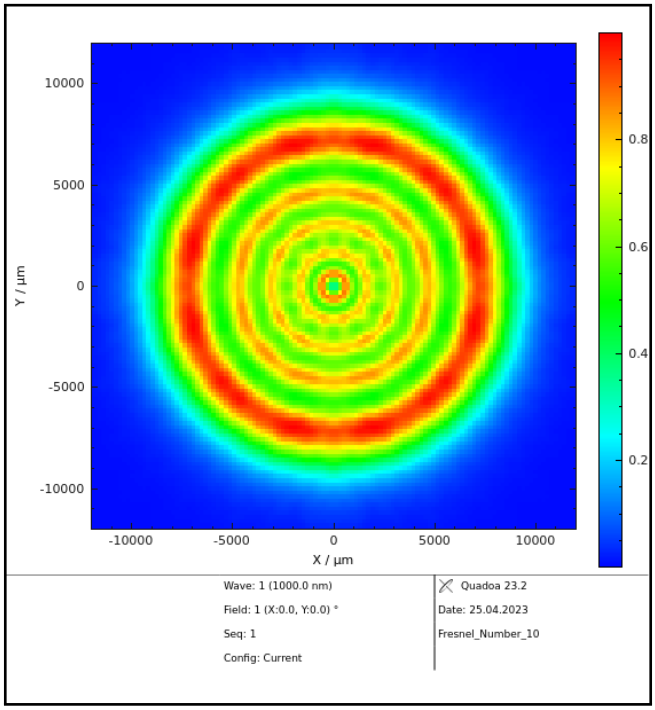


Figure 17.43: Example of Beamlet Propagation PSF Plot

The plot has the following parameters:

- *Sequence*: Sequence that defines the path of the light (order of Surfaces), Fields and Wavelengths for the raytrace.
- *Field*: Allows to select the Field
- *Wavelength*: Allows to select the Wavelength

- *Config*: Allows to select a specific state of the system defined by the Multiconfig Lookuptable. If "Current" is selected the plot will show the result for the config that is currently active.
- *Generate at Surface*: The default is 1. In this case the propagation will start directly from the first Surface in the Sequence. If any other Surface is chosen, the light will be traced by ray tracing to the selected surface. On the surface the phase and intensity distribution is used to generate the beamlets that are then propagated through the rest of the Sequence, which allows to simulate diffraction effects of intermediate blocking apertures.
- *Image Sampling*: Sampling of pixels in the image plane. The parameter defined the number of pixels for the grid in one direction so the total amount of pixels is the square of the number entered.
- *Image Scale [µm]*: This parameter defines the total half-size of the area on the image surface for the calculation of the PSF to be included in μm.
- *Pupil Sampling*: Sampling of beamlets in the pupil plane. The parameter defined the number of beamlets for the grid in one direction so the total amount of beamlets to be propagated is the square of the number entered.
- *Oversampling*: The overlap of beamlets to their neighbor beamlets. A large overlap leads to a smoother overall wavefront, however steep edges of the intensity profile (as e.g. in a top-hat profile) will also be smoothed out. A good compromise is typically between 2 and 4.
- *PSF Type*: "Intensity" plots the intensity of the complex amplitude. "Phase" plots the phase of the complex amplitude. "Real" or "Imaginary" plot the real/imaginary component of the complex amplitude and the "Abs()" the absolute value of the component.
- *Axis Type*: "Linear" or "Logarithmic"

17.9.3  Beamlet Propagation Through Focus PSF Plot

The *Beamlet Propagation Through Focus PSF Plot* shows the intensity or phase distribution of the electromagnetic field in the y-z plane. As a result a 2D cut through the focus of a system can be obtained. The parameter *Image Scale* defines the range in y as well as in z direction. Z = 0.0 is at the origin of the image surfaces coordinate system.

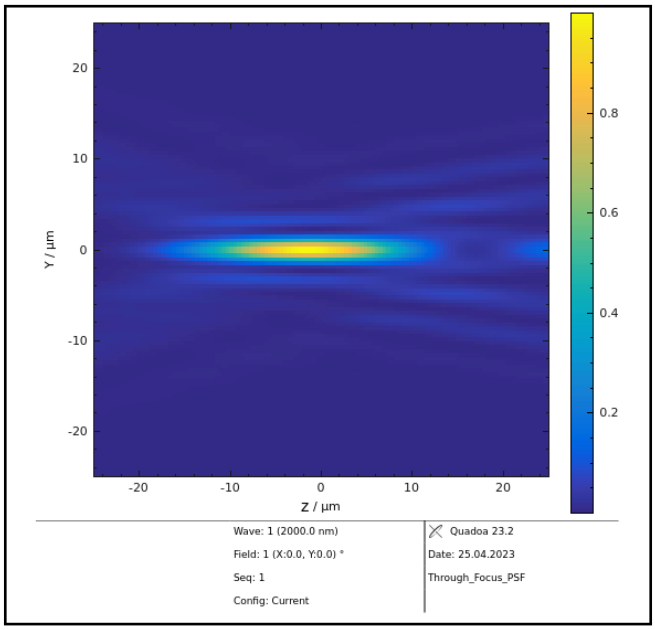


Figure 17.44: Example of Beamlet Propagation Through Focus PSF Plot

The plot has the following parameters:

- *Sequence*: Sequence that defines the path of the light (order of Surfaces), Fields and Wavelengths for the raytrace.
- *Field*: Allows to select the Field
- *Wavelength*: Allows to select the Wavelength
- *Config*: Allows to select a specific state of the system defined by the Multiconfig Lookuptable. If "Current" is selected the plot will show the result for the config that is currently active.
- *Generate at Surface*: The default is 1. In this case the propagation will start directly from the first Surface in the Sequence. If any other Surface is chosen, the light will be traced by ray tracing to the selected surface. On the surface the phase and intensity distribution is used to generate the beamlets that are then propagated through the rest of the Sequence, which allows to simulate diffraction effects of intermediate blocking apertures.
- *Image Sampling*: Sampling of pixels in the image plane. The parameter defined the number of pixels for the grid in one direction so the total amount of pixels is the square of the number entered.
- *Image Scale [µm]*: This parameter defines the total half-size of the area on the image surface for the calculation of the PSF to be included in μm.

- *Pupil Sampling*: Sampling of beamlets in the pupil plane. The parameter defined the number of beamlets for the grid in one direction so the total amount of beamlets to be propagated is the square of the number entered.
- *Oversampling*: The overlap of beamlets to their neighbor beamlets. A large overlap leads to a smoother overall wavefront, however steep edges of the intensity profile (as e.g. in a top-hat profile) will also be smoothed out. A good compromise is typically between 2 and 4.
- *PSF Type*: "Intensity" plots the intensity of the complex amplitude. "Phase" plots the phase of the complex amplitude. "Real" or "Imaginary" plot the real/imaginary component of the complex amplitude and the "Abs()" the absolute value of the component.
- *Axis Type*: "Linear" or "Logarithmic"

17.9.4  Gaussian Beam Interferogram Plot

The *Gaussian Beam Interferogram Plot* shows the intensity or phase distribution of the electromagnetic field in the x, y plane that is obtained by interfering (coherently adding) the fields of the two Sequences. The Gaussian Beam for each Sequence is propagated by a method similar to the ABCD matrix formalism for paraxial optics. This means that aberrations other than basic astigmatism are not taken into by this method. If the optical system has non negligible aberrations or general astigmatism it it recommended to use the *Beamlet Propagation Interferogram Plot* instead.

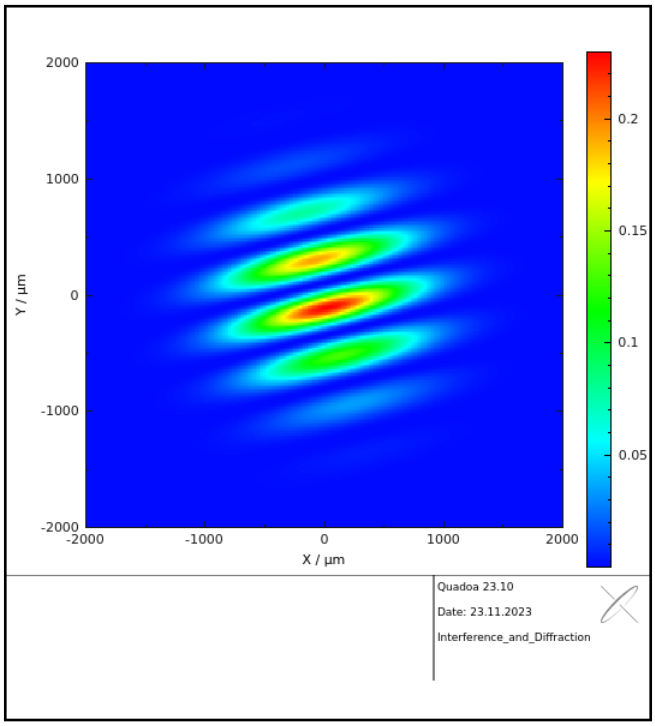


Figure 17.45: Example of Beamlet Propagation Interferogram Plot

The plot has the following parameters:

- *Sequence 1*: Sequence that defines the path of the light (order of Surfaces), Fields and Wavelengths for the raytrace.
- *Field 1*: Allows to select the Field for the first Sequence
- *Wavelength 1*: Allows to select the Wavelength for the first Sequence. The wavelength must be the same for both Sequence
- *Config 1*: Allows to select a specific state of the system defined by the Multiconfig Lookuptable for the first Sequence. If "Current" is selected the plot will show the result for the config that is currently active.
- *Sequence 2*: Sequence that defines the path of the light (order of Surfaces), Fields and Wavelengths for the raytrace.
- *Field 2*: Allows to select the Field for the second Sequence
- *Wavelength 2*: Allows to select the Wavelength for the second Sequence. The wavelength must be the same for both Sequence

- *Config 2*: Allows to select a specific state of the system defined by the Multiconfig Lookuptable for the second Sequence. If "Current" is selected the plot will show the result for the config that is currently active.
- *Image Sampling*: Sampling of pixels in the image plane. The parameter defined the number of pixels for the grid in one direction so the total amount of pixels is the square of the number entered.
- *Image Scale [µm]*: This parameter defines the total half-size of the area on the image surface for the calculation of the PSF to be included in μm.
- *PSF Type*: "Intensity" plots the intensity of the complex amplitude. "Phase" plots the phase of the complex amplitude.
- *Axis Type*: "Linear" or "Logarithmic"

17.9.5 Beamlet Propagation Interferogram Plot

The *Beamlet Propagation Interferogram Plot* shows the intensity or phase distribution of the electromagnetic field in the x, y plane that is obtained by interfering (coherently adding) the fields of the two Sequences. The fields are propagated by the beamlet propagation method.

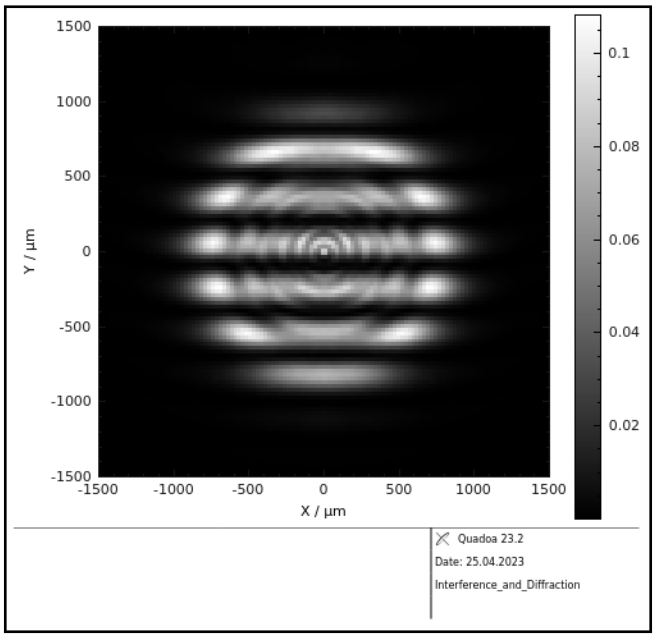


Figure 17.46: Example of Beamlet Propagation Interferogram Plot

The plot has the following parameters:

- *Sequence 1*: Sequence that defines the path of the light (order of Surfaces), Fields and Wavelengths for the raytrace.
- *Field 1*: Allows to select the Field for the first Sequence
- *Wavelength 1*: Allows to select the Wavelength for the first Sequence. The wavelength must be the same for both Sequence
- *Config 1*: Allows to select a specific state of the system defined by the Multiconfig Lookuptable for the first Sequence. If "Current" is selected the plot will show the result for the config that is currently active.
- *Sequence 2*: Sequence that defines the path of the light (order of Surfaces), Fields and Wavelengths for the raytrace.
- *Field 2*: Allows to select the Field for the second Sequence
- *Wavelength 2*: Allows to select the Wavelength for the second Sequence. The wavelength must be the same for both Sequence
- *Config 2*: Allows to select a specific state of the system defined by the Multiconfig Lookuptable for the second Sequence. If "Current" is selected the plot will show the result for the config that is currently active.
- *Generate at Surface*: The default is 1. In this case the propagation will start directly from the first Surface in the Sequence. If any other Surface is chosen, the light will be traced by ray tracing to the selected surface. On the surface the phase and intensity distribution is used to generate the beamlets that are then propagated through the rest of the Sequence, which allows to simulate diffraction effects of intermediate blocking apertures.
- *Image Sampling*: Sampling of pixels in the image plane. The parameter defined the number of pixels for the grid in one direction so the total amount of pixels is the square of the number entered.
- *Image Scale [µm]*: This parameter defines the total half-size of the area on the image surface for the calculation of the PSF to be included in μm.
- *Pupil Sampling*: Sampling of beamlets in the pupil plane. The parameter defined the number of beamlets for the grid in one direction so the total amount of beamlets to be propagated is the square of the number entered.

- *Oversampling*: The overlap of beamlets to their neighbor beamlets. A large overlap leads to a smoother overall wavefront, however steep edges of the intensity profile (as e.g. in a top-hat profile) will also be smoothed out. A good compromise is typically between 2 and 4.
- *PSF Type*: "Intensity" plots the intensity of the complex amplitude. "Phase" plots the phase of the complex amplitude.
- *Axis Type*: "Linear" or "Logarithmic"

17.9.6 Gaussian Beam Report

The *Gaussian Beam Report* shows the beam waist, waist position, divergence and rayleigh length for a gaussian beam. This report is only available for a Sequence that has the *Source Type* set to "Gaussian Beam".

The plot has the following parameters:

- *Sequence*: Sequence that defines the path of the light (order of Surfaces), Fields and Wavelengths for the raytrace.
- *Field*: Allows to select the Field.
- *Wavelength*: Allows to select the Wavelength.
- *Config*: Allows to select a specific state of the system defined by the Multiconfig Lookuptable. If "Current" is selected the plot will show the result for the config that is currently active.

17.10 Fiber Coupling

The Fiber Coupling Tools build into Quadoa allow the calculation of the fiber coupling efficiency when coupling light into a multi-mode or single-mode fiber. For multi-mode fiber coupling a ray based coupling integral is used. This means, that to be coupled into the fiber the x,y coordinate of the incoming rays need to be inside the circle defined by the fiber core diameter and the angle of the ray needs to be within the numerical aperture that is accepted by the fiber. For single mode fiber coupling the mode overlap integral method is used (see chapter 27).

17.10.1 Multi Mode Fiber Coupling Efficiency

The *Multi Mode Fiber Coupling Efficiency* report computes the coupling efficiency into a multi mode fiber. To be coupled the incoming rays need to lie inside the radius defined by the fiber core and within the angle that is accepted by the fibers numerical aperture.

The report has the following parameters:

- *Sequence*: Sequence that defines the path of the light (order of Surfaces), Fields and Wavelengths for the raytrace.
- *Field*: Allows to select the Field
- *Wavelength*: Allows to select the Wavelength.
- *Config*: Allows to select a specific state of the system defined by the Multiconfig Lookuptable. If "Current" is selected the plot will show the result for the con fig that is currently active.
- *Definition By*: "By NA" allows to specify the numerical aperture of the fiber where the index of the cladding material will be computed from this value. "By Index" allows to specify the refractive index of the core and the cladding, and the numerical aperture will be computed from these values.
- *NA*: Maximum numerical aperture the fiber accepts
- *Core Diameter / um:* Diameter of the fiber core in μm
- *Core Ref. Index:* Refractive index of the fiber core at the wavelength to be coupled.
- *Cladding Ref. Index:* Refractive index of the fiber cladding at the wavelength to be coupled.
- *Incl. Fresnel Loss*: If checked Fresnel losses at the fiber end surface will we included in the calculation.
- *Incl. in System Loss*: If checked losses due to Fresnel losses or bulk absorption that emerge from inside the optical system will we included in the calculation.

17.10.2 Single Mode Fiber Coupling Efficiency

The *Single Mode Fiber Coupling Efficiency* report computes the coupling efficiency into a single mode fiber. The efficiency is computed by the mode matching method (see chapter 27). The fiber can be either specified by the $\frac{1}{e^2}$ mode field diameter or by the $\frac{1}{e^2}$ numerical aperture.

The report has the following parameters:

- *Sequence*: Sequence that defines the path of the light (order of Surfaces), Fields and Wavelengths for the raytrace.
- *Field*: Allows to select the Field
- *Wavelength*: Allows to select the Wavelength.
- *Config*: Allows to select a specific state of the system defined by the Multiconfig Lookuptable. If "Current" is selected the plot will show the result for the con fig that is currently active.
- *Pupil Sampling*: Sampling of rays in the pupil plane. The parameter defined the number of rays for the grid in one direction so the total amount of rays to be traced is the square of the number entered. The value is used for the Huygens PSF as directly for the pupil functions complex amplitude evaluation.
- *Image Sampling*: Sampling of pixels in the image plane (on the fiber end). The parameter defined the number of pixels for the grid in one direction so the total amount of pixels is the square of the number entered. This value is only relevant for the calculation of the Huygens PSF when the *Overlap Integral* is set to "On Fiber End".
- *Definition By*: "By NA" allows to specify the $\frac{1}{e^2}$ numerical aperture of the fiber where the index of the cladding material will be computed from this value. "By MFD" allows to specify the $\frac{1}{e^2}$ mode field diameter of the fiber.
- *Fiber NA(1/e^2)*: Numerical aperture $\frac{1}{e^2}$ of the fiber
- *Fiber MFD(1/e^2) / um:* $\frac{1}{e^2}$ mode field diameter of the fiber in μm
- *Incl. in System Loss*: If checked losses due to Fresnel losses or bulk absorption that emerge from inside the optical system will we included in the calculation.

17.10.3 Beamlet Propagation Fiber Coupling Efficiency

The *Beamlet Propagation Fiber Coupling Efficiency* report is similar to the *Single Mode Fiber Coupling Efficiency*. However, instead using the *Huygens PSF* to calculate the complex amplitude of the incoming field on the fiber end the *Beamlet Propagation Fiber Coupling Efficiency* report uses the *Beamlet Propagation PSF*.

The report has the following parameters:

- *Sequence*: Sequence that defines the path of the light (order of Surfaces), Fields and Wavelengths for the raytrace.
- *Field*: Allows to select the Field
- *Wavelength*: Allows to select the Wavelength.
- *Config*: Allows to select a specific state of the system defined by the Multiconfig Lookuptable. If "Current" is selected the plot will show the result for the con fig that is currently active.
- *Pupil Sampling*: Sampling of beamlets in the pupil plane. The parameter defined the number of beamlets for the grid in one direction so the total amount of beamlets to be traced is the square of the number entered.
- *Image Sampling*: Sampling of pixels in the image plane (on the fiber end). The parameter defined the number of pixels for the grid in one direction so the total amount of pixels is the square of the number entered.
- *Oversampling*: The overlap of beamlets to their neighbor beamlets. A large overlap leads to a smoother overall wavefront, however steep edges of the intensity profile (as e.g. in a top-hat profile) will also be smoothed out. A good compromise is typically between 2 and 4.
- *Definition By*: "By NA" allows to specify the $\frac{1}{e^2}$ numerical aperture of the fiber where the index of the cladding material will be computed from this value. "By MFD" allows to specify the $\frac{1}{e^2}$ mode field diameter of the fiber.
- *Fiber NA(1/e^2)*: Numerical aperture $\frac{1}{e^2}$ of the fiber
- *Fiber MFD(1/e^2) / um:* $\frac{1}{e^2}$ mode field diameter of the fiber in μm
- *Incl. in System Loss*: If checked losses due to Fresnel losses or bulk absorption that emerge from inside the optical system will we included in the calculation.

17.10.4 Multi Mode Fiber Coupling Efficiency Scan

The *Multi Mode Fiber Coupling Efficiency Scan* plot computes the coupling efficiency in the same way as the *Multi Mode Fiber Coupling Efficiency Report*. In addition it allows to scan along one axis in x,y or z direction or tilt the fiber in x or y direction and plot the efficiency over the scan range.

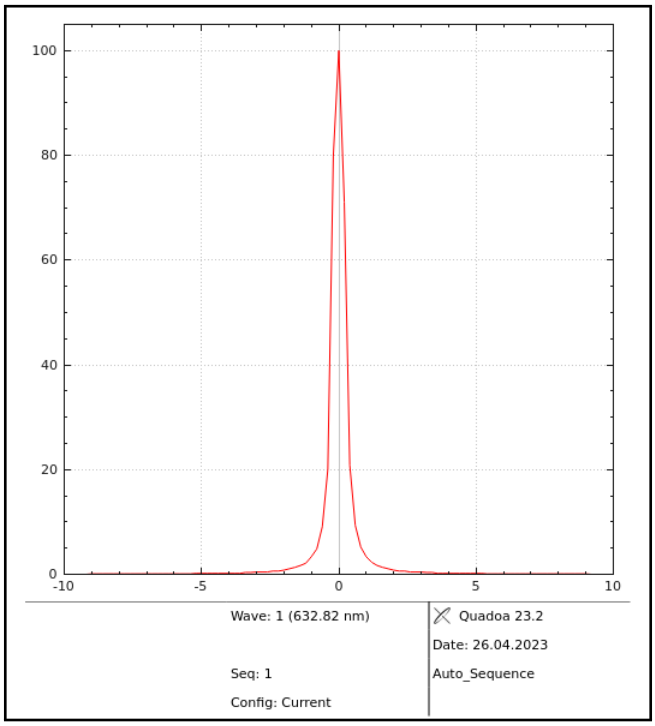


Figure 17.47: Example Multi Mode Fiber Coupling Scan Plot

The plot has the following parameters:

- *Sequence*: Sequence that defines the path of the light (order of Surfaces), Fields and Wavelengths for the raytrace.
- *Field*: Allows to select the Field
- *Wavelength*: Allows to select the Wavelength.
- *Config*: Allows to select a specific state of the system defined by the Multiconfig Lookuptable. If "Current" is selected the plot will show the result for the con fig that is currently active.
- *Definition By*: "By NA" allows to specify the numerical aperture of the fiber where the index of the cladding material will be computed from this value. "By Index" allows to specify the refractive index of the core and the cladding, and the numerical aperture will be computed from these values.
- *NA*: Maximum numerical aperture the fiber accepts
- *Core Diameter / um:* Diameter of the fiber core in μm
- *Core Ref. Index:* Refractive index of the fiber core at the wavelength to be coupled.
- *Cladding Ref. Index:* Refractive index of the fiber cladding at the wavelength to be coupled.

- *Incl. Fresnel Loss*: If checked Fresnel losses at the fiber end surface will we included in the calculation.
- *Incl. in System Loss*: If checked losses due to Fresnel losses or bulk absorption that emerge from inside the optical system will we included in the calculation.
- *Scan Direction*: Direction of the scan. Options are"Z", "X", "Y", "Rot X" or "Rot X"
- *Scan Range*: Rang from + the value to - the value entered around the nominal position. For rotation the unit is ○.

17.10.5 Single Mode Fiber Coupling Efficiency

The *Single Mode Fiber Coupling Efficiency Scan* plot computes the coupling efficiency in the same way as the *Single Mode Fiber Coupling Efficiency Report*. In addition it allows to scan along one axis in x,y or z direction or tilt the fiber in x or y direction and plot the efficiency over the scan range.

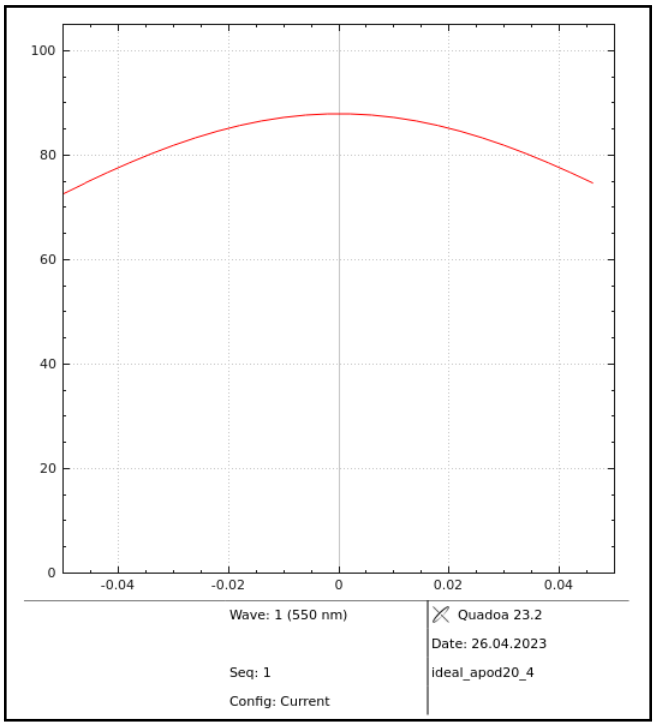


Figure 17.48: Example Single Mode Fiber Coupling Scan Plot

The report has the following parameters:

- *Sequence*: Sequence that defines the path of the light (order of Surfaces), Fields and Wavelengths for the raytrace.
- *Field*: Allows to select the Field
- *Wavelength*: Allows to select the Wavelength.
- *Config*: Allows to select a specific state of the system defined by the Multiconfig Lookuptable. If "Current" is selected the plot will show the result for the con fig that is currently active.
- *Pupil Sampling*: Sampling of rays in the pupil plane. The parameter defined the number of rays for the grid in one direction so the total amount of rays to be traced is the square of the number entered. The value is used for the Huygens PSF as directly for the pupil functions complex amplitude evaluation.
- *Image Sampling*: Sampling of pixels in the image plane (on the fiber end). The parameter defined the number of pixels for the grid in one direction so the total amount of pixels is the square of the number entered. This value is only relevant for the calculation of the Huygens PSF when the *Overlap Integral* is set to "On Fiber End".
- *Definition By*: "By NA" allows to specify the $\frac{1}{e^2}$ numerical aperture of the fiber where the index of the cladding material will be computed from this value. "By MFD" allows to specify the $\frac{1}{e^2}$ mode field diameter of the fiber.
- *Fiber NA(1/e2)*: Numerical aperture $\frac{1}{e^2}$ of the fiber
- *Fiber MFD(1/e2) / um:* $\frac{1}{e^2}$ mode field diameter of the fiber
- *Incl. in System Loss*: If checked losses due to Fresnel losses or bulk absorption that emerge from inside the optical system will we included in the calculation.
- *Scan Direction*: Direction of the scan. Options are"Z", "X", "Y", "Rot X" or "Rot X"
- *Scan Range*: Rang from + the value to - the value entered around the nominal position. For rotation the unit is ∘.

17.11 Polarization

17.11.1 Polarization Map

Visualization of the polarization state of a sequence at a surface interface via polarization ellipses.

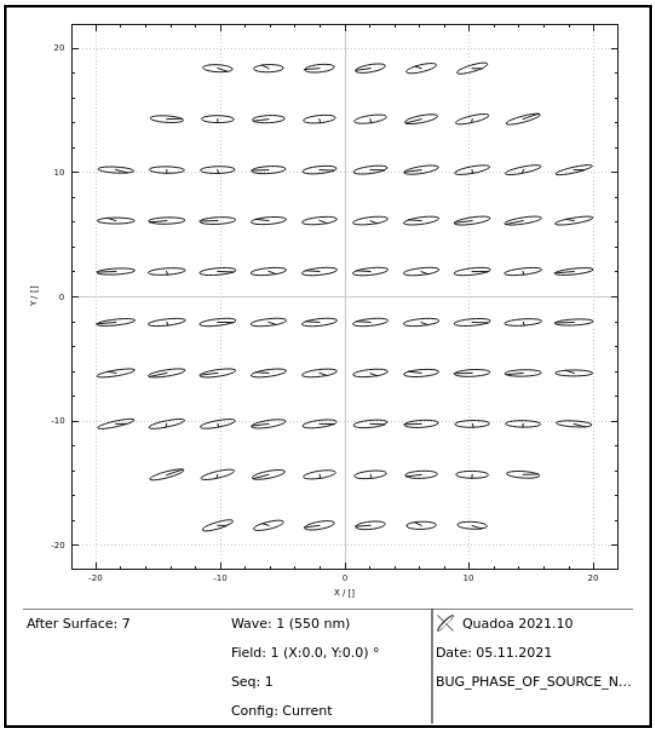


Figure 17.49: Example of Polarization Map

The plot has the following parameters:

- *Sequence*: Sequence that defines the path of the light (order of Surfaces), Fields and Wavelengths for the raytrace.
- *Field*: Allows to select the Field
- *Wavelength*: Allows to select the Wavelength.
- *Config*: Allows to select a specific state of the system defined by the Multiconfig Lookuptable. If "Current" is selected the plot will show the result for the config that is currently active.
- *Surface*: Surface at which to analyze the polarization
- *Sampling*: Rasterization of the pupil
- *Status:* Incoming ray before surface interaction or outgoing ray after surface interaction. *Note that these can have different local coordinate systems (15.1)*
- *Reference:* Reference for the evaluation of the polarization state. See 15.1 and 15.4.
- *Show Arrow:* Draw an indicator to visualize handedness.
- *Scale*: Size of the polarization ellipse drawn

17.11.2 Transmission Fan Plot

The *Transmission Fan Plot* shows the relative transmission of a fan of rays until to a certain Surface in the Sequence.

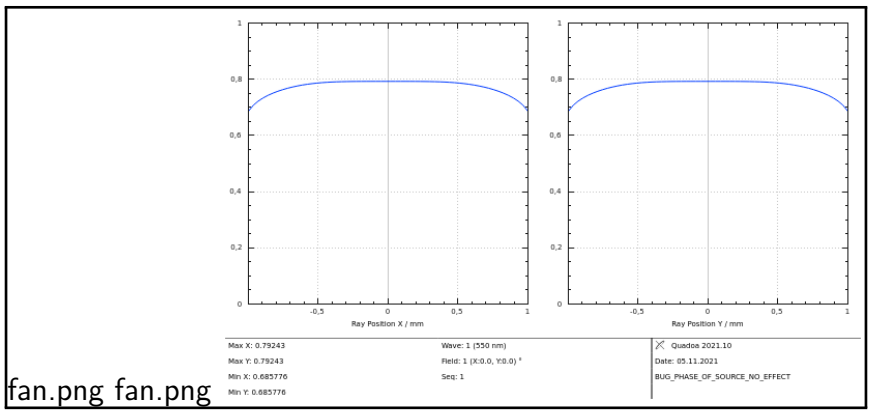


Figure 17.50: Example of Transmission Fan Plot

The plot has the following parameters:

- *Sequence*: Sequence that defines the path of the light (order of Surfaces), Fields and Wavelengths for the raytrace.
- *Field*: Allows to select the Field
- *Wavelength*: Allows to select the Wavelength.
- *Config*: Allows to select a specific state of the system defined by the Multiconfig Lookuptable. If "Current" is selected the plot will show the result for the config that is currently active.
- *Surface*: Surface at which to analyze the polarization
- *Sampling*: Rasterization of the pupil
- *Status:* Incoming ray before surface interaction or outgoing ray after surface interaction. *Note that these can have different local coordinate systems (15.1)*

17.11.3 Poincare Sphere

The *Poincare Sphere Plot* visualizes the polarization state of (partial) polarized light in form of a Poincare Sphere.

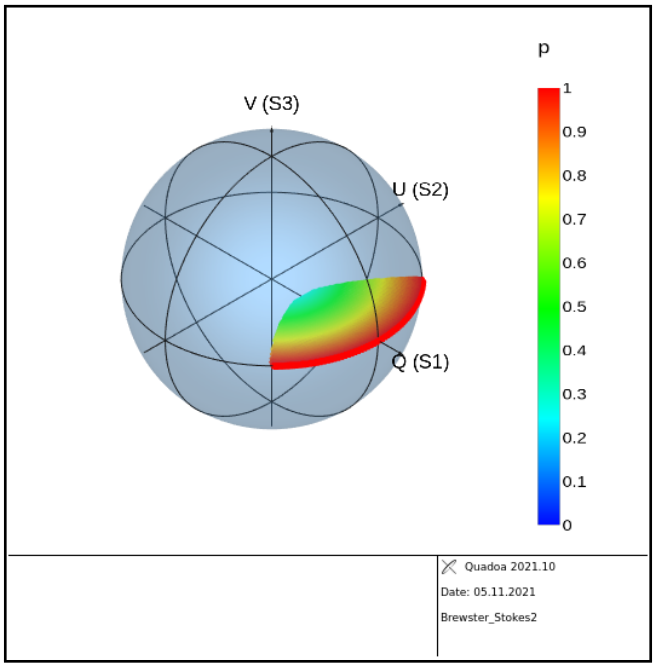


Figure 17.51: Example of Poincare Sphere Plot

The plot has the following parameters:

- *Sequence*: Sequence that defines the path of the light (order of Surfaces), Fields and Wavelengths for the raytrace.
- *Field*: Allows to select the Field
- *Wavelength*: Allows to select the Wavelength.
- *Config*: Allows to select a specific state of the system defined by the Multiconfig Lookuptable. If "Current" is selected the plot will show the result for the config that is currently active.
- *Surface*: Surface at which to analyze the polarization
- *Pupil Sampling*: Rasterization of the pupil
- *Status:* Incoming ray before surface interaction or outgoing ray after surface interaction. *Note that these can have different local coordinate systems (15.1)*
- *Reference:* Reference for the evaluation of the polarization state. See 15.1 and 15.4.
- *Show Projection*:

17.12 Lens

17.12.1 3D-View Window

Open another 3D View Editor to visualize the optical system in a 3D-view or to visualize the optical system in different configuration states. In contrast to the main 3D-View the *3D-View Windows* purpose is only for visualization but cannot be used for manipulating lenses.

The 3D-View has the following parameters:

- *Config*: Allows to select a specific state of the system defined by the Multiconfig Lookuptable. If "Current" is selected the plot will show the result for the config that is currently active.
- *Sequence*: show rays of a single or all Sequences in the view

17.12.2 Surface Form Plot

The *Surface Form Plot* shows the sag of a Surface in form of a 3D false-color plot or in Form of a Linecut.

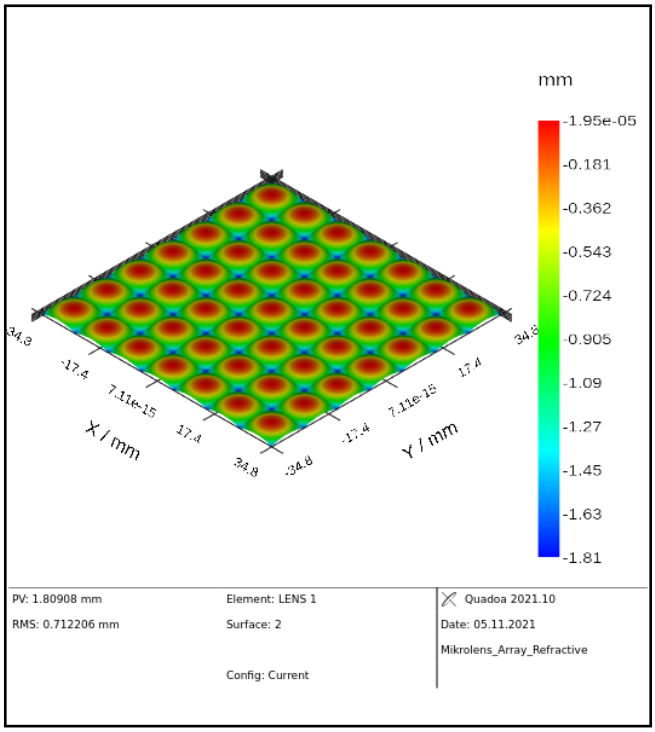


Figure 17.52: Example of Surface Form Plot

The plot has the following parameters:

- *Element*: Selects the Surface Lens, or other Element by ID
- *Surface*: Selectst the Surface if the Element consists of several Surfaces
- *Config*: Allows to select a specific state of the system defined by the Multiconfig Lookuptable. If "Current" is selected the plot will show the result for the config that is currently active.
- *Resolution*: Rasterization of the Surface

17.12.3 Surface Form Gradient Plot

The *Surface Form Gradient Plot* shows the gradient of a Surface in form of a 3D false-color plot or in Form of a Linecut.

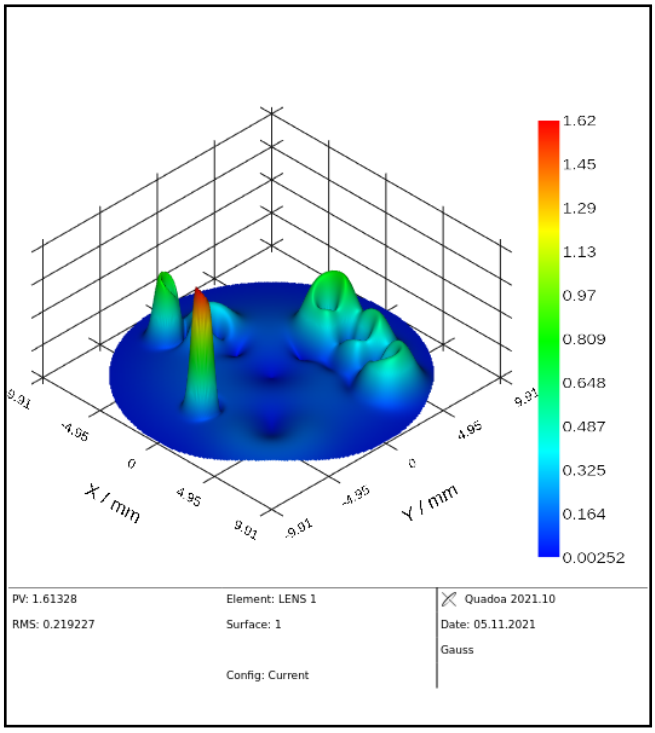


Figure 17.53: Example of Surface Form Gradient Plot

The plot has the following parameters:

- *Element*: Selects the Surface Lens, or other Element by ID
- *Surface*: Selectst the Surface if the Element consists of several Surfaces

- *Config*: Allows to select a specific state of the system defined by the Multiconfig Lookuptable. If "Current" is selected the plot will show the result for the config that is currently active.
- *Resolution*: Rasterization of the Surface
- *Direction*: Plot the gradient in x-direction "dx" in y-direction "dy" or the absolute gradient "|| ||"

17.12.4 Surface Transfer Plot

The *Surface Transfer Plot* shows the transmittance / reflectance or absorption of a surface in relationship to the angle of incident, wavelength and polarization state of a ray.

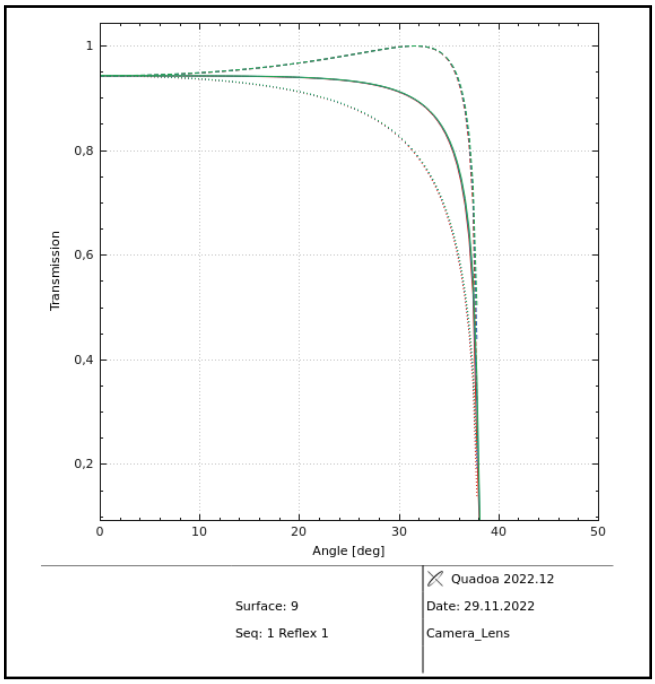


Figure 17.54: Example of Surface Transfer Plot

The plot has the following parameters:

- *Sequence*: Sequence that defines the Source from which the Wavelengths are obtained.
- *Wavelength*: Allows to select one specific Wavelength or "All".
- *Element*: Selects the Surface Lens, or other Element by ID
- *Surface*: Selectst the Surface if the Element consists of several Surfaces

- *Config*: Allows to select a specific state of the system defined by the Multiconfig Lookuptable. If "Current" is selected the plot will show the result for the config that is currently active.
- *Sampling*: Sampling of the plot
- *Type*: Plot the "Transmission", "Reflection" or "Absorption"
- *Variable*: Plot the transfer over "Angle of Incident" or "Wavelength"
- *Min [deg]*: Minimum angle of the range in the case *Variable* is set to "Angle of Incident"
- *Max [deg]*: Maximum angle of the range in the case *Variable* is set to "Angle of Incident"
- *Angle of Inc.*: Specifies the angle of incident in the case *Variable* is set to "Wavelength"
- *Show Polarization*: Show plot for S / P-polarized light

17.12.5 Surface Phase Plot

The *Surface Phase Plot* shows the phase function of a surface in form of a 3D false-color plot or in Form of a Linecut.

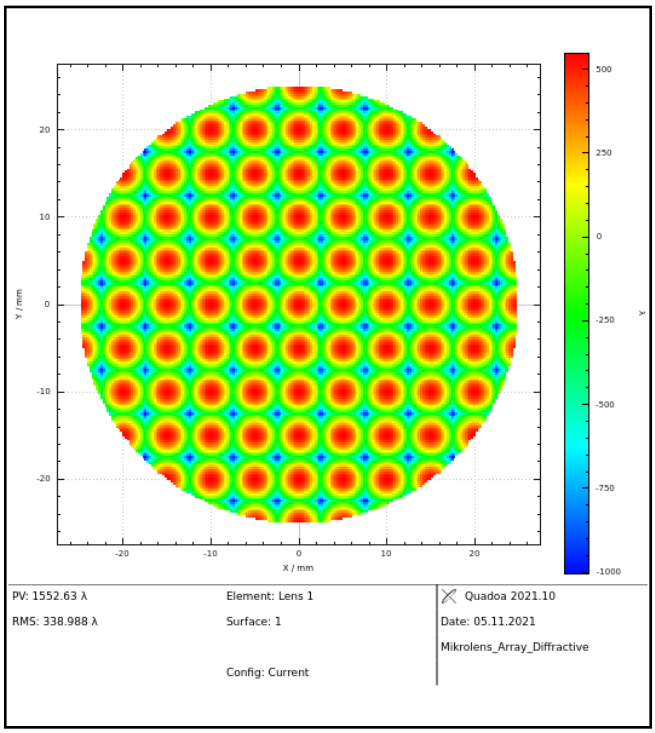


Figure 17.55: Example of Surface Phase Plot

The plot has the following parameters:

- *Element*: Selects the Surface Lens, or other Element by ID
- *Surface*: Selectst the Surface if the Element consists of several Surfaces
- *Config*: Allows to select a specific state of the system defined by the Multiconfig Lookuptable. If "Current" is selected the plot will show the result for the config that is currently active.
- *Resolution*: Rasterization of the Surface

17.12.6 Surface Phase Gradient Plot

The *Surface Phase Gradient Plot* shows the gradient of a Surfaces phase function in form of a 3D false-color plot or in Form of a Linecut.

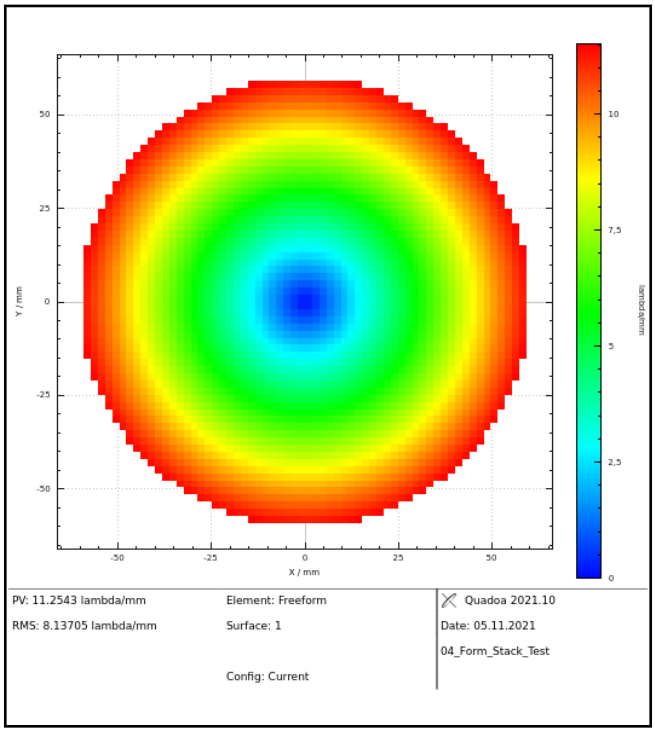


Figure 17.56: Example of Surface Phase Gradient Plot

The plot has the following parameters:

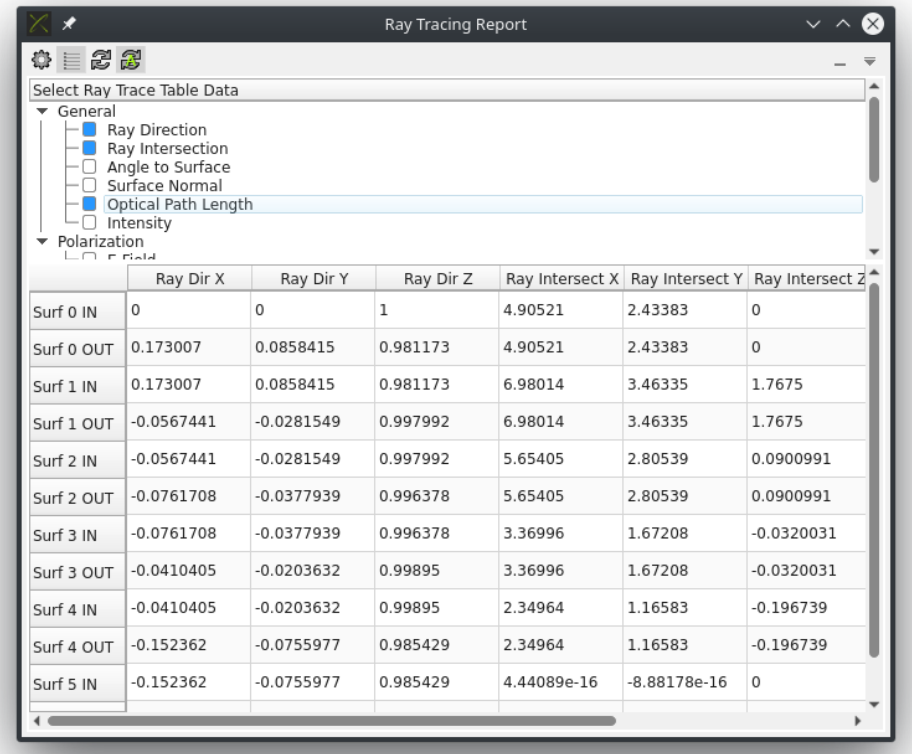
- *Element*: Selects the Surface Lens, or other Element by ID
- *Surface*: Selectst the Surface if the Element consists of several Surfaces
- *Config*: Allows to select a specific state of the system defined by the Multiconfig Lookuptable. If "Current" is selected the plot will show the result for the config that is currently active.

- *Resolution*: Rasterization of the Surface
- *Direction*: Plot the gradient in x-direction "dx" in y-direction "dy" or the absolute gradient "|| ||"

17.13 Reports

17.13.1 Raytrace Report

The *Raytrace Report* allows to trace single rays defined by their field and pupil position and access any ray data as ray-directions, ray-intersection-points, the angle to the surface, surface normal at the point of intersection, optical path length and intensity of the ray (flux). If polarization raytracing is enabled for the sequence following polarization data is also accessible: Electric field components, orientation and ellipticity of the polarization ellipse, the components of the Stokes vector as well as the degree of polarization.



	Ray Dir X	Ray Dir Y	Ray Dir Z	Ray Intersect X	Ray Intersect Y	Ray Intersect Z
Surf 0 IN	0	0	1	4.90521	2.43383	0
Surf 0 OUT	0.173007	0.0858415	0.981173	4.90521	2.43383	0
Surf 1 IN	0.173007	0.0858415	0.981173	6.98014	3.46335	1.7675
Surf 1 OUT	-0.0567441	-0.0281549	0.997992	6.98014	3.46335	1.7675
Surf 2 IN	-0.0567441	-0.0281549	0.997992	5.65405	2.80539	0.0900991
Surf 2 OUT	-0.0761708	-0.0377939	0.996378	5.65405	2.80539	0.0900991
Surf 3 IN	-0.0761708	-0.0377939	0.996378	3.36996	1.67208	-0.0320031
Surf 3 OUT	-0.0410405	-0.0203632	0.99895	3.36996	1.67208	-0.0320031
Surf 4 IN	-0.0410405	-0.0203632	0.99895	2.34964	1.16583	-0.196739
Surf 4 OUT	-0.152362	-0.0755977	0.985429	2.34964	1.16583	-0.196739
Surf 5 IN	-0.152362	-0.0755977	0.985429	4.44089e-16	-8.88178e-16	0

Figure 17.57: Example of Raytrace Report

The plot has the following parameters:

- *Sequence*: Sequence that defines the path of the light (order of Surfaces), Fields and Wavelengths for the raytrace.
- *Wavelength*: Allows to select the Wavelength.
- *Config*: Allows to select a specific state of the system defined by the Multiconfig Lookuptable. If "Current" is selected the plot will show the result for the config that is currently active.
- *Field X*: The absolute field x-coordinate of the ray in field units
- *Field Y*: The absolute field y-coordinate of the ray in field units
- *Normed Pupil X*: The normed pupil x-coordinate of the ray
- *Normed Pupil Y*: The normed pupil y-coordinate of the ray
- *Reference:* This parameter is only relevant for the polarization properties. (see 15.1 and 15.4).

17.13.2 System Data Report

The *System Data Report* gives an overview over over the basic system configuration, first order properties, Seidel aberrations as well as basic surface data.

The plot has the following parameters:

- *Sequence*: Sequence that defines the path of the light (order of Surfaces), Fields and Wavelengths for the raytrace.
- *Config*: Allows to select a specific state of the system defined by the Multiconfig Lookuptable. If "Current" is selected the plot will show the result for the config that is currently active.

17.13.3 Dimensional Data Report

The *Dimensional Data Report* shows the global coordinates of any surface and element in the system as well as the overall size of the system.

18 Optimization

Quaoda Optical CAD offers a wide range of optimization features to be used during the optical design process. For the Optimization most parameters of the Optical Elements like the elements position, orientation and thickness, the surface curvature or in general the Shape or the Phase Function of a Surface can be set as Variables. Furthermore, the optical materials of the Lenses can be optimized by either using a Model Material or by replacing the Catalog Materials with other materials (see chapter 18.3)

18.1 Variables

Most parameters can be set to be Variable (=independent variable) during optimization. To set a parameter as Variable the variable-icon next to the parameter has to be clicked. The parameter can be set to fixed again by clicking the icon again. If a parameter is Variable, the variable-icon next to the parameter will turn green and the Variable will show up inside the Merit Functions Variable List in the System Setup Editor. Constraints on the range of Variables are defined in the Variable List. If there is more than one Merit Function defined, the activation state of a Variable can be set independently for each Merit Function. To select the current Merit Function to work on, the Combo-Box above the *Optical Design Editor* is used. Some parameters, like e.g. the radius of a aperture, cannot be used as Variables since they do not have any effect on the ray paths. For such parameters the variable-icon is not available.

18.2 Constraints and Goals

Quadoa Optical CAD differentiates between Goals and Constraints as optimization targets. Goals specify target values for dependent variables and are used to quantify the grade of optimization via the Merit Function. Constraints are applied more strictly to specify a certain allowed range for independent and dependent variables. The distinction between these two targets is vague when it comes to "soft" constraints (see 18.2.4).

For independent variables, constraints are specified in the Variable List of the Merit Function. Constraints and Goals of dependent parameters are added to the Merit Function either globally or to a corresponding ray trace, depending on the type of the dependent variable.

18.2.1 Goals on dependent Variables

All Optimization Goals have the first four parameters in common:

Column	1	2	3	4
Name	Contrib	Value	Target	Weight
ID	contib	v	t	weight
Symbol	C	V	T	W
Editable	no	no	yes	yes

- *Contrib* shows the current contribution of each Optimization Goal to the Merit Function. It is calculated by the current *Value* minus the *Target* value times the *Weight* $C = (V - T) * W$.
- *Value* shows the current value of the Optimization Goal which is the (unweighted) result of the computation the Optimization Goal performs.
- *Target* defines the target value for the Optimization Goal. The optimizer will try to reduce $V - T$.
- *Weight* defines the weight of one Optimization Goal relative to the other ones. The default value is 1.0, meaning that all Optimization Goals are weighted equally.

18.2.2 Constraints on independent Variables

Any parameter that is set as Variable of a Merit Function can be constrained. This is done in the Variable List. To constraint the range of a Variable, the parameters *Lower Limit* as well as *Upper Limit* need to be set. In addition, the *Bound Type* can be selected. See chapter 18.2.4 for the possible options. When performing a Global Optimization, it is recommend to constraint as many Variables as possible, since this will reduce the search space for the optimization and lead to a shorter optimization time.

Note, that for Radius variables the meaning of Lower Limit and Upper Limit differs. Here instead of the Radius, the curvature is used to define the limits $C = \frac{1}{R}$. *The reason is that the Radius will jump form -infinity to +infinity when the surface changes the curvature from concave to convex, which would lead to confusing definitions of constraints.*

18.2.3 Constraints on dependent Variables

Most Optimization Targets can also be set as a Constraint for the optimization.

Column	1	2	3	4	5	6	7
Name	Contrib	Value	Type	Operator	Target	Weight	Tolerance
ID	contib	v	constrainttype	constraintop	t	weight	tolerance
Symbol	C	V_r	-	-	V_t	W	-
Editable	no	no	yes	yes	yes	yes	yes

- *Contrib* shows the current contribution of each Constraint to the Merit Function. It is calculated by the current *Value* minus the *Target* value times the *Weight* $C = (V_r - V_t) * W$ if the constraint is not fulfilled. If the constraint is fulfilled, the *Value* is always 0.0. If the bound *Type* is set to "Hard" this value just serves as an orientation, since for "Hard" constraints Lagrangian multipliers are used to define the constraint. See chapter 18.2.4.
- *Value* shows the current value of the Constraint which is the (unweighted) result of the computation the Constraint performs.
- *Target* defines the target value for the Constraint. The optimizer will try to keep the *Value* greater ($>$), smaller ($<$) or equal ($=$) to this value, depending on the setting of the *Operator* parameter.
- *Weight* defines the weight of one Constraint relative to the other Constraints as well as Optimization Goal in the case *Type* is set to "Soft". For "Hard" constraints this value is not used. See chapter 18.2.4.
- *Tolerance* defines the tolerance of one Constraint in the case "Hard" equality ($==$) constraints are used. This defines the range around the target value in which the constraint is considered to be fulfilled. See chapter 18.2.4.

18.2.4 Bound Types

Quadoa offers two kinds of Bound Types for Constraints, that are "Hard" and "Soft" bounds.

- "Soft" or Weighted Bounds work in the following way: If the constraint is fulfilled, then the contribution of the Constraint is 0.0 and it will not contribute to the Merit Function. If the constraint is not fulfilled, the contribution will be computed as $C = (V_r - V_t) * W$ where V_r is the value of the constraint, V_t is the *Target* which is the border of the bound and W is the weight relative to other Constraints and Optimization Goals. This means that if

the constraint is not fulfilled, the Constraints behaves as a regular Optimization Goal and if fulfilled it is ignored.

Note: A soft constraint on a dependent parameter with the operator "==" is identical to a Optimization Goal. This setting is only available for convenience.

- "Hard" or Lagrangian Multiplier Bounds do not contribute directly to the Merit Function. They force the optimizer - in addition to improving the merit function - to stay within the specified bounds. Optimization may be started outside of hard bounds, but this should be avoided whenever possible.

18.3 Substitute Material

Any Catalog Material that is used for an element in the *Optical Design Editor*, can be marked as Substitute. This means that the material may be replaced with some other material from within the substitute candidates specified in the Substitute List. To mark a Material as substitute candidate, the *Substitute* parameter has to be checked in the *Optical Design Editor*.

Note, that the attempt to substitute a Material will only be done when using the Extended or Global Optimizer. The Local Optimizer will strictly search for the local minimum of the Merit Function and will not perform any 'Jumps' as they would occur when a Material is switched. However, to optimize Materials within the Local Optimizer a Model Material with variable index of refraction and/or dispersion can be used.

18.4 Merit Function

The Merit Function is used to define all parameters that are needed for the Optimization. It is possible to have an arbitrary number of Merit Functions defined.

Column	1	2	3
Name	Value	Comment	Auto Apply
ID	v	comment	auto
Editable	no	yes	yes

- *Value* displays the current value of the Merit Function and is not editable. It is calculated from the (weighted) contributions of all corresponding global and raytrace optimization goals.

$$Value = \sum_{i=0}^{n} C_i = \sum_{i=0}^{n} (W_i * (V_i - T_i)^2) \tag{18.1}$$

- *Comment* is a free comment field to make it easier to find and identify the individual Merit Function when working with many different ones.
- *Auto Apply*: This parameter will - when activated - automatically trigger a local optimization with a timeout of 1.0 second, whenever the system gets edited by the user. This can be used to e.g. automatically adjust the position of a Surface to always lie in focus, or in general any automatic alignment of the system. Typically this should only be used for simple Merit Functions that can be computed fast, since for complex functions it may lead to a noticeable delay after each editing of the optical system.

The Merit Function is composed of the following child elements:

18.4.1 Variable List

The Variable List contains all parameters, that have been set as Variables for the corresponding Merit Function. Each Variable inside the list has the following parameters:

Column	0	1	2	3	4	5	6	7
Name	Name	Value	Location	Lower Limit	Upper Limit	Bound Type	Weight	Tolerance
Editable	no	yes	no	yes	yes	yes	yes	yes

- *Name* displays the Name of the parameter, which is the same as the parameters name inside the element it belongs to.
- *Value* is the current value of the parameter.
- *Location* displays where exactly the variable is located. E.g. "Lens 2 - Surface 1 - Form Zernike - Z3"
- *Lower Limit* defines the lower limit of the value range the variable is allowed to be in. See chapter 18.2.2.
- *Upper Limit* defines the upper limit of the value range the variable is allowed to be in. See chapter 18.2.2.
- *Bound Type* defines, how the bounds are applied inside the optimization. See chapter 18.2.4

- *Weight* is used for "Soft" bound variable constraints. See chapter 18.2.4
- *Tolerance* is used for "Hard" bound variable constraints. See chapter 18.2.4

18.4.2  Ray Trace

Typically, the optimization relies on some ray tracing results, for example when optimizing. the spot sizes, aberrations or any other ray related properties of the optical system. The Ray Trace item in the Merit Function is used to define this ray trace. In many cases only one Ray Trace item is needed per Merit Function, however it is possible to define an arbitrary amount of Ray Traces for each function. The Ray Trace object has the following parameters:

Column	1	2	3	4
Name	Sequence	Field	Wavelength	Config
ID	seq	field	wave	conf
Editable	yes	yes	yes	yes

- *Sequence* defines the Sequential Ray Trace definition to work with. All settings specified in the Sequence like the order of Surfaces, "Surface Actions", Field Values and Wavelengths will be copied 1:1 from the Sequence. The only exception is the Ray Distribution, that can be set individually for each Ray Trace in the Merit Function. This is done via the Distribution child object of each Ray Trace.
- *Field* is used to either select "All" Fields or just a certain Field of the Sequence to be traced in the ray trace.
- *Wavelength* is used to either select "All" Wavelengths or just a certain Wavelength of the Sequence to be traced in the ray trace.
- *Config* specifies which Configuration a system is to be set in the case, that there are more than one configuration specified via the Multiconfiguration Lookup Table.

Any Optimization Goal or Constraint on ray properties of the optical system will use the Ray Trace object it is assigned to, to evaluate its value.
This architecture is used, since it avoids unnecessary repeating of ray tracing computations - e.g. when the spot radius as well as a Zernike coefficient of the wavefront are to be optimized - since all Optimization Goals use the same ray trace data as input. If for some reason the ray trace data for two Optimization Goal should differ - e.g. be specified on different grids etc. -, this can be achieved by adding a second Ray Trace object and assigning the second Optimization Goal to it.

Note, that if optimizing e.g. zoom systems on multiple Configurations at a time, it is necessary to add multiple Ray Trace objects to the Merit Function.

18.4.2.1 Running Optimization with vignetted Rays

When the *Optimization Dialog* is opened the optimizer is initialized. During the initialization the optimizer will trace a first set of rays for any of the Raytraces defined in the Merit Function. If any of the rays are vignetted or do not reach the image surface of the Sequence for another reason - e.g. exceeding the maximum acceptance angle for transmission (total internal reflection) or lying outside of the definition range of a surface - a warning will appear about rays being vignetted. In this case the optimization can still be run, however any of the vignetted rays will be sorted out and do not contribute to the optimization. When the optimization is started a second time the optimizer is initialized again, and in many cases - since the system has changed - the number of vignetted rays might have been reduced. This sometimes leads to the effect that, for example, the spot size appears to be increased after optimization although the merit function is set up to reduce it. The reason is, that the size - regarding only the rays that where not vignetted in the first run - has been decreases, however due to the additional rays it is larger.

18.4.2.2 Ray Trace Goals and Constraints

Aberrations

- Spot Radius RMS

  Root mean square (RMS) distance of all ray positions from their centroid on the image surface. The centroid is calculated separately for each field. The rays z-coordinate is ignored in case of a curved image surface. *Note: This target cannot be used as a constraint.*

  The Optimization Goal has the following additional parameters:

Column	5	6	8	9
Name	Wavelength	Field	Ignore Lateral Chromatic Aberration	Target ID
ID	wave	field	ign_col	t_uid
Editable	yes	yes	yes	yes

  - *Wavelength* calculate spot radius for a certain wavelength or use all wavelengths that are defined in the Raytrace
  - *Field* calculate spot radius for a certain field or use all wavelengths that are defined in the Raytrace

– *Ignore Lateral Chromatic Aberration* subtracts the centroid for each Wavelength seperately.

Note: For Sequences with Image Space Type set to "Afocal" the Spot Radius Optimization Goal still calculates the spot radius in the image plane, not the collimation. To optimize for collimation the Collimation Optimization Goal has to be used.

- Spot Radius PV

Maximum distance of all ray positions from their centroid on the image surface. The centroid is calculated separately for each Field. The rays z-coordinate is ignored in case of a curved image surface. *Note: This target cannot be used as a constraint.*

The Optimization Goal has the following additional parameters:

Column	5	6	8	9
Name	Wavelength	Field	Ignore Lateral Chromatic Aberration	Target ID
ID	wave	field	ign_col	t_uid
Editable	yes	yes	yes	yes

– *Wavelength* calculate spot radius for a certain wavelength or use all wavelengths that are defined in the Raytrace

– *Field* calculate spot radius for a certain field or use all wavelengths that are defined in the Raytrace

– *Ignore Lateral Chromatic Aberration* subtracts the centroid for each Wavelength separately.

Note: For Sequences with Image Space Type set to "Afocal" the Spot Radius Optimization Goal still calculates the spot radius in the image plane, not the collimation. To optimize for collimation the Collimation Optimization Goal has to be used.

- Spot Size 1D RMS

Root mean square (RMS) distance of all rays single coordinates from their centroid on the image surface. The centroid is calculated separately for each Field. *Note: This target cannot be used as a constraint.*

The Optimization Goal has the following additional parameters:

Column	5	6	8	9	10
Name	Wavelength	Field	Direction	Ignore Lateral Chromatic Aberration	Target ID
ID	wave	field	dir	ign_col	t_uid
Editable	yes	yes	yes	yes	yes

  - *Wavelength* calculate spot radius for a certain wavelength or use all wavelengths that are defined in the Raytrace
  - *Field* calculate spot radius for a certain field or use all wavelengths that are defined in the Raytrace
  - *Direction* specifies the coordinate (X or Y).
  - *Ignore Lateral Chromatic Aberration* calculates the centroid for each Wavelength separately.

Note: For Sequences with Image Space Type set to "Afocal" the Spot Radius Optimization Goal still calculates the spot radius in the image plane, not the collimation. To optimize for collimation the Collimation Optimization Goal has to be used.

- Wavefront RMS

  Root mean square (RMS) of all rays optical path lengths (OPL) at the OPD Reference surface after subtracting their respective mean for each Field/Wavelength. *Note: This target cannot be used as a constraint.*

  The Optimization Goal has the following additional parameters:

Column	5	6	8	9
Name	Wavelength	Field	Substract Modes	Target ID
ID	wave	field	coo	t_uid
Editable	yes	yes	yes	yes

  - *Wavelength* calculate wavefront error for a certain wavelength or use all wavelengths that are defined in the Raytrace
  - *Field* calculate spot radius for a certain field or use all wavelengths that are defined in the Raytrace
  - *Subtract Modes* optionally subtracts tilt and/or defocus from the wavefronts before calculating the value.

- Wavefront PV

Peak-to-Valley (PV) of all rays optical path lengths (OPL) at the OPD Reference surface. *Note: This target cannot be used as a constraint.*

The Optimization Goal has the following additional parameters:

Column	5	6	8	9
Name	Wavelength	Field	Substract Modes	Target ID
ID	wave	field	coo	t_uid
Editable	yes	yes	yes	yes

- *Wavelength* calculate wavefront error for a certain wavelength or use all wavelengths that are defined in the Raytrace
- *Field* calculate spot radius for a certain field or use all wavelengths that are defined in the Raytrace
- *Subtract Modes* optionally subtracts tilt and/or defocus from the wavefronts before calculating the value.

-  Collimation RMS

Collimation of all rays at the image space surface. The collimation is defined as the angular deviation of the rays relative to the chief ray. *Note: This target cannot be used as a constraint.*

The Optimization Goal has the following additional parameters:

Column	5	6	8	9
Name	Wavelength	Field	Ignore Lateral Chromatic Aberration	Target ID
ID	wave	field	ign_col	t_uid
Editable	yes	yes	yes	yes

- *Wavelength* calculate wavefront error for a certain wavelength or use all wavelengths that are defined in the Raytrace
- *Field* calculate spot radius for a certain field or use all wavelengths that are defined in the Raytrace
- *Ignore Angular Chromatic Aberration* optionally ignore angular chromatic aberration. If checked the collimation is calculated separately for each Wavelength.

-  Lateral Chromatic Aberration

Calculates the maximum lateral distance of spots of all wavelengths in the Ray Trace on the image surface.

The Optimization Goal has the following additional parameters:

Column	5	6	7	8
Name	Field	Reference	Type	Target ID
ID	field	ref	focal_type	t_uid
Editable	yes	yes	yes	yes

  - *Field* selects a field number if "All" is selected in the Ray Trace field selector item.
  - *Reference* defines if the chief ray or the centroid of all rays is used to calculated the distance.
  - *Type* defines if the angle or position coordinates are used.

-  Longitudinal Chromatic Aberration

  Calculates the maximum z-distance of spots of all wavelengths in the Ray Trace on the image surface. This target always uses the on-axis field coordinates 0/0.

  The Optimization Goal has the following additional parameters:

Column	5
Name	Target ID
ID	t_uid
Editable	yes

-  Field Curvature

  Calculates the maximum z-distance of spots of all fields in the Ray Trace for the selected wavelength on the image surface.

  The Optimization Goal has the following additional parameters:

Column	5	6
Name	Wavelength	Target ID
ID	wave	t_uid
Editable	yes	yes

  - *Wavelength* selects a wavelength number if "All" is selected in the Ray Trace wavelength selector item.

-  Distortion

The distortion is computed by comparing the position/angle of the chief ray for the fields defined in the Ray Trace with the ideal position/angel that is computed by extrapolating the position/angle field of a small value ϵ near the optical axis.

The Optimization Goal has the following additional parameters:

Column	5	6	7	8	8
Name	Wavelength	Type	Range	Axis	Target ID
ID	wave	distort_type	range_type	axis	t_uid
Editable	yes	yes	yes	yes	yes

- *Wavelength* selects a wavelength number if "All" is selected in the Ray Trace wavelength selector item.
- *Type* Select the type for the calculation of the distortion. Available options are "F Theta" and "F tan(Theta)". In the case of "F Theta" a distortion free system is assumed to be a system, where the image height h_i increases linearly in relation to the object Field angle Θ. In case of "F tan(Theta)" a distortion free system is assumed to be a system, where the image height h_i increases linearly in relation to the tangent of the object Field angle $tan(\Theta)$ which is equal to a linear magnification over all field heights for a finite conjugate system.
- *Range* defines how to compute the distortion value. Options are "Max" (returns the maximum (positive) distortion that occurs for any field), "Min" (returns the minimum (negative) distortion that occurs for any field), "ABS" (returns the maximum absolute distortion that occurs for any field) or "ABS" (returns the maximum minus the minimum distortion that occurs for any field).
- *Axis* for rotationally symmetric systems "Radial" or "Y" is the correct setting. As long as only fields along the Y-axis are defined both will give the same distortion. For non-rotationally symmetric systems where a equal distortion along the X- and Y-axis is required "Radial" is the correct setting. For anamorphotic systems typically two Distortion Optimization Targets need to be defined one for the X- and one for the Y-axis

- Zernike

Performs a Zernike poloynomial fit on the selected wavefront at the OPD Reference surface and sets a single coefficient as the targets value.

The Optimization Goal has the following additional parameters:

Column	5	6	7	8	9	10	11	12
Name	Wavelength	Field	Coefficient	Fit Order	Fit Rad Mode	Fit Radius	Sort Order	Target ID
ID	wave	field	coeff	n_coeff	rad_mod	rad	sort	t_uid
Editable	yes	yes	yes	yes	yes	yes	yes	yes

  - *Wavelength* selects a wavelength number if "All" is selected in the Ray Trace wavelength selector item.
  - *Field* selects a field number if "All" is selected in the Ray Trace field selector item.
  - *Coefficient* defines the the zernike coefficient to be evaluated as the targets value (see 26.1).
  - *Fit Order* defines the number of coefficients to be used in the polynomial fit. Must be larger or equal to "Fit Order".
  - *Fit Rad Mode* defines how the polynomial fit radius is determined. "Auto" uses the cirumference of all available rays. "User Def" allows for manual entry. "From Aperture" uses the Reference Surface aperture.
  - *Fit Radius* defines the polynomial fit radius if "Fit Rad Mode" is set to "User Def".
  - *Sort Order* defines the mapping of coefficient indices to polynomial terms (see 26.1).

-  Tilt

Performs a Zernike poloynomial fit on the selected wavefront at the OPD Reference surface and sets a single tilt coefficient as the targets value.

The Optimization Goal has the following additional parameters:

Column	5	6	7	8	9	10
Name	Wavelength	Field	Direction	Fit Rad Mode	Fit Radius	Target ID
ID	wave	field	dir	rad_mod	rad	t_uid
Editable	yes	yes	yes	yes	yes	yes

  - *Wavelength* selects a wavelength number if "All" is selected in the Ray Trace wavelength selector item.
  - *Field* selects a field number if "All" is selected in the Ray Trace field selector item.
  - *Direction* defines whether the tilt coefficient in X- or Y-direction is calculated.
  - *Fit Rad Mode* defines how the polynomial fit radius is determined. "Auto" uses the cirumference of all available rays. "User Def" allows for manual entry. "From Aperture" uses the Reference Surface aperture.

  - *Fit Radius* defines the polynomial fit radius if "Fit Rad Mode" is set to "User Def".

-  Defocus

  Performs a Zernike poloynomial fit on the selected wavefront at the OPD Reference surface and sets the defocus coefficient as the targets value.

  The Optimization Goal has the following additional parameters:

Column	5	6	7	8	9
Name	Wavelength	Field	Fit Rad Mode	Fit Radius	Target ID
ID	wave	field	rad_mode	rad	t_uid
Editable	yes	yes	yes	yes	yes

  - *Wavelength* selects a wavelength number if "All" is selected in the Ray Trace wavelength selector item.
  - *Field* selects a field number if "All" is selected in the Ray Trace field selector item.
  - *Fit Rad Mode* defines how the polynomial fit radius is determined. "Auto" uses the cirumference of all available rays. "User Def" allows for manual entry. "From Aperture" uses the Reference Surface aperture.
  - *Fit Radius* defines the polynomial fit radius if "Fit Rad Mode" is set to "User Def".

-  Astigmatism

  Performs a Zernike poloynomial fit on the selected wavefront at the OPD Reference surface and sets a single astigmatism coefficient as the targets value.

  The Optimization Goal has the following additional parameters:

Column	5	6	7	8	9	10
Name	Wavelength	Field	Direction	Fit Rad Mode	Fit Radius	Target ID
ID	wave	field	dir	rad_mod	rad	t_uid
Editable	yes	yes	yes	yes	yes	yes

  - *Wavelength* selects a wavelength number if "All" is selected in the Ray Trace wavelength selector item.
  - *Field* selects a field number if "All" is selected in the Ray Trace field selector item.
  - *Direction* defines whether the 0° or 45° astigmatism coefficient is calculated.

- *Fit Rad Mode* defines how the polynomial fit radius is determined. "Auto" uses the circumference of all available rays. "User Def" allows for manual entry. "From Aperture" uses the Reference Surface aperture.
- *Fit Radius* defines the polynomial fit radius if "Fit Rad Mode" is set to "User Def".

-  Coma

  Performs a Zernike poloynomial fit on the selected wavefront at the OPD Reference surface and sets a single first-order coma coefficient as the targets value.

  The Optimization Goal has the following additional parameters:

Column	5	6	7	8	9	10
Name	Wavelength	Field	Direction	Fit Rad Mode	Fit Radius	Target ID
ID	wave	field	dir	rad_mod	rad	t_uid
Editable	yes	yes	yes	yes	yes	yes

  - *Wavelength* selects a wavelength number if "All" is selected in the Ray Trace wavelength selector item.
  - *Field* selects a field number if "All" is selected in the Ray Trace field selector item.
  - *Direction* defines whether the coma coefficient in X- or Y-direction is calculated.
  - *Fit Rad Mode* defines how the polynomial fit radius is determined. "Auto" uses the cirumference of all available rays. "User Def" allows for manual entry. "From Aperture" uses the Reference Surface aperture.
  - *Fit Radius* defines the polynomial fit radius if "Fit Rad Mode" is set to "User Def".

-  Spherical Aberration

  Performs a Zernike poloynomial fit on the selected wavefront at the OPD Reference surface and sets the first order spherical aberration coefficient as the targets value.

  The Optimization Goal has the following additional parameters:

Column	5	6	7	8	9
Name	Wavelength	Field	Fit Rad Mode	Fit Radius	Target ID
ID	wave	field	rad_mode	rad	t_uid
Editable	yes	yes	yes	yes	yes

  - *Wavelength* selects a wavelength number if "All" is selected in the Ray Trace wavelength selector item.

  - *Field* selects a field number if "All" is selected in the Ray Trace field selector item.
  - *Fit Rad Mode* defines how the polynomial fit radius is determined. "Auto" uses the cirumference of all available rays. "User Def" allows for manual entry. "From Aperture" uses the Reference Surface aperture.
  - *Fit Radius* defines the polynomial fit radius if "Fit Rad Mode" is set to "User Def".

-  FFT MTF Calculates the value of the MTF computed via the FFT PSF at a given spatial frequency. The parameters are the same as for the *FFT MTF Plot*

Column	5	6	7	8	9	10	11
Name	Wavelength	Field	Frequency	Pupil Sampling	Image Sampling	Direction	Target ID
ID	wave	field	freq	pupil_samp	imgl_samp	dir	t_uid
Editable	yes	yes	yes	yes	yes	yes	yes

  - *Wavelength* selects a wavelength number if "All" is selected in the Ray Trace wavelength selector item.
  - *Field* selects a Field number if "All" is selected in the Ray Trace field selector item.
  - *Frequency* spatial frequency at which to evaluate the MTF.
  - *Pupil Sampling*: Sampling of rays in the pupil plane. The parameter defined the number of rays for the grid in one direction so the total amount of rays to be traced is the square of the number entered.
  - *Image Sampling*: Sampling of pixels in the image plane. The parameter defined the number of pixels for the grid in one direction so the total amount of pixels is the square of the number entered.

-  Huygens MTF Calculates the value of the MTF computed via the Huygens PSF at a given spatial frequency. The parameters are the same as for the *Huygens MTF Plot*

Column	5	6	7	8	9
Name	Wavelength	Field	Frequency	Pupil Sampling	Image Sampling
ID	wave	field	freq	pupil_samp	img_samp
Editable	yes	yes	yes	yes	yes

Column	10	11	12
Name	Direction	Image Scale	Target ID
ID	dir	img_scale	t_uid
Editable	yes	yes	yes

- *Wavelength* selects a wavelength number if "All" is selected in the Ray Trace wavelength selector item.
- *Field* selects a Field number if "All" is selected in the Ray Trace field selector item.
- *Frequency* spatial frequency at which to evaluate the MTF.
- *Pupil Sampling*: Sampling of rays in the pupil plane. The parameter defined the number of rays for the grid in one direction so the total amount of rays to be traced is the square of the number entered.
- *Image Sampling*: Sampling of pixels in the image plane. The parameter defined the number of pixels for the grid in one direction so the total amount of pixels is the square of the number entered.

-  Geometric MTF Calculates the value of the MTF computed via the geometric PSF at a given spatial frequency. The parameters are the same as for the *Geometric MTF Plot*

Column	5	6	7	8	9	10	11
Name	Wavelength	Field	Frequency	Pupil Sampling	Image Sampling	Direction	Target ID
ID	wave	field	freq	pupil_samp	imgl_samp	dir	t_uid
Editable	yes	yes	yes	yes	yes	yes	yes

- *Wavelength* selects a wavelength number if "All" is selected in the Ray Trace wavelength selector item.
- *Field* selects a Field number if "All" is selected in the Ray Trace field selector item.
- *Frequency* spatial frequency at which to evaluate the MTF.
- *Pupil Sampling*: Sampling of rays in the pupil plane. The parameter defined the number of rays for the grid in one direction so the total amount of rays to be traced is the square of the number entered.
- *Image Sampling*: Sampling of pixels in the image plane. The parameter defined the number of pixels for the grid in one direction so the total amount of pixels is the square of the number entered.
- *Image Scale [µm]*: This parameter defines the total half-size of the area on the image surface for the calculation of the PSF to be included. If set to 0.0 the size will be estimated from the size of the Airy-disk. For aberrated systems this may lead to a cropping of the PSF. Therefore it is recommended to check the PSF before relying on the MTF evaluation.

-  Strehl Ratio Computing the real Strehl ration requires the computation intense computation of the PSF and is therefore often too expensive for optimization. Therefore the Strehl ratio goal is computed from a approximate formula, that is fast and still sufficiently accurate in most cases, especially when the Strehl ration is close to 1.0. (See also Strehl Ratio Huygens)

  The Optimization Goal has the following additional parameters:

Column	5	6	7
Name	Field	Wavelength	Target ID
ID	field	wave	t_uid
Editable	yes	yes	yes

  - *Wavelength* selects a wavelength number if "All" is selected in the Ray Trace wavelength selector item.
  - *Field* selects a field number if "All" is selected in the Ray Trace field selector item.

-  Strehl Ratio Huygens The Strehl Ratio Huygens calculates the strehl ratio by computing a Huygens PSF. This is more accurate than the Strehl Ratio but requires a costly computation.

  The Optimization Goal has the following additional parameters:

Column	5	6	7	8	9	10
Name	Field	Wavelength	Pupil Sampling	Image Sampling	Image Scale / um	Target ID
ID	field	wave	pupil_samp	img_samp	img_scale	t_uid
Editable	yes	yes	yes	yes	yes	yes

  - *Wavelength* selects a wavelength number if "All" is selected in the Ray Trace wavelength selector item.
  - *Field* selects a field number if "All" is selected in the Ray Trace field selector item.
  - *Pupil Sampling* Sampling of rays in the pupil plane
  - *Image Sampling* Sampling of pixels in the image plane
  - *Image Scale / um* Size of the PSF in the image plane to be evaluated

-  Fiber Coupling Efficiency Multi Mode Calculates the fiber coupling efficiency for a multi mode fiber. See the *Multi mode fiber coupling report*.

Column	5	6	7	8	9
Name	Wavelength	Field	Definition by	Core Diameter / um	Core Ref. Index
ID	wave	field	fibr_spec	core_dia	core_ind
Editable	yes	yes	yes	yes	yes

Column	10	11	12	13
Name	Cladding Ref. Index / NA	Incl. Fresnel	Incl. in System	Target ID
ID	cladd_ind_or_NA	fresnel_loss	sys_loss	t_uid
Editable	yes	yes	yes	yes

  - *Wavelength* selects a wavelength number if "All" is selected in the Ray Trace wavelength selector item.
  - *Field* selects a Field number if "All" is selected in the Ray Trace field selector item.
  - *Pupil Sampling*: Sampling of rays in the pupil plane. The parameter defined the number of rays for the grid in one direction so the total amount of rays to be traced is the square of the number entered. The value is used for the Huygens PSF as directly for the pupil functions complex amplitude evaluation.
  - *Image Sampling*: Sampling of pixels in the image plane (on the fiber end). The parameter defined the number of pixels for the grid in one direction so the total amount of pixels is the square of the number entered. This value is only relevant for the calculation of the Huygens PSF when the *Overlap Integral* is set to "On Fiber End".
  - *Definition By*: "By NA" allows to specify the numerical aperture of the fiber where the index of the cladding material will be computed from this value. "By Index" allows to specify the refractive index of the core and the cladding, and the numerical aperture will be computed from these values.
  - *NA*: Maximum numerical aperture the fiber accepts
  - *Core Diameter / um:* Diameter of the fiber core in μm
  - *Core Ref. Index:* Refractive index of the fiber core at the wavelength to be coupled.
  - *Cladding Ref. Index:* Refractive index of the fiber cladding at the wavelength to be coupled.
  - *Incl. Fresnel Loss*: If checked Fresnel losses at the fiber end surface will we included in the calculation.
  - *Incl. in System Loss*: If checked losses due to Fresnel losses or bulk absorption that emerge from inside the optical system will we included in the calculation.

-  Fiber Coupling Efficiency Single Mode Calculates the fiber coupling efficiency by computing the overlap integral in between the fiber-mode and the complex amplitude of the wavefront or PSF in image space. See the *Single mode fiber coupling report* and chapter 27.

Column	5	6	7	8
Name	Wavelength	Field	Pupil Sampling	Image Sampling
ID	wave	field	pupil_samp	img_samp
Editable	yes	yes	yes	yes

Column	9	10	11	12	13
Name	Overlap Integral	Definition by	NA / MFD	Incl. in System Losses	Target ID
ID	overlap_integral	fibr_spec_sm	mfd_or_na	sys_loss	t_uid
Editable	yes	yes	yes	yes	yes

  - *Wavelength* selects a wavelength number if "All" is selected in the Ray Trace wavelength selector item.
  - *Field* selects a Field number if "All" is selected in the Ray Trace field selector item.
  - *Pupil Sampling*: Sampling of rays in the pupil plane. The parameter defined the number of rays for the grid in one direction so the total amount of rays to be traced is the square of the number entered. The value is used for the Huygens PSF as directly for the pupil functions complex amplitude evaluation.
  - *Image Sampling*: Sampling of pixels in the image plane (on the fiber end). The parameter defined the number of pixels for the grid in one direction so the total amount of pixels is the square of the number entered. This value is only relevant for the calculation of the Huygens PSF when the *Overlap Integral* is set to "On Fiber End".
  - *Definition By*: "By NA" allows to specify the $\frac{1}{e^2}$ numerical aperture of the fiber where the index of the cladding material will be computed from this value. "By MFD" allows to specify the $\frac{1}{e^2}$ mode field diameter of the fiber.
  - *Fiber NA(1/e2)*: Numerical aperture $\frac{1}{e^2}$ of the fiber
  - *Fiber MFD(1/e2) / um:* $\frac{1}{e^2}$ mode field diameter of the fiber
  - *Incl. in System Loss*: If checked losses due to Fresnel losses or bulk absorption that emerge from inside the optical system will we included in the

-  Fiber Coupling Efficiency Beamlet Propagation Calculates the fiber coupling efficiency by computing the overlap integral in between the fiber-mode and the PSF in image space. See the *Beamlet propagation fiber coupling report* and chapter 27.

Column	5	6	7	8
Name	Wavelength	Field	Pupil Sampling	Image Sampling
ID	wave	field	pupil_samp	img_samp
Editable	yes	yes	yes	yes

Column	9	10	11	12	13
Name	Oversampling	Definition by	NA / MFD	Incl. in System Losses	Target ID
ID	oversample	fibr_spec_sm	mfd_or_na	sys_loss	t_uid
Editable	yes	yes	yes	yes	yes

  - *Wavelength* selects a wavelength number if "All" is selected in the Ray Trace wavelength selector item.
  - *Field* selects a Field number if "All" is selected in the Ray Trace field selector item.
  - *Pupil Sampling*: Sampling of beamlets in the pupil plane. The parameter defined the number of beamlets for the grid in one direction so the total amount of beamlets to be traced is the square of the number entered.
  - *Image Sampling*: Sampling of pixels in the image plane (on the fiber end). The parameter defined the number of pixels for the grid in one direction so the total amount of pixels is the square of the number entered.
  - *Oversampling*: The overlap of beamlets to their neighbor beamlets. A large overlap leads to a smoother overall wavefront, however steep edges of the intensity profile (as e.g. in a top-hat profile) will also be smoothed out. A good compromise is typically between 2 and 4.
  - *Definition By*: "By NA" allows to specify the $\frac{1}{e^2}$ numerical aperture of the fiber where the index of the cladding material will be computed from this value. "By MFD" allows to specify the $\frac{1}{e^2}$ mode field diameter of the fiber.
  - *Fiber NA(1/e2)*: Numerical aperture $\frac{1}{e^2}$ of the fiber
  - *Fiber MFD(1/e2) / um:* $\frac{1}{e^2}$ mode field diameter of the fiber
  - *Incl. in System Loss*: If checked losses due to Fresnel losses or bulk absorption that emerge from inside the optical system will we included in the calculation.

- Gaussian Beam Waist Size Calculates the size of the beam waist at a given surface. The *Source Type* of the Sequence needs to be set to "Gaussian Beam" in order to use this optimization goal.

  The Optimization Goal has the following additional parameters:

Column	5	6	7	8	9
Name	Field	Wavelength	Surface	Direction	Target ID
ID	field	wave	surf	dir	t_uid
Editable	yes	yes	yes	yes	yes

  - *Wavelength* selects a wavelength number if "All" is selected in the Ray Trace wavelength selector item.
  - *Field* selects a field number if "All" is selected in the Ray Trace field selector item.
  - *Surface* Surface at which to evaluate the waist size
  - *Direction* Selection of the axis (X or Y)

-  Gaussian Beam Waist dZ Calculated the waist position along the z-axis relative to the selected surface. The *Source Type* of the Sequence needs to be set to "Gaussian Beam" in order to use this optimization goal.

  The Optimization Goal has the following additional parameters:

Column	5	6	7	8	9
Name	Field	Wavelength	Surface	Direction	Target ID
ID	field	wave	surf	dir	t_uid
Editable	yes	yes	yes	yes	yes

  - *Wavelength* selects a wavelength number if "All" is selected in the Ray Trace wavelength selector item.
  - *Field* selects a field number if "All" is selected in the Ray Trace field selector item.
  - *Surface* Surface at which to evaluate the waist size
  - *Direction* Selection of the axis (X or Y)

-  Gaussian Beam Divergence Calculated the beam divergence at the selected surface. The *Source Type* of the Sequence needs to be set to "Gaussian Beam" in order to use this optimization goal.

  The Optimization Goal has the following additional parameters:

Column	5	6	7	8	9
Name	Field	Wavelength	Surface	Direction	Target ID
ID	field	wave	surf	dir	t_uid
Editable	yes	yes	yes	yes	yes

  - *Wavelength* selects a wavelength number if "All" is selected in the Ray Trace wavelength selector item.
  - *Field* selects a field number if "All" is selected in the Ray Trace field selector item.
  - *Surface* Surface at which to evaluate the waist size
  - *Direction* Selection of the axis (X or Y)

-  Gaussian Beam Rayleigh Length Calculated the Rayleigh length of the Gaussian beam at the selected surface. The *Source Type* of the Sequence needs to be set to "Gaussian Beam" in order to use this optimization goal.

  The Optimization Goal has the following additional parameters:

Column	5	6	7	8	9
Name	Field	Wavelength	Surface	Direction	Target ID
ID	field	wave	surf	dir	t_uid
Editable	yes	yes	yes	yes	yes

  - *Wavelength* selects a wavelength number if "All" is selected in the Ray Trace wavelength selector item.
  - *Field* selects a field number if "All" is selected in the Ray Trace field selector item.
  - *Surface* Surface at which to evaluate the waist size
  - *Direction* Selection of the axis (X or Y)

-  Gaussian Beam WF Curvature Calculated the wave front curvature of the Gaussian Beam at the selected surface. The *Source Type* of the Sequence needs to be set to "Gaussian Beam" in order to use this optimization goal.

  The Optimization Goal has the following additional parameters:

Column	5	6	7	8	9
Name	Field	Wavelength	Surface	Direction	Target ID
ID	field	wave	surf	dir	t_uid
Editable	yes	yes	yes	yes	yes

  - *Wavelength* selects a wavelength number if "All" is selected in the Ray Trace wavelength selector item.
  - *Field* selects a field number if "All" is selected in the Ray Trace field selector item.
  - *Surface* Surface at which to evaluate the waist size

– *Direction* Selection of the axis (X or Y)

Optical Properties

-  Image Space NA

  Calculates the working numerical aperture of the incident rays on the image surface. The maximum angle between the rays and their respective chief ray is used for this. This target always uses the on-axis field coordinates 0/0.

  The Optimization Goal has the following additional parameters:

Column	5	6
Name	Wavelength	Target ID
ID	wave	t_uid
Editable	yes	yes

  – *Wavelength* selects a wavelength number if "All" is selected in the Ray Trace wavelength selector item.

-  Object Space NA

  Calculates the working numerical aperture of the rays exiting the source surface. The maximum angle between the rays and their respective chief ray is used for this. This target always uses the on-axis field coordinates 0/0.

  The Optimization Goal has the following additional parameters:

Column	5	6
Name	Wavelength	Target ID
ID	wave	t_uid
Editable	yes	yes

  – *Wavelength* selects a wavelength number if "All" is selected in the Ray Trace wavelength selector item.

-  Image Space F#

  Calculates the working f/# of the incident rays on the image surface. The maximum angle between the rays and their respective chief ray is used for this. This target always uses the on-axis field coordinates 0/0.

  The Optimization Goal has the following additional parameters:

Column	5	6
Name	Wavelength	Target ID
ID	wave	t_uid
Editable	yes	yes

  - *Wavelength* selects a wavelength number if "All" is selected in the Ray Trace wavelength selector item.

-  Object Space F#

  Calculates the working f/# of the rays exiting the source surface. The maximum angle between the rays and their respective chief ray is used for this. This target always uses the on-axis field coordinates 0/0.

  The Optimization Goal has the following additional parameters:

Column	5	6
Name	Wavelength	Target ID
ID	wave	t_uid
Editable	yes	yes

  - *Wavelength* selects a wavelength number if "All" is selected in the Ray Trace wavelength selector item.

-  Effective F#

  Calculates the working f/# of the incident rays on the image surface for a object at infinity. The maximum angle between the rays and their respective chief ray is used for this. This target always uses the on-axis field coordinates 0/0.

  The Optimization Goal has the following additional parameters:

Column	5	6
Name	Wavelength	Target ID
ID	wave	t_uid
Editable	yes	yes

  - *Wavelength* selects a wavelength number if "All" is selected in the Ray Trace wavelength selector item.

-  Effective Focal Length

  Calculates the effective focal length for the sequence, assuming a object at infinity. This target always uses the on-axis field coordinates 0/0.

The Optimization Goal has the following additional parameters:

Column	5	6
Name	Wavelength	Target ID
ID	wave	t_uid
Editable	yes	yes

  - *Wavelength* selects a wavelength number if "All" is selected in the Ray Trace wavelength selector item.

-  Entrance Pupil Position

  Calculates the entrance pupil position for the sequence. The position is defined as the z-distance from the source surface. This target always uses the on-axis field coordinates 0/0.

  The Optimization Goal has the following additional parameters:

Column	5	6
Name	Wavelength	Target ID
ID	wave	t_uid
Editable	yes	yes

  - *Wavelength* selects a wavelength number if "All" is selected in the Ray Trace wavelength selector item.

-  Entrance Pupil Radius

  Calculates the entrance pupil radius for the sequence. This target always uses the on-axis field coordinates 0/0.

  The Optimization Goal has the following additional parameters:

Column	5	6
Name	Wavelength	Target ID
ID	wave	t_uid
Editable	yes	yes

  - *Wavelength* selects a wavelength number if "All" is selected in the Ray Trace wavelength selector item.

-  Exit Pupil Position

  Calculates the exit pupil position for the sequence. The position is defined as the z-distance from the image surface. This target always uses the on-axis field coordinates 0/0.

The Optimization Goal has the following additional parameters:

Column	5	6
Name	Wavelength	Target ID
ID	wave	t_uid
Editable	yes	yes

  - *Wavelength* selects a wavelength number if "All" is selected in the Ray Trace wavelength selector item.

-  Exit Pupil Radius

  Calculates the exit pupil radius for the sequence. This target always uses the on-axis field coordinates 0/0.

  The Optimization Goal has the following additional parameters:

Column	5	6
Name	Wavelength	Target ID
ID	wave	t_uid
Editable	yes	yes

  - *Wavelength* selects a wavelength number if "All" is selected in the Ray Trace wavelength selector item.

-  Magnification

  Calculates the magnification for the sequence. This target always uses the on-axis field coordinates 0/0.

  The Optimization Goal has the following additional parameters:

Column	5	6	7
Name	Wavelength	Direction	Target ID
ID	wave	dir	t_uid
Editable	yes	yes	yes

  - *Wavelength* selects a wavelength number if "All" is selected in the Ray Trace wavelength selector item.
  - *Direction* Defines if the target is calculated along the "X" or "Y"-axis.

-  Angular Magnification

Calculates the angular magnification for the sequence. This target always uses the on-axis field coordinates 0/0.

The Optimization Goal has the following additional parameters:

Column	5	6	7
Name	Wavelength	Direction	Target ID
ID	wave	dir	t_uid
Editable	yes	yes	yes

  - *Wavelength* selects a wavelength number if "All" is selected in the Ray Trace wavelength selector item.
  - *Direction* Defines if the target is calculated along the "X" or "Y"-axis.

-  Object Space Telecentricity

  Calculates the offense against telecentricity in object space. The value is obtained by calculating the angle under which the chief ray of all fields defined in the Ray Trace is launched at the object surface. This target has no additional parameters.

-  Imgage Space Telecentricity

  Calculates the offense against telecentricity in image space. The value is obtained by calculating the angle under which the chief ray of all fields defined in the Ray Trace hit the image surface. This target has no additional parameters.

Ray Data Target Types

All Ray Data Goals are split into three types:

- Chief Ray Goals

  Calculates the goal from the chief ray corresponding to the selected field/wavelength. All chief ray goals have the following parameters:

Column	5	6	7
Name	Wavelength	Field	Surface
ID	wave	field	surf
Editable	yes	yes	yes

  - *Wavelength* selects a wavelength number if "All" is selected in the Ray Trace wavelength selector item.

- *Field* selects a field number if "All" is selected in the Ray Trace wavelength selector item.
- *Surface* selects a surface from the sequence selected in the Ray Trace item.

- Single Ray Goals

  Calculates the goal from a single ray defined by field and pupil coordinates. All single ray goals have the following parameters:

Column	5	6	7	8	9	10
Name	Wavelength	Surface	Field X	Field Y	Normed Pupil X	Normed Pupil Y
ID	wave	surf	fx	fy	px	py
Editable	yes	yes	yes	yes	yes	yes

  - *Wavelength* selects a wavelength number if "All" is selected in the Ray Trace wavelength selector item.
  - *Field X* defines the absolute field x-coordinate of the ray.
  - *Field Y* defines the absolute field y-coordinate of the ray.
  - *Pupil X* defines the normalized pupil x-coordinate of the ray.
  - *Pupil Y* defines the normalized pupil y-coordinate of the ray.
  - *Surface* selects a surface from the sequence selected in the Ray Trace item.

- Multi Ray Goals

  Calculates the goal from all rays of the selected fields and wavelengths. This is either done on all rays seperately (RMS goals) or by calcting a single goal from all ray (e.g. MAX goals). All multi ray goals have the following parameters:

Column	5	6
Name	Wavelength	Field
ID	wave	field
Editable	yes	yes

  - *Wavelength* selects a wavelength number if "All" is selected in the Ray Trace wavelength selector item.
  - *Field* selects a field number if "All" is selected in the Ray Trace wavelength selector item.
  - *Surface* selects a surface from the sequence selected in the Ray Trace item.

Ray Data Targets

-  Ray Position Calculates the position of the ray in the local surface coordinate system.

  The Optimization Goal has the following additional parameters:

Column	End+1	End+2
Name	Coordinate	Target ID
ID	coo	t_uid
Editable	yes	yes

  - *Coordinate* defines if the target is calculated on the "X" or "Y"-Coordinate.

-  Ray Position Global Calculates the position of the ray in the global coordinate system.

  The Optimization Goal has the following additional parameters:

Column	End+1	End+2
Name	Coordinate	Target ID
ID	coo	t_uid
Editable	yes	yes

  - *Coordinate* Defines if the target is calculated on the "X" or "Y"-Coordinate.

-  Ray Incident Angle

  Calculates the incident ray angle to the surface normal.

  The Optimization Goal has the following additional parameters:

Column	End+1	End+2
Name	Unit	Target ID
ID	unit	t_uid
Editable	yes	yes

  - *Unit* defines if the target is calculated in degree or radians.

-  Ray Exit Angle

  Calculates the exit ray angle to the surface normal.

  The Optimization Goal has the following additional parameters:

Column	End+1	End+2
Name	Unit	Target ID
ID	unit	t_uid
Editable	yes	yes

  - *Unit* defines if the target is calculated in degree or radians.

-  Ray Refraction Angle

  Calculates the angle between the incoming and outgoing ray at a surface.

  The Optimization Goal has the following additional parameters:

Column	End+1	End+2
Name	Unit	Target ID
ID	unit	t_uid
Editable	yes	yes

  - *Unit* defines if the target is calculated in degree or radians.

-  Incoming Ray Angle

  Calculates the angle between the incoming ray and the surface coordinate systems X-Z or Y-Z plane.

  The Optimization Goal has the following additional parameters:

Column	End+1	End+2	End+3
Name	Direction	Unit	Target ID
ID	dir	unit	t_uid
Editable	yes	yes	yes

  - *Direction* defines which direction component to calculate. "X" calculates the angle between the ray and the local X-Z plane for example.
  - *Unit* defines if the target is calculated in degree or radians.

-  Outgoing Ray Angle

  Calculates the angle between the outgoing ray and the surface coordinate systems X-Z or Y-Z plane.

  The Optimization Goal has the following additional parameters:

Column	End+1	End+2	End+3
Name	Direction	Unit	Target ID
ID	dir	unit	t_uid
Editable	yes	yes	yes

  - *Direction* defines which direction component to calculate. "X" calculates the angle between the ray and the local X-Z plane for example.

  - *Unit* defines if the target is calculated in degree or radians.

-  Incoming Ray Angle Global

  Calculates the angle between the incoming ray and the global coordinate systems X-Z or Y-Z plane.

  The Optimization Goal has the following additional parameters:

Column	End+1	End+2	End+3
Name	Direction	Unit	Target ID
ID	dir	unit	t_uid
Editable	yes	yes	yes

  - *Direction* defines which direction component to calculate. "X" calculates the angle between the ray and the local X-Z plane for example.
  - *Unit* defines if the target is calculated in degree or radians.

-  Outgoing Ray Angle Global

  Calculates the angle between the outgoing ray and the global coordinate systems X-Z or Y-Z plane.

  The Optimization Goal has the following additional parameters:

Column	End+1	End+2	End+3
Name	Direction	Unit	Target ID
ID	dir	unit	t_uid
Editable	yes	yes	yes

  - *Direction* defines which direction component to calculate. "X" calculates the angle between the ray and the local X-Z plane for example.
  - *Unit* defines if the target is calculated in degree or radians.

-  OPL

  Calculates the absolute optical path length of a ray.

  The Optimization Goal has the following additional parameters:

Column	End+1
Name	Target ID
ID	t_uid
Editable	yes

-  Orientation

  Calculates the orientation of the rays polarization ellipse at the surface. This is defined as the cosine of the angle between the main axis an the x-axis at the coordinate reference.

  The Optimization Goal has the following additional parameters:

Column	End+1	End+2	End+3
Name	Status	Reference	Target ID
ID	status	ref	t_uid
Editable	yes	yes	yes

  – *Status* defines whether the polarization before or after the surface interaction is evaluated.

  – *Reference* defines the reference coordiante system for polarization evaluation (see 15.4).

-  Ellipticity

  Calculates the Ellipticity or the rays electrical field vector at the surface. Calculates zero for circular polarization and one for linear polarization.

  The Optimization Goal has the following additional parameters:

Column	End+1	End+2	End+3
Name	Status	Reference	Target ID
ID	status	ref	t_uid
Editable	yes	yes	yes

  – *Status* defines whether the polarization before or after the surface interaction is evaluated.

  – *Reference* defines the reference coordiante system for polarization evaluation (see 15.4).

-  Ray Energy

  Calculates the relative flux of the ray at the image surface.

  The Optimization Goal has the following additional parameters:

Column	End+1
Name	Target ID
ID	t_uid
Editable	yes

-  Transmitted Energy

Calculates the sum of the flux of all not vignetted rays at the image surface relative to the flux of all not vignetted rays at the object surface. This target can e.g. be used for the optimization of coatings layer thicknesses or Fresnel losses.

The Optimization Goal has the following additional parameters:

Column	7
Name	Target ID
ID	t_uid
Editable	yes

-  Geometric Enclose Energy

  Calculates the sum of the flux of all rays that lie within a certain area relative to the total flux of all rays that reaches the image surface.

  The Optimization Goal has the following additional parameters:

Column	7	8	9	10	11	12
Name	Size / um	Pupil Sampling	Image Sampling	Enclose Typ	Image Scale / um	Target ID
ID	size	pupil_samp	img_samp	enc_type	img_scale	t_uid
Editable	yes	yes	yes	yes	yes	yes

  - *Size / um* Size of the enclosed area. Depending on the value of *Enclose Type* this refers either to the radius, the half size of the square or the half size of the infinite stripe in the cross-section direction.
  - *Pupil Sampling*: Number of rays in x, and y direction for the rasterization of the pupil plane that is used for PSF computation.
  - *Image Sampling*: Rasterization of the pixels on the image surface
  - *Enclose Type*: "Encircle" plots the encirceld energy over the radius of the circular area. "Ensquared" plots the ensquared energy inside a quadratic area ranging from $\pm r$ in x and y direction. "Range X" plots the enclosed energy inside an area that is infinite in y direction and reaches from $-r$ to $+r$ in x direction. "Range Y" plots the enclosed energy inside an area that is infinite in x direction and reaches from $-r$ to $+r$ in y direction.
  - *Image Scale / um* If set to 0.0 the image scale on which to evaluate the energy will be automatically calculated from the convex hull around the spot. If a value other than 0.0 is specified this value will be used for the radius of the region in image space over which the energy summation is evaluated.

-  FFT Enclose Energy Calculates the enclosed energy by summing up the intensity of the pixels of the PSF calculated by the FFT method.

  The Optimization Goal has the following additional parameters:

Column	7	8	9	10	11
Name	Size / um	Pupil Sampling	Image Sampling	Enclose Typ	Target ID
ID	size	pupil_samp	img_samp	enc_type	t_uid
Editable	yes	yes	yes	yes	yes

  - *Size / um* Size of the enclosed area. Depending on the value of *Enclose Type* this refers either to the radius, the half size of the square or the half size of the infinite stripe in the cross-section direction.
  - *Pupil Sampling*: Sampling of rays in the pupil plane. The parameter defined the number of rays for the grid in one direction so the total amount of rays to be traced is the square of the number entered.
  - *Image Sampling*: Sampling of pixels in the image plane. The parameter defined the number of pixels for the grid in one direction so the total amount of pixels is the square of the number entered.
  - *Enclose Type*: "Encircle" plots the encirceld energy over the radius of the circular area. "Ensquared" plots the ensquared energy inside a quadratic area ranging from $\pm r$ in x and y direction. "Range X" plots the enclosed energy inside an area that is infinite in y direction and reaches from $-r$ to $+r$ in x direction. "Range Y" plots the enclosed energy inside an area that is infinite in x direction and reaches from $-r$ to $+r$ in y direction.

-  Huygens Enclose Energy Calculates the enclosed energy by summing up the intensity of the pixels of the PSF calculated by the Huygens integral method.

  The Optimization Goal has the following additional parameters:

Column	7	8	9	10	11	12
Name	Size / um	Pupil Sampling	Image Sampling	Enclose Typ	Image Scale / um	Target ID
ID	size	pupil_samp	img_samp	enc_type	img_scale	t_uid
Editable	yes	yes	yes	yes	yes	yes

  - *Size / um* Size of the enclosed area. Depending on the value of *Enclose Type* this refers either to the radius, the half size of the square or the half size of the infinite stripe in the cross-section direction.

- *Pupil Sampling*: Sampling of rays in the pupil plane. The parameter defined the number of rays for the grid in one direction so the total amount of rays to be traced is the square of the number entered.
- *Image Sampling*: Sampling of pixels in the image plane. The parameter defined the number of pixels for the grid in one direction so the total amount of pixels is the square of the number entered.
- *Enclose Type*: "Encircle" plots the encirceld energy over the radius of the circular area. "Ensquared" plots the ensquared energy inside a quadratic area ranging from $\pm r$ in x and y direction. "Range X" plots the enclosed energy inside an area that is infinite in y direction and reaches from $-r$ to $+r$ in x direction. "Range Y" plots the enclosed energy inside an area that is infinite in x direction and reaches from $-r$ to $+r$ in y direction.
- *Image Scale [µm]*: This parameter defines the total half-size of the area on the image surface to be plotted in μm. If set to 0.0 the size will be estimated from the size of the Airy-disk.

18.4.3 Paraxial Raytrace

In some cases it it preferred to use paraxial values for the optimization instead of the values obtained from real ray racing. The advantage of the paraxial Optimization Goals or Constraints is, that they typically can be computed faster compared to the real ray tracing ones. However in most cases it does not make a large difference since the rays to optimized e.g. the spot radius are already traced in the Merit Function and therefore adding a Paraxial Raytrace just for one optimization target even may be slower. Another reason for the use of paraxial data might be that the values are specified this way in the data sheet or for comparison with the results from other software packages that rely on paraxial data. The Paraxial Ray Trace object has the following parameters:

Column	1	2	3	4
Name	Sequence	Wavelength	Config	Axis
ID	seq	wave	conf	axis
Editable	yes	yes	yes	yes

- *Sequence* defines the Sequential Ray Trace definition to work with. All settings specified in the Sequence like the order of Surfaces, "Surface Actions", Field Values and Wavelengths will be copied 1:1 from the Sequence. The only exception is the Ray Distribution, that can be set individually for each Ray Trace in the Merit Function. This is done via the Distribution child object of each Ray Trace.

- *Field* is used to either select "All" Fields or just a certain Field of the Sequence to be traced in the ray trace.
- *Wavelength* is used to either select "All" Wavelengths or just a certain Wavelength of the Sequence to be traced in the ray trace.
- *Config* specifies which Configuration a system is to be set in the case, that there are more than one configuration specified via the Multiconfiguration Lookup Table.
- *Axis* specifies whether the paraxial ray trace is along the "X" or "Y" axis.

18.4.3.1 Paraxial Ray Trace Goals and Constraints

Paraxial Targets

- Effective Focal Length

  Calculates the effective focal length for the sequence, assuming a object at infinity.

  The Optimization Goal has the following additional parameters:

Column	5
Name	Target ID
ID	t_uid
Editable	yes

- Magnification

  Calculates the paraxial magnification for the sequence.

  The Optimization Goal has the following additional parameters:

Column	5
Name	Target ID
ID	t_uid
Editable	yes

- Angular Magnification

  Calculates the paraxial angular magnification for the sequence.

  The Optimization Goal has the following additional parameters:

Column	5
Name	Target ID
ID	t_uid
Editable	yes

- Seidel Sum

  Calculates the Seidel aberration for a lens. See discussion on Seidel Aberrations.

  The Optimization Goal has the following additional parameters:

Column	5	6	7
Name	Aberration	Type	Target ID
ID	abb	seidel_type	t_uid
Editable	yes	yes	yes

  – *Aberration* selects the Seidel aberration to be calculated.

  – *Type* selects the Seidel aberration type be calculated. Options are "S" and "W".

- Seidel Surface Contribution

  Calculates the Seidel aberration surface contribution for a lens. See discussion on Seidel Aberrations.

  The Optimization Goal has the following additional parameters:

Column	5	6	7	8
Name	Surface	Aberration	Type	Target ID
ID	surf	abb	seidel_type	t_uid
Editable	yes	yes	yes	yes

  – *Surface* selects a surface number for which to calculate the Seidel surface contribution.

  – *Aberration* selects the Seidel aberration to be calculated.

  – *Type* selects the Seidel aberration type be calculated. Options are "S" and "W".

- Cardinal Plane

  Calculates the position of the systems cardinal planes. The position is defined as the z-distance from the source surface.

  The Optimization Goal has the following additional parameters:

Column	5	6
Name	Plane	Target ID
ID	plane	t_uid
Editable	yes	yes

  - *Plane* Selects which cardinal plane to calculate. Options are "First Focal", "Second Focal", "First Principal" and "Second Principal".
    First Focal and First Principal are the object-side planes, Second Focal and Second Principal are image-side.

-  Image Distance

  Calculates the image position for an object at infinity.

  The Optimization Goal has the following additional parameters:

Column	5
Name	Target ID
ID	t_uid
Editable	yes

-  Entrance Pupil Position

  Calculates the paraxial entrance pupil position for the sequence. The position is defined as the z-distance from the source surface.

  The Optimization Goal has the following additional parameters:

Column	5
Name	Target ID
ID	t_uid
Editable	yes

-  Entrance Pupil Radius

  Calculates the paraxial entrance pupil radius for the sequence.

  The Optimization Goal has the following additional parameters:

Column	5
Name	Target ID
ID	t_uid
Editable	yes

-  Exit Pupil Position

  Calculates the paraxial exit pupil position for the sequence. The position is defined as the z-distance from the image surface.

  The Optimization Goal has the following additional parameters:

Column	5
Name	Target ID
ID	t_uid
Editable	yes

-  Exit Pupil Radius

  Calculates the paraxial exit pupil radius for the sequence.

  The Optimization Goal has the following additional parameters:

Column	5
Name	Target ID
ID	t_uid
Editable	yes

-  Image Space NA

  Calculates the image space numerical aperture based on the paraxial ray trace.

  The Optimization Goal has the following additional parameters:

Column	5
Name	Target ID
ID	t_uid
Editable	yes

-  Object Space NA

  Calculates the object space numerical aperture based on the paraxial ray trace.

  The Optimization Goal has the following additional parameters:

Column	5
Name	Target ID
ID	t_uid
Editable	yes

-  Image Space F# Calculates the paraxial working f/# in image space.

  The Optimization Goal has the following additional parameters:

Column	5
Name	Target ID
ID	t_uid
Editable	yes

-  Object Space F#

  Calculates the paraxial working f/# of the rays in object space.

  The Optimization Goal has the following additional parameters:

Column	5
Name	Target ID
ID	t_uid
Editable	yes

-  Ray Position Calculates the paraxial position of the ray in the local surface coordinate system.

  The Optimization Goal has the following additional parameters:

Column	5	6	7	8
Name	Surface	Field	Normed Pupil	Target ID
ID	surf	f	p	t_uid
Editable	yes	yes	yes	yes

  - *Surface* Surface at which to evaluate the angle
  - *Field* The field value in field units
  - *Normed Pupil* The normed pupil coordinate

-  Incoming Ray Angle

  Calculates the angle between the paraxial incoming ray and the surface coordinate systems X-Z or Y-Z plane.

  The Optimization Goal has the following additional parameters:

Column	5	6	7	8
Name	Surface	Field	Normed Pupil	Target ID
ID	surf	f	p	t_uid
Editable	yes	yes	yes	yes

  - *Surface* Surface at which to evaluate the angle
  - *Field* The field value in field units
  - *Normed Pupil* The normed pupil coordinate

-  Outgoing Ray Angle

  Calculates the angle between the paraxial outgoing ray and the surface coordinate systems X-Z or Y-Z plane.

  The Optimization Goal has the following additional parameters:

Column	5	6	7	8
Name	Surface	Field	Normed Pupil	Target ID
ID	surf	f	p	t_uid
Editable	yes	yes	yes	yes

  - *Surface* Surface at which to evaluate the angle
  - *Field* The field value in field units
  - *Normed Pupil* The normed pupil coordinate

18.4.4  Global Goals and Constraints

All goals and constraints that are not specific to a single Sequence are separated from the ray trace goals and constraints. These are parameters that are related to element geometry and materials. All Optimization Goals have the first six parameters in common.

Column	5	6
Name	Config	Element ID
ID	conf	
Editable	yes	yes

- *Config* specifies which Configuration a system is to be set in the case, that there are more than one configuration specified via the Multiconfiguration Lookup Table.
- *Element ID* defines the element on which the target is calculated

Note: Only rays that are traced within the Merit Function are considered when calculating global goals and constraints. For example the floating aperture of a surface used for calculating the edge thickness will be calculated only from rays used in the optimization.

-  Center Thickness

  The Optimization Goal has the following additional parameters:

Column	7	8
Name	Surface	Target ID
ID	surf	t_uid
Editable	yes	yes

  – *Surface* defines the first surface on which the target is calculated when the lens is a kitted element with more than 2 surfaces. The numbering follows the order in the Optical Design Editor. The second surface is always the following surface.

-  Edge Thickness

  The Optimization Goal has the following additional parameters:

Column	7	8
Name	Surface	Target ID
ID	surf	t_uid
Editable	yes	yes

  – *Element ID* defines the element for which the target is calculated.

  – *Surface* defines the first surface on which the target is calculated when the lens is a kitted element with more than 2 surfaces. The numbering follows the order in the Optical Design Editor. The second surface is always the following surface.

-  Center Air Gap

  The Optimization Goal has the following additional parameters:

Column	7
Name	Target ID
ID	t_uid
Editable	yes

  The air gap is calculated towards the following element in the Optical Design Editor. For User-Sequences where the order of elements in the Sequence differs from the order in the Optical Design Editor the General Distance Target is recommended.

-  Edge Air Gap

  The Optimization Goal has the following additional parameters:

Column	7
Name	Target ID
ID	t_uid
Editable	yes

  The air gap is calculated towards the following element in the Optical Design Editor. For User-Sequences where the order of elements in the Sequence differs from the order in the Optical Design Editor the General Distance Target is recommended.

-  Global Position Calculates the position of a surface vertex inside the global coordinate system. The Optimization Goal has the following additional parameters:

Column	7	8	9
Name	Surface	Coordinate	Target ID
ID	surf	coo	t_uid
Editable	yes	yes	yes

  - *Element ID* defines the element for which the target is calculated.
  - *Surface* defines the first surface on which the target is calculated when the lens is a kitted element with more than 2 surfaces. The numbering follows the order in the Optical Design Editor. The second surface is always the following surface.
  - *Coordinate* defines which coordinate of the position to return. Options are "X", "Y" and "Z".

-  Global Orientation Calculates the x, y or z component of a unit vector along the x, y or z axis of the local surface coordinate system inside the global coordinate system. The Optimization Goal has the following additional parameters:

Column	7	8	9	10
Name	Surface	Base Vector	Component	Target ID
ID	surf	vector	component	t_uid
Editable	yes	yes	yes	yes

  - *Element ID* defines the element for which the target is calculated.
  - *Surface* defines the first surface on which the target is calculated when the lens is a kitted element with more than 2 surfaces. The numbering follows the order in the Optical Design Editor. The second surface is always the following surface.

- *Base Vector* defines which axis of the local surface coordinate system is used. Options are "X", "Y" and "Z".
- *Component* defines which component of the vector inside the global coordinate system to return. Options are "X", "Y" and "Z".

- General Distance 3D

The Optimization Goal has the following additional parameters:

Column	6	7	8	9	10	11	12	13	14
Name	Element ID 1	Surface 1	X 1	Y 1	Element ID 2	Surface 2	X 2	Y 2	Target ID
ID	id1	surf1	x1	y1	id2	surf2	x2	y2	t_uid
Editable	yes	yes	yes	yes	yes	yes	yes	yes	yes

- *Element ID 1* defines the first element on which the target is calculated.
- *Surface 1* defines the first surface for the distance calculation if *Element ID 1* does not directly point to a Surface.
- *X 1* defines the X-coordinate on the first surface in mm.
- *Y 1* defines the Y-coordinate on the first surface in mm.
- *Element ID 2* defines the second element on which the target is calculated.
- *Surface 2* defines the first surface for the distance calculation if *Element ID 2* does not directly point to a Surface.
- *X 2* defines the X-coordinate on the second surface in mm.
- *Y 2* defines the Y-coordinate on the second surface in mm.

The distance is the 3D-distance between the two points on the two surfaces at the positions specified by *X 1*, *Y 1* and *X 2*, *Y 2*.

- General Distance 1D

The Optimization Goal has the following additional parameters:

Column	6	7	8	9	10	11	12	13	14	15
Name	Element ID 1	Surface 1	X 1	Y 1	Element ID 2	Surface 2	X 2	Y 2	Coordinate	Target ID
ID	id1	surf1	x1	y1	id2	surf2	x2	y2	coo	t_uid
Editable	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes

- *Element ID 1* defines the first element on which the target is calculated.

- *Surface 1* defines the first surface for the distance calculation if *Element ID 1* does not directly point to a Surface.
- *X 1* defines the X-coordinate on the first surface in mm.
- *Y 1* defines the Y-coordinate on the first surface in mm.
- *Element ID 2* defines the second element on which the target is calculated.
- *Surface 2* defines the first surface for the distance calculation if *Element ID 2* does not directly point to a Surface.
- *X 2* defines the X-coordinate on the second surface in mm.
- *Y 2* defines the Y-coordinate on the second surface in mm.
- *Coordinate* defines the axis along which the length is measured.

The distance is the 1D-distance between the two points on the two surfaces at the positions specified by *X 1*, *Y 1* and *X 2*, *Y 2*. The distance is measured along the X, Y or Z axis of the global coordinate system.

-  Aperture Radius

  Calculates the Physical Aperture of the selected surface.

  The Optimization Goal has the following additional parameters:

Column	7	8
Name	Surface	Target ID
ID	surf	t_uid
Editable	yes	yes

  - *Surface* defines the surface on which the target is calculated if the *Element ID* does not directly point to a Surface. The numbering follows the order in the Optical Design Editor.

-  Surface Sag

  Calculates the maximum sag of the selected surface on the

  The Optimization Goal has the following additional parameters:

Column	7	8	9	10
Name ID	Surface	Type	Resolution	Target ID
ID	surf	type	grid	t_uid
Editable	yes	yes	yes	yes

- *Surface* defines the surface on which the target is calculated if the *Element ID* does not directly point to a Surface. The numbering follows the order in the Optical Design Editor.
- *Type* defines how to measure the sag. Available options are "Min", "Max", "RMS", "Abs Max" and "PV".
- *Resolution* defines the grid of test points where the sag is evaluated. E.g. 5 would mean a grid of 5x5 points.

-  Aspheric Departure

  The Optimization Goal has the following additional parameters:

Column	7	8	9	10
Name	Surface	Rad Type	Radius	Target ID
ID	surf	rad_type	rad	t_uid
Editable	yes	yes	yes	yes

  - *Surface* defines the surface on which the target is calculated if the *Element ID* does not directly point to a Surface. The numbering follows the order in the Optical Design Editor.
  - *Rad Type* defines the method that is used to define the radius of the best fit sphere. "Edge" means, that the best fit sphere intersects the asphere at the vertex as well as the edge of the aperture. "Vertex" means that the radius of the best-fit-sphere is obtained by the radius of curvature of the asphere in the region of the vertex. "User" lets the user enter a radius manually from which the departure is calculated.
  - *Radius* defines the radius of the best-fit-sphere when *Rad Type* is set to "User".

-  Angle at Edge Calculates the Angle of the surface relative to the Y-Axis of the local surface coordinate system at the edge of the aperture in +Y direction.

  The Optimization Goal has the following additional parameters:

Column	7	8
Name	Surface	Target ID
ID	surf	t_uid
Editable	yes	yes

  - *Surface* defines the surface on which the target is calculated if the *Element ID* does not directly point to a Surface. The numbering follows the order in the Optical Design Editor.

- Surface Phase Gradient Calculates the gradient of the phase function of a surface.

  The Optimization Goal has the following additional parameters:

Column	7	8	9	10	11
Name	Surface	Direction	Type	Resolution	Target ID
ID	surf	dir	type	grid	t_uid
Editable	yes	yes	yes	yes	yes

  - *Surface* defines the surface on which the target is calculated if the *Element ID* does not directly point to a Surface. The numbering follows the order in the Optical Design Editor.
  - *Direction* defines whether the gradient is calculated in X-direction ("dX") or Y-direction ("dY") or the absolute gradient ("|| ||") is calculated.
  - *Type* defines how to measure the gradient. Available options are "Min", "Max", "RMS", "Abs Max" and "PV".

- Refractive Index

  Calculates the refractive index of an element material.

  The Optimization Goal has the following additional parameters:

Column	7	8	9
Name	Material	Wavelength	Target ID
ID	mat	lambda	t_uid
Editable	yes	yes	yes

  - *Material* defines the material of the element on which the target is calculated if "ID" is set to an element with multiple materials. The numbering follows the order in the Optical Design Editor.
  - *Wavelength* defines the wavelength in [nm] at which the refractive index is calculated.

- Abbe Number

  Calculates the Abbe Number of an element material.

  The Optimization Goal has the following additional parameters:

Column	7	8	9
Name	Material	Type	Target ID
ID	mat	type	t_uid
Editable	yes	yes	yes

  - *Material* defines the material of the element on which the target is calculated if "ID" is set to an element with multiple materials. The numbering follows the order in the Optical Design Editor.
  - *Type* defines the Abbe Number specification.

-  Model Material Offset

  Calculates a penalty function for Index / Abbe combinations of materials, that are not available in the Substitute Material Catalog.

  The Optimization Goal has the following additional parameters:

Column	6
Name	Target ID
ID	t_uid
Editable	yes

  - *ID* defines the element on which the target is calculated.
  - *Material* defines the material of the element on which the target is calculated if "ID" is set to an element with multiple materials. The numbering follows the order in the Optical Design Editor.
  - *Type* defines the Abbe Number specification.

18.4.5 User Defined Goals and Constraints

The previously listed optimization goals and constraints already cover the most common cases that are needed to setup a merit function. However, for some special cases it might be necessary to setup a Merit Function that requires some targets that are not available. For this purpose Quadoa provides two types of user defined targets, that can be defined either over math expressions or via a Python script.

18.4.5.1 Target ID

The *Target ID* parameter is a parameter most optimization goals and constraints have. It can be used to tag a goals or constraint with a unique ID in form of a string to refer to them from user defined Math Expression orPython Targets. If no user defined targets are defined the *Target ID* is ignored.

18.4.5.2 Math Expression and Python Target

- Math Expression Target

  Allows to insert a user defined math expression that can compute an arbitrary expression based on the value of other targets. The operands that are available for the math expression can be found in table 18.1. To set up a Math Expression Target first the Target ID of the targets to be used in the equation has to be defined. Assuming we have two targets defined, one with the Target ID "MyTarget1" and the second one "MyTarget2" we can set up a Math Expression target that computes "sin(MyTarget1) + 3.14 * MyTarget2" by entering this in the *Math Expression* parameter. Since all targets are computed from top to bottom, the Math Expression can only refer to other targets, positioned in rows that lie above the Math Expression.

  The Optimization Goal has the following additional parameters:

Column	6
Name	Math Expression
ID	
Editable	yes

Table 18.1: Overview on Math Expression Operands

Numeric Values		eg. 42, 1337, 3.1415 or pi
Parameters		Parameters defined in previous rows of the table
Basic Operators	+	plus
	-	minus
	*	multiplication
	/	division
	^	x to the power of y
Brackets	(	brackets
	)	brackets
Common Functions	abs	Abs. value of x
	exp	e to the power of x
	log	Nat. logarithm of x
	log10	Lob base 10 of x
	log2	Log base 2 of x
	sqrt	Squareroot of x
	root	nth root of x, root(x, n)
	sgn	sign of x (+1 if x>0 -1 if x>0)
Trigonometric Functions	acos	Arc cosine of x
	acosh	Inverse hyperbolic cosine of x
	asin	Arc sine of x
	asinh	Inverse hyperbolic sine of x
	atan	Arc tangent of x
	atanh	Inverse hyperbolic tangent of x
	atan2	Arc tangent of (x1 / x2), atan(x1, x2)
	cos	Cosine of x
	cosh	Hyperbolic cosine of x
	cot	Cotangent
	sin	Sine of x
	sinc	Sinc
	sinh	Hyperbolic sine of x
	tan	Tangent of x
	tanh	Hyperbolic tangent of x
	rad2deg	Convert x from radiant to degree
	deg2rad	Convert x from degrees to radians

-  Python Target The Python Target is the most flexible target that allows the user to define a merit function by providing a python script. As long it is possible to compute the merit function via a Math Expression that method should be preferred since in general the Math Expression will execute a lot faster that the Python Target. However this might not be possible in all cases.

  The Optimization Goal has the following additional parameters:

Column	6
Name	Class Name
ID	cname
Editable	yes

  The *Class Name* parameter is always available and specifies the python class in which the target is implemented. Depending on the additional parameters defined in the Python class additional parameters may be available. To make a Python class available through the python target it has to be implemented in the file called python_optimization_targets.py which needs to be located inside the python folder inside the User Data Folder. The default path is `C:\Users\UserName\Quadoa\python` on Windows or `/home/UserName/Quadoa/python` on Linux.

  Note that models with Python targets are not supported from within the Quadoa Python module or Matlab® toolbox or & C++ SDK.

  To define a Python Target Class the following methods have to be implemented:

  - **getParameters()** returns an array that contains the parameter names
  - **calc()** returns a double that is the value of the optimization target

  The Python Target has the capability to access the values of other Targets as well as variables. Further it has access to the following functions provided by Quadoa.

  - void traceSingleRay(double wave, double fx, double fy, double px, double py)
  - double getRayPosX(double surf)
  - double getRayPosY(double surf)
  - double getRayPosZ(double surf)
  - double getRayDirCosIncidentX(double surf)
  - double getRayDirCosIncidentY(double surf)

- double getRayDirCosIncidentZ(double surf)
- double getRayDirCosExitX(double surf)
- double getRayDirCosExitY(double surf)
- double getRayDirCosExitZ(double surf)
- double getRayOPL()
- double getGlobalRayPosX(double surf)
- double getGlobalRayPosY(double surf)
- double getGlobalRayPosZ(double surf)
- double getGlobalRayDirCosIncidentX(double surf)
- double getGlobalRayDirCosIncidentY(double surf)
- double getGlobalRayDirCosIncidentZ(double surf)
- double getGlobalRayDirCosExitX(double surf)
- double getGlobalRayDirCosExitY(double surf)
- double getGlobalRayDirCosExitZ(double surf)
- double getSurfaceNormalAtRayPosX(double surf)
- double getSurfaceNormalAtRayPosY(double surf)
- double getSurfaceNormalAtRayPosZ(double surf)
- double getTargetValue(string target_id)
- double getVariableValue(string var_id)

Example implementation for Python Target could look like this:

```
# Example Optimization Goal calculating the y-distance between two rays on the image surface
class RayDistanceY():
    # Method that tells Quadoa what parameters the Pyhton Target has
    def getParameters():
        return ["Wave", "Surf", "FieldX1", "FieldY1", "PupilX1", "PupilY1", "FieldX2", "FieldY2", "PupilX2", "PupilY2"]
    # Method to calc the (unweighted) value
    def calc(pyTarget, params):
        pyTarget.traceSingleRay(params["Wave"], params["FieldX1"], params["FieldY1"], params["PupilX1"], params["PupilY1"])
        x1 = pyTarget.getRayPosY(params["Surf"])
        pyTarget.traceSingleRay(params["Wave"], params["FieldX2"], params["FieldY2"], params["PupilX2"], params["PupilY2"])
        x2 = pyTarget.getRayPosY(params["Surf"])
        return x1-x2

# Example Optimization Goal computing the sum of two Targets with ID "T1" and "T2"
class AddT1AndT2():
    def getParameters():
        return []
    def calc(pyTarget, params):
        return pyTarget.getTargetValue("T1") + pyTarget.getTargetValue("T2")


# Example Optimization Goal computing the sum of two Variables with ID "T1" and "T2"
class AddV1AndV2():
    def getParameters():
        return []
    def calc(pyTarget, params):
        return pyTarget.getVariableValue("V1") + pyTarget.getVariableValue("V2")
```

18.5 Weight of Fields and Wavelengths

For many optimization tasks it is important to specify the relevance of different Fields or Wavelengths to each other, e.g. when optimizing a lens where the sharpness on the optical axis is more important than the sharpness at the outer fields. The relative weight of any Fields and Wavelengths is specified in the *Weight* parameter of each Field and Wavelength. The weight will be used when optimizing the Spot Radius RMS, PV and 1D as well for the Wavefront Error RMS and PV. For all other Optimization Goals, the weight is ignored, since it is typically more intuitive to directly specify the weight for each Optimization Goal in more complex Merit Functions.

18.6 Substitute Material Catalog

The Substitute Material List is used to specify a list of materials to used as Substitute Material. The optimizer will try to replace any of the Catalog Materials that are set as substitute candidate and - if the Merit Function can be improved - exchange the material in the model. The Model Material Offset Target uses the materials from the Substitute Material List as well to calculate the penalty based on the distance to the available materials.

It can be edited via the *Substitute Material Dialog* that can be found under the *Optimization Tab*.

18.7 Optimizer Types

Quadoa Optical CAD offers different kinds of Optimizers. Depending on the current stage of the design-process one or another may be more suitable for the task. As a rule of thumb, Global Optimization is typically used when searching the design space for possible solutions. The result of the Global Optimization can then be used as a starting point for the design process. If a coarse draft for the lens design already exists, Global Optimization typically is not needed. The Local- as well as the Extended Optimizer are typically the ones to work with most of the time. The The Local Optimizer will strictly search for the local minimum and is the fastest option. The Extended Optimizer will, besides searching for the local minimum, also try to find a better solution in the surrounding area of the current solution. This may include 'Jumps' to neighboring valleys in the potential landscape as well as non-continuous operations like Substituting a Material. When a good solution for a design is found, it is typically a good idea to try the Extended Optimizer for some while, since it is often capable to squeeze some more performance out of the design by exploring near areas that the strictly local optimizer cannot reach.

18.7.1 Local Optimization

The Local Optimizer searches for a local minimum in the design space by using one of the Internal Solver Types that can be specified in the Settings of the Merit Function. The advantage of this optimizer is, that it is the fastest, and - since it strictly optimizes the local minimum - it will not do any 'Jumps' to undesired solutions. Therefore, if the draft of the lens design is already finished and the design only needs a little tuning but no bigger change, the Local Optimizer optimizer typically is the best option. For interactively designing a lens by iterative manual tuning and optimizing the lens, the Local Optimizer is probably the most suitable as well.
The Dialog for local optimization specifies the following settings:

- *Solver* specifies the internal solver type. See chapter 18.8
- *xtol* specifies a tolerance for the change of the independent variable vector at which the optimization will be automatically aborted. The condition for the optimizier to finish is:

$$||\vec{x}_i - \vec{x}_{i-1}|| < xtol * (xtol + ||\vec{x}_i||) \tag{18.2}$$

- *ftol* specifies a tolerance for the merit function value change at which the optimization will be automatically aborted. The condition for the optimizier to finish is:

$$m_{i-1} - m_i < ftol * m_i \tag{18.3}$$

- *gtol* specifies a tolerance for the merit function derivates at which the optimization will be automatically aborted. The condition for the optimizier to finish is:

$$max_n(||\vec{J_n} \cdot \vec{b}||) < gtol \tag{18.4}$$

- *Max Cycles* specifies the maximum amount of internal loops to be performed, before the optimization is automatically canceled. This is measured by the number of times the Jacobian matrix is calculated.

Multiple exit conditions are available. The optimizer will finish if **one** of these are met. Any exit condition set to 0 will be ignored.

18.7.2 Extended Optimization

The Extended Optimizer searches for a minimum in the surrounding area of the current solution. Unlike the Local Optimizer, it is not strictly searching for the local minimum of the Merit Function, but is also capable to 'climb up the hill' to explore neighboring valleys of the current solution inside the design space. It is not limited to strictly continuous changes and can therefore also Substitute Materials. The Internal Solver Types to be used with the Extended Optimizer can be specified in the Settings of the Merit Function. The common application for the Extended Optimizer optimizer is typically to run it for some while, when there is already a promising solution for the design specification, but the performance is not quite reached or the user wants to try to find an even better solution.
The Extended Optimizer utilizes a local optimizer and therefore provides the corresponding Local Optimization Settings in the dialog. Additional settings are:

- *Max Sub Cycles* defines the maximum number of local optimization cycles to be performed.
- *Glass Selector* defines the glass selection approach.
  - *Random* selects random glasses from the substitution catalog
  - *Smart* selects glasses from the substitution catalog by their dispersion properties

If glass properties other than dispersion (e.g. transmission, TCE) have an impact on the merit function or if the model utilizes multiple thermal configurations *Random* should be selected. Otherwise *Smart* usually yields better results.

18.7.3 Global Optimization

The Global Optimizer searches for an arbitrary design, that is a local minimum of the Merit Function. Since the search space for optical designs is typically very high dimensional, it may take a long time to find a good solution. Furthermore, it can never be guaranteed that the best solution found is really the global optimum. The pure size of the design space makes it impossible to search it completely in a practical amount of time. Nevertheless, the Global Optimizer can be useful when searching for a good starting point for an optical design.
The Global Optimizer utilizes a local optimizer and therefore provides the corresponding Local Optimization Settings in the dialog. Additional settings are:

- *Max Sub Cycles* defines the maximum number of local optimization cycles to be performed.
- *Save Best* defines how many of the best results are saved during optimization.

Note, that since the Global Optimizer needs to optimize a wide range of systems that often lie outside the Constraints of the Dependent Parameters, it is recommended to use Soft Constraints during Global Optimization. The reason is, that it is often very time consuming to 'find the way out' of the constraint area, when starting far out in the forbidden parameter range. Since the power of the Global Optimization lies in the ability to try out a large number of possible options this leads to a reduced throughput of systems, and therefore a lower chance of finding a good solution in the same amount of time.

18.8 Solvers

Quadoa Optical CAD currently offers two types of solvers which are both based on the Levenberg-Marquardt (or Damped Least Squares) algorithm. These types of solvers require the number of optimization goals to be larger than the number of independent variables. Therefore it is usually recommended to use at least one RMS wavefront or spot size goal since these are compiled from a larger subset of goals for each ray.

- *DLS* uses a standard Levenberg-Marquardt algorithm which always takes a step when it improves the merit function

- *DLS TR* uses a trust-region variation of the Levenberg-Marquardt algorithm. A step is only taken if the system shows the expected linear behaviour within the variables trust region. If this is not the case - regardless of any merit function improvement - the trust region is reduced and the step is calculated again. This usually yields a more predictable result (the next local minimum of the merit function) but converges slower than the default DLS.

18.9 Optimization of Materials

Quadoa provides several methods for the optimization of materials, that come with different advantages and disadvantages. The following chapter gives a short overview on the methods.

18.9.1 Optimization by Substitution of Catalog Materials

When a Material is marked as *Substitute* in the *Optical Design Editor* it will be added to the list of materials that may be substituted during the optimization. The substitution requires an optimizer to be used that supports this feature, which is the case for the Extended Optimizer as well as the Global Optimizer. Furthermore at least one Material has to be added to the Substitute Material Catalog which contains any Materials that might be used by the optimizer. The advantage of optimizing materials this way is, that the model is directly defined by available real world materials. However the optimization can be slower compared to the method of optimizing model materials (see next chapter).

18.9.2 Optimization by optimizing Model Material Parameters

For the optimization using model materials it is necessary to convert any Material in the optical design to be optimized to a model type Material. When starting with a Catalog Material is can be easily converted to a Model Material via the right-click menu. The next step is to set the *Index* as well as the *Abbe Nr.* as variable. When running the optimization with material parameters as variables it is strongly recommended to add a Model Material Offset Target to the Merit Function. This target computes a penalty function for Index and Abbe Nr. combinations for which no real word material is available inside the Substitute Material Catalog. When a good design has been found using this optimization method, the model materials can be converted back to Catalog Materials via the Menu of the *Optical Design Editor* or via the right click menu of the Materials. After replacing the Materials it is typically required to run another short optimization since - depending on the weight of the Model Material Offset Target as well as the fit error of the Index/Abbe model, the

optical design might be a bit of. The advantage of this method is, that compared to the optimization by substitution of the materials the Local Optimization can be used. Furthermore, the method is typically faster compared to the substitution method.

19 Tolerance Analysis

The *Tolerancing Workbench* inside Quadoa Optical CAD allows the user to estimate the as-built performance of an optical system under real world conditions as well as to identify critical parameters for the alignment and accuracy of the optical components. Furthermore, this information can be used to optimize the system for a better as-build performance under looser tolerances. Overall, this will lead to an optical system that has a better as-build performance and can be produced with lower costs.
The difference between the as-build performance and the nominal design performance emerges from all the uncertainties that arise from the fabrication process of the optical system. For surfaces, this may be a form deviation from the nominal form, introduced e.g.through the polishing or diamond turning process. For elements like lenses, index variations of the glass substrate may change the refraction of the light, or the alignment uncertainty during the polishing process may lead to wedge errors. Also the manufacturing process for the mounting mechanics, as well as the alignment uncertainties of the optical and mechanical parts, will generate some errors regarding the position and orientation of single lenses or whole assemblies.

19.1 Setting up Tolerances

The first step for any tolerance analysis - independently on the exact analysis to be performed - is always to define the Tolerances for the system. The Tolerances are separated into three categories:

-  Surface Tolerance: This category holds tolerances that may be assigned to single Surfaces, like tilt or decenter as well as curvature or Form deviations.
-  Element Tolerance: This category holds tolerances that may be assigned to Single Elements like Position and Orientation, index or thickness tolerances.
-  Assembly Tolerance: This category holds tolerances that may be assigned to whole assemblies like Position and Orientation tolerances.

-  Multiconfig Tolerance: This category holds tolerances that may be assigned to Multiconfig Parameters.

19.1.1 Global and ID specific Tolerances

In general two kinds of tolerances, the Global Tolerances as well as Specific ID Tolerances, exist. Global Tolerances will be assigned to all objects inside the optical model if no Specific ID tolerance for that exact object exists. Specific ID Tolerances will only be assigned to the object with the *Identifier* the tolerance refers to (See chapter 7.7 on Element IDs).

If a Tolerance is a Global Tolerance, it will be displayed inside one of these Tolerance Containers:

each of these containers can only exist once inside a model. The Specific ID Tolerances can be added to one of the following Tolerance Containers:

The Specific ID Tolerance Containers have one parameter, which is the *Identifier* that is used to refer to the object (See chapter 7.7 on Element IDs).

Specific ID Tolerance Containers only need to be added, if the tolerances for a certain Surface, Element or Assembly differ from the tolerances in the model. This may be the case if the system is set up of parts from different manufacturers or some parts are fabricated using different processes. How to add Tolerance Containers to the model is explained in chapter 3.3.8.

Note, that if a tolerance for a Specific ID is defined more than once, always the first occasion of the tolerance will be used. For example if first a surface decenter tolerance in x direction is defined and afterwards a decenter tolerance for the same surface in x and y direction, the tolerance in x direction will have the value from the first definition, the tolerance in y direction the value from the second definition. Also if a tolerance for a specific element should be zero, but the tolerance is not zero for the global tolerances it has to be defined with a value of 0.0. Otherwise the default value from the global tolerances will apply.

19.2 Overview on Tolerance Parameters

Tolerance Parameters can be added to the Tolerance Containers to define the values of each tolerance. Depending on if the tolerance is valid for only certain elements or the whole system, a Specific ID Container or one of the Global Tolerance Containers has to be used (see chapter 19.1.1). Also depending on the type of the container only certain tolerances can be added.

All tolerance have the first four parameters in common. Depending on the *Statistic Distribution* *Param2* and *Param3* may be visible or not.

Column	1	2	3	4
Name	Distribution	Param1	Param2	Param3
ID	dist	dist_p0	dist_p1	dist_p2

The parameter *Distribution* defines the type of random distribution to be applied for a Monte Carlo Analysis (see chapter 19.4). For the Sensitivity Analysis the *Distribution* parameter has no effect on the result.

- If the *Distribution* parameter is set to "Rect", the random value will be somewhere in the interval $[-Range, Range]$ with equal chance for any of the values in between. The parameter *Value* defined the range of the distribution. For the sensitivity analysis the parameter perturbation is assumes to be $+Range$ in positive and $-Range$ in negative direction.
- If the *Distribution* parameter is set to "Rect Asymetric", the value will lie in the interval $[Min, Max]$ with equal chance for any of the values in between. For the sensitivity analysis the parameter perturbation is assumes to be Max in positive and Min in negative direction.
- If the *Distribution* parameter is set to "Triangluar", the random value will lie in the interval $[Min, Max]$ where the probability is 0.0 at Min and Max and rises linearly towards $Peak$ from both sides. For the sensitivity analysis the parameter perturbation is assumes to be Max in positive and Min in negative direction. For the sensitivity analysis the perturbation is taken as 3σ in both positive and negative direction.
- If the *Distribution* parameter is set to "Normal", the random value will be generated assuming a Gaussian normal distribution of the tolerance. The parameter *Value* defines the standard deviation of the distribution.
- If the *Distribution* parameter is set to "Skew Normal", the tolerance will be applied with the probability of a skew normal distribution. With the parameters *Sigma* (σ), *Shift* (μ_0)

and *Skewness* (α), the probability density function (PDF) of the skew normal distribution is defined as

$$PDF(x) = 2.0 * \sigma * PDF_{norm}(t) * CDF_{nomr}(\alpha * t) \tag{19.1}$$

where t is defined as

$$t = (x - \mu_0)/\sigma \tag{19.2}$$

For the sensitivity analysis the positive and negative values are assumed to be at the edge of the interval of the 99.73% (3.0σ) quantile.

- If the *Distribution* parameter is set to "Rayleigh", the random value will be generated assuming a Rayleigh distribution defined by the parameter *Sigma*. For negative *Sigma* the distribution is assumed to be towards the negative values. For the sensitivity analysis the perturbation is taken as 3σ in the direction of the Rayleigh distribution and 0.0 in the other direction.
- If the *Distribution* parameter is set to "Double Rayleigh", defines a distribution, where the PDF is defined as the sum of the PDF of two independent Rayleigh distributions, where the second distribution is weighted by *Relative Weight* relative to the first one. For the sensitivity analysis the perturbation is taken as 3σ of the single distribution, that spreads out most into positive or negative direction.
- If the *Distribution* parameter is set to "Parabolic", the random value will be generated assuming a parabolic distribution with the vertex of the parabola being located in the center between Min and Max with a chance of 0% and the highest chance is located at the Min and Max values.
- If the *Distribution* parameter is set to "Extreme", the random value will be set to exactly Min or Max by a 50% chance. For the sensitivity analysis the perturbation is Min in negative and Max in positive direction.

The following list gives an overview on the available tolerance parameters:

19.2.1 Position

The Position Tolerance defines the tolerance of the position of an element or assembly in x-, y- and z-direction inside the objects reference coordinate system. It has the following additional parameters:

Column	5
Name	Axis
ID	axis

The parameter *Axis* defines for which axis the tolerance should be applied. Following options are available: “X”, “Y”, “Z”, “XY”, “XYZ”. If tolerances for several axis with different values should be defined, this can be reached by defining multiple Position Tolerances with different settings for the values and axis.

19.2.2 Position Z

The Position Z Tolerance is used to tolerance the Z-Position of single surfaces like the image surface or a dummy surface. It can only be applied to The Specific ID Surface Tolerance Containers.

19.2.3 Angle

The Angle Tolerance defines the tolerance of the angular orientation of an element or assembly around the x-, y- and z-axis in the objects reference coordinate system. It has the following additional parameters:

Column	5
Name	Axis
ID	axis

The parameter *Axis* defines for which axis the tolerance should be applied. Following options are available: “X”, “Y”, “Z”, “XY”, “XYZ”. If tolerances for several axis with different values should be defined, this can be reached by defining multiple Angle Tolerances with different settings for the values and axis.

19.2.4 Surface Tilt

The Surface Tilt Tolerance defines the tolerance of the tilt of a surface around the x-, and y-axis in the objects reference coordinate system. It has the following additional parameters:

Column	5
Name	Axis
ID	axis

The parameter *Axis* defines for which axis the tolerance should be applied. Following options are available: "X", "Y", "XY". If tolerances for x and y with different values should be defined, this can be reached by defining two Tilt Tolerances with different settings for the values and axis.

19.2.5 Surface Decenter

The Surface Decenter Tolerance defines the tolerance of the decentering of a Surface around in x-, and y-direction in the objects reference coordinate system. It has the following additional parameters:

Column	5
Name	Axis
ID	axis

The parameter *Axis* defines for which axis the tolerance should be applied. Following options are available: "X", "Y", "XY". If tolerances for x and y with different values should be defined, this can be reached by defining two Tilt Tolerances with different settings for the values and axis.

19.2.6 General

The General Surface Tolerance defines the tolerance of any parameter of a surface feature, that has been added to the surface. It has the following additional parameters:

Column	5	6
Name	Surface Feature Row	Param
ID	feature	param

The parameter *Feature* is used to select the row of the surface feature item as it is listed in the *Optical Design Editor*. The parameter *Param* allows to select the parameter of the feature to be toleranced.

19.2.7 Thickness

The Thickness Tolerance defines the tolerance of the thickness of a lens. It has the following additional parameters:

Column	5
Name	Type
ID	type

The parameter *Type* defines how the thickness should add up for cemented doublets or triplets. This parameter is only important for the Monte-Carlo-Analysis. If the parameter is set to "Independent" then the position of each surface along the z axis will have a random error, however it will not influence the position of the following surfaces. If the parameter is set to "Add up" then the thickness error will affect the position of all following surfaces. In most cases "Add up" is the correct setting, since the thickness variation of the first part of a lens will lead to a shift of the position of all following parts when the lens stack is kitted together.

19.2.8 Index

The Index Tolerance defines the tolerance of the refractive index of the glass an element is made of. It has the following additional parameters:

Column	5
Name	Abs / Rel
ID	rel_type

The parameter *Abs / Rel* defines whether the tolerance is an absolute or a relative tolerance. If "Relative" is selected the index error is relative to the index of the material at the Fraunhofer d-line.

19.2.9 Abbe

The Abbe Tolerance defines the tolerance of the Abbe Vd of the glass an element is made of. It has the following additional parameters:

Column	5
Name	Abs / Rel
ID	rel_type

The parameter *Abs / Rel* defines whether the tolerance is an absolute or a relative tolerance. If "Relative" is selected the abbe error is relative to the Abbe Vd of the material.

19.2.10 Radius

The Radius Tolerance defines the tolerance of the radius of a surface. The tolerance can only be applied to surfaces that have a spherical or aspherical form. For general form deviations the Zernike Tolerance can be used. It has the following additional parameters:

Column	3	4
Name	Type	Wavelength
ID	rad_type	fringe_type

Depending on the setting of parameter *Type* the tolerance can be specified either "Absolute" , "Relative" of in "Fringes" . If "Fringes" is selected the conversion of radius error to fringes is calculated as follows:

$$Fringes = \frac{2\Delta R}{\lambda}(1 - \sqrt{1 - \frac{D}{2R}^2}) \tag{19.3}$$

19.2.11 Zernike

The Zernike Tolerance defines the tolerance of the Surface Form using a Zernike Polynomial to describe the sag deviation. The tolerance can be applied to any surface. The norm radius for the Zernike Polynomial will be automatically calculated from the surface aperture. If no aperture is defined for a surface, the radius of the floating aperture will be used. It has the following additional parameters:

Column	3	4	5
Name	Max Order	Piston	Tilt
ID	n_max	piston	tilt

- *Max Order* defines up to which order of polynomial modes the tolerance should be applied.
- *Pistion* defines if the piston term should be included. In general this parameter should be turned off, since the piston term will only lead to a displacement of the surface in z-direction which is already covered by the Position and Thickness tolerances.
- *Tilt* defines whether the tilt terms (Z_2 and Z_3) should be varied. In general, this is already considered through the surface tilt parameter and should therefore be turned off.

19.2.12 Multiconfig Param

The Multiconfig Param Tolerance defines the tolerance for a parameter that is defined in the Multiconfig Lookup Table. It has the following additional parameter:

Column	5
Name	Param
ID	param

The parameter *Param* allows to select the parameter from the Multiconfig Lookup Table to be toleranced.

19.3 Compensators

Some optical systems are build with tight tolerances, so that they can be directly used without the need of any alignment. This is especially the case for products with really large batch sizes - like e.g. lenses for smartphone cameras - where a manual alignment is not practicable. However, for many systems some kind of alignment process within the fabrication or the application is involved. The most common case is the adjustment of the image focus for a camera lens by the user. This means, that in that case tolerances, that lead to a variation of the focus of the system, do not affect the performance of the system, since the defocus can be easily compensated. Also for the setup of large complex systems, typically a manual alignment process is involved during the manufacturing process. For that, typically some kind of metrology is used to identify the misalignment of a component or assembly and then this information is used to adjust the alignment of the system. In addition the system performance will be increased, since some tolerances can be compensated by the manual alignment. Often these alignment strategies involve several steps, like first aligning assembly one, afterwards lens two and so on.

In Quadoa Optical CAD it is possible to model these alignment strategies reaching from simple focus adjustments to complex alignment procedures using Compensators. The Compensator object has the following parameters:

Column	1	2	3
Name	Comment	Merit Function	Sequence (optional)
ID	comment	merit	seq
Editable	yes	yes	yes

- *Comment* is a free text field used for documentation.
- *Merit Function* is used to select the Merit Function that is used for the optimization during compensation. Besides the Merit Functions that are defined by the user, two default Merit Functions are available: "Wavefront RMS" will adjust the Variables assigned to the Compen-

sator, so that the wavefront error is reduced to its minimum. "Spot Size RMS" will adjust the Variables assigned to the Compensator, so that the spot size is reduced to its minimum.

- *Sequence* is used to define the corresponding Sequence in case one of the default Merit Functions is used. If a user defined Merit Function is used this parameter is not available.

The activation state of a Variable can be set independently for each Compensator. To select the current Compensator to work on, the Combo-Box above the *Optical Design Editor* is used, by selecting the Compensator out of the drop-down menu.

If more than one Compensator is defined, the Compensator will be applied from top to bottom in the list. This allows to model complex alignment procedures, where one adjustment is performed after the other.

Any Tolerance with Compensators works like this: First the system including the tolerances is generated. Then the Compensators are applied one after another optimizing the Variables assigned to each Compensator using the Compensators Merit Function. In the next step, the system is evaluated using the Merit Function that is assigned for the evaluation of the system performance.

19.4 Monte-Carlo Tolerance Analysis

The Monte-Carlo Analysis is used to get a statistical estimation on the as-build performance of an optical system. Further, it is able to generate random exemplary systems that can be used to further analyze the optical properties of some random samples within the possible outcome. The Monte-Carlo Analysis will assign a random value to each Tolerance defined for the model. The value as well as the distribution for the individual tolerances is defined by the Tolerance Parameters. Each system will be evaluated, using the Merit Function that is assigned for the evaluation of the system performance. The number of systems to be generated for the statistics can be specified by the user. Further, it is possible to save all or only the best and worst outcome for further research.

19.5 Sensitivity Tolerance Analysis

The Sensitivity Analysis is used to identify tolerances that have a large impact on the overall system performance, or in general to estimate the impact of each individual tolerance on the system performance. The Sensitivity Analysis will generate one perturbed system for each tolerance and

evaluate its performance using the Merit Function that is assigned for the evaluation of the system performance. From this, it will generate a report that shows a sorted list on the impact of each single tolerance. The *Random Distribution* for each tolerance is ignored here. This means, that the report will always show the effect on the performance under the assumption that the tolerance is off by $\pm$ the value specified by the *Value* parameter.

19.6 Inverse Sensitivity Tolerance Analysis

20 Mechanics

An optical system can never exist alone without its mechanical mounting. Therefore, the integration of optical and mechanical design is of great importance. In Quadoa Optical CAD it is possible to include - besides the analytically defined Optical Elements - Mechanical Components directly inside the *Optical Design Editor*. This bidirectional exchange of files (Optical Design $\longleftrightarrow$ Mechanical CAD) makes the collaboration between optical and mechanical engineers a lot easier and allows to streamline the workflow. Besides common CAD formats that can be placed inside the optical model, some primitives are available as well, that can be used to sketch out first drafts for the optical mountings and mechanical parts. Any Mechanical Component can be placed inside the *Optical Design Editor* in the same way as any Optical Elements. The position is specified via the Position Parameters. To be able to import mechanical components into Quadoa a license that includes the Mechanics Analyzer Toolbox is required, which is not part of the basic Quadoa license. The export of CAD files is included in the basic version of Quadoa.

20.0.1 Primitives

The Primitive Mechanical Objects can be used to sketch a first design draft of the mechanical mounting or surrounding objects that need to be considered. They can be completely defined in Quadoa Optical CAD and do not require any additional CAD package to be used.

20.0.1.1  Cube

The Cube defines a simple cuboid shape. Besides the position parameters, the Cube has the following parameters:

Column	17	18	19
Name	Size X	Size Y	Size z
ID	sx	sy	sz

Where the parameters *Size X*, *Size Y* and *Size Z* define the size of the cube in length units. The origin of the cuboids coordinate system is at the center of gravity in the middle.

20.0.1.2  Sphere

The Sphere defines a simple spherical shape. Besides the position parameters, the Sphere has the following parameters:

Column	17
Name	Radius
ID	r

Where the parameters *Radius* define the radius of the sphere. The origin of the spheres coordinate system is at the center.

20.0.1.3  Cylinder

The Cylinder defines a simple cylindrical shape. Besides the position parameters, the Cylinder has the following parameters:

Column	17	18
Name	Length	Radius
ID	l	r

Where the parameters *Radius* define the radius of the cylinder and *Length* the length. The cylinders length is defined in z-direction of the local coordinate system and the origin of the cylinder is at the center of the left side.

20.0.1.4  Tube

The Tube defines a simple tube like shape. Besides the position parameters, the Tube has the following parameters:

Column	17	18	19
Name	Length	Inner Rad	Outer Rad
ID	l	rad_in	rad_out

Where the parameters *Inner Rad* define the radius of the hole, *Rad Outer* defines the radius of the outer cylinder and *Length* the length of the tube. The tubes length is defined in z-direction of the local coordinate system and the origin of the tube is at the center of the left side.

20.0.2 CAD Parts

The Mechanical CAD Parts allow it to directly import mechanical parts that have been generated by a third party CAD software into Quadoa Optical CAD. For stock optics many manufacturers provide some kind of CAD files for the lens or mirror mountings they sell besides the optical specification. Also when constructing mechanical parts for custom optical system, it is often useful to import the CAD. By importing the CAD, it can be directly checked if there are any errors in the design, e.g. a wrong dimension, that would lead to a false positioning of some component or some unwanted obscuration of the optical path due to some mounting that has not been considered. Following CAD file formats are supported:

20.0.2.1  STL

The STL Mech Part has, besides the Position Parameters the following parameters:

Column	17
Name	File Name
ID	file

The parameter *File* specifies the STL-file to be loaded. The file has to be placed within the same folder, where the optical model file (*.optx) is placed. If the model is saved at a new location, the file needs to be copied to the new folder as well.

20.0.2.2  STEP

The STEP Mech Part supports STEP files including STEP files with colored shapes. In the case of a colored STEP model, the default mechanics color will be replaced by the colors specified in the STEP file. The STEP Mech Part has, besides the Position Parameters the following parameters:

Column	17	18
Name	File Name	Precision
ID	file	prec

- *File* specifies the STEP-file to be loaded. The file has to be placed within the same folder, where the optical model file (*.optx) is placed. If the model is saved at a new location, the file needs to be copied to the new folder as well.
- *Precision* specifies the precision for the meshing of the file. The result of the meshing is used for the display in the *3D-View* as well as for the intersection calculation of rays. The value entered is the maximum allowed deviation in length units from the real shape in length units.

  Note that a small value for the meshing may lead - depending on the size as well as the shape of the CAD file - to a long computation time.

20.0.2.3 IGES

The IGES Mech Part has, besides the Position Parameters the following parameters:

Column	17	18
Name	File Name	Precision
ID	file	prec

- *File* specifies the IGES-file to be loaded. The file has to be placed within the same folder, where the optical model file (*.optx) is placed. If the model is saved at a new location, the file needs to be copied to the new folder as well.
- *Precision* specifies the precision for the meshing of the file. The result of the meshing is used for the display in the *3D-View* as well as for the intersection calculation of rays. The value entered is the maximum allowed deviation in length units from the real shape in length units.

  Note that a small value for the meshing may lead - depending on the size as well as the shape of the CAD file - to a long computation time.

20.0.3 Mechanics Analysis

The Ray Intersection Analysis allows to check for any collisions in between rays and mechanical parts by clicking on the Check Ray Intersection button. It can be found in the *Mechanics* tab under Analysis. If any intersection is found, the count of the intersecting rays will be displayed. To easily locate the parts of the model that lead to the intersections, the intersecting parts of the model will be highlighted in red within the *3D-View*. The highlighting can be cleared by clicking the Clear Highlight button.

21 Lens Catalog

The Lens Catalog holds a wide range of lenses that are available from manufacturers as stock optics. It can be found in the *Setup* tab under the *Lens Catalog* button. Lenses are sorted by manufacturer and can be browsed inside the catalog. Furthermore, it is possible to filter the lenses by type as well as the most important parameters like focal length or aperture diameter. Own lens catalogs containing optical components of other manufacturers or of some other Lenses or Assemblies that are commonly used, can be added to the catalog as well. To generate an user defined lens catalog, the catalog file needs to be copied to the 'lens' folder inside the Quadoa installation path. How to setup a catalog file is specified in 25.1.3.

22 Material Catalogs

In Quadoa Optical CAD, two types of *Material Catalogs* are available, the *Lens Material Catalogs* as well as the *Coating Material Catalog*.

22.1 Lens Material Catalog

The *Lens Material Catalog* contains a wide range of optical materials that are commonly used for lenses. Besides a wide range of optical glass types, common polymer optical materials as well as common liquids are included. The *Lens Material Catalog* can be found under the *Setup* tab by clicking on the *Materials* button. Furthermore, the catalog can be directly opened from the Materials *Name* field in the Optical Design Editor, by clicking on the catalog button besides the edit field. Another way to open the Materials is via the *Lens Wizard* by clicking on the catalog icon besides the *Material Name* edit field.

Inside the catalog the Materials can be filtered by their refractive index or Abbe number or searched by name. To get a quick overview on the properties of all Materials, the Index- and Abbe number are indicated in a small bar chart for every material. Besides that, an Abbe Diagram Plot is available, where the Materials are sorted in a plot with the x-axis showing the Abbe number and the y-axis the refractive index. To select a Material from the plot, the Materials can be clicked and then inserted by the *Insert Material* button.

If a Material is not included in the *Lens Material Catalogs*, it can be added by copying the material catalog file to the 'glass' folder inside the Quadoa installation path. How to setup a catalog file is specified in 25.1.2.1.

22.2 Coating Material Catalog

The *Coating Material Catalogs* contains the most common materials that are used for Thin Film Stack Coatings. Besides typical Fluorides and Oxides that are widely used for AR-coatings, some metal materials are included as well. If a material is not included in the *Lens Material Catalogs*, it can be added by copying the material catalog file to the 'coat' folder inside the Quadoa installation path. How to setup a catalog file is specified in 25.1.2.2.

23 Coatings Catalog

24 Scripting Interfaces

All algorithms used in Quadoa Optical CAD are encapsulated in a library, that gets called by the user interface. The library can also be called directly from a scripting language or from within own projects. This concept makes it possible to run any functions of Quadoa Optical CAD completely independent from the user interface. To get access to the scripting interfaces, the Toolbox Scripting Interface Python, Matlab® & C++ SDK is required, which is not part of the basic Quadoa license.

24.1 Python

The Quadoa Optical CAD Module for Python provides an interface for accessing the features of the QuadoaCore library directly from Python. Any Quadoa Optical CAD model (*.optx) can be loaded and simulated. The interface provides access to the basic features like:

- Viewing and changing parameters of the optical design (e.g. radius of curvature, lens positions, polynomial coefficients, ...)
- Viewing and changing parameters of the sequence definition (e.g. wavelengths, field definitions, ray distribution, aperture stop position, ...)
- Performing raytraces, compute PSF/MTF, simulate ghost images, run optimization, ...
- Obtain data from the raytrace (e.g. ray data, wavefront data, ...)
- Obtain data on lens/surface geometry (surface sag, surface phase map, gradient, ...)

24.1.1 Installation

The Python scripting interface can be installed to any Python environment using pip.

```
pip install /path/to/quadoa-python-wrapper.whl
```

The minimum supported version of python is Python 3.6

24.1.2 Function Overview

An overview over all available functions can be found in the Python-Documentation for Quadoa Optical CAD. The documentation is available under the *Help* tab on the *Python Help* button.

24.1.3 Python Wizard

To set up a new Python script, Quadoa Optical CAD offers the Python Script Wizard. It is available under the *Setup* tab on the *Pyhton Wizard* button. Here the user can choose between an empty script as well as some templates for common script tasks. The Pyhon script can directly be launched from within the GUI under the *Run* button.

Note that the script should be placed inside the folder where the model is located. Otherwise it is not available under the Run button from within the user interface.

24.1.3.1 Adding a Python Script Templates

A script template is basically a Python script that typically starts with the following lines:

```
########################################
#       Scripttemplate Example         #
########################################
from quadoa import *
import numpy as np
from matplotlib import pyplot as plt

# allocate a Quadoa core Object
core: QuadoaCore = QuadoaCore()

::LOAD_MATERIALS_AND_INIT_MODEL::
```

After these first scripting lines, any Quadoa function calls or any other Python code can be added.

To make the template available, the file needs to be saved with the file extension '.qpyt' and copied to the /scripttemplates/python/ folder inside the Quadoa installation. When using the template to generate a script, Quadoa Optical CAD will replace the ::LOAD_MATERIALS_AND_INIT_MODEL:: keyword with the call to load the model that is currently opened in Quadoa when generating the script as well as loading all catalogs that are necessary for the model like e.g. lens materials. The script can be called directly from the *Run* button in Quadoa Optical CAD, or also as standalone from any Python environment.

24.1.4 Simple Python Example

To work with Quadoa Optical CAD in Python, the first step is always to import the Quadoa module and allocate a QuaodaCore object:

```
from quadoa import QuadoaCore
core = QuadoaCore()
```

The QuadoaCore object can hold one instance of a Quadoa optical design file (*.optx). It allows to perform raytraces, manipulate the model by changing parameters or run an optimization, get ray data or other model related data as well to export data from the model (e.g. CAD files).

To be able to initialize the glass models used by the optical design, the glass catalogs that are used in the model need to be loaded first:

```
QuadoaBaseDir = "/path/to/Quadoa/"  # path to the Quadoa installation

core.loadMaterialFile(QuadoaBaseDir + "/glass/schott.glas")
core.loadMaterialFile(QuadoaBaseDir + "/glass/ohara.glas")
```

Here it is assumed, that the model uses some glass from Schott as well as from Ohara. After all glass catalogs are loaded, the model can be loaded:

```
MyModelsBaseDir = "/path/to/my/model"  # path to the folder with the model to load
core.loadModelFile(MyModelsBaseDir + "/My_Lens.optx")
```

If the model can be found under the path and all needed materials are correctly loaded, the function will return true. Otherwise an error messag will be displayed.

Now a raytrace can be performed:

```
core.traceAllRays()
```

After the raytrace, the ray data can be obtained from the model. To plot a simple spot diagram, this example will be used:

```
# first we need to specify which sequence,
# field, wavelength combination we want
# to get the data for and on which surface
seq_nr = 0   # the index of the sequnce we are interested in
field_nr = 0 # the index of the field
wave_nr = 0  # the index of the wavelength
surf_nr = core.getSequenceCameraSurface(seq_nr) # the last surface

# get the ray data
MySpotData = core.getRayPos(seq_nr, field_nr, wave_nr, surf_nr)
```

MySpotData now holds the x, y-position of the rays in the image plane. For accessing the data returned from Quadoa Optical CAD in combination with other Python librarys, the most convenient way is to convert the DataObjects into numpy arrays. (See DataObject).

Now the numpy- and the matplotlib library is used to plot the ray data. Therefore, both modules have to be imported (this should be added at the beginning of the script):

```
import numpy as np
from matplotlib import pyplot as plt

and finally plot the ray data:

# convert the DataObject to a numpy array
npspot = np.array(MySpotData, copy=False)

# plot the spot diagram
plt.plot(npspot[0,:], npspot[1,:], ".", markersize=1)
plt.set_xlabel("x / mm")
plt.set_ylabel("y / mm")
plt.axis('equal')

plt.show()
```

24.2 Matlab

The Quadoa Optical CAD Matlab Toolbox provides an interface for accessing the features of the QuadoaCore library directly from Matlab. Any Quadoa Optical CAD model (*.optx) can be loaded and simulated. The interface provides access to the basic features like:

- Viewing and changing parameters of the optical design (e.g. radius of curvature, lens positions, polynomial coefficients, ...)
- Viewing and changing parameters of the sequence definition (e.g. wavelengths, field definitions, ray distribution, aperture stop position, ...)
- Performing raytraces, compute PSF/MTF, simulate ghost images, run optimization, ...
- Obtain data from the raytrace (e.g. ray data, wavefront data, ...)
- Obtain data on lens/surface geometry (surface sag, surface phase map, gradient, ...)

Note that the features of the Quadoa Wave Optics Toolox are currently not supported within the Matlab® toolbox.

24.2.1 Installation

- extract the Quadoa Matalb Toolbox zip-archive to a folder on you PC
- Open Matlab
- Go to the 'Home' tab, in the 'Environment' section, click 'Set Path'. The Set Path dialog box appears.
- Click 'Add Path' and set the path to the folder where the Quadoa Matalb Toolbox was extracted to.
- Click 'Save' and close the dialog

24.2.2 Function Overview

An overview over all available functions can be found in the Quadoa-Matlab-Documentation. The documentation is available under the *Help* tab on the *Matlab Help* button.

24.2.3 Matlab Wizard

To set up a new Matlab script, Quadoa Optical CAD offers the Matlab Script Wizard. It is available under the *Setup* tab on the *Matlab Wizard* button. Here the user can choose between an empty script as well as some templates for common script tasks.

24.2.3.1 Adding a Matlab Script Templates

A script template is basically a Matlab script that typically starts with the following lines:

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%        Scripttemplate Example      %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% allocate a Quadoa core Object
core = QuadoaCore();

::LOAD_MATERIALS_AND_INIT_MODEL::
```

After these starting lines, any Quadoa function call or any other Matlab code can be added.

To make the template available, the file needs to be saved with the file extension '.qmlt' and copied to the /scripttemplates/matlab/ folder inside the Quadoa installation. When using the template to generate a script, Quadoa Optical CAD will replace the ::LOAD_MATERIALS_AND_INIT_MODEL:: keyword with the call to load the model that is currently opened in Quadoa when generating the script as well as loading all catalogs that are necessary for the model like e.g. lens materials.

24.2.4 Script Example

To work with Quadoa Optical CAD in Matlab the first step is always to allocate a QuaodaCore object:

```
core = QuadoaCore();
```

The QuadoaCore object can hold one instance of a Quadoa optical design file (*.optx). It allows to perform raytraces, manipulate the model by changing parameters or run a optimization, get ray data or other model related data as well to export data from the model (e.g. CAD files).

To be able to initialize the glass models used by the optical design, the glass catalogs that are used in the model need to be loaded first:

```
QuadoaBaseDir = '/path/to/Quadoa/'  % path to the Quadoa installation

core.loadMaterialFile([QuadoaBaseDir '/glass/schott.glas']);
core.loadMaterialFile([QuadoaBaseDir '/glass/ohara.glas']);
```

Here it is assumed, that the model uses some glass from Schott as well as from Ohara. After all glass catalogs are loaded, the model can be loaded:

```
MyModelsBaseDir = '/path/to/my/model';  % path to the folder with the model to load
core.loadModelFile([MyModelsBaseDir '/My_Lens.optx']);
```

If the model can be found under the path and all needed materials are correctly loaded, the function will return true. Otherwise an error message will be displayed.

We now can perform a raytrace::

```
core.traceAllRays();
```

After the raytrace, the ray data can be obtained from the model. To plot a simple spot diagram, this example can be used:

```
% first we need to specify which sequence,
% field, wavelength combination we want
% to get the data for and on which surface
seq_nr = 0;   % the index of the sequnce we are interested in
field_nr = 0; % the index of the field
wave_nr = 0;  % the index of the wavelength
surf_nr = core.getSequenceCameraSurface(seq_nr); % the image surface

% get the ray data
MySpotData = core.getRayPos(seq_nr, field_nr, wave_nr, surf_nr);
```

MySpotData now holds the x, y-position of the rays in the image plane and is of type of a matlab matrix. Thr raydata can now be plotted:

```
figure()
plot(MySpotData(:,1), MySpotData(:,2), 'r*');
xlabel("x / mm")
ylabel("y / mm")
```

25 File Formats

25.1 Quadoa File Formats

All file formats that are used in Quadoa Optical CAD for the model as well as the catalogs are basically XML files. The encoding used for the files is UTF-8. The advantage of using such a widely used open and clear text file format is, that a wide range of tools for editing the files exist. Further, the changes can be easily tracked with common versioning tools (see chapter 25.2), especially when working on a model as a team.

25.1.1 Optical Model File

The Quadoa Optical CAD Optical Model File has the file name extension '.optx'. Its basic structure is as follows:

```
<?xml version="1.0"?>
<optical_system_definition>
        <version major="2021" minor="6" />
        <geometry />
        <environment />
        <sequences />
        <multiconfig />
        <optimization />
        <substitute_materials />
        <tolerancing />
</optical_system_definition>
```

- The <geometry> node holds the data that is available over the *Optical Design Editor* via the gui.
- The <environment> node holds the environment material (Typically air)
- The <sequences> node holds any Sequence that is defined within the model
- The <multiconfig> node holds the Multiconfiguration Lookup Table values
- The <optimization> node holds any Merit Functions

- The <substitute_materials> node holds the Substitute Material Catalog defined for the optimization
- The <tolerancing> node holds the Tolerancing Parameters of the model.

The structure of the nested objects inside each of the main nodes is the same as it can be seen in the *Optical Design Editor* as well as in the *System Setup Editor*. For each object that is available in Quadoa Optical CAD, the parameters are listed in the parameter table. The 'ID' - row shows the parameter name, that is used within the file. The best way to learn the structure is to open a '.optx' file in a text or xml editor.

25.1.2 Material Catalog File

In Quadoa Optical CAD two types of *Material Catalogs* are available, the *Lens Material Catalogs* as well as the *Coating Material Catalog*. The file formats are described in the following chapters:

25.1.2.1 Lens Material Catalog File

The Lens Material Catalog File holds the date for the refractive materials of any optical components. The file has the file name extension '.glas'. To use a file from within the user interface, the file must be located either inside the 'glass' folder inside the Quadoa installation path, or alternatively in the same folder as the model that used the materials. When working from a Scripting Interface, the file may be located anywhere since the full path to the file needs to be specified for loading. Any Material Catalog File is set up as following:

```
<?xml version="1.0" encoding="utf-8"?>
<materialcatalogs>
    <materialcatalog name="MyGlassCatalog">
        <!-- here the single materials are added -->
  </materialcatalog>
</materialcatalogs>
```

Note, that the name of the glass catalog should always be the same as the file name, since the name will be used in the script template wizard for loading the corresponding material files.

The following material definitions are available:

- **Sellmeier Glass Specification**: the dispersion formula according to the Sellmeier Equation is specified as

$$n^2 - 1.0 = \frac{B_1 * \lambda^2}{\lambda^2 - C_1} + \frac{B_2 * \lambda^2}{\lambda^2 - C_2} + \frac{B_3 * \lambda^2}{\lambda^2 - C_3} \tag{25.1}$$

```
<?xml version="1.0" encoding="utf-8"?>
<materialcatalogs>
    <materialcatalog name="MyGlassCatalog">
        <sellmeier name="MyGlassName1" ref_temp="20.0"
                   B1="0.0" B2="0.0" B3="0.0"
                   C1="0.0" C2="0.0" C3="0.0"
                   D0="0.0" D1="0.0" D2="0.0" E0="0.0" E1="0.0" lambda_ref="0.0" />
        <sellmeier name="MyGlassName2" ref_temp="20.0"
                   B1="0.0" B2="0.0" B3="0.0"
                   C1="0.0" C2="0.0" C3="0.0"
                   D0="0.0" D1="0.0" D2="0.0" E0="0.0" E1="0.0" lambda_ref="0.0" />
        .
        .
        .
  </materialcatalog>
</materialcatalogs>
```

- **Extended Sellmeier Glass Specification**: the dispersion formula according to the Extended Sellmeier Equation is specified as

$$n^2 - 1.0 = \frac{B_1 * \lambda^2}{\lambda^2 - C_1} + \frac{B_2 * \lambda^2}{\lambda^2 - C_2} + \frac{B_3 * \lambda^2}{\lambda^2 - C_3} + \frac{B_4 * \lambda^2}{\lambda^2 - C_4} + \frac{B_5 * \lambda^2}{\lambda^2 - C_5} \tag{25.2}$$

```
<?xml version="1.0" encoding="utf-8"?>
<materialcatalogs>
    <materialcatalog name="MyGlassCatalog">
        <sellmeier_ext name="MyGlassName1" ref_temp="20.0"
                   B1="0.0" B2="0.0" B3="0.0" B4="0.0" B5="0.0"
                   C1="0.0" C2="0.0" C3="0.0" C4="0.0" C5="0.0"
                   D0="0.0" D1="0.0" D2="0.0" E0="0.0" E1="0.0" lambda_ref="0.0" />
        <sellmeier_ext name="MyGlassName2" ref_temp="20.0"
                   B1="0.0" B2="0.0" B3="0.0" B4="0.0" B5="0.0"
                   C1="0.0" C2="0.0" C3="0.0" C4="0.0" C5="0.0"
                   D0="0.0" D1="0.0" D2="0.0" E0="0.0" E1="0.0" lambda_ref="0.0" />
        .
        .
        .
  </materialcatalog>
</materialcatalogs>
```

- **Series Expansion Glass Specification**: the dispersion formula according to the Sereis Expansion Equation is specified as

$$n^2 = A_0 + A_1 * \lambda^2 + A_2 * \lambda^{-2} + A_3 * \lambda^{-4} + A_4 * \lambda^{-6} + A_5 * \lambda^{-8} \tag{25.3}$$

```
<?xml version="1.0" encoding="utf-8"?>
```

```
<materialcatalogs>
    <materialcatalog name="MyGlassCatalog">
        <series_expansion name="MyGlassName1" ref_temp="20.0"
                A0="0.0" A1="0.0" A2="0.0" A3="0.0" A4="0.0" A5="0.0"
                D0="0.0" D1="0.0" D2="0.0" E0="0.0" E1="0.0" lambda_ref="0.0" />
        <series_expansion name="MyGlassName2" ref_temp="20.0"
                A0="0.0" A1="0.0" A2="0.0" A3="0.0" A4="0.0" A5="0.0"
                D0="0.0" D1="0.0" D2="0.0" E0="0.0" E1="0.0" lambda_ref="0.0" />
        .
        .
        .
  </materialcatalog>
</materialcatalogs>
```

- **Series Expansion 2 Glass Specification**: the dispersion formula according to the Sereis Expansion 2 Equation is specified as

$$n^2 = A_0 + A_1 * \lambda^2 + A_2 * \lambda^4 + A_3 * \lambda^{-2} + A_4 * \lambda^{-4} + A_5 * \lambda^{-6} + A_6 * \lambda^{-8} + A_7 * \lambda^{-10} + A_8 * \lambda^{-12} \tag{25.4}$$

```
<?xml version="1.0" encoding="utf-8"?>
<materialcatalogs>
    <materialcatalog name="MyGlassCatalog">
        <series_expansion_2 name="MyGlassName1" ref_temp="20.0"
                A0="0.0" A1="0.0" A2="0.0" A3="0.0" A4="0.0" A5="0.0" A6="0.0" A7="0.0" A8="0.0"
                D0="0.0" D1="0.0" D2="0.0" E0="0.0" E1="0.0" lambda_ref="0.0" />
        <series_expansion_2 name="MyGlassName2" ref_temp="20.0"
                A0="0.0" A1="0.0" A2="0.0" A3="0.0" A4="0.0" A5="0.0" A6="0.0" A7="0.0" A8="0.0"
                D0="0.0" D1="0.0" D2="0.0" E0="0.0" E1="0.0" lambda_ref="0.0" />
        .
        .
        .
  </materialcatalog>
</materialcatalogs>
```

25.1.2.2 Coating Material Catalog File

The Coating Material Catalog File holds the data for the coating materials used by Thin Film Stack coatings. The file has the file name extension '.glas'. To use a file from within the user interface, the file must be located inside the 'coat' folder inside the Quadoa installation path. When working from a Scripting Interface, the file may be located anywhere since the full path to the file needs to be specified for loading. Any Material Catalog File is set up as following:

```
<?xml version="1.0" encoding="utf-8"?>
<materialcatalogs>
    <materialcatalog name="MyGlassCatalog">
        <!-- here the single materials are added -->
```

```
  </materialcatalog>
</materialcatalogs>
```

The coating materials are typically described in the form of lookup tables where *l* defines the wavelength in mm, *n* is the refractive index at the wavelength, and *k* is the complex part of the refractive index.

```
<?xml version="1.0" encoding="utf-8"?>
<materialcatalogs>
        <materialcatalog name="MyMaterialCatalotName1">
                <interp_mat name="MyLookupMaterialName" source="info on the source">
                        <nk l="0.0" n="0.0" k="0.0"/>
                        <nk l="0.0" n="0.0" k="0.0"/>
                        <nk l="0.0" n="0.0" k="0.0"/>
                        .
                        .
                        .
                </interp_mat>
                <interp_mat name="MyLookupMaterialName2" source="info on the source">
                        <nk l="0.0" n="0.0" k="0.0"/>
                        <nk l="0.0" n="0.0" k="0.0"/>
                        <nk l="0.0" n="0.0" k="0.0"/>
                        .
                        .
                        .
                </interp_mat>
                .
                .
                .
        </materialcatalog>
</materialcatalogs>
```

Note, that besides lookup table materials, it is also possible to use the other glass definitions used for lens materials in combination with Thin Film Stack coatings. However, the most common description is in the form of lookup tables.

25.1.3 Lens Catalog File

The Lens Catalog File formate is defined as following:

- The <lenscatalogs> top level node includes any Lens Catalog.
- The <lenscatalog> node defines a lens catalog. The catalogs may be nested to define sub catalogs, e.g. all singlets, doublets and so on from one manufacturer.
- The <lensinfo> node is used to specify some information about the lens. It is used to search and filter the lenses in the Lens Catalog. Here the focal length *f*, the aperture radius *ap* as well

the category the lens belongs to *basic_cat*, *main_cat* and *sub_cat* as well as the materials the lens is made of can be specified.

- The <lens> node inside each <lensinfo> node specifies the lens itself. It is set up in the same way as any lens in the Optical Model File.

Note that besides lenses also complete assemblies may be added to the catalog

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<lenscatalogs>
        <lenscatalog name="MyMainLensCatalog">
        <lenscatalog name="MySubCategoryLensCatalog">
        <lensinfo name="MyLensName" f="0.0" ap_r="0.0" basic_cat="lens" main_cat ="singlet" sub_cat ="planoconvex" material="Mat
        <lens>
        <material model_type="Catalog" name="MaterialName"/>
        <surf radius = "0.0">
        <float_ap name="radius" value="0.0" is_locked="true" />
        </surf>
        <surf z="0.0">
<surf radius = "0.0">
        <float_ap name="radius" value="0.0" is_locked="true" />
        </surf>
        </lens>
        </lensinfo>
        .
        .
        .
</lenscatalog>
</lenscatalog>
</lenscatalogs>
```

25.2 Versioning of Models and Catalalogs

Any Quadoa model file is a basic clear text ASCII xml file. The advantage of such file formats is, that it can be versioned with any common versioning tools like GIT or SVN. This makes it very comfortable especially when working on a project as a team.

25.3 Supported Third Party File Formats

25.3.1 Rayfile

Ray files are used to simulate real world light sources and can be obtained for a wide range of available light sources directly from the manufacturers. Besides that, they can be used to define custom (extended) sources with a specific angular distribution of the rays. Any ray file contains a list of a large number of rays that can be used as Source Distribution for any Sequence. To make a ray file available inside Quadoa the file has to be placed in the same folder the model is stored in. It then will be available via the *Ray File* parameters drop down menu in the Distribution. Quadoa supports ASCII as well as binary ray file formats.

An example ASCII rayfile looks like follows:

```
X_Pos Y_Pos Z_Pos X_Vec Y_Vec Z_Vec Flux
0.998108  1.4440763 −0.4057631 −0.2751806 0.4236943 −0.8629941 0.001
−2.2943354 −1.0808337 −1.0400954 −0.0198361 −0.689097 0.7243976 0.001
−0.2169416 1.1585445 −0.952772 0.9389154 −0.2882778 0.1879727 0.001
.
.
.
```

Each row contains the information about one ray. The first three columns define the rays x, y and z start position in the coordinate system of the first Surfaces within the Sequence in mm. The 4th to 6th column define the direction of the ray in x, y and z direction. The direction vector has always be normed to 1.0. The 7th column defines the flux of each ray.

The binary ray file format is composed of a header followed by the values for each ray. The header has a size of 208 bytes and is always ignored to be compatible with a wide range of ray file formats available.

After the header, each ray entry with a size of 28 bytes is defined as follows:

```
typedef struct
{
        float X, Y, Z;
        float L, M, N;
        float Flux;
} ray;
```

The *X*, *Y* and *Z* value define the rays start position in the coordinate system of the first Surfaces within the Sequence in mm. The *L*, *M* and *N* values define the direction of the ray in x, y and z direction. The direction vector has always be normed to 1.0. The *Flux* value defines the flux of

each ray.

25.3.1.1 Generationg Custom Rafiles

Custom ray files can easily be generated with any scripting language. Some example scripts are contained in the /examples/Source/ folder within the Quadoa installation path.

25.3.2 Import/Export of Third Party Models

To be able to work with models that have been generated with other optical design software, Quadoa Optical CAD is capable of importing and exporting Sequences to a list type structure as used in most sequential ray tracing software.

Note that if more than one Sequence is defined within a model, each Sequence has to be exported individually since most other software does not support Multi-Sequential ray tracing.

The Import and export feature can be found under the *File* tab on the *Import Optical Design* and *Export Optical Design* buttons. The *Import Assembly* button will also import lens data, however it will be grouped inside an Assembly and no Sequence data will be imported. This can be used to import e.g. lenses or objectives that are used as subassemblies of a larger model. Supported file types for the import and export are '.zmx' files, '.seq' files as well '.csv' files.

26 Polynomials

The following conventions apply to all polynomials used in Quadoa Optical CAD. This is especially the case for the polynomial Form Objects as well as the Polynomial Phase Objects. Furthermore, if e.g. a Zernike-Polynomial fit is performed to characterize a wavefront, the same conventions are used.

26.1 Zernike Polynomials

There are different kinds of Zernike Polynomials implemented in Quadoa Optical CAD. The main difference is the sorting of the polynomials in respect to their radial and angular modes. Furthermore, the norming prefactor may differ. The x, y and the polar coordiante for the polynomial evaluation is always normed to the norm radius r_n.

The even polynomials are defined by

$$Z_n^m(\rho, \varphi) = A \cdot R_n^m(\rho)\, \cos(m\,\varphi) \tag{26.1}$$

and the odd ones by

$$Z_n^{-m}(\rho, \varphi) = A \cdot R_n^m(\rho)\, \sin(m\,\varphi) \tag{26.2}$$

The radial polynomials $R_n^m(\rho)$ are defined to be

$$R_n^m(\rho) = \sum_{k=0}^{(n-m)/2} \frac{(-1)^k\,(n-k)!}{k!\,((n+m)/2-k)!\,((n-m)/2-k)!}\,\rho^{n-2\,k} \tag{26.3}$$

if $n-m$ is even and

$$R_n^m(\rho) = 0 \tag{26.4}$$

im $n-m$ is odd.

The prefactor A is calculated to fulfill the following normalization condition:

$$\int_0^{2\pi} \int_0^1 Z_j^2 \, \rho \, d\rho \, d\varphi = \pi \tag{26.5}$$

26.1.1 Zernike Polynomials

The Zernike Polynomial in Quadoa Optical CAD correspond to those defined in the ANSI Z80.28/ISO 24157 Standard. They are normed to fulfill the norming condtion in equation 26.5

The sorting of the Symmetric Zernike Polynomial is as shown in figure 26.1. The index of the Zernike coefficients in the Optical Design Tree Editor is counted row by row starting at the top with Z0 for piston from left to right in the figure. (See also [5])

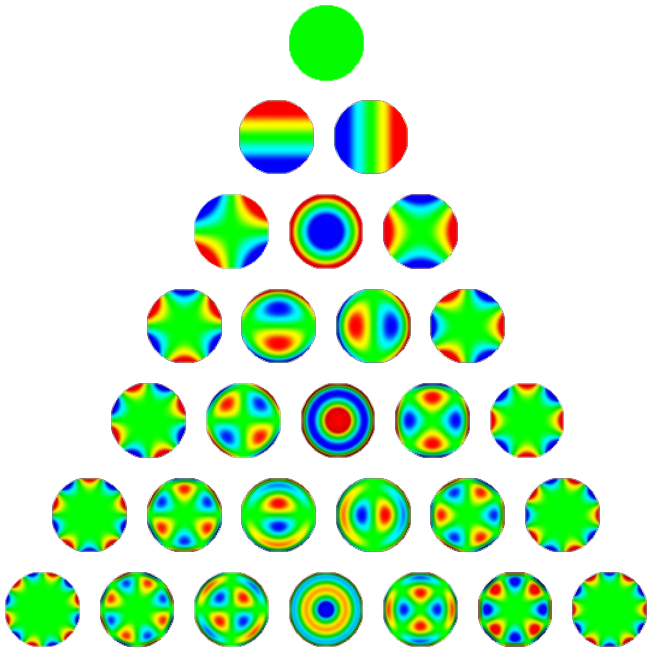


Figure 26.1: Zernike Polynomials with ANSI Z80.28/ISO 24157 sorting

The Formala to calculate the first 28 terms can be seen in table 26.1.

Zernike Coefficient	Formula
Z0	1
Z1	$2y$
Z2	$2x$
Z3	$\sqrt{6}(2xy)$
Z4	$\sqrt{3}(-1+2y^2+2x^2)$
Z5	$\sqrt{6}(-1y^2+1x^2)$
Z6	$\sqrt{8}(-1y^3+3x^2y)$
Z7	$\sqrt{8}(-2y+3y^3+3x^2y)$
Z8	$\sqrt{8}(-2x+3xy^2+3x^3)$
Z9	$\sqrt{8}(-3xy^2+1x^3)$
Z10	$\sqrt{10}(-4xy^3+4x^3y)$
Z11	$\sqrt{10}(-6xy+8xy^3+8x^3y)$
Z12	$\sqrt{5}(1-6y^2+6y^4-6x^2+12x^2y^2+6x^4)$
Z13	$\sqrt{10}(3y^2-4y^4-3x^2+4x^4)$
Z14	$\sqrt{10}(1y^4-6x^2y^2+1x^4)$
Z15	$\sqrt{12}(1y^5-10x^2y^3+5x^4y)$
Z16	$\sqrt{12}(4y^3-5y^5-12x^2y+10x^2y^3+15x^4y)$
Z17	$\sqrt{12}(3y-12y^3+10y^5-12x^2y+20x^2y^3+10x^4y)$
Z18	$\sqrt{12}(3x-12xy^2+10xy^4-12x^3+20x^3y^2+10x^5)$
Z19	$\sqrt{12}(12xy^2-15xy^4-4x^3-10x^3y^2+5x^5)$
Z20	$\sqrt{12}(5xy^4-10x^3y^2+1x^5)$
Z21	$\sqrt{14}(6xy^5-20x^3y^3+6x^5y)$
Z22	$\sqrt{14}(20xy^3-24xy^5-20x^3y+24x^5y)$
Z23	$\sqrt{14}(12xy-40xy^3+30xy^5-40x^3y+60x^3y^3+30x^5y)$
Z24	$\sqrt{7}(-1+12y^2-29y^4+20y^6+12x^2-59x^2y^2+59x^2y^4-29x^4+59x^4y^2+20x^6)$
Z25	$\sqrt{14}(-6y^2+20y^4-15y^6+6x^2-15x^2y^4-20x^4+15x^4y^2+15x^6)$
Z26	$\sqrt{14}(-5y^4+6y^6+30x^2y^2-30x^2y^4-5x^4-30x^4y^2+6x^6)$
Z27	$\sqrt{14}(-1y^6+15x^2y^4-15x^4y^2+1x^6)$

Table 26.1: Formula of Zernike Polynomials with ANSI Z80.28/ISO 24157 sorting

26.1.2 Zernike Fringe Polynomials

The Zernike Fringe Polynomials are only defined up to an order of 37. The mapping from $Z_n^m \rightarrow Z_j$ is defined by

$$j = \left(1 + \frac{n + |m|}{2}\right)^2 - 2|m| + \frac{1 - sgn(m)}{2} \tag{26.6}$$

This can also be seen in figure 26.2. Counting from top to bottom, and left to right and starting with Z1 (Piston) at the top.

The polynomials are not normed, so the absolute value of A is always 1.

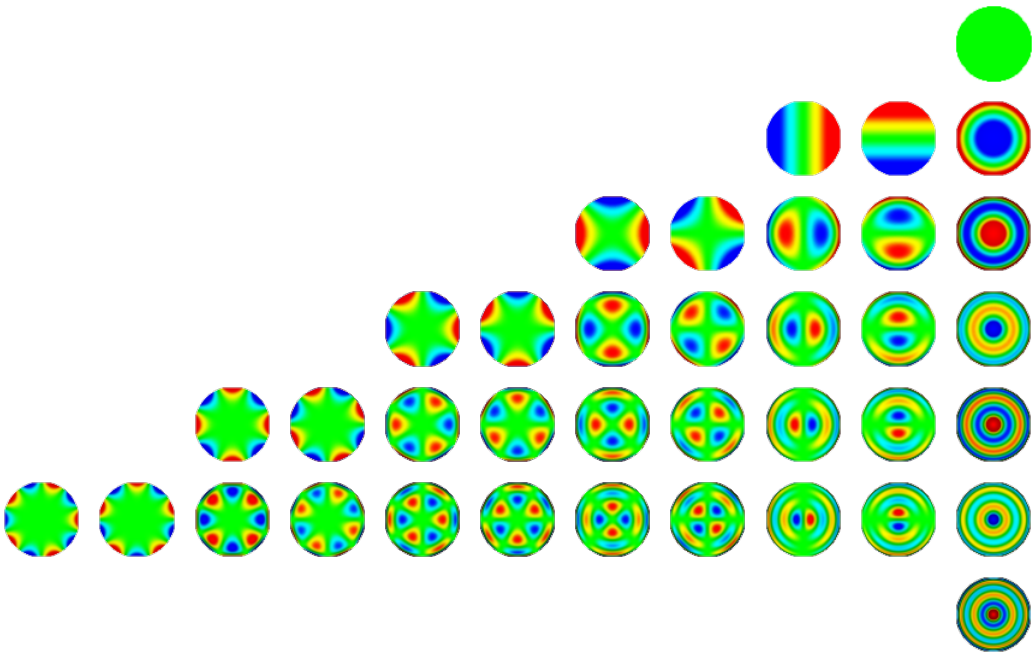


Figure 26.2: Zernike Polynomiald with low angular frequency first sorting

The formula for the Zernike Fringe Polynomials can be seen in table 26.2

Zern	Formula
Z1	1
Z2	$2x$
Z3	$2y$
Z4	$1(-1 + 2y^2 + 2x^2)$
Z5	$1(-1y^2 + 1x^2)$
Z6	$1(2xy)$
Z7	$1(-2x + 3xy^2 + 3x^3)$
Z8	$1(-2y + 3y^3 + 3x^2y)$
Z9	$1(1 - 6y^2 + 6y^4 - 6x^2 + 12x^2y^2 + 6x^4)$
Z10	$1(-3xy^2 + 1x^3)$
Z11	$1(-1y^3 + 3x^2y)$
Z12	$1(3y^2 - 4y^4 - 3x^2 + 4x^4)$
Z13	$1(-6xy + 8xy^3 + 8x^3y)$
Z14	$1(3x - 12xy^2 + 10xy^4 - 12x^3 + 20x^3y^2 + 10x^5)$
Z15	$1(3y - 12y^3 + 10y^5 - 12x^2y + 20x^2y^3 + 10x^4y)$
Z16	$1(-1 + 12y^2 - 30y^4 + 20y^6 + 12x^2 - 60x^2y^2 + 60x^2y^4 - 30x^4 + 60x^4y^2 + 20x^6)$
Z17	$1(1y^4 - 6x^2y^2 + 1x^4)$
Z18	$1(-4xy^3 + 4x^3y)$
Z19	$1(12xy^2 - 15xy^4 - 4x^3 - 10x^3y^2 + 5x^5)$
Z20	$1(4y^3 - 5y^5 - 12x^2y + 10x^2y^3 + 15x^4y)$
Z21	$1(-6y^2 + 20y^4 - 15y^6 + 6x^2 - 15x^2y^4 - 20x^4 + 15x^4y^2 + 15x^6)$
Z22	$1(12xy - 40xy^3 + 30xy^5 - 40x^3y + 60x^3y^3 + 30x^5y)$
Z23	$1(-4x + 30xy^2 - 60xy^4 + 35xy^6 + 30x^3 - 120x^3y^2 + 105x^3y^4 - 60x^5 + 105x^5y^2 + 35x^7)$
Z24	$1(-4y + 30y^3 - 60y^5 + 35y^7 + 30x^2y - 120x^2y^3 + 105x^2y^5 - 60x^4y + 105x^4y^3 + 35x^6y)$
Z25	$1(1 - 20y^2 + 90y^4 - 140y^6 + 70y^8 - 20x^2 + 180x^2y^2 - 420x^2y^4 + 280x^2y^6 + 90x^4 - 420x^4y^2 + 420x^4y^4 - 140x^6 + 280x^6y^2 + 70x^8)$
Z26	$1(5xy^4 - 10x^3y^2 + 1x^5)$
Z27	$1(1y^5 - 10x^2y^3 + 5x^4y)$
Z28	$1(-5y^4 + 6y^6 + 30x^2y^2 - 30x^2y^4 - 5x^4 - 30x^4y^2 + 6x^6)$
Z29	$1(20xy^3 - 24xy^5 - 20x^3y + 24x^5y)$
Z30	$1(-30xy^2 + 90xy^4 - 63xy^6 + 10x^3 + 60x^3y^2 - 105x^3y^4 - 30x^5 - 21x^5y^2 + 21x^7)$
Z31	$1(-10y^3 + 30y^5 - 21y^7 + 30x^2y - 60x^2y^3 + 21x^2y^5 - 90x^4y + 105x^4y^3 + 63x^6y)$
Z32	$1(10y^2 - 60y^4 + 105y^6 - 56y^8 - 10x^2 + 105x^2y^4 - 112x^2y^6 + 60x^4 - 105x^4y^2 - 105x^6 + 112x^6y^2 + 56x^8)$
Z33	$1(-20xy + 120xy^3 - 210xy^5 + 112xy^7 + 120x^3y - 420x^3y^3 + 336x^3y^5 - 210x^5y + 336x^5y^3 + 112x^7y)$
Z34	$1(5x - 60xy^2 + 210xy^4 - 280xy^6 + 126xy^8 - 60x^3 + 420x^3y^2 - 840x^3y^4 + 504x^3y^6 + 210x^5 - 840x^5y^2 + 756x^5y^4 - 280x^7 + 504x^7y^2 + 126x^9)$
Z35	$1(5y - 60y^3 + 210y^5 - 280y^7 + 126y^9 - 60x^2y + 420x^2y^3 - 840x^2y^5 + 504x^2y^7 + 210x^4y - 840x^4y^3 + 756x^4y^5 - 280x^6y + 504x^6y^3 + 126x^8y)$
Z36	$1(-1 + 30y^2 - 210y^4 + 560y^6 - 630y^8 + 252y^{10} + 30x^2 - 420x^2y^2 + 1680x^2y^4 - 2520x^2y^6 + 1260x^2y^8 - 210x^4 + 1680x^4y^2 - 3780x^4y^4 + 2520x^4y^6 + 560x^6 - 2520x^6y^2 + 2520x^6y^4 - 630x^8 + 1260x^8y^2 + 252x^{10})$
Z37	$1(1 - 42y^2 + 420y^4 - 1680y^6 + 3150y^8 - 2772y^{10} + 924y^{12} - 42x^2 + 840x^2y^2 - 5040x^2y^4 + 12600x^2y^6 - 13860x^2y^8 + 5544x^2y^{10} + 420x^4 - 5040x^4y^2 + 18900x^4y^4 - 27720x^4y^6 + 13860x^4y^8 - 1680x^6 + 12600x^6y^2 - 27720x^6y^4 + 18480x^6y^6 + 3150x^8 - 13860x^8y^2 + 13860x^8y^4 - 2772x^{10} + 5544x^{10}y^2 + 924x^{12})$

Table 26.2: Formula of Zernike Fringe Polynomials with

26.2 XYZ-Polynomial

Thy XYZ-Polynomial is a simple power series over x and y. The Name of the coefficient e.g. $X3\,Y2$ means that the formula for this coefficient is $z = x^3 \cdot y^2$. The order of the polynomial means the maximum power that is present in the series. The x, y coordiante for the polynomial evaluation is always normed to the norm radius r_n. Figure 26.3 shows a XYZ-Polynomial of power 4. The sorting in the Optical Design Tree is row by row from top to bottom and left to right.

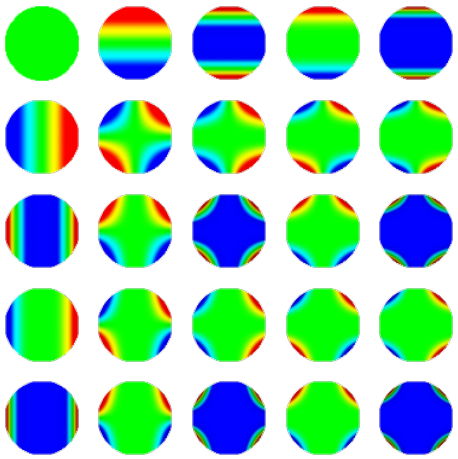


Figure 26.3: XYZ-Polynomial of 4th Order

26.3 Seidel Aberrations

27 Fiber Coupling

The calculations for the single mode fiber coupling efficiency relies on the mode matching technique. With F being the complex amplitude of the fiber mode and U being the complex amplitude of the beam the coupling efficiency T is calculated by

$$t = \frac{\int\int F(x,y)U'(x,y)dxdy}{\sqrt{\int\int F(x,y)F'(x,y)dxdy \int\int U(x,y)U'(x,y)dxdy}} \tag{27.1}$$

$$T = tt' \tag{27.2}$$

where U', F' and t' are the complex conjugates of the complex amplitudes. In theory the exact location where the mode matching integral is evaluated does not have any effect on the result. It could be on the fiber end as well as in the exit pupil of the system or anywhere else. However due to numerical issues sometimes a certain location has an advantage over the other. Therefore in Quadoa two options are available:

- *In the Exit Pupil:* When evaluated in the exit pupil the rays from the source are traced to the exit pupil and the Gaussian mode of the fiber is propagated backwards to the pupil surface where the modes overlap. This method is faster and in most cases leads to the same results.
- *On the Fiber End:* When evaluated on the fiber end the field is propagated up to the fiber surface, Depending on the method either by *Huygens PSF* or *Beamlet Propagation PSF* and then directly matched with the fiber mode. This method is slower but may in some cases be more reliable than the first method.

Note that the overlap integral method is extremely sensitive to slight changes in the sampling settings. To make sure the sampling is sufficient, it is therefore recommended to check the settings with the Huygens PSF or Beamlet Propagation PSF. Furthermore, the efficiency is extremely sensitive to small changes in the setup. Therefore - unless the system tolerances and alignment can be controlled extremely well - the computed value for the coupling efficiency should be viewed as the best possible limit rather than a truly achievable efficiency for most systems.

Bibliography

[1] Colbourne, Paul D.: Generally Astigmatic Gaussian Beam Representation and Optimization Using Skew Rays. In: *Classical Optics 2014*, Optica Publishing Group, 2014, IW3A.4

[2] Forbes, G. W.: Shape specification for axially symmetric optical surfaces. In: *Opt. Express* 15 (2007), Apr, Nr. 8, 5218–5226. http://dx.doi.org/10.1364/OE.15.005218. – DOI 10.1364/OE.15.005218

[3] G. W. Forbes, A. F. J. Ruoff R. J. Ruoff ; Kerwien, N.: Ray selection for optimization of rotationally symmetric systems. In: *Advanced Optical Technologies* 5 (2016), S. 237–247

[4] Goodman, J. W.: *Introduction to Fourier Optics*. 3rd. MaGraw-Hill, 2005

[5] Noll, R. J.: Zernike polynomials and atmospheric turbulence. In: *JOSA* 66 (1976), S. 207–211