VirtualLab Fusion Applications, Technology & Workflows

# Getting Started with VirtualLab Fusion

LightTrans International UG
Presenter: Olga Baladron-Zorita

# Links of Interest

- LightTrans website: www.LightTrans.com
- Find a VirtualLab Fusion distributor in your region: www.LightTrans.com/company/distributors
- You have further questions? Drop us a line at info@LightTrans.com
- Subscribe to our newsletter: www.LightTrans.com/newsletter
- Connect with us on the following social networks:
  - LinkedIn (www.linkedin.com/company/lighttrans)
  - Twitter (www.twitter.com/LightTrans)
  - YouTube (www.youtube.com/LightTransInternational)
- Check out our downloads page to see VirtualLab in action across a broad range of fields of application: www.LightTrans.com/resources/downloads
- Our webinars: www.LightTrans.com/products-services/learning/webinars
- Want to give VirtualLab Fusion a test drive? Request a trial version: www.LightTrans.com/resources/trial-software
- Interested in purchasing VirtualLab Fusion? Check out our products, licence model and learn more about additional evaluation possibilities: www.LightTrans.com/products-services/virtuallab-fusion/editions-toolboxes

# Quick Index:

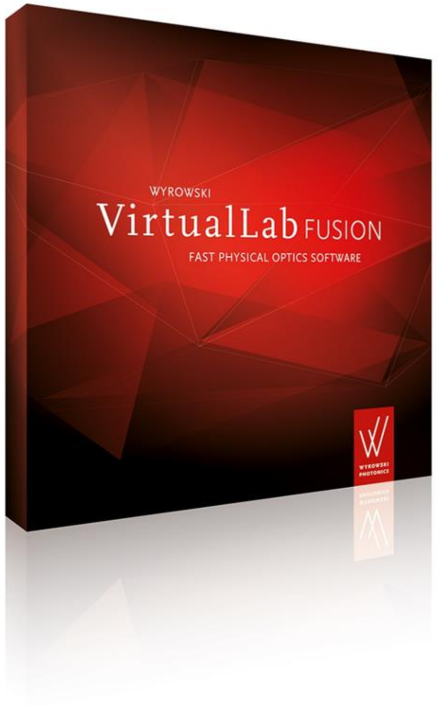
# Introduction to the Teams

# Who We Are

LightTrans

Founded 1999

General Distributor of the
Fast Physical Optics Software
**VirtualLab Fusion**

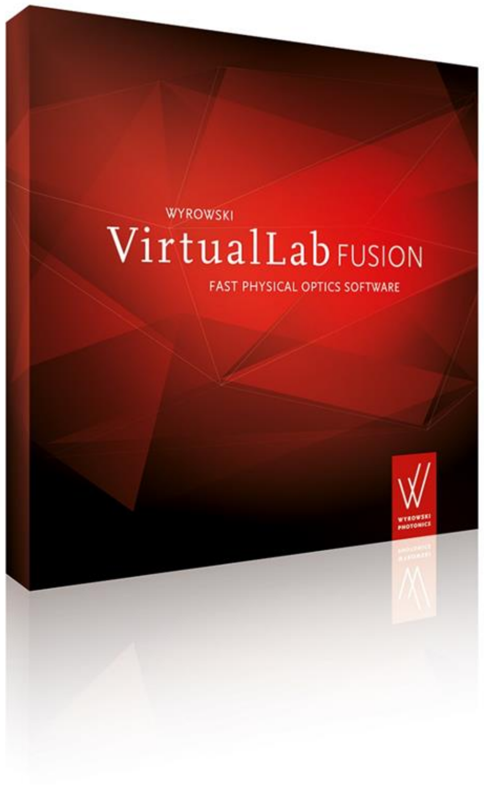



WYROWSKI PHOTONICS

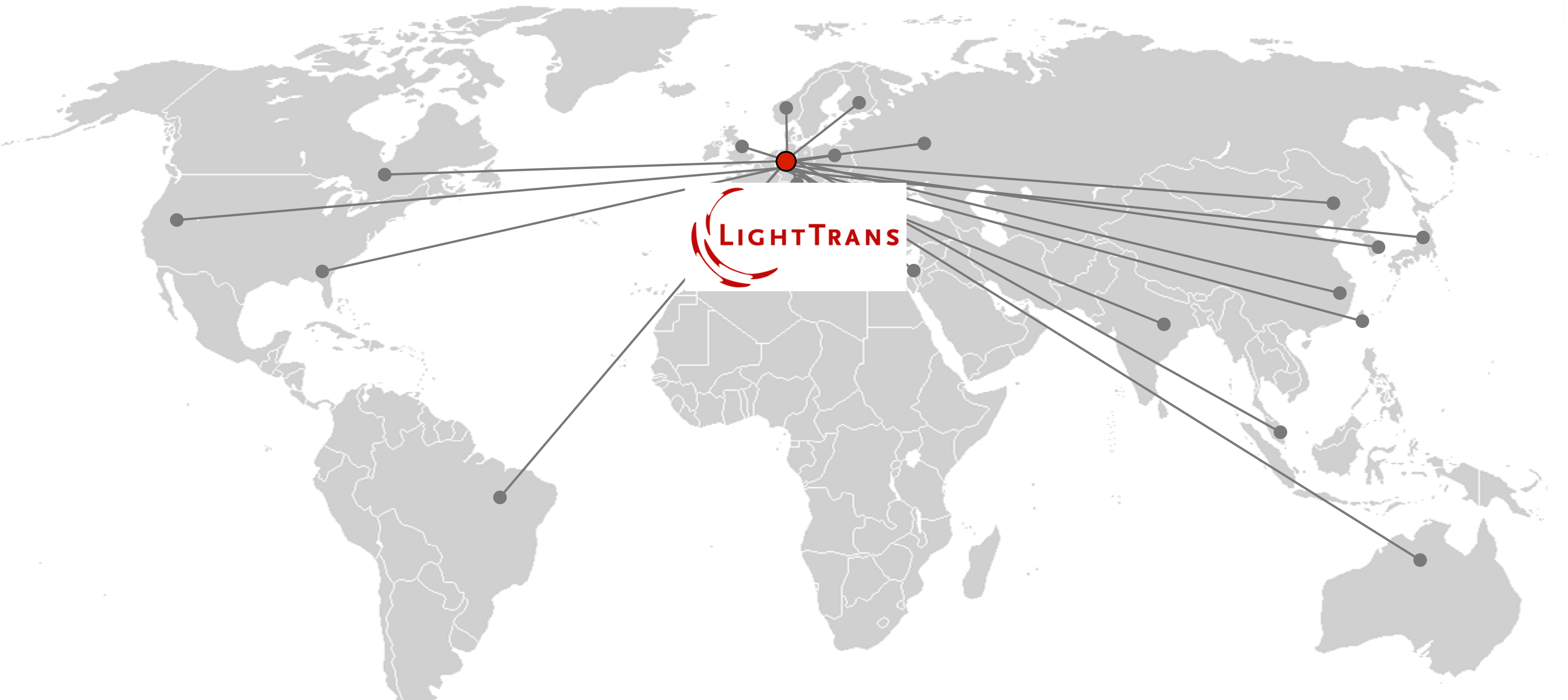
Founded 2014

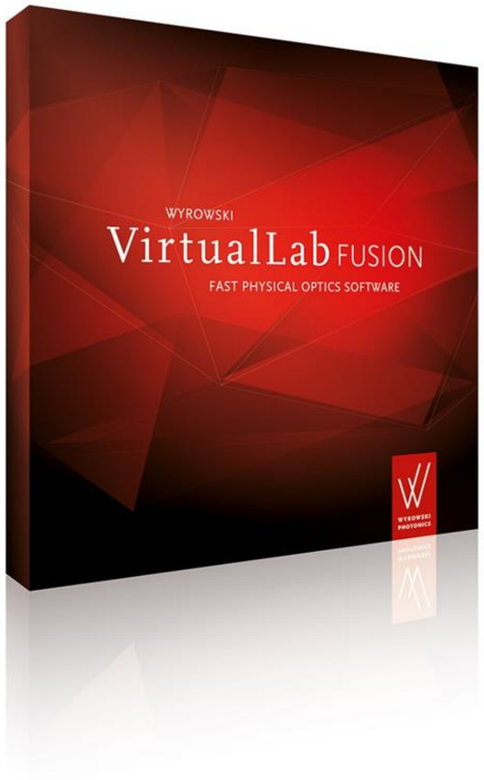
Developer of the
Fast Physical Optics Software
**VirtualLab Fusion**

# Fast Physical Optics Modeling and Design Software

# Fast Physical Optics Modeling and Design Software




LUEN SOFT

WAVE OPTO

T.E.M. Incorporated

INFOTEK

Part 1

# Light as an Electromagnetic Field

# The Objective Behind VirtualLab Fusion

To perform physical optics simulations of arbitrary optical systems

# The Objective Behind VirtualLab Fusion

To perform **physical optics simulations** of arbitrary optical systems

# The Objective Behind VirtualLab Fusion

To perform **physical optics simulations** of arbitrary optical systems

Finding the expression of the **six-dimensional vector field that solves Maxwell's equations** under the conditions imposed by the system in question

# The Objective Behind VirtualLab Fusion

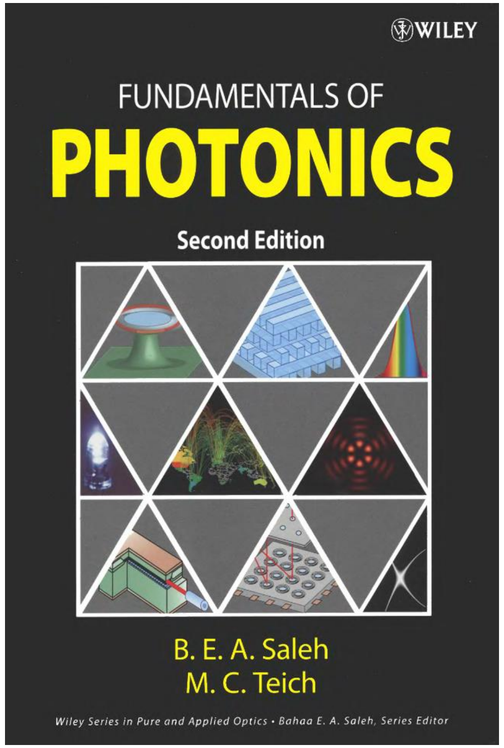
To perform **physical optics simulations** of arbitrary optical systems

Finding the expression of the **electromagnetic field** for the system in question

# The Objective Behind VirtualLab Fusion

To perform **physical optics simulations** of arbitrary optical systems

Finding the expression of the **electromagnetic field** for the system in question



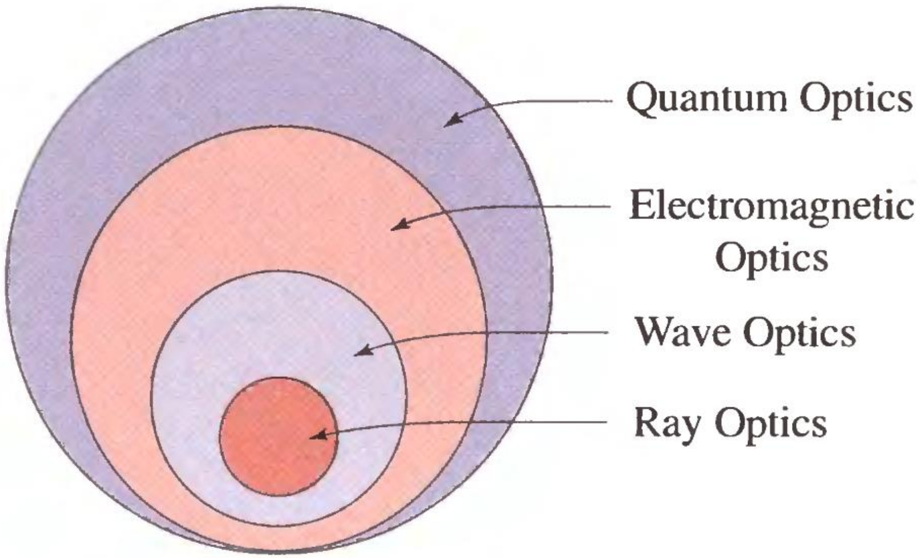
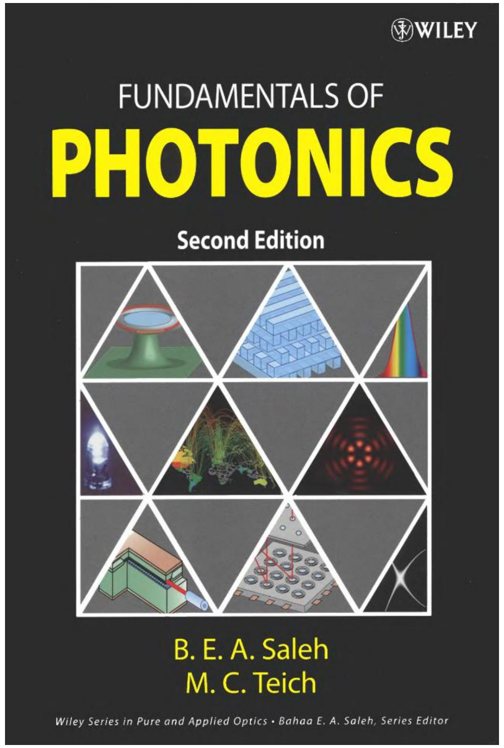



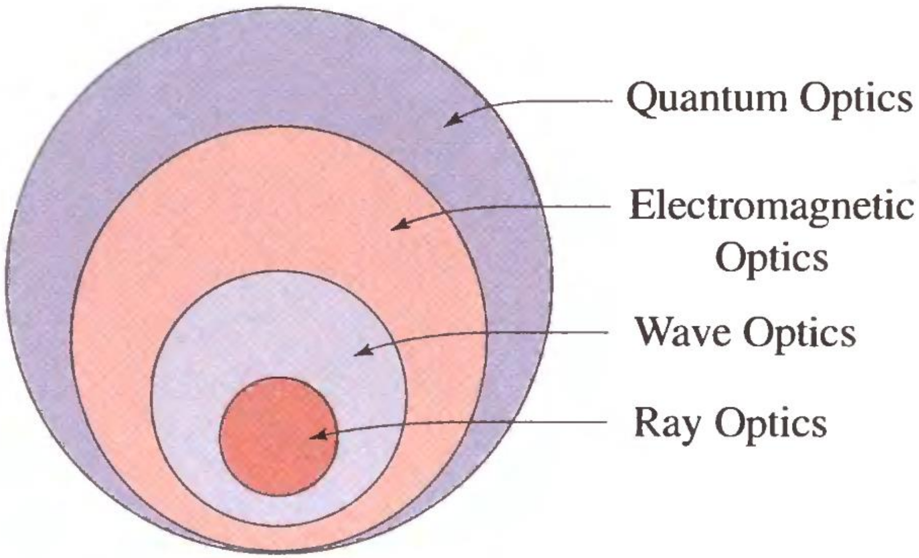

Image reference: “Fundamentals of Photonics”, B. E. A. Saleh and M. C. Teich, 2ndEdition, Wiley

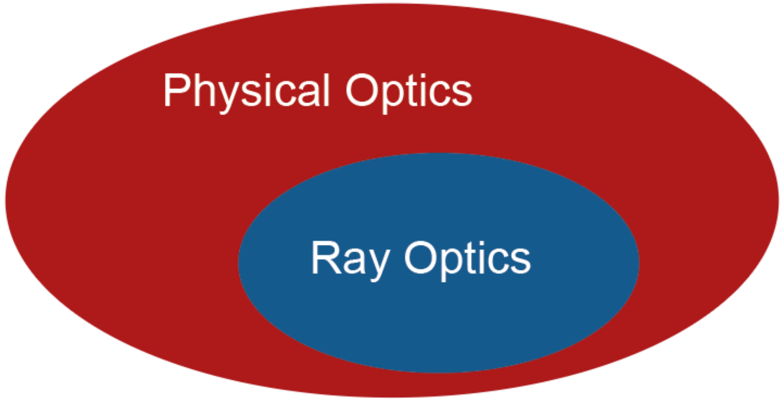
# The Objective Behind VirtualLab Fusion

To perform **physical optics simulations** of arbitrary optical systems

Finding the expression of the **electromagnetic field** for the system in question



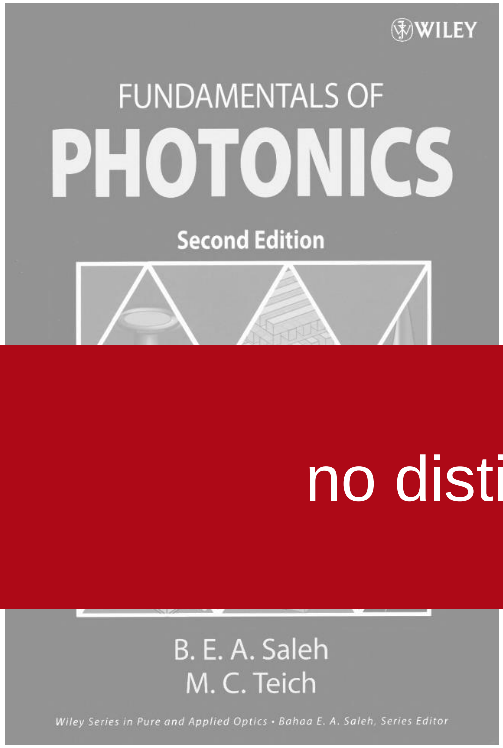

… in VirtualLab Fusion

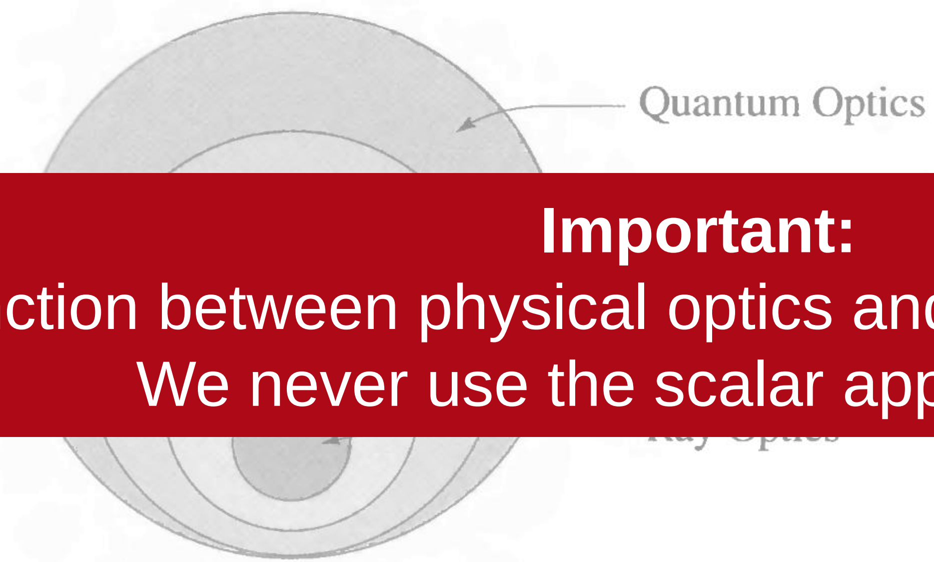
Image reference: “Fundamentals of Photonics”, B. E. A. Saleh and M. C. Teich, 2ndEdition, Wiley

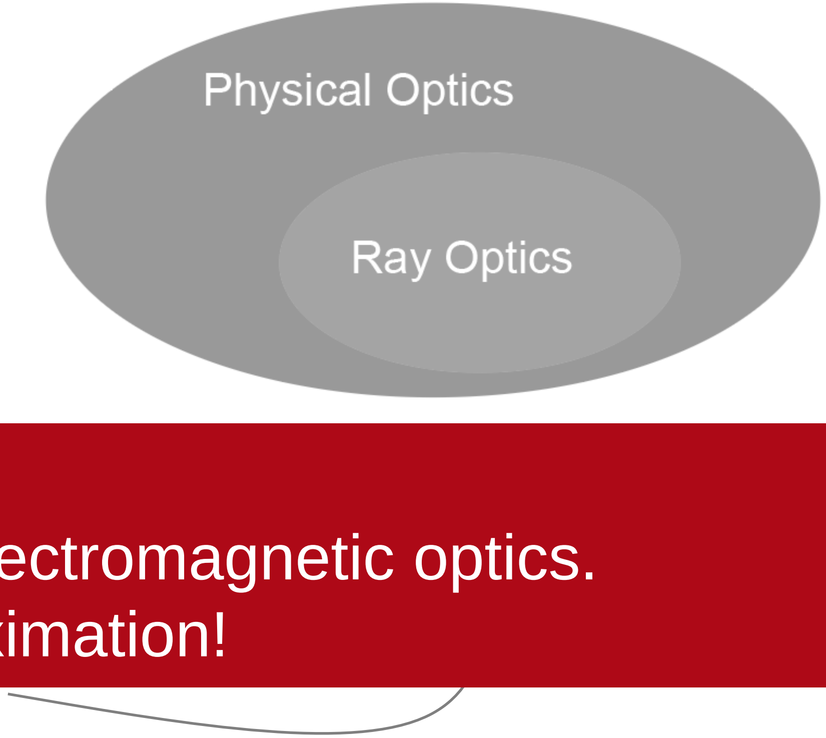
# The Objective Behind VirtualLab Fusion

To perform **physical optics simulations** of arbitrary optical systems

Finding the expression of the **electromagnetic field** for the system in question



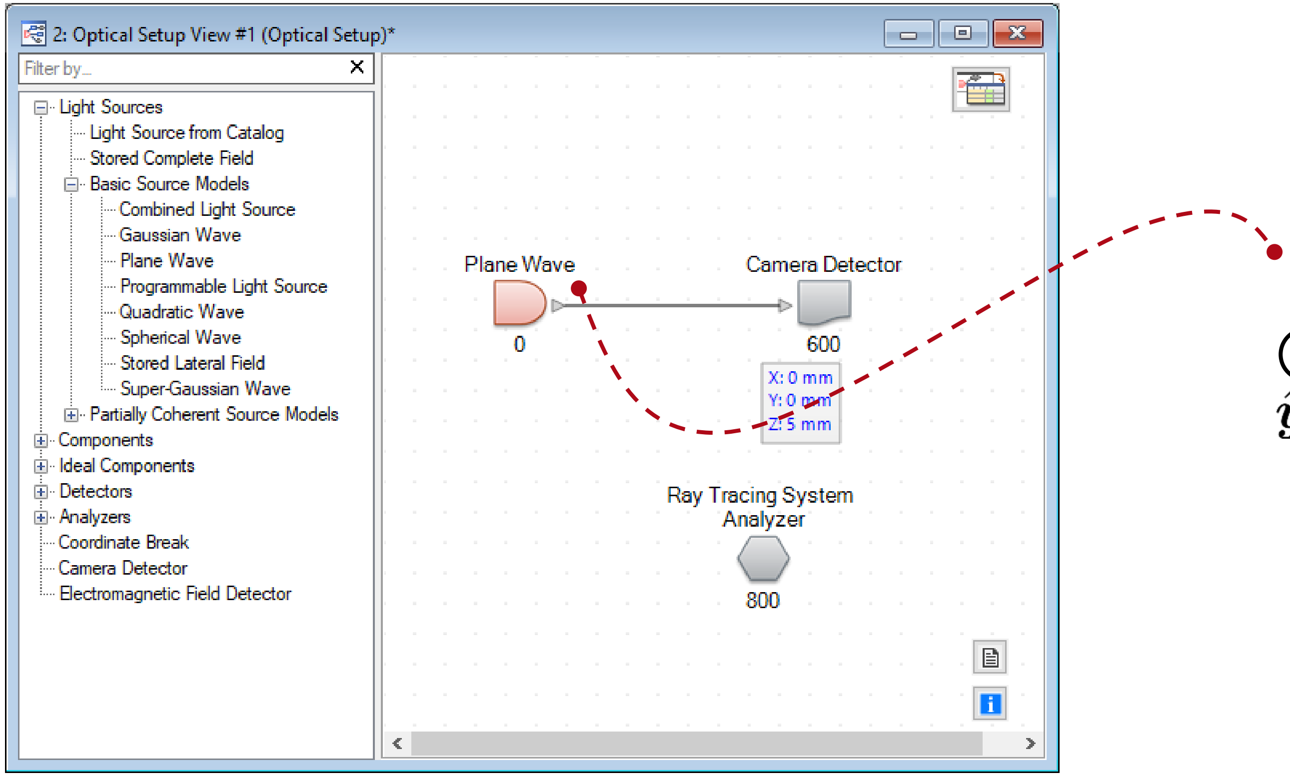



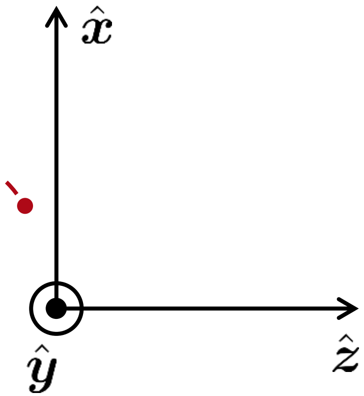

**Important:**
no distinction between physical optics and electromagnetic optics.
We never use the scalar approximation!

… in VirtualLab Fusion

Image reference: “Fundamentals of Photonics”, B. E. A. Saleh and M. C. Teich, 2ndEdition, Wiley

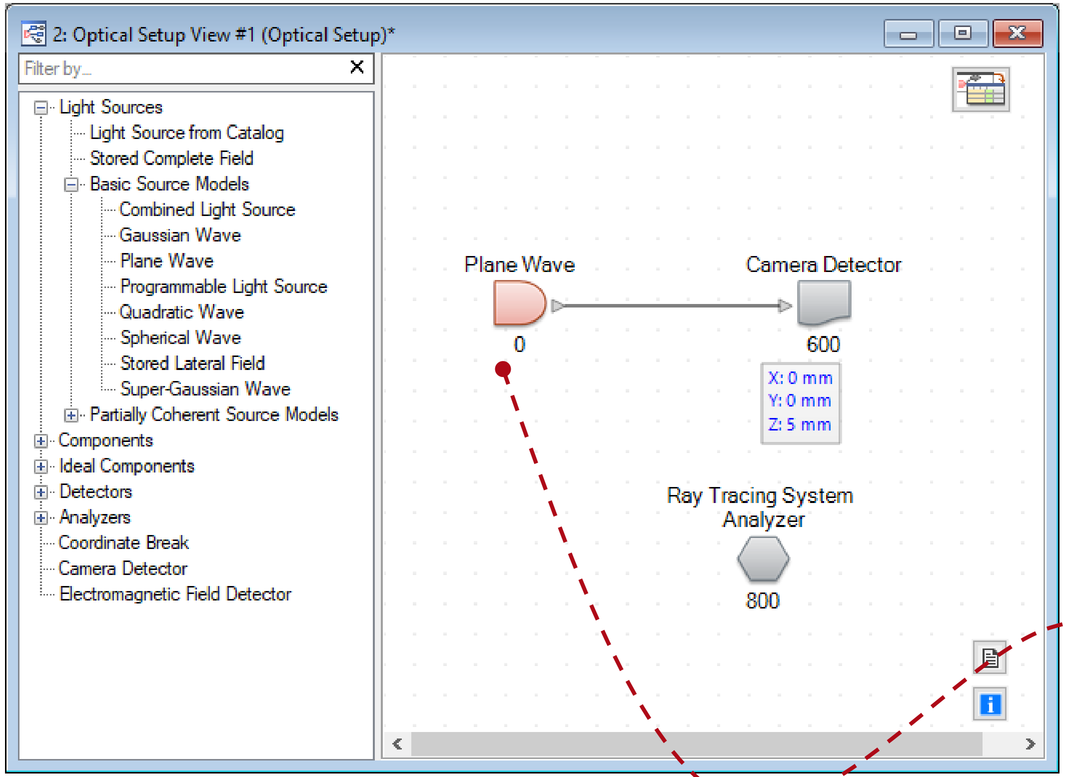
# How Is Light Generated in the System? (Basics)

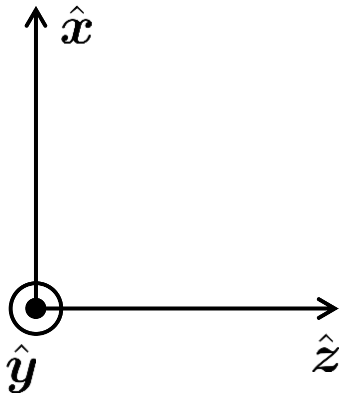



In VirtualLab Fusion, the source establishes the **global coordinate system** of the optical setup

# How Is Light Generated in the System? (Basics)



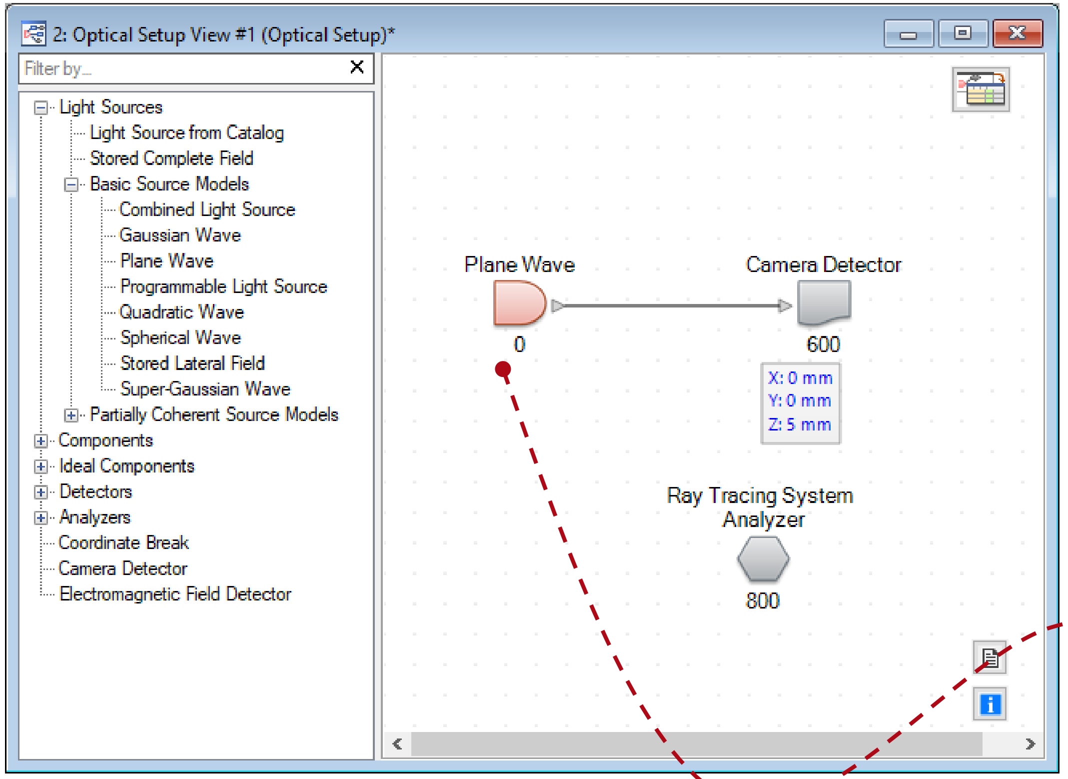



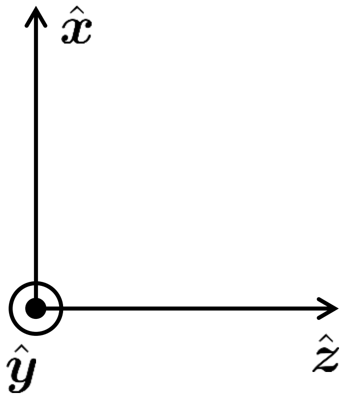

In VirtualLab Fusion, the source establishes the **global coordinate system** of the optical setup

$$\begin{cases} E_x(x,y) \\ E_y(x,y) \end{cases}$$

The source needs to define, per wavelength, the **function that spatially describes the x and y components** of the electric field at the input plane

# How Is Light Generated in the System? (Basics)



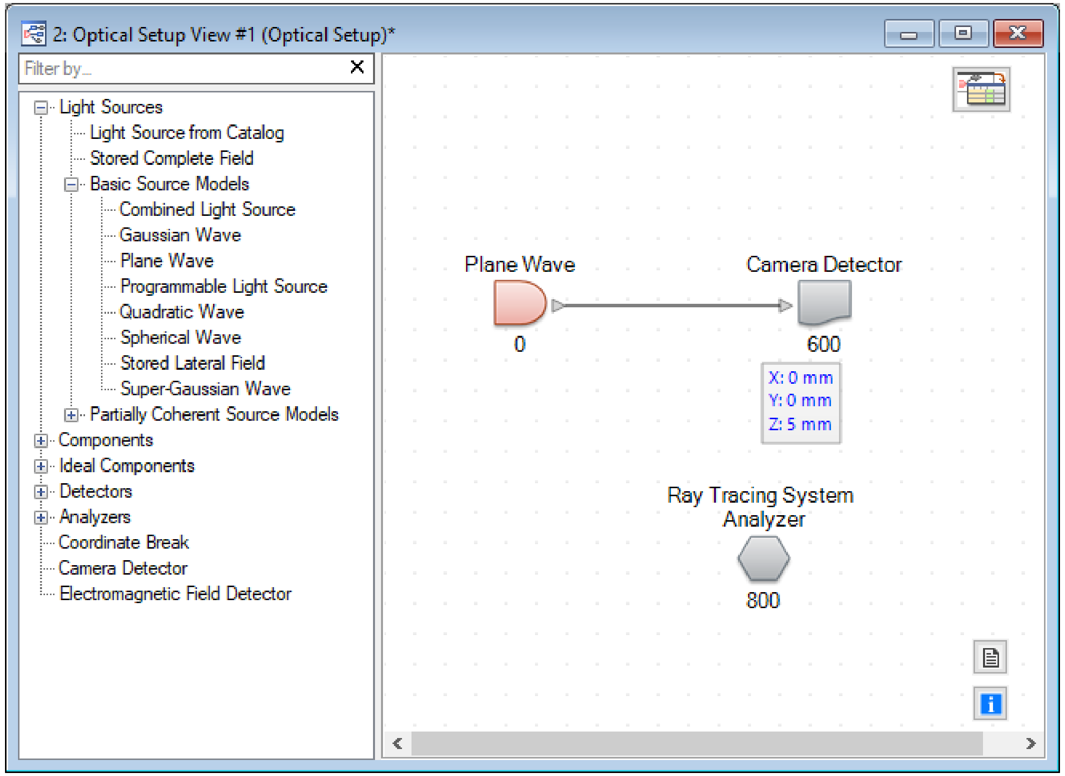

In VirtualLab Fusion, the source establishes the **global coordinate system** of the optical setup

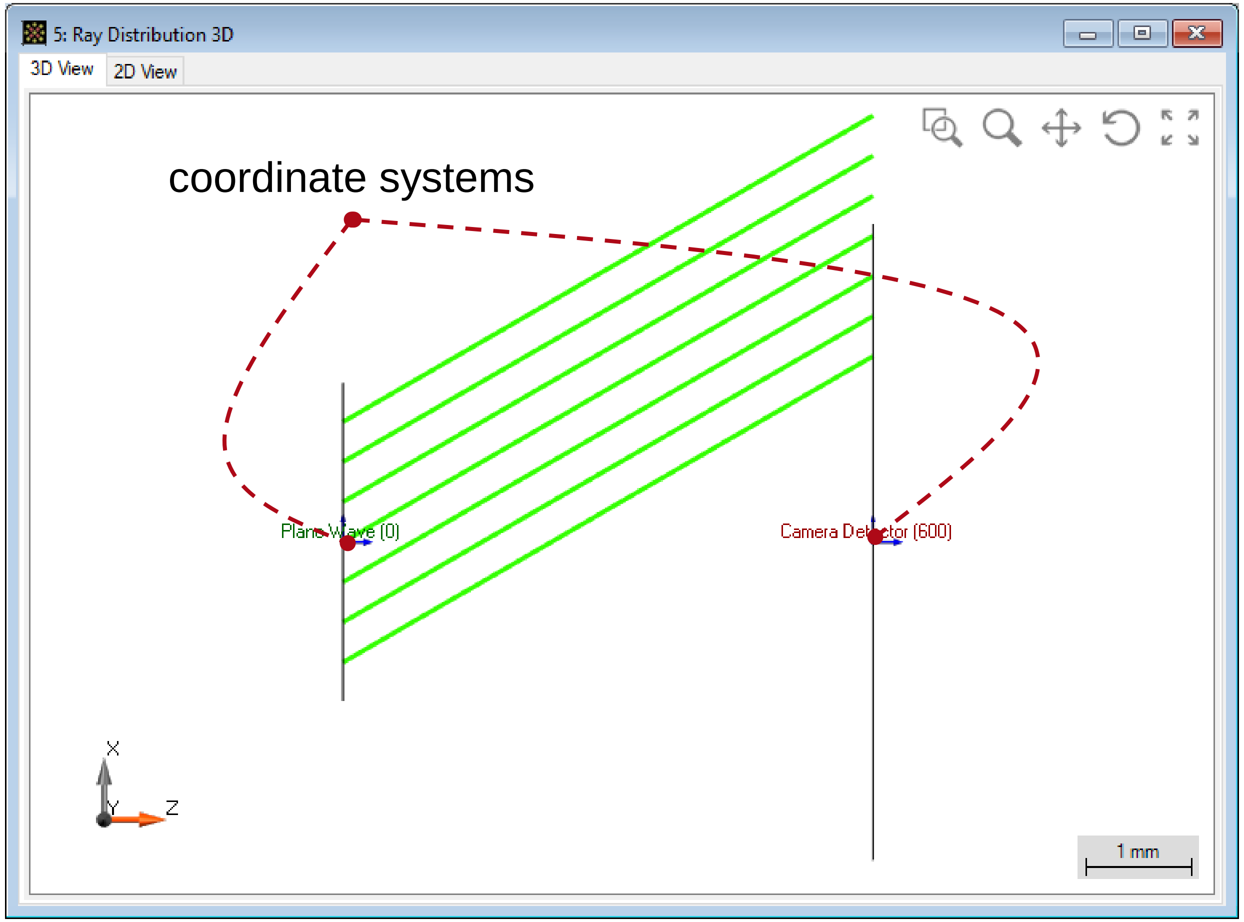
$$\begin{cases} \boldsymbol{E}_x(x,y) \\ \boldsymbol{E}_y(x,y) \end{cases}$$

The source needs to define, per wavelength, the **function that spatially describes the x and y components** of the electric field at the input plane

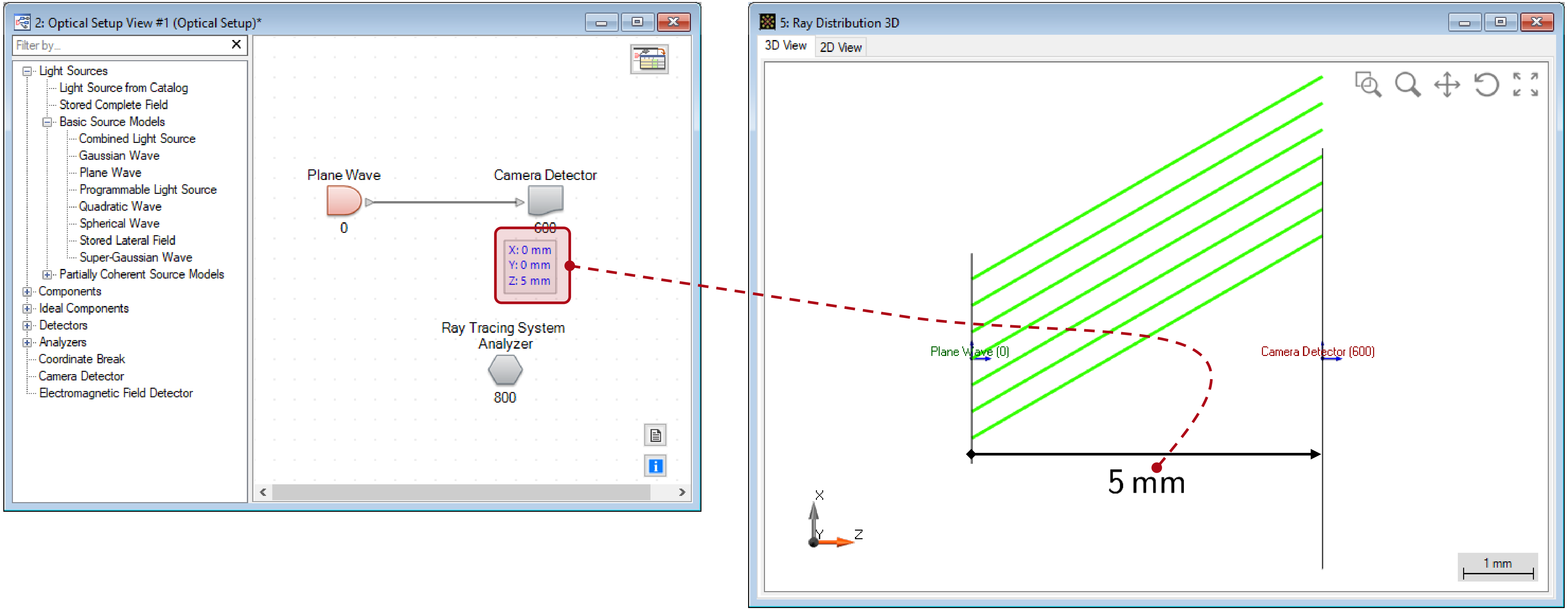
According to Maxwell's equations, **only two of the six electromagnetic components are independent** in a homogeneous, isotropic medium ➔ we arbitrarily select Ex and Ey; other components calculated on demand

# How Is Light Generated in the System? (Basics)

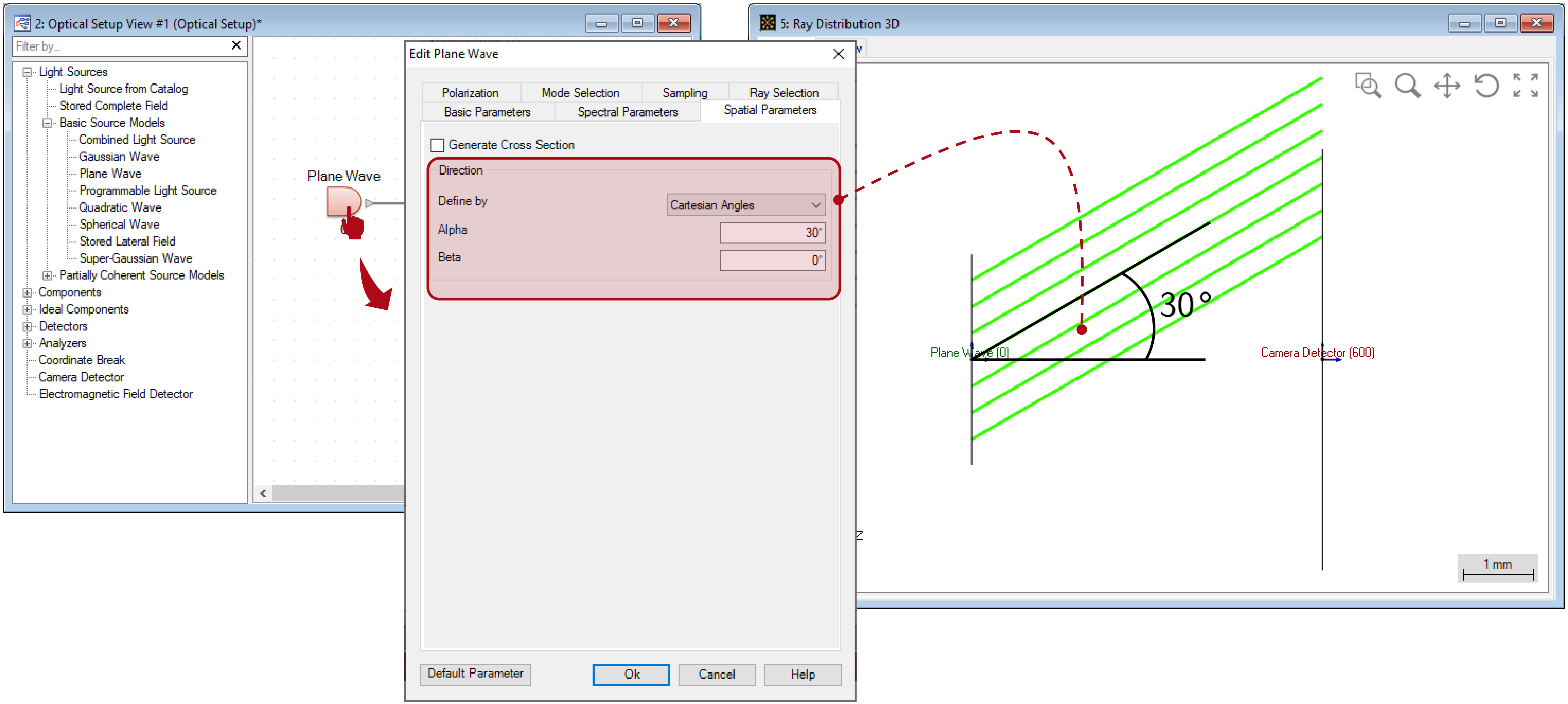


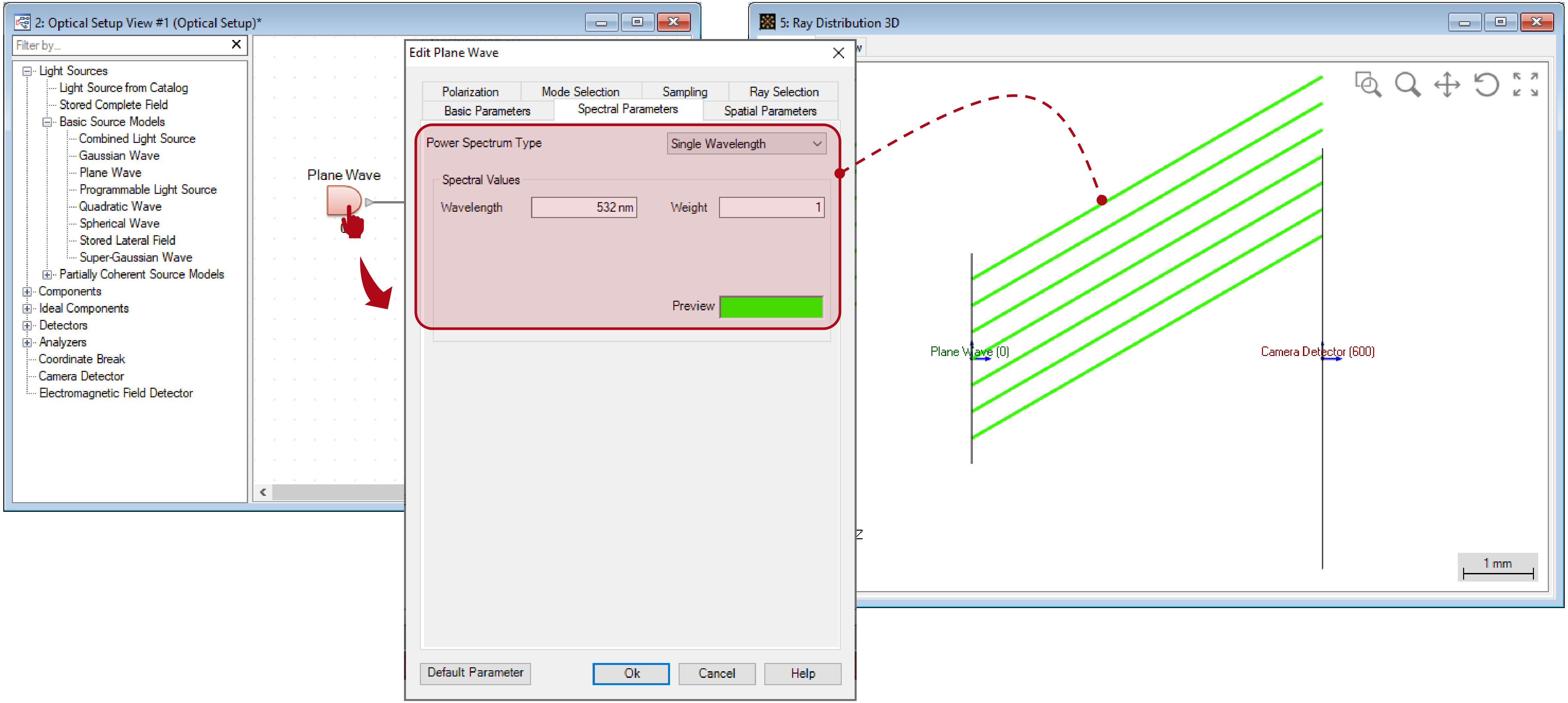
# How Is Light Generated in the System? (Basics)



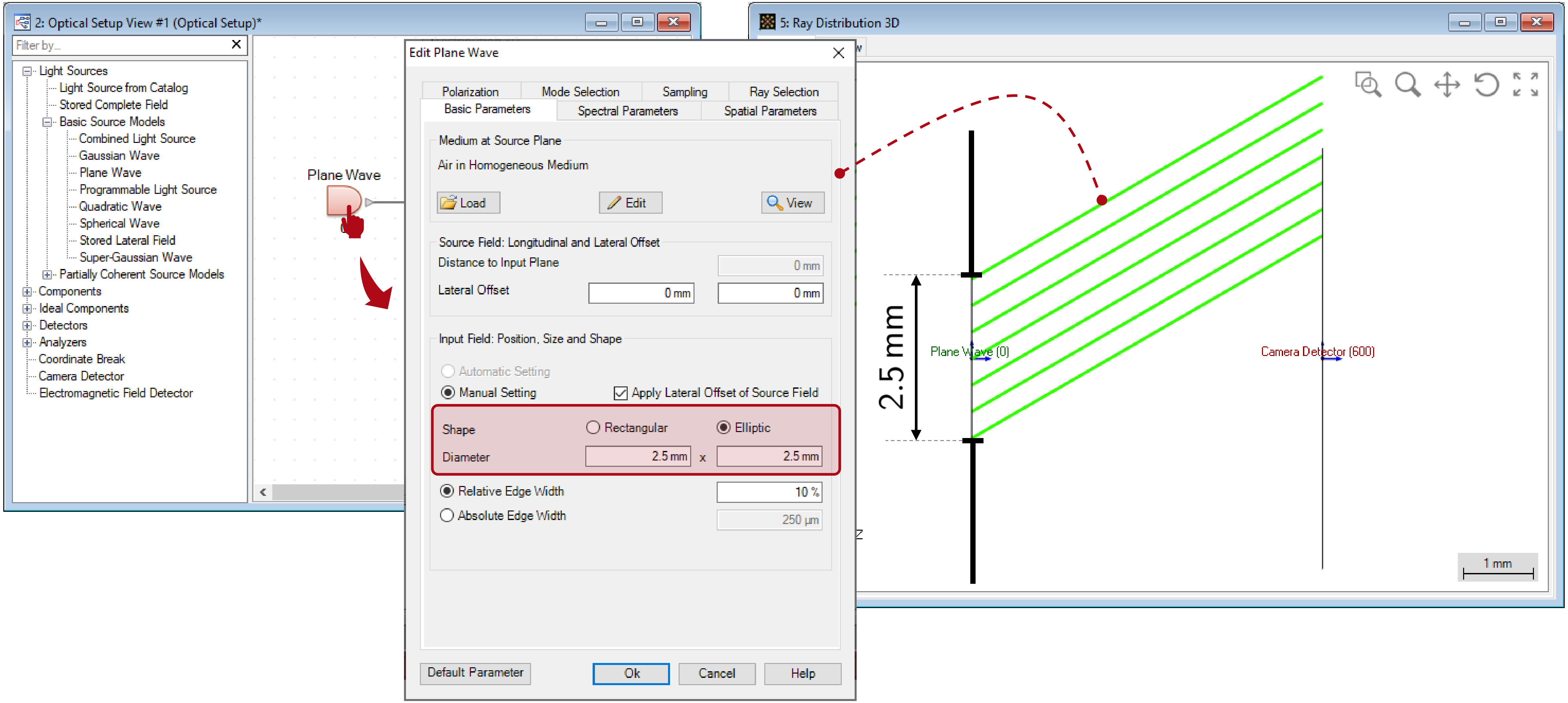
# How Is Light Generated in the System? (Basics)



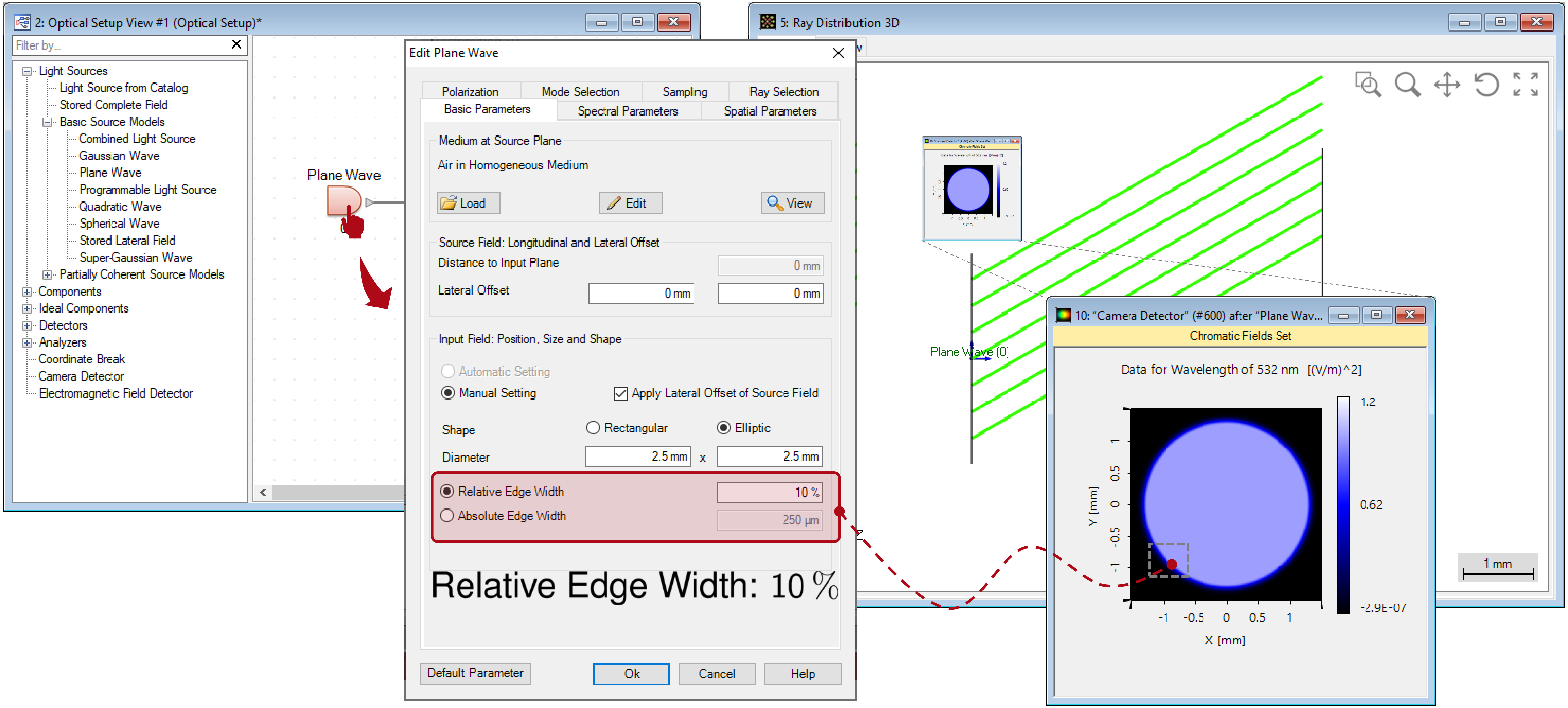
# How Is Light Generated in the System? (Basics)



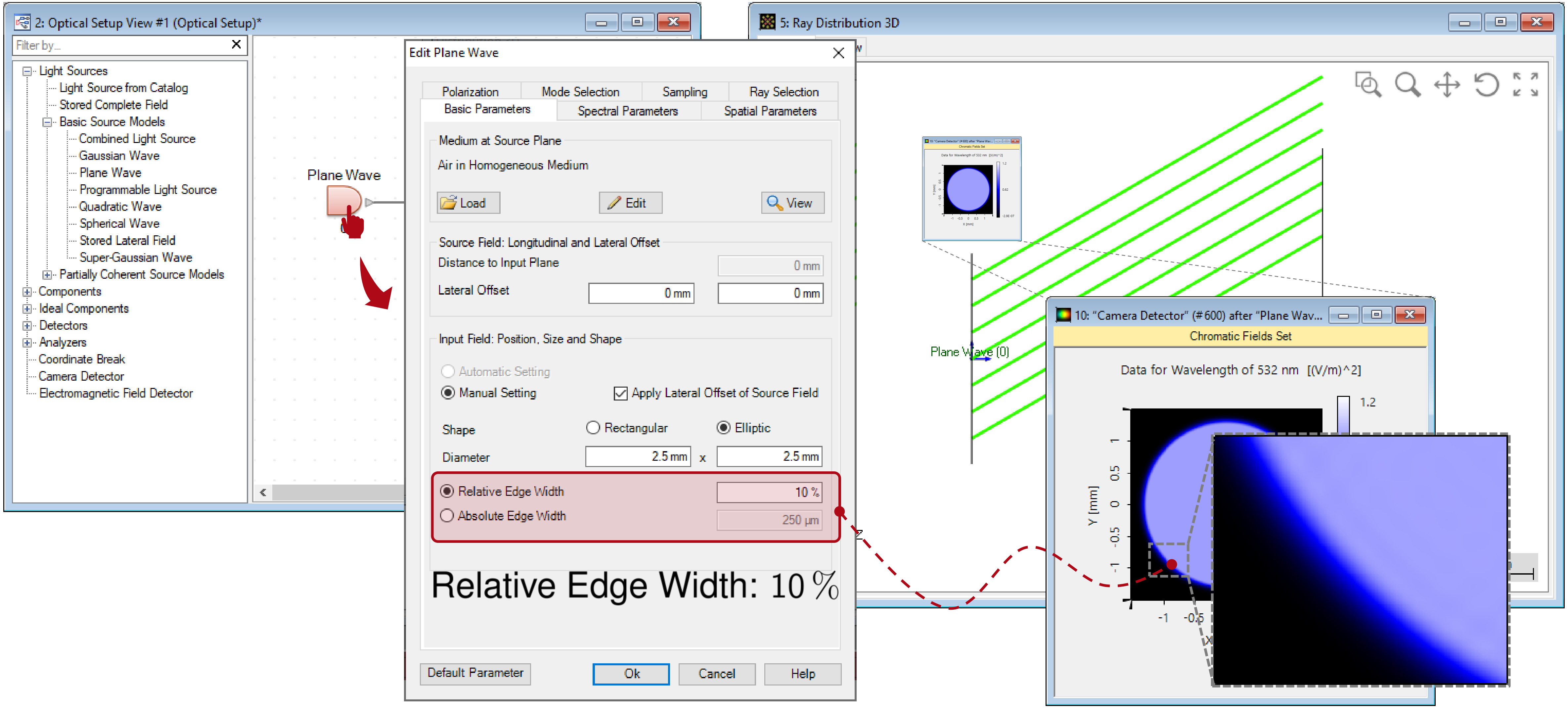
# How Is Light Generated in the System? (Basics)



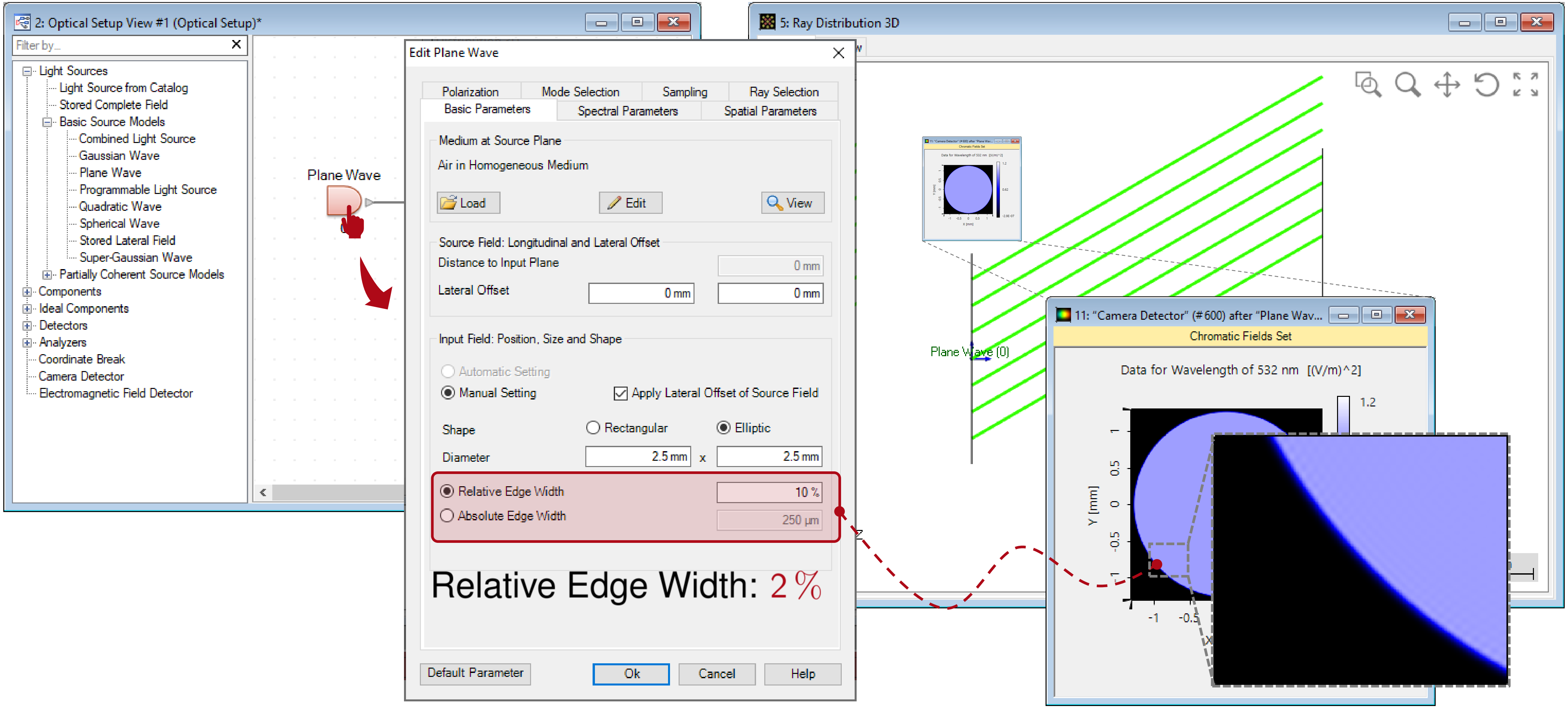
# How Is Light Generated in the System? (Basics)



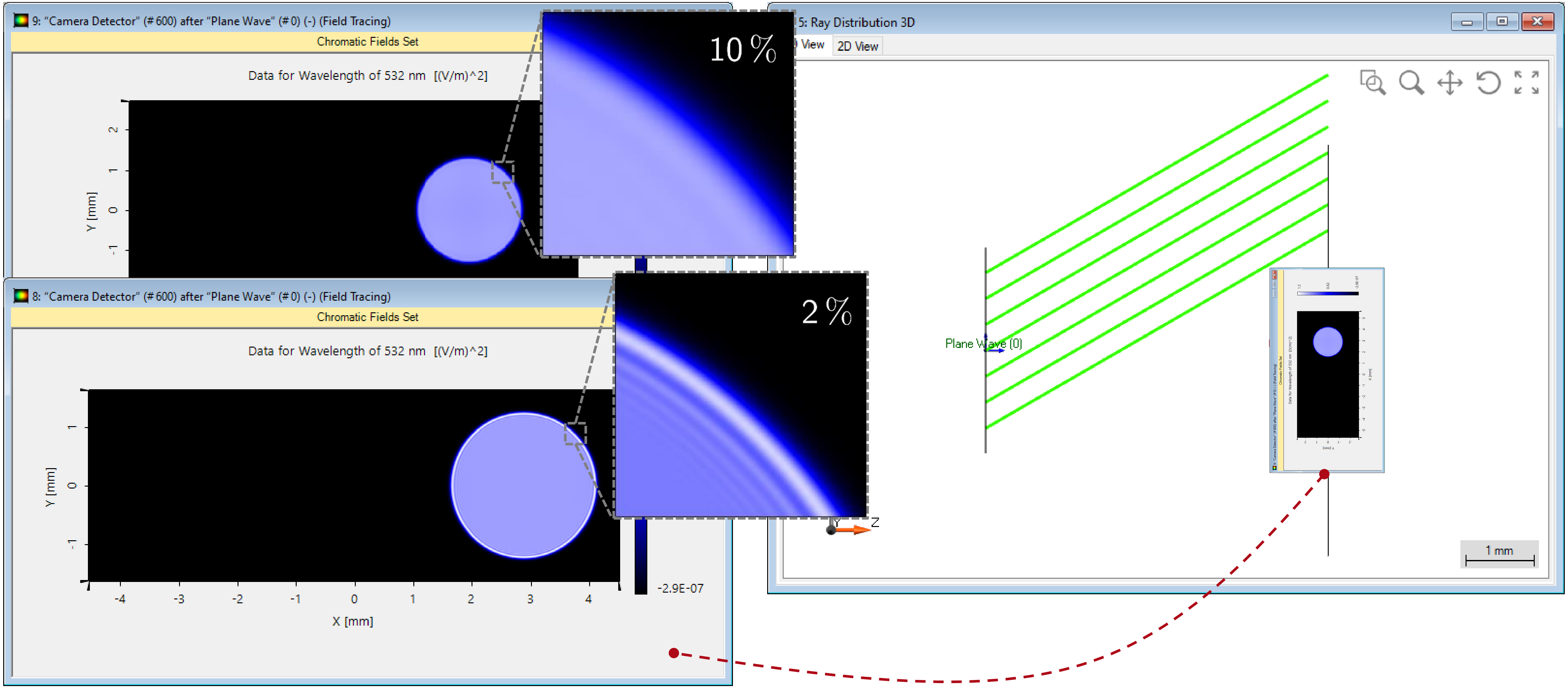
# How Is Light Generated in the System? (Basics)



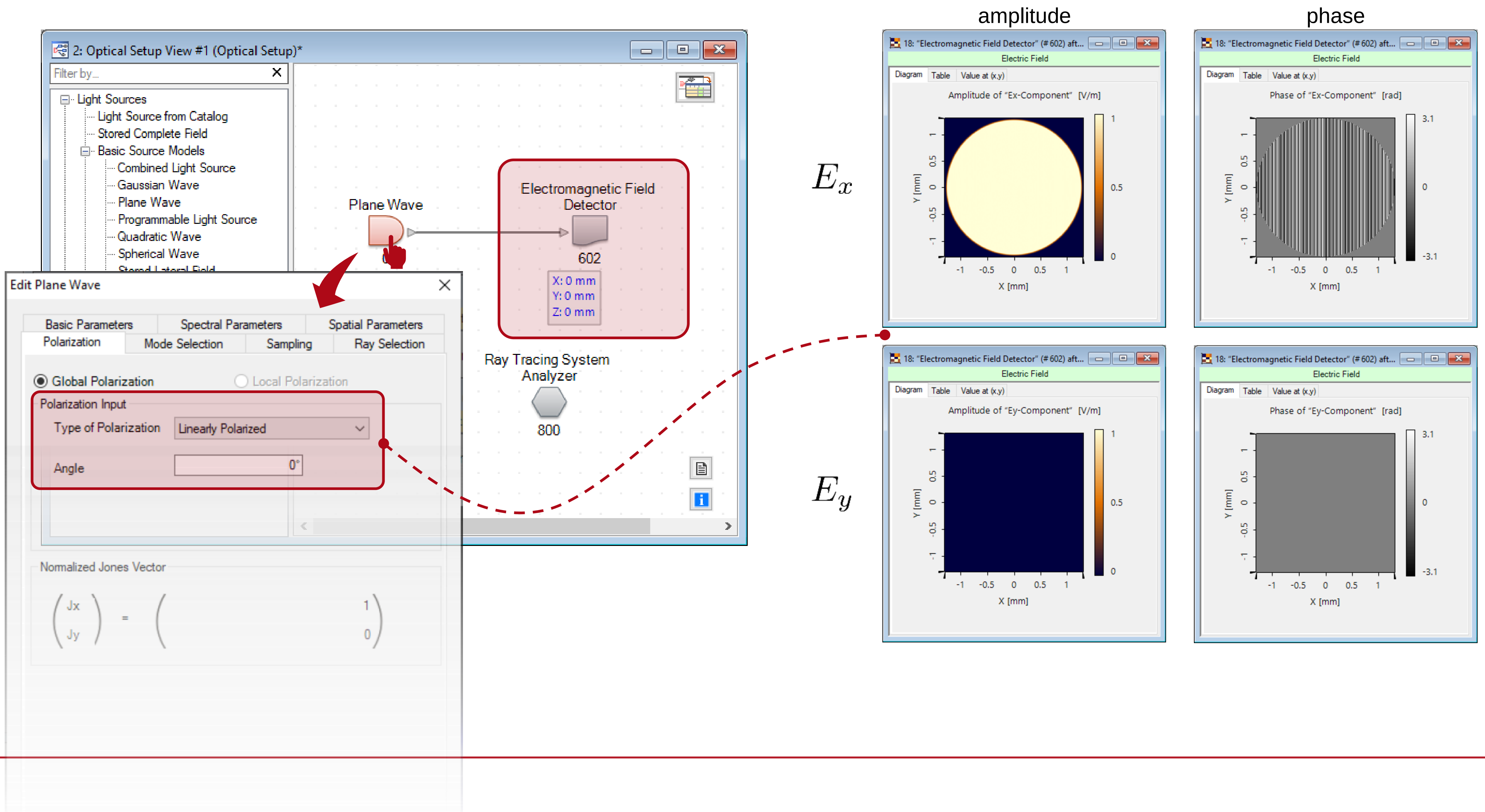
# How Is Light Generated in the System? (Basics)



# How Is Light Generated in the System? (Basics)



# How Is Light Generated in the System? (Basics)

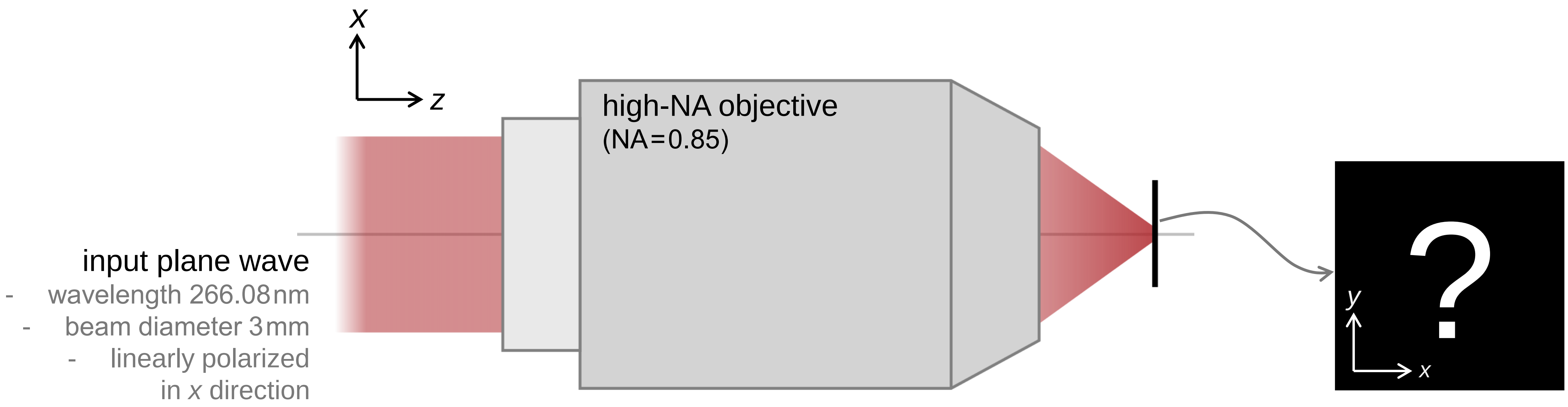


Example 1

# Analyzing High-NA Objective Lens Focusing

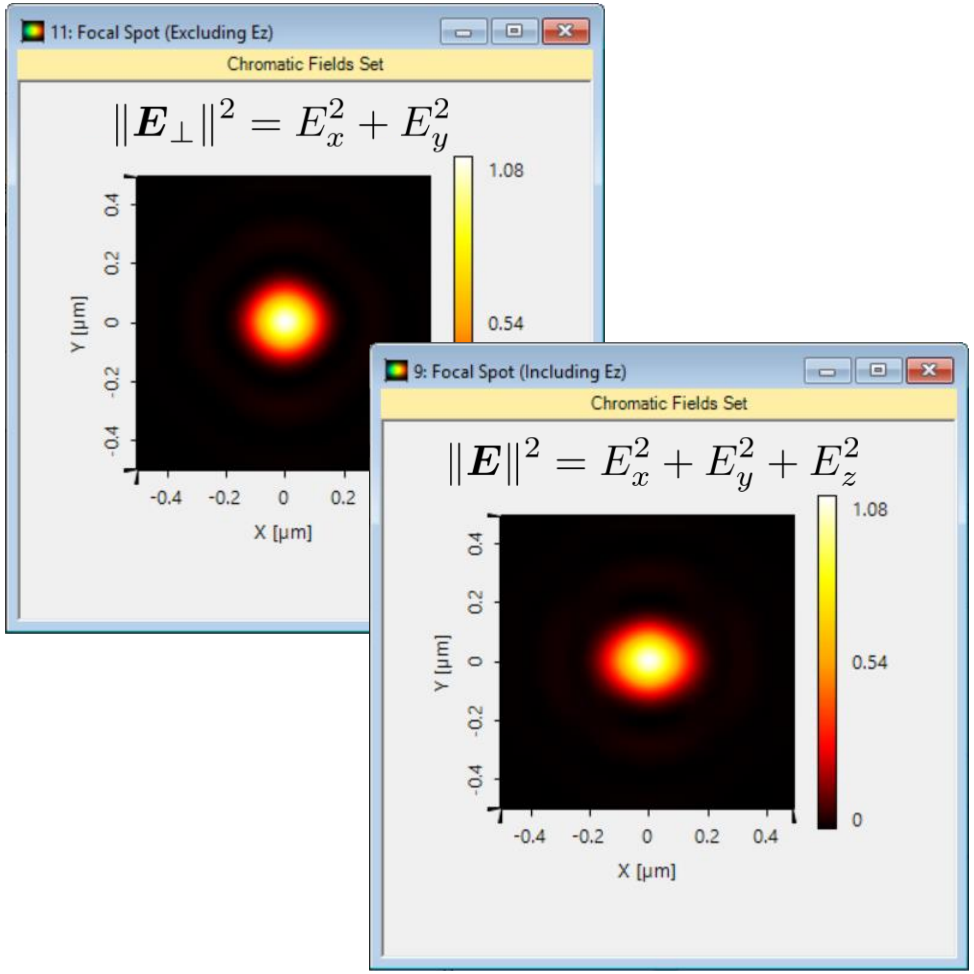
see the full Application Use Case

# Modeling Task



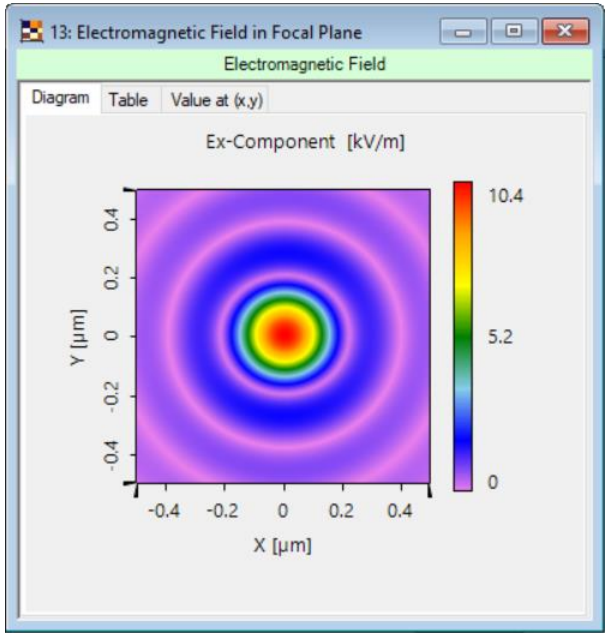
# Field Tracing Results (Camera Detector)

- The top figure shows the intensity by integrating $E_x$ and $E_y$ field components only.
- The bottom figure shows the intensity by integrating $E_x$, $E_y$ and $E_z$ components: an obvious asymmetry is seen due to the relatively large $E_z$ component in high-NA situation.

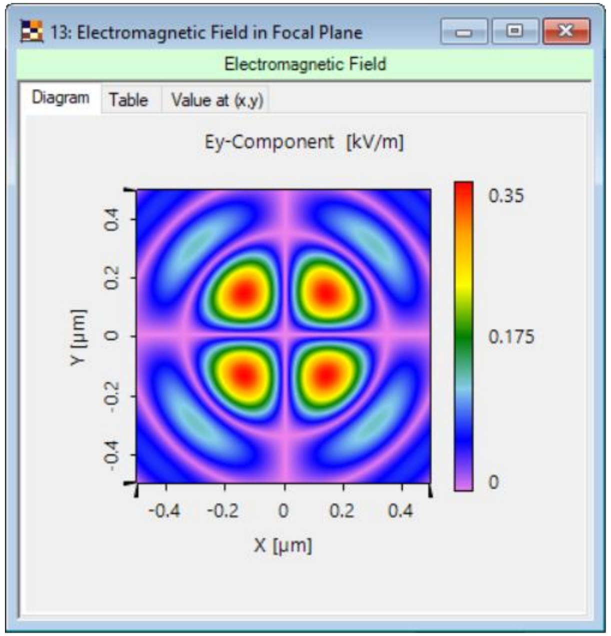


# Field Tracing Results (Electromagnetic Field Detector)

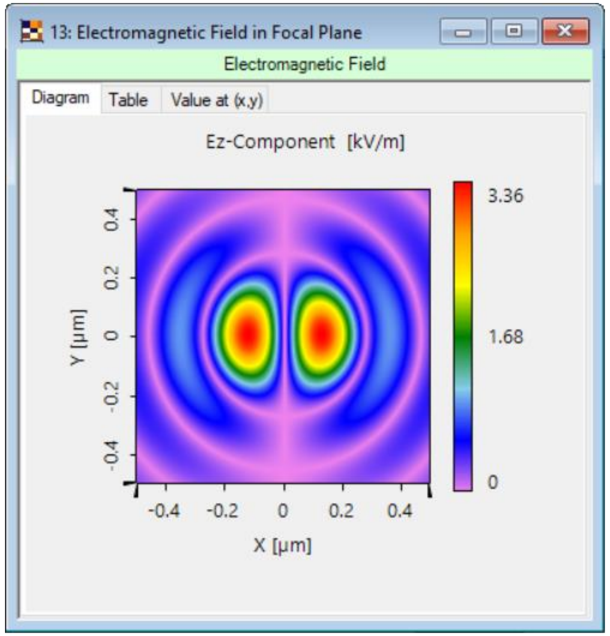
- All electromagnetic field components are obtained by using the Electromagnetic Field Detector.




amplitude of $E_x$



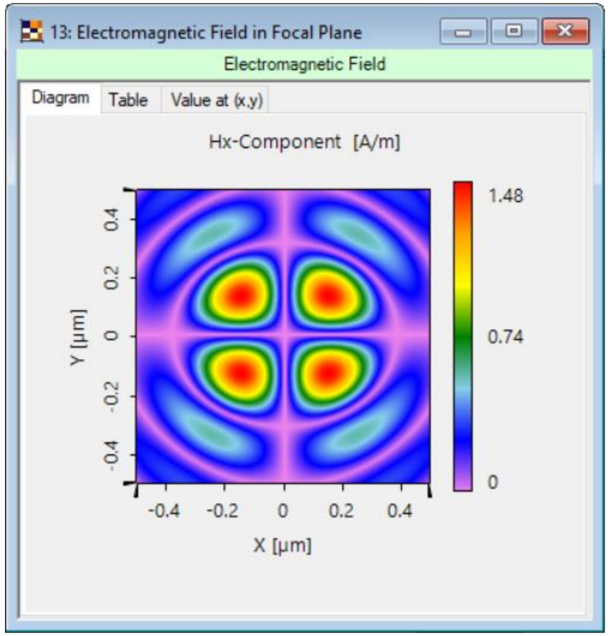

amplitude of $E_y$



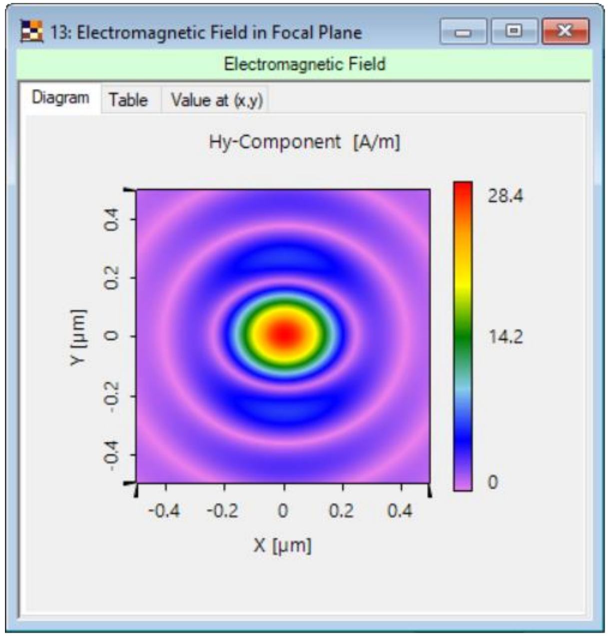

amplitude of $E_z$

# Field Tracing Results (Electromagnetic Field Detector)

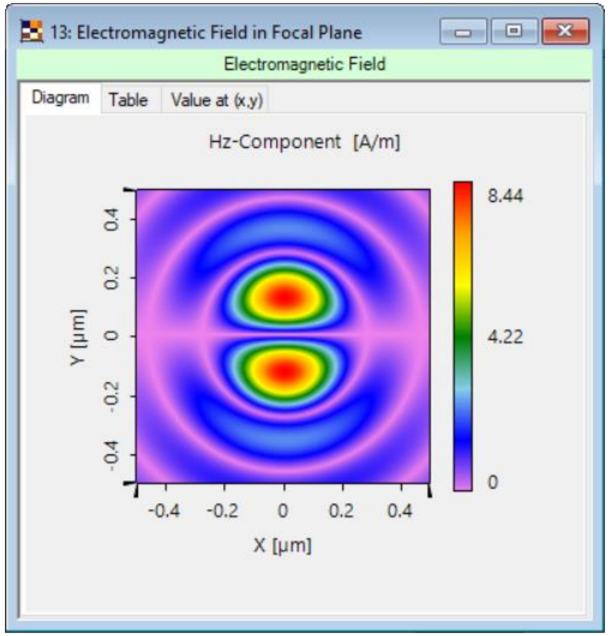
- All electromagnetic field components are obtained by using the Electromagnetic Field Detector.




amplitude of $H_x$




amplitude of $H_y$




amplitude of $H_z$

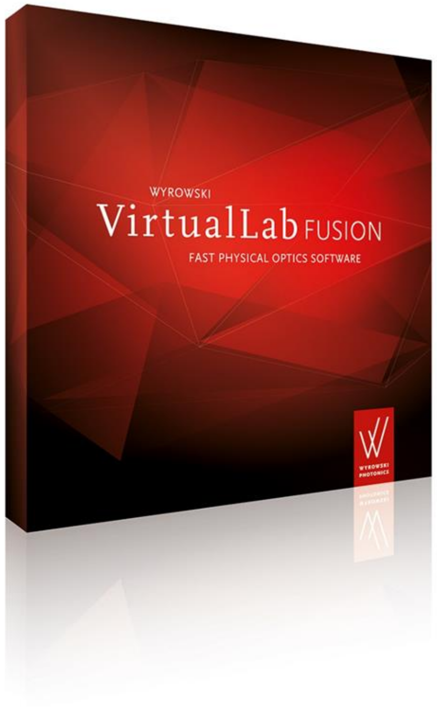
Part 2

# The Electromagnetic Field Solvers

# One Platform, Many Solvers

Fast physical optics simulations…

… made possible by **connecting field solvers**!

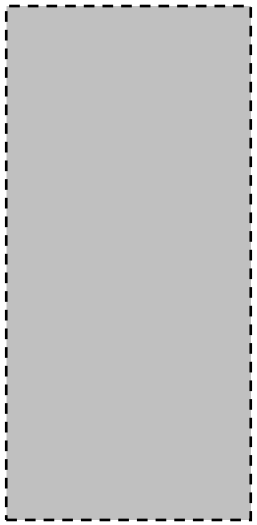



VirtualLab Fusion acts a software platform to connect electromagnetic field solvers in a **seamless, fully non-sequential manner**

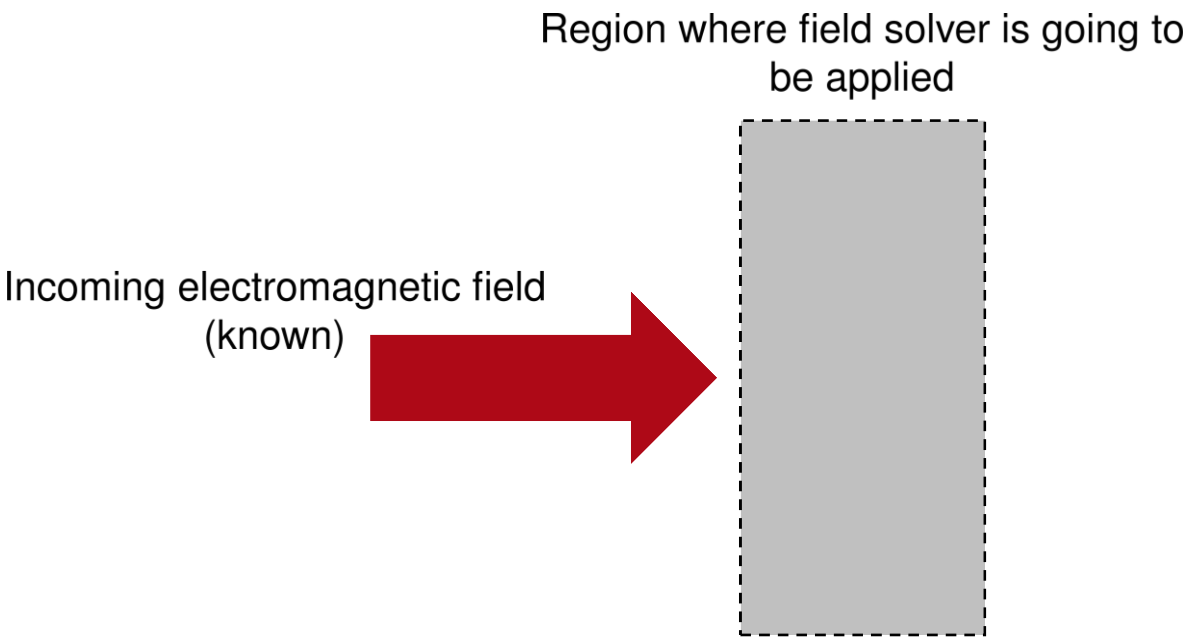
→ **solving Maxwell's equations for the whole system!**

# Electromagnetic Field Solvers

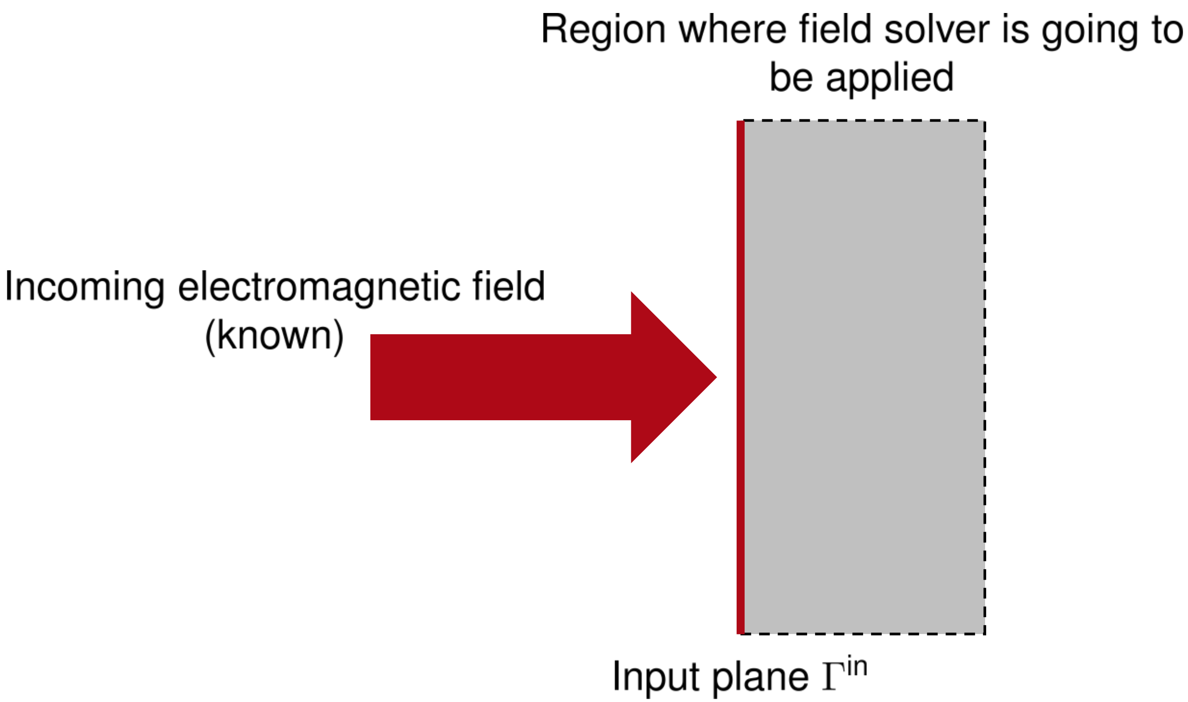
Region where field solver is going to be applied



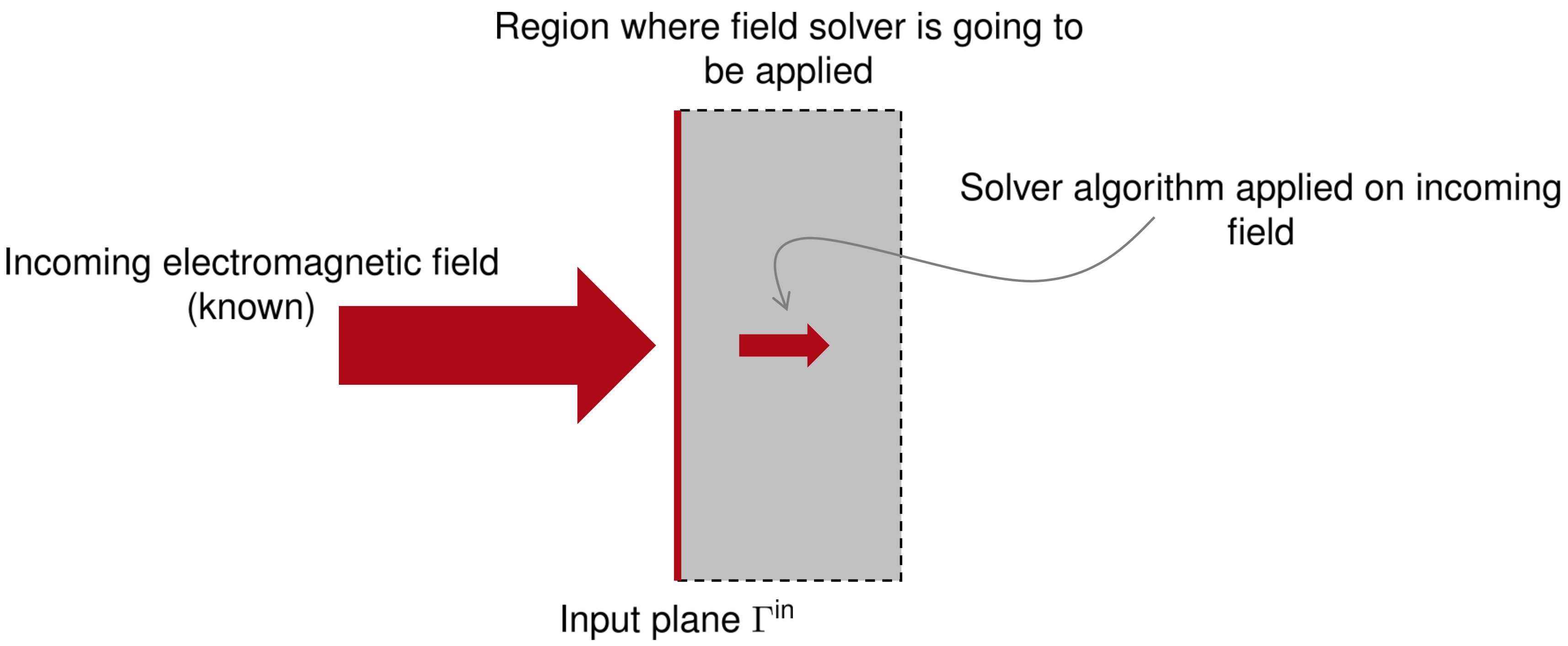
# Electromagnetic Field Solvers



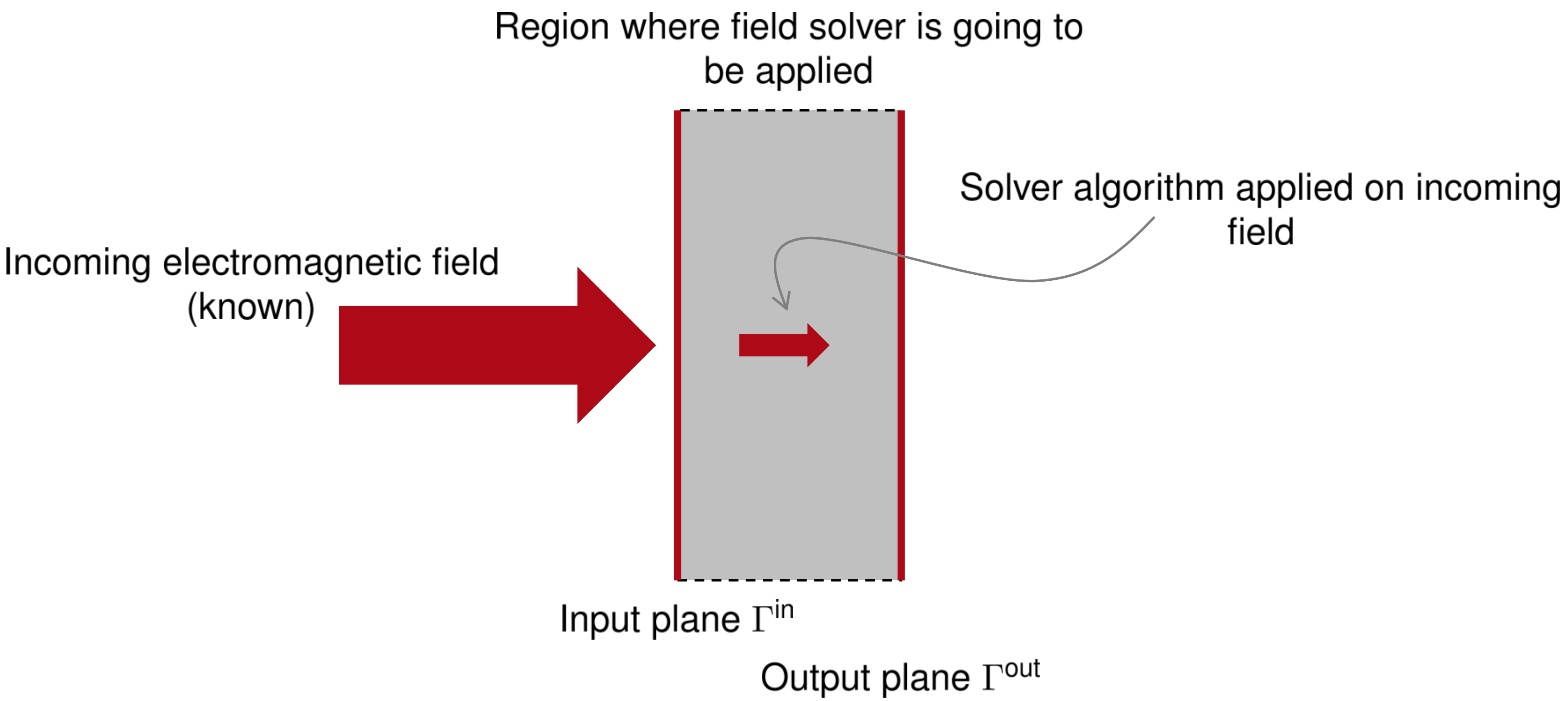
# Electromagnetic Field Solvers



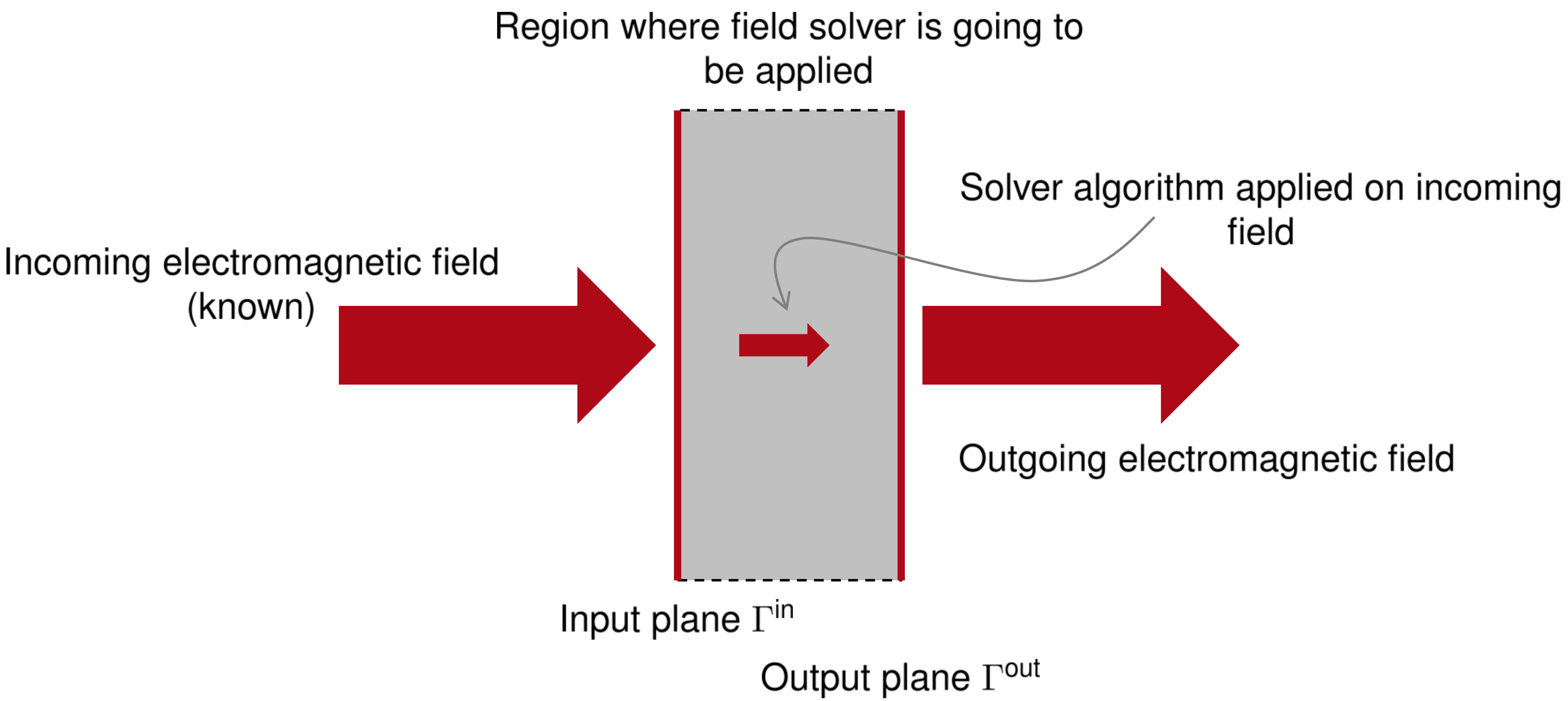
# Electromagnetic Field Solvers



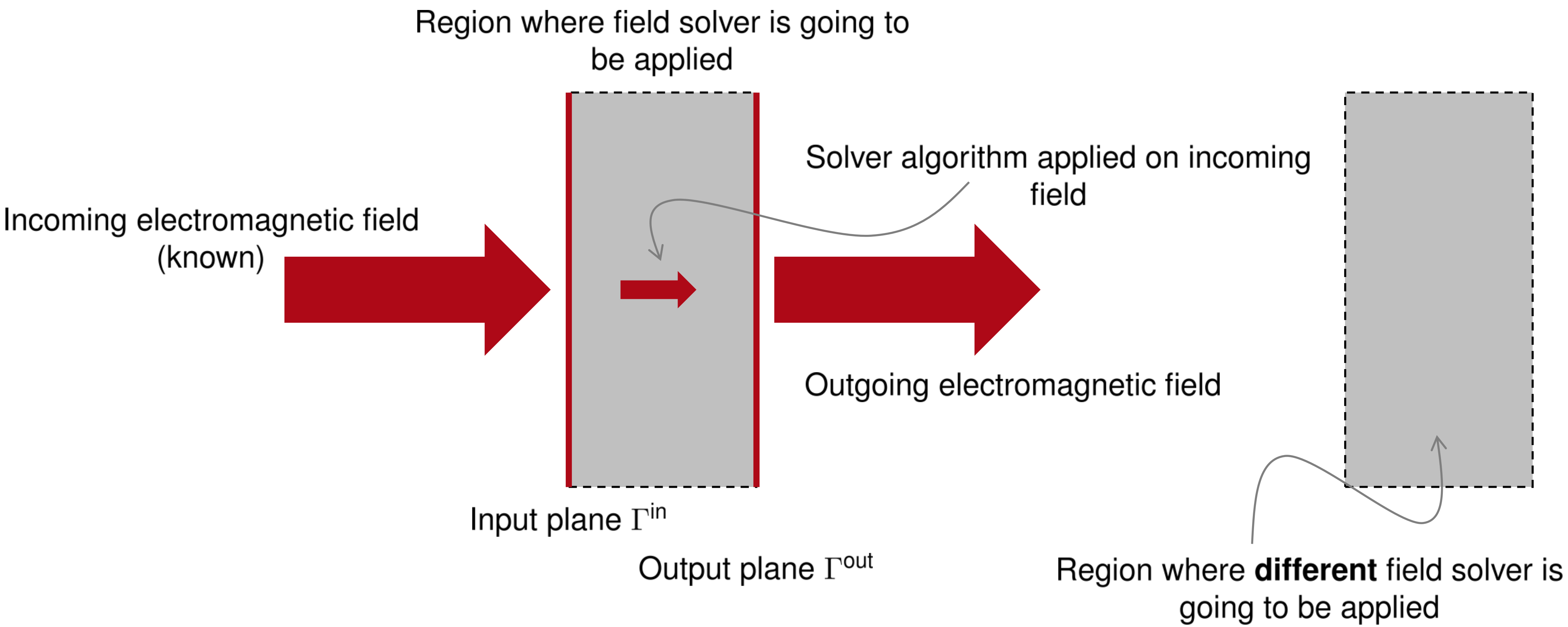
# Electromagnetic Field Solvers



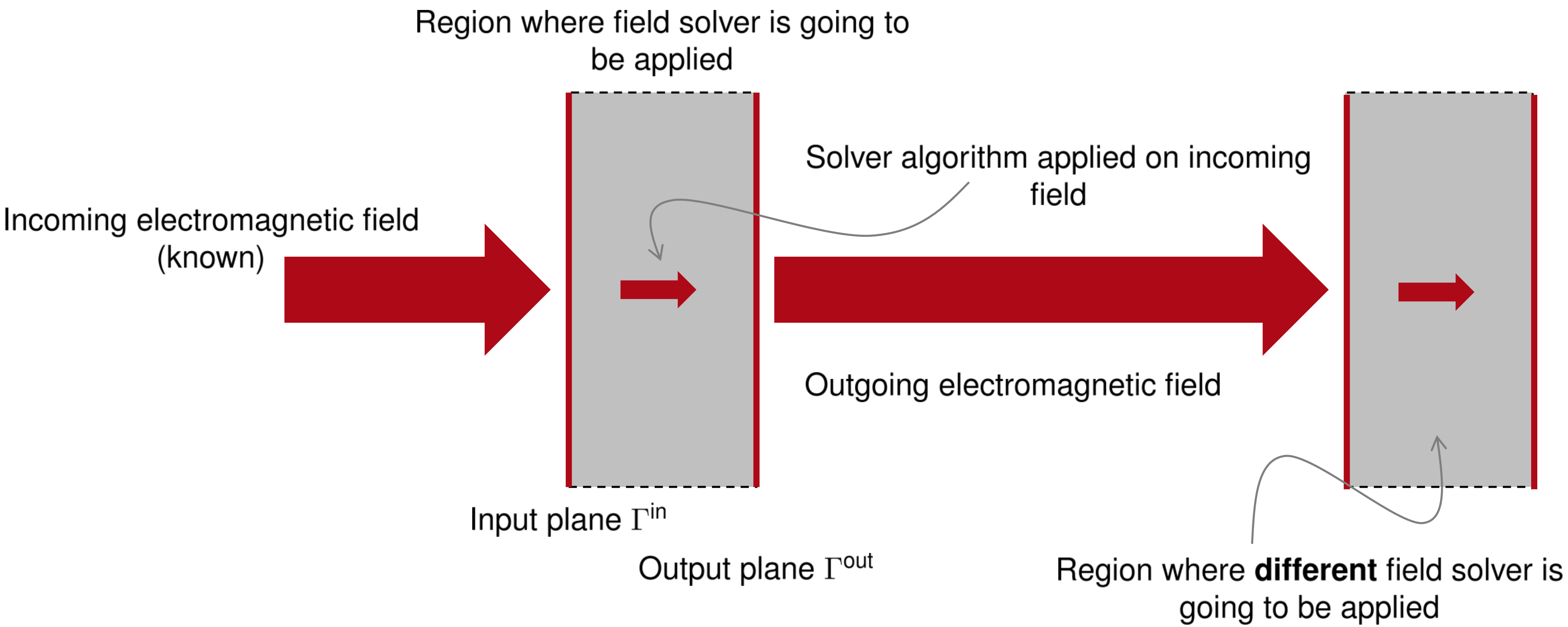
# Electromagnetic Field Solvers



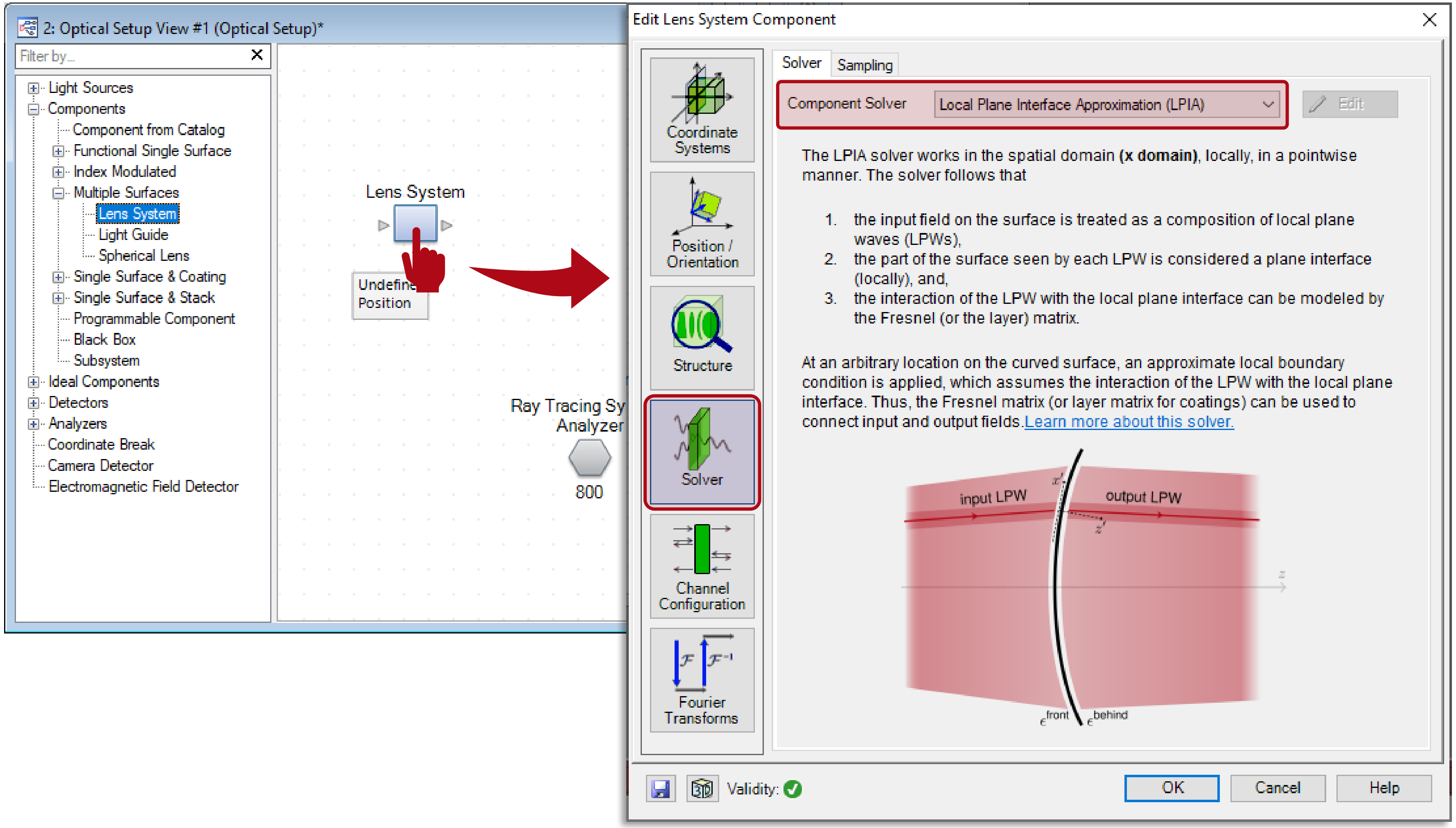
# Electromagnetic Field Solvers



# Electromagnetic Field Solvers

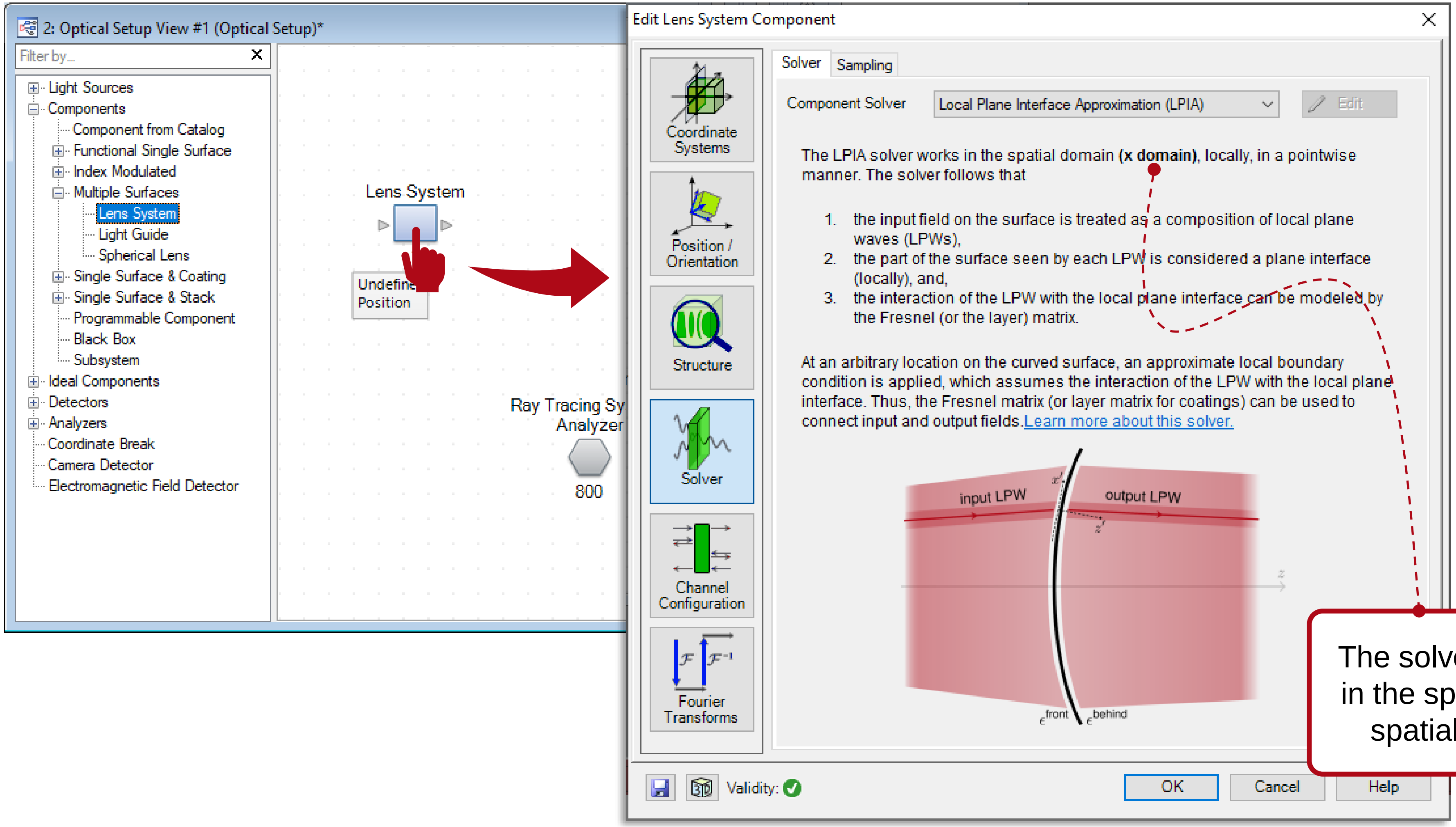


# Components & Solvers




In VirtualLab Fusion, including a certain type of component in your system means, in practice, selecting an **electromagnetic field solver** to model that part of the system

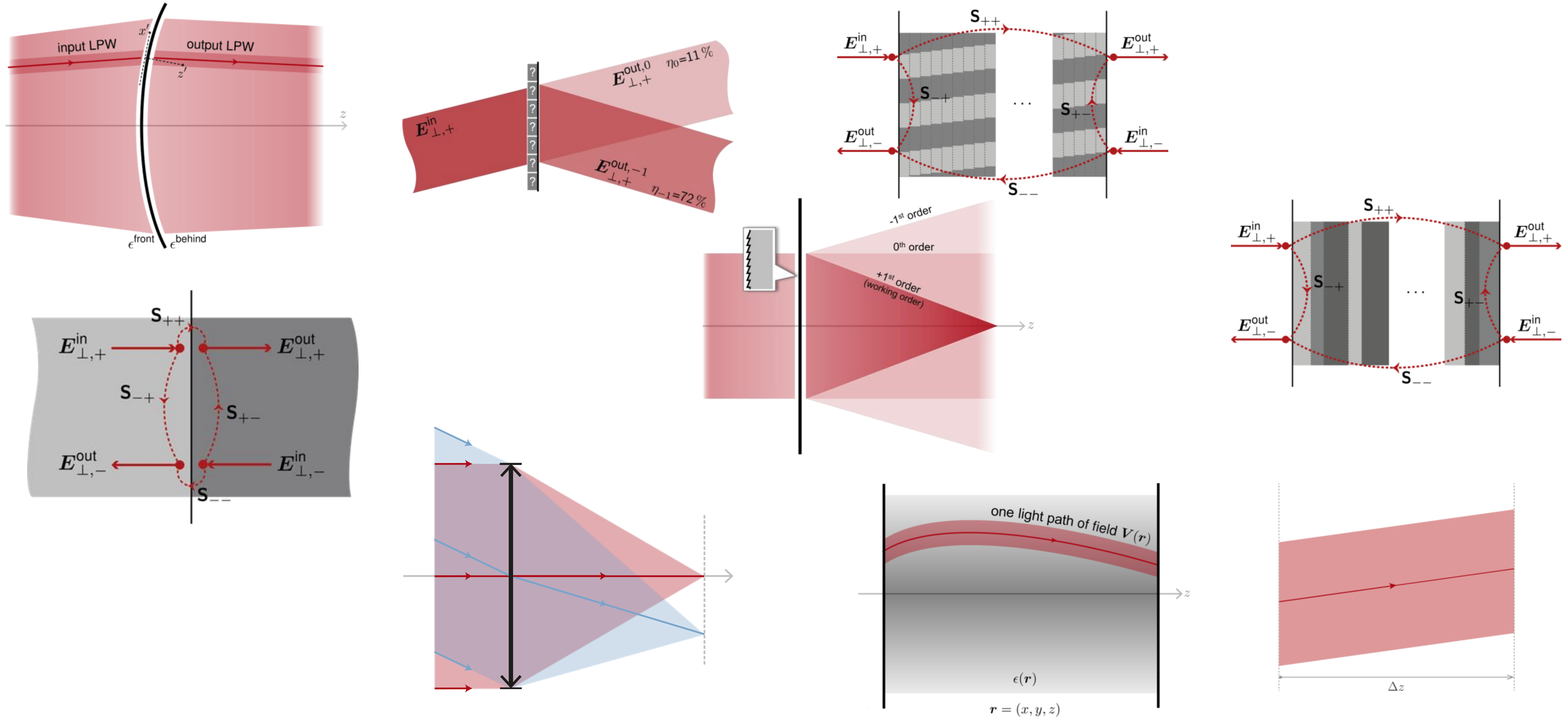
# Components & Solvers



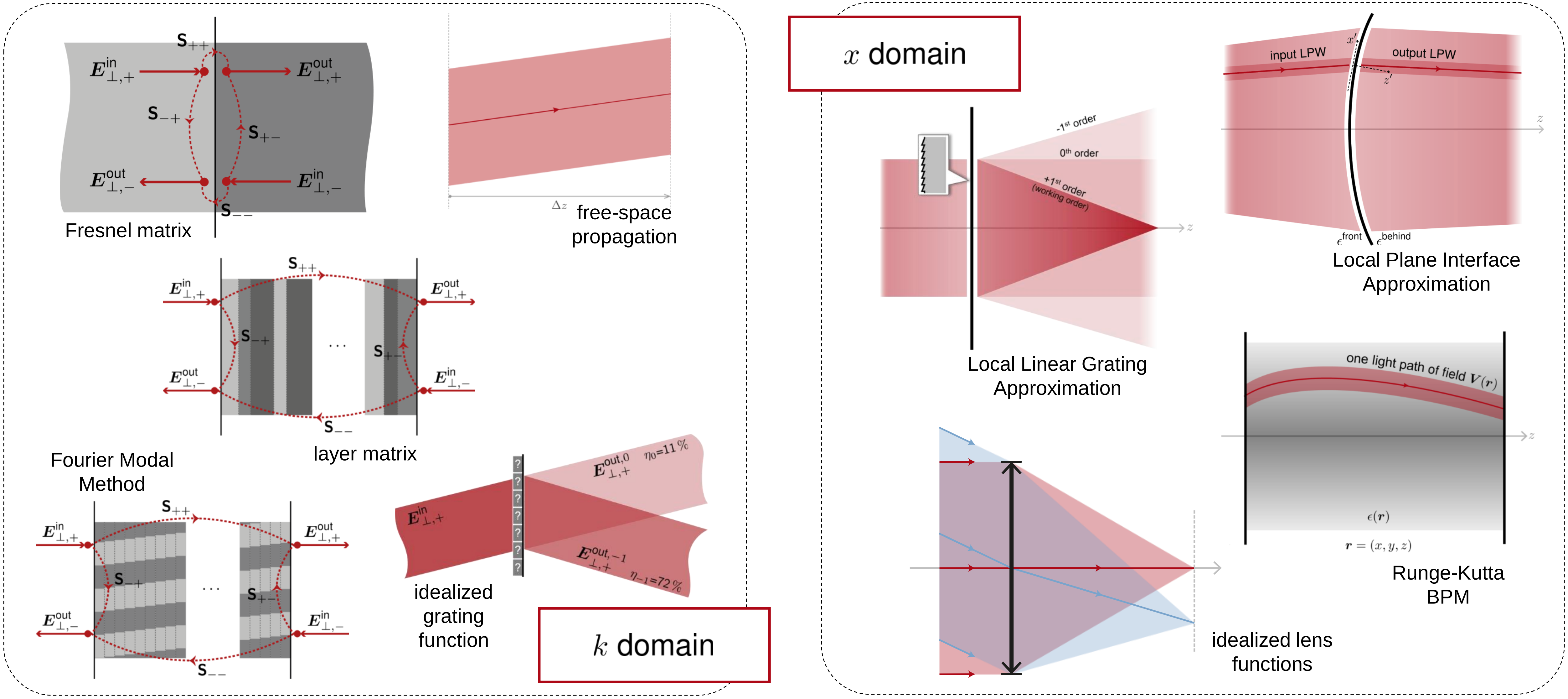

In VirtualLab Fusion, including a certain type of component in your system means, in practice, selecting an **electromagnetic field solver** to model that part of the system

The solvers may be implemented in the space (x) domain, or in the spatial-frequency (k) domain

# Why Different Fourier Domains?

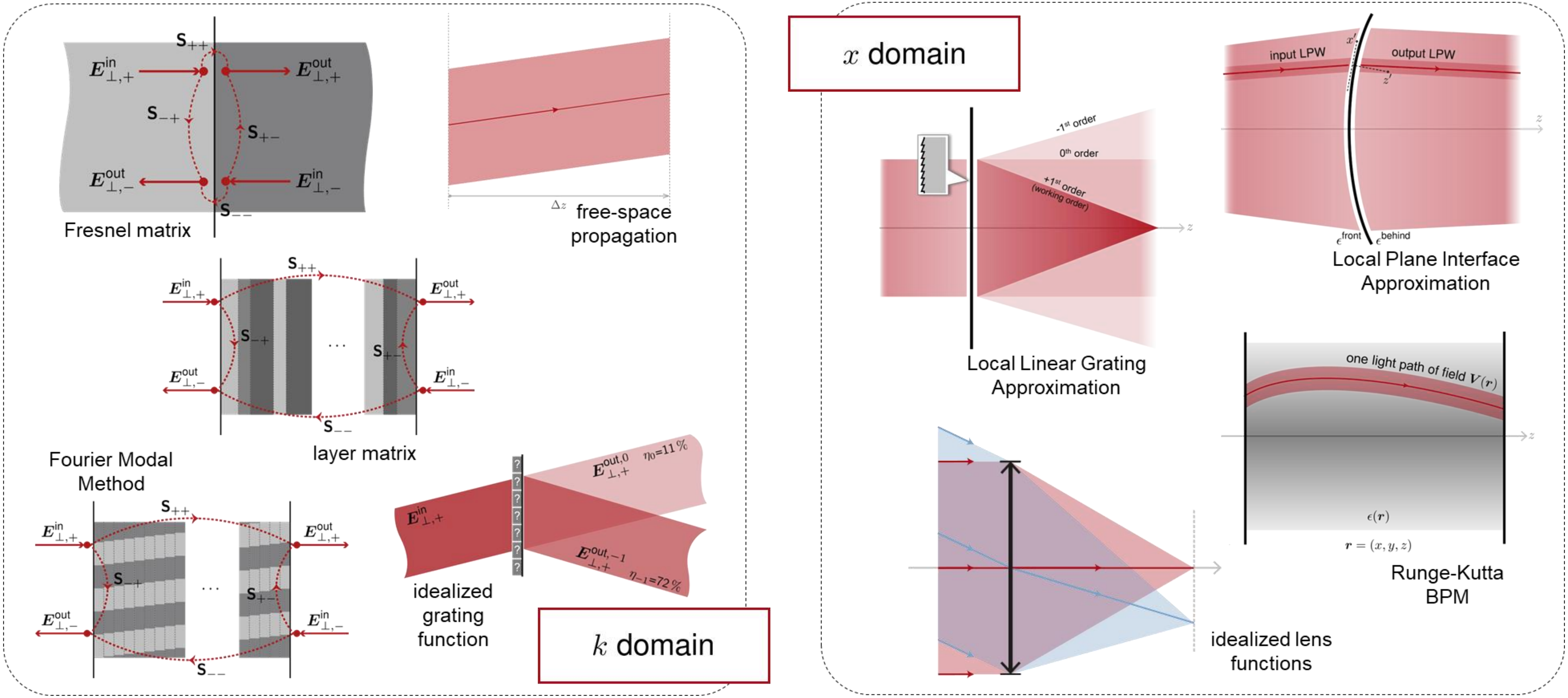


# Why Different Fourier Domains?

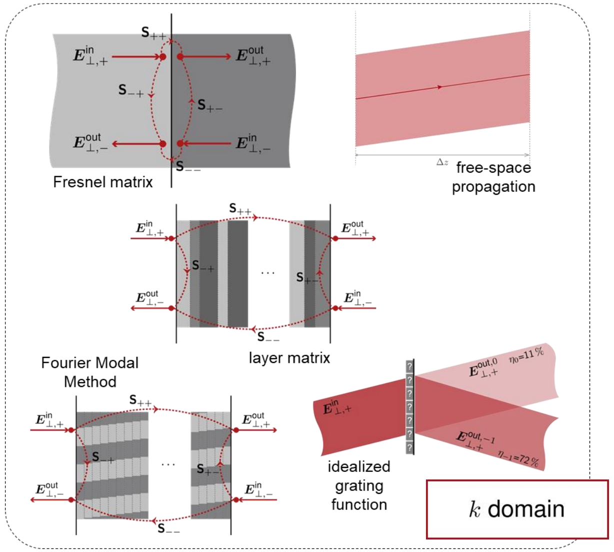


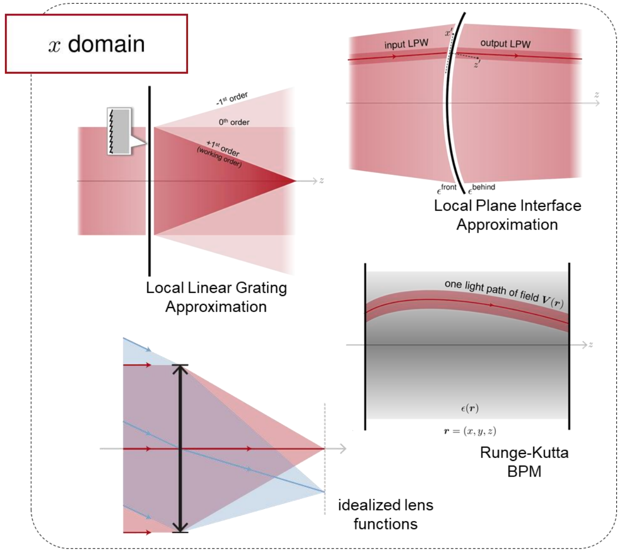

Hint: click on the logos for additional documentation on the solvers!

# Why Different Fourier Domains?



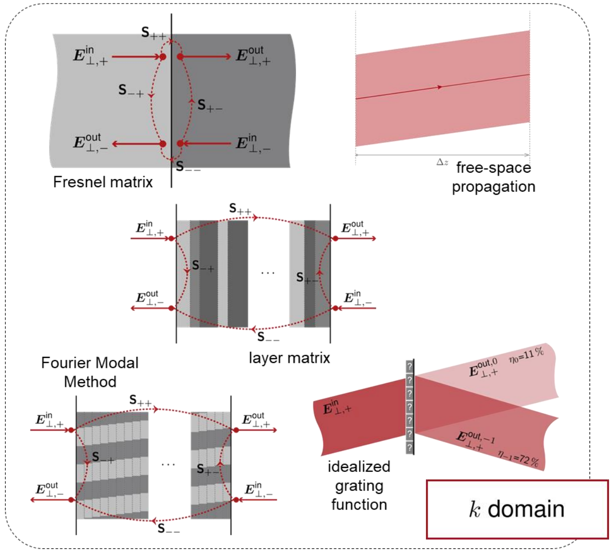
# Why Different Fourier Domains?

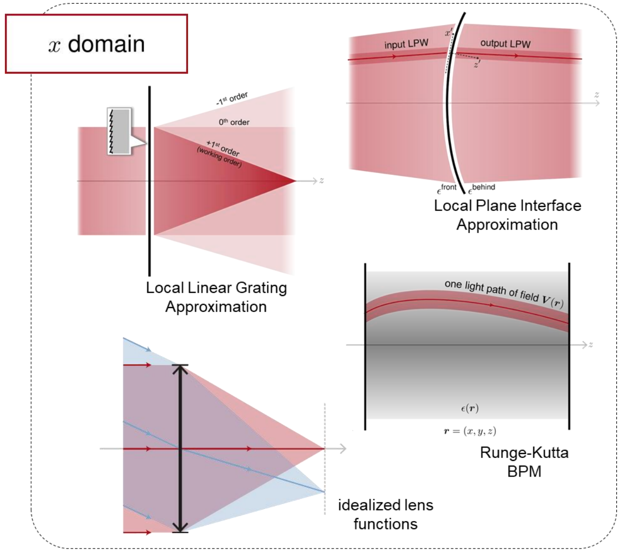


# Why Different Fourier Domains?

Although, in general, electromagnetic field solvers have an **integral** behaviour, with the resulting high numerical complexity, the characteristics of some of the most common optical components mean they can be modeled with **pointwise** operators in one of the Fourier domains – when this happens, it entails a **massive computational advantage**!



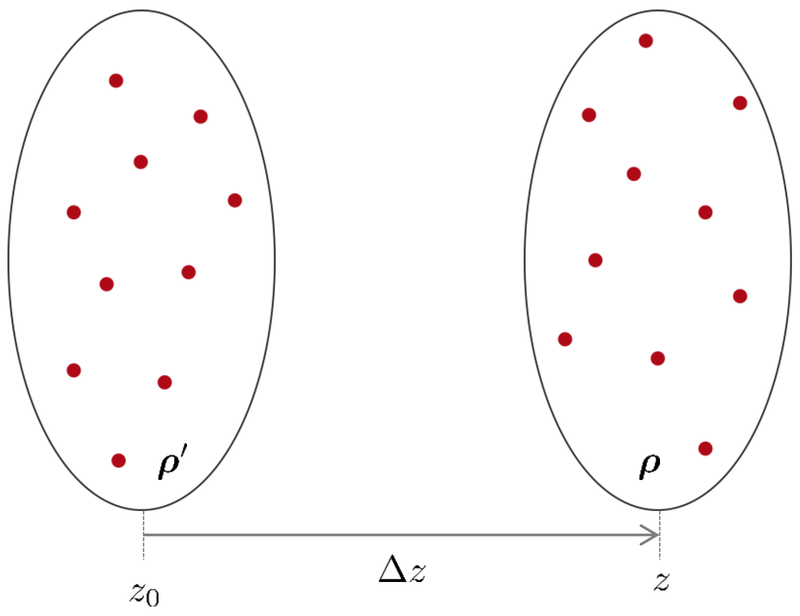
# Example: Free-Space Propagation

## Space domain

Rayleigh-Sommerfeld integral:

$$V_\ell^{\text{out}}(\boldsymbol{\rho}, z) \propto \iint_{-\infty}^{+\infty} V_\ell^{\text{in}}(\boldsymbol{\rho}', z_0) \frac{\mathrm{e}^{\mathrm{i}k_0 n R}}{R} \left(\mathrm{i}k_0 n - \frac{1}{R}\right) \frac{\Delta z}{R} \mathrm{d}^2\rho'$$

with $R = \sqrt{(x - x')^2 + (y - y')^2 + (\Delta z)^2}$



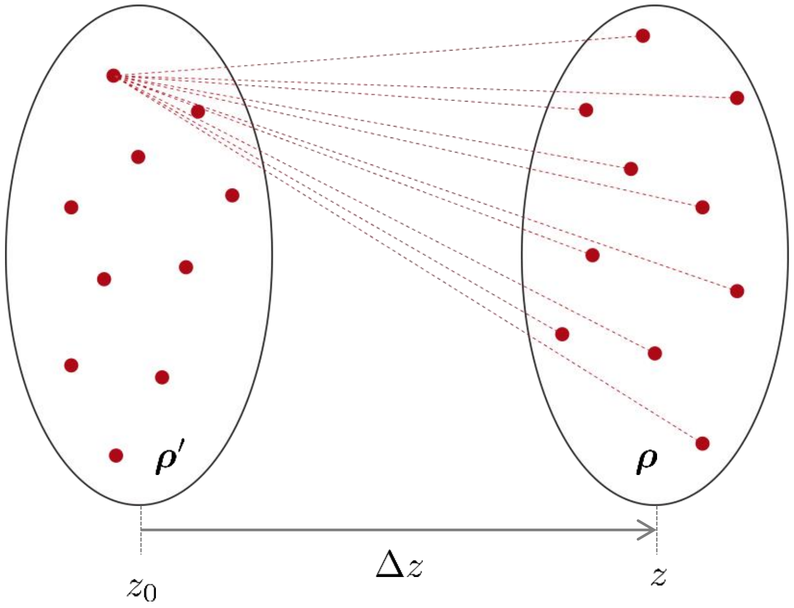
# Example: Free-Space Propagation

## Space domain

Rayleigh-Sommerfeld integral:

$$V_\ell^{\text{out}}(\boldsymbol{\rho}, z) \propto \iint_{-\infty}^{+\infty} V_\ell^{\text{in}}(\boldsymbol{\rho}', z_0) \frac{\mathrm{e}^{\mathrm{i}k_0 n R}}{R} \left(\mathrm{i}k_0 n - \frac{1}{R}\right) \frac{\Delta z}{R} \mathrm{d}^2 \rho'$$

with $R = \sqrt{(x - x')^2 + (y - y')^2 + (\Delta z)^2}$



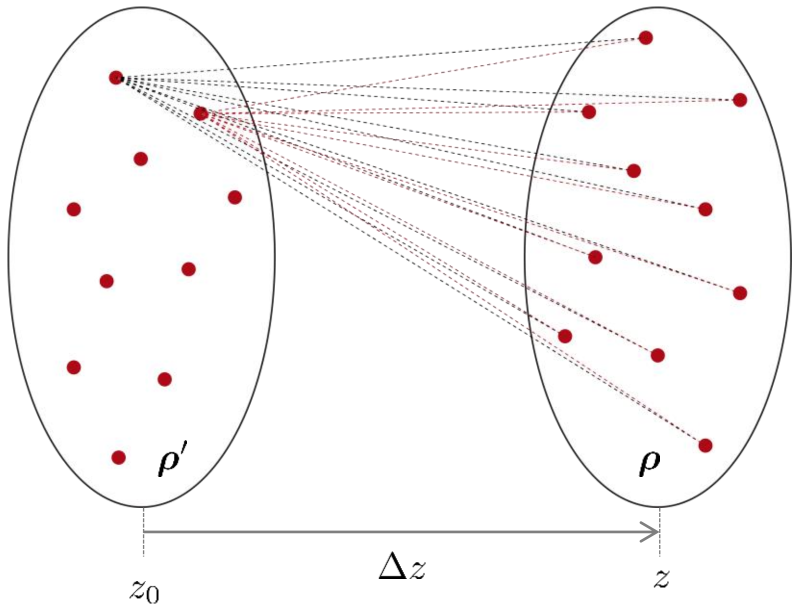
# Example: Free-Space Propagation

## Space domain

Rayleigh-Sommerfeld integral:

$$V_\ell^{\text{out}}(\boldsymbol{\rho}, z) \propto \iint_{-\infty}^{+\infty} V_\ell^{\text{in}}(\boldsymbol{\rho}', z_0) \frac{\mathrm{e}^{\mathrm{i}k_0 n R}}{R} \left(\mathrm{i}k_0 n - \frac{1}{R}\right) \frac{\Delta z}{R} \mathrm{d}^2\rho'$$

with $R = \sqrt{(x - x')^2 + (y - y')^2 + (\Delta z)^2}$



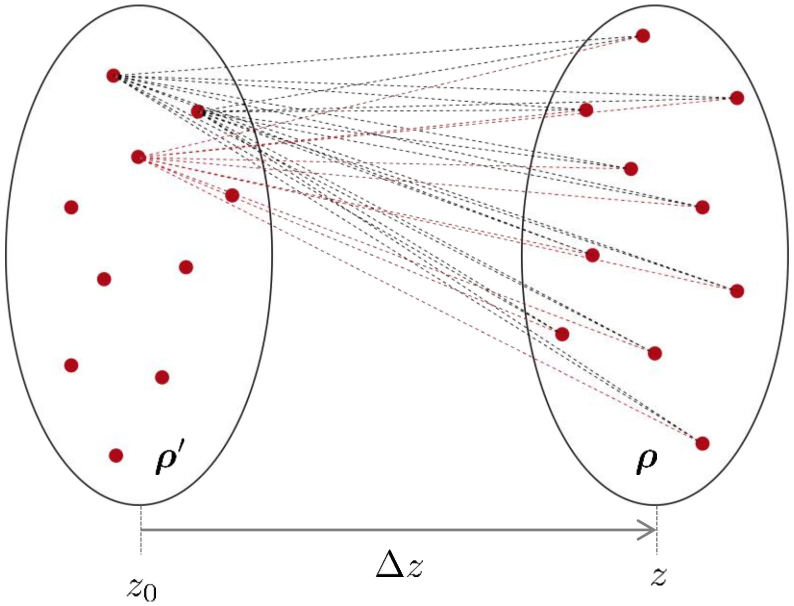
# Example: Free-Space Propagation

## Space domain

Rayleigh-Sommerfeld integral:

$$V_\ell^{\text{out}}(\boldsymbol{\rho}, z) \propto \iint_{-\infty}^{+\infty} V_\ell^{\text{in}}(\boldsymbol{\rho}', z_0) \frac{e^{ik_0 nR}}{R} \left( ik_0 n - \frac{1}{R} \right) \frac{\Delta z}{R} d^2\rho'$$

with $R = \sqrt{(x - x')^2 + (y - y')^2 + (\Delta z)^2}$



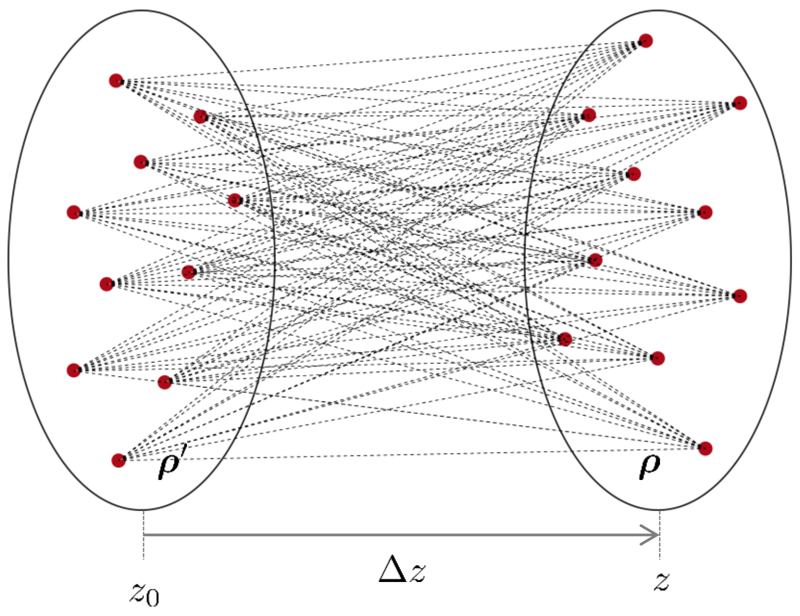
# Example: Free-Space Propagation

## Space domain

Rayleigh-Sommerfeld integral:

$$V_\ell^{\text{out}}(\boldsymbol{\rho}, z) \propto \iint_{-\infty}^{+\infty} V_\ell^{\text{in}}(\boldsymbol{\rho}', z_0) \frac{\mathrm{e}^{\mathrm{i}k_0 n R}}{R} \left(\mathrm{i}k_0 n - \frac{1}{R}\right) \frac{\Delta z}{R} \mathrm{d}^2 \rho'$$

with $R = \sqrt{(x - x')^2 + (y - y')^2 + (\Delta z)^2}$



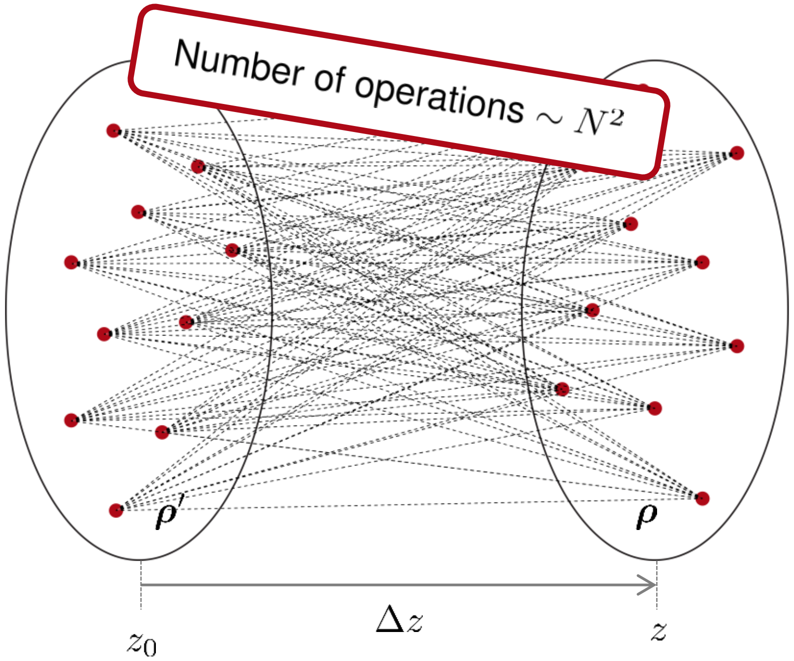
# Example: Free-Space Propagation

**Space domain**

Rayleigh-Sommerfeld integral:

$$V_\ell^{\text{out}}(\boldsymbol{\rho}, z) \propto \iint_{-\infty}^{+\infty} V_\ell^{\text{in}}(\boldsymbol{\rho}', z_0) \frac{e^{ik_0 n R}}{R} \left( ik_0 n - \frac{1}{R} \right) \frac{\Delta z}{R} d^2\rho'$$

with $R = \sqrt{(x - x')^2 + (y - y')^2 + (\Delta z)^2}$



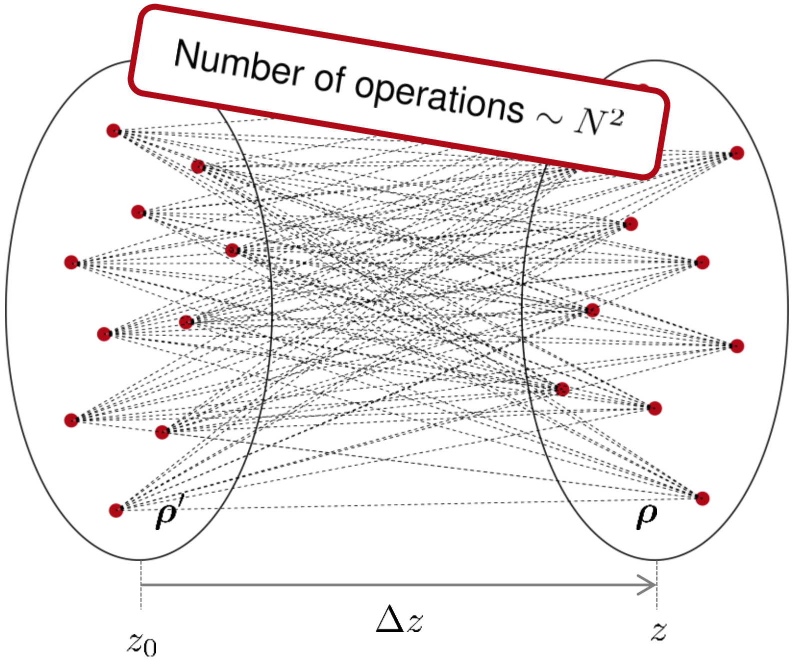
# Example: Free-Space Propagation

**Space domain**

Rayleigh-Sommerfeld integral:

$$V_\ell^{\text{out}}(\boldsymbol{\rho}, z) \propto \iint_{-\infty}^{+\infty} V_\ell^{\text{in}}(\boldsymbol{\rho}', z_0) \frac{e^{ik_0 n R}}{R}\left(ik_0 n - \frac{1}{R}\right)\frac{\Delta z}{R} d^2\rho'$$

with $R = \sqrt{(x-x')^2 + (y-y')^2 + (\Delta z)^2}$




**Spatial-frequency domain**

Plane-wave propagation operator:

$$\tilde{V}_\ell^{\text{out}}(\boldsymbol{\kappa}, z) = \tilde{V}_\ell^{\text{in}}(\boldsymbol{\kappa}, z_0) \times e^{ik_z(\kappa)\Delta z}$$

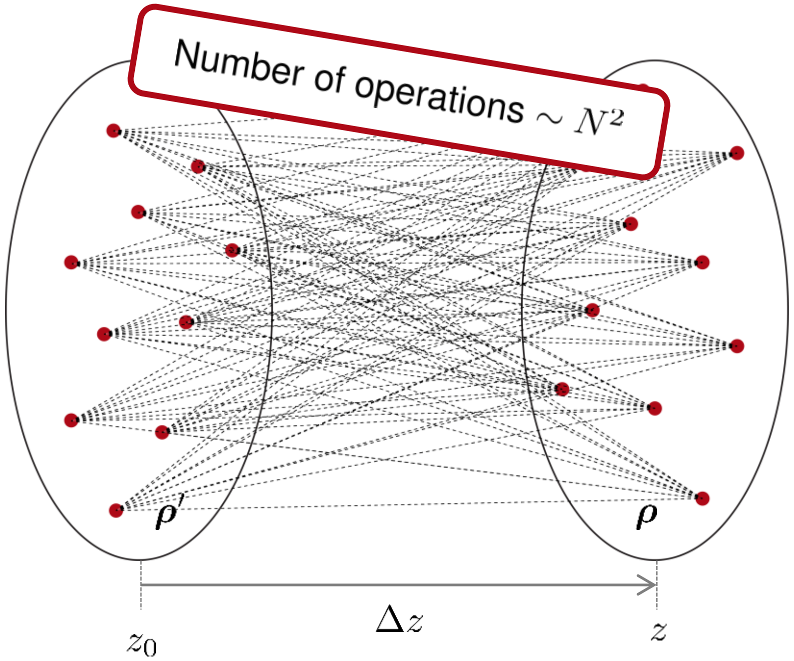
# Example: Free-Space Propagation

## Space domain

Rayleigh-Sommerfeld integral:

$$V_\ell^{\text{out}}(\boldsymbol{\rho}, z) \propto \iint_{-\infty}^{+\infty} V_\ell^{\text{in}}(\boldsymbol{\rho}', z_0) \frac{e^{ik_0 n R}}{R}\left(ik_0 n - \frac{1}{R}\right)\frac{\Delta z}{R} d^2\rho'$$

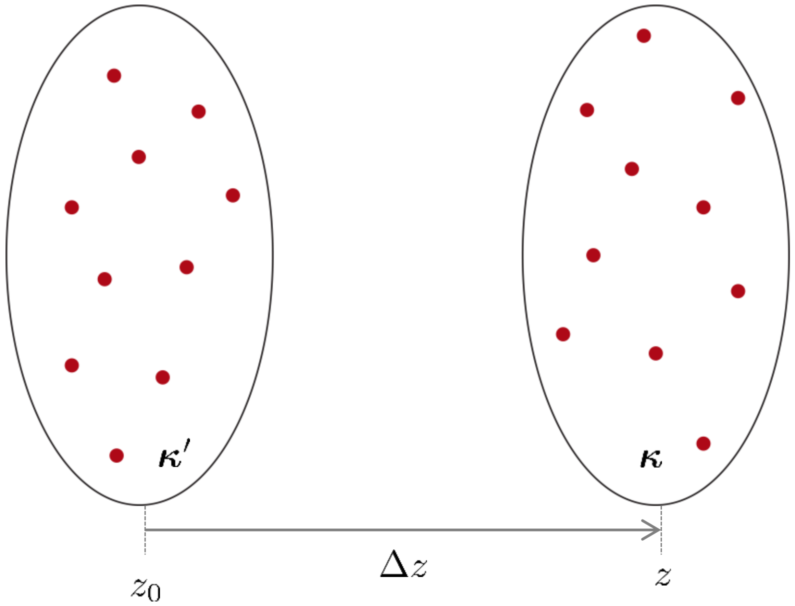
with $R = \sqrt{(x - x')^2 + (y - y')^2 + (\Delta z)^2}$




## Spatial-frequency domain

Plane-wave propagation operator:

$$\tilde{V}_\ell^{\text{out}}(\boldsymbol{\kappa}, z) = \tilde{V}_\ell^{\text{in}}(\boldsymbol{\kappa}, z_0) \times e^{ik_z(\kappa)\Delta z}$$



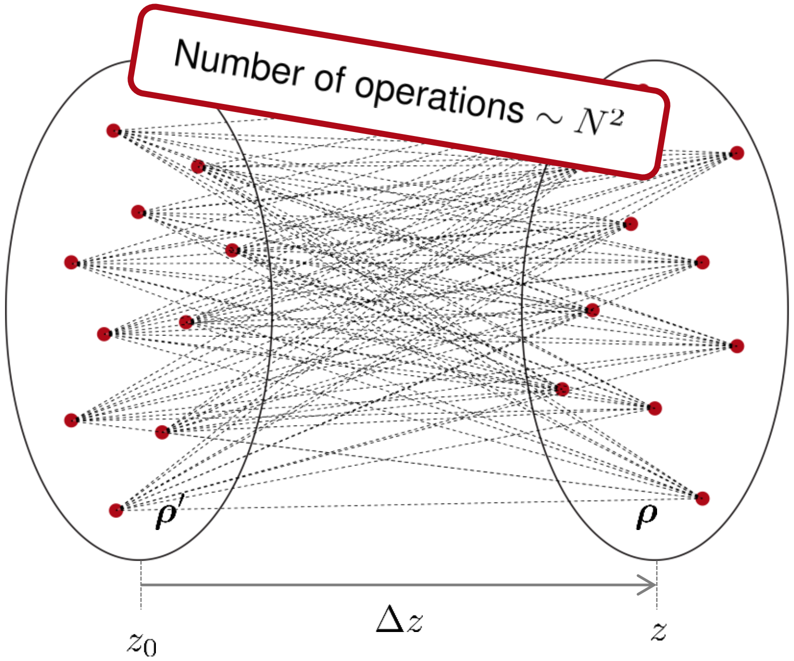
# Example: Free-Space Propagation

## Space domain

Rayleigh-Sommerfeld integral:

$$V_\ell^{\text{out}}(\boldsymbol{\rho}, z) \propto \iint_{-\infty}^{+\infty} V_\ell^{\text{in}}(\boldsymbol{\rho}', z_0) \frac{\mathrm{e}^{\mathrm{i}k_0 n R}}{R}\left(\mathrm{i}k_0 n - \frac{1}{R}\right)\frac{\Delta z}{R}\mathrm{d}^2\rho'$$

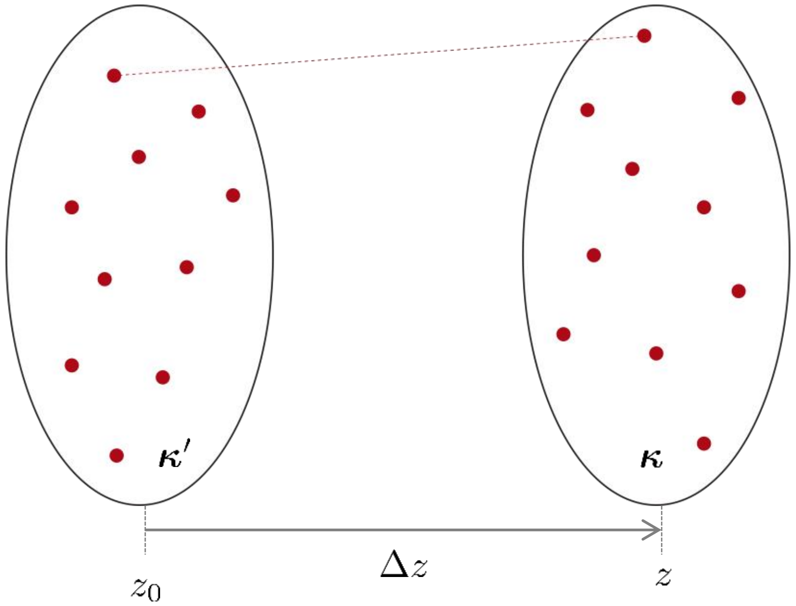
with $R = \sqrt{(x-x')^2 + (y-y')^2 + (\Delta z)^2}$




## Spatial-frequency domain

Plane-wave propagation operator:

$$\tilde{V}_\ell^{\text{out}}(\boldsymbol{\kappa}, z) = \tilde{V}_\ell^{\text{in}}(\boldsymbol{\kappa}, z_0) \times \mathrm{e}^{\mathrm{i}k_z(\kappa)\Delta z}$$



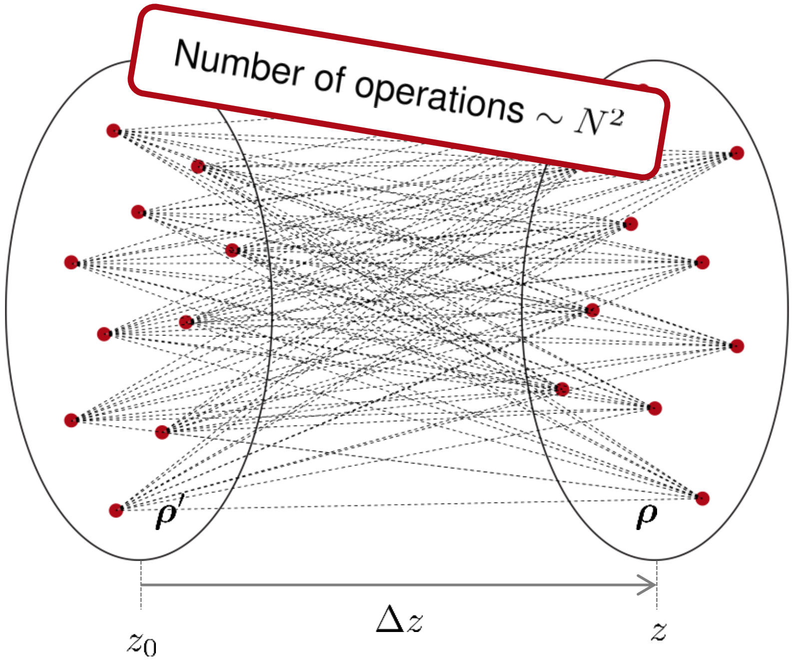
# Example: Free-Space Propagation

**Space domain**

Rayleigh-Sommerfeld integral:

$$V_\ell^{\text{out}}(\boldsymbol{\rho}, z) \propto \iint_{-\infty}^{+\infty} V_\ell^{\text{in}}(\boldsymbol{\rho}', z_0)\,\frac{\mathrm{e}^{\mathrm{i}k_0 n R}}{R}\left(\mathrm{i}k_0 n - \frac{1}{R}\right)\frac{\Delta z}{R}\mathrm{d}^2\rho'$$

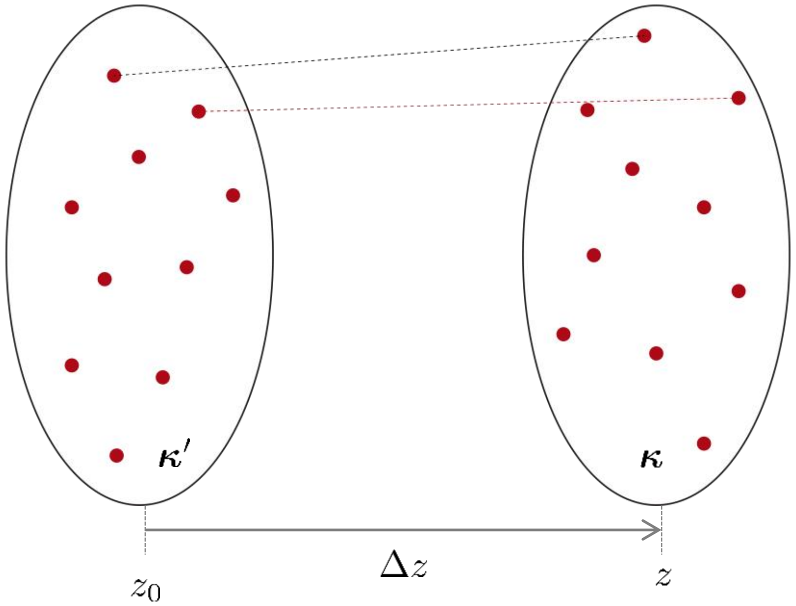
with $R = \sqrt{(x-x')^2 + (y-y')^2 + (\Delta z)^2}$




**Spatial-frequency domain**

Plane-wave propagation operator:

$$\tilde{V}_\ell^{\text{out}}(\boldsymbol{\kappa}, z) = \tilde{V}_\ell^{\text{in}}(\boldsymbol{\kappa}, z_0) \times \mathrm{e}^{\mathrm{i}k_z(\kappa)\Delta z}$$



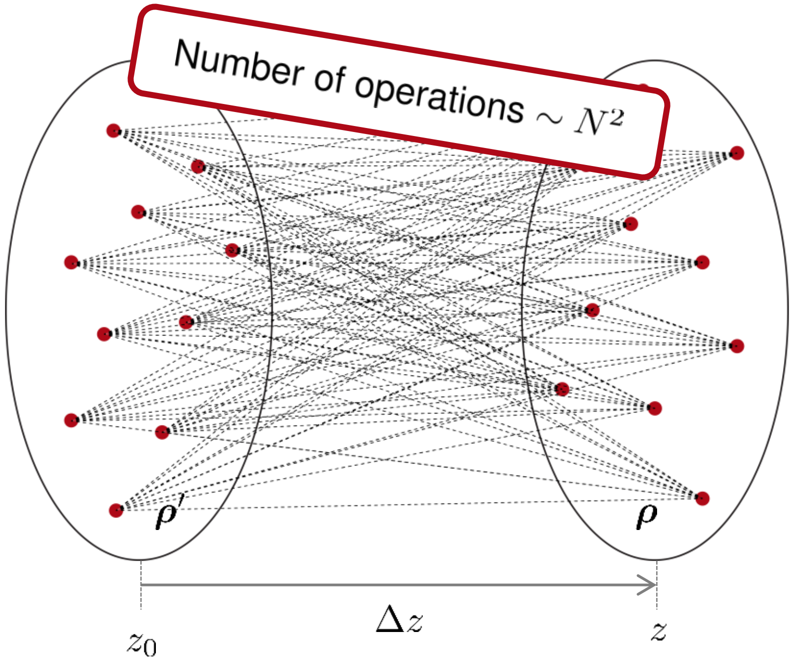
# Example: Free-Space Propagation

## Space domain

Rayleigh-Sommerfeld integral:

$$V_\ell^{\text{out}}(\boldsymbol{\rho}, z) \propto \iint_{-\infty}^{+\infty} V_\ell^{\text{in}}(\boldsymbol{\rho}', z_0)\,\frac{\mathrm{e}^{\mathrm{i}k_0 n R}}{R}\left(\mathrm{i}k_0 n - \frac{1}{R}\right)\frac{\Delta z}{R}\mathrm{d}^2\rho'$$

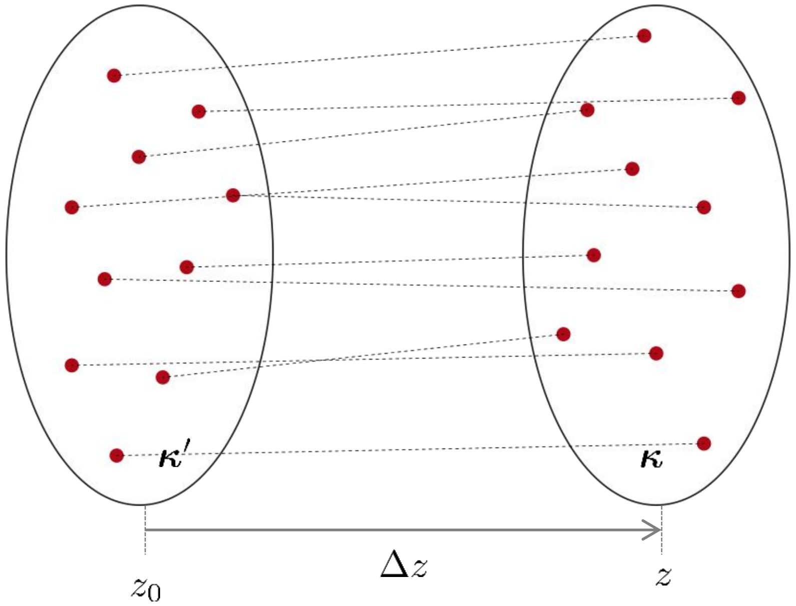
with $R = \sqrt{(x-x')^2 + (y-y')^2 + (\Delta z)^2}$




## Spatial-frequency domain

Plane-wave propagation operator:

$$\tilde{V}_\ell^{\text{out}}(\boldsymbol{\kappa}, z) = \tilde{V}_\ell^{\text{in}}(\boldsymbol{\kappa}, z_0) \times \mathrm{e}^{\mathrm{i}k_z(\kappa)\Delta z}$$



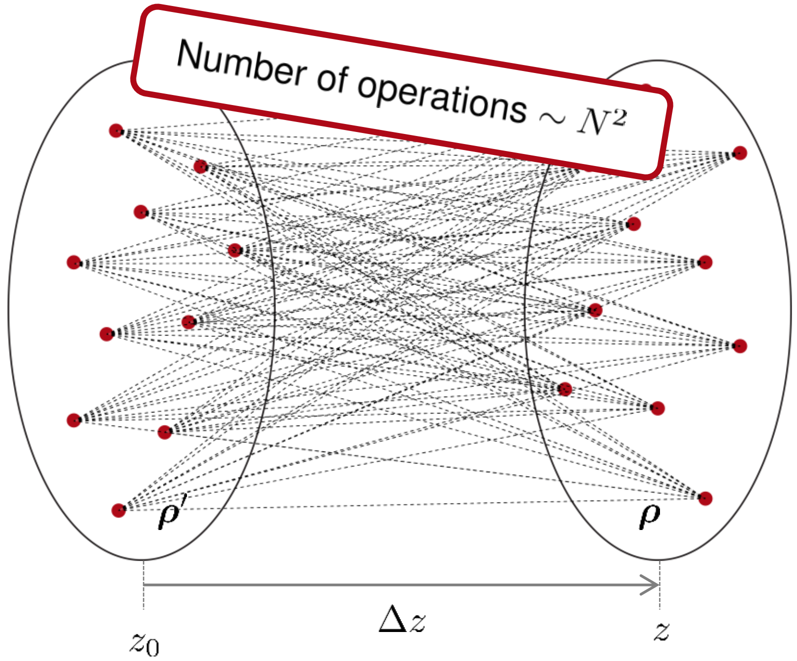
# Example: Free-Space Propagation

## Space domain

Rayleigh-Sommerfeld integral:

$$V_\ell^{\text{out}}(\boldsymbol{\rho}, z) \propto \iint_{-\infty}^{+\infty} V_\ell^{\text{in}}(\boldsymbol{\rho}', z_0) \frac{e^{ik_0 nR}}{R}\left(ik_0 n - \frac{1}{R}\right)\frac{\Delta z}{R} d^2\rho'$$

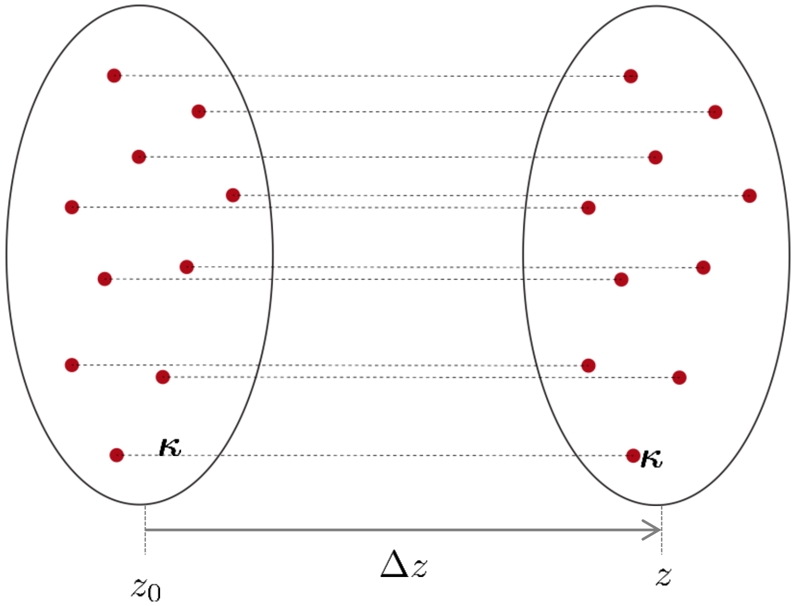
with $R = \sqrt{(x-x')^2 + (y-y')^2 + (\Delta z)^2}$




## Spatial-frequency domain

Plane-wave propagation operator:

$$\tilde{V}_\ell^{\text{out}}(\boldsymbol{\kappa}, z) = \tilde{V}_\ell^{\text{in}}(\boldsymbol{\kappa}, z_0) \times e^{ik_z(\kappa)\Delta z}$$



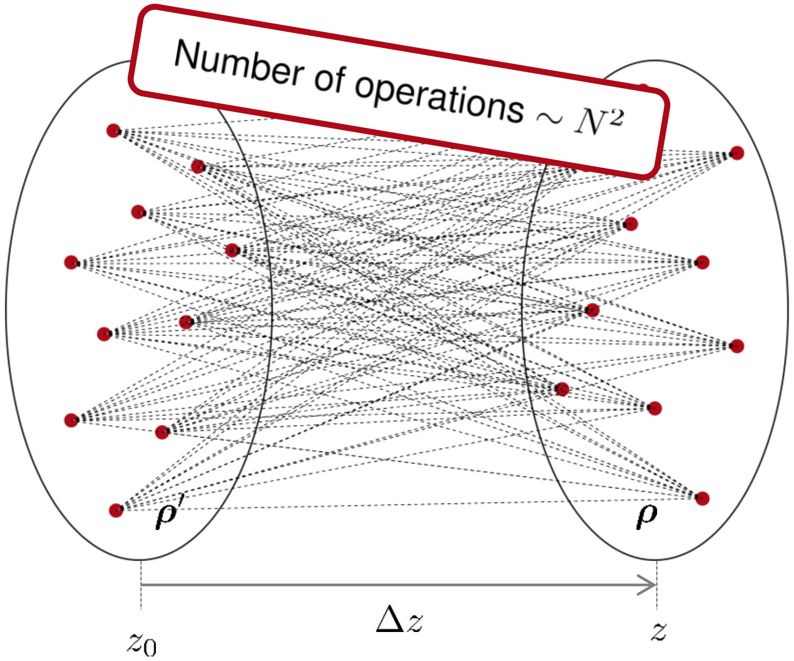
# Example: Free-Space Propagation

## Space domain

Rayleigh-Sommerfeld integral:

$$V_\ell^{\mathrm{out}}(\boldsymbol{\rho}, z) \propto \iint_{-\infty}^{+\infty} V_\ell^{\mathrm{in}}(\boldsymbol{\rho}', z_0) \frac{\mathrm{e}^{\mathrm{i}k_0 n R}}{R} \left(\mathrm{i}k_0 n - \frac{1}{R}\right) \frac{\Delta z}{R} \mathrm{d}^2\rho'$$

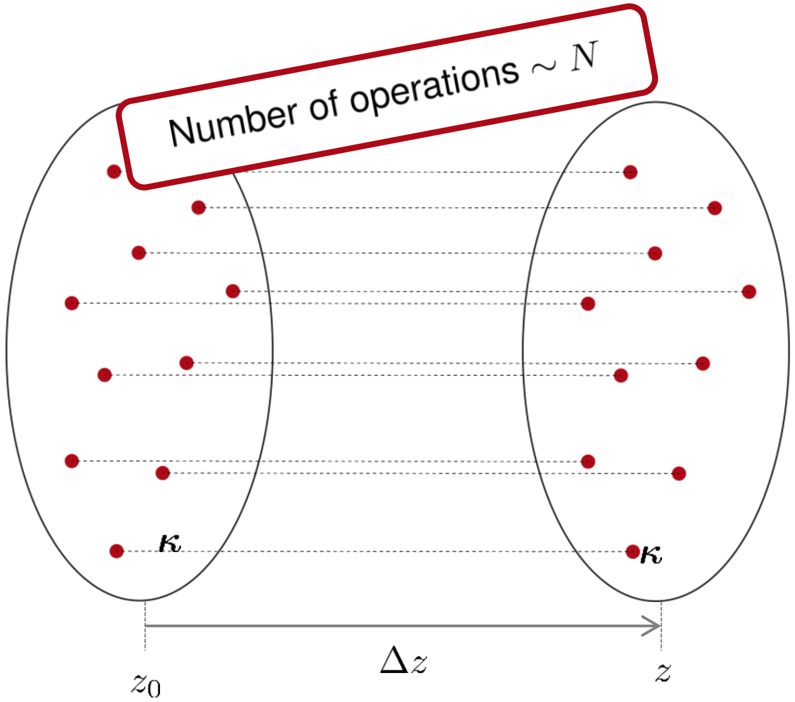
with $R = \sqrt{(x-x')^2 + (y-y')^2 + (\Delta z)^2}$




## Spatial-frequency domain

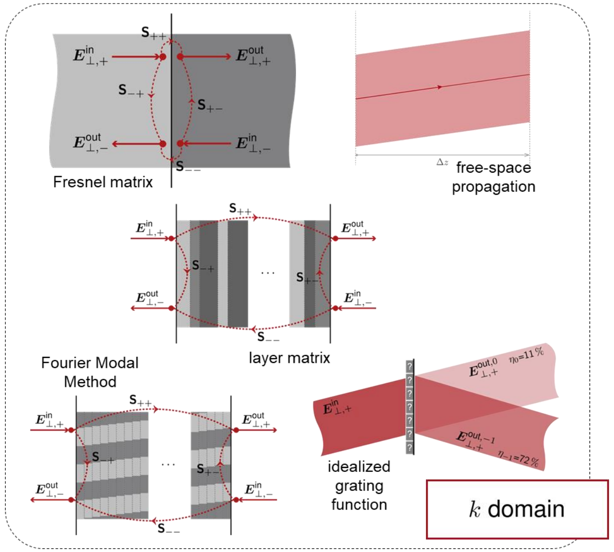
Plane-wave propagation operator:

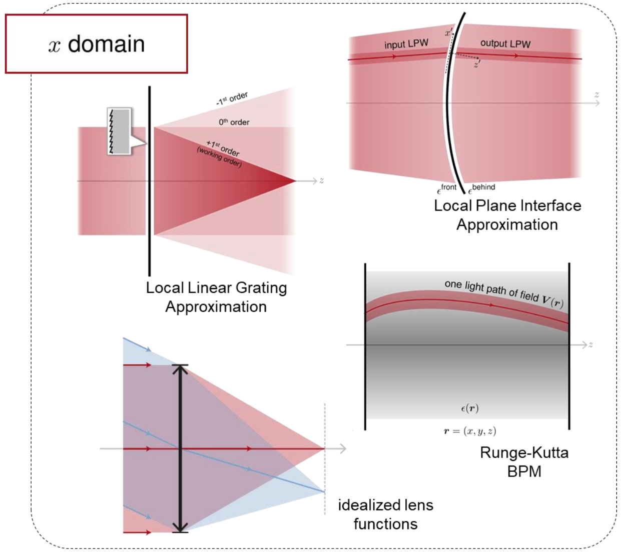
$$\tilde{V}_\ell^{\mathrm{out}}(\boldsymbol{\kappa}, z) = \tilde{V}_\ell^{\mathrm{in}}(\boldsymbol{\kappa}, z_0) \times \mathrm{e}^{\mathrm{i}k_z(\kappa)\Delta z}$$



# Why Different Fourier Domains?

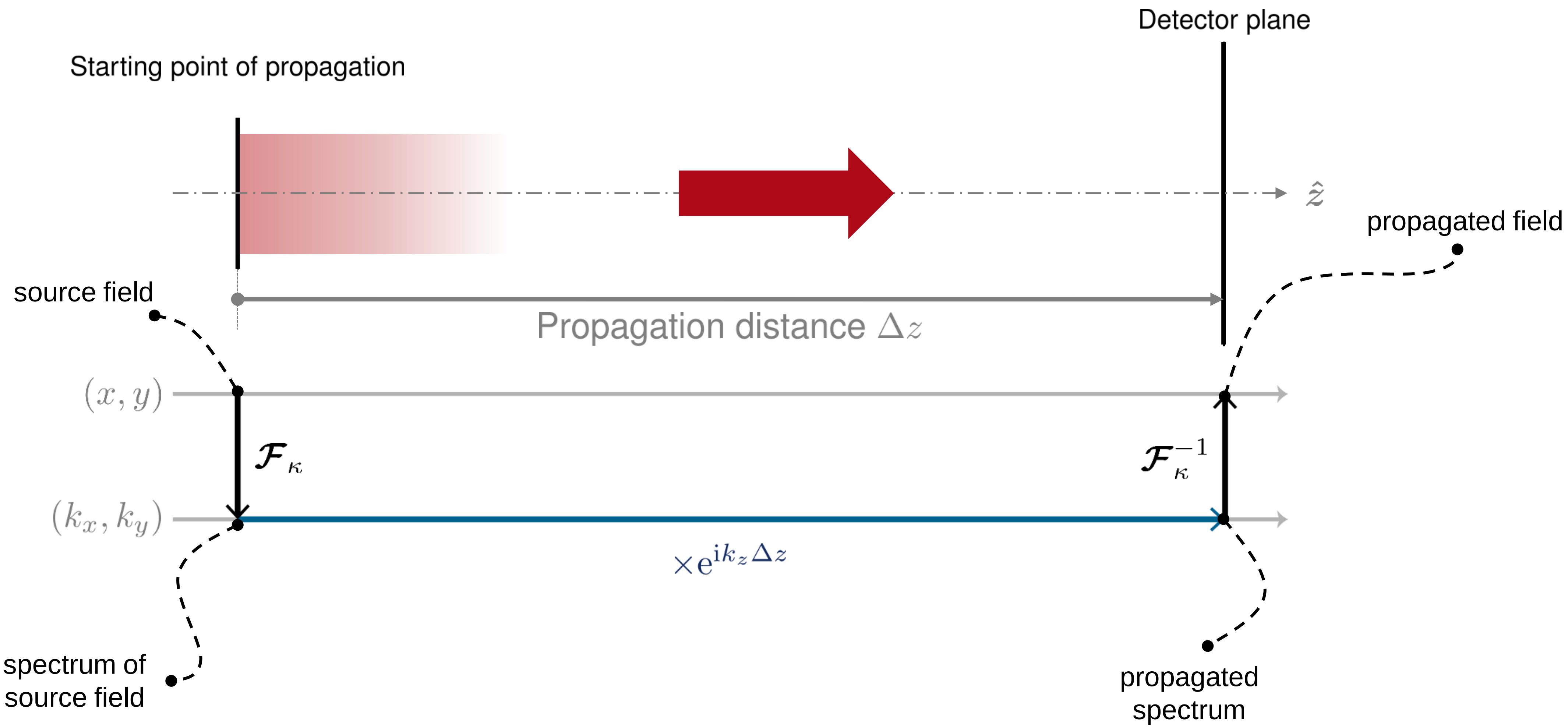
Although, in general, electromagnetic field solvers have an **integral** behaviour, with the resulting high numerical complexity, the characteristics of some of the most common optical components mean they can be modeled with **pointwise** operators in one of the Fourier domains – when this happens, it entails a **massive computational advantage**!



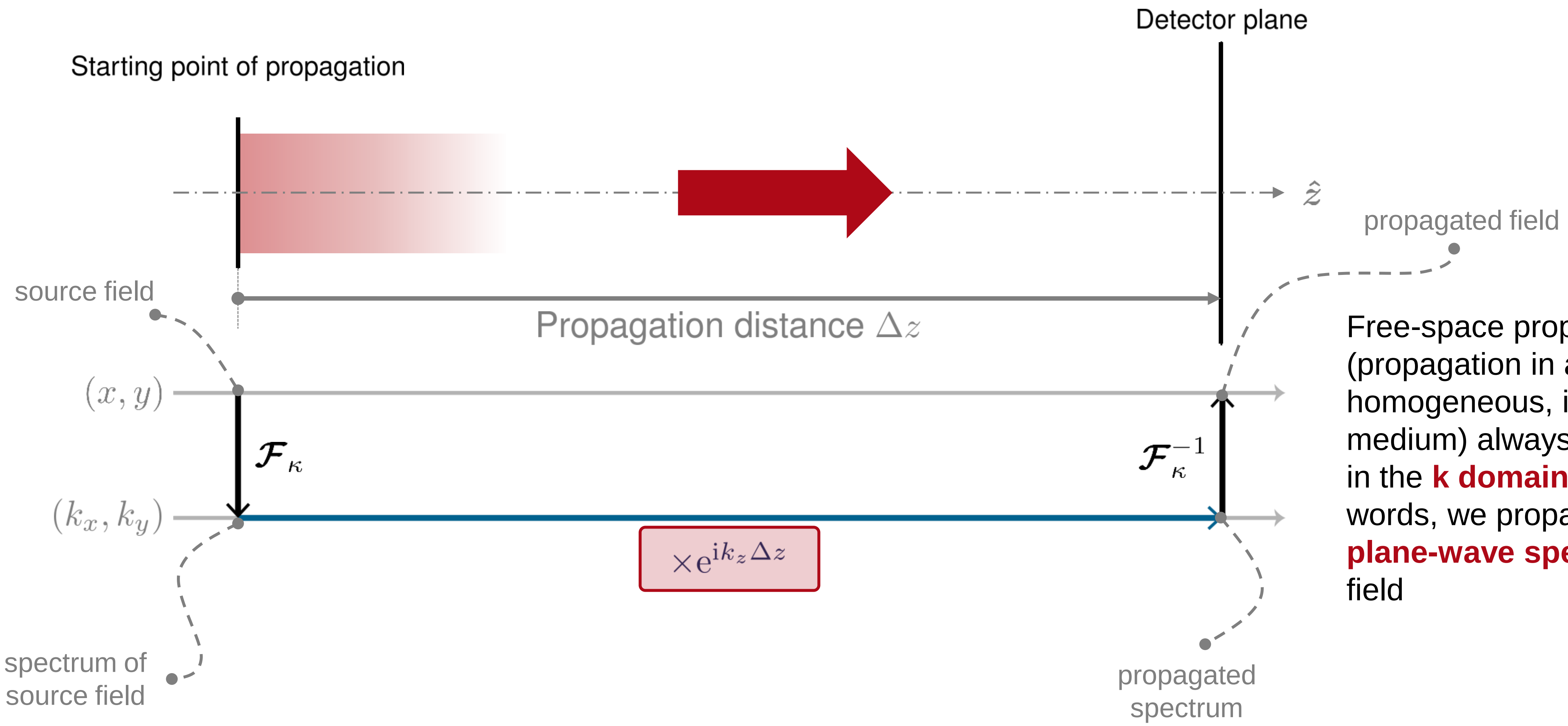




**Conclusion:** Whenever this mathematical property presents itself, we implement the solver in the domain where it exhibits **pointwise behaviour**!

# Free-Space Propagation

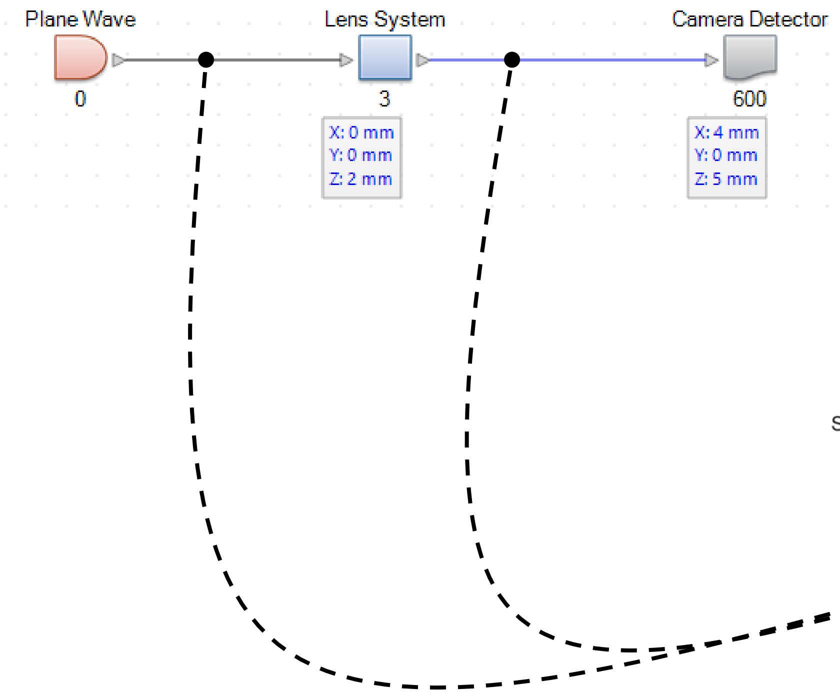


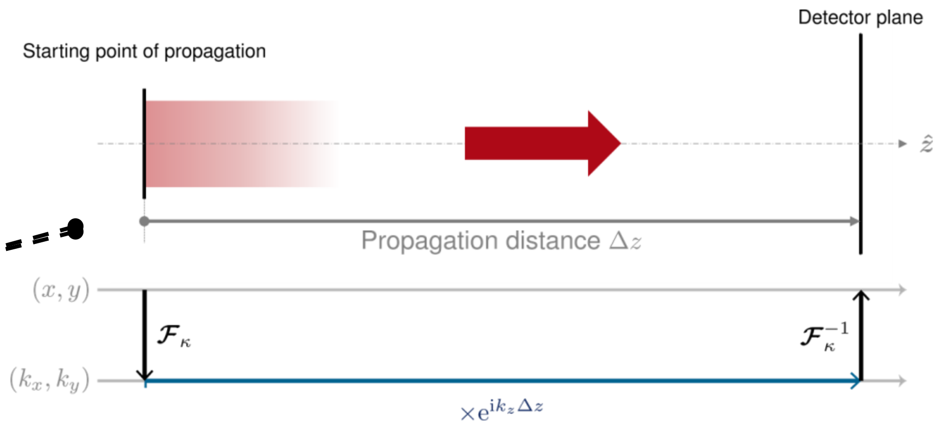
# Free-Space Propagation




Free-space propagation (propagation in a homogeneous, isotropic medium) always takes place in the **k domain** – in other words, we propagate the **plane-wave spectrum** of the field

# Free-Space Propagation

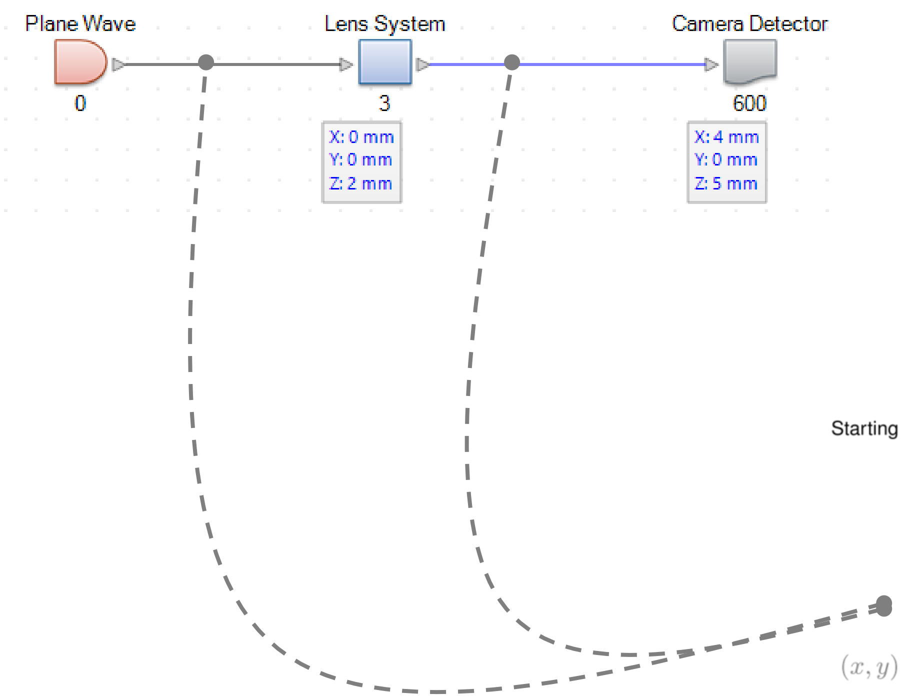


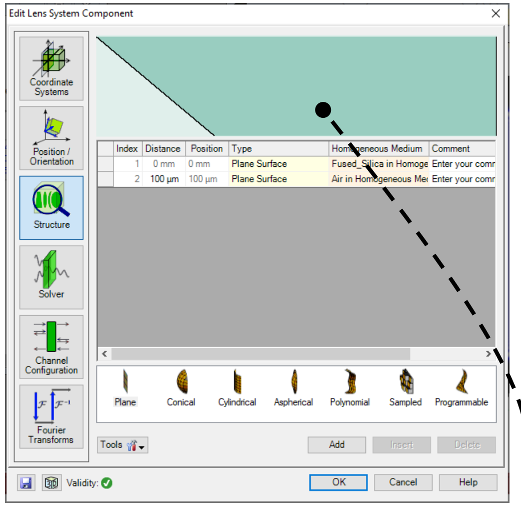




This refers to propagation between elements of an optical system…

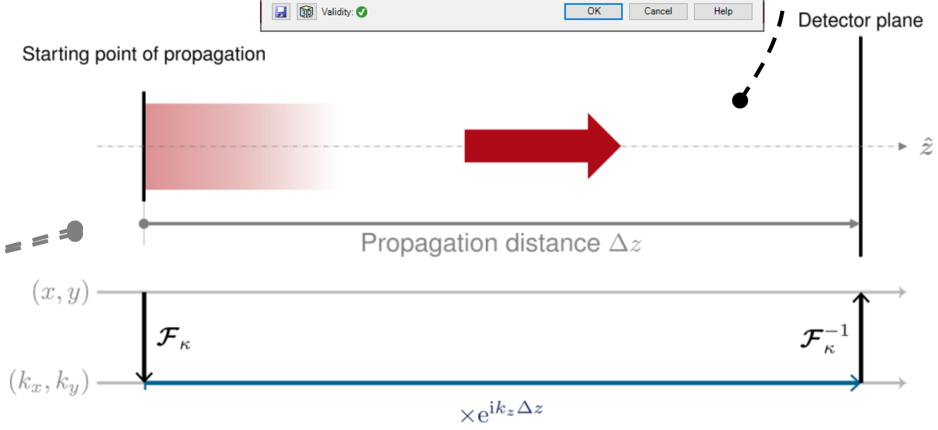
Free-space propagation (propagation in a homogeneous, isotropic medium) always takes place in the **k domain** – in other words, we propagate the **plane-wave spectrum** of the field

# Free-Space Propagation



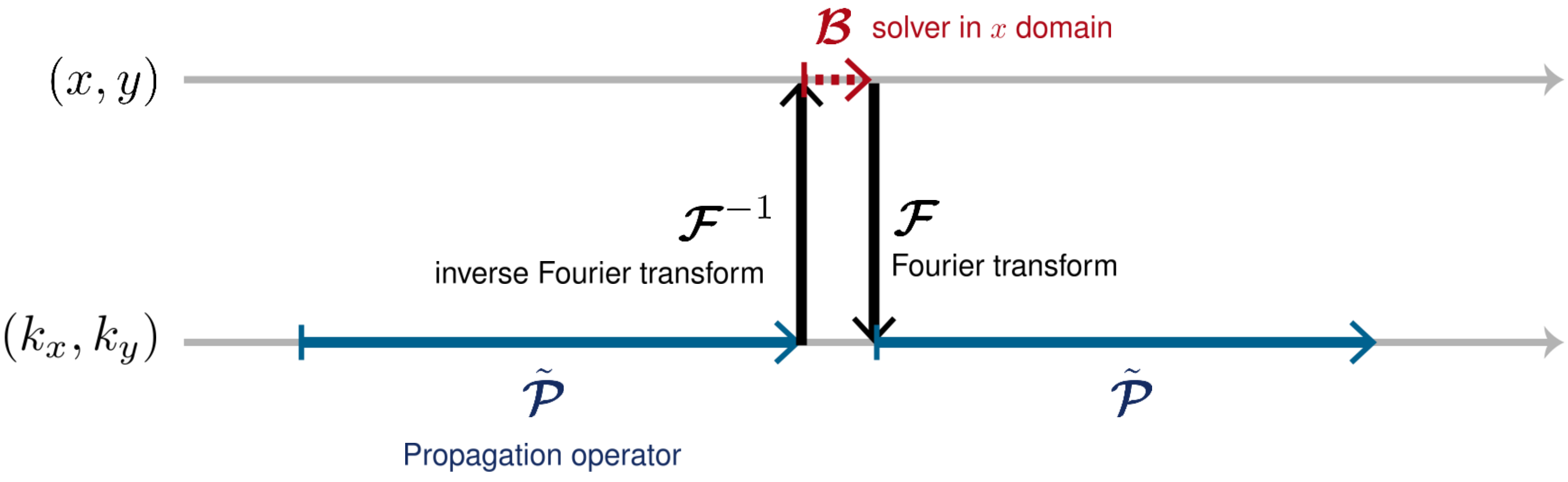

… as well as to internal propagation between surfaces in some cases, like the *Lens System* component




Free-space propagation (propagation in a homogeneous, isotropic medium) always takes place in the **k domain** – in other words, we propagate the **plane-wave spectrum** of the field

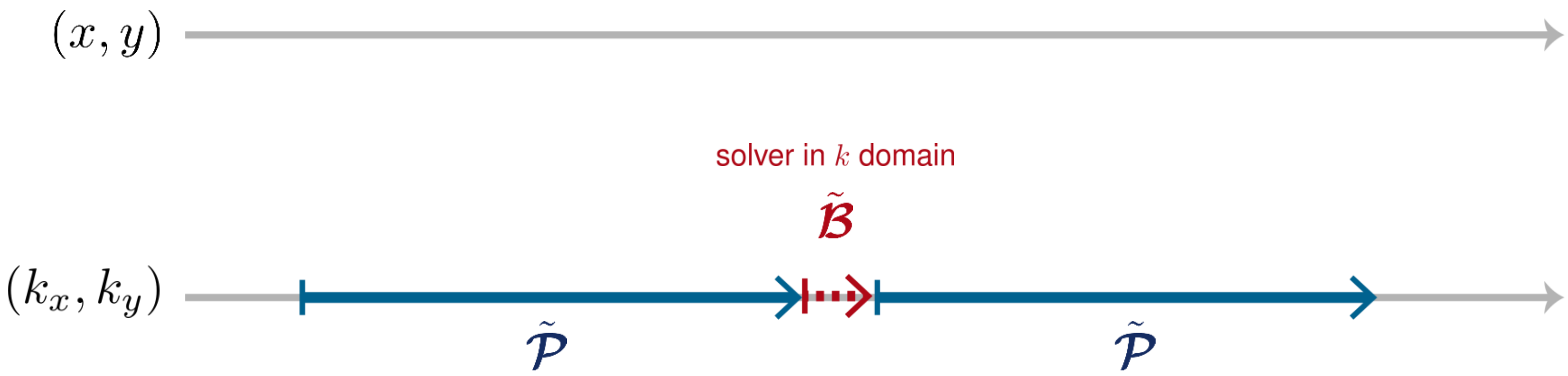
This refers to propagation between elements of an optical system…

# Domain of Application of the Solvers

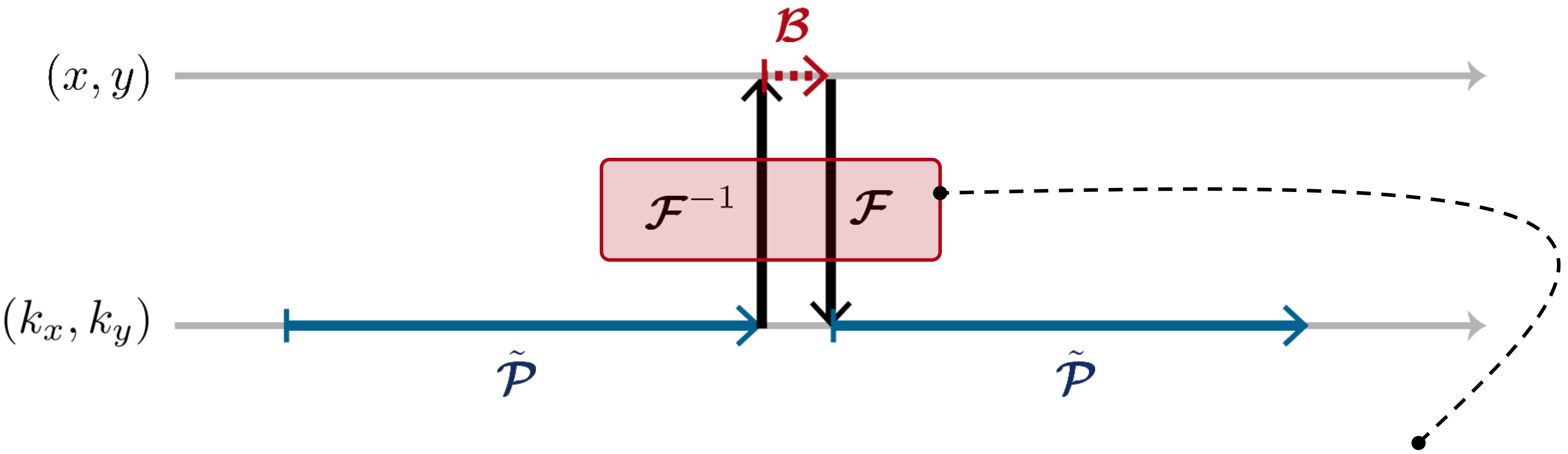



For solvers implemented in the space domain, the field-tracing sequence would look like this…

… while for solvers implemented in the k domain, the field-tracing sequence would look like this



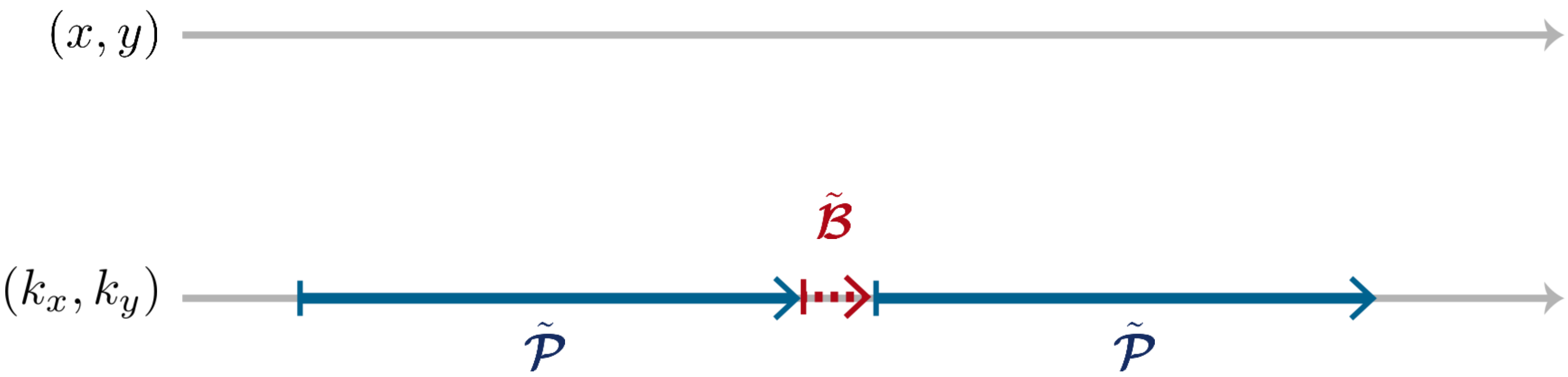
# Domain of Application of the Solvers




For solvers implemented in the space domain, the field-tracing sequence would look like this…

additional Fourier transform operations!

… while for solvers implemented in the k domain, the field-tracing sequence would look like this

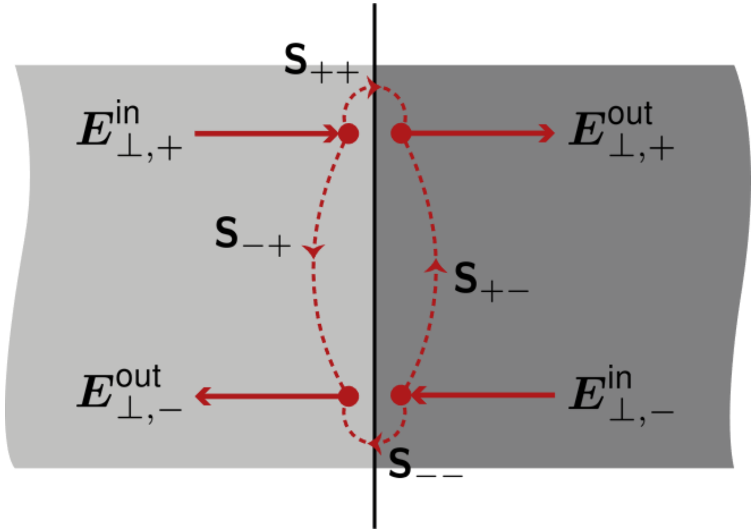


# The Special Case of the Plane Surface

# Possible Field Solvers for Plane Surfaces

- As an infinitely extended ideal plane surface ➔ **Fresnel Matrix**

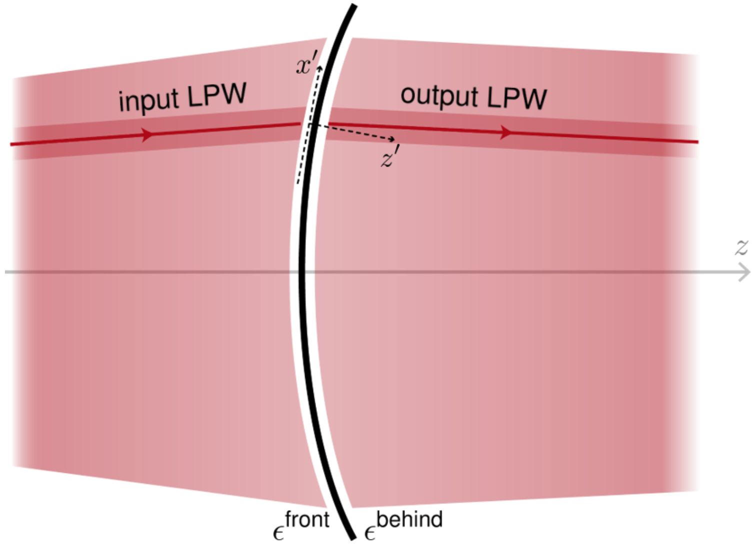
- Solver in **k domain**




- Implemented in *Plane Surface* component

- As a curved surface without curvature ➔ **Local Plane Interface Approximation (LPIA)**

- Solver in **x domain**

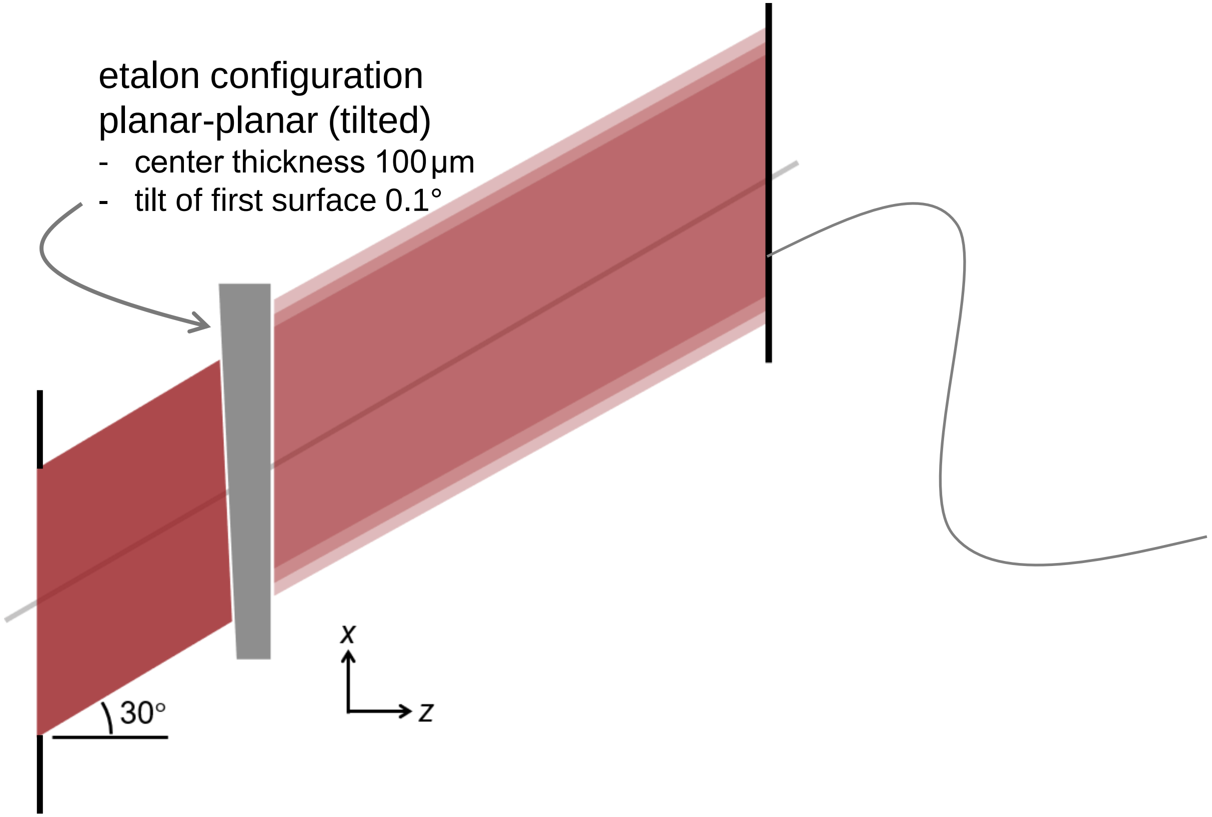



- Implemented in *Lens System* and *Curved Surface* components, among others

# Modeling an Etalon with Plane Interfaces…

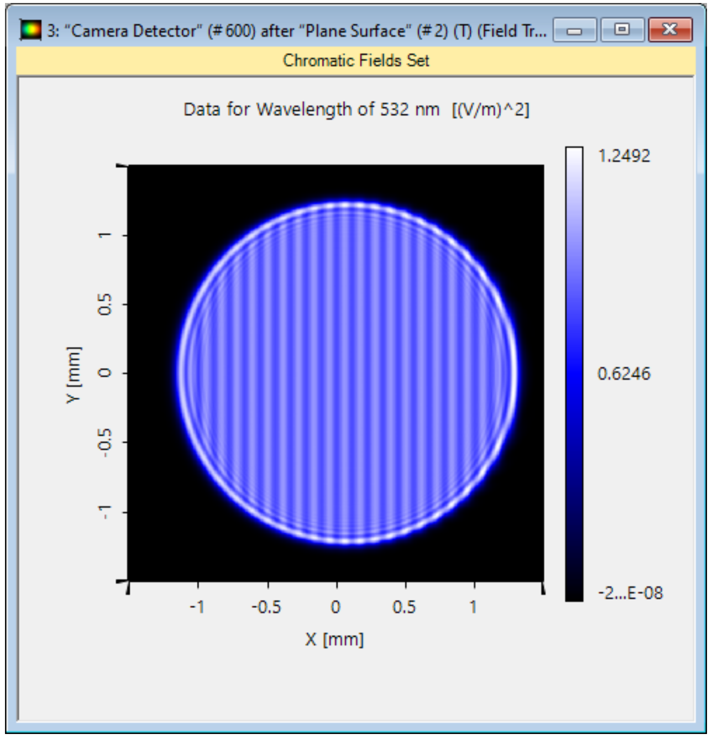
etalon configuration
planar-planar (tilted)

- center thickness 100 μm
- tilt of first surface 0.1°



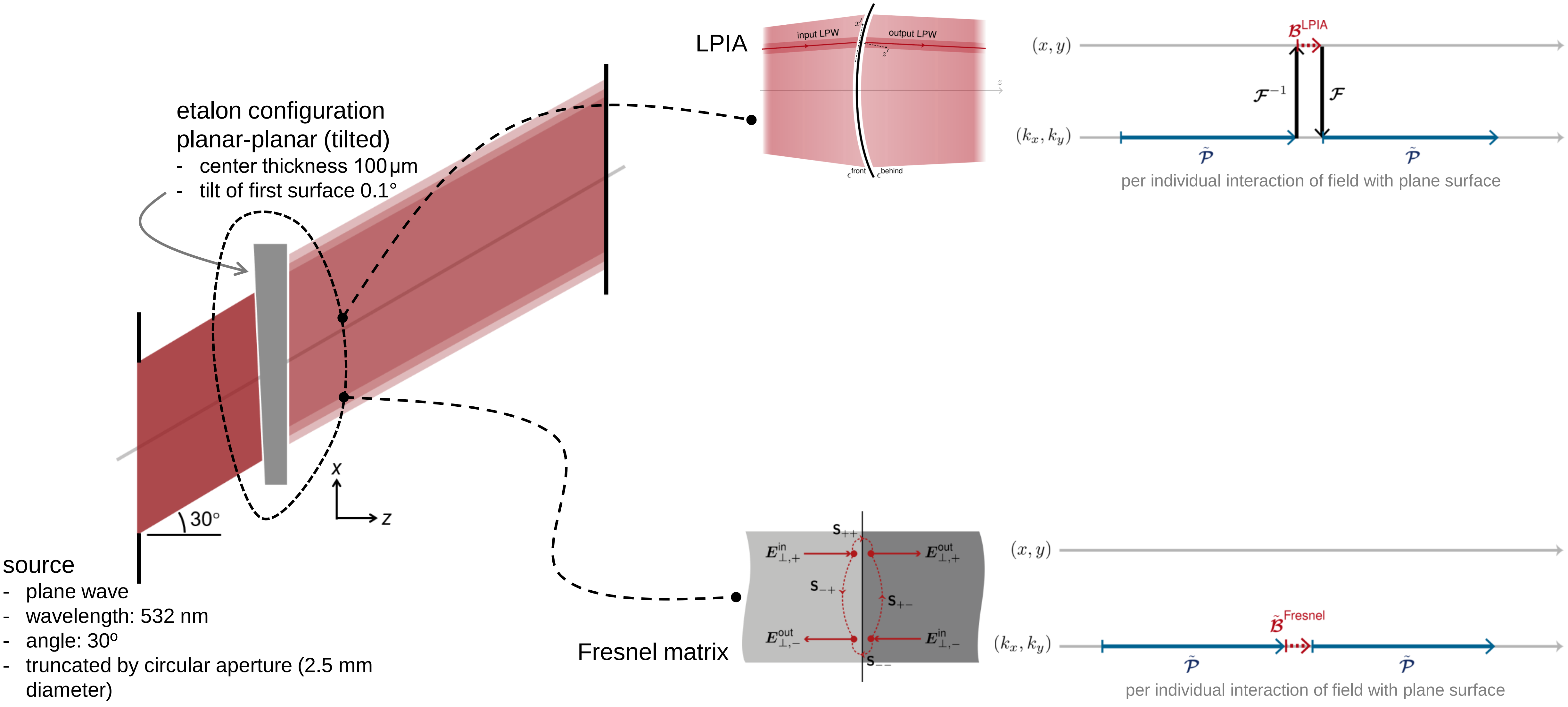

source

- plane wave
- wavelength: 532 nm
- angle: 30º
- truncated by circular aperture (2.5 mm diameter)




see the full Application Use Case:
*"Modeling of Etalon with Planar or Curved Surfaces"*

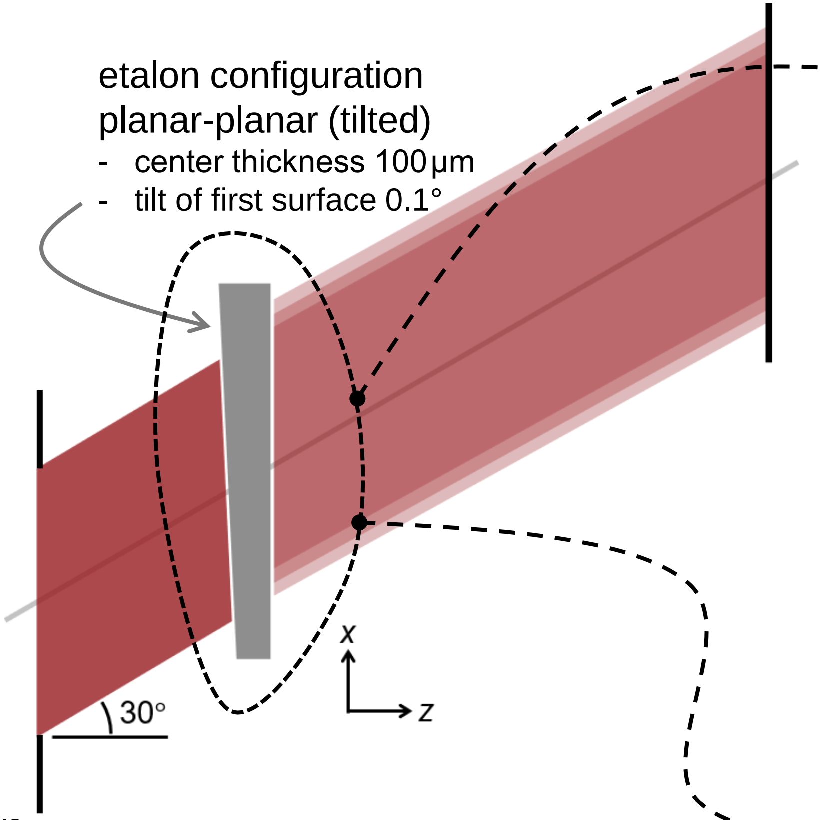
# … Two Ways



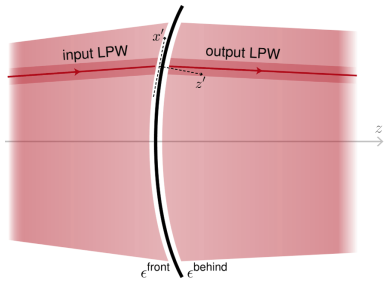
etalon configuration
planar-planar (tilted)
- center thickness 100 µm
- tilt of first surface 0.1°
x
z
30°
source
- plane wave
- wavelength: 532 nm
- angle: 30º
- truncated by circular aperture (2.5 mm diameter)
LPIA
input LPW
output LPW
front
behind
$(x, y)$
$(k_x, k_y)$
$\mathcal{B}^{\text{LPIA}}$
$\mathcal{F}^{-1}$
$\mathcal{F}$
$\tilde{\mathcal{P}}$
$\tilde{\mathcal{P}}$
per individual interaction of field with plane surface
Fresnel matrix
$S_{++}$
$S_{-+}$
$S_{+-}$
$S_{--}$
$E^{\text{in}}_{\perp,+}$
$E^{\text{out}}_{\perp,+}$
$E^{\text{out}}_{\perp,-}$
$E^{\text{in}}_{\perp,-}$
$(x, y)$
$(k_x, k_y)$
$\tilde{\mathcal{B}}^{\text{Fresnel}}$
$\tilde{\mathcal{P}}$
$\tilde{\mathcal{P}}$
per individual interaction of field with plane surface

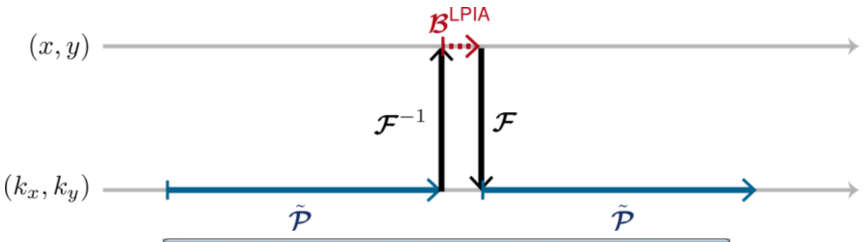
# … Two Ways

etalon configuration
planar-planar (tilted)
- center thickness 100 µm
- tilt of first surface 0.1°

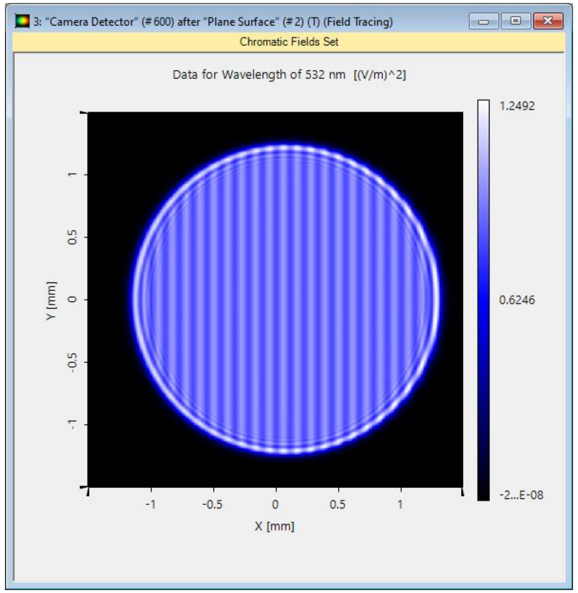



LPIA

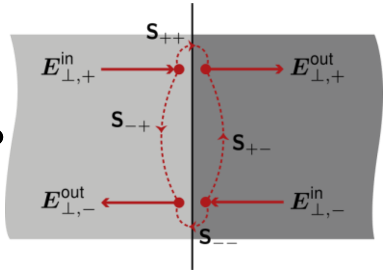


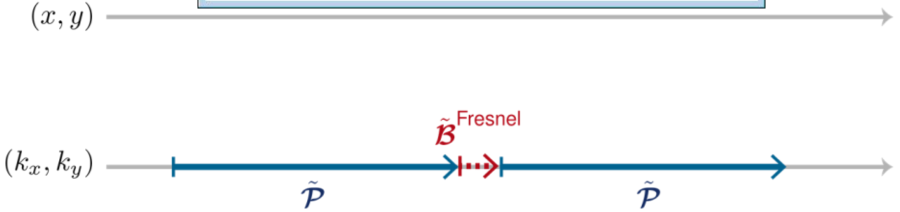

$\Delta t_{\text{simulation}} \sim 1.5\,\text{min}$ (*)




$\Delta t_{\text{simulation}} \sim 10\,\text{s}$ (*)



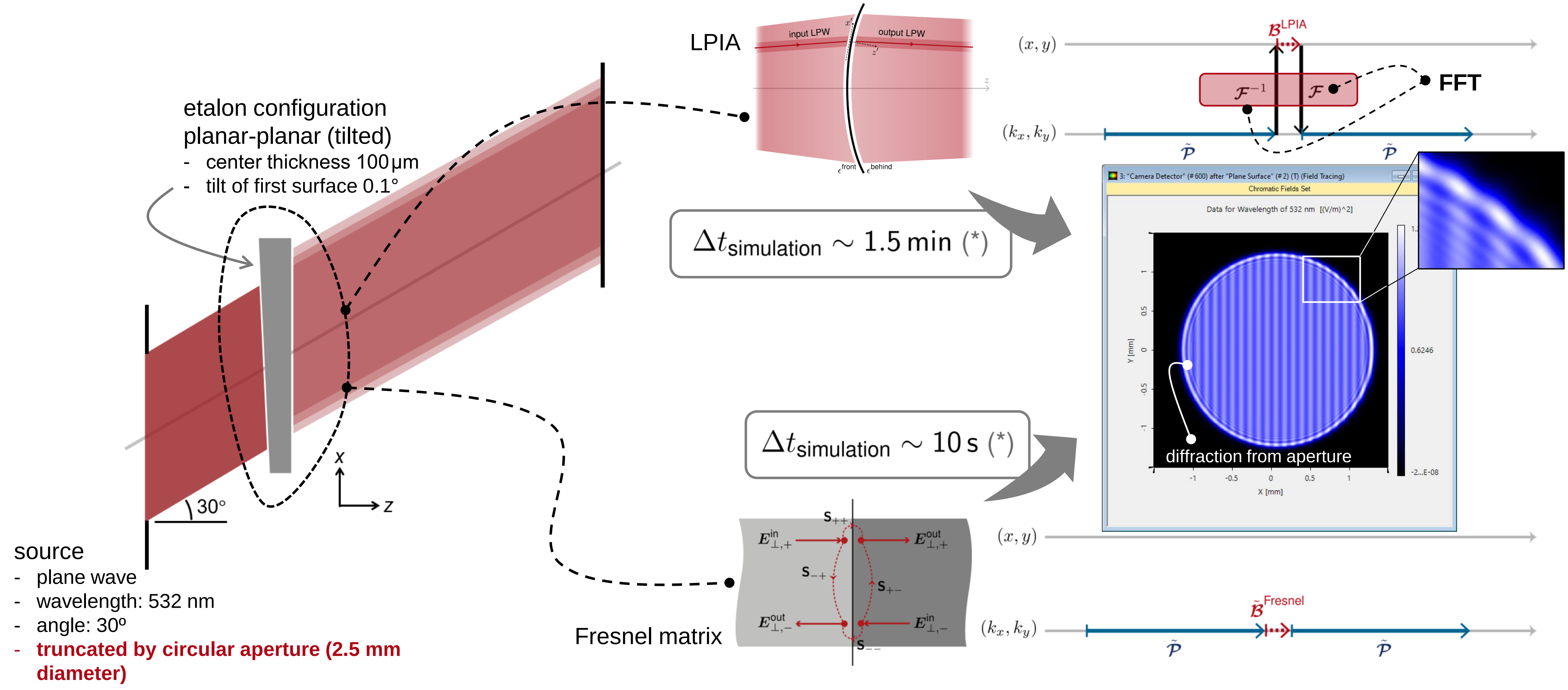

Fresnel matrix




source
- plane wave
- wavelength: 532 nm
- angle: 30º
- truncated by circular aperture (2.5 mm diameter)

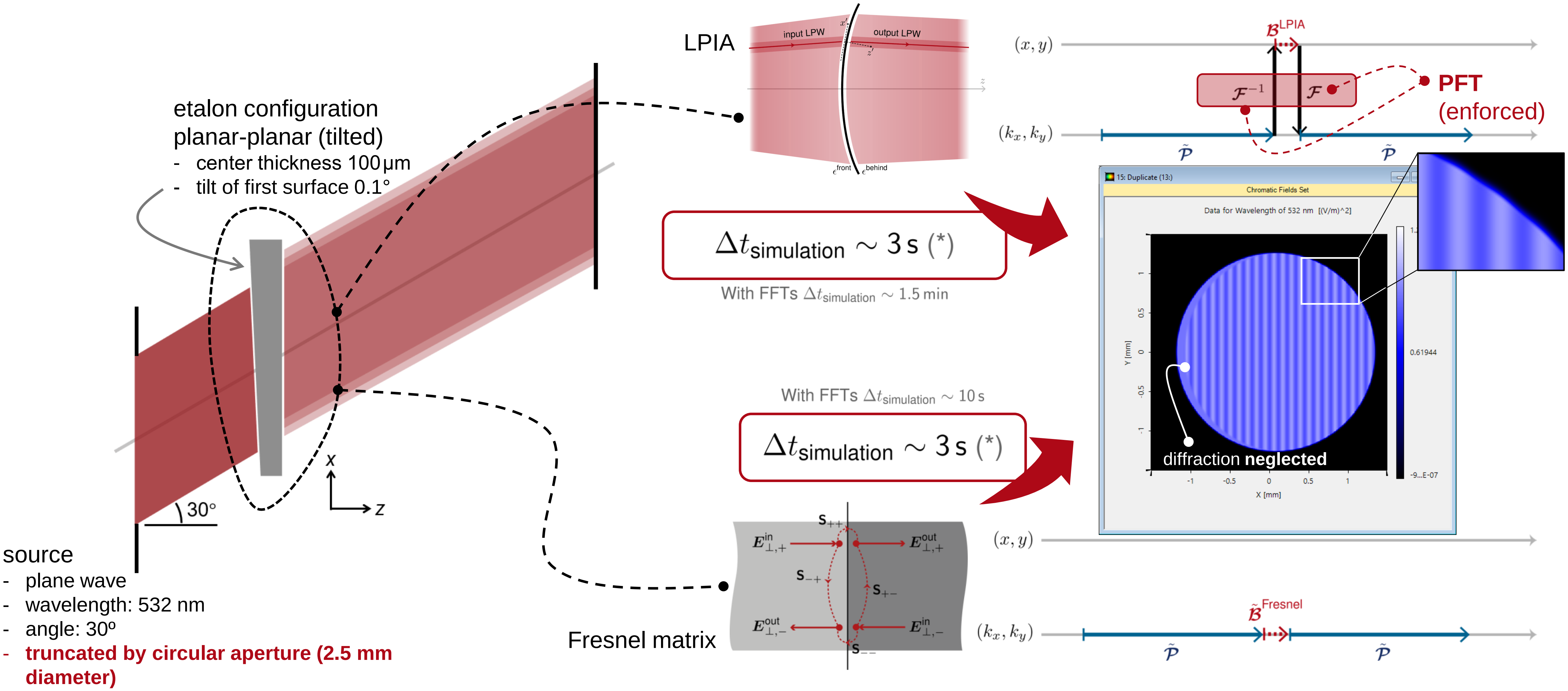
(*) Approximate simulation times on a PC with the following technical specs: processor Intel(R) Core(TM) i7-7700HQ CPU @ 2.80GHz and 32 GB RAM

# The Importance of the Fourier Transform




(*) Approximate simulation times on a PC with the following technical specs: processor Intel(R) Core(TM) i7-7700HQ CPU @ 2.80GHz and 32 GB RAM

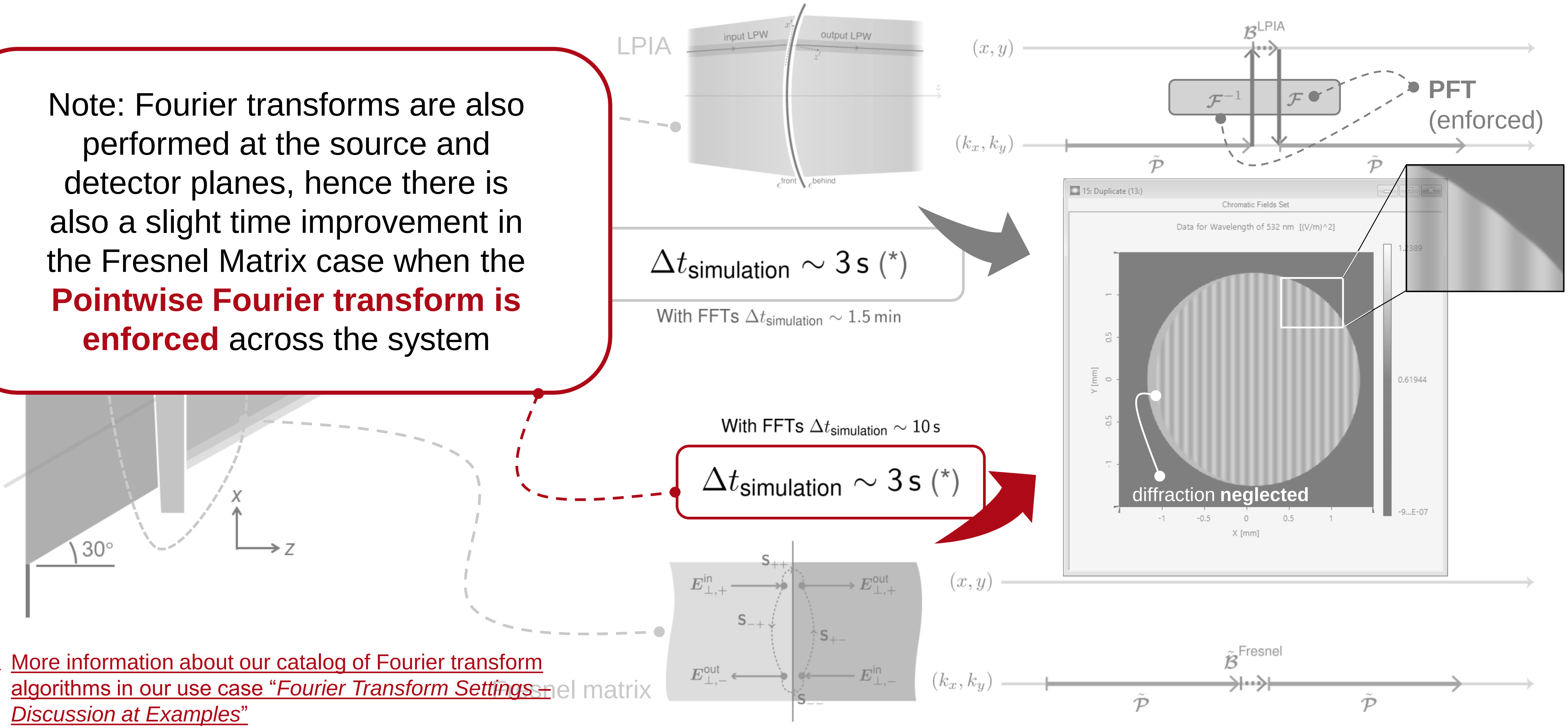
# The Importance of the Fourier Transform




(*) Approximate simulation times on a PC with the following technical specs: processor Intel(R) Core(TM) i7-7700HQ CPU @ 2.80GHz and 32 GB RAM

# The Importance of the Fourier Transform

Note: Fourier transforms are also performed at the source and detector planes, hence there is also a slight time improvement in the Fresnel Matrix case when the **Pointwise Fourier transform is enforced** across the system




[More information about our catalog of Fourier transform algorithms in our use case "*Fourier Transform Settings – Discussion at Examples*"]

# Practical Conclusions: Which Solver Do I Use?

Two possible solvers for plane interfaces in an optical system: the Fresnel Matrix and the Local Plane Interface Approximation (LPIA). Which one is more appropriate for your system depends on the circumstances:

**Fresnel Matrix:**

- Rigorous solver for ideal plane surface
- Works in spatial-frequency (k) domain
- Fewer Fourier transforms to be calculated ➔ potential numerical gain
- Assumes infinite surface

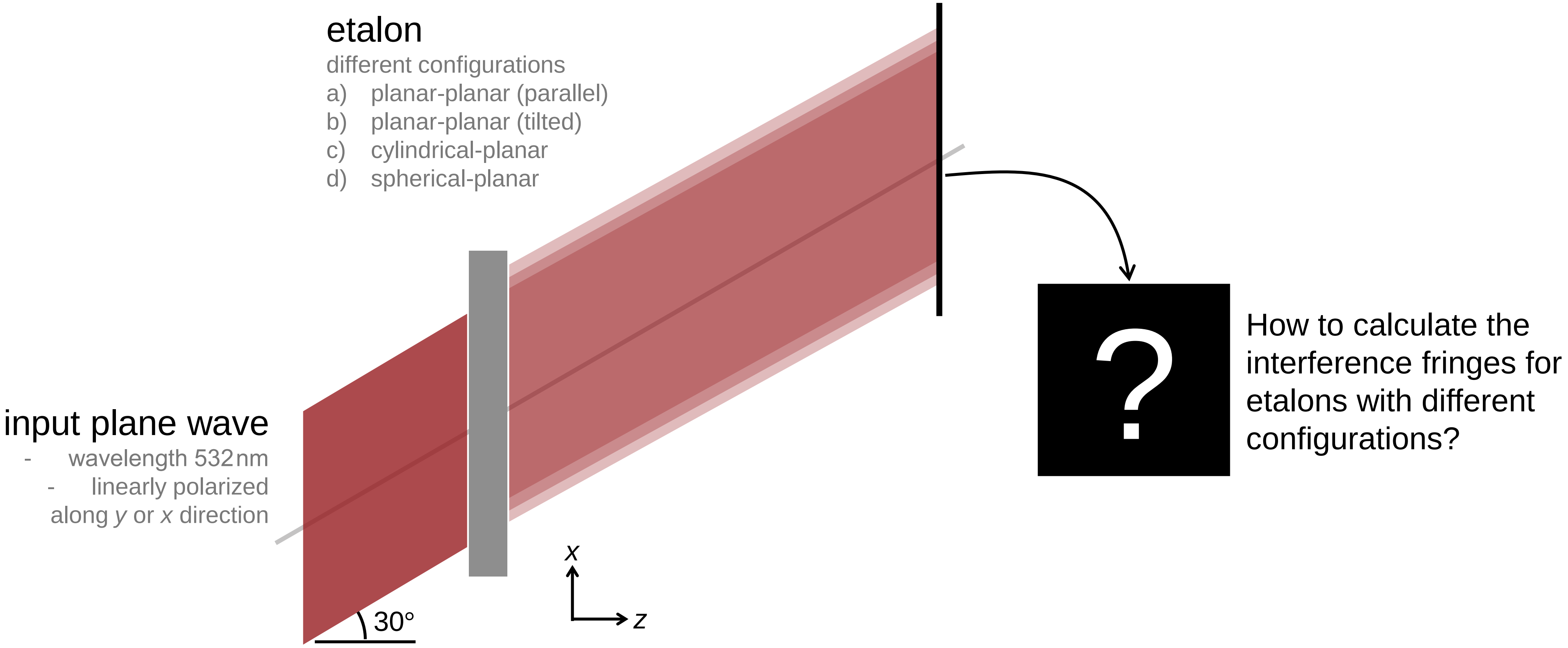
**LPIA:**

- Solver for curved surfaces
- Works in space (x) domain
- Requires computation of additional Fourier transforms
- Considers finite size (aperture) of surface

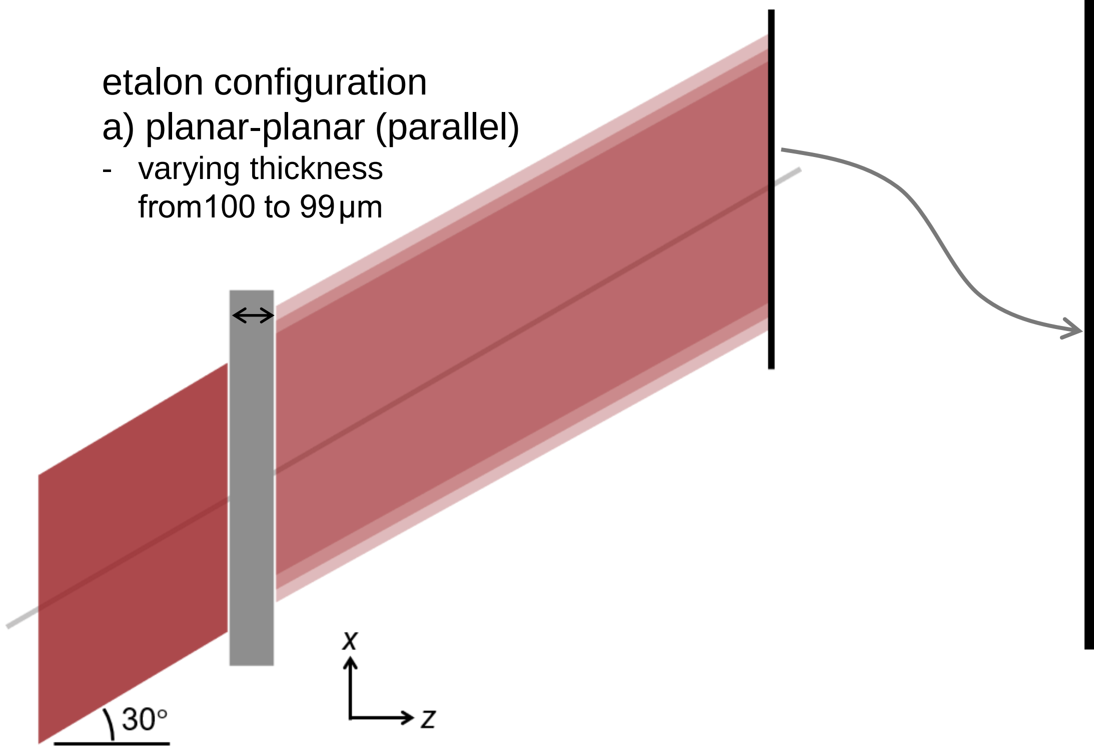
# Modeling of Etalon with Planar or Curved Surfaces

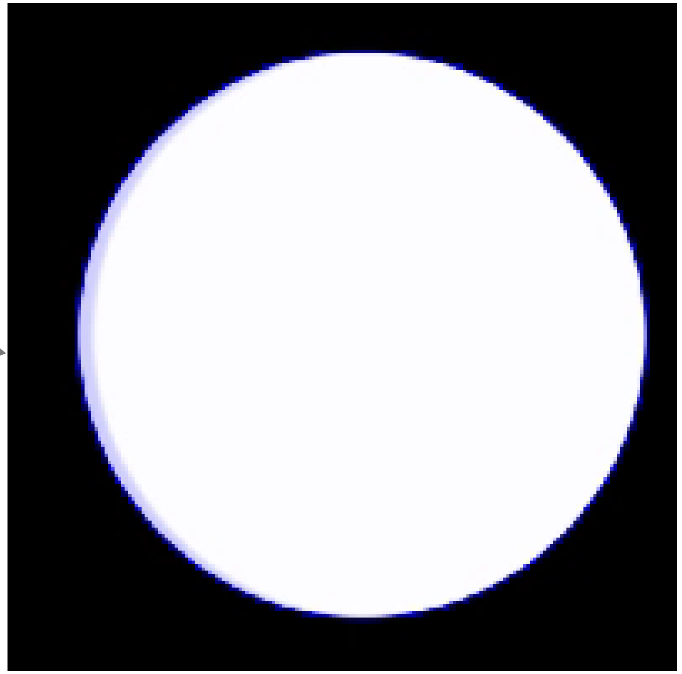
see the full Application Use Case

# Modeling Task



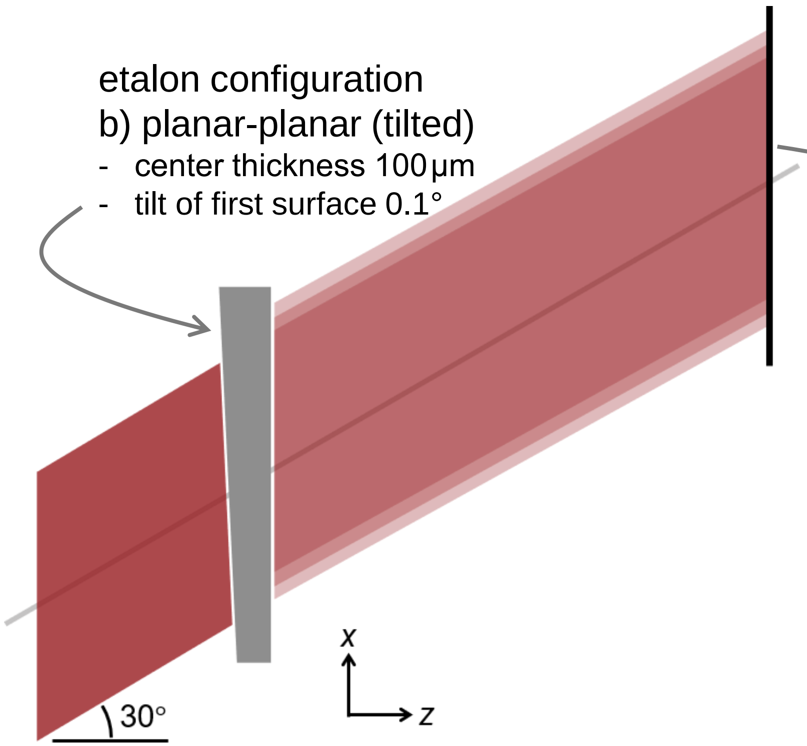
# Parallel Planar-Planar Surfaces

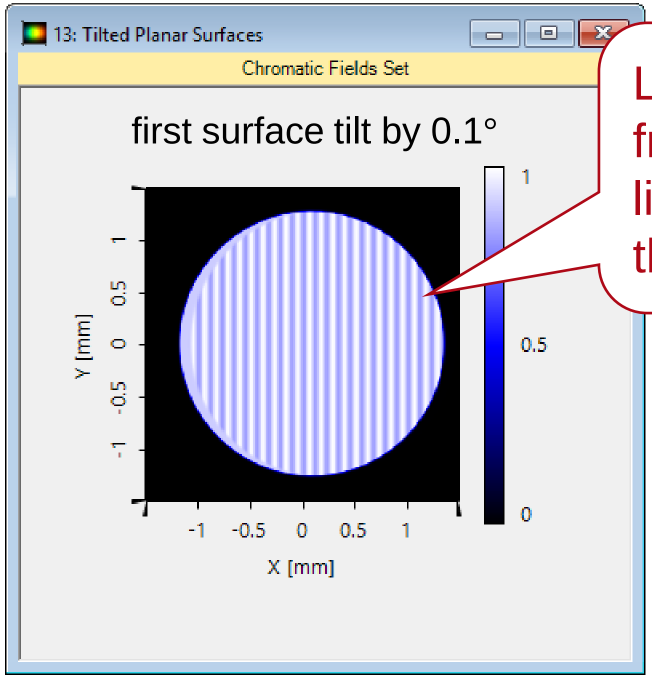


Constructive and destructive interference alternatively shows up when the thickness of etalon varies.

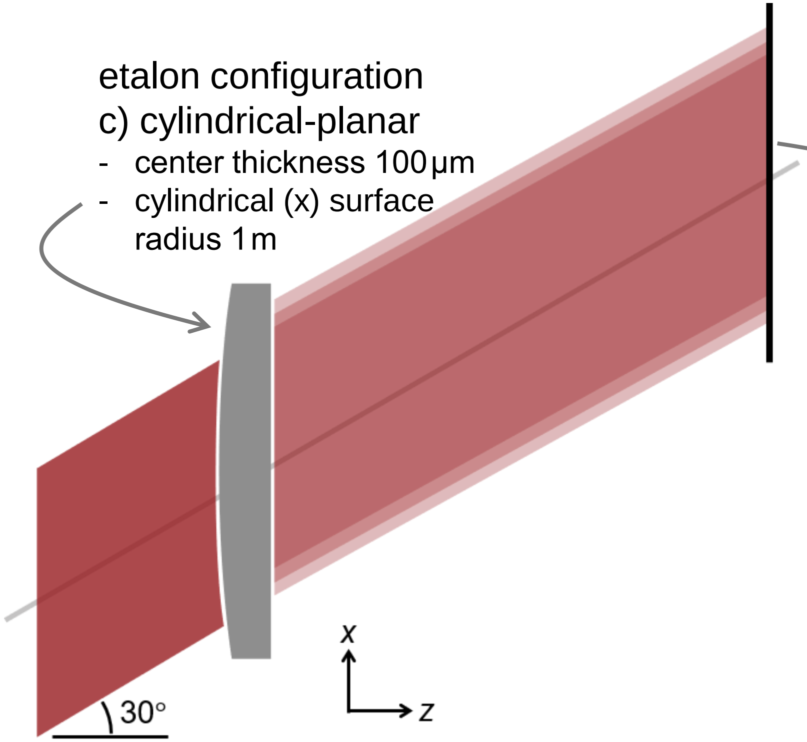
# Tilted Planar-Planar Surfaces

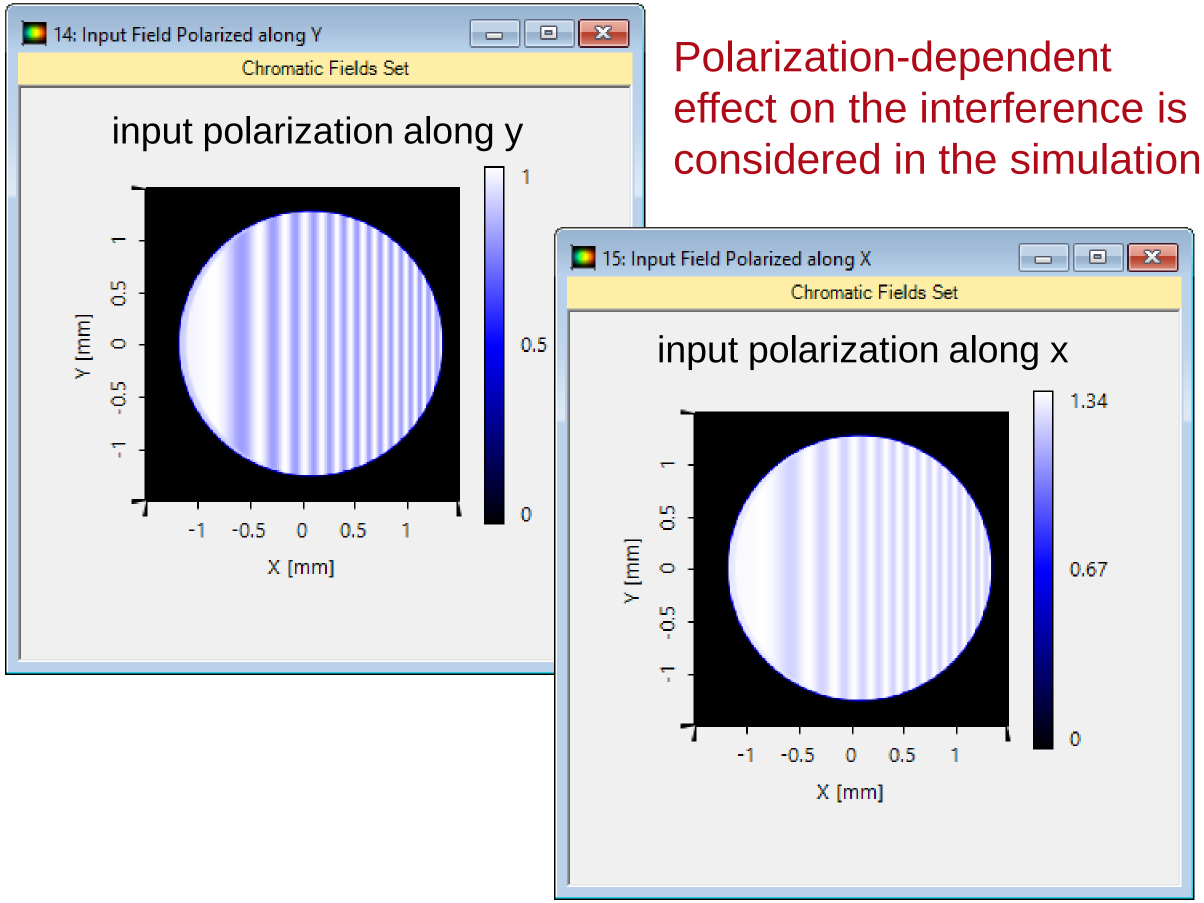



Linear interference fringes appear due to linear change of etalon thickness.

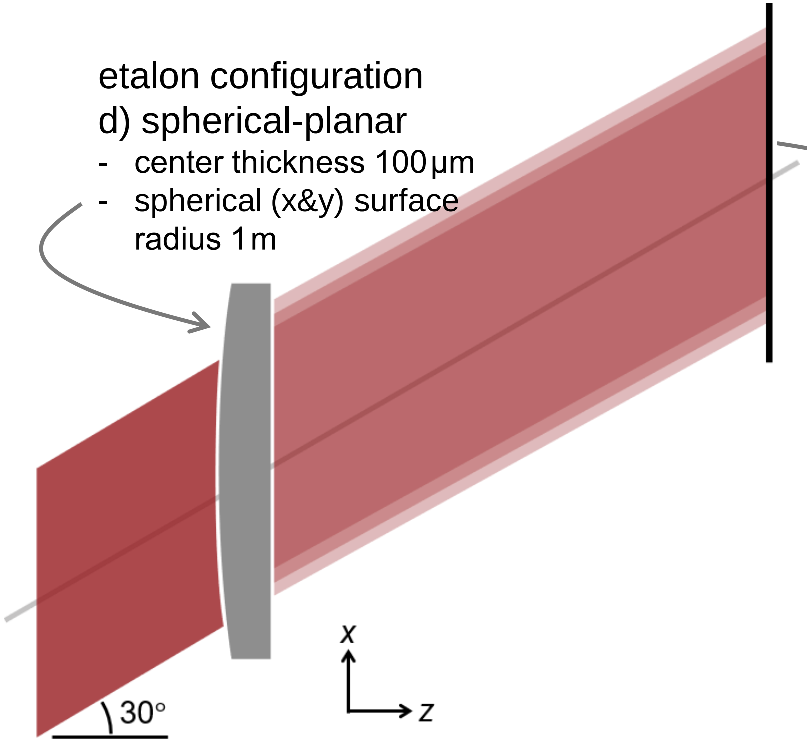
# Cylindrical-Planar Surfaces

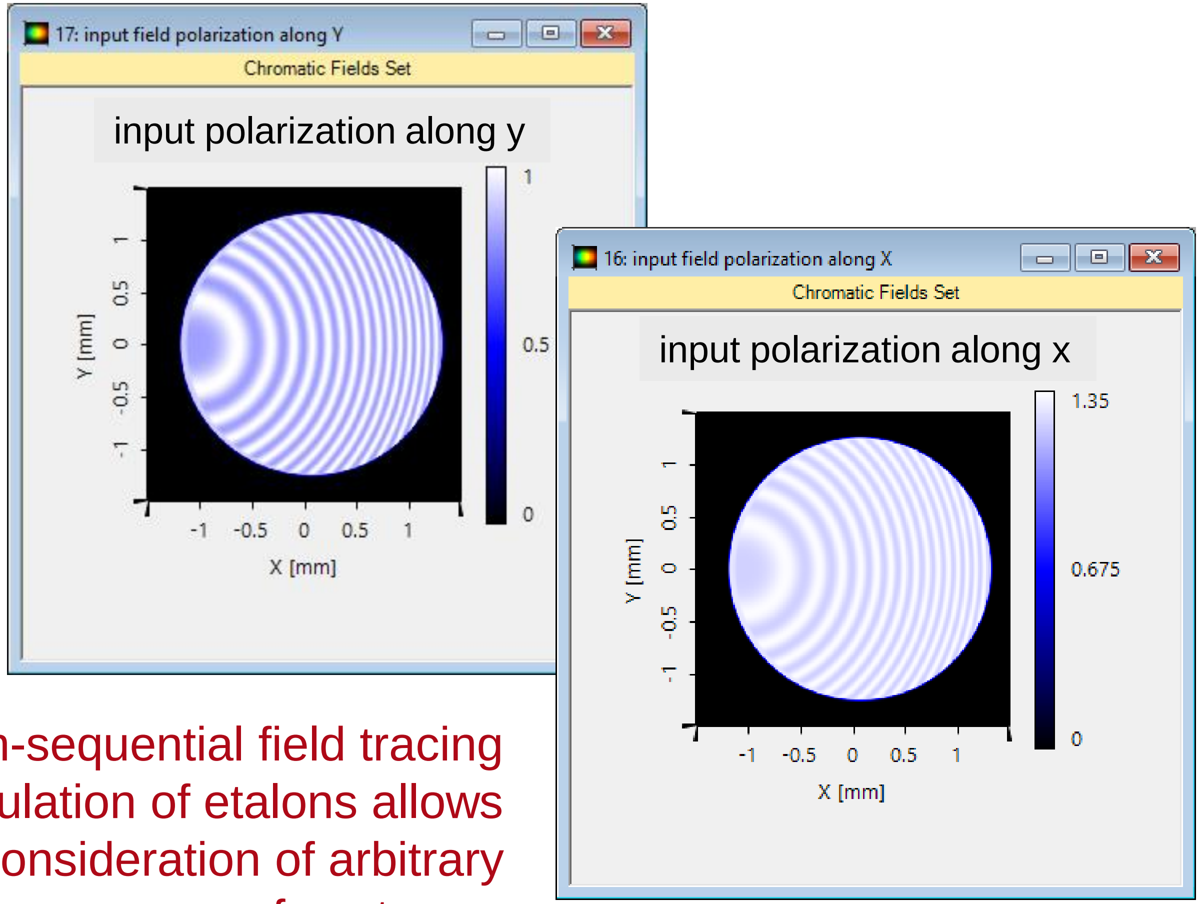



Polarization-dependent effect on the interference is considered in the simulation.

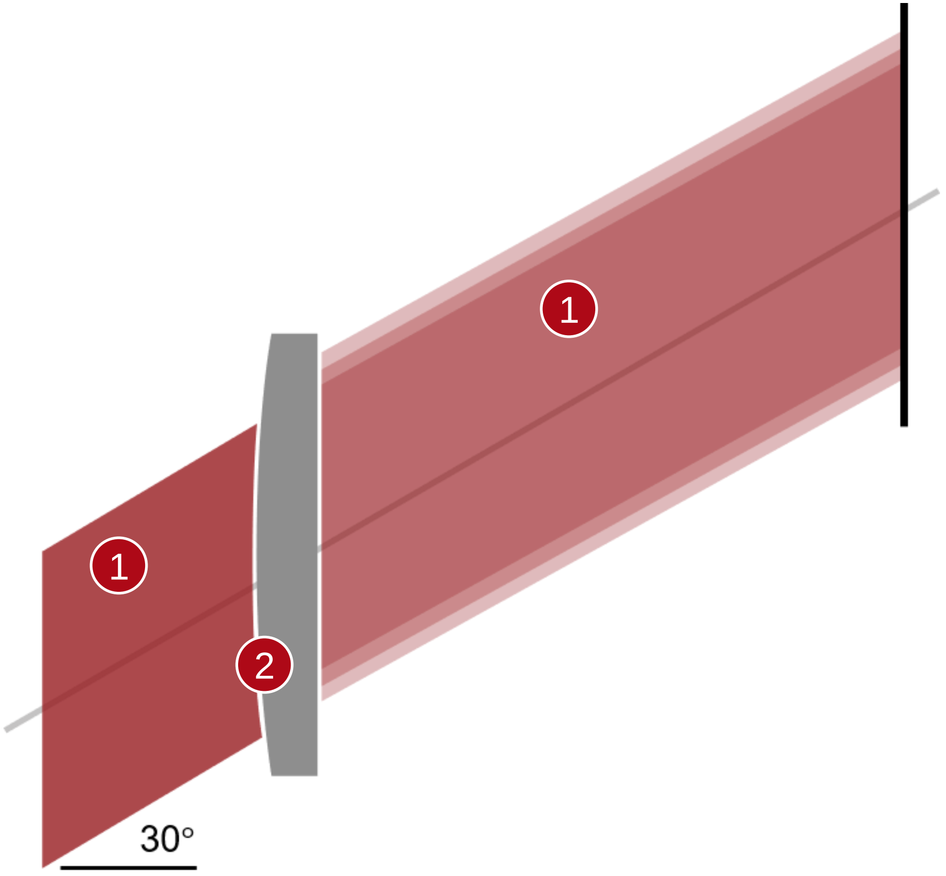
# Spherical-Planar Surfaces

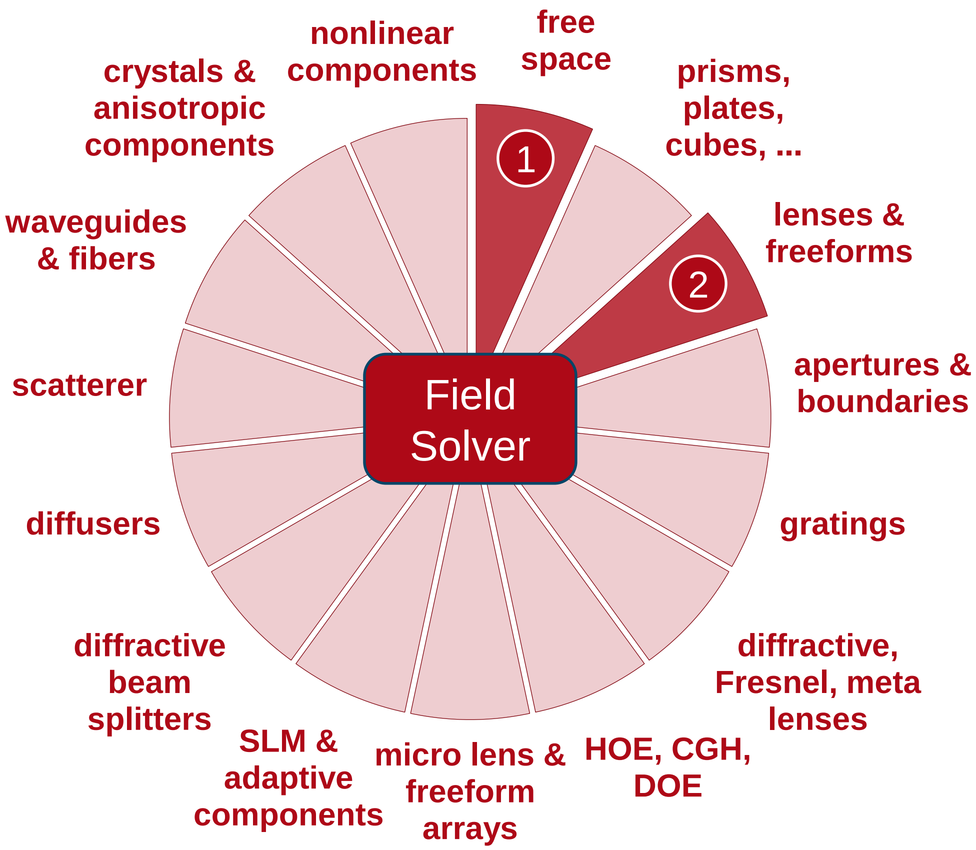



Non-sequential field tracing simulation of etalons allows the consideration of arbitrary surface types.

# VirtualLab Fusion Technologies

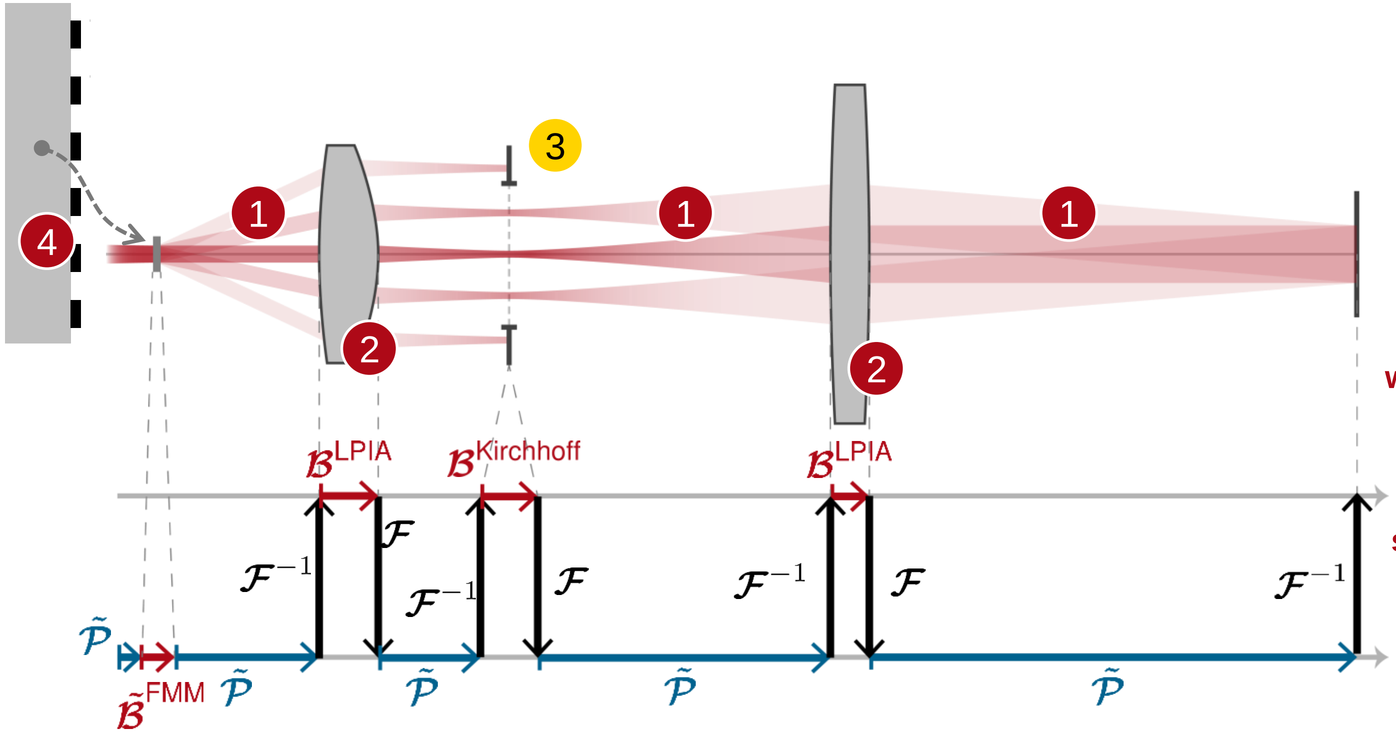


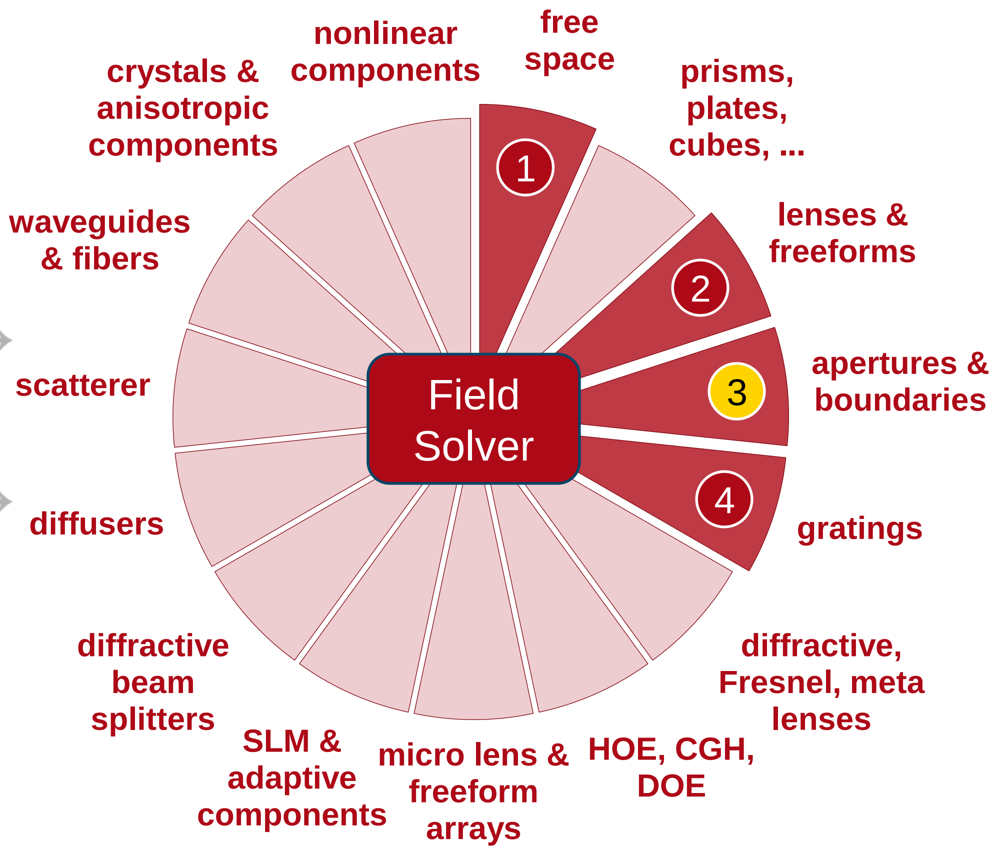



Part 3

# The Fourier Transform

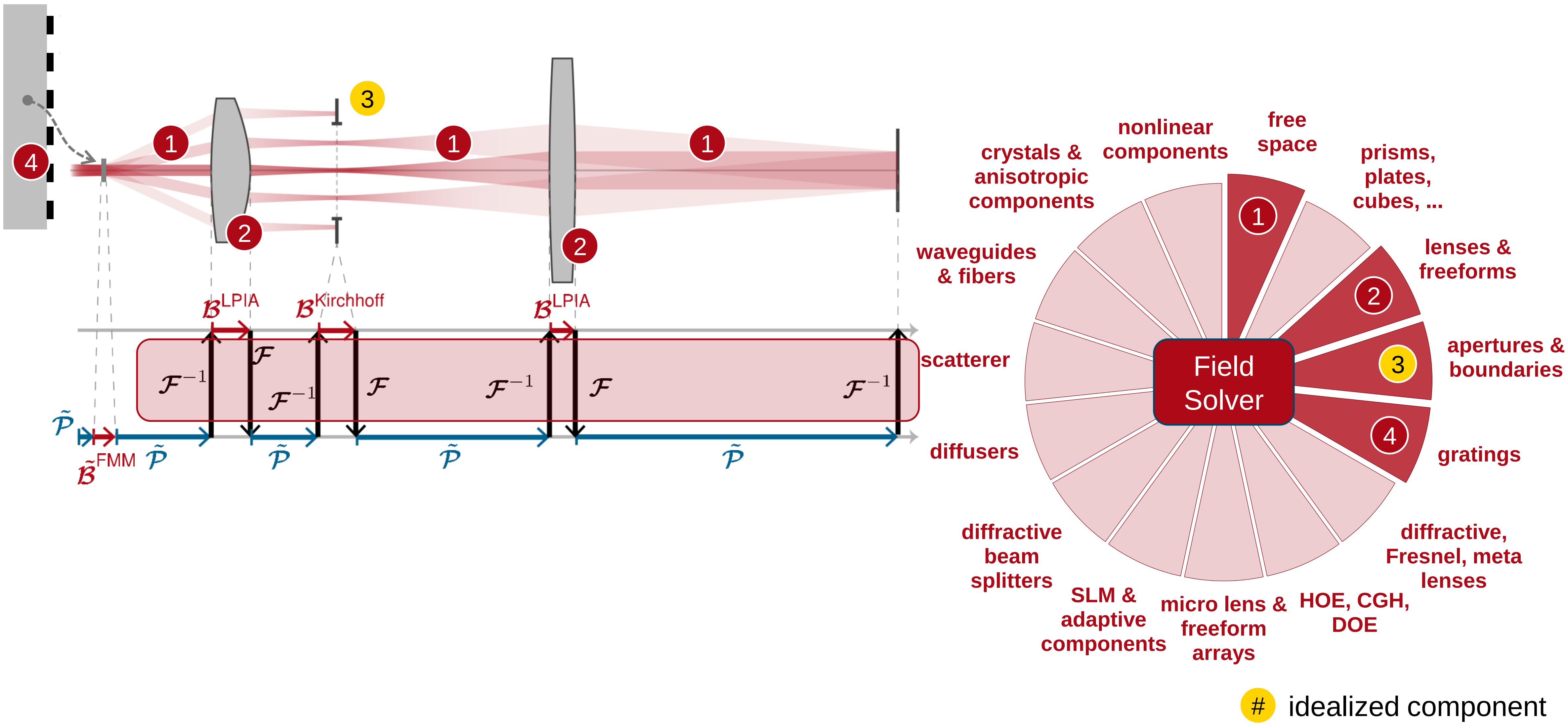
# Field Tracing Diagram



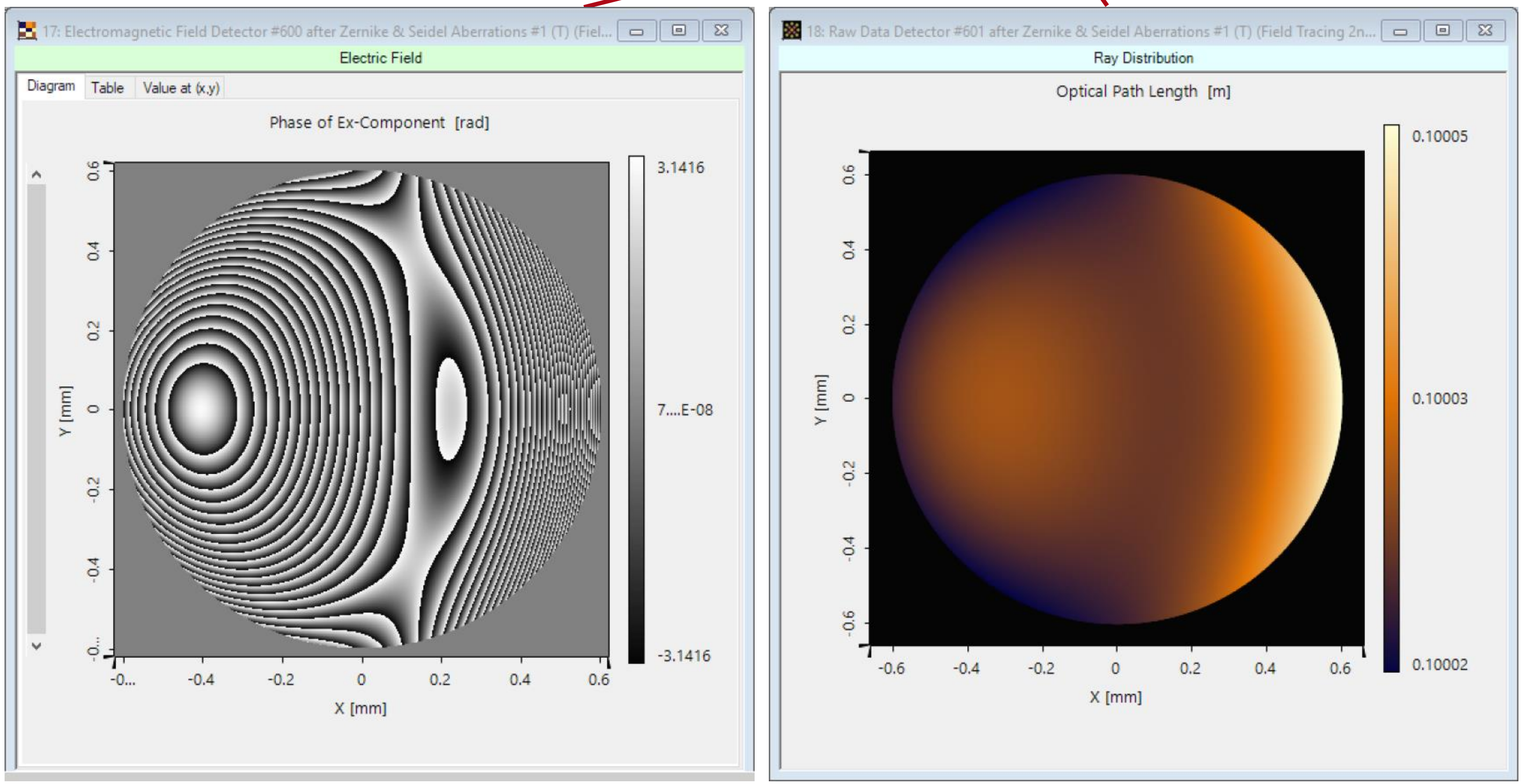

# idealized component

# Field Tracing Diagram



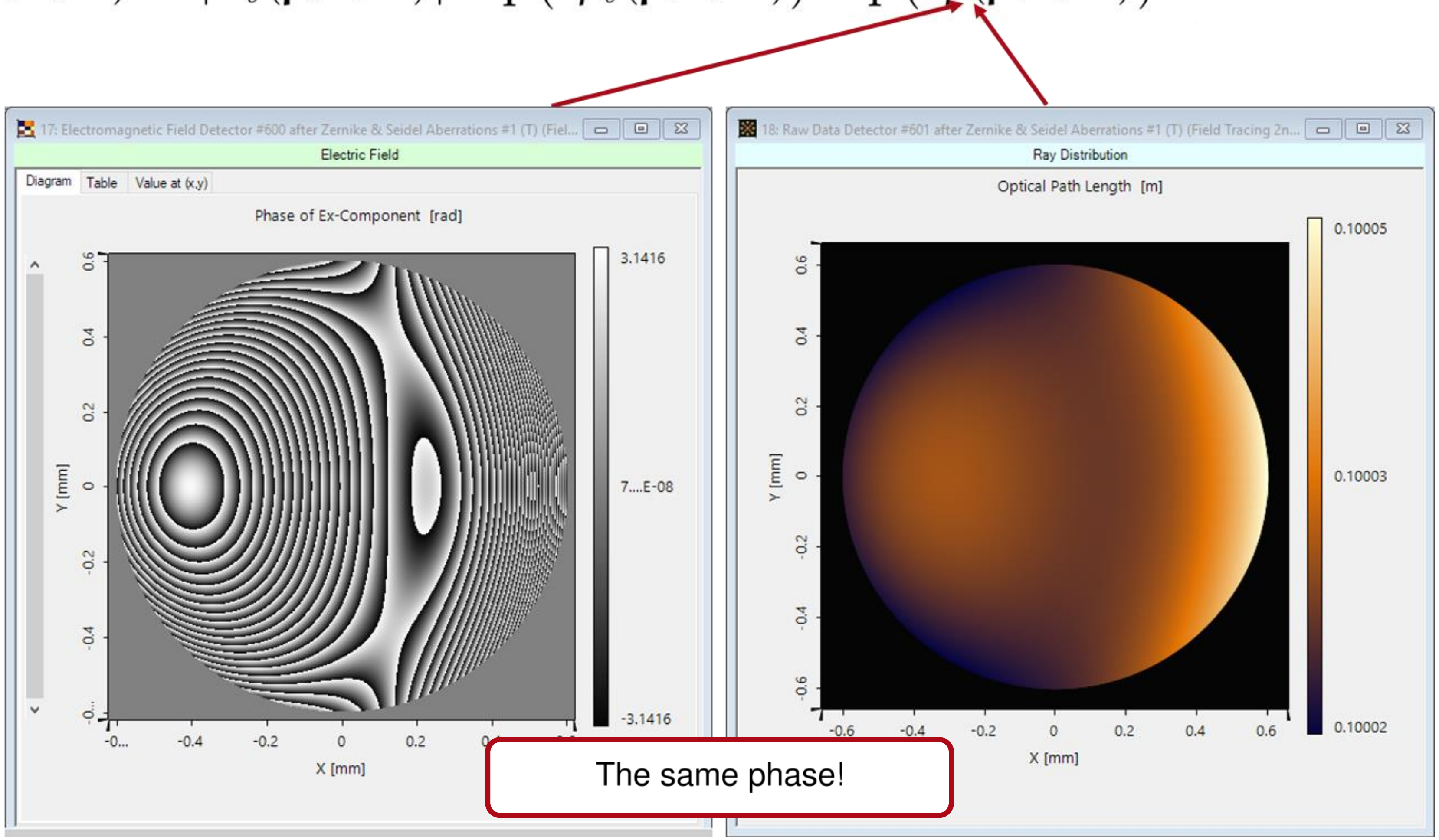
# Fourier-Transforming an Arbitrary Field

$$V_\ell(\boldsymbol{\rho},z,\omega) = |V_\ell(\boldsymbol{\rho},z,\omega)| \exp\left(\mathrm{i}\varphi_\ell(\boldsymbol{\rho},z,\omega)\right) \exp\left(\mathrm{i}\psi(\boldsymbol{\rho},z,\omega)\right)$$



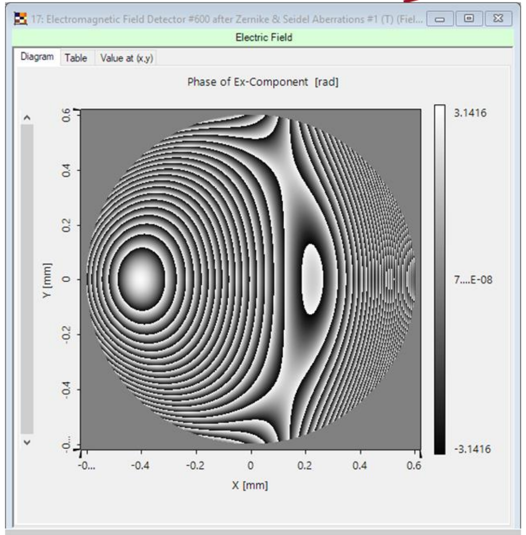
# Fourier-Transforming an Arbitrary Field

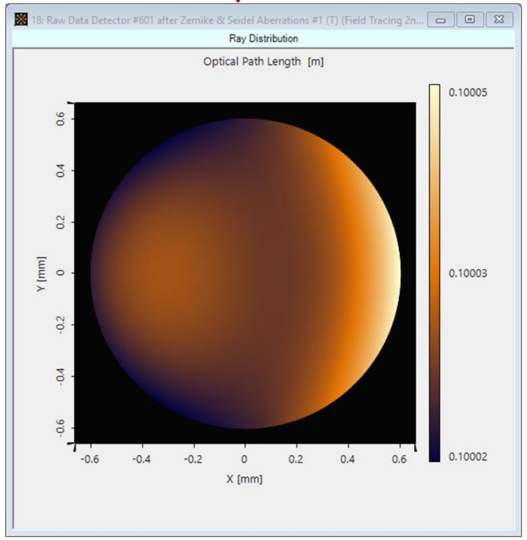
$$V_\ell(\boldsymbol{\rho},z,\omega) = |V_\ell(\boldsymbol{\rho},z,\omega)| \exp\left(\mathrm{i}\varphi_\ell(\boldsymbol{\rho},z,\omega)\right) \exp\left(\mathrm{i}\psi(\boldsymbol{\rho},z,\omega)\right)$$



# Fourier-Transforming an Arbitrary Field

$$V_\ell(\boldsymbol{\rho},z,\omega) = |V_\ell(\boldsymbol{\rho},z,\omega)| \exp\left(\mathrm{i}\varphi_\ell(\boldsymbol{\rho},z,\omega)\right) \exp\left(\mathrm{i}\psi(\boldsymbol{\rho},z,\omega)\right)$$

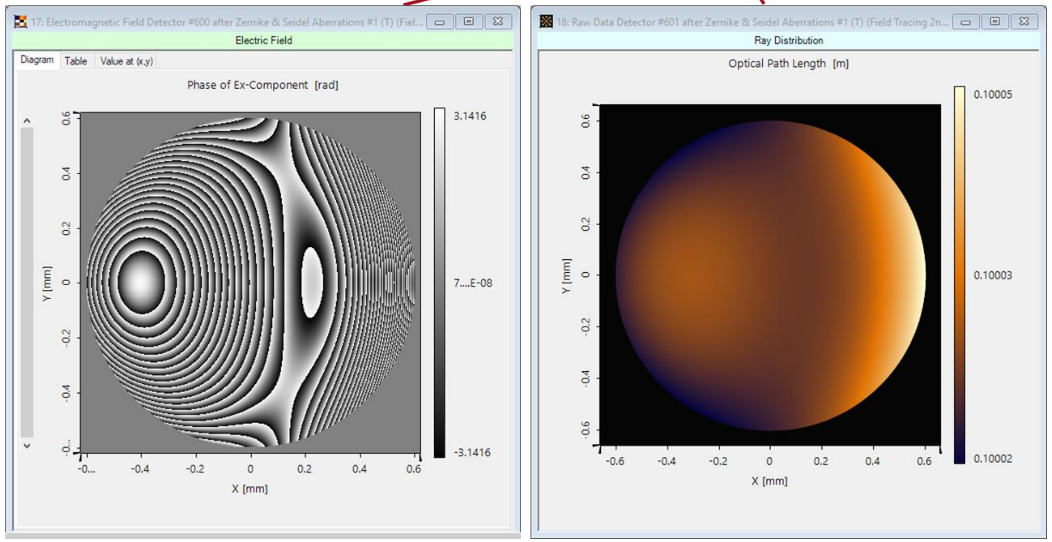


# Fourier-Transforming an Arbitrary Field

- Direct discretization of Fourier integral $\rightarrow$ DFT (Discrete Fourier Transform): numerical complexity $\sim N^2$

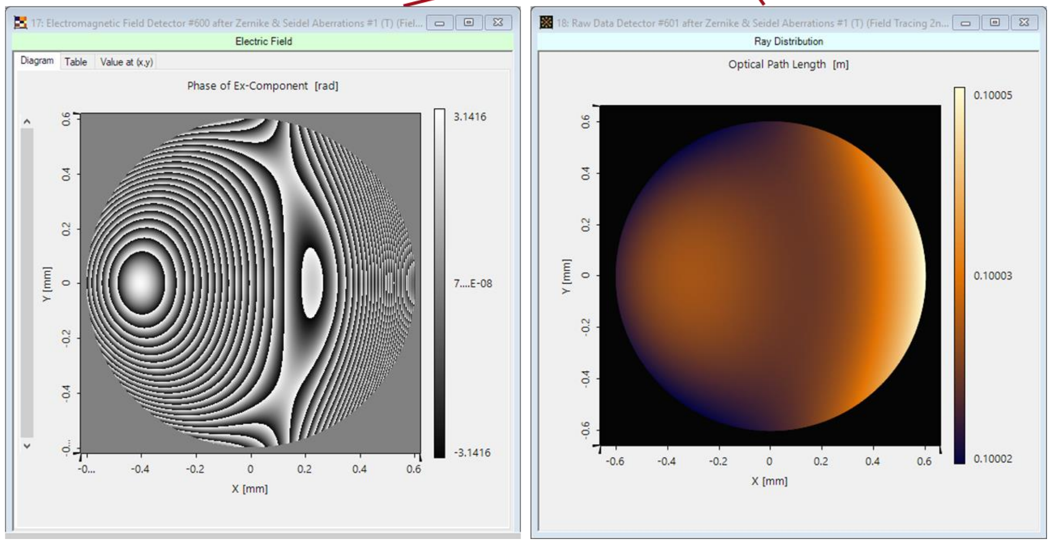
$$V_\ell(\boldsymbol{\rho},z,\omega) = |V_\ell(\boldsymbol{\rho},z,\omega)| \exp\big(\mathrm{i}\varphi_\ell(\boldsymbol{\rho},z,\omega)\big) \exp\big(\mathrm{i}\psi(\boldsymbol{\rho},z,\omega)\big)$$



# Fourier-Transforming an Arbitrary Field

- Direct discretization of Fourier integral $\rightarrow$ DFT (Discrete Fourier Transform): numerical complexity $\sim N^2$
- Numerical trick in Fast Fourier Transform (FFT) brings down computational effort to $\sim N \log N \rightarrow \sim N$

$$V_\ell(\boldsymbol{\rho}, z, \omega) = |V_\ell(\boldsymbol{\rho}, z, \omega)| \exp\left(\mathrm{i}\varphi_\ell(\boldsymbol{\rho}, z, \omega)\right) \exp\left(\mathrm{i}\psi(\boldsymbol{\rho}, z, \omega)\right)$$

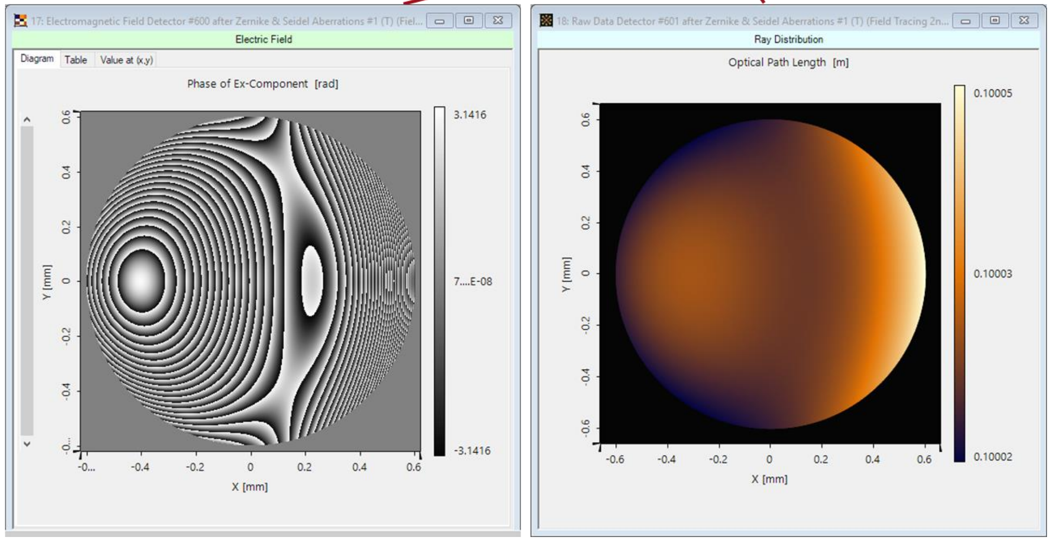


# Fourier-Transforming an Arbitrary Field

- Direct discretization of Fourier integral $\rightarrow$ DFT (Discrete Fourier Transform): numerical complexity $\sim N^2$
- Numerical trick in Fast Fourier Transform (FFT) brings down computational effort to $\sim N \log N \rightarrow \sim N$

This already means **pointwise** operation, right? Job done! ✓

$$V_\ell(\boldsymbol{\rho}, z, \omega) = |V_\ell(\boldsymbol{\rho}, z, \omega)| \exp\left(i\varphi_\ell(\boldsymbol{\rho}, z, \omega)\right) \exp\left(i\psi(\boldsymbol{\rho}, z, \omega)\right)$$

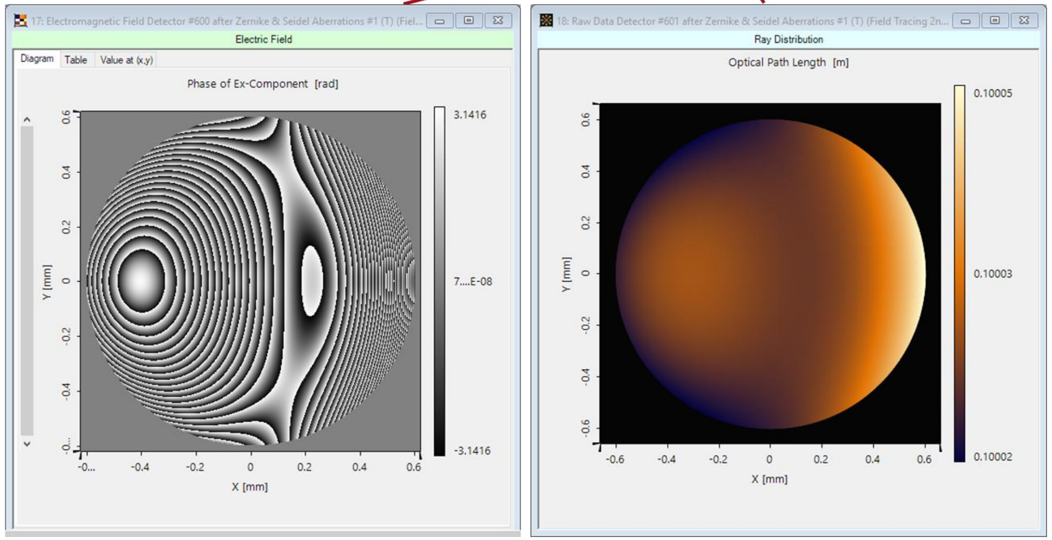


# Fourier-Transforming an Arbitrary Field

- Direct discretization of Fourier integral $\rightarrow$ DFT (Discrete Fourier Transform): numerical complexity $\sim N^2$
- Numerical trick in Fast Fourier Transform (FFT) brings down computational effort to $\sim N \log N \rightarrow \sim N$

This already means **pointwise** operation, right? Job done! ×

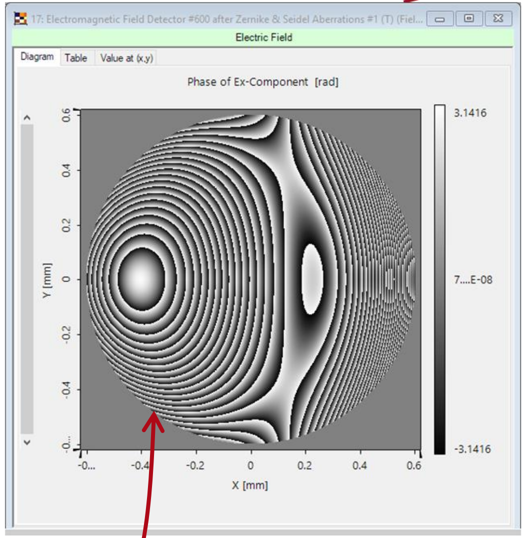
$$V_\ell(\boldsymbol{\rho},z,\omega) = |V_\ell(\boldsymbol{\rho},z,\omega)| \exp\left(\mathrm{i}\varphi_\ell(\boldsymbol{\rho},z,\omega)\right) \exp\left(\mathrm{i}\psi(\boldsymbol{\rho},z,\omega)\right)$$

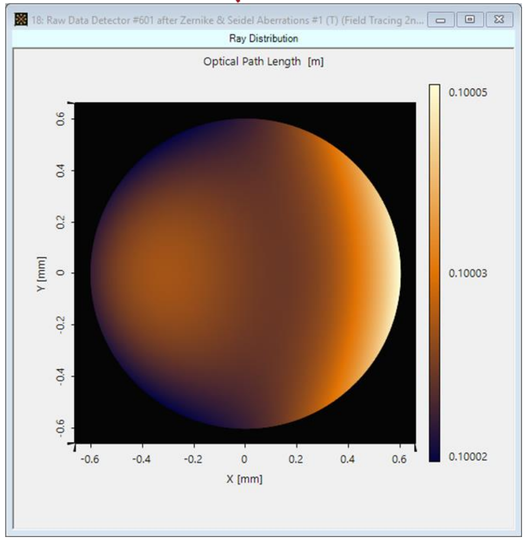


# Fourier-Transforming an Arbitrary Field

- Direct discretization of Fourier integral $\rightarrow$ DFT (Discrete Fourier Transform): numerical complexity $\sim N^2$
- Numerical trick in Fast Fourier Transform (FFT) brings down computational effort to $\sim N \log N \rightarrow \sim N$

$$V_\ell(\boldsymbol{\rho}, z, \omega) = |V_\ell(\boldsymbol{\rho}, z, \omega)| \exp\left(\mathrm{i}\varphi_\ell(\boldsymbol{\rho}, z, \omega)\right) \exp\left(\mathrm{i}\psi(\boldsymbol{\rho}, z, \omega)\right)$$



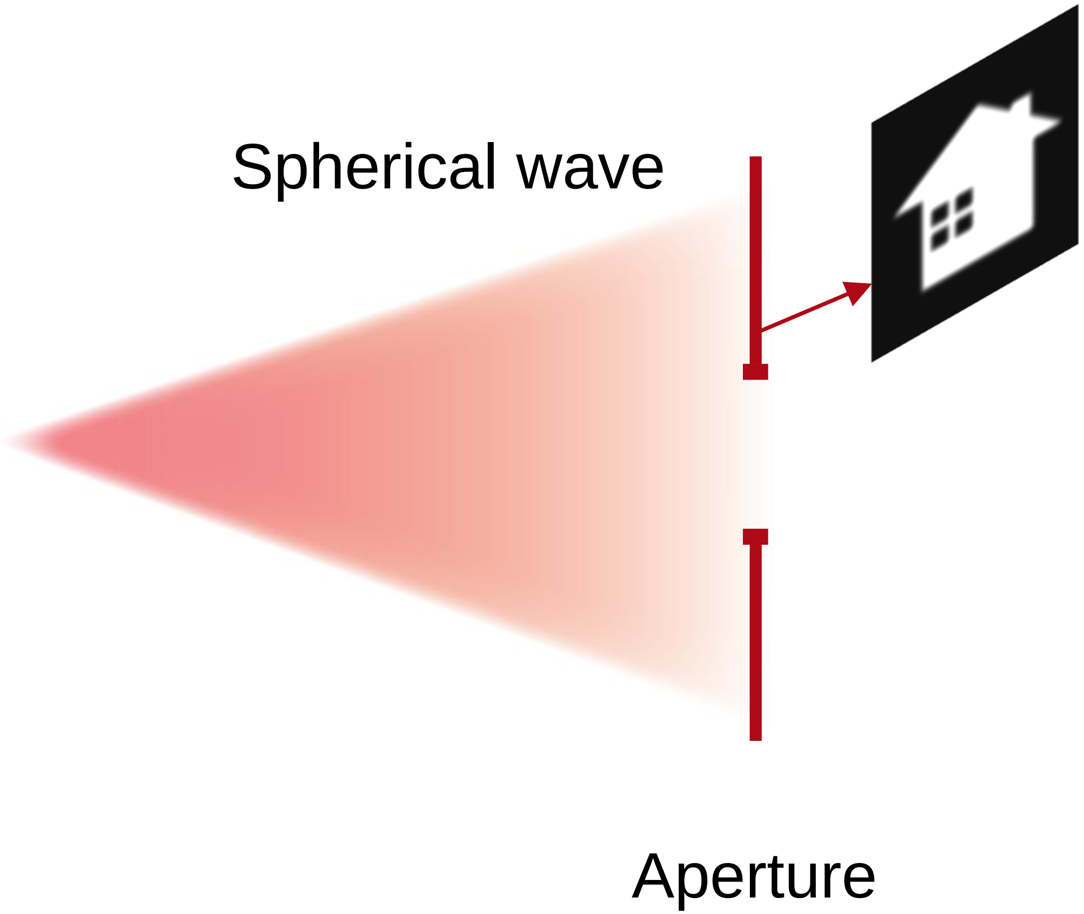




The FFT requires fulfilment of the Nyquist-Shannon sampling theorem $\Rightarrow$ wrapped phase **must** be well resolved!

$\Downarrow$

**Huge sample number $N$**

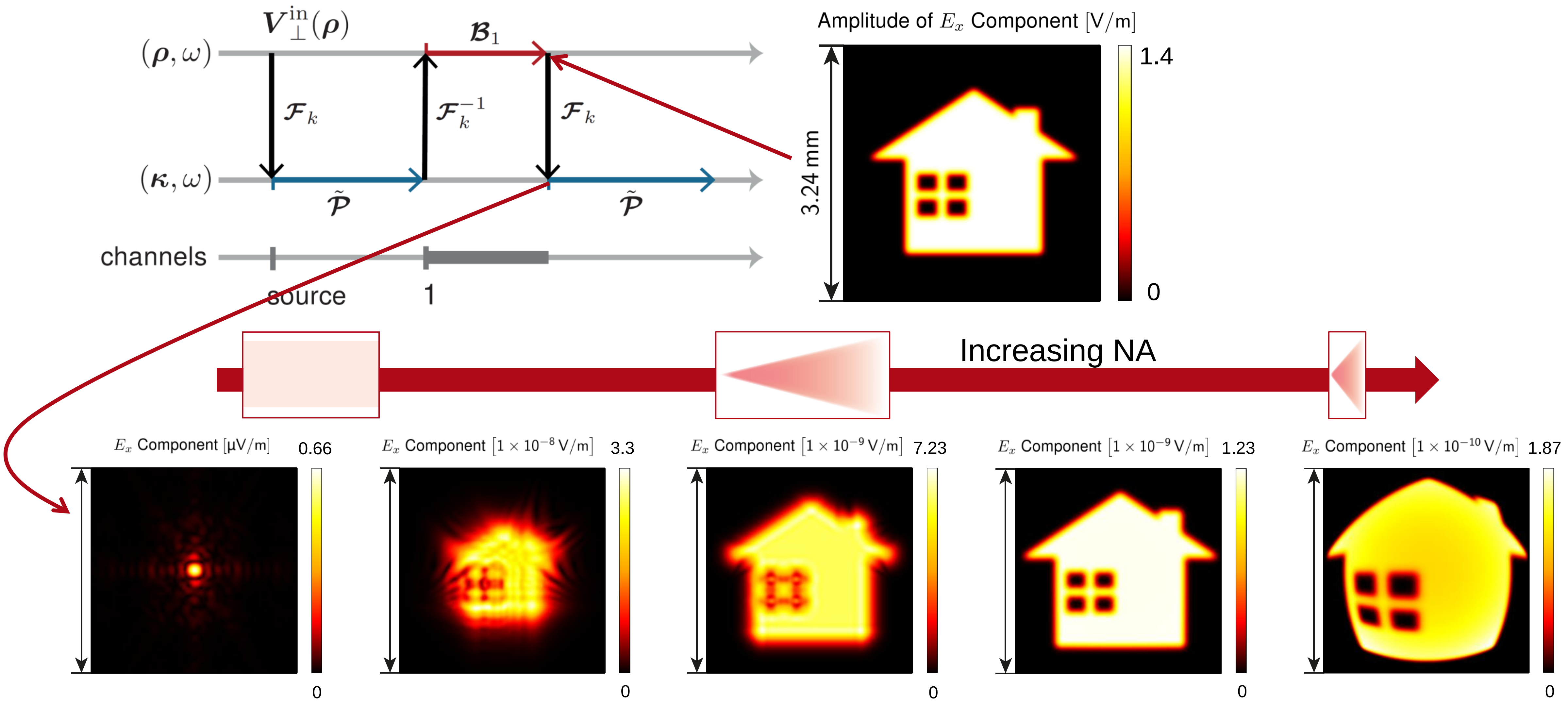
# Modeling the Propagation Through an Aperture



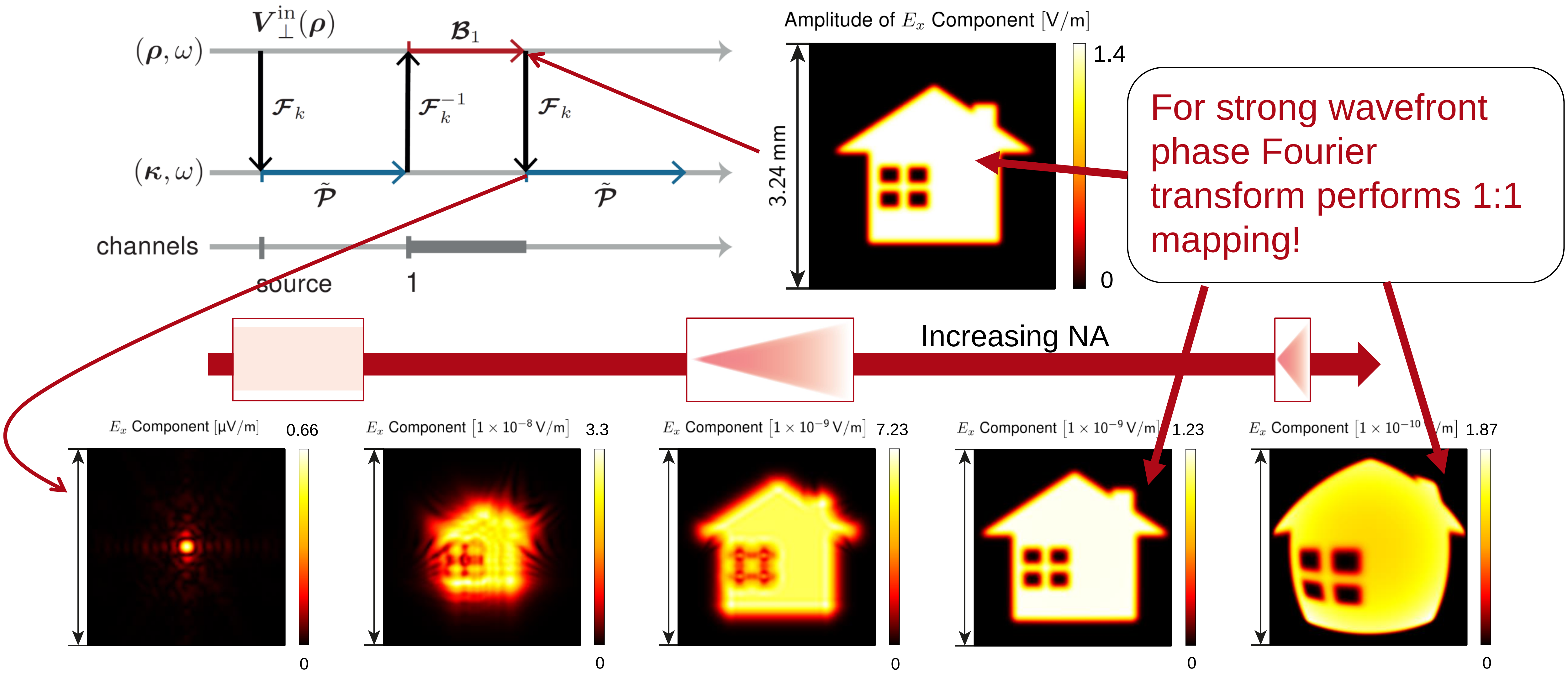

**Field after the aperture**

Amplitude of $E_x$ Component [V/m]

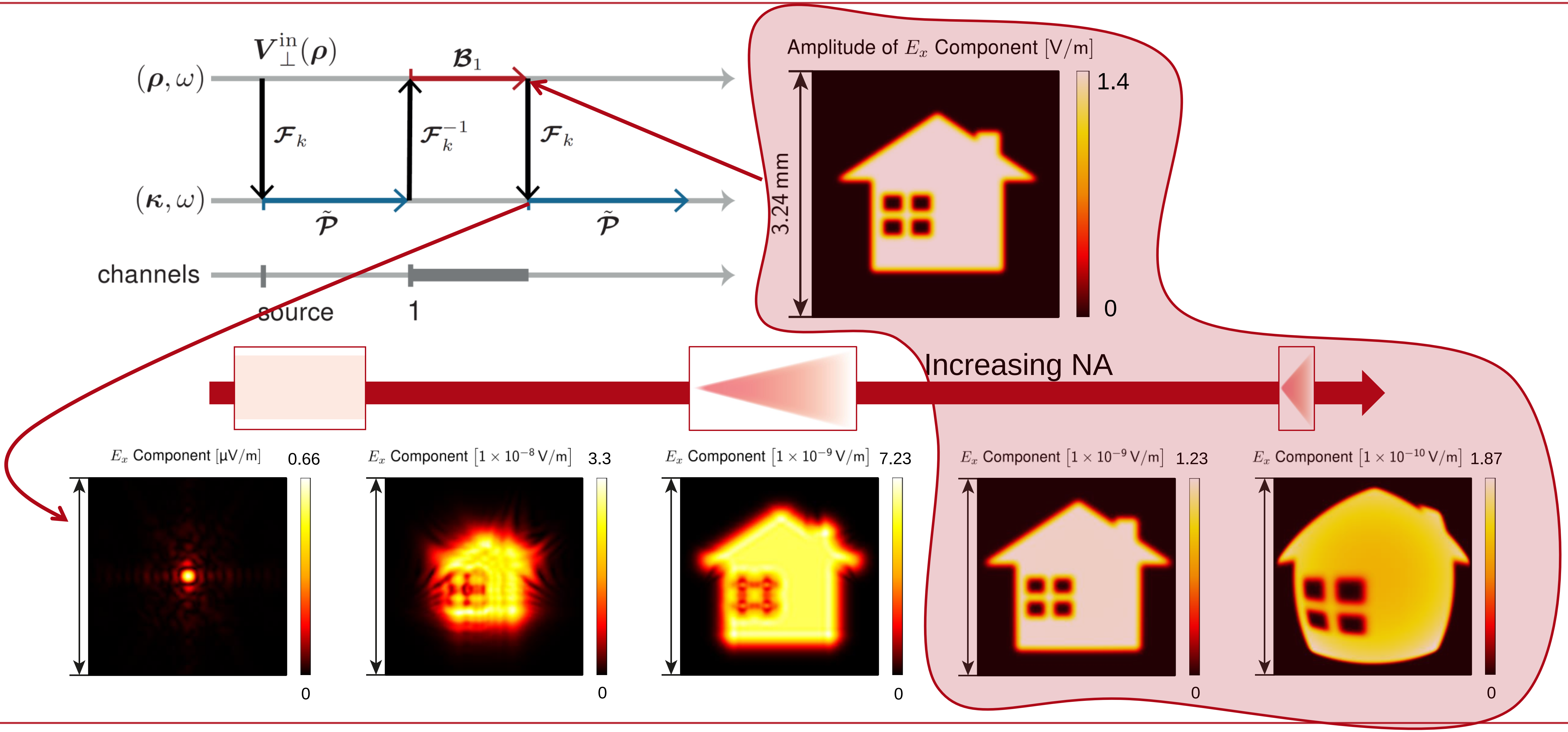
# Results of Fourier Transform



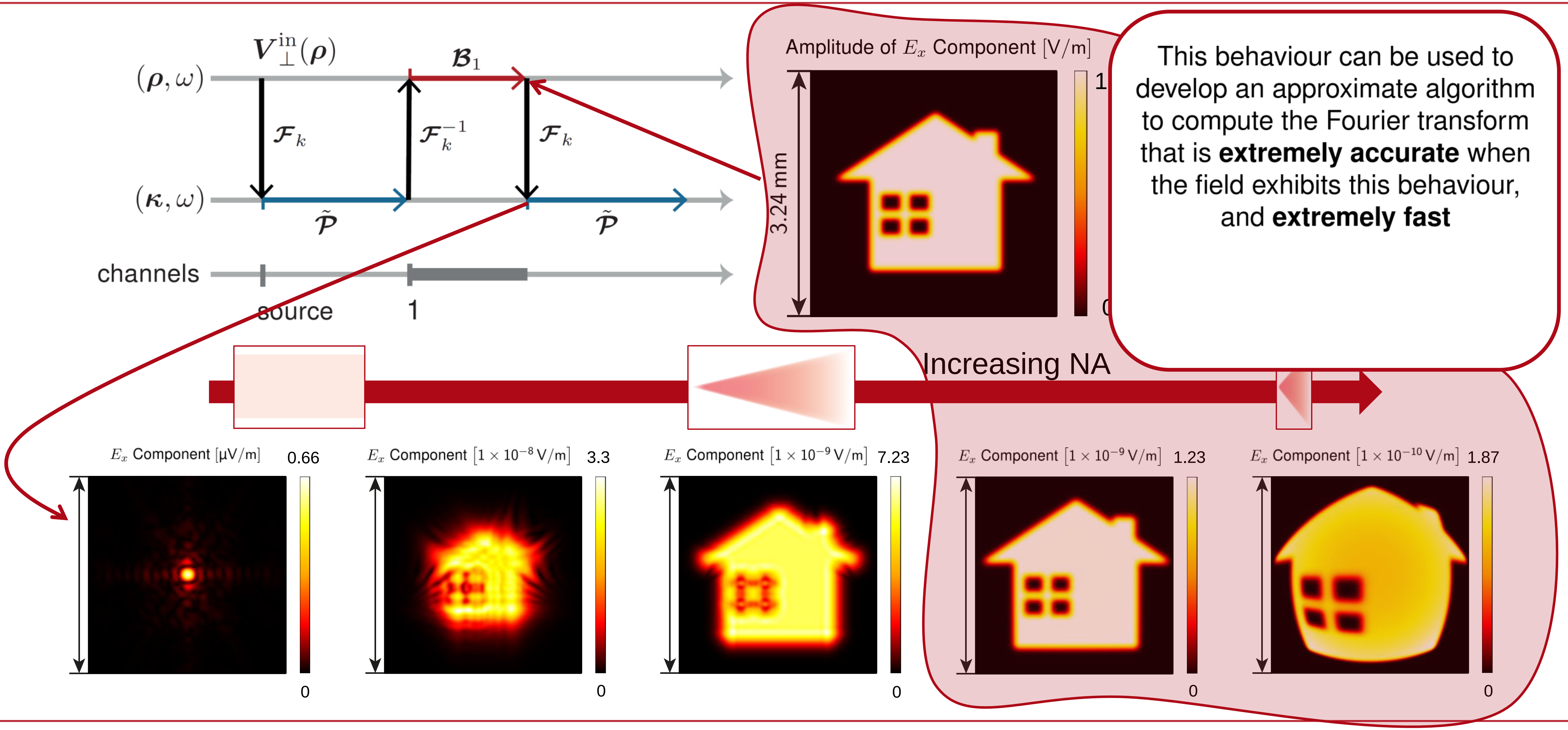
# Results of Fourier Transform



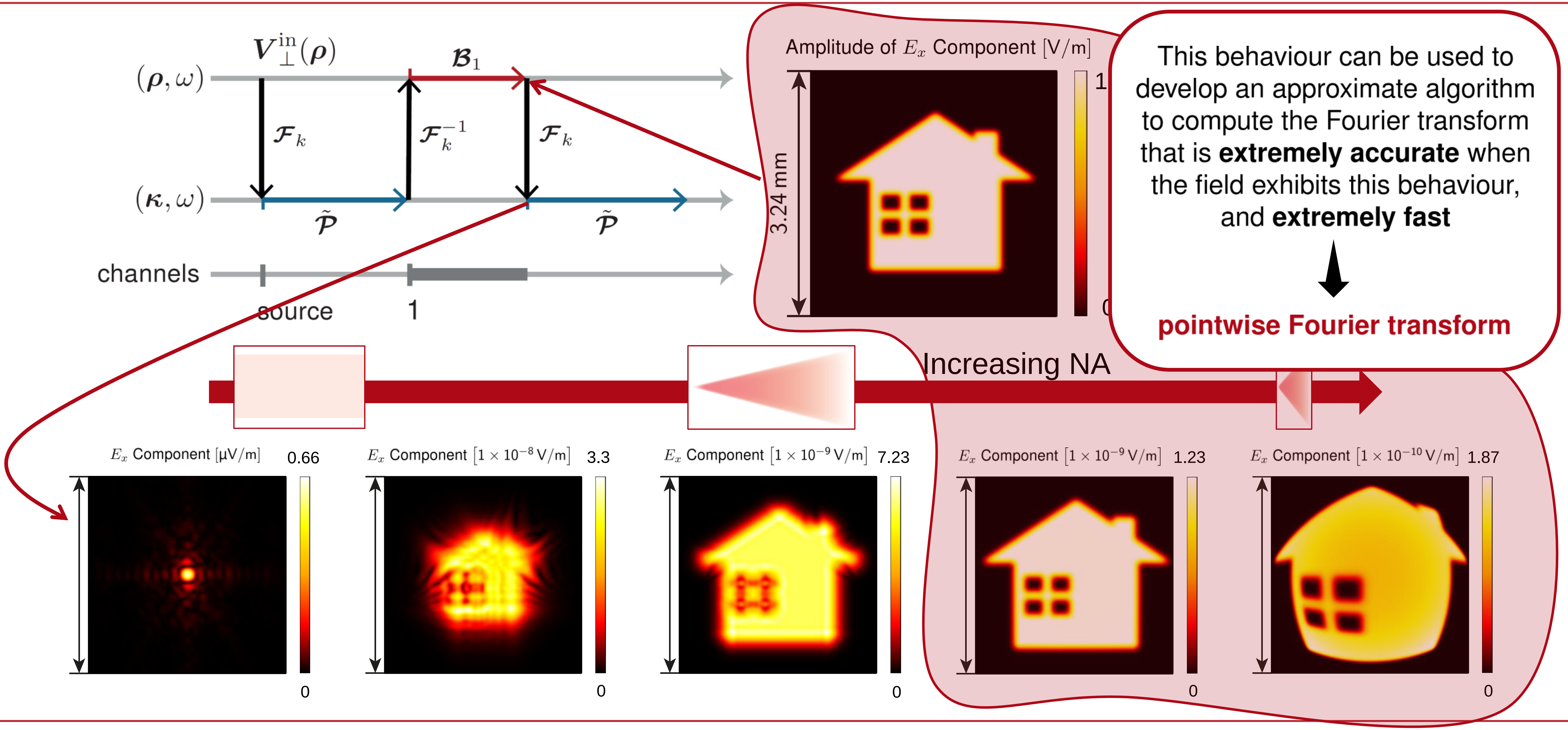
# Results of Fourier Transform



# Results of Fourier Transform



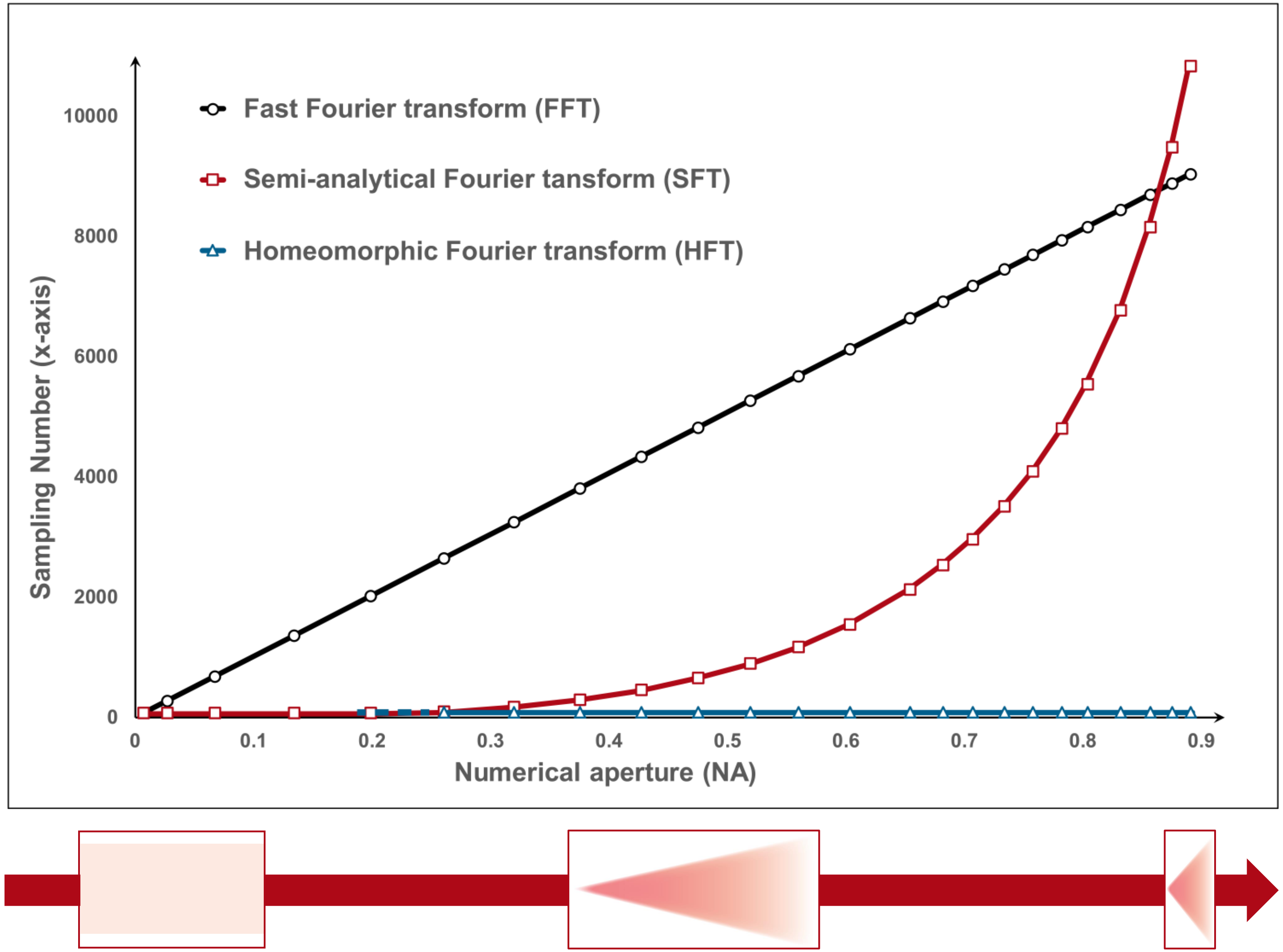
# Results of Fourier Transform




This behaviour can be used to develop an approximate algorithm to compute the Fourier transform that is **extremely accurate** when the field exhibits this behaviour, and **extremely fast**

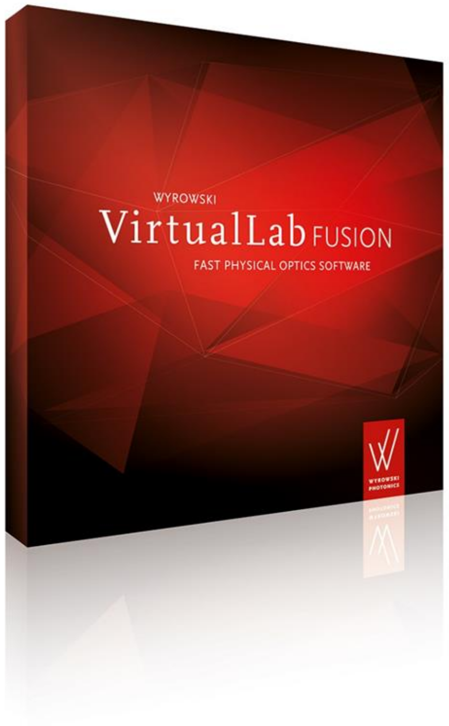
**pointwise Fourier transform**

# Comparison of the Sampling Effort



# Types of Fourier Transform Algorithms in VirtualLab Fusion

- Fast Fourier Transform (FFT)
  - Fast for weak wavefront phase
- Semianalytical Fourier transform (SFT)
  - Fast for wavefront phase with medium local gradient
- Pointwise Fourier transform (HFT)
  - Accurate for strong wavefront phase




→ Combination of Fourier transform algorithms essential for fast physical optics!

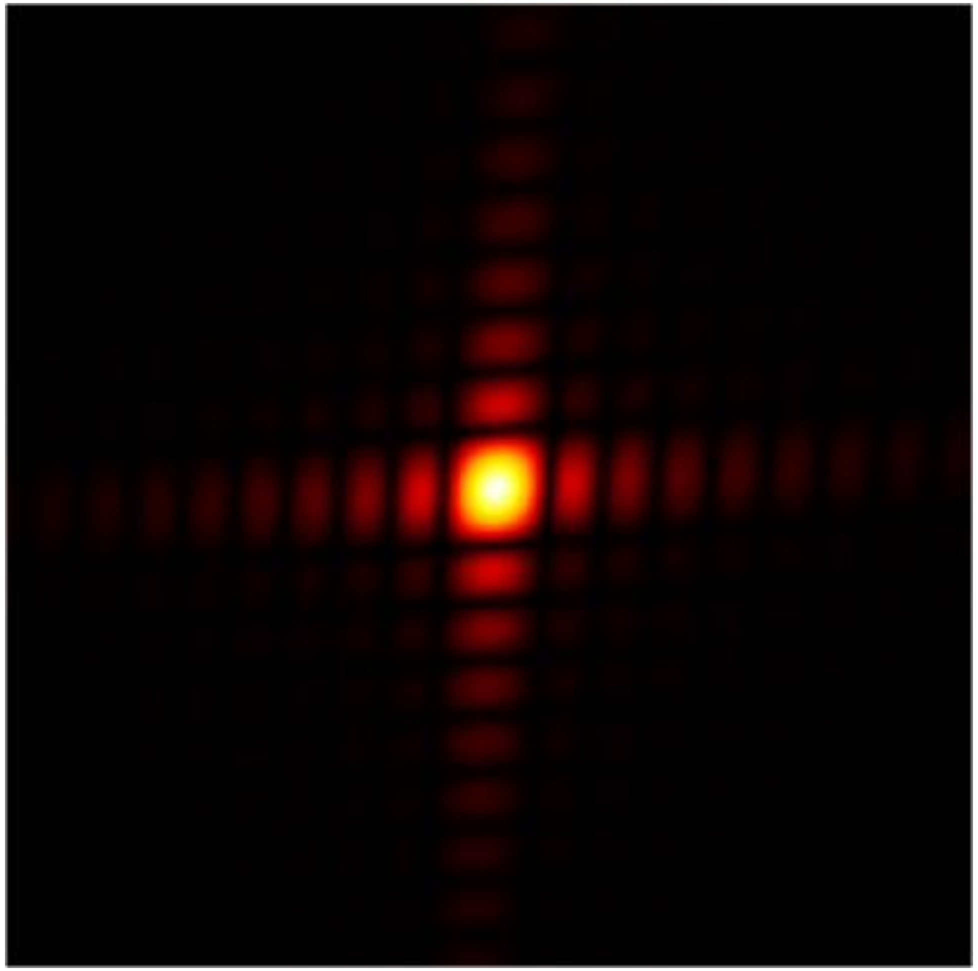
# References to Our Fourier Transform Publications

- *Theory and algorithm of the homeomorphic Fourier transform for optical simulations*, Z. Wang et al, Optics Express, **28**, 7, 2020

- *Application of the semi-analytical Fourier transform to electromagnetic modeling*, Z. Wang et al, Optics Express, **27**, 11, 2019

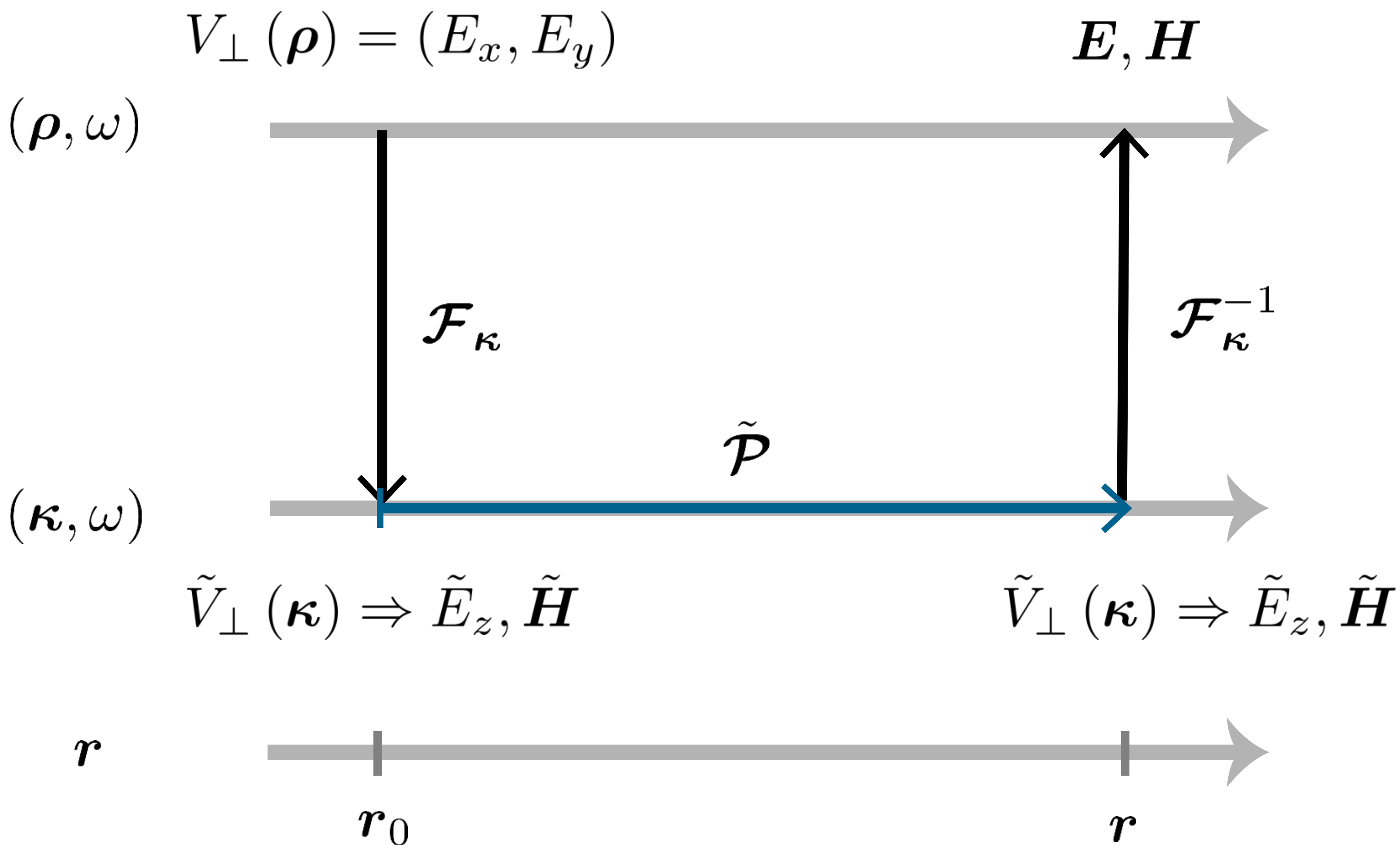
# Automatic Selection of Fourier Transform Techniques in Free-Space Propagation Operator

# Abstract



Accurate and efficient simulation of free-space propagation of electromagnetic fields is essential for physical optics modeling and design. VirtualLab Fusion has a unified free-space propagation concept. It is based on the spatial-frequency domain (k-domain) analysis. In combination with different Fourier transform techniques, it delivers numerical efficient solutions for different situations of free-space propagations. The selection of appropriate Fourier transform is automatic according to the situation.

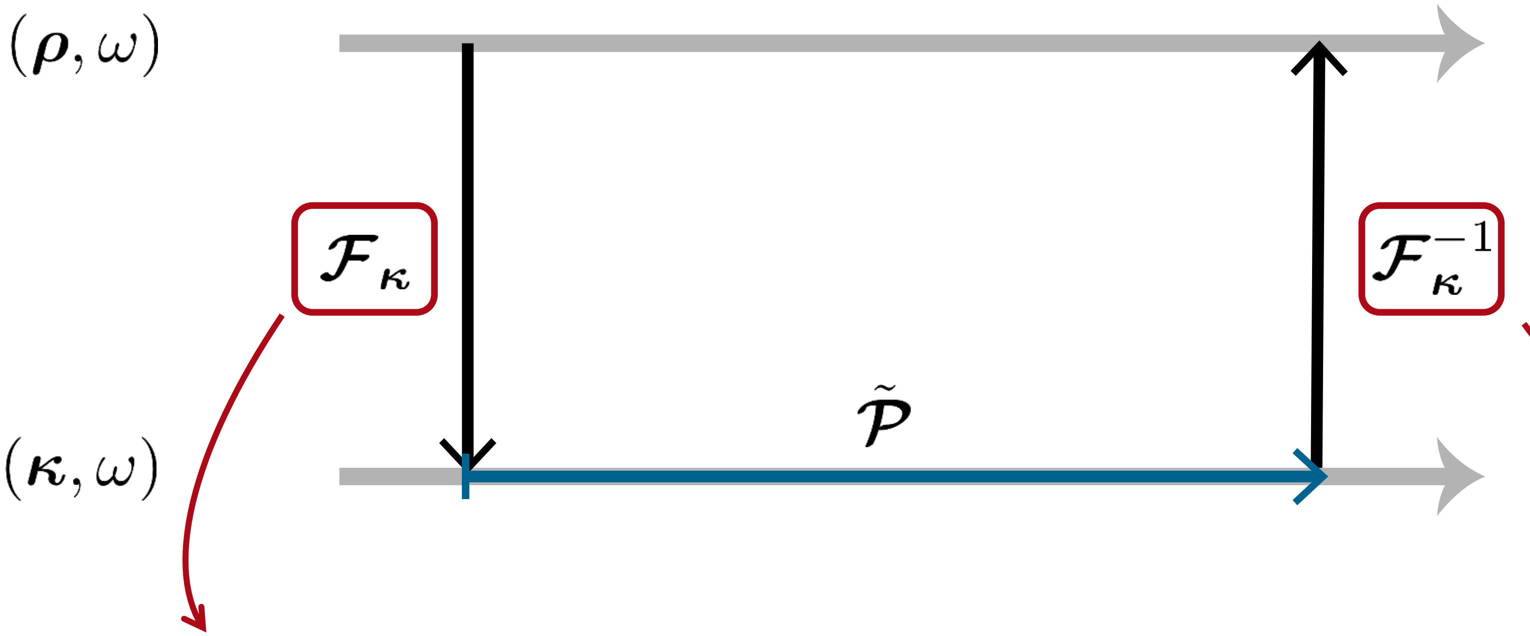
# Concept of Free-Space Propagation Operator




- Unified propagation operator in the k-domain
- Applicable for arbitrarily oriented planes
- Switching between two domains via Fourier transform
- References


- F. Wyrowski, "Unification of the geometric and diffractive theories of electromagnetic fields" Proc. DGaO, (2017)
- Z. Wang *et al.*, „Application of the semi-analytical Fourier transform to electromagnetic modeling," Opt. Express 27, 15335-15350 (2019)

# Available Fourier Transform Techniques in VirtualLab Fusion

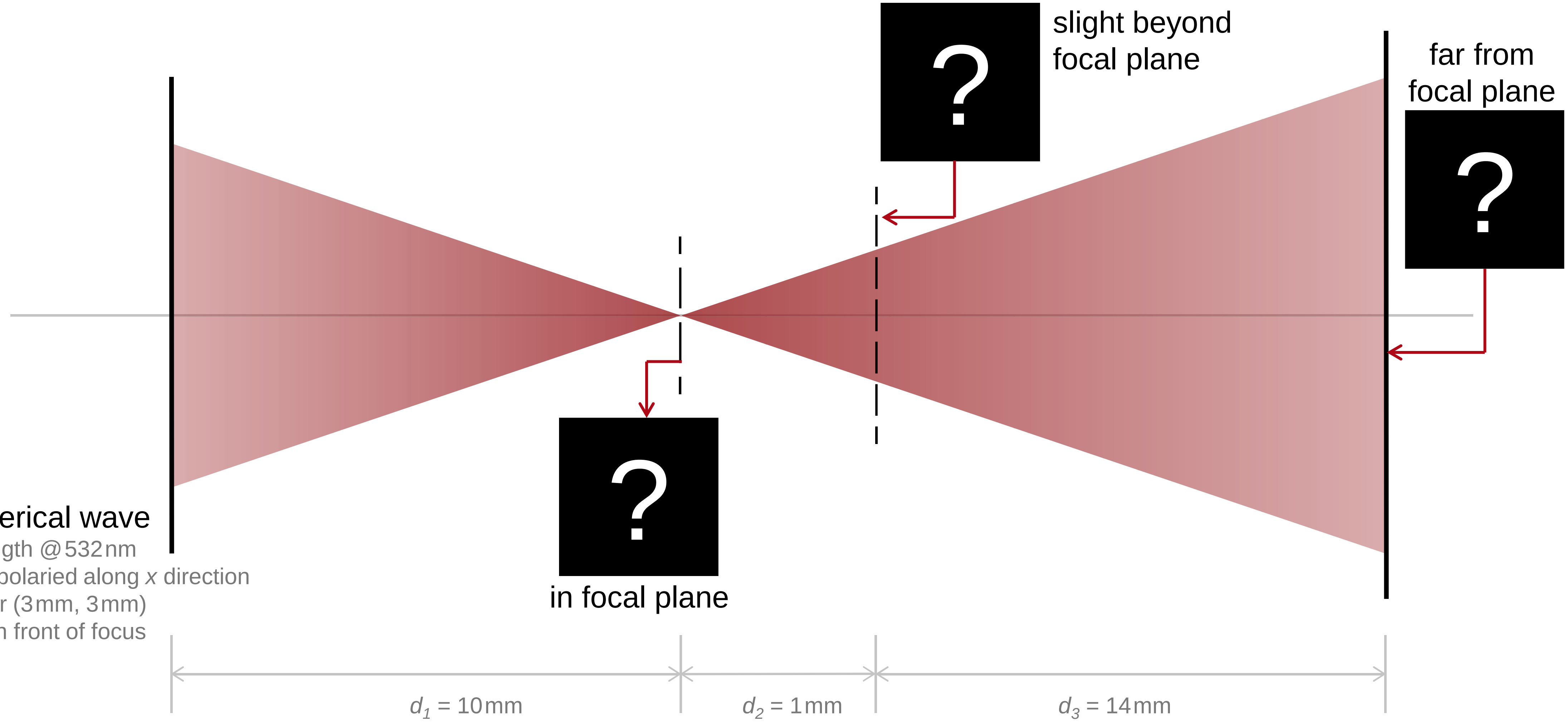



- $\mathcal{F}_{\boldsymbol{\kappa}}$: fast Fourier transform (FFT)
- $\mathcal{F}_{\boldsymbol{\kappa}}^{\text{semi}}$: semi-analytical Fourier transform (SFT)
- $\mathcal{F}_{\boldsymbol{\kappa}}^{\text{h}}$: homeomorphic Fourier transform (HFT)

- $\mathcal{F}_{\boldsymbol{\kappa}}^{-1}$: inverse fast Fourier transform (IFFT)
- $\mathcal{F}_{\boldsymbol{\kappa}}^{-1,\text{semi}}$: inverse semi-analytical Fourier transform (ISFT)
- $\mathcal{F}_{\boldsymbol{\kappa}}^{-1,\text{h}}$: inverse homeomorphic Fourier transform (IHFT)

# Example 1: Propagation of a Spherical Wave

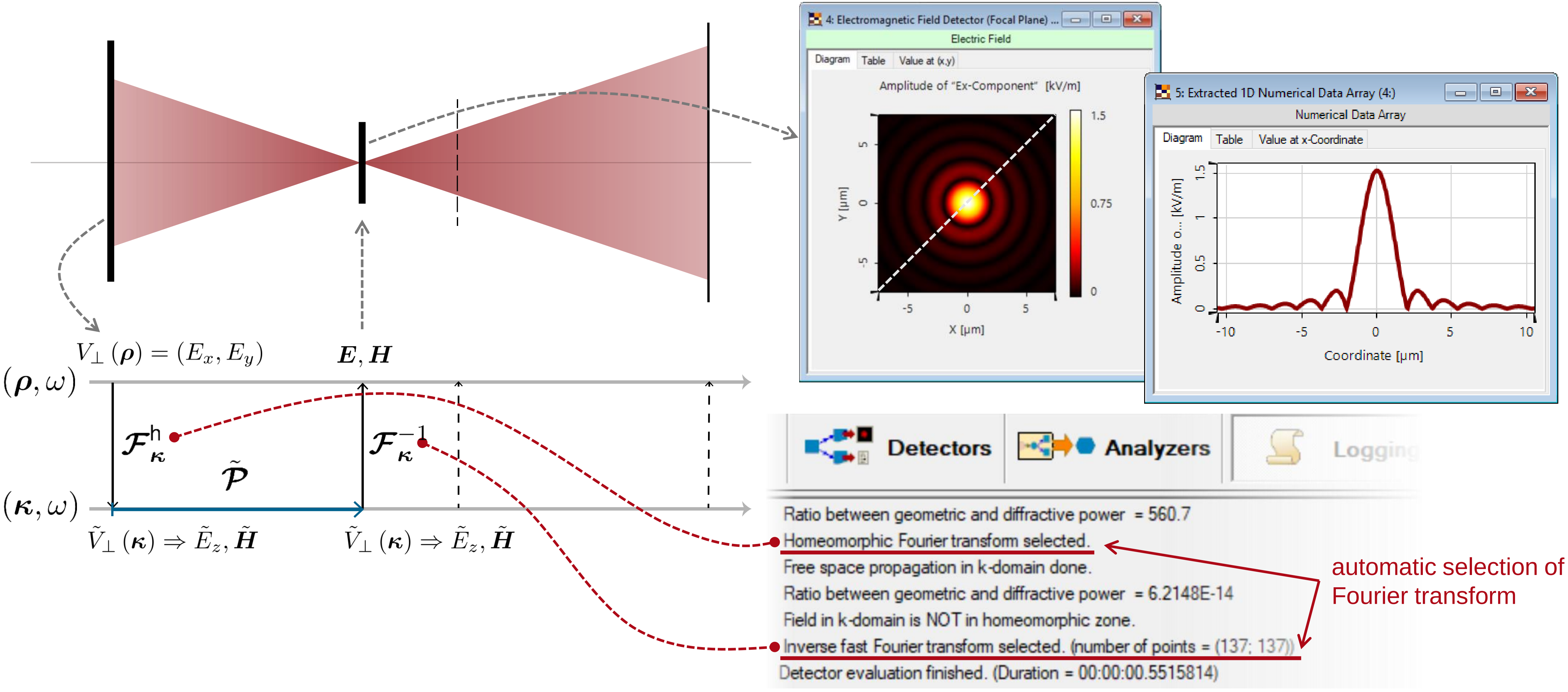
# Modeling Task



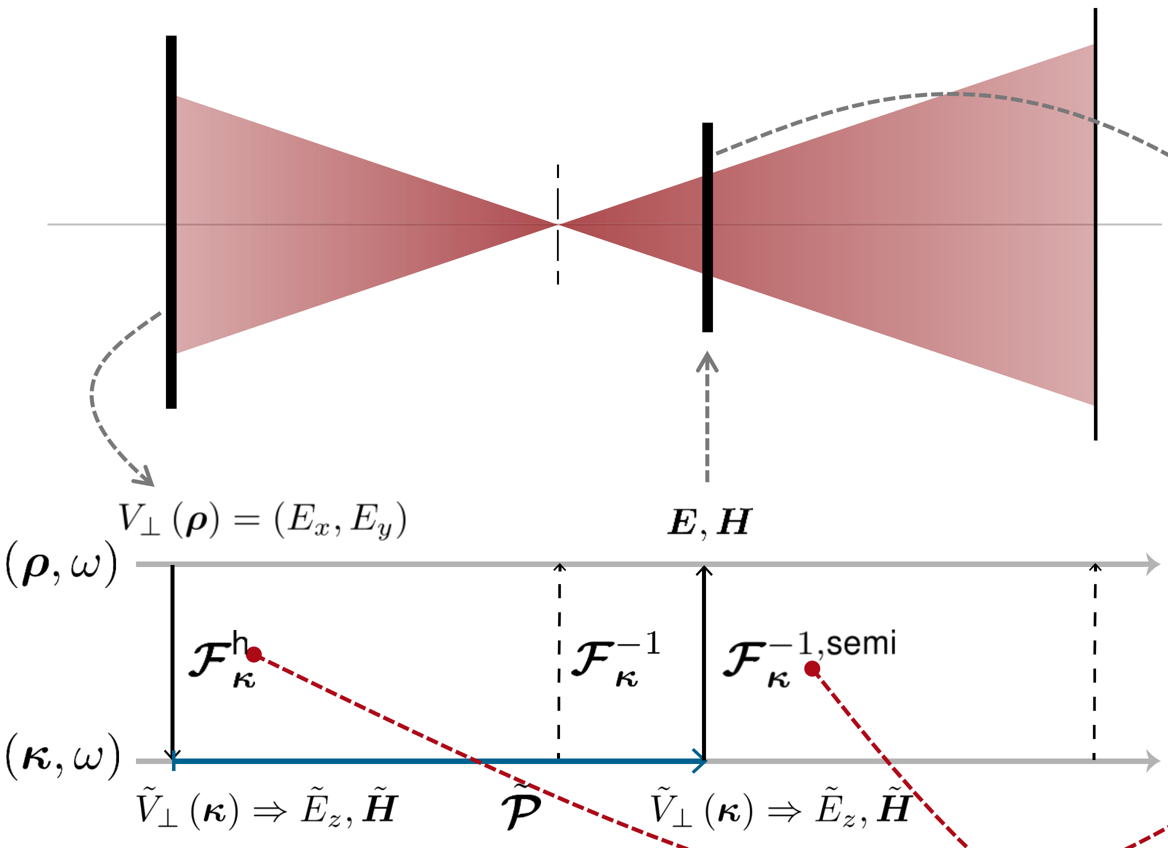

input spherical wave

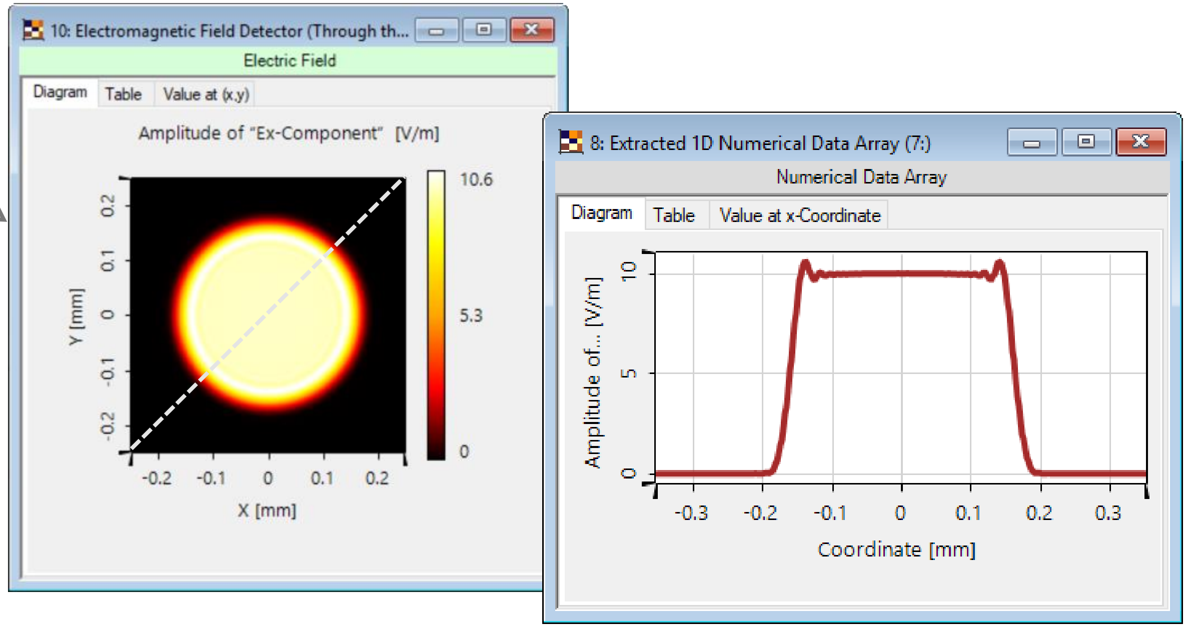
- wavelength @ 532 nm
- linearly polaried along *x* direction
- diameter (3 mm, 3 mm)
- 10 mm in front of focus

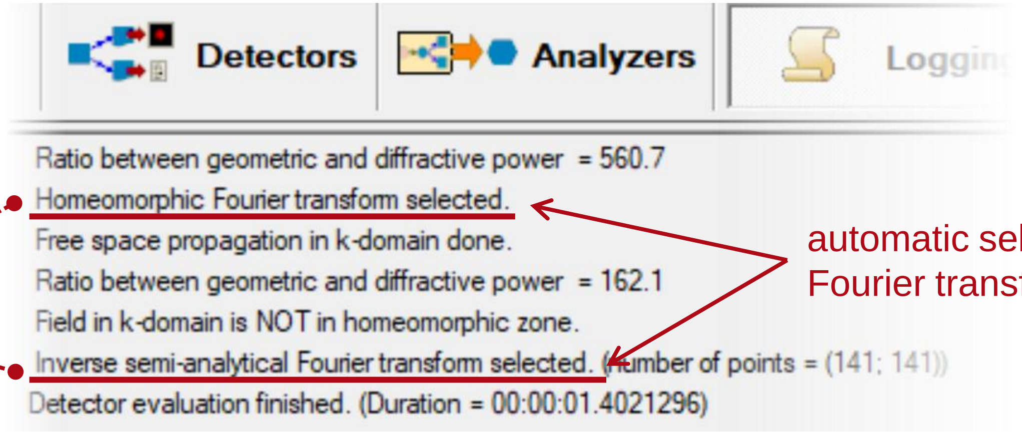
# Simulation Result: in Focal Plane



# Simulation Result: Slightly beyond Focal Plane

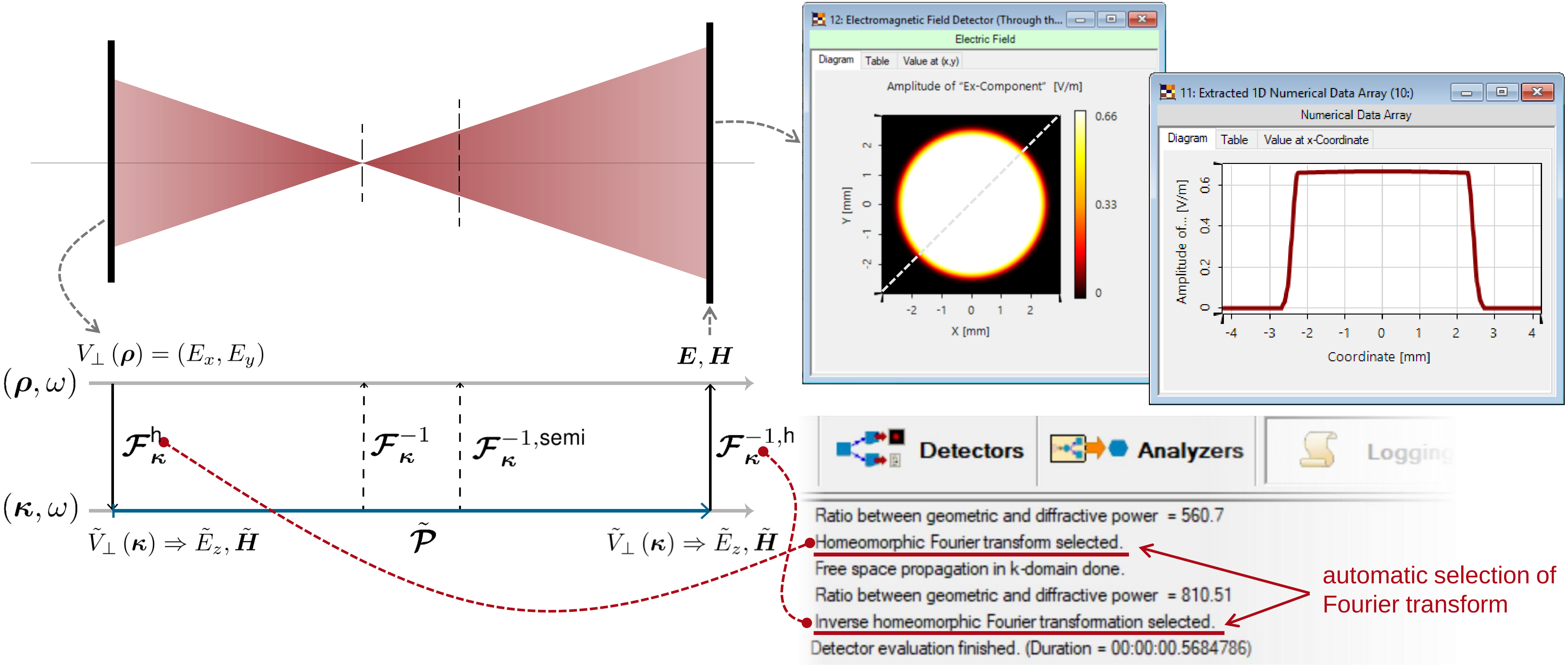



automatic selection of Fourier transform

# Simulation Result: Far from Focal Plane

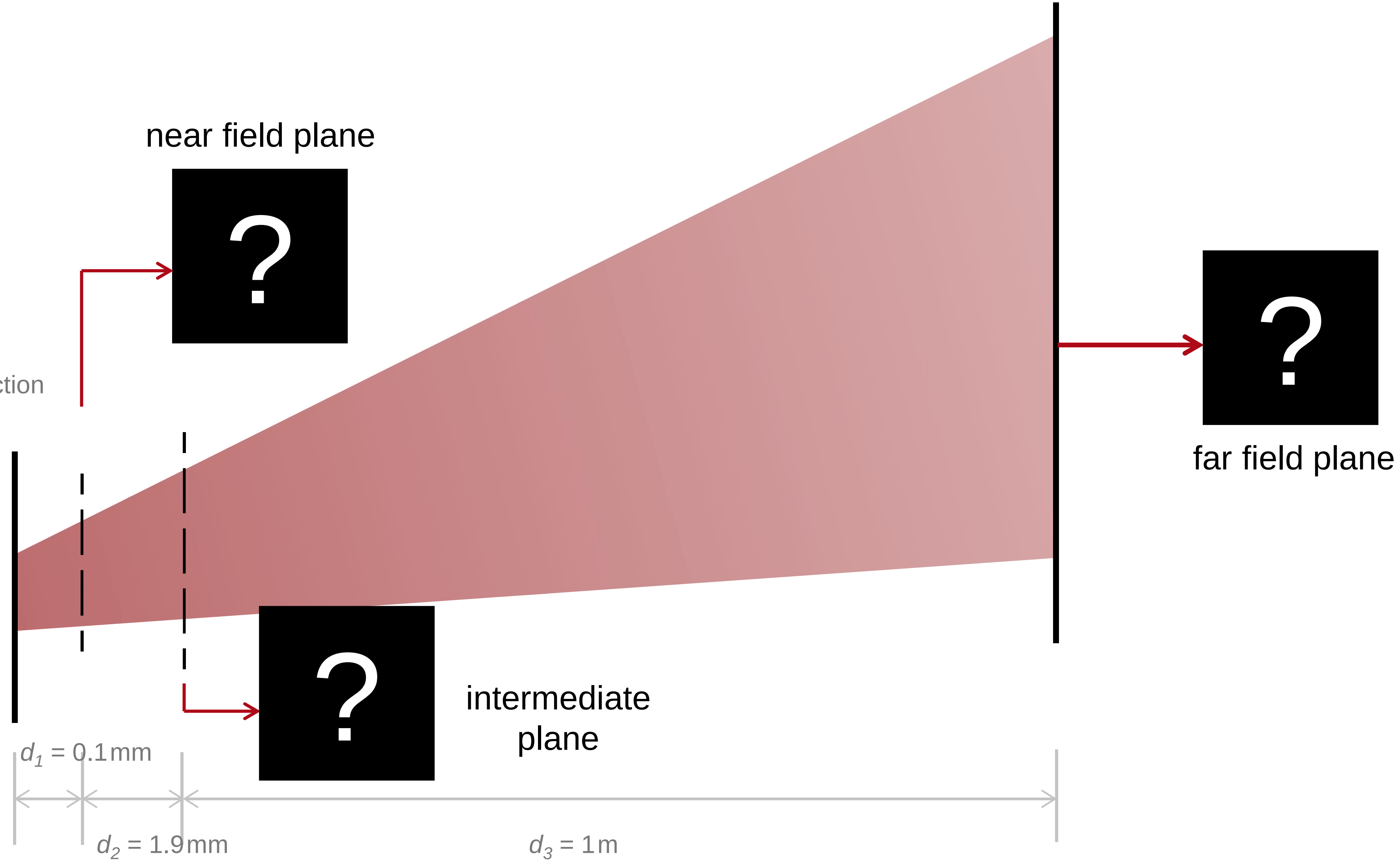


# Example 2: Propagation of a Truncated Plane Wave

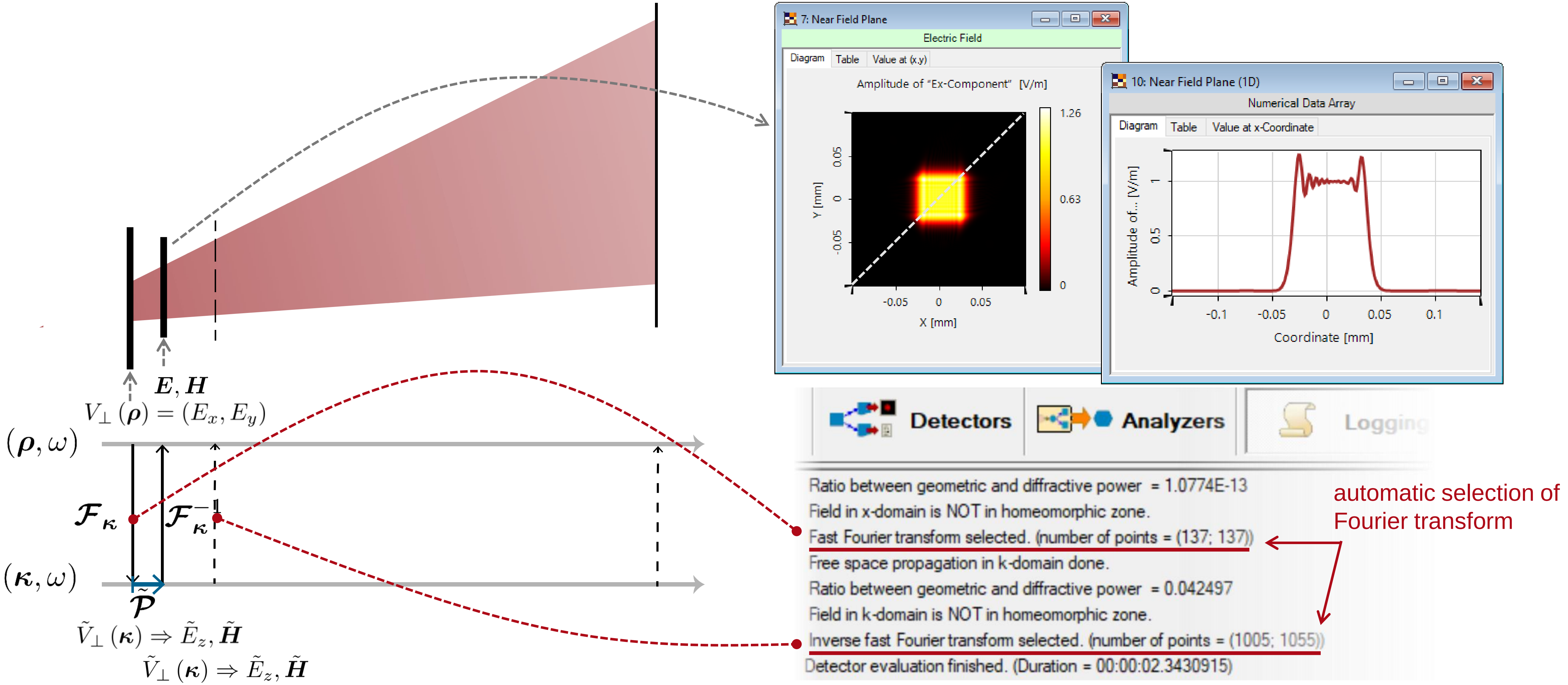
# Modeling Task

truncated plane wave

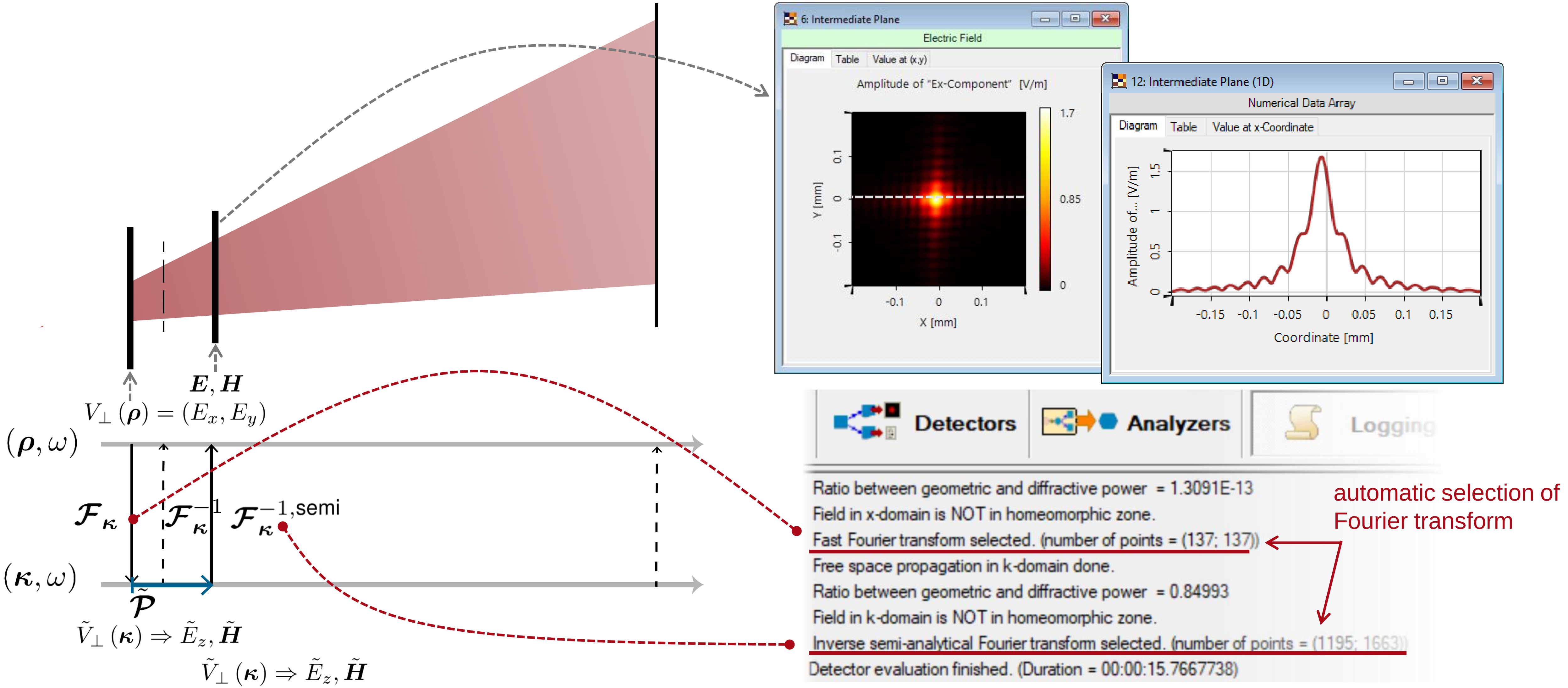
- wavelength @532nm
- linearly polaried along *x* direction
- shape: rectangular
- diameter: (50um, 50um)
- inclination(10°, 20°)



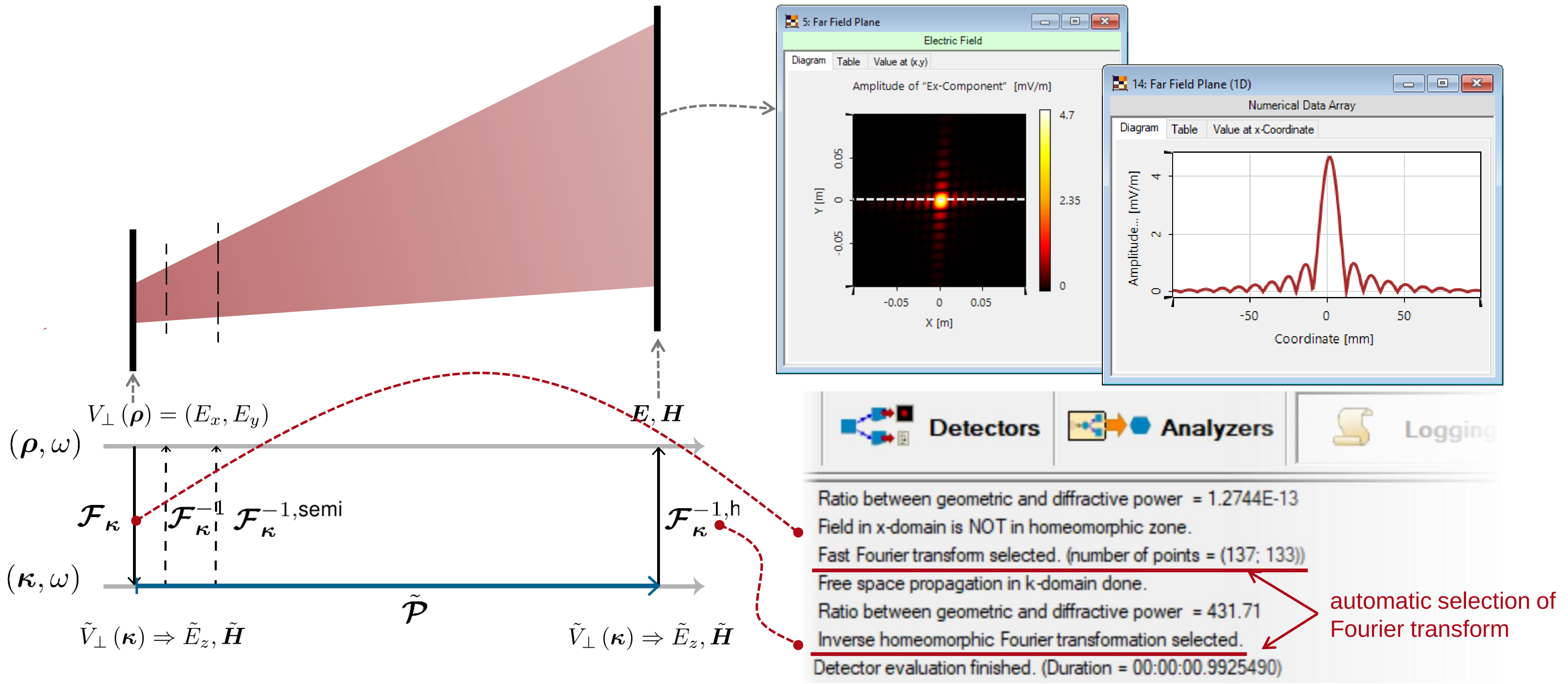
# Simulation Result: Near Field Plane



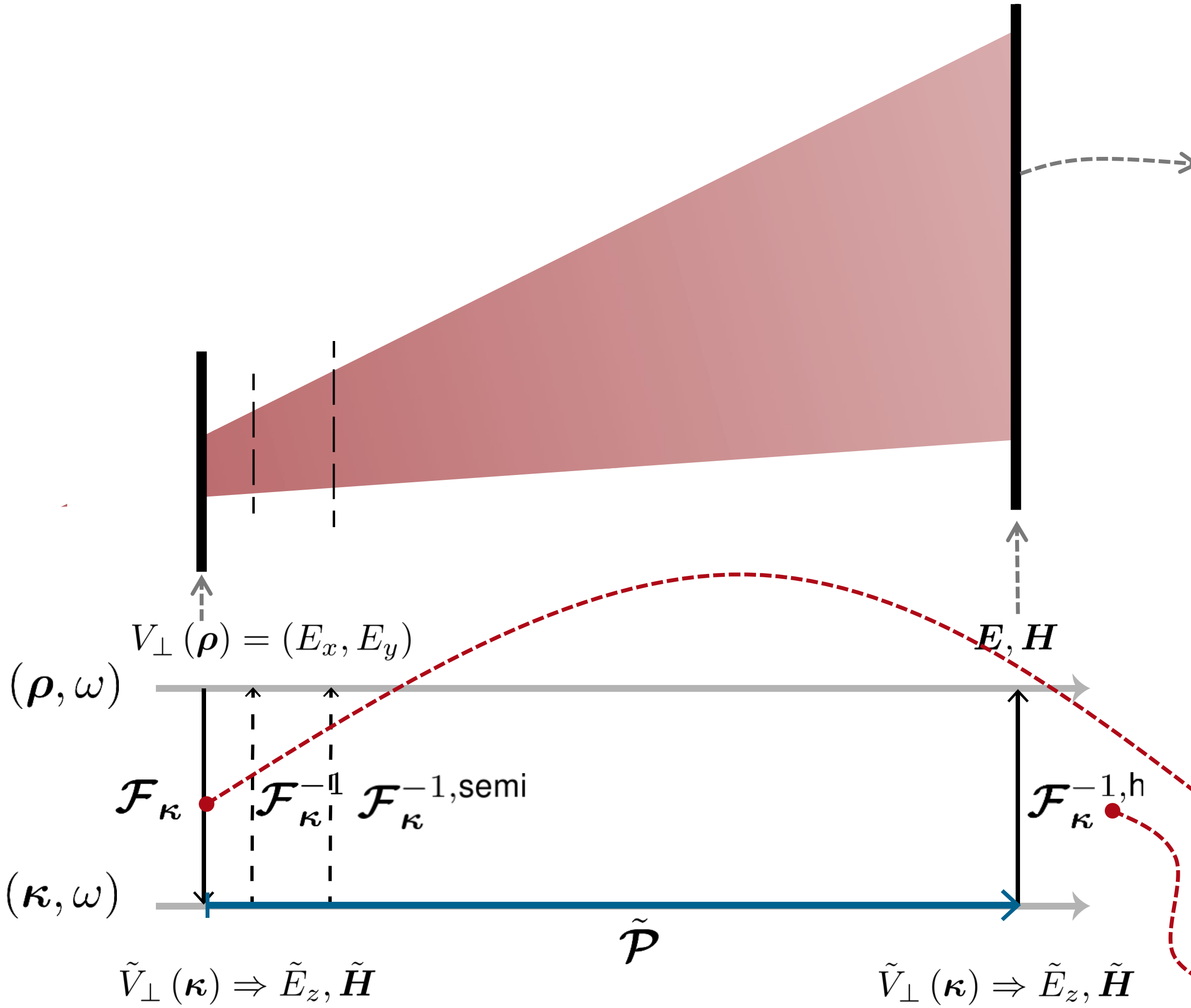
# Simulation Result: Intermediate Plane

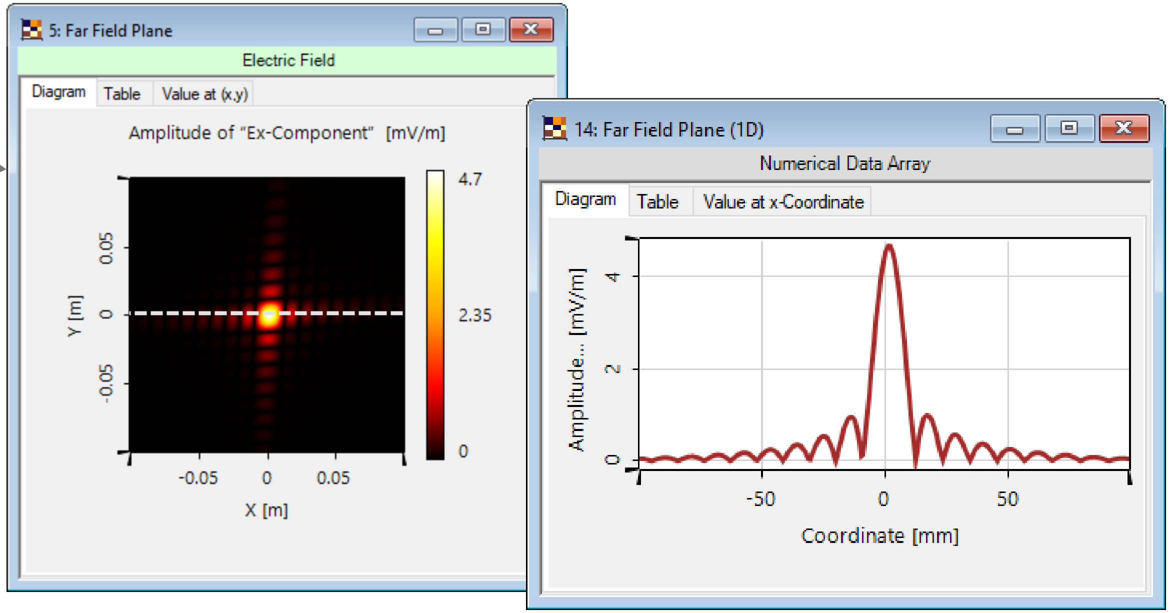


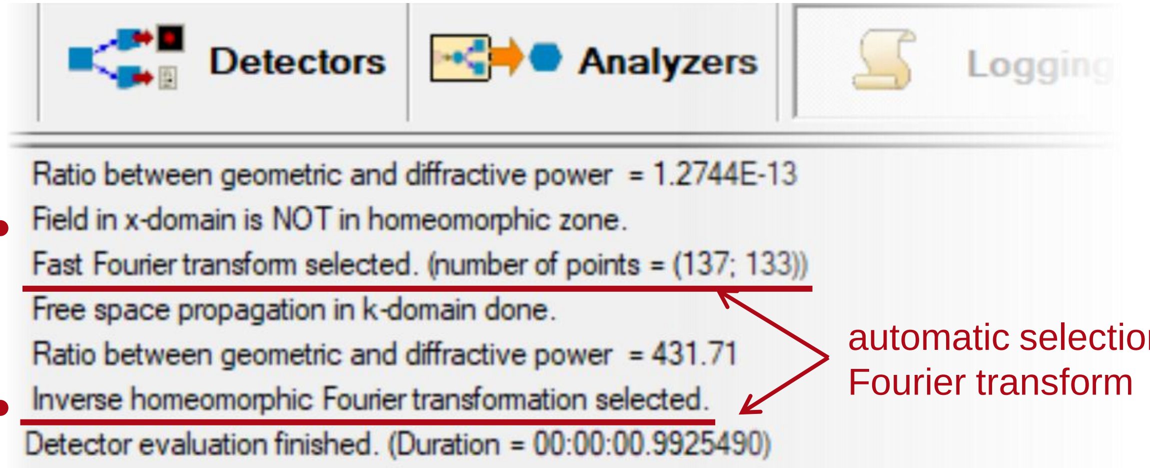
# Simulation Result: Far Field Plane



# Simulation Result: Far Field Plane



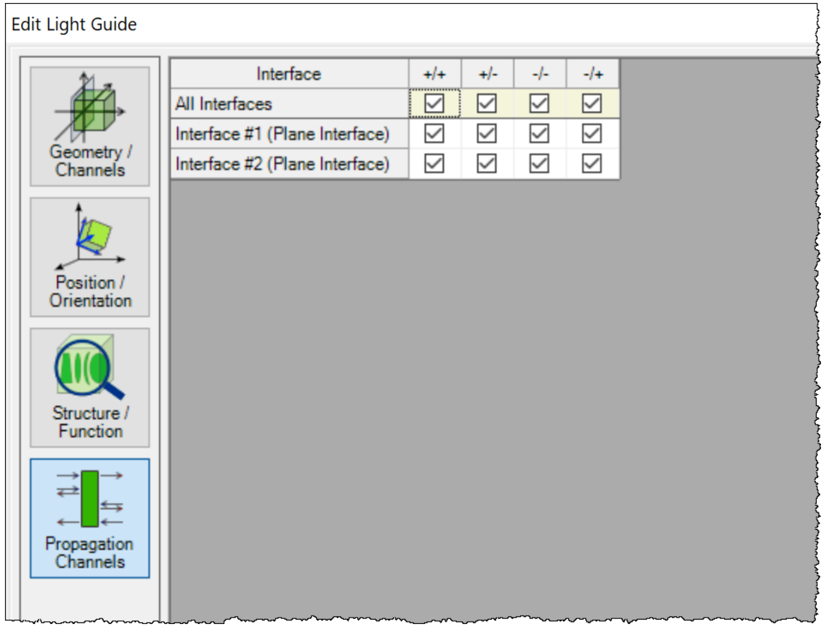

automatic selection of Fourier transform

Part 4

# The Channel Concept

# Surface Channels

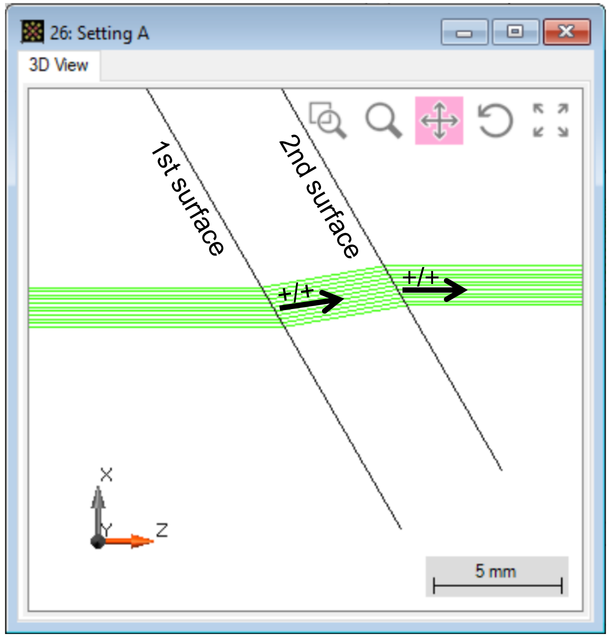
- Channel definition
  - There are four possible channels for each surface, at least one should be activated for the tracing.
  - Channels can be defined for each surface individually.
  - Different settings on channels leads to different modeling schemes.




| Channel | Description |
|---|---|
| +/+ | transmission (forward) |
| +/- | reflection (forward) |
| -/+ | reflection (backward) |
| -/- | transmission (backward) |

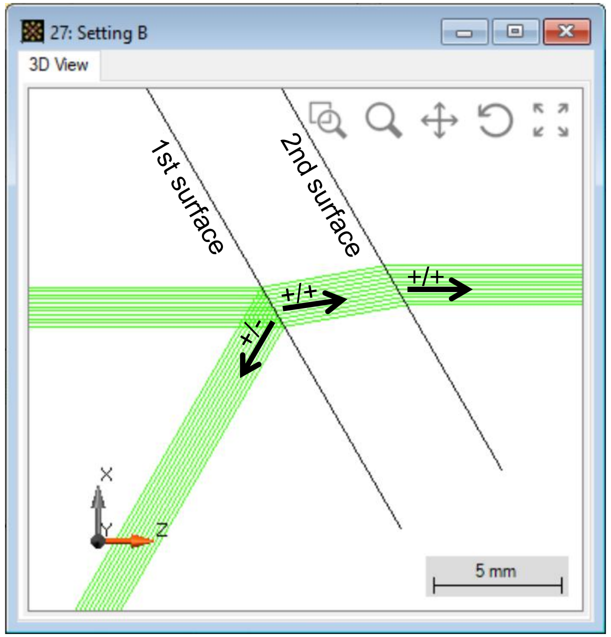
# Surface Channels

## Setting A



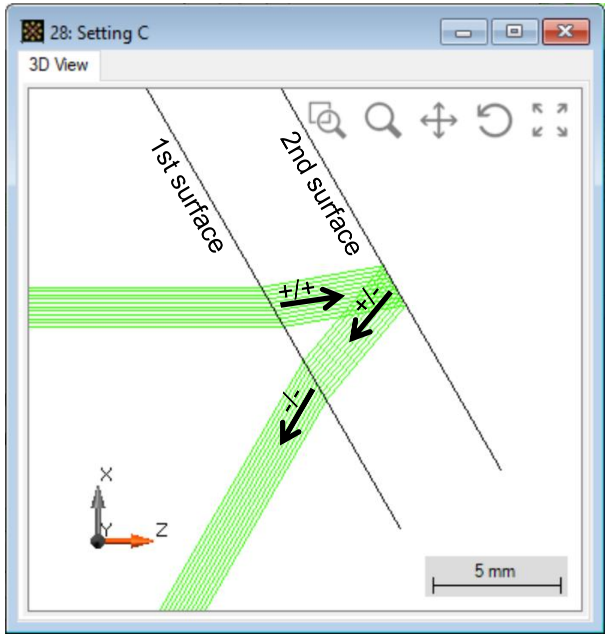

| Surface | +/+ | +/- | -/- | -/+ |
|---|---|---|---|---|
| 1st | × | | | |
| 2nd | × | | | |

## Setting B




| Surface | +/+ | +/- | -/- | -/+ |
|---|---|---|---|---|
| 1st | × | × | | |
| 2nd | × | | | |

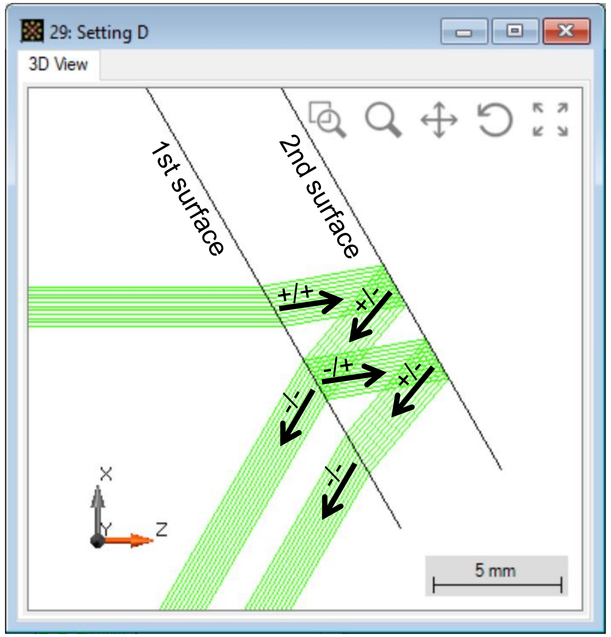
## Setting C




| Surface | +/+ | +/- | -/- | -/+ |
|---|---|---|---|---|
| 1st | × | | × | |
| 2nd | | × | | |

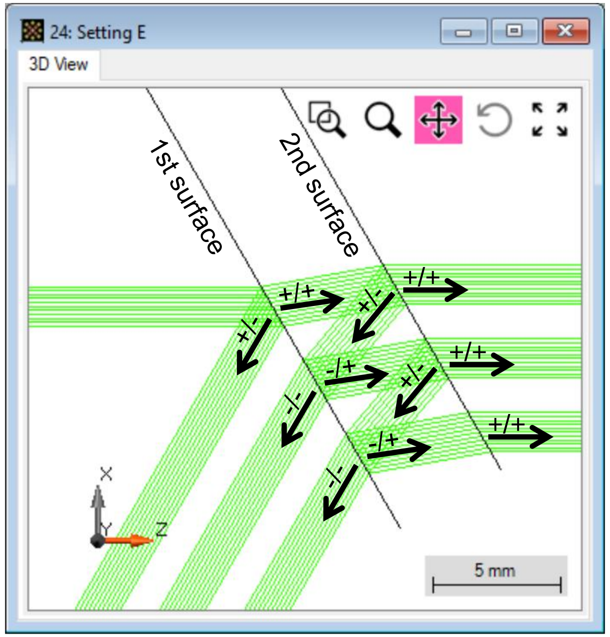
# Surface Channels

## Setting D




| Surface | +/+ | +/- | -/- | -/+ |
|---|---|---|---|---|
| 1st | × | | × | × |
| 2nd | | × | | |

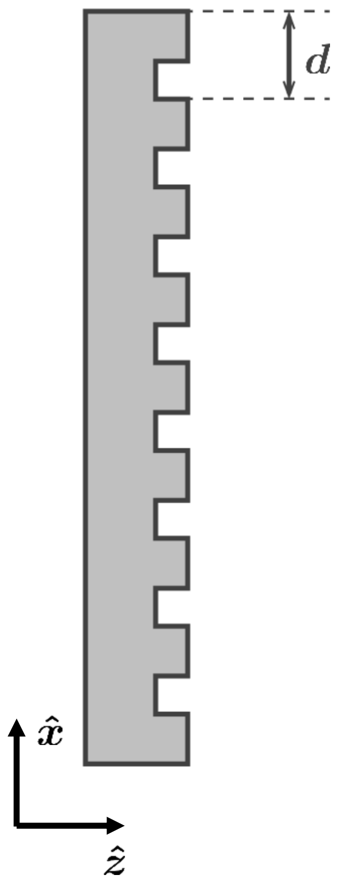
## Setting E



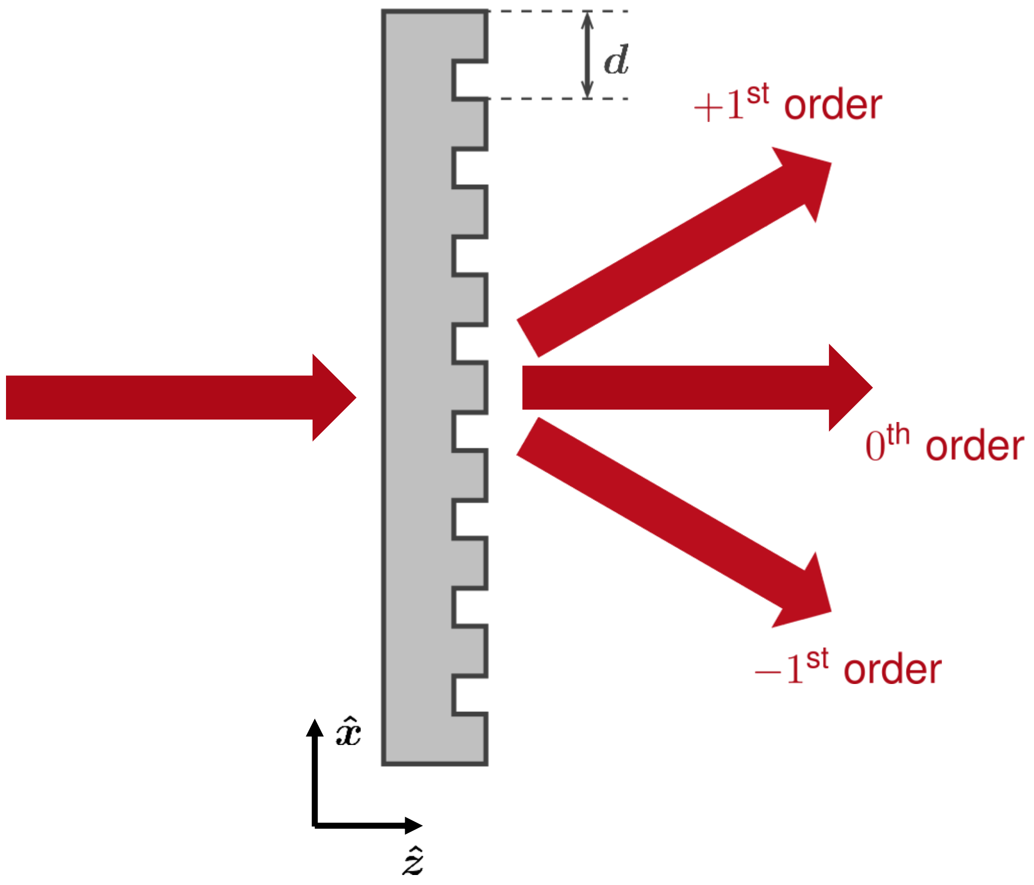

| Surface | +/+ | +/- | -/- | -/+ |
|---|---|---|---|---|
| 1st | × | × | × | × |
| 2nd | × | × | × | × |

Note: an activated channel does not necessarily lead to corresponding light path(s). E.g., the -/- and -/+ channel of 2nd interface do not influence the tracing, because there is no backward incidence.

# Grating Channels



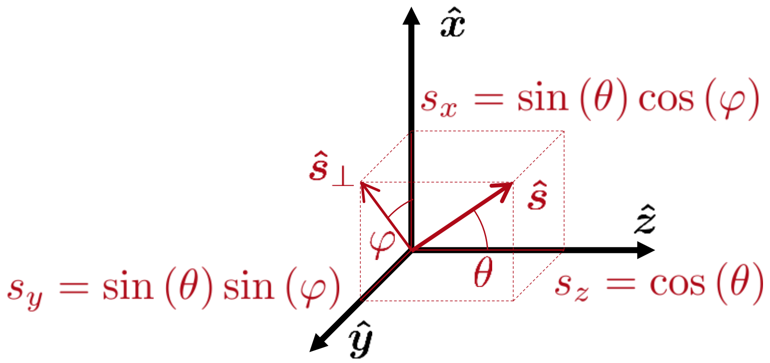
# Grating Channels

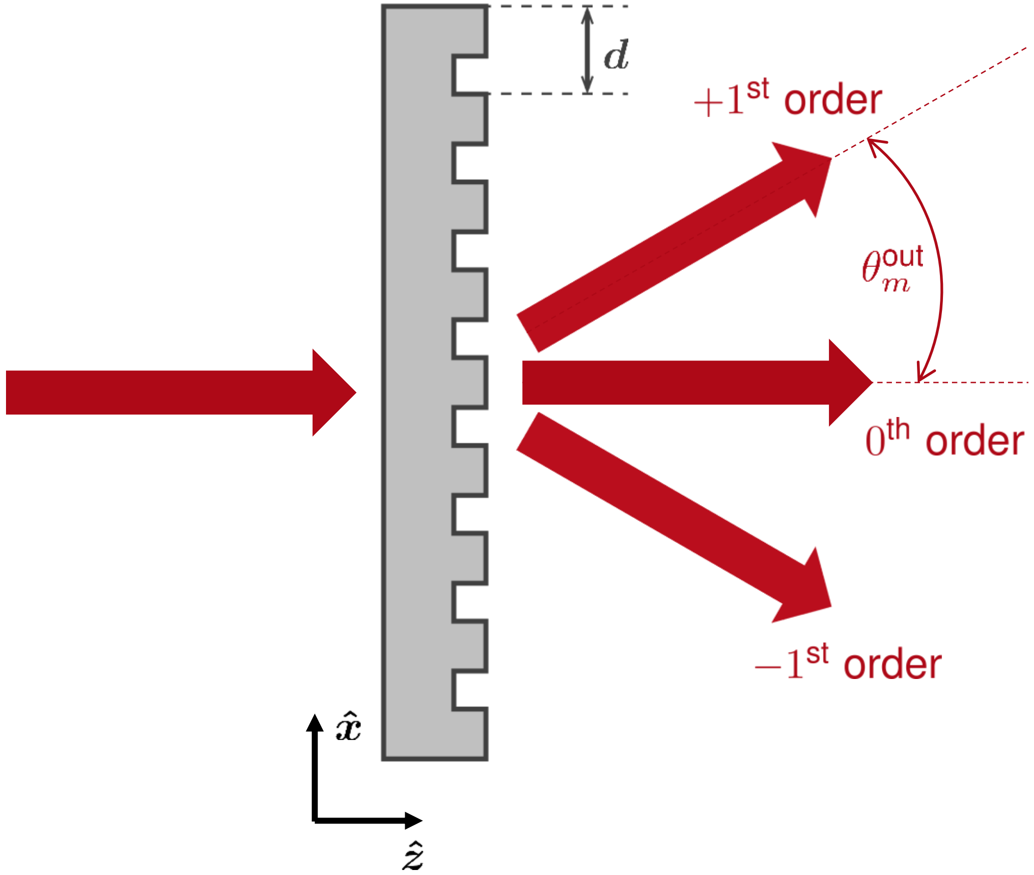


# Grating Channels

Grating equation:

- $\boldsymbol{\kappa}_m^{\text{out}} = \boldsymbol{\kappa}^{\text{in}} + m\boldsymbol{\Delta\kappa}$,
- where $\boldsymbol{\kappa}_m^{\text{out}} = \boldsymbol{k}_{m,\perp}^{\text{out}} = \left(k_{m,x}^{\text{out}}, k_{m,y}^{\text{out}}\right) = nk_0\hat{\boldsymbol{s}}_{m,\perp}^{\text{out}}$ (with $m$ the order),
- $\boldsymbol{\kappa}^{\text{in}} = \boldsymbol{k}_{\perp}^{\text{in}} = \left(k_x^{\text{in}}, k_y^{\text{in}}\right) = nk_0\hat{\boldsymbol{s}}_{\perp}^{\text{in}}$ and
- $\boldsymbol{\Delta\kappa} = \frac{2\pi}{\boldsymbol{d}}$

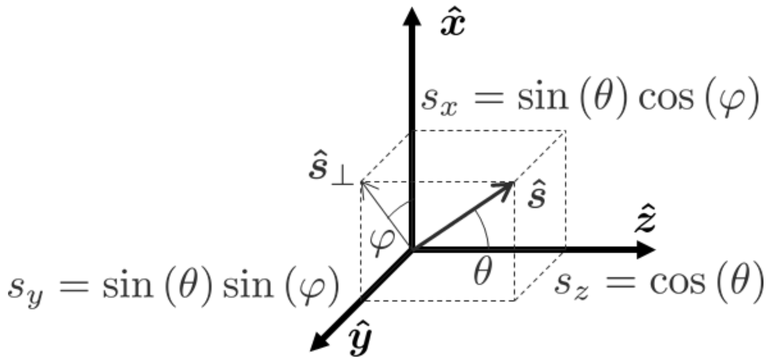


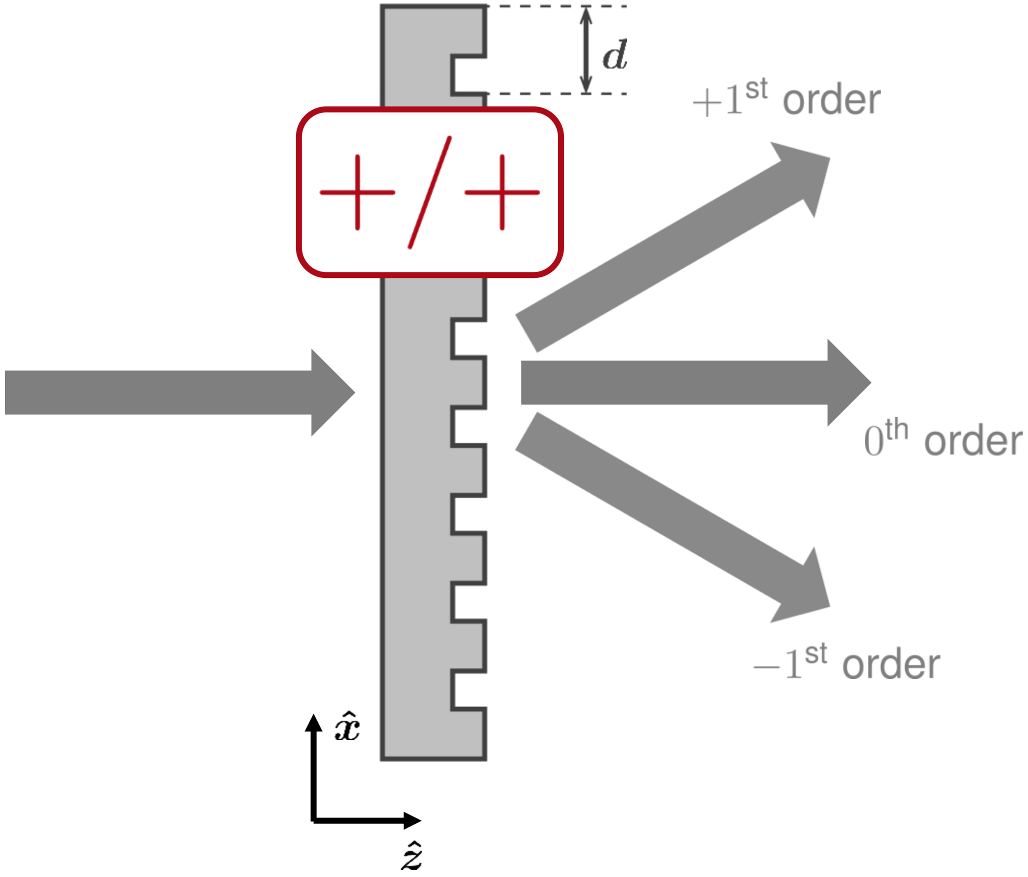



# Grating Channels

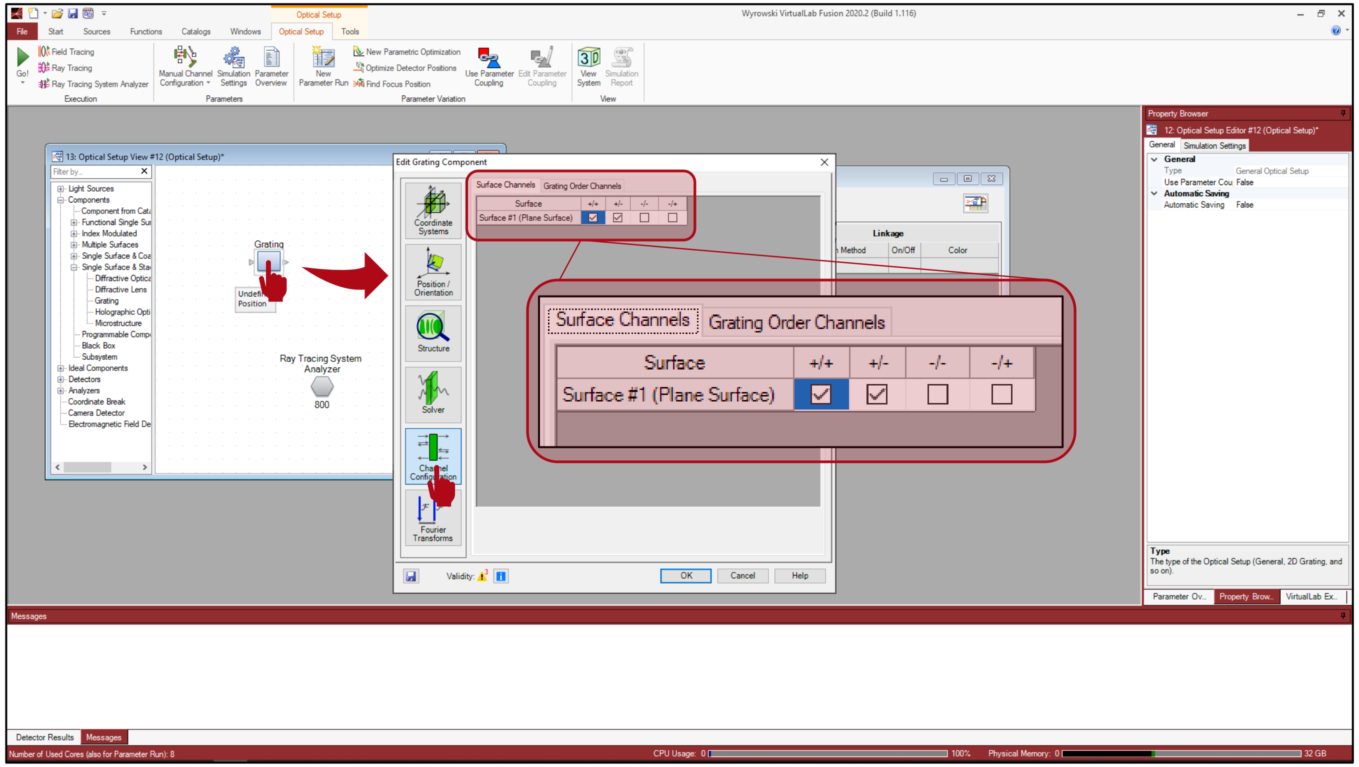
Grating equation:

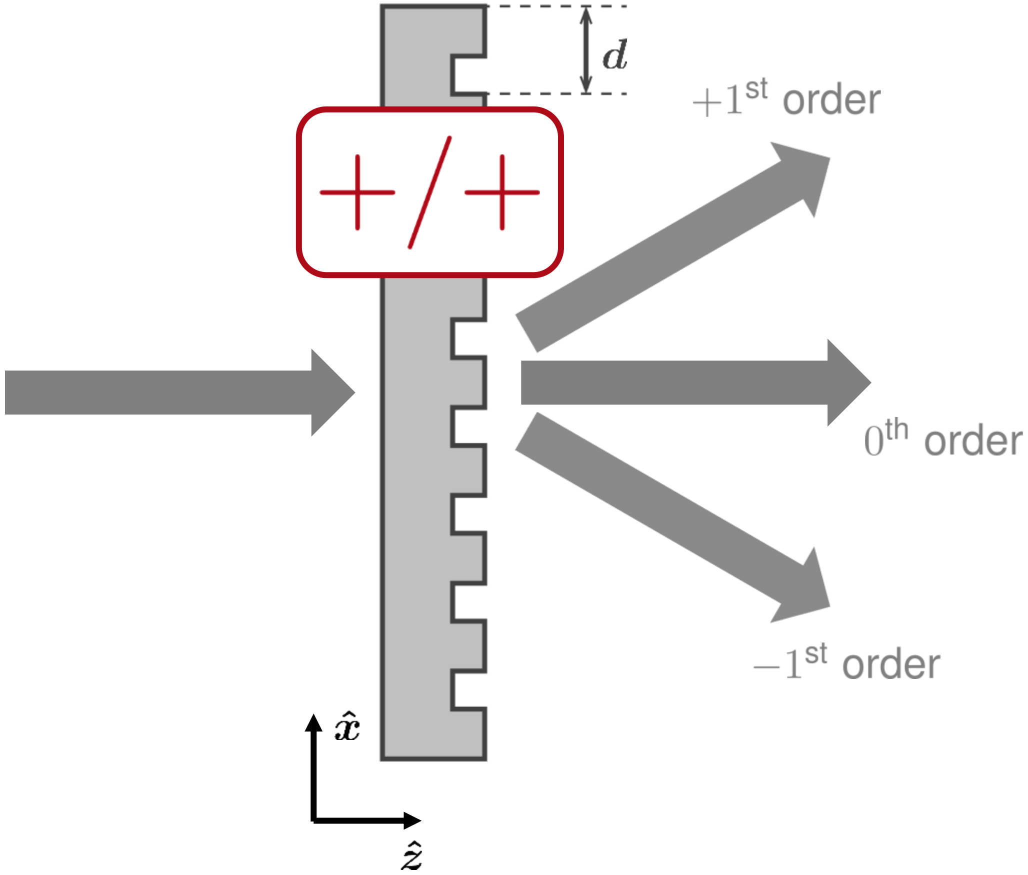
- $\boldsymbol{\kappa}_m^{\text{out}} = \boldsymbol{\kappa}^{\text{in}} + m\boldsymbol{\Delta\kappa}$,
- where $\boldsymbol{\kappa}_m^{\text{out}} = \boldsymbol{k}_{m,\perp}^{\text{out}} = \left(k_{m,x}^{\text{out}}, k_{m,y}^{\text{out}}\right) = nk_0\hat{\boldsymbol{s}}_{m,\perp}^{\text{out}}$ (with $m$ the order),
- $\boldsymbol{\kappa}^{\text{in}} = \boldsymbol{k}_{\perp}^{\text{in}} = \left(k_x^{\text{in}}, k_y^{\text{in}}\right) = nk_0\hat{\boldsymbol{s}}_{\perp}^{\text{in}}$ and
- $\boldsymbol{\Delta\kappa} = \frac{2\pi}{\boldsymbol{d}}$



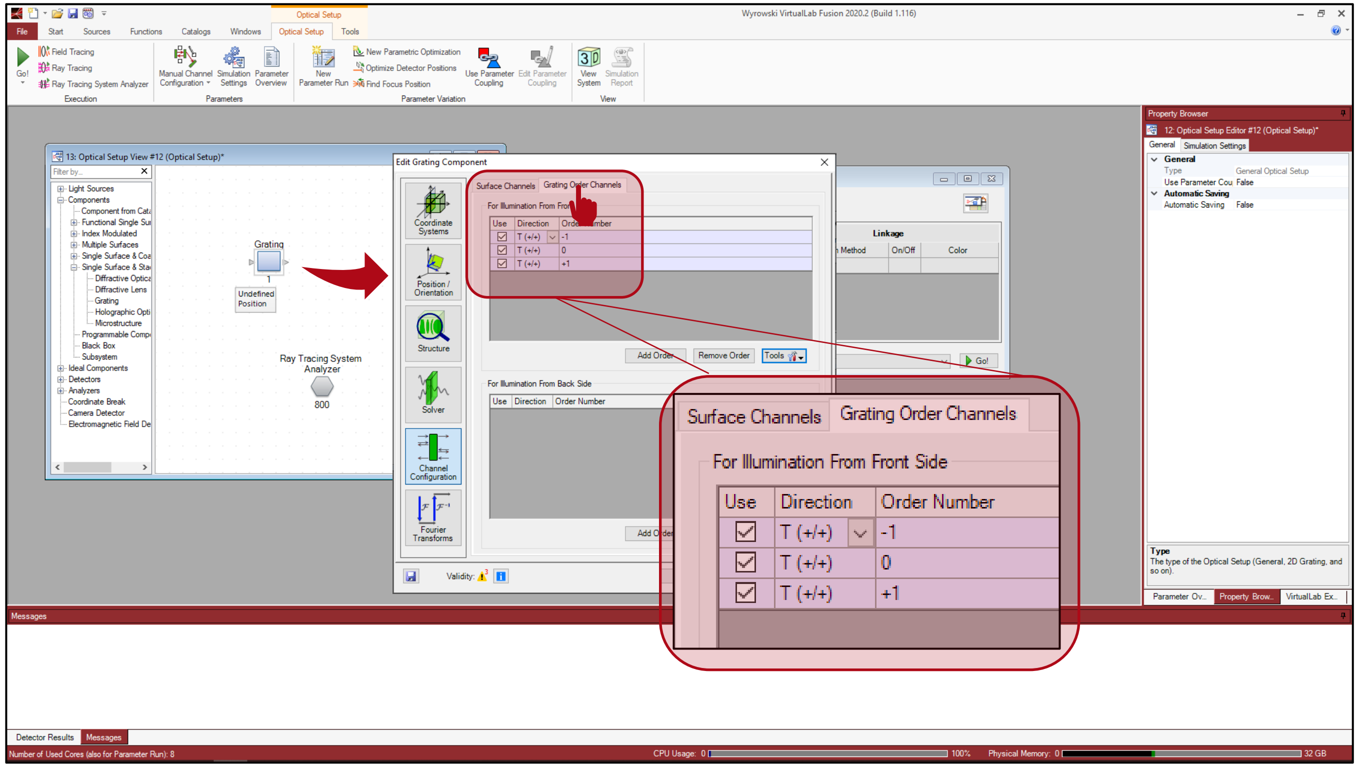



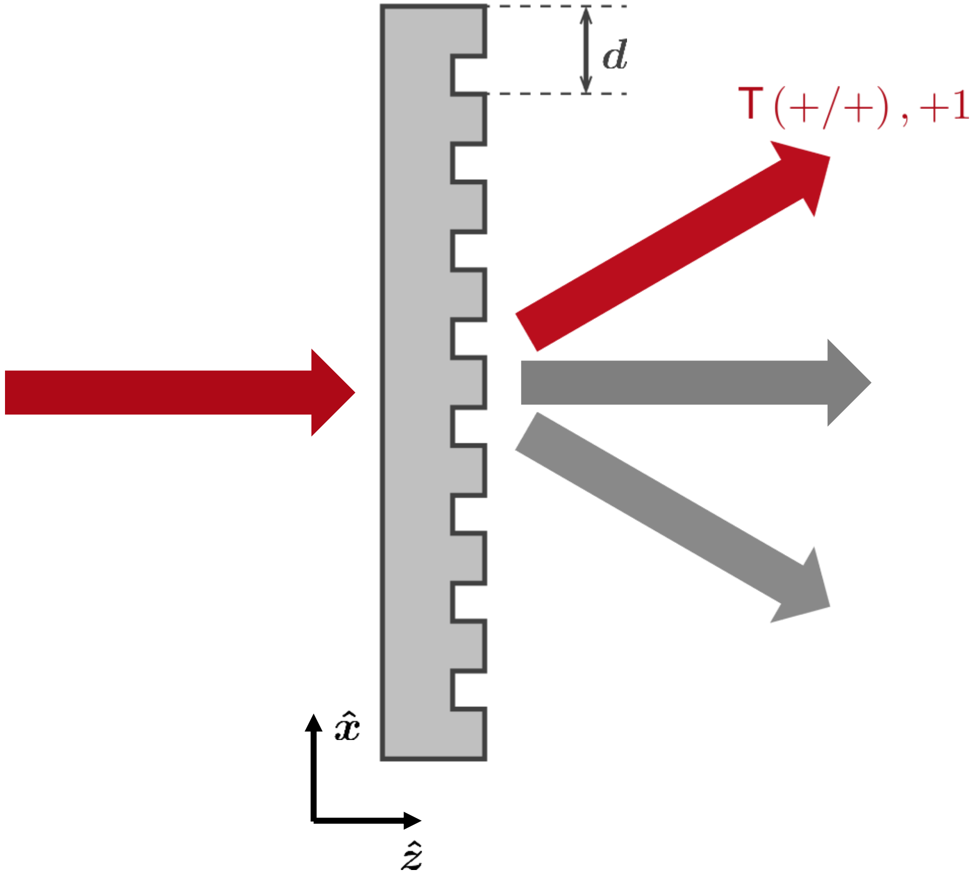
# Grating Channels



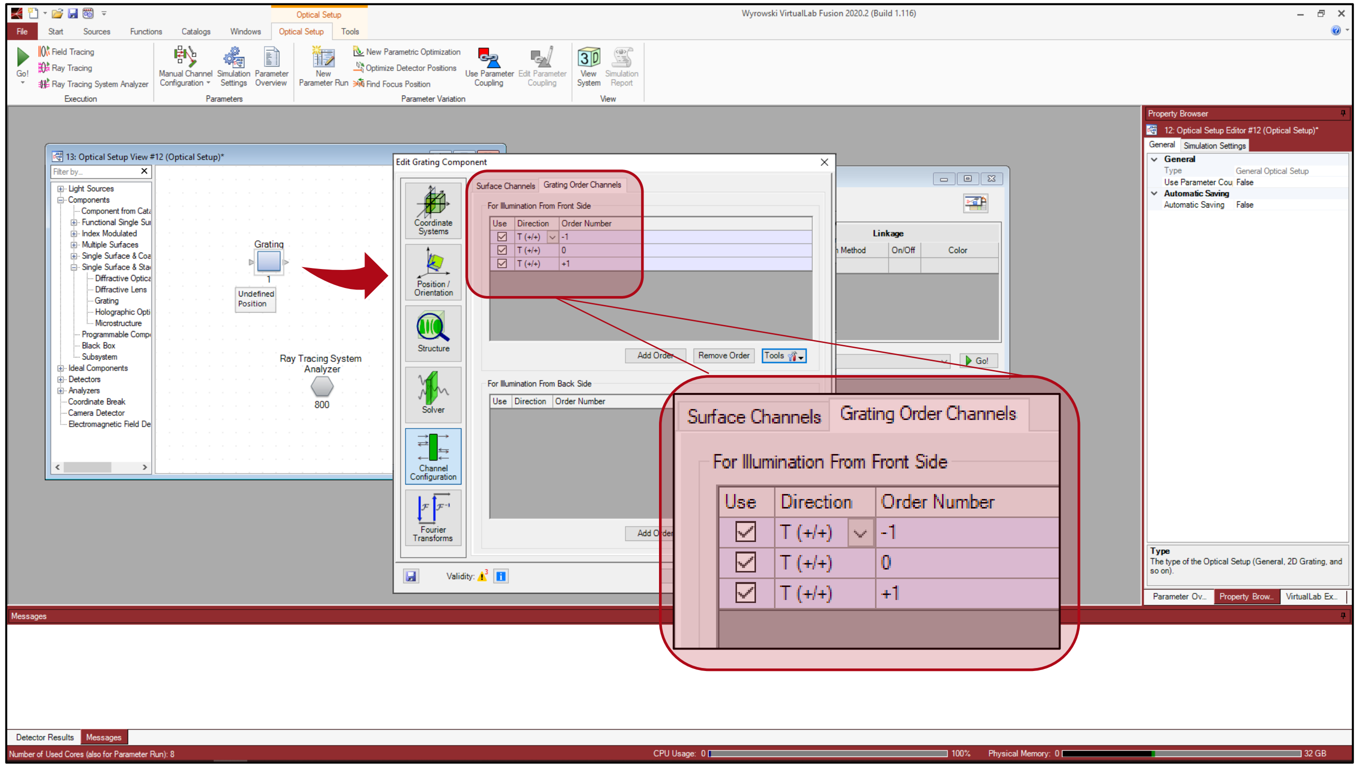



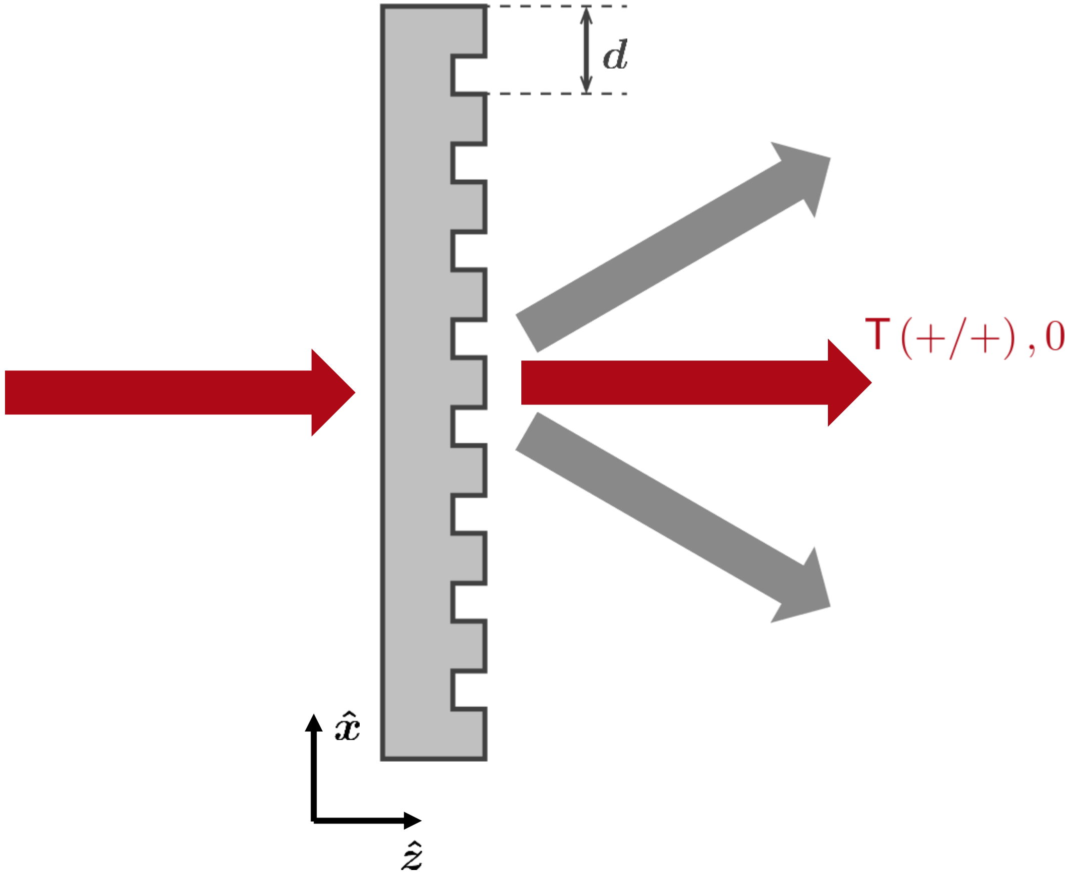
# Grating Channels



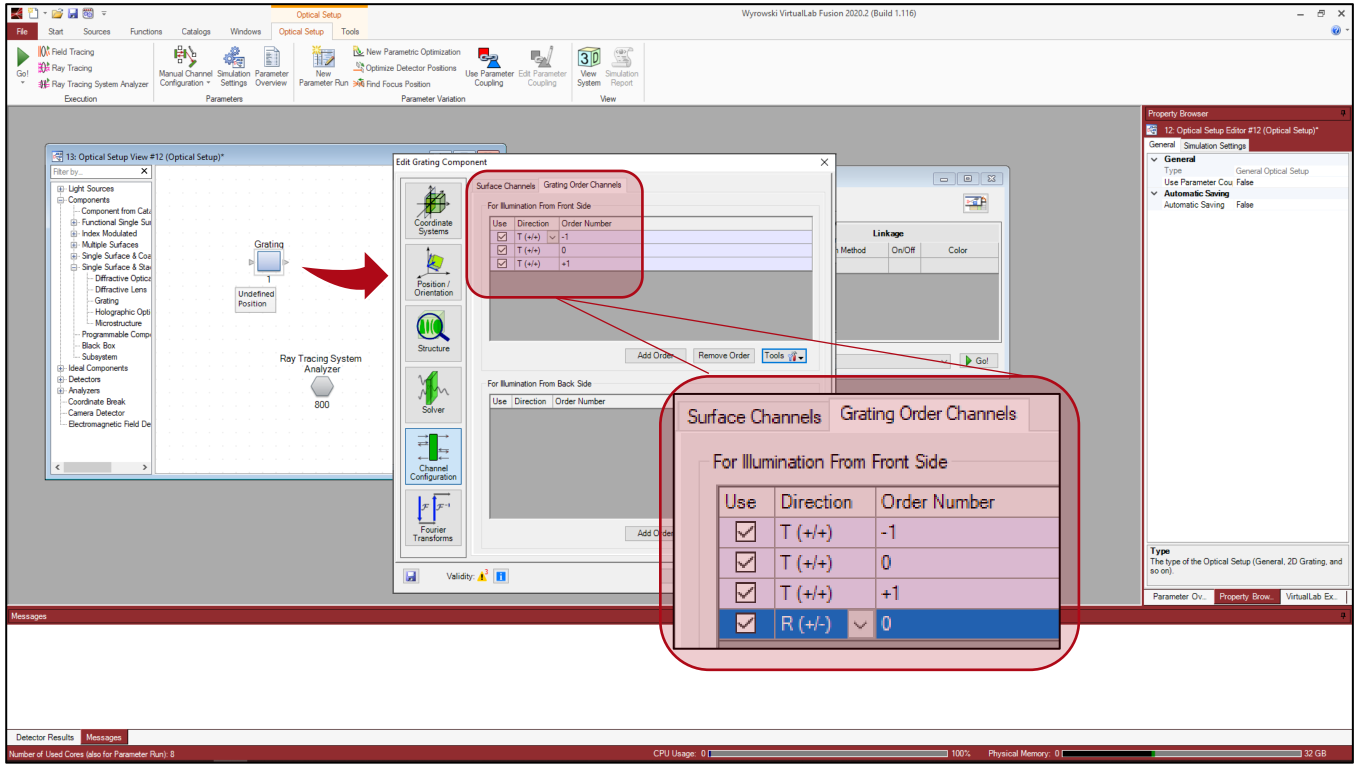



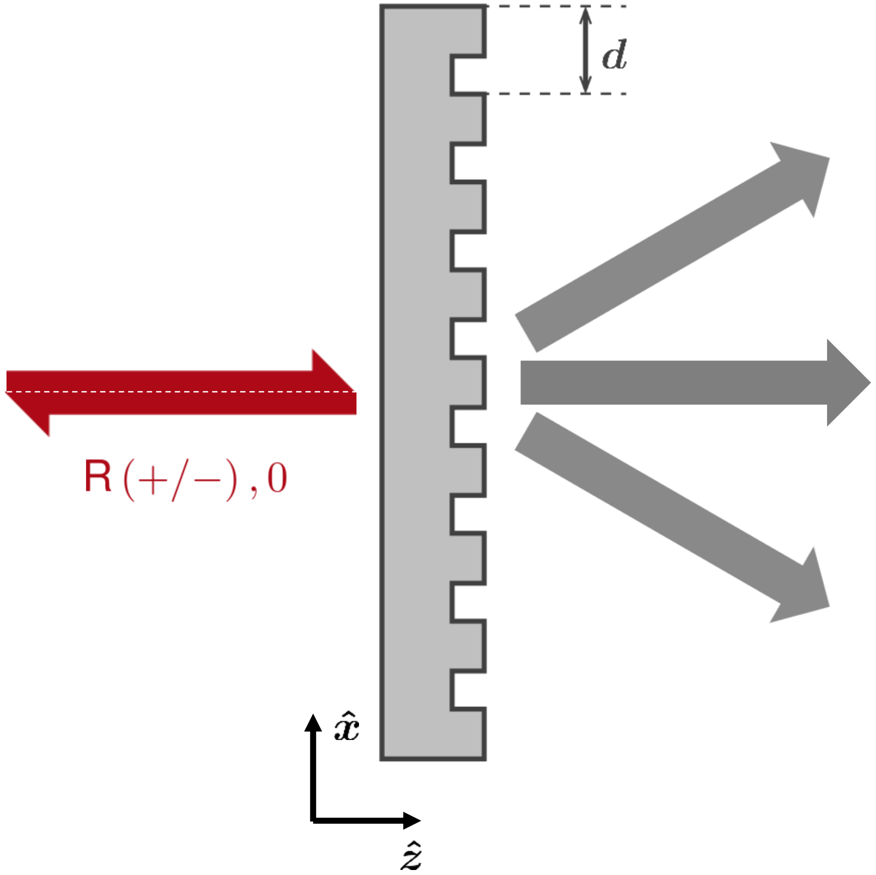
# Grating Channels



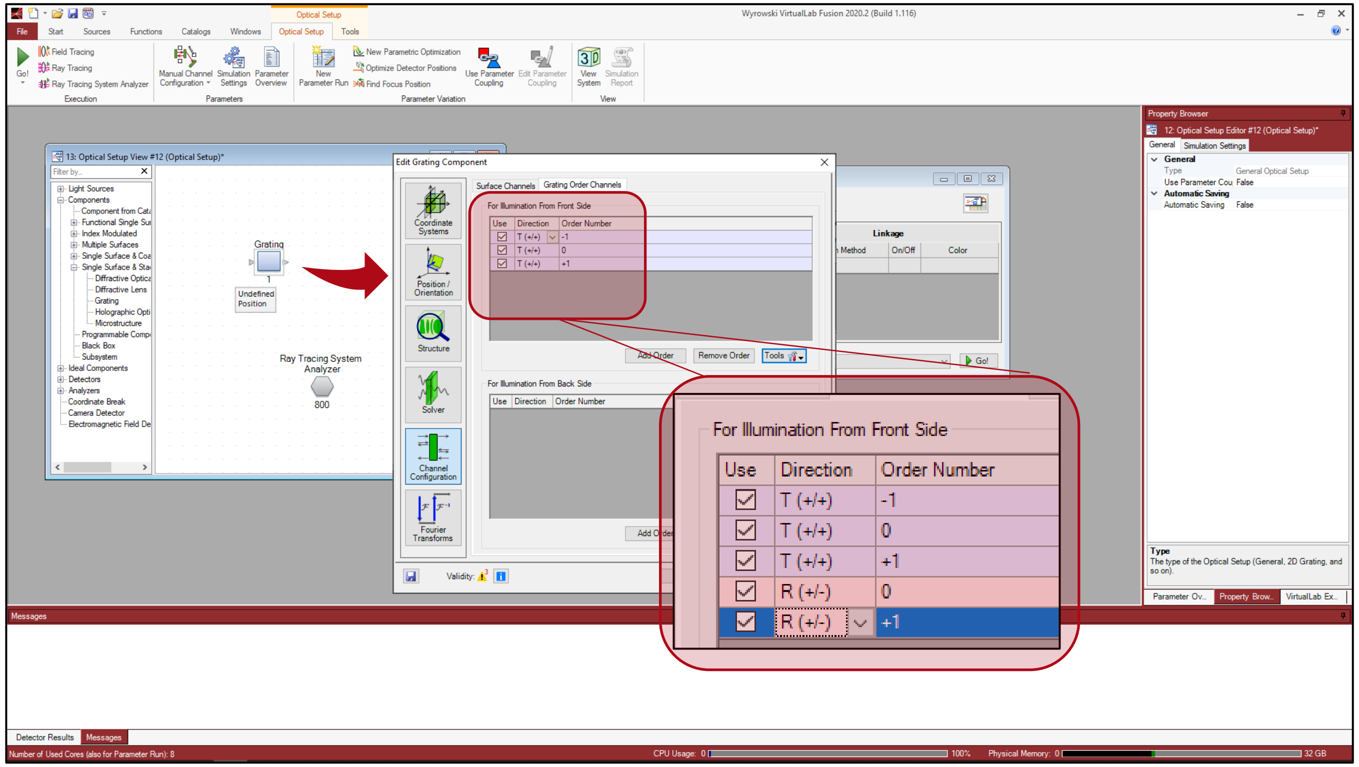



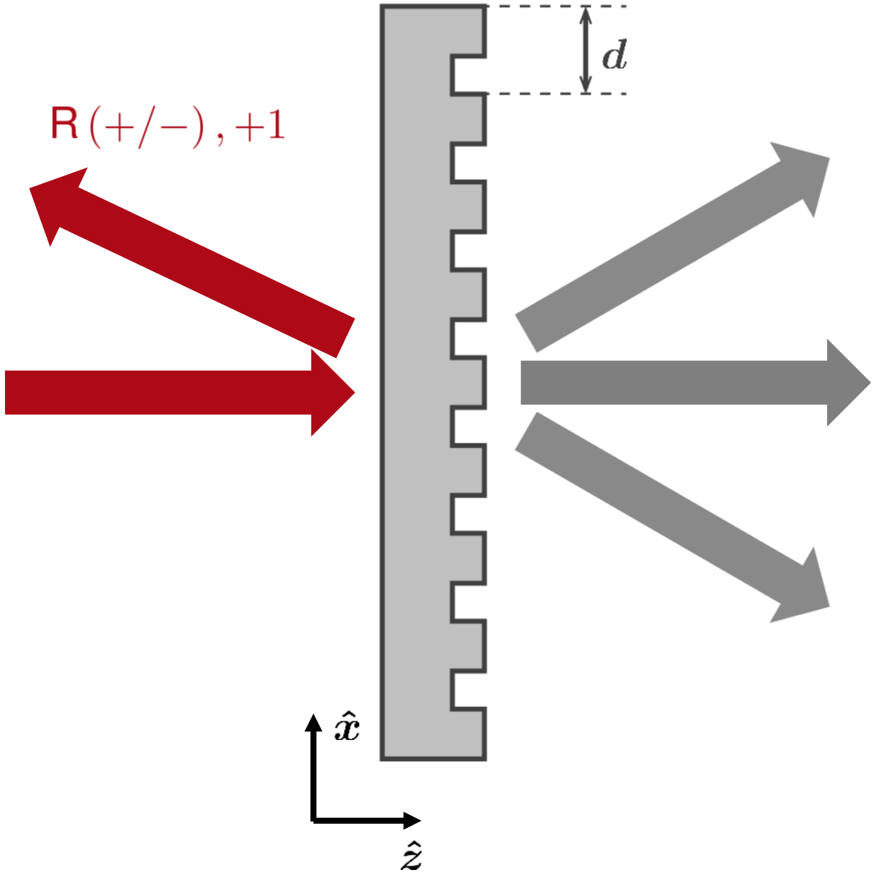
# Grating Channels



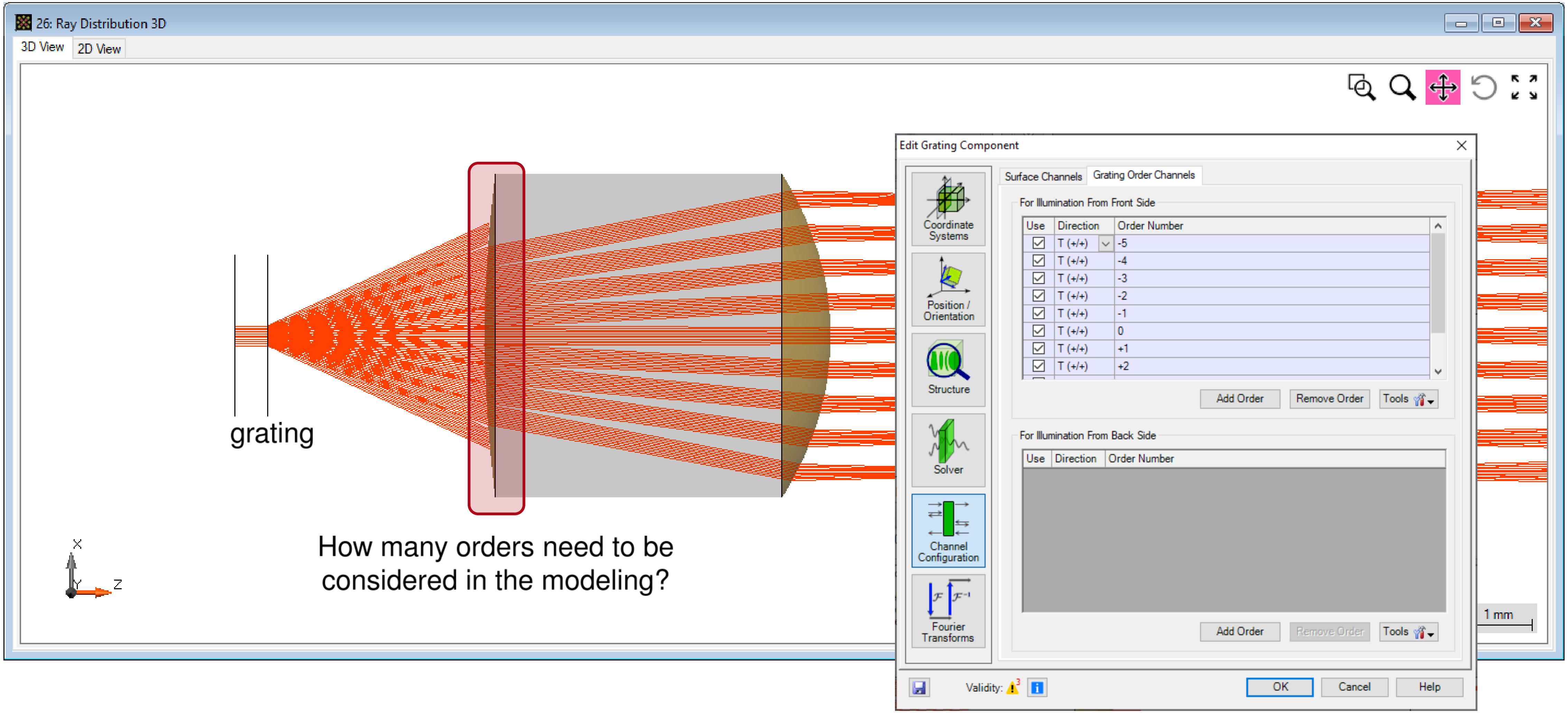



# Grating Channels



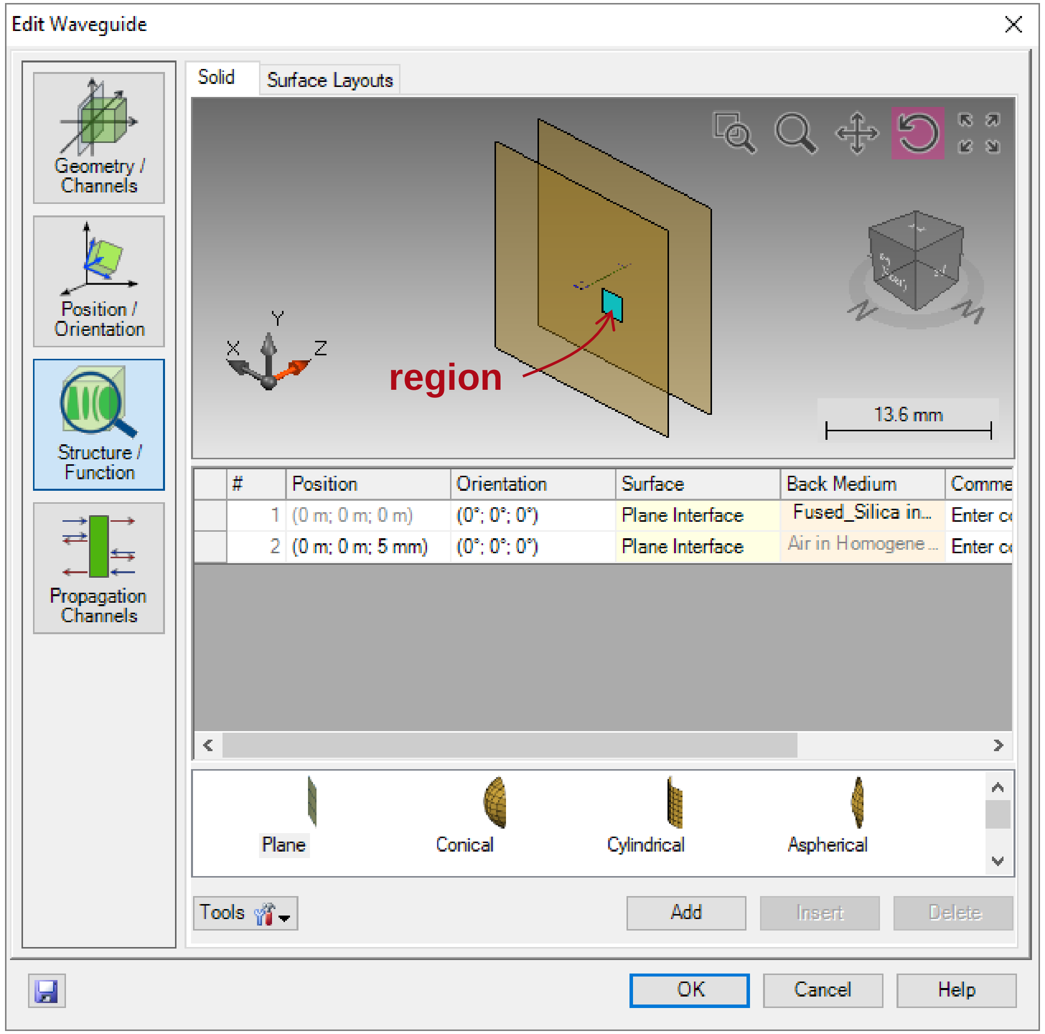



# Grating Channels in Abbe's Experiment



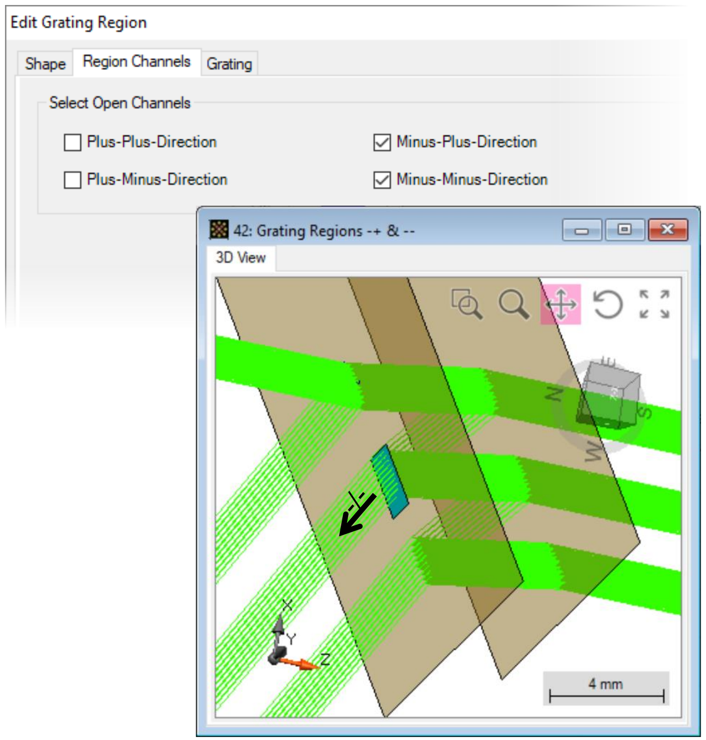
# Lateral Channels

- Region(s) on surface
  - It is possible to define individual Regions on a surface and define their optical properties individually, including the channel settings.



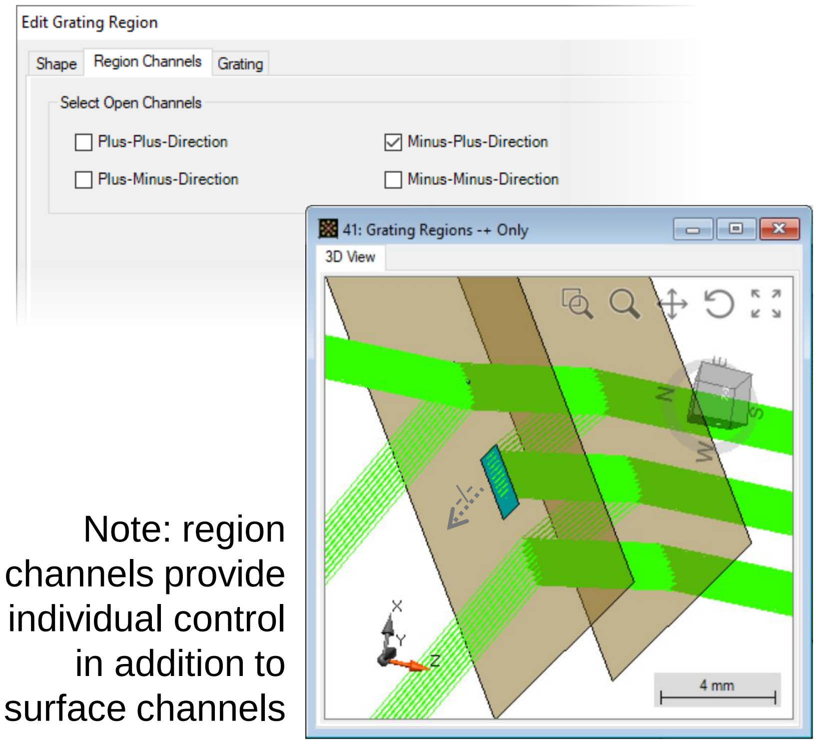
# Lateral Channels

- Region definition
  - Set up the channels for this region, following the same rule as for the surfaces.




region channels -/+, -/- on

Note: region channels provide individual control in addition to surface channels



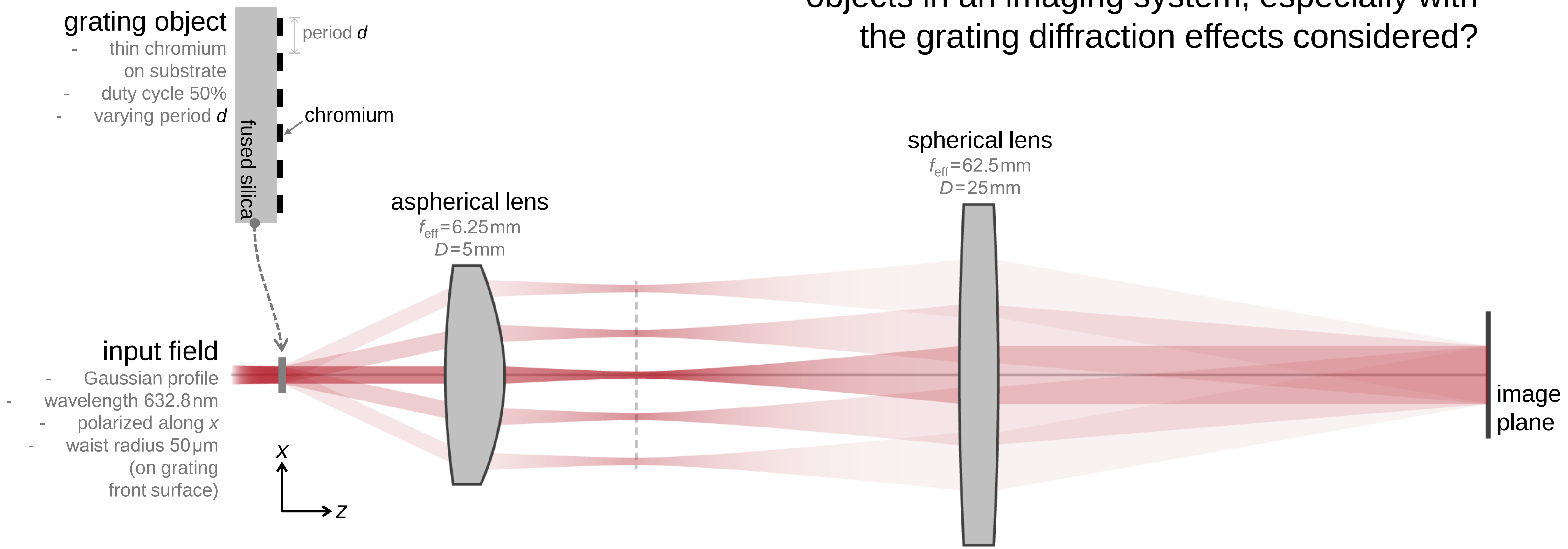

region channel -/+ on

# Demonstration of Abbe's Theory of Image Formation

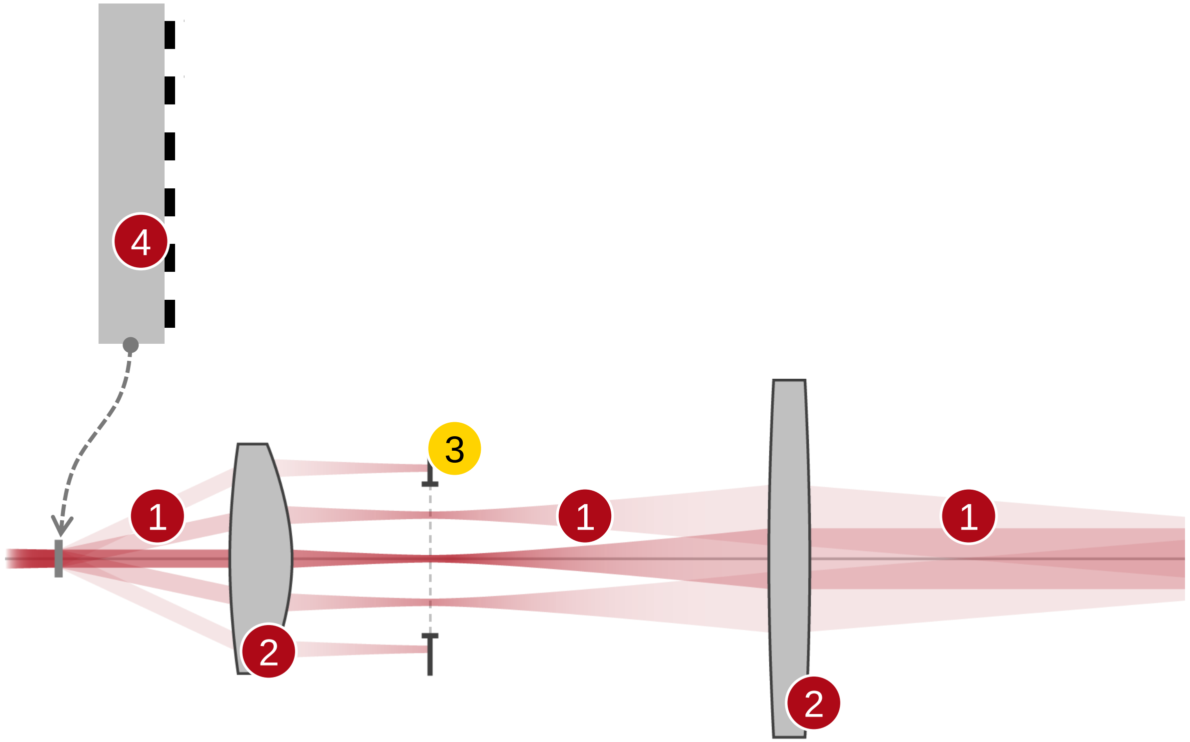
see the full Application Use Case

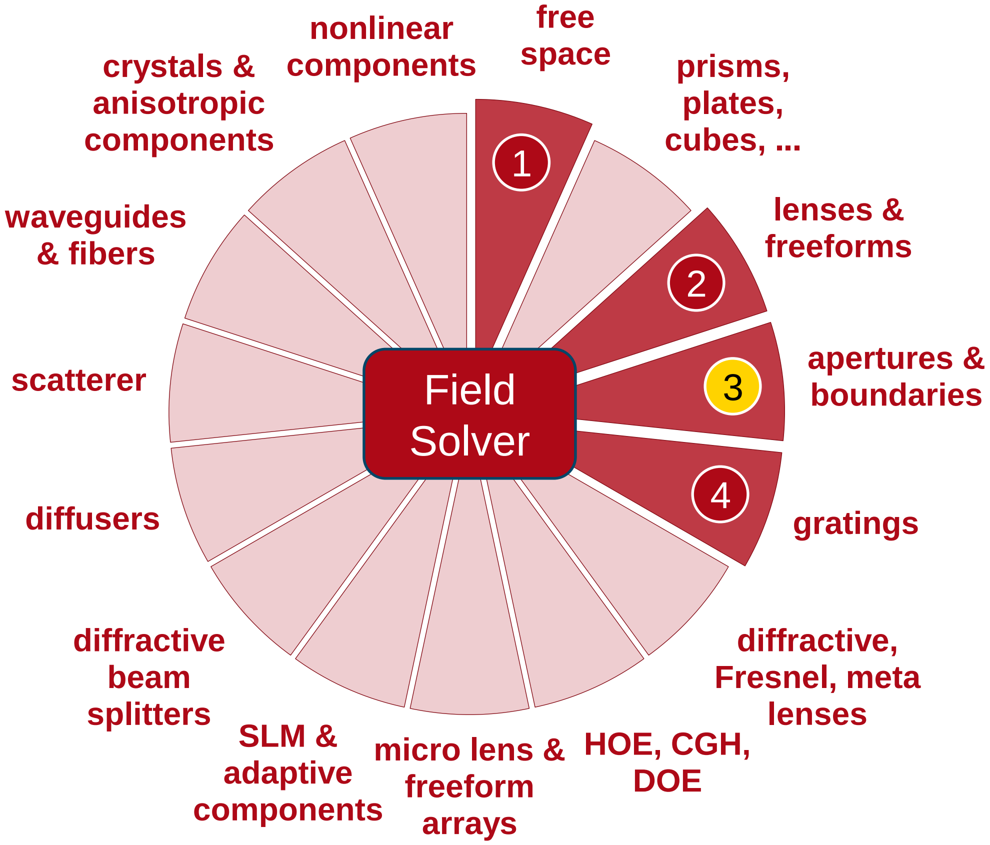
# Modeling Task – Imaging with Varying Grating Period

How to simulate the image formation for grating objects in an imaging system, especially with the grating diffraction effects considered?



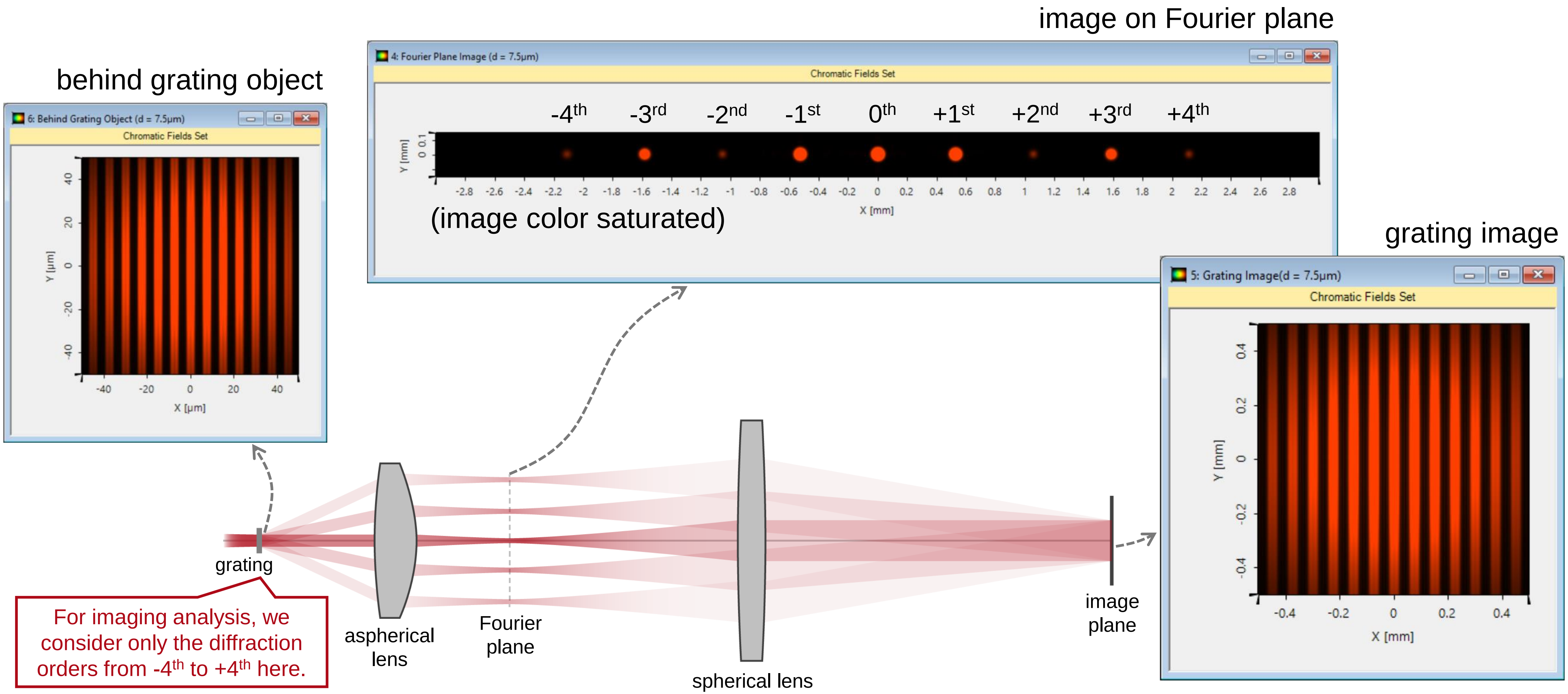
# VirtualLab Fusion Technologies



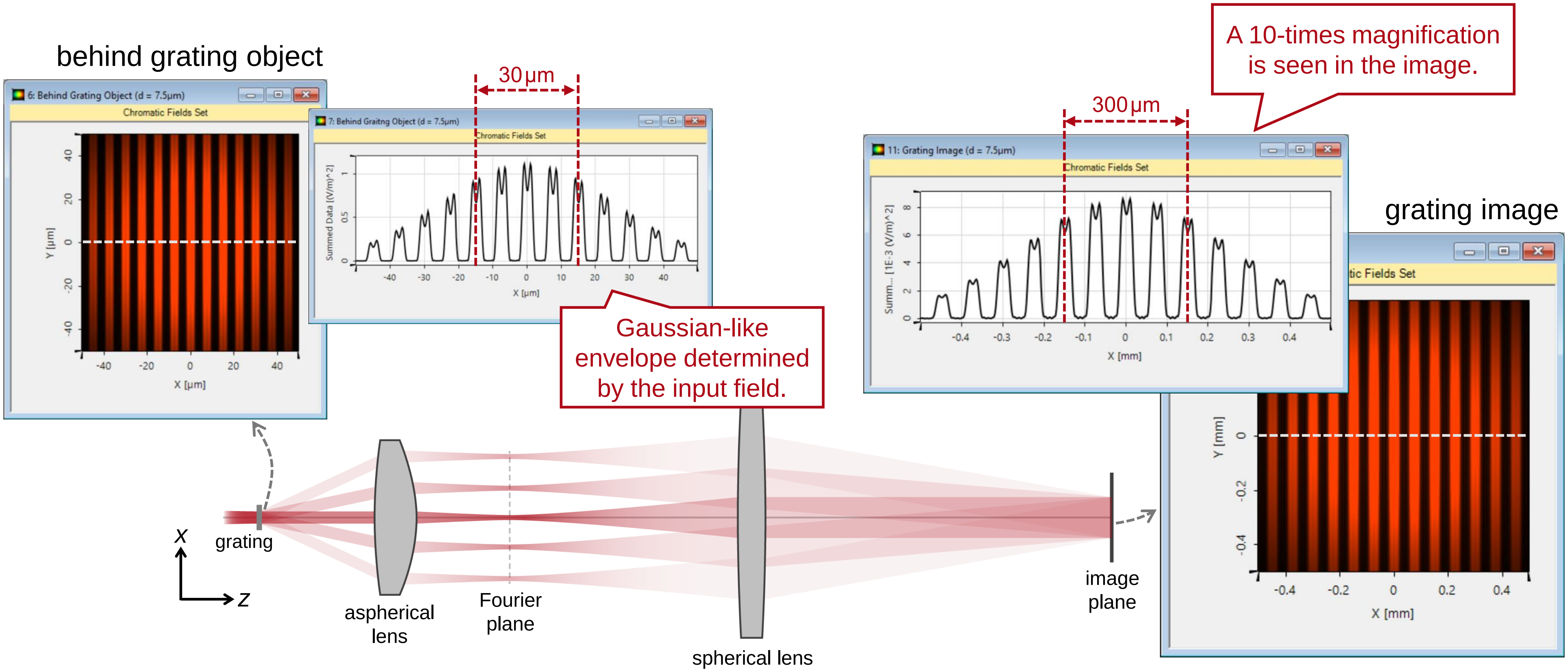




# idealized component

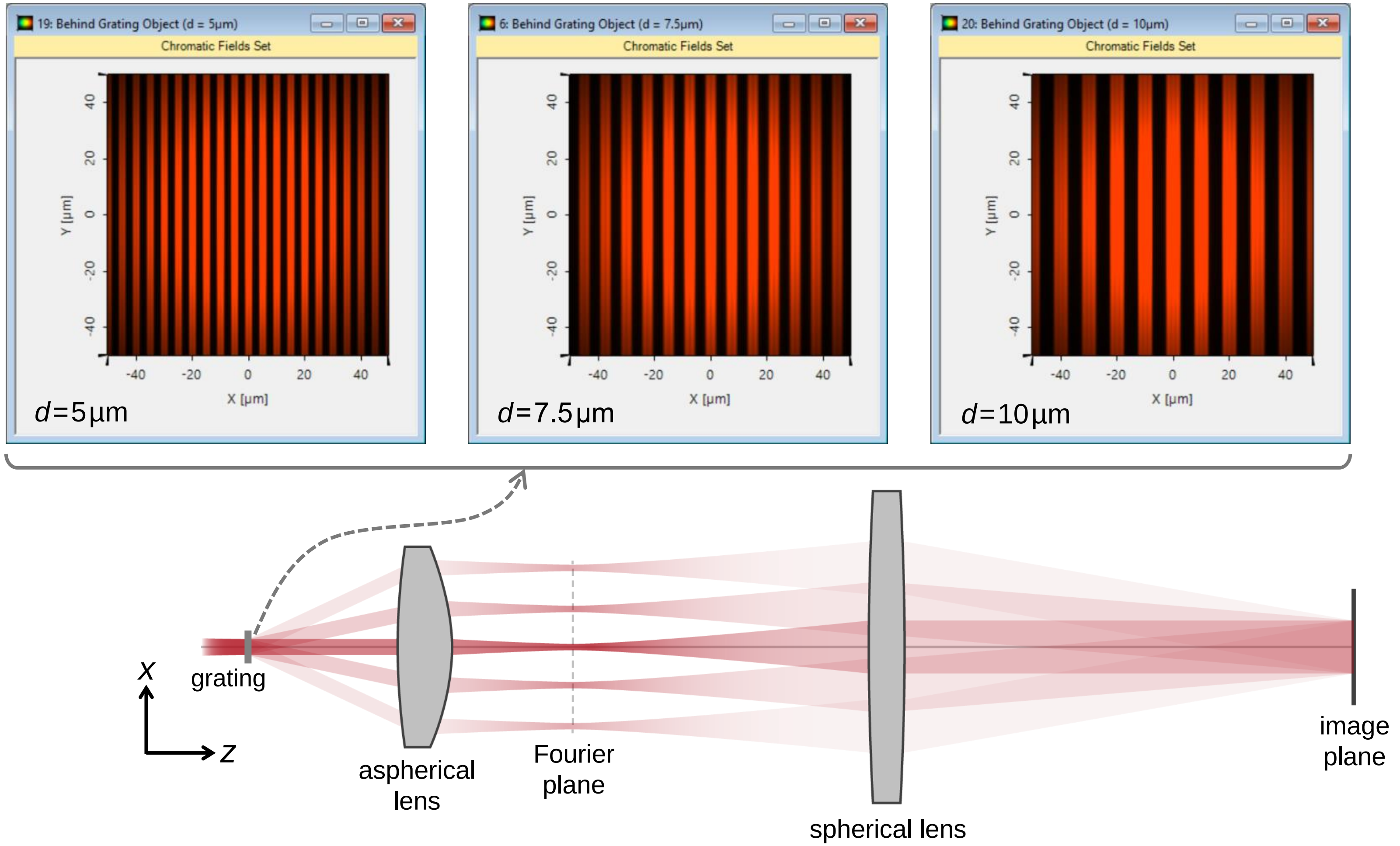
# Image Formation Analysis



# Image Formation Analysis



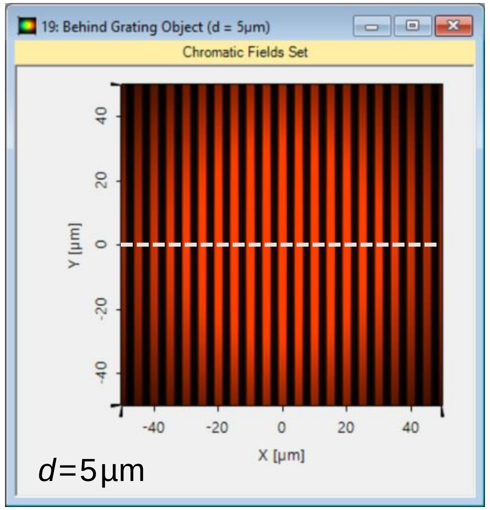
# Behind Grating Objects with Different Periods

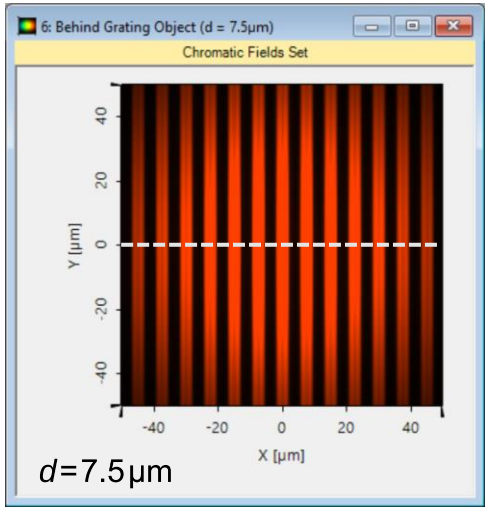


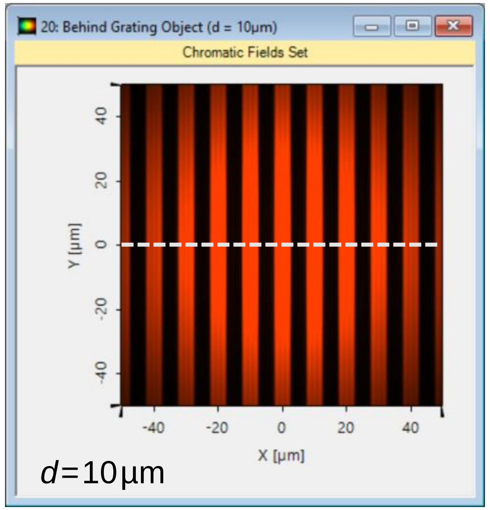

For imaging analysis, we consider only the diffraction orders that will enter the subsequent system:

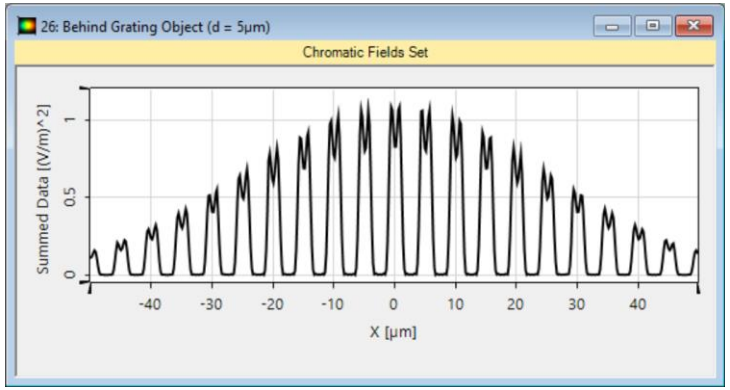
- $d$=5µm: -3rd to +3rd orders
- $d$=7.5µm: -4th to +4th orders
- $d$=10µm: -6th to +6th orders

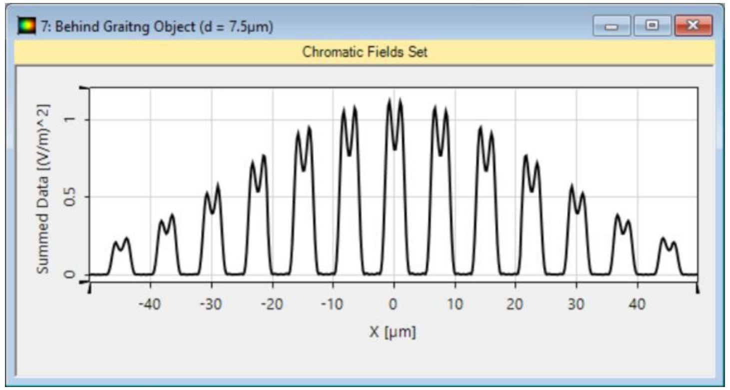
# Behind Grating Objects with Different Periods

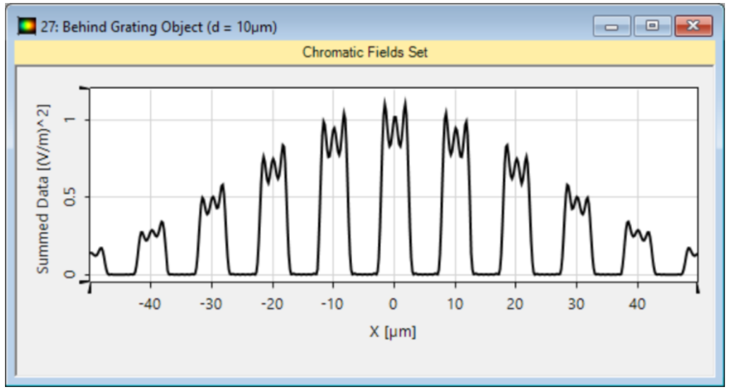


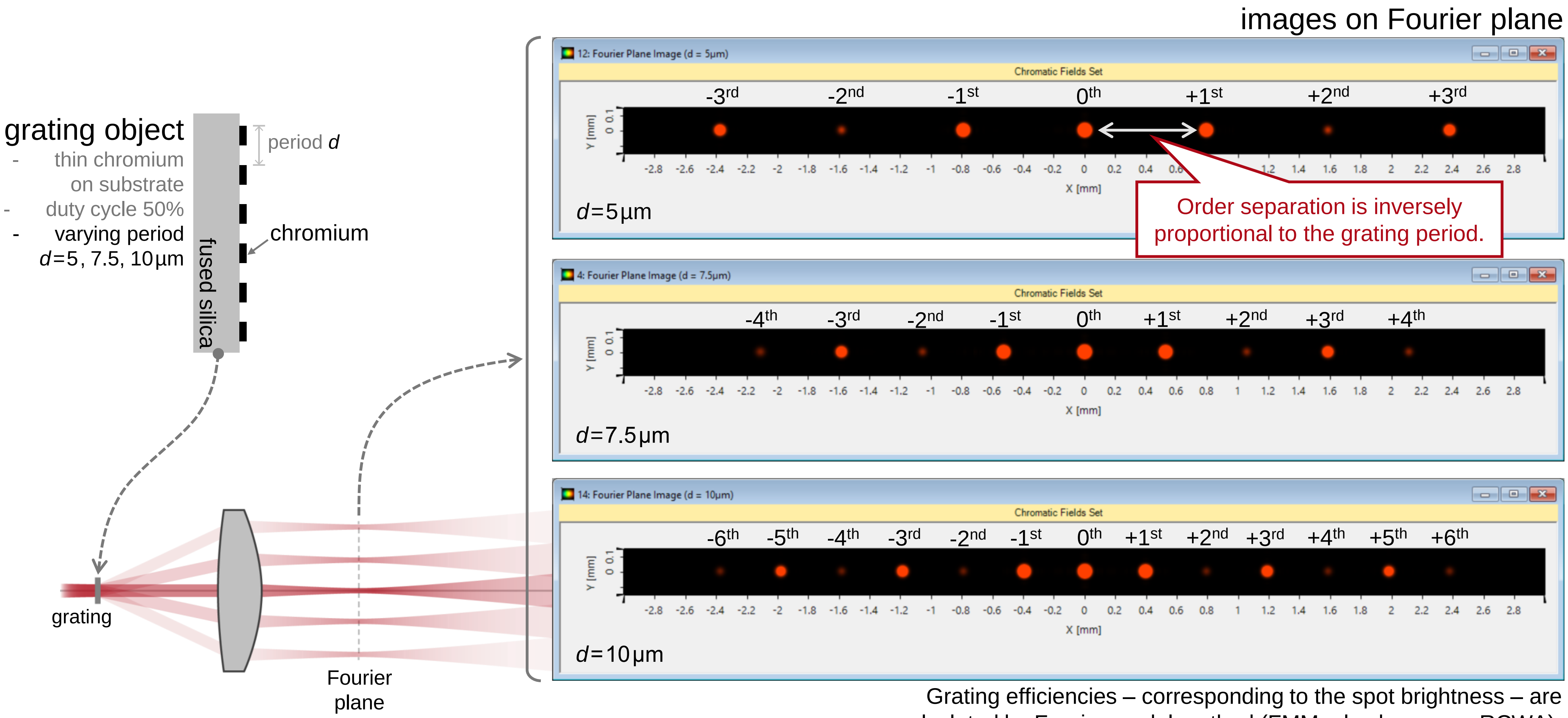



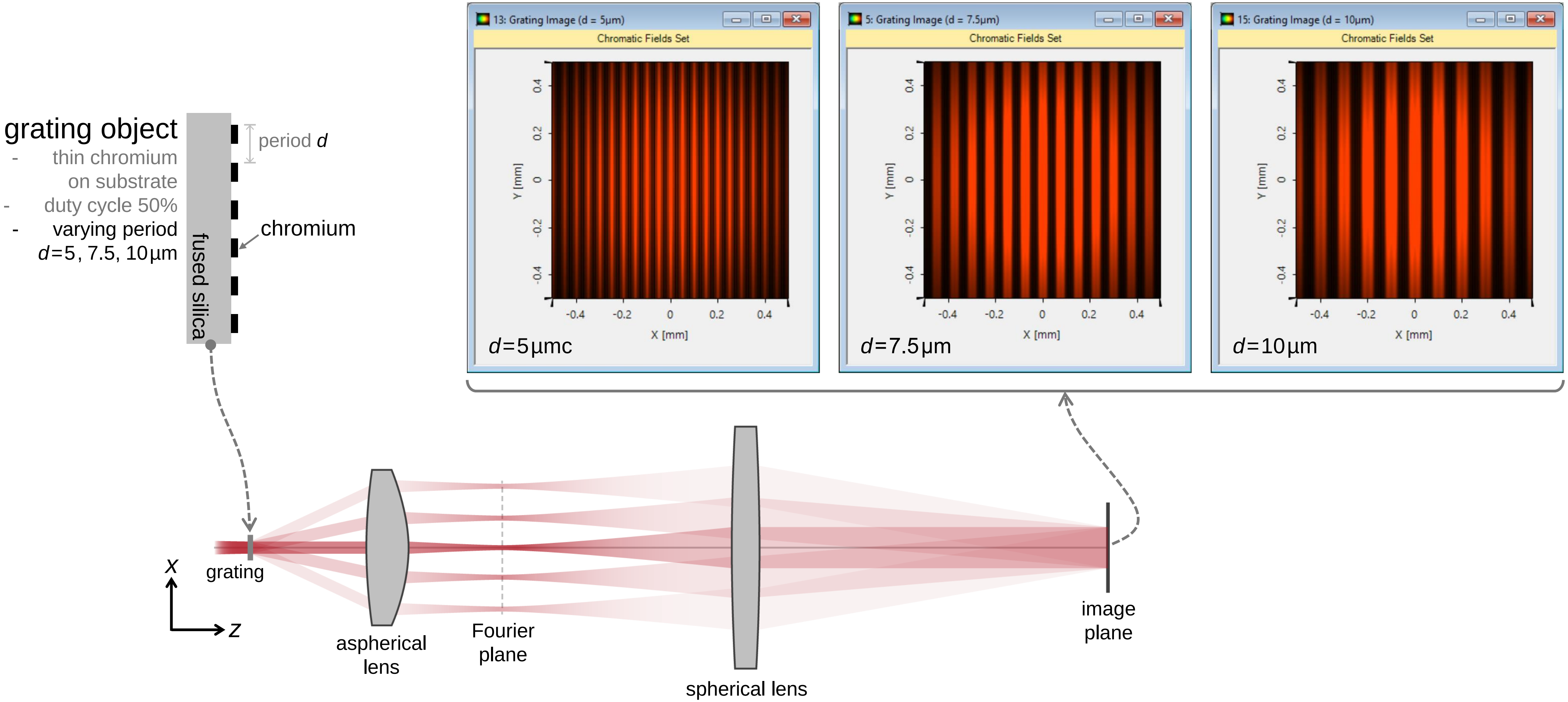



# Fourier Plane Images for Different Periods

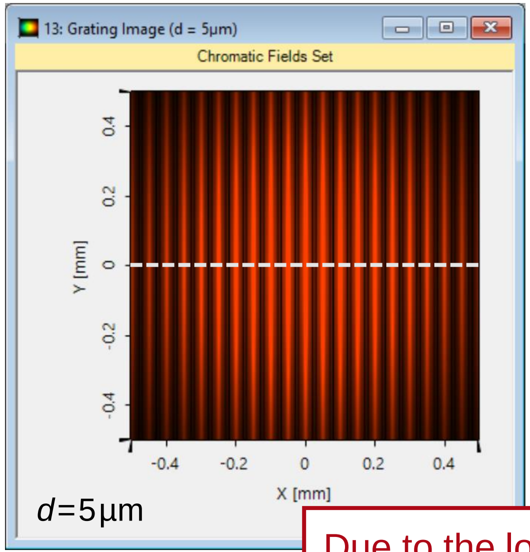


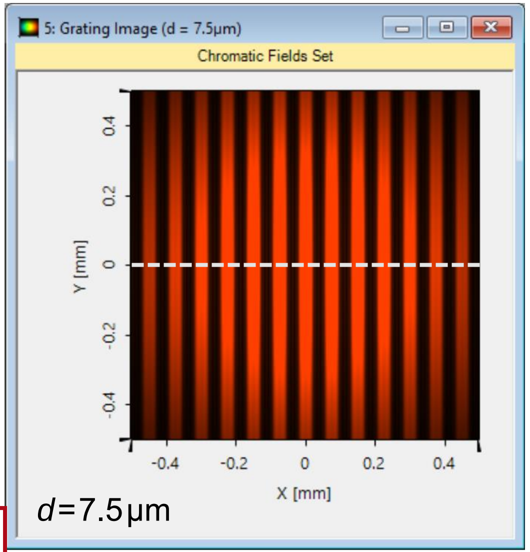

Grating efficiencies – corresponding to the spot brightness – are calculated by Fourier modal method (FMM, also known as RCWA).

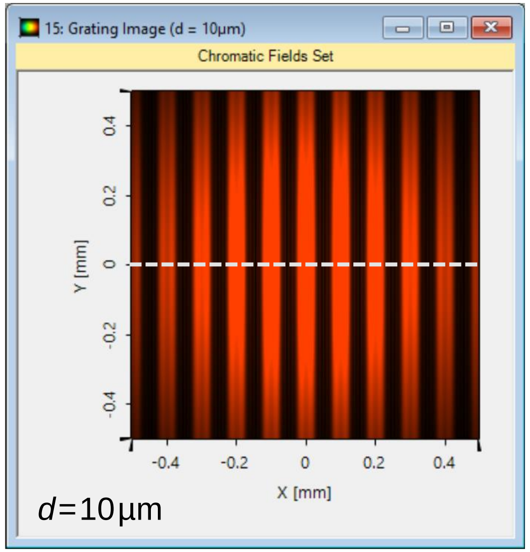
# Grating Images for Different Periods

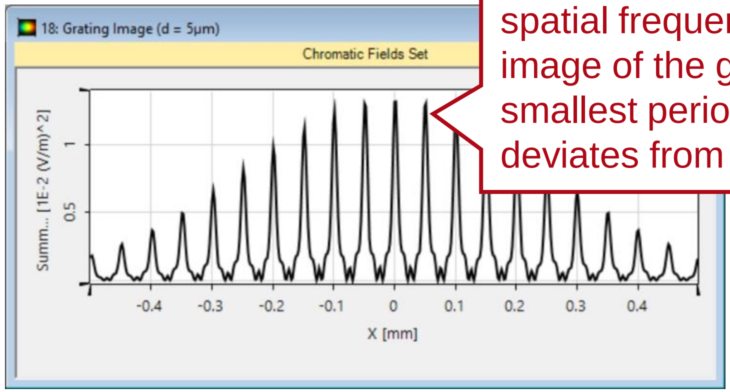


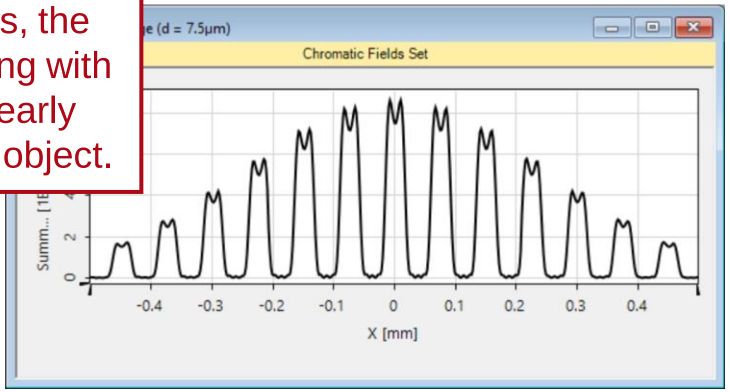
# Grating Images for Different Periods

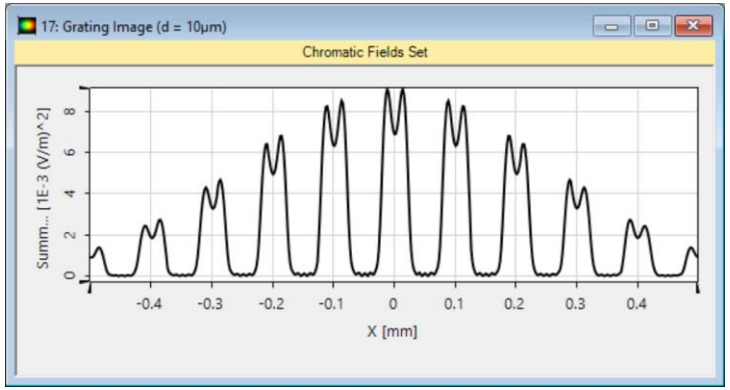


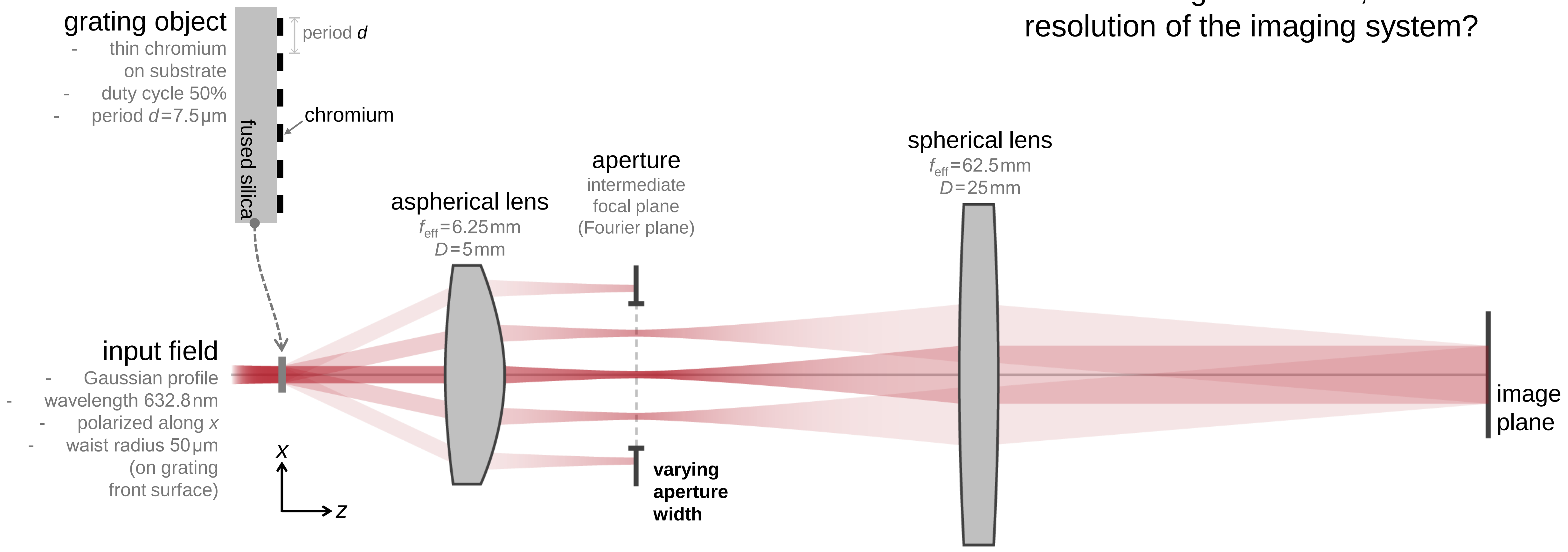




Due to the loss of high spatial frequencies, the image of the grating with smallest period clearly deviates from the object.

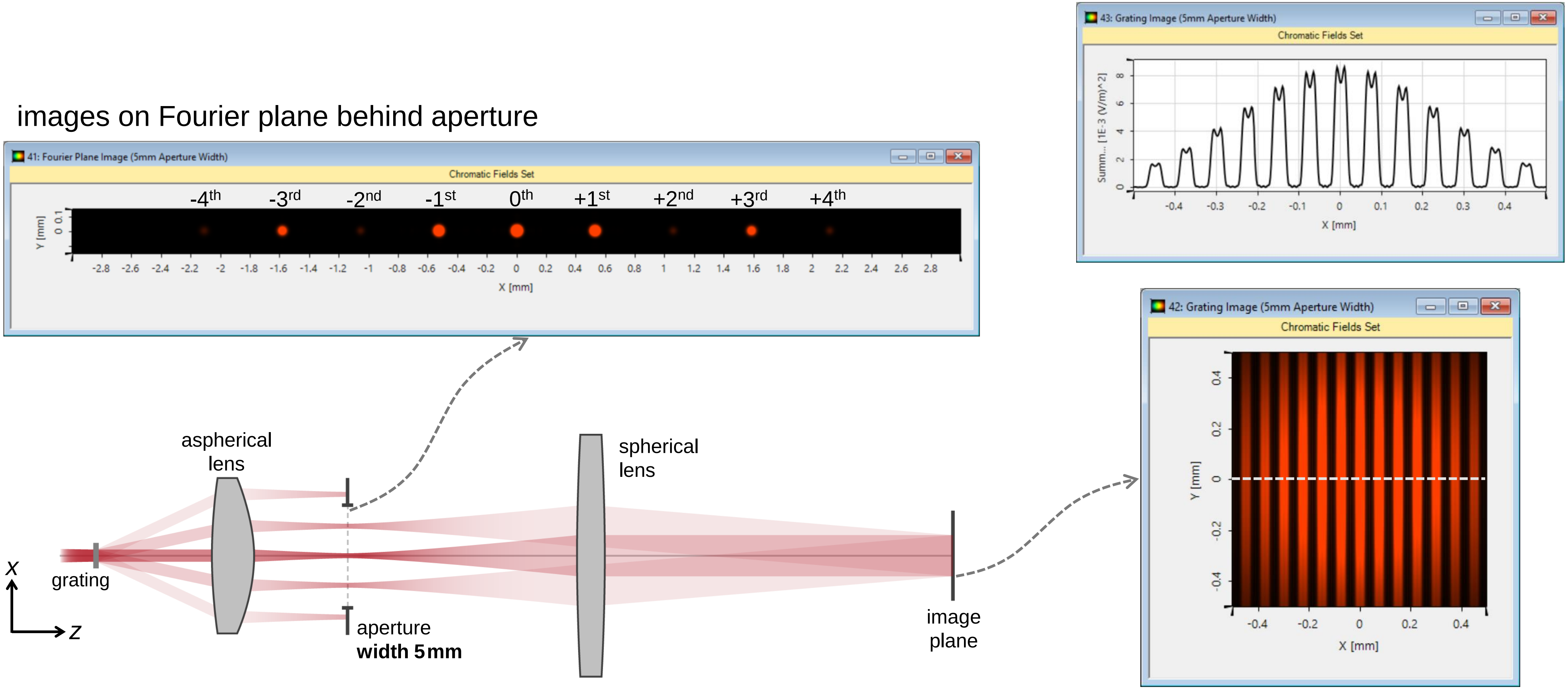
# Modeling Task – Aperture Effect in Fourier Plane

How can the aperture in the Fourier plane affect the image formation, and the resolution of the imaging system?

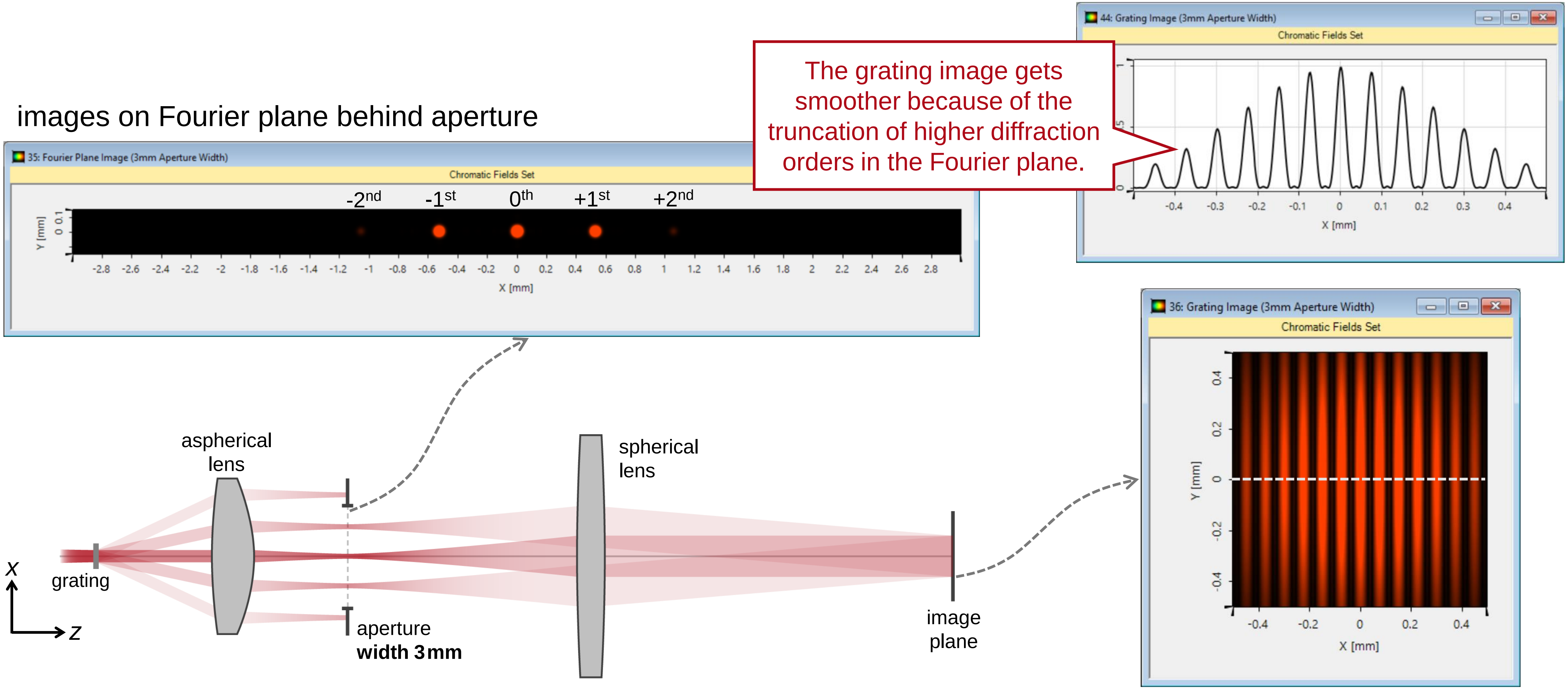


# Aperture Width 5mm

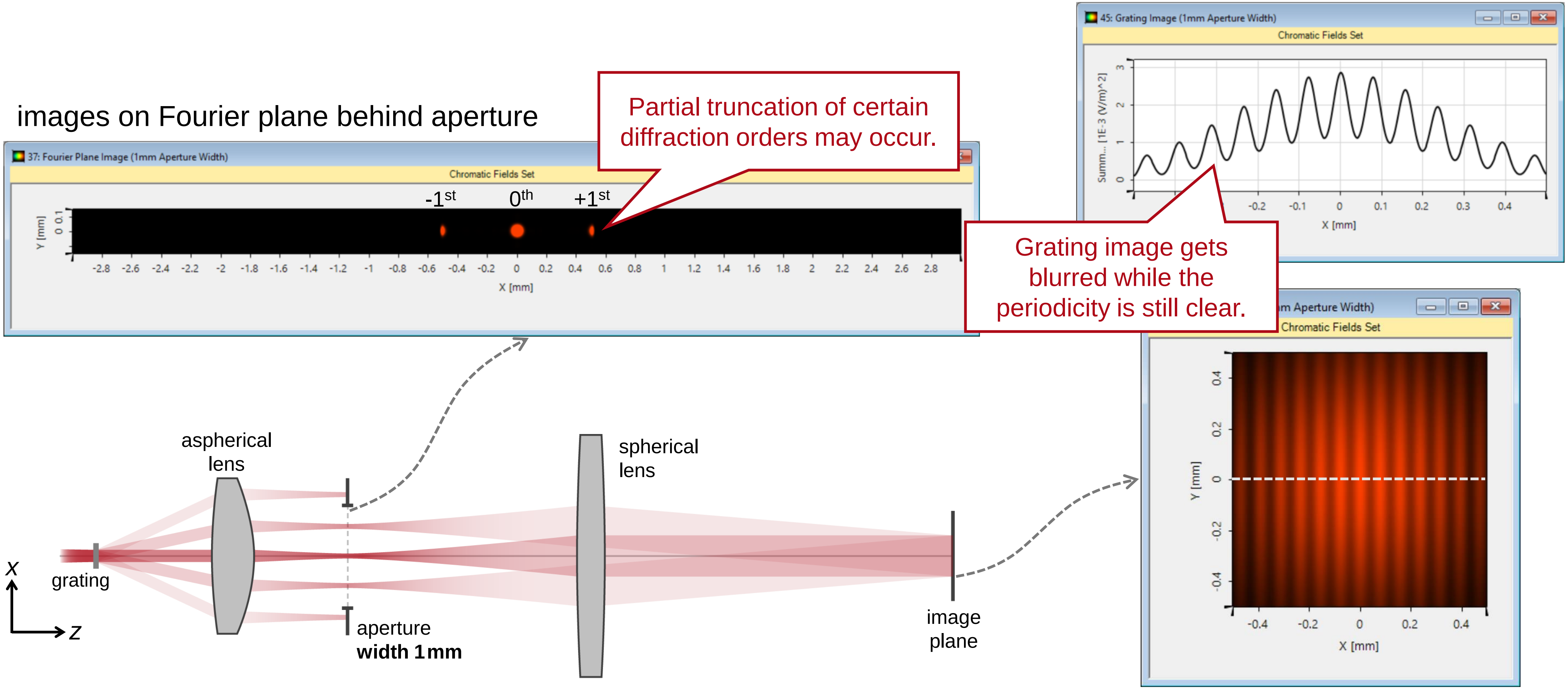
images on Fourier plane behind aperture



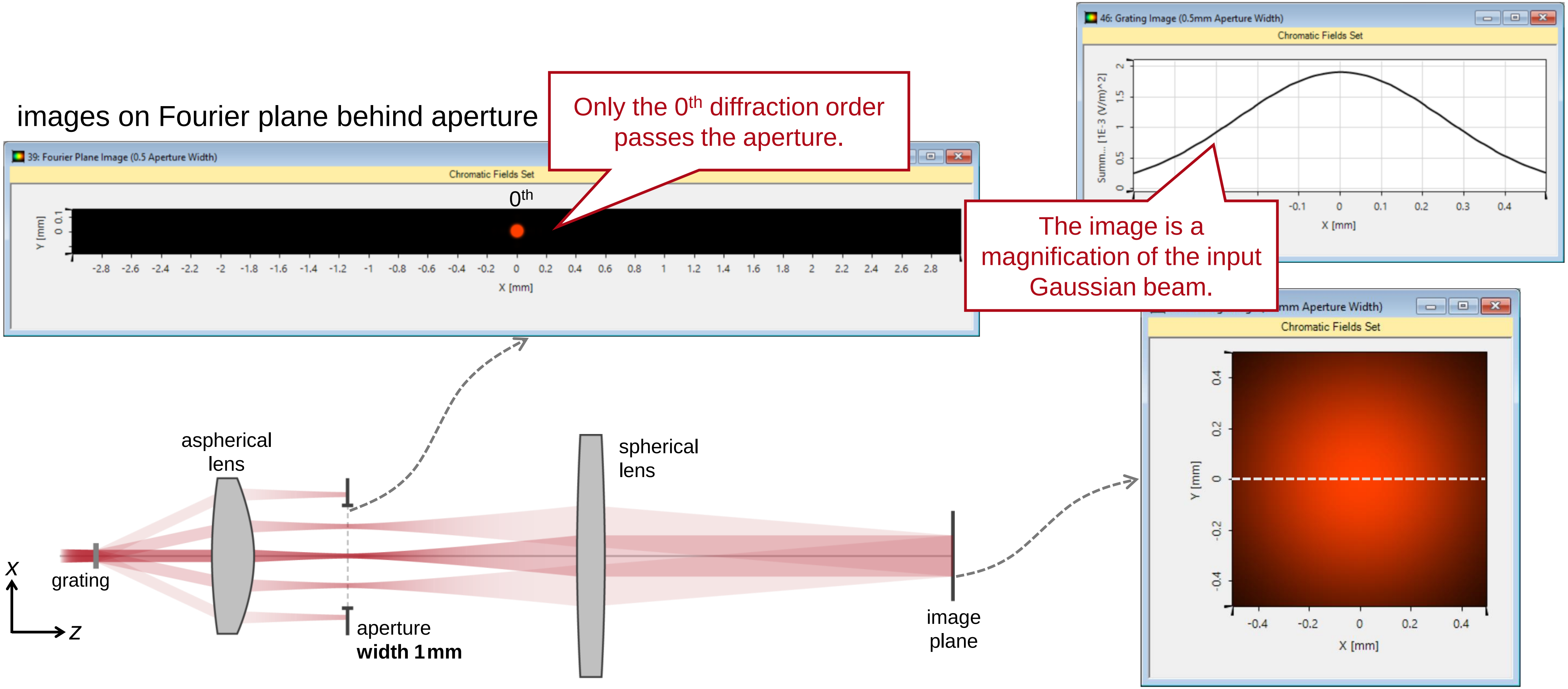
# Aperture Width 3 mm



# Aperture Width 1mm



# Aperture Width 0.5mm

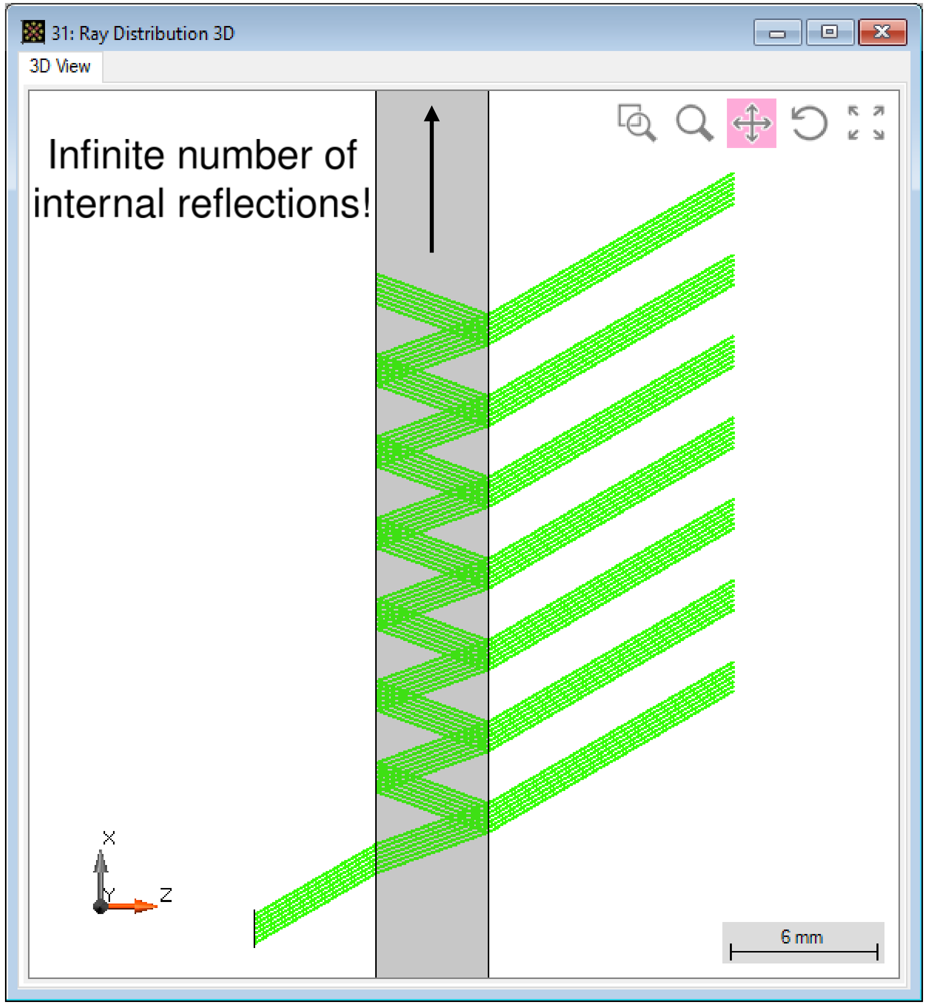


Part 5

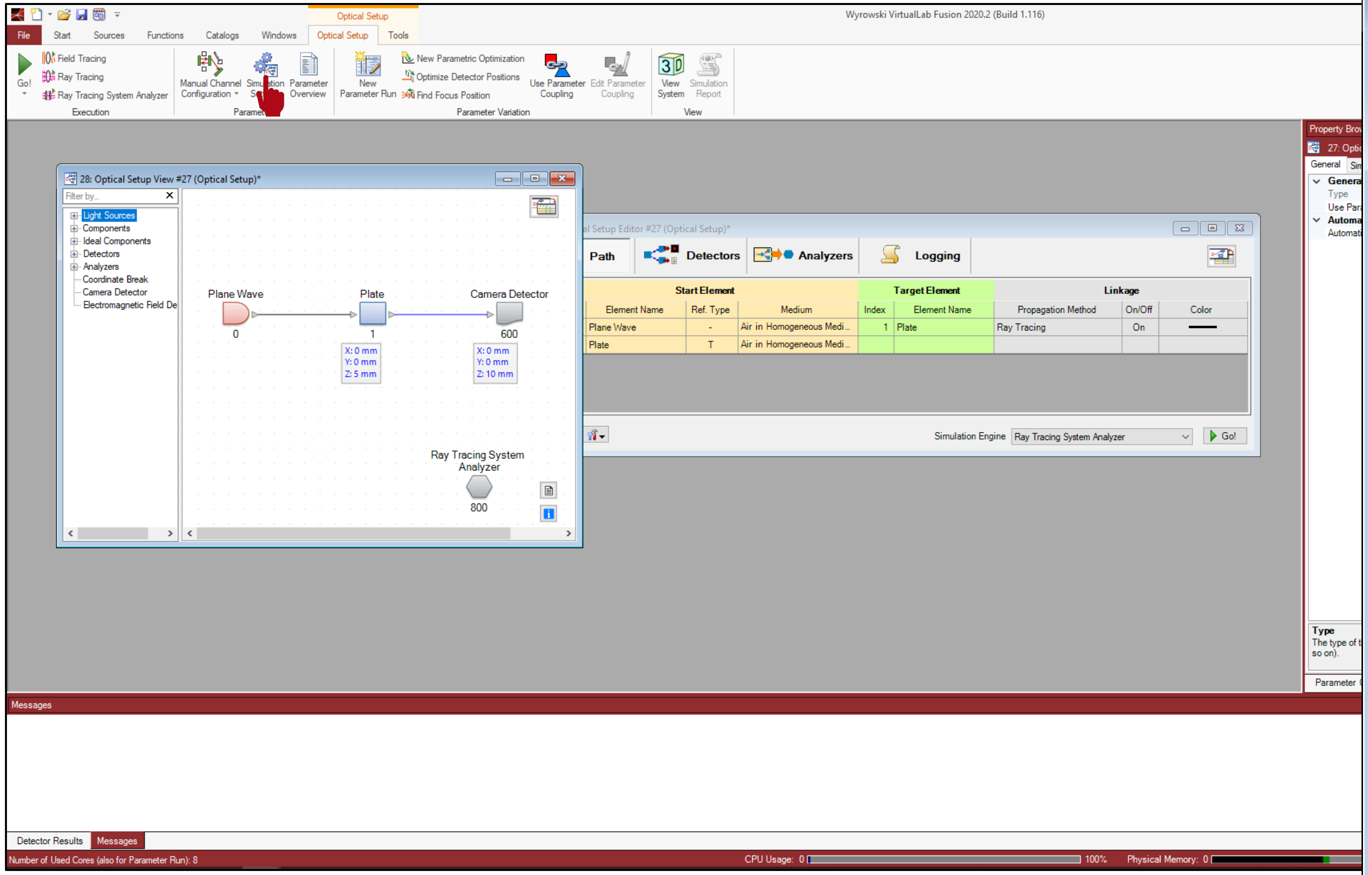
# The Light Path Finder

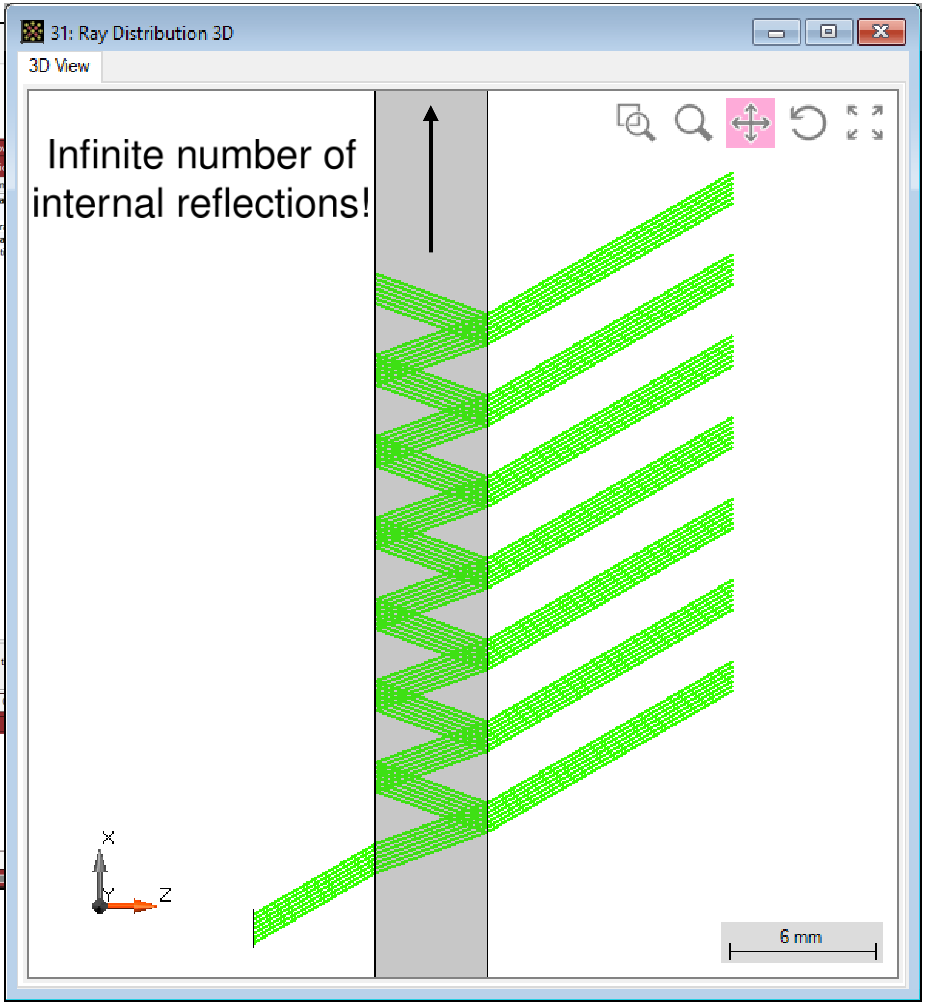
# The Light Path Finder

- The Light Path Finder performs a preliminary analysis of the open channels in the system and determines through which of them light will propagate and how.
- The Light Path Finder uses a physical-optics model without diffraction to analyse the system.
- After the Light Path Finder has determined the light paths that the field is going to follow, the Field Tracing engine takes over and simulates the fully fledged physical-optics propagation of the field along those paths.

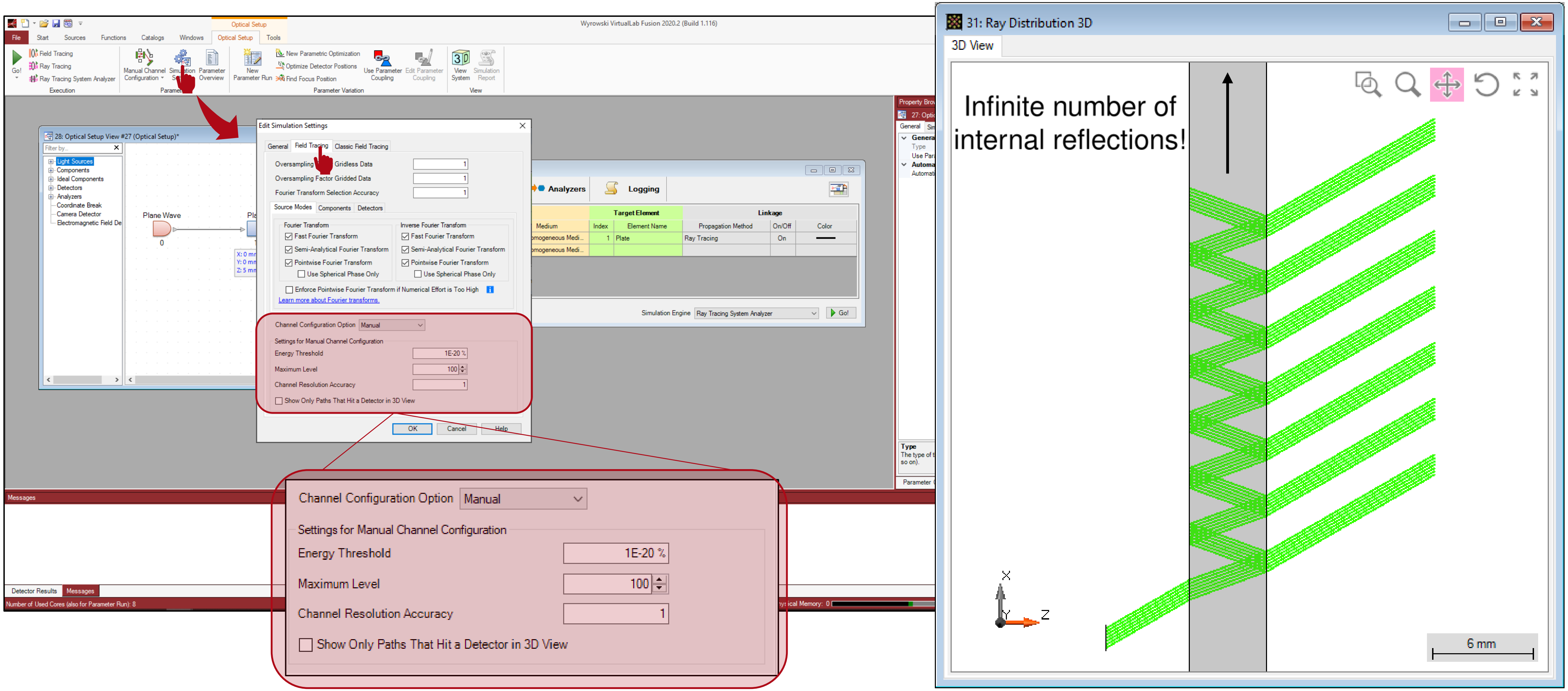


# The Light Path Finder

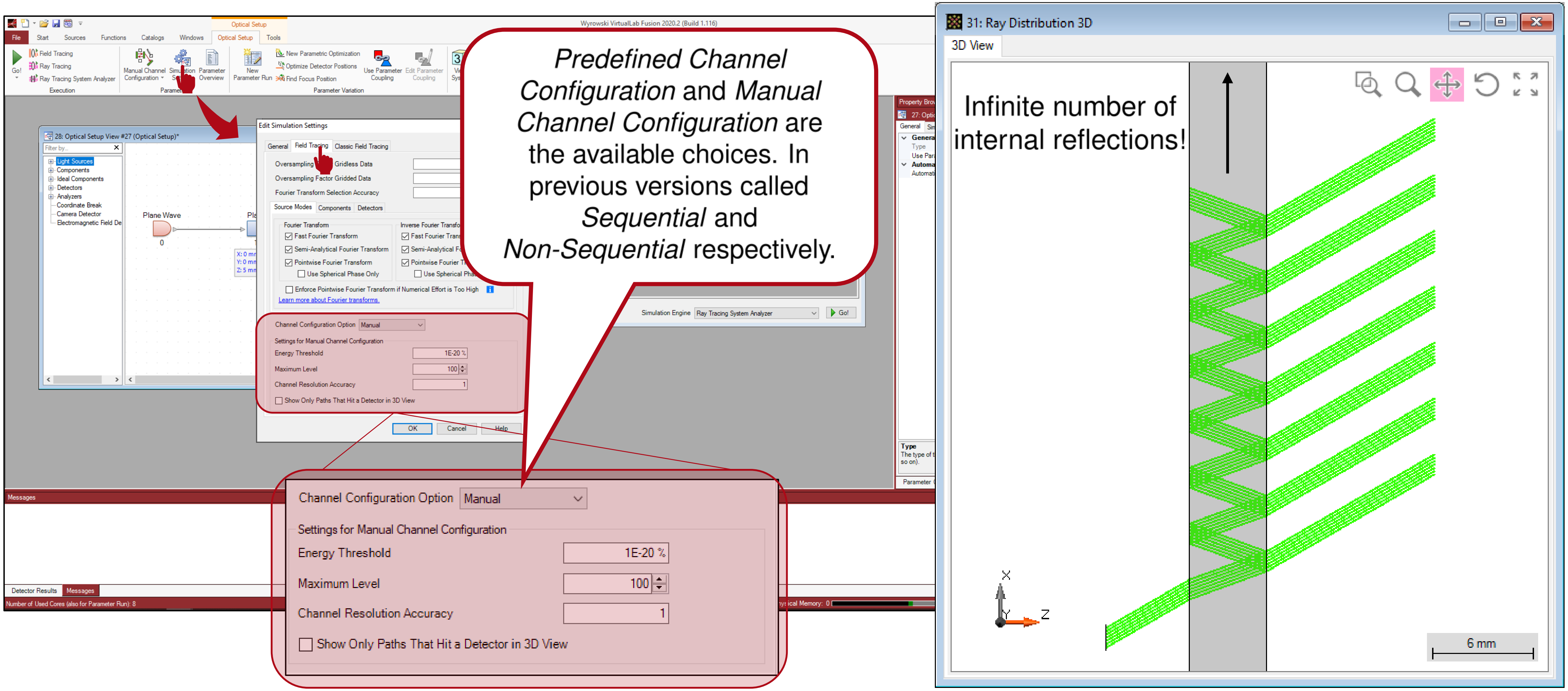




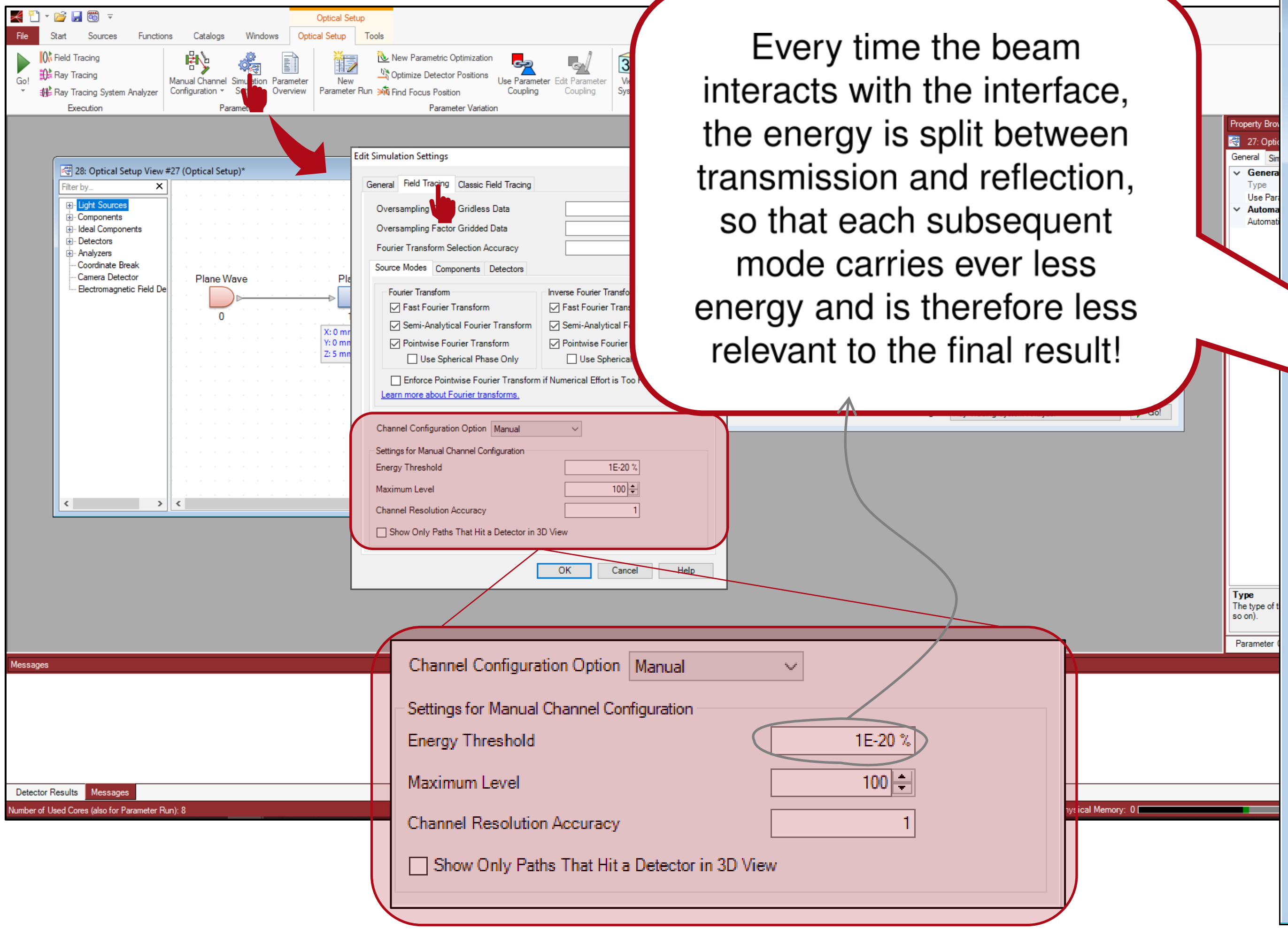
# The Light Path Finder



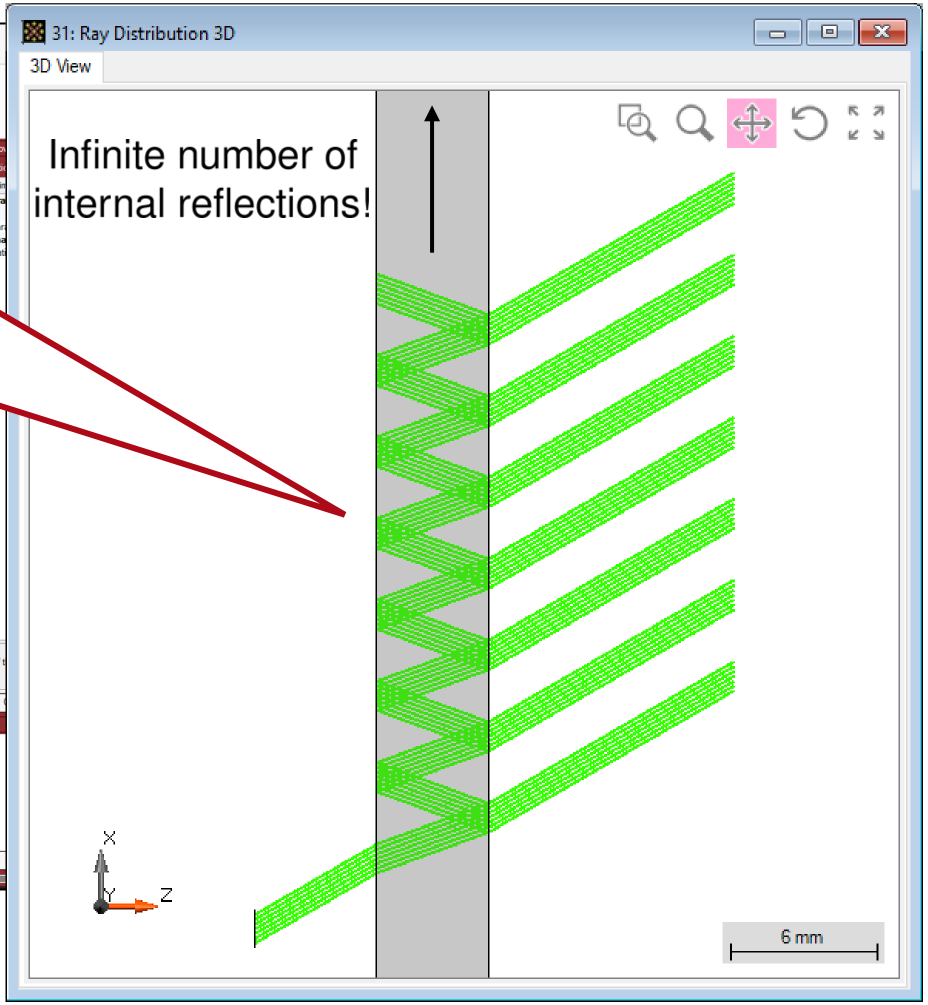
# The Light Path Finder



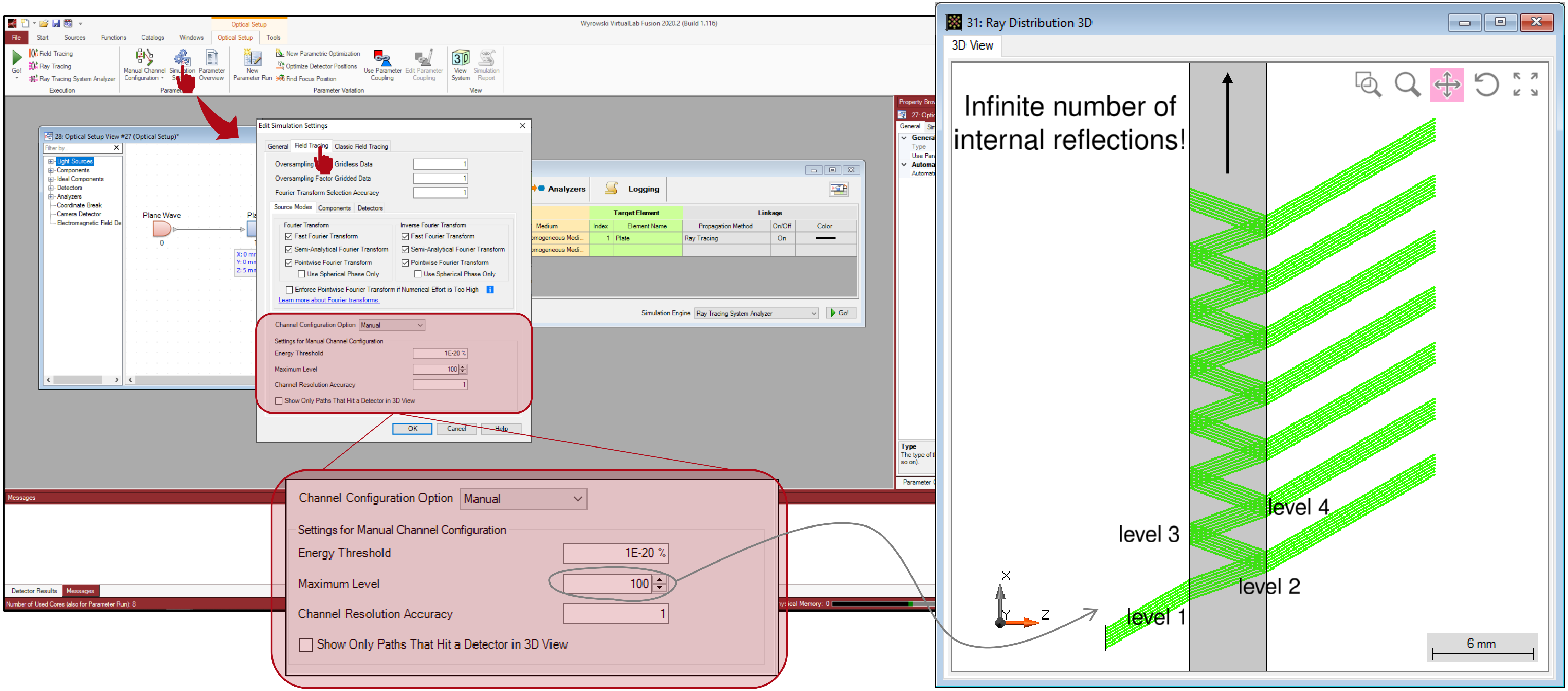
# The Light Path Finder



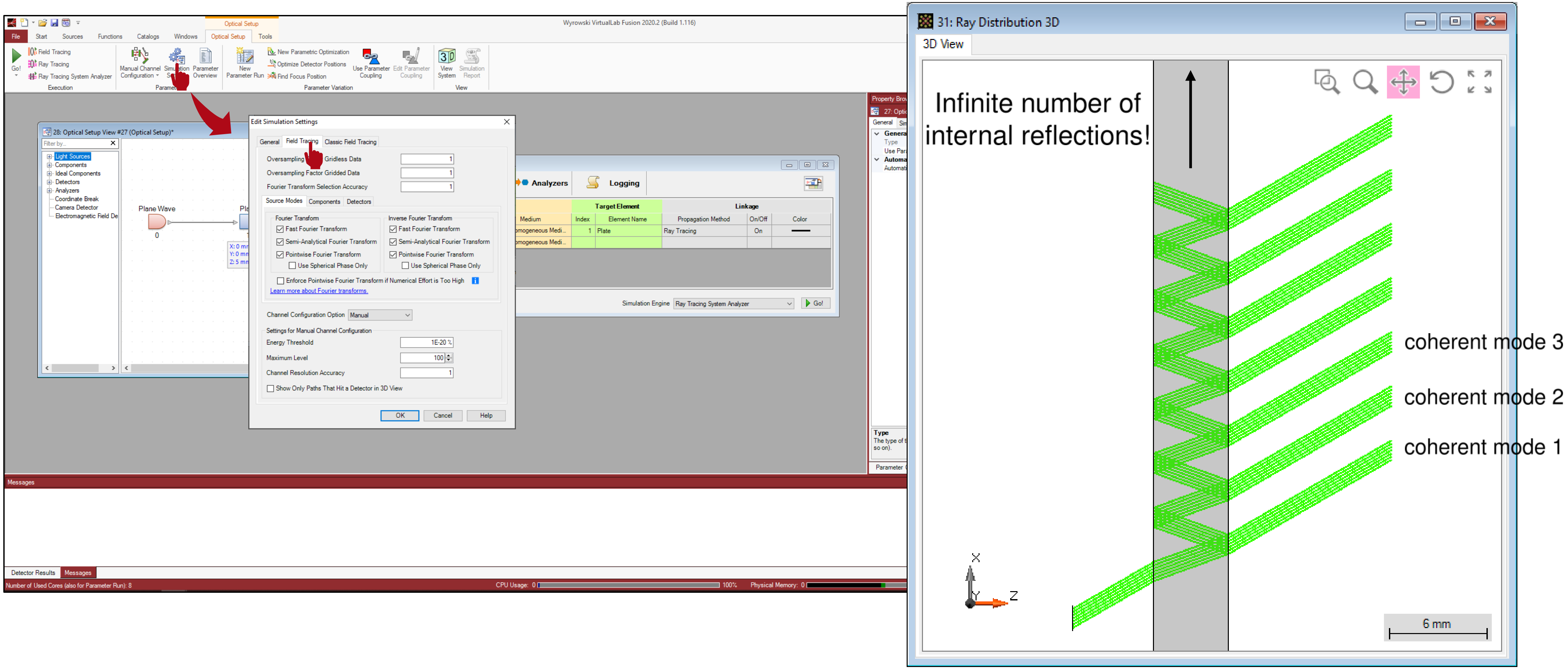

Every time the beam interacts with the interface, the energy is split between transmission and reflection, so that each subsequent mode carries ever less energy and is therefore less relevant to the final result!



# The Light Path Finder



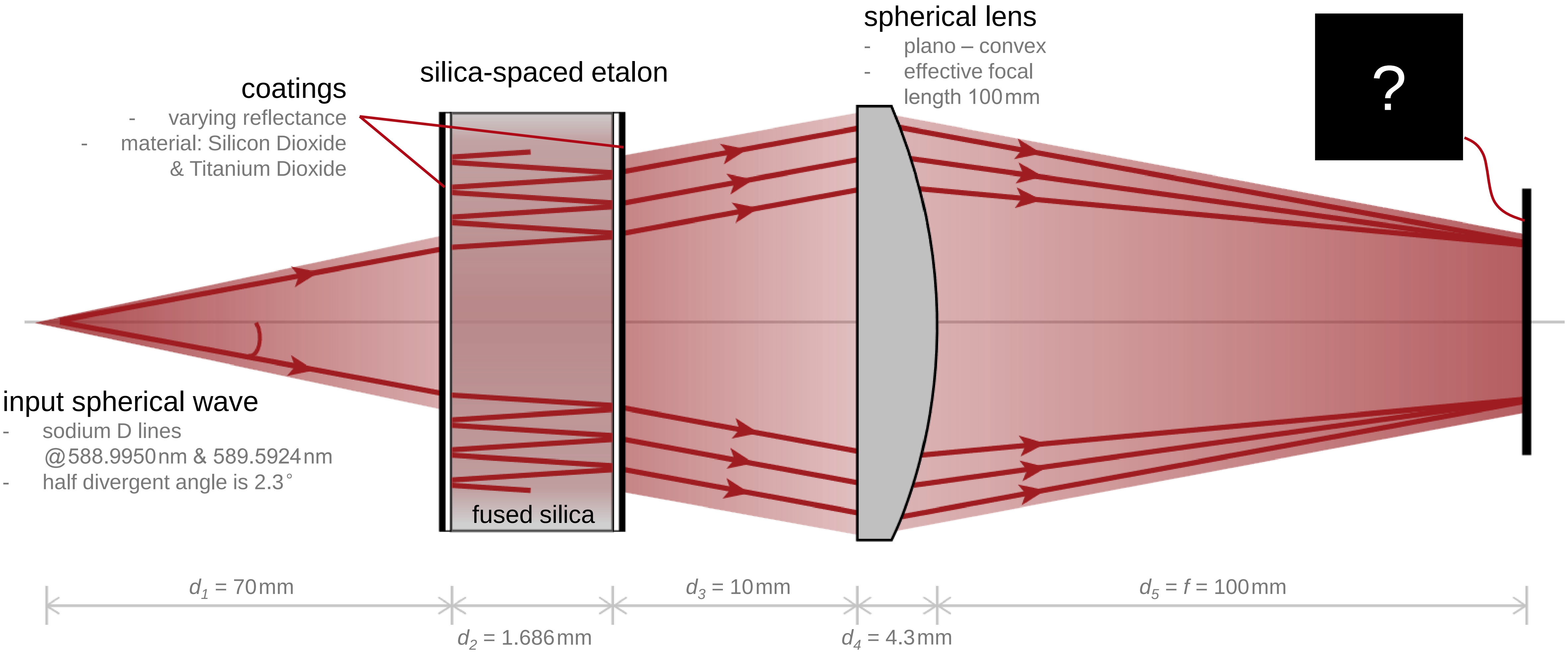
# The Light Path Finder and the Modes Reaching the Detector



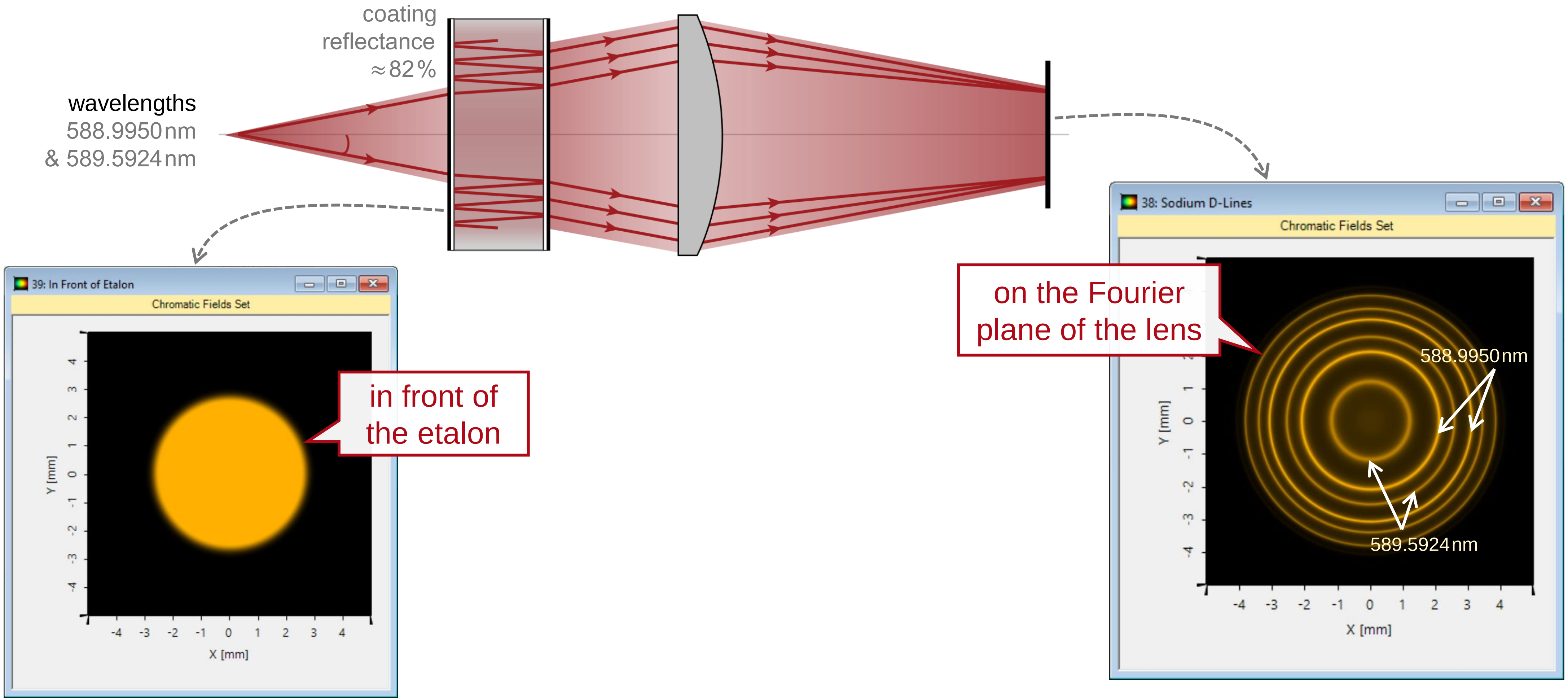
# Examination of Sodium D Lines with Fabry-Pérot Etalon

see the full Application Use Case

# Modeling Task

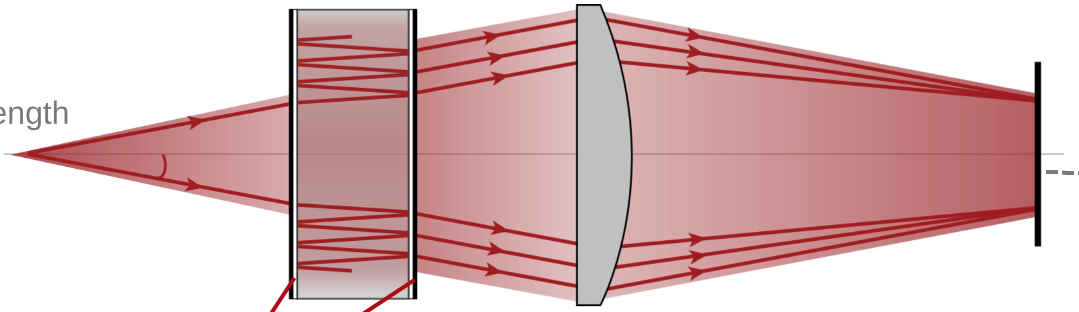


# Visualization of Both Spectrum Lines



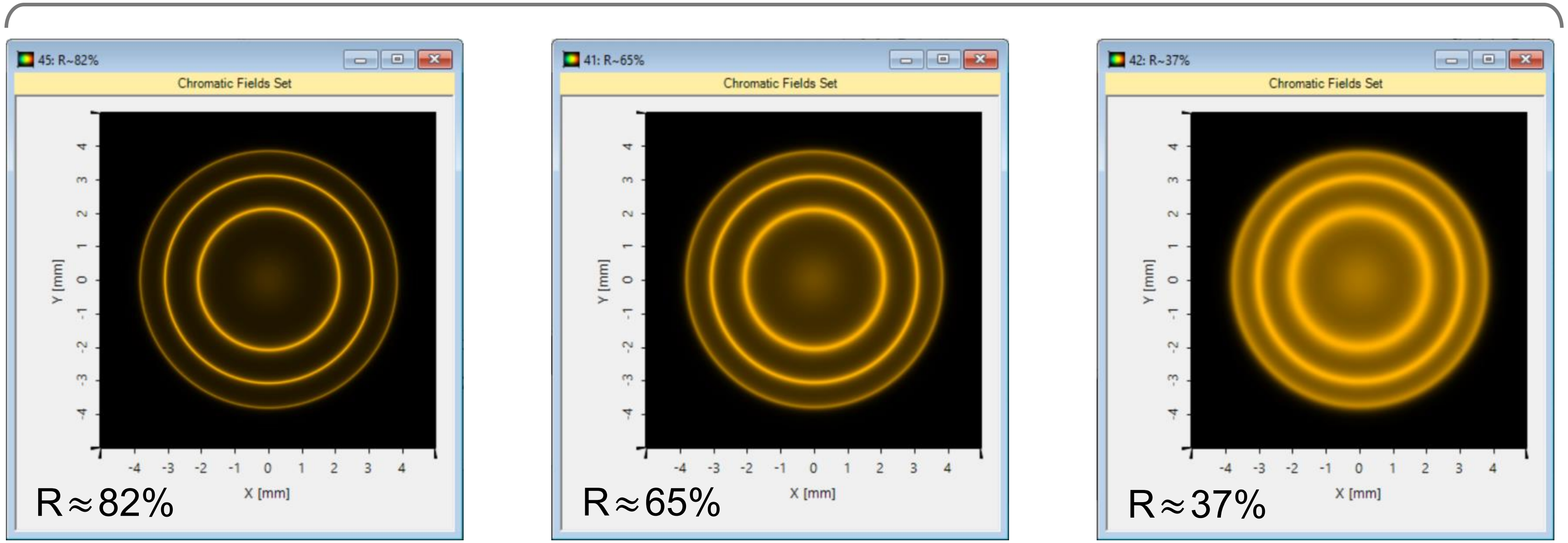
# Finesse vs. Coating Reflectance

single wavelength
589.9950 nm

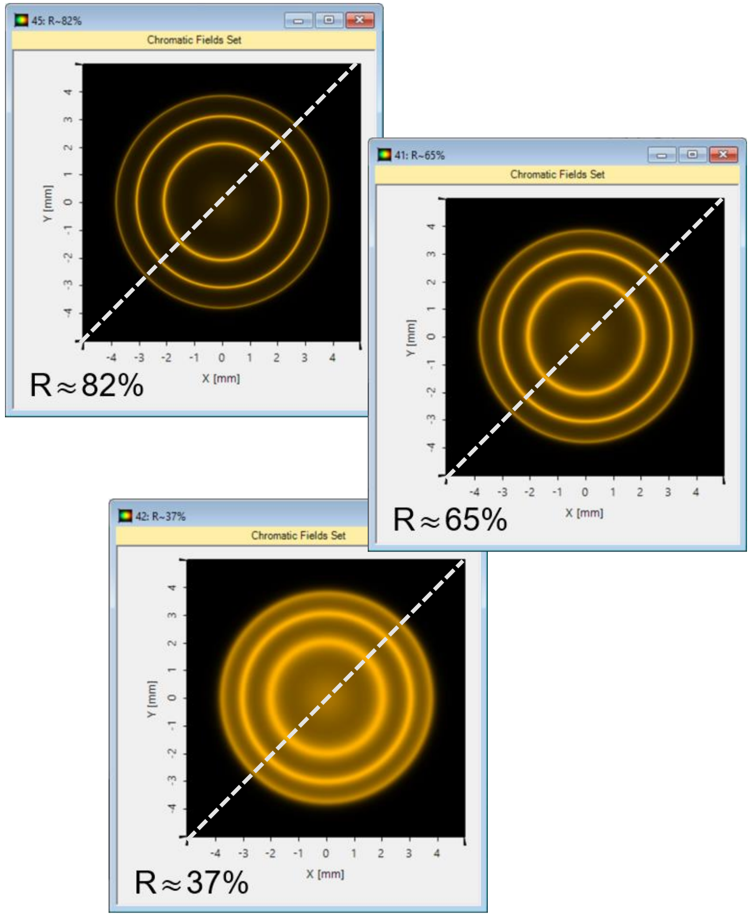


coatings
- varying reflectance: 82%, 65%, 37%

Sharpness of the interference fringes depends on the reflectance of the coatings on the etalon.

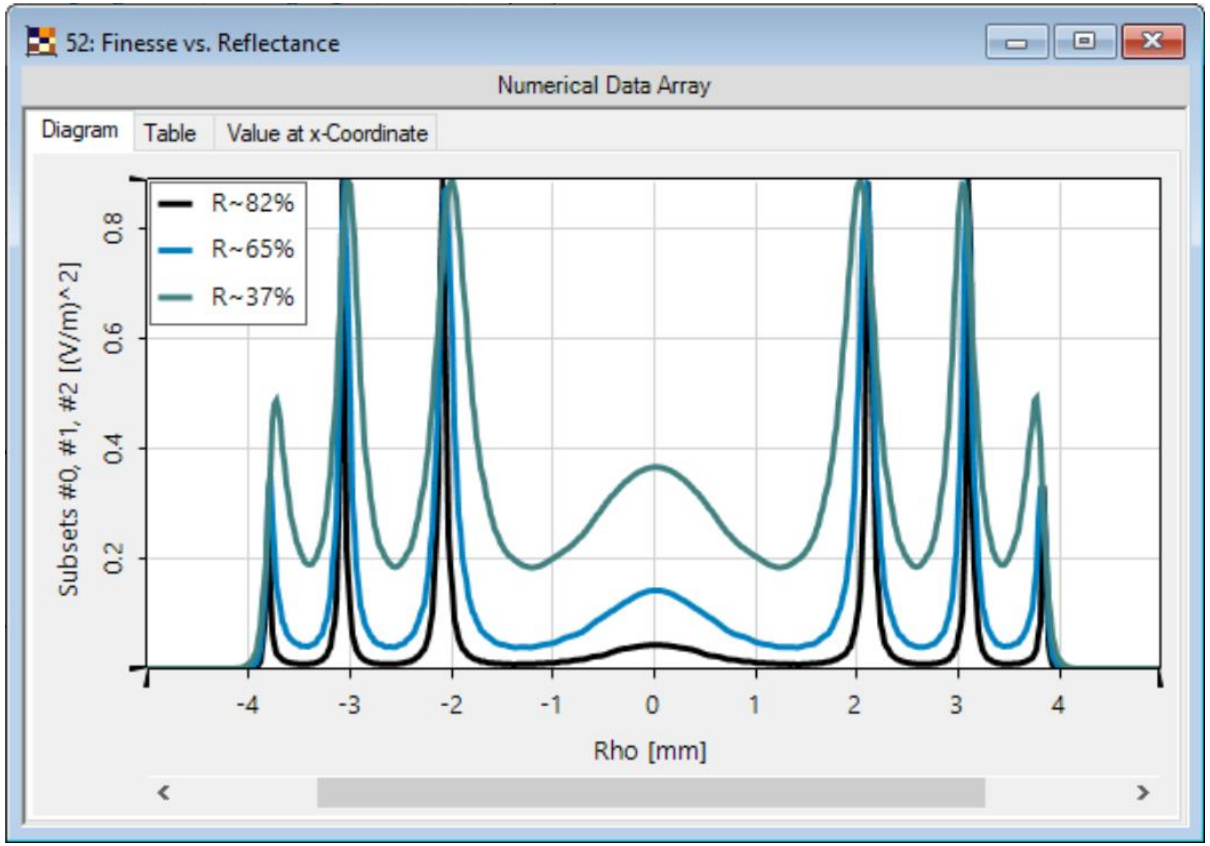


# Finesse vs. Coating Reflectance

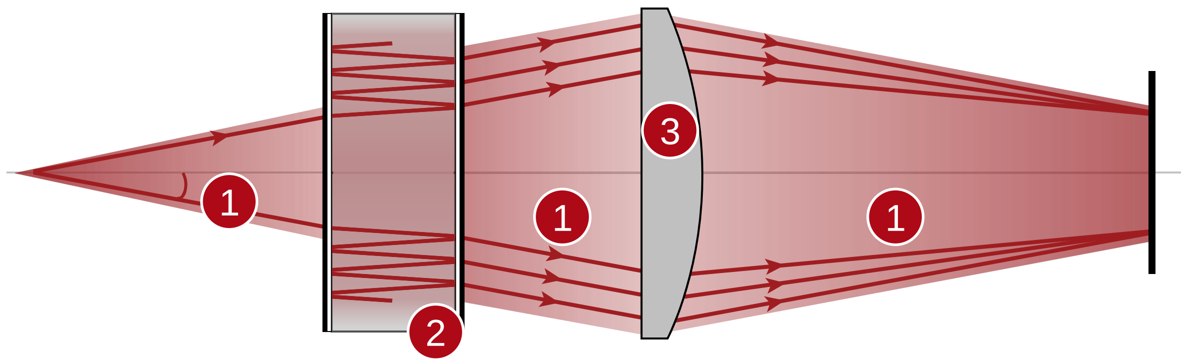


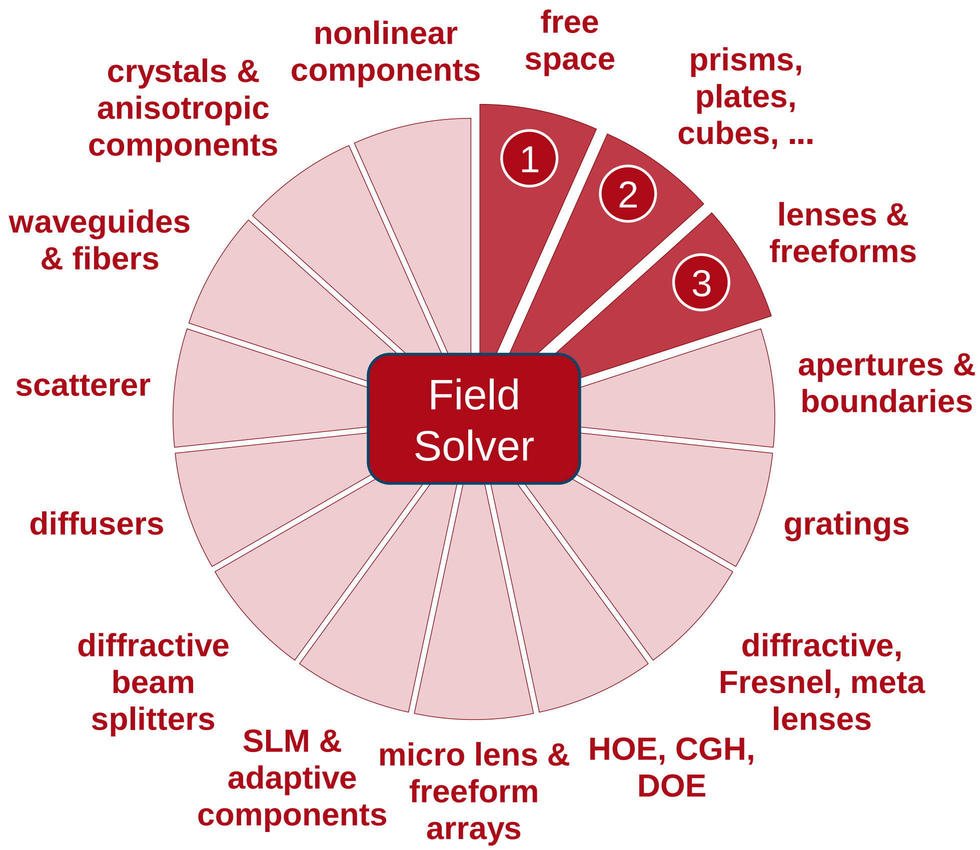

1D measurements along the radial direction

the higher the reflectance, the higher the finesse



# VirtualLab Fusion Technologies



Part 6

# The Source Mode Concept

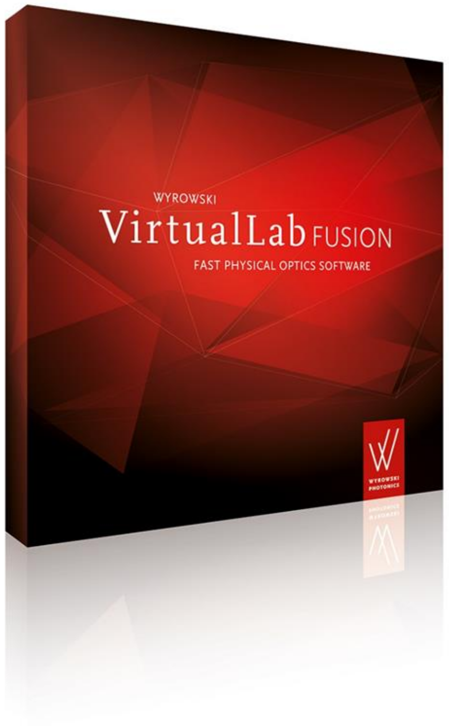
# How to Model the Different Properties of Light?

- Being able to replicate the physical **properties of the light** that enters a system is just as important for an optical simulation as being able to model the effect of the different components on the field.

- We are mainly talking about **polarization**, partial **temporal coherence**, partial **spatial coherence**.

- There are different mathematical strategies available.

- In VirtualLab Fusion we follow a single general approach which can be applied to the simulation of all these characteristics: the **source-mode concept**.

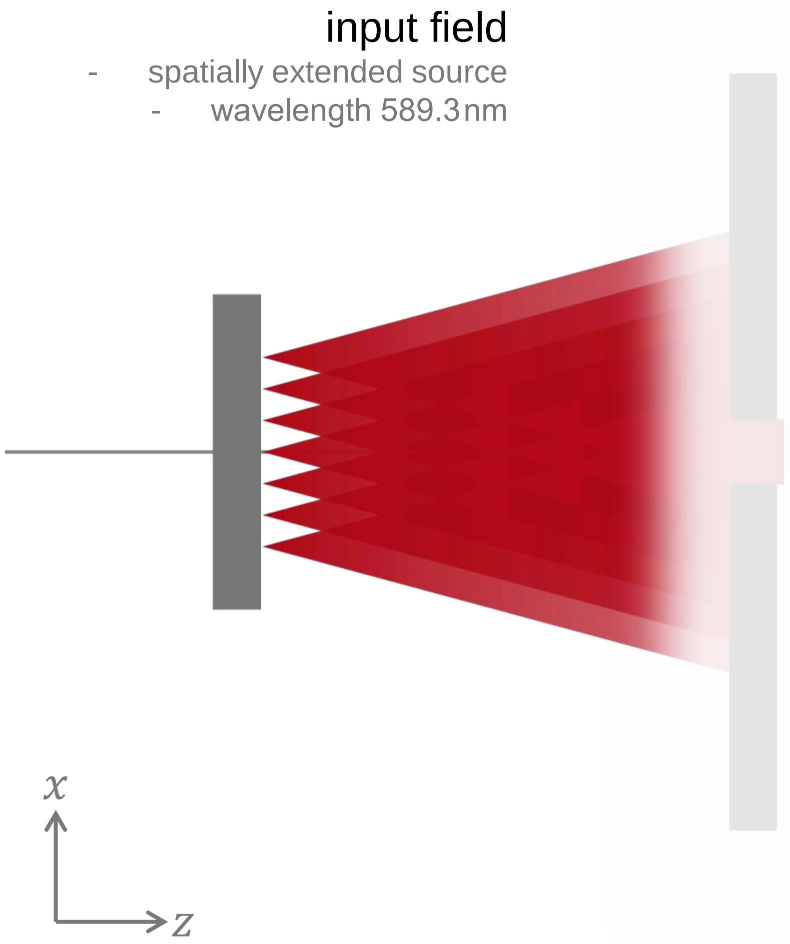


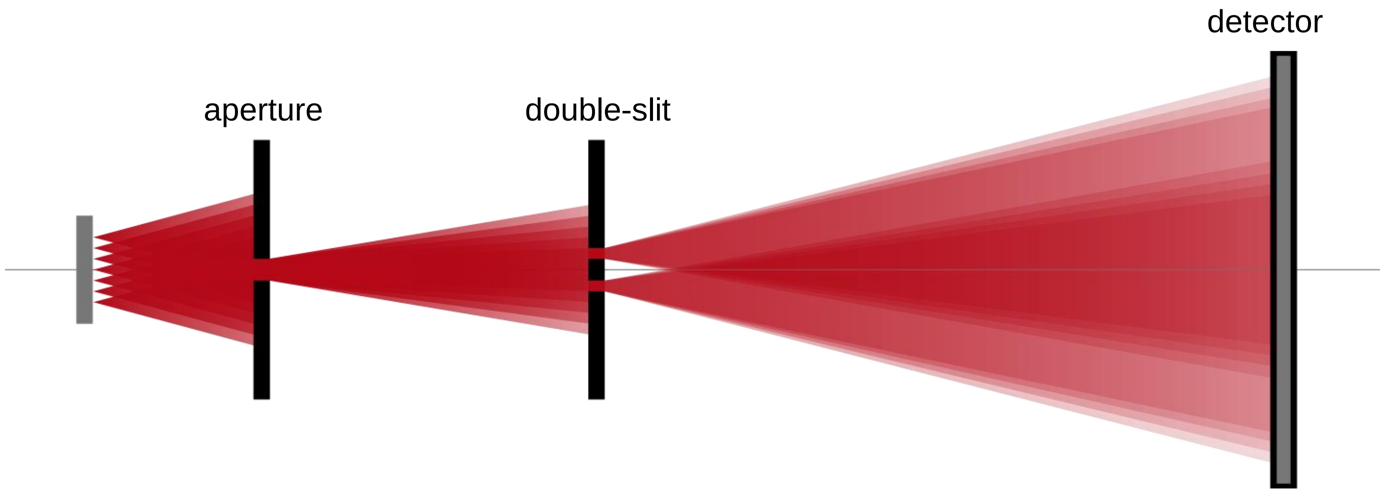
# The Source Mode Concept

- The light at the input plane of the system is represented by a finite set of modes.
- **Each of the modes is fully self-correlated.** This means that, if a source mode is split on its way through the system (for instance, by being partly transmitted and partly reflected at an interface) each of the resulting modes will be coherent to each other, and therefore capable of generating an interference pattern.
- **Each of the source modes can be coherent or incoherent to each of the other source modes,** depending on the property to be modelled and the characteristics of the source.
- There are different types of modes: spectral (wavelength) and spatial.
- **Finding the best set of modes for a given source in a given system is in general not a trivial task!**

# Partial Spatial Coherence with Shifted Elementary Field Method

# Shifted Elementary-Field Method

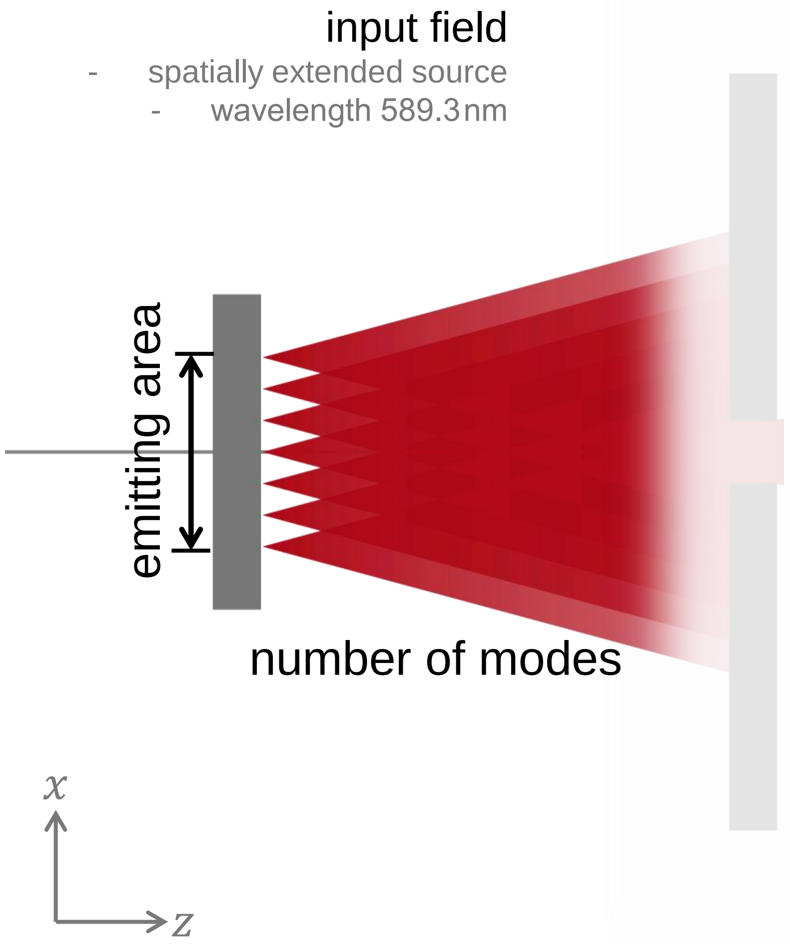



The source is represented by a superposition of spatially identical, but mutually uncorrelated and transversely displaced, fully coherent fields [J. Opt. Soc. Am. A 27 (9), 2010].

This means that each elementary field (mode) is fully coherent with itself, but fully incoherent with all the other elementary fields in the source. In this case the elementary fields are spherical fields each generated by a different point source.

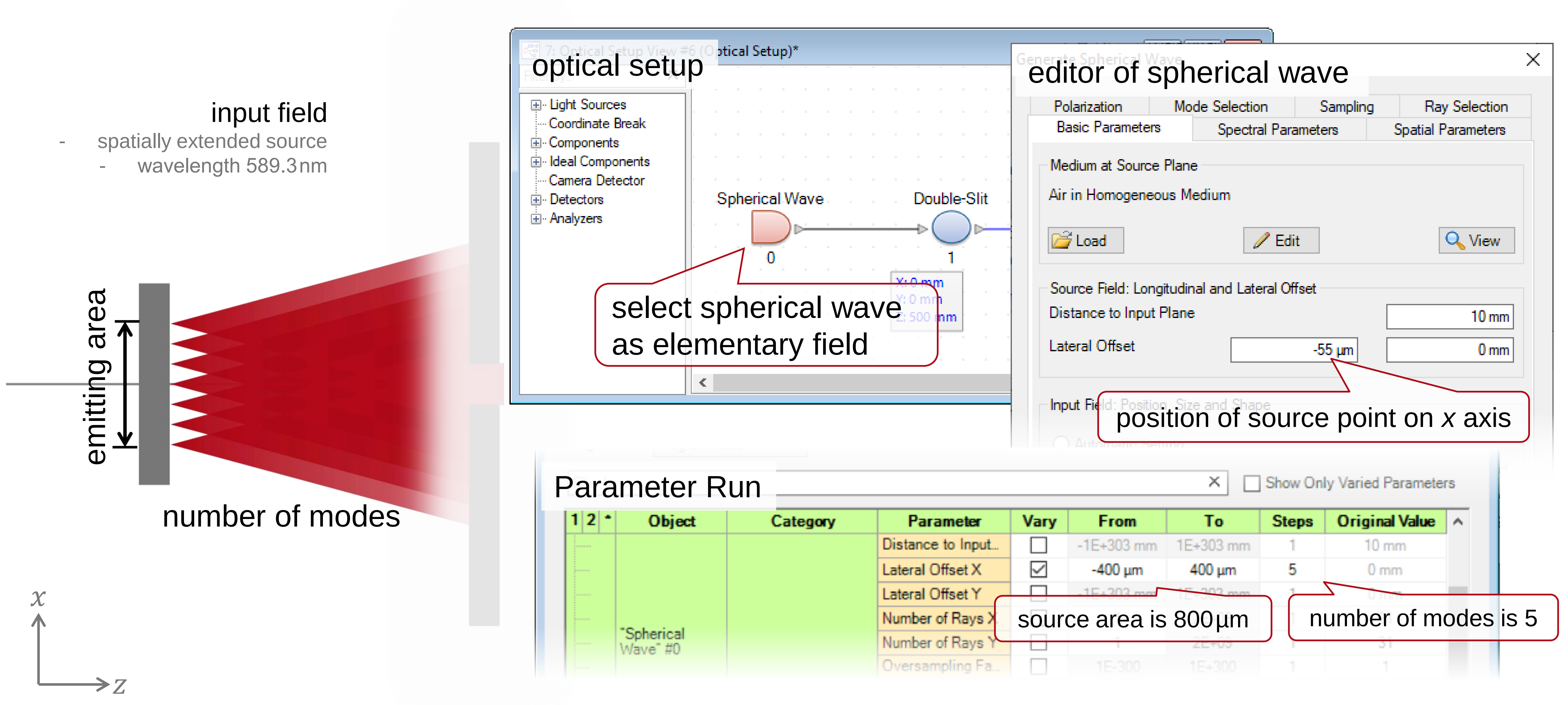
# Number of Elementary-Fields (Modes)



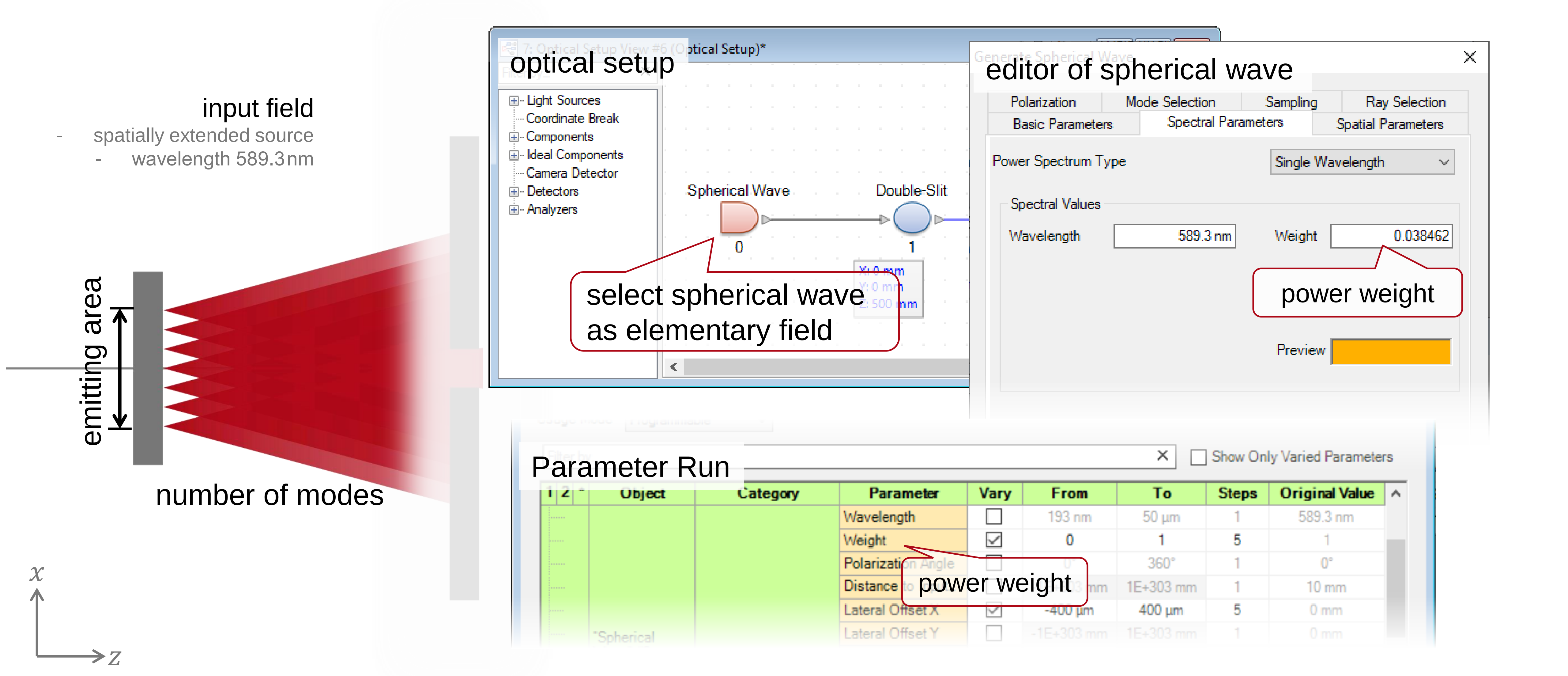

- The emitting area is fixed at 800 µm. The point source at the edge of this emitting area gives a weak interference pattern, which is negligible.
- The number of elementary fields (modes) should be large enough to achieve convergent and reliable results.
- So before performing the simulation, **we use a one-dimensional (1D) list of point sources along the x axis to check how many modes give convergent fringes along said axis in the detector**.
- To keep the power of the source constant, the power weight of each point source decreases as the number of modes increases.

$$\text{weight} = \frac{1.0}{\text{number of modes}}$$

# Configuration of Parameter Variation



# Configuration of Parameter Variation

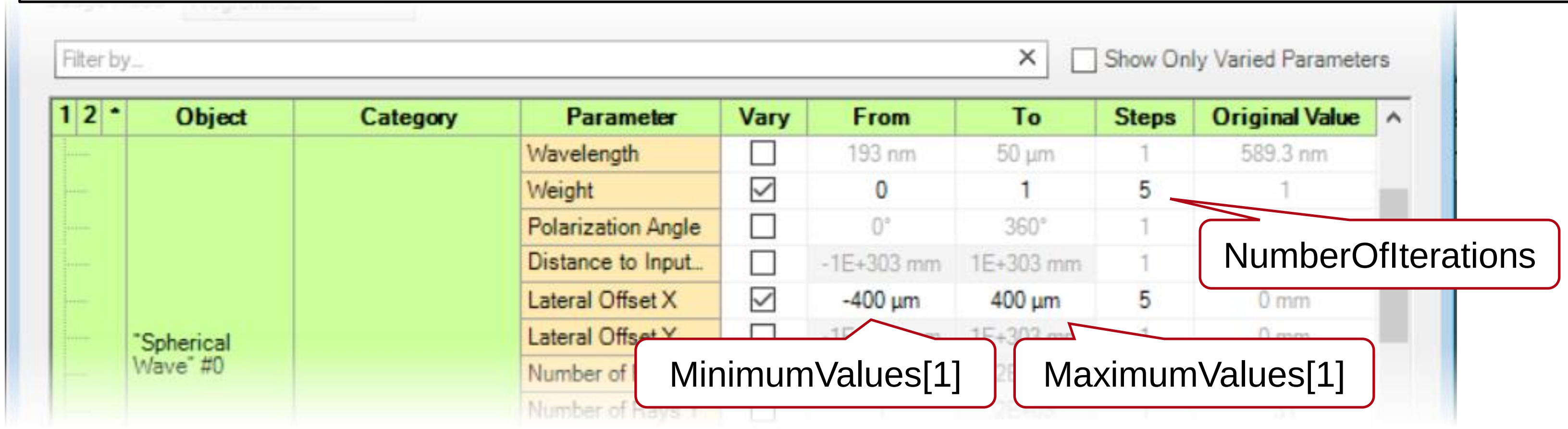


# Programmable Mode of Parameter Run

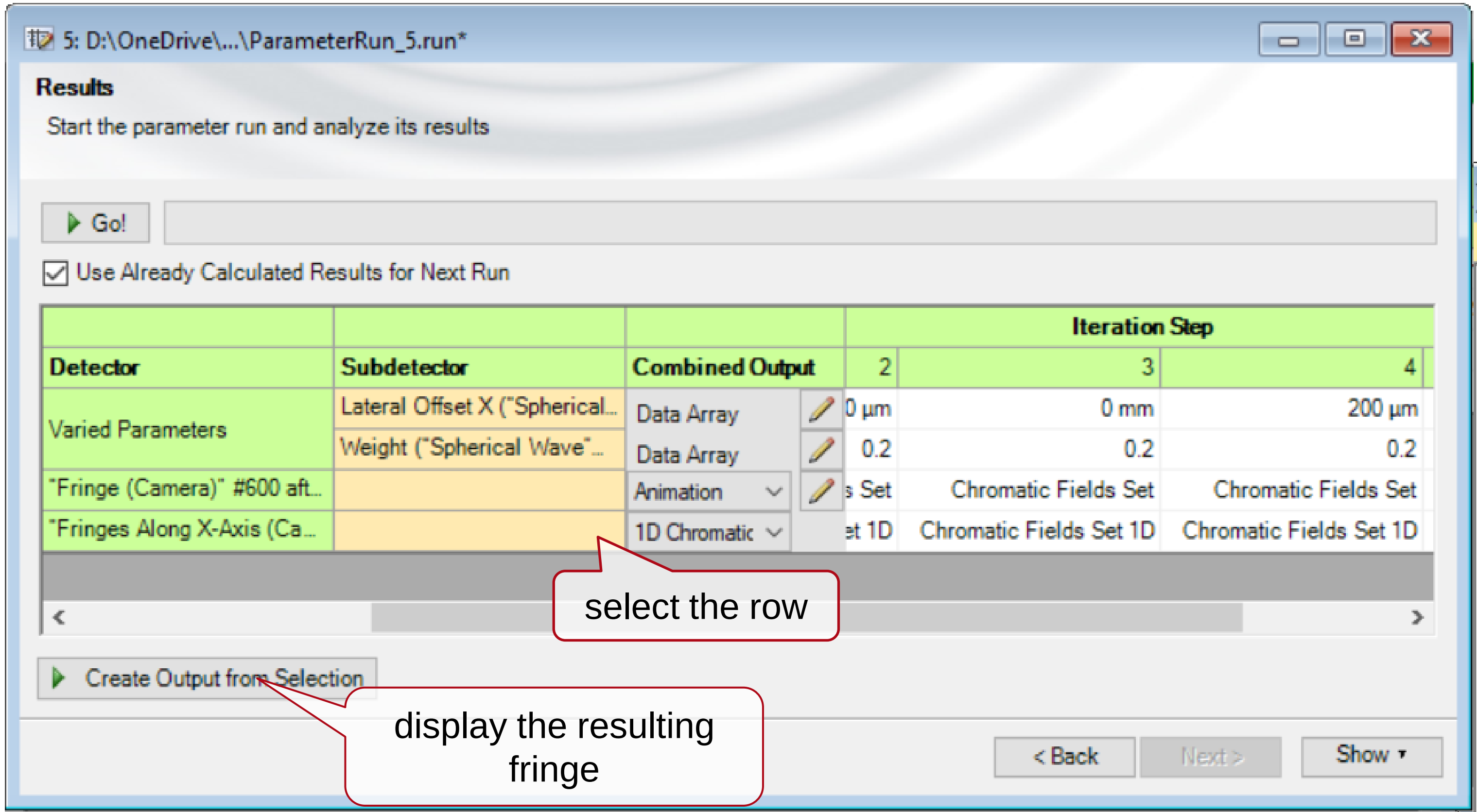
**Main Function**

```
double[,] parameters = new double[NumberOfParameters,NumberOfIterations];


double weight = 1.0 / NumberOfIterations;
for(int Index = 0;Index < NumberOfIterations;Index++)
{
    //power weight
    parameters[0, Index] = weight;
    //position of point source
    parameters[1, Index] = MinimumValues[1] + (MaximumValues[1] - MinimumValues[1]) / (NumberOfIterations - 1.0) * Index;
}

return parameters;
```

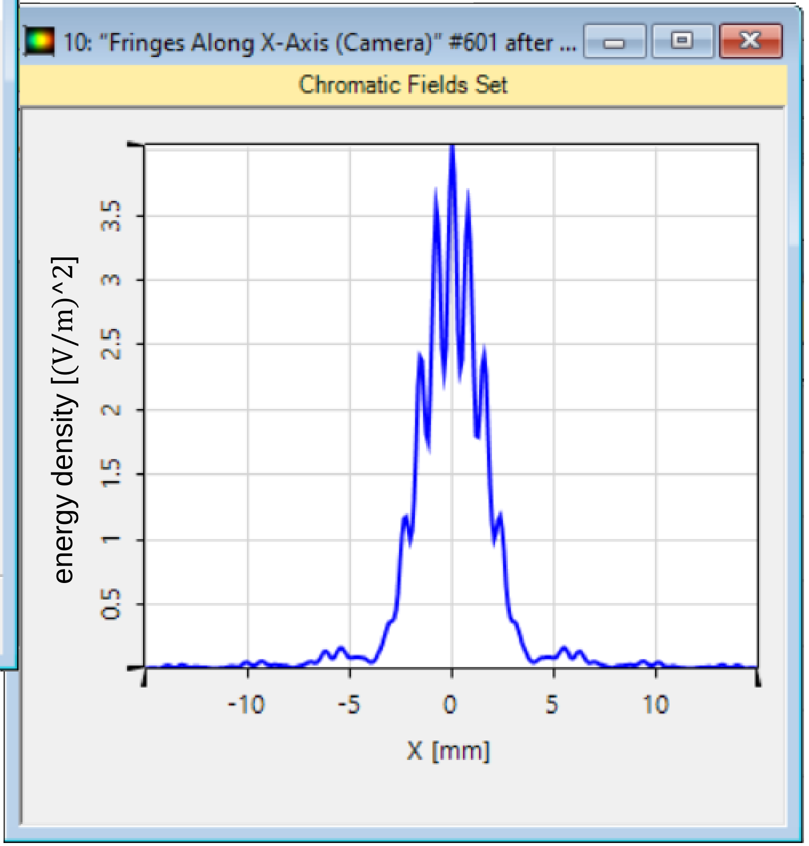
$$\text{weight} = \frac{1.0}{\text{number of modes}}$$



# Display of Resulting Fringe along *x* Axis

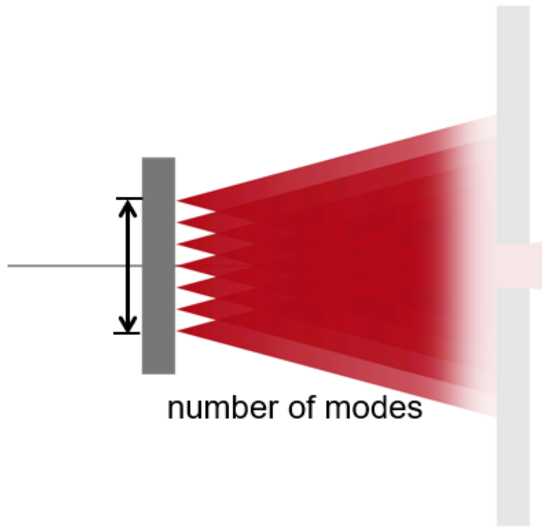


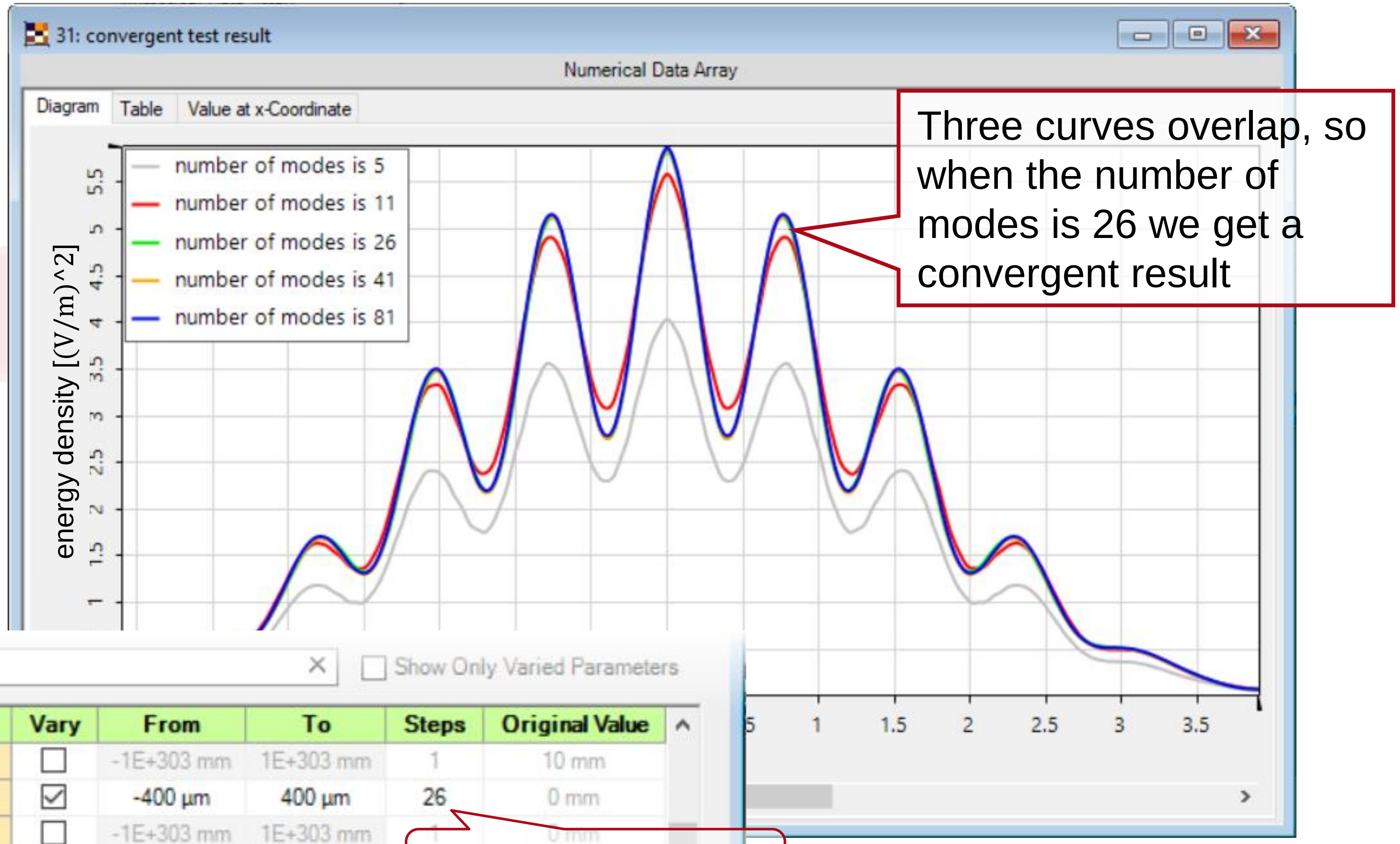

fringe along *x* axis



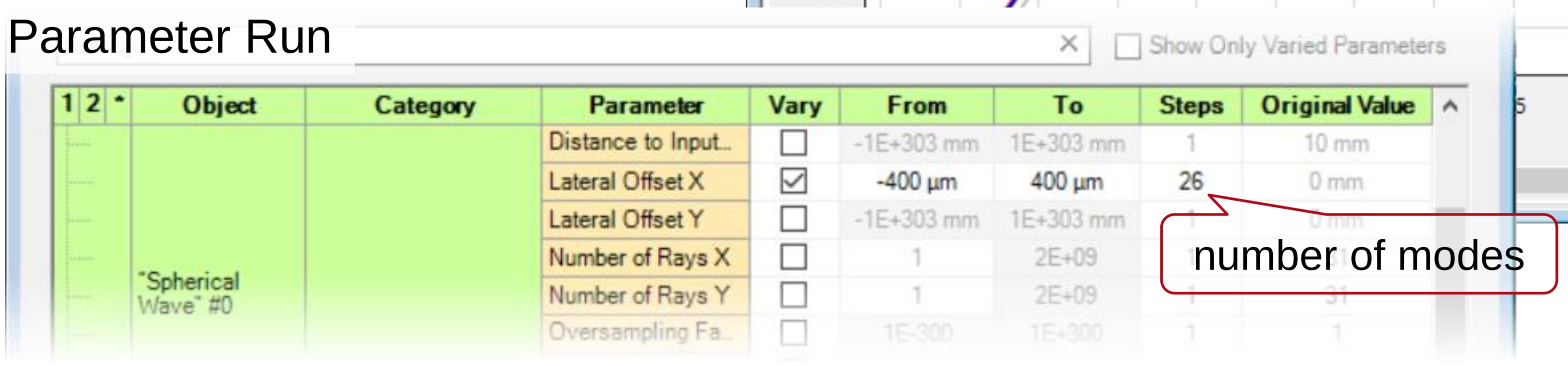

superposition of energy density
from different point sources

# Fringes with Different Number of Modes



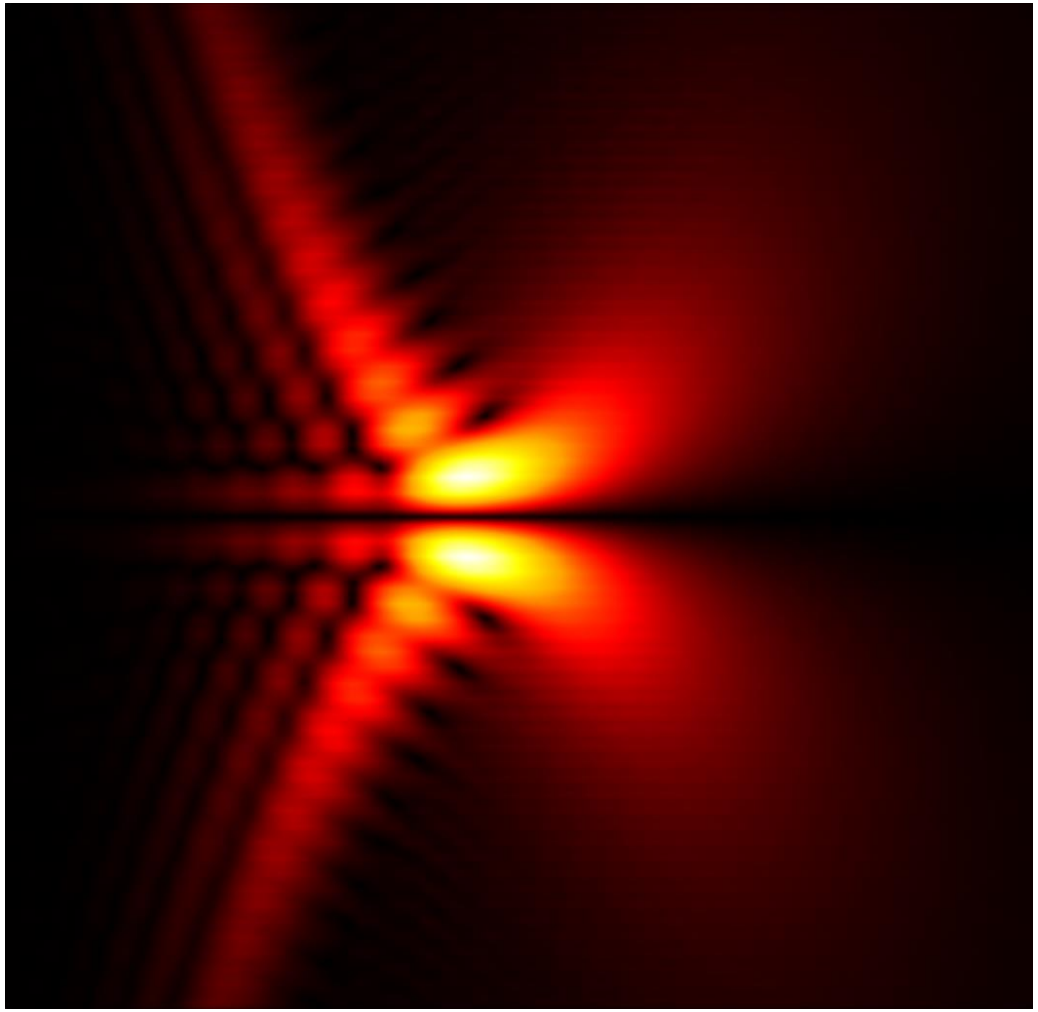

Parameter Run




| 1 | 2 | * | Object | Category | Parameter | Vary | From | To | Steps | Original Value |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | "Spherical Wave" #0 | | Distance to Input... | ☐ | -1E+303 mm | 1E+303 mm | 1 | 10 mm |
| | | | | | Lateral Offset X | ☑ | -400 µm | 400 µm | 26 | 0 mm |
| | | | | | Lateral Offset Y | ☐ | -1E+303 mm | 1E+303 mm | 1 | 0 mm |
| | | | | | Number of Rays X | ☐ | 1 | 2E+09 | | |
| | | | | | Number of Rays Y | ☐ | 1 | 2E+09 | | 31 |
| | | | | | Oversampling Fa... | ☐ | 1E-300 | 1E+300 | 1 | 1 |

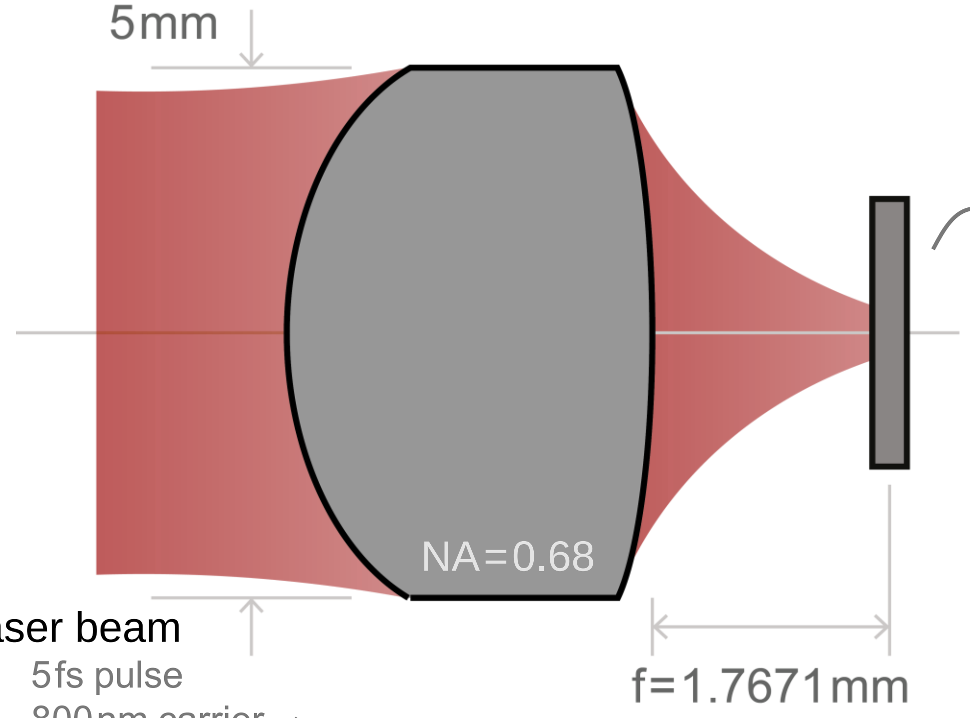
# Pulse Focusing with High-NA Lens

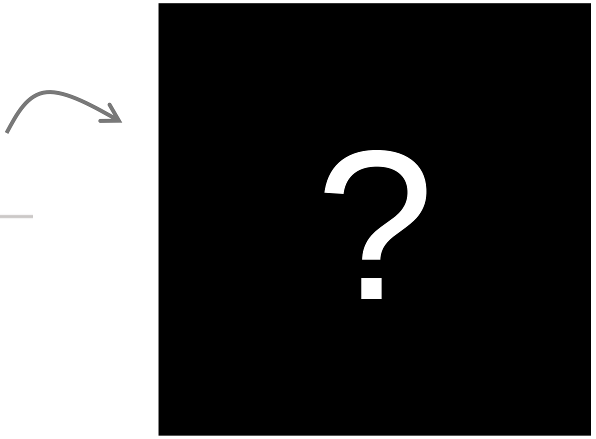
# Abstract



While for most other types of sources it is often accurate enough to labour under the stationary approximation, ultrashort pulses require a somewhat more nuanced approach, where the correlation between the different spectral modes is taken into account. We investigate here the effects of subjecting one such pulse to propagation through a lens with high numerical aperture, in terms of its spatial, as well as of its temporal, profile.

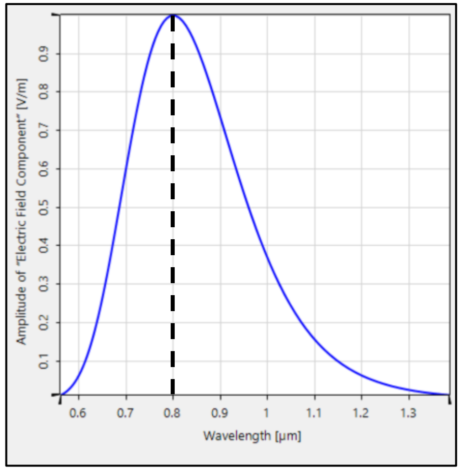
# Modeling Task



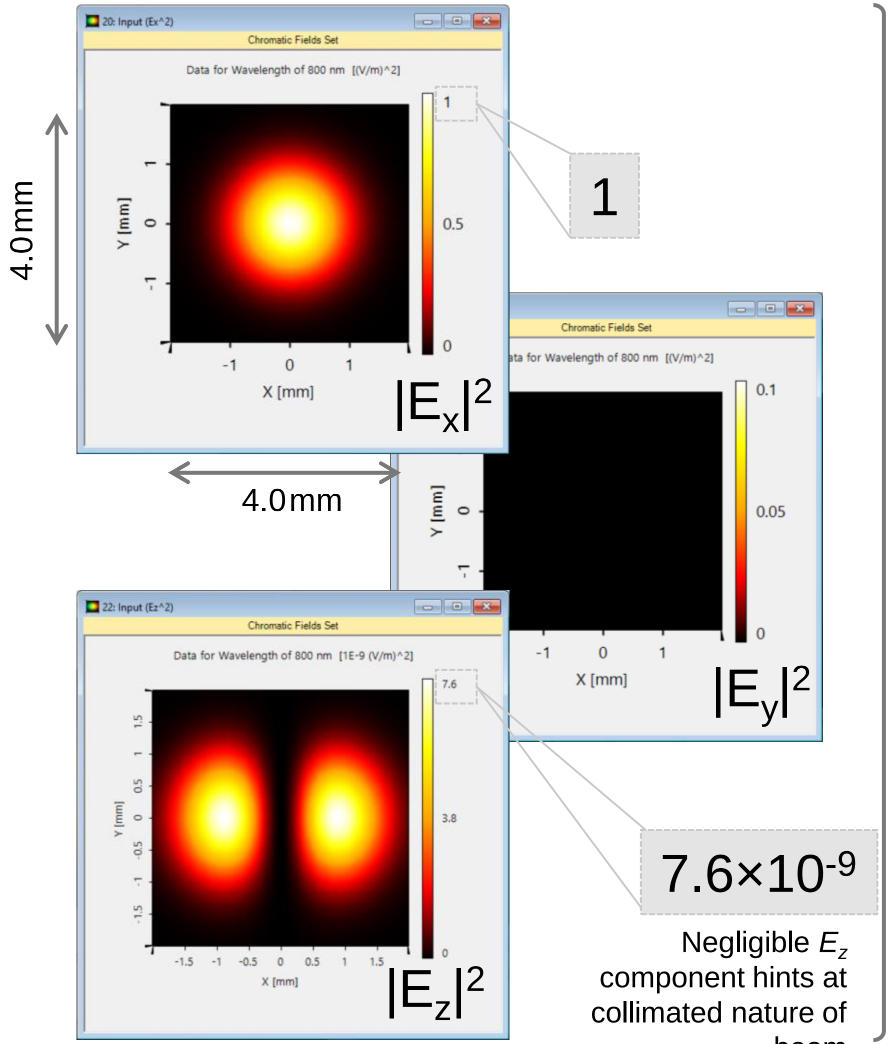




pulse evaluation
- spatial
- temporal

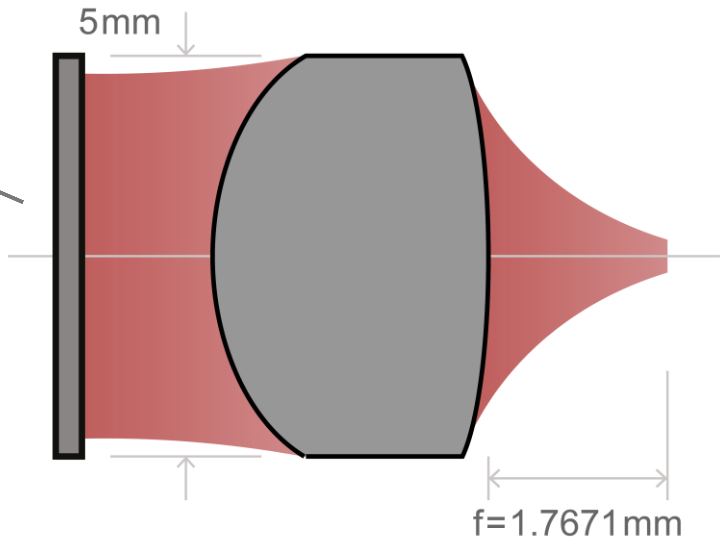
laser beam
- 5fs pulse
- 800nm carrier wavelength
- linearly polarized along *x*

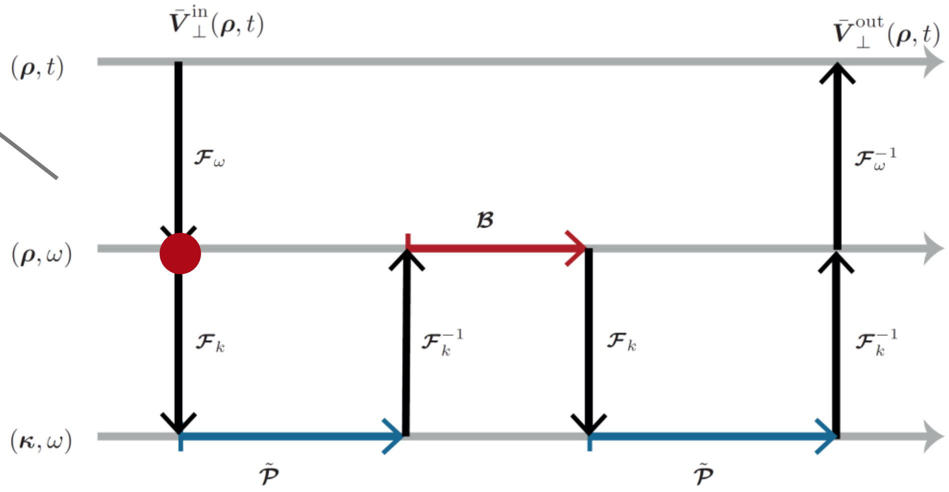


# Purely Spatial Analysis: Input Field (Carrier λ)

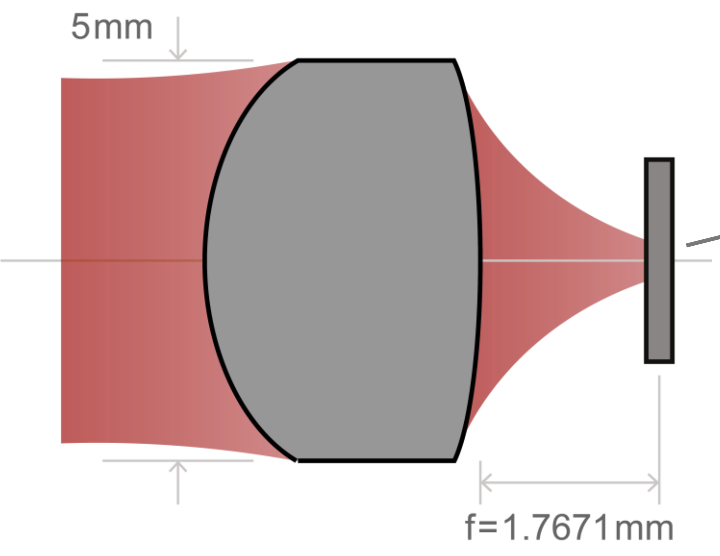


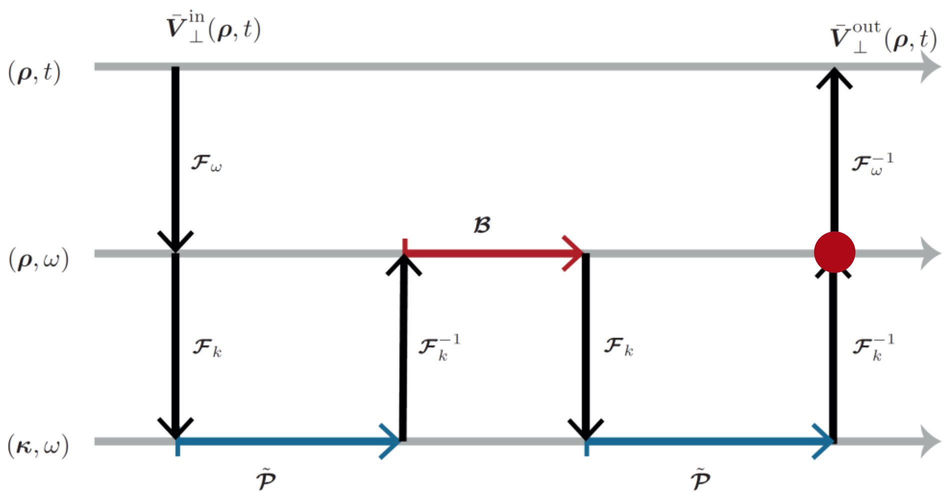

Negligible $E_z$ component hints at collimated nature of beam



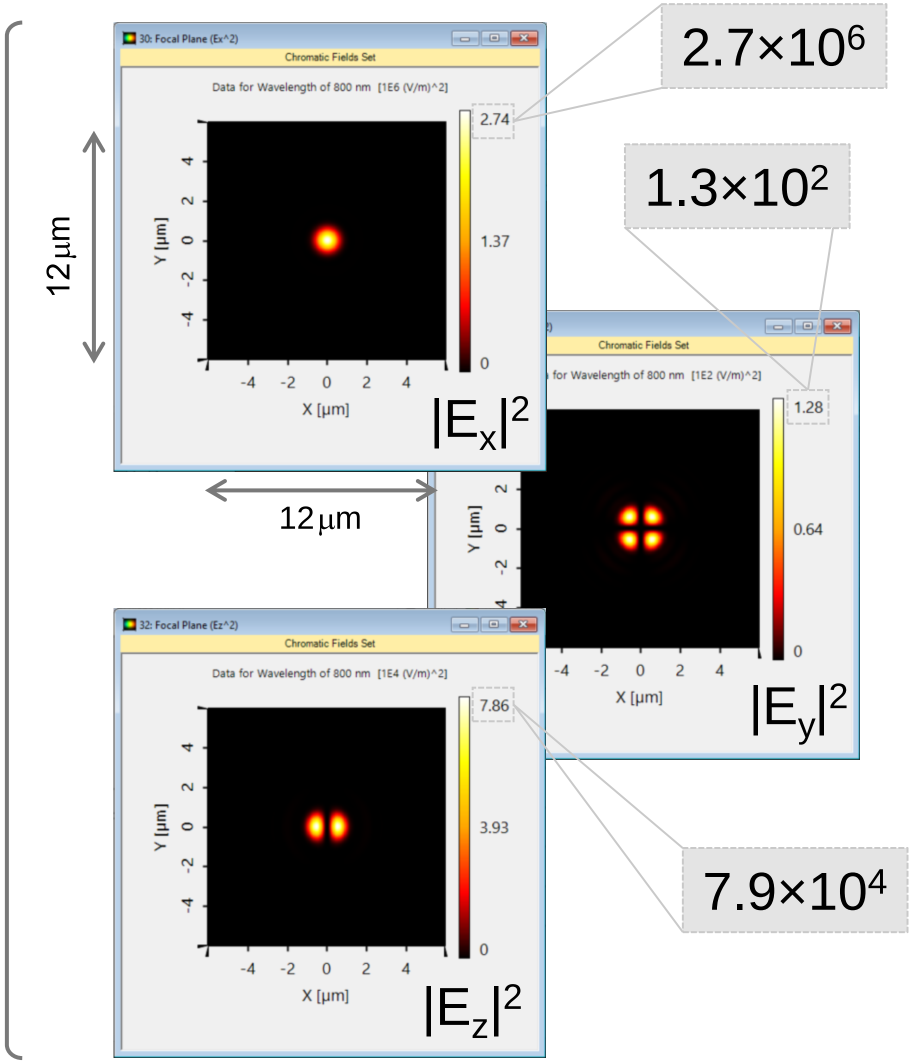
# Purely Spatial Analysis: Field at Focal Plane (Carrier λ)



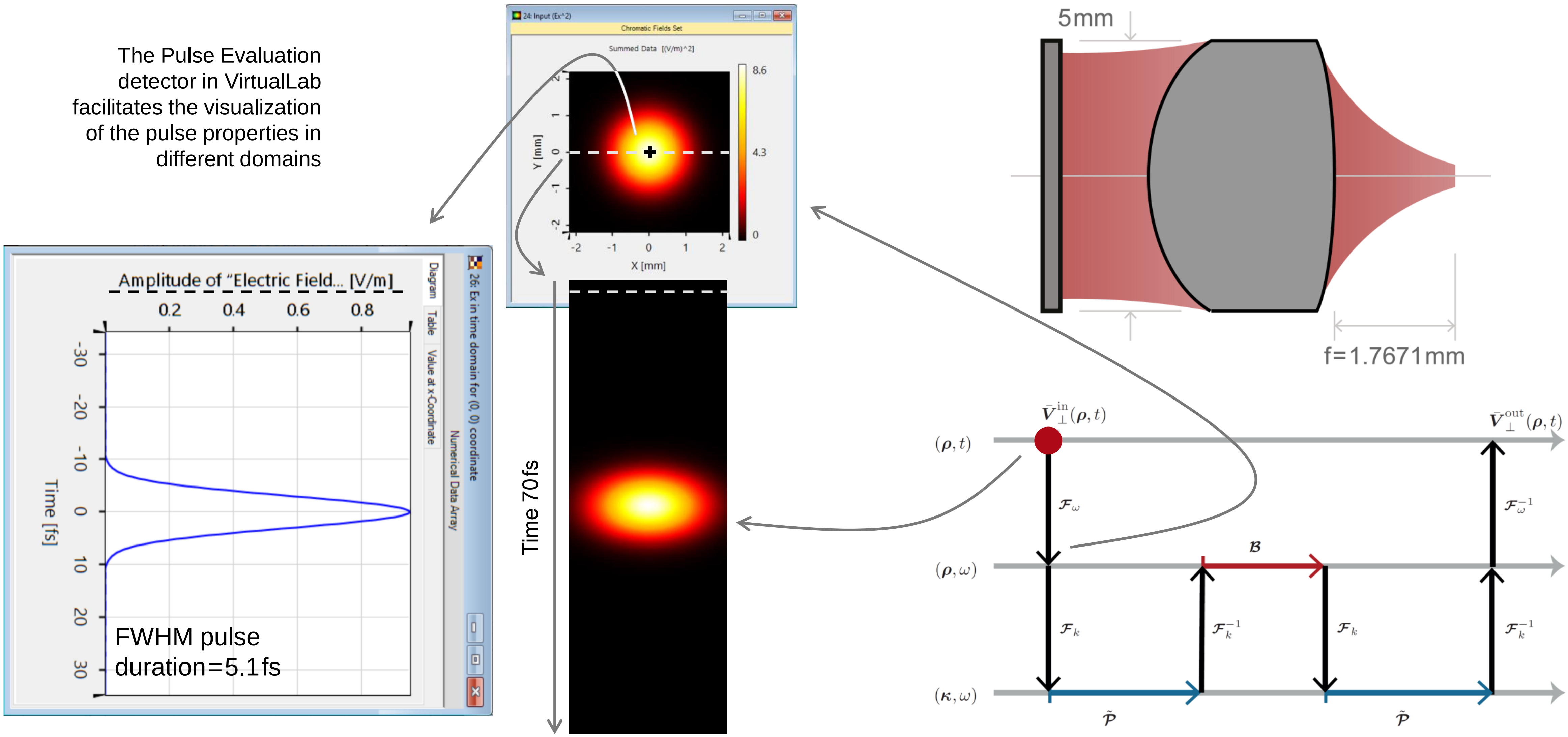

Note focusing effect on the size!

A small $E_y$ component appears from the cross-talk effected by the surfaces of the lens. The tangible $E_z$ component is the result of the high focusing of the beam



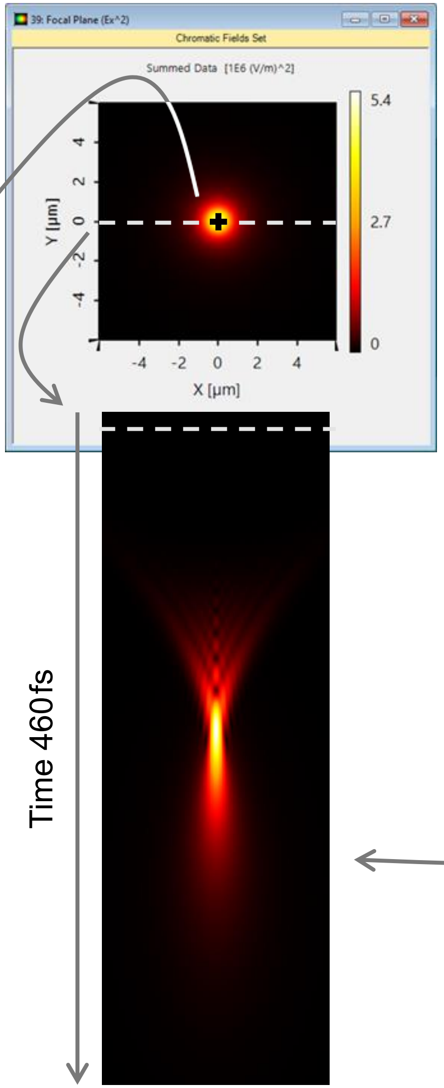
# Spatio-Temporal Analysis: Input Field ($E_x$ Component)

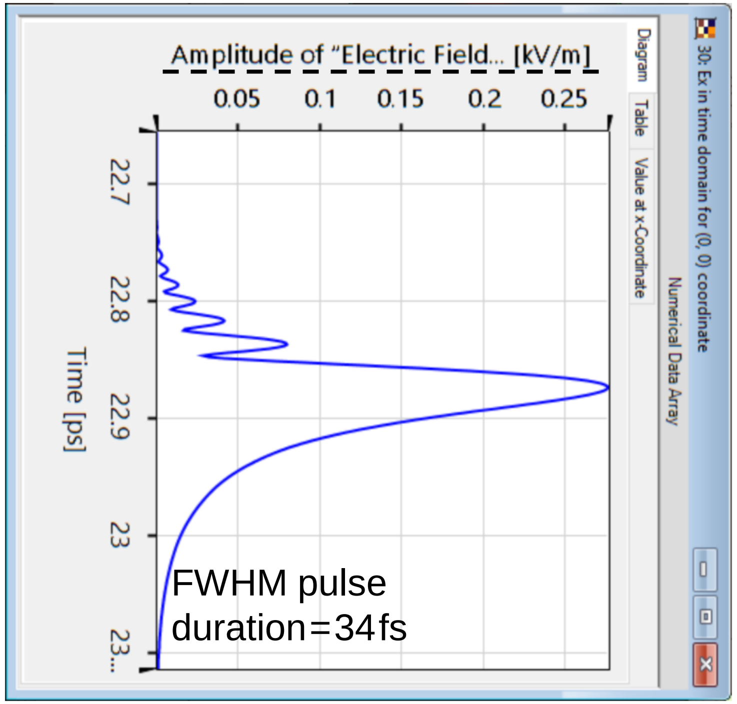
The Pulse Evaluation detector in VirtualLab facilitates the visualization of the pulse properties in different domains

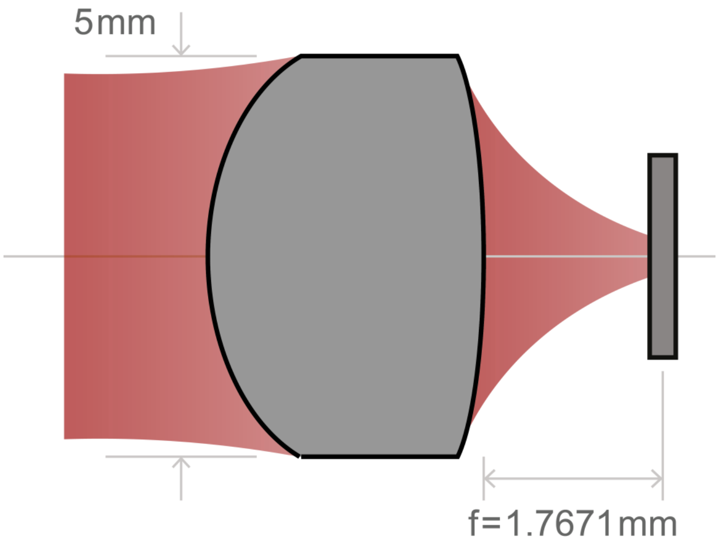


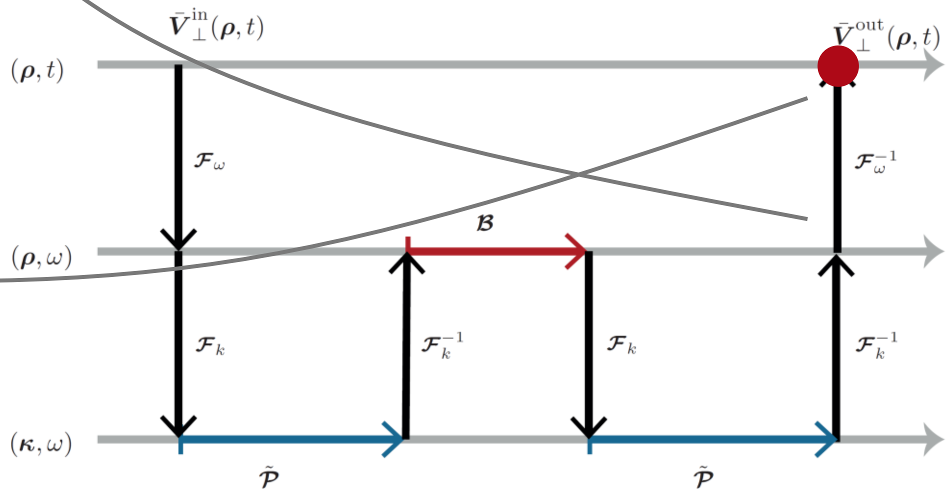
# Spatio-Temporal Analysis: Focus ($E_x$ Component)

The Pulse Evaluation detector in VirtualLab facilitates the visualization of the pulse properties in different domains

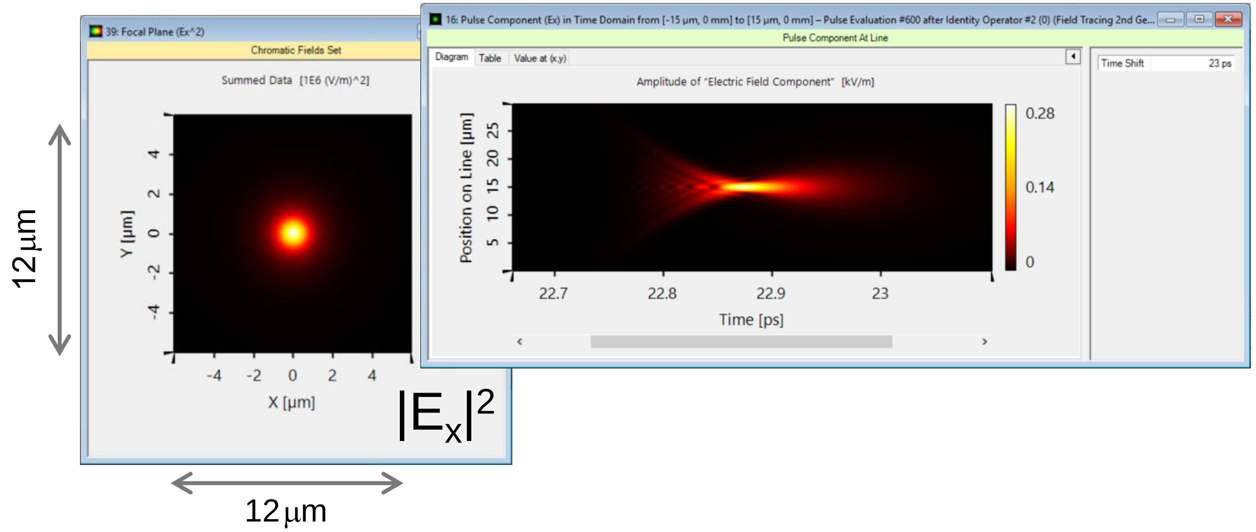


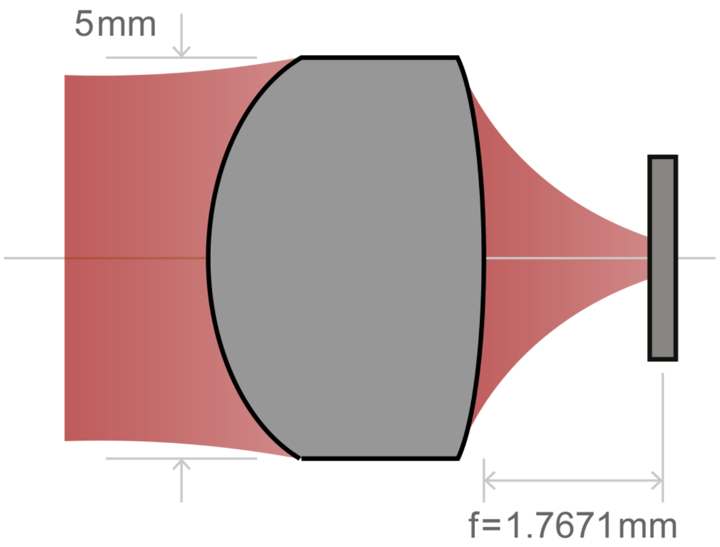



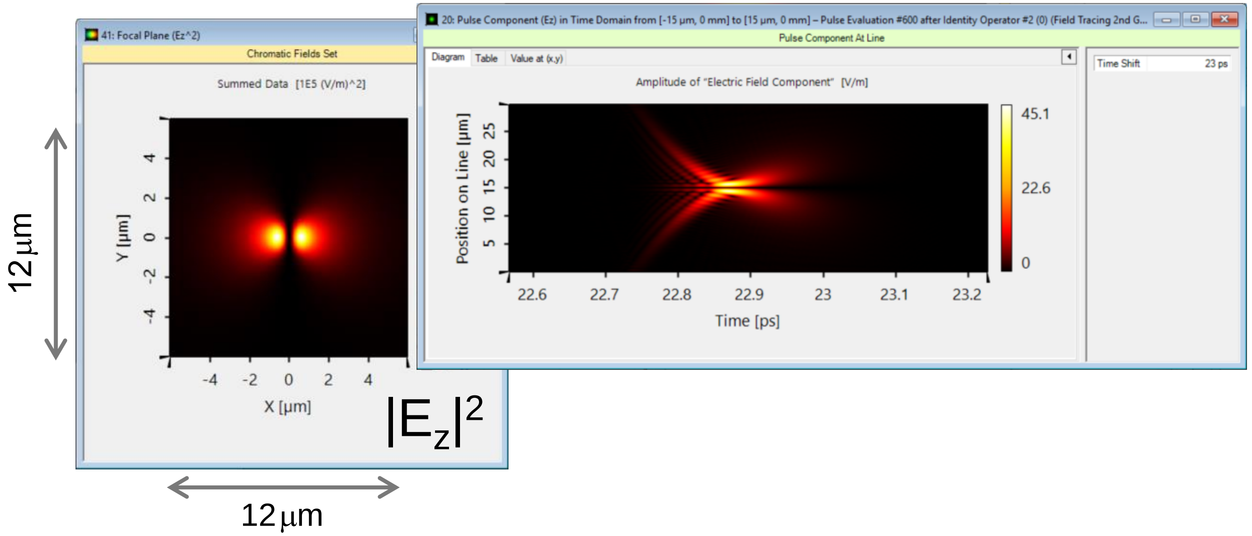



# Spatio-Temporal Analysis: Focus ($E_x$ and $E_z$)

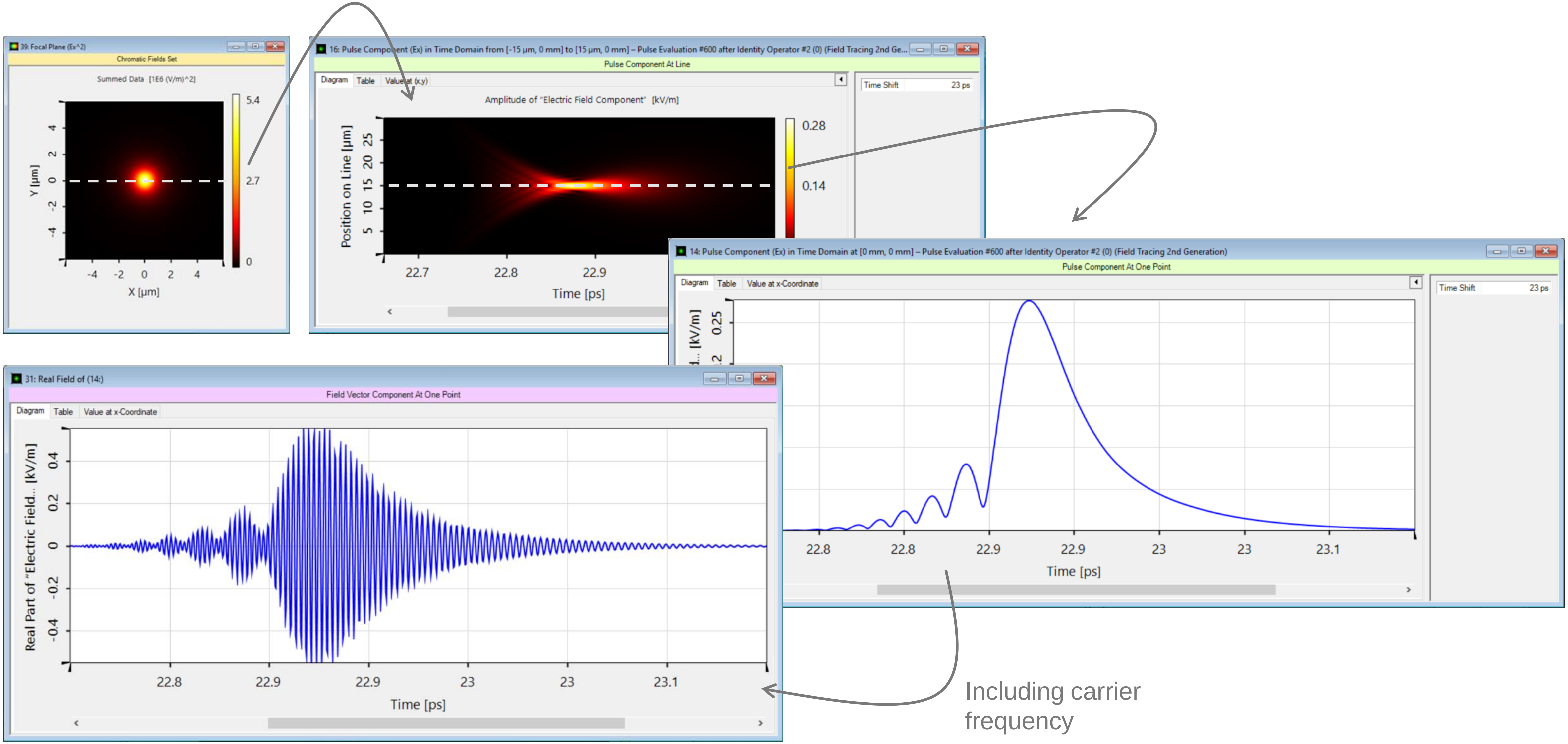


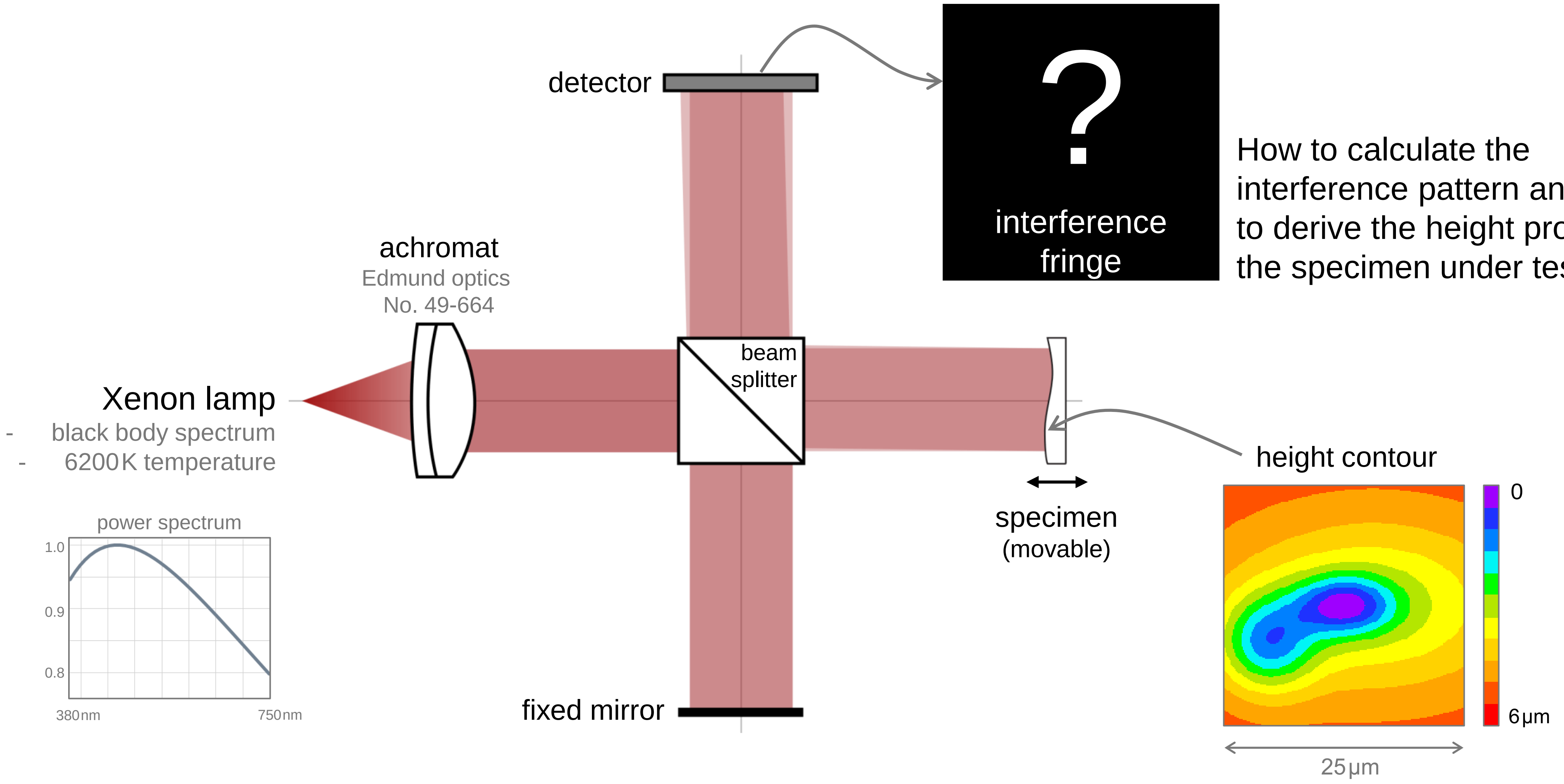

As always, consistent electromagnetic treatment in VirtualLab Fusion allows for the analysis of vectorial effects, also for ultra-short pulses

# Temporal Analysis: $E_x$ Component with Carrier Frequency



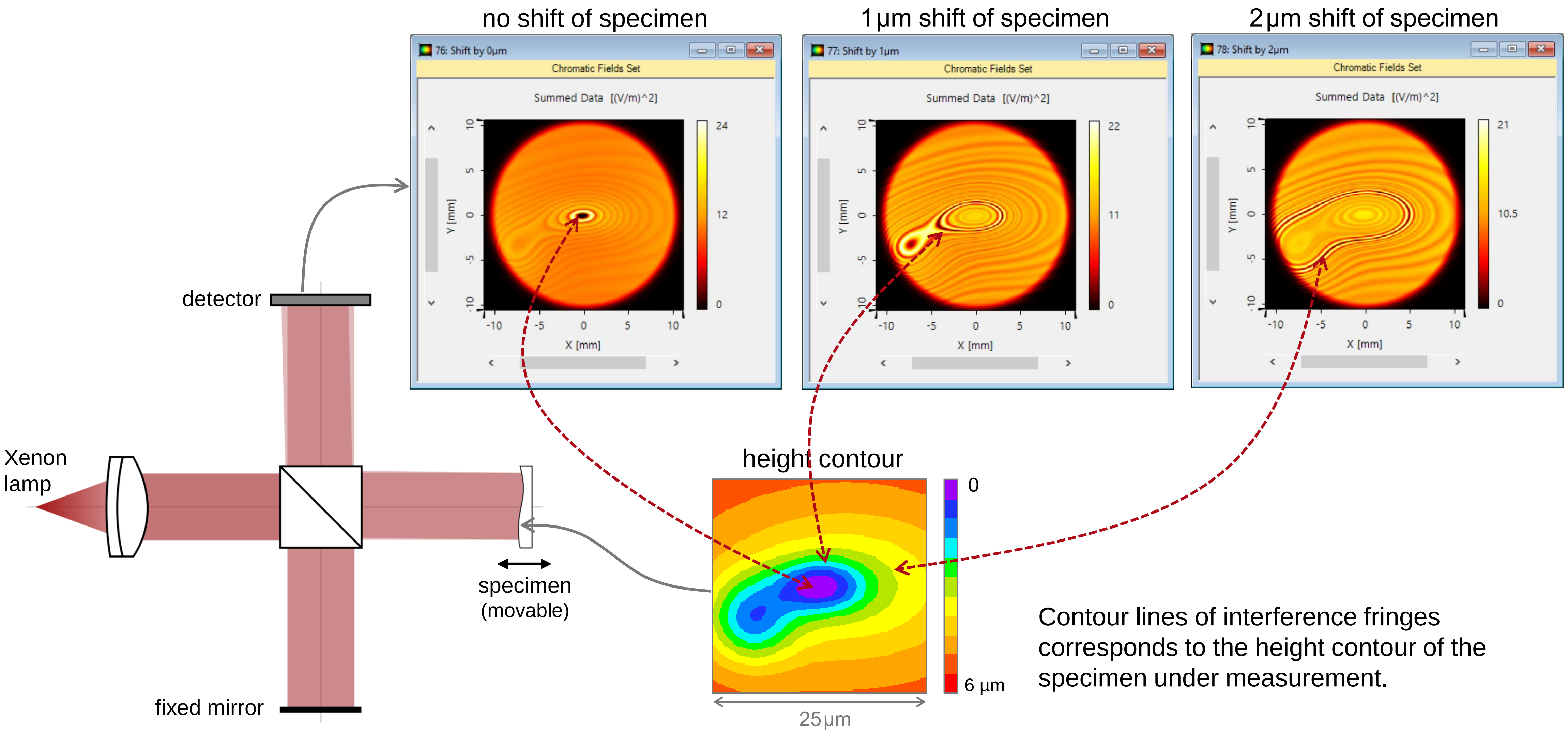
# Full-Field Optical Coherence Scanning Interferometry

# Modeling Task




How to calculate the interference pattern and even to derive the height profile of the specimen under test?

# Simulated Interference Fringes

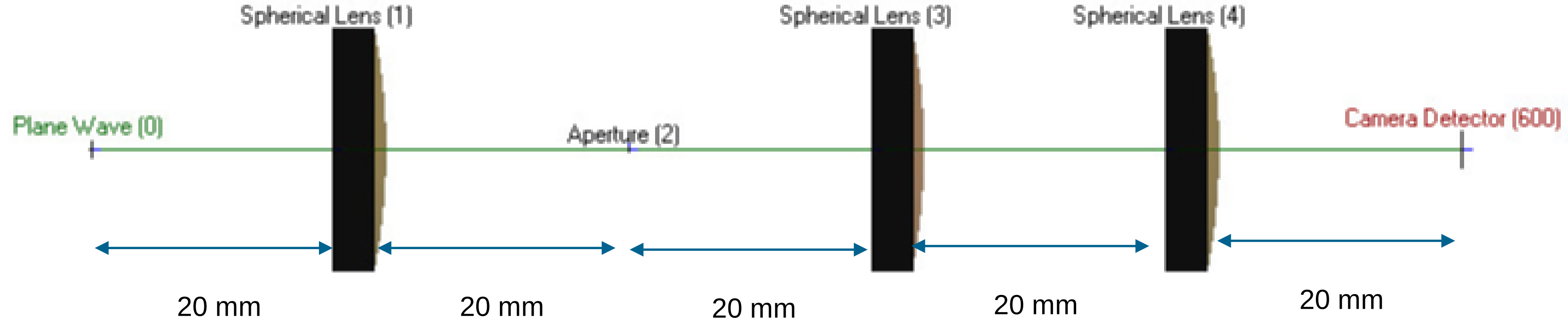



Contour lines of interference fringes corresponds to the height contour of the specimen under measurement.

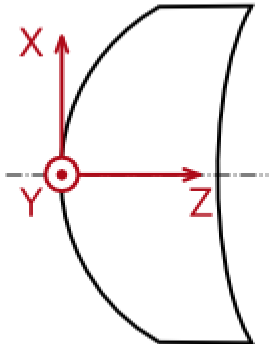
Part 7

# Position and Orientation

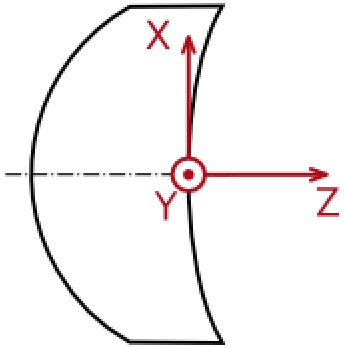
# Illustration: Position and Orientation



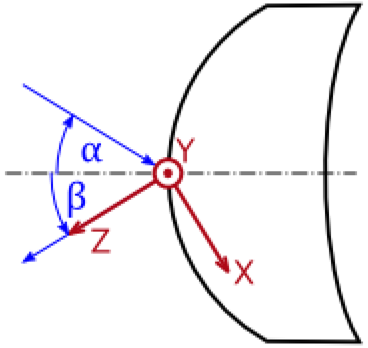
# Coordinate Systems



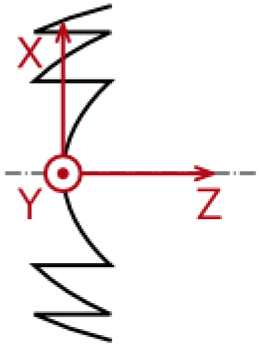

Input coordinate system



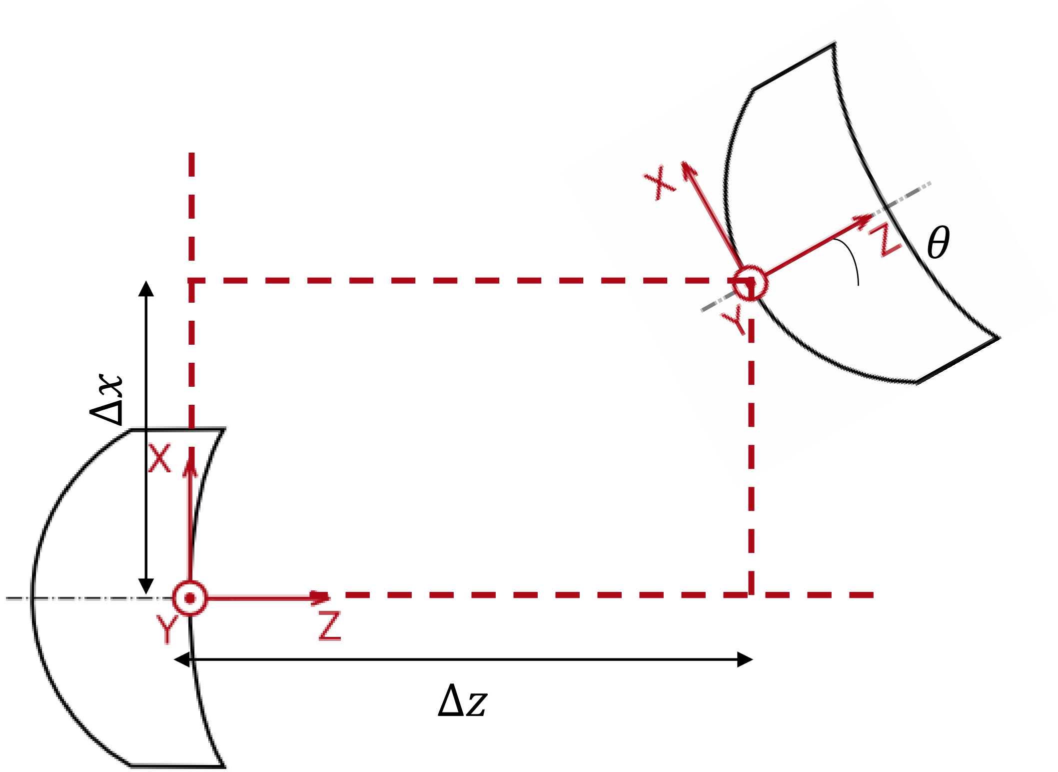

Transmission coordinate system



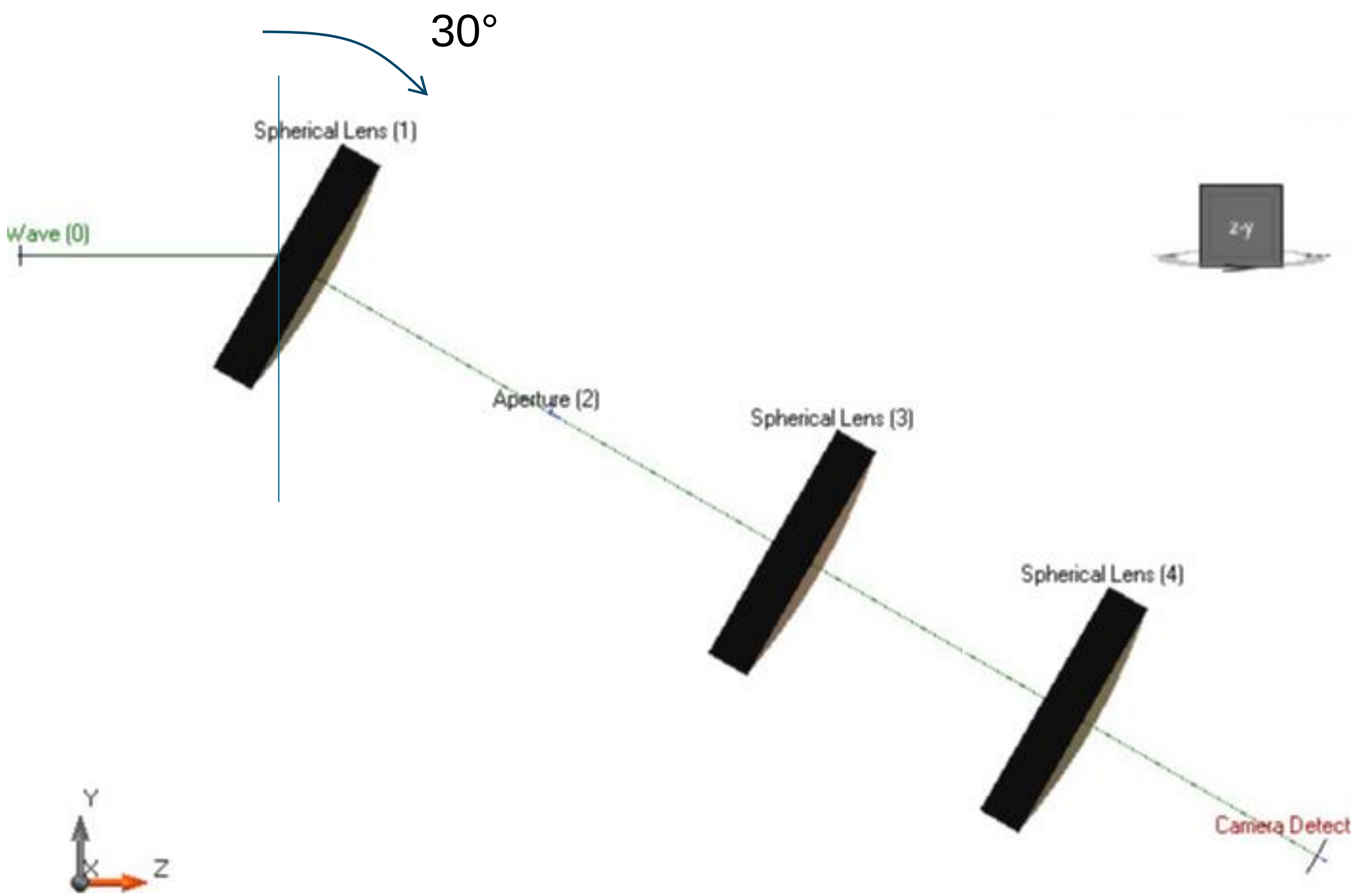

Reflection coordinate system



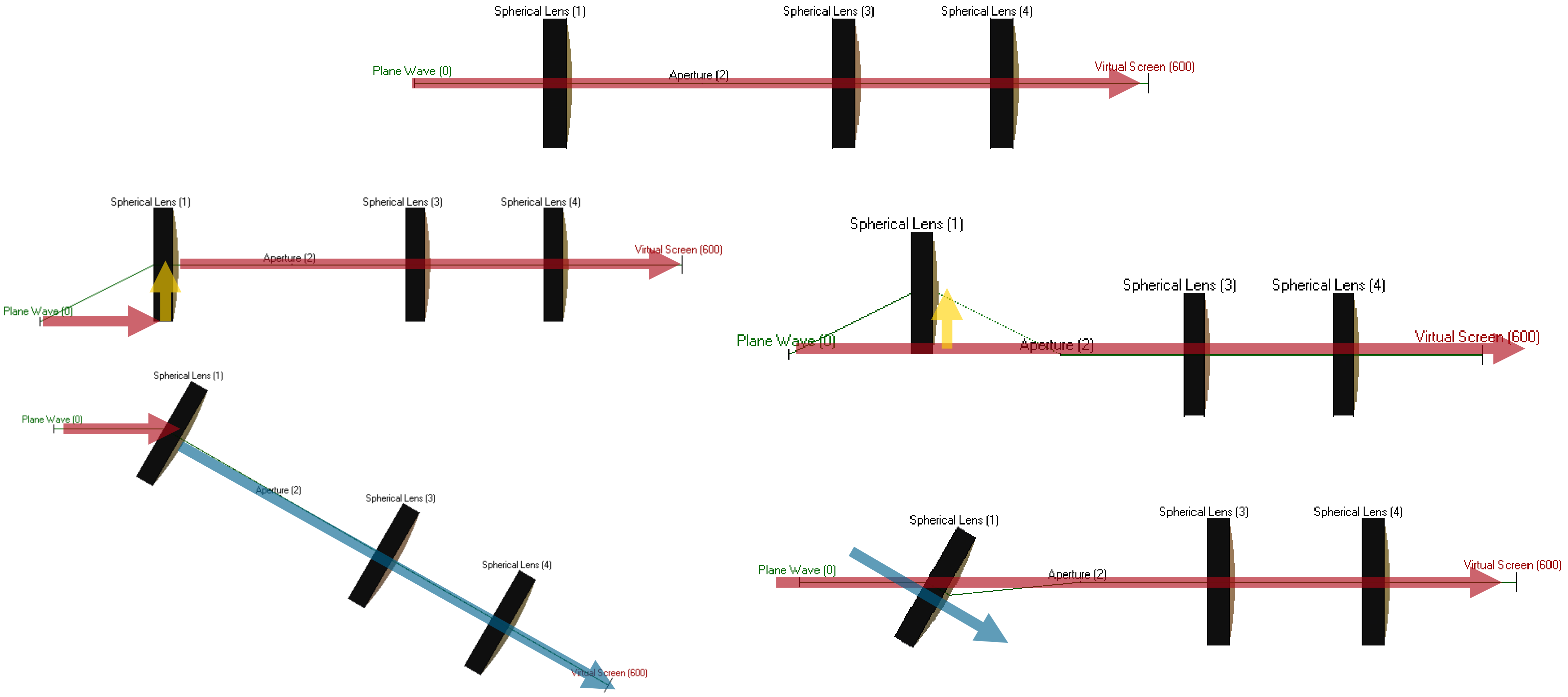
# Basal Positioning



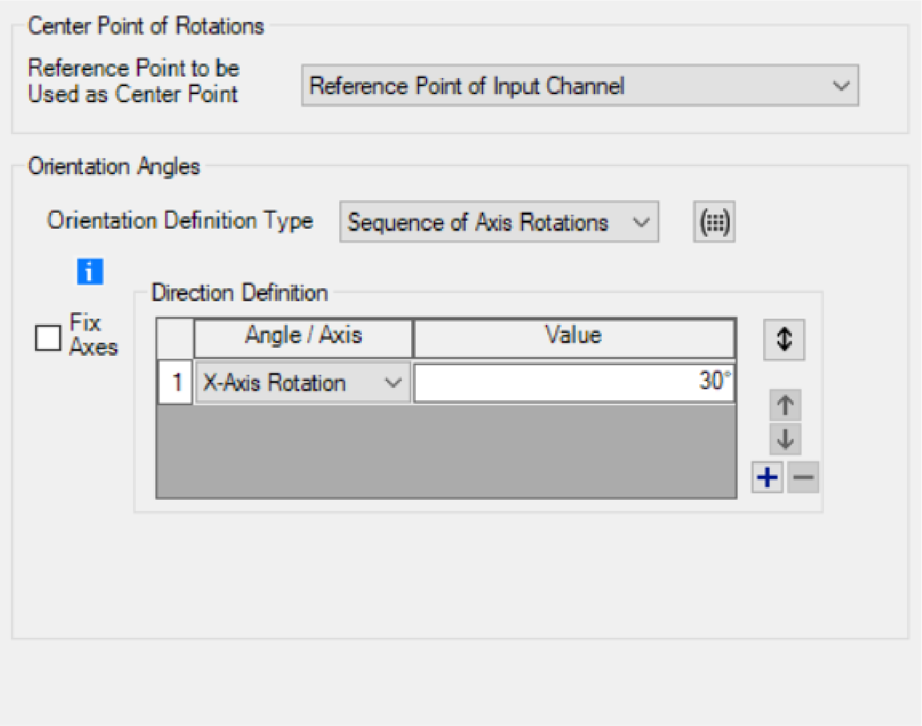
# Optical Setup

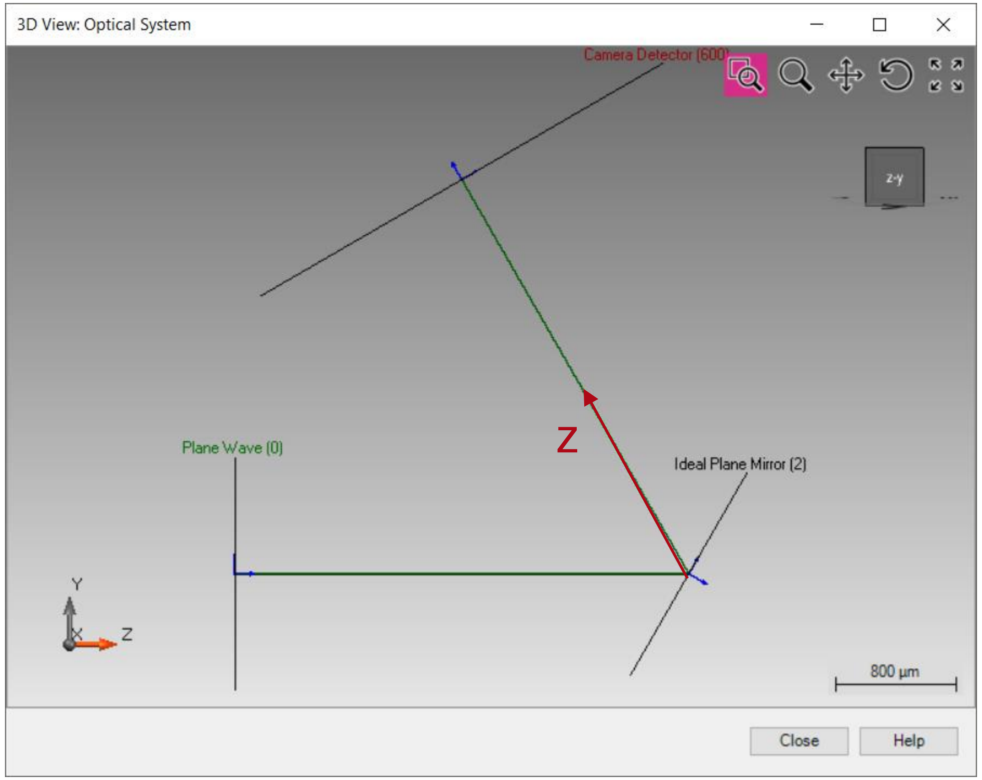


# Basal Positioning and Isolated Positioning



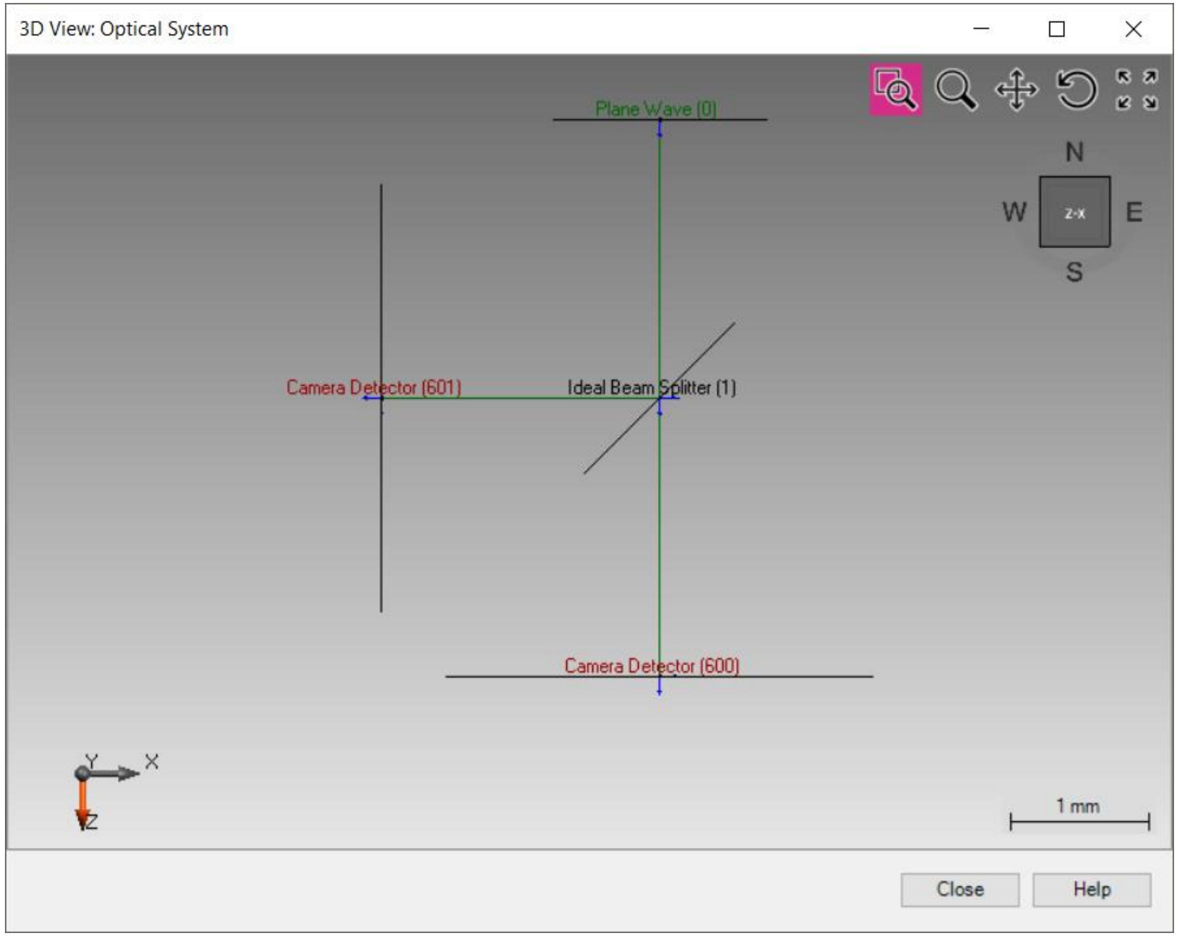
# Special Components: Mirror



# Special Components: Ideal Beam Splitter

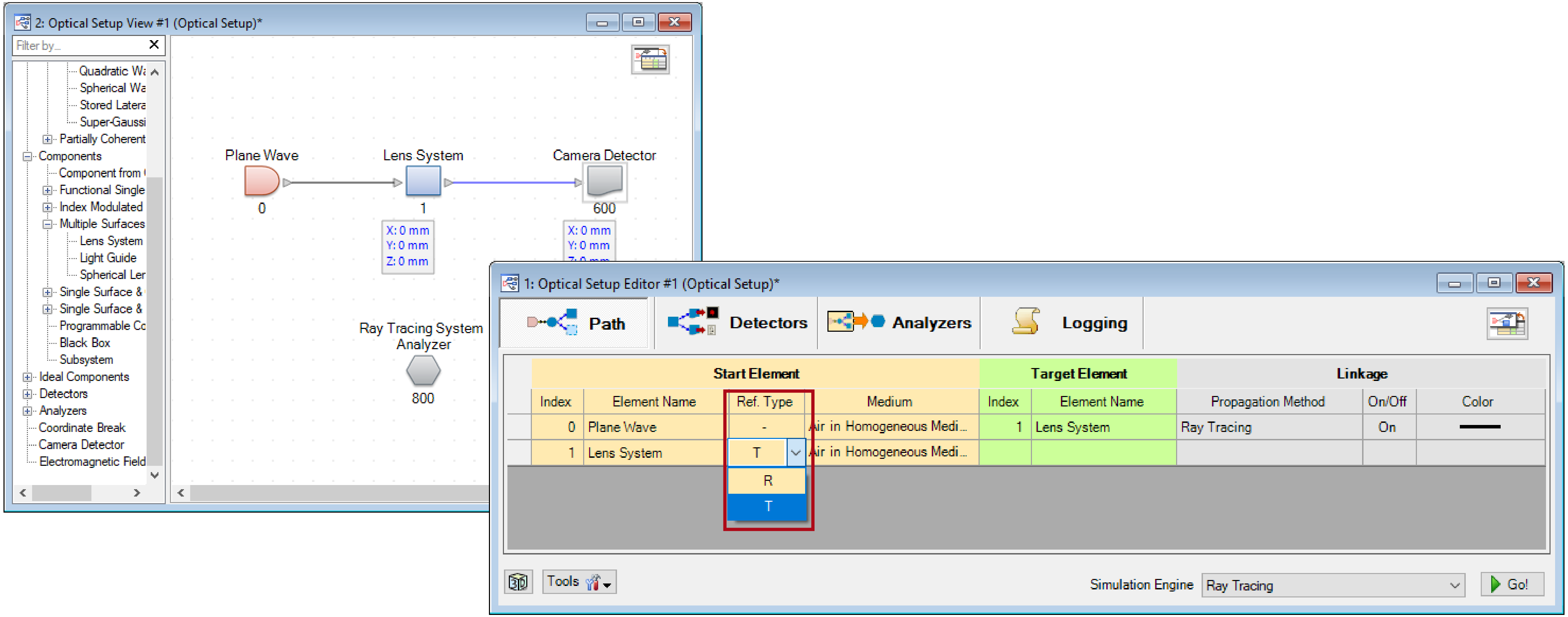
Two perpendicular output channels are automatically defined.



# Several Points about Positioning

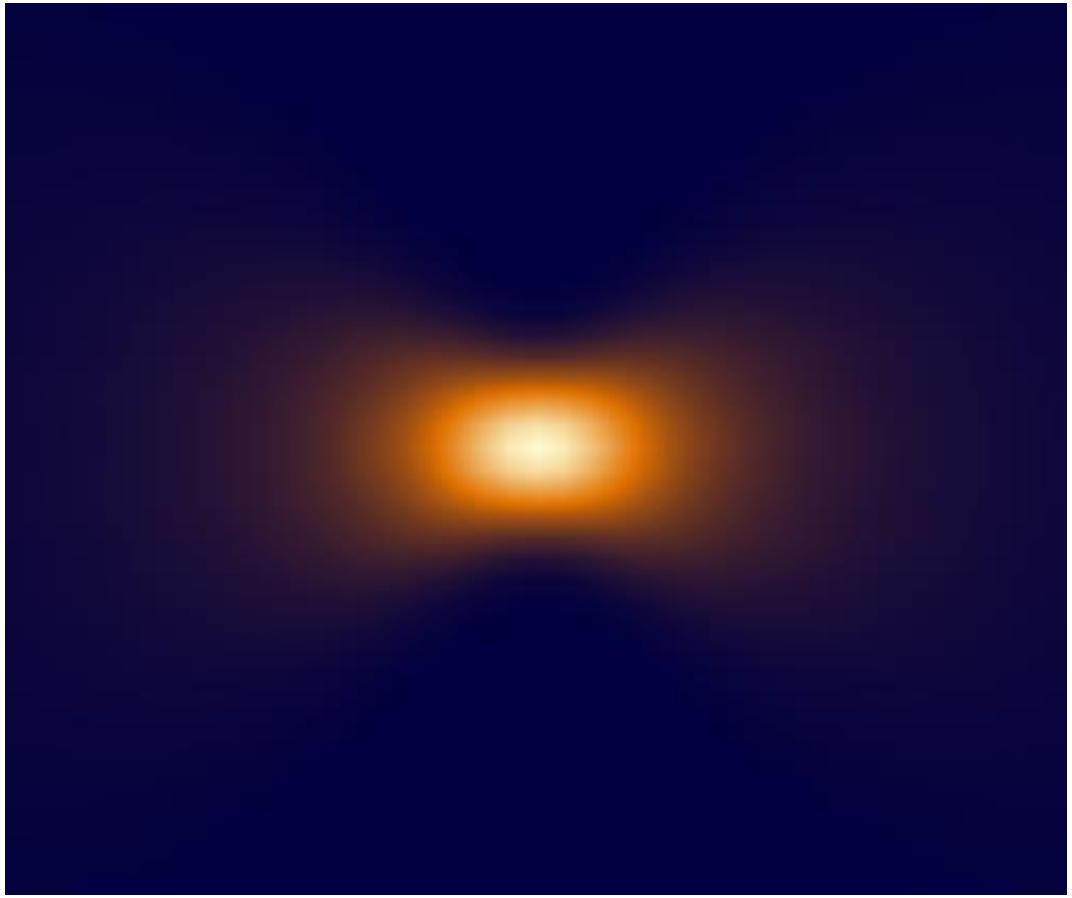
- Set position of one component: put the input CS in the output CS of previous component (Basal Positioning)
- Basal position: if one component shift or rotate, all the following path changes
- Isolated position: basal output CS doesn‘t change
- Special case:
  - Reflection CS: z-axis is calculated by component normal vector and input path
  - Beam splitter: two output CSs > 0 is transmission CS, 1 is reflection CS. Two z-axes are perpendicular

# Configuring Reference Type in VirtualLab Fusion



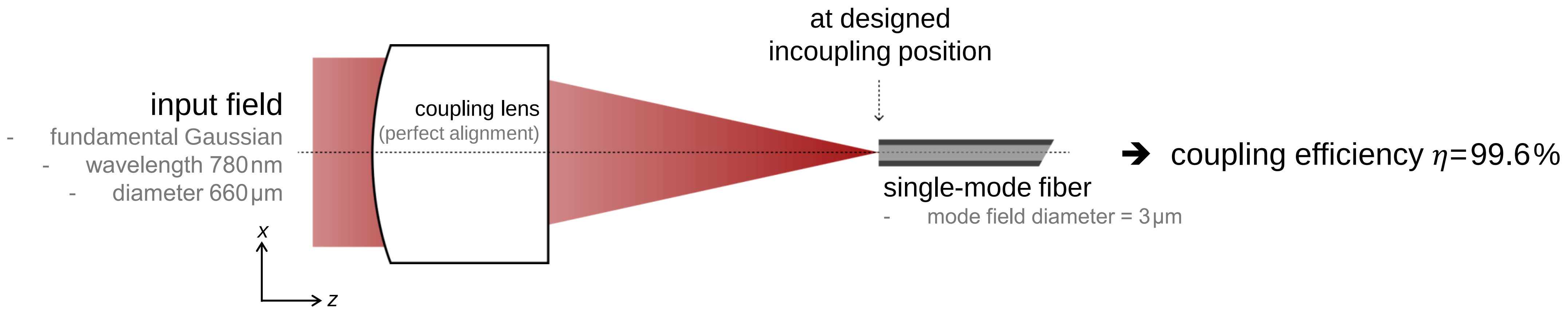
# Tolerance Analysis of a Fiber Coupling Setup

# Abstract

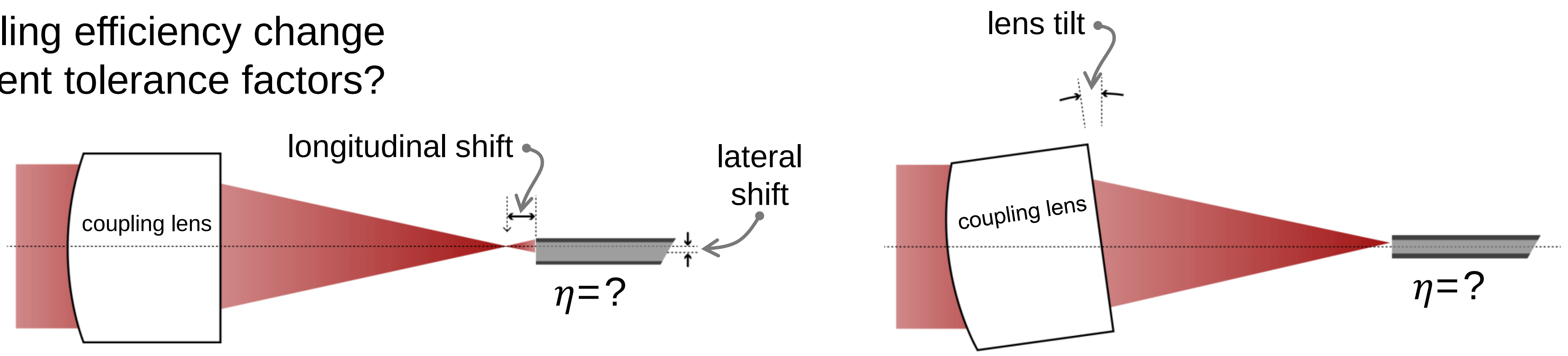


In modern optics, fibers can be found in various optical system, and it is usually of concern how much light can be coupled into fibers. The coupling efficiency can be sensitive to the system alignment, especially for single-mode fibers with relatively small core diameters. In this example, a well-designed fiber coupling lens is selected, and the coupling efficiency is evaluated with respect to different tolerance factors, such as the shift of fiber end position and the tilt of coupling lens.

# Modeling Task



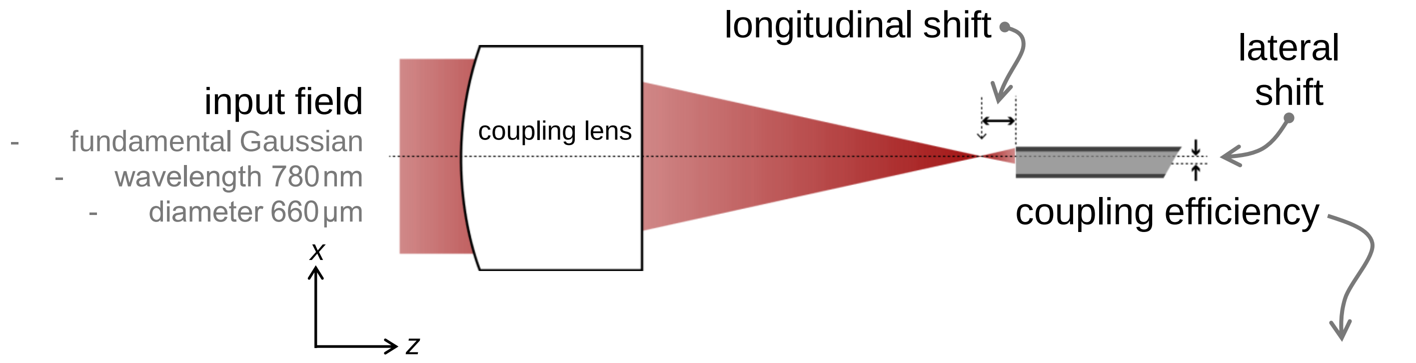

How does the coupling efficiency change with respect to alignment tolerance factors?



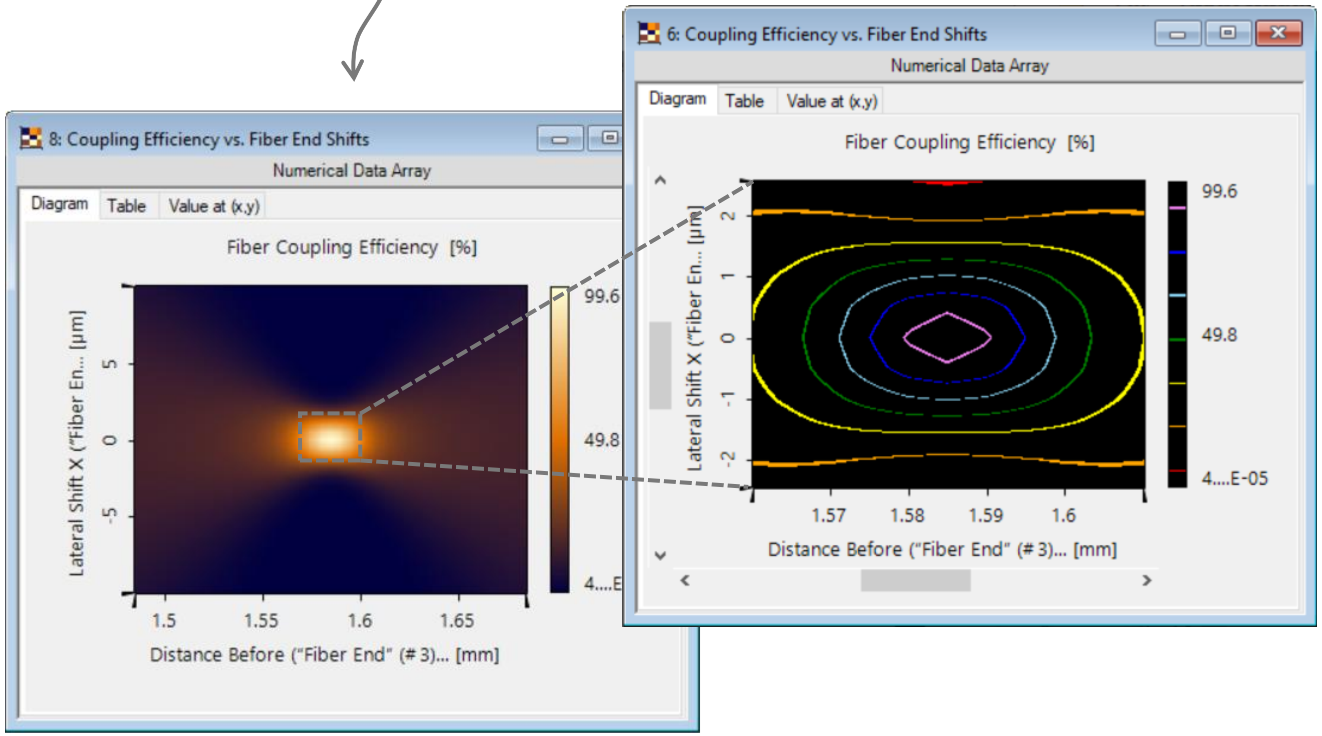

(a) shift of fiber end position

(b) tilt of coupling lens

# Coupling Efficiency vs. Fiber End Position Shift

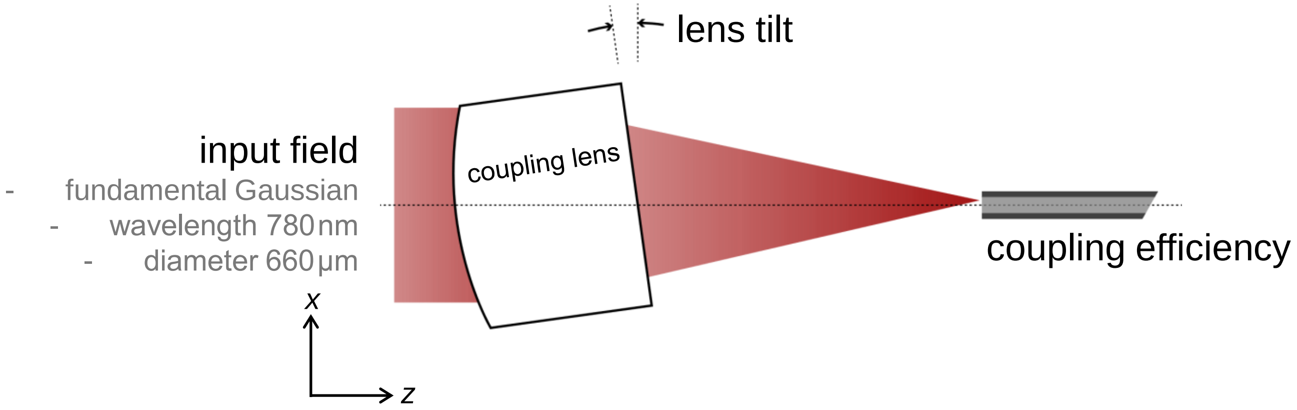



Contour plot helps with the identification of the parameter range for desired coupling efficiency threshold.

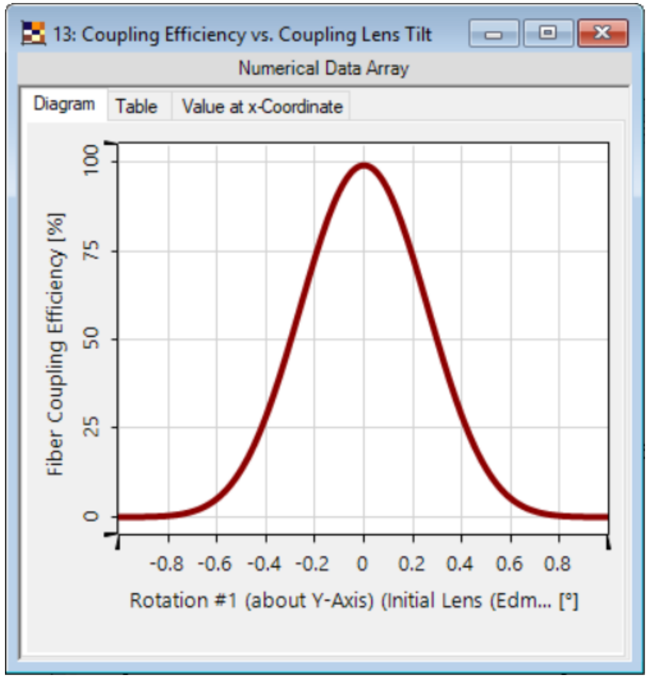



The coupling efficiency is scanned with respect to the fiber position shifts along both axial and lateral directions.

# Coupling Efficiency vs. Coupling Lens Tilt

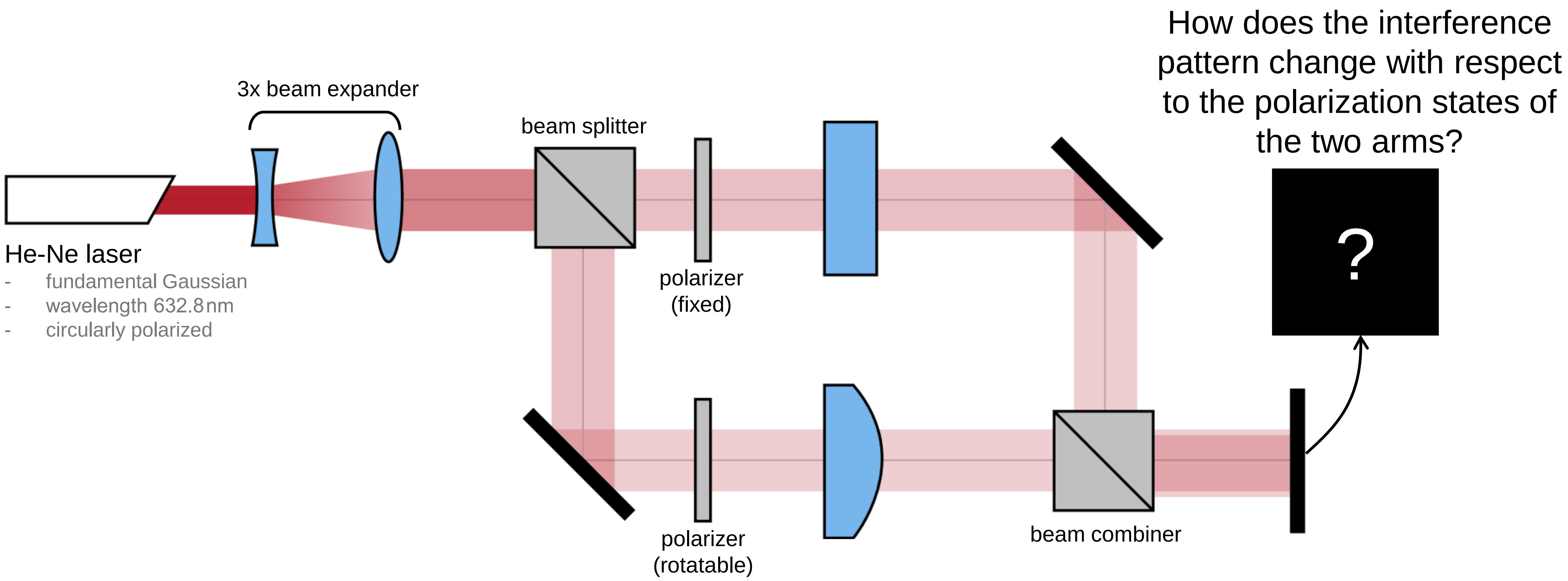



Physical-optics analysis of the coupling efficiency with respect to lens tilt. When lens tilt angle is within 0.1°, the coupling efficiency is still higher than 90%

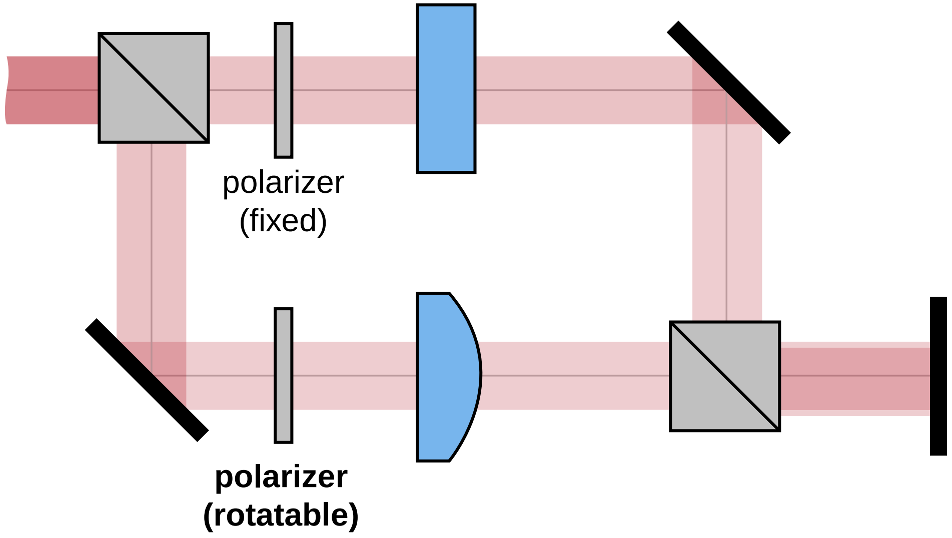


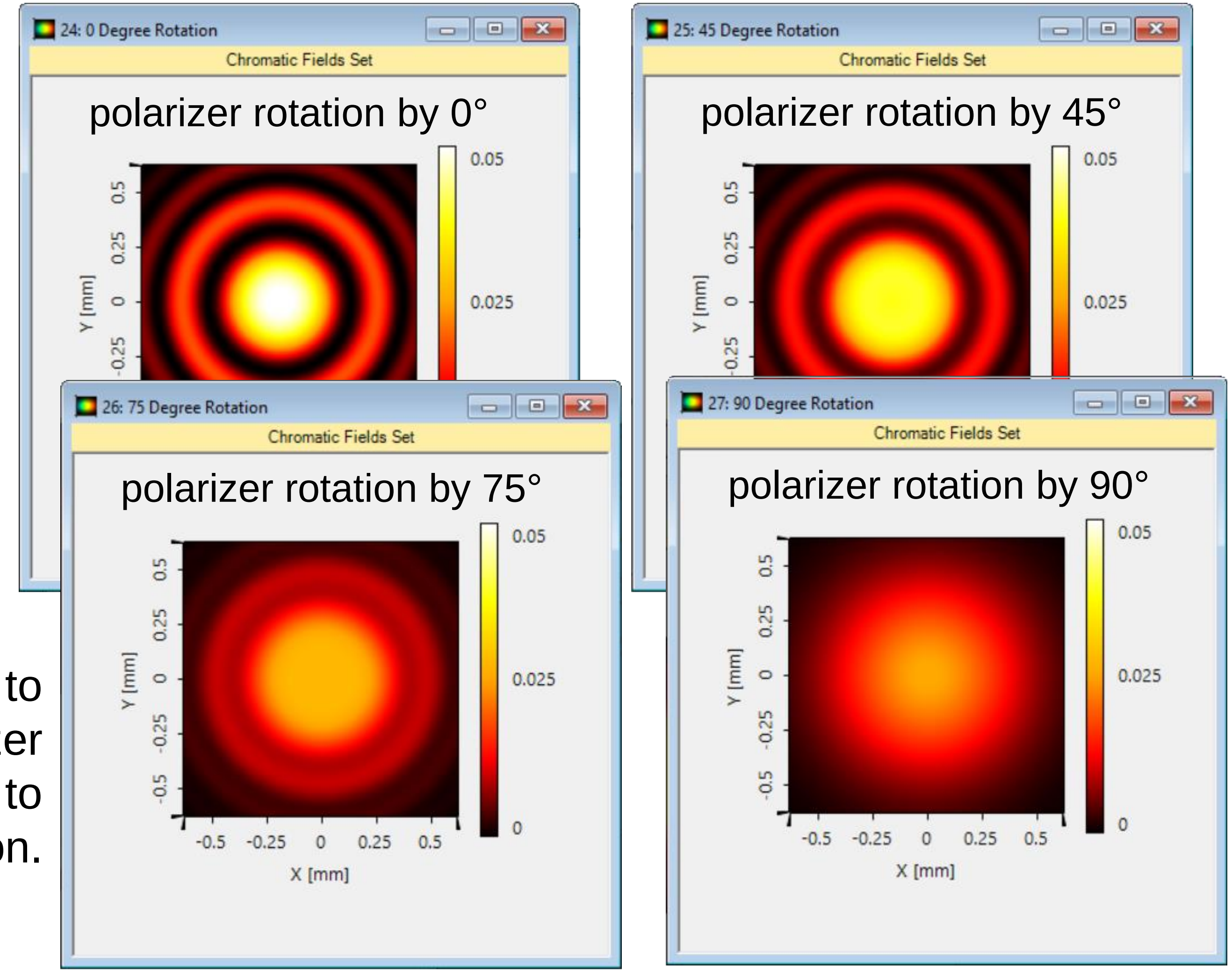
# Generation of Spatially Varying Polarization by Interference with Polarized Light

# Modeling Task



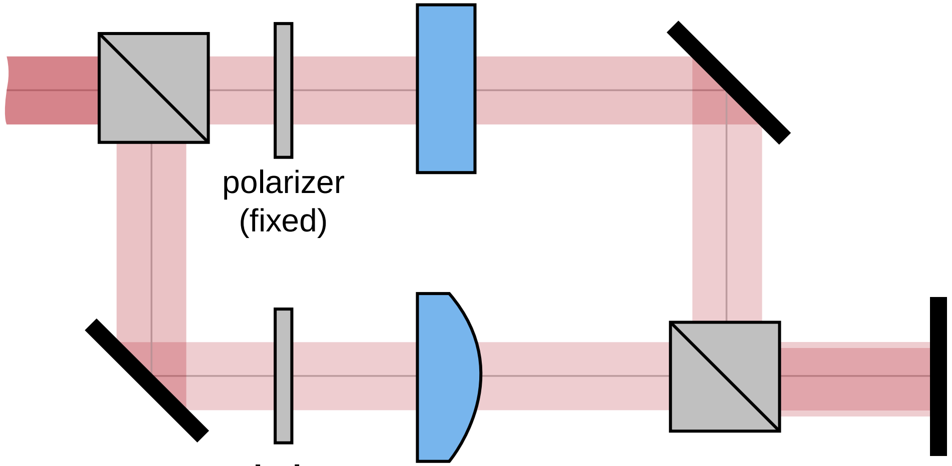
# Interference Pattern Changes with Polarizer Rotation

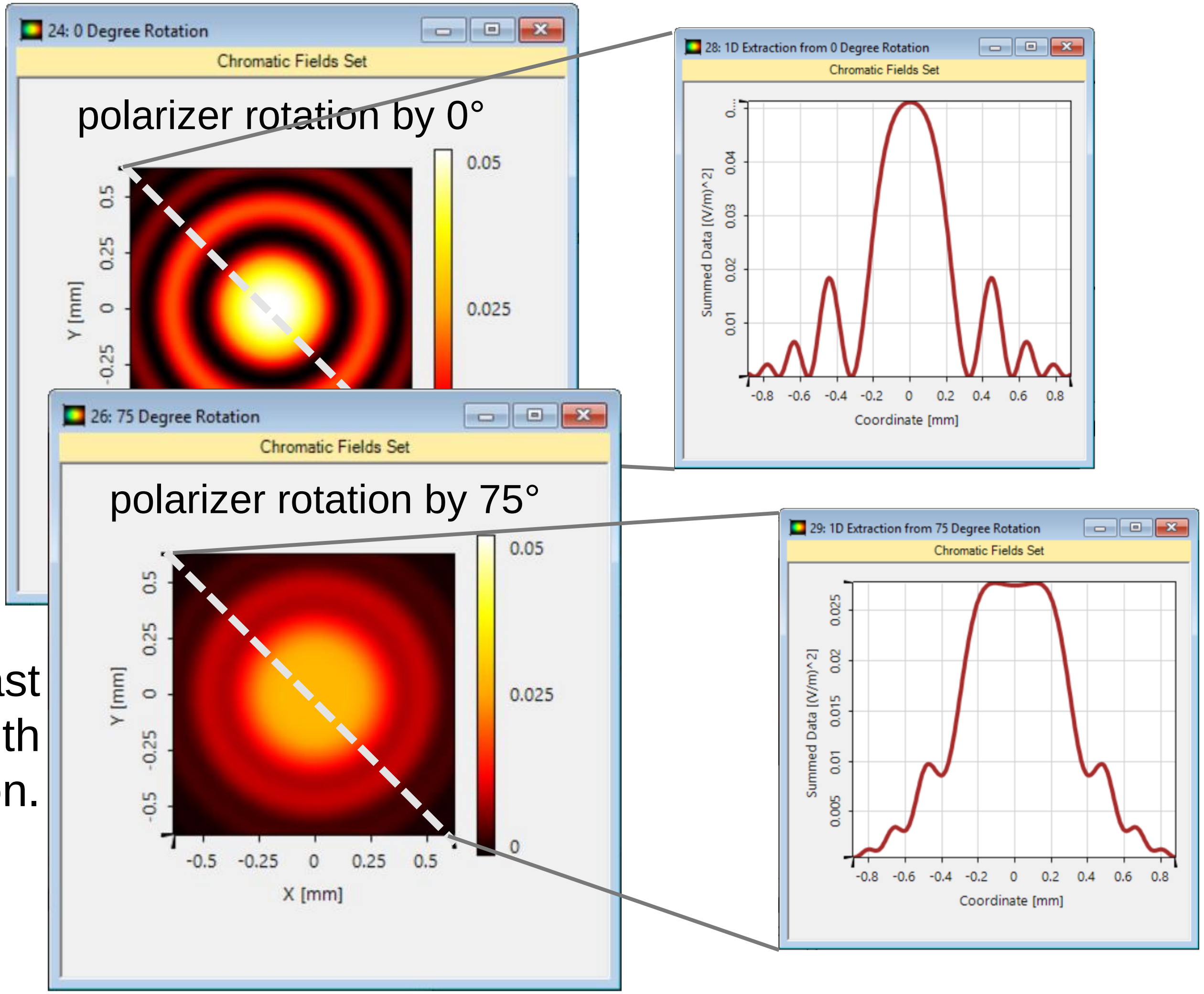



Interference fringes start to disappear, when polarizer rotates from parallel to orthogonal orientation.

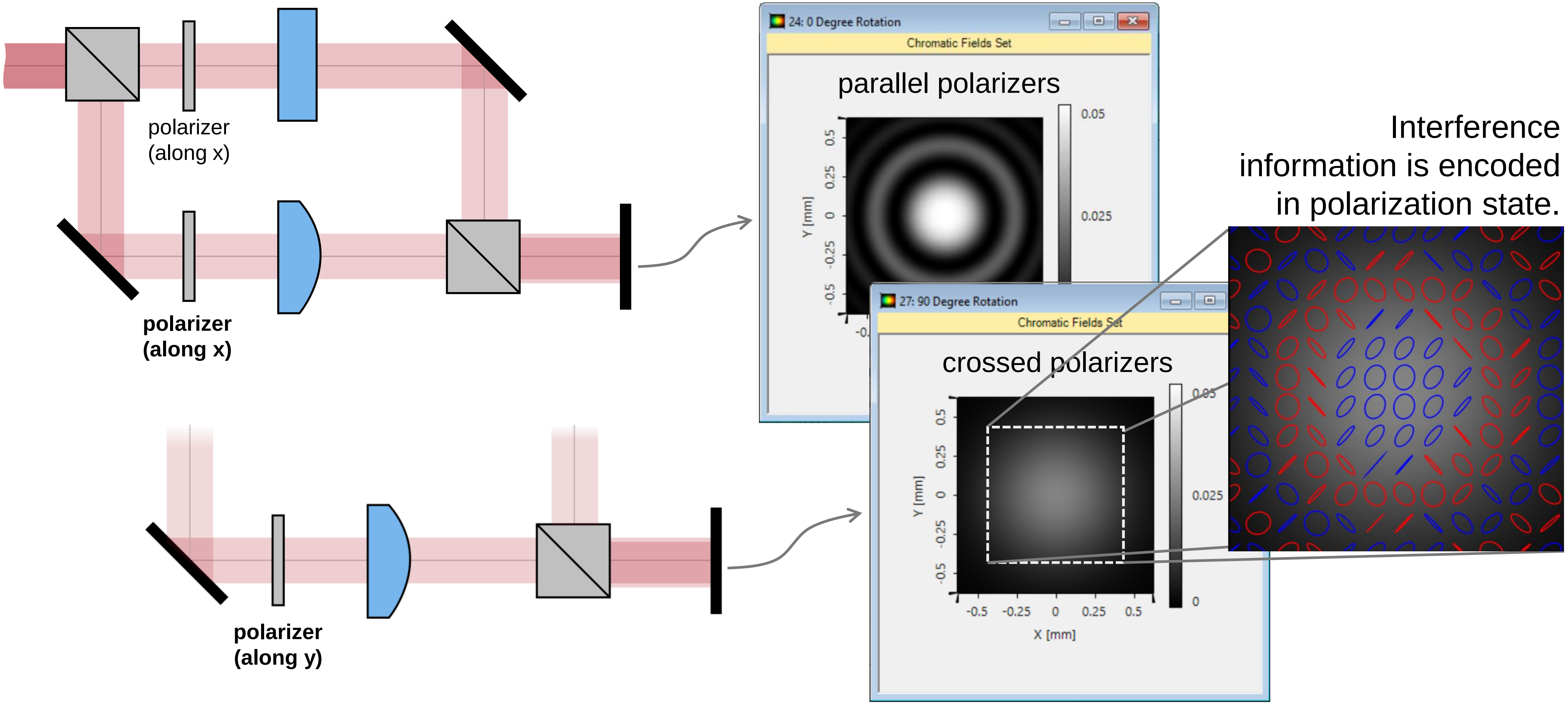
# Interference Pattern Changes with Polarizer Rotation




Fringe contrast changes with polarizer rotation.

# Interference Pattern



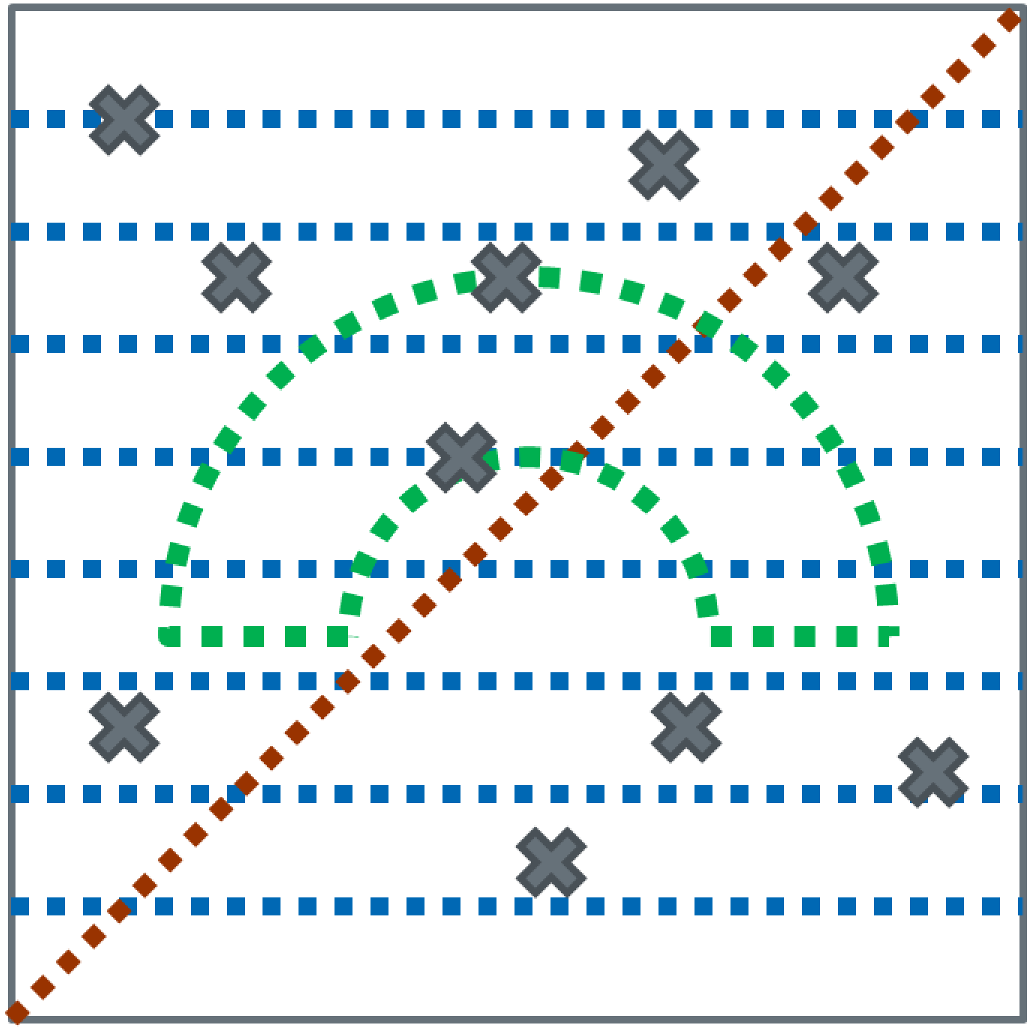

Interference information is encoded in polarization state.

# Annex

Some brief guides to useful features

# Parameter Run

# Abstract



For a given optical system, it is helpful to check its performance by controlling and varying selected parameters. VirtualLab Fusion provides a fully flexible and computationally efficient (via parallelization) Parameter Run, which enables the user specify different manners of parameter variations. As an example, it can be used for the tolerance analysis with respect to any system parameters under investigation. The analysis result can be visualized in different ways, such as single numbers, graphs, or even animations.

# Parameter Run Document

- The Parameter Run document allows the variation of the numerical parameters of an Optical Setup.

- It can be used e. g.
  - to investigate the system's sensitivity for parameter tolerances
  - to optimize parameters
  - to evaluate the changing profile of a beam in the vicinity of a focus
  - ...

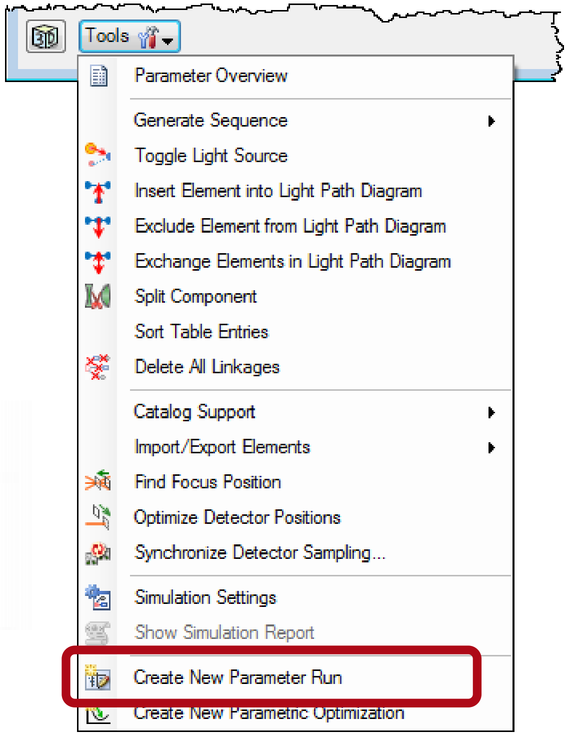
- One or multiple parameters can be varied.

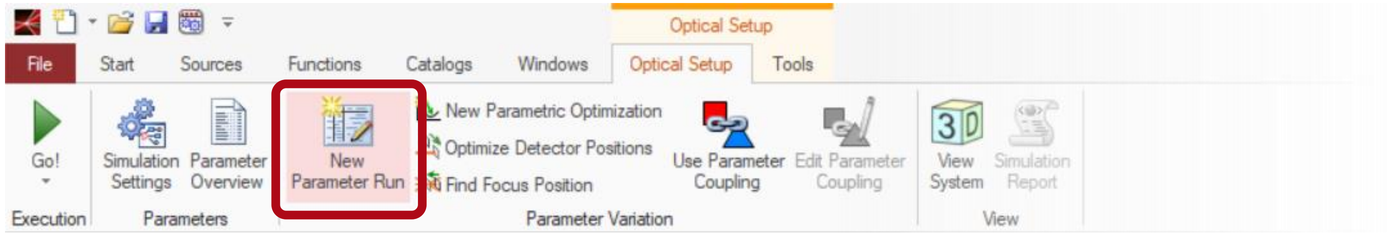
- Detector results are recorded within the Parameter Run document.

- A copy of the original Optical Setup is stored in the Parameter Run document.

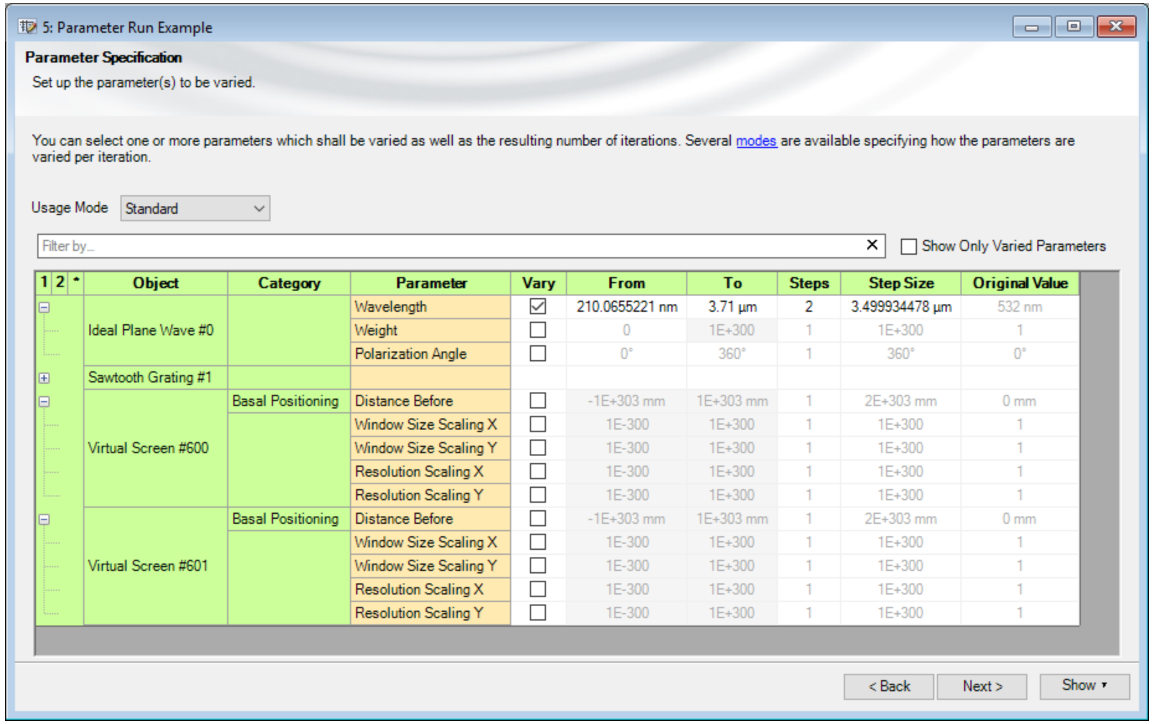
# New Parameter Run

- To generate a new Parameter Run an open and activated Optical Setup window is required.
- A new Parameter Run document can be generated via
  - ribbon
  - Optical Setup Tools
  - shortcut Ctrl + P



# Parameter Specification Page

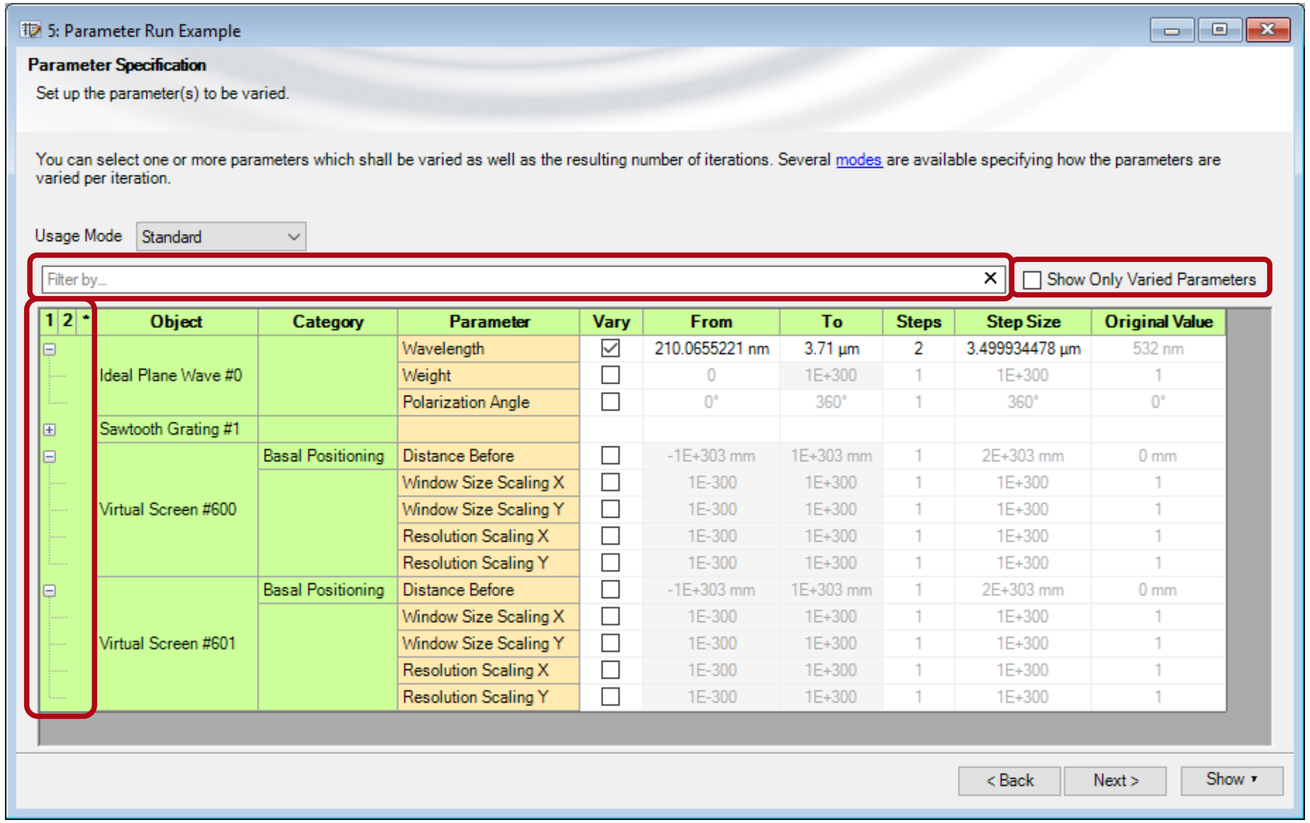



- This page allows you to select the parameters that should be varied.
- The parameter range and the number of steps can be specified.
- Four different Usage Modes (Standard, Programmable, Scanning, Random) will be Explained later.

# Parameter Specification Page

You can

- filter for specific parameters
- show only the ones that are already set for variation
- fold/unfold the parameter list for a clearer representation by using the first three columns



# Usage Modes

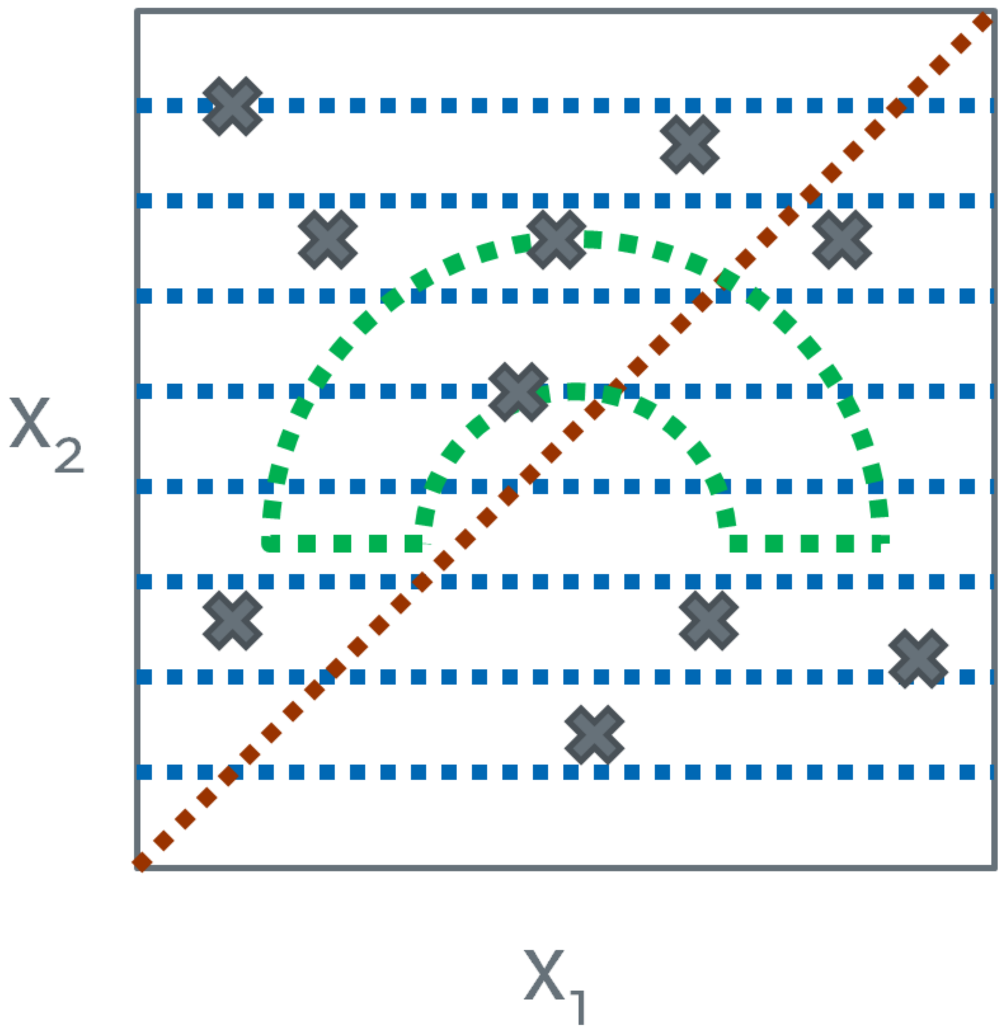
- Standard Mode:
  Linear variation of all selected parameters between minimum and maximum value.

- Programmable Mode:
  Customized parameter values per variation step. A table with the parameter values per variation step is filled by a snippet.

- Scanning Mode:
  Scan of parameter space – all possible parameter combinations are simulated.

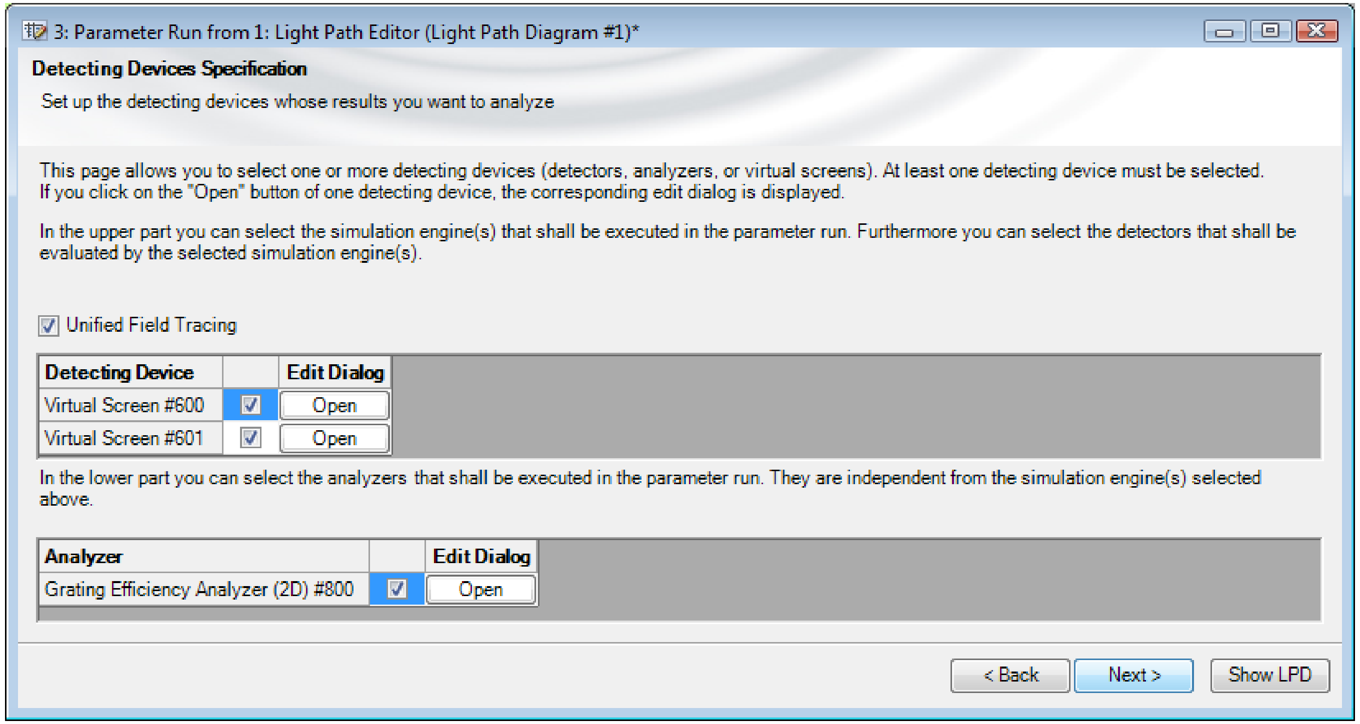
- Random Mode:
  Random variation of parameters between minimum and maximum value. Sometimes also called Monte-Carlo-Simulation. A seed can be used for reproducible results.

# Usage Modes




- Illustration of the different usage modes for the parameter run. A two-dimensional parameter space defined by two parameters $X_1$ and $X_2$ is shown.
- Red: Resulting parameter sets for the standard mode.
- Green: Example how the parameter sets can be generated by a snippet in the programmable mode.
- Blue: Resulting parameter sets for the scanning mode.
- Grey: Some randomly generated parameter sets.

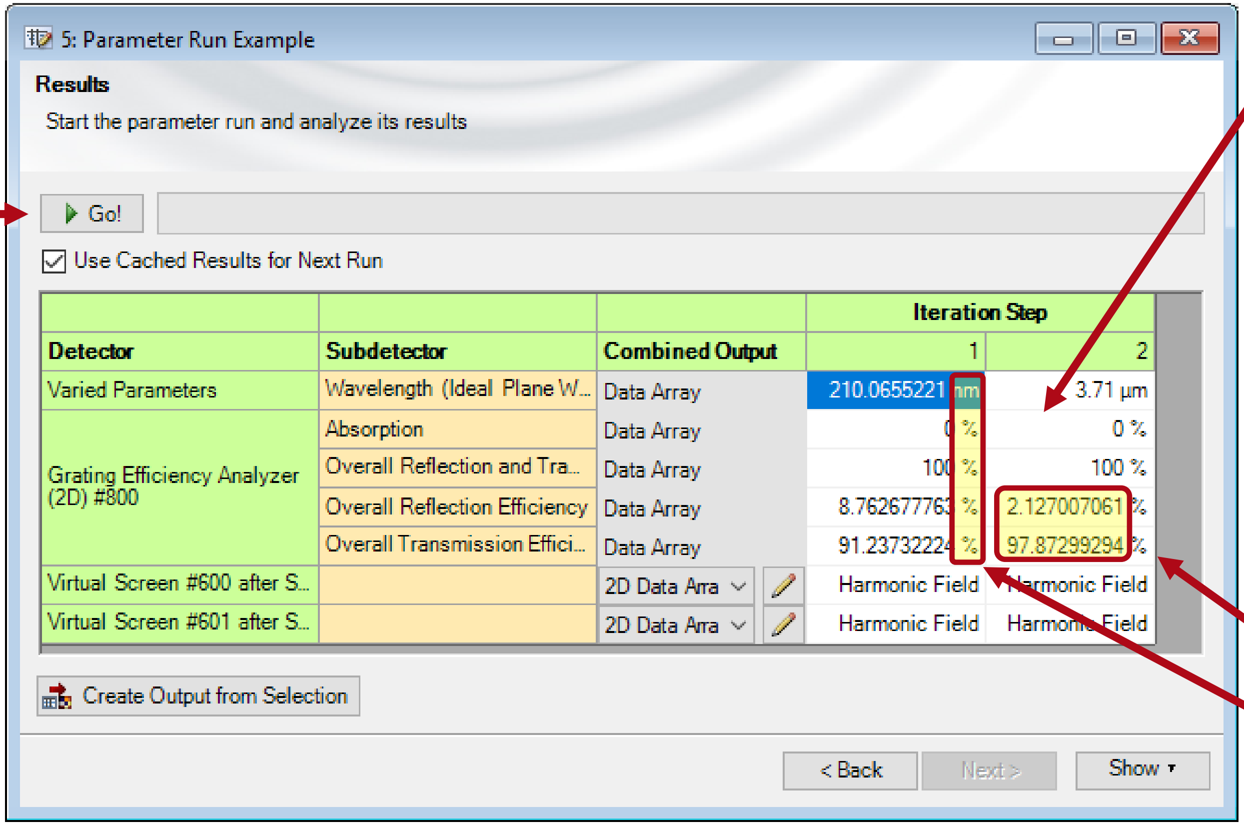
# Detecting Devices Specification Page



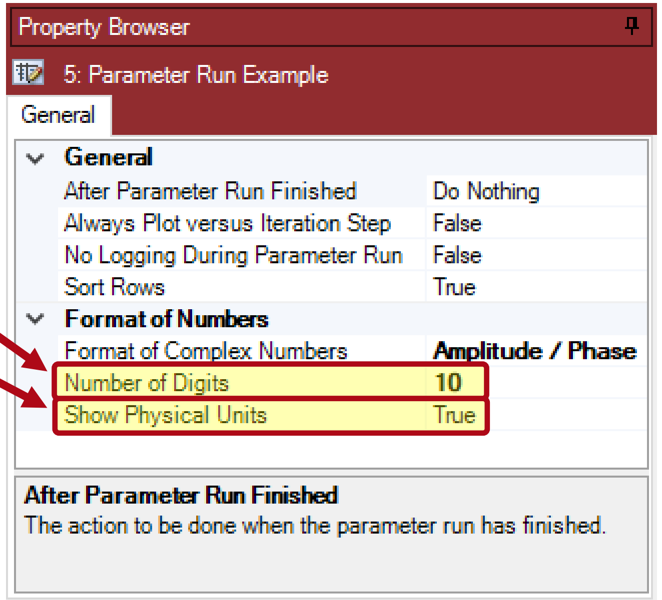

- This page allows to select which simulation engines, detectors, screens and analyzers are evaluated.
- The detecting devices can be configured after clicking Open to get to the edit dialog.

# Results Page

Starts and stops the parameter variation.

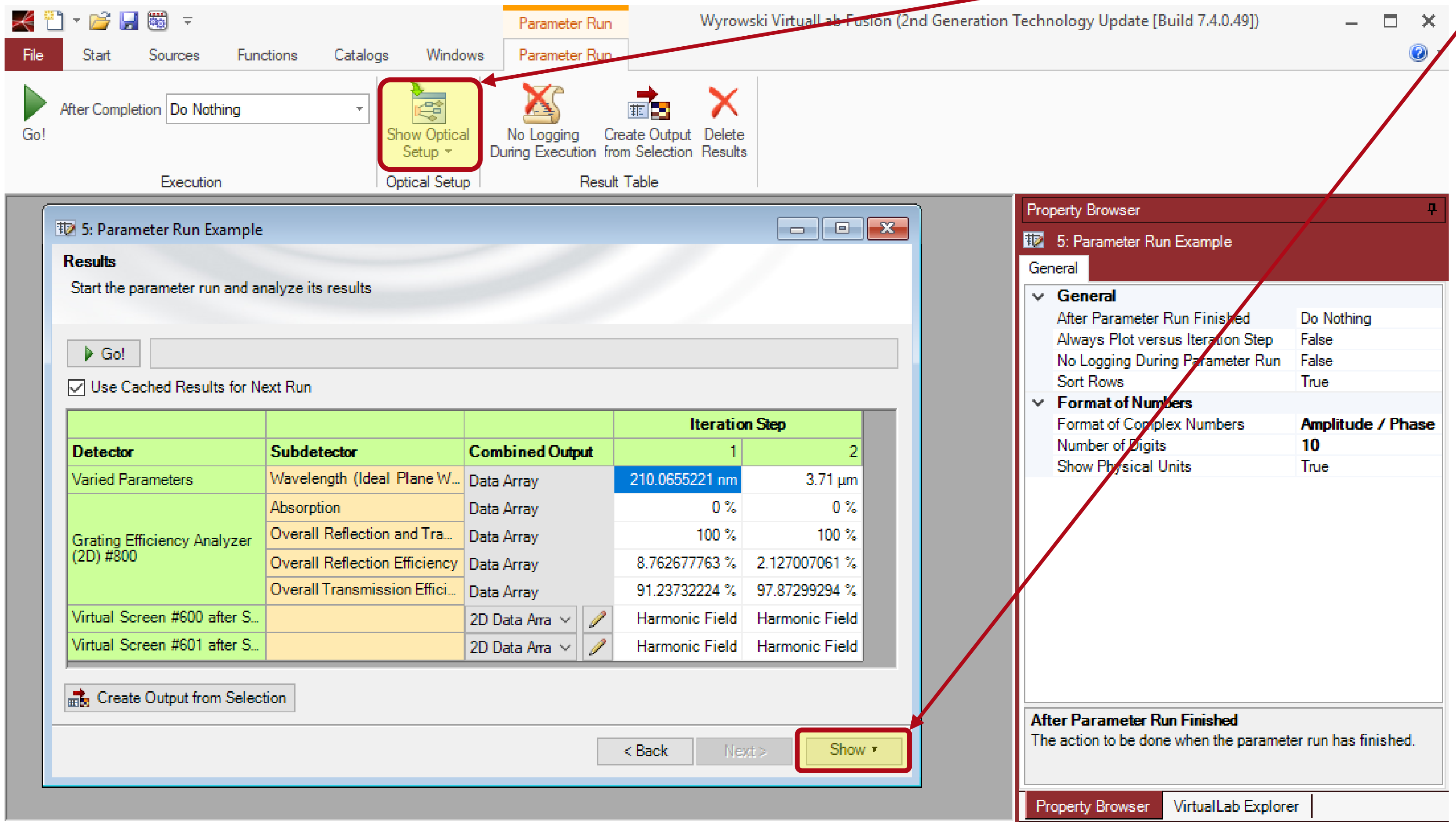



Simulation results: Double click on a document to view it in a separate window.




In the Property Browser you can change the formatting of the shown physical values (number of digits and whether physical units are shown) so that you can better export them to e.g. spread sheet programs via copy & paste.

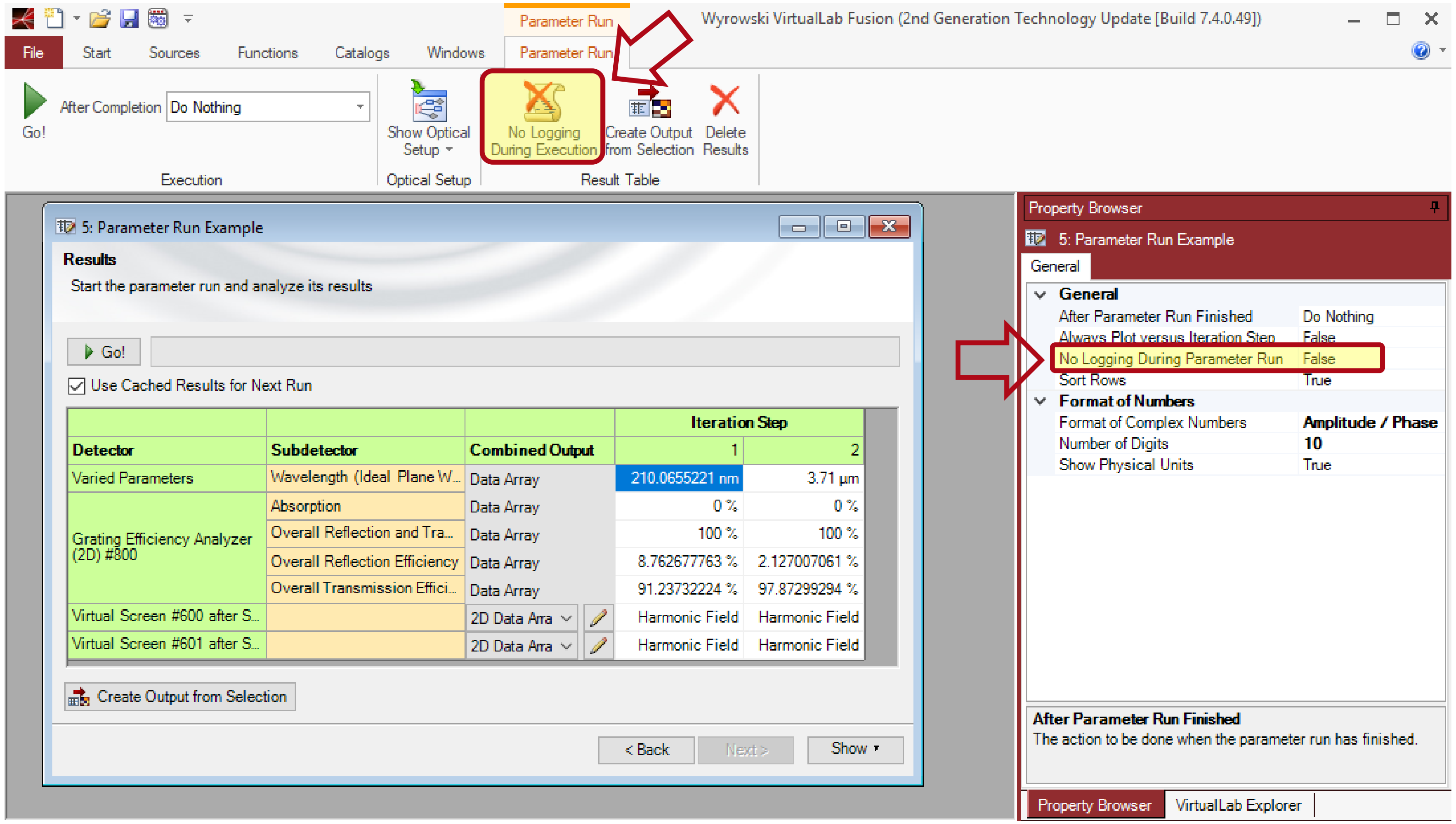
# Optical Setups within Parameter Run




Displays the optical setup:

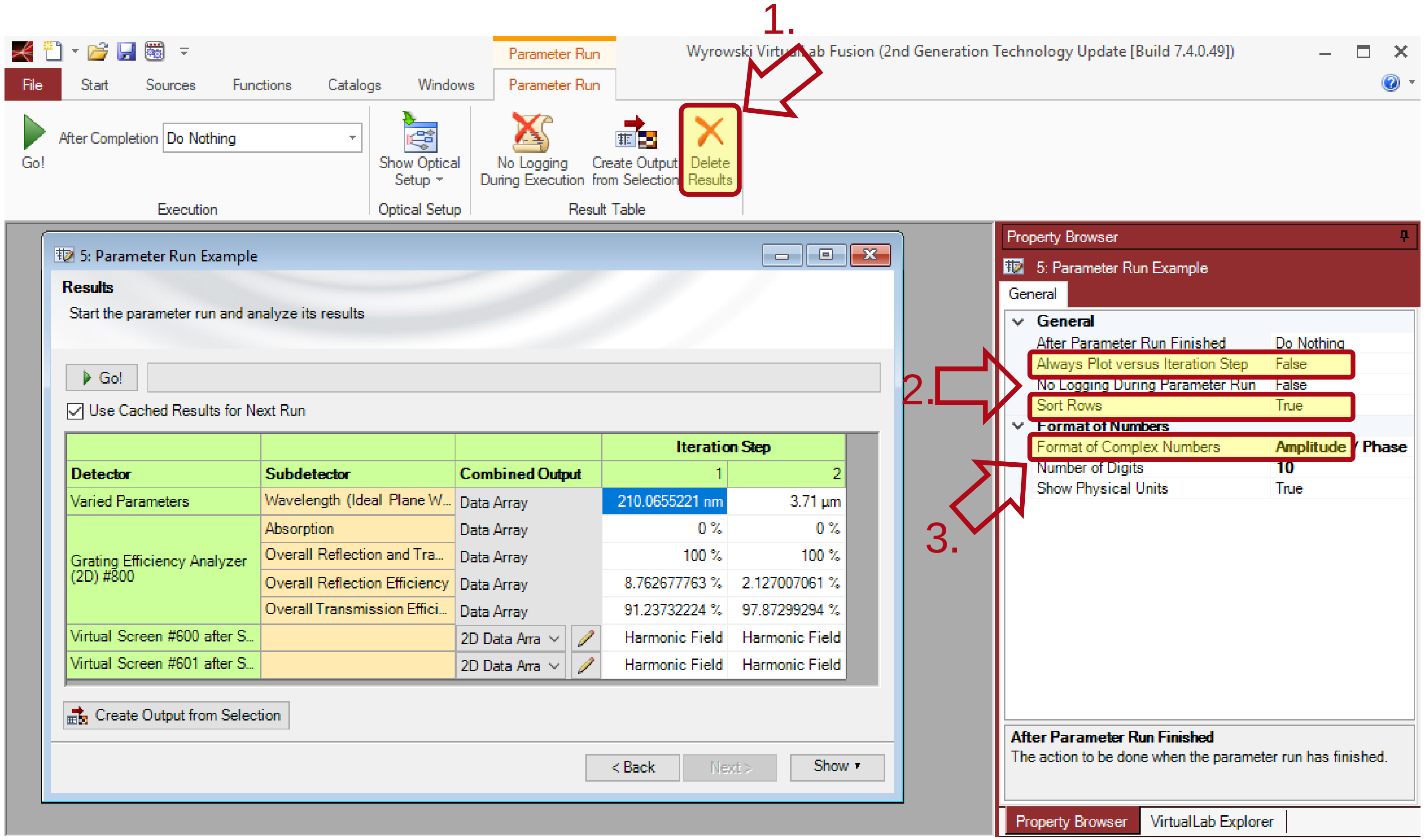
- initial
- from any iteration

# Logging of Parameter Run Results




- For time critical simulations especially for Parameter Runs with many iterations, the simulation time can be reduced by **deactivating the logging**.
- Thus the results are only shown after all iterations are finished.
- In order to see the results of a running Parameter Run document that have been produced so far, you can duplicate the document via the Windows ribbon; then VirtualLab creates a Paramter Run document of the current status with all already calculated results.

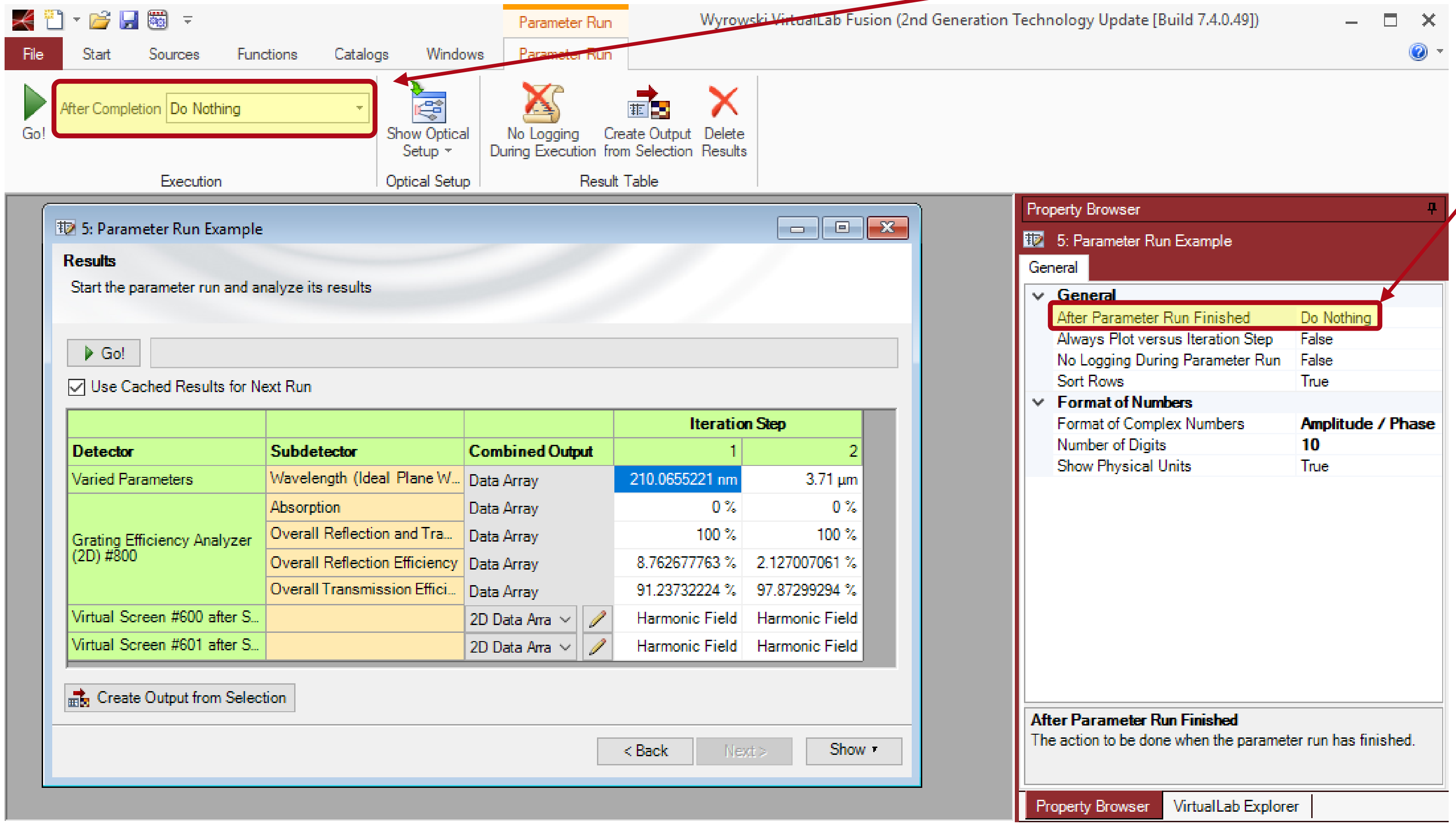
# Display of Parameter Run Results




1. It is possible to delete the results in order to save a smaller Parameter Run document (e.g. for email sending).
   (Sometimes the saving or opening of a Parameter Run document with many and/or huge results takes longer than the simulation of all iterations.)
2. The user can select different orders for the display of the results.
3. There are different options to display compex numbers.

# Saving (& Shutdownd) after Parameter Run Completion?

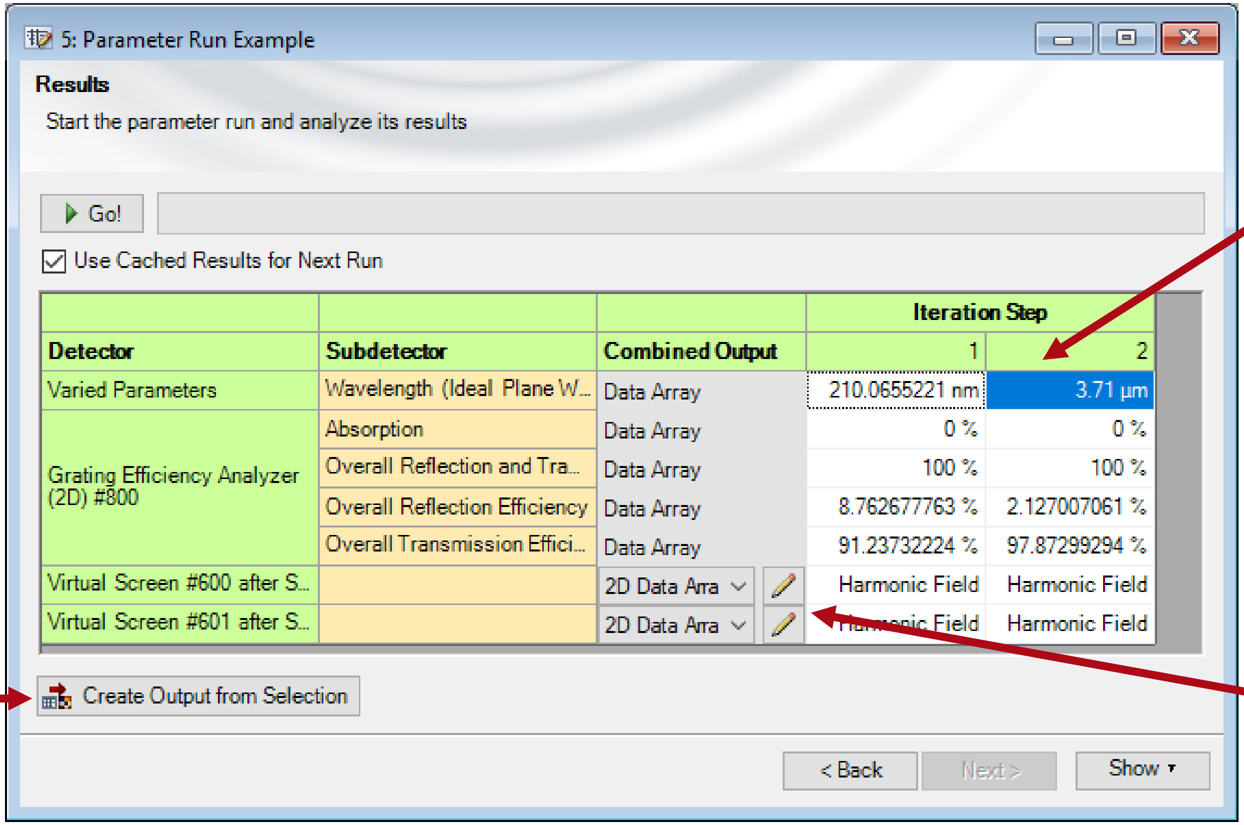
Allows you to save the results after the simulation has finished and then shut down your computer.



# Results Page – Combined Outputs

The results for each (sub-)detector can be combined into a Data Array, Animation, Harmonic Fields Set or Ray Distribution. Which combined outputs are available depends on the type and dimensionality of the original documents.

Create the combined output – or stop the creation if it takes too long. Clicking/Double clicking on a cell in the Detector or Subdetector column is a shortcut to selecting the whole row and start the output creation with the current combined output.



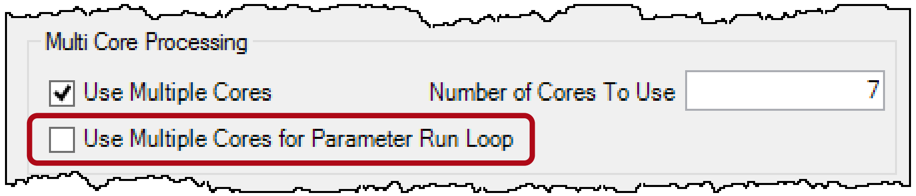

- Select the results to combine.
- Clicking on a cell in the Detector or Subdetector column selects the whole row.

- Choose the desired combined output.
- Several combined outputs can be configured by clicking on the pencil icon.

# Parallelization & Amount of Data

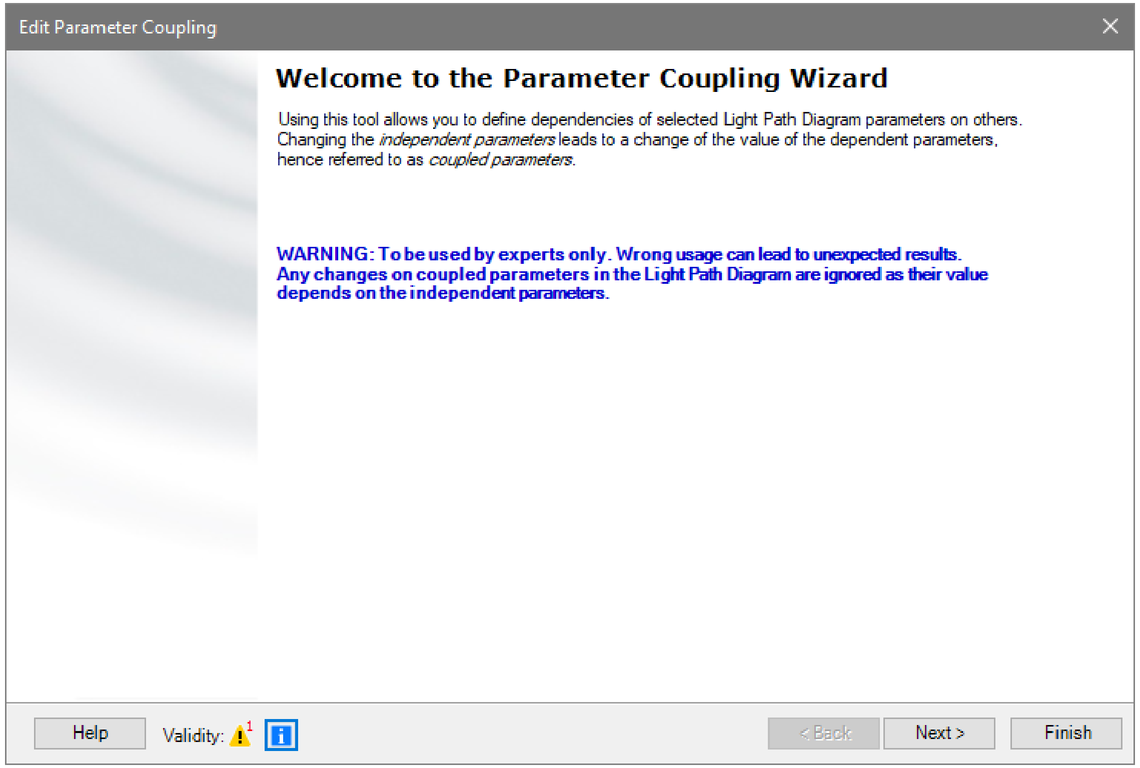
- The execution of the different iterations of a Parameter Run simulation is very well parallelized. Thus it represents a very efficient method to simulate many different settings very fast.

- But in case already one simulation is extremely memory consuming, parallel executions are out of the question. They would not be possible or slow down the whole process if VirtualLab may swap such large data on hard disc instead of keeping it in the RAM.
- Then the parallelization should be switched off for Parameter Run Loop.
- VirtualLab will still do parallel computations, as parallelization is also used within single system simulations.



# Parameter Coupling

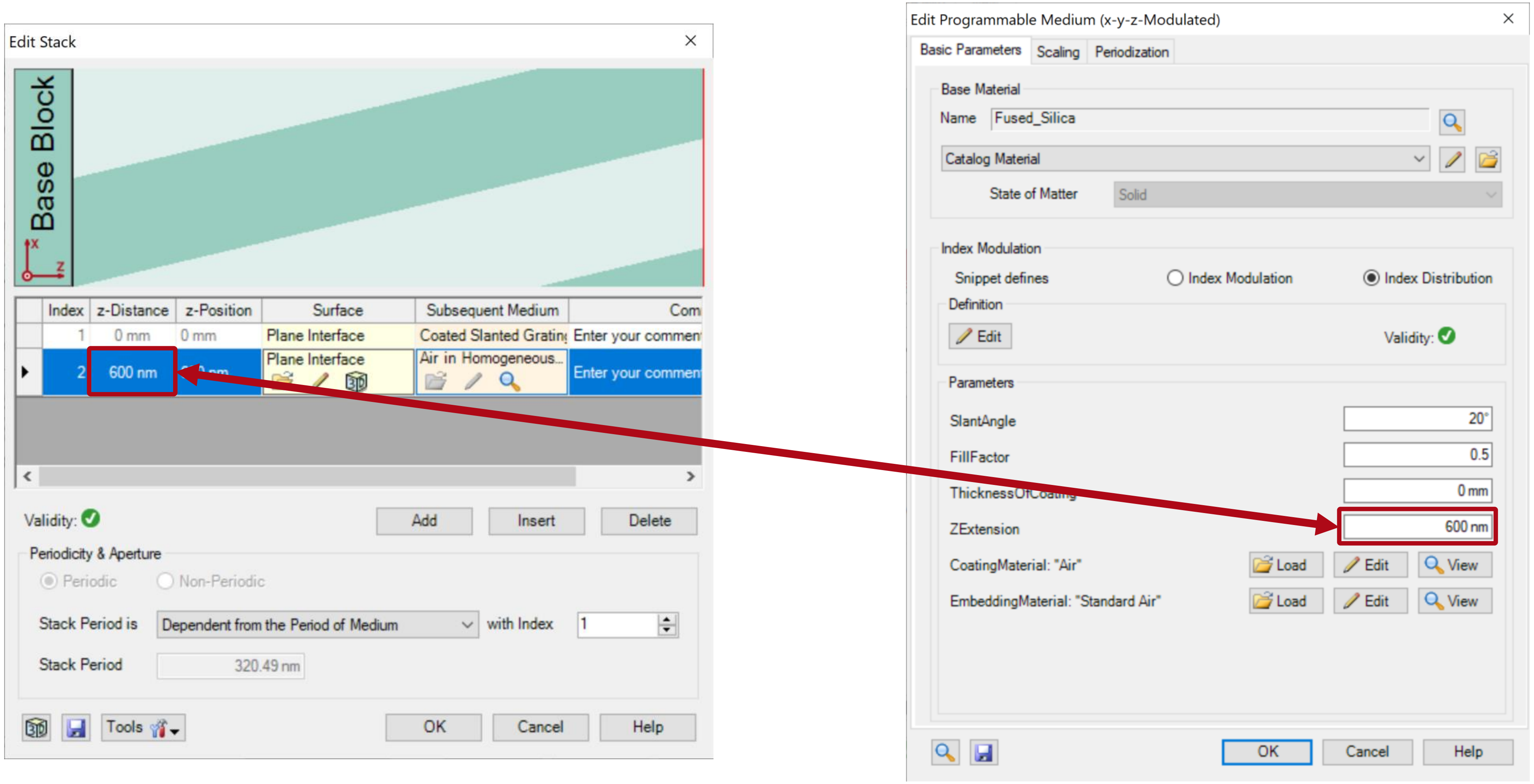
# Abstract



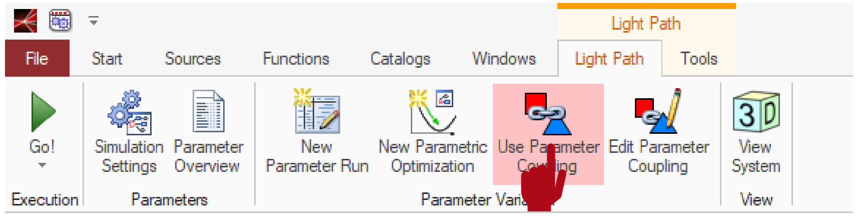

The parameter coupling feature of VirtualLab Fusion enables the coupling of parameters in an optical setup. The values can also be used to re-calculate other parameters of the system, so that a certain relationship between them is automatically maintained. Hence, this feature allows the user to instate complex dependencies for these parameters. For instance, in this example we use the Parameter Coupling to ensure that the z extension of a user-programmed slanted grating medium coincides with the thickness of the structured layer where it is contained.

# Task

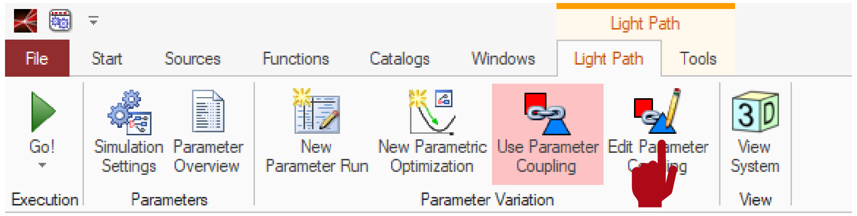
We wish to link two parameters of an optical system, so that they automatically take the same value. For this purpose, VirtualLab's Parameter Coupling feature is used.



# Set Up Parameter Coupling

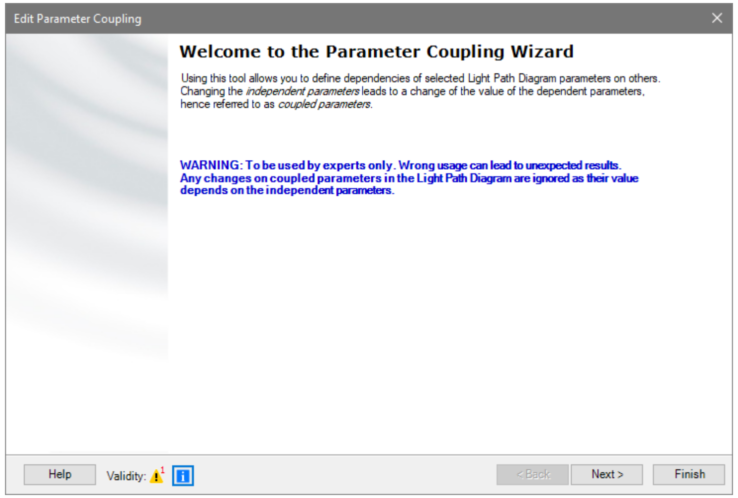



In order to use the parameter coupling feature of VirtualLab Fusion activate the option "*Use Parameter Coupling*" for the optical setup in question.




Afterwards, the "*Edit Parameter Coupling*" button is available.

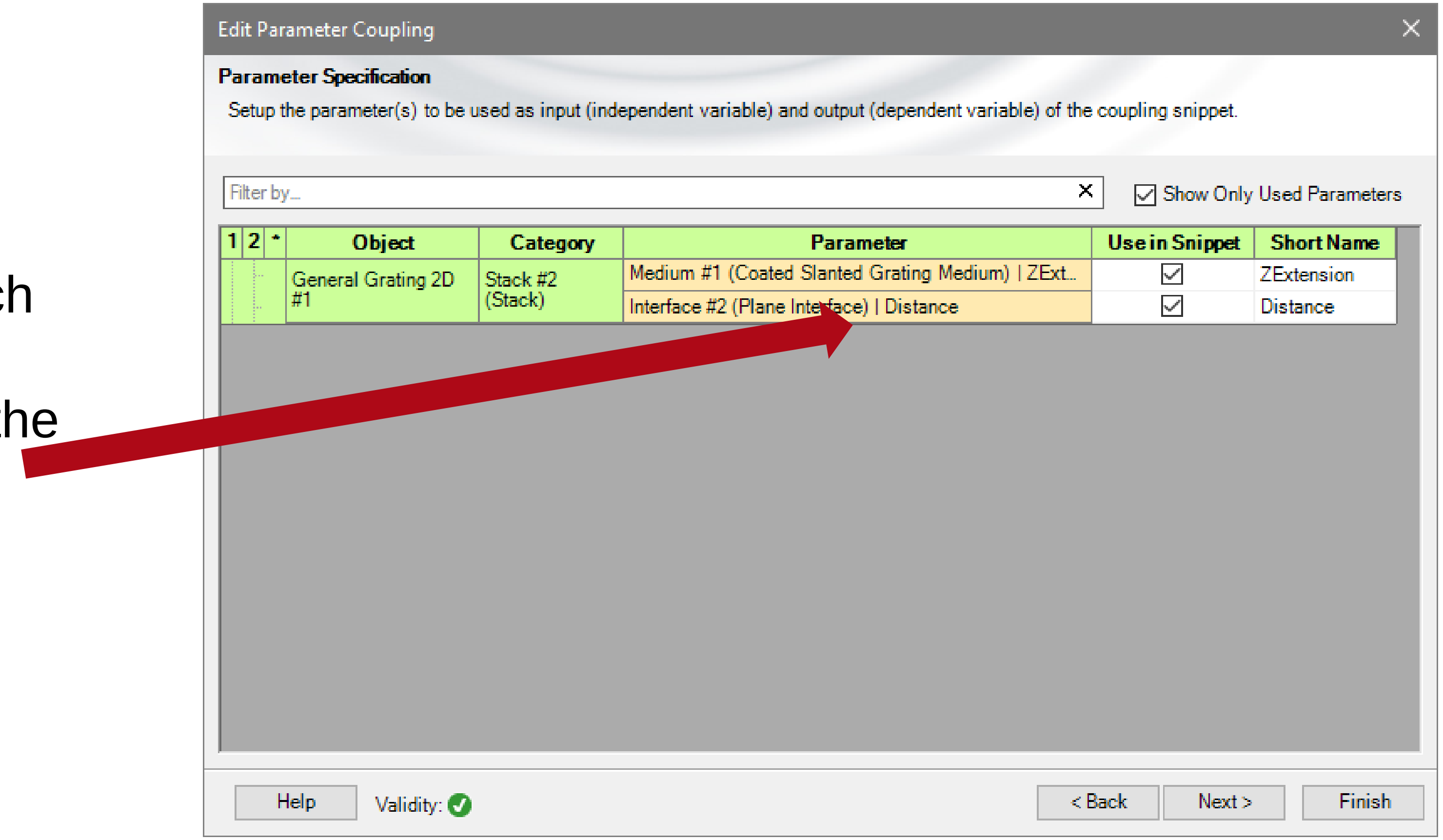
Clicking on the "*Edit Parameter Coupling*" button causes the parameter coupling wizard to appear.



# Choose Parameters Involved

By clicking “*Next*”, a table is shown which contains all parameters of the current optical setup.

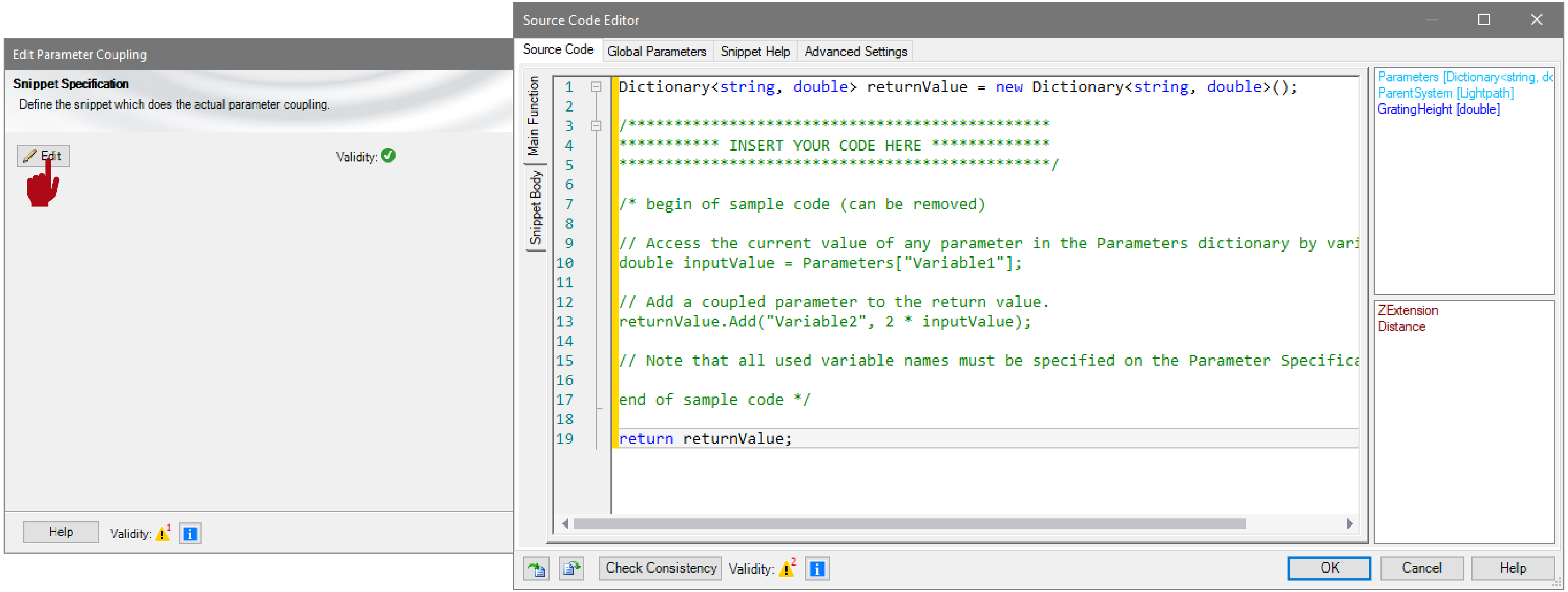
Please select all the parameters which are relevant for the coupling and necessary calculation. For instance, the parameters “*ZExtension*” and “*Distance*” are chosen in this case.



# Configure the Coupling of the Parameters

After choosing the parameters, the snippet which controls the coupling has to be set.

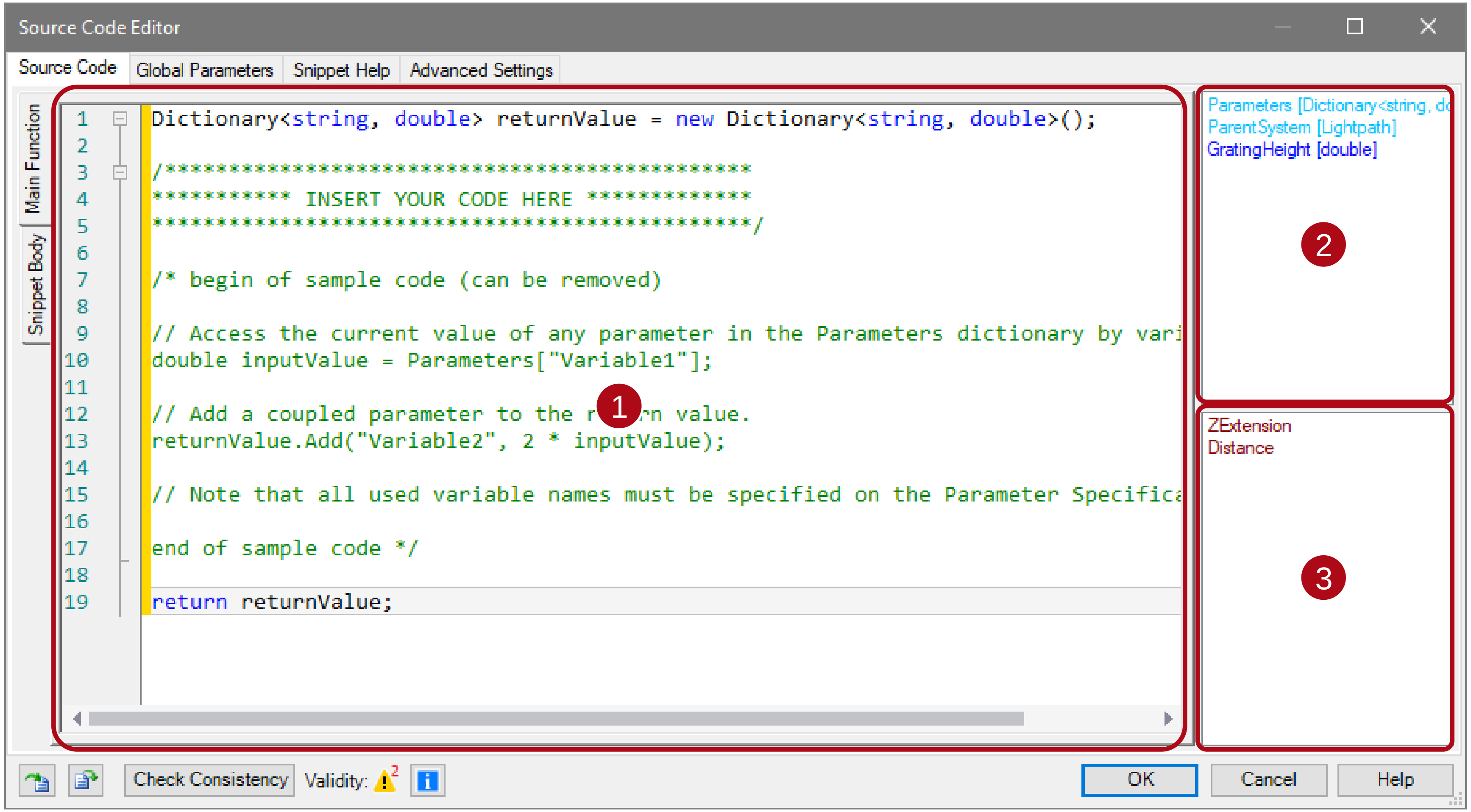
By clicking on "*Edit*" the source code editor opens.



# Configure the Coupling of Parameters

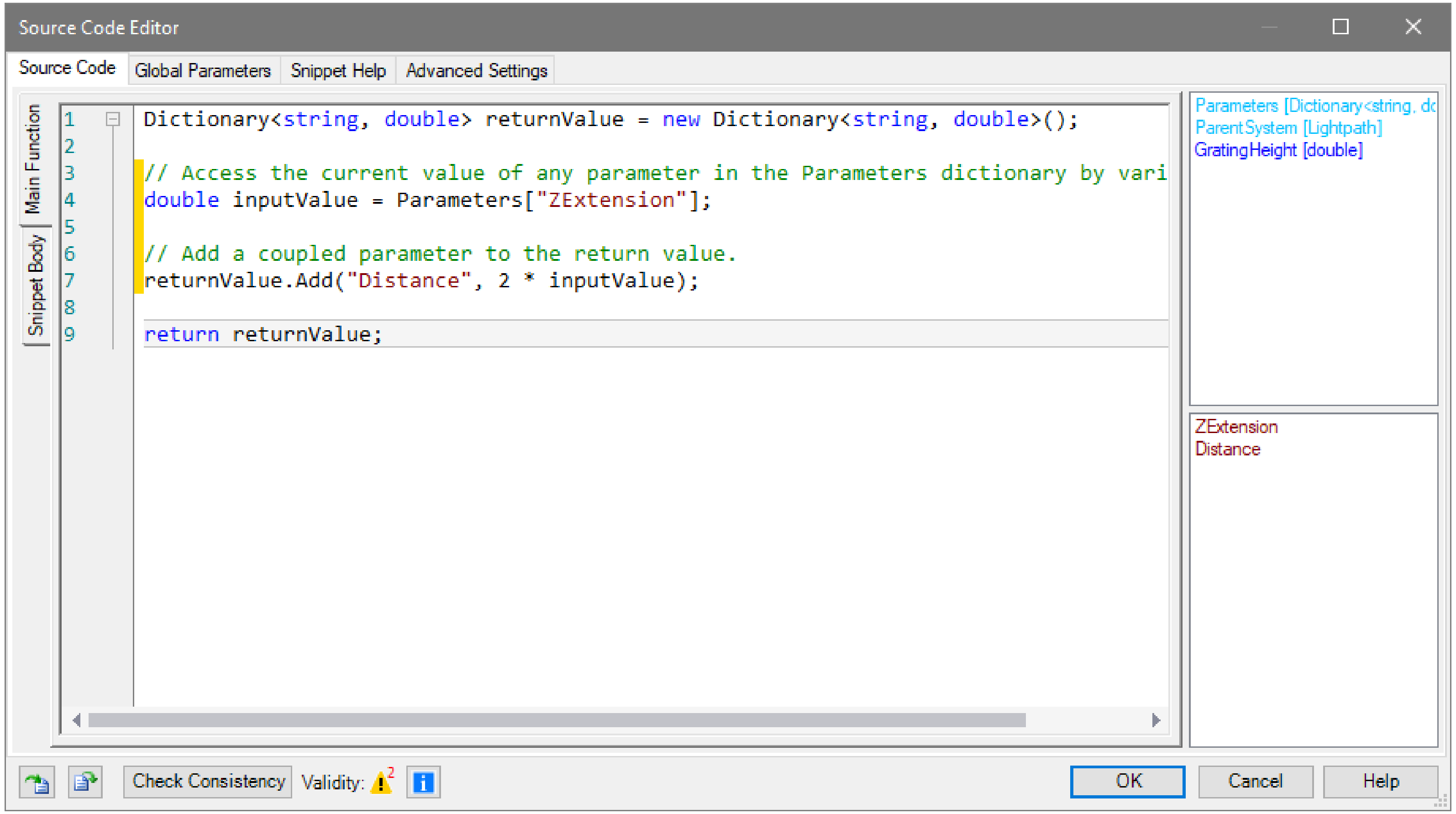
The source code tab contains three areas:

1. the source code (center area)
2. global variables/parameters (upper right area)
3. chosen system parameters (lower right)



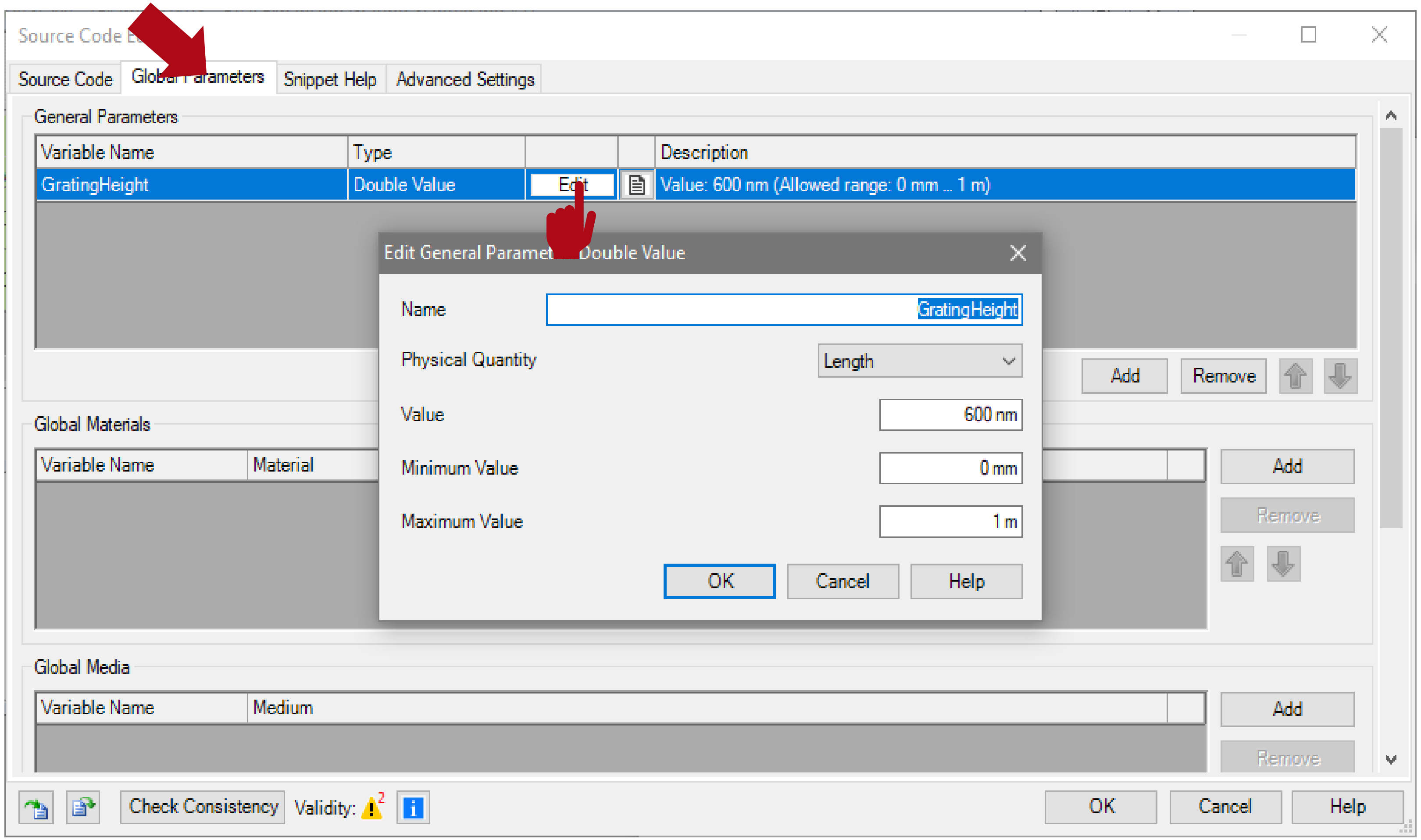
# General Example of Parameter Coupling

- In general, the chosen parameters have to be read from the dictionary and saved to a variable (line 4).
- Afterwards, that value can be used as output for another parameter, or play a role in its calculation, e.g. be doubled (line 7).



# Definition of Global Parameters

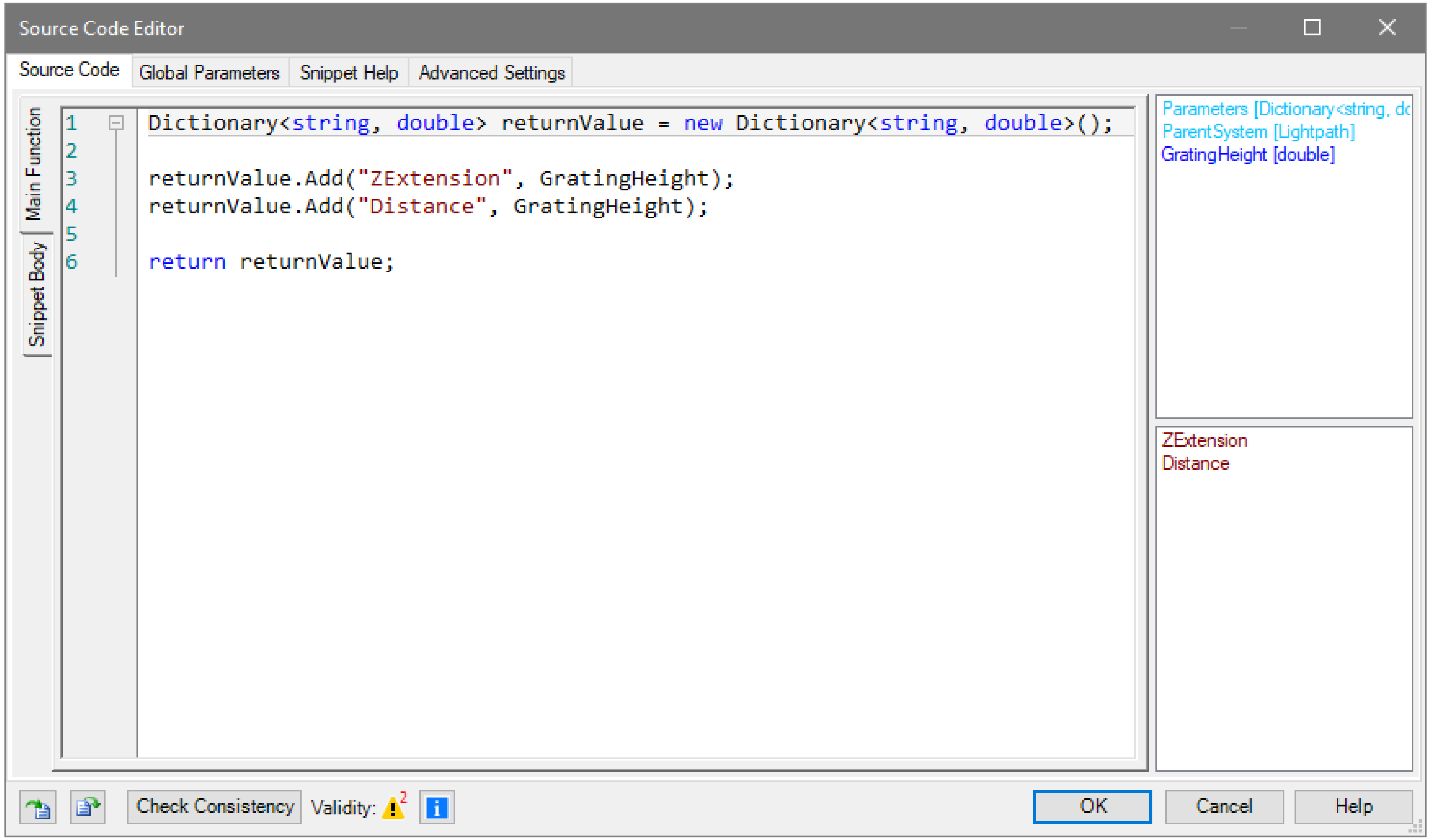
- In this particular example, it is helpful to define a new global variable, which later appears on the parameter coupling window.
- This can be done in the "*Global Parameters*" tab.
- The variable can be of different types and have different physical quantities attached.



# Particular Example of Parameter Coupling

- In this example, the global variable is used to return its value to both chosen parameters of the system.

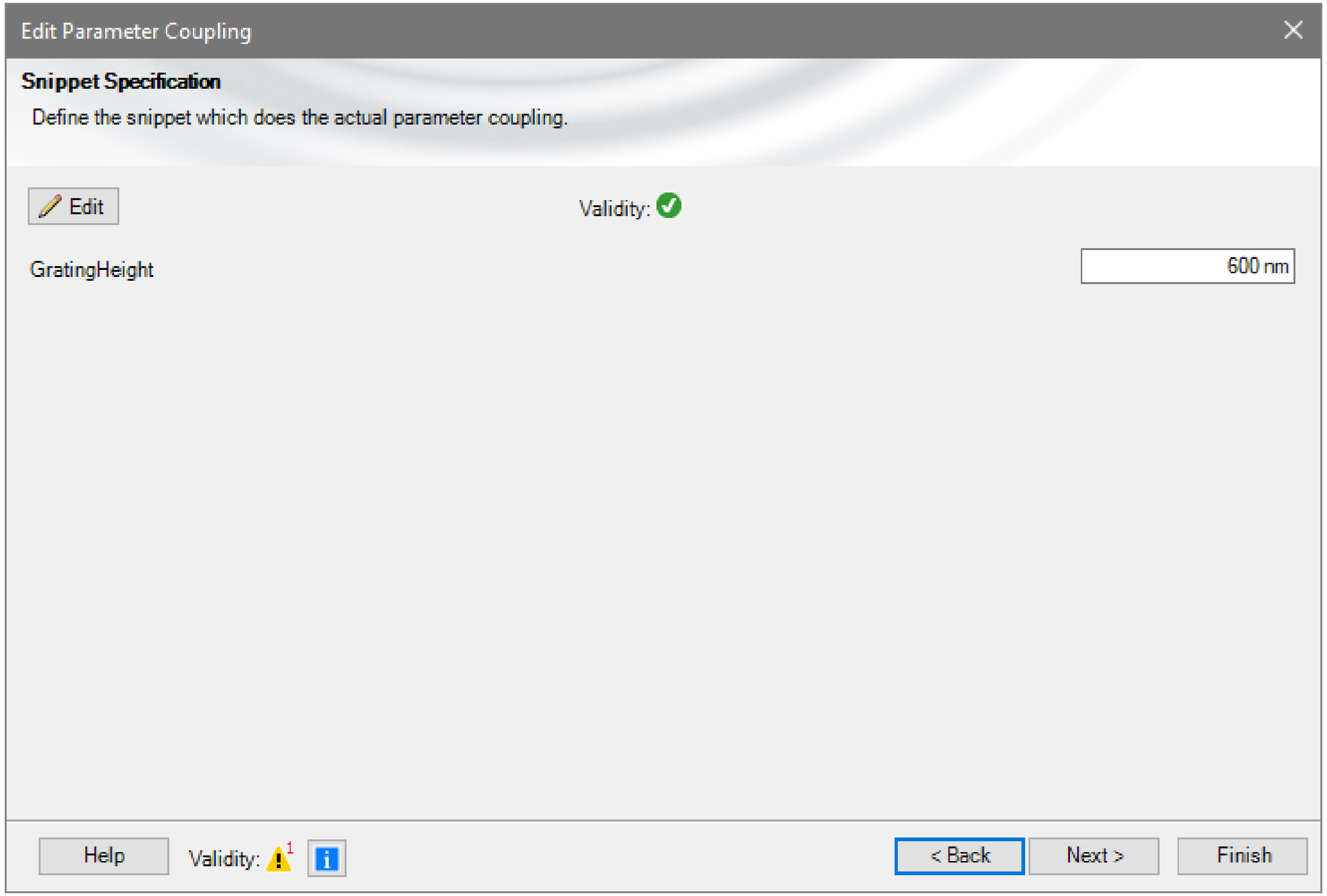
- Thus, no parameter has to be read from the dictionary or re-calculated.



# Particular Example of Parameter Coupling

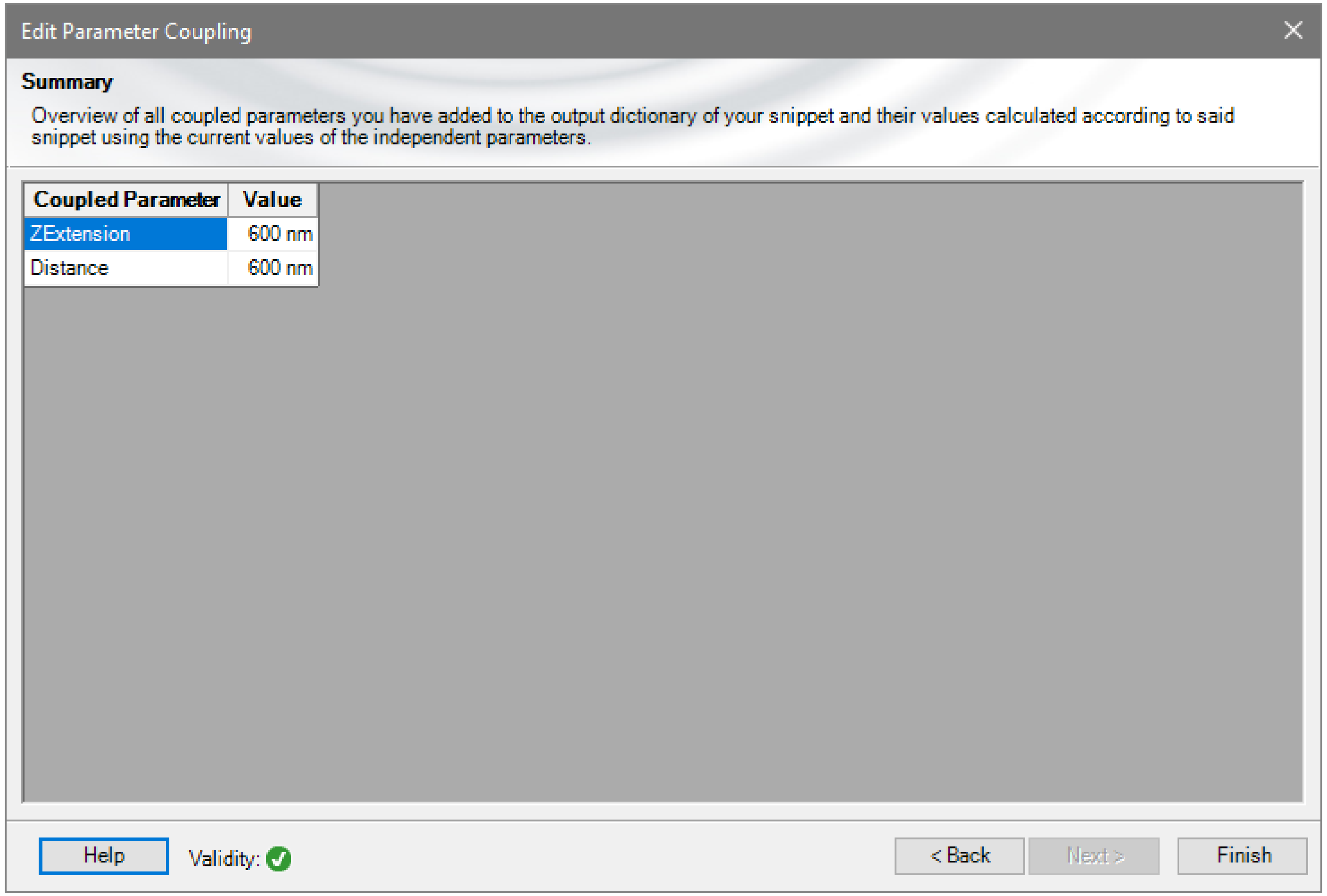
- After closing the source code editor, the defined global variable “*GratingHeight*” appears.

- When working with the system later on, the user will only be able to modify the value of this variable, which will in turn automatically affect the value of the system parameters. Trying to modify the value of the parameters themselves will have no effect.



# Final Check of the Set-up Parameter Coupling

- On the last page of the wizard, the returned parameters and values can be checked.



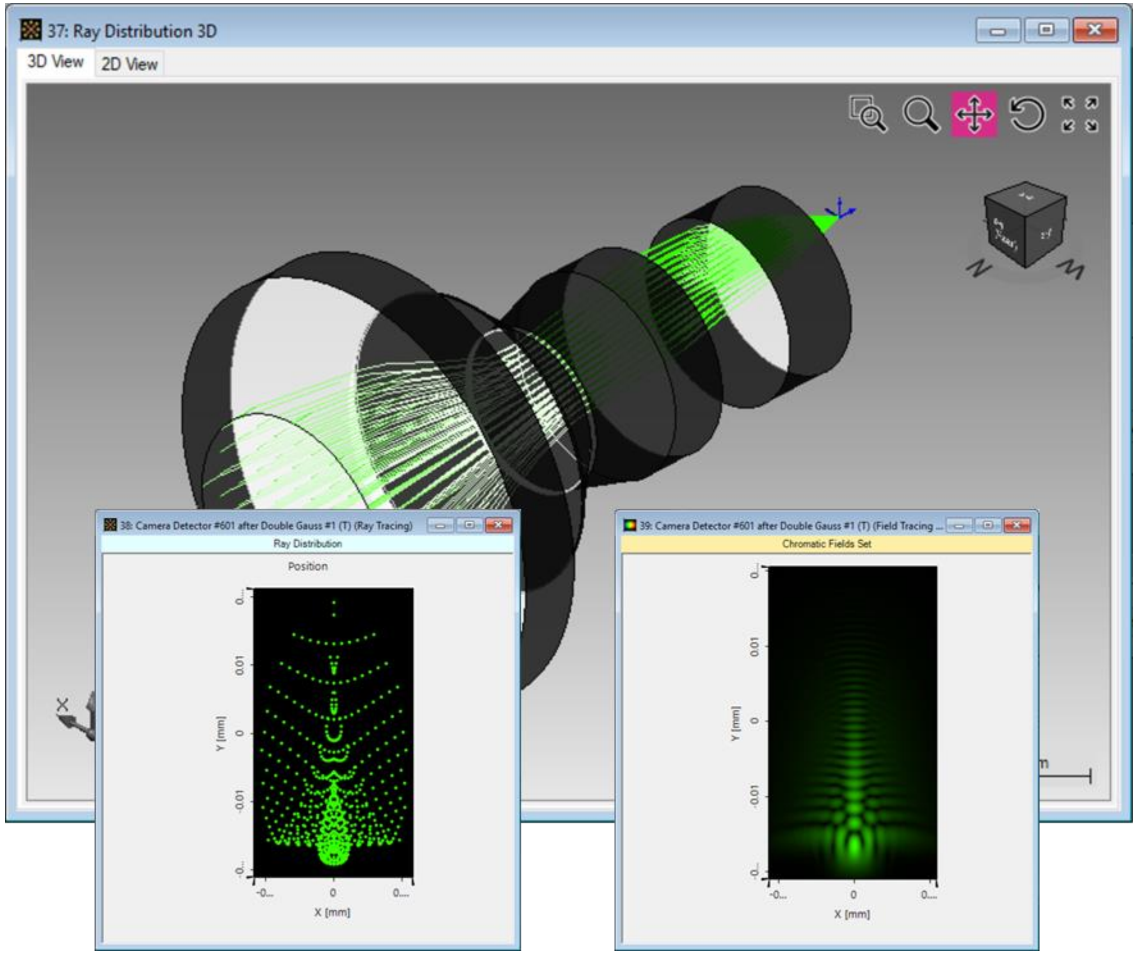
# Zemax Import

# Importing Zemax Files

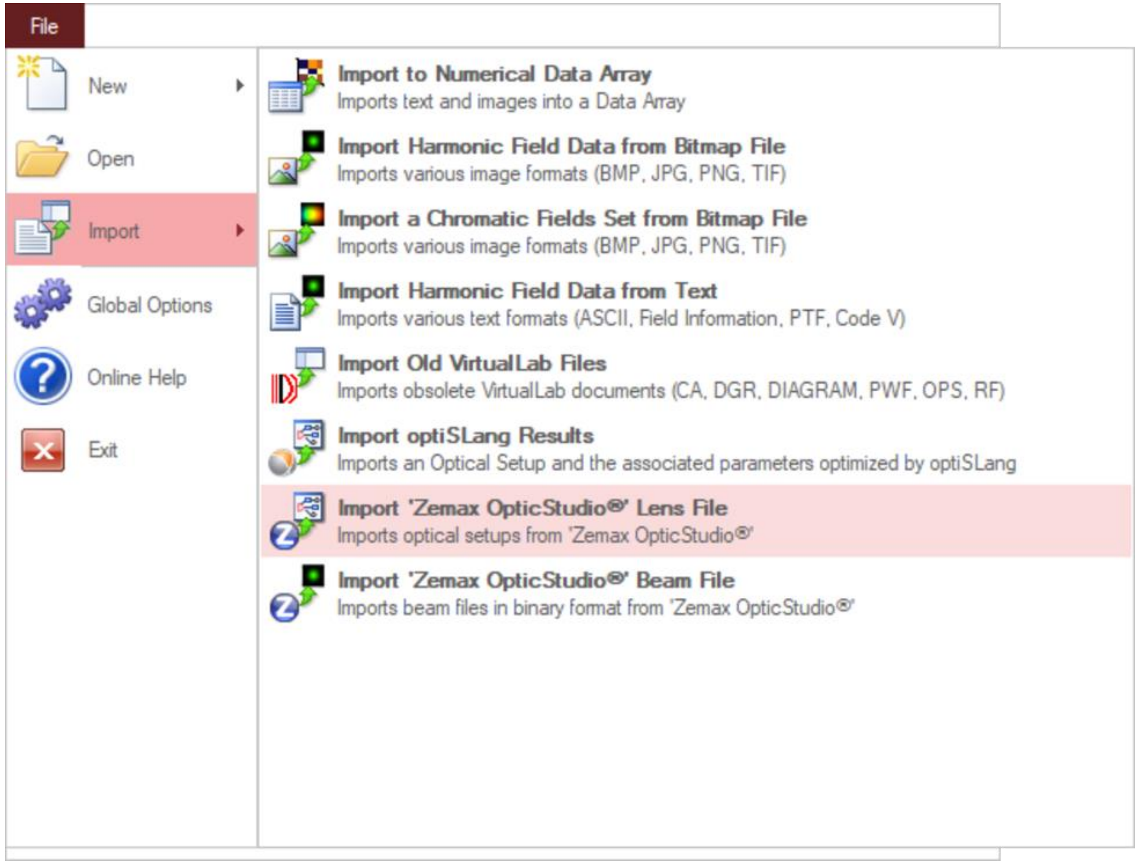
Being able to import Zemax files into VirtualLab constitutes an interesting feature for several reasons:

- Using field tracing in VirtualLab to simulate a system previously constructed in OpticStudio

- Some hardware manufacturers provide only Zemax files of their products



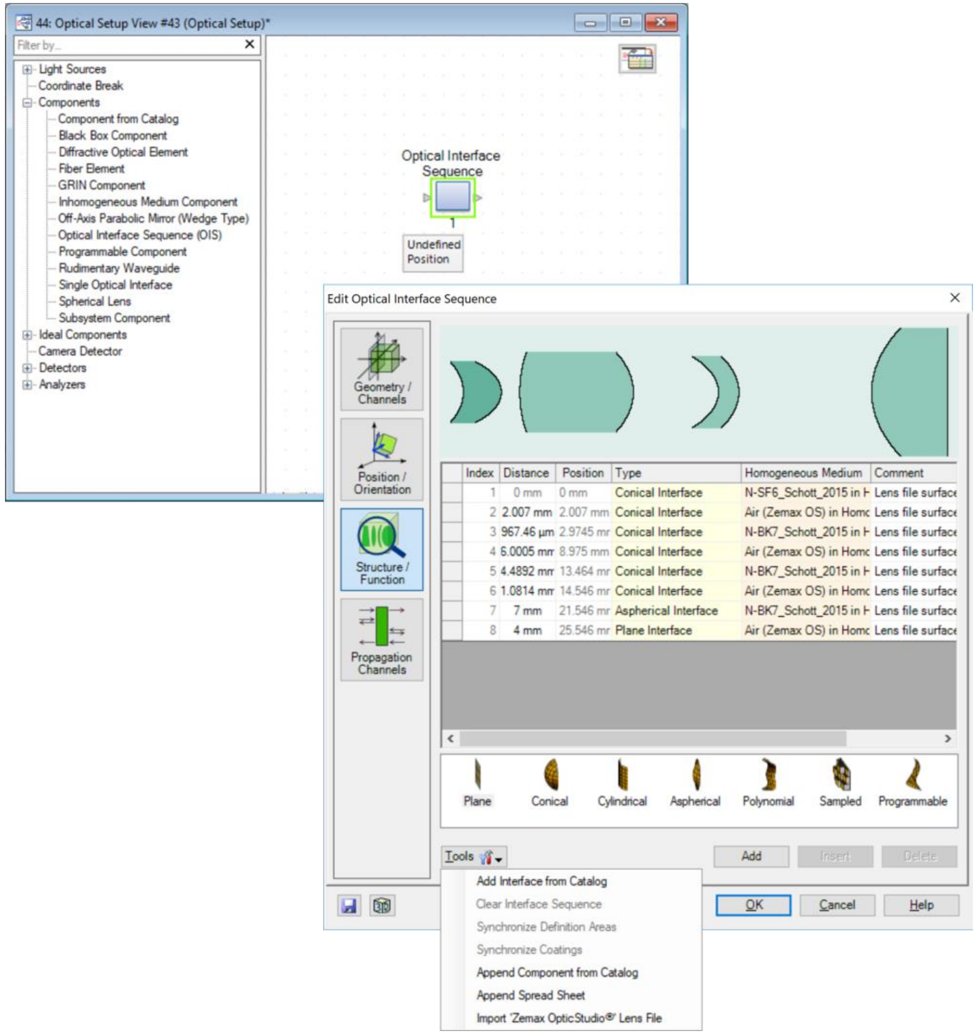
# Importing Zemax Files




There are two ways to go about importing a Zemax OpticStudio file into VirtualLab Fusion:

- Importing the full system file into VirtualLab Fusion

# Importing Zemax Files




There are two ways to go about importing a Zemax OpticStudio file into VirtualLab Fusion:

- Importing the full system file into VirtualLab Fusion
- Importing the system into an Optical Interface Sequence component in VirtualLab Fusion

# Two Zemax Import Mechanisms: Elementary and Advanced

In general the user has three workflow options to trigger the import:

- drag and drop a ZMX file into your VirtualLab window
- go via File menu → Import → Import Zemax OpticStudio Lens File
- use the Tools button in the configuration dialogue of the Optical Interface Sequence (OIS) component into which the system is to be imported

Which import mechanism is used depends only on whether Zemax is installed on the computer in question and on whether there is a valid running licence or not:

- The advanced import mechanism needs access to the Zemax installation to extract more detailed information of the system which is to be imported
- The elementary import mechanism needs only the Zemax glass catalogue directory in order to load the pertinent materials for the system

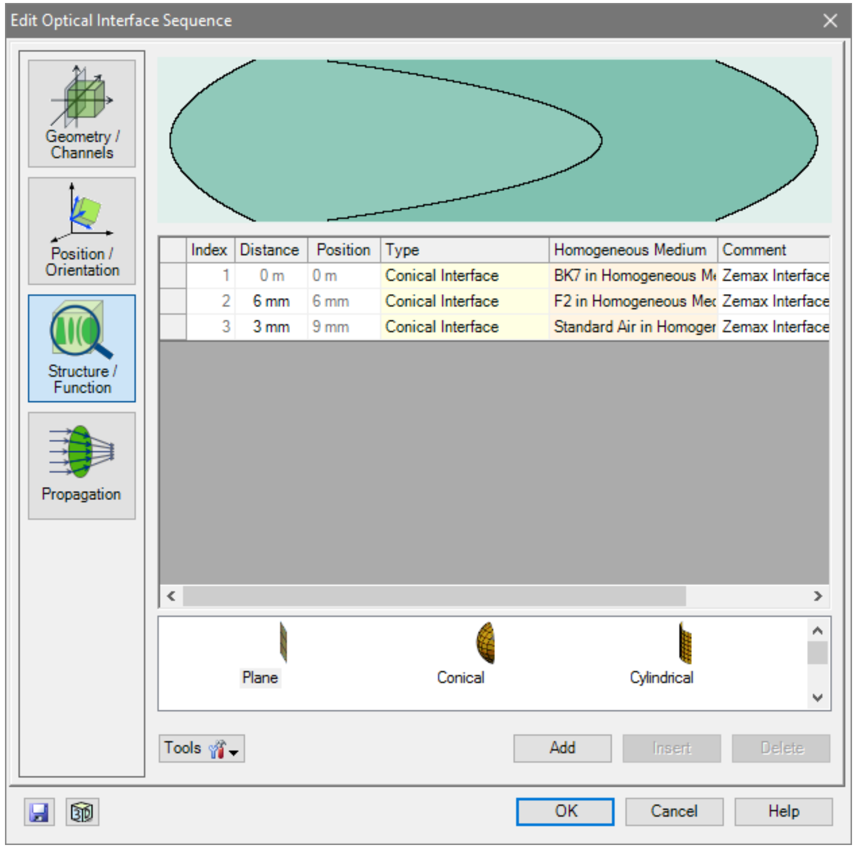
# Elementary Zemax Import

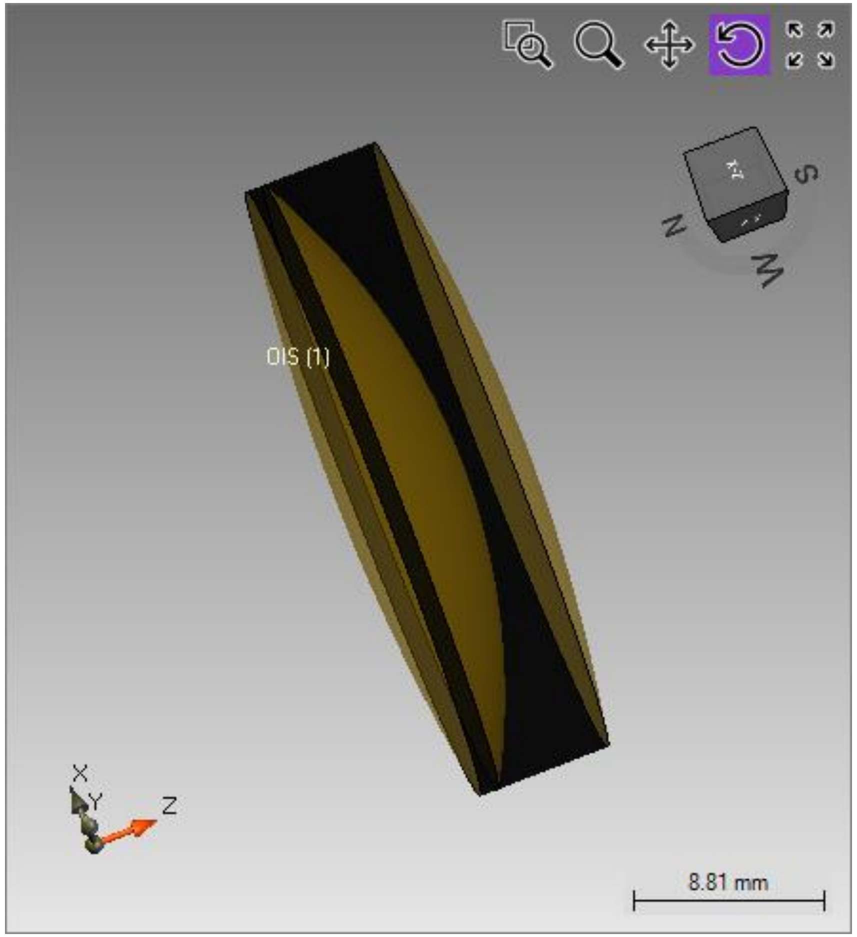
The **elementary Zemax** import is fully **sufficient** for the **import of most lens data** that is provided on manufacturers‘ websited via Zemax files.

Even though manufacturers also start to provide VirtualLab files, Zemax files are still a standard.

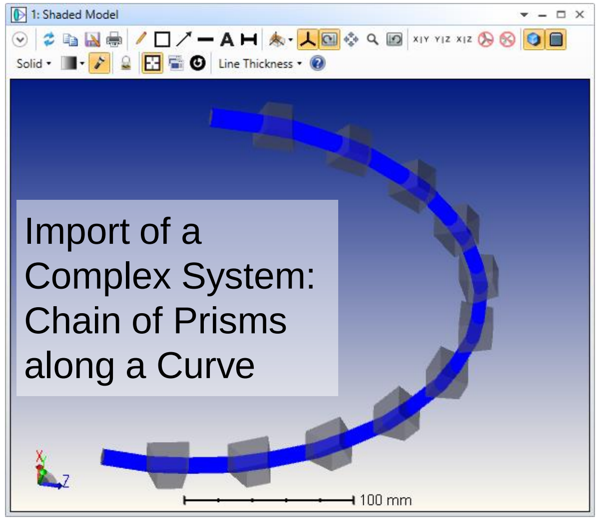
# Elementary Zemax Import

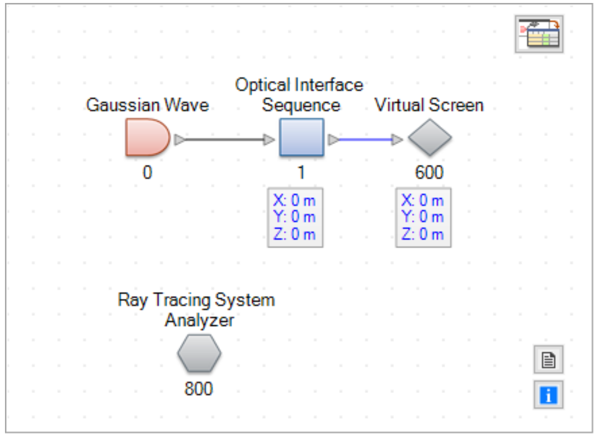
After the import some manual adaptions are needed, then the result looks like the following:

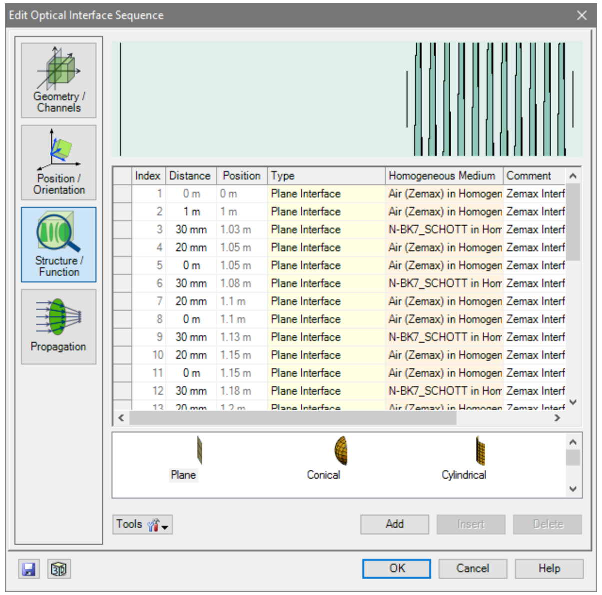


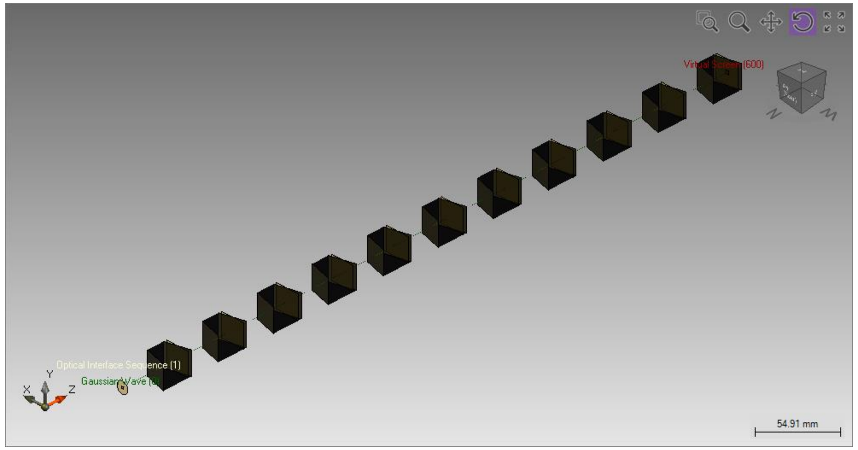
# Limits of the Elementary Zemax Import



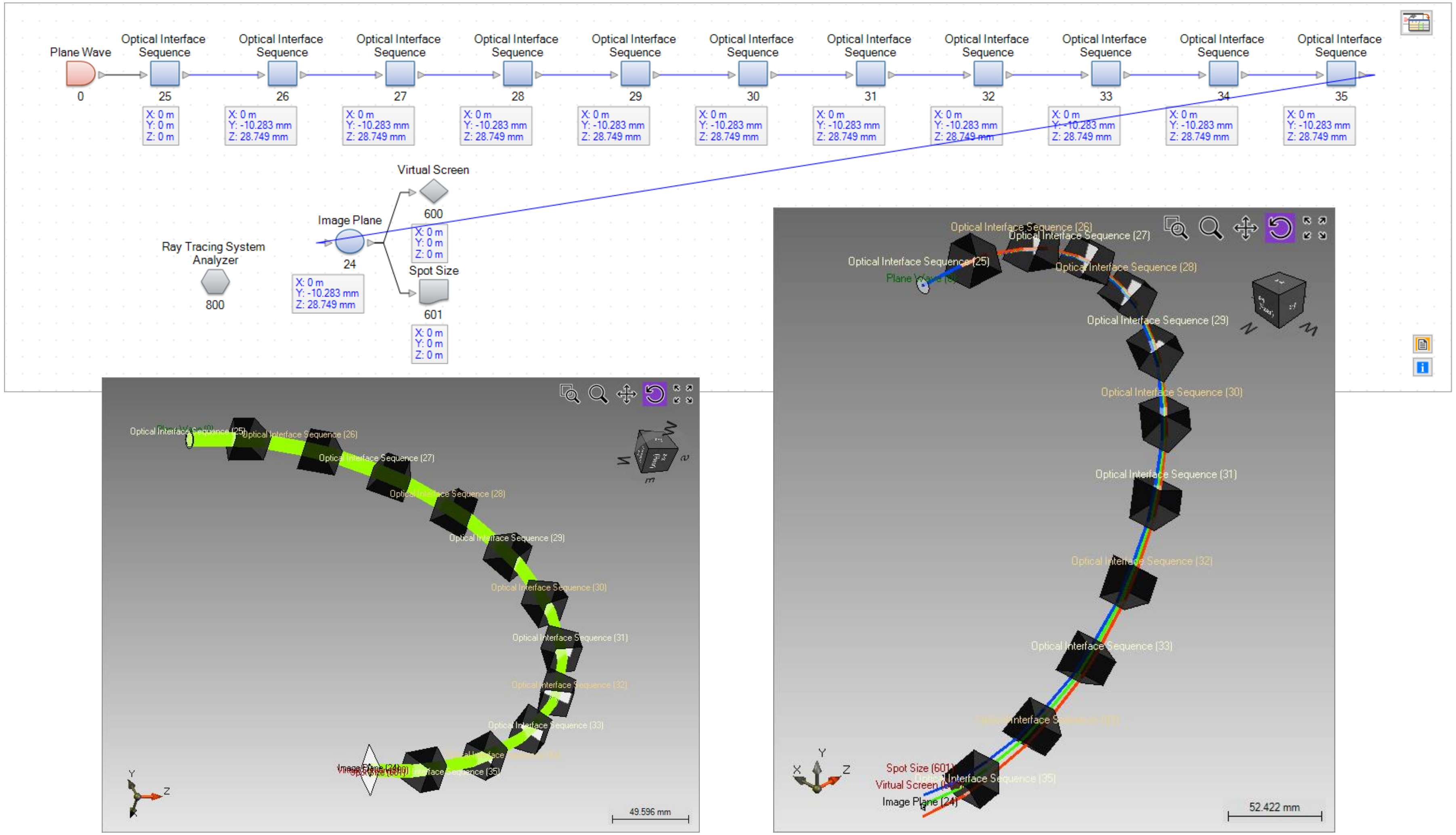




Messages
[08/09/2016 17:55:07] === Starting Zemax import ===
[08/09/2016 17:55:07] Warning: DISZ INFINITY detected at Zemax interface 0
[08/09/2016 17:55:07] Warning: Aperture diameter ZERO detected at Zemax interface 0 (default will be used).
[08/09/2016 17:55:07] Zemax Interface #4: unsupported type COORDBRK, using plane interface
[08/09/2016 17:55:07] Warning: Aperture diameter ZERO detected at Zemax interface 4 (default will be used).
[08/09/2016 17:55:07] Zemax Interface #7: unsupported type COORDBRK, using plane interface
[08/09/2016 17:55:07] Warning: Aperture diameter ZERO detected at Zemax interface 7 (default will be used).
[08/09/2016 17:55:07] Zemax Interface #10: unsupported type COORDBRK, using plane interface
[08/09/2016 17:55:07] Warning: Aperture diameter ZERO detected at Zemax interface 10 (default will be used).
[08/09/2016 17:55:07] Zemax Interface #13: unsupported type COORDBRK, using plane interface
[08/09/2016 17:55:07] Warning: Aperture diameter ZERO detected at Zemax interface 13 (default will be used).
[08/09/2016 17:55:07] Zemax Interface #16: unsupported type COORDBRK, using plane interface
[08/09/2016 17:55:07] Warning: Aperture diameter ZERO detected at Zemax interface 16 (default will be used).
[08/09/2016 17:55:07] Zemax Interface #19: unsupported type COORDBRK, using plane interface
[08/09/2016 17:55:07] Warning: Aperture diameter ZERO detected at Zemax interface 19 (default will be used).
[08/09/2016 17:55:07] Zemax Interface #22: unsupported type COORDBRK, using plane interface
[08/09/2016 17:55:07] Warning: Aperture diameter ZERO detected at Zemax interface 22 (default will be used).
[08/09/2016 17:55:07] Zemax Interface #25: unsupported type COORDBRK, using plane interface
[08/09/2016 17:55:07] Warning: Aperture diameter ZERO detected at Zemax interface 25 (default will be used).
[08/09/2016 17:55:07] Zemax Interface #28: unsupported type COORDBRK, using plane interface
[08/09/2016 17:55:07] Warning: Aperture diameter ZERO detected at Zemax interface 28 (default will be used).
[08/09/2016 17:55:07] Zemax Interface #31: unsupported type COORDBRK, using plane interface
[08/09/2016 17:55:07] Warning: Aperture diameter ZERO detected at Zemax interface 31 (default will be used).
[08/09/2016 17:55:07] Zemax Interface #34: unsupported type COORDBRK, using plane interface
[08/09/2016 17:55:07] Warning: Aperture diameter ZERO detected at Zemax interface 34 (default will be used).
[08/09/2016 17:55:07] === Zemax import finished ===



# Advanced Zemax Import



# Further Info about Advanced Zemax Import

- Due to the complexity and the differences of both programs (Zemax & VirtualLab) even the advanced import algorithm needs to make certain assumptions and has some limitations.
- **Example assumptions:**
  - The used wavelengths of the Zemax file are used to define the spectrum of the VirtualLab source.
  - The entrance pupil diameter of Zemax defines the input field size of the source in VirtualLab.
- **Example limitation:**
  Lateral positions and angles of the fields specified in Zemax are ignored. Such configurations need to be reset manually after the import.

For more detailed information please consult the manual or help file.