OPOSSUM v0.6.0 Released

We are thrilled to announce the release of **OPOSSUM v0.6.0**, the latest version of our optical design and simulation software!

# Overview

Over the past five releases, we have developed numerous concepts and features, enabling OPOSSUM to simulate real-life optical models. However, the software currently functions more like an optics library than a complete application, as it lacks a full user interface. File and data formats are also rapidly evolving. Using the software for real projects remains challenging and requires familiarity with the Rust programming language.

If you're willing to test OPOSSUM despite these limitations, we’d greatly appreciate your feedback. Feel free to report bugs and submit feature requests to our software repository.

# Highlights of v0.6.0

## Ghost Focus Analysis

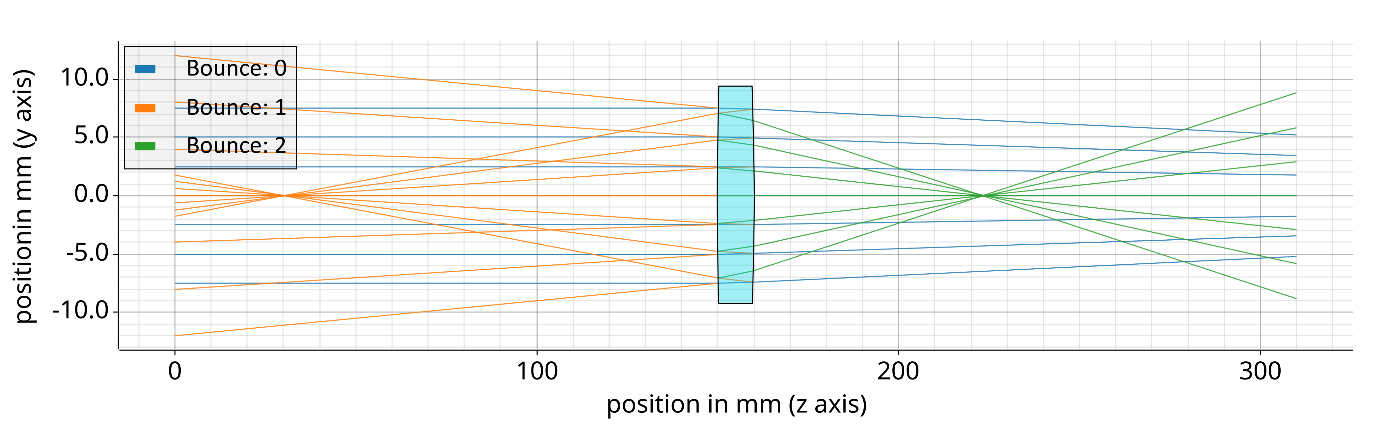
A major focus of this release is the implementation of fully automated **ghost focus analysis** for optical setups. Ghost focus effects, as shown in Fig. 1, are critical in the design of high-power and high-energy laser systems.

Figure : Schematic of a lens creating ghost foci by partially reflecting rays on both of its surfaces

While manual analysis of ghost reflections is possible with other software, it is often tedious—especially for multi-bounce scenarios.

OPOSSUM can now:

* Automatically analyze ghost reflection effects on any optical surface, accounting for a specified number of bounces.
* Assign fluence limits to each optical surface. The analysis report highlights surfaces exceeding these limits and identifies the source of problematic reflections.

This feature helps quickly pinpoint surfaces or geometries needing adjustment to prevent damage in optical systems. Future releases may extend this capability to consider bulk damages and air plasma formation.

A more elaborate example of a ghost focus analysis can be found **here** *(insert link to Announcement OPOSSUM v0\_6\_0 Analysis report.pdf*)

## Extended Fluence Calculation Strategies

The accurate calculation of fluence on optical surfaces is vital for identifying unwanted reflections that could damage components. OPOSSUM now supports multiple strategies for deriving fluence values from ray bundles, including:

* Voronoi density estimator (existing method)
* Simple binning algorithm
* Kernel density estimator (KDE)
* Helper rays defining an area for each main ray

## Basic Handling of Surface Coatings

To simulate ghost focus effects, it’s necessary to account for surface coatings. OPOSSUM now supports:

* Ideal anti-reflective coatings (zero reflectivity)
* Fresnel reflections (uncoated surfaces)
* Constant reflectivity values

Support for reflectivity curves and true coating simulations is planned for future releases.

## Enhanced Source Ray Distributions

Efficient ray tracing depends on thoughtful initial ray distribution. OPOSSUM now offers:

* Rectangular ray bundles with generalized Gaussian energy distributions
* Hexagonal distribution (in addition to random, hexapolar, and Sobol positioning strategies)

The hexagonal distribution improves uniformity and is particularly beneficial for fluence calculations.

## Additional Optical Elements

Two new optical elements have been introduced:

### ****Reflective Grating Node****

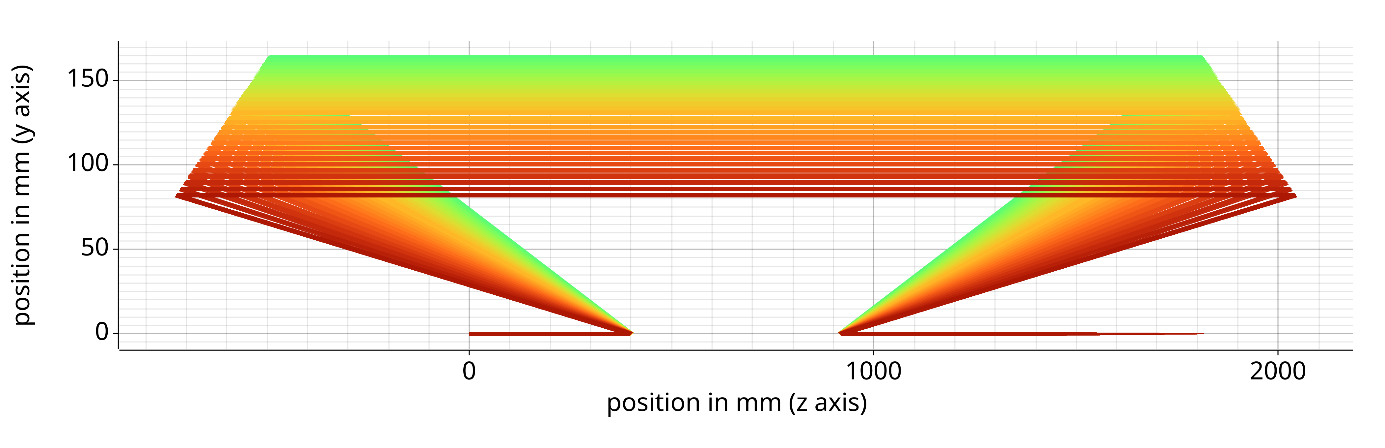
Enables simulations of pulse stretcher/compressor designs, essential for modern CPA laser systems.

Figure 2Example of a modeled Treacy-type compressor with a groove density of 1740 lines/mm

The example shows the ray-tracing through a Treacy-type compressor. To simplify the setup, the gratings can be aligned with respect to an “alignment wavelength” and a specified input- or output angle compared to the Littrow angle of the alignment wavelength.

### ****Parabolic Mirrors****

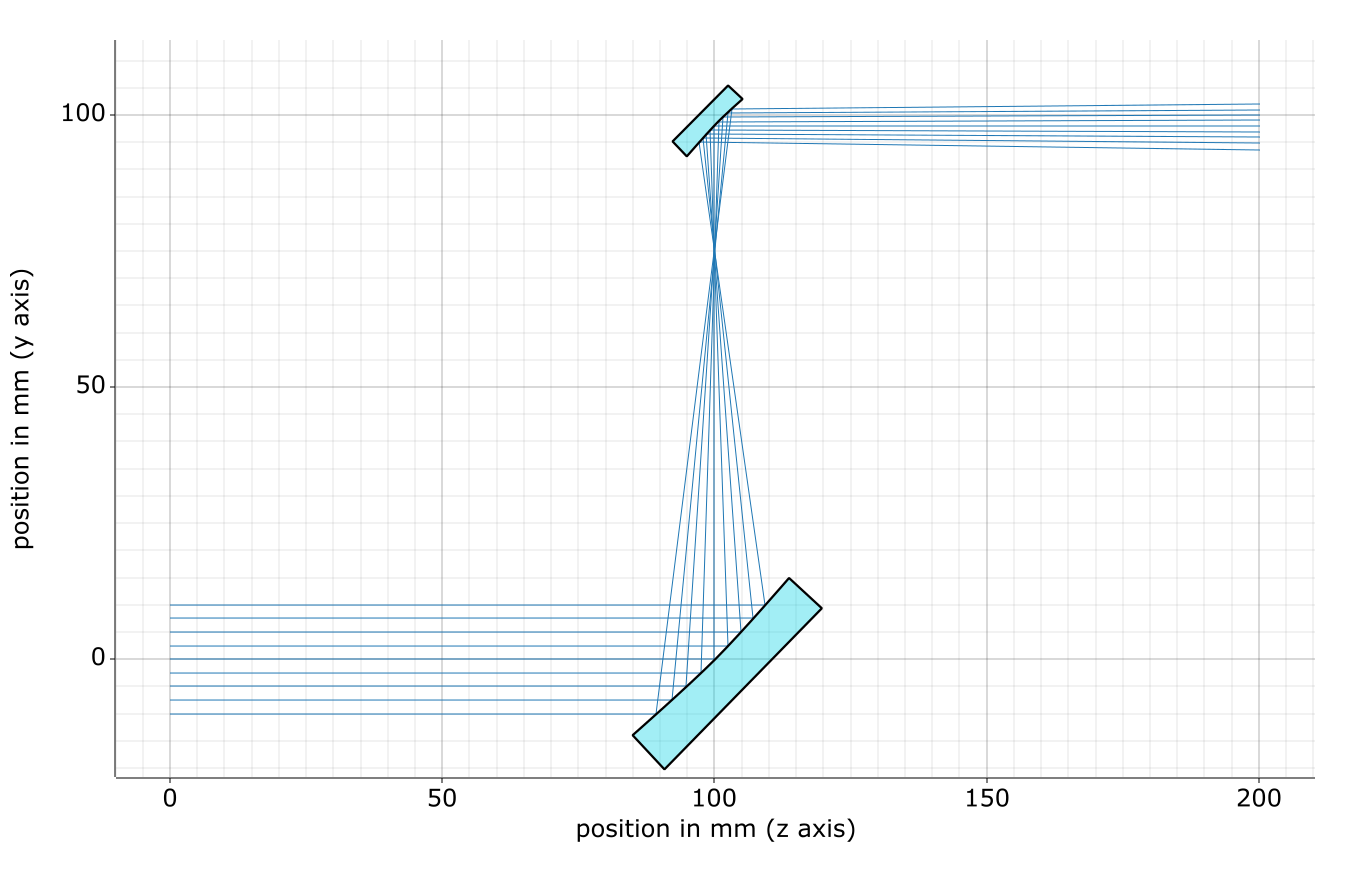
Supports arbitrary 3D orientations and off-axis angles for ease of use.

Figure : Example of a demagnifying 90° off-axis parabola telescope with slight misalignment. See attached report..

## Further Improvements

In addition to the key features above, v0.6.0 includes numerous bug fixes, smaller enhancements, and an extensively expanded test suite. Detailed information is available in the [release section](https://git.gsi.de/phelix/rust/opossum/-/releases) of our [repository](https://git.gsi.de/phelix/rust/opossum).

**Statistics:**

* 79 tickets closed
* > 250 repository commits
* > 800 unit tests
* > 88 % code coverage
* > 43,000 lines of code

# Looking Ahead: v0.7

The next release, v0.7, will introduce a **graphical user interface (GUI)** for generating OPOSSUM model files and conducting analyses. This marks the first step toward making the software accessible to a broader audience. We’re excited to receive your feedback and suggestions as we continue to improve OPOSSUM.

Stay tuned for updates!