Ray tracing in thermally loaded optical systems

Phillip Lino Rall\textsuperscript{1,2,∗} and Christoph Pflaum\textsuperscript{1,2}
\textsuperscript{1} University of Erlangen-Nuremberg, Chair for System Simulation (I10), Cauerstraße 11, Erlangen, Germany
\textsuperscript{2} Erlangen Graduate School in Advanced Optical Technologies (SAOT), Erlangen, Germany

High-power laser beams are usually imaged with the help of suitable optics. The propagation of such beams, through optical systems, often result in a focus shift. This is mainly due to the deposited power in the beam acting as a heat source in the optical elements. Finally, the resulting change in temperature leads to both, a deformation of the lens surface, and a change of the refractive index in the lens. Ray tracing and the finite element method are needed in order to analyze the above effects. The accuracy and the computational time of focus shift calculations, in both simulation techniques, are strongly influenced by the interpolation of the data. This paper will present, a newly proposed interpolation method, allowing a fast and accurate interpolation of data using an unstructured finite element grid. This method will specifically focus on the accurate calculation of laser beam thermal load inside an optical lens.

© 2021 The Authors. Proceedings in Applied Mathematics & Mechanics published by Wiley-VCH GmbH.

1 Introduction

High power laser systems are needed in multiple applications in the fields of material processing, medical treatment, or fundamental research. One of the most crucial points, in engineering such lasers systems, is stability in a high power regime. The temperature of a lens usually changes due to the induced thermal load, deposited by the beam, and leads to a deformation of the lens and a change in its refractive index. This causes the thermal lens effect, which induces a focal shift, as shown in Fig. 1. It is therefore of high interest, to simulate optical systems under the thermal influence, in order to obtain a thermally stable operating state and investigate compensation methods.

A new method of interpolating continuous data on the unstructured grid and efficiently modeling the heat source is presented in two parts, using the proposed framework. Continuous data interpolation and mapping of the heat source are the main focus of the computational effort in the presented simulation.

Interpolation is usually applied with help of an additional regular grid, on which one can define e.g. splines or polynomials, for high accuracy, or use trilinear interpolation on regular grids, for fast results. We propose a new interpolation method, which uses a smart trilinear interpolation directly on unstructured grids, allowing both fast and accurate simulation results. Additionally, a new method for mapping the heat source is presented, by only evaluating the surface intersection points of propagated rays. This contrasts with regular heat source mapping methods, as a significantly lower number of rays can be used, thereby speeding up simulation time.

The combined use of both proposed methods enable extensive parameter studies allowing for significantly more new design opportunities and further compensation methods.

2 Heat source modeling

An optical beam can be modeled with help of rays where each ray is associated with a certain power. All rays can then be propagated independently, starting at the vertices of the initial intensity distribution (see Fig. 2). It is first necessary to obtain the intensity in normal direction $I_\mathbf{n}$ to the element surface in order to represent the optical beam as a heat source. Also, the power flow inside the element, in propagation direction $\mathbf{s}$, is required.

In the geometrical optics regime, the intensity normal to a surface can be accurately approximated with rays. The intensity variation in a ray tube, formed by closely spaced rays, can then be written in the following manner, as in [1]

\begin{align}
I_\mathbf{n}(\xi, \eta, \sigma_{\text{front}}) &= I_\mathbf{n}(\xi, \eta, \sigma_{\text{init}}) \frac{dA_{\text{init}}}{dA_{\text{front}}}, \\
I_\mathbf{n}(\xi, \eta, \sigma_{\text{back}}) &= I_\mathbf{n}(\xi, \eta, \sigma_{\text{front}}) \frac{dA_{\text{front}}}{dA_{\text{back}}} \exp \left\{ -2k_0 \int_{\sigma_{\text{back}}}^{\sigma_{\text{front}}} n'' d\sigma \right\} (1 - c_{\text{abs}}), \label{eq:1} \\
I_\mathbf{s}(\xi, \eta, \sigma_{\text{front}}) &= \frac{I_\mathbf{n}(\xi, \eta, \sigma_{\text{front}})}{\mathbf{n} \cdot \mathbf{s}}, \\
I_\mathbf{s}(\xi, \eta, \sigma_{\text{back}}) &= \frac{I_\mathbf{n}(\xi, \eta, \sigma_{\text{back}})}{\mathbf{n} \cdot \mathbf{s}},
\end{align}

where $dA$ is the ray tube cross section at the corresponding planes, $\sigma$ the physical arc length of the ray, $\mathbf{n}$ is the surface normal, and $\mathbf{s}$ is the ray direction in the optical element. $2k_0n'' = \alpha_{\text{abs}}$ is the local absorption coefficient, attenuating the intensity,

\textsuperscript{∗} Corresponding author: email philip.lino.rall@fau.de
Fig. 1: Thermal lens effect visualized. Rays propagating through a nonaberrated optical element (solid arrows) and rays propagating through an element with thermal gradient and surface deformation (dashed arrows). The dashed line indicates the mechanical deformation, the gray color the temperature.

where $k_0$ is the wave number and $n''$ is the imaginary part of the refractive index. $c_{abs}$ is the surface absorption coefficient. The relation of intensity and cross section is also known as the intensity law of geometrical optics [2]. Therefore, Eq. (1) can be used to approximate the intensity along optical beams, only by calculating the corresponding area increment $dA$. The area is defined orthogonally to the surface normal $n$ on an optical element or a stop plane along a ray, and each ray has its own area increment. The area increment $dA$ can be calculated by summing one-third of the adjacent triangle areas, $\frac{1}{3}A_T$ (marked in red), see Fig. 2. This respectively allows calculation of intensity on arbitrary continuous surfaces, such as lenses or mirrors.

The weak formulation of time dynamic heat equation is given by

\[
\partial Q = I_{n}^{\text{front}} c_{abs} + I_{n}^{\text{back}} c_{abs},
\]

\[
Q = \alpha_{abs} I_{s}^{\text{back}} \frac{(\sigma - \sigma_{\text{front}})}{\sigma_{\text{front}} - \sigma_{\text{back}}} + \alpha_{abs} I_{s}^{\text{front}} \frac{(\sigma_{\text{back}} - \sigma)}{\sigma_{\text{front}} - \sigma_{\text{back}}},
\]

where $n$ is the surface normal on $\Omega$ and $s$ the direction of the energy flow. $\sigma_{\text{front}}$ and $\sigma_{\text{back}}$ are the points of the ray-surface intersection. Assuming that the ray can be approximated by a straight line, $\sigma$ is the corresponding trajectory in between. The ray trajectory is not a straight line in general, however, it is very close to one, such that the resulting error is negligible.

In comparison to the regular ray tracing method, where each ray is associated with a certain power, this method has the benefit of significantly reducing the number of rays necessary to obtain an accurate heat source. Additionally, this method utilizes power density rather than traditional ray tracing methods, which provide power per ray. Calculation of heat source requires power density rather than pure power.

**2.1 Heat equation**

The weak formulation of time dynamic heat equation is given by

\[
\kappa_0 \int_{\Omega} \nabla T v_h d(\Omega) + H \int_{\partial \Omega} T v_h d(\partial \Omega) + \int_{\Omega} \frac{\partial T}{\partial t} c_p \rho v_h d(\Omega)
= H \int_{\Omega} T v_h d(\partial \Omega) + \int_{\Omega} Q^\circ v_h d(\Omega) + \int_{\partial \Omega} \partial Q v_h d(\partial \Omega) \forall v_h \in H^1_0(\Omega),
\]

where $k_0$ is the thermal conductivity, $H$ the heat transfer coefficient, and $c_p \rho$ the volumetric heat capacity. The heat source terms $\partial Q$ and $Q^\circ$ are modeled by Eq. (2). The boundary $\partial \Omega$ can be split in actively cooled surfaces $\partial \Omega_a$ and passively cooled.
surfaces $\partial \Omega_p$. Actively cooled surfaces have contact with metal, which efficiently transfers heat, while passively cooled surfaces transfer heat only via air or thermal radiation. In case of strong cooling, $-\kappa_0 \frac{\partial T}{\partial n} \ll H(T - T_c)$ at the boundary $\Omega_a$, the Dirichlet boundary condition can be used, such that $T = T_c$ on $\partial \Omega_a$.

### 3 Ray tracing through in homogeneous media

Both, change in temperature and deformation, influence the propagation of rays through media. Temperature change induces a refractive index gradient, which deflects rays inside the medium, whereas deformation mainly alters refraction direction on the surface of optical elements. This article will not discuss deformation and will instead focus on refractive index.

The refractive index depends in good approximation linear on the temperature, such that

$$n \approx n_0 + \frac{dn}{dT} \Delta T = n_0 + \frac{dn}{dT} (T - T_0).$$

As the temperature $T$ is not constant, this effect induces refractive index gradients. In order to propagate rays in gradient media, it is necessary to solve the ray equation in the following manner

$$\frac{d}{d\sigma} \left( n(r) \frac{dr}{d\sigma} \right) = \nabla n(r), \quad n(r) \frac{dr}{d\sigma} = p,$$

where $r$ is the position of a ray, and $p$ its directional vector. This equation has to be discretized properly, e.g. by using Runge-Kutta [3] or Euler discretization. For all discretization schemes, it is crucial to obtain fast and accurate continuous data on the domain $\Omega$, in order to solve Eq. (5). Additional to the refractive index $n$, also the gradient $\nabla n$ is necessary.

### 4 Accurate and fast data interpolation

In order to solve the ray equation Eq. (5), and additionally consider deformation, it is necessary to obtain continuous data on discretized domain $\Omega_h$. Partial derivatives $\frac{\partial}{\partial x}$ and $\frac{\partial}{\partial \sigma}$ are also needed accordingly. Simulation accuracy thus not only depends on the finite element simulation but also on data interpolation.

We additionally introduce a new normal projection (NP) interpolation method in order to allow fast and accurate approximation of the trilinear Eq. (6). Using the data locality of ray tracing, this method is almost as fast as using the $O(1)$ trilinear interpolation on regular grids, while at the same time increasing accuracy.

Eq. (3) is solved with the finite element framework UGBlocks, which utilizes trilinear, blockstructured, transformed basis functions. Each cell consists of 8 corners $x_{000}, ..., x_{111}$, which can be used to formulate a trilinear interpolation. Each point $x$ inside the hexahedron can be described by the three coefficients $\alpha, \beta, \gamma \in [0, 1]$, such that

$$x(\alpha, \beta, \gamma) = a + b\alpha + c\beta + d\gamma + e\alpha\beta + f\beta\gamma + g\alpha\gamma + h\alpha\beta\gamma.$$ (6)

For interpolating a given point $x_{Int}$ in a hexahedron, the trilinear Eq. (6) has to be solved.

Using the NP interpolation method, this can be approximated by calculating the shortest distance of the point $x_{Int}$ to each of the six surfaces of the cell, such that

$$\alpha \approx \frac{d_N}{d_N + d_S}, \quad \beta \approx \frac{d_E}{d_E + d_W}, \quad \gamma \approx \frac{d_D}{d_D + d_T}, \quad d_i = n_i \cdot (x_{Int} - c_i),$$ (7)

where $d_i$ is the distance to the surface of the cell, $c_i$ is the central point of face $i$, and $n_i$ the corresponding normal vector. As an example, on the top side of Fig. 3, $c_1$ and $n_1$ are given by

$$n_T = \frac{(x_{111} - x_{001}) \times (x_{011} - x_{101})}{|(x_{111} - x_{001}) \times (x_{011} - x_{101})|}, \quad c_T = 0.25(x_{111} + x_{001} + x_{101} + x_{011}).$$ (8)

![Fig. 3: Grid, used to discretize a regular lens. Each hexahedron consists of 8 corners, $x_{000} \ldots x_{111}$. The points $r_{n-1} \ldots r_{n+2}$ represent a ray, entering and leaving a single hexahedron.](https://www.gamm-proceedings.com)
In general, the hexahedron \( H(\mathbf{r}_i) \), in which a point \( \mathbf{r}_i \) is contained, is unknown and has to be found first. This can be very expensive, especially for randomly distributed points. In the case of ray tracing, however, which accesses data at subsequent nearby points, the hexahedron finding can speed up significantly. Consider the points \( \mathbf{r}_{n-1} \ldots \mathbf{r}_{n+2} \) of Fig. 3. If the step size \( \Delta \sigma \) is small enough, \( H(\mathbf{r}_{n+2}) \) is always directly adjacent to \( H(\mathbf{r}_{n+1}) \). The new hexahedron can be found by evaluating Eq. (7) with the new point in the previous hexahedron. One or more of the coordinates \( \alpha, \beta, \gamma \) will not be in \([0, 1]\), which can be used to directly find the new hexahedron. For example, if Eq. (7) gives \( \alpha = 0.2, \beta = 0.3, \gamma = 1.1 \), then the correct hexahedron will be found by incrementing the hexahedron index, corresponding to the direction of \( \gamma \), by 1. Similarly, the same method can be applied to all rays, as they are propagated in close vicinity to each other. As a result, an expensive time consuming full hexahedron search is only once required for each simulation.

5 Simulation results

A comparison of errors, for different interpolation methods, are shown in Fig. 4. Different functions, similar to Lee Et.al. [4], are used. These figures show the interpolation error and time for evaluating a single point in a discretized lens domain \( \Omega \). A comparison of errors, for different interpolation methods, are shown in Fig. 4. Different functions, similar to Lee Et.al. [4], are used. These figures show the interpolation error and time for evaluating a single point in a discretized lens domain \( \Omega \), shown in Fig. 3, for different interpolation algorithms. Trilinear interpolation on an auxiliary regular grid, direct interpolation on the unstructured grid with both, NP and Newton’s iteration, as well as the Multilevel B-Splines (MBA) interpolation [4] are compared accordingly. The auxiliary regular grid is selected, as it has a finer mesh size than an unstructured grid, and results in approximately \( 20 \times \) more grid points.

Fig. 4 shows the major benefits of using the NP interpolation method. NP is almost as fast as using interpolation on a regular, trilinear grid, with much greater accuracy. This is especially true for the boundary of the domain \( \Omega \), which is why the maximal error difference is even more significant. Also, the NP method is much faster than solving Eq. (6) with Newton’s method, while giving no significant difference in interpolation error. Finally, the direct interpolation benefits from significant memory savings, as the additional regular auxiliary grid does not have to be initialized and stored.

For comparison, the MBA interpolation method, described in [4], is also shown. It has an even lower average interpolation error, for the price of being \( \sim 10^5 \times \) times slower than the auxiliary grid interpolation. Interestingly, the MBA has a better average interpolation error, however lacks in the maximums norm. Thus, the newly developed NP interpolation method combines both fast interpolation results and good accuracy.

![Fig. 4: Average interpolation time for a single point evaluation (left) average interpolation error (center) and maximum interpolation error (right).](image)

6 Conclusion

A new interpolation and heat source mapping method was developed for simulation of thermal effects in arbitrary optical systems. This new method allows for fast, accurate, and data saving interpolation while significantly reducing the number of rays required for heat source mapping.

Acknowledgements The authors gratefully acknowledge funding of the Erlangen Graduate School in Advanced Optical Technologies (SAOT) and the German Research Foundation. Open access funding enabled and organized by Projekt DEAL.

References

[1] F. Wyrowski and M. Kuhn, Journal of Modern Optics 58(5-6), 449–466 (2011).
[2] M. Born and E. Wolf, Principles of Optics: Electromagnetic Theory of Propagation, Interference and Diffraction of Light (7th Edition), 7th edition (Cambridge University Press, 1999).
[3] A. Sharma, D. V. Kumar, and A. K. Ghatak, Tracing rays through graded-index media: a new method (1982).
[4] S. Lee, G. Wolberg, and S. Shin, IEEE Transactions on Visualization and Computer Graphics 3(3), 228–244 (1997).