Discontinuous Galerkin methods for Liouville's equation of geometrical optics

Discontinuous Galerkin methods for Liouville's equation of geometrical optics

Robert Adrianus Maria van Gestel

Robert Adrianus Maria van Gestel
Discontinuous Galerkin methods for Liouville's equation of geometrical optics
Eindhoven University of Technology, 2023

The research described in this thesis was performed at the Centre for Analysis, Scientific Computing and Applications (CASA) within the Department of Mathematics and Computer Science at Eindhoven University of Technology, the Netherlands.

This work is part of the research program NWO-TTW Perspectief with project number P15-36, which is (partly) financed by the Netherlands Organisation for Scientific Research (NWO).
This work is part of the Free-Form Scattering Optics program funded by NWO and partners. Program website: `www.freeformscatteringoptics.com`.

A catalogue record is available from the Eindhoven University of Technology Library.

ISBN: 978-90-386-5783-7

Cover design: © Robert van Gestel.
Printed by: Gildeprint – Enschede.

Discontinuous Galerkin methods for Liouville's equation of geometrical optics

PROEFSCHRIFT

ter verkrijging van de graad van doctor aan de Technische Universiteit Eindhoven, op gezag van de rector magnificus prof. dr. S. K. Lenaerts, voor een commissie aangewezen door het College voor Promoties, in het openbaar te verdedigen op donderdag 22 juni 2023 om 13:30 uur

door

Robert Adrianus Maria van Gestel

geboren te Molenschot

Dit proefschrift is goedgekeurd door de promotoren en de samenstelling van de promotiecommissie is als volgt:

voorzitter:	prof. dr. J. J. Lukkien
1e promotor:	dr. ir. J. H. M. ten Thije Boonkkamp
2e promotor:	prof. dr. ir. W. L. IJzerman
copromotor:	dr. ir. M. J. H. Anthonissen
leden:	prof. dipl. -ing. G. J. Gassner (University of Cologne, Germany)
	prof. dr. rer. nat. C. D. Munz (University of Stuttgart, Germany)
	prof. dr. ir. J. J. W. van der Vegt (Universiteit Twente)
	prof. dr. K. P. Veroy-Grepl

Het onderzoek of ontwerp dat in dit proefschrift wordt beschreven is uitgevoerd in overeenstemming met de TU/e Gedragscode Wetenschapsbeoefening.

Contents

Chapter 1

Introduction

In non-imaging optics the goal is to design optical systems that transfer light from a known source distribution to a desired target distribution. These optical systems can consist of curved mirrors and lenses that redirect light via reflection or refraction, but also smoothly varying refractive index fields are possible. In optical design, the shapes of lenses or mirrors can be freeform, as opposed to the surfaces obeying a symmetry such as rotational symmetry. The freeform shape allows for a wider range of optical systems to be designed.

Non-imaging optics is a sub-field of geometrical optics in which light is described in terms of rays. For a constant refractive index field this means rays propagate along a straight line, whereas for a smoothly varying refractive index field rays follow a curved path.

Non-imaging optics started with applications in designing solar energy collectors and concentrators. Modern applications of non-imaging optics include street lighting [63], automotive headlamps [25, 100] and luminaires [72]. Non-imaging optics is different from imaging optics as usually imaging effects are undesirable in non-imaging optical systems [18].

The design process of optical systems, given a source distribution and desired target distribution, is an iterative process. An optical designer starts with an initial guess of the optical system and then the actual target distribution the optical system produces is computed. Based on the actual target distribution the optical designer uses their experience to adjust the optical system. The target distribution must be re-evaluated and the optical system readjusted. This process is repeated until the optical system produces a target distribution that is sufficiently close to the desired distribution.

The actual target distribution is typically computed with (quasi-)Monte Carlo ray tracing [41] in terms of the relevant photometric quantities, such as the illuminance or the luminous intensity. In (quasi-)Monte Carlo ray

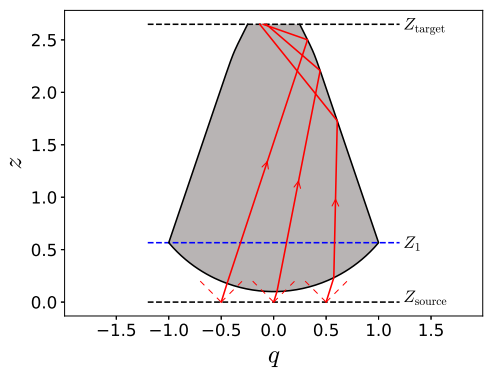
tracing millions or even billions of light rays have to be traced through an optical system to get sufficient resolution. This method can be expensive in computing the photometric quantities to high accuracy, due to its rather slow convergence. Furthermore, low accuracy results make the task of performing numerical optimisation on the optical system very challenging.

A different approach to ray tracing is based on a phase space description of light propagation [50, 76, 95]. Here, phase space is defined as the collection of all positions and direction coordinates of light rays. A single point in phase space corresponds to a single light ray, and its evolution is described by a Hamiltonian system, whenever the refractive index field is smooth. When a light ray hits an optical interface, that is a discontinuity in the refractive index field, the well-known laws of specular reflection or Snell's law of refraction have to be applied. The evolution of the light ray can be parametrised in terms of its arc-length, but also in terms of one its position coordinates. The latter is used in this thesis and the corresponding position coordinate is referred to as an evolution coordinate. Besides the evolution coordinate, phase space is four-dimensional for three-dimensional optics, and two-dimensional for two-dimensional optics.

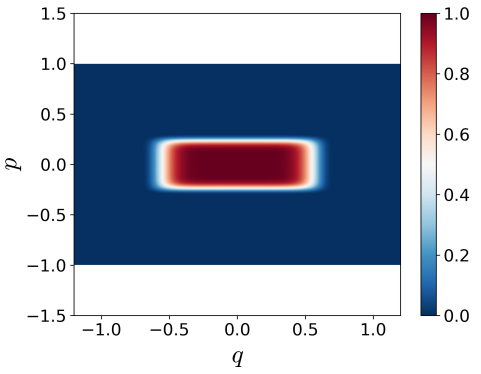
This still only describes how light rays propagate. An energy density is defined on phase space, that is known as the basic luminance. The evolution of the basic luminance is governed by Liouville's equation for geometrical optics, which is a first-order linear hyperbolic partial differential equation. The basic luminance is a fundamental quantity as from the basic luminance both the illuminance and luminous intensity can be computed by integration.

As an example, we consider a dielectric total internal reflection concentrator, a two-dimensional optical system, and the basic luminance, denoted ρ, computed at various heights of the system. The optical system and the basic luminance distributions are shown in Figure 1.1. In the first panel, the optical system is shown together with a few light rays. These light rays are first refracted at the first surface followed by specular reflection on one of the side walls before ending up at the target plane. In the second panel, the basic luminance distribution of the incident light, emitted by the source, is shown as a function of the phase space coordinates. These coordinates are the position q and the direction coordinate p. In the third panel, one can see the refractive effect the first surface has had on the distribution. In the fourth panel, the top and bottom patches correspond to light that was reflected at the side walls.

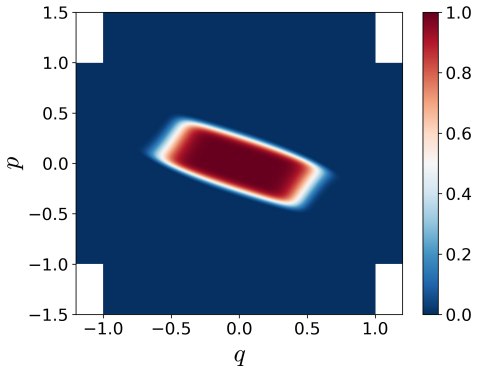
In the propagation of a single light ray we follow its trajectory, i.e., phase space coordinates, along the ray and take into account the basic luminance along the ray. On the other hand, for Liouville's equation the phase space



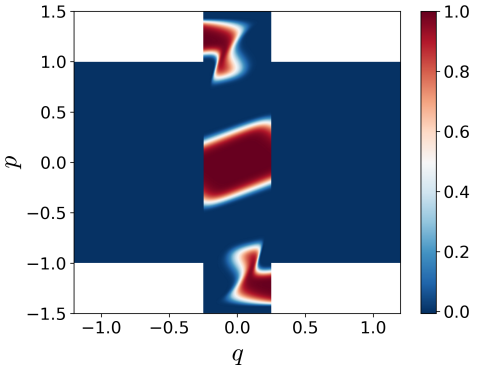
(a) Setup.



(b) Basic luminance ρ at $z = Z_{\text{source}}$.



(c) Basic luminance ρ at $z = Z_1$.



(d) Basic luminance ρ at $z = Z_{\text{target}}$.

Figure 1.1: A two-dimensional dielectric total internal reflection concentrator, where gray colour represents a refractive index $n = 1.5$ and white colour the background medium with $n = 1$.

coordinates are known and we see how the basic luminance evolves as a function of these coordinates. Thus by switching from the propagation of single light rays to Liouville's equation we move from a Lagrangian description to an Eulerian description.

The interaction of light at an optical interface can be modelled in various ways. For example, one can consider the surface to be scattering, otherwise known as diffuse reflection, where a single incident light ray is scattered into a distribution of outgoing light rays. Yet another phenomenon is Fresnel reflection which describes partial reflection and partial transmission of a single light ray, producing two outgoing light rays. Moreover, light can also undergo total internal reflection, also known as specular reflection, in which a light ray is entirely reflected. In all these cases the direction coordinates

of a light ray change discontinuously at an optical interface and the basic luminance is redistributed at the optical interface. These effects are described in terms of a jump condition. The jump condition essentially describes non-local boundary conditions at an optical interface in phase space. Besides the jump condition being a difficulty in and of itself, the optical interfaces can in general have arbitrary shapes. Hence, we have to deal with complicated geometries.

As Liouville's equation describes the evolution of the basic luminance on phase space, we also have to deal with the high-dimensionality of phase space. In addition, the two- and four-dimensional phase space domains evolve or move as a function of the evolution coordinate.

1.1 Discontinuous Galerkin methods

The focus of this thesis is on solving Liouville's equation using numerical methods that belong to the class of discontinuous Galerkin (DG) finite element methods. Finite element methods approximate the solution to a partial differential equation by partitioning the spatial domain into a set of finite elements and computing an expansion into a set of locally defined basis functions. The expansion is computed, by requiring a weak formulation of the equation to hold for a suitable set of test functions on each element. With test functions taken from the set of basis functions, the resulting method is a Galerkin method. Discontinuous Galerkin finite element methods separate themselves from the traditional class of continuous Galerkin finite element methods, by not imposing any continuity across the boundary between elements.

The first DG method was introduced by Reed and Hill [77] in 1973, to solve a linear hyperbolic partial differential equation for neutron transport. Later on in a series of papers [19–21, 23] by Cockburn and Shu et al., the authors established a framework to solve non-linear time-dependent problems using DG for space discretisation and Runge-Kutta methods for time discretisation. The DG methods were generalised to convection-diffusion equations in [5, 6, 22]. For a more extensive review of DG methods we refer the reader to [81].

An important development for hyperbolic problems was made by Qiu et al. [74] and Dumbser et al. [32] by introducing the Arbitrary Derivative (ADER) approach into the DG method. The DG method allows for arbitrary high order of accuracy in space. By using the ADER approach high order of accuracy in time can also be achieved. In the previously mentioned works, an element-local temporal Taylor expansion is computed where temporal derivatives are replaced with spatial derivatives using the Cauchy-Kovalewski or Lax-Wendroff procedure. This procedure becomes rather cumbersome for

non-linear partial differential equations. To allow for a more general treatment a local space-time Galerkin predictor method based on a space-time weak formulation was developed by Dumbser et al. [28, 29].

The ADER-DG schemes yield a fully-discrete explicit scheme as opposed to the multi-stage Runge-Kutta methods applied to the semi-discrete spatial discretisation by DG. In these DG methods neighbouring elements interact via a numerical flux defined at the boundary of each element. For each stage in a Runge-Kutta based DG method communication between the elements is required, where a single update step is completed after all stages have been computed. On the other hand, in the ADER-DG method an element-local, thus without requiring information from neighbours, predictor is computed and requires communication only once per update step. Because less communication is required, ADER-DG methods can achieve higher parallelisation efficiency than Runge-Kutta based DG methods; see [30, 36]. Furthermore, these ADER-DG methods require only as much memory storage as a forward Euler method.

The DG methods covered in this thesis use a polynomial basis with compact support on each element. Hence, these DG methods have a compact stencil. The order of the approximation is increased by using a higher-degree polynomial basis. DG methods can deal with complex geometries. Even curved boundaries can be accommodated by the use of curvilinear elements, see for instance [56, 59]. A moving mesh can also be used, e.g., see [60, 67].

DG methods have a particular flexibility in the sense that they can incorporate local adaptivity of the polynomial degree and adaptive mesh refinement. This is especially important in resolving small local features of the solution, without excessively increasing the computation time. Combining these techniques with local time stepping, in which each element is allowed to run at its own locally determined stepsize, makes these methods computationally very efficient. See, for instance, the works of Dumbser et al. [31] and Dumbser [27] where local time stepping is employed in the context of high order ADER-DG and ADER-WENO finite volume (FV) methods, and see Dumbser et al. [33] and Zanotti et al. [97] for the combination of adaptive mesh refinement, local time stepping and high order ADER-WENO FV and ADER-DG methods, respectively.

To summarise, DG methods have a compact stencil, and can deal with complex geometries and one can choose the order of approximation. Memory requirements for ADER-DG are fewer than traditional Runge-Kutta based DG methods, which is useful for Liouville's equation considering the high-dimensionality of phase space. These reasons combined make the DG methods very suitable for solving Liouville's equation numerically, whilst allowing

the potential for resolving local features without incurring excessively high computational costs.

An alternative to the Runge-Kutta or ADER based DG methods is one based on a semi-Lagrangian principle for time integration. Semi-Lagrangian methods are based on an exact or approximate evolution of the considered partial differential equation. For a hyperbolic partial differential equation, this means the solution is propagated along its characteristics. Semi-Lagrangian methods can be CFL-free, allowing the use of very large stepsizes. These kinds of methods are for example used in the Vlasov(-Poisson) simulation community [13, 35, 75] and used for weather prediction [40]. The efficiency of the semi-Lagrangian discontinuous Galerkin methods, because of the CFL-free property, make them also an attractive method for Liouville's equation, especially when light rays are piecewise straight lines.

1.2 Main results of this thesis

In this thesis Liouville's equation for geometrical optics is derived. Two jump conditions are presented, one in which light rays are either fully reflected or fully refracted and one which describes Fresnel reflections. The jump conditions are related to a maximum principle for Liouville's equation.

First, the discretisation of Liouville's equation for two-dimensional optics is considered. The discretisation of the jump condition and its inclusion into a DG method is tackled by deriving appropriate local energy balances that must hold at a flat optical interface. Together these balances, and the geometric connectivity at an optical interface are used in a least-squares matching procedure that ensures the DG method is energy conservative. The methodology was tested for a flat optical interface and resulted in the article [87]:

- R. A. M. van Gestel, M. J. H. Anthonissen, J. H. M. ten Thije Boonkkamp, and W. L. IJzerman. An energy conservative hp-method for Liouville's equation of geometrical optics. *Journal of Scientific Computing*, 89(1):1-35, 2021.

Next, the discretisation of the jump condition was formally extended to deal with curved optical interfaces by deriving local energy balances. Curved optical interfaces are geometrically dealt with in two ways in the DG discretisation. Mainly, we align the mesh with optical interfaces by allowing the mesh to move in a fully discrete ADER-DG method. However, the moving mesh method alone is not sufficient and thus we introduced a sub-cell interface method. This resulted in the following article [89]:

- R. A. M. van Gestel, M. J. H. Anthonissen, J. H. M. ten Thije Boonkkamp, and W. L. IJzerman. An ADER discontinuous Galerkin method on moving meshes for Liouville's equation of geometrical optics. *Journal of Computational Physics*, 2023.

A novel solver is developed that combines semi-Lagrangian DG elements, ADER-DG elements on a moving mesh and local time stepping. The hybrid solver yields improved performance by using the efficient semi-Lagrangian DG scheme away from optical interfaces. Local time stepping ensures the reduction in stepsize caused by the unavoidable small elements only have a local impact. This novel hybrid solver is described in the following article that has been submitted [88]:

- R. A. M. van Gestel, M. J. H. Anthonissen, J. H. M. ten Thije Boonkkamp, and W. L. IJzerman. A hybrid semi-Lagrangian DG and ADER-DG solver on a moving mesh for Liouville's equation of geometrical optics. *submitted to Journal of Computational Physics*, 2023.

The ADER-DG solver is used to perform a parameter study of a lens plate. The numerical solutions feature discontinuities, which are dealt with via a modal filter and limiter. Finally, we make a first attempt at solving Liouville's equation for three-dimensional optics (four-dimensional phase space). The use of moving and curvilinear elements on a four-dimensional mesh in an ADER-DG method are described.

Finally, these numerical methods have been implemented from scratch by the author of this thesis. Aside from the DGSEM, these methods have been coded in C++. A brief discussion on this piece of developed software is provided in Appendix D.

1.3 Outline of this thesis

In Chapter 2 non-imaging optics and the conservation of the basic luminance are discussed. The latter property and a Hamiltonian formulation for the propagation of light are combined to derive Liouville's equation. Two jump conditions are presented and are related to a maximum principle for Liouville's equation.

Next, an introduction to discontinuous Galerkin methods in one dimension is presented in Chapter 3 for a scalar hyperbolic partial differential equation. Core concepts such as polynomial interpolation, derivatives and integration are briefly discussed. Thereafter, a DG spectral element method and semi-Lagrangian DG methods are presented.

In Chapter 4 we semi-discretise Liouville's equation using a DG spectral element method. We present the discretisation of the jump condition by detailing local energy balances for a flat optical interface and a least-squares matching procedure. The DG method is applied to two examples, one featuring a smooth refractive index field, and one called the 'bucket of water' in which we showcase the energy conservative property of the DG method in the presence of a flat optical interface. In the latter example, we compare the method to quasi-Monte Carlo ray tracing for computing the illuminance.

The extension to curved optical interfaces of the DG method is described in Chapter 5. An ADER-DG method on a moving mesh with an explicit temporal Taylor series is used, and the sub-cell interface method is introduced. The discretisation of the jump condition is formally extended to curved optical interfaces. Local energy balances for curved optical interfaces are derived. The resulting (ADER-DG) solver is applied to two examples, a meniscus lens and a dielectric total internal reflection concentrator, and the performance is compared to quasi-Monte Carlo ray tracing.

The description of the novel hybrid solver that combines semi-Lagrangian DG elements, ADER-DG elements on a moving mesh and local time stepping is given in Chapter 6. The energy-conservative coupling between semi-Lagrangian DG and ADER-DG elements in the presence of hanging nodes, due to local time stepping, and the coupling between multiple ADER-DG elements with hanging nodes are discussed. The hybrid solver is applied to the two examples from Chapter 5 and the performance is compared to the pure ADER-DG solver and quasi-Monte Carlo ray tracing.

In Chapter 7 we discuss the discretisation of the jump condition that describes Fresnel reflections in the DG method. The result is again an energy-conserving DG method.

A parameter study of a lens plate is performed in Chapter 8. A modal filter and limiter are applied to deal with the oscillations, that result from discontinuities, in the numerical solution.

An ADER-DG method on a moving curvilinear four-dimensional phase space mesh is presented in Chapter 9. An explicit temporal Taylor series is used. The method is applied to two examples, a tilted cylinder and a compound parabolic concentrator, and the performance is compared to quasi-Monte Carlo ray tracing.

Finally, in Chapter 10 we present conclusions and give directions for possible future research.

Chapter 2

Non-imaging optics and Liouville's equation

In this chapter, we discuss non-imaging optics and detail key conservation properties. First, in Section 2.1 transfer of radiation will be formulated in terms of basic luminance, which is a conserved variable. Propagation of light is presented in terms of a Hamiltonian system for phase space coordinates, position and momentum, in Section 2.2. In Section 2.3, the concepts of non-imaging optics and the Hamiltonian formulation are combined to derive Liouville's equation, which describes the transport of basic luminance on phase space. A crucial component of modelling optical systems with Liouville's equation, is incorporating the effect of optical interfaces on the basic luminance. This is described by a jump condition that connects different parts in phase space, as the momentum changes discontinuously at an optical interface. This discontinuous change in momentum is described by the law of specular reflection and Snell's law of refraction. In Section 2.4 we present the jump condition that models Fresnel reflections, which describes partial reflections. Finally, we relate the jump condition to a maximum principle for Liouville's equation, which states that no new maxima can be generated.

2.1 Non-imaging optics

In non-imaging optics we consider the transfer of luminous or radiant flux between surfaces. A source emits a beam of radiation, carrying a finite amount of flux. This flux can be either a luminous flux, measured in lumen (lm), or a radiant flux, measured in Watts (W), depending on whether photometric or radiometric quantities are being used. The units of flux densities, think

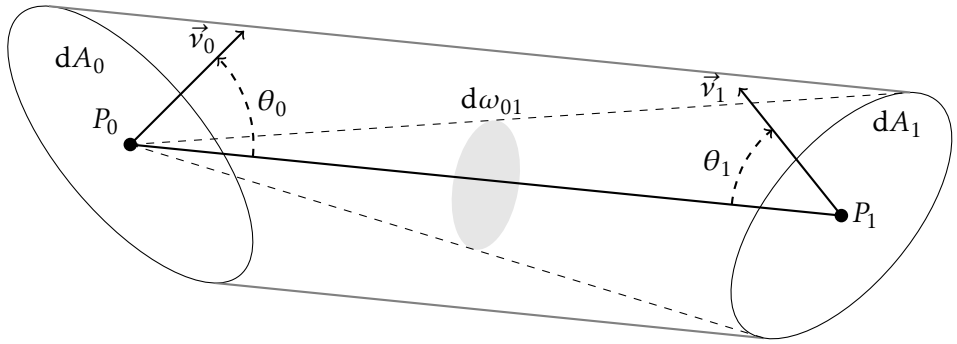


Figure 2.1: Illustration of a beam of radiation through a medium with constant refractive index n. The surfaces $\mathrm{d}A_0$, $\mathrm{d}A_1$ have normals $\vec{\nu}_0$ and $\vec{\nu}_1$, and P_0 and P_1 centroids of the surfaces that are a distance R apart.

of flux per surface area for example, only differ in the base unit of their respective flux. For photometric quantities this base unit is lumen, whereas for radiometric quantities the base unit is Watt. Photometric quantities take into account the sensitivity of the human eye to light. In this section, we will use the terminology from photometry and at the end of this section we present Table 2.1 describing the equivalent terminology from radiometry.

The luminous flux is denoted by the symbol Φ. In the absence of losses, by for example absorption, the total flux Φ throughout an optical system is conserved. A related quantity is the luminance that is denoted by ρ^*, which is defined as [18, 66]

$$\rho^* = \frac{\mathrm{d}\Phi}{\mathrm{d}A\cos\theta\,\mathrm{d}\omega}, \tag{2.1}$$

where $\mathrm{d}\Phi$ is an infinitesimal amount of flux carried by an infinitesimal beam, $\mathrm{d}A\cos\theta$ the projected area perpendicular to the beam with units m^2 and $\mathrm{d}\omega$ the solid angle measured in steradian sr. Hence, the unit for luminance is $\mathrm{lm\,m^{-2}\,sr^{-1}}$.

In the following we consider an infinitesimal surface element $\mathrm{d}A_0$ emitting radiation in the direction of $\mathrm{d}A_1$; see Figure 2.1. The centroids, P_0 and P_1, of the surface elements are connected by a line, the central ray, and the centroid points are a finite distance R apart. Furthermore, the angle between this line and the normal $\vec{\nu}_0$ to the surface element $\mathrm{d}A_0$ is denoted by θ_0 and a similar definition holds for θ_1. The radiation emitted from $\mathrm{d}A_0$ is considered to be an elementary light beam. The elementary light beam is composed of all the rays passing through both $\mathrm{d}A_0$ and $\mathrm{d}A_1$ [68].

In a medium of constant refractive index n the quantity ρ^* is conserved. This is derived as follows. The flux leaving the surface element $\mathrm{d}A_0$ and

arriving at $\mathrm{d}A_1$ can be described by

$$\mathrm{d}\Phi_0 = \rho_0^* \,\mathrm{d}A_0 \cos\theta_0 \mathrm{d}\omega_{01}, \tag{2.2a}$$

and the flux entering surface element $\mathrm{d}A_1$ reads

$$\mathrm{d}\Phi_1 = \rho_1^* \,\mathrm{d}A_1 \cos\theta_1 \mathrm{d}\omega_{10}, \tag{2.2b}$$

with ρ_0^* and ρ_1^* the luminance at their respective surfaces. Moreover, $\mathrm{d}\omega_{01}$ specifies an element of solid angle in the direction of the central ray, formed (subtended) at P_0 by $\mathrm{d}A_1$; see Figure 2.1. A similar definition holds for $\mathrm{d}\omega_{10}$. The elements of solid angles read

$$\mathrm{d}\omega_{01} = \frac{\cos\theta_1 \,\mathrm{d}A_1}{R^2}, \quad \mathrm{d}\omega_{10} = \frac{\cos\theta_0 \,\mathrm{d}A_0}{R^2}. \tag{2.3}$$

With these definitions the fluxes can be written as

$$\mathrm{d}\Phi_0 = \rho_0^* \,\mathrm{d}A_0 \cos\theta_0 \frac{\cos\theta_1 \,\mathrm{d}A_1}{R^2}, \tag{2.4a}$$

$$\mathrm{d}\Phi_1 = \rho_1^* \,\mathrm{d}A_1 \cos\theta_1 \frac{\cos\theta_0 \,\mathrm{d}A_0}{R^2}. \tag{2.4b}$$

Since it is a lossless system, conservation of energy tells us that $\mathrm{d}\Phi_0 = \mathrm{d}\Phi_1$ which in turn implies $\rho_0^* = \rho_1^*$. The above relations also imply étendue conservation. Here étendue is defined by [18]

$$\mathrm{d}\mathcal{U} = n^2 \mathrm{d}A \cos\theta \,\mathrm{d}\omega. \tag{2.5}$$

The expressions (2.4a)-(2.4b) can be written in terms of étendue as follows

$$\mathrm{d}\Phi_0 = \frac{\rho_0^*}{n^2}\mathrm{d}\mathcal{U}_0 \;\text{ with }\; \mathrm{d}\mathcal{U}_0 = n^2 \mathrm{d}A_0 \cos\theta_0 \mathrm{d}\omega_{01}, \tag{2.6a}$$

$$\mathrm{d}\Phi_1 = \frac{\rho_1^*}{n^2}\mathrm{d}\mathcal{U}_1 \;\text{ with }\; \mathrm{d}\mathcal{U}_1 = n^2 \mathrm{d}A_1 \cos\theta_1 \mathrm{d}\omega_{10}. \tag{2.6b}$$

From the relations for the solid angles (2.3) combined with (2.6) it is clear that étendue is conserved, i.e., $\mathrm{d}\mathcal{U}_0 = \mathrm{d}\mathcal{U}_1$.

Consider now the case where a beam of radiation hits an optical interface, i.e., a discontinuity in the refractive index, and the beam is refracted (transmitted). Following the derivation of Nicodemus [68], the relation between ρ^* before and after the optical interface can be derived by using the definition of flux and applying conservation of energy. Put differently, the flux incident on any surface element $\mathrm{d}A$ through any element of solid angle $\mathrm{d}\omega_\mathrm{i}$ in the first medium, must be equal to the flux transmitted from the same surface element

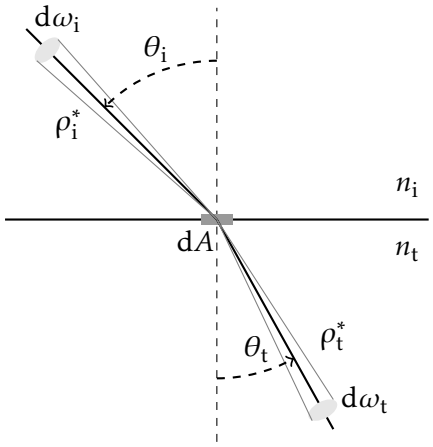


Figure 2.2: Illustration of a beam of radiation that is subject to refraction. After [66].

into the solid angle $\mathrm{d}\omega_\mathrm{t}$ in the second medium; see Figure 2.2. This holds assuming there are no losses by absorption, scattering or Fresnel reflections. Here the subscripts i and t denote the variables with respect to the incident and transmitted beams, respectively. The fluxes before and after the optical interface are given by

$$\mathrm{d}\Phi_\mathrm{i} = \rho_\mathrm{i}^* \,\mathrm{d}A \cos\theta_\mathrm{i} \mathrm{d}\omega_\mathrm{i} = \rho_\mathrm{i}^* \,\mathrm{d}A \cos\theta_\mathrm{i} \sin\theta_\mathrm{i} \mathrm{d}\theta_\mathrm{i} \mathrm{d}\varphi, \tag{2.7a}$$

$$\mathrm{d}\Phi_\mathrm{t} = \rho_\mathrm{t}^* \,\mathrm{d}A \cos\theta_\mathrm{t} \mathrm{d}\omega_\mathrm{t} = \rho_\mathrm{t}^* \,\mathrm{d}A \cos\theta_\mathrm{t} \sin\theta_\mathrm{t} \mathrm{d}\theta_\mathrm{t} \mathrm{d}\varphi, \tag{2.7b}$$

respectively, where we used $\mathrm{d}\omega = \sin\theta \mathrm{d}\theta \mathrm{d}\varphi$ with θ and φ denoting the polar and azimuthal angles in a spherical coordinate system with origin at the centroid of $\mathrm{d}A$. Note that the azimuthal angle does not change by refraction. Each ray within the beam is refracted according to the well-known Snell's law

$$n_\mathrm{i} \sin\theta_\mathrm{i} = n_\mathrm{t} \sin\theta_\mathrm{t}, \tag{2.8}$$

where n_i and n_t denote the refractive indices of the incident and transmitted media, respectively. Not every ray has the same angle of incidence in the beam. Consequently, the differentials $\mathrm{d}\theta_\mathrm{i}$ and $\mathrm{d}\theta_\mathrm{t}$ can be related by Snell's law, which is expressed as follows

$$n_\mathrm{i} \cos\theta_\mathrm{i} \,\mathrm{d}\theta_\mathrm{i} = n_\mathrm{t} \cos\theta_\mathrm{t} \,\mathrm{d}\theta_\mathrm{t}. \tag{2.9}$$

Combining equations (2.8) and (2.9) leads to

$$n_\mathrm{i}^2 \sin\theta_\mathrm{i} \cos\theta_\mathrm{i} \,\mathrm{d}\theta_\mathrm{i} = n_\mathrm{t}^2 \sin\theta_\mathrm{t} \cos\theta_\mathrm{t} \,\mathrm{d}\theta_\mathrm{t}. \tag{2.10}$$

Applying conservation of energy, $\mathrm{d}\Phi_\mathrm{i} = \mathrm{d}\Phi_\mathrm{t}$, substituting the expressions (2.7), and subsequently substituting (2.10) in the result leads to

$$\rho_\mathrm{i}^* \frac{1}{n_\mathrm{i}^2} n_\mathrm{t}^2 \sin\theta_\mathrm{t} \cos\theta_\mathrm{t} \,\mathrm{d}\theta_\mathrm{t} \,\mathrm{d}A \mathrm{d}\varphi = \rho_\mathrm{t}^* \sin\theta_\mathrm{t} \cos\theta_\mathrm{t} \,\mathrm{d}\theta_\mathrm{t} \,\mathrm{d}A \mathrm{d}\varphi, \tag{2.11}$$

so that the relation between ρ_i^* and ρ_t^* reads

$$\frac{\rho_\mathrm{i}^*}{n_\mathrm{i}^2} = \frac{\rho_\mathrm{t}^*}{n_\mathrm{t}^2}. \tag{2.12}$$

Relation (2.12) is known as *basic luminance invariance* in photometry [66], where the quantity ρ^*/n^2 is known as basic luminance. Additionally conservation of étendue over an optical interface holds, i.e., $\mathrm{d}\mathcal{U}_\mathrm{i} = \mathrm{d}\mathcal{U}_\mathrm{t}$, as is evident by multiplying (2.10) by $\mathrm{d}A\,\mathrm{d}\varphi$. A similar result to conservation of étendue and relation (2.12) can be derived for reflective surfaces.

The above concepts hold for three-dimensional optics, whereas in two-dimensional optics the basic luminance invariance is slightly altered with the luminance measured in lumen per meter per radian. A similar derivation for two-dimensional optics can be followed, see [18] for more details, where the following holds

$$\frac{\rho_\mathrm{i}^*}{n_\mathrm{i}} = \frac{\rho_\mathrm{t}^*}{n_\mathrm{t}}.$$

For both two- and three-dimensional optics we denote the basic luminance by ρ, which is defined by

$$\rho = \begin{cases} \dfrac{\rho^*}{n^2} & \text{for 3D optics,} \\[2ex] \dfrac{\rho^*}{n} & \text{for 2D optics,} \end{cases} \tag{2.13}$$

which is conserved in homogeneous media and across optical interfaces. Using this definition the luminous flux $\mathrm{d}\Phi$ can be expressed as

$$\mathrm{d}\Phi = \rho \,\mathrm{d}\mathcal{U}, \tag{2.14}$$

where the étendue for two- and three-dimensional systems reads [18]

$$\mathrm{d}\mathcal{U} = \begin{cases} n^2 \mathrm{d}A \cos\theta \,\mathrm{d}\omega & \text{for 3D optics,} \\ n \,\mathrm{d}l \cos\theta \,\mathrm{d}\theta & \text{for 2D optics,} \end{cases} \tag{2.15}$$

with $\mathrm{d}l$ denoting an infinitesimal line segment.

Table 2.1: Photometric and radiometric quantities for three-dimensional optics.

photometry	unit	radiometry	unit
luminous flux	lm	radiant flux	W
basic luminance	$\mathrm{lm\,m^{-2}\,sr^{-1}}$	basic radiance	$\mathrm{W\,m^{-2}\,sr^{-1}}$
illuminance	$\mathrm{lm\,m^{-2}}$	irradiance	$\mathrm{W\,m^{-2}}$
luminous intensity	$\mathrm{lm\,sr^{-1}}$	radiant intensity	$\mathrm{W\,sr^{-1}}$

From the basic luminance its integral quantities, such as luminous intensity and illuminance, can be determined. For instance, for three-dimensional optics the illuminance $\mathrm{d}E$ is defined by

$$\mathrm{d}E = \frac{\mathrm{d}\Phi}{\mathrm{d}A}. \tag{2.16}$$

Applying definition (2.14) for the luminous flux and definition (2.15) for étendue, $\mathrm{d}E$ can be written as

$$\mathrm{d}E = \rho\, n^2 \cos\theta\, \mathrm{d}\omega. \tag{2.17}$$

The luminous intensity is defined by

$$\mathrm{d}I = \frac{\mathrm{d}\Phi}{\mathrm{d}\omega}. \tag{2.18}$$

Applying again definitions (2.14) and (2.15), $\mathrm{d}I$ can be written as

$$\mathrm{d}I = \rho\, n^2 \cos\theta\, \mathrm{d}A. \tag{2.19}$$

Therefore, if the basic luminance is known then its integral quantities, the illuminance E, luminous intensity I and luminous flux Φ, can be computed by integration. For example, the luminous intensity I can be computed by integrating over an area according to (2.19).

In this brief introduction we have used terminology from photometric quantities. These quantities are summarised in Table 2.1, where also their radiometric counterparts are listed.

2.2 Hamiltonian optics

In geometrical optics the evolution of light rays in a beam of radiation can be cast in a Hamiltonian system. To work towards a Hamiltonian description, we will start by characterising a light ray as a curve through space. The optical

path length L along a curve $\mathcal{C}$, where position vectors $\vec{q}_0$ and $\vec{q}_1$ denote the endpoints of the curve, is defined by

$$L = \int_{\mathcal{C}} n(\vec{q}(s))\,\mathrm{d}s, \tag{2.20}$$

with $\vec{q} \in \mathbb{R}^3$ denoting the position vector of a point on the curve and s denoting the arc length.

Fermat's principle states that the path taken by a light ray between two points in space, is the path that makes the optical path length stationary [47]. From Fermat's principle one can derive an ordinary differential equation (ODE) for the position $\vec{q}$ describing the path of the light ray. The curve $\mathcal{C}$ is parametrised by $\vec{q}(t)$ with $\vec{q}(t_0) = \vec{q}_0$, $\vec{q}(t_1) = \vec{q}_1$ and $t \in [t_0, t_1]$, so that the optical path length can be written as

$$L = \int_{t_0}^{t_1} n(\vec{q}(t))\left|\vec{q}\,'(t)\right| \mathrm{d}t, \tag{2.21}$$

where $'$ denotes $\frac{\mathrm{d}}{\mathrm{d}t}$. The integrand is actually a Lagrangian $\mathcal{L}(\vec{q}, \vec{q}\,') = n(\vec{q})\left|\vec{q}\,'\right|$, so that directly from the Euler-Lagrange equations [1]

$$\frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial \mathcal{L}}{\partial \vec{q}\,'} - \frac{\partial \mathcal{L}}{\partial \vec{q}} = \vec{0}, \tag{2.22}$$

an equation for $\vec{q}$ can be found. Taking the appropriate partial derivatives leads to

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(n\frac{\vec{q}\,'}{\left|\vec{q}\,'\right|}\right) = \frac{\partial n}{\partial \vec{q}}\left|\vec{q}\,'\right|. \tag{2.23}$$

Now by choosing the parametrisation parameter equal to the arc length, i.e., $t = s$ we find that $\left|\vec{q}\,'\right| = 1$, so that equation (2.23) reduces to

$$\frac{\mathrm{d}}{\mathrm{d}s}\left(n\frac{\mathrm{d}\vec{q}}{\mathrm{d}s}\right) = \frac{\partial n}{\partial \vec{q}}. \tag{2.24}$$

Equation (2.24) is known as the ray equation.

An equivalent formulation can be derived using Hamiltonian optics. The Hamiltonian $\mathcal{H}$ is defined by a Legendre transformation of the Lagrangian as follows

$$\mathcal{H}(\vec{q}, \vec{p}) = \vec{p} \cdot \vec{q}\,' - \mathcal{L}(\vec{q}, \vec{q}\,'), \tag{2.25a}$$

with

$$\vec{p} = \frac{\partial \mathcal{L}}{\partial \vec{q}\,'} = n\frac{\vec{q}\,'}{\left|\vec{q}\,'\right|}, \tag{2.25b}$$

and $\vec{q}' = \vec{q}'(\vec{q}, \vec{p})$. Making use of the fact that $n > 0$ so that $\left|\vec{p}\right| = n$, one can rewrite relation (2.25b) to

$$\vec{q}' = \frac{\vec{p}}{\left|\vec{p}\right|}\left|\vec{q}'\right|, \tag{2.26}$$

so that

$$\mathcal{H} = \vec{p} \cdot \frac{\vec{p}}{\left|\vec{p}\right|}\left|\vec{q}'\right| - n\left|\vec{q}'\right| = \left(\left|\vec{p}\right| - n\right)\left|\vec{q}'\right|. \tag{2.27}$$

Now with the arc length as parameter, i.e., $t = s$, we have that $\left|\vec{q}'\right| = 1$, so that the final result reads

$$\mathcal{H}(\vec{q}, \vec{p}) = \left|\vec{p}\right| - n(\vec{q}). \tag{2.28}$$

Hamilton's equations [1] describe the evolution of $\vec{q}$ and $\vec{p}$ in the following first-order ODE system

$$\frac{\mathrm{d}\vec{q}}{\mathrm{d}s} = \frac{\partial \mathcal{H}}{\partial \vec{p}}, \tag{2.29a}$$

$$\frac{\mathrm{d}\vec{p}}{\mathrm{d}s} = -\frac{\partial \mathcal{H}}{\partial \vec{q}}. \tag{2.29b}$$

By taking the partial derivatives, the system can be written as

$$\frac{\mathrm{d}\vec{q}}{\mathrm{d}s} = \frac{\vec{p}}{n}, \tag{2.30a}$$

$$\frac{\mathrm{d}\vec{p}}{\mathrm{d}s} = \frac{\partial n}{\partial \vec{q}}. \tag{2.30b}$$

The Hamilton's equations (2.30) are a reformulation of the ray equation (2.24) as a first order ODE system, with $\vec{p} = n\frac{\mathrm{d}\vec{q}}{\mathrm{d}s}$. Note that from relation (2.25b) we derived that $\left|\vec{p}\right| = n$. This has the important meaning that the momentum vector $\vec{p}$ lies on the Descartes' sphere [95], i.e., a sphere with radius n.

Often we do not want to work with the arc length s as parameter and instead a more suitable choice is to parametrise the problem using one of the position coordinates, for instance, if we know that all light will propagate in a certain direction along an optical axis. In our case, we will use the third component of the position vector $\vec{q} \in \mathbb{R}^3$ as an evolution coordinate, which is denoted as z. The position vector can thus be written as

$$\vec{q} = \begin{pmatrix} \boldsymbol{q} \\ z \end{pmatrix}, \tag{2.31}$$

with $\boldsymbol{q} \in \mathbb{R}^2$. Furthermore, we split the momentum vector into its first two components $\boldsymbol{p} \in \mathbb{R}^2$ and its third component p_z. For the third component p_z

we can use that the momentum vector has a fixed length $\left|\vec{p}\right| = n$. That is we write the momentum vector as

$$\vec{p} = \begin{pmatrix} \boldsymbol{p} \\ p_z \end{pmatrix} = \begin{pmatrix} \boldsymbol{p} \\ \sigma\sqrt{n^2 - |\boldsymbol{p}|^2} \end{pmatrix}, \tag{2.32}$$

with $\sigma \in \{-1, 1\}$ denoting the sign of p_z, i.e., $\sigma = 1$ when $p_z \geq 0$ and $\sigma = -1$ when $p_z < 0$.

Hamilton's equations (2.30) describe the evolution of $(\vec{q}, \vec{p})$ as a function of the arc length s. With the reparametrisation in terms of z one can use the chain rule to find the evolution of these quantities in terms of z, i.e.,

$$\frac{\mathrm{d}\boldsymbol{q}}{\mathrm{d}z} = \frac{\mathrm{d}\boldsymbol{q}}{\mathrm{d}s}\frac{\mathrm{d}s}{\mathrm{d}z},$$
$$\frac{\mathrm{d}\boldsymbol{p}}{\mathrm{d}z} = \frac{\mathrm{d}\boldsymbol{p}}{\mathrm{d}s}\frac{\mathrm{d}s}{\mathrm{d}z}.$$

From equation (2.30a) one already knows $\frac{\mathrm{d}z}{\mathrm{d}s} = p_z/n$, so that $\frac{\mathrm{d}s}{\mathrm{d}z} = n/p_z$ provided $p_z \neq 0$. Together with the other expressions in (2.30) one quickly finds

$$\frac{\mathrm{d}\boldsymbol{q}}{\mathrm{d}z} = \frac{\boldsymbol{p}}{p_z},$$
$$\frac{\mathrm{d}\boldsymbol{p}}{\mathrm{d}z} = \frac{n}{p_z}\frac{\partial n}{\partial \boldsymbol{q}}.$$

Now by using the expression for p_z given in (2.32), we can write the evolution of $(\boldsymbol{q}, \boldsymbol{p})$ as the following Hamiltonian system

$$\frac{\mathrm{d}\boldsymbol{q}}{\mathrm{d}z} = \frac{\partial H}{\partial \boldsymbol{p}}, \tag{2.33a}$$
$$\frac{\mathrm{d}\boldsymbol{p}}{\mathrm{d}z} = -\frac{\partial H}{\partial \boldsymbol{q}}, \tag{2.33b}$$

with the Hamiltonian H given by

$$H(z, \boldsymbol{q}, \boldsymbol{p}) = -\sigma\sqrt{n(z, \boldsymbol{q})^2 - |\boldsymbol{p}|^2}. \tag{2.33c}$$

Taking the partial derivatives in Hamilton's equations leads to

$$\frac{\mathrm{d}\boldsymbol{q}}{\mathrm{d}z} = \frac{1}{\sigma\sqrt{n^2 - |\boldsymbol{p}|^2}}\boldsymbol{p}, \tag{2.34a}$$

$$\frac{\mathrm{d}\boldsymbol{p}}{\mathrm{d}z} = \frac{1}{\sigma\sqrt{n^2 - |\boldsymbol{p}|^2}} n\frac{\partial n}{\partial \boldsymbol{q}}. \tag{2.34b}$$

In the Hamiltonian system (2.33) and the definition of p_z the term σ describes the direction of light rays with respect to the z-axis, i.e., $\sigma = -1$ describes backward rays and $\sigma = 1$ describes forward rays.

2.3 Liouville's equation

In the previous sections, we have described the transfer of luminous flux between surfaces and described the evolution of a single light ray in terms of a Hamiltonian system. Here, both parts will be connected by describing the evolution of the basic luminance as a function of the z-coordinate, which will lead to Liouville's equation.

As mentioned before, the momentum vector $\vec{p}$ has a fixed length equal to the (local) refractive index n. This in turn means that the collection of all momentum vectors with fixed length n lies on the Descartes' sphere with radius n [95]. We denote the d-dimensional unit sphere as $\mathrm{S}^d \subset \mathbb{R}^{d+1}$ and the sphere with radius n as $\mathrm{S}^d(n)$, so that $\vec{p} \in \mathrm{S}^2(n)$. The restriction of the momentum vector to a sphere means the momentum vector can easily be expressed in terms of spherical coordinates, i.e.,

$$\vec{p} = (\boldsymbol{p}, p_z) = n\,(\sin\theta\cos\varphi, \sin\theta\sin\varphi, \cos\theta)\,, \tag{2.35}$$

where θ represents the polar angle, relating the direction of a light ray with respect to the z-axis, and φ the azimuthal angle for describing the direction in the $\boldsymbol{q}$-plane. The Jacobian determinant of $\boldsymbol{p}$ with respect to the polar and azimuthal angles θ and φ, can be computed as

$$\mathrm{d}\boldsymbol{p} = \mathrm{d}p_0\mathrm{d}p_1 = \det\left(\frac{\partial(p_0, p_1)}{\partial(\theta, \varphi)}\right)\mathrm{d}\theta\mathrm{d}\varphi = n^2\cos\theta\,\sin\theta\mathrm{d}\theta\mathrm{d}\varphi = n^2\cos\theta\mathrm{d}\omega, \tag{2.36}$$

with $\mathrm{d}\omega = \sin\theta\mathrm{d}\theta\mathrm{d}\varphi$. By noting that the differential area can be written as $\mathrm{d}A = \mathrm{d}\boldsymbol{q} = \mathrm{d}q_0\mathrm{d}q_1$ and using (2.36) we can write an element of étendue (2.5) as [18]

$$\mathrm{d}\mathcal{U} = \mathrm{d}\boldsymbol{q}\,\mathrm{d}\boldsymbol{p}, \tag{2.37}$$

i.e., relation (2.37) describes étendue in terms of a volume in phase space. Here phase space refers to the collection of all positions $\boldsymbol{q}$ and momenta $\boldsymbol{p}$.

At this point it is important to further specify what phase space looks like. For the position we can take $\vec{q}$ in $\mathbb{R}^3$ or in a subset of $\mathbb{R}^3$. For the momentum vector we have $\vec{p} \in \mathrm{S}^2(n)$, so that the momentum $\boldsymbol{p}$ is restricted by $|\boldsymbol{p}| \leq n$, i.e., $\boldsymbol{p}$ lies in a disc. Furthermore, the collection of momenta $\boldsymbol{p}$ lies in one of the two discs describing either forward ($\sigma = 1$) or backward ($\sigma = -1$) rays. At a

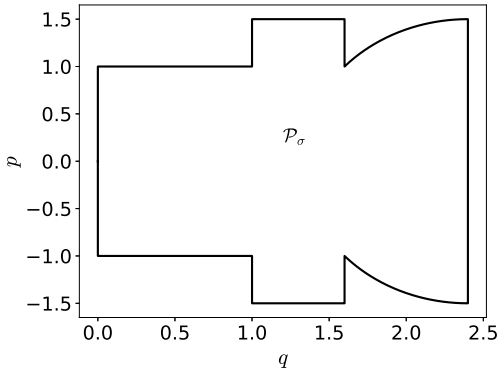


Figure 2.3: Sketch of two-dimensional phase space domain $\mathcal{P}_\sigma$ described by $q \in [0, 2.4]$ and $p \in [-n(q), n(q)]$.

plane $z = \text{const}$, the collection of all positions $\boldsymbol{q}$ and momenta on either disk combine to the four-dimensional phase space domain $\mathcal{P}_\sigma$ for either forward rays ($\sigma = 1$) or backward rays ($\sigma = -1$). An example of a two-dimensional phase space domain, with $\boldsymbol{q}$ and $\boldsymbol{p}$ replaced by scalars, is shown in Figure 2.3.

The σ-subscript in $\mathcal{P}_\sigma$ is there to stress the fact that light rays can jump from one phase space domain to the other. For example, a flat reflective mirror at a fixed z-value will cause light rays to change from forward to backward propagating, and vice versa. The change in σ describes a jump from one phase space domain to the other. In what follows, if we refer to a symbol for just forward-propagating light then we will use a subscript f, e.g., $\mathcal{P}_\mathrm{f}$, rather than its value. Similarly, for backward-propagating light a subscript b is used, e.g., $\mathcal{P}_\mathrm{b}$.

In Section 2.1 it was shown that the basic luminance of a beam of light remains constant when the refractive index field is homogeneous or whenever the beam of light is refracted. Furthermore, étendue was also conserved in these cases. As a reminder, this was all shown with the assumption that there are no losses due to, e.g., scattering, Fresnel reflections or absorption. These properties, the conservation of étendue and the basic luminance invariance, can be generalised by using results from Hamiltonian optics. First, the flow generated by Hamilton's equations describe symplectic transformations which means that a volume element of phase space $\mathrm{d}\mathcal{U} = \mathrm{d}\boldsymbol{q}\mathrm{d}\boldsymbol{p}$ remains constant [1]. The luminous flux $\mathrm{d}\Phi$ is related to the basic luminance ρ_σ defined on phase space $\mathcal{P}_\sigma$ by $\mathrm{d}\Phi = \rho_\sigma \, \mathrm{d}\mathcal{U}$, where again σ is used to emphasise the propagation direction. Second, if the beam of light is propagated along the z-axis over some distance, then in the absence of losses, $\mathrm{d}\Phi$ remains constant. Since the phase space volume element $\mathrm{d}\mathcal{U}$ remains constant, ρ_σ must remain invariant.

Whenever ρ_σ is sufficiently smooth one can write this invariance of ρ_σ as follows

$$\frac{\mathrm{d}}{\mathrm{d}z}\rho_\sigma(z, \boldsymbol{q}(z), \boldsymbol{p}(z)) = 0. \tag{2.38}$$

Assuming sufficient smoothness, one can derive Liouville's equation by taking the total $\frac{\mathrm{d}}{\mathrm{d}z}$-derivative in (2.38), yielding

$$\frac{\partial \rho_\sigma}{\partial z} + \frac{\partial H}{\partial \boldsymbol{p}} \cdot \frac{\partial \rho_\sigma}{\partial \boldsymbol{q}} - \frac{\partial H}{\partial \boldsymbol{q}} \cdot \frac{\partial \rho_\sigma}{\partial \boldsymbol{p}} = 0, \tag{2.39}$$

where we have made use of Hamilton's equations (2.33). Liouville's equation is a linear hyperbolic partial differential equation (PDE). The method of characteristics [65] is a useful tool that turns a hyperbolic PDE into a system of ODEs along characteristic curves. Here, we actually started with describing the characteristic curves to derive Liouville's equation. The location of a characteristic curve is given by $(\boldsymbol{q}(z), \boldsymbol{p}(z))$ which satisfies the Hamilton's equations (2.33), and the value along the curve satisfies (2.38) so that the basic luminance ρ_σ along a characteristic remains constant. Note that a light ray coincides with characteristic curves. Furthermore, we will also refer to characteristic curves as light rays even if the basic luminance has a zero value. This is important, since regions in phase space where the basic luminance has a zero value are still transported according to Liouville's equation. Moreover, the volumes of these regions do not shrink or expand by the principle of conservation of phase space volume.

The advective form of Liouville's equation (2.39) can be transformed to a conservative form, i.e.,

$$\frac{\partial \rho_\sigma}{\partial z} + \nabla \cdot (\rho_\sigma \boldsymbol{u}) = 0 \tag{2.40a}$$

with the velocity field $\boldsymbol{u}$ defined by

$$\boldsymbol{u} = \begin{pmatrix} \frac{\partial H}{\partial \boldsymbol{p}} \\ -\frac{\partial H}{\partial \boldsymbol{q}} \end{pmatrix} = \frac{1}{\sigma\sqrt{n^2 - |\boldsymbol{p}|^2}} \begin{pmatrix} \boldsymbol{p} \\ n\frac{\partial n}{\partial \boldsymbol{q}} \end{pmatrix}, \tag{2.40b}$$

where we have defined $\nabla = \left(\frac{\partial}{\partial \boldsymbol{q}}, \frac{\partial}{\partial \boldsymbol{p}}\right)$ and we have used that the velocity field $\boldsymbol{u}$ is divergence-free. That latter property holds as

$$\nabla \cdot \boldsymbol{u} = \frac{\partial}{\partial \boldsymbol{q}} \cdot \frac{\partial H}{\partial \boldsymbol{p}} - \frac{\partial}{\partial \boldsymbol{p}} \cdot \frac{\partial H}{\partial \boldsymbol{q}} = 0, \tag{2.41}$$

where the order of the differential operators can be interchanged when H is sufficiently smooth.

Note that Liouville's equation describes the evolution of the basic luminance on phase space, where phase space in general is not some straightforward domain. Phase space is not just simply a Cartesian product between a position space and a momentum space, unless $n = \text{const}$. Additionally, if the refractive index field depends on z then phase space changes as a function of z. Specifically, for an optical interface where its location changes as a function of z, the optical interface manifests itself as a moving boundary in phase space when treating z as the evolution coordinate.

At an optical interface the Hamiltonian H is discontinuous and therefore (2.38) is not valid at an optical interface. However, the basic luminance does remain invariant, in the absence of losses, even when light is reflected or refracted; see Section 2.1. Consequently, at an optical interface we enforce invariance of the basic luminance together with Snell's law of refraction or the law of specular reflection. This is expressed in the following jump condition

$$\rho_{\sigma(z^+)}(z^+, \boldsymbol{q}(z^+), \boldsymbol{p}(z^+)) = \rho_{\sigma(z^-)}(z^-, \boldsymbol{q}(z^-), \boldsymbol{p}(z^-)), \tag{2.42a}$$

where we explicitly denote σ as $\sigma(z^\pm)$ since it can change, and the superscript $\pm$ denotes one-sided limits towards the optical interface that correspond to incident and outgoing light for $-$ and $+$, respectively. We compute the full momentum vector $(\boldsymbol{p}, p_z)(z^+)$ as

$$(\boldsymbol{p}, p_z)(z^+) = \mathcal{S}\Big((\boldsymbol{p}, p_z)(z^-); n_0, n_1, \vec{\nu}\Big) \text{ and } \operatorname{sgn} p_z(z^+) = \sigma(z^+). \tag{2.42b}$$

In (2.42a)-(2.42b) we explicitly denote the sign of p_z with $\sigma(z^\pm)$. The change in momentum (2.42b) is described by vectorial versions of the law of specular reflection and Snell's law of refraction, which depend on the refractive indices of the incident and transmitted media denoted by n_0 and n_1, respectively, and the surface unit normal $\vec{\nu} \in \mathrm{S}^2$ at the point $(\boldsymbol{q}(z^-), z^-)$. To be explicit, in equation (2.42b) the function $\mathcal{S}$ can either describe refraction or specular reflection, relating the incident momentum $\vec{\imath} = (\boldsymbol{p}, p_z) \in \mathrm{S}^2(n_0)$ to an outgoing momentum as follows

$$\mathcal{S}(\vec{\imath}; n_0, n_1, \vec{\nu}) = \begin{cases} \mathcal{S}_{\mathrm{R}} = \vec{\imath} - 2\psi\vec{\nu} & \text{if } \delta \le 0, \\ \mathcal{S}_{\mathrm{T}} = \vec{\imath} - (\psi + \sqrt{\delta})\vec{\nu} & \text{if } \delta > 0, \end{cases} \tag{2.43a}$$

with

$$\psi = \vec{\imath} \cdot \vec{\nu} \text{ and } \delta = n_1^2 - n_0^2 + \psi^2. \tag{2.43b}$$

The sign of the normal should be taken such that $\psi \le 0$, i.e., $\vec{\nu}$ points towards the medium of the incident ray. In (2.43) the incident light ray is either subject

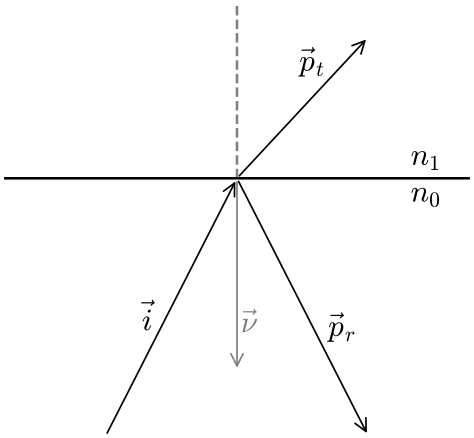


Figure 2.4: Reflection and refraction at an optical interface. Here $\vec{p}_r = \mathcal{S}_{\mathrm{R}}(\vec{i}; n_0, n_1, \vec{\nu})$ and $\vec{p}_t = \mathcal{S}_{\mathrm{T}}(\vec{i}; n_0, n_1, \vec{\nu})$.

to reflection or refraction depending on the sign of δ, where in the case of $\delta \leq 0$ reflection occurs and is referred to as total internal reflection. The expressions for $\mathcal{S}_{\mathrm{R}}$ and $\mathcal{S}_{\mathrm{T}}$ are nothing new and can, for example, be found in [18].

In the definition of $\mathcal{S}$ we have specified the functions for reflection and refraction, sometimes called transmission, denoted by $\mathcal{S}_{\mathrm{R}}$ and $\mathcal{S}_{\mathrm{T}}$, respectively. These are both depicted in Figure 2.4. Reflection transforms quantities on a Descartes' sphere, that is $\mathcal{S}_{\mathrm{R}} : \mathrm{S}^2(n_0) \to \mathrm{S}^2(n_0)$. For refraction $\mathcal{S}_{\mathrm{T}}$ it can happen that $\delta < 0$ so that the result yields complex numbers, whereas for $\delta \geq 0$ it takes a momentum from $\mathrm{S}^2(n_0)$ and returns a momentum on $\mathrm{S}^2(n_1)$. If one wishes to model a perfect reflecting mirror, then one can use the jump condition (2.42) with $\mathcal{S}$ replaced by $\mathcal{S}_{\mathrm{R}}$ in expression (2.42b).

We will often shorten the notation to write $\mathcal{S}(\vec{p}) = \mathcal{S}(\vec{p}; n_0, n_1, \vec{\nu})$, and similarly for $\mathcal{S}_{\mathrm{R}}$ and $\mathcal{S}_{\mathrm{T}}$, where the refractive indices and normal should be clear from the context. Moreover, we will need reflection and refraction in reversed directions, which we denote by $\mathcal{S}^{-1}$, i.e., for reflection the expression reads

$$\mathcal{S}_{\mathrm{R}}^{-1}(\vec{p}; n_0, n_1, \vec{\nu}) = -\mathcal{S}_{\mathrm{R}}(-\vec{p}; n_0, n_1, \vec{\nu}), \tag{2.44a}$$

and for refraction [92]

$$\mathcal{S}_{\mathrm{T}}^{-1}(\vec{p}; n_0, n_1, \vec{\nu}) = -\mathcal{S}_{\mathrm{T}}(-\vec{p}; n_1, n_0, -\vec{\nu}). \tag{2.44b}$$

Here the incident momentum can be computed given an outgoing momentum, i.e.,

$$\vec{p}_r = \mathcal{S}_{\mathrm{R}}(\vec{i}; n_0, n_1, \vec{\nu}) \;\Rightarrow\; -\vec{i} = \mathcal{S}_{\mathrm{R}}(-\vec{p}_r; n_0, n_1, \vec{\nu}),$$

and

$$\vec{p}_t = \mathcal{S}_{\mathrm{T}}(\vec{i}; n_0, n_1, \vec{\nu}) \;\Rightarrow\; -\vec{i} = \mathcal{S}_{\mathrm{T}}(-\vec{p}_t; n_1, n_0, -\vec{\nu}).$$

For completeness, we remark that the relations (2.42) and (2.43) state that light can only be either fully reflected or fully refracted, that is, there are no partial reflections, so Fresnel reflections are not taken into account in (2.42). Fresnel reflections are discussed in the next section.

As discussed before, the basic luminance remains constant along characteristic curves of Liouville's equation. At an optical interface these characteristics change discontinuously according to (2.42b)-(2.43) where by (2.42a) the basic luminance remains constant. Solving Hamilton's equations and applying (2.42b)-(2.43) at an optical interface, is commonly referred to as ray tracing [18]. For simple optical systems it can be manageable to trace a light ray from a certain target at, e.g., $z = Z$, to the source at $z = 0$, such that we can determine the exact solution to Liouville's equation at $z = Z$. In particular, this will be applied to verify the numerical methods used in later chapters that solve Liouville's equation.

As shown in Section 2.1 from the basic luminance, both the illuminance and the luminous intensity can be determined. Assume that the basic luminance is known on phase space at $z = \text{const}$. Then, from relation (2.17) for $\mathrm{d}E$ and relation (2.36), the illuminance can be computed by integrating over all momenta $\boldsymbol{p} \in P$, i.e.,

$$E(z, \boldsymbol{q}) = \int_P \rho_\sigma(z, \boldsymbol{q}, \boldsymbol{p})\, \mathrm{d}\boldsymbol{p}, \tag{2.45}$$

where P denotes the momentum space. Note that we use either the forward or the backward basic luminance distribution to compute the illuminance and not both. This makes sense physically because in non-imaging optics one for example measures the illuminance on a table's surface, where only one direction is relevant. From relation (2.19) for $\mathrm{d}I$ and relation (2.35), the luminous intensity can be computed by integrating over all positions $\boldsymbol{q} \in Q$, i.e.,

$$I(z, \boldsymbol{p}) = \int_Q \rho_\sigma(z, \boldsymbol{q}, \boldsymbol{p}) p_z(z, \boldsymbol{q}, \boldsymbol{p}) n(z, \boldsymbol{q})\, \mathrm{d}\boldsymbol{q}, \tag{2.46}$$

where Q denotes the position space. The total amount of luminous flux Φ can be simply computed by integrating over all positions and momenta, i.e.,

$$\Phi(z) = \int_{\mathcal{P}_\sigma} \rho_\sigma(z, \boldsymbol{q}, \boldsymbol{p})\, \mathrm{d}\mathcal{U}\,; \tag{2.47}$$

cf. (2.14). With these definitions, the main quantities of interest in optics can thus be easily computed if we can solve Liouville's equation to obtain the basic luminance.

2.4 Jump conditions and maximum principle

The jump condition (2.42) describes how the basic luminance is redistributed. This jump condition is a simplified model, as a more realistic model would be to use Fresnel reflections. In Fresnel reflections a single light ray is split into two light rays upon striking an optical interface. One light ray corresponds to a reflected light ray and the other to a transmitted light ray, and the incident basic luminance is distributed among both outgoing rays. In this section, we study the jump condition (2.42) and present a new jump condition that describes Fresnel reflections. Moreover, for both jump conditions we will show that no new maxima of the basic luminance can be generated at an optical interface and how this relates to a maximum principle for Liouville's equation.

The notation used in describing the jump condition (2.42) is quite convoluted. Therefore, we simplify the notation by omitting the position as it remains constant at an optical interface and we make use of the full momentum vector to succinctly write the jump condition as

$$\rho^+(\vec{p}^+) = \rho^-(\vec{p}^-). \tag{2.48}$$

Here, we simply write $\vec{p}$ as the argument for ρ with the meaning that $\rho(\vec{p})$ should be understood as $\rho_\sigma(\boldsymbol{p})$ with $\vec{p} = (\boldsymbol{p}, \sigma\,|p_z|)$. Furthermore, as before the outgoing momentum $\vec{p}^+$ is computed from the incident momentum $\vec{p}^-$ by

$$\vec{p}^+ = \mathcal{S}(\vec{p}^-). \tag{2.49}$$

Consider Liouville's equation with only forward-propagating light and no optical interfaces, i.e.,

$$\frac{\partial \rho_\mathrm{f}}{\partial z} + \nabla \cdot (\rho_\mathrm{f} \boldsymbol{u}) = 0, \tag{2.50a}$$

with initial condition

$$\rho_\mathrm{f}(0, \boldsymbol{q}, \boldsymbol{p}) = \rho_0(\boldsymbol{q}, \boldsymbol{p}), \tag{2.50b}$$

and zero inflow boundary conditions. In other words, there is only a light source that coincides with $z = 0$, so that $\rho = 0$ at an inflow boundary. The solution to (2.50) satisfies a (strict) maximum principle. Namely, if $\rho_0(\boldsymbol{q}, \boldsymbol{p}) \in [0, \rho_\mathrm{max}]$ for any $(\boldsymbol{q}, \boldsymbol{p})$, with $\rho_\mathrm{max} = \max_{(\boldsymbol{q},\boldsymbol{p})} \rho_0(\boldsymbol{q}, \boldsymbol{p})$, then $\rho_\mathrm{f}(z, \boldsymbol{q}, \boldsymbol{p}) \in [0, \rho_\mathrm{max}]$ for any $(z, \boldsymbol{q}, \boldsymbol{p})$. This fact can be easily derived as the basic luminance remains constant along characteristic curves, see equation (2.38), from which one concludes that the solution is completely determined by the boundary and initial conditions. Since the boundary conditions have zero inflow, the solution

can take on non-zero values only when a characteristic curve connects to the initial condition. As ρ_f remains constant along characteristics, the solution cannot exceed the maximum ρ_{max}.

This result can be generalised to Liouville's equation considering both forward- and backward-propagating light, and an arbitrary number of optical interfaces as follows. Again, we consider zero inflow boundary conditions but now ρ_{max} denotes the maximum over the initial conditions for both forward- and backward-propagating light, i.e., the initial conditions and ρ_{max} read

$$\rho_\sigma(0, \boldsymbol{q}, \boldsymbol{p}) = \rho_{\sigma,0}(\boldsymbol{q}, \boldsymbol{p}) \text{ and } \rho_{max} = \max_{(\sigma, \boldsymbol{q}, \boldsymbol{p})} \rho_{\sigma,0}(\boldsymbol{q}, \boldsymbol{p}). \tag{2.51}$$

The only important difference for deriving the maximum principle compared to the previous case is that now there are optical interfaces. Characteristic curves can still be followed, however, it can happen that a characteristic curve intersects an optical interface, that is, a light ray strikes an optical interface. In that case, we need to apply the jump condition (2.48). The jump condition obviously satisfies a maximum principle, e.g., when $\rho^-(\vec{p}^-) = \rho_{max}$ then $\rho^+(\vec{p}^+) = \rho_{max}$ so that no new maxima can be generated through the jump condition. Whenever an optical interface is hit, we need to follow the new characteristic curve starting from the intersection point and repeat the procedure. The procedure is repeated until either the characteristic curve intersects the boundary of the domain or $z = 0$, so that either the zero inflow boundary condition or the initial conditions provide the value of ρ along the curve. Hence, Liouville's equation with the jump condition (2.48) satisfies a maximum principle.

Fresnel reflections describe a more realistic interaction of light striking an optical interface. When one considers Fresnel reflections light can be either reflected via total internal reflection, or light is partially reflected and partially transmitted. As light is electromagnetic radiation and consists of oscillating electromagnetic waves, the interaction between light and an optical interface depends on the so-called polarisation of the electromagnetic wave. The direction of the electric field describes the polarisation, which can be decomposed in a perpendicular and parallel component with respect to the plane of incidence. The ratio between the reflected and incident electric fields is given by the Fresnel equations for either state of polarisation (perpendicular or parallel). These relations can be derived from the theory of electromagnetism [45, 47].

From the Fresnel equations, reflection coefficients can be derived that describe the fraction of energy that is reflected. The Fresnel reflection coefficients for a non-magnetic medium can be written in terms of the incident and transmitted angles denoted θ_i and θ_t, respectively. The Fresnel reflection coef-

ficients $\mathcal{R}$ for the energy for parallel ($\mathcal{R}_\parallel$) and perpendicular ($\mathcal{R}_\perp$) polarisation read [45, 47]

$$\mathcal{R}_\parallel = \left| \frac{n_0 \cos\theta_\mathrm{t} - n_1 \cos\theta_\mathrm{i}}{n_0 \cos\theta_\mathrm{t} + n_1 \cos\theta_\mathrm{i}} \right|^2, \tag{2.52a}$$

$$\mathcal{R}_\perp = \left| \frac{n_0 \cos\theta_\mathrm{i} - n_1 \cos\theta_\mathrm{t}}{n_0 \cos\theta_\mathrm{i} + n_1 \cos\theta_\mathrm{t}} \right|^2, \tag{2.52b}$$

where n_0 and n_1 denote the refractive indices of the incident and transmitted media, respectively. Note that the argument of $|\cdot|$ can be a complex number and, consequently, the Fresnel reflection coefficients yield the value 1 when the imaginary part is non-zero. This happens when there is no real solution to Snell's law (2.8) and describes the case of total internal reflection.

The Fresnel reflection coefficients can be rewritten in terms of the incident momentum vector $\vec{i}$ and normal vector $\vec{\nu}$. In expression (2.43) the incident angle is related to $\psi = \vec{i} \cdot \vec{\nu} \leq 0$ by $\psi = -n_0 \cos\theta_\mathrm{i}$. Moreover, by Snell's law (2.8) we have that

$$n_0 \sqrt{1 - \cos^2\theta_\mathrm{i}} = n_1 \sqrt{1 - \cos^2\theta_\mathrm{t}},$$

so that δ in (2.43) can be expressed in terms of θ_t as

$$\begin{aligned} \delta &= n_1^2 - n_0^2 + \psi^2 = n_1^2 - n_0^2(1 - \cos^2\theta_\mathrm{i}) \\ &= n_1^2 - n_1^2(1 - \cos^2\theta_\mathrm{t}) = n_1^2 \cos^2\theta_\mathrm{t}. \end{aligned}$$

Using the relations for ψ and δ, the Fresnel reflection coefficients can be expressed as

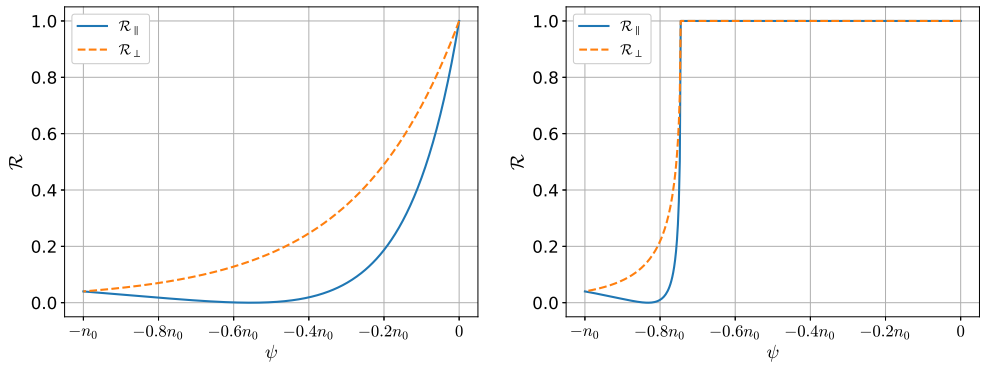
$$\mathcal{R}_\parallel(\vec{i}; n_0, n_1, \vec{\nu}) = \left| \frac{n_1^2 \psi + n_0^2 \sqrt{\delta}}{n_1^2 \psi - n_0^2 \sqrt{\delta}} \right|^2, \tag{2.53a}$$

$$\mathcal{R}_\perp(\vec{i}; n_0, n_1, \vec{\nu}) = \left| \frac{\psi + \sqrt{\delta}}{\psi - \sqrt{\delta}} \right|^2, \tag{2.53b}$$

with

$$\psi = \vec{i} \cdot \vec{\nu} \text{ and } \delta = n_1^2 - n_0^2 + \psi^2, \tag{2.53c}$$

and $\vec{\nu}$ the unit normal vector. Recall that the sign of the normal vector is chosen such that $\psi \leq 0$. The functions $\mathcal{R}_\parallel$ and $\mathcal{R}_\perp$ are depicted in Figure 2.5, where in the left panel $n_0 < n_1$, with $n_0 = 1$ and $n_1 = 1.5$, and in the right panel $n_0 > n_1$, with $n_0 = 1.5$ and $n_1 = 1$. The function $\mathcal{R}_\parallel$ attains a value of 0 at a certain angle of incident light, which is known as the Brewster's angle. All incident light at this specific angle is entirely transmitted.



(a) $n_0 = 1$ and $n_1 = 1.5$.

(b) $n_0 = 1.5$ and $n_0 = 1$.

Figure 2.5: The Fresnel reflection coefficients as a function of $\psi = \vec{i} \cdot \vec{\nu}$.

Armed with the Fresnel reflection coefficients we can now write what happens to ρ at an optical interface. For an incident light ray with momentum $\vec{i}$, the basic luminance is split into a reflected part with value $\mathcal{R}(\vec{i}; n_0, n_1, \vec{\nu})\rho^-(\vec{i})$ and a transmitted part with value $(1 - \mathcal{R}(\vec{i}; n_0, n_1, \vec{\nu}))\rho^-(\vec{i})$. For the jump condition we are interested in how the basic luminance combines multiple incident values at a given outgoing momentum vector $\vec{p}$. The Fresnel reflection equivalent of the jump condition (2.48) reads

$$\rho^+(\vec{p}) = \mathcal{R}(\vec{i}_\mathrm{r}; n_1, n_0, \vec{\nu})\rho^-(\vec{i}_\mathrm{r}) + \Big(1 - \mathcal{R}(\vec{i}_\mathrm{t}; n_0, n_1, -\vec{\nu})\Big)\rho^-(\vec{i}_\mathrm{t}), \tag{2.54a}$$

with

$$\vec{i}_\mathrm{r} = \mathcal{S}_\mathrm{R}^{-1}(\vec{p}; n_1, n_0, \vec{\nu}) \quad \text{and} \quad \vec{i}_\mathrm{t} = \mathcal{S}_\mathrm{T}^{-1}(\vec{p}; n_0, n_1, -\vec{\nu}). \tag{2.54b}$$

Here $\vec{i}_\mathrm{r}$ and $\vec{i}_\mathrm{t}$ describe the incident momentum vectors that after reflection or transmission have momentum $\vec{p}$; see Figure 2.6. Moreover, take note that in the computation of $\vec{i}_\mathrm{t}$ and $\mathcal{R}(\vec{i}_\mathrm{t})$ the normals have a minus sign due to the sign convention.

Now one might wonder, if in relation (2.54) $\rho^-(\vec{i}_\mathrm{r}) = \rho_\mathrm{max}$ and $\rho^-(\vec{i}_\mathrm{t}) = \rho_\mathrm{max}$ whether $\rho^+(\vec{p})$ can take on a value larger than ρ_max. In what follows we will show that relation (2.54) describes a convex combination of $\rho^-(\vec{i}_\mathrm{r})$ and $\rho^-(\vec{i}_\mathrm{t})$ hence, $\rho^+(\vec{p}) \in [0, \rho_\mathrm{max}]$ and thus no new maxima can be generated. To show that the relation describes a convex combination, we directly note that $\mathcal{R} \geq 0$. What remains to show is that the reflection coefficients sum to 1.

We start by computing $\mathcal{R}(\vec{i}_\mathrm{r}; n_1, n_0, \vec{\nu})$. From relation (2.44a) for $\mathcal{S}_\mathrm{R}^{-1}$ we

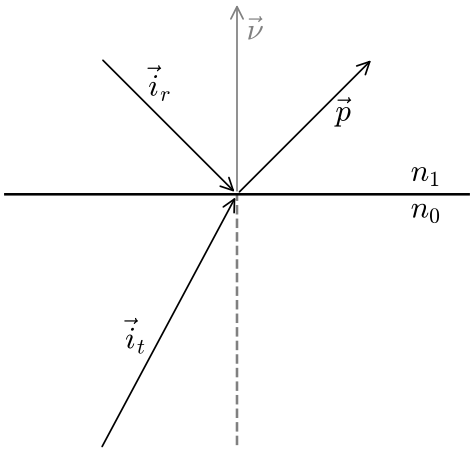


Figure 2.6: Incident light rays with momenta $\vec{i}_r$ and $\vec{i}_t$ have after reflection and refraction, respectively, the momentum $\vec{p}$.

find that

$$\begin{aligned}\vec{i}_{\mathrm{r}} = -\mathcal{S}_{\mathrm{R}}(-\vec{p}; n_1, n_0, \vec{\nu}) &= -\left[-\vec{p} - 2(-\vec{p}\cdot\vec{\nu})\vec{\nu}\right] \\ &= \vec{p} - 2(\vec{p}\cdot\vec{\nu})\vec{\nu},\end{aligned} \tag{2.55}$$

so that

$$\vec{i}_{\mathrm{r}}\cdot\vec{\nu} = -\vec{p}\cdot\vec{\nu} = -\phi, \tag{2.56}$$

where $\phi = \vec{p}\cdot\vec{\nu} \geq 0$ by the sign convention for the normal. The reflection coefficients for the reflected part then read

$$\mathcal{R}_{\parallel}(\vec{i}_{\mathrm{r}}; n_1, n_0, \vec{\nu}) = \left| \frac{-n_0^2\phi + n_1^2\sqrt{n_0^2 - n_1^2 + \phi^2}}{-n_0^2\phi - n_1^2\sqrt{n_0^2 - n_1^2 + \phi^2}} \right|^2, \tag{2.57a}$$

$$\mathcal{R}_{\perp}(\vec{i}_{\mathrm{r}}; n_1, n_0, \vec{\nu}) = \left| \frac{-\phi + \sqrt{n_0^2 - n_1^2 + \phi^2}}{-\phi - \sqrt{n_0^2 - n_1^2 + \phi^2}} \right|^2. \tag{2.57b}$$

Next, we compute $\mathcal{R}(\vec{i}_{\mathrm{t}}; n_0, n_1, -\vec{\nu})$. From the relation (2.44b) for $\mathcal{S}_{\mathrm{T}}^{-1}$ we find that

$$\begin{aligned}\vec{i}_{\mathrm{t}} = -\mathcal{S}_{\mathrm{T}}(-\vec{p}; n_1, n_0, \vec{\nu}) &= -\left[-\vec{p} - \left(-\vec{p}\cdot\vec{\nu} + \sqrt{n_0^2 - n_1^2 + (-\vec{p}\cdot\vec{\nu})^2}\right)\vec{\nu}\right] \\ &= \vec{p} + \left(-\phi + \sqrt{n_0^2 - n_1^2 + \phi^2}\right)\vec{\nu},\end{aligned} \tag{2.58}$$

so that

$$-\vec{i}_t \cdot \vec{\nu} = -\phi + \phi - \sqrt{n_0^2 - n_1^2 + \phi^2} = -\sqrt{n_0^2 - n_1^2 + \phi^2}. \tag{2.59}$$

Furthermore, to compute the reflection coefficients we have

$$\sqrt{\delta} = \sqrt{n_1^2 - n_0^2 + (\vec{i}_t \cdot \vec{\nu})^2} = \sqrt{n_1^2 - n_0^2 + n_0^2 - n_1^2 + \phi^2} = \phi,$$

where the last step follows from $\phi \geq 0$. The reflection coefficients for the transmitted part can be written as

$$\mathcal{R}_{\parallel}(\vec{i}_t; n_0, n_1, -\vec{\nu}) = \left| \frac{-n_0^2\phi + n_1^2\sqrt{n_0^2 - n_1^2 + \phi^2}}{-n_0^2\phi - n_1^2\sqrt{n_0^2 - n_1^2 + \phi^2}} \right|^2, \tag{2.60a}$$

$$\mathcal{R}_{\perp}(\vec{i}_t; n_0, n_1, -\vec{\nu}) = \left| \frac{-\phi + \sqrt{n_0^2 - n_1^2 + \phi^2}}{-\phi - \sqrt{n_0^2 - n_1^2 + \phi^2}} \right|^2, \tag{2.60b}$$

where in the numerator within $|\cdot|$ we have multiplied by -1. Clearly, expressions (2.57) and (2.60) are equivalent, hence, the following relation holds

$$\mathcal{R}(\vec{i}_r; n_1, n_0, \vec{\nu}) + \left(1 - \mathcal{R}(\vec{i}_t; n_0, n_1, -\vec{\nu})\right) = 1, \tag{2.61}$$

proving that the jump condition (2.54) is indeed a convex combination.

To prove that there is a maximum principle for Liouville's equation with Fresnel reflections at optical interfaces, we can follow the same steps as before. The only difference is that at an optical interface, the jump condition (2.54) states there are two contributions. Now since the jump condition (2.54) is a convex combination of two basic luminance values no new maxima can be generated at an optical interface. So, we can conclude that also in this case Liouville's equation satisfies a maximum principle.

2.5 Summary

In this chapter the main equation of this thesis was derived, that is Liouville's equation (2.40). Liouville's equation describes the transport of basic luminance in phase space. Here, basic luminance has its origins in non-imaging optics while phase space is related to the position and momentum of a light ray as described by Hamiltonian optics. Specular reflection and refraction at an optical interface are incorporated by a jump condition. When there is no partial reflection, that is, just pure reflection or pure refraction then the jump

condition can be concisely written in the form (2.48). Fresnel reflections can be modelled using the jump condition (2.54).

Although not considered in this thesis, surface scattering, also known as diffuse reflection, can also be modelled via the jump condition. In surface scattering, a single incident light ray is after diffuse reflection turned into a distribution of light rays, where the basic luminance is redistributed among all outgoing light rays. For the jump condition this amounts to taking the basic luminance for an outgoing direction equal to an integral of the basic luminance for a range of incident light rays multiplied by a properly chosen probability density function.

Liouville's equation for three-dimensional optics is described on a four-dimensional phase space domain, which together with the evolution coordinate makes it a five-dimensional problem. If instead we consider two-dimensional optics, then the position vector $\boldsymbol{q}$ and momentum vector $\boldsymbol{p}$ become scalars. Moreover, Liouville's equation then describes the transport of the basic luminance on a two-dimensional phase space domain. In general, solving Liouville's equation analytically, especially for three-dimensional optics, for even a few optical interfaces can already become quite complicated, depending on the shapes of the optical interfaces. Therefore, we will resort to discretisation schemes for numerically approximating the solution. In the next chapter, an introduction is given into to the discretisation methods of interest.

Chapter 3

Discontinuous Galerkin methods in one dimension

In the previous chapter, we derived Liouville's equation and described jump conditions at optical interfaces. Solving Liouville's equation for general optical systems analytically can be very complicated. Hence, we will use discontinuous Galerkin methods to numerically approximate the solution. In this chapter, an introduction to these methods is presented for a one-dimensional transport problem.

To that end, consider the following hyperbolic PDE describing transport of a physical quantity $u = u(t, x)$

$$\frac{\partial u}{\partial t} + \frac{\partial f}{\partial x} = 0, \tag{3.1}$$

with $f = f(t, x, u)$ the flux. The initial condition at $t = 0$ reads $u(0, x) = u_0(x)$ defined on some interval $[X_0, X_1]$. To compute the evolution of u one can use an analytical method such as the method of characteristics. This method works well for simple problems, however, this method can become quite cumbersome to derive analytical expressions especially in higher dimensional settings with complex geometries and/or complicated boundary conditions, as for example in Liouville's equation. One therefore has to resort to computing numerical approximations using suitable discretisation schemes.

There exist many discretisation schemes for solving hyperbolic PDEs such as equation (3.1). For example there are (first-order) upwind finite volume (FV) schemes, weighted essentially non-oscillatory (WENO) schemes and semi-Lagrangian type schemes.

The methods we consider in this work belong to the class of discontinuous Galerkin finite element methods. Discontinuous Galerkin finite element

methods derive their approximations from a weak formulation of the PDE. The weak formulation for the PDE (3.1) on an interval $[x_{k-1/2}, x_{k+1/2}]$ is derived as follows. The PDE is multiplied by a test function $\psi = \psi(x)$ and integrated over the interval $[x_{k-1/2}, x_{k+1/2}]$ leading to

$$\int_{x_{k-1/2}}^{x_{k+1/2}} \left(\frac{\partial u}{\partial t} + \frac{\partial f}{\partial x} \right) \psi \, \mathrm{d}x = 0. \tag{3.2}$$

Taking the time derivative outside the integral and applying integration by parts to the flux term, yields

$$\frac{\mathrm{d}}{\mathrm{d}t} \int_{x_{k-1/2}}^{x_{k+1/2}} u\psi \, \mathrm{d}x = \int_{x_{k-1/2}}^{x_{k+1/2}} f \frac{\mathrm{d}\psi}{\mathrm{d}x} \, \mathrm{d}x - [f\psi]_{x=x_{k-1/2}}^{x_{k+1/2}}. \tag{3.3}$$

Next, an approximation of u on the interval $[x_{k-1/2}, x_{k+1/2}]$ is introduced. A feasible choice is to expand u into a set of basis functions

$$u(t,x) \approx u_{\mathrm{h}}(t,x) = \sum_{i=0}^{N} u_i(t)\phi_i(x), \tag{3.4}$$

where u_{h} denotes the (discrete) approximation to u and $\{\phi_i\}_{i=0}^{N}$ denote the set of basis functions. In a Galerkin method the test function in the weak formulation is taken from the same set of basis functions that are used in the expansion, that is, we require the weak formulation (3.3) to hold for $\psi = \phi_i$ with $i = 0, \ldots, N$.

To further discretise the weak formulation one needs to choose suitable basis functions and choose how to evaluate the integrals. There are many excellent books on discontinuous Galerkin methods where these choices are discussed, such as the book by Hesthaven & Warburton [51] and the book by Kopriva [57]. The latter book discusses the discontinuous Galerkin spectral element method (DGSEM). The DGSEM will be presented in Section 3.2. And in Section 3.3 discontinuous Galerkin methods based on a semi-Lagrangian framework are presented, but first we will illustrate some core concepts that are used in the discontinuous Galerkin methods and describe their implementation details.

3.1 Interpolation, derivatives and integration

In the expansion (3.4) we are presented with a choice of what basis functions to use. One could use for example Legendre polynomials; in this case the coefficients u_i are called modal coefficients. In this work we will be using an

expansion in Lagrange polynomials $\{\ell_i\}_{i=0}^N$, that is the expansion of a function $u = u(x)$ reads

$$u(x) \approx u_{\mathrm{h}}(x) = \sum_{i=0}^{N} u_i \ell_i(x), \tag{3.5}$$

with the Lagrange polynomials defined by

$$\ell_i(x) = \prod_{\substack{j=0 \\ j \neq i}}^{N} \frac{x - x_j}{x_i - x_j}, \tag{3.6}$$

and the nodes $\{x_i\}_{i=0}^N$ will be specified in what follows. The Lagrange polynomials satisfy the following property

$$\ell_i(x_j) = \delta_{ij} = \begin{cases} 1 & \text{if } i = j, \\ 0 & \text{if } i \neq j, \end{cases} \tag{3.7}$$

where δ_{ij} is called the Kronecker delta. If we now evaluate the expansion (3.5) at a node x_j, then we can use property (3.7) to obtain

$$u_{\mathrm{h}}(x_j) = \sum_{i=0}^{N} u_i \ell_i(x_j) = \sum_{i=0}^{N} u_i \delta_{ij} = u_j. \tag{3.8}$$

Hence, with this choice of basis functions the coefficients u_j are called nodal coefficients, where at each node x_j the function $u_{\mathrm{h}}(x)$ takes on the value u_j.

What now if we need to evaluate expansion (3.5) at a value x that does not correspond to a node? From a computational viewpoint the expansion (3.5) is expensive to evaluate, as each Lagrange polynomial (3.6) requires $\mathcal{O}(N)$ flops so that evaluating $u_{\mathrm{h}}(x)$ requires $\mathcal{O}(N^2)$ flops. Fortunately, relation (3.5) can be rewritten into two alternative ways that are more useful for implementation [57]. Consider a polynomial p of degree N written in the Lagrange basis, i.e.,

$$p(x) = \sum_{i=0}^{N} p_i \ell_i(x). \tag{3.9}$$

The first alternate way reads

$$p(x) = \varphi(x) \sum_{i=0}^{N} p_i \frac{\omega_i}{x - x_i}, \tag{3.10}$$

with

$$\varphi(x) = \prod_{i=0}^{N} (x - x_i), \tag{3.11a}$$

$$\omega_i = \prod_{\substack{j=0 \\ j \neq i}}^{N} \frac{1}{x_i - x_j}. \tag{3.11b}$$

The second alternate way can be obtained by setting $p(x) = 1$ in (3.10) such that $p_i = 1$, which leads to the following relation

$$\varphi(x) \sum_{i=0}^{N} \frac{\omega_i}{x - x_i} = 1, \tag{3.12}$$

so that (3.10) can be written as

$$p(x) = \frac{\sum_{i=0}^{N} p_i \dfrac{\omega_i}{x - x_i}}{\sum_{i=0}^{N} \dfrac{\omega_i}{x - x_i}}. \tag{3.13}$$

This latter relation (3.13) is known as the barycentric formula of Lagrange interpolation [9, 57] and the coefficients ω_i are known as the barycentric weights. If we need to evaluate $p(x)$ at many different points, then we can precompute the barycentric weights and store them. After this initial step $p(x)$ can be computed with $\mathcal{O}(N)$ flops using expression (3.13).

In addition to evaluating $p(x)$ at certain points, we will also require derivatives of $p(x)$. The derivative of $p(x)$ at a node x_i reads

$$\frac{\mathrm{d}p}{\mathrm{d}x}(x_i) = \sum_{j=0}^{N} p_j \frac{\mathrm{d}\ell_j}{\mathrm{d}x}(x_i) = \sum_{j=0}^{N} D_{ij} p_j, \tag{3.14}$$

where

$$D_{ij} = \frac{\mathrm{d}}{\mathrm{d}x} \ell_j(x_i), \tag{3.15}$$

defines the coefficients of the first-order derivative matrix $\boldsymbol{D} = \left(D_{ij}\right)$. The coefficients D_{ij} can be expressed in terms of the barycentric weights and the nodal points as follows [57]

$$D_{ij} = \frac{\omega_j}{\omega_i} \frac{1}{x_i - x_j}, \quad i \neq j, \tag{3.16a}$$

and the diagonal elements are given by

$$D_{ii} = -\sum_{\substack{j=0 \\ j\neq i}}^{N} D_{ij}, \tag{3.16b}$$

where the last expression is due to the fact that the derivative of a constant function vanishes, e.g., take $p(x) = 1$ in (3.14). Higher-order derivatives of $p(x)$ can be found by applying the first-order derivative matrix repeatedly, e.g.,

$$\frac{\mathrm{d}^2 p}{\mathrm{d}x^2}(x_i) = \sum_{j,k=0}^{N} D_{ik} D_{kj} p_j = \sum_{j=0}^{N} D_{ij}^{(2)} p_j, \tag{3.17}$$

where $D_{ij}^{(2)}$ represents the coefficients of the second-order derivative matrix. In general, the coefficients of the mth-order derivative matrix $\boldsymbol{D}^{(m)} = \left(D_{ij}^{(m)}\right)$ can be computed as [57]

$$D_{ij}^{(m)} = \frac{m}{x_i - x_j}\left(\frac{\omega_j}{\omega_i} D_{ii}^{(m-1)} - D_{ij}^{(m-1)}\right), \quad i \neq j, \tag{3.18a}$$

with the diagonal elements given by

$$D_{ii}^{(m)} = -\sum_{\substack{j=0 \\ j\neq i}}^{N} D_{ij}^{(m)}. \tag{3.18b}$$

In general, applying an mth-order derivative matrix to coefficients $\{p_j\}_{j=0}^{N}$ can be performed as a matrix-vector multiplication resulting in new coefficients $\{p_j^{(m)}\}_{j=0}^{N}$ that interpolate the mth derivative, that is we can write

$$\frac{\mathrm{d}^m p}{\mathrm{d}x^m}(x) = \sum_{j=0}^{N} p_j \frac{\mathrm{d}^m \ell_j}{\mathrm{d}x^m}(x) = \sum_{j=0}^{N} p_j^{(m)} \ell_j(x), \tag{3.19a}$$

where the latter equality follows because a degree $N - m$ polynomial can be exactly represented in a degree N polynomial basis. Here the coefficients $\{p_i^{(m)}\}_{i=0}^{N}$ are defined by

$$p_i^{(m)} = \sum_{j=0}^{N} D_{ij}^{(m)} p_j, \tag{3.19b}$$

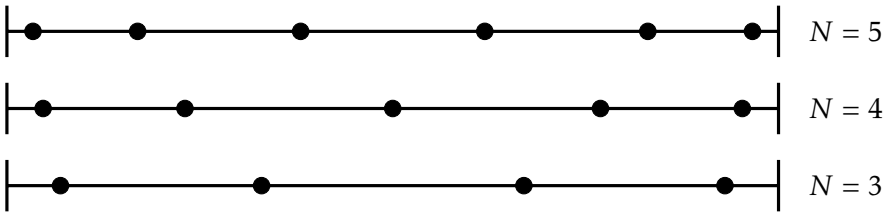


Figure 3.1: Gauss-Legendre quadrature nodes denoted by bullets on the unit interval $E = [0, 1]$.

which can be seen by evaluating (3.19a) at a point x_i. Finally we remark that given a certain point set, we can precompute the derivative matrices and reuse them during computation.

To evaluate the integrals that appear in the weak formulation (3.3) we will employ Gauss-Legendre quadrature. An integral over any finite interval $[a, b]$ can be transformed to an integral over the unit interval $E = [0, 1]$ with a simple affine transformation. The integral of a function g over the unit interval E is approximated with an $(N + 1)$-point Gauss-Legendre quadrature as

$$\int_E g(\xi)\,\mathrm{d}\xi \approx \sum_{n=0}^{N} w_n g(\xi_n), \tag{3.20}$$

where the quadrature rule is defined over the interval E with nodes $\{\xi_i\}_{i=0}^N$ and weights $\{w_i\}_{i=0}^N$. The weights satisfy $w_i > 0$, and the nodes ξ_i are roots of the shifted $(N + 1)$th Legendre polynomial and satisfy $0 < \xi_i < 1$ [14]. Moreover, an $(N + 1)$-point Gauss-Legendre quadrature rule integrates polynomials of degree $2N + 1$ exactly. These Gauss-Legendre quadrature nodes are illustrated in Figure 3.1 for several values of N.

The nodes for the Lagrange polynomials (3.6) have thus far not been specified. For the nodes we take the $(N + 1)$-point Gauss-Legendre quadrature nodes on the unit interval E. With this set of nodes the Lagrange polynomials are in fact orthogonal to each other with respect to the L_2-inner product on E, that is they satisfy

$$\int_E \ell_i(\xi)\ell_j(\xi)\,\mathrm{d}\xi = \sum_{n=0}^{N} w_n \ell_i(\xi_n)\ell_j(\xi_n) = \sum_{n=0}^{N} w_n \delta_{in}\delta_{jn} = w_i \delta_{ij}, \tag{3.21}$$

where the integral is exactly evaluated by $(N + 1)$-point Gauss-Legendre quadrature.

Before moving onto the DGSEM, we discuss a connection between Gauss-Legendre quadrature and barycentric weights that might not be so well-known. In the discussion of the barycentric formula for Lagrange interpolation (3.13)

the computation of all the barycentric weights takes $\mathcal{O}(N^2)$ flops. Wang & Xiang [93] have shown that with the special case of taking Gauss-Legendre quadrature nodes as nodes in the Lagrange polynomials, the barycentric weights can actually be expressed as

$$\omega_j = (-1)^j \sqrt{\left(1 - X_j^2\right) w_j}, \quad \text{with } j = 0, \ldots, N, \tag{3.22}$$

with $X_j = 2\xi_j - 1$ the quadrature nodes on $[-1, 1]$. So, in fact all the barycentric weights together can be computed in $\mathcal{O}(N)$ flops. Therefore, one can even evaluate the Lagrange interpolation polynomial (3.13) in $\mathcal{O}(N)$ flops without any precomputation step, as long as the Gauss-Legendre quadrature nodes and weights are available.

3.2 Discontinuous Galerkin spectral element method

In the DGSEM the domain of interest $[X_0, X_1]$ is partitioned into N_{elements} non-overlapping intervals by placing a set of grid points $X_0 = x_{-1/2} < x_{1/2} < \ldots < x_{N_{\text{elements}}+1/2} = X_1$ on the domain. The kth interval is defined as $[x_{k-1/2}, x_{k+1/2}]$. These intervals are referred to as elements or cells, with $k = 1, \ldots, N_{\text{elements}}$. Every element can be mapped from the unit reference interval $E = [0, 1]$ to the element's respective interval by the following transformation $x(\xi) = x_{k-1/2} + \xi \Delta x_k$ with $\xi \in E$ and $\Delta x_k = x_{k+1/2} - x_{k-1/2}$. This transforms the PDE (3.1) to

$$\frac{\partial u}{\partial t} + \frac{1}{\Delta x_k} \frac{\partial f}{\partial \xi} = 0,$$

which we rewrite to

$$\mathcal{J} \frac{\partial u}{\partial t} + \frac{\partial f}{\partial \xi} = 0, \tag{3.23}$$

where we have multiplied the PDE by Δx_k and $\mathcal{J} = \Delta x_k$ denotes the Jacobian of the transformation. The discontinuous Galerkin method is based on a weak formulation of (3.23), therefore we proceed by multiplying (3.23) with a test function $\phi = \phi(\xi)$ and integrate over the unit interval E so that we obtain

$$\int_E \left(\mathcal{J} \frac{\partial u}{\partial t} + \frac{\partial f}{\partial \xi} \right) \phi \, \mathrm{d}\xi = 0. \tag{3.24}$$

For the first term in the parenthesis of equation (3.24) the time derivative is taken outside the integral, whereas for the second term integration by parts is applied so that we arrive at

$$\frac{\mathrm{d}}{\mathrm{d}t} \int_E \mathcal{J} u \phi \, \mathrm{d}\xi = \int_E f \frac{\mathrm{d}\phi}{\mathrm{d}\xi} \, \mathrm{d}\xi - [F\phi]_{\xi=0}^{1}. \tag{3.25}$$

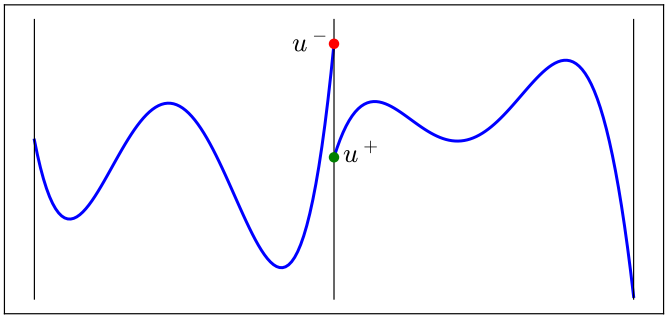


Figure 3.2: Solution on two elements: the red bullet refers to u^- and the green bullet refers to u^+.

In the last term of (3.25) we have replaced the flux f with a numerical flux F that uniquely defines the flux at the edge of an element. This numerical flux is necessary because in the discontinuous Galerkin method continuity across elements is not explicitly required. The numerical flux depends on both values at the edge, i.e., $F = F(u^-, u^+)$ with u^- and u^+ the values of u on both sides of the edge; see Figure 3.2.

On each element we will approximate both the solution u and the flux f by expansions into a basis of Lagrange polynomials, i.e., the following approximations are made

$$u(t,\xi) \approx u_\mathrm{h}(t,\xi) = \sum_{i=0}^{N} u_i(t)\ell_i(\xi), \tag{3.26a}$$

$$f(t,\xi) \approx f_\mathrm{h}(t,\xi) = \sum_{i=0}^{N} f_i(t)\ell_i(\xi), \tag{3.26b}$$

where we take $f_i(t) = f(t, x_i, u_i(t))$ with $x_i = x_{k-1/2} + \xi_i \Delta x_k$ and the Lagrange polynomials have nodes at the $(N+1)$-point Gauss-Legendre quadrature nodes. These expansions can now be inserted into the integrals in (3.25). Next, we require the weak formulation to hold for test functions that are taken equal to the set of basis functions. In order words we require the weak form to hold for $\phi = \ell_j$, that is we need to solve

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_E \mathcal{J} u_\mathrm{h}\ell_j \,\mathrm{d}\xi = \int_E f_\mathrm{h}\frac{\mathrm{d}\ell_j}{\mathrm{d}\xi}\,\mathrm{d}\xi - \left[F\ell_j\right]_{\xi=0}^{1}, \tag{3.27}$$

for $j = 0,\ldots,N$. Consider now the term on the left-hand side of (3.27). This

term is evaluated as

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_E \mathcal{J} u_\mathrm{h}\ell_j \,\mathrm{d}\xi = \mathcal{J}\frac{\mathrm{d}}{\mathrm{d}t}\sum_{i=0}^{N} u_i(t)\int_E \ell_i(\xi)\ell_j(\xi)\,\mathrm{d}\xi = \mathcal{J}\frac{\mathrm{d}u_j}{\mathrm{d}t}w_j, \tag{3.28}$$

where in the last equality we have used the orthogonality of the basis functions (3.21). For the first term on the right-hand side of (3.27) we obtain

$$\begin{aligned}\int_E f_\mathrm{h}\frac{\mathrm{d}\ell_j}{\mathrm{d}\xi}\,\mathrm{d}\xi &= \sum_{i=0}^{N} f_i \int_E \ell_i(\xi)\frac{\mathrm{d}\ell_j(\xi)}{\mathrm{d}\xi}\,\mathrm{d}\xi = \sum_{i=0}^{N} f_i \sum_{n=0}^{N} w_n \ell_i(\xi_n)\frac{\mathrm{d}\ell_j}{\mathrm{d}\xi}(\xi_n) \\ &= \sum_{i=0}^{N} f_i \sum_{n=0}^{N} w_n \delta_{in} D_{nj} = \sum_{i=0}^{N} w_i f_i D_{ij},\end{aligned} \tag{3.29}$$

where the integral is exactly evaluated, and we have applied property (3.7) and used the derivative matrix (3.15). Expanding now the second term on the right-hand side of (3.27) we obtain

$$\left[F\ell_j\right]_{\xi=0}^{1} = F(t,1)\ell_j(1) - F(t,0)\ell_j(0), \tag{3.30}$$

where the numerical fluxes at the element boundaries depend on the value of u from this element and the value from its neighbouring element. In case of a physical boundary there is no neighbouring element, but in that case we can include boundary conditions into the numerical flux.

Inserting the three terms (3.28)-(3.30) into equation (3.27) results into an ODE system for the expansion coefficients of the solution, that reads

$$\mathcal{J}\frac{\mathrm{d}u_j}{\mathrm{d}t} = \sum_{i=0}^{N} \hat{D}_{ji} f_i - \left[F(t,1)\frac{\ell_j(1)}{w_j} - F(t,0)\frac{\ell_j(0)}{w_j}\right], \tag{3.31}$$

for $j = 0,\ldots,N$, and where we define

$$\hat{D}_{ji} = D_{ij}\frac{w_i}{w_j}. \tag{3.32}$$

Finally, we remark that the DGSEM (3.31) has a local conservation property, i.e., the integral of u_h on an element changes according to the fluxes F at the boundary of the element. That is,

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}\int_E \mathcal{J} u_\mathrm{h}\,\mathrm{d}\xi = \sum_{j=0}^{N}\mathcal{J} w_j\frac{\mathrm{d}u_j}{\mathrm{d}t} &= \sum_{j=0}^{N}\left(\sum_{i=0}^{N} D_{ij} w_i f_i - \left[F(t,1)\ell_j(1) - F(t,0)\ell_j(0)\right]\right) \\ &= -\left[F(t,1) - F(t,0)\right],\end{aligned} \tag{3.33}$$

where we used that $\sum_j D_{ij} = 0$ and $\sum_j \ell_j = 1$.

Equation (3.31) represents the ODE system for 1 element only. For every element we have such an ODE system, so that by collecting all the coefficients for all elements into a vector $\boldsymbol{u}$ we can write it as one big generic ODE system which reads

$$\frac{\mathrm{d}\boldsymbol{u}}{\mathrm{d}t} = \boldsymbol{g}(t, \boldsymbol{u}(t)). \tag{3.34}$$

Such an ODE system can be integrated numerically using various methods, such as linear multistep methods or Runge-Kutta methods. Popular Runge-Kutta methods are, for example, the classical explicit fourth-order Runge-Kutta method, but also explicit low storage Runge-Kutta methods are commonplace for DG discretisations of PDEs [15, 53, 94]. The latter type of methods minimize the storage requirements when solving an ODE system, where some methods require only two storage locations per ODE. This is in particular important when a PDE is discretised in multiple dimensions, where storage requirements is one of the major considerations in choosing an ODE solver.

The DGSEM combined with an explicit Runge-Kutta method has to satisfy a stability condition that restricts the stepsize Δt when updating the ODE system (3.34). The stability restrictions for DGSEM with an explicit Runge-Kutta method in 1D, can be described by the Courant-Friedrichs-Lewy (CFL) condition that reads [24]

$$\Delta t \leq \frac{\mathrm{CFL}}{2N+1} \frac{h_{\min}}{a_{\max}}, \tag{3.35}$$

with $h_{\min} = \min_k \Delta x_k$ the minimal element size, $a_{\max} = \max \left| \frac{\partial f}{\partial u} \right|$ the maximal velocity and CFL a constant coefficient that typically satisfies $0 < \mathrm{CFL} < 1$.

3.3 Semi-Lagrangian discontinuous Galerkin methods

The DGSEM is one type of discontinuous Galerkin methods to semi-discretise the PDE (3.1). In this section, we will encounter two different methods for discretisation that belong to the class of semi-Lagrangian (SL) discontinuous Galerkin methods. For semi-Lagrangian methods, much like the DGSEM, the approximate solution is computed on a (usually) fixed set of grid points or mesh, as in any Eulerian approach. Semi-Lagrangian methods use the Lagrangian evolution of the solution, to compute the update on the grid or mesh. For the hyperbolic PDE (3.1) this means the solution is propagated along the characteristics. Semi-Lagrangian methods can be CFL free, that is there is no stability restriction on the stepsize Δt and, hence, can be more

efficient than standard discontinuous Galerkin methods. SL(DG) methods are especially popular in the Vlasov-Poisson simulation community [10, 13, 34, 35, 75, 80] and they are also used for atmospheric modelling [40, 46].

One of the key components of the SLDG method is the method of characteristics [65], so we start with a brief introduction of this method.

3.3.1 Method of characteristics

Consider the following one-dimensional hyperbolic advection equation

$$\frac{\partial u}{\partial t} + a\frac{\partial u}{\partial x} = 0, \tag{3.36}$$

with a the velocity field that can in general vary in space and time. Let the initial condition be denoted by $u_0(x) = u(0,x)$. The solution of (3.36) can be described in terms of its characteristics. The equations for the characteristics are derived as follows. Consider $u^*(t) = u(t,x(t))$ along some curve $x = x(t)$. The total time derivative of u^* reads

$$\frac{\mathrm{d}u^*}{\mathrm{d}t} = \frac{\partial u^*}{\partial t} + \frac{\mathrm{d}x}{\mathrm{d}t}\frac{\partial u^*}{\partial x}. \tag{3.37}$$

Now by taking $\frac{\mathrm{d}x}{\mathrm{d}t} = a(t,x)$ the right-hand side of equation (3.37) reduces to 0 by virtue of (3.36), so that u^* remains constant along this curve. Thus, the PDE is reduced to a set of (ODEs), viz.

$$\frac{\mathrm{d}u^*}{\mathrm{d}t} = 0, \tag{3.38a}$$

$$\frac{\mathrm{d}x}{\mathrm{d}t} = a(t,x(t)). \tag{3.38b}$$

The system of ODEs (3.38) describes the characteristics: its location satisfies (3.38b) and its value along the curve satisfies (3.38a). Solving the system of ODEs (3.38) for individual characteristics can be seen as a Lagrangian approach. By integrating the system of ODEs from 0 to t we find

$$u(t,x(t)) = u(0,x(0)), \tag{3.39a}$$

$$x(t) = x(0) + \int_0^t a(s,x(s))\,\mathrm{d}s, \tag{3.39b}$$

so that combined with the initial condition $u(0,x) = u_0(x)$ the following representation of the solution holds

$$u(t,x) = u_0\left(x - \int_0^t a(s,x(s))\,\mathrm{d}s\right). \tag{3.40}$$

In the special case where a is a constant, the curve $x = x(t)$ becomes a straight line given by $x(t) = x(0) + at$. Hence, equation (3.40) simplifies to

$$u(t,x) = u_0(x - at), \tag{3.41}$$

stating that the solution at a time t is an unperturbed translation of the initial condition over a distance at. Note that for a constant refractive index field n in Liouville's equation (2.40a) the characteristics curves also become straight lines.

3.3.2 Semi-Lagrangian discontinuous Galerkin in flux form

Let us return again to the PDE (3.1) but now we restrict ourselves to a linear flux, that is $f = au$. Similar to the derivation of the DGSEM, we partition the interval $[X_0, X_1]$ into intervals $[x_{k-1/2}, x_{k+1/2}]$. Contrary to the DGSEM, we will not transform the PDE to a reference domain. The weak formulation of the PDE is derived by multiplying it by a test function $\psi = \psi(x)$ and integrating over the interval $[x_{k-1/2}, x_{k+1/2}]$ which leads to

$$\int_{x_{k-1/2}}^{x_{k+1/2}} \left(\frac{\partial u}{\partial t} + \frac{\partial f}{\partial x} \right) \psi \, \mathrm{d}x = 0. \tag{3.42}$$

Taking the time derivative outside the integral and applying integration by parts to the flux term, yields

$$\frac{\mathrm{d}}{\mathrm{d}t} \int_{x_{k-1/2}}^{x_{k+1/2}} u\psi \, \mathrm{d}x = \int_{x_{k-1/2}}^{x_{k+1/2}} f \frac{\mathrm{d}\psi}{\mathrm{d}x} \, \mathrm{d}x - [f\psi]_{x=x_{k-1/2}}^{x_{k+1/2}}. \tag{3.43}$$

Subsequently, we integrate (3.43) over a time interval $[t^n, t^{n+1}]$ so that we obtain

$$\int_{x_{k-1/2}}^{x_{k+1/2}} u^{n+1}\psi \, \mathrm{d}x - \int_{x_{k-1/2}}^{x_{k+1/2}} u^n \psi \, \mathrm{d}x = \int_{t^n}^{t^{n+1}} \left(\int_{x_{k-1/2}}^{x_{k+1/2}} f \frac{\mathrm{d}\psi}{\mathrm{d}x} \, \mathrm{d}x - [f\psi]_{x=x_{k-1/2}}^{x_{k+1/2}} \right) \mathrm{d}t, \tag{3.44}$$

where we use the shorthand notation $u^n = u(t^n, \cdot)$.

In SLDG methods the exact evolution of u^n to a time $t \in [t^n, t^{n+1}]$ is used in the right-hand side of (3.44). Let T_τ be the exact evolution operator, which is defined such that $T_\tau(u^n)$ denotes the exact evolution of u^n, that starts at $t = t^n$ and propagates to $t = t^n + \tau$. For example, in the case of a constant velocity field a the exact evolution operator applied to u^n can directly be written as

$$T_\tau(u^n) = u^n(x - a\tau), \tag{3.45}$$

cf. (3.41).

The solution on each element is approximated by an expansion into Lagrange polynomials, e.g.,

$$u_{\mathrm{h},k}^n(x) = \sum_{i=0}^{N} u_{k,i}^n \ell_i\left(\frac{x - x_{k-1/2}}{\Delta x_k}\right) \quad \text{for} \quad x \in [x_{k-1/2}, x_{k+1/2}] \tag{3.46}$$

represents the expansion for the kth element. Furthermore, we use u_{h}^n to denote the full piecewise polynomial solution defined on $[X_0, X_1]$. With the piecewise polynomial representation of the solution, the exact evolution operator, and the flux $f = au$, the weak formulation (3.44) can be written as

$$\begin{aligned}\int_{x_{k-1/2}}^{x_{k+1/2}} u_{\mathrm{h}}^{n+1}\psi \,\mathrm{d}x - \int_{x_{k-1/2}}^{x_{k+1/2}} u_{\mathrm{h}}^{n}\psi \,\mathrm{d}x = \int_0^{\Delta t} \Bigg(\int_{x_{k-1/2}}^{x_{k+1/2}} aT_\tau\left(u_{\mathrm{h}}^n\right) \frac{\mathrm{d}\psi}{\mathrm{d}x} \,\mathrm{d}x \\ - \left[aT_\tau\left(u_{\mathrm{h}}^n\right)\psi\right]_{x=x_{k-1/2}}^{x_{k+1/2}} \Bigg) \mathrm{d}\tau,\end{aligned} \tag{3.47}$$

where $\Delta t = t^{n+1} - t^n$. The SLDG formulation (3.47) can directly be used with the test function taken equal to each of the basis functions. This type of formulation was for example used in [75]. In that paper, the authors approximate the spatial integral in the right-hand side with quadrature and rewrite the τ-integral in terms of a spatial integral that is evaluated exactly. In the right-hand side of the SLDG formulation (3.47) there is a volume term and a boundary term, where the latter represents the fluxes at the boundary of an element. Therefore, we refer to the formulation (3.47) as an SLDG in flux form.

3.3.3 Semi-Lagrangian discontinuous Galerkin in direct form

In [34, 35, 80] an SLDG formulation different from (3.47) is used. The SLDG formulation (3.47) can be rewritten, assuming everything is exactly evaluated, into a different form as follows. Applying integration by parts in x on the first integral in the right-hand side of (3.47) leads to

$$\int_{x_{k-1/2}}^{x_{k+1/2}} u_{\mathrm{h}}^{n+1}\psi \,\mathrm{d}x - \int_{x_{k-1/2}}^{x_{k+1/2}} u_{\mathrm{h}}^{n}\psi \,\mathrm{d}x = \int_0^{\Delta t} \int_{x_{k-1/2}}^{x_{k+1/2}} \psi \frac{\partial}{\partial x}\left(-aT_\tau\left(u_{\mathrm{h}}^n\right)\right) \mathrm{d}x \,\mathrm{d}\tau. \tag{3.48}$$

Now one can make use of the PDE (3.1), change the order of integration and evaluate the τ-integral to obtain

$$\int_{x_{k-1/2}}^{x_{k+1/2}} u_{\mathrm{h}}^{n+1}\psi\,\mathrm{d}x - \int_{x_{k-1/2}}^{x_{k+1/2}} u_{\mathrm{h}}^{n}\psi\,\mathrm{d}x = \int_0^{\Delta t}\int_{x_{k-1/2}}^{x_{k+1/2}} \psi\frac{\partial}{\partial t}T_\tau\left(u_{\mathrm{h}}^n\right)\mathrm{d}x\,\mathrm{d}\tau$$
$$= \int_{x_{k-1/2}}^{x_{k+1/2}} \left(T_{\Delta t}\left(u_{\mathrm{h}}^n\right) - T_0\left(u_{\mathrm{h}}^n\right)\right)\psi\,\mathrm{d}x. \quad (3.49)$$

The operator T_0 is simply the identity operator so that $T_0(u_{\mathrm{h}}^n) = u_{\mathrm{h}}^n$, consequently the second terms on both sides of (3.49) cancel. Thus the final result reads

$$\int_{x_{k-1/2}}^{x_{k+1/2}} u_{\mathrm{h}}^{n+1}\psi\,\mathrm{d}x = \int_{x_{k-1/2}}^{x_{k+1/2}} T_{\Delta t}\left(u_{\mathrm{h}}^n\right)\psi\,\mathrm{d}x, \quad (3.50)$$

where the test function ψ is taken equal to the basis functions that appear in the expansion (3.46). The weak formulation (3.50) is referred to as the direct form of the Lagrangian Galerkin method in [75], and in other papers just referred to as SLDG [34, 35, 80]. We will also just refer to it as SLDG.

In the SLDG method we solve (3.50), where the solution on each element u_{h}^n is represented in a polynomial basis by (3.46). The SLDG method can thus be interpreted as a translation of the piecewise polynomial solution to t^{n+1} followed by an L_2-projection onto the polynomial basis.

Consider now the case of a constant velocity field a. Then, the exact evolution operator is given by (3.45) and the SLDG method (3.50) can be written as

$$\int_{x_{k-1/2}}^{x_{k+1/2}} u_{\mathrm{h}}^{n+1}(x)\ell_j\left(\frac{x-x_{k-1/2}}{\Delta x_k}\right)\mathrm{d}x = \int_{x_{k-1/2}}^{x_{k+1/2}} u_{\mathrm{h}}^{n}(x-a\Delta t)\ell_j\left(\frac{x-x_{k-1/2}}{\Delta x_k}\right)\mathrm{d}x, \quad (3.51)$$

for $j = 0,\ldots,N$. The translation of the polynomial solution u_{h}^n of each element gives rise to a piecewise polynomial with discontinuities in the integration interval $[x_{k-1/2}, x_{k+1/2}]$. If we would apply Gauss-Legendre quadrature directly to the right-hand side integral of (3.51) without considering these discontinuities then we would lose the conservative properties of a DG scheme. The location of the discontinuities can, however, be computed so that the right-hand side integral can be split into multiple integrals where on each part the integrand is regular. Although not necessary, it is beneficial to restrict ourselves to a uniform mesh spacing so that $\Delta x_k = \Delta x$ for all elements. Let us now use the shorthand notation

$$\hat{\ell}_j(x) = \ell_j\left(\frac{x-x_{k-1/2}}{\Delta x}\right). \quad (3.52)$$

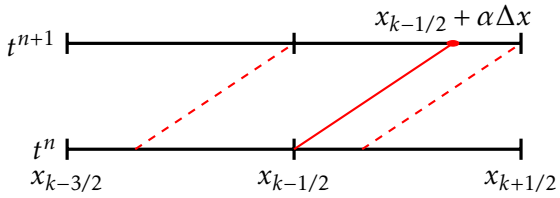


Figure 3.3: Location of the discontinuity on an element. Red lines indicate characteristic lines.

The left-hand side of (3.51) is easily evaluated due to the orthogonality of the basis functions to be

$$\sum_{i=0}^{N} u_{k,i}^{n+1} \int_{x_{k-1/2}}^{x_{k+1/2}} \hat{\ell}_i(x)\hat{\ell}_j(x)\,\mathrm{d}x = \Delta x \sum_{i=0}^{N} u_{k,i}^{n+1} \int_E \ell_i(\xi)\ell_j(\xi)\,\mathrm{d}\xi = \Delta x\, w_j u_{k,j}^{n+1}. \tag{3.53}$$

For the evaluation of the right-hand side of (3.51) there is only one discontinuity due to the mesh spacing being uniform. A sketch of how the discontinuity is located is shown in Figure 3.3. The location of the discontinuity in $[x_{k-1/2}, x_{k+1/2}]$ can be written as $x_{k-1/2}+\alpha\Delta x$ with $0 \le \alpha < 1$. This α is directly related to the propagation distance $a\Delta t$, i.e., we write

$$a\Delta t = m\Delta x + \alpha\Delta x, \tag{3.54}$$

with $m \in \mathbb{Z}$. The integer m can directly be found as

$$m = \lfloor \frac{a\Delta t}{\Delta x} \rfloor, \tag{3.55}$$

with $\lfloor\cdot\rfloor$ denoting the floor operation that returns the first integer that is smaller than or equal to the given argument.

The integral on the right-hand side of (3.51) is first split into two integrals at the point $x_{k-1/2}+\alpha\Delta x$, i.e.,

$$\begin{aligned}\int_{x_{k-1/2}}^{x_{k+1/2}} u_{\mathrm{h}}^n(x-a\Delta t)\hat{\ell}_j(x)\,\mathrm{d}x = &\int_{x_{k-1/2}}^{x_{k-1/2}+\alpha\Delta x} u_{\mathrm{h}}^n(x-a\Delta t)\hat{\ell}_j(x)\,\mathrm{d}x \\ &+ \int_{x_{k-1/2}+\alpha\Delta x}^{x_{k+1/2}} u_{\mathrm{h}}^n(x-a\Delta t)\hat{\ell}_j(x)\,\mathrm{d}x,\end{aligned} \tag{3.56}$$

where we have used the definition for $\hat{\ell}_j$. In the integrals, u_{h}^n takes on values from one element for each integral. With definition (3.54) we can compute the indices of these elements. For example, for the first integral in the right-hand

side of (3.56) we obtain

$$\int_{x_{k-1/2}}^{x_{k-1/2}+\alpha\Delta x} u_{\mathrm{h}}^n(x-a\Delta t)\hat{\ell}_j(x)\,\mathrm{d}x = \int_{x_{k-1/2}}^{x_{k-1/2}+\alpha\Delta x} u_{\mathrm{h},k-m-1}^n(x-a\Delta t)\hat{\ell}_j(x)\,\mathrm{d}x$$
$$= \sum_{i=0}^{N} u_{k-m-1,i}^n \int_{x_{k-1/2}}^{x_{k-1/2}+\alpha\Delta x} \ell_i\left(\frac{x-a\Delta t-x_{k-m-1-1/2}}{\Delta x}\right)\ell_j\left(\frac{x-x_{k-1/2}}{\Delta x}\right)\mathrm{d}x. \tag{3.57}$$

Note that the sum of $a\Delta t = m\Delta x + \alpha\Delta x$ and $x_{k-m-1-1/2} = x_{k-1/2} - m\Delta x - \Delta x$ is given by

$$a\Delta t + x_{k-m-1-1/2} = x_{k-1/2} + (\alpha-1)\Delta x.$$

Therefore, the remaining integral in (3.57) can be written as

$$\int_{x_{k-1/2}}^{x_{k-1/2}+\alpha\Delta x} \ell_i\left(\frac{x-x_{k-1/2}-(\alpha-1)\Delta x}{\Delta x}\right)\ell_j\left(\frac{x-x_{k-1/2}}{\Delta x}\right)\mathrm{d}x$$
$$= \alpha\Delta x\int_0^1 \ell_i(s\alpha+1-\alpha)\,\ell_j(s\alpha)\,\mathrm{d}s. \tag{3.58}$$

For the second integral in the right-hand side of (3.56) we obtain

$$\int_{x_{k-1/2}+\alpha\Delta x}^{x_{k+1/2}} u_{\mathrm{h}}^n(x-a\Delta t)\hat{\ell}_j(x)\,\mathrm{d}x = \int_{x_{k-1/2}+\alpha\Delta x}^{x_{k+1/2}} u_{\mathrm{h},k-m}^n(x-a\Delta t)\hat{\ell}_j(x)\,\mathrm{d}x$$
$$= \sum_{i=0}^{N} u_{k-m,i}^n \int_{x_{k-1/2}+\alpha\Delta x}^{x_{k+1/2}} \ell_i\left(\frac{x-a\Delta t-x_{k-m-1/2}}{\Delta x}\right)\ell_j\left(\frac{x-x_{k-1/2}}{\Delta x}\right)\mathrm{d}x. \tag{3.59}$$

Note that the sum of $a\Delta t = m\Delta x + \alpha\Delta x$ and $x_{k-m-1/2} = x_{k-1/2} - m\Delta x$ is given by

$$a\Delta t + x_{k-m-1/2} = x_{k-1/2} + \alpha\Delta x,$$

and thus the remaining integral in (3.59) can be written as

$$\int_{x_{k-1/2}+\alpha\Delta x}^{x_{k+1/2}} \ell_i\left(\frac{x-x_{k-1/2}-\alpha\Delta x}{\Delta x}\right)\ell_j\left(\frac{x-x_{k-1/2}}{\Delta x}\right)\mathrm{d}x$$
$$= (1-\alpha)\Delta x\int_0^1 \ell_i\Big(s(1-\alpha)\Big)\ell_j\Big(\alpha+s(1-\alpha)\Big)\mathrm{d}s. \tag{3.60}$$

Inserting the expressions (3.53), (3.57)-(3.60) into (3.51) leads to an update formula for the expansion coefficients as

$$u_{k,j}^{n+1} = \sum_{i=0}^{N} A_{ji}u_{k-m-1,i}^n + \sum_{i=0}^{N} B_{ji}u_{k-m,i}^n \quad \text{for } j=0,\ldots,N, \tag{3.61a}$$

where A_{ji} and B_{ji} describe the coefficients of the matrices $\boldsymbol{A} = (A_{ji})$ and $\boldsymbol{B} = (B_{ji})$. These coefficients are defined by

$$A_{ji} = \frac{\alpha}{w_j} \int_0^1 \ell_i\Big(s\alpha + 1 - \alpha\Big)\ell_j\Big(s\alpha\Big)\,\mathrm{d}s, \tag{3.61b}$$

$$B_{ji} = \frac{1-\alpha}{w_j} \int_0^1 \ell_i\Big(s(1-\alpha)\Big)\ell_j\Big(\alpha + s(1-\alpha)\Big)\,\mathrm{d}s. \tag{3.61c}$$

The integrals for the coefficients A_{ji} and B_{ji} given by (3.61b)-(3.61c) are exactly evaluated with $(N+1)$-point Gauss-Legendre quadrature. Furthermore, note that due to the uniform mesh spacing the matrices are independent of the element index k, so that the same matrices can be used for all elements.

Since the right-hand side of the SLDG formulation (3.51) is evaluated exactly, the SLDG scheme has the conservative properties of a DG scheme. Moreover, when $\alpha = 0$ in (3.54) the scheme describes an exact shift of moments, i.e., an exact shift of the solution from one element to a different element. And lastly the SLDG scheme is CFL free owing to its Lagrangian type evolution.

3.4 Summary

In this chapter different DG methods have been described, such as the discontinuous Galerkin spectral element method and two different formulations of semi-Lagrangian discontinuous Galerkin methods, given by equations (3.47) and (3.50). The former method provides a semi-discretisation of the PDE leading to a large ODE system, whereas the SLDG methods lead to a fully discrete scheme. For the DG methods we use numerical tools such as approximation by an expansion into Lagrange polynomials, from which derivatives can be easily computed. Furthermore, the integrals that appear in the weak formulations are evaluated by suitable Gauss-Legendre quadrature rules. In the next chapter, as a first step, Liouville's equation on a two-dimensional phase space is solved by applying the DGSEM.

Chapter 4

DG spectral element method for 2D optics

In this chapter, we will apply a two-dimensional variant of the discontinuous Galerkin spectral element method (DGSEM) for the spatial discretisation of Liouville's equation[1]. The method does not enforce continuity across the boundary of each element. This property makes the method particularly suitable for the discontinuous solutions across optical interfaces. The discretisation of the jump condition at an optical interface is not straightforward and will be treated for a flat optical interface in this chapter.

In the following chapters, we will first focus on two-dimensional optics. For two-dimensional optics the position and momentum on phase space are denoted by q and p, respectively, and represent scalars. Here, we consider only forward-propagating light so that Liouville's equation reads

$$\frac{\partial \rho}{\partial z} + \nabla \cdot \boldsymbol{f} = 0, \tag{4.1a}$$

where $\nabla = \left(\frac{\partial}{\partial q}, \frac{\partial}{\partial p}\right)$ and the flux vector $\boldsymbol{f}$ now reads

$$\boldsymbol{f} = \rho \boldsymbol{u} = \rho \begin{pmatrix} \frac{\partial H}{\partial p} \\ -\frac{\partial H}{\partial q} \end{pmatrix}. \tag{4.1b}$$

The Hamiltonian H for two-dimensional optics reduces to

$$H(z, q, p) = -\sqrt{n(z, q)^2 - p^2}, \tag{4.2}$$

[1]This chapter is based on the published article: R. A. M. van Gestel, M. J. H. Anthonissen, J. H. M. ten Thije Boonkkamp, and W. L. IJzerman. An energy conservative hp-method for Liouville's equation of geometrical optics. *Journal of Scientific Computing*, 89(1):1-35, 2021.

and consequently the velocity $\boldsymbol{u}$ reads

$$\boldsymbol{u} = \frac{1}{\sqrt{n^2 - p^2}} \begin{pmatrix} p \\ n \frac{\partial n}{\partial q} \end{pmatrix}. \tag{4.3}$$

The DGSEM for two-dimensional domains is discussed by Kopriva in [57]. The phase space domain is partitioned into elements, where on each element the solution is approximated using a polynomial. The DGSEM as described in [57] can directly deal with refractive index fields that are continuous everywhere. On the other hand, at an optical interface the discretisation needs to be modified.

At an optical interface a jump condition describes how the basic luminance is redistributed together with Snell's law of refraction or the law of specular reflection. The jump condition describes non-local boundary conditions for the basic luminance in phase space. Our contribution consists of describing the treatment of these optical interfaces so that the scheme obeys energy conservation. In the DGSEM the elements communicate using numerical fluxes. Snell's law and the law of specular reflection are incorporated in these numerical fluxes at an optical interface. In addition to the discontinuous change in the direction coordinate described by these laws, a single element before the optical interface might contribute to multiple elements after the optical interface. This connection to multiple elements is similar to fully non-conforming geometries when using subdomain refinement [7, 8]. Kopriva et al. outlined such a strategy for the DGSEM in [61]. In [12] an analysis of this method is presented by Bui-Thanh and Ghattas. Across an optical interface the numerical fluxes are discontinuous and therefore we have to take a different approach. Inspired by [61], we present a method that directly incorporates the laws of optics and obeys energy conservation.

First, the semi-discretisation with the DGSEM is outlined in Sections 4.1-4.2. Second, the discretisation at optical interfaces and the energy balances required are detailed in Section 4.3. After that, results are presented for the DGSEM applied to two test cases in Section 4.4.

4.1 Weak formulation

For phase space discretisation, the two-dimensional phase space domain $\mathcal{P}$ is covered with straight-sided quadrilaterals $\Omega^k \subset \mathcal{P}$ with k the index of the element. In a more general discretisation, the boundaries of quadrilaterals are allowed to be curved, such that curved boundaries from physical constraints can be modelled appropriately. In fact, when the refractive index field changes

continuously as a function of q, then the maximum allowed momentum varies as a function of q due to the restriction of $\vec{p}$ to Descartes' sphere. This restriction can be accommodated by curved boundaries when solving Liouville's equation; see [90]. For a discussion on DGSEM with curved quadrilateral elements, see for example [17, 51, 57, 58]. In this chapter we only consider straight-sided quadrilaterals.

Each quadrilateral Ω^k has four vertices $\{\boldsymbol{x}_1, \boldsymbol{x}_2, \boldsymbol{x}_3, \boldsymbol{x}_4\}$ labelled in counterclockwise direction where $\boldsymbol{x} = (q, p)$ and we have omitted the element index (superscript k); see Figure 4.1. For ease of computation, the reference square $\chi = [0,1]^2$ is mapped to each quadrilateral Ω^k, transforming a point in the reference domain $(\xi, \eta) \in \chi$ to a point in physical space $\boldsymbol{x}(\xi, \eta) \in \mathcal{P}$ using the following bilinear transformation

$$\boldsymbol{x}(\xi,\eta) = (1-\xi)(1-\eta)\boldsymbol{x}_1 + \xi(1-\eta)\boldsymbol{x}_2 + \xi\eta\boldsymbol{x}_3 + (1-\xi)\eta\boldsymbol{x}_4. \tag{4.4}$$

The Jacobian of the transformation is given by $\frac{\partial(q,p)}{\partial(\xi,\eta)} = \left(\frac{\partial \boldsymbol{x}}{\partial \xi}, \frac{\partial \boldsymbol{x}}{\partial \eta}\right)$, where the columns read

$$\frac{\partial \boldsymbol{x}}{\partial \xi} = \begin{pmatrix} \frac{\partial q}{\partial \xi} \\ \frac{\partial p}{\partial \xi} \end{pmatrix} = (1-\eta)(\boldsymbol{x}_2 - \boldsymbol{x}_1) + \eta(\boldsymbol{x}_3 - \boldsymbol{x}_4), \tag{4.5a}$$

$$\frac{\partial \boldsymbol{x}}{\partial \eta} = \begin{pmatrix} \frac{\partial q}{\partial \eta} \\ \frac{\partial p}{\partial \eta} \end{pmatrix} = (1-\xi)(\boldsymbol{x}_4 - \boldsymbol{x}_1) + \xi(\boldsymbol{x}_3 - \boldsymbol{x}_2). \tag{4.5b}$$

The divergence term in (4.1a) can be rewritten by applying the chain rule resulting in

$$\nabla \cdot \boldsymbol{f} = \frac{1}{\mathcal{J}} \nabla_\xi \cdot \tilde{\boldsymbol{f}}, \tag{4.6}$$

where $\mathcal{J} = \frac{\partial q}{\partial \xi}\frac{\partial p}{\partial \eta} - \frac{\partial q}{\partial \eta}\frac{\partial p}{\partial \xi}$ denotes the Jacobian determinant, $\nabla_\xi = \left(\frac{\partial}{\partial \xi}, \frac{\partial}{\partial \eta}\right)$ and $\tilde{\boldsymbol{f}}$ is an auxiliary flux defined by the product of the adjoint Jacobian matrix and the flux $\boldsymbol{f}$, i.e.,

$$\tilde{\boldsymbol{f}} = \begin{pmatrix} \frac{\partial p}{\partial \eta} & -\frac{\partial q}{\partial \eta} \\ -\frac{\partial p}{\partial \xi} & \frac{\partial q}{\partial \xi} \end{pmatrix} \boldsymbol{f}. \tag{4.7}$$

Applying the transformation (4.6) to Liouville's equation (4.1a), we obtain

$$\frac{\partial \rho}{\partial z} + \frac{1}{\mathcal{J}} \nabla_\xi \cdot \tilde{\boldsymbol{f}} = 0, \tag{4.8}$$

where $\rho = \rho(z, \xi, \eta)$.

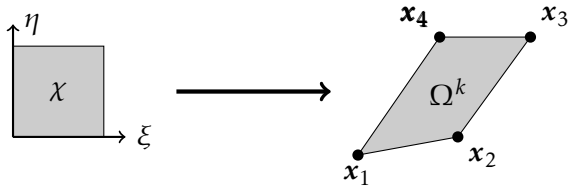


Figure 4.1: Mapping from reference square χ to a quadrilateral Ω^k.

The weak formulation of Liouville's equation is obtained by first multiplying the PDE (4.8) by the Jacobian determinant $\mathcal{J}$ and by a smooth test function ϕ, and subsequently integrating over the reference domain χ. This results in

$$\int_\chi \phi \mathcal{J} \frac{\partial \rho}{\partial z} \,\mathrm{d}\boldsymbol{\xi} + \int_\chi \phi \nabla_{\boldsymbol{\xi}} \cdot \tilde{\boldsymbol{f}} \,\mathrm{d}\boldsymbol{\xi} = 0. \tag{4.9}$$

The second term is rewritten by applying the product rule and Gauss's theorem, so that

$$\begin{aligned}\int_\chi \phi \nabla_{\boldsymbol{\xi}} \cdot \tilde{\boldsymbol{f}} \,\mathrm{d}\boldsymbol{\xi} &= \int_\chi \left(\nabla_{\boldsymbol{\xi}} \cdot \left(\phi \tilde{\boldsymbol{f}}\right) - \left(\nabla_{\boldsymbol{\xi}} \phi\right) \cdot \tilde{\boldsymbol{f}}\right) \mathrm{d}\boldsymbol{\xi} \\ &= \int_{\partial\chi} \phi \tilde{\boldsymbol{f}} \cdot \hat{\boldsymbol{N}} \,\mathrm{d}\sigma - \int_\chi \left(\nabla_{\boldsymbol{\xi}} \phi\right) \cdot \tilde{\boldsymbol{f}} \,\mathrm{d}\boldsymbol{\xi},\end{aligned}$$

where $\hat{\boldsymbol{N}}$ is the outward unit normal on $\partial\chi$ and the orientation of the closed curve $\partial\chi$ is counter-clockwise. Using this, we obtain the weak formulation of Liouville's equation on the reference domain

$$\int_\chi \phi \mathcal{J} \frac{\partial \rho}{\partial z} \,\mathrm{d}\boldsymbol{\xi} + \int_{\partial\chi} \phi \tilde{\boldsymbol{f}} \cdot \hat{\boldsymbol{N}} \,\mathrm{d}\sigma - \int_\chi \left(\nabla_{\boldsymbol{\xi}} \phi\right) \cdot \tilde{\boldsymbol{f}} \,\mathrm{d}\boldsymbol{\xi} = 0. \tag{4.10}$$

Note that for strong solutions we require the flux to be differentiable, hence, $H(z,q,p)$ given by (4.2) should be twice differentiable. However, the DGSEM uses the weak form of the solution and only requires the flux to be continuous, therefore, $H(z,q,p)$ being once continuously differentiable is sufficient. For optical interfaces this is not sufficient since the refractive index field is discontinuous and, therefore, $H(z,q,p)$ and also the flux are discontinuous. In particular, for these interfaces we require a special treatment of the fluxes which we will discuss in Section 4.3.

4.2 Approximating the solution with DGSEM

The solution ρ in equation (4.10) is approximated by an expansion in basis functions. For these basis functions we use a tensor product of one-

dimensional Lagrange polynomials that are of degree N. For the one-dimensional Lagrange polynomial of degree N we take nodes at the $(N+1)$-point Gauss-Legendre quadrature nodes defined over the unit interval $[0,1]$. To approximate the weak formulation (4.10), we expand both the solution and the flux in Lagrange polynomials. The expansions read

$$\rho(z,\xi,\eta) \approx \rho_{\mathrm{h}}(z,\xi,\eta) = \sum_{i,j=0}^{N} \rho_{ij}(z)\ell_i(\xi)\ell_j(\eta), \tag{4.11a}$$

$$\tilde{\boldsymbol{f}}(z,\xi,\eta) \approx \tilde{\boldsymbol{f}}_{\mathrm{h}}(z,\xi,\eta) = \sum_{i,j=0}^{N} \tilde{\boldsymbol{f}}_{ij}(z)\ell_i(\xi)\ell_j(\eta). \tag{4.11b}$$

The coefficients ρ_{ij} and $\tilde{\boldsymbol{f}}_{ij}$ are related to the location of an element's interior node (q_{ij}, p_{ij}), by $\rho_{ij}(z) = \rho(z, q_{ij}, p_{ij})$ and $\tilde{\boldsymbol{f}}_{ij}(z) = \tilde{\boldsymbol{f}}(z, q_{ij}, p_{ij})$. The auxiliary flux coefficients $\tilde{\boldsymbol{f}}_{ij}(z)$ are related to ρ by

$$\tilde{\boldsymbol{f}}_{ij}(z) = \tilde{\boldsymbol{u}}_{ij}(z)\rho_{ij}(z), \tag{4.12}$$

with $\tilde{\boldsymbol{u}}_{ij}$ the transformed velocity, similarly defined to (4.7). Here the velocity $\tilde{\boldsymbol{u}}_{ij}(z) = \tilde{\boldsymbol{u}}(z, q_{ij}, p_{ij})$ depends on z if the refractive index n depends on z. In the following, we omit $\tilde{\boldsymbol{f}}_{ij}$'s dependence on z for ease of notation.

Next, we have to approximate the integrals in equation (4.10). The test function ϕ is chosen to be in the same basis as the solution ρ, resulting in a Galerkin method. Therefore, taking

$$\phi(\xi,\eta) = \ell_i(\xi)\ell_j(\eta), \tag{4.13}$$

allows us to derive $(N+1)^2$ equations for the $(N+1)^2$ coefficients ρ_{ij}. Combining this with the approximations (4.11a) and (4.11b) for ρ and $\tilde{\boldsymbol{f}}$ we can approximate the integrals using the Gauss-Legendre quadrature rules. The Gauss-Legendre quadrature rules for higher-dimensional integrals on tensor-product domains, e.g., a square, are applied by treating the higher-dimensional integral as an iterative integral. That is, the 1D Gauss-Legendre quadrature rule is applied per dimension.

Substituting the approximation (4.11a) in the first term of (4.10), we obtain

$$\begin{aligned}\int_\chi \phi\mathcal{J}\frac{\partial\rho_{\mathrm{h}}}{\partial z}\,\mathrm{d}\boldsymbol{\xi} &= \int_\chi \ell_i(\xi)\ell_j(\eta)\mathcal{J}(\xi,\eta)\left(\sum_{k,l=0}^{N}\frac{\mathrm{d}\rho_{kl}(z)}{\mathrm{d}z}\ell_k(\xi)\ell_l(\eta)\right)\mathrm{d}\boldsymbol{\xi}\\ &= \sum_{n,m=0}^{N} w_n w_m \ell_i(\xi_n)\ell_j(\eta_m)\mathcal{J}(\xi_n,\eta_m)\left(\sum_{k,l=0}^{N}\frac{\mathrm{d}\rho_{kl}(z)}{\mathrm{d}z}\ell_k(\xi_n)\ell_l(\eta_m)\right),\end{aligned}$$

where $\{\eta_m\}_{m=0}^N$ describe the same $(N+1)$-point Gauss-Legendre quadrature nodes as $\{\xi_n\}_{n=0}^N$. Applying the Kronecker property (3.7) of the Lagrange polynomials, the sums reduce to

$$\int_\chi \phi \mathcal{J} \frac{\partial \rho_\mathrm{h}}{\partial z} \,\mathrm{d}\boldsymbol{\xi} = w_i w_j \mathcal{J}_{ij} \frac{\mathrm{d}\rho_{ij}(z)}{\mathrm{d}z}, \tag{4.14}$$

where $\mathcal{J}_{ij} = \mathcal{J}(\xi_i, \eta_j)$. Note that the integral is exactly evaluated for the given combination of a bilinear mapping $\boldsymbol{x}(\xi, \eta)$ and Lagrangian polynomials, since the integrand is a polynomial of degree $2N+1$ in ξ and in η. The chosen Gauss-Legendre quadrature rule is exact for this bivariate polynomial.

For the third term in (4.10), we substitute the approximation (4.11b) and denote $\tilde{\boldsymbol{f}} = (\tilde{f}, \tilde{g})$, resulting in

$$\begin{aligned}
\int_\chi \left(\nabla_\xi \phi\right) \cdot \tilde{\boldsymbol{f}}_\mathrm{h} \,\mathrm{d}\boldsymbol{\xi} &= \int_\chi \left(\ell_i'(\xi)\ell_j(\eta)\tilde{f}(\xi,\eta) + \ell_i(\xi)\ell_j'(\eta)\tilde{g}(\xi,\eta)\right) \mathrm{d}\boldsymbol{\xi} \\
&= \sum_{n,m=0}^{N} w_n w_m \left(\ell_i'(\xi_n)\ell_j(\eta_m)\tilde{f}(\xi_n,\eta_m) + \ell_i(\xi_n)\ell_j'(\eta_m)\tilde{g}(\xi_n,\eta_m)\right) \\
&= w_j \sum_{n=0}^{N} w_n D_{ni} \tilde{f}_{nj} + w_i \sum_{m=0}^{N} w_m D_{mj} \tilde{g}_{im},
\end{aligned}$$

where we have used the definition of the differentiation matrix (3.15). Furthermore, with the auxiliary matrix $\hat{D}_{ij}$ defined in (3.32), we obtain

$$\int_\chi \left(\nabla_\xi \phi\right) \cdot \tilde{\boldsymbol{f}}_\mathrm{h} \,\mathrm{d}\boldsymbol{\xi} = w_i w_j \left(\sum_{n=0}^{N} \hat{D}_{in} \tilde{f}_{nj} + \sum_{m=0}^{N} \hat{D}_{jm} \tilde{g}_{im} \right). \tag{4.15}$$

In what follows, we will replace the flux appearing in the boundary integral from equation (4.10) with a numerical flux $\tilde{\boldsymbol{F}} = \left(\tilde{F}, \tilde{G}\right)$. The boundary integral can be split into four parts and evaluated for each boundary segment; see Figure 4.2. Along each segment the numerical flux $\tilde{\boldsymbol{F}}$ is described by a Lagrange polynomial of degree N with nodes at the boundary nodes shown in the figure, i.e., along each segment the numerical flux can be represented by a one-dimensional expansion in Lagrange polynomials. For the bottom part, with $\eta = 0$, the integral can be exactly evaluated using Gauss-Legendre quadrature, such that we obtain

$$\int_0^1 \ell_i(\xi)\ell_j(0)\tilde{\boldsymbol{F}}(\xi,0) \cdot (-\hat{\boldsymbol{\eta}}) \,\mathrm{d}\xi = -w_i \ell_j(0) \tilde{G}(\xi_i, 0). \tag{4.16}$$

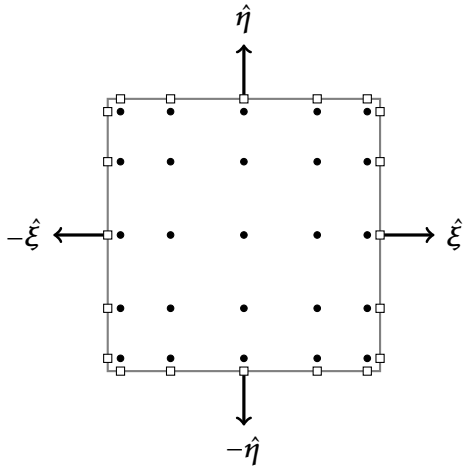


Figure 4.2: Reference square with polynomial degree $N = 4$. The unit normals are denoted by arrows, interior nodes are denoted by bullets and boundary points by open squares.

Similarly, we can compute the other components and the result for the full boundary integral reads

$$\begin{aligned}\int_{\partial\chi} \phi\tilde{\boldsymbol{F}}\cdot\hat{\boldsymbol{N}}\,\mathrm{d}\sigma =& w_j\left(\ell_i(1)\tilde{F}\left(1,\eta_j\right)-\ell_i(0)\tilde{F}\left(0,\eta_j\right)\right)\\ &+ w_i\left(\ell_j(1)\tilde{G}(\xi_i,1)-\ell_j(0)\tilde{G}(\xi_i,0)\right).\end{aligned} \tag{4.17}$$

In the discontinuous Galerkin spectral element method the elements communicate by fluxes through the faces of each element. The solution at the boundary between two elements is allowed to be discontinuous, thus the limit towards the boundary of an element can have two values, one for each element it touches. The flux on the boundary must be replaced by a numerical flux so that the neighbouring elements can communicate. The numerical flux depends on the left and right states of ρ at the boundary, i.e., ρ_{L} and ρ_{R}. For the numerical flux we take the upwind flux, i.e.,

$$\tilde{\boldsymbol{F}}\cdot\hat{\boldsymbol{N}} = \left(\tilde{\boldsymbol{u}}\cdot\hat{\boldsymbol{N}}\right)\begin{cases}\rho_{\mathrm{L}} & \text{if } \tilde{\boldsymbol{u}}\cdot\hat{\boldsymbol{N}}\geq 0,\\ \rho_{\mathrm{R}} & \text{if } \tilde{\boldsymbol{u}}\cdot\hat{\boldsymbol{N}}<0.\end{cases} \tag{4.18}$$

Note that $\tilde{F} = \tilde{\boldsymbol{F}}\cdot\hat{\boldsymbol{\xi}}$ and $\tilde{G} = \tilde{\boldsymbol{F}}\cdot\hat{\boldsymbol{\eta}}$.

Next, we substitute expressions (4.14), (4.15) and (4.17) in equation (4.10), so that we obtain the semi-discrete ODE system for the expansion coefficients

$\rho_{ij}(z)$:

$$\begin{aligned}\mathcal{J}_{ij}\frac{\mathrm{d}\rho_{ij}(z)}{\mathrm{d}z} &= \sum_{n=0}^{N}\hat{D}_{in}\tilde{f}_{nj}+\sum_{m=0}^{N}\hat{D}_{jm}\tilde{g}_{im}\\ &-\left[\frac{\ell_i(1)}{w_i}\tilde{F}(1,\eta_j)-\frac{\ell_i(0)}{w_i}\tilde{F}(0,\eta_j)+\frac{\ell_j(1)}{w_j}\tilde{G}(\xi_i,1)-\frac{\ell_j(0)}{w_j}\tilde{G}(\xi_i,0)\right],\end{aligned} \tag{4.19}$$

for $i = 0,\ldots,N$, $j = 0,\ldots,N$ and with the numerical fluxes $\tilde{\boldsymbol{F}} = \left(\tilde{F},\tilde{G}\right)$ given by (4.18). This ODE system can be solved using any numerical time integrator, e.g., the classical fourth-order Runge-Kutta method. Other popular choices in the literature are explicit low-storage Runge-Kutta methods, see [15, 53, 94].

The discontinuous Galerkin spectral element method approximates the exact solution by an Nth degree polynomial, so the global spatial error e for a typical mesh size Δx behaves as

$$e = \mathcal{O}(\Delta x^{N+1}). \tag{4.20}$$

Furthermore, the scheme is restricted by stability in terms of a CFL condition. For discontinuous Galerkin methods on quadrilaterals there is no direct known bound for the CFL condition. For triangular grids the relation between the Courant number and the shape of the triangles is studied in [16, 85].

4.3 Optical interfaces

At an optical interface the jump condition (2.42) describes the invariance of ρ across an interface together with a discontinuous change in the momentum. The change in momentum is computed according to the law of specular reflection or Snell's law of refraction. The jump condition thus describes how phase space is connected at an interface, i.e., it represents a non-local boundary condition.

In the DGSEM the optical interface in phase space is represented by a collection of momentum intervals, which are edges of elements, for both sides of the optical interface. In DG methods the solution is discontinuous across the boundary of its elements. The solution at the optical interface is given by a piecewise polynomial on either side of the interface. Refraction and reflection causes the elements to be connected in a non-trivial manner at the interface. For example, one single element can contribute to multiple elements on the other side. This occurs because both Snell's law and the law of reflection are non-linear in the momentum p.

From the discussion in Chapter 2 we know that the total luminous flux after reflection/refraction should remain constant, i.e., there should be energy conservation. Energy conservation is directly related to the fluxes over the boundary of a domain, which in terms of the DGSEM relates to the numerical fluxes (4.17) leaving an element. Hence, the jump condition (2.42) should be incorporated into the numerical fluxes of each element, and the numerical fluxes should be computed in such a way that they satisfy energy conservation at the optical interface. Note that for light incident on the optical interface we need to leave ρ free, whilst for outgoing light we need to prescribe the value of ρ. Here, incident and outgoing are directly related to the velocity field at the optical interface, e.g., if the velocity field is directed towards the interface then light is incident and if the velocity field is directed away from the interface then light is outgoing.

In the next sections, the discretisation at the optical interface for a test case is elaborated in the following steps. First, we will derive local energy balances that relate the fluxes on the incident side to the fluxes on the outgoing side. Second, we need to describe the connectivity of the elements at the optical interface. Third, we will use a least-squares matching that includes the jump condition and we will add an energy conservation constraint, described by the local energy balances. This results in values of ρ on the outgoing side, from which we compute the numerical fluxes.

4.3.1 Local energy balances

We consider the test case of a flat optical interface parallel to the z-axis. The refractive index field reads

$$n(q) = \begin{cases} n_0 & \text{for } q \leq q_0, \\ n_1 & \text{for } q > q_0. \end{cases} \tag{4.21}$$

This optical interface simplifies the discretisation of the jump condition. Specifically, the normal reads $\vec{\nu} = (\pm 1, 0)$ so that upon reflection and refraction the z-component of the full momentum vector, p_z, is preserved, cf. (2.43). Only the effect of reflection and refraction on p needs to be considered, hence, we introduce the function S that simply takes the first component of $\mathcal{S}$, defined in (2.43). The function S for two-dimensional optics at a surface with unit normal vector $\vec{\nu} = (\nu_q, \nu_z)$ reads

$$S(p; n_0, n_1, \vec{\nu}) = \begin{cases} S_\mathrm{R} = p - 2\psi\nu_q & \text{if } \delta \leq 0, \\ S_\mathrm{T} = p - (\psi + \sqrt{\delta})\nu_q & \text{if } \delta > 0, \end{cases} \tag{4.22a}$$

with

$$\psi = \begin{pmatrix} p \\ \sigma\sqrt{n_0^2 - p^2} \end{pmatrix} \cdot \begin{pmatrix} \nu_q \\ \nu_z \end{pmatrix} \text{ and } \delta = n_1^2 - n_0^2 + \psi^2. \tag{4.22b}$$

Moreover, we will use the shorthand notation $S(p) = S(p; n_0, n_1, \vec{\nu})$, use the notation S_T to describe the q-component of $\mathcal{S}_\mathrm{T}$ and we will use S_T^{-1} to denote the q-component for refraction in reverse, and similarly for S_R. Since p_z is preserved the propagation direction of light remains forward and thus the jump condition (2.42) simplifies to

$$\rho(z^+, q^+, p^+) = \rho(z^-, q^-, p^-) \text{ with } p^+ = S(p^-), \tag{4.23}$$

and we take $\sigma = 1$ in (4.22).

An optical interface in phase space is represented by line segments parallel to the p-axis at some constant q-value, therefore, only the q-component of the flux (4.1b) needs to be considered, i.e.,

$$f(z, q, p) = \rho(z, q, p)\frac{p}{\sqrt{n(z, q)^2 - p^2}}, \tag{4.24}$$

where we have substituted the velocity given by (4.3). The optical interface (4.21) has two sides, where on one side the refractive index takes on the value n_0 and the other side the value n_1. At a fixed z-value either side of the optical interface is represented by a line segment, representing the momentum domain. These line segments can be further partitioned according to whether light rays are incident or outgoing. The line segments on the optical interface describing incident light are denoted by L and the ones describing outgoing light are denoted R. The line segments denoting outgoing light can be further split into two parts, i.e., $R = R_\mathrm{R} \cup R_\mathrm{T}$. One part denoted by R_R corresponds to light getting there via reflection, and the other part denoted by R_T corresponds to light getting there via transmission/refraction.

In what follows, we assume that light is initially in the medium with refractive index n_0 and that $n_0 > n_1$. For the sake of simplicity, we will ignore the incident light that strikes the interface from the medium n_1. Hence, we can consider that L and R_R lie entirely in the medium with n_0, and that R_T lies entirely in the medium with n_1. To distinguish the momentum taken from the incident or outgoing line segments, we write $p \in L$ and $\bar{p} \in R$. First, consider the integral of the flux entering an arbitrary momentum interval $[\bar{p}_1, \bar{p}_2] \subseteq R_\mathrm{T}$. The integral reads

$$\int_{\bar{p}_1}^{\bar{p}_2} \rho(z^+, q_0^+, \bar{p}) \frac{\bar{p}}{\sqrt{n_1^2 - \bar{p}^2}} \, \mathrm{d}\bar{p}, \tag{4.25}$$

where (z^+, q_0^+) denotes the limit towards the optical interface from the line segment R_T. Relation (4.23) implies

$$\int_{\bar{p}_1}^{\bar{p}_2} \rho(z^+, q_0^+, \bar{p}) \frac{\bar{p}}{\sqrt{n_1^2 - \bar{p}^2}} \,\mathrm{d}\bar{p} = \int_{\bar{p}_1}^{\bar{p}_2} \rho(z^-, q_0^-, S_\mathrm{T}^{-1}(\bar{p})) \frac{\bar{p}}{\sqrt{n_1^2 - \bar{p}^2}} \,\mathrm{d}\bar{p}, \tag{4.26}$$

where (z^-, q_0^-) denotes the limit towards the optical interface from the line segment L. Subsequently, we transform the integral using $\bar{p} = S_\mathrm{T}(p)$ resulting in

$$\int_{\bar{p}_1}^{\bar{p}_2} \rho(z^+, q_0^+, \bar{p}) \frac{\bar{p}}{\sqrt{n_1^2 - \bar{p}^2}} \,\mathrm{d}\bar{p} = \int_{p_1}^{p_2} \rho(z^-, q_0^-, p) \frac{S_\mathrm{T}(p)}{\sqrt{n_1^2 - S_\mathrm{T}(p)^2}} \frac{\mathrm{d}S_\mathrm{T}(p)}{\mathrm{d}p} \,\mathrm{d}p, \tag{4.27}$$

where $\bar{p}_i = S_\mathrm{T}(p_i)$ for $i = 1, 2$, and $[p_1, p_2] \subseteq L$. The relation for reflection can be derived similarly by considering the integral of the flux entering an arbitrary momentum interval $[\bar{p}_3, \bar{p}_4] \subseteq R_\mathrm{R}$. We obtain the relation

$$\int_{\bar{p}_3}^{\bar{p}_4} \rho(z^+, q_0^+, \bar{p}) \frac{\bar{p}}{\sqrt{n_0^2 - \bar{p}^2}} \,\mathrm{d}\bar{p} = \int_{p_3}^{p_4} \rho(z^-, q_0^-, p) \frac{S_\mathrm{R}(p)}{\sqrt{n_0^2 - S_\mathrm{R}(p)^2}} \frac{\mathrm{d}S_\mathrm{R}(p)}{\mathrm{d}p} \,\mathrm{d}p, \tag{4.28}$$

with $\bar{p}_i = S_\mathrm{R}(p_i)$ for $i = 3, 4$, and $[p_3, p_4] \subseteq L$. The relations (4.27) and (4.28) describe how the fluxes leaving L are related to the fluxes entering R_T or R_R, respectively. Henceforth, they are known as energy conservation constraints.

For the particular optical interface (4.21) the constraints can be rewritten, so that they actually show the relation between the flux on the incident and outgoing sides of the optical interface. Since, we assumed that light was initially in the medium with refractive index n_0, the optical interface normal on the full position space is given by $\vec{\nu} = (\nu_q, \nu_z) = (-1, 0)$. The function S, given by (4.22), reduces for this flat interface to

$$\bar{p} = S(p) = \begin{cases} S_\mathrm{R} = -p & \text{if } p \leq p_\mathrm{c}, \\ S_\mathrm{T} = \sqrt{n_1^2 - n_0^2 + p^2} & \text{if } p > p_\mathrm{c}, \end{cases} \tag{4.29}$$

with $p_\mathrm{c} = \sqrt{n_0^2 - n_1^2}$ the critical momentum. Then, the energy conservation constraint for R_T, given by (4.27), can be simplified by noting that for refraction the following relations hold

$$\frac{\mathrm{d}S_\mathrm{T}(p)}{\mathrm{d}p} = \frac{p}{S_\mathrm{T}(p)}, \quad \sqrt{n_1^2 - S_\mathrm{T}(p)^2} = \sqrt{n_0^2 - p^2},$$

where the latter relation simply describes preservation of p_z. With these relations we obtain

$$\int_{\bar{p}_1}^{\bar{p}_2} \rho(z^+, q_0^+, \bar{p}) \frac{\bar{p}}{\sqrt{n_1^2 - \bar{p}^2}} \,\mathrm{d}\bar{p} = \int_{p_1}^{p_2} \rho(z^-, q_0^-, p) \frac{p}{\sqrt{n_0^2 - p^2}} \,\mathrm{d}p, \tag{4.30a}$$

and similarly for reflection, the constraint for R_R given by (4.28) reduces to

$$\int_{\bar{p}_3}^{\bar{p}_4} \rho(z^+, q_0^+, \bar{p}) \frac{\bar{p}}{\sqrt{n_0^2 - \bar{p}^2}} \,\mathrm{d}\bar{p} = \int_{p_3}^{p_4} \rho(z^-, q_0^-, p) \frac{p}{\sqrt{n_0^2 - p^2}} \,\mathrm{d}p. \tag{4.30b}$$

Here relations (4.30) describe the (local) balances of fluxes at an optical interface.

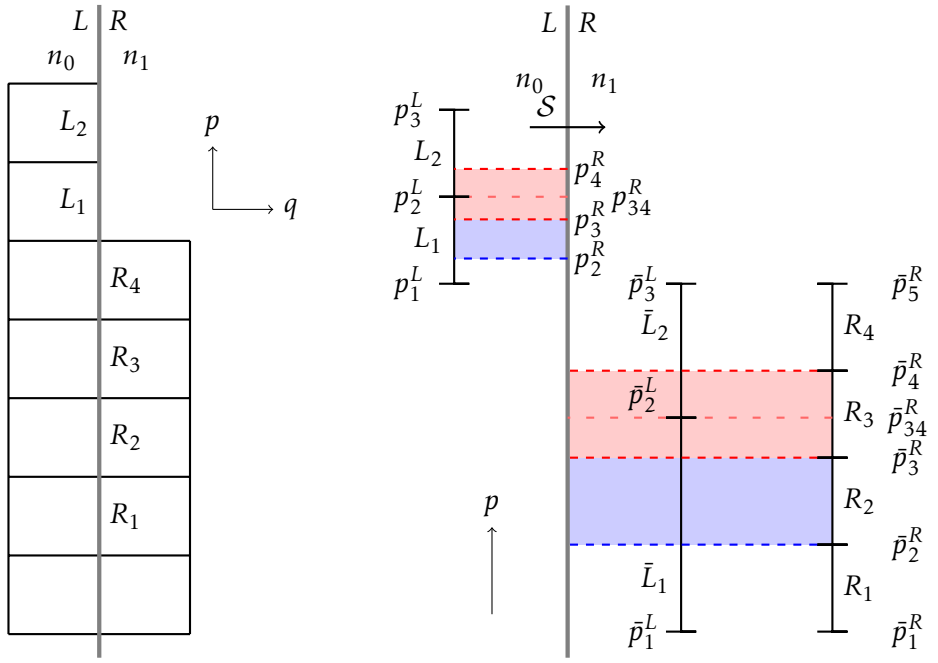
The balances (4.30) have to be combined with relation (4.23) to ensure the scheme conserves energy. However, the coupling between edges belonging to the line segments L and R is not straightforward, which will be discussed in the next section.

4.3.2 Geometric connectivity

First, consider only the refractive part of the optical interface. Elements adjacent to the optical interface in phase space are shown in Figure 4.3a. These elements have edges on the optical interface and they are denoted by L_i ($i = 1, 2$) and R_j ($j = 1, 2, 3, 4$). Due to refraction, the value of ρ in the elements that contain R_j as an edge is determined by the flow through the elements that contain L_1 and L_2. In fact, taking a closer look at how Snell's law connects the line segments from L to R in momentum space at the optical interface, we obtain for example Figure 4.3b. In Figure 4.3b L_i and R_j denote line segments along the optical interface. The basic luminance ρ along the line segments L_i and R_j are represented by their inner-element solution evaluated at the optical interface. To simplify notation, we denote these polynomials along the optical interface by $\rho^{L_i}(p)$ with $i = 1, 2$ and $\rho^{R_j}(p)$ with $j = 1, 2, 3, 4$. For example:

$$\rho^{L_i}(p) = \sum_{j=0}^{N} \rho_j^{L_i} \ell_j\big(\zeta(p)\big), \tag{4.31}$$

where $\zeta = \zeta(p) \in [0, 1]$ denotes the line segment's local reference coordinate along the interface. Similar to (4.31) we will use $F^{L_i}(p)$ to denote the polynomial describing the flux for line segment L_i, etc. In the discretisation to be described, we will transform these polynomials defined over line segments to



(a) Elements in phase space connected due to law of refraction.

(b) Illustration of the geometry at the optical interface.

Figure 4.3: Conservative handling of fluxes. The incident and transmitted momenta are related by Snell's law of refraction.

the unit reference interval $[0,1]$ and will denote the equivalent polynomial of (4.31) with a reference coordinate as argument as

$$\rho^{L_i}(\zeta) = \sum_{j=0}^{N} \rho_j^{L_i} \ell_j(\zeta). \tag{4.32}$$

The transformation from the line segment to the unit reference interval is a straightforward affine transformation. From the context it should be clear whether we evaluate the polynomial in terms of momentum or in terms of its reference coordinates.

In Figure 4.3b also virtual line segments $\bar{L}_i$ are shown. The virtual line segment $\bar{L}_i$ is the image of L_i under S, i.e.,

$$\bar{L}_i = S(L_i). \tag{4.33}$$

Hence, the endpoints of these line segments are found by applying S to the endpoints of L_i, i.e., $\bar{p}_i^L = S(p_i^L)$. Note that due to Snell's law, the line segments

L_i are stretched or compressed in the momentum direction. Computing the endpoints $\bar{p}_i^L$ allows us to determine which line segments before the optical interface contribute to a single line segment after the optical interface. From the figure we see that part of L_1 contributes to R_2, given by the blue coloured region. Therefore, a relation connecting $\rho^{L_1}(p)$ and $\rho^{R_2}(p)$ on opposite sides of the optical interface must be found. Hence, as a first step applying relation (4.23) to a polynomial on L_i, allows us to find the corresponding ρ on $\bar{L}_i$, i.e.,

$$\rho^{\bar{L}_i}(\bar{p}) = \rho^{L_i}\left(S^{-1}(\bar{p})\right) = \rho^{L_i}(p), \quad \text{with } \bar{p} = S(p). \tag{4.34}$$

The coupling between line segments that do not exactly match, as shown in Figure 4.3b, is similar to what is known as a geometrically non-conforming mesh [55, 61]. In [61] the authors describe a discontinuous Galerkin method for non-conforming meshes, applied to Maxwell's equations that form a hyperbolic system of PDEs. In their approach for non-conforming interfaces the solutions are first transferred to an intermediate construct called a 'mortar', and on this mortar the numerical fluxes are computed and transferred back to the corresponding elements. The transfer of the solutions and numerical fluxes is done using a least-squares matching, with integrals evaluated using Gauss-Legendre quadrature [12].

We will take a slightly different approach since in Liouville's equation for optics the flux $\boldsymbol{f}$ is discontinuous across an optical interface. Relation (4.34) describes how ρ transforms across an optical interface. For this reason, we use a least-squares matching of the polynomials describing ρ along either side of the interface with the function S directly incorporated. An additional constraint is used to satisfy energy conservation.

For the reflective part of a flat interface that is parallel to the z-axis, $\bar{p} = S(p)$ reduces to $\bar{p} = -p$; see (4.29). The conservative treatment of these types of optical interfaces is easily accommodated by choosing a mesh such that the elements and nodes are symmetric with respect to the line $p = 0$, and the constraint (4.30b) is easily satisfied. Due to this choice of mesh each node $\bar{p}_j \in R_{\mathrm{R}}$ will exactly correspond to $-\bar{p}_j = p_j \in L$ and a point-by-point transfer of ρ can be made. From now on, we will present the method considering only refraction.

4.3.3 Contribution from one element

From Figure 4.3b we see that the line segment R_2 only depends on the solution in L_1. The polynomial ρ^{R_2} must thus be computed from the polynomial ρ^{L_1} with the additional constraint of energy conservation. That is, the integral of the flux within the blue interval on either side of the optical interface

should be equal, analogous to equation (4.30a). Therefore, the constrained least-squares approximation reads

$$\min_{\rho^{R_2}\in\mathbb{P}_N} \quad \int_{\bar{p}_2^R}^{\bar{p}_3^R} \left[\rho^{R_2}(\bar{p}) - \rho^{\bar{L}_1}(\bar{p})\right]^2 \mathrm{d}\bar{p}, \tag{4.35a}$$

$$\text{subject to} \quad \int_{\bar{p}_2^R}^{\bar{p}_3^R} F^{R_2}(\bar{p})\mathrm{d}\bar{p} = \int_{p_2^R}^{p_3^R} F^{L_1}(p)\,\mathrm{d}p. \tag{4.35b}$$

Here, $[p_2^R, p_3^R] \subseteq L_1 = [p_1^L, p_2^L]$ and the momenta on both sides are related by $p_i^R = S^{-1}(\bar{p}_i^R)$; see Figure 4.3b. Furthermore, the numerical fluxes are defined as expansions in the Lagrange polynomial basis on Gauss-Legendre nodes, similar to (4.31), with flux coefficients $F_j = u_j\rho_j$. The minimisation of the integral in (4.35a) requires finding a polynomial that matches in the least-squares sense, while the constraint (4.35b) ensures that the scheme conserves energy.

The integrals in the constrained minimisation problem (4.35) are transformed to reference line segments. Specifically, the integral on the left-hand side of (4.35b) and the integral in (4.35a) are transformed to the reference line segment along R_2, while the integral on the right-hand side of (4.35b) is transformed to the reference line segment along L_1. Omitting the element's subscripts, applying relation (4.34) and introducing an auxiliary function Ξ, we obtain

$$\begin{aligned} \min_{\rho^R\in\mathbb{P}_N} \quad & \int_0^1 \left[\rho^R(\zeta) - \rho^L\big(\Xi(\zeta)\big)\right]^2 \mathrm{d}\zeta, \\ \text{subject to} \quad & \Delta\bar{p}^R \int_0^1 F^R(\zeta)\,\mathrm{d}\zeta = \Delta p^L \int_{\sigma^L}^{\sigma^L+\lambda^L} F^L(\zeta)\,\mathrm{d}\zeta, \end{aligned} \tag{4.36}$$

where $\Delta\bar{p}^R = \bar{p}_3^R - \bar{p}_2^R$ and $\Delta p^L = p_2^L - p_1^L$. Furthermore, the coefficients $\sigma^L \in [0,1]$ and $\lambda^L \in [0,1]$ denote the offset and scaling in L_1's reference frame, so that $p(\sigma^L) = p_2^R$ and $p(\sigma^L + \lambda^L) = p_3^R$ in L_1. Finally, the auxiliary function Ξ reads

$$\Xi(\zeta; p^L, \Delta p^L, \bar{p}^R, \Delta\bar{p}^R) = \frac{S^{-1}\left(\bar{p}^R + \zeta\,\Delta\bar{p}^R\right) - p^L}{\Delta p^L}, \tag{4.37}$$

for which we use the shorthand notation $\Xi(\zeta)$ in (4.36). This function relates the reference frame coordinates for the momentum interval $[\bar{p}^R, \bar{p}^R + \Delta\bar{p}^R]$ past the optical interface to the reference frame coordinates on the momentum interval $[p^L, p^L + \Delta p^L]$ before the optical interface.

Next, we write the constrained minimisation problem (4.36) in terms of a Lagrange function $\mathcal{L}$ with a Lagrange multiplier μ for the energy conservation

constraint, i.e.,

$$\begin{aligned}\mathcal{L} = \tfrac{1}{2}\int_0^1 \left[\rho^R(\zeta) - \rho^L\Big(\Xi(\zeta)\Big)\right]^2 \mathrm{d}\zeta \\ + \mu\left[\Delta\bar{p}^R \int_0^1 F^R(\zeta)\,\mathrm{d}\zeta - \Delta p^L \int_{\sigma^L}^{\sigma^L+\lambda^L} F^L(\zeta)\,\mathrm{d}\zeta\right].\end{aligned} \tag{4.38}$$

The coefficients ρ_j^R for the polynomial $\rho^R \in \mathbb{P}_N$ can then be computed by solving

$$\frac{\partial \mathcal{L}}{\partial \rho_i^R} = 0, \quad \text{for } i = 0,\ldots,N, \tag{4.39a}$$

$$\frac{\partial \mathcal{L}}{\partial \mu} = 0. \tag{4.39b}$$

Recalling that both F^L and F^R are written as expansions in the Lagrange polynomial basis on Gauss-Legendre nodes, we obtain for the energy conservation constraint (4.39b)

$$\Delta\bar{p}^R \int_0^1 \sum_{j=0}^{N} u_j^R \rho_j^R \ell_j(\zeta)\,\mathrm{d}\zeta = \Delta p^L \int_{\sigma^L}^{\sigma^L+\lambda^L} \sum_{j=0}^{N} u_j^L \rho_j^L \ell_j(\zeta)\,\mathrm{d}\zeta, \tag{4.40}$$

where we have used $F_j = u_j \rho_j$.

To evaluate the second integral, we transform it to the reference interval $[0,1]$ using $\zeta(\xi) = \sigma^L + \xi\lambda^L$. Next, we replace both integrals with Gauss-Legendre quadrature to find the exact values, since both integrands are at most Nth degree polynomials. The result is

$$\Delta\bar{p}^R \sum_{j=0}^{N} w_j u_j^R \rho_j^R = \Delta p^L\, \lambda^L \sum_{j=0}^{N} u_j^L \rho_j^L \sum_{k=0}^{N} w_k \ell_j\Big(\sigma^L + \xi_k \lambda^L\Big), \tag{4.41}$$

with ξ_k and w_k the Gauss-Legendre nodes and weights, respectively.

Recalling that the polynomials ρ^L and ρ^R are written as an expansion in a Lagrange polynomial basis, cf. (4.31), we can rewrite the equations (4.39a) to

$$0 = \int_0^1 \left[\rho^R(\zeta) - \rho^L\Big(\Xi(\zeta)\Big)\right]\ell_i(\zeta)\,\mathrm{d}\zeta + \mu\Delta\bar{p}^R \int_0^1 u_i^R \ell_i(\zeta)\,\mathrm{d}\zeta, \quad \text{for } i = 0,\ldots,N. \tag{4.42}$$

To evaluate the first integral, we introduce a generic auxiliary variable S_{ij}, given by

$$S_{ij}\left(\sigma^R, \lambda^R\right) = \int_{\sigma^R}^{\sigma^R+\lambda^R} \ell_i(\zeta)\ell_j(\Xi(\zeta))\,\mathrm{d}\zeta, \tag{4.43}$$

with Ξ defined in (4.37). The integral is evaluated by transforming to the reference interval and subsequently applying Gauss-Legendre quadrature. The integration interval $[\sigma^R, \sigma^R + \lambda^R]$ of S_{ij} depends on how large R_j is compared to $\bar{L}_i$. In this case, the entire line segment R_2 fits in $\bar{L}_1$, therefore, the integral over R_2 is transformed to a reference line segment $[0,1]$, corresponding to $\sigma^R = 0$ and $\lambda^R = 1$.

The integrals in (4.42) are evaluated using Gauss-Legendre quadrature which results in

$$\sum_{j=0}^{N} M_{ij}\rho_j^R + \mu\Delta\bar{p}^R u_i^R w_i = \sum_{j=0}^{N} S_{ij}(0,1)\,\rho_j^L, \quad \text{for } i = 0,\dots,N, \tag{4.44a}$$

with

$$M_{ij} = \int_0^1 \ell_i(\zeta)\ell_j(\zeta)\,\mathrm{d}\zeta, \tag{4.44b}$$

and S_{ij} given by (4.43). Using the orthogonality of the Lagrange polynomials on Gauss-Legendre nodes we find, cf. (3.21), that

$$M_{ij} = w_i\delta_{ij}. \tag{4.45}$$

The coefficients M_{ij} and S_{ij} are elements of the matrices $\boldsymbol{M}, \boldsymbol{S} \in \mathbb{R}^{(N+1)\times(N+1)}$. The matrix $\boldsymbol{M}$ is a diagonal matrix containing the Gauss-Legendre quadrature weights, i.e., $\boldsymbol{M} = \mathrm{diag}(\boldsymbol{w})$ with $\boldsymbol{w} = (w_0, w_1, \dots, w_N)^{\mathrm{T}}$. Here the superscript T denotes the transpose.

By defining

$$\alpha_j^R = \Delta\bar{p}^R u_j^R, \quad \beta_j^L = \Delta p^L\,\lambda^L u_j^L \sum_{k=0}^{N} w_k \ell_j\left(\sigma^L + \xi_k\lambda^L\right), \tag{4.46}$$

we can write (4.41) as

$$\sum_{j=0}^{N} w_j\alpha_j^R\rho_j^R = \sum_{j=0}^{N}\beta_j^L\rho_j^L. \tag{4.47}$$

Here, α_j^R and β_j^L describe the components of the vectors $\boldsymbol{\alpha}^R$ and $\boldsymbol{\beta}^L$. We can write the linear system given by (4.44) and (4.47) for $\boldsymbol{\rho}^R = (\rho_0^R, \rho_1^R, \dots, \rho_N^R)^{\mathrm{T}}$ and μ compactly in matrix-vector form:

$$\begin{pmatrix} \mathrm{diag}(\boldsymbol{w}) & \boldsymbol{\alpha}^R \circ \boldsymbol{w} \\ \left(\boldsymbol{\alpha}^R \circ \boldsymbol{w}\right)^{\mathrm{T}} & 0 \end{pmatrix} \begin{pmatrix} \boldsymbol{\rho}^R \\ \mu \end{pmatrix} = \begin{pmatrix} \boldsymbol{S} \\ \left(\boldsymbol{\beta}^L\right)^{\mathrm{T}} \end{pmatrix} \boldsymbol{\rho}^L, \tag{4.48}$$

where we take the arguments for $\boldsymbol{S}$ as understood. Furthermore, $\circ$ denotes the Hadamard product between two vectors, e.g., let $\boldsymbol{a} = (a_0, a_1, \ldots, a_N)^{\mathrm{T}}$ and $\boldsymbol{b} = (b_0, b_1, \ldots, b_N)^{\mathrm{T}}$ then $\boldsymbol{a} \circ \boldsymbol{b} = (a_0 b_0, a_1 b_1, \ldots, a_N b_N)^{\mathrm{T}}$. Let $\boldsymbol{A}$ denote the matrix on the left-hand side, i.e.,

$$\boldsymbol{A} = \begin{pmatrix} \operatorname{diag}(\boldsymbol{w}) & \boldsymbol{\alpha}^R \circ \boldsymbol{w} \\ \left(\boldsymbol{\alpha}^R \circ \boldsymbol{w}\right)^{\mathrm{T}} & 0 \end{pmatrix}. \tag{4.49}$$

The determinant of this matrix reads

$$\det(\boldsymbol{A}) = -\left(\sum_{i=0}^{N} \left(\alpha_i^R\right)^2 w_i \right) \prod_{i=0}^{N} w_i, \tag{4.50}$$

see Appendix A for a derivation. From expression (4.50) it can readily be seen that the matrix is only singular if all coefficients satisfy $\alpha_i^R = 0$, or equivalently if all velocities satisfy $u_i = 0$, which would mean no flux can enter the element from that side. Hence, we can safely assume that the matrix $\boldsymbol{A}$ is regular.

An analytical inverse for the matrix $\boldsymbol{A}$ is derived in Appendix A, and reads

$$\boldsymbol{A}^{-1} = \frac{1}{r} \begin{pmatrix} \boldsymbol{B} & -\boldsymbol{\alpha}^R \\ \left(-\boldsymbol{\alpha}^R\right)^{\mathrm{T}} & 1 \end{pmatrix}, \quad r = -\sum_{i=0}^{N} \left(\alpha_i^R\right)^2 w_i, \tag{4.51}$$

where the coefficients of the matrix $\boldsymbol{B}$ read

$$B_{ij} = \begin{cases} \left(\alpha_i^R\right)^2 + \frac{r}{w_i} & \text{if } i = j, \\ \alpha_i^R \alpha_j^R & \text{if } i \neq j. \end{cases} \tag{4.52}$$

Now, we can directly obtain an expression for the Dirichlet boundary condition values $\boldsymbol{\rho}^R$ in terms of $\boldsymbol{\rho}^L$, i.e.,

$$\boldsymbol{\rho}^R = \frac{1}{r} \begin{pmatrix} \boldsymbol{B} & -\boldsymbol{\alpha}^R \end{pmatrix} \begin{pmatrix} \boldsymbol{S} \\ \left(\boldsymbol{\beta}^L\right)^{\mathrm{T}} \end{pmatrix} \boldsymbol{\rho}^L =: \boldsymbol{C} \boldsymbol{\rho}^L. \tag{4.53}$$

Note that for problems where the refractive index n does not depend on z, the coefficient matrix $\boldsymbol{C}$ relating $\boldsymbol{\rho}^R$ and $\boldsymbol{\rho}^L$ can be pre-computed and re-used during integration along the z-axis.

4.3.4 Contributions from multiple elements

From Figure 4.3b we see that the element R_3 depends on both $\bar{L}_1$ and $\bar{L}_2$. The idea remains the same, i.e., to use a least-squares matching with a constraint to

ensure that the scheme is energy conservative. The constrained least-squares problem for R_3 reads

$$\min_{\rho^{R_3}\in\mathbb{P}_N} \quad \int_{\bar{p}_3^R}^{\bar{p}_4^R} \left[\rho^{R_3}(\bar{p}) - \rho^{\bar{L}}(\bar{p})\right]^2 \mathrm{d}\bar{p}, \tag{4.54a}$$

$$\text{subject to} \quad \int_{\bar{p}_3^R}^{\bar{p}_4^R} F^{R_3}(\bar{p})\,\mathrm{d}\bar{p} = \int_{p_3^R}^{p_{34}^R} F^{L_1}(p)\,\mathrm{d}p + \int_{p_{34}^R}^{p_4^R} F^{L_2}(p)\,\mathrm{d}p, \tag{4.54b}$$

where $p_{34}^R = S^{-1}(\bar{p}_{34}^R)$ and $\bar{p}_{34}^R$ is the momentum value where the intervals $\bar{L}_1$ and $\bar{L}_2$ meet; see Figure 4.3b. Furthermore, $\rho^{\bar{L}}$ contains the contributions from ρ^{L_1} and ρ^{L_2}, and is defined by

$$\rho^{\bar{L}}(\bar{p}) = \begin{cases} \rho^{L_1}\left(S^{-1}(\bar{p})\right) & \text{for } \bar{p}_3^R \le \bar{p} \le \bar{p}_{34}^R, \\ \rho^{L_2}\left(S^{-1}(\bar{p})\right) & \text{for } \bar{p}_{34}^R < \bar{p} \le \bar{p}_4^R. \end{cases} \tag{4.55}$$

The integrals in (4.54) are transformed to their respective line segments, e.g., the integral on the left-hand side of (4.54b) and the integral in (4.54a) are transformed to the reference interval $[0,1]$ along R_3, such that we obtain

$$\min_{\rho^R\in\mathbb{P}_N} \quad \int_0^1 \left[\rho^R(\zeta) - \rho^L\left(\Xi^L(\zeta)\right)\right]^2 \mathrm{d}\zeta \tag{4.56a}$$

$$\begin{aligned} \text{subject to} \quad \Delta\bar{p}^R \int_0^1 F^R(\zeta)\,\mathrm{d}\zeta = \Delta p^{L_1} \int_{\sigma^{L_1}}^{\sigma^{L_1}+\lambda^{L_1}} F^{L_1}(\zeta)\,\mathrm{d}\zeta \\ + \Delta p^{L_2} \int_{\sigma^{L_2}}^{\sigma^{L_2}+\lambda^{L_2}} F^{L_2}(\zeta)\,\mathrm{d}\zeta, \end{aligned} \tag{4.56b}$$

with

$$\Xi^L(\zeta) = \begin{cases} \Xi\left(\zeta; p_1^L, \Delta p^{L_1}, \bar{p}_3^R, \bar{p}_{34}^R - \bar{p}_3^R\right) & \text{for } 0 \le \zeta \le \kappa, \\ \Xi\left(\zeta; p_2^L, \Delta p^{L_2}, \bar{p}_{34}^R, \bar{p}_4^R - \bar{p}_{34}^R\right) & \text{for } \kappa < \zeta \le 1, \end{cases} \tag{4.56c}$$

where we write R instead of R_3 for brevity, and $\kappa \in [0,1]$ is defined such that $p(\kappa) = \bar{p}_{34}^R$ in R_3 and $\Delta\bar{p}^R = \bar{p}_4^R - \bar{p}_3^R$, $\Delta p^{L_1} = p_2^L - p_1^L$ and $\Delta p^{L_2} = p_3^L - p_2^L$. Here, σ^{L_i} and λ^{L_i} for $i = 1, 2$, again denote the offset and scaling on the reference line segment and are defined by using the appropriate affine transformations. Note that $\sigma^{L_1} + \lambda^{L_1} = 1$ and $\sigma^{L_2} = 0$, however, for illustration purposes we will keep using the variables rather than these values. The Lagrange function $\mathcal{L}$

for this constrained minimisation problem reads

$$\begin{aligned}\mathcal{L} = \tfrac{1}{2}\int_0^1 \left[\rho^R(\zeta) - \rho^L\left(\Xi^L(\zeta)\right)\right]^2 \mathrm{d}\zeta + \mu\Bigg[\Delta\bar{p}^R \int_0^1 F^R(\zeta)\,\mathrm{d}\zeta \\ -\Delta p^{L_1}\int_{\sigma^{L_1}}^{\sigma^{L_1}+\lambda^{L_1}} F^{L_1}(\zeta)\,\mathrm{d}\zeta - \Delta p^{L_2}\int_{\sigma^{L_2}}^{\sigma^{L_2}+\lambda^{L_2}} F^{L_2}(\zeta)\,\mathrm{d}\zeta\Bigg].\end{aligned} \tag{4.57}$$

The coefficients ρ_j^R for the polynomial $\rho^R \in \mathbb{P}_N$ can be found by solving

$$\begin{aligned}\frac{\partial \mathcal{L}}{\partial \rho_i^R} &= 0, \quad \text{for } i = 0,\dots,N,\\ \frac{\partial \mathcal{L}}{\partial \mu} &= 0.\end{aligned}$$

Following the same steps as in Section 4.3.3, we obtain the system of equations

$$\sum_{j=0}^{N} M_{ij}\rho_j^R + \mu w_i \alpha_i^R = \sum_{j=0}^{N}\left[S_{ij}(0,\kappa)\rho_j^{L_1} + S_{ij}(\kappa,1-\kappa)\rho_j^{L_2}\right], \quad \text{for } i = 0,\dots,N, \tag{4.58a}$$

$$\sum_{j=0}^{N} w_j \alpha_j^R \rho_j^R = \sum_{j=0}^{N}\beta_j^{L_1}\rho_j^{L_1} + \sum_{j=0}^{N}\beta_j^{L_2}\rho_j^{L_2}, \tag{4.58b}$$

with

$$\alpha_j^R = \Delta\bar{p}^R u_j^R, \tag{4.58c}$$

$$\beta_j^{L_1} = \Delta p^{L_1}\lambda^{L_1} u_j^{L_1}\sum_{k=0}^{N} w_k \ell_j\left(\sigma^{L_1} + \xi_k\lambda^{L_1}\right), \tag{4.58d}$$

$$\beta_j^{L_2} = \Delta p^{L_2}\lambda^{L_2} u_j^{L_2}\sum_{k=0}^{N} w_k \ell_j\left(\sigma^{L_2} + \xi_k\lambda^{L_2}\right). \tag{4.58e}$$

The linear system described by (4.58) can once again be assembled into a matrix-vector form:

$$\begin{pmatrix}\mathrm{diag}(\boldsymbol{w}) & \boldsymbol{\alpha}^R \circ \boldsymbol{w}\\ \left(\boldsymbol{\alpha}^R \circ \boldsymbol{w}\right)^{\mathrm{T}} & 0\end{pmatrix}\begin{pmatrix}\boldsymbol{\rho}^R\\ \mu\end{pmatrix} = \begin{pmatrix}\boldsymbol{S}^{L_1}\\ \left(\boldsymbol{\beta}^{L_1}\right)^{\mathrm{T}}\end{pmatrix}\boldsymbol{\rho}^{L_1} + \begin{pmatrix}\boldsymbol{S}^{L_2}\\ \left(\boldsymbol{\beta}^{L_2}\right)^{\mathrm{T}}\end{pmatrix}\boldsymbol{\rho}^{L_2}, \tag{4.59}$$

where we have used the shorthand notation $\boldsymbol{S}^{L_1} = \left(S_{ij}(0,\kappa)\right)$ and $\boldsymbol{S}^{L_2} = \left(S_{ij}(\kappa,1-\kappa)\right)$. Note that the matrix on the left-hand side is exactly the same as the matrix

obtained in the previous section, except for possibly different values for α_j^R. Therefore, we can again solve the linear system explicitly for the Dirichlet boundary condition values $\boldsymbol{\rho}^{\boldsymbol{R}}$, resulting in

$$\boldsymbol{\rho}^R = \frac{1}{r}\begin{pmatrix}\boldsymbol{B} & -\boldsymbol{\alpha}^R\end{pmatrix}\left[\begin{pmatrix}\boldsymbol{S}^{L_1}\\ \left(\boldsymbol{\beta}^{L_1}\right)^{\mathrm{T}}\end{pmatrix}\boldsymbol{\rho}^{L_1} + \begin{pmatrix}\boldsymbol{S}^{L_2}\\ \left(\boldsymbol{\beta}^{L_2}\right)^{\mathrm{T}}\end{pmatrix}\boldsymbol{\rho}^{L_2}\right], \tag{4.60}$$

cf. (4.53). This result can of course be generalised to K elements contributing to $\boldsymbol{\rho}^R$, resulting in

$$\boldsymbol{\rho}^R = \frac{1}{r}\begin{pmatrix}\boldsymbol{B} & -\boldsymbol{\alpha}^R\end{pmatrix}\left[\sum_{k=1}^{K}\begin{pmatrix}\boldsymbol{S}^{L_k}\\ \left(\boldsymbol{\beta}^{L_k}\right)^{\mathrm{T}}\end{pmatrix}\boldsymbol{\rho}^{L_k}\right]. \tag{4.61}$$

4.3.5 Overview

To summarise, during a z-step the numerical fluxes over the optical interface are evaluated as follows. First, the elements are identified that have an edge on the optical interface. Those elements are separated into elements with velocities directed towards the optical interface, denoted L, and elements with velocities directed away from the optical interface, denoted R. For the elements from L the solution is evaluated at edges on the optical interface. The numerical flux over the edges for the elements L can be directly computed as there is no constraint on ρ. For each element from R there is a Dirichlet boundary condition on the edge at the optical interface given by (4.23), that is incorporated into the numerical flux.

The value for the Dirichlet boundary condition is determined from the elements L as follows. To determine which elements from L contribute to a single R element, $S(p)$ is applied to the momentum boundaries of the elements L. Subsequently, the geometric quantities relating the element sizes are computed. Next, the momenta p at the quadrature nodes for evaluation of the integral S_{ij} are determined. Subsequently, we apply S^{-1} to these nodes and compute Ξ using (4.37). Hereafter, the integrals S_{ij} are evaluated and the coefficients β_j^L, α_j^R are computed. Finally, the values for the Dirichlet boundary condition can be found from their contributing L-elements by applying (4.61).

4.4 Results

Numerical experiments were performed for two examples. The first example features light propagating through a gradient-index medium. The smooth refractive index field of the medium fits naturally into the DGSEM for solving

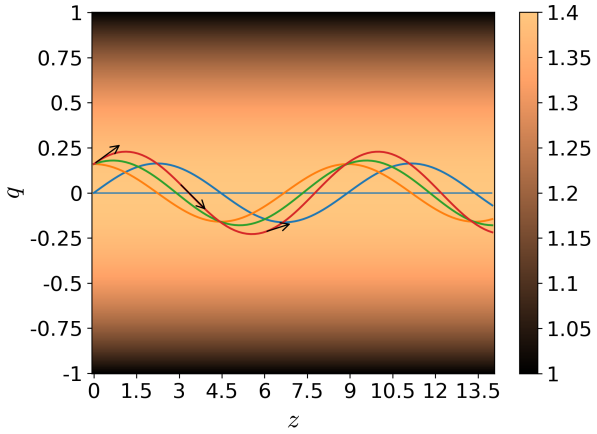


Figure 4.4: Elliptic waveguide: background colour indicates the refractive index value $n(q)$, and the solid lines represent ray trajectories. Arrows indicate the direction of the ray, i.e., the momenta (p_z, p).

Liouville's equation. For such optical systems ray-tracers usually have to resort to difficult to obtain closed-form expressions for the trajectories of the rays [3], or use symplectic integrators to solve Hamilton's equations for every ray [70]. Solving Liouville's equation with the DGSEM directly provides the energy distribution, i.e., the basic luminance ρ for the optical system. Furthermore, the method conserves energy by design.

The second example features a single optical interface. The problem exhibits both total internal reflection and refraction. At the optical interface we apply the strategy outlined in Section 4.3. Furthermore, a comparison is made between solving Liouville's equation using the DGSEM and applying quasi-Monte Carlo ray tracing [41]. The illuminance is solved using both methods and the performance of both methods is tested.

4.4.1 Elliptic waveguide

As a first example, we consider the elliptic waveguide [95] which features a smooth refractive index field given by

$$n(q) = \begin{cases} \sqrt{n_0^2 - \kappa^2 q^2} & \text{if } \kappa |q| \leq \sqrt{n_0^2 - 1}, \\ 1 & \text{otherwise.} \end{cases} \tag{4.62}$$

The parameters n_0 and κ are taken to be $n_0 = 1.4$ and $\kappa = \sqrt{n_0^2 - 1}$. The refractive index field and several rays are shown in Figure 4.4. We observe that

the elliptic waveguide contains light much like an optical fibre. Hamilton's equations (2.33) for rays inside the elliptic waveguide read

$$\begin{aligned} \frac{\mathrm{d}q}{\mathrm{d}z} &= -\frac{p}{H}, \\ \frac{\mathrm{d}p}{\mathrm{d}z} &= \frac{\kappa^2}{H} q, \end{aligned} \tag{4.63}$$

with $H = -\sqrt{n^2 - p^2}$, cf. (4.2). Since the refractive index field does not depend on z, the Hamiltonian H remains constant for each ray. The solution of (4.63) reads

$$\begin{aligned} q_{\text{exact}}(z) &= q_0 \cos\left(\frac{\kappa}{H} z\right) - \frac{p_0}{\kappa} \sin\left(\frac{\kappa}{H} z\right), \\ p_{\text{exact}}(z) &= p_0 \cos\left(\frac{\kappa}{H} z\right) + \kappa q_0 \sin\left(\frac{\kappa}{H} z\right), \end{aligned} \tag{4.64}$$

where the initial conditions are given by $(q_{\text{exact}}(0), p_{\text{exact}}(0)) = (q_0, p_0)$. Note that from the refractive index field n and the Hamiltonian H we obtain [95]

$$\kappa^2 q_{\text{exact}}^2 + p_{\text{exact}}^2 = n_0^2 - H^2, \tag{4.65}$$

where the right-hand side is constant when we move along the z-axis. We can readily see that the trajectories follow an elliptical path in phase space, hence, the name elliptic waveguide.

Let the function $\varphi_{m,k}$, with both m and k positive integers, be defined as

$$\varphi_{m,k}(x) = \begin{cases} \cos^{m+1}\left(\frac{\pi}{2} x^k\right) & \text{if } |x| < 1, \\ 0 & \text{otherwise,} \end{cases} \tag{4.66}$$

which is a C_0^m-function, meaning its first m derivatives are continuous and it has compact support. The function $\varphi_{m,k}$ is plotted in Figure 4.5 for $m = 7$ and $m = 28$ with $k = 2$. We solve Liouville's equation (4.1a) with the following initial condition

$$\rho_0(q, p) = \varphi_{m,k}\left(\frac{q}{\sigma_q}\right) \varphi_{m,k}\left(\frac{p}{\sigma_p}\right), \tag{4.67}$$

at $z = 0$ and on the boundary of the domain we leave ρ free whenever the velocity field is pointing out of the domain, otherwise we prescribe $\rho = 0$. In (4.67) we take $m = 7$, $\sigma_q = 0.25$ and $\sigma_p = 0.1$.

Next, a numerical solution to Liouville's equation is computed using the DGSEM. The ODE system (4.19) is integrated using the low-storage fourth-order Runge-Kutta method by Zingg and Chisholm [101]. The numerical

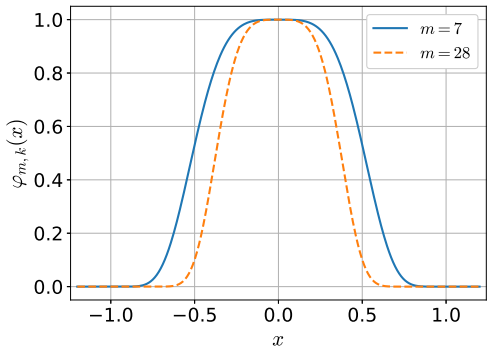


Figure 4.5: Function $\varphi_{m,k}$ for $m = 7$ and $m = 28$ with $k = 2$.

solution is integrated from $z = 0$ to $z = Z = 3$. The result using a degree 6 polynomial ($N = 6$) and $K = 16 \times 16 = 256$ rectangular elements is shown in Figure 4.6, together with the initial condition. The numerical solution at $z = Z$ has roughly the same phase space area and is approximately a rotation of the initial condition. Moreover, the maximum absolute relative deviation in energy conservation is measured. Here, the luminous flux in the solution and the luminous flux leaving the system, through the boundary of phase space, are added. This value should be equivalent to the initial luminous flux at $z = 0$ if the method is energy conservative. The luminous flux inside the domain is computed by integrating ρ over the phase space domain, i.e.,

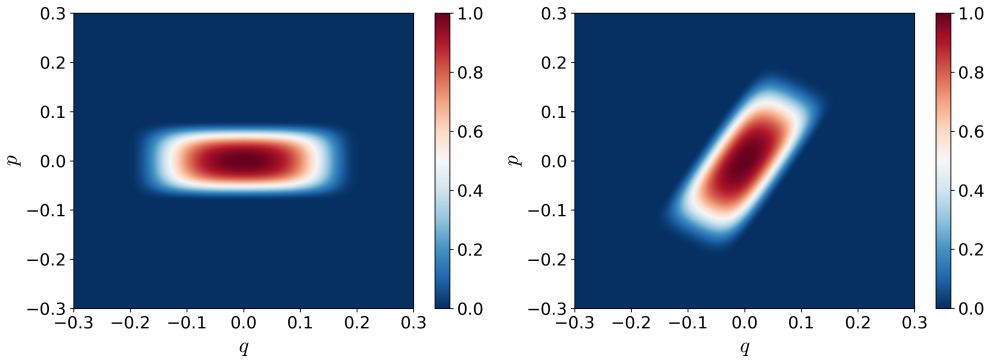
$$\int_{\mathcal{P}} \rho(z, q, p) \, \mathrm{d}\mathcal{U}. \tag{4.68}$$

The maximum absolute relative deviation of energy conservation during stepping was $1.78 \cdot 10^{-15}$, i.e., the scheme is energy conservative up to machine precision as expected.

Furthermore, a convergence test is performed by changing the number of elements K and varying the polynomial degree from $N = 1, 2, \ldots, 6$. The numerical solution is compared to the exact solution, which can be found from the trajectory of the rays given by the expressions (4.64). The expressions describe the evolution of a ray, given the initial conditions of the ray, that is, starting at $z = 0$ with (q_0, p_0) the solution is known at an arbitrary z in the point $(q_{\text{exact}}(z), p_{\text{exact}}(z))$ as

$$\rho(z, q_{\text{exact}}(z), p_{\text{exact}}(z)) = \rho_0(q_0, p_0).$$

To determine the analytical solution to Liouville's equation we apply the method of characteristics [65]. This amounts to tracing the ray backwards



(a) At $z = 0$. **(b)** At $z = Z$.

Figure 4.6: Elliptic waveguide: basic luminance distributions $\rho(z, q, p)$. Parameters are $N = 6$, $K = 256$, $Z = 3$.

starting from an arbitrary z in the point (q_0, p_0), to $z = 0$ where the coordinates are given by $(q_{\text{exact}}(-z), p_{\text{exact}}(-z))$. The exact solution, therefore, reads

$$\rho(z, q_0, p_0) = \rho_0(q_{\text{exact}}(-z), p_{\text{exact}}(-z)), \tag{4.69}$$

with $q_{\text{exact}}(z)$ and $p_{\text{exact}}(z)$ given in (4.64).

Using the exact solution (4.69) we can evaluate the discretisation error for which we take the L_1-norm, i.e.,

$$e_{\text{DG}} = \int_{\mathcal{P}} \left|\rho_{\text{DG}}(Z, q, p) - \rho(Z, q, p)\right| \mathrm{d}q\mathrm{d}p, \tag{4.70}$$

where ρ_{DG} denotes the numerical solution and ρ denotes the exact solution (4.69). The integrals in (4.70) are evaluated using Gauss-Legendre quadrature with $N + 3$ nodes on each element. We assume the convergence order γ_{DG} satisfies the empirical relation

$$e_{\text{DG}}(K) = C_{\text{DG}} K^{-\gamma_{\text{DG}}/2}, \tag{4.71}$$

with $C_{\text{DG}} > 0$ an arbitrary constant. The convergence order γ_{DG} is computed from two subsequent data points, i.e., $\gamma_{\text{DG}} = 2\log(e_{\text{DG}}(K_2)/e_{\text{DG}}(K_1))/\log(K_1/K_2)$ with K_1 and K_2 denoting the number of elements.

The convergence data is shown in Table 4.1. The spatial discretisation is done using an Nth degree polynomial, and therefore the spatial order of accuracy is $N + 1$. The temporal discretisation is done using a fourth-order explicit Runge-Kutta method, where we choose Δz to be the maximum allowed step such that the temporal integration is stable. Furthermore, a

Table 4.1: Elliptic waveguide: convergence data with L_1 denoting e_{DG} and $\mathcal{O}(L_1)$ denoting the convergence order γ_{DG}.

	$N=1$		$N=2$		$N=3$	
K	L_1	$\mathcal{O}(L_1)$	L_1	$\mathcal{O}(L_1)$	L_1	$\mathcal{O}(L_1)$
16	8.86e-03		5.47e-03		3.18e-03	
64	3.93e-03	1.17	1.71e-03	1.68	7.36e-04	2.11
256	1.62e-03	1.28	3.19e-04	2.43	3.28e-05	4.49
1024	4.27e-04	1.92	2.81e-05	3.51	2.07e-06	3.99
4096	8.32e-05	2.36	2.98e-06	3.24	1.21e-07	4.10
	$N=4$		$N=5$		$N=6$	
K	L_1	$\mathcal{O}(L_1)$	L_1	$\mathcal{O}(L_1)$	L_1	γ_{DG}
16	2.15e-03		1.17e-03		5.17e-04	
64	1.68e-04	3.68	5.82e-05	4.33	1.29e-05	5.32
256	6.37e-06	4.72	7.02e-07	6.37	9.89e-08	7.03
1024	1.56e-07	5.35	1.05e-08	6.06	6.51e-10	7.25
4096	4.34e-09	5.17	1.49e-10	6.14	4.67e-12	7.12

uniform rectangular mesh is used, where upon mesh refinement the mesh size in each direction is halved and similarly Δz is halved to ensure stability.

The global error depends on whether the spatial or temporal discretisation errors dominate. From Table 4.1, we observe that the spatial discretisation error dominates for the polynomial degrees $N = 1$ to $N = 6$. Choosing a smaller Δz-step in the numerical experiments did not influence the discretisation error. The results show that we obtain the expected $N + 1$ order of convergence.

In Figure 4.7 we show the CPU time along with the discretisation error. From the Figure it is clear that a high degree polynomial is more efficient than the lower degree counter parts.

4.4.2 Bucket of water

To illustrate that the strategy outlined in Section 4.3 for handling optical interfaces is energy conservative, we apply it to a test case. The test case 'bucket of water' introduced by van Lith et al. [91, 92] is a suitable choice. The refractive index field for this problem is given by

$$n(q) = \begin{cases} n_0, & \text{if } q \leq 0, \\ n_1, & \text{if } q > 0, \end{cases} \tag{4.72}$$

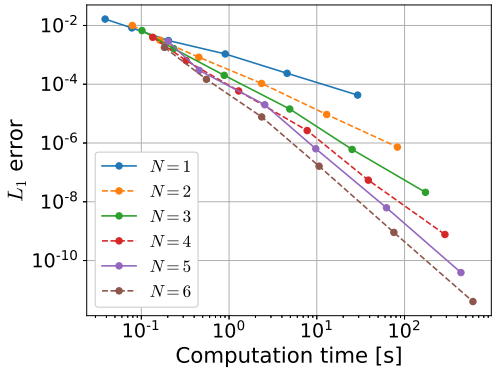


Figure 4.7: Elliptic waveguide: discretisation error L_1 as a function of CPU time.

where we take $n_0 = 1.4$ and $n_1 = 1$. Using an initial basic luminance ρ_0 that is non-zero in the region described by $q < 0$ and $p > 0$, the solution features both refraction and total internal reflection in two separate quadrants of phase space. The exact solution reads [92]

$$\rho(z,q,p) = \begin{cases} \rho_0\left(q - z\frac{p}{\sqrt{n_0^2-p^2}}, p\right) & \text{if } q<0, p\geq 0, \\ \rho_0\left(z\frac{p}{\sqrt{n_0^2-p^2}} - q, -p\right) & \text{if } q<0, -p_c<p<0, \\ \rho_0\left((\delta z - z)\frac{\bar{p}}{\sqrt{n_0^2-\bar{p}^2}}, \bar{p}\right) & \text{if } q>0, p\geq 0, \\ 0 & \text{otherwise,} \end{cases} \tag{4.73a}$$

where $p_c = \sqrt{n_0^2 - n_1^2}$, $\left(\bar{p}, \bar{p}_z\right) = -\mathcal{S}_T\left((-p, -\sqrt{n_1^2 - p^2}); n_1, n_0, -\vec{\nu}\right)$ with $\vec{\nu} = (-1, 0)$, and

$$\delta z = \frac{q}{p}\sqrt{n_1^2 - p^2}. \tag{4.73b}$$

The region described by $\{q < 0, p \geq 0\}$ features propagation through the medium with refractive index n_0. The region $\{q < 0, -p_c < p < 0\}$ describes light that was reflected at the optical interface, and the region $\{q > 0, p \geq 0\}$ describes light that was refracted.

As an initial condition we use

$$\rho_0(q,p) = \varphi_{m,k}\left(\frac{q-q_0}{\sigma_q}\right)\left[\varphi_{m,k}\left(\frac{p-p_0}{\sigma_{p,0}}\right) + \varphi_{m,k}\left(\frac{p-p_1}{\sigma_{p,1}}\right)\right], \tag{4.74}$$

with $\varphi_{m,k}$ defined in (4.66) and on the part of the boundary of the domain that is not on the optical interface, we prescribe $\rho = 0$ whenever the velocity field

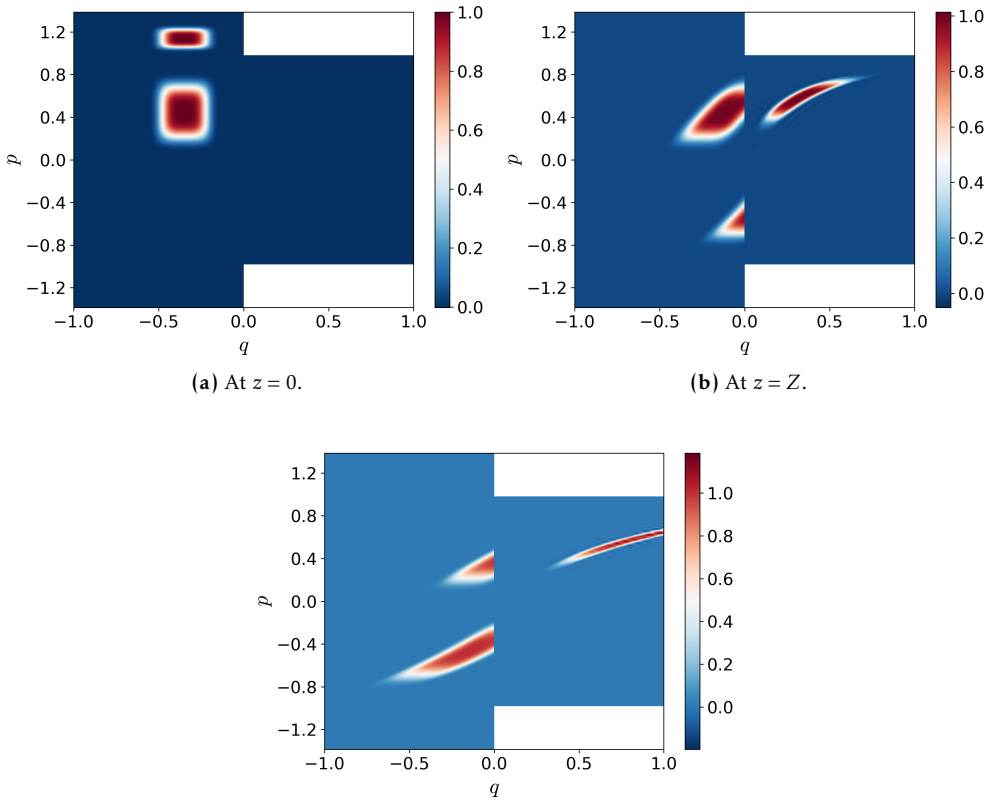
is pointing into the domain, otherwise we leave ρ free. Since the q-position is restricted to $q \in [-1, 1]$, this means that at $q = \pm 1$ we place virtual detectors that capture any luminous flux leaving the system. For the parameters in (4.74), we take $q_0 = -0.35$, $\sigma_q = 0.25$, $p_0 = 0.45$, $\sigma_{p,0} = 0.45$, $p_1 = \frac{1}{2}(1.3 + p_c)$ and $\sigma_{p,1} = 1.3 - p_1$. Furthermore, we take $m = 7$ unless specified otherwise. See Figure 4.8a.

Again, the explicit fourth-order Runge-Kutta method from the previous example is used with a constant Δz-step as determined by the stability of the temporal integration. The numerical solution is integrated from $z = 0$ to $z = Z = 0.7$ and $z = 2Z$, and is shown in Figure 4.8, together with the initial condition. The result was obtained using a degree 6 polynomial ($N = 6$) and $K = 480$ rectangular elements. The mesh uses only rectangular elements and is almost uniform. To easily treat the optical interface, we have compressed and expanded the elements below and above the critical momentum $p_c \approx 0.98$, in the p-direction such that the critical momentum is aligned with the edges of these elements. The mesh spacings for $K = 480$ are $\Delta q = 0.1$ and $\Delta p \approx 0.1$.

In second and third panels of Figure 4.8 the quadrants featuring reflection and refraction can be clearly distinguished, while the solution is, as expected, perfectly discontinuous along the optical interface $q = 0$. The solutions at $z = Z$ and $z = 2Z$ feature undershoot and overshoot, which are due to oscillations in the refracted region where the solution is under resolved. Furthermore, at $z = 2Z$ some light has passed $q = 1$, meaning some energy has hit the detectors. We observe that a total 7.5 % of the initial luminous flux has hit the detectors at $z = 2Z$. Taking into account the luminous flux on the detectors, we compute the relative error in the total luminous flux as a function of z which is plotted in Figure 4.9a. The plot shows that the method obeys energy conservation up to machine precision.

Furthermore, to show that the optical interface treatment does not incur any penalty on the convergence order, we compute the discretisation error for this example as defined in (4.70). The convergence data for $N = 1, \ldots, 6$ is shown in Table 4.2. Also for this example, we observe that the spatial discretisation error is dominant and choosing smaller Δz-steps did not result in different discretisation errors. Moreover, the expected spatial order of convergence $N + 1$ is obtained.

Next, we verify the exponential convergence of DGSEM by increasing the polynomial degree, whilst keeping the number of elements fixed to $K = 1920$ and choosing $m = 28$ in (4.74). For temporal integration a fixed number of $2 \cdot 10^4$ z-steps are performed, chosen such that the temporal integration error does not interfere with the convergence test. The result is shown in Figure 4.9b and exponential convergence is observed.

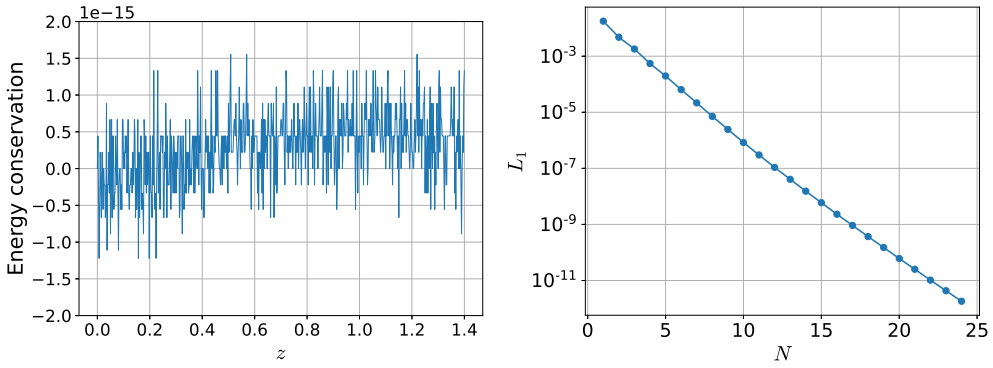


(a) At $z = 0$.

(b) At $z = Z$.

(c) At $z = 2Z$.

Figure 4.8: Bucket of water: basic luminance distributions $\rho(z, q, p)$. Parameters are $N = 6$, $K = 480$, $Z = 0.7$.



(a) Relative error in the total luminous flux, computed with $N = 6$ and $K = 480$.

(b) Polynomial degree refinement with $K = 1920$ and $m = 28$.

Figure 4.9: Bucket of water.

Table 4.2: Bucket of water: convergence data.

	$N=1$		$N=2$		$N=3$	
K	L_1	$\mathcal{O}(L_1)$	L_1	$\mathcal{O}(L_1)$	L_1	$\mathcal{O}(L_1)$
480	4.93e-02		1.71e-02		8.70e-03	
1920	1.82e-02	1.44	4.90e-03	1.80	1.23e-03	2.83
7680	6.25e-03	1.54	6.61e-04	2.89	8.07e-05	3.92
30720	1.56e-03	2.00	5.82e-05	3.50	3.71e-06	4.44
	$N=4$		$N=5$		$N=6$	
K	L_1	$\mathcal{O}(L_1)$	L_1	$\mathcal{O}(L_1)$	L_1	$\mathcal{O}(L_1)$
480	4.15e-03		2.03e-03		1.03e-03	
1920	3.55e-04	3.55	1.08e-04	4.24	3.36e-05	4.94
7680	1.17e-05	4.93	1.98e-06	5.77	3.79e-07	6.47
30720	3.08e-07	5.24	3.10e-08	6.00	3.37e-09	6.81

Comparison with ray tracing

We compare quasi-Monte Carlo ray tracing to solving Liouville's equation using the DGSEM. Solving Liouville's equation already has two advantages, i.e., it conserves energy and provides a more complete picture because we compute the basic luminance instead of its integrated quantities, the illuminance or luminous intensity. The latter advantage also comes at a price of having to solve a two-dimensional problem in phase space followed by integration to compute these quantities. Ray tracing on the other hand can directly use bins on a one-dimensional grid to compute either the illuminance or luminous intensity.

For a fair comparison, we compute the illuminance E defined by

$$E(z,q) = \int_{-n(z,q)}^{n(z,q)} \rho(z,q,p)\,\mathrm{d}p, \tag{4.75}$$

for this test case using both quasi-Monte Carlo ray tracing and the DGSEM. For quasi-Monte Carlo ray tracing we fix the number of bins to $B = 1000$ and employ a uniform grid on $q \in [-1,1]$, i.e.,

$$Q_j = (j-1)\Delta q - 1, \quad j = 1,\dots,B+1, \tag{4.76}$$

with $\Delta q = \frac{2}{B}$. The jth bin is defined by $[Q_j, Q_{j+1}]$ with midpoint $q_j = \frac{1}{2}(Q_j + Q_{j+1})$. The global error for quasi-Monte Carlo integration using a 2D Sobol

sequence behaves as $\mathcal{O}(\log(M)^2/M)$ with M the number of 2D points [37]. The 2D points are in our case the initial phase space coordinates $(q_i, p_i) \in \mathcal{P}$ of each ray. For more details on quasi-Monte Carlo integration, see [64]. In the bucket of water example $M = N_{\mathrm{RT}}$ denotes the number of rays and we use a fixed number of bins.

For the DGSEM we compute the basic luminance followed by integration over p to obtain the illuminance. Ray tracing defines an average illuminance on each bin, hence, for a fair comparison we also average the illuminance for the DGSEM when computing the discretisation error. For the discretisation error we take the L_1-norm and compare the numerical solution to the exact illuminance, which is computed by integrating the exact basic luminance (4.73) numerically up to machine precision.

Once again we take the initial condition (4.74) and (4.66) with $m = 7$. The illuminance computed using ray tracing with $N_{\mathrm{RT}} = 0.64 \cdot 10^6$ rays and the illuminance obtained with DGSEM on a mesh with $K = 480$ elements and $N = 4$ are shown in Figure 4.10a, together with the exact solution. The ray tracing (RT) solution is noisy, which is inherent to the method due to the quasi-random Monte Carlo process, while the DGSEM solution is almost indistinguishable from the exact solution.

The discretisation error for ray tracing for an increasing number of rays is shown in Table 4.3, while the results for the DGSEM with increasing number of elements K is shown in Table 4.4. In the tables e_{RT} and e_{DG} denote the errors for ray tracing and solving Liouville's equation using the DGSEM, respectively, while t_{RT} and t_{DG} denote their respective computation times using only a single core. Furthermore, γ_{RT} is estimated from the empirical relation

$$e_{\mathrm{RT}} = C_{\mathrm{RT}} N_{\mathrm{RT}}^{-\gamma_{\mathrm{RT}}}, \tag{4.77}$$

while γ_{DG} is estimated from the empirical relation (4.71).

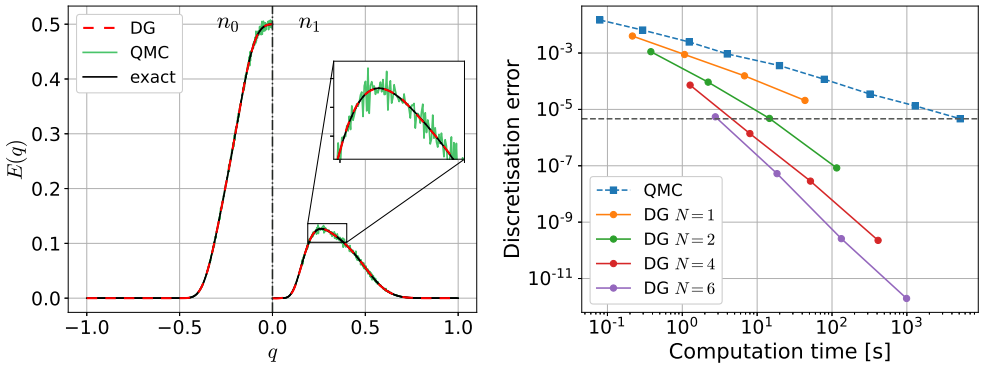
From the tables we observe that ray tracing uses $2.62 \cdot 10^9$ rays and takes almost an hour and a half, while the DGSEM achieves roughly the same accuracy in only 8.0 seconds when using 1920 elements. Varying the polynomial degree results in the performance graph shown in Figure 4.10b. It can be observed that the DGSEM always achieves a better accuracy for $N \geq 1$ compared to quasi-Monte Carlo ray tracing in the same amount of time. The DGSEM significantly outperforms ray tracing and, moreover, can achieve high accuracies in reasonable time.

Table 4.3: Bucket of water: discretisation error e_{RT}, convergence rate γ_{RT} and CPU time t_{RT} using ray tracing (RT) for computing the illuminance. Number of bins is fixed to $B = 1000$.

N_{RT} $(\cdot 10^6)$	e_{RT}	γ_{RT}	t_{RT}
0.04	1.49e-02		0.079 s
0.16	6.52e-03	0.59	0.295 s
0.64	2.46e-03	0.70	1.239 s
2.56	9.28e-04	0.70	3.996 s
10.24	3.58e-04	0.69	19.865 s
40.96	1.17e-04	0.81	1 min 19 s
163.84	3.50e-05	0.87	5 min 22 s
655.36	1.33e-05	0.70	21 min 36 s
2621.44	4.65e-06	0.76	1 h 26 min 6 s

Table 4.4: Bucket of water: discretisation error e_{DG}, convergence rate γ_{DG} and CPU time t_{DG} using the DGSEM (DG) with $N = 4$ for computing the illuminance.

K	e_{DG}	γ_{DG}	t_{DG}
480	7.28e-05		1.271 s
1920	1.39e-06	5.71	7.998 s
7680	2.86e-08	5.60	51.524 s
30720	2.26e-10	6.98	6 min 52 s



(a) Illuminance $E(q)$.

(b) Performance comparison.

Figure 4.10: Left: the illuminance computed using ray tracing (RT) with $N_{\mathrm{RT}} = 0.64 \cdot 10^6$ rays and the DGSEM with $K = 480$ and $N = 4$ (DG). Right: the error as a function of the computation time for both methods. The results were computed at $z = Z = 0.7$.

4.5 Concluding remarks

The DGSEM has been applied to solve Liouville's equation. The method clearly demonstrates high order convergence whenever the solution is sufficiently smooth. The discretisation of the non-local boundary conditions at optical interfaces was shown to be energy conservative in an example, and moreover, the expected convergence rate of the DGSEM was still observed. Furthermore, the DGSEM was compared to quasi-Monte Carlo ray tracing for computing the illuminance. For the 'bucket of water' example, the results show that the DGSEM can compute the illuminance to high accuracies in less time than ray tracing. In particular, for a fourth-degree polynomial, the DGSEM has a computation time of 8.0 seconds, while ray tracing took 1 hour and 26 minutes to achieve almost the same accuracy.

In the next chapter, we will consider curved optical interfaces. A curved optical interface is represented by a moving boundary in phase space. To deal with a moving boundary, a DG method on a moving mesh is presented. Furthermore, the discretisation of the non-local boundary conditions at an optical interface is formally extended to the general case of arbitrary curved interfaces.

Chapter 5

ADER-DG on a moving mesh for 2D optics

In this chapter, the discretisation of Liouville's equation for arbitrary curved optical interfaces is considered[1]. Curved optical interfaces manifest themselves as moving boundaries in phase space. To accommodate moving boundaries we employ an Arbitrary Lagrangian-Eulerian (ALE) formulation [60, 67]. In the ALE formulation Liouville's equation is transformed from a moving domain to a static domain with an appropriate transformation. This allows us to align the mesh with the optical interfaces.

The DG approach allows for arbitrary high order of accuracy in space. By combining DG with an Arbitrary Derivative (ADER) approach one can also achieve arbitrary high order of accuracy in the evolution coordinate. The ADER methodology was first developed for finite volume methods by Titarev and Toro in [82–84]. Later it was extended to DG schemes by Qiu et al. [74] and Dumbser et al. [32]. In those works, an element-local temporal Taylor expansion is computed where temporal derivatives are replaced with spatial derivatives using the Cauchy-Kovalewski or Lax-Wendroff procedure. This procedure becomes rather cumbersome for non-linear partial differential equations since it is problem dependent. To allow for a more general treatment a local space-time Galerkin predictor method based on a space-time weak formulation was developed by Dumbser et al. [28, 29]. For recent applications of the latter approach see for example [36, 96, 97]. For a comparison between different ADER approaches, we refer the reader to [39].

[1] This chapter is based on the published article: R. A. M. van Gestel, M. J. H. Anthonissen, J. H. M. ten Thije Boonkkamp, and W. L. IJzerman. An ADER discontinuous Galerkin method on moving meshes for Liouville's equation of geometrical optics. *Journal of Computational Physics*, 2023.

The ADER-DG schemes yield a fully-discrete explicit scheme as opposed to a semi-discrete multi-stage scheme when DG is combined with an explicit Runge-Kutta method.

In [11, 38], ADER-DG methods in the ALE formulation have been used with a local space-time Galerkin predictor. Furthermore, in [2], the authors employ an explicit Taylor series on a moving mesh for a spatially one-dimensional setting, but only for up to second order in time. Higher order in time predictors are achieved using continuous explicit Runge-Kutta schemes [71]. In this work, we employ the Cauchy-Kovalewski procedure to derive an element-local Taylor expansion on a moving mesh in a spatially two-dimensional setting, up to arbitrary order of accuracy, and we will exploit specifics of our problem to simplify the procedure. That is, a particular choice of mesh and mesh movement yields simpler computations for the considered optical systems.

The moving mesh method alone is not sufficient to solve Liouville's equation for geometrical optics numerically, as for certain optical systems it can lead to an increasingly smaller mesh spacing which via a CFL stability condition leads to an ever decreasing stepsize. Therefore, we introduce a new method, the sub-cell interface method, to resolve this issue. Similar to the ADER-DG method, it is also based on a weak formulation over a phase space element, where now an optical interface is allowed to cut the element into two pieces during a single step.

At an optical interface the non-local boundary conditions have to be incorporated into the DG scheme as numerical fluxes across the optical interface. Snell's law of refraction and the law of specular reflection both depend on the unit surface normal of the optical interface which can change for curved interfaces. This makes dealing with optical interfaces rather complicated. Moreover, for arbitrary curved interfaces light rays can have a change in their propagation direction, e.g., a light ray can go from forward to backward propagating. Here, we extend the method described in Section 4.3 by formally incorporating this change in propagation direction into the discretisation at optical interfaces. Moreover, we formulate and prove energy balances that should hold discretely for curved optical interfaces. Consequently, we are able to deal with arbitrary curved optical interfaces in an energy-conserving manner.

First, the discretisation of Liouville's equation on a moving mesh using DG is discussed in Section 5.2. Second, in Section 5.3 we develop the necessary temporal Taylor expansions used in the ADER approach. Next, we present the discretisation using the sub-cell interface method and how to deal with optical interfaces in Sections 5.4 and 5.5. As the mesh movement can cause large deformations, we briefly discuss mesh refinement in Section 5.6. Numerical

experiments and comparisons with quasi-Monte Carlo ray tracing are carried out in Section 5.7.

5.1 Liouville's equation

In what follows we will only consider two-dimensional optics and we will consider only forward-propagating light rays unless stated otherwise. Thus we take $\sigma = 1$ and omit the σ in Liouville's equation (2.40a), so that we can write equation (2.40a) as

$$\frac{\partial \rho}{\partial z} + \nabla \cdot (\rho \boldsymbol{u}) = 0, \tag{5.1}$$

with $\nabla = (\frac{\partial}{\partial q}, \frac{\partial}{\partial p})$. Furthermore, we will only consider piecewise constant refractive index fields. A curved optical interface given by $q = Q(z)$ manifests itself as a moving boundary in phase space. The phase space domain, hence, is z-dependent, so we denote the phase space domain as $\mathcal{P}(z)$.

5.2 DG on a moving mesh

We employ an Arbitrary Lagrangian-Eulerian discontinuous Galerkin (ALE-DG) method, where we can prescribe a velocity to move the mesh such that it remains aligned with optical interfaces. In other words, we consider the DG method on a moving mesh. The phase space domain $\mathcal{P}(z)$ is partitioned into Cartesian elements, where each element is a Cartesian product of one-dimensional intervals. Let $\Omega(z) = [Q_0(z), Q_1(z)] \times [P_0, P_1]$ denote one such a Cartesian element.

First, we transform Liouville's equation (5.1) to a static reference domain by considering the following transformation from the reference square $\chi = [0,1]^2$ to the element $\Omega(z)$, which reads

$$\boldsymbol{x}(\tau, \boldsymbol{\xi}) = \begin{pmatrix} q(\tau,\xi) \\ p(\eta) \end{pmatrix} = \begin{pmatrix} (1-\xi)Q_0(\tau) + \xi Q_1(\tau) \\ (1-\eta)P_0 + \eta P_1 \end{pmatrix} = \begin{pmatrix} Q_0(\tau) + \xi \Delta q(\tau) \\ P_0 + \eta \Delta p \end{pmatrix}, \tag{5.2}$$

where $\Delta q(\tau) = Q_1(\tau) - Q_0(\tau)$, $\Delta p = P_1 - P_0$, $z = \tau$ and $\boldsymbol{\xi} = (\xi, \eta)$. Let us introduce $\rho^*(\tau, \boldsymbol{\xi}) = \rho(\tau, \boldsymbol{x}(\tau, \boldsymbol{\xi}))$. The τ-derivative of ρ^* reads

$$\frac{\partial \rho^*}{\partial \tau} = \frac{\partial \rho}{\partial z}\frac{\mathrm{d}z}{\mathrm{d}\tau} + \boldsymbol{v} \cdot \nabla \rho, \tag{5.3}$$

where $\boldsymbol{v} = \frac{\partial \boldsymbol{x}}{\partial \tau}$ denotes the mesh velocity, i.e., the velocity at which we move the mesh. Subsequently, we insert Liouville's equation (5.1), use $z = \tau$ and apply

the product rule on the last term of (5.3) which leads after some rewriting to

$$\frac{\partial \rho^*}{\partial \tau} = -\nabla \cdot (\rho^*(\boldsymbol{u} - \boldsymbol{v})) - \rho^* \nabla \cdot \boldsymbol{v}. \tag{5.4}$$

The transformation (5.2) of the spatial domain to the reference domain, transforms the divergence of a general function $\boldsymbol{g} = (g_1, g_2)$ as

$$\nabla \cdot \boldsymbol{g} = \frac{1}{\mathcal{J}} \nabla_\xi \cdot \tilde{\boldsymbol{g}} \text{ with } \tilde{\boldsymbol{g}} = \begin{pmatrix} g_1 \Delta p \\ g_2 \Delta q \end{pmatrix}, \tag{5.5}$$

where $\mathcal{J}(\tau) = \Delta q(\tau) \Delta p$ denotes the Jacobian determinant of the transformation (5.2), $\nabla_\xi = (\frac{\partial}{\partial \xi}, \frac{\partial}{\partial \eta})$ denotes the gradient on the reference domain and the quantities with a tilde denote their transformed counterparts. Consequently, transforming the divergences of (5.4) to the reference domain yields

$$\frac{\partial \rho^*}{\partial \tau} = -\frac{1}{\mathcal{J}} \nabla_\xi \cdot (\rho^*(\tilde{\boldsymbol{u}} - \tilde{\boldsymbol{v}})) - \frac{\rho^*}{\mathcal{J}} \nabla_\xi \cdot \tilde{\boldsymbol{v}}. \tag{5.6}$$

Next, we multiply (5.6) by $\mathcal{J}$ and make use of the so-called geometric conservation law [67]

$$\frac{\mathrm{d}\mathcal{J}}{\mathrm{d}\tau} = \frac{\mathrm{d}}{\mathrm{d}\tau} (\Delta q(\tau) \Delta p) = \nabla_\xi \cdot \tilde{\boldsymbol{v}}, \tag{5.7}$$

so that we obtain

$$\frac{\partial (\rho^* \mathcal{J})}{\partial \tau} + \nabla_\xi \cdot (\rho^*(\tilde{\boldsymbol{u}} - \tilde{\boldsymbol{v}})) = 0. \tag{5.8}$$

The geometric conservation law states that mesh motion does not disturb a uniform solution [67], i.e., if ρ^* is uniform then (5.8) reduces to the geometric conservation law (5.7). Finally, we omit the $*$ in (5.8) and introduce the transformed flux $\tilde{\boldsymbol{f}}$, so that the conservation law on the reference domain can be written as

$$\frac{\partial (\rho \mathcal{J})}{\partial \tau} + \nabla_\xi \cdot \tilde{\boldsymbol{f}} = 0 \text{ with } \tilde{\boldsymbol{f}} = \rho(\tilde{\boldsymbol{u}} - \tilde{\boldsymbol{v}}). \tag{5.9}$$

Typically, the numerical approximation of ρ at $z = z^t$ is known and we want to evolve the numerical solution to $z = z^{t+1}$, where t denotes the step index. The DG method is based on the weak formulation. The weak formulation of equation (5.9) with test function $\phi_k = \phi_k(\boldsymbol{\xi})$ is written as

$$\int_{z^t}^{z^{t+1}} \int_\chi \phi_k \left(\frac{\partial (\rho \mathcal{J})}{\partial \tau} + \nabla_\xi \cdot \tilde{\boldsymbol{f}} \right) \mathrm{d}\xi \, \mathrm{d}\tau = 0. \tag{5.10}$$

We integrate the first term in the parenthesis with respect to τ, and we apply the product rule and Gauss's theorem to the second term on the left-hand side

of equation (5.10), which yields

$$\int_\chi (\rho\mathcal{J})^{t+1}\,\phi_k\,\mathrm{d}\boldsymbol{\xi} - \int_\chi (\rho\mathcal{J})^{t}\,\phi_k\,\mathrm{d}\boldsymbol{\xi} = \int_{z^t}^{z^{t+1}} \left(\int_\chi \left(\nabla_{\boldsymbol{\xi}}\phi_k\right)\cdot\tilde{\boldsymbol{f}}\,\mathrm{d}\boldsymbol{\xi} - \int_{\partial\chi} \phi_k \tilde{\boldsymbol{F}}\cdot\hat{\boldsymbol{N}}\,\mathrm{d}\sigma \right)\mathrm{d}\tau, \tag{5.11}$$

where we have replaced the flux in the boundary integral with a numerical flux $\tilde{\boldsymbol{F}}$ and $\hat{\boldsymbol{N}}$ denotes the outward unit normal on the reference domain χ. The numerical flux $\tilde{\boldsymbol{F}}$ depends on the left and right states at the interface denoted by ρ^- and ρ^+, respectively. For the numerical flux, we employ the upwind flux, i.e.,

$$\tilde{\boldsymbol{F}}(\rho^-,\rho^+)\cdot\hat{\boldsymbol{N}} = (\tilde{\boldsymbol{u}}-\tilde{\boldsymbol{v}})\cdot\hat{\boldsymbol{N}} \begin{cases} \rho^- & \text{if } (\tilde{\boldsymbol{u}}-\tilde{\boldsymbol{v}})\cdot\hat{\boldsymbol{N}} \geq 0, \\ \rho^+ & \text{if } (\tilde{\boldsymbol{u}}-\tilde{\boldsymbol{v}})\cdot\hat{\boldsymbol{N}} < 0. \end{cases} \tag{5.12}$$

The numerical solution on each element is represented by an expansion into basis functions. As basis functions we employ a tensor-product of one-dimensional Lagrange polynomials ℓ_i of degree N, defined on Gauss-Legendre quadrature nodes $\{\xi_i\}_{i=0}^N$ over the interval $[0,1]$. The Gauss-Legendre quadrature nodes have associated quadrature weights $\{w_i\}_{i=0}^N$. On the reference domain χ the basis functions are denoted as ϕ_l, which formally are given by

$$\phi_l(\boldsymbol{\xi}) = \ell_i(\xi)\ell_j(\eta) \text{ with } l = (N+1)j + i + 1, \tag{5.13}$$

and $i = 0,1,\ldots,N$ and $j = 0,1,\ldots,N$. The expansion of ρ in terms of these basis functions reads

$$\rho_{\mathrm{h}}(z^t,\boldsymbol{\xi}) = \sum_{l=1}^{N_d} \rho_l^t \phi_l(\boldsymbol{\xi}), \tag{5.14}$$

where $N_d = (N+1)^2$ denotes the number of degrees of freedom. Inserting the expansion (5.14) into the left-hand side of equation (5.11), yields

$$\begin{aligned} \sum_{l=1}^{N_d} \left(\int_\chi \phi_l\phi_k\,\mathrm{d}\boldsymbol{\xi} \right)\left((\rho_l\mathcal{J})^{t+1} - (\rho_l\mathcal{J})^t \right) = \\ \int_{z^t}^{z^{t+1}} \left(\int_\chi \left(\nabla_{\boldsymbol{\xi}}\phi_k\right)\cdot\tilde{\boldsymbol{f}}\,\mathrm{d}\boldsymbol{\xi} - \int_{\partial\chi} \phi_k \tilde{\boldsymbol{F}}\cdot\hat{\boldsymbol{N}}\,\mathrm{d}\sigma \right)\mathrm{d}\tau, \end{aligned} \tag{5.15}$$

with $(\rho_l\mathcal{J})^t = \rho_l^t\mathcal{J}^t$ and where $\mathcal{J}^t$ denotes the numerical approximation of $\mathcal{J}(z^t)$. The basis functions (5.13) are orthogonal with respect to the L_2-inner product on $\chi = [0,1]^2$, i.e.,

$$\int_\chi \phi_l\phi_k\,\mathrm{d}\boldsymbol{\xi} = W_k\delta_{lk}, \tag{5.16}$$

which can be derived by applying the orthogonality of the Lagrange polynomials (3.21). As a result, the coefficient W_k can be expressed as $W_k = w_i w_j$ with $\{w_i\}_{i=0}^{N}$ the Gauss-Legendre quadrature weights. Applying relation (5.16) to equation (5.15) leads to

$$W_k\left((\rho_k\mathcal{J})^{t+1} - (\rho_k\mathcal{J})^t\right) = \int_{z^t}^{z^{t+1}} \left(\int_\chi \left(\nabla_\xi \phi_k\right)\cdot \tilde{\boldsymbol{f}}\,\mathrm{d}\boldsymbol{\xi} - \int_{\partial\chi} \phi_k \tilde{\boldsymbol{F}}\cdot\hat{\boldsymbol{N}}\,\mathrm{d}\sigma\right)\mathrm{d}\tau. \quad (5.17)$$

By letting $k = 1, 2, \ldots, N_d$ we arrive at N_d equations for the expansion coefficients ρ_l^{t+1}. All the integrals in equation (5.17) are evaluated with $(N+1)$-point Gauss-Legendre quadrature. For the right-hand side of equation (5.17) we require the solution ρ at intermediate levels of $[z^t, z^{t+1}]$. In the next section, we will describe how these values are computed using the ADER approach.

In addition to solving equation (5.17), we also solve the trajectory equation for the vertices $Q_i(\tau)$ with $i = 0, 1$, i.e.,

$$\frac{\mathrm{d}Q_i}{\mathrm{d}\tau} = V_i(\tau), \quad (5.18)$$

with $V_i(\tau)$ the mesh velocity at the vertex Q_i. We solve (5.18) in the same way as equation (5.17), i.e., we compute

$$Q_i^{t+1} = Q_i^t + \int_{z^t}^{z^{t+1}} V_i(\tau)\,\mathrm{d}\tau, \quad (5.19)$$

where Q_i^t denotes the numerical approximation of $Q_i(z^t)$, etc., by applying $(N+1)$-point Gauss-Legendre quadrature. After computing the new vertex locations Q_0^{t+1} and Q_1^{t+1}, the Jacobian $\mathcal{J}$ is updated using

$$\mathcal{J}^{t+1} = (Q_1^{t+1} - Q_0^{t+1})\Delta p, \quad (5.20)$$

in agreement with solving

$$\mathcal{J}^{t+1} = \mathcal{J}^t + \Delta p \int_{z^t}^{z^{t+1}} (V_1(\tau) - V_0(\tau))\,\mathrm{d}\tau, \quad (5.21)$$

which is the integration of the geometric conservation law (5.7).

5.3 z-integration using local ADER predictor

5.3.1 Moving element

To compute the right-hand side of equation (5.17) we generally require the solution ρ at intermediate levels of $[z^t, z^{t+1}]$. In the ADER approach one

computes a predictor approximating the z-evolution locally on each element without considering neighbouring elements. In particular, we employ a Taylor expansion about the old level and subsequently apply the Cauchy-Kovalewski procedure [32, 39] where we repeatedly replace τ-derivatives with spatial derivatives using the governing equation.

The Taylor expansion up to degree M about the old level $\tau = z^t$, where the solution is known, on the reference domain reads

$$\rho(z^t + \tau, \boldsymbol{\xi}) \approx \sum_{k=0}^{M} \frac{1}{k!} \tau^k \frac{\partial^k \rho}{\partial \tau^k}(z^t, \boldsymbol{\xi}). \tag{5.22}$$

We require the governing equation of ρ on the reference domain. Therefore, we first rewrite equation (5.6) in an advective form by using the product rule on the first term on the right-hand side, so that we obtain

$$\frac{\partial \rho}{\partial \tau} = -\frac{1}{\mathcal{J}} \Big[\rho \nabla_{\boldsymbol{\xi}} \cdot (\tilde{\boldsymbol{u}} - \tilde{\boldsymbol{v}}) + (\tilde{\boldsymbol{u}} - \tilde{\boldsymbol{v}}) \cdot \nabla_{\boldsymbol{\xi}} \rho + \rho \nabla_{\boldsymbol{\xi}} \cdot \tilde{\boldsymbol{v}} \Big],$$

where $*$ is omitted. Subsequently, we apply $\mathcal{J}^{-1} \nabla_{\boldsymbol{\xi}} \cdot \tilde{\boldsymbol{u}} = \nabla \cdot \boldsymbol{u}$, cf. (5.5), and $\nabla \cdot \boldsymbol{u} = 0$, cf. (2.41), and that the last and first term in the brackets cancel. Consequently, we have

$$\frac{\partial \rho}{\partial \tau} = -\frac{1}{\mathcal{J}} (\tilde{\boldsymbol{u}} - \tilde{\boldsymbol{v}}) \cdot \nabla_{\boldsymbol{\xi}} \rho. \tag{5.23}$$

Recall that we consider only a piecewise constant refractive index field and thus by relation (2.40b) the last component of $\boldsymbol{u}$ is zero. Moreover, we consider only mesh movement with respect to the q-axis so that we can write $\boldsymbol{u} = (u_0, 0)$ and $\boldsymbol{v} = (v_0, 0)$. This allows us to rewrite relation (5.23) as

$$\frac{\partial \rho}{\partial \tau} = -\frac{1}{\Delta q} (u_0 - v_0) \frac{\partial \rho}{\partial \xi}, \tag{5.24}$$

where we have used $\tilde{\boldsymbol{u}} - \tilde{\boldsymbol{v}} = (\Delta p\,(u_0 - v_0), 0)$, cf. (5.5), and $\mathcal{J} = \Delta q \Delta p$. To simplify the notation it is convenient to write (5.24) as

$$\frac{\partial \rho}{\partial \tau} = c(\tau, \boldsymbol{\xi}) \frac{\partial \rho}{\partial \xi} \quad \text{with} \quad c(\tau, \xi) = a(\tau) + \xi\, b(\tau), \tag{5.25a}$$

where we use that the velocity field v_0 is linear in ξ, cf. (5.2), such that a and b read

$$a(\tau) = \frac{1}{\Delta q} \left(\frac{\partial Q_0}{\partial \tau} - u_0 \right) \text{ and } b(\tau) = \frac{1}{\Delta q} \frac{\partial \Delta q}{\partial \tau}, \tag{5.25b}$$

where we omit u_0's dependence on η.

From equation (5.25a) we can express higher-order τ-derivatives solely in terms of spatial derivatives as follows. First, we take the τ-derivative and the ξ-derivative of equation (5.25a) such that we obtain

$$\frac{\partial^2 \rho}{\partial \tau^2} = \frac{\partial c}{\partial \tau}\frac{\partial \rho}{\partial \xi} + c\frac{\partial^2 \rho}{\partial \xi \partial \tau},$$
$$\frac{\partial^2 \rho}{\partial \xi \partial \tau} = b\frac{\partial \rho}{\partial \xi} + c\frac{\partial^2 \rho}{\partial \xi^2}.$$

Combining both relations leads to

$$\frac{\partial^2 \rho}{\partial \tau^2} = \left(\frac{\partial c}{\partial \tau} + bc\right)\frac{\partial \rho}{\partial \xi} + c^2\frac{\partial^2 \rho}{\partial \xi^2}. \tag{5.26}$$

Similarly, expressions for higher-order τ-derivatives can be found. For higher derivatives the expressions can become rather large, however, they can still be found with the aid of a computer algebra programme. For example, the third-order derivative reads

$$\frac{\partial^3 \rho}{\partial \tau^3} = \left(\frac{\partial^2 c}{\partial \tau^2} + 2b\frac{\partial c}{\partial \tau} + \frac{\partial b}{\partial \tau}c + b^2 c\right)\frac{\partial \rho}{\partial \xi} + 3c\left(\frac{\partial c}{\partial \tau} + bc\right)\frac{\partial^2 \rho}{\partial \xi^2} + c^3\frac{\partial^3 \rho}{\partial \xi^3}. \tag{5.27}$$

Finally, we can insert the relations for the τ-derivatives into the Taylor expansion (5.22).

From an implementation point of view it is more efficient to rearrange the Taylor expansion (5.22) by expressing it in terms of ξ-derivatives, and thus reducing the number of ξ-derivative evaluations. If we consider all the τ-derivatives from order 0 to M, then we rewrite the Taylor expansion (5.22) as follows

$$\rho(z^t + \tau, \xi) \approx \sum_{k=0}^{M} \mathcal{C}_k(\tau, \xi)\frac{\partial^k \rho}{\partial \xi^k}(z^t, \xi), \tag{5.28}$$

where $\mathcal{C}_k(\tau, \xi)$ is the coefficient of the k-th ξ-derivative. For example, for $M = 3$ the coefficients $\mathcal{C}_k(\tau, \xi)$ read

$$\begin{aligned}
\mathcal{C}_0(\tau, \xi) &= 1, \\
\mathcal{C}_1(\tau, \xi) &= \tau c + \frac{\tau^2}{2}\left(\frac{\partial c}{\partial \tau} + bc\right) + \frac{\tau^3}{3!}\left(\frac{\partial^2 c}{\partial \tau^2} + 2b\frac{\partial c}{\partial \tau} + \frac{\partial b}{\partial \tau}c + b^2 c\right), \\
\mathcal{C}_2(\tau, \xi) &= \frac{\tau^2}{2}c^2 + \frac{\tau^3}{3!}3c\left(\frac{\partial c}{\partial \tau} + bc\right), \\
\mathcal{C}_3(\tau, \xi) &= \frac{\tau^3}{3!}c^3,
\end{aligned} \tag{5.29}$$

where b and c are evaluated at $(z^t, \boldsymbol{\xi})$.

Finally, we combine the Taylor expansion (5.28) with the expansion for ρ_h (5.14) to compute the spatial derivatives, completing the local ADER predictor for moving elements. Consequently, we can compute ρ at the required Gauss-Legendre quadrature points to compute the right-hand side of equation (5.17).

The complete scheme obeys a property called constant state preservation, whenever the refractive index field is constant. Constant state preservation means that a uniform solution must remain uniform, hence, obeying (5.7) and (5.8). Numerically, this means that the discretisation will exactly (up to machine precision) preserve a constant state, which needs to be independent of the mesh motion. This property is proven in Appendix B.

5.3.2 Static element

In the special case where an element does not move, i.e., $v_0(\tau, \xi) = 0$, the advection equation (5.24) reduces to

$$\frac{\partial \rho}{\partial \tau} = -\frac{u_0}{\Delta q}\frac{\partial \rho}{\partial \xi}, \tag{5.30}$$

and hence, the higher-order τ-derivatives can easily be found to be

$$\frac{\partial^k \rho}{\partial \tau^k} = \left(-\frac{u_0}{\Delta q}\right)^k \frac{\partial^k \rho}{\partial \xi^k}. \tag{5.31}$$

Hence, the Taylor expansion on a static element reads

$$\rho(z^t + \tau, \boldsymbol{\xi}) \approx \sum_{k=0}^{M} \frac{1}{k!}\left(-\frac{u_0 \tau}{\Delta q}\right)^k \frac{\partial^k \rho}{\partial \xi^k}(z^t, \boldsymbol{\xi}). \tag{5.32}$$

Another consequence of $v_0(\tau, \xi) = 0$ is that the flux $\tilde{\boldsymbol{f}}$ reduces to $\tilde{\boldsymbol{f}} = \rho \tilde{\boldsymbol{u}}$. The transformed velocity $\tilde{\boldsymbol{u}}$ does not depend on τ, since we are considering piecewise constant refractive index fields. Therefore, inserting the Taylor expansion (5.32) into the flux $\tilde{\boldsymbol{f}}$ results in a flux that is a polynomial in τ. Similarly, the numerical flux (5.12) with the expansion (5.32) is a polynomial in τ.

Once again, we insert the Taylor expansion (5.32) into the right-hand side of equation (5.17). The τ-integral can now easily be computed without quadrature since the integrand is just a polynomial in τ. Hence, for static elements we compute the τ-integral analytically.

Since no quadrature rule for the τ-integral is necessary, a static element is cheaper to update than a moving element. Moreover, the coefficients in the Taylor expansion of a static element only depend on η via u_0 and are less complicated than those in a moving element; one can see this by comparing expansion (5.32) to (5.28)-(5.29).

5.3.3 CFL condition

The ALE-ADER-DG method described thus far is an explicit one-step high order DG method. This explicit method must also obey a CFL stability condition, which imposes a condition on the stepsize Δz and reads for explicit DG schemes as [36, 96]

$$\Delta z \leq \frac{1}{2d} \frac{\mathrm{CFL}}{2N+1} \min_e \frac{h_e}{w_{\max,e}}, \tag{5.33}$$

where the minimum runs over all elements, $2d$ denotes the dimension of phase space ($d = 1$), and h_e denotes a characteristic element size for the element e and $w_{\max,e}$ denotes the maximum velocity on the element e. We refer the reader to [28] for a von Neumann stability analysis of ADER-DG schemes for a linear scalar advection equation in 1D.

Chalmers and Krivodonova [16] have shown that the CFL condition depends on the width of a cell along the characteristic direction of flow. In our case, we have $\boldsymbol{u} - \boldsymbol{v} = (u_0 - v_0, 0)$ meaning the flow direction is along the q-axis. This means that the CFL condition (5.33) effectively reduces to

$$\Delta z \leq \frac{1}{2d} \frac{\mathrm{CFL}}{2N+1} \min_e \frac{\Delta q_e}{|u_0 - v_0|_{\max,e}}. \tag{5.34}$$

The CFL condition (5.34) thus states that the mesh spacing Δp has no impact on the CFL condition, something that was shown in [16]. In our numerical experiments we employ (5.34) to compute the maximum stable stepsize Δz.

5.4 Sub-cell interface method

The moving mesh approach described in Sections 5.2 and 5.3 has its limitations regarding the optical interfaces we can solve for. In particular, by the CFL condition (5.34) the maximum stable stepsize Δz_{stable} scales with $1/\left|u_{0,\max} - v_{0,\max}\right|$. If we want to align the mesh with an optical interface given by $q = Q(z)$, then at the optical interface $v_0 = \frac{\mathrm{d}Q}{\mathrm{d}z}$. Consequently, if $\left|\frac{\mathrm{d}Q}{\mathrm{d}z}\right| \to \infty$ at some z_0-value, as for example in Figure 5.1, then as we approach z_0 we have to take rapidly decreasing stepsizes Δz by CFL condition (5.34). To resolve

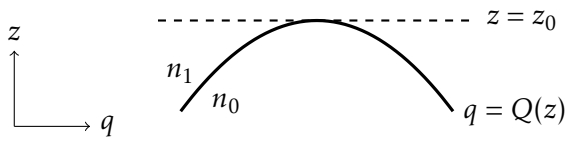
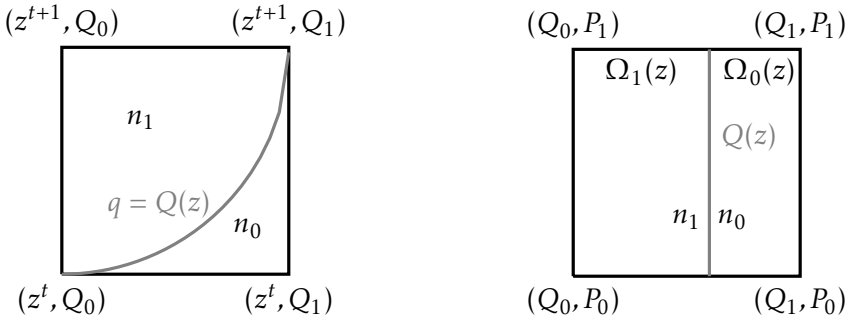


Figure 5.1: Optical interface given by $q = Q(z)$ with $\left|\frac{\mathrm{d}Q}{\mathrm{d}z}\right| \to \infty$ at z_0.



(a) Element in (q, z)-space.

(b) Element in phase space.

Figure 5.2: Two-dimensional cross sections of element T.

this problem we propose a new method, we refer to as the sub-cell interface method.

The starting point of the sub-cell interface method is to define a three-dimensional control volume $T = [z^t, z^{t+1}] \times [Q_0, Q_1] \times [P_0, P_1]$, where an optical interface separates T into two parts; see Figure 5.2. Let the optical interface be defined by $q = Q(z)$, $Q_0 \leq Q(z) \leq Q_1$, then we require the optical interface to either satisfy $Q_0 = Q(z^t)$ and $Q_1 = Q(z^{t+1})$, or $Q_1 = Q(z^t)$ and $Q_0 = Q(z^{t+1})$, such that the optical interface connects diagonally opposite corners. This ensures that there is no discontinuity in the refractive index field on the interval $[Q_0, Q_1]$ at $z = z^t$ and $z = z^{t+1}$. Further, we restrict ourselves in the following exposition to monotonic optical interfaces within the control volume, i.e., $Q(z)$ is a monotonic increasing or decreasing function. This is no real restriction as the derivation can be straightforwardly extended to deal with non-monotone optical interfaces. Other configurations where the optical interface does not connect diagonally opposite corners are also possible, however, they are not considered in this chapter.

The phase space domain of T is cut into two pieces along the optical interface where we denote the pieces with $\Omega_0(z) = [Q(z), Q_1] \times [P_0, P_1]$ and $\Omega_1(z) = [Q_0, Q(z)] \times [P_0, P_1]$; see Figure 5.2b. The two parts combined are denoted as $\Omega = \Omega_0(z) \cup \Omega_1(z) = [Q_0, Q_1] \times [P_0, P_1]$.

We start the derivation of the weak formulation by introducing the test function $\hat{\phi}_k = \hat{\phi}_k(\boldsymbol{x})$ defined on the domain Ω. The test function is related to the test function $\phi_k(\boldsymbol{\xi})$ by a transformation of Ω to the reference domain that is given by

$$\boldsymbol{x}(\boldsymbol{\xi}) = \begin{pmatrix} q(\xi) \\ p(\eta) \end{pmatrix} = \begin{pmatrix} Q_0 + \xi \Delta q \\ P_0 + \eta \Delta p \end{pmatrix}, \tag{5.35}$$

which is just the transformation (5.2) for a static element, where as before $\Delta q = Q_1 - Q_0$ and $\Delta p = P_1 - P_0$. Hence, the test function $\hat{\phi}_k$ is formally given by $\hat{\phi}_k(\boldsymbol{x}) = \phi_k\big(\boldsymbol{\xi}(\boldsymbol{x})\big)$, where $\boldsymbol{\xi}(\boldsymbol{x})$ denotes the inverse of the transformation (5.35).

Next, we multiply Liouville's equation (5.1) with the test function $\hat{\phi}_k$ and integrate over the element Ω to obtain

$$\int_\Omega \left(\frac{\partial \rho}{\partial z} + \nabla \cdot (\rho \boldsymbol{u}) \right) \hat{\phi}_k \, \mathrm{d}\mathcal{U} = 0. \tag{5.36}$$

Consider now the first term on the left-hand side of (5.36). Since ρ is discontinuous across an optical interface, we first split the domain as $\Omega = \Omega_0(z) \cup \Omega_1(z)$ and then apply Reynolds' transport theorem [62] to each part, yielding

$$\begin{aligned} \int_\Omega \frac{\partial \rho}{\partial z} \hat{\phi}_k \, \mathrm{d}\mathcal{U} &= \sum_{i=0}^{1} \int_{\Omega_i(z)} \frac{\partial \rho}{\partial z} \hat{\phi}_k \, \mathrm{d}\mathcal{U} \\ &= \frac{\mathrm{d}}{\mathrm{d}z} \int_\Omega \rho \hat{\phi}_k \, \mathrm{d}\mathcal{U} - \sum_{i=0}^{1} \int_{\partial\Omega_i(z)} \hat{\phi}_k \rho \boldsymbol{v} \cdot \hat{\boldsymbol{N}} \, \mathrm{d}\sigma, \end{aligned} \tag{5.37a}$$

with $\hat{\boldsymbol{N}}$ denoting once again the outward unit normal. The velocity $\boldsymbol{v}$ is defined as

$$\boldsymbol{v} = \begin{cases} \left(\frac{\mathrm{d}Q}{\mathrm{d}z}, 0 \right) & \text{if } q = Q(z), \\ \boldsymbol{0} & \text{otherwise}, \end{cases} \tag{5.37b}$$

i.e., the velocity $\boldsymbol{v}$ is only non-zero at the optical interface.

For the second term in (5.36) we use once again $\Omega = \Omega_0(z) \cup \Omega_1(z)$ and apply the product rule and Gauss's theorem to both parts so that we obtain

$$\sum_{i=0}^{1} \int_{\Omega_i(z)} \nabla \cdot (\rho \boldsymbol{u}) \hat{\phi}_k \, \mathrm{d}\mathcal{U} = \sum_{i=0}^{1} \int_{\partial\Omega_i(z)} \hat{\phi}_k \rho \boldsymbol{u} \cdot \hat{\boldsymbol{N}} \, \mathrm{d}\sigma - \int_{\Omega_i(z)} \left(\nabla \hat{\phi}_k \right) \cdot (\rho \boldsymbol{u}) \, \mathrm{d}\mathcal{U}. \tag{5.38}$$

Substituting (5.37a) and (5.38) into equation (5.36) yields

$$\frac{\mathrm{d}}{\mathrm{d}z}\int_\Omega \rho\hat{\phi}_k \,\mathrm{d}\mathcal{U} = \sum_{i=0}^{1}\int_{\Omega_i(z)} \left(\nabla\hat{\phi}_k\right)\cdot(\rho\boldsymbol{u})\,\mathrm{d}\mathcal{U} - \int_{\partial\Omega_i(z)} \hat{\phi}_k\rho\,(\boldsymbol{u}-\boldsymbol{v})\cdot\hat{\boldsymbol{N}}\,\mathrm{d}\sigma. \quad (5.39)$$

To complete the weak formulation we integrate over the interval $[z^t, z^{t+1}]$ yielding

$$\int_\Omega \rho^{t+1}\,\hat{\phi}_k\,\mathrm{d}\mathcal{U} - \int_\Omega \rho^t\,\hat{\phi}_k\,\mathrm{d}\mathcal{U} = \mathcal{R}_\mathrm{V} - \mathcal{R}_\mathrm{S} \quad (5.40a)$$

with the volume term $\mathcal{R}_\mathrm{V}$ and surface term $\mathcal{R}_\mathrm{S}$ defined as

$$\mathcal{R}_\mathrm{V} = \int_{z^t}^{z^{t+1}} \sum_{i=0}^{1}\int_{\Omega_i(z)} \left(\nabla\hat{\phi}_k\right)\cdot(\rho\boldsymbol{u})\,\mathrm{d}\mathcal{U}\,\mathrm{d}z, \quad (5.40b)$$

$$\mathcal{R}_\mathrm{S} = \int_{z^t}^{z^{t+1}} \sum_{i=0}^{1}\int_{\partial\Omega_i(z)} \hat{\phi}_k\rho\,(\boldsymbol{u}-\boldsymbol{v})\cdot\hat{\boldsymbol{N}}\,\mathrm{d}\sigma\,\mathrm{d}z. \quad (5.40c)$$

Both $\boldsymbol{u}$ and $\boldsymbol{v}$ have only one non-zero component because the refractive index field is piecewise constant and because of (5.37b). Hence, we can write $\boldsymbol{u} = (u_0, 0)$ and $\boldsymbol{v} = (v_0, 0)$. The surface term $\mathcal{R}_\mathrm{S}$ can, therefore, be written as

$$\begin{aligned}\mathcal{R}_\mathrm{S} &= \int_{z^t}^{z^{t+1}} \left[\int_{P_0}^{P_1} \hat{\phi}_k\rho(u_0 - v_0)\mathrm{d}p\Big|_{q=Q(z)^+}^{Q_1} + \int_{P_0}^{P_1} \hat{\phi}_k\rho(u_0 - v_0)\mathrm{d}p\Big|_{q=Q_0}^{Q(z)^-}\right]\mathrm{d}z \\ &= \int_{z^t}^{z^{t+1}} \left[\int_{P_0}^{P_1} \hat{\phi}_k\rho u_0\mathrm{d}p\Big|_{q=Q_0}^{Q_1} + \int_{P_0}^{P_1} \hat{\phi}_k\rho\left(u_0 - \frac{\mathrm{d}Q}{\mathrm{d}z}\right)\mathrm{d}p\Big|_{q=Q(z)^+}^{Q(z)^-}\right]\mathrm{d}z, \end{aligned} \quad (5.41)$$

where $Q(z)^\pm$ denote one-sided limits towards $Q(z)$. The simplifications also allow us to write the volume term as

$$\mathcal{R}_\mathrm{V} = \int_{z^t}^{z^{t+1}} \left[\int_{P_0}^{P_1}\int_{Q_0}^{Q(z)} \frac{\partial\hat{\phi}_k}{\partial q}\rho u_0\,\mathrm{d}q\mathrm{d}p + \int_{P_0}^{P_1}\int_{Q(z)}^{Q_1} \frac{\partial\hat{\phi}_k}{\partial q}\rho u_0\,\mathrm{d}q\mathrm{d}p\right]\mathrm{d}z. \quad (5.42)$$

An alternative formulation for the expressions (5.41) and (5.42) is desired, since $\left|\frac{\mathrm{d}Q}{\mathrm{d}z}\right| \to \infty$ for the optical interface shown in Figure 5.1. Therefore, we make use of the (local) inverse of the monotonic function $Q(z)$ on the interval $[z^t, z^{t+1}]$, which we denote by $Z(q)$, i.e., $z = Q^{-1}(q) = Z(q)$. Applying a change of variables to the second integral in (5.41) yields

$$\mathcal{R}_\mathrm{S} = \int_{z^t}^{z^{t+1}} \int_{P_0}^{P_1} \hat{\phi}_k\rho u_0\mathrm{d}p\Big|_{q=Q_0}^{Q_1}\mathrm{d}z + \int_{Q_0}^{Q_1}\int_{P_0}^{P_1} \hat{\phi}_k\rho\left(u_0\frac{\mathrm{d}Z}{\mathrm{d}q} - 1\right)\mathrm{d}p\Big|_{z=Z(q)^-}^{Z(q)^+}\mathrm{d}q, \quad (5.43)$$

where $Z(q)^{\pm}$ denote one-sided limits towards $Z(q)$. One can easily verify that (5.43) holds when $Q(z)$ is monotonically increasing and monotonically decreasing. In the volume term (5.42) we change the order of the integrals so that we obtain

$$\mathcal{R}_{\mathrm{V}} = \int_{Q_0}^{Q_1} \left[\int_{P_0}^{P_1} \int_{z^t}^{Z(q)} \frac{\partial \hat{\phi}_k}{\partial q} \rho u_0 \,\mathrm{d}z\mathrm{d}p + \int_{P_0}^{P_1} \int_{Z(q)}^{z^{t+1}} \frac{\partial \hat{\phi}_k}{\partial q} \rho u_0 \,\mathrm{d}z\mathrm{d}p \right] \mathrm{d}q. \tag{5.44}$$

We refer to expressions (5.41)-(5.42) as the $Q(z)$-formulation and to expressions (5.43)-(5.44) as the $Z(q)$-formulation. Either formulation has its benefit, depending on the shape of the optical interface. Note that in the surface term $\mathcal{R}_{\mathrm{S}}$ the fluxes ρu_0 are replaced with an upwind numerical flux, equivalent to (5.12), except at the optical interface. At the optical interface we compute the numerical fluxes using the methods that we will present in Section 5.5. In the results section we will describe which formulation we choose.

Making yet again use of the expansion of ρ described by (5.14) and inserting it into the left-hand side of equation (5.40a), subsequently transforming Ω to the static reference domain χ via (5.35) and applying the orthogonality of the basis functions (5.16) leads to

$$\begin{aligned} \int_{\Omega} \rho^{t+1} \hat{\phi}_k \,\mathrm{d}\mathcal{U} - \int_{\Omega} \rho^{t} \hat{\phi}_k \,\mathrm{d}\mathcal{U} &= \sum_{l=1}^{N_d} \left(\int_{\chi} \phi_l \phi_k \,\mathrm{d}\xi \right) \mathcal{J} \left(\rho_l^{t+1} - \rho_l^{t} \right), \\ &= \mathcal{J} W_k \left(\rho_k^{t+1} - \rho_k^{t} \right) \end{aligned} \tag{5.45}$$

where $\mathcal{J} = \Delta q \Delta p$ is the Jacobian determinant of the transformation (5.35), which is a constant for this type of element. The final formulation then reads

$$\mathcal{J} W_k \left(\rho_k^{t+1} - \rho_k^{t} \right) = \mathcal{R}_{\mathrm{V}} - \mathcal{R}_{\mathrm{S}}, \tag{5.46}$$

where we either use the $Q(z)$- or the $Z(q)$-formulation for $\mathcal{R}_{\mathrm{V}}$ and $\mathcal{R}_{\mathrm{S}}$. All the integrals that appear in (5.46) are approximated by $(N+1)$-point Gauss-Legendre quadrature. To complete the scheme we need to describe how we determine ρ at intermediate levels of $[z^t, z^{t+1}]$.

The optical interface cutting the cell in two pieces makes it rather complicated to construct a Taylor expansion, since at the optical interface we have to apply the jump condition (2.42). For this reason, we use that ρ remains invariant along a light ray, i.e., we apply (2.38) combined with the jump condition (2.42). At $z = z^t$ we know the solution, hence, if we trace a characteristic from any final point $(z_{\mathrm{f}}, \boldsymbol{x}_{\mathrm{f}})$ back to $z = z^t$ then we can determine its value at the final point. In other words, we apply the method of characteristics to determine ρ. The characteristic that ends at final point $(z_{\mathrm{f}}, \boldsymbol{x}_{\mathrm{f}})$ can be traced

back by solving Hamilton's equations (2.33), which can be concisely written as

$$\frac{\mathrm{d}\boldsymbol{x}}{\mathrm{d}z} = \boldsymbol{u}(z, \boldsymbol{x}), \tag{5.47a}$$

with final condition

$$\boldsymbol{x}(z_\mathrm{f}) = \boldsymbol{x}_\mathrm{f}, \tag{5.47b}$$

with the velocity field $\boldsymbol{u}$ given by (2.40b). At an optical interface we have to apply the law of specular reflection or Snell's law of refraction. Solving Hamilton's equations and applying the laws of optics at optical interfaces is known as ray tracing [18], where in this particular case it is local ray tracing. Note that with a piecewise constant refractive index field the light rays are piecewise straight lines in the (q, z)-plane. Therefore, solving (5.47) reduces to computing the intersections of light rays with optical interfaces and the plane $z = z^t$.

In case the light ray strikes the plane $z = z^t$ at phase space coordinates $\boldsymbol{x}^t$, we need to determine the value of ρ. After computing $\boldsymbol{x}^t$ we perform a search over the elements of the mesh to find from which element the characteristic originated. Once the correct element has been identified, the point $\boldsymbol{x}^t$ can be transformed to reference coordinates $\boldsymbol{\xi}$ on that particular element such that we can compute the value using expression (5.14). This process of determining the value at a new level and tracing back the characteristic, is also known as a semi-Lagrangian step. The semi-Lagrangian step is the key component in semi-Lagrangian methods such as the semi-Lagrangian discontinuous Galerkin methods presented in [35, 75, 79]. We employ the semi-Lagrangian step at every quadrature node required to evaluate the integrals in $\mathcal{R}_\mathrm{V}$ and $\mathcal{R}_\mathrm{S}$, which then allows us to update ρ_h using equation (5.46).

5.5 Optical interfaces

At an optical interface the momentum of a light ray changes discontinuously according to the law of specular reflection or Snell's law of refraction. This leads combined with the invariance of the basic luminance to the jump condition (2.42). The jump condition describes how phase space is connected at an optical interface.

In terms of the mesh used in the DG method described thus far, an optical interface at a fixed point (z, q) is described by a collection of momentum intervals, on either side of the interface, describing the momentum domain. In the ALE-ADER-DG method we align the mesh with the optical interface

and, therefore, each momentum interval corresponds to a face of an element. In the sub-cell interface method the optical interface cuts an element in two parts, where at the optical interface each momentum interval can be interpreted as a face. On both sides of the optical interface, we know a piecewise polynomial solution ρ at $z = z^t$ and we can compute, with either the Taylor expansions (5.28) or (5.32), or the semi-Lagrangian step, values on the interval $[z^t, z^{t+1}]$.

Since the piecewise polynomial solution ρ is in general a discontinuous function of the momentum p, we have to be careful in how we compute the numerical flux. For example, it can happen that the flux leaving one face of the optical interface is determined by the fluxes striking multiple faces, whilst the total flux should remain the same by conservation of energy. In other words, the laws of optics cause the elements to be connected in a highly non-trivial way at the optical interface, moreover, a change in the normal of the optical interface causes a change in the connectivity of the elements.

In Section 4.3 we presented a method that incorporates the jump condition (2.42) in an energy-conserving manner into a DG spectral element method, for a fixed optical interface described by $q = \text{const}$. In this section, we will summarise the key components of the method and the extension to arbitrary curved optical interfaces. In particular, for arbitrary curved optical interfaces we have to separate the contributions at an optical interface into forward- and backward-propagating light. To that end, we need to transform a one-dimensional momentum interval to the Descartes' sphere (circle in 2D), compute the corresponding incident light and subsequently split it into forward-propagating and backward-propagating contributions. This process is sketched in Figure 5.3 and the formal procedure is elaborated in the following section.

5.5.1 Partitioning of momentum intervals

To facilitate the usage of the vectorial laws of reflection and refraction described by (2.43), we first introduce some notation to transfer from the full momentum vector (p, p_z) to p and back. We define $C_\sigma(n) = \{(p, p_z) \in \mathrm{S}^1(n) \mid \operatorname{sgn} p_z = \sigma\}$ with sgn the sign function defined as $\operatorname{sgn}(x) = 1$ if $x \geq 0$ and $\operatorname{sgn}(x) = -1$ if $x < 0$, so that $C_\mathrm{f}(n) \cup C_\mathrm{b}(n) = \mathrm{S}^1(n)$. Recall again that we write subscript f for $\sigma = 1$ and b for $\sigma = -1$. Given the momentum vector $(p, p_z) \in C_\sigma(n_0)$ we compute the momentum p with the mapping $P_\sigma : C_\sigma(n_0) \to [-n_0, n_0]$ as $(p, p_z) \mapsto p$. Furthermore, given a momentum p we compute its full momentum vector with the (inverse) mapping $P_\sigma^{-1} : [-n_0, n_0] \to C_\sigma(n_0)$ as $p \mapsto (p, \sigma\sqrt{n_0^2 - p^2})$.

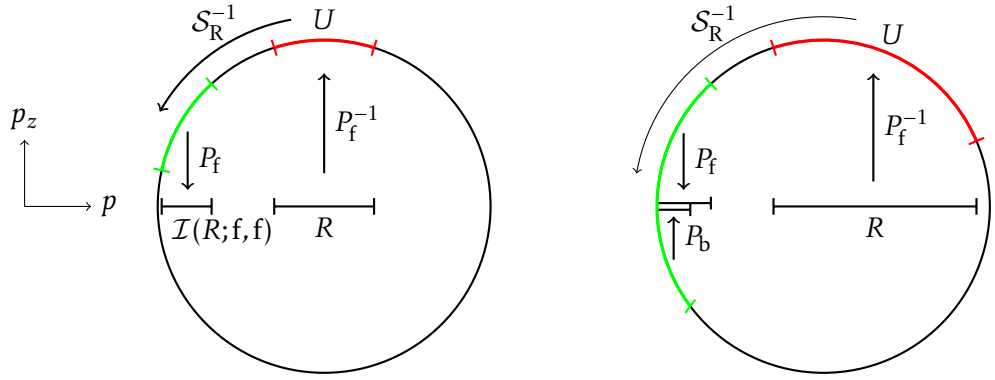


Figure 5.3: Two examples where we apply the mappings $P_{\sigma-}$, $P_{\sigma+}^{-1}$, and apply $\mathcal{S}_R^{-1}$ on the sphere $\mathrm{S}^1(n_0)$. On the left the incident light only has $\sigma = 1$, whereas on the right both forward and backward incident light contribute.

We aim towards defining a formal procedure for finding the incident light, given that we know the momenta values after reflection or refraction. For ease of presentation we will only consider reflection. Let R now be some momentum interval $R = [p_0, p_1] \subset [-n_0, n_0]$ describing light after reflection. Applying now P_σ^{-1} to R yields

$$P_\sigma^{-1}(R) = \left\{ (p, p_z) \in C_\sigma(n_0) \mid p \in R, p_z = \sigma\sqrt{n_0^2 - p^2} \right\}, \tag{5.48a}$$

and similarly for $U = P_\sigma^{-1}(R)$ we can transfer back down to R with the mapping P_σ, i.e.,

$$P_\sigma(U) = \{p \mid (p, p_z) \in U\}. \tag{5.48b}$$

For a physical interpretation of P_σ and P_σ^{-1} see Figure 5.3. Let now $U = P_\sigma^{-1}(R)$, so that $U \subset C_\sigma(n_0)$ contains the momentum vectors of the reflected light. Then the momentum vectors of the incident light can be found by applying $\mathcal{S}_{\mathrm{R}}^{-1}$ to $U = P_\sigma^{-1}(R)$, i.e.,

$$\mathcal{S}_{\mathrm{R}}^{-1}(P_\sigma^{-1}(R)) = \left\{ \mathcal{S}_{\mathrm{R}}^{-1}\big((p, p_z)\big) \,\middle|\, (p, p_z) \in P_\sigma^{-1}(R) \right\}. \tag{5.49}$$

We can restrict $\mathcal{S}_{\mathrm{R}}^{-1}(P_\sigma^{-1}(R))$ to $C_\sigma(n_0)$ by computing its intersection with either $C_{\mathrm{f}}(n_0)$ or $C_{\mathrm{b}}(n_0)$. Subsequently applying P_σ yields the momentum values on $[-n_0, n_0]$ corresponding to the incident light. This process is illustrated in Figure 5.3.

Finally, the actions are combined such that we can formally find the forward ($\sigma = 1$) or backward ($\sigma = -1$) incident light by applying $\mathcal{I}$ to R, where

$\mathcal{I}(R;\sigma^-,\sigma^+)$ is defined as

$$\mathcal{I}(R;\sigma^-,\sigma^+) = P_{\sigma-}\left(\mathcal{S}_{\mathrm{R}}^{-1}(P_{\sigma+}^{-1}(R)) \cap C_{\sigma-}(n_0)\right), \tag{5.50}$$

here the − and + are used to distinguish incident and reflected light, respectively. The result of (5.50) is shown in Figure 5.3. Similar to (5.50) we define the operation also for transmission with $\mathcal{S}_{\mathrm{R}}^{-1}$ replaced by $\mathcal{S}_{\mathrm{T}}^{-1}$ with appropriate changes to the refractive indices.

With the formal definition of the incident light, we can relate the total flux for incident and outgoing light at the optical interface for the interval R with $\sigma^+ = 1$. The total flux leaving R is equal to the flux striking the intervals $\mathcal{I}(R;\mathrm{b},\mathrm{f})$ and $\mathcal{I}(R;\mathrm{f},\mathrm{f})$, which correspond to the intervals of incident light with $\sigma^- = -1$ and $\sigma^- = 1$, respectively. This is expressed in the following energy balance

$$\int_R \rho_{\mathrm{f}}\left(u_0 - \frac{\mathrm{d}Q}{\mathrm{d}z}\right)\Big|_+ \mathrm{d}p = \int_{\mathcal{I}(R;\mathrm{b},\mathrm{f})} \rho_{\mathrm{b}}\left(u_0 - \frac{\mathrm{d}Q}{\mathrm{d}z}\right)\Big|_- \mathrm{d}p + \int_{\mathcal{I}(R;\mathrm{f},\mathrm{f})} \rho_{\mathrm{f}}\left(u_0 - \frac{\mathrm{d}Q}{\mathrm{d}z}\right)\Big|_- \mathrm{d}p, \tag{5.51}$$

where $\cdot|_\pm$ denotes, once more, one-sided limits towards the optical interface. A proof for the energy balance (5.51) is given in Appendix C. In general, forward- and backward-propagating light can contribute to an interval R. We separate the contributions by partitioning the interval as $R = R_0 \cup R_1$ so that $\mathcal{I}(R_0;\mathrm{b},\mathrm{f}) = \emptyset$ and $\mathcal{I}(R_1;\mathrm{f},\mathrm{f}) = \emptyset$. Hence with the partitioning of R the energy balance (5.51) leads to the following two balances

$$\int_{R_0} \rho_{\mathrm{f}}\left(u_0 - \frac{\mathrm{d}Q}{\mathrm{d}z}\right)\Big|_+ \mathrm{d}p = \int_{\mathcal{I}(R_0;\mathrm{f},\mathrm{f})} \rho_{\mathrm{f}}\left(u_0 - \frac{\mathrm{d}Q}{\mathrm{d}z}\right)\Big|_- \mathrm{d}p, \tag{5.52a}$$

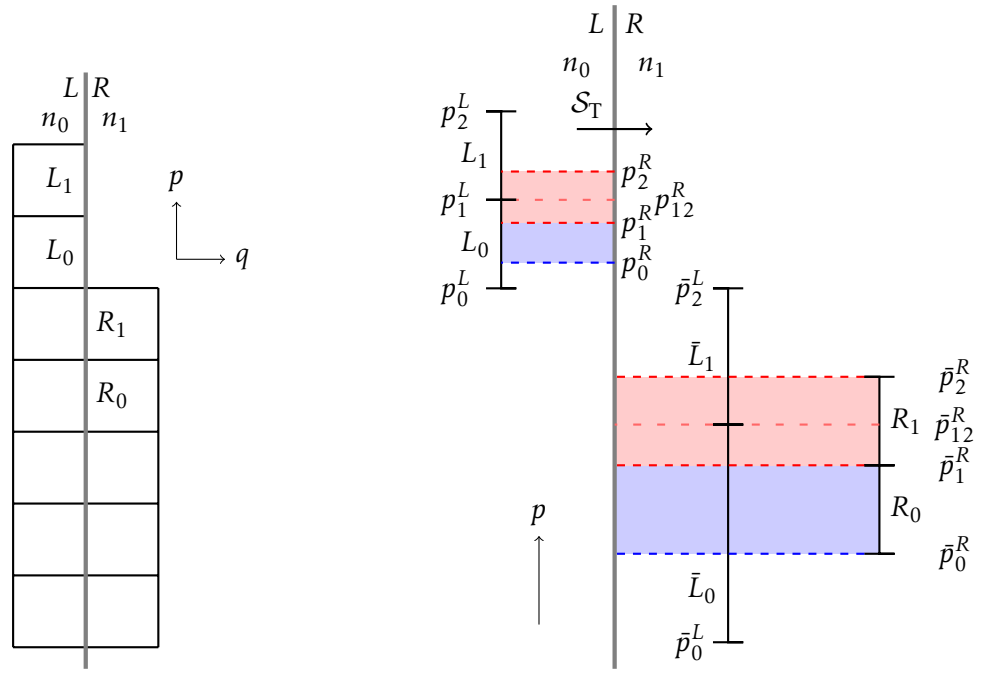
$$\int_{R_1} \rho_{\mathrm{f}}\left(u_0 - \frac{\mathrm{d}Q}{\mathrm{d}z}\right)\Big|_+ \mathrm{d}p = \int_{\mathcal{I}(R_1;\mathrm{b},\mathrm{f})} \rho_{\mathrm{b}}\left(u_0 - \frac{\mathrm{d}Q}{\mathrm{d}z}\right)\Big|_- \mathrm{d}p. \tag{5.52b}$$

We remark that for transmission only $\mathcal{S}_{\mathrm{R}}^{-1}$ needs to be replaced with $\mathcal{S}_{\mathrm{T}}^{-1}$ in (5.50).

Recall, that we only consider forward-propagating light, i.e., we solve Liouville's equation (5.1) where $\sigma = 1$. This means that in general we do not know ρ_{b}. Depending on the optical system and initial/boundary conditions it is not necessary to solve for backward-propagating light. In particular, for the examples presented in Section 5.7, backward-propagating light does not play a role. Hence, we simply take $\rho_{\mathrm{b}} = 0$ in (5.52).

5.5.2 Energy-conserving fluxes

Consider now an optical interface $q = Q(z)$ with slope $\frac{\mathrm{d}Q}{\mathrm{d}z}$ separating the media with refractive indices n_0 and n_1. An example of a geometry of the elements



(a) Geometry in phase space.

(b) Connectivity of faces.

Figure 5.4: Sketch of the geometry at an optical interface.

is sketched in Figure 5.4a. The figure shows a number of faces on both sides of the optical interface. Let L_0 and L_1 be the faces where light strikes the interface, and R_0 and R_1 the faces where light leaves the interface, in other words, at L_0 and L_1 the velocity field is directed towards the optical interface while at R_0 and R_1 the velocity field is directed away from the optical interface.

Let us now consider a face R_i $(i = 0, 1)$ and for sake of simplicity assume that all its corresponding incident light is forward propagating; see Figure 5.4b. With this assumption we have $\sigma^- = 1$ and $\sigma^+ = 1$, and the energy balance for the face reads

$$\int_{R_i} \rho \left(u_0 - \frac{\mathrm{d}Q}{\mathrm{d}z} \right) \Big|_+ \mathrm{d}p = \int_{\mathcal{I}(R_i;\mathrm{f},\mathrm{f})} \rho \left(u_0 - \frac{\mathrm{d}Q}{\mathrm{d}z} \right) \Big|_- \mathrm{d}p, \tag{5.53}$$

where we omit the σ subscript for ρ. The energy balance (5.53) will be important in ensuring energy conservation in the ALE-ADER-DG scheme. The energy balance for an interface described by $z = Z(q)$, which is a (local) inverse of $q = Q(z)$, can be found by multiplying the balance (5.53) with $\frac{\mathrm{d}Z}{\mathrm{d}q}$

and applying $\frac{\mathrm{d}Q}{\mathrm{d}z}\frac{\mathrm{d}Z}{\mathrm{d}q} = 1$, resulting in the energy balance

$$\int_{R_i} \rho \left(u_0 \frac{\mathrm{d}Z}{\mathrm{d}q} - 1 \right)\Big|_{+} \mathrm{d}p = \int_{\mathcal{I}(R_i;\mathrm{f},\mathrm{f})} \rho \left(u_0 \frac{\mathrm{d}Z}{\mathrm{d}q} - 1 \right)\Big|_{-} \mathrm{d}p. \tag{5.54}$$

Due to the partitioning of the momentum intervals described in the previous section we know the value of σ for the incident and the reflected/refracted light and, therefore, introduce the shorthand notation $S(p) = S(p; n_0, n_1, \vec{\nu})$. Here $S(p)$ simply takes the first component of $\mathcal{S}$ in (2.43). In particular, in the case considered here, we have $\sigma^- = 1$ and $\sigma^+ = 1$, so that the function $S(p)$ reads

$$S(p) = \begin{cases} S_{\mathrm{R}} = p - 2\psi\nu_q & \text{if } \delta \leq 0, \\ S_{\mathrm{T}} = p - (\psi + \sqrt{\delta})\nu_q & \text{if } \delta > 0, \end{cases} \tag{5.55a}$$

with

$$\psi = \left(\frac{p}{\sqrt{n_0^2 - p^2}} \right) \cdot \begin{pmatrix} \nu_q \\ \nu_z \end{pmatrix} \text{ and } \delta = n_1^2 - n_0^2 + \psi^2. \tag{5.55b}$$

Moreover, we will use the notation S_{T} to describe the q-component of $\mathcal{S}_{\mathrm{T}}$ and similarly we will use S_{T}^{-1} to denote the q-component for refraction in reverse.

We proceed by determining for each face R_i the contributing faces. Therefore, we first transform the faces L_0 and L_1 to the other side of the optical interface by applying Snell's law of refraction $\mathcal{S}_{\mathrm{T}}$ resulting in the virtual faces $\bar{L}_0$ and $\bar{L}_1$ with $\bar{L}_i = S_{\mathrm{T}}(L_i)$. The virtual faces can now be related to the face R_i; see Figure 5.4b.

As mentioned before, we have to be careful in how we compute the numerical flux in order to ensure we obey the energy balance (5.53) discretely. At a fixed point (z, q) on the optical interface we can write the solution on each face as a polynomial of the momentum p. We denote the polynomial on a face L_i by $\rho^{L_i}(p) \in \mathbb{P}_N$. Application of the jump condition (2.42) allows us to relate ρ on the face L_i to its counterpart on the virtual face $\bar{L}_i$ by

$$\rho^{\bar{L}_i}(\bar{p}) = \rho^{L_i}\left(S_{\mathrm{T}}^{-1}(\bar{p}) \right) = \rho^{L_i}(p), \text{ with } \bar{p} = S_{\mathrm{T}}(p). \tag{5.56}$$

Combining relation (5.56) with the geometric connectivity of the faces from Figure 5.4a allows us to describe how we compute the polynomial $\rho^{R_i}(p) \in \mathbb{P}_N$ for each face R_i. For example, the polynomial on face R_1 depends on ρ at the faces L_0 and L_1. The polynomial $\rho^{R_1} \in \mathbb{P}_N$ must thus be computed from a piecewise polynomial ρ^L with the additional constraint of the energy

balance (5.53), therefore, we pose the problem as a constrained least-squares problem that reads

$$\min_{\rho^{R_1}\in\mathbb{P}_N} \int_{\bar{p}_1^R}^{\bar{p}_2^R} \left(\rho^{R_1}(\bar{p}) - \rho^L(S_{\mathrm{T}}^{-1}(\bar{p}))\right)^2 \mathrm{d}\bar{p}, \tag{5.57a}$$

$$\text{subject to } \int_{\bar{p}_1^R}^{\bar{p}_2^R} F^{R_1}(\bar{p})\,\mathrm{d}\bar{p} = \int_{p_1^R}^{p_2^R} F^L(p)\,\mathrm{d}p, \tag{5.57b}$$

where $p_1^R = S_{\mathrm{T}}^{-1}(p_1^R)$, etc., and ρ^L and F^L denote piecewise polynomials given by

$$\rho^L(p) = \begin{cases} \rho^{L_0}(p) & \text{if } p \in L_0, \\ \rho^{L_1}(p) & \text{if } p \in L_1 \end{cases} \text{ and } F^L(p) = \begin{cases} F^{L_0}(p) & \text{if } p \in L_0, \\ F^{L_1}(p) & \text{if } p \in L_1. \end{cases}$$

Here, the numerical flux $F^{L_i}(p)$ is written in a basis of Lagrange polynomials, i.e.,

$$F^{L_i}(p) = \sum_{j=0}^{N} \rho_j^{L_i} a_j \ell_j(\eta(p)) \text{ with } a_j = u_0 - \left.\frac{\mathrm{d}Q}{\mathrm{d}z}\right|_{(-,p_j)}, \tag{5.58}$$

where $\eta(p)$ denotes a transformation from the face L_i to the reference interval $[0,1]$, and $\{p_j\}_{j=0}^N$ denote the Gauss-Legendre quadrature points on the interval L_i. The numerical flux F^{R_1} is similarly written as in (5.58), where the coefficients $\rho_j^{R_1}$ are the expansion coefficients of polynomial ρ^{R_1} that are to be determined from (5.57).

The constrained least-squares problem (5.57) is now solved as described in Section 4.3. In short, we start by writing the problem in terms of a Lagrangian with a Lagrangian multiplier and subsequently impose the requirements for a stationary point and apply $(N+1)$-point Gauss-Legendre quadrature on each (part of a) face resulting in a linear system for the $N+1$ coefficients $\rho_j^{R_1}$ and a Lagrange multiplier. The linear system is solved analytically, see A, to obtain the $N+1$ coefficients $\rho_j^{R_1}$ on the face R_1. Finally, from these coefficients we can compute the numerical flux that is used in the ALE-ADER-DG method or the sub-cell interface method.

In a similar manner, the method can be applied when considering total internal reflection where in the equations (5.53), (5.56) and (5.57) refraction should be replaced by reflection, i.e., replacing S_{T} by S_{R}. Moreover, the method works for arbitrary configurations of faces at an optical interface.

In the implementation we require a point search algorithm to determine the connectivity between faces. In practice, this point search is efficiently implemented via a binary search. For more details, see Appendix D.

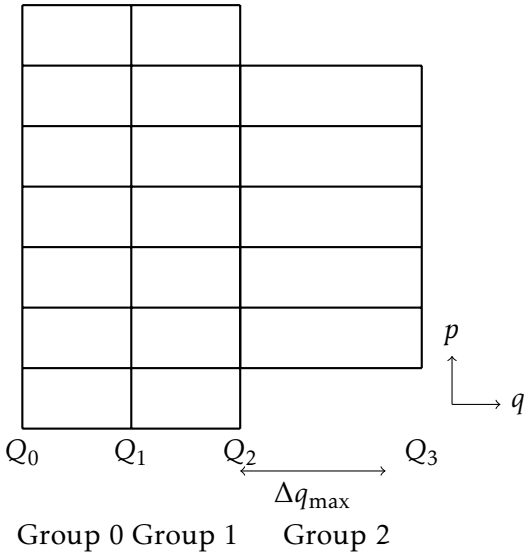


Figure 5.5: Sketch of a mesh. Group 2 is a candidate for mesh refinement and will be split into two smaller groups with the mesh refinement algorithm.

5.6 Mesh refinement

In the moving mesh method small or large elements can appear. Moreover, we might want to prepare the mesh such that we can apply the sub-cell interface method and still have control over the stepsize Δz. In the description of the ALE-ADER-DG scheme we have made use of Cartesian elements. Furthermore, the mesh is only allowed to move in the q-direction and the p-component of the velocity is zero, hence, for mesh refinement we only need to consider the geometry in the q-direction. Thus, we do not consider mesh refinement along the p-direction, and interpret the mesh as a collection of q-intervals with multiple Cartesian elements per q-interval. Then, at every q-interval we collect all the elements that share the same q-interval; see Figure 5.5. We refer to such a collection of elements as a group. The groups are sorted based on their q-values. In addition to simplifying the mesh refinement, this specific structure also simplifies the search for the correct element in the semi-Lagrangian step.

The mesh refinement algorithm requires a minimum mesh spacing $\Delta q_{\min}$ and a maximum mesh spacing $\Delta q_{\max} = \alpha \Delta q_{\min}$, with $\alpha > 1$. In the mesh refinement algorithm we loop over all the groups. If either of the current group and the next group have a mesh spacing Δq smaller than $\Delta q_{\min}$ and have a cumulative mesh spacing smaller than $\Delta q_{\max}$, then we combine these

two groups and as a result coarsen the mesh. Otherwise, if the current group's mesh spacing is bigger than $\Delta q_{\max}$, then we split the current group at its midpoint into two smaller groups. Since ρ is discontinuous across an optical interface, we only allow the coarsening of groups that share the same refractive index. We iterate the mesh refinement procedure until no more groups have been modified in an iteration.

The coarsening of the groups is performed by means of an L_2-projection. For example, if we consider two adjacent elements with known piecewise polynomial $w(\boldsymbol{x})$ defined on $\Omega = [Q_0, Q_1] \times [P_0, P_1]$, then we compute the polynomial

$$\rho(\boldsymbol{x}) = \sum_{l=1}^{N_d} \rho_l \hat{\phi}_l(\boldsymbol{x})$$

by solving

$$\int_\Omega (\rho(\boldsymbol{x}) - w(\boldsymbol{x}))\, \hat{\phi}_k(\boldsymbol{x})\, \mathrm{d}\mathcal{U} = 0 \quad \text{for } k = 1, 2, \ldots, N_d, \tag{5.59}$$

where the basis functions are once again defined by a transformation to the reference domain, such that it is equivalent to (5.14). By solving (5.59) we compute the polynomial ρ that minimises the L_2-norm of $\rho - w$. The refining, or splitting, of a group is also performed by means of an L_2-projection, which in this case is equivalent to interpolating the given polynomial. We remark that solving (5.59) exactly implies energy conservation, since the constant function 1 is contained in the span of the basis functions so that $\int_\Omega \rho \, \mathrm{d}\mathcal{U} = \int_\Omega w \, \mathrm{d}\mathcal{U}$.

5.7 Results

In the following we will discuss two examples, a meniscus lens and a dielectric total internal reflection concentrator. To solve Liouville's equation, we apply a few fixed settings for these problems. Namely, we take $M = N$ in the Taylor expansions (5.28) and (5.32), such that the ALE-ADER-DG scheme has a formal $(N+1)$th order accuracy in space and z. Moreover, we use the CFL condition (5.34) with CFL = 0.9 fixed and we take $\alpha = 2.25$ in the mesh refinement procedure, as described in Section 5.6. In the sub-cell interface method we choose the $Z(q)$-formulation given by equations (5.43)-(5.44).

The ALE-ADER-DG scheme was implemented in C++. Computing derivatives of a polynomial and interpolating a polynomial can all be re-arranged into a small matrix multiplication. These operations are performed by using the optimised *libxsmm* library [48, 49] for small matrix multiplications on Intel machines. For more implementation details, see Appendix D. All the

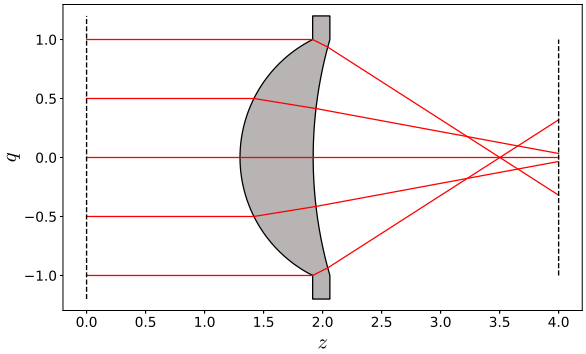


Figure 5.6: Meniscus lens with a couple of light rays. The refractive index of the lens is $n_1 = 1.5$ and for the background medium $n_0 = 1$.

simulations were performed using a single core of a laptop that has an Intel Core i7–7700HQ CPU @ 2.80GHz.

5.7.1 Meniscus lens

As a first example we consider the meniscus lens, that features two spherically curved surfaces. The geometry of the meniscus lens for three-dimensional optics is rotationally symmetric, so that for two-dimensional optics we will take a cross section of the spherical surfaces which leads to circles. The geometry that we consider is shown in Figure 5.6. The meniscus lens features two circle segments, which satisfy

$$q^2 + (z - z_c)^2 = R^2. \tag{5.60}$$

For the left circle we take $z_c = 2.42$ and $R = 1.12$, whereas for the right circle we take $z_c = 5.52$ and $R = 3.6$. For this example, the q-domain is given by the interval $[-1.2, 1.2]$ for $z \leq z_2 = 5.52 - \sqrt{3.6^2 - 1} \approx 2.06$ and $[-1, 1]$ for $z > z_2$. Here, $z = z_2$ is the plane that intersects the right circle at $q = \pm 1$. One can imagine the meniscus lens being fixed onto some physical system such that at $z = z_2$ the light striking at $q < -1$ and $q > 1$ is fully absorbed. To numerically solve for the meniscus lens we apply the sub-cell interface method only for one single step at $z = z_c - R$ for each circle, and for the remaining curved part of the lens we apply the moving mesh method to align the mesh with the optical interface. The remaining parts of the system do not require a moving mesh, hence, we simply use a static mesh in those regions. In the moving mesh method we prescribe the mesh velocity at the optical interface by writing (5.60) as $q = Q(z)$, such that the mesh velocity is given by $\frac{\mathrm{d}Q}{\mathrm{d}z}$ at the

interface. In the sub-cell interface method the intersections of a light ray with the surface described by equation (5.60) are computed analytically.

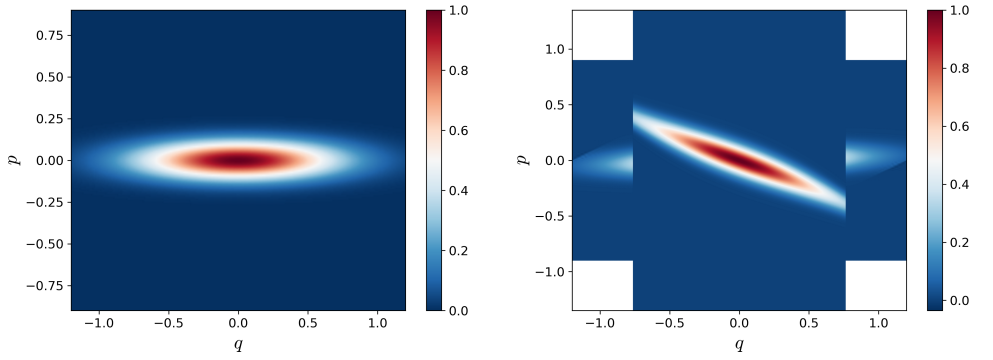
To show the effects of the lens we compute a numerical solution. At $z = 0$ we start with a Gaussian distribution, given by

$$\rho_0(q,p) = \exp\left(-\frac{q^2}{2\sigma_q^2}\right)\exp\left(-\frac{p^2}{2\sigma_p^2}\right), \tag{5.61}$$

where we take $\sigma_q = 0.5$ and $\sigma_p = 0.08$. For this particular problem we limit the maximum momentum, since the velocity $\boldsymbol{u}$ (2.40b) blows up as $|p|$ approaches n, therefore, we limit the maximum momentum to $0.9n(z,q)$. Furthermore, we choose mesh spacings $\Delta q_{\max} = 0.2$ and $\Delta p = 0.09$, and use $N = 7$. Recall that $\Delta q_{\min} = \Delta q_{\max}/\alpha$. Initially the mesh has 520 elements and at the end at $z = 4$ the mesh contains 360 elements. The initial condition and the numerical solution at various z-levels are shown in Figure 5.7. From last panel in the figure, we observe that the initial condition has been compressed in the q-direction and expanded in the p-direction. Moreover, one can see values below 0 on the target distribution at $z = 4$ which is due to a cut-off of the initial distribution. The cut-off generates a discontinuity in the distribution, which appears as an oscillation resulting in undershoot in the numerical solution.

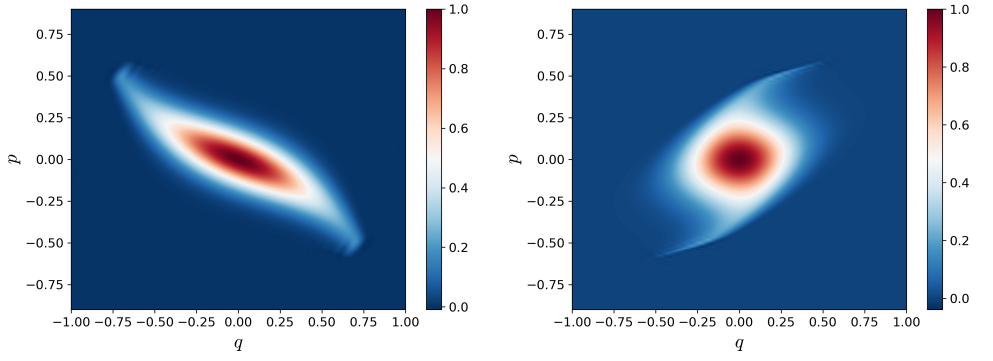
The optical interface discretisation as described in Section 5.5 should be energy-conserving for this example. Therefore, the luminous flux inside the domain plus the luminous flux leaving the domain through the physical boundaries of the system (excluding optical interfaces) should remain constant. The former is computed by integrating ρ over the phase space domain, whereas the latter is computed by adding the numerical fluxes that leave the system. We compute the absolute relative deviation from energy conservation at every step and find that the maximum deviation from energy conservation is $2.44 \cdot 10^{-15}$ and, thus, we observe energy conservation up to machine precision.

Next, we compare two strategies to apply the moving mesh method. As explained in Section 5.3, the update of a moving element is more expensive than that of a static element. Since we only require the moving mesh method to align optical interfaces with the mesh, we can use the freedom in the mesh velocity to optimise for better performance by using fewer moving elements. In the first strategy, which we will refer to as the global strategy, we let the mesh velocity at a certain z-position be a piecewise linear interpolant between points where we prescribe the mesh velocity. These points are the boundary of the domain where the mesh velocity is 0, and the optical interfaces where the mesh velocity is computed according to the shape of the optical interface. In the second strategy, which we call the local strategy, we only move elements adjacent to an optical interface, while the other elements remain fixed. An



(a) $z = 0$.

(b) $z = 1.6$.



(c) $z = 2.4$.

(d) $z = 4$.

Figure 5.7: Distributions of ρ for the meniscus lens with Gaussian initial condition computed with the $N = 7$ ALE-ADER-DG scheme.

example of the mesh velocity for both strategies is shown in Figure 5.8.

To compare both strategies we measure the computation time in the moving mesh region of the example, which is the region between the two z-planes $z = z_1 = 1.3$, with $(z, q) = (z_1, 0)$ the left-most point on the left surface, and $z = z_2$. For the global strategy we denote the computation time for this region with t_{global} and for the local strategy we denote the computation time with t_{local}. The speed up $t_{\text{global}}/t_{\text{local}}$ is plotted in Figure 5.9 as a function of the polynomial degree N for various refinement levels r. A refinement level r indicates

$$\Delta q_{r,\max} = 2^{-r} \Delta q_{0,\max} \text{ and } \Delta p_r = 2^{-r} \Delta p_0, \tag{5.62}$$

where we choose $\Delta q_{0,\max} = 0.4$ and $\Delta p_0 = 0.2$ for $r = 0$. From Figure 5.9 we observe that only moving the mesh elements close to the optical interfaces is significantly more efficient, especially for larger N values. Therefore, in the

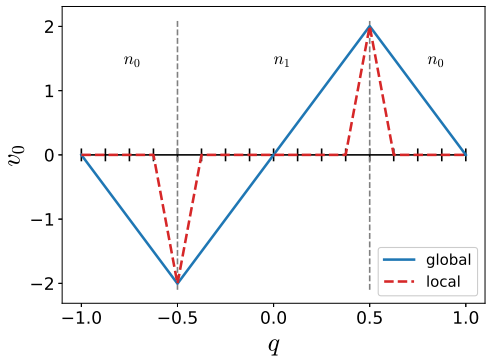


Figure 5.8: Global versus local strategy of defining the mesh velocity v_0 for the meniscus lens at some z. Black intervals denote the q-intervals of elements and gray dashed lines denote optical interfaces.

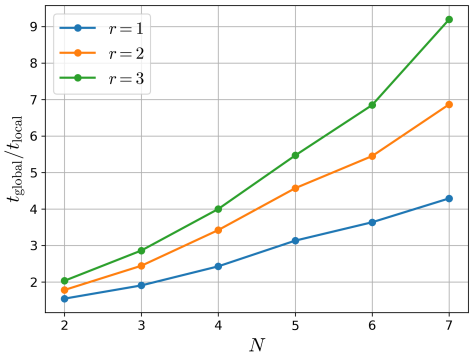


Figure 5.9: Speed up $t_{\text{global}}/t_{\text{local}}$ of the moving mesh portion of the meniscus lens.

rest of the chapter we will only use the local strategy where only elements next to the optical interface are moving.

Next, we study the convergence of the scheme. To that end, we use an initial condition so that the solution at $z = 4$ is sufficiently smooth. In particular, we take

$$\rho_0(q,p) = \varphi_{m,k}\left(\frac{q}{\lambda_q}\right)\varphi_{m,k}\left(\frac{p}{\lambda_p}\right), \tag{5.63}$$

with parameters $\lambda_q = 0.5$ and $\lambda_p = 0.25$. For the function $\varphi_{m,k}$, given by (4.66), we choose $m = 10$ and $k = 2$ and a plot of the function is shown in Figure 5.10. With the chosen initial condition, the exact solution at $z = 4$ can be obtained by tracing light rays backwards through the circle segments of the lens, i.e., we apply the method of characteristics. The convergence results for the L_2 and L_∞ norms are listed in Table 5.1, where the convergence rate is measured

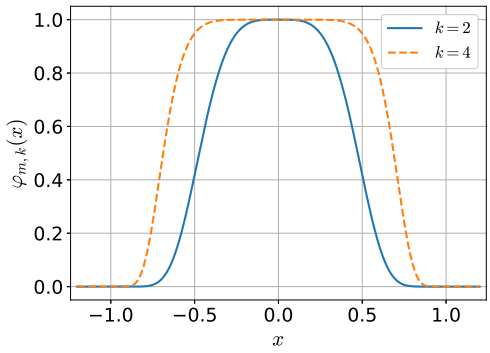


Figure 5.10: Function $\varphi_{m,k}$ for $k = 2$ and $k = 4$ with $m = 10$.

as $\log_2(e_{r-1}/e_r)$ with e_r the error for refinement level r. The computed orders of convergence are in good agreement with the expected $N+1$ order of convergence.

As a final study for this example, we compare solving Liouville's equation with the ALE-ADER-DG method to quasi-Monte Carlo ray tracing. The illuminance is computed using both methods. For quasi-Monte Carlo ray tracing we fix the number of bins B and employ a uniform grid on the target interval $q \in [-1,1]$ at $z = 4$. For more details see Section 4.4.2. For this particular example, ray tracing can compute exact intersections for each part of the meniscus lens, avoiding the need for a root-finder.

To compare the performance of both methods, we want to compute the error as the L_∞-norm of the illuminance. The quasi-Monte Carlo method computes an average illuminance on each bin, therefore, we also compute the average illuminance for the ALE-ADER-DG scheme when computing the error. Once again, we take the initial condition (5.63) such that we can use the exact solution to Liouville's equation to compute the exact illuminance.

The comparison of both methods is plotted in Figure 5.11, where the error is plotted as a function of the computation time. For quasi-Monte Carlo ray tracing we choose $B = 200$ and vary the number of rays used. To be specific, for the first data point we use $N_{\mathrm{RT}} = 31250$ rays and quadruple the number of rays for each subsequent point, such that at the last point we are using $N_{\mathrm{RT}} = 2.048 \cdot 10^9$ rays. For the ALE-ADER-DG method we choose finer mesh spacings for subsequent points, see (5.62).

From Figure 5.11 it can be seen that for a 10 second computation time the ALE-ADER-DG scheme with $N = 3$ achieves roughly 1 order of magnitude lower error compared to quasi-Monte Carlo ray tracing, and for $N = 5$ and $N = 7$ the difference in error has increased to roughly 2 orders of magnitude.

Table 5.1: Convergence data for the meniscus lens example with the ALE-ADER-DG scheme.

r	L_2	$\mathcal{O}(L_2)$	L_∞	$\mathcal{O}(L_\infty)$
		$N = 2$		
0	6.41e-02		3.69e-01	
1	1.22e-02	2.40	1.10e-01	1.74
2	1.87e-03	2.70	2.40e-02	2.20
3	2.46e-04	2.93	3.83e-03	2.64
4	3.13e-05	2.98	5.17e-04	2.89
		$N = 3$		
0	3.86e-02		2.80e-01	
1	4.41e-03	3.13	4.93e-02	2.51
2	3.60e-04	3.61	6.24e-03	2.98
3	2.44e-05	3.88	4.40e-04	3.83
4	1.56e-06	3.97	2.91e-05	3.92
		$N = 4$		
0	2.42e-02		1.89e-01	
1	1.48e-03	4.03	1.89e-02	3.32
2	6.77e-05	4.45	1.25e-03	3.92
3	2.37e-06	4.84	4.66e-05	4.75
4	7.60e-08	4.96	1.58e-06	4.88
		$N = 5$		
0	1.47e-02		1.45e-01	
1	5.08e-04	4.85	9.70e-03	3.90
2	1.25e-05	5.34	2.79e-04	5.12
3	2.27e-07	5.78	5.45e-06	5.68
4	3.70e-09	5.94	9.26e-08	5.88
		$N = 6$		
0	8.81e-03		9.49e-02	
1	1.76e-04	5.64	3.61e-03	4.72
2	2.33e-06	6.24	5.91e-05	5.93
3	2.17e-08	6.74	6.14e-07	6.59
4	1.80e-10	6.92	5.08e-09	6.92
		$N = 7$		
0	5.35e-03		6.07e-02	
1	6.06e-05	6.47	1.38e-03	5.46
2	4.36e-07	7.12	1.28e-05	6.76
3	2.10e-09	7.70	6.28e-08	7.67
4	8.98e-12	7.87	2.73e-10	7.85

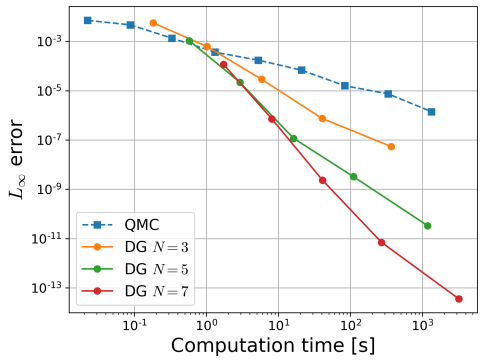


Figure 5.11: Comparison between quasi-Monte Carlo (QMC) ray tracing and ALE-ADER-DG scheme (DG) for the meniscus lens.

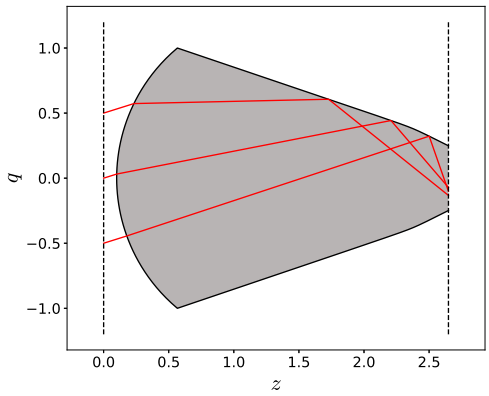


Figure 5.12: A DTIRC and a couple of light rays. The gray colour represents a refractive index $n_1 = 1.5$ and the white colour the background medium with $n_0 = 1$.

Moreover, the ALE-ADER-DG scheme converges much faster than the quasi-Monte Carlo ray tracing method, in other words the ALE-ADER-DG scheme is more efficient to compute high accuracy solutions.

5.7.2 Dielectric TIR concentrator

As a second example we consider the dielectric TIR concentrator (DTIRC). The geometry that we consider is shown in Figure 5.12. The optical system concentrates light that is emitted within a certain acceptance angle, from $z = 0$ towards the target in the (dielectric) medium with $n_1 = 1.5$. The rays shown in Figure 5.12 are first refracted at a circularly shaped surface, followed by reflection at one of the side walls. These side walls are designed such that the

Table 5.2: Parameters for the DTIRC.

Parameter	Value
z_c	1.405407
R	1.305407
a_1	-0.423579
b_0	-0.194090
b_1	-0.875464
b_2	0.191880
Z_{target}	2.648668

light rays satisfy the condition for total internal reflection. Details about the design process of such a system can be found in [18, 69]. The circularly shaped surface of the device is given by (5.60), whereas the top side wall satisfies $q = Q_{\text{top}}(z)$ with $q > 0$ and the bottom side wall is given by $q = -Q_{\text{top}}(z)$. Here $Q_{\text{top}}(z)$ reads

$$Q_{\text{top}}(z) = a_0 + a_1 z + b_0 \sqrt{1 + b_1 z + b_2 z^2}, \tag{5.64}$$

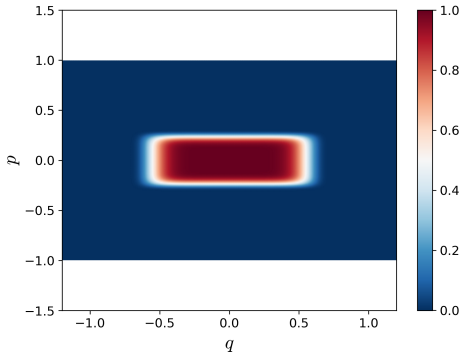
and the target is placed at $z = Z_{\text{target}}$. The parameters for the DTIRC are listed in Table 5.2. The parameter a_0 is fixed by requiring that the circle segment connects to the top side wall at $q = 1$ and $z = Z_1 = z_c - \sqrt{R^2 - 1} \approx 0.566308$, yielding the value $a_0 = 1.3519991422999297$.

As initial condition we use

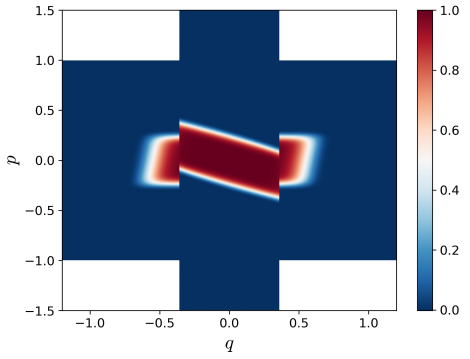
$$\rho_0(q, p) = \varphi_{m,k}\left(\frac{q}{\lambda_q}\right)\varphi_{m,k}\left(\frac{p}{\lambda_p}\right), \tag{5.65}$$

with $\varphi_{m,k}$ defined in (4.66) and parameters $m = 10$, $k = 4$, $\lambda_q = 0.8$ and $\lambda_p = \sin(20\,\text{deg})$. Furthermore, we limit the maximum momentum to $\sin(85\,\text{deg})\, n(z, q)$. Then, with mesh spacings $\Delta q_{\max} = 0.1$, $\Delta p \approx 0.11$, and taking $N = 6$, we compute with the ALE-ADER-DG scheme the numerical solutions. The resulting distributions are shown in Figure 5.13, where the initial condition and the numerical solutions at various z-levels are shown. At $z = 0.15$ a large part of the initial condition has been refracted at the lens surface and at $z = \frac{1}{2} Z_{\text{target}}$ everything has been refracted. At $z = \frac{4}{5} Z_{\text{target}}$ some of the light has been reflected, resulting in the small patches at the top and bottom. At $z = \frac{9}{10} Z_{\text{target}}$ and $z = Z_{\text{target}}$ one can see that more of the light has been reflected. Furthermore, the light remains contained within the dielectric medium n_1 as expected.

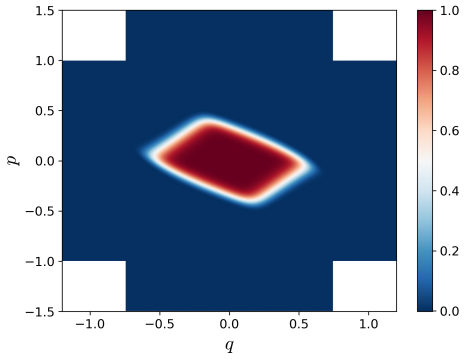
As was done in the meniscus lens example, we will compare quasi-Monte Carlo ray tracing and the ALE-ADER-DG scheme for computing the illu-



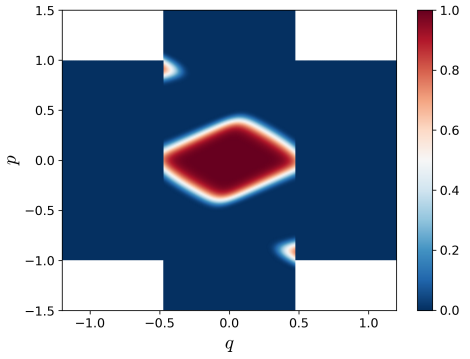
(a) $z = 0$.



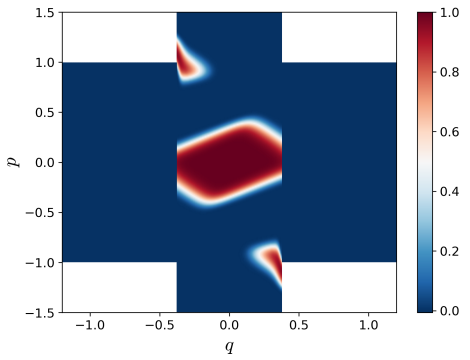
(b) $z = 0.15$.



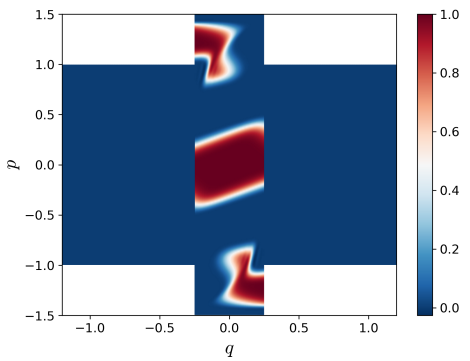
(c) $z = \frac{1}{2} Z_{\text{target}}$.



(d) $z = \frac{4}{5} Z_{\text{target}}$.



(e) $z = \frac{9}{10} Z_{\text{target}}$.



(f) $z = Z_{\text{target}}$.

Figure 5.13: Distributions of ρ for the DTIRC computed with the $N = 6$ ALE-ADER-DG scheme.

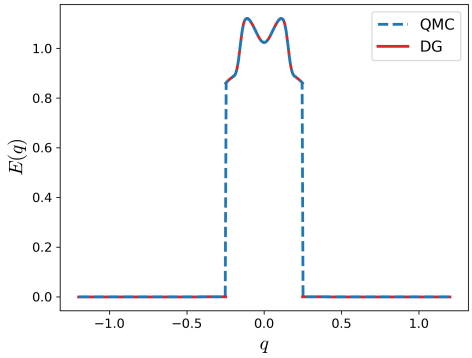


Figure 5.14: Illuminance at $z = Z_{\text{target}}$ for the DTIRC computed with quasi-Monte Carlo ray tracing (QMC) on $B = 400$ bins and the $N = 6$ ALE-ADER-DG (DG) scheme.

minance. For this example, we modify the quasi-Monte Carlo ray tracing grid to ensure no bin cuts the side walls given by $q = \pm Q_{\text{top}}(Z_{\text{target}}) \approx \pm 0.248562$. Specifically, we modify the grid to be piecewise uniform, so that the grid spacing is uniform on the q-intervals $[-1.2, -Q_{\text{top}}(Z_{\text{target}})]$, $[-Q_{\text{top}}(Z_{\text{target}}), Q_{\text{top}}(Z_{\text{target}})]$ and $[Q_{\text{top}}(Z_{\text{target}}), 1.2]$. In quasi-Monte Carlo ray tracing we compute exact intersections with the circle, whereas for intersections with the side wall we employ a version of Newton's method that resorts to bisection when necessary.

The resulting illuminance at $z = Z_{\text{target}}$ for both methods is shown in Figure 5.14, where for QMC we use $B = 400$ bins and $N_{\text{RT}} = 8 \cdot 10^6$ rays and for the ALE-ADER-DG scheme we integrate the solution shown in Figure 5.13. In the figure the solutions for both methods are almost indistinguishable by eye.

Next, we compare the performance of both methods where we once again compute the error as the L_∞-norm of the average illuminance. To compute the error, we use a reference solution computed with the ALE-ADER-DG scheme with $N = 7$, $\Delta q_{\max} = 0.025$ and $\Delta p = 0.0125$ ($r = 4$ in (5.62)).

For quasi-Monte Carlo ray tracing we fix the number of bins to $B = 400$. The comparison between quasi-Monte Carlo ray tracing and the ALE-ADER-DG scheme is plotted in Figure 5.15. For QMC the first data point corresponds to $N_{\text{RT}} = 31250$, whereas the last data point corresponds to $N_{\text{RT}} = 2.048 \cdot 10^9$ rays. For the ALE-ADER-DG method we choose the mesh spacings (5.62) with $r = 0, 1, 2, 3$. One can observe from Figure 5.15, that the ALE-ADER-DG scheme with $N = 3$ for $r = 1, 2$ is beaten by quasi-Monte Carlo ray tracing in terms of accuracy whereas for $r = 3$ both methods perform similar. For $N = 5$ and $N = 7$ the ALE-ADER-DG scheme achieves better accuracy. We remark that this particular example is computationally expensive for the ALE-ADER-

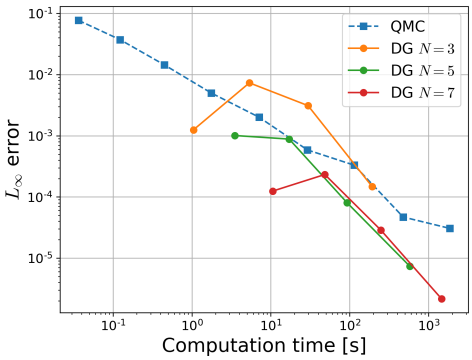


Figure 5.15: Comparison between quasi-Monte Carlo (QMC) ray tracing and ALE-ADER-DG scheme (DG) for the dielectric TIR concentrator.

DG scheme since a significant part of the mesh does not cover the solution. For example, the solution is only active in $[-Q_{\text{top}}(Z_{\text{target}}), Q_{\text{top}}(Z_{\text{target}})]$ as can be seen in Figure 5.13. To reduce the computational cost in the inactive region, one could use an adaptive mesh refinement approach with a refinement criterion based on the solution, see for example [33].

5.8 Concluding remarks

We have solved Liouville's equation for two-dimensional optical systems on a moving mesh using the ALE-ADER-DG scheme. The non-local boundary conditions at optical interfaces are dealt with in an energy-conserving manner. We numerically verified that the optical interface discretisation is energy-conserving up to machine precision. In the ALE-ADER-DG scheme an arbitrary order of accuracy for smooth solutions can be chosen both in space and the evolution coordinate z. The expected order of convergence was verified in a numerical example. Moreover, in the ADER approach we made a distinction between moving and static elements. For the mesh velocity we found that letting only elements that are adjacent to an optical interface move, leads to a more efficient scheme as static elements are cheaper to update than moving elements.

The performance of the ALE-ADER-DG scheme was compared to quasi-Monte Carlo ray tracing for computing the illuminance. The numerical experiments show that the ALE-ADER-DG scheme is much more efficient than QMC ray tracing for computing high-accuracy solutions. Moreover, in the first example the ALE-ADER-DG scheme achieves two orders of magnitude lower error compared to QMC ray tracing within only 10 seconds of computation

time. In the second example, the ALE-ADER-DG scheme still outperforms QMC ray tracing although less pronounced. For this particular example, the ALE-ADER-DG scheme is computationally expensive since the solution is only active within a small region. Hence, an adaptive mesh refinement approach will reduce the computational cost in the inactive region.

Instead of pursuing an adaptive mesh refinement approach, we will take another route to increase performance. In the next chapter, we will combine the ALE-ADER-DG scheme with a semi-Lagrangian DG method. Moreover, we let go of the very strict global stepsize and instead allow (groups of) elements to update with a locally chosen stepsize.

Chapter 6

A hybrid semi-Lagrangian DG and ADER-DG solver on a moving mesh for 2D optics

Solving Liouville's equation using the ADER-DG scheme from the previous chapter can lead to various inefficiencies in the performance, depending on the considered optical system. To that end, a solver is developed that mitigates these issues[1]. Recall that Liouville's equation describes the evolution of the basic luminance ρ on phase space, which for two-dimensional optics reads

$$\frac{\partial \rho_\sigma}{\partial z} + \nabla \cdot (\rho_\sigma \boldsymbol{u}) = 0, \tag{6.1a}$$

with the velocity field $\boldsymbol{u}$ given by

$$\boldsymbol{u} = \begin{pmatrix} u_0 \\ u_1 \end{pmatrix} = \frac{1}{p_z} \begin{pmatrix} p \\ n \frac{\partial n}{\partial q} \end{pmatrix}, \tag{6.1b}$$

where $p_z = \sigma \sqrt{n^2 - p^2}$. At an optical interface the jump condition describes non-local boundary conditions for ρ, for which we take the jump condition (2.42).

In Liouville's equation (6.1) the z-coordinate is used as an evolution coordinate and phase space is defined at $z = \text{const}$ planes. A curved optical interface is, therefore, represented as a moving boundary in phase space. In

[1] This chapter is based on the submitted article: R. A. M. van Gestel, M. J. H. Anthonissen, J. H. M. ten Thije Boonkkamp, and W. L. IJzerman. A hybrid semi-Lagrangian DG and ADER-DG solver on a moving mesh for Liouville's equation of geometrical optics. *submitted to Journal of Computational Physics*, 2023.

the Chapter 5 we employed an Arbitrary Derivative discontinuous Galerkin (ADER-DG) method on a moving mesh such that the mesh can be aligned with optical interfaces. The ADER-DG scheme allows for arbitrary high order of accuracy in space and time. Here, time corresponds to the evolution coordinate z. Optical interfaces can be of various shapes and therefore accommodating wildly varying interfaces can lead to large dynamical changes of the mesh. Even worse is that very small elements cannot always be avoided. As the ADER-DG method is fully explicit it has to obey a CFL condition that restricts the stepsize along the z-axis.

A common situation in non-imaging optics is that the refractive index field is piecewise constant. Hence, away from optical interfaces the characteristics of Liouville's equation (6.1) are simply straight lines. Recall that the characteristics coincide with light rays; see Section 2.3. The property that characteristics reduce to straight lines was used in previous chapter to simplify the Taylor expansion in the ADER approach.

In this chapter, we will exploit once more the fact that characteristics reduce to straight lines in specific regions. Specifically, this property is used in a semi-Lagrangian discontinuous Galerkin (SLDG) scheme. The SLDG scheme will only be employed away from optical interfaces.

An important feature of SLDG methods is that they can be CFL-free, allowing the use of very large stepsizes. Semi-Lagrangian (discontinuous Galerkin) methods are especially popular in the Vlasov(-Poisson) simulation community [10, 13, 34, 35, 75, 80] and they are used for atmospheric modelling [40, 46]. These methods update the numerical solution in each time step by using an approximate or exact evolution of the old solution, which for Liouville's equation means that the solution is propagated along the characteristics. SLDG methods are attractive for Liouville's equation in regions where the exact evolution is simple to compute, making the SLDG method way more efficient than the ADER-DG method.

When solving Liouville's equation we cannot choose an arbitrarily large stepsize for the proposed SLDG scheme, since characteristics striking an optical interface would lead to complicated behaviour. Therefore, we do not allow characteristics emanating from SLDG type elements to cross an optical interface. This in turn restricts the maximum stepsize that can be taken.

The more general situation of allowing characteristics to cross an optical interface, would lead to very complicated integration formulae if high integration accuracy is desired and, therefore, seems impossible to implement in an efficient manner. Especially, considering that the discontinuous change in momentum at an optical interface depends on the unit surface normal at the point the light ray strikes the optical interface, which can vary between

different light rays. Moreover, inaccuracies in the evaluation of the weak formulation due to the usage of numerical integration can result in a loss of the CFL-free property; see [75].

Close to an optical interface we opt to use an ADER-DG scheme, hence, the jump condition (2.42) only needs to be taken into account in the ADER-DG scheme. The discretisation for the ADER-DG scheme and the discretisation of the jump condition were described in Chapter 5.

Several improvements can be made over the ADER-DG solver discussed in Chapter 5 when trying to accommodate the optical interfaces. In particular, we will improve upon three main issues that limit the efficiency of the solver. First, very small elements cannot always be avoided when we align the mesh with optical interfaces. By the CFL condition these elements reduce the stepsize and because of a global time stepping algorithm all elements in the mesh are affected. Second, the CFL condition for moving elements depends on how fast they move. At an optical interface described by $q = Q(z)$, the aligning of the mesh with the optical interface requires at least one edge of the element to move at a speed $\frac{\mathrm{d}Q}{\mathrm{d}z}$. It can happen that the maximum stepsize is impeded by a large value of $\left|\frac{\mathrm{d}Q}{\mathrm{d}z}\right|$. Third, if on a large z-interval $[z_0, z_1]$ the refractive index field remains constant over the entire mesh, then we know that all characteristics will be simply straight lines. Nevertheless, the ADER-DG scheme still has to obey a CFL condition and thus wastes valuable computational resources.

The third issue is easily resolved by switching to an SLDG scheme in the proper region. The SLDG scheme can take one big step from $z = z_0$ to $z = z_1$, owing to its CFL-free nature. The first and second issue are resolved by employing local time stepping. In the most extreme case local time stepping allows every single element in the mesh to run at its own local time step. This is for instance used in [27, 31]. Here, local time stepping is used in a clustered way, i.e., the local time stepping is only (q, z)-dependent, thus independent of the momentum p. Furthermore, elements updated via ADER-DG that are close to an optical interface are required to take the same (local) stepsize, and also elements updated via SLDG always take the same stepsize. The coupling between the SLDG and ADER-DG elements is facilitated with the aid of an intermediate ADER-DG element that connects both elements. The intermediate element has specific restrictions to ensure efficiency and has a different stepsize compared to its neighbours.

In summary, the hybrid solver developed in this chapter combines the SLDG and the ADER-DG schemes together with local time stepping. The hybrid solver naturally divides the considered two-dimensional position domain $(q, z) \in \mathcal{Q} \subset \mathbb{R}^2$ into different regions, describing where the SLDG scheme

and where the ADER-DG scheme need to be used. An intermediate element efficiently couples an SLDG region with an ADER-DG region.

This chapter is organised as follows. In Section 6.1 the setup of the hybrid solver is first elaborated. Thereafter, in Sections 6.2-6.4 the new components for the hybrid solver are presented. In Section 6.5 the hybrid solver is tested on some problems, validating the high order of accuracy and efficiency of the proposed scheme. Moreover, the performance of the solver is compared to the more traditional method of quasi-Monte Carlo ray tracing for computing the illuminance.

6.1 Setup of the hybrid solver

In what follows the main topic of interest is the discretisation of Liouville's equation but not the discretisation of the jump condition (2.42) at an optical interface. It suffices to consider only forward-propagating light rays for the following discussion, as a change in propagation direction (σ) can only happen at an optical interface. Hence, we take $\sigma = 1$ in (6.1) and omit the σ-subscript to write Liouville's equation as

$$\frac{\partial \rho}{\partial z} + \nabla \cdot (\rho \boldsymbol{u}) = 0, \tag{6.2a}$$

with the velocity field $\boldsymbol{u}$ given by

$$\boldsymbol{u} = \begin{pmatrix} u_0 \\ u_1 \end{pmatrix} = \frac{1}{\sqrt{n^2 - p^2}} \begin{pmatrix} p \\ n \frac{\partial n}{\partial q} \end{pmatrix}. \tag{6.2b}$$

In the hybrid solver we divide the two-dimensional position domain $\mathcal{Q}$ into different types of elements, where away from optical interfaces the efficient SLDG scheme is used and close to optical interfaces ADER-DG elements are used. The SLDG elements are not restricted by a CFL condition unlike the ADER-DG elements and, consequently, we take larger steps with SLDG elements than with the ADER-DG elements. This results in hanging nodes along the z-axis.

An example of a qz-grid in the vicinity of an optical interface can be seen in Figure 6.1. Here, we distinguish three types of elements which are the SLDG elements in red, the ADER-DG elements in gray and the ADER-DG elements in blue that are adjacent to an SLDG element. The mesh is aligned with the optical interface by letting the gray ADER-DG elements adjacent to the optical interface move. That is, the optical interface coincides with edges of elements that touch the optical interface. The blue ADER-DG elements are added

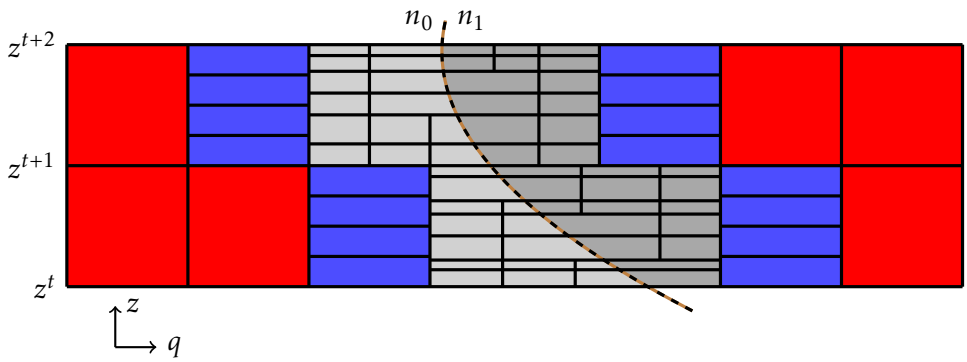


Figure 6.1: Sketch of the qz-grid showing the usage of local time stepping. Gray indicates ADER-DG elements with lighter shade representing elements in the medium with refractive index n_0 and darker shade elements in the medium with refractive index n_1. Red indicates SLDG elements, blue indicates ADER-DG elements that couple to an SLDG element, and the brown-black dashed line indicates the optical interface. The optical interface coincides with edges of elements that touch the optical interface.

specifically to allow an efficient transition region from SLDG elements to gray ADER-DG elements. These blue ADER-DG elements together with adjacent SLDG elements form a local uniform grid such that the SLDG elements are updated in an efficient manner. This will be explained in more detail in Section 6.4.

In the figure, one can also see the effect of local time stepping. The SLDG elements take one direct step to the next level, whereas the blue ADER-DG elements require multiple steps to reach the same level. The gray ADER-DG elements take steps that do not match with their blue neighbours, which leads to hanging nodes along the z-axis.

In what follows, we will first discuss the SLDG scheme and, thereafter, discuss the local conservation property of the scheme. This result is then used to describe a conservative coupling with local time stepping between ADER-DG and SLDG elements.

6.2 Semi-Lagrangian DG

Semi-Lagrangian DG schemes can be formulated in different ways. The formulation we use reads

$$\int_\Omega \rho^{t+1} \hat{\phi}_k \, \mathrm{d}\mathcal{U} = \int_\Omega T_{\Delta z}\left(\rho^t\right) \hat{\phi}_k \, \mathrm{d}\mathcal{U}, \tag{6.3}$$

with $\mathrm{d}\mathcal{U} = \mathrm{d}q\mathrm{d}p$. Here, T_τ denotes the exact evolution operator associated with Liouville's equation (6.2), which is defined so that $T_\tau(\rho^t)$ denotes the exact evolution of ρ^t, that starts at $z = z^t$ and propagates to $z = z^t + \tau$. Equation (6.3), with a polynomial expansion for ρ^{t+1}, can be interpreted as an exact advection of the old solution ρ^t to $z = z^{t+1}$, followed by an L_2-projection into the polynomial expansion. For a generic velocity field it can be complicated to evaluate (6.3) exactly. However, for a constant refractive index n one can easily compute the exact evolution by using the method of characteristics [65]. The action of the exact evolution operator simply reads

$$T_\tau\left(\rho^t\right) = \rho^t\left(q - \tau\frac{p}{\sqrt{n^2 - p^2}}, p\right). \tag{6.4}$$

Equation (6.3) with $T_\tau(\rho^t)$ given by (6.4) is solved in the following manner. We assume, for elements that are updated via this formulation, that the mesh is locally uniform in the grid spacing Δq. Consider the rectangular element $\Omega = [Q_0, Q_0 + \Delta q] \times [P_0, P_0 + \Delta p]$. On this rectangular element we consider the expansion of ρ^t in basis functions similar to (5.14). The SLDG scheme requires values from multiple elements to update the solution, hence, it is convenient to work with an expansion of ρ^t that is defined in terms of phase space coordinates $\boldsymbol{x} = (q, p)$. Specifically, the expansion of ρ on the element Ω reads

$$\rho_{\mathrm{h}}^t(\boldsymbol{x}) = \sum_{l=1}^{N_d} \rho_l^t \hat{\phi}_l(q,p) = \sum_{i,j=0}^{N} \rho_{ij}^t \ell_i\left(\frac{q - Q_0}{\Delta q}\right)\ell_j\left(\frac{p - P_0}{\Delta p}\right) \quad \boldsymbol{x} \in \Omega. \tag{6.5}$$

Here, the basis functions $\hat{\phi}_l$ are defined as follows

$$\hat{\phi}_l(q,p) = \ell_i\left(\frac{q - Q_0}{\Delta q}\right)\ell_j\left(\frac{p - P_0}{\Delta p}\right) \text{ with } l = (N+1)j + i + 1. \tag{6.6}$$

At this point it is important to stress the fact that $\rho_{\mathrm{h}}^t(\boldsymbol{x})$ denotes the combined solution of all elements, where on each element $\rho_{\mathrm{h}}^t(\boldsymbol{x})$ is given by (6.5).

Since ρ^{t+1} is expanded into the set of basis functions $\hat{\phi}_l$ the left-hand side of (6.3) can be readily evaluated due to the orthogonality of these basis functions, cf. (3.21), as follows

$$\begin{aligned}\int_\Omega \rho_{\mathrm{h}}^{t+1}\hat{\phi}_k \,\mathrm{d}\mathcal{U} &= \sum_{l=1}^{N_d} \rho_l^{t+1} \int_\Omega \hat{\phi}_l(q,p)\hat{\phi}_k(q,p)\,\mathrm{d}q\mathrm{d}p \\ &= \mathcal{J}\sum_{n,m=0}^{N} \rho_{nm}^{t+1} \int_\chi \ell_n(\xi)\ell_m(\eta)\ell_i(\xi)\ell_j(\eta)\,\mathrm{d}\xi\,\mathrm{d}\eta \\ &= \mathcal{J} w_i w_j \rho_{ij}^{t+1},\end{aligned} \tag{6.7}$$

with $\mathcal{J} = \Delta q \Delta p$. For the right-hand side of (6.3) with (6.4) substituted, the integral reads

$$\int_\Omega T_{\Delta z}\left(\rho_\mathrm{h}^t\right)\hat{\phi}_k \,\mathrm{d}\mathcal{U} = \int_{Q_0}^{Q_0+\Delta q}\int_{P_0}^{P_0+\Delta p} \rho_\mathrm{h}^t\left(q - \Delta z \frac{p}{\sqrt{n^2-p^2}}, p\right)\hat{\phi}_k(q,p)\,\mathrm{d}p\mathrm{d}q. \tag{6.8}$$

The momentum integral in (6.8) is approximated by $(N+1)$-point Gauss-Legendre quadrature and the Kronecker property (3.7) is applied, such that we obtain

$$\begin{aligned}\int_{Q_0}^{Q_0+\Delta q}\int_{P_0}^{P_0+\Delta p} \rho_\mathrm{h}^t\left(q - \Delta z \frac{p}{\sqrt{n^2-p^2}}, p\right)\hat{\phi}_k(q,p)\,\mathrm{d}p\mathrm{d}q \\ \approx w_j \Delta p \int_{Q_0}^{Q_0+\Delta q} \rho_\mathrm{h}^t\left(q-\delta_j, p_j\right)\ell_i\left(\frac{q-Q_0}{\Delta q}\right)\mathrm{d}q,\end{aligned} \tag{6.9a}$$

with

$$\delta_j = \Delta z \frac{p_j}{\sqrt{n^2-p_j^2}}, \tag{6.9b}$$

and $p_j = P_0 + \xi_j \Delta p$. The remaining integrand is not continuous, instead, it is piecewise polynomial. Away from the boundary of phase space, there will be only one discontinuity over the integration interval for every p_j since the mesh is locally uniform in the mesh spacing Δq. The location of the discontinuity is denoted as $Q_0 + \alpha_j \Delta q$ with $0 \le \alpha_j < 1$, and can be easily computed; see Figure 6.2. The remaining q-integral is then split into two parts so that both integrands are simply polynomials of degree $2N$ in q, i.e.,

$$\begin{aligned}\int_{Q_0}^{Q_0+\Delta q} \rho_\mathrm{h}^t\left(q-\delta_j, p_j\right)\ell_i\left(\frac{q-Q_0}{\Delta q}\right)\mathrm{d}q = \int_{Q_0}^{Q_0+\alpha_j\Delta q} \rho_\mathrm{h}^t\left(q-\delta_j, p_j\right)\ell_i\left(\frac{q-Q_0}{\Delta q}\right)\mathrm{d}q \\ + \int_{Q_0+\alpha_j\Delta q}^{Q_0+\Delta q} \rho_\mathrm{h}^t\left(q-\delta_j, p_j\right)\ell_i\left(\frac{q-Q_0}{\Delta q}\right)\mathrm{d}q.\end{aligned} \tag{6.10}$$

The remaining integrals are exactly evaluated with $(N+1)$-point Gauss-Legendre quadrature.

We remark that boundary conditions are already included in the action of the exact evolution operator on ρ_h^t. In the examples studied in Section 6.5 there are zero-inflow boundary conditions on the (appropriate) boundary of phase space. Hence, it can happen that because of these boundary conditions an integral in the right-hand side of (6.10) evaluates to 0.

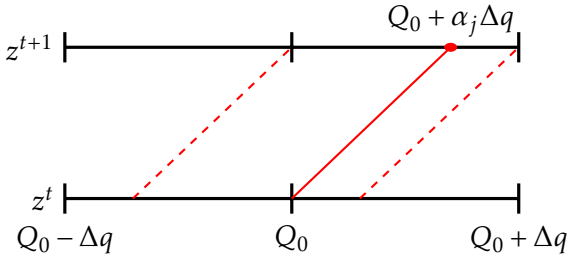


Figure 6.2: Sketch of locating the discontinuity on an element for a fixed momentum value. Specifically the red lines are light rays with fixed momentum.

Inserting relations (6.7), (6.9a) and (6.10) into (6.3) leads to the following update for the expansion coefficients

$$\begin{aligned}\Delta q\, w_i \rho_{ij}^{t+1} = &\int_{Q_0}^{Q_0+\alpha_j\Delta q} \rho_{\mathrm{h}}^t\left(q-\delta_j, p_j\right)\ell_i\left(\frac{q-Q_0}{\Delta q}\right)\mathrm{d}q \\ &+ \int_{Q_0+\alpha_j\Delta q}^{Q_0+\Delta q} \rho_{\mathrm{h}}^t\left(q-\delta_j, p_j\right)\ell_i\left(\frac{q-Q_0}{\Delta q}\right)\mathrm{d}q,\end{aligned} \tag{6.11}$$

for $i = 0,\ldots,N$, $j = 0,\ldots,N$.

One downside of SLDG schemes is that if the stepsize or mesh spacing dynamically changes during the stepping procedure, then the integrals in (6.11) have to be re-evaluated each step for each element. The evaluation of Lagrange polynomials is quite expensive, hence, for efficiency purposes we require the mesh to be locally uniform with grid spacing Δq and also require the semi-Lagrangian stepsize Δz_{SL} to be uniform for multiple steps. The right-hand side of (6.11) is written in terms of two matrix-vector products for $j = 0,1,\ldots,N$. The details of these steps can be found in Section 3.3. These matrices are cached and reused for multiple steps. Sometimes we have to change the stepsize Δz_{SL} to accommodate optical interfaces, so that a re-computation of these matrices is unavoidable. This will be made more clear for the examples discussed in Section 6.5.

6.3 Local conservation property

In what follows, we will couple SLDG elements to neighbouring ADER-DG elements using fluxes defined on their common boundary. The SLDG scheme uses a completely different principle compared to the ADER-DG scheme, hence, it is not apparent on how these elements can be coupled in an energy-conserving manner. Furthermore, it is no longer clear in the SLDG scheme

whether it is energy conservative, due to the approximation of the momentum integral. Therefore, we will rewrite the SLDG scheme to show that it satisfies a local energy balance. That is, the change in the integral of ρ on an element is directly related to the fluxes on the boundary of the element. Similarly, for a static ADER-DG element we will rewrite the formulation in terms of phase space coordinates, rather than reference domain coordinates. These results will be important in describing the fluxes between SLDG and ADER-DG elements using local time stepping in Section 6.4.

6.3.1 SLDG

Let $\int_{I,N} g(x)\,\mathrm{d}x$, for an arbitrary interval $I = [a, b]$, denote the approximation of $\int_I g(x)\,\mathrm{d}x$ by the $(N+1)$-point Gauss-Legendre quadrature rule. Thus $\int_{I,N} g(x)\,\mathrm{d}x$ has the meaning

$$\int_{I,N} g(x)\,\mathrm{d}x = |I| \sum_{n=0}^{N} w_n g(a + |I|\,\xi_n), \tag{6.12}$$

with $|I| = b - a$, and $\{w_n\}_{n=0}^N$ and $\{\xi_n\}_{n=0}^N$ denoting the quadrature weights and points on the interval $[0, 1]$. The same notation is used for multidimensional integrals, where the multidimensional integral is evaluated as an iterated integral. Let $Q = [Q_0, Q_0 + \Delta q]$ and $P = [P_0, P_0 + \Delta p]$, so that $\Omega = Q \times P$. The SLDG scheme (6.11) can be rewritten with the aid of the above notation.

In the derivation of (6.11), we have approximated the momentum integral and inserted the exact evolution operator (6.4). With the introduced notation, the SLDG scheme can, starting from equation (6.3), be rewritten as

$$\int_\Omega \rho_{\mathrm{h}}^{t+1} \hat{\phi}_k \,\mathrm{d}\mathcal{U} = \int_{P,N} \int_Q T_{\Delta z}\left(\rho_{\mathrm{h}}^t\right) \hat{\phi}_k \,\mathrm{d}\mathcal{U}, \tag{6.13}$$

with $T_\tau(\rho_{\mathrm{h}}^t)$ given by (6.4). The integral over Q is exactly evaluated in the SLDG scheme as described in Section 6.2. Note that (6.13) represents a reformulation of the SLDG scheme (6.11) multiplied by $\Delta p\, w_j$. This can be seen by applying relation (6.7), by writing out the quadrature rule for the momentum integral on the right-hand side of (6.13) and substituting (6.4), i.e.,

$$\mathcal{J} w_i w_j \rho_{ij}^{t+1} = \Delta p\, w_j \int_Q \rho_{\mathrm{h}}^t \left(q - \Delta z \frac{p_j}{\sqrt{n^2 - p_j^2}}, p_j \right) \ell_i\left(\frac{q - Q_0}{\Delta q}\right) \mathrm{d}q.$$

The right-hand side of equation (6.13) can be rewritten as

$$\int_{P,N}\int_{Q} T_{\Delta z}\left(\rho_{\mathrm{h}}^{t}\right)\hat{\phi}_k \,\mathrm{d}\mathcal{U} = \int_{P,N}\int_{Q}\int_{Z}\frac{\partial}{\partial\tau}\left(T_{\tau}\left(\rho_{\mathrm{h}}^{t}\right)\right)\mathrm{d}\tau\,\hat{\phi}_k \,\mathrm{d}\mathcal{U} + \int_{P,N}\int_{Q} T_0\left(\rho_{\mathrm{h}}^{t}\right)\hat{\phi}_k \,\mathrm{d}\mathcal{U}, \tag{6.14}$$

with $Z = [0, \Delta z]$. Inserting Liouville's equation (6.2) for a constant refractive index field leads to

$$\begin{aligned}\int_{P,N}\int_{Q} T_0\left(\rho_{\mathrm{h}}^{t}\right)\hat{\phi}_k \,\mathrm{d}\mathcal{U} + \int_{P,N}\int_{Q}\int_{Z}\frac{\partial}{\partial\tau}\left(T_{\tau}\left(\rho_{\mathrm{h}}^{t}\right)\right)\mathrm{d}\tau\,\hat{\phi}_k \,\mathrm{d}\mathcal{U} \\ = \int_{\Omega}\rho_{\mathrm{h}}^{t}\,\hat{\phi}_k \,\mathrm{d}\mathcal{U} + \int_{P,N}\int_{Q}\int_{Z}\frac{\partial}{\partial q}\left(-u_0 T_{\tau}\left(\rho_{\mathrm{h}}^{t}\right)\right)\mathrm{d}\tau\,\hat{\phi}_k \,\mathrm{d}\mathcal{U},\end{aligned} \tag{6.15}$$

where we have used that T_0 is simply the identity operator, i.e., $T_0(\rho_{\mathrm{h}}^{t}) = \rho_{\mathrm{h}}^{t}$, and thus the momentum integral in the first term is exactly evaluated. As the integral over Q is exactly evaluated, we can apply integration by parts and change the order of integration to obtain

$$\begin{aligned}&\int_{\Omega}\rho_{\mathrm{h}}^{t}\,\hat{\phi}_k \,\mathrm{d}\mathcal{U} + \int_{P,N}\int_{Q}\int_{Z}\frac{\partial}{\partial q}\left(-u_0 T_{\tau}\left(\rho_{\mathrm{h}}^{t}\right)\right)\mathrm{d}\tau\,\hat{\phi}_k \,\mathrm{d}\mathcal{U} \\ &\quad = \int_{\Omega}\rho_{\mathrm{h}}^{t}\,\hat{\phi}_k \,\mathrm{d}\mathcal{U} + \int_{P,N}\int_{Z}\left(\int_{Q}\frac{\partial\hat{\phi}_k}{\partial q}u_0 T_{\tau}\left(\rho_{\mathrm{h}}^{t}\right)\mathrm{d}q - \left[\hat{\phi}_k u_0 T_{\tau}\left(\rho_{\mathrm{h}}^{t}\right)\right]_{q=Q_0}^{Q_0+\Delta q}\right)\mathrm{d}\tau\mathrm{d}p.\end{aligned} \tag{6.16}$$

Hence, by inserting the result (6.16) into (6.13) we obtain

$$\begin{aligned}\int_{\Omega}\rho_{\mathrm{h}}^{t+1}\hat{\phi}_k \,\mathrm{d}\mathcal{U} - \int_{\Omega}\rho_{\mathrm{h}}^{t}\hat{\phi}_k \,\mathrm{d}\mathcal{U} = \int_{P,N}\int_{Z}\left(\int_{Q}\frac{\partial\hat{\phi}_k}{\partial q}u_0 T_{\tau}\left(\rho_{\mathrm{h}}^{t}\right)\mathrm{d}q \right. \\ \left. - \left[\hat{\phi}_k u_0 T_{\tau}\left(\rho_{\mathrm{h}}^{t}\right)\right]_{q=Q_0}^{Q_0+\Delta q}\right)\mathrm{d}\tau\mathrm{d}p.\end{aligned} \tag{6.17}$$

Summing over all test functions and using that $\sum_k \hat{\phi}_k = 1$ leads to

$$\int_{\Omega}\rho_{\mathrm{h}}^{t+1}\,\mathrm{d}\mathcal{U} - \int_{\Omega}\rho_{\mathrm{h}}^{t}\,\mathrm{d}\mathcal{U} = -\int_{P,N}\int_{Z}\left[u_0 T_{\tau}\left(\rho_{\mathrm{h}}^{t}\right)\right]_{q=Q_0}^{Q_0+\Delta q}\mathrm{d}\tau\mathrm{d}p, \tag{6.18}$$

that is, the change in the integral of ρ on an element is directly related to the fluxes on the boundary of the element. The mesh is constructed so that away from optical interfaces two neighbouring elements share an entire edge, e.g., the edge $\{Q_0\} \times P$. Thus the momentum quadrature nodes along this edge coincide for both elements. This implies that the flux leaving an element at this edge is entering the neighbouring element. The formulation (6.17) will serve as a basis for coupling ADER-DG and SLDG elements in a conservative manner.

6.3.2 ADER-DG for static element

The ADER-DG scheme (5.11) can be rewritten in terms of integrals over the physical phase space domain, rather than the reference domain. For a static element we can write the mapping between the two domains as

$$\boldsymbol{x}(\boldsymbol{\xi}) = \begin{pmatrix} q(\xi) \\ p(\eta) \end{pmatrix} = \begin{pmatrix} Q_0 + \xi \Delta q \\ P_0 + \eta \Delta p \end{pmatrix}, \tag{6.19}$$

where $\boldsymbol{x} \in \Omega = [Q_0, Q_0 + \Delta q] \times [P_0, P_0 + \Delta p]$. The left-hand side of equation (5.11) is rewritten by transforming the integrals using the above mapping, i.e.,

$$\int_\chi \phi_k \rho^t \mathcal{J} \, \mathrm{d}\boldsymbol{\xi} = \int_\Omega \hat{\phi}_k \rho^t \, \mathrm{d}\mathcal{U}, \tag{6.20}$$

where the basis functions on the reference domain ϕ_k are related to basis functions on Ω by $\hat{\phi}_k(\boldsymbol{x}(\boldsymbol{\xi})) = \phi_k(\boldsymbol{\xi})$. Recall that for a static element $\mathcal{J} = \Delta q \Delta p$ is a constant.

For the terms on the right-hand side of equation (5.11) we apply the chain rule to write

$$\nabla_{\boldsymbol{\xi}} \phi_k(\boldsymbol{\xi}) = \nabla_{\boldsymbol{\xi}} \hat{\phi}_k(\boldsymbol{x}(\boldsymbol{\xi})) = \begin{pmatrix} \frac{\partial \hat{\phi}_k}{\partial \boldsymbol{x}} \cdot \frac{\partial \boldsymbol{x}}{\partial \xi} \\ \frac{\partial \hat{\phi}_k}{\partial \boldsymbol{x}} \cdot \frac{\partial \boldsymbol{x}}{\partial \eta} \end{pmatrix} = \begin{pmatrix} \Delta q \, \frac{\partial \hat{\phi}_k}{\partial q} \\ \Delta p \, \frac{\partial \hat{\phi}_k}{\partial p} \end{pmatrix},$$

and thereafter taking the scalar product with $\tilde{\boldsymbol{f}}$ and using $\tilde{\boldsymbol{v}} = \boldsymbol{0}$ leads to

$$\nabla_{\boldsymbol{\xi}} \phi_k(\boldsymbol{\xi}) \cdot \tilde{\boldsymbol{f}} = \begin{pmatrix} \Delta q \, \frac{\partial \hat{\phi}_k}{\partial q} \\ \Delta p \, \frac{\partial \hat{\phi}_k}{\partial p} \end{pmatrix} \cdot \begin{pmatrix} \Delta p \, \rho u_0 \\ \Delta q \, \rho u_1 \end{pmatrix} = \mathcal{J} \left(\nabla_{\boldsymbol{x}} \hat{\phi}_k \right) \cdot (\rho \boldsymbol{u}). \tag{6.21}$$

For the boundary integral in equation (5.11) we first introduce ρ_{uw} as the upwind value of ρ at an edge, defined similarly to the upwind flux (5.12). As a result, the integrals on the right-hand side of (5.11) can be rewritten to

$$\begin{aligned} \int_Z \left[\int_\chi \left(\nabla_{\boldsymbol{\xi}} \phi_k \right) \cdot \tilde{\boldsymbol{f}} \, \mathrm{d}\boldsymbol{\xi} - \int_{\partial\chi} \phi_k \tilde{\boldsymbol{F}} \cdot \hat{\boldsymbol{N}} \, \mathrm{d}\sigma \right] \mathrm{d}\tau \\ = \int_Z \left[\int_\chi \left(\nabla_{\boldsymbol{\xi}} \phi_k \right) \cdot \tilde{\boldsymbol{f}} \, \mathrm{d}\boldsymbol{\xi} - \int_{\partial\chi} \phi_k \rho_{\mathrm{uw}} \tilde{\boldsymbol{u}} \cdot \hat{\boldsymbol{N}} \, \mathrm{d}\sigma \right] \mathrm{d}\tau \\ = \int_Z \left[\int_\Omega \left(\nabla_{\boldsymbol{x}} \hat{\phi}_k \right) \cdot (\rho \boldsymbol{u}) \, \mathrm{d}\mathcal{U} - \int_{\partial\Omega} \hat{\phi}_k \rho_{\mathrm{uw}} \boldsymbol{u} \cdot \hat{\boldsymbol{N}} \, \mathrm{d}\sigma \right] \mathrm{d}\tau, \end{aligned} \tag{6.22}$$

whereby the Jacobian determinant from (6.21) and the transformation of the integral cancel. As we consider an element $\Omega = Q \times P = [Q_0, Q_0 + \Delta q] \times [P_0, P_0 + \Delta p]$ with a constant refractive index field, the velocity simplifies to $\boldsymbol{u} = (u_0, 0)$. Hence, the integrals reduce to

$$\int_Z \left[\int_\Omega \left(\nabla_{\boldsymbol{x}} \hat{\phi}_k \right) \cdot (\rho \boldsymbol{u}) \, \mathrm{d}\mathcal{U} - \int_{\partial\Omega} \hat{\phi}_k \rho_{\mathrm{uw}} \boldsymbol{u} \cdot \hat{\boldsymbol{N}} \, \mathrm{d}\sigma \right] \mathrm{d}\tau = \int_Z \left[\int_\Omega \frac{\partial \hat{\phi}_k}{\partial q} u_0 \, \rho \, \mathrm{d}\mathcal{U} - \int_P \left[\hat{\phi}_k u_0 \, \rho_{\mathrm{uw}} \right]_{q=Q_0}^{Q_0+\Delta q} \mathrm{d}p \right] \mathrm{d}\tau. \tag{6.23}$$

Let ρ_{Taylor} denote the Taylor expansion for a static element (5.32). Then, by using relation (6.20) for the first two integrals on the left-hand side of (5.11) and with the quadrature notation introduced in Section 6.3.1, we can write the ADER-DG scheme in the following alternative representation

$$\int_\Omega \rho_{\mathrm{h}}^{t+1} \hat{\phi}_k \, \mathrm{d}\mathcal{U} - \int_\Omega \rho_{\mathrm{h}}^{t} \hat{\phi}_k \, \mathrm{d}\mathcal{U} = \int_Z \left(\int_{\Omega,N} \frac{\partial \hat{\phi}_k}{\partial q} u_0 \, \rho_{\mathrm{Taylor}} \, \mathrm{d}\mathcal{U} - \int_{P,N} \left[\hat{\phi}_k u_0 \, \rho_{\mathrm{Taylor,uw}} \right]_{q=Q_0}^{Q_0+\Delta q} \mathrm{d}p \right) \mathrm{d}\tau. \tag{6.24}$$

With this representation the ADER-DG scheme looks similar to the SLDG scheme (6.17), with the main difference being the use of an element local Taylor series rather than the exact evolution of the old solution. Summing over all test functions in equation (6.24) leads to

$$\int_\Omega \rho_{\mathrm{h}}^{t+1} \, \mathrm{d}\mathcal{U} - \int_\Omega \rho_{\mathrm{h}}^{t} \, \mathrm{d}\mathcal{U} = - \int_Z \int_{P,N} \left[u_0 \, \rho_{\mathrm{Taylor,uw}} \right]_{q=Q_0}^{Q_0+\Delta q} \mathrm{d}\tau \mathrm{d}p, \tag{6.25}$$

showing that the ADER-DG scheme also exhibits a local conservation property.

6.4 Local time stepping

The SLDG scheme (6.11) for arbitrary stepsize and mesh spacing has one big downside. Namely, the exact evaluation of the integrals requires function evaluations of test functions and the solution at many different positions, which is very expensive. In general ADER-DG elements do not have this issue. Therefore, to remedy this issue we use a local uniform mesh spacing and local uniform stepsize for semi-Lagrangian elements. The mesh spacing Δq for SLDG elements is fixed to be Δq_{SL}. The stepsize for SLDG elements Δz_{SL}

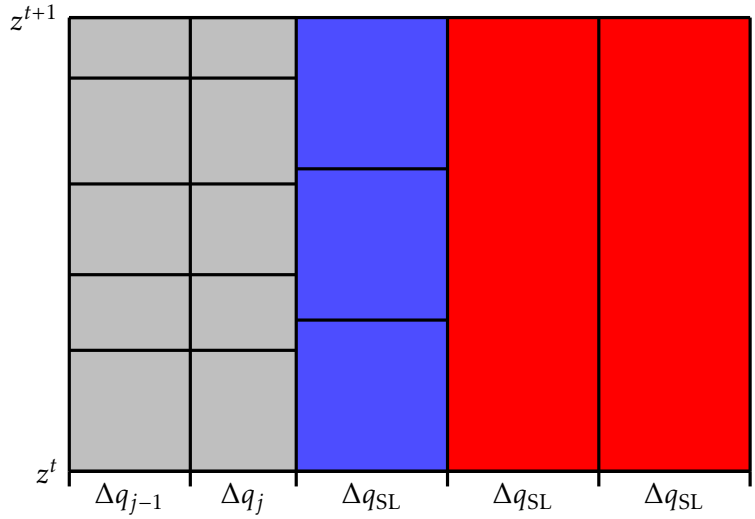


Figure 6.3: Example of local time stepping. Gray indicates standard ADER elements with full local time stepping, red indicates a semi-Lagrangian element, and blue indicates an ADER element coupling to a semi-Lagrangian element.

depends on the geometry of the optical system. For instance, if there are no optical interfaces then there is just free propagation and we will take Δz_{SL} as large as is possible considering all elements.

In case there are optical interfaces, then close to the optical interface we will use ADER-DG elements and away from the optical interface, if possible, we will use the very efficient SLDG elements. Since, we do not want characteristics crossing an optical interface, we limit the stepsize Δz_{SL} by

$$\Delta z_{\mathrm{SL}} \leq \frac{\Delta q_{\mathrm{SL}}}{u_{\max}}, \tag{6.26}$$

where $u_{\max}$ denotes the maximum absolute value of the velocity field $\boldsymbol{u}$. Using this criterion we can naturally couple SLDG and ADER-DG elements in an energy-conserving manner. The requirement that characteristics are not allowed to cross optical interfaces, places a condition on how the mesh should be chosen locally around optical interfaces.

To explain the local time stepping algorithm and the coupling between the SLDG and ADER-DG elements we consider the example drawn in Figure 6.3, which shows the geometry in qz-space without the momentum axis. A distinction between three types of elements can be made. The same colour coding of elements is used as before. Gray indicates ADER-DG elements, red indicates SLDG elements and blue indicates ADER-DG elements coupling

to an SLDG element. All elements start at the common level z^t and end at the level z^{t+1}. The SLDG elements take one direct step to the next level, whereas the ADER-DG elements take multiple steps to reach the level z^{t+1}. Furthermore, not all ADER-DG elements take the same stepsize as indicated in Figure 6.3. Hence, we have to deal with hanging nodes along the z-axis, which impacts the computation of the fluxes. What remains to describe is how an ADER-DG element and its neighbouring SLDG element are coupled, and how two ADER-DG elements are coupled. This will be discussed in what follows.

Recall that the CFL condition (5.34) for the ADER-DG scheme reads

$$\Delta z \leq \frac{1}{2d} \frac{\mathrm{CFL}}{2N+1} \min_e \frac{\Delta q_e}{|u_0 - v_0|_{\max,e}}, \tag{6.27}$$

where $2d$ denotes the dimension of phase space ($d = 1$), CFL denotes a constant coefficient that will be specified later and $|u_0 - v_0|_{\max,e}$ denotes the maximum absolute velocity on an element e. For the global time stepping used in Chapter 5 the minimum runs over all elements. On the contrary, with local time stepping the minimum should be taken over a subset of all elements. In the solver, SLDG regions (red) enclose a region of ADER-DG elements (blue and gray), where in the SLDG region the stepsize is independent from the ADER-DG region. The maximum stepsize for ADER-DG elements in an ADER-DG region is then determined by taking the minimum in (6.27) over all elements in its respective ADER-DG region.

It is important to remark that condition (6.26) does not depend on the factor $\mathrm{CFL}/(2d(2N+1))$ and the mesh velocity, in contrast to the CFL condition (6.27) for ADER-DG elements.

In Figure 6.3 the coupling ADER-DG element (blue) has the same width Δq_{SL} as the semi-Lagrangian elements. This ensures that the neighbouring SLDG element (red) can use the cached matrices to update the solution. Moreover, characteristics emanating from the common edge, shared by the SLDG and ADER-DG elements, cannot propagate to the gray ADER-DG region by virtue of condition (6.26).

6.4.1 Coupling SLDG and ADER-DG elements

Semi-Lagrangian DG elements are directly updated by taking one step with stepsize $\Delta z_{\mathrm{SL}} = z^{t+1} - z^t$ from $z = z^t$ to $z = z^{t+1}$, using the scheme (6.11). The ADER-DG (blue) element that shares an edge with an SLDG element, instead uses the scheme (5.17) and in general takes multiple (sub-)steps to reach the level $z = z^{t+1}$. ADER-DG elements normally communicate via fluxes computed

at an edge, whereas SLDG elements do not explicitly compute the flux. In Section 6.3.1 it was shown that the SLDG scheme (6.11) is equivalent to the SLDG scheme (6.17). Hence, the fluxes are implicit in the SLDG formulation. To ensure an energy-conserving coupling we will modify the fluxes at the common edge between the two element types.

Consider the following two elements, a static ADER-DG element $[Q_0 - \Delta q_{\mathrm{SL}}, Q_0] \times [P_0, P_0 + \Delta p]$ and an SLDG element $[Q_0, Q_0 + \Delta q_{\mathrm{SL}}] \times [P_0, P_0 + \Delta p]$ that share the common edge $q = Q_0$. The fluxes at the edge $q = Q_0$ need to be computed in a way that is energy-conserving. For the SLDG element the flux across the edge $q = Q_0$ over the interval $Z = [0, \Delta z_{\mathrm{SL}}]$ can be directly computed from the boundary term in the right-hand side of (6.17). On the other hand, the ADER-DG element requires N_{steps} sub-steps to reach the same level as the SLDG element. The interval Z is partitioned into N_{steps} subintervals with $0 = \tau^{(0)} < \tau^{(1)} < \ldots < \tau^{(N_{\mathrm{steps}})} = \Delta z_{\mathrm{SL}}$ denoting the sub-levels and the subintervals are given by $Z^{(n)} = [\tau^{(n)}, \tau^{(n+1)}]$ with $n = 0, \ldots, N_{\mathrm{steps}} - 1$.

For the SLDG element the boundary term at the edge $q = Q_0$ can be straightforwardly computed from the last term in (6.17) as

$$\begin{aligned} &\int_{P,N} \int_{Z} \ell_i(0) \ell_j\left(\frac{p - P_0}{\Delta p}\right) u_0(p) T_\tau(\rho_{\mathrm{h}}^t)(Q_0, p) \,\mathrm{d}\tau \mathrm{d}p \\ &\qquad = w_j \Delta p \, \ell_i(0) \int_{Z} u_0(p_j) T_\tau(\rho_{\mathrm{h}}^t)(Q_0, p_j) \,\mathrm{d}\tau, \end{aligned} \tag{6.28}$$

where we have used that $\hat{\phi}_k = \ell_i(0)\ell_j\big((p - P_0)/\Delta p\big)$ at the edge and the Kronecker property (3.7).

On the other hand, in the ADER-DG scheme (6.24) the boundary term for the nth sub-step at the edge $q = Q_0$ would normally contribute

$$\begin{aligned} &\int_{Z^{(n)}} \int_{P,N} \ell_i(1) \ell_j\left(\frac{p - P_0}{\Delta p}\right) u_0(p) \rho_{\mathrm{Taylor,uw}}(z^t + \tau, Q_0, p) \,\mathrm{d}p \mathrm{d}\tau \\ &\qquad = w_j \Delta p \, \ell_i(1) \int_{Z^{(n)}} u_0(p_j) \rho_{\mathrm{Taylor,uw}}(z^t + \tau, Q_0, p_j) \,\mathrm{d}\tau, \end{aligned} \tag{6.29}$$

where we have used that $\hat{\phi}_k = \ell_i(1)\ell_j\big((p - P_0)/\Delta p\big)$ at the edge and the Kronecker property (3.7). To have a conservative coupling between the ADER-DG and SLDG elements, it is required that the flux leaving the ADER-DG element is equal to the flux entering the SLDG element over the entire z-interval $[z^t, z^{t+1}]$ or, equivalently, over Z. To that end, we replace the flux computed in the ADER-DG scheme with the flux computed via the SLDG formulation over the sub-interval $Z^{(n)}$. Specifically, we replace the boundary contribution (6.29)

with

$$\begin{aligned}\int_{P,N}\int_{Z^{(n)}} \ell_i(1)\ell_j\left(\frac{p-P_0}{\Delta p}\right) u_0(p) T_\tau(\rho_{\mathrm{h}}^t)(Q_0,p)\,\mathrm{d}\tau\mathrm{d}p \\ = w_j \Delta p\, \ell_i(1) \int_{Z^{(n)}} u_0(p_j) T_\tau(\rho_{\mathrm{h}}^t)(Q_0,p_j)\,\mathrm{d}\tau.\end{aligned} \tag{6.30}$$

Take note that we always use the solution at level $z = z^t$ to compute (6.30), hence, these contributions are computed before taking any step with the ADER-DG element.

It is obvious that the total flux over the entire z-interval $[z^t, z^{t+1}]$ is the same for both schemes at the edge. Indeed, summing (6.30) over the sub-steps and summing over the basis functions (and recall that $\sum_i \ell_i = 1$) yields

$$\sum_{n=0}^{N_{\text{steps}}-1} \int_{P,N}\int_{Z^{(n)}} u_0(p) T_\tau(\rho_{\mathrm{h}}^t)(Q_0,p)\,\mathrm{d}\tau\mathrm{d}p = \sum_{j=0}^{N} w_j \Delta p \int_Z u_0(p_j) T_\tau(\rho_{\mathrm{h}}^t)(Q_0,p_j)\,\mathrm{d}\tau, \tag{6.31}$$

and summing (6.28) over the basis functions yields

$$\int_{P,N}\int_{Z} u_0(p) T_\tau(\rho_{\mathrm{h}}^t)(Q_0,p)\,\mathrm{d}\tau\mathrm{d}p = \sum_{j=0}^{N} w_j \Delta p \int_Z u_0(p_j) T_\tau(\rho_{\mathrm{h}}^t)(Q_0,p_j)\,\mathrm{d}\tau. \tag{6.32}$$

As the fluxes that appear in the right-hand sides of relations (6.31)-(6.32) are identical, we have that the coupling between the ADER-DG and SL-DG elements is energy conservative.

Note that the exact evolution operator (6.4) in (6.30) reads

$$T_\tau(\rho_{\mathrm{h}}^t)(Q_0,p_j) = \rho_{\mathrm{h}}^t\left(Q_0 - \tau\frac{p_j}{\sqrt{n^2-p_j^2}}, p_j\right) \tag{6.33}$$

and thus by virtue of $0 \le \tau^{(n)} \le \Delta z_{\mathrm{SL}}$ we have that the propagation distance $\tau u_0(p_j) = \tau p_j/\sqrt{n^2-p_j^2}$ is either positive or negative for all $\tau \in Z$. Therefore, only the ρ_{h}^t from either the element left or the element right of the edge $q = Q_0$ is needed to evaluate the integral. This holds for all sub-steps as by construction $\Delta z_{\mathrm{SL}} u_{\max} \le \Delta q_{\mathrm{SL}}$, see (6.26), and Δq_{SL} is the width (in the q-direction) of both elements adjacent to the edge.

The integral in the right-hand side of (6.30) can be precomputed independently of the expansion coefficients of ρ_{h}^t, such that the evaluation of (6.30) can be written in terms of a scalar product for every j. This requires that

the sub-levels $\tau^{(n)}$ for the ADER-DG element are chosen beforehand. These sub-levels are chosen according to

$$\tau^{(n)} = n\frac{\Delta z_{\mathrm{SL}}}{N_{\mathrm{steps}}} \quad \text{with } N_{\mathrm{steps}} = \left\lceil \frac{2d(2N+1)}{\mathrm{CFL}} \right\rceil, \tag{6.34}$$

with $\lceil \cdot \rceil$ denoting the ceiling operation that returns the first integer that is equal to or larger than the given argument.

6.4.2 Coupling two ADER-DG elements

The coupling of fluxes between two ADER-DG elements with local time stepping only requires modifications of the boundary integral in the ADER-DG scheme (5.17). The volume integral in (5.17) does not involve any information about the solution from neighbours and, thus, can always be directly computed using the local ADER predictor, which is given by the Taylor expansion (5.28) or (5.32). Due to the local time stepping we have hanging nodes in the z-direction; see Figure 6.3. Therefore, in the boundary integral the entire z-interval for an element is, in general, split into multiple intervals at each edge. How the z-interval at an edge is split depends on the stepsizes of the element itself and its direct neighbours. In practice, the splitting of the z-interval follows a relatively straightforward recipe as we do not allow the local ADER predictor to be evaluated backward in time, i.e., we do not evaluate the Taylor expansion for $\tau < 0$, and we first update the elements where their next level is less than or equal to the next level of its neighbouring elements.

In Figure 6.3 we consider three types of elements. The first type represents pure ADER-DG, either static or moving, elements (gray), the second type represents SLDG elements (red), and the third type represents static ADER-DG elements that share an edge with an SLDG element (blue). The gray elements are the only elements that can incorporate optical interfaces, but should also be able to move freely. As the alignment of the mesh with optical interfaces can cause a large change in an element's size we will also require a mesh refinement procedure, which was described in Section 5.6. Mesh refinement can only be performed when elements are at a common z-level, hence, gray elements always share the same z-level as neighbouring gray elements. A group of gray elements, that is defined by a gray element having at least one edge in common with another element from the group, is always updated simultaneously.

Consider the elements shown in Figure 6.4. In the local time stepping algorithm we update the elements in a certain order, that depends on which

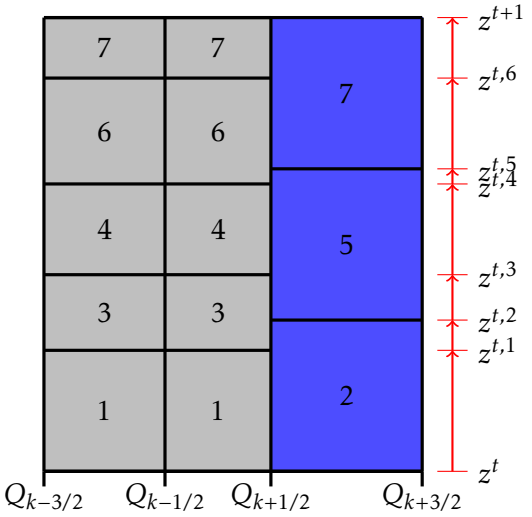


Figure 6.4: Example of local time stepping coupling between ADER elements. Gray indicates standard ADER elements and blue indicates an ADER element coupling to a semi-Lagrangian element. The numbers within the elements indicate the order in which the elements are updated.

elements can be updated given a configuration of the z-levels of each element. The presented local time stepping algorithm follows a similar description as the algorithms given in [27, 31]. As the intermediate z-levels are not synchronised between elements, it is convenient to speak of update cycles. In each update cycle only a subset of elements are updated. Since a group of gray elements is required to have one common z-level, we either need to update the gray elements or the blue elements in Figure 6.4 in an update cycle. From the figure, we see that in the first cycle we need to update the gray elements since their next level $z^{t,1}$ is the lowest among their neighbours. The numerical fluxes at the edges of the elements can directly be computed as the ADER predictor is always available. The numerical fluxes between the gray and blue elements at the common edge $q = Q_{k+1/2}$ are computed over the z-interval $[z^t, z^{t,1}]$. In the next cycle, we can update the blue elements. As part of the numerical fluxes over the common edge $q = Q_{k+1/2}$ for the z-interval $[z^t, z^{t,1}]$ have already been computed, all that remains is computing the numerical fluxes for the z-interval $[z^{t,1}, z^{t,2}]$. For this step we use the updated numerical solution for the gray elements that is defined at the level $z = z^{t,1}$. After the blue elements, we update the gray elements to the level $z = z^{t,3}$. This process is continued, resulting in integration to the z-levels shown in Figure 6.4, till finally all elements share the common level $z = z^{t+1}$.

6.4.3 Overview

The order of element updates can be summarised as follows. In the local time stepping algorithm involving all three types of elements we first need to compute the coupling fluxes between SLDG (red) and ADER-DG (blue) elements as they depend on the solution at the common level $z = z^t$. Thereafter, the SLDG elements need to be updated to the level $z = z^{t+1}$ and after that we can apply local time stepping, as described in Section 6.4.2, whereby the ADER-DG elements (blue and gray) are updated in multiple steps to reach the level $z = z^{t+1}$.

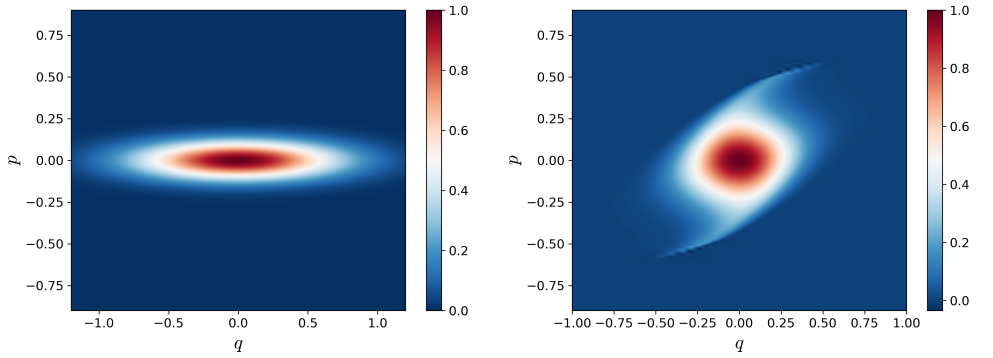
6.5 Results

In the following we will test the novel hybrid SLDG and ADER-DG scheme on the same two examples from previous chapter, a meniscus lens and a dielectric total internal reflection concentrator. The hybrid scheme is compared to the pure ADER-DG scheme. To solve Liouville's equation we fix a few parameters for the problems. Namely, in the Taylor expansions (5.28) and (5.32) we take $M = N$, so that both schemes have a formal $(N+1)$th order accuracy in space and z. The CFL condition (6.27) is applied with CFL $= 0.9$ fixed, and the influence of neighbouring elements on the CFL condition is either local as in the hybrid SLDG and ADER-DG scheme or completely global as in the pure ADER-DG scheme. In the mesh refinement procedure we take $\alpha = 2.25$. Both schemes were implemented in C++, with the implementation details discussed in the Appendix D. All the computations were performed using a single core of a laptop which has an Intel Core i7–11800H CPU @ 2.30GHz.

6.5.1 Meniscus lens

We consider the meniscus lens example detailed in Section 5.7.1. For this example the refractive index field is constant between $z = 0$ and $z = z_1 = 1.3$ (front of the first circle), and between $z = z_2$ and $z = 4$ (at the end). In these regions the light rays are straight lines and, therefore, the SLDG scheme can be easily applied over the entire phase space mesh to take one big step. Hence, we step directly from $z = 0$ to $z = z_1$ and from $z = z_2$ to $z = 4$ with the SLDG scheme. In the region with optical interfaces, we employ both ADER-DG and SLDG elements with local time stepping.

To show the effects of the lens we compute a numerical solution. At $z = 0$



(a) $z = 0$. **(b)** $z = 4$.

Figure 6.5: Distributions of ρ for the meniscus lens with Gaussian initial condition computed with the $N = 7$ hybrid SLDG and ADER-DG scheme.

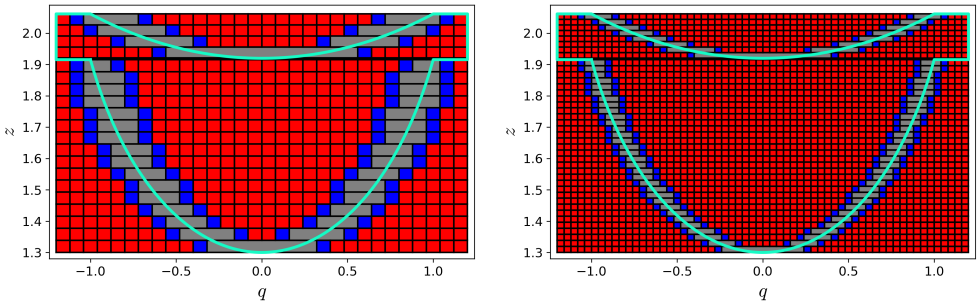
we start with a Gaussian distribution, given by

$$\rho_0(q,p) = \exp\left(-\frac{q^2}{2\sigma_q^2}\right)\exp\left(-\frac{p^2}{2\sigma_p^2}\right), \tag{6.35}$$

where we take $\sigma_q = 0.5$ and $\sigma_p = 0.08$. We limit the maximum momentum since the velocity $\boldsymbol{u}$, cf. (6.2b), blows up as $|p|$ approaches n. The maximum momentum is taken to be $0.9n(z,q)$. Furthermore, we choose mesh spacings $\Delta q_{\max} = 0.16$, $\Delta q_{\mathrm{SL}} = 0.08$ and $\Delta p = 0.1$, and use $N = 7$. Note that $\Delta q_{\min} = \Delta q_{\max}/\alpha$. Initially at $z = 0$ the phase space mesh has 540 elements and at the end, at $z = 4$, the mesh contains 450 elements. The initial condition and the numerical solution are shown in Figure 6.5. From the figure, we observe that the initial condition has been compressed in the q-direction and expanded in the p-direction. Moreover, one can see values below 0 on the target distribution which is due to a cut-off of the initial distribution. The cut-off generates a discontinuity in the distribution, which appears as an oscillation resulting in undershoot in the numerical solution.

At each common integration level, which are the z-levels for SLDG elements, the total luminous flux, including the fluxes leaving the system through physical boundaries except for the optical interface, was computed. This total luminous flux should remain constant if the scheme is energy-conserving. The maximum absolute relative deviation was $1.55 \cdot 10^{-15}$, hence, the scheme is energy-conserving up to machine precision. Consequently, the local time stepping procedure explained in Section 6.4 is indeed energy conservative.

The qz-mesh of the hybrid SLDG and ADER-DG scheme in a part of the region with optical interfaces is shown in Figure 6.6. Here, we use the same



(a) Coarse mesh. **(b)** Fine mesh.

Figure 6.6: The qz-mesh for the meniscus lens that is used in the hybrid SLDG and ADER-DG scheme. Mesh size parameters for coarse mesh are $\Delta q_{\max} = 0.16$ and $\Delta q_{\mathrm{SL}} = 0.08$, for the finer mesh these values are halved.

color coding of elements in qz-space as before, i.e., red denotes SLDG elements, blue denotes ADER-DG elements that couple to an SLDG element and the green curve represents the meniscus lens. For the blue and gray ADER-DG elements we omit the sub-steps. Moreover, the gray elements of the mesh are combined into blocks because the mesh can change at each sub-step. The figure on the right is generated with a mesh where the mesh size parameters have been halved compared to the coarse mesh values $\Delta q_{\max} = 0.16$ and $\Delta q_{\mathrm{SL}} = 0.08$. Furthermore, one can see that the size of the gray region shrinks upon halving the mesh size parameters.

Next, we perform a convergence study for the meniscus lens. The initial condition we use reads

$$\rho_0(q,p) = \varphi_{m,k}\left(\frac{q}{\lambda_q}\right)\varphi_{m,k}\left(\frac{p}{\lambda_p}\right), \tag{6.36}$$

with parameters $m = 10$, $k = 2$, $\lambda_q = 0.5$ and $\lambda_p = 0.25$. The function $\varphi_{m,k}$ is given by (4.66). With the chosen initial condition, the exact solution at $z = 4$ can be obtained by tracing light rays backwards through the circle segments of the lens, i.e., we apply the method of characteristics. The convergence is studied on a sequence of meshes that have mesh size parameters chosen as

$$\Delta q_{r,\max} = 2^{-r}\Delta q_{0,\max},\ \Delta q_{r,\mathrm{SL}} = 2^{-r}\Delta q_{0,\mathrm{SL}} \text{ and } \Delta p_r = 2^{-r}\Delta p_0, \tag{6.37}$$

where r denotes the refinement level and we choose $\Delta q_{0,\max} = 0.2$, $\Delta q_{0,\mathrm{SL}} = 0.1$ and $\Delta p_0 = 0.1$. The convergence results for the L_2 and L_∞ norms are computed for both the pure ADER-DG and the hybrid SLDG and ADER-DG schemes and are listed in Tables 6.1-6.2, where the convergence rate is measured as $\log_2(e_{r-1}/e_r)$ with e_r the error for refinement level r. For both schemes the

listed errors are comparable and the computed orders of convergence are in good agreement with the expected $N+1$ order of convergence.

In Table 6.2 the last two columns denote the CPU time t_{CPU} for the hybrid scheme and the speed-up of the hybrid scheme relative to the pure ADER-DG scheme. From the speed-up column we see that the hybrid SLDG and ADER-DG scheme is significantly faster than the pure ADER-DG scheme, ranging from being $1.6-3$ times faster on coarse meshes to achieving a speed-up of $10-20$ on finer meshes. The speed-up increases as r is increased; this is attributed to two different effects.

The first effect is due to the hybrid scheme taking two big steps to go from $z=0$ to $z=z_1$, and from $z=z_2$ to $z=4$, independent of the refinement level. The ADER-DG scheme instead has to obey the CFL condition (6.27) so that the number of steps in those regions increases when the refinement level increases.

The second effect is explained by the fact that the CPU time of the hybrid SLDG and ADER-DG scheme do not scale cubically with one over the mesh spacing. For the pure ADER-DG scheme one expects the CPU time to satisfy a cubic relation as for each refinement level we (approximately) double the amount of elements per direction and by virtue of the CFL condition (6.27) the stepsizes are halved as well, so that in total the amount of work increases roughly by a factor 8. In contrast, the hybrid SLDG and ADER-DG scheme shows initially a quadratic scaling in the CPU time for coarse meshes as one can observe from Table 6.2. This is observed because initially the computation time in the gray region shown in Figure 6.6 dominates the CPU time. Again, upon halving the mesh spacings Δq and Δp we must also halve the stepsize Δz for ADER-DG elements by virtue of the CFL condition (6.27). Additionally, the halving of the mesh spacing Δq also leads to a decrease of the area of the gray region. Therefore, roughly the gray region features a doubling in the number of elements and we need to take twice as many steps, so that the workload increases by a factor 4. Of course, at some point the cost of the SLDG elements becomes dominant again so that we should roughly observe a factor 8 increase in CPU time upon halving the mesh size parameters.

Next, we compare the DG schemes to quasi-Monte Carlo (QMC) ray tracing for computing the illuminance. The same setup as used in Section 5.7.1 is used. For both methods we compute the L_∞-norm of the average illuminance. The initial condition (6.36) is used, so that an exact solution to Liouville's equation is available which leads to an exact illuminance.

The performance of QMC ray tracing compared to the DG schemes for Liouville's equation is shown in Figure 6.7. For QMC ray tracing we employ $B=200$ bins and each subsequent point in the Figure quadruples the number

Table 6.1: Convergence data for the meniscus lens example with the pure ADER-DG scheme.

r	L_2	$\mathcal{O}(L_2)$	L_∞	$\mathcal{O}(L_\infty)$
Pure ADER-DG				
$N = 3$				
0	4.41e-03		4.93e-02	
1	3.60e-04	3.61	6.24e-03	2.98
2	2.44e-05	3.88	4.40e-04	3.83
3	1.56e-06	3.97	2.91e-05	3.92
4	9.77e-08	3.99	1.88e-06	3.95
$N = 4$				
0	1.48e-03		1.89e-02	
1	6.77e-05	4.45	1.25e-03	3.92
2	2.37e-06	4.84	4.66e-05	4.75
3	7.60e-08	4.96	1.58e-06	4.88
4	2.38e-09	4.99	5.05e-08	4.97
$N = 5$				
0	5.08e-04		9.70e-03	
1	1.25e-05	5.34	2.79e-04	5.12
2	2.27e-07	5.78	5.45e-06	5.68
3	3.70e-09	5.94	9.26e-08	5.88
4	5.85e-11	5.98	1.48e-09	5.96
$N = 6$				
0	1.76e-04		3.61e-03	
1	2.33e-06	6.24	5.91e-05	5.93
2	2.17e-08	6.74	6.14e-07	6.59
3	1.80e-10	6.92	5.08e-09	6.92
$N = 7$				
0	6.26e-05		1.38e-03	
1	4.36e-07	7.16	1.28e-05	6.76
2	2.10e-09	7.70	6.28e-08	7.67
3	8.98e-12	7.87	2.73e-10	7.85

Table 6.2: Convergence data for the meniscus lens example with the hybrid SLDG and ADER-DG with local time stepping (LTS). The last column denotes the speed-up relative to the pure ADER-DG scheme.

r	L_2	$\mathcal{O}(L_2)$	L_∞	$\mathcal{O}(L_\infty)$	t_{CPU} [s]	speed-up
Hybrid SLDG and ADER-DG with LTS						
			$N = 3$			
0	4.43e-03		5.08e-02		0.226	1.85
1	3.63e-04	3.61	6.58e-03	2.95	1.075	2.81
2	2.46e-05	3.88	4.62e-04	3.83	6.549	4.33
3	1.57e-06	3.97	3.04e-05	3.92	45.341	7.88
4	9.81e-08	4.00	1.92e-06	3.98	399.003	12.66
			$N = 4$			
0	1.49e-03		1.97e-02		0.457	1.80
1	6.79e-05	4.45	1.25e-03	3.98	2.054	2.64
2	2.37e-06	4.84	4.66e-05	4.75	10.989	4.25
3	7.64e-08	4.96	1.62e-06	4.85	67.644	8.79
4	2.40e-09	4.99	5.19e-08	4.96	522.316	16.91
			$N = 5$			
0	5.08e-04		9.64e-03		0.790	1.80
1	1.25e-05	5.34	2.82e-04	5.09	3.376	2.62
2	2.27e-07	5.78	5.43e-06	5.70	16.514	4.40
3	3.71e-09	5.94	9.18e-08	5.89	101.539	8.97
4	5.89e-11	5.98	1.50e-09	5.93	712.916	18.77
			$N = 6$			
0	1.76e-04		3.59e-03		1.502	1.68
1	2.33e-06	6.24	5.89e-05	5.93	6.202	2.41
2	2.18e-08	6.74	6.21e-07	6.57	28.343	3.98
3	1.80e-10	6.92	5.19e-09	6.90	144.794	10.02
			$N = 7$			
0	6.06e-05		1.38e-03		2.557	1.62
1	4.36e-07	7.12	1.28e-05	6.75	10.082	2.31
2	2.11e-09	7.69	6.36e-08	7.65	44.870	3.77
3	9.07e-12	7.86	2.73e-10	7.86	219.777	9.79

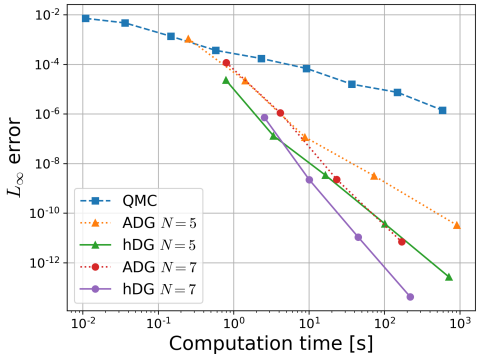


Figure 6.7: Comparison between quasi-Monte Carlo (QMC) ray tracing, the pure ADER-DG scheme (ADG), and the hybrid SLDG and ADER-DG scheme (hDG) for the meniscus lens.

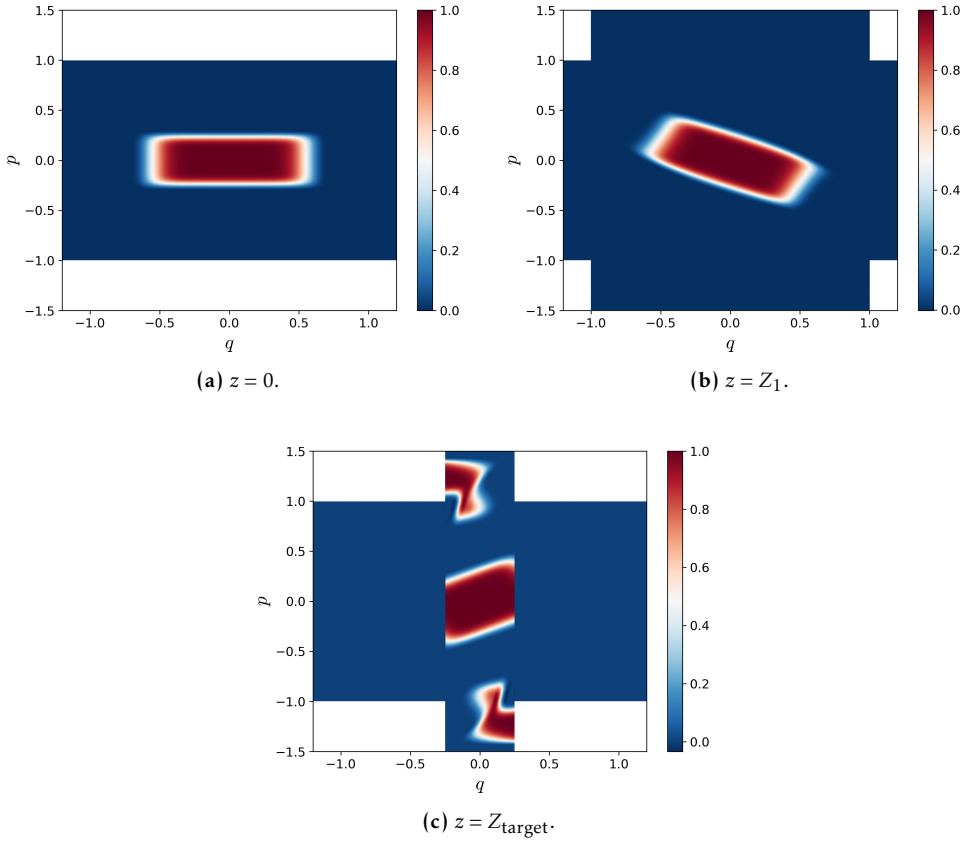
of rays compared to the previous one. Initially the number of rays is $N_{\mathrm{RT}} = 31250$ and the last point corresponds to $N_{\mathrm{RT}} = 2.048 \cdot 10^9$ rays. For the DG schemes we use the previously mentioned sequence of meshes whereby the mesh size parameters are halved upon refinement, as described by (6.37). Here, the first data point for the ADER-DG scheme corresponds to $r = -1$, whereas for the hybrid scheme the first point corresponds to $r = 0$. Clearly, for this example the hybrid SLDG and ADER-DG scheme can achieve higher accuracy than QMC ray tracing in equal computation time. For instance, a computation time of about 10 seconds leads to an increase in accuracy by more than a factor 10,000 when comparing the hybrid scheme to QMC ray tracing. Moreover, the hybrid SLDG and ADER-DG scheme converges faster to high accuracies than the pure ADER-DG scheme and much faster than QMC ray tracing.

6.5.2 Dielectric TIR concentrator

As a second example we consider the dielectric TIR concentrator (DTIRC) that is detailed in Section 5.7.2. For the initial condition we employ

$$\rho_0(q,p) = \varphi_{m,k}\left(\frac{q}{\lambda_q}\right)\varphi_{m,k}\left(\frac{p}{\lambda_p}\right), \tag{6.38}$$

with parameters $m = 10$, $k = 4$, $\lambda_q = 0.8$ and $\lambda_p = \sin(20\,\mathrm{deg})$. The maximum momentum is limited to $\sin(85\,\mathrm{deg})\, n(z,q)$. The hybrid SLDG and ADER-DG solver is used to compute numerical solutions, with parameters $N = 3$, $\Delta q_{\max} = 0.1$, $\Delta q_{\mathrm{SL}} = 0.05$ and $\Delta p \approx 0.052$. The resulting distributions at $z = 0$, $z = Z_1$ and $z = Z_{\mathrm{target}}$ are shown in Figure 6.8. At $z = Z_1$ all initial light has



(a) $z = 0$.

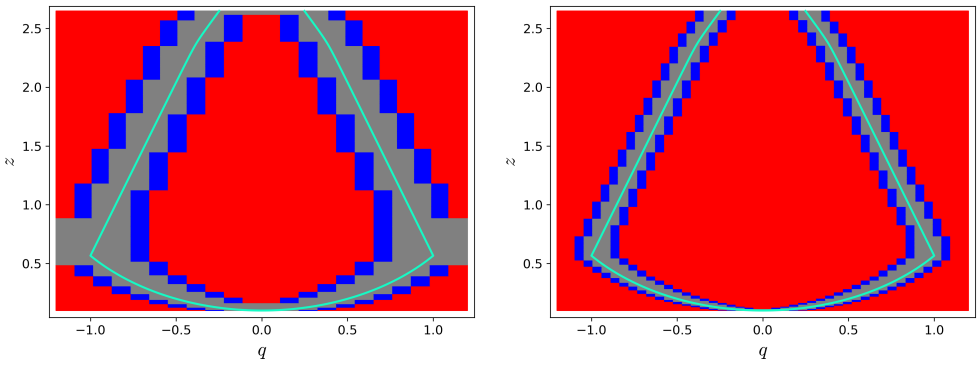
(b) $z = Z_1$.

(c) $z = Z_{\text{target}}$.

Figure 6.8: Distributions of ρ for the DTIRC computed with the $N = 3$ hybrid SLDG and ADER-DG scheme.

been refracted into the dielectric medium and at $z = Z_{\text{target}}$ one can see that a part of the distribution has reflected at the side walls resulting in the bottom and top patches. Light is also fully contained within the dielectric medium.

The qz-mesh is shown in the right panel of Figure 6.9 for a fine mesh, that corresponds to the chosen mesh size parameters. In the left panel of Figure 6.9 we consider a coarse mesh, where the mesh size parameters have been doubled compared to the fine mesh. The gridlines in the qz-mesh are not shown in Figure 6.9. The initial step (not shown in the figure) from $z = 0$ to $z = z_c - R = 0.1$ consists of a single step with the SLDG scheme. Note that for the coarse mesh close to $z \approx Z_1$ we entirely use ADER-DG elements in the background medium n_0. The reason for this is that there is not sufficient space to fit twice the mesh spacing Δq_{SL} and also allow enough space for gray ADER-DG elements for the purpose of merging elements.



(a) Coarse mesh. **(b)** Fine mesh.

Figure 6.9: The qz-mesh for the DTIRC that is used in the hybrid SLDG and ADER-DG scheme. Mesh size parameters for coarse mesh are $\Delta q_{\max} = 0.2$ and $\Delta q_{\mathrm{SL}} = 0.1$, whereas for the finer mesh these are $\Delta q_{\max} = 0.1$ and $\Delta q_{\mathrm{SL}} = 0.05$.

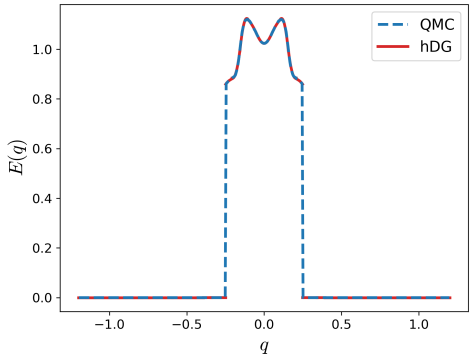


Figure 6.10: Illuminance at $z = Z_{\mathrm{target}}$ for the DTIRC computed with quasi-Monte Carlo ray tracing (QMC) on $B = 400$ bins and the $N = 3$ hybrid SLDG and ADER-DG (hDG) scheme.

Next, QMC ray tracing and the DG solvers are compared for computing the illuminance. In QMC ray tracing we cannot use a uniform grid on $q \in [-1.2, 1.2]$ as this would cause the two bins, which cut the side wall, to have two refractive indices. Hence, we modify the grid to be piecewise uniform, with uniform grid distributions on the q-intervals $[-1.2,\ -Q_{\mathrm{top}}(Z_{\mathrm{target}})]$, $[-Q_{\mathrm{top}}(Z_{\mathrm{target}}),\ Q_{\mathrm{top}}(Z_{\mathrm{target}})]$ and $[Q_{\mathrm{top}}(Z_{\mathrm{target}}),\ 1.2]$. In QMC ray tracing the intersections with the circle segment are exactly computed, whereas for the intersection with the side walls we employ a Newton method that resorts to bisection when necessary.

In Figure 6.10 we show the illuminance computed from the basic luminance profile in the last panel of Figure 6.8, alongside a QMC ray tracing

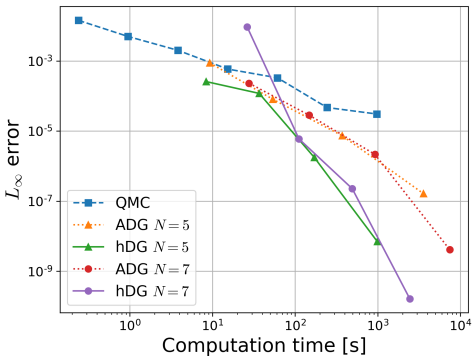


Figure 6.11: Comparison between quasi-Monte Carlo (QMC) ray tracing, the pure ADER-DG scheme (ADG) and the hybrid SLDG and ADER-DG (hDG) scheme for the dielectric TIR concentrator.

solution for which we used $B = 400$ bins and traced $N_{\text{RT}} = 8 \cdot 10^6$ rays. The illuminance profiles of both methods are almost indistinguishable by eye, showing the good agreement in the profile between the two methods.

Next, we will compare the performance of the DG schemes with QMC ray tracing by computing the error in the illuminance profile. Once more, the illuminance is averaged for the DG solution and for the error we take the L_∞-norm between a numerical solution and a reference solution. As a reference solution we use the illuminance computed with the hybrid SLDG and ADER-DG scheme on a very fine grid with parameters $N = 7$, $\Delta q_{\max} = 1.25 \cdot 10^{-2}$, $\Delta q_{\text{SL}} = 6.25 \cdot 10^{-3}$ and $\Delta p \approx 6.5 \cdot 10^{-3}$.

The comparison between the DG schemes and QMC ray tracing is shown in Figure 6.11. The DG schemes compute numerical solutions on a sequence of meshes for which the mesh size parameters satisfy (6.37) with $r = 0, 1, 2, 3$. The mesh size parameters for $r = 0$ are given by $\Delta q_{0,\max} = 0.2$, $\Delta q_{0,\text{SL}} = 0.1$ and $\Delta p_0 = 0.1$. For QMC ray tracing the first data point corresponds to $N_{\text{RT}} = 5 \cdot 10^5$ and each subsequent dot represents a quadrupling in the number of rays, and consequently the last data point corresponds to $N_{\text{RT}} = 2.048 \cdot 10^9$ rays. From the figure one can observe that both the pure ADER-DG and the hybrid scheme can reach higher accuracy in less computation time than QMC ray tracing. Once more, we see that the hybrid SLDG and ADER-DG scheme converges faster to high accuracies than the pure ADER-DG scheme, and much faster than QMC ray tracing.

6.6 Concluding remarks

A novel hybrid scheme, combining ADER-DG elements on a moving mesh, SLDG elements and local time stepping, has been presented. The scheme yields improved performance over the pure ADER-DG scheme by employing the very efficient SLDG elements when possible. The SLDG scheme allows large steps to be taken in regions without optical interfaces. Local time stepping severely diminishes the effect of stepsize reduction, which is caused by small elements or a large mesh velocity. These building blocks led to an improved solver for Liouville's equation for piecewise constant refractive index fields.

Numerical experiments indicate the increased performance of the hybrid scheme over the pure ADER-DG scheme, whilst exhibiting the expected $N+1$ order of convergence for sufficiently smooth solutions. In the meniscus lens example we saw that the hybrid scheme is faster by a factor of roughly 1.6 to 10 for computation times up to 4 minutes. The increased performance also allows faster convergence to high accuracies compared to the pure ADER-DG scheme. Moreover, in the shown examples the hybrid SLDG and ADER-DG scheme outperforms QMC ray tracing by reaching higher accuracies in equal computation time. In particular, for the meniscus lens example the hybrid scheme computed a more than $10,000$ more accurate solution that QMC ray tracing within a computation time of 10 seconds.

Local time stepping has been used to allow efficient computation in the presence of small elements or large mesh velocities. Another use would be to introduce local stepsizes that depend on the momentum values an element describes. This can be beneficial as the velocity field given by (6.2b) rapidly increases for large absolute momentum values approaching n. On the other hand, this would severely complicate the scheme near the optical interface because phase space is in contact with each other at completely different momentum values due to the law of specular reflection and Snell's law of refraction. Moreover, the optical system restricts the stepsize that can be taken for SLDG elements by the condition that characteristics are not allowed to cross optical interfaces which is momentum dependent.

So far, we have assumed in the model that light can either be fully reflected or fully refracted. An alternative model would be to replace the jump condition (2.48) with the jump condition (2.54) that models Fresnel reflections, where light can be partially reflected. In the design of the hybrid scheme, the SLDG elements are chosen in such a way that characteristics are not allowed to cross optical interfaces. Therefore, modelling Fresnel reflections only impacts the ADER-DG elements that touch the optical interface. Including

Fresnel reflections in the DG schemes is the topic of the next chapter. Finally, the hybrid SLDG and ADER-DG scheme can be extended to deal with three-dimensional optics, where at each fixed z-value phase space is four-dimensional. The main building blocks of the scheme can be relatively easily extended to a four-dimensional phase space. The largest difficulty lies in the fact that aligning the mesh with curved optical interfaces requires the use of curvilinear elements, while SLDG schemes for arbitrary element shapes can be very difficult, if not impossible, to implement in an efficient manner. Hence, a key design principle would be to dynamically adapt the mesh, away from optical interfaces, such that the two-dimensional position space is covered by square elements with local uniform mesh spacing.

Chapter 7

Incorporating Fresnel reflections

Fresnel reflections describe a partial reflection and partial transmission of light striking an optical interface. Recall that Fresnel reflections are included in the jump condition for the basic luminance by combining two incident light rays, with momentum vectors $\vec{i}_\mathrm{r}$ and $\vec{i}_\mathrm{t}$, to one outgoing direction $\vec{p}$ as follows

$$\rho^+(\vec{p}) = \mathcal{R}(\vec{i}_\mathrm{r}; n_1, n_0, \vec{\nu})\rho^-(\vec{i}_\mathrm{r}) + \Big(1 - \mathcal{R}(\vec{i}_\mathrm{t}; n_0, n_1, -\vec{\nu})\Big)\rho^-(\vec{i}_\mathrm{t}), \tag{7.1a}$$

with

$$\vec{i}_\mathrm{r} = \mathcal{S}_\mathrm{R}^{-1}(\vec{p}; n_1, n_0, \vec{\nu}) \quad \text{and} \quad \vec{i}_\mathrm{t} = \mathcal{S}_\mathrm{T}^{-1}(\vec{p}; n_0, n_1, -\vec{\nu}), \tag{7.1b}$$

where $\mathcal{R}$ denotes the Fresnel reflection coefficient given by (2.53). The relation between $\vec{i}_\mathrm{r}$, $\vec{i}_\mathrm{t}$ and $\vec{p}$ is sketched in black in Figure 7.1. It can also happen that light is reflected according to total internal reflection, i.e., so that $\mathcal{R} = 1$. This situation is sketched in red in Figure 7.1.

The Fresnel reflection coefficients $\mathcal{R}$ are sketched in Figure 7.2. In the figure one can see the discontinuous change in the derivative of $\mathcal{R}$ for $n_0 > n_1$ at the critical momentum (angle). Dealing with Fresnel reflections at an optical interface requires a modification of the discretisation at an optical interface. By employing a newly derived energy balance we can ensure energy conservation. Special care must be taken close to the critical momentum, as the behaviour goes from partial reflection to full reflection.

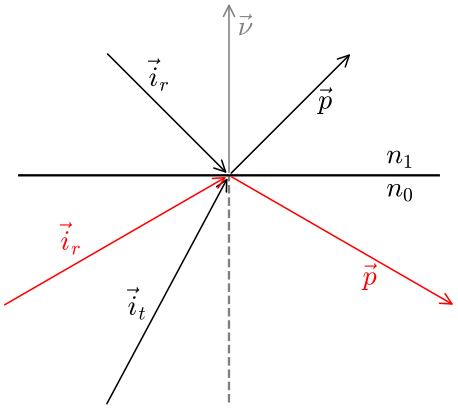
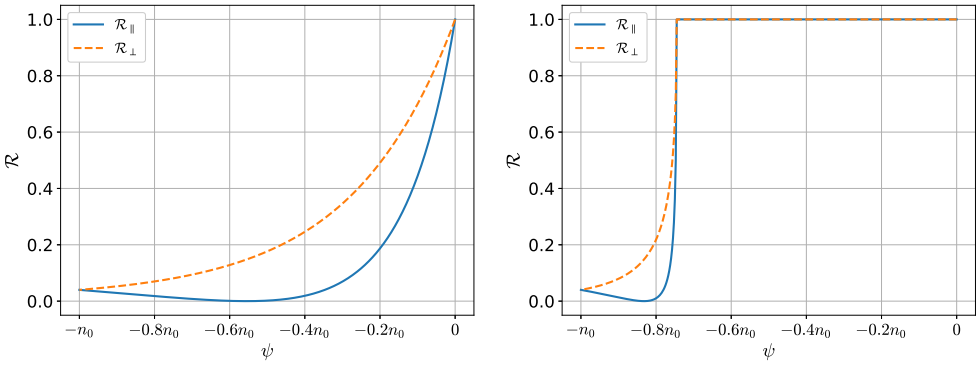


Figure 7.1: Incident light rays with momenta $\vec{i}_r$ and $\vec{i}_t$ have after reflection and refraction, respectively, the momentum $\vec{p}$. Black represents Fresnel reflection and red represents total internal reflection ($\mathcal{R} = 1$).



(a) $n_0 = 1$ and $n_1 = 1.5$.

(b) $n_0 = 1.5$ and $n_0 = 1$.

Figure 7.2: The Fresnel reflection coefficients as a function of $\psi = \vec{i} \cdot \vec{\nu}$.

7.1 Energy balance for Fresnel reflections

In Section 5.5 we derived an energy balance for the jump condition $\rho^+(\vec{p}) = \rho^-(\vec{i})$ with $\vec{p} = \mathcal{S}(\vec{i})$. We derived expressions for the incident light corresponding to a given momentum interval for outgoing light. In this section we let $\mathcal{I}_\mathrm{R}(R;\sigma_\mathrm{inc},\sigma)$ denote the incident light that is propagating in the direction σ_inc and that after reflection corresponds to a momentum interval R where light is propagating in the direction σ. The definition for $\mathcal{I}_\mathrm{R}$ is given by (5.50). Similarly, $\mathcal{I}_\mathrm{T}(R;\sigma_\mathrm{inc},\sigma)$ denotes the incident light that after transmission ends up at R.

For the reflection coefficient $\mathcal{R}$ as it appears in the jump condition (7.1) the short-hand notation $\mathcal{R}(p)$ will be used in what follows. The energy balance at an optical interface with jump condition (7.1) reads

$$\begin{aligned}\int_R \rho_\sigma \left(u_0 - \frac{\mathrm{d}Q}{\mathrm{d}z}\right)\Big|_+ \mathrm{d}p = \sum_{\sigma_\mathrm{inc}\in\{\mathrm{b},\mathrm{f}\}} \int_{\mathcal{I}_\mathrm{R}(R;\sigma_\mathrm{inc},\sigma)} \mathcal{R}\rho_{\sigma_\mathrm{inc}} \left(u_0 - \frac{\mathrm{d}Q}{\mathrm{d}z}\right)\Big|_- \mathrm{d}p \\ + \int_{\mathcal{I}_\mathrm{T}(R;\sigma_\mathrm{inc},\sigma)} (1-\mathcal{R})\,\rho_{\sigma_\mathrm{inc}} \left(u_0 - \frac{\mathrm{d}Q}{\mathrm{d}z}\right)\Big|_- \mathrm{d}p,\end{aligned} \tag{7.2}$$

where we can see the contributions due to partial reflection and transmission. A proof of the energy balance (7.2) can be found in Appendix C.1.

At this point it is important to remark that if the energy balance (7.2) is satisfied, then the fluxes across the optical interface are energy-conserving. That is, the flux on the incident side is equal to the flux on the outgoing side. This can be seen, by considering the energy balance (7.2) over all outgoing light, i.e., the sum of ρ_f and ρ_b for all outgoing directions. Consequently, for the integrals on the right-hand side of (7.2) we have to integrate over all incident light. The incident light $\mathcal{I}_\mathrm{R}$ contains both light for total internal reflection ($\mathcal{R} = 1$) and for partial reflection ($0 \leq \mathcal{R} < 1$), whereas $\mathcal{I}_\mathrm{T}$ contains only light for partial transmission ($0 \leq \mathcal{R} < 1$). The integration domains for partial reflection and transmission, match so that the coefficients $\mathcal{R}$ and $1-\mathcal{R}$ in the first and second integral sum to 1. As a result, the right-hand side of (7.2) describes the flux on the incident side of the optical interface.

7.2 Discretisation at optical interface

The least-squares procedure with energy constraint of Section 5.5 needs to be modified to include Fresnel reflections. We split the jump condition into two contributions $\rho = \rho_\mathrm{R} + \rho_\mathrm{T}$, that describe reflection and transmission. These

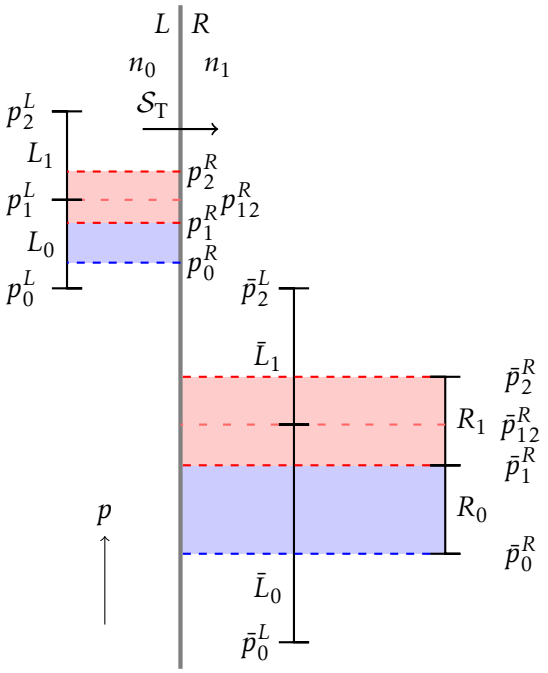


Figure 7.3: Connectivity of faces.

contributions are given by

$$\rho_\mathrm{R}^+(\vec{p}) = \mathcal{R}(\vec{i}_\mathrm{r}; n_1, n_0, \vec{\nu})\rho^-(\vec{i}_\mathrm{r}), \tag{7.3a}$$

$$\rho_\mathrm{T}^+(\vec{p}) = \Big(1 - \mathcal{R}(\vec{i}_\mathrm{t}; n_0, n_1, -\vec{\nu})\Big)\rho^-(\vec{i}_\mathrm{t}), \tag{7.3b}$$

respectively.

At a fixed point (z, q) on the optical interface the numerical solution on each face is written as a polynomial in the momentum p in the DG methods which were covered in previous chapters. The geometric connectivity of the faces at an optical interface becomes slightly more difficult as an incident momentum interval results into two contributions via reflection and transmission.

Considering the geometry in Figure 7.3 we need to describe how we compute a polynomial $\rho_\mathrm{T}^{R_i} \in \mathbb{P}_N$ for partial transmission for each face R_i. For the face R_1 the polynomial $\rho_\mathrm{T}^{R_1}$ must be computed from a piecewise polynomial ρ^L with an energy-conservation constraint as described by the energy balance. Because we consider only transmission, only one integral for each incident propagation direction remains on the right-hand side of the energy balance (7.2). For ease of presentation, assume that the incident light and outgoing light are both forward propagating, so that $\mathcal{I}_\mathrm{T}(R_i; \mathrm{b}, \mathrm{f}) = \emptyset$ and only

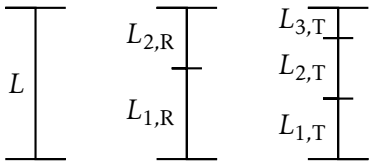


Figure 7.4: Partitioning of an incident momentum interval L into integration intervals $L_{i,\mathrm{R}}$ and $L_{i,\mathrm{T}}$.

one integral remains in the energy balance. For the polynomial $\rho_{\mathrm{T}}^{R_1}$ we pose the following constrained least-squares problem

$$\min_{\rho_{\mathrm{T}}^{R_1} \in \mathbb{P}_N} \int_{\bar{p}_1^R}^{\bar{p}_2^R} \left[\rho_{\mathrm{T}}^{R_1}(\bar{p}) - \left(1 - \mathcal{R}(S_{\mathrm{T}}^{-1}(\bar{p}))\right)\rho^L(S_{\mathrm{T}}^{-1}(\bar{p}))\right]^2 \mathrm{d}\bar{p}, \tag{7.4a}$$

$$\text{subject to} \int_{\bar{p}_1^R}^{\bar{p}_2^R} F_{\mathrm{T}}^{R_1}(\bar{p})\,\mathrm{d}\bar{p} = \int_{p_1^R}^{p_2^R} (1 - \mathcal{R}(p))\,F^L(p)\,\mathrm{d}p, \tag{7.4b}$$

where $p_1^R = S_{\mathrm{T}}^{-1}(\bar{p}_1^R)$, etc., and ρ^L and F^L denote piecewise polynomials given by

$$\rho^L(p) = \begin{cases} \rho^{L_0}(p) & \text{if } p \in L_0, \\ \rho^{L_1}(p) & \text{if } p \in L_1, \end{cases} \quad \text{and} \quad F^L(p) = \begin{cases} F^{L_0}(p) & \text{if } p \in L_0, \\ F^{L_1}(p) & \text{if } p \in L_1. \end{cases}$$

The numerical flux $F_{\mathrm{T}}^{R_i}$ for an arbitrary face R_i is written in a basis of Lagrange polynomials, i.e.,

$$F_{\mathrm{T}}^{R_i}(p) = \sum_{j=0}^{N} \rho_{\mathrm{T},j}^{R_i} a_j \ell_j(\eta(p)) \quad \text{with} \quad a_j = u_0 - \left.\frac{\mathrm{d}Q}{\mathrm{d}z}\right|_{(+,p_j)}, \tag{7.5}$$

where $\eta(p)$ denotes a transformation from the face R_i to the reference interval $[0,1]$, and $\{p_j\}_{j=0}^N$ denote the Gauss-Legendre quadrature points on the interval R_i. The coefficients $\rho_{\mathrm{T},j}^{R_i}$ are the expansion coefficients of the polynomial $\rho_{\mathrm{T}}^{R_i}$ to be determined.

Since we now have Fresnel reflections, the geometric connectivity of a single incident interval L depends on whether we consider reflection or transmission. The geometric connectivity of the incident interval L has an impact on how numerical integration is performed, as we split the integration interval at points of discontinuities. An example of how L must be split is sketched in Figure 7.4. The incident flux over the interval L can be separated into

contributions of each part of L as follows

$$\begin{aligned}\int_L F(p)\,\mathrm{d}p &= \int_{L_{1,\mathrm{T}}\cup L_{2,\mathrm{T}}\cup L_{3,\mathrm{T}}} (1-\mathcal{R}(p))\,F(p)\,\mathrm{d}p + \int_{L_{1,\mathrm{R}}\cup L_{2,\mathrm{R}}} \mathcal{R}(p)F(p)\,\mathrm{d}p \\ &= \sum_{n=1}^{N_\mathrm{T}} \int_{L_{n,\mathrm{T}}} (1-\mathcal{R}(p))\,F(p)\,\mathrm{d}p + \sum_{n=1}^{N_\mathrm{R}} \int_{L_{n,\mathrm{R}}} \mathcal{R}(p)F(p)\,\mathrm{d}p,\end{aligned} \tag{7.6}$$

with $F(p)$ denoting the numerical flux expanded into a basis of Lagrange polynomials, cf. (7.5), and N_T and N_R denote the number of integration intervals for the interval L for transmission and reflection, respectively. Each part on the right-hand side of (7.6) represents a (partial) contribution for the incident side of the constraint in any least-squares problem, e.g., the right-hand side of (7.4b). In the solution procedure of the constrained least-squares problem we apply $(N+1)$-point Gauss-Legendre quadrature to these integrals. Applying the Gauss-Legendre quadrature to both sides of relation (7.6) leads to

$$\int_{L,N} F(p)\,\mathrm{d}p \neq \sum_{n=1}^{N_\mathrm{T}} \int_{L_{n,\mathrm{T}},N} (1-\mathcal{R}(p))\,F(p)\,\mathrm{d}p + \sum_{n=1}^{N_\mathrm{R}} \int_{L_{n,\mathrm{R}},N} \mathcal{R}(p)F(p)\,\mathrm{d}p, \tag{7.7}$$

where the quadrature notation from Section 6.3.1 is used. Consequently, the additive property of integrals is lost after applying Gauss-Legendre quadrature. This is because the quadrature nodes for $L_{n,\mathrm{T}}$ and $L_{n,\mathrm{R}}$ do not match, and because the integrals are not exactly evaluated since $\mathcal{R}(p)$ is a non-linear function in p.

The issue is resolved by replacing $\mathcal{R}F$ with an interpolant through Gauss-Legendre quadrature nodes on the incident side. Specifically, for every face L_i describing incident light we will replace $\mathcal{R}F^{L_i}$ with $[\mathcal{R}F]_\mathrm{h}^{L_i}$ that reads

$$\mathcal{R}F^{L_i} \approx [\mathcal{R}F]_\mathrm{h}^{L_i}(p) = \sum_{j=0}^{N} \mathcal{R}_j \rho_j^{L_i} a_j \ell_j(\eta(p)), \tag{7.8a}$$

and replace $(1-\mathcal{R})F^{L_i}$ with $[(1-\mathcal{R})F]_\mathrm{h}^{L_i}$ that reads

$$(1-\mathcal{R})F^{L_i} \approx [(1-\mathcal{R})F]_\mathrm{h}^{L_i}(p) = \sum_{j=0}^{N} (1-\mathcal{R}_j) \rho_j^{L_i} a_j \ell_j(\eta(p)), \tag{7.8b}$$

with a_j and $\mathcal{R}_j$ given by

$$a_j = u_0 - \left.\frac{\mathrm{d}Q}{\mathrm{d}z}\right|_{(-,p_j)} \text{ and } \mathcal{R}_j = \mathcal{R}(p_j), \tag{7.8c}$$

and p_j denotes the jth Gauss-Legendre quadrature node on the face L_i and $\eta(p)$ describes an affine linear transformation from $p \in L_i$ to the reference interval $[0,1]$. Replacing the appropriate terms in relation (7.6) with (7.8) and applying Gauss-Legendre quadrature leads to

$$\int_{L,N} F(p)\mathrm{d}p = \sum_{n=1}^{N_\mathrm{T}} \int_{L_{n,\mathrm{T}},N} [(1-\mathcal{R})F]_\mathrm{h}\,\mathrm{d}p + \sum_{n=1}^{N_\mathrm{R}} \int_{L_{n,\mathrm{R}},N} [\mathcal{R}F]_\mathrm{h}\,\mathrm{d}p, \tag{7.9}$$

as the integrals are now exactly evaluated by the $(N+1)$-point Gauss-Legendre quadrature.

With the appropriate terms on the incident side replaced by (7.8) we can now formulate the constrained least-squares problem that results in an energy-conserving discretisation. For the polynomial $\rho_\mathrm{T}^{R_1}$ we solve

$$\min_{\rho_\mathrm{T}^{R_1}\in\mathbb{P}_N} \int_{\bar{p}_1^R}^{\bar{p}_2^R} \left[\rho_\mathrm{T}^{R_1}(\bar{p}) - \left(1-\mathcal{R}(S_\mathrm{T}^{-1}(\bar{p}))\right)\rho^L(S_\mathrm{T}^{-1}(\bar{p}))\right]^2 \mathrm{d}\bar{p}, \tag{7.10a}$$

$$\text{subject to} \int_{\bar{p}_1^R}^{\bar{p}_2^R} F_\mathrm{T}^{R_1}(\bar{p})\,\mathrm{d}\bar{p} = \int_{p_1^R}^{p_2^R} [(1-\mathcal{R})F]_\mathrm{h}^L(p)\,\mathrm{d}p, \tag{7.10b}$$

where $p_1^R = S_\mathrm{T}^{-1}(\bar{p}_1^R)$, etc., and ρ^L and $[(1-\mathcal{R})F]_\mathrm{h}^L(p)$ denote piecewise polynomials on the incident side with the latter given by

$$[(1-\mathcal{R})F]_\mathrm{h}^L(p) = \begin{cases} [(1-\mathcal{R})F]_\mathrm{h}^{L_0}(p) & \text{if } p \in L_0, \\ [(1-\mathcal{R})F]_\mathrm{h}^{L_1}(p) & \text{if } p \in L_1. \end{cases}$$

Analogously, if we want to compute the reflected contribution represented by the polynomial $\rho_\mathrm{R}^{R_2} \in \mathbb{P}_N$ on a face $R_2 = [\bar{p}_3^R, \bar{p}_4^R]$. Then, we solve

$$\min_{\rho_\mathrm{R}^{R_3}\in\mathbb{P}_N} \int_{\bar{p}_3^R}^{\bar{p}_4^R} \left[\rho_\mathrm{R}^{R_3}(\bar{p}) - \mathcal{R}(S_\mathrm{R}^{-1}(\bar{p}))\rho^L(S_\mathrm{R}^{-1}(\bar{p}))\right]^2 \mathrm{d}\bar{p}, \tag{7.11a}$$

$$\text{subject to} \int_{\bar{p}_3^R}^{\bar{p}_4^R} F_\mathrm{R}^{R_1}(\bar{p})\,\mathrm{d}\bar{p} = \int_{p_3^R}^{p_4^R} [\mathcal{R}F]_\mathrm{h}^L(p)\,\mathrm{d}p, \tag{7.11b}$$

where $p_3^R = S_\mathrm{R}^{-1}(\bar{p}_3^R)$, etc., and ρ^L and $[\mathcal{R}F]_\mathrm{h}^L(p)$ denote piecewise polynomials on the incident side.

The different behaviour of partial reflections and total internal reflection represents an issue in the strategy outlined above. Specifically, ρ_T only takes information from incident light with $\delta > 0$, whereas ρ_R takes information from all incident light. This means the partition of an incident momentum

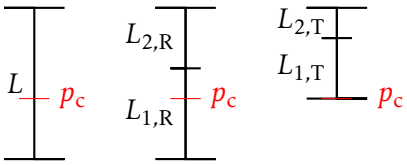


Figure 7.5: Partitioning of an incident momentum interval L near critical momentum p_c into integration intervals $L_{i,R}$ and $L_{i,T}$.

interval L that contains the critical momentum p_c, for which $\delta = 0$, needs to be modified. In that case, the partitioning might look as in Figure 7.5. The incident interval L can be partitioned into two parts along $\delta = 0$ as follows $L = L_{\delta\leq 0} \cup L_{\delta>0}$. The parts $\{L_{i,T}\}_{i=1}^{N_T}$ together cover $L_{\delta>0}$, whereas the parts $\{L_{i,R}\}_{i=1}^{N_T}$ cover L entirely.

Energy conservation is lost because the interpolant $[\mathcal{R}F]_h$ does not satisfy the property that it reduces to the polynomial interpolant F for all $p \in L_{\delta\leq 0}$, whereas for $p \in L_{\delta>0}$ we have $[\mathcal{R}F]_h + [(1-\mathcal{R})F]_h = F$. Indeed, the flux balance (7.9) is no longer satisfied.

The issue is resolved by modifying the polynomial interpolant $[\mathcal{R}F]_h$. The polynomial interpolant $[\mathcal{R}F]_h$ on the face L with $p_c \in L$ is replaced by

$$[\mathcal{R}F]_h^L(p) = \begin{cases} \sum_{j=0}^{N} \mathcal{R}_{j,\delta\leq 0} F_{j,\delta\leq 0}\, \ell_j\Big(\eta_{\delta\leq 0}(p)\Big) & \text{if } p \in L_{\delta\leq 0}, \\ \sum_{j=0}^{N} \mathcal{R}_{j,\delta>0} F_{j,\delta>0}\, \ell_j\Big(\eta_{\delta>0}(p)\Big) & \text{if } p \in L_{\delta>0}, \end{cases} \tag{7.12}$$

with $\eta_{\delta\leq 0}(p)$ an affine linear transformation for $p \in L_{\delta\leq 0}$ to the reference interval $[0,1]$ and similarly for $\eta_{\delta>0}(p)$. The coefficients $F_{j,\delta\leq 0}$ and $\mathcal{R}_{j,\delta\leq 0}$ are defined as

$$F_{j,\delta\leq 0} = F\Big(p_{j,\delta\leq 0}\Big) \quad \text{and} \quad \mathcal{R}_{j,\delta\leq 0} = \mathcal{R}\Big(p_{j,\delta\leq 0}\Big), \tag{7.13}$$

with $p_{j,\delta\leq 0}$ the jth Gauss-Legendre quadrature node on $L_{\delta\leq 0}$, and analogously the coefficients $F_{j,\delta>0}$ and $\mathcal{R}_{j,\delta>0}$ are defined as

$$F_{j,\delta>0} = F\Big(p_{j,\delta>0}\Big) \quad \text{and} \quad \mathcal{R}_{j,\delta>0} = \mathcal{R}\Big(p_{j,\delta>0}\Big), \tag{7.14}$$

with $p_{j,\delta>0}$ the jth Gauss-Legendre quadrature node on $L_{\delta>0}$. The polynomial interpolant $[(1-\mathcal{R})F]_h$ is defined similar to (7.12) but with $\mathcal{R}_j$ replaced with $1-\mathcal{R}_j$. With these definitions, the flux balance (7.9) is satisfied for the specific face that contains the critical momentum.

The constrained least-squares problems (7.10)-(7.11) are solved as described in Section 4.3. In short, either problem is written in terms of a Lagrangian with a Lagrangian multiplier for the energy-conservation constraint.

Subsequently, we impose the requirements for a stationary point and apply $(N+1)$-point Gauss-Legendre quadrature on each (part of a) face. The result is a linear system for the $N+1$ expansion coefficients for the polynomial $\rho_{\mathrm{R/T}}$ and a Lagrange multiplier. Exactly the same matrix structure is found as before, so that the matrix is inverted as described in Appendix A. Finally, from the expansion coefficients the numerical fluxes are computed to be used in the DG schemes.

7.3 Validation energy-conserving numerical fluxes

The discretisation at an optical interface as outlined in this chapter is used on a test case to validate the energy-conservation property. As a test case we take the 'bucket of water' example [92], also used in Section 4.4. For this problem the refractive index field is given by

$$n(q) = \begin{cases} n_0, & \text{if } q \leq 0, \\ n_1, & \text{if } q > 0, \end{cases} \tag{7.15}$$

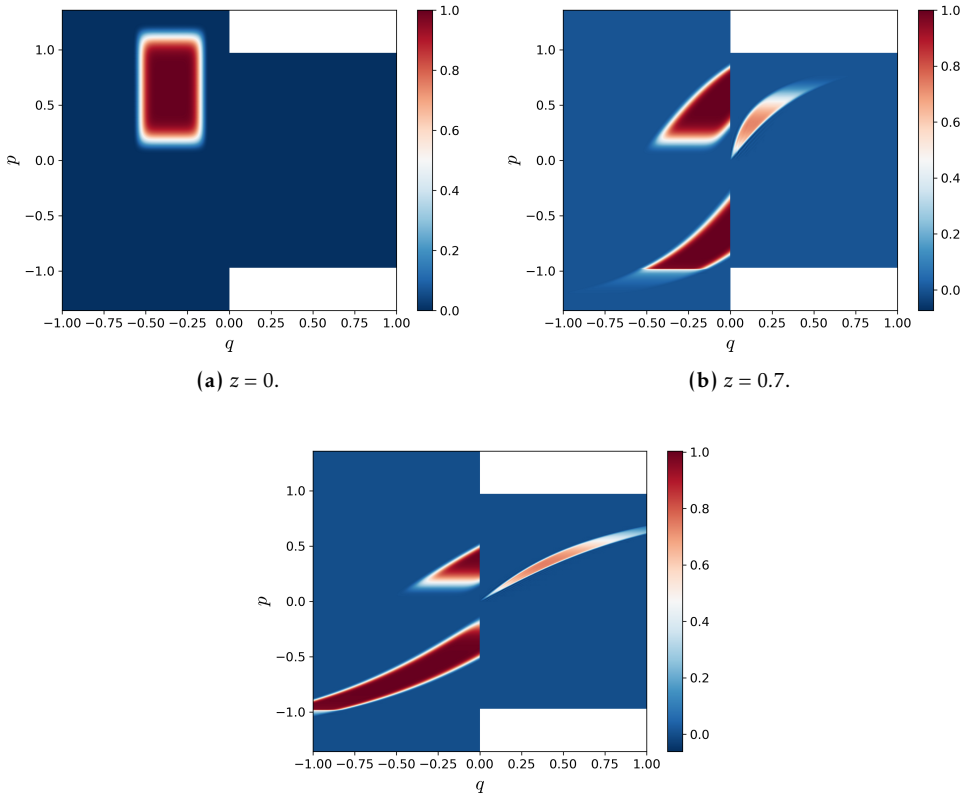
where we take $n_0 = 1.4$ and $n_1 = 1$. Using an initial basic luminance ρ_0 that is non-zero in the region described by $q < 0$ and $p > 0$, results in a solution that features (partial) refraction, total internal reflection and partial reflection. Specifically, the critical momentum is $p_{\mathrm{c}} = \sqrt{n_0^2 - n_1^2} \approx 0.980$ and light described by $q < 0$ and $0 < p \leq p_{\mathrm{c}}$ will undergo total internal reflection, whereas for $p > p_{\mathrm{c}}$ light will undergo Fresnel reflection. The optical interface, $q = 0$, is perpendicular to the z-axis, hence, p_z is preserved upon reflection/refraction at the interface. Consequently, forward-propagating light remains forward-propagating light. For the initial condition ρ_0 at $z = 0$ we take

$$\rho_0(q,p) = \varphi_{m,k}\left(\frac{q-q_0}{\sigma_q}\right)\varphi_{m,k}\left(\frac{p-p_0}{\sigma_p}\right), \tag{7.16}$$

with $m = 6$, $k = 4$, $q_0 = -0.35$, $\sigma_q = 0.25$, $p_0 = 0.65$, $\sigma_p = 0.65$. The function $\varphi_{m,k}$ is defined in (4.66). The maximum momentum is limited to $p_{\max}$ such that the maximum velocity is $p_{\max}/\sqrt{n^2 - p_{\max}^2} = 4$, which leads to $p_{\max} \approx 0.970\, n(q)$. Furthermore, we assume that light is unpolarised which leads to the Fresnel reflection coefficient

$$\mathcal{R}_{\text{unpolarised}} = \tfrac{1}{2}\left(\mathcal{R}_{\parallel} + \mathcal{R}_{\perp}\right). \tag{7.17}$$

The initial condition and the numerical solution at $z = 0.7$ and $z = 1.4$ are shown in Figure 7.6. For this example, we used the ADER-DG scheme with



(a) $z = 0$.

(b) $z = 0.7$.

(c) $z = 1.4$.

Figure 7.6: Distributions of ρ for the bucket of water example with Fresnel reflections computed with the $N = 7$ ADER-DG scheme.

$N = 7$ and a mesh with uniform mesh spacing $\Delta q = 0.05$ and $\Delta p \approx 0.0485$ resulting in 1920 rectangular elements. From the second panel in the figure one can observe in the region $q < 0$, $p < 0$ that part of the distribution got there via total internal reflection and part of the distribution via Fresnel reflection. In the region $q > 0$, $p > 0$ light was partially transmitted. Values below 0 appear in the region $q > 0$, $p > 0$ due to the steep gradients in the solution, which are under-resolved and appear as oscillations in the numerical solution. In the last panel, the solution has propagated further with more light being reflected at the interface.

The luminous flux inside the domain plus the luminous flux leaving the domain through the physical boundaries of the system (excluding the optical interface) should remain constant if the scheme is energy-conserving. We compute the absolute relative deviation from energy conservation at every step

and find that the maximum deviation from energy conservation is $4.88 \cdot 10^{-15}$ and, thus, we observe energy conservation up to machine precision. The discretisation of the jump condition (7.1) at an optical interface as outlined in this chapter is indeed energy-conserving.

Chapter 8

A lens plate

A lens plate is a microlens array where a single microlens is repeated in a regular pattern. Such a lens plate is for instance used in office lighting. A spatially uniform Lambertian source (uniform ρ in phase space) is transformed by the lens plate to a certain target distribution. For the purposes of office lighting, the lighting system is fixed onto the ceiling and should illuminate the desks etc. Light that exits the lighting system at large angles, where the angle is measured with respect to the normal of the ceiling directed towards the floor, is undesirable. Therefore, the target light distribution should contain very little energy for light at large angles. In this chapter, we will study the effect of different design parameters on the output distribution produced by the lens plate[1].

A three-dimensional lens plate is shown in Figure 8.1, where one can see the cones structured into an array. In Figure 8.2 we consider a two-dimensional cross section of the lens plate. The triangles at the top are cross sections of a cone. In practice, the number of triangles can be much larger than ten. To model the lens plate we consider a single microlens, i.e. a unit cell, and prescribe periodic boundary conditions. The geometry of a lens plate causes multiple effects to appear, such as total internal reflection, and partial reflection and transmission via Fresnel reflection. These effects combined with a spatially uniform Lambertian source as initial condition leads in general to discontinuities and discontinuous derivatives for the basic luminance distribution. We will employ the ADER-DG scheme on a moving mesh to solve Liouville's equation. The discontinuities will give rise to oscillations in the numerical solution if left unchecked. To resolve these problems we will apply a modal filter and a limiter.

[1] I would like to thank Gilles Vissenberg from Signify Research for our discussions of the results.

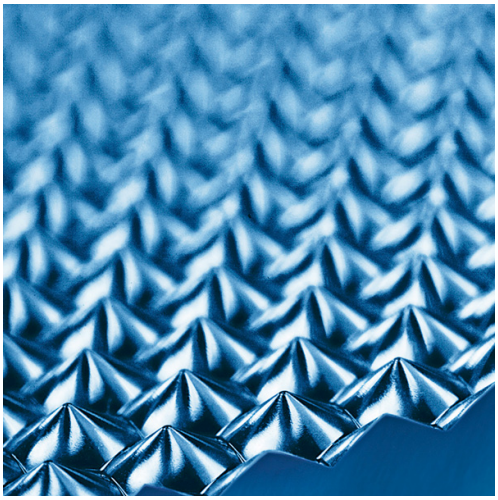


Figure 8.1: A lens plate. Source: [52].

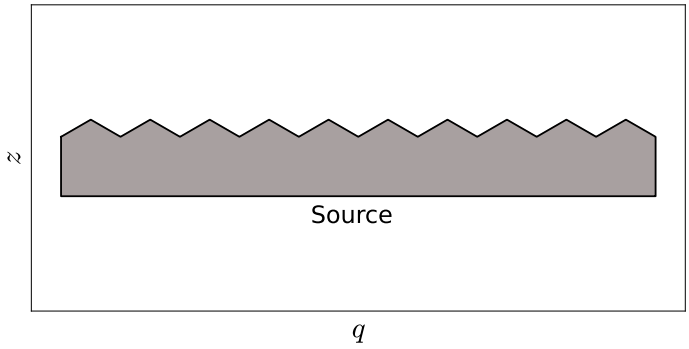


Figure 8.2: Two-dimensional cross section of the lens plate. White represents the background medium with $n_0 = 1$ while gray represents a medium with index $n_1 = 1.5$.

At an optical interface the propagation direction of a light ray can change. Specifically for this application accounting for light that undergoes this change will be important. Consequently, we will solve Liouville's equation in an iterative manner, by first solving for forward-propagating light and then for backward-propagating light etc. We assume once more that light is unpolarised so that the reflection coefficient in (2.54) is given by

$$\mathcal{R}_{\text{unpolarised}} = \tfrac{1}{2}\left(\mathcal{R}_{\parallel} + \mathcal{R}_{\perp}\right). \tag{8.1}$$

The jump condition (2.54) is discretised via straightforward interpolation, rather than the energy-conservative method discussed in previous chapter.

8.1 Modal filter and limiter

As mentioned the basic luminance can be non-smooth or even discontinuous. At an optical interface we apply a jump condition for ρ, hence, the distribution already is discontinuous. In the DG methods from previous sections, the mesh is aligned with the optical interface. Consequently, if ρ is discontinuous only at optical interfaces, then this presents no real issue. The DG method can naturally deal with non-smooth distributions as long as the discontinuities of the solution are aligned with the mesh, see the 'bucket of water' problem in Section 4.4.2. In contrast, if the distribution contains a discontinuity that is not aligned with the mesh, then this will give rise to the well-known Gibbs' phenomenon in the numerical solution. That is, for the high-degree polynomial basis used in the DG methods a discontinuity presents itself as high-frequency oscillations.

To deal with discontinuities, we will apply two techniques. The first technique is a modal filter [44], that effectively tries to diminish the high-frequency components of the numerical solution. Second, we will apply the scaling limiter introduced by Zhang & Shu [98], that tries to make the numerical solutions bounds-preserving (without any overshoot or undershoot). Both techniques are only applied when the element is identified as a troubled element. To identify which elements are troubled we use a shock indicator.

The shock indicator is based on the one introduced by Persson and Peraire [73]. It estimates the regularity of the numerical solution from the magnitudes of the high-frequency content. As in the DG method discussed before, the numerical solution on an element is described by the following polynomial

$$\rho_{\text{h}}(\xi) = \sum_{i,j=0}^{N} \rho_{ij}\ell_i(\xi)\ell_j(\eta). \tag{8.2}$$

From the polynomial (8.2) it is not clear what the magnitudes of the high frequencies are. Therefore, we transform it from the Lagrange basis to the modal basis described by Legendre polynomials.

The Legendre polynomials defined over the unit interval $[0,1]$ are denoted $L_i(\xi)$ and are orthogonal on $[0,1]$, viz.,

$$\int_0^1 L_i(\xi)L_j(\xi)\,\mathrm{d}\xi = \frac{1}{2i+1}\delta_{ij}. \tag{8.3}$$

The polynomial (8.2) is rewritten in the modal basis as

$$\rho_\mathrm{h}(\boldsymbol{\xi}) = \sum_{i,j=0}^{N} \rho_{ij}\ell_i(\xi)\ell_j(\eta) = \sum_{i,j=0}^{N} \hat{\rho}_{ij}L_i(\xi)L_j(\eta), \tag{8.4}$$

where $\hat{\rho}_{ij}$ represent the modal coefficients. Next, a truncated expansion is created by dropping the highest modes, i.e.,

$$\rho_{\mathrm{h,trunc}}(\boldsymbol{\xi}) = \sum_{\substack{i,j=0 \\ i+j<N}}^{N-1} \hat{\rho}_{ij}L_i(\xi)L_j(\eta). \tag{8.5}$$

The indicator is defined as

$$S = \frac{\left\|\rho_\mathrm{h} - \rho_{\mathrm{h,trunc}}\right\|_2^2}{\left\|\rho_\mathrm{h}\right\|_2^2}, \tag{8.6}$$

where $\|\cdot\|_2$ denotes the L_2-norm on the reference domain $[0,1]^2$. Persson and Peraire argue in [73] that for smooth solutions their indicator should scale as $1/N^4$. An activation function α is defined as follows

$$\alpha(s) = \begin{cases} 0 & \text{if } s \le s_\mathrm{ref} - \kappa, \\ \frac{s - s_\mathrm{ref} + \kappa}{2\kappa} & \text{if } s_\mathrm{ref} - \kappa < s \le s_\mathrm{ref} + \kappa \quad \text{with } s = \log_{10}(S), \\ 1 & \text{if } s > s_\mathrm{ref} + \kappa. \end{cases} \tag{8.7}$$

Here, 2κ is the width of the activation ramp and the other parameter takes on the value $s_\mathrm{ref} = -4 - 4.25\log_{10}(N)$ [4].

The modal filter and limiter are both activated when $\alpha(s) > 0$. For the modal filter we modify the polynomial (8.4) as follows

$$\rho_\mathrm{h}^*(\boldsymbol{\xi}) = \sum_{i,j=0}^{N} \sigma\left(\frac{i+j}{N}\right)\hat{\rho}_{ij}L_i(\xi)L_j(\eta), \tag{8.8}$$

with $\sigma(x)$ the following filter function

$$\sigma(x) = \exp\left[-\nu\alpha x^p\right] \tag{8.9}$$

where $\nu > 0$ represents the strength of the filter and $p > 0$ the degree of the filter. For more details about modal filters and its relation to artificial viscosity methods we refer the reader to [4, 42, 44, 54, 73]. We remark that the modal filter (8.8) is energy conservative as $\sigma(0) = 1$ and thus the average of the polynomial, $\hat{\rho}_{00}$, is not modified.

The limiter by Zhang & Shu [98, 99] works by applying a scaling to the polynomial around its average. If the solution to Liouville's equation is expected to satisfy $\rho(z, \boldsymbol{x}) \in [m, M]$, with m the minimal and M the maximal value of ρ. Then the limiter replaces the polynomial ρ_{h}^* given by (8.8) with ρ_{h}^{**} as follows

$$\rho_{\mathrm{h}}^{**}(\boldsymbol{\xi}) = \theta\left(\rho_{\mathrm{h}}^*(\boldsymbol{\xi}) - \bar{\rho}\right) + \bar{\rho} \tag{8.10a}$$

with

$$\theta = \min\left\{\left|\frac{M-\bar{\rho}}{M_e-\bar{\rho}}\right|, \left|\frac{m-\bar{\rho}}{m_e-\bar{\rho}}\right|, 1\right\}, \tag{8.10b}$$

and $\bar{\rho}$ denotes the average over the element, and m_e and M_e are given by

$$m_e = \min_{i,j} \rho_{ij} \quad \text{and} \quad M_e = \max_{i,j} \rho_{ij}, \tag{8.10c}$$

where $i \in \{0, 1, \ldots, N\}$ and $j \in \{0, 1, \ldots, N\}$. The scaling limiter (8.10) preserves the average of the polynomial, hence, it is energy conservative.

When an element satisfies $\alpha(s) > 0$, the modal filter is first applied followed by the use of the limiter.

8.2 Geometry of the lens plate

The geometry of the unit cell of a lens plate, which is a cross section of a cone with a rounded top, is shown in Figure 8.3. In the figure, three design parameters are shown. First, the thickness d and second the half-angle of the top θ. Third is the so-called rounding radius R that models the top as a circle segment. The reasoning behind this parameter is due to the production process. It is difficult to manufacture a very sharp top (R is close to 0), so that the top part is effectively a triangle. As can be seen in the figure, the circle segment connects to the flat sides of the top triangular region and the surface normal varies continuously. The parameter θ is defined independent of R. As

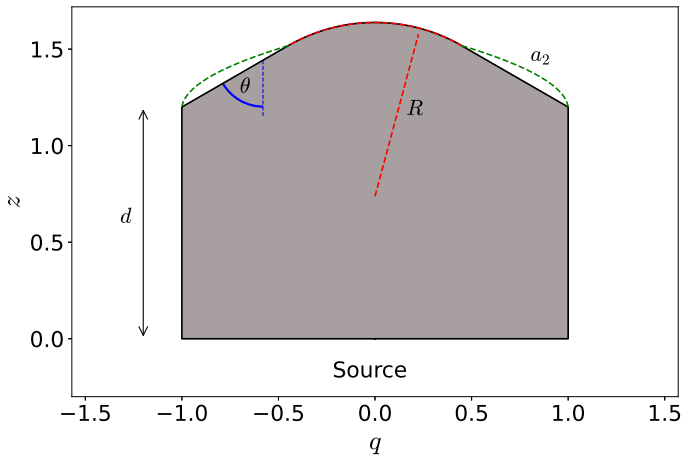


Figure 8.3: Unit cell for $d = 1.2$, $\theta = 60\,\text{deg}$, $R = 0.9$ and $a_2 = 0$. Green dashed lines represents an alternative cell with $a_2 = 5$. White represents the background medium with $n_0 = 1$ while gray represents a medium with index $n_1 = 1.5$.

a fourth parameter we introduce a_2 that curves the right side, $q > 0$, of the triangle according to $q = Q_\text{r}(z)$ with

$$Q_\text{r}(z) = Q_0 + a_1(z-d) + a_2(z-d)(Z_1 - z) \quad z \in [d, Z_1], \tag{8.11}$$

where $Q_0 = 1$, and the coefficients Z_1 and a_1 are determined by the parameters R, θ and d. Here, $z = Z_1$ corresponds to the lowest z-value of the circle segment. Note that if $a_2 = 0$, then the sides are flat as shown in Figure 8.3. For $a_2 \neq 0$, the surface normal is no longer continuous. The curved left side of the triangle is given by $q = -Q_\text{r}(z)$.

At $q = \pm 1$ periodic boundary conditions are applied and at $z = 0$ there is a light source. As described before the light source is a spatially uniform Lambertian source. The initial condition for the forward light is described by

$$\rho_\text{f}(z = 0^+, q, p) = \begin{cases} 1 & \text{if } |p| < n_0 - \varepsilon, \\ 0 & \text{otherwise,} \end{cases} \tag{8.12}$$

with ε describing the cut-off due to the maximum momentum.

Note that the light source is actually outside the lens plate. However, the light that is partially reflected backwards at the interface $z = 0$ is assumed to be uniformly scattered again towards the interface $z = 0$. Hence, if this process is repeated an infinite number of times, effectively all light will be transmitted from the source into the plate (without any losses). In the solution process we thus start inside the lens plate at $z = 0^+$ with the initial condition (8.12).

In Liouville's equation light can only have a change in propagation direction at an optical interface. When we solve Liouville's equation for forward-propagating light, $\sigma = \mathrm{f}$, we do not yet know the distribution of the backward-propagating light ρ_b at an optical interface. Hence, we first assume that $\rho_\mathrm{b} = 0$ everywhere and perform one solve of Liouville's equation from $z = 0$ to the top of the device at $z = Z_{\mathrm{top}}$. During the solution process we can then compute the light that changes direction at an optical interface. Next, we solve for ρ_b by starting at the top $z = Z_{\mathrm{top}}$ and stepping to $z = 0$. In this process we can now account for light that has changed direction at an optical interface. This process is repeated in an iterative manner, by alternating between solving for ρ_f and ρ_b, for a fixed number of iterations.

The phase space mesh at a fixed z-level can in general be different between iterations. This is mainly caused by the mesh refinement algorithm. But this does not represent an issue when accounting for light that changed propagation direction. The only requirement we need is that the (z,p)-mesh, i.e., the discrete z-levels and faces (momentum intervals), at an optical interface remains the same throughout the iterative solution process. This ensures that the quadrature nodes of the z- and p-integrals in the ADER-DG scheme match for both directions. In the scheme this is accommodated by using the exact same z-levels for both directions. The criterion for the momentum mesh is easily met by using the same mesh spacing.

We consider the lens plate with $d = 1.2$, $\theta = 60\,\mathrm{deg}$, $R = 0.05$. At $z = d$ the regions corresponding to $n = n_0$ have zero width in physical space along the q-direction, and starts to increase when z increases. For instance, for the right region the width between $q = 1$ and the optical interface is given by

$$1 - Q_\mathrm{r}(z) = a_1(z-d) + a_2(z-d)(Z_1 - z),$$

where we used $Q_0 = 1$. Since the width is zero at $z = d$, the phase space volume is also zero where $n = n_0$. Thus, if we would assign an element to this phase space volume, then by the CFL-condition we cannot take any step. To resolve this issue, we slightly perturb the interfaces near the edges $q = \pm 1$ to $q = \pm(1-\delta)$ with $\delta > 0$ at $z = d$. As a consequence, we have a flat interface at $z = d$ with $q \in [1-\delta, 1]$, see Figure 8.4. Similarly, there is flat interface at $z = d$ with $q \in [-1, -(1-\delta)]$. In (8.11) we set $Q_0 = 1 - \delta$ and slightly perturb a_1, such that $q = Q_\mathrm{r}(z)$ connects to the circle. The value for δ is fixed to $\delta = 10^{-5}$ for all presented results.

We consider a maximum of five iterations, i.e., five forward solves and five backward solves. The maximum momentum is set to $p_{\max} = 0.999n$, such that $\varepsilon = 0.001 n_0$, and CFL = 0.9 and $\alpha = 2.25$ in the mesh refinement procedure. The parameters for the modal filter are $\nu = 10^{-4}$, $p = 4$, $\kappa = 0.5$. The obtained

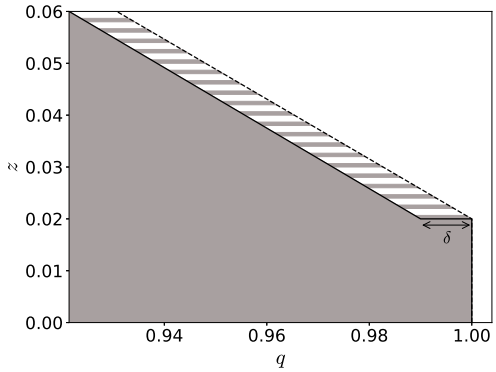


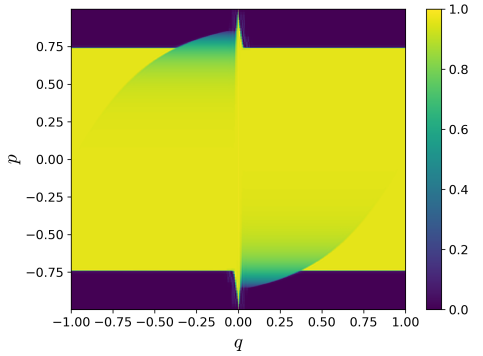
Figure 8.4: Approximation of the lens plate at $z = d$ with $\delta = 10^{-2}$. Dashed line represents the actual lens plate, the solid line the approximated geometry. White represents the background medium with $n_0 = 1$ while gray represents a medium with index $n_1 = 1.5$.

numerical distributions with $\Delta q_{\max} = 4.8 \cdot 10^{-2}$, $\Delta p = 1.82 \cdot 10^{-2}$ and $N = 5$ are shown in Figure 8.5. The forward distributions are computed at $z = Z_{\text{top}}$, whereas the backward distributions are computed at $z = 0^+$ (just before light strikes $z = 0$).

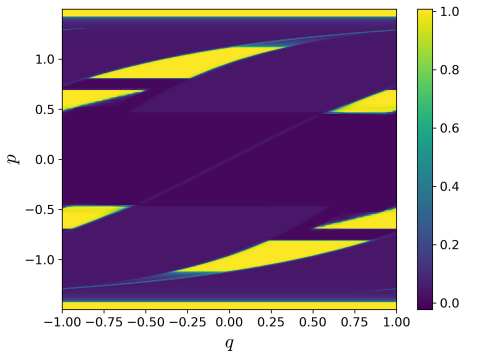
From the figures one can see there is practically no undershoot or overshoot in the numerical solutions. Moreover, there are no visible large oscillations on the boundaries of the patches in the second panel. See also Figure 8.6 for cross sections of the distributions of ρ at $q = -0.5$, which corresponds to the first two panels of Figure 8.5. There one can also see the absence of undershoot and overshoot.

In the first panel of Figure 8.5 the gradients in the distributions are caused by partial transmission, in which the reflectivity depends on the incident angle. In the second panel, one can see patches where in the interior ρ takes on a value close to 1, hence, light got there via total internal reflection. Furthermore, it might be difficult to see but there is a region on the second panel where $\rho \approx 0$ and regions where ρ is small, where the latter corresponds to partial reflection. This is more clearly visible in the cross section shown in the second panel of Figure 8.6. In the last four panels of Figure 8.5 we observe more smaller structures appearing in the solution and in the last two panels we observe a decrease in the maximum value of ρ. These last two panels hence correspond to a relatively low amount of luminous flux.

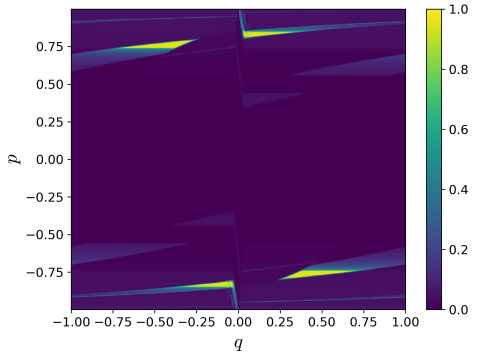
At $z = Z_{\text{top}}$ the basic luminance ρ_{f} for the individual iterations are added. As mentioned before the phase space meshes for different iterations do not match. The meshes have to match so that the solutions can be added together. Therefore, the solutions are first projected onto a finer uniform mesh with $\Delta q =$



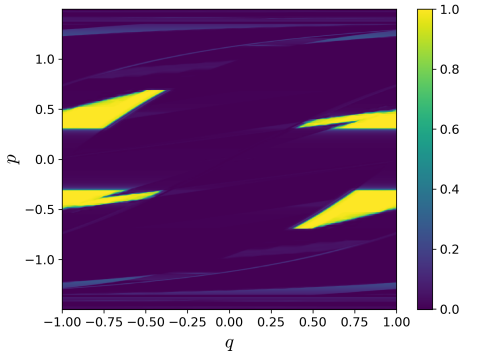
(a) Forward iteration 1.



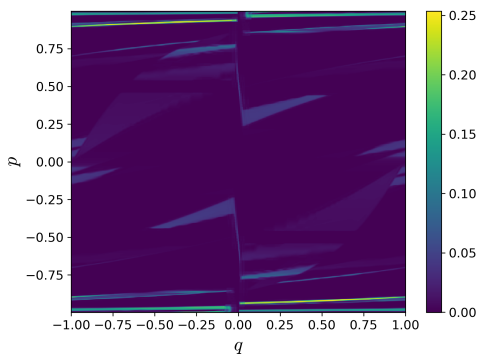
(b) Backward iteration 1.



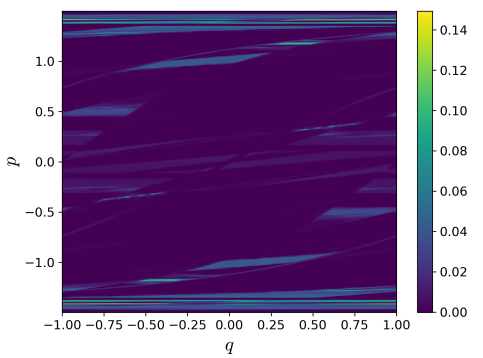
(c) Forward iteration 2.



(d) Backward iteration 2.

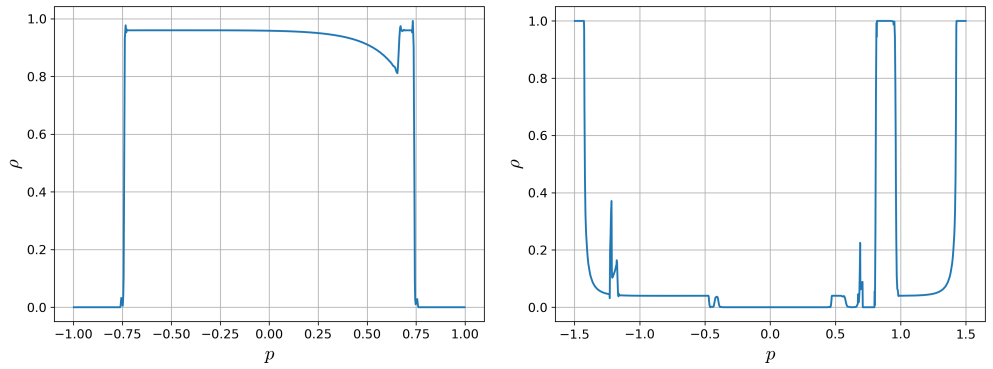


(e) Forward iteration 3.



(f) Backward iteration 3.

Figure 8.5: Distributions of ρ for the lens plate at different iterations solved with the $N = 5$ ADER-DG scheme.



(a) Forward iteration 1.

(b) Backward iteration 1.

Figure 8.6: Cross sections of ρ for the lens plate at $q = -0.5$ solved with the $N = 5$ ADER-DG scheme.

$5.1 \cdot 10^{-3}$ and $\Delta p = 1.82 \cdot 10^{-2}$ before being added. The resulting distribution after five iterations is shown in Figure 8.7. We observe the additional patches for large momenta $|p| > 0.74$ compared to the first forward solution shown in the first panel in Figure 8.5. Although the individual iterative solutions seemed to obey the limits of $\rho \in [0,1]$ pretty well, the combined solution clearly does not.

Finally, the luminous intensity I is computed. The luminous intensity is computed by evaluating

$$I(p) = \int_Q \rho_f(Z_{\text{top}}, q, p) p_z(p) \, \mathrm{d}q, \tag{8.13}$$

with $p_z = \sqrt{n_0^2 - p^2}$ and $Q = [-1,1]$. The intensity is computed for the cumulative distributions and the result is shown in Figure 8.8. The difference between the cumulative intensities for four and five iterations seems to be very small. In fact, the fifth forward solve of Liouville's equation added a relative amount of 0.09% of the initial luminous flux to the final solution. The center profile of the distribution is close to a Lambertian profile, i.e., the profile is close to a semi-circular one. The light in the tails of the distribution represents unwanted light. As mentioned in the introduction, for this lens plate it is important to suppress light at large angles. The luminous intensity will be the only profile we will be looking at in the parameter study described in the following section.

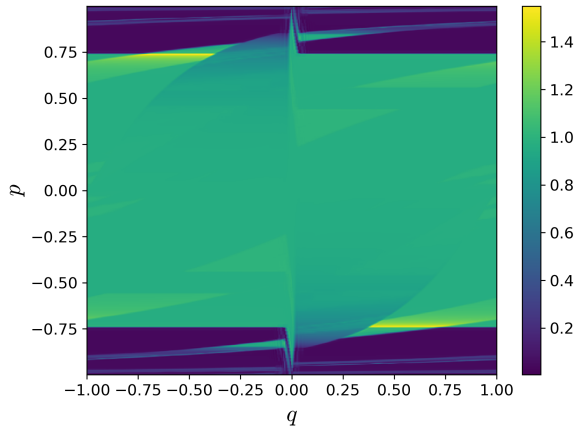


Figure 8.7: Total distribution of ρ_f after five iterations for the lens plate solved with the $N = 5$ ADER-DG scheme.

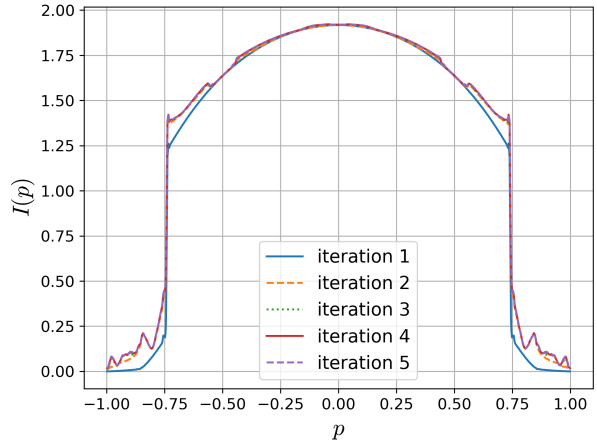


Figure 8.8: Luminous intensity for the cumulative distributions of ρ_f for five iterations solved with the $N = 5$ ADER-DG scheme.

8.3 Parameter study

In the following the distributions are computed with the $N = 4$ ADER-DG scheme with CFL = 0.5, $\Delta q_{\max} = 6.4 \cdot 10^{-2}$ and $\Delta p = 3.22 \cdot 10^{-2}$. The parameters for the modal filter remain the same.

8.3.1 Rounding of the top

First, we investigate the effect of the rounding of the top. The top part of the plate is approximated by a circle segment with radius R. The effect of the radius R parameter on the design is plotted in Figure 8.9.

We fix the other parameters to $\theta = 60\,\text{deg}$, $d = 2$, $a_2 = 0$. Varying the parameter R yields the intensity distributions shown in Figure 8.10. The results show an intensity that is symmetric about $p = 0$, and has a jump around $p \approx 0.74$ which corresponds to the angle $\phi \approx 47.7\,\text{deg}$, measured with respect to the z-axis. Thus rays at large angles have less energy, as is desired. By increasing the radius from $R = 10^{-3}$ to larger values we can see in the zoomed in plots that the intensity for light rays at larger angles keeps increasing. As light at the large angles is undesired, a small value of R should be chosen in the design.

8.3.2 Thickness

Next, we investigate the effect of the thickness d. We start with the spatially uniform Lambertian distribution (8.12) at $z = 0^+$. In the first solve of ρ_f this uniform distribution will not change between $z = 0$ and $z = d$ as the refractive index remains constant and there are periodic boundary conditions. After that light will undergo changes in the triangular region, producing some light that

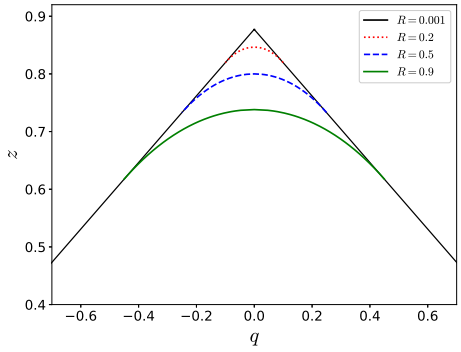
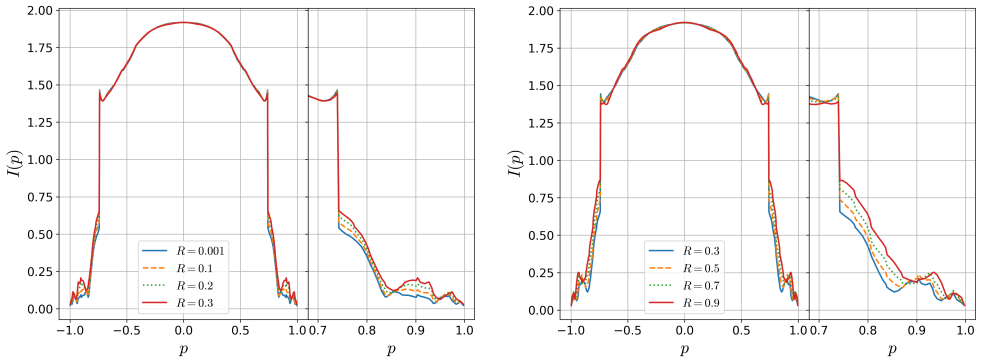


Figure 8.9: Design variations for varying R. Parameters: $\theta = 60\,\text{deg}$, $d = 0.3$ and $a_2 = 0$.



(a) $R = 0.001$ to $R = 0.3$.

(b) $R = 0.3$ to $R = 0.9$.

Figure 8.10: Luminous intensity $I(p)$ for varying R, right part is zoomed in. Parameters: $\theta = 60\,\text{deg}$, $d = 2$ and $a_2 = 0$.

travels backward to the source. The thickness d can thus be used to control where the backward light ends up.

We fix the other parameters to $\theta = 60\,\text{deg}$, $R = 0.05$, $a_2 = 0$. By varying the parameter d we obtain the results shown in Figure 8.11. From the results one can see that the parameter d has a complex effect on the tail of the intensity distribution. For example for $d = 0$, the intensity is high in the tail of the distribution. As we slowly increase d in steps of 0.1 to $d = 0.5$ first a decrease and then again an increase occurs, whereas $d = 0$ and $d = 0.5$ perform similar. In the second panel shown in 8.11 the least intensity in the tail is obtained for $d = 1$, while in the third panel this occurs at $d = 1.25$. The result could probably be fine tuned to find optimal d values with least intensity in the tail.

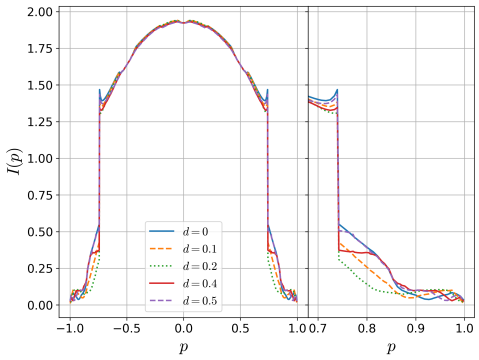
8.3.3 Triangle half-angle

Third, we vary the triangle half-angle. Varying the half-angle θ leads to the results shown in Figure 8.12. The results show how the jump in intensity shifts for different θ values. Moreover, for $\theta = 45\,\text{deg}$, $\theta = 50\,\text{deg}$ and $\theta = 55\,\text{deg}$ large oscillations in the tail of the intensity profile are visible.

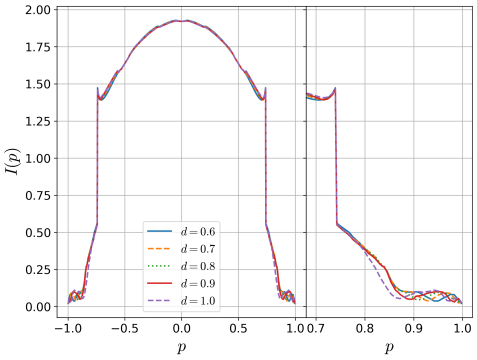
8.3.4 Deviation from triangle

Finally, we vary the shape of the triangle from having flat sides to having curved sides using the parameter a_2. The usual triangle is obtained when $a_2 = 0$. The effect of this parameter is plotted in Figure 8.13.

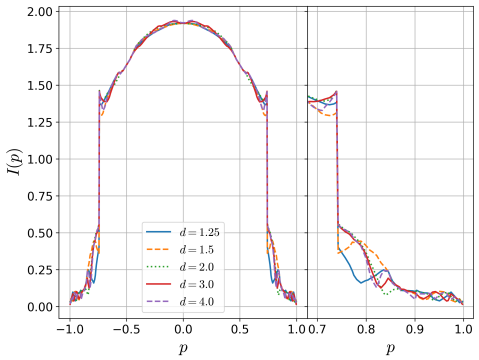
By varying the parameter a_2 whilst fixing $\theta = 60\,\text{deg}$, $R = 0.05$, $d = 2$ we obtain the intensity profiles shown in Figure 8.14. From the first figure we



(a) $d = 0$ to $d = 0.5$.

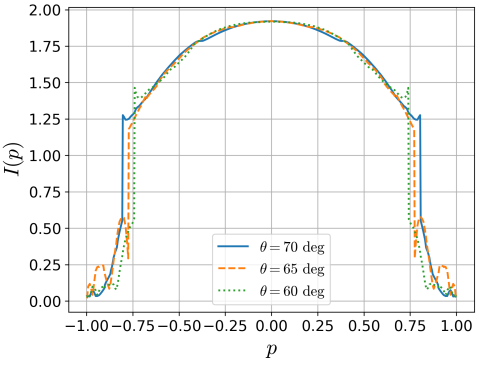


(b) $d = 0.6$ to $d = 1.0$.

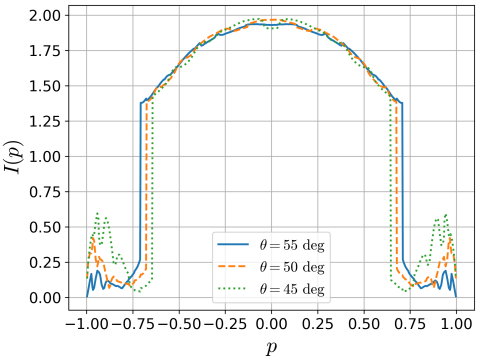


(c) $d = 1.25$ to $d = 4.0$.

Figure 8.11: Luminous intensity $I(p)$ for varying d. Parameters: $\theta = 60\,\text{deg}$, $R = 0.05$ and $a_2 = 0$.



(a) $\theta = 60\,\text{deg}$ to $\theta = 70\,\text{deg}$.



(b) $\theta = 45\,\text{deg}$ to $\theta = 55\,\text{deg}$.

Figure 8.12: Luminous intensity $I(p)$ for varying θ. Parameters: $R = 0.05$, $d = 2$ and $a_2 = 0$.

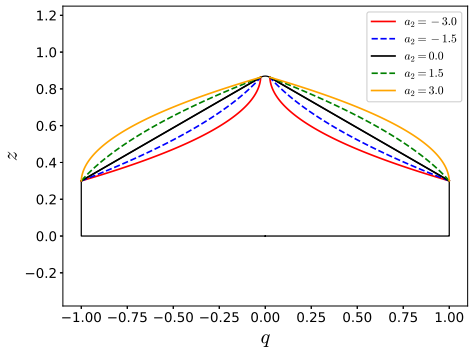
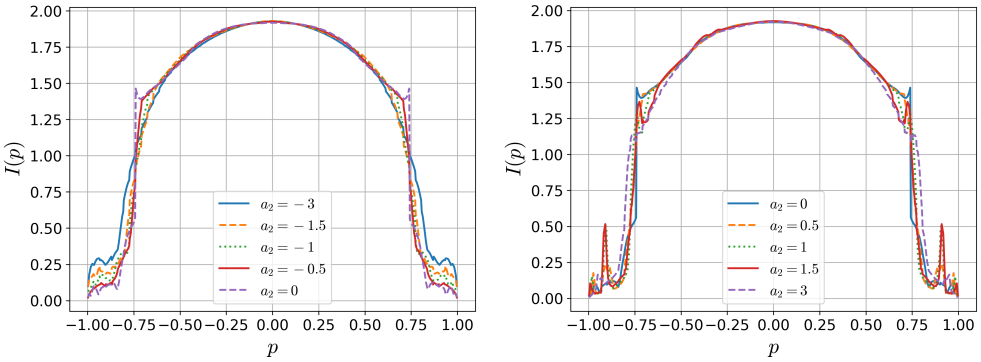


Figure 8.13: Design variations for varying a_2. Parameters: $\theta = 60\,\text{deg}$, $R = 0.05$ and $d = 0.3$.



(a) $a_2 = -3$ to $a_2 = 0$.

(b) $a_2 = 0$ to $a_2 = 3$.

Figure 8.14: Luminous intensity $I(p)$ for varying a_2. Parameters: $\theta = 60\,\text{deg}$, $R = 0.05$ and $d = 2$.

can see that as a_2 decreases from 0 to −3 that the jump in intensity is smeared out. For very low a_2 values the intensity in the center profile ($p \in [-0.74, 0.74]$) starts to deviate from a Lambertian profile. For $a_2 = -0.5$ the intensity profile is closest to that of $a_2 = 0$. The intensity in the tail is still higher than compared to $a_2 = 0$.

For the positive values of a_2 a peak occurs around $p \approx 0.84$, except for $a_2 = 3$. Similar to the negative values of a_2, the jump in intensity is a bit smeared out and for large values of a_2 the center profile deviates from a Lambertian profile. Considering all the profiles, the one for $a_2 = 0$ achieves the lowest intensity in the tail of the distribution.

8.4 Concluding remarks

The ADER-DG scheme was applied to solve for the basic luminance distribution of a lens plate. A modal filter and limiter were applied to reduce oscillations in the presence of discontinuities and to control undershoot and overshoot of the numerical solution. The parameter study of the lens plate revealed that having a sharp top, i.e., low value of R, reduces the unwanted light at large angles. For the thickness d there are optimal values that reduce the energy in the tail of the luminous intensity distribution. Varying the triangle half-angle θ shows how the location of the jump in the luminous intensity changes. Finally, curving the sides of the triangle with the parameter a_2 leads in general to less sharp jumps in the luminous intensity. Thus in general having a flat triangle, i.e., $a_2 = 0$, seems to be best for reducing the energy in the tail of the intensity distribution.

For future research the three-dimensional variant of the lens plate could be studied by solving Liouville's equation. In the next chapter, we make a first step towards solving three-dimensional optical systems by detailing an ADER-DG method on a moving mesh that describes the four-dimensional phase space.

Chapter 9

ADER-DG on a moving mesh for 3D optics

At last, we arrive at the topic 'three-dimensional optics'. The term 'three-dimensional' however does not accurately capture the dimensionality of the problem when considering non-zero étendue optics. For instance, at a fixed plane $z = \text{const}$ phase space is four-dimensional rather than the two-dimensional phase space domains from previous chapters.

The higher dimensionality of phase space is not the only added difficulty. In general optical interfaces can be arbitrary curved surfaces in physical space and, therefore, at a fixed plane $z = \text{const}$ optical interfaces can be arbitrary curves. Aligning the mesh on phase space with optical interfaces thus requires the usage of elements that are bounded by curved surfaces. These type of elements are in general known as curvilinear elements.

In this chapter, the ALE-ADER-DG scheme from Chapter 5 is extended to deal with a four-dimensional phase space. First, the discretisation of Liouville's equation on a moving curvilinear mesh is discussed in Section 9.1. Second, in Section 9.2, the necessary temporal Taylor expansions, which are used in the ADER approach, are developed. In Section 9.3 we discuss the discretisation at an optical interface. Finally, we present some first results in Section 9.4 from numerical experiments.

9.1 DG on a moving curvilinear mesh

As before, we consider Liouville's equation for only forward-propagating light rays ($\sigma = 1$), which is given by

$$\frac{\partial \rho}{\partial z} + \nabla \cdot (\rho \boldsymbol{u}) = 0, \tag{9.1}$$

where $\nabla = (\frac{\partial}{\partial \boldsymbol{q}}, \frac{\partial}{\partial \boldsymbol{p}})$ and

$$\boldsymbol{u} = \frac{1}{\sqrt{n(z,\boldsymbol{q})^2 - |\boldsymbol{p}|^2}} \begin{pmatrix} \boldsymbol{p} \\ n \frac{\partial n}{\partial \boldsymbol{q}} \end{pmatrix}. \tag{9.2}$$

For the backward-propagating light rays we assume the basic luminance distribution to be 0 everywhere.

We consider piecewise constant refractive index fields which is the most common situation in geometrical optics. This somewhat simplifies the required geometry as the momentum domain ($|\boldsymbol{p}| \leq n$) does not vary within an element, since the refractive index field can then be chosen to be constant within each element. We partition the four-dimensional phase space into elements that are an image of a four-dimensional unit hypercube, called the tesseract. The tesseract is a reference element given by $E^4 = [0,1]^4$.

In the ALE-ADER-DG scheme considered in Chapter 5 we covered the two-dimensional phase space with rectangular elements, which consisted of a tensor product of two one-dimensional intervals describing the position and momentum, respectively. To partition the four-dimensional phase space we employ a similar strategy. Specifically, we use a tensor product of two two-dimensional quadrilaterals. As before, the elements are allowed to move along the position domain as a function of the evolution coordinate z. The mapping for the phase space coordinates $\boldsymbol{x} \in \mathbb{R}^4$ is given by

$$\boldsymbol{x}(\tau, \xi, \eta) = \begin{pmatrix} \boldsymbol{q}(\tau, \xi) \\ \boldsymbol{p}(\eta) \end{pmatrix}, \tag{9.3}$$

and $z = \tau$, and where $\boldsymbol{q} \in \mathbb{R}^2$ and $\boldsymbol{p} \in \mathbb{R}^2$ describe mappings in terms of spatial coordinates $\xi \in \mathbb{R}^2$ and $\eta \in \mathbb{R}^2$, respectively, that can map from $E^2 = [0,1]^2$ to a quadrilateral bounded by curves.

To allow for arbitrary geometries, we approximate the curves of a quadrilateral using an isoparametric approach. This means that we approximate each curve by a polynomial interpolant of the same order as the numerical

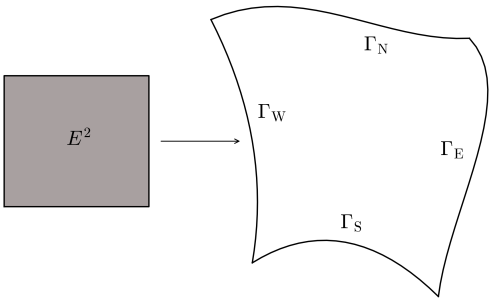


Figure 9.1: Mapping between the unit reference square E^2 and a quadrilateral bounded by curves.

solution. In particular, we approximate a curve $\Gamma : [0,1] \to \mathbb{R}^2$, either $\boldsymbol{q}$ or $\boldsymbol{p}$, by

$$\Gamma_h(s) = \sum_{k=0}^{N} \Gamma(s_k)\ell_k^{\mathrm{GLC}}(s), \tag{9.4}$$

where ℓ_k^{GLC} represents the kth Lagrange polynomial of degree N with nodes at the $(N+1)$-point Gauss-Lobatto-Chebyshev quadrature nodes over the interval $[0,1]$. These quadrature nodes include the boundary points of the interval. Hence, the interpolated curve will exactly coincide with the curve at these boundary points.

The mapping between a set of physical coordinates $\boldsymbol{y} \in \mathbb{R}^2$ and the unit reference square E^2, with coordinates $\boldsymbol{\xi} = (\xi, \eta) \in E^2$, reads as follows [57]

$$\begin{aligned} \boldsymbol{y}(\tau, \boldsymbol{\xi}) = {} & (1-\xi)\Gamma_W(\tau,\eta) + \xi\Gamma_E(\tau,\eta) + (1-\eta)\Gamma_S(\tau,\xi) + \eta\Gamma_N(\tau,\xi) \\ & - \Big[(1-\xi)(1-\eta)\Gamma_W(\tau,0) + (1-\xi)\eta\Gamma_W(\tau,1) \\ & + \xi(1-\eta)\Gamma_E(\tau,0) + \xi\eta\Gamma_E(\tau,1)\Big], \end{aligned} \tag{9.5}$$

with Γ_W denoting the western curved boundary etc., and the subscript h has been dropped for brevity. In (9.5) the mapping also depends on τ, so that the element can move as a function of τ. The mapping is illustrated in Figure 9.1. If in (9.5) the curved boundaries are just straight line segments, then the mapping will be a bilinear interpolation in $\boldsymbol{\xi}$.

To derive a weak formulation for Liouville's equation, we first transform it from a physical domain to the reference domain. This entails transforming the divergence. First, the chain rule is used to relate gradients on $\boldsymbol{q} = (q_0, q_1)$

to $\boldsymbol{\xi} = (\xi_0, \xi_1)$, i.e.,

$$\frac{\partial}{\partial q_0} = \frac{\partial \xi_0}{\partial q_0}\frac{\partial}{\partial \xi_0} + \frac{\partial \xi_1}{\partial q_0}\frac{\partial}{\partial \xi_1}, \tag{9.6a}$$

$$\frac{\partial}{\partial q_1} = \frac{\partial \xi_0}{\partial q_1}\frac{\partial}{\partial \xi_0} + \frac{\partial \xi_1}{\partial q_1}\frac{\partial}{\partial \xi_1}. \tag{9.6b}$$

Analogous relations exists between $\boldsymbol{p} = (p_0, p_1)$ and $\boldsymbol{\eta} = (\eta_0, \eta_1)$. The gradient $\nabla = (\frac{\partial}{\partial \boldsymbol{q}}, \frac{\partial}{\partial \boldsymbol{p}})$ can be written as

$$\begin{pmatrix} \frac{\partial}{\partial \boldsymbol{q}} \\ \frac{\partial}{\partial \boldsymbol{p}} \end{pmatrix} = \begin{pmatrix} \left(\frac{\partial \boldsymbol{\xi}}{\partial \boldsymbol{q}}\right)^{\mathrm{T}} & \mathbf{0} \\ \mathbf{0} & \frac{\partial \boldsymbol{\eta}}{\partial \boldsymbol{p}}^{\mathrm{T}} \end{pmatrix} \begin{pmatrix} \frac{\partial}{\partial \boldsymbol{\xi}} \\ \frac{\partial}{\partial \boldsymbol{\eta}} \end{pmatrix}, \tag{9.7}$$

with the superscript T denoting the transpose.

The divergence of an arbitrary vector field $\boldsymbol{F} = (\boldsymbol{f}, \boldsymbol{g}) = (f_0, f_1, g_0, g_1)$ reads

$$\nabla \cdot \boldsymbol{F} = \frac{\partial}{\partial \boldsymbol{q}} \cdot \boldsymbol{f} + \frac{\partial}{\partial \boldsymbol{p}} \cdot \boldsymbol{g}. \tag{9.8}$$

With the chain rule (9.6) we have for the first term on the right-hand side

$$\frac{\partial}{\partial \boldsymbol{q}} \cdot \boldsymbol{f} = \frac{\partial \xi_0}{\partial q_0}\frac{\partial f_0}{\partial \xi_0} + \frac{\partial \xi_1}{\partial q_0}\frac{\partial f_0}{\partial \xi_1} + \frac{\partial \xi_0}{\partial q_1}\frac{\partial f_1}{\partial \xi_0} + \frac{\partial \xi_1}{\partial q_1}\frac{\partial f_1}{\partial \xi_1}. \tag{9.9}$$

Next, we have $\frac{\partial \boldsymbol{\xi}}{\partial \boldsymbol{q}}^{\mathrm{T}} = \left(\frac{\partial \boldsymbol{q}}{\partial \boldsymbol{\xi}}\right)^{-\mathrm{T}}$. This leads to

$$\left(\frac{\partial \boldsymbol{\xi}}{\partial \boldsymbol{q}}\right)^{\mathrm{T}} = \begin{pmatrix} \frac{\partial \xi_0}{\partial q_0} & \frac{\partial \xi_1}{\partial q_0} \\ \frac{\partial \xi_0}{\partial q_1} & \frac{\partial \xi_1}{\partial q_1} \end{pmatrix} = \begin{pmatrix} \frac{\partial q_0}{\partial \xi_0} & \frac{\partial q_1}{\partial \xi_0} \\ \frac{\partial q_0}{\partial \xi_1} & \frac{\partial q_1}{\partial \xi_1} \end{pmatrix}^{-1} = \frac{1}{\mathcal{J}_q} \begin{pmatrix} \frac{\partial q_1}{\partial \xi_1} & -\frac{\partial q_1}{\partial \xi_0} \\ -\frac{\partial q_0}{\partial \xi_1} & \frac{\partial q_0}{\partial \xi_0} \end{pmatrix}, \tag{9.10}$$

with $\mathcal{J}_q$ the Jacobian determinant of $\frac{\partial \boldsymbol{q}}{\partial \boldsymbol{\xi}}$. With relation (9.10) we can write equation (9.9) as

$$\frac{\partial}{\partial \boldsymbol{q}} \cdot \boldsymbol{f} = \frac{1}{\mathcal{J}_q}\left[\frac{\partial q_1}{\partial \xi_1}\frac{\partial f_0}{\partial \xi_0} - \frac{\partial q_1}{\partial \xi_0}\frac{\partial f_0}{\partial \xi_1} - \frac{\partial q_0}{\partial \xi_1}\frac{\partial f_1}{\partial \xi_0} + \frac{\partial q_0}{\partial \xi_0}\frac{\partial f_1}{\partial \xi_1}\right]. \tag{9.11}$$

Relation (9.11) can be written into a conservative form, i.e.,

$$\frac{\partial}{\partial \boldsymbol{q}} \cdot \boldsymbol{f} = \frac{1}{\mathcal{J}_q}\left[\frac{\partial}{\partial \xi_0}\left(\frac{\partial q_1}{\partial \xi_1} f_0 - \frac{\partial q_0}{\partial \xi_1} f_1\right) + \frac{\partial}{\partial \xi_1}\left(-\frac{\partial q_1}{\partial \xi_0} f_0 + \frac{\partial q_0}{\partial \xi_0} f_1\right)\right]. \tag{9.12}$$

As the transformation for $\boldsymbol{p}$ and $\boldsymbol{\eta}$ is completely analogous, we can write

$$\frac{\partial}{\partial \boldsymbol{p}} \cdot \boldsymbol{g} = \frac{1}{\mathcal{J}_p}\left[\frac{\partial}{\partial \eta_0}\left(\frac{\partial p_1}{\partial \eta_1} g_0 - \frac{\partial p_0}{\partial \eta_1} g_1\right) + \frac{\partial}{\partial \eta_1}\left(-\frac{\partial p_1}{\partial \eta_0} g_0 + \frac{\partial p_0}{\partial \eta_0} g_1\right)\right]. \tag{9.13}$$

Combined the divergence $\nabla \cdot \boldsymbol{F}$ is written as

$$\nabla \cdot \boldsymbol{F} = \frac{1}{\mathcal{J}}\left(\frac{\partial}{\partial \boldsymbol{\xi}} \cdot \tilde{\boldsymbol{f}} + \frac{\partial}{\partial \boldsymbol{\eta}} \cdot \tilde{\boldsymbol{g}}\right), \tag{9.14a}$$

with $\mathcal{J} = \mathcal{J}_q \mathcal{J}_p$. Here, $\mathcal{J}_q = \mathcal{J}_q(\tau, \boldsymbol{\xi})$ and $\mathcal{J}_p = \mathcal{J}_p(\boldsymbol{\eta})$ and

$$\tilde{\boldsymbol{f}} = \mathcal{J}_p \begin{pmatrix} \frac{\partial q_1}{\partial \xi_1} & -\frac{\partial q_0}{\partial \xi_1} \\ -\frac{\partial q_1}{\partial \xi_0} & \frac{\partial q_0}{\partial \xi_0} \end{pmatrix} \boldsymbol{f}, \tag{9.14b}$$

$$\tilde{\boldsymbol{g}} = \mathcal{J}_q \begin{pmatrix} \frac{\partial p_1}{\partial \eta_1} & -\frac{\partial p_0}{\partial \eta_1} \\ -\frac{\partial p_1}{\partial \eta_0} & \frac{\partial p_0}{\partial \eta_0} \end{pmatrix} \boldsymbol{g}. \tag{9.14c}$$

The steps to transform Liouville's equation from a moving physical domain to the static reference domain is similar to what was done in Chapter 5. The resulting equations on the reference domain read

$$\frac{\partial \mathcal{J}}{\partial \tau} = \nabla_{\boldsymbol{\mu}} \cdot \tilde{\boldsymbol{v}}, \tag{9.15a}$$

$$\frac{\partial (\rho \mathcal{J})}{\partial \tau} + \nabla_{\boldsymbol{\mu}} \cdot \tilde{\boldsymbol{f}} = 0, \tag{9.15b}$$

with $\boldsymbol{v} = \frac{\partial \boldsymbol{x}}{\partial \tau}$ and

$$\tilde{\boldsymbol{f}} = \rho(\tilde{\boldsymbol{u}} - \tilde{\boldsymbol{v}}), \tag{9.15c}$$

$\boldsymbol{\mu} = (\boldsymbol{\xi}, \boldsymbol{\eta}) \in E^4$ and $\nabla_{\boldsymbol{\mu}} = (\frac{\partial}{\partial \boldsymbol{\xi}}, \frac{\partial}{\partial \boldsymbol{\eta}})$. Here, $\tilde{\boldsymbol{u}} = (\tilde{u}_0, \tilde{u}_1, \tilde{u}_2, \tilde{u}_3)$ denotes the transformed velocity of $\boldsymbol{u} = (u_0, u_1, u_2, u_3)$ and is given by

$$\begin{pmatrix} \tilde{u}_0 \\ \tilde{u}_1 \end{pmatrix} = \mathcal{J}_p \begin{pmatrix} \frac{\partial q_1}{\partial \xi_1} & -\frac{\partial q_0}{\partial \xi_1} \\ -\frac{\partial q_1}{\partial \xi_0} & \frac{\partial q_0}{\partial \xi_0} \end{pmatrix} \begin{pmatrix} u_0 \\ u_1 \end{pmatrix}, \tag{9.16a}$$

$$\begin{pmatrix} \tilde{u}_2 \\ \tilde{u}_3 \end{pmatrix} = \mathcal{J}_q \begin{pmatrix} \frac{\partial p_1}{\partial \eta_1} & -\frac{\partial p_0}{\partial \eta_1} \\ -\frac{\partial p_1}{\partial \eta_0} & \frac{\partial p_0}{\partial \eta_0} \end{pmatrix} \begin{pmatrix} u_2 \\ u_3 \end{pmatrix}, \tag{9.16b}$$

and similarly for $\tilde{\boldsymbol{v}}$.

As before, in the DG method we employ a tensor product of one-dimensional Lagrange polynomials ℓ_i of degree N, defined on Gauss-Legendre quadrature nodes over the interval $[0,1]$, to form basis functions ϕ_l. These basis functions are orthogonal on the reference domain E^4 with respect to the L_2-inner product, i.e.,

$$\int_{E^4} \phi_l \phi_k \, \mathrm{d}\boldsymbol{\mu} = W_k \delta_{lk}, \tag{9.17}$$

where W_k can be expressed in terms of a product of four one-dimensional quadrature weights and δ_{lk} the Kronecker delta. The expansion of ρ in terms of these basis functions reads

$$\rho_{\mathrm{h}}(z^t, \boldsymbol{\mu}) = \sum_{l=1}^{N_d} \rho_l^t \phi_l(\boldsymbol{\mu}), \tag{9.18}$$

where $N_d = (N+1)^4$ denotes the number of degrees of freedom and ρ_l^t the expansion coefficients.

The weak formulation of equation (9.15b) with test function ϕ_k reads

$$W_k \left((\rho_k \mathcal{J}_k)^{t+1} - (\rho_k \mathcal{J}_k)^t \right) = \int_{z^t}^{z^{t+1}} \left(\int_{E^4} \left(\nabla_{\boldsymbol{\mu}} \phi_k \right) \cdot \tilde{\boldsymbol{f}} \, \mathrm{d}\boldsymbol{\mu} - \int_{\partial E^4} \phi_k \tilde{\boldsymbol{F}} \cdot \hat{\boldsymbol{N}} \, \mathrm{d}\sigma \right) \mathrm{d}\tau, \tag{9.19}$$

with $\tilde{\boldsymbol{F}}$ the upwind numerical flux, given by (5.12), and $\hat{\boldsymbol{N}}$ the outward unit normal. By letting $k = 1, 2, \ldots, N_d$ we arrive at N_d equations for the expansion coefficients ρ_l^{t+1}. All the integrals in equation (9.19) are evaluated with $(N+1)$-point Gauss-Legendre quadrature. For the right-hand side of equation (9.19) we compute the solution ρ at intermediate levels in $[z^t, z^{t+1}]$ using the local ADER predictor described in Section 9.2.

The Jacobian $\mathcal{J}$ at a quadrature node $\boldsymbol{\mu}_k \in E^4$, i.e., $\mathcal{J}_k$, is updated by integrating the geometric conservation law (9.15a) as follows

$$\int_{z^t}^{z^{t+1}} \frac{\partial \mathcal{J}_k}{\partial \tau} \, \mathrm{d}\tau = \int_{z^t}^{z^{t+1}} \nabla_{\boldsymbol{\mu}} \cdot \tilde{\boldsymbol{v}} \big|_{\boldsymbol{\mu}=\boldsymbol{\mu}_k} \, \mathrm{d}\tau,$$

which is equivalent to

$$\mathcal{J}_k^{t+1} - \mathcal{J}_k^t = \int_{z^t}^{z^{t+1}} \nabla_{\boldsymbol{\mu}} \cdot \tilde{\boldsymbol{v}} \big|_{\boldsymbol{\mu}=\boldsymbol{\mu}_k} \, \mathrm{d}\tau. \tag{9.20}$$

The integral on the right-hand side of equation (9.20) is evaluated with the $(N+1)$-point Gauss-Legendre quadrature.

9.2 z-integration using local ADER predictor

To compute the right-hand side of equation (9.19) we employ the ADER approach to approximate the z-evolution locally on each element without considering neighbouring elements. This evolution is approximated by a Taylor expansion about the old level and subsequently applying the Cauchy-Kovalewski procedure [32, 39] to replace τ-derivatives with spatial derivatives using the governing equation.

The Taylor expansion up to degree M about the old level $\tau = z^t$, where the solution is known, on the reference domain reads

$$\rho(z^t + \tau, \boldsymbol{\mu}) \approx \sum_{k=0}^{M} \frac{1}{k!} \tau^k \frac{\partial^k \rho}{\partial \tau^k}(z^t, \boldsymbol{\mu}). \tag{9.21}$$

In the Cauchy-Kovalewski procedure we start with the advective form of the governing equation. Analogous to what was shown in Section 5.3 the advective form of equation (9.15b) can be written as

$$\frac{\partial \rho}{\partial \tau} = -\frac{1}{\mathcal{J}} (\tilde{\boldsymbol{u}} - \tilde{\boldsymbol{v}}) \cdot \nabla_{\boldsymbol{\mu}} \rho. \tag{9.22}$$

Recall that we consider only a piecewise constant refractive index field and thus the last two components of $\boldsymbol{u}$ are zero, as a consequence of relation (9.2). Moreover, the last two components of $\boldsymbol{v}$ are zero, as the momentum domain does not evolve as a function of z. This allows us to rewrite relation (9.22) as

$$\frac{\partial \rho}{\partial \tau} = -\frac{1}{\mathcal{J}} \begin{pmatrix} \tilde{u}_0 - \tilde{v}_0 \\ \tilde{u}_1 - \tilde{v}_1 \end{pmatrix} \cdot \begin{pmatrix} \frac{\partial \rho}{\partial \xi_0} \\ \frac{\partial \rho}{\partial \xi_1} \end{pmatrix}. \tag{9.23}$$

Note that in relation (9.23) the velocity depends on the matrix elements of $\frac{\partial \boldsymbol{q}}{\partial \boldsymbol{\xi}}$ by virtue of the definition of $\tilde{\boldsymbol{u}}$; cf. (9.16). Hence, the velocity field is significantly more complicated than in Section 5.3 where the velocity was linear in ξ.

Nevertheless, the Cauchy-Kovalewski procedure is used on equation (9.23). Expressions for the higher order τ-derivatives are found with the aid of a compute algebra programme and inserted into the Taylor expansion (9.21). To improve computational performance, the terms in the Taylor expansion are rearranged to

$$\rho(z^t + \tau, \boldsymbol{\mu}) \approx \sum_{i,j=0}^{M} \frac{1}{(i+j)!} \mathcal{C}_{ij}(\tau, \boldsymbol{\mu}) \frac{\partial^{i+j} \rho}{\partial \xi_0^i \partial \xi_1^j}(z^t, \boldsymbol{\mu}), \tag{9.24}$$

where $\mathcal{C}_{ij}(\tau, \boldsymbol{\mu})$ are found by the computer algebra programme.

Finally, the spatial derivatives in the Taylor expansion (9.24) can be computed from the expansion of ρ_h (9.18). Consequently, ρ can be computed using the local ADER predictor at the required Gauss-Legendre quadrature points to compute the right-hand side of equation (9.19).

9.3 Optical interfaces

Recall that at an optical interface a jump condition must be implemented. We use that light is either fully reflected or fully refracted, such that the jump condition is given by

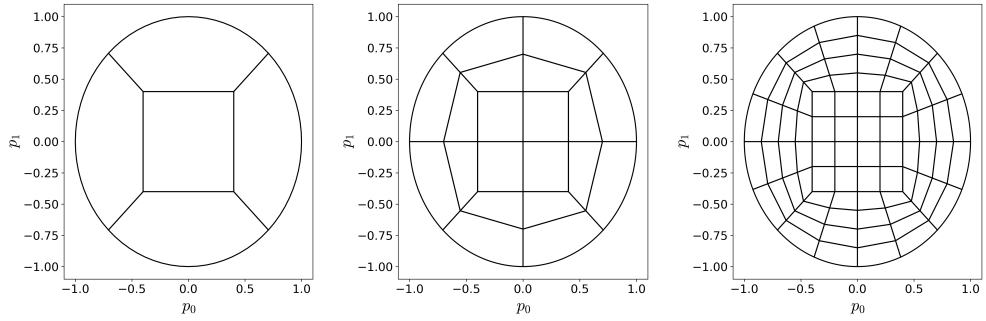
$$\rho_{\sigma(z^+)}(z^+, \boldsymbol{q}(z^+), \boldsymbol{p}(z^+)) = \rho_{\sigma(z^-)}(z^-, \boldsymbol{q}(z^-), \boldsymbol{p}(z^-)), \tag{9.25a}$$

where the full momentum vector $(\boldsymbol{p}, p_z)(z^+)$ is computed as

$$(\boldsymbol{p}, p_z)(z^+) = \mathcal{S}\Big((\boldsymbol{p}, p_z)(z^-); n_0, n_1, \vec{v}\Big) \text{ and } \operatorname{sgn} p_z(z^+) = \sigma(z^+) \tag{9.25b}$$

with $\mathcal{S}$ defined in relation (2.43).

As a first step towards three-dimensional optics, we implement the jump condition via straightforward interpolation. That is, at an optical interface the outgoing momentum $(\boldsymbol{p}, p_z)(z^+)$ is known and we compute its respective incident momentum $(\boldsymbol{p}, p_z)(z^-)$. Then, ρ is determined by interpolation at the optical interface, i.e., evaluating the expansion (9.18). To do this procedure efficiently at an optical interface, we need to be careful in how we represent the momentum domain. Our approach is discussed next.



(a) Mesh with $r_p = 0$.

(b) Mesh with $r_p = 1$.

(c) Mesh with $r_p = 2$.

Figure 9.2: Sequence of meshes for the momentum disk with $n = 1$. The square elements in the center cover the area $[-0.4, 0.4] \times [-0.4, 0.4]$.

The momentum domain is described by a disk $|\boldsymbol{p}| < n$. The coarsest mesh, refinement $r_p = 0$, for the momentum disk can be seen in the first panel of Figure 9.2. In the second and third panel of the figure we show the meshes that result when we refine the mesh, where in each refinement the number of elements is quadrupled. The outer ring of elements of each mesh consists of curvilinear elements, whereas towards the center bilinear elements are used. The center area $[-0.4, 0.4] \times [-0.4, 0.4]$ is covered by square elements.

The momentum of a light ray changes discontinuously at an optical interface, as described by (9.25b). In the discretisation of the jump condition (9.25) this means that we need to do a point search over the momentum disk. The naive way of searching would be to loop over every element in the mesh and then to compute the reference coordinates that correspond to the point by inverting the mapping associated to the element. Then the correct element is found if these coordinates belong to $E^2 = [0, 1]^2$.

A better way is to exploit properties of the mesh. The sequence of meshes that are generated have a particular structure that makes this point search much more efficient. Specifically, the meshes consist of a square area with uniform mesh spacing and four non-square regions which are related to each other by a simple rotation about the origin. This is most easily seen in the coarsest mesh, left panel in Figure 9.2. Thus if a point is inside one of the four regions, then we can always relate the search to just one single region, e.g., the top one.

Searching inside the top region is easily accommodated by the fact that straight line segments connect from the circle to the square block. These line segments are depicted in green in Figure 9.3. A binary search is used to find the two line segments between which the point is located. Note that we only have to know on which side the point is located for each line segment, which can be efficiently computed. The binary search reduces the search space from 4^{r_p} elements to 2^{r_p} elements. Since all these elements are bilinear except the one element that touches the circle, it is again efficient to use straight line segments to search for the correct element. The line segments used are depicted in red in Figure 9.3. Again a binary search is employed. Only in case the curvilinear element needs to be checked, we have to compute the reference coordinates that correspond to the point and then check if these reference coordinates belong to $E^2 = [0, 1]^2$.

The complete point search is summarised as follows. The point search starts by checking whether the point is inside the square area. If it is, then due to the uniform mesh spacing the corresponding element can be found in constant time. Otherwise, we determine in which of the four non-square regions the point lies. The region is searched by employing a binary search

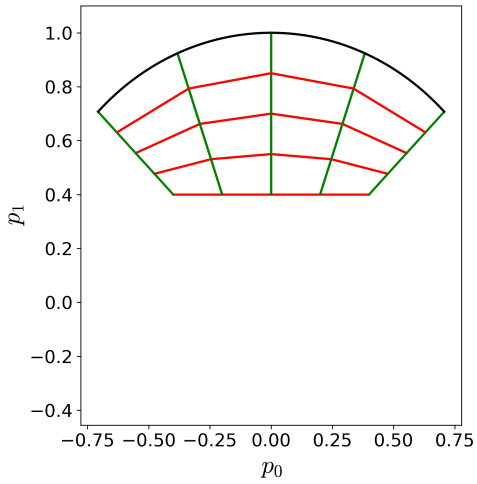


Figure 9.3: Line segments (green and red) used for two binary searches on the momentum disk with $r_p = 2$.

over the 2^{r_p} green line segments (we ignore the first one), which can be done in $\mathcal{O}(\log(2^{r_p}))$ time. Finally another binary search is used over the remaining 2^{r_p} red line segments, which can again be done in $\mathcal{O}(\log(2^{r_p}))$ time. We conclude that in total, searching for a point can be done in $\mathcal{O}(\log(2^{r_p}))$ time.

9.4 Results

We discuss two examples, a tilted cylinder and a compound parabolic concentrator. To solve Liouville's equation, we fix some parameters. Namely, we take $N = M$ in the Taylor expansion (9.24). Moreover, we use the CFL condition (5.33) with CFL = 0.9.

The ALE-ADER-DG scheme was implemented in C++. In the implementation we use a one-touch policy of ρ for computing the integrals in the weak formulation. This means that the local ADER predictor on an element only has to be computed once during a step. Solving Liouville's equation on a moving four-dimensional phase space is computationally costly. Hence, for the simulations presented in this chapter we have run the code on a single node of a cluster, that has a dual socket of Intel Xeon Platinum 8260 CPU @ 2.40GHz each having 24 cores for a total of 48 cores. The code has been parallelised with OpenMP [26] and we employ 48 threads in each simulation. The parallelisation strategy and other implementation details are discussed in Appendix D.

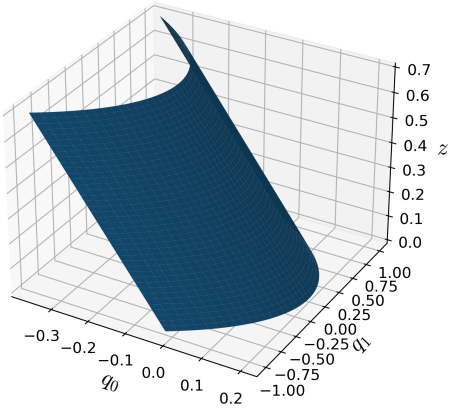


Figure 9.4: The surface of the tilted cylinder that lies in $\boldsymbol{q} \in [-1,1]^2$ and $z \in [0, 0.7]$.

9.4.1 Tilted cylinder

To test convergence, we consider the (partial) surface of a tilted cylinder to separate two different media of refractive indices $n_0 = 1$ and $n_1 = 1.5$. This surface of the tilted cylinder satisfies the equation

$$[q_0 - (b + az)]^2 + q_1^2 = R^2, \tag{9.26}$$

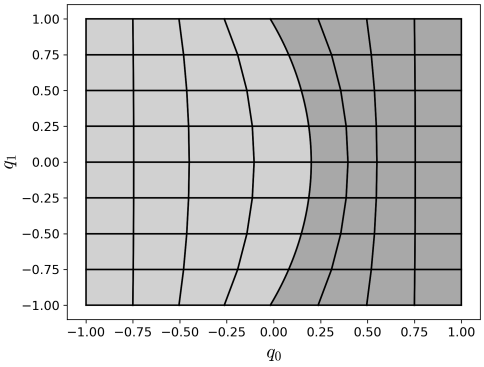
where $(b + az, 0, z)$ are the points along the axis of the cylinder and R denotes the radius. As parameters we take $R = 2.4$, $b = -2.2$ and $a = -0.5$. The two-dimensional position domain we consider is given by $\boldsymbol{q} \in [-1,1]^2$. The surface of the tilted cylinder over $z \in [0, 0.7]$ is shown in Figure 9.4.

The mesh velocity at the optical interface $\boldsymbol{q} = \boldsymbol{Q}(z)$, with $\boldsymbol{Q}(z)$ determined from (9.26), is given by $\frac{\mathrm{d}\boldsymbol{Q}}{\mathrm{d}z} = (a, 0)$. The mesh velocity at an arbitrary position $\boldsymbol{q}$ is prescribed by a piecewise linear interpolation along the q_0-axis, between the optical interface and the edge $q_0 = -1$ for $n(z, \boldsymbol{q}) = n_0$ and the edge $q_0 = 1$ for $n(z, \boldsymbol{q}) = n_1$.

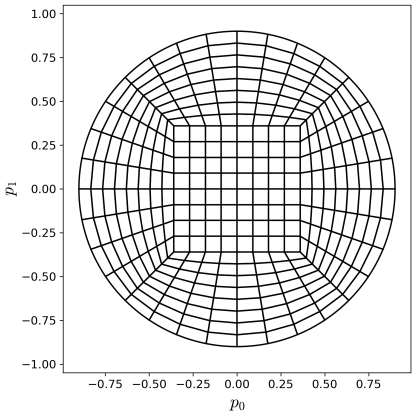
For the convergence test we take the initial condition

$$\rho_0(\boldsymbol{q}, \boldsymbol{p}) = \varphi_{m,k}\left(\frac{q_0 - q_{0,\mathrm{c}}}{\lambda_{q_0}}\right)\varphi_{m,k}\left(\frac{q_1}{\lambda_{q_1}}\right)\varphi_{m,k}\left(\frac{p_0 - p_{0,\mathrm{c}}}{\lambda_{p_0}}\right)\varphi_{m,k}\left(\frac{p_1}{\lambda_{p_1}}\right), \tag{9.27}$$

where $\varphi_{m,k}$ is defined in (4.66) and with parameters $m = 10$, $k = 2$, $q_{0,\mathrm{c}} = -0.4$, $p_{0,\mathrm{c}} = 0.4$, $\lambda_{q_0} = 0.4$, $\lambda_{q_1} = 0.5$ and $\lambda_{p_0} = \lambda_{p_1} = 0.4$. The initial condition is chosen such that light with non-zero basic luminance at $z = Z_{\mathrm{target}} = 0.7$ has either been refracted at the optical interface or has propagated freely. The maximum momentum is limited to $0.9n(z, \boldsymbol{q})$, i.e., the momentum domain



(a) Position mesh at $z = 0$ with left of the interface $n = n_0$ (light gray) and right $n = n_1$ (dark gray).



(b) Momentum mesh for $n = n_0$.

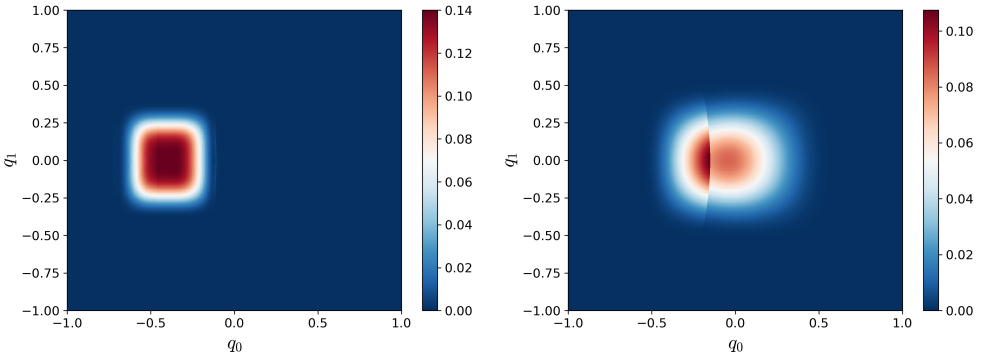
Figure 9.5: The initial position mesh and the momentum mesh for $n = n_0$ describing the tensor-product mesh for the tilted cylinder example with $N = 5$.

satisfies $|\boldsymbol{p}| \leq 0.9n(z, \boldsymbol{q})$. The resulting numerical solution is computed for $N = 5$ for a coarse mesh. The tensor-product mesh consists of 64 position elements and 320 momentum elements for a total of $K = 20480$ elements. The position mesh at $z = 0$ and the momentum mesh for $n = n_0$ are shown in Figure 9.5. The illuminance associated with the initial and resulting numerical solutions are shown in Figure 9.6 with the illuminance $E(z, \boldsymbol{q})$ computed according to relation (2.45). The illuminance at $z = 0.7$ features a jump at the optical interface.

Next, the convergence in the basic luminance ρ is studied. With the chosen initial condition, the exact solution is determined using the method of characteristics where a light ray with non-zero basic luminance is either refracted at the optical interface or can be directly traced back to $z = 0$. For a refinement level r, the position mesh consists of 2^r by 2^r elements with half of them left of the interface and half of them on the right of the interface. The momentum mesh consists of $20 \cdot 4^r$ elements.

The convergence data is shown in Table 9.1 for the L_2-norm in the error of ρ, the degrees of freedoms (DoFs) and the computation time. For $N = 2$ and $N = 3$ the expected convergence rate of $N + 1$ is observed. For $N > 3$ the convergence rate at the finest computed level is close to N and another level of computation might verify the expected convergence rate. Due to the large computation time the computation has not been carried out.

From the results it is clear that the higher-degree polynomials are more efficient in getting to a low error, than the lower-degree polynomials. For example, $N = 7$ with $r = 2$ achieves a factor 10 lower error than $N = 4$ with $r =$



(a) $z = 0$. **(b)** $z = 0.7$.

Figure 9.6: Distributions of the illuminance $E(z, \boldsymbol{q})$ for the tilted cylinder example computed with the $N = 5$ ALE-ADER-DG scheme.

3 in less than half of the computation time. We remark that the computation time t_{CPU} increases roughly by a factor 32 when refining the tensor-product mesh once, as observed in the table. This is due to the total number of elements increasing by a factor 16 and the number of Δz-steps doubling, upon one refinement of the mesh.

Finally, we compare solving Liouville's equation with the DG scheme to quasi-Monte Carlo (QMC) ray tracing for computing the illuminance. For QMC ray tracing the position domain is covered with a mesh of 100 by 100 straight-sided quadrilateral elements. A coarser mesh with 20 by 20 elements is shown in Figure 9.7. Note that the quadrilateral elements are aligned with the optical interface, which is necessary for a fair comparison between both methods.

In the QMC ray tracing method finding the correct bin/element is no longer a simple task as in the two-dimensional optics case. Providing an efficient search algorithm is important. If one would simply loop over all elements and check if a point lies inside the element, this would make the search algorithm the most expensive part of the algorithm by far.

The mesh consists of straight-sided quadrilaterals, but the position domain is still a square. A very efficient search algorithm is used by overlaying the position domain with a uniform background mesh. For each square element in the background mesh we compute the overlapping elements of the actual mesh and store the IDs. Then, for a sufficiently fine background mesh each background element is associated with a maximum of four quadrilateral elements. Searching on the uniform background mesh is done in constant time. This results in a maximum of four quadrilateral elements to be searched.

Table 9.1: Convergence data for the tilted cylinder example with the ALE-ADER-DG scheme.

r	L_2	$\mathcal{O}(L_2)$	DoFs	t_{CPU} [s]
			$N = 1$	
0	7.52e-02		2.05e+04	0.196
1	3.80e-02	0.98	3.28e+05	1.945
2	1.58e-02	1.26	5.24e+06	38.467
3	6.49e-03	1.29	8.39e+07	1011.788
			$N = 2$	
0	4.36e-02		1.04e+05	0.9787
1	1.52e-02	1.52	1.66e+06	11.2904
2	5.30e-03	1.52	2.65e+07	254.8082
3	6.32e-04	3.07	4.25e+08	6551.5381
			$N = 3$	
0	2.84e-02		3.28e+05	3.3276
1	9.59e-03	1.57	5.24e+06	55.5880
2	1.14e-03	3.08	8.39e+07	1285.1046
3	5.89e-05	4.27	1.34e+09	37411.2721
			$N = 4$	
0	1.73e-02		8.00e+05	12.3650
1	3.63e-03	2.25	1.28e+07	194.4017
2	2.78e-04	3.71	2.05e+08	4904.5057
3	1.56e-05	4.16	3.28e+09	146363.6469
			$N = 5$	
0	1.50e-02		1.66e+06	30.9944
1	1.58e-03	3.25	2.65e+07	508.3959
2	4.20e-05	5.24	4.25e+08	12367.5604
			$N = 6$	
0	1.03e-02		3.07e+06	69.6762
1	7.19e-04	3.84	4.92e+07	1206.9368
2	8.18e-06	6.46	7.87e+08	29712.0880
			$N = 7$	
0	5.31e-03		5.24e+06	151.5022
1	2.38e-04	4.48	8.39e+07	2652.7242
2	1.90e-06	6.97	1.34e+09	64157.2102

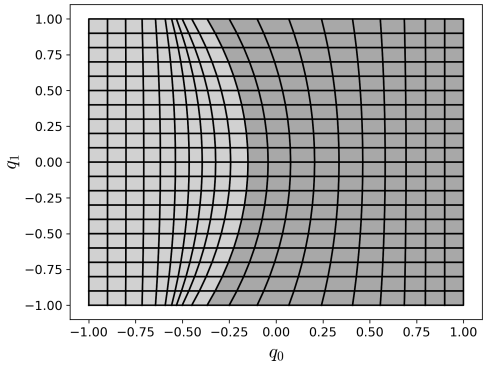


Figure 9.7: A coarse position mesh of 20 by 20 straight-sided quadrilateral elements used for QMC ray tracing. Light gray represents $n = n_0$ and dark gray represents $n = n_1$.

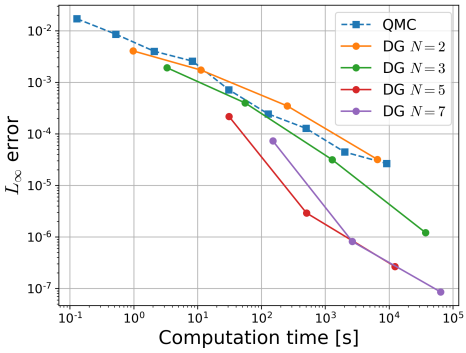


Figure 9.8: Comparison between quasi-Monte Carlo (QMC) ray tracing and the ALE-ADER-DG scheme (DG) for the tilted cylinder.

For the cases considered here, the search algorithm takes roughly the same time as the ray tracing part.

To measure the performance we compute the error as the L_∞-norm of the average illuminance. The comparison of the ALE-ADER-DG scheme and the QMC ray tracing method is shown in Figure 9.8. For QMC ray tracing the initial point corresponds to 10^6 rays and each subsequent point quadruples the number of rays, so that the final point corresponds to $1.638 \cdot 10^{10}$ rays. From the figure one can observe that the ALE-ADER-DG scheme achieves higher accuracy for $N \geq 3$ compared to QMC ray tracing using the same computation time. Moreover, at roughly 10 minutes computation time the ALE-ADER-DG scheme with $N = 5$ achieves a 44 times lower error than QMC ray tracing. The ALE-ADER-DG scheme is more efficient in computing high accuracy solutions, as is evident from its faster convergence.

9.4.2 Compound parabolic concentrator

For the second example we consider a compound parabolic concentrator (CPC). The dielectric total internal reflection concentrator from Chapters 5-6 is a variation of the CPC. Instead of using a dielectric, the CPC is used in air with $n = 1$ and the side walls of a CPC are coated with a reflective material, which we assume to be perfectly reflective. A CPC for two-dimensional optics is shown in Figure 9.9. It is designed such that all light entering from the top at $z = Z$ within a certain acceptance angle θ is accepted, and light with a larger angle is rejected; see [18]. The rejected light leaves the top again, via multiple reflections. The accepted light is concentrated onto the plane at $z = 0$. The accepted light takes an angle from $[-\theta, \theta]$ at $z = Z$ and leaves at an angle in $[-\frac{\pi}{2}, \frac{\pi}{2}]$ at $z = 0$. In total, the spatial distribution of light is squeezed while the angular distribution is expanded.

The right wall for the two-dimensional CPC is given by $q = Q_\mathrm{r}(z)$ with Q_r given by [90]

$$Q_\mathrm{r}(z) = \frac{a_1 + b_1 z + \sqrt{a_2 + b_2 z}}{2\cos^2\theta} \tag{9.28a}$$

with the coefficients given by

$$a_1 = d\,(\cos(2\theta) - 3 - 4\sin\theta), \tag{9.28b}$$
$$a_2 = -8 d a_1, \tag{9.28c}$$
$$b_1 = -\sin(2\theta), \tag{9.28d}$$
$$b_2 = 8d\,(2\cos\theta + \sin(2\theta)), \tag{9.28e}$$

and d denotes the half-width at $z = 0$ (the exit). The left wall is simply given by $q = -Q_\mathrm{r}(z)$. The optic has a length Z that is given by

$$Z = d\frac{(1 + \sin\theta)\cos\theta}{\sin^2\theta}, \tag{9.29}$$

and the half-width at $z = Z$ is given by $d/\sin\theta$. Note that the velocity at the optical interface is determined from (9.28) as $\frac{\mathrm{d}Q_\mathrm{r}}{\mathrm{d}z}$.

For three-dimensional optics the CPC is rotated about the z-axis. In 3D the concentrator no longer has a sharp cut-off acceptance angle; see Chaves [18] for more details. Here, we will reverse the direction of light so that light enters at $z = 0$ and leaves at $z = Z$. The CPC will then cause the spatial distribution of light to be expanded and the angular distribution to be squeezed.

At $z = 0$ we consider the initial basic luminance distribution to be Lambertian, i.e., we take $\rho_0(\boldsymbol{q}, \boldsymbol{p}) = 1$. The parameters for the CPC are given by

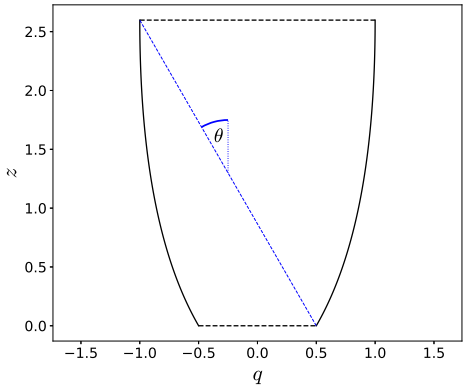
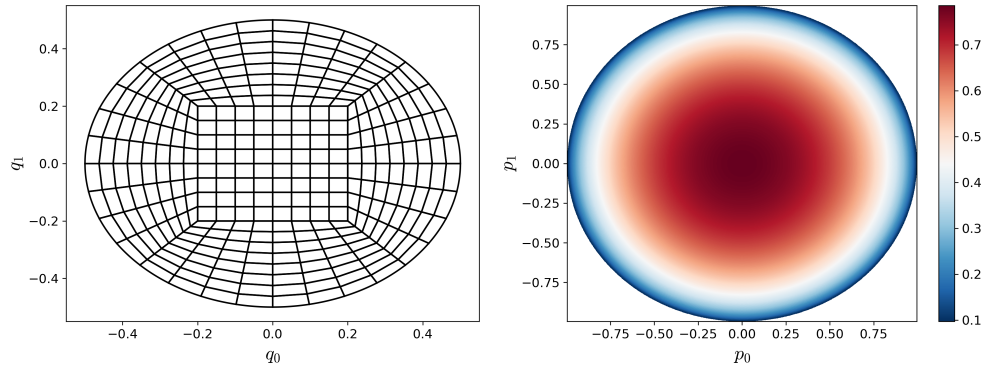


Figure 9.9: The two-dimensional compound parabolic concentrator for $\theta = 30\,\text{deg}$, $d = 0.5$ with optic length $Z \approx 2.598$.

$\theta = 30\,\text{deg}$ and $d = 0.5$. As we consider the walls to be fully reflective we replace $\mathcal{S}$ in the jump condition (9.25) with $\mathcal{S}_\text{R}$. The maximum momentum is limited to $n\sin(85\,\text{deg})$. The initial position mesh and luminous intensity I are shown in Figure 9.10, where the latter is computed from its definition (2.46). In the last two panels of the figure, one can also see the position mesh and luminous intensity at $z = Z$ computed with the $N = 3$ ALE-ADER-DG scheme at a refinement level $r = 2$. The refinement level $r = 2$ means that both the position and momentum mesh contain $20 \cdot 4^r = 320$ elements, for a total of $K = 102400$ elements. The luminous intensity at $z = Z$ shows a rotationally symmetric profile with a sharp jump in intensity at $|\boldsymbol{p}| = 0.5 = n\sin(\theta)$, as expected.

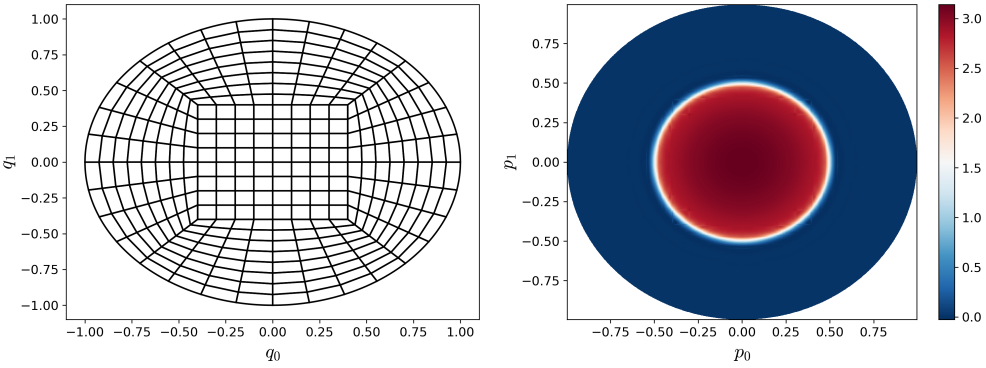
Next, we compare the luminous intensity distributions for different parameters of the ALE-ADER-DG scheme. The distributions are shown in Figure 9.11. In the first two panels for $N = 2$ one can observe that the luminous intensity is not well resolved since the mesh is visible in the solution. A similar effect is just slightly visible for the $N = 5$ with $r = 1$ solution.

Finally, a cross section of the luminous intensity along $p_1 = 0$ is presented in Figure 9.12. In the figure one can see that the $N = 2$ with $r = 1$ profile shows a less sharp jump in the intensity around $|p_0| = 0.5$, compared to the other profiles. The $N = 5$ with $r = 1$ and $N = 3$ with $r = 2$ profiles agree pretty well, except for some oscillations in the $N = 5$ solution. Note that there is some undershoot in the luminous intensity, which should be a non-negative quantity.



(a) Position mesh at $z = 0$.

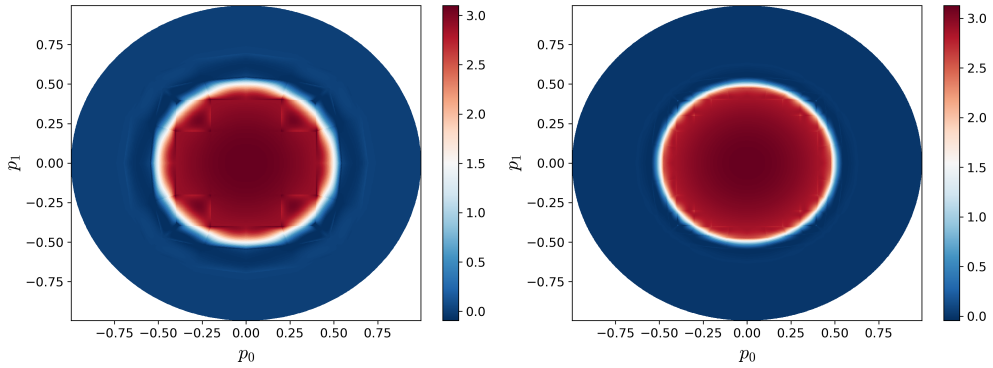
(b) Luminous intensity I at $z = 0$.



(c) Position mesh at $z = Z$.

(d) Luminous intensity I at $z = Z$.

Figure 9.10: The position mesh and luminous intensity for the CPC computed with the $N = 3$ ALE-ADER-DG scheme.



(a) $N = 2$ and $r = 1$.

(b) $N = 2$ and $r = 2$.

(c) $N = 3$ and $r = 2$.

(d) $N = 5$ and $r = 1$.

Figure 9.11: The luminous intensity distributions at $z = Z$ for the CPC computed with the ALE-ADER-DG scheme.

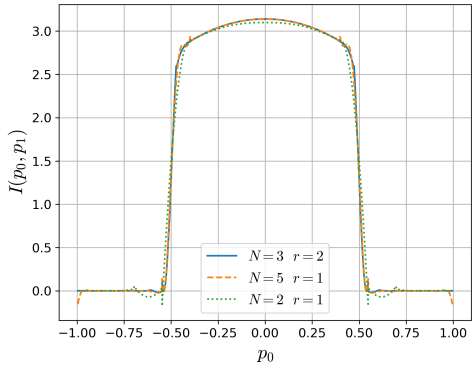


Figure 9.12: A cross section at $p_1 = 0$ for the luminous intensity distributions at $z = Z$ for the CPC computed with the ALE-ADER-DG scheme.

9.5 Concluding remarks

We have solved Liouville's equation for three-dimensional optical systems on a moving four-dimensional phase space mesh, with curvilinear elements, using the ALE-ADER-DG scheme. The ALE-ADER-DG scheme shows favourable properties, such as high order convergence for smooth solutions. In an example, we also compared the performance of the scheme to quasi-Monte Carlo ray tracing for computing the illuminance. Despite the high dimensionality of Liouville's equation, the ALE-ADER-DG scheme can still converge faster to high accuracy solutions. Moreover, we observed that the $N = 5$ ALE-ADER-DG scheme achieves a 100 times lower error than QMC ray tracing in roughly equal amounts of computation time. The ALE-ADER-DG scheme was also used to compute luminous intensity profiles for the CPC.

The scheme is a first step towards solving Liouville's equation on a four-dimensional phase space mesh for geometrical optics. There are some improvements that can still be made. First, the optical interface is not discretised in an energy-conserving manner. Second, the luminous intensity for the CPC showed undershoots and oscillations, which can be mitigated by extending the modal filter and limiter from Chapter 8 to four-dimensional phase space. Third, the performance can be improved by using fewer moving elements and by extending the hybrid semi-Lagrangian DG and ADER-DG solver from Chapter 6 to four-dimensional phase space.

Chapter 10

Conclusions and future research

10.1 Summary and conclusions

The aim of this thesis was to develop and apply discontinuous Galerkin methods to solve Liouville's equation for geometrical optics. First, Liouville's equation was derived from a Hamiltonian formulation of the propagation of light rays and conservation principles of non-imaging optics. Liouville's equation describes the transport of the basic luminance through an optical system, on phase space. At an optical interface, a jump condition describes how the basic luminance is redistributed in terms of non-local boundary conditions. The DGSEM was first applied to solve Liouville's equation for two-dimensional optics. The discretisation of the jump condition for a flat optical interface is obtained by using a least-squares matching procedure, together with the geometric connectivity and local energy balances.

In the ADER-DG solver curved optical interfaces are dealt with by aligning the phase space mesh with an optical interface, through the use of a moving mesh. The moving mesh method alone is not sufficient, as for certain optical systems it can lead to an increasingly smaller mesh spacing which causes an ever decreasing stepsize due to a CFL condition. The sub-cell interface method was introduced to resolve this particular issue. This method and the ADER-DG method are both fully discrete explicit methods. Mesh refinement is used to deal with large deformations of the mesh. The discretisation of the jump condition is extended to arbitrary curved optical interfaces by formulating and proving local energy balances. These local energy balances are used in a least-squares matching procedure. In the ADER-DG solver we found that allowing only elements adjacent to the optical interface to move, leads to a more efficient scheme. For the ADER-DG scheme an arbitrary order of accuracy for smooth solutions can be chosen both in space and the evolution

coordinate z. The expected order of convergence was verified in an example. The ADER-DG scheme also proved to be more efficient in computing the illuminance compared to quasi-Monte Carlo ray tracing in the considered examples. For the meniscus lens example, an error that is two orders of magnitude lower is achieved compared to QMC ray tracing within 10 seconds of computation time. Alternatively, for an error of roughly 10^{-6} the ADER-DG scheme is a factor 100 times faster than QMC ray tracing. This speed-up increases for lower errors as the ADER-DG scheme converges faster to high accuracy.

A novel hybrid semi-Lagrangian DG and ADER-DG solver on a moving mesh with local time stepping was introduced. This hybrid solver resolves inefficiencies in the pure ADER-DG solver caused by the unavoidable very small elements, a large mesh velocity and free-propagation over a large z-interval. Away from optical interfaces a semi-Lagrangian DG method is used, whereas close to an optical interface we use the ADER-DG scheme on a moving mesh. Hanging nodes in the z-direction are introduced by the use of local time stepping. The coupling between semi-Lagrangian DG and ADER-DG elements, and between multiple ADER-DG elements, in the presence of hanging nodes, is dealt with in an energy-conserving manner. Local time stepping severely diminishes the effect of stepsize reduction and the semi-Lagrangian DG scheme allows large steps to be taken in regions without optical interfaces. Numerical experiments indicate the increased performance of the hybrid solver over the pure ADER-DG scheme. In the meniscus lens example the hybrid solver is faster by a factor of roughly 1.6 to 10 for computation times up to 4 minutes, whilst achieving the same accuracy! Moreover, in the design of the hybrid solver the type of jump condition only impacts the ADER-DG elements, so that including Fresnel reflections or surface scattering only requires a modification of the ADER-DG elements at optical interfaces.

Three different solvers have been discussed, the DGSEM, the ADER-DG solver and the hybrid semi-Lagrangian DG and ADER-DG solver. For the latter two, we primarily focused on piecewise constant refractive index fields. For these type of optics, the performance of the hybrid solver clearly demonstrates that it is the best way to efficiently solve Liouville's equation. If instead one is interested in smoothly varying refractive index fields, then the DGSEM can be used straightforwardly. The ADER-DG scheme could also be adapted to this case, however, the computation of the temporal Taylor expansion would become more expensive.

The inclusion of Fresnel reflections at an optical interface in the DG methods is achieved by employing newly derived energy balances and by

a modification of the least-squares matching procedure. This results in energy-conserving numerical fluxes at any optical interface. To solve Liouville's equation for the lens plate we needed to iterate between solving for forward- and backward-propagating light. A modal filter and limiter were applied to deal with discontinuities in the solution, which resulted in the solutions per iteration showing no practical undershoot and overshoot, nor large unphysical oscillations. A parameter study for the lens plate showcases the effects of the individual parameters, which can be used to find optimal values for design.

Finally, we made a first step towards solving Liouville's equation for three-dimensional optics on a moving four-dimensional phase space with curvilinear elements. The mesh is a tensor product between a position mesh and a momentum mesh, where both can feature curvilinear elements to accurately capture curved boundaries. The developed ADER-DG solver features an arbitrary order of accuracy for smooth solutions in both space and the evolution coordinate z. A convergence test showcased the high order of convergence in an example. Despite the high dimensionality of the problem, the ADER-DG scheme proved to be more efficient in computing the illuminance compared to QMC ray tracing in an example. For instance, for roughly 10 minutes computation time the ADER-DG scheme with $N = 5$ achieves a 100 times lower error than QMC ray tracing. Moreover, the ADER-DG scheme converges faster to high accuracy.

10.2 Future research

In some of the earlier chapters ideas for future research were already mentioned. For instance, a jump condition for surface scattering can be described, in which the basic luminance for an outgoing direction is equal to an integral of the basic luminance for a range of incident light rays multiplied by a probability density function. Including surface scattering in a DG method for Liouville's equation has a completely different effect than in QMC ray tracing methods. In QMC ray tracing methods a single incident light ray would after diffuse reflection need to be represented by a finite number of outgoing rays that would then accurately capture the effect of the probability function. This can severely increase the number of rays required to accurately approximate the target distribution. In Liouville's equation instead, we just see the basic luminance being redistributed, as we already solve for (almost) all light. Due to the split in forward- and backward-propagating light we would still need to iterate.

For Fresnel reflections we always assumed that light was unpolarised, for which the reflection coefficient is then the average of the parallel and

perpendicular reflection coefficients. As the reflection coefficients depend on the polarisation state, light will be partially polarised after interacting with an optical interface. This effect could be properly incorporated into the solver by keeping track of the basic luminance for either polarisation state.

The theory of optimal control for PDEs can be used to perform freeform optical design for non-zero étendue optical systems. Van Lith already made a first step in this direction for a single control parameter in [90]. The new solvers developed in this thesis could readily be used to speed-up the optimisation procedure.

In this thesis an ADER-DG method on a moving mesh was described for three-dimensional optics. To accommodate more general optical systems, the solver needs to be extended with a mesh refinement procedure and a sub-cell interface method analogously to the two-dimensional optics situation in Chapter 5. Moreover, the computation of energy-conserving fluxes at an optical interface needs to be extended to three-dimensional optics.

The hybrid SLDG and ADER-DG solver with local time stepping can also be extended to three-dimensional optics. There, the position mesh can locally, away from optical interfaces, be covered with square position elements. Then, for any SLDG element information only has to be taken from four elements for a fixed momentum $\boldsymbol{p}$.

In the hybrid solver local time stepping was used in a clustered way, where the local stepsizes only depend on the position and the maximum velocity. As an alternative, letting the local stepsizes depend on the momentum could be worthwhile. Since the velocity field in Liouville's equation rapidly increases for large absolute momentum values approaching n, this could result in significant speed-ups.

A straightforward extension to speed-up the hybrid solver could be made by adding shared-memory parallelisation. This can be done by using OpenMP [26] or Intel's Threading Building Blocks [78]. Parallelisation can be achieved by parallelising the different regions the solver defines. First, the update to the semi-Lagrangian DG region can be parallelised across multiple threads. Then, in a loop over each ADER-DG region that encloses an optical interface, each update to the contained elements can be parallelised.

In this thesis the z-coordinate was used as an evolution coordinate, which resulted in some complications when dealing with the geometry of optical interfaces. An alternative idea would be to directly discretise both z and phase space at the same time using a discontinuous Galerkin method, which can be interpreted as a space-time discontinuous Galerkin method as described by van der Vegt et al. in [86]. In this method the z-coordinate would be treated on an equal footing as the other position coordinates. The advantages are

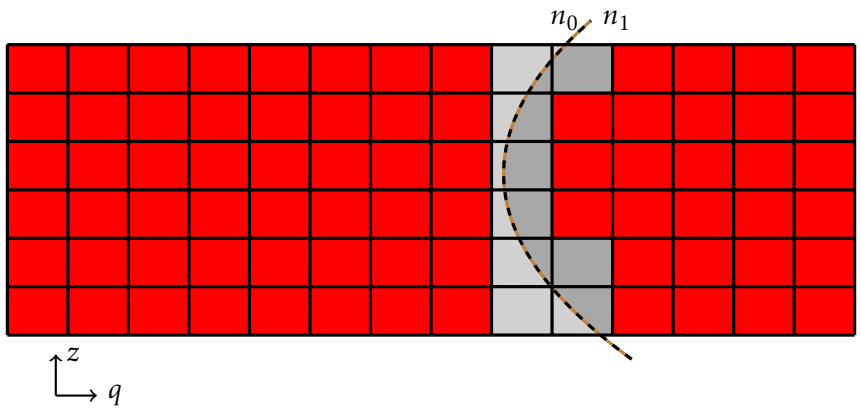


Figure 10.1: Sketch of the qz-grid with an optical interface (brown-black dashed line). In each red block the solution is straightforward and can be transformed to boundary data. In the remaining gray region the degrees of freedoms are placed.

that one could avoid the strict CFL condition and one does not need to iterate between forward- and backward-propagating light, as was needed in the lens plate example. The downsides are that one ends up with a large linear system that needs to be solved, and that one would have to discretise an additional coordinate, thus, significantly increasing the degrees of freedoms. For piecewise constant refractive index fields the number of degrees of freedoms can be significantly reduced by locally inserting the exact evolution, similar to how the semi-Lagrangian DG method was used in the hybrid solver. Specifically, for the red blocks shown in Figure 10.1 the solution can be easily computed from the inflow boundary conditions on the boundary of each red block. Moreover, for a region comprised of red blocks one only needs to compute the inflow boundary conditions on the boundary of such a region, i.e., only degrees of freedoms on the boundary of such a region are required. This can effectively reduce the degrees of freedoms required to solve Liouville's equation.

Appendix A

Analytical inverse least-squares matrix with constraint

In Chapter 4.3 the matrix $\boldsymbol{A} \in \mathbb{R}^{(N+2)\times(N+2)}$ defined in equation (4.49), arises from a least-squares problem with a linear constraint. The matrix $\boldsymbol{A}$ has a structure that allows for analytical computation of its determinant and inverse. Omitting the superscripts from (4.49) the matrix $\boldsymbol{A}$ reads

$$\boldsymbol{A} = \begin{pmatrix} \operatorname{diag}(\boldsymbol{w}) & \boldsymbol{\alpha} \circ \boldsymbol{w} \\ (\boldsymbol{\alpha} \circ \boldsymbol{w})^{\mathrm{T}} & 0 \end{pmatrix}, \tag{A.1}$$

which can be rewritten as

$$\boldsymbol{A} = \operatorname{diag}(\bar{\boldsymbol{w}})\, \boldsymbol{Q} \tag{A.2a}$$

with $\bar{\boldsymbol{w}} = (w_0, w_1, \ldots, w_N, 1)$ and with $\boldsymbol{Q}$ defined by

$$\boldsymbol{Q} = \begin{pmatrix} \boldsymbol{I} & \boldsymbol{\alpha} \\ (\boldsymbol{\alpha} \circ \boldsymbol{w})^{\mathrm{T}} & 0 \end{pmatrix}, \tag{A.2b}$$

The determinant of $\boldsymbol{A}$ is equal to product of the determinant of $\operatorname{diag}(\bar{\boldsymbol{w}})$ and the determinant of $\boldsymbol{Q}$, i.e., $\det(\boldsymbol{A}) = \det(\operatorname{diag}(\bar{\boldsymbol{w}}))\det(\boldsymbol{Q})$. The determinant of

$\boldsymbol{Q}$ can be found by Laplace (cofactor) expansion along the first row, i.e.,

$$\begin{aligned}\det(\boldsymbol{Q}) = &\begin{vmatrix} 1 & 0 & \dots & \alpha_1 \\ 0 & 1 & \dots & \alpha_2 \\ \vdots & & \ddots & \vdots \\ \alpha_1 w_1 & \alpha_2 w_2 & \dots & 0 \end{vmatrix} \\ &+ (-1)^{1+N+2}\alpha_0 \begin{vmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \vdots & & & \ddots & \vdots \\ \alpha_0 w_0 & \alpha_1 w_1 & \alpha_2 w_2 & \dots & \alpha_N w_N \end{vmatrix}.\end{aligned} \tag{A.3}$$

The second term on the right-hand side can be easily evaluated using a cofactor expansion along the first column, since it has all zeros except for $\alpha_0 w_0$ and the remaining minor is the determinant of an identity matrix. Therefore, we obtain

$$\begin{aligned}(-1)^{1+N+2}\alpha_0 \begin{vmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \vdots & & & \ddots & \vdots \\ \alpha_0 w_0 & \alpha_1 w_1 & \alpha_2 w_2 & \dots & \alpha_N w_N \end{vmatrix} &= (-1)^{1+N+2}(-1)^{1+N+1}\alpha_0^2 w_0 \\ &= -\alpha_0^2 w_0.\end{aligned} \tag{A.4}$$

The first term on the right-hand side of (A.3) can again be expanded along the first row, resulting in

$$\begin{aligned}\begin{vmatrix} 1 & 0 & \dots & \alpha_1 \\ 0 & 1 & \dots & \alpha_2 \\ \vdots & & \ddots & \vdots \\ \alpha_1 w_1 & \alpha_2 w_2 & \dots & 0 \end{vmatrix} &= \begin{vmatrix} 1 & 0 & \dots & \alpha_2 \\ 0 & 1 & \dots & \alpha_3 \\ \vdots & & \ddots & \vdots \\ \alpha_2 w_2 & \alpha_3 w_3 & \dots & 0 \end{vmatrix} \\ &\quad + (-1)^{1+N+1}\alpha_1 \begin{vmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \vdots & & & \ddots & \vdots \\ \alpha_1 w_1 & \alpha_2 w_2 & \alpha_3 w_3 & \dots & \alpha_N w_N \end{vmatrix} \\ &= \begin{vmatrix} 1 & 0 & \dots & \alpha_2 \\ 0 & 1 & \dots & \alpha_3 \\ \vdots & & \ddots & \vdots \\ \alpha_2 w_2 & \alpha_3 w_3 & \dots & 0 \end{vmatrix} - \alpha_1^2 w_1.\end{aligned}$$

Repeating these steps we obtain the expression for r, defined in (4.51), i.e.,

$$r = \det(\boldsymbol{Q}) = -\sum_{i=0}^{N} \alpha_i^2 w_i, \tag{A.5}$$

so that

$$\det(\boldsymbol{A}) = -\sum_{i=0}^{N} \alpha_i^2 w_i \prod_{i=0}^{N} w_i. \tag{A.6}$$

Because $\boldsymbol{A}^{-1} = \boldsymbol{Q}^{-1}(\mathrm{diag}(\bar{\boldsymbol{w}}))^{-1}$ and the inverse of the diagonal matrix is trivial, all that remains is finding $\boldsymbol{Q}^{-1}$. The derivation of $\boldsymbol{Q}^{-1}$ is briefly outlined for a 3×3 matrix, as it can easily be extended to the generic case. We start with the augmented matrix

$$\left(\begin{array}{ccc|ccc} 1 & 0 & \alpha_0 & 1 & 0 & 0 \\ 0 & 1 & \alpha_1 & 0 & 1 & 0 \\ \alpha_0 w_0 & \alpha_1 w_1 & 0 & 0 & 0 & 1 \end{array}\right). \tag{A.7}$$

First, we subtract $\alpha_i w_i$ times the $(i+1)$th row from the last row, resulting in

$$\left(\begin{array}{ccc|ccc} 1 & 0 & \alpha_0 & 1 & 0 & 0 \\ 0 & 1 & \alpha_1 & 0 & 1 & 0 \\ 0 & 0 & r & -\alpha_0 w_0 & -\alpha_1 w_1 & 1 \end{array}\right).$$

Next for the first two rows, we multiply α_i/r times the last row and subtract it from the $(i+1)$th row. Moreover, we divide the last row by r, so that we obtain

$$\left(\begin{array}{ccc|ccc} 1 & 0 & 0 & 1+\alpha_0\frac{\alpha_0 w_0}{r} & \alpha_0\frac{\alpha_1 w_1}{r} & -\frac{\alpha_0}{r} \\ 0 & 1 & 0 & \alpha_1\frac{\alpha_0 w_0}{r} & 1+\alpha_1\frac{\alpha_1 w_1}{r} & -\frac{\alpha_1}{r} \\ 0 & 0 & 1 & -\frac{\alpha_0 w_0}{r} & -\frac{\alpha_1 w_1}{r} & \frac{1}{r} \end{array}\right).$$

We now have

$$\begin{aligned} \boldsymbol{A}^{-1} = \boldsymbol{Q}^{-1}\left(\mathrm{diag}\left(\bar{\boldsymbol{w}}\right)\right)^{-1} &= \frac{1}{r}\begin{pmatrix} r+\alpha_0\alpha_0 w_0 & \alpha_0\alpha_1 w_1 & -\alpha_0 \\ \alpha_1\alpha_0 w_0 & r+\alpha_1\alpha_1 w_1 & -\alpha_1 \\ -\alpha_0 w_0 & -\alpha_1 w_1 & 1 \end{pmatrix}\begin{pmatrix} \frac{1}{w_0} & & \\ & \frac{1}{w_1} & \\ & & 1 \end{pmatrix} \\ &= \frac{1}{r}\begin{pmatrix} \frac{r}{w_0}+\alpha_0^2 & \alpha_0\alpha_1 & -\alpha_0 \\ \alpha_1\alpha_0 & \frac{r}{w_1}+\alpha_1^2 & -\alpha_1 \\ -\alpha_0 & -\alpha_1 & 1 \end{pmatrix}. \end{aligned}$$

For the general case we obtain

$$\boldsymbol{A}^{-1} = \frac{1}{r}\begin{pmatrix} \boldsymbol{B} & -\boldsymbol{\alpha} \\ -\boldsymbol{\alpha}^{\mathrm{T}} & 1 \end{pmatrix} \tag{A.8a}$$

with r defined in (A.5) and where the coefficients of the matrix $\boldsymbol{B} = (B_{ij})$ read

$$B_{ij} = \begin{cases} \alpha_i^2 + \frac{r}{w_i} & \text{if } i = j, \\ \alpha_i \alpha_j & \text{if } i \neq j. \end{cases} \tag{A.8b}$$

By introducing $\bar{\boldsymbol{\alpha}} = (\alpha_0, \alpha_1, \ldots, \alpha_N, -1) \in \mathbb{R}^{N+2}$ and $\bar{\boldsymbol{y}} = (w_0^{-1}, w_1^{-1}, \ldots, w_N^{-1}, 0)$, the matrix $\boldsymbol{A}^{-1}$ can be written as

$$\boldsymbol{A}^{-1} = \operatorname{diag}(\bar{\boldsymbol{y}}) + \frac{1}{r}\bar{\boldsymbol{\alpha}} \otimes \bar{\boldsymbol{\alpha}}, \tag{A.9}$$

where $\otimes$ denotes the dyadic/tensor product between two vectors. From expression (A.9) it is obvious that the matrix-vector product $\boldsymbol{A}^{-1}\boldsymbol{b}$ can be efficiently implemented requiring only $\mathcal{O}(N)$ operations.

Appendix B

Constant state preservation in the ALE-ADER-DG scheme

In this section we will prove that the ALE-ADER-DG scheme described in Chapter 5 by the approximation of equation (5.17) together with the Taylor expansion (5.28) is constant state preserving when the refractive index field is constant. Since the integrals in equation (5.17) are computed with $(N+1)$-point Gauss-Legendre quadrature, we will employ the notation introduced in 6.3 to denote the usage of quadrature. That is, the application of $(N+1)$-point Gauss-Legendre quadrature to approximate the integral $\int_I g(x)\,\mathrm{d}x$ with interval $I = [a,b]$ is denoted as $\int_{I,N} g(x)\,\mathrm{d}x$ with the meaning

$$\int_{I,N} g(x)\,\mathrm{d}x = |I| \sum_{n=0}^{N} w_n g(a + |I|\,\xi_n), \tag{B.1}$$

with $|I| = b - a$, and $\{w_n\}_{n=0}^{N}$ and $\{\xi_n\}_{n=0}^{N}$ denoting the quadrature weights and points on the interval $[0,1]$. Similarly, the notation is used for multidimensional integrals where the multidimensional integral is evaluated as an iterated integral.

With the above notation equation (5.17) with integrals approximated by quadrature is written as

$$W_k\left((\rho_k \mathcal{J})^{t+1} - (\rho_k \mathcal{J})^{t}\right) = \int_{Z,N} \left(\int_{\chi,N} \left(\nabla_\xi \phi_k\right) \cdot \tilde{\boldsymbol{f}}\,\mathrm{d}\boldsymbol{\xi} - \int_{\partial\chi,N} \phi_k \tilde{\boldsymbol{F}} \cdot \hat{\boldsymbol{N}}\,\mathrm{d}\sigma \right) \mathrm{d}\tau, \tag{B.2}$$

with $Z = [z^t, z^{t+1}]$. Assume now that ρ is a constant at z^t, i.e., $\rho(z^t, \boldsymbol{\xi}) = c = \text{const}$, then the Taylor expansion (5.28) is equal to $\rho(z^t + \tau, \boldsymbol{\xi}) = c$ since all spatial derivatives are zero. The numerical flux (5.12) is consistent, i.e., for

$\rho^- = c$ and $\rho^+ = c$ the numerical flux (5.12) yields

$$\tilde{\boldsymbol{F}}(c,c)\cdot\hat{\boldsymbol{N}} = c\,(\tilde{\boldsymbol{u}}-\tilde{\boldsymbol{v}})\cdot\hat{\boldsymbol{N}}. \tag{B.3}$$

Inserting $\rho(z^t,\boldsymbol{\xi}) = c$ into the left-hand side and $\rho(z^t+\tau,\boldsymbol{\xi}) = c$ into the right-hand side of equation (B.2), and using (B.3) leads to

$$\begin{aligned} &W_k\Big((\rho_k\mathcal{J})^{t+1} - c\mathcal{J}^t\Big) = \\ &\qquad c\int_{Z,N}\left(\int_{\chi,N}\Big(\nabla_{\boldsymbol{\xi}}\phi_k\Big)\cdot(\tilde{\boldsymbol{u}}-\tilde{\boldsymbol{v}})\,\mathrm{d}\boldsymbol{\xi} - \int_{\partial\chi,N}\phi_k\,(\tilde{\boldsymbol{u}}-\tilde{\boldsymbol{v}})\cdot\hat{\boldsymbol{N}}\,\mathrm{d}\sigma\right)\mathrm{d}\tau. \end{aligned} \tag{B.4}$$

Next, we will prove for a constant refractive index field $n = n_0$ that

$$\int_{\chi,N}\Big(\nabla_{\boldsymbol{\xi}}\phi_k\Big)\cdot\tilde{\boldsymbol{u}}\,\mathrm{d}\boldsymbol{\xi} - \int_{\partial\chi,N}\phi_k\tilde{\boldsymbol{u}}\cdot\hat{\boldsymbol{N}}\,\mathrm{d}\sigma = 0. \tag{B.5}$$

Using (5.5) we can write $\tilde{\boldsymbol{u}} = (u_0\,\Delta p, u_1\,\Delta q)$, where for a constant refractive index field we have $u_1 = 0$ so that $\tilde{\boldsymbol{u}} = (u_0\,\Delta p, 0)$. Combining this with the transformation of χ to $\Omega(z)$ as described by (5.2), leads to

$$\tilde{\boldsymbol{u}}(\xi,\eta) = \begin{pmatrix} u_0\Big(p(\eta)\Big)\Delta p \\ 0 \end{pmatrix}. \tag{B.6}$$

Let $I = [0,1]$ denote the unit interval and recall that $\phi_k(\boldsymbol{\xi}) = \ell_i(\xi)\ell_j(\eta)$, see (5.13), then we can write the left-hand side of (B.5) as

$$\begin{aligned} &\int_{\chi,N}\Big(\nabla_{\boldsymbol{\xi}}\phi_k\Big)\cdot\tilde{\boldsymbol{u}}\,\mathrm{d}\boldsymbol{\xi} - \int_{\partial\chi,N}\phi_k\tilde{\boldsymbol{u}}\cdot\hat{\boldsymbol{N}}\,\mathrm{d}\sigma \\ &\qquad = \Delta p\int_{I,N}\int_{I,N}\frac{\mathrm{d}\ell_i(\xi)}{\mathrm{d}\xi}\ell_j(\eta)\;u_0(p(\eta))\,\mathrm{d}\xi\,\mathrm{d}\eta \\ &\qquad\qquad - \Delta p\Big(\ell_i(1)-\ell_i(0)\Big)\int_{I,N}\ell_j(\eta)\;u_0(p(\eta))\,\mathrm{d}\eta \\ &\qquad = \Delta p\int_{I,N}\ell_j(\eta)\;u_0(p(\eta))\,\mathrm{d}\eta\left[\int_{I,N}\frac{\mathrm{d}\ell_i(\xi)}{\mathrm{d}\xi}\mathrm{d}\xi - (\ell_i(1)-\ell_i(0))\right]. \end{aligned} \tag{B.7}$$

The integral in the square brackets is evaluated exactly since $\dfrac{\mathrm{d}\ell_i(\xi)}{\mathrm{d}\xi}$ is a polynomial of degree $N-1$, hence,

$$\int_{I,N}\frac{\mathrm{d}\ell_i(\xi)}{\mathrm{d}\xi}\mathrm{d}\xi = \int_0^1\frac{\mathrm{d}\ell_i(\xi)}{\mathrm{d}\xi}\mathrm{d}\xi = \ell_i(1)-\ell_i(0). \tag{B.8}$$

Therefore, we obtain

$$\int_{\chi,N} \left(\nabla_{\boldsymbol{\xi}} \phi_k\right) \cdot \tilde{\boldsymbol{u}} \, \mathrm{d}\boldsymbol{\xi} - \int_{\partial\chi,N} \phi_k \tilde{\boldsymbol{u}} \cdot \hat{\boldsymbol{N}} \, \mathrm{d}\sigma = 0, \tag{B.9}$$

proving relation (B.5). Substituting relation (B.5) in (B.4) leads to

$$\begin{aligned} &W_k\left((\rho_k \mathcal{J})^{t+1} - c\mathcal{J}^t\right) = \\ &\quad c \int_{Z,N} \left(\int_{\chi,N} \left(\nabla_{\boldsymbol{\xi}} \phi_k\right) \cdot (-\tilde{\boldsymbol{v}}) \, \mathrm{d}\boldsymbol{\xi} - \int_{\partial\chi,N} \phi_k \, (-\tilde{\boldsymbol{v}}) \cdot \hat{\boldsymbol{N}} \, \mathrm{d}\sigma \right) \mathrm{d}\tau. \end{aligned} \tag{B.10}$$

The velocity field $\boldsymbol{v} = \frac{\partial \boldsymbol{x}}{\partial \tau}$ is linear in $\boldsymbol{\xi}$, cf. (5.2), and since the test functions are polynomials of degree N in ξ and η the integrals over χ and $\partial\chi$ are evaluated exactly. Thus, we can exchange $\int_{\chi,N}$ with $\int_{\chi}$ and subsequently apply Gauss's theorem and the product rule, so that we obtain

$$\begin{aligned} W_k\left((\rho_k \mathcal{J})^{t+1} - c\mathcal{J}^t\right) &= c \int_{Z,N} \left(\int_{\chi} \left(\nabla_{\boldsymbol{\xi}} \phi_k\right) \cdot (-\tilde{\boldsymbol{v}}) \, \mathrm{d}\boldsymbol{\xi} - \int_{\partial\chi} \phi_k \, (-\tilde{\boldsymbol{v}}) \cdot \hat{\boldsymbol{N}} \, \mathrm{d}\sigma \right) \mathrm{d}\tau \\ &= c \int_{Z,N} \int_{\chi} \phi_k \nabla_{\boldsymbol{\xi}} \cdot \tilde{\boldsymbol{v}} \, \mathrm{d}\boldsymbol{\xi} \, \mathrm{d}\tau. \end{aligned} \tag{B.11}$$

Applying the geometric conservation law (5.7) to replace $\nabla_{\boldsymbol{\xi}} \cdot \tilde{\boldsymbol{v}}$ with $\frac{\mathrm{d}\mathcal{J}}{\mathrm{d}\tau}$ leads to

$$W_k\left((\rho_k \mathcal{J})^{t+1} - c\mathcal{J}^t\right) = c \left(\int_{Z,N} \frac{\mathrm{d}\mathcal{J}}{\mathrm{d}\tau} \, \mathrm{d}\tau \right) \int_{\chi} \phi_k \, \mathrm{d}\boldsymbol{\xi}. \tag{B.12}$$

Note that $\phi_k(\boldsymbol{\xi}) = \ell_i(\xi)\ell_j(\eta)$, so that a quick computation by applying the $(N+1)$-point Gauss-Legendre quadrature shows

$$\int_{\chi} \phi_k \, \mathrm{d}\boldsymbol{\xi} = \int_{\chi} \ell_i(\xi)\ell_j(\eta) \, \mathrm{d}\xi \mathrm{d}\eta = w_i w_j = W_k,$$

where the latter equality follows from the definition of W_k. Combining this with the fact that we integrate $\frac{\mathrm{d}\mathcal{J}}{\mathrm{d}\tau}$ numerically, see (5.21), as opposed to exact integration, leads to

$$W_k\left(\rho_k^{t+1} \mathcal{J}^{t+1} - c\mathcal{J}^t\right) = c \, W_k\left(\mathcal{J}^{t+1} - \mathcal{J}^t\right), \tag{B.13}$$

from which we can directly obtain $\rho_k^{t+1} = c$ for $k = 1, \ldots, N_d$ showing that the scheme is constant state preserving.

Appendix C

Local energy balances at an optical interface

In Section 5.5 we formulated the energy balance (5.51) that relates the fluxes across an optical interface for the jump condition $\rho^+(\vec{p}) = \rho^-(\vec{i})$ with $\vec{p} = \mathcal{S}(\vec{i})$. The total flux leaving a momentum interval R is related to the flux striking the intervals $\mathcal{I}(R;\mathrm{b},\sigma)$ and $\mathcal{I}(R;\mathrm{f},\sigma)$ by

$$\int_R \rho_\sigma \left(u_0 - \frac{\mathrm{d}Q}{\mathrm{d}z}\right)\Big|_+ \mathrm{d}p = \int_{\mathcal{I}(R;\mathrm{b},\sigma)} \rho_\mathrm{b} \left(u_0 - \frac{\mathrm{d}Q}{\mathrm{d}z}\right)\Big|_- \mathrm{d}p + \int_{\mathcal{I}(R;\mathrm{f},\sigma)} \rho_\mathrm{f} \left(u_0 - \frac{\mathrm{d}Q}{\mathrm{d}z}\right)\Big|_- \mathrm{d}p, \tag{C.1}$$

with $\mathcal{I}$ defined in (5.50). As described in Section 5.5.1 we partition the interval R as $R = R_0 \cup R_1$, so that $\mathcal{I}(R_0;\mathrm{b},\sigma) = \emptyset$ and $\mathcal{I}(R_1;\mathrm{f},\sigma) = \emptyset$. Henceforth, we will assume $\mathcal{I}(R;\mathrm{b},\sigma) = \emptyset$, so that the balance (C.1) reduces to

$$\int_R \rho \left(u_0 - \frac{\mathrm{d}Q}{\mathrm{d}z}\right)\Big|_+ \mathrm{d}p = \int_{\mathcal{I}(R;\mathrm{f},\sigma)} \rho \left(u_0 - \frac{\mathrm{d}Q}{\mathrm{d}z}\right)\Big|_- \mathrm{d}p, \tag{C.2}$$

where we omit the subscript σ for ρ. Here, we will prove the balance (C.2) for reflection and refraction. Let the optical interface be given by $q = Q(z)$ and let prime $'$ denote differentiation with respect to z, i.e., $' = \frac{\mathrm{d}}{\mathrm{d}z}$, then we have the following unit normal

$$\vec{\nu} = \begin{pmatrix} \nu_q \\ \nu_z \end{pmatrix} = \frac{\pm 1}{\sqrt{1 + Q'(z)^2}} \begin{pmatrix} -1 \\ Q'(z) \end{pmatrix}. \tag{C.3}$$

In what follows the sign of the normal is not important.

Furthermore, we will require the jump condition (2.42) which we will shorten to $\rho^-(p^-) = \rho^+(p^+)$ where we omit the position coordinates as they

remain constant at the optical interface and omit σ for sake of simplicity, and use the shorthand notation $p^+ = S(p^-)$ where S was introduced in Section 5.5. Recall that the $\pm$ superscripts denote one-sided limits towards the optical interface, where the $-$ refers to the incident side while the $+$ denotes the outgoing side, i.e., after reflection or refraction.

We start proving relation (C.2) by transforming its left-hand side by the use of the jump condition and subsequently making a coordinate transformation by using $p = S(i_q)$ where i_q denotes the q-component of the incident momentum vector, i.e.,

$$\begin{aligned}\int_R \rho^+(p)\,(u_0^+(p) - Q'(z))\,\mathrm{d}p &= \int_R \rho^-\left(S^{-1}(p)\right)(u_0^+(p) - Q'(z))\,\mathrm{d}p \\ &= \int_{\mathcal{I}(R;\mathrm{f},\sigma)} \rho^-\left(i_q\right)\left(u_0^+\left(S(i_q)\right) - Q'(z)\right)\frac{\mathrm{d}S(i_q)}{\mathrm{d}i_q}\mathrm{d}i_q. \end{aligned} \tag{C.4}$$

From (C.2) it then follows that the following relation should hold

$$\int_{\mathcal{I}(R;\mathrm{f},\sigma)} \rho^-\left(i_q\right)\left(u_0^+\left(S(i_q)\right) - Q'(z)\right)\frac{\mathrm{d}S(i_q)}{\mathrm{d}i_q}\mathrm{d}i_q = \int_{\mathcal{I}(R;\mathrm{f},\sigma)} \rho^-\left(i_q\right)\left(u_0^-(i_q) - Q'(z)\right)\mathrm{d}i_q, \tag{C.5}$$

where upon subtracting the right-hand side from the left-hand side, we see that relation (C.5) holds for arbitrary ρ^- if

$$\left(u_0^+\left(S(i_q)\right) - Q'(z)\right)\frac{\mathrm{d}S(i_q)}{\mathrm{d}i_q} = u_0^-(i_q) - Q'(z). \tag{C.6}$$

Note that relation (C.6) must hold independently of whether we assume $\mathcal{I}(R;\mathrm{b},\sigma) = \emptyset$ or $\mathcal{I}(R;\mathrm{f},\sigma) = \emptyset$. In other words, if we would have assumed $\mathcal{I}(R;\mathrm{f},\sigma) = \emptyset$ instead, we would still end up again with relation (C.6). We will first prove relation (C.6) for reflection and then for refraction.

Recall that the law of reflection transforms an incident momentum $\vec{i} = (i_q, i_z)$ to the reflected momentum $\vec{r} = (r_q, r_z)$, by

$$\vec{r} = \vec{i} - 2\psi\vec{\nu} \quad \text{with } \vec{i} = \begin{pmatrix} i_q \\ \sigma\sqrt{n^2 - i_q^2} \end{pmatrix} \text{ and } \psi = \vec{i}\cdot\vec{\nu},$$

where σ denotes the direction of the light ray and n is the refractive index of the incident and reflected light ray. The q-component of the law of reflection is denoted as $S_\mathrm{R}(i_q)$, see (5.55), and its derivative reads

$$\frac{\mathrm{d}S_\mathrm{R}(i_q)}{\mathrm{d}i_q} = 1 - 2\nu_q\left(\nu_q - \frac{i_q}{i_z}\nu_z\right).$$

Note that the velocity field $u_0(i_q) = i_q/i_z$, see relation (2.40b). Hence, we write the velocities $u_0^{\pm}$ in terms of the vectors $\vec{i}$ and $\vec{r}$, i.e.,

$$u_0^-(i_q) = \frac{i_q}{i_z} \text{ and } u_0^+(S_{\mathrm{R}}(i_q)) = \frac{r_q}{r_z}.$$

Moreover, $Q'(z)$ can be written as $Q'(z) = -\nu_z/\nu_q$, cf. (C.3). Therefore, relation (C.6) can be written as

$$\left(\frac{r_q}{r_z} + \frac{\nu_z}{\nu_q}\right)\left(1 - 2\nu_q\left(\nu_q - \frac{i_q}{i_z}\nu_z\right)\right) = \frac{i_q}{i_z} + \frac{\nu_z}{\nu_q}. \tag{C.7}$$

We proceed by subtracting the right-hand side from the left-hand side so that we obtain

$$0 = \frac{r_q}{r_z} - \frac{i_q}{i_z} - 2\nu_q\left(\frac{r_q}{r_z} + \frac{\nu_z}{\nu_q}\right)\left(\nu_q - \frac{i_q}{i_z}\nu_z\right).$$

Next, we rewrite the terms to a common denominator of $r_z i_z$ as follows

$$\begin{aligned} 0 &= \frac{r_q i_z}{r_z i_z} - \frac{i_q r_z}{r_z i_z} - 2\nu_q \frac{1}{r_z}\left(r_q + r_z\frac{\nu_z}{\nu_q}\right)\frac{1}{i_z}\left(i_z\nu_q - i_q\nu_z\right) \\ &= \frac{1}{r_z i_z}\Big[r_q i_z - i_q r_z - 2\left(r_q\nu_q + r_z\nu_z\right)\left(i_z\nu_q - i_q\nu_z\right)\Big]. \end{aligned} \tag{C.8}$$

The term $r_q\nu_q + r_z\nu_z = \vec{r}\cdot\vec{\nu} = -\psi$, which can be derived by multiplying the law of reflection with $\vec{\nu}$. The expression (C.8) is rewritten using cross products, i.e., let $\boldsymbol{i} = (i_q, i_z, 0)$ denote the 3-vector of $\vec{i}$ etc. Moreover, let $\hat{\boldsymbol{e}}_3 = (0,0,1)$, then from expression (C.8) we find

$$\begin{aligned} \frac{1}{r_z i_z}\Big[(\boldsymbol{r}\times\boldsymbol{i})\cdot\hat{\boldsymbol{e}}_3 + 2\psi\,(\boldsymbol{\nu}\times\boldsymbol{i})\cdot\hat{\boldsymbol{e}}_3\Big] &= \frac{1}{r_z i_z}\Big[(\boldsymbol{i} - 2\psi\boldsymbol{\nu})\times\boldsymbol{i} + 2\psi\boldsymbol{\nu}\times\boldsymbol{i}\Big]\cdot\hat{\boldsymbol{e}}_3 \\ &= \frac{1}{r_z i_z}\Big[-2\psi\boldsymbol{\nu}\times\boldsymbol{i} + 2\psi\boldsymbol{\nu}\times\boldsymbol{i}\Big]\cdot\hat{\boldsymbol{e}}_3 \\ &= 0, \end{aligned}$$

completing the proof for reflection.

Now for refraction, recall that Snell's law of refraction transforms an incident momentum $\vec{i} = (i_q, i_z)$ to the transmitted/refracted momentum $\vec{t} = (t_q, t_z)$. Snell's law of refraction reads

$$\vec{t} = \vec{i} - \left(\psi + \sqrt{\delta}\right)\vec{\nu} \quad \text{with } \delta = n_1^2 - n_0^2 + \psi^2,$$

with again $\psi = \vec{i}\cdot\vec{\nu}$ and where n_0 and n_1 are the incident and transmitted media, respectively. The q-component of Snell's law is written as $S_{\mathrm{T}}(i_q)$, see (5.55), and its derivative reads

$$\frac{\mathrm{d}S_{\mathrm{T}}(i_q)}{\mathrm{d}i_q} = 1 - \nu_q\left(1 + \frac{\psi}{\sqrt{\delta}}\right)\left(\nu_q - \frac{i_q}{i_z}\nu_z\right).$$

As before, we can write the velocities $u_0^{\pm}$ in terms of the vectors $\vec{i}$ and $\vec{t}$, i.e.,

$$u_0^-(i_q) = \frac{i_q}{i_z} \text{ and } u_0^+(S_{\mathrm{T}}(i_q)) = \frac{t_q}{t_z}.$$

Therefore, relation (C.6) can be written as

$$\left(\frac{t_q}{t_z} + \frac{\nu_z}{\nu_q}\right)\left(1 - \nu_q\left(1 + \frac{\psi}{\sqrt{\delta}}\right)\left(\nu_q - \frac{i_q}{i_z}\nu_z\right)\right) = \frac{i_q}{i_z} + \frac{\nu_z}{\nu_q}. \tag{C.9}$$

We proceed by subtracting the right-hand side from the left-hand side such that we obtain

$$0 = \frac{t_q}{t_z} - \frac{i_q}{i_z} - \nu_q\left(1 + \frac{\psi}{\sqrt{\delta}}\right)\left(\frac{t_q}{t_z} + \frac{\nu_z}{\nu_q}\right)\left(\nu_q - \frac{i_q}{i_z}\nu_z\right).$$

Next, we rewrite terms to a common denominator of $t_z i_z$ as follows

$$\begin{aligned} 0 &= \frac{t_q i_z}{t_z i_z} - \frac{i_q t_z}{t_z i_z} - \nu_q\left(1 + \frac{\psi}{\sqrt{\delta}}\right)\frac{1}{t_z}\left(t_q + t_z\frac{\nu_z}{\nu_q}\right)\frac{1}{i_z}\left(i_z\nu_q - i_q\nu_z\right) \\ &= \frac{1}{t_z i_z}\left[t_q i_z - i_q t_z - \left(1 + \frac{\psi}{\sqrt{\delta}}\right)\left(t_q\nu_q + t_z\nu_z\right)\left(i_z\nu_q - i_q\nu_z\right)\right]. \end{aligned} \tag{C.10}$$

The term $t_q\nu_q + t_z\nu_z = \vec{t}\cdot\vec{\nu} = -\sqrt{\delta}$, which can be derived by multiplying Snell's law of refraction with $\vec{\nu}$. Once again, we employ 3-vectors to write expression (C.10) as

$$\begin{aligned} \frac{1}{t_z i_z}\Big[(\boldsymbol{t}\times\boldsymbol{i})\cdot\hat{\boldsymbol{e}}_3 + \left(\sqrt{\delta}+\psi\right)(\boldsymbol{\nu}\times\boldsymbol{i})\cdot\hat{\boldsymbol{e}}_3\Big] &= \frac{1}{t_z i_z}\Big[\left(\boldsymbol{i} - \left(\psi+\sqrt{\delta}\right)\boldsymbol{\nu}\right)\times\boldsymbol{i} + \left(\psi+\sqrt{\delta}\right)\boldsymbol{\nu}\times\boldsymbol{i}\Big]\cdot\hat{\boldsymbol{e}}_3 \\ &= \frac{1}{t_z i_z}\Big[-\left(\psi+\sqrt{\delta}\right)\boldsymbol{\nu}\times\boldsymbol{i} + \left(\psi+\sqrt{\delta}\right)\boldsymbol{\nu}\times\boldsymbol{i}\Big]\cdot\hat{\boldsymbol{e}}_3 \\ &= 0, \end{aligned}$$

completing the proof for refraction.

Finally, we remark that for an interface given by $z = Z(q)$ relation (C.6) needs to multiplied by ν_q/ν_z to find the same result, where the normal now reads

$$\vec{\nu} = \frac{\pm 1}{\sqrt{1 + \frac{\mathrm{d}Z}{\mathrm{d}q}^2}}\begin{pmatrix}\frac{\mathrm{d}Z}{\mathrm{d}q}\\ -1\end{pmatrix}.$$

C.1 Extension to Fresnel reflections

In Section 7.1 we formulated the energy balance (7.2) when considering Fresnel reflections with the jump condition given by (7.1). The total flux leaving a momentum interval R is related to the flux striking the incident intervals

$$\begin{aligned}\int_R \rho_\sigma\,(u_0 - Q'(z))\Big|_+ \,\mathrm{d}p = \sum_{\sigma_{\mathrm{inc}}\in\{\mathrm{b},\mathrm{f}\}} \int_{\mathcal{I}_{\mathrm{R}}(R;\sigma_{\mathrm{inc}},\sigma)} \mathcal{R}\rho_{\sigma_{\mathrm{inc}}}\,(u_0 - Q'(z))\Big|_- \,\mathrm{d}p \\ + \int_{\mathcal{I}_{\mathrm{T}}(R;\sigma_{\mathrm{inc}},\sigma)} (1-\mathcal{R})\,\rho_{\sigma_{\mathrm{inc}}}\,(u_0 - Q'(z))\Big|_- \,\mathrm{d}p,\end{aligned} \tag{C.11}$$

with $\mathcal{I}_{\mathrm{R}}$ and $\mathcal{I}_{\mathrm{T}}$ denoting the incident light that after reflection and transmission, respectively, correspond to the momentum interval R with propagation direction given by σ. The propagation direction of the incident light is given by σ_{inc}.

The energy balance (C.11) is proven by applying the jump condition to its left-hand side and making a coordinate transformation. Thus, we start the proof by transforming the left-hand side of relation (7.2) by applying the jump condition for Fresnel reflections (7.1), which leads to

$$\begin{aligned}&\int_R \rho_\sigma^+(p)\Big(u_0^+(p) - Q'(z)\Big)\,\mathrm{d}p \\ &= \sum_{\sigma_{\mathrm{inc}}\in\{\mathrm{b},\mathrm{f}\}} \int_{R_{\mathrm{R},\sigma_{\mathrm{inc}}}} \mathcal{R}\Big(S_{\mathrm{R}}^{-1}(p)\Big)\rho_{\sigma_{\mathrm{inc}}}^-\Big(S_{\mathrm{R}}^{-1}(p)\Big)\Big(u_0^+(p) - Q'(z)\Big)\,\mathrm{d}p \\ &\qquad + \int_{R_{\mathrm{T},\sigma_{\mathrm{inc}}}} \Big[1 - \mathcal{R}\Big(S_{\mathrm{T}}^{-1}(p)\Big)\Big]\rho_{\sigma_{\mathrm{inc}}}^-\Big(S_{\mathrm{T}}^{-1}(p)\Big)\Big(u_0^+(p) - Q'(z)\Big)\,\mathrm{d}p.\end{aligned} \tag{C.12}$$

Here, $R_{\mathrm{R},\sigma_{\mathrm{inc}}} \subset R$ contains the outgoing light, for which its corresponding incident light has propagation direction σ_{inc} and the outgoing light is just a reflection (subscript R) of the incident light. Moreover, if we would compute the reflection of the incident light $\mathcal{I}_{\mathrm{R}}(R;\sigma_{\mathrm{inc}},\sigma)$ we would obtain $R_{\mathrm{R},\sigma_{\mathrm{inc}}}$. Analogously, computing the refraction of the incident light $\mathcal{I}_{\mathrm{T}}(R;\sigma_{\mathrm{inc}},\sigma)$ leads to $R_{\mathrm{T},\sigma_{\mathrm{inc}}}$.

Applying the transformation $p = S_{\mathrm{R}}(i_q)$ and $p = S_{\mathrm{T}}(i_q)$ to the first and

second integral, respectively, on the right-hand side of (C.12) leads to

$$\begin{aligned}\int_R \rho_\sigma^+(p)\Big(u_0^+(p)-Q'(z)\Big)\,\mathrm{d}p \\ &= \sum_{\sigma_{\mathrm{inc}}\in\{\mathrm{b},\mathrm{f}\}} \int_{\mathcal{I}_{\mathrm{R}}(R;\sigma_{\mathrm{inc}},\sigma)} \mathcal{R}(i_q)\rho_{\sigma_{\mathrm{inc}}}^-(i_q)\Big(u_0^+\Big(S_{\mathrm{R}}(i_q)\Big)-Q'(z)\Big)\frac{\mathrm{d}S_{\mathrm{R}}(i_q)}{\mathrm{d}i_q}\,\mathrm{d}i_q \\ &\quad + \int_{\mathcal{I}_{\mathrm{T}}(R;\sigma_{\mathrm{inc}},\sigma)} \Big[1-\mathcal{R}(i_q)\Big]\rho_{\sigma_{\mathrm{inc}}}^-(i_q)\Big(u_0^+\Big(S_{\mathrm{T}}(i_q)\Big)-Q'(z)\Big)\frac{\mathrm{d}S_{\mathrm{T}}(i_q)}{\mathrm{d}i_q}\,\mathrm{d}i_q,\end{aligned} \tag{C.13}$$

From relation (C.11) it then follows that the following relation should hold

$$\begin{aligned}&\sum_{\sigma_{\mathrm{inc}}\in\{\mathrm{b},\mathrm{f}\}} \int_{\mathcal{I}_{\mathrm{R}}(R;\sigma_{\mathrm{inc}},\sigma)} \mathcal{R}(i_q)\rho_{\sigma_{\mathrm{inc}}}^-(i_q)\Big(u_0^-(i_q)-Q'(z)\Big)\,\mathrm{d}i_q \\ &\qquad + \int_{\mathcal{I}_{\mathrm{T}}(R;\sigma_{\mathrm{inc}},\sigma)} \Big(1-\mathcal{R}(i_q)\Big)\rho_{\sigma_{\mathrm{inc}}}^-(i_q)\Big(u_0^-(i_q)-Q'(z)\Big)\,\mathrm{d}i_q \\ &= \sum_{\sigma_{\mathrm{inc}}\in\{\mathrm{b},\mathrm{f}\}} \int_{\mathcal{I}_{\mathrm{R}}(R;\sigma_{\mathrm{inc}},\sigma)} \mathcal{R}(i_q)\rho_{\sigma_{\mathrm{inc}}}^-(i_q)\Big(u_0^+\Big(S_{\mathrm{R}}(i_q)\Big)-Q'(z)\Big)\frac{\mathrm{d}S_{\mathrm{R}}(i_q)}{\mathrm{d}i_q}\,\mathrm{d}i_q \\ &\qquad + \int_{\mathcal{I}_{\mathrm{T}}(R;\sigma_{\mathrm{inc}},\sigma)} \Big[1-\mathcal{R}(i_q)\Big]\rho_{\sigma_{\mathrm{inc}}}^-(i_q)\Big(u_0^+\Big(S_{\mathrm{T}}(i_q)\Big)-Q'(z)\Big)\frac{\mathrm{d}S_{\mathrm{T}}(i_q)}{\mathrm{d}i_q}\,\mathrm{d}i_q.\end{aligned} \tag{C.14}$$

If relation (C.14) holds for arbitrary ρ^-, then from the first integrals on both sides we see that the following relation must hold

$$\begin{aligned}\int_{\mathcal{I}_{\mathrm{R}}(R;\sigma_{\mathrm{inc}},\sigma)} \mathcal{R}(i_q)\rho_{\sigma_{\mathrm{inc}}}^-(i_q)\Big(u_0^-(i_q)-Q'(z)\Big)\,\mathrm{d}i_q \\ &= \int_{\mathcal{I}_{\mathrm{R}}(R;\sigma_{\mathrm{inc}},\sigma)} \mathcal{R}(i_q)\rho_{\sigma_{\mathrm{inc}}}^-(i_q)\Big(u_0^+\Big(S_{\mathrm{R}}(i_q)\Big)-Q'(z)\Big)\frac{\mathrm{d}S_{\mathrm{R}}(i_q)}{\mathrm{d}i_q}\,\mathrm{d}i_q,\end{aligned}$$

or equivalently, by subtracting the right-hand side from the left-hand side, then for arbitrary ρ^- the following relation must hold

$$\Big(u_0^+\Big(S_{\mathrm{R}}(i_q)\Big)-Q'(z)\Big)\frac{\mathrm{d}S_{\mathrm{R}}(i_q)}{\mathrm{d}i_q} = u_0^-(i_q)-Q'(z). \tag{C.15a}$$

Similarly, for the second integrals on both sides this leads to the relation

$$\Big(u_0^+\Big(S_{\mathrm{T}}(i_q)\Big)-Q'(z)\Big)\frac{\mathrm{d}S_{\mathrm{T}}(i_q)}{\mathrm{d}i_q} = u_0^-(i_q)-Q'(z). \tag{C.15b}$$

The relations (C.15) are equivalent to relation (C.6), hence, they were already proven in this appendix. So to conclude, the energy balance (C.11) for Fresnel reflections has been proven.

Appendix D

Details of the developed software

The numerical methods developed in this thesis were all implemented from scratch by the author of this thesis. The results from Chapter 4 were computed with a code developed in MATLAB. All the other results were computed with a code written in C++, making use of options from C++17. The numerical simulations performed with the software generates data that is written to disk. The data is then post-processed using Python, with all of the plotting done with Matplotlib. As the development of the software was also a big part of this thesis, in the sense of validating and applying the presented ideas and methods, details of the written C++ software are contained in this appendix.

Software development can follow different principles, which are in general ideas or guidelines on how to structure the data and how to separate different concerns. Two main principles were followed, the first being a *data-oriented design* and the second being the *object-oriented programming* paradigm. The former is prominently seen in the data layout, which in general is a structure of arrays. Object-oriented programming is used to extend main classes, via inheritance, into specific sub-classes. This approach together with a separation of concerns, allows the code to be easily extended.

An important aspect of the written code is its performance. A lot of operations such as interpolation and taking derivatives of a polynomial, are performed as a small matrix multiplication. The *libxsmm* library [48, 49] for small matrix multiplications on Intel machines is used to ensure high performance. The implementation of methods, such as polynomial interpolation at arbitrary nodes or the computation of a Taylor series, were also timed and benchmarked using Google's benchmark library [43].

D.1 Structure of the code

The main structure of the code can be divided in the important classes that describe the geometry of the problem, the representation of the piecewise polynomial DG solution, the discretisation of the PDE and the physics at optical interfaces. First we detail the main classes for the two-dimensional solver of Liouville's equation on a moving 2D phase space. Later in this section, we will present the extension to the four-dimensional solver on a moving curvilinear 4D phase space.

The two-dimensional phase space domain is covered with rectangles, i.e. the elements, that is described in the class *Mesh*. The connectivity of the elements is described in the class *Faces*, where the connectivity is stored for each edge. Additionally, we also store the type of each edge, describing whether it is an interior edge, a boundary edge, or an optical interface edge. The tensor-product nature of the mesh is stored in an additional class, which details which elements share the same q-interval. Due to this structure searching for the element that a point belongs to is divided into 2 one-dimensional searches.

Mesh movement is only allowed in the q-direction and is described by the class *MeshVelocity* that contains basic methods for setting the mesh velocity and its derivatives (as is required for the ADER approach), which require user-defined functions to be passed. Mesh refinement is detailed by a set of methods, that describe the coarsening and refinement of elements. Moreover, there are methods that detail the expanding/shrinking of the mesh (for a flat optical interface at $z = \text{const}$) along the p-direction. Also there is a method that is used to make the mesh locally uniform (in Δq) for the hybrid SLDG and ADER-DG solver. The automatic mesh refinement is encapsulated in the class *AMRController*, which provided with $\Delta q_{\min}$ and $\Delta q_{\max}$ can update the mesh, or a collection of elements, to the desired mesh spacings.

The main discretisation of Liouville's equation is described by the class *LiouvilleSolver*. This class stores two fields for the piecewise polynomial solution, its current state and its update, and stores the velocity field, the mesh velocity, the mesh and a so-called instance of the class *NodalBasis*. The *NodalBasis* class stores all pre-computed information regarding the one-dimensional polynomial basis, i.e., the Lagrange polynomials $\ell_i(\xi)$, and interpolation matrices to $\xi = 0$ and $\xi = 1$, derivative matrices, and the Gauss-Legendre quadrature nodes and weights. The solver provides methods that compute the volume and surface terms of the weak formulation, for static and moving type of elements. The discretisation of the sub-cell interface method is also included, but additionally requires information about the geometry of the optical interface and a method that describes the solution of a local ray trace, i.e., a

method that provides the location of the characteristic in the semi-Lagrangian step. For convenience there is a method that when called fully automatically computes the ADER-DG discretisation and updates the solution, for multiple steps till the user tells it to stop. At every step a callback method is called before computing the discretisation, which serves as the point to perform automatic mesh refinement and to update the solver's configuration. The optical interface discretisation can also be computed automatically.

The discretisation of the optical interfaces, i.e., the jump condition, is handled by separate classes that derive from the base class *OpticalInterfaceFlux*. The base class stores geometric information at the optical interface and has a method for performing a binary search. Their exist two sub-classes that implement the, either fully reflective or refractive, jump condition using the straightforward interpolation (non-conservative) and the conservative method as detailed in Section 5.5. Similarly, for the jump condition describing Fresnel reflections there exist two sub-classes with either the straightforward interpolation and the conservative method as detailed in Chapter 7. A fifth sub-class exists when one wants to compute the coupling from forward to backward light and vice versa as used in Chapter 8. All these classes require the computation of reflection $\mathcal{S}_{\mathrm{R}}$ and transmission $\mathcal{S}_{\mathrm{T}}$, and reflection coefficients $\mathcal{R}$, which are all incorporated in a collection of methods that deal with optics.

The modal filter and limiter from Chapter 8 are described by classes that can be passed to the LiouvilleSolver.

The discretisation and local time stepping for the hybrid SLDG and ADER-DG solver is implemented in the class *LiouvilleSolverLTS*, which is a sub-class of *LiouvilleSolver*. It features a few data structures related to the efficient computation of the semi-Lagrangian DG scheme and the coupling fluxes, and the storage of elements in the gray ADER-DG regions. In the former data structures, the operators are pre-computed when necessary before being applied to update the solution or to compute the coupling fluxes.

For the extension of the *LiouvilleSolver* to three-dimensional optics, on a moving curvilinear four-dimensional phase space domain, a new mesh class has been designed. The four-dimensional mesh consists of a tensor product between 2 two-dimensional curvilinear meshes. The two-dimensional curvilinear mesh stores again the geometry of the elements, which can now be quadrilaterals with either straight-sided edges or curved edges. Since the Gauss-Lobatto-Chebyshev nodes are used for representing this geometry, the *NodalBasis* class is extended to provide an implementation for this point set and the Gauss-Legendre point set.

The new class *LiouvilleSolver4D* exploits the tensor-product structure of the mesh, by computing and caching the required position geometry at z-

quadrature nodes for each position element. The discretisation then follows a two-stage process, where the outer loop is over the position elements and the inner loop over the momentum elements. The inner loop is parallelised with OpenMP [26]. For efficiency purposes a one-touch policy is adopted, in which each unique element only computes its local ADER predictor only once. The discretisation of the optical interface is again implemented using a base class. At the moment there is only one sub-class which implements the jump condition via a straightforward interpolation. The search on the momentum disk is implemented in a separate class *DiskSearch*. The search algorithm is explained in Section 9.3.

In addition to the solvers of Liouville's equation, the quasi-Monte Carlo (QMC) ray tracers were also implemented in C++. Both the two- and three-dimensional optics ray tracers follow similar principles. An optical interface/surface is represented by a class *Interface*. The physical geometry is represented by a collection of these interfaces in the class *OpticalSystem*. Each interface class is required to implement an intersection method. Then, in the *OpticalSystem* class tracing a ray is relatively simple and the optical laws are applied upon intersection. The QMC process is implemented in a general class *QuasiMonteCarloBase*, which can draw the initial ray coordinates from the Sobol sequence from the *boost* library. Derived classes only need to implement how to search for the correct bin given a ray's final position coordinates. For 2D optics with a uniform grid on the target plane leads to a constant search time for each ray, whereas for a non-uniform grid the search algorithm is done with a straightforward loop. It was found that the cost of the search algorithm was low compared to the cost of ray tracing. For 3D optics with a square domain on the target plane, an efficient algorithm was implemented for a general mesh by the use of a uniform background mesh as detailed in Chapter 9.

D.2 Testing and validation

During the development of the software, multiple test cases (that are not covered in the results of this thesis) were written to test and validate the written code. Tests range from checking whether the methods related to optics are correct, to validating the implementation of the local ADER predictor and the mesh refinement algorithms. Furthermore, the correctness of the energy-conservative discretisation of the jump condition is validated by checking whether it is actually conservative and testing the convergence for manufactured solutions.

Convergence of the DG discretisations is usually checked initially for the

example where $n(z, \boldsymbol{q}) = \text{const}$, providing a simple sanity check of the code.

Besides these test examples, the code is also partially exposed to Python via *pybind11*. This allows the C++ defined classes and methods to be imported in Python. This is useful for quick prototyping of new parts of software in Python before implementing them in C++. For example, the modal filter and limiter were first tested in Python, whilst making use of the C++ *NodalBasis* class in Python.

If desired the entire C++ code could be exposed to Python allowing for quick implementations of optical systems.

Bibliography

[1] V. I. Arnold. *Mathematical Methods of Classical Mechanics*, volume 60. Springer Science & Business Media, 2013.

[2] J. Badwaik, P. Chandrashekar, and C. Klingenberg. Single-Step Arbitrary Lagrangian–Eulerian Discontinuous Galerkin Method for 1-D Euler Equations. *Communications on Applied Mathematics and Computation*, 2(4):541–579, 2020.

[3] M. Bahrami and A. V. Goncharov. Geometry-invariant GRIN lens: Finite ray tracing. *Optics Express*, 22(23):27797–27810, 2014.

[4] G. E. Barter and D. L. Darmofal. Shock capturing with PDE-based artificial viscosity for DGFEM: Part I. Formulation. *Journal of Computational Physics*, 229(5):1810–1827, 2010.

[5] F. Bassi and S. Rebay. A high-order accurate discontinuous finite element method for the numerical solution of the compressible Navier–Stokes equations. *Journal of Computational Physics*, 131(2):267–279, 1997.

[6] C. E. Baumann and J. T. Oden. A discontinuous hp finite element method for convection-diffusion problems. *Computer Methods in Applied Mechanics and Engineering*, 175(3-4):311–341, 1999.

[7] C. Bernardi, Y. Maday, and G. S. Landriani. Noncorming matching conditions for coupling spectral and finite element methods. *Applied Numerical Mathematics*, 6(1-2):65–84, 1989.

[8] C. Bernardi, Y. Maday, and A. T. Patera. Domain Decomposition by the Mortar Element Method. In *Asymptotic and Numerical Methods for Partial Differential Equations with Critical Parameters*, pages 269–286. Springer, 1993.

[9] J.-P. Berrut and L. N. Trefethen. Barycentric Lagrange interpolation. *SIAM review*, 46(3):501–517, 2004.

[10] N. Besse, E. Deriaz, and É. Madaule. Adaptive multiresolution semi-Lagrangian discontinuous Galerkin methods for the Vlasov equations. *Journal of Computational Physics*, 332:376–417, 2017.

[11] W. Boscheri and M. Dumbser. Arbitrary-Lagrangian–Eulerian discontinuous Galerkin schemes with a posteriori subcell finite volume limiting on moving unstructured meshes. *Journal of Computational Physics*, 346:449–479, 2017.

[12] T. Bui-Thanh and O. Ghattas. Analysis of an hp-nonconforming discontinuous Galerkin spectral element method for wave propagation. *SIAM Journal on Numerical Analysis*, 50(3):1801–1826, 2012.

[13] X. Cai, S. Boscarino, and J.-M. Qiu. High order semi-Lagrangian discontinuous Galerkin method coupled with Runge-Kutta exponential integrators for nonlinear Vlasov dynamics. *Journal of Computational Physics*, 427:110036, 2021.

[14] C. Canuto, M. Y. Hussaini, A. Quarteroni, and T. A. Zang. *Spectral Methods: Fundamentals in Single Domains*. Springer Science & Business Media, 2007.

[15] M. H. Carpenter and C. A. Kennedy. Fourth-order 2N-storage Runge-Kutta schemes. NASA TM 109112, 1994.

[16] N. Chalmers and L. Krivodonova. A robust CFL condition for the discontinuous Galerkin method on triangular meshes. *Journal of Computational Physics*, 403:109095, 2020.

[17] J. Chan, R. J. Hewett, and T. Warburton. Weight-adjusted discontinuous Galerkin methods: curvilinear meshes. *SIAM Journal on Scientific Computing*, 39(6):A2395–A2421, 2017.

[18] J. Chaves. *Introduction to Nonimaging Optics*. CRC press, 2017.

[19] B. Cockburn, S. Hou, and C.-W. Shu. The Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws. IV. The multidimensional case. *Mathematics of Computation*, 54(190):545–581, 1990.

[20] B. Cockburn, S.-Y. Lin, and C.-W. Shu. TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws III: one-dimensional systems. *Journal of Computational Physics*, 84(1):90–113, 1989.

[21] B. Cockburn and C.-W. Shu. TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws. II. General framework. *Mathematics of Computation*, 52(186):411–435, 1989.

[22] B. Cockburn and C.-W. Shu. The local discontinuous Galerkin method for time-dependent convection-diffusion systems. *SIAM Journal on Numerical Analysis*, 35(6):2440–2463, 1998.

[23] B. Cockburn and C.-W. Shu. The Runge–Kutta discontinuous Galerkin method for conservation laws V: multidimensional systems. *Journal of Computational Physics*, 141(2):199–224, 1998.

[24] B. Cockburn and C.-W. Shu. Runge–Kutta discontinuous Galerkin methods for convection-dominated problems. *Journal of Scientific Computing*, 16(3):173–261, 2001.

[25] A. Cvetkovic, O. Dross, J. Chaves, P. Benitez, J. C. Miñano, and R. Mohedano. Etendue-preserving mixing and projection optics for high-luminance LEDs, applied to automotive headlamps. *Optics Express*, 14(26):13014, 2006.

[26] L. Dagum and R. Menon. OpenMP: an industry standard API for shared-memory programming. *Computational Science & Engineering, IEEE*, 5(1):46–55, 1998.

[27] M. Dumbser. Arbitrary-Lagrangian–Eulerian ADER–WENO finite volume schemes with time-accurate local time stepping for hyperbolic conservation laws. *Computer Methods in Applied Mechanics and Engineering*, 280:57–83, 2014.

[28] M. Dumbser, D. S. Balsara, E. F. Toro, and C.-D. Munz. A unified framework for the construction of one-step finite volume and discontinuous Galerkin schemes on unstructured meshes. *Journal of Computational Physics*, 227(18):8209–8253, 2008.

[29] M. Dumbser, C. Enaux, and E. F. Toro. Finite volume schemes of very high order of accuracy for stiff hyperbolic balance laws. *Journal of Computational Physics*, 227(8):3971–4001, 2008.

[30] M. Dumbser, F. Fambri, M. Tavelli, M. Bader, and T. Weinzierl. Efficient implementation of ADER discontinuous Galerkin schemes for a scalable hyperbolic PDE engine. *Axioms*, 7(3):63, 2018.

[31] M. Dumbser, M. Käser, and E. F. Toro. An arbitrary high-order Discontinuous Galerkin method for elastic waves on unstructured meshes-V. Local time stepping and p-adaptivity. *Geophysical Journal International*, 171(2):695–717, 2007.

[32] M. Dumbser and C.-D. Munz. Building blocks for arbitrary high order discontinuous Galerkin schemes. *Journal of Scientific Computing*, 27(1-3):215–230, 2006.

[33] M. Dumbser, O. Zanotti, A. Hidalgo, and D. S. Balsara. ADER-WENO finite volume schemes with space–time adaptive mesh refinement. *Journal of Computational Physics*, 248:257–286, 2013.

[34] L. Einkemmer. High performance computing aspects of a dimension independent semi-Lagrangian discontinuous Galerkin code. *Computer Physics Communications*, 202:326–336, 2016.

[35] L. Einkemmer. A performance comparison of semi-Lagrangian discontinuous Galerkin and spline based Vlasov solvers in four dimensions. *Journal of Computational Physics*, 376:937–951, 2019.

[36] F. Fambri, M. Dumbser, S. Köppel, L. Rezzolla, and O. Zanotti. ADER discontinuous Galerkin schemes for general-relativistic ideal magnetohydrodynamics. *Monthly Notices of the Royal Astronomical Society*, 477(4):4543–4564, 2018.

[37] C. Filosa. *Phase Space Ray Tracing for Illumination Optics*. PhD thesis, Eindhoven University of Technology, 2018.

[38] E. Gaburro, W. Boscheri, S. Chiocchetti, C. Klingenberg, V. Springel, and M. Dumbser. High order direct Arbitrary-Lagrangian-Eulerian schemes on moving Voronoi meshes with topology changes. *Journal of Computational Physics*, 407:109167, 2020.

[39] G. J. Gassner, M. Dumbser, F. Hindenlang, and C.-D. Munz. Explicit one-step time discretizations for discontinuous Galerkin and finite volume schemes based on local predictors. *Journal of Computational Physics*, 230(11):4232–4247, 2011.

[40] F. X. Giraldo and M. Restelli. A study of spectral element and discontinuous Galerkin methods for the Navier–Stokes equations in nonhydrostatic mesoscale atmospheric modeling: Equation sets and test cases. *Journal of Computational Physics*, 227(8):3849–3877, 2008.

[41] A. S. Glassner. *An Introduction to Ray Tracing*. Elsevier, 1989.

[42] J. Glaubitz, A. Nogueira, J. L. Almeida, R. Cantão, and C. Silva. Smooth and compactly supported viscous sub-cell shock capturing for discontinuous Galerkin methods. *Journal of Scientific Computing*, 79:249–272, 2019.

[43] Google. `https://github.com/google/benchmark`. Accessed: 2023-05-24.

[44] D. Gottlieb and J. S. Hesthaven. Spectral methods for hyperbolic problems. *Journal of Computational and Applied Mathematics*, 128(1-2):83–131, 2001.

[45] D. J. Griffiths. *Introduction to Electrodynamics*. American Association of Physics Teachers, 2005.

[46] W. Guo, R. D. Nair, and J.-M. Qiu. A conservative semi-Lagrangian discontinuous Galerkin scheme on the cubed sphere. *Monthly Weather Review*, 142(1):457–475, 2014.

[47] E. Hecht et al. *Optics*, volume 4. Addison Wesley San Francisco, 2002.

[48] A. Heinecke, G. Henry, M. Hutchinson, and H. Pabst. LIBXSMM: accelerating small matrix multiplications by runtime code generation. In *SC'16: Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, pages 981–991. IEEE, 2016.

[49] A. Heinecke, H. Pabst, and G. Henry. LIBXSMM: A high performance library for small matrix multiplications. *Poster and Extended Abstract Presented at SC*, 15, 2015.

[50] A. M. Herkommer. Phase space optics: an alternate approach to freeform optical systems. *Optical Engineering*, 53(3):031304, 2013.

[51] J. S. Hesthaven and T. Warburton. *Nodal Discontinuous Galerkin Methods: Algorithms, Analysis, and Applications*. Springer Science & Business Media, 2007.

[52] Jungbecker. `http://www.jungbecker.de/en/optics/standard-products/`. Accessed: 2023-03-29.

[53] D. I. Ketcheson. Runge-Kutta methods with minimum storage implementations. *Journal of Computational Physics*, 229(5):1763–1773, 2010.

[54] A. Klöckner, T. Warburton, and J. S. Hesthaven. Viscous shock capturing in a time-explicit discontinuous Galerkin method. *Mathematical Modelling of Natural Phenomena*, 6(3):57–83, 2011.

[55] D. A. Kopriva. A conservative staggered-grid Chebyshev multidomain method for compressible flows. II. A semi-structured method. *Journal of Computational Physics*, 128(2):475–488, 1996.

[56] D. A. Kopriva. Metric identities and the discontinuous spectral element method on curvilinear meshes. *Journal of Scientific Computing*, 26:301–327, 2006.

[57] D. A. Kopriva. *Implementing Spectral Methods for Partial Differential Equations: Algorithms for Scientists and Engineers*. Springer Science & Business Media, 2009.

[58] D. A. Kopriva and G. J. Gassner. Geometry effects in nodal discontinuous Galerkin methods on curved elements that are provably stable. *Applied Mathematics and Computation*, 272:274–290, 2016.

[59] D. A. Kopriva, F. J. Hindenlang, T. Bolemann, and G. J. Gassner. Free-stream preservation for curved geometrically non-conforming discontinuous Galerkin spectral elements. *Journal of Scientific Computing*, 79:1389–1408, 2019.

[60] D. A. Kopriva, A. R. Winters, M. Bohm, and G. J. Gassner. A provably stable discontinuous Galerkin spectral element approximation for moving hexahedral meshes. *Computers & Fluids*, 139:148–160, 2016.

[61] D. A. Kopriva, S. L. Woodruff, and M. Y. Hussaini. Computation of electromagnetic scattering with a non-conforming discontinuous spectral element method. *International Journal for Numerical Methods in Engineering*, 53(1):105–122, 2002.

[62] L. G. Leal. *Advanced Transport Phenomena: Fluid Mechanics and Convective Transport Processes*, volume 7. Cambridge University Press, 2007.

[63] X.-H. Lee, I. Moreno, and C.-C. Sun. High-performance LED street lighting using microlens arrays. *Optics Express*, 21(9):10612–10621, 2013.

[64] G. Leobacher and F. Pillichshammer. *Introduction to quasi-Monte Carlo Integration and Applications*. Springer, 2014.

[65] R. M. M. Mattheij, S. W. Rienstra, and J. H. M. ten Thije Boonkkamp. *Partial Differential Equations: Modeling, Analysis, Computation*. SIAM, 2005.

[66] W. R. McCluney. *Introduction to Radiometry and Photometry*. Artech House, 2014.

[67] C. A. A. Minoli and D. A. Kopriva. Discontinuous Galerkin spectral element approximations on moving meshes. *Journal of Computational Physics*, 230(5):1876–1902, 2011.

[68] F. E. Nicodemus. Radiance. *American Journal of Physics*, 31(5):368–377, 1963.

[69] X. Ning, R. Winston, and J. O'Gallagher. Dielectric totally internally reflecting concentrators. *Applied Optics*, 26(2):300–305, 1987.

[70] H. Ohno. Symplectic ray tracing based on Hamiltonian optics in gradient-index media. *JOSA A*, 37(3):411–416, 2020.

[71] B. Owren and M. Zennaro. Derivation of efficient, continuous, explicit Runge–Kutta methods. *SIAM Journal on Scientific and Statistical Computing*, 13(6):1488–1501, 1992.

[72] W. A. Parkyn. Illumination lenses designed by extrinsic differential geometry. In *International Optical Design Conference*, pages LWB–2. Optica Publishing Group, 1998.

[73] P.-O. Persson and J. Peraire. Sub-cell shock capturing for discontinuous Galerkin methods. In *44th AIAA aerospace sciences meeting and exhibit*, page 112, 2006.

[74] J. Qiu, M. Dumbser, and C.-W. Shu. The discontinuous Galerkin method with Lax–Wendroff type time discretizations. *Computer Methods in Applied Mechanics and Engineering*, 194(42-44):4528–4543, 2005.

[75] J.-M. Qiu and C.-W. Shu. Positivity preserving semi-Lagrangian discontinuous Galerkin formulation: Theoretical analysis and application to the Vlasov–Poisson system. *Journal of Computational Physics*, 230(23):8386–8409, 2011.

[76] D. Rausch, M. Rommel, A. M. Herkommer, and T. Talpur. Illumination design for extended sources based on phase space mapping. *Optical Engineering*, 56(6):065103, 2017.

[77] W. H. Reed and T. R. Hill. Triangular mesh methods for the neutron transport equation. Technical report, Los Alamos Scientific Lab., N. Mex.(USA), 1973.

[78] J. Reinders. *Intel threading building blocks: outfitting C++ for multi-core processor parallelism*. O'Reilly Media, Inc., 2007.

[79] M. Restelli, L. Bonaventura, and R. Sacco. A semi-Lagrangian discontinuous Galerkin method for scalar advection by incompressible flows. *Journal of Computational Physics*, 216(1):195–215, 2006.

[80] J. A. Rossmanith and D. C. Seal. A positivity-preserving high-order semi-Lagrangian discontinuous Galerkin scheme for the Vlasov–Poisson equations. *Journal of Computational Physics*, 230(16):6203–6232, 2011.

[81] C.-W. Shu. Discontinuous Galerkin method for time-dependent problems: survey and recent developments. *Recent Developments in Discontinuous Galerkin Finite Element Methods for Partial Differential Equations: 2012 John H Barrett Memorial Lectures*, pages 25–62, 2014.

[82] V. A. Titarev and E. F. Toro. ADER: Arbitrary high order Godunov approach. *Journal of Scientific Computing*, 17(1-4):609–618, 2002.

[83] E. F. Toro and V. A. Titarev. ADER schemes for scalar non-linear hyperbolic conservation laws with source terms in three-space dimensions. *Journal of Computational Physics*, 202(1):196–215, 2005.

[84] E. F. Toro and V. A. Titarev. Derivative Riemann solvers for systems of conservation laws and ADER methods. *Journal of Computational Physics*, 212(1):150–165, 2006.

[85] T. Toulorge and W. Desmet. CFL conditions for Runge-Kutta discontinuous Galerkin methods on triangular grids. *Journal of Computational Physics*, 230(12):4657–4678, 2011.

[86] J. J. van der Vegt and H. van der Ven. Space–time discontinuous Galerkin finite element method with dynamic grid motion for inviscid compressible flows: I. General formulation. *Journal of Computational Physics*, 182(2):546–585, 2002.

[87] R. A. M. van Gestel, M. J. H. Anthonissen, J. H. M. ten Thije Boonkkamp, and W. L. IJzerman. An energy conservative hp-method for Liouville's equation of geometrical optics. *Journal of Scientific Computing*, 89(1):1–35, 2021.

[88] R. A. M. van Gestel, M. J. H. Anthonissen, J. H. M. ten Thije Boonkkamp, and W. L. IJzerman. A hybrid semi-Lagrangian DG and ADER-DG solver on a moving mesh for Liouville's equation of geometrical optics. *Available at SSRN 4435276*, 2023.

[89] R. A. M. van Gestel, M. J. H. Anthonissen, J. H. M. ten Thije Boonkkamp, and W. L. IJzerman. An ADER discontinuous Galerkin method on moving meshes for Liouville's equation of geometrical optics. *Journal of Computational Physics*, page 112208, 2023.

[90] B. S. van Lith. *Principles of Computational Illumination Optics*. PhD thesis, Eindhoven University of Technology, 2017.

[91] B. S. van Lith, J. H. M. ten Thije Boonkkamp, and W. L. IJzerman. Active flux schemes on moving meshes with applications to geometric optics. *Journal of Computational Physics: X*, 3:100030, 2019.

[92] B. S. van Lith, J. H. M. ten Thije Boonkkamp, W. L. IJzerman, and T. W. Tukker. A novel scheme for Liouville's equation with a discontinuous Hamiltonian and applications to geometrical optics. *Journal of Scientific Computing*, 68(2):739–771, 8 2016.

[93] H. Wang and S. Xiang. On the convergence rates of Legendre approximation. *Mathematics of Computation*, 81(278):861–877, 2012.

[94] J. Williamson. Low-storage Runge-Kutta schemes. *Journal of Computational Physics*, 35(1):48–56, 1980.

[95] K. B. Wolf. *Geometric Optics on Phase Space*. Springer Science & Business Media, 2004.

[96] O. Zanotti, F. Fambri, and M. Dumbser. Solving the relativistic magnetohydrodynamics equations with ADER discontinuous Galerkin methods, a posteriori subcell limiting and adaptive mesh refinement. *Monthly Notices of the Royal Astronomical Society*, 452(3):3010–3029, 2015.

[97] O. Zanotti, F. Fambri, M. Dumbser, and A. Hidalgo. Space–time adaptive ADER discontinuous Galerkin finite element schemes with a posteriori sub-cell finite volume limiting. *Computers & Fluids*, 118:204–224, 2015.

[98] X. Zhang and C.-W. Shu. On maximum-principle-satisfying high order schemes for scalar conservation laws. *Journal of Computational Physics*, 229(9):3091–3120, 2010.

[99] X. Zhang and C.-W. Shu. Maximum-principle-satisfying and positivity-preserving high-order schemes for conservation laws: survey and new developments. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 467(2134):2752–2776, 2011.

[100] X. Zhu, Q. Zhu, H. Wu, and C. Chen. Optical design of LED-based automotive headlamps. *Optics & Laser Technology*, 45:262–266, 2013.

[101] D. W. Zingg and T. T. Chisholm. Runge-Kutta methods for linear ordinary differential equations. *Applied Numerical Mathematics*, 31(2):227–238, 1999.

Summary

Discontinuous Galerkin methods for Liouville's equation of geometrical optics

Illumination optics deals with the design of optical systems for illumination purposes, such as street lighting, automotive headlamps and lighting for offices. (Quasi-)Monte Carlo ray tracing is typically employed to compute important quantities, for instance, the illuminance and the luminous intensity, by tracing a large number of light rays through the optical system. (Quasi-)Monte Carlo ray tracing suffers from a slow convergence and can, therefore, be expensive in computing the photometric quantities to high accuracy.

An alternative to tracing many light rays is based on solving Liouville's equation. Liouville's equation for geometrical optics governs the evolution of the basic luminance on phase space. From the basic luminance the illuminance and luminous intensity can be computed by integration. Phase space refers to a collection of positions and momenta representing the direction coordinates of light. Whenever a light ray strikes an optical interface, that is a discontinuity in the refractive index field, its direction coordinates changes discontinuously. This change is governed by the laws of specular reflection or Snell's law of refraction. This results in non-local boundary conditions for the basic luminance at an optical interface.

The main focus of this thesis is the development of solvers for Liouville's equation using discontinuous Galerkin (DG) finite element methods. First, the discretisation for two-dimensional optics, that is a two-dimensional phase space together with an evolution coordinate, is considered. Optical interfaces are incorporated into the method by moving the mesh, so that the mesh is aligned with optical interfaces. The non-local boundary conditions are a difficulty in and of itself, and have to satisfy energy conservation constraints. To that end, a method was developed that conserves energy by satisfying local energy balances discretely. The moving mesh method alone was not sufficient to describe the optical systems of interest, therefore, a special type of element was introduced where an optical interface is allowed to cut this special element into two parts. For temporal discretisation an explicit Runge-

Kutta method and Arbitrary Derivative (ADER) methods have been applied. In an example, it was shown that the ADER-DG scheme can achieve two orders of magnitude lower error in the illuminance compared to quasi-Monte Carlo (QMC) ray tracing within 10 seconds of computation time. For sufficiently smooth solutions, the ADER-DG scheme converges faster to high accuracy than QMC ray tracing.

A new solver has been developed which allows for substantial performance gains compared to the ADER-DG solver. It uses semi-Lagrangian (SL) DG elements away from optical interfaces, and ADER-DG elements close to an interface. The different elements are coupled through a time-accurate local time stepping (LTS) approach, where in this case SLDG elements take a much larger step than ADER-DG elements. In an example, this new solver managed to be faster than the ADER-DG solver by a factor 1.6 to 10 times for computation times up to 4 minutes.

The solvers have been applied to a real-world example, a lens plate that is used in office lighting. In this example, Fresnel reflections have been introduced into the solver. Lastly, the ADER-DG solver is extended to three-dimensional optics where phase space is four dimensional, in addition to the evolution coordinate axis, making it a very challenging problem computationally.

List of Publications

Journal articles

1. R. A. M. van Gestel, M. J. H. Anthonissen, J. H. M. ten Thije Boonkkamp, and W. L. IJzerman. An energy conservative hp-method for Liouville's equation of geometrical optics. *Journal of Scientific Computing*, 89(1):1-35, 2021.

2. R. A. M. van Gestel, M. J. H. Anthonissen, J. H. M. ten Thije Boonkkamp, and W. L. IJzerman. An ADER discontinuous Galerkin method on moving meshes for Liouville's equation of geometrical optics. *Journal of Computational Physics*, 2023.

3. R. A. M. van Gestel, M. J. H. Anthonissen, J. H. M. ten Thije Boonkkamp, and W. L. IJzerman. A hybrid semi-Lagrangian DG and ADER-DG solver on a moving mesh for Liouville's equation of geometrical optics. *submitted to Journal of Computational Physics*, 2023.

Conference contributions

1. R. A. M. van Gestel, M. J. H. Anthonissen, J. H. M. ten Thije Boonkkamp, and W. L. IJzerman. An energy conservative hp-scheme for light propagation using Liouville's equation for geometrical optics. *In European Physical Journal Web of Conferences*, vol. 238, p. 02005. EDP Sciences, 2020.

2. R. A. M. van Gestel, M. J. H. Anthonissen, J. H. M. ten Thije Boonkkamp, and W. L. IJzerman. A discontinuous Galerkin method to solve Liouville's equation of geometrical optics. *In European Physical Journal Web of Conferences*, vol. 266, p. 02005. 2022.

3. R. A. M. van Gestel, M. J. H. Anthonissen, J. H. M. ten Thije Boonkkamp, and W. L. IJzerman. An Energy-Preserving High Order Method for Liouville's Equation of Geometrical Optics. *To appear in Spectral and*

High Order Methods for Partial Differential Equations ICOSAHOM 2020+1, 2023.

Oral presentations at scientific conferences

1. R. A. M. van Gestel, M. J. H. Anthonissen, J. H. M. ten Thije Boonkkamp, and W. L. IJzerman. An energy conservative method on phase space using Liouville's equation for geometrical optics. *TOM 2 - Computational, Adaptive and Freeform Optics, European Optical Society Annual Meeting (EOSAM)*, online conference, 7 - 11 September, 2020.

2. R. A. M. van Gestel, M. J. H. Anthonissen, J. H. M. ten Thije Boonkkamp, and W. L. IJzerman. An energy conservative high order method for Liouville's equation of geometrical optics. *International Conference on Spectral and High Order Methods (ICOSAHOM)*, online conference, 12 - 16 July, 2021.

3. R. A. M. van Gestel, M. J. H. Anthonissen, J. H. M. ten Thije Boonkkamp, and W. L. IJzerman. Computational illumination optics using Liouville's equation. *TOM 2 - Computational, Adaptive and Freeform Optics, European Optical Society Annual Meeting (EOSAM)*, Porto, Portugal, 12 - 16 September, 2022.

In preparation

1. R. A. M. van Gestel, M. J. H. Anthonissen, J. H. M. ten Thije Boonkkamp, and W. L. IJzerman. A parameter study of a 2D lens plate using Liouville's equation of geometrical optics.

2. R. A. M. van Gestel, M. J. H. Anthonissen, J. H. M. ten Thije Boonkkamp, and W. L. IJzerman. An ADER discontinuous Galerkin method on four-dimensional moving meshes for Liouville's equation of geometrical optics.

Curriculum Vitae

Robert van Gestel was born on the 26th of August 1994 in Molenschot, the Netherlands. After finishing his HAVO degree in 2011 at the Onze Lieve Vrouwelyceum in Breda, he studied Applied Physics at Fontys University of Applied Sciences for 1 year. In 2012, he continued his studies in Applied Physics at the Eindhoven University of Technology, obtaining his Master of Science degree in 2019. For his Master's thesis, he developed new flux approximation schemes for systems of conservation laws with applications to multicomponent mixtures. In June 2019, he started a PhD project at the Centre for Analysis, Scientific Computing and Applications (CASA), of which the results can be found in this thesis.

Acknowledgments

A lot of people have supported me over the years, which are too many to name. So I will keep it short.

Martijn, Wilbert and Jan, thank you for supervising and supporting me, and for giving me the freedom in exploring suitable numerical methods.

Thanks to my colleagues at CASA, especially my old and new office mates, for being there and supporting me.

Thanks to my friends for their support and for providing me with the necessary distractions.

My family, mom, dad, my brother and sister, thank you for always being there for me.