An Iterative Least-Squares Method for Generated Jacobian Equations in Freeform Optical Design

Citation for published version (APA):

Document license:
TAVERNE

DOI:
10.1137/20M1338940

Document status and date:
Published: 09/03/2021

Document Version:
Publisher’s PDF, also known as Version of Record (includes final page, issue and volume numbers)

Please check the document version of this publication:
• A submitted manuscript is the version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher’s website.
• The final author version and the galley proof are versions of the publication after peer review.
• The final published version features the final layout of the paper including the volume, issue and page numbers.

Link to publication

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.
• Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
• You may not further distribute the material or use it for any profit-making activity or commercial gain
• You may freely distribute the URL identifying the publication in the public portal.

If the publication is distributed under the terms of Article 25fa of the Dutch Copyright Act, indicated by the “Taverne” license above, please follow below link for the End User Agreement:
www.tue.nl/taverne

Take down policy
If you believe that this document breaches copyright please contact us at:
openaccess@tue.nl
providing details and we will investigate your claim.

Download date: 18. Nov. 2021
AN ITERATIVE LEAST-SQUARES METHOD FOR GENERATED JACOBIAN EQUATIONS IN FREEFORM OPTICAL DESIGN

L. B. ROMIJN\textsuperscript{\dagger}, J. H. M. TEN THIJE BOONKKAMP\textsuperscript{\dagger}, M. J. H. ANTHONISSEN\textsuperscript{\dagger}, AND W. L. IJZERMAN\textsuperscript{\ddagger}

Abstract. The design of freeform optical surfaces is an inverse problem in illumination optics. Combining the laws of geometrical optics and energy conservation gives rise to a generalized Monge–Ampère equation. The underlying mathematical structure of some optical systems allows for an optimal-transport formulation of the problem with an associated cost function. This motivates the design of optimal-transport-based numerical algorithms. However, not all optical systems can be cast in the framework of optimal transport. In this paper, we derive a formulation in terms of generating functions where the generalized Monge–Ampère equation becomes a generated Jacobian equation. We consider two example systems: System 1 is a single freeform lens with a point source and far-field target, and System 2 is a single freeform reflector with a parallel source beam and near-field target. We introduce a novel derivation of the generating functions via Hamilton’s characteristics. We can associate a cost function to System 1, and we compare the performance of the numerical algorithm to a previous optimal-transport-based version. System 2 cannot be formulated as an optimal-transport problem, which demonstrates the wider applicability of the new version of the algorithm to any optical system that can be described by a smooth generating function.

Key words. geometrical optics, optimal mass transport, generated Jacobian equation, generalized Monge–Ampère equation, least-squares method, near-field reflector problem

AMS subject classifications. 35J66, 35J96, 49K20, 65K10, 65N99

DOI. 10.1137/20M1338940

1. Introduction. A common problem across different disciplines in mathematics is that of finding a measure-preserving map from a given space $\mathcal{X}$ to a given space $\mathcal{Y}$, within a certain family of admissible mappings. Examples of this problem are the rearrangement of mass from one distribution into another in an optimal way [5], the analysis of semigeostrophic flows [12], the coupling of probability measures to maximize covariance [49, p. 41], stable matching problems in economics [38], and the inverse design of freeform (i.e., without any symmetries) optical surfaces. Many of these problems concern two open domains $\mathcal{X}, \mathcal{Y} \subset \mathbb{R}^d$ and continuous densities $f : \mathcal{X} \to [0, \infty)$, $g : \mathcal{Y} \to [0, \infty)$. The aim is to find a mapping $m : \mathcal{X} \to \mathcal{Y}$ that maps $f$ into $g$. In other words, for every (Borel) set $A \subset \mathcal{X}$ we have the energy conservation relation

\begin{equation}
\int_A f(x) \, dx = \int_{m(A)} g(y) \, dy,
\end{equation}

where $m$ satisfies the optimality conditions of the problem.
which by substituting the mapping \( y = m(x) \) reduces to the Jacobian equation

\[
(1.2) \quad \det(Dm(x)) = \frac{f(x)}{g(m(x))},
\]

where \( Dm(x) \) is the \( d \times d \) Jacobi matrix of \( m \) with respect to \( x \), assuming \( \det(Dm) > 0 \). In traditional optimal-transport problems, the map must minimize a transportation cost; i.e., we need to find an \( m \) that satisfies (1.2) and minimizes

\[
(1.3) \quad \int_{\mathcal{X}} c(x, m(x)) f(x) \, dx,
\]

where \( c(x, y) \) denotes the transportation cost of one unit of mass, e.g., one unit of luminous flux (1 lm), from \( x \in \mathcal{X} \) to \( y \in \mathcal{Y} \).

There exist the dual Kantorovich potentials \( u_1 : \mathcal{X} \rightarrow \mathbb{R} \) and \( u_2 : \mathcal{Y} \rightarrow \mathbb{R} \) such that

\[
(1.4) \quad u_1(x) = \sup_{y \in \mathcal{Y}} u_2(y) - c(x, y) \quad \text{and} \quad u_2(y) = \inf_{x \in \mathcal{X}} [u_1(x) + c(x, y)],
\]

i.e., \( u_1(x) \) is a c-convex function; cf. the definition introduced in [51, p. 54]. Interchanging the supremum and infimum defines \( u_1(x) \) as a c-concave function. Under certain conditions on the cost function, the mapping \( y = m(x) \) is given implicitly by the stationary point of \( c(x, y) + u_1(x) \), which results in the relation \( \nabla_x c(x, y) + \nabla u_1(x) = 0 \). Substituting the mapping into (1.2) gives a generalized Monge–Ampère equation, which is a fully nonlinear second-order elliptic partial differential equation containing the Hessian matrix \( D^2 u_1(x) \) [51, p. 282].

In freeform optical design, many optical systems can be described by an equation of the form

\[
(1.5) \quad u_2(y) - u_1(x) = c(x, y),
\]

where \( y = m(x) \) and \( u_1 \) is related to the location of the optical surface. In optimal-transport theory, it can be proven that the optical mapping \( m \) satisfying this relation minimizes the transportation cost in (1.3) (and the potentials \( u_1 \) and \( u_2 \) are maximizers for the corresponding dual Kantorovich problem) [51, Theorem 5.10, p. 57].

If we replace the expression \( u_1(x) = u_2(y) - c(x, y) \) with an arbitrary relation \( u_1(x) = G(x, y, u_2(y)) \), where \( G = G(x, y, z) \) is the so-called generating function, we can also derive the mapping and the corresponding generalized Monge–Ampère equation. This equation is called a generated Jacobian equation (GJE). Notions of c-transforms, c-convexity, and regularity in optimal-transport theory naturally extend to GJEs. Trudinger [50] introduced the framework of GJEs and drew examples from geometrical optics which cannot be formulated as optimal-transport problems; i.e., an equation of the form (1.5) cannot be derived using any parametrization of the surface. Prime examples of optical systems which cannot be put in the optimal-transport framework are a parallel source beam or point source reflecting or refracting on a surface in combination with a near-field target.

In this paper, we formulate the generating functions for two optical systems by using a novel derivation via Hamilton’s characteristics. We consider in System 1 a point-to-far-field lens and in System 2 a parallel-to-near-field reflector. We use Hamilton’s characteristic functions of optical path length, which represent the physical constraints of the optical system to connect coordinates of the source domain to
coordinates of the target domain. System 1 can also be viewed from an optimal-
transport point of view, and we can derive an equation of the form (1.5). For System
2, such an equation does not exist.

The GJE can be written in terms of the generating function and solved numeri-
cally by using a least-squares algorithm. Originally, this method was developed for the
standard Monge–Ampère equation [44], considering the parallel-to-far-field problem.
In this case, \( c(x, y) = x \cdot y \), and (1.4) reduces to the Legendre–Fenchel transform and
\( m(x) = \nabla u(x) \). However, for many optical systems the cost function is more
complicated and the optical map is no longer a gradient. The numerical procedure was
extended to nonquadratic cost functions in [56] (parallel-to-parallel double freeform
reflector/lens) and [47, 48] (point-to-far-field single reflector/lens). In this paper, we
further generalize the numerical procedure to a generating-function framework, which
allows us to consider a new range of optical systems that cannot be formulated as
optimal-transport problems. The numerical procedure works by computing the optical
map and optical surface in an iterative procedure which minimizes the global
defect in the energy balance. The optical map and surface are computed with a
formulation involving the unique inverse \( H \) of the generating function \( G \), such that
\( u_1(x) = G(x, y, H(x, y, u_1(x)) \). In fact, the function \( H \) is equal to Hamilton’s char-
acteristic function, i.e., the angular characteristic for System 1 and the point charac-
teristic for System 2. We also impose a transport boundary condition by minimizing
the deviation of the map of the boundary of the source to the boundary of the target.

This paper is structured as follows. In subsection 1.1, we provide an overview
of the literature concerning GJEs in freeform optics. In section 2, we present the
derivation of the generating functions and corresponding GJEs. In section 3, the
generalized least-squares method is described, including a novel step to compute the
optical surface. In section 4, we test the performance of the algorithm. Finally, we
make some concluding remarks in section 5.

1.1. Literature review. Trudinger [50] coined the term GJEs and was mo-
tivated by examples from geometrical optics. The author introduces definitions of
G-convexity, G-transforms, and G-subdifferentials, which are direct analogues of
c-convexity, c-transforms, and c-subdifferentials, respectively, in optimal-transport
theory. Conditions for the existence and regularity of smooth solutions are presented,
directly following the optimal-transport approach in [10, 30]. Local regularity results
are derived from the Urbas–Trudinger–Wang regularity theory [51, p. 318]. More re-
results on the existence and regularity of solutions can be found in [25, 28]. However,
proving the existence, uniqueness, and regularity of solutions to a general GJE is more
complicated than performing analysis on optimal-transport GJEs; see, for instance,
the point-to-near-field reflector problem in [29]. A brief overview of open problems in
the analysis of GJEs is given in [24].

We mention some results on the mathematical analysis of the GJEs corresponding
to the optical systems we consider in this paper. The existence and uniqueness of weak
solutions to the point-to-far-field lens system (System 1) is established in [26, 27]. The
authors also treat the near-field case. For the same problem with a reflector instead
of a lens, conditions were derived in [21, 22, 33, 41, 52, 53]. The existence of globally
smooth solutions for the parallel-to-near-field reflector was established in [33], noting
extensions to the lens problem. (The corresponding far-field case results in the stan-
dard Monge–Ampère equation, for which the existence, uniqueness, and regularity of
solutions is well established by Brenier’s theorem [5].) While a formulation as a linear
Kantorovich problem for System 2 is not possible, a formulation as a nonlinear Kan-
torovich problem and the corresponding generalized Monge–Ampère equation were
In geometrical optics, there exists a wide range of numerical algorithms solving the standard Monge–Ampère equation \([2, 20, 45, 55]\) and generalized Monge–Ampère equations. For generalized Monge–Ampère equations, the numerical strategies can be roughly categorized as (1) methods that directly solve the generalized Monge–Ampère equation \([4, 6, 7, 54]\), (2) optimization strategies for the corresponding Monge–Kantorovich mass transportation problem \([8, 16, 17, 22, 26, 27, 39, 40, 42, 47, 48, 53, 55, 56]\), and (3) methods which indirectly compute the surface by using ray mapping techniques \([11, 15, 18, 19, 35, 36]\). For an in-depth overview, we refer the reader to \([48]\).

To the best of our knowledge, there is one numerical procedure which attempts to find the solution to general GJEs. Abedin and Gutiérrez \([1]\) find solutions to general GJEs by generalizing the method of De Leo, Gutiérrez, and Mawi \([14]\). De Leo, Gutiérrez, and Mawi \([14]\) present the point-to-far-field lens problem with a discrete target and use Oliker’s method of supporting ellipsoids/paraboloids, originally proposed in \([43]\). The idea is to construct a freeform reflector or lens for a point source from a union or intersection of a set of supporting ellipsoids, each having one focus located at the point source and the other one at a discrete target point \(y_i\) in the near field. In the case of a far-field target, the ellipsoids of revolution converge to paraboloids of revolution whose focus is located at the origin. The iterative scheme works by optimizing the polar radius of each supporting quadric surface and is shown to converge within a finite number of iterations. An iterative optimization algorithm was introduced in \([9, 31]\) and further developed, extended, and applied in \([11, 13, 19, 36]\). Abedin and Gutiérrez \([1]\) redesign the algorithm proposed in \([14]\) to consider general GJEs by defining the supporting surfaces as graphs of the generating function \(G(x, y, z_i)\) for each discrete target point \(y_i\). The algorithm iteratively optimizes \(z_i\) for each \(y_i\) by minimizing the discrepancy between the integral of the source intensity over all points \(x\) on the inverse mapping of \(y_i\) and the discrete target intensity at the point \(y_i\). The resulting solution of the GJE is formed by taking the intersection of the supporting graphs \(G(x, y_i, z_i)\). While the authors present an application of the algorithm to the parallel-to-near-field reflector problem, they only check that the generating function satisfies the assumptions required to establish convergence of the algorithm.

2. Mathematical formulation. In this section, we derive the generating functions of two optical systems, which we will combine with energy conservation to derive the generalized Monge–Ampère equations. We consider two optical systems: System 1 is a lens surface for a point source and far-field target intensity, and System 2 is a reflector surface for a parallel incoming beam and near-field target illuminance. Figure 1 schematically illustrates both systems.

**System 1:** A cone of light emanates from the point source located at \(O\) of the Cartesian coordinate system with \((x, y, z) \in \mathbb{R}^3\). The point source emits rays of light radially outward in the direction \(\hat{s} = \hat{e}_r\), where \(\hat{e}_r\) is the radial basis vector in the spherical coordinate system. The light rays are unaltered by the first spherical surface of the lens, which has refractive index \(n\). The freeform lens surface \(\mathcal{L}\) refracting \(\hat{s}\) into the direction \(\hat{t}\) is described by the parametric equation \(\mathcal{L} : r(\phi, \theta) = u(\phi, \theta) \hat{e}_r\), where \(u(\phi, \theta) > 0\) is the radial parameter that describes the location of the lens surface, \(0 \leq \phi \leq \pi\) is the zenith, and \(0 \leq \theta < 2 \pi\) is the azimuth in the spherical coordinate system. The intensity of the source is given by \(f(\phi, \theta) \text{ [lm/sr]}\), and the required target intensity in the far field is denoted by \(g(\psi, \chi) \text{ [lm/sr]}\), where \((\psi, \chi)\) represents a different set of spherical coordinates, with zenith \(0 \leq \psi \leq \pi\) and azimuth \(0 \leq \chi < 2 \pi\). The origin of the coordinate system describing the target is the lens.
Point source with intensity \(f(\phi, \theta) \text{ [lm/sr]}\)

Target with required intensity \(g(\psi, \chi) \text{ [lm/sr]}\)

(a) System 1: Point-to-far-field lens

Parallel beam with emittance \(f(x) \text{ [lm/m}^2\text{]}\)

Target with required illuminance \(g(y) \text{ [lm/m}^2\text{]}\)

(b) System 2: Parallel-to-near-field reflector

Fig. 1. (a) System 1: Single lens converting the intensity \(f(\phi, \theta)\) of a point source into a far-field target intensity \(g(\psi, \chi)\). (b) System 2: Single reflector converting the emittance \(f(x)\) of a parallel beam into a near-field target illuminance \(g(y)\).

surface approximated as a point in space (i.e., all directions \(\hat{t}\) emanate from the same point). This approximation is called the far-field approximation. We transform coordinates on the source and target domains from spherical to stereographic. This is convenient since the vectors \(\hat{s} = (s_1, s_2, s_3)^T\) and \(\hat{t} = (t_1, t_2, t_3)^T\) are defined on the unit sphere \(S^2\). Hence, \(|\hat{s}| = |\hat{t}| = 1\). We define

\[
(2.1a) \quad x(\hat{s}) = \frac{1}{1 + s_3} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \frac{1}{1 + \cos(\phi)} \begin{pmatrix} \sin(\phi) \cos(\theta) \\ \sin(\phi) \sin(\theta) \end{pmatrix},
\]

\[
(2.1b) \quad y(\hat{t}) = \frac{1}{1 + t_3} \begin{pmatrix} y_1 \\ y_2 \end{pmatrix} = \frac{1}{1 + \cos(\psi)} \begin{pmatrix} \sin(\psi) \cos(\chi) \\ \sin(\psi) \sin(\chi) \end{pmatrix},
\]

with corresponding inverse projections

\[
(2.2) \quad \hat{s}(x) = \hat{e}_r = \frac{1}{1 + |x|^2} \begin{pmatrix} 2x_1 \\ 2x_2 \\ 1 - |x|^2 \end{pmatrix}, \quad \hat{t}(y) = \frac{1}{1 + |y|^2} \begin{pmatrix} 2y_1 \\ 2y_2 \\ 1 - |y|^2 \end{pmatrix}.
\]

We represent the incoming rays \(\hat{s}\) and the outgoing rays \(\hat{t}\) by using stereographic projections from the south pole \((0, 0, -1)\) of \(S^2\) onto the plane \(z = 0\). The stereographic projections in (2.1a) and (2.1b) are undefined at the south pole, and we consider \(s_3, t_3 \neq -1\) and \(0 \leq \phi, \psi < \pi\). For more details, see [48]. We define our source domain \(\mathcal{X}\) as the supporting domain of \(\hat{f}(x) = f(\phi(x), \theta(x))\), and our target domain \(\mathcal{Y}\) as the image under the mapping \(m\), i.e., \(\mathcal{Y} = m(\mathcal{X})\), where we let \(\hat{g}(y) = g(\psi(y), \chi(y))\). We refer to \(m : \mathcal{X} \rightarrow \mathcal{Y}\) as the optical map \(y = m(x)\) from the source set of stereographic coordinates \(\mathcal{X}\) to the target set of stereographic coordinates \(\mathcal{Y}\).

**System 2**: We consider a parallel-to-near-field reflector. The light source is a parallel beam in the positive \(z\)-direction of the Cartesian coordinate system with
GENERATED JACOBIAN EQUATIONS

\( z = 0 \)

\( L : r(\phi, \theta) = u(\phi, \theta) \hat{e}_r \)

\( O \)

\( P \)

\( Q \)

\( Q' \)

\( q_t \)

\( \hat{s} \)

\( \hat{t} \)

\( n \)

\( \hat{n} \)

\( q_s \)

\( z = 0 \)

\( R : z = u(x) \)

\( O \)

\( P \)

\( Q \)

\( \hat{s} \)

\( \hat{t} \)

\( n \)

\( \hat{n} \)

\( q_t \)

\( q_s \)

\( (a) \) System 1: Point-to-far-field lens

\( (b) \) System 2: Parallel-to-near-field reflector

Fig. 2. Sketch of Systems 1 and 2 showing the points P and Q used for the derivation of Hamilton’s characteristics.

\((x_1, x_2, z) \in \mathbb{R}^3\), and we denote \( x = (x_1, x_2) \). The source emits light in the direction \( \hat{s} = \hat{e}_z \). The surface of the reflector is described by \( \mathcal{R} : z = u(x) \). The surface \( \mathcal{R} \) reflects the ray \( \hat{s} \) in direction \( \hat{t} \). We are given an emittance of the source domain \( f(x) \) in \([\text{lm}/\text{m}^2]\) and a target illuminance in the near field given by \( g(y) \) in \([\text{lm}/\text{m}^2]\), where \( x \) and \( y \) are the local coordinates of the source and target planes, respectively. Analogous to System 1, we define our source domain \( \mathcal{X} \) as the supporting domain of \( f \), and our target domain \( \mathcal{Y} \) as the image under the mapping \( \mathbf{m} \), i.e., \( \mathcal{Y} = \mathbf{m}(\mathcal{X}) \).

In this paper, we will derive the optical mapping \( \mathbf{m} \) and corresponding surfaces \( u \) using the generating functions of the optical systems. Note that similar notations are used for both systems; it will be clear from the context which optical system we are considering. For System 1, \( x \in \mathcal{X} \) and \( y \in \mathcal{Y} \) denote the stereographic coordinates of the source and target domains, respectively. For System 2, \( x \in \mathcal{X} \) and \( y \in \mathcal{Y} \) denote the local Cartesian coordinates of the source and target domains, respectively.

2.1. The generating-function approach. In this section, we derive the generating functions for Systems 1 and 2. We use Hamilton’s characteristic functions, which are measures of the optical path length between specified source and target planes [3, 34]. The characteristics are classified as the point characteristic \( V \) (equal to the optical path length between two points), the two mixed characteristics \( W \) and \( W^* \), and the angular characteristic \( T \).

Figure 2 displays Systems 1 and 2 schematically. An incident ray propagates in the direction \( \hat{s} \) which intercepts the freeform surface at the point \( P \) and reflects/refracts in the direction \( \hat{t} \). We assume a target plane at \( z = 0 \), where the ray intersects at point \( Q \).

2.1.1. System 1: Point-to-far-field lens. The position and direction coordinate vectors on the source plane are given by the two-vectors \( q_s = 0 \) and \( p_s = (n s_1, n s_2)^T \), respectively, with \( n \) the refractive index. The position and direction coordinates on the target plane are given by \( q_t \) and \( p_t = (t_1, t_2)^T \), respectively. We define \( p^*_s = (s_1, s_2)^T \). Hence, the point \( P \) where the incoming ray hits the freeform surface of the lens, as shown schematically in Figure 2a, has coordinates \( P(u(\hat{s}) p^*_s, u(\hat{s}) s_3) \).

The point characteristic (or optical path length) between point \( O(q_s, 0) \) and the
virtual point \( Q(q_t, 0) \) is given by
\[
V(q_s, q_t) = n u(\hat{s}) - d(P, Q),
\]
(2.3)
\[
d(P, Q) = \sqrt{|q_t - u(\hat{s}) p^*_s|^2 + (-u(\hat{s}) s_3)^2},
\]
where \( n u(\hat{s}) \) is the optical path length from \( O \) to \( P \). \( d(P, Q) \) denotes the Euclidean distance between \( P \) and \( Q \). Note that the minus sign in front of \( d(P, Q) \) is a consequence of \( Q \) being a virtual image point.

Hamilton’s angular characteristic, which depends on the direction of the ray at the source plane and the direction at the target plane, is given by
\[
T(p_s, p_t) = V(q_s, q_t) + q_s \cdot p_s - q_t \cdot p_t.
\]
(2.4)
Geometrically, the angular characteristic can be interpreted as the optical path length between the intersection of the source ray with the plane perpendicular to it going through the origin of the source plane, and the intersection of the target ray with the plane perpendicular to it going through the origin of the target plane. More specifically, in our case the angular characteristic is the optical path length from \( O \) to \( Q' \) in Figure 2a. Mathematically, the angular characteristic is obtained by applying two Legendre transformations to the point characteristic. For more details, we refer the reader to [34, p. 104] and [3, p. 137].

The spatial coordinate \( q_s \) at the source plane and spatial coordinate \( q_t \) at the target plane are given as [34, p. 105]
\[
q_s = \frac{\partial T}{\partial p_s} = 0, \quad q_t = -\frac{\partial T}{\partial p_t}.
\]
(2.5)
By the first equation we conclude that the angular characteristic \( T \) is independent of the direction coordinate \( p_s \), and the expression for \( T \) reads as
\[
T(p_t) = V(q_s, q_t) - q_s \cdot p_t = n u(\hat{s}) - d(P, Q) - q_t \cdot p_t.
\]
(2.6)
We can derive
\[
p_t = -\frac{q_t - u(\hat{s}) p^*_s}{d(P, Q)}, \quad t_3 = -\frac{0 - u(\hat{s}) s_3}{d(P, Q)} = \frac{u(\hat{s}) s_3}{d(P, Q)},
\]
(2.7)
and
\[
T(p_t) = n u(\hat{s}) - \frac{1}{d(P, Q)} \left[ |q_t - u(\hat{s}) p^*_s|^2 - q_t \cdot (q_t - u(\hat{s}) p^*_s) + (u(\hat{s}) s_3)^2 \right]
\]
(2.8)
\[= n u(\hat{s}) - u(\hat{s}) (p_t \cdot p^*_s + s_3 t_3) = u(\hat{s})(n - \hat{s} \cdot \hat{t}).\]

Note that \( n - \hat{s} \cdot \hat{t} > 0 \) for \( n > 1 \). Solving this equation for \( u(\hat{s}) \) we obtain
\[
u(\hat{s}) = \frac{T(p_t)}{n - \hat{s} \cdot \hat{t}}.
\]
(2.9)
Changing to stereographic coordinates using (2.2) gives
\[
u(x) = T(p_t) \left( n - 1 + \frac{2 |x - y|^2}{(1 + |x|^2)(1 + |y|^2)} \right)^{-1},
\]
(2.10)
where, for ease of notation, we continue to use the variable $u$ to represent the optical surface, but now as a function of $x$. Now, we construct the generating function from the relation $u(x) = G(x, y, z)$, with $z = T(p_i)$, as

$$G(x, y, z) = z \left( n - 1 + \frac{2 |x - y|^2}{(1 + |x|^2)(1 + |y|^2)} \right)^{-1}. \tag{2.11}$$

Note that $z = T(p_i)$ is dependent on the outgoing ray $i$ and hence $z = z(y)$ is a function of $y$.

Previously [47, 48], we introduced $u_1(x) = \log(u(x))$ and $-u_2(y) = \log(T(p_i))$ to rewrite (2.10) and derive a relation of the form $u_1(x) + u_2(y) = c(x, y)$. This results in a c-convex solution defined as a sup/sup pair and a concave pair defined as an inf/inf pair. However, in this paper we use the definition in (1.4) by letting $u_1(x) = -\log(u(x))$ such that

$$u_2(y) - u_1(x) = -\log \left( n - 1 + \frac{2 |x - y|^2}{(1 + |x|^2)(1 + |y|^2)} \right) = c(x, y), \tag{2.12}$$

where $c(x, y)$ is a logarithmic cost function in optimal-transport theory. This is a relation of the form $u_2(y) - u_1(x) = c(x, y)$ for the location of the optical surface $u$, where $u_1(x) = -\log(u(x))$.

### 2.1.2. System 2: Parallel-to-near-field reflector.

The position and direction coordinates on the source plane are given by the two-vectors $q_s$, respectively. The position and direction coordinates on the target plane are given by $q_t = y$ and $p_t = (t_1, t_2)^T$, respectively. The point $P$ where the incoming ray hits the reflector is given by $P(x, u(x))$, as shown schematically in Figure 2b.

We choose the source plane to coincide with the target plane. The point characteristic between a point $(x, 0)$ on the source plane and $Q(y, 0)$ on the target plane is given by

$$V(q_s, q_t) = u(x) + d(P, Q), \tag{2.13}$$

where $u(x)$ is the optical path length from $(x, 0)$ to $P$, and $d(P, Q)$ denotes the optical path length between $P$ and $Q$, which is equal to the Euclidean distance.

The direction coordinate $p_s$ at the source plane and direction coordinate $p_t$ at the target plane are given as [34, p. 98]

$$p_s = -\frac{\partial V}{\partial q_s} = 0, \quad p_t = \frac{\partial V}{\partial q_t}. \tag{2.14}$$

By the first equation we conclude that the point characteristic $V$ is independent of the space coordinate $q_s$. Hence,

$$V(q_t) = u(x) + \sqrt{|y - x|^2 + u(x)^2}. \tag{2.15}$$

Solving this equation for $u(x)$ we obtain

$$u(x) = \frac{1}{2} V(q_t) - \frac{1}{2} \frac{V(q_t)}{V(q_t)} |x - y|^2. \tag{2.16}$$
(Note that if we invert this equation again to get an expression for $V(q_i)$ we obtain two solutions. By defining $Q$ as a real image, i.e., light rays are reflected back onto the source plane, we restrict ourselves to the positive solution.)

Now, we construct the generating function from the relation $u(x) = G(x, y, z)$ with $z = V(q_i)$ as presented in [28];

$$
G(x, y, z) = \frac{z}{2} - \frac{1}{2z} |x - y|^2,
$$

which is the same function as the one presented in [28] if we replace $z$ by $-z$ (and as in [50] if we replace $z$ by $1/z$). Note that $z = z(y)$ is a function of $y$.

### 2.1.3. G-convex functions.

The equation $G(x, y, z) = u(x)$ with $G$ given in (2.11) or (2.17) has many solutions for $u(x)$. We let $u_1(x) = u(x)$ and $u_2(y) = z(y)$.

Then we have

$$
\forall x \in \mathcal{X}, \forall y \in \mathcal{Y} : 
\quad u_1(x) = G(x, y, u_2(y)).
$$

We define $H(x, y, G(x, y, u_2(y))) = u_2(y)$ as the unique inverse of $G$ for a given $x \in \mathcal{X}$ and $y \in \mathcal{Y}$, assuming a unique inverse exists. Then, for all $x \in \mathcal{X}, y \in \mathcal{Y}$ we have that

$$
\forall x \in \mathcal{X} : 
\quad u_1(x) = G(x, y, u_2(y)) \iff u_2(y) = H(x, y, u_1(x)),
$$

i.e., for fixed $x, y$, we have that $G(x, y, \cdot)$ and $H(x, y, \cdot)$ are each other’s inverses.

Hence, we get that for all $x \in \mathcal{X}, y \in \mathcal{Y}, z \in \mathbb{R}$ we have $G_z > 0$ or $G_z < 0$, since $G$ should be injective with respect to the third argument ($G_z = 0$ is possible at an isolated point).

For System 1, the function $H$ is the angular characteristic $T(p_i)$ rewritten in stereographic coordinates (cf. (2.10) and (2.11)), i.e.,

$$
H(x, y, z) = z \left(n - 1 + \frac{2 |x - y|^2}{(1 + |x|^2)(1 + |y|^2)}\right).
$$

For System 2, the function $H$ is the point characteristic $V(q_i)$ (cf. (2.16) and (2.17)), i.e.,

$$
H(x, y, z) = z + \sqrt{|y - x|^2 + z^2}.
$$

We can find a unique solution by assuming that $u_1$ is a G-convex (or G-concave) function. The function $u_1(x)$ is G-convex and $u_2(y)$ is H-concave if

\begin{align*}
(2.22a) \quad & \forall x \in \mathcal{X} : 
\quad u_1(x) = \max_{y \in \mathcal{Y}} G(x, y, u_2(y)), \\
(2.22b) \quad & \forall y \in \mathcal{Y} : 
\quad u_2(y) = \min_{x \in \mathcal{X}} H(x, y, u_1(x)),
\end{align*}

or $u_1(x)$ is G-concave and $u_2(y)$ is H-convex if

\begin{align*}
(2.23a) \quad & \forall x \in \mathcal{X} : 
\quad u_1(x) = \min_{y \in \mathcal{Y}} G(x, y, u_2(y)), \\
(2.23b) \quad & \forall y \in \mathcal{Y} : 
\quad u_2(y) = \max_{x \in \mathcal{X}} H(x, y, u_1(x)).
\end{align*}

We replaced the supremum and infimum in the usual definition by a maximum and minimum, respectively, by assuming that $\mathcal{X}$ and $\mathcal{Y}$ are compact. In section SM1 in
the supplementary materials to this article, linked from the main article webpage, we show that for both systems we have the property \(G_z > 0\), which results in the max/min pair (2.22) for a G-convex solution and in the min/max pair (2.23) for a G-concave solution.

Since we need an equation for \(u_1\), rather than using the generating function \(G\) to derive an expression for the mapping and the associated GJE, as in [25, 28, 50], we proceed by directly using \(H\), i.e., Hamilton's characteristic function.

By the implicit function theorem the mapping \(y = m(x, u_1(x), \nabla u_1(x))\) is given implicitly as the critical point of (2.22b) or (2.23b), i.e.,

\[
\nabla_x H(x, y, u_1(x)) + H_z(x, y, u_1(x)) \nabla u_1(x) = 0.
\]

For simplicity, since \(u_1 = u_1(x)\), we define \(H^*(x, y) = H(x, y, u_1(x))\) and rewrite (2.24) in the shorter form

\[
\nabla_x H^*(x, y) = 0,
\]

and we use the implicit function theorem to denote the mapping \(y = m(x)\) as a function of \(x\) only.

A sufficient condition for a minimum in (2.22b) or a maximum in (2.23b) is for the Hessian matrix \(-D_{xx}H^*(x, m(x)) = P\) to be symmetric negative definite (SND) or symmetric positive definite (SPD), respectively. For an SND matrix, we need \(\text{tr}(P) \leq 0\) and \(\text{det}(P) \geq 0\). On the other hand, for an SPD matrix we need \(\text{tr}(P) \geq 0\) and \(\text{det}(P) \geq 0\). Note that \(P\) is symmetric. In this paper, we choose to compute G-convex solutions and require \(P\) to be SND.

Substituting \(y = m(x)\) in (2.25) and differentiating again with respect to \(x\) gives

\[
D_{xx}H^*(x, m(x)) + D_{xy}H^*(x, m(x)) \, \text{D} m(x) = 0,
\]

where \(D_{xx}H^*\) is the Hessian matrix of \(H^*\) with respect to \(x\), \(D_{xy}H^*\) is the matrix of mixed second-order partial derivatives with respect to \(x\) and \(y\), and \(\text{D} m(x)\) is the \(2 \times 2\) Jacobi matrix of \(m\) with respect to \(x\). Using that \(-D_{xx}H^*(x, m(x)) = P\), we find

\[
P = D_{xy}H^*(x, m(x)) \, \text{D} m(x)
\]

(2.27)

\[
= \left( D_{xy} H(x, y, u_1(x)) + \nabla u_1(x) \, (\nabla_y H_z(x, y, u_1(x)))^T \right) \, \text{D} m(x).
\]

We define the matrix \(C = D_{xy}H^*(x, y)\), which we call the mixed Hessian matrix, and rewrite (2.27) as

\[
P(x) = C(x, m(x), u_1(x)) \, \text{D} m(x).
\]

Assuming the mixed Hessian matrix \(C\) is invertible, the mapping \(m(x)\) is given by the critical point of (2.25). In the following section, we derive the generalized Monge–Ampère equation by combining the matrix equation (2.28) with energy conservation.

2.2. Energy conservation.

2.2.1. System 1: Point-to-far-field lens. We require that all light from the source ends up at the target and that energy is conserved, i.e.,

\[
\int_A f(\phi, \theta) \, dS(\phi, \theta) = \int_{\hat{\mathcal{T}}_A} g(\psi, \chi) \, dS(\psi, \chi),
\]

(2.29)
for an arbitrary set $\mathcal{A} \subset S^2$ and image set $\mathcal{I}(\mathcal{A}) \subset S^2$. Note that this image set corresponds to the far-field approximation. If we substitute $\mathcal{s} = \mathcal{s}(\mathcal{x})$ and $\mathcal{t} = \mathcal{t}(\mathcal{y})$ from (2.2), we can write (2.29) as

\begin{equation}
(2.30) \quad \int_{\mathcal{A}(\mathcal{A})} \mathcal{f}(\mathcal{x}) \left| \frac{\partial \mathcal{s}}{\partial x_1} \times \frac{\partial \mathcal{s}}{\partial x_2} \right| \, d\mathcal{x} = \int_{\mathcal{I}(\mathcal{A})} \mathcal{g}(\mathcal{y}) \left| \frac{\partial \mathcal{t}}{\partial y_1} \times \frac{\partial \mathcal{t}}{\partial y_2} \right| \, d\mathcal{y}.
\end{equation}

Note that for global energy conservation we choose $\mathcal{x}(\mathcal{A}) = \mathcal{X}$ and $\mathcal{y}(\mathcal{I}(\mathcal{A})) = \mathcal{Y}$. We can derive that

\begin{equation}
(2.31) \quad \left| \frac{\partial \mathcal{s}}{\partial x_1} \times \frac{\partial \mathcal{s}}{\partial x_2} \right| = \frac{4}{(1 + |\mathcal{x}|^2)^2}, \quad \left| \frac{\partial \mathcal{t}}{\partial y_1} \times \frac{\partial \mathcal{t}}{\partial y_2} \right| = \frac{4}{(1 + |\mathcal{y}|^2)^2}.
\end{equation}

Substituting (2.31) and the mapping $\mathcal{y} = \mathcal{m}(\mathcal{x})$ into the energy conservation relation (2.30) gives

\begin{equation}
(2.32) \quad \int_{\mathcal{A}(\mathcal{A})} \frac{4 \mathcal{f}(\mathcal{x})}{(1 + |\mathcal{x}|^2)^2} \, d\mathcal{x} = \int_{\mathcal{X}(\mathcal{A})} \frac{4 \mathcal{g}(\mathcal{m}(\mathcal{x}))}{(1 + |\mathcal{m}(\mathcal{x})|^2)^2} |\det(\mathcal{m}(\mathcal{x}))| \, d\mathcal{x}.
\end{equation}

We can rewrite (2.32) as the generalized Monge–Ampère equation

\begin{equation}
(2.33) \quad \det(\mathcal{m}(\mathcal{x})) = \frac{\mathcal{f}(\mathcal{x}) (1 + |\mathcal{m}(\mathcal{x})|^2)^2}{\mathcal{g}(\mathcal{m}(\mathcal{x})) (1 + |\mathcal{x}|^2)^2} = F_1(\mathcal{x}, \mathcal{m}(\mathcal{x}), u_1(\mathcal{x})),
\end{equation}

where we omit the absolute value sign of the determinant and restrict ourselves to a positive Jacobian of the mapping, and we introduce $F_1(\mathcal{x}, \mathcal{m}(\mathcal{x}), u_1(\mathcal{x}))$ to equal the total right-hand side. Using matrix equation (2.28) this can be rewritten as

\begin{equation}
(2.34) \quad \frac{\det(P(\mathcal{x}))}{\det(C(\mathcal{x}, \mathcal{m}(\mathcal{x}), u_1(\mathcal{x})))} = F_1(\mathcal{x}, \mathcal{m}(\mathcal{x}), u_1(\mathcal{x})).
\end{equation}

\subsection{2.2.2. System 2: Parallel-to-near-field reflector.}

We have the energy balance

\begin{equation}
(2.35) \quad \int_{\mathcal{B}} \mathcal{f}(\mathcal{x}) \, d\mathcal{x} = \int_{\mathcal{Y}(\mathcal{B})} \mathcal{g}(\mathcal{y}) \, d\mathcal{y}
\end{equation}

for an arbitrary set $\mathcal{B} \subset \mathcal{X}$ and image set $\mathcal{y}(\mathcal{B}) \subset \mathcal{Y}$. Substituting the mapping $\mathcal{y} = \mathcal{m}(\mathcal{x})$ gives

\begin{equation}
(2.36) \quad \int_{\mathcal{B}} \mathcal{f}(\mathcal{x}) \, d\mathcal{x} = \int_{\mathcal{Y}(\mathcal{B})} \mathcal{g}(\mathcal{m}(\mathcal{x})) |\det(\mathcal{m}(\mathcal{x}))| \, d\mathcal{x}.
\end{equation}

Analogous to the previous section, we can rewrite (2.36) to the generalized Monge–Ampère equation

\begin{equation}
(2.37) \quad \frac{\det(P(\mathcal{x}))}{\det(C(\mathcal{x}, \mathcal{m}(\mathcal{x}), u_1(\mathcal{x})))} = \frac{\mathcal{f}(\mathcal{x})}{\mathcal{g}(\mathcal{m}(\mathcal{x}))} = F_2(\mathcal{x}, \mathcal{m}(\mathcal{x}), u_1(\mathcal{x})),
\end{equation}

where we introduce $F_2(\mathcal{x}, \mathcal{m}(\mathcal{x}), u_1(\mathcal{x}))$ to equal the total right-hand side.
2.2.3. **Transport boundary condition.** We define the corresponding transport boundary condition to (2.34) and (2.37) as

\begin{equation}
\mathbf{m}(\partial \mathcal{X}) = \partial \mathcal{Y},
\end{equation}

stating that all light from the boundary of the source \( \mathcal{X} \) is mapped to the boundary of the target \( \mathcal{Y} \) [44, 45], which is a consequence of the edge-ray principle [46] and explained in detail in [47]. Equation (2.38) is equivalent, under some restrictions, to \( \mathbf{m}(\mathcal{X}) = \mathcal{Y} \), stating that all the light from the source arrives at the target. A necessary and sufficient condition for the equivalence is that \( \mathbf{m} \) is bijective, which requires certain convexity/concavity assumptions on the surface \( u \). For the optical systems discussed in this paper, the derivations of these assumptions are highly non-trivial. However, in the numerical procedure in this paper we locally enforce energy conservation by a one-to-one correspondence, assuming that the optical mapping \( \mathbf{m} \) is globally bijective, and use the transport boundary condition (2.38). Frequently, (2.38) is written as \( \mathbf{m}(\partial \mathcal{X}) \subset \partial \mathcal{Y} \), but since we require \( \mathcal{Y} \) to be the image of \( \mathcal{X} \) under the mapping \( \mathbf{m} \) we use an equal sign.

3. **Numerical method.** We compute the mapping \( \mathbf{m} \) and surface \( u_1 \) from (2.34) or (2.37) by using a generalized least-squares method [44, 45, 56, 47, 48]. The mapping \( \mathbf{m} \) and surface \( u_1 \) can be calculated efficiently by an iterative procedure that involves finding the numerical solution of a constrained minimization problem (pointwise), imposing the transport boundary condition (pointwise), computing the numerical solution of a linear elliptic boundary value problem, and solving a Neumann problem.

In this section, we give a brief overview of the generalized least-squares method. In previous work [47, 48, 56], we first computed \( \mathbf{m} \) in an iterative procedure and subsequently calculated \( u_1 \) from the converged mapping. The extension of the least-squares algorithm to a generating-function approach lies in the additional dependency of the mixed Hessian matrix \( \mathbf{C} \) in (2.34) and (2.37) on the surface \( u_1(x) \).

To compute the mapping \( \mathbf{m} \), we write the Monge–Ampère equations (2.34) and (2.37) as the matrix equation (2.28), with \( \mathbf{P}(x) \) an SND matrix satisfying \( \det(\mathbf{P}(x)) = F_i(x, \mathbf{m}(x), u_1(x)) \det(\mathbf{C}(x, \mathbf{m}(x), u_1(x))) \) with \( i = 1, 2 \). We write \( \mathbf{m} = \mathbf{m}(x) \) and enforce the matrix equation (2.28) by minimizing the functional

\begin{equation}
J_I[\mathbf{m}, \mathbf{P}] = \frac{1}{2} \int_{\mathcal{X}} \| \mathbf{C} \mathbf{Dm} - \mathbf{P} \|^2 \, d\mathbf{x},
\end{equation}

under the constraint \( \det(\mathbf{P}) = F_i \det(\mathbf{C}) \). The norm used is the Frobenius norm. To impose the transport boundary condition (2.38), we minimize the functional

\begin{equation}
J_B[\mathbf{m}, \mathbf{b}] = \frac{1}{2} \oint_{\partial \mathcal{X}} |\mathbf{m} - \mathbf{b}|^2 \, ds
\end{equation}

over \( \mathbf{b} \), where \( |\cdot| \) denotes the \( L_2 \)-norm and \( \mathbf{b} \) is a function from the source boundary to the target boundary, i.e., \( \mathbf{b} : \partial \mathcal{X} \rightarrow \partial \mathcal{Y} \). By minimizing this functional we aim to impose \( \mathbf{m}(\partial \mathcal{X}) = \partial \mathcal{Y} \), which holds if \( J_B[\mathbf{m}, \mathbf{b}] = 0 \). This is equivalent, under some restrictions, to \( \mathbf{m}(\mathcal{X}) = \mathcal{Y} \), stating that all the light from the source arrives at the target; see [47]. We combine the functionals \( J_I \) and \( J_B \) by a weighted average as

\begin{equation}
J[\mathbf{m}, \mathbf{P}, \mathbf{b}] = \alpha J_I[\mathbf{m}, \mathbf{P}] + (1 - \alpha) J_B[\mathbf{m}, \mathbf{b}],
\end{equation}

with \( 0 < \alpha < 1 \).
To compute $u_1 = u_1(x)$ from $m$, we use (2.24) and minimize the functional,

\begin{equation}
I[u_1, m] = \frac{1}{2} \int_X |\nabla_x H(x, m, u_1) + H_z(x, m, u_1) \nabla u_1|^2 \, dx,
\end{equation}

detailed in subsection 3.1, where $| \cdot |$ denotes the $L_2$-norm.

We use initial guesses $m^0$ and $u_0^1$, specified shortly, and mixed Hessian matrix $C(\cdot, m^0, u_0^1)$. Let $n = 0$, and compute

\begin{align}
(3.5a) & \quad b^{n+1} = \arg\min_{b \in B} J_B[m^n, b], \\
(3.5b) & \quad P^{n+1} = \arg\min_{P \in \mathcal{P}(m^n)} J_I[m^n, P], \\
(3.5c) & \quad m^{n+1} = \arg\min_{m \in \mathcal{M}} J[m, P^{n+1}, b^{n+1}], \\
(3.5d) & \quad u_1^{n+1} = \arg\min_{u_1 \in \mathcal{U}} I[u_1, m^{n+1}],
\end{align}

where the minimization steps are performed over the spaces

\begin{align}
(3.6a) & \quad B = \{ b \in C^1(\partial \mathcal{X})^2 : b(x) \in \partial \mathcal{Y} \}, \\
(3.6b) & \quad \mathcal{P}(m) = \{ P \in C^1(\mathcal{X})^{2 \times 2} : P \text{ SND, } \det(P) = F_1(\cdot, m) \det(C(\cdot, m)) \}, \\
(3.6c) & \quad \mathcal{M} = C^2(\mathcal{X})^2, \\
(3.6d) & \quad \mathcal{U} = C^2(\mathcal{X}).
\end{align}

After each iteration we compute $C(\cdot, m^{n+1}, u_1^{n+1})$.

As initial guess $m^0$ we map the smallest bounding box enclosing $\mathcal{X}$ to the smallest bounding box enclosing $\mathcal{Y}$. Without loss of generality we assume the bounding box of the source $\mathcal{X}$ has rectangular shape $[a_{\min}, a_{\max}] \times [b_{\min}, b_{\max}]$ and the bounding box of the target $\mathcal{Y}$ has rectangular shape $[c_{\min}, c_{\max}] \times [d_{\min}, d_{\max}]$. In order to find a G-convex $u_1$, we specify the initial guess $m^0 = (m_0^0, m_0^1)^T$ as the linear mapping

\begin{align}
(3.7a) & \quad m_1^0 = \frac{x_1 - a_{\min}}{a_{\max} - a_{\min}} c_{\max} + \frac{a_{\max} - x_1}{a_{\max} - a_{\min}} c_{\min}, \\
(3.7b) & \quad m_2^0 = \frac{x_2 - b_{\min}}{b_{\max} - b_{\min}} d_{\max} + \frac{b_{\max} - x_2}{b_{\max} - b_{\min}} d_{\min}.
\end{align}

The corresponding Jacobian matrix $Dm^0$ is diagonal and SPD. We also need to initialize the surface $u_1$. We take $u_1^0(x) = c$, with $c$ a constant, which is a spherical surface for System 1 and a flat surface for System 2.

Using these initial guesses we can show that $\det(P^0) > 0$ since $\det(C(\cdot, m^0, u_1^0)) > 0$ and $\det(Dm^0) > 0$. Moreover, $\text{tr}(P^0) = 1/2 \text{tr}(C(\cdot, m^0, u_1^0)) \text{tr}(Dm^0)$ since we can show that the diagonal elements of $C = D_{x_0} H(x, m^0, u_1^0)$ are equal in this case for both Systems 1 and 2, and $Dm^0$ is a diagonal matrix. Substituting $m^0$ and $u_1^0$ we find that $\text{tr}(C(\cdot, m^0, u_1^0)) \leq 0$ and $\text{tr}(Dm^0) > 0$. Hence, $P^0 = C(\cdot, m^0) Dm^0$ is SND; i.e., we calculate a G-convex pair.

We discretize the source domain $\mathcal{X}$ using a standard rectangular $N_1 \times N_2$ grid for some $N_1, N_2 \in \mathbb{N}$ and introduce $x_{ij} = (x_{1,i}, x_{2,j})$ with

\begin{align}
(3.8a) & \quad x_{1,i} = a_{\min} + (i - 1) h_1, \quad h_1 = \frac{a_{\max} - a_{\min}}{N_1 - 1}, \quad i = 1, \ldots, N_1, \\
(3.8b) & \quad x_{2,j} = b_{\min} + (j - 1) h_2, \quad h_2 = \frac{b_{\max} - b_{\min}}{N_2 - 1}, \quad j = 1, \ldots, N_2.
\end{align}
The minimization steps (3.5a), (3.5b), and (3.5c) are described in detail in [47, 48]. The operations in (3.5a) and (3.5b) are pointwise minimization steps which can be performed very efficiently. In contrast, the minimization step (3.5c) cannot be performed pointwise. Using calculus of variations, we obtain two coupled elliptic equations for the components \( m_1 \) and \( m_2 \) of \( \mathbf{m} \). This gives [56]

\[
\nabla \cdot (C^T C \mathbf{m}) = \nabla \cdot (C^T \mathbf{P}), \quad x \in \mathcal{X},
\]

(3.9a)

\[
(1 - \alpha) \mathbf{m} + \alpha (C^T C \mathbf{m}) \, \mathbf{n} = (1 - \alpha) \mathbf{b} + \alpha \, \mathbf{P} \, \mathbf{n}, \quad x \in \partial \mathcal{X}.
\]

(3.9b)

where \( \mathbf{n} \) is the unit outward normal at the boundary \( \partial \mathcal{X} \). We discretize (3.9) using the finite volume method [48].

Minimization step (3.5d) is the new step required for the generating-function approach. It is explained in detail in the next section. After each iteration we compute the matrix \( C(\cdot, \mathbf{m}^{n+1}, u_1^{n+1}) \). Figure 3 shows a flow chart of the steps in the numerical procedure. The stopping criterion for the iterative procedure is explained in section 4.

3.1. Minimization procedure for \( u_1 \). To find \( u_1 \), we use the implicit relations in (2.24) and (2.25), i.e.,

\[
\nabla_x H^*(x, y) = \nabla_x H(x, y, u_1(x)) + H_z(x, y, u_1(x)) \, \nabla u_1(x) = 0.
\]

(3.10)

We can compute \( u_1 \) by minimizing the functional

\[
I[u_1, y] = \frac{1}{2} \int_{\mathcal{X}} |\nabla_x H(x, y, u_1)|^2 \, dx.
\]

(3.11)

Analogous to the minimization procedure for \( \mathbf{m} \) detailed in [47, 48], we compute the first variation of \( \delta I[u_1, y](v) \) with respect to \( u_1 \) for \( v \in C^2(\mathcal{X}) \) as

\[
\delta I[u_1, y](v) = \lim_{\epsilon \to 0} \frac{1}{\epsilon} \left( I[u_1 + \epsilon v, y] - I[u_1, y] \right) = \frac{d}{d\epsilon} I[u_1 + \epsilon v, y] \bigg|_{\epsilon=0}
\]

\[
= \frac{d}{d\epsilon} \int_{\mathcal{X}} (\nabla_x H(x, y, u_1 + \epsilon v) + H_z(x, y, u_1 + \epsilon v)) \nabla(u_1 + \epsilon v)^2 \, dx \bigg|_{\epsilon=0}
\]

(3.12)

\[
= \int_{\mathcal{X}} \nabla_x H^* \cdot \frac{d}{d\epsilon} \left( \nabla_x H(x, y, u_1 + \epsilon v) + H_z(x, y, u_1 + \epsilon v) \nabla(u_1 + \epsilon v) \right) \bigg|_{\epsilon=0} \, dx.
\]
where \( H^* = H^*(\mathbf{x}, \mathbf{y}) = H(\mathbf{x}, \mathbf{y}, u_1) \). Using Taylor expansions, the short-hand notation \( H = H(\mathbf{x}, \mathbf{y}, u_1) \), and the expression \( O(\varepsilon^2) \) for the higher than first-order terms, we obtain

\[
\delta I[u_1, \mathbf{y}](v) = \int_\mathcal{X} \nabla_x H^* \cdot \frac{d}{dc} \left( \nabla_x H + \varepsilon v \nabla_x H_z 
right|_{c=0} \right) dx \\
+ H_z \nabla u_1 + \varepsilon v H_{zz} \nabla u_1 + \varepsilon H_z \nabla v + O(\varepsilon^2) \\
= \int_\mathcal{X} \nabla_x H^* \cdot \left( v \nabla_x H_z + v H_{zz} \nabla u_1 + H_z \nabla v \right) dx \\
= \int_\mathcal{X} \nabla_x H^* \cdot \left( \nabla_x H_z + H_{zz} \nabla u_1 \right) v dx + \int_\mathcal{X} \nabla_x H^* \cdot H_z \nabla v dx \\
= \int_\mathcal{X} \frac{1}{2} \frac{d}{dz} \left| \nabla_x H^* \right|^2 v dx + \int_\mathcal{X} \nabla_x H^* \cdot H_z \nabla v dx \\
(3.13)
\]

For the second integral, we use Gauss’s theorem and the vector-scalar product rule, i.e.,

\[
\int_\mathcal{X} \mathbf{F} \cdot \nabla v \, dx = \oint_{\partial \mathcal{X}} \mathbf{F} \cdot \mathbf{n} \, ds - \int_\mathcal{X} (\nabla \cdot \mathbf{F}) \, v \, dx,
\]

with \( \mathbf{F} = H_z \nabla_x H^* \). We now set \( \mathbf{y} = \mathbf{m}(\mathbf{x}) \) and let \( \nabla \cdot \) denote the divergence operator with respect to \( \mathbf{x} \), taking into account the dependencies \( \mathbf{y} = \mathbf{m}(\mathbf{x}) \) and \( u_1 = u_1(\mathbf{x}) \) via the chain rule. The gradient \( \nabla_x \) still only works on the first variable of \( H(\mathbf{x}, \mathbf{y}, u_1) \) and \( H^*(\mathbf{x}, \mathbf{y}) \). Hence, we obtain

\[
\delta I[u_1, \mathbf{m}](v) = \int_\mathcal{X} \frac{1}{2} \frac{d}{dz} \left| \nabla_x H^* \right|^2 v - (\nabla \cdot (H_z \nabla_x H^*)) v \, dx \\
+ \oint_{\partial \mathcal{X}} H_z \nabla_x H^* \cdot \mathbf{n} \, ds.
(3.15)
\]

The minimizer is given by \( \delta I[u_1, \mathbf{m}](v) = 0 \) for all \( v \in C^2(\mathcal{X}) \) and results in the BVP

\[
\nabla \cdot (H_z \nabla_x H^*) = \frac{1}{2} \frac{d}{dz} \left| \nabla_x H^* \right|^2, \quad \mathbf{x} \in \mathcal{X}, \\
H_z \nabla_x H^* \cdot \mathbf{n} = 0, \quad \mathbf{x} \in \partial \mathcal{X}.
(3.16a, 3.16b)
\]

Substituting \( \mathbf{m}^{n+1} \) and the function \( H(\mathbf{x}, \mathbf{m}^{n+1}, u_1) \) at iteration \( n \), this is a Neumann problem for \( u_1 \) which has a corresponding discretization matrix with incomplete rank. We calculate a unique solution by prescribing the average value of \( u_1 \) as a constraint which adds an extra row to the discretization matrix. The details are explained in section SM2 of the supplementary materials to this article.

The Neumann problem only has a solution if the compatibility condition is satisfied. Integrating (3.16b) over \( \partial \mathcal{X} \) and using Gauss’s theorem gives

\[
0 = \int_{\partial \mathcal{X}} H_z (\nabla_x H + H_z \nabla u_1) \cdot \mathbf{n} \, ds = \int_{\mathcal{X}} \nabla \cdot (H_z (\nabla_x H + H_z \nabla u_1)) \, dx,
(3.17)
\]

and integrating (3.16a) over \( \mathcal{X} \) this reduces to

\[
\int_{\mathcal{X}} \frac{1}{2} \frac{d}{dz} \left| \nabla_x H + H_z \nabla u_1 \right|^2 \, dx = 0.
(3.18)
\]
4. Numerical results. In this section, we test the generalized least-squares algorithm on two example problems. First, we consider System 1 and compare the performance of the new generating-function approach to the optimal-transport approach in [47, 48, 56]. Second, for System 2 we challenge the numerical algorithm to compute a reflector surface that transforms a parallel beam into a projection of the SIAM logo in the near field. The laptop used for the calculations has an Intel Core i7-7700HQ CPU 2.8 GHz with 32 GB of RAM.

4.1. Comparison cost-function and generating-function approach: Ellipsoidal point-to-far-field lens. For System 1, we compare the cost-function approach to the generating-function approach. In the cost-function approach, we take the log-transformation to derive (2.12) and consider solutions as a \(c\)-convex pair

\[
\forall x \in \mathcal{X} : \quad u_1(x) = \max_{y \in \mathcal{Y}} (u_2(y) - c(x, y)),
\]

\[
\forall y \in \mathcal{Y} : \quad u_2(y) = \min_{x \in \mathcal{X}} (u_1(x) + c(x, y)),
\]

or a \(c\)-concave pair if

\[
\forall x \in \mathcal{X} : \quad u_1(x) = \min_{y \in \mathcal{Y}} (u_2(y) - c(x, y)),
\]

\[
\forall y \in \mathcal{Y} : \quad u_2(y) = \max_{x \in \mathcal{X}} (u_1(x) + c(x, y)),
\]

which resulted in the matrix \(C = C(x, m(x)) = D_{xy}c\), without a dependency on \(u_1\). We use the least-squares procedure as presented in [47, 48]. Note that in these papers we employed a slightly different definition of \(c\)-convexity and \(c\)-concavity by deriving a relation of the form \(u_1(x) + u_2(y) = c(x, y)\). This results in a \(c\)-convex solution defined as a max/max pair and a concave pair defined as a min/min pair. Using the generating-function approach, we consider the generating function in (2.11), the pair (2.22), and the extended algorithm presented in this paper.

We consider an ellipsoidal lens surface

\[
\frac{x^2}{a^2} + \frac{y^2}{b^2} + \frac{z^2}{c^2} = 1,
\]

with given constants \(a \neq 0, b \neq 0, c \neq 0\). By substituting \(x = u_1 s_1, y = u_1 s_2, z = u_1 s_3\) and changing to stereographic coordinates using (2.2), we can derive an expression for the surface \(u_1\) as

\[
u_1(x) = \frac{1 + |x|^2}{2 \sqrt{\left(\frac{x_1}{a}\right)^2 + \left(\frac{x_2}{b}\right)^2 + \left(\frac{x_3}{c}\right)^2}}.
\]

Subsequently calculating \(\nabla u_1\) and using the implicit relation for the mapping in (2.24), we can solve for the mapping \(m(x)\). (The solution is a long and complicated expression and not included here for brevity.) We consider a square source domain \(\mathcal{X} = [-0.5, 0.5]^2\) and choose \(a = 2, b = 1\), and \(c = 1\). Using the mapping \(m(x)\) we compute the target domain \(\mathcal{Y}\) and the corresponding target boundary. We set the right-hand side \(F_1(x, m(x), u_1(x))\) using the expression in (2.34) since the solution for the mapping \(m\) is only an exact solution for a source intensity \(f(x)\) and target intensity \(g(y)\) such that (2.34) is satisfied. For both methods, we use the initial mapping in (3.7) for a \(c\)-convex and \(G\)-convex \(u_1\), respectively. For the generating-function approach, we initialize the surface \(u_1\) to a spherical surface.
To investigate convergence of the numerical algorithm, we introduce the norms

\[ (4.5) \quad \|A\|_{2 \times 2} = \left( \int_{\mathcal{X}} \|A\|^2 \, dx \right)^{1/2}, \quad \|a\|_2 = \left( \int_{\partial \mathcal{X}} |a|^2 \, ds \right)^{1/2}, \]

for \( A \in [C^1(\mathcal{X})]^{2 \times 2} \) and \( a \in [C^1(\partial \mathcal{X})]^2 \), as described in [45]. Let \( J^n_I = J_I[m^n, P^n] \) and \( J^n_B = J_B[m^n, b^n] \). We can derive that

\[ (4.6a) \quad \left| \sqrt{J^{n+1}_I} - \sqrt{J^n_I} \right| \leq \frac{1}{\sqrt{2}} \left( \|C^{n+1} Dm^{n+1} - C^n Dm^n\|_{2 \times 2} + \|P^{n+1} - P^n\|_{2 \times 2} \right) =: c^n_I, \]

\[ (4.6b) \quad \left| \sqrt{J^{n+1}_B} - \sqrt{J^n_B} \right| \leq \frac{1}{\sqrt{2}} \left( \|m^{n+1} - m^n\|_2 + \|b^{n+1} - b^n\|_2 \right) =: c^n_B, \]

i.e., \( c^n_I \) is an upper bound for the change in \( \sqrt{J_I} \) due to updating \( Dm \) and \( P \), and \( c^n_B \) is an upper bound for the change in \( \sqrt{J_B} \) due to changes in \( m \) and \( b \). Figure 4 shows \( J_I \) and \( J_B \) for \( N = 100 \). It also displays the changes in \( C Dm, m|_{\partial \mathcal{X}} \) (\( m \) on the boundary), \( P \), and \( b \). We take the number of boundary points required for minimization step (3.5a) to be \( N_b = 100 \) and weighting parameter \( \alpha = 0.2 \) from previous experiments as in [47]. The functionals \( J_I \) and \( J_B \) reach a plateau at a certain iteration number, while the individual error terms continue to decrease up to machine precision. We use the stopping criterion

\[ (4.7) \quad c^n_I \leq 0.1 \sqrt{J^n_I} \quad \text{and} \quad c^n_B \leq 0.1 \sqrt{J^n_B}, \]

i.e., the change in \( \sqrt{J_I} \) and \( \sqrt{J_B} \) is relatively less than 0.1. Table 1 shows the results for several \( N \times N \) grids with logarithmic least-squares fits. The c-convex, G-convex, and exact surfaces for \( N = 100 \) are plotted in Figure 4b. The surfaces largely overlap and deviate slightly from the exact solution. The number of iterations required increases sublinearly with \( N \). We see that \( J_I \) and \( J_B \) have approximately third- to fourth-order convergence. Using Simpson’s rule we calculated the integral given in the compatibility condition in (3.18). The last row of Table 1 shows nearly second-order convergence.

Figure 5 shows the maximum absolute differences between the computed mapping and surface with the exact solution. We observe almost second-order convergence for both methods, but the G-convex solution is closer to the exact solution for all grid sizes.

Figure 6 shows the calculation times of the minimization procedures for \( P, b, m \) and the computation of \( u_1 \) as a function of \( N = N_1 = N_2 \). The slopes of performed logarithmic least-squares fits are also displayed. The calculation time for the minimization procedure for \( P \) (linear in \( N \)) is better than expected since it is sublinear in the number of grid points. The calculation times for the minimization procedures for \( b \) (linear in \( N \)) and \( m \) (quadratic in \( N \)) are as expected. The calculation time of \( u_1 \) should be at least linear in the number of grid points and thus quadratic in \( N \). For the c-convex solution, \( u_1 \) is only computed once at the end of the iterative procedure. For the G-convex solution, \( u_1 \) is recomputed at each iteration by solving the Neumann problem (3.16). The computation time shows a quadratic growth in \( N \).

For a c-convex solution, the total calculation time for one iteration is approximately proportional to \( N^{1.7} \) (see Figure 6a), and with the number of iterations growing.
Fig. 4. “Ellipsoidal lens” problem: convergence history for $N = 100$ for both methods. We calculate a c-convex solution and G-convex solution $u_1$, respectively, with parameter values $\alpha = 0.2$, $N_0 = 100$. The mapping of the G-convex solution is shown in (a), and the c-convex and G-convex surfaces are shown in (b) together with the exact solution.

sublinearly in $N$ (see Table 1), the total calculation time scales roughly with $N^2$, as displayed in the second row for the c-convex results of Table 1. For a G-convex solution, the total calculation time for one iteration is approximately proportional to $N^{1.8}$ (see Figure 6b), and with the number of iterations scaling as $N^{0.4}$ (see Table 1), the total calculation time scales roughly with $N^2$ as well, as shown in the second row of the G-convex results of Table 1.

In summary, the G-convex solution is more accurate than the c-convex solution, requiring fewer iterations and having approximately the same computation time.

4.2. Reduction in surface calculations. As we increase the grid size $N$ the computation time of $u_1$, i.e., the new minimization step (3.5d), grows most steeply. Since this step could become problematic at large grid sizes, we investigate whether we can reduce the number of updates of the surface $u_1$. We rerun the experiment of the previous section for a G-convex pair considering a grid size of $N \times N = 100 \times 100$, but we do not perform minimization step (3.5d) at every iteration. We increase the
Table 1

Number of iterations, total computation time (in seconds), and residuals in the least-squares algorithm for the “ellipsoidal lens” problem.

<table>
<thead>
<tr>
<th>Grids</th>
<th>20 x 20</th>
<th>40 x 40</th>
<th>60 x 60</th>
<th>80 x 80</th>
<th>100 x 100</th>
<th>Fits</th>
</tr>
</thead>
<tbody>
<tr>
<td>c-convex</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Iterations</td>
<td>211</td>
<td>430</td>
<td>593</td>
<td>720</td>
<td>824</td>
<td>N^0.8</td>
</tr>
<tr>
<td>Time [s]</td>
<td>2</td>
<td>6</td>
<td>16</td>
<td>35</td>
<td>61</td>
<td>N^2.2</td>
</tr>
<tr>
<td>J_I</td>
<td>7.5 x 10^{-4}</td>
<td>5.1 x 10^{-5}</td>
<td>9.9 x 10^{-6}</td>
<td>3.1 x 10^{-6}</td>
<td>1.3 x 10^{-6}</td>
<td>N^{-4.0}</td>
</tr>
<tr>
<td>J_B</td>
<td>3.0 x 10^{-4}</td>
<td>4.3 x 10^{-5}</td>
<td>1.1 x 10^{-5}</td>
<td>3.7 x 10^{-6}</td>
<td>1.6 x 10^{-6}</td>
<td>N^{-3.3}</td>
</tr>
<tr>
<td>G-convex</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Iterations</td>
<td>118</td>
<td>172</td>
<td>200</td>
<td>221</td>
<td>242</td>
<td>N^0.4</td>
</tr>
<tr>
<td>Time [s]</td>
<td>1</td>
<td>3</td>
<td>7</td>
<td>15</td>
<td>28</td>
<td>N^1.8</td>
</tr>
<tr>
<td>J_I</td>
<td>1.7 x 10^{-4}</td>
<td>1.3 x 10^{-5}</td>
<td>2.7 x 10^{-6}</td>
<td>8.9 x 10^{-7}</td>
<td>3.7 x 10^{-7}</td>
<td>N^{-3.8}</td>
</tr>
<tr>
<td>J_B</td>
<td>2.7 x 10^{-5}</td>
<td>2.8 x 10^{-6}</td>
<td>6.7 x 10^{-7}</td>
<td>2.4 x 10^{-7}</td>
<td>1.1 x 10^{-7}</td>
<td>N^{-3.5}</td>
</tr>
<tr>
<td>Compatibility</td>
<td>9.8 x 10^{-5}</td>
<td>2.8 x 10^{-5}</td>
<td>1.4 x 10^{-5}</td>
<td>8.1 x 10^{-6}</td>
<td>5.4 x 10^{-6}</td>
<td>N^{-1.8}</td>
</tr>
</tbody>
</table>

Fig. 5. “Ellipsoidal lens” problem: (a) maximum absolute differences between the components of the final mapping \( \mathbf{m} = (m_1, m_2)^T \) and the exact mapping. (b) Maximum absolute difference between the final surface \( u \) and the exact solution.

period \( T_{u_1} \) of updating \( u_1 \) from every one to every 100 iterations. Figure 7 shows \( J_I, J_B \), and the changes in \( C \mathbf{Dm}, \mathbf{m}|_{\partial X} \) (\( m \) on the boundary), \( \mathbf{P} \), and \( b \) for two example periods. The functionals \( J_I \) and \( J_B \) plateau to approximately the same value. The values of \( J_I \) and \( J_B \) temporarily increase immediately after the iteration when \( u_1 \) is updated. In Table 2, we see that the final values of \( J_I \) and \( J_B \) remain roughly constant and the number of iterations increases slightly. The total computation time decreases significantly when increasing \( T_{u_1} \) from 1 to 20, but further extending the
period to 50 and 100 iterations increases the total computation time again. This is due to an increase in the number of iterations in order to reach the stopping criterion. For all periods, the maximum absolute difference between the computed mapping and the exact solution is approximately $5.2 \times 10^{-4}$ for $m_1$ and $3.6 \times 10^{-4}$ for $m_2$. The maximum absolute difference between $u_1$ with the exact solution is approximately $4.0 \times 10^{-4}$ for all periods. The lowest computation time of 18 seconds is reached at $T_{u_1} = 20$, lower than for the c-convex and G-convex solutions in Table 1 (61 and 28 seconds), while maintaining lower values for $J_I$ and $J_B$ and a higher solution accuracy than the c-convex solution. Hence, we can reduce the computation time by updating the surface $u_1$ less frequently.

4.3. SIAM parallel-to-near-field reflector. We compute a freeform reflector surface that converts the light from a parallel incoming beam into a near-field target illuminance distribution corresponding to a picture.

We consider the square source domain $\mathcal{X} = [-1, 1]^2$ and a uniform light distri-
Fig. 7. “Ellipsoidal lens” problem: convergence history for $N = 100$. We calculate a G-convex solution $u_1$ with parameter values $\alpha = 0.2$, $N_b = 100$. The surface calculation period $T_{u_1}$ is increased from 1 to 100 iterations.

Table 2

<table>
<thead>
<tr>
<th>$T_{u_1}$</th>
<th>1</th>
<th>5</th>
<th>10</th>
<th>20</th>
<th>50</th>
<th>100</th>
</tr>
</thead>
<tbody>
<tr>
<td>Iterations</td>
<td>242</td>
<td>240</td>
<td>239</td>
<td>239</td>
<td>257</td>
<td>293</td>
</tr>
<tr>
<td>Time [s]</td>
<td>27</td>
<td>20</td>
<td>19</td>
<td>18</td>
<td>21</td>
<td>22</td>
</tr>
<tr>
<td>$J_I$</td>
<td>$3.7 \times 10^{-7}$</td>
<td>$3.7 \times 10^{-7}$</td>
<td>$3.7 \times 10^{-7}$</td>
<td>$3.7 \times 10^{-7}$</td>
<td>$3.7 \times 10^{-7}$</td>
<td>$3.8 \times 10^{-7}$</td>
</tr>
<tr>
<td>$J_B$</td>
<td>$1.1 \times 10^{-7}$</td>
<td>$1.1 \times 10^{-7}$</td>
<td>$1.1 \times 10^{-7}$</td>
<td>$1.1 \times 10^{-7}$</td>
<td>$1.1 \times 10^{-7}$</td>
<td>$1.1 \times 10^{-7}$</td>
</tr>
<tr>
<td>Compatibility</td>
<td>$5.4 \times 10^{-6}$</td>
<td>$5.2 \times 10^{-6}$</td>
<td>$5.0 \times 10^{-6}$</td>
<td>$4.2 \times 10^{-6}$</td>
<td>$5.1 \times 10^{-6}$</td>
<td>$1.6 \times 10^{-4}$</td>
</tr>
</tbody>
</table>

bution $f(x) = 1/4$. The reflected rays are projected on a screen in the near field, parallel to the source plane. The required illumination $g(y)$ is derived from the gray scale values of the SIAM logo.

The gray scale values of the picture prescribe the illuminance. However, the conversion from the colored image to gray scale values creates black regions in the target distribution for which $g(y) = 0$. To avoid division by 0 in the right-hand side of (2.37), we increase values of $g(y)$ which are below a threshold of 15% of its maximum value to this threshold. We normalize $g(y)$ by the total target flux, which we determine by dividing the region on the projection screen into quadrants with the pixels of the picture as nodes and approximating the integral of $g(y)$ using the two-dimensional composite trapezoidal rule.

We use the least-squares algorithm to compute the optical map $m$ and the reflector surface. We calculate $u_1$ every 20th iteration ($T_{u_1} = 20$) and use a $500 \times 500$ grid. We use the initial guess $m^0$ given in (3.7) to compute a G-convex $u_1$. The optical map, plotted using a coarsened version of the source grid, the reflector surface, and convergence results are shown in Figure 8.
Subsequently, we validated the resulting reflector image using ray tracing. We traced $1500 \times 1500$ rays with quasi-random positions (quasi–Monte Carlo) from source to near field. We determine the intersection of the rays with the optical surface by constructing a Delaunay triangulation of the surface and using a triangle-ray intersection algorithm proposed in [37]. The resulting target illuminance $g(y)$ is plotted in Figure 8c.

The average computation time per iteration is 2.3 seconds (2.1 seconds without computation of $u_1$ and 5.0 seconds with computation of $u_1$). The total computation time is 11.4 minutes. Using Simpson’s rule, the compatibility integral in (3.18) evaluates to $-2.1 \times 10^{-5}$.

The ray trace image closely resembles the logo, and all letters of the logo are readable. The key difficulty for solving generalized Monge–Ampère equations numerically is to find a method that does not degrade as the source and target densities...
(especially the target density) become close to zero in some places, particularly when the effective support of the target density becomes nonconvex. The numerical method in this paper does not produce discontinuous mappings, but setting the background brightness to a low value suffices to make the smallest letters readable.

5. Conclusions. In this paper, we introduced a numerical procedure to solve generated Jacobian equations and applied it to two optical systems: System 1 is a point-to-far-field lens, and System 2 is a parallel-to-near-field reflector. First, we used Hamilton’s characteristic functions to derive the generating function and showed that the optical mapping can be derived by considering a G-convex solution for the location of the optical surface $u_1$. Combining the optical mapping with energy conservation resulted in a GJE, a.k.a. the generalized Monge–Ampère equation.

We extended a least-squares method priorly used for optimal-transport problems to a generating-function framework. The difference with the optimal-transport approach is the additional step during the iterative procedure to compute the optical surface. We compared the performance of the algorithm to the previous optimal-transport approach for an exact solution given as an ellipsoidal lens surface. We concluded that the algorithm performs better in accuracy and similar in computation time. The computation time can, however, be reduced by updating the optical surface less frequently. We also tested the algorithm for a reflector converting the light of a parallel source into a picture on a screen in the near field and verified the solution via ray tracing. This problem cannot be cast as an optimal-transport problem.

By formulating the generalized Monge–Ampère equation as a generated Jacobian equation and using G-convexity theory to find a solution, we extended the applicability of the least-squares procedure to a much wider range of optical systems. In future research, we aim to apply the method to problems involving extended light sources and double freeform surfaces, and to take into account optical phenomena such as scattering and aberrations. Moreover, we would like to extend the method to other coordinate systems and explore the applicability of the algorithm in other fields of science, engineering, and/or economics.

REFERENCES


